

(19) 日本国特許庁(JP)

(12) 公開特許公報(A)

(11) 特許出願公開番号

特開2014-28063

(P2014-28063A)

(43) 公開日 平成26年2月13日(2014.2.13)

|                               |                      |             |
|-------------------------------|----------------------|-------------|
| (51) Int.Cl.                  | F I                  | テーマコード (参考) |
| <b>A 6 3 F</b> 7/02 (2006.01) | A 6 3 F 7/02 3 0 4 Z | 2 C 0 8 2   |
| <b>A 6 3 F</b> 5/04 (2006.01) | A 6 3 F 7/02 3 3 4   | 2 C 0 8 8   |
|                               | A 6 3 F 5/04 5 1 2 C |             |
|                               | A 6 3 F 5/04 5 1 2 Z |             |

審査請求 未請求 請求項の数 3 O L (全 486 頁)

|           |                              |          |                     |
|-----------|------------------------------|----------|---------------------|
| (21) 出願番号 | 特願2012-170435 (P2012-170435) | (71) 出願人 | 597044139           |
| (22) 出願日  | 平成24年7月31日 (2012.7.31)       |          | 株式会社大都技研            |
|           |                              | (74) 代理人 | 100101214           |
|           |                              |          | 弁理士 森岡 正樹           |
|           |                              | (72) 発明者 | 岡田 康弘               |
|           |                              |          | 東京都台東区東上野一丁目1番14号 株 |
|           |                              |          | 式会社大都技研内            |
|           |                              | (72) 発明者 | 斉藤 健                |
|           |                              |          | 東京都台東区東上野一丁目1番14号 株 |
|           |                              |          | 式会社大都技研内            |

最終頁に続く

(54) 【発明の名称】 遊技台

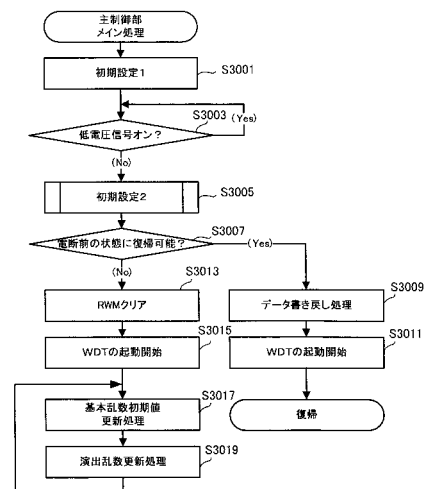
(57) 【要約】

【課題】 安定した遊技制御をおこなうことができる遊技台を提供する。

【解決手段】

リセット回路は、開始指示を受けて経過時間の計測を開始し、再開始指示を受けて経過時間の計測を再開し、経過時間が所定時間となったらCPUに対して復帰指示を行う。CPUは、遊技制御処理を最初から実行し、判定処理によって所定の記憶領域に所定の電断データが記憶されていると判定された場合には、復帰データを用いた復帰処理を実行することが可能であり、判定処理によって所定の記憶領域に所定の電断データが記憶されていないと判定された場合には、復帰処理を実行することが不可能であり、再開始指示を行う処理を割込み制御において実行し、少なくとも復帰処理を実行した後であり、かつ最初の割込み制御が実行される前に、リセット回路に対して開始指示を行う処理を実行する。

【選択図】 図 8 8



## 【特許請求の範囲】

## 【請求項 1】

複数の遊技制御処理を実行するCPUと、  
前記複数の遊技制御処理を正常に復帰させるためのリセット回路と、  
遊技に関するデータを記憶するRWMと、備え、  
前記RWMは、  
前記複数の遊技制御処理を正常に復帰させるための復帰データを記憶し、  
電源ステータスの情報を記憶する所定の記憶領域を有するものであり、  
前記リセット回路は、  
開始指示を受けたことに基づいて経過時間の計測を開始し、  
再開指示を受けたことに基づいて計測している該経過時間を初期化するとともに、初期化された該経過時間の計測を再び開始し、  
該経過時間が予め定められた所定の時間となったか否かを判定し、  
該経過時間が該所定の時間となったと判定したことに基づいて前記CPUに対して復帰指示を行うものであり、  
前記CPUは、  
前記複数の遊技制御処理それぞれをメイン制御および所定の周期ごとに実行される割り込み制御のうちの少なくともいずれか一方の制御において実行し、  
電源が遮断される場合において該電源が遮断されることを示す所定のデータを前記所定の記憶領域に記憶させる処理を実行し、  
前記復帰指示を受けた場合には該所定のデータを該所定の記憶領域に記憶させる処理を実行せず、  
該電源が投入された場合または前記復帰指示を受けた場合において該複数の遊技制御処理を最初から実行し、  
該複数の遊技制御処理を最初から実行する場合において該所定の記憶領域に該所定のデータが記憶されているか否かを判定する判定処理を実行し、  
該判定処理によって該所定の記憶領域に該所定のデータが記憶されていると判定された場合には、前記復帰データを用いた復帰処理を実行することが可能であり  
該判定処理によって該所定の記憶領域に該所定のデータが記憶されていないと判定された場合には、該復帰処理を実行することが不可能であり、  
前記再開指示を行う処理を前記割り込み制御において実行し、  
少なくとも該復帰処理を実行した後であり、かつ最初の該割り込み制御が実行される前に、前記リセット回路に対して前記開始指示を行う処理を実行するものであることを特徴とする遊技台。

## 【請求項 2】

請求項 1 に記載の遊技台であって、  
前記CPU、前記リセット回路および前記RWMを少なくとも有するマイクロプロセッサを備え、  
前記マイクロプロセッサは、  
少なくとも前記リセット回路によって前記経過時間が前記所定の時間となったと判定されたことに基づいて、セキュリティチェックが行われるセキュリティモードに所定の期間に亘り滞在した後、前記CPUによって前記複数の遊技制御処理が実行されるユーザーモードへ移行するものであることを特徴とする遊技台。

## 【請求項 3】

複数の遊技制御処理を実行するCPUと、  
前記複数の遊技制御処理を正常に復帰させるためのリセット回路と、  
遊技に関するデータを記憶するRWMと、備え、  
前記RWMは、  
前記複数の遊技制御処理を正常に復帰させるための復帰データを記憶し、  
電源ステータスの情報を記憶する所定の記憶領域を有するものであり、

前記リセット回路は、開始指示を受けたことに基づいて経過時間の計測を開始し、  
再開指示を受けたことに基づいて計測している該経過時間を初期化するとともに、初期化された該経過時間の計測を再び開始し、  
該経過時間が予め定められた所定の時間となったか否かを判定し、  
該経過時間が該所定の時間となったと判定したことに基づいて前記CPUに対して復帰指示を行うものであり、  
前記CPUは、  
前記複数の遊技制御処理それぞれをメイン制御および所定の周期ごとに実行される割込み制御のうちの少なくともいずれか一方の制御において実行し、  
電源が遮断される場合において該電源が遮断されることを示す所定のデータを前記所定の記憶領域に記憶させる処理を実行し、  
前記復帰指示を受けた場合には該所定のデータを該所定の記憶領域に記憶させる処理を実行せず、  
該電源が投入された場合または前記復帰指示を受けた場合において該複数の遊技制御処理を最初から実行し、  
該複数の遊技制御処理を最初から実行する場合において該所定の記憶領域に該所定のデータが記憶されているか否かを判定する判定処理を実行し、  
該判定処理によって該所定の記憶領域に該所定のデータが記憶されていると判定された場合には、前記復帰データを用いた復帰処理を実行することが可能であり、  
該判定処理によって該所定の記憶領域に該所定のデータが記憶されていないと判定された場合には、前記復帰データを初期化する初期化処理を実行し、  
前記再開指示を行う処理を前記割込み制御において実行し、  
少なくとも該初期化処理を実行した後であり、かつ最初の該割込み制御が実行される前に、前記リセット回路に対して前記開始指示を行う処理を実行するものであることを特徴とする遊技台。

10

20

#### 【発明の詳細な説明】

#### 【技術分野】

#### 【0001】

本発明は、回胴遊技機（スロットマシン）や弾球遊技機（パチンコ機）に代表される遊技台に関する。

30

#### 【背景技術】

#### 【0002】

従来の遊技台（例えば、スロットマシン、パチンコ機）は、図柄表示部の有効入賞ラインに沿って所定の図柄の組合せを停止表示させることで、遊技者が所定の利益を獲得できるように構成されていたり、遊技領域に設けられた所定の入賞口に遊技球を進入させることで、遊技者が所定の利益を獲得できるように構成されている。

このような遊技台では、遊技の興味が高められなければ遊技者に飽きられることから遊技制御が複雑になる傾向にある。さらに、遊技者による不正な行為、静電気ノイズなどの外乱、プログラムのコーディングミスなど様々な要因で、遊技制御が不安定になる場合があるが、このような問題点を解決するために、リセット受信からソフト起動までの時間をランダム化させる遊技台が提案されている（例えば、特許文献1参照）。

40

#### 【先行技術文献】

#### 【特許文献】

#### 【0003】

【特許文献1】特開2009-34162号公報

#### 【発明の概要】

#### 【発明が解決しようとする課題】

#### 【0004】

しかしながら、特許文献1記載の遊技台では、電源投入時の不正行為を有効に阻止する

50

ことができるものの、他のタイミングや要因については更なる改良が求められている。

【0005】

本発明は、このような従来の問題点を解決するためになされたものであって、安定した遊技制御をおこなうことができる遊技台を提供することを目的とする。

【課題を解決するための手段】

【0006】

上記目的は、

複数の遊技制御処理を実行するCPUと、

前記複数の遊技制御処理を正常に復帰させるためのリセット回路と、

遊技に関するデータを記憶するRWMと、備え、

前記RWMは、

前記複数の遊技制御処理を正常に復帰させるための復帰データを記憶し、

電源ステータスの情報を記憶する所定の記憶領域を有するものであり、

前記リセット回路は、

開始指示を受けたことに基づいて経過時間の計測を開始し、

再開指示を受けたことに基づいて計測している該経過時間を初期化するとともに、初期化された該経過時間の計測を再び開始し、

該経過時間が予め定められた所定の時間となったか否かを判定し、

該経過時間が該所定の時間となったと判定したことに基づいて前記CPUに対して復帰指示を行うものであり、

前記CPUは、

前記複数の遊技制御処理それぞれをメイン制御および所定の周期ごとに実行される割込み制御のうちの少なくともいずれか一方の制御において実行し、

電源が遮断される場合において該電源が遮断されることを示す所定のデータを前記所定の記憶領域に記憶させる処理を実行し、

前記復帰指示を受けた場合には該所定のデータを該所定の記憶領域に記憶させる処理を実行せず、

該電源が投入された場合または前記復帰指示を受けた場合において該複数の遊技制御処理を最初から実行し、

該複数の遊技制御処理を最初から実行する場合において該所定の記憶領域に該所定のデータが記憶されているか否かを判定する判定処理を実行し、

該判定処理によって該所定の記憶領域に該所定のデータが記憶されていると判定された場合には、前記復帰データを用いた復帰処理を実行することが可能であり

該判定処理によって該所定の記憶領域に該所定のデータが記憶されていないと判定された場合には、該復帰処理を実行することが不可能であり、

前記再開指示を行う処理を前記割込み制御において実行し、

少なくとも該復帰処理を実行した後であり、かつ最初の該割込み制御が実行される前に、前記リセット回路に対して前記開始指示を行う処理を実行するものであることを特徴とする遊技台

によって達成される。

【発明の効果】

【0007】

本発明に係る遊技台によれば、安定した遊技制御をおこなうことができる。

【図面の簡単な説明】

【0008】

【図1】パチンコ機を正面側（遊技者側）から見た外観斜視図である。

【図2】パチンコ機を背面側から見た外観図である。

【図3】遊技盤を正面から見た略示正面図である。

【図4】制御部の回路ブロック図を示したものである。

【図5】（a）特図の停止図柄態様の一例を示したものである。（b）装飾図柄の停止図

10

20

30

40

50

柄態様の一例を示したものである。(c) 普図の停止表示図柄の一例を示したものである。

【図6】主制御部300から基本回路302の主要部のみを抜き出して示した図である。

【図7】(a) CPU304のメモリマップを示した図である。(b) (a)に示すメモリマップのうちのRWM領域の詳細を示した図である。(c) CPU304のI/Oマップを示した図である。

【図8】初期化処理の流れを示すフローチャートである。

【図9】システムリセット後の起動処理のタイムチャートである。

【図10】主制御部メイン処理の流れを示すフローチャートである。

【図11】(a) 初期設定1で行うTレジチェック処理の流れを示すフローチャートである。(b) Tレジチェック処理のプログラムリストの一例である。 10

【図12】(a) 変形例に係るTレジチェック処理の流れを示すフローチャートである。(b) 変形例に係るTレジチェック処理のプログラムリストの一例である。

【図13】図11(b)のプログラムリストからLD命令のみを抜き出して示した図である。

【図14】主制御部タイマ割込処理の流れを示すフローチャートである。

【図15】(a) 第1副制御部のCPUが実行するメイン処理のフローチャートである。(b) 第1副制御部のコマンド受信割込み処理のフローチャートである。(c) 第1副制御部のタイマ割込処理のフローチャートである。(d) 第1副制御部の画像制御処理のフローチャートである。 20

【図16】(a) 第2副制御部500のCPU504が実行するメイン処理のフローチャートである。(b) 第2副制御部500のコマンド受信割込処理のフローチャートである。(c) 第2副制御部500のタイマ割込処理のフローチャートである。

【図17】ROM306に記憶されるデータの一例を示した図である。

【図18】主制御部300の命令データの構成の一例を示した図である。

【図19】(a) 主制御部300の命令データの上位ビットと下位ビットを示した図である。(b) 命令データテーブルの一例を示した図である。

【図20】CALL命令によるアドレスの移動の一例を概念的に示したものである。

【図21】CALL命令の命令データと補足データの一例を示した図である。

【図22】(a) EXESUB命令の構成の一例を示した図である。(b) 図19(b)の空き領域に対してEXESUB命令を割り当てた命令データテーブルを示した図である。 30

【図23】EXESUB命令によるアドレスの移動の一例を概念的に示した図である。

【図24】EXESUB命令によるサブルーチンの先頭アドレスの呼び出しの一例を示したものである。

【図25】EXESUB命令の命令データと補足データの一例を示した図である。

【図26】主制御部タイマ割込処理の制御プログラムの一例を示したプログラムリストである。

【図27】ROM領域が、EXESUB命令によって呼び出すことのできる領域よりも、制御プログラムデータが記憶されているROM制御領域が広くなるように構成されている場合を示した図である。 40

【図28】(a) (b) EXESUB命令(EXESUBmn)によって呼び出すことのできるサブルーチンの先頭アドレスの領域を拡張した場合の一例を示した図である。

【図29】(a) 特図2状態更新処理および特図1状態更新処理の制御プログラムの一部を抜き出したものである。(b) 普図状態更新処理の制御プログラムの一部を抜き出したものである。(c) 従来のメモリ読み出し処理の一部を抜き出したものである。(d) RAMエリア(RWM領域)に設けた変数領域の一部を示した図である。(e) 本実施形態に係るメモリ読み出し処理の一部を抜き出したものである。

【図30】図29(c)に示す従来のメモリ読み出し処理の機械語と、図29(e)に示す本実施形態に係るメモリ読み出し処理の機械語と、を比較した図である。 50

【図 3 1】特図 1 関連抽選処理の流れを示すフローチャートである。

【図 3 2】(a) テーブル選択用テーブルの一例を示した図である。(b) 第 1 変動パターン選択テーブルの一例を示した図である。(c) 第 2 変動パターン選択テーブルの一例を示した図である。(d) RWM 領域のアドレス F 0 4 0 H ~ F 0 4 2 H に対応する R A M 3 0 8 の記憶領域に設けられる保留数記憶領域、乱数 1 記憶領域、乱数 2 記憶領域を示した図である。

【図 3 3】主制御部 3 0 0 の R O M 3 0 6 に記憶されるデータテーブルの定義の一部と、R A M 3 0 8 に設けられる変数格納領域の定義の一部を示したプログラムリストの一例である。

【図 3 4】特別図柄変動時間抽選処理の流れを示すフローチャートである。

10

【図 3 5】特別図柄変動時間抽選処理のプログラムリストの一例である。

【図 3 6】制御部 3 0 0 (の C P U 3 0 4) が備える命令の一部と、その説明を示した図である。

【図 3 7】(a) 主制御部 3 0 0 の命令データの上位ビットと下位ビットを示した図である。(b) 命令データテーブルの一例を示した図である。

【図 3 8】テーブル選択用テーブル、第 1 変動パターン選択テーブル、および第 2 変動パターン選択テーブルの変形例を示した図である。

【図 3 9】変形例に係る特別図柄変動時間抽選処理の流れを示すフローチャートである。

【図 4 0】(a) R O M 3 0 6 に記憶される初期設定データテーブルの一例を示した図である。(b) 初期設定処理後の R A M 3 0 8 の記憶領域の一部を示した図である。(c) 初期設定処理の流れを示すフローチャートである。

20

【図 4 1】初期設定処理のプログラムリストの一例である。

【図 4 2】(a) 主制御部 3 0 0 が備える第 2 特殊命令の一部を示した図である。(b) 第 2 特殊命令を実行する前後のフラグレジスタの状態を示した図である。

【図 4 3】C P J R 命令を用いた処理の一例を示した図である。

【図 4 4】(a) 従来の命令を用いたプログラムの一例を示した図である。(b) C P J R 命令を用いたプログラムの一例である。

【図 4 5】C P J R 命令を用いた他の処理の一例を示した図である。

【図 4 6】処理 V を含む、C P J R 命令を用いた他の処理の一例を示した図である。

【図 4 7】C P R T 命令を用いた処理の一例を示した図である。

30

【図 4 8】(a) C P R T 命令を用いたプログラムの一例である。(b) 従来の命令を用いたプログラムの一例を示した図である。

【図 4 9】主制御部 3 0 0 が備える第 3 特殊命令の一部を示した図である。

【図 5 0】「R E S m Z n、r」命令の具体例を示した図である。

【図 5 1】「R E S m Z n、(r r)」命令の具体例を示した図である。

【図 5 2】主制御部 3 0 0 の命令マップのうち、R E S 命令に該当する部分を抜き出して示した図である。

【図 5 3】変形例 1 に係る R E S 命令を示した図である。

【図 5 4】変形例 1 に係る R E S 命令の具体例を示した図である。

【図 5 5】変形例 1 に係る R E S 命令を用いたプログラムの一例を示した図である。

40

【図 5 6】変形例 2 に係る R E S 命令を示した図である。

【図 5 7】変形例 2 に係る R E S 命令の具体例を示した図である。

【図 5 8】変形例 2 に係る R E S 命令の具体例を示した図である。

【図 5 9】主制御部 3 0 0 の命令マップのうち、変形例 2 に係る R E S 命令に該当する部分を抜き出して示した図である。

【図 6 0】R E S 命令の他の変形例を示した図である。

【図 6 1】(a) 所定ビットのアクセスを禁止しない R E S 命令を備えた場合の命令マップと、所定ビットのアクセスを禁止する R E S 命令を備えた場合の命令マップを示した図である。(b) R E S 命令の第 1 オペランドの数値の大小と、R E S 命令の命令マップの並び順を異ならせた例を示した図である。

50

【図 6 2】 (a) ユーザプログラムの一部であるプログラム起動設定処理の一例を示した図である。(b) ユーザプログラムの一部である電源投入時設定処理の一例を示した図である。

【図 6 3】 主制御部タイマ割込処理に相当するユーザプログラムの一部を示した図である。

【図 6 4】 主制御部 3 0 0 が備える第 4 特殊命令の内容と、主制御部 3 0 0 の R O M 領域を示した図である。

【図 6 5】 主制御部 3 0 0 の命令マップのうち、第 4 特殊命令に係る R S T 命令に該当する部分を抜き出して示した図である。

【図 6 6】 スロットマシンを正面側（遊技者側）から見た外観斜視図である。

10

【図 6 7】 前面扉を開けた状態のスロットマシンを示す正面図である。

【図 6 8】 制御部の回路ブロック図を示したものである。

【図 6 9】 (a) 各リール（左リール、中リール、右リール）に施される図柄の配列を平面的に展開して示した図である。(b) 入賞役（作動役を含む）の種類、各入賞役に対応する図柄組合せ、各入賞役の作動または払出を示した図である。

【図 7 0】 主制御部メイン処理の流れを示すフローチャートである。

【図 7 1】 主制御部タイマ割込処理の流れを示すフローチャートである。

【図 7 2】 (a) 第 1 副制御部の C P U 1 4 0 4 が実行するメイン処理のフローチャートである。(b) 第 1 副制御部のコマンド受信割込処理のフローチャートである。(c) 第 1 副制御部のタイマ割込処理のフローチャートである。

20

【図 7 3】 (a) 第 2 副制御部の C P U 1 5 0 4 が実行するメイン処理のフローチャートである。(b) 第 2 副制御部のコマンド受信割込処理のフローチャートである。(c) 第 2 副制御部のタイマ割込処理のフローチャートである。(d) 第 2 副制御部の画像制御処理のフローチャートである。

【図 7 4】 本発明の他の適用例を示した図である。

【図 7 5】 マイクロプロセッサ 3 0 0 0 を用いた場合の基本回路 3 0 2 の一構成例を示す図である。

【図 7 6】 リセットの流れを示すフローチャートである。

【図 7 7】 図 7 5 に示す乱数生成回路 3 1 8 の内部構成図である。

【図 7 8】 図 7 7 に示すノイズフィルタ 3 1 8 5 による処理の一例を示す図である。

30

【図 7 9】 図 7 7 に示す乱数更新回路 3 1 8 3 の詳細を示す図である。

【図 8 0】 乱数生成範囲の最大値が設定されていない場合において出力される乱数の範囲を示す図である。

【図 8 1】 図 8 0 とは異なる最大値が設定された場合に出力される乱数の範囲を示す図である。

【図 8 2】 最大値と最小値を設定した乱数生成範囲において、乱数を取得することができる範囲を示す図である。

【図 8 3】 本実施形態の遊技台で使用される乱数の導出源を表で示す図である。

【図 8 4】 割込み制御回路 3 1 0 0 の内部レジスタ 3 1 0 1 に用意された第 1 内部情報レジスタを説明するための図である。

40

【図 8 5】 周波数監視回路 3 1 8 2 における異常の検出例を示す図である。

【図 8 6】 乱数監視回路 3 1 8 4 における異常の検出例を示す図である。

【図 8 7】 乱数が一巡する周期とタイマ割込みの周期を比較した図である。

【図 8 8】 主制御部メイン処理の流れを示すフローチャートである。

【図 8 9】 主制御部メイン処理における初期設定 2 の流れを示すフローチャートである。

【図 9 0】 ステップ S 1 0 5 3 における乱数生成回路初期設定処理の流れを示すフローチャートである。

【図 9 1】 主制御部タイマ割込処理の流れを示すフローチャートである。

【図 9 2】 (a) 普図関連抽選処理の流れを示すフローチャートである。(b) 普図抽選テーブルを示す図である。

50

【図 9 3】特図関連抽選処理の流れを示すフローチャートである。

【図 9 4】(a) 特図抽選テーブルを示す図である。(b) 停止図柄抽選テーブルを示す図である。

【図 9 5】デバイス監視処理の流れを示すフローチャートである。

【図 9 6】ステップ S 2 1 7 における入賞受付処理のうち、特図当選乱数および普図当選乱数を取得する処理の流れを示した図である。

【図 9 7】(a) 電源オフの場合の動作を示す図である。(b) 瞬断の場合の動作を示す図である。

【図 9 8】乱数生成範囲の問題点を示す図である。

【図 9 9】図 7 9 で説明した乱数更新回路 3 1 8 3 の変形例を示す図である。

10

【図 1 0 0】払出制御部メイン処理の流れを示すフローチャートである。

【図 1 0 1】払出制御部タイマ割り込み処理の流れを示すフローチャートである。

【図 1 0 2】割込み制御回路 3 1 0 0 の内部レジスタ 3 1 0 1 に用意された第 2 内部情報レジスタを説明するための図である。

【図 1 0 3】割込み制御回路 3 1 0 0 の内部レジスタ 3 1 0 1 に用意された第 3 内部情報レジスタを説明するための図である。

【図 1 0 4】割込み制御回路 3 1 0 0 の内部レジスタ 3 1 0 1 に用意された第 4 内部情報レジスタを説明するための図である。

【図 1 0 5】システムリセットの流れを乱数更新との関係を含めて示す図である。

【図 1 0 6】(a) 遊技制御プログラムの初期設定で実行される処理を模式的に示した図である。(b) (a) の処理の概念を説明するための図である。

20

【図 1 0 7】図 7 5 に示すマイクロプロセッサ 3 0 0 0 のリセット制御回路 3 1 4 の変形例を示した図である。

【図 1 0 8】主制御部 3 0 0 と第 1 副制御部 4 0 0 とのうち主制御部 3 0 0 のみで W D T 3 1 4 1 のタイムアウトが生じた場合の例を示すタイミングチャートである。

【図 1 0 9】(a) 主制御部の構成の第 1 変形例を示した図である。(b) 主制御部の構成の第 2 変形例を示した図である。(c) ランダム遅延回路 3 1 7、電圧監視回路 3 3 8、および W D T 3 1 4 を有して構成されたりセット回路の構成例を示した図である。

【図 1 1 0】(a) 図 1 0 9 (c) の第 1 変形例を示した図である。(b) 図 1 0 9 (c) の第 2 変形例を示した図である。

30

【図 1 1 1】(a) 図 1 0 9 (c) に示すリセット回路の第 3 変形例を示した図である。

(b) C P U 3 0 4 に接続される各種 I C の一部を示した図である。

【図 1 1 2】図 1 1 1 (b) に示すリセット信号出力端子 X R S T O の接続の第 1 変形例を示した図である。

【図 1 1 3】図 4 を用いて説明した基本回路 3 0 2 と、この基本回路 3 0 2 に接続される各種 I C の一部を示した図である。

【図 1 1 4】(a) 固定延長時間の設定例を示したものである。(b) ランダム延長時間の設定例を示したものである。

【図 1 1 5】W D T 3 1 4 のタイムアウト時間の設定例を示したものである。

【図 1 1 6】(a) ~ (d) リセット出力端子 X S R S T O 端子の出力信号の変化と、セキュリティモードの状態変化を示した図である。

40

【図 1 1 7】(a)、(b) 固定延長処理の終了タイミングと、リセット出力端子 X R S T O から H レベルの信号を出力するタイミングが一致するが、ランダム延長時間が異なる例を示した図である。(c)、(d) 固定延長処理の終了タイミングと、リセット出力端子 X R S T O から H レベルの信号を出力するタイミングが一致するが、固定延長時間が異なる例を示した図である。

【図 1 1 8】固定延長処理を契機にリセット出力信号を変化させる例を示した図である。

【図 1 1 9】アドレス信号、データ信号、制御信号、およびリセット出力信号の変化と、セキュリティモードの状態変化を示した図である。

【図 1 2 0】(a) 外部リセット後の他の出力信号の状態変化を示した図である。(b)

50



内部リセット後の他の出力信号の状態変化を示した図である。

【図 1 2 1】(a) 外部リセット後の他の出力信号の状態変化の他の例を示した図である。  
(b) 内部リセット後の他の出力信号の状態変化の他の例を示した図である。

【図 1 2 2】(a) 外部リセット後のセキュリティモードの開始タイミングの例を示した図である。  
(b) 内部リセット後のセキュリティモードの開始タイミングの例を示した図である。

【図 1 2 3】本発明の基本概念を示した図である。

【図 1 2 4】(a) 図 2 を用いて説明した電源基板 1 8 2、払出基板 1 7 0、および主基板 1 5 6 の接続例を示した図である。  
(b) (a) に対応する従来の接続態様を示した図である。

【図 1 2 5】変形例に係る主制御部メイン処理の流れを示すフローチャートであり、上記図 1 0 に対応するフローチャートである。

【図 1 2 6】主制御部メイン処理における初期設定 2 の流れを示すフローチャートである。

【図 1 2 7】本発明に係る所定の出力端子の信号出力の一例を説明するための図である。

【図 1 2 8】本発明に係る所定の出力端子の信号出力の一例を説明するための図である。

【図 1 2 9】ユーザリセット時におけるシステムクロック信号、アドレス信号、データ信号、制御信号、およびリセット出力信号の変化を示した図である。

【図 1 3 0】主制御部における電断時処理の流れを示すフローチャートである。

【図 1 3 1】図 7 0 のステップ S 2 1 0 1 における初期設定処理のフローチャートである。

【図 1 3 2】図 7 5 に示すタイマ回路 3 1 1 の構成を詳しく示したブロック図である。

【図 1 3 3】図 7 5 に示すカウンタ回路 3 1 2 の構成を詳しく示したブロック図である。

【図 1 3 4】制御レジスタ 3 1 2 1 を詳しく説明するための図である。

【図 1 3 5】図 7 (a) に示すメモリマップ上における内蔵 R A M エリアや内蔵レジスタエリアの読み込みや書き込みタイミングを説明するためのタイミングチャートである。

【図 1 3 6】図 7 (b) に示す I / O マップ上における内蔵レジスタエリアの読み込みや書き込みタイミングを説明するためのタイミングチャートである。

【図 1 3 7】図 7 5 に示すマイクロプロセッサ 3 0 0 0 内部におけるカウンタ回路 3 1 2 への信号の流れを内部のみ示す図である。

【図 1 3 8】カウンタ回路 0 を不定期な更新とする例を示した図である。

【図 1 3 9】カウンタ回路 0 をカウンタ回路 1 の更新ソースとする変形例 1 を示した図である。

【図 1 4 0】カウンタ回路 0 とカウンタ回路 1 で 2 バイトカウンタとして機能させる変形例 2 を示した図である。

【図 1 4 1】カウンタ回路 0 の更新ソースをカウンタ回路 1 の更新ソースとする変 3 1 例を示した図である。

【図 1 4 2】カウンタ回路 0 とカウンタ回路 1 を相互にカスケード接続する変形例 4 を示す図である。

【図 1 4 3】マイクロプロセッサ 3 0 0 0 (基本回路) の外部における外部メモリの読み込みや書き込みチップセレクトタイミングを説明するためのタイミングチャートである。

【図 1 4 4】外部 I / O の読み込みや書き込みチップセレクトタイミングを説明するためのタイミングチャートである。

【図 1 4 5】図 7 5 に示すマイクロプロセッサ 3 0 0 0 内部におけるカウンタ回路 3 1 2 への信号の流れを内部のみ示す図である。

【図 1 4 6】主制御部タイマ割込処理の流れを示すフローチャートである。

【図 1 4 7】(a) 第 1 副制御部の C P U が実行するメイン処理のフローチャートである。  
(b) 第 1 副制御部のコマンド受信割込み処理のフローチャートである。  
(c) 第 1 副制御部のタイマ割込処理のフローチャートである。  
(d) 第 1 副制御部の画像制御処理のフローチャートである。

10

20

30

40

50

【図148】(a)第2副制御部500のCPU504が実行するメイン処理のフローチャートである。(b)第2副制御部500のコマンド受信割込処理のフローチャートである。(c)第2副制御部500のタイマ割込処理のフローチャートである。

【図149】コマンド設定送信処理の流れを示すフローチャートである。

【図150】大当り中用コマンド送信処理の流れを示すフローチャートである。

【図151】ブロック転送処理の流れを示すフローチャートである。

【図152】(a)～(d)大当り遊技中に参照されるコマンドテーブルの一例を示した図である。

【図153】本発明の特徴部分のみを抜き出して示したフローチャートである。

【図154】(a)、(b)本発明の特徴部分を上記図153と異なる視点で示したフローチャートである。

10

【図155】本発明の特徴部分に対応する従来技術の処理の流れを示したフローチャートであり、上記図153に対応するフローチャートである。

【図156】(a)本発明に係る大当り中用コマンド送信処理においてブロック転送を行う場合の所要時間を示した図であり。(b)従来技術に係る大当り中用コマンド送信処理においてブロック転送を行う場合の所要時間を示した図である。

【図157】補正時間待機処理の配置場所の例を示したフローチャートである。

【図158】割込み処理の流れを模式的に示した図である。

【図159】コマンド設定送信処理の流れを示すフローチャートである。

【図160】(a)～(f)遊技中に参照されるコマンドテーブルの一例を示した図である。

20

【図161】主制御部300が備える第5特殊命令の一部と、その説明を示した図である。

【図162】第5特殊命令のバイト数とステート数を比較した図である。

【図163】ZフラグやSZフラグを変化させる命令の一例を示した図である。

【図164】タイマ更新処理の流れを示すフローチャートである。

【図165】(a)本発明に係るタイマ更新処理のプログラムの一例を示した図である。(b)従来のタイマ更新処理のプログラムの一例を示した図である。

【図166】(a)表示図柄抽選の流れを示すフローチャートである。(b)抽選準備処理の流れを示すフローチャートである。(c)データ選択処理の流れを示すフローチャートである。

30

【図167】本発明に係る抽選準備処理のプログラムの一例を示した図である。

【図168】抽選処理の流れを示すフローチャートである。

【図169】本発明に係る抽選処理のプログラムの一例を示した図である。

【図170】従来の抽選処理のプログラムの一例を示した図である。

【図171】払出要求数送信処理の流れを示すフローチャートである。

【図172】本発明に係る払出要求数送信処理のプログラムの一例を示した図である。

【図173】本発明の概要を示した図である。

【図174】本発明の変形例を示した図である。

【図175】本発明の他の概要を示した図である。

40

【図176】WARU命令の変形例を示した図である。

【図177】パチンコ機を正面側(遊技者側)から見た外観斜視図である。

【図178】遊技盤を正面から見た略示正面図である。

【図179】パチンコ機を背面側から見た外観斜視図である。

【図180】(a)特図の停止表示態様の一例を示したものである。(b)装飾図柄の一例を示したものである。(c)普図の停止表示態様の一例を示したものである。

【図181】制御部の回路ブロック図を示したものである。

【図182】基本回路の内部構成図である。

【図183】基本回路に接続される乱数用水晶発振器とシステム用水晶発振器の配線パターンの一例を示した回路ブロック図である。

50

【図 1 8 4】乱数回路の内部構成図である。

【図 1 8 5】乱数回路が備える周波数監視回路の内部構成図である。

【図 1 8 6】主制御部メイン処理の流れを示すフローチャートである。

【図 1 8 7】主制御部タイマ割り込み処理の流れを示すフローチャートである。

【図 1 8 8】入賞受付処理の流れを示すフローチャートである。

【図 1 8 9】特図状態更新処理の流れを示すフローチャートである。

【図 1 9 0】払出制御部メイン処理の流れを示すフローチャートである。

【図 1 9 1】(a) 払出制御部タイマ割り込み処理の流れを示すフローチャートである。

(b) ストロープ割り込み処理の流れを示すフローチャートである。

【図 1 9 2】(a) 副制御部メイン処理の流れを示すフローチャートである。(b) コマンド入力処理の流れを示すフローチャートである。(c) 変動パターン選択処理の流れを示すフローチャートである。(d) 図柄停止処理の流れを示すフローチャートである。(e) ストロープ割り込み処理の流れを示すフローチャートである。(f) チャンスボタン割り込み処理の流れを示すフローチャートである。(g) 変数更新割り込み処理の流れを示すフローチャートである。 10

【図 1 9 3】基本回路が搭載されるパッケージの外観斜視図であり、同図 (b) は、パッケージの視認非容易部の近傍を示す側面図である。

【図 1 9 4】(a) 基本回路が搭載されるパッケージの平面図である。(b) パッケージの端子と下側基板のみを抜き出して示す外観斜視図である。

【図 1 9 5】(a) 変形例 1 に係るパッケージの下側基板の一部分を示す外観斜視図である。(b) 変形例 2 に係るパッケージの下側基板の一部分を示す外観斜視図である。(c) 変形例 3 に係るパッケージの下側基板の一部分を示す外観斜視図である。(d) 変形例 4 に係るパッケージの下側基板の一部分を示す外観斜視図である。 20

【図 1 9 6】(a) 変形例 5 に係るパッケージの外観斜視図である。(b) 変形例 6 に係るパッケージの外観斜視図である。(c) 変形例 7 に係るパッケージの外観斜視図である。(d) 変形例 8 に係るパッケージの外観斜視図である。

【図 1 9 7】(a)、(b) スリット状の切欠き部を形成したパッケージの一例を示した外観斜視図である。

【図 1 9 8】(a) 円弧状の切欠き部を形成したパッケージの一例を示した外観斜視図である。(b)、(c) 切欠き部を拡大して示す部分拡大図である。 30

【図 1 9 9】(a) 切欠き部の上面にシールが貼付されたパッケージの一例を示した外観斜視図である。(b) パッケージを折り曲げた様子を示す外観斜視図である。(c) 折り曲げた後のパッケージを示す外観斜視図である。

【図 2 0 0】(a) ～ (e) 変形例に係るパッケージの平面図である。(f) ～ (g) 変形例に係るパッケージを長手方向から見た側面図である。

【図 2 0 1】(a) ICソケットの外観斜視図である。(b) ICソケットにパッケージを実装した状態を示す外観斜視図である。

【図 2 0 2】パッケージの端子配置例を示す平面図である。

【図 2 0 3】(a) 変形例 1 に係る ICソケットの外観斜視図である。(b) ICソケットにパッケージを実装した状態を示す外観斜視図である。 40

【図 2 0 4】(a) 変形例 2 に係る ICソケットの外観斜視図である。(b) ICソケットのソケット端子が実装される基板の平面図である。

【図 2 0 5】(a) 図 2 0 4 (a) における A-A 線に沿った断面を模式的に示した図である。(b) 図 2 0 4 (a) における B-B 線に沿った断面を模式的に示した図である。

【図 2 0 6】変形例 1 に係るコネクタ部とソケット端子の断面を模式的に示した図である。

【図 2 0 7】変形例 2 に係るコネクタ部とソケット端子の断面を模式的に示した図である。

【図 2 0 8】(a) 変形例 3 に係る ICソケットの外観斜視図である。(b) 変形例 4 に係る ICソケットの外観斜視図である。(c) 変形例 5 に係る ICソケットの外観斜視図 50

である。

【図 2 0 9】 I C ソケットを基板に実装した例を示した外観斜視図である。

【図 2 1 0】 (a) I C ソケットが実装されるランドと、これらのランドと乱数用水晶発振器およびシステム用水晶発振器を接続する信号線の一例を示した図である。(b) ランド近傍の部分拡大図である。

【図 2 1 1】 I C ソケットの外縁を説明するための図である。

【図 2 1 2】 基板 1 0 0 2 を側方から見た側面図である。

【図 2 1 3】 変形例 1 に係る配線パターンを示した図である。

【図 2 1 4】 (a) 変形例 2 に係る配線パターンを示した図である。(b) 変形例 3 に係る配線パターンを示した図である。

10

【図 2 1 5】 変形例 4 に係る配線パターンを示した図である。

【図 2 1 6】 (a) 変形例 5 に係る配線パターンを示した図である。(b)、(c) 変形例 5 に係る配線パターンを変更した図である。

【図 2 1 7】 変形例 6 に係る配線パターンを示した図である。

【図 2 1 8】 上述の I C ソケットを実装する基板の平面図である。

【図 2 1 9】 変形例 1 に係る基板の平面図である。

【図 2 2 0】 変形例 2 に係る基板の平面図である。

【図 2 2 1】 変形例 3 に係る基板の平面図である。

【図 2 2 2】 スロットマシンの外観斜視図を示したものである。

【図 2 2 3】 主制御部の回路ブロック図を示したものである。

20

【図 2 2 4】 副制御部の回路ブロック図を示したものである。

【図 2 2 5】 主制御部のメイン処理の流れを示すフローチャートである。

【図 2 2 6】 遊技メダル投入処理の流れを示すフローチャートである。

【図 2 2 7】 乱数取得処理の流れを示すフローチャートである。

【図 2 2 8】 タイマ割込み処理の流れを示すフローチャートである。

【図 2 2 9】 電源基板 1 8 2、払出基板 1 7 0、および主基板 1 5 6 間の電源供給例を示す図である。

【図 2 3 0】 主基板 1 5 6 内の電源供給例を示す図である。

【図 2 3 1】 電源基板 1 8 2 と主基板 1 5 6 内の電源供給例を示す図である。

【図 2 3 2】 主基板 1 5 6 内の電源供給例を示す図である。

30

【図 2 3 3】 パチンコ機 1 0 0 の電源投入後の起動処理のタイムチャートである。

【図 2 3 4】 図 2 3 3 の破線の楕円 O で囲んだ範囲を拡大した監視電圧を示す図である。

【図 2 3 5】 W D T 3 1 4 の最初のリスタート処理が時刻 T - 低電圧より後に実行されるようにした W D T の駆動方法 (その 1) を示す図である。

【図 2 3 6】 W D T 3 1 4 の最初のリスタート処理が時刻 T - 低電圧より後に実行されるようにした W D T の駆動方法 (その 2) を示す図である。

【図 2 3 7】 W D T 3 1 4 の最初のリスタート処理が時刻 T - 低電圧より後に実行されるようにした W D T の駆動方法 (その 3) を示す図である。

【図 2 3 8】 主制御部メイン処理の流れを示すフローチャートである。

【図 2 3 9】 主制御部メイン処理における遅延処理の流れを示すフローチャートである。

40

【図 2 4 0】 主制御部メイン処理における遅延処理の流れを示すフローチャートである。

【図 2 4 1】 主制御部メイン処理の流れを示すフローチャートである。

【図 2 4 2】 主制御部メイン処理における遅延処理の流れを示すフローチャートである。

【図 2 4 3】 主制御部タイマ割込処理の流れを示すフローチャートである。

【図 2 4 4】 主制御部タイマ割込処理におけるデバイス監視処理の流れを示すフローチャートである。

【図 2 4 5】 乱数生成回路から 1 6 ビット乱数値を R A M 3 0 8 に取込む手順を示す図である。

【図 2 4 6】 乱数生成回路から 1 6 ビット乱数値を R A M 3 0 8 に取込む手順を示す図である。

50

【図 2 4 7】 (a) は、図 2 4 5 および図 2 4 6 に示す乱数取込み（取得）手順において実行される乱数取得判定処理を示すフローチャートである。(b) は、図 2 4 5 および図 2 4 6 に示す乱数取込み（取得）手順において実行される乱数取得処理を示すフローチャートである。

【図 2 4 8】 (a) は、図 2 4 5 および図 2 4 6 に示す乱数取込み（取得）手順において実行される図 2 4 7 (a) に示す乱数取得判定処理をより詳細に説明するフローチャートである。(b) は、図 2 4 5 および図 2 4 6 に示す乱数取込み（取得）手順において実行される図 2 4 7 (b) に示す乱数取得処理をより詳細に説明するフローチャートである。

【図 2 4 9】 所定のソフトラッチ乱数値レジスタに乱数生成回路 9 0 0 6 から逐次入力される 1 6 ビット乱数値を直接取込む方法を示す図である。

10

【図 2 5 0】 図 2 4 9 に示す乱数直接取込みにおいて実行される乱数取得判定／取得処理を示すフローチャートである。

【図 2 5 1】 図 2 4 9 に示す乱数直接取込み（取得）手順において実行される図 2 5 0 に示す乱数取得判定／取得処理をより詳細に説明するフローチャートである。

【図 2 5 2】 図 2 4 5 や図 2 4 6 に示した乱数値取込みの手順を記述したプログラムコードの一例を示す図である。

【図 2 5 3】 図 2 4 9 に示した乱数値直接取込みの手順を記述したプログラムコードの一例を示す図である。

【図 2 5 4】 第三の命令を用いることの利点について説明する図である。

【図 2 5 5】 (a) は大当り判定テーブルの一例を、(b) は高確率状態移行判定テーブルの一例を、(c) はタイマ番号決定テーブルの一例を、それぞれ示した図である。

20

【図 2 5 6】 (a) は払出要求数テーブルの一例を、(b) は払出加工テーブルの一例を、(c) はタイマ制御データテーブルの一例を、(d) はジャンプテーブルの一例を、(e) は復元データテーブルの一例を、それぞれ示した図である。

【図 2 5 7】 同パチンコ機の主制御部リセット割り込み処理の流れを示したフローチャートである。

【図 2 5 8】 同パチンコ機の機能限定ウェイト処理の流れを示したフローチャートである。

【図 2 5 9】 同パチンコ機の主制御部タイマ割り込み処理の流れを示したフローチャートである。

30

【図 2 6 0】 同パチンコ機の払出要求数送信処理の流れを示したフローチャートである。

【図 2 6 1】 同パチンコ機のタイマ更新処理の流れを示したフローチャートである。

【図 2 6 2】 同パチンコ機のコマンド設定送信処理の流れを示したフローチャートである。

【図 2 6 3】 同パチンコ機の基本コマンド設定送信処理の流れを示したフローチャートである。

【図 2 6 4】 同パチンコ機のコマンド送信処理の流れを示したフローチャートである。

【図 2 6 5】 同パチンコ機の復帰コマンド設定処理の流れを示したフローチャートである。

【図 2 6 6】 同パチンコ機の払出制御部リセット処理の流れを示したフローチャートである。

40

【図 2 6 7】 同パチンコ機のコマンド解析処理の流れを示したフローチャートである。

【図 2 6 8】 同パチンコ機のコマンド受信割り込み処理の流れを示したフローチャートである。

【図 2 6 9】 同パチンコ機の払出制御部タイマ割り込み処理の流れを示したフローチャートである。

【図 2 7 0】 1 6 ビット乱数回路の内部構成図である。

【図 2 7 1】 8 ビット乱数回路の内部構成図である。

【図 2 7 2】 主制御部メイン処理の流れを示すフローチャートであり、図 2 7 2 (a) はプログラム起動設定処理の流れを示すフローチャートであり、図 2 7 2 (b) は、電源投

50

入時処理の流れを示すフローチャートであり、図 272 (c) は、初期設定 2 の流れを示すフローチャートである。

【図 273】主制御部メイン処理における遅延処理の流れを示すフローチャートである。

【図 274】主制御部タイマ割込処理の流れを示すフローチャートである。

【図 275】主制御部タイマ割込処理におけるデバイス監視処理の流れを示すフローチャートである。

【図 276】本実施形態の遊技台（パチンコ機）で使用される乱数の導出源を表で示す図である。

【図 277】本実施形態の遊技台（スロットマシン）で使用される乱数の導出源を表で示す図である。

10

【図 278】(a) は、基本回路 5302 が搭載されるパッケージ 10100a およびパッケージ 10100b の平面図である。(b) は、パッケージ 10100a およびパッケージ 10100b を接続部材 10100c を用いて接続した形状を示す平面図である。(c) は、同側面図である。(d) は、同斜視図である。

【図 279】(a) は、基本回路 5302 が搭載されるパッケージ 10120a およびパッケージ 10120b を接続配線部 10120c で接続した形状を示す平面図である。(b) は、同側面図である。(c) は、同斜視図である。

【図 280】図 278 (b)、(c)、(d) に示すパッケージ 10100a およびパッケージ 10100b を接続部材 10100c で一体化して収容ケース 10103 で覆った状態を示す図である。

20

【図 281】図 279 (a)、(b)、(c) に示すパッケージ 10120a およびパッケージ 10120b を接続配線部 10120c で一体化して収容ケース 10104 で覆った状態を示す図である。

【図 282】特図始動口ポートの状態と乱数取得との関係を示す図である。

【図 283】特図始動口ポートの状態と乱数取得との関係を示す図である。

【図 284】実施形態 7 の変形例に係るパチンコ機（遊技台）に用いられるカウンタ回路 312 について説明する図である。

【図 285】CPU 304 によって実行される処理 A～処理 D を示す図である。

【図 286】CPU 304 によって実行される処理 A～処理 D を示す図である。

【図 287】割込み初期設定レジスタの設定例を示す図である。

30

【図 288】割込み要求が発生した場合のジャンプ先アドレスと RST 命令でのジャンプ先アドレスとの関連について説明する図である。

【図 289】実施形態 8 の変形例について説明する図である。

【発明を実施するための形態】

【0009】

<実施形態 1>

以下、図面を用いて、本発明の実施形態 1 に係る遊技台（パチンコ機 100）について詳細に説明する。

【0010】

<全体構成>

40

まず、図 1 を用いて、本発明の実施形態 1 に係るパチンコ機 100 の全体構成について説明する。なお、同図はパチンコ機 100 を正面側（遊技者側）から見た外観斜視図である。

【0011】

パチンコ機 100 は、外部的構造として、外枠 102 と、本体 104 と、前面枠扉 106 と、球貯留皿付扉 108 と、発射装置 110 と、遊技盤 200 と、をその前面に備える。

【0012】

外枠 102 は、遊技機設置営業店に設けられた設置場所（島設備等）へと固定させるための縦長形状から成る木製の枠部材である。本体 104 は、内枠と呼ばれ、外枠 102

50

の内部に備えられ、ヒンジ部 112 を介して外枠 102 に回動自在に装着された縦長形状の遊技機基軸体となる部材である。また、本体 104 は、枠状に形成され、内側に空間部 114 を有している。また、本体 104 が開放された場合、本体 104 の開放を検出する不図示の内枠開放センサを備える。

#### 【0013】

前面枠扉 106 は、ロック機能付きで且つ開閉自在となるようにパチンコ機 100 の前面側となる本体 104 の前面に対しヒンジ部 112 を介して装着され、枠状に構成されることでその内側を開口部とした扉部材である。なお、この前面枠扉 106 には、開口部にガラス製又は樹脂製の透明板部材 118 が設けられ、前面側には、スピーカ 120 や枠ランプ 122 が取り付けられている。前面枠扉 106 の後面と遊技盤 200 の前面とで遊技領域 124 を区画形成する。また、前面枠扉 106 が開放された場合、前面枠扉 106 の開放を検出する不図示の前面枠扉開放センサを備える。

10

#### 【0014】

球貯留皿付扉 108 は、パチンコ機 100 の前面において本体 104 の下側に対して、ロック機能付きで且つ開閉自在となるように装着された扉部材である。球貯留皿付扉 108 は、複数の遊技球（以下、単に「球」と称する場合がある）が貯留可能で且つ発射装置 110 へと遊技球を案内させる通路が設けられている上皿 126 と、上皿 126 に貯留しきれない遊技球を貯留する下皿 128 と、遊技者の操作によって上皿 126 に貯留された遊技球を下皿 128 へと排出させる球抜ボタン 130 と、遊技者の操作によって下皿 128 に貯留された遊技球を遊技球収集容器（俗称、ドル箱）へと排出させる球排出レバー 132 と、遊技者の操作によって発射装置 110 へと案内された遊技球を遊技盤 200 の遊技領域 124 へと打ち出す球発射ハンドル 134 と、遊技者の操作によって各種演出装置 206 の演出態様に変化を与えるチャンスボタン 136 と、チャンスボタン 136 を発光させるチャンスボタンランプ 138 と、遊技店に設置されたカードユニット（CRユニット）に対して球貸し指示を行う球貸操作ボタン 140 と、カードユニットに対して遊技者の残高の返却指示を行う返却操作ボタン 142 と、遊技者の残高やカードユニットの状態を表示する球貸表示部 144 と、を備える。また、下皿 128 が満タンであることを検出する不図示の下皿満タンセンサを備える。

20

#### 【0015】

発射装置 110 は、本体 104 の下方に取り付けられ、球発射ハンドル 134 が遊技者に操作されることによって回動する発射杆 146 と、遊技球を発射杆 146 の先端で打突する発射槌 148 と、を備える。

30

#### 【0016】

遊技盤 200 は、前面に遊技領域 124 を有し、本体 104 の空間部 114 に臨むように、所定の固定部材を用いて本体 104 に着脱自在に装着されている。なお、遊技領域 124 は、遊技盤 200 を本体 104 に装着した後、開口部から観察することができる。

#### 【0017】

図 2 は、図 1 のパチンコ機 100 を背面側から見た外観図である。パチンコ機 100 の背面上部には、上方に開口した開口部を有し、遊技球を一時的に貯留するための球タンク 150 と、この球タンク 150 の下方に位置し、球タンク 150 の底部に形成した連通孔を通過して落下する球を背面右側に位置する払出装置 152 に導くためのタンクレール 154 とを配設している。

40

#### 【0018】

払出装置 152 は、筒状の部材からなり、その内部には、不図示の払出モータとスプロケットと払出センサとを備えている。スプロケットは、払出モータによって回転可能に構成されており、タンクレール 154 を通過して払出装置 152 内に流下した遊技球を一時的に滞留させると共に、払出モータを駆動して所定角度だけ回転することにより、一時的に滞留した遊技球を払出装置 152 の下方へ 1 個ずつ送り出すように構成している。

#### 【0019】

払出センサは、スプロケットが送り出した遊技球の通過を検知するためのセンサであり

50

、遊技球が通過しているときにハイまたはローの何れか一方の信号を、遊技球が通過していないときはハイまたはローの何れか他方の信号を払出制御部 600 へ出力する。なお、この払出センサを通過した遊技球は、不図示の球ルールを通過してパチンコ機 100 の表側に配設した上皿 126 に到達するように構成しており、パチンコ機 100 は、この構成により遊技者に対して球の払い出しを行う。

#### 【0020】

払出装置 152 の図中左側には、遊技全般の制御処理を行う主制御部 300 を構成する主基板 156 を収納する主基板ケース 158、主制御部 300 が生成した処理情報に基づいて演出に関する制御処理を行う第 1 副制御部 400 を構成する第 1 副基板 160 を収納する第 1 副基板ケース 162、第 1 副制御部 400 が生成した処理情報に基づいて演出に関する制御処理を行う第 2 副制御部 500 を構成する第 2 副基板 164 を収納する第 2 副基板ケース 166、遊技球の払出に関する制御処理を行う払出制御部 600 を構成するとともに遊技店員の操作によってエラーを解除するエラー解除スイッチ 168 を備える払出基板 170 を収納する払出基板ケース 172、遊技球の発射に関する制御処理を行う発射制御部 630 を構成する発射基板 174 を収納する発射基板ケース 176、各種電氣的遊技機器に電源を供給する電源制御部 660 を構成するとともに遊技店員の操作によって電源をオンオフする電源スイッチ 178 と電源投入時に操作されることによって RWM クリア信号を主制御部 300 に出力する RWM クリアスイッチ 180 とを備える電源基板 182 を収納する電源基板ケース 184、および払出制御部 600 とカードユニットとの信号の送受信を行う C R インターフェース部 186 を配設している。

#### 【0021】

図 3 は、遊技盤 200 を正面から見た略示正面図である。遊技盤 200 には、外ルール 202 と内ルール 204 とを配設し、遊技球が転動可能な遊技領域 124 を区画形成している。

#### 【0022】

遊技領域 124 の略中央には、演出装置 206 を配設している。この演出装置 206 には、略中央に装飾図柄表示装置 208 を配設し、その周囲に、普通図柄表示装置 210 と、第 1 特別図柄表示装置 212 と、第 2 特別図柄表示装置 214 と、普通図柄保留ランプ 216 と、第 1 特別図柄保留ランプ 218 と、第 2 特別図柄保留ランプ 220 と、高確中ランプ 222 を配設している。なお、以下、普通図柄を「普図」、特別図柄を「特図」と称する場合がある。

#### 【0023】

演出装置 206 は、演出可動体 224 を動作して演出を行うものであり、詳細については後述する。装飾図柄表示装置 208 は、装飾図柄ならびに演出に用いる様々な表示を行うための表示装置であり、本実施例では液晶表示装置 (Liquid Crystal Display) によって構成する。この装飾図柄表示装置 208 は、左図柄表示領域 208 a、中図柄表示領域 208 b、右図柄表示領域 208 c および演出表示領域 208 d の 4 つの表示領域に分割し、左図柄表示領域 208 a、中図柄表示領域 208 b および右図柄表示領域 208 c はそれぞれ異なった装飾図柄を表示し、演出表示領域 208 d は演出に用いる画像を表示する。さらに、各表示領域 208 a、208 b、208 c、208 d の位置や大きさは、装飾図柄表示装置 208 の表示画面内で自由に変更することを可能としている。なお、装飾図柄表示装置 208 として液晶表示装置を採用しているが、液晶表示装置でなくとも、種々の演出や種々の遊技情報を表示可能に構成されていればよく、例えば、ドットマトリクス表示装置、7 セグメント表示装置、有機 EL (Electro Luminescence) 表示装置、リール (ドラム) 式表示装置、リーフ式表示装置、プラズマディスプレイ、プロジェクタを含む他の表示デバイスを採用してもよい。

#### 【0024】

普図表示装置 210 は、普図の表示を行うための表示装置であり、本実施例では 7 セグメント LED によって構成する。第 1 特図表示装置 212 および第 2 特図表示装置 214 は、特図の表示を行うための表示装置であり、本実施例では 7 セグメント LED によって



構成する。

#### 【0025】

普図保留ランプ216は、保留している普図変動遊技（詳細は後述）の数を示すためのランプであり、本実施例では、普図変動遊技を所定数（例えば、2つ）まで保留することを可能としている。第1特図保留ランプ218および第2特図保留ランプ220は、保留している特図変動遊技（詳細は後述）の数を示すためのランプであり、本実施例では、特図変動遊技を所定数（例えば、4つ）まで保留することを可能としている。高確中ランプ222は、遊技状態が大当たりが発生し易い高確率状態であること、または高確率状態になることを示すためのランプであり、遊技状態が大当たりが発生し難い低確率状態から高確率状態にする場合に点灯し、高確率状態から低確率状態にする場合に消灯する。

10

#### 【0026】

また、この演出装置206の周囲には、所定の球進入口、例えば、一般入賞口226と、普図始動口228と、第1特図始動口230と、第2特図始動口232と、可変入賞口234を配設している。

#### 【0027】

一般入賞口226は、本実施例では遊技盤200に複数配設しており、この一般入賞口226への入球を所定の球検出センサ（図示省略）が検出した場合（一般入賞口226に入賞した場合）、払出装置152を駆動し、所定の個数（例えば、10個）の球を賞球として上皿126に排出する。上皿126に排出した球は遊技者が自由に取り出すことが可能であり、これらの構成により、入賞に基づいて賞球を遊技者に払い出すようにしている。なお、一般入賞口226に入球した球は、パチンコ機100の裏側に誘導した後、遊技島側に排出する。本実施例では、入賞の対価として遊技者に払い出す球を「賞球」、遊技者に貸し出す球を「貸球」と区別して呼ぶ場合があり、「賞球」と「貸球」を総称して「球（遊技球）」と呼ぶ。

20

#### 【0028】

普図始動口228は、ゲートやスルーチャッカーと呼ばれる、遊技領域124の所定の領域を球が通過したか否かを判定するための装置で構成しており、本実施例では遊技盤200の左側に1つ配設している。普図始動口228を通過した球は一般入賞口226に入球した球と違って、遊技島側に排出することはない。球が普図始動口228を通過したことを所定の球検出センサが検出した場合、パチンコ機100は、普図表示装置210による普図変動遊技を開始する。

30

#### 【0029】

第1特図始動口230は、本実施例では遊技盤200の中央に1つだけ配設している。この第1特図始動口230への入球を所定の球検出センサが検出した場合、後述する払出装置152を駆動し、所定の個数（例えば、3個）の球を賞球として上皿126に排出するとともに、第1特図表示装置212による特図変動遊技を開始する。なお、第1特図始動口230に入球した球は、パチンコ機100の裏側に誘導した後、遊技島側に排出する。

#### 【0030】

第2特図始動口232は、電動チューリップ（電チュー）と呼ばれ、本実施例では第1特図始動口230の真下に1つだけ配設している。この第2特図始動口232は、左右に開閉自在な羽根部材232aを備え、羽根部材232aの閉鎖中は球の入球が不可能であり、普図変動遊技に当選し、普図表示装置210が当り図柄を停止表示した場合に羽根部材232aが所定の時間間隔、所定の回数で開閉する。第2特図始動口232への入球を所定の球検出センサが検出した場合、払出装置152を駆動し、所定の個数（例えば、4個）の球を賞球として上皿126に排出するとともに、第2特図表示装置214による特図変動遊技を開始する。なお、第2特図始動口232に入球した球は、パチンコ機100の裏側に誘導した後、遊技島側に排出する。

40

#### 【0031】

可変入賞口234は、大入賞口またはアタッカーと呼ばれ、本実施例では遊技盤200

50

の中央部下方に１つだけ配設している。この可変入賞口２３４は、開閉自在な扉部材２３４ａを備え、扉部材２３４ａの閉鎖中は球の入球が不可能であり、特図変動遊技に当選して特図表示装置が大当たり図柄を停止表示した場合に扉部材２３４ａが所定の時間間隔（例えば、開放時間２９秒、閉鎖時間１．５秒）、所定の回数（例えば１５回）で開閉する。可変入賞口２３４への入球を所定の球検出センサが検出した場合、払出装置１５２を駆動し、所定の個数（例えば、１５個）の球を賞球として上皿１２６に排出する。なお、可変入賞口２３４に入球した球は、パチンコ機１００の裏側に誘導した後、遊技島側に排出する。

#### 【００３２】

さらに、これらの入賞口や始動口の近傍には、風車と呼ばれる円盤状の打球方向変換部材２３６や、遊技釘２３８を複数個、配設していると共に、内レール２０４の最下部には、いずれの入賞口や始動口にも入賞しなかった球をパチンコ機１００の裏側に誘導した後、遊技島側に排出するためのアウト口を設けている。

#### 【００３３】

このパチンコ機１００は、遊技者が上皿１２６に貯留している球を発射レールの発射位置に供給し、遊技者の操作ハンドルの操作量に応じた強度で発射モータを駆動し、発射杆１４６および発射槌１４８によって外レール２０２、内レール２０４を通過させて遊技領域１２４に打ち出す。そして、遊技領域１２４の上部に到達した球は、打球方向変換部材２３６や遊技釘２３８等によって進行方向を変えながら下方に落下し、入賞口（一般入賞口２２６、可変入賞口２３４）や始動口（第１特図始動口２３０、第２特図始動口２３２）に入賞するか、いずれの入賞口や始動口にも入賞することなく、または普図始動口２２８を通過するのみでアウト口２４０に到達する。

#### 【００３４】

<演出装置２０６>

次に、パチンコ機１００の演出装置２０６について説明する。

#### 【００３５】

この演出装置２０６の前面側には、遊技球の転動可能な領域にワープ装置２４２およびステージ２４４を配設し、遊技球の転動不可能な領域に演出可動体２２４を配設している。また、演出装置２０６の背面側には、装飾図柄表示装置２０８および遮蔽装置２４６（以下、扉と称する場合がある）を配設している。すなわち、演出装置２０６において、装飾図柄表示装置２０８および遮蔽手段は、ワープ装置２４２、ステージ２４４、および演出可動体２２４の後方に位置することとなる。

#### 【００３６】

ワープ装置２４２は、演出装置２０６の左上方に設けたワープ入口２４２ａに入った遊技球を演出装置２０６の前面下方のステージ２４４にワープ出口２４２ｂから排出する。ステージ２４４は、ワープ出口２４２ｂから排出された球や遊技盤２００の釘などによって乗り上げた球などが転動可能であり、ステージ２４４の中央部には、通過した球が第１特図始動口２３０へ入球し易くなるスペシャルルート２４４ａを設けている。

#### 【００３７】

演出可動体２２４は、本実施形態では人間の右腕の上腕と前腕を模した上腕部２２４ａと前腕部２２４ｂとからなり、肩の位置に上腕部２２４ａを回動させる不図示の上腕モータと肘の位置に前腕部２２４ｂを回動させる不図示の前腕モータを備える。演出可動体２２４は、上腕モータと前腕モータによって装飾図柄表示装置２０８の前方を移動する。

#### 【００３８】

遮蔽装置２４６は、格子状の左扉２４６ａおよび右扉２４６ｂからなり、装飾図柄表示装置２０８および前面ステージ２４４の間に配設する。左扉２４６ａおよび右扉２４６ｂの上部には、不図示の２つのプーリに巻き回したベルトをそれぞれ固定している。すなわち、左扉２４６ａおよび右扉２４６ｂは、モータによりプーリを介して駆動するベルトの動作に伴って左右にそれぞれ移動する。遮蔽手段は、左扉２４６ａおよび右扉２４６ｂを閉じた状態ではそれぞれの内側端部が重なり、遊技者が装飾図柄表示装置２０８を視認し

10

20

30

40

50

難しいように遮蔽する。左扉 246 a および右扉 246 b を開いた状態ではそれぞれの内側端部が装飾図柄表示装置 208 の表示画面の外側端部と若干重なるが、遊技者は装飾図柄表示装置 208 の表示の全てを視認可能である。また、左扉 246 a および右扉 246 b は、それぞれ任意の位置で停止可能であり、例えば、表示した装飾図柄がどの装飾図柄であるかを遊技者が識別可能な程度に、装飾図柄の一部だけを遮蔽するようなことができる。なお、左扉 246 a および右扉 246 b は、格子の孔から後方の装飾図柄表示装置 208 の一部を視認可能にしてもよいし、格子の孔の障子部分を半透明のレンズ体で塞ぎ、後方の装飾図柄表示装置 208 による表示を漠然と遊技者に視認させるようにしてもよいし、格子の孔の障子部分を完全に塞ぎ（遮蔽し）、後方の装飾図柄表示装置 208 を全く視認不可にしてもよい。

10

#### 【0039】

##### <制御部>

次に、図 4 を用いて、このパチンコ機 100 の制御部の回路構成について詳細に説明する。なお、同図は制御部の回路ブロック図を示したものである。

#### 【0040】

パチンコ機 100 の制御部は、大別すると、主に遊技の進行（例えば、遊技者による操作の検出、遊技状態の遷移、遊技媒体の払出制御、当否判定など）を制御する主制御部 300 と、主制御部 300 が送信するコマンド信号（以下、単に「コマンド」と呼ぶ）に応じて主に演出の制御を行う第 1 副制御部 400 と、第 1 副制御部 400 より送信されたコマンドに基づいて各種機器を制御する第 2 副制御部 500 と、主制御部 300 が送信するコマンドに応じて主に遊技球の払い出しに関する制御を行う払出制御部 600 と、遊技球の発射制御を行う発射制御部 630 と、パチンコ機 100 に供給される電源を制御する電源制御部 660 と、によって構成している。

20

#### 【0041】

##### <主制御部>

まず、パチンコ機 100 の主制御部 300 について説明する。

#### 【0042】

主制御部 300 は、主制御部 300 の全体を制御する基本回路 302 を備えており、この基本回路 302 には、CPU 304 と、制御プログラムや各種データを記憶するための ROM 306 と、一時的にデータを記憶するための RAM 308 と、各種デバイスの入出力を制御するための I/O 310 と、時間や回数等を計測するためのカウンタタイマ 312 と、プログラム処理の異常を監視する WDT 314 を搭載している。なお、ROM 306 や RAM 308 については他の記憶装置を用いてもよく、この点は後述する第 1 副制御部 400 についても同様である。この基本回路 302 の CPU 304 は、水晶発振器 316 b が出力する所定周期のクロック信号をシステムクロックとして入力して動作する。

30

#### 【0043】

また、基本回路 302 には、水晶発振器 316 a が出力するクロック信号を受信する度に 0~65535 の範囲で数値を変動させるハードウェア乱数カウンタとして使用しているカウンタ回路 318（この回路には 2 つのカウンタを内蔵しているものとする）と、所定の球検出センサ、例えば各始動口、入賞口、可変入賞口を通過する遊技球を検出するセンサや、前面枠扉開放センサや内枠開放センサや下皿満タンセンサを含む各種センサ 320 が出力する信号を受信し、増幅結果や基準電圧との比較結果をカウンタ回路 318 および基本回路 302 に出力するためのセンサ回路 322 と、所定の図柄表示装置、例えば第 1 特図表示装置 212 や第 2 特図表示装置 214 の表示制御を行うための駆動回路 324 と、所定の図柄表示装置、例えば普図表示装置 210 の表示制御を行うための駆動回路 326 と、各種状態表示部 328（例えば、普図保留ランプ 216、第 1 特図保留ランプ 218、第 2 特図保留ランプ 220、高確中ランプ 222 等）の表示制御を行うための駆動回路 330 と、所定の可動部材、例えば第 2 特図始動口 232 の羽根部材 232 a や可変入賞口 234 の扉部材 234 a 等を開閉駆動する各種ソレノイド 332 を制御するための駆動回路 334 を接続している。

40

50

## 【0044】

なお、第1特図始動口230に球が入賞したことを球検出センサ320が検出した場合には、センサ回路322は球を検出したことを示す信号をカウンタ回路318に出力する。この信号を受信したカウンタ回路318は、第1特図始動口230に対応するカウンタのそのタイミングにおける値をラッチし、ラッチした値を、第1特図始動口230に対応する内蔵のカウント値記憶用レジスタに記憶する。また、カウンタ回路318は、第2特図始動口232に球が入賞したことを示す信号を受信した場合も同様に、第2特図始動口232に対応するカウンタのそのタイミングにおける値をラッチし、ラッチした値を、第2特図始動口232に対応する内蔵のカウント値記憶用レジスタに記憶する。

## 【0045】

さらに、基本回路302には、情報出力回路336を接続しており、主制御部300は、この情報出力回路336を介して、外部のホールコンピュータ（図示省略）等が備える情報入力回路350にパチンコ機100の遊技情報（例えば、遊技状態）を出力する。

## 【0046】

また、主制御部300には、電源制御部660から主制御部300に供給している電源の電圧値を監視する電圧監視回路338を設けており、この電圧監視回路338は、電源の電圧値が所定の値（本実施例では9V）未満である場合に電圧が低下したことを示す低電圧信号を基本回路302に出力する。

## 【0047】

また、主制御部300には、電源が投入されると起動信号（リセット信号）を出力する起動信号出力回路（リセット信号出力回路）340を設けており、CPU304は、この起動信号出力回路340から起動信号を入力した場合に、遊技制御を開始する（詳細は後述する）。

## 【0048】

また、主制御部300は、第1副制御部400にコマンドを送信するための出力インタフェースと、払出制御部600にコマンドを送信するための出力インタフェースをそれぞれ備えており、この構成により、第1副制御部400および払出制御部600との通信を可能としている。なお、主制御部300と第1副制御部400および払出制御部600との情報通信は一方方向の通信であり、主制御部300は第1副制御部400および払出制御部600にコマンド等の信号を送信できるように構成しているが、第1副制御部400および払出制御部600からは主制御部300にコマンド等の信号を送信できないように構成している。

## 【0049】

## &lt;副制御部&gt;

次に、パチンコ機100の第1副制御部400について説明する。第1副制御部400は、主に主制御部300が送信したコマンド等に基づいて第1副制御部400の全体を制御する基本回路402を備えており、この基本回路402には、CPU404と、制御プログラムや各種演出データを記憶するためのROM406と、一時的にデータを記憶するためのRAM408と、各種デバイスの入出力を制御するためのI/O410と、時間や回数等を計測するためのカウンタタイマ412を搭載している。この基本回路402のCPU404は、水晶発信器414が出力する所定周期のクロック信号をシステムクロックとして入力して動作する。なお、ROM406は、制御プログラムと各種演出データとを別々のROMに記憶させてもよい。

## 【0050】

また、基本回路402には、スピーカ120（およびアンプ）の制御を行うための音源IC416と、各種ランプ418（例えば、チャンスボタンランプ138）の制御を行うための駆動回路420と、遮蔽装置246の駆動制御を行うための駆動回路432と、遮蔽装置246の現在位置を検出する遮蔽装置センサ430と、チャンスボタン136の押下を検出するチャンスボタン検出センサ426と、遮蔽装置センサ430やチャンスボタン検出センサ426からの検出信号を基本回路402に出力するセンサ回路428と、C

P U 4 0 4 からの信号に基づいて R O M 4 0 6 に記憶された画像データ等を読み出して V R A M 4 3 6 のワークエリアを使用して表示画像を生成して装飾図柄表示装置 2 0 8 に画像を表示する V D P 4 3 4 (ビデオ・ディスプレイ・プロセッサ)と、を接続している。

#### 【0051】

次に、パチンコ機 1 0 0 の第 2 副制御部 5 0 0 について説明する。第 2 副制御部 5 0 0 は、第 1 副制御部 4 0 0 が送信した制御コマンドを入力インタフェースを介して受信し、この制御コマンドに基づいて第 2 副制御部 5 0 0 の全体を制御する基本回路 5 0 2 を備えており、この基本回路 5 0 2 は、C P U 5 0 4 と、一時的にデータを記憶するための R A M 5 0 8 と、各種デバイスの入出力を制御するための I / O 5 1 0 と、時間や回数等を計測するためのカウンタタイマ 5 1 2 を搭載している。基本回路 5 0 2 の C P U 5 0 4 は、水晶発振器 5 1 4 が出力する所定周期のクロック信号をシステムクロックとして入力して動作し、第 2 副制御部 5 0 0 の全体を制御するための制御プログラム及びデータ、画像表示用のデータ等が記憶された R O M 5 0 6 が設けられている。

10

#### 【0052】

また、基本回路 5 0 2 には、演出可動体 2 2 4 の駆動制御を行うための駆動回路 5 1 6 と、演出可動体 2 2 4 の現在位置を検出する演出可動体センサ 4 2 4 と、演出可動体センサ 4 2 4 からの検出信号を基本回路 5 0 2 に出力するセンサ回路 5 1 8 と、遊技盤用ランプ 5 3 2 の制御を行うための遊技盤用ランプ駆動回路 5 3 0 と、遊技台枠用ランプ 5 4 2 の制御を行うための遊技台枠用ランプ駆動回路 5 4 0 と、遊技盤用ランプ駆動回路 5 3 0 と遊技台枠用ランプ駆動回路 5 4 0 との間でシリアル通信による点灯制御を行うシリアル通信制御回路 5 2 0 と、を接続している。

20

#### 【0053】

< 払出制御部、発射制御部、電源制御部 >

次に、パチンコ機 1 0 0 の払出制御部 6 0 0、発射制御部 6 3 0、電源制御部 6 6 0 について説明する。

#### 【0054】

払出制御部 6 0 0 は、主に主制御部 3 0 0 が送信したコマンド等の信号に基づいて払出装置 1 5 2 の払出モータ 6 0 2 を制御すると共に、払出センサ 6 0 4 が出力する制御信号に基づいて賞球または貸球の払い出しが完了したか否かを検出すると共に、インタフェース部 6 0 6 を介して、パチンコ機 1 0 0 とは別体で設けられたカードユニット 6 0 8 との通信を行う。

30

#### 【0055】

発射制御部 6 3 0 は、払出制御部 6 0 0 が出力する、発射許可または停止を指示する制御信号や、球発射ハンドル 1 3 4 内に設けた発射強度出力回路が出力する、遊技者による球発射ハンドル 1 3 4 の操作量に応じた発射強度を指示する制御信号に基づいて、発射杆 1 4 6 および発射槌 1 4 8 を駆動する発射モータ 6 3 2 の制御や、上皿 1 2 6 から発射装置 1 1 0 に球を供給する球送り装置 6 3 4 の制御を行う。

#### 【0056】

電源制御部 6 6 0 は、パチンコ機 1 0 0 に外部から供給される交流電源を直流化し、所定の電圧に変換して主制御部 3 0 0、第 1 副制御部 4 0 0 等の各制御部や払出装置 1 5 2 等の各装置に供給する。さらに、電源制御部 6 6 0 は、外部からの電源が断たれた後も所定の部品 (例えば主制御部 3 0 0 の R A M 3 0 8 等) に所定の期間 (例えば 1 0 日間) 電源を供給するための蓄電回路 (例えば、コンデンサ) を備えている。なお、本実施形態では、電源制御部 6 6 0 から払出制御部 6 0 0 と第 2 副制御部 5 0 0 に所定電圧を供給し、払出制御部 6 0 0 から主制御部 3 0 0 と第 2 副制御部 5 0 0 と発射制御部 6 3 0 に所定電圧を供給しているが、各制御部や各装置に他の電源経路で所定電圧を供給してもよい。

40

#### 【0057】

< 図柄の種類 >

次に、図 5 (a) ~ (c) を用いて、パチンコ機 1 0 0 の第 1 特別図柄表示装置 2 1 2

50

、第2特別図柄表示装置214、装飾図柄表示装置208、普通図柄表示装置210が停止表示する特図および普図の種類について説明する。同図(a)は特図の停止図柄態様の一例を示したものである。

#### 【0058】

第1特図始動口230に球が入球したことを第1始動口センサが検出したことを条件として特図1変動遊技が開始され、第2特図始動口232に球が入球したことを第2始動口センサが検出したことを条件として特図2変動遊技が開始される。特図1変動遊技が開始されると、第1特別図柄表示装置212は、7個のセグメントの全点灯と、中央の1個のセグメントの点灯を繰り返す「特図1の変動表示」を行う。また、特図2変動遊技が開始されると、第2特別図柄表示装置214は、7個のセグメントの全点灯と、中央の1個のセグメントの点灯を繰り返す「特図2の変動表示」を行う。これらの「特図1の変動表示」および「特図2の変動表示」が本発明にいう図柄の変動表示の一例に相当する。そして、特図1の変動開始前に決定した変動時間（本発明にいう変動時間が相当）が経過すると、第1特別図柄表示装置212は特図1の停止図柄態様を停止表示し、特図2の変動開始前に決定した変動時間（これも本発明にいう変動時間が相当）が経過すると、第2特別図柄表示装置214は特図2の停止図柄態様を停止表示する。したがって、「特図1の変動表示」を開始してから特図1の停止図柄態様を停止表示するまで、あるいは「特図2の変動表示」を開始してから特図2の停止図柄態様を停止表示するまでが本発明にいう図柄変動停止表示の一例に相当し、以下、この「特図1又は2の変動表示」を開始してから特図1又は2の停止図柄態様を停止表示するまでの一連の表示を図柄変動停止表示と称する。後述するように、図柄変動停止表示は複数回、連続して行われることがある。同図(a)には、図柄変動停止表示における停止図柄態様として「特図A」から「特図J」までの10種類の特図が示されており、図中の白抜きの部分が消灯するセグメントの場所を示し、黒塗りの部分が点灯するセグメントの場所を示している。

#### 【0059】

「特図A」は15ラウンド(15R)特別大当り図柄であり、「特図B」は15R大当り図柄である。本実施形態のパチンコ機100では、後述するように、特図変動遊技における大当りか否かの決定はハードウェア乱数の抽選によって行い、特別大当りか否かの決定はソフトウェア乱数の抽選によって行う。大当りと特別大当りの違いは、次回の特図変動遊技で、大当りに当選する確率が高い(特別大当り)か低い(大当り)かの違いである。以下、この大当りに当選する確率が高い状態のことを特図高確率状態と称し、その確率が低い状態のことを特図低確率状態と称する。また、15R特別大当り遊技終了後および15R大当り遊技終了後はいずれも時短状態に移行する。時短については詳しくは後述するが、時短状態に移行する状態のことを普図高確率状態と称し、時短状態に移行しない状態のことを普図低確率状態と称する。15R特別大当り図柄である「特図A」は、特図高確率普図高確率状態であり、15R大当り図柄である「特図B」は、特図低確率普図高確率状態である。これらの「特図A」および「特図B」は、遊技者に付与する利益量が相対的に大きな利益量になる図柄である。

#### 【0060】

「特図C」は突然確変と称される2R大当り図柄であり、特図高確率普図高確率状態である。すなわち、15Rである「特図A」と比べて、「特図C」は2Rである点異なる。「特図D」は突然時短と称される2R大当り図柄であり、特図低確率普図高確率状態である。すなわち、15Rである「特図B」と比べて、「特図D」は2Rである点異なる。

#### 【0061】

「特図E」は隠れ確変と称される2R大当り図柄であり、特図高確率普図低確率状態である。「特図F」は突然通常と称される2R大当り図柄であり、特図低確率普図低確率状態である。これら「特図E」および「特図F」はいずれも、2Rであるとともに、時短状態に移行しない状態である。

#### 【0062】

「特図 G」は第 1 小当り図柄であり、「特図 H」は第 2 小当り図柄であり、何れも特図低確率普図低確率状態である。ここにいう小当りは、2 R 時短無し大当りと同じものに相当する。すなわち、この「特図 G」、「特図 H」は「特図 F」と同じ状態であるが、両者では装飾図柄表示装置 208 に表示される演出が異なり、あえて、同じ状態でも「特図 G」、「特図 H」と「特図 F」を設けておくことで、遊技の興趣を高めている。

#### 【0063】

また、「特図 I」は第 1 はずれ図柄であり、「特図 J」は第 2 はずれ図柄であり、遊技者に付与する利益量が相対的に小さな利益量になる図柄である。

#### 【0064】

なお、本実施形態のパチンコ機 100 には、15 R 特別大当り図柄として「特図 A」以外の図柄も用意されており、15 R 大当り図柄等の他の図柄についても同様である。

#### 【0065】

図 5 (b) は装飾図柄の一例を示したものである。本実施形態の装飾図柄には、「装飾 1」～「装飾 10」の 10 種類がある。第 1 特図始動口 230 または第 2 特図始動口 232 に球が入賞したこと、すなわち、第 1 特図始動口 230 に球が入球したことを第 1 始動口センサが検出したこと、あるいは第 2 特図始動口 232 に球が入球したことを第 2 始動口センサが検出したことを条件にして、装飾図柄表示装置 208 の左図柄表示領域 208 a、中図柄表示領域 208 b、右図柄表示領域 208 c の各図柄表示領域に、「装飾 1」→「装飾 2」→「装飾 3」→・・・「装飾 9」→「装飾 10」→「装飾 1」→・・・の順番で表示を切り替える「装飾図柄の変動表示」を行う。そして、「特図 B」の 15 R 大当りを報知する場合には、図柄表示領域 208 a～208 c に 15 R 大当りに対応する、同じ装飾図柄が 3 つ並んだ図柄組合せ（例えば「装飾 1－装飾 1－装飾 1」や「装飾 2－装飾 2－装飾 2」等）を停止表示する。「特図 A」の 15 R 特別大当りを報知する場合には、同じ奇数の装飾図柄が 3 つ並んだ図柄組合せ（例えば「装飾 3－装飾 3－装飾 3」や「装飾 7－装飾 7－装飾 7」等）を停止表示する。

#### 【0066】

また、「特図 E」の隠れ確変と称される 2 R 大当り、「特図 F」の突然通常と称される 2 R 大当り、あるいは「特図 G」の第 1 小当り、「特図 H」の第 2 小当りを報知する場合には、「装飾 1－装飾 2－装飾 3」を停止表示する。さらに、「特図 C」の突然確変と称される 2 R 大当り、あるいは「特図 D」の突然時短と称される 2 R 大当りを報知する場合には、「装飾 1－装飾 3－装飾 5」を停止表示する。一方、「特図 I」の第 1 はずれ、「特図 J」の第 2 はずれを報知する場合には、図柄表示領域 208 a～208 c に同図 (b) に示す図柄組合せ以外の図柄組合せを停止表示する。

#### 【0067】

図 5 (c) は普図の停止表示図柄の一例を示したものである。本実施形態の普図の停止表示態様には、当り図柄である「普図 A」と、外れ図柄である「普図 B」の 2 種類がある。普図始動口 228 を球が通過したことを上述のゲートセンサが検出したことに基づいて、普通図柄表示装置 210 は、7 個のセグメントの全点灯と、中央の 1 個のセグメントの点灯を繰り返す「普図の変動表示」を行う。そして、普図変動遊技の当選を報知する場合には「普図 A」を停止表示し、普図変動遊技の外れを報知する場合には「普図 B」を停止表示する。この同図 (c) においても、図中の白抜きの部分が消灯するセグメントの場所を示し、黒塗りの部分が点灯するセグメントの場所を示している。

#### 【0068】

＜主制御部の基本回路＞

次に、上述の主制御部 300 の基本回路 302 について詳細に説明する。図 6 は、主制御部 300 から基本回路 302 の主要部のみを抜き出して示した図である。

#### 【0069】

主制御部 300 の基本回路 302 は、主制御部 300 全体の制御を行う CPU 304 と、CPU 304 が実行する制御プログラムや CPU 304 が参照する各種データを記憶するための ROM 306 と、CPU 304 が一時的にデータを記憶するための RAM 308

と、を有している。

#### 【0070】

＜主制御部のCPUの内蔵レジスタ＞

CPU304は、カウンタ回路、タイマ回路、シリアル通信回路、乱数回路、演算回路、リセット／割込みコントローラ等を制御するための制御レジスタのほか、CPU304が制御プログラムを実行する際に使用するレジスタなどの内蔵レジスタを備える。内蔵レジスタには、本発明の特徴点の一つであるTレジスタ（特殊レジスタ）と、汎用レジスタと、プログラムカウンタ（PC）、命令レジスタ等が含まれる。汎用レジスタの種類は特に限定されないが、本実施形態では、Iレジスタ、Aレジスタ、Fレジスタ、Bレジスタ、Cレジスタ、Dレジスタ、Eレジスタ、Hレジスタ、Lレジスタの8ビットレジスタを備えており、BレジスタとCレジスタ、DレジスタとEレジスタ、HレジスタとLレジスタは、それぞれ、16ビット長の演算が可能なBCレジスタ、DEレジスタ、HLレジスタの16ビットレジスタ（ペアレジスタ）としても機能するように構成されている。

10

#### 【0071】

また、本実施形態では、内蔵デバイスが出力する割込みベクタ（所定のデバイスから読み込んだ割込みベクタ）の値を下位8ビット、Iレジスタ（割込みベクタレジスタ、インタラプトレジスタ）の内容を上位8ビットとした合計16ビットのアドレスを、割込処理のアドレスとして読み込んで、後述する割込処理に分岐（ジャンプ）するように構成している。例えば、Iレジスタの値が00H、割込みベクタの値が60Hのときに割込処理の要求が発生した場合には、0060H（上位バイトがIレジスタの00H、下位バイトが割込みベクタの60H）で示されるアドレスに記憶された値（例えば、12H）を下位バイト、続く0061Hで示されるアドレスに記憶された値（例えば、04H）を上位バイトとする2バイトのアドレス（例えば、0412H）がPCにロードされ、このアドレスを先頭アドレスとする割込処理に分岐（ジャンプ）するように構成している。

20

#### 【0072】

Fレジスタは、命令が実行されると、その実行結果に従ってFレジスタ内のビット7から順にSフラグ（サインフラグ）、Zフラグ（ゼロフラグ）、SZフラグ（セカンドゼロフラグ）、Hフラグ（ハーフキャリフラグ）、ビット3（空きビット）、P/Vフラグ（パリティ／オーバーフローフラグ）、Nフラグ（サブトラクトフラグ）、Cフラグ（キャリーフラグ）の各フラグが、1または0に変化したり、あるいは変化しなかったりする。SZフラグは、Zフラグが変化する命令ではZフラグ同様に変化するとともに、Z80互換命令の全てのLD命令、全ての16ビット算術演算命令（INC命令、DEC命令も含む）、ローテートシフト命令のRLCA命令、RLA命令、RRCA命令、RRA命令、「IN A, (n)」命令で変化する。

30

#### 【0073】

なお、16ビット演算命令のADC命令、SBC命令は、Zフラグが変化し、他の命令ではZフラグは変化せず、16ビット演算命令の全てでSZフラグは変化する。また、「IN A, (n)」命令は、SZフラグが変化し、Zフラグが変化しない。Fレジスタ内のSZフラグや空きビットの位置は上記したビット以外のビットであってもよい。なお、後述する特殊命令の場合にSZフラグが変化しないようにしてもよい。

40

#### 【0074】

また、本実施形態では、A、F、B、C、D、E、H、Lの各汎用レジスタ（主レジスタ）に対応する補助レジスタ（裏レジスタ）として、A'、F'、B'、C'、D'、E'、H'、L'の各汎用レジスタを備えているが、特殊レジスタであるTレジスタのみ補助レジスタを備えていない。よって、Tレジスタの使用用途は限定され、Tレジスタの使用頻度は汎用レジスタに比べて低くなるため、不用意にTレジスタの値が書き換えられてしまうような事態（Tレジスタの内容が破壊されてしまうような事態）を未然に防止できる場合がある。なお、補助レジスタの値は各種命令によって直接読み書きすることができず、特殊な交換命令（EX命令、EXX命令）によって主レジスタの値と補助レジスタの値を交換することのみが許されている。ここでは、説明の便宜上、レジスタの値をPUS

50



H命令・POP命令等の命令で変更し易いレジスタを汎用レジスタとしたが、これに限らず、Aレジスタ（アキュムレータ）とFレジスタ（フラグレジスタ）など特別な役割を持つレジスタを汎用レジスタに含めなくてもよい。

#### 【0075】

また、ここでは、TレジスタをCPU304に内蔵する例を示したが、例えば、CPU304の外部に設けてもよいし、CPU304とは別体のRAM308の一部に設けてもよい。すなわち、Tレジスタは、制御プログラムを記憶するための記憶手段（本実施形態ではROM306）と物理的に異なる場所に設けられていることが好ましい。さらには、Tレジスタに（後述する）特定の値を書き込む処理を行う制御部と、ROM306から制御プログラムを読み出して遊技制御を行う制御部と、が異なることが好ましい。制御部を異ならせることで、制御プログラムの改造と特定の値の不正取得の両方を行うことを困難にすることができるためである。

10

#### 【0076】

＜主制御部のCPUのメモリ空間とI/O空間＞

次に、図7を用いて、主制御部300のCPU304のメモリ空間とI/O空間について説明する。なお、同図（a）はCPU304のメモリマップを示した図であり、同図（b）はCPU304のI/Oマップを示した図である。

#### 【0077】

主制御部300のCPU304は、上述の内蔵レジスタ、ROM306、RAM308などにアクセスするためのメモリ空間と、上述のI/O310との間で制御信号の入出力を行うためのI/O空間を有している。具体的には、同図（a）のメモリマップに示すように、主制御部300のROM306のアドレス／データ信号線は、メモリ空間のROM領域（本実施形態では、0000H～2FFFFH（Hは16進数を示す。以下、同じ）のうち、0000H～2FBFH。2FC0H～2FFFFHはプログラム管理エリア）に割り当てられ、CPU304は、このROM領域を指定してROM306からのデータの読み出しを行う。また、本実施形態では、ROM領域を第一領域～第八領域の8つの領域に分けて管理を行っており、第一領域は0000Hから始まる領域としている。

20

#### 【0078】

また、内蔵レジスタのアドレス／データ信号線は、メモリ空間のレジスタエリア（本実施形態では、FE00H～FEBFH）に割り当てられ、CPU304は、このレジスタエリアを介して内蔵レジスタからのデータの読み出しやレジスタへのデータの書き込みを行う。

30

#### 【0079】

また、RAM308のアドレス／データ信号線は、メモリ空間のRWM領域（本実施形態では、F000H～F3FFH）に割り当てられ、CPU304は、このRWM領域を指定してRAM308からのデータの読み出しやRAM308へのデータの書き込みを行う。なお、メモリ空間の他の領域（本実施形態では、3000H～EFFFFH、F400H～FDDFH、FEC0H～FFFFFFH）は非使用領域とされている。

#### 【0080】

また、同図（b）のI/Oマップに示すように、主制御部300のI/O310は、I/O空間のワークエリア（本実施形態では、00H～BFH）に割り当てられ、CPU304は、このI/O空間を介して、I/O310との間で制御信号の入出力を行う。なお、I/O空間の他の領域（本実施形態では、C0H～FFH）は非使用領域とされている。

40

#### 【0081】

＜初期化処理＞

次に、図8を用いて、主制御部300の初期化処理について説明する。なお、同図は初期化処理の流れを示すフローチャートである。

#### 【0082】

上述したように、主制御部300には、電源が投入されると起動信号（システムリセッ

50

ト信号)を出力する起動信号出力回路(リセット信号出力回路)340を設けている。この起動信号を入力した基本回路302のCPU304は、後述するセキュリティモードまたはユーザモードに移行する前に、この初期化処理を実行する。

#### 【0083】

具体的には、ステップS51では、CPU304に内蔵されたTレジスタに、特定の値(本実施形態では、上述のRWM領域の先頭アドレスF000Hの上位バイトであるF0H)を記憶する。ステップS52では、Tレジスタ以外の他のレジスタに初期値をセットした後にステップS53に進み、後述する主制御部メイン処理を実行する。

#### 【0084】

なお、Tレジスタに記憶する特定の値はF0Hに限定されないことは言うまでもなく、例えば、上述のレジスタエリアの先頭アドレス(本実施形態ではFE00H)の上位バイトであるFEHをTレジスタに記憶するように構成すれば、後述するRAM308へのアクセスと同様の原理で、内蔵レジスタへのアクセス効率を高めることができる場合がある。また、第一のタイミングではTレジスタにRWM領域の先頭アドレスの上位バイトを設定し、その後(または、その前)の第二のタイミングでTレジスタにレジスタエリアの先頭アドレスの上位バイトを設定するように構成してもよい。また、所定条件が成立している場合(例えば、所定の端子にLレベルの信号が入力されている場合)にはTレジスタにRWM領域の先頭アドレスの上位バイトを設定し、所定条件が成立していない場合には、Tレジスタにレジスタエリアの先頭アドレスの上位バイトを設定するように構成してもよい。

#### 【0085】

また、この初期化処理において制御プログラムによらずに(自動的に)TレジスタにRWM領域の先頭アドレスの上位バイトを設定(または／およびTレジスタにレジスタエリアの先頭アドレスの上位バイトを設定)し、この初期化処理後に制御プログラム(例えば、後述するステップS101の初期設定1処理、ステップS107の初期設定2処理、ステップS113の初期化処理など)によって(手動で)Tレジスタにレジスタエリアの先頭アドレスの上位バイトを設定(または／およびTレジスタにRWM領域の先頭アドレスの上位バイトを設定)するように構成してもよい。

#### 【0086】

ここで、上述のリセット割込みには、システムリセット割込み(システムリセット端子に一定期間のリセット信号を入力するなど、外部イベントが要因となって行われるリセット割込み)と、ユーザリセット割込み(ユーザリセット端子に一定期間のリセット信号を入力するなど、外部イベントが要因となって行われるリセット割り込みや、WDTのタイムアウトや指定エリア外のアクセスなど、内部イベントが要因となって行われるリセット割込み)の2種類の割込みがある。システムリセット割込みによるリセットでは、制御レジスタや汎用レジスタを含む全ての内部回路が初期化され(例えば、レジスタに応じた初期値(例えばIレジスタに00Hがセットされる))、ユーザリセット割込みによるリセットでは、内部回路の一部が初期化される。

#### 【0087】

図9は、システムリセット後の起動処理のタイムチャートである。システムリセット(XSRST端子に所定期間(例えば、システムクロックの4周期分の期間)以上のLレベルの信号が入力された後にHレベルの信号が入力された場合)による初期化処理を実行した後は、所定条件が成立している場合(例えば、所定の端子にLレベルの信号が入力されている場合)はセキュリティモードに移行し、所定条件が成立していない場合はユーザモードに移行する。ここで、セキュリティモードは、ユーザプログラムの認証を行うモードである。具体的には、システムリセット入力時にユーザプログラムを基に計算された認証コードが正しいかどうかの再計算を行い、結果(セキュリティチェックの結果)がOK(異常なし)の場合、このモードを終了して自動的にユーザモードに移行可能とし、セキュリティチェックの結果がNG(異常あり)の場合、CPU304が停止するように構成している。なお、認証コードは、ROM書き込み時にユーザプログラム(以下、「制御プロ

グラム」と称する場合がある）とともにROM 306に書き込まれる。

#### 【0088】

このセキュリティモードにおいてセキュリティチェックが終了した後、一定の期間の経過を待つ固定延長処理（以下、「固定遅延機能」ともいう）が終了したときにXRSTO端子からHレベルのリセット出力を出力し、その後、ランダムな期間の経過を待つランダム延長処理（以下、「ランダム遅延機能」ともいう）が終了した場合にユーザモードに移行するように構成されている（固定延長処理とランダム延長処理を合わせて「セキュリティモード延長処理」や「セキュリティモード延長機能」ともいう）。ユーザモードは、ROMのリセットアドレス（上述のROM領域の第一領域の先頭アドレスである0000H）からユーザプログラムを実行するモードであり、後述する主制御部メイン処理が開始される。なお、本実施形態では、CPU 304がユーザプログラムを実行する際に1回の処理（1ステート）で命令レジスタに読みこむデータのバイト長を1バイト（8ビット）長としているが、1ステート当りのデータ長は特に限定されず、例えば、16ビット長や32ビット長でもよい。

10

#### 【0089】

また、本実施形態では、上述の初期化処理はセキュリティモードやユーザモードへの移行前に（自動的に（制御プログラムによらずに））実行されるように構成しているが、セキュリティモードやユーザモードへの移行後に実行されるよう構成してもよい。また、リセットの種類（ユーザリセット割込みの場合には割込み要因の種類）によって、初期化処理の実行タイミングが異なるように構成してもよい。

20

#### 【0090】

具体的には、直前に発生したリセット要因を管理するためのレジスタ（内蔵レジスタのうちの一つである）を参照して、直前に発生したリセット要因が（1）システムリセットであるか否か、（2）WDTのタイムアウトによるユーザリセットであるか否か、（3）指定エリア外のアクセスによるユーザリセットであるか否か、を判定し、判定結果に応じて初期化処理の実行タイミングを決定するように構成してもよい。また、初期化処理でセットされる特定の値と初期値は、リセットの種類やレジスタの種類によって異なるように構成してもよい（例えば、汎用レジスタはシステムリセットでもユーザリセットでも初期値として00Hがセットされ、Tレジスタはシステムリセット時は特定の値としてF0Hがセットされるがユーザリセット時は初期化されないように構成してもよい）。

30

#### 【0091】

また、セキュリティモードの時間は、設定した時間分だけ延長することも、ランダムな時間分だけ延長することも可能である。ランダムな時間分だけ延長することにより、不正行為者はユーザプログラムの開始タイミングが狙いにくくなる場合がある。また、ユーザリセット（例えばWDTのタイムアウトによるリセット）からユーザプログラムの実行開始までの時間をランダムな時間に設定可能な構成としてもよい。このような構成とすれば、ユーザプログラムの開始タイミングを狙った不正行為を未然に防止できる場合がある。

#### 【0092】

<主制御部メイン処理>

次に、図10を用いて、主制御部300のCPU 304が実行する主制御部メイン処理について説明する。なお、同図は主制御部メイン処理の流れを示すフローチャートである。

40

#### 【0093】

ステップS101では、初期設定1を行う。この初期設定1では、CPU 304のスタックポインタ（SP）へのスタック初期値の設定（仮設定）、割込マスクの設定、I/O 310の初期設定、RAM 308に記憶する各種変数の初期設定、WDT 314の起動やタイムアウト時間の設定、Tレジスタチェック処理（詳細は後述）等を行う。本実施形態では、WDT 314のタイムアウト時間として32.8msに相当する数値を初期値として設定する。なお、ここでは、WDT 314の起動とタイムアウト時間の設定を、主制御部メイン処理のメインループに入る直前で行う例を示したが、例えば、上述の初期化処理など

50

、他の処理の中で行ってもよい。また、ステップ S 1 0 3 では、W D T 3 1 4 のカウンタの値をクリアし、W D T 3 1 4 による時間計測を再始動（リスタート）する。なお、W D T 3 1 4 のクリア&リスタートをメインループ開始直後に行う例を示したが、例えば、後述する主制御部タイマ割込処理の中で定期的に行ってもよい。

#### 【0094】

また、図示は省略するが、基本回路 3 0 2 は、W D T 3 1 4 を制御するレジスタとして、スタートレジスタとクリアレジスタを備えている。スタートレジスタは、W D T 3 1 4 の起動／停止を制御するための 8 ビット長のレジスタであり、W D T 3 1 4 を起動するときには起動を指示する値（例えば、C C H）を設定し（書き込み）、W D T 3 1 4 を停止するときには停止を指示する値（例えば、3 3 H）を設定するように構成されている。

10

#### 【0095】

また、このスタートレジスタは、リセット時にプログラム管理エリアに「ソフトウェアによる起動」を設定（選択）したときに有効となるレジスタであり、その場合、停止を指示する値（例えば、3 3 H）が初期値として設定される。したがって、「ソフトウェアによる起動」を設定（選択）したときに W D T 3 1 4 を起動するためには、遊技制御プログラムによって（手動で）起動を指示する値（例えば、C C H）を設定する必要がある。一方、リセット時に「ソフトウェアによる起動」を設定（選択）しなかったときは、スタートレジスタには、起動を指示する値（例えば、C C H）が遊技制御プログラムによらずに（自動的に）初期値として設定され、遊技制御プログラムによらずに（自動的に）W D T 3 1 4 が起動される。

20

#### 【0096】

クリアレジスタは、W D T 3 1 4 のクリアとリスタートを制御するための 8 ビット長のレジスタであり、所定の第一の値（例えば、5 5 H）を設定した後に、所定の第二の値（例えば、A A H）を設定したときに W D T 3 1 4 がクリアされ、直後にリスタートされる。なお、クリアレジスタの値を読み込んだときは、所定の固定値（例えば、F F H）が読みだされる。また、詳細は後述するが、本実施形態では、C P U 3 0 4 の所定記憶領域に、4 ビット長の設定値データ 0 0 0 0 B ~ 1 0 0 0 B のいずれかを予め設定することによって、W D T 3 1 4 のタイムアウト時間を自由に設定できるように構成されている（図 1 1 5 参照）。なお、W D T が手動起動される前（停止された状態で）にクリアレジスタに第一の値、第二の値の順に値が設定されたとしても W D T 3 1 4 は上記したクリアとリスタートの処理を実行しないように構成されている。

30

#### 【0097】

ステップ S 1 0 5 では、低電圧信号がオンであるか否か、すなわち、電圧監視回路 3 3 8 が、電源制御部 6 6 0 から主制御部 3 0 0 に供給している電源の電圧値が所定の値（本実施形態では 9 V）未満である場合に電圧が低下したことを示す低電圧信号を出力しているか否かを監視する。そして、低電圧信号がオンの場合（C P U 3 0 4 が電源の遮断を検知した場合）にはステップ S 1 0 3 に戻り、低電圧信号がオフの場合（C P U 3 0 4 が電源の遮断を検知していない場合）にはステップ S 1 0 7 に進む。なお、電源が投入された直後で未だ上記所定の値（9 V）に達しない場合にもステップ S 1 0 3 に戻り、供給電圧がその所定の値以上になるまで、ステップ S 1 0 5 は繰り返し実行される。

40

#### 【0098】

ステップ S 1 0 7 では、初期設定 2 を行う。この初期設定 2 では、後述する主制御部タイマ割込処理を定期毎に実行するための周期を決める数値をカウンタタイマ 3 1 2 に設定する処理、I / O 3 1 0 の所定のポート（例えば試験用出力ポート、第 1 副制御部 4 0 0 への出力ポート）からクリア信号を出力する処理、R A M 3 0 8 への書き込みを許可する設定等を行う。

#### 【0099】

ステップ S 1 0 9 では、電源の遮断前（電断前）の状態に復帰するか否かの判定を行い、電断前の状態に復帰しない場合（主制御部 3 0 0 の基本回路 3 0 2 を初期状態にする場合）には初期化処理（ステップ S 1 1 3）に進む。

50

## 【0100】

具体的には、最初に、電源基板に設けたRWMクリアスイッチ180を遊技店の店員などが操作した場合に送信されるRAMクリア信号がオン（操作があったことを示す）であるか否か、すなわちRAMクリアが必要であるか否かを判定し、RAMクリア信号がオンの場合（RAMクリアが必要な場合）には、基本回路302を初期状態にすべくステップS113に進む。一方、RAMクリア信号がオフの場合（RAMクリアが必要でない場合）には、RAM308に設けた電源ステータス記憶領域に記憶した電源ステータスの情報を読み出し、この電源ステータスの情報がサスペンドを示す情報であるか否かを判定する。そして、電源ステータスの情報がサスペンドを示す情報でない場合には、基本回路302を初期状態にすべくステップS113に進み、電源ステータスの情報がサスペンドを示す情報である場合には、RAM308の所定の領域（例えば全ての領域）に記憶している1バイトデータを初期値が0である1バイト構成のレジスタに全て加算することによりチェックサムを算出し、算出したチェックサムの結果が特定の値（例えば0）であるか否か（チェックサムの結果が正常であるか否か）を判定する。そして、チェックサムの結果が特定の値（例えば0）の場合（チェックサムの結果が正常である場合）には電断前の状態に復帰すべくステップS111に進み、チェックサムの結果が特定の値（例えば0）以外である場合（チェックサムの結果が異常である場合）には、パチンコ機100を初期状態にすべくステップS113に進む。同様に電源ステータスの情報が「サスペンド」以外の情報を示している場合にもステップS113に進む。

10

## 【0101】

20

ステップS111では、復電時処理を行う。この復電時処理では、電断時にRAM308に設けられたスタックエリアに記憶しておいたスタックポインタの値を読み出し、スタックポインタに再設定（本設定）する。また、後述するように、電断時にRAM308に設けられたレジスタ退避領域に記憶しておいた所定のレジスタの値を読み出し、これらのレジスタに再設定した後、割込許可（EI）の設定を行う。以降、CPU304が、再設定後のスタックポインタやレジスタに基づいて制御プログラムを実行する結果、パチンコ機100は電源断時の状態に復帰する。すなわち、電断直前にタイマ割込処理（後述）に分岐する直前に行った（ステップS115内の所定の）命令の次の命令から処理を再開する。

## 【0102】

30

本実施形態に係るCPU304は、PUSH命令として、「PUSH TI」命令、「PUSH ALL」命令、「PUSH GRP」命令を備えている。「PUSH TI」命令は、T、Iレジスタに記憶されている値を上述のスタックエリアに退避させる命令であり、具体的には、PUSH命令実行時のスタックポインタをSPとした場合、PUSH命令実行時にTレジスタに記憶されている値を、（SP-1）で示されるアドレスに退避させ、PUSH命令実行時にIレジスタに記憶されている値を、（SP-2）で示されるアドレスに退避させる。また、「PUSH ALL」命令は、PUSH命令実行時にT、I、A、F、B、C、D、E、H、L、IX<sub>H</sub>、IX<sub>L</sub>、IY<sub>H</sub>、IY<sub>L</sub>レジスタに記憶されている値を、この順番でスタックエリアに退避させる命令である。また、「PUSH GRP」命令は、PUSH命令実行時にA、F、B、C、D、E、H、Lレジスタに記憶されている値を、この順番でスタックエリアに退避させる命令である。

40

## 【0103】

また、CPU304は、POP命令として、「POP TI」命令、「POP ALL」命令、「POP GRP」命令を備えている。「POP TI」命令は、スタックエリアに記憶されている値をT、Iレジスタに復帰させる（RAM308の記憶領域からT、Iレジスタにデータをロードする）命令であり、具体的には、POP命令実行時のスタックポインタをSPとした場合、POP命令実行時に（SP-1）で示されるアドレスに記憶されている値をTレジスタに復帰させ、POP命令実行時に（SP-2）で示されるアドレスに記憶されている値をIレジスタに復帰させる。また、「POP ALL」命令は、POP命令実行時にスタックエリアに記憶されている値を、T、I、A、F、B、C、

50

D、E、H、L、 $IX_H$ 、 $IX_L$ 、 $IY_H$ 、 $IY_L$ レジスタに復帰させる命令である。また、「PUSH GRP」命令は、POP命令実行時にスタックエリアに記憶された値を、A、F、B、C、D、E、H、Lレジスタに復帰させる命令である。

#### 【0104】

なお、Tレジスタをスタックエリアから復帰させるためには、「POP TI」命令を使用する必要があるが、Iレジスタを使用しない場合でも、Iレジスタにもスタックエリアの値が復帰されてしまうことになる。このように、Tレジスタをオペランドとする命令の汎用性を無くすことによって（使い難くすることによって）、Tレジスタをオペランドとする命令が多用されることを未然に防ぐことができ、Tレジスタの値が不用意に書き替ってしまうような事態（Tレジスタの内容が破壊されてしまうような事態）を回避できる場合がある。

10

#### 【0105】

また、CPU304は、割込み要求の受付を許可する命令（EI命令）と、割込み要求の受付を禁止する命令（DI命令）を備えており、上述の「POP TI」命令は、DI命令によって割込み要求の受付が禁止されている場合にのみ実行するように構成され、EI命令によって割込み要求の受付が許可されている場合に実行しないように構成されている。このように、割込み要求の受付が許可されている状態において「POP TI」命令の実行しないように構成すれば、「POP TI」命令の実行によって、割込みベクタレジスタやインタラプトレジスタとして機能するIレジスタに、スタックエリアから予期しない値が復帰されてしまうおそれがなく、割込み処理を確実に実行することで信頼性の高い遊技制御を行うことができる場合がある。

20

#### 【0106】

また、CPU304は、上述の命令の他にも、8ビットロード（転送）命令、16ビットロード（転送）命令、スタック操作命令（16ビットロード命令の一種）、8ビット算術論理演算命令、16ビット算術論理演算命令、乗算・除算命令、アキュムレータ操作命令、MPUコントロール命令、交換命令、ブロック転送命令、ブロックサーチ命令、ビット操作命令、ローテート・シフト命令、無条件ジャンプ命令・条件付きジャンプ命令、コール命令・リターン命令、入出力命令、複合命令などを備えている。

#### 【0107】

本実施形態では、電断時には、「PUSH GRP」命令によってT、Iレジスタを含まない、A、F、B、C、D、E、H、Lレジスタの値をスタックエリアに退避（記憶）させ、復電時には、「POP GRP」命令によってT、Iレジスタを含まない、A、F、B、C、D、E、H、Lレジスタの値をスタックエリアから復帰させるように構成しており、Tレジスタについては「PUSH GRP」命令による退避と「POP GRP」命令による復帰の両方ができないように構成している（これらの命令のオペランドとしてTレジスタが指定できないように構成している）。また、本実施形態では、加算命令（INC命令）や減算命令（DEC命令）のオペランドとしてもTレジスタが指定できないように構成している。後述するように、TレジスタはRAM308へのアクセスにおいて重要な役割を果たす特殊レジスタであるが、コーディングミスなどによってTレジスタに予期しない値が設定されてしまうような事態を確実に回避することができ、安定した遊技制御が可能である。また、TレジスタについてはPUSH命令による退避とPOP命令による復帰の両方を行わないため、電断時や復帰時の処理を速めることができる場合がある。

30

40

#### 【0108】

また、主制御部300における基本回路302に搭載されているRAM308には、送信情報記憶領域が設けられている。このステップS111では、その送信情報記憶領域に、復電コマンドをセットする。この復電コマンドは、電源断時の状態に復帰したことを表すコマンドであり、後述する、主制御部300のタイマ割込処理におけるステップS233において、第1副制御部400へ送信される。また、ステップS116では、RAMクリアの場合はRAMクリアした後に、RAMクリアしない場合は各レジスタヘデータを復帰させた後に、WDT314の起動開始を行う。

50

## 【0109】

ステップS113では、初期化処理を行う。この初期化処理では、割込禁止(DI)の設定、スタックポインタへのスタック初期値の設定(本設定)、RAM308の全ての記憶領域の初期化などを行う。さらにここで、主制御部300のRAM308に設けられた送信情報記憶領域に正常復帰コマンドをセットする。この正常復帰コマンドは、主制御部300の初期化処理(ステップS113)が行われたことを表すコマンドであり、復電コマンドと同じく、主制御部300のタイマ割込処理におけるステップS233において、第1副制御部400へ送信される。また、ステップS114では、RAMクリアの場合はRAMクリアした後に、RAMクリアしない場合は各レジスタへデータを復帰させた後に、WDT314の起動開始を行う。

10

## 【0110】

ステップS115では、割込禁止の設定を行った後、基本乱数初期値更新処理を行う。この基本乱数初期値更新処理では、普図当選乱数カウンタ、および特図乱数値カウンタの初期値をそれぞれ生成するための2つの初期値生成用乱数カウンタと、普図タイマ乱数値、および特図タイマ乱数値それぞれを生成するための2つの乱数カウンタを更新する。例えば、普図タイマ乱数値として取り得る数値範囲が0~100とすると、RAM308に設けた普図タイマ乱数値を生成するための乱数カウンタ記憶領域から値を取得し、取得した値に1を加算してから元の乱数カウンタ記憶領域に記憶する。このとき、取得した値に1を加算した結果が101であれば0を元の乱数カウンタ記憶領域に記憶する。他の初期値生成用乱数カウンタ、乱数カウンタもそれぞれ同様に更新する。なお、初期値生成用乱数カウンタは、後述するステップS207でも更新する。主制御部300は、所定の周期ごとに開始するタイマ割込処理を行っている間を除いて、ステップS115の処理を繰り返し実行する。

20

## 【0111】

<Tレジチェック処理>

図11(a)は、初期設定1で行うTレジチェック処理の流れを示すフローチャートであり、同図(b)は、Tレジチェック処理のプログラムリストの一例である。

## 【0112】

このTレジチェック処理のステップS131では、Tレジスタの値をAレジスタにセットし(LD A, T)、次のステップS132では、Aレジスタの値が特定の値(F0H)に一致するか否かを判定し(SUB F0H, JP Z, SEI JYOU)、一致しない場合(異常の場合; Aレジスタの値から特定の値(F0H)を減算した値が0以外の値となりZフラグが0になった場合)にはステップS133に進み、一致した場合(正常の場合; Aレジスタの値から特定の値(F0H)を減算した値が0となりZフラグが1になった場合)には、Tレジスタの値が正常であると判定してTレジチェック処理を終了し、後続の処理に進む。ステップS133では、Tレジスタの値が異常であると判定してTレジスタに特定の値(F0H)をセットした後に(LD T, F0H)、Tレジチェック処理を終了して後続の処理に進む。

30

## 【0113】

なお、Tレジチェック処理は図11に示す処理に限定されるものではなく、例えば、図12に示すような処理でもよい。図12(a)は、変形例に係るTレジチェック処理の流れを示すフローチャートであり、同図(b)は、変形例に係るTレジチェック処理のプログラムリストの一例である。

40

## 【0114】

この変形例に係るTレジチェック処理では、上記ステップS133の処理に替えて、ステップS135のエラー処理を行うように構成している。具体的には、ステップS132においてAレジスタの値が特定の値(F0H)に一致しないと判定した場合(異常の場合; Aレジスタの値から特定の値(F0H)を減算した値が0以外の値となりZフラグが0になった場合)にはステップS135に進みエラー処理を行う(SUB F0H, JP NZ, m o E r r o r)。このエラー処理の内容は特に限定されないが、例えば、無限ル

50

ープに入って後続の主制御部メイン処理を実行しないように構成したり、後続の主制御部メイン処理において液晶表示装置やスピーカなどを用いてエラーを報知したりすることが考えられる。なお、Tレジチェック処理の実行タイミングは初期設定1に限らない。また、Tレジチェック処理の実行タイミングは初期設定1（WDTRিস্টার্ট前、低電圧信号～処理前）でのみ実行するようにしてもよい。

#### 【0115】

##### <LD命令>

次に、主制御部300が備えるLD命令について説明する。LD命令（LD OP1, OP2）は、第1オペランドOP1で示されるレジスタに、第2オペランドOP2で示されるレジスタに記憶された値（または、第2オペランドOP2で示されるイミディエイト値（即値または直値ともいう。以下同じ）を記憶するための命令である。例えば、上述のTレジチェック処理の実行前にTレジスタにF0Hが記憶されている状態で「LD A, T」の命令を実行した場合、Aレジスタには第2オペランドで示されるTレジスタに記憶された値、すなわちF0Hが記憶される。また、「LD A, F0H」の命令を実行した場合にも、Aレジスタには第2オペランドで示されるイミディエイト値、すなわちF0Hが記憶される。

#### 【0116】

一方、第1オペランドOP1をTレジスタとするLD命令では、第2オペランドはイミディエイト値に限定されており、第2オペランドにレジスタを設定することは禁止されている（機械語にアセンブルされる過程でエラーとなりアセンブルすることができない）。CPU304用のアセンブラでエラーが発生するのは、CPU304に当該命令が備えられていないからである。上記LD命令の場合と同様に、例えば既述のTレジスタに関する「PUSH GRP」命令による退避と「POP GRP」命令による復帰の両方ができないように構成している（これらの命令のオペランドとしてTレジスタが指定できないように構成している）というのは、アセンブリでエラーとなることであり、CPU304用の命令のオペランドとして指定できない、つまり、CPU304はTレジスタに関し「PUSH GRP」命令と「POP GRP」命令を備えていないのである。

したがって、上述のTレジチェック処理で説明したように、第2オペランドをイミディエイト値とするLD命令（先の例では「LD T, F0H」）のプログラミング（コーディング）は認められるが、第2オペランドにレジスタを設定するLD命令（例えば「LD T, A」）のプログラミング（コーディング）は認められない。このように、第1オペランドOP1をTレジスタとするLD命令では、Tレジスタに対して直値でデータを転送するよう構成されているので、セットしているデータを確認し易くなる場合がある。また、Tレジスタに対して他のレジスタを介してデータを転送できないように構成されているので、Tレジスタの値を不用意に変更することができなくなる結果、コーディングミスの発生を減少させることができる場合がある。

#### 【0117】

図13はプログラムリストからLD命令のみを抜き出して示した図である。この例では、5つのLD命令を使用しているが、例えば、(4)のLD命令「LD B, A」の実行後にBレジスタに記憶される値についてデバッグでトレースを行う場合を考える。この場合、LD命令「LD B, A」の第2オペランドであるAレジスタについて、プログラムリストを遡ってトレースすることが必要であるが、例えば、「LD A」という検索キーでプログラムリストの検索を行った場合、(1)のLD命令「LD A, B」に加えて(3)のLD命令「LD A, H」がヒットすることになる。すなわち、第1オペランドにAレジスタを設定したLD命令は、第2オペランドにレジスタを設定することと、第2オペランドをイミディエイト値とすることの両方が認められているため使用頻度が高く、コーディングの見直し作業や、デバッグ（例えば、Aレジスタのトレースなど）の効率を低下させる一因となり得る。なお、ここでは、最も利用頻度が高いAレジスタについて説明したが、Tレジスタを除く他のレジスタを利用した場合にも同様のことが言える。

#### 【0118】



一方、第1オペランドにTレジスタを設定したLD命令は、第2オペランドにレジスタを設定することが禁止され、第2オペランドをイミディエイト値とすることのみが認められているため、Aレジスタを利用したLD命令よりも相対的に使用頻度が低く、コーディングの見直し作業や、デバッグ（例えば、Tレジスタのトレースなど）の効率を高めることが可能となり、制御プログラムの開発効率が向上する場合がある。なお、Tレジスタは、INC命令やDEC命令（算術論理演算命令）でTレジスタの値を変化させることを可能にしてもよい。

#### 【0119】

##### ＜主制御部タイマ割込処理＞

次に、図14を用いて、主制御部300のCPU304が実行する主制御部タイマ割込処理について説明する。なお、同図は主制御部タイマ割込処理の流れを示すフローチャートである。

#### 【0120】

主制御部300は、所定の周期（本実施形態では約2msに1回）でタイマ割込信号を発生するカウンタタイマ312を備えており、このタイマ割込信号を契機として主制御部タイマ割込処理を所定の周期で開始する。なお、タイマ割込み周期は設定により適宜変更することが可能である。

#### 【0121】

ステップS201では、タイマ割込開始処理を行う。このタイマ割込開始処理では、CPU304の各レジスタの値をスタック領域に一時的に退避する処理などを行う。

#### 【0122】

ステップS203では、WDT314のカウント値が初期設定値（本実施形態では32.8ms）を超えてWDT割込が発生しないように（処理の異常を検出しないように）、WDTを定期的に（本実施形態では、主制御部タイマ割込の周期である約2msに1回）リスタートを行う。

#### 【0123】

ステップS205では、入力ポート状態更新処理を行う。この入力ポート状態更新処理では、I/O310の入力ポートを介して、上述の前面枠扉開放センサや内枠開放センサや下皿満タンセンサ、各種の球検出センサを含む各種センサ320の検出信号を入力して検出信号の有無を監視し、RAM308に各種センサ320ごとに区画して設けた信号状態記憶領域に記憶する。球検出センサの検出信号を例にして説明すれば、前々回のタイマ割込処理（約4ms前）で検出した各々の球検出センサの検出信号の有無の情報を、RAM308に各々の球検出センサごとに区画して設けた前回検出信号記憶領域から読み出し、この情報をRAM308に各々の球検出センサごとに区画して設けた前々回検出信号記憶領域に記憶し、前回のタイマ割込処理（約2ms前）で検出した各々の球検出センサの検出信号の有無の情報を、RAM308に各々の球検出センサごとに区画して設けた今回検出信号記憶領域から読み出し、この情報を上述の前回検出信号記憶領域に記憶する。また、今回検出した各々の球検出センサの検出信号を、上述の今回検出信号記憶領域に記憶する。

#### 【0124】

また、ステップS205では、上述の前々回検出信号記憶領域、前回検出信号記憶領域、および今回検出信号記憶領域の各記憶領域に記憶した各々の球検出センサの検出信号の有無の情報を比較し、各々の球検出センサにおける過去3回分の検出信号の有無の情報が入賞判定パターン情報と一致するか否かを判定する。一個の遊技球が一つの球検出センサを通過する間に、約2msという非常に短い間隔で起動を繰り返すこの主制御部タイマ割込処理は何回か起動する。このため、主制御部タイマ割込処理が起動する度に、上述のステップS205では、同じ遊技球が同じ球検出センサを通過したことを表す検出信号を確認することになる。この結果、上述の前々回検出信号記憶領域、前回検出信号記憶領域、および今回検出信号記憶領域それぞれに、同じ遊技球が同じ球検出センサを通過したことを表す検出信号が記憶される。すなわち、遊技球が球検出センサを通過し始めたときには、前

10

20

30

40

50

々回検出信号無し、前回検出信号有り、今回検出信号有りになる。本実施形態では、球検出センサの誤検出やノイズを考慮して、検出信号無しの後に検出信号が連続して2回記憶されている場合には、入賞があったと判定する。図4に示す主制御部300のROM306には、入賞判定パターン情報（本実施形態では、前々回検出信号無し、前回検出信号有り、今回検出信号有りであることを示す情報）が記憶されている。このステップS205では、各々の球検出センサにおいて過去3回分の検出信号の有無の情報が、予め定めた入賞判定パターン情報（本実施形態では、前々回検出信号無し、前回検出信号有り、今回検出信号有りであることを示す情報）と一致した場合に、一般入賞口226、可変入賞口234、第1特図始動口230、および第2特図始動口232への入球、または普図始動口228の通過があったと判定する。すなわち、これらの入賞口226、234やこれらの始動口230、232、228への入賞があったと判定する。例えば、一般入賞口226への入球を検出する一般入賞口センサにおいて過去3回分の検出信号の有無の情報が上述の入賞判定パターン情報と一致した場合には、一般入賞口226へ入賞があったと判定し、以降の一般入賞口226への入賞に伴う処理を行うが、過去3回分の検出信号の有無の情報が上述の入賞判定パターン情報と一致しなかった場合には、以降の一般入賞口226への入賞に伴う処理を行わずに後続の処理に分岐する。なお、主制御部300のROM306には、入賞判定クリアパターン情報（本実施形態では、前々回検出信号有り、前回検出信号無し、今回検出信号無しであることを示す情報）が記憶されている。入賞が一度あったと判定した後は、各々の球検出センサにおいて過去3回分の検出信号の有無の情報が、その入賞判定クリアパターン情報に一致するまで入賞があったとは判定せず、入賞判定クリアパターン情報に一致すれば、次からは上記入賞判定パターン情報に一致するか否かの判定を行う。なお、入賞判定パターン情報は上記例に限らず適宜変更してもよい。例えば、入賞判定パターン情報として、前々回の1つ前の回での検出信号無し、前々回検出信号有り、前回検出信号有り、今回検出信号有りであってもよいし、前々回の1つ前の回での検出信号有り、前々回検出信号有り、前回検出信号有り、今回検出信号無しであってもよい。

10

20

#### 【0125】

ステップS207およびステップS209では、基本乱数初期値更新処理および基本乱数更新処理を行う。これらの基本乱数初期値更新処理および基本乱数更新処理では、上記ステップS115で行った初期値生成用乱数カウンタの値の更新を行い、次に主制御部300で使用する、普図当選乱数値、特図1乱数値、および特図2乱数値をそれぞれ生成するための2つの乱数カウンタを更新する。例えば、普図当選乱数値として取り得る数値範囲が0～100とすると、RAM308に設けた普図当選乱数値を生成するための乱数カウンタ記憶領域から値を取得し、取得した値に1を加算してから元の乱数カウンタ記憶領域に記憶する。このとき、取得した値に1を加算した結果が101であれば0を元の乱数カウンタ記憶領域に記憶する。また、取得した値に1を加算した結果、乱数カウンタが一周していると判定した場合にはそれぞれの乱数カウンタに対応する初期値生成用乱数カウンタの値を取得し、乱数カウンタの記憶領域にセットする。例えば、0～100の数値範囲で変動する普図当選乱数値生成用の乱数カウンタから値を取得し、取得した値に1を加算した結果が、RAM308に設けた所定の初期値記憶領域に記憶している前回設定した初期値と等しい値（例えば7）である場合に、普図当選乱数値生成用の乱数カウンタに対応する初期値生成用乱数カウンタから値を初期値として取得し、普図当選乱数値生成用の乱数カウンタにセットすると共に、普図当選乱数値生成用の乱数カウンタが次に1周したことを判定するために、今回設定した初期値を上述の初期値記憶領域に記憶しておく。また、普図当選乱数値生成用の乱数カウンタが次に1周したことを判定するための上述の初期値記憶領域とは別に、特図乱数生成用の乱数カウンタが1周したことを判定するための初期値記憶領域をRAM308に設けている。なお、本実施形態では特図1の乱数値を取得するためのカウンタと特図2の乱数値を取得するためのカウンタとを別に設けたが、同一のカウンタを用いてもよい。

30

40

#### 【0126】

50

ステップS 2 1 1では、演出乱数更新処理を行う。この演出乱数更新処理では、主制御部3 0 0で使用する演出用乱数値を生成するための乱数カウンタを更新する。

#### 【0 1 2 7】

ステップS 2 1 3では、タイマ更新処理を行う。このタイマ更新処理では、普通図柄表示装置2 1 0に図柄を変動・停止表示する時間を計時するための普図表示図柄更新タイマ、第1特別図柄表示装置2 1 2に図柄を変動・停止表示する時間を計時するための特図1表示図柄更新タイマ、第2特図表示装置2 1 4に図柄を変動・停止表示する時間を計時するための特図2表示図柄更新タイマ、所定の入賞演出時間、所定の開放時間、所定の閉鎖時間、所定の終了演出期間などを計時するためのタイマなどを含む各種タイマを更新する。

10

#### 【0 1 2 8】

ステップS 2 1 5では、入賞口カウンタ更新処理を行う。この入賞口カウンタ更新処理では、入賞口2 2 6、2 3 4や始動口2 3 0、2 3 2、2 2 8に入賞があった場合に、RAM 3 0 8に各入賞口ごと、あるいは各始動口ごとに設けた賞球数記憶領域の値を読み出し、1を加算して、元の賞球数記憶領域に設定する。

#### 【0 1 2 9】

また、ステップS 2 1 7では、入賞受付処理を行う。この入賞受付処理では、第1特図始動口2 3 0、第2特図始動口2 3 2、普図始動口2 2 8および可変入賞口2 3 4への入賞があったか否かを判定する。ここでは、ステップS 2 0 5における入賞判定パターン情報と一致するか否かの判定結果を用いて判定する。第1特図始動口2 3 0へ入賞があった場合且つRAM 3 0 8に設けた対応する保留数記憶領域が満タンでない場合、カウンタ回路3 1 8の当選用カウンタ値記憶用レジスタから値を特図1当選乱数値として取得するとともに特図1乱数値生成用の乱数カウンタから値を特図1乱数値として取得して対応する乱数値記憶領域に格納する。第2特図始動口2 3 2へ入賞があった場合且つRAM 3 0 8に設けた対応する保留数記憶領域が満タンでない場合、カウンタ回路3 1 8の当選用カウンタ値記憶用レジスタから値を特図2当選乱数値として取得するとともに特図2乱数値生成用の乱数カウンタから値を特図2乱数値として取得して対応する乱数値記憶領域に格納する。普図始動口2 2 8へ入賞があった場合且つRAM 3 0 8に設けた対応する保留数記憶領域が満タンでない場合、普図当選乱数値生成用の乱数カウンタから値を普図当選乱数値として取得して対応する乱数値記憶領域に格納する。可変入賞口2 3 4へ入賞があった場合には、可変入賞口用の入賞記憶領域に、可変入賞口2 3 4に球が入球したことを示す情報を格納する。

20

30

#### 【0 1 3 0】

ステップS 2 1 9では、払出要求数送信処理を行う。なお、払出制御部6 0 0に出力する出力予定情報および払出要求情報は、例えば1バイトで構成しており、ビット7にストロープ情報（オンの場合、データをセットしていることを示す）、ビット6に電源投入情報（オンの場合、電源投入後一回目のコマンド送信であることを示す）、ビット4～5に暗号化のための今回加工種別（0～3）、およびビット0～3に暗号化加工後の払出要求数を示すようにしている。

#### 【0 1 3 1】

ステップS 2 2 1では、普図状態更新処理を行う。この普図状態更新処理は、普図の状態に対応する複数の処理のうちの1つの処理を行う。例えば、普図変動表示の途中（上述する普図表示図柄更新タイマの値が1以上）における普図状態更新処理では、普通図柄表示装置2 1 0を構成する7セグメントLEDの点灯と消灯を繰り返す点灯・消灯駆動制御を行う。この制御を行うことで、普通図柄表示装置2 1 0は普図の変動表示（普図変動遊技）を行う。

40

#### 【0 1 3 2】

また、普図変動表示時間が経過したタイミング（普図表示図柄更新タイマの値が1から0になったタイミング）における普図状態更新処理では、当りフラグがオンの場合には、当り図柄の表示態様となるように普通図柄表示装置2 1 0を構成する7セグメントLED

50

の点灯・消灯駆動制御を行い、当りフラグがオフの場合には、外れ図柄の表示態様となるように普通図柄表示装置 210 を構成する 7 セグメント LED の点灯・消灯駆動制御を行う。また、主制御部 300 の RAM 308 には、普図状態更新処理に限らず各種の処理において各種の設定を行う設定領域が用意されている。ここでは、上記点灯・消灯駆動制御を行うとともに、その設定領域に普図停止表示中であることを示す設定を行う。この制御を行うことで、普通図柄表示装置 210 は、当り図柄（図 5（c）に示す普図 A）および外れ図柄（図 5（c）に示す普図 B）いずれか一方の図柄の確定表示を行う。さらにその後、所定の停止表示期間（例えば 500 m 秒間）、その表示を維持するために RAM 308 に設けた普図停止時間管理用タイマの記憶領域に停止期間を示す情報を設定する。この設定により、確定表示された図柄が所定期間停止表示され、普図変動遊技の結果が遊技者に報知される。

10

#### 【0133】

また、普図変動遊技の結果が当りであれば、後述するように、普図当りフラグがオンされる。この普図当りフラグがオンの場合には、所定の停止表示期間が終了したタイミング（普図停止時間管理用タイマの値が 1 から 0 になったタイミング）における普図状態更新処理では、RAM 308 の設定領域に普図作動中を設定するとともに、所定の開放期間（例えば 2 秒間）、第 2 特図始動口 232 の羽根部材 232 a の開閉駆動用のソレノイド（332）に、羽根部材 232 a を開放状態に保持する信号を出力するとともに、RAM 308 に設けた羽根開放時間管理用タイマの記憶領域に開放期間を示す情報を設定する。

#### 【0134】

20

また、所定の開放期間が終了したタイミング（羽根開放時間管理用タイマの値が 1 から 0 になったタイミング）で開始する普図状態更新処理では、所定の閉鎖期間（例えば 500 m 秒間）、羽根部材の開閉駆動用のソレノイド 332 に、羽根部材を閉鎖状態に保持する信号を出力するとともに、RAM 308 に設けた羽根閉鎖時間管理用タイマの記憶領域に閉鎖期間を示す情報を設定する。

#### 【0135】

また、所定の閉鎖期間が終了したタイミング（羽根閉鎖時間管理用タイマの値が 1 から 0 になったタイミング）で開始する普図状態更新処理では、RAM 308 の設定領域に普図非作動中を設定する。さらに、普図変動遊技の結果が外れであれば、後述するように、普図外れフラグがオンされる。この普図外れフラグがオンの場合には、上述した所定の停止表示期間が終了したタイミング（普図停止時間管理用タイマの値が 1 から 0 になったタイミング）における普図状態更新処理でも、RAM 308 の設定領域に普図非作動中を設定する。普図非作動中の場合における普図状態更新処理では、何もせずに次のステップ S223 に移行するようにしている。

30

#### 【0136】

ステップ S223 では、普図関連抽選処理を行う。この普図関連抽選処理では、普図変動遊技および第 2 特図始動口 232 の開閉制御を行っておらず（普図の状態が非作動中）、且つ、保留している普図変動遊技の数が 1 以上である場合に、上述の乱数値記憶領域に記憶している普図当選乱数値に基づいた乱数抽選により普図変動遊技の結果を当選とするか、不当選とするかを決定する当り判定をおこない、当選とする場合には RAM 308 に設けた当りフラグにオンを設定する。不当選の場合には、当りフラグにオフを設定する。また、当り判定の結果に関わらず、次に上述の普図タイマ乱数値生成用の乱数カウンタの値を普図タイマ乱数値として取得し、取得した普図タイマ乱数値に基づいて複数の変動時間のうちから普図表示装置 210 に普図を変動表示する時間を 1 つ選択し、この変動表示時間を、普図変動表示時間として、RAM 308 に設けた普図変動時間記憶領域に記憶する。なお、保留している普図変動遊技の数は、RAM 308 に設けた普図保留数記憶領域に記憶するようにしており、当り判定をするたびに、保留している普図変動遊技の数から 1 を減算した値を、この普図保留数記憶領域に記憶し直すようにしている。また当り判定に使用した乱数値を消去する。

40

#### 【0137】

50

次いで、特図 1 および特図 2 それぞれについての特図状態更新処理を行うが、最初に、特図 2 についての特図状態更新処理（特図 2 状態更新処理）を行う（ステップ S 2 2 5）。この特図 2 状態更新処理は、特図 2 の状態に応じて、次の 8 つの処理のうちの 1 つの処理を行う。例えば、特図 2 変動表示の途中（上述の特図 2 表示図柄更新タイマの値が 1 以上）における特図 2 状態更新処理では、第 2 特別図柄表示装置 2 1 4 を構成する 7 セグメント L E D の点灯と消灯を繰り返す点灯・消灯駆動制御を行う。この制御を行うことで、第 2 特別図柄表示装置 2 1 4 は特図 2 の変動表示（特図 2 変動遊技）を行う。また、コマンド設定送信処理（ステップ S 2 3 3）で回転開始設定送信処理を実行させることを示す所定の送信情報を上述の送信情報記憶領域に追加記憶してから処理を終了する。

#### 【0 1 3 8】

また、主制御部 3 0 0 の R A M 3 0 8 の当否判定結果記憶領域には、1 5 R 大当りフラグ、2 R 大当りフラグ、第 1 小当りフラグ、第 2 小当りフラグ、第 1 はずれフラグ、第 2 はずれフラグ、特図確率変動フラグ、および普図確率変動フラグそれぞれのフラグが用意されている。特図 2 変動表示時間が経過したタイミング（特図 2 表示図柄更新タイマの値が 1 から 0 になったタイミング）で開始する特図 2 状態更新処理では、1 5 R 大当りフラグはオン、特図確率変動フラグもオン、普図確率変動フラグもオンの場合には図 5（a）に示す特図 A、1 5 R 大当りフラグはオン、特図確率変動フラグはオフ、普図確率変動フラグはオンの場合には特図 B、2 R 大当りフラグはオン、特図確率変動フラグもオン、普図確率変動フラグもオンの場合には特図 C、2 R 大当りフラグはオン、特図確率変動フラグはオフ、普図確率変動フラグはオンの場合には特図 D、2 R 大当りフラグはオン、特図確率変動フラグもオン、普図確率変動フラグはオンの場合には特図 E、2 R 大当りフラグはオン、特図確率変動フラグはオフ、普図確率変動フラグもオフの場合には特図 F、第 1 小当りフラグがオンの場合には特図 G、第 2 小当りフラグがオンの場合には特図 H、第 1 はずれフラグがオンの場合には特図 I、第 2 はずれフラグがオンの場合には特図 I それぞれの態様となるように、第 2 特別図柄表示装置 2 1 4 を構成する 7 セグメント L E D の点灯・消灯駆動制御を行い、R A M 3 0 8 の設定領域に特図 2 停止表示中であることを表す設定を行う。この制御を行うことで、第 2 特別図柄表示装置 2 1 4 は、1 5 R 特別大当り図柄（特図 A）、1 5 R 大当り図柄（特図 B）、突然確変図柄（特図 C）、突然時短図柄（特図 D）、隠れ確変図柄（特図 E）、突然通常図柄（特図 F）、第 1 小当り図柄（特図 G）、第 2 小当り図柄（特図 H）、第 1 はずれ図柄（特図 I）、および第 2 はずれ図柄（特図 J）のいずれか一つの図柄の確定表示を行う。さらにその後、所定の停止表示期間（例えば 5 0 0 m 秒間）その表示を維持するために R A M 3 0 8 に設けた特図 2 停止時間管理用タイマの記憶領域に停止期間を示す情報を設定する。この設定により、確定表示された特図 2 が所定期間停止表示され、特図 2 変動遊技の結果が遊技者に報知される。また、R A M 3 0 8 に設けられた時短回数記憶部に記憶された時短回数が 1 以上であれば、その時短回数から 1 を減算し、減算結果が 1 から 0 となった場合は、特図確率変動中（詳細は後述）でなければ、時短フラグをオフする。さらに、大当り遊技中（特別遊技状態中）にも、時短フラグをオフする。

#### 【0 1 3 9】

また、コマンド設定送信処理（ステップ S 2 3 3）で回転停止設定送信処理を実行させることを示す所定の送信情報を上述の送信情報記憶領域に追加記憶するとともに、変動表示を停止する図柄が特図 2 であることを示す特図 2 識別情報を、後述するコマンドデータに含める情報として R A M 3 0 8 に追加記憶してから処理を終了する。

#### 【0 1 4 0】

また、特図 2 変動遊技の結果が大当りであれば、後述するように、大当りフラグがオンされる。この大当りフラグがオンの場合には、所定の停止表示期間が終了したタイミング（特図 2 停止時間管理用タイマの値が 1 から 0 になったタイミング）における特図 2 状態更新処理では、R A M 3 0 8 の設定領域に特図 2 作動中を設定するとともに、所定の入賞演出期間（例えば 3 秒間）すなわち装飾図柄表示装置 2 0 8 による大当りを開始することを遊技者に報知する画像を表示している期間待機するために R A M 3 0 8 に設けた特図 2

待機時間管理用タイマの記憶領域に入賞演出期間を示す情報を設定する。また、コマンド設定送信処理（ステップS 2 3 3）で入賞演出設定送信処理を実行させることを示す所定の送信情報を上述の送信情報記憶領域に追加記憶する。

#### 【0141】

また、所定の入賞演出期間が終了したタイミング（特図2待機時間管理用タイマの値が1から0になったタイミング）で開始する特図2状態更新処理では、所定の開放期間（例えば29秒間、または可変入賞口234に所定球数（例えば10球）の遊技球の入賞を検出するまで）可変入賞口234の扉部材234aの開閉駆動用のソレノイド（332）に、扉部材234aを開放状態に保持する信号を出力するとともに、RAM308に設けた扉開放時間管理用タイマの記憶領域に開放期間を示す情報を設定する。また、コマンド設定送信処理（ステップS 2 3 3）で大入賞口開放設定送信処理を実行させることを示す所定の送信情報を上述の送信情報記憶領域に追加記憶する。

10

#### 【0142】

また、所定の開放期間が終了したタイミング（扉開放時間管理用タイマの値が1から0になったタイミング）で開始する特図2状態更新処理では、所定の閉鎖期間（例えば1.5秒間）可変入賞口234の扉部材234aの開閉駆動用のソレノイド（332）に、扉部材234aを閉鎖状態に保持する信号を出力するとともに、RAM308に設けた扉閉鎖時間管理用タイマの記憶領域に閉鎖期間を示す情報を設定する。また、コマンド設定送信処理（ステップS 2 3 3）で大入賞口閉鎖設定送信処理を実行させることを示す所定の送信情報を上述の送信情報記憶領域に追加記憶する。

20

#### 【0143】

また、この扉部材の開放・閉鎖制御を所定回数（本実施例では15ラウンドか2ラウンド）繰り返し、終了したタイミングで開始する特図2状態更新処理では、所定の終了演出期間（例えば3秒間）すなわち装飾図柄表示装置208による大当りを終了することを遊技者に報知する画像を表示している期間待機するように設定するためにRAM308に設けた演出待機時間管理用タイマの記憶領域に演出待機期間を示す情報を設定する。また、普図確率変動フラグがオンに設定されていれば、この大当り遊技の終了と同時に、RAM308に設けられた時短回数記憶部に時短回数100回をセットするとともに、RAM308に設けられた時短フラグをオンする。なお、その普図確率変動フラグがオフに設定されていれば、時短回数記憶部に時短回数をセットすることなく、また時短フラグをオンすることもない。ここにいう時短とは、特図変動遊技における大当りを終了してから、次の大当りを開始するまでの時間を短くするため、パチンコ機が遊技者にとって有利な状態になることをいう。この時短フラグがオンに設定されていると、普図高確率状態である。普図高確率状態では普図低確率状態に比べて、普図変動遊技に大当たりする可能性が高い。また、普図高確率状態の方が、普図低確率状態に比べて普図変動遊技の変動時間および特図変動遊技の変動時間は短くなる。さらに、普図高確率状態では普図低確率状態に比べて、第2特別始動口232の一对の羽根部材232aの1回の開放における開放時間が長くなりやすい。加えて、普図高確率状態では普図低確率状態に比べて、一对の羽根部材232aは多く開きやすい。また、上述のごとく、時短フラグは、大当り遊技中（特別遊技状態中）にはオフに設定される。したがって、大当り遊技中には、普図低確率状態が維持される。これは、大当り遊技中に普図高確率状態であると、大当り遊技中に可変入賞口234に所定の個数、遊技球が入球するまでの間に第2特図始動口232に多くの遊技球が入球し、大当り中に獲得することができる遊技球の数が多くなってしまい射幸性が高まってしまいうという問題があり、これを解決するためのものである。

30

40

#### 【0144】

さらに、コマンド設定送信処理（ステップS 2 3 3）で終了演出設定送信処理を実行させることを示す所定の送信情報を上述の送信情報記憶領域に追加記憶する。

#### 【0145】

また、所定の終了演出期間が終了したタイミング（演出待機時間管理用タイマの値が1から0になったタイミング）で開始する特図2状態更新処理では、RAM308の設定領

50

域に特図 2 非作動中を設定する。さらに、特図 2 変動遊技の結果が外れであれば、後述するように、はずれフラグがオンされる。このはずれフラグがオンの場合には、上述した所定の停止表示期間が終了したタイミング（特図 2 停止時間管理用タイマの値が 1 から 0 になったタイミング）における特図 2 状態更新処理でも、RAM 308 の設定領域に特図 2 非作動中を設定する。特図 2 非作動中の場合における特図 2 状態更新処理では、何もせずに次のステップ S 2 2 7 に移行するようにしている。

#### 【0146】

続いて、特図 1 についての特図状態更新処理（特図 1 状態更新処理）を行う（ステップ S 2 2 7）。この特図 1 状態更新処理では、特図 1 の状態に応じて、上述の特図 2 状態更新処理で説明した各処理を行う。この特図 1 状態更新処理で行う各処理は、上述の特図 2 状態更新処理で説明した内容の「特図 2」を「特図 1」と読み替えた処理と同一であるため、その説明は省略する。なお、特図 2 状態更新処理と特図 1 状態更新処理の順番は逆でもよい。

#### 【0147】

ステップ S 2 2 5 およびステップ S 2 2 7 における特図状態更新処理が終了すると、今度は、特図 1 および特図 2 それぞれについての特図関連抽選処理を行う。ここでも先に、特図 2 についての特図関連抽選処理（特図 2 関連抽選処理）を行い（ステップ S 2 2 9）、その後で、特図 1 についての特図関連抽選処理（特図 1 関連抽選処理）を行う（ステップ S 2 3 1）。これらの特図関連抽選処理についても、主制御部 300 が特図 2 関連抽選処理を特図 1 関連抽選処理よりも先に行うことで、特図 2 変動遊技の開始条件と、特図 1 変動遊技の開始条件が同時に成立した場合でも、特図 2 変動遊技が先に変動中となるため、特図 1 変動遊技は変動を開始しない。また、装飾図柄表示装置 208 による、特図変動遊技の大当たり判定の結果の報知は、第 1 副制御部 400 によって行われ、第 2 特図始動口 232 への入賞に基づく抽選の抽選結果の報知が、第 1 特図始動口 230 への入賞に基づく抽選の抽選結果の報知よりも優先して行われる。

#### 【0148】

ステップ S 2 3 3 では、コマンド設定送信処理を行い、各種のコマンドが第 1 副制御部 400 に送信される。なお、第 1 副制御部 400 に送信する出力予定情報は例えば 16 ビットで構成しており、ビット 15 はストロブ情報（オンの場合、データをセットしていることを示す）、ビット 11～14 はコマンド種別（本実施形態では、基本コマンド、図柄変動開始コマンド、図柄変動停止コマンド、入賞演出開始コマンド、終了演出開始コマンド、大当たりラウンド数指定コマンド、復電コマンドなどコマンドの種類を特定可能な情報）、ビット 0～10 はコマンドデータ（コマンド種別に対応する所定の情報）で構成している。

#### 【0149】

具体的には、ストロブ情報は上述のコマンド送信処理でオン、オフするようにしている。また、コマンド種別が図柄変動開始コマンドの場合であればコマンドデータに、15 R 大当たりフラグや 2 R 大当たりフラグの値、特図確率変動フラグの値、特図関連抽選処理で選択したタイマ番号などを示す情報を含み、図柄変動停止コマンドの場合であれば、15 R 大当たりフラグや 2 R 大当たりフラグの値、特図確率変動フラグの値などを含み、入賞演出コマンドおよび終了演出開始コマンドの場合であれば、特図確率変動フラグの値などを含み、大当たりラウンド数指定コマンドの場合であれば特図確率変動フラグの値、大当たりラウンド数などを含むようにしている。コマンド種別が基本コマンドを示す場合は、コマンドデータにデバイス情報、第 1 特図始動口 230 への入賞の有無、第 2 特図始動口 232 への入賞の有無、可変入賞口 234 への入賞の有無などを含む。

#### 【0150】

また、上述の回転開始設定送信処理では、コマンドデータに RAM 308 に記憶している、15 R 大当たりフラグや 2 R 大当たりフラグの値、特図確率変動フラグの値、特図 1 関連抽選処理および特図 2 関連抽選処理で選択したタイマ番号、保留している第 1 特図変動遊技または第 2 特図変動遊技の数などを示す情報を設定する。上述の回転停止設定送信処理

10

20

30

40

50

では、コマンドデータにRAM 308に記憶している、15R大当りフラグや2R大当りフラグの値、特図確率変動フラグの値などを示す情報を設定する。上述の入賞演出設定送信処理では、コマンドデータに、RAM 308に記憶している、入賞演出期間中に装飾図柄表示装置208・各種ランプ418・スピーカ120に出力する演出制御情報、特図確率変動フラグの値、保留している第1特図変動遊技または第2特図変動遊技の数などを示す情報を設定する。上述の終了演出設定送信処理では、コマンドデータに、RAM 308に記憶している、演出待機期間中に装飾図柄表示装置208・各種ランプ418・スピーカ120に出力する演出制御情報、特図確率変動フラグの値、保留している第1特図変動遊技または第2特図変動遊技の数などを示す情報を設定する。上述の大入賞口開放設定送信処理では、コマンドデータにRAM 308に記憶している大当りラウンド数、特図確率変動フラグの値、保留している第1特図変動遊技または第2特図変動遊技の数などを示す情報を設定する。上述の大入賞口閉鎖設定送信処理では、コマンドデータにRAM 308に記憶している大当りラウンド数、特図確率変動フラグの値、保留している第1特図変動遊技または第2特図変動遊技の数などを示す情報を設定する。また、このステップS233では一般コマンド特図保留増加処理も行われる。この一般コマンド特図保留増加処理では、コマンドデータにRAM 308の送信用情報記憶領域に記憶している特図識別情報（特図1または特図2を示す情報）、予告情報（事前予告情報、偽事前予告情報、または事前予告無情報のいずれか）を設定する。

10

#### 【0151】

第1副制御部400では、受信した出力予定情報に含まれるコマンド種別により、主制御部300における遊技制御の変化に応じた演出制御の決定が可能になるとともに、出力予定情報に含まれているコマンドデータの情報に基づいて、演出制御内容を決定することができるようになる。

20

#### 【0152】

ステップS235では、外部出力信号設定処理を行う。この外部出力信号設定処理では、RAM 308に記憶している遊技情報を、情報出力回路336を介してパチンコ機100とは別体の情報入力回路350に出力する。

#### 【0153】

ステップS237では、デバイス監視処理を行う。このデバイス監視処理では、ステップS205において信号状態記憶領域に記憶した各種センサの信号状態を読み出して、所定のエラーの有無、例えば前面枠扉開放エラーの有無または下皿満タンエラーの有無などを監視し、前面枠扉開放エラーまたは下皿満タンエラーを検出した場合に、第1副制御部400に送信すべき送信情報に、前面枠扉開放エラーの有無または下皿満タンエラーの有無を示すデバイス情報を設定する。また、各種ソレノイド332を駆動して第2特図始動口232や、可変入賞口234の開閉を制御したり、表示回路324、326、330を介して普通図柄表示装置210、第1特別図柄表示装置212、第2特別図柄表示装置214、各種状態表示部328などに出力する表示データを、I/O310の出力ポートに設定する。また、払出要求数送信処理（ステップS219）で設定した出力予定情報を出力ポート（I/O310）を介して第1副制御部400に出力する。

30

#### 【0154】

ステップS239では、低電圧信号がオンであるか否かを監視する。そして、低電圧信号がオンの場合（電源の遮断を検知した場合）にはステップS243に進み、低電圧信号がオフの場合（電源の遮断を検知していない場合）にはステップS241に進む。

40

#### 【0155】

ステップS241では、タイマ割込終了処理を行う。このタイマ割込終了処理では、ステップS201で一時的に退避した各レジスタの値を元の各レジスタに設定したり、割込許可の設定などを行い、その後、主制御部メイン処理に復帰する。

#### 【0156】

一方、ステップS243では、復電時に電断時の状態に復帰するための特定の変数やスタックポインタを復帰データとしてRAM 308の所定の領域に退避し、入出力ポートの

50



初期化等の電断処理を行い、その後、主制御部メイン処理に復帰する。

#### 【0157】

＜第1副制御部400の処理＞

次に、図15を用いて、第1副制御部400の処理について説明する。なお、同図(a)は、第1副制御部400のCPU404が実行するメイン処理のフローチャートである。同図(b)は、第1副制御部400のストロブ割込み処理のフローチャートである。同図(c)は、第1副制御部400のタイマ変数更新割込処理のフローチャートである。同図(d)は、第1副制御部400の画像制御処理のフローチャートである。

#### 【0158】

まず、同図(a)のステップS301では、各種の初期設定を行う。電源投入が行われると、まずS301で初期化処理が実行される。この初期化処理では、入出力ポートの初期設定や、RAM408内の記憶領域の初期化処理等を行う。

#### 【0159】

ステップS303では、タイマ変数が10以上か否かを判定し、タイマ変数が10となるまでこの処理を繰り返し、タイマ変数が10以上となったときには、ステップS305の処理に移行する。ステップS305では、タイマ変数に0を代入する。

#### 【0160】

ステップS307では、コマンド処理を行う。第1副制御部400のCPU404は、主制御部300からコマンドを受信したか否かを判別する。

#### 【0161】

ステップS309では、演出制御処理を行う。例えば、S307で新たなコマンドがあった場合には、このコマンドに対応する演出データをROM406から読み出す等の処理を行い、演出データの更新が必要な場合には演出データの更新処理を行う。

#### 【0162】

ステップS311では、チャンスボタンの押下を検出していた場合、ステップS309で更新した演出データをチャンスボタンの押下に応じた演出データに変更する処理を行う。ステップS313では、S309で読み出した演出データの中にVDP434への命令がある場合には、この命令をVDP434に出力する(詳細は後述)。

#### 【0163】

ステップS315では、S309で読み出した演出データの中に音源IC416への命令がある場合には、この命令を音源IC416に出力する。ステップS317では、S309で読み出した演出データの中に各種ランプ418への命令がある場合には、この命令を駆動回路420に出力する。

#### 【0164】

ステップS319では、S309で読み出した演出データの中に遮蔽装置246への命令がある場合には、この命令を駆動回路432に出力する。ステップS321では、S309で読み出した演出データの中に第2副制御部500に送信する制御コマンドがある場合には、この制御コマンドを出力する設定を行い、S303へ戻る。

#### 【0165】

次に、同図(b)を用いて、第1副制御部400のコマンド受信割込処理について説明する。このコマンド受信割込処理は、第1副制御部400が、主制御部300が出力するストロブ信号を検出した場合に実行する処理である。コマンド受信割込処理のステップS401では、主制御部300が出力したコマンドを未処理コマンドとしてRAM408に設けたコマンド記憶領域に記憶する。

#### 【0166】

次に、同図(c)を用いて、第1副制御部400のCPU404によって実行する第1副制御部タイマ割込処理について説明する。第1副制御部400は、所定の周期(本実施例では2msに1回)でタイマ割込を発生するハードウェアタイマを備えており、このタイマ割込を契機として、タイマ割込処理を所定の周期で実行する。

#### 【0167】

10

20

30

40

50

第1副制御部タイマ割込処理のステップS501では、第1副制御部メイン処理におけるステップS303において説明したRAM408のタイマ変数記憶領域の値に、1を加算して元のタイマ変数記憶領域に記憶する。従って、ステップS303において、タイマ変数の値が10以上と判定されるのは20ms毎(2ms×10)となる。

#### 【0168】

第1副制御部タイマ割込処理のステップS503では、ステップS319で設定された第2副制御部500への制御コマンドの送信や、演出用乱数値の更新処理等を行う。

#### 【0169】

次に、同図(d)を用いて、第1副制御部400のメイン処理におけるステップS313の画像制御処理について説明する。同図は、画像制御処理の流れを示すフローチャートを示した図である。

10

#### 【0170】

ステップS601では、画像データの転送指示を行う。ここでは、CPU404は、まず、VRAM436の表示領域Aと表示領域Bの描画領域の指定をスワップする。これにより、描画領域に指定されていない表示領域に記憶された1フレームの画像が装飾図柄表示装置208に表示される。次に、CPU404は、VDP434のアトリビュートレジスタに、位置情報等テーブルに基づいてROM座標(ROM406の転送元アドレス)、VRAM座標(VRAM436の転送先アドレス)などを設定した後、ROM406からVRAM436への画像データの転送開始を指示する命令を設定する。VDP434は、アトリビュートレジスタに設定された命令に基づいて画像データをROM406からVRAM436に転送する。その後、VDP436は、転送終了割込信号をCPU404に対して出力する。

20

#### 【0171】

ステップS603では、VDP434からの転送終了割込信号が入力されたか否かを判定し、転送終了割込信号が入力された場合はステップS605に進み、そうでない場合は転送終了割込信号が入力されるのを待つ。ステップS605では、演出シナリオ構成テーブルおよびアトリビュートデータなどに基づいて、パラメータ設定を行う。ここでは、CPU404は、ステップS601でVRAM436に転送した画像データに基づいてVRAM436の表示領域AまたはBに表示画像を形成するために、表示画像を構成する画像データの情報(VRAM436の座標軸、画像サイズ、VRAM座標(配置座標)など)をVDP434に指示する。VDP434はアトリビュートレジスタに格納された命令に基づいてアトリビュートに従ったパラメータ設定を行う。

30

#### 【0172】

ステップS607では、描画指示を行う。この描画指示では、CPU404は、VDP434に画像の描画開始を指示する。VDP434は、CPU404の指示に従ってフレームバッファにおける画像描画を開始する。

#### 【0173】

ステップS609では、画像の描画終了に基づくVDP434からの生成終了割込み信号が入力されたか否かを判定し、生成終了割込み信号が入力された場合はステップS611に進み、そうでない場合は生成終了割込み信号が入力されるのを待つ。ステップS611では、RAM408の所定の領域に設定され、何シーンの画像を生成したかをカウントするシーン表示カウンタをインクリメント(+1)して処理を終了する。

40

#### 【0174】

<第2副制御部500の処理>

次に、図16を用いて、第2副制御部500の処理について説明する。なお、同図(a)は、第2副制御部500のCPU504が実行するメイン処理のフローチャートである。同図(b)は、第2副制御部500のコマンド受信割込処理のフローチャートである。同図(c)は、第2副制御部500のタイマ割込処理のフローチャートである。

#### 【0175】

まず、同図(a)のステップS701では、各種の初期設定を行う。電源投入が行われ

50

ると、まず S 7 0 1 で初期化処理が実行される。この初期化処理では、入出力ポートの初期設定や、R A M 5 0 8 内の記憶領域の初期化処理等を行う。

#### 【0176】

ステップ S 7 0 3 では、タイマ変数が 1 0 以上か否かを判定し、タイマ変数が 1 0 となるまでこの処理を繰り返し、タイマ変数が 1 0 以上となったときには、ステップ S 7 0 5 の処理に移行する。

#### 【0177】

ステップ S 7 0 5 では、タイマ変数に 0 を代入する。ステップ S 7 0 7 では、コマンド処理を行う。第 2 副制御部 5 0 0 の C P U 5 0 4 は、第 1 副制御部 4 0 0 の C P U 4 0 4 からコマンドを受信したか否かを判別する。ステップ S 7 0 9 では、演出制御処理を行う。例えば、S 7 0 7 で新たなコマンドがあった場合には、このコマンドに対応する演出データを R O M 5 0 6 から読み出す等の処理を行い、演出データの更新が必要な場合には演出データの更新処理を行う。

#### 【0178】

ステップ S 7 1 1 では、第 1 副制御部 4 0 0 からの遊技盤用ランプ 5 3 2 や遊技台枠用ランプ 5 4 2 への命令がある場合には、この命令をシリアル通信制御回路 5 2 0 に出力する。

#### 【0179】

ステップ S 7 1 3 では、第 1 副制御部 4 0 0 からの演出可動体 2 2 4 への命令がある場合には、この命令を駆動回路 5 1 6 に出力し、S 7 0 3 に戻る。

次に、同図 (b) を用いて、第 2 副制御部 5 0 0 のコマンド受信割込処理について説明する。このコマンド受信割込処理は、第 2 副制御部 5 0 0 が、第 1 副制御部 4 0 0 が出力するストローブ信号を検出した場合に実行する処理である。コマンド受信割込処理のステップ S 8 0 1 では、第 1 副制御部 4 0 0 が出力したコマンドを未処理コマンドとして R A M 5 0 8 に設けたコマンド記憶領域に記憶する。

#### 【0180】

次に、同図 (c) を用いて、第 2 副制御部 5 0 0 の C P U 5 0 4 によって実行する第 2 副制御部タイマ割込処理について説明する。第 2 副制御部 5 0 0 は、所定の周期（本実施例では 2 m s に 1 回）でタイマ割込を発生するハードウェアタイマを備えており、このタイマ割込を契機として、タイマ割込処理を所定の周期で実行する。

#### 【0181】

第 2 副制御部タイマ割込処理のステップ S 9 0 1 では、第 2 副制御部メイン処理におけるステップ S 7 0 3 において説明した R A M 5 0 8 のタイマ変数記憶領域の値に、1 を加算して元のタイマ変数記憶領域に記憶する。従って、ステップ S 7 0 3 において、タイマ変数の値が 1 0 以上と判定されるのは 2 0 m s 毎（2 m s × 1 0）となる。第 2 副制御部タイマ割込処理のステップ S 9 0 3 では、演出用乱数値の更新処理等を行う。

#### 【0182】

##### <主制御部の R O M>

次に、図 1 7 を用いて、主制御部 3 0 0 の R O M 3 0 6 に記憶されるデータの種類のついて説明する。なお、同図は、上述の R O M 3 0 6 に記憶されるデータの一例を示した図である。

#### 【0183】

R O M 3 0 6 の記憶領域に対応する R O M 領域（本実施形態では、0 0 0 0 H ~ 2 F F F H の 1 6 K バイト領域）は、R O M 制御領域（本実施形態では、0 0 0 0 H ~ 0 B B 0 H）、非使用領域（本実施形態では、0 B B 1 H ~ 0 F F F H）、R O M データ領域（本実施形態では、1 0 0 0 H ~ 1 8 F D H）およびその他領域（本実施形態では、1 8 F E H ~ 2 F F F H）で構成されている。この R O M 制御領域に対応する R O M 3 0 6 の記憶領域には、C P U 3 0 4 が実行する複数種類の命令それぞれに対応する命令データ（オペコード）や C P U 3 0 4 がそれぞれの命令を実行するために必要な補足データ（オペランド）によって構成される制御プログラム用のデータ（単に、制御プログラムデータと称す

10

20

30

40

50

る場合がある)が記憶され、非使用領域に対応するROM306の記憶領域には、特定値(本実施形態では、00H)が一律に記憶され、ROMデータ領域に対応するROM306の記憶領域には、上記の制御プログラムによって参照される参照データ(例えば、上述の各種抽選データ)が記憶されている。また、その他領域に対応するROM306の記憶領域には、制御プログラムを管理するための管理データ(例えば、制御プログラムデータの最終アドレスを示すデータ)などが記憶されている。なお、本実施形態では、ROM領域の各々のアドレスに対応するROM306の記憶領域には、1バイト(8ビット)のデータが記憶可能であり、上述の各データ(命令データ、補足データ、参照データ、管理データ)が1バイトを超えるバイト数(例えば、2バイト)のデータである場合には、ROM制御領域の連続する複数のアドレスに対応するROM306の記憶領域に1バイト毎に分割して記憶している。

10

#### 【0184】

また、制御プログラムは、複数のサブルーチンによって構成されており、後述するCALL命令やEXESUB(EXECUTE SUBROUTINE)命令などによってサブルーチンの先頭アドレスに移動して、当該サブルーチンを順次実行することが可能になっている。

#### 【0185】

ところで、遊技台の分野では、制御プログラムの検査効率化の観点から、制御プログラムデータを記憶する記憶領域を制限する必要があるため、本実施形態では、上述の通りROM制御領域を0000H~0BB0Hに制限している。また、不正改造の抑止の観点から、制御プログラムデータ間に非使用領域や非使用のデータを設けないようにROM制御領域の若いアドレスに詰めて制御用プログラムデータを記憶する必要があるため、開発段階で非使用となった制御プログラムデータの記憶領域を埋めるために別の制御プログラムデータを分割して割り当てるなどの作業が発生し、結果としてサブルーチンを呼び出す制御プログラムデータが多数必要となってくる。さらに、制御プログラムの開発効率を向上させるためや、制御プログラムのデバッグ作業等のメンテナンス作業を簡便にするためにもサブルーチンを多用しているため、サブルーチンを呼び出す制御プログラムデータが多数必要となってくる。

20

#### 【0186】

このような開発環境の中で遊技の興趣を高めることを可能にするためには、ROM制御領域に対応するROM306の記憶領域に記憶されている従来の制御プログラムデータ(例えば、上述のサブルーチンを呼び出す制御プログラムデータ)を圧縮し、ROM制御領域に対応するROM306の記憶領域に新しい制御プログラムデータを追加することが可能な記憶領域を確保することが必要となってくる。

30

#### 【0187】

特に、所定の周期(例えば、10ms)毎に所定の処理を実行するタイマ割込み処理を実行するための制御プログラムには、複数の遊技制御処理のうちランダムに発生する複数の外部情報(各センサの変化等)を検出する処理(例えば、入力ポート状態更新処理(ステップS205))や外部情報の検出結果に基づいた処理(例えば、デバイス監視処理(ステップS237))など多くのサブルーチンを呼び出すための制御プログラムデータが含まれているため、このタイマ割込み処理を実行するための制御プログラムに含まれるサブルーチンを呼び出す制御プログラムデータの圧縮が必要となってくる。

40

#### 【0188】

また、安定した遊技制御をおこなえるようにするためには、上述のタイマ割込み処理を重点的に安定化させなければならない。具体的には、コーディングミスを抑止すること、所定の周期(例えば、10ms)毎に所定の処理を実行させるために(リアルタイム性を確保するために)処理速度を向上させることなどが必須となるため、サブルーチンを呼び出す制御プログラムに関する工夫が必要となる(詳細は後述する)。

#### 【0189】

<主制御部の命令データ>

50

次に、主制御部 300 の命令データについて説明する。図 18 は、主制御部 300 の命令データの構成の一例を示した図である。また、図 19 (a) は、主制御部 300 の命令データの上位ビットと下位ビットを示した図であり、同図 (b) は、命令データテーブルの一例を示した図である。

#### 【0190】

主制御部 300 の命令データは、例えば、命令データが 1 バイト (8 ビット) のデータによって構成される場合には、最大で 256 パターン定義することが可能である。なお、定義された命令データは ROM 306 に記憶されているものではなく、CPU 304 にハード的に組み込まれているものである。また、1 バイトの命令データについて、各々を上位 4 ビットと下位 4 ビットによって分類分けすると、上記パターンと命令データの関係性は、図 19 (b) に示される通りとなる。例えば、「INC BC (BC レジスタを 1 つ加算する命令)」に使用される INC (命令データ) は、上位ビット 0H (0000B)、下位ビット 3H (0011B) に割り当てられ、「DEC B (B レジスタを 1 つ減算する命令)」に使用される DEC (命令データ) は、上位ビット 0H (0000B)、下位ビット 5H (0101B) に割り当てられている。

#### 【0191】

なお、1 バイト長で定義可能な 256 パターンの命令データを上回るパターン数の命令データを割り当てる方法として、例えば、上記 1 バイトで定義される 256 パターンのうち 1 パターンに対してさらに新たな 1 バイトで定義される 256 パターンを対応付けることで最大 511 パターン (255 パターン (1 バイトの命令データ) + 256 パターン (2 バイトの命令データ)) の命令データを定義する方法があるが、拡張した命令データは、2 バイトのデータによって構成されるため、この命令を使用すると、上述の 1 バイトの命令データを使用するよりも ROM 制御領域に対応する ROM 306 の記憶領域を圧迫してしまう。

#### 【0192】

そこで、本実施形態では、図 19 (b) の太線で囲った領域、すなわち、上位ビット 1H (0001B) と下位ビット 8H (1000B) ~ FH (1111B) の組合せと、上位ビット 3H (0011B) と下位ビット 8H (1000B) ~ FH (1111B) の組合せからなる領域を、後述する EXESUB 命令を割り当てるための特別な領域としている。このように、従来、命令データが割り当てられていない、または、割り当てられていた命令データが削除された空き領域を特別な領域として利用することで、限られたハードウェア資源 (例えば、1 バイトで定義可能なパターン (256 パターン)) を有効利用することができ、命令データによる ROM 制御領域に対応する ROM 306 の記憶領域の圧迫を抑制することができる。

#### 【0193】

また、命令データは、アセンブル後の機械語のチェックを簡便にするためにも役割ごとにグループ化 (一の上位ビットに対して連続する複数の下位ビットの組合せからなる領域、または、一の下位ビットに対して連続する複数の上位ビットの組合せからなる領域に、同系統の役割を持つ命令データを割り当てる) して命令データテーブルに割り当てている (例えば、上位ビット 8H (1000B) と下位ビット 0H (0000B) ~ 7H (0111B) の組合せで定義される部分における ADD 命令)。近年では、チップの開発負担を軽減するため、従来の命令データテーブルに割り当てている命令データの削減、または削減した命令データを割り当てていた部分への新たな命令データを追加するなどしてチップの開発を行う。このため、アセンブル後の機械語のチェックを簡便にしつつも、データ圧縮のために命令データを新たに追加する際には、追加する同系統の命令データをグループにして割り当てていく必要があり、本実施形態では、EXESUB 命令をグループにして特別な領域に割り当てることで解決を図っている。さらに、小分けされた特別な領域に対して新たな命令データを効率良く割り当てるためには、追加する命令データのグループを小分けのグループにして割り当てていく必要があり、本実施形態では、さらに、EXESUB 命令を小分けのグループにして小分けされた特別な領域に割り当てることで解決を

図っている。

【0194】

＜CALL命令＞

次に、上述の制御プログラムの一つであるCALL命令について説明する。図20は、CALL命令によるアドレスの移動の一例を概念的に示したものであり、図21は、CALL命令の命令データと補足データの一例を示したものである。

【0195】

例えば、実行中の制御プログラムのアドレスが0700Hで、移動先のサブルーチンの先頭アドレスが03FFHの場合には、図20の符号Xで示すように、両者の相対アドレス（移動元のアドレスと移動先のアドレスの差）は、0301H（=0700H-03FFH）であり、この相対アドレスは2バイトのデータで表現可能である。ここで、この制御プログラムをCALL命令によって表現すると、図21（a）に示すように、CALL命令を示す1バイトの命令データと移動先のアドレスを示す2バイトの補足データで合計3バイトの制御プログラムデータが必要となる。

【0196】

また、実行中の制御プログラムのアドレスが0700Hで、移動先のサブルーチンの先頭アドレスが075FHの場合には、図20の符号Yで示すように、両者の相対アドレスは、005FH（=075FH-03FFH）であり、この相対アドレスは1バイトのデータで表現可能である。この場合であっても、この制御プログラムをCALL命令によって表現すると、図21（b）に示すように、CALL命令を示す1バイトの命令データと移動先のアドレスを示す2バイトの補足データで合計3バイトの制御プログラムデータが必要である。

【0197】

すなわち、CALL命令を実行するための制御プログラムは、移動先のアドレス（2バイトで定義可能な記憶領域（0000H～FFFFH）内の任意のアドレス）に関わらず、3バイトの制御プログラムデータが必要となる。

【0198】

なお、CALL命令を実行する際にCPU304は、具体的に、PCレジスタ（プログラムカウンタ）に記憶されているデータ（2バイト）をスタックエリアに記憶させる処理を実行した後に、CALL命令を実行するための補足データ（移動先のサブルーチンの先頭アドレスを示すデータ（2バイト））をPCレジスタ（プログラムカウンタ）に記憶させる処理（サブルーチンの先頭アドレスに移動）を実行することで、サブルーチンを呼び出すことが可能となる。

【0199】

＜EXESUB命令＞

次に、上述の制御プログラムの一つであるEXESUB命令について説明する。図22（a）は、EXESUB命令の構成の一例を示した図であり、同図（b）は、図19（b）の空き領域に対してEXESUB命令を割り当てた命令データテーブルを示した図である。

【0200】

EXESUB命令は、上述のCALL命令同様にサブルーチンを呼び出すことが可能となる命令という点では同じであるが、命令データ／補足データの構造および制御プログラムデータに必要な記憶容量が異なる（詳細は後述する）。

【0201】

＜EXESUB命令／命令データ＞

EXESUB命令の命令データは、本実施形態では、最上位ビット7（MSB）から最下位ビット0（LSB）の順番で、ビット7の0（固定値）、ビット6の0（固定値）、ビット5のα（可変値）、ビット4の1（固定値）、ビット3の1（固定値）、ビット2のβ（可変値）、ビット1のγ（可変値）、ビット0のδ（可変値）の合計8ビット（1バイト）で構成されている。すなわち、EXESUB命令は上述の可変値を変えることに

よって定義されるパターン（2の4乗＝16パターン）の命令データとなる。なお、ビット5の $\alpha$ 、ビット2の $\beta$ 、ビット1の $\gamma$ およびビット0の $\delta$ の合計4ビットのデータである $m$ は、0H～FHの数値範囲を表現可能であり、呼び出すサブルーチンの先頭アドレスを識別可能な識別情報（バイナリ形式）の一部である。

#### 【0202】

また、EXESUB命令の命令データは、上述の通り、命令データテーブルのうちの同図（b）に示す太線で囲った領域、すなわち、上位ビット1H（0001B）と下位ビット8H（1000B）～FH（1111B）の組合せと、上位ビット3H（0011B）と下位ビット8H（0111B）～FH（1111B）の組合せからなる空き領域（従来、命令データが割り当てられていない、または、割り当てられていた命令データが削除された領域）に割り当てられる。

10

#### 【0203】

なお、上述の通り、EXESUB命令の命令データが、可変値のビットと可変値のビットの間に、固定値のビットが入るように構成されていることにより、EXESUB命令の命令データをグループ化しつつ、上述の空き領域に割り当てることが可能になる。よって、EXESUB命令データの機械語のチェックを簡便となり、コーディングミスを抑制することができる。

#### 【0204】

さらに、上述の通り、EXESUB命令の命令データが、1または複数の固定値のビット（例えば、上記のビット4、ビット3）を挟んで上位にある可変値のビット数（例えば、1（上記のビット5））と比較して当該固定値のビットを挟んで下位にある可変値のビット数（例えば、3（上記のビット2、ビット1、ビット0））が大きくなる（記憶容量が大きくなる）ように構成されていることにより、EXESUB命令の命令データをグループ化しつつも、きめ細かく上述の空き領域に割り当てることが可能になる。よって、限られたハードウェア資源（例えば、1バイトで定義可能なパターン（256パターン））を有効利用することができ、命令データの圧縮によるROM制御領域に対応するROM306の記憶領域に記憶されている従来の制御プログラムデータの圧縮が可能となる。なお、1または複数の固定値のビットを挟んで下位にある可変値のビット数と比較して当該固定値を挟んで上位にある可変値のビット数が大きくなる（記憶容量が大きくなる）ように構成した場合であっても、上述の効果を奏する。

20

30

#### 【0205】

なお、上述の可変値 $\alpha$ 、 $\beta$ 、 $\gamma$ および $\delta$ とEXESUB命令の命令データのビット7～0の対応関係は上述の対応関係に限らず、例えば、ビット4に可変値 $\alpha$ を対応させ、同様に、ビット2に可変値 $\beta$ 、ビット1に可変値 $\gamma$ 、ビット0に可変値 $\delta$ を対応させてもよく、上述した対応関係であればよい。

#### 【0206】

さらに、命令データを構成する可変値 $\alpha$ 、 $\beta$ 、 $\gamma$ および $\delta$ の順序は、任意の順序としてもよいが、上述の順序（上位ビットから順に $\alpha$ 、 $\beta$ 、 $\gamma$ 、 $\delta$ ）、すなわち、後述するアドレス $mn$ の $m$ を上位ビットから $\alpha$ 、 $\beta$ 、 $\gamma$ 、 $\delta$ の順に並べることで定義可能になる順序とすることで、EXESUB命令データの機械語のチェックを簡便となり、コーディングミスを抑制することができる。

40

#### 【0207】

<EXESUB命令／補足データ>

EXESUB命令の補足データは、本実施形態では、00H～FFHの数値範囲を示す8ビットのデータである $n$ で構成されている。

よって、EXESUB命令は、命令データに含まれる4ビットのデータである $m$ と、8ビットの補足データである $n$ を、この順番で並べて（ $m$ を上位、 $n$ を下位として）構成される合計12ビットのアドレス $mn$ によって定義可能なアドレス領域（0000H～0FFFH）に先頭アドレスを持つサブルーチンを呼び出すことができる命令となる。すなわち、 $n$ は呼び出すサブルーチンの先頭アドレスを識別可能な識別情報（バイナリ形式）から

50

、当該識別情報の一部（m）を除いた情報の少なくとも一部である。

#### 【0208】

＜EXESUB命令／サブルーチンの呼び出し＞

図23は、EXESUB命令によるアドレスの移動の一例を概念的に示した図である。本実施形態では、m（4ビットのデータ）は、同図（a）に示すように、ROM領域のアドレス0000H～2FFFHのうち、上位1バイトの下位4ビット（0H～FH）を示し、n（8ビットのデータ）は、同図（b）に示すように、上位アドレスmによって表現される100H単位の記憶領域（0m00H～0mFFFH：m=0H～FH）を1ブロックとしたときに、当該1ブロック内のアドレス（00H～FFFH）を示す値である。

#### 【0209】

なお、移動先のアドレスを設定可能なアドレス領域を、nまたはmのいずれのデータを上位アドレスのデータとして定義しても、後述する制御プログラムデータの削減の効果は得られるが、上述の通り、命令データに含まれるmを上位アドレスのデータとして定義することで、図23（a）に示すように、アドレス領域に対してEXESUB命令の命令データをグループ化して対応付けることができ、コーディングミスを抑制することができる。

#### 【0210】

また、EXESUB命令の命令データは、呼び出すサブルーチンの先頭アドレスの一部を構成するデータであるmを含んで構成されていることから、移動先のアドレスデータを間違ってしまった場合であっても、間違ってしまったデータが命令データに含まれていれば、制御プログラムの実行が停止されるため、コーディングミスを抑制することができる。

#### 【0211】

図24は、EXESUB命令によるサブルーチンの先頭アドレスの呼び出しの一例を示したものであり、図25は、EXESUB命令の命令データと補足データの一例を示した図である。

#### 【0212】

例えば、実行中の制御プログラムのアドレスが0700Hで、移動先のサブルーチンの先頭アドレスが03FFHの場合を挙げる。この制御プログラムをCALL命令で表現すると、上述の通り、図24の符号X'で示す両者の相対アドレス（移動元のアドレスと移動先のアドレスの差）は、0301H（=0700H-03FFH）であり、この相対アドレスは2バイトのデータで表現可能であり、1バイトの命令データと2バイトの補足データで合計3バイトの制御プログラムデータが必要である。

#### 【0213】

一方、この制御プログラムをEXESUB命令で表現すると、移動先のアドレスへの移動は概念的にX''で示すものとなり、この相対アドレスは、1バイトのデータで表現可能な00FFH（=03FFH-0300H）となる。すなわち、EXESUB命令は、図25（a）に示すように、00011011Bで表される1バイトの命令データと、11111111Bで表される1バイトの補足データで合計2バイトの制御プログラムデータが必要となる。

#### 【0214】

また、実行中の制御プログラムのアドレスが0700Hで、呼び出し先のサブルーチンの先頭アドレスが075FHの場合を挙げる。この場合は、この制御プログラムがCALL命令またはEXESUB命令のいずれで表現する場合であっても、上述の通り、図24の符号Y'で示すように、両者の相対アドレスは、1バイトで表現可能な005FH（=0C5FH-0C00H）であり、この相対アドレスは1バイトのデータで表現可能であるが、この制御プログラムをCALL命令で表現すると、1バイトの命令データと2バイトの補足データで合計3バイトの制御プログラムデータが必要となる。

#### 【0215】

一方、この制御プログラムをEXESUB命令で表現すると、図25（b）に示すよう



に、00011111Bで表される1バイトの命令データと、01011111Bで表される1バイトの補足データの合計2バイトの制御プログラムデータが必要となり、上述のCALL命令よりも、1命令当りのデータを1バイト削減することができる。

#### 【0216】

よって、EXESUB命令を実行するための制御プログラムは、移動先のアドレス（12ビットで定義可能な記憶領域（0000H～0FFFH）内の任意のアドレス）に関わらず、2バイトの制御プログラムデータが必要となり、上述のCALL命令よりも、1命令当りの制御プログラムデータを1バイト削減することができる。

#### 【0217】

すなわち、サブルーチンを呼び出す制御プログラムに上述のEXESUB命令を用いることで、従来のサブルーチンを呼び出す制御プログラム（例えば、CALL命令）を圧縮することができ、ROM制御領域に対応するROM306の記憶領域に記憶されている従来の制御プログラムデータを圧縮することが可能となる。

#### 【0218】

なお、EXESUB命令を実行する際にCPU304は、具体的に、PCレジスタ（プログラムカウンタ）に記憶されているデータ（2バイト）をスタックエリアに記憶させる処理を実行した後に、EXESUB命令を実行するための補足データ（移動先のサブルーチンの先頭アドレスを示すデータ（2バイト））をPCレジスタ（プログラムカウンタ）に記憶させる処理（サブルーチンの先頭アドレスに移動）を実行することで、サブルーチンを呼び出すことが可能となる。

#### 【0219】

また、サブルーチンを呼び出す制御プログラムに上述のEXESUB命令を用いることで、上述制御プログラムの圧縮に起因して従来（例えば、CALL命令）よりもサブルーチンを呼び出す制御プログラムの処理速度を向上させる（ステート数を少なくする）ことができる。

#### 【0220】

さらに、上述のように、ROM領域は、上述のEXESUB命令によって呼び出すことができる領域（本実施形態では、0000H～0FFFH）に対応するROM306の記憶領域に、全ての制御プログラムデータが記憶されるようにROM制御領域（本実施形態では、0000H～0BB1H）を設け、少なくとも、上述のEXESUB命令によって呼び出すことができる領域（本実施形態では、0000H～0FFFH）から、制御プログラムデータが記憶されているROM制御領域（本実施形態では、0000H～0BB1H）を除いた領域（本実施形態では、0BB1H～0FFFH）を、非使用領域となるように構成されるとともに、非使用領域の後にROMデータ領域が続くように構成されている。

#### 【0221】

このようにROM領域を構成した遊技台において、サブルーチンを呼び出す制御プログラムに上述のEXESUB命令を用いる（ROM制御領域に対応するROM306の記憶領域にEXESUB命令の制御プログラムデータを記憶する）ことで、CPU304が制御プログラムデータおよび非使用データを除くデータを命令データとして直接読み込むことがないため、補足データのコーディングミスによるCPU304の誤作動を防止できる場合がある。

#### 【0222】

<制御プログラムの一例>

図26は、上述の主制御部タイマ割込処理の制御プログラムの一例を示したプログラムリストである。なお、「\*\*」はEXESUB命令の補足データを示す。

#### 【0223】

CPU304は、タイマ割込み処理を開始すると、まずWDTをリスタートする処理を実行し、所定の処理（詳細は省略）を実行し、その後、EXESUB命令（1）～（18）を用いて、それぞれ、以下のサブルーチンを読み出す処理を実行する。入力ポート状態

更新処理、基本乱数初期値更新処理、基本乱数更新処理、演出乱数更新処理、タイマ更新処理、入賞口カウンタ更新処理、入賞受付処理、払出要求数送信処理、普図状態更新処理、普図関連抽選処理特図先読み制御処理、特図2状態更新処理、特図1状態更新処理、特図2関連抽選処理、特図1関連抽選処理、コマンド設定送信処理、外部出力信号設定処理、デバイス監視処理、を実行し、その他の処理（詳細は省略）を実行する。なお、詳細は省略するが、上述のEXESUB命令（1）～（18）によって呼び出されるサブルーチンの先頭アドレスはROM制御領域（本実施形態では、0000H～0BB0H）に含まれている。

#### 【0224】

このように、タイマ割込み処理におけるサブルーチンを呼び出す制御プログラムに上述のEXESUB命令を用いる（ROM制御領域のうちタイマ割り込み処理の領域に対応するROM306の記憶領域にEXESUB命令の制御プログラムデータを記憶する）ことで、タイマ割込み処理を安定化させることができる。

#### 【0225】

ところで、遊技台の開発では、一般的に従来の制御プログラムの一部を変更（削除や追加など）することで新たな遊技台を開発している。特に、遊技制御の安定化を図るためにできるだけ制御プログラムの変更は避けることが望まれる。

#### 【0226】

よって、ROM領域（例えば、タイマ割込み処理の領域）におけるサブルーチンを呼び出す制御プログラムすべてに、上述のEXESUB命令を用いる必要はなく、必要（例えば、上述の制御プログラムデータの圧縮の要求の有無）に応じて適宜用いればよい。さらに、上述のEXESUB命令は、従来の1バイトの命令データと2バイトの補足データで構成される3バイト命令の制御プログラムデータ（例えば、CALL命令）を、1バイトの命令データと1バイトの補足データで構成される2バイト命令の制御プログラムデータに圧縮した制御プログラムであるが、命令データと補足データで構成される制御プログラムを圧縮する制御プログラムであればよい。すなわち、EXESUB命令は、呼び出すサブルーチンの先頭アドレスを識別可能な識別情報（バイナリ形式）の一部を含んで構成される命令データと、当該識別情報から当該命令データに含まれた識別情報を除いた情報の少なくとも一部で構成される補足データとで構成される制御プログラムであればよく、さらに言えば、上述の命令データおよび補足データそれぞれが、CPU304が一回の処理で読み込むデータ長（例えば、8ビット）の整数倍であればよい。

#### 【0227】

＜ROM制御領域の変形例＞

図27は、ROM領域が、上述のEXESUB命令（EXESUBmn）によって呼び出すことができる領域（本実施形態では、0000H～0FFFH）よりも、制御プログラムデータが記憶されているROM制御領域（本実施形態では、0000H～11B8H）が広がるように構成されている場合を示した図である。また、図28（a）および（b）は、EXESUB命令（EXESUBmn）によって呼び出すことができるサブルーチンの先頭アドレスの領域を拡張した場合の一例を示した図である。

#### 【0228】

例えば、図27に示すように、ROM制御領域（本実施形態では、0000H～11B8H）が、EXESUB命令（EXESUBmn）で呼び出すことができるサブルーチンの先頭アドレスの領域（本実施形態では、0000H～0FFFH）より広い場合、ROM制御領域は、EXESUB対象領域（本実施形態では、0000H～0FFFH）と、EXESUB非対象領域（例えば、1000H～11B8H）で構成されることとなる。

#### 【0229】

この場合、EXESUB非対象領域に対応するアドレスを先頭アドレスに持つサブルーチンを呼び出す命令に上述のCALL命令を用いて補ったとしても、上述の通り、遊技の興趣を高めることを可能にしつつ、安定した遊技制御を実現することは可能である。

#### 【0230】

10

20

30

40

50

しかし、図 28 に示すように、EXESUB 命令 (EXESUBmn) によって呼び出すことができるサブルーチンの先頭アドレスの領域を拡張することで、上述の効果をより高めることが可能になる。

#### 【0231】

具体的な手段としては、図 28 (a) および (b) に示すように、後述する可変値  $\sigma$  を用いて、EXESUB 命令 (EXESUBmn) に要する記憶容量を維持しつつ、EXESUB 命令 (EXESUBmn) の命令データを拡張するものである。なお、以下は、同図に新たに追加した可変値  $\sigma$  について説明することとする。

#### 【0232】

可変値  $\sigma$  は、EXESUB 命令 (後述の EXESUBm'n) を拡張するために用いられ、図 28 (a) に示すように、EXESUB 命令 (後述の EXESUBm'n) の命令データ (本実施形態では、8 ビット) のビット 6 を構成する値であり、呼び出すサブルーチンの先頭アドレスを示すアドレス m'n のうち m' の値に応じて変化するものである。

#### 【0233】

また、命令データテーブルには、図 28 (b) に示すように、可変値  $\sigma$  によって拡張された EXESUB 命令 (EXESUBm'n) が、同図 (b) に示す太線で囲った領域、すなわち、上位ビット 5H (0101B) と下位ビット 8H (1000B) および 9H (1001B) の組合せからなる空き領域に割り当てられている。

#### 【0234】

ここで、m' は、図 22 (a) で示した m と同一の役割を持つデータであるが、上述の m がビット 5 の  $\alpha$ 、ビット 2 の  $\beta$ 、ビット 1 の  $\gamma$  およびビット 0 の  $\delta$  の合計 4 ビットのデータであったのに対して、m' はビット 6 の  $\sigma$ 、ビット 5 の  $\alpha$ 、ビット 2 の  $\beta$ 、ビット 1 の  $\gamma$  およびビット 0 の  $\delta$  の合計 5 ビットのデータである点が異なり、m' は、00H ~ 11H の数値範囲を表現可能なデータであり、呼び出すサブルーチンの先頭アドレスを識別可能な識別情報 (バイナリ形式) の一部である。

#### 【0235】

このように、上述の EXESUB 命令 (EXESUBmn) の命令データのビット 6 に可変値  $\sigma$  を設けた EXESUB 命令 (EXESUBm'n) を用いて、アドレス mn によって定義可能なアドレス領域 (000H ~ FFH) を拡張 (アドレス m'n によって定義可能なアドレス領域を 0000H ~ 11FFH に) することで、上述の EXESUB 非対象領域に対応するアドレスを先頭アドレスに持つサブルーチンを EXESUB 命令 (EXESUBm'n) を用いて呼び出すことができる。

#### 【0236】

なお、上述の可変値  $\sigma$ 、 $\alpha$ 、 $\beta$ 、 $\gamma$  および  $\delta$  と EXESUB 命令 (EXESUBm'n) の命令データのビット 7 ~ 0 の対応関係は上述の対応関係に限らず、例えば、ビット 5 に可変値  $\sigma$  を対応させ、同様に、ビット 4 に可変値  $\alpha$ 、ビット 2 に可変値  $\beta$ 、ビット 1 に可変地  $\gamma$ 、ビット 0 に可変値  $\delta$  を対応させてもよく、図 22 を用いて説明した対応関係と同じであればよい。

#### 【0237】

さらに、命令データを構成する可変値  $\sigma$ 、 $\alpha$ 、 $\beta$ 、 $\gamma$  および  $\delta$  の順序は、任意の順序としてもよいが、上述の順序 (上位ビットから順に  $\sigma$ 、 $\alpha$ 、 $\beta$ 、 $\gamma$ 、 $\delta$ )、すなわち、後述するアドレス m'n の m' を上位ビットから  $\sigma$ 、 $\alpha$ 、 $\beta$ 、 $\gamma$ 、 $\delta$  の順に並べることで定義可能になる順序とすることで、EXESUB 命令データの機械語のチェックを簡便となり、コーディングミスを抑制することができる。

#### 【0238】

このように、上述の ROM 制御領域 (例えば、0000H ~ 11B8H) に対応させて上述の EXESUB 命令 (EXESUBm'n) の命令データを設けることで、ROM 制御領域に対応する ROM 306 の記憶領域に記憶されている従来の制御プログラムデータの圧縮が可能となる。

#### 【0239】

10

20

30

40

50

ところで、図17および図27で示したように、制限されたROM制御領域の中でも開発された遊技台ごとに設けられたROM制御領域が異なる場合があるが、このような遊技台それぞれに対して、上述のEXESUBmnまたはEXESUBm'nそれぞれを命令データとして備えているCPUを使い分けて開発を行うことは、開発工数およびコストの観点からも無駄であるとともに、コーディングミスの原因ともなり、安定した遊技制御の実現を困難にしてしまうこととなる。よって、図17で示したROM制御領域を持つ遊技台であっても、EXESUB命令にEXESUBm'nを用いた方が望ましい。

#### 【0240】

この場合であっても、EXESUBm'nによって呼び出すことができる領域（本実施形態では、0000H~11FFH）よりも、制御プログラムデータが記憶されているROM制御領域（本実施形態では、0000H~0BB1H）が狭くなるように構成されているため、少なくとも、上述のEXESUBm'nによって呼び出すことができる領域から、制御プログラムデータが記憶されているROM制御領域を除いた領域（本実施形態では、0BB1H~11FFH）を、hituka用領域で構成するとともに、非使用領域の後にROMデータ領域が続くように構成すればよい。

#### 【0241】

＜特図状態更新処理、普図状態更新処理＞

次に、図29を用いて、上述の特図2状態更新処理（ステップS225）、特図1状態更新処理（ステップS227）、普図状態更新処理（ステップS221）の制御プログラムの一部について説明する。なお、同図（a）は特図2状態更新処理および特図1状態更新処理の制御プログラムの一部を抜き出したものであり、同図（b）は普図状態更新処理の制御プログラムの一部を抜き出したものである。

#### 【0242】

例えば、同図（a）に示す特図状態更新処理では、まず最初に、「LD L, Low.vbTokuzuStatus」の命令を実行することによって、同図（d）においてvbTokuzuStatusとして定義されたアドレス（この例では、RAM308に設けた特図ステータス用記憶領域を示すアドレスであるF060H）の下位バイト（この例では、60H）をLレジスタに記憶する。

#### 【0243】

続いて、「JP moReadMemory」の命令を実行することによって、同図（c）に示すmoReadMemory（メモリ読出し処理）に移動し、当該処理を実行する。このメモリ読出し処理では、まず最初に「LD H, F0H」の命令を実行することによってHレジスタにF0Hを記憶した後、「LD A, (HL)」の命令を実行することによってHLレジスタで示されるアドレス（この例では、F060H）の内容（この例では、特図ステータス）をAレジスタに記憶する。続いて、「AND A」の命令を実行した後に、読み出し元の特図状態更新処理に戻る。なお、同図（b）に示す普図状態更新処理も共通のサブルーチンであるmoReadMemory（メモリ読出し処理）を利用することによって、RAM308の普図ステータス用記憶領域に記憶された普図ステータスの読みだしを行う。このように、同図（c）に示す従来のメモリ読出し処理を利用した場合には、「LD H, F0H」という命令により、特図ステータス用記憶領域や普図ステータス用記憶領域のアドレスの上位（F0H）が解析され、不正行為を誘発するおそれがある。このため、本実施形態では、同図（c）に示す従来のメモリ読出し処理に替えて、同図（e）に示すメモリ読出し処理を採用する。

#### 【0244】

この本実施形態に係るメモリ読出し処理では、「LD A, T」の命令を実行することによって、Tレジスタの値をAレジスタに記憶するが、上述のように、Tレジスタには特定の値（F0H）が記憶されているため、AレジスタにはF0Hが記憶される。以降の処理は従来の処理と同様であるが、イミディエイト値を利用する代わりにTレジスタを利用しているため、特図ステータス用記憶領域や普図ステータス用記憶領域のアドレスの上位（F0H）が解析されるおそれが少なく、不正行為を未然に防止できる場合がある。また

、イミディエイト値を利用した場合には、例えば、F 0 HをF 1 Hと間違えて入力するなど、コーディングミスの可能性が高くなるが、Tレジスタを利用すればコーディングミスを未然に防止できる場合がある。

#### 【0245】

図30は、図29(c)に示す従来のメモリ読出し処理の機械語と、図29(e)に示す本実施形態に係るメモリ読出し処理の機械語と、を比較した図である。従来のメモリ読出し処理における1行目の「LD H, F 0 H」命令をアセンブルした場合、機械語は「26 F 0 H」となり「F 0 H」を含むため、従来のメモリ読出し処理では、特図ステータス用記憶領域や普図ステータス用記憶領域のアドレスの上位(F 0 H)が解析されてしまうおそれがある。一方、本実施形態のメモリ読出し処理における1行目の「LD A, T」命令をアセンブルした場合、機械語は「ED 0 1 H」となり「F 0 H」を含まないため、本実施形態のメモリ読出し処理では、特図ステータス用記憶領域や普図ステータス用記憶領域のアドレスの上位(F 0 H)が解析されるおそれが極めて低く、不正行為を未然に防止できる場合がある。

#### 【0246】

＜特図1関連抽選処理＞

次に、図31を用いて、上述の主制御部タイマ割込処理における特図1関連抽選処理について説明する。なお、同図は、特図1関連抽選処理の流れを示すフローチャートである。

#### 【0247】

ステップS1001では、特別図柄乱数移行処理を行う。この特別図柄乱数移行処理では、一次記憶した乱数値を、RAM308に設けた特別図柄関連抽選乱数の格納領域に移行(記憶)する。ステップS1002では、特別図柄当り抽選処理を行う。この特別図柄当り抽選処理では、上述の特図乱数値(特図1乱数値、特図2乱数値)と、ROM306に予め記憶している特別図柄抽選データを比較する抽選を行い、当該抽選結果に応じて、上述の15R大当りフラグ、2R大当りフラグ、第1小当りフラグ、第2小当りフラグ、第1はずれフラグ、第2はずれフラグ、特図確率変動フラグ、および普図確率変動フラグなどを設定し、特別図柄遊技状態を変動中に設定する。

#### 【0248】

ステップS1003では、表示図柄抽選処理を行う。この表示図柄抽選処理では、RAM308に記憶している表示図柄判定乱数と、ROM306に予め記憶している表示図柄抽選データを比較する抽選を行い、当該抽選結果に応じて、15R特別大当り図柄(特図A)、15R大当り図柄(特図B)、突然確変図柄(特図C)、突然時短図柄(特図D)、隠れ確変図柄(特図E)、突然通常図柄(特図F)、第1小当り図柄(特図G)、第2小当り図柄(特図H)、第1はずれ図柄(特図I)、および第2はずれ図柄(特図J)のいずれか一つの図柄を設定する。

#### 【0249】

ステップS1004では、特別図柄変動時間抽選処理を行う(詳細は後述する)。ステップS1005では、特別図柄変動時間設定処理を行う。この特別図柄変動時間設定処理では、上記ステップS1004の特別図柄変動時間抽選処理で取得した変動パターン(パターン番号)に応じたタイマを特別図柄変動タイマとして設定(記憶)する。ステップS1006では、特別図柄保留内情報移行処理を行う。この特別図柄保留内情報移行処理では、特別図柄保留内情報を変動順が先の格納領域に移行(記憶)した後に処理を終了する。

#### 【0250】

なお、ここでは一例として、上述の主制御部タイマ割込処理における特図1関連抽選処理(ステップS231)について説明したが、例えば、特図2関連抽選処理(ステップS229)や普図関連抽選処理(ステップS223)についても同様の処理を適用することができる。

#### 【0251】

10

20

30

40

50

### ＜主制御部のデータテーブルと変数記憶領域＞

次に、図 3 2 および図 3 3 を用いて、主制御部 3 0 0 の R O M 3 0 6 に記憶されるデータテーブルの一部と、R A M 3 0 8 に設けられる変数記憶領域の一部について説明する。

#### 【0 2 5 2】

##### ＜テーブル選択用テーブル＞

図 3 2 ( a ) は、テーブル選択用テーブルの一例を示した図であり、図 3 3 は、主制御部 3 0 0 の R O M 3 0 6 に記憶されるデータテーブルの定義の一部と、R A M 3 0 8 に設けられる変数格納領域の定義の一部を示したプログラムリストの一例である。

#### 【0 2 5 3】

図 3 2 ( a ) に示すテーブル選択用テーブルには、後述する第 1 変動パターン選択テーブルが記憶される R O M 3 0 6 のアドレスを示す飛先アドレスが、特図変動遊技の保留数毎に 0 個用～3 個用の 4 つに区別されて記憶されている。具体的には、R O M データ領域のアドレス 1 6 0 0 H ～ 1 6 0 1 H に対応する R O M 3 0 6 の 2 バイトの記憶領域には、保留数が 0 の場合に参照する第 1 変動パターン選択テーブルの飛先アドレスとして 1 7 4 F H が記憶されている ( 1 6 0 0      D E F B    1 7 4 F H ) 。また、R O M データ領域のアドレス 1 6 0 2 H ～ 1 6 0 3 H に対応する R O M 3 0 6 の 2 バイトの記憶領域には、保留数が 1 の場合に参照する第 1 変動パターン選択テーブルの飛先アドレスとして 1 7 4 F H が記憶されている ( 1 6 0 2      D E F B    1 7 4 F H ) 。また、R O M データ領域のアドレス 1 6 0 4 H ～ 1 6 0 5 H に対応する R O M 3 0 6 の 2 バイトの記憶領域には、保留数が 2 の場合に参照する第 1 変動パターン選択テーブルの飛先アドレスとして 1 7 4 F H が記憶されている ( 1 6 0 4      D E F B    1 7 4 F H ) 。また、R O M データ領域のアドレス 1 6 0 6 H ～ 1 6 0 7 H に対応する R O M 3 0 6 の 2 バイトの記憶領域には、保留数が 3 の場合に参照する第 1 変動パターン選択テーブルの飛先アドレスとして 1 7 5 5 H が記憶されている ( 1 6 0 6      D E F B    1 7 5 5 H ) 。

#### 【0 2 5 4】

##### ＜第 1 変動パターン選択テーブル＞

図 3 2 ( b ) は、第 1 変動パターン選択テーブルの一例を示した図である。この第 1 変動パターン選択テーブルには、当該テーブルのアドレスと、後述する第 2 変動パターン選択テーブルが記憶されたアドレスとの差 ( オフセット ) と、乱数の個数が記憶されている。具体的には、R O M データ領域のアドレス 1 7 5 0 H に対応する R O M 3 0 6 の 1 バイトの記憶領域には、オフセットとして A E H が記憶されているとともに、このアドレスに連続する 1 7 5 1 H に対応する R O M 3 0 6 の 1 バイトの記憶領域には、乱数の個数として 2 0 が記憶され ( 1 7 5 0 H      D E F B    A E H ,    2 0 ) 、以降の R O M データ領域のアドレス 1 7 5 2 H ～ 1 7 5 9 H に対応する R O M 3 0 6 の記憶領域には、図 3 2 ( b ) および図 3 3 に示すように、1 バイト長のオフセットと、1 バイト長の乱数の個数が交互に連続して記憶されている。なお、第 1 変動パターン選択テーブルにおいて乱数の個数は 1 0 進数で示している。

#### 【0 2 5 5】

##### ＜第 2 変動パターン選択テーブル＞

図 3 2 ( c ) は、第 2 変動パターン選択テーブルの一例を示した図である。この第 2 変動パターン選択テーブルには、パターン番号と、乱数の個数が記憶されている。具体的には、R O M データ領域のアドレス 1 8 0 0 H に対応する R O M 3 0 6 の 1 バイトの記憶領域には、パターン番号として 0 1 H が記憶されているとともに、このアドレスに連続する 1 8 0 1 H で示される 1 バイトの領域には、乱数の個数として 2 5 5 が記憶され ( 1 8 0 0 H      D E F B    0 1 H ,    2 5 5 ) 、以降の R O M データ領域のアドレス 1 8 0 2 H ～ 1 8 0 F H に対応する R O M 3 0 6 の記憶領域には、図 3 2 ( c ) および図 3 3 に示すように、1 バイト長のパターン番号と、1 バイト長の乱数の個数が交互に連続して記憶されている。なお、第 2 変動パターン選択テーブルにおいて乱数の個数は 1 0 進数で示している。

## 【0256】

## ＜変数記憶領域＞

図32(d)は、RWM領域のアドレスF040H～F042Hに対応するRAM308の記憶領域に設けられる保留数記憶領域、乱数1記憶領域、乱数2記憶領域を示した図である。RWM領域のアドレスF040Hに対応するRAM308の1バイトの記憶領域には保留数記憶領域が設けられ（F040 v b H o r y u u S P 1 D E F S 1）、この保留数記憶領域には保留数が記憶される。また、アドレスF041Hに対応するRAM308の1バイトの記憶領域には乱数1記憶領域が設けられ（F041 v b R n d H e n d o u 1 D E F S 1）、この乱数1記憶領域には乱数1が記憶される。また、アドレスF042Hに対応するRAM308の1バイトの記憶領域には乱数2記憶領域が設けられ（F042 v b R n d H e n d o u 2 D E F S 1）、この乱数2記憶領域には乱数2が記憶される。

10

## 【0257】

## ＜特別図柄変動時間抽選処理＞

次に、図34および図35を用いて、上述の特図1関連抽選処理における特別図柄変動時間抽選処理について説明する。なお、図34は、特別図柄変動時間抽選処理の流れを示すフローチャートであり、図35は、特別図柄変動時間抽選処理のプログラムリストの一例である。

## 【0258】

ステップS1101では、演出用乱数取得処理を行う。この演出用乱数取得処理では、上述の特図1関連抽選処理で取得した特図1乱数値（0～255）を、上述の乱数値1記憶領域（RWM領域のアドレスF041H）に記憶し、上述の特図2関連抽選処理で取得した特図2乱数値（0～255）を、上述の乱数値2記憶領域（RWM領域のアドレスF042H）に記憶する。

20

## 【0259】

次のステップS1102では、上述のテーブル選択用テーブル（図32(a)）の先頭アドレス（本実施形態では、1600H）を、HLレジスタに転送する（LD HL, tableSelectTable）。これにより、HLレジスタに1600Hが記憶される。

## 【0260】

次のステップS1103では、上述の保留数記憶領域（RWM領域のアドレスF040H）に記憶された保留数を、Aレジスタに転送する（LD A, (40H)）。ここで、「LD A, (n:イミディエイト値)」命令は、図36に示すように、Tレジスタを上位、イミディエイト値を下位で示すアドレスのデータをAレジスタにロードする命令である。本実施形態では、TレジスタにF0Hを記憶しているため、「LD A, (40H)」命令を実行した場合、Tレジスタ（F0H）を上位、イミディエイト値（40H）を下位で示すアドレス（F040H）のデータ、すなわち保留数記憶領域に記憶された保留数がAレジスタにロード（転送）される。例えば、保留数記憶領域に記憶された保留数が2の場合には、Aレジスタに02Hがロード（転送）される。

30

## 【0261】

次のステップS1104では、Aレジスタの値にAレジスタの値を加算する（ADD A, A）。先の例では、Aレジスタの値（02H）にAレジスタの値（02H）を加算することによってAレジスタに04Hが記憶される。次のステップS1105では、HLレジスタにAレジスタの値を加算する（ADDTWOONE HL, A）。詳細は後述するが、ADDTWOONE命令（ADDTWOONE HL, A）は、HLレジスタに記憶された2バイト長の値に、Aレジスタに記憶された1バイト長の値を加算するための命令である。先の例では、HLレジスタ（1600H）にAレジスタの値（04H）を加算することによってHLレジスタに1604Hが記憶される。

40

## 【0262】

次のステップS1106では、HLレジスタが示すアドレスの値をDEレジスタに転送

50

する (LDTWO DE, (HL))。ここで、「LDTWO DE, (HL)」命令は、図 36 に示すように、HL レジスタが示すアドレスのデータを DE レジスタにロードする命令である。先の例では、HL レジスタが示すアドレス (1604H) の値 (174FH) を DE レジスタに転送することによって DE レジスタには 174FH が記憶される。

【0263】

次のステップ S1107 では、DE レジスタの値と HL レジスタの値を交換する (EX DE, HL)。先の例では、DE レジスタの値 (174FH) と HL レジスタの値 (1604H) を交換することによって、HL レジスタには 174FH が記憶され、DE レジスタには 1604H が記憶される。

【0264】

次のステップ S1108 では、上述の乱数 1 記憶領域に記憶された乱数 1 (特別図柄変動時間判定乱数 1) を A レジスタに転送する (LDT A, (41H))。本実施形態では、T レジスタに F0H を記憶しているため、「LDT A, (41H)」命令を実行した場合、T レジスタ (F0H) を上位、イミディエイト値 (41H) を下位で示すアドレス (F041H) のデータ、すなわち乱数 1 記憶領域に記憶された乱数 1 が A レジスタにロード (転送) される。例えば、乱数 1 記憶領域に記憶された乱数 1 が 250 の場合には、A レジスタに 250 が記憶される。

【0265】

次のステップ S1109 では、HL レジスタの値で示されるアドレスの次の連続した 2 つのアドレスに格納された値を BC レジスタに転送する (INCTENSOU BC, (HL))。詳細は後述するが、「INCTENSOU BC, (HL)」命令は、HL レジスタに記憶された 2 バイト長の値に 1 を加算し、加算後の HL レジスタの値で示されるアドレスに記憶された 1 バイト長の値を、BC レジスタのうちの下位の C レジスタに記憶した後に、HL レジスタに記憶された 2 バイト長の値にさらに 1 を加算し、加算後の HL レジスタの値で示されるアドレスに記憶された 1 バイト長の値を、BC レジスタのうちの上位の B レジスタに記憶するための命令である。先の例では、HL レジスタに記憶された 2 バイト長の値 (174FH) に 1 を加算し、加算後の HL レジスタの値 (1750H) で示されるアドレスに記憶された 1 バイト長の値 (AEH) を、BC レジスタのうちの下位の C レジスタに記憶した後に、HL レジスタに記憶された 2 バイト長の値にさらに 1 を加算し、加算後の HL レジスタの値 (1751H) で示されるアドレスに記憶された 1 バイト長の値 (20) を、BC レジスタのうちの上位の B レジスタに記憶する。

【0266】

次のステップ S1110 では、A レジスタの値から B レジスタの値を減算する (SUB B)。先の例では、A レジスタの値 (250) から B レジスタの値 (20) を減算する。次のステップ S1111 では、ステップ S1110 の減算の結果、キャリーが発生したか否か、すなわち減算の結果が負の値になったかどうかを判定し、該当する場合にはステップ S1112 に進み、該当しない場合には該当するまでステップ S1109 ~ S1111 の処理を繰り返し実行する (JP NC, LOOP01)。先の例では、ステップ S1109 ~ S1111 の処理を 3 回繰り返すことによってキャリーが発生し、ステップ S1112 に進む際には、B レジスタの値が 8、C レジスタ値が B0H、HL レジスタの値が 1755H になる。

【0267】

次のステップ S1112 では、HL レジスタの値に C レジスタの値を加算する (ADDTWOONE)。先の例では、HL レジスタの値 (1755H) に C レジスタの値 (B0H) を加算することによって HL レジスタには 1805H が記憶される。次のステップ S1113 では、上述の乱数 2 記憶領域 (本実施形態では、F042H) に記憶された乱数 2 (特別図柄変動時間判定乱数 2) を A レジスタに転送する (LDT A, (42H))。本実施形態では、T レジスタに F0H を記憶しているため、「LDT A, (42H)」命令を実行した場合、T レジスタ (F0H) を上位、イミディエイト値 (42H) を下位で示すアドレス (F042H) のデータ、すなわち乱数 2 記憶領域に記憶された乱数 2

10

20

30

40

50



がAレジスタにロード（転送）される。例えば、乱数2が148の場合には、Aレジスタに148が記憶される。

#### 【0268】

次のステップS1114では、HLレジスタの値で示されるアドレスの次の連続した2つのアドレスに格納された値をBCレジスタに転送する（INCTENSOUBC,（HL））。先の例では、HLレジスタに記憶された2バイト長の値（1805H）に1を加算し、加算後のHLレジスタの値（1806H）で示されるアドレスに記憶された1バイト長の値（04H）を、BCレジスタのうちの下位のCレジスタに記憶した後に、HLレジスタに記憶された2バイト長の値にさらに1を加算し、加算後のHLレジスタの値（1807H）で示されるアドレスに記憶された1バイト長の値（128）を、BCレジスタのうちの上位のBレジスタに記憶する。

10

#### 【0269】

次のステップS1115では、Aレジスタの値からBレジスタの値を減算する（SUBB）。先の例では、Aレジスタの値（148）からBレジスタの値（128）を減算する。次のステップS1116では、ステップS1115の減算の結果、キャリーが発生したか否か、すなわち減算の結果が負の値になったかどうかを判定し、該当する場合にはステップS1117に進み、該当しない場合には該当するまでステップS1114～S1116の処理を繰り返し実行する（JPNC, LOOP02）。先の例では、ステップS1114～S1116の処理を2回繰り返すことによってキャリーが発生し、ステップS1117に進む際には、Bレジスタの値が32、Cレジスタの値が05H、HLレジスタの値が1809Hになる。ステップS1117では、その他の処理を行った後に処理を終了する。なお、ここでは一例として、特図1や特図2に関する処理について説明したが、例えば、普図の保留数や乱数値についても同様の処理を適用することができる。

20

#### 【0270】

##### <主制御部の特殊命令>

次に、図36および図37を用いて、主制御部300が備える特殊命令について説明する。なお、図36は、主制御部300が備える命令の一部と、その説明を示した図である。また、図37（a）は、主制御部300の命令データの上位ビットと下位ビットを示した図であり、同図（b）は、命令データテーブルの一例を示した図である。

#### 【0271】

30

##### <特殊命令／LD T命令>

特殊命令の一つである「LD T A、（n：イミディエイト値）」命令は、上述のとおり、Tレジスタを上位、イミディエイト値を下位で示すアドレスのデータをAレジスタにロードする命令であり、例えば、上述の特別図柄変動時間抽選処理のステップS1103では、「LD T A、（40H）」命令を実行することによって、Tレジスタ（F0H）を上位、イミディエイト値（40H）を下位で示すアドレス（F040H）のデータ、すなわち保留数記憶領域に記憶された保留数をAレジスタにロード（転送）している。

#### 【0272】

また、図示はしないが、主制御部300は、さらに、Aレジスタに記憶されたデータを、Tレジスタを上位、イミディエイト値を下位で示すアドレスにロードするための「LD T（n：イミディエイト値），A」命令を備えている。一方、Tレジスタを上位、汎用レジスタ（例えば、Hレジスタ）を下位で示すアドレスのデータをAレジスタにロードする命令（例えば、「LD T A，H」のような命令）や、Aレジスタに記憶されたデータを、Tレジスタを上位、汎用レジスタ（例えば、Hレジスタ）を下位で示すアドレスにロードするための命令（例えば、「LD T H，A」のような命令）は備えていない。

40

#### 【0273】

すなわち、LD T命令によって指定可能なアドレスの下位バイトは、イミディエイト値に限定されており、汎用レジスタを指定することは禁止されている（機械語にアセンブルされる過程でエラーとなりアセンブルすることができない）。このため、LD T命令では、プログラムリストの確認などを行う際にアドレスの確認が容易となる上に、Tレジスタ

50

の値（アドレスの上位バイト）を不用意に変更することができなくなる結果、コーディングミスや不具合の発生を減少させることができる場合がある。

#### 【0274】

##### <特殊命令／LD TWO 命令>

特殊命令の一つである「LD TWO OP1, (OP2)」命令は、上述のとおり、第2オペランドOP2で示される2つのレジスタが示すアドレスのデータを第1オペランドOP1で示される2つのレジスタにロードする命令であり、「LD TWO (OP2), OP1」命令は、第2オペランドOP2で示される2つのレジスタが示すアドレスに第1オペランドOP1で示される2つのレジスタのデータをロードする命令であり、例えば、上述の特別図柄変動時間抽選処理のステップS1106では、「LD TWO DE, (HL)」命令を実行することによって、HLレジスタが示すアドレス（1604H）の値（174FH）をDEレジスタにロード（転送）してDEレジスタには174FHを記憶している。OP2がHLレジスタではないBCレジスタまたはDEレジスタである場合、OP1がBCレジスタ、DEレジスタ、HLレジスタ、IXレジスタ、IYレジスタとなり、OP2がHLレジスタである場合のみ、OP1がBCレジスタ、DEレジスタ、HLレジスタ、ACレジスタ、AEレジスタ、BDレジスタ、IXレジスタ、IYレジスタとなる。LD TWO 命令は、AレジスタやHLレジスタを保護するのに適している。

#### 【0275】

また、LD TWO 命令では、必要なステート数は14ステート、必要なバイト数は2バイトである。一方、従来の命令によってLD TWO 命令と同様の処理を行う場合には、図36に示すように、必要なステート数は14ステート（＝4＋6＋4）、必要なバイト数は3バイト（＝1＋1＋1）である。したがって、HLレジスタが示すアドレスのデータをDEレジスタにロードする場合にLD TWO 命令を使用すれば、制御プログラムの処理時間を短縮して制御負担を軽減することができる上に、プログラムコード量を削減することができ、メモリ容量が制限されているROMの記憶領域を有効に利用することができる場合がある。また、遊技性を高めるためにプログラムコードが増えたとしても、プログラム容量の増加や処理速度の低下を抑止できる場合がある。また、プログラム容量を削減することによって、開発者のコーディングミスやチェックミスを減らし、デバッグが容易で開発効率を高めることが可能となることに加えて、遊技台として適正な機能を有するか否かを外部機関が確認する際にも、確認作業のミスを防止でき、遊技者が安心して遊技を行うことができる遊技台を提供できる場合がある。

#### 【0276】

##### <特殊命令／ADD TWO ONE 命令>

特殊命令の一つであるADD TWO ONE 命令（ADD TWO ONE OP1, OP2）は、第1オペランドOP1で示される2つのレジスタに記憶された2バイト長の値に、第2オペランドOP2で示されるレジスタに記憶された1バイト長の値（または、第2オペランドOP2で示される1バイト長のイミディエイト値）を加算するための命令である。ADD TWO ONE 命令の命令コードの上位1バイトは固定値（この例ではEDH）とし、下位バイトは、命令データテーブルの上位ビット4H（0100B）と下位ビット0H（0000B）～2H（0010B）の組合せと、命令データテーブルの上位ビット4H（0100B）と下位ビット4H（0100B）～7H（0001B）、9H（1001H）の組合せに割り当てている。このように、従来は空き領域とされていた命令の領域を利用することで、限られたハードウェア資源を有効利用することができる上に、空き領域に不正な命令（隠し命令）を埋め込むような不正行為を未然に防止することができ、遊技の公平性を担保できる場合がある。なお、本実施形態では、特殊命令の命令コードを2バイト長としているが、命令コードのバイト長は特に限定されず、例えば1バイト長でもよい。また、ADD TWO ONE 命令のうちの特定の命令（例えば、「ADD TWO ONE HL, A」命令）のみ、他のADD TWO ONE 命令よりも命令コードが短く（例えば、1バイト長）てもよい。また、「ADD TWO ONE OP1, A」命令のうちの特定の命令（例えば、「ADD TWO ONE HL, A」命令）のみ、他のADD TWO ONE

NE命令よりも命令コードが短く（例えば、1バイト長）てもよい。この場合、該命令の利便性を高めることができる。

#### 【0277】

「ADDTWOONE HL, C」命令を例に挙げると、第1オペランドで示されるペアレジスタHLに記憶された2バイト長の値に、第2オペランドで示されるCレジスタに記憶された1バイト長の値を加算する場合に使用される命令であり、例えば、上述の特別図柄変動時間抽選処理のステップS1112では、「ADDTWOONE HL, C」を実行することによって、HLレジスタに記憶された2バイト長の値（1755H）に、Cレジスタに記憶された1バイト長の値（B0H）を加算することによって、HLレジスタの値を1805Hに更新している。なお、ADDTWOONE命令（ADDTWOONE OP1, OP2）は、キャリーが発生し難いが、HLレジスタに記憶された2バイト長の値（例えば、FF01H）にCレジスタに記憶された1バイト長の値（例えば、FFH）を加算する場合には、キャリーが発生する。すなわち、ADDTWOONE命令（ADDTWOONE OP1, OP2）を実行後にキャリーが発生する場合には、HレジスタがFFHであると判断することができる。

#### 【0278】

また、「ADDTWOONE HL, C」命令では、必要なステート数は8ステート、必要なバイト数は2バイトである。一方、従来の命令によって「ADDTWOONE HL, C」命令と同様の処理を行う場合には、図36に示すように、必要なステート数は22ステート（＝4＋4＋10＋4）、必要なバイト数は6バイト（＝1＋1＋3＋1）である。したがって、HLレジスタに記憶された2バイト長の値にCレジスタに記憶された1バイト長の値を加算する場合に「ADDTWOONE HL, C」命令を使用すれば、制御プログラムの処理時間を短縮して制御負担を軽減することができる上に、プログラムコード量を削減することができ、メモリ容量が制限されているROMの記憶領域を有効に利用することができる場合がある。また、遊技性を高めるためにプログラムコードが増えたとしても、プログラム容量の増加や処理速度の低下を抑止できる場合がある。また、プログラム容量を削減することによって、開発者のコーディングミスやチェックミスを減らし、デバッグが容易で開発効率を高めることが可能となることに加えて、遊技台として適正な機能を有するか否かを外部機関が確認する際にも、確認作業のミスを防止でき、遊技者が安心して遊技を行うことができる遊技台を提供できる場合がある。

#### 【0279】

また、従来CPUに実行可能に備えられている8ビット演算命令では、「ADD A, E」命令のように、加算結果（「SUB A, E」命令の場合は減算結果）をAレジスタに入れるしかなかったが、ADDTWOONE命令を実行して「ADDTWOONE HL, E」命令とすることで「ADD L, E」を可能にしており、Aレジスタを使用せずに温存することができる（使用用途が広いAレジスタを他の命令で使うことが可能となる結果、プログラム容量を削減できる場合がある）。また、「LD H, 00H」命令を実行してHレジスタに00Hを転送した後に、「ADDTWOONE HL, E」命令を実行してLレジスタにEレジスタの値を加算して「SRL H」命令を実行してLレジスタが桁あふれしたかどうかをキャリーフラグが立ったか否かで確認することでAレジスタを使用せずに温存することができる（使用用途が広いAレジスタを他の命令で使うことが可能となる結果、プログラム容量を削減できる場合がある）。

#### 【0280】

すなわち、「ADD A, L」命令は実行可能であり、「ADD L, A」命令は実行不可能であるが、「ADDTWOONE HL, A」命令を実行可能にした命令セットを備えたことにより、命令セットにおける命令数を1つ増やすだけで、8ビット演算の加算結果を格納可能なレジスタを増やすことができるとともに、16ビットー8ビット演算も可能にしており、命令セットを1命令増やすだけで2つの機能を実現可能としている。

#### 【0281】

換言すると、加算命令「特定のレジスタペア←特定のレジスタペア＋特定のレジスタ」

を命令セットに含むとともに、特定のレジスタペアの下位側のレジスタを8ビット演算の  
アキュムレータとは異なるレジスタとした。

#### 【0282】

従来では、命令セットにおける命令数を増やしたくなかったため、16ビット演算と8  
ビット演算しか搭載せず、さらに16ビットのアキュムレータは複数のレジスタペアのう  
ちの特定レジスタペアのみとし、8ビットのアキュムレータは複数のレジスタのうちの特  
定レジスタのみとしていたが、上記構成により、少ない命令数の増加で複数の機能を実現  
することができる。

#### 【0283】

また、HLレジスタが0か否かを確認するために従来では、「LD A, H」命令でH  
レジスタの値をAレジスタに転送した後に「OR A, L」命令を実行後に0フラグが立  
つか否かでHLレジスタが0か否かを確認していた。すなわち、8ビット演算を実行して  
0か否かを確認しているため、8ビット演算のアキュムレータ（この場合はAレジスタ）  
を使用してしまっていた。しかし、「ADDTWOONE HL, n(00H)」命令後  
に0フラグが立つか否かを確認することによってAレジスタを使用しないで済む。すなわ  
ち、「ADDTWOONE HL, n」命令は、16ビットレジスタペアに8ビットの直  
値を加算する機能と、16ビットレジスタペアが0か否かを確認する機能と、Aレジスタ  
を温存する機能とを備えている。また、命令に直値を使用しているので、プログラムソー  
スの可読性を向上できる。

#### 【0284】

<特殊命令／INCTENSOU命令>

特殊命令の一つであるINCTENSOU命令（INCTENSOU OP1, OP2  
）は、第2オペランドOP2で示される2つのレジスタに記憶された2バイト長の値に1  
を加算し、加算後の2つのレジスタの値で示されるアドレスに記憶された1バイト長の値  
を、第1オペランドOP1で示される2つのレジスタのうちの下位のレジスタに記憶した  
後に、第2オペランドOP2で示される2つのレジスタに記憶された2バイト長の値にさ  
らに1を加算し、加算後の2つのレジスタの値で示されるアドレスに記憶された1バイト  
長の値を、第1オペランドOP1で示される2つのレジスタのうちの上位のレジスタに記  
憶するための命令である。INCTENSOU命令の命令コードの上位1バイトは固定値  
（この例ではEDH）とし、下位バイトは、命令データテーブルの上位ビット1H(00  
01B)と下位ビット8H(1000B)～9H(10001B)の組合せと、命令デー  
タテーブルの上位ビットDH(1101B)と下位ビット4H(0100B)～6H(0  
110B)の組合せに割り当てている。本例では、第1オペランドOP1にアキュムレー  
タを含む機械命令のコードを連番（例えば、「INCTENSOU AC, (HL)」命  
令のコードがEDD4H、「INCTENSOU AE, (HL)」のコードがEDD5  
H）にしている。また、第1オペランドOP1にアキュムレータを含む命令のうちの特定  
のINCTENSOU命令の機械命令のコード（例えば、「INCTENSOU AE,  
(HL)」命令のコードがEDD5H）と、第1オペランドOP1にアキュムレータを含  
まない命令のうちの特定のINCTENSOU命令の機械命令のコード（例えば、「IN  
CTENSOU BD, (HL)」命令のコードがEDD6H）を連番にしている。また  
、第1オペランドOP1にアキュムレータを含まずペアレジスタを含む命令のうちの複数  
のINCTENSOU命令の機械命令のコード（例えば、「INCTENSOU BC,  
(HL)」命令のコードがED18H、「INCTENSOU DE, (HL)」命令の  
コードがED19H）と、第1オペランドOP1にアキュムレータを含まない命令のう  
ちの特定のINCTENSOU命令の機械命令のコード（例えば、「INCTENSOU  
BD, (HL)」命令のコードがEDD6H）とを離れた番号にしている。このように、  
従来は空き領域とされていた命令の領域を利用することで、限られたハードウェア資源を  
有効利用することができる上に、空き領域に不正な命令（隠し命令）を埋め込むような不  
正行為を未然に防止することができ、遊技の公平性を担保できる場合がある。

#### 【0285】

「INCTENSOU BC, (HL)」命令を例に挙げると、第2オペランドで示されるペアレジスタHLに記憶された2バイト長の値に1を加算し、加算後のペアレジスタHLの値で示されるアドレス(HL)に記憶された1バイト長の値を、第1オペランドで示されるペアレジスタBCのうちの下のレジスタCに記憶した後に、第2オペランドで示されるペアレジスタHLに記憶された2バイト長の値にさらに1を加算し、加算後のペアレジスタHLの値で示されるアドレス(HL)に記憶された1バイト長の値を、第1オペランドで示されるペアレジスタBCのうちの上のレジスタBに記憶するための命令である。INCTENSOU命令は、HLレジスタの値で示されるアドレスの次以降のアドレスに格納されている値をレジスタに転送するのに適している。また、INCTENSOU命令は、該命令実行後にHLレジスタに記憶された値が示すアドレスに格納された値に基づいて該命令を繰り返すか否かの判断処理を行う場合、繰り返すと判断した場合には該命令でHLレジスタにプラス1することでHLレジスタに記憶された値が示すアドレスを次のアドレスにできるために該命令を使用するのに適している。

10

#### 【0286】

例えば、上述の特別図柄変動時間抽選処理のステップS1109では、「INCTENSOU BC, (HL)」を実行することによって、HLレジスタに記憶された2バイト長の値(174FH)に1を加算し、加算後のHLレジスタの値(1750H)で示されるアドレスに記憶された1バイト長の値(AEH)を、BCレジスタのうちの下のレジスタCに記憶した後に、HLレジスタに記憶された2バイト長の値にさらに1を加算し、加算後のHLレジスタの値(1751H)で示されるアドレスに記憶された1バイト長の値(20)を、BCレジスタのうちの上のレジスタBに記憶している。

20

#### 【0287】

また、「INCTENSOU BC, (HL)」命令では、必要なステート数は14ステート、必要なバイト数は2バイトである。一方、従来の命令によってINCTENSOU命令と同様の処理を行う場合には、図36に示すように、必要なステート数は26ステート(=6+7+6+7)、必要なバイト数は4バイト(=1+1+1+1)である。したがって、HLレジスタの値で示されるアドレスの次の連続した2つのアドレスに格納された値をBCレジスタに転送する場合に「INCTENSOU BC, (HL)」命令を使用すれば、制御プログラムの処理時間を短縮して制御負担を軽減することができる上に、プログラムコード量を削減することができ、メモリ容量が制限されているROMの記憶領域を有効に利用することができる場合がある。

30

#### 【0288】

また、遊技性を高めるためにプログラムコードが増えたとしても、プログラム容量の増加や処理速度の低下を抑止できる場合がある。特に、本実施形態では、ステップS1109～S1111のループ処理、またはステップS1114～S1116のループ処理においてINCTENSOU命令を繰り返し実行することになるため、処理速度の短縮の効果が大きい。また、プログラム容量を削減することによって、開発者のコーディングミスやチェックミスを減らし、デバッグが容易で開発効率を高めることが可能となることに加えて、遊技台として適正な機能を有するか否かを外部機関が確認する際にも、確認作業のミスを防止でき、遊技者が安心して遊技を行うことができる遊技台を提供できる場合がある。

40

#### 【0289】

<特殊命令/TENSOUINC命令>

特殊命令の一つであるTENSOUINC命令(TENSOUINC OP1, OP2)は、第2オペランドOP2で示される2つのレジスタの値で示されるアドレスに記憶された1バイト長の値を、第1オペランドOP1で示される2つのレジスタのうちの下のレジスタに記憶し、第2オペランドOP2で示される2つのレジスタに記憶された2バイト長の値に1を加算した後に、第2オペランドOP2で示される2つのレジスタの値で示されるアドレスに記憶された1バイト長の値を、第1オペランドOP1で示される2つのレジスタのうちの上のレジスタに記憶し、第2オペランドOP2で示される2つのレジスタに記憶された2バイト長の値にさらに1を加算するための命令である。TENSOU

50

INC命令の命令コードの上位1バイトは固定値（この例ではEDH）とし、下位バイトは、命令データテーブルの上位ビット1H（0001B）と下位ビット6H（0110B）～7H（0111B）の組合せと、命令データテーブルの上位ビットDH（1101B）と下位ビット0H（0000B）～2H（0010B）の組合せに割り当てている。本例では、第1オペランドOP1にアキュムレータを含む機械命令のコードを連番（例えば、「TENSOUINC AC, (HL)」命令のコードがEDD0H、「TENSOUINC AE, (HL)」のコードがEDD1H）にしている。また、第1オペランドOP1にアキュムレータを含む命令のうちの特定のTENSOUINC命令の機械命令のコード（例えば、「TENSOUINC AE, (HL)」命令のコードがEDD1H）と、第1オペランドOP1にアキュムレータを含まない命令のうちの特定のTENSOUINC命令の機械命令のコード（例えば、「TENSOUINC BD, (HL)」命令のコードがEDD2H）を連番にしている。また、第1オペランドOP1にアキュムレータを含まずペアレジスタを含む命令のうちの複数のTENSOUINC命令の機械命令のコード（例えば、「TENSOUINC BC, (HL)」命令のコードがED16H、「TENSOUINC DE, (HL)」命令のコードがED17H）と、第1オペランドOP1にアキュムレータを含まない命令のうちの特定のTENSOUINC命令の機械命令のコード（例えば、「TENSOUINC BD, (HL)」命令のコードがEDD2H）とを離れた番号にしている。また、TENSOUINC命令とINCTENSOU命令は、同一の第1オペランドOP1の命令はTENSOUINC命令のほうがINCTENSOU命令よりも番号が若い（例えば、「TENSOUINC BC, (HL)」命令のコードがED16H、「INCTENSOU BC, (HL)」命令のコードがED18H）。また、第1オペランドOP1にアキュムレータを含むTENSOUINC命令の機械命令のコード（例えば、「TENSOUINC AC, (HL)」命令のコードがEDD0H）と、第1オペランドOP1にアキュムレータを含むINCTENSOU命令の機械命令のコード（例えば、「INCTENSOU AC, (HL)」命令のコードがEDD4H）とは離れた番号、および／またはTENSOUINC命令とINCTENSOU命令以外の命令の機械命令のコード（例えば、コードEDD3H）の前後にしている。

#### 【0290】

「TENSOUINC BC, (HL)」命令を例に挙げると、第2オペランドで示されるペアレジスタHLの値で示されるアドレス（HL）に記憶された1バイト長の値を、第1オペランドで示されるペアレジスタBCのうちの下位のレジスタCに記憶し、第2オペランドで示されるペアレジスタHLに記憶された2バイト長の値に1を加算した後に、第2オペランドで示されるペアレジスタHLの値で示されるアドレス（HL）に記憶された1バイト長の値を、第1オペランドで示されるペアレジスタBCのうちの上位のレジスタBに記憶し、第2オペランドで示されるペアレジスタHLに記憶された2バイト長の値にさらに1を加算するための命令である。TENSOUINC命令は、HLレジスタの値で示されるアドレス以降のアドレスに格納されている値をレジスタに転送するのに適している。また、TENSOUINC命令は、該命令を実行後にBレジスタの値が0か否かで該命令を繰り返すか否かの判断処理を行う場合、繰り返すと判断した場合には判断前にHLレジスタにプラス1されているために該命令を使用するのに適している。また、繰り返し処理を行わない場合にもTENSOUINC命令後はHLレジスタの値をプラス1した状態で次の処理を行うことができるので該命令を使用するのに適している。

#### 【0291】

また、「TENSOUINC BC, (HL)」命令では、必要なステート数は14ステート、必要なバイト数は2バイトである。一方、従来の命令によってTENSOUINC命令と同様の処理を行う場合には、図36に示すように、必要なステート数は26ステート（＝7＋6＋7＋6）、必要なバイト数は4バイト（＝1＋1＋1＋1）である。したがって、HLレジスタの値で示されるアドレスと次のアドレスに格納された値をBCレジスタに転送する場合に「TENSOUINC BC, (HL)」命令を使用すれば、制御プログラムの処理時間を短縮して制御負担を軽減することができる上に、プログラムコ

ード量を削減することができ、メモリ容量が制限されているROMの記憶領域を有効に利用することができる場合がある。また、遊技性を高めるためにプログラムコードが増えたとしても、プログラム容量の増加や処理速度の低下を抑止できる場合がある。また、プログラム容量を削減することによって、開発者のコーディングミスやチェックミスを減らし、デバッグが容易で開発効率を高めることが可能となることに加えて、遊技台として適正な機能を有するか否かを外部機関が確認する際にも、確認作業のミスを防止でき、遊技者が安心して遊技を行うことができる遊技台を提供できる場合がある。なお、INC TEN SOU命令とTEN SOU INC命令は、CPUに両方実行可能に備えるようにしても良いし、何れか一方のみ実行可能に備えるようにしてもよい。

#### 【0292】

##### <特殊命令／CPT命令>

特殊命令の一つである「CPT A, (n:イミディエイト値)」命令は、Aレジスタの内容と、Tレジスタを上位、イミディエイト値を下位で示すアドレスのデータを比較する命令であり、例えば、TレジスタにF0Hが記憶された状態で「CPT A, (40H)」を実行した場合、Aレジスタの内容と、Tレジスタを上位、イミディエイト値を下位で示すアドレス(F040H)のデータが比較される。

##### <特殊命令／ADDONETWO命令>

特殊命令の一つであるADDONETWO命令(ADDONETWO OP1, OP2)は、第1オペランドOP1で示されるレジスタに記憶された1バイト長の値に、第2オペランドOP2で示される2つのレジスタに記憶された2バイト長の値(または、第2オペランドOP2で示される2バイト長のイミディエイト値)を加算するための命令である。ADDONETWO命令の命令コードの上位1バイトは固定値(この例ではEDH)とし、下位バイトは、命令データテーブルの上位ビット4H(0100B)と下位ビットAH(1010B)の組合せに割り当てている。このように、従来は空き領域とされていた命令の領域を利用することで、限られたハードウェア資源を有効利用することができる上に、空き領域に不正な命令(隠し命令)を埋め込むような不正行為を未然に防止することができ、遊技の公平性を担保できる場合がある。なお、ADDONETWO命令(ADDONETWO OP1, OP2)は、キャリーが発生し易く、Aレジスタに記憶された1バイト長の値(例えば、00H)にHLレジスタに記憶された2バイト長の値(例えば、0100H)を加算する場合には、キャリーが発生する。すなわち、Hレジスタが00H以外である場合にADDONETWO命令(ADDONETWO OP1, OP2)を実行することでキャリーを発生させることができる。なお、ADDTWOONE命令とADDONETWO命令は、CPUに両方備えるようにしても良いし、何れか一方のみ備えるようにしてもよい。

#### 【0293】

##### <特殊命令／SUBTWOONE命令>

特殊命令の一つであるSUBTWOONE命令(SUBTWOONE OP1, OP2)は、第1オペランドOP1で示される2つのレジスタに記憶された2バイト長の値から、第2オペランドOP2で示されるレジスタに記憶された1バイト長の値(または、第2オペランドOP2で示される1バイト長のイミディエイト値)を減算するための命令である。SUBTWOONE命令の命令コードの上位1バイトは固定値(この例ではEDH)とし、下位バイトは、命令データテーブルの上位ビット4H(0100B)と下位ビット8H(1000B)の組合せに割り当てている。このように、従来は空き領域とされていた命令の領域を利用することで、限られたハードウェア資源を有効利用することができる上に、空き領域に不正な命令(隠し命令)を埋め込むような不正行為を未然に防止することができ、遊技の公平性を担保できる場合がある。なお、SUBTWOONE命令(SUBTWOONE OP1, OP2)は、キャリーが発生し難いが、HLレジスタに記憶された2バイト長の値(例えば、0001H)からAレジスタに記憶された1バイト長の値(例えば、FFH)を減算する場合には、キャリーが発生する。すなわち、SUBTWO

10

20

30

40

50

ONE命令 (SUBTWOONE OP1, OP2) を実行後にキャリーが発生する場合には、Hレジスタが00Hであると判断することができる。

#### 【0294】

＜データテーブルの変形例＞

次に、図38を用いて、上述のデータテーブルの変形例について説明する。なお、同図は、テーブル選択用テーブル、第1変動パターン選択テーブル、および第2変動パターン選択テーブルの変形例を示した図であり、上記図32に相当する図である。

#### 【0295】

同図(b)に示す第1変動パターン選択テーブルでは、上記図32(b)に示す第1変動パターン選択テーブルに対して、乱数の個数の項目を乱数の上限の項目に変更している。また、同図(c)に示す第2変動パターン選択テーブルでは、上記図32(c)に示す第2変動パターン選択テーブルに対して、乱数の個数の値を変更している。

#### 【0296】

＜特別図柄変動時間抽選処理の変形例＞

次に、図39を用いて、上記図34に示す特別図柄変動時間抽選処理の変形例について説明する。なお、同図は、変形例に係る特別図柄変動時間抽選処理の流れを示すフローチャートである。

#### 【0297】

この変形例に係る特別図柄変動時間抽選処理では、上記図34に示す特別図柄変動時間抽選処理の「INCTENSOU BC, (HL)」命令を用いたステップS1109の処理やステップS1114の処理を、「INCTENSOU AC, (HL)」命令を用いたステップS1150の処理やステップS1152の処理に変更するとともに、ステップS1151、S1153において、上述のCPT命令を利用している。

#### 【0298】

具体的には、ステップS1150では、HLレジスタの値で示されるアドレスの次の連続した2つのアドレスに格納された値をACレジスタに転送する (INCTENSOU AC, (HL))。例えば、HLレジスタの値が174FHの場合には、HLレジスタの値で示されるアドレスの次の連続した2つのアドレス (1750H、1751H) に格納された値をACレジスタに転送することによって、Aレジスタに19を、CレジスタにAEHを記憶する。

#### 【0299】

次のステップS1151では、「CPT A, (41H)」を実行することによって、Aレジスタの内容と、Tレジスタ (F0H) を上位、イミディエイト値 (41H) を下位で示すアドレス (F041H) のデータ、すなわち、上述の乱数1記憶領域に記憶された乱数1と、を比較する (Aレジスタの値から乱数1を減算する)。次のステップS1154では、ステップS1151の比較 (減算) の結果、キャリーが発生しないか否か、すなわち減算の結果が正の値になったかどうかを判定し、該当する場合にはステップS1112に進み、該当しない場合には該当するまでステップS1150～S1151の処理を繰り返し実行する。

#### 【0300】

また、ステップS1152では、HLレジスタの値で示されるアドレスの次の連続した2つのアドレスに格納された値をACレジスタに転送する (INCTENSOU AC, (HL))。例えば、HLレジスタの値が17FFHの場合には、HLレジスタの値で示されるアドレスの次の連続した2つのアドレス (1800H、1801H) に格納された値をACレジスタに転送することによって、Aレジスタに255を、Cレジスタに01Hを記憶する。

#### 【0301】

次のステップS1153では、「CPT A, (42H)」を実行することによって、Aレジスタの内容と、Tレジスタ (F0H) を上位、イミディエイト値 (42H) を下位で示すアドレス (F042H) のデータ、すなわち、上述の乱数2記憶領域に記憶された

10

20

30

40

50



乱数 2 を比較する（A レジスタの値から乱数 2 を減算する）。次のステップ S 1 1 5 5 では、ステップ S 1 1 5 3 の比較（減算）の結果、キャリーが発生しないか否か、すなわち減算の結果が正の値になったかどうかを判定し、該当する場合にはステップ S 1 1 1 7 に進み、該当しない場合には該当するまでステップ S 1 1 5 2 ～ S 1 1 5 3 の処理を繰り返し実行する。

#### 【0302】

##### ＜初期設定処理＞

次に、図 4 0 および図 4 1 を用いて、上述の主制御部メイン処理の初期設定 2（ステップ S 1 0 7）で実行される初期設定処理について説明する。なお、図 4 0（a）は、ROM 3 0 6 に記憶される初期設定データテーブルの一例を示した図であり、同図（b）は、初期設定処理後の RAM 3 0 8 の記憶領域の一部を示した図であり、同図（c）は、初期設定処理の流れを示すフローチャートである。また、図 4 1 は、初期設定処理のプログラムリストの一例である。

10

#### 【0303】

ステップ S 1 2 0 1 では、上述の TENSOUINC 命令（「TENSOUINC BD, (HL)」）によって、HL レジスタの値で示されるアドレス（HL）に記憶された 1 バイト長の値を、BD レジスタのうちの下位の D レジスタに記憶し、HL レジスタに記憶された 2 バイト長の値に 1 を加算した後に、HL レジスタの値で示されるアドレス（HL）に記憶された 1 バイト長の値を、BC レジスタのうちの上位の B レジスタに記憶し、HL レジスタに記憶された 2 バイト長の値にさらに 1 を加算する。例えば、HL レジスタに、同図（a）に示す初期設定データテーブルの先頭アドレスである 1 0 0 D H を設定した場合には、D レジスタには F 0 H が、B レジスタには 3 が記憶される。

20

#### 【0304】

次のステップ S 1 2 0 2 では、上述の TENSOUINC 命令（「TENSOUINC AE, (HL)」）によって、HL レジスタの値で示されるアドレス（HL）に記憶された 1 バイト長の値を、AE レジスタのうちの下位の E レジスタに記憶し、HL レジスタに記憶された 2 バイト長の値に 1 を加算した後に、HL レジスタの値で示されるアドレス（HL）に記憶された 1 バイト長の値を、AE レジスタのうちの上位の A レジスタに記憶し、HL レジスタに記憶された 2 バイト長の値にさらに 1 を加算する。例えば、先の例では、E レジスタには、1 0 0 F H に記憶されている値である 1 7 H が、A レジスタには、1 0 1 0 に記憶されている値である 0 1 H が記憶され、HL レジスタの値は 1 0 1 1 H に更新される。

30

#### 【0305】

次のステップ S 1 2 0 3 では、A レジスタの値を DE レジスタの値で示されるアドレス（DE）に記憶する。先の例では、A レジスタの値である 0 1 H を、DE レジスタの値で示されるアドレスである RAM 3 0 8 の F 0 1 7 H に記憶する（同図（b）参照）。

#### 【0306】

次のステップ S 1 2 0 4 では、B レジスタの値から 1 を減算し、減算結果が 0 になるまでステップ S 1 2 0 2 ～ S 1 2 0 3 の処理を繰り返し実行し（ステップ S 1 2 0 2 の処理のアドレスまで相対ジャンプして処理を実行し）、減算結果が 0 になった場合にはステップ S 1 2 0 5 に進む。

40

#### 【0307】

先の例では、B レジスタが 2 の場合には、ステップ S 1 2 0 2 の処理により、E レジスタには、1 0 1 1 H に記憶されている値である 2 0 H が、A レジスタには、1 0 1 2 H に記憶されている値である 1 6 H が記憶され、HL レジスタの値は 1 0 1 3 H に更新され、ステップ S 1 2 0 3 の処理により、A レジスタの値である 1 6 H を、DE レジスタの値で示されるアドレスである RAM 3 0 8 の F 0 2 0 H に記憶する（同図（b）参照）。

#### 【0308】

また、B レジスタが 1 の場合には、ステップ S 1 2 0 2 の処理により、E レジスタには、1 0 1 3 H に記憶されている値である 3 A H が、A レジスタには、1 0 1 4 H に記憶さ

50

れている値である 06H が記憶され、HL レジスタの値は 1015H に更新され、ステップ S1203 の処理により、A レジスタの値である 06H を、DE レジスタの値で示されるアドレスである RAM308 の F03AH に記憶する（同図（b）参照）。なお、上記 F017H、F020H、F03AH に記憶する遊技設定（遊技制御に用いる各種情報）としては、例えば電源ステータス、払出制御コマンド加工種別、表示器表示ステータスなどが挙げられ、また、遊技設定の数は、複数でもよいし、1 つのみでもよい。また、初期設定処理（moDataSet）は、処理開始時の HL レジスタの値を異ならせることにより（HL レジスタに設定するアドレスを、初期設定データテーブルとは異なる他のテーブルの先頭アドレスなどに設定することにより）、複数の処理から呼び出し可能な（例えば、HL レジスタを引数として呼び出し可能な）汎用的なデータ設定処理モジュールとして扱ってもよい。この場合、プログラム容量の削減や処理速度の向上を図ることができる。

#### 【0309】

この初期設定処理では、ステップ S1201、S1202 で TENSOUINC 命令を使用しているため、ステート数を従来の 26 ステートから 14 ステートに削減することができ、処理時間を短縮することができる上に、バイト数を従来の 4 バイトから 2 バイトに削減することができ、プログラム容量を削減することができる。特に、この例では、ステップ S1202～S204 のループ処理において TENSOUINC 命令を繰り返し実行することになるため、処理速度の短縮の効果が高い。

#### 【0310】

また、TENSOUINC 命令の中でも、A レジスタ（アキュムレータ）と他のレジスタのペアをオペランドとする命令（本実施形態では、TENSOUINC AC, (HL)、TENSOUINC AE, (HL)）は、A レジスタの値を変更すると同時に、ペアレジスタ（本実施形態では、BC レジスタと DE レジスタ）の上位バイトのレジスタ（本実施形態では、B レジスタと D レジスタ）を固定しつつ、ペアレジスタの下位バイトのレジスタ（本実施形態では、C レジスタと E レジスタ）の値だけを変更することができるため、従来よりも高度なプログラミングが可能で、プログラム容量の削減や処理速度の向上を図ることができる。

#### 【0311】

また、TENSOUINC 命令の中でも、第 1 のペアレジスタ（本実施形態では、BC レジスタ）の上位バイトのレジスタ（本実施形態では、B レジスタ）と、第 2 のペアレジスタ（本実施形態では、DE レジスタ）の上位バイトのレジスタ（本実施形態では、D レジスタ）のペアをオペランドとする命令（本実施形態では、TENSOUINC BD, (HL)）は、第 1 のペアレジスタと第 2 のペアレジスタの上位バイトのレジスタの値のみを変更することができるため、従来よりも高度なプログラミングが可能で、プログラム容量の削減や処理速度の向上を図ることができる。

#### 【0312】

また、TENSOUINC 命令の中でも、オペランドに B レジスタを含む命令（本実施形態では、TENSOUINC BC, (HL)、TENSOUINC BD, (HL)）を利用し、かつ、上述の初期設定処理のステップ S1202～S1204 のループ処理のように、B レジスタによってループ処理の回数を判定するように構成すれば、ループ処理のプログラム容量の削減や処理速度の向上を図ることができる。

#### 【0313】

なお、CPU304 は、TENSOUINC 命令と INCTENSOU 命令両方を実行可能であり、遊技制御プログラムは第一の命令（TENSOUINC）によるテーブルサーチと第二の命令（INCTENSOU）によるテーブルサーチの両方を実行するようにしてもよい。

#### 【0314】

「テーブルサーチ」は一般的な用語どおりの意味としてもよく、例えば、一または複数の行と一または複数の列から構成されたテーブル（例えば、図 32、図 38 の第 1 変動パ

ターン選択テーブル・第2変動パターン選択テーブル、図40の初期設定データテーブル)から対応する行を探し出し、その行の一または複数の列の値を取り出す処理を示してもよいし、探し出した行のテーブル内の行の位置(例えば行番号)を導出する処理としてもよい。同様に対応する列を探し出し、その列の一または複数の行の値を取り出す処理を示すものとしてもよいし、探し出した列のテーブル内の列の位置(例えば列番号)を導出する処理としてもよい。

#### 【0315】

ここで対応する行を探し出す処理は、探している一または複数の値が、テーブル内の特定の行における一または複数の特定の列の値と一致または関係する場合に、当該特定の行に対応する行とするような処理としてもよい。対応する列を探し出す処理も同様としてもよい。

10

#### 【0316】

＜主制御部の第2特殊命令＞

次に、主制御部300が備える第2特殊命令について説明する。図42(a)は、主制御部300が備える第2特殊命令の一部を示した図である。

#### 【0317】

＜第2特殊命令／CPRT命令(演算+リターン命令)＞

第2特殊命令の一つである「CPRTZ r」命令(または、「CPRTZ (rr)」命令)は、オペランドrが示すレジスタ(または、オペランド(rr)が示すペアレジスタrrに格納されたアドレス)に記憶された値から0を減算し、Zフラグが1にセットされる状態の場合(減算結果が0の場合。以下、同じ)に、スタックポインタSPが示すスタック領域SPに記憶されたデータをプログラムカウンタPCの下位アドレスに、また、スタックポインタSPが示すスタック領域SP+1に記憶されたデータをプログラムカウンタPCの上位アドレスにそれぞれロード(復帰)した後にスタックポインタSPを2つ加算してからZフラグを1にセットし、Zフラグが0にクリアされる状態の場合(減算結果が0以外の場合。以下、同じ)に、後続の処理のアドレスをプログラムカウンタPCにセットしてからZフラグを0にクリアする命令である。なお、プログラムカウンタPCをセットするタイミングとZフラグを0にクリアするタイミングは、逆でもよいし、同時でもよい。

20

#### 【0318】

また、「CPRTNZ r」命令(または、「CPRTNZ (rr)」命令)は、オペランドrが示すレジスタ(または、オペランド(rr)が示すペアレジスタrrに格納されたアドレス)に記憶された値から0を減算し、Zフラグが0にクリアされる状態の場合に、スタックポインタSPが示すスタック領域SPに記憶されたデータをプログラムカウンタPCの下位アドレスに、また、スタックポインタSPが示すスタック領域SP+1に記憶されたデータをプログラムカウンタPCの上位アドレスにそれぞれロード(復帰)した後にスタックポインタSPを2つ加算してからZフラグを0にクリアし、Zフラグが1にセットされる状態の場合に、後続の処理のアドレスをプログラムカウンタPCにセットしてからZフラグを1にセットする命令である。なお、オペランドrが示すレジスタとしては、B、C、D、E、H、Lレジスタなどが挙げられ、オペランドrrが示すペアレジスタとしては、BC、DE、HLなどが挙げられる(次のCPJR命令も同様)。

30

40

#### 【0319】

＜第2特殊命令／CPJR命令(演算+ジャンプ命令)＞

第2特殊命令の一つである「CPJRZ r, e」命令(または、「CPJPZ (r, e)」命令)は、オペランドrが示すレジスタ(または、オペランド(rr)が示すペアレジスタrrに格納されたアドレス)に記憶された値から0を減算し、Zフラグが1にセットされる状態の場合にプログラムカウンタPC+e(eは+127~-128の数値)が示すアドレスにジャンプしてからZフラグを1にセットし、Zフラグが0にクリアされる状態の場合に、後続の処理のアドレスをプログラムカウンタPCにセットしてからZフラグを0にクリアする命令である。また、「CPJRNZ r, e」命令(または、

50

「CPJPNZ (rr)、e」命令)は、オペランドrが示すレジスタ(または、オペランドrrが示すペアレジスタ)に記憶された値から0を減算し、Zフラグが0にクリアされる状態の場合にプログラムカウンタPC+e(eは+127~-128の数値)が示すアドレスにジャンプしてからZフラグを0にクリアし、Zフラグが1にセットされる状態の場合に、後続の処理のアドレスをプログラムカウンタPCにセットしてからZフラグを1にセットする命令である。

#### 【0320】

また、「CPJRC r、n、e」命令(または、「CPJRC (rr)、n、e」命令)は、オペランドrが示すレジスタ(または、オペランド(rr)が示すペアレジスタrrに格納されたアドレス)に記憶された値からn(8ビット長のイミディエイト値)を減算し、Cフラグが1にセットされる状態の場合(減算結果が0の場合。以下、同じ)に、プログラムカウンタPC+e(eは+127~-128の数値)が示すアドレスにジャンプしてからCフラグを1にセットし、Cフラグが0にクリアされる状態の場合(減算結果が0以外の場合。以下、同じ)に、後続の処理のアドレスをプログラムカウンタPCにセットしてからCフラグを0にセットする命令である。また、「CPJRNC r、n、e」命令(または、「CPJRNC (rr)、n、e」命令)は、オペランドrが示すレジスタ(または、オペランド(rr)が示すペアレジスタrrに格納されたアドレス)に記憶された値からn(8ビット長のイミディエイト値)を減算し、Cフラグが0にクリアされる状態の場合に、プログラムカウンタPC+e(eは+127~-128の数値)が示すアドレスにジャンプしてからCフラグを0にクリアし、Cフラグが1にセットされる状態の場合に、後続の処理のアドレスをプログラムカウンタPCにセットしてからCフラグを1にセットする命令である。

#### 【0321】

図42(b)は、第2特殊命令を実行する前後のフラグレジスタの状態を示した図である。主制御部300は、Sフラグ、Zフラグ、SZフラグ、Hフラグ、P/Vフラグ、Nフラグ、Cフラグを有する8ビットのフラグレジスタを備えている。このフラグレジスタに含まれるSフラグ、Zフラグ、SZフラグ、Hフラグ、Cフラグは、第2特殊命令の実行や、その他の算術論理演算命令の実行によって1にセットされるか0にクリアされる。

#### 【0322】

<CPJR命令を用いた処理>

次に、CPJR命令を用いた処理について説明する。図43は、CPJR命令を用いた処理の一例を示した図である。同図に示す第一のモジュールでは、処理Xを実行した後に、CPJR命令を実行し、所定条件が成立した場合に処理Zにジャンプする処理を行い、所定条件が成立しなかった場合に後続の処理Yを実行する処理を行う。

#### 【0323】

ここで、「所定条件が成立した場合」とは、上述の「CPJRZ r、e」命令(または、「CPJRZ (rr)、e」命令)であれば、Zフラグが1にセットされる状態となった場合であり、上述の「CPJPNZ r、e」命令(または、「CPJRNZ (rr)、e」命令)であれば、Zフラグが0にクリアされる状態になった場合である。また、上述の「CPJRC r、n、e」命令(または、「CPJPC (rr)、n、e」命令)であれば、Cフラグが1にセットされる状態になった場合であり、「CPJRNC r、n、e」命令(または、「CPJPNC (rr)、n、e」命令)であれば、Cフラグが0にクリアされる状態になった場合である。

#### 【0324】

また、CPJR命令の実行前に行われる処理Xにおけるフラグレジスタの内容が第1の内容αであった場合、CPJR命令の実行により、後続の処理Y、処理Zにおけるフラグレジスタの内容は第2の内容βに変化する。しかしながら、本実施形態では、CPJR命令の実行中にフラグレジスタの内容を参照しないことに加えて、CPJR命令の実行後の処理Y、処理Zでも、CPJR命令によって変化したフラグレジスタの内容を一切参照す

ることなく処理を行うように構成している。

#### 【0325】

このように、C P J R命令の実行中にフラグレジスタの内容を参照しないことに加えて、C P J R命令の実行によりフラグレジスタの内容を変化させるが、C P J R命令の実行後に行う所定の処理においてはフラグレジスタの内容を参照することなく処理を行うように構成すれば、フラグレジスタの内容の変化とプログラムの実行の流れの関連性を無くすることができ、プログラムの解析を困難とし、不正なプログラムの改ざんを未然に防止できる場合がある。なお、この例では、C P J R命令の実行中にフラグレジスタの内容を参照しないように構成したが、C P J R命令の実行中にはフラグレジスタの内容を参照し、C P J R命令の実行後に行う所定の処理においてはC P J R命令の実行前に変化したフラグレジスタの内容を参照することなく処理を行うように構成しても、同様の効果を得ることができる場合がある（以下、同様）。

10

#### 【0326】

図44（a）は、従来の命令を用いたプログラムの一例を示した図であり、同図（b）は、C P J R命令を用いたプログラムの一例である。例えば、Aレジスタの値が0であるか0以外であるかを判定し、Aレジスタの値が0の場合に所定の処理にジャンプするプログラムを組む場合、同図（a）に示す従来のプログラムでは、（1-1）に示すAND命令（AND A）と、（1-2）に示すJR命令（JR Z、ITmrRnwCt1020）の2つの命令によって実現する必要があるが、同図（b）に示す本実施形態のプログラムでは、（2-1）に示す1つのC P J R命令（C P J R Z A、ITmrRnwCt1020）だけで同一の処理を実現している。

20

#### 【0327】

このように、所定条件が成立するか否かを判定し、所定条件が成立した場合に所定の処理にジャンプするプログラムを組む場合にC P J R命令を用いれば、従来よりもプログラムコードを減らすことができ、限られたメモリを有効に活用できる場合がある。

#### 【0328】

また、同図（b）に示す本実施形態のプログラムでは、（2-1）に示すC P J R命令（C P J R Z A、ITmrRnwCt1020）の実行後の処理Y、処理Zでは、C P J R命令によって変化したフラグレジスタの内容を一切参照することなく処理を行うように構成している。このように、C P J R命令の実行によりフラグレジスタの内容を変化させる一方で、C P J R命令の実行後に行う所定の処理においてはフラグレジスタの内容を参照することなく処理を行うように構成すれば、フラグレジスタの内容の変化とプログラムの実行の流れの関連性を無くすることができ、プログラムの解析を困難とし、不正なプログラムの解析を未然に防止できる場合がある。

30

#### 【0329】

なお、同図（b）に示す本実施形態のプログラムの処理Zに続く処理Vでは、C P J R命令の実行後に他の命令（この例では、「OR（HL）」命令）によって変化したフラグレジスタの内容（この例では、Zフラグの内容）に基づいた処理（この例では、JR命令）を行っている。このように、フラグレジスタの内容を参照しない処理Zとフラグレジスタの内容を参照する処理Vを連続させれば、フラグレジスタの内容の変化とプログラムの実行の流れの関連性を複雑にすることができ、プログラムの解析を困難とし、不正なプログラムの解析を未然に防止できる場合がある。

40

#### 【0330】

さらに、この例では、処理ZでLD命令（ロード命令）を実行しているが、LD命令は命令の実行後にZフラグを変化させずにSZフラグを変化させる命令の一つであるため、「LD A、（HL）」命令の実行後にはZフラグは変化しないが、SZフラグが変化する。一方、その後の処理Vでは「JR Z、ITmrRnwCtL040」を実行しているが、ここでは、処理Zの「LD A、（HL）」命令の実行によって変化していないZフラグを参照して分岐処理を行っている。

#### 【0331】

50

このように、S ZフラグとZフラグを含む複数のフラグの値を変化させる第1の命令（例えば、C P J R命令）を実行し、当該フラグを参照しない第2の命令（例えば、D E C A命令）を実行し、ZフラグとS Zフラグのうちいずれか一方のみが変化する第3の命令（例えば、L D命令）を実行し、変化しない方（変化した方）のフラグを参照する第4の命令（例えば、J R Z命令）を実行するように構成すれば、フラグレジスタの内容の変化とプログラムの実行の流れの関連性を複雑にすることができ、プログラムの解析を困難とし、不正なプログラムの解析を未然に防止できる場合がある。

#### 【0332】

図45は、C P J R命令を用いた他の処理の一例を示した図である。同図に示す第一のモジュールでは、処理Xを実行した後に、C P J R命令を実行し、所定条件が成立した場合に、第一のモジュールとは異なる第二のモジュールにジャンプして処理Zを行い、所定条件が成立しなかった場合に後続の処理Yを実行している。このように、C P J R命令を用いて所定条件が成立した場合に所定の処理にジャンプするプログラムを組む場合には、同一のモジュール内の処理だけではなく、他のモジュールの処理にジャンプさせてもよい。

10

#### 【0333】

図46は、上述の処理Vを含む、C P J R命令を用いた他の処理の一例を示した図である。同図に示す第二のモジュールでは、処理Zを実行した後に、算術論理演算命令を実行し、当該命令の実行によって変化したフラグレジスタの内容 $\gamma$ に基づいて分岐処理を行っている。このように、C P J R命令の実行によって変化したフラグレジスタに基づいた処理は行わず、C P J R命令以外の命令（この例では、算術論理演算命令）の実行によって変化したフラグレジスタに基づいた処理を行うように構成すれば、フラグレジスタの内容の変化とプログラムの実行の流れの関連性を複雑にすることができ、プログラムの解析を困難とし、不正なプログラムの解析を未然に防止できる場合がある。

20

#### 【0334】

<C P R T命令を用いた処理>

次に、C P R T命令を用いた処理について説明する。図47は、C P R T命令を用いた処理の一例を示した図である。同図に示す第一のモジュールでは、C A L L命令を実行して第二のモジュールを呼び出し、呼び出し先の第二のモジュールで処理Xを実行した後に、C P R T命令を実行し、所定条件が成立した場合に呼び出し元の第一のモジュールに戻って処理Zを行い、所定条件が成立しなかった場合に後続の処理Yを実行している。

30

#### 【0335】

ここで、「所定条件が成立した場合」とは、上述の「C P R T Z r」命令（または、「C P R T Z (r r)」命令）であれば、Zフラグが1にセットされる状態になった場合であり、上述の「C P R T N Z r」命令（または、「C P R T N Z (r r)」命令）であれば、Zフラグが0にクリアされる状態になった場合である。

#### 【0336】

また、C P R T命令の実行前に行われる処理Xにおけるフラグレジスタの内容が第1の内容 $\alpha$ であった場合、C P R T命令の実行により、後続の処理Y、処理Zにおけるフラグレジスタの内容は第2の内容 $\beta$ に変化する。しかしながら、C P R T命令の実行中にフラグレジスタを参照しないことに加えて、C P R T命令の実行後の処理Y、処理Zでも、C P R T命令によって変化したフラグレジスタの内容を一切参照することなく処理を行うように構成している。

40

#### 【0337】

このように、C P R T命令の実行中にフラグレジスタを参照しないことに加えて、C P R T命令の実行によりフラグレジスタの内容を変化させる一方で、C P R T命令の実行後に行う所定の処理においてはフラグレジスタの内容を参照することなく処理を行うように構成すれば、フラグレジスタの内容の変化とプログラムの実行の流れの関連性を無くすることができ、プログラムの解析を困難とし、不正なプログラムの解析を未然に防止できる場合がある。

50

## 【0338】

図48(a)は、C P R T命令を用いたプログラムの一例を示した図であり、同図(b)は、従来の命令を用いたプログラムの一例である。例えば、Aレジスタの値が0であるか0以外であるかを判定し、Aレジスタの値が0の場合に所定の処理にリターン(復帰)するプログラムを組む場合、同図(b)に示す従来のプログラムでは、(1-1)に示すAND命令(AND A)と(1-2)に示すRET命令(RET Z)の2つの命令によって実現する必要があるが、同図(a)に示す本実施形態のプログラムでは、(2-1)に示す1つのC P R T命令(C P R T Z A)だけで同一の処理を実現している。

## 【0339】

このように、所定条件が成立するか否かを判定し、所定条件が成立した場合に所定の処理にリターン(復帰)するプログラムを組む場合にC P R T命令を用いれば、従来よりもプログラムコードを減らすことができ、限られたメモリを有効に活用できる場合がある。

## 【0340】

また、同図(a)に示す本実施形態のプログラムでは、(2-1)に示すC P R T命令(C P R T Z A)の実行後の処理Y、処理Zでは、C P R T命令によって変化したフラグレジスタの内容を一切参照することなく処理を行うように構成している。このように、C P R T命令の実行によりフラグレジスタの内容を変化させる一方で、C P R T命令の実行後に行う所定の処理においてはフラグレジスタの内容を参照することなく処理を行うように構成すれば、フラグレジスタの内容の変化とプログラムの実行の流れの関連性を無くすことができ、プログラムの解析を困難とし、不正なプログラムの解析を未然に防止できる場合がある。

## 【0341】

なお、同図(a)に示す本実施形態のプログラムの処理Zに続く処理Vでは、C P R T命令の実行後に他の命令によって変化したフラグレジスタの内容に基づいた処理を行っている。このように、フラグレジスタの内容を参照しない処理Zとフラグレジスタの内容を参照する処理Vを連続させれば、フラグレジスタの内容の変化とプログラムの実行の流れの関連性を複雑にすることができ、プログラムの解析を困難とし、不正なプログラムの解析を未然に防止できる場合がある。

## 【0342】

<主制御部の第3特殊命令>

次に、主制御部300が備える第3特殊命令について説明する。図49は、主制御部300が備える第3特殊命令の一部を示した図である。

## 【0343】

<第3特殊命令／RES命令>

第3特殊命令の一つである「RESmZ n、r」命令(mは0~7の数値)は、第2オペランドrで示されるレジスタに記憶されている値のうち、mで示されるビットのアクセスを禁止するとともに、第1オペランドで示されるビットn(nは、mを除く0~7の数値)を0にリセットする命令である。また、「RESmZ n、(rr)」命令(mは0~7の数値)は、第2オペランド(rr)で示されるペアレジスタrrに格納されたアドレスに記憶されているアドレスに格納された値のうち、mで示されるビットのアクセスを禁止するとともに、第1オペランドで示されるnビット(nは、mを除く0~7の数値)を0にリセットする命令である。

## 【0344】

例えば、図50(a)に示すように、「RES2Z 2、L」命令は、第2オペランドLで示されるLレジスタに記憶されている値のうち、mで示されるビット(この例では、ビット2)のアクセスを禁止するとともに、第1オペランドで示されるビット2を0にリセットする命令であり、この命令ではLレジスタのビット2は0にリセットされない。一方、同図(b)に示すように、「RES2Z 4、L」命令は、第2オペランドLで示されるLレジスタに記憶されている値のうち、mで示されるビット(この例では、ビット2)のアクセスを禁止するとともに、第1オペランドで示されるビット4を0にリセットす

10

20

30

40

50

る命令であり、この命令では、Lレジスタのビット2のアクセスは禁止されるがビット4は0にリセットされる。

#### 【0345】

また、図51(a)に示すように、「RES2Z 2、(HL)」命令は、第2オペランド(HL)で示されるペアレジスタHLに記憶されているアドレス(この例では、F010H)に格納された値のうち、mで示されるビット(この例では、ビット2)のアクセスを禁止するとともに、第1オペランドで示されるビット2を0にリセットする命令であり、この命令では、F010Hに格納されている値のビット2は0にリセットされない。一方、同図(b)に示すように、「RES2Z 4、(HL)」命令は、第2オペランド(HL)で示されるペアレジスタHLに記憶されているアドレス(この例では、F010H)に格納された値のうち、mで示されるビット(この例では、ビット2)のアクセスを禁止するとともに、第1オペランドで示されるビット4を0にリセットする命令であり、この命令では、F010Hに格納されている値のビット2のアクセスは禁止されるがビット4が0にリセットされる。

#### 【0346】

このようなRES命令によれば、所定のビットのアクセスを禁止しつつ、同時に他のビットを0にリセットすることができるため、所定のビットの数値を保護することができ、所定のビットが予期しない数値に変化して不具合を起こすような事態を未然に防止できる場合がある。

#### 【0347】

図52は、主制御部300の命令マップのうち、RES命令に該当する部分を抜き出して示した図である。上述のとおり、「RES2Z 2、L」命令と「RES2Z 2、(HL)」命令は、ビット2のアクセスが禁止されビット2がリセットできない命令であるため(実質的に意味の無い命令であるため)、同図の太線の四角で示すように、「RES2Z 2、L」命令に該当する25Hの部分と、「RES2Z 2、(HL)」命令に該当する26Hの部分は空き領域とされている。このように、命令マップの一部に空き領域を設けることによって、命令マップを使ったプログラム(機械語)の解析を困難にすることができる場合がある。

#### 【0348】

##### <RES命令の変形例1>

図53は、変形例1に係るRES命令を示した図である。RES命令の変形である「RESmZ n、r」命令(mは1~254の10進数の数値)は、第2オペランドrで示されるレジスタに記憶されている値のうち、10進数のmを1バイトの2進数に変換したときに1がセットされるビットのアクセスを禁止するとともに、第1オペランドで示されるビットn(nは、mを1バイトの2進数に変換したときに1がセットされないビット)を0にリセットする命令である。また、「RESmZ n、(rr)」命令(mは1~254の10進数の数値)は、第2オペランドrrで示されるペアレジスタに記憶されているアドレスに格納された値のうち、10進数のmを1バイトの2進数に変換したときに1がセットされるビットのアクセスを禁止するとともに、第1オペランドで示されるnビット(nは、mを2進数に変換したときに1がセットされないビット番号)を0にリセットする命令である。

#### 【0349】

例えば、図54(a)に示すように、「RES2Z 4、L」命令は、第2オペランドLで示されるレジスタLに記憶されている値のうち、10進数の2を1バイトの2進数に変換したときに1がセットされるビット(この例では、ビット1)のアクセスを禁止するとともに、第1オペランドで示されるビット4を0にリセットする命令である。また、同図(b)に示すように、「RES3Z 4、L」命令は、第2オペランドLで示されるレジスタLに記憶されている値のうち、10進数の3を1バイトの2進数に変換したときに1がセットされるビット(この例では、ビット1、ビット0)のアクセスを禁止するとともに、第1オペランドで示されるビット4を0にリセットする命令である。



## 【0350】

図55は、変形例1に係るRES命令を用いたプログラムの一例を示した図である。上述のとおり、「RESmZ n、(rr)」命令(mは1～254の10進数の数値)は、第2オペランド(rr)で示されるペアレジスタrrに記憶されているアドレスに格納された値のうち、10進数のmを1バイトの2進数に変換したときに1がセットされるビットのアクセスを禁止するとともに、第1オペランドで示されるnビット(nは、mを2進数に変換したときに1がセットされないビット番号)を0にリセットする命令である。

## 【0351】

したがって、同図の(3-1)に示す「RES251Z 2(=ebCS20pnSGt)、(HL)」命令は、第2オペランド(HL)で示されるペアレジスタHLに記憶されているアドレス(この例では、F096H(=v0ptCS2))に格納された値のうち、10進数の251を1バイトの2進数に変換したときに1がセットされるビット(この例では、ビット0～1、3～7)のアクセスを禁止するとともに、第1オペランドで示されるビット(この例では、ビット2)を0にリセットする命令である。

## 【0352】

このようなRES命令によれば、所定のビットのアクセスを禁止しつつ、同時に他のビットを0にリセットすることができるため、所定のビットの数値を保護することができ、従来のビット操作命令(例えば、CLR命令、SET命令、RES命令、BIT命令など)のように、コーディングミスなどによって所定のビットが予期しない数値に変更されてしまうような事態を未然に防止でき、デバックやテストの作業を短縮化でき、また、安定した遊技制御を行うことができる場合がある。また、アクセスを禁止するビットを10進数の数値で指定することができるため、プログラムの解析を困難にすることができる場合がある。

## 【0353】

<RES命令の変形例2>

図56は、変形例2に係るRES命令を示した図である。RES命令の変形である「RESmZ n、r」命令(mは0～7の数値)は、第2オペランドrで示されるレジスタを含むペアレジスタに記憶されている値のうち、mで示されるビットのアクセスを禁止するとともに、第1オペランドで示されるビットn(nは、mを除く0～8の数値)を0にリセットする命令である。また、「RESmZ n、(rr)」命令(mは0～7の数値)は、第2オペランド(rr)で示されるペアレジスタrrに記憶されているアドレスと、このアドレスに連続するアドレスに格納された2バイト長の値のうち、mで示されるビットのアクセスを禁止するとともに、第1オペランドで示されるビットn(nは、mを除く0～8の数値)を0にリセットする命令である。

## 【0354】

例えば、図57に示すように、「RES2Z 8、L」命令は、第2オペランドLで示されるレジスタLを含むペアレジスタHLに記憶されている値のうち、2で示されるビット(ビット2)のアクセスを禁止するとともに、第1オペランドで示されるビット8(この例では、HLレジスタの上位バイトであるHレジスタのビット0)を0にリセットする命令である。

## 【0355】

また、図58に示すように、「RES2Z 8、(HL)」命令は、第2オペランド(HL)で示されるペアレジスタHLに記憶されているアドレス(この例では、F010H)と、このアドレスに連続するアドレス(この例では、F011H)に格納された2バイト長の値のうち、2で示されるビット(ビット2)のアクセスを禁止するとともに、第1オペランドで示されるビット8(この例では、F010Hに連続するF011Hに格納された値のビット0値)を0にリセットする命令である。

## 【0356】

図59は、主制御部300の命令マップのうち、変形例2に係るRES命令に該当する部分を抜き出して示した図である。上述のとおり、「RES2Z 2、L」命令と「RE

10

20

30

40

50

S 2 Z 2、(H L)」命令は、ビット2のアクセスが禁止されビット2がリセットできない命令であるため、同図に太線の四角で示すように、これらの命令に替えて、「R E S 2 Z 8、L」命令と「R E S 2 Z 8、(H L)」命令を割り当てている。このように、命令マップの一部の並び順を不規則にすることによって、プログラム（機械語）の解析を困難にすることができる場合がある。

#### 【0357】

＜R E S命令の他の変形例＞

図60は、R E S命令の他の変形例を示した図である。例えば、同図(a)に示すR E S命令では、上述の変形例1の「R E S m Z n、r」命令または「R E S m Z n、(r r)」命令(mは1～254の10進数の数値)において、数値mに0が指定できるように変更し、0を指定した場合には全てのビットのアクセスを許可するように構成している。また、同図(b)に示すR E S命令では、上述の変形例1の「R E S m Z n、r」命令または「R E S m Z n、(r r)」命令(mは1～254の10進数の数値)において、数値mの指定が省略できるように変更し、指定を省略した場合には全てのビットのアクセスを許可するように構成している。

#### 【0358】

図61(a)は、所定ビットのアクセスを禁止しないR E S命令を備えた場合の命令マップと、所定ビットのアクセスを禁止するR E S命令を備えた場合の命令マップを示した図である。このように、所定ビットのアクセスを禁止しないR E S命令を備えた場合と、所定ビットのアクセスを禁止するR E S命令を備えた場合で命令マップの一部の並び順を変更することによって、プログラム（機械語）の解析を困難にすることができる場合がある。

#### 【0359】

また、同図(b)は、R E S命令の第1オペランドの数値の大小と、R E S命令の命令マップの並び順を異ならせた例を示した図である。このように、R E S命令の第1オペランドの数値が8→1→2→3→4→5→6→7と不規則に並ぶようにR E S命令を命令マップに割り当てることによって、プログラム（機械語）の解析を困難にすることができる場合がある。

#### 【0360】

＜ユーザプログラムの具体例＞

次に、上述のユーザプログラムの具体例について説明する。上述のとおり、主制御部300のCPU304がユーザモードに移行すると、ROM領域の第一領域の先頭アドレスである0000H以降に記憶された命令が順次、実行されてユーザプログラムが開始される。

#### 【0361】

図62(a)は、ユーザプログラムの一部であるプログラム起動設定処理の一例を示した図であり、同図(b)は、ユーザプログラムの一部である電源投入時設定処理の一例を示した図である。同図に示すユーザプログラムでは、最初にプログラム起動設定処理(アドレス0000H～0005Hに記憶された命令)が実行されることによって、スタックポインタや割込みモードの設定が行われ、その後に電源投入時設定処理(アドレス0062H)にジャンプするように構成されている。電源投入時処理では、アドレス0062H以降に記憶された命令が順次、実行されることによって、W D Tへの動作許可及び初期値の設定や、電源電圧の監視処理が行われる。

#### 【0362】

図63は、上述の主制御部タイマ割込処理に相当するユーザプログラムの一部を示した図である。主制御部300の基本回路302は、カウンタタイマ312からタイマ割込信号が入力された場合に、同図に示すような割込みベクトルテーブルを参照し、割込みベクトルテーブルに定義されたアドレスにジャンプするように構成されており、この例では、タイマ割込みに対応して定義されたアドレス040AHにジャンプするように構成されている。主制御部タイマ割込処理では、アドレス040AH以降に記憶された命令が順次、

実行されることによって、レジスタの退避や、同図（１）～（１８）に示す各種処理が、この順番で行われる。

#### 【０３６３】

＜主制御部の第４特殊命令＞

次に、主制御部３００が備える第４特殊命令について説明する。図６４は、主制御部３００が備える第４特殊命令の内容と、主制御部３００のＲＯＭ領域を示した図である。

#### 【０３６４】

＜第４特殊命令／ＲＳＴ命令＞

第４特殊命令の一つである「ＲＳＴ」命令は、固定のアドレスをプログラムカウンタＰＣに設定することによって、固定のアドレスにジャンプするための命令である。

10

#### 【０３６５】

本実施形態では、プログラムカウンタＰＣにストアされるアドレスが異なる８種類のＲＳＴ命令を備えており、ＲＳＴ命令の命令データは、５ビットの命令コード（オペコード）と、３ビットのオペランドによって構成されている。具体的には、命令データＣＦＨ～ＦＦＨの７種類のＲＳＴ命令のビット７～６、２～０の５ビットは、ＲＳＴ命令の命令コード（オペコード）を、それ以外のビット５～３の３ビット（太線で囲まれた部分）は、オペランドとして、各々のＲＳＴ命令の実行によって分岐することが可能なジャンプ先のアドレスの一部（この例では、ジャンプ先のアドレスのビット５～３を示す。アドレスのビット２～０は１固定）を示している。

#### 【０３６６】

例えば、命令データＣＦＨのＲＳＴ命令のビット５～３は、００１Ｂであるため、ジャンプ先のアドレスのビット５～３が００１Ｂで、ビット２～０が０（固定）、すなわち０００８Ｈであることを示しており、命令データＤ７ＨのＲＳＴ命令のビット５～３は０１０Ｂであるため、ジャンプ先のアドレスのビット５～３が０１０Ｂで、ビット２～０が０（固定）、すなわち００１０Ｈであることを示している。また、その他のＲＳＴ命令のジャンプ先のアドレスは図に示す通りである。

20

#### 【０３６７】

一方、命令データＣ７ＨのＲＳＴ命令のビット７～６、２～０の５ビットは、ＲＳＴ命令の命令コード（オペランド）を示しており、それ以外のビット５～３の３ビット（太線で囲まれた部分）は、このＲＳＴ命令の実行によって分岐することが可能なジャンプ先のアドレス（この例では、ビット５～３に格納される数値０００Ｂからは予測が不可能な００４０Ｈ）を示している。すなわち、この命令データＣ７ＨのＲＳＴ命令では、仮に機械語に基づいてオペコードやオペランドを解析できたとしても、当該ＲＳＴ命令の実行によって分岐することが可能なジャンプ先のアドレスを解析することが極めて困難とされている。

30

#### 【０３６８】

また、これらのＲＳＴ命令の命令データは、上述の通り、５ビットの命令コード（オペコード）と３ビットのオペランドによって構成されているため、ユーザはオペランドを自由に設定することができず（ＲＳＴ命令によって分岐することが可能なジャンプ先のアドレスを自由に設定することができず）、上述の８種類のＲＳＴ命令によって分岐することが可能なジャンプ先のアドレスを選択することだけが許されている。

40

#### 【０３６９】

なお、この例では、命令データにオペランドを含めることによって、当該命令によって分岐することが可能なジャンプ先のアドレスを規制する例を示したが、命令データにオペランドを含めなくてもよい。例えば、「ＲＳＴ ００４０Ｈ」のように、オペコード（ＲＳＴ）とオペランド（００４０Ｈ）を別々に構成した上で、指定可能なアドレス以外のアドレス（例えば、００００Ｈ）がオペランドとして設定された場合にアセンブルエラーにして機械語に変換できないように構成するなど、設定可能なオペランドを制限してもよい。また、命令データにオペランドを含む構成を採用した場合でも、指定可能なアドレス以外のアドレスがオペランドとして設定された場合にアセンブルエラーにして機械語に変換

50

できないように構成するなど、設定可能なオペランドを制限してもよい。

#### 【0370】

すなわち、本実施形態に係る RST 命令では 8 種類のアドレス (0008H、0010H、0018H、0020H、0028H、0030H、0038H、0040H) だけにジャンプすることが可能となっており、例えば、RST 命令によって ROM 領域の第一領域の先頭アドレス 0000H にジャンプすることができないように構成されている。このため、不正プログラムを組み込んで先頭アドレスにジャンプさせる (ユーザプログラムを先頭アドレスから開始させて不正に初期状態とする) ような行為を未然に防止できる場合がある。

#### 【0371】

また、命令データ C7H の RST 命令では、命令データには含まれない数値である 0040H をジャンプ先のアドレスとしているため、逆アセンブルなどによってジャンプ先のアドレスやプログラムの流れを特定することが極めて困難で、不正なプログラム改造を未然に防止できる場合がある。また、RST 命令では、図 65 の命令マップに示すように、命令コードの順番を昇順 (C7H→CFH→…) に設定しているのに対して、ジャンプ先のアドレスの順番 (0040H→0008H→…) を昇順にはしないように構成しているため、命令の分析を行うことが極めて困難で、不正なプログラム改造を未然に防止できる場合がある。

#### 【0372】

##### <実施形態 2>

以下、図面を用いて、本発明の実施形態 2 に係る遊技台 (スロットマシン 1100) について詳細に説明する。

#### 【0373】

##### <全体構成>

まず、図 66 を用いて、本発明の実施形態 2 に係るスロットマシン 1100 の全体構成について説明する。なお、同図はスロットマシン 1100 を正面側 (遊技者側) から見た外観斜視図である。

#### 【0374】

同図に示すスロットマシン 1100 は、本体 1101 と、本体 1101 の正面に取り付けられ、本体 1101 に対して開閉可能な前面扉 1102 と、を備える。本体 1101 の中央内部には、(図示省略) 外周面に複数種類の図柄が配置されたリールが 3 個 (左リール 1110、中リール 1111、右リール 1112) 収納され、スロットマシン 1100 の内部で回転できるように構成されている。これらのリール 1110 乃至 1112 はステッピングモータ等の駆動装置により回転駆動される。

#### 【0375】

本実施形態において、各図柄は帯状部材に等間隔で適当数印刷され、この帯状部材が所定の円形筒状の枠材に貼り付けられて各リール 1110 乃至 1112 が構成されている。リール 1110 乃至 1112 上の図柄は、遊技者から見ると、図柄表示窓 1113 から縦方向に概ね 3 つ表示され、合計 9 つの図柄が見えるようになっている。そして、各リール 1110 乃至 1112 を回転させることにより、遊技者から見える図柄の組み合わせが変動することとなる。つまり、各リール 1110 乃至 1112 は複数種類の図柄の組合せを変動可能に表示する表示装置として機能する。なお、このような表示装置としてはリール以外にも液晶表示装置等の電子画像表示装置も採用できる。また、本実施形態では、3 個のリールをスロットマシン 1100 の中央内部に備えているが、リールの数やリールの設置位置はこれに限定されるものではない。

#### 【0376】

各々のリール 1110 乃至 1112 の背面には、図柄表示窓 1113 に表示される個々の図柄を照明するためのバックライト (図示省略) が配置されている。バックライトは、各々の図柄ごとに遮蔽されて個々の図柄を均等に照射できるようにすることが望ましい。なお、スロットマシン 1100 内部において各々のリール 1110 乃至 1112 の近傍に

10

20

30

40

50

は、投光部と受光部から成る光学式センサ（図示省略）が設けられており、この光学式センサの投光部と受光部の間をリールに設けられた一定の長さの遮光片が通過するように構成されている。このセンサの検出結果に基づいてリール上の図柄の回転方向の位置を判断し、目的とする図柄が入賞ライン上に表示されるようにリール 1 1 1 0 乃至 1 1 1 2 を停止させる。

#### 【0377】

入賞ライン表示ランプ 1 1 2 0 は、有効となる入賞ライン 1 1 1 4 を示すランプである。有効となる入賞ラインは、遊技媒体としてベットされたメダルの数によって予め定められている。入賞ライン 1 1 1 4 は 5 ラインあり、例えば、メダルが 1 枚ベットされた場合、中段の水平入賞ラインが有効となり、メダルが 2 枚ベットされた場合、上段水平入賞ラインと下段水平入賞ラインが追加された 3 本が有効となり、メダルが 3 枚ベットされた場合、右下り入賞ラインと右上り入賞ラインが追加された 5 ラインが入賞ラインとして有効になる。なお、入賞ライン 1 1 1 4 の数については 5 ラインに限定されるものではなく、また、例えば、メダルが 1 枚ベットされた場合に、中段の水平入賞ライン、上段水平入賞ライン、下段水平入賞ライン、右下り入賞ラインおよび右上り入賞ラインの 5 ラインを有効な入賞ラインとして設定してもよく、ベット数に関係なく、一律に同一数の入賞ラインを有効な入賞ラインとして設定してもよい。

10

#### 【0378】

告知ランプ 1 1 2 3 は、例えば、後述する内部抽選において特定の入賞役（具体的には、ボーナス）に内部当選していること、または、ボーナス遊技中であることを遊技者に知らせるランプである。遊技メダル投入可能ランプ 1 1 2 4 は、遊技者が遊技メダルを投入可能であることを知らせるためのランプである。再遊技ランプ 1 1 2 2 は、前回の遊技において入賞役の一つである再遊技に入賞した場合に、今回の遊技が再遊技可能であること（メダルの投入が不要であること）を遊技者に知らせるランプである。リールパネルランプ 1 1 2 8 は演出用のランプである。

20

#### 【0379】

ベットボタン 1 1 3 0 乃至 1 1 3 2 は、スロットマシン 1 1 0 0 に電子的に貯留されているメダル（クレジットという）を所定の枚数分投入するためのボタンである。本実施形態においては、ベットボタン 1 1 3 0 が押下される毎に 1 枚ずつ最大 3 枚まで投入され、ベットボタン 1 1 3 1 が押下されると 2 枚投入され、ベットボタン 1 1 3 2 が押下されると 3 枚投入されるようになっている。以下、ベットボタン 1 1 3 2 は MAX ベットボタンとも言う。なお、遊技メダル投入ランプ 1 1 2 9 は、投入されたメダル数に応じた数のランプを点灯させ、規定枚数のメダルの投入があった場合、遊技の開始操作が可能な状態であることを知らせる遊技開始ランプ 1 1 2 1 が点灯する。

30

#### 【0380】

メダル投入口 1 1 4 1 は、遊技を開始するに当たって遊技者がメダルを投入するための投入口である。すなわち、メダルの投入は、ベットボタン 1 1 3 0 乃至 1 1 3 2 により電子的に投入することもできるし、メダル投入口 1 1 4 1 から実際のメダルを投入（投入操作）することもでき、投入とは両者を含む意味である。貯留枚数表示器 1 1 2 5 は、スロットマシン 1 1 0 0 に電子的に貯留されているメダルの枚数を表示するための表示器である。遊技情報表示器 1 1 2 6 は、各種の内部情報（例えば、ボーナス遊技中のメダル払出枚数）を数値で表示するための表示器である。払出枚数表示器 1 1 2 7 は、何らかの入賞役に入賞した結果、遊技者に払出されるメダルの枚数を表示するための表示器である。貯留枚数表示器 1 1 2 5、遊技情報表示器 1 1 2 6、および、払出枚数表示器 1 1 2 7 は、7 セグメント（SEG）表示器とした。

40

#### 【0381】

スタートレバー 1 1 3 5 は、リール 1 1 1 0 乃至 1 1 1 2 の回転を開始させるためのレバー型のスイッチである。即ち、メダル投入口 1 1 4 1 に所望するメダル枚数を投入するか、ベットボタン 1 1 3 0 乃至 1 1 3 2 を操作して、スタートレバー 1 1 3 5 を操作すると、リール 1 1 1 0 乃至 1 1 1 2 が回転を開始することとなる。スタートレバー 1 1 3 5

50

に対する操作を遊技の開始操作と言う。

#### 【0382】

ストップボタンユニット1136には、ストップボタン1137乃至1139が設けられている。ストップボタン1137乃至1139は、スタートレバー1135の操作によって回転を開始したリール1110乃至1112を個別に停止させるためのボタン型のスイッチであり、各リール1110乃至1112に対応づけられている。以下、ストップボタン1137乃至1139に対する操作を停止操作と言い、最初の停止操作を第1停止操作、次の停止操作を第2停止操作、最後の停止操作を第3停止操作という。なお、各ストップボタン1137乃至1139の内部に発光体を設けてもよく、ストップボタン1137乃至1139の操作が可能である場合、該発光体を点灯させて遊技者に知らせることもできる。

10

#### 【0383】

メダル返却ボタン1133は、投入されたメダルが詰まった場合に押下してメダルを取り除くためのボタンである。精算ボタン1134は、スロットマシン1100に電子的に貯留されたメダル、ベットされたメダルを精算し、メダル払出口1155から排出するためのボタンである。ドアキー孔1140は、スロットマシン1100の前面扉1102のロックを解除するためのキーを挿入する孔である。ストップボタンユニット1136の下部には、機種名の表示と各種証紙の貼付とを行うタイトルパネル1162が設けられている。タイトルパネル1162の下部には、メダル払出口1155、メダルの受け皿1161が設けられている。

20

#### 【0384】

音孔1180はスロットマシン1100内部に設けられているスピーカの音を外部に出力するための孔である。前面扉1102の左右各部に設けられたサイドランプ1144は遊技を盛り上げるための装飾用のランプである。前面扉1102の上部には演出装置1160が配設されており、演出装置1160の上部には音孔1143が設けられている。この演出装置1160は、水平方向に開閉自在な2枚の右シャッタ1163a、左シャッタ1163bからなるシャッタ（遮蔽装置）1163と、このシャッタ1163の奥側に配設された液晶表示装置1157（演出画像表示装置）を備えており、右シャッタ1163a、左シャッタ1163bが液晶表示装置1157の手前で水平方向外側に開くと液晶表示装置1157の表示画面がスロットマシン1100正面（遊技者側）に出現する構造となっている。なお、液晶表示装置でなくとも、種々の演出画像や種々の遊技情報を表示可能な表示装置であればよく、例えば、複数セグメントディスプレイ（7セグディスプレイ）、ドットマトリクスディスプレイ、有機ELディスプレイ、プラズマディスプレイ、リール（ドラム）、或いは、プロジェクタとスクリーンとからなる表示装置等でもよい。また、表示画面は、方形をなし、その全体を遊技者が視認可能に構成している。本実施形態の場合、表示画面は長方形であるが、正方形でもよい。また、表示画面の周縁に不図示の装飾物を設けて、表示画面の周縁の一部が該装飾物に隠れる結果、表示画面が異形に見えるようにすることもできる。表示画面は本実施形態の場合、平坦面であるが、曲面をなしていてもよい。

30

#### 【0385】

図67は、前面扉102を開けた状態のスロットマシン1100を示す正面図である。本体1101は、上面板1261、左側の側面板1260、右側の側面板1260、下面板1264および背面板1242で囲われ、前面に開口する箱体である。本体1101の内部には、背面板1242の上部に設けた通風口1249と重ならない位置に、内部に主制御基板を収納した主制御基板収納ケース1210が配置され、この主制御基板収納ケース1210の下方に、3つのリール1110乃至1112が配置されている。主制御基板収納ケース1210及びリール1110乃至1112の側方、即ち向って左側の側面板1260には、内部に副制御基板を収納した副制御基板収納ケース1220が配設してある。また、向かって右側の側面板1260には、主制御基板に接続されて、スロットマシン1100の情報を外部装置に出力する外部集中端子板1248が取り付けられている。

40

50

## 【0386】

そして、下面板1264には、メダル払出装置1180（バケットに溜まったメダルを払出す装置）が配設され、このメダル払出装置1180の上方、即ちリール1110乃至1112の下方には、電源基板を有する電源装置1252が配設され、電源装置1252正面には電源スイッチ1244を配設している。電源装置1252は、スロットマシン1100に外部から供給される交流電源を直流化し、所定の電圧に変換して後述する主制御部1300、副制御部1400、1500等の各制御部、各装置に供給する。さらには、外部からの電源が断たれた後も所定の部品（例えば主制御部1300のRAM1308等）に所定の期間（例えば10日間）電源を供給するための蓄電回路（例えばコンデンサ）を備えている。

10

## 【0387】

メダル払出装置1180の右側には、メダル補助収納庫1240が配設してあり、この背後にはオーバフロー端子が配設されている（図示省略）。電源装置1252には、電源コード1264を接続する電源コード接続部が設けられ、ここに接続された電源コード1264が、本体1101の背面板1242に開設した電源コード用穴1262を通して外部に延出している。

## 【0388】

前面扉1102は、本体1101の左側の側面板1260にヒンジ装置1276を介して蝶着され、図柄表示窓1113の上部には、演出装置1160、および、この演出装置1160を制御する演出制御基板（図示省略）、上部スピーカ1272、を設けている。図柄表示窓1113の下部には、投入されたメダルを選別するためのメダルセクタ1170、このメダルセクタ1170が不正なメダル等をメダル受皿1161に落下させる際にメダルが通過する通路1266等を設けている。さらに、音孔1180に対応する位置には低音スピーカ1277を設けている。

20

## 【0389】

図68を用いて、スロットマシン1100の制御部の回路構成について詳細に説明する。なお、同図は制御部の回路ブロック図を示したものである。

## 【0390】

スロットマシン1100の制御部は、大別すると、主に遊技の進行（例えば、遊技者による操作の検出、遊技状態の遷移、遊技媒体の払出制御、当否判定など）を制御する主制御部1300と、主制御部1300が送信するコマンド信号（以下、単に「コマンド」と呼ぶ）に応じて、主に演出の制御を行う第1副制御部1400と、第1副制御部1400より送信されたコマンドに基づいて各種機器を制御する第2副制御部1500と、によって構成されている。

30

## 【0391】

## &lt;主制御部&gt;

まず、スロットマシン1100の主制御部1300について説明する。主制御部1300は、主制御部1300の全体を制御する基本回路1302を備えており、この基本回路1302には、CPU1304と、制御プログラムデータ、入賞役の内部抽選時に用いる抽選データ、リールの停止位置等を記憶するためのROM1306と、一時的にデータを記憶するためのRAM1308と、各種デバイスの入出力を制御するためのI/O1310と、時間や回数等を計測するためのカウンタタイマ1312と、WDT（ウォッチドックタイマ）1314を搭載している。なお、ROM1306やRAM1308については他の記憶装置を用いてもよく、この点は後述する第1副制御部1400や第2副制御部500についても同様である。この基本回路1302のCPU1304は、水晶発振器1314が出力する所定周期のクロック信号をシステムクロックとして入力して動作する。さらには、CPU1304は、電源が投入されるとROM1306の所定エリアに格納された分周用のデータをカウンタタイマ1312に送信し、カウンタタイマ1312は受信した分周用のデータを基に割り込み時間を決定し、この割り込み時間ごとに割り込み要求をCPU1304に送信する。CPU1304は、この割り込み要求を契機に各センサ等の監

40

50

視や駆動パルスの送信を実行する。例えば、水晶発振器 1 3 1 4 が出力するクロック信号を 8 M H z、カウンタタイマ 1 3 1 2 の分周値を 1 / 2 5 6、ROM 1 3 0 6 の分周用のデータを 4 7 に設定した場合、割り込みの基準時間は、 $2 5 6 \times 4 7 \div 8 \text{ M H z} = 1. 5 0 4 \text{ m s}$ となる。

#### 【0 3 9 2】

主制御部 1 3 0 0 は、0 ~ 6 5 5 3 5 の範囲で数値を変動させるハードウェア乱数カウンタとして使用している乱数値生成回路 1 3 1 6 と、電源が投入されると起動信号（リセット信号）を出力する起動信号出力回路 1 3 3 8 を備えており、CPU 1 3 0 4 は、この起動信号出力回路 1 3 3 8 から起動信号が入力された場合に、遊技制御を開始する（後述する主制御部メイン処理を開始する）。

10

#### 【0 3 9 3】

また、主制御部 1 3 0 0 には、センサ回路 1 3 2 0 を備えており、CPU 1 3 0 4 は、割り込み時間ごとに各種センサ 1 3 1 8（ベットボタン 1 1 3 0 センサ、ベットボタン 1 1 3 1 センサ、ベットボタン 1 1 3 2 センサ、メダル投入口 1 1 4 1 から投入されたメダルのメダル受付センサ、スタートレバー 1 1 3 5 センサ、ストップボタン 1 1 3 7 センサ、ストップボタン 1 1 3 8 センサ、ストップボタン 1 1 3 9 センサ、精算ボタン 1 1 3 4 センサ、メダル払出装置 1 1 8 0 から払い出されるメダルのメダル払出センサ、リール 1 1 1 0 のインデックスセンサ、リール 1 1 1 1 のインデックスセンサ、リール 1 1 1 2 のインデックスセンサ、等）の状態を監視している。

20

#### 【0 3 9 4】

なお、センサ回路 1 3 2 0 がスタートレバーセンサの H レベルを検出した場合には、この検出を示す信号を乱数発生回路 1 3 1 6 に出力する。この信号を受信した乱数発生回路 1 3 1 6 は、そのタイミングにおける値をラッチし、抽選に使用する乱数値を格納するレジスタに記憶する。

#### 【0 3 9 5】

メダル受付センサは、メダル投入口 1 1 4 1 の内部通路に 2 個設置されており、メダルの通過有無を検出する。スタートレバー 1 1 3 5 センサは、スタートレバー 1 1 3 5 内部に 2 個設置されており、遊技者によるスタート操作を検出する。ストップボタン 1 1 3 7 センサ、ストップボタン 1 1 3 8 センサ、および、ストップボタン 1 1 3 9 は、各々のストップボタン 1 1 3 7 乃至 1 1 3 9 に設置されており、遊技者によるストップボタンの操

30

#### 【0 3 9 6】

ベットボタン 1 1 3 0 センサ、ベットボタン 1 1 3 1 センサ、および、ベットボタン 1 1 3 2 センサは、メダル投入ボタン 1 1 3 0 乃至 1 1 3 2 のそれぞれに設置されており、RAM 1 3 0 8 に電子的に貯留されているメダルを遊技への投入メダルとして投入する場合の投入操作を検出する。精算ボタン 1 1 3 4 センサは、精算ボタン 1 1 3 4 に設けられている。精算ボタン 1 1 3 4 が一回押されると、電子的に貯留されているメダルを精算する。メダル払出センサは、メダル払出装置 1 1 8 0 が払い出すメダルを検出するためのセンサである。なお、以上の各センサは、非接触式のセンサであっても接点式のセンサであってもよい。

40

#### 【0 3 9 7】

リール 1 1 1 0 のインデックスセンサ、リール 1 1 1 1 のインデックスセンサ、および、リール 1 1 1 2 のインデックスセンサは、各リール 1 1 1 0 乃至 1 1 1 2 の取付台の所定位置に設置されており、リールフレームに設けた遮光片が通過するたびに L レベルになる。CPU 1 3 0 4 は、この信号を検出すると、リールが 1 回転したものと判断し、リールの回転位置情報をゼロにリセットする。

#### 【0 3 9 8】

主制御部 1 3 0 0 は、リール装置 1 1 1 0 乃至 1 1 1 2 に設けたステッピングモータを駆動する駆動回路 1 3 2 2、投入されたメダルを選別するメダルセレクト 1 1 7 0 に設けたソレノイドを駆動する駆動回路 1 3 2 4、メダル払出装置 1 1 8 0 に設けたモータを駆

50



動する駆動回路1326、各種ランプ1338（入賞ライン表示ランプ1120、告知ランプ1123、遊技メダル投入可能ランプ1124、再遊技ランプ1122、遊技メダル投入ランプ1129は、遊技開始ランプ1121、貯留枚数表示器1125、遊技情報表示器1126、払出枚数表示器1127）を駆動する駆動回路1328を備えている。

#### 【0399】

また、基本回路1302には、情報出力回路1334（外部集中端子板1248）を接続しており、主制御部1300は、この情報出力回路1334を介して、外部のホールコンピュータ（図示省略）等が備える情報入力回路1652にスロットマシン1100の遊技情報（例えば、遊技状態）を出力する。

#### 【0400】

また、主制御部1300は、電源管理部（図示しない）から主制御部1300に供給している電源の電圧値を監視する電圧監視回路1330を備えており、電圧監視回路1330は、電源の電圧値が所定の値（本実施例では9V）未満である場合に電圧が低下したことを示す低電圧信号を基本回路1302に出力する。

#### 【0401】

また、主制御部1300は、第1副制御部1400にコマンドを送信するための出力インタフェースを備えており、第1副制御部1400との通信を可能としている。なお、主制御部1300と第1副制御部1400との情報通信は一方方向の通信であり、主制御部1300は第1副制御部1400にコマンド等の信号を送信できるように構成しているが、第1副制御部1400からは主制御部1300にコマンド等の信号を送信できないように構成している。

#### 【0402】

##### <副制御部>

次に、スロットマシン1100の第1副制御部1400について説明する。第1副制御部1400は、主制御部1300が送信した制御コマンドを入力インタフェースを介して受信し、この制御コマンドに基づいて第1副制御部1400の全体を制御する基本回路1402を備えており、この基本回路1402は、CPU1404と、一時的にデータを記憶するためのRAM1408と、各種デバイスの入出力を制御するためのI/O1410と、時間や回数等を計測するためのカウンタタイマ1412を搭載している。基本回路1402のCPU1404は、水晶発振器1414が出力する所定期期のクロック信号をシステムクロックとして入力して動作する。ROM406は、第1副制御部1400の全体を制御するための制御プログラム及びデータ、バックライトの点灯パターンや各種表示器を制御するためのデータ等を記憶する。

#### 【0403】

CPU1404は、所定のタイミングでデータバスを介してROM1406の所定エリアに格納された分周用のデータをカウンタタイマ1412に送信する。カウンタタイマ1412は、受信した分周用のデータを基に割り込み時間を決定し、この割り込み時間ごとに割り込み要求をCPU1404に送信する。CPU1404は、この割り込み要求のタイミングをもとに、各ICや各回路を制御する。

#### 【0404】

また、第1副制御部1400には、音源IC1418を設けており、音源IC1418に出力インタフェースを介してスピーカ1272、1277を設けている。音源IC1418は、CPU1404からの命令に応じてアンプおよびスピーカ1272、1277から出力する音声の制御を行う。音源IC1418には音声データが記憶されたS-ROM（サウンドROM）が接続されており、このROMから取得した音声データをアンプで増幅させてスピーカ1272、1277から出力する。

#### 【0405】

また、第1副制御部1400には、駆動回路1422が設けられ、駆動回路1422に入出力インタフェースを介して各種ランプ1420（上部ランプ、下部ランプ、サイドランプ1144、タイトルパネル1162ランプ、等）が接続されている。

## 【0406】

また、CPU1404は、出力インタフェースを介して第2副制御部1500へ信号の送受信を行う。第2副制御部1500は、演出画像表示装置1157の表示制御を含む演出装置1160の各種制御を行う。なお、第2副制御部1500は、例えば、液晶表示装置1157の表示の制御を行う制御部、各種演出用駆動装置の制御を行う制御部（例えば、シャッタ1163のモータ駆動を制御する制御部）とするなど、複数の制御部で構成するようにしてもよい。

## 【0407】

第2副制御部1500は、第1副制御部1400が送信した制御コマンドを入力インタフェースを介して受信し、この制御コマンドに基づいて第2副制御部1500の全体を制御する基本回路1502を備えており、この基本回路1502は、CPU1504と、一時的にデータを記憶するためのRAM1508と、各種デバイスの入出力を制御するためのI/O1510と、時間や回数等を計測するためのカウンタタイマ1512と、を搭載している。基本回路1502のCPU1504は、水晶発振器1514が出力する所定周期のクロック信号をシステムクロックとして入力して動作する。ROM506は、第2副制御部1500の全体を制御するための制御プログラム及びデータ、画像表示用のデータ等を記憶する。

## 【0408】

CPU1504は、所定のタイミングでデータバスを介してROM1506の所定エリアに格納された分周用のデータをカウンタタイマ1512に送信する。カウンタタイマ1512は、受信した分周用のデータを基に割り込み時間を決定し、この割り込み時間ごとに割り込み要求をCPU1404に送信する。CPU1504は、この割り込み要求のタイミングをもとに、各ICや各回路を制御する。

## 【0409】

また、第2副制御部1500には、シャッタ1163のモータを駆動する駆動回路1530を設けており、駆動回路1530には出力インタフェースを介してシャッタ1163を設けている。この駆動回路1530は、CPU1504からの命令に応じてシャッタ1163に設けたステッピングモータ（図示省略）に駆動信号を出力する。また、第2副制御部1500には、センサ回路1532を設けており、センサ回路1532には入力インタフェースを介してシャッタセンサ1538を接続している。CPU1504は、割り込み時間ごとにシャッタセンサ1538状態を監視している。

## 【0410】

また、第2副制御部1500には、VDP1534（ビデオ・ディスプレイ・プロセッサ）を設けており、このVDP1534には、バスを介してROM1506、VRAM1536が接続されている。VDP1534は、CPU1504からの信号に基づいてROM1506に記憶された画像データ等を読み出し、VRAM1536のワークエリアを使用して表示画像を生成し、演出画像表示装置1157に画像を表示する。

## 【0411】

## &lt;図柄配列&gt;

次に、図69（a）を用いて、上述の各リール1110乃至1112に施される図柄配列について説明する。なお、同図は、各リール（左リール1110、中リール1111、右リール1112）に施される図柄の配列を平面的に展開して示した図である。

## 【0412】

各リール1110乃至1112には、同図の右側に示す複数種類（本実施形態では8種類）の図柄が所定コマ数（本実施形態では、番号0～20の21コマ）だけ配置されている。また、同図の左端に示した番号0～20は、各リール1110乃至1112上の図柄の配置位置を示す番号である。例えば、本実施形態では、左リール1110の番号1のコマには「リプレイ」の図柄、中リール1111の番号0のコマには「ベル」の図柄、右リール1112の番号2のコマには「スイカ」の図柄、がそれぞれ配置されている。

## 【0413】

### ＜入賞役の種類＞

次に、図 6 9 (b) を用いて、スロットマシン 1 1 0 0 の入賞役の種類について説明する。なお、同図は入賞役（作動役を含む）の種類、各入賞役に対応する図柄組合せ、各入賞役の作動または払出を示している。

#### 【0 4 1 4】

本実施形態における入賞役のうち、ビッグボーナス（BB 1、BB 2）および、レギュラーボーナス（RB）はボーナス遊技に移行する役として、また、再遊技（リプレイ）は新たにメダルを投入することなく再遊技が可能となる役として、それぞれ入賞役とは区別され「作動役」と呼ばれる場合があるが、本実施形態における「入賞役」には、作動役である、ビッグボーナス、レギュラーボーナス、再遊技が含まれる。また、本実施形態にお

10

#### 【0 4 1 5】

スロットマシン 1 1 0 0 の入賞役には、ビッグボーナス（BB 1、BB 2）と、レギュラーボーナス（RB）と、小役（チェリー、スイカ、ベル）と、再遊技（リプレイ）がある。なお、入賞役の種類は、これに限定されるものではなく、任意に採用できることは言うまでもない。

#### 【0 4 1 6】

「ビッグボーナス（BB 1、BB 2）」（以下、単に、「BB」と称する場合がある）は、入賞により特別遊技であるビッグボーナス遊技（BB 遊技）が開始される特別役（作動役）である。対応する図柄組合せは、BB 1 が「白 7－白 7－白 7」、BB 2 が「青 7－青 7－青 7」である。また、BB 1、BB 2 についてはフラグ持越しを行う。すなわち、BB 1、BB 2 に内部当選すると、これを示すフラグが立つ（主制御部 1 3 0 0 の RAM 1 3 0 8 の当否判定結果記憶領域に記憶される）が、その遊技において BB 1、BB 2 に入賞しなかったとしても、入賞するまで内部当選を示すフラグが立った状態が維持され、次遊技以降でも BB 1、BB 2 に内部当選中となり、BB 1 に対応する図柄組み合わせ「白 7－白 7－白 7」、BB 2 に対応する図柄組み合わせ「青 7－青 7－青 7」が、揃って入賞する状態にある。

20

#### 【0 4 1 7】

「レギュラーボーナス（RB）」は、入賞によりレギュラーボーナス遊技（RB 遊技）が開始される特殊役（作動役）である。対応する図柄組合せは、「ボーナスーボーナスーボーナス」である。なお、RB についても上述の BB と同様にフラグ持越しを行う。但し、（詳細は後述するが）ビッグボーナス遊技（BB 遊技）においては、レギュラーボーナス遊技（RB 遊技）が内部当選することや、図柄組み合わせが入賞ライン上に表示されること、を開始条件とせず、ビッグボーナス遊技の開始後からレギュラーボーナス遊技を開始し、1 回のレギュラーボーナス遊技を終了した場合には次のレギュラーボーナス遊技をすぐに開始するような自動的にレギュラーボーナス遊技を開始させる設定としてもよい。「小役（チェリー、スイカ、ベル）」（以下、単に、「チェリー」、「スイカ」、「ベル」と称する場合がある）は、入賞により所定数のメダルが払い出される入賞役で、対応する図柄組合せは、チェリーが「チェリーーANYーANY」、スイカが「スイカースイカースイカ」、ベルが「ベルーベルーベル」である。また、対応する払出枚数は同図に示す通りである。なお、「チェリーーANYーANY」の場合、左リール 1 1 1 0 の図柄が「チェリー」であればよく、中リール 1 1 1 1 と右リール 1 1 1 2 の図柄はどの図柄でもよい。

30

40

#### 【0 4 1 8】

「再遊技（リプレイ）」は、入賞により次回の遊技でメダル（遊技媒体）の投入を行うことなく遊技を行うことができる入賞役（作動役）であり、メダルの払出は行われない。なお、対応する図柄組合せは、再遊技は「リプレイーリプレイーリプレイ」である。

### ＜遊技状態の種類＞

50

次に、スロットマシン 1 1 0 0 の遊技状態の種類について説明する。遊技状態とは、抽選などにおいて選択する抽選データの種別を識別するための情報である。本実施形態では、スロットマシン 1 1 0 0 の遊技状態は、通常遊技と、B B 遊技と、R B 遊技と、ビッグボーナス (B B) およびレギュラーボーナス (R B) の内部当選遊技と、に大別した。但し、内部当選遊技は、通常遊技に含まれる区分けであってもよい。

#### < 通常遊技 >

通常遊技に内部当選する入賞役には、ビッグボーナス (B B) と、レギュラーボーナス (R B) と、再遊技 (リプレイ) と、小役 (チェリー、スイカ、ベル) がある。

#### 【 0 4 1 9 】

「ビッグボーナス (B B)」は、入賞により特別遊技であるビッグボーナス遊技 (B B 遊技) が開始される特別役 (作動役) である。レギュラーボーナス (R B)」は、入賞によりレギュラーボーナス遊技 (R B 遊技) を開始する特殊役 (作動役) である。「再遊技 (リプレイ)」は、入賞により次の遊技でメダルの投入を行うことなく遊技を行うことができる入賞役 (作動役) であり、メダルの払出も行われない。「小役」は、入賞により所定数のメダルが払い出される入賞役である。なお、各々の役の内部当選確率は、通常遊技に用意された抽選データから、各々の役に対応付けされた抽選データの範囲に該当する数値データを、内部抽選時に取得される乱数値の範囲の数値データ (例えば 6 5 5 3 5) で除した値で求められる。通常遊技に用意された抽選データは、予めいくつかの数値範囲に分割され、各数値範囲に各々の役やハズレに対応付けしている。内部抽選を実行した結果得られた乱数値が、何れの役に対応する抽選データに対応する値であったかを判定し、内部抽選役を決定する。この抽選データは少なくとも 1 つの役の当選確率を異ならせた設定 1 ~ 設定 6 が用意され、遊技店の係員等はいずれかの設定値を任意に選択し、設定することができる。

#### 【 0 4 2 0 】

通常遊技は、内部抽選の結果が概ねハズレ (ビッグボーナス (B B)、レギュラーボーナス (R B)、再遊技 (リプレイ) および小役に当選していない) となる設定、又は、停止表示結果がいずれの役の図柄組合せに該当しないハズレの停止表示結果が概ね導出される設定がされており、獲得するメダルの総数が、投入したメダルの総数に満たない遊技状態になっている。よって、遊技者にとっては不利益となる遊技状態である。但し、予め定めた条件を満たした場合 (例えば、特定の図柄組み合わせが表示された場合) には、再遊技の内部当選の確率を上昇させる変動をさせてもよい遊技状態であり、この場合、遊技に用いられるメダルの消費が抑えられ、小役の入賞によって所定数のメダルが払い出されることにより、獲得するメダルの総数が、投入したメダルの総数を超える遊技状態になり、遊技者にとっては利益となる遊技状態になる場合がある。

#### 【 0 4 2 1 】

##### < B B 遊技 >

B B 遊技は、遊技者にとっては利益となる遊技状態になるように設定されている。つまり、B B 遊技は、獲得するメダルの総数が、投入したメダルの総数を超える遊技状態となる。B B 遊技は、本実施形態では、ビッグボーナス (B B) の入賞により開始され、R B 遊技 (後述する) を連続して繰り返し実行可能になっており、遊技中に予め定められた一の数 (例えば、4 6 5 枚) を超えるメダルが獲得された場合に終了する。但し、B B 遊技は R B 遊技を複数回数実行可能であればよく、例えば、R B 遊技を開始する役 (図柄組み合わせは例えば、リプレイーリプレイーリプレイ) を設定し、この役が内部当選した場合、または、入賞した場合に、R B 遊技を開始するように設定してもよい。さらには、B B 遊技は、B B 遊技中の R B 遊技を除く B B 一般遊技を予め定めた回数 (例えば、3 0 回) 実行した場合、または、B B 遊技中に実行した R B 遊技の回数が予め定めた回数に達した場合 (例えば、3 回) に終了するようにしてもよい。

#### 【 0 4 2 2 】

##### < R B 遊技 >

R B 遊技は、遊技者にとっては利益となる遊技状態になるように設定されている。つま

り、R B遊技は、獲得するメダルの総数が、投入したメダルの総数を超える遊技状態となる。R B遊技は、本実施形態では、レギュラーボーナス（R B）の入賞により開始され、予め定めた一の役が内部当選の確率を上昇させる変動（例えば、「設定1」「通常遊技」に設定された「小役1」の内部当選確率 $1/15$ を、予め定めた一の値である内部当選確率 $1/1.2$ に上昇させる）をし、予め定めた一の数（例えば、8回）の入賞があった場合に終了する。R B遊技は、予め定めた回数（少なくとも2回）の入賞があった場合（例えば、8回）、または、R B遊技中に実行したR B遊技の回数が予め定めた回数に達した場合（例えば、8回）に終了するようにしてもよい。上述したB B遊技は、R B遊技を複数回数実行可能であるので、一回のR B遊技を行った場合には、B B遊技で得られるメダルの総数よりも少ないメダル数を獲得して終了することとなる。

10

#### 【0423】

＜ビッグボーナス（B B）およびレギュラーボーナス（R B）の内部当選遊技＞

ビッグボーナス（B B）およびレギュラーボーナス（R B）の内部当選遊技に内部当選する入賞役には、再遊技（リプレイ）と、小役がある。ビッグボーナス（B B）およびレギュラーボーナス（R B）は内部当選することはなく、ビッグボーナス（B B）かレギュラーボーナス（R B）に対応する図柄組み合わせを入賞させることが可能となっている遊技状態である。

#### 【0424】

但し、ビッグボーナス（B B）およびレギュラーボーナス（R B）に内部当選した次遊技から、再遊技の内部当選の確率を変動させてもよく、例えば、再遊技の内部当選の確率を上昇させる変動をさせて、ビッグボーナス（B B）およびレギュラーボーナス（R B）に対応する図柄組み合わせが入賞するまでの間は、獲得するメダルの総数が、投入したメダルの総数とほぼ同じとなる遊技状態とし、通常遊技と比べると遊技者にとっては利益となる遊技状態としてもよい。なお、B B遊技、R B遊技は両者とも遊技者にとって利益となる遊技状態であるため、総じて、ボーナス遊技、又は、特別遊技と称する場合がある。

20

#### 【0425】

＜主制御部メイン処理＞

次に、図70を用いて、主制御部1300のCPU1304が実行する主制御部メイン処理について説明する。なお、同図は主制御部メイン処理の流れを示すフローチャートである。

30

#### 【0426】

上述したように、主制御部1300には、電源が投入されると起動信号（リセット信号）を出力する起動信号出力回路（リセット信号出力回路）1338を設けている。この起動信号を入力した基本回路1302のCPU1304は、リセット割込によりリセットスタートしてROM1306に予め記憶している制御プログラムに従って、この主制御部メイン処理を実行する。

#### 【0427】

電源投入が行われると、まず、ステップS2101で各種の初期設定を行う。この初期設定では、CPU1304のスタックポインタ（SP）へのスタック初期値の設定、割込禁止の設定、I/O1310の初期設定、RAM1308に記憶する各種変数の初期設定、WDT1314への動作許可及び初期値の設定等を行う。ステップS2103ではメダル投入・スタート操作受付処理を実行する。ここではメダルの投入の有無をチェックし、メダルの投入に応じて入賞ライン表示ランプ1120を点灯させる。なお、前回の遊技で再遊技に入賞した場合は、前回の遊技で投入されたメダル枚数と同じ数のメダルを投入する処理を行うので、遊技者によるメダルの投入が不要となる。また、スタートレバー1135が操作されたか否かのチェックを行い、スタートレバー1135の操作があればステップS2105へ進む。

40

#### 【0428】

ステップS2105では投入されたメダル枚数を確定し、有効な入賞ラインを確定する。ステップS2107では後述する主制御部タイマ割込み処理で格納された乱数値を取得

50

する。ステップS 2 1 0 9では、現在の遊技状態に応じてROM 1 3 0 6に格納されている入賞役抽選テーブルを読み出し、これとステップS 2 1 0 7で取得した乱数値とを用いて内部抽選を行う。内部抽選の結果、いずれかの入賞役（作動役を含む）に内部当選した場合、その入賞役のフラグがONになる。ステップS 2 1 1 1では内部抽選結果に基づき、リール停止データを選択する。なお、本実施形態では乱数値生成回路1 3 1 6から取得した乱数をそのまま内部抽選に使用するように構成しているが、取得した乱数を加工し、加工した乱数を使用して内部抽選を行うように構成してもよい。

#### 【0 4 2 9】

ステップS 2 1 1 3では全リール1 1 1 0乃至1 1 1 2の回転を開始させる。ステップS 2 1 1 5では、ストップボタン1 1 3 7乃至1 1 3 9の受け付けが可能になり、いずれかのストップボタンが押されると、押されたストップボタンに対応するリール1 1 1 0乃至1 1 1 2の何れかをステップS 2 1 1 1で選択したリール停止制御データに基づいて停止させる。全リール1 1 1 0乃至1 1 1 2が停止するとステップS 2 1 1 7へ進む。ステップS 2 1 1 7では、入賞判定を行う。ここでは、有効化された入賞ライン1 1 1 4上に、何らかの入賞役に対応する図柄組合せが表示された場合にその入賞役に入賞したと判定する。例えば、有効化された入賞ライン上に「ベルーベルーベル」が揃っていたならばベル入賞と判定する。ステップS 2 1 1 9では払い出しのある何らかの入賞役に入賞していれば、その入賞役に対応する枚数のメダルを入賞ライン数に応じて払い出す。ステップS 2 1 2 1では遊技状態制御処理を行う。遊技状態制御処理では、通常遊技、BB遊技、RB遊技、内部当選遊技、の各遊技状態の移行に関する処理を行い、それらの開始条件、終了条件の成立により、遊技状態を移行する。以上により1ゲームが終了する。以降ステップS 2 1 0 3へ戻って上述した処理を繰り返すことにより遊技が進行することになる。

#### 【0 4 3 0】

＜主制御部1 3 0 0タイマ割込処理＞

次に、図7 1を用いて、主制御部1 3 0 0のCPU 1 3 0 4が実行する主制御部タイマ割込処理について説明する。なお、同図は主制御部タイマ割込処理の流れを示すフローチャートである。

#### 【0 4 3 1】

主制御部1 3 0 0は、所定の周期（本実施形態では約2 m sに1回）でタイマ割込信号を発生するカウンタタイマ1 3 1 2を備えており、このタイマ割込信号を契機として主制御部タイマ割込処理を所定の周期で開始する。

#### 【0 4 3 2】

ステップS 2 2 0 1では、タイマ割込開始処理を行う。このタイマ割込開始処理では、CPU 1 3 0 4の各レジスタの値をスタック領域に一時的に退避する処理などを行う。ステップS 2 2 0 3では、WDT 1 3 1 4のカウント値が初期設定値（本実施形態では3 2 . 8 m s）を超えてWDT割込が発生しないように（処理の異常を検出しないように）、WDT 1 3 1 4を定期的に（本実施形態では、主制御部タイマ割込の周期である約2 m sに1回）リスタートを行う。

#### 【0 4 3 3】

ステップS 2 2 0 5では、入力ポート状態更新処理を行う。この入力ポート状態更新処理では、I/O 1 3 1 0の入力ポートを介して、各種センサ1 3 1 8のセンサ回路1 3 2 0の検出信号を入力して検出信号の有無を監視し、RAM 1 3 0 8に各種センサ1 3 1 8ごとに区画して設けた信号状態記憶領域に記憶する。より具体的には、この信号状態記憶領域それぞれに対して前々回、前回、今回の割込み時における検出信号の有無を記憶している。

#### 【0 4 3 4】

ステップS 2 2 0 7では、各種遊技処理を行う。具体的には、割込みステータスを取得し（各種センサ1 3 1 8からの信号に基づいて各種割込みステータスを取得する）、この割込みステータスに従った処理を行う。例えば、この割込みステータスが遊技開始のためのスタートレバー操作を待っている状態を示す場合、上述した信号記憶領域のうちスター

トレバーの操作を検出するためのセンサに対応する領域を参照し、前々回、前回、今回の割込み時における検出信号の有無のパターンが予め定められた判定パターンと一致した場合に乱数値生成回路1316の内部抽選用のチャンネルの乱数をラッチするためのラッチ信号を出力する処理（具体的には、ソフトラッチレジスタ3186の内部抽選用のチャンネルに対応する領域に所定の値を書き込む処理）を実行するとともに、このラッチ信号によってラッチされた乱数を取得するためのリード信号を出力する処理（具体的には、図77に示す乱数レジスタ3188のうちの特図1のチャンネルに対応する乱数レジスタに対してリード信号を出力する処理）を実行してラッチされている乱数を取得して対応する乱数値記憶領域に格納する。なお、本実施形態では、スタートレバーの操作があったか否かの判定として、前々回、前回および今回の検出信号が予め定められた判定パターン情報と一致した否かを判定していたが、このような方法に限らず、例えば、前回および今回の検出信号が予め定められた入賞判定パターン情報（例えば、前回検出信号なし、今回検出信号あり（いわゆるアップエッジ））と一致したか否かを判定するように構成してもよい。すなわち、スタートレバーの操作を検出するためのセンサの検出信号の有無を監視し、この監視結果の履歴が予め定められたパターンと一致したか否かを判定するように構成すればよい。

10

また、本実施形態では、乱数値生成回路1316から乱数を取得する方法として内部抽選用のチャンネルの乱数をラッチするためのラッチ信号を出力する処理を行った後にラッチした乱数を取得するリード信号を出力する方法を採用しているが、内部抽選用のチャンネルの乱数を直読みする処理（具体的には、乱数レジスタ3188の値を直接リードする処理）を実行して取得するようにしてもよい。

20

また、本実施形態では、内部抽選用の乱数を取得するためのリード信号を出力する処理（乱数レジスタ3188の値を直接リードする処理とは異なる）を主制御部タイマ割込み処理において実行するように構成しているが、この処理に関しては上述した主制御部メイン処理の内部抽選を行う直前に実行するように構成してもよい。

また、内部抽選用の乱数を取得するためのリード信号を出力する処理とともに、主制御部タイマ割込み処理において内部抽選処理を行うように構成してもよい。

ステップS2209では、タイマ更新処理を行う。各種タイマをそれぞれの時間単位により更新する。

#### 【0435】

30

ステップS2211では、コマンド設定送信処理を行い、各種のコマンドが第1副制御部1400に送信される。なお、第1副制御部1400に送信する出力予定情報は本実施形態では16ビットで構成しており、ビット15はストロブ情報（オンの場合、データをセットしていることを示す）、ビット11～14はコマンド種別（本実施形態では、基本コマンド、スタートレバー受付コマンド、演出抽選処理に伴う演出コマンド、リール1110乃至1112の回転を開始に伴う回転開始コマンド、ストップボタン1137乃至1139の操作の受け付けに伴う停止ボタン受付コマンド、リール1110乃至1112の停止処理に伴う停止位置情報コマンド、メダル払出処理に伴う払出枚数コマンド及び払出終了コマンド、遊技状態を示すコマンド等）、ビット0～10はコマンドデータ（コマンド種別に対応する所定の情報）で構成されている。第1副制御部1400では、受信した出力予定情報に含まれるコマンド種別により、主制御部1300における遊技制御の変化に応じた演出制御の決定が可能になるとともに、出力予定情報に含まれているコマンドデータの情報に基づいて、演出制御内容を決定することができるようになる。

40

#### 【0436】

ステップS2213では、外部出力信号設定処理を行う。この外部出力信号設定処理では、RAM1308に記憶している遊技情報を、情報出力回路1334を介してスロットマシン1100とは別体の情報入力回路1652に出力する。

#### 【0437】

ステップS2215では、デバイス監視処理を行う。このデバイス監視処理では、まずはステップS2205において信号状態記憶領域に記憶した各種センサ1318の信号状

50

態を読み出して、メダル投入異常及びメダル払出異常等に関するエラーの有無を監視し、エラーを検出した場合には（図示省略）エラー処理を実行させる。さらに、現在の遊技状態に応じて、メダルセクタ 1 1 7 0（メダルセクタ 1 1 7 0 内に設けたソレノイドが動作するメダルブロック）、各種ランプ 1 3 3 8、各種の 7 セグメント（S E G）表示器の設定を行う。

#### 【0 4 3 8】

ステップ S 2 2 1 7 では、低電圧信号がオンであるか否かを監視する。そして、低電圧信号がオンの場合（電源の遮断を検知した場合）にはステップ S 2 2 2 1 に進み、低電圧信号がオフの場合（電源の遮断を検知していない場合）にはステップ S 2 2 1 9 に進む。

なお、本実施形態では、電源の遮断を判定する処理（ステップ S 2 2 1 7）および電断処理（ステップ S 2 2 2 1）を主制御部の割込み制御において実行しているが、必ずしも主制御部の割込み制御で行う必要はなく、低電圧信号の入力に基づいて強制割込みを発生させ、この強制割込みによって電断処理が実行されるように構成してもよい。ただし、このように強制割込みによって実行される電断処理では、本実施形態における電断処理（ステップ S 2 2 2 1）の処理に加え、上述したタイマ割込み開始処理（ステップ S 2 2 0 1）において実行される C P U 1 3 0 4 の各レジスタの値をスタック領域に一時的に退避する処理も同時に実行する必要がある。

#### 【0 4 3 9】

ステップ S 2 2 1 9 では、タイマ割込終了処理を終了する各種処理を行う。このタイマ割込終了処理では、ステップ S 2 2 0 1 で一時的に退避した各レジスタの値を元の各レジスタに設定等行う。その後、主制御部メイン処理に復帰する。一方、ステップ S 2 2 2 1 では、復電時に電断時の状態に復帰するための特定の変数やスタックポインタを復帰データとして R A M 1 3 0 8 の所定の領域に退避し、入出力ポートの初期化等の電断処理を行い、その後、主制御部メイン処理に復帰する。

#### 【0 4 4 0】

< 第 1 副制御部 4 0 0 の処理 >

次に、図 7 2 を用いて、第 1 副制御部 1 4 0 0 の処理について説明する。なお、同図（a）は、第 1 副制御部 1 4 0 0 の C P U 1 4 0 4 が実行するメイン処理のフローチャートであり、同図（b）は、第 1 副制御部 1 4 0 0 のコマンド受信割込処理のフローチャートである。また、同図（c）は、第 1 副制御部 1 4 0 0 のタイマ割込処理のフローチャートである。

#### 【0 4 4 1】

まず、同図（a）のステップ S 2 3 0 1 では、各種の初期設定を行う。電源投入が行われると、まずステップ S 2 3 0 1 で初期化処理が実行される。この初期化処理では、入出力ポートの初期設定や、R A M 1 4 0 8 内の記憶領域の初期化処理等を行う。

#### 【0 4 4 2】

ステップ S 2 3 0 3 では、タイマ変数が 1 0 以上か否かを判定し、タイマ変数が 1 0 となるまでこの処理を繰り返し、タイマ変数が 1 0 以上となったときには、ステップ S 2 3 0 5 の処理に移行する。ステップ S 2 3 0 5 では、タイマ変数に 0 を代入する。ステップ S 2 3 0 7 では、コマンド処理を行う。コマンド処理では第 1 副制御部 1 4 0 0 の C P U 1 4 0 4 は、主制御部 1 3 0 0 からコマンドを受信したか否かを判別する。

#### 【0 4 4 3】

ステップ S 2 3 0 9 では、演出制御処理を行う。この演出制御処理では、例えば、ステップ S 2 3 0 7 で新たなコマンドがあった場合には、このコマンドに対応する処理を行う。この処理には、例えば、演出データを R O M 1 4 0 6 から読み出す等の処理を行い、演出データの更新が必要な場合には演出データの更新処理を行うことが含まれる。

#### 【0 4 4 4】

ステップ S 2 3 1 1 では、ステップ 2 3 0 9 の処理結果に基づいて音制御処理を行う。例えば、ステップ S 2 3 0 9 で読み出した演出データの中に音源 I C 1 4 1 8 への命令がある場合には、この命令を音源 I C 1 4 1 8 に出力する。ステップ S 2 3 1 3 では、ステ



ップ2309の処理結果に基づいてランプ制御処理を行う。例えば、ステップS2309で読み出した演出データの中に各種ランプ1420への命令がある場合には、この命令を駆動回路1422に出力する。

#### 【0445】

ステップS2315では、ステップ2309の処理結果に基づいて第2副制御部500に制御コマンドを送信する設定を行う情報出力処理を行う。例えば、ステップS2309で読み出した演出データの中に第2副制御部1500に送信する制御コマンドがある場合には、この制御コマンドを出力する設定を行い、ステップS2303へ戻る。

#### 【0446】

次に、同図(b)を用いて、第1副制御部1400のコマンド受信割込処理について説明する。このコマンド受信割込処理は、第1副制御部1400が、主制御部1300が出力するストロブ信号を検出した場合に実行する処理である。コマンド受信割込処理のステップS2401では、主制御部1300が出力したコマンドを未処理コマンドとしてRAM1408に設けたコマンド記憶領域に記憶する。

#### 【0447】

次に、同図(c)を用いて、第1副制御部1400のCPU1404によって実行する第1副制御部タイマ割込処理について説明する。第1副制御部1400は、所定の周期(本実施形態では2msに1回)でタイマ割込を発生するハードウェアタイマを備えており、このタイマ割込を契機として、タイマ割込処理を所定の周期で実行する。

#### 【0448】

ステップS2501では、第1副制御部メイン処理におけるステップS2303において説明したRAM1408のタイマ変数記憶領域の値に、1を加算して元のタイマ変数記憶領域に記憶する。従って、ステップS2303において、タイマ変数の値が10以上と判定されるのは20ms毎(2ms×10)となる。ステップS2503では、ステップS2315で設定された第2副制御部1500への制御コマンドの送信や、演出用乱数値の更新処理等を行う。

#### 【0449】

##### <第2副制御部500の処理>

次に、図73を用いて、第2副制御部1500の処理について説明する。なお、同図(a)は、第2副制御部1500のCPU1504が実行するメイン処理のフローチャートであり、同図(b)は、第2副制御部1500のコマンド受信割込処理のフローチャートである。また、同図(c)は、第2副制御部1500のタイマ割込処理のフローチャートであり、同図(d)は、第2副制御部1500の画像制御処理のフローチャートである。

#### 【0450】

まず、同図(a)のステップS2601では、各種の初期設定を行う。電源投入が行われると、まずステップS2601で初期化処理が実行される。この初期化処理では、入出力ポート初期設定や、RAM1508内の記憶領域の初期化処理等を行う。

#### 【0451】

ステップS2603では、タイマ変数が10以上か否かを判定し、タイマ変数が10となるまでこの処理を繰り返し、タイマ変数が10以上となったときには、ステップS2605の処理に移行する。ステップS2605では、タイマ変数に0を代入する。ステップS2607では、コマンド処理を行う。コマンド処理では第2副制御部1500のCPU1504は、第1副制御部1400のCPU1404からコマンドを受信したか否かを判別する。ステップS2609では、演出制御処理を行う。この演出制御処理では、例えば、ステップS2607で新たなコマンドがあった場合には、このコマンドに対応する処理を行う。この処理には、例えば、演出データをROM1506から読み出す等の処理を行い、演出データの更新が必要な場合には演出データの更新処理を行うことが含まれる。

#### 【0452】

ステップS2611では、ステップS2609の処理結果に基づいてシャッタ制御処理を行う。例えば、ステップS2609で読み出した演出データの中にシャッタ制御の命令

10

20

30

40

50

がある場合には、この命令に対応するシャッタ制御を行う。ステップS 2 6 1 3では、ステップS 2 6 0 9の処理結果に基づいて画像制御処理を行う。例えば、ステップS 2 6 0 9読み出した演出データの中に画像制御の命令がある場合には、この命令に対応する画像制御を行い（詳細は後述する）、ステップS 2 6 0 3へ戻る。

#### 【0 4 5 3】

次に、同図（b）を用いて、第2副制御部1 5 0 0のコマンド受信割込処理について説明する。このコマンド受信割込処理は、第2副制御部1 5 0 0が、第1副制御部1 4 0 0が出力するストロブ信号を検出した場合に実行する処理である。コマンド受信割込処理のステップS 2 7 0 1では、第1副制御部1 4 0 0が出力したコマンドを未処理コマンドとしてRAM 1 5 0 8に設けたコマンド記憶領域に記憶する。

10

#### 【0 4 5 4】

次に、同図（c）を用いて、第2副制御部1 5 0 0のCPU 1 5 0 4によって実行する第2副制御部タイマ割込処理について説明する。第2副制御部1 5 0 0は、所定の周期（本実施形態では2 m sに1回）でタイマ割込を発生するハードウェアタイマを備えており、このタイマ割込を契機として、タイマ割込処理を所定の周期で実行する。

#### 【0 4 5 5】

ステップS 2 8 0 1では、第2副制御部メイン処理におけるステップS 2 6 0 3において説明したRAM 1 5 0 8のタイマ変数記憶領域の値に、1を加算して元のタイマ変数記憶領域に記憶する。従って、ステップS 2 6 0 3において、タイマ変数の値が1 0以上と判定されるのは2 0 m s毎（2 m s×1 0）となる。ステップS 2 8 0 3では、演出用乱数値の更新処理等を行う。

20

#### 【0 4 5 6】

次に、同図（d）を用いて、第2副制御部1 5 0 0のメイン処理におけるステップS 2 6 1 3の画像制御処理について説明する。同図は、画像制御処理の流れを示すフローチャートを示した図である。

#### 【0 4 5 7】

ステップS 2 9 0 1では、画像データの転送指示を行う。ここでは、CPU 1 5 0 4は、まず、VRAM 1 5 3 6の表示領域Aと表示領域Bの描画領域の指定をスワップする。これにより、描画領域に指定されていない表示領域に記憶された1フレームの画像が演出画像表示装置1 1 5 7に表示される。次に、CPU 1 5 0 4は、VDP 1 5 3 4のアトリビュートレジスタに、位置情報等テーブルに基づいてROM座標（ROM 1 5 0 6の転送元アドレス）、VRAM座標（VRAM 1 5 3 6の転送先アドレス）などを設定した後、ROM 1 5 0 6からVRAM 1 5 3 6への画像データの転送開始を指示する命令を設定する。VDP 1 5 3 4は、アトリビュートレジスタに設定された命令に基づいて画像データをROM 1 5 0 6からVRAM 1 5 3 6に転送する。その後、VDP 1 5 3 4は、転送終了割込信号をCPU 1 5 0 4に対して出力する。

30

#### 【0 4 5 8】

ステップS 2 9 0 3では、VDP 1 5 3 4からの転送終了割込信号が入力されたか否かを判定し、転送終了割込信号が入力された場合はステップS 2 9 0 5に進み、そうでない場合は転送終了割込信号が入力されるのを待つ。ステップS 2 9 0 5では、演出シナリオ構成テーブルおよびアトリビュートデータなどに基づいて、パラメータ設定を行う。ここでは、CPU 1 5 0 4は、ステップS 2 9 0 1でVRAM 1 5 3 6に転送した画像データに基づいてVRAM 1 5 3 6の表示領域AまたはBに表示画像を形成するために、表示画像を構成する画像データの情報（VRAM 1 5 3 6の座標軸、画像サイズ、VRAM座標（配置座標）など）をVDP 1 5 3 4に指示する。VDP 1 5 3 4はアトリビュートレジスタに格納された命令に基づいてアトリビュートに従ったパラメータ設定を行う。

40

#### 【0 4 5 9】

ステップS 2 9 0 7では、描画指示を行う。この描画指示では、CPU 1 5 0 4は、VDP 1 5 3 4に画像の描画開始を指示する。VDP 1 5 3 4は、CPU 1 5 0 4の指示に従ってフレームバッファにおける画像描画を開始する。ステップS 2 9 0 9では、画像の

50

描画終了に基づくVDP1534からの生成終了割込み信号が入力されたか否かを判定し、生成終了割込み信号が入力された場合はステップS2911に進み、そうでない場合は生成終了割込み信号が入力されるのを待つ。ステップS2911では、RAM1508の所定の領域に設定され、何シーンの画像を生成したかをカウントするシーン表示カウンタをインクリメント(+1)して処理を終了する。

#### 【0460】

このようなスロットマシン1100の主制御部1300に対しても、次に説明する本発明の特徴点の一つ、複数、または全てを適用することができる。

#### 【0461】

以上説明したように、本実施形態に係るパチンコ機100（またはスロットマシン1100）は、遊技の進行を中心とした遊技制御を行うCPU（例えば、CPU304（またはCPU1304））を有する主制御部（例えば、主制御部300（または主制御部1300））と、前記主制御部の遊技制御に基づいて遊技の演出制御を行う副制御部（例えば、第1副制御部400、第2副制御部500（または、第1副制御部1400、第2副制御部1500））と、を備えた遊技台であって、前記CPUは、第一レジスタ（例えば、「POP TI命令」、「PUSH TI命令」などのロード命令で使うことができるが、演算命令では使うことができない（演算命令によって直接、自身の値を変化させることができない）Tレジスタ）と第二レジスタ（例えば、Aレジスタ、Fレジスタ、Bレジスタ、Cレジスタ、Dレジスタ、Eレジスタ、Hレジスタ、Lレジスタ、およびこれらの裏レジスタ）を備えるとともに、前記第一レジスタに前記第二レジスタを介してデータを転送する命令（例えば「LD T, A」命令のように、TレジスタにAレジスタを介して直接、データを転送する命令）を備えず、前記第一レジスタに直値（イミディエイト値や即値ともいう）によりデータを設定する命令（例えば、「LD T, F0H」命令）を備えることを特徴とする遊技台である。

#### 【0462】

本実施形態に係るパチンコ機100（またはスロットマシン1100）によれば、第一のレジスタに対して直値でデータを転送するよう構成されているので、セットしているデータを確認し易くなる場合がある。また、第一のレジスタに対して他のレジスタを介してデータを転送できないように構成されているので、第一のレジスタの値を不用意に変更することができなくなる結果、コーディングミスの発生を減少させることができる場合がある。また、第一レジスタは直値によるデータの転送のみが認められているため、レジスタを介したデータの転送と直値によるデータの転送の両方が認められている他のレジスタよりも相対的に使用頻度が低くなる。このため、コーディングの見直し作業や、デバッグ（例えば、Tレジスタのトレースなど）の効率を高めることが可能となる。

#### 【0463】

また、本実施形態に係るパチンコ機100（またはスロットマシン1100）は、CPUを有し、遊技制御を行う遊技制御手段を備えた遊技台であって、前記遊技台は、パチンコ機またはスロットマシンであり、前記CPUは、マイクロコンピュータに内蔵され、前記CPUは、少なくとも特定レジスタ（例えば、Tレジスタ）を備え、前記CPUは、前記特定レジスタに値をセットする機能のうち、ロード命令を受けたことに基づいて行われるものとしては、直値により値をセットする機能のみを有する遊技台である。

#### 【0464】

本実施形態に係るパチンコ機100（またはスロットマシン1100）によれば、ロード命令では特定レジスタに対して直値以外にデータをセットできないので、特定レジスタの値を不用意に変更することができなくなる結果、コーディングミスの発生を減少させることができ、安定した遊技制御を行うことができる。また、特定レジスタにセットしているデータをソースコード上で目視で容易に確認することができるため、コーディングの見直し作業やデバッグの効率を高めることができ、従来よりもミスの発生を大幅に減らすことができ、安定した遊技制御を行うことができる。また、特定レジスタは直値によるデータのセットのみが認められているため、レジスタを介したデータのセットと直値によるデ

ータのセットの両方が認められている他のレジスタよりも相対的に使用頻度が低くなる。このため、コーディングの見直し作業やデバッグ（例えば、Tレジスタのトレースなど）の効率を高めることができる場合がある。

#### 【0465】

なお、本発明は、「プログラミングの際にロード命令を用いない」ことを特徴とするものではなく、「CPUは、特定レジスタに値をセットする機能のうち、ロード命令を受けたことに基づいて行われるものとしては、直値により値をセットする機能のみを有する」ため、「プログラミングの際に、特定レジスタについては直値により値をセットするロード命令しか用いることができない」ことを特徴としている。

#### 【0466】

すなわち、本発明では、プログラミングの際に、たとえ特定レジスタに他のレジスタを介して値をセットするロード命令を用いようとしても、CPU自体がそのようなロード命令を実行（理解）する機能を有していないため、通常は、アセンブラ言語を機械語にアセンブルする過程で、CPUが実行（理解）できない命令として排除され、機械語に変換できないように構成される。このため、プログラマーは、プログラミングの際に、たとえ特定レジスタに他のレジスタを介して値をセットするロード命令を用いたくても、用いることができない。なお、本明細書では、アセンブリ言語を使ってプログラミングを行う例を示しているが、機械語を使ってプログラミングを行った場合も事情は同じで、プログラマーは、プログラミングの際に、たとえ特定レジスタに他のレジスタを介して値をセットするロード命令を用いたくても、CPUが実行（理解）できない以上、当該命令を用いることができない。

#### 【0467】

このような理由により、本発明では、上述の顕著な効果を得ることができるが、従来技術では、ロード命令に関して本発明のような制約がないため、プログラマーは、プログラミングの際に、特定レジスタに他のレジスタを介して値をセットするロード命令を自由に用いることができる。「プログラミングの際にロード命令を用いるか否か（記述するか否か）は、プログラムの内容等に基づく必要に応じて当業者が適宜選択し得る事項である」ため、プログラマーは、特定レジスタに他のレジスタを介して値をセットするロード命令を用いるか否かを適宜選択することができるが、従来技術では当該命令を用いることができるという選択肢がある以上、上述の本発明特有の顕著な効果と同等の効果を確実に得ることはできない。

#### 【0468】

また、仮にプログラミングの際には当該命令を用いないというルールを策定したとしても、ルールを守るか守らないかはプログラマー（人間）に依存することであり、本発明のように、特定レジスタに他のレジスタを介して値をセットするロード命令の使用を100%排除できるものでもない。

#### 【0469】

さらに、特定レジスタに他のレジスタを介して値をセットするロード命令を自由に用いることができる環境では、通常は、アセンブラ言語を機械語にアセンブルする過程で、CPUが実行（理解）できる命令として、当該命令を機械語に変換できるように構成される。このため、従来技術においては、プログラマーは、プログラミングの際に、たとえ特定レジスタに他のレジスタを介して値をセットするロード命令を用いないように注意していたとしても、アセンブラの最適化などによって当該命令に相当する機械語にアセンブルされてしまう可能性を100%排除することができない。

#### 【0470】

また、前記第二レジスタは、一の特定レジスタ（例えば、Aレジスタ）と複数の非特定レジスタ（例えば、Fレジスタ、Bレジスタ、Cレジスタ、Dレジスタ、Eレジスタ、Hレジスタ、Lレジスタ、およびこれらの裏レジスタ）で構成されており、前記CPUは、前記非特定レジスタに前記第一レジスタを介してデータを転送する命令（例えば、「LDB, T」命令のように、BレジスタにTレジスタを介してデータを転送する命令）を備

えず、前記特定レジスタに前記第一レジスタを介してデータを転送する命令（例えば、「LD A, T」命令）を備えてもよい。

#### 【0471】

このような構成とすれば、第一のレジスタを他のレジスタに比べて使いにくくすることができるため、第一のレジスタの値を不用意に変更することができなくなる結果、コーディングミスの発生を減少させることができる場合がある。

#### 【0472】

また、前記第一レジスタおよび前記第二レジスタは、8ビット長の値を記憶可能な8ビットレジスタであり、前記非特定レジスタは、他のレジスタと合わせて16ビット長の演算が可能なレジスタの組み合わせ（例えば、BCレジスタ、DEレジスタ、HLレジスタ）を構成可能なレジスタであるが、前記第一レジスタは、他のレジスタと合わせて前記16ビット長の演算が可能なレジスタの組み合わせを構成不可能なレジスタであってもよい。

10

#### 【0473】

このような構成とすれば、第一のレジスタを他のレジスタに比べて使いにくくすることができるため、第一のレジスタの値を不用意に変更することができなくなる結果、コーディングミスの発生を減少させることができる場合がある。なお、Tレジスタは、他のレジスタと合わせて16ビット長の演算が可能なレジスタの組み合わせを構成不可能であるが、上述の通り、「PUSH TI」命令や、「POP TI」命令では、Iレジスタと合わせて16ビット長の退避・復帰が可能である。換言すれば、Tレジスタは、演算命令のオペランドとしては利用できないが、転送命令のオペランドとしては利用することができるレジスタである。

20

#### 【0474】

また、前記遊技制御に用いるデータを記憶する記憶手段を備え、前記記憶手段は、前記CPUが実行するプログラムを記憶するROM（例えば、ROM306（またはROM1306））と、前記CPUが前記遊技制御を行う際に用いるデータを一時記憶可能なRAM（例えば、RAM308（またはRAM1308））と、を少なくとも有し、前記CPUは、前記遊技制御を開始する前（例えば、「電源投入後、主制御部300のCPU304がユーザプログラム（制御プログラム）を実行する前」、「システムリセット後、主制御部300のCPU304がユーザプログラム（制御プログラム）を実行する前」、「セキュリティモードに移行した直後」、「セキュリティチェックの結果がOKとなってユーザモードに移行した直後」、「システムリセットによる初期化処理の期間中」など）に、前記RAMの先頭アドレスのうちの上位アドレスを示す特定の値（例えば、F0H）を、（前記プログラムによらずに、）前記第一レジスタに設定してもよい。

30

#### 【0475】

このような構成とすれば、RAMのデータを読み書きする場合には、第一のレジスタをRAMの上位アドレスを示す値として固定することで、RAMの下位アドレスだけを指定すればよく、プログラムコード量の削減や処理速度の向上を実現することが可能となる上に、第一のレジスタの値を不用意に変更することができなくなる結果、コーディングミスの発生を減少させることができる場合がある。

40

#### 【0476】

また、CPUと、遊技制御プログラムが記憶されたROMと、データを一時記憶可能なRAMと、を内蔵するマイクロコンピュータを備えた遊技台であって、前記遊技台は、パチンコ機またはスロットマシンであり、前記CPUは、少なくとも特定のレジスタを有し、前記RAMの先頭アドレスの上位バイトは、前記ROMの先頭アドレスの上位バイトよりも大きく、前記CPUは、前記特定のレジスタに値をセットする機能のうち、ロード命令を受けたことに基づいて行われるものとしては、直値により値をセットする機能のみを有し、前記CPUは、前記特定のレジスタに値をセットする機能のうち、前記ロード命令を受けたことに基づいて行われるもの以外のものとしては、前記RAMの先頭アドレスの上位バイトと同じ値を初期値としてセットする機能を少なくとも有していてもよい。

50

## 【0477】

このような構成とすれば、RAMの上位アドレスを示す値を特定レジスタの初期値としてセットすることで、RAMのデータを読み書きする場合には、RAMの下位アドレスだけを指定すればよく、プログラムコード量の削減や処理速度の向上を実現することができる。しかも、特定レジスタには、初期値以外では、ロード命令による直値だけしかセットすることができないので、特定レジスタの値を不用意に変更することができなくなる結果、アドレスの設定ミスや、RAMへのデータの読み書きミスなどの発生を回避することができ、安定した遊技制御を行うことができる。また、頻繁に演算に用いられることにより、意図しない演算結果が記憶されたレジスタ経由で、間違った値が特定のレジスタにセットされることを防止できるとともに、初期値としてRAMの先頭アドレスの上位バイトと同じ値が特定のレジスタにセットされるので、電源投入後、特定のレジスタを用いてRAMのアクセスを行う場合であれば、特定のレジスタに値をセットする操作をせずに、特定のレジスタを用いてRAMのアクセスを行うことが可能になり、当該「RAMのアクセスを行う」ことについて、特定のレジスタの値が誤っていることから意図しないアドレスの情報をアクセスしてしまうといった不具合の防止効果をさらに高めることができる。

10

## 【0478】

また、プログラムを書き換えただけでは遊技を思い通りに制御することができないため、プログラムの書き換えによる不正行為を効果的に防止できる場合がある。特に、CPUのユーザによって設定されたユーザプログラムを実行可能なユーザモードと、該ユーザプログラムを実行不可能な非ユーザモード（例えば、セキュリティモード）と、を備え、該非ユーザモードにおいて前記特定の値を第一レジスタに設定すれば、特定の値が他の値に改ざんされたり、特定の値が読みだされたりするような不正行為を未然に防止できる場合がある。

20

## 【0479】

また、前記RAMは、第一エリア（例えば、F000H～F3FFHのワークエリア（または、I/OマップドI/O方式でアクセス可能なエリア））と第二エリア（例えば、7FE0H～7FFFHのスタックエリア（または、メモリマップドI/O方式でアクセス可能なエリア））を有して構成されており、前記特定の値は、前記第一エリアの上位アドレス（F0H）を示す値であってもよい。

## 【0480】

このような構成とすれば、RAMの第一エリアのデータを読み書きする場合には、第一のレジスタをRAMの第一エリアの上位アドレスを示す値として固定することで、RAMの第一エリアの下位アドレスだけを指定すればよく、プログラムコード量の削減や処理速度の向上を実現することが可能となる上に、第一のレジスタの値を不用意に変更することができなくなる結果、コーディングミスの発生を減少させることができる場合がある。また、第一エリアの上位アドレスを固定することによって、第一エリアの読み書きを行うつもりが誤って第二エリアの読み書きを行ってしまうようなミスを未然に防止できる場合がある。

30

## 【0481】

また、本実施形態に係るパチンコ機100（またはスロットマシン1100）は、遊技の進行を中心とした遊技制御を行うCPU（例えば、CPU304（またはCPU1304）と、前記遊技制御に用いるデータを記憶する記憶手段と、を備えた遊技台であって、前記記憶手段は、前記CPUが実行するプログラムを記憶するROM（例えば、ROM306（またはROM1306））と、前記CPUが前記遊技制御を行う際に用いるデータを一時記憶可能なRAM（例えば、RAM308（またはRAM1308））と、を少なくとも有し、前記CPUは、前記遊技制御を開始する前（例えば、「電源投入後、主制御部300のCPU304がユーザプログラム（制御プログラム）を実行する前」、「システムリセット後、主制御部300のCPU304がユーザプログラム（制御プログラム）を実行する前」、「セキュリティモードに移行した直後」、「セキュリティチェックの結果がOKとなってユーザモードに移行した直後」、「システムリセットによる初期化処理

40

50

の期間中」など)に、前記RAMの先頭アドレスのうちの上位アドレスを示す特定の値(例えば、F0H)を、(前記プログラムによらずに、)特定レジスタ(例えば、ROM306とは別体のCPU304に内蔵されたTレジスタ)に設定することを特徴とする遊技台である。

#### 【0482】

本実施形態に係るパチンコ機100(またはスロットマシン1100)によれば、RAMのデータを読み書きする場合には、特定レジスタをRAMの上位アドレスを示す値として固定することで、RAMの下位アドレスだけを指定すればよく、プログラムコード量の削減や処理速度の向上を実現することが可能となる上に、特定レジスタの値を不用意に変更することができなくなる結果、コーディングミスの発生を減少させることができる場合がある。

10

#### 【0483】

また、本実施形態に係るパチンコ機100(またはスロットマシン1100)は、CPUと、遊技制御プログラムを記憶するROMと、データを一時記憶可能なRAMと、を内蔵するマイクロコンピュータを備えた遊技台であって、前記遊技台は、パチンコ機またはスロットマシンであり、前記CPUは、少なくとも特定レジスタを備え、前記RAMの先頭アドレスの上位バイトは、前記ROMの先頭アドレスの上位バイトよりも大きく、前記遊技制御プログラムによらずに、前記RAMの先頭アドレスの上位バイトと同じ値が、初期値として前記特定レジスタにセットされることを特徴とする遊技台である。

#### 【0484】

本実施形態に係るパチンコ機100(またはスロットマシン1100)によれば、遊技制御プログラム内でRAMのアドレスの上位バイトを記述する必要がなくなるので、不正行為者にプログラムを不正に解析されたとしてもRAMのアドレスが特定困難である上に、遊技制御プログラムを書き換えただけでは遊技を思い通りに制御することができないため、遊技制御プログラムの書き換えによる不正行為を効果的に防止することができる。また、RAMのデータを読み書きする場合には、特定レジスタをRAMの上位アドレスを示す値として固定することで、RAMの下位アドレスだけを指定すればよく、プログラムコード量の削減や処理速度の向上を実現することが可能となる上に、特定レジスタの値を不用意に変更することができなくなる結果、コーディングミスの発生を減少させることができる。

20

30

#### 【0485】

また、プログラムを書き換えただけでは遊技を思い通りに制御することができないため、プログラムの書き換えによる不正行為を効果的に防止できる場合がある。特に、CPUのユーザによって設定されたユーザプログラムを実行可能なユーザモードと、該ユーザプログラムを実行不可能な非ユーザモード(例えば、セキュリティモード)と、を備え、該非ユーザモードにおいて前記特定の値を第一レジスタに設定すれば、特定の値が他の値に改ざんされたり、特定の値が読みだされたりするような不正行為を未然に防止できる場合がある。

#### 【0486】

また、前記CPUは、非特定レジスタ(例えば、Aレジスタ、Fレジスタ、Bレジスタ、Cレジスタ、Dレジスタ、Eレジスタ、Hレジスタ、Lレジスタ、およびこれらの裏レジスタ)をさらに備えるとともに、前記特定レジスタに前記非特定レジスタを介してデータを転送する命令(例えば「LD T, A」命令のように、TレジスタにAレジスタを介してデータを転送する命令)を備えず、前記特定レジスタに直値によりデータを設定する命令(例えば、「LD T, F0H」命令)を備えてもよい。

40

#### 【0487】

このような構成とすれば、特定レジスタに対して直値でデータを転送するよう構成されているので、セットしているデータを確認し易くなる場合がある。また、特定レジスタに対して他のレジスタを介してデータを転送できないように構成されているので、特定レジスタの値を不用意に変更することができなくなる結果、コーディングミスの発生を減少さ

50

せることができる場合がある。また、特定レジスタは直値によるデータの転送のみが認められているため、レジスタを介したデータの転送と直値によるデータの転送の両方が認められている他のレジスタよりも相対的に使用頻度が低くなる。このため、コーディングの見直し作業や、デバッグ（例えば、Tレジスタのトレースなど）の効率を高めることが可能となる。

#### 【0488】

また、前記CPUは、前記特定レジスタに直値によりデータを転送する命令に基づいて、前記特定レジスタに前記特定の値を転送することができるようにしてもよい。

#### 【0489】

このような構成とすれば、ループ処理などで汎用レジスタの空きがなくなった場合に臨時的に特定レジスタを使用した後に、初期値に戻すことができ、利便性を高めることができる。

#### 【0490】

また、前記CPUは、前記RAMから前記特定レジスタにデータを転送する命令に基づいて、前記特定レジスタに前記特定の値を転送（例えば、「POP T1」命令による転送）することができるように構成してもよい。

#### 【0491】

このような構成とすれば、特定レジスタを特定の処理に使用することが可能となり、特定レジスタの利便性を高めることができる場合がある。

#### 【0492】

また、前記RAMは、第一エリア（例えば、F000H～F3FFHのワークエリア（または、I/OマップドI/O方式でアクセス可能なエリア））と第二エリア（例えば、7FE0H～7FFFHのスタックエリア（または、メモリマップドI/O方式でアクセス可能なエリア））を有して構成されており、前記特定の値は、前記第一エリアの上位アドレス（F0H）を示す値であってもよい。

#### 【0493】

このような構成とすれば、RAMの第一エリアのデータを読み書きする場合には、特定レジスタをRAMの第一エリアの上位アドレスを示す値として固定することで、RAMの第一エリアの下位アドレスだけを指定すればよく、プログラムコード量の削減や処理速度の向上を実現することが可能となる上に、特定レジスタの値を不用意に変更することができなくなる結果、コーディングミスの発生を減少させることができる場合がある。また、第一エリアの上位アドレスを固定することによって、第一エリアの読み書きを行うつもりが誤って第二エリアの読み書きを行ってしまうようなミスを未然に防止できる場合がある。

#### 【0494】

また、前記CPUは、所定の割込み条件が成立したことに基づいて実行される割込み処理（例えば、主制御部メイン処理）を実行可能であるとともに、電源投入後、最初の前記割込み処理を実行する前に、前記特定レジスタに前記特定の値が設定されているかを判定する処理（例えば、Tレジチェック処理）を実行してもよい。なお、「電源投入後、最初の前記割込み処理を実行する前」の例としては、他にも、「電源投入後、主制御部メイン処理のステップS105の低電圧状態の確認前」や、「電源投入後、主制御部メイン処理のステップS109の電断前の状態に復帰する否かの確認前」や、「電源投入後、第1副制御部の立ち上りの確認前」などを挙げることができる。また、「判定する処理」の結果に応じて、上記図11に示すように、特定レジスタに特定の値を設定したり、上記図12に示すように、エラー処理を実行してもよい。

#### 【0495】

このような構成とすれば、特定レジスタの値を確認することで遊技台が正常動作をしているか否かを判断できる場合がある。

#### 【0496】

また、本実施形態に係るパチンコ機100（またはスロットマシン1100）は、複数

10

20

30

40

50



のアドレスそれぞれで示される記憶領域（例えば、ROM領域）に制御プログラムデータを含む複数種類のデータ（例えば、非使用データや参照データ）を記憶したROM（例えば、ROM306）と、前記ROMに記憶された前記制御プログラムデータに基づいて所定の周期毎に実行される割込み処理（例えば、主制御部タイマ割込み処理）を含む複数種類の遊技制御処理を実行するCPU（例えば、CPU304）と、を備え、前記ROMは、1又は複数の前記制御プログラムデータであって、前記CPUが実行する複数種類の命令それぞれに対応した命令データ（オペコード）と、1又は複数の前記制御プログラムデータであって、前記CPUが該命令を実行するために必要な補足データ（オペランド）と、を記憶し、該命令データであって、特定のアドレス（例えば、サブルーチンの先頭のアドレス）を識別可能にする特定識別情報（例えば、サブルーチンの先頭のアドレスをバイナリ形式で表現した情報）の一部である第1の識別情報を示す第1のアドレスデータ（例えば、m）および該第1のアドレスデータと異なる別データ（例えば、EXESUB命令の命令データからmを除いたデータ）で構成され、該特定のアドレスで示される記憶領域に記憶されたデータを前記CPUに読み込ませる特定命令（例えば、EXESUB命令）に対応した特定命令データ（例えば、EXESUB命令の命令データ）と、該補足データであって、該特定識別情報から該第1の識別情報を除いた情報のうちの少なくとも一部である第2の識別情報を示す第2のアドレスデータ（例えば、n）で構成され、前記CPUが該特定命令を実行するために必要な特定補足データと、を前記割込み処理で実行される命令に対応するように（例えば、ROM制御領域のうちタイマ割り込み処理の領域に対応するROM306の記憶領域（タイマ割込み処理に対応したアドレスで示される記憶領域）に）記憶していることを特徴とする遊技台である。

10

20

#### 【0497】

本実施形態に係るパチンコ機100（またはスロットマシン1100）によれば、命令データに含まれる第1のアドレスデータmを上位アドレスのデータとして定義することで、アドレス領域に対して命令データをグループ化して対応付けることができ、コーディングミスを抑制することができる場合がある。また、命令データは、呼び出すサブルーチンの先頭アドレスの一部を構成するデータを含んで構成されていることから、移動先のアドレスデータを間違ってしまった場合であっても、間違ってしまったデータが命令データに含まれていれば、制御プログラムの実行が停止される場合があるため、コーディングミスを抑制することができる場合がある。また、従来のサブルーチンを呼び出す制御命令（例えば、CALL命令）を圧縮することができ、ROM制御領域に対応するROM306の記憶領域に記憶されている従来の制御プログラムデータを圧縮することが可能となる。また、制御プログラムの圧縮に起因して従来（例えば、CALL命令）よりもサブルーチンを呼び出す制御プログラムの処理速度を向上させる（ステート数を少なくする）ことができる場合がある。

30

#### 【0498】

また、本実施形態に係るパチンコ機100（またはスロットマシン1100）は、複数のアドレスそれぞれで示される記憶領域に制御プログラムデータおよび該制御プログラムデータに基づいて参照される参照データを含む複数種類のデータを記憶したROMと、前記ROMに記憶された前記制御プログラムデータおよび前記参照データに基づいて所定の周期毎に実行される割込み処理を含む複数種類の遊技制御処理を実行するCPUと、を備え、前記ROMは、1又は複数の前記制御プログラムデータであって、前記CPUが実行する複数種類の命令それぞれに対応した命令データと、1又は複数の前記制御プログラムデータであって、前記CPUが該命令を実行するために必要な補足データと、を記憶し、前記命令データであって、特定のアドレスを識別可能にする特定識別情報の一部である第1の識別情報を示す第1のアドレスデータおよび該第1のアドレスデータと異なる別データで構成され、該特定のアドレスで示される記憶領域に記憶されたデータを前記CPUに読み込ませる特定命令に対応した特定命令データと、前記補足データであって、前記特定識別情報から前記第1の識別情報を除いた情報である第2の識別情報を示す第2のアドレスデータで構成され、前記CPUが該特定命令を実行するために必要な特定補足データと

40

50

、を前記割込み処理で実行される命令に対応するように記憶し、前記特定命令の実行により前記CPUが読み込み可能な特定の記憶領域に全ての前記制御プログラムデータを記憶し、全ての制御プログラムデータが記憶された前記特定の記憶領域とは異なる記憶領域に、前記参照データを記憶していることを特徴とする遊技台である。

#### 【0499】

本実施形態に係るパチンコ機100（またはスロットマシン1100）によれば、CPUが特定命令を実行する際に、誤って参照データを読み込んでしまい、当該参照データに基づいて意図しない遊技制御処理を行ってしまうような事態を未然に防止することができ、遊技制御の安定化を図ることができる。また、特定命令は、特定の記憶領域に記憶された全ての制御プログラムデータを読み込み可能に構成されているため、制御プログラムデータにおける呼出元と呼出先の配置（記憶場所）が制限されることがなく、制御プログラムの設計に自由度を持たせることができる。また、遊技台の分野においては、複数種類の遊技台において制御プログラムデータ（例えば、使用頻度の高いサブルーチン）を共用する一方で、各遊技台の仕様や性能に応じて参照データ（例えば、抽選値や演出データ）を異ならせることが一般的であるため、遊技台に特に好適である。

#### 【0500】

また、複数のアドレスそれぞれで示される記憶領域に制御プログラムデータおよび該制御プログラムデータに基づいて参照される参照データを含む複数種類のデータを記憶したROMと、前記ROMに記憶された前記制御プログラムデータおよび前記参照データに基づいて所定の周期毎に実行される割込み処理を含む複数種類の遊技制御処理を実行するCPUと、を備え、前記ROMは、1又は複数の前記制御プログラムデータであって、前記CPUが実行する複数種類の命令それぞれに対応した命令データと、1又は複数の前記制御プログラムデータであって、前記CPUが該命令を実行するために必要な補足データと、を記憶し、前記命令データであって、特定のアドレスを識別可能にする特定識別情報の一部である第1の識別情報を示す第1のアドレスデータおよび該第1のアドレスデータと異なる別データで構成され、該特定のアドレスで示される記憶領域に記憶されたデータを前記CPUに読み込ませる特定命令に対応した特定命令データと、前記補足データであって、前記特定識別情報から前記第1の識別情報を除いた情報である第2の識別情報を示す第2のアドレスデータで構成され、前記CPUが該特定命令を実行するために必要な特定補足データと、を前記割込み処理で実行される命令に対応するように記憶し、前記特定命令の実行により前記CPUが読み込み可能な特定の記憶領域に全ての前記制御プログラムデータを記憶し、全ての制御プログラムデータが記憶された前記特定の記憶領域とは異なる記憶領域に、前記参照データを記憶していてもよい。

#### 【0501】

このような構成とすれば、CPUが特定命令を実行する際に、誤って参照データを読み込んでしまい、当該参照データに基づいて意図しない遊技制御処理を行ってしまうような事態を未然に防止することができ、遊技制御の安定化を図ることができる。また、特定命令は、特定の記憶領域に記憶された全ての制御プログラムデータを読み込み可能に構成されているため、制御プログラムデータにおける呼出元と呼出先の配置（記憶場所）が制限されることがなく、制御プログラムの設計に自由度を持たせることができる。また、遊技台の分野においては、複数種類の遊技台において制御プログラムデータ（例えば、使用頻度の高いサブルーチン）を共用する一方で、各遊技台の仕様や性能に応じて参照データ（例えば、抽選値や演出データ）を異ならせることが一般的であるため、遊技台に特に好適である。

#### 【0502】

また、前記ROMは、少なくとも前記第1の識別情報、前記第2の識別情報の順序に定義することにより前記特定のアドレスを識別可能にする前記特定識別情報（例えば、mn）のうち、前記第1の識別情報を示す前記第1のアドレスデータおよび前記別データで構成される前記特定命令データと、該特定識別情報のうち前記第2の識別情報を示す前記第2のアドレスデータで構成される前記特定補足データと、を前記割込み処理で実行される

命令に対応するように記憶してもよい。

【0503】

このような構成とすれば、命令データは、呼び出すサブルーチンの先頭アドレスの一部を構成するデータを含んで構成されていることから、移動先のアドレスデータを間違ってしまった場合であっても、間違ってしまったデータが命令データに含まれていれば、制御プログラムの実行が停止されるため、コーディングミスを抑止することができる場合がある。

【0504】

また、前記ROMは、少なくとも前記第1のアドレスデータの一部である第1の構成データ（例えば、EXESUB命令の命令データのビット7）、前記別データの少なくとも一部である第2の構成データ（例えば、EXESUB命令の命令データのビット6～3）、前記第1のアドレスデータから該第1の構成データを除いた第3の構成データ（例えば、EXESUB命令の命令データのビット2～0）の順序で構成された前記特定命令データを前記割込み処理で実行される命令に対応するように記憶してもよい。

10

【0505】

このような構成とすれば、命令データをグループ化しつつも、きめ細かく命令データテーブルの空き領域に割り当てることが可能になる。よって、限られたハードウェア資源（例えば、1バイトで定義可能なパターン（256パターン））を有効利用することができ、命令データの圧縮によるROM制御領域に対応するROMの記憶領域に記憶されている従来の制御プログラムデータの圧縮が可能となる。

20

【0506】

また、前記第3の構成データ（例えば、EXESUB命令の命令データのビット2～0）が前記第1の構成データ（例えば、EXESUB命令の命令データのビット7）と比較してデータ容量が大きくなる、または、前記第1の構成データが前記第3の構成データと比較してデータ容量が大きくなる（例えば、3ビットに対して1ビット）ように構成された前記特定命令データを前記割込み処理で実行される命令に対応するように記憶してもよい。

【0507】

このような構成とすれば、命令データをグループ化しつつも、きめ細かく命令データテーブルの空き領域に割り当てることが可能になる。よって、限られたハードウェア資源（例えば、1バイトで定義可能なパターン（256パターン））を有効利用することができ、命令データの圧縮によるROM制御領域に対応するROMの記憶領域に記憶されている従来の制御プログラムデータの圧縮が可能となる。

30

【0508】

また、前記ROMは、前記特定命令の実行により前記CPUが読み込み可能なデータが記憶されている特定の記憶領域（例えば、0000H～0FFFHや0000H～11FFFH）に全ての前記制御プログラムデータを記憶するとともに、該特定の記憶領域から前記制御プログラムデータが記憶された制御記憶領域（例えば、ROM制御領域に対応するROM306の記憶領域）を除いた記憶領域（例えば、非使用領域に対応するROM306の記憶領域）の全てに前記遊技制御処理の実行に使用されない非使用データ（例えば、00H）を記憶するようにしてもよい。

40

【0509】

このような構成とすれば、CPU304が非使用データおよび制御プログラムデータを除くデータ（例えば、参照データや管理データ）を直接命令データとして読み込まれることがないため、補足データのコーディングミスによるCPU304の誤作動を防止できる場合がある。

【0510】

また、前記ROMは、複数のアドレスそれぞれで示される記憶領域に特定のデータ長（例えば、1バイト（8ビット））で表される複数種類のデータを記憶するものであり、前記CPUは、前記ROMの記憶された一のアドレスに示される記憶領域に記憶されたデー

50

タを一処理毎に読み込むことで前記割込み処理を含む複数種類の遊技制御処理を実行するものであり、前記ROMは、さらに、前記特定のデータ長の2以上の整数倍である第1のデータ長（例えば、2バイト（16ビット））で表現される前記特定のアドレスを識別可能な前記特定識別情報の一部である前記第1の識別情報を示す前記第1のアドレスデータおよび前記別データで構成される前記特定命令データと、該第1のデータ長よりも前記特定のデータ長の整数倍のデータ長だけ短い第2のデータ長（例えば、1バイト（8ビット））で表現される前記第2のアドレスデータで構成される前記特定補足データと、を前記割込み処理で実行される命令に対応するように記憶してもよい。

#### 【0511】

このような構成とすれば、命令データをグループ化しつつも、きめ細かく命令データテーブルの空き領域に割り当てることが可能になる。よって、限られたハードウェア資源（例えば、1バイトで定義可能なパターン（256パターン））を有効利用することができ、命令データの圧縮によるROM制御領域に対応するROMの記憶領域に記憶されている従来の制御プログラムデータの圧縮が可能となる。

#### 【0512】

また、本実施形態に係るパチンコ機100（またはスロットマシン110）は、複数の制御プログラムデータで構成される制御プログラムを記憶しているROM（例えば、ROM306）と、前記制御プログラムに基づいて遊技制御を実行するCPU（例えば、CPU304）と、を備え、前記ROMは、前記複数の制御プログラムデータのうち、第1のデータ（例えば、EXESUB命令（EXESUBmn）の命令データにおける、ビット6およびビット7）、第2のデータ（例えば、EXESUB命令（EXESUBmn）の命令データにおける、ビット5（ $\alpha$ ））、第3のデータ（例えば、EXESUB命令（EXESUBmn）の命令データにおける、ビット3およびビット4）、第4のデータ（例えば、EXESUB命令（EXESUBmn）の命令データにおける、ビット0（ $\delta$ ）、ビット1（ $\gamma$ ）、ビット2（ $\beta$ ））、第5のデータ（例えば、EXESUB命令（EXESUBmn）の命令データにおける、補足データ（n））を含んで構成された特定の制御プログラムデータ（EXESUB命令（EXESUBmn）の命令データを、少なくとも、該第2のデータ、該第3のデータ、該第4のデータ、順番に並べて（例えば、アドレスの若い順番）記憶するものであり、前記CPUは、前記特定の制御プログラムデータを読み込んだ場合に、前記第1および第3のデータを用いて特定される第1の命令（例えば、サブルーチンの先頭のアドレスを呼び出す命令）と、前記第2のデータ、前記第4のデータおよび前記第5のデータを用いて特定される第2の命令（例えば、呼び出すサブルーチンの先頭アドレスを特定する命令）と、で特定される特定命令（例えば、EXESUB命令）を所定周期で実行される割込み処理（例えば、タイマ割込み処理）中に実行することを特徴とする遊技台とも言える。

#### 【0513】

なお、前記ROMは、前記第1のデータと前記第2のデータの間に、前記第2のデータと前記第3のデータの間に、前記第3のデータと前記第4のデータの間のうち少なくともいずれか1つのデータ間に別のデータを含んで構成される前記特定の制御プログラムデータを記憶するものであってもよいが、前記ROMは、前記第1のデータ、前記第2のデータ、前記第3のデータおよび前記第4のデータでのみで構成される前記特定の制御プログラムデータを記憶するものであると、よりアセンブル後の機械語のチェックが簡便になり、コーディングミスを抑制することができる。

#### 【0514】

また、前記第1のデータ、前記第2のデータ、第3のデータおよび前記第4のデータを合わせたデータ長と、前記第4のデータのデータ長を同一のデータ長（例えば、1バイト（8ビット）長）としてもよい。さらに、前記CPUが一度に読み込むデータのデータ長と前記第4のデータのデータ長を同一のデータ長としても良い。

#### 【0515】

また、本実施形態に係るパチンコ機100（またはスロットマシン1100）は、プロ

10

20

30

40

50

グラムデータおよび一時的なデータを記憶する記憶手段（例えば、ROM 306、RAM 308（またはROM 1306、RAM 1308））と、前記記憶手段に記憶されているデータを用いて遊技の進行を中心とした遊技制御を行う主制御部（例えば、主制御部 300（または、主制御部 1300））と、前記主制御部の遊技制御に基づいて遊技の演出制御を行う副制御部（例えば、第1副制御部 400、第2副制御部 500（または、第1副制御部 1400、第2副制御部 1500））と、を備えた遊技台であって、前記主制御部は、第一のビット長（例えば、8ビット長）からなる複数のレジスタ（例えば、Aレジスタ、Fレジスタ、Bレジスタ、Cレジスタ、Dレジスタ、Eレジスタ、Hレジスタ、Lレジスタ、Tレジスタ）を有し、前記第一のビット長の倍のビット長である第二のビット長（例えば、16ビット長）の演算が可能なレジスタの組み合わせである前記複数のレジスタ（例えば、BCレジスタ、DEレジスタ、HLレジスタ）のうちの第一のレジスタと第二のレジスタ（例えば、HレジスタとLレジスタ）に格納された値によって示される前記記憶手段における第一のアドレス（例えば、1750H）、および該第一のアドレスと連続する第二のアドレス（例えば、1751H）に格納された第一の値（例えば、AEH）と第二の値（例えば、19）を、前記複数のレジスタのうちの第三のレジスタ（例えば、Cレジスタ）と第四のレジスタ（例えば、Aレジスタ）にそれぞれ転送する所定の命令（例えば、「INC TENSOU AC, (HL)」命令）を備え、該所定の命令を前記遊技制御（例えば、特別図柄変動時間抽選処理）において実行し、前記第三のレジスタと前記第四のレジスタは、前記第二のビット長の演算が可能なレジスタの組み合わせ（BCレジスタ、DEレジスタ、HLレジスタ）とは異なることを特徴とする遊技台である。なお、

「第一のアドレスと連続する第二のアドレス」は第一のアドレスよりも大きなアドレスに限定されず、第一のアドレスよりも小さなアドレスでもよく、先の例では、第一アドレスが1750H、第二アドレスが174FHでもよい。

#### 【0516】

本実施形態に係るパチンコ機100（またはスロットマシン1100）によれば、1回の命令によって、連続して配置されたアドレスに格納された2つのデータを、レジスタペアを構成しない2種類のレジスタに格納することができるため、プログラム容量を削減することができ、遊技性を向上させるためのプログラムに容量を割り当てることができ、遊技の興趣を高めることが可能となる。また、処理時間を短縮でき、安定した遊技制御を行うことができる場合がある。また、プログラム容量を削減することでプログラムが見やすくなり、コーディングミスやバグの発生を抑制できる場合がある。また、第三のレジスタと第四のレジスタとはレジスタペアではないため、16ビット演算処理を行う場合にプログラムを注意して作成することができ、コーディングミスやバグの発生を抑制できる場合がある。

#### 【0517】

また、本実施形態に係るパチンコ機100（またはスロットマシン1100）は、少なくとも第一から第四のレジスタを備え、上位レジスタおよび下位レジスタからなるレジスタペアを用いた処理を実行可能であるCPUと、遊技制御プログラムを記憶するROMと、を内蔵するマイクロコンピュータを備えた遊技台であって、前記遊技台は、パチンコ機またはスロットマシンであり、前記CPUは、第一の命令と第二の命令を含む複数の命令を実行可能であり、少なくとも前記第一のレジスタと前記第二のレジスタの二つのレジスタの組み合わせは、前記レジスタペアのうちの所定のレジスタペアであり、前記第三のレジスタと前記第四のレジスタの少なくともいずれかは、前記上位レジスタであり、前記第一の命令が実行されたことに基づいて、前記所定のレジスタペアに格納された値によって示される第一のアドレスが示す領域に格納された第一の値が前記第四のレジスタにセットされ、該第一のアドレスと連続する第二のアドレスが示す領域に格納された第二の値が前記第三のレジスタにセットされ、該所定のレジスタペアの値が該第二のアドレスと連続するアドレスを示す値になり、前記第二の命令が実行されたことに基づいて、前記所定のレジスタペアに格納された値によって示される第一のアドレスと連続する第二のアドレスが示す領域に格納された第二の値が前記第四のレジスタにセットされ、該第二のアドレスと

連続する第三のアドレスが示す領域に格納された第三の値が前記第三のレジスタにセットされ、該所定のレジスタペアの値が該第三のアドレスを示す値になり、前記CPUは、前記レジスタペアとは異なる前記第三のレジスタと前記第四のレジスタの二つのレジスタの組み合わせに前記ROMに格納された値を一命令でセットすることが可能であるとともに、前記所定のレジスタペアの値を変化させる所定の機能としては、前記第一の命令を受けたことに基づいて値をセットする第一の機能と前記第二の命令を受けたことに基づいて値をセットする第二の機能の二つのみを有することを特徴とする遊技台である。

#### 【0518】

本実施形態に係るパチンコ機100（またはスロットマシン1100）によれば、レジスタペアよりも汎用性の低い第三のレジスタと第四のレジスタの二つの組み合わせに値を一命令でセットする機能を限定したため、遊技者の利益に係る値を不正に改ざんする不正行為を不正者が行い難くすることができる。また、第一の命令と第二の命令を前後の処理に応じて使い分けることができ、遊技制御プログラムの容量を削減することができる。また、第一の命令と第二の命令とを従来の命令で構成した場合と比較して遊技制御プログラムの容量を削減でき、遊技性を向上させるためのプログラムに容量を割り当てることができる。また、第三のレジスタと第四のレジスタはレジスタペアよりも汎用性が低いため、プログラムを注意して作成することができ、コーディングミスやバグの発生を抑制できる。また、命令の実行中に割込み処理が発生し、割込み処理によって命令実行途中のレジスタの値が意図しない値に書き換えられてしまったり、命令の実行が妨げられて処理が遅延してしまうといったことがなく、安定した遊技制御を行うことができる場合がある。

#### 【0519】

また、少なくとも第一から第四のレジスタを備え、上位レジスタおよび下位レジスタからなるレジスタペアを用いた処理を実行可能であるCPUと、遊技制御プログラムを記憶するROMと、を内蔵するマイクロコンピュータを備えた遊技台であって、前記遊技台は、パチンコ機またはスロットマシンであり、前記CPUは、第一の命令と第二の命令を含む複数の命令を実行可能であり、少なくとも前記第一のレジスタと前記第二のレジスタの二つのレジスタの組み合わせは、前記レジスタペアのうちの所定のレジスタペアであり、前記第三のレジスタと前記第四のレジスタの少なくともいずれかは、前記上位レジスタであり、前記第一の命令が実行されたことに基づいて、前記所定のレジスタペアに格納された値によって示される第一のアドレスが示す領域に格納された第一の値が前記第四のレジスタにセットされ、該第一のアドレスと連続する第二のアドレスが示す領域に格納された第二の値が前記第三のレジスタにセットされ、該所定のレジスタペアの値が該第二のアドレスと連続するアドレスを示す値になり、前記第二の命令が実行されたことに基づいて、前記所定のレジスタペアに格納された値によって示される第一のアドレスと連続する第二のアドレスが示す領域に格納された第二の値が前記第四のレジスタにセットされ、該第二のアドレスと連続する第三のアドレスが示す領域に格納された第三の値が前記第三のレジスタにセットされ、該所定のレジスタペアの値が該第三のアドレスを示す値になり、前記CPUは、前記レジスタペアとは異なる前記第三のレジスタと前記第四のレジスタの二つのレジスタの組み合わせに前記ROMに格納された値を一命令でセットすることが可能であるとともに、前記所定のレジスタペアの値を変化させる所定の機能としては、前記第一の命令を受けたことに基づいて値をセットする第一の機能と前記第二の命令を受けたことに基づいて値をセットする第二の機能の二つを少なくとも有し、前記CPUは、少なくとも特定レジスタを備え、前記CPUは、前記特定レジスタに値をセットする機能のうち、ロード命令を受けたことに基づいて行われるものとしては、直値により値をセットする機能のみを有していてもよい。

#### 【0520】

このような構成とすれば、レジスタペアよりも汎用性の低い第三のレジスタと第四のレジスタの二つの組み合わせに値を一命令でセットする機能を限定したため、遊技者の利益に係る値を不正に改ざんする不正行為を不正者が行い難くすることができる。また、第一

の命令と第二の命令を前後の処理に応じて使い分けることができ、遊技制御プログラムの容量を削減することができる。また、第一の命令と第二の命令とを従来の命令で構成した場合と比較して遊技制御プログラムの容量を削減でき、遊技性を向上させるためのプログラムに容量を割り当てることができ、遊技の興趣を高めることができる。また、第三のレジスタと第四のレジスタはレジスタペアよりも汎用性が低いため、プログラムを注意して作成することができ、コーディングミスやバグの発生を抑制できる。また、命令の実行中に割込み処理が発生し、割込み処理によって命令実行途中のレジスタの値が意図しない値に書き換えられてしまったり、命令の実行が妨げられて処理が遅延してしまうといったことがなく、安定した遊技制御を行うことができる。

#### 【0521】

10

また、前記第三のレジスタと前記第四のレジスタの少なくとも何れかは、前記第二のビット長の演算が可能なレジスタの組み合わせを構成してもよい（例えば、Bレジスタは、16ビット長の演算が可能なレジスタペアであるBCレジスタを構成する）。

#### 【0522】

このような構成とすれば、16ビット演算を行うレジスタペアの一方のレジスタの値を固定にすることができ、所定の命令前に設定した一方のレジスタの値を退避することなく所定の命令後の処理で使うことができプログラム容量を削減することができる場合がある。また、16ビット演算処理を行う場合にプログラムを注意して作成することができ、コーディングミスやバグの発生を抑えることができる場合がある。

#### 【0523】

20

また、前記第三のレジスタと前記第四のレジスタの少なくとも何れかは、前記第二のビット長の演算が可能なレジスタの組み合わせの上位レジスタであってもよい（例えば、Bレジスタは、16ビット長の演算が可能なレジスタペアであるBCレジスタの上位レジスタである）。

#### 【0524】

このような構成とすれば、16ビット演算を行うレジスタペアの上位8ビットの値を変更することができ、レジスタペアで示されるアドレスの上位を変更して所定の命令後の処理で使うことができる場合がある。例えば、第三のレジスタと第四のレジスタの各々を違う処理に利用（例えば、一方をINCTENSOU命令で利用し、他方をループ処理の回数をカウントするためのカウンタとして利用）することができ、数の限られたレジスタを有効に活用することができる。また、16ビット演算処理を行う場合にプログラムを注意して作成することができ、コーディングミスやバグの発生を抑えることができる場合がある。

30

#### 【0525】

また、前記所定の命令は、前記第一のレジスタと前記第二のレジスタのうちの下位レジスタの値に所定値加算する処理（例えば、HLレジスタに2を加算する処理）を含んでいてもよい。

#### 【0526】

このような構成とすれば、所定の命令終了後に所定のアドレスから所定値加算されたアドレスの値をレジスタに設定した状態で次の処理を行うことができ、例えば、アドレスを1加算する命令を所定値回実行するようなプログラムに比べ、プログラム容量を削減できる場合がある。

40

#### 【0527】

また、前記第一のアドレスは、前記所定の命令を実行する前の前記第一のレジスタと前記第二のレジスタに格納された値によって示されるアドレス（例えば、174FH）の次のアドレス（例えば、1750H）であってもよい。

#### 【0528】

このような構成とすれば、繰り返し処理の際に最初にアドレスを指定することができ、繰り返し処理を抜けた際には所定の命令前のアドレスから所定値加算されたアドレスの値をレジスタに設定した状態で次の処理を行うことができる。特に、2種類のデータを交互

50

に記憶したデータテーブルを利用する場合には、データテーブルから所望のデータを素早く読み出すことが可能となる。

#### 【0529】

また、本実施形態に係るパチンコ機100（またはスロットマシン1100）は、プログラムデータおよび一時的なデータを記憶する記憶手段（例えば、ROM306、RAM308（またはROM1306、RAM1308））と、前記記憶手段に記憶されているデータを用いて遊技の進行を中心とした遊技制御を行う主制御部（例えば、主制御部300（または、主制御部1300））と、前記主制御部の遊技制御に基づいて遊技の演出制御を行う副制御部（例えば、第1副制御部400、第2副制御部500（または、第1副制御部1400、第2副制御部1500））と、を備えた遊技台であって、前記主制御部は、第一のビット長（例えば、8ビット長）の値を記憶可能な第一のレジスタ（例えば、Cレジスタ）および前記第一のビット長の整数倍の第二のビット長（例えば、16ビット長）の値を記憶可能な第二のレジスタ（例えば、HLレジスタ）を含む複数のレジスタ（例えば、BCレジスタ、DEレジスタ、HLレジスタ）を有し、前記第二のレジスタの値に前記第一のレジスタの値を加算して前記第二のレジスタに格納する所定の命令（例えば、「ADDTWOONE HL, C」命令）を備え、該所定の命令を前記遊技制御（例えば、特別図柄変動時間抽選処理）において実行することを特徴とする遊技台である。なお、第二のレジスタは、第一のビット長の整数倍の第二のビット長の値を記憶可能であればよく、例えば、第一のレジスタのビット長が4ビットである場合には4の倍数（4、8、12、…）をビット長とするレジスタでもよく、第一のレジスタのビット長が16ビットである場合には16の倍数（16、32、64、…）をビット長とするレジスタでもよい。

10

20

#### 【0530】

本実施形態に係るパチンコ機100（またはスロットマシン1100）によれば、桁上がり処理を別途行う必要がないため、プログラム容量を削減することができ、遊技性を向上させるためのプログラムに容量を割り当てることができ、遊技の興趣を高めることが可能となる。また、処理時間を短縮でき、安定した遊技制御を行うことができる場合がある。また、プログラム容量を削減することでプログラムが見やすくなり、コーディングミスやバグの発生を抑制できる場合がある。さらに、ビット長が異なるレジスタ同士の加算演算を行うことができるため、例えば、従来は16ビット長のレジスタペアを2つ使用して行っていた加算演算を、16ビット長のレジスタペアと8ビット長のレジスタで行うことができる場合があり、数の限られたレジスタを有効に活用することができる。

30

#### 【0531】

また、本実施形態に係るパチンコ機100（またはスロットマシン1100）は、8ビット長の値を記憶可能な第一から第三のレジスタを少なくとも備えるCPUを備え、前記CPUは、16ビット長の値を記憶可能な第四のレジスタを少なくとも備え、前記CPUは、前記第一のレジスタおよび前記第二のレジスタからなるレジスタペアを用いた処理を実行可能であり、前記CPUを内蔵するマイクロコンピュータを備えた遊技台であって、前記遊技台は、パチンコ機またはスロットマシンであり、前記CPUは、前記第四のレジスタの値に前記第三のレジスタの値を加算して該第四のレジスタに加算結果をセットすることが可能な第一の命令を実行可能であり、前記第一の命令は、前記レジスタペアの値に前記第三のレジスタの値を加算して該レジスタペアに加算結果をセットすることが可能であり、前記CPUは、前記レジスタペアの値から前記第三のレジスタの値を減算して該レジスタペアに減算結果をセットする第二の命令を実行可能であり、前記CPUにおける前記16ビット長の前記レジスタペアに対して前記8ビット長の値を加減算する機能は、前記第一の命令および前記第二の命令の両方によって実現されるが、前記CPUにおける前記16ビット長の前記第四のレジスタに対して前記8ビット長の値を加減算する機能は、前記第一の命令によってのみ実現されることを特徴とする遊技台である。

40

#### 【0532】

本実施形態に係るパチンコ機100（またはスロットマシン1100）によれば、ビッ

50



ト長が異なるレジスタの値同士の加減算を行うことができるため、例えば、従来は16ビット長のレジスタペアを2つ使用して行っていた加減算を、16ビット長のレジスタペアと8ビット長のレジスタで行うことができる場合があり、数の限られたレジスタを有効に活用することができる。しかも、レジスタの値同士で加減算を行うため、RAMなどを用いた場合に比べて加減算の処理時間を短縮でき、安定した遊技制御を行うことができる。また、第四のレジスタを用いた加減算を第一の命令のみで実行可能に構成することにより、遊技者の利益に係る値を不正に改ざんする不正行為（例えば、レジスタペアに対して有利な数値を加算するような命令を不正に埋め込むような行為）を不正者が行い難くすることができる。また、第一の命令と第二の命令とを従来の命令で構成した場合と比較して遊技制御プログラムの容量を削減できるため、遊技性を向上させるためのプログラムに容量を割り当てることができ、遊技の興趣を高めることができる場合がある。

10

#### 【0533】

また、少なくとも第一から第四のレジスタを備え、上位レジスタおよび下位レジスタからなるレジスタペアを用いた処理を実行可能であるCPUと、遊技制御プログラムを記憶するROMと、を内蔵するマイクロコンピュータを備えた遊技台であって、前記遊技台は、パチンコ機またはスロットマシンであり、前記CPUは、第一の命令と第二の命令を含む複数の命令を実行可能であり、少なくとも前記第一のレジスタと前記第二のレジスタの二つのレジスタの組み合わせは、前記レジスタペアのうちの所定のレジスタペアであり、前記第三のレジスタと前記第四のレジスタの少なくともいずれかは、前記上位レジスタであり、前記第一の命令が実行されたことに基づいて、前記所定のレジスタペアに格納された値によって示される第一のアドレスが示す領域に格納された第一の値が前記第四のレジスタにセットされ、該第一のアドレスと連続する第二のアドレスが示す領域に格納された第二の値が前記第三のレジスタにセットされ、該所定のレジスタペアの値が該第二のアドレスと連続するアドレスを示す値になり、前記第二の命令が実行されたことに基づいて、前記所定のレジスタペアに格納された値によって示される第一のアドレスと連続する第二のアドレスが示す領域に格納された第二の値が前記第四のレジスタにセットされ、該第二のアドレスと連続する第三のアドレスが示す領域に格納された第三の値が前記第三のレジスタにセットされ、該所定のレジスタペアの値が該第三のアドレスを示す値になり、前記CPUは、前記レジスタペアとは異なる前記第三のレジスタと前記第四のレジスタの二つのレジスタの組み合わせに前記ROMに格納された値を一命令でセットすることが可能であるととも、前記所定のレジスタペアの値を変化させる所定の機能としては、前記第一の命令を受けたことに基づいて値をセットする第一の機能と前記第二の命令を受けたことに基づいて値をセットする第二の機能の二つを少なくとも有し、前記CPUは、少なくとも特定レジスタを備え、前記CPUは、前記特定レジスタに値をセットする機能のうち、ロード命令を受けたことに基づいて行われるものとしては、直値により値をセットする機能のみを有していてもよい。

20

30

#### 【0534】

このような構成とすれば、レジスタペアよりも汎用性の低い第三のレジスタと第四のレジスタの二つの組み合わせに値を一命令でセットする機能を限定したため、遊技者の利益に係る値を不正に改ざんする不正行為を不正者が行い難くすることができる。また、第一の命令と第二の命令を前後の処理に応じて使い分けることができ、遊技制御プログラムの容量を削減することができる。また、第一の命令と第二の命令とを従来の命令で構成した場合と比較して遊技制御プログラムの容量を削減でき、遊技性を向上させるためのプログラムに容量を割り当てることができ、遊技の興趣を高めることができる。また、第三のレジスタと第四のレジスタはレジスタペアよりも汎用性が低いため、プログラムを注意して作成することができ、コーディングミスやバグの発生を抑制できる。また、命令の実行中に割込み処理が発生し、割込み処理によって命令実行途中のレジスタの値が意図しない値に書き換えられてしまったり、命令の実行が妨げられて処理が遅延してしまうといったことがなく、安定した遊技制御を行うことができる。

40

#### 【0535】

50

また、前記第一のレジスタ（例えば、Cレジスタ）は、第二のビット長の演算が可能なレジスタ（例えば、BCレジスタ）の組み合わせの下位レジスタであってもよい。

#### 【0536】

このような構成とすれば、レジスタペアの上位8ビットの値を固定にすることができ、16ビット演算処理を行う場合にプログラムを注意して作成することができ、コーディングミスやバグの発生を抑制できる場合がある。また、レジスタペアの上位レジスタと下位レジスタの各々を違う処理に利用（例えば、一方をADDTWOONE命令で利用し、他方をループ処理の回数をカウントするためのカウンタとして利用）することができ、数の限られたレジスタを有効に活用することができる。

#### 【0537】

また、前記主制御部は、前記第二のレジスタ（例えば、HLレジスタ）に格納された値によって示される所定のアドレスに基づく連続する2つのアドレスに格納された第一の値と第二の値を、前記複数のレジスタのうちの前記第一のレジスタ（例えば、Cレジスタ）と第三のレジスタ（例えば、Bレジスタ）にそれぞれ転送する所定の第二の命令（例えば、「INCTENSOU BC, (HL)」命令）を備え、該所定の第二の命令を実行した後、前記所定の命令を行ってもよい。

#### 【0538】

このような構成とすれば、連続する2つのアドレスに格納された値の何れかを第一レジスタに加算することで、所定のアドレスに関連したアドレスを示す値を第一レジスタに設定した状態で次の処理を行うことができ、プログラム容量を削減できる場合がある。

#### 【0539】

また、前記主制御部は、前記所定の第二の命令を実行した後に所定の判定処理（例えば、上記ステップS1110～S1111に示すような、Bレジスタ（第三のレジスタ）から特定の値（RAMの所定記憶領域に記憶された値など）を減算してキャリーが発生するか否かを判定する処理）を行い、該所定の判定処理の結果が所定の値のときに（例えば、キャリーが発生したときに）前記所定の命令を行ってもよい。

#### 【0540】

このような構成とすれば、所定の判定処理の結果が所定の値のときに、所定のアドレスに関連したアドレスを示す値を第一レジスタに設定した状態で次の処理を行うことができ、プログラム容量を削減できる場合がある。

#### 【0541】

また、前記主制御部は、前記所定の命令を行った後、再度前記所定の第二の命令と前記所定の判定処理を実行してもよい（例えば、上記ステップS1109でINCTENSOU命令を実行し、上記ステップS1112でADDTWOONE命令を実行した後に、上記ステップS1114でINCTENSOU命令を再度実行し、上記ステップS1116でBレジスタから特定の値を減算してキャリーが発生するか否かを判定する）。

#### 【0542】

このような構成とすれば、例えば、2段階抽選の2回目のテーブルアドレスを設定することができ、2段階抽選のプログラムを見やすく作ることができ、コーディングミスやバグの発生を抑制できる場合がある。

#### 【0543】

また、本実施形態に係るパチンコ機100（またはスロットマシン1100）は、遊技制御を行うCPUを有する主制御部（例えば、主制御部300（または、主制御部1300））と、前記主制御部の遊技制御に基づいて遊技の演出制御を行う副制御部（例えば、第1副制御部400、第2副制御部500（または、第1副制御部1400、第2副制御部1500））と、を備えた遊技台であって、前記CPUは、所定のフラグ（例えば、Zフラグ、SZフラグ）を備え、前記主制御部は、第一の処理と第二の処理の間に実行される処理であって、前記第一の処理において第一の内容であった前記フラグの内容を、前記第二の処理の開始前に第二の内容に変化させる第三の処理を実行し、前記第二の処理は、前記第三の処理の実行結果が所定の結果である場合に行われる処理であって、前記第二の

10

20

30

40

50

内容に変化した前記フラグを参照しない処理（例えば、C P R T 命令、C P J R 命令）であることを特徴とする遊技台である。

【0544】

本実施形態に係るパチンコ機100（またはスロットマシン1100）によれば、フラグの内容の変化とプログラムの実行の流れの関連性を無くすことができ、プログラムの解析を困難とし、不正なプログラムの改ざんを未然に防止できる場合がある。このため、安定した遊技制御をおこなうことができ、遊技の公平性を担保することができる場合がある。

【0545】

また、前記第三の処理は、前記第一の内容であった前記フラグを参照しない処理であってもよい。

10

【0546】

このような構成とすれば、フラグの内容の変化とプログラムの実行の流れの関連性を無くすことができ、プログラムの解析を困難とし、不正なプログラムの改ざんを未然に防止できる場合がある。このため、安定した遊技制御をおこなうことができ、遊技の公平性を担保することができる場合がある。

【0547】

また、本実施形態に係るパチンコ機100（またはスロットマシン1100）は、遊技制御を行うC P Uを有する主制御部（例えば、主制御部300（または、主制御部1300））と、前記主制御部の遊技制御に基づいて遊技の演出制御を行う副制御部（例えば、第1副制御部400、第2副制御部500（または、第1副制御部1400、第2副制御部1500））と、を備えた遊技台であって、前記C P Uは、記憶手段（例えば、R O M 306、R A M 308、レジスタ、プログラムカウンタ）の所定領域へのアクセスが規制された命令（例えば、R E S 命令、R S T 命令）を備え、前記命令を前記遊技制御において実行することを特徴とする遊技台である。

20

【0548】

本実施形態に係るパチンコ機100（またはスロットマシン1100）によれば、所定領域のアクセスを禁止することができるため、所定領域を保護することができ、所定領域の内容が予期しない数値に変化して不具合を起こすような事態を未然に防止できる場合がある。

30

【0549】

また、前記命令は、前記所定領域を除いた領域のうち予め定められた特定領域へのアクセスが可能な命令であってもよい。

【0550】

このような構成とすれば、所定領域のアクセスを禁止しつつ、同時に特定領域の内容を読み書きすることができるため、所定領域の内容を保護することができるとともに、プログラム設計の自由度を低下させることがない。

【0551】

また、前記C P Uは、前記所定領域へのアクセスが可能な命令をさらに備えてもよい。このような構成とすれば、所定領域のアクセスを禁止しつつ、同時に特定領域の内容を読み書きすることができるため、所定領域の内容を保護することができるとともに、プログラム設計の自由度を低下させることがない。

40

【0552】

また、前記命令を、前記記憶手段の所定領域にジャンプすることができないように構成すれば、不正プログラムを組み込んでユーザプログラムの先頭アドレスにジャンプさせる（ユーザプログラムを先頭アドレスから開始させて不正に初期状態とする）ような行為を未然に防止できる場合がある。

【0553】

なお、本発明に係る遊技台は、上述のパチンコ機100やスロットマシン1100に限定されない。したがって、例えば、上記実施形態では、リトルエンディアンのC P Uの例

50

を示したが、ビッグエンディアンのCPUに適用することもできる。また、8ビットCPUに限定されず、16ビットCPU、32ビットCPU等にも適用できる。また、命令の名称は上記実施形態で示した名称に限定されるものではない。

#### 【0554】

また、主制御部300のCPU304（または、主制御部1300のCPU1304）に適用した例を示したが、他の制御部のCPUに適用してもよい。また、記憶手段は、ROM、RAMに限定されず、他の記憶手段を適用してもよい。

#### 【0555】

また、図74（a）に示す、「紙幣投入口20002に紙幣を投入し、ベット20004およびスタート20006操作に基づいて抽選を実行し、抽選結果を抽選結果表示装置20008で表示し、当選時には特典コイン数を残クレジット数に加算し、キャッシュアウト20009が選択された場合には、レシート発行機20010から残クレジット数に対応するコードが記載されたレシートを発行するカジノマシン20000」に本発明を適用してもよい。また、アレンジボール遊技機や、じゃん球遊技機、スマートボール等に適用してもよい。

#### 【0556】

また、同図（b）に示すように、本発明の構成を含む制御部を備えている携帯電話機20100、同図（c）に示すように、本発明の構成を含む制御部を備えているポータブルゲーム機20200、本発明の構成を含む制御部を備えている家庭用テレビゲーム機20300、に本発明を適用してもよい。

#### 【0557】

より具体的には、同図（b）における携帯電話機20100は、遊技者によって操作される操作部と、ゲームに関するデータを携帯電話回線を通して取得するデータ取得部と、取得したゲームに関するデータを記憶する記憶部と、記憶部に記憶したデータと操作部の操作とに基づいてゲームの制御を行う、本発明の構成を含む制御部と、を備えている。

#### 【0558】

また、同図（c）におけるポータブルゲーム機20200は、遊技者によって操作される操作部と、ゲームに関するデータを所定の記憶媒体（DVD等）から取得するデータ取得部と、取得したゲームに関するデータを記憶する記憶部と、記憶部に記憶したデータと操作部の操作とに基づいてゲームの制御を行う、本発明の構成を含む制御部と、を備えている。

#### 【0559】

また、同図（c）における家庭用テレビゲーム機20300は、遊技者によって操作される操作部と、ゲームに関するデータを所定の記憶媒体（DVD等）から取得するデータ取得部と、取得したゲームに関するデータを記憶する記憶部と、記憶部に記憶したデータと操作部の操作とに基づいてゲームの制御を行う、本発明の構成を含む制御部を備えている。

#### 【0560】

さらには、同図（d）に示すように、本発明の構成を含む制御部を備えるデータサーバ20400に適用してもよい。このデータサーバ20400からインターネット回線を介して同図（d）に示す家庭用テレビゲーム機20300に本発明の構成を含むプログラムコードをダウンロードするような場合がある。また、パチンコ機等の実機の動作を家庭用ゲーム機用として擬似的に実行するようなゲームプログラムにおいても、本発明を適用することができる。

#### 【0561】

また、本発明に係る遊技台は、封入式遊技機に適用することもできる。また、主制御部、第1副制御部、および第2副制御部をワンチップで構成してもよいし、主制御部と第1副制御部で双方向の通信が可能に構成してもよい。また、主制御部と第1副制御部で双方向の通信を可能とする一方で、第1副制御部から第2副制御部への通信は一方向の通信としてもよい。

10

20

30

40

50

## 【0562】

## &lt;実施形態3&gt;

次に、図面を用いて、本発明の実施形態3に係るパチンコ機について詳細に説明する。

## 【0563】

## &lt;主制御部の基本回路の構成例&gt;

次に、図75を用いて、主制御部300の基本回路302の構成例について説明する。同図は、マイクロプロセッサ3000を用いた場合の基本回路302の一構成例を示す図である。

## 【0564】

図75に示すマイクロプロセッサ3000には、本発明にいう電氣的制御手段の一例に相当し、図4に示す基本回路302の機能と、図4に示す乱数生成回路318の機能を有するものである。なお、図75では、図4に示す構成に対応する部分を、図4で用いた符号と同じ符号を付して説明する。

## 【0565】

図75に示すマイクロプロセッサ3000には、CPU304、ROM（内蔵ROM）306、RAM（内蔵RAM）308、外部バス制御回路3110、パラレル入力ポート3102、アドレスデコード回路3103、タイマ回路311、カウンタ回路312、リセット制御回路314に加え、割込み制御回路3100、クロック回路3200、乱数生成回路318が備えられており、これら全てが内部バス3300を介して互いに接続されている。内蔵ROM306は、記憶部の一例に相当する。なお、外部バス制御回路3110、パラレル入力ポート3102、アドレスデコード回路3103の3つが、図4のI/O310に相当するものである。

## 【0566】

以下、上記説明した各部の詳細について説明する。まず、CPU304、ROM306、RAM308については、図4にて説明した通りである。外部バス制御回路3110は、I/Oリクエスト端子（XIORQ端子）、メモリリクエスト端子（XMREQ端子）、リード信号端子（XRD端子）、ライト信号端子（XWR端子）、16ビット幅のアドレス出力端子（A0端子～A15端子）、および8ビット幅の入出力端子であるデータ入出力端子（D0端子～D7端子）を有する。本実施形態では、このうちデータ入出力端子（D0端子～D7端子）は、図4に示す各駆動回路324、326、330、334へのデータ出力と、各周辺制御回路からのデータ入力に用いられている。このデータ入出力端子（D0端子～D7端子）によるデータの入出力先は、アドレス出力端子（A0端子～A15端子）から出力されるアドレス信号、およびアドレスデコード回路3103から出力されるチップセレクト信号を用いて切り替えられる。

## 【0567】

パラレル入力ポート3102は、4つの入力端子（P0端子～P3端子）を有する。これらの入力端子（P0端子～P3端子）は、図4に示すセンサ回路322に接続されており、センサ回路322からの信号の入力に用いられている。本実施形態では、第1特図始動口230への入球を検出する球検出センサからの信号がP0端子に入力され、第2特図始動口232への入球を検出する球検出センサからの信号がP1端子に入力され、普図始動口228への入球を検出する球検出センサからの信号がP2端子に入力される。また、センサ回路322からの信号は、乱数生成回路318が生成する乱数をCPU304に取得させるためのラッチ信号として、乱数生成回路318に出力する。この動作については後述する。

## 【0568】

アドレスデコード回路3103は、14の出力端子（XCS0端子～XCS13端子）を有する。この出力端子（XCS0端子～XCS13端子）は、マイクロプロセッサ3000の外部にある周辺制御回路に接続されており、外部バス制御回路3110のデータ入出力端子（D0端子～D7端子）から出力されるデータの送信先を切り替えるためのチップセレクト信号等の出力に用いられている。

## 【0569】

タイマ回路311は、時間の計測に用いられる。なお、タイマ回路311は設定された計測時間を過ぎると、タイムアウト信号をカウンタ回路312に出力する。一方、カウンタ回路312は、各種信号の立ち上がり（あるいは立ち下がり）の回数の計測に用いられる。このカウンタ回路で計測される信号には、マイクロプロセッサ3000のシステムクロックの他、上記タイマ回路からのタイムアウト信号、メモリの読み書き信号、メモリリクエスト信号、外部入出力の信号、割込みに対する応答信号等も計測することができる。

## 【0570】

リセット制御回路314は、システムリセット入力端子（XSRST端子）と、リセット出力端子（XRSTO端子）の2つの端子を有する。このシステムリセット入力端子（XSRST端子）は電圧監視回路338に接続されている。このシステムリセット入力端子（XSRST端子）からシステムリセット信号（例えば一定時間Lレベルの信号）が入力されると、リセット制御回路314は、マイクロプロセッサ3000の内部の回路に対してこのシステムリセット信号を出力するとともに、マイクロプロセッサ3000の外部にある周辺制御回路に対してリセット出力端子（XRSTO端子）からリセット信号（例えば、LレベルからHレベルへの立ち上がり信号）が出力される。この場合、マイクロプロセッサ3000では、システムリセットと称する処理が実行され、各回路が初期化される。このシステムリセットが実行される一例として、電源投入時が挙げられる。なお、このシステムリセットについては後述する。

## 【0571】

また、リセット制御回路314は、ウォッチドッグタイマ（WDT）3141（復帰指示手段の一例に相当）、および指定エリア外走行禁止回路3142を備えている。WDT3141がタイムアウトになった場合や、CPU304が所定の範囲以外のアドレスを参照（指定エリア外走行）した場合には、リセット制御回路314は、マイクロプロセッサ3000の内部の回路に対してシステムリセット信号およびユーザリセット信号のいずれかを出力する。なお、システムリセット信号およびユーザリセット信号のどちらを出力するかは、ROM306内のプログラム管理エリア（詳細は後述）の設定に従う。また、マイクロプロセッサ3000の外部にある周辺制御回路に対しては、リセット出力端子（XRSTO端子）からリセット信号が出力される。

## 【0572】

マイクロプロセッサ3000では、設定によって上記のシステムリセットか、あるいはユーザリセットと称する処理のいずれかを実行させることができる。なお、ユーザリセットでは、各回路が必要に応じて初期化される。このユーザリセットについては後述する。

## 【0573】

上記指定エリア外走行は、プログラムが想定外の動作をしていることを意味する。この場合、CPU304が本来プログラムとして扱われるはずのないコードにより動作することになる。このような状況は、プログラムミスによる所謂暴走した状態の他に、何らかの不正によって生じている可能性がある。この場合、上記のシステムリセットの処理により、正常な動作に復帰させることができるようになっている。また、WDT3141がタイムアウトになった場合としては、プログラムミスによる暴走した状態や、電圧降下によりCPU304が本来設計した動作を行うことができなくなった場合等がある。この場合にも、上記のシステムリセットの処理により、正常な動作に復帰させることができるようになっている。

## 【0574】

割込み制御回路3100は、外部入力や内部状態の変化に応じて適宜処理を実行させるために割込みを発生させる。この割込み処理には、例えば外部からの入力（センサによる信号）を受け付けた場合に実行する処理がある。本実施形態では、乱数生成回路318が、乱数を取得するためのラッチ信号を受信した際にも、割込み処理を実行（割込みを発生）させることができるようになっている。なお、割込み制御回路3100は、内部情報レジスタ3101を備えており、この内部情報レジスタ3101には、乱数生成回路318

で乱数更新周期を決める外部クロック（カウントクロック）の周期の異常、および乱数の更新に関するの異常、さらに、直前に発生したリセット要因の情報等が格納される。なお、この内部情報レジスタ 3101 は、異常検出情報保持手段の一例に相当する。

#### 【0575】

クロック回路 3200 は、図 4 に示す水晶発振器 316b（以下、システム用水晶発振器 316b と称する場合がある）から外部クロック入力端子（EX 端子）を介して入力される外部クロック（この例では、24MHz のクロック）を所定の分周比（この例では、1/2）で分周し、分周後のシステムクロック（この例では、12MHz のクロック）をこのマイクロプロセッサ 3000 内部の各回路に供給する。また、このシステムクロックをシステムクロック出力端子（CLKO 端子）を介してマイクロプロセッサ 3000 外部の周辺制御回路に出力する。

乱数生成回路 318 は、乱数を更新するためのクロック信号（カウントクロック）を用いて、乱数のラッチ信号を受信したときにこの更新された乱数を乱数レジスタ内に保持するものである。本実施形態では、水晶発振器 316a から外部クロック入力端子（RCK 端子）を介して入力される外部クロック信号を所定の分周比（この例では、1/2）で分周してこのカウントクロックに用いているが、マイクロプロセッサ 3000 内部のクロック信号を用いることもでき、この場合は水晶発振器 316a は不要となる。乱数レジスタに保持された値は、乱数として読み出して使用することができる。なお、乱数レジスタから乱数を読み出すと、乱数レジスタが次の乱数をラッチすることを許容する許容状態とすることができる。この乱数生成回路 318 の詳細は後述する。

#### 【0576】

続いて、図 76 を用いて、リセットが実行された場合の処理の流れについて説明する。同図は、リセットの流れを示すフローチャートである。このフローチャートは、リセット制御回路 314 にシステムリセット信号が入力された場合、WDT 3141 がタイムアウトになった場合、あるいは指定エリア外走行を検知した場合に、マイクロプロセッサ 3000 の各回路で実行される処理である。

#### 【0577】

最初のステップ SH01 では、実行されるリセット動作が、システムリセット動作か否か判定される。マイクロプロセッサ 3000 で実行されるリセット動作には、システムリセット動作とユーザリセット動作の 2 つがある。ここで、システムリセット動作を実行する場合には、ステップ SH03 に進む。このシステムリセット動作は本発明にいう復帰処理の一例に相当する。また、システムリセット動作でないリセット動作、すなわちユーザリセット動作を実行する場合には、ステップ SH11 に進む。

#### 【0578】

ステップ SH03 では、第 1 内部回路初期化処理が実行される。この第 1 内部回路初期化処理は、CPU 304 のコアや内蔵レジスタ（タイマ回路 311、カウンタ回路 312、パラレル入力ポート 3102、RAM 308 アクセスプロテクトレジスタ、割込み制御回路 3100、乱数生成回路 318 を制御するレジスタ）の値を初期化する。この第 1 内部回路初期化処理が終了すると、ステップ SH05 に進む。

#### 【0579】

ステップ SH05 では、セキュリティチェック処理が実行される。このセキュリティチェック処理では、ユーザプログラムを基に計算された値が、ROM 306 のプログラム管理エリアに記憶されている認証コードを表す値と一致するか否かの判定を行う。すなわち、認証コードが正しいか否かの再計算を行う。ここにいうユーザプログラムが、本発明にいう記憶部（内蔵 ROM 306）に記憶された内容の一例に相当し、内蔵 ROM 306 のプログラム管理エリアに記憶されている認証コードが、本発明にいう所定の認証情報の一例に相当する。認証コードが正しい場合にはステップ SH07 に進み、そうでない場合には CPU 304 の動作を停止させる。本実施形態の遊技台では、このようにしてセキュリティチェックを行なうため、遊技台により安定した制御を行なわせることができる。

#### 【0580】

ステップSH07では、固定延長処理が実行される。この固定延長処理では、予め設定した固定時間（例えば、ROM306のプログラム管理エリア内のセキュリティ時間設定の0～2ビットを用いて設定）だけセキュリティモードを延長する。例えば、nをプログラム管理エリアに設定された時間とし、システムクロックをSCLKとした場合に、 $3n \times 2^24 / SCLK$ 秒だけ延長する。なお、この延長時間の経過時にXRSTO端子からリセット信号を出力させる。その後ステップSH09に進む。

#### 【0581】

ステップSH09では、ランダム延長処理が実行される。このランダム延長処理では、予めランダムに選択されたランダム時間（例えば、プログラム管理エリア内のセキュリティ時間設定の3～4ビットを用いて設定）だけセキュリティモードを延長する。例えばショートモードが設定されている場合には0～Sμ秒、ミドルモードが設定されている場合には0～Mμ秒、ロングモードが設定されている場合には、0～Lμ秒それぞれ延長する（ $S < M < L$ ）。この処理が終了するとユーザモードに移行し、CPU304は、メモリマップの0000H番地から処理を開始する。本実施形態では、主制御部300のメイン処理が開始されることになる。

#### 【0582】

一方、ステップSH01で、ユーザリセット動作を実行する場合に進むステップSH11では、第2内部回路初期化処理が実行される。この第2内部回路初期化処理は、CPU304のコアや、乱数生成回路318を制御するレジスタを除く内蔵レジスタ（タイマ回路311、カウンタ回路312、パラレル入力ポート3102、RAM308アクセスプロテクトレジスタ、割込み制御回路3100）の値を初期化する。すなわち、この第2内部回路初期化処理では、乱数生成回路318を制御するレジスタの値はリセット前の状態が維持される。この第2内部回路初期化処理が終了すると、ユーザモードに移行し、CPU304は、メモリマップの0000H番地から処理を開始する。本実施形態では、主制御部300のメイン処理が開始されることになる。

#### 【0583】

##### <乱数生成回路>

次に、図77を用いて、図75に示すマイクロプロセッサ3000の乱数生成回路318の詳細について説明する。同図は、図75に示す乱数生成回路318の内部構成図である。

#### 【0584】

乱数生成回路318は、それぞれ異なる乱数を生成する4つの乱数生成チャンネルCH1～CH4を備えている。なお、各チャンネル回路の内部構成は同一であるため、図77では、1つの乱数生成チャンネルCH1を示し、残りの乱数生成チャンネルCH2～CH4については、図示を簡略化している。以下の説明では、この乱数生成チャンネルCH1を中心に乱数生成回路318について説明する。

#### 【0585】

乱数生成回路318は、初期設定レジスタ3181と、周波数監視回路3182と、乱数更新回路3183と、乱数監視回路3184と、ノイズフィルタ3185と、ソフトラッチレジスタ3186と、ラッチ選択レジスタ3187と、乱数レジスタ3188と、乱数ラッチフラグレジスタ3189と、乱数割込み制御レジスタ3180を備えている。なお、周波数監視回路3182および乱数監視回路3184は、更新異常検出手段の一例に相当する。

#### 【0586】

初期設定レジスタ3181には、RCK端子からの外部クロック信号とシステムクロック（内部クロック信号）のうち、いずれのクロック信号を乱数更新回路3183で使用するかを決定するクロック選択情報が、プログラム管理エリア内の情報に基づいて設定される。RCK端子からの外部クロック信号とシステムクロック（内部クロック信号）は乱数更新回路3183の手前に設けられたマルチプレクサに入力される。初期設定レジスタ3181が上記クロック選択情報に従って更新クロック選択信号をこのマルチプレクサに入力



することにより、この更新クロック選択信号によって選択されたクロック信号が乱数更新回路 3 1 8 3 に入力されるようになっている。なお、外部クロック信号が選択されている場合、所定の分周比（この例では、 $1/2$ ）で分周されたクロック信号が乱数更新回路 3 1 8 3 に入力される。なお、分周されたクロック信号が、内部クロックよりも低い周波数でない場合には使用することができない。なお、本実施形態では、外部クロック信号が選択されているものとして説明を続ける。

#### 【0587】

周波数監視回路 3 1 8 2 は、内部クロック信号および外部クロック信号の周期を監視し、この周期が一定でなくなった場合に、クロック信号に異常があることを示す情報を、内部情報レジスタ 3 1 0 1 に出力する。

10

#### 【0588】

乱数更新回路 3 1 8 3 は、初期設定レジスタ 3 1 8 1 によって選択されたクロック信号が入力され、このクロック信号の周期に従って乱数を更新する。更新した乱数の値は、乱数監視回路 3 1 8 4 および乱数レジスタ 3 1 8 8 にそれぞれ出力される。この乱数更新回路 3 1 8 3 の詳細については、図 7 9～図 8 2 を用いて後述する。

#### 【0589】

なお、乱数値の更新は、乱数更新回路 3 1 8 3 に最大値が設定されない C H については、ユーザモードに移行した時に開始（自動開始）されるように構成される一方、乱数更新回路 3 1 8 3 に最大値が設定される C H については、最大値が設定された時に開始（手動開始）されるように構成されている。すなわち、乱数の最大値は、乱数生成チャンネル C H 1～C H 4 で個別に設定可能である。また、最大値が設定されない C H および最大値が設定される C H ともに一度更新が開始されると電源供給が断たれるかシステムリセットが発生した場合にのみ更新が停止されるものであり、予期せぬ異常が発生した場合を除きこれらの事象以外では更新を停止することができないように構成されている。

20

さらに、最大値が設定される C H に関しては、乱数更新回路 3 1 8 3 に設けられた最大値設定レジスタに最大値が設定されたことに基づいて更新が開始され、この最大値設定レジスタはシステムリセットごとに一回のみライト可能なレジスタ（システムリセット時には初期化（更新範囲の最大値である F F F F H が初期値として設定される）され、ユーザリセット発生時にはユーザリセット発生前に設定されていた値が維持されるレジスタ）である。すなわち、乱数が更新されている場合にユーザプログラムで最大値設定レジスタをライトする処理が実行されたとしても、最大値設定レジスタが更新されることがなく、乱数生成回路 3 1 8 3 はその影響を受けないように構成されている。一方、リードに関してはユーザプログラムで自由にすることができるよう構成されている。

30

#### 【0590】

また、16ビット乱数値および8ビット乱数値の一方を初期値が設定可能に構成し、他方を初期値が設定不可能に構成してもよいし、乱数が正常に更新されているか否かについて一方を監視可能とし、他方を監視不能としてもよい。また、後述する内部リセットの場合と外部リセットの場合で乱数値の初期値を異ならせてもよく（例えば、内部リセット時の初期値は 3 3 H、外部リセット時の初期値は C C H）、この場合、ユーザプログラムの初期設定などにおいて、乱数値の初期値に基づいてリセットの種類を判別し、リセットの種類に合わせた処理を実行してもよい。

40

#### 【0591】

乱数監視回路 3 1 8 4 は、乱数更新回路 3 1 8 3 からの入力に基づいて、乱数が正常に更新されているか否かを監視する。乱数の更新に異常がある場合、この乱数の更新に異常があることを示す情報を、内部情報レジスタ 3 1 0 1 に出力する。

#### 【0592】

なお、上記の説明では、乱数生成回路 3 1 8 の内部に周波数監視回路 3 1 8 2 および乱数監視回路 3 1 8 4 が設けられている構成であったが、乱数生成回路 3 1 8 の外部に設けてもよい。また、上記の説明では、乱数更新回路 3 1 8 3 とは別に周波数監視回路 3 1 8 2 および乱数監視回路 3 1 8 4 が設けられている構成であったが、乱数更新回路 3 1 8 3

50

の内部にこれらの回路を設けてもよい。

#### 【0593】

ノイズフィルタ3185には、P0端子～P3端子からの入力、パラレル入力ポート3102を介してラッチ信号として入力される。なお、図77では、P3端子、およびそのP3端子に接続された回路等は図示省略している。上述のごとく、P0端子には、第1特図始動口230への入球を検出する球検出センサからの信号が入力され、P1端子には、第2特図始動口232への入球を検出する球検出センサからの信号が入力され、P2端子には、普図始動口228への入球を検出する球検出センサからの信号が入力される。さらに、ノイズフィルタ3185には、初期設定レジスタ3181によって選択されたクロック信号が入力される。このクロック信号を利用して、P0端子～P3端子からの入力に生じるノイズを除去した上で、ラッチ信号を検出する。このラッチ信号が検出されると、ハードラッチ信号が乱数レジスタ3188の手前に設けられたマルチプレクサに出力される。なお、このノイズ除去の詳細については、図78を用いて後述する。

10

#### 【0594】

ソフトラッチレジスタ3186には、乱数レジスタ3188から乱数をラッチすることを示す情報がCPU304からの指示によって適宜設定される。この情報が、ソフトラッチ信号として乱数レジスタ3188の手前に設けられたマルチプレクサに出力される。

#### 【0595】

ラッチ選択レジスタ3187には、乱数レジスタ3188の手前に設けられたマルチプレクサからハードラッチ信号とソフトラッチ信号の何れを出力させるか、すなわち、乱数レジスタ3188にどちらのラッチ信号を入力するのかわかる情報を、CPU304からの指示によって適宜設定される。この情報を設定することにより、ハードラッチ信号とソフトラッチ信号を適宜使用して、乱数を取得することができる。

20

#### 【0596】

乱数レジスタ3188には、3つの信号が入力される。1つ目の信号は、乱数更新回路3183から出力された乱数を表す信号である。2つ目の信号は、手前に設けられたマルチプレクサから出力された乱数ラッチ信号（ハードラッチ信号とソフトラッチ信号のうち、ラッチ選択レジスタ3187で設定された信号）である。3つ目の信号は、乱数の読み取りを示すリード信号である。

#### 【0597】

乱数レジスタ3188には、乱数更新回路3183によって更新された乱数を示す信号が常に入力されている。ここで、乱数ラッチ信号が入力されると、この入力タイミングにおける乱数が乱数レジスタ3188内にラッチ（保持）される。このとき、乱数レジスタ3188からは、乱数がラッチされていることを示すセット信号が、乱数ラッチフラグレジスタ3189に出力される。このときCPU304は、ラッチされている乱数を取得することができる。なお、CPU304により乱数が取得されると、乱数レジスタ3188にリード信号が入力される。この信号により新たな乱数ラッチ信号が入力された場合に乱数をラッチすることを許容する許容状態となる。換言すれば、一度乱数をラッチすると、リード信号が入力されるまで新たに乱数をラッチすることができない非許容状態となる。なお、リード信号が入力されてもラッチされている乱数は保持し続けるため、CPU304はラッチされている同一のタイミングでラッチされた乱数を何度でも取得することができる。このように構成することで乱数ラッチ信号を出力するセンサ回路におけるチャタリングの影響を押さえることができる。なお、リード信号が入力されたことを示すクリア信号が乱数ラッチフラグレジスタ3189に出力される。また、図77に示すように、乱数レジスタ3188が複数あることにより、同じ乱数更新回路から生成された乱数を様々なタイミングで取得することができる。

30

40

#### 【0598】

乱数ラッチフラグレジスタ3189には、乱数レジスタ3188に乱数がラッチされているか否かを示す情報が記憶される。なお、乱数ラッチフラグレジスタ3189は、乱数レジスタ3188にラッチされた乱数（16ビット乱数の全てまたは一部や、8ビット乱

50

数の全てまたは一部)が読み出されたときに自動的にクリアされるように構成してもよいし、CPUによってクリアするように構成してもよい。

#### 【0599】

乱数割込み制御レジスタ3180には、乱数レジスタ3188に乱数が保持された際に割込み制御回路に割込みを発生させるか否かを示す情報が設定される。この情報は乱数レジスタ3188毎に設定することができる。例えば、第1特図始動口230への入球により乱数発生チャンネルCH1の乱数がラッチされた場合には、割込みを発生させ、第2特図始動口232への入球により乱数発生チャンネルCH2の乱数がラッチされた場合には、割込みを発生させない、というような設定をすることができる。

#### 【0600】

上記説明した乱数生成回路318を用いて、本実施形態では、普図始動口282、第1特図始動口230、第2特図始動口232のそれぞれに入球したタイミングで乱数を取得する。なお、任意のタイミングでCPU304に乱数を取得させるように、プログラムを実行させることもできる。さらに異なる乱数生成チャンネルを使用してそれぞれ独立の乱数を取得し、それぞれのチャンネル毎に乱数生成範囲の最大値を設定することができる。

#### 【0601】

なお、本実施形態では、個別のチャンネルに対して乱数生成範囲を変更することができ、設定されていない場合には所定の最大値(例えば65535)が適用される例について説明したが、これ以外に例えば、乱数生成範囲を変更する場合に全てのチャンネルに対して乱数生成範囲を設定することを必要とする構成であってもよい。

#### 【0602】

なお、普図始動口282に入球したタイミングで取得された乱数は、後述する普図当選乱数として用いられる。また、第1特図始動口230に入球したタイミングで取得された乱数は、加工(例えば、乱数生成回路とは異なる回路(例えば、基本回路302やカウンタ回路312)でカウントアップ(ダウン)される値を乱数に加算(減算)する処理など)されて、後述する特図1当選乱数として用いられる。さらに、第2特図始動口232に入球したタイミングで取得された乱数は、加工されて、後述する特図2当選乱数として用いられる。

#### 【0603】

なお、乱数の加工は、第1特図始動口230および第2特図始動口232に入球したタイミングで取得される乱数に限らず、他の契機(例えば、普図始動口282へに入球したタイミングなど)によって取得される乱数に対して行ってもよい。さらに、上述した乱数生成範囲の最大値を設定する場合においては、設定した最大値に応じた加工(例えば、この最大値を超えない値を乱数に加算する処理など)すれば良い。

#### 【0604】

また、乱数の加工は必ずしも行う必要はないが、乱数の加工を行うことは乱数の狙いうちの防止に有効に働く。例えば加工した後の乱数が更新されたか否かをCPU304によって監視した場合、乱数生成回路318内での乱数の更新が正常に行われているか否かを正確に判断することができない。このように、乱数を加工する場合には、上述の乱数監視回路3184による乱数の更新の監視がより有効に働くこととなる。

#### 【0605】

続いて、図78を用いてノイズフィルタ3185の動作について説明する。同図は、図77に示すノイズフィルタ3185による処理の一例を示す図である。

#### 【0606】

上述したように、ノイズフィルタ3185には、P0端子～P3端子からの入力が、パラレル入力ポート3102を介してラッチ信号として入力され、初期設定レジスタ3181によって選択されたクロック信号が入力される。このノイズフィルタ3185は、例えば図78に示すように、クロック信号のダウンエッジ(HレベルからLレベルへの立ち下り信号)が4回連続で入力されるまで、P0端子～P3端子からの入力信号がHレベルであれば、ハードウェアラッチ信号を出力するように構成されている。単純に信号の立ち上

10

20

30

40

50

がりおよび立ち下がりだけを基にハードウェアラッチ信号を出力すると、ノイズが入った場合にもハードウェアラッチ信号が出力されてしまう。このため、上記のようにクロック信号を用いて、一定期間のセンサ信号を検出させることで、ノイズによりハードウェアラッチ信号が出力されてしまわないように構成されている。

#### 【0607】

上記説明では、P0端子～P3端子からの入力によりハードラッチ信号を出力させる例について説明したが、例えば、P0端子～P3端子からの入力を一度CPU304で受け付けた上で乱数を取得する方法もある。この場合には、ハードラッチ信号ではなく、ソフトラッチレジスタ3186から出力されたソフトラッチ信号により、乱数をラッチさせることになる。この場合、上記ノイズフィルタによる処理と同様の処理がCPU304により実行される。すなわち、P0端子～P3端子からの入力が所定期間継続したか否かを判定し、これを満たす場合にP0端子～P3端子からの入力が正しく行われたと判定する。その後、ソフトラッチ信号を出力するようにソフトラッチレジスタ3186を設定する。

#### 【0608】

次に、図79を用いて、図77に示す乱数更新回路3183の詳細について説明する。同図は、図77に示す乱数更新回路3183の詳細を示す図である。

#### 【0609】

図79(a)に示す乱数更新回路3183は、カウンタ回路3183aと、スタート値選択回路3183bと、最大値設定レジスタ3183cを備えている。なお、図77では、乱数更新回路3183に対して外部クロック信号および内部クロック信号のいずれかが乱数更新のための更新クロック信号として入力されることが示されているが、この図79に示す乱数更新回路3183には、さらにCPU304から最大値データが入力される。

#### 【0610】

カウンタ回路3183aからは、乱数（乱数データ）が出力される。この乱数の初期値は、スタート値選択回路3183bに設定されている。この初期値には、固定値（例えば0）、プログラム管理エリア内に記憶されているマイクロプロセッサ3000のIDナンバーを基にした値、および、乱数生成範囲内で抽選された値のいずれかが設定される。

#### 【0611】

カウンタ回路3183aには、更新クロック信号の1クロック毎に1が加算されるカウンタが設けられており、このカウンタの値に、スタート値選択回路3183bから入力された乱数の初期値を加えた値を乱数として出力する。

#### 【0612】

カウンタ回路3183aから出力される乱数の最大値は、最大値設定レジスタ3183cに設定されている。なお、乱数の最大値は、CPU304からの命令に従って最大値設定レジスタ3183cに設定される。図79(b)には、この最大値が設定されている16ビットのレジスタが示されている。CPU304から最大値を設定する命令がない場合の初期値は、FFFFH（65535）である。なお、この最大値設定レジスタ3183cの変わりに、例えば、ROM306のプログラム管理エリア内に最大値を設定する領域を設け、設定された最大値を参照する構成であってもよい。

#### 【0613】

カウンタ回路3183aは、出力する乱数が最大値を超えた場合、この最大値に1を加えた値を減算した上で、この値を乱数として出力する。例えば最大値が65535の場合において、初期値とカウンタの値を足した値が、65534、65535、65536、65537の順に更新されるとすると、出力される乱数は、65534、65535、0（65536－（65535＋1））、1（65537－（65535＋1））のようになる。すなわち、最大値を超えると出力される乱数が0に戻るようになる。

#### 【0614】

また、カウンタ回路3183aの内部に設けられたカウンタの値が、最大値レジスタ3183cに設定された最大値になると、このカウンタの値が0にクリアされるとともに、カウンタ回路3183aからスタート値選択回路3183bに乱数の出力が一巡したこと

10

20

30

40

50

を示す信号（一巡信号）が出力される。この一巡信号を受けたスタート値選択回路 3 1 8 3 b では、乱数の初期値が更新される。この際更新される初期値が、最大値設定レジスタ 3 1 8 3 c に設定されている乱数の最大値を超えている場合、例えば、固定値を用いたり、一度現在の乱数の最大値を乗じてから設定可能な乱数生成範囲の最大値（6 5 5 3 5）で除算した値を用いることにより、乱数の最大値を超えない値（乱数生成範囲内の値）が初期値として再設定される。無論、最初からこの再設定方法などを用いて、乱数生成範囲内の値が初期値として設定されるようにしてもよい。

#### 【0 6 1 5】

なお、本実施形態では、最大値設定レジスタ 3 1 8 3 c に設定された最大値は、カウンタ回路 3 1 8 3 a からの出力が一巡した時点で自動的に更新されるが、例えば C P U 3 0 4 からの更新命令が来た時点で更新されるようにしてもよい。

#### 【0 6 1 6】

ここで、上記の乱数生成の範囲を図 8 0 および図 8 1 を用いて説明する。図 8 0 は、乱数生成範囲の最大値が設定されていない場合（デフォルトの状態）において出力される乱数の範囲を示す図である。また、図 8 1 は、図 8 0 とは異なる最大値が設定された場合に出力される乱数の範囲を示す図である。

#### 【0 6 1 7】

まず、図 8 0（a）では  $n$  周期目（ $n - 1$  回目の一巡信号が出力された直後）の状況について説明する。図 8 0（a）には、0 ～ 6 5 5 3 5 の乱数生成範囲が示されている。また、出力される乱数の初期値がスタート値（X）として示されている。

#### 【0 6 1 8】

乱数更新回路 3 1 8 3 からは、更新クロックの 1 クロック毎に乱数が更新されて出力される。より具体的には、最初にスタート値（X）が出力され、その後 1 クロック毎に、 $X + 1$ 、 $X + 2$  の順で更新された乱数が出力される。この乱数の値が乱数生成範囲の最大値である 6 5 5 3 5 になると、上記説明した処理によって、次に出力される乱数は 0 に戻る。その後、1、2、の順で更新された乱数が出力され、 $X - 1$  が出力された時点でこの乱数生成範囲を一巡したことになる。なお、このとき、乱数更新回路 3 1 8 3 内のカウンタ回路 3 1 8 3 a のカウンタの値は最大値と同じ値になっている。スタート値選択回路 3 1 8 3 b に乱数の出力が一巡したことを示す信号（一巡信号）が出力される。

#### 【0 6 1 9】

続いて 1 クロックが入力されると、乱数更新回路 3 1 8 3 の出力は  $n + 1$  周期目（ $n$  回目の一巡信号が出力された直後）に入る。この  $n + 1$  周期目の状況について、図 8 0（b）を用いて説明する。まず、乱数更新回路 3 1 8 3 内のカウンタ回路 3 1 8 3 a のカウンタの値は 0 にクリアされる。また、スタート値選択回路 3 1 8 3 b では、新たな初期値が設定される。図 8 0（b）には、この新たな初期値がスタート値（X'）として示されている。この新たな初期値と、カウンタ回路 3 1 8 3 a のカウンタの値により、上記説明した流れと同様の流れで乱数が出力される。

#### 【0 6 2 0】

次に、図 8 1（a）では、図 8 0 に示す乱数生成範囲の最大値よりも小さい最大値が設定された場合の  $n$  周期目（ $n - 1$  回目の一巡信号が出力された直後）の状況について説明する。図 8 1（a）には、0 ～ 6 5 5 3 5 のうち、設定された最大値によって図 8 0（a）に示す乱数生成範囲よりも狭くなった乱数生成範囲が示されている。なお、出力されない乱数の範囲が左下がりのハッチングで示されている。また、出力される乱数の初期値がスタート値（Y）として示されている。

#### 【0 6 2 1】

図 8 0 を用いて説明した状況と同様に、乱数更新回路 3 1 8 3 からは、最初にスタート値（Y）が出力され、その後 1 クロック毎に、 $Y + 1$ 、 $Y + 2$  の順で更新された乱数が出力される。この乱数の値が設定された最大値になると、上記説明した処理によって、次に出力される乱数は 0 に戻る。すなわち、左下がりのハッチングで示された範囲の乱数は出力されない。その後、1、2、の順で更新された乱数が出力され、 $Y - 1$  が出力された時

10

20

30

40

50

点でこの乱数生成範囲を一巡したことになる。なお、このとき、乱数更新回路 3 1 8 3 内のカウンタ回路 3 1 8 3 a のカウンタの値は設定された最大値と同じ値になっている。スタート値選択回路 3 1 8 3 b に乱数の出力が一巡したことを示す信号（一巡信号）が出力される。

#### 【0622】

続いて 1 クロックが入力されると、乱数更新回路 3 1 8 3 の出力は  $n + 1$  周期目（ $n$  回目の一巡信号が出力された直後）に入る。この  $n + 1$  周期目の状況について、図 8 1（b）を用いて説明する。図 8 0（b）を用いて説明した場合と同様、乱数更新回路 3 1 8 3 内のカウンタ回路 3 1 8 3 a のカウンタの値は 0 にクリアされる。また、スタート値選択回路 3 1 8 3 b では、新たな初期値が設定される。図 8 1（b）には、この新たな初期値がスタート値（ $Y'$ ）として示されている。この新たな初期値と、カウンタ回路 3 1 8 3 a のカウンタの値により、上記説明した流れと同様の流れで乱数が出力される。

10

#### 【0623】

なお、上記のように乱数生成チャンネルの最大値を適切に設定する以外に、例えば、ある乱数生成チャンネルでは乱数の最大値が 6 5 5 3 5 であるが、別の乱数生成チャンネルでは乱数の最大値が 2 5 5 である、というように、チャンネル毎に乱数生成範囲を予め異ならせておいてもよい。この場合には、必要とされる乱数生成範囲に応じてチャンネルを選択すればよい。この場合には、最大値設定レジスタの容量削減や、最大値設定の処理負担を軽減することができる。さらに、予め設定可能な最大値を複数設定した上で、これらの中から適切な最大値を選択して設定するようにしても、最大値設定レジスタの容量削減や、最大値設定の処理負担を軽減することができる。

20

#### 【0624】

以上、乱数更新回路 3 1 8 3 の乱数の出力について説明した。以下この乱数更新回路 3 1 8 3 の変形例について図 8 2 を用いて説明する。同図は、最大値と最小値を設定した乱数生成範囲において、乱数を取得することができる範囲を示す図である。

#### 【0625】

図 7 9 から図 8 1 の例では、乱数生成範囲の最大値を変更する例について説明したが、例えば、最小値を設定できるものであってもよい。さらに、最大値だけでなく最小値を設定するレジスタを用意し、図 8 2（a）に示すように、最小値、最大値を設定し、乱数生成範囲を設定できるものであってもよい。加えて、図 8 2（b）に示すように、複数の乱数生成範囲を設定できるものであってもよい。すなわち、出力される乱数の範囲を設定出来る構成であればよい。また、上記説明では、説明を分かり易くするため出力される乱数が 1 ずつ増加する例について説明したが、これ以外の乱数の更新方法であってもよい。

30

#### 【0626】

以下、図 8 3 を用いて本実施形態の遊技台で使用される乱数と、その主な導出源について説明する。同図は、本実施形態の遊技台で使用される乱数の導出源を表で示す図である。なお、この図で示される各乱数は、後述のフローチャートで用いられる。

#### 【0627】

まず、特図 1 当選乱数、特図 2 当選乱数、および普図当選乱数は、上記乱数生成回路 3 1 8 の乱数生成チャンネル CH 1 ～ CH 4 によって生成される乱数が基になる。この値を必要に応じて適宜加工してこれらの乱数として使用する。

40

#### 【0628】

大当り用特図乱数、小当り用特図乱数、およびハズレ用特図乱数は、後述する主制御部タイマ割込み処理において生成される。すなわちこれらの乱数は、所謂ソフトウェア乱数である。なお、これらの乱数を生成する際に用いられる初期値生成用乱数は、主制御部メイン処理および主制御部タイマ割込み処理で生成される。

#### 【0629】

特図変動時間決定用乱数、および普図変動時間決定用乱数は、カウンタ回路 3 1 2 の値を乱数として用いる。本実施形態のカウント回路 3 1 2 は、マイクロプロセッサ 3 0 0 0 のシステムクロックの他、上記タイマ回路からのタイムアウト信号、メモリの読み書き信

50

号、メモリリクエスト信号、外部入出力の信号等もカウンタの対象として用いることができる。このため、これらを組み合わせることで規則性のない値を導出させ、上記の乱数に使用している。演出用乱数は、主制御部メイン処理および主制御部タイマ割込み処理で生成される。

#### 【0630】

次に、図84を用いて、図75に示す割込み制御回路3100に設けられた内部情報レジスタ3101の詳細について説明する。上述したように、内部情報レジスタ3101には、乱数生成回路318で乱数更新周期を決める外部クロック（カウントクロック）の周期の異常、および生成された乱数値の異常、さらに、直前に発生したユーザリセットのリセット要因の情報等が格納される。この内部レジスタ3101には、第1内部情報レジスタと第2内部情報レジスタと、第3内部情報レジスタとが用意されている。

10

#### 【0631】

図84は、割込み制御回路3100の内部レジスタ3101に用意された第1内部情報レジスタを説明するための図である。図84（a）には、第1内部情報レジスタのうち、乱数生成回路318の異常を示す情報が記憶される範囲が示されている。この範囲は8ビットで構成されており、図84（b）には、これらの各ビットがどのような情報を示すものが表で示されている。

#### 【0632】

CPU304は、後述するタイマ割込みが実行される度にこの第1内部情報レジスタの内容を確認する（図95のステップS3235a）。この内容は、異常が発生した場合に遊技の進行を止める等の処理を行うか否かの判定に用いられる。例えば、本実施形態では、乱数生成回路318で異常が検出されると、入賞受付処理が実行されないように構成されている（図95のステップS3235cおよび図96参照）。この他、球の打ち出しを行わないようにしたり、払出しを行わないようにしたり、あるいは、変動タイマの減算を行わないようにしてもよい。特に、乱数更新に異常があった場合、乱数更新回路3183に大電流が流れている状態（ラッチアップ状態）である可能性があるため、一度電源をオフにすることを報知するようにしてもよい。

20

#### 【0633】

ビット番号0は、乱数生成回路318の外部クロック（更新クロック）の異常を示すビットであり、0は異常なし、1は異常ありを示す。ビット番号1～4は、乱数生成回路318で生成された乱数の異常を、各チャンネル毎に示すビットであり、0は異常なし、1は異常ありを示す。ビット番号5～7は、不使用（0に固定）である。

30

#### 【0634】

第1内部情報レジスタの値が1である場合、CPU304から読み出されると0に設定（クリア）される。なお、内部情報レジスタはCPU304に一度に読み出されるため、第1内部情報レジスタが保持している値を0に設定（クリア）したい場合は、CPU304に第1内部情報レジスタの値を読み出した後に読み出した値を破棄すれば良い。また、上述の通り、システムリセットおよびユーザリセットのいずれかの処理が実行された場合に、この内部情報レジスタは、CPU304から読み出された場合と同様に各ビットは0に設定（クリア）されることとなる。

40

#### 【0635】

以上、乱数生成回路318の異常がどのように記憶されているかについて説明したが、この例と同様に上記リセット要因を示す情報（システムリセットおよびユーザリセットのうちのいずれのリセットが発生したのかを識別可能な情報）についても、第1内部情報レジスタに記憶されている。なお、上記説明では、乱数生成チャンネルの異常をビット番号1～4によってそれぞれのチャンネル毎に表していたが、1つのビットによっていずれかのチャンネルで異常が起きたか否かを表すようにしてもよい（全てのチャンネルで共通にしてもよい）。なお、本実施形態では、内部情報レジスタ3101が割込み制御回路3100内に設けられているが、他の回路に設けられているものであってもよい。なお、割込み制御回路3100の内部レジスタ3101に用意された第2内部情報レジスタおよび第

50

3 内部情報レジスタについては後述する。

#### 【0636】

次に、図77に示す周波数監視回路3182における異常の検出例について、図85を用いて説明する。同図は、周波数監視回路3182における異常の検出例を示す図である。図85には、一番上からそれぞれ、正常時の動作、異常時の動作例1（異常時1）、異常時の動作例2（異常時2）が示されている。これらの例では、マイクロプロセッサ3000の内部クロックと、乱数生成回路318に入力される外部クロック（RCK）に加え、外部クロックの1周期に対する内部クロックの周期の比率が示されている。この検出動作では、外部クロックと内部クロックの周期の比率を監視し、この比率が変化した場合を異常として検出する。

10

#### 【0637】

図85（a）に示す正常時の例では、外部クロックの1周期の間に内部クロックが2.5周期ある状態が継続している。すなわち、外部クロックと内部クロックの周期の比率が変化しないため、異常を示す情報は出力されない。

#### 【0638】

次に、図85（b）に示す異常時の動作例1（異常時1）では、外部クロックの1周期の間に内部クロックが4.5周期ある状態が示されている。図85（a）に示す状態からこの図85（b）に示す状態になった場合には、この時点で異常を示す情報が出力される。

#### 【0639】

次に、図85（c）に示す異常時の動作例2（異常時2）の最初には、外部クロックの1周期の間に内部クロックが0.5周期ある状態が示されている。仮に、図85（a）に示す状態からこの図85（c）の最初の状態になった場合には、この時点で異常を示す情報が出力される。さらに、図85（c）には、外部クロックの1周期の間に内部クロックが0.5周期ある状態に続き、外部クロックの1周期の間に内部クロックが2.5周期ある状態が継続している。この周期の比率が変化した時点でも異常を示す情報が出力される。

20

#### 【0640】

上記の例では、外部クロックと内部クロックのいずれか一方の周期が異常になると、異常を示す情報が出力されることになる。すなわち、内部クロックを乱数の更新クロックとして使用した場合でも、異常を検出することができる。なお、外部クロックと内部クロックの比率が同じであれば、双方の周波数が変更されても異常を示す情報が出力されない。この場合、例えばプログラム管理エリア内に外部クロック又は内部クロックの周波数を示す情報を記憶させておき、この情報を用いて外部クロック又は内部クロックの異常を検出させることも可能である。なお、更新クロックの異常を検出する方法は上述した方法に限らず、更新クロックの周期の異常を検出可能な方法であれば他の方法を採用しても良い。

30

#### 【0641】

続いて、図77に示す乱数監視回路3184における異常の検出例について、図86を用いて説明する。同図は、乱数監視回路3184における異常の検出例を示す図である。この検出例では、1回の乱数の更新毎に更新前後の乱数を比較し、同じ乱数が生成されていないか否かをチェックする。仮に同じ乱数が生成されている場合には、異常を示す情報が出力される。図86（a）の例では、更新前後で同じ乱数が生成されていないため、異常を示す情報は出力されない。これに対して、図86（b）の例では、更新の途中で同じ乱数が生成されており（乱数4が2回生成されている）、この時点（2回目の4が出力された時点）で異常を示す情報が出力される。なお、上記の例では、乱数が更新される度にチェックを行っているが、この乱数更新周期よりも長い所定の周期毎でチェックさせてもよい。

40

#### 【0642】

上記説明したように乱数生成回路318では、周波数監視回路3182と乱数監視回路3184から異常を示す情報が内部情報レジスタ3101に対して出力される。また、内

50



部情報レジスタ 3101 の値は、タイマ割込み毎に CPU 304 によって読み出される。

#### 【0643】

このように周波数監視回路 3182 および乱数監視回路 3184 の双方を用いて乱数生成回路 318 の異常を監視することで、更新クロックの周波数に異常があるが、乱数の更新には異常がない異常状態（乱数監視回路 3184 のみでは検出できない異常状態）と、更新クロックの周波数に異常がないが、乱数の更新には異常がある異常状態の双方の異常状態を的確に把握することが可能になり、遊技制御の安定化を図ることができる。

#### 【0644】

従来の遊技台では、遊技者に有利な抽選結果が導出されやすくなるように抽選に用いられる乱数に狙いを定めて不正が行われることがあり、遊技の公平性を担保することが困難になっている。しかし、本実施形態の遊技台によれば、上記の乱数生成回路 318 によって抽選に用いられる乱数の異常を把握したうえで対処することができるため、乱数の狙いうちを抑止して遊技の公平性を担保することができる。なお、周波数監視回路 3182 および乱数監視回路 3184 は内部情報レジスタ 3101 に記憶されている情報に関わらず継続してそれぞれ監視を行っている。

#### 【0645】

ここで、内部情報レジスタ 3101 の値が、乱数生成回路 318 により生成される乱数が一巡する周期（乱数生成範囲内の全ての値を各 1 回出力するのに必要な期間、以下、乱数一巡周期と称する）だけ保持され、その後クリアされる構成であった場合の問題点について図 87 を用いて説明する。同図は、乱数が一巡する周期とタイマ割込みの周期を比較した図である。図 87 の一番上には、一番下に示すタイマ割込み周期  $t_4$  よりも長い乱数一巡周期  $t_1$  が示されている。また、図 87 の上から二番目および三番目には、タイマ割込み周期  $t_4$  よりも短い乱数一巡周期  $t_2$  および  $t_3$  がそれぞれ示されている。

#### 【0646】

上述したように内部情報レジスタ 3101 の内容はタイマ割込み毎に確認される。例えば乱数一巡周期が、図 87 に示す乱数一巡周期  $t_1$  のようにタイマ割込み周期  $t_4$  よりも長い場合、内部情報レジスタ 3101 に保持された値がクリアされる前にこの値を読み取ることができる。しかし、本実施形態では、乱数生成範囲の最大値を設定することができるため、これに従って乱数一巡周期が短くなってしまう事態が生じ得る。例えば乱数一巡周期が、図 87 に示す乱数一巡周期  $t_2$  および  $t_3$  のようにタイマ割込み周期  $t_4$  よりも短い場合、内部情報レジスタ 3101 に保持された値を読み取る前にこの値がクリアされる場合がある。すなわち、CPU 304 が異常を示す情報を取得することが出来なくなる場合がある。

#### 【0647】

本実施形態では、内部情報レジスタ 3101 の値が一旦セットされると、この値が読みとられるまでその値が保持される。すなわち、正常に戻ったか否かにかかわらずこの値が維持される。さらに、CPU 304 がこの値を読み取ると、読み取られた部分の値がクリアされるようになっている。このため、上記のように乱数一巡周期が変わってしまったとしても、内部情報レジスタ 3101 から異常を示す情報を取得することができる。なお、異常を示す値が読み取られた後に、異常が継続していれば再度異常を示す値がセットされる。

#### 【0648】

＜主制御部メイン処理＞

次に、図 88 を用いて、図 4 に示す主制御部 300 の CPU 304 が実行する主制御部メイン処理について説明する。なお、同図は主制御部メイン処理の流れを示すフローチャートである。

#### 【0649】

この主制御部メイン処理は上述のユーザモードにおける処理に相当し、システムリセットがかかった場合でも、ユーザリセットがかかった場合でも実行される処理である。図 4 に示す主制御部 300 の RAM 308 には、大当たり用特図 1 乱数カウンタ、小当たり用特図

10

20

30

40

50

1 乱数カウンタ、ハズレ用特図 1 乱数カウンタ、およびこれらのカウンタの特図 2 用のカウンタが設けられている。また、その RAM 308 には、特図 1 の保留数、特図 1 当選乱数、大当り用特図 1 乱数、小当り用特図 1 乱数、ハズレ用特図 1 乱数、特図 1 当否判定結果、特図 1 決定結果、特図 1 変動時間、およびこれらの、保留数や乱数や結果の特図 2 用のものがそれぞれ記憶される。また RAM 308 には、当否判定（抽選）の開始を保留することができる最大数（この例では 4 つ）の領域に区分けされた保留記憶部が特図 1 と特図 2 で別々に用意されている。特図 1 の保留記憶部には、後述するように、特図 1 当選乱数、大当り用特図 1 乱数、小当り用特図 1 乱数、ハズレ用特図 1 乱数、および特図 1 変動時間決定用乱数の 5 つの乱数を 1 セットにしてこれら 5 つの乱数が入賞順（保留順）に 1 セットずつ 1 領域ごとに格納される。

10

#### 【0650】

上述したように、図 4 に示す主制御部 300 には、電源が投入されると起動信号（リセット信号）を出力する起動信号出力回路（リセット信号出力回路）340 を設けている。この起動信号を入力した基本回路 302 の CPU 304 は、リセット割込によりリセットスタートして ROM 306 に予め記憶している制御プログラムに従って図 88 に示す主制御部メイン処理を実行する。

#### 【0651】

ステップ S 3001 では、初期設定 1 を行う。この初期設定 1 では、CPU 304 のスタックポインタ（SP）へのスタック初期値の設定（仮設定）、割込マスクの設定、I/O 310 の初期設定、RAM 308 に記憶する各種変数の初期設定等を行う。

20

#### 【0652】

ステップ S 3003 では、低電圧信号がオンであるか否か、すなわち、電圧監視回路 338 が、電源制御部 660 が第 2 副制御部 500 を介して主制御部 300 に供給している電源の電圧値が所定の値（本実施形態では 9 V）未満である場合に電圧が低下したことを示す低電圧信号を出力しているか否かを監視する。そして、低電圧信号がオンの場合（CPU 304 が電源の遮断を検知した場合）には繰り返しこのステップ S 3003 を実行し、低電圧信号がオフの場合（CPU 304 が電源の遮断を検知していない場合）にはステップ S 3005 に進む。なお、電源が投入された直後で未だ上記所定の値（9 V）に達しない場合にも、供給電圧がその所定の値以上になるまでステップ S 3003 は繰り返し実行される。ステップ S 3005 では、初期設定 2 を行う。

30

#### 【0653】

図 89 は、主制御部メイン処理における初期設定 2 の流れを示すフローチャートである。まず、ステップ S 3051 では、カウンタ回路 312 に最大値や更新ソースを決める数値を設定する処理等、カウンタ回路 312 に関する設定処理を行う。なお、後述する主制御部タイマ割込処理を定期毎に実行するための周期を決める数値をタイマ回路 311 に設定する処理等、タイマ回路 311 に関する設定処理も併せて行う。ステップ S 3052 では、第 1 副制御部 400 への出力ポートからクリア信号を出力する処理を行い、ステップ S 3053 に進む。ステップ S 3053 では乱数生成回路初期設定処理を行い、次いで、ステップ S 3054 で RAM 308 への書き込みを許可する設定を行って、この初期設定 2 は終了になる。

40

#### 【0654】

図 90 は、ステップ S 3053 における乱数生成回路初期設定処理の流れを示すフローチャートである。この乱数生成回路初期設定処理は、上述のユーザモードにおいて行われる、図 77 に示す乱数生成回路 318 の初期設定処理である。まず、乱数の取り得る範囲（乱数生成範囲）範囲の設定を行い（ステップ S 3053 a）、ステップ S 3053 b に進む。上述のごとく、図 77 に示す乱数生成回路 318 では、生成する乱数の最大幅は 0 ～ 65535 であり、この最大幅が乱数生成範囲のデフォルトになり、ここでは、図 79 ～ 図 82 を用いて詳細に説明したように、その乱数生成範囲をデフォルトとは異なる範囲に設定することができる。図 88 に示す主制御部メイン処理は、リセット（システムリセットまたはユーザリセット）がかかる度に実行される処理であるため、乱数生成範囲の設

50

定もリセットがかかる度に実行される。このようにすることで、図 9 8、図 9 9 を用いて後述する乱数生成範囲の異常による問題を防止することができる。

#### 【0655】

ステップ S 3 0 5 3 b では、図 7 7 に示す乱数レジスタ 3 1 8 8 の読み出しを行うとともに、読み出した乱数を破棄する処理を行い、ステップ S 3 0 5 3 c に進む。この処理によって、乱数生成範囲のリセットがかかるごとに乱数レジスタ 3 1 8 8 が乱数をラッチすることを許容する許容状態になる。この許容状態においては、仮に異常の可能性がある乱数が乱数レジスタに残っている場合でもすぐに乱数を更新させることができるため、異常の可能性がある乱数の使用を防止することができる。

#### 【0656】

本実施形態では、乱数生成回路 3 1 8 では、特図当選乱数の元になる乱数と普図当選乱数とを生成するが、制御状態に応じて出力チャンネルが異なる。すなわち、特図当選乱数の元になる乱数は、非確率変動中（特図低確率状態）にはチャンネル C H 1 から出力され、確率変動中（特図高確率状態）にはチャンネル C H 2 から出力される。また、普図当選乱数は、普図低確率状態（非電サボ中）にはチャンネル C H 3 から出力され、普図高確率状態（電サボ中）にはチャンネル C H 4 から出力される。なお、乱数生成回路 3 1 8 は、特図当選乱数そのものを生成するものであってもよいし、特図当選乱数の元になる乱数を生成するものであってもよい。また、普図当選乱数そのものを生成するものであってもよいし、普図当選乱数の元になる乱数を生成するものであってもよい。ステップ S 3 0 5 3 c では、これら 4 チャンネル総てのチャンネルについてステップ S 3 0 5 3 a とステップ S 3 0 5 3 b が実行されたか否かを判定し、全チャンネルについて終了していなければ、ステップ S 3 0 5 3 a に戻り、1 チャンネルごとに未了のチャンネルについて処理を行う。なお、乱数生成範囲の設定は、状態に応じて使用されているチャンネルのみを実行してもよいし、乱数生成範囲を設定されるチャンネルのみ実行するようにしてもよい。一方、全チャンネルについて終了していれば、この乱数生成回路初期設定処理は終了になる。

#### 【0657】

なお、主制御部のタイマ割込みはこの時点では禁止されており、乱数生成回路初期設定処理はタイマ割込み許可前に行っていることになるため、新たに乱数がラッチされたとしても各種抽選に使用されることはなく、抽選処理の安定化を図ることができる。

#### 【0658】

ところで、本実施例では、システムリセットが発生した（乱数の更新が停止されている）場合、および例えばユーザリセットが発生した場合のようにユーザプログラムが最初から実行される（乱数の更新が行われている）場合のいずれの場合であっても必ず乱数回路初期化処理が実行されるように構成されている。必ずしもこのように構成する必要はないが、このように構成することで、ユーザプログラムの解析結果と乱数生成回路 3 1 8 の動作が合わなくなるため、基本回路 3 0 2 の動作の解析が困難になり、不正を防止することができる。従って、不正により基本回路 3 0 2 の動作が不安定にされることがなく、遊技制御の安定化を図ることができる。仮に、不正により基本回路 3 0 2 の動作が解析された場合、遊技者に有利な操作が行われてしまう虞があるが、上記の構成であればこのような不正が防止できるため、遊技制御の安定化を図ることができる。

#### 【0659】

さらに、乱数生成回路 3 1 8 が乱数更新を行うか否かに関わらず、C P U 3 0 4 に乱数生成回路初期設定処理を実行する指示を行わせた場合、ユーザリセットが実行された場合の処理を統一することができる。すなわち、ユーザリセット後に不安定な状態となった場合にも分岐処理を行わずにすむため、遊技制御の安定化を図ることができる。なお、上述した内容はユーザリセットがかかった場合について述べたが、システムリセットがかかった場合についても同様の効果を得ることができる。

なお、上述した通り、本実施例では、システムリセットが発生した場合と、ユーザリセット等が発生した場合とで乱数生成回路初期設定処理の処理内容が同一となっているが、この処理内容をシステムリセットが発生した場合と、ユーザリセット等が発生した場合と

10

20

30

40

50

で異ならせるようにしてもよい。具体的には、ユーザリセットが発生した場合には、最大値を設定する処理（最大値設定レジスタにROM 306に記憶されている対応CHの最大値をライトする処理）を実行する一方で、ユーザリセット等が発生した時には、最大値を確認する処理（最大値設定レジスタの値をリードしてROM 306に記憶されている対応するCHの最大値と比較する処理）を行うとともに、最大値が一致しなかった場合には所定の処理（例えば、遊技の進行を停止させる処理、外部に異常を示す信号を送信する処理、異常を報知する処理、異常が発生したことを記憶する処理など）を実行するように構成してもよい。なお、システムリセットが発生したのかユーザリセット等が発生したのかを識別方法としては、内部情報レジスタ（詳細は後述）に記憶されている情報を参照するようにすればよい。

10

#### 【0660】

図88に示す主制御部メイン処理におけるステップS3007では、電源の遮断前（電断前）の状態に復帰するか否かの判定を行い、電断前の状態に復帰しない場合（主制御部300の基本回路302を初期状態にする場合）にはRWMクリア処理（ステップS3013）に進む。

#### 【0661】

具体的には、最初に、図2に示す電源基板182に設けたRAMクリアスイッチ180を遊技店の店員などが操作した場合に送信されるRAMクリア信号がオン（操作があったことを示す）であるか否か、すなわちRAMクリアが必要であるか否かを判定し、RAMクリア信号がオンの場合（RAMクリアが必要な場合）には、基本回路302を初期状態にすべくステップS3013に進む。一方、RAMクリア信号がオフの場合（RAMクリアが必要でない場合）には、RAM 308に設けた電源ステータス記憶領域に記憶した電源ステータスの情報を読み出し、この電源ステータスの情報がサスペンドを示す情報であるか否かを判定する。そして、電源ステータスの情報がサスペンドを示す情報でない場合には、基本回路302を初期状態にすべくステップS3013に進み、電源ステータスの情報がサスペンドを示す情報である場合には、RAM 308の所定の領域（例えば全ての領域）に記憶している1バイトデータを初期値が0である1バイト構成のレジスタに全て加算することによりチェックサムを算出し、算出したチェックサムの結果が、電断前にRAM 308に設定した値と一致するか否か（チェックサムの結果が正常であるか否か）を判定する。そして、チェックサムの結果が特定の値の場合（チェックサムの結果が正常である場合）には電断前の状態に復帰すべくステップS3009に進み、チェックサムの結果が特定の値以外である場合（チェックサムの結果が異常である場合）には、パチンコ機100を初期状態にすべくステップS3013に進む。同様に電源ステータスの情報が「サスペンド」以外の情報を示している場合にもステップS3013に進む。

20

30

#### 【0662】

ステップS3009では、データ書き戻し処理を行う。このデータ書き戻し処理では、電断時にRAM 308に設けられたスタックポインタ退避領域に記憶しておいたスタックポインタの値を読み出し、スタックポインタに再設定（本設定）する。また、電断時にRAM 308に設けられたレジスタ退避領域に記憶しておいた各レジスタの値を読み出し、各レジスタに再設定した後、割込許可の設定を行う。以降、CPU 304が、再設定後のスタックポインタやレジスタに基づいて制御プログラムを実行する結果、パチンコ機100は電源断時の状態に復帰する。すなわち、電断直前にタイマ割込処理（後述）に分岐する直前に行った命令の次の命令から処理を再開する。また、図4に示す主制御部300における基本回路302に搭載されているRAM 308には、送信情報記憶領域が設けられている。このステップS3009では、その送信情報記憶領域に、復電コマンドをセットする。この復電コマンドは、電源断時の状態に復帰したことを表すコマンドであり、後述する、主制御部300のタイマ割込処理におけるステップS231において、第1副制御部400へ送信される。

40

#### 【0663】

ステップS3011では、WDT 3141を起動させる処理を行う。ここでは、WDT

50

3 1 4 1 の起動許可及び初期値の設定等を行う。なお、本実施形態では、W D T 3 1 4 1 に、初期値として 3 2 . 8 m s に相当する数値を設定する。

#### 【0 6 6 4】

ステップ S 3 0 1 3 では、R W M クリア処理を行う。この R W M クリア処理では、R A M 3 0 8 の全ての記憶領域の初期化を行う。また、主制御部のタイマ割込み許可の設定、スタックポインタへのスタック初期値の設定（本設定）なども併せて行う。さらにここで、主制御部 3 0 0 の R A M 3 0 8 に設けられた送信情報記憶領域に正常復帰コマンドをセットする。この正常復帰コマンドは、主制御部 3 0 0 の R W M クリア処理（ステップ S 3 0 1 3）が行われたことを表すコマンドであり、復電コマンドと同じく、主制御部 3 0 0 のタイマ割込処理におけるステップ S 2 3 1 において、第 1 副制御部 4 0 0 へ送信される。ステップ S 3 0 1 5 では、ステップ S 3 0 1 1 と同様、W D T 3 1 4 1 を起動させる処理を行う。

なお、本実施例では、少なくともタイマ割込み許可の設定を行った後の主制御部メイン処理において W D T 3 1 4 1 のリスタートを行わず、後述する主制御部タイマ割込処理でのみ W D T 3 1 4 1 のリスタート（詳細は後述）を行うように構成するとともに、主制御部のタイマ割込み許可の設定後に W D T 3 1 4 1 を起動させる処理を行うように構成している。このように構成することで、正常に主制御部のタイマ割込処理が実行されている場合にのみ W D T 3 1 4 1 のカウント値が初期設定値（本実施形態では 3 2 . 8 m s）を超えることなく（W D T 3 1 4 1 割込が発生せずに）、その後の処理が行われるようになる。特に、主制御部メイン処理で乱数生成回路から乱数を取得（ラッチされた乱数の取得のみ、または乱数のラッチおよびラッチされた乱数の取得）する遊技台においては、主制御部に異常がある場合（主制御部タイマ割込処理が正常に実行されていない場合）に乱数が取得されてしまうことを防止することができる。

#### 【0 6 6 5】

ステップ S 3 0 1 7 では、基本乱数初期値更新処理を行う。ここにいう基本乱数とは、ソフトウェア乱数である、大当り用特図乱数、小当り用特図乱数、およびハズレ用特図乱数が相当する。なお、各乱数には特図 1 用の乱数と特図 2 用の乱数が存在するが、以降の説明では、特に断りを入れない限り両者を区別することなく単に特図として説明する。この基本乱数初期値更新処理では、大当り用特図乱数カウンタ、小当り用特図乱数カウンタ、およびハズレ用特図乱数カウンタそれぞれの初期値を生成するための初期値生成用乱数カウンタを更新する。各カウンタが R A M 3 0 8 に設けられている。なお、初期値生成用乱数カウンタは、後述するステップ S 2 0 4 でも更新する。

#### 【0 6 6 6】

ステップ S 3 0 1 9 では演出乱数更新処理を行う。ここにいう演出乱数もソフトウェア乱数であって、この演出乱数は、演出を決定する元になる乱数のことであり、本実施形態では、後述する先読み予告を実行するか否かを抽選する際に用いられる乱数等が相当する。この演出乱数更新処理では、R A M 3 0 8 に設けられた演出乱数カウンタを更新する。なお、演出乱数カウンタは、後述するステップ S 2 1 1 でも更新する。

#### 【0 6 6 7】

主制御部 3 0 0 は、所定の周期ごとに開始するタイマ割込処理を行っている間を除いて、ステップ S 3 0 1 7 およびステップ S 3 0 1 9 の処理を繰り返し実行する。これらのステップ S 3 0 1 7 およびステップ S 3 0 1 9 は、主処理の一例に相当する。

#### 【0 6 6 8】

＜主制御部タイマ割込処理＞

次に、図 9 1 を用いて、主制御部 3 0 0 の C P U 3 0 4 が実行する主制御部タイマ割込処理について説明する。なお、同図は主制御部タイマ割込処理の流れを示すフローチャートである。

#### 【0 6 6 9】

図 4 に示す主制御部 3 0 0 は、所定の周期（本実施形態では約 4 m s に 1 回）でタイマ割込信号を発生するタイマ回路 3 1 1 を備えており、このタイマ割込信号を契機として主

制御部タイマ割込処理を所定の周期で開始する。

【0670】

ステップS3201では、タイマ割込スタート処理を行う。このタイマ割込スタート処理では、CPU304の各レジスタの値をスタック領域に一時的に退避する処理などを行う。ステップS3203では、WDT3141のカウント値が初期設定値（本実施形態では32.8ms）を超えてWDT3141割込が発生しないように（処理の異常を検出しないように）、WDT3141を定期的に（本実施形態では、主制御部タイマ割込の周期である約4msに1回）リスタートを行う。

【0671】

ステップS3205では、入力ポート状態更新処理を行う。この入力ポート状態更新処理では、I/O310の入力ポートを介して、各種の球検出センサを含む図4に示す各種センサ320の検出信号を入力して検出信号の有無を監視し、RAM308に各種センサ320ごとに区画して設けた信号状態記憶領域に記憶する。球検出センサの検出信号を例にして説明すれば、前回のタイマ割込処理（約8ms前）で検出した各々の球検出センサの検出信号の有無の情報を、RAM308に各々の球検出センサごとに区画して設けた前回検出信号記憶領域から読み出し、この情報をRAM308に各々の球検出センサごとに区画して設けた前々回検出信号記憶領域に記憶し、前回のタイマ割込処理（約4ms前）で検出した各々の球検出センサの検出信号の有無の情報を、RAM308に各々の球検出センサごとに区画して設けた今回検出信号記憶領域から読み出し、この情報を上述の前回検出信号記憶領域に記憶する。また、今回検出した各々の球検出センサの検出信号を、上述の今回検出信号記憶領域に記憶する。

【0672】

また、ステップS3205では、上述の前々回検出信号記憶領域、前回検出信号記憶領域、および今回検出信号記憶領域の各記憶領域に記憶した各々の球検出センサの検出信号の有無の情報を比較し、各々の球検出センサにおける過去3回分の検出信号の有無の情報が入賞判定パターン情報と一致するか否かを判定する。一個の遊技球が一つの球検出センサを通過する間に、約4msという非常に短い間隔で起動を繰り返すこの主制御部タイマ割込処理は何回か起動する。このため、主制御部タイマ割込処理が起動する度に、上述のステップS3205では、同じ遊技球が同じ球検出センサを通過したことを表す検出信号を確認することになる。この結果、上述の前々回検出信号記憶領域、前回検出信号記憶領域、および今回検出信号記憶領域それぞれに、同じ遊技球が同じ球検出センサを通過したことを表す検出信号が記憶される。すなわち、遊技球が球検出センサを通過し始めたときには、前々回検出信号無し、前回検出信号有り、今回検出信号有りになる。本実施形態では、球検出センサの誤検出やノイズを考慮して、検出信号無しの後に検出信号が連続して2回記憶されている場合には、入賞があったと判定する。図4に示す主制御部300のROM306には、入賞判定パターン情報（本実施形態では、前々回検出信号無し、前回検出信号有り、今回検出信号有りであることを示す情報）が記憶されている。このステップS3205では、各々の球検出センサにおいて過去3回分の検出信号の有無の情報が、予め定めた入賞判定パターン情報（本実施形態では、前々回検出信号無し、前回検出信号有り、今回検出信号有りであることを示す情報）と一致した場合に、一般入賞口226、可変入賞口234、第1特図始動口230、および第2特図始動口232への入球、または普図始動口228の通過があったと判定する。すなわち、これらの入賞口234、230やこれらの始動口230、232、228への入賞があったと判定する。例えば、一般入賞口226への入球を検出する一般入賞口センサにおいて過去3回分の検出信号の有無の情報が上述の入賞判定パターン情報と一致した場合には、一般入賞口226へ入賞があったと判定し、以降の一般入賞口226への入賞に伴う処理を行うが、過去3回分の検出信号の有無の情報が上述の入賞判定パターン情報と一致しなかった場合には、以降の一般入賞口226への入賞に伴う処理を行わずに後続の処理に分岐する。なお、主制御部300のROM306には、入賞判定クリアパターン情報（本実施形態では、前々回検出信号有り、前回検出信号無し、今回検出信号無しであることを示す情報）が記憶されている。入賞が

10

20

30

40

50

一度あったと判定した後は、各々の球検出センサにおいて過去3回分の検出信号の有無の情報が、その入賞判定クリアパターン情報に一致するまで入賞があったとは判定せず、入賞判定クリアパターン情報に一致すれば、次からは上記入賞判定パターン情報に一致するか否かの判定を行う。なお、本実施形態では、一般入賞口226、可変入賞口234、第1特図始動口230、第2特図始動口232、普図始動口228へ入賞があったか否かの判定として、前々回、前回および今回の検出信号が予め定められた入賞判定パターン情報と一致した否かを判定していたが、このような方法に限らず、例えば、前回および今回の検出信号が予め定められた入賞判定パターン情報（例えば、前回検出信号なし、今回検出信号あり（いわゆるアップエッジ））と一致したか否かを判定するように構成してもよい。すなわち、球検出センサからの検出信号の有無を監視し、この監視結果の履歴が予め定められたパターンと一致したか否かを判定するように構成すればよい。

#### 【0673】

ステップS3207およびステップS3209では、基本乱数初期値更新処理および基本乱数更新処理を行う。これらの基本乱数初期値更新処理および基本乱数更新処理では、上記ステップS3017で行った初期値生成用乱数カウンタの値の更新を行い、次に主制御部300で使用する、大当り用特図乱数、小当り用特図乱数、およびハズレ用特図乱数それぞれを生成するための乱数カウンタを更新する。例えば、大当り用特図乱数として取り得る数値範囲が0～100とすると、RAM308に設けた大当り用特図乱数を生成するための乱数カウンタ記憶領域から値を取得し、取得した値に1を加算してから元の乱数カウンタ記憶領域に記憶する。このとき、取得した値に1を加算した結果が101であれば0を元の乱数カウンタ記憶領域に記憶する。また、取得した値に1を加算した結果、乱数カウンタが一周していると判定した場合にはそれぞれの乱数カウンタに対応する初期値生成用乱数カウンタの値を取得し、乱数カウンタの記憶領域にセットする。例えば、0～100の数値範囲で変動する大当り用特図乱数生成用の乱数カウンタから値を取得し、取得した値に1を加算した結果が、RAM308に設けた所定の初期値記憶領域に記憶している前回設定した初期値と等しい値（例えば7）である場合に、大当り用特図乱数生成用の乱数カウンタに対応する初期値生成用乱数カウンタから値を初期値として取得し、大当り用特図乱数生成用の乱数カウンタにセットすると共に、大当り用特図乱数生成用の乱数カウンタが次に1周したことを判定するために、今回設定した初期値を上述の初期値記憶領域に記憶しておく。なお、本実施形態では特図1に関する乱数を取得するためのカウンタと特図2に関する乱数を取得するためのカウンタとを別々に設けているが、同一のカウンタを用いてもよい。

#### 【0674】

ステップS3211では、演出乱数更新処理を行う。この演出乱数更新処理では、上記ステップS3019と同様に、主制御部300で使用する演出用乱数を生成するための乱数カウンタを更新する。ステップS3213では、タイマ更新処理を行う。このタイマ更新処理では、普通図柄表示装置210に図柄を変動・停止表示する時間を計時するための普図表示図柄更新タイマ、第1特図表示装置212に図柄を変動・停止表示する時間を計時するための特図1表示図柄更新タイマ、第2特図表示装置214に図柄を変動・停止表示する時間を計時するための特図2表示図柄更新タイマ、所定の入賞演出時間、所定の開放時間、所定の閉鎖時間、所定の終了演出期間などを計時するためのタイマなどを含む各種タイマを更新する。

#### 【0675】

ステップS3215では、入賞口カウンタ更新処理を行う。この入賞口カウンタ更新処理では、入賞口234、230や始動口230、232、228に入賞があった場合に、RAM308に各入賞口ごと、あるいは各始動口ごとに設けた賞球数記憶領域の値を読み出し、1を加算して、元の賞球数記憶領域に設定する。

#### 【0676】

また、ステップS3217では、入賞受付処理を行う。この入賞受付処理では、第1特図始動口230に入賞があり、且つ、保留している特図1変動遊技の数が所定数（本実施

形態では4)未満である場合には、所定の始動情報を取得する。すなわち、保留数が所定数未満であれば、図4に示す乱数生成回路318から、特図1当選乱数の元になるハードウェア乱数を得、加工を施すことによって特図1当選乱数を取得する。この特図1当選乱数を取得する処理をより具体的に説明すると、乱数生成回路318の特図1に対応するチャンネルの乱数をラッチするためのラッチ信号を出力する処理(具体的には、ソフトウェアレジスタ3186の特図1のチャンネルに対応する領域に所定の値を書き込む処理)を実行するとともに、このラッチ信号によってラッチされた乱数を取得するためのリード信号を出力する処理(具体的には、乱数レジスタ3188のうちの特図1のチャンネルに対応する乱数レジスタに対してリード信号を出力する処理)を実行してラッチされている乱数を取得するとともに、取得した乱数を加工(例えば、ユーザプログラムで生成しているソフトウェア乱数を加算する処理)する処理である。なお、第2特図始動口232に入賞があった場合も同様の処理を実行する。なお、本実施形態では、乱数生成回路318から乱数を取得する方法として特図1および2に対応するチャンネルの乱数をラッチするためのラッチ信号を出力する処理を行った後にラッチした乱数を取得するリード信号を出力する方法を採用しているが、特図1および2に対応するチャンネルの乱数を直読みする処理(具体的には、乱数レジスタ3188の値を直接リードする処理)を実行して取得するようにしてもよい。さらに、本実施形態では、特図1および2のラッチされた乱数を取得するためのリード信号を出力する処理(乱数レジスタ3188の値を直接リードする処理とは異なる)を主制御部タイマ割込み処理において実行するように構成しているが、この処理に関しては上述した主制御部メイン処理において実行するように構成してもよく、この場合は、後述する特図関連抽選処理を合わせて主制御部メイン処理において実行するように構成すればよく、特図1および2に対応するチャンネルの乱数値を直読みする処理を実行する場合においては、特図関連抽選処理のみを主制御部メイン処理において実行するように構成すればよい。また、本実施形態では、特図1当選乱数および特図2当選乱数それぞれを、乱数生成回路318から取得した乱数を加工して導出しているが、このような場合に限らず、乱数生成回路318から取得した乱数を特図1当選乱数および特図2当選乱数としてもよい。また、RAM308に設けた乱数カウンタ記憶領域から、大当り用特図1乱数、小当り用特図1乱数、およびハズレ用特図1乱数を取得する。大当り用特図1乱数、小当り用特図1乱数、およびハズレ用特図1乱数は、RAM308に設けられたソフトウェア乱数カウンタから導出されたソフトウェア乱数を加工した値(ソフトウェア乱数の値+Rレジスタの値+1)である。さらに、図4に示すカウンタ回路312から特図1変動時間決定用乱数を取得する。図4に示す乱数生成回路318、カウンタ回路312、RAM308に設けられたソフトウェア乱数カウンタ、および乱数加工を施す主制御部300を併せたものが、始動情報を生成して導出するものであり、始動情報導出手段(第1の始動情報導出手段、第2の始動情報導出手段)の一例に相当する。ここで取得された各種乱数(始動情報)は、RAM308に設けた特図1の保留記憶部の、入賞順(保留順)に応じた空いている領域に、1セットの始動情報として記憶される。この特図1の保留記憶部は、第1特図始動口230(第1の始動領域)に遊技球が進入した場合に取得した始動情報を所定の第1上限個数(ここでは4個)まで記憶可能な第1の始動情報記憶手段に相当する。このとき各種乱数(始動情報)をRAM308に設けた一時領域に一旦記憶し、その一時領域に記憶された値を特図1の保留記憶部に記憶してもよく、この場合、一時領域を第1の始動情報記憶手段としてもよいし、特図1の保留記憶部および一時領域を第1の始動情報記憶手段としてもよい。また、主制御部300のCPU304は、RAM308に記憶されている特図1の保留数の値に1を加算し、特図1の保留数が1増加する。したがって、主制御部300のCPU304が保留手段の一例に相当する。また、特図2についても、特図1と同様に始動情報である各乱数を取得し、取得した乱数をRAM308に設けた特図2の保留記憶部に、1セットの始動情報として同様に記憶され、さらに、RAM308に記憶されている特図2の保留数の値に1を加算する。特図2の保留記憶部は、第2特図始動口232(第2の始動領域)に遊技球が進入した場合に取得した始動情報を所定の第2上限個数(ここでは4個)まで記憶可能な第2の始動情報記憶手段に相当す

10

20

30

40

50



る。このとき各種乱数（始動情報）をRAM308に設けた一時領域に一旦記憶し、その一時領域に記憶された値を特図2の保留記憶部に記憶してもよく、この場合一時領域を第2の始動情報記憶手段としてもよいし、特図2の保留記憶部および一時領域を第2の始動情報記憶手段としてもよい。

#### 【0677】

また、普図始動口228を球が通過したことを検出し、且つ、保留している普図変動遊技の数が所定数（本実施形態では4）未満の場合には、そのタイミングで、図4に示す乱数生成回路318から普図当選乱数を所得し、RAM308に設けた特図用とは別の乱数記憶領域に記憶する。この乱数生成回路318から普図当選乱数を取得する点についても、さらに後述する。

10

#### 【0678】

また、この入賞受付処理では、所定の球検出センサにより、第1特図始動口230、第2特図始動口232、普図始動口228、または可変入賞口234の入賞（入球）を検出した場合に、第1副制御部400に送信すべき送信情報に、第1特図始動口230、第2特図始動口232、普図始動口228、および可変入賞口234の入賞（入球）の有無を示す入賞受付情報を設定する。なお、特図の始動情報にしても普図の始動情報にしても、保留数がそれぞれの所定数以上であれば始動情報を取得せずに、ステップS3219に進む。

#### 【0679】

ステップS3219では、払出要求数送信処理を行う。図4に示す払出制御部600に出力する出力予定情報および払出要求情報は1バイトで構成しており、ビット7にストロープ情報（オンの場合、データをセットしていることを示す）、ビット6に電源投入情報（オンの場合、電源投入後一回目のコマンド送信であることを示す）、ビット4～5に暗号化のための今回加工種別（0～3）、およびビット0～3に暗号化加工後の払出要求数を示すようにしている。

20

#### 【0680】

ステップS3221では、普図状態更新処理を行う。この普図状態更新処理は、普図の状態に対応する複数の処理のうちの1つの処理を行う。例えば、普図変動表示の途中（上述する普図表示図柄更新タイマの値が1以上）における普図状態更新処理では、普通図柄表示装置210を構成する7セグメントLEDの点灯と消灯を繰り返す点灯・消灯駆動制御を行う。この制御を行うことで、普通図柄表示装置210は普図の変動表示（普図変動遊技）を行う。

30

#### 【0681】

また、普図変動表示時間が経過したタイミング（普図表示図柄更新タイマの値が1から0になったタイミング）における普図状態更新処理では、当りフラグがオンの場合には、当り図柄の表示態様となるように普通図柄表示装置210を構成する7セグメントLEDの点灯・消灯駆動制御を行い、当りフラグがオフの場合には、ハズレ図柄の表示態様となるように普通図柄表示装置210を構成する7セグメントLEDの点灯・消灯駆動制御を行う。また、主制御部300のRAM308には、普図状態更新処理に限らず各種の処理において各種の設定を行う設定領域が用意されている。ここでは、上記点灯・消灯駆動制御を行うとともに、その設定領域に普図停止表示中であることを示す設定を行う。この制御を行うことで、普通図柄表示装置210は、当り図柄（図5（c）に示す普図A）およびハズレ図柄（図5（c）に示す普図B）いずれか一方の図柄の確定表示を行う。さらにその後、所定の停止表示期間（例えば500m秒間）、その表示を維持するためにRAM308に設けた普図停止時間管理用タイマの記憶領域に停止期間を示す情報を設定する。この設定により、確定表示された図柄が所定期間停止表示され、普図変動遊技の結果が遊技者に報知される。

40

#### 【0682】

また、普図変動遊技の結果が当りであれば、後述するように、普図当りフラグがオンされる。この普図当りフラグがオンの場合には、所定の停止表示期間が終了したタイミング

50

（普図停止時間管理用タイマの値が1から0になったタイミング）における普図状態更新処理では、RAM 308の設定領域に普図作動中を設定するとともに、所定の開放期間（例えば2秒間）、第2特図始動口232の羽根部材2321の開閉駆動用のソレノイド（332）に、羽根部材2321を開放状態に保持する信号を出力するとともに、RAM 308に設けた羽根開放時間管理用タイマの記憶領域に開放期間を示す情報を設定する。このようにして一对の羽根部材2321の開放制御を行う主制御部300のCPU 304が、可変始動領域制御を行う可変始動領域制御手段の一例に相当する。一方、非電サポ状態であれば、RAM 308の設定領域に普図非作動中を設定するとともに、第2特図始動口232の羽根部材2321の開閉駆動用のソレノイド（332）には、何ら信号を出力しない。こうすることで、羽根部材2321は閉じた状態のままになる。なお、羽根部材2321を閉じた状態に維持するための信号を必ず出力するようにしてもよい。

10

#### 【0683】

また、電サポ状態であった場合には、所定の開放期間が終了したタイミング（羽根開放時間管理用タイマの値が1から0になったタイミング）で開始する処理では、所定の閉鎖期間（例えば0.1秒間）、羽根部材2321の開閉駆動用のソレノイド（332）に、羽根部材2321を閉鎖状態に保持する信号を出力するとともに、RAM 308に設けた羽根閉鎖時間管理用タイマの記憶領域に閉鎖期間を示す情報を設定する。

#### 【0684】

また、電サポ状態であった場合には、所定の閉鎖期間が終了したタイミング（羽根閉鎖時間管理用タイマの値が1から0になったタイミング）で開始する普図状態更新処理において、RAM 308の設定領域に普図非作動中を設定する。さらに、普図変動遊技の結果がハズレであれば、後述するように、普図ハズレフラグがオンされる。この普図ハズレフラグがオンの場合には、上述した所定の停止表示期間が終了したタイミング（普図停止時間管理用タイマの値が1から0になったタイミング）における普図状態更新処理でも、RAM 308の設定領域に普図非作動中を設定する。普図非作動中の場合における普図状態更新処理では、何もせずに次のステップS3223に移行するようにしている。続いて、ステップS3223では普図関連抽選処理を実行する。

20

#### 【0685】

図92（a）は、普図関連抽選処理の流れを示すフローチャートである。図92（a）に示す普図関連抽選処理では、まず、普図保留情報があるか否かを判定する（ステップS3223a）。ここにいう普図保留情報とは、普図の保留数を指す。すなわち、ここでは、保留している普図変動遊技の数が1以上であるか否かを判定する。なお、普図の保留数をデータとして持っていなくとも、例えば、保留に対応した乱数（普図当選乱数）を普図保留情報として認識するようにしても良い。普図保留情報がなければ、この普図関連抽選処理は終了になり、普図保留情報があればステップS3223bに進む。ステップS3223bでは、普図変動遊技が行われているか否かを判定し、行われていればこの普図関連抽選処理は終了になり、行われていなければステップS3223cに進む。ステップS3223cでは、第2特図始動口232の開閉制御が行われているか否か（普図作動中か否か）を判定し、普図作動中であればこの普図関連抽選処理は終了になり、普図非作動中であればステップS3223dに進む。ステップS3223dでは、上述の乱数記憶領域に記憶している普図当選乱数に基づいた乱数抽選を行う。

30

40

#### 【0686】

図92（b）は、普図抽選テーブルを示す図である。このテーブルは、主制御部300のROM 306に記憶されている。主制御部300のCPU 304は、RAM 308の乱数記憶領域から普図当選乱数を取り出し、時短フラグを参照し、時短フラグがオンであれば普図高確率状態（電サポ中）であるため、取得した普図当選乱数から普図高確率状態の普図当選データを引き、キャリアが発生した場合（普図当選乱数よりも普図当選データの値が大きい場合）は普図当選とし、キャリアが発生しなかった場合は普図ハズレになる。すなわち、普図当選乱数範囲は0～9になる。普図高確率状態では、普図当選乱数は、図77に示す乱数生成回路318のチャンネルCH4から出力される。このチャンネルCH

50

4 には、図 90 に示す乱数生成回路初期設定処理における乱数生成範囲の設定（ステップ S 3053a）において 0～9 の乱数生成範囲が設定されており、普図高確率状態の普図当選乱数の取り得る範囲は 0～9 になる。したがって、普図高確率状態における普図当選確率は 1 / 1 になる。一方、時短フラグがオフであれば普図低確率状態（非電サポ中）であるため、取得した普図当選乱数から、普図高確率状態のデータと同じ普図低確率状態の普図当選データを引き、キャリアが発生した場合（普図当選乱数よりも普図当選データの値が大きい場合）は普図当選とし、キャリアが発生しなかった場合は普図ハズレになる。すなわち、普図低確率状態の場合でも普図当選乱数範囲は 0～9 になる。普図低確率状態では、普図当選乱数は、図 77 に示す乱数生成回路 318 のチャンネル CH3 から出力される。このチャンネル CH3 には、乱数生成範囲の設定（ステップ S 3053a）において 0～999 の乱数生成範囲が設定されており、普図低確率状態の普図当選乱数の取り得る範囲は 0～999 になる。したがって、普図低確率状態における普図当選確率は 1 / 100 になる。

10

#### 【0687】

本実施形態では、普図当選乱数の取り得る範囲（乱数生成範囲）を、デフォルトの 0～65535 から 0～999 に制限することで大当り確率を、きりのよい 1 / 100 にすることができる。また、普図当選データを、普図高確率状態と普図低確率状態で同じ値にすることができる。乱数生成範囲を制限せずに確率を決定することは開発工数の増大を招くとともに誤った確率設計の原因となってしまう場合がある。特に、一の契機に基づいて複数回の抽選を行う場合に煩雑さはより顕著になってしまう。よって、乱数生成範囲を制限することは開発工数を削減するとともに、抽選処理の安定化を図ることができるといえる。また、普図当選データを、普図高確率状態と普図低確率状態で共通化することも、開発工数を削減するとともに、抽選処理の安定化を図ることができるといえる。

20

#### 【0688】

普図抽選に当選した場合には RAM 308 に設けた当りフラグをオンに設定する。ハズレ（不当選）の場合には、当りフラグをオフに設定する。また、普図抽選の結果に関わらず、図 4 に示すカウンタ回路 312 から普図変動時間決定用乱数を取得し、取得した普図変動時間決定用乱数に基づいて複数の変動時間のうちから普図表示装置 210 に普図を変動表示する時間を 1 つ選択し、この変動表示時間を、普図変動表示時間として、RAM 308 に設けた普図変動時間記憶領域に記憶する。普図変動時間決定用乱数値を用いた抽選で複数の変動時間のうちから普図変動時間を 1 つ選択する主制御部 300 の CPU 304 が、抽選手段の一例に相当する。ここで、普図変動時間の抽選処理は、普図の当りか否かを抽選する当否判定とは異なる。なお、保留している普図変動遊技の数は、RAM 308 に設けた普図保留数記憶領域に記憶するようにしており、ステップ S 3223e を実行するたびに、保留している普図変動遊技の数から 1 を減算した値を、この普図保留数記憶領域に記憶し直すようにしている。また、カウンタ回路 312 からの普図変動時間決定用乱数の取得は、普図始動口 228 への入賞時に行ってもよい。

30

#### 【0689】

ステップ S 3223e では、上述の乱数記憶領域から、先の普図抽選に使用した普図当選乱数を消去し、この普図関連抽選処理は終了になる。続いて、特図先読み処理（ステップ S 3225）が実行される。この先読み処理では、まず、RAM 308 に設けられた特図 1 の保留記憶部内の特図 1 当選乱数を先読みするか、あるいは特図 2 の保留記憶部内の特図 2 当選乱数を先読みする。なお、ここでの先読みとは始動情報を当否判定（本抽選）の前に先に読むことを意味するが、以降の先読み処理では、先読みという言葉を用いて、先（当否判定（本抽選）の結果）を読むという意味で使うことがある。このステップ S 3225 では、後述する特図関連処理（ステップ S 3229）で用いる図 94（a）に示す特図抽選テーブルの内容と同じ内容の事前判定用テーブルを用い、先読みした特図当選乱数に基づく当否判定の事前判定を行う。なお、特図関連処理では、特図抽選テーブルを用いて特図変動遊技の当否判定を改めて行い、ここでの判定結果は、あくまで事前判定の結果になる。当否判定の事前判定では、「大当り」という結果か、あるいは「大当り」以外と

40

50

いう結果が導出され、「大当り」という結果の場合には、RAM 308に設けられた特図1の保留記憶部内の大当り用特図1乱数を先読みするか、あるいは特図2の保留記憶部内の大当り用特図2乱数を先読みする。続いて、ステップS3229の特図関連処理で用いる図94(b)に示す停止図柄抽選テーブルの内容と同じ内容の事前判定用テーブルを用い、先読みした大当り用特図乱数に基づく特図の停止図柄の事前判定を行う。

#### 【0690】

なお、特図関連処理では、停止図柄抽選テーブルを用いて特図の停止図柄の抽選を改めて行い、ここでの判定結果は、あくまで事前判定の結果になる。こうして、特図の停止図柄を事前判定すると、先読み予告の実行可否抽選を行う。この先読み予告は、特図関連抽選処理（ステップS3229）が実行される前、すなわち当否判定が行われる前に、当該特図関連抽選処理で行われる当否判定の結果が大当り（ここではより限定して15Rの大当り（15R特別大当りか15R大当り））になることを予告するための報知である。ここでの先読み予告には、停止図柄の事前判定結果が15Rの大当り図柄（特図Aまたは特図B）でなくても、15Rの大当りになるかのように偽りで予告する偽の先読み予告も含まれる。すなわち、先読み予告は、当否判定の結果が15R大当りになる可能性があることを表したり、遊技者に示唆する事前報知、あるいは当否判定の結果が15R大当りになることを遊技者に期待させる事前報知であるといえる。先読み予告の実行可否抽選を行うタイミングで、RAM 308に設けられた演出乱数カウンタから演出乱数（例えば、取得する範囲は0～99）を取得し、取得した演出乱数に基づいて実行可否抽選を行う。なお、停止図柄の事前判定結果を第1副制御部400に送信し、この実行可否抽選は、第1副制御部400が行うようにしてもよい。

#### 【0691】

次に、特図1および特図2それぞれについての特図状態更新処理（ステップS3227）を行うが、最初に、特図2についての特図状態更新処理を行い、次いで、特図1についての特図状態更新処理を行う。特図2状態更新処理は、特図2の状態に応じて、次の8つの処理のうちの1つの処理を行う。例えば、特図2変動表示の途中（上述の特図2表示図柄更新タイマの値が1以上）における特図2状態更新処理では、第2特別図柄表示装置214を構成する7セグメントLEDの点灯と消灯を繰り返す点灯・消灯駆動制御を行う。この制御を行うことで、第2特別図柄表示装置214は特図2の変動表示（特図2変動遊技）を行う。また、コマンド設定送信処理（ステップS3231）で一般コマンド回転開始設定送信処理を実行させることを示す所定の送信情報を上述の送信情報記憶領域に追加記憶してから処理を終了する。

#### 【0692】

また、主制御部300のRAM 308には、15R大当りフラグ、2R大当りフラグ、第1小当りフラグ、第2小当りフラグ、第1ハズレフラグ、第2ハズレフラグ、確変フラグ、および時短フラグそれぞれのフラグが用意されている。特図2変動表示時間が経過したタイミング（特図2表示図柄更新タイマの値が1から0になったタイミング）で開始する特図2状態更新処理では、後述する特図関連抽選処理における特図決定結果（特図の停止図柄態様）に基づいて第2特図表示装置214を構成する7セグメントLEDの点灯・消灯駆動制御を行い、RAM 308の設定領域に特図2停止表示中であることを表す設定を行う。この制御を行うことで、第2特別図柄表示装置214は、15R特別大当り図柄（特図A）、15R大当り図柄（特図B）、2R特別大当り図柄（特図C）、突然時短図柄（特図D）、隠れ確変図柄（特図E）、突然通常図柄（特図F）、第1小当り図柄（特図G）、第2小当り図柄（特図H）、第1ハズレ図柄（特図I）、および第2ハズレ図柄（特図J）のいずれか一つの図柄の確定表示を行う。さらにその後、所定の停止表示期間（例えば500m秒間）その表示を維持するためにRAM 308に設けた特図2停止時間管理用タイマの記憶領域に停止期間を示す情報を設定する。この設定により、確定表示された特図2が所定期間停止表示され、特図2変動遊技の結果が遊技者に報知される。また、RAM 308に設けられた電サポ回数記憶部に値がセットされている場合には、その値が1以上であれば、その時短回数から1を減算し、減算結果が1から0となった場合は、

特図確率変動中でなければ、時短フラグをオフする。さらに、大当り遊技中や小当り遊技中にも、時短フラグをオフする。すなわち、主制御部300のCPU304は、大当り遊技状態中および小当り遊技状態中（第二の制御状態中）である場合に、非電サポ状態（第一の進入率制御状態）に移行させる。

#### 【0693】

また、後述するコマンド設定送信処理（ステップS3231）で一般コマンド回転停止設定送信処理を実行させることを示す所定の送信情報を上述の送信情報記憶領域に追加記憶するとともに、変動表示を停止する図柄が特図2であることを示す特図2識別情報を、後述するコマンドデータに含める情報としてRAM308に追加記憶してから処理を終了する。

10

#### 【0694】

また、特図2変動遊技の結果が大当りであれば、大当りフラグがオンされる。この大当りフラグがオンの場合には、所定の停止表示期間が終了したタイミング（特図2停止時間管理用タイマの値が1から0になったタイミング）における特図2状態更新処理では、RAM308の設定領域に特図2作動中を設定するとともに、所定の入賞演出期間（例えば3秒間）すなわち装飾図柄表示装置208による大当りを開始することを遊技者に報知する画像を表示している期間待機するためにRAM308に設けた特図2待機時間管理用タイマの記憶領域に入賞演出期間を示す情報を設定する。また、コマンド設定送信処理（ステップS3231）で一般コマンド入賞演出設定送信処理を実行させるために上述の送信情報記憶領域に5Hを送信情報（コマンド種別）として追加記憶する。

20

#### 【0695】

また、所定の入賞演出期間が終了したタイミング（特図2待機時間管理用タイマの値が1から0になったタイミング）で開始する特図2状態更新処理では、所定の開放期間（例えば29秒間、または可変入賞口234に所定球数（例えば10球）の遊技球の入賞を検出するまで）可変入賞口234の扉部材2341の開閉駆動用のソレノイド（332）に、扉部材2341を開放状態に保持する信号を出力するとともに、RAM308に設けた扉開放時間管理用タイマの記憶領域に開放期間を示す情報を設定する。また、コマンド設定送信処理（ステップS3231）で一般コマンド大入賞口開放設定送信処理を実行させるために上述の送信情報記憶領域に7Hを送信情報（コマンド種別）として追加記憶する。

30

#### 【0696】

また、所定の開放期間が終了したタイミング（扉開放時間管理用タイマの値が1から0になったタイミング）で開始する特図2状態更新処理では、所定の閉鎖期間（例えば1.5秒間）可変入賞口234の扉部材2341の開閉駆動用のソレノイド（332）に、扉部材2341を閉鎖状態に保持する信号を出力するとともに、RAM308に設けた扉閉鎖時間管理用タイマの記憶領域に閉鎖期間を示す情報を設定する。また、コマンド設定送信処理（ステップS3231）で大入賞口閉鎖設定送信処理を実行させることを示す所定の送信情報を上述の送信情報記憶領域に追加記憶する。

#### 【0697】

また、この扉部材の開放・閉鎖制御を所定回数（本実施例では15ラウンドか2ラウンド）繰り返し、終了したタイミングで開始する特図2状態更新処理では、所定の終了演出期間（例えば3秒間）すなわち装飾図柄表示装置208による大当りを終了することを遊技者に報知する画像を表示している期間待機するように設定するためにRAM308に設けた演出待機時間管理用タイマの記憶領域に演出待機期間を示す情報を設定する。

40

#### 【0698】

以上説明したように、主制御部300のCPU304は、大当り遊技状態中に、可変入賞口234の扉部材2341の開閉状態の変化制御を行う可変入賞制御手段の一例に相当する。なお、主制御部300のROM306には、可変入賞口234の扉部材2341の開閉パターンが記憶されており、主制御部300のCPU304は、そのROM306から、特図変動遊技の当否判定に応じた開閉パターンを取得する。

50

## 【0699】

また、主制御部300のCPU304は、特図決定結果が表す停止図柄態様に基づいて、大当り遊技の終了と同時に、RAM308に設けられた確変フラグや時短フラグをオンに設定する。すなわち、主制御部300のCPU304は、後述する特図抽選処理で特図決定結果が「特図A」や「特図C」である場合には確変フラグと時短フラグの双方をオンに設定する。また、特図決定結果が「特図E」である場合には確変フラグと時短フラグのうち確変フラグのみをオンに設定する。さらに、特図決定結果が「特図B」や「特図D」である場合には確変フラグと時短フラグのうち時短フラグのみをオンに設定するとともにRAM308に設けられた電サポ回数記憶部に電サポ回数100回をセットする。確変フラグがオンに設定されていると、特図高確率状態（確率変動中）であり、大当り遊技終了後に大当りに当選する確率が高くなっている状態（特図高確率状態）である。一方、確変フラグがオンに設定されていない（オフに設定されている）と、特図低確率状態である。したがって、確変フラグの設定状態は、当否判定（特図の抽選）の結果に影響を与える。また、時短フラグがオンに設定されていると電サポ状態であり、電チューが開きやすい（例えば当りやすい）、一回の当りに基づく開放時間が長い、一回の当りに基づく開放回数が多いなど可変始動領域制御が遊技者に有利になるように行われる。反対に、時短フラグがオフに設定されていると非電サポ状態であり、可変始動領域制御が遊技者に不利になるように行われる。したがって、時短フラグの設定状態は、可変始動領域制御にも影響を与える。よって、確変フラグおよび／または時短フラグの設定状態を表す情報は、遊技制御情報の一例に相当し、主制御部300のCPU304は遊技制御情報決定手段の一例に相当する。

10

20

## 【0700】

さらに、コマンド設定送信処理（ステップS3231）で一般コマンド終了演出設定送信処理を実行させるために上述の送信情報記憶領域に6Hを送信情報（コマンド種別）として追加記憶する。

## 【0701】

また、所定の終了演出期間が終了したタイミング（演出待機時間管理用タイマの値が1から0になったタイミング）で開始する特図2状態更新処理では、RAM308の設定領域に特図2非作動中を設定する。さらに、特図2変動遊技の結果がハズレであれば、ハズレフラグがオンされる。このハズレフラグがオンの場合には、上述した所定の停止表示期間が終了したタイミング（特図2停止時間管理用タイマの値が1から0になったタイミング）における特図2状態更新処理でも、RAM308の設定領域に特図2非作動中を設定する。特図2非作動中の場合における特図2状態更新処理では、何もせずに次の処理に移行するようにしている。

30

## 【0702】

特図2状態更新処理が終了すると、特図1状態更新処理を行う。この特図1状態更新処理では、特図1の状態に応じて、上述の特図2状態更新処理で説明した各処理を行う。この特図1状態更新処理で行う各処理は、上述の特図2状態更新処理で説明した内容の「特図2」を「特図1」と読み替えた処理と同一であるため、その説明は省略する。なお、特図2状態更新処理と特図1状態更新処理の順番は逆でもよい。

40

## 【0703】

ステップS3227における特図状態更新処理が終了すると、今度は、特図1および特図2それぞれについての特図関連抽選処理を行う。この特図関連抽選処理を実行する主制御部300のCPU304が当否判定手段の一例に相当する。主制御部300は、最初に特図2についての処理（特図2関連抽選処理）を行い、その後、特図1についての処理（特図1関連抽選処理）を行う。このように、主制御部300が特図2関連抽選処理を特図1関連抽選処理よりも先に行うことで、同じタイミングで、第1特図始動口230に遊技球が進入したことに基づいて始動情報を取得し、かつ第2特図始動口232に遊技球が進入したことに基づいて始動情報を取得した場合や、特図2変動遊技の開始条件と、特図1変動遊技の開始条件が同時に成立した場合や、特図2変動遊技の開始条件と特図1変動遊

50

技の開始条件の両方が成立している場合でも、特図2変動遊技が先に変動中となるため、特図1変動遊技は変動を開始しない。すなわち、本実施形態のパチンコ機100は、特図2優先変動を行うものであり、第2特図始動口232への入賞に基づく抽選（特図2の当否判定）を、第1特図始動口230への入賞に基づく抽選（特図1の当否判定）よりも優先して行う。言い換えれば、本実施形態のパチンコ機100では、第1の特別始動領域に遊技球が進入した場合に第1の乱数記憶領域に乱数を最大保留数まで格納し、第2の特別始動領域に遊技球が入賞した場合に第2の乱数記憶領域に乱数を最大保留数まで格納する入賞記憶部と、前記第1の乱数記憶領域および前記第2の乱数記憶領域の両方に乱数が記憶されている場合に、前記第1の乱数記憶領域に前記乱数が記憶された時期および前記第2の乱数記憶領域に前記乱数が記憶された時期とは無関係に該第2の乱数記憶領域に記憶されている乱数に基づいて当否判定を行うとともに、前記第1の乱数記憶領域に乱数が記憶されておらず、かつ前記第2の乱数記憶領域に乱数が記憶されている場合には、該第2の乱数記憶領域に記憶されている乱数に基づいて当否判定を行い、前記第2の乱数記憶領域に乱数が記憶されておらず、かつ前記第1の乱数記憶領域に乱数が記憶されている場合には、該第1の乱数記憶領域に記憶されている乱数に基づいて当否判定を行う当否判定手段を備えている。また、第1特図表示装置212あるいは第2特図表示装置214による特図変動遊技の大当たり判定の結果の報知は、主制御部300で行われ、第2特図始動口232への入賞に基づく当否判定の結果報知が、第1特図始動口230への入賞に基づく当否判定の結果報知よりも優先して行われ、当否判定が行われていない始動情報として、特図1の始動情報と特図2の始動情報のうちの特図1の始動情報のみが残っている状態で、特図2の始動情報が新たに記憶された場合には、新たに記憶された特図2の始動情報に基づく当否判定の結果の報知が、既に記憶されていた特図1の始動情報に基づく当否判定の結果の報知よりも先に行われる。また、始動情報を取得する始動情報取得手段は、第1の始動情報記憶手段および第2の始動情報記憶手段のうちの両方に始動情報が記憶されている場合には、該第2の始動情報記憶手段から始動情報を取得し、該第1の始動情報記憶手段および該第2の始動情報記憶手段のうちの一方に始動情報が記憶されている場合には、始動情報が記憶されている始動情報記憶手段から始動情報を取得するものである。なお、特図2状態更新処理に続いて先に特図2関連抽選処理を行い、それから、特図1状態更新処理を行い、その後、特図1関連抽選処理を行うようにしてもよい。

#### 【0704】

図93は、特図関連抽選処理の流れを示すフローチャートである。この図93に示す特図関連抽選処理では、特図1と特図2を区別しないで示しているが、先に特図2についてステップS3229a～ステップS3229fまでの処理を行い、その後、特図1についてステップS3229a～ステップS3229fまでの処理を行う。ここでは、特図2と特図1を区別しないで説明する。

#### 【0705】

図93に示す特図関連抽選処理では、まず、特図保留情報があるか否かを判定する（ステップS3229a）。ここに特図保留情報とは、特図の保留数を指す。すなわち、ここでは、保留している特図変動遊技の数が1以上であるか否かを判定する。なお、特図の保留数をデータとして持っていないなくとも、例えば、保留に対応した乱数（特図当選乱数）を特図保留情報として認識するようにしても良い。特図保留情報がなければ、この特図関連抽選処理は終了になり、特図保留情報があればステップS3229bに進む。ステップS3229bでは、特図表示装置（212，214）が特図変動表示中であるか、または停止表示中であるか否かを判定し、いずれかの表示中である場合には、この特図関連抽選処理は終了になり、いずれの表示中でもない場合には、ステップS3229cに進む。ステップS3229cでは、特図作動中であるか否かを判定し、特図作動中であれば、この特図関連抽選処理は終了になり、特図非作動中であれば、ステップS3229dに進む。

#### 【0706】

ステップS3229dでは特図抽選処理を行う。ここではまず、RAM308に設けられた特図の保留記憶部から、最も過去に格納した始動情報である1セット分の乱数（特図

当選乱数、大当り用特図乱数、小当り用特図乱数、ハズレ用特図乱数、および特図変動時間決定用乱数)を取り出し、その保留記憶部にまだ格納されている始動情報(乱数のセット)を、今記憶されている領域から隣の領域に移し替える。すなわち、最も過去に格納した始動情報を特図の保留記憶部から取り出し、さらに特図の保留記憶部に始動情報が格納されていれば、N番目に古い始動情報を特図2の保留記憶部におけるN-1番目に古い始動情報として設定したことになる。また、RAM308に記憶している保留数を1減算する。RAM308の特図の保留記憶部から1セット分の乱数(特図当選乱数、大当り用特図乱数、小当り用特図乱数、ハズレ用特図乱数、および特図変動時間決定用乱数)を取り出す処理を行う主制御部300のCPU304が、始動情報取得手段の一例に相当する。

【0707】

図94(a)は特図抽選テーブルを示す図である。このテーブルは、主制御部300のROM306に記憶されている。

【0708】

主制御部300のCPU304は、RAM308の保留記憶部から始動情報を取り出すと、確変フラグを参照し、確変フラグがオンであれば確率変動中(特図高確率状態)であるため、取得した特図当選乱数から確率変動中の特図大当り当選データを引き、キャリアが発生した場合(特図当選乱数よりも特図当選データの値が大きい場合)は特図大当りに当選とし、キャリアが発生しなかった場合は特図大当りには不当選になる。すなわち、特図大当り当選乱数範囲は0~124になる。確率変動中では、特図当選乱数の元になる乱数は、図77に示す乱数生成回路318のチャンネルCH2から出力される。このチャンネルCH2には、図90に示す乱数生成回路初期設定処理における乱数生成範囲の設定(ステップS3053a)において0~4999の乱数生成範囲が設定されており、確率変動中の特図当選乱数の取り得る範囲は0~4999になる。したがって、確率変動中における特図大当りの当選確率は1/40になる。

【0709】

一方、確変フラグがオフであれば非確率変動中(特図低確率状態)であるため、取得した特図当選乱数から、確率変動中のデータと同じ非確率変動中の特図大当り当選データを引き、キャリアが発生した場合(特図当選乱数よりも特図当選データの値が大きい場合)は特図大当りに当選とし、キャリアが発生しなかった場合は特図大当りには不当選になる。すなわち、非確率変動中でも特図大当り当選乱数範囲は0~124になる。非確率変動中では、特図当選乱数の元になる乱数は、図77に示す乱数生成回路318のチャンネルCH1から出力される。このチャンネルCH1には、乱数生成範囲の設定(ステップS3053a)において0~49999の乱数生成範囲が設定されており、非確率変動中の特図当選乱数の取り得る範囲は0~49999になる。したがって、非確率変動中における特図大当りの当選確率は1/400になる。

【0710】

本実施形態では、特図当選乱数の取り得る範囲(乱数生成範囲)を、デフォルトの0~65535から0~49999に制限することで大当り確率を、きりのよい1/400にすることができる。また、特図大当り当選データを、確率変動中と非確率変動中で同じ値にすることができる。乱数生成範囲を制限せずに確率を決定することは開発工数の増大を招くとともに誤った確率設計の原因となってしまう場合がある。特に、一の契機に基づいて複数回の抽選を行う場合に煩雑さはより顕著になってしまう。よって、乱数生成範囲を制限することは開発工数を削減するとともに、抽選処理の安定化を図ることができるといえる。また、特図大当り当選データを、確率変動中と非確率変動中で共通化することも、開発工数を削減するとともに、抽選処理の安定化を図ることができるといえる。

【0711】

図94(a)では、特図大当りの抽選テーブルしか示していないが、特図小当りの当選データとして所定の値が用意されている。主制御部300のCPU304は、取得した特図当選乱数から特図大当り当選データを引き、キャリアが発生しなかった場合(特図当選乱数よりも特図当選データの値が小さい場合)には、今度は、特図大当り当選データを引

10

20

30

40

50



いた値から特図小当り当選データを引き、ここでキャリアが発生した場合（特図当選乱数よりも特図小当り当選データの値が大きい場合）は特図小当りに当選とし、ここでもキャリアが発生しなかった場合はハズレになる。なお、ハズレの当選データを別途設けておいてもよい。こうして、特図の「大当り」、「小当り」、「ハズレ」が判定され、判定結果を特図当否判定結果として得る。なお、保留している特図変動遊技の数は、RAM 308に設けた特図保留数記憶領域に記憶するようにしており、ステップS 3 2 2 9 dを実行するたびに、保留している特図変動遊技の数から1を減算した値を、この特図保留数記憶領域に記憶し直すようにしている。

#### 【0712】

次いで、図93に示す特図関連抽選処理では、特図当否判定結果に基づいて特図の停止図柄をソフトウェア乱数を用いて抽選する（ステップS 3 2 2 9 e）。図94（b）は、停止図柄抽選テーブルを示す図である。このテーブルも、主制御部300のROM 306に記憶されている。この停止図柄抽選テーブルは、当否判定結果ごとに特図の停止図柄態様（図5（a）参照）に対応した図柄当選データが規定されている。また、同図（b）には、当選確率も示されている。

#### 【0713】

主制御部300のCPU 304は、当否判定結果が大当りの場合であれば、RAM 308の保留記憶部から先に取得した1セット分の乱数のうちの大当り用特図乱数（取り得る数値範囲は0～99）から特図A、特図B、・・・特図Fの順序で各停止図柄に対応した図柄当選データを徐々に減算し、キャリアが発生した場合の停止図柄を特図決定結果とする。また、当否判定結果が小当りの場合であれば、RAM 308の保留記憶部から先に取得した1セット分の乱数のうちの小当り用特図乱数（取り得る数値範囲は0～99）から特図Gの図柄当選データを引き、キャリアが発生した場合は「特図G」に当選とし、キャリアが発生しなかった場合には、特図Gの図柄当選データを引いた値からさらに特図Hの図柄当選データを引き、ここではキャリアが発生するため「特図H」に当選とする。なお、特図Hの図柄当選データを引く前に、「特図H」に当選したと判定してもよい。このようにして、当否判定結果が小当りの場合には、特図決定結果として「特図G」または「特図H」を決定する。さらに、当否判定結果がハズレの場合であれば、RAM 308の保留記憶部から先に取得した1セット分の乱数のうちのハズレ用特図乱数（取り得る数値範囲は0～99）から特図Iの図柄当選データを引き、キャリアが発生した場合は「特図I」に当選とし、キャリアが発生しなかった場合には、特図Iの図柄当選データを引いた値からさらに特図Jの図柄当選データを引き、ここではキャリアが発生するため「特図J」に当選とする。なおここでも、特図Jの図柄当選データを引く前に、「特図J」に当選したと判定してもよい。このようにして、当否判定結果がハズレの場合には、特図決定結果として「特図I」または「特図J」を決定する。なお、第1副制御部400では、ここで決定された特図決定結果に応じた装飾図柄の組合せである停止図柄態様を決定する。

#### 【0714】

また、RAM 308の保留記憶部から先に取得した1セット分の乱数のうちの特図変動時間決定用乱数（取り得る数値範囲は0～255）を取得し、取得した特図変動時間決定用乱数を用いた抽選で、特図決定結果に基づいて、複数の変動時間のうちから特図表示装置（212，214）に特図を変動表示する時間（特図変動時間）を1つ選択する。なお、第1副制御部400では、ここで決定された特図変動時間に応じた装飾図柄表示装置208の演出態様を決定する。特図変動時間決定用乱数値を用いた抽選で複数の変動時間のうちから特図変動時間を1つ選択する主制御部300のCPU 304も、抽選手段の一例に相当する。ここで、特図変動時間の抽選処理は、特図の大当りか否かを抽選する当否判定とは異なり、特図の図柄を決定する抽選とも異なる、演出に関する抽選処理に相当する。すなわち、制御状態の変更とは無関係であって、賞球の払出にも無関係な抽選処理である。

#### 【0715】

ステップS 3 2 2 9 fでは、RAM 308の保留記憶部から先に取得した1セット分の

10

20

30

40

50

乱数を消去し、この特図関連抽選処理は終了になる。ステップS3229の特図関連抽選処理に続いて行われるステップS3231では、コマンド設定送信処理を行い、各種のコマンドが第1副制御部400に送信される。なお、第1副制御部400に送信する出力予定情報は本実施形態では16ビットで構成しており、ビット15はストロブ情報（オンの場合、データをセットしていることを示す）、ビット11～14はコマンド種別（本実施形態では、基本コマンド、図柄変動開始コマンド、図柄変動停止コマンド、入賞演出開始コマンド、終了演出開始コマンド、当りラウンド数指定コマンド、復電コマンド、RAMクリアコマンド、特図保留増加コマンド、普図保留増加コマンドなどコマンドの種類を特定可能な情報）、ビット0～10はコマンドデータ（コマンド種別に対応する所定の情報）で構成している。

10

#### 【0716】

具体的には、ストロブ情報は上述のコマンド送信処理でオン、オフするようにしている。また、コマンド種別が図柄変動開始コマンドの場合であればコマンドデータに、特図停止図柄を表す情報、制御状態を表す情報（時短フラグおよび確変フラグの設定状態を表す情報）、特図変動時間を表す情報などを示す情報を含み、図柄変動停止コマンドの場合であれば、特図停止図柄を表す情報（特図決定結果）、制御状態を表す情報などを含み、入賞演出開始コマンドおよび終了演出開始コマンドの場合であれば、制御状態を表す情報などを含み、当りラウンド数指定コマンドの場合であれば制御状態を表す情報、当りラウンド数などを含むようにしている。コマンド種別が基本コマンドを示す場合は、コマンドデータにデバイス情報、第1特図始動口230への入賞の有無、第2特図始動口232への入賞の有無、可変入賞口234への入賞の有無などを含む。

20

#### 【0717】

また、上述の一般コマンド回転開始設定送信処理では、コマンドデータにRAM308に記憶している、特図停止図柄を表す情報（特図決定結果）、制御状態を表す情報、特図変動時間を表す情報、保留している第1特図変動遊技または第2特図変動遊技の数などを示す情報を設定する。

#### 【0718】

上述の一般コマンド回転停止設定送信処理では、コマンドデータにRAM308に記憶している、特図停止図柄を表す情報（特図決定結果）、制御状態を表す情報などを示す情報を設定する。上述の一般コマンド入賞演出開始設定送信処理では、コマンドデータに、RAM308に記憶している、入賞演出期間中に装飾図柄表示装置208・各種ランプ418・スピーカ120に出力する演出制御情報、制御状態を表す情報、保留している第1特図変動遊技または第2特図変動遊技の数などを示す情報を設定する。入賞演出開始コマンドを受信した第1副制御部400は、その入賞演出開始コマンドに基づいて第2副制御部500に入賞演出制御コマンドを送信する。入賞演出制御コマンドを受信した第2副制御部500は、装飾図柄表示装置208に、大当り遊技が開始されることを遊技者に報知する画像を所定のオープニング演出期間（例えば3秒間）表示させ、大当り遊技が開始する。

30

#### 【0719】

上述の一般コマンド終了演出開始設定送信処理では、コマンドデータに、RAM308に記憶している、演出待機期間中に装飾図柄表示装置208・各種ランプ418・スピーカ120に出力する演出制御情報、制御状態を表す情報、保留している第1特図変動遊技または第2特図変動遊技の数などを示す情報を設定する。終了演出開始コマンドを受信した第1副制御部400は、その終了演出開始コマンドに基づいて第2副制御部500に終了演出制御コマンドを送信する。終了演出制御コマンドを受信した第2副制御部500は、装飾図柄表示装置208に大当りを終了することを遊技者に報知する画像を所定の終了演出期間（例えば3秒間）表示させ、大当り遊技が終了する。

40

#### 【0720】

上述の一般コマンド大入賞口開放設定送信処理では、コマンドデータに、RAM308に記憶している当りラウンド数、現在のラウンド数、制御状態を表す情報などを示す情報を設定する。上述の一般コマンド大入賞口閉鎖設定送信処理では、コマンドデータに、R

50

AM308に記憶している現在のラウンド数、制御状態を表す情報、保留している第1特図変動遊技または第2特図変動遊技の数などを示す情報を設定する。

#### 【0721】

また、このステップS3231では一般コマンド特図保留増加処理も行われる。この一般コマンド特図保留増加処理では、特図保留増加コマンドのコマンドデータにRAM308の送信用情報記憶領域に記憶している特図識別情報（特図1または特図2を示す情報）、制御状態を表す情報、事前判定した特図1あるいは特図2の情報を設定する。

#### 【0722】

さらに、このステップS3231では一般コマンド普図保留増加処理も行われる。この一般コマンド普図保留増加処理では、普図保留増加コマンドのコマンドデータに、制御状態を表す情報等を設定する。また、主制御部300から第1副制御部400には、普図絡みのコマンドとして、普図の変動表示が開始した（する）ことを表す普図変動開始コマンドも送信される。なお、主制御部300から第1副制御部400には、普図の変動表示が停止した（する）ことを表す普図変動停止コマンドや、一对の羽根部材2321が開放を開始した（する）ことを表す電チュー開放開始コマンドや、一对の羽根部材2321が閉鎖した（する）ことを表す電チュー閉鎖コマンドを送信するようにしてもよい。

#### 【0723】

第1副制御部400では、受信した出力予定情報に含まれるコマンド種別により、主制御部300における遊技制御の変化に応じた演出制御の決定が可能になるとともに、出力予定情報に含まれているコマンドデータの情報に基づいて、演出制御内容を決定することができるようになる。また、第1副制御部400では、コマンドに含まれている当りラウンド数と現在のラウンド数に基づき、当り全ラウンドが終了するまでの残りラウンド数を取得する。

#### 【0724】

また、このコマンド設定送信処理では、図4に示す払出制御部600にもコマンドを送信する。払出制御部600に出力する出力予定情報および払出要求情報は1バイトで構成しており、ビット7にストロープ情報（オンの場合、データをセットしていることを示す）、ビット6に電源投入情報（オンの場合、電源投入後一回目のコマンド送信であることを示す）、ビット4～5に暗号化のための今回加工種別（0～3）、およびビット0～3に暗号化加工後の払出要求数を示すようにしている。

#### 【0725】

次に、図91に示す主制御部タイマ割込処理では、外部出力信号設定処理（ステップS3233）を行う。この外部出力信号設定処理では、RAM308に記憶している遊技情報を、情報出力回路336を介してパチンコ機100とは別体の情報入力回路350に出力する。ステップS3235では、デバイス監視処理を行う。

#### 【0726】

図95は、デバイス監視処理の流れを示すフローチャートである。まず、ステップS3235aでは、内部情報レジスタ3101の情報を取得する。なお、このステップS3235aはタイマ割込み毎に実行されるため、内部情報レジスタ3101の情報はタイマ割込み毎に取得されることになる。

#### 【0727】

ステップS3235bでは、ステップS3235aで取得した情報に基づいて、乱数生成回路318に異常があるか否かを判定する。なお、ここでの乱数生成回路の異常とは、周波数監視回路3182の異常および乱数監視回路3184の異常の双方を指す。ここで、異常があると判定された場合にはステップS3235cに進み、異常があると判定されなかった場合にはステップS3235eに進む。

#### 【0728】

ステップS3235cでは、乱数生成回路異常フラグをオンに設定する。なお、このフラグはRAM308内に記憶されている。より具体的には、周波数監視回路3182の異常を示すフラグおよび乱数監視回路3184の異常を示すフラグがある。以下では、これ

10

20

30

40

50

らのフラグを総称して乱数生成回路異常フラグと称する場合がある。

#### 【0729】

続いて実行されるステップS3235dでは、乱数生成回路異常フラグ設定コマンドの送信準備が実行され、ステップS3235eに進む。なお、この乱数生成回路異常フラグ設定コマンドは、周波数監視回路3182の異常か、乱数監視回路3184の異常か識別可能な情報を含むコマンドである。

#### 【0730】

ステップS3235eでは、その他デバイス監視処理が実行される。例えば、上述のステップS3205において信号状態記憶領域に記憶した各種センサの信号状態を読み出して、ガラス枠開放エラーの有無または下皿満タンエラーの有無などを監視し、ガラス枠開放エラーまたは下皿満タンエラーを検出した場合に、第1副制御部400に送信すべき送信情報に、ガラス枠開放エラーの有無または下皿満タンエラーの有無を示すデバイス情報を設定する。また、図4に示す各種ソレノイド332を駆動して第2特図始動口232や、可変入賞口234の開閉を制御したり、駆動回路324、326、330を介して普通図柄表示装置210、第1特図表示装置212、第2特図表示装置214、各種状態表示部328などに出力する表示データを、I/O310の出力ポートに設定する。また、払出要求数送信処理（ステップS3219）で設定した出力予定情報をI/O310の出力ポートを介して第1副制御部400に出力する。これらの処理が終了すると、このステップS3235のデバイス監視処理は終了になる。

#### 【0731】

続いて実行されるステップS3237では、電源の遮断（電断）を検出したか否かを判定するために、低電圧信号がオンであるか否かを判定する。そして、低電圧信号がオフの場合（電源の遮断を検知していない場合）にはステップS3239に進み、低電圧信号がオンの場合（電源の遮断を検知した場合）にはステップS3241に進む。

#### 【0732】

ステップS3239では、タイマ割込終了処理を行う。このタイマ割込終了処理では、ステップS3201で一時的に退避した各レジスタの値を元の各レジスタに設定したり、割り込み許可の設定などを行い、その後、図88に示す主制御部メイン処理に復帰する。一方、ステップS3241では、復電時に電断時の状態に復帰するための電断時処理を行う。なお、本実施形態では、電源の遮断を判定する処理（ステップS3237）および電断処理（ステップS3241）を主制御部の割込み制御において実行しているが、必ずしも主制御部の割込み制御で行う必要はなく、低電圧信号の入力に基づいて強制割込みを発生させ、この強制割込みによって電断処理が実行されるように構成してもよい。ただし、このように強制割込みによって実行される電断処理では、本実施形態における電断処理（ステップS3241）の処理に加え、上述したタイマ割込み開始処理（S3201）において実行されるCPU304の各レジスタの値をスタック領域に一時的に退避する処理も同時に実行する必要がある。

#### 【0733】

図130は、主制御部における電断時処理の流れを示すフローチャートである。

この電断時処理では、まず、スタックポインタを復帰データとしてRAM308の所定の領域に退避し（ステップS3431）、次いで、RAM308に設けた電源ステータス記憶領域に記憶した電源ステータスをサスペンドを示す情報に更新する（ステップS3432）。続いて、RAM308の所定の領域（例えば全ての領域）に記憶している1バイトデータを初期値が0である1バイト構成のレジスタに全て加算することによりチェックサムを算出し、算出したチェックサムの結果をRAM308に設定する（ステップS3433）。最後に、RAM308へのアクセスを禁止に設定し（ステップS3434）、この電断時処理は終了になり、やがてパチンコ機100は電断する。なお、電圧が完全に低下する前に復帰した場合には、図75に示すリセット制御回路314内に設けられたWDT3141がタイムアウトすることによってリセット制御回路314によりリセットがかかり設定されたりリセット動作（ここではシステムリセット）が実行される。

以上説明した主制御部タイマ割込み処理も、主処理の一例に相当する。

#### 【0734】

続いて、図91に示す入賞受付処理（ステップS3217）のうち、特図当選乱数および普図当選乱数を取得する処理について説明する。図96は、ステップS3217における入賞受付処理のうち、特図当選乱数および普図当選乱数を取得する処理の流れを示した図である。

#### 【0735】

ステップS3217aでは、乱数生成回路異常フラグがオンに設定されているか否かを判定する。このフラグは、図95のステップS3235cで設定されるフラグであり、乱数更新用のクロックの周波数に異常があった場合や、図77に示す乱数更新回路3183で更新される乱数が正常に更新されなかった場合等に、対応するフラグがオンに設定される。このフラグがオンに設定されている場合には、この入賞受付処理は終了になり、異常がなかった場合にはステップS3217bに進む。

#### 【0736】

ステップS3217bでは、第1特図始動口230に入賞したことを表す特図1の入球信号があり、且つ、保留している特図変動遊技の数が所定数（本実施形態では4）未満であるか否かを判定し、否定的な場合にはステップS3217fへ進み、肯定的な場合にはステップS3217cに進む。ステップS3217cでは、RAM308に用意された確変フラグを参照し、確率変動中（特図高確率状態）か否かを判定する。確変フラグがオフに設定されていれば、非確率変動中（特図低確率状態）であり、ステップS3217dが実行される。このステップS3217dでは、乱数生成回路318のチャンネルCH1の乱数レジスタから乱数を取得する。本実施形態では乱数をラッチする契機となる入球信号とは別に、CPUに同様の信号が送られ、CPUはその信号を受信したに基づいて所定のチャンネル（ここではCH1）における信号に対応する乱数レジスタからラッチされている乱数を取得する。CPU304は、取得した乱数にRレジスタの値を加算し、さらに1を加えた値を特図1当選乱数として得、ステップS3217fに進む。なお、CPU304は、図90に示すステップS3053aにおいて設定された乱数生成範囲を認識しており、その乱数生成範囲に基づいて加算処理を行う。すなわち、加算した値が乱数生成範囲の最大値を超える場合には当該乱数生成範囲の最小値に残りの値を加算する。

#### 【0737】

一方、確変フラグがオンに設定されていれば、確率変動中であり、ステップS3217eが実行される。このステップS3217eでは、乱数生成回路318のチャンネルCH2の乱数レジスタから乱数を取得し、ステップS3217dと同様に、CPU304は、取得した乱数にRレジスタの値を加算し、さらに1を加えた値を特図1当選乱数として得、ステップS3217fに進む。

#### 【0738】

ステップS3217fでは、第2特図始動口232に入賞したことを表す特図2の入球信号があり、且つ、保留している特図変動遊技の数が所定数（本実施形態では4）未満であるか否かを判定し、否定的な場合にはステップS3217jへ進み、肯定的な場合にはステップS3217gに進む。ステップS3217gでも、ステップS3217cと同様に、RAM308に用意された確変フラグを参照し、確変フラグがオフに設定されていれば（非確率変動中であれば）、乱数生成回路318のチャンネルCH1の乱数レジスタから乱数を取得し、ステップS3217dと同様に、CPU304は、取得した乱数にRレジスタの値を加算し、さらに1を加えた値を、ここでは特図2当選乱数として得（ステップS3217H）、ステップS3217jに進む。一方、確変フラグがオンに設定されていれば（確率変動中であれば）、乱数生成回路318のチャンネルCH2の乱数レジスタから乱数を取得し、同じくステップS3217dと同様に、CPU304は、取得した乱数にRレジスタの値を加算し、さらに1を加えた値を特図2当選乱数として得（ステップS3217i）、ステップS3217jに進む。

#### 【0739】

ステップ S 3 2 1 7 j では、普図始動口 2 2 8 に入賞したことを表す普図の入球信号があり、且つ、保留している普図変動遊技の数が所定数（本実施形態では 2）未満であるかを判定し、否定的な場合には入賞受付処理は終了になり、肯定的な場合にはステップ S 3 2 1 7 k に進む。ステップ S 3 2 1 7 k では、R A M 3 0 8 に用意された時短フラグを参照し、時短フラグがオフに設定されていれば、普図低確率状態（非電サボ中）であり、ステップ S 3 2 1 7 l が実行される。ステップ S 3 2 1 7 l では、乱数生成回路 3 1 8 のチャンネル C H 3 の乱数レジスタから普図当選乱数として乱数を取得し、この入賞受付処理は終了になる。一方、時短フラグがオンに設定されていれば、普図高確率状態（電サボ中）であり、ステップ S 3 2 1 7 m が実行される。ステップ S 3 2 1 7 m では、乱数生成回路 3 1 8 のチャンネル C H 4 の乱数レジスタから普図当選乱数として乱数を取得し、入賞受付処理は終了になる。

10

#### 【0740】

ここで、上記説明した内容を踏まえて、遊技台への電圧供給が低下した場合の動作について説明する。例えば大当り中の状態で停電などによって電源が切られた場合、再度電源投入によって遊技台が初期状態になると、遊技者にとって著しく不利益な状態になる。このような事態にならないよう、コンデンサに蓄えられた電力により遊技台の状態を保持する電断時処理（図 9 1 のステップ S 3 2 4 1）が実行される。この電断時処理は、電源が切られた場合の他、静電気などの要因により供給電圧が一時的に低下した場合にも実行される。以下、電源が切られた場合（以下、電源オフと称する）の動作と、供給電圧が一時的に低下した場合（以下、瞬断と称する）の動作について、図 9 7 を用いて説明する。同図（a）は、電源オフの場合の動作を示す図であり、（b）は、瞬断の場合の動作を示す図である。

20

#### 【0741】

電源が切られた場合、電圧が供給されなくなる（電断発生）が、すぐに電圧は 0 にはならず、コンデンサに蓄えられた電力により供給電圧が徐々に降下する。供給電圧が所定の電圧（本実施形態では 9 V）まで効果すると電圧監視回路 3 3 8 から低電圧信号が C P U 3 0 4 に送信される。この信号を受信したことにより、主制御部タイマ割込み処理のステップ S 3 2 3 7 により電断が発生したと判定され、ステップ S 3 2 4 1 の電断時処理が実行される。図 9 7（a）には、電断が発生してから供給電圧が 9 V に降下するまで通常処理が実行され、供給電圧が 9 V になった時点で電断時処理が開始されていることが示されている。この電断時処理により、R A M 3 0 8 の内部に遊技台の状態を示すデータが記憶される。なお、本実施形態では、この電断時処理の実行に必要な時間（主制御部電断時処理猶予時間）を十分確保することができるコンデンサが備えられている。その後、電断時処理が終了すると、そのまま待機状態になる。その後電源の供給が再開されない場合、電圧が 0 に落ちて遊技台の動作が停止する。

30

#### 【0742】

一方、瞬断の場合にも、電圧監視回路 3 3 8 から低電圧信号が C P U 3 0 4 に送信されることにより、上記電断時処理が実行された後、そのまま待機状態になる。図 9 7（b）には、一時的な電圧降下により、電断時処理が開始されていることが示されている。ここで、電圧が動作電圧まで回復しても、待機状態が継続する。この待機状態が継続していると、主制御部 3 0 0 のリセット制御回路 3 1 4 の W D T 3 1 4 1 がリスタートされず（図 9 1 のステップ S 3 2 0 3 が実行されない）、W D T 3 1 4 1 がタイムアウトする。このことによりリセット制御回路 3 1 4 はシステムリセット信号を出力する。その後、図 8 8 に示す主制御部メイン処理が再開すると、電断時処理で記憶されていたデータに基づいてステップ S 3 0 0 9 のデータ書き戻し処理を含む復帰処理（ステップ S 3 0 0 9 ～ステップ S 3 0 1 1）が実行される。図 9 7（b）には、W D T 3 1 4 1 のタイムアウトによるリセット信号が送信されたことにより、待機状態から復帰処理（リセット処理）に移っていることが示されている。この復帰処理により遊技台の状態が電断時処理前の状態に復帰する。なお、ここで、電断時処理で記憶されていたデータに不具合があった場合は、電断時処理前の状態に復帰できないと判定され、R A M 3 0 8 の内容がクリアされる（図 8 8

40

50

のステップ S 3 0 1 3)。

【0743】

次に、本実施形態の乱数生成回路 3 1 8 により、乱数の生成範囲を設定することの効果について説明し、その後問題点について説明する。上述したように本実施形態の遊技台が有する乱数生成回路 3 1 8 では、抽選結果の基になる乱数の生成範囲を設定することができる。この機能を利用して、遊技状態の設定が容易になる。

【0744】

まず、乱数生成範囲が 0 ～ 6 5 5 3 5 の場合について説明する。例えば、これらの乱数が当りかハズレのいずれかに対応している場合に、当選確率が  $1 / 4 0 0$  (0. 2 5 %) になるように設計しようとしたとする。生成される乱数は 6 5 5 3 6 通りあるが、6 5 5 3 6 の  $1 / 4 0 0$  を計算すると 1 6 3. 8 4 となり、整数にならない。仮にこの数字に近い 1 6 4 通りの乱数を当りに対応させた場合、厳密には  $1 / 4 0 0$  の当選確率にならない。

10

【0745】

ここで、上記の乱数生成範囲が 0 ～ 4 9 9 9 9 に設定されたとする。上記の場合と同様に当選確率が  $1 / 4 0 0$  (0. 2 5 %) になるようにするには、生成される乱数 5 0 0 0 0 通りの  $1 / 4 0 0$ 、すなわち 1 2 5 通りの値を当りに対応させればよい。例えば、5 0 0 0 0 通り乱数のうち 0 ～ 1 2 4 までの値を当りに対応させ、それ以外の 1 2 5 ～ 4 9 9 9 9 の値をハズレに対応付けることで、当選確率を  $1 / 4 0 0$  にすることができる。すなわち、当選確率が設計値通り (例えば  $1 / 4 0 0$ ) になるように、乱数生成範囲 (例えば 0 ～ 4 9 9 9 9) および当りに対応する乱数の範囲 (例えば 0 ～ 1 2 4) をそれぞれ設定することができる。

20

【0746】

以上説明したように、乱数生成範囲を適切に設定することによって、所望の当選確率を設定することができる。これによって、例えば小当りに対応する乱数生成範囲を除外して大当りに対応する乱数生成範囲を残したり、あるいは大当りに対応する乱数生成範囲の半分に該当する乱数が生成されないようにすることも可能である。特に図 8 2 を用いて説明したような乱数生成範囲が設定できれば、最大値だけを設定する場合と比較して遊技台の設計が容易になる。このため、乱数生成範囲を設定することは、遊技台の設計において有用であると言える。

30

【0747】

次に、乱数生成範囲を異ならせることにより、当選確率を変更する方法について説明する。例えば、0 ～ 4 9 9 9 9 の乱数生成範囲のうち、0 ～ 1 2 4 までの値を当りに対応させた状態から、乱数生成範囲の最大値を 4 9 9 9 9 から 2 4 9 に変更したとする。この場合、0 ～ 1 2 4 までの値が当りに対応し、1 2 5 ～ 2 4 9 までの値がハズレに対応し、当選確率は  $1 / 2$  になる。すなわち、当り又はハズレに対応する値を固定したまま乱数生成範囲だけを異ならせることにより、当選確率を容易に変更することができる。なお、上記説明では、よりわかり易くするために、当りの範囲を偏らせた例について説明したが、当りの範囲が偏っていてもよい。

【0748】

生成された乱数に基づいて当りかハズレかを判定する方法として、これらの対応を定めたテーブルを用意しておき、これを使用する方法が考えられる。しかし、この方法で当選確率を変更しようとした場合、当選確率に対応するテーブルをそれぞれ用意しておき、遊技の状態に応じてこれらを使い分ける処理を実行させる必要がある。上記説明した方法では、当り又はハズレに対応する値を定めたテーブルが一つですむため、メモリの容量負担が少なく済む。当然のことながら、テーブルを切り替える処理は不要である。

40

【0749】

なお、本実施形態では、乱数生成範囲の異なる複数の乱数生成チャンネルを用意し、これらを使い分けることにより遊技中の当選確率を変更しているが、一つのチャンネルに対して適宜乱数生成範囲を変更する構成を用いて当選確率を変更させてもよい。また、本実

50

施形態では、乱数を加工した上で抽選に用いる場合があるが、この場合、加工を考慮したうえで対応するテーブルを用意すればよい。

#### 【0750】

ここで、図98を用いて、乱数生成範囲が意図せずに変更されてしまった場合の問題について説明する。同図は、乱数生成範囲の問題点を示す図である。例えば、静電気等の要因によって上述した瞬断が生じ、その後復帰処理（ユーザリセット）が実行されたとする。この場合、乱数生成回路318の内部レジスタは、そのまま復帰処理以前の状態が維持される。ところが、電圧の変化によって内部レジスタの情報が書き換わる可能性がある。これによって乱数生成範囲が意図せずに変更されると、当選確率が変わってしまうことになる。

10

#### 【0751】

例えば、非確率変動状態において、図98（a）の上段に示すように、0～49999の乱数生成範囲が設定されており、さらにこのうち0～124（左下がりのハッチングで示す範囲）が当りに対応する乱数であるとする。この状態で静電気等の要因が生じ、乱数生成範囲が0～250になったとする。この場合、当選確率が1/400から1/2になってしまうため、店側にとっては大きな損害になってしまう。

#### 【0752】

また、確率変動状態において、図98（b）の上段に示すように、0～49999の乱数生成範囲が設定されており、さらにこのうち0～124（左下がりのハッチングで示す範囲）が当りに対応する乱数であるとする。この状態で静電気等の要因が生じ、乱数生成範囲が0～59999になったとする。この場合、当選確率が1/40から1/480になってしまうため、遊技者にとって著しく不利益になる。

20

#### 【0753】

上記の例で説明したように、乱数が生成される数値範囲を任意の数値範囲に制限する手法を用いて抽選処理を行うと、何らかの異常で制限した数値範囲が変化した場合に抽選処理の不安定化を招いてしまうという問題が生じる。

#### 【0754】

このような事態を防止するため、本実施形態の遊技台では、図90のステップS3053aにおいて、乱数生成範囲を復帰処理の要因が発生した直前の状態に設定し直す処理を実行させる。このように再度乱数生成範囲を設定し直すことによって、上記のような問題が生じないようにすることができる。すなわち、乱数が生成される数値範囲を任意の数値範囲に制限する手法を用いて抽選処理を行いつつも、抽選処理の不安定化を招くことを抑止することができる。

30

#### 【0755】

なお、本実施形態では、遊技台の起動時に乱数生成範囲が設定されるため、復帰処理の要因が発生した直前の状態が、初期状態と同じであるが、例えば、遊技中に乱数生成範囲を変更可能な構成の場合には、直前の状態に戻す処理が実行されればよい。

#### 【0756】

また、本実施形態のように、乱数生成範囲が設定されていたチャンネルに対しては乱数生成範囲を設定し直すとともに、乱数生成範囲が設定されていないチャンネルに対しては乱数生成範囲を設定し直さない構成の場合、初期化されるチャンネルと初期化されないチャンネルが存在することになる。この場合、基本回路302の外部からチャンネルが初期化されたか否かがわからないため、不正を防止する効果がある。

40

#### 【0757】

さらに、本実施形態の遊技台では、図90のステップS3053bにおいて、乱数レジスタ3188をラッチすることを許容する許容状態にしている。このようにすることで、乱数レジスタ3188にラッチされている乱数に異常があった場合に、この乱数が使用される頻度を下げることができる。この際乱数ラッチフラグレジスタ3189の内容は、乱数レジスタ3188が乱数をラッチすることが可能な状態であることを示す情報に書き換えられる。なお、本実施形態では、ユーザプログラムによって、乱数レジスタ3188を

50



許容状態にすることと、乱数生成範囲の設定を同時に実行している（図 8 9 参照）が、ハードウェアを用いて許容状態にするようにしてもよい。なお、本実施形態の遊技台では乱数レジスタ 3 1 8 8 が許容状態になっても新たに乱数をラッチするまでは乱数は残った状態になるが、可能であれば乱数レジスタ 3 1 8 8 の値を所定の値に設定（例えば 0 に設定）するようにしてもよい。この場合には、異常の可能性のある乱数を使用させないようにすることができる。

#### 【0758】

上記説明した再設定は、WDT 3 1 4 1 によるユーザリセットが実行された場合には、そのユーザリセットが実行されてから、駆動電圧の判定（図 8 8 のステップ S 3 0 0 3）を経て実行される（図 8 9 のステップ S 3 0 5 3）。この判定処理に要する時間は電圧の供給具合に依存するため、一定ではない。このため再設定を反映した乱数が出力され始めるタイミングにランダム性を持たせることができ、不正防止に効果がある。なお上述した、CPU 3 0 4 からの更新命令が来た時点で乱数の最大値を更新する構成の場合にも、同様の効果がある。

10

#### 【0759】

なお、上記の説明では、電圧の変化によって生じる問題およびその対処法についての一例について説明したが、他にも例えば乱数生成回路 3 1 8 のスタート値選択回路 3 1 8 3 b 内の値が書き換わってしまう可能性もある。この場合、本来生成されるはずのない値が、乱数更新回路 3 1 8 3 から出力されてしまう虞がある。このような問題に対しても、上記の乱数生成範囲の例と同様に、スタート値選択回路 3 1 8 3 b 内の値を復帰処理の要因が発生した直前の状態に設定し直す処理を実行させることにより、上記のような問題が生じないようにすることができる。また、乱数生成範囲に応じて、別途スタート値選択回路 3 1 8 3 b 内の値を設定し直すようにしてもよい。すなわち、乱数生成回路 3 1 8 で設定可能な値を必要に応じて適切に設定し直す処理を実行させることにより、電圧の変化により異常な動作を起こさないようにすることができる。

20

#### 【0760】

また、電圧の変化以外にも、例えば、指定エリア外走行によって予期せぬ動作が生じた場合にも、上記のような問題が生じる虞がある。特に、指定エリア外走行の場合は、プログラムミス等によって同じ状況が再現される可能性が高く、電圧が変化した場合よりも不正をしやすくなる虞がある。従って、本実施形態では、指定エリア外走行が生じた場合でも、図 9 0 のステップ S 3 0 5 3 a およびステップ S 3 0 5 3 b による再設定を行うようにしている。

30

#### 【0761】

ここで、上記説明を踏まえて、図 8 8 の主制御部メイン処理における WDT 3 1 4 1 の起動開始のタイミング（ステップ S 3 0 1 1、ステップ S 3 0 1 5）について説明を補足する。通常 WDT では、プログラムの実行とは独立してカウントアップが行われている。この構成により、プログラムの実行に何らかの問題があっても、これに影響されずにリセット動作（ユーザリセット）を実行することができる。このため、WDT は初期設定時に起動（再起動）される必要がある。しかし、この起動タイミングによっては、問題が生じることがある。この問題について、図 8 8 におけるステップ S 3 0 0 1 の初期設定 1 の直後に WDT を起動する場合を例に挙げて説明する。

40

#### 【0762】

ユーザリセットによって RAM 3 0 8 に復帰用のデータが記憶されている状態で、主制御部メイン処理が開始されたとする。まず、ステップ S 3 0 0 1 によって初期設定 1 が実行され、次いで WDT が起動する。次に、駆動電圧が十分確保されるまでステップ S 3 0 0 3 による待機処理が実行される。

#### 【0763】

例えば供給電圧の不安定に起因して上記のユーザリセットが実行された場合等に、この待機処理の実行時間が長くなる場合がある。ここで待機処理に時間がかかると、後のステップ S 3 0 0 7 やステップ S 3 0 0 9 の実行中に WDT がタイムアウトして、再度ユーザ

50

リセットが実行されてしまう場合がある。この場合、再度主制御部メイン処理が最初から実行されるため、RAM 308に記憶されている復帰用のデータがレジスタに戻されるまでの処理時間が長くなってしまう。

#### 【0764】

さらに、例えばステップS3009において、RAM 308のデータをレジスタに戻すとともに、RAM 308のデータを消去する処理が実行されている場合には、RAM 308のデータの一部が消去されている状態でユーザリセットが実行される可能性もある。この場合には、再度主制御部メイン処理が最初から実行されても、RAM 308の復帰用のデータは不完全なものであり、ステップS3007で復帰可能と判定されないことになる。仮にステップS3007で復帰可能と判定されてもステップS3009のデータ書き戻し処理で完全にユーザリセット前の状態に復帰させることができない。

#### 【0765】

このような問題を回避するため、本実施形態では、WDT 3141によるタイムアウトではシステムリセットを実行するようにしているが、本実施形態ではさらに、WDT 3141によるタイムアウトでユーザリセットを実行してもよいように、駆動電圧の判定後から主制御部タイマ割込み処理が許可されるまでの間にWDT 3141を起動するように構成している（ステップS3011、ステップS3015）。なお、上記構成は一例であり、ユーザリセット後に主制御部メイン処理が開始されてから、主制御部タイマ割込み処理が許可されるまでの間にWDT 3141がタイムアウトしないように、WDT 3141を起動させるようにすることが好ましい。また、ユーザリセットが行われてから、主制御部によって遊技の進行が開始される（本実施形態では、主制御部メイン処理のステップS3017～ステップS3019が開始され、かつ、主制御部タイマ割込み処理が開始される）までの間に、WDT 3141がタイムアウトしないように、WDT 3141を起動させることが好ましい。特に、ユーザリセット後に遊技を復帰させるための処理（遊技の進行を開始可能にするための処理）に、液晶画像表示装置（装飾図柄表示装置）の初期設定処理を実行させるための待機処理が含まれる場合には、上記の問題がより生じやすくなるため、上記説明したタイミングでのWDT 3141の起動がより効果的である。さらに、主制御部タイマ割込み処理の最初の処理において、WDT 3141を起動させるようにしても、上記の問題を回避することができる。なお、上記説明ではユーザリセットが実行された場合を例に説明しているが、システムリセットが実行された場合であっても同様である。

#### 【0766】

ここで、図99を用いて、図79で説明した乱数更新回路3183の変形例について説明する。同図は、図79で説明した乱数更新回路3183の変形例を示す図である。この図99（a）に示す乱数更新回路3183nは、図79で説明した乱数更新回路3183に、乱数列変更回路3183dを加えたものである。

#### 【0767】

乱数列変更回路3183dは、予め内部に複数の乱数テーブルが設けられている。図99（b）には、65536個の乱数の並びを予め定めた複数の乱数列が示されている。これらの複数の乱数列を使用する際には、同じ乱数列を継続して使用するか、一つの乱数列を一周した時点で別の乱数列に変更するかを設定することができる。さらに、別の乱数列に変更する場合には、指示がなくても自動的に変更を行うか、あるいはその都度変更を指示するか否かを設定することができる。なお、カウンタ回路3183aから一巡信号を受けた時点で、一つの乱数列を一周したと判定される。

#### 【0768】

乱数列変更回路3183d内で使用する乱数列が決定されると、カウンタ回路3183aから出力されるカウント値に従って、この乱数列から値が取り出される。この取り出された値が乱数として出力される。例えば図99（b）において、乱数列Bを使用することが決定した場合に、カウント値「5」が入力されると、乱数として「9806」が出力される（図99（b）の太枠で示す部分参照）。

10

20

30

40

50

## 【0769】

ここで、最大値設定レジスタ3183cに最大値が設定された場合における処理の例について説明する。図81では、最大値設定レジスタ3183cに設定された最大値に従ってカウンタ回路3183aから出力される値の範囲が変更される例について説明した。なお、この図81では、このカウンタ回路3183aから出力される値をそのまま乱数として用いていた。ここでは、カウンタ回路3183aから出力される値を乱数ではなくカウンタ値として用い、乱数列変更回路3183dに入力する。

## 【0770】

例えば、最大値が9999に設定された場合、0～9999のカウント値がカウンタ回路3183aから出力されることになる。ここで、乱数列変更回路3183dには、最大値設定レジスタ3183cから最大値データを取得させ、乱数列の値のうち最大値を超える値を削除して空いた部分を詰める処理を実行させる。この処理により0～9999の値をランダムに並べた乱数列が用意されることになる。この乱数列を用いて上記カウント値に対応する値を乱数として出力させると、乱数生成範囲を0～9999とする乱数を出力させることができる。なお、上記方法は一例であり、例えば、カウンタ値に対応する値を乱数列から参照したときに最大値を超えている場合には以降の値を順次参照し、参照された値が最大値を超えない値であればその値を出力する、というように他の方法を用いてもよい。

## 【0771】

ここで、上記の乱数更新回路3183nを用いた場合における、乱数監視回路3184における異常の検出例について説明する。この場合には、1回の乱数の更新前後においてカウンタ回路3183aから出力されるカウント値の値を比較し、同じカウント値が出力されていないか否かをチェックする。仮に同じカウント値が出力された場合には、異常を示す情報が出力される。なお、チェックのタイミングについては、この乱数更新周期よりも長い所定の周期毎であってもよい。なお、乱数監視回路3184に乱数列変更回路3183dから出力される乱数を監視させてもよいが、上記したようにカウンタ回路3183aから出力されるカウント値を監視するようにすることで監視精度を高めるとともに、監視負担を低減することができる。

## 【0772】

また、図77に示す乱数生成回路318において説明したことと同様に、乱数生成回路318の外部に乱数監視回路3183nを設けてもよく、また、乱数更新回路3183nの内部に乱数監視回路3184を設けてもよい。

## 【0773】

次に、図100を用いて、図4に示す払出制御部600が実行する払出制御部メイン処理について説明する。なお、同図は払出制御部メイン処理の流れを示すフローチャートである。

## 【0774】

図4に示す払出制御部600は、CPU、RAM、ROM、I/Oポート、および電圧監視回路を備えている。払出制御部600が備える電圧監視回路も、主制御部300が備える電圧監視回路と同じく、電源管理部650から払出制御部600に供給している電源の電圧値が所定の値（本実施形態では9V）未満である場合に電圧が低下したことを示す低電圧信号を払出制御部600のCPUに出力する。なお、主制御部300の電圧監視回路338から低電圧信号が出力されていても、この払出制御部600の電圧監視回路からは低電圧信号は出力されていないことはある。

## 【0775】

払出制御部600には、主制御部300のCPU304から電源投入情報を含めたコマンドが送信されてくる。このコマンドを受信したことに基づいて、払出制御部600のCPUは、払出制御部メイン処理を開始する。なお、主制御部300のCPU304から電源投入情報を含めたコマンドを受信する前に払出制御部メイン処理を開始してもよく、該コマンドを受信するまで所定の処理（例えば、後述するステップS3603～ステップS

3610)を実行せずにスキップするようにしてもよい。

【0776】

まず、ステップS3501では、初期設定1を行う。この初期設定1では、払出制御部600のCPUのスタックポインタ(SP)へのスタック初期値の設定等を行う。ステップS3502では、払出制御部600の電圧監視回路から低電圧信号が出力されているか否か、すなわち低電圧信号がオンであるか否かを判定する。低電圧信号がオンの場合(電源の遮断を検知した場合)にはステップS3502の処理を繰り返し実行し、低電圧信号がオフの場合(電源の遮断を検知していない場合)にはステップS3503に進む。

【0777】

ステップS3503では、初期設定2を行う。この初期設定2では、後述する払出制御部タイマ割り込み処理を定期毎に実行するための周期を決める数値をカウンタ・タイマに設定する処理、払出制御部600のRAMへの書き込みを許可する設定、I/Oポートの初期設定等を行う。

【0778】

ステップS3504では、電源の遮断前(電断前)の状態に復帰するか否かの判定を行い、電断前の状態に復帰しない場合(パチンコ機100を初期状態にする場合)にはステップS3506に進み、電断前の状態に復帰する場合にはステップS3505に進む。具体的には、このステップS3504でも、主制御部300メイン処理のステップS3009と同様な処理が行われ、RAMクリアが必要な場合には、パチンコ機100を初期状態にすべくステップS3506に進む。一方、RAMクリアが必要でない場合には、払出制御部600のRAMから電源ステータスの情報を読み出し、電源ステータスの情報が、サスペンドを示す情報でない場合にはパチンコ機100を初期状態にすべくステップS3506に進み、サスペンドを示す情報である場合には払出制御部600のRAMについてチェックサムを実行し、チェックサムの結果が正常である場合には電断前の状態に復帰すべくステップS3505に進み、チェックサムの結果が異常である場合には、パチンコ機100を初期状態にすべくステップS3506に進む。同様に電源ステータスの情報が「サスペンド」以外の情報を示している場合にもステップS3506に進む。

【0779】

ステップS3505では、復電時処理を行う。この復電時処理では、払出制御部600のRAMの記憶領域のうち、復電時にクリアすべき記憶領域(コマンドを格納するためのコマンドバッファ、エラー状態を記憶するためのエラーステータスなどを除く記憶領域)の初期化などを行う。

【0780】

ステップS3506では、初期化処理を行う。この初期化処理では、割り込み禁止の設定、スタックポインタへのスタック初期値の設定、払出制御部600のRAMの所定の領域(例えば、全ての記憶領域)の初期化などを行う。

【0781】

ステップS3507では、初期設定3を行う。この初期設定3では、払出制御部600のRAMに設けたエラーステータス記憶領域に記憶したエラーステータスのうち、不正払出エラーと払出超過エラー以外の情報をクリアしたり、割り込み許可の設定などを行う。

【0782】

ステップS3508では、主制御部300から入力したデータの中に未解析データがあるか無いかを判定し、未解析データがある場合にはステップS3509でコマンド解析処理を行い、未解析データがない場合にはステップS3510に進む。

【0783】

ステップS3510では、ステップS3502と同じく、低電圧信号がオフであるか否かを監視し、低電圧信号がオフの場合(電源の遮断を検知していない場合)にはステップS3508に戻り、低電圧信号がオンの場合(電源の遮断を検知した場合)にはステップS3511に進む。

【0784】

10

20

30

40

50

ステップ S 3 5 1 1 では、電断時処理を行う。この電断時処理では、払出制御部 6 0 0 の R A M に設けたスタックポインタ退避領域に現在のスタックポインタの値を記憶し、上述の電源ステータス記憶領域にサスペンドを示す情報を設定する。また、払出制御部 6 0 0 の R A M の所定の領域（例えば全ての領域）に記憶している 1 バイトデータを初期値が 0 である 1 バイト構成のレジスタに全て加算し、チェックサム算出用数値記憶領域に記憶している値からその加算した結果を減算した値をチェックサム（電断時チェックサム）として算出し、算出した電断時チェックサムを上述のチェックサム算出用数値記憶領域に記憶し、払出制御部 6 0 0 の R A M への書き込みを禁止する設定などを行う。なお、ステップ S 3 5 1 1 の後にステップ S 3 5 0 2 と同じく低電圧信号がオフであるか否かを監視する処理を行い、低電圧信号がオフの場合（電源の遮断を検知していない場合）には払出し制御部メイン処理の先頭（ステップ S 3 5 0 1）に戻るようにしてもよい。

10

#### 【0785】

＜払出制御部タイマ割り込み処理＞

次に、図 1 0 1 を用いて、払出制御部 6 0 0 の C P U が実行する払出制御部タイマ割り込み処理について説明する。なお、同図は払出制御部タイマ割り込み処理の流れを示すフローチャートである。

#### 【0786】

払出制御部 6 0 0 は、所定の周期（本実施形態では 1 m s に 1 回）でタイマ割り込みを発生するカウンタ回路も備えており、このタイマ割り込みを契機として払出制御部タイマ割り込み処理を所定の周期で開始する。

20

#### 【0787】

ステップ S 3 6 0 1 では、タイマ割り込みスタート処理を行う。このタイマ割り込みスタート処理では、払出制御部 6 0 0 の C P U の各レジスタの値をスタック領域に一時的に退避する処理などを行う。払出制御部 6 0 0 には、ウォッチドッグタイマ（W D T）も用意されている。ステップ S 3 6 0 2 ではこの W D T のリスタートを行う。

#### 【0788】

ステップ S 3 6 0 3 では、入力ポート状態更新処理を行う。この入力ポート状態更新処理では、払出制御部 6 0 0 の I / O ポートの値を取得して、図 4 に示す払出センサ 6 0 4 等の状態などを検出する。ステップ S 3 6 0 4 では、タイマ更新処理を行う。このタイマ更新処理では、払出報知用 L E D の点灯／消灯時間、図 4 に示す払出モータ 6 0 2 の駆動／非駆動時間などを計時するためのタイマなどを含む各種タイマを更新する。

30

#### 【0789】

ステップ S 3 6 0 5 では、エラー監視処理を行う。払出制御部 6 0 0 の I / O ポートには、主制御部 3 0 0 から、下皿満タンエラー信号やガラス枠開放エラー信号等の各種のエラー信号が送られてくる。払出制御部 6 0 0 の R A M には、エラーステータス記憶領域や、遊技媒体貸出情報記憶領域や、払出完了数チェック記憶領域や、モータ駆動量記憶領域が設けられている。エラーステータス記憶領域には、各エラー信号に対応した情報が記憶される。このエラー監視処理では、各種のエラー信号がオンか否かを判定し、オンの場合には、エラーステータス記憶領域に対応したエラーを示す情報を記憶し、オフの場合には、エラーステータス記憶領域に対応したエラーの解除を示す情報を記憶する。また、主制御部 3 0 0 と払出制御部 6 0 0 との間の通信が可能かどうかを、主制御部からのコマンドが所定時間（例えば 1 0 0 0 m s）継続して受信できないか否かで判定し、エラーステータス記憶領域に、通信可能な場合には通信可能であることを示す情報を、反対に通信不可能な場合には通信不可能であることを（主制御通信エラー）を示す情報を記憶する。加えて、図 4 に示すカードユニット（C R ユニット）6 0 8 と払出制御部 6 0 0 との間の通信が可能かどうかについても、カードユニット 6 0 8 からの信号が所定時間（例えば 1 0 0 0 m s）継続して受信できないか否かで判定し、エラーステータス記憶領域に、通信可能な場合には通信可能であることを示す情報を、反対に通信不可能な場合には通信不可能であることを（C R ユニット未接続エラー）を示す情報を記憶する。また、払出制御部 6 0 0 の I / O ポートには、エラー解除信号も送られてくる。このエラー監視処理では、エラー解

40

50

除信号がオンであるか否かも判定し、エラー解除信号がオンである場合には、エラーステータス記憶領域に記憶している情報を初期化して、エラーを解除する。

#### 【0790】

ステップS3606では、CRユニット通信処理を行う。このCRユニット通信処理では、図4に示すインターフェース部606から遊技媒体貸出信号を受信して遊技媒体貸出信号センサ信号がオンであるか否かを判定し、遊技媒体貸出信号がオンの場合（インターフェース部606からの球貸要求を入力した場合）には、RAMに設けた遊技媒体貸出情報記憶領域に遊技媒体の貸出要求があったことを示す情報を記憶する。

#### 【0791】

ステップS3607では、払出動作管理処理を行う。上述のエラーステータス記憶領域には、不正払出エラーの情報や払出超過エラーの情報も記憶されている。この払出動作管理処理では、エラーステータス記憶領域から不正払出エラーの情報、および払出超過エラーの情報を読み出し、いずれのエラーも発生していない場合には、図4に示す払出センサ604からの信号（以下、払出センサ信号と称する場合がある）に基づいて払出個数の監視を行う。すなわち、所定のエラーが発生している場合には払出モータ602の駆動停止、すなわち払出装置からの賞媒体（例えば遊技球）の払出を停止するようにしている。具体的には、払出センサ604の信号を検出して払出センサ信号がオンであるか否かを判定し、払出センサ信号がオンの場合（払出センサを球が通過した場合）には払出完了数チェックに1を加算して払出完了数チェック記憶領域に記憶する。また、賞球および貸出球の要求が無いときに払出センサ信号がオンになった場合には、上述のエラーステータス記憶領域に不正払出エラーを示す情報を記憶し、賞球数または貸出球数が各々の要求数を超え、その超過数が所定数以上になった場合には、上述のエラーステータス記憶領域に払出超過エラーを示す情報を記憶する。

#### 【0792】

また、上述のエラーステータス記憶領域から、下皿満タンエラーを表す情報、不正払出エラーの情報、および払出超過エラーの情報を読み出し、いずれのエラーも発生していない場合に、払出開始監視処理、初期位置検索動作処理、通常払出動作処理、リトライ動作処理、逆回転動作処理のいずれかの処理を行う。

#### 【0793】

上述のごとく、払出装置152は、払出モータ602によって回転可能に構成されたスプロケットを備えている。このスプロケットは、タンクレール154を通過して払出装置152内に流下した遊技球を一時的に滞留させると共に、払出モータ602を駆動して所定角度だけ回転することにより、一時的に滞留した遊技球を払出装置152の下方へ1個ずつ送り出す。払出開始監視処理では、貸出要求数、および賞球要求数が0であり、次賞球要求数が0以外の場合は、賞球要求数に次賞球要求数をセットし、次賞球要求数をクリアする。また、払出モータ602の位置が不確定の場合（動作モードが初期位置検索動作モードの場合）には、払出完了数チェックから1を減算して払出完了数チェック記憶領域に記憶し、払出モータ602の位置が確定している場合（動作モードが通常払出動作モードの場合）には、払出完了数チェックとして払出完了数チェック記憶領域に0を設定する。また、賞球要求数を、払出モータ602を駆動する量（モータ駆動量）に変換し、これをRAMに設けたモータ駆動量記憶領域に記憶すると共に、RAMに設けたモータ制御データテーブルを参照してモータ駆動量に対応するモータ駆動制御データを選択し、正転を示すモータ駆動制御データをI/Oポートを介して、払出モータ602を制御するモータ制御回路に出力する。これにより、モータ制御回路は払出モータ602のモータの励磁位置を所定回変化させてスプロケットを正方向に回転駆動する。

#### 【0794】

初期位置検索動作処理および通常払出動作処理では、払出モータ602の駆動終了後に、払出完了数チェック記憶領域から払出完了チェックを読み出し、払出完了チェックが0の場合には、払出開始監視処理を実行する準備を行い、払出完了チェックが0以外の場合には、エラーステータス記憶領域に払出装置エラーを示す情報を設定すると共に、リト

ライ動作処理を実行する準備を行う。

【0795】

リトライ動作処理では、所定の時間が経過するのを待ち（リトライ動作開始待ちタイマが0になるのを待ち）、リトライ動作開始待ちタイマが0になった場合には、逆回転動作処理を実行する準備を行う。逆回転操作処理では、上述のモータ制御データテーブルを参照してモータ駆動量に対応するモータ駆動制御データを選択し、逆転を示すモータ駆動制御データをI/Oポートを介してモータ制御回路に出力する。これにより、モータ制御回路は払出モータ602の励磁位置を所定回変化させてスプロケットを逆回転駆動する。また、逆回転操作処理では、払出モータ602の駆動終了後に払出開始監視処理を実行する準備を行う。

10

【0796】

ステップS3608では、払出モータ駆動監視処理を行う。この払出モータ駆動監視処理では、駆動開始監視処理、加速駆動処理、定速駆動処理、ブレーキ駆動処理、駆動終了処理のいずれかの処理を行う。

【0797】

駆動開始監視処理では、上述のエラーステータス記憶領域から下皿満タンエラーの情報、不正払出エラーの情報、および払出超過エラーの情報を読み出し、いずれのエラーも発生していない場合には、上述のモータ制御データテーブルを参照してモータ駆動量に対応するモータ駆動制御データを選択し、正転を示すモータ駆動制御データをI/Oポートを介してモータ制御回路に出力する。これにより、モータ制御回路は払出モータ602の励磁位置を所定回変化させスプロケットを正方向に回転駆動する。

20

【0798】

加速駆動処理および定速駆動処理では、スプロケットの初期位置検索動作中、または、逆回転動作中の場合を除き、払出モータ602の励磁位置を16回変化させるごとに払出完了数チェックから1を減算して払出完了数チェック記憶領域に記憶する。また、更新後の払出完了数チェックが-4未満になった場合には、ブレーキ駆動処理を実行する準備を行う。さらに、上述の遊技媒体貸出情報記憶領域から遊技媒体貸出情報を読み出して、遊技媒体の貸出要求があったことを示す情報の有無を判定し、遊技媒体の貸出要求があったことを示す情報がある場合（賞球の払出中にインターフェース部606からの球貸要求を入力した場合）にも、ブレーキ駆動処理を実行する準備を行う。ブレーキ駆動処理では、所定の時間が経過するのを待ち（モータ駆動管理タイマが0になるのを待ち）、モータ駆動管理タイマが0になった場合には、駆動終了処理を実行する準備を行い、駆動終了処理では、モータ駆動の後処理を行う。

30

【0799】

ステップS3609では、外部出力信号設定処理を行う。この外部出力信号設定処理では、払出制御部600のRAMに記憶している遊技情報（例えば払出センサ信号を入力するたびに出力する賞球信号）を、情報出力回路（図示せず）を介してパチンコ機100とは別体の情報入力回路350に出力する。

【0800】

ステップS3610では、ポート出力処理を行う。このポート出力処理では、払出制御部600のI/Oポートから、図4に示す発射制御部630に発射を許可する発射許可信号を出力する。ただし、上述の主制御通信エラーやCRユニット未接続エラーが生じている場合には、発射許可信号を出力しない状態に設定する。したがって、主制御部300が電断し、払出制御部600には電力が供給されている状態では、上述のステップS3605において主制御通信エラーが認められ、このポート出力処理で、発射許可信号を出力しない状態に設定される。

40

【0801】

ステップS3611では、タイマ割り込みエンド処理を行う。このタイマ割り込みエンド処理では、ステップS3601で一時的に退避した各レジスタの値を元の各レジスタに設定したり、割り込み許可の設定などを行い、払出制御部タイマ割り込み処理は終了にな

50

る。

#### 【0802】

##### <第2内部情報レジスタ>

続いて、図84に第1内部情報レジスタを示した割込み制御回路3100の第2内部情報レジスタについて説明する。図102は、割込み制御回路3100の内部情報レジスタ3101に用意された第2内部情報レジスタを説明するための図である。

#### 【0803】

図102に示すように、第2内部情報レジスタは8ビットのレジスタであるが、7～3ビット目は「0」が設定されたままであり、使用されていない。また、2～0ビット目は、初期値は「0」である。

10

#### 【0804】

本実施形態では、電源投入によって、電圧監視回路338からシステムリセット信号がリセット制御回路314に入力される。すると、リセット制御回路314からは、マイクロプロセッサ3000の内部の回路に対し内部バス3300を介して、システムリセット信号1が出力される。また、本実施形態では、WDT3141がタイムアウトになった場合には、リセット制御回路314からは、マイクロプロセッサ3000の内部の回路に対し内部バス3300を介して、WDTのタイムアウト信号が出力されたことを表すシステムリセット信号2が出力される。さらに、CPU304が所定の範囲以外のアドレスを参照（指定エリア外走行）した場合には、リセット制御回路314からは、マイクロプロセッサ3000の内部の回路に対し内部バス3300を介して、指定エリア外走行禁止信号が出力されたことを表すシステムリセット信号3が出力される。

20

#### 【0805】

割込み制御回路3100は、内部バス3300を介してシステムリセット信号1を取得すると、2ビット目に「1」を設定する。すなわち、第1内部情報レジスタの2ビット目は、直前に発生したリセット要因が電源投入によるリセットであることを示すビットに相当する。

#### 【0806】

また、割込み制御回路3100は、内部バス3300を介してシステムリセット信号2を取得すると、1ビット目に「1」を設定する。すなわち、第1内部情報レジスタの1ビット目は、直前に発生したリセット要因がWDT3141のタイムアウトであることを示すビットに相当する。

30

#### 【0807】

さらに、割込み制御回路3100は、内部バス3300を介してシステムリセット信号3を取得すると、0ビット目に「1」を設定する。すなわち、第1内部情報レジスタの0ビット目は、直前に発生したリセット要因がCPU304の指定エリア外走行であることを示すビットに相当する。各ビットの値は、CPU304がその値を読み取るとクリアされる（初期値の「0」になる）ようになっている。なお、この第2内部情報レジスタを、図84に示す第1内部情報レジスタで構成してもよい。

#### 【0808】

##### <第3内部情報レジスタ>

図103は、割込み制御回路3100の内部情報レジスタ3101に用意された第3内部情報レジスタを説明するための図である。図103に示すように、第3内部情報レジスタも8ビットのレジスタであるが、7～2ビット目は「0」が設定されたままであり、使用されていない。

40

#### 【0809】

図103に示す第3内部情報レジスタの1ビット目は、WDT3141がタイムアウトした場合に、ユーザリセット動作を行わせるか、あるいはシステムリセット動作を行わせるかを設定するためのビットである。ユーザリセット動作を行わせる場合には「0」を設定し、システムリセット動作を行わせる場合には「1」を設定する。

#### 【0810】

50



また、0ビット目は、CPU 304の指定エリア外走行が生じた場合に、ユーザリセット動作を行わせるか、あるいはシステムリセット動作を行わせるかを設定するためのビットである。このビットでも、ユーザリセット動作を行わせる場合には「0」を設定し、システムリセット動作を行わせる場合には「1」を設定する。

#### 【0811】

ここで説明した1ビット目と0ビット目それぞれに設定される値は、内蔵ROM 306のプログラム管理エリアに記憶されている。プログラム管理エリアに記憶されているこれらの値は、図88に示す主制御部メイン処理における初期設定2（ステップS3007）において、プログラム管理エリアから呼び出され、1ビット目と0ビット目それぞれに設定される。本実施形態では、いずれのビットにもシステムリセット動作を行わせる「1」が設定される。なお、1ビット目にユーザリセット動作を行わせる「0」を設定し、0ビット目にシステムリセット動作を行わせる「1」を設定してもよい。また、1ビット目にシステムリセット動作を行わせる「1」を設定し、0ビット目にユーザリセット動作を行わせる「0」を設定してもよい。

10

#### 【0812】

なお、この第3内部情報レジスタも、図84に示す第1内部情報レジスタの空きビット（未使用ビット）で構成してもよく、あるいは、上述の図102に示す第2内部情報レジスタの空きビット（未使用ビット）で構成してもよい。また、リセット発生時には、ROM 306のプログラム管理エリアに記憶されたリセット動作設定データの値を直接参照するようにしてもよい。この場合、第3内部情報レジスタに値を設定する処理は行わなくてもよい。

20

#### 【0813】

##### <第4内部情報レジスタ>

図104は、割込み制御回路3100の内部レジスタ3101に用意された第4内部情報レジスタを説明するための図である。図104に示すように、第4内部情報レジスタは8ビットのレジスタであるが、7～2ビット目は「0」が設定されたままであり、使用されていない。

#### 【0814】

この第4内部情報レジスタの1ビット目は、乱数生成回路318の外部クロック（更新クロック）RCKの異常を示すビットである。更新クロックに異常がある場合には、第4内部情報レジスタの1ビット目から「1（異常あり）」の情報が読みだされ、更新クロックに異常がない場合には、第4内部情報レジスタの1ビット目から「0（異常なし）」の情報が読みだされる。

30

#### 【0815】

また、0ビット目は、リセットが発生した場合に、直前のリセット要因がシステムリセット（電源投入によるリセット）であるか否かを示すためのビットである。割込み制御回路3100は、内部バス3300を介してシステムリセット信号1を取得すると、この0ビット目に「1」を設定する。直前のリセット要因がシステムリセットである場合には、第4内部情報レジスタの0ビット目から「1（システムリセット発生）」の情報が読みだされ、直前のリセット要因がシステムリセットでない場合（リセット要因がWDT 3141のタイムアウトである場合、または、リセット要因がCPU 304の指定エリア外走行である場合）には、第4内部情報レジスタの0ビット目から「0（システムリセット未発生）」の情報が読みだされる。

40

#### 【0816】

##### <その他の内部情報レジスタ>

また、図示は省略するが、マイクロプロセッサ3000は、WDT 3141を制御するレジスタとして、スタートレジスタとクリアレジスタを備えている。スタートレジスタは、WDT 3141の起動／停止を制御するための8ビット長のレジスタであり、WDT 3141を起動するときには起動を指示する値（例えば、CCH）を設定し（書き込み）、WDT 3141を停止するときには停止を指示する値（例えば、33H）を設定するよう

50

に構成されている。

#### 【0817】

また、このスタートレジスタは、リセット時にプログラム管理エリアに「ソフトウェアによる起動」を設定（選択）したときに有効となるレジスタであり、その場合、停止を指示する値（例えば、33H）が初期値として設定される。したがって、「ソフトウェアによる起動」を設定（選択）したときにWDT3141を起動するためには、遊技制御プログラムによって（手動で）起動を指示する値（例えば、CCH）を設定する必要がある。一方、リセット時に「ソフトウェアによる起動」を設定（選択）しなかったときは、スタートレジスタには、起動を指示する値（例えば、CCH）が遊技制御プログラムによらずに（自動的に）初期値として設定され、遊技制御プログラムによらずに（自動的に）WDT3141が起動される。

10

#### 【0818】

クリアレジスタは、WDT3141のクリアとリスタートを制御するための8ビット長のレジスタであり、所定の第一の値（例えば、55H）を設定した後に、所定の第二の値（例えば、AAH）を設定したときにWDT3141がクリアされ、直後にリスタートされる。なお、クリアレジスタの値を読み込んだときは、所定の固定値（例えば、FFH）が読みだされる。

#### 【0819】

ここでもう一度、システムリセットの流れについて、乱数更新との関係を含めて説明する。図105は、システムリセットの流れを乱数更新との関係を含めて示す図である。これまで、静電気などの要因により瞬断が生じた場合に、WDT3141のタイムアウトによってリセット制御回路314からシステムリセット信号（ここではシステムリセット信号2）が出力され、リセット動作が行われることを説明した。また、WDT3141のタイムアウトに限らず、例えば、CPU304の指定エリア外走行が生じた場合などにも、リセット制御回路314からシステムリセット信号（ここではシステムリセット信号3）が出力され、リセット動作が行われる。このリセット動作では、遊技台はセキュリティモードに移行し、セキュリティチェック処理（図76に示すステップSH05）が実行される。

20

#### 【0820】

WDT3141のタイムアウトやCPU304の指定エリア外走行は、遊技処理の正常な進行を妨げる異常の一例に相当する。また、システムリセット信号2およびシステムリセット信号3は復帰指示の一例に相当する。さらに、図76に示すリセット制御回路314は、上記異常を検出する異常検出部の一例に相当する。

30

#### 【0821】

なお、図105では、図示省略したが、セキュリティモードではまず、図76に示すステップSH03の第1内部回路初期化処理が実行され、CPU304のコアやタイマ回路311、カウンタ回路312、パラレル入力ポート3102、RAM308アクセスプロテクトレジスタ、割込み制御回路3100、乱数生成回路318を制御するレジスタといった内蔵レジスタの値が初期化される。なお、乱数生成回路318を制御するレジスタが初期化されることから、乱数更新範囲が再設定されるとともに、乱数値の初期値も再設定される。CPU304の指定エリア外走行が生じたときは特に、乱数更新範囲に何らかの影響が及ぼされている可能性があるため、乱数更新範囲の再設定を行うことが好ましい。

40

#### 【0822】

続いて、固定延長処理（ステップSH07）が行われ、この固定延長処理が終了すると、リセット制御回路314は、マイクロプロセッサ3000の外部にある周辺制御回路に対して、リセット出力端子（XRSTO端子）からリセット信号を出力する。すなわち、XRSTO端子からのリセット信号（リセット出力）は、固定延長処理の終了後（セキュリティモード中）に出力され、ユーザモードの開始とは異なるタイミングで出力される。

#### 【0823】

次に、ランダム延長処理（ステップSH09）が行われ、遅延処理の時間が不定になる

50

。ランダム延長処理が終了すると、遊技台はユーザモードに移行し、遊技制御用プログラム（ここでは主制御部メイン処理）が開始される。

#### 【0824】

この遊技制御用プログラムでは、最初に、各種の初期設定（図88に示すステップS3001～ステップS3015）が行われる。この初期設定では、図106（a）に示すように、上述の第4内部情報レジスタの1ビット目から「更新クロック異常を示すビット」の値を読み出した後に、読み出した値を汎用レジスタにセットする（書き込む）処理を実行する一方で、第4内部情報レジスタの0ビット目から「リセット要因がシステムリセットであるか否かを示すビット」を読み出す処理は実行するが、読み出した値を汎用レジスタにセットする（書き込む）処理を実行しないように構成している。

10

#### 【0825】

換言すると、同図（b）に示すように、マイクロプロセッサ3000は、第1の要因（例えば、乱数生成回路318の更新クロックの状態）に基づいて変化する内部情報レジスタのビット（例えば、第4内部情報レジスタのビット1）の値を読み出して、読み出した値を汎用レジスタにセットする（書き込む）一方で、第1の要因とは異なる第2の要因（例えば、リセット要因）に基づいて変化する内部情報レジスタのビット（例えば、第4内部情報レジスタのビット0）の値を読み出すが、読み出した値を汎用レジスタにセットしない（書き込まない）ように構成されている。

#### 【0826】

初期設定を終えると、続いて、ソフトウェア乱数の更新処理（ステップS3017およびステップS3019）が行われ、次いで、最初に発生する大当りの乱数取得タイミングに到達する。ここで、最初に発生する大当りの乱数取得タイミングに乱数を不正に取得されると、強制的に大当りにされてしまうことがある。しかしながら、ランダム延長処理によって遅延処理の時間が不定になっているため、上述のリセット出力を不正にとらえて、最初に発生する大当りの乱数取得のタイミングを図ろうとしても、上記所定時間も不定になり、不正行為を防止することができる。

20

#### 【0827】

図107は、上記図75に示すマイクロプロセッサ3000のリセット制御回路314の変形例を示した図である。この変形例に係るリセット回路1314は、図75に示すリセット制御回路314が備える指定エリア外走行禁止回路3142、ウォッチドックタイマ（WDT）3141に替えて、信号出力タイミング制御回路1314aを備えている。この信号出力タイミング制御回路1314aは、2つのリセット入力端子XSRST1、リセット入力端子XSRST2から入力するリセット入力信号に基づいて、リセット出力端子XRSTOから出力するリセット出力信号の出力タイミングを決定するための回路である。

30

#### 【0828】

図108は、主制御部300と第1副制御部400とのうち主制御部300のみでWDT3141のタイムアウトが生じた場合の例を示すタイミングチャートである。図108に示すタイミングチャートでは、図の左から右に向かって時間（T）が経過していく。図の左端の状態では、主制御部300にも第1副制御部400にも12Vの電圧が供給され、主制御部300も第1副制御部400も通常処理を行っている。この例では、主制御部300は特図の変動表示を開始し、第1副制御部400は、主制御部300による特図変動表示の開始に合わせて、ここでは不図示の第2副制御部500を介して、装飾図柄表示装置208に装飾図柄の変動表示を開始させる。図108に示すタイミングチャートにおける（a）～（g）の符号は、符号が記された位置のタイミングを示すものであり、図108の下方には、そのタイミングにおける装飾図柄表示装置208の様子が示されている。

40

#### 【0829】

主制御部300が特図変動表示を開始して間もなく主制御部300では瞬断が生じ、一時的な電圧降下により、電断時処理が行われ、待機状態に移行する。図97（b）を用い

50

て説明したように、待機状態では、主制御部 300 の WDT 3141 がリスタートされず、WDT 3141 がタイムアウトする。リセット制御回路 314 からは、WDT 3141 のタイムアウトに基づいてシステムリセット信号 2 が出力される。

#### 【0830】

システムリセット信号 2 が出力されたことにより、主制御部 300 は、図 76 に示すリセット処理を開始する。主制御部 300 では、このリセット処理で、第 1 内部回路初期化処理（ステップ SH03）が実行された後に、セキュリティチェック処理（ステップ SH05）が実行される。そして、セキュリティチェック処理が終了してもユーザモード（図 88 に示す主制御部メイン処理）にはすぐには移行せず、遅延処理が行われ、リセット処理が継続する。この遅延処理では、図 76 に示す固定延長処理（ステップ SH07）が実行される。なお、上述のごとく、固定延長処理が終了すると、リセット制御回路 314 は、マイクロプロセッサ 3000 の外部にある周辺制御回路に対して、リセット出力端子（XRSTO 端子）からリセット信号を出力するが、このリセット信号は、第 1 副制御部 400 に送信されるものではない。さらに、遅延処置では、ランダム延長処理（ステップ SH09）が実行され、最大 30 秒のランダム時間だけさらに、ユーザモードへの移行が遅延される。これらステップ SH07 およびステップ SH09 は、本発明にいう遅延処理の一例に相当する。

#### 【0831】

また、上述のごとく、電断時処理が開始されてから所定時間（例えば、1000ms）が経過するまでに、図 4 に示す払出制御部 600 が主制御部 300 からのコマンドを受信できない場合には、発射制御部 630 への発射許可信号の出力を停止する。

#### 【0832】

一方、この例では、主制御部 300 では瞬断が生じたものの、第 1 副制御部 400 には継続して 12V の電圧が供給されており、主制御部 300 が電断時処理やリセット処理を行っている間も、第 1 副制御部 400 は通常処理を継続する。この結果、演出実行手段は演出を続けており、装飾図柄表示装置 208 でも、装飾図柄の変動表示が継続されており、主制御部 300 が電断時処理を開始して間もない（b）のタイミングでは、左図柄表示領域 208a で装飾図柄の停止表示が行われる（いわゆる左図柄停止）。また、（c）のタイミングでは、右図柄表示領域 208c で装飾図柄の停止表示が行われ（いわゆる右図柄停止）、リーチ状態になる。続いて、（d）のタイミングでは、スペシャルリーチ（ここではいわゆるトリプルリーチ）に発展し、（e）のタイミングでは、リーチはずれ演出が一旦行われ、（f）のタイミングでは、装飾図柄の組合せ（ここでは「装飾 5－装飾 4－装飾 5」）が揺れ変動している。第 1 副制御部 400 は、主制御部 300 から図柄変動停止コマンドが送られてきたことに基づいて、装飾図柄の組合せを完全停止させて確定停止を行う。ところが、主制御部 300 が電断時処理やリセット処理を行っている間は、第 1 副制御部 400 には主制御部 300 からコマンドが送信されない。

#### 【0833】

この結果、特図の変動時間が経過しても揺れ変動が継続することになる。いつまでも続く揺れ変動を見た遊技者は、パチンコ機 100 が壊れたのではないかと不安に思い、遊技店の店員を呼ぶことが期待でき、遊技店員に瞬断やリセットが発生したことを知らせることができる場合がある。すなわち、従来の遊技台では、遊技中に静電気などによる瞬間的な電圧低下でリセットが発生した場合にリセット処理を行なう技術が考案されているが、リセット処理には僅かな時間しかかからないため、遊技店側がリセットの発生を把握できず、遊技台に安定した制御を行なわせることができなかったが、本実施形態の遊技台によれば、遊技制御の開始を遅延させることで、遊技者を介してリセットの発生を遊技店員が把握できる。また、リセットの発生を遊技店員が把握できるため、リセットの発生要因を排除する（例えば、遊技台裏の配線等を離す）などして遊技台に安定した制御を行なわせることができる。さらに、この例では、リーチはずれ演出が行われているが、リーチ当り演出が行われてから揺れ変動がいつも以上に継続した場合には、大当りを期待していた遊技者に対して、パチンコ機 100 が壊れ、有利な状態にならないように変化してしまった

のではないかとといった遊技者の不安を一層煽ることができ、遊技者に店員を呼び出させ、遊技店員に瞬断が発生したことを知らせることができる場合がある。

#### 【0834】

また、リセット処理が行われている期間は、発射許可信号の出力が停止されており、発射装置110からの遊技球の発射ができない。そのため、装飾図柄表示装置208による表示は継続しているのに、図1に示す球発射ハンドル134をいくら操作しても遊技球が発射されず、遊技者は、パチンコ機100が壊れたのではないかと不安に思い、遊技店の店員を呼ぶことが期待でき、遊技店員に瞬断やリセットが発生したことを知らせることができる場合がある。すなわち、遅延処理中は発射を停止させることで、遊技者を介してリセットの発生を遊技店員が把握できる場合がある。

10

#### 【0835】

さらに、XRSTO信号が立ち下がるタイミングで主制御部300の各回路（例えば、駆動回路324～334）にクリア信号が出力され、電断時処理やリセット処理が行われている期間は、主制御部300が駆動する、普図表示装置210や、第1特図表示装置212や、第2特図表示装置214や、普図保留ランプ216や、第1特図保留ランプ218や、第2特図保留ランプ220や、高確中ランプ222等の遊技表示手段も表示されず、これらの非表示に気付いた、遊技者は、パチンコ機100が壊れたのではないかと不安に思い、遊技店の店員を呼ぶことが期待でき、遊技店員に瞬断やリセットが発生したことを知らせることができる場合がある。特に、普図保留ランプ216や、第1特図保留ランプ218や、第2特図保留ランプ220や、高確中ランプ222といった遊技者が獲得した権利に基づく状態を表示する表示器が、非表示になると、パチンコ機100が壊れて、これまで獲得した権利が消滅してしまったのではないかとといった遊技者の不安を一層煽ることができ、遊技者に店員を呼び出させ、遊技店員に瞬断やリセットが発生したことを知らせることができる場合がある。また、見方を変えれば、店員に遊技客に対してお詫びをする機会が与えられ、遊技客とのコミュニケーションが図れ、アットホームな遊技店を実現する手助けになる可能性がある。

20

#### 【0836】

主制御部300では、ランダム延長処理（ステップSH09）が終了すると、ここで初めてユーザモードに移行し、通常処理（図88に示す主制御部メイン処理）が開始される。図88に示す主制御部メイン処理では、主制御部300への供給電圧が12Vまで復帰しているため、図89に示す初期設定2が行われ、第1副制御部400には、ここで初めてクリア信号が出力される（ステップS3052）。第1副制御部400は、クリア信号を受信すると各種の初期設定を行う。この初期設定で、装飾図柄表示装置208に初期設定時の画面（デフォルト画面）を表示させる。なお、WDT3141でタイムアウトが生じた場合にも、装飾図柄表示装置208にはデフォルト画面が表示される。図108に示す（g）のタイミングでは、装飾図柄表示装置208に「お待ち下さい。」という文字表示がなされている。すなわち、装飾図柄表示装置208の画面は、いつまでたっても終わらなかった揺れ変動から「お待ち下さい。」という文字表示の画面に切り替わり、この一連の表示を見ていた遊技者は、パチンコ機100が壊れたのではないかと不安に思い、遊技店の店員を呼ぶことが期待でき、遊技店員に瞬断が発生したことを知らせることができる場合がある。

30

40

#### 【0837】

なお、主制御部メイン処理が開始されると、主制御部300から払出制御部600にもコマンドが送信されるようになり、発射制御部630へ発射許可信号が出力されて、遊技球の発射ができるようになる。

#### 【0838】

この例では、主制御部300がリセット処理から通常処理に移行するタイミングは、装飾図柄の組合せの揺れ変動中（（f）のタイミング以降）であったが、揺れ変動前（（f）のタイミングより前）であっても、装飾図柄表示装置208にはデフォルト画面が表示される。

50

## 【0839】

以上説明した図108では、WDT3141のタイムアウトが生じた場合の例であったが、CPU304の指定エリア外走行が生じた場合にも同様に、遊技者を介してリセットの発生を遊技店員が把握でき、遊技台に安定した制御を行なわせることができる。また、リセットがかかるようなその他の異常が生じた場合でも、システムリセット動作を行わせるようにしておけば、システムリセット中に遅延処理が実行され、遊技者を介してリセットの発生を遊技店員が把握でき、遊技台に安定した制御を行なわせることができる場合がある。

## 【0840】

以上の説明では、『電源の遮断が検出されるまで繰り返し実行される主処理（例えば、ステップS3017、S3019、および主制御部タイマ割込み処理）を含む遊技に関する遊技処理を実行する遊技制御部（例えば、主制御部300）、および前記遊技制御部が前記遊技処理を実行中に該遊技処理の正常な進行を妨げる異常（例えば、WDT3141のタイムアウトやCPU304の指定エリア外走行）を検出したことに基づいて復帰指示（例えば、システムリセット信号2, 3）を行う異常検出部（例えば、リセット制御回路314）が設けられた電氣的制御手段（例えば、マイクロプロセッサ3000）を備え、前記遊技制御部（例えば、主制御部300）は、前記復帰指示に基づいて前記遊技処理の実行を中止するものであって、前記電氣的制御手段（例えば、マイクロプロセッサ3000）は、前記遊技制御部が前記遊技処理の実行を開始するための復帰処理（例えば、システムリセット動作）を前記復帰指示に基づいて行い、該復帰処理を実行している最中に該遊技処理の開始を遅延させる遅延処理（例えば、ステップSH07, SH09）を実行するものであることを特徴とする遊技台。』についての説明がなされている。

## 【0841】

ここで、前記遊技処理の遊技処理プログラムを記憶した記憶部（例えば、内蔵ROM306）を備え、前記電氣的制御手段は、前記遊技処理の実行を再開するための初期化处理（例えば、ステップSH03の第1内部回路初期化处理）を前記復帰指示に基づいて行うものであって、前記遊技制御部は、前記復帰指示に基づいて前記遊技処理の実行を中止し、前記初期化处理が行われた後、前記記憶部から前記遊技処理プログラムを順次読み出して前記遊技処理を再開するものであって、前記電氣的制御手段は、前記初期化处理を完了してから前記遊技処理を開始するまでの間に、該遊技処理の開始時期を遅延させる遅延処理を実行するものであってもよい。すなわち、ここにいう復帰処理は、初期化处理と遅延処理を併せた処理になる。

## 【0842】

また、電源の遮断が検出されるまで繰り返し実行される主処理を含む遊技に関する遊技処理を実行する遊技制御手段と、前記遊技制御手段が前記遊技処理を実行中に該遊技処理の進行を妨げる異常を検出したことに基づいて復帰指示を行う異常検出手段とを備え、前記遊技制御手段は、前記復帰指示に基づいて前記遊技処理の実行を中止し、該遊技処理の実行を再開するための復帰処理を行うものであって、該復帰処理を実行している最中に該遊技処理の再開を遅延させる遅延処理を実行するものであることを特徴とする遊技台。の態様であってもよい。この態様によれば、前記電氣的制御手段といった、ハードウェア処理とソフトウェア処理の双方を行うことができる手段に代えて、遊技制御手段といったソフトウェア処理のみを行う手段によって、上記復帰処理を行い、その復帰処理を実行している最中に上記遅延処理を実行する。なお、前記遊技処理の遊技処理プログラムを記憶した第1記憶部と、前記復帰処理の復帰プログラムを記憶した第2記憶部とを備え、前記遊技制御手段は、前記1記憶部から前記遊技処理プログラムを順次読み出して前記遊技処理を実行するものであって、前記電氣的制御手段は、前記復帰プログラムにしたがって前記復帰処理を実行するものであってもよい。

## 【0843】

この場合、安定した遊技制御を行うことができる場合がある。すなわち、遊技制御の開始を遅延させることで、遊技者を介してリセットの発生を遊技店員が把握できる場合があ

10

20

30

40

50

る。また、リセットの発生を遊技店員が把握できるため、リセットの発生要因（例えば、静電気を発する電気部品）を特定して対処する（例えば、遊技台裏の配線の配置等を見直す）ことができ、遊技台に安定した制御を行なわせることができる場合がある。また、復帰処理中は遊技処理が行われなため、リセット発生時に遊技領域を転動していた遊技球が復帰処理中に第1特図始動口230に進入したとしても進入に基づく処理が行われず、遊技者に不安を抱かせて遊技店員を呼び易くすることが出来る場合がある。また、遊技店員は、遊技球を第1特図始動口230などに手で入れることによって、遊技台の状態を確認することができる場合がある。また、復帰処理中は遊技処理が行われなため、本体104または前面枠扉106を開放したとしても開放に基づく処理が行われず、遊技店員が本体104または前面枠扉106を開放することで、遊技台の状態を確認することができる場合がある。

10

#### 【0844】

また、これまでの説明において、『前記電氣的制御手段（例えば、マイクロプロセッサ3000）は、異常が発生したか否かを判定するセキュリティチェック処理を前記復帰処理中に行う（例えば、ステップSH05のセキュリティチェック処理を行う）ものであることを特徴とする遊技台。』についても説明がなされている。

#### 【0845】

この場合、復帰処理中にセキュリティチェック処理も行うため、復帰処理が行われている間の時間を有効に活用することができるとともに、セキュリティを向上させることが出来る場合がある。

20

#### 【0846】

なお、前記異常検出部は、前記異常のうちの第1の異常（例えば、CPU304の指定エリア外走行）があったことに基づいて前記復帰指示のうちの第1の復帰指示（例えば、システムリセット信号）を出力し、前記異常のうちの第2の異常（例えば、WDT3141のタイムアウト）があったことに基づいて前記復帰指示のうちの第2の復帰指示（例えば、ユーザリセット信号）を出力し、前記電氣的制御手段は、前記第1の復帰指示に基づいて前記復帰処理を実行している最中に前記遅延処理を実行し、前記第2の復帰指示に基づいて前記復帰処理を実行している最中に前記遅延処理を実行しないものであってもよい。

#### 【0847】

この場合、異常の種類に応じて遊技処理が開始されるまでの時間が異なるため、遊技店員が異常の種類を判別することができる場合がある。

30

#### 【0848】

また、前記第1の異常は、前記第2の異常よりも異常度が高いものであってもよい。

#### 【0849】

この場合、異常度の高い第1の異常が発生したときに復帰処理で遅延処理が行われるため、遊技店員は異常度が高い異常に基づくリセットが発生したことを確認することができる場合がある。

#### 【0850】

また、前記電氣的制御手段は、前記第1の復帰指示に基づいて前記復帰処理を実行している最中に前記セキュリティチェック処理を実行し、前記第2の復帰指示に基づいて前記復帰処理を実行している最中に前記セキュリティチェック処理を実行しないものであってもよい。

40

#### 【0851】

なお、遊技領域に向けて遊技球を発射する発射手段（例えば、発射装置110）を備え、前記発射手段は、前記遅延処理が実行されている間は遊技球の発射を停止するものであってもよい。

#### 【0852】

この場合、遊技者は、パチンコ機100が壊れたのではないかと不安に思い、遊技店の店員を呼ぶことが期待でき、遊技店員に瞬断やリセットが発生したことを知らせることが

50

できる場合がある。すなわち、遅延処理中は発射を停止させることで、遊技者を介してリセットの発生を遊技店員が把握できる場合がある。

#### 【0853】

また、前記遊技処理の遊技処理プログラムを記憶した記憶部（例えば、内蔵ROM306）を備え、前記遊技制御部（例えば、主制御部300）は、前記記憶部から前記遊技処理プログラムを読み出して前記遊技処理を実行するものであって、前記記憶部（例えば、内蔵ROM306）は、この記憶部に記憶された内容（例えば、ユーザプログラム）に基づいて得られる所定の認証情報（例えば、認証コード）を格納したものであり、前記電氣的制御手段（例えば、マイクロプロセッサ3000）は、前記記憶部に記憶されている内容に基づいて前記認証情報を前記復帰処理中に取得し、取得した認証情報が該記憶部に格納されている前記所定の認証情報に一致するか否かの判定を該復帰処理中に行う（例えば、ステップSH05のセキュリティチェック処理を行う）ものであってもよい。

10

#### 【0854】

この場合、復帰処理中に所定の認証情報に一致するか否かの判定を行うため、復帰処理が行われている間の時間を有効に活用することができるとともに、セキュリティを向上させることが出来る場合がある。

#### 【0855】

なお、前記電氣的制御手段は、取得した認証情報が該記憶部に格納されている前記所定の認証情報に一致したことに基づいて、前記遅延処理を実行するものであってもよい。

#### 【0856】

また、前記遊技制御部は、取得した認証情報が該記憶部に格納されている前記所定の認証情報に不一致であったことに基づいて、動作を停止するものであってもよい。

20

#### 【0857】

また、前記電氣的制御手段は、前記遊技処理の開始を、予め定まった時間遅延させる第一遅延処理（例えば、ステップSH07の固定延長処理）、および変更可能な時間遅延させる第二遅延処理（例えば、ステップSH09のランダム延長処理）のうち少なくともいずれか一方の処理を行うものであってもよい。

#### 【0858】

また、前記電氣的制御手段は、前記遊技処理の開始を予め定まった時間遅延させる第一遅延処理を行った後で、外部に対してリセット指示（例えば、リセット出力端子（XRS T O端子）からリセット信号）を出力し、その後、該遊技処理の開始を変更可能な時間遅延させる第二遅延処理を行うものであってもよい。ここにいう変更可能時間とは、前記電氣的制御手段の製造時や設定時に、複数の用意された時間の中から選択された時間であってよいし、任意に変えられる時間であってよい。

30

#### 【0859】

この場合、遊技処理の開始タイミングと外部に対してのリセット指示の出力タイミングとが異なるため、有利な乱数が取得可能な最初のタイミングを不正者が解析困難にすることができ、不正者の不正行為を防止することができる場合がある。

#### 【0860】

また、前記遊技制御部は、前記遊技処理の中で、所定の数値範囲（例えば、乱数生成範囲）で数値を繰り返し更新し更新した数値を用いて所定の抽選処理（例えば、特図の当否判定）を実行するものであって、前記電氣的制御手段は、前記復帰処理の中で、前記数値を初期化する（例えば、ステップSH03の第1内部回路初期化処理を実行する）ものであってもよい。

40

#### 【0861】

また、前記遊技制御部によって制御される遊技表示手段（例えば、普図表示装置210や、第1特図表示装置212や、第2特図表示装置214や、普図保留ランプ216や、第1特図保留ランプ218や、第2特図保留ランプ220や、高確中ランプ222）を備え、前記遊技表示手段は、前記遅延処理が実行されている間は非表示になるものであってもよい。

50



## 【0862】

この場合、遊技表示手段の表示が行われるはずが行われなため、遊技者に不安を抱かせて遊技店員を呼び易くすることが出来る場合がある。また、遊技店員は、遊技球を第1始動口230などに手で入れることによって、遊技台の状態を確認することができる場合がある。

## 【0863】

また、遊技球が進入可能な第1の状態と、該第1の状態より遊技球な進入困難な第2の状態とに変化可能な可変入賞手段（例えば、第2特図始動口232、可変入賞口234）を備え、前記可変入賞手段は、前記第1の状態の場合に前記復帰指示があったことに基づいて前記第2の状態に変化するものであってもよい。

10

## 【0864】

この場合、可変入賞手段が第1の状態であるべきなのに第2の状態となるため、遊技者に不安を抱かせて遊技店員を呼び易くすることが出来る場合がある。また、遊技店員は、遊技球を第1始動口230などに手で入れることによって、遊技台の状態を確認することができる場合がある。なお、前記可変入賞手段は、前記第1の状態の場合に前記復帰指示があったことに基づいて該第1の状態を維持するものであってもよい。この場合、予め定められた所定時間（例えば、開放時間）以上第1の状態となるため、遊技者に不安を抱かせて遊技店員を呼び易くすることが出来る場合がある。

## 【0865】

また、前記遊技制御部からの指示に基づいて演出を実行する演出実行手段（例えば、装飾図柄表示装置208）を備え、前記演出実行手段は、前記遅延処理が実行されている間も演出を継続するものであってもよい。ここで、前記演出実行手段は、前記遊技表示手段よりも大きな表示領域を有するものであってもよい。

20

## 【0866】

この場合、遅延処理中に遊技者を退屈させないようにすることができる場合がある。

また、前記電氣的制御手段は、前記復帰指示に基づいて、第1復帰処理と第2復帰処理を選択的に実行するものであり、前記第1復帰処理の復帰動作を記憶した第1復帰動作記憶部と、前記第2復帰処理の復帰動作を記憶した第2復帰動作記憶部と、を備え、前記電氣的制御手段は、前記第1復帰処理を選択した場合には前記第1復帰動作記憶部に記憶された復帰動作にしたがって前記第1復帰処理を実行し、前記第2復帰処理を選択した場合には前記第2復帰動作記憶部に記憶された復帰動作にしたがって前記第2復帰処理を実行するものであってもよい。

30

## 【0867】

また、前記遊技制御部は、前記遊技処理の中で、所定の数値範囲で数値を繰り返し更新し更新した数値を用いて所定の抽選処理を実行するものであって、前記電氣的制御手段は、前記復帰指示に基づいて、前記数値を初期化する第1復帰処理（例えば、ステップSH03の第1内部回路初期化处理）と、該数値を該復帰指示がなされる前の値に保持する第2復帰処理（例えば、ステップSH11の第2内部回路初期化处理）を選択的に実行するものであってもよい。

## 【0868】

また、本発明を所定数の遊技球を循環して使用する封入式遊技機に適用してもよい。また、主制御部300と副制御部400は双方向通信でもよいし、主制御部300と副制御部400の機能を備えた1つの制御部を代わりに設けてもよい。

40

## 【0869】

従来より、遊技台として、遊技盤の遊技領域に遊技球の落下の方向に変化を与える障害物や、遊技球が入賞可能な入賞口、始動口、可変入賞口などを設け、これらに遊技球が入賞すると賞球を払い出すなど遊技者に特典が与えられるようにした弾球遊技機（パチンコ機）が知られている。この弾球遊技機では、遊技球が始動口へ進入したことに基づいて当否判定の抽選処理を行い、当否判定の結果に応じて、可変入賞口を所定時間開放させる等、遊技者に有利な遊技状態を発生させるようにしている。

50

## 【0870】

また、メダル（遊技媒体）を投入してスタートレバーを操作することでリールを回転させるとともに、内部抽選によって役を内部決定し、ストップボタンを操作することでリールを停止させたときに、図柄表示窓上に内部決定に応じて予め定められた図柄の組み合わせが表示されると役が成立し、メダルの払い出しを伴う役が成立した場合には、規定数のメダルを払い出すなど遊技者に特典が与えられるような回胴遊技機（スロットマシン）も知られている。

## 【0871】

ところで、遊技店の営業中に、遊技台の電気系統に異常が発生し、遊技台の電源が遮断（電断）することがある。電源断した遊技台は、その後、遊技に支障がないように即座に自動的に復電する（電源が投入される）。このため、遊技台が電断したことに遊技店側がその場では気づかないことがある。また、遊技台のプログラムが想定外の動作をした場合等、何らかの異常が生じた場合には、一旦リセットがかかる。しかしながらリセットがかかるような異常が生じた場合でも、遊技に支障がないように即座に自動的に復帰する。

## 【0872】

遊技台の電気系統に異常が発生した原因や、リセットがかかるような異常が生じた原因としては、様々なことが考えられ、遊技店側としては電断が生じたことや、リセットがかかるような異常が生じたことを、その場で把握したい場合がある。

## 【0873】

一方、上記遊技台によれば、電気系統の異常が発生したことや、リセットがかかるような異常が生じたことを、遊技店側がその場で把握する機会を生じさせやすい。また、遊技制御の開始を遅延させることで、遊技者を介してリセットの発生を遊技店員が把握でき、遊技店員がリセットの発生要因を特定して対処する（例えば、遊技台裏の配線の配置等を見直す）ことができ、遊技台に安定した制御を行なわせることができる場合がある。

## 【0874】

以下、これまで説明したことも含めて付記する。

## 【0875】

（付記1）

遊技に関する複数種類の処理を含む遊技制御を行う遊技制御手段と、

前記遊技制御手段の処理の進行に関する異常を検出するとともに、該異常を検出した場合には前記遊技制御手段に復帰指示を行う異常検出手段と、

前記遊技制御手段および前記異常検出手段に所定の電圧を供給する電圧供給手段と、を備えた遊技台であって、

前記遊技制御手段は、

前記異常検出手段から前記復帰指示を受けたことに基づいて行われる復帰処理と、該復帰処理の後に開始され、所定の終了条件が成立するまで繰り返し実行される主処理と、を行い、

前記復帰処理は、前記遊技制御の開始を遅延させる遅延処理を含むことを特徴とする遊技台。

## 【0876】

（付記2）

乱数を生成するものであって、該乱数が生成される数値範囲である乱数生成範囲を設定可能な乱数生成手段（乱数生成回路318等）と、

前記乱数生成手段により生成された乱数を用いる抽選処理を含む、遊技に関する複数種類の処理を行う遊技制御手段（CPU304等）と、

前記遊技制御手段の処理の進行に関する異常があるか否かを検出する処理を実行するとともに、該異常を検出した場合には前記遊技制御手段の処理を特定の処理から再開させて前記遊技制御手段の処理を正常に復帰させるための復帰指示（システムリセット又はユーザリセット）を前記遊技制御手段に対して行う異常検出手段（リセット制御回路314等）と、を備え、

10

20

30

40

50

前記乱数生成手段は、

前記乱数生成範囲を設定するための乱数範囲設定指示を前記遊技制御手段から受けたことに基づいて前記乱数生成範囲を更新する生成範囲更新処理（図 8 9 のステップ S 3 0 5 3）を実行するものであり、

前記遊技制御手段は、

前記異常検出手段から前記復帰指示を受けた際に、該復帰指示が行われる前に設定された前記乱数生成範囲と同じ数値範囲を前記乱数生成範囲として設定するための前記乱数範囲設定指示を前記乱数生成手段に対して行う（図 8 9 のステップ S 3 0 5 3）ものであることを特徴とする遊技台（パチンコ機 1 0 0 やスロットマシン 1 1 0 0）。

【0 8 7 7】

10

（付記 3）

付記 2 記載の遊技台であって、

前記異常検出手段は、

所定の開始条件が成立したことに基づいて経過時間の計測を開始し、該経過時間が特定時間を超えたか否かを判定するとともに、該経過時間が特定時間を超えたと判定したことに基づいて前記遊技制御手段に復帰指示を行う（図 7 5 の W D T 3 1 4 1 の説明参照）ものであることを特徴とする遊技台。

【0 8 7 8】

（付記 4）

所定の数値範囲で所定の周期ごとに更新され、遊技に関する抽選に用いられる乱数を生成する乱数生成手段（乱数生成回路 3 1 8 等）と、

20

前記乱数の更新に関する異常を検出する更新異常検出手段（周波数監視回路 3 1 8 2 と乱数監視回路 3 1 8 4）と、

前記異常検出手段により前記異常が検出されたことに基づいて、所定条件が成立するまで前記異常検出手段により前記異常が検出されたことを示す異常検出情報を保持する異常検出情報保持手段（内部情報レジスタ 3 1 0 1）と、

所定のタイミングで前記異常検出情報保持手段を参照し、前記異常検出情報保持手段により前記異常検出情報が保持されていたことに基づいて、所定の異常対応処理を実行する異常対応処理手段（C P U 3 0 4 と図 9 5 のデバイス監視処理および入賞受付処理）と、

30

【0 8 7 9】

（付記 5）

付記 4 記載の遊技台であって、

前記異常検出情報保持手段は、

前記異常検出情報を保持している状態において前記異常対応処理手段により参照されたことに基づいて、前記異常検出情報が消去されるものであること（内部情報レジスタの説明参照）を特徴とする遊技台。

【0 8 8 0】

<主制御部の第 1 変形例>

図 1 0 9 （a）は、主制御部の構成の第 1 変形例を示した図である。この例では、C P U 3 0 4 は、リセット信号入力端子 X S R S T を備え、このリセット信号入力端子 X S R S T には、ランダム遅延回路 3 1 7 が接続されている。なお、端子名称の先頭に付した「X」の文字は、端子に入出力する信号がローアクティブであることを示しているが、ハイアクティブでもよいことは言うまでもない（以下同様）。

40

【0 8 8 1】

また、ランダム遅延回路 3 1 7 の入力端子には、電圧監視回路 3 3 8 が出力する低電圧信号と、W D T 3 1 4 が出力する W D T 起動信号の 2 つが入力されている。電圧監視回路 3 3 8 は、電源制御部 6 6 0 から主制御部 3 0 0 に供給している電源の電圧値が所定の値（本実施形態では 9 V）未満である場合に電圧が低下したことを示す低電圧信号（例えば、L レベルの信号）を出力する。また、W D T 3 1 4 は、W D T のタイムアウトによって

50

W D T 起動信号（例えば、L レベルの信号）を出力する。

#### 【0882】

ランダム遅延回路 3 1 7 は、電圧監視回路 3 3 8 が出力する低電圧信号、または、W D T 3 1 4 が出力する W D T 起動信号を入力端子から入力した後、ランダムに選択した遅延時間の経過を待って、当該入力信号を C P U 3 0 4 のリセット信号入力端子 X S R S T に向けて出力する。図 9 を用いて説明したように、C P U 3 0 4 は、リセット信号入力端子 X S R S T に所定期間（例えば、システムクロックの 4 周期分の期間）以上の L レベルの信号が入力された後に H レベルの信号が入力された場合にシステムリセット（初期化）される。このため、この例では、C P U 3 0 4 は、電圧監視回路 3 3 8 が出力する低電圧信号、または、W D T 3 1 4 が出力する W D T 起動信号のいずれかに遅延時間を付加した信号に基づいてシステムリセット（初期化）される。

10

#### 【0883】

##### <主制御部の第 2 変形例>

図 1 0 9 （b）は主制御部の構成の第 2 変形例を示した図である。この例では、W D T 3 1 4 が出力する W D T 起動信号を、ランダム遅延回路 3 1 7 を介さずに C P U 3 0 4 のリセット信号入力端子 X S R S T に向けて直接、出力するように構成している。このため、この例では、C P U 3 0 4 は、電圧監視回路 3 3 8 が出力する低電圧信号の出力タイミングを遅延時間だけ遅延させた信号、または、W D T 3 1 4 が出力する W D T 起動信号（遅延時間なし）に基づいてシステムリセット（初期化）される。

20

#### 【0884】

同図（c）は、上述のランダム遅延回路 3 1 7、電圧監視回路 3 3 8、および W D T 3 1 4 を有して構成されたリセット回路の構成例を示した図である。この例では、C P U 3 0 4 を、（1）W D T 3 1 4 が出力する W D T 起動信号（遅延時間なし）に基づいてシステムリセット、（2）W D T 3 1 4 が出力する W D T 起動信号にランダム遅延回路 3 1 7 で選択した遅延時間だけ出力タイミングを遅延させた信号に基づいてシステムリセット、（3）W D T 3 1 4 が出力する W D T 起動信号をリセット信号入力端子 X S R S T に入力させずに低電圧信号のみに基づいてシステムリセット、の 3 つの状態のいずれかを C P U 3 0 4 が選択可能に構成している。また、ランダム遅延回路 3 1 7 によって付加される遅延時間の設定が可能で、W D T 起動信号に付加する遅延時間と低電圧信号に付加する遅延時間を個別に設定可能に構成している。これらの選択や設定、W D T 3 1 4 のスタートやリセットは、例えば、C P U 3 0 4 のデータ入出力端子 D 0 ～D 7 を介してリセット回路に内蔵したレジスタに設定値を書き込んだり、リセット回路の所定の入力端子に所定のレベルの信号（設定値に対応する信号）を入力したりすることで行われる。

30

#### 【0885】

##### <リセット回路の第 1 変形例>

図 1 1 0 （a）は、図 1 0 9 （c）に示すリセット回路の第 1 変形例を示した図である。この例では、リセット回路に内蔵された第一 W D T 3 1 4 a とは別に、C P U 3 0 4 にも第二 W D T 3 1 4 b を内蔵している。そして、C P U 3 0 4 は、リセット信号入力端子 X S R S T に入力するリセット信号によってシステムリセット（外部リセット）を行うか、内蔵する第二 W D T 3 1 4 b のタイムアウトによってシステムリセット（内部リセット）を行うか、の選択が可能なレジスタを備えている。

40

#### 【0886】

##### <リセット回路の第 2 変形例>

同図（b）は、図 1 0 9 （c）に示すリセット回路の第 2 変形例を示した図である。この例では、C P U 3 0 4 が、リセット回路に内蔵していたランダム遅延回路 3 1 7 を内蔵しているとともに、リセット信号出力端子 X R S T O を備えている。そして、C P U 3 0 4 は、外部リセットに基づいてリセット信号出力端子 X R S T O からリセット信号を出力するか、内部リセットに基づいてリセット信号出力端子 X R S T O からリセット信号を出力するか、の選択が可能なレジスタを備えている。また、C P U 3 0 4 は、リセット信号出力端子 X R S T O から出力するリセット信号に、ランダム遅延回路 3 1 7 が選択する遅

50

延時間を付与するか否かの選択が可能なレジスタも備えており、また、外部リセットに基づくリセット信号の出力タイミングを遅延させるための遅延時間と、内部リセットに基づくリセット信号の出力タイミングを遅延させるための遅延時間を個別に設定可能に構成している。

#### 【0887】

＜リセット回路の第3変形例＞

図111(a)は、図109(c)に示すリセット回路の第3変形例を示した図である。この例では、電圧監視回路338から出力される低電圧信号の出力タイミングを遅延させる遅延時間を設定するための第1ランダム遅延回路317aと、WDTが出力するWDT起動信号の出力タイミングを遅延させる遅延時間を設定するための第2ランダム遅延回路317bと、を別に設けている。すなわち、この例では、CPU304を、(1)電圧監視回路338が出力する低電圧信号に、第1ランダム遅延回路317aで選択した遅延時間だけ出力タイミングを遅延させた信号に基づいてシステムリセット、(2)WDT314が出力するWDT起動信号に、第2ランダム遅延回路317bで選択した遅延時間だけ出力タイミングを遅延させた信号に基づいてシステムリセット、(3)WDT314が出力するWDT起動信号(遅延時間なし)に基づいてシステムリセット、の3つの状態のいずれかをCPU304が選択可能に構成している。また、低電圧信号やWDT起動信号の出力タイミングを遅延させるための各々の遅延時間を第1ランダム遅延回路317a、第2ランダム遅延回路317bによって個別に設定可能に構成している。これらの選択や設定、WDT314のスタートやリセットは、例えば、CPU304のデータ入出力端子D0～D7を介してリセット回路に内蔵したレジスタに設定値を書き込んだり、リセット回路の所定の入力端子に所定のレベルの信号(設定値に対応する信号)を入力したりすることで行われる。

#### 【0888】

＜リセット信号出力端子の接続の一例＞

図111(b)は、CPU304に接続される各種ICの一部を示した図である。この例では、CPU304は、リセットに関連する端子として、同図(a)に示す電圧監視回路338が出力する低電圧信号に基づいて生成されるリセット信号が入力されるリセット信号入力端子XSRST1と、同図(a)に示すWDT314が出力するWDT起動信号に基づいて生成されるリセット信号が入力されるリセット信号入力端子XSRST2と、リセット信号を出力するためのリセット信号出力端子XRSTOの3つの端子を備える。

#### 【0889】

図9を用いて説明したように、CPU304は、リセット信号入力端子XSRST1(またはリセット信号入力端子XSRST2)に所定期間(例えば、システムクロックの4周期分の期間)以上のLレベルの信号が入力された後に、当該入力端子にHレベルの信号が入力された場合にシステムリセット(初期化)される。

#### 【0890】

また、リセット信号出力端子XRSTOには、複数(この例では4つ)のIC11～IC14の入力端子と、他の回路の入力端子が接続されている。ここで、IC11～IC14や他の回路としては、例えば、アドレスデコーダやフリップフロップ等のロジックIC、図4に示される各種回路・IC・装置(例えば、乱数値生成回路、起動信号出力回路、駆動回路、音源IC、センサ回路、遮蔽装置、装飾図柄表示装置)を適用することができる。また、IC11～IC14や他の回路の入力端子としては、回路全体を初期化するためのリセット信号が入力されるリセット端子や、回路の一部を初期化するための信号が入力されるCLR端子などを適用することができる。

#### 【0891】

＜リセット信号出力端子の接続の第1変形例＞

図112は、図111(b)に示すリセット信号出力端子XRSTOの接続の第1変形例を示した図である。この例では、CPU304は、上述のリセット信号出力端子XRSTOをIC11の入力端子9に接続することに加えて、8本のデータ信号端子D0～D7

を、IC11のデータ信号入力端子1～入力端子8の各々に接続している。また、IC11は、データ信号入力端子1～入力端子8の各々に対応する出力端子1～8を備えており、これらの出力端子1～8を、図示しない第1副制御部400のCPU404の8本のデータ信号入力端子に接続している。すなわち、CPU304のデータ信号線が、IC11を介して第1副制御部400のCPU404に入力されており、CPU304と第1副制御部400のCPU404は、IC11を介して通信可能に構成されている。なお、CPU304とCPU404を、IC11に加えて（または替えて）、上述のIC12～IC14の一部または全てを介して通信可能に構成してもよい。また、CPU304とCPU404を、これらのIC11～IC14に加えて（または替えて）、中継基板を介して通信可能に構成してもよい。

10

#### 【0892】

<リセット信号出力端子の接続の第2変形例>

図113は、図4を用いて説明した基本回路302と、この基本回路302に接続される各種ICの一部を示した図である。なお、ここでは、主制御部300の基本回路302について説明するが、主制御部300の基本回路302に替えて（または、加えて）第1副制御部400の基本回路402、第2副制御部500の基本回路502、その他の制御部（例えば、払出制御部600）の基本回路に相当する回路に適用することもできる。

#### 【0893】

基本回路302は、リセットに関連する端子として、上述のリセット信号入力端子XSRSTとリセット信号出力端子XRSTOの2つの端子を備える。リセット信号入力端子XSRSTには、IC01（例えば、上述のリセット回路や電源監視回路）の出力端子が接続されており、図9を用いて説明したように、基本回路302のCPU304は、リセット信号入力端子XSRSTに所定期間（例えば、システムクロックの4周期分の期間）以上のLレベルの信号が入力された後にHレベルの信号が入力された場合にシステムリセット（初期化）される。

20

#### 【0894】

また、リセット信号出力端子XRSTOには、複数（この例では4つ）のIC11～IC14の入力端子と、他の回路の入力端子が接続されている。ここで、IC11～IC14や他の回路としては、例えば、アドレスデコーダやフリップフロップ等のロジックIC、図4に示される各種回路・IC・装置（例えば、乱数値生成回路、起動信号出力回路、駆動回路、音源IC、センサ回路、遮蔽装置、装飾図柄表示装置）を適用することができる。また、IC11～IC14や他の回路の入力端子としては、回路全体を初期化するためのリセット信号が入力されるリセット端子や、回路の一部を初期化するための信号が入力されるCLR端子などを適用することができる。

30

#### 【0895】

<固定延長時間とランダム延長時間>

次に、図114を用いて、システムリセット後に基本回路302によって実行される固定延長処理とランダム延長処理の時間設定について説明する。図9を用いて説明したように、CPU304は、セキュリティモードにおいてセキュリティチェック処理→固定延長処理→ランダム延長処理の順番で各々の処理を実行した後、ユーザモードに移行し、遊技制御用プログラムを実行するように構成されている。本実施形態では、固定延長処理の処理時間（固定延長時間）とランダム延長処理の処理時間（ランダム延長時間）の各々をCPU304に予め設定することによって、セキュリティモード（＝セキュリティチェック処理＋固定延長処理＋ランダム延長処理）の時間を、設定した時間分だけ延長することが可能である。

40

#### 【0896】

図114（a）は固定延長時間の設定例を示したものである。この例では、CPU304の所定記憶領域に、3ビット長の設定値データ000B～111Bのいずれかを予め設定することによって、同図に示す時間だけ固定延長時間を延長できるように構成されている。なお、この例では、設定値データが111Bの場合を除き、固定延長時間は、「2の

50

2 2 乗 × ( 1 / S C L K ) × 設定値データ」の計算式によって算出される。

#### 【0897】

例えば、設定値データを000Bに設定した場合には、システムクロックSCLKの周波数とは無関係に固定延長時間が0（延長しない）に設定され、設定値データを001Bに設定した場合、システムクロックSCLKの周波数が8MHzのときは固定延長時間が約525ms（=2の2乗×（1/8MHz）×1）に設定され、システムクロックSCLKの周波数が10MHzのときは固定延長時間が約420ms（=2の2乗×（1/10MHz）×1）に設定され、システムクロックSCLKの周波数が12MHzのときは固定延長時間が約350ms（=2の2乗×（1/12MHz）×1）に設定される。

10

#### 【0898】

なお、この例では、固定延長時間をシステムクロックSCLKに基づいて算出する例を示したが、例えば、他の設定値（例えば、後述するランダム延長時間）などに基づいて算出してもよいし、システムクロックSCLKと他の設定値の両方に基づいて算出してもよい。また、システムクロックSCLKと固定延長時間を反比例の関係にする例を示したが、両者が比例関係にあってもよい。また、設定値データが111Bの場合には、システムクロックSCLKの周波数とは無関係に固定延長時間を共通の約3000msに設定する例を示したが、設定値データが111Bの場合も上述の計算式によって算出してもよいし、最大値を設定してもよい。また、設定値データのデータ長は3ビットに限定されず、延長時間の種類を増やしてもよい。また、固定延長時間に0を含めずに0より大きい時間に限定してもよい（CPU304が必ず固定延長処理を実行するものとしてもよい）。また、内部リセットと外部リセットで、異なる固定延長時間の設定を可能に構成してもよい。

20

#### 【0899】

図114（b）はランダム延長時間の設定例を示したものである。この例では、CPU304の所定記憶領域に、2ビット長の設定値データ00B～11Bのいずれかを予め設定することによって、同図に示す時間だけランダム延長時間を延長することができるように構成されている。例えば、設定値データを00Bに設定した場合、システムクロックSCLKの周波数とは無関係にランダム延長時間が0（延長しない）に設定され、設定値データを01Bに設定した場合、システムクロックSCLKの周波数が8MHzのときはランダム延長時間が0～0.5msのランダムな時間（CPU304のコアによってランダムに選択された時間）に設定され、システムクロックSCLKの周波数が10MHzのときはランダム延長時間が0～0.4msのランダムな時間に設定され、システムクロックSCLKの周波数が12MHzのときはランダム延長時間が0～0.3msのランダムな時間に設定される。

30

#### 【0900】

なお、この例では、ランダム延長時間をシステムクロックSCLKによって変化させる例を示したが、異なるシステムクロックSCLK間で同一に設定してもよい。また、他の設定値（例えば、上述の固定延長時間）などに基づいて算出してもよいし、システムクロックSCLKと他の設定値の両方に基づいて算出してもよい。また、設定値データのデータ長は2ビットに限定されず、延長時間の種類を増やしてもよい。また、ランダム延長時間に0を含めずに0より大きい時間に限定してもよい（CPU304が必ずランダム延長処理を実行するものとしてもよい）。また、内部リセットと外部リセットで、異なるランダム延長時間の設定を可能に構成してもよい。また、固定延長時間とランダム延長時間の一方について基準クロックを可変とし、他方については基準クロックを固定に構成してもよい。

40

#### 【0901】

<WDTタイムアウト時間>

次に、図115を用いて、WDT314のタイムアウト時間設定について説明する。なお、同図はWDT314のタイムアウト時間の設定例を示したものである。この例では、

50

CPU 304 の所定記憶領域に、4ビット長の設定値データ 0000B～1000B のいずれかを予め設定することによって、同図に示す時間だけタイムアウト時間を設定できるように構成されている。例えば、設定値データを 0000B に設定した場合には、システムクロック SCLK の周波数とは無関係にタイムアウト時間が禁止 (WDT を使用しない) に設定され、設定値データを 0001B に設定した場合、システムクロック SCLK の周波数が 8MHz のときはタイムアウト時間が約 65ms に設定され、システムクロック SCLK の周波数が 10MHz のときはタイムアウト時間が約 52ms に設定され、システムクロック SCLK の周波数が 12MHz のときはタイムアウト時間が約 44ms に設定される。

#### 【0902】

なお、この例では、タイムアウト時間をシステムクロック SCLK によって変化させる例を示したが、異なるシステムクロック SCLK 間で同一に設定してもよい。また、他の設定値 (例えば、上述の固定延長時間やランダム延長時間) などに基づいて算出してもよいし、システムクロック SCLK と他の設定値の両方に基づいて算出してもよい。また、設定値データのデータ長は 4ビットに限定されず、タイムアウト時間の種類を増やしてもよい。また、タイムアウト時間が上述の固定延長時間よりも小さくなるように設定しているが、タイムアウト時間が上述の固定延長時間よりも大きくなるように設定してもよい。

#### 【0903】

##### <リセット出力信号とセキュリティモード>

次に、上述のリセット出力端子 XSRSTO の出力信号の変化と、セキュリティモードの状態変化との関係について説明する。図 116 (a) に示す例は、上記図 9 と同一の状態を示しており、リセット入力端子 XSRST に所定期間以上の L レベルの信号が入力された後に H レベルの信号が入力されたことによってセキュリティモードに移行している。そして、このセキュリティモードにおいて、最初にセキュリティチェック処理が実行され、次に、上述の固定延長時間の設定に応じた期間の経過を待つ固定延長処理が実行され、当該固定延長処理が終了したときにリセット出力端子 XSRSTO から H レベルの信号を出力し、その後、上述のランダム延長時間の設定に応じたランダムな期間の経過を待つランダム延長処理が終了した場合にユーザモードに移行し、遊技制御用プログラムの実行が開始されている。すなわち、この例では、固定延長処理の終了タイミング (=ランダム延長処理の開始タイミング) と、リセット出力端子 XSRSTO から H レベルの信号を出力するタイミングを一致させている。

#### 【0904】

また、図 116 (b) に示す例は、固定延長処理が終了した後、所定時間が経過したときにリセット出力端子 XSRSTO から H レベルの信号を出力し、その後、ランダム延長処理が終了した場合にユーザモードに移行し、遊技制御用プログラムの実行が開始されている。すなわち、この例では、固定延長処理が終了した後、ランダム延長処理が終了してユーザモードに移行する前 (遊技制御用プログラムの実行が開始される前) にリセット出力端子 XSRSTO から H レベルの信号を出力している。なお、所定時間 (固定延長処理が終了してから、リセット出力端子 XSRSTO から H レベルの信号を出力するまでの時間) は、固定時間でもよいし可変時間でもよい。また、固定時間を採用した場合には、例えば、固定延長処理が終了してから、システムクロック SCLK が所定クロック (例えば、4クロック) 経過した後のシステムクロック SCLK 立上がりエッジ (または、立下りエッジ) までの時間を、所定時間に設定してもよい。また、可変時間を採用した場合には、システムリセットの度に所定時間をランダムな時間に変化させてもよい。

#### 【0905】

##### <ランダム延長時間を変化させる場合>

図 117 (a)、(b) は、固定延長処理の終了タイミング (=ランダム延長処理の開始タイミング) と、リセット出力端子 XSRSTO から H レベルの信号を出力するタイミングが一致するが、ランダム延長時間が異なる例を示した図である。すなわち、同図 (a) と (b) では、固定延長時間の設定値が同一 (例えば、設定値 = 001B、SCLK = 約

10

20

30

40

50



8 MHz) であるとともに、ランダム延長時間の設定値も同一 (例えば、設定値 = 0 1 B、SCLK = 8 MHz) であるが、同図 (a) では、ランダム延長時間の数値範囲 (例えば、0 ~ 0. 5 ms) から時間 A 1 (例えば、0. 2 ms) が選択されている一方で、同図 (b) では、ランダム延長時間の数値範囲 (例えば、0 ~ 0. 5 ms) から、時間 A 1 よりも長い時間 A 2 (例えば、0. 3 ms) が選択されている。なお、この例では、ランダム延長時間の数値範囲として 0 ~ 0. 5 ms を選択しているため、ランダム延長時間が 0 (ランダム延長時間無し) になる場合も含まれている。

#### 【0906】

このように、固定延長時間やランダム延長時間の設定が同じであってもランダム延長時間 (A 1、A 2) を変化させることによって、リセット出力端子 X R S T O の H レベル信号出力から、ユーザモードが開始されるまでの時間 (A 1、A 2) を、システムリセットの度に変化させることができる。このため、リセット信号出力端子 X R S T O に接続された I C 1 1 ~ I C 1 4 や他の回路に H レベルの信号を出力してから (I C 1 1 ~ I C 1 4 や他の回路を起動したり初期化してから)、遊技制御用プログラムが実行されるまでの時間を変化させることができ、C P U 3 0 4 の各端子からの出力信号や C P U 3 0 4 の周辺 I C の動作などを手掛かりにして遊技制御用プログラムの実行タイミング (例えば、当否判定の抽選タイミング) を把握するような不正行為を未然に防止できる場合がある。なお、この例では、ランダム延長時間の設定値を同一にしたが、両者のランダム延長時間の設定値を異ならせてもよい。

#### 【0907】

< 固定延長時間を変化させる場合 >

図 1 1 7 (c)、(d) は、固定延長処理の終了タイミング (= ランダム延長処理の開始タイミング) と、リセット出力端子 X R S T O から H レベルの信号を出力するタイミングが一致するが、固定延長時間が異なる例を示した図である。すなわち、同図 (c) と (d) では、ランダム延長時間の設定値データは同一 (例えば、設定値 = 0 1 B、SCLK = 8 MHz) であり、同図 (a) および (b) でランダム延長時間の数値範囲 (例えば、0 ~ 0. 5 ms) から同一の時間 A 1 (例えば、0. 2 ms) が設定される一方で、同図 (c) では、固定延長時間を時間 B 1 (例えば、設定値 = 0 0 1 B、SCLK = 8 MHz、約 5 2 5 ms)、同図 (d) では、固定延長時間を時間 B 1 よりも長い時間 B 2 (例えば、設定値 = 0 1 0 B、SCLK = 8 MHz、約 1 0 5 0 ms) に設定している (B 1 < B 2)。

#### 【0908】

このように、ランダム延長時間 (A 1) が同じであっても固定延長時間 (B 1、B 2) を変化させることによって、固定延長処理が開始されてからユーザモードが開始されるまでの時間 (C 1、C 2) を変化させることができる。このため、C P U 3 0 4 のシステムリセットから遊技制御用プログラムが実行されるまでの時間をシステムリセットの度に変化させることができ、C P U 3 0 4 のリセット入力端子 X S R S T に入力するリセット信号などを手掛かりにして遊技制御用プログラムの実行タイミング (例えば、当否判定の抽選タイミング) を把握するような不正行為を未然に防止できる場合がある。なお、図 1 0 9 では、固定延長時間とランダム延長時間の一方を異ならせる例を示したが、両者が異なるようにすれば、遊技制御用プログラムの実行タイミングを把握するような不正行為を、より確実に防止できる場合がある。

#### 【0909】

< 固定延長処理を契機にリセット出力信号を変化させる場合 >

図 1 1 8 は、固定延長処理を契機にリセット出力信号を変化させる例を示した図である。この例では、固定延長処理の開始タイミング (= セキュリティチェック処理の終了タイミング) と、リセット出力端子 X R S T O から H レベルの信号を出力するタイミングを一致させている。なお、固定延長処理を契機にリセット出力信号を変化させればよいので、例えば、固定延長処理の開始前 (例えば、固定延長処理の開始タイミングから 2 0 0 ms 前) にリセット出力信号を出力してもよいし、固定延長時間の設定値に応じて遅延時間を

決定し、当該遅延時間の経過を待ってからリセット出力信号を出力してもよい。また、内部リセット（内蔵するWDTのタイムアウトに基づくリセット）か、外部リセット（XSRST端子への信号入力に基づくリセット）かを判別し、当該リセットの種類に応じてリセット出力信号の出力タイミングを決定してもよい。

#### 【0910】

また、過去のリセットの種類を記憶する記憶手段を備え、当該記憶手段に記憶されたりリセットの種類履歴に基づいてリセット出力信号の出力タイミングを変化させてもよく、例えば、内部リセットが複数回連続した場合には、リセット出力信号の出力タイミングを前回よりも遅らせたり早めたりしてもよい。

#### 【0911】

＜他の出力信号とセキュリティモード＞

図119は、CPU304から出力されるアドレス信号、データ信号、制御信号、およびリセット出力信号の変化と、セキュリティモードの状態変化を示した図である。この例では、CPU304は、固定延長処理が終了したとき（ランダム延長処理を開始したとき）にリセット出力端子XRSSTOからHレベルの信号を出力するとともに、同じタイミングで外部のIC（例えば、ROM306、RAM308）との間でデータのリード（またはライト）を開始している。このため、この例では、リセット出力端子XRSSTOからHレベルの信号を出力すると同時に、アドレス信号A0～A15端子から16ビットのアドレス情報（この例では、0000H）の出力を開始し、制御信号XM1端子からアドレス情報のラッチタイミング（外部のICがアドレス情報を取り込むタイミング）を示す制御信号を出力し、続けて、データ信号D0～D7端子から8ビットのデータ情報（この例では、オペコード）の出力を開始し、制御信号XM1端子からデータ情報のラッチタイミング（外部のICがデータ情報を取り込むタイミング）を示す制御信号を出力している。

#### 【0912】

なお、この例では、リセット出力端子XRSSTOからHレベルの信号を出力すると同時にアドレス信号と制御信号の出力を開始しているが、例えば、リセット出力よりも前にアドレス信号と制御信号の出力を開始してもよいし、リセット出力よりも後（例えば、セキュリティモードからユーザモードに移行した直後）にアドレス信号と制御信号の出力を開始してもよい。なお、XM1端子から出力する制御信号は、例えば、CPU304のマシンサイクルを示す出力信号でもよいし、メモリ空間やI/O空間へのリクエスト出力信号でもよいし、リードサイクルやライトサイクルを示す出力信号でもよい。

#### 【0913】

図120（a）は外部リセット後の他の出力信号の状態変化を示した図であり、同図（b）は内部リセット後の他の出力信号の状態変化を示した図である。この例では、外部リセットの場合には、ランダム延長処理よりも所定時間A1だけ前に、アドレス信号A0～A15端子から16ビットのアドレス情報（この例では、0000H）の出力を開始し、制御信号XM1端子からアドレス情報のラッチタイミングを示す制御信号を出力している。一方、内部リセットの場合には、ランダム延長処理よりも所定時間A2（A2>A1）だけ前に、アドレス信号A0～A15端子から16ビットのアドレス情報（この例では、0000H）の出力を開始し、制御信号XM1端子からアドレス情報のラッチタイミングを示す制御信号を出力している。すなわち、この例では、外部リセットと内部リセットで、アドレス信号A0～A15端子の信号出力と、制御信号XM1端子の信号出力のタイミングを異ならせている。

#### 【0914】

また、内部リセットの場合には、WDT314のタイムアウト直後に、アドレス信号A0～A15端子から16ビットのアドレス情報（この例では、E000H）の出力を行うように構成し、外部リセット直後のアドレス情報（例えば、0000H）と異ならせている。なお、内部リセット直後に出力するアドレス情報と、外部リセット直後に出力するアドレス情報を同一に設定してもよい。

#### 【0915】

10

20

30

40

50

図 1 2 1 (a) は外部リセット後の他の出力信号の状態変化の他の例を示した図であり、同図 (b) は内部リセット後の他の出力信号の状態変化の他の例を示した図である。この例では、内部リセットの場合には、外部リセットのリセット解除時間 B 1 よりも長い時間 B 2 の経過を待って、内部リセットを解除するように設定している。なお、本構成と、上記図 1 1 5 を用いて説明した各構成を組み合わせることで実現してもよい。

#### 【0916】

＜セキュリティモードの開始タイミング＞

図 1 2 2 (a) は外部システムリセット後のセキュリティモードの開始タイミングの一例を示した図であり、同図 (b) は内部システムリセット後のセキュリティモードの開始タイミングの一例を示した図である。この例では、外部システムリセットの場合には、当該外部システムリセットの要因が発生してから（例えば、リセット入力信号 X S R S T の立下りエッジを検出してから）、時間 C 1（例えば、システムクロック S C L K の 4 クロック分の時間）が経過した後にセキュリティモードが開始されるように構成している。一方、内部システムリセットの場合には、内部リセットの要因が発生してから（例えば、W D T タイマがタイムアウトしてから）、時間 C 2（例えば、システムクロック S C L K の 1 クロック分の時間）が経過した後にセキュリティモードが開始されるように構成している。

#### 【0917】

なお、この例では、外部リセットの要因が発生してからセキュリティモードが開始されるまでの時間 C 1 を、内部リセットの要因が発生してからセキュリティモードが開始されるまでの時間 C 2 よりも長くしているが ( $C 1 > C 2$ )、外部リセットの要因が発生してからセキュリティモードが開始されるまでの時間 C 1 を、内部リセットの要因が発生してからセキュリティモードが開始されるまでの時間 C 2 よりも短くしてもよい ( $C 1 < C 2$ )。

#### 【0918】

また、上述の例では、セキュリティモードに含まれるランダム延長処理の実行時間（ランダム延長時間）や固定延長処理（固定延長時間）をリセットの要因によらずに自由に設定可能としたが、例えば、外部リセットと内部リセットで、ランダム延長時間の設定用レジスタと固定延長時間の設定用レジスタをそれぞれ種類だけ備えるように構成し、外部リセットと内部リセットで、ランダム延長時間や固定延長時間を個別に設定することを禁止してもよい（共通の設定だけを許可してもよい）。

#### 【0919】

図 1 2 3 は、本発明の基本概念を示した図である。本発明では、第 1 の事象に基づいて第 2 の事象を開始するときに、第 1 の事象が第 1 の要因に起因する場合と、第 1 の事象が第 2 の要因に起因する場合と、で第 2 の事象を開始するタイミングを異ならせることができる。

#### 【0920】

例えば、上記図 1 2 2 を用いて説明したセキュリティモードの開始タイミングについて考えると、第 1 の事象（リセット）に基づいて第 2 の事象（セキュリティモード）を開始する場合に、第 1 の事象が第 1 の要因（外部システムリセット）に起因する場合と、第 1 の事象が第 2 の要因（内部システムリセット）に起因する場合と、で第 2 の事象（セキュリティモード）を開始するタイミングを異ならせることができる。このため、第 1 の要因（外部システムリセット）が発生してから第 2 の事象（セキュリティモード）が開始されるまでの時間 C 1 を、第 2 の要因（内部システムリセット）が発生してから第 2 の事象（セキュリティモード）が開始されるまでの時間 C 2 よりも長くしたり ( $C 1 > C 2$ )、第 1 の要因（外部システムリセット）が発生してから第 2 の事象（セキュリティモード）が開始されるまでの時間 C 1 を、第 2 の要因（内部システムリセット）が発生してから第 2 の事象（セキュリティモード）が開始されるまでの時間 C 2 よりも短くすることができる。

#### 【0921】

また、セキュリティモードを開始するタイミングを異ならせることができるため、その後実行されるユーザモードを開始するタイミングも異ならせることも可能となる。例えば、第1の事象が第1の要因に起因する場合と、第1の事象が第2の要因に起因する場合と、でセキュリティモードの実行時間が同じであっても、セキュリティモードを開始するタイミングを異ならせれば、第1の事象が第1の要因に起因する場合と、第1の事象が第2の要因に起因する場合と、でユーザモードを開始するタイミングを異ならせることができる。

#### 【0922】

また、第1の事象が第1の要因に起因する場合と、第1の事象が第2の要因に起因する場合と、でセキュリティモードを開始するタイミングを異ならせなくても、セキュリティモードの実行時間を異ならせれば、第1の事象が第1の要因に起因する場合と、第1の事象が第2の要因に起因する場合と、でユーザモードを開始するタイミングを異ならせることができる。

#### 【0923】

＜他の実施形態＞

次に、他の実施形態に係るパチンコ機について説明する。本実施形態に記載した複数の構成のうち、1つの構成に記載している内容を、他の構成に適用することでより遊技の幅を広げられる場合がある。

#### 【0924】

例えば、図124(a)は、図2を用いて説明した電源基板182、払出基板170、および主基板156の接続例を示した図である。電源基板182は、RAMクリアスイッチ180と、このRAMクリアスイッチ180の操作が検出された場合に、検出信号を出力するための検出信号出力端子と、を備える。電源基板182のCPUは、RAMクリアスイッチ180の操作の有無を定期的に検出し、検出結果（例えば、操作ありの場合にはHレベルの信号、操作なしの場合にはLレベルの信号）を検出信号出力端子から払出基板170に向けて出力する。

#### 【0925】

払出基板170は、電源基板182の検出信号出力端子に接続される検出信号入力端子と、この検出信号入力端子を介して取得したRAMクリアスイッチ180の操作情報を一時的に記憶するためのRAMクリアスイッチ操作情報一時憶手段171（例えば、RAM、CPUのレジスタ、フリップフロップ）と、このRAMクリアスイッチ操作情報一時憶手段171に記憶された操作情報を出力するための操作情報出力端子と、を備える。払出基板170のCPUは、電源投入後やシステムリセット後の初期処理などにおいて、検出信号入力端子に入力された信号を検出し、RAMクリアスイッチ180が操作されているか否かを示す操作情報（例えば、Hレベルの信号を検出した場合には操作ありを示す1の情報、Lレベルの信号を検出した場合には操作無しの情報を示す0の情報）をRAMクリアスイッチ操作情報一時憶手段171の所定記憶領域に記憶するとともに、当該操作情報に対応する情報（例えば、操作ありの操作情報の場合にはHレベルの信号、操作無しの操作情報の場合にはLレベルの信号）を操作情報出力端子から主基板156に向けて出力する。

#### 【0926】

主基板156は、払出基板170の操作情報出力端子に接続される操作情報入力端子（上述のI/O410）と、RAMクリアタイミング報知用LED155と、を備える。主基板156に搭載されたCPU304は、初期設定において、RAMクリアタイミング報知用LED155を点灯させた後に（RAMクリアが可能であることを報知した後に）、操作情報入力端子（I/O310）の入力信号を確認し、Hレベルの信号を検出した場合には、RAM308の所定領域の初期化（RAMクリア）を行う。

#### 【0927】

図125は、変形例に係る主制御部メイン処理の流れを示すフローチャートであり、上記図10に対応するフローチャートである。なお、図10に示すフローチャートと同一の

10

20

30

40

50

部分については同一の符号を付し、その説明は省略する。また、図126は、主制御部メイン処理における初期設定2の流れを示すフローチャートである。

#### 【0928】

CPU304は、初期設定2のステップS128において、RAMクリアタイミング報知用LED155を点灯させた後に（RAMクリアが可能であることを報知した後に）、操作情報入力端子（I/O310）の入力信号を確認し、RAMクリアスイッチ180が操作されているか否かを示す操作情報（例えば、Hレベルの信号を検出した場合には操作ありを示す1の情報、Lレベルの信号を検出した場合には操作無しの情報を示す0の情報）をRAM308の所定記憶領域に記憶する。

#### 【0929】

また、初期設定2の実行後のステップS109の判定処理では、最初に、RAM308に操作ありを示す1の情報が記憶されているか否かを判定し、該当する場合（RAMクリアが必要な場合）には、基本回路302を初期状態にすべくステップS113に進む。一方、操作なしを示す0の情報が記憶されている場合（RAMクリアが必要でない場合）には、RAM308に設けた電源ステータス記憶領域に記憶した電源ステータスの情報を読み出し、この電源ステータスの情報がサスペンドを示す情報であるか否かを判定する。そして、電源ステータスの情報がサスペンドを示す情報でない場合には、基本回路302を初期状態にすべくステップS113に進み、電源ステータスの情報がサスペンドを示す情報である場合には、RAM308の所定の領域（例えば全ての領域）に記憶している1バイトデータを初期値が0である1バイト構成のレジスタに全て加算することによりチェックサムを算出し、算出したチェックサムの結果が特定の値（例えば0）であるか否か（チェックサムの結果が正常であるか否か）を判定する。そして、チェックサムの結果が特定の値（例えば0）の場合（チェックサムの結果が正常である場合）には電断前の状態に復帰すべくステップS111に進み、チェックサムの結果が特定の値（例えば0）以外である場合（チェックサムの結果が異常である場合）には、パチンコ機を初期状態にすべくステップS113に進む。同様に電源ステータスの情報が「サスペンド」以外の情報を示している場合にもステップS113に進む。

#### 【0930】

図124に戻って、同図（b）は、図124（a）に対応する従来の接続態様を示した図である。従来の接続態様では、主基板156において上述の固定延長時間として設定データ111B（約30秒）を設定した場合、セキュリティモードが終了してユーザーモードに移行し、遊技制御用プログラムが開始されるのは、最低でも約30秒後となる。このため、主基板156の遊技制御用プログラムにおいて操作情報入力端子（I/O410）に入力される信号を検出する構成を採用した場合、RAMクリアスイッチ180が最低でも約30秒以上操作され続けなければ、RAMクリアスイッチ180の操作の有無を検出することができない上に、外部からは、主基板156がRAMクリアスイッチ180の操作を検出するタイミングさえも知ることができない。

#### 【0931】

これに対して、同図（a）に示す本実施形態に係る接続態様では、遊技店の店員などは、RAMクリアタイミング報知用LED155の点灯／消灯を確認することでRAMクリアが可能か否かを即座に判断することができる。このため、RAMクリアタイミング報知用LED155の点灯を待ってRAMクリアスイッチ180を操作することで、確実にRAMクリアを行うことができる上に、RAMクリアスイッチ180を長時間に亘って操作し続けるような煩わしさからも解放され、利便性を高めることができる場合がある。また、RAMクリアタイミング報知用LED155の点灯前（CPU304の遊技制御用プログラムが開始される前に）にRAMクリアスイッチ180が操作されてしまった場合でも、払出基板170のCPUが起動していれば操作情報を払出基板170が記憶することが可能である。このため、主基板156は、払出基板170のRAMクリアスイッチ操作情報一時記憶手段171に記憶された操作情報に基づいて、の遊技制御用プログラムの開始後にRAMクリアを確実に実行することができる場合がある。

## 【0932】

なお、本実施形態では、電源基板182、払出基板170、および主基板156の間の信号の入出力を入出力端子を介して行う例を示したが、例えば、通信線を介した制御コマンドの送受信によるものでもよい。また、RAMクリアスイッチ180は、電源基板182に配置される例に限定されず、例えば、払出基板170に配置してもよい。

## 【0933】

＜主制御部の他の命令＞

次に、主制御部300が備える他の命令について説明する。なお、以降の説明において、「r」、「s」、「ss」、または「qq」は任意のレジスタを表す略号であり、「k」はTレジスタとペアになる下位アドレスを表す略号であり、「n」は8ビット長の数値（16進数）を表す略号であり、「mn」は16ビット長の数値（16進数）を表す略号であり、「（ ）」はカッコ内の値が示すアドレスのデータを表す略号である。

## 【0934】

＜他の命令／8ビットロード命令＞

8ビットロード命令は、8ビット長のデータを所定記憶領域にロードする命令であり、例えば、LD命令、LDT命令、またはCLRT命令などが該当する。

## 【0935】

例えば、(LD A, T)命令は、Tレジスタに記憶されたデータを、Aレジスタにロードする命令であり、(LD T, n)命令は、8ビット長の数値nを、Tレジスタにロードする命令である。

## 【0936】

(LDT A, (r))命令は、Tレジスタに記憶されたデータを上位バイト、rで示されるレジスタ(A, B, C, D, E, H, L)に記憶されたデータを下位バイトとするアドレスに記憶されたデータを、Aレジスタにロードする命令であり、(LDT A, (k))命令は、Tレジスタに記憶されたデータを上位バイト、kで示されるレジスタに記憶されたデータを下位バイトとするアドレスに記憶されたデータを、Aレジスタにロードする命令であり、(LDT r, (k))命令は、Tレジスタに記憶されたデータを上位バイト、kで示されるレジスタに記憶されたデータを下位バイトとするアドレスに記憶されたデータを、rで示されるレジスタ(B, C, D, E, H, L)にロードする命令である。

## 【0937】

(LDT (r), A)命令は、Aレジスタに記憶されたデータを、Tレジスタに記憶されたデータを上位バイト、rで示されるレジスタ(A, B, C, D, E, H, L)に記憶されたデータを下位バイトとするアドレスにロードする命令であり、(LDT (k), A)命令は、Aレジスタに記憶されたデータを、Tレジスタに記憶されたデータを上位バイト、kで示されるレジスタに記憶されたデータを下位バイトとするアドレスにロードする命令であり、(LDT (k), r)命令は、rで示されるレジスタ(B, C, D, E, H, L)に記憶されたデータを、Tレジスタに記憶されたデータを上位バイト、kで示されるレジスタに記憶されたデータを下位バイトとするアドレスにロードする命令である。

## 【0938】

(LDT (k), (ss))命令は、ssで示されるペアレジスタ(BC, DE, HL)に記憶されたデータが示すアドレスに記憶されたデータを、Tレジスタに記憶されたデータを上位バイト、kで示されるレジスタに記憶されたデータを下位バイトとするアドレスにロードする命令であり、(LDT (k), (HL+v))命令は、HLレジスタに記憶されたデータにvで示されるデータ(v=1~7)を加算した加算結果が示すアドレスに記憶されたデータを、Tレジスタに記憶されたデータを上位バイト、kで示されるレジスタに記憶されたデータを下位バイトとするアドレスにロードする命令であり、(LDT (k), (HL+d))命令は、HLレジスタに記憶されたデータにdで示されるデータ(d=8~127、-1~-128)を加算した加算結果が示すアドレスに記憶さ

10

20

30

40

50

れたデータを、Tレジスタに記憶されたデータを上位バイト、kで示されるレジスタに記憶されたデータを下位バイトとするアドレスにロードする命令であり、(LDT (ss), (k))命令は、Tレジスタに記憶されたデータを上位バイト、kで示されるレジスタに記憶されたデータを下位バイトとするアドレスに記憶されたデータを、ssで示されるペアレジスタ(BC, DE, HL)に記憶されたデータが示すアドレスにロードする命令である。

#### 【0939】

(LDT (HL+v), (k))命令は、Tレジスタに記憶されたデータを上位バイト、kで示されるレジスタに記憶されたデータを下位バイトとするアドレスに記憶されたデータを、HLレジスタに記憶されたデータにvで示されるデータ(v=1~7)を加算した加算結果が示すアドレスにロードする命令であり、(LDT (HL+d), (k))命令は、Tレジスタに記憶されたデータを上位バイト、kで示されるレジスタに記憶されたデータを下位バイトとするアドレスに記憶されたデータを、HLレジスタに記憶されたデータにdで示されるデータ(d=8~127, -1~-128)を加算した加算結果が示すアドレスにロードする命令であり、(LDT (k), n)命令は、nで示される値を、Tレジスタに記憶されたデータを上位バイト、kで示されるレジスタに記憶されたデータを下位バイトとするアドレスにロードする命令である。

#### 【0940】

(CLRT (k))命令は、数値の0を、Tレジスタに記憶されたデータを上位バイト、kで示されるレジスタに記憶されたデータを下位バイトとするアドレスにロードする命令である。

#### 【0941】

<他の命令／16ビットロード命令>

16ビットロード命令は、16ビット長のデータを所定記憶領域にロードする命令であり、例えば、LDT命令、LDWT命令、CLR命令、CLRWT命令などが該当する。具体的には、(LDT BC, n)命令は、Tレジスタに記憶されたデータをBレジスタにロードし、かつ、nで示される数値をCレジスタにロードする命令であり、(LDT DE, n)命令は、Tレジスタに記憶されたデータをDレジスタにロードし、かつ、nで示される数値をEレジスタにロードする命令であり、(LDT HL, n)命令は、Tレジスタに記憶されたデータをHレジスタにロードし、かつ、nで示される数値をLレジスタにロードする命令である。

#### 【0942】

(LDT IX, n)命令は、Tレジスタに記憶されたデータを16ビット長のIXレジスタの上位8ビットにロードし、かつ、nで示される数値を16ビット長のIXレジスタの下位8ビットにロードする命令であり、(LDT IY, n)命令は、Tレジスタに記憶されたデータを16ビット長のIYレジスタの上位バイトにロードし、かつ、nで示される数値を16ビット長のIYレジスタの下位バイトにロードする命令である。

#### 【0943】

(LDT ss, (k))命令は、Tレジスタに記憶されたデータを上位バイト、kで示されるレジスタに記憶されたデータを下位バイトとするアドレスに記憶されたデータを、ssで示されるペアレジスタ(BC, DE)の上位バイトにロードし、かつ、Tレジスタに記憶されたデータを上位バイト、kで示されるレジスタに記憶されたデータを下位バイトとするアドレスに1を加算したアドレスに記憶されたデータを、ssで示されるペアレジスタ(BC, DE)の下位バイトにロードする命令である。

#### 【0944】

(LDT qq, (k))命令は、Tレジスタに記憶されたデータを上位バイト、kで示されるレジスタに記憶されたデータを下位バイトとするアドレスに記憶されたデータを、qqで示されるペアレジスタ(IX, IY)の上位バイトにロードし、かつ、Tレジスタに記憶されたデータを上位バイト、kで示されるレジスタに記憶されたデータを下位バイトとするアドレスに1を加算したアドレスに記憶されたデータを、qqで示されるペア

レジスタ (IX, IY) の下位バイトにロードする命令である。

【0945】

(LDT HL, (k)) 命令は、Tレジスタに記憶されたデータを上位バイト、kで示されるレジスタに記憶されたデータを下位バイトとするアドレスに記憶されたデータを、HLレジスタのLレジスタにロードし、かつTレジスタに記憶されたデータを上位バイト、kで示されるレジスタに記憶されたデータを下位バイトとするアドレスに1を加算したアドレスに記憶されたデータを、HLレジスタのHレジスタにロードする命令である。

【0946】

(LDT (k), ss) 命令は、ssで示されるペアレジスタ (BC, DE, HL) の下位バイトに記憶されたデータを、Tレジスタに記憶されたデータを上位バイト、kで示されるレジスタに記憶されたデータを下位バイトとするアドレスにロードし、かつssで示されるペアレジスタ (BC, DE, HL) の上位バイトに記憶されたデータを、Tレジスタに記憶されたデータを上位バイト、kで示されるレジスタに記憶されたデータを下位バイトとするアドレスに1を加算したアドレスにロードする命令である。

【0947】

(LDT (k), qq) 命令は、qqで示されるペアレジスタ (IX, IY) の下位バイトに記憶されたデータを、Tレジスタに記憶されたデータを上位バイト、kで示されるレジスタに記憶されたデータを下位バイトとするアドレスにロードし、かつqqで示されるペアレジスタ (IX, IY) の上位バイトに記憶されたデータを、Tレジスタに記憶されたデータを上位バイト、kで示されるレジスタに記憶されたデータを下位バイトとするアドレスに1を加算したアドレスにロードする命令である。

【0948】

(LDT (k), HL) 命令は、Lレジスタに記憶されたデータを、Tレジスタに記憶されたデータを上位バイト、kで示されるレジスタに記憶されたデータを下位バイトとするアドレスにロードし、かつHレジスタに記憶されたデータを、Tレジスタに記憶されたデータを上位バイト、kで示されるレジスタに記憶されたデータを下位バイトとするアドレスに1を加算したアドレスにロードする命令である。

【0949】

(LDWT (r), mn) 命令は、Tレジスタに記憶されたデータを上位バイト、rで示されるレジスタ (A, B, C, D, E, H, L) に記憶されたデータを下位バイトとするアドレスにロードし、かつ16ビットの数値であるmn+1で示されるアドレスに記憶されたデータを、Tレジスタに記憶されたデータを上位バイト、rで示されるレジスタ (A, B, C, D, E, H, L) に記憶されたデータを下位バイトとするアドレスに1を加算したアドレスにロードする命令である。

【0950】

(CLR (ss)) 命令は、ssで示されるペアレジスタ (BC, DE) に数値の0を記憶する命令であり、(CLRW (ss)) 命令は、ssで示されるペアレジスタ (BC, DE, HL) に記憶されたデータが示すアドレスに数値の0を記憶するとともに、当該アドレスに1を加算したアドレスに数値の0を記憶する命令であり、(CLRW (qq)) 命令は、qqで示されるレジスタ (IX, IY) に記憶されたデータが示すアドレスに数値の0を記憶するとともに、当該アドレスに1を加算したアドレスに数値の0を記憶する命令であり、(CLRW (qq+d)) 命令は、qqで示されるレジスタ (IX, IY) に記憶されたデータにdで示されるデータ (d=8~127, -1~-128) を加算した加算結果が示すアドレスに、数値の0を記憶する命令である。

【0951】

(CLRT (k)) 命令は、Tレジスタに記憶されたデータを上位バイト、kで示されるレジスタに記憶されたデータを下位バイトとするアドレスに、数値の0を記憶する命令であり、(CLRTW (k)) 命令は、Tレジスタに記憶されたデータを上位バイト、kで示されるレジスタに記憶されたデータを下位バイトとするアドレスにロードし、かつ、数値の0を、Tレジスタに記憶されたデータを上位バイト、kで示されるレジスタに

10

20

30

40

50



記憶されたデータを下位バイトとするアドレスに1を加算したアドレスに数値の0を記憶する命令である。

#### 【0952】

このように、 $ss$ で示されるレジスタ(BC, DE)や、 $qq$ で示されるレジスタ(IX, IY)は、(CLR ( $ss$ ))命令、(CLRW ( $ss$ ))命令、(CLRW ( $qq+d$ ))命令のオペランドとして指定が可能である。一方、Tレジスタ(特殊レジスタ)は、CLRT命令またはCLRW命令のオペランドとしてのみ指定が可能であることに加えて、オペランドの一部(本実施形態では、 $k$ で示されるアドレスの下位1バイト)は直値でのみ指定が可能である。このため、Tレジスタの値が予期せずに0にクリアされてしまったり、Tレジスタに予期しない値が記憶されてしまうような事態を未然に防止できる場合がある。

10

#### 【0953】

＜他の命令／スタック操作命令＞

スタック操作命令は、スタック領域にデータを退避したり、スタック領域からデータを復帰したりするための命令であり、例えば、PUSH命令、POP命令などが該当する。

#### 【0954】

例えば、(PUSH  $ss$ )命令は、 $ss$ で示されるペアレジスタ(AF, BC, DE, HL)の上位バイトに記憶されたデータを、(SP(スタックポインタ)-1)で示されるアドレスに退避し、かつ、 $ss$ で示されるペアレジスタ(AF, BC, DE, HL)の下位バイトに記憶されたデータを、(SP-2)で示されるアドレスに退避した後に、SPから2を減算する命令である。また、(PUSH TI)命令は、Tレジスタに記憶されたデータを、(SP-1)で示されるアドレスに退避し、かつ、Iレジスタに記憶されたデータを、(SP-2)で示されるアドレスに退避した後に、SPから2を減算する命令である。また、(PUSH  $qq$ )命令は、 $qq$ で示されるペアレジスタ(IX, IY)の上位バイトに記憶されたデータを、(SP-1)で示されるアドレスに退避し、かつ、 $qq$ で示されるペアレジスタ(IX, IY)の下位バイトに記憶されたデータを、(SP-2)で示されるアドレスに退避した後に、SPから2を減算する命令である。

20

#### 【0955】

(PUSH ALL)命令は、全てのレジスタ(T, I, A, F, B, C, D, E, H, L, IXの上位バイト, IXの下位バイト, IYの上位バイト, IYの下位バイト)に記憶されたデータを、この順番で1バイトずつスタック領域に退避した後に、SPから14(レジスタの総数)を減算する命令である。また、(PUSH GPR)命令は、一部のレジスタ(A, F, B, C, D, E, H, L)に記憶されたデータを、この順番で1バイトずつスタック領域に退避した後に、SPから8を減算する命令である。

30

#### 【0956】

(POP  $ss$ )命令は、SPで示されるアドレスに記憶されたデータを、 $ss$ で示されるペアレジスタ(AF, BC, DE, HL)の上位バイトに復帰し、かつ、(SP+1)で示されるアドレスに記憶されたデータを、 $ss$ で示されるペアレジスタ(AF, BC, DE, HL)の下位バイトに復帰した後に、SPに2を加算する命令である。また、(POP TI)命令は、SPで示されるアドレスに記憶されたデータを、Iレジスタに復帰し、かつ、(SP+1)で示されるアドレスに記憶されたデータを、Tレジスタに復帰した後に、SPに2を加算する命令である。また、(POP  $qq$ )命令は、SPで示されるアドレスに記憶されたデータを、 $qq$ で示されるペアレジスタ(IX, IY)の下位バイトに復帰し、かつ、(SP+1)で示されるアドレスに記憶されたデータを、 $qq$ で示されるペアレジスタ(IX, IY)の上位バイトに復帰した後に、SPに2を加算する命令である。

40

#### 【0957】

(POP ALL)命令は、SP、(SP+1)、…、(SP+12)、(SP+13)に記憶されたデータを、この順番で全てのレジスタ(IYの下位バイト, IYの上位バイト, IXの下位バイト, IXの上位バイト, L, H, E, D, C, B, F, A, I, T

50

）に1バイトずつ復帰（IYの下位バイト←（SP）、IYの上位バイト←（SP+1）、…、I←（SP+12）、T←（SP+13））した後に、SPに14を加算する命令である。また、（POP GPR）命令は、SP、（SP+1）、…、（SP+6）、（SP+7）に記憶されたデータを、この順番で一部のレジスタ（L、H、E、D、C、B、F、A）に1バイトずつ復帰（L←（SP）、H←（SP+1）、…、F←（SP+6）、A←（SP+7））した後に、SPに8を加算する命令である。

#### 【0958】

＜他の命令／算術論理演算命令＞

算術論理演算命令は、各種の論理演算を行うための命令であり、例えば、ADD命令、SUB命令、ANDT命令、ORT命令、XORT命令、CPT命令、INC命令、INCT命令、INCWT命令、DEC命令、DECT命令、DECWT命令などが該当する。

10

#### 【0959】

例えば、（ADD A, r）命令は、Aレジスタに記憶されたデータと、rで示されるレジスタ（A、B、C、D、E、H、L）に記憶されたデータを加算し、加算結果をAレジスタに記憶する命令であり、（ADDT r, (k)）命令は、rで示されるレジスタ（B、C、D、E、H、L）に記憶されたデータと、Tレジスタに記憶されたデータを上位バイト、kで示されるレジスタに記憶されたデータを下位バイトとするアドレスに記憶されたデータを加算し、加算結果をrで示されるレジスタ（B、C、D、E、H、L）に記憶する命令である。

20

#### 【0960】

例えば、（SUB r）命令は、Aレジスタに記憶されたデータから、rで示されるレジスタ（A、B、C、D、E、H、L）に記憶されたデータを減算し、減算結果をAレジスタに記憶する命令であり、（SUBT r, (k)）命令は、rで示されるレジスタ（B、C、D、E、H、L）に記憶されたデータから、Tレジスタに記憶されたデータを上位バイト、kで示されるレジスタに記憶されたデータを下位バイトとするアドレスに記憶されたデータを減算し、減算結果をrで示されるレジスタ（B、C、D、E、H、L）に記憶する命令である。

#### 【0961】

例えば、（ANDT (k)）命令は、Aレジスタに記憶されたデータと、Tレジスタに記憶されたデータを上位バイト、kで示されるレジスタに記憶されたデータを下位バイトとするアドレスに記憶されたデータと、の論理積（AND）を算出し、演算結果をAレジスタに記憶する命令であり、（ANDT r, (k)）命令は、rで示されるレジスタ（B、C、D、E、H、L）に記憶されたデータと、Tレジスタに記憶されたデータを上位バイト、kで示されるレジスタに記憶されたデータを下位バイトとするアドレスに記憶されたデータと、の論理積（AND）を算出し、演算結果をrで示されるレジスタ（B、C、D、E、H、L）に記憶する命令である。

30

#### 【0962】

（ORT (k)）命令は、Aレジスタに記憶されたデータと、Tレジスタに記憶されたデータを上位バイト、kで示されるレジスタに記憶されたデータを下位バイトとするアドレスに記憶されたデータと、の論理和（OR）を算出し、演算結果をAレジスタに記憶する命令であり、（ORT r, (k)）命令は、rで示されるレジスタ（B、C、D、E、H、L）に記憶されたデータと、Tレジスタに記憶されたデータを上位バイト、kで示されるレジスタに記憶されたデータを下位バイトとするアドレスに記憶されたデータと、の論理和（OR）を算出し、演算結果をrで示されるレジスタ（B、C、D、E、H、L）に記憶する命令である。

40

#### 【0963】

（XORT (k)）命令は、Aレジスタに記憶されたデータと、Tレジスタに記憶されたデータを上位バイト、kで示されるレジスタに記憶されたデータを下位バイトとするアドレスに記憶されたデータと、の排他的論理和（XOR）を算出し、演算結果をAレジ

50

スタに記憶する命令であり、(X O R T r, (k)) 命令は、rで示されるレジスタ (B, C, D, E, H, L) に記憶されたデータと、Tレジスタに記憶されたデータを上位バイト、kで示されるレジスタに記憶されたデータを下位バイトとするアドレスに記憶されたデータと、の排他的論理和 (X O R) を算出し、演算結果をrで示されるレジスタ (B, C, D, E, H, L) に記憶する命令である。

#### 【0964】

(C P T (k)) 命令は、Aレジスタに記憶されたデータから、Tレジスタに記憶されたデータを上位バイト、kで示されるレジスタに記憶されたデータを下位バイトとするアドレスに記憶されたデータを減算し、演算結果がオーバーフローした場合に所定のフラグを1にセットする一方で、オーバーフローしなかった場合に所定のフラグを0にクリアするとともに、Sフラグ、Zフラグ、TZフラグ、Hフラグ、またはキャリーフラグ (Cフラグ) に記憶された値を0から1 (または1から0) に変化させる命令である。(C P T r, (k)) 命令は、rで示されるレジスタ (B, C, D, E, H, L) に記憶されたデータから、Tレジスタに記憶されたデータを上位バイト、kで示されるレジスタに記憶されたデータを下位バイトとするアドレスに記憶されたデータを減算し、演算結果がオーバーフローした場合に所定のフラグを1にセットする一方で、オーバーフローしなかった場合に所定のフラグを0にクリアするとともに、Sフラグ、Zフラグ、TZフラグ、Hフラグ、またはキャリーフラグ (Cフラグ) に記憶された値を0から1 (または1から0) に変化させる命令である。

#### 【0965】

(I N C T) 命令は、Tレジスタに記憶されたデータに1を加算する命令であり、(I N C T (r)) 命令は、Tレジスタに記憶されたデータを上位バイト、rで示されるレジスタ (A, B, C, D, E, H, L) に記憶されたデータを下位バイトとするアドレスに記憶されたデータに1を加算する命令であり、(I N C T (k)) 命令は、Tレジスタに記憶されたデータを上位バイト、kで示されるレジスタに記憶されたデータを下位バイトとするアドレスに記憶されたデータに1を加算する命令である。また、(I N C W T (k)) 命令は、Tレジスタに記憶されたデータを上位バイト、kで示されるレジスタに記憶されたデータを下位バイトとするアドレスと、当該アドレスに1を加算したアドレスに記憶された2バイト長のデータに1を加算する命令である。

#### 【0966】

(D E C T) 命令は、Tレジスタに記憶されたデータから1を減算する命令であり、(D E C T (r)) 命令は、Tレジスタに記憶されたデータを上位バイト、rで示されるレジスタ (A, B, C, D, E, H, L) に記憶されたデータを下位バイトとするアドレスに記憶されたデータから1を減算する命令であり、(D E C T (k)) 命令は、Tレジスタに記憶されたデータを上位バイト、kで示されるレジスタに記憶されたデータを下位バイトとするアドレスに記憶されたデータから1を減算する命令である。また、(I N C W T (k)) 命令は、Tレジスタに記憶されたデータを上位バイト、kで示されるレジスタに記憶されたデータを下位バイトとするアドレスと、当該アドレスに1を加算したアドレスに記憶された2バイト長のデータから1を減算する命令である。

#### 【0967】

<他の命令／ローテート シフト命令>

算術論理演算命令は、各種の論理演算を行うための命令であり、例えば、R L C 命令、R L C T 命令、R R C 命令、R R C T 命令、R L 命令、R L T 命令、R R 命令、R R T 命令、S L A 命令、S L A T 命令、S R A 命令、S R A T 命令、S R L 命令、S R L T 命令などが該当する。

#### 【0968】

例えば、(R L C r) 命令は、rで示されるレジスタ (A, B, C, D, E, H, L) に記憶されたデータを左に (L S B から M S B 側に) 1ビットだけローテートし、M S B からシフトアウトされたビットを L S B とキャリーフラグに記憶する命令であり、(R L C (H L)) 命令は、H L レジスタに記憶されたデータを左に1ビットだけローテー

10

20

30

40

50

とし、MSBからシフトアウトされたビットをLSBとキャリーフラグに記憶する命令であり、(RLC (qq+d)) 命令は、qqで示されるレジスタ (IX, IY) に記憶されたデータにdで示されるデータ (d=8~127、-1~-128) を加算した加算結果が示すアドレスに記憶されたデータを左に1ビットだけローテートし、MSBからシフトアウトされたビットをLSBとキャリーフラグに記憶する命令であり、(RLCT (k)) 命令は、Tレジスタに記憶されたデータを上位バイト、kで示されるレジスタに記憶されたデータを下位バイトとするアドレスに記憶されたデータを左に1ビットだけローテートし、MSBからシフトアウトされたビットをLSBとキャリーフラグに記憶する命令である。

#### 【0969】

(RRC r) 命令は、rで示されるレジスタ (A, B, C, D, E, H, L) に記憶されたデータを右に (MSBからLSB側に) 1ビットだけローテートし、LSBからシフトアウトされたビットをMSBとキャリーフラグに記憶する命令であり、(RRC (HL)) 命令は、HLレジスタに記憶されたデータを右に1ビットだけローテートし、LSBからシフトアウトされたビットをMSBとキャリーフラグに記憶する命令であり、(RRC (qq+d)) 命令は、qqで示されるレジスタ (IX, IY) に記憶されたデータにdで示されるデータ (d=8~127、-1~-128) を加算した加算結果が示すアドレスに記憶されたデータを右に1ビットだけローテートし、LSBからシフトアウトされたビットをMSBとキャリーフラグに記憶する命令であり、(RLCT (k)) 命令は、Tレジスタに記憶されたデータを上位バイト、kで示されるレジスタに記憶されたデータを下位バイトとするアドレスに記憶されたデータを右に1ビットだけローテートし、LSBからシフトアウトされたビットをMSBとキャリーフラグに記憶する命令である。

#### 【0970】

(RL r) 命令は、rで示されるレジスタ (A, B, C, D, E, H, L) に記憶されたデータを左に (LSBからMSB側に) 1ビットだけローテートし、キャリーフラグに記憶されていたビットをLSBに記憶し、MSBからシフトアウトされたビットをキャリーフラグに記憶する命令であり、(RL (HL)) 命令は、HLレジスタに記憶されたデータを左に1ビットだけローテートし、キャリーフラグに記憶されていたビットをLSBに記憶し、MSBからシフトアウトされたビットをキャリーフラグに記憶する命令であり、(RL (qq+d)) 命令は、qqで示されるレジスタ (IX, IY) に記憶されたデータにdで示されるデータ (d=8~127、-1~-128) を加算した加算結果が示すアドレスに記憶されたデータを左に1ビットだけローテートし、キャリーフラグに記憶されていたビットをLSBに記憶し、MSBからシフトアウトされたビットをキャリーフラグに記憶する命令であり、(RLT (k)) 命令は、Tレジスタに記憶されたデータを上位バイト、kで示されるレジスタに記憶されたデータを下位バイトとするアドレスに記憶されたデータを左に1ビットだけローテートし、キャリーフラグに記憶されていたビットをLSBに記憶し、MSBからシフトアウトされたビットをキャリーフラグに記憶する命令である。

#### 【0971】

(RR r) 命令は、rで示されるレジスタ (A, B, C, D, E, H, L) に記憶されたデータを右に (MSBからLSB側に) 1ビットだけローテートし、キャリーフラグに記憶されていたビットをMSBに記憶し、LSBからシフトアウトされたビットをキャリーフラグに記憶する命令であり、(RR (HL)) 命令は、HLレジスタに記憶されたデータを右に1ビットだけローテートし、キャリーフラグに記憶されていたビットをMSBに記憶し、LSBからシフトアウトされたビットをキャリーフラグに記憶する命令であり、(RR (qq+d)) 命令は、qqで示されるレジスタ (IX, IY) に記憶されたデータにdで示されるデータ (d=8~127、-1~-128) を加算した加算結果が示すアドレスに記憶されたデータを右に1ビットだけローテートし、キャリーフラグに記憶されていたビットをMSBに記憶し、LSBからシフトアウトされたビットをキャ

10

20

30

40

50

リーフラグに記憶する命令であり、(RR T (k)) 命令は、Tレジスタに記憶されたデータを上位バイト、kで示されるレジスタに記憶されたデータを下位バイトとするアドレスに記憶されたデータを右に1ビットだけローテートし、キャリーフラグに記憶されていたビットをMSBに記憶し、LSBからシフトアウトされたビットをキャリーフラグに記憶する命令である。

#### 【0972】

(SLA r) 命令は、rで示されるレジスタ(A, B, C, D, E, H, L)に記憶されたデータを左に(LSBからMSB側に)1ビットだけシフトし、MSBからシフトアウトされたビットをキャリーフラグに記憶し、LSBに0を記憶する命令であり、(SLA (HL)) 命令は、HLレジスタに記憶されたデータを左に1ビットだけシフトし、MSBからシフトアウトされたビットをキャリーフラグに記憶し、LSBに0を記憶する命令であり、(SLA (qq+d)) 命令は、qqで示されるレジスタ(IX, IY)に記憶されたデータにdで示されるデータ(d=8~127、-1~-128)を加算した加算結果が示すアドレスに記憶されたデータを左に1ビットだけシフトし、MSBからシフトアウトされたビットをキャリーフラグに記憶し、LSBに0を記憶する命令であり、(SLAT (k)) 命令は、Tレジスタに記憶されたデータを上位バイト、kで示されるレジスタに記憶されたデータを下位バイトとするアドレスに記憶されたデータを左に1ビットだけシフトし、MSBからシフトアウトされたビットをキャリーフラグに記憶し、LSBに0を記憶する命令である。

#### 【0973】

(SRA r) 命令は、rで示されるレジスタ(A, B, C, D, E, H, L)に記憶されたデータを右に(MSBからLSB側に)1ビットだけシフトし、LSBからシフトアウトされたビットをキャリーフラグに記憶し、MSBに直前のMSBの値を記憶する命令であり、(SLA (HL)) 命令は、HLレジスタに記憶されたデータを右に1ビットだけシフトし、LSBからシフトアウトされたビットをキャリーフラグに記憶し、MSBに直前のMSBの値を記憶する命令であり、(SLA (qq+d)) 命令は、qqで示されるレジスタ(IX, IY)に記憶されたデータにdで示されるデータ(d=8~127、-1~-128)を加算した加算結果が示すアドレスに記憶されたデータを右に1ビットだけシフトし、LSBからシフトアウトされたビットをキャリーフラグに記憶し、MSBに直前のMSBの値を記憶する命令であり、(SLAT (k)) 命令は、Tレジスタに記憶されたデータを上位バイト、kで示されるレジスタに記憶されたデータを下位バイトとするアドレスに記憶されたデータを右に1ビットだけシフトし、LSBからシフトアウトされたビットをキャリーフラグに記憶し、MSBに直前のMSBの値を記憶する命令である。

#### 【0974】

(SRL r) 命令は、rで示されるレジスタ(A, B, C, D, E, H, L)に記憶されたデータを右に(MSBからLSB側に)1ビットだけシフトし、LSBからシフトアウトされたビットをキャリーフラグに記憶し、MSBに0を記憶する命令であり、(SRL (HL)) 命令は、HLレジスタに記憶されたデータを右に1ビットだけシフトし、LSBからシフトアウトされたビットをキャリーフラグに記憶し、MSBに0を記憶する命令であり、(SRL (qq+d)) 命令は、qqで示されるレジスタ(IX, IY)に記憶されたデータにdで示されるデータ(d=8~127、-1~-128)を加算した加算結果が示すアドレスに記憶されたデータを右に1ビットだけシフトし、LSBからシフトアウトされたビットをキャリーフラグに記憶し、MSBに0を記憶する命令であり、(SRLT (k)) 命令は、Tレジスタに記憶されたデータを上位バイト、kで示されるレジスタに記憶されたデータを下位バイトとするアドレスに記憶されたデータを右に1ビットだけシフトし、LSBからシフトアウトされたビットをキャリーフラグに記憶し、MSBに0を記憶する命令である。

#### 【0975】

<他の命令／ビット操作命令>

10

20

30

40

50

ビット操作命令は、ビット単位で各種の操作を行うための命令であり、例えば、B I T命令、S E T T命令、R E S T命令などが該当する。

#### 【0976】

例えば、(B I T T b, (k)) 命令は、Tレジスタに記憶されたデータを上位バイト、kで示されるレジスタに記憶されたデータを下位バイトとするアドレスに記憶されたデータを反転して所定のフラグに記憶する命令であり、(S E T T b, (k)) 命令は、Tレジスタに記憶されたデータを上位バイト、kで示されるレジスタに記憶されたデータに1をセットする命令であり、(R E S T b, (k)) 命令は、Tレジスタに記憶されたデータを上位バイト、kで示されるレジスタに記憶されたデータを下位バイトとするアドレスに記憶されたデータに0をセットする命令である。

10

#### 【0977】

<他の命令／演算&ジャンプ命令>

演算&ジャンプ命令は、演算と当該演算の演算結果に基づく分岐処理を行うための命令であり、例えば、J C P T命令や、J T T命令や、J B I T T命令などが該当する。

#### 【0978】

例えば、(J C P T Z, A, (k), e) 命令は、Aレジスタに記憶されたデータから、Tレジスタに記憶されたデータを上位バイト、kで示されるレジスタに記憶されたデータを下位バイトとするアドレスに記憶されたデータを減算し、Zフラグが1の場合にはeで示されるアドレスをP C (プログラムカウンタ) にロードし (eで示されるアドレスに分岐し)、Zフラグが0の場合には後続の命令を実行させる命令であり、(J C P T N Z, A, (k), e) 命令は、Aレジスタに記憶されたデータから、Tレジスタに記憶されたデータを上位バイト、kで示されるレジスタに記憶されたデータを下位バイトとするアドレスに記憶されたデータを減算し、Zフラグが0の場合にはeで示されるアドレスをP C (プログラムカウンタ) にロードし (eで示されるアドレスに分岐し)、Zフラグが1の場合には後続の命令を実行させる命令である。

20

#### 【0979】

(J C P T C, A, (k), e) 命令は、Aレジスタに記憶されたデータから、Tレジスタに記憶されたデータを上位バイト、kで示されるレジスタに記憶されたデータを下位バイトとするアドレスに記憶されたデータを下位バイトとするアドレスに記憶されたデータを減算し、キャリーフラグが1の場合にはeで示されるアドレスをP C (プログラムカウンタ) にロードし (eで示されるアドレスに分岐し)、キャリーフラグが0の場合には後続の命令を実行させる命令であり、(J C P T N C, A, (k), e) 命令は、Aレジスタに記憶されたデータから、Tレジスタに記憶されたデータを上位バイト、kで示されるレジスタに記憶されたデータを下位バイトとするアドレスに記憶されたデータを減算し、キャリーフラグが0の場合にはeで示されるアドレスをP C (プログラムカウンタ) にロードし (eで示されるアドレスに分岐し)、キャリーフラグが1の場合には後続の命令を実行させる命令である。

30

#### 【0980】

(J C P T Z, (k), s s, e) 命令は、Tレジスタに記憶されたデータを上位バイト、kで示されるレジスタに記憶されたデータを下位バイトとするアドレスと、当該アドレスに1を加算したアドレスに記憶された2バイトのデータから、s sで示されるペアレジスタ (B C, D E) に記憶された2バイトのデータを減算し、Zフラグが1の場合にはeで示されるアドレスをP C (プログラムカウンタ) にロードし (eで示されるアドレスに分岐し)、Zフラグが0の場合には後続の命令を実行させる命令であり、(J C P T N Z, (k), s s, e) 命令は、Tレジスタに記憶されたデータを上位バイト、kで示されるレジスタに記憶されたデータを下位バイトとするアドレスと、当該アドレスに1を加算したアドレスに記憶された2バイトのデータから、s sで示されるペアレジスタ (B C, D E) に記憶された2バイトのデータを減算し、Zフラグが0の場合にはeで示されるアドレスをP C (プログラムカウンタ) にロードし (eで示されるアドレスに分岐し)、Zフラグが1の場合には後続の命令を実行させる命令である。

40

50

## 【0981】

(JCPT C, (k), ss, e) 命令は、Tレジスタに記憶されたデータを上位バイト、kで示されるレジスタに記憶されたデータを下位バイトとするアドレスと、当該アドレスに1を加算したアドレスに記憶された2バイトのデータから、ssで示されるペアレジスタ(BC, DE)に記憶された2バイトのデータを減算し、キャリーフラグが1の場合にはeで示されるアドレスをPC(プログラムカウンタ)にロードし(eで示されるアドレスに分岐し)、キャリーフラグが0の場合には後続の命令を実行させる命令であり、(JCPT NC, (k), ss, e) 命令は、Tレジスタに記憶されたデータを上位バイト、kで示されるレジスタに記憶されたデータを下位バイトとするアドレスと、当該アドレスに1を加算したアドレスに記憶された2バイトのデータから、ssで示されるペアレジスタ(BC, DE)に記憶された2バイトのデータを減算し、キャリーフラグが0の場合にはeで示されるアドレスをPC(プログラムカウンタ)にロードし(eで示されるアドレスに分岐し)、キャリーフラグが1の場合には後続の命令を実行させる命令である。

10

## 【0982】

(JTT Z, (k) e) 命令は、Tレジスタに記憶されたデータを上位バイト、kで示されるレジスタに記憶されたデータを下位バイトとするアドレスに記憶されたデータから、数値の0を減算し、Zフラグが1の場合にはeで示されるアドレスをPC(プログラムカウンタ)にロードし(eで示されるアドレスに分岐し)、Zフラグが0の場合には後続の命令を実行させる命令であり、(JTT NZ, (k) e) 命令は、Tレジスタに記憶されたデータを上位バイト、kで示されるレジスタに記憶されたデータを下位バイトとするアドレスに記憶されたデータから、数値の0を減算し、Zフラグが0の場合にはeで示されるアドレスをPC(プログラムカウンタ)にロードし(eで示されるアドレスに分岐し)、Zフラグが1の場合には後続の命令を実行させる命令である。

20

## 【0983】

(JBITT Z, b, (k), e) 命令は、Tレジスタに記憶されたデータを上位バイト、kで示されるレジスタに記憶されたデータを下位バイトとするアドレスに記憶されたデータのうち、bで示されるビットの反転データをZフラグに記憶し、Zフラグが1の場合にはeで示されるアドレスをPC(プログラムカウンタ)にロードし(eで示されるアドレスに分岐し)、Zフラグが0の場合には後続の命令を実行させる命令であり、(JBITT NZ, b, (k), e) 命令は、Tレジスタに記憶されたデータを上位バイト、kで示されるレジスタに記憶されたデータを下位バイトとするアドレスに記憶されたデータのうち、bで示されるビットの反転データをZフラグに記憶し、Zフラグが0の場合にはeで示されるアドレスをPC(プログラムカウンタ)にロードし(eで示されるアドレスに分岐し)、Zフラグが1の場合には後続の命令を実行させる命令である。

30

## 【0984】

<他の命令／演算&リターン命令>

演算&リターン命令は、演算と当該演算の演算結果に基づく復帰処理を行うための命令であり、例えば、RTT命令などが該当する。

## 【0985】

例えば、(RTT Z, (k)) 命令は、Tレジスタに記憶されたデータを上位バイト、kで示されるレジスタに記憶されたデータを下位バイトとするアドレスに記憶されたデータから、数値の0を減算し、Zフラグが1の場合にはSPと(SP+1)に記憶されたアドレスをPC(プログラムカウンタ)にロードし(SPと(SP+1)で示されるアドレスに復帰し)、Zフラグが0の場合には後続の命令を実行させる命令であり、(RTT NZ, (k)) 命令は、Tレジスタに記憶されたデータを上位バイト、kで示されるレジスタに記憶されたデータを下位バイトとするアドレスに記憶されたデータから、数値の0を減算し、Zフラグが0の場合にはSPと(SP+1)に記憶されたアドレスをPC(プログラムカウンタ)にロードし(SPと(SP+1)で示されるアドレスに復帰し)、Zフラグが1の場合には後続の命令を実行させる命令である。

40

50

## 【0986】

<他の命令／複合命令>

複合命令は、複数の処理を一つにした命令であり、例えば、INCPULD命令、INCPWULD命令、DECPULD命令、DECPWULD命令などが該当する。

## 【0987】

(INCPULD A, r) 命令は、Aレジスタに記憶されたデータから、rで示されるレジスタ (B, C, D, E, H, L) に記憶されたデータを減算し、キャリーフラグが1の場合にAレジスタに記憶されたデータに1を加算し、キャリーフラグが0の場合にAレジスタに0を記憶する命令であり、(INCPULD (HL), r) 命令は、HLレジスタに記憶されたデータが示すアドレスに記憶されたデータから、rで示されるレジスタ (B, C, D, E, H, L) に記憶されたデータを減算し、キャリーフラグが1の場合に、HLレジスタに記憶されたデータが示すアドレスに記憶されたデータに1を加算し、キャリーフラグが0の場合に、HLレジスタに記憶されたデータが示すアドレスに記憶されたデータに0を記憶する命令である。

10

## 【0988】

(INCPULD A, n) 命令は、Aレジスタに記憶されたデータから、数値nを減算し、キャリーフラグが1の場合にAレジスタに記憶されたデータに1を加算し、キャリーフラグが0の場合にAレジスタに0を記憶する命令であり、(INCPULD (HL), n) 命令は、HLレジスタに記憶されたデータが示すアドレスに記憶されたデータから、数値nを減算し、キャリーフラグが1の場合に、HLレジスタに記憶されたデータが示すアドレスに記憶されたデータに1を加算し、キャリーフラグが0の場合に、HLレジスタに記憶されたデータが示すアドレスに記憶されたデータに0を記憶する命令である。

20

## 【0989】

(INCPULD HL, mn) 命令は、HLレジスタに記憶されたデータから、数値mnを減算し、キャリーフラグが1の場合にHLレジスタに記憶されたデータに1を加算し、キャリーフラグが0の場合にHLレジスタに0を記憶する命令であり、(INCPWULD (HL), mn) 命令は、HLレジスタに記憶されたデータが示すアドレスと、HLレジスタに記憶されたデータ+1が示すアドレスに記憶されたデータから、数値mnを減算し、キャリーフラグが1の場合に、HLレジスタに記憶されたデータが示すアドレスと、HLレジスタに記憶されたデータ+1が示すアドレスに記憶されたデータに1を加算し、キャリーフラグが0の場合に、HLレジスタに記憶されたデータが示すアドレスと、HLレジスタに記憶されたデータ+1が示すアドレスに記憶されたデータに0を記憶する命令である。

30

## 【0990】

(DECPULD A, r) 命令は、Aレジスタに記憶されたデータから1を減算し、キャリーフラグが1の場合に、rで示されるレジスタ (B, C, D, E, H, L) に記憶されたデータをAレジスタに記憶し、キャリーフラグが0の場合に、Aレジスタから1を減算する命令であり、(DECPULD (HL), r) 命令は、HLレジスタに記憶されたデータが示すアドレスに記憶されたデータから、rで示されるレジスタ (B, C, D, E, H, L) に記憶されたデータを減算し、キャリーフラグが1の場合に、rで示されるレジスタ (B, C, D, E, H, L) に記憶されたデータを、HLレジスタに記憶されたデータが示すアドレスに記憶されたデータに記憶し、キャリーフラグが0の場合に、HLレジスタに記憶されたデータが示すアドレスに記憶されたデータから1を減算する命令である。

40

## 【0991】

(DECPULD A, n) 命令は、Aレジスタに記憶されたデータから、数値nを減算し、キャリーフラグが1の場合に、Aレジスタに数値nを記憶し、キャリーフラグが0の場合に、Aレジスタに記憶されたデータから1を減算する命令であり、(DECPULD (HL), n) 命令は、HLレジスタに記憶されたデータが示すアドレスに記憶されたデータから、数値nを減算し、キャリーフラグが1の場合に、HLレジスタに記憶されたデ

50



ータが示すアドレスに記憶されたデータに数値  $n$  を記憶し、キャリーフラグが 0 の場合に、HL レジスタに記憶されたデータが示すアドレスに記憶されたデータから 1 を減算する命令である。

#### 【0992】

(DECP LD HL,  $mn$ ) 命令は、HL レジスタに記憶されたデータから、数値  $mn$  を減算し、キャリーフラグが 1 の場合に、HL レジスタに数値  $mn$  を記憶し、キャリーフラグが 0 の場合に、HL レジスタに記憶されたデータから 1 を減算する命令であり、(DECP WLD (HL),  $mn$ ) 命令は、HL レジスタに記憶されたデータが示すアドレスと、HL レジスタに記憶されたデータ + 1 が示すアドレスに記憶されたデータから、数値  $mn$  を減算し、キャリーフラグが 1 の場合に、HL レジスタに記憶されたデータが示すアドレスと、HL レジスタに記憶されたデータ + 1 が示すアドレスに記憶されたデータに数値  $mn$  を記憶し、キャリーフラグが 0 の場合に、HL レジスタに記憶されたデータが示すアドレスと、HL レジスタに記憶されたデータ + 1 が示すアドレスに記憶されたデータから 1 を減算する命令である。

#### 【0993】

例えば、上述の主制御部メイン処理の基本乱数初期値更新処理(ステップ S115)において、A レジスタに普図タイマ乱数値、数値  $n$  に普図タイマ乱数値の最大値(例えば、100)をそれぞれ記憶(ロード)した後に、(INCP LD A,  $n$ ) 命令を実行すれば、キャリーフラグが 1 の場合、すなわち、普図タイマ乱数値が最大値ではない場合には、普図タイマ乱数値に 1 を加算し、キャリーフラグが 0 の場合、すなわち、普図タイマ乱数値が最大値になった場合には、普図タイマ乱数値に 0 を記憶することができる。このため、(INCP LD A,  $n$ ) 命令だけで、普図タイマ乱数値の更新を容易に行うことができ、プログラム容量の削減や、処理速度の向上を図ることができる場合がある。

#### 【0994】

また、上述の主制御部メイン処理の基本乱数初期値更新処理(ステップ S115)において、A レジスタに普図タイマ乱数値、数値  $n$  に普図タイマ乱数値の最大値(例えば、100)をそれぞれ記憶(ロード)した後に、(DECP LD A,  $n$ ) 命令を実行すれば、キャリーフラグが 1 の場合、すなわち、普図タイマ乱数値が負の値になった場合には、普図タイマ乱数値に普図タイマ乱数値の最大値を記憶し、キャリーフラグが 0 の場合、すなわち、普図タイマ乱数値が負の値ではない場合には、普図タイマ乱数値から 1 を減算することができる。このため、(DECP LD A,  $n$ ) 命令だけで、普図タイマ乱数値の更新を容易に行うことができ、プログラム容量の削減や、処理速度の向上を図ることができる場合がある。なお、この例では、A レジスタのみで乱数値を更新する例を示したが、他のレジスタ(例えば、R レジスタ)を併用したり、ハードウェア乱数値を併用してもよい。

#### 【0995】

以上説明したように、上記実施形態に係る遊技台は、CPU (例えば、CPU 304) と、遊技制御プログラムが少なくとも記憶されている ROM (例えば、ROM 306) と、を少なくとも内蔵するマイクロプロセッサ(例えば、基本回路 302)を備えた遊技台であって、前記遊技台は、パチンコ機またはスロットマシンであり、前記マイクロプロセッサは、少なくとも所定の出力端子(例えば、リセット出力端子 XRS TO)を有し、前記マイクロプロセッサは、少なくともランダム延長機能(例えば、ランダム延長処理)を有し、前記ランダム延長機能は、前記遊技制御プログラムの実行開始タイミングをランダムに変化させることが可能なものであり、前記所定の出力端子からの出力は、少なくとも所定のタイミングで第一のレベルから該第一のレベル(例えば、ローレベルの信号)よりも高い第二のレベル(例えば、ハイレベルの信号)に変化するものであり、前記所定のタイミングは、前記ランダム延長機能の実行前である、ことを特徴とする遊技台である。

#### 【0996】

また、上記実施形態に係る遊技台は、CPU (例えば、CPU 304) と、遊技制御プログラムを記憶する ROM (例えば、ROM 306) と、を少なくとも内蔵するマイクロ

プロセッサ（例えば、基本回路 302）を備えた遊技台であって、前記マイクロプロセッサは、少なくとも所定の出力端子（例えば、リセット出力端子 X R S T O）を有し、前記マイクロプロセッサは、前記所定の出力端子（例えば、リセット出力端子 X R S T O）からの所定の信号出力（例えば、リセット出力信号出力）の立上りタイミングまたは立下りタイミング（例えば、ローレベル信号からハイレベル信号に変化させる立上りタイミング）を、前記遊技制御プログラムの実行開始前に行われるランダム遅延処理の開始前に実行可能である、ことを特徴とする遊技台である。

#### 【0997】

図 127 および図 128 は、本発明に係る所定の出力端子の信号出力の一例を説明するための図である。同図には、所定の出力端子の一例であるリセット出力端子 X R S T O の出力信号の電圧変化と、本発明に係る第一のレベルであるか否かを判断するための基準電圧であるローレベルしきい値電圧  $V_L$  と、本発明に係る第二のレベルであるか否かを判断するための基準電圧であるハイレベルしきい値電圧  $V_H$  を示している。

#### 【0998】

この例では、リセット出力端子 X R S T O の出力信号の電圧がローレベルしきい値電圧  $V_L$  よりも低い場合には第一のレベル（ローレベル）と規定し、リセット出力端子 X R S T O から出力する電圧がハイレベルしきい値電圧  $V_H$  よりも高い場合には第二のレベル（ハイレベル）と規定している。なお、この例では、ローレベルしきい値電圧  $V_L$  とハイレベルしきい値電圧  $V_H$  にヒステリシスを持たせているが、本発明はこれに限定されず、例えば、ローレベルしきい値電圧  $V_L$  とハイレベルしきい値電圧  $V_H$  を同一の電圧（例えば、ハイレベルの最高電圧が 5 V、ローレベルの最低電圧が 0 V の場合に、両者の平均の 2.5 V）に設定してもよい。

#### 【0999】

図 127（a）では、リセット出力端子 X R S T O の出力信号の電圧を、ランダム延長処理の実行開始後に、第一のレベル（ローレベル）の電圧からローレベルしきい値電圧  $V_L$  を超える電圧に変化させた後に、ランダム延長処理の実行中に、さらにハイレベルしきい値電圧  $V_H$  を超える電圧に変化させて第二のレベル（ハイレベル）に変化させている。同図（b）では、リセット出力端子 X R S T O の出力信号の電圧を、ランダム延長処理の実行開始とほぼ同時に、第一のレベル（ローレベル）の電圧からローレベルしきい値電圧  $V_L$  を超える電圧に変化させた後に、ランダム延長処理の実行中に、さらにハイレベルしきい値電圧  $V_H$  を超える電圧に変化させて第二のレベル（ハイレベル）に変化させている。

#### 【1000】

また、同図（c）では、リセット出力端子 X R S T O の出力信号の電圧を、ランダム延長処理の実行開始前に、第一のレベル（ローレベル）の電圧からローレベルしきい値電圧  $V_L$  を超える電圧に変化させた後に、ランダム延長処理の実行中に、ランダム延長処理の実行開始後に、さらにハイレベルしきい値電圧  $V_H$  を超える電圧に変化させて第二のレベル（ハイレベル）に変化させている。同図（d）では、リセット出力端子 X R S T O の出力信号の電圧を、ランダム延長処理の実行開始前に、第一のレベル（ローレベル）の電圧からローレベルしきい値電圧  $V_L$  を超える電圧に変化させた後に、ランダム延長処理の実行開始とほぼ同時に、さらにハイレベルしきい値電圧  $V_H$  を超える電圧に変化させて第二のレベル（ハイレベル）に変化させている。

#### 【1001】

一方、同図（e）では、リセット出力端子 X R S T O の出力信号の電圧を、ランダム延長処理の実行開始前に、第一のレベル（ローレベル）の電圧からローレベルしきい値電圧  $V_L$  を超える電圧に変化させた後に、ランダム延長処理の実行開始前に、さらにハイレベルしきい値電圧  $V_H$  を超える電圧に変化させて第二のレベル（ハイレベル）に変化させている。図 128（a）では、リセット出力端子 X R S T O の出力信号の電圧を、ランダム延長処理の実行開始前、かつ、固定延長処理の実行終了後に、第一のレベル（ローレベル）の電圧からローレベルしきい値電圧  $V_L$  を超える電圧に変化させた後に、ランダム延長

処理の実行開始前、かつ、固定延長処理の実行終了後に、さらにハイレベルしきい値電圧  $V_H$  を超える電圧に変化させて第二のレベル（ハイレベル）に変化させている。また、同図（b）では、リセット出力端子  $X R S T O$  の出力信号の電圧を、ランダム延長処理の実行開始前、かつ、固定延長処理の実行終了とほぼ同時に、第一のレベル（ローレベル）の電圧からローレベルしきい値電圧  $V_L$  を超える電圧に変化させた後に、ランダム延長処理の実行開始前、かつ、固定延長処理の実行終了後に、さらにハイレベルしきい値電圧  $V_H$  を超える電圧に変化させて第二のレベル（ハイレベル）に変化させている。

#### 【1002】

また、前記マイクロプロセッサは、前記所定の出力端子（例えば、リセット出力端子  $X R S T O$ ）からの所定の信号出力（例えば、リセット出力信号出力）を、前記遊技制御プログラムの実行開始前に行われるランダム遅延処理の終了前（または、ランダム遅延処理中）に実行可能であっても、同様の効果を得ることができる。

10

#### 【1003】

上記実施形態に係る遊技台によれば、所定の信号出力のタイミングと遊技制御プログラムの実行開始タイミングをずらすことが可能なため、遊技制御処理の特定の処理（例えばソフト乱数の更新処理、初期値更新処理等）の狙い撃ちを防止することができる場合がある。

#### 【1004】

また、前記所定の出力端子はリセット信号を出力可能な端子であってもよい。このような構成とすれば、例えば、リセット信号を所定の電子部品に接続した場合、遊技制御前に電子部品が動作可能となり、不正な電波を送信した場合に電子部品が動作してしまうため、不正行為を発見しやすくなる場合がある。また、遊技制御プログラムの開始前に電子部品が動作可能なので、電子部品の作動開始タイミングと、遊技制御プログラムの実行開始タイミングとが異なると思わせることにより、遊技制御処理の特定の処理の狙い撃ちを防止することができる場合がある。

20

#### 【1005】

また、少なくとも、セキュリティモードおよびユーザモードを備え、前記遊技制御プログラムは、前記ユーザモードで実行されるユーザプログラムであってもよい。

#### 【1006】

このような構成とすれば、所定の信号出力のタイミングとユーザプログラムの実行開始タイミングをずらすことが可能なため、遊技制御処理の特定の処理（例えばソフト乱数の更新処理、初期値更新処理等）の狙い撃ちを防止することができる場合がある。

30

#### 【1007】

また、前記遊技制御プログラムの実行開始は、システムリセットおよびユーザリセットのうちの少なくともいずれか一方を受け付けた後でおこなわれるものであってもよい。

#### 【1008】

このような構成とすれば、システムリセットまたはユーザリセットの度に所定の信号出力のタイミングと遊技制御プログラムの実行開始タイミングをずらすことが可能なため、遊技制御処理の特定の処理（例えばソフト乱数の更新処理、初期値更新処理等）の狙い撃ちを防止することができる場合がある。

40

#### 【1009】

また、少なくとも所定の入力端子を有する所定の電子部品（例えば、 $I C 1 1 \sim I C 1 4$ ）を備え、前記所定の入力端子（例えば、リセット入力端子）は、前記所定の出力端子からの前記所定の信号を入力可能な端子であってもよい。

#### 【1010】

このような構成とすれば、遊技制御プログラムの開始前に電子部品が動作可能なので、電子部品の作動開始タイミングと、遊技制御プログラムの実行開始タイミングとが異なると思わせることにより、遊技制御処理の特定の処理の狙い撃ちを防止することができる場合がある。

#### 【1011】

50

図129は、ユーザリセット時におけるシステムクロック信号、アドレス信号、データ信号、制御信号、およびリセット出力信号の変化を示した図である。上述のとおり、本実施形態のユーザリセットには、WDTのタイムアウトによるユーザリセットと、指定エリア外のアクセスによるユーザリセットの2種類があり、プログラム管理エリアの設定により、ユーザリセットを有効に設定している場合に機能する。

#### 【1012】

WDTのタイムアウトによるユーザリセットは、WDTタイムアウト信号に基づいて発生するリセットであり、当該リセット中はアドレス信号として第一の固定値（例えば、E000H）を出力するように構成している。一方、指定エリア外のアクセスによるユーザリセットは、指定エリア外走行禁止信号に基づいて発生するリセットであり、当該リセット中はアドレス信号として第二の固定値（例えば、8000Hなど、E000H以外の値）を出力するように構成している。

10

#### 【1013】

ユーザリセットが発生すると、CPUコア、タイマ回路、演算回路、入出力ポート、通信回路、割込みコントローラなどが初期化され、リセットアドレス（0000H）よりユーザプログラム（遊技制御プログラム）の実行を開始し、起動されていたウォッチドッグタイマも制御プログラムによらずに（自動的に）クリア&リスタートされる。

#### 【1014】

このように、前記マイクロプロセッサは、所定の条件が成立した場合（例えば、図115に示すタイムアウト時間が経過した場合）に、所定の信号（例えば、WDT起動信号（WDTタイムアウト信号）や、CPU内部の信号）を少なくとも発生させることが可能なウォッチドッグタイマ（例えば、WDT314、第一WDT314a、第二WDT314b）と、正当な指定エリア以外の記憶領域のコードが実行されたことに基づいて、指定エリア外走行禁止信号を少なくとも発生させることが可能な指定エリア外走行禁止回路（例えば、指定エリア外走行禁止回路3142）と、を少なくとも有し、前記指定エリア外走行禁止信号が発生してから前記所定のタイミングの間に開始されるセキュリティチェック中に、前記ウォッチドッグタイマのカウンタが停止している期間が少なくとも含まれていてもよい。

20

#### 【1015】

なお、本発明に係る「ウォッチドッグタイマのカウンタが停止している」には、カウンタが一時的に停止している場合、カウンタが初期値や0に設定された後にカウンタが一時的に停止している場合、カウンタのカウントは継続するが、カウンタがタイムアウトする前にカウンタをクリア&リスタートし続けることによってカウンタが実質的に停止する（タイムアウトさせない）場合、カウンタの値をセキュリティチェック時間の最大処理時間よりも長く設定し、セキュリティチェック中のタイムアウトを回避することによってカウンタが実質的に停止する（タイムアウトさせない）場合などが含まれる。また、「ウォッチドッグタイマのカウンタが停止している期間」には、指定エリア外走行禁止信号の発生から当該信号の発生直後までの期間、指定エリア外走行禁止信号の発生後からシステムクロックが所定クロック経過した後（例えば、指定エリア外走行禁止信号の発生後からシステムクロックの1クロック後の立ち下がりエッジまたは立ち上がりエッジまで）、リセット出力端子XRSTOの出力信号の立上りまでの期間の一部やすべてなども含まれる。また、WDTをマイコンに内蔵し、WDTタイムアウトが発生してから前記所定のタイミングの間に開始されるセキュリティチェック中に、WDT用カウンタが停止している期間が少なくとも含まれ、WDTをマイコンに内蔵し、指定エリア外走行禁止信号が発生してから前記所定のタイミングの間に開始されるセキュリティチェック中に、WDT用カウンタが停止している期間が少なくとも含まれる。

30

40

#### 【1016】

このような構成とすれば、指定エリア外走行禁止信号に基づくリセット処理を行う際に、ウォッチドッグタイマのタイムアウトを制御プログラムによらずに（自動的に）停止させることができ、指定エリア外走行禁止信号に基づくリセット処理を迅速に行うことがで

50

きる場合がある。

【1017】

また、本実施形態に係るパチンコ機100は、CPU（例えば、CPU304）と、遊技制御プログラムが少なくとも記憶されているROM（例えば、ROM306）と、を少なくとも備えた遊技台であって、前記遊技台は、パチンコ機またはスロットマシンであり、所定の条件が成立した場合（例えば、図115に示すタイムアウト時間が経過した場合）に、所定の信号（例えば、WDT起動信号（WDTタイムアウト信号）や、CPU内部の信号）を少なくとも発生させるウォッチドッグタイマ（例えば、WDT314、第一WDT314a、第二WDT314b）を備え、前記CPUは、前記所定の信号の発生を契機として前記遊技制御プログラムを所定のアドレス（例えば、図62に示す0000H番地）から実行可能であり、前記所定の信号の発生から前記遊技制御プログラムの実行開始までの時間を調整可能に構成されている、ことを特徴とする遊技台である。

10

【1018】

本実施形態に係るパチンコ機100によれば、遊技制御用プログラムが実行されるまでの時間を変化させることができ、遊技制御用プログラムの実行タイミング（例えば、当否判定の抽選タイミング）を把握するような不正行為を未然に防止でき、安定した遊技制御をおこなうことができる場合がある。

【1019】

なお、本発明に係るウォッチドッグタイマは、CPUと別体のもの、CPUに内蔵のもの、の両方を含み、前者の場合、リセット信号はウォッチドッグタイマからCPUに入力されるため信号線が外部に露出するが、後者の場合、リセット信号はCPU内部で配線されるため信号線は外部には露出しない。したがって、後者の場合には、WDTのタイムアウトに基づいてリセット信号が発生したか否か（CPUにリセットがかかったか否か）を隠蔽することができ、遊技制御用プログラムの実行タイミングを把握するような不正行為を、より確実に防止できる場合がある。

20

【1020】

また、前記CPUと前記ROMを少なくとも内蔵するマイクロプロセッサ（例えば、基本回路302）を備え、前記マイクロプロセッサは、少なくとも所定の出力端子（例えば、リセット出力端子XRSTO）を有し、前記所定の出力端子は、少なくとも所定のタイミングで立ち上がることが可能なものであり、前記所定のタイミングは、前記遊技制御プログラムの実行開始前に行われるランダム遅延処理の開始前であってもよい。

30

【1021】

このような構成とすれば、所定の信号出力のタイミングと遊技制御プログラムの実行開始タイミングをずらすことが可能なため、遊技制御処理の特定の処理（例えばソフト乱数の更新処理、初期値更新処理等）の狙い撃ちを防止することができる場合がある。

【1022】

また、前記CPUは、所定の出力端子（例えば、リセット出力端子XRSTO）からの信号（例えば、リセット出力信号出力）の出力の立上りタイミングまたは立下りタイミング（例えば、ローレベル信号からハイレベル信号に変化させる立上りタイミング）を、前記遊技制御プログラムの実行開始前に行われるランダム遅延処理の開始前に実行可能であってもよい。

40

【1023】

このような構成とすれば、遊技制御用プログラムが実行されるまでの時間を変化させながらも、リセット信号の発生から所定の信号の出力までの時間を一定にすることができ、安定した遊技制御をおこなうことができる場合がある。

【1024】

また、前記所定の出力端子はリセット信号を出力可能な端子であってもよい。このような構成とすれば、例えば、リセット信号を所定の電子部品に接続した場合、遊技制御前に電子部品が動作可能となり、不正な電波を送信した場合に電子部品が動作してしまうため、不正行為を発見しやすくなる場合がある。また、遊技制御プログラムの開始前に電子部

50

品が動作可能なので、電子部品の作動開始タイミングと、遊技制御プログラムの実行開始タイミングとが異なると思わせることにより、遊技制御処理の特定の処理の狙い撃ちを防止することができる場合がある。

#### 【1025】

また、少なくとも、セキュリティモードおよびユーザモードを備え、前記遊技制御プログラムは、前記ユーザモードで実行されるユーザプログラムであってもよい。

#### 【1026】

このような構成とすれば、所定の信号出力のタイミングとユーザプログラムの実行開始タイミングをずらすことが可能なため、遊技制御処理の特定の処理（例えばソフト乱数の更新処理、初期値更新処理等）の狙い撃ちを防止することができる場合がある。

10

#### 【1027】

また、リセット要因を示す所定の情報（例えば、直前に発生したリセット要因がシステムリセットであるか否かを示す情報）が少なくともセットされるレジスタ（例えば、図104に示す第4内部情報レジスタ）を備え、前記遊技制御プログラムは、前記所定の情報に基づく処理（例えば、第4内部情報レジスタから読み出した情報を汎用レジスタに書き込む処理）を含まないように（実行しないように）構成してもよい。

#### 【1028】

このような構成とすれば、故意にCPUのリセットを発生させて特定の処理（例えばソフト乱数の更新処理、初期値更新処理等）の実行タイミングなどを狙うような不正行為を防止できる場合がある。

20

#### 【1029】

また、前記マイクロプロセッサは、前記ウォッチドッグタイマと、正当な指定エリア以外の記憶領域のコードが実行されたことに基づいて、指定エリア外走行禁止信号を少なくとも発生させることが可能な指定エリア外走行禁止回路（例えば、指定エリア外走行禁止回路3142）と、を少なくとも有し、前記指定エリア外走行禁止信号が発生してから前記所定のタイミングの間に開始されるセキュリティチェック中に、前記ウォッチドッグタイマのカウンタが停止している期間が少なくとも含まれていてもよい。

#### 【1030】

このような構成とすれば、指定エリア外走行禁止信号に基づくリセット処理を行う際に、ウォッチドッグタイマのタイムアウトを制御プログラムによらずに（自動的に）停止させることができ、指定エリア外走行禁止信号に基づくリセット処理を迅速に行うことができる場合がある。

30

#### 【1031】

また、本実施形態に係るパチンコ機100は、CPU（例えば、CPU304）と、遊技制御プログラムが少なくとも記憶されているROM（例えば、ROM306）と、を少なくとも備えた遊技台であって、前記遊技台は、パチンコ機またはスロットマシンであり、前記CPUは、少なくとも特定のレジスタ（例えば、Tレジスタ）を備え、前記遊技制御プログラムは、所定の処理（例えば、図31に示す特図1関連抽選処理）と、該所定の処理から呼び出される所定のサブ処理（例えば、図34のステップS1103の「保留数記憶領域（RWM領域のアドレスF040H）に記憶された保留数を、Aレジスタに転送する処理」）を少なくとも含み、前記所定の処理は、第一の処理と第二の処理（例えば、図31に示す、特別図柄乱数移行処理、特別図柄当り抽選処理、表示図柄抽選処理、特別図柄変動時間抽選処理、特別図柄変動時間設定処理、特別図柄保留内情報移行処理のうちの2つの処理である第一の処理と第二の処理）を少なくとも含み、前記所定のサブ処理は、前記第一の処理と前記第二の処理の両方から呼び出され、前記特定のレジスタを用いた処理は、前記所定のサブ処理で少なくとも実行可能であることを特徴とする遊技台である。

40

#### 【1032】

なお、本発明に係る第一の処理は、主制御部メイン処理（メインループ処理）から呼び出される処理（例えば、図10の主制御部メイン処理に示す各処理）、本発明に係る第二

50

の処理は、主制御部タイマ割込処理（タイマ割込処理）から呼び出される処理（例えば、図14の主制御部タイマ割込処理に示す各処理）でもよく、本発明に係る所定のサブ処理は、これらの主制御部メイン処理から呼び出される処理と、主制御部タイマ割込処理から呼び出される処理の両方から呼び出されるものでもよい。

#### 【1033】

本実施形態に係るパチンコ機100によれば、2つの異なる処理（第一の処理と第二の処理）から呼び出されるサブ処理において、共通の特定のレジスタを用いた処理を実行するため、遊技制御プログラムを簡素化することができ、プログラムのコード量を従来よりも減らすことができる上に、コーディングミスの発生を減少させることができ、安定した遊技制御を行うことができる場合がある。

10

#### 【1034】

また、前記CPUは、所定の命令（例えば、図64に示すRST命令）を少なくとも実行可能であり、前記所定の命令は、該命令が実行された処理から、予め定められたアドレス（例えば、0008H、0010H、0018H、0020H、0028H、0030H、0038H、0040Hの8種類のアドレス）に配置された処理のみを呼出可能な命令であり、前記所定のサブ処理は、前記第一の処理と前記第二の処理の両方から前記所定の命令を用いて呼出可能な処理のうちの一つであってもよい。

#### 【1035】

このような構成とすれば、遊技制御プログラムの解析や改ざんが困難となるため、不正にサブ処理を読み出すプログラムを組み込んで不正な利益を得ようとする行為や、不正にRAMに記憶された内部情報を取得するような行為を未然に防止できる場合がある。

20

#### 【1036】

また、前記CPUは、前記特定のレジスタに値をセットする機能のうち、ロード命令を受けたことに基づいて行われるもの以外のものとして、所定の値を初期値としてセットする機能（例えば、8ビットロード（転送）命令、16ビットロード（転送）命令、スタック操作命令（16ビットロード命令の一種）、複合命令など）を少なくとも有してもよい。

#### 【1037】

このような構成とすれば、遊技制御プログラムの設計の自由度を高めることができ、安定した遊技制御を行うことができる場合がある。

30

#### 【1038】

また、演出制御を行う演出制御手段（例えば、第1副制御部400または第2副制御部500）を備え、前記CPUは、前記演出制御手段に搭載されず、遊技制御を行う遊技制御手段（例えば、主制御部300）にのみに搭載されていてもよい。また、演出制御を行う演出制御手段（例えば、第1副制御部400または第2副制御部500）と、払出制御を行う払出制御手段（例えば、払出制御部600）と、を備え、前記CPUは、前記演出制御手段に搭載されず、遊技制御を行う遊技制御手段（例えば、主制御部300）および前記払出制御手段のうち少なくとも一方に搭載されていてもよい。

#### 【1039】

このような構成とすれば、遊技制御プログラムの解析や改ざんを、より困難にすることができる場合がある。

40

#### 【1040】

また、前記CPUと前記ROMを少なくとも内蔵するマイクロプロセッサ（例えば、基本回路302）を備え、前記マイクロプロセッサは、少なくとも所定の出力端子（例えば、リセット出力端子XRSTO）を有し、前記所定の出力端子は、少なくとも所定のタイミング（例えば、ランダム延長機能の実行前）で立ち上がることが可能なものであり、前記マイクロプロセッサは、所定の条件が成立した場合（例えば、図115に示すタイムアウト時間が経過した場合）に、所定の信号（例えば、WDT起動信号（WDTタイムアウト信号）や、CPU内部の信号）を少なくとも発生させることが可能なウォッチドッグタイマ（例えば、WDT314、第一WDT314a、第二WDT314b）と、正当な指

50

定エリア以外の記憶領域のコードが実行されたことに基づいて、指定エリア外走行禁止信号を少なくとも発生させることが可能な指定エリア外走行禁止回路（例えば、指定エリア外走行禁止回路 3 1 4 2）と、を少なくとも有し、前記指定エリア外走行禁止信号が発生してから前記所定のタイミングの間に開始されるセキュリティチェック中に、前記ウォッチドッグタイマのカウンタが停止している期間が少なくとも含まれていてもよい。

#### 【1041】

このような構成とすれば、指定エリア外走行禁止信号に基づくリセット処理を行う際に、ウォッチドッグタイマのタイムアウトを制御プログラムによらずに（自動的に）停止させることができ、指定エリア外走行禁止信号に基づくリセット処理を迅速に行うことができる場合がある。

#### 【1042】

また、所定の周期で数値を変化させる乱数回路と、所定の抽出条件が成立したことに基づいて前記乱数回路から数値を抽出し、少なくとも該数値に基づいて遊技に関する抽選を行う処理を含む遊技制御処理を行うための CPU と、前記遊技制御処理の内容が予め記憶されている ROM と、一時的にデータを記憶するための RAM と、を含んで構成されたマイクロコンピュータを備え、前記 CPU は、電源が遮断される場合に実行される処理であって、前記 RAM の所定領域に当該処理が行われたことを示す所定のデータを記憶させる処理を含む電断時処理を行い、前記 CPU は、前記所定のデータが記憶されている場合には、前記 RAM 領域を初期化する初期化処理を行うことなく前記遊技制御処理を復帰させることが可能である一方、前記所定のデータが記憶されていない場合には、該初期化処理を行うことなく前記遊技制御処理を復帰させることが不可能であり、前記マイクロコンピュータは、電源が投入された場合に入力される起動信号が入力されてから前記遊技制御処理が開始されるまでの時間の長さをランダムに変動させる時間変動処理を行う時間変動用回路、および正当な指定エリア以外の記憶領域のコードが実行されたことに基づいて指定エリア外走行禁止信号を発生させる指定エリア外走行禁止回路を、含んで構成され、前記 CPU は、前記起動信号が入力されたことに基づいて実行されるシステムリセットの場合には、前記マイクロコンピュータによって、前記 ROM が記憶している前記遊技制御処理の内容に異常がないかをチェックする処理および前記時間変動処理を含むセキュリティチェックが行われた後に、前記遊技制御処理を開始し、前記 CPU は、前記指定エリア外走行禁止信号が発生したことに基づいて実行されるユーザリセットの場合には、前記マイクロコンピュータによって、前記セキュリティチェックが行われないことで、前記時間変動処理が行われず、前記遊技制御処理を開始し、前記システムリセットおよび前記ユーザリセットのいずれの場合であっても、前記 CPU によって前記電断処理が行われず、前記所定領域に前記所定のデータが記憶されないことを特徴とする遊技台としてもよい。

#### 【1043】

次に、図 131 を用いて、図 70 のステップ S 2101 における初期設定処理の詳細について説明する。同図は、図 70 のステップ S 2101 における初期設定処理のフローチャートである。

#### 【1044】

まず、最初に実行されるステップ S 2101 a では、初期設定 1 を行う。この初期設定 1 では、CPU 1304 のスタックポインタ（SP）へのスタック初期値の設定（仮設定）、割込マスクの設定、I/O 1310 の初期設定、RAM 1308 に記憶する各種変数の初期設定等を行う。

#### 【1045】

ステップ S 2101 b では、低電圧信号がオンであるか否か、すなわち、電圧監視回路 1330 が、主制御部 1300 に供給されている電源の電圧値が所定の値（本実施形態では 9 V）未満である場合に電圧が低下したことを示す低電圧信号を出力しているか否かを監視する。そして、低電圧信号がオンの場合（CPU 1304 が電源の遮断を検知した場合）には繰り返しこのステップ S 2101 b を実行し、低電圧信号がオフの場合（CPU



1304が電源の遮断を検知していない場合)にはステップS2101cに進む。なお、電源が投入された直後で未だ上記所定の値(9V)に達しない場合にも、供給電圧がその所定の値以上になるまでステップS2101bは繰り返し実行される。

【1046】

ステップS2101cでは、初期設定2を行う。この処理の詳細は、図89を用いて説明した内容と同様であるため、説明を省略する。

【1047】

ステップS2101dでは、設定キースイッチがオンであるか否か判定される。ここで、設定キースイッチとは、スロットマシン1100における遊技者の有利度(例えば1~6まで6種類の有利度)を設定するためのスイッチである。このスイッチがオンである場合にはステップS2101iに進み、そうでない場合にはステップS2101eに進む。

10

【1048】

ステップS2101eでは、RAM1308に記憶されているデータに異常があるか否か判定される。ここで判定されるデータは、スロットマシン1100の電源が落されるかあるいは上述の瞬断によってRAM1308に退避されたデータである(図71のステップS2221参照)。すなわち、RAM1308にデータが確実に退避されているか否かが、このステップS2101eで判定される。なお、この処理の詳細は、図88のステップS3007において説明した内容と同様である。このデータに異常がある場合にはステップS2101kに進み、異常がない場合にはステップS2101fに進む。

【1049】

ステップS2101fでは、強制RWMクリアがON状態か否かを判定する。具体的には、電源が投入され、RWMクリアボタン(不図示)が長押し(例えば、5秒以上の押下)されたことに基づき、強制RWMクリアのON状態とする。そして、強制RWMクリアがON状態の場合はステップS2101jに移行し、OFF状態の場合はステップS2101gに進む。

20

【1050】

ステップS2101gでは、RAM1308に記憶されたデータをCPU1304のレジスタに書き戻し、レジスタの状態を電断処理が実行される直前の状態に復帰させる処理が実行される。なお、この処理の詳細は、図88のステップS3009において説明した内容と同様である。この処理の後ステップS2101hに進む。

30

【1051】

ステップS2101hでは、WDT1314を起動させる処理を行う。ここでは、WDT1314の起動許可及び初期値の設定等を行う。なお、本実施形態では、WDT1314に、初期値として32.8msに相当する数値を設定する。その後、この初期設定処理を終了する。

【1052】

ステップS2101dにおいて、設定キースイッチがオンの場合に進むステップS2101iでは、設定キーの状態に従って設定値変更処理が実行される。その後、ステップS2101jに進む。

【1053】

ステップS2101jでは、RWMクリア処理を行う。この処理の詳細は、図88のステップS3103において説明した内容と同様である。この処理の後、ステップS2101hに進む。

40

【1054】

ステップS2101eにおいて、RAM1308のデータに異常があると判定された場合に進むステップS2101kでは、RWMエラー処理を行う。このRWMエラー処理では、使用スタックエリアを除く全てのRAM(RWM)1308の記憶領域をクリアする準備などを行った後に、無限ループ状態に移行する。なお、この状態からは、電源を入れなおした後、設定キースイッチを操作することで遊技が開始できるようになる。

【1055】

50

図132は、図75に示すタイマ回路311の構成を詳しく示したブロック図である。

【1056】

タイマ回路311は、8ビットのプログラマブルカウンタを3チャンネル内蔵している。すなわち、図132に示すように、3つのタイマ回路（タイマ回路0～2）が用意されている。各タイマ回路は、ダウンカウンタ回路であって、それぞれ独立した動作モードの設定が可能である。これら3つのタイマ回路の構成はいずれも同じであるため、ここではタイマ回路0を用いて説明する。

【1057】

タイマ回路311は、プリスケアラレジスタ3111、制御レジスタ3112、カウンタ設定レジスタ3113、およびカウンタレジスタ3114の4種類のレジスタと、プリスケアラ3115と、8ビットカウンタ3116を有する。

10

【1058】

このタイマ回路311は、プリスケアラレジスタ3111で選択されたカウントクロックを8ビットカウンタ3116のクロック源とする。プリスケアラレジスタ3111は、8ビットカウンタ3116を動作させるためのクロック数を選択する8ビットのレジスタである。このプリスケアラレジスタ3111に、初期値である「00H」を設定した場合には、8ビットカウンタ3116を動作させるためのカウントクロックは停止になる。「01H」を設定した場合には、カウントクロックはシステムクロック（SCLK）になる。また、カウントクロックは、「02H」を設定した場合にはシステムクロック／2になり、「03H」を設定した場合にはシステムクロック／3になり、・・・「FEH」を設定した場合にはシステムクロック／254になり、「FFH」を設定した場合にはシステムクロック／255になる。

20

【1059】

制御レジスタ3112は、割込みフラグ3110のモニタ、割込みの許可や禁止、割込みフラグ3110のクリア、8ビットカウンタ3116の各種動作条件を設定する8ビットのレジスタである。

【1060】

このタイマ回路311では、カウンタ設定レジスタ3113にカウント値を書き込むことによって、8ビットカウンタ3116のダウンカウントが開始される。カウンタ設定レジスタ3113は、8ビットカウンタ3116のカウント値を設定するレジスタである。なお、設定値に「00H」を書き込んだ場合は、8ビットカウンタ3116は動作しない。

30

【1061】

カウンタレジスタ3114は、CPUレジスタリード信号の入力タイミングに合わせて8ビットカウンタ3116の値を読み出すためのレジスタである。

【1062】

8ビットカウンタ3116がタイムアウトすると、タイムアウト信号が8ビットカウンタ3116からカウンタ回路312に出力される。なお、割込みが許可されている場合には、割込みフラグ3110から割込み信号が割込み制御回路3100に出力される。また、8ビットの制御レジスタの0ビット目に、「0」を設定しておく、ワンタイムモード（初期値）になり、タイムアウト発生後、8ビットカウンタ3116は動作を停止するが、「1」を設定しておく、リピートモードになり、タイムアウト発生後も、8ビットカウンタ3116は継続して動作を行い、タイムアウト信号が定期的に出力される。

40

【1063】

以上説明した、各レジスタの設定値は、内蔵ROM306のプログラムデータエリアに記憶されている。プログラムデータエリアに記憶されている各設定値は、図88に示す主制御部メイン処理における初期設定2（ステップS3005）において、プログラムデータエリアから呼び出され、各レジスタに設定されるか、あるいは主制御部メイン処理や図91に示す主制御部割込処理において呼び出され、各レジスタに設定される。

【1064】

50

図 1 3 2 に示すタイマ回路 3 1 1 では、プリスケールレジスタ 3 1 1 1 に、初期値の「0 0 H」を設定しておくか、あるいはカウンタ設定レジスタ 3 1 1 3 に「0 0 H」を書き込んでおけば、8 ビットカウンタ 3 1 1 6 はダウンカウントを開始せず、タイムアウト信号は出力されない。

#### 【1 0 6 5】

図 1 3 3 は、図 7 5 に示すカウンタ回路 3 1 2 の構成を詳しく示したブロック図である。

#### 【1 0 6 6】

カウンタ回路 3 1 2 は、8 ビットのプログラマブルカウンタを 4 チャンネル内蔵している。すなわち、図 1 3 3 に示すように、4 つのカウンタ回路（カウンタ回路 0 ～ 3）が用意されている。各カウンタ回路は、アップカウンタ回路であって、それぞれ独立した動作モードの設定が可能である。これら 4 つのカウンタ回路の構成はいずれも同じであるため、ここではカウンタ回路 0 を用いて説明する。

#### 【1 0 6 7】

カウンタ回路は、制御レジスタ 3 1 2 1、8 ビットカウンタ 3 1 2 2、カウンタ設定レジスタ 3 1 2 3、およびカウンタレジスタ 3 1 2 4 を有する。制御レジスタ 3 1 2 1 は、内部バスを流れる各種の信号を取得可能であり、図 1 3 3 には、制御レジスタ 3 1 2 1 が、8 種類の信号を取得可能であることが示されている。これら 8 種類の信号は、システムクロック信号（S C L K）、図 1 3 2 に示すタイマ回路 0 ～ 2 それぞれからのタイムアウト信号、C P U コアのメモリリクエスト信号（X M R E Q 信号）、C P U コアのライト信号（X W R 信号）、C P U コアの I O リクエスト信号（X I O R Q 信号）、および C P U コアの割込み要求応答信号（マスカブル割込みアクノリッジサイクル；X I N T A C K 信号）である。制御レジスタ 3 1 2 1 は、8 ビットカウンタ 3 1 2 2 を動作させるためのクロック源を選択する 8 ビットのレジスタである。すなわち、8 本の信号線から入力されるいずれの信号をクロック源にするかを選択するものである。また、制御レジスタ 3 1 2 1 は、8 ビットカウンタ 3 1 2 2 の許可や禁止を設定するレジスタでもある。

#### 【1 0 6 8】

図 1 3 4 は、制御レジスタ 3 1 2 1 を詳しく説明するための図である。

#### 【1 0 6 9】

図 1 3 4 に示すように、制御レジスタ 3 1 2 1 は 8 ビットのレジスタである。5 ～ 7 ビット目は「0」が設定されたままであり、使用されていない。

#### 【1 0 7 0】

4 ビット目は 8 ビットカウンタ 3 1 2 2 の許可や禁止を設定するビットである。この 4 ビット目に、初期値である「0」が設定されると 8 ビットカウンタ 3 1 2 2 は動作を停止し、「1」が設定されると 8 ビットカウンタ 3 1 2 2 は動作を開始する。

#### 【1 0 7 1】

3 ビット目は 8 ビットカウンタ 3 1 2 2 のカスケード接続を設定するビットである。詳細は後述するが、カウンタ回路 0 の 8 ビットカウンタ 3 1 2 2 と、カウンタ回路 1 の 8 ビットカウンタ 3 1 2 2 はカスケード接続が可能であり、カウンタ回路 2 の 8 ビットカウンタ 3 1 2 2 と、カウンタ回路 3 の 8 ビットカウンタ 3 1 2 2 もカスケード接続が可能である。すなわち、カウンタ回路 0 の 8 ビットカウンタ 3 1 2 2 と、カウンタ回路 1 の 8 ビットカウンタ 3 1 2 2 が第 1 の組になっており、カウンタ回路 2 の 8 ビットカウンタ 3 1 2 2 と、カウンタ回路 3 の 8 ビットカウンタ 3 1 2 2 が第 2 の組になっている。同一組内で、いずれか一方の制御レジスタ 3 1 2 1 の 3 ビット目に「1」が設定されると、カスケード接続が優先される。すなわち、他方の制御レジスタ 3 1 2 1 の 3 ビット目に「0」が設定されていても、カスケード接続が設定され、「1」は「0」より優先される設定値である。

#### 【1 0 7 2】

2 ～ 0 ビット目は、8 ビットカウンタ 3 1 2 2 のクロック源を設定するビットである。すなわち、カウンタ回路 3 1 2 の更新源になる信号を設定するためのビットである。この

10

20

30

40

50

2～0ビット目に、初期値の「000」が設定されるとシステムクロック信号（SCLK）が8ビットカウンタ3122のクロック源になり、「001」が設定されるとタイマ回路0からのタイムアウト信号がクロック源になり、「010」が設定されるとタイマ回路1からのタイムアウト信号がクロック源になり、「011」が設定されるとタイマ回路2からのタイムアウト信号がクロック源になる。システムクロック信号（SCLK）は、制御レジスタ3121に定期的に入力される信号である。また、タイマ回路311をリピートモードに設定しておく、タイムアウト信号が制御レジスタ3121に定期的に入力される。一方、メモリリクエスト信号（XMREQ信号）、ライト信号（XWR信号）、I/Oリクエスト信号（XIORQ信号）、割込み要求応答信号（XINTACK信号）は、制御レジスタ3121に不定期的に入力される信号である。2～0ビット目に、「100」が設定されるとライト信号（XWR信号）が8ビットカウンタ3122のクロック源になり、「101」が設定されるとメモリリクエスト信号（XMREQ信号）がクロック源になり、「110」が設定されるとI/Oリクエスト信号（XIORQ信号）がクロック源になり、「111」が設定されると割込み要求応答信号（XINTACK信号）がクロック源になる。このクロック源を設定する設定値は、内蔵ROM306のプログラムデータエリアに記憶されており、図88に示す主制御部メイン処理における初期設定2（ステップS3005）において、プログラムデータエリアから呼び出され、制御レジスタ3121に設定される。なお、基本回路の内部的には、内蔵RAM308への信号は、メモリリクエスト信号（XMREQ信号）とI/Oリクエスト信号（XIORQ信号）とのうち、メモリマップド方式を採用した場合には、メモリリクエスト信号（XMREQ信号）が用いられ、I/Oマップド方式を採用した場合にはI/Oリクエスト信号（XIORQ信号）が用いられる。以上説明したように、制御レジスタ3121の2ビット目が、定期的に入力される信号をクロック源とした（第2の条件とした）場合には「0」であり、不定期的に入力される信号をクロック源とした（第1の条件とした）場合には「1」であり、所定ビット目の一例に相当する。こうしておくことで、開発者や設定者は、制御レジスタ3121の2ビット目を見るだけで、クロック源に定期に入力される信号が用いられているか否かを確認することができる。なお、制御レジスタ3121は、上記構成に限定されず、例えば、7ビット目が空きビット、6ビット目が8ビットカウンタ3122の許可や禁止を設定するビット（0：動作停止、1：動作開始（初期値））、5ビット目がCPUコアのマスカブル割込みアクノリッジサイクルで更新するかを設定するビット（0：更新する、1：更新しない（初期値））、4ビット目がCPUコアのXIORQ信号で更新するかを設定するビット（0：更新する、1：更新しない（初期値））、3ビット目がCPUコアのXMREQ信号で更新するかを設定するビット（0：更新する、1：更新しない（初期値））、2ビット目がCPUコアのXWR信号で更新するかを設定するビット（0：更新する、1：更新しない（初期値））、1～0ビット目がどのタイマの信号で更新するかを設定するビット（00：タイマ回路0からのタイムアウト信号、01：タイマ回路1からのタイムアウト信号、10：タイマ回路2からのタイムアウト信号、11：未選択（初期値））、のように構成してもよく、5～0ビットに全て1が書き込まれた場合に更新源としてシステムクロック信号（SCLK）が設定されるようにしてもよく、カウンタ回路1の制御レジスタ3121の1～0ビット目が00の場合にはタイマ回路0からのタイムアウト信号に代えてカウンタ回路0の8ビットカウンタ3122のタイムアウト信号を更新源としてもよく、カウンタ回路3の制御レジスタ3121の1～0ビット目が00の場合にはタイマ回路2からのタイムアウト信号に代えてカウンタ回路0の8ビットカウンタ3122のタイムアウト信号を更新源としてもよい。また、カウンタ回路0はSCLK、カウンタ回路1はXWR信号、カウンタ回路2はXMREQ信号、カウンタ回路3はXIORQ信号を更新源として自動的に起動されるものであってもよく、更新範囲を0～255としてもよい（カウンタ設定レジスタ3123に初期値として255を設定してもよい）。この場合、電源立ち上げ時には、プログラムによって更新源やカウント値を設定するよりも前にカウント動作を行い、カウント動作中にプログラムによって更新源の設定（変更）やカウント値の設定（変更）を行うようにしてもよい。この場合、カウント動作中に更新

10

20

30

40

50

源やカウンタ値を変更することで不正者による有利な値の取得を困難にできる場合がある。なお、カウンタ回路 3 1 2 は、更新源としてシステムクロック信号 (S C L K) を設定した場合に、他の更新源 (X W R 信号、X M R E Q 信号、X I O R Q 信号、タイマ回路 0 のタイムアウト信号、タイマ回路 1 のタイムアウト信号、タイマ回路 2 のタイムアウト信号) を設定不可能であってもよい。換言すると、カウンタ回路は、複数の更新源のうちの第一の更新源を設定した場合に、第一の更新源を除く他の更新源を設定不可能であってもよい。

また、カウンタ回路 3 1 2 は、更新源として X W R 信号、X M R E Q 信号、X I O R Q 信号、タイマ回路 n のタイムアウト信号のうちの少なくとも 2 以上の更新源を設定可能であってもよい。換言すると、カウンタ回路は、複数の更新源のうちの第一の更新源を除く

10

#### 【1 0 7 3】

また、カウンタ設定レジスタ 3 1 2 3 は、アップカウンタ回路である 8 ビットカウンタ 3 1 2 2 のカウンタ値 (最大値) を設定する 8 ビットのレジスタである。本実施形態では、0 ~ 2 5 5 のカウンタ値を設定することができ、「2 5 5」を書き込んだ場合には、カウンタ値として最大値の 2 5 5 が設定される。この場合には 0 ~ 2 5 5 が所定の数値範囲になる。なお、設定値に「0 0」を設定した場合には、8 ビットカウンタの動作を停止するようにしてもよい。なお、各カウンタ回路は、ダウンカウンタ回路であってもよく、ダウンカウンタ回路である場合には、カウンタ設定レジスタ 3 1 2 3 は、最小値を設定する

20

#### 【1 0 7 4】

カウンタレジスタ 3 1 2 4 は、C P U レジスタリード信号の入力タイミングに合わせて 8 ビットカウンタ 3 1 2 2 の値を読み出すためのレジスタである。

#### 【1 0 7 5】

以上説明した、各レジスタの設定値は、内蔵 R O M 3 0 6 のプログラムデータエリアに記憶されている。プログラム管理エリアに記憶されている各設定値は、図 8 8 に示す主制御部メイン処理における初期設定 2 (ステップ S 3 0 0 5) において、プログラムデータエリアから呼び出され、各レジスタに設定されるか、あるいは主制御部メイン処理や図 9 1 に示す主制御部割込処理において呼び出され、各レジスタに設定される。カウンタ設定レジスタ 3 1 2 3 にカウンタ値が書き込まれると、8 ビットカウンタ 3 1 2 2 の値が「0 0 H」にクリアされ、その後、制御レジスタ 3 1 2 1 の 4 ビット目に「1」が書き込まれると、選択されたクロック源で 8 ビットカウンタ 3 1 2 2 のアップカウントが開始される。

30

#### 【1 0 7 6】

以上説明したカウンタ回路 3 1 2 は、数値更新手段の一例に相当する。なお、8 ビットカウンタ 3 1 1 6 の値を C P U レジスタリード信号の入力に応じて読み出す構成を示したが、これに限定されるものではなく、ハードラッチ乱数値レジスタとハードラッチ選択レジスタを備えてもよく、8 ビットカウンタ 3 1 1 6 の値をハードラッチ乱数値レジスタに常に入力するようにしてもよく、パラレル入力ポートを介して P I 0 ~ P I 5 信号がそれぞれ入力されるハードラッチ選択レジスタからのラッチ信号がハードラッチ乱数値レジスタに入力された場合に値をラッチ (保持) するようにしてもよく、値のラッチによるセット信号と値の取得によるクリア信号が入力されるフラグレジスタを備えてもよい。また、この場合、ハードラッチ選択レジスタは、外部端子入力による 8 ビットカウンタ 3 1 1 6 の値の取込み条件の設定 (0 : 値を読み込まないと次の値をラッチしない (初期値)、1 : 値を読み込まなくても次の値をラッチ)、どの外部端子入力で 8 ビットカウンタ 3 1 1 6 の値を取り込むかの設定 (P I 0 ~ P I 5 信号) を設定可能に構成してもよい。なお、他の構成において、乱数生成回路 3 1 8 や乱数回路 5 3 1 6 で乱数を生成することに代えてカウンタ回路 3 1 2 で乱数を生成するようにしてもよい。

40

#### 【1 0 7 7】

続いて、図 7 (a) に示すメモリマップ上における内蔵 R A M エリアや内蔵レジスタエ

50

リアの読み込みや書き込みタイミングについて説明する。

#### 【1078】

図135は、図7(a)に示すメモリマップ上における内蔵RAMエリアや内蔵レジスタエリアの読み込みや書き込みタイミングを説明するためのタイミングチャートである。

#### 【1079】

図135では、図の左から右に向かって時間(T)が経過していく(以降のタイミングチャートにおいても同じ)。この図135に示すタイミングチャートには、上から順にシステムクロック出力信号(CLKO)、16本の信号線によって入出力されるアドレス入出力信号(A0~A15)、メモリリクエスト信号(XMREQ)、読み込み時のリード信号(XRD)、読み込み時の内蔵RAMエリアや内蔵レジスタエリアからのデータ出力信号(D0~D7)、書き込み時のライト信号(XWR)、および書き込み時の内蔵RAMエリアや内蔵レジスタエリアへのデータ出力信号(D0~D7)が示されている。なお、破線はハイインピーダンス状態を表す。

#### 【1080】

アドレス入出力信号(A0~A15)は、図7(a)のメモリマップにおいて読み書きするアドレスを指定している。メモリリクエスト信号(XMREQ)がLレベルのときに読み書きが許可され、リード信号(XRD)がLレベルのときに読み込みが許可され、ライト信号(XWR)もLレベルのときに書き込みが許可される。したがって、メモリリクエスト信号(XMREQ)がLレベルかつリード信号(XRD)もLレベルのときに、アドレス入出力信号(A0~A15)で指定されたアドレスを読み込むことができ、メモリリクエスト信号(XMREQ)がLレベルかつライト信号(XWR)もLレベルのときに、アドレス入出力信号(A0~A15)で指定されたアドレスに書き込むことができるが、各信号では時間的ズレが生じている。しかも、クロック出力信号(CLKO)は定期的な信号であるのに対して、制御レジスタ3121に入力される、メモリリクエスト信号(XMREQ)やライト信号(XWR)は、プログラムに従って任意のタイミングで出力される不定期な信号であり、定期的なクロック出力信号(CLKO)に対して同期していないことがわかる。言い換えれば、メモリリクエスト信号(XMREQ)やライト信号(XWR)は、CPU304が内蔵メモリや内蔵レジスタとのやり取りのための命令(所定の命令)に基づいて不定期に出力する信号である。より詳細に説明すると、CPU304は、内蔵ROM306のデータの読み出し(例えば、LD A, (ROM\_\_ADD01)命令)でメモリリクエスト信号(XMREQ)を出力し、内蔵RAM308とのデータの読み出しや書き込み(例えば、LD (RAM\_\_ADD01), A命令)でメモリリクエスト信号(XMREQ)やライト信号(XWR)を出力する。上述のごとく、本実施形態では、これら不定期に発生するメモリリクエスト信号(XMREQ)やライト信号(XWR)をカウンタ回路312の更新ソースとして利用することができる。不定期に発生するメモリリクエスト信号(XMREQ)やライト信号(XWR)は所定の信号の一例に相当する。

#### 【1081】

続いて、図7(b)に示すI/Oマップ上における内蔵レジスタエリアの読み込みや書き込みタイミングについて説明する。

#### 【1082】

図136は、図7(b)に示すI/Oマップ上における内蔵レジスタエリアの読み込みや書き込みタイミングを説明するためのタイミングチャートである。

#### 【1083】

この図136に示すタイミングチャートでも、上から順に各信号が示されており、図135に示すメモリリクエスト信号(XMREQ)に代えてI/Oリクエスト信号(XIORQ)が示されている。なおここでも、破線はハイインピーダンス状態を表す。

#### 【1084】

また、アドレス入出力信号(A0~A15)は、図7(b)のメモリマップにおいて読み書きするアドレスを指定している。I/Oリクエスト信号(XIORQ)がLレベルのと

10

20

30

40

50

きに読み書きが許可され、リード信号（XRD）がLレベルのときに読み込みが許可され、ライト信号（XWR）もLレベルのときに書き込みが許可される。したがって、I/Oリクエスト信号（XIORQ）がLレベルかつリード信号（XRD）もLレベルのときに、アドレス入出力信号（A0～A15）で指定されたアドレスを読み込むことができ、I/Oリクエスト信号（XIORQ）がLレベルかつライト信号（XWR）もLレベルのときに、アドレス入出力信号（A0～A15）で指定されたアドレスに書き込むことができるが、ここでも各信号では時間的ズレが生じている。しかも、制御レジスタ3121に入力される、I/Oリクエスト信号（XIORQ）やライト信号（XWR）も、プログラムに従って任意のタイミングで出力される不定期な信号であり、定期的なクロック出力信号（CLKO）に対して同期していないことがわかる。言い換えれば、I/Oリクエスト信号（XIORQ）も、CPU304が内蔵メモリや内蔵レジスタとのやり取りのための命令（所定の命令）に基づいて不定期に出力する信号である。より詳細に説明すると、CPU304は、IN命令やOUT命令の際にI/Oリクエスト信号（XIORQ）を出力する。上述のごとく、本実施形態では、これら不定期に発生するI/Oリクエスト信号（XIORQ）も

10

カウンタ回路312の更新ソースとして利用することができる。不定期に発生するI/Oリクエスト信号（XIORQ）も所定の信号の一例に相当する。

#### 【1085】

図137は、図75に示すマイクロプロセッサ3000内部におけるカウンタ回路312への信号の流れを内部のみ示す図である。

20

#### 【1086】

この図137には、CPU304、マイクロプロセッサ3000に内蔵されたROM306（以下、単に内蔵ROM306と称する）、同じくマイクロプロセッサ3000に内蔵されたRAM308（以下、単に内蔵RAM308と称する）、タイマ回路311、カウンタ回路312、割込み制御回路3100、およびクロック回路320が内部バス3300を介して互いに接続されている様子が示されている。

#### 【1087】

また、メモリマップド方式を採用した場合には、メモリリクエスト信号（XMREQ信号）が、CPU304から内蔵ROM306や内蔵RAM308に送られ、カウンタ回路312にも取得されることが示され、I/Oマップド方式を採用した場合には、I/Oリクエスト信号（XIORQ信号）が、CPU304から内蔵RAM308に送られ、カウンタ回路312にも取得されることが示されている。ライト信号（XWR信号）は、CPU304から内蔵RAM308に送られ、カウンタ回路312にも取得されることが示されている。なお、本実施形態では、カウンタ回路312の8ビットカウンタ3122のクロック源としてリード信号（XRD）は選択できないが、カウンタ回路312はリード信号（XRD）も取得可能であり、そのリード信号（XRD）をクロック源として利用してもよい。さらに、割込み制御回路3100からCPU304に送られる割込み要求信号（XINT信号）を受けてCPU304から割込み制御回路3100に割込み要求応答信号（XINTACK信号）が出力される。割込み要求信号は、定期的に出力されるものもあれば不定期に出力されるもの（例えばハードウェア割込み）もあり、割込み要求応答信号（XINTACK信号）は、結局のところ不定期に出力される信号になり、所定の信号の一例に相当する。図137では、この割込み要求応答信号（XINTACK信号）がカウンタ回路312に取得されることも示されている。また、タイマ回路311に含まれるタイマ回路0～2それぞれからのタイムアウト信号は、カウンタ回路312に送られ、クロック回路320からはシステムクロック信号（CLK）がカウンタ回路312に送られる。不定期に発生するメモリリクエスト信号（XMREQ）やライト信号（XWR）やI/Oリクエスト信号（XIORQ）や割込み要求応答信号（XINTACK信号）を出力するCPU304は、信号出力手段としての一機能を担っている。また、内蔵ROM306や内蔵RAM308は、記憶手段の一例に相当する。

30

40

#### 【1088】

50

図 1 3 8 は、カウンタ回路 0 を不定期な更新とする例を示した図である。

#### 【1 0 8 9】

図 1 3 8 の上方には機能ブロック図が示され、下方にはソースコードが示されている（以下、図 1 4 2 まで同じ）。このソースコードでは、カウンタ回路 0 は P C C 0 で表され、カウンタ回路 1 は P C C 1 で表されている（以下、同じ）。カウンタ回路 0 のカウンタ設定レジスタ 3 1 2 3 にも、カウンタ回路 1 のカウンタ設定レジスタ 3 1 2 3 にも、カウント値として最大値の「2 5 5」が設定されている。また、カウンタ回路 0 の制御レジスタ 3 1 2 1 には「0 0 0 1 0 1 0 1」が設定されており、4 ビット目に動作開始を表す「1」が設定され、2～0 ビット目には、8 ビットカウンタ 3 1 2 2 のクロック源をメモリリクエスト信号（X M R E Q 信号）とする「1 0 1」が設定されている。一方、カウンタ回路 1 の制御レジスタ 3 1 2 1 には「0 0 0 1 0 0 0 1」が設定されており、4 ビット目に動作開始を表す「1」が設定され、2～0 ビット目には、8 ビットカウンタ 3 1 2 2 のクロック源をタイマ回路 0 からのタイムアウト信号とする「0 0 1」が設定されている。さらに、いずれの制御レジスタ 3 1 2 1 にも、3 ビット目にはカスケード接続未設定を表す「0」が設定されており、この例では、カウンタ回路 0 とカウンタ回路 1 はカスケード接続されていない。なお、図 1 3 8 に示すソースコードでは、タイマ回路 0 は P T C 0 で表されており、タイマ回路 0 のプリスケアラレジスタ 3 1 1 1 で、システムクロックを  $1/240$  に分周し、タイマ回路 0 の 8 ビットカウンタ 3 1 1 6 のクロック源にすることが設定されている。

#### 【1 0 9 0】

図 1 3 8 に示す例では、カウンタ回路 0 の 8 ビットカウンタ 3 1 2 2 が不定期なメモリリクエスト信号（X M R E Q 信号）で更新される。一方、カウンタ回路 1 の 8 ビットカウンタ 3 1 2 2 は定期的なタイマ回路 0 からのタイムアウト信号で更新される。

#### 【1 0 9 1】

各カウンタ回路 3 1 2 のカウンタレジスタ 3 1 2 4 は、C P U レジスタリード信号の入力タイミングに合わせて 8 ビットカウンタ 3 1 2 2 の値を読み出し、C P U 3 0 4 は、カウンタレジスタ 3 1 2 4 が読み出した値を内部バスを介して取得する。この例では、カウンタ回路 0 のカウント値を、特図変動時間決定用乱数値として用いる。

#### 【1 0 9 2】

図 1 3 9 は、カウンタ回路 0 をカウンタ回路 1 の更新ソースとする変形例 1 を示した図である。以下、図 1 3 8 に示す例との相違点を中心に説明し、図 1 3 8 に示す例と同じ点は説明を省略することがある。

#### 【1 0 9 3】

図 1 3 9 に示すカウンタ回路 0 の制御レジスタ 3 1 2 1 には「0 0 0 1 1 1 0 1」が設定されており、3 ビット目にはカスケード接続の設定を表す「1」が設定されている。一方、カウンタ回路 1 の制御レジスタ 3 1 2 1 には「0 0 0 1 0 0 0 1」が設定されており、3 ビット目にはカスケード接続未設定を表す「0」が設定されている。上述のごとく、カウンタ回路 0 とカウンタ回路 1 の組では、いずれか一方の制御レジスタ 3 1 2 1 の 3 ビット目に「1」が設定されていると、カスケード接続が設定される。したがって、図 1 3 9 に示すカウンタ回路 0 とカウンタ回路 1 はカスケード接続されている。制御レジスタ 3 1 2 1 の 3 ビット目に「1」が設定されたカウンタ回路 0 は、0 から 2 5 5 までカウントアップするカウンタ回路であり、カウント値が 2 5 5 に達し、さらにカウント値を 1 更新する場合には、カウンタ回路 1 に桁溢れ信号（特定信号）を送り、カウント値を 0 に戻す。カウンタ回路 1 は、定期的なタイマ回路 0 からのタイムアウト信号で更新されるが、カウンタ回路 0 からの桁溢れ信号によっても更新され、カウンタ回路 1 の更新ソースが複数となり、カウンタ回路 1 のランダム性が増す。しかも、カウンタ回路 0 は、不定期なメモリリクエスト信号（X M R E Q 信号）で更新されるカウンタ回路であるため、桁溢れ信号も不定期に送られてくる信号になる。なお、カウンタ回路 1 の更新ソースを不定期な更新ソース（例えば、X W R 信号等）にすれば、ランダム性をより高めることができる。この例では、カウンタ回路 1 のカウント値を、特図変動時間決定用乱数値として用いる。



## 【1094】

なお、この例では、「1」が設定された方のカウンタ回路から桁溢れ信号が出力されるが、これに限らず、桁溢れ信号を出力するカウンタ回路を予め定めておいてもよい。

## 【1095】

図140は、カウンタ回路0とカウンタ回路1で2バイトカウンタとして機能させる変形例2を示した図である。以下、図139に示す例との相違点を中心に説明し、図139に示す例と同じ点は説明を省略することがある。

## 【1096】

図140に示すカウンタ回路1の制御レジスタ3121には「00010010」が設定されており、2～0ビット目には、8ビットカウンタ3122のクロック源をタイマ回路1からのタイムアウト信号とする「010」が設定されている。したがって、図140に示すカウンタ回路1の8ビットカウンタ3116は、タイマ回路1からのタイムアウト信号が入力されると更新される。さらに、図140に示すソースコードでは、タイマ回路1はPTC1で表されており、タイマ回路1のプリスケアラレジスタ3111には、タイマ回路1の8ビットカウンタ3116を動作させるためのカウントクロックを停止させる「0」が設定されている。このため、タイマ回路1の8ビットカウンタ3116からはタイムアウト信号は出力されず、カウンタ回路1は、クロック源としてタイマ回路1からのタイムアウト信号が設定されてはいるものの、そのタイムアウト信号をクロック源にしては更新されない。しかしながら、図139に示す例と同じく、カウンタ回路1には、カウンタ回路0から桁溢れ信号が送られ、カウンタ回路1は、この桁溢れ信号によって更新される。すなわち、カウンタ回路1は、カウンタ回路0からの桁溢れ信号のみをクロック源（更新源）にする。このように、タイマ回路1のカウントクロックを停止することで、実質的にカウンタ回路1の更新ソースをカウンタ回路0の桁溢れのみとすることができ、カウンタ回路0とカウンタ回路1のカウンタ値を用いて2バイトカウンタとして利用することができる。この例では、カウンタ回路0とカウンタ回路1の組によるカウント値を、特図変動時間決定用乱数値として用いる。

## 【1097】

図141は、カウンタ回路0の更新ソースをカウンタ回路1の更新ソースとする変形例3を示した図である。以下、図139に示す例との相違点を中心に説明し、図139に示す例と同じ点は説明を省略することがある。

## 【1098】

図141に示すカウンタ回路0のカウンタ設定レジスタ3123には、カウント値として「0」が設定されており、カウンタ回路0の最大値は0になる。すなわち、カウンタ回路0では所定の数値範囲が1になる。この結果、メモリリクエスト信号（XMRREQ信号）がカウンタ回路0に入力されるたびに桁溢れ信号がカウンタ回路1に出力され、カウンタ回路0の更新ソースをカウンタ回路1の更新ソースとしても扱うことができ、カウンタ回路1のランダム性が増す。また、カウンタ回路1の更新ソースを不定期な信号（例えば、XWR信号等）にすることで、ランダム性がさらに増す。この例でも、カウンタ回路1のカウント値を、特図変動時間決定用乱数値として用いる。

## 【1099】

図142は、カウンタ回路0とカウンタ回路1を相互にカスケード接続する変形例4を示す図である。以下、図139に示す例との相違点を中心に説明し、図139に示す例と同じ点は説明を省略することがある。

## 【1100】

図142に示すカウンタ回路0の制御レジスタ3121にも「00011101」が設定されており、3ビット目にはカスケード接続の設定を表す「1」が設定されている。また、図142に示すカウンタ回路1の制御レジスタ3121には「00011110」が設定されており、3ビット目にはカスケード接続の設定を表す「1」が設定され、2～0ビット目には、不定期に入力されるライト信号（XWR信号）をクロック源とする「100」が設定されている。したがって、この例では、一組の組を構成する両カウンタ回路で

10

20

30

40

50

カスケード接続の設定が行われており、互いに桁溢れ信号の送受信を行う。すなわち、カウンタ回路0では、カウント値が255に達し、さらにカウント値を1更新する場合には、カウンタ回路1にカウンタ回路0の桁溢れ信号を送り、カウンタ回路1でも、カウント値が255に達し、さらにカウント値を1更新する場合には、カウンタ回路0にカウンタ回路1の桁溢れ信号を送る。この結果、カウンタ回路0とカウンタ回路1がそれぞれ更新ソースが複数になり、両カウンタ回路のランダム性が増す。この例では、カウンタ回路0のカウント値を普図変動時間決定用乱数値として用い、カウンタ回路1のカウント値を特図変動時間決定用乱数値として用いる。

#### 【1101】

以上説明したカウンタ回路312では、カウント値の更新にランダム性を持たせることができる。カウンタ回路312のカウント値は、普図変動時間決定用乱数値や特図変動時間決定用乱数値として用いられ、抽選処理のランダム性を高めることができる。

#### 【1102】

続いて、他の実施形態について説明する。以下、既に述べた実施形態との相違点を中心に説明し、既に述べた実施形態と同じ点は説明を省略することがある。

#### 【1103】

今まで説明した実施形態では、マイクロプロセッサ3000（基本回路）の内部における内蔵RAMエリアや内蔵レジスタエリアを対象にしていたが、本実施形態では、マイクロプロセッサ3000（基本回路）の外部における外部メモリや外部レジスタ、および外部の周辺機器や周辺回路を対象にする。

#### 【1104】

図143は、マイクロプロセッサ3000（基本回路）の外部における外部メモリの読み込みや書き込みチップセレクトタイミングを説明するためのタイミングチャートである。

#### 【1105】

この図143に示すタイミングチャートには、上から順にシステムクロック出力信号（CLKO）、8本の信号線によって入出力されるアドレス入出力信号（A0～A15）、メモリリクエスト信号（XMREQ）、読み込み時のリード信号（XRD）、読み込み時のチップセレクト信号（XCS0～XCS13）、読み込み時の外部メモリからのデータ入力信号（D0～D7）、書き込み時のライト信号（XWR）、書き込み時のチップセレクト信号（XCS0～XCS13）、および書き込み時の外部メモリへのデータ出力信号（D0～D7）が示されている。なお、破線はハイインピーダンス状態を表す。

#### 【1106】

読み込み時のチップセレクト信号（XCS0～XCS13）によって読み込むべきチップセットが選択され、選択されたチップセットにおける読み込むべきアドレスがアドレス入出力信号（A0～A15）によって指定される。また、書き込み時のチップセレクト信号（XCS0～XCS13）によって書き込むべきチップセットが選択され、選択されたチップセットにおける書き込むべきアドレスがアドレス入出力信号（A0～A15）によって指定される。また、メモリリクエスト信号（XMREQ）がLレベルのときに読み書きが許可され、読み込み時のリード信号（XRD）がLレベルのときに読み込みが許可され、書き込み時のライト信号（XWR）もLレベルのときに書き込みが許可される。したがって、メモリリクエスト信号（XMREQ）がLレベルかつリード信号（XRD）もLレベルのときに、選択されたチップセットから指定されたアドレスのデータを読み込むことができ、メモリリクエスト信号（XMREQ）がLレベルかつライト信号（XWR）もLレベルのときに、選択されたチップセットの指定されたアドレスにデータを書き込むことができるが、各信号では時間的ズレが生じている。しかも、クロック出力信号（CLKO）は定期的な信号であるのに対して、メモリリクエスト信号（XMREQ）やライト信号（XWR）は、プログラムに従って任意のタイミングで出力される不定期な信号、すなわち、外部メモリとのやり取りのために不定期に発生する信号であり、定期的なクロック出力信号（CLKO）に対して同期していないことがわかる。本実施形態でも、これら不

10

20

30

40

50

定期に発生するメモリリクエスト信号 (X M R E Q) やライト信号 (X W R) が制御レジスタ 3 1 2 1 に入力され、これら不定期の信号をカウンタ回路 3 1 2 の更新ソースとして利用することができる。

#### 【1 1 0 7】

続いて、マイクロプロセッサ 3 0 0 0 (基本回路) の外部の周辺機器や周辺回路 (外部 I / O) に対する入出力について説明する。

#### 【1 1 0 8】

図 1 4 4 は、外部 I / O の読み込みや書き込みチップセレクトタイミングを説明するためのタイミングチャートである。

#### 【1 1 0 9】

この図 1 4 4 に示すタイミングチャートでも、上から順に各信号が示されており、図 1 3 5 に示すメモリリクエスト信号 (X M R E Q) に代えて I O リクエスト信号 (X I O R Q) が示されている。なおここでも、破線はハイインピーダンス状態を表す。

#### 【1 1 1 0】

また、読み込み時のチップセレクト信号 (X C S 0 ~ X C S 1 3) によって読み込むべきチップセットが選択され、選択されたチップセットにおける読み込むべきアドレスがアドレス入出力信号 (A 0 ~ A 1 5) によって指定される。また、書き込み時のチップセレクト信号 (X C S 0 ~ X C S 1 3) によって書き込むべきチップセットが選択され、選択されたチップセットにおける書き込むべきアドレスがアドレス入出力信号 (A 0 ~ A 1 5) によって指定される。また、I O リクエスト信号 (X I O R Q) が L レベルのときに読み書きが許可され、読み込み時のリード信号 (X R D) が L レベルのときに読み込みが許可され、書き込み時のライト信号 (X W R) も L レベルのときに書き込みが許可される。したがって、I O リクエスト信号 (X I O R Q) が L レベルかつリード信号 (X R D) も L レベルのときに、選択されたチップセットから指定されたアドレスのデータを読み込むことができ、I O リクエスト信号 (X I O R Q) が L レベルかつライト信号 (X W R) も L レベルのときに、選択されたチップセットの指定されたアドレスにデータを書き込むことができるが、ここでも各信号では時間的ズレが生じている。しかも、クロック出力信号 (C L K O) は定期的な信号であるのに対して、I O リクエスト信号 (X I O R Q) やライト信号 (X W R) は、プログラムに従って任意のタイミングで出力される不定期な信号、すなわち、外部の周辺機器や周辺回路 (外部 I / O) とのやり取りのために不定期に発生する信号であり、定期的なクロック出力信号 (C L K O) に対して同期していないことがわかる。第 2 実施形態でも、これら不定期に発生する I O リクエスト信号 (X I O R Q) やライト信号 (X W R) が制御レジスタ 3 1 2 1 に入力され、これら不定期の信号をカウンタ回路 3 1 2 の更新ソースとして利用することができる

図 1 4 5 は、図 7 5 に示すマイクロプロセッサ 3 0 0 0 内部におけるカウンタ回路 3 1 2 への信号の流れを内部のみ示す図である。

#### 【1 1 1 1】

この図 1 4 5 には、C P U 3 0 4、タイマ回路 3 1 1、カウンタ回路 3 1 2、割込み制御回路 3 1 0 0、クロック回路 3 2 0、外部バス制御回路 3 1 1 0、およびアドレスデコーダ回路 3 1 0 3 が内部バス 3 3 0 0 を介して互いに接続されている様子が示されている。また、図 1 4 5 には、その他回路 4 0 0 0 も示されている。その他回路 4 0 0 0 には、マイクロプロセッサ 3 0 0 0 の外部の周辺機器や周辺回路 (例えば、図 4 に示す主制御部 3 0 0 の各駆動回路 3 2 4、3 2 6、3 3 0、3 3 4 等) が相当する。

#### 【1 1 1 2】

また、メモリリクエスト信号 (X M R E Q 信号) は、C P U 3 0 4 からその他回路 4 0 0 0 の内蔵 R O M や内蔵 R A M に送られ、カウンタ回路 3 1 2 にも取得されることが示され、I O リクエスト信号 (X I O R Q 信号) は、C P U 3 0 4 からその他回路 4 0 0 0 の内蔵レジスタに送られ、カウンタ回路 3 1 2 にも取得されることが示されている。また、チップセレクト信号 (X C S 0 ~ X C S 1 3) は、C P U 3 0 4 からその他回路 4 0 0 0 の内蔵 R O M や内蔵 R A M や内蔵レジスタに送られ、カウンタ回路 3 1 2 にも取得される

10

20

30

40

50

ことが示されている。ライト信号（XWR信号）は、CPU304からその他回路400の内蔵RAMや内蔵レジスタに送られ、カウンタ回路312にも取得されることが示されている。なお、本実施形態でも、カウンタ回路312の8ビットカウンタ3122のクロック源としては、先の実施形態と同じ信号が用いられ、リード信号（XRD）は選択できないが、カウンタ回路312はリード信号（XRD）も取得可能であり、そのリード信号（XRD）をクロック源として利用してもよい。さらに、不定期に出力される割込み要求応答信号（XINTACK信号）もカウンタ回路312に取得されることが示されている。また、タイマ回路311に含まれるタイマ回路0～2それぞれからのタイムアウト信号は、カウンタ回路312に送られ、クロック回路320からはシステムクロック信号（SCLK）がカウンタ回路312に送られる。

10

#### 【1113】

この第2実施形態においても、カウンタ回路312の更新にランダム性を持たせることができる。カウント回路312のカウント値は、普図変動時間決定用乱数値や特図変動時間決定用乱数値として用いられ、抽選処理のランダム性を高めることができる。

#### 【1114】

なお、以上説明した実施形態では、カウンタ回路312のカウント値を、普図変動時間決定用乱数値や特図変動時間決定用乱数値として用いているが、普図の当りや特図の大当りか否かを抽選する当否判定や、特図の図柄を決定する抽選に用いてもよい。すなわち、制御状態の変更に関係する抽選や、賞球の払出に關係する抽選に用いてもよい。あるいは、その他の演出に関する抽選処理（例えば、先読み予告の実行可否抽選）に用いてもよい。

20

#### 【1115】

＜主制御部タイマ割込処理＞の変形例

次に、図146を用いて、主制御部300のCPU304が実行する主制御部タイマ割込処理の変形例について説明する。なお、同図は主制御部タイマ割込処理の流れを示すフローチャートである。

#### 【1116】

主制御部300は、所定の周期（本実施形態では約4msに1回）でタイマ割込信号を発生するカウンタタイマ312を備えており、このタイマ割込信号を契機として主制御部タイマ割込処理を所定の周期で開始する。

30

#### 【1117】

ステップS4201では、タイマ割込開始処理を行う。このタイマ割込開始処理では、CPU304の各レジスタの値をスタック領域に一時的に退避する処理などを行う。

#### 【1118】

ステップS4203では、WDT314のカウント値が初期設定値（本実施形態では32.8ms）を超えてWDT割込が発生しないように（処理の異常を検出しないように）、WDTを定期的に（本実施形態では、主制御部タイマ割込の周期である約4msに1回）リスタートを行う。

#### 【1119】

ステップS4205では、入力ポート状態更新処理を行う。この入力ポート状態更新処理では、I/O310の入力ポートを介して、上述の前面枠扉開放センサや内枠開放センサや下皿満タンセンサ、各種の球検出センサを含む各種センサ320の検出信号を入力して検出信号の有無を監視し、RAM308に各種センサ320ごとに区画して設けた信号状態記憶領域に記憶する。球検出センサの検出信号を例にして説明すれば、前回のタイマ割込処理（約4ms前）で検出した各々の球検出センサの検出信号の有無の情報を、RAM308に各々の球検出センサごとに区画して設けた前回検出信号記憶領域から読み出し、この情報をRAM308に各々の球検出センサごとに区画して設けた前々回検出信号記憶領域に記憶し、前回のタイマ割込処理（約4ms前）で検出した各々の球検出センサの検出信号の有無の情報を、RAM308に各々の球検出センサごとに区画して設けた今回検出信号記憶領域から読み出し、この情報を上述の前回検出信号記憶領域に記憶する。ま

40

50

た、今回検出した各々の球検出センサの検出信号を、上述の今回検出信号記憶領域に記憶する。

#### 【1120】

また、ステップS4205では、上述の前々回検出信号記憶領域、前回検出信号記憶領域、および今回検出信号記憶領域の各記憶領域に記憶した各々の球検出センサの検出信号の有無の情報を比較し、各々の球検出センサにおける過去3回分の検出信号の有無の情報が入賞判定パターン情報と一致するか否かを判定する。一個の遊技球が一つの球検出センサを通過する間に、約4msという非常に短い間隔で起動を繰り返すこの主制御部タイマ割込処理は何回か起動する。このため、主制御部タイマ割込処理が起動する度に、上述のステップS4205では、同じ遊技球が同じ球検出センサを通過したことを表す検出信号を確認することになる。この結果、上述の前々回検出信号記憶領域、前回検出信号記憶領域、および今回検出信号記憶領域それぞれに、同じ遊技球が同じ球検出センサを通過したことを表す検出信号が記憶される。すなわち、遊技球が球検出センサを通過し始めたときには、前々回検出信号無し、前回検出信号有り、今回検出信号有りになる。本実施形態では、球検出センサの誤検出やノイズを考慮して、検出信号無しの後に検出信号が連続して2回記憶されている場合には、入賞があったと判定する。図4に示す主制御部300のROM306には、入賞判定パターン情報（本実施形態では、前々回検出信号無し、前回検出信号有り、今回検出信号有りであることを示す情報）が記憶されている。このステップS4205では、各々の球検出センサにおいて過去3回分の検出信号の有無の情報が、予め定めた入賞判定パターン情報（本実施形態では、前々回検出信号無し、前回検出信号有り、今回検出信号有りであることを示す情報）と一致した場合に、一般入賞口226、可変入賞口234、第1特図始動口230、および第2特図始動口232への入球、または普図始動口228の通過があったと判定する。すなわち、これらの入賞口226、234やこれらの始動口230、232、228への入賞があったと判定する。例えば、一般入賞口226への入球を検出する一般入賞口センサにおいて過去3回分の検出信号の有無の情報が上述の入賞判定パターン情報と一致した場合には、一般入賞口226へ入賞があったと判定し、以降の一般入賞口226への入賞に伴う処理を行うが、過去3回分の検出信号の有無の情報が上述の入賞判定パターン情報と一致しなかった場合には、以降の一般入賞口226への入賞に伴う処理を行わずに後続の処理に分岐する。なお、主制御部300のROM306には、入賞判定クリアパターン情報（本実施形態では、前々回検出信号有り、前回検出信号無し、今回検出信号無しであることを示す情報）が記憶されている。入賞が一度あったと判定した後は、各々の球検出センサにおいて過去3回分の検出信号の有無の情報が、その入賞判定クリアパターン情報に一致するまで入賞があったとは判定せず、入賞判定クリアパターン情報に一致すれば、次からは上記入賞判定パターン情報に一致するか否かの判定を行う。なお、本実施形態では、入賞口226、234や始動口230、232、228へ入賞があったか否かの判定として、前々回、前回および今回の検出信号が予め定められた入賞判定パターン情報と一致した否かを判定していたが、このような方法に限らず、例えば、前回および今回の検出信号が予め定められた入賞判定パターン情報（例えば、前回検出信号なし、今回検出信号あり（いわゆるアップエッジ））と一致したか否かを判定するように構成してもよい。すなわち、球検出センサからの検出信号の有無を監視し、この監視結果の履歴が予め定められたパターンと一致したか否かを判定するように構成すればよい。

#### 【1121】

ステップS4207およびステップS4209では、基本乱数初期値更新処理および基本乱数更新処理を行う。これらの基本乱数初期値更新処理および基本乱数更新処理では、上記ステップS115で行った初期値生成用乱数カウンタの値の更新を行い、次に主制御部300で使用する、普図当選乱数値を生成するためのカウンタと、特図1乱数値および特図2乱数値それぞれを加工するためのカウンタ（以下、特図当選乱数加工用カウンタ）を更新する。例えば、普図当選乱数値として取り得る数値範囲が0～100とすると、RAM308に設けた普図当選乱数値を生成するための乱数カウンタ記憶領域から値を取得

し、取得した値に1を加算してから元の乱数カウンタ記憶領域に記憶する。このとき、取得した値に1を加算した結果が101であれば0を元の乱数カウンタ記憶領域に記憶する。また、取得した値に1を加算した結果、乱数カウンタが一周していると判定した場合にはそれぞれの乱数カウンタに対応する初期値生成用乱数カウンタの値を取得し、乱数カウンタの記憶領域にセットする。例えば、0～100の数値範囲で変動する普図当選乱数値生成用の乱数カウンタから値を取得し、取得した値に1を加算した結果が、RAM308に設けた所定の初期値記憶領域に記憶している前回設定した初期値と等しい値（例えば7）である場合に、普図当選乱数値生成用の乱数カウンタに対応する初期値生成用乱数カウンタから値を初期値として取得し、普図当選乱数値生成用の乱数カウンタにセットすると共に、普図当選乱数値生成用の乱数カウンタが次に1周したことを判定するために、今回設定した初期値を上述の初期値記憶領域に記憶しておく。また、普図当選乱数値生成用の乱数カウンタが次に1周したことを判定するための上述の初期値記憶領域とは別に、特図乱数生成用の乱数カウンタが1周したことを判定するための初期値記憶領域をRAM308に設けている。なお、本実施形態では特図1の乱数値および特図2の乱数値を加工するためのカウンタを同一のカウンタとして設けたが、それぞれ異なるカウンタとしてもよい。

10

#### 【1122】

ステップS4211では、演出乱数更新処理を行う。この演出乱数更新処理では、主制御部300で使用する演出用乱数値を生成するための乱数カウンタを更新する。

#### 【1123】

ステップS4213では、タイマ更新処理を行う。このタイマ更新処理では、普通図柄表示装置210に図柄を変動・停止表示する時間を計時するための普図表示図柄更新タイマ、第1特別図柄表示装置212に図柄を変動・停止表示する時間を計時するための特図1表示図柄更新タイマ、第2特図表示装置214に図柄を変動・停止表示する時間を計時するための特図2表示図柄更新タイマ、所定の入賞演出時間、所定の開放時間、所定の閉鎖時間、所定の終了演出期間などを計時するためのタイマなどを含む各種タイマを更新する。

20

#### 【1124】

ステップS4215では、入賞口カウンタ更新処理を行う。この入賞口カウンタ更新処理では、入賞口226、234や始動口230、232、228に入賞があった場合に、RAM308に各入賞口ごと、あるいは各始動口ごとに設けた賞球数記憶領域の値を読み出し、1を加算して、元の賞球数記憶領域に設定する。

30

#### 【1125】

また、ステップS4217では、入賞受付処理を行う。この入賞受付処理では、第1特図始動口230、第2特図始動口232、普図始動口228および可変入賞口234への入賞があったか否かを判定する。ここでは、ステップS4205における入賞判定パターン情報と一致するか否かの判定結果を用いて判定する。第1特図始動口230へ入賞があった場合、且つRAM308に設けた対応する保留数記憶領域が満タンでない場合には、乱数生成回路318の特図1に対応するチャンネルの乱数をラッチするためのラッチ信号を出力する処理（具体的には、ソフトラッチレジスタ3186の特図1のチャンネルに対応する領域に所定の値を書き込む処理）を実行するとともに、このラッチ信号によってラッチされた乱数を取得するためのリード信号を出力する処理（具体的には、乱数レジスタ3188のうちの特図1のチャンネルに対応する乱数レジスタに対してリード信号を出力する処理）を実行してラッチされている乱数を取得して対応する乱数値記憶領域に格納し、さらに、特図当選乱数加工用カウンタから値を取得して対応する乱数値記憶領域に格納する。第2特図始動口232へ入賞があった場合も第1特図始動口へ入賞があった場合と同様の処理を実行する。なお、本実施形態では、乱数生成回路318から乱数を取得する方法として特図1および2に対応するチャンネルの乱数をラッチするためのラッチ信号を出力する処理を行った後にラッチした乱数を取得するリード信号を出力する方法を採用しているが、特図1および2に対応するチャンネルの乱数を直読みする処理（具体的には、

40

50

乱数レジスタ 3 1 8 8 の値を直接リードする処理) を実行して取得するようにしてもよい。続いて普図始動口 2 2 8 へ入賞があった場合、且つ R A M 3 0 8 に設けた対応する保留数記憶領域が満タンでない場合、普図当選乱数値生成用の乱数カウンタから値を普図当選乱数値として取得して対応する乱数値記憶領域に格納する。可変入賞口 2 3 4 へ入賞があった場合には、可変入賞口用の入賞記憶領域に、可変入賞口 2 3 4 に球が入球したことを示す情報を格納する。なお、本実施形態では、特図 1 および 2 のラッチされた乱数を取得するためのリード信号を出力する処理(乱数レジスタ 3 1 8 8 の値を直接リードする処理とは異なる)を主制御部タイマ割込み処理において実行するように構成しているが、この処理に関しては上述した主制御部メイン処理において実行するように構成してもよく、この場合は、後述する特図関連抽選処理を合わせて主制御部メイン処理において実行するように構成すればよく、特図 1 および 2 に対応するチャンネルの乱数値を直読みする処理を実行する場合においては、特図関連抽選処理のみを主制御部メイン処理において実行するように構成すればよい。

10

#### 【1 1 2 6】

ステップ S 4 2 1 9 では、払出要求数送信処理を行う。なお、払出制御部 6 0 0 に出力する出力予定情報および払出要求情報は、例えば 1 バイトで構成しており、ビット 7 にストロブ情報(オンの場合、データをセットしていることを示す)、ビット 6 に電源投入情報(オンの場合、電源投入後一回目のコマンド送信であることを示す)、ビット 4 ~ 5 に暗号化のための今回加工種別(0 ~ 3)、およびビット 0 ~ 3 に暗号化加工後の払出要求数を示すようにしている。

20

#### 【1 1 2 7】

ステップ S 4 2 2 1 では、普図状態更新処理を行う。この普図状態更新処理は、普図の状態に対応する複数の処理のうちの 1 つの処理を行う。例えば、普図変動表示の途中(上述する普図表示図柄更新タイマの値が 1 以上)における普図状態更新処理では、普通図柄表示装置 2 1 0 を構成する 7 セグメント L E D の点灯と消灯を繰り返す点灯・消灯駆動制御を行う。この制御を行うことで、普通図柄表示装置 2 1 0 は普図の変動表示(普図変動遊技)を行う。

#### 【1 1 2 8】

また、普図変動表示時間が経過したタイミング(普図表示図柄更新タイマの値が 1 から 0 になったタイミング)における普図状態更新処理では、当りフラグがオンの場合には、当り図柄の表示態様となるように普通図柄表示装置 2 1 0 を構成する 7 セグメント L E D の点灯・消灯駆動制御を行い、当りフラグがオフの場合には、外れ図柄の表示態様となるように普通図柄表示装置 2 1 0 を構成する 7 セグメント L E D の点灯・消灯駆動制御を行う。また、主制御部 3 0 0 の R A M 3 0 8 には、普図状態更新処理に限らず各種の処理において各種の設定を行う設定領域が用意されている。ここでは、上記点灯・消灯駆動制御を行うとともに、その設定領域に普図停止表示中であることを示す設定を行う。この制御を行うことで、普通図柄表示装置 2 1 0 は、当り図柄(図 5 (c) に示す普図 A)および外れ図柄(図 5 (c) に示す普図 B)いずれか一方の図柄の確定表示を行う。さらにその後、所定の停止表示期間(例えば 5 0 0 m 秒間)、その表示を維持するために R A M 3 0 8 に設けた普図停止時間管理用タイマの記憶領域に停止期間を示す情報を設定する。この設定により、確定表示された図柄が所定期間停止表示され、普図変動遊技の結果が遊技者に報知される。

30

40

#### 【1 1 2 9】

また、普図変動遊技の結果が当りであれば、後述するように、普図当りフラグがオンされる。この普図当りフラグがオンの場合には、所定の停止表示期間が終了したタイミング(普図停止時間管理用タイマの値が 1 から 0 になったタイミング)における普図状態更新処理では、R A M 3 0 8 の設定領域に普図作動中を設定するとともに、所定の開放期間(例えば 2 秒間)、第 2 特図始動口 2 3 2 の羽根部材 2 3 2 a の開閉駆動用のソレノイド(3 3 2)に、羽根部材 2 3 2 a を開放状態に保持する信号を出力するとともに、R A M 3 0 8 に設けた羽根開放時間管理用タイマの記憶領域に開放期間を示す情報を設定する。

50

## 【1130】

また、所定の開放期間が終了したタイミング（羽根開放時間管理用タイマの値が1から0になったタイミング）で開始する普図状態更新処理では、所定の閉鎖期間（例えば500m秒間）、羽根部材の開閉駆動用のソレノイド332に、羽根部材を閉鎖状態に保持する信号を出力するとともに、RAM308に設けた羽根閉鎖時間管理用タイマの記憶領域に閉鎖期間を示す情報を設定する。

## 【1131】

また、所定の閉鎖期間が終了したタイミング（羽根閉鎖時間管理用タイマの値が1から0になったタイミング）で開始する普図状態更新処理では、RAM308の設定領域に普図非作動中を設定する。さらに、普図変動遊技の結果が外れであれば、後述するように、  
10 普図外れフラグがオンされる。この普図外れフラグがオンの場合には、上述した所定の停止表示期間が終了したタイミング（普図停止時間管理用タイマの値が1から0になったタイミング）における普図状態更新処理でも、RAM308の設定領域に普図非作動中を設定する。普図非作動中の場合における普図状態更新処理では、何もせずに次のステップS4223に移行するようにしている。

## 【1132】

ステップS4223では、普図関連抽選処理を行う。この普図関連抽選処理では、普図変動遊技および第2特図始動口232の開閉制御を行っておらず（普図の状態が非作動中）、且つ、保留している普図変動遊技の数が1以上である場合に、上述の乱数値記憶領域に記憶している普図当選乱数値に基づいた乱数抽選により普図変動遊技の結果を当選とするか、不当選とするかを決定する当り判定をおこない、当選とする場合にはRAM308に設けた当りフラグにオンを設定する。不当選の場合には、当りフラグにオフを設定する。  
20 。30

## 【1133】

次いで、特図1および特図2それぞれについての特図状態更新処理を行うが、最初に、特図2についての特図状態更新処理（特図2状態更新処理）を行う（ステップS4225）。この特図2状態更新処理は、特図2の状態に応じて、次の8つの処理のうちの1つの処理を行う。例えば、特図2変動表示の途中（上述の特図2表示図柄更新タイマの値が1以上）における特図2状態更新処理では、第2特別図柄表示装置214を構成する7セグメントLEDの点灯と消灯を繰り返す点灯・消灯駆動制御を行う。この制御を行うことで、第2特別図柄表示装置214は特図2の変動表示（特図2変動遊技）を行う。また、コマンド設定送信処理（ステップS4233）で回転開始設定送信処理を実行させることを示す所定の送信情報を上述の送信情報記憶領域に追加記憶してから処理を終了する。  
40

## 【1134】

また、主制御部300のRAM308の当否判定結果記憶領域には、15R大当りフラグ、2R大当りフラグ、第1小当りフラグ、第2小当りフラグ、第1はずれフラグ、第2はずれフラグ、特図確率変動フラグ、および普図確率変動フラグそれぞれのフラグが用意されている。特図2変動表示時間が経過したタイミング（特図2表示図柄更新タイマの値が1から0になったタイミング）で開始する特図2状態更新処理では、15R大当りフラグはオン、特図確率変動フラグもオン、普図確率変動フラグもオンの場合には図5（a）に示す特図A、15R大当りフラグはオン、特図確率変動フラグはオフ、普図確率変動フラグはオンの場合には特図B、2R大当りフラグはオン、特図確率変動フラグもオン、普図確率変動フラグもオンの場合には特図C、2R大当りフラグはオン、特図確率変動フラ  
50



グはオフ、普図確率変動フラグはオンの場合には特図D、2R大当りフラグはオン、特図確率変動フラグもオン、普図確率変動フラグはオンの場合には特図E、2R大当りフラグはオン、特図確率変動フラグはオフ、普図確率変動フラグもオフの場合には特図F、第1小当りフラグがオンの場合には特図G、第2小当りフラグがオンの場合には特図H、第1はずれフラグがオンの場合には特図I、第2はずれフラグがオンの場合には特図Iそれぞれの態様となるように、第2特別図柄表示装置214を構成する7セグメントLEDの点灯・消灯駆動制御を行い、RAM308の設定領域に特図2停止表示中であることを表す設定を行う。この制御を行うことで、第2特別図柄表示装置214は、15R特別大当り図柄（特図A）、15R大当り図柄（特図B）、突然確変図柄（特図C）、突然時短図柄（特図D）、隠れ確変図柄（特図E）、突然通常図柄（特図F）、第1小当り図柄（特図G）、第2小当り図柄（特図H）、第1はずれ図柄（特図I）、および第2はずれ図柄（特図J）のいずれか一つの図柄の確定表示を行う。さらにその後、所定の停止表示期間（例えば500m秒間）その表示を維持するためにRAM308に設けた特図2停止時間管理用タイマの記憶領域に停止期間を示す情報を設定する。この設定により、確定表示された特図2が所定期間停止表示され、特図2変動遊技の結果が遊技者に報知される。また、RAM308に設けられた時短回数記憶部に記憶された時短回数が1以上であれば、その時短回数から1を減算し、減算結果が1から0となった場合は、特図確率変動中（詳細は後述）でなければ、時短フラグをオフする。さらに、大当り遊技中（特別遊技状態中）にも、時短フラグをオフする。

10

#### 【1135】

20

また、コマンド設定送信処理（ステップS4233）で回転停止設定送信処理を実行させることを示す所定の送信情報を上述の送信情報記憶領域に追加記憶するとともに、変動表示を停止する図柄が特図2であることを示す特図2識別情報を、後述するコマンドデータに含める情報としてRAM308に追加記憶してから処理を終了する。

#### 【1136】

また、特図2変動遊技の結果が大当りであれば、後述するように、大当りフラグがオンされる。この大当りフラグがオンの場合には、所定の停止表示期間が終了したタイミング（特図2停止時間管理用タイマの値が1から0になったタイミング）における特図2状態更新処理では、RAM308の設定領域に特図2作動中を設定するとともに、所定の入賞演出期間（例えば3秒間）すなわち装飾図柄表示装置208による大当りを開始することを遊技者に報知する画像を表示している期間待機するためにRAM308に設けた特図2待機時間管理用タイマの記憶領域に入賞演出期間を示す情報を設定する。また、コマンド設定送信処理（ステップS4233）で入賞演出設定送信処理を実行させることを示す所定の送信情報を上述の送信情報記憶領域に追加記憶する。

30

#### 【1137】

また、所定の入賞演出期間が終了したタイミング（特図2待機時間管理用タイマの値が1から0になったタイミング）で開始する特図2状態更新処理では、所定の開放期間（例えば29秒間、または可変入賞口234に所定球数（例えば10球）の遊技球の入賞を検出するまで）可変入賞口234の扉部材234aの開閉駆動用のソレノイド（332）に、扉部材234aを開放状態に保持する信号を出力するとともに、RAM308に設けた扉開放時間管理用タイマの記憶領域に開放期間を示す情報を設定する。また、コマンド設定送信処理（ステップS4233）で大入賞口開放設定送信処理を実行させることを示す所定の送信情報を上述の送信情報記憶領域に追加記憶する。

40

#### 【1138】

また、所定の開放期間が終了したタイミング（扉開放時間管理用タイマの値が1から0になったタイミング）で開始する特図2状態更新処理では、所定の閉鎖期間（例えば1.5秒間）可変入賞口234の扉部材234aの開閉駆動用のソレノイド（332）に、扉部材234aを閉鎖状態に保持する信号を出力するとともに、RAM308に設けた扉閉鎖時間管理用タイマの記憶領域に閉鎖期間を示す情報を設定する。また、コマンド設定送信処理（ステップS4233）で大入賞口閉鎖設定送信処理を実行させることを示す所定

50

の送信情報を上述の送信情報記憶領域に追加記憶する。

【1139】

また、この扉部材の開放・閉鎖制御を所定回数（本実施例では15ラウンドか2ラウンド）繰り返し、終了したタイミングで開始する特図2状態更新処理では、所定の終了演出期間（例えば3秒間）すなわち装飾図柄表示装置208による大当りを終了することを遊技者に報知する画像を表示している期間待機するように設定するためにRAM308に設けた演出待機時間管理用タイマの記憶領域に演出待機期間を示す情報を設定する。また、普図確率変動フラグがオンに設定されていれば、この大当り遊技の終了と同時に、RAM308に設けられた時短回数記憶部に時短回数100回をセットするとともに、RAM308に設けられた時短フラグをオンする。なお、その普図確率変動フラグがオフに設定されていれば、時短回数記憶部に時短回数をセットすることなく、また時短フラグをオンすることもない。ここにいう時短とは、特図変動遊技における大当りを終了してから、次の大当りを開始するまでの時間を短くするため、パチンコ機が遊技者にとって有利な状態になることをいう。この時短フラグがオンに設定されていると、普図高確率状態である。普図高確率状態では普図低確率状態に比べて、普図変動遊技に大当りする可能性が高い。また、普図高確率状態の方が、普図低確率状態に比べて普図変動遊技の変動時間および特図変動遊技の変動時間は短くなる。さらに、普図高確率状態では普図低確率状態に比べて、第2特別始動口232の一对の羽根部材232aの1回の開放における開放時間が長くなりやすい。加えて、普図高確率状態では普図低確率状態に比べて、一对の羽根部材232aは多く開きやすい。また、上述のごとく、時短フラグは、大当り遊技中（特別遊技状態中）にはオフに設定される。したがって、大当り遊技中には、普図低確率状態が維持される。これは、大当り遊技中に普図高確率状態であると、大当り遊技中に可変入賞口234に所定の個数、遊技球が入球するまでの間に第2特図始動口232に多くの遊技球が入球し、大当り中に獲得することができる遊技球の数が多くなってしまい射幸性が高まってしまいうという問題があり、これを解決するためのものである。

【1140】

さらに、コマンド設定送信処理（ステップS4233）で終了演出設定送信処理を実行させることを示す所定の送信情報を上述の送信情報記憶領域に追加記憶する。

【1141】

また、所定の終了演出期間が終了したタイミング（演出待機時間管理用タイマの値が1から0になったタイミング）で開始する特図2状態更新処理では、RAM308の設定領域に特図2非作動中を設定する。さらに、特図2変動遊技の結果が外れであれば、後述するように、はずれフラグがオンされる。このはずれフラグがオンの場合には、上述した所定の停止表示期間が終了したタイミング（特図2停止時間管理用タイマの値が1から0になったタイミング）における特図2状態更新処理でも、RAM308の設定領域に特図2非作動中を設定する。特図2非作動中の場合における特図2状態更新処理では、何もせずに次のステップS4227に移行するようにしている。

【1142】

続いて、特図1についての特図状態更新処理（特図1状態更新処理）を行う（ステップS4227）。この特図1状態更新処理では、特図1の状態に応じて、上述の特図2状態更新処理で説明した各処理を行う。この特図1状態更新処理で行う各処理は、上述の特図2状態更新処理で説明した内容の「特図2」を「特図1」と読み替えた処理と同一であるため、その説明は省略する。なお、特図2状態更新処理と特図1状態更新処理の順番は逆でもよいし、第1特図始動口230または第2特図始動口232への入賞順序（入球順序）にしたがって、入賞（入球）の早い特図に対応する特図状態更新処理を優先して実行してもよい。

【1143】

ステップS4225およびステップS4227における特図状態更新処理が終了すると、今度は、特図1および特図2それぞれについての特図関連抽選処理を行う。ここでも先に、特図2についての特図関連抽選処理（特図2関連抽選処理）を行い（ステップS42

29)、その後で、特図1についての特図関連抽選処理(特図1関連抽選処理)を行う(ステップS4229)。より具体的には、特図1および特図2それぞれにおいて、乱数生成回路318から取得した乱数値を、特図当選乱数加工用カウンタから取得した乱数値を用いて加工(例えば、乱数生成回路318から取得した乱数値に対して特図当選乱数加工用カウンタから取得した乱数値を加算する処理)を行い、加工された乱数値を用いて抽選を行う。なお、本実施例では、特図1および特図2それぞれにおいて、乱数生成回路318から取得した乱数値を特図当選乱数加工用カウンタから取得した乱数値を用いて加工するように構成しているが、このような加工を行うことなく、乱数生成回路から取得した乱数値を直に使用して特図関連抽選処理を行うように構成してもよい。すなわち、乱数生成回路318から取得した乱数値を用いた特図関連抽選処理を行うように構成すればよい。

これらの特図関連抽選処理についても、主制御部300が特図2関連抽選処理を特図1関連抽選処理よりも先に行うことで、特図2変動遊技の開始条件と、特図1変動遊技の開始条件が同時に成立した場合でも、特図2変動遊技が先に変動中となるため、特図1変動遊技は変動を開始しない。また、装飾図柄表示装置208による、特図変動遊技の大当たり判定の結果の報知は、第1副制御部400によって行われ、第2特図始動口232への入賞に基づく抽選の抽選結果の報知が、第1特図始動口230への入賞に基づく抽選の抽選結果の報知よりも優先して行われる。

10

#### 【1144】

ステップS4233では、コマンド設定送信処理を行う。詳細は後述するが、このコマンド設定送信処理では、各種のコマンドを第1副制御部400に送信する処理を行う。なお、第1副制御部400では、主制御部300から受信した出力予定情報に含まれるコマンド種別により、主制御部300における遊技制御の変化に応じた演出制御の決定が可能になるとともに、出力予定情報に含まれているコマンドデータの情報に基づいて、演出制御内容を決定することができるようになる。

20

#### 【1145】

ステップS4235では、外部出力信号設定処理を行う。この外部出力信号設定処理では、RAM308に記憶している遊技情報を、情報出力回路336を介してパチンコ機100とは別体の情報入力回路350に出力する。

#### 【1146】

ステップS4237では、デバイス監視処理を行う。このデバイス監視処理では、ステップS4205において信号状態記憶領域に記憶した各種センサの信号状態を読み出して、所定のエラーの有無、例えば前面枠扉開放エラーの有無または下皿満タンエラーの有無などを監視し、前面枠扉開放エラーまたは下皿満タンエラーを検出した場合に、第1副制御部400に送信すべき送信情報に、前面枠扉開放エラーの有無または下皿満タンエラーの有無を示すデバイス情報を設定する。また、各種ソレノイド332を駆動して第2特図始動口232や、可変入賞口234の開閉を制御したり、表示回路324、326、330を介して普通図柄表示装置210、第1特別図柄表示装置212、第2特別図柄表示装置214、各種状態表示部328などに出力する表示データを、I/O310の出力ポートに設定する。また、払出要求数送信処理(ステップS4219)で設定した出力予定情報を出力ポート(I/O310)を介して第1副制御部400に出力する。

30

40

#### 【1147】

ステップS4239では、低電圧信号がオンであるか否かを監視する。そして、低電圧信号がオンの場合(電源の遮断を検知した場合)にはステップS4243に進み、低電圧信号がオフの場合(電源の遮断を検知していない場合)にはステップS4241に進む。

#### 【1148】

ステップS4241では、タイマ割込終了処理を行う。このタイマ割込終了処理では、ステップS4201で一時的に退避した各レジスタの値を元の各レジスタに設定したり、割込許可の設定などを行い、その後、主制御部メイン処理に復帰する。

#### 【1149】

一方、ステップS4243では、復電時に電断時の状態に復帰するための特定の変数や

50

スタックポインタを復帰データとしてRAM 308の所定の領域に退避し、入出力ポートの初期化等の電断処理を行い、その後、主制御部メイン処理に復帰する。

#### 【1150】

＜第1副制御部400の処理＞

次に、図147を用いて、第1副制御部400の処理について説明する。なお、同図（a）は、第1副制御部400のCPU404が実行するメイン処理のフローチャートである。同図（b）は、第1副制御部400のストロブ割込み処理のフローチャートである。同図（c）は、第1副制御部400のタイマ変数更新割込処理のフローチャートである。同図（d）は、第1副制御部400の画像制御処理のフローチャートである。

#### 【1151】

まず、同図（a）のステップS4301では、各種の初期設定を行う。電源投入が行われると、まずS4301で初期化処理が実行される。この初期化処理では、入出力ポートの初期設定や、RAM 408内の記憶領域の初期化処理等を行う。

#### 【1152】

ステップS4303では、タイマ変数が10以上か否かを判定し、タイマ変数が10となるまでこの処理を繰り返し、タイマ変数が10以上となったときには、ステップS4305の処理に移行する。ステップS4305では、タイマ変数に0を代入する。

#### 【1153】

ステップS4307では、コマンド処理を行う。第1副制御部400のCPU404は、主制御部300からコマンドを受信したか否かを判別する。

#### 【1154】

ステップS4309では、演出制御処理を行う。例えば、ステップS4307で新たなコマンドがあった場合には、このコマンドに対応する演出データをROM 406から読み出す等の処理を行い、演出データの更新が必要な場合には演出データの更新処理を行う。

#### 【1155】

ステップS4311では、チャンスボタンの押下を検出していた場合、ステップS4309で更新した演出データをチャンスボタンの押下に応じた演出データに変更する処理を行う。ステップS4313では、ステップS4309で読み出した演出データの中にVDP 434への命令がある場合には、この命令をVDP 434に出力する（詳細は後述）。

#### 【1156】

ステップS4315では、ステップS4309で読み出した演出データの中に音源IC 416への命令がある場合には、この命令を音源IC 416に出力する。ステップS4317では、ステップS4309で読み出した演出データの中に各種ランプ418への命令がある場合には、この命令を駆動回路420に出力する。

#### 【1157】

ステップS4319では、ステップS4309で読み出した演出データの中に遮蔽装置246への命令がある場合には、この命令を駆動回路432に出力する。ステップS4321では、ステップS4309で読み出した演出データの中に第2副制御部500に送信する制御コマンドがある場合には、この制御コマンドを出力する設定を行い、ステップS4303へ戻る。

#### 【1158】

次に、同図（b）を用いて、第1副制御部400のコマンド受信割込処理について説明する。このコマンド受信割込処理は、第1副制御部400が、主制御部300が出力するストロブ信号を検出した場合に実行する処理である。コマンド受信割込処理のステップS4401では、主制御部300が出力したコマンドを未処理コマンドとしてRAM 408に設けたコマンド記憶領域に記憶する。

#### 【1159】

次に、同図（c）を用いて、第1副制御部400のCPU404によって実行する第1副制御部タイマ割込処理について説明する。第1副制御部400は、所定の周期（本実施例では2msに1回）でタイマ割込を発生するハードウェアタイマを備えており、このタ

10

20

30

40

50

イマ割込を契機として、タイマ割込処理を所定の周期で実行する。

【1160】

第1副制御部タイマ割込処理のステップS4501では、第1副制御部メイン処理におけるステップS4303において説明したRAM408のタイマ変数記憶領域の値に、1を加算して元のタイマ変数記憶領域に記憶する。従って、ステップS4303において、タイマ変数の値が10以上と判定されるのは20ms毎(2ms×10)となる。

【1161】

第1副制御部タイマ割込処理のステップS4503では、ステップS4319で設定された第2副制御部500への制御コマンドの送信や、演出用乱数値の更新処理等を行う。

【1162】

次に、同図(d)を用いて、第1副制御部400のメイン処理におけるステップS4313の画像制御処理について説明する。同図は、画像制御処理の流れを示すフローチャートを示した図である。

【1163】

ステップS4601では、画像データの転送指示を行う。ここでは、CPU404は、まず、VRAM436の表示領域Aと表示領域Bの描画領域の指定をスワップする。これにより、描画領域に指定されていない表示領域に記憶された1フレームの画像が装飾図柄表示装置208に表示される。次に、CPU404は、VDP434のアトリビュートレジスタに、位置情報等テーブルに基づいてROM座標(ROM406の転送元アドレス)、VRAM座標(VRAM436の転送先アドレス)などを設定した後、ROM406からVRAM436への画像データの転送開始を指示する命令を設定する。VDP434は、アトリビュートレジスタに設定された命令に基づいて画像データをROM406からVRAM436に転送する。その後、VDP436は、転送終了割込信号をCPU404に対して出力する。

【1164】

ステップS4603では、VDP434からの転送終了割込信号が入力されたか否かを判定し、転送終了割込信号が入力された場合はステップS4605に進み、そうでない場合は転送終了割込信号が入力されるのを待つ。ステップS4605では、演出シナリオ構成テーブルおよびアトリビュートデータなどに基づいて、パラメータ設定を行う。ここでは、CPU404は、ステップS4601でVRAM436に転送した画像データに基づいてVRAM436の表示領域AまたはBに表示画像を形成するために、表示画像を構成する画像データの情報(VRAM436の座標軸、画像サイズ、VRAM座標(配置座標)など)をVDP434に指示する。VDP434はアトリビュートレジスタに格納された命令に基づいてアトリビュートに従ったパラメータ設定を行う。

【1165】

ステップS4607では、描画指示を行う。この描画指示では、CPU404は、VDP434に画像の描画開始を指示する。VDP434は、CPU404の指示に従ってフレームバッファにおける画像描画を開始する。

【1166】

ステップS4609では、画像の描画終了に基づくVDP434からの生成終了割込み信号が入力されたか否かを判定し、生成終了割込み信号が入力された場合はステップS4611に進み、そうでない場合は生成終了割込み信号が入力されるのを待つ。ステップS4611では、RAM408の所定の領域に設定され、何シーンの画像を生成したかをカウントするシーン表示カウンタをインクリメント(+1)して処理を終了する。

【1167】

<第2副制御部500の処理>

次に、図148を用いて、第2副制御部500の処理について説明する。なお、同図(a)は、第2副制御部500のCPU504が実行するメイン処理のフローチャートである。同図(b)は、第2副制御部500のコマンド受信割込処理のフローチャートである。同図(c)は、第2副制御部500のタイマ割込処理のフローチャートである。

10

20

30

40

50

## 【1168】

まず、同図（a）のステップS4701では、各種の初期設定を行う。電源投入が行われると、まずS4701で初期化処理が実行される。この初期化処理では、入出力ポートの初期設定や、RAM508内の記憶領域の初期化処理等を行う。

## 【1169】

ステップS4703では、タイマ変数が10以上か否かを判定し、タイマ変数が10となるまでこの処理を繰り返し、タイマ変数が10以上となったときには、ステップS4705の処理に移行する。

## 【1170】

ステップS4705では、タイマ変数に0を代入する。ステップS4707では、コマンド処理を行う。第2副制御部500のCPU504は、第1副制御部400のCPU404からコマンドを受信したか否かを判別する。ステップS4709では、演出制御処理を行う。例えば、S4707で新たなコマンドがあった場合には、このコマンドに対応する演出データをROM506から読み出す等の処理を行い、演出データの更新が必要な場合には演出データの更新処理を行う。

10

## 【1171】

ステップS4711では、第1副制御部400からの遊技盤用ランプ532や遊技台枠用ランプ542への命令がある場合には、この命令をシリアル通信制御回路520に出力する。

## 【1172】

ステップS4713では、第1副制御部400からの演出可動体224への命令がある場合には、この命令を駆動回路516に出力し、S4703に戻る。次に、同図（b）を用いて、第2副制御部500のコマンド受信割込処理について説明する。このコマンド受信割込処理は、第2副制御部500が、第1副制御部400が出力するストローブ信号を検出した場合に実行する処理である。コマンド受信割込処理のステップS4801では、第1副制御部400が出力したコマンドを未処理コマンドとしてRAM508に設けたコマンド記憶領域に記憶する。

20

## 【1173】

次に、同図（c）を用いて、第2副制御部500のCPU504によって実行する第2副制御部タイマ割込処理について説明する。第2副制御部500は、所定の周期（本実施例では2msに1回）でタイマ割込を発生するハードウェアタイマを備えており、このタイマ割込を契機として、タイマ割込処理を所定の周期で実行する。

30

## 【1174】

第2副制御部タイマ割込処理のステップS4901では、第2副制御部メイン処理におけるステップS4703において説明したRAM508のタイマ変数記憶領域の値に、1を加算して元のタイマ変数記憶領域に記憶する。従って、ステップS4703において、タイマ変数の値が10以上と判定されるのは20ms毎（2ms×10）となる。第2副制御部タイマ割込処理のステップS4903では、演出用乱数値の更新処理等を行う。

## 【1175】

<コマンド設定送信処理>

40

次に、上述の主制御部タイマ割込処理におけるコマンド設定送信処理（ステップS4233）について説明する。図149は、コマンド設定送信処理の流れを示すフローチャートである。

## 【1176】

このコマンド設定送信処理のステップS4233aでは、大当たり中用コマンド送信処理を行う。詳細は後述するが、この大当たり中用コマンド送信処理では、大当たり遊技中に第1副制御部400に送信するコマンドの設定を行う。また、次のステップS4233bでは、その他のコマンド（例えば、基本コマンド、図柄変動開始コマンド、図柄変動停止コマンド、入賞演出開始コマンド、終了演出開始コマンド、復電コマンドなど）を設定するコマンド設定送信処理を行った後に、処理を終了する。

50

## 【1177】

## &lt;大当たり中用コマンド送信処理&gt;

次に、上述のコマンド設定送信処理における大当たり中用コマンド送信処理（ステップS4233a）について説明する。図150は、大当たり中用コマンド送信処理の流れを示すフローチャートである。

## 【1178】

この大当たり中用コマンド送信処理のステップS5101では、RAM308に設けたコマンド格納領域をクリア（初期化）する。また、ステップS5102では、大当たり遊技の進行状況（大当たり遊技の開始時、ラウンド開始時、ラウンド終了時、大当たり遊技の終了時など）や、大当たり時の特図の種類などに基づいて、ROM306に予め記憶している複数種類のコマンドデータの中から、特定のコマンドデータを抽出する。また、抽出したコマンドデータが記憶されている記憶領域の先頭番地を、RAM308に設けた転送元番地記憶領域に転送元番地として設定（記憶）する。なお、ここでは、転送元番地を一時的に記憶する記憶手段としてRAM308を適用しているが、レジスタなどの他の記憶手段を適用してもよい（以下に説明する転送先番地や転送数なども同様である）。

10

## 【1179】

次に、ステップS5103では、コマンド格納領域の先頭番地を、RAM308に設けた転送先番地記憶領域に転送先番地として設定（記憶）し、ステップS5104では、コマンドデータのバイト数を、RAM308に設けた転送数記憶領域に転送数として設定（記憶）する。

20

## 【1180】

ステップS5105では、ブロック転送処理を行う。詳細は後述するが、このブロック転送処理では、転送元番地に設定された記憶領域に記憶されたデータを、転送先番地に設定された記憶領域に転送する処理等を行う。ステップS5106では、コマンド格納領域に転送されたコマンドを第1副制御部400に送信した後に処理を終了する。なお、コマンドを第1副制御部400に送信する処理は、上述の主制御部メイン処理で実行してもよい。

## 【1181】

## &lt;ブロック転送処理&gt;

次に、上述の大当たり中用コマンド送信処理におけるブロック転送処理（ステップS5105）について説明する。図151は、ブロック転送処理の流れを示すフローチャートである。

30

## 【1182】

このブロック転送処理のステップS5201では、上述の転送数に応じて補正時間を設定する。詳細は、図156を用いて説明するが、例えば、転送数が1の場合には、転送数が1の場合に要するブロック転送処理の処理時間と、転送数が3の場合に要するブロック転送処理の処理時間との差を補正時間として設定し、転送数が2の場合には、転送数が2の場合に要するブロック転送処理の処理時間と、転送数が3の場合に要するブロック転送処理の処理時間との差を補正時間として設定する。

## 【1183】

ステップS5202では、コマンドデータのうち、上述の転送元番地が示す記憶領域に記憶されたデータを、コマンド格納領域のうち、上述の転送先番地が示す記憶領域に転送する。ステップS5203では、転送元番地を1つ加算し、ステップS5204では、転送先番地を1つ加算する。

40

## 【1184】

また、ステップS5205では、転送数から1を減算し、ステップS5206では、減算後の転送数が0であるか否かを判定し、該当する場合（コマンドデータの転送が全て完了した場合）にはステップS5207に進み、該当しない場合（コマンドデータの転送が残っている場合）にはステップS5202に戻ってブロック転送処理を継続する。ステップS5207では、ステップS5201で設定した補正時間が経過するのを待ってから処

50

理を終了する。

#### 【1185】

＜コマンドテーブル＞

次に、ROM306に予め記憶されるコマンドテーブルについて説明する。図152（a）～（d）は、大当り遊技中に参照されるコマンドテーブルの一例を示した図である。

#### 【1186】

これらのコマンドテーブルは、ROM306に予め記憶されており、固定長（この例では、1バイト長）の先頭番地と、固定長（この例では、1バイト長）のバイト数と、可変長のコマンドデータ（この例では、1バイト長のコマンド識別子と、可変長のコマンドパラメータ）で構成される。

10

#### 【1187】

先頭番地は、コマンドデータのうち、1バイト目のコマンドデータ（この例では、コマンド識別子）が記憶されるROM306の番地を示しており、例えば、同図（a）に示す大当り開始コマンド1の1バイト目のコマンドデータ（11H）は、ROM306の番地100に記憶されていることを示している。また、バイト数は、コマンドデータのデータ長を示しており、例えば、同図（a）に示す大当り開始コマンド1のコマンドデータのデータ長は2バイトであることを示している。

#### 【1188】

同図（a）は、大当り遊技の開始時に参照されるコマンドテーブルの一例である。このコマンドテーブルには、大当り遊技の開始時に送信される大当り開始コマンド1～6の2バイト長のコマンドデータが規定され、コマンドデータの1バイト目に記憶されたコマンド識別子（11H）は、大当り開始コマンドであることを示している。また、コマンドデータの2バイト目には、大当り図柄識別用情報が記憶され、それぞれ、01Hは特図A、02Hは特図B、03Hは特図C、04Hは特図D、05Hは特図E、06Hは特図Fであることを示している。

20

#### 【1189】

同図（b）は、大当り遊技のラウンド開始時に参照されるコマンドテーブルの一例である。このコマンドテーブルには、大当り遊技のラウンド開始時に送信されるラウンド開始コマンド1～29の3バイト長のコマンドデータが規定され、コマンドデータの1バイト目に記憶されたコマンド識別子（12H）は、ラウンド開始コマンドであることを示している。また、コマンドデータの2バイト目には、大当り図柄識別用情報が記憶され、それぞれ、01Hは特図A、02Hは特図B、03Hは特図C、04Hは特図D、05Hは特図E、06Hは特図Fであることを示している。また、コマンドデータの3バイト目には、ラウンド数識別用情報が記憶され、ラウンド数を16進数で示している。

30

#### 【1190】

同図（c）は、大当り遊技のラウンド終了時に参照されるコマンドテーブルの一例である。このコマンドテーブルには、大当り遊技のラウンド終了時に送信されるラウンド終了コマンド1～15の2バイト長のコマンドデータが規定され、コマンドデータの1バイト目に記憶されたコマンド識別子（13H）は、ラウンド終了コマンドであることを示している。また、コマンドデータの2バイト目には、ラウンド数識別用情報が記憶され、ラウンド数を16進数で示している。

40

#### 【1191】

同図（d）は、大当り遊技の終了時に参照されるコマンドテーブルの一例である。このコマンドテーブルには、大当り遊技の終了時に送信される大当り終了コマンド1の1バイト長のコマンドデータが規定され、コマンドデータの1バイト目に記憶されたコマンド識別子（14H）は、大当り終了コマンドであることを示している。なお、この大当り終了コマンドはコマンドパラメータが無いコマンドである。

#### 【1192】

＜本発明と従来技術との対比＞

次に、本発明と従来技術の違いについて説明する。図153は、本発明の特徴部分のみ

50



を抜き出して示したフローチャートである。

【1193】

同図に示すように、上述の大当り中用コマンド送信処理のうち、ステップS5104の処理（転送数に応じて補正時間を設定する処理）の直後から、ステップS5106の処理（コマンド送信処理）の直前までに実行されるブロック転送処理は、処理X、処理Y、処理Zの3つに分けて考えることができる。

【1194】

処理Xは、ステップS5201の処理（転送数に応じて補正時間を設定する処理）の直後から、ステップS5206の処理（転送数が0であるか否かを判定し、判定結果に応じてステップS5202またはS5207にジャンプする処理：以下、条件分岐処理と称する場合がある）の直前までに実行される処理であり、ステップS5202～S5205の複数の処理で構成される。処理Yは、ステップS5206の処理（条件分岐処理）の直前から、このステップS5206の処理（条件分岐処理）の直後までの処理、すなわち、ステップS5206の処理（条件分岐処理）で構成される。処理Zは、ステップS5206の処理（条件分岐処理）の直後から、ステップS5207の処理（補正時間待機処理）を少なくとも含み、ステップS5106の処理（コマンド送信処理）の直前までの処理で構成される。

【1195】

図154（a）、（b）は、本発明の特徴部分を上記図153とは異なる視点で示したフローチャートである。同図（a）に示すように、上述の大当り中用コマンド送信処理は、ステップS5301～S5305の処理に分けて考えることができる。具体的には、ステップS5301は、第1の処理の準備処理（条件設定処理）であり、例えば、上記図153のステップS5201の処理（転送数に応じて補正時間を設定する処理）などが該当する。

【1196】

また、ステップS5302は、第1の処理（条件分岐処理）であり、例えば、上記図153のステップS5206の処理（条件分岐処理）が該当する。ステップS5303は、第2の処理（第1の処理内容）であり、例えば、上記図153のステップS5202～S5205の複数の処理（処理Xのうち、ステップS5206の処理（条件分岐処理）の後に実行される処理X）が該当する。

【1197】

また、ステップS5304は、第2の処理（第2の処理内容）であり、例えば、上記図153のステップS5207の処理（補正時間待機処理）を少なくとも含んだ、ステップS5106の処理（コマンド送信処理）の直前までの処理（処理Z）が該当する。なお、このステップS5304の第2の処理（第2の処理内容）には、同図（b）に示すように、条件分岐処理から第3の処理に分岐（ジャンプ）だけの処理も含まれ、演算命令、入出力命令、ロード命令のように、レジスタ、メモリ、スタック、I/Oなどを操作する命令には限定されない。

【1198】

また、第1の処理と第2の処理は、複数の命令（例えば、所定の条件を判断する第1の命令、条件が成立した場合に処理Aを実行する第2の命令、条件が成立しなかった場合に処理Bを実行する第3の命令）で構成する場合に限定されず、単一の命令（例えば、所定の条件を判断し、条件が成立した場合に処理Aを実行し、条件が成立しなかった場合に処理Bを実行する命令）で実行可能に構成してもよい。このように、一つの命令で第1の処理と第2の処理を実行可能とすれば、第1の処理と第2の処理に対応するプログラムコード量を削減することができたり、第1の処理と第2の処理を実行するために必要なステート数を削減できる場合がある。

【1199】

さらに、単一の命令のバリエーションとして、例えば、所定の条件を判断し、条件が成立した場合に処理Bを実行し、条件が成立しなかった場合に処理Aを実行する命令を備え

10

20

30

40

50

れば、所定の条件に応じて命令を使い分けることができ、プログラムを簡素化することができる場合がある。また、所定の演算を行い、演算結果が第1の結果の場合に処理Aを実行し、演算結果が第2の結果の場合に処理Bを実行する命令や、所定の演算を行い、演算結果が第1の結果の場合に処理Bを実行し、演算結果が第2の結果の場合に処理Aを実行する命令を備えれば、コーディングの利便性が高まり、プログラムの開発期間を短縮化できる場合がある。また、所定の条件を判断し、条件が成立した場合に処理Aに分岐し、条件が成立しなかった場合に処理Bに分岐する命令や、所定の演算を行い、演算結果が第1の結果の場合に処理Aに分岐し、演算結果が第2の結果の場合に処理Bに分岐する命令を備えてもよい。

#### 【1200】

また、ステップS5305は、第3の処理であり、例えば、上記図153のステップS5106の処理（コマンド送信処理）が該当する。なお、第3の処理の処理内容は特に限定されず、上述の第2の処理（第1の処理内容）および第2の処理（第2の処理内容）の直後に実行される処理であればよい。

#### 【1201】

一方、図155は、本発明の特徴部分に対応する従来技術の処理の流れを示したフローチャートであり、上記図153に対応するフローチャートである。この従来技術の処理を上記図153のフローチャートと同様に考えると、上述の大当り中コマンド送信処理に相当する処理のうち、ステップS5401の処理（転送数としてコマンドのバイト数を設定する処理）の直後から、ステップS5407の処理（コマンド送信処理）の直前までに実行される処理は、処理X'、処理Y'、処理Z'の3つに分けて考えることができる。

#### 【1202】

処理X'は、ステップS5401の処理（転送数としてコマンドのバイト数を設定する処理）の直後から、ステップS5406の処理（転送数が0であるか否かを判定し、判定結果に応じてステップS5402またはS5407にジャンプする処理：以下、条件分岐処理と称する場合がある）の直前までに実行される処理であり、ステップS5402～S5405の複数の処理で構成される。処理Yは、ステップS5406の処理（条件分岐処理）の直前から、このステップS5406の処理（条件分岐処理）の直後までの処理、すなわち、ステップS5406の処理（条件分岐処理）で構成される。処理Zは、ステップS5406の処理（条件分岐処理）の直後から、ステップS5407の処理（コマンド送信処理）の直前までの処理で構成される。

#### 【1203】

図156（a）は、本発明に係る大当り中用コマンド送信処理においてブロック転送を行う場合の所要時間を示した図であり、同図（b）は、従来技術に係る大当り中用コマンド送信処理においてブロック転送を行う場合の所要時間を示した図である。

#### 【1204】

同図（b）に示すように、従来技術に係る大当り中用コマンド送信処理では、ステップS5401において転送バイト数として1を設定した場合、第1の処理（処理Y'）のみを行うため、所要時間の合計は処理Y'（1回）の処理時間T1となる。また、ステップS5401において転送バイト数として2を設定した場合、第1の処理（処理Y'）を2回と、第2の処理（処理X'のうち、ステップS5406の処理（条件分岐処理）の後に実行される処理X'）を1回行うため、所要時間の合計は処理Y'（2回）の処理時間T1×2と処理X'（1回）の処理時間T2を合算した処理時間T1×2+T2となる。また、ステップS5401において転送バイト数として3を設定した場合、第1の処理（処理Y'）を3回と、第2の処理（処理X'のうち、ステップS5406の処理（条件分岐処理）の後に実行される処理X'）を2回行うため、所要時間の合計は処理Y'（1回）の処理時間T1×3と処理X'（2回）の処理時間T2×2を合算した処理時間T1×3+T2×2となる。

#### 【1205】

すなわち、従来技術に係る大当り中用コマンド送信処理では、転送バイト数によって処

10

20

30

40

50

理時間が異なっている。このため、転送バイト数の違いによる処理時間の差を考慮した作業をしなければならず、作業が煩雑になり開発期間の短縮化が極めて困難である。

#### 【1206】

一方、本発明に係る大当り中用コマンド送信処理では、ステップS5104において転送バイト数として1を設定した場合、第1の処理（処理Y）と、第2の処理（処理Zに含まれる補正時間待機処理）を行うため、所要時間の合計は処理Y（1回）の処理時間T3と、処理Zに含まれる補正時間待機処理の処理時間 $T3 \times 2 + T4 \times 2$ （2バイト分）を合算した処理時間 $T3 \times 3 + T4 \times 2$ となる。また、ステップS5104において転送バイト数として2を設定した場合、第1の処理（処理Y）を2回と、第2の処理（処理Xのうち、ステップS5206の処理（条件分岐処理）の後に実行される処理X）を1回と、第2の処理（処理Zに含まれる補正時間待機処理）を行うため、所要時間の合計は処理Y（1回）の処理時間 $T3 \times 2$ と、処理X（1回）の処理時間T4と、処理Zに含まれる補正時間待機処理の処理時間 $T3 + T4$ （1バイト分）を合算した処理時間 $T3 \times 3 + T4 \times 2$ となる。また、ステップS5104において転送バイト数として3を設定した場合、第1の処理（処理Y）を3回と、第2の処理（処理Xのうち、ステップS1206の処理（条件分岐処理）の後に実行される処理X）を2回行うため、所要時間の合計は処理Y（1回）の処理時間 $T3 \times 3$ と、処理X（2回）の処理時間 $T4 \times 2$ を合算した処理時間 $T3 \times 3 + T4 \times 2$ となる。

10

#### 【1207】

すなわち、本発明に係る大当り中用コマンド送信処理では、転送バイト数に関わらず処理時間が同一（ $T3 \times 3 + T4 \times 2$ ）となっている。このため、上述の従来技術の課題を解決することが可能であり、具体的には、転送バイト数の違いによる処理時間の差がなくなり、デバックやテストで確認しなければならないパターンが大幅に削減できるため、デバックやテストの作業時間を短縮することができる上に、処理時間の差を考慮する必要がなくなるため、プログラム設計を簡素化することができる。

20

#### 【1208】

また、第1の処理と第2の処理に続く第3の処理の開始タイミング（第1の処理を開始する直前のタイミングを基準として、第3の処理が開始されるタイミング）を転送バイト数に関わらず同一にすることができるため、転送バイト数の違いによる第3の処理の開始タイミングの差がなくなり、デバックやテストで確認しなければならないパターンが大幅に削減できるため、デバックやテストの作業時間を短縮することができる上に、開始タイミングの差を考慮する必要がなくなるため、プログラム設計を簡素化することができる。

30

#### 【1209】

なお、上記図153および154では、補正時間待機処理を第2の処理に含める例を示したが、補正時間待機処理は、第1の処理準備処理（条件設定処理）の直後から第3の処理の直前までに実行すればよく、図157（a）、（b）に示すように、補正時間待機処理を、補正時間待機処理配置位置1～7で示される位置のいずれかに配置してもよい。

#### 【1210】

< 割込み処理の処理時間 >

次に、割込み処理の処理時間について説明する。図158（a）、（b）は割込み処理の流れを模式的に示した図である。割込み処理が複数の処理（例えば、処理1、処理2、処理3、・・・、処理X）で構成されている場合、各々の処理の処理時間は割込みタイミングによって異なるため、例えば、同図（a）に示す例では、割込みタイミング1に比べて、割込みタイミング2における処理1、処理2、処理3の処理時間が短くなっている結果、これらの処理の後に実行される処理Xの実行開始タイミングが、割込みタイミング1と割込みタイミング2で異なっている。また、同図（b）に示す例では、割込みタイミング1に比べて、割込みタイミング2における処理2の処理時間が短くなっている結果、これらの処理の後に実行される処理Xの実行開始タイミングが、割込みタイミング1と割込みタイミング2で異なっている。

40

#### 【1211】

50

このような割込み処理（例えば、上述の主制御部タイマ割込処理）を設計する場合、この割込み処理によって割り込まれる処理（例えば、上述の主制御部メイン処理）のリアルタイム性を確保するために、割込み処理の最大処理時間を考慮する必要がある。例えば、同図（a）、（b）の上段に示す割込みタイミング1が最大処理時間を示していると仮定すると、他の割込みタイミングでの処理が全て同図（a）、（b）の下段に示す割込みタイミング2のように、割込みタイミング1の最大処理時間以内に収まるかどうか、さらには、処理Xが同一のタイミングで実行開始できるかどうかなどを検証しなければならない。

#### 【1212】

上述のとおり、従来技術に係る大当り中用コマンド送信処理では、転送バイト数によって処理時間が異なっている。このため、このような処理を割込み処理に含めた場合には、デバッグやテストにおいて転送バイト数の違いによる処理時間の差を考慮した作業をしなければならず、作業が煩雑になり開発期間の短縮化が極めて困難である。

#### 【1213】

一方、本発明に係る大当り中用コマンド送信処理では、転送バイト数に関わらず処理時間が同一となっている。このため、このような処理を割込み処理に含めた場合でも、当該処理については処理時間の差を考慮する必要が無く、デバックやテストで確認しなければならないパターンが大幅に削減できるため、デバックやテストの作業時間を短縮することができる上に、処理時間の差を考慮する必要がなくなるため、プログラム設計を簡素化することができる。

#### 【1214】

さらに、タイマ割込み処理は、通常、予め定めた所定の周期で実行できるように設計されるが、予期しない事象が発生した場合（例えば、割込み処理に含まれる処理の一部に想定以上の処理時間がかかってしまった場合や、優先順位の高い割込み処理によって想定以上の待ち時間が発生してしまった場合など）には、いわゆる揺らぎ（ジッター）が生じてしまうことがある。このような揺らぎが発生した場合、所定の周期で実行されるべき処理が実行できない事態が生じうるが、本発明によれば、割込み処理内の処理時間の統一を図ることができるため、このような時間の揺らぎを無くし、処理の実行開始タイミングを同一にすることができる場合がある。

#### 【1215】

特に、パチンコ機100においては、上記ステップS4205の入力ポート状態更新処理（球検出センサの検出信号を入力して検出信号の有無を監視する処理）や、スロットマシン1100においては、ステップS2205の入力ポート状態更新処理（各種センサ1318のセンサ回路1320の検出信号を入力して検出信号の有無を監視する処理）や、ステップS2211のコマンド設定送信処理（各種モータに関する制御データを他の制御部に送信する処理）などの実行開始タイミングを割込みタイミングによらずに同一にすることが可能となるため、遊技制御の安定化を図ることができ、遊技の公平性を担保できる場合がある。

#### 【1216】

＜コマンド設定送信処理＞

次に、上述の図71に示す主制御部タイマ割込処理におけるコマンド設定送信処理（ステップS2211）について説明する。図159は、コマンド設定送信処理の流れを示すフローチャートである。

#### 【1217】

このコマンド設定送信処理のステップS2211aでは、RAM1308に設けたコマンド格納領域をクリア（初期化）する。また、ステップS2211bでは、遊技の進行状況（遊技開始時、レバー受付時、左リール停止時、中リール停止時、右リール停止時、全リール停止時ど）などに基づいて、ROM1306に予め記憶している複数種類のコマンドデータの中から、特定のコマンドデータを抽出する。また、抽出したコマンドデータが記憶されている記憶領域の先頭番地を、RAM1308に設けた転送元番地記憶領域に転

10

20

30

40

50

送元番地として設定（記憶）する。なお、ここでは、転送元番地を一時的に記憶する記憶手段としてRAM 1308を適用しているが、レジスタなどの他の記憶手段を適用してもよい（以下に説明する転送先番地や転送数なども同様である）。

#### 【1218】

次に、ステップS2211cでは、コマンド格納領域の先頭番地を、RAM 1308に設けた転送先番地記憶領域に転送先番地として設定（記憶）し、ステップS2211dでは、コマンドデータのバイト数を、RAM 1308に設けた転送数記憶領域に転送数として設定（記憶）する。

#### 【1219】

ステップS2211eでは、ブロック転送処理を行う。なお、このブロック転送処理は、上記図151を用いて説明したブロック転送処理と同一であるため説明は省略する。ステップS2211fでは、コマンド格納領域に転送されたコマンドを第1副制御部1400に送信した後に処理を終了する。なお、コマンドを第1副制御部1400に送信する処理は、上述の主制御部メイン処理で実行してもよい。

#### 【1220】

＜コマンドテーブル＞

次に、ROM 1306に予め記憶されるコマンドテーブルについて説明する。図160(a)～(d)は、遊技中に参照されるコマンドテーブルの一例を示した図である。

#### 【1221】

これらコマンドテーブルは、ROM 1306に予め記憶されており、固定長（この例では、1バイト長）の先頭番地と、固定長（この例では、1バイト長）のバイト数と、可変長のコマンドデータ（この例では、1バイト長のコマンド識別子と、可変長のコマンドパラメータ）で構成される。

#### 【1222】

先頭番地は、コマンドデータのうち、1バイト目のコマンドデータ（この例では、コマンド識別子）が記憶されるROM 1306の番地を示しており、例えば、同図(a)に示す遊技開始コマンド1の1バイト目のコマンドデータ(A1H)は、ROM 1306の番地1000に記憶されていることを示している。また、バイト数は、コマンドデータのデータ長を示しており、例えば、同図(a)に示す遊技開始コマンド1のコマンドデータのデータ長は2バイトであることを示している。

#### 【1223】

同図(a)は、遊技開始時に参照されるコマンドテーブルの一例である。このコマンドテーブルには、遊技開始時に送信される遊技開始コマンド1～2の2バイト長のコマンドデータが規定され、コマンドデータの1バイト目に記憶されたコマンド識別子(A1H)は、遊技開始コマンドであることを示している。また、コマンドデータの2バイト目には、再遊技識別用情報が記憶され、それぞれ、01Hは再遊技作動あり、02Hは再遊技作動無しであることを示している。

#### 【1224】

同図(b)は、レバー（スタートレバー1135）の受付時に参照されるコマンドテーブルの一例である。このコマンドテーブルには、レバー受付時に送信されるレバー受付コマンド1～12の3バイト長のコマンドデータが規定され、コマンドデータの1バイト目に記憶されたコマンド識別子(A3H)は、レバー受付コマンドであることを示している。また、コマンドデータの2バイト目には、遊技状態識別用情報が記憶され、それぞれ、01Hは通常状態中、02HはBB中、03HはRB中、04Hはボーナス内部当選中であることを示している。また、コマンドデータの3バイト目には、当選役識別用情報が記憶され、それぞれ、01Hはハズレ、02Hは小役、03Hは再遊技役、04HはRB、05HはBBであることを示している。

#### 【1225】

同図(c)は、左リール110の停止時に参照されるコマンドテーブルの一例である。このコマンドテーブルには、左リール110の停止時に送信される左リール停止コマンド

10

20

30

40

50

1の1バイト長のコマンドデータが規定され、コマンドデータの1バイト目に記憶されたコマンド識別子(A4H)は、左リール停止コマンドであることを示している。なお、この左リール停止コマンドはコマンドパラメータが無いコマンドである。

#### 【1226】

同図(d)は、中リール111の停止時に参照されるコマンドテーブルの一例である。このコマンドテーブルには、中リール111の停止時に送信される中リール停止コマンド1の1バイト長のコマンドデータが規定され、コマンドデータの1バイト目に記憶されたコマンド識別子(A5H)は、中リール停止コマンドであることを示している。なお、この中リール停止コマンドはコマンドパラメータが無いコマンドである。

#### 【1227】

同図(e)は、右リール112の停止時に参照されるコマンドテーブルの一例である。このコマンドテーブルには、右リール112の停止時に送信される右リール停止コマンド1の1バイト長のコマンドデータが規定され、コマンドデータの1バイト目に記憶されたコマンド識別子(A6H)は、右リール停止コマンドであることを示している。なお、この右リール停止コマンドはコマンドパラメータが無いコマンドである。

#### 【1228】

同図(f)は、全リール停止時に参照されるコマンドテーブルの一例である。このコマンドテーブルには、全リール110~112の停止時に送信される全リール停止コマンド1~5の2バイト長のコマンドデータが規定され、コマンドデータの1バイト目に記憶されたコマンド識別子(A7H)は、全リール停止コマンドであることを示している。また、コマンドデータの2バイト目には、当選役識別用情報が記憶され、それぞれ、01Hはハズレ、02Hは小役、03Hは再遊技役、04HはRB、05HはBBであることを示している。

#### 【1229】

このようなスロットマシン1100の主制御部1300に対しても、次に説明する本発明の特徴点の一つ、複数、または全てを適用することができる。

#### 【1230】

以上説明したように、本実施形態に係るパチンコ機100(またはスロットマシン1100)は、遊技に関する複数の処理を実行する遊技制御手段(例えば、主制御部300(または主制御部1300))を備えた遊技台であって、前記遊技制御手段は、第一処理(例えば、図154(a)の第1の処理)、該第一処理の要求に応じて処理の内容が第一の内容および第二の内容のうち一方の内容となる第二処理(例えば、図154(a)の第2の処理)、第三処理(例えば、図154(a)の第3の処理)の順に処理を実行し、前記第二処理の内容が前記第一の内容(例えば、図154(a)の第1の処理内容)および前記第二の内容(例えば、図154(a)の第2の処理内容)のうちいずれの場合であっても同一のタイミングで(例えば、図154(a)の第1の処理の準備処理(条件設定処理)の開始直前を基準として、第3の処理が開始されるタイミングが同一のタイミングで、または、図154(a)の第1の処理(条件分岐処理)の開始直前を基準として、第3の処理が開始されるタイミングが同一のタイミングで、など)前記第三処理の実行を開始することを特徴とする遊技台である。

#### 【1231】

本実施形態に係るパチンコ機100(またはスロットマシン1100)によれば、第2の処理の内容が違っていても第2の処理の処理時間の差をなくすことができ、デバックやテストで確認しなければならないパターンが大幅に削減できるため、デバックやテストの作業時間を短縮することができる上に、処理時間の差を考慮する必要がなくなるため、プログラム設計を簡素化することができる場合がある。また、遊技制御手段において処理状況に応じた分岐処理を行いつつも、遊技制御の安定化を図ることができる場合がある。

#### 【1232】

また、前記遊技制御手段は、前記第三処理として前記第二処理の内容に応じて第三の内容および第四の内容のうち一方の内容を実行し、前記第二処理において前記第一の内容を

10

20

30

40

50

実行した場合には、前記第三処理として前記第三の内容を実行し、前記第二処理において前記第二の内容を実行した場合には、前記第三処理として前記第四の内容を実行してもよい。

### 【1233】

このような構成とすれば、分岐処理の結果、異なる処理が実行される場合においても遊技制御の安定化を図ることができる場合がある。

なお、本実施形態に係るパチンコ機100においては、大当たり中用コマンド送信処理を例に挙げ、本実施形態に係るスロットマシン1100においては、コマンド設定送信処理を例に挙げたが、このような処理に限らず、様々な処理や命令に適用可能である。

### 【1234】

＜主制御部の第2特殊命令＞

次に、主制御部300が備える第5特殊命令について説明する。図161は、主制御部300が備える第5特殊命令の一部と、その説明を示した図である。

### 【1235】

＜第5特殊命令／WARU命令＞

第5特殊命令の一つである「WARU r、A、n（n：イミディエイト値）」命令は、第3オペランドが示す数値nで第2オペランドが示すAレジスタに記憶されたデータを除算し、除算結果の商を第1オペランドrが示すレジスタ（B、C、D、E、H、Lレジスタのいずれかのレジスタ）に格納し、除算結果の余りを第2オペランドが示すAレジスタに格納する命令である。また、除算結果の余りが0の場合にZフラグ、SZフラグが1にセットされ、除算結果の余りが0以外の場合にZフラグ、SZフラグが0にセットされる。また、除数が0の場合にCフラグが1にセットされ、かつ商と余りがFFH（2バイトの場合はFFFFH）となり、除数が0以外の場合にCフラグが0にセットされる。

### 【1236】

また、「WARU A、r」命令は、第2オペランドが示すレジスタrに記憶されたデータで第1オペランドが示すAレジスタに記憶されたデータを除算し、除算結果の商を第1オペランドが示すAレジスタに格納し、除算結果の余りを第2オペランドが示すレジスタrに格納する命令である。また、「WARU ss、HL」命令は、第2オペランドが示すペアレジスタHLに記憶されたデータで第1オペランドが示すssレジスタ（BC、DEレジスタ）に記憶されたデータを除算し、除算結果の商を第1オペランドが示すssレジスタに格納し、除算結果の余りを第2オペランドが示すペアレジスタHLに格納する命令である。また、「WARU HL、ss」命令は、第2オペランドが示すペアレジスタss（BC、DEレジスタ）に記憶されたデータで第1オペランドが示すHLレジスタに記憶されたデータを除算し、除算結果の商を第1オペランドが示すHLレジスタに格納し、除算結果の余りを第2オペランドが示すペアレジスタssに格納する命令である。

### 【1237】

＜第5特殊命令／JP命令＞

第5特殊命令の一つである「JP SZ、mn命令」は、SZフラグが1の場合にmnが示すアドレスにジャンプし、SZフラグが0の場合に後続の処理を継続する命令であり、「JP NSZ、mn命令」は、SZフラグが0の場合にmnが示すアドレスにジャンプし、SZフラグが1の場合に後続の処理を継続する命令である。

### 【1238】

＜第5特殊命令／JR命令＞

第5特殊命令の一つである「JR SZ、e命令」は、SZフラグが1の場合にプログラムカウンタPC+eが示すアドレスにジャンプし、SZフラグが0の場合に後続の処理を継続する命令であり、「JR NSZ、mn命令」は、SZフラグが0の場合にプログラムカウンタPC+eが示すアドレスにジャンプし、SZフラグが1の場合に後続の処理を継続する命令である。

### 【1239】

＜第5特殊命令／CALL命令＞

10

20

30

40

50

第5特殊命令の一つである「CALL SZ、mn命令」は、SZフラグが1の場合に、プログラムカウンタPCに記憶された上位アドレスをスタックポインタSP-1が示すスタック領域に、また、プログラムカウンタPCに記憶された下位アドレスをスタックポインタSP-2が示すスタック領域にそれぞれストア（退避）した後、mnが示すアドレスにジャンプし、SZフラグが0の場合に後続の処理を継続する命令である。また、「CALL NSZ、mn命令」は、SZフラグが0の場合に、プログラムカウンタPCに記憶された上位アドレスをスタックポインタSP-1が示すスタック領域に、また、プログラムカウンタPCに記憶された下位アドレスをスタックポインタSP-2が示すスタック領域にそれぞれストア（退避）した後、mnが示すアドレスにジャンプし、SZフラグが1の場合に後続の処理を継続する命令である。

10

#### 【1240】

##### <第5特殊命令／RET命令>

第5特殊命令の一つである「RET SZ命令」は、SZフラグが1の場合に、スタックポインタSPが示すスタック領域SPに記憶されたデータをプログラムカウンタPCの下位アドレスに、また、スタックポインタSPが示すスタック領域SP+1に記憶されたデータをプログラムカウンタPCの上位アドレスにそれぞれロード（復帰）した後にスタックポインタSPを2つ加算し、SZフラグが0の場合に後続の処理を継続する命令である。また、「RET NSZ命令」は、SZフラグが0の場合に、スタックポインタSPが示すスタック領域SPに記憶されたデータをプログラムカウンタPCの下位アドレスに、また、スタックポインタSPが示すスタック領域SP+1に記憶されたデータをプログラムカウンタPCの上位アドレスにそれぞれロード（復帰）した後にスタックポインタSPを2つ加算し、SZフラグが1の場合に後続の処理を継続する命令である。

20

#### 【1241】

##### <第5特殊命令／バイト数・ステート数>

図162は、第5特殊命令のバイト数とステート数を比較した図である。同図に示すように、JP命令およびCALL命令は、Zフラグをオペランドとする命令と、SZフラグをオペランドとする命令でバイト数とステート数が異なり、Zフラグをオペランドとする命令よりも、SZフラグをオペランドとする命令のほうがバイト数が1バイト、ステート数が4ステート多くなっている。一方、JR命令およびRET命令は、Zフラグをオペランドとする命令と、SZフラグをオペランドとする命令でバイト数とステート数が同じになっている。

30

#### 【1242】

##### <Zフラグ、SZフラグ>

図163は、ZフラグやSZフラグを変化させる命令の一例を示した図である。主制御部300は、命令の実行後にZフラグとSZフラグを同じ値に変化させる命令と、命令の実行後にZフラグを変化させずにSZフラグを変化させる命令と、命令の実行後にZフラグを変化させてSZフラグを変化させない命令と、を備えている。例えば、第5特殊命令の一つであるWARU命令は、命令の実行後にZフラグとSZフラグを同じ値に変化させる命令の一つであり、特殊命令の一つであるINCTENSO命令は、命令の実行後にZフラグを変化させずにSZフラグを変化させる命令の一つである。

40

#### 【1243】

##### <タイマ更新処理>

次に、図164を用いて、上述の主制御部タイマ割込処理におけるタイマ更新処理（ステップS213）について説明する。なお、同図は、タイマ更新処理の流れを示すフローチャートである。

#### 【1244】

ステップS5501では、RAM308に設けた更新タイマ数（更新が必要なタイマの数）を取得し、ステップS5502では、更新の対象とするタイマ（更新対象タイマ）が記憶されているRAM308の記憶領域の先頭アドレスを設定する。また、ステップS5503では、更新対象タイマの値が0か0以外かを判定し、0以外の場合にはステップS

50



5504に進み、0の場合にはステップS5505に進む。

【1245】

ステップS5504では、更新対象タイマを更新し、ステップS5505では、未更新のタイマがあるか無いかを判定し、未更新のタイマがある場合にはステップS5503に戻って他のタイマの更新を継続し、未更新のタイマが無い場合には処理を終了する。

【1246】

図165(a)は、本発明に係るタイマ更新処理のプログラムの一例を示した図であり、同図165(b)は、従来のタイマ更新処理のプログラムの一例を示した図である。

【1247】

従来のタイマ更新処理のプログラムでは、上記ステップS5503において更新対象タイマの値が0か0以外かを判定する場合、まず、プログラムの(1-3)で示す「LD A、(HL)」命令を実行することによって、HLレジスタで示されるアドレスに記憶された更新対象タイマの値を、Aレジスタにロード(格納)する。続いて、プログラムの(1-4)で示す「AND A」命令を実行することによって、Aが0の場合にはZフラグを変化させ、Aが0以外の場合にはZフラグを変化させないように構成し、次のプログラムの(1-5)で示す、Zフラグを第1オペランドとするJR命令で、Zフラグが変化している場合(Aが0の場合)には、JR命令の第2オペランドで示されるアドレスにジャンプし、Zフラグが変化していない場合(Aが0以外の場合)には、後続の処理(この例では、DEC命令)を実行するようにしている。

【1248】

一方、本発明に係るタイマ更新処理のプログラムで同様の判定を行う場合、まず、従来のタイマ更新処理のプログラムと同様に、プログラムの(1-1)で示す「LD A、(HL)」命令を実行することによって、HLレジスタで示されるアドレスに記憶された更新対象タイマの値を、Aレジスタにロード(格納)する。ここで、LD命令(ロード命令)は、上記図162に示す通り、命令の実行後にZフラグを変化させずにSZフラグを変化させる命令の一つであるため、「LD A、(HL)」命令の実行後にはZフラグは変化しないが、SZフラグが変化する。続いて、本発明では、プログラムの(1-2)で示す、SZフラグを第1オペランドとするJR命令で、SZフラグが変化している場合には、JR命令の第2オペランドで示されるアドレスにジャンプし、SZフラグが変化していない場合には、後続の処理(この例では、DEC命令)を実行するようにしている。

【1249】

すなわち、本発明に係るタイマ更新処理のプログラムでは、従来のタイマ更新処理のプログラムの(1-4)で示す「AND A」命令(Aレジスタの値が0か否かを確認する処理)が不要となる。このため、従来よりもプログラムのコード量を減らすことができる上に、処理速度を高めることができ、遊技制御を安定して行うことができる。

【1250】

<表示図柄抽選処理>

次に、図166を用いて、上述の特図1関連抽選処理における表示図柄抽選(ステップS1003)について説明する。なお、同図は、表示図柄抽選の流れを示すフローチャートである。

【1251】

同図(a)に示す表示図柄抽選処理のステップS5601では、ROM306に設けた表示図柄抽選テーブルのアドレス(テーブルアドレス)を取得する。また、ステップS5602では、抽選準備処理を行い、ステップS5603では、抽選処理を行う。ステップS5604では、当り種別情報設定処理を行った後に処理を終了する。

【1252】

また、同図(b)に示す抽選準備処理のステップS5611では、オフセット値を取得し、ステップS5612では、取得済みの表示図柄抽選テーブルのアドレスを1つ加算して更新する。ステップS5613では、データ選択処理を行い、ステップS5614では、データ選択処理の結果である取得値が0か0以外かを判定し、0の場合には処理を終了

し、0以外の場合にはステップS5615に進む。ステップS5615では、抽選用乱数値を取得した後に処理を終了する。

【1253】

また、同図(c)に示すデータ選択処理のステップS5621では、テーブルアドレスにオフセット値を加算し、ステップS5622では、テーブルアドレスの値をAレジスタに転送する。また、ステップS5623では、Aレジスタの値を1つ減算し、ステップS5624では、テーブルアドレスの値をAレジスタに転送した後に処理を終了する。

【1254】

図167は、本発明に係る抽選準備処理のプログラムの一例を示した図である。本発明に係る抽選準備処理のプログラムでは、プログラムの(3-1)で示すRST命令を実行することによって、データ選択処理にジャンプし、このデータ選択処理のプログラムの(3-3)で示す「DEC A」命令を実行することによって、Aレジスタの値を1つ減算する。ここで、8ビットDEC命令は、上記図162に示す通り、命令の実行後にZフラグとSZフラグを同じ値に変化させる命令の一つであるため、「DEC A」命令の実行後にはZフラグとSZフラグが同じ値に変化する。

【1255】

続いて、プログラムの(3-4)で示す「LD A, (HL)」命令を実行することによって、テーブルアドレスの値をAレジスタに転送する。ここで、LD命令(ロード命令)は、上記図162に示す通り、命令の実行後にZフラグを変化させずにSZフラグを変化させる命令の一つであるため、「LD A, (HL)」命令の実行後にはZフラグは変化しないが、SZフラグが変化する。

【1256】

続いて、プログラムの(3-5)で示す「RET」命令を実行することによって、抽選準備処理のプログラムの(3-1)で示すRST命令の直後に戻り、(3-2)で示す「RET Z」命令を実行する。ここで、「RET Z命令」は、Zフラグが1の場合に、スタックポインタSPが示すスタック領域SPに記憶されたデータをプログラムカウンタPCの下位アドレスに、また、スタックポインタSPが示すスタック領域SP+1に記憶されたデータをプログラムカウンタPCの上位アドレスにそれぞれロード(復帰)した後にスタックポインタSPを2つ加算し、Zフラグが0の場合に後続の処理を継続する命令である。すなわち、(3-2)で示す「RET Z」命令では、(3-3)で示す「DEC A」命令の実行後には変化するが、次の(3-4)で示す「LD A, (HL)」命令の実行後には変化しないZフラグに応じて、表示図柄抽選処理に復帰する処理か、後続の処理を継続する処理のいずれかの処理を行う。

【1257】

<抽選処理>

次に、図168を用いて、上述の表示図柄抽選における抽選処理(ステップS5603)について説明する。なお、同図は、抽選処理の流れを示すフローチャートである。

【1258】

ステップS5701では、表示図柄抽選用乱数値を取得し、ステップS5702では、ステップS5701で取得した乱数値を除算する。ステップS5703では、ROM308に記憶している抽選データアドレスを取得し、ステップS5704では、ステップS5703で取得した抽選データアドレスに除算結果の商を加算する。ステップS5705では、除算結果の余りをAレジスタに転送し、ステップS5706では、Aレジスタの値と2を比較する。

【1259】

ステップS5707では、Zフラグが1か0かを判定し、0の場合にはステップS5708に進み、1の場合にはステップS5714に進む。ステップS5708では、Aレジスタの値と1を比較し、ステップS5709では、Zフラグが1か0かを判定し、0の場合にはステップS5710に進み、1の場合にはステップS5712に進む。

【1260】

ステップS5710では、抽選データアドレスの値をAレジスタに転送し、次のステップS5711では、抽選結果データをAレジスタにセットして処理を終了する。また、ステップS5712では、抽選データアドレスの値をAレジスタに転送し、次のステップS5713では、抽選結果データをAレジスタにセットして処理を終了する。また、ステップS5714では、抽選データアドレスの値をAレジスタに転送し、次のステップS5715では、抽選結果データをAレジスタにセットして処理を終了する。

【1261】

図169は、本発明に係る抽選処理のプログラムの一例を示した図であり、図170は、従来の抽選処理のプログラムの一例を示した図である。

【1262】

本発明に係る抽選準備処理のプログラムでは、ROMエリアに記憶された抽選テーブルに示されるように、3つの抽選値N、M、L（N、M、Lは正の整数）に対して「 $N \times 6 + M \times 6 + L$ 」の計算式からなる演算を施すことによって、1バイトのROMに対して3つの抽選値を対応付けして記憶している。このため、図170に示す、1バイトのROMに対して1つの抽選値を対応付けして記憶するような従来の抽選テーブルに比べ、メモリ容量を1/3に削減することができる。

【1263】

また、本発明に係る抽選準備処理のプログラムでは、1バイトのROMに対して3つの抽選値を対応付けして記憶した抽選テーブルから抽選値を取得するために、抽選値Lに対しては、（5-1）で示されるようにWARU命令を1回実行し、抽選値Mに対しては、（5-2）で示されるようにWARU命令を2回実行し、抽選値Nに対しては、（5-3）で示されるようにWARU命令を3回実行している。

【1264】

<払出要求数送信処理>

次に、図171を用いて、上述の主制御部タイマ割込処理における払出要求数送信処理（ステップS219）について説明する。なお、同図は、払出要求数送信処理の流れを示すフローチャートである。

【1265】

ステップS5801では、払出要求があるか無いかを判定し、払出要求がある場合にはステップS5802に進み、払出要求が無い場合には処理を終了する。ステップS5802では、送信対象となる払出要求カウンタを1つ減算し、ステップS5803では、送信対象に応じた賞球数コマンドデータを取得し、ステップS5804では、払出制御部600に送信する賞球コマンドを設定する。

【1266】

ステップS5805では、賞球数にメイン賞球数カウンタの値を加算し、ステップS5806では、賞球数を除算する。また、ステップS5807では、メイン賞球数カウンタを更新し、ステップS5808では、メイン賞球情報信号出力要求数を更新した後に処理を終了する。

【1267】

図172は、本発明に係る払出要求数送信処理のプログラムの一例を示した図である。このプログラムの（5-1）で示すLD命令では、メイン賞球数カウンタが記憶されているRAM308のアドレス（この例では、F060H）をHLレジスタに格納し、次の（5-2）で示すADD命令では、Aレジスタの値（賞球数）に、HLレジスタが示すアドレスに記憶されたメイン賞球カウンタの値（メイン賞球数）を加算している。

【1268】

また、次の（5-3）で示すWARU命令では、第3オペランドが示す数値（この例では、10）で第2オペランドが示すAレジスタに記憶されたデータを除算し、除算結果の商を第1オペランドrが示すレジスタ（この例では、Cレジスタ）に格納し、除算結果の余りを第2オペランドが示すAレジスタに格納する。例えば、Aレジスタに記憶されたデータが23である場合、Cレジスタには、23（Aレジスタに記憶されたデータ）÷10

10

20

30

40

50

（第 3 オペランドが示す数値）の除算の商である 2 が C レジスタに格納され、除算の余りである 3 が A レジスタに格納される。

【 1 2 6 9 】

また、次の（5－4）で示す LD 命令では、A レジスタに記憶されたデータ（除算の商。先の例では 3）を、RAM 308 のアドレス F060H に記憶されたメイン賞球カウンタに、次回の賞球数として格納し、次の（5－5）～（5－8）に示す命令では、C レジスタに記憶されたデータ（除算の余り。先の例では 2）を、RAM 308 のアドレス F061H に記憶されたメイン賞球情報信号出力要求数記憶領域に、メイン賞球情報信号出力要求数として格納する。

【 1 2 7 0 】

10

また、本実施形態に係るパチンコ機 100（またはスロットマシン 1100）は、遊技制御を行う CPU（例えば、CPU 304（または、CPU 1304））を有する主制御部（例えば、主制御部 300（または、主制御部 1300））と、前記主制御部の遊技制御に基づいて遊技の演出制御を行う副制御部（例えば、第 1 副制御部 400、第 2 副制御部 500（または、第 1 副制御部 1400、第 2 副制御部 1500））と、を備えた遊技台であって、前記 CPU は、第一のフラグと第二のフラグとを有するフラグレジスタと、前記第一のフラグと前記第二のフラグとを処理実行結果に基づいて同じ値に変化させる第一の処理と、処理実行結果に基づいて前記第一のフラグの値を変化させず前記第二のフラグの値を変化させる第二の処理と、を備え、前記第一の処理と前記第二の処理とを前記遊技制御において実行することを特徴とする遊技台である。

20

【 1 2 7 1 】

例えば、本実施形態に係るパチンコ機 100（またはスロットマシン 1100）の主制御部は、図 173 に示すように、フラグ A とフラグ B とを有し、フラグ A とフラグ B とを処理実行結果に基づいて同じ値  $\beta$  に変化させる第一の処理と、処理実行結果に基づいてフラグ A の値を変化させずフラグ B の値を  $\gamma$  に変化させる第二の処理と、を備え、前記第 1 の処理と前記第 2 の処理とを前記遊技制御において実行する。

【 1 2 7 2 】

本実施形態に係るパチンコ機 100（またはスロットマシン 1100）によれば、遊技の興趣を高めることを可能にしつつ、安定した遊技制御をおこなうことができる。すなわち、第二の処理実行後は第二の処理実行前の第一のフラグの値が保護されるため、その後の処理で第一のフラグの値を活用することができ、他のレジスタに第一のフラグの値を退避させる無駄な処理をする必要が無く、レジスタを占有せず且つプログラム容量を削減でき、遊技性を向上させるためのプログラムに容量を割り当てることができ、遊技の興趣を高めることが可能となる。また、プログラム容量を削減することでプログラムが見やすくなり、コーディングミスやバグの発生を抑制できる。また、第一の処理実行後は第一のフラグと第二のフラグが同じ値に変化するため、いずれかのフラグの値を用いて処理を実行したとしても同じ結果を得ることができ、状況に応じた適切な処理を選択してプログラムを記述することができ、プログラム容量や処理速度などを最適化することができる場合がある。

30

【 1 2 7 3 】

40

また、前記 CPU は、前記第一の処理において、処理実行結果が第一の値であることに基づいて前記第一のフラグと前記第二のフラグを第二の値に変化させ、処理実行結果が前記第一の値と異なることに基づいて前記第一のフラグと前記第二のフラグを前記第一の値に変化させ、前記第二の処理において、処理実行結果が前記第一の値であることに基づいて前記第二のフラグを前記第二の値に変化させ、処理実行結果が前記第一の値と異なることに基づいて前記第二のフラグを第一の値に変化させてもよい。

【 1 2 7 4 】

このような構成とすれば、第一の処理実行後は第一のフラグと第二のフラグが処理実行結果に基づく同じ値に変化するため、いずれかのフラグの値を用いて処理を実行したとしても同じ結果を得ることができ、状況に応じた適切な処理を選択してプログラムを記述す

50

ることができ、プログラム容量や処理速度などを最適化することができる場合がある。

【1275】

また、所定ビット長の特定レジスタを有し、前記CPUは、前記第一の処理において、処理実行結果を前記特定レジスタに格納し、前記特定レジスタの値が前記第一の値であることに基づいて前記第一のフラグと前記第二のフラグを前記第二の値に変化させ、前記特定レジスタの値が前記第一の値と異なることに基づいて前記第一のフラグと前記第二のフラグを前記第一の値に変化させ、前記第二の処理において、処理実行結果を前記特定レジスタに格納し、前記特定レジスタの値が前記第一の値であることに基づいて前記第二のフラグを前記第二の値に変化させ、前記特定レジスタの値が前記第一の値と異なることに基づいて前記第二のフラグを前記第一の値に変化させてもよい。

10

【1276】

このような構成とすれば、第一の処理実行後は第一のフラグと第二のフラグが特定レジスタの値に基づく同じ値に変化するため、いずれかのフラグの値を用いて処理を実行したとしても同じ結果を得ることができ、状況に応じた適切な処理を選択してプログラムを記述することができ、プログラム容量や処理速度などを最適化することができる場合がある。

【1277】

また、前記第一の処理の実行によって変化した前記第一のフラグの値に基づいて第一の処理を行う処理と、前記第一の処理の実行によって変化した前記第二のフラグの値に基づいて前記第一の処理と同じ処理を行う処理とを備えてもよい。

20

【1278】

例えば、図174(a)に示すように、第一の処理(SUB命令)の実行によって変化した第一のフラグ(Zフラグ)の値に基づいて第一の処理(分岐処理)を行う第三の処理(ZフラグをオペランドとするJR命令)と、同図(b)に示すように、第一の処理(SUB命令)の実行によって変化した第二のフラグ(SZフラグ)の値に基づいて第一の処理と同じ処理(分岐処理)を行う第四の処理(SZフラグをオペランドとするJR命令)とを備えてもよい。

【1279】

このような構成とすれば、異なるフラグの変化に基づいて同一の処理を実行することができ、処理に応じて第三の処理と第四の処理を使い分けることができる場合がある。特に、第三の処理と第四の処理のバイト数とステート数を同じにすれば、プログラムのチェックや、デバックやテストなどの作業が容易になる場合がある。

30

【1280】

また、前記CPUは、前記第一のフラグの値を判定する第三の処理と、前記第二のフラグの値を判定する第四の処理と、を備え、前記第三の処理において、前記第一のフラグの値が前記第一の値または前記第二の値であるか否かを判定し、前記第四の処理において、前記第一のフラグの値が前記第一の値または前記第二の値であるか否かを判定してもよい。

【1281】

このような構成とすれば、第一の処理実行後は第一のフラグの値と第二のフラグの値が同じとなるので、第一の処理実行後に第三の処理と第四の処理を使い分けることで、プログラムの容量や可読性を向上させることができる場合がある。また、第二の処理実行後は第二の処理実行前の第一のフラグの値が保護されるため、第三の処理で第一のフラグの値を活用することができ、他のレジスタに第一のフラグの値を退避させる無駄な処理をする必要が無く、レジスタを占有せず且つプログラム容量を削減でき、遊技性を向上させるためのプログラムに容量を割り当てることができ、遊技の興趣を高めることが可能となる。

40

【1282】

また、前記第二の処理実行後に前記第四の処理を実行すれば、第四の処理で第二の処理の実行結果を判定することができるので、第二の処理の実行結果を確認する処理(第一のフラグを変化させる処理)を省くことができ、プログラム容量を削減することができる場

50

合がある。

【1283】

また、前記CPUは、前記第三の処理または前記第四の処理において、前記判定結果が真である場合に所定のアドレスに移行してもよく、また、前記判定結果が真のときより偽の方が処理速度が速い場合、前記第三の処理と前記第四の処理のサイズ（バイト数）と速さ（ステート数）が同一であれば（例えば、JR命令、RET命令）、プログラムコード量や処理速度を気にすることなく処理を使い分けることができる場合がある。

【1284】

また、前記判定結果が真のときと偽のときとで処理速度が同じ場合、前記第三の処理のほうが前記第四の処理よりサイズと速さが小さくてもよい（例えば、JP命令）。また、前記第二の処理は、上記実施形態で示した処理に限定されず、例えば、ロード命令、16ビットのADD命令、INC命令、DEC命令、直値を入れるIN命令、Aレジスタ専用のローテートシフト命令などでもよい。

【1285】

また、前記CPUが前記第二の処理を実行した後の前記第一のフラグは、該第二の処理を実行する前の値を維持するものであってもよい。

【1286】

また、本実施形態に係るパチンコ機100（またはスロットマシン1100）は、遊技制御を行うCPU（例えば、CPU304（または、CPU1304））を有する主制御部（例えば、主制御部300（または、主制御部1300））と、前記主制御部の遊技制御に基づいて遊技の演出制御を行う副制御部（例えば、第1副制御部400、第2副制御部500（または、第1副制御部1400、第2副制御部1500））と、を備えた遊技台であって、前記CPUは、複数種類のフラグを備えた所定のフラグレジスタと複数のレジスタを有し、前記複数のレジスタのうちの第一のレジスタと第二のレジスタの少なくとも何れかに格納された値を所定の値で所定の演算を行い、前記演算の第一の結果を前記第一のレジスタに格納し、前記演算の第二の結果を前記第二のレジスタに格納し、前記演算の第二の結果である前記第二のレジスタに格納された値に基づいて前記複数種類のフラグのうちの特定のフラグの値を変化させる所定の処理を備え、該所定の処理を前記遊技制御において実行することを特徴とする遊技台である。

【1287】

例えば、本実施形態に係るパチンコ機100（またはスロットマシン1100）は、図175に示すように、Zフラグと複数の記憶部1、記憶部2を有し、記憶部1と記憶部2の少なくとも何れかに格納された値7を所定の値3で除算を行い、除算結果の商2を前記記憶部1に格納し、除算結果の余り1を前記記憶部2に格納し、前記記憶部2に格納された値に基づいてZフラグを1から0に変化させる除算処理を備え、該除算処理を前記遊技制御において実行する。

【1288】

本実施形態に係るパチンコ機100（またはスロットマシン1100）によれば、遊技の興趣を高めることを可能にしつつ、安定した遊技制御をおこなうことができる。すなわち、第二のレジスタに格納された値（第二の結果）に基づいて複数種類のフラグの何れかの値が変化するので、プログラムを注意して作成でき、コーディングミスやバグを抑制することができ、安定した遊技制御をおこなうことができる場合がある。また、第一の結果と第二の結果をそれぞれレジスタに格納するため、所定の処理後に2つの結果を利用してその他の処理を行うことができ、プログラム容量を削減することでプログラムが見やすくなり、コーディングミスやバグの発生を抑制できる場合がある。

【1289】

また、前記所定の処理は、前記第二のレジスタに格納された値が第一の値であることに基づいて第二の値を前記フラグレジスタに格納し、前記第二のレジスタに格納された値が前記第一の値と異なることに基づいて前記複数種類のフラグのうちの前記特定のフラグの値を前記第一の値に変化させてもよい。

## 【1290】

このような構成とすれば、第二のレジスタに格納された処理実行結果（第二の結果）に基づく値に基づいて特定のフラグの値が変化するので、プログラムを注意して作成でき、コーディングミスやバグを抑制することができ、安定した遊技制御をおこなうことができる場合がある。

## 【1291】

また、前記CPUは、前記所定の処理において、前記演算の第一の結果である前記第一のレジスタに格納された値に関わらずに前記演算の第二の結果である前記第二のレジスタに格納された値に基づいて前記複数種類のフラグのうちの特定のフラグの値を変化させてもよい。

10

## 【1292】

また、前記所定の処理は、第一のオペランドが特定値の場合に、第二のオペランドと第三のオペランドで指定したレジスタを所定の値に変化させるとともに、前記所定の処理で変化する複数種類のフラグのうちの前記特定のフラグ以外のフラグを変化させる処理でもよい。

## 【1293】

例えば、図176(a)に示すWARU命令は、第一のオペランド（三番目のオペランド）が特定値（0）の場合に、第二のオペランド（一番目のオペランド）と第三のオペランド（二番目のオペランド）で指定したレジスタ（Cレジスタ、Aレジスタ）を所定の値（FFH）に変化させるとともに、前記所定の処理で変化する複数種類のフラグのうちの特定のフラグ以外のフラグ（Cフラグ）を変化させる処理でもよい。

20

## 【1294】

このような構成とすれば、同図(b)に示すような2つの処理で実現するよりも、処理数を削減できるため、処理時間を早くすることができ、プログラムコード量も削減できる場合がある。また、指定した2つのレジスタに所定の値を1処理で格納することができ、プログラムの可読性を向上させることができる場合がある。

## 【1295】

なお、Cフラグ（キャリーフラグ）は、足し算（例えば、ADD命令、INC命令）実行時の桁落ち、引き算（例えば、SUB命令、DEC命令）実行時の桁借り、ローテートシフト命令（例えば、RLCA命令）実行時のCフラグに入る値が1のときなどで値が1に変化するため、特定の処理（例えば、ADD命令、DEC命令、RLCA命令）と前記所定の処理（WARU命令）とではCフラグの変化条件が異なる。また、被除数を特定値（0）の除数で除算した場合、第一のレジスタおよび第二のレジスタの値における演算前の値がつぶされて（演算前の値に関わらず上書きして）同一の特定の値（FFH）になる。また、前記所定の値は、前記所定の処理の演算結果として、前記第二のオペランド（一番目のオペランド）と前記第三のオペランド（二番目のオペランド）で指定したレジスタ（Cレジスタ、Aレジスタ）に格納する値として取りえない値（除算の結果として、商と余りが取り得ない値（例えば、商と余りが両方ともFFH））であってもよい。

30

## 【1296】

また、前記所定の処理は、最後に指定されるオペランドの値で最後から2番目に指定されるオペランドの値に対して前記所定の演算を行い、最初に指定されるオペランドである前記第一のレジスタに前記演算の第一の結果を格納し、最初から2番目に指定されるオペランドである前記第二のレジスタに前記演算の第二の結果を格納し、前記演算の第一の結果である前記第一のレジスタに格納された値に関わらずに前記演算の第二の結果である前記第二のレジスタに格納された値に基づいて前記複数種類のフラグの何れかの値を変化させるものであってもよい。

40

## 【1297】

この場合、最初から2番目に指定されるオペランドである第二のレジスタの値に基づいて複数種類のフラグの何れかの値が変化するので、プログラムを注意して作成でき、コーディングミスやバグを抑制することができ、安定した遊技制御をおこなうことができる場

50

合がある。

【1298】

また、前記最後に指定するオペランドの値が前記所定の値であってもよい。この場合、プログラムの可読性を向上させることができる場合がある。

【1299】

また、プログラムデータおよび一時的なデータを記憶する記憶手段（例えば、ROM 306、RAM 308（またはROM 1306、RAM 1308））の所定のアドレスが示す領域に複数のデータを含み、前記CPUは、前記所定の処理を所定回数行って該所定回数に応じた前記複数のデータの何れかを取得するものであってもよい。

【1300】

この場合、プログラム容量を削減することでプログラムが見やすくなり、コーディングミスやバグの発生を抑制できる場合がある。

【1301】

また、前記CPUは、前記所定の処理を行った後、前記第一のレジスタに格納した値を前記記憶手段の第一の特定アドレスが示す領域に格納し、前記第二のレジスタに格納した値を前記第一の特定アドレスに隣接する第二の特定アドレスが示す領域に格納するものであってもよい。

【1302】

この場合、プログラムが見やすくなり、コーディングミスやバグの発生を抑制できる場合がある。

【1303】

以下、図面を用いて、本発明の実施形態4に係るパチンコ機（遊技台）について詳細に説明する。

【1304】

＜全体構成＞

まず、図177を用いて、パチンコ機5100の全体構成について説明する。なお、同図はパチンコ機5100を正面側（遊技者側）から見た外観斜視図である。

【1305】

パチンコ機5100は、ガラス製または樹脂製の透明板部材5152および透明部材保持枠（ガラス枠）5154からなる扉部材5156の奥側に視認可能に配設した後述する遊技盤（盤面）5102を備えている。

【1306】

また、発射杆5138および発射槌5140の下方には、発射杆5138を制御して遊技領域5104に向けて球の発射強度の操作を行うための操作ハンドル5148を配設していると共に、貯留皿5144の下方には、貯留皿5144に貯留できない溢れ球を貯留するための下皿5150を設けている。

【1307】

図178は、遊技盤5102を正面から見た略示正面図である。遊技盤5102には、外レール5106と内レール5108とを配設し、遊技球（以下、単に「球」と称する場合がある。）が転動可能な遊技領域5104を区画形成している。

【1308】

遊技領域5104の略中央には、演出装置5200を配設している。この演出装置5200には、略中央に横長の装飾図柄表示装置5110を配設し、その周囲に、普通図柄表示装置5112と、特別図柄表示装置5114と、普通図柄保留ランプ5116と、特別図柄保留ランプ5118と、高確中ランプ5120を配設している。なお、以下、普通図柄を「普図」、特別図柄を「特図」と称する場合がある。

【1309】

演出装置5200は、可動部を動作して演出を行うものであり、詳細については後述する。装飾図柄表示装置5110は、装飾図柄ならびに演出に用いる様々な画像を表示するための表示装置であり、本実施例では液晶表示装置（Liquid Crystal D

10

20

30

40

50



display)によって構成する。この装飾図柄表示装置5110は、左図柄表示領域5110a、中図柄表示領域5110b、右図柄表示領域5110cおよび演出表示領域5110dの4つの表示領域に分割し、左図柄表示領域5110a、中図柄表示領域5110bおよび左図柄表示領域5110cはそれぞれ異なった装飾図柄を表示し、演出表示領域5110dは演出に用いる画像を表示する。さらに、各表示領域5110a、5110b、5110c、5110dの位置や大きさは、装飾図柄表示装置5110の表示画面内で自由に変更することを可能としている。なお、装飾図柄表示装置5110は、液晶表示装置に代えて、ドットマトリクス表示装置、7セグメント表示装置、EL(Electro Luminescence)表示装置、ドラム式表示装置、リーフ式表示装置等他の表示デバイスを採用してもよい。

10

#### 【1310】

普図表示装置5112は、普図の表示を行うための表示装置であり、本実施例では7セグメントLEDによって構成する。特図表示装置5114は、特図の表示を行うための表示装置であり、本実施例では7セグメントLEDによって構成する。

#### 【1311】

普図保留ランプ5116は、保留している普図変動遊技の数を示すためのランプであり、本実施例では、普図変動遊技を2つまで保留することを可能としている。特図保留ランプ5118は、保留している特図変動遊技の数を示すためのランプであり、本実施例では、特図変動遊技を4つまで保留することを可能としている。高確中ランプ5120は、遊技状態が高確率状態(後述する大当り遊技の当選確率を通常確率よりも高く設定した遊技状態)であること、または高確率状態になることを示すためのランプであり、遊技状態を低確率状態(後述する大当り遊技の当選確率を通常確率に設定した遊技状態)から高確率状態にする場合に点灯し、高確率状態から低確率状態にする場合に消灯する。

20

#### 【1312】

また、この演出装置5200の周囲には、一般入賞口5122と、普図始動口5124と、第1特図始動口5126と、第2特図始動口5128と、可変入賞口5130を配設している。一般入賞口5122は、本実施例では遊技盤5102に複数配設しており、この一般入賞口5122への入球を所定の球検出センサ(図示省略)が検出した場合(一般入賞口5122に入賞した場合)、後述する払出装置552を駆動し、所定の個数(本実施例では10個)の球を賞球として貯留皿5144に排出する。貯留皿5144に排出した球は遊技者が自由に取り出すことが可能であり、これらの構成により、入賞に基づいて賞球を遊技者に払い出すようにしている。なお、一般入賞口5122に入球した球は、パチンコ機5100の裏側に誘導した後、遊技島側に排出する。本実施例では、入賞の対価として遊技者に払い出す球を「賞球」、遊技者に貸し出す球を「貸球」と区別して呼ぶ場合があり、「賞球」と「貸球」を総称して「球(遊技球)」と呼ぶ。

30

#### 【1313】

普図始動口5124は、ゲートやスルーチャッカーと呼ばれる、遊技領域の所定の領域を球が通過したか否かを判定するための装置で構成しており、本実施例では遊技盤5102の左側に1つ配設している。普図始動口5124を通過した球は一般入賞口5122に入球した球と違って、遊技島側に排出することはない。球が普図始動口5124を通過したことを所定の玉検出センサが検出した場合、パチンコ機5100は、普図表示装置5112による普図変動遊技を開始する。

40

#### 【1314】

第1特図始動口5126は、本実施例では遊技盤5102の中央に1つだけ配設している。この第1特図始動口5126への入球を所定の球検出センサが検出した場合、後述する払出装置552を駆動し、所定の個数(本実施例では3個)の球を賞球として貯留皿5144に排出するとともに、特図表示装置5114による特図変動遊技を開始する。なお、第1特図始動口5126に入球した球は、パチンコ機5100の裏側に誘導した後、遊技島側に排出する。

#### 【1315】

50

第2特図始動口5128は、電動チューリップ（電チュー）と呼ばれ、本実施例では第1特図始動口5126の真下に1つだけ配設している。この第2特図始動口5128は、左右に開閉自在な羽根を備え、羽根の閉鎖中は球の入球が不可能であり、普図変動遊技に当選し、普図表示装置5112が当り図柄を停止表示した場合に羽根が所定の時間間隔、所定の回数で開閉する。第2特図始動口5128への入球を所定の球検出センサが検出した場合、後述する払出装置552を駆動し、所定の個数（本実施例では5個）の球を賞球として後述する貯留皿5144に排出するとともに、特図表示装置5114による特図変動遊技を開始する。なお、第2特図始動口5128に入球した球は、パチンコ機5100の裏側に誘導した後、遊技島側に排出する。

#### 【1316】

可変入賞口5130は、大入賞口またはアタッカーと呼ばれ、本実施例では遊技盤5102の中央部下方に1つだけ配設している。この可変入賞口5130は、開閉自在な扉部材を備え、扉部材の閉鎖中は球の入球が不可能であり、特図変動遊技に当選し、特図表示装置5114が大当り図柄を停止表示した場合に扉部材が所定の時間間隔（例えば、開放時間29秒、閉鎖時間1.5秒）、所定の回数（例えば15回）で開閉する。可変入賞口5130への入球を所定の球検出センサが検出した場合、後述する払出装置552を駆動し、所定の個数（本実施例では15球）の球を賞球として貯留皿5144に排出する。なお、可変入賞口5130に入球した球は、パチンコ機5100の裏側に誘導した後、遊技島側に排出する。

#### 【1317】

さらに、これらの入賞口や始動口の近傍には、風車と呼ばれる円盤状の打球方向変換部材5132や、遊技釘5134を複数個、配設していると共に、内レール5108の最下部には、いずれの入賞口や始動口にも入賞しなかった球をパチンコ機5100の裏側に誘導した後、遊技島側に排出するためのアウト口5136を設けている。

#### 【1318】

このパチンコ機5100は、遊技者が貯留皿5144に貯留している球を発射レール5142の発射位置に供給し、遊技者の操作ハンドル5148の操作量に応じた強度で発射モータ602を駆動し、発射杆5138および発射槌5140によって外レール5106、内レール5108を通過させて遊技領域5104に打ち出す。そして、遊技領域5104の上部に到達した球は、打球方向変換部材5132や遊技釘5134等によって進行方向を変えながら下方に落下し、入賞口（一般入賞口5122、可変入賞口5130）や始動口（第1特図始動口5126、第2特図始動口5128）に入賞するか、いずれの入賞口や始動口にも入賞することなく、または普図始動口5124を通過するのみでアウト口5136に到達する。

#### 【1319】

##### <演出装置>

次に、パチンコ機5100の演出装置5200について説明する。この演出装置5200の前面側には、ワープ装置5230およびステージを配設し、演出装置5200の背面側には、装飾図柄表示装置5110および遮蔽手段5250を配設している。すなわち、演出装置5200において、装飾図柄表示装置5110および遮蔽手段5250は、ワープ装置5230およびステージの後方に位置することとなる。

#### 【1320】

ワープ装置5230は、演出装置5200の左上方に設けた入球口5232に入った遊技球を演出装置5200の前面下方の前面ステージ5234に排出し、さらに、前面ステージ5234に排出した遊技球が前面ステージ5234の中央部後方に設けた第2の入球口5236に入った場合は、遊技球を、第1特図始動口5126の上方である演出装置5200の下部中央に設けた排出口5238から第1特図始動口5126に向けて排出するものである。この排出口5238から排出した遊技球は特図始動口5126に入球しやすくなっている。

#### 【1321】

10

20

30

40

50

遮蔽手段 5250 は、格子状の左扉 5250 a および右扉 5250 b からなり、装飾図柄表示装置 5110 および前面ステージ 5234 の間に配設する。左扉 5250 a および右扉 5250 b の上部には、図示しない 2 つのプーリに巻き回したベルトをそれぞれ固定している。すなわち、左扉 5250 a および右扉 5250 b は、モータによりプーリを介して駆動するベルトの動作に伴って左右にそれぞれ移動する。遮蔽手段 5250 は、左右扉 5250 a、5250 b を閉じた状態ではそれぞれの内側端部が重なり、遊技者が装飾図柄表示装置 5110 を視認し難いように遮蔽する。左右扉 5250 a、5250 b を開いた状態ではそれぞれの内側端部が装飾図柄表示装置 5110 の表示画面の外側端部と若干重なるが、遊技者は装飾図柄表示装置 5110 の表示の全てを視認可能である。また、左右扉 5250 a、5250 b は、それぞれ任意の位置で停止可能であり、例えば、表示した装飾図柄がどの装飾図柄であるかを遊技者が識別可能な程度に、装飾図柄の一部だけを遮蔽するようなことができる。なお、左右扉 5250 a、5250 b は、格子の孔から後方の装飾図柄表示装置 5110 の一部を視認可能にしてもよいし、格子の孔の障子部分を半透明のレンズ体で塞ぎ、後方の装飾図柄表示装置 5110 による表示を漠然と遊技者に視認させるようにしてもよいし、格子の孔の障子部分を完全に塞ぎ（遮蔽し）、後方の装飾図柄表示装置 5110 を全く視認不可にしてもよい。

#### 【1322】

図 179 は、パチンコ機 5100 を背面側から見た外観斜視図である。パチンコ機 5100 の背面上部には、上方に開口した開口部を有し、球を一時的に貯留するための球タンク 5152 と、この球タンク 5152 の下方に位置し、球タンク 5152 の底部に形成した連通孔を通過して落下する球を背面右側に位置する払出装置 5154 に導くためのタンクレール 5153 とを配設している。

#### 【1323】

払出装置 5154 は、筒状の部材からなり、その内部には、スプロケット 5157 と払出センサ 5158 とを備えている。スプロケット 5157 は、モータによって回転可能に構成されており、タンクレール 5153 を通過して払出装置 5154 内に落下した球を一時的に滞留させると共に、モータを駆動して所定角度だけ回転することにより、一時的に滞留した球を払出装置 5154 の下方へ 1 個ずつ送り出すように構成している。

#### 【1324】

払出センサ 5158 は、スプロケット 5157 が送り出した球の通過を検知するためのセンサであり、球が通過しているときにオンの信号を出力し、球が通過していないときはオフの信号を出力する。なお、この払出センサ 5158 を通過した球は、図示しない球レールを通過してパチンコ機 5100 の表側に配設した貯留皿 5144 に到達するように構成しており、パチンコ機 5100 は、この構成により遊技者に対して球の払い出しを行う。

#### 【1325】

払出装置 5154 の左側には、後述する主制御部 5300 を構成する主基板 5161 と、後述する副制御部 5400 を構成するサブ基板 5164 とを配設している。また、これら主基板 5161 やサブ基板 5164 の下方には、後述する発射制御部 5600 を構成する発射基板 5166 と、後述する電源管理部 5650 を構成する電源基板 5162 と、後述する払出制御部 5550 を構成する払出基板 5165 と、この払出基板 5165 に接続した CR インターフェース部 5163 とを配設している。

#### 【1326】

##### <図柄の種類>

次に、図 180 (a) ~ (c) を用いて、パチンコ機 5100 の特図表示装置 5114、装飾図柄表示装置 5110、普図表示装置 5112 が停止表示する特図および普図の種類について説明する。

#### 【1327】

同図 (a) は特図の停止表示態様の一例を示したものである。本実施例の特図の停止表示態様には、大当り図柄である「特図 1」と、特別大当り図柄である「特図 2」と、外れ

図柄である「特図 3」の 3 種類がある。第 1 特図始動口 5 1 2 6 または第 2 特図始動口 5 1 2 8 に球が入賞したことを所定の球検出センサが検出したことを条件として特図変動遊技を開始した場合には、特図表示装置 5 1 1 4 は、7 個のセグメントの全点灯と、中央の 1 個のセグメントの点灯を繰り返す「特図の変動表示」を行う。そして、特図の変動開始前に決定した変動時間が経過すると、特図変動遊技の当選を報知する場合には「特図 1」または「特図 2」を停止表示し、特図変動遊技の外れを報知する場合には「特図 3」を停止表示する。なお、図中の白抜きの部分が消灯するセグメントの場所を示し、黒塗りの部分が点灯するセグメントの場所を示している。

#### 【1 3 2 8】

同図 (b) は装飾図柄の一例を示したものである。本実施例の装飾図柄には、「装飾 1」～「装飾 10」の 10 種類がある。第 1 特図始動口 5 1 2 6 または第 2 特図始動口 5 1 2 8 に球が入賞したことを所定の球検出センサが検出したことを条件にして、装飾図柄表示装置 5 1 1 0 の左図柄表示領域 5 1 1 0 a、中図柄表示領域 5 1 1 0 b、右図柄表示領域 5 1 1 0 c の各図柄表示領域に、「装飾 1」→「装飾 2」→「装飾 3」→・・・「装飾 9」→「装飾 10」→「装飾 1」→・・・の順番で表示を切り替える「装飾図柄の変動表示」を行う。そして、大当りを報知する場合には、図柄表示領域 5 1 1 0 a～5 1 1 0 c に大当りに対応する図柄組合せ（本実施例では、同一の数字の装飾図柄の組合せ（例えば、「装飾 2－装飾 2－装飾 2」））を停止表示し、特別大当りを報知する場合には、特別大当りに対応する図柄組合せ（本実施例では、同一の奇数番号数字の装飾図柄の組合せ（例えば、「装飾 1－装飾 1－装飾 1」））を停止表示する。なお、大当りに対応する図柄の組合せを停止表示した場合には、大当り遊技または特別大当り遊技を開始し、特別大当りに対応する図柄の組合せを停止表示した場合には、特別大当り遊技を開始する。また、外れを報知する場合には、図柄表示領域 5 1 1 0 a～5 1 1 0 c に大当りに対応する図柄組合せ以外の図柄組合せを停止表示した後で、保留している装飾図柄の変動表示があれば、その変動表示を開始する。

#### 【1 3 2 9】

同図 (c) は普図の停止表示態様の一例を示したものである。本実施例の普図の停止表示態様には、当り図柄である「普図 1」と、外れ図柄である「普図 2」の 2 種類がある。普図始動口 5 1 2 4 を球が通過したことを所定の球検出センサが検出したことを条件として普図表示遊技を開始した場合には、普図表示装置 5 1 1 2 は、7 個のセグメントの全点灯と、中央の 1 個のセグメントの点灯を繰り返す「普図の変動表示」を行う。そして、普図変動遊技の当選を報知する場合には「普図 1」を停止表示し、普図変動遊技の外れを報知する場合には「普図 2」を停止表示する。

#### 【1 3 3 0】

##### <制御部>

次に、図 1 8 1 を用いて、このパチンコ機 5 1 0 0 の制御部の回路構成について詳細に説明する。なお、同図は制御部の回路ブロック図を示したものである。

#### 【1 3 3 1】

パチンコ機 5 1 0 0 の制御部は、大別すると、遊技の中枢部分を制御する主制御部 5 3 0 0 と、主制御部 5 3 0 0 が送信するコマンド信号（以下、単に「コマンド」と呼ぶ）に応じて、主に演出の制御を行う副制御部 5 4 0 0 と、主制御部 5 3 0 0 が送信するコマンドに応じて、主に遊技球の払い出しに関する制御を行う払出制御部 5 5 5 0 と、遊技球の発射制御を行う発射制御部 5 6 0 0 と、パチンコ機 5 1 0 0 に供給される電源を制御する電源管理部 5 6 5 0 によって構成している。

#### 【1 3 3 2】

##### <主制御部>

まず、パチンコ機 5 1 0 0 の主制御部 5 3 0 0 について説明する。主制御部 5 3 0 0 は、主制御部 5 3 0 0 の全体を制御する基本回路 5 3 0 2 を備えており、この基本回路 5 3 0 2 には、CPU 5 3 0 4 と、制御プログラムや各種データを記憶するための ROM 5 3 0 6 と、一時的にデータを記憶するための RAM 5 3 0 8 と、各種デバイスの入出力を制

御するための I/O 5310 と、時間や回数等を計測するためのカウンタタイマ 5312 と、WDT（ウォッチドッグタイマ）5313 を搭載している。なお、ROM 5306 や RAM 5308 については他の記憶手段を用いてもよく、この点は後述する副制御部 5400 についても同様である。この基本回路 5302 の CPU 5304 は、水晶発振器 5314b が出力する外部クロックをクロック回路 5340（図 182 参照。詳細は後述）で所定の分周比（この例では、1/2）で分周したシステムクロック SCLK を、基本クロックとして入力して動作する。

#### 【1333】

また、基本回路 5302 には、水晶発振器 5314a が出力する乱数用クロック RCK を受信する度に 0~65535 の範囲で数値を変動させるハードウェア乱数カウンタとして使用しているカウンタ回路（乱数回路）5316 を搭載している（詳細は後述する）。

10

#### 【1334】

また、基本回路 5302 には、各始動口、入賞口の入り口および可変入賞口の内部に設けた球検出センサを含む各種センサ 5318 が出力する信号を受信し、増幅結果や基準電圧との比較結果を基本回路 5302 に出力するためのセンサ回路 5320 と、特図表示装置 5114 の表示制御を行うための表示回路 5322 と、普図表示装置 5112 の表示制御を行うための表示回路 5324 と、各種状態表示部 5326（普図保留ランプ 5116、特図保留ランプ 5118、高確中ランプ 5118 等）の表示制御を行うための表示回路 5328 と、第 2 特図始動口 5128 や可変入賞口 5130 等を開閉駆動する各種ソレノイド 5330 を制御するためのソレノイド回路 5332 と、電源が投入されると CPU 5304 に対して起動信号（リセット信号）を出力する起動信号出力回路（リセット信号出力回路）5338 を接続している。

20

#### 【1335】

なお、第 1 特図始動口 5126 に球が入賞したことを球検出センサ 5318 が検出した場合には、センサ回路 5320 は球を検出したことを示す信号をカウンタ回路 5316 に出力する。この信号を受信したカウンタ回路 5316 は、第 1 特図始動口 5126 に対応するカウンタのそのタイミングにおける値をラッチし、ラッチした値を、第 1 特図始動口 5126 に対応する内蔵のカウンタ値記憶用レジスタに記憶する。また、カウンタ回路 5316 は、第 2 特図始動口 5128 に球が入賞したことを示す信号を受信した場合も同様に、第 2 特図始動口 5128 に対応するカウンタのそのタイミングにおける値をラッチし、ラッチした値を、第 2 特図始動口 5128 に対応する内蔵のカウンタ値記憶用レジスタに記憶する。また、カウンタ回路 5316 は、普図始動口 5124 に球が入賞したことを示す信号を受信した場合も同様に、普図始動口 5124 に対応するカウンタのそのタイミングにおける値をラッチし、ラッチした値を、普図始動口 5124 に対応する内蔵のカウンタ値記憶用レジスタに記憶する。

30

#### 【1336】

さらに、基本回路 5302 には、情報出力回路 5334 を接続しており、主制御部 5300 は、この情報出力回路 5334 を介して、外部のホールコンピュータ（図示省略）等が備える情報入力回路 5652 にパチンコ機 5100 の遊技情報（例えば、遊技状態）を出力する。

40

#### 【1337】

また、主制御部 5300 は、副制御部 5400 にコマンドを送信するための出力インタフェースと、払出制御部 5550 にコマンドを送信するための出力インタフェースをそれぞれ備えており、この構成により、副制御部 5400 および払出制御部 5550 との通信を可能としている。なお、主制御部 5300 と副制御部 5400 および払出制御部 5550 との情報通信は一方方向の通信であり、主制御部 5300 は副制御部 5400 および払出制御部 5550 にコマンド等の信号を送信できるように構成しているが、副制御部 5400 および払出制御部 5550 からは主制御部 5300 にコマンド等の信号を送信できないように構成している。

#### 【1338】

50

### ＜副制御部＞

次に、パチンコ機 5100 の副制御部 5400 について説明する。副制御部 5400 は、主に主制御部 5300 が送信したコマンド等に基づいて副制御部 5400 の全体を制御する基本回路 5402 を備えており、この基本回路 5402 には、CPU 5404 と、制御プログラムや各種データを記憶するための ROM 5406 と、一時的にデータを記憶するための RAM 5408 と、各種デバイスの入出力を制御するための I/O 5410 と、時間や回数等を計測するためのカウンタタイマ 5412 を搭載している。この基本回路 5402 の CPU 5404 は、水晶発振器 5414 が出力する所定周期のクロック信号をシステムクロックとして入力して動作する。

#### 【1339】

また、基本回路 5402 には、スピーカ 5416（およびアンプ）の制御を行うための音源 IC 5418 と、各種ランプ 5420 の制御を行うための表示回路 5422 と、演出装置 5200 の演出用可動体等を駆動する駆動装置であるソレノイドまたはモータ等が含まれる各種演出用駆動装置 5424 の制御を行うための演出用駆動装置制御回路 5426 と、装飾図柄表示装置（液晶表示装置）5110 および遮蔽手段 5250 の制御を行うための副制御部 5500 と、チャンスボタン 5146 の押下を検出して信号を出力するチャンスボタン検出回路 5380 を接続している。

#### 【1340】

### ＜払出制御部、発射制御部、電源管理部＞

次に、パチンコ機 5100 の払出制御部 5550、発射制御部 5600、電源管理部 5650 について説明する。払出制御部 5550 は、主に主制御部 5300 が送信したコマンド等の信号に基づいて払出装置 5552 を制御すると共に、払出センサ 5554 が出力する制御信号に基づいて賞球または貸球の払い出しが完了したか否かを検出すると共に、インタフェース部 5556 を介して、パチンコ機 5100 とは別体で設けられたカードユニット 5654 との通信を行う。

#### 【1341】

発射制御部 5600 は、払出制御部 5550 が出力する、発射許可または停止を指示する制御信号や、操作ハンドル 5148 内に設けた発射強度出力回路が出力する、遊技者による発射ハンドル 5148 の操作量に応じた発射強度を指示する制御信号に基づいて、発射杆 5138 および発射槌 5140 を駆動する発射モータ 5602 の制御や、貯留皿 5144 から発射レール 5142 に球を供給する球送り装置 5604 の制御を行う。

#### 【1342】

電源管理部 5650 は、パチンコ機 5100 に外部から供給される交流電源を直流化し、所定の電圧に変換して主制御部 5300、副制御部 5400 等の各制御部や払出装置 5552 等の各装置に供給する。さらに、電源管理部 5650 は、外部からの電源が断たれた後も所定の部品（例えば主制御部 5300 の RAM 5308 等）に所定の期間（例えば 10 日間）電源を供給するための蓄電回路（例えばコンデンサ）を備えている。

#### 【1343】

### ＜主制御部の基本回路＞

次に、図 182 を用いて、主制御部 5300 の基本回路 5302 について詳細に説明する。なお、同図は基本回路の内部構成図である。

#### 【1344】

基本回路 5302 は、上述の CPU 5304、ROM（内蔵 ROM）5306、RAM（内蔵 RAM）5308、カウンタタイマ 5312（タイマ回路 5312a、カウンタ回路 5312b）、カウンタ回路（乱数回路）5316、I/O 5310（外部バス制御回路 5310a、WDT 5313 内蔵のリセット制御回路 5310b、パラレル入力ポート 5310c、アドレスデコード回路 5310d）、に加えて、クロック回路 5340 などを備える。

#### 【1345】

クロック回路 5340 は、上述の水晶発振器 5314b（以下、システム用水晶発振器

10

20

30

40

50

5 3 1 4 bと称する場合がある) から EX 端子を介して入力される外部クロック EX (この例では、2 4 M H z のクロック) を所定の分周比 (この例では、1 / 2) で分周し、分周後のシステムクロック S C L K (この例では、1 2 M H z のクロック) を C P U コアや内部の各回路に供給する回路である。乱数回路 5 3 1 6 は、詳細は後述するが、上述の水晶発振器 5 3 1 4 a (以下、乱数用水晶発振器 5 3 1 4 a と称する場合がある) から R C K 端子を介して入力される乱数用クロック R C K (この例では、1 0 M H z のクロック) に基づいてカウント値のカウントを行って乱数値を発生させるための回路である。なお、本実施例では、乱数用クロック R C K の周波数を、システムクロック S C L K の周波数 (この例では、1 2 M H z) 未満であって、後述する主制御部タイマ割り込み処理の周期としてカウンタ・タイマ 5 3 1 2 に設定する周期 (この例では 2 m s) 以上の周波数に設定している。

#### 【1 3 4 6】

図 1 8 3 は、基本回路 5 3 0 2 に接続される乱数用水晶発振器 5 3 1 4 a とシステム用水晶発振器 5 3 1 4 b の配線パターンの一例を示した回路ブロック図である。

#### 【1 3 4 7】

本実施例では、基本回路 5 3 0 2 とシステム用水晶発振器 5 3 1 4 b を結ぶ信号線 L s の長さを、基本回路 5 3 0 2 と乱数用水晶発振器 5 3 1 4 a を結ぶ信号線 L r の長さよりも短く設定している ( $L r > L s$ )。このような構成により、システム用水晶発振器 5 3 1 4 b を基本回路 5 3 0 2 の近傍に配置することができ、且つ、基本回路 5 3 0 2 とシステム用水晶発振器 5 3 1 4 b を結ぶ信号線 L s の長さを短くすることができるため、システム用水晶発振器 5 3 1 4 b から出力される外部クロック EX の信号を、安定して基本回路 5 3 0 2 のクロック回路 5 3 4 0 に供給するとともに、この外部クロック EX の信号に基づいて生成されるシステムクロック S C L K の信号も、安定して C P U コアや内部の各回路に供給することができる。

#### 【1 3 4 8】

また、クロック信号の供給源と供給先を結ぶ信号線の長さは、クロック信号に外乱が加わらないようになるべく短く設定するのが一般的であるが、基本回路 5 3 0 2 と乱数用水晶発振器 5 3 1 4 a を結ぶ信号線 L r の長さを敢えて長めに設定し、信号線 L r を通る乱数用クロック R C K の信号が外乱の影響を受けやすいように構成することによって、乱数用クロック R C K の信号に基づいてカウント値のカウントを行う乱数回路 5 3 1 6 のカウント値の更新タイミングにバラツキを与え、乱数回路 5 3 1 6 が生成する乱数値のランダム性をさらに高めることができる。

#### 【1 3 4 9】

さらに、基本回路 5 3 0 2 とシステム用水晶発振器 5 3 1 4 b を結ぶ信号線 L s の長さと、基本回路 5 3 0 2 と乱数用水晶発振器 5 3 1 4 a を結ぶ信号線 L r の長さを異ならせることにより、基本回路 5 3 0 2 近傍の部品の配置の自由度を高めることができる。

#### 【1 3 5 0】

##### < 乱数回路 >

次に、図 1 8 4 および図 1 8 5 を用いて、基本回路 5 3 0 2 が備える乱数回路 (カウンタ回路) 5 3 1 6 について詳細に説明する。なお、図 1 8 4 は乱数回路の内部構成図であり、図 1 8 5 は乱数回路が備える周波数監視回路の内部構成図である。

#### 【1 3 5 1】

乱数回路 5 3 1 6 は、乱数回路 C H 1 ~ C H 4 の 4 つの乱数回路を備える。なお、乱数回路 C H 2 ~ 4 の内部構成は、乱数回路 C H 1 と同一であるため、図示は省略している。

#### 【1 3 5 2】

乱数回路 C H 1 ~ C H 4 は、乱数更新回路 5 3 1 6 g に取り込まれた乱数値を格納するための乱数レジスタ 5 3 1 6 b と、乱数レジスタ 5 3 1 6 b に乱数値が取り込まれたことを示すための乱数ラッチフラグレジスタ 5 3 1 6 c と、乱数割り込みを制御するための乱数割り込み制御レジスタ 5 3 1 6 d と、乱数更新回路 5 3 1 6 g において乱数値が正常に更新されたかどうかを更新毎に監視し、更新に異常があった場合に乱数値異常信号を出力する

乱数監視回路 5316e と、この乱数監視回路 5316e から乱数値異常信号が入力された場合に乱数値異常信号状態ビットを 1 にセットする内部情報レジスタ 5316f と、を有して構成されている。なお、乱数監視回路 5316e は、乱数更新回路 5316g に内蔵してもよいし、乱数回路 CH1～CH4 で共通に設けてもよい（共通に設ける場合には、CH1～CH4 が識別できるようにフラグを設けてもよいし、フラグを共通に設けてもよい）。

#### 【1353】

この乱数回路 5316 は、各チャネル毎に異なった乱数列を持つ 2 種類の 16 ビット乱数値を発生させることが可能であるとともに、ROM 5306 の所定領域に設けた乱数使用設定、乱数初期設定、乱数取込設定の内容を変更することによって、乱数回路 CH1～CH4 の使用／未使用、初期値・乱数列変更方法の選択、乱数値の更新周期、乱数値の取り込み方法、割込条件などを変更することが可能である。本実施例では、P0 端子は第 1 特図始動口 5126 に入球があった場合に出力される特図 1 始動口入球検出信号、P1 端子は第 2 特図始動口 5128 に入球があった場合に出力される特図 2 始動口入球検出信号、P2 端子は普図始動口 5124 に入球があった場合に出力される普図始動口入球検出信号がそれぞれ入力されるようにセンサ回路 5320 と接続されているが、これに限定されない。

#### 【1354】

また、乱数回路 CH1～CH4 は、周波数監視回路 5316a を備えている。この周波数監視回路 5316a は、図 185 に拡大して示すように、上述のクロック回路 5340 から出力されるシステムクロック SCLK が入力される平滑回路と、上述の水晶発振器 5314a から RCK 端子を介して入力される乱数用クロック RCK が入力される平滑回路と、平滑後のシステムクロック SCLK の周波数と乱数用クロック RCK の周波数を比較し、RCK の周波数がシステムクロック SCLK の周波数以下 ( $RCK < SCLK$ ) である場合に、乱数用クロック RCK の周波数が異常になったと判定し内部情報レジスタのクロック信号状態ビットを 1 にセットする比較器と、を備えている。

#### 【1355】

主制御部 5300 の CPU 5304 は、所定のタイミングで、内部情報レジスタの乱数値異常信号状態ビットとクロック信号状態ビットを参照する。そして、乱数値異常信号状態ビットが 1 にセットされている場合（乱数更新回路 5316g において乱数値の更新に異常があった場合）には、乱数更新回路 5316g に何らかの異常が発生したと判定し、乱数値異常信号状態ビットが 0 にセットされている場合には、乱数更新回路 5316g が正常であると判定する。

#### 【1356】

また、内部情報レジスタのクロック信号状態ビットが 1 にセットされている場合（乱数用クロック RCK の周波数がシステムクロック SCLK の周波数以下 ( $RCK < SCLK$ ) である場合）には、乱数用クロック RCK に何らかの異常（例えば、乱数回路 5316 と水晶発振器 5314a とを結ぶ配線パターンの断線）が発生したと判定し、内部情報レジスタのクロック信号状態ビットが 0 にセットされている場合（乱数用クロック RCK の周波数がシステムクロック SCLK の周波数より大きい ( $RCK > SCLK$ ) 場合）には、乱数用クロック RCK が正常であると判定する。

#### 【1357】

なお、本実施例では、乱数用クロック RCK の周波数がシステムクロック SCLK の周波数以下 ( $RCK < SCLK$ ) である場合に乱数用クロック RCK の異常と判定しているが、本発明はこれに限定されず、例えば、異常と判定する閾値を複数種類設け、この複数種類の閾値の中から 1 つの閾値を選択可能に構成してもよい。また、システムクロック SCLK の周波数や、乱数用クロック RCK の周波数は、本実施例で示した数値に限定されるものではない。たとえば、所定の情報（たとえば ROM に記憶された情報）に基づいて、 $RCK < 2 \times SCLK$ 、 $RCK < 1/2 \times SCLK$  など、異常の判定方式を設定できるようにしてもよい。なお、理想的には、12MHz のシステムクロック SCLK に



対して、10MHzの乱数用クロックRCKが3MHz以下（システムクロックの1/4以下）の周波数になった場合に異常とすることが望ましい。

#### 【1358】

##### ＜主制御部メイン処理＞

次に、図186を用いて、主制御部5300のCPU5304が実行する主制御部メイン処理について説明する。なお、同図は主制御部メイン処理の流れを示すフローチャートである。

#### 【1359】

上述したように、主制御部5300には、電源が投入されると起動信号（リセット信号）を出力する起動信号出力回路（リセット信号出力回路）5338を設けている。この起動信号を入力した基本回路5302のCPU5304は、リセット割り込みによりリセットスタートしてROM5306に予め記憶している制御プログラムに従って処理を実行する。

10

#### 【1360】

ステップS6101では、初期設定1を行う。この初期設定1では、CPU5304のスタックポインタ（SP）へのスタック初期値の設定、割り込みマスクの設定、I/Oポート5310の初期設定、RAM5308に記憶する各種変数の初期設定、WDT5313への動作許可及び初期値の設定等を行う。なお、本実施例では、WDT5313に、初期値として32.8msに相当する数値を設定する。

#### 【1361】

ステップS6102では、WDT5313のカウンタの値をクリアし、WDT5313による時間計測を再始動する。

20

#### 【1362】

ステップS6103では、低電圧信号がオンであるか否か、すなわち、電圧監視回路5336が、電源管理部5650から主制御部5300に供給している電源の電圧値が所定の値（本実施例では9V）未満である場合に電圧が低下したことを示す低電圧信号を出力しているか否かを監視する。そして、低電圧信号がオンの場合（CPU5304が電源の遮断を検知した場合）にはステップS6102に戻り、低電圧信号がオフの場合（CPU5304が電源の遮断を検知していない場合）にはステップS6104に進む。

#### 【1363】

ステップS6104では、初期設定2を行う。この初期設定2では、後述する主制御部タイマ割り込み処理を定期毎に実行するための周期を決める数値をカウンタ・タイマ5312に設定する処理、I/O5310の所定のポート（例えば試験用出力ポート、副制御部5400への出力ポート）からクリア信号を出力する処理、RAM5308への書き込みを許可する設定等を行う。

30

#### 【1364】

ステップS6105では、電源の遮断前（電断前）の状態に復帰するか否かの判定を行い、電断前の状態に復帰しない場合（主制御部5300の基本回路5302を初期状態にする場合）にはステップS6107に進む。同様に電源ステータスの情報が「サスペンド（電断時処理が行われたことを示す情報）」以外の情報を示している場合にもステップS6108に進む。

40

#### 【1365】

具体的には、最初に、電源基板に設けた操作部を遊技店の店員などが操作した場合に送信されるRAMクリア信号がオン（操作があったことを示す）であるか否か、すなわちRAMクリアが必要であるか否かを判定し、RAMクリア信号がオンの場合（RAMクリアが必要な場合）には、基本回路5302を初期状態にすべくステップS6107に進む。一方、RAMクリア信号がオフの場合（RAMクリアが必要でない場合）は、RAM5308に設けた電源ステータス記憶領域に記憶した電源ステータスの情報を読み出し、この電源ステータスの情報がサスペンドを示す情報であるか否かを判定する。そして、電源ステータスの情報がサスペンドを示す情報でない場合には、基本回路5302を初期状態に

50

すべくステップ S 6 1 0 7 に進み、電源ステータスの情報がサスペンドを示す情報である場合には、R A M 5 3 0 8 の所定の領域（例えば全ての領域）に記憶している 1 バイトデータを初期値が 0 である 1 バイト構成のレジスタに全て加算することによりチェックサムを算出し、算出したチェックサムの結果が特定の値（例えば 0）であるか否か（チェックサムの結果が正常であるか否か）を判定する。そして、チェックサムの結果が特定の値（例えば 0）の場合（チェックサムの結果が正常である場合）には電断前の状態に復帰すべくステップ S 6 1 0 6 に進み、チェックサムの結果が特定の値（例えば 0）以外である場合（チェックサムの結果が異常である場合）には、パチンコ機 5 1 0 0 を初期状態にすべくステップ S 6 1 0 7 に進む。同様に電源ステータスの情報が「サスペンド」以外の情報を示している場合にもステップ S 6 1 0 7 に進む。

10

#### 【1 3 6 6】

ステップ S 6 1 0 6 では、復電時処理を行う。この復電時処理では、電断時に R A M 5 3 0 8 に設けられたスタックポインタ退避領域に記憶しておいたスタックポインタを読み出し、スタックポインタに再設定する。また、電断時に R A M 5 3 0 8 に設けられたレジスタ退避領域に記憶しておいた各レジスタの値を読み出し、各レジスタに再設定した後、割り込み許可の設定を行う。以降、C P U 5 3 0 4 が、再設定後のスタックポインタやレジスタに基づいて制御プログラムを実行する結果、パチンコ機 5 1 0 0 は電源断時の状態に復帰する。すなわち、電断直前にタイマ割り込み処理（後述）に分岐する直前に行った（ステップ S 6 1 0 8、ステップ S 6 1 0 9 内の所定の）命令の次の命令から処理を再開する。

20

#### 【1 3 6 7】

ステップ S 6 1 0 7 では、初期化処理を行う。この初期化処理では、割り込み禁止の設定、スタックポインタへのスタック初期値の設定、R A M 5 3 0 8 の全ての記憶領域の初期化などを行う。

#### 【1 3 6 8】

ステップ S 6 1 0 8 では、割り込み禁止の設定を行った後、基本乱数更新処理を行う。この基本乱数更新処理では、普図タイマ乱数値、特図タイマ乱数値をそれぞれ生成するための 2 つの乱数カウンタを更新する。例えば、普図タイマ乱数値として取り得る数値範囲が 0 ~ 2 0 とすると、R A M 5 3 0 8 に設けた普図タイマ乱数値を生成するための乱数カウンタ記憶領域から値を取得し、取得した値に 1 を加算してから元の乱数カウンタ記憶領域に記憶する。このとき、取得した値に 1 を加算した結果が 2 1 であれば 0 を元の乱数カウンタ記憶領域に記憶する。他の乱数カウンタもそれぞれ同様に更新する。また、この基本乱数更新処理の終了後に割り込み許可の設定を行ってステップ S 6 1 0 9 に進む。

30

#### 【1 3 6 9】

ステップ S 6 1 0 9 では、演出乱数更新処理を行う。この演出乱数更新処理では、主制御部 5 3 0 0 で使用する演出用乱数値を生成するための乱数カウンタを更新する。

#### 【1 3 7 0】

主制御部 5 3 0 0 は、所定の周期ごとに開始するタイマ割り込み処理を行っている間を除いて、ステップ S 6 1 0 8 および S 6 1 0 9 の処理を繰り返し実行する。

#### 【1 3 7 1】

＜主制御部タイマ割り込み処理＞

次に、図 1 8 7 を用いて、主制御部 5 3 0 0 の C P U 5 3 0 4 が実行する主制御部タイマ割り込み処理について説明する。なお、同図は主制御部タイマ割り込み処理の流れを示すフローチャートである。

40

#### 【1 3 7 2】

主制御部 5 3 0 0 は、所定の周期（本実施例では約 2 m s に 1 回）でタイマ割り込み信号を発生するカウンタ・タイマ 5 3 1 2 を備えており、このタイマ割り込み信号を契機として主制御部タイマ割り込み処理を所定の周期で開始する。

#### 【1 3 7 3】

ステップ S 6 2 0 1 では、タイマ割り込みスタート処理を行う。このタイマ割り込みス

50

タート処理では、CPU 5304の各レジスタの値をスタック領域に一時的に退避する処理などを行う。

【1374】

ステップS6202では、WDT 5313のカウント値が初期設定値（本実施例では32.8ms）を超えてWDT割り込みが発生しないように（処理の異常を検出しないように）、WDTを定期的に（本実施例では、主制御部タイマ割り込みの周期である約2msに1回）リスタートを行う。

【1375】

ステップS6203では、入力ポート状態更新処理を行う。この入力ポート状態更新処理では、I/O 5310の入力ポートを介して、上述のガラス枠5154が開放状態または閉鎖状態のいずれの状態であるかを検出するための開放センサ、上述の下皿5150が球で一杯になったか否かを検出するための下皿満タンセンサ、および複数の球検出センサを含む各種センサ5318の検出信号を入力して検出信号の有無を監視し、RAM 5308に各種センサ5318ごとに区画して設けた信号状態記憶領域に記憶する。本実施例では、前々回のタイマ割り込み処理（約4ms前）で検出した各々の球検出センサの検出信号の有無の情報を、RAM 5308に各々の球検出センサごとに区画して設けた前回検出信号記憶領域から読み出し、この情報をRAM 5308に各々の球検出センサごとに区画して設けた前々回検出信号記憶領域に記憶し、前回のタイマ割り込み処理（約2ms前）で検出した各々の球検出センサの検出信号の有無の情報を、RAM 5308に各々の球検出センサごとに区画して設けた今回検出信号記憶領域から読み出し、この情報を上述の前回検出信号記憶領域に記憶する。また、今回検出した各々の球検出センサの検出信号を、上述の今回検出信号記憶領域に記憶する。

【1376】

ステップS6204およびステップS6205では、大当り種別用乱数更新処理および基本乱数更新処理を行う。これらの大当り種別用乱数更新処理および基本乱数更新処理では、次に主制御部5300で使用する大当り種別用乱数等を更新する。なお、更新の方法は上述のステップS6108およびS6109と同様である。

【1377】

ステップS6206では、演出乱数更新処理を行う。この演出乱数更新処理では、主制御部5300で使用する演出用乱数値を生成するための乱数カウンタを更新する。

【1378】

ステップS6207では、タイマ更新処理を行う。このタイマ更新処理では、普通図柄表示装置5112に図柄を変動・停止表示する時間を計時するための普図表示図柄更新タイマ、特別図柄表示装置5114に図柄を変動・停止表示する時間を計時するための特図表示図柄更新タイマ、所定の入賞演出時間、所定の開放時間、所定の閉鎖時間、所定の終了演出期間などを計時するためのタイマなどを含む各種タイマを更新する。

【1379】

ステップS6208では、入賞口カウンタ更新処理を行う。この入賞口カウンタ更新処理では、入賞口（一般入賞口5122、第1、第2特図始動口5126、5128、および可変入賞口5130）に入賞（入球）があった場合に、RAM 5308に各入賞口ごとに設けた賞球数記憶領域の値を読み出し、1を加算して、元の賞球数記憶領域に設定する。なお、乱数用クロックRCKが異常であっても入賞に対する賞球払出数の増加を制限する理由がないため、後述するステップS6209のように乱数用クロックRCKに異常がある場合でも特別な処理を行わないが、不正者に対する懲罰的な意味で乱数用クロックRCKに異常がある場合に賞球数を加算しない等の所定の処理を行ってもよい。また、次のステップS6209では、入賞受付処理を行う（詳細は後述する）。ステップS6210では、払出要求数送信処理を行う。なお、払出制御部550に出力する出力予定情報および払出要求情報は1バイトで構成しており、ビット7にストロブ情報（オンの場合、データをセットしていることを示す）、ビット6に電源投入情報（オンの場合、電源投入後一回目のコマンド送信であることを示す）、ビット4～5に今回加工種別（0～3）、お

10

20

30

40

50

よびビット 0 ～ 3 に加工後の払出要求数を示すようにしている。

#### 【1380】

ステップ S 6 2 1 1 では、普図状態更新処理を行う。この普図状態更新処理は、普図の状態に対応する複数の処理のうちの 1 つの処理を行う。例えば、普図変動中（後述する普図汎用タイマの値が 1 以上）における普図状態更新処理では、普図表示装置 5 1 1 2 を構成する 7 セグメント L E D の点灯と消灯を繰り返す点灯・消灯駆動制御を行う。

#### 【1381】

また、普図変動表示時間が経過したタイミング（普図表示図柄更新タイマの値が 1 から 0 になったタイミング）における普図状態更新処理では、当りフラグがオンの場合には、上述の普図 1 の態様となるように普図表示装置 5 1 1 2 を構成する 7 セグメント L E D の点灯・消灯駆動制御を行い、当りフラグがオフの場合には、上述の普図 2 の態様となるように普図表示装置 5 1 1 2 を構成する 7 セグメント L E D の点灯・消灯駆動制御を行うと共に、その後、所定の停止表示期間（例えば 5 0 0 m 秒間）その表示を維持するために R A M 5 3 0 8 に設けた普図停止時間管理用タイマの記憶領域に停止期間を示す情報を設定する。この設定により普図の停止表示を行い、普図変動遊技の結果を遊技者に報知するようにしている。

#### 【1382】

また、所定の停止表示期間が終了したタイミング（普図停止時間管理用タイマの値が 1 から 0 になったタイミング）で開始する普図状態更新処理では、当りフラグがオンの場合には、所定の開放期間（例えば 2 秒間）、第 2 特図始動口 5 1 2 8 の羽根部材の開閉駆動用のソレノイド 5 3 3 0 に、羽根部材を開放状態に保持する信号を出力するとともに、R A M 5 3 0 8 に設けた羽根開放時間管理用タイマの記憶領域に開放期間を示す情報を設定する。

#### 【1383】

また、所定の開放期間が終了したタイミング（羽根開放時間管理用タイマの値が 1 から 0 になったタイミング）で開始する普図状態更新処理では、所定の閉鎖期間（例えば 5 0 0 m 秒間）、羽根部材の開閉駆動用のソレノイド 5 3 3 0 に、羽根部材を閉鎖状態に保持する信号を出力するとともに、R A M 5 3 0 8 に設けた羽根閉鎖時間管理用タイマの記憶領域に閉鎖期間を示す情報を設定する。

#### 【1384】

また、所定の閉鎖期間を経過したタイミング（羽根閉鎖時間管理用タイマの値が 1 から 0 になったタイミング）で開始する普図状態更新処理では、普図の状態を非作動中に設定する。普図の状態が非作動中の場合における普図状態更新処理では、何もせずに次のステップ S 6 2 1 2 に移行するようにしている。

#### 【1385】

ステップ S 6 2 1 2 では、普図関連抽選処理を行う。この普図関連抽選処理では、普図変動遊技および第 2 特図始動口 5 1 2 8 の開閉制御を行っておらず（普図の状態が非作動中）、且つ、保留している普図変動遊技の数が 1 以上である場合に、上述の乱数値記憶領域に記憶している普図当選乱数値に基づいた乱数抽選により普図変動遊技の結果を当選とするか、不当選とするかを決定する当り判定をおこない、当選とする場合には R A M 5 3 0 8 に設けた当りフラグにオンを設定する。不当選の場合には、当りフラグにオフを設定する。また、当り判定の結果に関わらず、次に上述の普図タイマ乱数値生成用の乱数カウンタの値を普図タイマ乱数値として取得し、取得した普図タイマ乱数値に基づいて複数の変動時間のうちから普図表示装置 5 1 1 2 に普図を変動表示する時間を 1 つ選択し、この変動表示時間を、普図変動表示時間として、R A M 5 3 0 8 に設けた普図変動時間記憶領域に記憶する。なお、保留している普図変動遊技の数は、R A M 5 3 0 8 に設けた普図保留数記憶領域に記憶するようにしており、当り判定をするたびに、保留している普図変動遊技の数から 1 を減算した値を、この普図保留数記憶領域に記憶し直すようにしている。また当り判定に使用した乱数値を消去する。

#### 【1386】

ステップS 6 2 1 3では、特図状態更新処理を行う（詳細は後述する）。ステップS 6 2 1 4では、特図関連抽選処理を行う。この特図関連抽選処理では、特図変動遊技および可変入賞口5 1 3 0の開閉制御を行っておらず（特図の状態が非作動中）、且つ、保留している特図変動遊技の数が1以上である場合に、大当たり判定テーブル、高確率状態移行判定テーブル、タイマ番号決定テーブルなどを使用した各種抽選のうち、最初に大当たり判定を行う。具体的には、ステップS 6 2 0 3で乱数値記憶領域に記憶した特図当選乱数値が、大当たり判定テーブルの第1特図始動口用抽選データの数値範囲であるか否かを判定し、特図当選乱数値が第1特図始動口用抽選データの数値範囲である場合には、特図変動遊技の当選と判定してRAM 5 3 0 8に設けた大当たりフラグの格納領域に大当たりとなることを示す情報を設定する（ここで、大当たりの情報をRAM 5 3 0 8に設定することを大当たりフラグをオンに設定するという）。一方、特図当選乱数値が第1特図始動口用抽選データの数値範囲以外である場合には、特図変動遊技の外れと判定してRAM 5 3 0 8に設けた大当たりフラグの格納領域に外れとなることを示す情報を設定する（ここで、外れの情報をRAM 5 3 0 8に設定することを大当たりフラグをオフに設定するという）。なお、保留している特図変動遊技の数は、RAM 5 3 0 8に設けた特図保留数記憶領域に記憶するようにしており、当り判定をするたびに、保留している特図変動遊技の数から1を減算した値を、この特図保留数記憶領域に記憶し直すようにしている。また、当り判定に使用した乱数値を消去する。

10

#### 【1 3 8 7】

大当たりフラグにオンを設定した場合には、次に確変移行判定を行う。具体的には、大当たり種別用乱数が、移行判定乱数の数値範囲であるか否かを判定し、特図乱数値が抽選データの数値範囲である場合には、RAM 5 3 0 8に設けた確変（確率変動）フラグの格納領域に、特別大当たり遊技を開始することを示す情報を設定する。（ここで、特別大当たり遊技開始の情報をRAM 5 3 0 8に設定することを確変フラグをオンに設定するという）。一方、大当たり種別用乱数が抽選データの数値範囲以外である場合には、上述の確変フラグの格納領域に、大当たり遊技を開始することを示す情報を設定する（ここで、大当たり遊技開始の情報をRAM 5 3 0 8に設定することを確変フラグをオフに設定するという）。

20

#### 【1 3 8 8】

大当たり判定の結果に関わらず、次にタイマ番号を決定する処理を行う。具体的には、上述の特図タイマ乱数値生成用の乱数カウンタの値を特図タイマ乱数値として取得する。大当たりフラグの値、および取得した特図タイマ乱数値を含むタイマ乱数の数値範囲に対応するタイマ番号を選択し、RAM 5 3 0 8に設けた所定のタイマ番号格納領域に記憶する。さらに、そのタイマ番号に対応する変動時間を、特図変動表示時間として、上述の特図表示図柄更新タイマに記憶し、コマンド設定送信処理（ステップS 6 2 1 5）で一般コマンド回転開始設定送信処理を実行させるために上述の送信情報記憶領域に0 1 Hを送信情報（一般情報）として追加記憶してから処理を終了する。

30

#### 【1 3 8 9】

ステップS 6 2 1 5では、コマンド設定送信処理を行う。なお、副制御部5 4 0 0および払出制御部5 5 0に送信する出力予定情報（コマンド）は1 6ビットで構成しており、ビット1 5はストロブ情報（オンの場合、データをセットしていることを示す）、ビット1 1～1 4はコマンド種別（0 0 Hの場合は基本コマンド、0 1 Hの場合は図柄変動開始コマンド、0 4 Hの場合は図柄変動停止コマンド、0 5 Hの場合は入賞演出開始コマンド、0 6 Hの場合は終了演出開始コマンド、0 7 Hの場合は大当たりラウンド数指定コマンド、0 E Hの場合は復電コマンド、0 F Hの場合はRAMクリアコマンド、1 0 Hの場合は乱数クロック異常コマンドをそれぞれ示すなど、コマンドの種類を特定可能な情報）、ビット0～1 0はコマンドデータ（コマンド種別に対応する所定の情報）で構成している。

40

#### 【1 3 9 0】

具体的には、ストロブ情報はコマンド送信処理でオン、オフするようにしている。また、コマンド種別が図柄変動開始コマンドの場合であればコマンドデータに、大当たりフラ

50

グの値、確変フラグの値、特図関連抽選処理で選択したタイマ番号などを示す情報を含み、図柄変動停止コマンドの場合であれば、大当たりフラグの値、確変フラグの値などを含み、入賞演出コマンドおよび終了演出開始コマンドの場合であれば、確変フラグの値などを含み、大当たりラウンド数指定コマンドの場合であれば確変フラグの値、大当たりラウンド数などを含むようにしている。コマンド種別が基本コマンドを示す場合は、コマンドデータにデバイス情報、第1特図始動口5126への入賞の有無、第2特図始動口5128への入賞の有無、可変入賞口5130への入賞の有無などを含む。

#### 【1391】

また、上述の一般コマンド回転開始設定送信処理では、コマンド種別に01H、コマンドデータにRAM5308に記憶している大当たりフラグの値、確変フラグの値、特図関連抽選処理で選択したタイマ番号、保留している特図変動遊技の数などを示す情報を設定する。上述の一般コマンド回転停止設定送信処理では、コマンド種別に04H、コマンドデータにRAM5308に記憶している大当たりフラグの値、確変フラグの値などを示す情報を設定する。上述の一般コマンド入賞演出設定送信処理では、コマンド種別に05H、コマンドデータにRAM5308に記憶している入賞演出期間中に装飾図柄表示装置5110・各種ランプ5420・スピーカ5416に出力する演出制御情報、確変フラグの値、保留している特図変動遊技の数などを示す情報を設定する。上述の一般コマンド終了演出設定送信処理では、コマンド種別に06H、コマンドデータにRAM5308に記憶している演出待機期間中に装飾図柄表示装置5110・各種ランプ5420・スピーカ5416に出力する演出制御情報、確変フラグの値、保留している特図変動遊技の数などを示す情報を設定する。上述の一般コマンド大入賞口開放設定送信処理では、コマンド種別に07H、コマンドデータにRAM5308に記憶している大当たりラウンド数、確変フラグの値、保留している特図変動遊技の数などを示す情報を設定する。上述の一般コマンド大入賞口閉鎖設定送信処理では、コマンド種別に08H、コマンドデータにRAM5308に記憶している大当たりラウンド数、確変フラグの値、保留している特図変動遊技の数などを示す情報を設定する。副制御部5400では、受信した出力予定情報に含まれるコマンド種別により、主制御部5300における遊技制御の変化に応じた演出制御の決定が可能になるとともに、出力予定情報に含まれているコマンドデータの情報に基づいて、演出制御内容を決定することができるようになる。

#### 【1392】

ステップS6216では、外部出力信号設定処理を行う。この外部出力信号設定処理では、RAM5308に記憶している遊技情報を、情報出力回路5334を介してパチンコ機5100とは別体の情報入力回路5652に出力する。

#### 【1393】

ステップS6217では、デバイス監視処理を行う。このデバイス監視処理では、ステップS6203において信号状態記憶領域に記憶した各種センサの信号状態を読み出して、ガラス枠開放エラーの有無または下皿満タンエラーの有無などを監視し、ガラス枠開放エラーまたは下皿満タンエラーを検出した場合に、副制御部5400に送信すべき送信情報に、ガラス枠開放エラーの有無または下皿満タンエラーの有無を示すデバイス情報を設定する。また、各種ソレノイド5330を駆動して第2特図始動口5128や、可変入賞口5130の開閉を制御したり、表示回路5322、5324、5328を介して普図表示装置5112、特図表示装置5114、各種状態表示部5326などに出力する表示データを、I/O5310の出力ポートに設定する。また、払出要求数送信処理（ステップS6210）で設定した出力予定情報を出力ポート5310を介して副制御部5400に出力する。

#### 【1394】

ステップS6218では、低電圧信号がオンであるか否かを監視する。そして、低電圧信号がオンの場合（電源の遮断を検知した場合）にはステップS6220に進み、低電圧信号がオフの場合（電源の遮断を検知していない場合）にはステップS6219に進む。

#### 【1395】

10

20

30

40

50

ステップ S 6 2 1 9 では、タイマ割り込みエンド処理を行う。このタイマ割り込みエンド処理では、ステップ S 6 2 0 1 で一時的に退避した各レジスタの値を元の各レジスタに設定したり、割り込み許可の設定などを行う。

#### 【1396】

ステップ S 6 2 2 0 では、電源管理部 5 6 5 0 から主制御部 5 3 0 0 に供給している電源の電圧値を監視する電圧監視回路が、所定の値以下である場合に電圧が低下したことを示す電圧低下信号を出力しているか否か、すなわち電源の遮断を検知したか否かを監視し、電源の遮断を検知した場合には、復電時に電断時の状態に復帰するための特定の変数やスタックポインタを復帰データとして R A M 5 3 0 8 の所定の領域に退避し、入出力ポートの初期化等の電断処理を行う。また、電源ステータスを「サスペンド」に設定する。

10

#### 【1397】

##### <入賞受付処理>

次に、図 1 8 8 を用いて、上述の主制御部タイマ割り込み処理における入賞受付処理について説明する。なお、同図は入賞受付処理の流れを示すフローチャートである。

#### 【1398】

ステップ S 6 3 0 1 ～ステップ S 6 3 0 3 では、前々回第 1 特図始動口検出信号記憶領域、前回第 1 特図始動口検出信号記憶領域、および今回第 1 特図始動口検出信号記憶領域の各記憶領域に記憶した第 1 特図始動口球検出センサの検出信号の有無の情報を比較し、第 1 特図始動口球検出センサにおける過去 3 回分の検出信号の有無の情報が予め定めた入賞パターン情報と一致するか否かを判定する。そして、第 1 特図始動口球検出センサにおいて過去 3 回分の検出信号の有無の情報が、予め定めた入賞判定パターン情報（本実施例では、今回第 1 特図始動口検出信号がオン、前回第 1 特図始動口検出信号がオン、前々回第 1 特図始動口検出信号がオフであることを示す情報）と一致した場合に、第 1 特図始動口 5 1 2 6 への入球があったと判定する。例えば、第 1 特図始動口球検出センサにおいて過去 3 回分の検出信号の有無の情報が上述の入賞判定パターン情報と一致した場合には、第 1 特図始動口 5 1 2 6 への入球があったと判定し、以降の第 1 特図始動口 5 1 2 6 への入球に伴う処理を行うが、過去 3 回分の検出信号の有無の情報が上述の入賞判定パターン情報と一致しなかった場合には、以降の第 1 特図始動口 5 1 2 6 の入球に伴う処理を行わずに処理を終了する。

20

#### 【1399】

ステップ S 6 3 0 4 では、上述の内部情報レジスタのクロック信号状態ビットを参照し、乱数用クロックの周波数に異常があるか無いかを判定する。そして、乱数用クロックの周波数に異常がある場合にはステップ S 6 3 0 5 の処理を実行することなくステップ S 6 3 0 6 に進み、乱数用クロックの周波数に異常がない場合には、ステップ S 6 3 0 5 に進む。

30

#### 【1400】

ステップ S 6 3 0 5 では、カウンタ回路（乱数回路）5 3 1 6 から乱数を取得する。より具体的には、第 1 特図始動口 5 1 2 6 に入賞があり、且つ、保留している特図変動遊技の数が 4 未満である場合には、第 1 特図始動口 5 1 2 6 に対応するカウンタ回路 5 3 1 6 の乱数値レジスタから取り出した値を特図当選乱数値として取得、または取り出した値に「所定の加工」を行なった値を特図当選乱数値として取得する。また、上述の特図乱数値生成用の乱数カウンタから取り出した値を特図乱数値として取得、または取り出した値に「所定の加工」を行なった値を特図乱数値として取得し、R A M 5 3 0 8 に設けた乱数値記憶領域に特図当選乱数値と共に記憶する。

40

#### 【1401】

ステップ S 6 3 0 6 ～ステップ S 6 3 0 8 では、前々回第 2 特図始動口検出信号記憶領域、前回第 2 特図始動口検出信号記憶領域、および今回第 2 特図始動口検出信号記憶領域の各記憶領域に記憶した第 2 特図始動口球検出センサの検出信号の有無の情報を比較し、第 2 特図始動口球検出センサにおける過去 3 回分の検出信号の有無の情報が予め定めた入賞パターン情報と一致するか否かを判定する。そして、第 2 特図始動口球検出センサにおい

50

て過去3回分の検出信号の有無の情報が、予め定めた入賞判定パターン情報（本実施例では、今回第2特図始動口検出信号がオン、前回第2特図始動口検出信号がオン、前々回第2特図始動口検出信号がオフであることを示す情報）と一致した場合に、第2特図始動口5130への入球があったと判定する。例えば、第2特図始動口球検出センサにおいて過去3回分の検出信号の有無の情報が上述の入賞判定パターン情報と一致した場合には、第2特図始動口5128への入球があったと判定し、以降の第2特図始動口5128への入球に伴う処理を行うが、過去3回分の検出信号の有無の情報が上述の入賞判定パターン情報と一致しなかった場合には、以降の第2特図始動口5128の入球に伴う処理を行わずに処理を終了する。

#### 【1402】

ステップS6309では、上述の内部情報レジスタのクロック信号状態ビットを参照し、乱数用クロックの周波数に異常があるか無いかを判定する。そして、乱数用クロックの周波数に異常がある場合にはステップS6310の処理を実行することなくステップS6311に進み、乱数用クロックの周波数に異常がない場合には、ステップS6310に進む。

#### 【1403】

ステップS6310では、カウンタ回路（乱数回路）5316から乱数を取得する。より具体的には、第2特図始動口5128に入賞があり、且つ、保留している特図変動遊技の数が4未満である場合には、第2特図始動口5128に対応するカウンタ回路5316の乱数値レジスタから取り出した値を特図当選乱数値として取得、または取り出した値に「所定の加工」を行なった値を特図当選乱数値として取得する。また、上述の特図乱数値生成用の乱数カウンタから取り出した値を特図乱数値として取得、または取り出した値に「所定の加工」を行なった値を特図乱数値として取得し、RAM5308に設けた乱数値記憶領域に特図当選乱数値と共に記憶する。

#### 【1404】

ステップS6311～ステップS6313では、前々回普図始動口検出信号記憶領域、前回普図始動口検出信号記憶領域、および今回普図始動口検出信号記憶領域の各記憶領域に記憶した普図始動口球検出センサの検出信号の有無の情報を比較し、普図始動口球検出センサにおける過去3回分の検出信号の有無の情報が予め定めた入賞パターン情報と一致するか否かを判定する。そして、普図始動口球検出センサにおいて過去3回分の検出信号の有無の情報が、予め定めた入賞判定パターン情報（本実施例では、今回普図始動口検出信号がオン、前回普図始動口検出信号がオン、前々回普図始動口検出信号がオフであることを示す情報）と一致した場合に、普図始動口5124を球が通過したと判定する。例えば、普図始動口球検出センサにおいて過去3回分の検出信号の有無の情報が上述の入賞判定パターン情報と一致した場合には、普図始動口5124の球の通過があったと判定し、以降の普図始動口5124の球の通過に伴う処理を行うが、過去3回分の検出信号の有無の情報が上述の入賞判定パターン情報と一致しなかった場合には、以降の普図始動口5124の球の通過に伴う処理を行わずに処理を終了する。

#### 【1405】

ステップS6314では、上述の内部情報レジスタのクロック信号状態ビットを参照し、乱数用クロックの周波数に異常があるか無いかを判定する。そして、乱数用クロックの周波数に異常がある場合にはステップS6315の処理を実行することなく処理を終了し、乱数用クロックの周波数に異常がない場合には、ステップS6315に進む。

#### 【1406】

ステップS6315では、カウンタ回路（乱数回路）5316から乱数を取得する。より具体的には、普図始動口5124に入賞があり、且つ、保留している普図変動遊技の数が2未満である場合には、普図始動口5124に対応するカウンタ回路5316の乱数値レジスタから値を普図当選乱数値として取得する。また、上述の普図乱数値生成用の乱数カウンタから値を普図乱数値として取得し、RAM5308に設けた乱数値記憶領域に普図当選乱数値と共に記憶する。

10

20

30

40

50



## 【1407】

なお、本実施例においては、CPU5304は、各種始動口検出信号に入力がオン→オンと、最低で1回の割込み周期分の期間（本実施例では2ms）を経なければカウンタ回路5316の乱数値レジスタの値をラッチしないように構成されている。一方、乱数値レジスタは、各種始動口検出信号が入力された場合のラッチ信号は、乱数用クロックRCKの4周期分のオンレベルの信号を経なければラッチが行われなくなっている。本実施例では乱数用クロックRCKは10MHzであるため、最大でも $0.4\mu s + 0.1\mu s = 0.5\mu s$ 未満でラッチは完了する。すなわち、上記ステップS6305などで乱数値レジスタから値を取得する前に各種始動口検出信号によるラッチは確実に終了しているように構成されている。

10

## 【1408】

## &lt;特図状態更新処理&gt;

次に、図189を用いて、上述の主制御部タイマ割り込み処理における特図状態更新処理について説明する。なお、同図は特図状態更新処理の流れを示すフローチャートである。

## 【1409】

ステップS6401では、内部情報レジスタのクロック信号状態ビットを参照し、乱数用クロックの周波数に異常があるか無いかを判定する。そして、乱数用クロックの周波数に異常がある場合にはステップS6402～S6409の処理を実行することなく、ステップS6410に進み、乱数用クロックの周波数に異常がない場合には、ステップS6402に進む。

20

## 【1410】

ステップS6402では、特図変動中であるか否かを判定し、該当する場合にはステップS6403に進み、該当しない場合にはステップS6406に進む。ステップS6403では、特図表示装置5112を構成する7セグメントLEDが、前回は消灯であったか否かを判定し、前回は消灯の場合はステップS6405に進んで7セグメントLEDを点灯し、前回は点灯の場合はステップS6404に進んで7セグメントLEDを消灯する。これにより、7セグメントLEDの点灯と消灯を繰り返す点灯・消灯駆動制御を行う。

## 【1411】

ステップS6406では、特図変動停止であるか否かを判定し、該当する場合にはステップS6407に進み、該当しない場合は処理を終了する。

30

## 【1412】

ステップS6407では、大当たりフラグの情報に応じて特図を停止表示する。具体的には、大当たりフラグがオンで確変フラグがオフの場合には特図表示装置5114に、上述の特図1、大当たりフラグがオンで確変フラグがオンの場合には特図表示装置5114に、上述の特図2、大当たりフラグがオフの場合には、上述の特図3の態様となるように特図表示装置5112を構成する7セグメントLEDの点灯・消灯駆動制御を行う。

## 【1413】

ステップ408では、停止表示期間を設定する。具体的には、所定の停止表示期間（例えば500m秒間）、特図の表示を維持するためにRAM5308に設けた特図停止時間管理用タイマの記憶領域に停止期間を示す情報を設定する。この設定により特図の停止表示をおこない、特図変動遊技の結果を遊技者に報知するようにしている。

40

## 【1414】

ステップS6409では、停止コマンドの出力設定を行った後、処理を終了する。具体的には、コマンド設定送信処理（上記ステップS6215）で一般コマンド回転停止設定送信処理を実行させるために上述の送信情報記憶領域に02Hを送信情報（一般情報）として追加記憶する。

## 【1415】

また、図示はしないが、所定の停止表示期間が終了したタイミング（特図停止時間管理

50

用タイマの値が1から0になったタイミング)で開始する特図状態更新処理では、大当たりフラグがオンの場合には、所定の入賞演出期間(例えば3秒間)すなわち装飾図柄表示装置5110による大当たりを開始することを遊技者に報知する画像を表示している期間待機するためにRAM5308に設けた特図待機時間管理用タイマの記憶領域に入賞演出期間を示す情報を設定する。また、所定の入賞演出期間が終了したタイミング(特図待機時間管理用タイマの値が1から0になったタイミング)で開始する特図状態更新処理では、所定の開放期間(例えば29秒間、または可変入賞口5130に所定球数(例えば10球)の遊技球の入賞を検出するまで)可変入賞口5130の扉部材の開閉駆動用のソレノイド5330に、扉部材を開放状態に保持する信号を出力するとともに、RAM5308に設けた扉開放時間管理用タイマの記憶領域に開放期間を示す情報を設定する。また、所定の開放期間が終了したタイミング(扉開放時間管理用タイマの値が1から0になったタイミング)で開始する特図状態更新処理では、所定の閉鎖期間(例えば1.5秒間)可変入賞口5130の扉部材の開閉駆動用のソレノイド5330に、扉部材を閉鎖状態に保持する信号を出力するとともに、RAM5308に設けた扉閉鎖時間管理用タイマの記憶領域に閉鎖期間を示す情報を設定する。また、この扉部材の開放・閉鎖制御を所定回数(例えば15ラウンド)繰り返し、終了したタイミングで開始する特図状態更新処理では、所定の終了演出期間(例えば3秒間)すなわち装飾図柄表示装置5110による大当たりを終了することを遊技者に報知する画像を表示している期間待機するように設定するためにRAM5308に設けた演出待機時間管理用タイマの記憶領域に演出待機期間を示す情報を設定する。また、所定の終了演出期間が終了したタイミング(演出待機時間管理用タイマの値が1から0になったタイミング)で開始する特図状態更新処理では、特図の状態を非作動中に設定する。

10

20

#### 【1416】

ステップS6410では、大当たりフラグをオフ(ハズレ)に設定し、ステップS6411では、特図の変動を停止する。また、次のステップS6412では、コマンド設定送信処理(上記ステップS6215)で乱数用クロック異常による停止設定送信処理を実行させるために上述の送信情報記憶領域に10Hを送信情報(一般情報)として追加記憶して処理を終了する。なお、乱数用クロックRCKに異常があった場合は、特図または／および普図の表示を消してもよい。

#### 【1417】

30

##### <払出制御部メイン処理>

次に、図190を用いて、払出制御部5550が実行する払出制御部メイン処理について説明する。なお、同図は払出制御部メイン処理の流れを示すフローチャートである。

#### 【1418】

払出制御部5550には、電源が投入されるとリセット信号を出力するリセット信号出力回路を設けている。このリセット信号を入力した払出制御部5400のCPUは、リセット割り込みによりリセットスタートしてROMに予め記憶している制御プログラムに従って処理を実行する。

#### 【1419】

ステップS6501では、初期設定1を行う。この初期設定1では、CPUのスタックポインタ(SP)へのスタック初期値の設定等を行う。

40

#### 【1420】

ステップS6502では、低電圧信号がオンであるか否か、すなわち、電圧監視回路が、電源管理部5650から払出制御部5550に供給している電源の電圧値が所定の値(本実施例では9V)未満である場合に電圧が低下したことを示す低電圧信号を出力しているか否かを監視する。そして、低電圧信号がオンの場合(電源の遮断を検知した場合)にはステップS6502の処理を繰り返し実行し、低電圧信号がオフの場合(電源の遮断を検知していない場合)にはステップS6503に進む。

#### 【1421】

ステップS6503では、初期設定2を行う。この初期設定2では、後述する払出制御

50

部タイマ割り込み処理を定期毎に実行するための周期を決める数値をカウンタ・タイマに設定する処理、RAMへの書き込みを許可する設定、I/Oポートの初期設定等を行う。

【1422】

ステップS6504では、電源の遮断前（電断前）の状態に復帰するか否かの判定を行い、電断前の状態に復帰しない場合（パチンコ機5100を初期状態にする場合）にはステップS6506に進み、電断前の状態に復帰する場合にはステップS6505に進む。

【1423】

具体的には、最初に、電源基板に設けた操作部を遊技店の店員などが操作した場合に送信されるRAMクリア信号がオン（操作があったことを示す）であるか否か、すなわちRAMクリアが必要であるか否かを判定し、RAMクリア信号がオンの場合（RAMクリアが必要な場合）には、パチンコ機5100を初期状態にすべくステップS6506に進む。一方、RAMクリア信号がオフの場合（RAMクリアが必要でない場合）は、RAMに設けた電源ステータス記憶領域に記憶した電源ステータスの情報を読み出し、この電源ステータスの情報がサスペンドを示す情報であるか否かを判定する。そして、電源ステータスの情報がサスペンドを示す情報でない場合には、パチンコ機5100を初期状態にすべくステップS6506に進み、電源ステータスの情報がサスペンドを示す情報である場合には、RAMの所定の領域（例えば全ての領域）に記憶している1バイトデータを初期値が0である1バイト構成のレジスタに全て加算することによりチェックサムを算出し、算出したチェックサムの結果が特定の値（例えば0）であるか否か（チェックサムの結果が正常であるか否か）を判定する。そして、チェックサムの結果が0の場合（チェックサムの結果が正常である場合）には電断前の状態に復帰すべくステップS6505に進み、チェックサムの結果が0以外である場合（チェックサムの結果が異常である場合）には、パチンコ機5100を初期状態にすべくステップS6506に進む。同様に電源ステータスの情報が「サスペンド」以外の情報を示している場合にもステップS6506に進む。

【1424】

ステップS6505では、復電時処理を行う。この復電時処理では、RAMの記憶領域のうち、復電時にクリアすべき記憶領域（コマンドを格納するためのコマンドバッファ、エラー状態を記憶するためのエラーステータスなどを除く記憶領域）の初期化などを行う。

【1425】

ステップS6506では、初期化処理を行う。この初期化処理では、割り込み禁止の設定、スタックポインタへのスタック初期値の設定、RAMの所定の領域（例えば、全ての記憶領域）の初期化などを行う。

【1426】

ステップS6507では、初期設定3を行う。この初期設定3では、RAMに設けたエラーステータス記憶領域に記憶したエラーステータスのうち、不正払出エラーと払出超過エラー以外の情報をクリアしたり、割り込み許可の設定などを行う。

【1427】

ステップS6508では、主制御部5300から入力したデータの中に未解析データがあるか無いかを判定し、未解析データがある場合にはステップS6509でコマンド解析処理を行い、未解析データがない場合にはステップS6510に進む。

【1428】

ステップS6510では、低電圧信号がオフであるか否かを監視し、低電圧信号がオフの場合（電源の遮断を検知していない場合）にはステップS6508に戻り、低電圧信号がオンの場合（電源の遮断を検知した場合）にはステップS6511に進む。

【1429】

ステップS6511では、電断時処理を行う。この電断時処理では、RAMに設けたスタックポインタ退避領域に現在のスタックポインタの値を記憶し、上述の電源ステータス記憶領域にサスペンドを示す情報を設定する。また、RAMの所定の領域（例えば全ての領域）に記憶している1バイトデータを初期値が0である1バイト構成のレジスタに全て

加算し、チェックサム算出用数値記憶領域に記憶している値からその加算した結果を減算した値をチェックサム（電断時チェックサム）として算出し、算出した電断時チェックサムを上述のチェックサム算出用数値記憶領域に記憶し、RAMへの書き込みを禁止する設定などを行う。

#### 【1430】

ステップS6512では、低電圧信号がオンであるか否かを監視し、低電圧信号がオンの場合（電源の遮断を検知した場合）にはステップS6512の処理を繰返し実行して低電圧信号がオフになるのを待ち、低電圧信号がオフの場合（電源の遮断を検知していない場合）にはステップS6501に戻り、払出制御部リセット割り込み処理を最初から開始する。

10

#### 【1431】

＜払出制御部タイマ割り込み処理＞

次に、図191（a）を用いて、払出制御部5550のCPUが実行する払出制御部タイマ割り込み処理について説明する。なお、同図は払出制御部タイマ割り込み処理の流れを示すフローチャートである。

#### 【1432】

払出制御部5550は、所定の周期（本実施例では1msに1回）でタイマ割り込みを発生するカウンタ・タイマを備えており、このタイマ割り込みを契機として払出制御部タイマ割り込み処理を所定の周期で開始する。

#### 【1433】

ステップS6601では、タイマ割り込みスタート処理を行う。このタイマ割り込みスタート処理では、CPUの各レジスタの値をスタック領域に一時的に退避する処理などを行う。ステップS6602では、ポート入力管理処理を行う。このポート入力管理処理では、I/Oポートの値を取得して、各種センサの状態などを検出する。

20

#### 【1434】

ステップS6603では、タイマ更新管理処理を行う。このタイマ更新管理処理では、払出報知用LEDの点灯／消灯時間、モータ駆動／非駆動時間などを計時するためのタイマなどを含む各種タイマを更新する。

#### 【1435】

ステップS6604では、エラー管理処理を行う。このエラー管理処理では、主制御部5300から乱数クロック異常コマンドを受信しているか否かを判定し、該当する場合にはRAMに設けたエラーステータス記憶領域に乱数クロック異常を示す情報を記憶する。また、I/Oポートに入力する皿満杯信号を検出して皿満杯信号がオンであるか否かを判定し、皿満杯信号がオンの場合（下皿5150が球で一杯になっている場合）には、RAMに設けたエラーステータス記憶領域に皿満杯エラーを示す情報を記憶し、皿満杯信号がオフの場合（下皿5150に球を貯留する空きができた場合）には、エラーステータス記憶領域に皿満杯エラーの解除を示す情報を記憶する。また、主制御部5300と払出制御部5550との間の通信回線が断線などにより通信可能かどうかを検出し、通信可能な場合には、RAMに設けたエラーステータス記憶領域に通信可能であることを示す情報を、また通信不可能な場合には通信不可能であることを示す情報を記憶する。また、エラー管理処理では、I/Oポートに入力するエラー解除スイッチ信号を検出してエラー解除信号がオンであるか否かを判定し、エラー解除信号がオンである場合には、エラーステータス記憶領域に記憶している、乱数クロック異常を示す情報、不正払出エラーの情報、または、払出超過エラーの情報を初期化して、これらのエラーを解除する。

30

40

#### 【1436】

なお、乱数クロック異常を示す情報に関しては、他のエラーと異なりエラー解除スイッチ信号を検出した場合でもエラーが解除されないようにしてもよい。なぜなら、乱数クロック異常の場合は、エラーを復帰した場合に保留された特図の変動を行うとして、その特図抽選にかかる乱数の信頼性が極めて低いため、エラー解除で復帰させて変動後の遊技を行わせたくない場合があるためである。また、「複数の異常状態があり、第1の種類の異

50

常状態は異常解除入力により異常状態を解除可能に設けられ、第2の種類の異常状態は異常解除入力により異常状態を解除不能に設けられ、第2の種類の異常状態は少なくとも乱数クロックの異常状態を含む」ことは、他の制御部やスロットマシンにも適用できる。この場合、「第2の種類の異常状態は、電源断時にもバックアップにより第2の種類の異常状態の記憶が保持されるようにし」てもよく、さらに「第2の種類の異常状態の記憶は、第2の種類の異常状態を含む複数の記憶情報の初期化により記憶を解除できるようにし」てもよく、さらに「複数の記憶情報は、遊技者の利益に関わる情報」を含んでもよく、さらに「遊技者の利益に関わる情報は乱数クロックに基づいて生成される乱数に基づいて抽選により決定される情報」であってもよい。これにより、乱数クロック異常を確実に報知できると共に、乱数クロックに異常があった場合に、乱数クロックにより基づいて抽選された利益を確実にクリアすることができるため、乱数クロック周波数の異常により遊技店が不利益を被ることを防止することができる。

10

#### 【1437】

ステップS6605では、CRユニット通信管理処理を行う。このCRユニット通信管理処理では、上述のエラーステータス記憶領域に乱数クロック異常を示す情報が記憶されているか否かを判定し、乱数クロック異常を示す情報が記憶されている場合には、遊技媒体の貸出を行わないようにする。また、CRユニットに対してカードを返却する信号を出力しないようにする。具体的には、遊技媒体の貸出処理を飛ばしたり、CRユニットの返却処理を飛ばしたりする。一方、乱数クロック異常を示す情報が記憶されていない場合には、インターフェース部5556から遊技媒体貸出信号を受信して遊技媒体貸出信号センサ信号がオンであるか否かを判定し、遊技媒体貸出信号がオンの場合（インターフェース部5556からの球貸要求を入力した場合）には、RAMに設けた遊技媒体貸出情報記憶領域に遊技媒体の貸出要求があったことを示す情報を記憶する。

20

#### 【1438】

ステップS6606では、払出管理処理を行う。この払出管理処理では、上述のエラーステータス記憶領域から、乱数クロック異常を示す情報、不正払出エラーの情報、および払出超過エラーの情報を読み出し、いずれのエラーも発生していない場合に、センサ回路を介して入力する払出センサの信号（以下、払出センサ信号と称する場合がある）に基づいて払出個数の監視を行う。すなわち、所定のエラー（ここでは、いずれかのエラー）が発生している場合にはモータの駆動、すなわち払出装置からの賞媒体（例えば遊技球）の払出を停止するようにしている。

30

#### 【1439】

また、上述のエラーステータス記憶領域から、乱数クロック異常を示す情報、皿満杯エラーの情報、不正払出エラーの情報、および払出超過エラーの情報を読み出し、いずれのエラーも発生していない場合に、払出開始監視処理、初期位置検索動作処理、通常払出動作処理、リトライ動作処理、逆回転動作処理のいずれかの処理を行う。

#### 【1440】

払出開始監視処理では、貸出要求数、および賞球要求数が0であり、次賞球要求数が0以外の場合は、賞球要求数に次賞球要求数をセットし、次賞球要求数をクリアする。また、スプロケットを駆動するモータの位置が不確定の場合（動作モードが初期位置検索動作モードの場合）には、払出完了数チェックから1を減算して払出完了数チェック記憶領域に記憶し、スプロケットを駆動するモータの位置が確定している場合（動作モードが通常払出動作モードの場合）には、払出完了数チェックとして払出完了数チェック記憶領域に0を設定する。また、賞球要求数を、スプロケットのモータを駆動する量（モータ駆動量）に変換し、これをRAMに設けたモータ駆動量記憶領域に記憶すると共に、RAMに設けたモータ制御データテーブルを参照してモータ駆動量に対応するモータ駆動制御データを選択し、正転を示すモータ駆動制御データをI/Oポートを介してモータ制御回路に出力する。これにより、モータ制御回路はスプロケットのモータの励磁位置を所定回変化してスプロケットを正方向に回転駆動する。

40

#### 【1441】

50

初期位置検索動作処理および通常払出動作処理では、モータの駆動終了後に、払出完了数チェック記憶領域から払出完了チェックを読み出し、払出完了チェックが0の場合には、払出開始監視処理を実行する準備を行い、払出完了チェックが0以外の場合には、エラーステータス記憶領域に払出装置エラーを示す情報を設定すると共に、リトライ動作処理を実行する準備を行う。

#### 【1442】

リトライ動作処理では、所定の時間が経過するのを待ち（リトライ動作開始待ちタイマが0になるのを待ち）、リトライ動作開始待ちタイマが0になった場合には、逆回転動作処理を実行する準備を行う。逆回転操作処理では、上述のモータ制御データテーブルを参照してモータ駆動量に対応するモータ駆動制御データを選択し、逆転を示すモータ駆動制御データをI/Oポートを介してモータ制御回路に出力する。これにより、モータ制御回路はスプロケットのモータの励磁位置を所定回変化してスプロケットを逆回転駆動する。また、逆回転操作処理では、モータの駆動終了後に払出開始監視処理を実行する準備を行う。

10

#### 【1443】

ステップS6607では、モータ駆動管理処理を行う。このモータ駆動管理処理では、駆動開始監視処理、加速駆動処理、定速駆動処理、ブレーキ駆動処理、駆動終了処理のいずれかの処理を行う。

#### 【1444】

駆動開始監視処理では、上述のエラーステータス記憶領域から、乱数クロック異常を示す情報、皿満杯エラーの情報、不正払出エラーの情報、および払出超過エラーの情報を読み出し、いずれのエラーも発生していない場合に、モータ制御データテーブルを参照してモータ駆動量に対応するモータ駆動制御データを選択し、正転を示すモータ駆動制御データをI/Oポートを介してモータ制御回路に出力する。これにより、モータ制御回路はスプロケットのモータの励磁位置を所定回変化してスプロケットを正方向に回転駆動する。

20

#### 【1445】

加速駆動処理および定速駆動処理では、スプロケットが初期位置検索動作中、または、逆回転動作中の場合を除き、モータの励磁位置を16回変化させるごとに払出完了数チェックから1を減算して払出完了数チェック記憶領域に記憶する。また、更新後の払出完了数チェックが-4未満になった場合には、ブレーキ駆動処理を実行する準備を行う。さらに、上述の遊技媒体貸出情報記憶領域から遊技媒体貸出情報を読み出して、遊技媒体の貸出要求があったことを示す情報の有無を判定し、遊技媒体の貸出要求があったことを示す情報がある場合（賞球の払出中にインターフェース部5556からの球貸要求を入力した場合）にも、ブレーキ駆動処理を実行する準備を行う。

30

#### 【1446】

ブレーキ駆動処理では、所定の時間が経過するのを待ち（モータ駆動管理タイマが0になるのを待ち）、モータ駆動管理タイマが0になった場合には、駆動終了処理を実行する準備を行い、駆動終了処理では、モータ駆動の後処理を行う。

#### 【1447】

ステップS6608では、LED管理処理を行う。このLED管理処理では、エラーステータス記憶領域の乱数クロック異常を示す情報が異常を示している場合には、乱数クロックの周波数に異常が発生していることを遊技者に報知するためのLEDを点灯させ、エラーステータス記憶領域の乱数クロック異常を示す情報が異常を示していない場合には、そのLEDを消灯させる。また、エラーステータス記憶領域の不正払出エラー情報が不正払出エラーが発生中であることを示している場合には、不正払出エラーが発生していることを遊技者に報知するためのLEDを点灯させるとともに、不正払出エラーが発生していないことを示している場合にはそのLEDを消灯させる。また、エラーステータス記憶領域の払出超過エラー情報が払出超過エラーが発生中であることを示している場合には、払出超過エラーが発生していることを遊技者に報知するためのLEDを点灯させるとともに、払出超過エラーが発生していないことを示している場合にはそのLEDを消灯させる

40

50

。

#### 【1448】

ステップS6609では、信号出力管理処理を行う。この信号出力管理処理では、RAMに記憶している遊技情報（例えば払出センサ信号を入力するたびに出力する賞球信号）を、情報出力回路（図示省略）を介してパチンコ機5100とは別体の情報入力回路（図示省略）に出力する。

#### 【1449】

ステップS6610では、タイマ割り込みエンド処理を行う。このタイマ割り込みエンド処理では、ステップS6601で一時的に退避した各レジスタの値を元の各レジスタに設定したり、割り込み許可の設定などを行う。

10

#### 【1450】

##### <ストローブ割り込み処理>

次に、図191（b）を用いて、払出制御部5550のCPUが実行するストローブ割り込み処理について説明する。なお、同図はストローブ割り込み処理の流れを示すフローチャートである。

#### 【1451】

払出制御部5550には主制御部5300が出力するストローブ信号が入力されており、主制御部5300が払出制御部5550に対してコマンドを送信すると、このストローブ信号が払出制御部5550に入力され、払出制御部5550のCPUにストローブ割り込みが通知される。

20

#### 【1452】

ストローブ割り込みを検出した払出制御部5550のCPUは、ストローブ割り込み処理を実行し、ステップS6701において主制御部5300から受信したコマンドをRAMの所定記憶領域に記憶する。

#### 【1453】

##### <副制御部メイン処理>

次に、図192（a）を用いて、副制御部5400のCPU5404が実行する副制御部メイン処理について説明する。なお、同図は副制御部メイン処理の流れを示すフローチャートである。

#### 【1454】

副制御部5400には、電源が投入されるとリセット信号を出力するリセット信号出力回路を設けている。このリセット信号を入力した基本回路5402のCPU5404は、リセット割り込みによりリセットスタートしてROM406に予め記憶した制御プログラムに従って処理を実行し、まず、ステップS6801で各種の初期設定を行う。この初期設定では、入出力ポートの初期設定や、各種変数の初期化等を行う。ステップS6802では、コマンド入力処理（詳細は後述）を行う。

30

#### 【1455】

ステップS6803では、I/O5410の出力ポートを介して副制御部5500にコマンドを出力する。ステップS6804では、後述するタイマ変数記憶領域の値が10以上であるか否かを判定する。タイマ変数記憶領域の値が10以上である場合はステップS6805に進み、タイマ変数記憶領域の値が10未満である場合にはステップS6802に進む。ステップS6805では、タイマ変数記憶領域に0を格納する。

40

#### 【1456】

ステップS6806では、演出データ更新処理を行う。この演出データ更新処理では、後述するコマンド記憶領域の内容を確認し、主制御部5300から乱数クロック異常コマンドを受信しているか否かを判断する。そして、乱数クロック異常コマンドを受信している場合には、装飾図柄表示装置5110、遮蔽手段5250、スピーカ5416、各種ランプ5420および演出装置5200の演出用可動体等によって乱数クロックが異常であることを外部に報知するための動作制御データの更新を行う。一方、乱数クロック異常コマンドを受信していない場合には、後述する変動パターン選択処理で記憶する変動番号、

50

仮停止図柄の組合せ、および停止図柄の組合せの種別の更新を行うと共に、装飾図柄の変動表示を開始してからの経過時間に基づいて装飾図柄表示装置 5 1 1 0、遮蔽手段 5 2 5 0、スピーカ 5 4 1 6、各種ランプ 5 4 2 0 および演出装置 5 2 0 0 の演出用可動体等による演出を制御するための動作制御データの更新を行う。

#### 【1 4 5 7】

ステップ S 6 8 0 6 では、決定された演出情報が示している態様で装飾図柄変動表示を行うように次回実行する上記ステップ S 6 8 0 3 の処理で副制御部 5 5 0 0 に出力するコマンド（例えば左に装飾 7 を停止することを指示するコマンドや遮蔽手段 5 2 5 0 を動作させるコマンド等）を R A M 5 4 0 8 に設けた液晶コマンド格納領域に格納する等、後述するステップ S 6 8 0 7、S 6 8 0 8、S 6 8 0 9 によるスピーカ 5 4 1 6、各種ランプ 5 4 2 0、および演出用可動体を制御する準備を行う。また、所定の条件が成立している場合には所定の演出を実行するか否か、例えばチャンスボタンを用いた演出を行うか否か等の抽選を行う。

10

#### 【1 4 5 8】

ステップ S 6 8 0 7 では、音出力処理を行う。この音出力処理では、上記ステップ S 6 8 0 6 で取得したスピーカ制御用の情報に含まれるスピーカ 5 4 1 6 に出力する音声データを I / O 5 4 1 0 の出力ポートに設定し、スピーカ 5 4 1 6 の出力制御を音源 I C 5 4 1 8 に行わせる。例えば、上記ステップ S 6 8 0 6 で乱数クロックが異常であることを外部に報知するための動作制御データがセットされている場合には、乱数クロックが異常であることをスピーカ 5 4 1 6 を用いて音声（例えば、警告音）で報知させる。

20

#### 【1 4 5 9】

ステップ S 6 8 0 8 では、ランプ制御処理を行う。このランプ制御処理では、上記ステップ S 6 8 0 6 で取得した各種ランプ制御用の情報に含まれる各種ランプ 5 4 2 0 に出力するランプの点灯・消灯を示すデータ等を I / O 5 4 1 0 の出力ポートに設定し、各種ランプ 5 4 2 0 の点灯や消灯の制御を表示回路 5 4 2 2 に行わせる。例えば、上記ステップ S 6 8 0 6 で乱数クロックが異常であることを外部に報知するための動作制御データがセットされている場合には、乱数クロックが異常であることを各種ランプ 5 4 2 0 を用いて光（例えば、点滅表示）で報知させる。

#### 【1 4 6 0】

ステップ S 6 8 0 9 では、演出用駆動装置制御処理を行う。この演出用駆動装置制御処理では、上記ステップ S 6 8 0 6 で取得した演出用可動体の制御用の情報に含まれる動作タイミングを示すデータ等を I / O 5 4 1 0 の出力ポートに設定し、演出用可動体等を駆動する各種演出用駆動装置 5 4 2 4 の制御を演出用駆動装置制御回路 5 4 2 6 に行わせる。例えば、上記ステップ S 6 8 0 6 で乱数クロックが異常であることを外部に報知するための動作制御データがセットされている場合には、乱数クロックが異常であることを演出用可動体を用いて動き（例えば、停止）で報知させる。

30

#### 【1 4 6 1】

副制御部 5 4 0 0 は、後述するストロープ処理、チャンスボタン処理、または副制御部タイマ割り込み処理による中断を除いて、以降、ステップ S 6 8 0 2 ～ S 6 8 0 9 の処理を繰り返し実行する。

40

#### 【1 4 6 2】

##### <コマンド入力処理>

次に、図 1 9 2 (b) を用いて、上記副制御部メイン処理におけるコマンド入力処理について説明する。同図はコマンド入力処理の流れを示すフローチャートである。

#### 【1 4 6 3】

ステップ S 6 8 1 0 では、後述するコマンド記憶領域の内容を確認し、未処理のコマンドが残っているか否かを判断する。そして、コマンド記憶領域に未処理のコマンドが残っている場合にはステップ S 6 8 1 2 に進み、コマンド記憶領域に未処理のコマンドが残っていない場合には処理を終了して副制御部メイン処理に復帰する。

#### 【1 4 6 4】

50



図192(c)は変動パターン選択処理の流れを示すフローチャートであり、同図(d)は図柄停止処理の流れを示すフローチャートである。ステップS6821では、コマンド記憶領域に記憶している未処理コマンドのうちの次に処理すべき未処理コマンドの種類に基づいて、図192(c)に示す変動パターン選択処理(例えば未処理コマンドが上記図柄変動開始コマンドに基づいて実行する)や、同図(d)に示す図柄停止処理等を行う。未処理コマンドに基づく処理は他にも備えている。例えば、大当たり中に可変入賞口5130の開放制御を開始するたびに主制御部5300が出力し、大当たり開始後の可変入賞口5130の開放回数を示す情報を含むラウンド開始コマンドが未処理コマンドである場合に行うラウンド開始処理等である。その他の処理は、ここでは割愛する。

#### 【1465】

変動パターン選択処理のステップS6821では、未処理コマンドに含まれている上記大当たりフラグの値、確変フラグの値、およびタイマ番号を抽出し、RAM5408のそれぞれの記憶領域に記憶する。また、上述の変動番号選択テーブルや図柄決定テーブルを参照して演出データ(本実施例では変動番号、仮停止図柄・停止図柄の組合せ等)を選択し、これをRAM5408に設けた記憶領域に記憶した後、処理を終了する。

#### 【1466】

図柄停止処理のステップS6841では、上記図柄記憶領域に記憶している停止図柄の組合せを構成する3つの装飾図柄を装飾図柄表示装置5110の左、中、右図柄表示領域5110a~5110cの3つの表示領域に表示するように設定して処理を終了する。また、上記ラウンド開始処理では未処理コマンドに含まれている上記大当たり開始後の可変入賞口5130の開放回数を示す情報を抽出し、RAM5408の記憶領域に記憶する。

#### 【1467】

##### <ストロープ割り込み処理>

次に、図192(e)を用いて、副制御部5400のストロープ割り込み処理について説明する。なお、同図はストロープ割り込み処理の流れを示すフローチャートである。

#### 【1468】

このストロープ割り込み処理は、副制御部5400が、主制御部5300が出力するストロープ信号を検出した場合に実行する処理である。ストロープ割り込み処理のステップS6861では、主制御部5300が出力したコマンドを未処理コマンドとしてRAM5408に設けた上記コマンド記憶領域に記憶する。

#### 【1469】

##### <チャンスボタン割り込み処理>

次に、図192(f)を用いて、副制御部5400のチャンスボタン割り込み処理について説明する。なお、同図はチャンスボタン割り込み処理の流れを示すフローチャートである。

#### 【1470】

このチャンスボタン割り込み処理は、副制御部5400がチャンスボタン検出回路5364によってチャンスボタン5146の操作を検出した場合に実行する処理である。

#### 【1471】

チャンスボタン割り込み処理のステップS6881では、RAM5408の検知カウンタ記憶領域に記憶している、チャンスボタン5146の押下回数を計測するための検知カウンタから値を取得し、取得した値に1を加算してから元の検知カウンタ記憶領域に記憶する。

#### 【1472】

##### <変数更新割り込み処理>

次に、図192(g)を用いて、副制御部5400のCPU5404によって実行する変数更新割り込み処理について説明する。なお、同図は変数更新割り込み処理の流れを示すフローチャートである。

#### 【1473】

副制御部5400は、所定の周期(本実施例では2msに1回)でタイマ割り込みを発

10

20

30

40

50

生するハードウェアタイマを備えており、このタイマ割り込みを契機として、変数更新割り込み処理を所定の周期で実行する。

#### 【1474】

変数更新割り込み処理のステップS6901では、RAM5408のタイマ変数記憶領域の値に1を加算して元のタイマ変数記憶領域に記憶する。従って、ステップS6304において、タイマ変数の値が10以上と判定されるのは20ms毎(2ms×10)となる。

#### 【1475】

##### ＜基本回路302のパッケージ＞

次に、上述の主制御部5300の基本回路5302が搭載されるパッケージについて詳細に説明する。図193(a)は、基本回路5302が搭載されるパッケージ5680の外観斜視図であり、同図(b)は、パッケージ5680の視認非容易部の近傍を示す側面図である。

#### 【1476】

パッケージ5680の短手方向両側には、パッケージ5680の長手方向に沿って複数の端子5682が配置されているが、パッケージ(マイクロコンピュータ)5680は、端子5682を設けることで該端子5682によって、該マイクロコンピュータ5680の裏面における第一の領域(例えば、裏面下方の空間)、該裏面に対向する前記基板の表面における第二の領域(例えば、表面上方の空間)および前記裏面と前記表面との間の空間における第三の領域(例えば、ICソケットを介してマイクロコンピュータを基板に実装する場合には、ICソケットの表面や裏面近傍の空間を含む)のうちの少なくとも一つである特定の領域を視認困難にする視認非容易部5680aと、端子5682を設けないことによって、視認非容易部5680aよりも特定の領域を視認容易にする視認容易部5680bと、を有して構成されている。

#### 【1477】

このような構成により、視認容易部5680bを通してパッケージ5680の一方側(例えば、表側)から他方側(例えば、裏側)を視認することが可能となるため、例えば、パッケージ5680の裏側や、パッケージ5680と基板(またはICソケット)の隙間などに不正な回路を後付けするような不正行為を容易に発見することができ、不正行為を未然に防止できる場合がある。

#### 【1478】

特に、この例では視認容易部5680bを短手方向側面に設けているため、長手方向側面に設けた場合に比べ、パッケージ5680の裏側の中でも特に視認することが難しく、かつ不正部品などが取付し易い裏側中央近傍を間近で視認することが可能となり、より確実に不正行為を防止できる場合がある。さらに、視認容易部5680bを長手方向の中央部近傍に設けているため、長手方向の端部近傍に設けた場合に比べ、パッケージ5680の裏側の広い範囲をいろいろな角度から視認することが可能で、より確実に不正行為を防止できる場合がある。なお、視認容易部を長手方向側面にも設ければ、視認範囲をさらに広げることができ、より確実に不正行為を防止できる場合がある。

#### 【1479】

##### ＜パッケージの変形例＞

次に、パッケージ5700の変形例について説明する。図194(a)は、基本回路5302が搭載されるパッケージ5700の平面図であり、同図(b)は、パッケージ5700の端子と下側基板のみを抜き出して示す外観斜視図である。

#### 【1480】

パッケージ5700は、板状体の上側基板5702および下側基板5704と、これらの上側基板5702および下側基板5704の間に挟まれるように配置される基本回路(CPUコア)5302と、この基本回路5302の接続端子の各々にボンディングワイヤを介して接続される複数の端子(リードフレーム)5706と、を備えており、複数の端子5706が短手方向両側に長手方向に沿って配設されたDIP(Dual Inline Package

10

20

30

40

50

）である。パッケージ 5700 の上側基板 5702 および下側基板 5704 の各々は、透明部材からなる視認容易部 5702a、5704a と、非透明部材からなる視認非容易部 5702b、5704b を、長手方向に交互に配置して構成されている。なお、上側基板 5702 における視認容易部 5702a と下側基板 5704 における視認容易部 5704a は、長手方向の同じ位置に積層され、上側基板 5702 における視認非容易部 5702b と下側基板 5704 における視認非容易部 5704b は、長手方向の同じ位置に積層されている。

#### 【1481】

このような構成により、視認容易部 5702a、5704a を通してパッケージ 5700 の一方側（例えば、表側）から他方側（例えば、裏側）を視認することが可能となるため、例えば、パッケージ 5700 の裏側や、パッケージ 5700 と基板（または IC ソケット）の隙間などに不正な回路を後付けするような不正行為を容易に発見することができ、不正行為を未然に防止できる場合がある。

10

#### 【1482】

なお、視認容易部 5702a、5704a は、パッケージ 5700 の裏側が視認可能な部材で構成されていればよく、例えば、全体が半透明部材で構成されていてもよく、また、少なくとも一部が透明部材や半透明部材で構成され、他の部位が非透明部材で構成されていてもよい。また、視認非容易部 5702b、5704b は、パッケージ 5700 の裏側が視認不能または視認困難な部材で構成されていればよく、例えば、全体が視認不能または視認困難な半透明部材で構成されていてもよい。また、視認容易部 5702a、5704a と視認非容易部 5702b、5704b を、長手方向の同じ位置に交互に配置したが、例えば、視認容易部 5702a、5704a のみでパッケージを形成してもよく、また、視認容易部 5702a、5704a と視認非容易部 5702b、5704b を、短手方向に交互に配置してもよい。

20

#### 【1483】

＜パッケージの変形例／変形例 1＞

次に、パッケージ 5700 の変形例について説明する。図 195（a）は変形例 1 に係るパッケージ 5710 の下側基板 5714（5714a、5714b）の一部分を示す外観斜視図である。この変形例 1 に係るパッケージ 5710 の下側基板 5714（図示はしないが上側基板も同様の構成である。以降の変形例も同じ）は、透明部材からなる視認容易部 5714a と、非透明部材からなる視認非容易部 5714b を長手方向に交互に配置している点はパッケージ 5700 と同様であるが、視認容易部 5714a における端子 5716 の配置態様と、視認非容易部 5714b における端子 5716 の配置態様を異ならせている。

30

#### 【1484】

具体的には、視認容易部 5714a においては、視認容易部 5714a の長手方向の長さ L1 に対して 1 つの端子 5716 を配置し、視認非容易部 5714b においては、視認容易部 5714a の長手方向の長さ L1（この例では、視認容易部 5714a の長手方向の長さ L1 と同じ長さ）に対して 3 つの端子 5716 を略等間隔に配置しており、視認非容易部 5714b における端子 5716 の数（この例では 3 つ）よりも、視認容易部 5714a における端子 5716 の数（この例では 1 つ）が少なくなるように構成している。

40

#### 【1485】

換言すれば、視認容易部 5714a に配置された端子 5716 と、視認非容易部 5714b に配置された端子 5716 のうち、最も視認容易部 5714a に近い端子 5716 との間の距離 L2 と、視認非容易部 5714b に配置された複数の端子 5716 の間の距離 L3 が異なっており、距離 L3 よりも距離 L2 が長くなるように構成している。

#### 【1486】

このような構成により、端子 5716 の数が少ない（視認を妨げる部材が少ない）視認容易部 5714a を通してパッケージ 5710 の一方側から他方側を確実に視認できるとともに、端子 5716 の数が多くても視認性に影響を与えない視認非容易部 5714b に

50

多くの端子 5716 を配置することができる場合がある。また、隣接する端子のノイズの影響を受けやすい信号線を、端子密度の低い視認容易部 5714a に配置された端子 5716 に優先的に接続すれば、視認性を確保しながらも、同時に、ノイズなどに起因する誤作動を未然に防止できる場合がある。

#### 【1487】

＜パッケージの変形例／変形例 2＞

図 195 (b) は、変形例 2 に係るパッケージ 5720 の下側基板 5724 (5724a、5724b) の一部分を示す外観斜視図である。この変形例 2 に係るパッケージ 5720 の基板は、透明部材からなる視認容易部 5724a と、非透明部材からなる視認非容易部 5724b を長手方向に交互に配置している点はパッケージ 5700 と同様であるが、視認非容易部 5724b には複数 (この例では 3 つ) の端子 5726 を配置しているのに対して、視認容易部 5724a には端子 5726 を 1 つも配置しないように構成している。

10

#### 【1488】

このような構成により、視認容易部 5724a における視認性をより一層確保でき、パッケージ 5720 の一方側から他方側を確実に視認できる場合がある。また、隣接する端子のノイズの影響を受けやすい複数の信号線を、視認容易部 5724a の一方の側 (例えば、同図の右側) の視認非容易部 5724b の端子 5726 と、視認容易部 5724a の他方の側 (例えば、同図の左側) の視認非容易部 5724b の端子 5726 と、に離間して接続すれば、視認性を確保しながらも、同時に、ノイズなどに起因する誤作動を未然に防止できる場合がある。

20

#### 【1489】

＜パッケージの変形例／変形例 3＞

図 195 (c) は、変形例 3 に係るパッケージ 5730 の下側基板 5732 (5732a、5732b) の一部分を示す外観斜視図である。この変形例 3 に係るパッケージ 5730 の基板は、非透明部材からなる視認非容易部 5732b の一部に、透明部材からなる立方体形状の視認容易部 5732a を形成するとともに、視認非容易部 5732b には複数 (この例では 3 つ) の端子 5736 を配置しているが、視認容易部 5732a には端子 5736 を 1 つも配置しないように構成している。

#### 【1490】

このような構成により、視認容易部と視認非容易部を長手方向に交互に配置する場合に比べ、パッケージ 5730 の基板のコストを低減することができるとともに、視認容易部 5732a を通してパッケージ 5730 の一方側から他方側を視認することができる。

30

#### 【1491】

なお、視認容易部 5732a は、パッケージ 5730 の基板の短手方向両側に形成してもよいし、短手方向片側のみに形成してもよいが、両側に形成すれば、視認性をより高めることができる場合がある。また、視認容易部 5732a の数は一つでもよいし、複数でもよいが、複数形成すれば、視認性をより高めることができる場合がある。

#### 【1492】

＜パッケージの変形例／変形例 4＞

図 195 (d) は、変形例 4 に係るパッケージ 5740 の下側基板 5742 (5742a、5742b) の一部分を示す外観斜視図である。この変形例 4 に係るパッケージ 5740 の基板は、非透明部材からなる視認非容易部 5742b の一部に、内方に窪んだ矩形形状の切欠き部 5742c を形成しており、切欠き部 5742c を形成していない部位には複数 (この例では 3 つ) の端子 5746 を配置しているが、切欠き部 5742c には端子 5746 を 1 つも配置しないように構成している。

40

#### 【1493】

このような構成により、視認容易部と視認非容易部を別部材で構成する場合に比べ、パッケージ 5740 の基板のコストを低減することができるとともに、切欠き部 5742c を通してパッケージ 5740 の一方側から他方側を視認することができる。

50

## 【1494】

なお、切欠き部5742cは、パッケージ5740の基板の短手方向両側に形成してもよいし、短手方向片側のみに形成してもよいが、両側に形成すれば、視認性をより高めることができる場合がある。また、切欠き部5742cの数は一つでもよいし、複数でもよいが、複数形成すれば、視認性をより高めることができる場合がある。

## 【1495】

また、パッケージ5740に切欠き部5742cを形成することによってパッケージ5740の強度が低下し、例えば、基板やICソケットなどに実装された状態のパッケージ5740を取り外す際にパッケージ5740が破損しやすい状態（折れやすい状態）となるが、変形例4に係るパッケージ5740では、敢えてパッケージ5740の強度を低下させ、取り外しを困難にすることで、パッケージ5740が不正に取り外されて細工が施されるような不正行為を未然に防止できるように構成している。

## 【1496】

## &lt;パッケージの変形例／変形例5&gt;

図196(a)は、変形例5に係るパッケージ5750の外観斜視図である。この変形例5に係るパッケージ5750の基板は、全体が非透明部材で構成されている。また、平面視左側の一端部近傍と他端部近傍の各々に、内方に窪んだ矩形状の切欠き部5752を形成する一方で、平面視右側の中央近傍に、同形状の切欠き部5752を1つ形成し、3つの切欠き部5752には端子5756を1つも配置しないように構成している。

## 【1497】

このような構成により、視認容易部と視認非容易部を別部材で構成する場合に比べ、パッケージ5752の基板のコストを低減することができるとともに、平面視左側に配置する切欠き部5752の数および配置場所と、平面視右側に配置する切欠き部5752の数および配置場所を異ならせているため、例えば、右側から視認できない場所を左側から視認することができる場合があり、一層、視認性を高めることができる場合がある。

## 【1498】

## &lt;パッケージの変形例／変形例6&gt;

図196(b)は、変形例6に係るパッケージ5760の外観斜視図である。この変形例6に係るパッケージ5760の基板（上側基板5764、下側基板5762）は、全体が非透明部材で構成されている。また、パッケージ5760の長手方向中央近傍に、内方に窪んだ細径部5765を形成しており、上側基板5764と下側基板5762は、この細径部5765に向かって短手方向と厚み方向の外径が徐々に細くなる先細り形状とされている。また、細径部5765は、パッケージ5760全体の部位の中で最も厚みが薄く幅が狭い部位とされている。

## 【1499】

このような構成により、視認容易部と視認非容易部を別部材で構成する場合に比べ、パッケージ5760の基板のコストを低減することができるとともに、細径部5765の周囲の空間を通してパッケージ5760の一方側から他方側を様々な角度から視認でき、視認性を高めることができる場合がある。

## 【1500】

なお、細径部5765は、パッケージ5760の長手方向中央近傍以外の位置に形成してもよいし、複数形成してもよいが、細径部5765を複数形成すれば、視認性をより高めることができる場合がある。

## 【1501】

また、パッケージ5760に細径部5765を形成することによってパッケージ5760の強度が低下し、例えば、基板やICソケットなどに実装された状態のパッケージ5760を取り外す際にパッケージ5760が破損しやすい状態（折れやすい状態）となるが、変形例6に係るパッケージ5760では、敢えてパッケージ5760の強度を低下させ、取り外しを困難にすることで、パッケージ5760が不正に取り外されて細工が施されるような不正行為を未然に防止できるように構成している。

## 【1502】

＜パッケージの変形例／変形例7＞

図196(c)は、変形例7に係るパッケージ5770の外観斜視図である。この変形例7に係るパッケージ5770の基板（上側基板5774、下側基板5772）は、全体が非透明部材で構成されている。また、平面視左側および右側の長手方向中央近傍に、上側基板5774と下側基板5772の間に挟まれて長手方向中央近傍に配置されている基本回路5302を跨ぐようにして、内方に窪んだ矩形状の切欠き部5775を対向するように形成し、切欠き部5775には端子5776を1つも配置しないように構成している。

## 【1503】

このような構成により、視認容易部と視認非容易部を別部材で構成する場合に比べ、パッケージ5770の基板のコストを低減することができるとともに、平面視左側および右側に配置された複数の切欠き部5775を通してパッケージ5770の一方側から他方側を様々な角度から視認でき、視認性を高めることができる場合がある。

## 【1504】

また、平面視左側および右側の長手方向の同じ位置に切欠き部5775を形成することによってパッケージ5770の強度が低下し、基板やICソケットなどに実装された状態のパッケージ5770を取り外す際にパッケージ5770が破損しやすい状態（折れやすい状態）となるが、変形例7に係るパッケージ5770では、敢えてパッケージ5770の強度を低下させ、取り外しを困難にすることで、パッケージ5770が不正に取り外されて細工が施されるような不正行為を未然に防止できるように構成している。

## 【1505】

＜パッケージの変形例／変形例8＞

図196(d)は、変形例8に係るパッケージ5780の外観斜視図である。この変形例8に係るパッケージ5780の基板の平面視右側の端部には、ラウンド加工されたラウンド部5780aを形成しており、このラウンド部5780aには端子5786を配置しない一方で、基板の平面視左側の端部には、ラウンド部5780aに相当するラウンド加工を施さずに端子5786を配置するように構成している。すなわち、変形例8に係るパッケージ5780では、基板の平面視右側に配置される端子5786の数よりも、パッケージ5780の基板の平面視左側に配置される端子5786の数が多くなるように構成している。

## 【1506】

このような構成により、ラウンド部5780aの有無を手掛かりにパッケージ5780の向きを容易に確認することが可能となり、ICソケットへの逆挿しなどを未然に防止することができる上に、ラウンド部5780aによって形成される空間を通してパッケージ5780の一方側から他方側を視認できる場合がある。なお、変形例8に係るパッケージ5780では、上述の変形例7に係るパッケージ5770のように切欠き部5775を形成していないが、ラウンド部5780aに加えて（または替えて）、同一形状（または異なる形状）の切欠き部を形成してもよい。すなわち、他の実施例に係るパッケージの構成のうち、一つ、複数、または全ての構成を本実施例に適用してもよい（他の実施例についても同様）。

## 【1507】

＜パッケージのその他の変形例＞

図197(a)、(b)は、スリット状の切欠き部を形成したパッケージの一例を示した外観斜視図である。同図(a)に示すパッケージには、隣接する端子の間にスリット状の切欠き部を複数（この例では、3つ）形成しており、同図(b)に示すパッケージには、隣接する端子の間にスリット状の切欠き部を1つ形成している。このような構成によっても、切欠き部を通してパッケージの下方の空間を視認することができる。

## 【1508】

また、図198(a)は、円弧状の切欠き部を形成したパッケージの一例を示した外観

斜視図である。上述の例では、同図（b）に示すように、パッケージに矩形状の切欠き部を設けたが、この場合、切欠き部の加工が容易となる上に、パッケージの外形は矩形状であることが多いため、切欠き部を設けてもデザイン上で違和感を与えることが少ない。一方、同図（a）、（c）に示すような円弧状の切欠き部を設けた場合、デザイン性を高めることができる上に、切欠き部を他の部位に比べて目立たせることができるため、パッケージを検査する際などに注目を集めやすい場合がある。

#### 【1509】

また、図199（a）は、切欠き部の上面に、型番などを表すシールが貼付されたパッケージの一例を示した外観斜視図である。この例では、同図（b）に示すように、パッケージをICソケットや基板などから取り外そうとして切欠き部を中心としてパッケージが折れ曲がった場合などに、同図（c）に示すように、その痕跡がシールに残るように構成しているため、パッケージの取り外しが行われた否かを即座に判定できる場合がある。なお、型番などを切欠き部の上面に直接、印字（印刷）してもよく、この場合、パッケージをICソケットや基板などから取り外そうとして切欠き部を中心としてパッケージが折れ曲がったときに、印字部分の文字の形状が崩れたり、塗装が落ちたりすることにより、不正行為の痕跡を確実に残すことができる場合がある。

#### 【1510】

また、図200（a）～（g）は、パッケージの形状の変形例を示した図であり、同図（a）～（e）は変形例に係るパッケージの平面図、同図（f）～（g）は変形例に係るパッケージを長手方向から見た側面図である。

#### 【1511】

例えば、同図（a）に示すように、パッケージの長手方向中央近傍の両側に、内方に窪んだ平面視三角形の切欠き部を形成してもよいし、同図（b）に示すように、同図（a）の切欠き部に加えて、パッケージの底面に内方に窪んだ矩形状の切欠き部を形成してもよい。また、同図（c）に示すように、パッケージの長手方向端部近傍の両側に、内方に窪んだ平面視三角形の切欠き部を形成してもよいし、同図（d）に示すように、パッケージの長手方向一方側の端部近傍に、内方に窪んだ平面視三角形の切欠き部を形成する一方で、長手方向他方側の端部近傍に、同形状の切欠き部を形成してもよい。

#### 【1512】

また、同図（e）に示すように、パッケージの長手方向一方側の端部近傍に、内方に窪んだ平面視三角形の切欠き部を長手方向に複数並べて形成する一方で、長手方向他方側の端部近傍に、同形状の切欠き部を長手方向に複数並べて形成してもよい。さらに、同図（f）に示すように、パッケージの厚み方向の片面に、内方に窪んだ切欠き部を形成してもよいし、同図（g）に示すように、パッケージの厚み方向の両面に、内方に窪んだ切欠き部を形成してもよい。

#### 【1513】

##### <ICソケット>

次に、上述の変形例4に係るパッケージ5740が実装されるICソケットの一例について説明する。図201（a）は、ICソケット5850の外観斜視図であり、同図（b）は、ICソケット5850にパッケージ5740を実装した状態を示す外観斜視図である。

#### 【1514】

ICソケット5850は、矩形状の3つの開口部5852aが形成された板状体のベース5852と、このベース5852の長手方向両側に配置された複数のソケット端子5854と、を有して構成されている。ベース5852の長手方向両側には、パッケージ5740の端子5746に相当する位置に、この端子5746を挿入・固定するための円筒形状の複数のコネクタ部5852bが形成され、このコネクタ部5852bの各々からは、パッケージ5740の実装面とは反対方向に向けて棒状のソケット端子5854が突出形成されている。このような構成により、パッケージ5740をICソケット5850に実装した場合（パッケージ5740の端子5746を、ICソケット5850のコネクタ部

5852bに挿入・固定した場合)に、パッケージ5740の端子5746と、ICソケット5850のソケット端子5854が電氣的に接続される。

#### 【1515】

また、ICソケット5850の開口部5852aの一つは、パッケージ5740の切欠き部5742cに相当する位置に形成されており、パッケージ5740をICソケット5850に実装した場合でも、切欠き部5742cを通してパッケージ5740の一方側から他方側(例えば、パッケージ5740の正面視上側から下側のみならず、パッケージ5740の正面視斜め上側からパッケージ5740の下方空間の一部)を視認することができる。

#### 【1516】

10

特に、ICソケット5850のソケット端子5854は、ICソケット5850を基板に実装した場合に、ベース5852を基板から所定高さに浮かせて保持するだけの長さを有しているため、ソケット端子5854の側方からパッケージ5740の下側の空間を視認することができることに加えて、切欠き部5742cを通した視認性をより高めることができる場合がある。また、パッケージ5740をICソケット5850に実装した場合に、パッケージ5740の端子5746が、パッケージ5740をICソケット5850の実装面から所定高さに浮かせて保持するだけの長さを有していれば、パッケージ5740の端子5746の側方からパッケージ5740の下側の空間を視認することができることに加えて、切欠き部5742cを通した視認性をより高めることができる場合がある。

#### 【1517】

20

＜パッケージの端子＞

次に、変形例9に係るパッケージ5790の主要な端子について説明する。図202は、パッケージ5790の端子配置例を示す平面図である。なお、詳細は後述するが、同図における○(丸印)は、パッケージ5790が実装されるICソケットのソケット端子の位置(または、パッケージ5790がICソケットを介さずに基板に実装される場合には、基板のランドの位置)を示している。

#### 【1518】

同図に示すように、パッケージ5790の平面視左側には、端子番号1～35の35個の端子を上から下に向かって一列に配置し、パッケージ5790の平面視右側には、端子番号36～71の36個の端子を下から上に向かって一列に配置しているパッケージ5790の端子番号1～端子番号17の領域には、16本のアドレスバスA15～A0のうちの8本の上位アドレスバスA15～A8が配置されている。これらの上位アドレスバスA15～A8は、それぞれ端子番号15、13、11、9、7、5、3、1に配置されており、端子番号15から端子番号1にかけて一つ置きに配置されている。また、端子番号1～端子番号17の領域には、2つのグランド電位端子VSS5、VSS1が配置されており、グランド電位端子VSS5は端子番号1～端子番号17の領域の中央近傍(端子番号11)に、また、グランド電位端子VSS1は端子番号18に最も近い位置(端子番号17)に、それぞれ配置されている。

30

#### 【1519】

また、パッケージ5790の端子番号54～端子番号71の領域には、16本のアドレスバスA15～A0のうちの8本の下位アドレスバスA7～A0が配置されている。これらの上位アドレスバスA7～A0は、それぞれ端子番号55、57、59、61、63、65、67、69に配置されており、端子番号55から端子番号69にかけて一つ置きに配置されている。また、端子番号54～端子番号71の領域には、2つのグランド電位端子VSS4、VSS3が配置されており、グランド電位端子VSS4は端子番号53から最も遠い位置(端子番号71)に、また、グランド電位端子VSS3は端子番号53に最も近い位置(端子番号54)に、それぞれ配置されている。

40

#### 【1520】

また、パッケージ5790の端子番号18～端子番号35の領域には、正電源端子VDD1と、内蔵RAMのバックアップ用電源端子VBBと、グランド電位端子VSS2が配

50



置されており、正電源端子VDD1は端子番号17から2番目に近い位置（端子番号19）に、また、バックアップ用電源端子VBB1はVDD1の隣、すなわち端子番号17から3番目に近い位置（端子番号20）に、また、グランド電位端子VSS2は端子番号17から最も遠い位置（端子番号35）に、それぞれ配置されている。さらに、パッケージ5790の端子番号17に最も近い位置（端子番号18）には、クロック入力端子EXが配置されている。

#### 【1521】

また、パッケージ5790の端子番号36～端子番号53の領域には、正電源端子VDD25、VDD2と、グランド電位端子VSS6が配置されており、正電源端子VDD25は端子番号54に最も近い位置（端子番号53）に、また、正電源端子VDD2はVDD25の隣、すなわち端子番号54に2番目に近い位置（端子番号52）に、また、グランド電位端子VSS6は端子番号36～端子番号53の領域の中央近傍（端子番号44）に、それぞれ配置されている。さらに、パッケージ5790の端子番号54から遠い端子番号36から端子番号43には、8本のデータ入出力バスD7～D0が配置されている。

10

#### 【1522】

##### <ICソケットの変形例／変形例1>

次に、ICソケットの変形例について説明する。図203（a）は変形例1に係るICソケット5860の外観斜視図であり、同図（b）は、ICソケット5860にパッケージ5740を実装した状態を示す外観斜視図である。

#### 【1523】

変形例1に係るソケット5860は、上述のソケット5850が有する開口部5852aを備えていない点が異なっている。このような開口部5852aを備えないソケット5860にパッケージ5740を実装した場合でも、パッケージ5740が切欠き部5742cを備えているため、切欠き部5742cを通してパッケージ5740の一方側から他方側を視認することができる。

20

#### 【1524】

特に、パッケージ5740の端子5746が、パッケージ5740をICソケット5860に実装した場合に、パッケージ5740をICソケット5860から所定高さに浮かせて保持するだけの長さを有していれば、パッケージ5740の端子5746の側方からパッケージ5740の下側空間を視認することができることに加えて、切欠き部5742cを通した視認性をより高めることができる場合がある。一方、パッケージ5740とICソケット5860との間に形成される空間は、電子部品が配置できるほどの高さを有していないため、視認性を確保しながらも、同時に、不正な電子部品の取り付けなどを難しくしている。

30

#### 【1525】

##### <ICソケットの変形例／変形例2>

次に、ICソケットの変形例について説明する。図204（a）は変形例2に係るICソケット5870の外観斜視図であり、同図（b）は、ICソケット5870のソケット端子が実装される基板の平面図である。

#### 【1526】

変形例2に係るICソケット5870は、ベース5872に対するソケット端子5874の配置態様が上述のICソケット5850、5860と異なっている。具体的には、ICソケット5870では、パッケージの端子が挿入されるコネクタ部5872bが長手方向全域に亘って略等間隔に直線状に配置されているのに対して、各々のコネクタ部5872bから突出形成されるソケット端子5874が長手方向全域に亘って千鳥状に配置されている。また、同図（b）に示すように、ICソケット5870が実装される基板5878には、ICソケット5870のソケット端子5874の配置態様に合わせてランド5878aが千鳥状に配置・形成されている。

40

#### 【1527】

##### <コネクタ部とソケット端子>

50

図205(a)は、図204(a)におけるA-A線に沿った断面を模式的に示した図であり、図205(b)は、図204(a)におけるB-B線に沿った断面を模式的に示した図である。図205(a)に示すように、複数のソケット端子5874のうち、ICソケット5870の短手方向内側に突出するソケット端子5874は、コネクタ部5872bの軸心Laから短手方向内側方向に所定距離L8だけ離れた位置にある軸心Lbを有し、ICソケット5870の短手方向外側端面から所定距離L9だけ離れた位置に配置されている。一方、図205(b)に示すように、複数のソケット端子5874のうち、ICソケット5870の短手方向外側に突出するソケット端子5874は、コネクタ部5872bの軸心Laから短手方向外側方向に所定距離L8だけ離れた位置にある軸心Lcを有し、ICソケット5870の短手方向外側端面から所定距離L10 ( $L10 < L9$ ) だけ離れた位置に配置されている。

10

#### 【1528】

また、この例では、コネクタ部5872bの長さ、ソケット端子5874の長さは、ICソケット5870全体で同一長さとしてされている。すなわち、ICソケット5870を基板に実装し、ICソケット5870にパッケージを実装した場合に、パッケージの端子とコネクタ部5872bとの接点と、ソケット端子5874と基板の接点の2つの接点間の距離は、ICソケット5870全体でほぼ同一長さになるように構成されている。

#### 【1529】

＜コネクタ部とソケット端子の変形例／変形例1＞

図206(a)、(b)は、変形例1に係るコネクタ部とソケット端子の断面を模式的に示した図であり、それぞれ図205(a)、(b)に相当する図である。変形例1に係るコネクタ部には、図206(a)に示す第1コネクタ部5880と、同図(b)に示す第2コネクタ部5882の2種類があり、互いに形状が異なっている。なお、図示は省略するが、この例では、パッケージの端子が挿入される第1コネクタ部5880、第2コネクタ部5882がICソケットの長手方向全域に亘って略等間隔に直線状に配置されているのに対して、第1コネクタ部5880、第2コネクタ部5882から突出形成されるソケット端子5884がICソケットの長手方向全域に亘って千鳥状に配置されている。

20

#### 【1530】

第1コネクタ部5880は、軸心Ldに沿ってICソケットの表面から裏面まで真っ直ぐに延びる直線形状からなる。また、第2コネクタ部5882は、軸心Ldに沿ってICソケットの表面から裏面に向かって真っ直ぐに延びる第1直線部と、この第1直線部を基端として裏面に向かって斜めに延びる屈曲部と、この屈曲部を基端として軸心Ldとは異なる軸心Leに沿ってICソケットの裏面まで真っ直ぐに延びる第2直線部と、によって構成されている。一方、ソケット端子5884の長さは全て同一長さとしてされており、第1コネクタ部5880に接続されるソケット端子5884は、第1コネクタ部5880と同じ軸心Ld上に第1コネクタ部5880を基端として延出形成され、第2コネクタ部5882に接続されるソケット端子5884は、第2コネクタ部5882の第2直線部と同じ軸心Le上に第2コネクタ部5882の第2直線部を基端として延出形成されている。

30

#### 【1531】

したがって、この例では、第1コネクタ部5880の長さL11とソケット端子5884の長さの合計は、第2コネクタ部5882の長さL12 ( $L12 > L11$ ) とソケット端子5884の長さの合計よりも短くなっている。すなわち、このような構成を備えるICソケットを基板に実装し、ICソケットにパッケージを実装した場合に、パッケージの端子と第1コネクタ部5880との接点と、ソケット端子5884と基板の接点の2つの接点間の距離は、パッケージの端子と第2コネクタ部5882との接点と、ソケット端子5884と基板の接点の2つの接点間の距離よりも短くなるように構成されている。

40

#### 【1532】

＜コネクタ部とソケット端子の変形例／変形例2＞

図207(a)～(d)は、変形例2に係るコネクタ部とソケット端子の断面を模式的に示した図である。同図(a)、(b)に示す第1コネクタ部5892、第2コネクタ部

50

5893は、ICソケットの表面から裏面に向かって真っ直ぐに延びる直線部と、この直線部を基端として裏面まで斜めに延びる屈曲部と、によって構成されており、同図(a)に示す第1コネクタ部5892と同図(b)に示す第2コネクタ部5893で屈曲部の屈曲方向を反対にしている。なお、図示は省略するが、この例では、パッケージの端子が挿入される第1コネクタ部5892、第2コネクタ部5893の各々の直線部が、ICソケットの長手方向全域に亘って略等間隔に直線状に配置されているのに対して、第1コネクタ部5892、第2コネクタ部5893の各々の屈曲部を基端として突出形成されるソケット端子5894が、ICソケットの長手方向全域に亘って千鳥状に配置されている。

#### 【1533】

一方、同図(c)、(d)に示すコネクタ部5895は、ICソケットの表面から裏面まで真っ直ぐに延びる直線形状からなる。また、第1ソケット端子5896と第2ソケット端子5897は、コネクタ部5895を基端として斜めに延びる屈曲部と、この屈曲部を基端としてICソケットの厚み方向に真っ直ぐに延びる直線部と、によって構成されており、同図(c)に示す第1ソケット端子5896と同図(d)に示す第2ソケット端子5897で屈曲部の屈曲方向を反対にしている。なお、図示は省略するが、この例では、パッケージの端子が挿入されるコネクタ部5895が、ICソケットの長手方向全域に亘って略等間隔に直線状に配置されているのに対して、第1ソケット端子5896と第2ソケット端子5897が、ICソケットの長手方向全域に亘って千鳥状に配置されている。

#### 【1534】

##### < ICソケットの変形例／変形例3 >

図208(a)は変形例3に係るICソケット5900の外観斜視図である。上述のICソケット5850では、実装されるパッケージ5740の切欠き部5742cに相当する位置に開口部を形成したが、この例のように、実装されるパッケージ5740の切欠き部5742cに相当する位置に、ICソケット5900の両側をつなぐ補強部材5900aを設けてもよい。また、上述のICソケット5850では、パッケージ5740の切欠き部5742cに相当する位置に、コネクタ部やソケット端子を配置しないように構成したが、この例のように、パッケージ5740の切欠き部5742cに相当する位置に、コネクタ部5900bやソケット端子5900cを配置してもよい。このような構成とすれば、パッケージの一方側から他方側の視認性を確保しつつ、ICソケットの強度を高めたり、ソケット端子の本数を増やしたりすることができる場合がある。また、コネクタ部5900bやソケット端子5900cを配置することによって、パッケージ5740とICソケット5900との間に電子部品などを挿入して取り付けることが困難となるため、不正行為を未然に防止できる場合がある。

#### 【1535】

##### < ICソケットの変形例／変形例4 >

図208(b)は変形例4に係るICソケット5910の外観斜視図である。この例のように、実装されるパッケージ5740の切欠き部5742cに相当する位置に、切欠き部5742cの形状や大きさに合わせた凹部5910aを形成してもよい。このような構成とすれば、パッケージの切欠き部だけでなくICソケット5910の凹部5910aを通してパッケージの一方側から他方側を視認でき、視認性を高めることができる場合がある。

#### 【1536】

##### < ICソケットの変形例／変形例5 >

図208(c)は変形例5に係るICソケット5920の外観斜視図である。この例のように、パッケージ5740の切欠き部5742cに相当する位置に、切欠き部5742cの形状や大きさに合わせた(例えば、切欠き部5742cに嵌合する形状で透明部材からなる)凸部5920aを形成してもよい。このような構成とすれば、凸部5920aによってパッケージとの間に隙間を作らないようにすることで、パッケージの裏に不正器具などが設置されることを未然に防止できる場合があり、特に、凸部5920aを透明部材や半透明部材で構成すれば、凸部5920aを通してパッケージの一方側から他方側を視

認でき、不正器具などが設置されたことを容易に発見できる場合がある。なお、ＩＣパッケージの形状は、同図（ａ）～（ｃ）に示すように、パッケージの形状に合わせた形状に限定されず、例えば、ＩＣパッケージが実装される基板の形状に合わせた形状（例えば、基板が凸部を備える場合に、この凸部と相補的形状を有する凹部を備えるＩＣソケットなど）としてもよい。

#### 【１５３７】

また、図２０９（ａ）、（ｂ）は、ＩＣソケットを基板に実装した例を示した外観斜視図である。同図（ａ）に示す例では、ＩＣソケット５９３０の厚みを、基板５９３２に実装されている、その他の電子部品の厚みよりも薄くしている。このような構成とすれば、ＩＣソケット５９３０にパッケージを実装した場合でも、他の電子部品によってパッケージを見ずらくし、パッケージの存在を分かりにくくすることができるため、パッケージに対する不正行為を防止できる場合がある。また、上述の視認容易部や切欠き部が配置される、ＩＣソケットの短手方向には、電子部品を配置しないように構成しているため、視認容易部や切欠き部を介して、ＩＣソケットに実装されたパッケージの裏側などを確実に視認できる場合がある。

10

#### 【１５３８】

また、同図（ｂ）に示す例では、ＩＣソケット５９４０の厚みを、ＩＣソケット５９４０の長手方向一方側の基板５９４２上に実装されている複数の電子部品５９４４の厚みよりも厚くしている。このような構成とすれば、ＩＣソケット５９４０の長手方向一方側からＩＣソケット５９４０を観察した場合に、ＩＣソケット５９４０の存在が確認し易くなるため、パッケージの実装や取り外しが容易となり、作業性を高めることができる場合がある。また、同図（ａ）の例に比べて、パッケージの熱を基板５９４２外部に逃がしやすくなるため、熱に起因する不具合を未然に防止できる場合がある。なお、この例では、電子部品５９４４の配置場所をＩＣソケット５９４０の長手方向一方側としたが、長手方向両側でもよいし、長手方向以外の方向でもよい。

20

#### 【１５３９】

また、ＩＣソケット５９４０の長手方向のうち、ＩＣソケット５９４０の延長上には、電子部品を配置しないことが好ましい。このような構成とすれば、ＩＣソケットの長手方向から、ＩＣソケットに実装されたパッケージの裏側などを視認できる場合がある。

#### 【１５４０】

また、同図（ｃ）に示すように、基板５９５０の表面を覆う透明部材からなる上ケース５９５３と、基板５９５０の裏面を覆う下ケース５９５４を備える場合、上ケース５９５３における、基板５９５０に実装されたＩＣソケット５９５１やパッケージ５９５２が配置される位置に、ＩＣソケット５９５１やパッケージ５９５２に向かって凹んだ凹部５９５３ａを形成してもよい。このような構成とすれば、上ケース５９５３によって基板５９５０が保護されているような場合でも、ＩＣソケット５９５１やパッケージ５９５２の裏側を確実に視認できる場合がある。

30

#### 【１５４１】

##### <配線パターン>

次に、上述のＩＣソケットが実装される基板の配線パターンについて説明する。図２１０（ａ）は、ＩＣソケットが実装されるランドと、これらのランドと乱数用水晶発振器５３１４ａおよびシステム用水晶発振器５３１４ｂを接続する信号線の一例を示した図であり、同図（ｂ）は、ランド近傍の部分拡大図である。また、図２１１は、ＩＣソケットの外縁を説明するための図である。

40

#### 【１５４２】

この例では、基板１００２上に、ＩＣソケットが実装される（ＩＣソケットのソケット端子が接続される）複数のランド６００４ａ、６００４ｂ（一部のみ図示）が形成されているとともに、ＩＣソケットの実装部から所定の距離を置いて、乱数用水晶発振器５３１４ａとシステム用水晶発振器５３１４ｂが実装されている。ランド６００４ａ、６００４ｂは、基板６００２に実装されるＩＣソケットのソケット端子に対応して配置されており

50

、この例では、I Cソケットが基板6002に実装された場合のI Cソケットの外縁を表す外縁ラインOLから所定距離L21だけ離れた位置を中心とする円形状の外側ランド6004aと、外縁ラインOLから所定距離L22 ( $L22 > L21$ ) だけ離れた位置を中心とする円形状の内側ランド6004bの2種類がある。

#### 【1543】

ここで、「I Cソケットの外縁」とは、図211に示すように、基板6002に実装された状態におけるI Cソケット6001の短手方向外側の面6001aを含む仮想平面VPをいい、この仮想平面VPと基板6002の表面とが交わって形成される線が、上述の外縁ラインOLである。

#### 【1544】

システム用水晶発振器5314bは、第1信号線6006を介して外側ランド6004aに接続され、乱数用水晶発振器5314aは、第2信号線6008を介して内側ランド6004bに接続されている。第1信号線6006の長さは、システム用水晶発振器5314bの外部クロックEX端子から外縁ラインOLまでの距離L20と、外縁ラインOLから外側ランド6004a (この例では、I Cソケットのコネクト部を介してパッケージのEX端子に接続されるランド) の中心までの距離L21を足した数値となり、第2信号線6008の長さは、乱数用水晶発振器5314aの乱数用クロックRCK端子から外縁ラインOLまでの距離L20と、外縁ラインOLから内側ランド6004b (この例では、I Cソケットのコネクト部を介してパッケージのRCK端子に接続されるランド) の中心までの距離L22を足した数値となる。すなわち、この例では、第1信号線6006の長さ ( $L20 + L21$ ) よりも、第2信号線6008の長さ ( $L20 + L22$ ) を長くしている。さらに、この例では、外側ランド6004aからパッケージのEX端子までの距離よりも、内側ランド6004bからパッケージのRCK端子までの距離が近くなるように、すなわち、内側ランド6004bからパッケージのRCK端子を通して入力する乱数用クロックRCKの信号が、パッケージから発生するノイズなどの影響を受けやすいように設定している。

#### 【1545】

また、図212は、基板6002を側方から見た側面図である。第1信号線6006は、平板状の基板表面に沿ってシステム用水晶発振器5314bとI Cソケットを高低差なく真っ直ぐに接続しており、第2信号線6008は、平板状の基板表面に沿って乱数用水晶発振器5314aとI Cソケットを高低差無く真っ直ぐに接続している。

#### 【1546】

このような構成により、システム用水晶発振器5314bをパッケージ (基本回路5302) の近傍に配置することができ、且つ、パッケージとシステム用水晶発振器5314bを結ぶ第1信号線6006の長さを短くすることができるため、システム用水晶発振器5314bから出力される外部クロックEXの信号を、安定して基本回路5302のクロック回路5340に供給することができるとともに、この外部クロックEXの信号に基づいて生成されるシステムクロックSCLKの信号も、安定してCPUコアや内部の各回路に供給することができる。

#### 【1547】

また、クロック信号の供給源と供給先を結ぶ信号線の長さは、クロック信号に外乱が加わらないようになるべく短く設定するのが一般的であるが、基本回路5302と乱数用水晶発振器5314aを結ぶ第2信号線6008の長さを敢えて長めに設定し、第2信号線6008を通る乱数用クロックRCKの信号が外乱の影響を受けやすいように構成することによって、乱数用クロックRCKの信号に基づいてカウント値のカウントを行う乱数回路5316のカウント値の更新タイミングにバラツキを与え、乱数回路5316が生成する乱数値のランダム性をさらに高めることができる。

#### 【1548】

特に、第1信号線6006のうち、I Cソケットによって隠れない部位 (システム用水晶発振器5314bの外部クロックEX端子から外縁ラインOLまでの信号線) と、第2

10

20

30

40

50

信号線 6008 のうち、IC ソケットによって隠れない部位（乱数用水晶発振器 5314 a の乱数用クロック RCK 端子から外縁ライン OL までの信号線）は同じ長さに設定されているため、一見しただけでは第 2 信号線 6008 の長さが長めに設定されていることを把握することができない。このため、不正行為者が不正行為を行う可能性が高まるおそれがあるが、乱数用クロック RCK の信号の異常を検出することができるため、不正行為を確実に発見できる。

#### 【1549】

＜配線パターンの変形例／変形例 1＞

図 213 は、変形例 1 に係る配線パターンを示した図である。この例では、基板 6012 上に、IC ソケットが実装される外側ランド 6014 a、内側ランド 6014 b（一部のみの図示）が形成されているとともに、（乱数用水晶発振器 5314 a を兼ねる）システム用水晶発振器 5314 b が実装されている。

10

#### 【1550】

システム用水晶発振器 5314 b は、第 1 信号線 6016 を介して外側ランド 6014 a に接続されるとともに、第 2 信号線 6018 を介して内側ランド 6014 b に接続される。第 1 信号線 6016 の長さは、システム用水晶発振器 5314 b の外部クロック EX 端子から外縁ライン OL までの距離 L30 と、外縁ライン OL から外側ランド 6014 a（この例では、IC ソケットのコネクタ部を介してパッケージの RCK 端子に接続されるランド）の中心までの距離 L31 を足した数値となり、第 2 信号線 6018 の長さは、システム用水晶発振器 5314 b の外部クロック EX 端子から外縁ライン OL までの距離 L32（ $L32 > L30$ ）と、外縁ライン OL から内側ランド 6014 b（この例では、IC ソケットのコネクタ部を介してパッケージの RCK 端子に接続されるランド）の中心までの距離 L33（ $L33 > L31$ ）を足した数値となる。すなわち、この例では、第 1 信号線 6016 の長さ（ $L30 + L31$ ）よりも、第 2 信号線 6018 の長さ（ $L32 + L33$ ）を長くしている。

20

#### 【1551】

このような構成により、システム用水晶発振器 5314 b をパッケージ（基本回路 5302）の近傍に配置することができ、且つ、パッケージとシステム用水晶発振器 5314 b を結ぶ第 1 信号線 6016 の長さを短くすることができるため、システム用水晶発振器 5314 b から出力される外部クロック EX の信号を、安定して基本回路 5302 のクロック回路 5340 に供給することができるとともに、この外部クロック EX の信号に基づいて生成されるシステムクロック SCLK の信号も、安定して CPU コアや内部の各回路に供給することができる。

30

#### 【1552】

また、クロック信号の供給源と供給先を結ぶ信号線の長さは、クロック信号に外乱が加わらないようにならなく短く設定するのが一般的であるが、第 2 信号線 6018 の長さを敢えて長めに設定し、第 2 信号線 6018 を通る乱数用クロック RCK の信号が外乱の影響を受けやすいように構成することによって、乱数用クロック RCK の信号に基づいてカウント値のカウントを行う乱数回路 5316 のカウント値の更新タイミングにバラツキを与え、乱数回路 5316 が生成する乱数値のランダム性をさらに高めることができる。

40

#### 【1553】

また、第 1 信号線 6016 と第 2 信号線 6018 の一部を共有化することによって、配線パターンを簡素化できる場合がある。

#### 【1554】

＜配線パターンの変形例／変形例 2、3＞

図 214（a）は、変形例 2 に係る配線パターンを示した図である。この例は、上記図 210（a）に示した配線パターンと略同一であるが、システム用水晶発振器 5314 b が接続される外側ランド 6004 a と、乱数用水晶発振器 5314 a が接続される内側ランド 6004 b との間に、2 つの外側ランド 6004 a を配置している。そして、一方の外側ランド 6004 a をパッケージの第 1 の電源端子（例えば、上述の正電源端子 VDD

50

1) に接続し、他方の外側ランド 6 0 0 4 a をパッケージの第 2 の電源端子 (例えば、上述のバックアップ用電源端子 V B B) に接続する点が異なっている。

【1 5 5 5】

一方、同図 (b) は、変形例 3 に係る配線パターンを示した図である。この例は、同図 (a) に示した配線パターンと類似しているが、パッケージの第 1 の電源端子に接続された外側ランド 6 0 0 4 a からの電源線 6 0 3 3 を、スルーホールを介して基板 6 0 3 2 の裏面に配線し、パッケージの第 2 の電源端子に接続された外側ランド 6 0 0 4 a からの電源線 6 0 3 4 を、基板 6 0 3 2 の表面に乱数値水晶発振器 5 3 1 4 a の近傍を通して配線している。

【1 5 5 6】

このような構成により、第 1 信号線 6 0 0 6 を通る外部クロック E X の信号が電源線による影響を受けにくくすることができ、システム用水晶発振器 5 3 1 4 b から出力される外部クロック E X の信号を、安定して基本回路 5 3 0 2 のクロック回路 5 3 4 0 に供給することができるとともに、この外部クロック E X の信号に基づいて生成されるシステムクロック S C L K の信号も、安定して C P U コアや内部の各回路に供給することができる。

【1 5 5 7】

また、第 2 信号線 6 0 0 8 の近傍に電源線 6 0 3 4 を配線して第 2 信号線 6 0 0 8 を通る乱数用クロック R C K の信号が外乱の影響を受けやすいように構成することによって、乱数用クロック R C K の信号に基づいてカウント値のカウントを行う乱数回路 5 3 1 6 のカウント値の更新タイミングにバラツキを与え、乱数回路 5 3 1 6 が生成する乱数値のランダム性をさらに高めることができる。

【1 5 5 8】

＜配線パターンの変形例／変形例 4＞

図 2 1 5 は、変形例 4 に係る配線パターンを示した図である。この例では、システム用水晶発振器 5 3 1 4 b が接続されるランド 6 0 4 4 b の上側に配置されたランド 6 0 4 4 a が、パッケージのグランド端子 (例えば、上述のグランド電位端子 V S S 1) に接続されているとともに、システム用水晶発振器 5 3 1 4 b が接続されるランド 6 0 4 4 b の下側に配置されたランド 6 0 4 4 c が、パッケージの第 1 の電源端子 (例えば、上述の正電源端子 V D D 1) に接続されている。また、乱数用水晶発振器 5 3 1 4 a が接続されるランド 6 0 4 4 e の上側に配置されたランド 6 0 4 4 d が、パッケージの第 2 の電源端子 (例えば、上述のバックアップ用電源端子 V B B) に接続されているとともに、乱数用水晶発振器 5 3 1 4 a が接続されるランド 6 0 4 4 e の下側に配置されたランド 6 0 4 4 f が、パッケージの制御信号線 (例えば、チップセレクト端子) に接続されている。

【1 5 5 9】

このように、第 1 信号線 6 0 0 6 の近傍に信号の変化が少ない線 (電源線、グランド線) を配置することで、第 1 信号線 6 0 0 6 を通る外部クロック E X の信号が影響を受けにくくすることができ、システム用水晶発振器 5 3 1 4 b から出力される外部クロック E X の信号を、安定して基本回路 5 3 0 2 のクロック回路 5 3 4 0 に供給することができるとともに、この外部クロック E X の信号に基づいて生成されるシステムクロック S C L K の信号も、安定して C P U コアや内部の各回路に供給することができる。

【1 5 6 0】

また、第 2 信号線 6 0 0 8 の近傍に信号の変化が多い線 (制御信号線) を配置することで、第 2 信号線 6 0 0 8 を通る乱数用クロック R C K の信号が外乱の影響を受けやすいように構成することによって、乱数用クロック R C K の信号に基づいてカウント値のカウントを行う乱数回路 5 3 1 6 のカウント値の更新タイミングにバラツキを与え、乱数回路 5 3 1 6 が生成する乱数値のランダム性をさらに高めることができる。

【1 5 6 1】

＜配線パターンの変形例／変形例 5＞

図 2 1 6 (a) は、変形例 5 に係る配線パターンを示した図である。この例では、基板上の外縁ライン O L から所定距離 L 4 1 だけ離れた位置に、3 つのランド 6 0 4 0 a、6

10

20

30

40

50

040b、6040cが配置され、ランド6040aがグラウンド線6042に接続され、ランド6040bがシステム用水晶発振器のシステムクロック線6044に接続され、ランド6040cが電源線6046に接続されている。また、グラウンド線6042が接続されたランド6040aの外縁と、システムクロック線6044が接続されたランド6040bの外縁との間の距離W1と、システムクロック線6044が接続されたランド6040bの外縁と、電源線6046が接続されたランド6040cの外縁との間の距離W1が同一距離とされている。

#### 【1562】

これに対して、同図(b)に示す例では、基板上の外縁ラインOLから所定距離L41だけ離れた位置に、グラウンド線6042が接続されたランド6040aと、電源線6046が接続されたランド6040cが配置され、基板上の外縁ラインOLから所定距離L42 ( $L42 > L41$ )だけ離れた位置に、システムクロック線6044が接続されたランド6040bが配置されている。また、ランド6040bをランド6040a、6040cよりも外縁ラインOLから離れた位置に配置することによって、グラウンド線6042が接続されたランド6040aの外縁と、システムクロック線6044の外縁との間の距離W2と、システムクロック線6044の外縁と、電源線が接続されたランド6040cの外縁との間の距離W2の両方を、同図(a)に示す距離W1よりも長くしている。

#### 【1563】

また、同図(c)に示す例では、基板上の外縁ラインOLから所定距離L42だけ離れた位置に、グラウンド線6042が接続されたランド6040aと、電源線6046が接続されたランド6040cが配置され、基板上の外縁ラインOLから所定距離L41 ( $L41 < L42$ )だけ離れた位置に、システムクロック線6044が接続されたランド6040bが配置されている。また、ランド6040a、6040cをランド6040bよりも外縁ラインOLから離れた位置に配置することによって、グラウンド線6042の外縁と、システムクロック線6044が接続されたランド6040bの外縁との間の距離W2と、システムクロック線6044が接続されたランド6040bと、電源線6046の外縁との間の距離W2の両方を、同図(a)に示す距離Laよりも長くしている。

#### 【1564】

同図(b)、(c)に示す構成によれば、システムクロック線6044を、グラウンド線6042や電源線6046から、ランドの膨らみに相当する距離だけ離れた位置に配置することができ、システムクロック線6044を通る外部クロックEXの信号が影響を受けにくくすることができ、システム用水晶発振器5314bから出力される外部クロックEXの信号を、安定して基本回路5302のクロック回路5340に供給することができる。また、この例では、グラウンド線6042の外縁と、システムクロック線6044が接続されたランド6040bの外縁との間の距離と、システムクロック線6044が接続されたランド6040bと、電源線6046の外縁との間の距離を同じ距離W2としたが、例えば、一方を他方よりも長くしてもよい。

#### 【1565】

<配線パターンの変形例／変形例6>

図217は、変形例6に係る配線パターンを示した図である。この例では、基板上の外縁ラインOLから所定距離L51だけ離れた位置に、2つのランド6050a、6050bが配置され、基板上の外縁ラインOLから所定距離L52 ( $L52 > L51$ )だけ離れた位置に、1つのランド6050cが配置されている。また、ランド6050aはシステム用水晶発振器5314bの外部クロックEX端子に接続されており、両者を接続する信号線は屈曲されずに直線形状とされている。一方、ランド6050cは乱数用水晶発振器5314bの乱数用クロックRCK端子に接続されており、両者を接続する信号線は途中で屈曲されている。

#### 【1566】



このような構成により、システム用水晶発振器 5 3 1 4 b をパッケージ（基本回路 5 3 0 2）の近傍に配置することができ、且つ、パッケージとシステム用水晶発振器 5 3 1 4 b を結ぶ第 1 信号線 6 0 5 2 の長さを短くすることができるため、システム用水晶発振器 5 3 1 4 b から出力される外部クロック E X の信号を、安定して基本回路 5 3 0 2 のクロック回路 5 3 4 0 に供給することができるとともに、この外部クロック E X の信号に基づいて生成されるシステムクロック S C L K の信号も、安定して C P U コアや内部の各回路に供給することができる。

#### 【1 5 6 7】

また、クロック信号の供給源と供給先を結ぶ信号線の長さは、クロック信号に外乱が加わらないようになるべく短く設定するのが一般的であるが、基本回路 5 3 0 2 と乱数用水晶発振器 5 3 1 4 a を結ぶ第 2 信号線 6 0 5 4 の長さを取って長めに設定し、第 2 信号線 6 0 5 4 を通る乱数用クロック R C K の信号が外乱の影響を受けやすいように構成することによって、乱数用クロック R C K の信号に基づいてカウント値のカウントを行う乱数回路 5 3 1 6 のカウント値の更新タイミングにバラツキを与え、乱数回路 5 3 1 6 が生成する乱数値のランダム性をさらに高めることができる。

#### 【1 5 6 8】

##### < 基板 >

図 2 1 8 は、上述の I C ソケットを実装する基板の平面図である。同図に示す基板 6 0 5 0 は、I C ソケットのソケット端子を実装するための複数のランド 6 0 5 2 を有する矩形状の I C ソケット実装部 6 0 5 4 を 4 つ備えるとともに、この I C ソケット実装部 6 0 5 4 の周囲をグランド領域 6 0 5 6 としている。また、複数のランド 6 0 5 2 のうち、パッケージのグランド端子に接続される 2 つのランド 6 0 5 2 a は、信号線などを介することなく、直接、グランド領域 6 0 5 6 と連通されている。なお、本実施形態に係る基板は、部品面でもよく、ハンダ面でもよい。

#### 【1 5 6 9】

このような構成により、基板とランドを接続するグランド線を従来よりも削減できる場合がある上に、回路変更などによってグランド電位が取りたい場合にグランド電位を取りやすくすることができ、利便性を高めることができる場合がある。

#### 【1 5 7 0】

##### < 基板の変形例／変形例 1 >

図 2 1 9 は、変形例 1 に係る基板の平面図である。同図に示す基板 6 0 6 0 は、I C ソケットのソケット端子を実装するための複数のランド 6 0 6 2 を有する 4 つの I C ソケット実装部 6 0 6 4 を備えるとともに、基板 6 0 6 0 に実装される I C ソケットの一方側から他方側を跨ぐようにして電源領域 6 0 6 8 が配置されている。また、複数のランド 6 0 6 2 のうち、パッケージの電源端子にされる 2 つのランド 6 0 6 2 a は、信号線などを介することなく、直接、電源領域 6 0 6 8 と連通されている。

#### 【1 5 7 1】

このような構成により、基板とランドを接続する電源線を従来よりも削減できる場合がある上に、回路変更などによって電源が取りたい場合に電源を取りやすくすることができ、利便性を高めることができる場合がある。

#### 【1 5 7 2】

##### < 基板の変形例／変形例 2 >

図 2 2 0 は、変形例 2 に係る基板の平面図である。同図に示す基板 6 0 7 0 は、I C ソケットのソケット端子を実装するための複数のランド 6 0 7 2 を有する矩形状の 4 つの I C ソケット実装部 6 0 7 4 を備えているとともに、これらの I C ソケット実装部 6 0 7 4 の各々には、複数（この例では 4 つ）のランド 6 0 7 2 が千鳥状に配置されている。また、この例では、複数のランド 6 0 7 2 のうち、I C ソケットの外側に配置されたソケット端子に対応するランドのみを、基板 6 0 7 0 に実装された電子部品などに接続し、内側に配置されたランドは、電子部品などに接続せずに未使用としている。なお、この例では、ランドが千鳥状に配置された実装部を 4 つ備えたが、4 つの実装部のうちの一つ、複数、

10

20

30

40

50

または全ての実装部においてランドを千鳥状に配置してもよく、また、実装部の数も４つに限定されない。

#### 【１５７３】

このような構成により、隣接する信号線の間に所定の距離を設けることができ、他の信号線によるノイズなどの影響を減らすことができる上に、回路設計が容易になる場合がある。

#### 【１５７４】

＜基板の変形例／変形例３＞

図２２１は、変形例３に係る基板の平面図である。同図に示す基板６０８０は、ＩＣソケットのソケット端子を実装するための複数（この例では１６）のランド６０８２を備えており、これらのランド６０８２が千鳥状に配置されている。また、この例では、複数のランド６０８２のうち、ＩＣソケットの外側に配置されたソケット端子に対応するランドのみを、基板６０８０に実装された電子部品などに接続し、内側に配置されたランドは、電子部品などに接続せずに未使用としている。また、この例では、基板６０８０の全ての領域のうち、複数のランド６０８２と、電子部品などへの配線を除いた領域をグランド領域６０８４としている。

10

#### 【１５７５】

このような構成により、基板とランドを接続するグランド線を従来よりも削減できる場合がある上に、回路変更などによってグランド電位が取りたい場合にグランド電位を取りやすくすることができ、利便性を高めることができる場合がある。また、隣接する信号線の間に所定の距離を設けることができ、他の信号線によるノイズなどの影響を減らすことができる上に、回路設計が容易になる場合がある。

20

#### 【１５７６】

以上説明したように、本実施形態に係るパチンコ機５１００は、遊技制御を行う遊技制御用のマイクロコンピュータ（例えば、図１９３（ａ）に示すパッケージ５６８０）が設けられた基板を有する主制御手段（例えば、主制御部５３００）と、前記主制御手段の遊技制御に基づいて遊技の演出制御を行う副制御手段（例えば、第１副制御部５４００）と、を備えた遊技台であって、前記マイクロコンピュータは、端子（例えば、端子５６８２）を設けることで該端子によって、該マイクロコンピュータの裏面における第一の領域、該裏面に対向する前記基板の表面における第二の領域および前記裏面と前記表面との間の空間における第三の領域のうちの少なくとも一つである特定の領域を視認困難にする視認非容易部（例えば、視認非容易部５６８０ａ）と、前記端子を設けないことによって、前記視認非容易部よりも前記特定の領域を視認容易にする視認容易部（例えば、視認容易部６８０ｂ）と、を有して構成されていることを特徴とする遊技台である。

30

#### 【１５７７】

換言すれば、前記マイクロコンピュータは、端子を設けることで該端子によって視認困難となる範囲を有する視認非容易部と、前記端子を設けないことで前記非視認容易部よりも視認困難となる範囲が狭くなる視認容易部と、を有して構成されていることを特徴とする遊技台である。なお、本発明に係る「視認困難」とは、特定の領域を完全に見ることができない場合（視認が不可能な場合）に限定されず、視認容易部に比べて特定の領域が僅かに視認し難い状態も含む概念である。一方、本発明に係る「視認容易」とは、特定の領域を全て見ることができる場合に限定されず、視認非容易部に比べて特定の領域が僅かに視認し易い状態も含む概念である。また、本発明に係る「特定の領域」は、マイクロコンピュータの端子の表面や、マイクロコンピュータの端子と基板間を接続するためのＩＣソケットの表面などを含む概念である。

40

#### 【１５７８】

本実施形態に係るパチンコ機５１００によれば、視認容易部を通して遊技制御用マイクロコンピュータの裏側近傍の特定の領域を視認することができ、マイクロコンピュータの裏側などに不正部品を取り付けるような不正行為を未然に防止することができる場合がある。また、視認非容易部によって特定の領域の存在をメーカやパチンコ店以外の人には知

50

られ難くする一方で、視認容易部によってメーカやパチンコ店の人には特定の領域に不正部品が設けられていないかどうかを確認しやすくすることができ、不正改造を防止できる場合がある。

また、前記視認非容易部における端子は、前記マイクロコンピュータの短手方向両側のみ設けられていてもよい（例えば、図１９６（ｂ）に示すパッケージ５７６０、図１９６（ｃ）に示すパッケージ５７７０などを参照）。

#### 【１５７９】

このような構成とすれば、短手方向両側の視認容易部を通して遊技制御用マイクロコンピュータの裏側近傍の特定の領域を様々な角度から視認することができるため、さらに不正部品などを見つけることが容易となる場合がある。

10

#### 【１５８０】

また、前記視認容易部（例えば、図１９５（ｃ）に示す視認非容易部５７３２ａ、図１９５（ｄ）に示す切欠き部５７４２ｃ、図１９７（ａ）、（ｂ）に示すスリット、図１９６（ｄ）に示すラウンド部５７８０ａ、図１９８（ａ）に示す円弧状の切欠き部）が、前記遊技制御用マイクロコンピュータの短手方向片側に設けられていてもよい。

#### 【１５８１】

このような構成とすれば、短手方向片側の視認容易部を通して遊技制御用マイクロコンピュータの裏側近傍の特定の領域を視認することができるため、さらに不正部品などを見つけることが容易となる場合がある。

#### 【１５８２】

また、前記遊技制御用マイクロコンピュータには長手方向に沿って複数の端子が配置され、前記視認容易部（例えば、図１９６（ｂ）に示す細径部５７６５、図１９６（ｃ）に示す切欠き部５７７５、図１９６（ａ）、（ｂ）に示すスリット）が、前記端子間に設けられていてもよい。

20

#### 【１５８３】

このような構成とすれば、多くの端子を配置することが可能でありながら、端子間の視認容易部を通して遊技制御用マイクロコンピュータの裏側近傍の特定の領域を視認することができるため、さらに不正部品などを見つけることが容易となる場合がある。

#### 【１５８４】

また、前記視認容易部（例えば、図１９６（ｂ）に示す細径部５７６５、図１９６（ｃ）に示す切欠き部５７７５、図２００（ａ）～（ｃ）に示す平面視三角形の切欠き部）が、前記短手方向両側に対向して設けられていてもよい。

30

#### 【１５８５】

このような構成とすれば、対向配置された視認容易部を通して遊技制御用マイクロコンピュータの裏側を視認することができるため、さらに不正部品などを見つけることが容易となる場合がある。

#### 【１５８６】

なお、上記実施形態においては、遊技球を遊技媒体としたパチンコ機の例を示したが、本発明はこれに限定されるものではなく、例えば、以下に説明するような、メダル（コイン）を遊技媒体としたスロットマシンなどにも適用可能である。

40

#### 【１５８７】

##### <全体構成>

まず、図２２２を用いて、他の例に係るスロットマシン６１００の全体構成について説明する。なお、同図はスロットマシン６１００の外観斜視図を示したものである。

#### 【１５８８】

スロットマシン６１００は、略箱状の本体６１０１と、この本体６１０１の前面開口部に取り付けられた前面扉６１０２とを有して構成されている。スロットマシン６１００の本体６１０１の中央内部には、外周面に複数種類の図柄が所定コマ数だけ配置されたリールが３個（左リール６１１０、中リール６１１１、右リール６１１２）収納され、スロットマシン６１００の内部で回転できるように構成されている。各図柄は帯状部材に等間隔

50

で適当数印刷され、この帯状部材が所定の円形枠材に貼り付けられて各リール6110乃至6112が構成されている。リール6110乃至6112上の図柄は、遊技者から見ると、図柄表示窓6113から縦方向に概ね3つ表示され、合計9つの図柄が見えるようになっている。そして、各リール6110乃至6112を回転させることにより、遊技者から見える図柄の組み合わせが変動することとなる。なお、本実施例1では、3個のリールをスロットマシン6100の中央内部に備えているが、リールの数やリールの設置位置はこれに限定されるものではない。

#### 【1589】

また、図柄表示窓6113の外枠には、点滅や点灯などの点灯制御によって、後述する有効ラインや入賞ラインを報知するためのライン表示LED（図示省略）が配置されている。

10

#### 【1590】

さらに、スロットマシン6100内部において各々のリール6110乃至6112の近傍には、投光部と受光部からなる光学式センサ（図示省略）が設けられており、この光学式センサの投光部と受光部の間を、リールに設けられた一定の長さの遮光片が通過するように構成されている。このセンサの検出結果に基づいてリール上の図柄の回転方向の位置を判断し、目的とする図柄が入賞ライン6114上に表示されるようにリール6110乃至6112を停止させる。

#### 【1591】

入賞ライン表示ランプ6120は、有効となる入賞ラインを示すランプである。有効となる入賞ラインは、スロットマシン6100に投入されたメダルの数によって予め定められている。5本の入賞ライン6114のうち、例えば、メダルが1枚投入された場合、中段の水平入賞ラインが有効となり、メダルが2枚投入された場合、上段水平入賞ラインと下段水平入賞ラインが追加された3本が有効となり、メダルが3枚投入された場合、右下り入賞ラインと右上り入賞ラインが追加された5本が入賞ラインとして有効になる。なお、入賞ライン6114の数については5本に限定されるものではない。

20

#### 【1592】

スタートランプ6121は、リール6110乃至6112が回転することができる状態にあることを遊技者に知らせるランプである。再遊技ランプ6122は、前回の遊技において入賞役の一つである再遊技役に入賞した場合に、今回の遊技が再遊技可能であること（メダルの投入が不要であること）を遊技者に知らせるランプである。告知ランプ6123は、内部抽選において、特定の入賞役（例えば、BB（ビッグボーナス）やRB（レギュラーボーナス）等のボーナス）に内部当選していることを遊技者に知らせるランプである。メダル投入ランプ6124は、メダルの投入が可能であることを知らせるランプである。払出枚数表示器6125は、何らかの入賞役に入賞した結果、遊技者に払出されるメダルの枚数を表示するための表示器である。遊技回数表示器6126は、メダル投入時のエラー表示や、ビッグボーナスゲーム中（BBゲーム中）の遊技回数、所定の入賞役の入賞回数等を表示するための表示器である。貯留枚数表示器6127は、スロットマシン6100に電子的に貯留されているメダルの枚数を表示するための表示器である。リールパネルランプ6128は、演出用のランプである。

30

40

#### 【1593】

メダル投入ボタン6130、6131は、スロットマシン6100に電子的に貯留されているメダルを所定の枚数分投入するためのボタンである。本実施例1においては、メダル投入ボタン6130が押下される毎に1枚ずつ最大3枚まで投入され、メダル投入ボタン6131が押下されると3枚投入されるようになっている。メダル投入口6134は、遊技を開始するに当たって遊技者がメダルを投入するための投入口である。すなわち、メダルの投入は、メダル投入ボタン6130又は6131により電子的に投入することもできるし、メダル投入口6134から実際のメダルを投入することもできる。精算ボタン6132は、スロットマシン6100に電子的に貯留されたメダル及びベットされたメダルを精算し、メダル払出口6155よりメダル受皿6156に排出するためのボタンである

50

。メダル返却ボタン6133は、投入されたメダルが詰まった場合に押下してメダルを取り除くためのボタンである。

#### 【1594】

スタートレバー6135は、遊技の開始操作を行うためのレバー型のスイッチである。即ち、メダル投入口6134に所望する枚数のメダルを投入して、スタートレバー6135を操作すると、これを契機としてリール6110乃至6112が回転し、遊技が開始される。ストップボタン6137乃至6139は、スタートレバー6135の操作によって回転を開始したリール6110乃至6112に対する停止操作を行うためのボタンであり、各リール6110乃至6112に対応して設けられている。そして、いずれかのストップボタン6137乃至6139を操作すると対応するいずれかのリール6110乃至6112が停止することになる。

10

#### 【1595】

ドアキー孔6140は、スロットマシン6100の前面扉6102のロックを解除するためのキーを挿入する孔である。メダル払出口6155は、メダルを払出するための払出口である。メダル受皿6156は、メダル払出口6155から払出されたメダルを溜めるための器である。なお、メダル受皿6156は、本実施例1では発光可能な受皿を採用している。

#### 【1596】

上部ランプ6150、サイドランプ6151、中央ランプ6152、腰部ランプ6153、下部ランプ6154は、遊技を盛り上げるための装飾用のランプである。演出装置6190は、例えば開閉自在な扉装置（シャッター）6163が前面に取り付けられた液晶表示装置を含み、この演出装置6190には、例えば小役告知等の各種の情報が表示される。音孔6160は、スロットマシン6100内部に設けられているスピーカの音を外部に出力するための孔である。タイトルパネル6162には、スロットマシン6100を装飾するための図柄が描かれる。

20

#### 【1597】

##### <主制御部300>

次に、図223および図224を用いて、このスロットマシン6100の制御部の回路構成について詳細に説明する。

#### 【1598】

スロットマシン6100の制御部は、大別すると、遊技の中核部分を制御する主制御部6300と、主制御部6300より送信されたコマンドに応じて各種機器を制御する副制御部6400と、副制御部6400より送信されたコマンドに応じて液晶表示装置6157や扉装置6163を制御する扉・液晶画面表示制御部6490によって構成されている。

30

#### 【1599】

##### <主制御部>

まず、図223を用いて、スロットマシン6100の主制御部6300について説明する。なお、同図は主制御部6300の回路ブロック図を示したものである。

#### 【1600】

主制御部6300は、主制御部6300の全体を制御するための演算処理装置であるCPU6310や、CPU6310が各ICや各回路と信号の送受信を行うためのデータバス及びアドレスバスを備え、その他、以下に述べる構成を有する。クロック補正回路6314は、水晶発振器6311から発振されたクロックを分周してCPU6310に供給する回路である。例えば、水晶発振器6311の周波数が12MHzの場合に、分周後のクロックは6MHzとなる。CPU6310は、クロック補正回路6314により分周されたクロックをシステムクロックとして受け入れて動作する。

40

#### 【1601】

また、CPU6310には、センサやスイッチの状態を常時監視するためのタイマ割り込み処理の周期やモータの駆動パルスの送信周期を設定するためのタイマ回路6315が

50

バスを介して接続されている。CPU 6310は、電源が投入されると、データバスを介してROM 6312の所定エリアに格納された分周用のデータをタイマ回路6315に送信する。タイマ回路6315は、受信した分周用のデータを基に割り込み時間を決定し、この割り込み時間ごとに、割り込み要求をCPU 6310に送信する。CPU 6310は、この割り込み要求を契機に、各センサ等の監視や駆動パルスの送信を実行する。例えば、CPU 6310のシステムクロックを6MHz、タイマ回路6315の分周値を1/256、ROM 6312の分周用のデータを44に設定した場合、この割り込みの基準時間は、 $256 \times 44 \div 6\text{MHz} = 1.877\text{ms}$ となる。

#### 【1602】

さらに、CPU 6310には、各ICを制御するためのプログラム、入賞役の内部抽選時に用いる抽選データ、リールの停止位置等の各種データを記憶しているROM 6312や、一時的なデータを保存するためのRAM 6313が接続されている。これらのROM 6312やRAM 6313については他の記憶手段を用いてもよく、この点は後述する副制御部6400においても同様である。また、CPU 6310には、外部の信号を受信するための入力インタフェース6360が接続され、割り込み時間ごとに入力インタフェース6360を介して、スタートレバーセンサ6321、ストップボタンセンサ6322、メダル投入ボタンセンサ6323、精算スイッチセンサ6324、メダル払い出しセンサ6326、電源判定回路6327の状態を検出し、各センサを監視している。

#### 【1603】

メダル投入センサ6320は、メダル投入口6134に投入されたメダルを検出するためのセンサである。スタートレバーセンサ6321はスタートレバー6135の操作を検出するためのセンサである。ストップボタンセンサ6322はストップボタン6137乃至6139のいずれかが押された場合、どのストップボタンが押されたかを検出するためのセンサである。メダル投入ボタンセンサ6323はメダル投入ボタン6130、6131のいずれかが押下された場合、どのメダル投入ボタンが押されたかを検出するためのセンサである。精算スイッチセンサ6324は、精算ボタン6132に設けられており、精算ボタン6132が一回押されると、貯留されているメダル及びベットされているメダルが精算されて払い出されることになる。メダル払い出しセンサ6326は、払い出されるメダルを検出するためのセンサである。電源判定回路6327は、スロットマシン6100に供給される電源の遮断を検出するための回路である。

#### 【1604】

CPU 6310には、更に、入力インタフェース6361、出力インタフェース6370、6371がアドレスデコード回路6350を介してアドレスバスに接続されている。CPU 6310は、これらのインタフェースを介して外部のデバイスと信号の送受信を行っている。入力インタフェース6361には、インデックスセンサ6325が接続されている。インデックスセンサ6325は、各リール6110乃至6112の取付台の所定位置に設置されており、リール6110乃至6112に設けた遮光片がこのインデックスセンサ6325を通過するたびにハイレベルになる。CPU 6310は、この信号を検出すると、リールが1回転したものと判断し、リールの回転位置情報をゼロにリセットする。出力インタフェース6370には、リールを駆動させるためのモータを制御するリールモータ駆動部6330と、ホッパー（バケットにたまっているメダルをメダル払出口6155から払出すための装置。）のモータを駆動するためのホッパーモータ駆動部6331と、遊技ランプ6340（具体的には、入賞ライン表示ランプ6120、スタートランプ6121、再遊技ランプ6122、告知ランプ6123、メダル投入ランプ6124等）と、7セグメント（SEG）表示器6341（払出枚数表示器6125、遊技情報表示器6126、貯留枚数表示器6127等）が接続されている。

#### 【1605】

また、CPU 6310には、乱数発生回路6317がデータバスを介して接続されている。乱数発生回路6317は、水晶発振器6311及び水晶発振器6316から発振されるクロックに基づいて、一定の範囲内で値をインクリメントし、そのカウント値をCPU

10

20

30

40

50

6310に出力することのできるインクリメントカウンタであり、後述する入賞役の内部抽選をはじめ各種抽選処理に使用される。

#### 【1606】

また、図示はしないが、主制御部6300は、クロック補正回路6314から入力するシステムクロックの周波数と乱数発生回路6317に inputsする乱数用クロックの周波数を比較し、乱数用クロックの周波数がシステムクロック以下の周波数である場合に、内部情報レジスタのクロック信号状態ビットを1にセットする周波数発生回路を備えている。主制御部6300のCPU6310は、内部情報レジスタのクロック信号状態ビットが1にセットされている場合（乱数用クロックの周波数がシステムクロックの周波数以下である場合）には、乱数用クロックに何らかの異常（例えば、乱数発生回路6317と水晶発振器6316とを結ぶ配線パターンの断線）が発生したと判定し、内部情報レジスタのクロック信号状態ビットが0にセットされている場合（乱数用クロックの周波数がシステムクロックの周波数未満である場合）には、乱数用クロックが正常であると判定する。また、図示はしないが、主制御部6300は、外部の照合機と接続し、チップの照合（チップの真贋についてのチェック）を行う照合用ブロックブロックを備える。また、乱数発生回路6317から水晶発振器6316までの第1信号線が、クロック補正回路6314から水晶発振器6311までの第2信号線よりも長くなるように配線している。

10

#### 【1607】

CPU6310のデータバスには、副制御部6400にコマンドを送信するための出力インタフェース6371が接続されている。主制御部6300と副制御部6400との情報通信は一方方向の通信であり、主制御部6300は副制御部6400へコマンドを送信するが、副制御部6400から主制御部6300へ何らかのコマンド等を送信することはできない。

20

#### 【1608】

##### <副制御部400>

次に、図224を用いて、スロットマシン6100の副制御部6400について説明する。なお、同図は副制御部6400の回路ブロック図を示したものである。

#### 【1609】

副制御部6400は、主制御部6300より送信された主制御コマンド等に基づいて副制御部6400の全体を制御する演算処理装置であるCPU6410や、CPU6410が各IC、各回路と信号の送受信を行うためのデータバス及びアドレスバスを備え、以下に述べる構成を有する。クロック補正回路6414は、水晶発振器6411から発振されたクロックを補正し、補正後のクロックをシステムクロックとしてCPU6410に供給する回路である。

30

#### 【1610】

また、CPU6410にはタイマ回路6415がバスを介して接続されている。CPU6410は、所定のタイミングでデータバスを介してROM6412の所定エリアに格納された分周用のデータをタイマ回路6415に送信する。タイマ回路6415は、受信した分周用のデータを基に割り込み時間を決定し、この割り込み時間ごとに、割り込み要求をCPU6410に送信する。CPU6410は、この割り込み要求のタイミングをもとに、各ICや各回路を制御する。

40

#### 【1611】

また、CPU6410には、副制御部6400の全体を制御するための命令及びデータ、ライン表示LEDの点灯パターンや各種表示器を制御するためのデータが記憶されたROM6412や、データ等を一時的に保存するためのRAM6413が各バスを介して接続されている。

#### 【1612】

さらに、CPU6410には、外部の信号を送受信するための入出力インタフェース6460が接続されており、入出力インタフェース6460には、図柄表示窓6113の外枠に配設され、点滅や点灯などの点灯制御によって有効ラインや入賞ラインを報知するた

50

めのライン表示LED 6420、前面扉6102の開閉を検出するための扉センサ6421、RAM 6413のデータをクリアにするためのリセットスイッチ6422が接続されている。

#### 【1613】

CPU 6410には、データバスを介して主制御部6300から主制御コマンドを受信するための入力インタフェース6461が接続されており、入力インタフェース6461を介して受信したコマンドに基づいて、遊技全体を盛り上げる演出処理等が実行される。また、CPU 6410のデータバスとアドレスバスには、音源IC 6480が接続されている。音源IC 6480は、CPU 6410からの命令に応じて音声の制御を行う。また、音源IC 6480には、音声データが記憶されたROM 6481が接続されており、音源IC 6480は、ROM 6481から取得した音声データをアンプ6482で増幅させてスピーカ6483から出力する。CPU 6410には、主制御部6300と同様に、外部ICを選択するためのアドレスデコード回路6450が接続されており、アドレスデコード回路6450には、主制御部6300からのコマンドを受信するための入力インタフェース6461、時計IC 6423、7セグメント表示器6440への信号を出力するための出力インタフェース6472等が接続されている。

#### 【1614】

時計IC 6423が接続されていることで、CPU 6410は、現在時刻を取得することが可能である。7セグメント表示器6440は、スロットマシン6100の内部に設けられており、たとえば副制御部6400に設定された所定の情報を遊技店の係員等が確認できるようになっている。更に、出力インタフェース6470には、デマルチプレクサ6419が接続されている。デマルチプレクサ6419は、出力インタフェース6470から送信された信号を各表示部等に分配する。即ち、デマルチプレクサ6419は、CPU 6410から受信されたデータに応じて上部ランプ6150、サイドランプ6151、中央ランプ6152、腰部ランプ6153、下部ランプ6154、リールパネルランプ6128、タイトルパネルランプ6170、払出口ストロボ6171を制御する。タイトルパネルランプ6170は、タイトルパネル6162を照明するランプであり、払出口ストロボ6171は、メダル払い出し口6155の内側に設置されたストロボタイプのランプである。なお、CPU 6410は、扉・液晶画面制御部6490への信号送信は、デマルチプレクサ6419を介して実施する。扉・液晶画面制御部6490は、液晶表示装置6157及び扉装置6163を制御する制御部である。

#### 【1615】

##### <メイン処理>

次に、図225を用いて、主制御部6300のメイン処理について説明する。なお、同図は、主制御部6300のメイン処理の流れを示すフローチャートである。

#### 【1616】

遊技の基本的制御は、主制御部6300のMain CPU 6310が中心になって行い、電源断等を検知しないかぎり、Main CPU 6310が同図のメイン処理を繰り返し実行する。

#### 【1617】

ステップS7101では、初期設定1を行う。この初期設定1では、CPU 6310のスタックポインタ(SP)へのスタック初期値の設定、割り込みマスクの設定、RAM 6313に記憶する各種変数の初期設定、ウォッチドックタイマ(WDT)への動作許可及び初期値の設定等を行う。ステップS7102では、WDTのカウンタの値をクリアし、WDTによる時間計測を再始動する。

#### 【1618】

ステップS7103では、低電圧信号がオンであるか否か、すなわち、電圧判定回路6327が、主制御部6300に供給している電源の電圧値が所定の値(本実施例では9V)未満である場合に電圧が低下したことを示す低電圧信号を出力しているか否かを監視する。そして、低電圧信号がオンの場合(CPU 6310が電源の遮断を検知した場合)に



はステップ S 7 1 0 2 に戻り、低電圧信号がオフの場合（C P U 6 3 1 0 が電源の遮断を検知していない場合）にはステップ S 7 1 0 4 に進む。

【1 6 1 9】

ステップ S 7 1 0 4 では、初期設定 2 を行う。この初期設定 2 では、後述するタイマ割り込み処理を定期毎に実行するための周期を決める数値をタイマ回路 6 3 1 5 に設定する処理、R A M 6 3 1 3 の書き込みを許可する設定等を行う。

【1 6 2 0】

ステップ 1 1 0 5 では、設定キーセンサの状態に基づいて設定キーを受け付けているか否かを判断する。そして、設定キーを受け付けている場合にはステップ S 7 1 0 6 の設定変更処理に進み、設定キーを受け付けていない場合にはステップ S 7 1 0 7 に進む。

10

【1 6 2 1】

ステップ S 7 1 0 6 では、設定変更処理を行う。この設定変更処理では、最初に、機能限定ウェイトの設定を解除すると共に、割り込み許可の設定を行った後、設定値の変更を行う。

【1 6 2 2】

ステップ S 7 1 0 7 では、電断前（電源の遮断前）の状態に復帰するか否かを判断する。そして、電断前の状態に復帰する場合にはステップ S 7 1 0 8 に進み、そうでない場合にはステップ S 7 1 1 0 に進む。

【1 6 2 3】

ステップ S 7 1 0 8 では、（後述する）タイマ割り込み処理において割り込み機能を所定時間ウェイトさせるべく、機能限定ウェイトの設定を行った後（R A M 6 3 1 3 の所定記憶領域に機能限定中であることを示す情報を記憶した後）、ステップ S 7 1 0 9 では、スタックポインタに、電断時に記憶したスタックポインタを再設定するなど、復電時の処理を行って処理を終了する。

20

【1 6 2 4】

ステップ S 7 1 1 0 では、遊技メダル投入処理を行う。詳細は後述するが、この遊技メダル投入処理では、機能限定ウェイトの設定を解除すると共に、割り込み許可の設定を行った後、メダル投入に関する処理を行う。

【1 6 2 5】

ステップ S 7 1 1 1 では、遊技のスタート操作に関する処理を行う。ここでは、スタートレバーセンサ 6 3 2 1 の状態に基づいてスタートレバー 6 1 3 5 が操作されたか否かのチェックを行い、スタート操作されたと判断した場合は、投入されたメダル枚数を確定する。ステップ S 7 1 1 2 では、乱数取得処理を行う（詳細は後述する）。ステップ S 7 1 1 3 では、ステップ S 7 1 1 2 で取得した乱数値と、R O M 6 3 1 2 に格納した抽選データを用いて、入賞役の内部抽選を行う。

30

【1 6 2 6】

ステップ S 7 1 1 4 では、リール回転開始処理により、全リール 6 1 1 0 ～6 1 1 2 の回転を開始する。この際、ステップ S 7 1 1 3 の内部抽選の結果等に基づき、リール 6 1 1 0 ～6 1 1 2 の停止位置を規定したリール停止制御テーブルを選択する。

【1 6 2 7】

ステップ S 7 1 1 5 では、ストップボタンセンサ 6 3 2 2 の状態に基づいてストップボタン 6 1 3 7 ～6 1 3 9 が操作されたか否かのチェックを行い、ストップボタン 6 1 3 7 ～6 1 3 9 が操作されたと判断した場合は、ステップ S 7 1 1 4 で選択したリール停止制御テーブルに基づいて、押されたストップボタン 6 1 3 7 ～6 1 3 9 に対応するリール 6 1 1 0 ～6 1 1 2 の回転を停止する。そして、インデックスセンサ 6 3 2 5 の状態に基づいて全リール 6 1 1 0 ～6 1 1 2 が停止したか否かを判定し、全リール 6 1 1 0 ～6 1 1 2 が停止した場合にはステップ S 7 1 1 6 に進み、そうでない場合にはストップボタン 6 1 3 7 ～6 1 3 9 が操作され全リール 6 1 1 0 ～6 1 1 2 が停止するのを待つ。

40

【1 6 2 8】

ステップ S 7 1 1 6 では、ストップボタン 6 1 3 7 ～6 1 3 9 が押されることによって

50

停止した全リール 6 1 1 0 ～ 6 1 1 2 の図柄の入賞判定を行う。ここでは、有効化された入賞ライン 6 1 1 4 上に、内部当選した役またはフラグ持越し中の役に対応する図柄組合せが揃った（表示された）場合にその役に入賞したと判定する。ステップ S 7 1 1 7 では、メダル払出処理を行う。このメダル払出処理では、払い出しのある何らかの役に入賞していれば、その役に対応する枚数のメダルを払い出す。ステップ S 7 1 1 8 では、遊技を終了するための制御や、遊技状態を移行するための制御を行う。例えば、B B 入賞や R B 入賞の場合に次回から B B 遊技または R B 遊技が開始できるように準備し、それらの最終遊技では、次回から通常遊技が開始できるように準備する。

#### 【1 6 2 9】

以上により 1 遊技が終了し、以降、このメイン処理を繰り返すことにより遊技が進行することになる。

10

#### 【1 6 3 0】

##### <遊技メダル投入処理>

次に、図 2 2 6 を用いて、上記メイン処理における遊技メダル投入処理について説明する。なお、同図は遊技メダル投入処理の流れを示すフローチャートである。

#### 【1 6 3 1】

ステップ S 7 3 0 1 では、内部情報レジスタのクロック信号状態ビットを参照し、クロック信号状態ビットが 1 にセットされているか否かを判定する。そして、クロック信号状態ビットが 1 にセットされている場合（乱数用クロック R C K に異常がある場合）はステップ S 7 3 0 2 に進み、そうでない場合はステップ S 7 3 0 3 に進む。

20

#### 【1 6 3 2】

ステップ S 7 3 0 2 では、乱数用クロック R C K に異常がある旨の情報を他の制御部（例えば、副制御部 6 4 0 0）に送信するために、乱数用クロックの異常による停止コマンドの出力設定をした後、処理を終了する。なお、この乱数用クロックの異常による停止コマンドを受信した副制御部 6 4 0 0 は、乱数用クロックの異常を、液晶表示装置 6 1 5 7 による表示、スピーカ 6 4 8 4 による音声、扉装置（可動物） 6 1 6 3 による動きなどによって報知する。

#### 【1 6 3 3】

ステップ S 7 3 0 3 では、遊技メダルの投入の有無を確認し、遊技メダルが投入された場合にはステップ S 7 3 0 4 に進んで賭け枚数増加処理を行い、遊技メダルが投入されていない場合にはステップ S 7 3 0 5 に進む。

30

#### 【1 6 3 4】

ステップ S 7 3 0 5 では、精算ボタンの操作の有無を確認し、精算ボタンの操作があった場合にはステップ S 7 3 0 6 に進んで遊技媒体の精算処理を行い、精算ボタンの操作が無かった場合にはステップ S 7 3 0 7 に進む。

#### 【1 6 3 5】

ステップ S 7 3 0 7 では、遊技メダルの賭け枚数が 0 より大きいかな否かを判定し、該当する場合にはステップ S 7 3 0 8 に進み、該当しない場合にはステップ S 7 3 0 1 に戻る。

40

#### 【1 6 3 6】

ステップ S 7 3 0 8 では、スタート操作の有無を確認し、スタート操作があった場合には処理を終了し、スタート操作が無い場合にはステップ S 7 3 0 1 に戻る。

#### 【1 6 3 7】

##### <乱数取得処理>

次に、図 2 2 7 を用いて、上記メイン処理における乱数取得処理について説明する。なお、同図は乱数取得処理の流れを示すフローチャートである。

#### 【1 6 3 8】

ステップ S 7 4 0 1 では、内部情報レジスタのクロック信号状態ビットを参照し、クロック信号状態ビットが 1 にセットされているか否かを判定する。そして、クロック信号状態ビットが 1 にセットされている場合（乱数用クロック R C K に異常がある場合）はステ

50

ップ S 7 4 0 2 に進み、そうでない場合はステップ S 7 4 0 3 に進む。

【1639】

ステップ S 7 4 0 2 では、乱数用クロック R C K に異常がある旨の情報を他の制御部（例えば、副制御部 6 4 0 0）に送信するために、乱数用クロックの異常による停止コマンドの出力設定をした後、処理を終了する。なお、この乱数用クロックの異常による停止コマンドを受信した副制御部 6 4 0 0 は、乱数用クロックの異常を、液晶表示装置 6 1 5 7 による表示、スピーカ 6 4 8 4 による音声、扉装置（可動物） 6 1 6 3 による動きなどによって報知する。次のステップ S 7 4 0 3 では、乱数発生回路 6 3 1 7 から乱数を取得した後に、処理を終了する。

【1640】

10

＜タイマ割込み処理＞

次に、図 2 2 8 を用いて、主制御部 6 3 0 0 の M a i n C P U 6 3 1 0 が実行するタイマ割込み処理について説明する。なお、同図はタイマ割込み処理の流れを示すフローチャートである。

【1641】

M a i n C P U 6 3 1 0 は、所定の周期（本実施例では約 1 . 5 0 4 m s に 1 回）でタイマ割込みを発生させるハードウェアタイマを備えており、タイマ割込み処理を、タイマ割込みを契機として所定の周期で実行する。

【1642】

ステップ S 7 2 0 1 では、M a i n C P U 6 3 1 0 の各レジスタをスタックに退避する。ステップ S 7 2 0 2 では、入力ポートの値を取得して、各種センサ（メダル投入センサ 6 3 2 0、スタートレバーセンサ 6 3 2 1、ストップボタンセンサ 6 3 2 2、メダル払出センサ 6 3 2 6 など）の状態を検出する。より具体的には、メダル投入センサ 6 3 2 0、スタートレバーセンサ 6 3 2 1、ストップボタンセンサ 6 3 2 2、メダル払出センサ 6 3 2 6 などの複数のセンサが出力する信号を入力し、これら複数のセンサからの出力がそれぞれオンかオフかを判定する。

20

【1643】

ステップ S 7 2 0 3 では、割込み機能が限定中であるか否か、すなわち、上記メイン処理において機能限定ウェイトを設定しているか否か（R A M 6 3 1 3 の所定記憶領域に機能限定中であることを示す情報を記憶しているか否か）を判断する。そして、割込み機能が限定中である場合（機能限定ウェイトを設定している場合）にはステップ S 7 2 0 4 ～ S 7 2 0 8 の処理を行うことなくステップ S 7 2 0 9 に進み、割込み機能が限定中でない場合（機能限定ウェイトを設定していない場合）にはステップ S 7 2 0 4 に進む。

30

【1644】

ステップ S 7 2 0 4 では、遊技メダル投入受付処理を行う。詳細は後述するが、この遊技メダル投入受付処理では、現在の遊技状態や、一遊技に必要な規定メダル数などに基づいて、遊技メダルの投入に関する処理を行う。

【1645】

ステップ S 7 2 0 5 では、リール 6 1 1 0 ～ 6 1 1 2 の回転制御や停止制御などのリール制御処理を行い、ステップ S 7 2 0 6 では、その他の処理を行う。ステップ S 7 2 0 7 では、遊技ランプ 6 3 4 0（上部ランプ 6 1 5 0、サイドランプ 6 1 5 1、中央ランプ 6 1 5 2、腰部ランプ 6 1 5 3、下部ランプ 6 1 5 4、受皿ランプ 6 1 5 6 など）に出力するデータの更新を行う。

40

【1646】

ステップ S 7 2 0 8 では、常時エラー監視処理を行う。この常時エラー監視処理では、上記ステップ 1 2 0 2 で検出したメダル払出センサ 6 3 2 6 などの状態に基づいて、ホッパーなどのエラーの監視と、エラーの検出情報の更新を行う。

【1647】

ステップ S 7 2 0 9 では、集中端子板信号設定処理を行う。この集中端子板信号設定処理では、R A M 6 3 1 3 に記憶したセキュリティ状態情報などに基づいて状態複合信号を

50

生成し、シリアル出力する。また、RAM 6313に記憶したメダル投入情報などに基づいてメダル投入信号、メダル払出信号、外部信号1～4を生成し、パラレル出力する。

【1648】

ステップS7210では、7SEG表示器6341に出力するデータの設定を行い、ステップS7211では、リールモータ制御部6330やホッパーモータ制御部6331などに出力するデータの設定を行う。また、内部情報レジスタのクロック信号状態ビットを参照し、内部情報レジスタのクロック信号状態ビットが1にセットされている場合（乱数用クロックの周波数がシステムクロックの周波数以下である場合）には、乱数用クロックに何らかの異常（例えば、乱数発生回路6317と水晶発振器6316とを結ぶ配線パターンの断線）が発生したと判定し、7SEG表示器6341を用いてエラーを示す情報を表示する。一方、内部情報レジスタのクロック信号状態ビットが0にセットされている場合（乱数用クロックの周波数がシステムクロックの周波数未満である場合）には、乱数用クロックが正常であると判定し、次のステップS7212に進む。

10

【1649】

ステップS7212では、副制御部6400に制御コマンドを出力し、ステップS7213では、RAM 6313に記憶している各種のカウンタタイマ（例えば、上述のセキュリティ信号出力待機タイマなど）を更新する。

【1650】

ステップS7214では、上述の電圧低下信号がオンであるか否か、すなわち、スロットマシン6100に所定の電圧が供給されているか否かを判断する。そして、電圧低下信号がオンである場合（スロットマシン6100に所定の電圧が供給されていない場合）にはステップS7217に進み、電圧低下信号がオフである場合（スロットマシン6100に所定の電圧が供給されている場合）にはステップS7215に進む。

20

【1651】

ステップS7215では、ステップS7201で退避したレジスタを復帰し、ステップS7216では、割込みを許可する設定を行う。

【1652】

ステップS7217～S7220では、電断処理を行う。より具体的には、ステップS7217では、現在のスタックポインタを保存し、ステップS7218では、電源ステータスを「電断」に更新する。また、ステップS7219では、RAM 6313に記憶した値を用いてチェックサムを計算・設定し、ステップS7220では、RAM 6313の書き込みを禁止に設定した後、無限ループとなる。

30

【1653】

このようなスロットマシン6100のCPU 6310やCPU 6410（のパッケージ）に上述の本発明を適用すれば、視認容易部を通して遊技制御用マイクロコンピュータの裏側を視認することができ、マイクロコンピュータの裏側などに不正部品を取り付けるような不正行為を未然に防止することができる場合がある。

【1654】

なお、本発明に係る遊技台は、上記実施形態に係るパチンコ機やスロットマシンに限定されず、例えば、封入式遊技機に適用することもできる。また、主制御部、第1副制御部、および第2副制御部をワンチップで構成してもよいし、主制御部と第1副制御部で双方向の通信が可能に構成してもよい。また、主制御部と第1副制御部で双方向の通信を可能とする一方で、第1副制御部から第2副制御部への通信は一方向の通信としてもよい。

40

【1655】

また、本発明の実施の形態に記載された作用および効果は、本発明から生じる最も好適な作用および効果を列挙したに過ぎず、本発明による作用および効果は、本発明の実施の形態に記載されたものに限定されるものではない。また、実施例に記載した複数の構成のうち、1つの構成に記載している内容を、他の構成に適用することでより遊技の幅を広げられる場合がある。

【1656】

50

### ＜実施形態 5＞

以下、図面を用いて、本発明の実施形態 5 に係るパチンコ機（遊技台）について詳細に説明する。

#### 【1657】

図 229 は、図 2 を用いて説明した電源基板 182、払出基板 170、および主基板 156 間の電源供給例を示している。図 229 (a) に示す例では、各種電氣的遊技機器に電源を供給する電源制御部 660 を備える電源基板 182 から電源ライン p101 と電源ライン p102 とが引き出され、電源ライン p101 と電源ライン p102 が共に、遊技球の払出に関する制御処理を行う払出制御部 600 を備える払出基板（払出制御基板）170 に接続されている。電源投入後に一定時間経過したら電源ライン p101 には DC（直流）12V の電圧が印加され、電源ライン p102 には DC 12V と異なる直流電圧が印加される。払出基板 170 からは電源ライン p103 が引き出され、電源ライン p103 は、遊技全般の制御処理を行う主制御部 300 を構成する主基板（主制御基板）156 に接続されている。電源ライン p103 は払出基板 170 内で電源ライン p101 に接続されている。従って、電源投入後に一定時間経過したら主基板 156 には払出基板 170 を介して電源基板 182 から DC 12V の電圧が供給される。なお、これらの電圧値は一例であって、駆動する回路によってこれらと異なる電圧値の電圧を印加するようにしてももちろんよい。

#### 【1658】

図 229 (b) に示す例では、電源基板 182 から電源ライン p104 と電源ライン p105、および電源ライン p106 が引き出されている。電源ライン p104 は主基板 156 に接続されている。電源ライン p105 と電源ライン p106 は共に、払出基板 170 に接続されている。電源投入後に一定時間経過したら電源ライン p104 と電源ライン p105 には DC 12V の電圧が印加され、電源ライン p106 には DC 12V と異なる直流電圧が印加される。つまり、電源投入後に一定時間経過したら主基板 156 には払出基板 170 を介さずに、電源基板 182 から直接に DC 12V の電圧が供給される。

#### 【1659】

図 229 (c) に示す例では、電源基板 182 から電源ライン p107、電源ライン p108、電源ライン p109、および電源ライン p110 が引き出されている。電源ライン p107 と電源ライン p108 は主基板 156 に接続されている。電源ライン p109 と電源ライン p110 は共に、払出基板 170 に接続されている。電源投入後に一定時間経過したら電源ライン p107 と電源ライン p109 には DC 12V の電圧が印加され、電源ライン p108 には DC 5V の電圧が印加され、電源ライン p110 には DC 5V / 12V とは異なる直流電圧が印加される。つまり、電源投入後に一定時間経過したら主基板 156 には払出基板 170 を介さずに、電源基板 182 から直接に DC 5V と DC 12V の 2 種類の電圧が供給される。

#### 【1660】

図 230 は、主基板 156 内の電源供給例を示している。本例では図 229 (a) に示す構成を用いて説明する。払出基板 170 から引き出された電源ライン p103 は主基板 156 内に引き込まれて、DC-DC コンバータ 7001 の入力端子に接続されている。DC-DC コンバータ 7001 は電源ライン p103 に印加された DC 12V の電圧を DC 5V に変換する機能を有している。DC-DC コンバータ 7001 で生成された DC 5V の電圧は電源ライン p111 に印加される。

#### 【1661】

電源ライン p111 には複数の分岐ラインが接続されており、所定の分岐ラインがリセット IC 7003 の電源端子 A に接続されている。電源投入後に一定時間経過したら電源端子 A に DC 5V の電圧が駆動電圧として印加され、これによりリセット IC 7003 が駆動される。また、電源ライン p111 から分岐した別の分岐ラインがマイコン（CPU）304 の電源端子 F に接続されている。電源投入後に一定時間経過したら電源端子 F に DC 5V の電圧がマイコン 304 の駆動電圧として印加される。なお、リセット IC 70

10

20

30

40

50

03とマイコン304の駆動電圧は同じでなく異なってもよい。また、リセットIC7003やマイコン304の駆動電圧の電圧値自体と電源ラインp111に印加される電圧の電圧値が同じでなく異なっているようにいてもよい。

#### 【1662】

リセットIC7003の信号入力端子Bは別の分岐ラインを介して電源ラインp111に接続されている。リセットIC7003は、信号入力端子Bの端子電圧をモニタして、マイコン304の端子Fに印加されるマイコン304の駆動電圧の状態を把握するようになっている。一方、リセットIC7003の信号入力端子Cは所定の分岐ラインを介して電源ラインp103に接続されている。リセットIC7003は、信号入力端子Cの端子電圧をモニタして、電源ラインp103に印加されている電圧の状態を把握するようになっている。

10

#### 【1663】

リセットIC7003には少なくとも2つの信号出力端子D、Eが設けられている。信号出力端子Dからはリセット信号（起動信号）が出力され、信号出力端子Eからは低電圧信号が出力される。リセットIC7003は、信号入力端子Bの端子電圧がマイコン304の動作保証電圧（例えば、4.5V）を超えていないと判断したら、信号出力端子Dに出力しているリセット信号をネゲート（無効）状態にする。一方、信号入力端子Bの端子電圧がマイコン304の動作保証電圧を超えたとは判断したら、信号出力端子Dに出力しているリセット信号をアサート（有効）状態にする。つまり、信号入力端子Bの端子電圧がマイコン304の動作保証電圧（例えば、4.5V）を超えていない場合に、信号出力端子Dに出力されているリセット信号がネゲート状態になり、信号入力端子Bの端子電圧がマイコン304の動作保証電圧を超えた場合には、信号出力端子Dに出力されているリセット信号がアサート状態になる。なお、リセットIC7003の端子の数は任意であり図示のものは一例示であり、説明に必要な端子を図示しているだけであって、端子の数、配置位置は図示のものと同じでもよいし同じでなくてもよい。

20

#### 【1664】

リセットIC7003の信号出力端子Dは、信号ラインp112を介してマイコン304の信号入力端子Gに接続されている。信号入力端子Gはシステムリセット（XSRST）端子として用いられている。このため、信号ラインp112に出力されているリセット信号がアサート状態になると、マイコン304はセキュリティモードに移行する。このように、リセットIC7003は、信号出力端子Dがリセット信号出力端子として機能して図4に示す起動信号出力回路340としての役割を果たしている。

30

#### 【1665】

リセットIC7003は、信号入力端子Cの端子電圧が所定の電圧値（例えば、9V）を超えていないと判断したら、信号出力端子Eに出力している低電圧信号をアサート状態にする。一方、信号入力端子Cの端子電圧が所定の電圧値を超えたとは判断したら、信号出力端子Eに出力している低電圧信号をネゲート状態にする。

#### 【1666】

リセットIC7003の信号出力端子Eは、信号ラインp113を介してマイコン304の信号入力端子Hに接続されている。信号入力端子Hはデータ端子D0～D7のいずれか1つのデータ端子として用いられている。例えば、マイコン304は、信号ラインp113に出力されている低電圧信号がネゲート状態になるまで、例えば図10のステップS103とステップS105の処理を繰り返す。このように、リセットIC7003は、信号出力端子Eが定電圧信号出力端子として機能して図4に示す電圧監視回路338としての役割を果たしている。

40

#### 【1667】

この例では信号ラインp113上の低電圧信号がアサート状態からネゲート状態（あるいはその逆）になるときの信号入力端子Cの所定の閾値電圧値を9Vとしているがこれに限られない。例えば図14に示す主制御部タイマ割込処理のステップS239およびステップS243における電断検出から電断処理に要する期間中にマイコン304が電断処理

50

を継続できる電圧が電源端子Fに印加されていればよい。DC-DCコンバータ7001から電源ラインp111に出力されるDC5V電圧は、DC-DCコンバータ7001にのDC12V電圧が7V程度に低下する辺りから徐々に低下を始める。そこで、信号入力端子Cの端子電圧が所定の電圧値（例えば、9V）未満になると、信号出力端子Eに出力している低電圧信号をネゲート状態からアサート状態にするようにしている。所定の電圧値として例えば9Vを用いれば、電源ラインp111の電圧が徐々に低下しても主制御部タイマ割込処理のステップS239およびステップS243における電断検出から電断処理に要する期間中にマイコン304が動作を継続できる。

#### 【1668】

図231は、電源基板182と主基板156内の電源供給例を示している。図230に示す電源供給構成と比較して説明すると、図230で主基板156内に配置されていたDC-DCコンバータ7001とリセットIC7003が、図231に示す構成では電源基板182上に配置されている。これにより、電源ラインp111と信号ラインp112およびp113は電源基板182および主基板156の間を跨いで配線されている。このような構成によれば、DC5V、DC12V、およびその他の電圧値の電源を電源基板182で全て供給可能になるので、主基板156や払出基板170に電源供給用回路を設ける必要がなくなり、基板上の回路レイアウトに余裕を持たせることができる場合がある。

#### 【1669】

図232は、主基板156内の電源供給例を示している。図232(a)は、CPU304の外部マスカブル割込み端子XINTと外部ノンマスカブル割込み端子XNMIに電源ラインp111の分岐ラインを接続して、両端子XINTおよびXNMIにDC5Vの電圧を常時印加している状態を示している。こうすることにより、両端子XINTおよびXNMIに入力する割込み信号は常にネゲート状態を維持するため、割込み端子による外部割込みを使用しない場合には、ノイズ等による誤動作を確実に防止することができる。図232(b)に示す構成は、図230に示す構成におけるCPU304の端子Hが外部ノンマスカブル割込み端子XNMIとして機能している場合を例示している。なお、割込み端子の符号のXは、ローアクティブを示しておりL信号が入力されてアサート状態となるが、ハイアクティブでアサート状態になる回路の場合はL信号（例えば、0V）を常に入力させるようにする。

#### 【1670】

図233は、パチンコ機100の電源投入後の起動処理のタイムチャートである。図中左から右に向かって時間の経過を表している。図下段から中段に示す動作モード、XRSSTO端子、およびXSRSST端子は図9に示すものと同一である。図中段から上段に向かっては低電圧信号、マイコン電圧、および監視電圧を示している。低電圧信号は、例えば図230に示すリセットIC7003の端子Eから信号ラインp113を介してCPU304の端子H（所定のD端子）に出力される信号である。マイコン電圧は、例えば図230に示すCPU304の電源端子Fに印加される電源ラインp111上の電圧レベルを示している。監視電圧は、リセットIC7003の信号入力端子Cに入力する電源ラインp103に印加された電圧である。

#### 【1671】

図233に示すように、電源投入後、システムリセット信号がアサート状態になりXSRSST端子の信号レベルがHになった時刻T-XSRSSTで、監視電圧は例えば約7V、マイコン電圧は例えば約4.5Vになっている。システムリセット信号がアサート状態になるとセキュリティモードに移行し、セキュリティチェック処理に続きランダム延長処理が実行される。本例ではセキュリティチェックが終了した後でランダム延長処理が開始する前にXRSSTO端子のリセット信号がアサート状態になる。ランダム延長処理の終了後にユーザモードに移行し、遊技制御用プログラムが実行されて主制御部メイン処理が開始される。

#### 【1672】

電源投入後に監視電圧は所定の電圧値まで徐々に上昇する。それに伴いマイコン電圧も

10

20

30

40

50

上昇し、システムリセット信号出力直後にほぼ5Vの電源電圧が得られる。監視電圧はその後にも上昇し破線で示す閾値電圧レベル（下限電圧であり、例えば9V）Sを超えた時刻Tー低電圧で低電圧信号がアサート状態からネゲート状態となる。さらに所定時間経過すると監視電圧は所定の電圧値（CPU304の動作電圧値以上、電源ラインの上限電圧未満の所定の電圧）としてほぼ12Vとなる。本例ではWDT314を自動起動させておりユーザモード開始直後からWDT314が自動起動される。その後、主制御部メイン処理のステップS103でWDT314のクリア&リスタート処理が繰り返し実行される。図233ではWDT314の起動直後の最初のWDTクリア&リスタートを例示している。

#### 【1673】

図234は、図233の破線の楕円Oで囲んだ範囲を拡大した監視電圧（電源ラインp103に印加されている電圧）を示している。図234（a）、（b）は図中左から右に向かって時間の経過を表している。縦方向は監視電圧の電圧値が7V～12Vまでを1V刻みで示している。図234（a）は監視電圧の理想的な上昇を示している。図234（b）はWDT314のクリア&リスタートにより生じる問題を説明する図である。図234（a）に示すように理想的には時刻T1で監視電圧が閾値電圧（9V）を超え、その後監視電圧がさらに上昇する途中で主制御部300の最初のタイマ割込処理が開始される。監視電圧は時刻Tですでに閾値電圧を超えているためリセットICの端子Eから出力される低電圧信号は最初のタイマ割込処理が開始時点ですでにネゲート状態になっている。このため最初の主制御部タイマ割込処理での電断検出の判断処理（ステップS239）では電断状態は検出されない。

#### 【1674】

一方、図234（b）はWDT314が自動起動した後に、主制御部メイン処理のステップS103で実行されるWDT314のクリア&リスタートが繰り返し実行されている状態を示している。図234（b）に示すようにWDT314のクリア&リスタート処理が発生する度に電源ラインp103に印加された電圧に電圧降下（ノイズ）現象が生じている。このWDT314のクリア&リスタートの繰り返し処理が電源ラインp103に印加される電圧が上昇を始める初期の段階から行われると、電圧降下の積み重ねが多くなるため、監視電圧が最初に閾値電圧を超える時刻T2が図234（a）に示す時刻T1より遅くなってしまう。このように、WDT314のクリア&リスタートの繰り返し処理が電源ラインp103の印加電圧を所定電圧にまで上昇させるのを遅らせてしまい、主制御部メイン処理の処理時間が無用に長くなってしまうという問題を有している。

#### 【1675】

さらに、時刻T2で監視電圧が閾値を超えてリセットIC7003の信号出力端子Eの低電圧信号はアサート状態からネゲート状態になっても、初回の主制御部タイマ割込処理におけるWDTクリア&リスタートにより監視電圧に電圧降下が生じると、主制御部タイマ割込み処理の電断検出の判断処理（ステップS239）の時点で、再び監視電圧が閾値レベル未満に低下してリセットIC7003の信号出力端子Eの低電圧信号がネゲート状態からアサート状態に戻ってしまう。こうなると、最初の主制御部タイマ割込処理の電断検出の判断処理（ステップS239）で電断状態と判断されてしまうため、電断処理（ステップS243）が実行されてしまう場合が生じ得る。

#### 【1676】

上記問題点を解決するには、少なくとも、WDT314の最初のクリア&リスタート処理が、監視電圧が閾値電圧レベルSを超えて低電圧信号がアサート状態からネゲート状態に変化した時刻Tー低電圧より後に実行されるようにすればよい。WDT314の最初のクリア&リスタート処理が時刻Tー低電圧より確実に後になるようにするには、WDT314の起動（手動起動または自動起動）を時刻Tー低電圧より後に開始すればよい。WDT314の起動は最先でユーザモードの開始時点になるので、ユーザモード開始時点より以前に時刻Tー低電圧が到来すればよい。

#### 【1677】

図235は、ユーザモード開始時点より以前に時刻Tー低電圧を到来させて、WDT3

10

20

30

40

50



1 4 の最初のクリア&リスタート処理が時刻 T－低電圧より後に実行されるようにした W D T の駆動方法（その 1）を示している。図 2 3 5 は図 2 3 3 と同様のパチンコ機 1 0 0 の電源投入後の起動処理のタイムチャートである。図中左から右に向かって時間の経過を表している。図 2 3 5 の構成は図 2 3 3 と同一であるので同一構成についての説明は省略する。

#### 【1 6 7 8】

ユーザモード開始時点より以前に時刻 T－低電圧を到来させるにはセキュリティモード期間中に時刻 T－低電圧を到来させるようにする。セキュリティモード期間中に時刻 T－低電圧を到来させるには、予めセキュリティモード期間に要する時間を把握しておく。また、実験やシミュレーションにより得た監視電圧の電圧上昇プロファイルに基づき電源投入から時刻 T－低電圧までに要する時間を把握しておく。これに基づき、電源投入から時刻 T－X S R S T までの時間とセキュリティモード期間の時間の合計が、電源投入から時刻 T－低電圧までの時間より長くなるように時刻 T－X S R S T を決める。なお本例では、セキュリティモード期間内にランダム延長処理が含まれている。従って、セキュリティモード期間中に確実に時刻 T－低電圧を到来させるには、電源投入から時刻 T－X S R S T までの時間とセキュリティチェック処理に要する時間の合計が、電源投入から時刻 T－低電圧までの時間より長くなるように時刻 T－X S R S T を決める。

#### 【1 6 7 9】

また、マイコン電圧の上昇途中で時刻 T－X S R S T を設定する場合には、リセット I C 7 0 0 3 の信号入力端子 B の端子電圧のレベルを判断する閾値電圧をマイコン 3 0 4 の動作保証電圧（例えば、4. 5 V）より高い電圧（例えば 4. 8 V）に設定する。また、図 2 3 5 に示すように、マイコン電圧が D C 5 V で安定した状態で時刻 T－X S R S T が到来するようにする場合には、リセット I C 7 0 0 3 の信号入力端子 B の端子電圧のレベルを判断する閾値電圧を例えば 4. 8 ～ 5. 0 V のいずれかに設定すると共に、閾値電圧に到達してから時刻 T－X S R S T になるまでの期間だけシステムリセット信号をアサート状態にする時期を遅延させる遅延処理を施すようにする。

なお、電源ライン p 1 1 1 からの分岐ラインの途中で分圧して目標値 D C 5 V よりかなり低い電圧をリセット I C 7 0 0 3 の信号入力端子 B に印加するようにしてもよい。例えば、閾値電圧を約 1. 2 2 V に設定して、信号入力端子 B の端子電圧が約 1. 2 2 V になるとシステムリセット信号が出力されるように構成してもよい。つまり、マイコン 3 0 4 には動作保証電圧以上の電圧が印加されている場合には、信号入力端子 B に約 1. 2 2 V の電圧が印加される。

#### 【1 6 8 0】

このように、リセット I C 7 0 0 3 から出力されるリセット信号がアサート状態になる時期を従来に比して遅延させることにより、図 2 3 4 を用いて説明した問題を生じることなく安定してパチンコ機 1 0 0 の起動処理を行うことができる。

#### 【1 6 8 1】

図 2 3 6 は、ユーザモード開始時点より以前に時刻 T－低電圧を到来させて、W D T 3 1 4 の最初のクリア&リスタート処理が時刻 T－低電圧より後に実行されるようにした W D T の駆動方法（その 2）を示している。図 2 3 6 は図 2 3 3 と同様のパチンコ機 1 0 0 の電源投入後の起動処理のタイムチャートである。図中左から右に向かって時間の経過を表している。図 2 3 6 の構成は図 2 3 3 と同一であるので同一構成についての説明は省略する。

#### 【1 6 8 2】

ユーザモード開始時点より以前に時刻 T－低電圧を到来させるにはセキュリティモード期間中に時刻 T－低電圧を到来させるために、予めセキュリティモード期間に要する時間を把握しておく。また、実験やシミュレーションにより得た監視電圧の電圧上昇プロファイルに基づき電源投入から時刻 T－低電圧までに要する時間を把握しておく。これに基づき、電源投入から時刻 T－X S R S T までの時間とセキュリティモード期間の時間の合計が、電源投入から時刻 T－低電圧までの時間より長くなるようにセキュリティモード期間

の時間を決める。

#### 【1683】

また、実験やシミュレーションによりマイコン電圧の電圧上昇プロファイルを把握しておき、マイコン電圧の上昇中に時刻T-XSRSTが到来するようにさせる場合には、リセットIC7003の信号入力端子Bの端子電圧のレベルを判断する閾値電圧をマイコン304の動作保証電圧（例えば、4.5V）より高い電圧（例えば4.8V）に設定する。また、図235に示すように、マイコン電圧がDC5Vで安定した状態で時刻T-XSRSTが到来するようにする場合には、リセットIC7003の信号入力端子Bの端子電圧のレベルを判断する閾値電圧を例えば4.8～5.0Vのいずれかに設定すると共に、閾値電圧に到達してから時刻T-XSRSTになるまでの期間だけシステムリセット信号をアサート状態にする時期を遅延させる遅延処理を施すようにする。

10

#### 【1684】

このようにして決めた電源投入から時刻T-XSRSTまでの時間とセキュリティモードに要する時間の合計が、電源投入から時刻T-低電圧までの時間より長くなるようにセキュリティモード内での固定延長処理に要する時間を決める。固定延長処理に要する時間は対応する数値をプログラム管理エリアに予め設定しておく。なお本例では、セキュリティモード期間内にランダム延長処理が含まれている。従って、セキュリティモード期間中に確実に時刻T-低電圧を到来させるには、電源投入から時刻T-XSRSTまでの時間とセキュリティチェック処理に要する時間と、固定延長処理の時間との合計が、電源投入から時刻T-低電圧までの時間より長くなるように、例えば図114（a）に示すテーブルに基づいて固定延長処理の時間を決める。

20

#### 【1685】

このように、固定延長処理に要する時間を調節することにより、図234を用いて説明した問題を生じることなく安定してパチンコ機100の起動処理を行うことができる。固定延長処理に要する時間を調節するだけで、リセットIC7003や周辺回路に設計変更を加える必要が生じないので低コストで容易に安定してパチンコ機100の起動処理を行うことができる。

#### 【1686】

また、WDT314の最初のクリア&リスタート処理を時刻T-低電圧より確実に後にする方法として、ユーザモード開始後に時刻T-低電圧が到来するようにしてもよい。この場合は、遊技制御用プログラムにおいて、時刻T-低電圧の到来を判断し、時刻T-低電圧が到来したと判断したら、WDT314を手動起動する。

30

#### 【1687】

図237は、ユーザモード開始時点より後に時刻T-低電圧を到来させて、WDT314の最初のクリア&リスタート処理が時刻T-低電圧より後に実行されるようにしたWDTの駆動方法（その3）を示している。図237は図233と同様のパチンコ機100の電源投入後の起動処理のタイムチャートである。図中左から右に向かって時間の経過を表している。図237の構成は図233と同一であるので同一構成についての説明は省略する。

#### 【1688】

ユーザモード開始時点より後に時刻T-低電圧を到来させるにはセキュリティモード以降に時刻T-低電圧を到来させるようにする。セキュリティモード以降に時刻T-低電圧を到来させるには、予めセキュリティモード期間に要する時間を把握しておく。また、実験やシミュレーションにより得た監視電圧の電圧上昇プロファイルに基づき電源投入から時刻T-低電圧までに要する時間を把握しておく。これに基づき、電源投入から時刻T-XSRSTまでの時間とセキュリティモード期間の時間の合計が、電源投入から時刻T-低電圧までの時間より短くなるように、時刻T-XSRSTとセキュリティモード期間の時間のいずれか一方または両方を決める。なお本例では、セキュリティモード期間内にランダム延長処理が含まれている。従って、セキュリティモード期間の後に確実に時刻T-低電圧を到来させるには、電源投入から時刻T-XSRSTまでの時間と、セキュリティ

40

50

チェック処理に要する時間と、固定延長処理の時間と、ランダム延長処理の最大時間との合計が、電源投入から時刻 T - 低電圧までの時間より短くなるように、時刻 T - X S R S T までの時間、セキュリティチェック処理に要する時間、固定延長処理の時間、ランダム延長処理の最大時間を決める。

#### 【1689】

また、マイコン電圧の電圧上昇プロファイルを把握しておき、マイコン電圧の上昇中に時刻 T - X S R S T が到来するように、リセット I C 7 0 0 3 の信号入力端子 B の端子電圧のレベルを判断する閾値電圧をマイコン 3 0 4 の動作保証電圧（例えば、4.5V）より高い電圧（例えば4.8V）に設定する。

#### 【1690】

上記観点に基づき、時刻 T - X S R S T までの時間、セキュリティチェック処理に要する時間、固定延長処理の時間、ランダム延長処理の最大時間を調整することにより、主制御部 3 0 0 のメイン処理の開始後で主制御部 3 0 0 のタイマ割込み処理への移行を許可する最初の割込み許可前に、時刻 T - 低電圧を到来させ、且つ W D T 3 1 4 を手動起動することができる。

#### 【1691】

遊技制御用プログラムには、低電圧信号がネゲート状態になっているか否かを判断する処理（1）を追加し、低電圧信号がネゲート状態になったら W D T 3 1 4 を手動起動する処理（2）を付加しておく。こうすることにより、図 2 3 4 を用いて説明した問題を生じることなく安定してパチンコ機 1 0 0 の起動処理を行うことができる。

#### 【1692】

また、パチンコ機 1 0 0 の電源投入後の起動処理のタイムチャートが図 2 3 3 に示すように既知の場合には、遊技制御用プログラムに上記（1）、（2）の処理を追加するだけで、図 2 3 4 を用いて説明した問題を生じることなく安定してパチンコ機 1 0 0 の起動処理を行うことができる。

#### 【1693】

＜主制御部メイン処理と主制御部タイマ割込処理＞

次に、図 2 3 8 および図 2 3 9 を用いて、主制御部 3 0 0 の C P U 3 0 4 が実行する主制御部メイン処理と主制御部タイマ割込処理について説明する。図 2 3 8 は主制御部メイン処理の流れを示すフローチャートであって、図 1 0 に示す主制御部メイン処理とほぼ同様のフローで構成されている。図 2 3 9 は主制御部タイマ割込処理の流れを示すフローチャートであって、図 1 4 に示す主制御部タイマ割込処理とほぼ同様のフローで構成されている。図 2 3 8 および図 2 3 9 において、図 1 0 および図 1 4 に示す処理と同一の処理には同一のステップ番号を付してその説明は省略する。

#### 【1694】

まず、図 2 3 9 に示す制御部タイマ割込処理では、図 1 4 に示す処理と同様に、タイマ割込開始処理（ステップ S 2 0 1）からデバイス監視処理（ステップ S 2 3 7）までが順次実行される。次いでステップ S 2 3 9 では、低電圧信号がオン（アサート状態）であるか否かを監視する。そして、低電圧信号がオンの場合（電源の遮断を検知した場合）にはステップ S 2 4 3 に進み、低電圧信号がオフの場合（電源の遮断を検知していない場合）にはステップ S 2 4 1 に進む。ステップ S 2 4 1 では、タイマ割込終了処理を行う。このタイマ割込終了処理では、ステップ S 2 0 1 で一時的に退避した各レジスタの値を元の各レジスタに設定したり、割込許可の設定などを行い、その後、図 2 3 8 に示す主制御部メイン処理に復帰する。

#### 【1695】

一方、ステップ S 2 4 3 では、復電時に電断時の状態に復帰するための特定の変数やスタックポインタを復帰データとして R A M 3 0 8 の所定の領域に退避し、入出力ポートの初期化等の電断処理を行う。また、W D T 3 1 4 を手動起動させている場合はステップ S 2 4 3 で W D T 3 1 4 を停止させるようにしてもよい。

#### 【1696】

10

20

30

40

50

次いで、ステップS 2 4 4では、低電圧信号がオンであるか否か、すなわち、電圧監視回路3 3 8が、電源制御部6 6 0から主制御部3 0 0に供給している電源の電圧値が所定の値（本実施形態では9 V）未満である場合に電圧が低下したことを示す低電圧信号を出力しているか否かを監視する。そして、低電圧信号がオンの場合（C P U 3 0 4が電源の遮断を検知した場合）には低電圧信号を出力しているか否かを監視する処理を繰り返し、低電圧信号がオフの場合（C P U 3 0 4が電源の遮断を検知していない場合）には図2 3 8に示す主制御部メイン処理の初期設定1（ステップS 1 0 1）に移行し、主制御部メイン処理を続行する。W D T 3 1 4を手動起動させている場合において、ステップS 2 4 3でW D T 3 1 4を停止させている場合には、W D T タイムアウトによるリセット処理への移行をさせずに、ステップS 2 4 4の処理を繰り返すことができる。例えば特図変動遊技中に、ステップS 2 4 3での電断処理中にW D T 3 1 4を一時停止して、低電圧で安定した場合（ステップS 2 4 4を無限ループした場合）になると、副制御部4 0 0側は特図変動遊技の終了を示すコマンドを受信できないので、いつまでも特図変動遊技中の状態や揺れ変動の状態となる。

#### 【1 6 9 7】

また、この構成にすることにより、図1 2 0（b）に示すようなW D T 3 1 4のタイムアウト直後にアドレス信号A 0～A 1 5端子から出力される1 6ビットのアドレス情報（例えば、E 0 0 0 H）を出力しないようにすることができる。これにより不正者による不正使用を防止することができる場合がある。また、この構成によればセキュリティモードを省略してユーザモードから処理を開始できるのでパチンコ機1 0 0の起動を早くすることができる場合がある。なお、ステップS 2 4 4から主制御部メイン処理への移行先はステップS 1 0 1に限る必要はない。例えば、ステップS 1 0 1～S 1 0 5の処理を省略して、ステップS 2 4 4からステップS 1 0 7の初期設定2から実行するようにしてももちろんよい。

#### 【1 6 9 8】

##### <主制御部タイマ割込処理>

次に、図2 4 0を用いて、主制御部3 0 0のC P U 3 0 4が実行する主制御部タイマ割込処理の変形例について説明する。図2 4 0（a）は本変形例に係る主制御部タイマ割込処理の流れを示すフローチャートである。本変形例に係る主制御部タイマ割込処理の流れは、図1 0に示すタイマ割込開始処理（ステップS 2 0 1）からタイマ割込終了処理（ステップS 2 4 1）までの処理において、電断検出の判断処理（ステップS 2 3 9）と電断処理（ステップS 2 4 3）とが省かれている点に特徴を有している。図2 4 0において、図1 0に示す処理と同一の処理には同一のステップ番号を付してその説明は省略する。

#### 【1 6 9 9】

図2 4 0（b）は主制御部3 0 0のC P U 3 0 4でのノンマスカブル割込（N M I）処理の流れを示すフローチャートである。ノンマスカブル割込みは、C P U 3 0 4が割込み禁止状態でも無条件に受け付けられる割込みである。ノンマスカブル割込みはX N M I端子に入力する割込み信号をアサート状態（本例では、ローレベル）にすることにより実行される。本変形例では、ノンマスカブル割込みが発生すると電断処理（ステップS 2 4 3）を実行する。このように主制御部3 0 0のC P U 3 0 4が実行する主制御部タイマ割込処理から電断処理を分離し、ノンマスカブル割込みの発生を条件に電断処理を行うようにしている。こうすることにより、安定した遊技制御を行うことができる。なお、N M I処理の電断処理（ステップS 2 4 3）の後に図2 3 9の低電圧状態判定処理（ステップS 2 4 4）を追加してもよい。N M I処理中にタイマ割込みがあった場合は、N M I処理終了後にタイマ割込処理に移行するようにしてもよい。

#### 【1 7 0 0】

##### <主制御部メイン処理>

次に、図2 4 1を用いて、主制御部3 0 0のC P U 3 0 4が実行する主制御部メイン処理の変形例について説明する。なお、同図は主制御部メイン処理の流れを示すフローチャートである。

10

20

30

40

50

## 【1701】

上述したように、主制御部300には、電源が投入されると起動信号（リセット信号）を出力する起動信号出力回路（リセット信号出力回路）340を設けている。この起動信号を入力した基本回路302のCPU304は、リセット割込によりリセットスタートしてROM306に予め記憶している制御プログラムに従って図241に示す主制御部メイン処理を実行する。

## 【1702】

ステップS8101では、初期設定1を行う。この初期設定1では、CPU304のスタックポインタ（SP）へのスタック初期値の設定（仮設定）、割込マスクの設定、I/O310の初期設定、RAM308に記憶する各種変数の初期設定、WDT314への動作許可及び初期値の設定等を行う。なお、本実施形態では、WDT314に、初期値として32.8msに相当する数値を設定する。

10

## 【1703】

ステップS8103では、低電圧信号がオンであるか否か、すなわち、電圧監視回路338が、電源制御部660から主制御部300に供給している電源の電圧値が所定の値（本実施形態では9V）未満である場合に電圧が低下したことを示す低電圧信号を出力しているか否かを監視する。そして、低電圧信号がオンの場合（CPU304が電源の遮断を検知した場合）にはステップS8103の処理を繰り返し実行し、低電圧信号がオフの場合（CPU304が電源の遮断を検知していない場合）にはステップS8107に進む。なお、電源が投入された直後で未だ上記所定の値（9V）に達しない場合にも、供給電圧がその所定の値以上になるまで、ステップS8103の処理を繰り返し実行する。

20

## 【1704】

ステップS8105では、初期設定2を行う。この初期設定2では、後述する主制御部タイマ割込処理を定期毎に実行するための周期を決める数値をカウンタタイマ312に設定する処理、I/O310の所定の出力ポート（例えば試験用出力ポート、第1副制御部400への出力ポート）からクリア信号を出力する処理、RAM308への書き込みを許可する設定等を行う。

## 【1705】

ステップS8107では、電源の遮断前（電断前）の状態に復帰するか否かの判定を行い、電断前の状態に復帰しない場合（主制御部300の基本回路302を初期状態にする場合）には初期化処理（ステップS8115）に進む。

30

## 【1706】

具体的には、最初に、電源基板に設けたRWMクリアスイッチ180を遊技店の店員などが操作した場合に送信されるRAMクリア信号がオン（操作があったことを示す）であるか否か、すなわちRAMクリアが必要であるか否かを判定し、RAMクリア信号がオンの場合（RAMクリアが必要な場合）には、基本回路302を初期状態にすべくステップS8115に進む。一方、RAMクリア信号がオフの場合（RAMクリアが必要でない場合）には、RAM308に設けた電源ステータス記憶領域に記憶した電源ステータスの情報を読み出し、この電源ステータスの情報がサスペンドを示す情報であるか否かを判定する。そして、電源ステータスの情報がサスペンドを示す情報でない場合には、基本回路302を初期状態にすべくステップS8115に進み、電源ステータスの情報がサスペンドを示す情報である場合には、RAM308の所定の領域（例えば全ての領域）に記憶している1バイトデータを初期値が0である1バイト構成のレジスタに全て加算することによりチェックサムを算出し、算出したチェックサムの結果が特定の値（例えば0）であるか否か（チェックサムの結果が正常であるか否か）を判定する。そして、チェックサムの結果が特定の値（例えば0）の場合（チェックサムの結果が正常である場合）には、その後の処理を遅延させるべくステップS8109に進んでから、電断前の状態に復帰すべくステップS8111に進む。チェックサムの結果が特定の値（例えば0）以外である場合（チェックサムの結果が異常である場合）には、パチンコ機100を初期状態にすべくステップS8115に進む。同様に電源ステータスの情報が「サスペンド」以外の情報を示し

40

50

ている場合にもステップ S 8 1 1 5 に進む。

【1707】

ステップ S 8 1 0 9 では、遅延処理を実行する（詳細は後述）。

【1708】

ステップ S 8 1 1 1 では、復電時処理を行う。この復電時処理では、電断時に R A M 3 0 8 に設けられたスタックポインタ退避領域に記憶しておいたスタックポインタの値を読み出し、スタックポインタに再設定（本設定）する。また、電断時に R A M 3 0 8 に設けられたレジスタ退避領域に記憶しておいた各レジスタの値を読み出し、各レジスタに再設定した後、割込許可の設定を行う。以降、C P U 3 0 4 が、再設定後のスタックポインタやレジスタに基づいて制御プログラムを実行する結果、パチンコ機 1 0 0 は電源断時の状態に復帰する。すなわち、電断直前にタイマ割込処理（後述）に分岐する直前に行った（ステップ S 8 1 2 1 内の所定の）命令の次の命令から処理を再開する。また、図 4 に示す主制御部 3 0 0 における基本回路 3 0 2 に搭載されている R A M 3 0 8 には、送信情報記憶領域が設けられている。このステップ S 8 1 1 1 では、その送信情報記憶領域に、復電コマンドをセットする。この復電コマンドは、電源断時の状態に復帰したことを表すコマンドであり、後述する、主制御部 3 0 0 のタイマ割込処理におけるステップ S 8 2 3 3 において、第 1 副制御部 4 0 0 へ送信される。

【1709】

ステップ S 8 1 1 3 では、W D T 3 1 4 による時間計測を開始する。

【1710】

ステップ S 8 1 1 5 では、初期化処理を行う。この初期化処理では、割込禁止の設定、スタックポインタへのスタック初期値の設定（本設定）、R A M 3 0 8 の全ての記憶領域の初期化などを行う。さらにここで、主制御部 3 0 0 の R A M 3 0 8 に設けられた送信情報記憶領域に正常復帰コマンドをセットする。この正常復帰コマンドは、主制御部 3 0 0 の初期化処理（ステップ S 8 1 1 5）が行われたことを表すコマンドであり、復電コマンドと同じく、主制御部 3 0 0 のタイマ割込処理におけるステップ S 8 2 3 3 において、第 1 副制御部 4 0 0 へ送信される。

【1711】

ステップ S 8 1 1 7 では、遅延処理を実行する（詳細は後述）。

【1712】

ステップ S 8 1 1 9 では、W D T 3 1 4 のカウンタの値をクリアし、W D T 3 1 4 による時間計測を開始する。

【1713】

ステップ S 8 1 2 1 では、割込禁止の設定を行った後、基本乱数初期値更新処理を行う。この基本乱数初期値更新処理では、普図当選乱数カウンタ、および特図乱数値カウンタの初期値をそれぞれ生成するための 2 つの初期値生成用乱数カウンタと、普図タイマ乱数値、および特図タイマ乱数値それぞれを生成するための 2 つの乱数カウンタを更新する。例えば、普図タイマ乱数値として取り得る数値範囲が 0 ～ 1 0 0 とすると、R A M 3 0 8 に設けた普図タイマ乱数値を生成するための乱数カウンタ記憶領域から値を取得し、取得した値に 1 を加算してから元の乱数カウンタ記憶領域に記憶する。このとき、取得した値に 1 を加算した結果が 1 0 1 であれば 0 を元の乱数カウンタ記憶領域に記憶する。他の初期値生成用乱数カウンタ、乱数カウンタもそれぞれ同様に更新する。なお、初期値生成用乱数カウンタは、後述するステップ S 8 2 0 7 でも更新する。なお、ステップ S 8 1 2 1 では、基本乱数初期値更新処理の他に後述する基本乱数更新処理を行ってもよいし、基本乱数更新処理のみを行ってもよい。

【1714】

主制御部 3 0 0 は、所定の周期ごとに開始するタイマ割込処理を行っている間を除いて、ステップ S 8 1 2 1 の処理を繰り返し実行する。

【1715】

<遅延処理>

10

20

30

40

50

次に、図 2 4 2 を用いて、主制御部 3 0 0 の C P U 3 0 4 が実行する遅延処理について説明する。なお、同図は主制御部メイン処理における遅延処理の流れを示すフローチャートである。

【1 7 1 6】

ステップ S 8 1 3 1 では、I / O の所定の出力ポートからクリア信号を送信する処理を実行する。

【1 7 1 7】

ステップ S 8 1 3 3 では、第 1 副制御部 4 0 0 の起動を待つ周期を決める副制御部起動待ちカウンタを初期設定（例えば 9 0 を設定）する。

【1 7 1 8】

ステップ S 8 1 3 5 では、割込み機能限定カウンタを初期設定（例えば 2 5 0 を設定）する。割込み機能限定カウンタは、4 m s 毎のタイマ割込み処理で減算される。

【1 7 1 9】

ステップ S 8 1 3 7 では、基本乱数初期値更新処理を実行する（詳細は後述）。なお、ステップ S 8 1 3 7 では、基本乱数初期値更新処理の他に後述する基本乱数更新処理を行ってもよいし、基本乱数更新処理のみを行ってもよい。

【1 7 2 0】

ステップ S 8 1 3 9 では、コマンド送信処理を行い、各種のコマンドが第 1 副制御部 4 0 0 に送信される。

【1 7 2 1】

ステップ S 8 1 4 1 では、割込み機能限定カウンタが 0 であるか否かの判定を行い、割込み機能限定カウンタが 0 以外の場合、ステップ S 8 1 3 7 の処理に戻る。一方、割込み機能限定カウンタが 0 の場合、すなわち、ステップ S 8 1 3 5 において割込み機能限定カウンタに 2 5 0 が設定され、4 m s 毎のタイマ割込み処理で減算されて 1 0 0 0 m s = 1 秒経過した場合、ステップ S 8 1 4 3 に進む。

【1 7 2 2】

ステップ S 8 1 4 3 では、副制御部起動待ちカウンタを 1 減算する。

【1 7 2 3】

ステップ S 8 1 4 5 では、副制御部起動待ちカウンタが 0 であるか否かの判定を行い、副制御部起動待ちカウンタが 0 以外の場合、ステップ S 8 1 3 5 に戻る。一方、副制御部起動待ちカウンタが 0 の場合、すなわち、ステップ S 8 1 3 3 において副制御部起動待ちカウンタに 9 0 が設定され、9 0 秒経過した場合、ステップ S 8 1 4 7 に進む。

【1 7 2 4】

ステップ S 8 1 4 7 では、その他の処理を実行する。

【1 7 2 5】

<主制御部タイマ割込処理>

次に、図 2 4 3 を用いて、主制御部 3 0 0 の C P U 3 0 4 が実行する主制御部タイマ割込処理について説明する。なお、同図は主制御部タイマ割込処理の流れを示すフローチャートである。

【1 7 2 6】

主制御部 3 0 0 は、所定の周期（本実施形態では約 2 m s に 1 回）でタイマ割込信号を発生するカウンタタイマ 3 1 2 を備えており、このタイマ割込信号を契機として主制御部タイマ割込処理を所定の周期で開始する。

【1 7 2 7】

ステップ S 8 2 0 1 では、タイマ割込開始処理を行う。このタイマ割込開始処理では、C P U 3 0 4 の各レジスタの値をスタック領域に一時的に退避する処理などを行う。

【1 7 2 8】

ステップ S 8 2 0 3 では、W D T 3 1 4 のカウント値が初期設定値（本実施形態では 3 2 . 8 m s ）を超えて W D T 割込が発生しないように（処理の異常を検出しないように）、W D T を定期的に（本実施形態では、主制御部タイマ割込の周期である約 2 m s に 1 回

10

20

30

40

50

）クリア&リスタートを行う。

#### 【1729】

ステップS8205では、入力ポート状態更新処理を行う。この入力ポート状態更新処理では、I/O310の入力ポートを介して、上述の前面枠扉開放センサや内枠開放センサや下皿満タンセンサ、各種の球検出センサを含む各種センサ320の検出信号を入力して検出信号の有無を監視し、RAM308に各種センサ320ごとに区画して設けた信号状態記憶領域に記憶する。球検出センサの検出信号を例にして説明すれば、前々回のタイマ割込処理（約4ms前）で検出した各々の球検出センサの検出信号の有無の情報を、RAM308に各々の球検出センサごとに区画して設けた前回検出信号記憶領域から読み出し、この情報をRAM308に各々の球検出センサごとに区画して設けた前々回検出信号記憶領域に記憶し、前回のタイマ割込処理（約2ms前）で検出した各々の球検出センサの検出信号の有無の情報を、RAM308に各々の球検出センサごとに区画して設けた今回検出信号記憶領域から読み出し、この情報を上述の前回検出信号記憶領域に記憶する。また、今回検出した各々の球検出センサの検出信号を、上述の今回検出信号記憶領域に記憶する。

10

#### 【1730】

また、ステップS8205では、上述の前々回検出信号記憶領域、前回検出信号記憶領域、および今回検出信号記憶領域の各記憶領域に記憶した各々の球検出センサの検出信号の有無の情報を比較し、各々の球検出センサにおける過去3回分の検出信号の有無の情報が入賞判定パターン情報と一致するか否かを判定する。一個の遊技球が一つの球検出センサを通過する間に、約2msという非常に短い間隔で起動を繰り返すこの主制御部タイマ割込処理は何回か起動する。このため、主制御部タイマ割込処理が起動する度に、上述のステップS8205では、同じ遊技球が同じ球検出センサを通過したことを表す検出信号を確認することになる。この結果、上述の前々回検出信号記憶領域、前回検出信号記憶領域、および今回検出信号記憶領域それぞれに、同じ遊技球が同じ球検出センサを通過したことを表す検出信号が記憶される。すなわち、遊技球が球検出センサを通過し始めたときには、前々回検出信号無し、前回検出信号有り、今回検出信号有りになる。本実施形態では、球検出センサの誤検出やノイズを考慮して、検出信号無しの後に検出信号が連続して2回記憶されている場合には、入賞があったと判定する。図4に示す主制御部300のROM306には、入賞判定パターン情報（本実施形態では、前々回検出信号無し、前回検出信号有り、今回検出信号有りであることを示す情報）が記憶されている。このステップS8205では、各々の球検出センサにおいて過去3回分の検出信号の有無の情報が、予め定めた入賞判定パターン情報（本実施形態では、前々回検出信号無し、前回検出信号有り、今回検出信号有りであることを示す情報）と一致した場合に、一般入賞口226、可変入賞口234、第1特図始動口230、および第2特図始動口232への入球、または普図始動口228の通過があったと判定する。すなわち、これらの入賞口226、234やこれらの始動口230、232、228への入賞があったと判定する。例えば、一般入賞口226への入球を検出する一般入賞口センサにおいて過去3回分の検出信号の有無の情報が上述の入賞判定パターン情報と一致した場合には、一般入賞口226へ入賞があったと判定し、以降の一般入賞口226への入賞に伴う処理を行うが、過去3回分の検出信号の有無の情報が上述の入賞判定パターン情報と一致しなかった場合には、以降の一般入賞口226への入賞に伴う処理を行わずに後続の処理に分岐する。なお、主制御部300のROM306には、入賞判定クリアパターン情報（本実施形態では、前々回検出信号有り、前回検出信号無し、今回検出信号無しであることを示す情報）が記憶されている。入賞が一度あったと判定した後は、各々の球検出センサにおいて過去3回分の検出信号の有無の情報が、その入賞判定クリアパターン情報に一致するまで入賞があったとは判定せず、入賞判定クリアパターン情報に一致すれば、次からは上記入賞判定パターン情報に一致するか否かの判定を行う。

20

30

40

#### 【1731】

ステップS8207およびステップS8209では、基本乱数初期値更新処理および基

50



本乱数更新処理を行う。これらの基本乱数初期値更新処理および基本乱数更新処理では、上記ステップS 8 1 1 5で行った初期値生成用乱数カウンタの値の更新を行い、次に主制御部3 0 0で使用する、普図当選乱数値、特図1乱数値、および特図2乱数値をそれぞれ生成するための2つの乱数カウンタを更新する。例えば、普図当選乱数値として取り得る数値範囲が0～1 0 0とすると、R A M 3 0 8に設けた普図当選乱数値を生成するための乱数カウンタ記憶領域から値を取得し、取得した値に1を加算してから元の乱数カウンタ記憶領域に記憶する。このとき、取得した値に1を加算した結果が1 0 1であれば0を元の乱数カウンタ記憶領域に記憶する。また、取得した値に1を加算した結果、乱数カウンタが一周していると判定した場合にはそれぞれの乱数カウンタに対応する初期値生成用乱数カウンタの値を取得し、乱数カウンタの記憶領域にセットする。例えば、0～1 0 0の数値範囲で変動する普図当選乱数値生成用の乱数カウンタから値を取得し、取得した値に1を加算した結果が、R A M 3 0 8に設けた所定の初期値記憶領域に記憶している前回設定した初期値と等しい値（例えば7）である場合に、普図当選乱数値生成用の乱数カウンタに対応する初期値生成用乱数カウンタから値を初期値として取得し、普図当選乱数値生成用の乱数カウンタにセットすると共に、普図当選乱数値生成用の乱数カウンタが次に1周したことを判定するために、今回設定した初期値を上述の初期値記憶領域に記憶しておく。また、普図当選乱数値生成用の乱数カウンタが次に1周したことを判定するための上述の初期値記憶領域とは別に、特図乱数生成用の乱数カウンタが1周したことを判定するための初期値記憶領域をR A M 3 0 8に設けている。なお、本実施形態では特図1の乱数値を取得するためのカウンタと特図2の乱数値を取得するためのカウンタとを別に設けたが、同一のカウンタを用いてもよい。

#### 【1 7 3 2】

ステップS 8 2 1 1では、演出乱数更新処理を行う。この演出乱数更新処理では、主制御部3 0 0で使用する演出用乱数値を生成するための乱数カウンタを更新する。

#### 【1 7 3 3】

ステップS 8 2 1 2では、割込み機能限定カウンタが0であるか否かの判定を行い、割込み機能限定カウンタが0以外の場合、ステップS 8 2 1 3に進む。一方、割込み機能限定カウンタが0の場合、ステップS 8 2 1 4に進む。

#### 【1 7 3 4】

ステップS 8 2 1 3では、割込み機能限定カウンタを1減算し、その後、ステップS 8 2 3 9に進む。

#### 【1 7 3 5】

ステップS 8 2 1 4では、特定異常検出フラグ（詳しくは後述する。）がONであるか否かの判定を行い、特定異常検出フラグがONの場合、ステップS 8 2 3 9に進む。一方、特定異常検出フラグがOFFの場合、ステップS 8 2 1 5に進む。

#### 【1 7 3 6】

ステップS 8 2 1 5では、タイマ更新処理を行う。このタイマ更新処理では、普通図柄表示装置2 1 0に図柄を変動・停止表示する時間を計時するための普図表示図柄更新タイマ、第1特別図柄表示装置2 1 2に図柄を変動・停止表示する時間を計時するための特図1表示図柄更新タイマ、第2特図表示装置2 1 4に図柄を変動・停止表示する時間を計時するための特図2表示図柄更新タイマ、所定の入賞演出時間、所定の開放時間、所定の閉鎖時間、所定の終了演出期間などを計時するためのタイマなどを含む各種タイマを更新する。

#### 【1 7 3 7】

ステップS 8 2 1 6では、入賞口カウンタ更新処理を行う。この入賞口カウンタ更新処理では、入賞口2 2 6、2 3 4や始動口2 3 0、2 3 2、2 2 8に入賞があった場合に、R A M 3 0 8に各入賞口ごと、あるいは各始動口ごとに設けた賞球数記憶領域の値を読み出し、1を加算して、元の賞球数記憶領域に設定する。

#### 【1 7 3 8】

また、ステップS 8 2 1 7では、入賞受付処理を行う。この入賞受付処理では、第1特

図始動口 2 3 0、第 2 特図始動口 2 3 2、普図始動口 2 2 8 および可変入賞口 2 3 4 への入賞があったか否かを判定する。ここでは、ステップ S 8 2 0 3 における入賞判定パターン情報と一致するか否かの判定結果を用いて判定する。第 1 特図始動口 2 3 0 へ入賞があった場合且つ R A M 3 0 8 に設けた対応する保留数記憶領域が満タンでない場合、カウンタ回路 3 1 8 の当選用カウンタ値記憶用レジスタから値を特図 1 当選乱数値として取得するとともに特図 1 乱数値生成用の乱数カウンタから値を特図 1 乱数値として取得して対応する乱数値記憶領域に格納する。第 2 特図始動口 2 3 2 へ入賞があった場合且つ R A M 3 0 8 に設けた対応する保留数記憶領域が満タンでない場合、カウンタ回路 3 1 8 の当選用カウンタ値記憶用レジスタから値を特図 2 当選乱数値として取得するとともに特図 2 乱数値生成用の乱数カウンタから値を特図 2 乱数値として取得して対応する乱数値記憶領域に格納する。普図始動口 2 2 8 へ入賞があった場合且つ R A M 3 0 8 に設けた対応する保留数記憶領域が満タンでない場合、普図当選乱数値生成用の乱数カウンタから値を普図当選乱数値として取得して対応する乱数値記憶領域に格納する。可変入賞口 2 3 4 へ入賞があった場合には、可変入賞口用の入賞記憶領域に、可変入賞口 2 3 4 に球が入球したことを示す情報を格納する。

#### 【1739】

ステップ S 8 2 1 9 では、払出要求数送信処理を行う。なお、払出制御部 6 0 0 に出力する出力予定情報および払出要求情報は、例えば 1 バイトで構成しており、ビット 7 にストロブ情報（オンの場合、データをセットしていることを示す）、ビット 6 に電源投入情報（オンの場合、電源投入後一回目のコマンド送信であることを示す）、ビット 4 ～ 5 に暗号化のための今回加工種別（0 ～ 3）、およびビット 0 ～ 3 に暗号化加工後の払出要求数を示すようにしている。

#### 【1740】

ステップ S 8 2 2 1 では、普図状態更新処理を行う。この普図状態更新処理は、普図の状態に対応する複数の処理のうちの 1 つの処理を行う。例えば、普図変動表示の途中（上述する普図表示図柄更新タイマの値が 1 以上）における普図状態更新処理では、普通図柄表示装置 2 1 0 を構成する 7 セグメント L E D の点灯と消灯を繰り返す点灯・消灯駆動制御を行う。この制御を行うことで、普通図柄表示装置 2 1 0 は普図の変動表示（普図変動遊技）を行う。

#### 【1741】

また、普図変動表示時間が経過したタイミング（普図表示図柄更新タイマの値が 1 から 0 になったタイミング）における普図状態更新処理では、当りフラグがオンの場合には、当り図柄の表示態様となるように普通図柄表示装置 2 1 0 を構成する 7 セグメント L E D の点灯・消灯駆動制御を行い、当りフラグがオフの場合には、外れ図柄の表示態様となるように普通図柄表示装置 2 1 0 を構成する 7 セグメント L E D の点灯・消灯駆動制御を行う。また、主制御部 3 0 0 の R A M 3 0 8 には、普図状態更新処理に限らず各種の処理において各種の設定を行う設定領域が用意されている。ここでは、上記点灯・消灯駆動制御を行うとともに、その設定領域に普図停止表示中であることを示す設定を行う。この制御を行うことで、普通図柄表示装置 2 1 0 は、当り図柄（図 5（c）に示す普図 A）および外れ図柄（図 5（c）に示す普図 B）いずれか一方の図柄の確定表示を行う。さらにその後、所定の停止表示期間（例えば 5 0 0 m 秒間）、その表示を維持するために R A M 3 0 8 に設けた普図停止時間管理用タイマの記憶領域に停止期間を示す情報を設定する。この設定により、確定表示された図柄が所定期間停止表示され、普図変動遊技の結果が遊技者に報知される。

#### 【1742】

また、普図変動遊技の結果が当りであれば、後述するように、普図当りフラグがオンされる。この普図当りフラグがオンの場合には、所定の停止表示期間が終了したタイミング（普図停止時間管理用タイマの値が 1 から 0 になったタイミング）における普図状態更新処理では、R A M 3 0 8 の設定領域に普図作動中を設定するとともに、所定の開放期間（例えば 2 秒間）、第 2 特図始動口 2 3 2 の羽根部材 2 3 2 a の開閉駆動用のソレノイド（

3 3 2) に、羽根部材 2 3 2 a を開放状態に保持する信号を出力するとともに、R A M 3 0 8 に設けた羽根開放時間管理用タイマの記憶領域に開放期間を示す情報を設定する。

【1 7 4 3】

また、所定の開放期間が終了したタイミング（羽根開放時間管理用タイマの値が 1 から 0 になったタイミング）で開始する普図状態更新処理では、所定の閉鎖期間（例えば 5 0 0 m 秒間）、羽根部材の開閉駆動用のソレノイド 3 3 2 に、羽根部材を閉鎖状態に保持する信号を出力するとともに、R A M 3 0 8 に設けた羽根閉鎖時間管理用タイマの記憶領域に閉鎖期間を示す情報を設定する。

【1 7 4 4】

また、所定の閉鎖期間が終了したタイミング（羽根閉鎖時間管理用タイマの値が 1 から 0 になったタイミング）で開始する普図状態更新処理では、R A M 3 0 8 の設定領域に普図非作動中を設定する。さらに、普図変動遊技の結果が外れであれば、後述するように、普図外れフラグがオンされる。この普図外れフラグがオンの場合には、上述した所定の停止表示期間が終了したタイミング（普図停止時間管理用タイマの値が 1 から 0 になったタイミング）における普図状態更新処理でも、R A M 3 0 8 の設定領域に普図非作動中を設定する。普図非作動中の場合における普図状態更新処理では、何もせずに次のステップ S 8 2 2 3 に移行するようにしている。

【1 7 4 5】

ステップ S 8 2 2 3 では、普図関連抽選処理を行う。この普図関連抽選処理では、普図変動遊技および第 2 特図始動口 2 3 2 の開閉制御を行っておらず（普図の状態が非作動中）、且つ、保留している普図変動遊技の数が 1 以上である場合に、上述の乱数値記憶領域に記憶している普図当選乱数値に基づいた乱数抽選により普図変動遊技の結果を当選とするか、不当選とするかを決定する当り判定をおこない、当選とする場合には R A M 3 0 8 に設けた当りフラグにオンを設定する。不当選の場合には、当りフラグにオフを設定する。また、当り判定の結果に関わらず、次に上述の普図タイマ乱数値生成用の乱数カウンタの値を普図タイマ乱数値として取得し、取得した普図タイマ乱数値に基づいて複数の変動時間のうちから普図表示装置 2 1 0 に普図を変動表示する時間を 1 つ選択し、この変動表示時間を、普図変動表示時間として、R A M 3 0 8 に設けた普図変動時間記憶領域に記憶する。なお、保留している普図変動遊技の数は、R A M 3 0 8 に設けた普図保留数記憶領域に記憶するようにしており、当り判定をするたびに、保留している普図変動遊技の数から 1 を減算した値を、この普図保留数記憶領域に記憶し直すようにしている。また当り判定に使用した乱数値を消去する。

【1 7 4 6】

次いで、特図 1 および特図 2 それぞれについての特図状態更新処理を行うが、最初に、特図 2 についての特図状態更新処理（特図 2 状態更新処理）を行う（ステップ S 8 2 2 5）。この特図 2 状態更新処理は、特図 2 の状態に応じて、次の 8 つの処理のうちの 1 つの処理を行う。例えば、特図 2 変動表示の途中（上述の特図 2 表示図柄更新タイマの値が 1 以上）における特図 2 状態更新処理では、第 2 特別図柄表示装置 2 1 4 を構成する 7 セグメント L E D の点灯と消灯を繰り返す点灯・消灯駆動制御を行う。この制御を行うことで、第 2 特別図柄表示装置 2 1 4 は特図 2 の変動表示（特図 2 変動遊技）を行う。

【1 7 4 7】

また、コマンド設定送信処理（ステップ S 8 2 3 3）で回転開始設定送信処理を実行させることを示す所定の送信情報を上述の送信情報記憶領域に追加記憶してから処理を終了する。

【1 7 4 8】

また、主制御部 3 0 0 の R A M 3 0 8 には、1 5 R 大当りフラグ、2 R 大当りフラグ、第 1 小当りフラグ、第 2 小当りフラグ、第 1 はずれフラグ、第 2 はずれフラグ、特図確率変動フラグ、および普図確率変動フラグそれぞれのフラグが用意されている。特図 2 変動表示時間が経過したタイミング（特図 2 表示図柄更新タイマの値が 1 から 0 になったタイミング）で開始する特図 2 状態更新処理では、1 5 R 大当りフラグはオン、特図確率変動

10

20

30

40

50

フラグもオン、普図確率変動フラグもオンの場合には図5(a)に示す特図A、15R大当りフラグはオン、特図確率変動フラグはオフ、普図確率変動フラグはオンの場合には特図B、2R大当りフラグはオン、特図確率変動フラグもオン、普図確率変動フラグもオンの場合には特図C、2R大当りフラグはオン、特図確率変動フラグはオフ、普図確率変動フラグはオンの場合には特図D、2R大当りフラグはオン、特図確率変動フラグもオン、普図確率変動フラグはオンの場合には特図E、2R大当りフラグはオン、特図確率変動フラグはオフ、普図確率変動フラグもオフの場合には特図F、第1小当りフラグがオンの場合には特図G、第2小当りフラグがオンの場合には特図H、第1はずれフラグがオンの場合には特図I、第2はずれフラグがオンの場合には特図Iそれぞれの態様となるように、第2特別図柄表示装置214を構成する7セグメントLEDの点灯・消灯駆動制御を行い、RAM308の設定領域に特図2停止表示中であることを表す設定を行う。この制御を行うことで、第2特別図柄表示装置214は、15R特別大当り図柄(特図A)、15R大当り図柄(特図B)、突然確変図柄(特図C)、突然時短図柄(特図D)、隠れ確変図柄(特図E)、突然通常図柄(特図F)、第1小当り図柄(特図G)、第2小当り図柄(特図H)、第1はずれ図柄(特図I)、および第2はずれ図柄(特図J)のいずれか一つの図柄の確定表示を行う。さらにその後、所定の停止表示期間(例えば500m秒間)その表示を維持するためにRAM308に設けた特図2停止時間管理用タイマの記憶領域に停止期間を示す情報を設定する。この設定により、確定表示された特図2が所定期間停止表示され、特図2変動遊技の結果が遊技者に報知される。また、RAM308に設けられた時短回数記憶部に記憶された時短回数が1以上であれば、その時短回数から1を減算し、減算結果が1から0となった場合は、特図確率変動中(詳細は後述)でなければ、時短フラグをオフする。さらに、大当り遊技中(特別遊技状態中)にも、時短フラグをオフする。

#### 【1749】

また、コマンド設定送信処理(ステップS8233)で回転停止設定送信処理を実行させることを示す所定の送信情報を上述の送信情報記憶領域に追加記憶するとともに、変動表示を停止する図柄が特図2であることを示す特図2識別情報を、後述するコマンドデータに含める情報としてRAM308に追加記憶してから処理を終了する。

#### 【1750】

また、特図2変動遊技の結果が大当りであれば、後述するように、大当りフラグがオンされる。この大当りフラグがオンの場合には、所定の停止表示期間が終了したタイミング(特図2停止時間管理用タイマの値が1から0になったタイミング)における特図2状態更新処理では、RAM308の設定領域に特図2作動中を設定するとともに、所定の入賞演出期間(例えば3秒間)すなわち装飾図柄表示装置208による大当りを開始することを遊技者に報知する画像を表示している期間待機するためにRAM308に設けた特図2待機時間管理用タイマの記憶領域に入賞演出期間を示す情報を設定する。また、コマンド設定送信処理(ステップS8233)で入賞演出設定送信処理を実行させることを示す所定の送信情報を上述の送信情報記憶領域に追加記憶する。

#### 【1751】

また、所定の入賞演出期間が終了したタイミング(特図2待機時間管理用タイマの値が1から0になったタイミング)で開始する特図2状態更新処理では、所定の開放期間(例えば29秒間、または可変入賞口234に所定球数(例えば10球)の遊技球の入賞を検出するまで)可変入賞口234の扉部材234aの開閉駆動用のソレノイド(332)に、扉部材234aを開放状態に保持する信号を出力するとともに、RAM308に設けた扉開放時間管理用タイマの記憶領域に開放期間を示す情報を設定する。また、コマンド設定送信処理(ステップS8233)で大入賞口開放設定送信処理を実行させることを示す所定の送信情報を上述の送信情報記憶領域に追加記憶する。

#### 【1752】

また、所定の開放期間が終了したタイミング(扉開放時間管理用タイマの値が1から0になったタイミング)で開始する特図2状態更新処理では、所定の閉鎖期間(例えば1.

5 秒間) 可変入賞口 2 3 4 の扉部材 2 3 4 a の開閉駆動用のソレノイド (3 3 2) に、扉部材 2 3 4 a を閉鎖状態に保持する信号を出力するとともに、RAM 3 0 8 に設けた扉閉鎖時間管理用タイマの記憶領域に閉鎖期間を示す情報を設定する。また、コマンド設定送信処理 (ステップ S 8 2 3 3) で大入賞口閉鎖設定送信処理を実行させることを示す所定の送信情報を上述の送信情報記憶領域に追加記憶する。

#### 【1 7 5 3】

また、この扉部材の開放・閉鎖制御を所定回数 (本実施例では 1 5 ラウンドか 2 ラウンド) 繰り返し、終了したタイミングで開始する特図 2 状態更新処理では、所定の終了演出期間 (例えば 3 秒間) すなわち装飾図柄表示装置 2 0 8 による大当りを終了することを遊技者に報知する画像を表示している期間待機するように設定するために RAM 3 0 8 に設けた演出待機時間管理用タイマの記憶領域に演出待機期間を示す情報を設定する。また、普図確率変動フラグがオンに設定されていれば、この大当り遊技の終了と同時に、RAM 3 0 8 に設けられた時短回数記憶部に時短回数 1 0 0 回をセットするとともに、RAM 3 0 8 に設けられた時短フラグをオンする。なお、その普図確率変動フラグがオフに設定されていれば、時短回数記憶部に時短回数をセットすることなく、また時短フラグをオンすることもない。ここにいう時短とは、特図変動遊技における大当りを終了してから、次の大当りを開始するまでの時間を短くするため、パチンコ機が遊技者にとって有利な状態になることをいう。この時短フラグがオンに設定されていると、普図高確率状態である。普図高確率状態では普図低確率状態に比べて、普図変動遊技に大当りする可能性が高い。また、普図高確率状態の方が、普図低確率状態に比べて普図変動遊技の変動時間および特図変動遊技の変動時間は短くなる。さらに、普図高確率状態では普図低確率状態に比べて、第 2 特別始動口 2 3 2 の一对の羽根部材 2 3 2 a の 1 回の開放における開放時間が長くなりやすい。加えて、普図高確率状態では普図低確率状態に比べて、一对の羽根部材 2 3 2 a は多く開きやすい。また、上述のごとく、時短フラグは、大当り遊技中 (特別遊技状態中) にはオフに設定される。したがって、大当り遊技中には、普図低確率状態が維持される。これは、大当り遊技中に普図高確率状態であると、大当り遊技中に可変入賞口 2 3 4 に所定の個数、遊技球が入球するまでの間に第 2 特図始動口 2 3 2 に多くの遊技球が入球し、大当り中に獲得することができる遊技球の数が多くなってしまい射幸性が高まってしまいうという問題があり、これを解決するためのものである。

#### 【1 7 5 4】

さらに、コマンド設定送信処理 (ステップ S 8 2 3 3) で終了演出設定送信処理を実行させることを示す所定の送信情報を上述の送信情報記憶領域に追加記憶する。

#### 【1 7 5 5】

また、所定の終了演出期間が終了したタイミング (演出待機時間管理用タイマの値が 1 から 0 になったタイミング) で開始する特図 2 状態更新処理では、RAM 3 0 8 の設定領域に特図 2 非作動中を設定する。さらに、特図 2 変動遊技の結果が外れであれば、後述するように、はずれフラグがオンされる。このはずれフラグがオンの場合には、上述した所定の停止表示期間が終了したタイミング (特図 2 停止時間管理用タイマの値が 1 から 0 になったタイミング) における特図 2 状態更新処理でも、RAM 3 0 8 の設定領域に特図 2 非作動中を設定する。特図 2 非作動中の場合における特図 2 状態更新処理では、何もせずに次のステップ S 8 2 2 7 に移行するようにしている。

#### 【1 7 5 6】

続いて、特図 1 についての特図状態更新処理 (特図 1 状態更新処理) を行う (ステップ S 8 2 2 7)。この特図 1 状態更新処理では、特図 1 の状態に応じて、上述の特図 2 状態更新処理で説明した各処理を行う。この特図 1 状態更新処理で行う各処理は、上述の特図 2 状態更新処理で説明した内容の「特図 2」を「特図 1」と読み替えた処理と同一であるため、その説明は省略する。なお、特図 2 状態更新処理と特図 1 状態更新処理の順番は逆でもよい。

#### 【1 7 5 7】

ステップ S 8 2 2 5 およびステップ S 8 2 2 7 における特図状態更新処理が終了すると

、今度は、特図 1 および特図 2 それぞれについての特図関連抽選処理を行う。ここでも先に、特図 2 についての特図関連抽選処理（特図 2 関連抽選処理）を行い（ステップ S 8 2 2 9）、その後で、特図 1 についての特図関連抽選処理（特図 1 関連抽選処理）を行う（ステップ S 8 2 3 1）。これらの特図関連抽選処理についても、主制御部 3 0 0 が特図 2 関連抽選処理を特図 1 関連抽選処理よりも先に行うことで、特図 2 変動遊技の開始条件と、特図 1 変動遊技の開始条件が同時に成立した場合でも、特図 2 変動遊技が先に変動中となるため、特図 1 変動遊技は変動を開始しない。また、装飾図柄表示装置 2 0 8 による、特図変動遊技の大当たり判定の結果の報知は、第 1 副制御部 4 0 0 によって行われ、第 2 特図始動口 2 3 2 への入賞に基づく抽選の抽選結果の報知が、第 1 特図始動口 2 3 0 への入賞に基づく抽選の抽選結果の報知よりも優先して行われる。

10

**【1758】**

ステップ S 8 2 3 2 では、デバイス監視処理を実行する（詳細は後述）。

**【1759】**

ステップ S 8 2 3 3 では、コマンド設定送信処理を行い、各種のコマンド（例えば、ステップ S 8 2 3 2 のデバイス監視処理で特定異常があった場合は、特定異常を示す信号）が第 1 副制御部 4 0 0 に送信される。なお、第 1 副制御部 4 0 0 に送信する出力予定情報は例えば 1 6 ビットで構成しており、ビット 1 5 はストロブ情報（オンの場合、データをセットしていることを示す）、ビット 1 1 ~ 1 4 はコマンド種別（本実施形態では、基本コマンド、図柄変動開始コマンド、図柄変動停止コマンド、入賞演出開始コマンド、終了演出開始コマンド、大当たりラウンド数指定コマンド、復電コマンド、F R A M（登録商標）クリアコマンドなどコマンドの種類を特定可能な情報）、ビット 0 ~ 1 0 はコマンドデータ（コマンド種別に対応する所定の情報）で構成している。

20

**【1760】**

具体的には、ストロブ情報は上述のコマンド送信処理でオン、オフするようにしている。また、コマンド種別が図柄変動開始コマンドの場合であればコマンドデータに、1 5 R 大当たりフラグや 2 R 大当たりフラグの値、特図確率変動フラグの値、特図関連抽選処理で選択したタイマ番号などを示す情報を含み、図柄変動停止コマンドの場合であれば、1 5 R 大当たりフラグや 2 R 大当たりフラグの値、特図確率変動フラグの値などを含み、入賞演出コマンドおよび終了演出開始コマンドの場合であれば、特図確率変動フラグの値などを含み、大当たりラウンド数指定コマンドの場合であれば特図確率変動フラグの値、大当たりラウンド数などを含むようにしている。コマンド種別が基本コマンドを示す場合は、コマンドデータにデバイス情報、第 1 特図始動口 2 3 0 への入賞の有無、第 2 特図始動口 2 3 2 への入賞の有無、可変入賞口 2 3 4 への入賞の有無などを含む。

30

**【1761】**

また、上述の回転開始設定送信処理では、コマンドデータに R A M 3 0 8 に記憶している、1 5 R 大当たりフラグや 2 R 大当たりフラグの値、特図確率変動フラグの値、特図 1 関連抽選処理および特図 2 関連抽選処理で選択したタイマ番号、保留している第 1 特図変動遊技または第 2 特図変動遊技の数などを示す情報を設定する。上述の回転停止設定送信処理では、コマンドデータに R A M 3 0 8 に記憶している、1 5 R 大当たりフラグや 2 R 大当たりフラグの値、特図確率変動フラグの値などを示す情報を設定する。上述の入賞演出設定送信処理では、コマンドデータに、R A M 3 0 8 に記憶している、入賞演出期間中に装飾図柄表示装置 2 0 8 ・各種ランプ 4 1 8 ・スピーカ 1 2 0 に出力する演出制御情報、特図確率変動フラグの値、保留している第 1 特図変動遊技または第 2 特図変動遊技の数などを示す情報を設定する。上述の終了演出設定送信処理では、コマンドデータに、R A M 3 0 8 に記憶している、演出待機期間中に装飾図柄表示装置 2 0 8 ・各種ランプ 4 1 8 ・スピーカ 1 2 0 に出力する演出制御情報、特図確率変動フラグの値、保留している第 1 特図変動遊技または第 2 特図変動遊技の数などを示す情報を設定する。上述の大入賞口開放設定送信処理では、コマンドデータに R A M 3 0 8 に記憶している大当たりラウンド数、特図確率変動フラグの値、保留している第 1 特図変動遊技または第 2 特図変動遊技の数などを示す情報を設定する。上述の大入賞口閉鎖設定送信処理では、コマンドデータに R A M 3 0 8

40

50

に記憶している大当たりラウンド数、特図確率変動フラグの値、保留している第1特図変動遊技または第2特図変動遊技の数などを示す情報を設定する。また、このステップS8233では一般コマンド特図保留増加処理も行われる。この一般コマンド特図保留増加処理では、コマンドデータにRAM308の送信用情報記憶領域に記憶している特図識別情報（特図1または特図2を示す情報）、予告情報（事前予告情報、偽事前予告情報、または事前予告無情報のいずれか）を設定する。

#### 【1762】

第1副制御部400では、受信した出力予定情報に含まれるコマンド種別により、主制御部300における遊技制御の変化に応じた演出制御の決定が可能になるとともに、出力予定情報に含まれているコマンドデータの情報に基づいて、演出制御内容を決定することができるようになる。

10

#### 【1763】

ステップS8235では、外部出力信号設定処理を行う。具体的には、外部端子板信号や試射試験信号などを設定する。また、ステップS8232のデバイス監視処理で特定異常があった場合は、特定異常を示す信号を設定する。この外部出力信号設定処理では、RAM308に記憶している遊技情報を、情報出力回路336を介してパチンコ機100とは別体の情報入力回路350に出力する。

#### 【1764】

ステップS8239では、低電圧信号がオンであるか否かを監視する。そして、低電圧信号がオンの場合（電源の遮断を検知した場合）にはステップS8243に進み、低電圧信号がオフの場合（電源の遮断を検知していない場合）にはステップS8241に進む。

20

#### 【1765】

ステップS8241では、タイマ割込終了処理を行う。このタイマ割込終了処理では、ステップS8201で一時的に退避した各レジスタの値を元の各レジスタに設定したり、割込許可の設定などを行い、その後、図241に示す主制御部メイン処理に復帰する。

#### 【1766】

一方、ステップS8243では、復電時に電断時の状態に復帰するための特定の変数やスタックポインタを復帰データとしてRAM308の所定の領域に退避し、入出力ポートの初期化等の電断処理を行い、その後、上述の主制御部メイン処理に復帰する。

#### 【1767】

なお、デバイス監視処理（ステップS8232）で監視するエラーとしては、次に示すようなものが考えられる。

30

#### 【1768】

エラー種別として、下受け皿満タンエラー（下皿満タンエラー）、払出装置エラー、払出超過エラー、不正払出エラー、払出個数スイッチエラー、主制御通信エラー、CRユニット未接続エラー、CRユニット通信エラー、磁気異常エラー、外部クロック異常エラー、磁界異常エラー、スイッチレベル異常エラー、枠開放エラー（前面枠扉開放エラー）、スイッチ未接続エラー、衝撃センサエラーなどがある。なお、これらのエラーうち、下受け皿満タンエラー等は、遊技機内に封入された遊技球を循環使用する封入式遊技機には適用されない。

40

#### 【1769】

下受け皿満タンエラーとは、下皿満タンスイッチ信号のオン状態が、メイン基板の所定の入力ポートによってタイマ割り込みでオンを1回読み込んだとき（オフ状態からオン状態に切り換わるとき）にメインから払出に出力されるものである。

#### 【1770】

払出装置エラーとは、払出モータが払出要求数分の駆動を終了後（実際は+ブレーキ期間+無励磁期間経過後）の次の割り込み内で、払出数（払出個数スイッチのカウント数）が払出要求数よりも少ない場合に払出装置エラーの発生を検知するものである。1個ずつ遊技球を払い出すリトライ処理が完了することにより、当該エラー状態が解除される。

#### 【1771】

50

払出超過エラーとは、払出要求数に対して実際に払い出された遊技球が10個多く検知されたとき払出動作を停止するものである。この払出超過エラーは、エラー解除スイッチ168によって解除するエラーであり、エラー解除スイッチ168を操作するか、RAMクリアスイッチ180による初期化(RAMクリア)によって当該エラー状態が解除される。

#### 【1772】

不正払出エラーとは、払出要求が発生していない状態で遊技球の払出を検知(払出個数スイッチがカウント)したとき払出動作を停止するものである。この不正払出エラーは、エラー解除スイッチ168によって解除するエラーであり、エラー解除スイッチ168を操作するか、RAMクリアスイッチ180による初期化(RAMクリア)によって当該エラー状態が解除される。

10

#### 【1773】

払出個数スイッチエラーとは、払出基板170に制御される遊技球の払出個数スイッチ(不図示)のエラーであり、払出個数スイッチの接続異常を検知したとき(メイン基板は関係なし)払出動作を停止するものである。この払出個数スイッチエラーは、エラー解除スイッチ168によって解除するエラーであり、スイッチの接続が正しいか確認した後、エラー解除スイッチ168を操作するか、RAMクリアスイッチ180による初期化(RAMクリア)によって当該エラー状態が解除される。

#### 【1774】

主制御通信エラーとは、主制御接続確認信号のオフ状態を検知した場合、または主制御動作確認信号のオン状態を2割り込み継続して検知した場合に発生するエラーであり、メイン基板からのコマンド受信が不能になるものである。但し、払出要求数が残存している場合は、その分までは払い出す。主制御接続確認信号のオン状態を検知した場合で、かつ、主制御動作確認信号のオフ状態を2割り込み継続して検知した場合に当該エラー状態が解除される。

20

#### 【1775】

CRユニット未接続エラーとは、CRユニット未接続エラーが解除状態のときに、CRユニット接続信号(VL信号)のオフ状態を検知し、16ms継続した場合に発生するものであり、貸出停止(賞球の払出には影響なし)となるものである。CRユニット接続信号のON状態を検知し、16ms継続した場合に当該エラー状態が解除される。なお、エラー解除した後、CRユニットREADY信号(BRDY信号)およびCRユニット貸出要求完了確認信号(BRQ信号)がオフ状態となるまでCRユニットとの通信が無効状態になる。

30

#### 【1776】

ここで、CRユニットとインタフェースの接続信号関係について説明する。各種信号が、CRユニットからインタフェースを介して払出基板へと伝達される場合、CRユニット接続信号とは、CRユニット内の+18V電源からなるVL信号である。CRユニットREADY信号とは、CRユニットが貸し出しの処理中であることを伝達するBRDY信号である。CRユニット貸出要求確認信号とは、CRユニットが基本単位分25個の貸出要求と貸出指示を伝達するBRQ信号である。

40

#### 【1777】

CRユニット通信エラーとは、貸出要求発生前にBRQ信号のオン状態を検知した場合に発生するものである(なお、他の条件もあるが、ここでは特に関係がないため割愛する)。遊技機側から通信異常を通知した後、BRDY信号およびBRQ信号をオフ状態にすることで当該エラー状態が解除される。

#### 【1778】

磁気異常エラーとは、所定量を超える磁力を検出したとき主制御動作を停止するものである。外部クロック異常エラーとは、乱数更新用クロックに異常を検出した場合に発生するエラーである。磁界異常エラーとは、所定量を超える磁界を検出した場合に発生するエラーである。

50



## 【1779】

スイッチレベル異常エラーとは、入賞スイッチを検出し、1 s 継続した場合に発生するエラーである。枠開放エラーとは、ガラス枠または内枠の開放を検出した場合に発生するエラーである。スイッチ未接続エラーとは、入賞スイッチ（不図示）のエラーであり、入賞スイッチの接続異常を検出した場合に発生するエラーである。

## 【1780】

衝撃センサエラーとは、衝撃センサが所定量を超える衝撃を検出したとき電源断まで継続してエラーを報知するものである。

## 【1781】

＜デバイス監視処理＞

次に、図244を用いて、主制御部300のCPU304が実行するデバイス監視処理について説明する。なお、同図は主制御部タイマ割込処理におけるデバイス監視処理の流れを示すフローチャートである。

## 【1782】

ステップS8251では、上述のエラーの有無を示すデバイス情報を初期設定する。

## 【1783】

ステップS8253では、信号状態記憶領域に記憶した各種センサの信号状態を読み出して、上述のエラーの有無を監視し、当該エラーを検出した場合に、ステップS8255に進む。一方、上述のエラーを検出しなかった場合には、当該デバイス監視処理を終了する。

## 【1784】

ステップS8255では、第1副制御部400に送信すべき送信情報に、ステップS8253で検出したエラーを示す異常情報（デバイス情報）を設定する。

## 【1785】

ステップS8257では、ステップS8255で設定した異常情報の中に、特定異常情報があるか否かを判定し、特定異常情報がある場合に、ステップS8257に進む。一方、特定異常情報がない場合に、ステップS8261に進む。なお、特定異常情報は、上述のエラーについての異常情報の中から、予め設定されている1つまたは複数のエラーの異常情報のことである。例えば、磁気異常エラーの異常情報を特定異常情報として予め設定しておいてもよい。

## 【1786】

ステップS8259では、特定異常検出フラグをオンにする。

## 【1787】

ステップS8261では、デバイスコマンド送信要求を設定する。

## 【1788】

次に、以上説明した実施形態6によるパチンコ機100の特徴的構成について説明する。

（1）本実施の形態によるパチンコ機100は、

遊技に関する複数種類の遊技制御処理それぞれを、メイン制御（例えば、主制御部メイン処理）および所定の割込み周期ごとに行われる割込み制御（例えば、主制御部タイマ割込処理）のうちの、少なくともいずれか一方の制御において実行する遊技制御手段（たとえば、主制御部300、ユーザプログラム）と、

前記遊技制御手段からの開始指示（例えば、自動起動処理または手動起動処理）を受けたことに基づいて経過時間の計測を開始し、該経過時間が特定時間（例えば、プログラム管理エリアに設定したタイムアウト時間）を超えたか否かを判定するとともに、該経過時間が特定時間を超えたと判定したに基づいて前記遊技制御手段に対して復帰指示（例えば、内部バスへのタイムアウト信号の出力）を行い、前記遊技制御手段から初期化指示（例えば、リスタート指示）を受けたことに基づいて該経過時間を初期化する復帰指示手段（例えば、WDT314）と、

監視している電源ライン（例えば、電源ラインp103）の電圧（例えば、DC12V

10

20

30

40

50

）が下限電圧（例えば、DC 9 V）よりも低下した場合に、前記遊技制御手段に出力する低電圧信号をアサート状態にし、前記電源ラインの電圧が下限電圧よりも高くなった場合に、前記低電圧信号をネゲート状態にすることが可能な電圧監視手段（例えば、リセットIC 7003）と

を備えた遊技台であって、

前記開始指示は、前記低電圧信号が前記アサート状態から前記ネゲート状態に変化した後に発せられることを特徴とする。

#### 【1789】

当該構成を備えたパチンコ機100によれば、電源監視信号としての低電圧信号がアサート状態からネゲート状態に変化した後にWDTが起動される。例えば、監視電圧が9 V以上になったら電源監視信号がネゲート状態になる。また、監視電圧が9 V未満になったら電源監視信号がアサート状態になる。低電圧信号がアサート状態からネゲート状態に変化した後にWDTが起動されるには、具体的には、（1）回路によってリセットICからリセット信号が出力されるタイミングを遅らせる、（2）固定延長機能の使用、（3）WDT手動起動が挙げられる。

#### 【1790】

従来は電源投入直後にWDTを起動し、電圧が安定するまでループ処理を行っていた。そして、そのループ処理内でWDTクリア&リスタートを繰り返し行っていた。この場合、電圧が安定するまで定期的にWDTクリア&リスタートの信号が出力されるので、該信号出力タイミングを基準にして特定処理の実行時期を判別しやすくなるという問題があった。また、電圧が安定する前にWDTクリア&リスタートを繰り返し行っていたので、ノイズの影響によって電圧が安定するまでの時間が長くなってしまいう問題があった。

本発明によれば、電圧安定までの時間を短縮、WDTクリア&リスタートの信号で特定処理の実行時期のを判別し難くできる場合がある。

#### 【1791】

また、WDTタイムアウトの場合はシステムリセット動作をさせることにより、特定処理の実行時期を更に判別困難にできる場合があり、ユーザプログラムの解析困難にできる場合がある。

#### 【1792】

また、WDTを手動起動させた場合において、電断処理が発生した場合にも、電断処理中にWDTを停止させないようにしてもよい。主制御部のタイマ割込み制御処理では低電圧信号を検出すると電断処理が実行される。この場合はパチンコ機の電気系統に異常が発生していることが考えられる。このときにWDTの残存タイマの値が短くなっていると直ちにタイムアウト信号が出力されてシステムリセットがかかってしまうおそれがある。電断処理前にWDTを一時停止させるのが普通であるが、停電などの通常の電圧降下の場合にはWDTに悪影響はほとんどなく、不正行為の場合には一時的に高い電圧がかけられるケースがありWDTに悪影響がある。つまり普通の遊技者には電断処理が完了するようにして、不正行為者には電断処理が完了しないようにするため、WDTを停止させないようにしてもよい。また、瞬停（9 V未満に電圧が低下してすぐに9 V以上に復活する状態）の場合であってもWDTを停止させないようにしてもよい。

#### 【1793】

また、磁界検知信号、磁気検知信号、電波検知信号、振動検知信号等が所定の端子に入力されている場合であってもWDTの起動を開始させ、さらにWDTのクリア&リスタートを実行するようにしてもよい。これらの検知信号に基づくエラー報知処理を行うよりも前に、WDTタイムアウトによって異常が発生していることを確認できる場合がある（特にエラー報知処理をタイマ割込み処理内で実行する場合に有効であり、さらに場合によってはタイマ割込み移行前に気づける場合がある。

#### 【1794】

払出制御部600にカードユニット（CRユニット）608が接続されていない状態であってもWDTの手動起動をすることにより、遊技機に異常が発生していることを確認で

10

20

30

40

50

きる場合がある。

【1795】

電断処理後に監視電圧が9Vに復帰した場合にWDT手動起動を再度行う（電断処理ではWDTを停止させない）ようにしてもよい。WDTのタイムアウト信号によるシステムリセット処理をさせないのでA0～A15のアドレスバスにモニタ信号（E000H）が出力されず所定のアドレスに復帰したことを判別し難くすることができる場合がある。

【1796】

また、電源投入後の最初のモジュール（スタックポインタを設定など）とは異なるモジュールで、WDTを起動させるようにしてもよい。こうすることにより、ユーザプログラムの解析を困難にできる場合がある。

10

【1797】

WDT手動起動とWDTクリア&リスタートの位置は、例えば以下のタイミングが挙げられる。

（a）電源投入後の低電圧信号出力判定ループ処理より前、当該処理中、または当該処理より後、のいずれか。

（b）内蔵レジスタに所定のデータを設定する処理より前、当該処理中、または当該処理より後、のいずれか。

（c）電源ステータスを参照する処理（分岐する処理）より前、当該処理中、または当該処理より後、のいずれか。

（d）チェックSUMのループ処理より前、当該処理中、または当該処理より後、のいずれか。

20

（e）内蔵RAMアクセス許可より前、当該処理中、または当該処理より後、のいずれか。

（f）復帰起動または初期起動より前、当該処理中、または当該処理より後、のいずれか。

（g）割込み許可より前、当該処理中、または当該処理より後、のいずれか。

（h）副制御部の起動待ちより前、当該処理中、または当該処理より後、のいずれか。

（i）ソフト乱数の更新開始より前、当該処理中、または当該処理より後、のいずれか。

（j）サブへの通信確認コマンド（サブ起動開始コマンド、完了コマンド）、払出への通信確認コマンドなどのコマンド送出より前、当該処理中、または当該処理より後、のいずれか。

30

（k）乱数最大値設定処理より前、当該処理中、または当該処理より後、のいずれか。

【1798】

（2）上記パチンコ機100であって、

前記開始指示は、前記メイン制御から前記割込み制御への移行を許可する最初の割込み許可前に発せられることを特徴とする。

【1799】

（3）上記パチンコ機100であって、

前記遊技制御手段を少なくとも有するマイクロプロセッサを備え、

前記マイクロプロセッサは、

40

少なくとも、前記復帰指示手段によって、前記経過時間が前記特定時間を超えたと判定された場合には、セキュリティチェックが行われるとともに、該セキュリティチェックの後に前記遊技制御が行われるユーザモードへ移行するまでの時間をランダムに変動させるランダム延長処理が少なくとも行われる、セキュリティモードに滞在した後、該ユーザモードへ移行するものであることを特徴とする。

【1800】

当該構成を備えたパチンコ機100によれば、ユーザモードに移行する前にランダム延長処理を実行するので、遊技制御プログラムの実行開始タイミングをずらすことが可能なため、遊技制御処理の特定の処理（例えばソフト乱数の更新処理、初期値更新処理等）の実行時期を判別困難にでき、これらの特定の処理の狙い撃ちを防止することができる場合

50

がある。

【1801】

(4) 上記パチンコ機100であって、前記ランダム延長処理の開始前に、前記マイクロプロセッサの所定の出力端子から所定の信号が出力されることを特徴とする。

(5) 上記パチンコ機100であって、前記所定の端子は、前記マイクロプロセッサ以外の外部機器に対してリセット信号を出力可能なリセット端子であることを特徴とする。

【1802】

当該構成を備えたパチンコ機100によれば、ランダム延長処理の前に外部機器に対してリセット信号を出力し、ランダム延長処理の後に遊技制御プログラムの実行を開始することができるので、遊技制御処理の特定の処理（例えばソフト乱数の更新処理、初期値更新処理等）の実行時期を判別困難にでき、これらの特定の処理の狙い撃ちを防止することができる場合がある。

10

【1803】

(6) 上記パチンコ機100であって、前記遊技制御手段から送出されるコマンドに基づいて演出制御を行う副制御手段を有することを特徴とする。

【1804】

当該構成を備えたパチンコ機100によれば、電源投入後、WDT手動起動開始前に電断した場合において、主制御部300がメイン処理に復帰して各種処理終了後のタイマ割込み処理内の所定の処理実行時に払出制御部600に対して復帰コマンドを送信する構成の場合、かつ払出制御部600はこのコマンドを受信しないと発射許可しない構成の場合は、遊技球の発射ができないことで遊技機に異常が発生していることを確認できる場合がある。

20

【1805】

<実施形態6>

以下、図面を用いて、本発明の実施形態6に係るパチンコ機（遊技台）について詳細に説明する。

【1806】

図245は、乱数生成回路から16ビット乱数値をRAM308に取込む手順を示している。図の右側には後程説明する実施形態7に係る図270に示す16ビット乱数回路9000の乱数生成回路9006で更新される16ビット乱数値が図の上から下に進む時系列で行表示されている。各行の左側は16ビット乱数値の上位バイトを示し、右側は下位バイトを示している。上から下に向かって16ビット乱数値が順次更新されている状態を示している。本例では乱数値の最大値が設定されていないので、16ビット乱数値が順次更新されて0～65535（十進数）まで更新される。図の中央にはマイコンの内蔵レジスタのうち乱数値取込みに関連するレジスタとして、上から順に、乱数値ソフトラッチレジスタ、ソフトラッチ乱数値レジスタ（下位）、ソフトラッチ乱数値レジスタ（上位）、乱数ソフトラッチフラグレジスタが示されている。図の左側には、RAM308内で連続した2つの所定アドレスの記憶領域に格納されるデータを示している。

30

【1807】

図245に示す乱数値取込みでは、ユーザプログラムにより内部レジスタの乱数ソフトラッチレジスタに対し「1」が書き込まれる（矢印（1）参照）。これにより、所定のソフトラッチ乱数値レジスタにラッチ信号が出力されて、乱数生成回路9006から入力されている16ビット乱数値がラッチされる（（2）参照）。ラッチ処理では、ラッチされた16ビット乱数値の下位8ビットがソフトラッチ乱数値レジスタ（下位）に取り込まれ（矢印（3）参照）、次いで、上位8ビットがソフトラッチ乱数値レジスタ（上位）に取り込まれる（矢印（4）参照）。乱数値がソフトラッチ乱数値レジスタに取り込まれると、乱数ソフトラッチフラグレジスタの対応ビットに「1」がセットされる（矢印（5）参照）。次いで、乱数ソフトラッチフラグレジスタの対応ビットに「1」がセットされたら、ユーザプログラム内の第一の命令によりソフトラッチ乱数値レジスタ（下位）の値をRAM308内の所定アドレスの記憶領域に当否判定乱数（下位）として記憶し（矢印（

40

50

6) 参照)、次いで第二の命令によりソフトラッチ乱数値レジスタ(上位)の値をRAM 308内の所定アドレスの次のアドレスの記憶領域に当否判定乱数(上位)として記憶する(矢印(7)参照)。なお、ソフトラッチ乱数値レジスタの値をRAM 308内の所定アドレスの記憶領域に記憶したら乱数ソフトラッチフラグレジスタの対応ビットに「0」がセットされる(矢印(8)参照)。

#### 【1808】

図246も、乱数生成回路9006から16ビット乱数値をRAM 308に取込む手順を示している。図246内の構成要素の配置は図245と同様なのでその説明は省略する。本例では乱数値の最大値が100に設定されている。このため、16ビット乱数値が順次更新されて0~100(十進数)まで更新される。

10

#### 【1809】

図246に示す乱数値取込みでは、図245の場合と同様に、ユーザプログラムにより内部レジスタの乱数ソフトラッチレジスタに対し「1」が書き込まれる(矢印(1)参照)。これにより、所定のソフトラッチ乱数値レジスタにラッチ信号が出力されて、乱数生成回路9006から入力されている16ビット乱数値がラッチされる((2)参照)。ラッチ処理では、ラッチされた16ビット乱数値の下位8ビットがソフトラッチ乱数値レジスタ(下位)に取り込まれ(矢印(3)参照)、次いで、上位8ビットがソフトラッチ乱数値レジスタ(上位)に取り込まれる(矢印(4)参照)。乱数値がソフトラッチ乱数値レジスタに取り込まれると、乱数ソフトラッチフラグレジスタの対応ビットに「1」がセットされる(矢印(5)参照)。次いで、乱数ソフトラッチフラグレジスタの対応ビットに「1」がセットされたことを条件に、ユーザプログラム内の第一の命令によりソフトラッチ乱数値レジスタ(下位)の値をRAM 308内の所定アドレスの記憶領域に当否判定乱数(下位)として記憶し(矢印(6)参照)、次いで第二の命令によりソフトラッチ乱数値レジスタ(上位)の値をRAM 308内の所定アドレスの次のアドレスの記憶領域に当否判定乱数(上位)として記憶する(矢印(7)参照)。なお、ソフトラッチ乱数値レジスタの値をRAM 308内の所定アドレスの記憶領域に記憶したら乱数ソフトラッチフラグレジスタの対応ビットに「0」がセットされる(矢印(8)参照)。このように、乱数値の最大値が256未満で上位バイトが常に0であるためデータ取込みが不要な場合であっても、上位8ビットも一緒に取り込むようにユーザプログラムを組んでいる。

20

#### 【1810】

図247(a)は、図245および図246に示す乱数取込み(取得)手順において実行される乱数取得判定処理を示すフローチャートである。この処理は、例えば、主制御部タイマ割込処理の入賞受付処理(ステップS217)で実行される。同一の割込み内で特図1乱数値や、特図2乱数値、普図乱数値など複数ラッチする場合は、入賞受付処理(ステップS217)内でこの乱数取得判定処理を複数回繰り返す。特図1始動口230や、特図2始動口232、普図始動口228の球検出センサから読み取ったデータが入賞判定パターンに一致する場合に、本処理に移行する。1つの割込み内で例えば特図1乱数値と特図2乱数値をラッチする場合において、同時成立の場合は例えば特図2乱数値を先にラッチして、次の割込みで特図1乱数値をラッチするようにしてもよい。また、同時成立の場合は、1割込み内で特図1乱数値と特図2乱数値とをラッチするようにしてもよい。つまり、同時成立の場合に特図1乱数値と特図2乱数値とを別割込みでラッチすることはないようにしてもよい。

30

40

#### 【1811】

乱数取得判定処理において、主制御部300は、まず、ラッチ条件が成立しているか否かを判断する(ステップS9001)。ラッチ条件としては、対象(特図1、特図2、あるいは普図)を特定し、特定した対象の現在の保留個数を取得し、該保留個数が最大でないかを判定することが挙げられる。保留個数が最大であればラッチ条件が成立せず、保留個数が最大未満であればラッチ条件が成立する。保留個数以外にも他のラッチ条件が含まれてもよい。他のラッチ条件として、例えば、エラー状態であるか否かをラッチ条件の一部に含ませてもよい。さらに、他のラッチ条件として、遊技機における常識的な条件が全

50

て含まれるようにしてももちろんよい。

#### 【1812】

主制御部300は、ステップS9001でラッチ条件が成立すると判断したら、ステップS9002に移行して、所定の乱数値をラッチする。ラッチ条件が不成立であると判断したらステップS9002を省略して乱数取得判定処理を終了する。

#### 【1813】

図247(b)は、図245および図246に示す乱数取込み(取得)手順において実行される乱数取得処理を示すフローチャートである。この処理は、例えば、主制御部タイマ割込処理の入賞受付処理(ステップS217)で実行される。同一割込み処理内で複数ラッチする場合(特図1乱数値や、特図2乱数値、普図乱数値等)は、ステップS217内でこの乱数取得処理を所定回数繰り返す。

10

#### 【1814】

乱数取得処理において、主制御部300は、まず、乱数取得条件が成立しているか否かを判断する(ステップS9003)。乱数取得条件としては、対象(特図1、特図2、あるいは普図)を特定し、特定した対象に対応する乱数ソフトラッチフラグレジスタの所定ビットに「1」がセットされているか否かを判定する。

#### 【1815】

主制御部300は、ステップS9003で乱数取得条件が成立すると判断したら、ステップS9004に移行して、所定の乱数値を取得する。乱数取得条件が不成立であると判断したらステップS9004を省略して乱数取得処理を終了する。ステップS9004では、内蔵レジスタからの乱数取得とともに、このタイミングでソフト乱数を取得するようにしてもよい。そして、内蔵レジスタから取得した乱数にソフト乱数を加算して当否判定用乱数を生成するようにしてもよい。

20

#### 【1816】

図248(a)は、図245および図246に示す乱数取込み(取得)手順において実行される図247(a)に示す乱数取得判定処理をより詳細に説明するフローチャートである。主制御部300は、まず、特図1始動口230や、特図2始動口232、普図始動口228の球検出センサから読み取ったデータが入賞判定パターンに一致するか否かを判定する(ステップS9005)。球検出センサから読み取ったデータが入賞判定パターンに一致する場合はステップS9006に移行し、一致しない場合は乱数取得判定処理を終了する。球検出センサから読み取ったデータが入賞判定パターンに一致する場合は、主制御部300は、ラッチ条件を満たしているか否かを判断するため、保留個数が非満タン状態であるか否かを判断する(ステップS9006)。保留個数が非満タン状態である場合はステップS9007に移行し、満タン状態である場合は乱数取得判定処理を終了する。

30

#### 【1817】

主制御部300は、保留個数が非満タン状態である場合はステップS9007に移行し、乱数ソフトラッチレジスタの所定ビットに「1」をセットする。これにより、図245または図246に示す矢印(1)で示す処理が実行され、所定のソフトラッチ乱数値レジスタにラッチ信号が出力される。

#### 【1818】

次いで、ステップS9008において、乱数ソフトラッチフラグレジスタの所定ビットが「0」であるか否かが判断され、乱数ソフトラッチフラグレジスタの所定ビットが「0」であればステップS9008に移行し、所定ビットが「0」でなければ乱数取得判定処理を終了する。

40

#### 【1819】

乱数ソフトラッチフラグレジスタの所定ビットが「0」であれば、ステップS9010において、乱数生成回路9006から入力されている16ビット乱数値が所定のソフトラッチ乱数値レジスタにラッチされる(図245または図246に示す矢印(3)および(4)参照)。

#### 【1820】

50

次いで、ステップ S 9 0 1 1 で乱数ソフトラッチフラグレジスタの所定ビットに「1」がセットされて（図 2 4 5 または図 2 4 6 に示す矢印（5）参照）、乱数取得判定処理が終了する。

#### 【1 8 2 1】

図 2 4 8（b）は、図 2 4 5 および図 2 4 6 に示す乱数取込み（取得）手順において実行される図 2 4 7（b）に示す乱数取得処理をより詳細に説明するフローチャートである。主制御部 3 0 0 は、乱数ソフトラッチフラグレジスタの所定ビットが「1」であるか否かを判断する（ステップ S 9 0 1 2）。所定ビットが「1」であればステップ S 9 0 1 3 に移行し、所定ビットが「1」でなければ乱数取得処理を終了する。

#### 【1 8 2 2】

乱数ソフトラッチフラグレジスタの所定ビットが「1」であれば、ステップ S 9 0 1 3 において、乱数ソフトラッチ乱数値レジスタ（下位）から 1 6 ビット乱数値の下位バイトを R A M 3 0 8 内の所定アドレスの記憶領域に当否判定乱数（下位）としてリード（読み込み）する（図 2 4 5 または図 2 4 6 に示す矢印（6）参照）。次いで、ステップ S 9 0 1 4 において、乱数ソフトラッチ乱数値レジスタ（上位）から 1 6 ビット乱数値の上位バイトを R A M 3 0 8 内の所定アドレスの次のアドレスの記憶領域に当否判定乱数（上位）としてリードする（図 2 4 5 または図 2 4 6 に示す矢印（7）参照）。次いで、ステップ S 9 0 1 5 において、1 6 ビット乱数値の上位バイトを R A M 3 0 8 内の所定アドレスの次のアドレスの記憶領域にリードしたことに基づいて、乱数ソフトラッチフラグレジスタの対応ビットに「0」がセットされる（図 2 4 5 または図 2 4 6 に示す矢印（8）参照）。

#### 【1 8 2 3】

図 2 4 9 は、所定のソフトラッチ乱数値レジスタに乱数生成回路 9 0 0 6 から逐次入力される 1 6 ビット乱数値を直接取込む方法を示している。図 2 4 5 に示す乱数値取込みでは、ユーザプログラムに記述した第三の命令によりソフトラッチ乱数値レジスタから 1 6 ビット乱数値を直接取込むことができる。第三の命令を用いることにより、マイコン内蔵レジスタ内の乱数値取込みに関連するレジスタのうち、乱数ソフトラッチレジスタと乱数ソフトラッチフラグレジスタは不必要となる。

#### 【1 8 2 4】

第三の命令だけで、乱数ソフトラッチ乱数値レジスタ（下位）から 1 6 ビット乱数値の下位バイトを R A M 3 0 8 内の所定アドレスの記憶領域に当否判定乱数（下位）として読み込み、続いて、乱数ソフトラッチ乱数値レジスタ（上位）から 1 6 ビット乱数値の上位バイトを R A M 3 0 8 内の所定アドレスの次のアドレスの記憶領域に当否判定乱数（上位）として読み込むことができる（矢印（A）参照）。後程プログラムコードを用いて説明するが、第三の命令で下位のレジスタを指定するだけで上位のレジスタに格納されている値も取り込むことができる。機械語からは上位アドレスの判別が不能になるため、不正者によるコード解析を困難にすることができる。

#### 【1 8 2 5】

図 2 4 5 や図 2 4 6 に示す乱数値取込み方法では、1 6 ビット乱数値の上位バイトを R A M 3 0 8 内の所定アドレスの次のアドレスの記憶領域に読み込んだことに基づいて、乱数ソフトラッチフラグレジスタの対応ビットに「0」がセットされるが、本例の直接取込み方法では、乱数ソフトラッチフラグレジスタの対応ビットは元々「0」であり、本処理の後にも「0」が維持されている。このため、第三の命令の実行後に同一のソフトラッチ乱数値レジスタから第一の命令と第二の命令を使って乱数値を取り出す場合にも、図 2 4 8（a）の乱数取得判定処理のステップ S 9 0 0 8 をそのまま適用できる。

#### 【1 8 2 6】

図 2 5 0 は、図 2 4 9 に示す乱数直接取込みにおいて実行される乱数取得判定／取得処理を示すフローチャートである。この処理は、例えば、主制御部タイマ割込処理の入賞受付処理（ステップ S 2 1 7）で実行される。同一の割込み内で特図 1 乱数値や、特図 2 乱数値、普図乱数値など複数の乱数値を直接取得する場合は、入賞受付処理（ステップ S 2

10

20

30

40

50

17) 内でこの乱数取得判定／取得処理を複数回繰り返す。特図1始動口230や、特図2始動口232、普図始動口228の球検出センサから読み取ったデータが入賞判定パターンに一致する場合に、本処理に移行する。1つの割込み内で例えば特図1乱数値と特図2乱数値を直接取得する場合において、同時成立の場合は例えば特図2乱数値を先に取得して、次の割込みで特図1乱数値を取得するようにしてもよい。また、同時成立の場合は、1割込み内で特図1乱数値と特図2乱数値とを直接取得するようにしてもよい。つまり、同時成立の場合に特図1乱数値と特図2乱数値とを別割込みで直接取得することはないようにしてもよい。

#### 【1827】

乱数取得判定／取得処理において、主制御部300は、まず、乱数取得条件が成立しているか否かを判断する(ステップS9016)。乱数取得条件としては、対象(特図1、特図2、あるいは普図)を特定し、特定した対象の現在の保留個数を取得し、該保留個数が最大でないかを判定することが挙げられる。保留個数が最大であれば乱数取得条件が成立せず、保留個数が最大未満であれば乱数取得条件が成立する。保留個数以外にも他の乱数取得条件が含まれてもよい。他の乱数取得条件として、例えば、エラー状態であるか否かを乱数取得条件の一部に含ませてもよい。さらに、他の乱数取得条件として、遊技機における常識的な条件が全て含まれるようにしてももちろんよい。

#### 【1828】

主制御部300は、ステップS9016で乱数取得条件が成立すると判断したら、ステップS9017に移行して、所定の乱数値を直接取得する。ステップS9017では、内蔵レジスタからの乱数直接取得とともに、このタイミングでソフト乱数を取得するようにしてもよい。そして、内蔵レジスタから取得した乱数にソフト乱数を加算して当否判定用乱数を生成するようにしてもよい。乱数取得条件が不成立であると判断したらステップS9017を省略して乱数取得判定／取得処理を終了する。

#### 【1829】

この乱数取得判定／取得処理においては、第一の命令と第二の命令とを用いる乱数取得の場合に必須であった、特定した対象に対応する乱数ソフトラッチフラグレジスタの所定ビットに「1」がセットされているか否かを判定する処理が不要となる。

#### 【1830】

図251(a)、(b)は、図249に示す乱数直接取込み(取得)手順において実行される図250に示す乱数取得判定／取得処理をより詳細に説明するフローチャートである。図251(a)は乱数取得判定処理を示している。主制御部300は、まず、特図1始動口230や、特図2始動口232、普図始動口228の球検出センサから読み取ったデータが入賞判定パターンに一致するか否かを判断する(ステップS9018)。球検出センサから読み取ったデータが入賞判定パターンに一致する場合はステップS9019に移行し、一致しない場合は乱数取得判定処理を終了する。球検出センサから読み取ったデータが入賞判定パターンに一致する場合は、主制御部300は、乱数取得条件を満たしているか否かを判断するため、保留個数が非満タン状態であるか否かを判断する(ステップS9019)。保留個数が非満タン状態である場合はステップS9020に移行し、満タン状態である場合は乱数取得判定処理を終了する。

#### 【1831】

主制御部300は、保留個数が非満タン状態である場合は乱数取得条件を満たしているものとしてステップS9020に移行し、所定のフラグをオンに設定する。所定のフラグは、マイコン内蔵のレジスタではなく、ユーザプログラムによるRAM308内の所定領域に設定している。

#### 【1832】

図251(b)は、乱数取得処理を示している。主制御部300は、所定のフラグがオンか否かを判断する(ステップS9021)。所定のフラグがオンであればステップS9022に移行し、所定のフラグがオンでなければ乱数取得処理を終了する。

#### 【1833】

10

20

30

40

50



所定のフラグがオンであれば、ステップ S 9 0 2 2 において、ソフトラッチ乱数値レジスタから乱数値をリードする。具体的には、乱数ソフトラッチ乱数値レジスタ（下位）から 1 6 ビット乱数値の下位バイトを R A M 3 0 8 内の所定アドレスの記憶領域に当否判定乱数（下位）としてリードし、続いて乱数ソフトラッチ乱数値レジスタ（上位）から 1 6 ビット乱数値の上位バイトを R A M 3 0 8 内の所定アドレスの次のアドレスの記憶領域に当否判定乱数（上位）としてリードする（図 2 4 9 の矢印 A 参照）。次いで、ステップ S 9 0 2 3 において、1 6 ビット乱数値の上位バイトを R A M 3 0 8 内の所定アドレスの次のアドレスの記憶領域にリードしたことに基づいて、所定のフラグをオフに設定する。

#### 【1 8 3 4】

図 2 5 2 は、図 2 4 5 や図 2 4 6 に示した乱数値取込みの手順を記述したプログラムコードの一例を示している。ラッチ条件（例えば、保留非満タンを少なくとも含む条件）が成立すると、コード中の R O M エリアに記述されているように、乱数ソフトラッチレジスタに「1」をセットする。また、別モジュールで乱数ソフトラッチフラグレジスタの値を取得して、取得した値を「1」と比較し、取得した値が「1」であれば、モジュール「m o R N D G E T」にジャンプする。モジュール「m o R N D G E T」では、ソフトラッチ乱数値レジスタ（下位）のアドレスを取得し、取得したアドレスに格納されたソフトラッチ乱数値レジスタ（下位）の値を取得し、取得したソフトラッチ乱数値レジスタ（下位）の値を R A M に読み込む。次いで、ソフトラッチ乱数値レジスタ（上位）のアドレスを取得し、取得したアドレスに格納されたソフトラッチ乱数値レジスタ（上位）の値を取得し、取得したソフトラッチ乱数値レジスタ（上位）の値を R A M に読み込む。

#### 【1 8 3 5】

図 2 5 3 は、図 2 4 9 に示した乱数値直接取込みの手順を記述したプログラムコードの一例を示している。図中取消線で示すコードは図 2 5 2 に比して不要となったコードを示している。乱数取得条件（例えば、保留非満タンを少なくとも含む条件）が成立すると、コード中の R O M エリアに記述されているモジュール「m o R N D G E T」にジャンプする。モジュール「m o R N D G E T」では、L D W 命令（第三の命令）により、ラッチ処理をせずに、また、ペアレジスタ（H L）を介さずに一命令で R A M 3 0 8 に 1 6 ビット乱数値を読み込みことができる。なお、図 2 4 9 および図 2 5 3 に示す例において、直接 R A M 3 0 8 に乱数値を取り込むのではなく一旦ペアレジスタをに乱数値を取り込んで、ペアレジスタを介して R A M 3 0 8 に乱数値を取り込むようにしてもよい。

#### 【1 8 3 6】

（L D W （R A N S U 0 \_\_ L）, R A N S U \_\_ R E G 0 \_\_ L）命令は、R A N S U \_\_ R E G 0 \_\_ L の値を R A N S U 0 \_\_ L で示されるアドレスの記憶領域にロードし、R A N S U \_\_ R E G 0 \_\_ H の値を R A N S U 0 \_\_ L で示されるアドレスの次のアドレスである R A N S U 0 \_\_ H で示されるアドレスの記憶領域にロードする命令である。

#### 【1 8 3 7】

このように、第三の命令を用いた乱数値直接取込みによれば、第一の命令及び第二の命令を用いた乱数値取込みに比してプログラムが簡素になり作成が容易になる。さらに、図 2 5 4 を用いて第三の命令を用いることの利点について説明する。

#### 【1 8 3 8】

図 2 5 4 （a）および（b）は、第一の命令および第二の命令を用いた場合の処理の流れの一部を示している。図 2 5 4 （c）および（d）は、第三の命令を用いた場合の処理の流れの一部を示している。図 2 5 4 （a）および（b）は、第一の命令および第二の命令を用いた場合の処理の流れを示している。

#### 【1 8 3 9】

第一の命令および第二の命令を用いた乱数値取得の流れは、乱数ソフトラッチレジスタに「1」をセットし、ソフトラッチ乱数値レジスタに乱数を取り込まれ、乱数ソフトラッチフラグレジスタにフラグがセットされ、該フラグを参照し、フラグが立っている場合に R A M に乱数を取得し、該フラグをクリアする。図 2 5 4 （a）に示す例では、リード許可条件が成立してリードを許可する処理において、ラッチ信号およびラッチ完了信号（セ

ット信号)が出力される。次いで、乱数リード処理において、リード完了信号(クリア信号)が出力される。この場合、ラッチ信号の出力からラッチ完了信号あるいはリード完了信号のタイミングで乱数取得処理の実行時期が推測され易い。

#### 【1840】

これに対し、第三の命令を用いた乱数値取得では、ユーザプログラム上のフラグを用いており、乱数ソフトラッチレジスタと乱数ソフトラッチフラグレジスタを用いないため、これらに値をセットする処理は行われない。このため、図254(c)に示すように、リード許可条件が成立してリードを許可する処理において、RAM内に設定したフラグが変化するだけでラッチ信号およびラッチ完了信号は出力されず、乱数リード処理において、リード完了信号(クリア信号)だけが出力される。このように、ラッチ信号が出力されず、リード完了信号だけが出力されるので乱数取得処理の実行時期を推測するのが困難にすることができる。

10

#### 【1841】

図254(b)に示す例も第一の命令および第二の命令を用いた乱数値取得である。リード許可条件が成立してリードを許可する処理において、ラッチ信号およびラッチ完了信号(セット信号)が出力される。次いで、1または複数のプログラムモジュールによる処理を経て、乱数リード処理が行われる。乱数リード処理において、リード完了信号が出力される。この場合、ラッチ信号やラッチ完了信号の出力からリード完了信号の出力までの間隔を長くすることができるため、乱数取得処理の実行時期の推測を困難にすることができる場合がある。

20

#### 【1842】

図254(d)に示す例は第三の命令を用いた乱数値取得である。リード許可条件が成立してリードを許可する処理において、ラッチ信号およびラッチ完了信号は出力されない。次いで、1または複数のプログラムモジュールによる処理を経て、乱数リード処理が行われる。乱数リード処理において、リード完了信号が出力される。この場合も、ラッチ信号が出力されず、リード完了信号だけが出力され、さらにリードを許可する処理からリード完了信号の出力までの間隔を長くすることができるため、乱数取得処理の実行時期を推測するのが困難にすることができる。また、リードを許可する処理では、特図1、特図2、あるいは普図の1または複数の乱数値の取得を許可することができる。また、他の乱数(例えば、ソフト乱数)を当該処理で取得するようにしてもよい。また、乱数リード処理では、特図1、特図2、あるいは普図の1または複数の乱数値を取得することができる。他の乱数(例えば、ソフト乱数)を当該処理で取得するようにしてもよい。

30

#### 【1843】

<主制御部のデータテーブル>

#### 【1844】

次に、図255を用いて、パチンコ機100の主制御部300のROM306が記憶しているデータテーブルについて説明する。なお、同図(a)は大当たり判定テーブルの一例を、同図(b)は高確率状態移行判定テーブルの一例を、同図(c)はタイマ番号決定テーブルの一例を、それぞれ示した図である。

#### 【1845】

図255(a)に示す大当たり判定テーブルは、RAM308に設けた遊技状態格納領域に記憶している情報(以下、単に遊技情報と称する。)の種類と、第1特図始動口230に球が入賞したことを所定の球検出センサが検出した場合に使用する抽選データ(第1特図始動口用抽選データ)と、第2特図始動口232に球が入賞したことを所定の球検出センサが検出した場合に使用する抽選データ(第2特図始動口用抽選データ)と、を対応付けて記憶したデータテーブルである。主制御部300の基本回路302は、これらの情報を用いて特図変動遊技を当選(大当たり)とするか、不当選(外れ)とするかの決定、すなわち大当たり判定を行う。なお、遊技情報は、特図変動遊技を所定の低確率で当選と判定する低確率状態を示す情報、および低確率よりも高い高確率で特図変動遊技を当選と判定する高確率状態を示す情報などを含むが、以下、これらを単に低確率状態および高確率状態

40

50

と称する。また、遊技状態格納領域に記憶する情報には別の情報もあるが、これらの情報については後述する。

#### 【1846】

第1特図始動口用抽選データは、第1特図始動口230に球が入賞したことを所定の球検出センサが検出した場合に開始する特図変動遊技の結果を決定するために使用する抽選データである。例えば、遊技状態が低確率状態の場合、取得した特図当選乱数値（乱数値については後述する）が10001～10187であるときは、特図変動遊技の当選と判定する。一方、取得した特図当選乱数値が10001～10187以外の数値である場合には、特図変動遊技の外れと判定する。なお、本実施例では、特図当選乱数値の取り得る数値範囲は0～65535（数値範囲の大きさは65536）、低確率状態における第1特図始動口用抽選データが示す数値範囲は10001～10187（数値範囲の大きさは187）であるから、低確率状態の第1特図始動口230への球の入賞に基づく特図変動遊技の当選確率は、約 $1/350.4$ （ $=187/65536$ ）である。これに対して、高確率状態における第1特図始動口用抽選データが示す数値範囲は30001～31310（数値範囲の大きさは1310）であるから、高確率状態の第1特図始動口230への球の入賞に基づく特図変動遊技の当選確率は約 $1/50.0$ （ $=1310/65536$ ）であり、第1特図始動口230への球の入賞に基づく特図変動遊技の当選確率は、低確率状態よりも高確率状態の方が高くなるように設定している。

#### 【1847】

第2特図始動口用抽選データは、第2特図始動口232に球が入賞したことを所定の球検出センサが検出した場合に開始する特図変動遊技の結果を決定するために使用する抽選データである。例えば、遊技状態が低確率状態の場合、取得した特図当選乱数値が20001～20187であるときには、特図変動遊技の当選と判定する。一方、取得した特図当選乱数値が20001～20187以外の数値である場合には、特図変動遊技の外れと判定する。なお、本実施例では、特図当選乱数値の取り得る数値範囲は0～65535（数値範囲の大きさは65536）、低確率状態における第2特図始動口用抽選データが示す数値範囲は20001～20187（数値範囲の大きさは187）であるから、低確率状態の第2特図始動口128への球の入賞に基づく特図変動遊技の当選確率は、約 $1/350.4$ （ $=187/65536$ ）である。これに対して、高確率状態における第2特図始動口用抽選データが示す数値範囲は40001～41310（数値範囲の大きさは1310）であるから、高確率状態の第2特図始動口232への球の入賞に基づく特図変動遊技の当選確率は約 $1/50.0$ （ $=1310/65536$ ）であり、第2特図始動口232への球の入賞に基づく特図変動遊技の当選確率は、低確率状態よりも高確率状態の方が高くなるように設定している。

#### 【1848】

図255（b）に示す高確率状態移行判定テーブルは、上述の大当たり判定の結果、大当たりと判定した場合に使用する抽選データを記憶したデータテーブルである。主制御部300の基本回路302は、これらの情報を用いて特図変動遊技の終了後に大当たり遊技を開始するか、または特別大当たり遊技を開始するかの判定、すなわち確変移行判定を行う。例えば、取得した特図乱数値（乱数値については後述する）が11～74の数値である場合には、特図変動遊技の終了後に特別大当たり遊技を開始する。一方、取得した特図乱数値が11～74の数値以外である場合には、特図変動遊技の終了後に大当たり遊技を開始する。なお、本実施例では、特図乱数値の取り得る数値範囲は0～127（数値範囲の大きさは128）、抽選データの移行判定乱数の範囲は11～74（数値範囲の大きさは64）であるから、大当たり判定の結果が当選である場合に確変移行判定の結果を当選にする確率、すなわち特別大当たりを開始する確率は $1/2$ （ $=64/128$ ）である。

#### 【1849】

図255（c）に示すタイマ番号決定テーブルは、特図表示装置212、214による特図の変動表示を開始してから停止表示をするまでの変動時間を示すタイマ番号を決定するための抽選に使用する抽選データを記憶したデータテーブルである。主制御部300の

基本回路 302 は、これらの情報と、上述の大当たり判定結果（後述する大当たりフラグの値）および後述するタイマ乱数の値に基づいて、タイマ番号を決定する。そして、決定したタイマ番号に対応する変動時間を、特図表示図柄更新タイマ（図 256（c）参照）の初期値として設定する。

#### 【1850】

なお、本実施例では、特図タイマ乱数値（乱数値については後述する）の取り得る数値範囲は 0～65535（数値範囲の大きさは 65536）、上述の大当たり判定結果が不当選の場合は、タイマ 1 のタイマ乱数の範囲は 0～60235（数値範囲の大きさは 60236）であるから、タイマ番号としてタイマ 1（変動時間 5 秒）を選択する確率は  $60236 / 65536$  である。また、タイマ番号として、タイマ 2（変動時間 10 秒）を選択する確率は  $4250 / 65536$ 、タイマ 3（変動時間 20 秒）を選択する確率は  $800 / 65536$ 、タイマ 3（変動時間 40 秒）を選択する確率は  $250 / 65536$  である。一方、大当たり判定結果が当選の場合は、タイマ 2 のタイマ乱数の範囲は 0～15535（数値範囲の大きさは 15536）であるから、タイマ番号としてタイマ 2（変動時間 10 秒）を選択する確率は  $15535 / 65536$  である。また、タイマ番号として、タイマ 3（変動時間 20 秒）を選択する確率は  $9000 / 65536$ 、タイマ 4（変動時間 40 秒）を選択する確率は  $38000 / 65536$ 、タイマ 5（変動時間 50 秒）を選択する確率は  $3000 / 65536$  である。

#### 【1851】

次に、図 256（a）～（d）を用いて、パチンコ機 100 の主制御部 300 の ROM 306 が記憶しているデータテーブルについて説明する。なお、同図（a）は払出要求数テーブルの一例を、同図（b）は払出加工テーブルの一例を、同図（c）はタイマ制御データテーブルの一例を、同図（d）はジャンプテーブルの一例をそれぞれ示した図である。

#### 【1852】

図 256（a）に示す払出要求数テーブルは、パチンコ機 100 が賞球として遊技者に払い出す賞球数を記憶したデータテーブルである。詳細は後述するが、主制御部 300 は、この払出要求数テーブルを用いて払出制御部 600 に出力する払出要求数を選択し、払出要求数に加工を施した後、払出制御部 600 に出力する。例えば、第 1 特図始動口 230 への入球を所定の球検出センサが検出した場合には、主制御部 300 は、払出要求数として 3 を選択し、払出要求数に加工を施した後、払出制御部 600 に出力する。第 2 特図始動口 232、一般入賞口 226、または可変入賞口 234 への入球を所定の球検出センサが検出した場合も同様にして、主制御部 300 は、払出制御部 600 に払出要求数として 4、10、または 15 をそれぞれ選択し、払出要求数に加工を施した後、払出制御部 600 に出力する。なお、詳細は後述するが、払出要求数を入力した払出制御部 600 は、主制御部 300 が加工を施した払出要求数の加工を解除して払出要求数を取得した後、払出装置 152 を駆動し、本実施例では、第 1 特図始動口 230 への入球では 3 個の球を、第 2 特図始動口 232 への入球では 4 個の球を、一般入賞口 226 への入球では 10 個の球を、可変入賞口 234 への入球では 15 個の球を、それぞれ賞球として貯留皿 126 に排出する。

#### 【1853】

図 256（b）に示す払出加工テーブルは、主制御部 300 が払出制御部 600 に出力する払出要求数を加工するために用いるデータテーブルである。詳細は後述するが、主制御部 300 は、この払出加工テーブルを用いて払出制御部 600 に出力する払出要求数を加工する。例えば、加工種別が 0 の場合には、主制御部 300 は、払出要求数と、加工データの 00H（数値の後ろの H は数値が 16 進数であることを示す。以下同じ）を用いて、払出要求数に加工を施す。加工種別が 1、2、または 3 の場合も同様にして、主制御部 300 は、払出要求数と、加工データの 13H、2EH、または 39H を用いて、払出要求数に加工を施す。なお、加工データのビット 0～4 はマスクの情報を示し、ビット 5、6 は加工種別の情報を示している。

## 【1854】

図256(c)に示すタイマ制御データテーブルは、主制御部300が各種のタイマを更新するために用いるデータテーブルである。詳細は後述するが、主制御部300は、このタイマ制御データテーブルを用いて、更新周期の異なる複数種類のタイマ（この例では、約2ms毎に更新するタイマ1～9の9種類のタイマと、約10ms毎に更新するタイマ10の1種類のタイマ）を更新する。

## 【1855】

図256(d)に示すジャンプテーブルは、主制御部300のコマンド設定送信処理（詳細は後述）における複数種類の送信処理を識別するためにRAM308に設けた送信情報記憶領域の送信情報に含んでいる一般情報と、コマンド設定送信処理における各送信処理の先頭アドレスを記憶したデータテーブルである。詳細は後述するが、主制御部300は、このジャンプテーブルを用いて、一般情報からジャンプ先を選択し、選択したジャンプ先の処理を実行する。例えば、一般情報が01Hの場合には、主制御部300は、ジャンプ先として一般コマンド回転開始設定送信処理を選択し、この一般コマンド回転開始設定送信処理を実行する。他の一般情報も同様にして、一般情報が02Hの場合には一般コマンド回転停止設定送信処理を選択・実行し、一般情報が04Hの場合には一般コマンド入賞演出設定送信処理を選択・実行し、一般情報が08Hの場合には一般コマンド終了演出設定送信処理を選択・実行し、一般情報が10Hの場合には一般コマンド大入賞口開放設定送信処理を選択・実行し、一般情報が20Hの場合には一般コマンド大入賞口閉鎖設定送信処理を選択・実行する。これら処理の実行後に、送信情報記憶領域の送信情報を初期化し、ステップS9317の外部出力信号設定処理に処理を移行させる。

## 【1856】

<払出制御部のデータテーブル>

## 【1857】

次に、図256(e)を用いて、パチンコ機100の払出制御部600のROMが記憶しているデータテーブルについて説明する。なお、同図(e)は復元データテーブルの一例を示した図である。

## 【1858】

図256(e)に示す復元データテーブルは払出制御部600が主制御部300から入力した、加工を施した払出要求数の加工を解除するために用いるデータテーブルである。詳細は後述するが、払出制御部600は、この復元データテーブルを用いて主制御部300から入力した、加工を施した払出要求数の加工を解除する。例えば、加工種別が0の場合には、払出制御部600は、主制御部300から入力した払出要求数と、加工データの00Hを用いて、払出要求数の加工を解除する。加工種別が1、2、または3の場合も同様にして、払出制御部600は、主制御部300から入力した払出要求数と、加工データの03H、0EH、または09Hを用いて、払出要求数の加工を解除する。

## 【1859】

<主制御部リセット割り込み処理>

## 【1860】

次に、図257を用いて、主制御部300のCPU304が実行する主制御部リセット割り込み処理について説明する。なお、同図は主制御部リセット割り込み処理の流れを示すフローチャートである。

## 【1861】

上述したように、主制御部300には、電源が投入されると起動信号（リセット信号）を出力する起動信号出力回路（リセット信号出力回路）340を設けている。この起動信号を入力した基本回路302のCPU304は、リセット割り込みによりリセットスタートしてROM306に予め記憶している制御プログラムに従って処理を実行する。

## 【1862】

ステップS9101では、初期設定1を行う。この初期設定1では、CPU304のスタックポインタ（SP）へのスタック初期値の設定、割り込みマスクの設定、I/Oポー

10

20

30

40

50

ト 3 1 0 の初期設定、R A M 3 0 8 に記憶する各種変数の初期設定、W D T 3 1 4 への初期値の設定等を行う。なお、本実施例では、W D T 3 1 4 に、初期値として 3 2 . 8 m s に相当する数値を設定する。

#### 【1 8 6 3】

ステップ S 9 1 0 2 では、W D T 3 1 4 のリセットを行い、W D T 3 1 4 による時間計測を再始動する。

#### 【1 8 6 4】

ステップ S 9 1 0 3 では、低電圧信号がオンであるか否か、すなわち、電圧監視回路 3 3 8 が、電源制御部 6 6 0 から主制御部 3 0 0 に供給している電源の電圧値が所定の値（本実施例では 9 V）未満である場合に電圧が低下したことを示す低電圧信号を出力しているか否かを監視する。そして、低電圧信号がオンの場合（C P U 3 0 4 が電源の遮断を検知した場合）にはステップ S 9 1 0 2 に戻り、低電圧信号がオフの場合（C P U 3 0 4 が電源の遮断を検知していない場合）にはステップ S 9 1 0 4 に進む。

#### 【1 8 6 5】

ステップ S 9 1 0 4 では、初期設定 2 を行う。この初期設定 2 では、後述する主制御部 タイマ割り込み処理を定期毎に実行するための周期を決める数値をカウンタ・タイマ 3 1 2 に設定する処理、I / O 3 1 0 の所定のポート（例えば試験用出力ポート、演出制御部への出力ポート）からクリア信号を出力する処理、R A M 3 0 8 への書き込みを許可する設定等を行う。

#### 【1 8 6 6】

ステップ S 9 1 0 5 では、電源の遮断前（電断前）の状態に復帰するか否かの判定を行い、電断前の状態に復帰しない場合（主制御部 3 0 0 の基本回路 3 0 2 を初期状態にする場合）にはステップ S 9 1 0 8 に進む。同様に電源ステータスの情報が「サスペンド」以外の情報を示している場合にもステップ S 9 1 0 8 に進む。

#### 【1 8 6 7】

具体的には、最初に、電源基板に設けた操作部を遊技店の店員などが操作した場合に送信される R A M クリア信号がオン（操作があったことを示す）であるか否か、すなわち R A M クリアが必要であるか否かを判定し、R A M クリア信号がオンの場合（R A M クリアが必要な場合）には、基本回路 3 0 2 を初期状態にすべくステップ S 9 1 0 8 に進む。一方、R A M クリア信号がオフの場合（R A M クリアが必要でない場合）は、R A M 3 0 8 に設けた電源ステータス記憶領域に記憶した電源ステータスの情報を読み出し、この電源ステータスの情報がサスペンドを示す情報であるか否かを判定する。そして、電源ステータスの情報がサスペンドを示す情報でない場合には、基本回路 3 0 2 を初期状態にすべくステップ S 9 1 0 8 に進み、電源ステータスの情報がサスペンドを示す情報である場合には、R A M 3 0 8 の所定の領域（例えば全ての領域）に記憶している 1 バイトデータを初期値が 0 である 1 バイト構成のレジスタに全て加算することによりチェックサムを算出し、算出したチェックサムの結果が特定の値（例えば 0）であるか否か（チェックサムの結果が正常であるか否か）を判定する。そして、チェックサムの結果が特定の値（例えば 0）の場合（チェックサムの結果が正常である場合）には電断前の状態に復帰すべくステップ S 9 1 0 6 に進み、チェックサムの結果が特定の値（例えば 0）以外である場合（チェックサムの結果が異常である場合）には、パチンコ機 1 0 0 を初期状態にすべくステップ S 9 1 0 8 に進む。同様に電源ステータスの情報が「サスペンド」以外の情報を示している場合にもステップ S 9 1 0 8 に進む。

#### 【1 8 6 8】

ステップ S 9 1 0 6 では、機能限定ウェイト処理を行う（詳細は後述する）。

#### 【1 8 6 9】

ステップ S 9 1 0 7 では、復電時処理を行う。この復電時処理では、電断時に R A M 3 0 8 に設けられたスタックポインタ退避領域に記憶しておいたスタックポインタを読み出し、スタックポインタに再設定する。また、電断時に R A M 3 0 8 に設けられたレジスタ退避領域に記憶しておいた各レジスタの値を読み出し、各レジスタに再設定した後、割り

10

20

30

40

50

込み許可の設定を行う。以降、CPU 304が、再設定後のスタックポインタやレジスタに基づいて制御プログラムを実行する結果、パチンコ機100は電源断時の状態に復帰する。すなわち、電断直前にタイマ割り込み処理（後述）に分岐する直前に行った（ステップS9110、ステップS9111内の所定の）命令の次の命令から処理を再開する。

#### 【1870】

ステップS9108では、初期化処理を行う。この初期化処理では、割り込み禁止の設定、スタックポインタへのスタック初期値の設定、RAM 308の全ての記憶領域の初期化などを行う。

#### 【1871】

ステップS9109では、機能限定ウェイト処理を行う（詳細は後述する）。 10

#### 【1872】

ステップS9110では、割り込み禁止の設定を行った後、基本乱数初期値更新処理を行う。この基本乱数初期値更新処理では、普図当選乱数カウンタ、および特図乱数値カウンタの初期値をそれぞれ生成するための2つの初期値生成用乱数カウンタと、普図タイマ乱数値、特図タイマ乱数値をそれぞれ生成するための2つの乱数カウンタを更新する。例えば、普図タイマ乱数値として取り得る数値範囲が0～20とすると、RAM 308に設けた普図タイマ乱数値を生成するための乱数カウンタ記憶領域から値を取得し、取得した値に1を加算してから元の乱数カウンタ記憶領域に記憶する。このとき、取得した値に1を加算した結果が21であれば0を元の乱数カウンタ記憶領域に記憶する。他の初期値生成用乱数カウンタ、乱数カウンタもそれぞれ同様に更新する。また、この基本乱数初期値更新処理の終了後に割り込み許可の設定を行ってステップS9111に進む。 20

#### 【1873】

ステップS9111では、演出乱数更新処理を行う。この演出乱数更新処理では、主制御部300で使用する演出用乱数値を生成するための乱数カウンタを更新する。

#### 【1874】

主制御部300は、所定の周期ごとに開始するタイマ割り込み処理を行っている間を除いて、ステップS9110およびステップS9111の処理を繰り返し実行する。

#### 【1875】

<機能限定ウェイト処理>

#### 【1876】

次に、図258を用いて、上述の主制御部リセット割り込み処理における機能限定ウェイト処理（ステップS9106、ステップS9109）について説明する。なお、同図は機能限定ウェイト処理の流れを示すフローチャートである。 30

#### 【1877】

ステップS9201では、上述の電源ステータス記憶領域に記憶した電源ステータスの情報に機能限定状態を示す情報を設定すると共に、RAM 308に設けた機能限定タイマ記憶領域に機能限定タイマの値として所定の数値（本実施例では3）を設定する。

#### 【1878】

ステップS9202では、機能限定状態終了条件が成立しているか否か（本実施例では、上述の機能限定タイマ記憶領域から機能限定タイマの値を読み出し、機能限定タイマの値が0になっているか否か）を判定する。そして、機能限定状態終了条件が成立している場合（本実施例では、機能限定タイマが0である場合）はステップS9203に進み、機能限定状態終了条件が成立していない場合（本実施例では、機能限定タイマが0でない場合）はステップS9202の処理を繰り返し実行し、機能限定状態終了条件が成立するのを待つ（本実施例では、機能限定タイマが0になるのを待つ）。 40

#### 【1879】

ステップS9203では、上述の電源ステータス記憶領域に電源ステータスとして正常状態を示す情報を設定すると共に、RAM 308に設けた加工種別記憶領域に、後述する加工種別の初期値を設定する。

#### 【1880】

## &lt;主制御部タイマ割り込み処理&gt;

## 【1881】

次に、図259を用いて、主制御部300のCPU304が実行する主制御部タイマ割り込み処理について説明する。なお、同図は主制御部タイマ割り込み処理の流れを示すフローチャートである。

## 【1882】

主制御部300は、所定の周期（本実施例では約2msに1回）でタイマ割り込み信号を発生するカウンタ・タイマ312を備えており、このタイマ割り込み信号を契機として主制御部タイマ割り込み処理を所定の周期で開始する。

## 【1883】

ステップS9301では、タイマ割り込みスタート処理を行う。このタイマ割り込みスタート処理では、CPU304の各レジスタの値をスタック領域に一時的に退避する処理などを行う。

## 【1884】

ステップS9302では、WDT314のカウント値が初期設定値（本実施例では32.8ms）を超えてWDT割り込みが発生しないように（処理の異常を検出しないように）、WDTを定期的に（本実施例では、主制御部タイマ割り込みの周期である約2msに1回）クリア&リスタートを行う。

## 【1885】

ステップS9303では、入力ポート状態更新処理を行う。この入力ポート状態更新処理では、I/O310の入力ポートを介して、上述のガラス枠開放センサ、前枠開放センサ、下皿満タンセンサ、複数の球検出センサを含む各種センサ320の検出信号を入力して検出信号の有無を監視し、RAM308に各種センサ320ごとに区画して設けた信号状態記憶領域に記憶する。本実施例では、前々回のタイマ割り込み処理（約4ms前）で検出した各々の球検出センサの検出信号の有無の情報を、RAM308に各々の球検出センサごとに区画して設けた前回検出信号記憶領域から読み出し、この情報をRAM308に各々の球検出センサごとに区画して設けた前々回検出信号記憶領域に記憶し、前回のタイマ割り込み処理（約2ms前）で検出した各々の球検出センサの検出信号の有無の情報を、RAM308に各々の球検出センサごとに区画して設けた今回検出信号記憶領域から読み出し、この情報を上述の前回検出信号記憶領域に記憶する。また、今回検出した各々の球検出センサの検出信号を、上述の今回検出信号記憶領域に記憶する。

## 【1886】

また、ステップS9303では、上述の前々回検出信号記憶領域、前回検出信号記憶領域、および今回検出信号記憶領域の各記憶領域に記憶した各々の球検出センサの検出信号の有無の情報を比較し、各々の球検出センサにおける過去3回分の検出信号の有無の情報が一致するか否かを判定する。そして、各々の球検出センサにおいて過去3回分の検出信号の有無の情報が、予め定めた入賞判定パターン情報（本実施例では、前々回検出信号無し、前回検出信号有り、今回検出信号有りであることを示す情報）と一致した場合に、入賞口（一般入賞口226、可変入賞口234）や始動口（第1特図始動口230、第2特図始動口232）への入球、または普図始動口228の通過があったと判定する。例えば、一般入賞口226への入球を検出する球検出センサにおいて過去3回分の検出信号の有無の情報が上述の入賞判定パターン情報と一致した場合には、一般入賞口226へ入球したと判定し、以降の一般入賞口226への入球に伴う処理を行うが、過去3回分の検出信号の有無の情報が上述の入賞判定パターン情報と一致しなかった場合には、以降の一般入賞口226への入球に伴う処理を行わずに後続の処理に分岐する。

## 【1887】

ステップS9304およびステップS9305では、基本乱数初期値更新処理および基本乱数更新処理を行う。これらの基本乱数初期値更新処理および基本乱数更新処理では、上記ステップS9110で行った初期値生成用乱数カウンタの値の更新を行い、次に主制御部300で使用する普図当選乱数値および特図乱数値をそれぞれ生成するための2つの

10

20

30

40

50



乱数カウンタを更新する。例えば、普図当選乱数値として取り得る数値範囲が0～100とすると、RAM308に設けた普図当選乱数値を生成するための乱数カウンタ記憶領域から値を取得し、取得した値に1を加算してから元の乱数カウンタ記憶領域に記憶する。このとき、取得した値に1を加算した結果が101であれば0を元の乱数カウンタ記憶領域に記憶する。また、取得した値に1を加算した結果、乱数カウンタが一周していると判定した場合にはそれぞれの乱数カウンタに対応する初期値生成用乱数カウンタの値を取得し、乱数カウンタの記憶領域にセットする。例えば、0～100の数値範囲で変動する普図当選乱数値生成用の乱数カウンタから値を取得し、取得した値に1を加算した結果が、RAM308に設けた所定の初期値記憶領域に記憶している前回設定した初期値と等しい値（例えば7）である場合に、普図当選乱数値生成用の乱数カウンタに対応する初期値生成用乱数カウンタから値を初期値として取得し、普図当選乱数値生成用の乱数カウンタにセットすると共に、普図当選乱数値生成用の乱数カウンタが次に1周したことを判定するために、今回設定した初期値を上述の初期値記憶領域に記憶しておく。なお、普図当選乱数値生成用の乱数カウンタが次に1周したことを判定するための上述の初期値記憶領域とは別に、特図乱数生成用の乱数カウンタが1周したことを判定するための初期値記憶領域をRAM308に設けている。

10

**【1888】**

ステップS9306では、演出乱数更新処理を行う。この演出乱数更新処理では、主制御部300で使用する演出用乱数値を生成するための乱数カウンタを更新する。

**【1889】**

ステップS9307では、上述の電源ステータス記憶領域に記憶した電源ステータスの情報を読み出し、読み出した電源ステータスの情報が機能限定状態を示す情報であるか否かを判定する。そして、電源ステータスの情報が機能限定状態を示す情報の場合には、後続のステップS9308～ステップS9317の処理を行うことなくステップS9318に進み、そうでない場合にはステップS9308に進む。

20

**【1890】**

ステップS9308では、タイマ更新処理を行う。詳細は後述するが、このタイマ更新処理では、普通図柄表示装置210に図柄を変動・停止表示する時間を計時するための普通図表示図柄更新タイマ、特別図柄表示装置212、214に図柄を変動・停止表示する時間を計時するための特図表示図柄更新タイマ、所定の入賞演出時間、所定の開放時間、所定の閉鎖時間、所定の終了演出期間などを計時するためのタイマなどを含む各種タイマを更新する。

30

**【1891】**

ステップS9309では、入賞口カウンタ更新処理を行う。この入賞口カウンタ更新処理では、入賞口（一般入賞口226、第1、第2特図始動口230、232、および可変入賞口234）に入賞（入球）があった場合に、RAM308に各入賞口ごとに設けた賞球数記憶領域の値を読み出し、1を加算して、元の賞球数記憶領域に設定する。

**【1892】**

また、ステップS9310では、入賞受付処理を行う。この入賞受付処理では、第1、第2特図始動口230、232に入賞があり、且つ、保留している特図変動遊技の数が4未満である場合には、入賞した始動口に対応するカウンタ回路318のカウンタ値記憶用レジスタから値を特図当選乱数値として取得する。また、上述の特図乱数値生成用の乱数カウンタから値を特図乱数値として取得し、RAM308に設けた乱数値記憶領域に特図当選乱数値と共に記憶する。また、普図始動口228を球が通過したことを検出し、且つ、保留している普図変動遊技の数が2未満の場合には、そのタイミングにおける普図当選乱数値生成用の乱数カウンタの値を普図当選乱数値として取得し、RAM308に設けた上述の特図用とは別の乱数値記憶領域に記憶する。また、この入賞受付処理では、所定の球検出センサにより第1、第2特図始動口230、232、普図始動口228、または可変入賞口234の入賞（入球）を検出した場合に、第1副制御部400に送信すべき送信情報に、第1、第2特図始動口230、232、普図始動口228、および可変入賞口2

40

50

34の入賞（入球）の有無を示す入賞受付情報を設定する。

【1893】

ステップS9311では、払出要求数送信処理を行う（詳細は後述する）。なお、払出制御部600に出力する出力予定情報および払出要求情報は1バイトで構成しており、ビット7にストロブ情報（オンの場合、データをセットしていることを示す）、ビット6に電源投入情報（オンの場合、電源投入後一回目のコマンド送信であることを示す）、ビット4～5に今回加工種別（0～3）、およびビット0～3に加工後の払出要求数を示すようにしている。

【1894】

ステップS9312では、普図状態更新処理を行う。この普図状態更新処理は、普図の状態に対応する複数の処理のうちの1つの処理を行う。例えば、普図変動中（後述する普図汎用タイマの値が1以上）における普図状態更新処理では、普図表示装置112を構成する7セグメントLEDの点灯と消灯を繰り返す点灯・消灯駆動制御を行う。

【1895】

また、普図変動表示時間が経過したタイミング（普図表示図柄更新タイマの値が1から0になったタイミング）における普図状態更新処理では、当りフラグがオンの場合には、図5（c）に示す普図Aの態様となるように普図表示装置210を構成する7セグメントLEDの点灯・消灯駆動制御を行い、当りフラグがオフの場合には、図5（c）に示す普図Bの態様となるように普図表示装置210を構成する7セグメントLEDの点灯・消灯駆動制御を行うと共に、その後、所定の停止表示期間（例えば500m秒間）その表示を維持するためにRAM308に設けた普図停止時間管理用タイマの記憶領域に停止期間を示す情報を設定する。この設定により普図の停止表示を行い、普図変動遊技の結果を遊技者に報知するようにしている。

【1896】

また、所定の停止表示期間が終了したタイミング（普図停止時間管理用タイマの値が1から0になったタイミング）で開始する普図状態更新処理では、当りフラグがオンの場合には、所定の開放期間（例えば2秒間）、第2特図始動口232の羽根部材の開閉駆動用のソレノイド332に、羽根部材を開放状態に保持する信号を出力するとともに、RAM308に設けた羽根開放時間管理用タイマの記憶領域に開放期間を示す情報を設定する。

【1897】

また、所定の開放期間が終了したタイミング（羽根開放時間管理用タイマの値が1から0になったタイミング）で開始する普図状態更新処理では、所定の閉鎖期間（例えば500m秒間）、羽根部材の開閉駆動用のソレノイド332に、羽根部材を閉鎖状態に保持する信号を出力するとともに、RAM308に設けた羽根閉鎖時間管理用タイマの記憶領域に閉鎖期間を示す情報を設定する。

【1898】

また、所定の閉鎖期間を経過したタイミング（羽根閉鎖時間管理用タイマの値が1から0になったタイミング）で開始する普図状態更新処理では、普図の状態を非作動中に設定する。普図の状態が非作動中の場合における普図状態更新処理では、何もせずに次のステップS9313に移行するようにしている。

【1899】

ステップS9313では、普図関連抽選処理を行う。この普図関連抽選処理では、普図変動遊技および第2特図始動口232の開閉制御を行っておらず（普図の状態が非作動中）、且つ、保留している普図変動遊技の数が1以上である場合に、上述の乱数値記憶領域に記憶している普図当選乱数値に基づいた乱数抽選により普図変動遊技の結果を当選とするか、不当選とするかを決定する当り判定をおこない、当選とする場合にはRAM308に設けた当りフラグにオンを設定する。不当選の場合には、当りフラグにオフを設定する。また、当り判定の結果に関わらず、次に上述の普図タイマ乱数値生成用の乱数カウンタの値を普図タイマ乱数値として取得し、取得した普図タイマ乱数値に基づいて複数の変動時間のうちから普図表示装置210に普図を変動表示する時間を1つ選択し、この変動表

10

20

30

40

50

示時間を、普図変動表示時間として、RAM 308に設けた普図変動時間記憶領域に記憶する。なお、保留している普図変動遊技の数は、RAM 308に設けた普図保留数記憶領域に記憶するようにしており、当たり判定をするたびに、保留している普図変動遊技の数から1を減算した値を、この普図保留数記憶領域に記憶し直すようにしている。また当たり判定に使用した乱数値を消去する。

#### 【1900】

ステップS9314では、特図状態更新処理を行う。この特図状態更新処理は、特図の状態に応じて、次の8つの処理のうちの1つの処理を行う。例えば、特図変動中（後述する特図汎用タイマの値が1以上）における特図状態更新処理では、特図表示装置212、214を構成する7セグメントLEDの点灯と消灯を繰り返す点灯・消灯駆動制御を行う。

10

#### 【1901】

また、特図変動表示時間が経過したタイミング（特図表示図柄更新タイマの値が1から0になったタイミング）で開始する特図状態更新処理では、大当たりフラグがオンで確変フラグがオンの場合には特図表示装置212、214に図5（a）に示す特図A等、大当たりフラグがオンで確変フラグがオフの場合には特図表示装置212、214に図5（a）に示す特図B等、大当たりフラグがオフの場合には、図3（a）に示す特図I等の態様となるように特図表示装置212、214を構成する7セグメントLEDの点灯・消灯駆動制御を行うと共に、その後、所定の停止表示期間（例えば500m秒間）その表示を維持するためにRAM 308に設けた特図停止時間管理用タイマの記憶領域に停止期間を示す情報を設定する。この設定により特図の停止表示をおこない、特図変動遊技の結果を遊技者に報知するようにしている。また、コマンド設定送信処理（ステップS9316）で一般コマンド回転停止設定送信処理を実行させるために上述の送信情報記憶領域に02Hを送信情報（一般情報）として追加記憶する。

20

#### 【1902】

また、所定の停止表示期間が終了したタイミング（特図停止時間管理用タイマの値が1から0になったタイミング）で開始する特図状態更新処理では、大当たりフラグがオンの場合には、所定の入賞演出期間（例えば3秒間）すなわち装飾図柄表示装置208による大当たりを開始することを遊技者に報知する画像を表示している期間待機するためにRAM 308に設けた特図待機時間管理用タイマの記憶領域に入賞演出期間を示す情報を設定する。また、コマンド設定送信処理（ステップS9316）で一般コマンド入賞演出設定送信処理を実行させるために上述の送信情報記憶領域に04Hを送信情報（一般情報）として追加記憶する。

30

#### 【1903】

また、所定の入賞演出期間が終了したタイミング（特図待機時間管理用タイマの値が1から0になったタイミング）で開始する特図状態更新処理では、所定の開放期間（例えば29秒間、または可変入賞口234に所定球数（例えば10球）の遊技球の入賞を検出するまで）可変入賞口234の扉部材の開閉駆動用のソレノイド332に、扉部材を開放状態に保持する信号を出力するとともに、RAM 308に設けた扉開放時間管理用タイマの記憶領域に開放期間を示す情報を設定する。また、コマンド設定送信処理（ステップS9316）で一般コマンド大入賞口開放設定送信処理を実行させるために上述の送信情報記憶領域に10Hを送信情報（一般情報）として追加記憶する。

40

#### 【1904】

また、所定の開放期間が終了したタイミング（扉開放時間管理用タイマの値が1から0になったタイミング）で開始する特図状態更新処理では、所定の閉鎖期間（例えば1.5秒間）可変入賞口234の扉部材の開閉駆動用のソレノイド332に、扉部材を閉鎖状態に保持する信号を出力するとともに、RAM 308に設けた扉閉鎖時間管理用タイマの記憶領域に閉鎖期間を示す情報を設定する。また、コマンド設定送信処理（ステップS9316）で一般コマンド大入賞口閉鎖設定送信処理を実行させるために上述の送信情報記憶領域に20Hを送信情報（一般情報）として追加記憶する。

50

## 【1905】

また、この扉部材の開放・閉鎖制御を所定回数（例えば15ラウンド）繰り返し、終了したタイミングで開始する特図状態更新処理では、所定の終了演出期間（例えば3秒間）すなわち装飾図柄表示装置208による大当りを終了することを遊技者に報知する画像を表示している期間待機するように設定するためにRAM308に設けた演出待機時間管理用タイマの記憶領域に演出待機期間を示す情報を設定する。また、コマンド設定送信処理（ステップS9316）で一般コマンド終了演出設定送信処理を実行させるために上述の送信情報記憶領域に08Hを送信情報（一般情報）として追加記憶する。

## 【1906】

また、所定の終了演出期間が終了したタイミング（演出待機時間管理用タイマの値が1から0になったタイミング）で開始する特図状態更新処理では、特図の状態を非作動中に設定する。特図の状態が非作動中の場合における特図状態更新処理では、何もせずに次のステップS9315に移行するようにしている。

## 【1907】

ステップS9315では、特図関連抽選処理を行う。この特図関連抽選処理では、特図変動遊技および可変入賞口234の開閉制御を行っておらず（特図の状態が非作動中）、且つ、保留している特図変動遊技の数が1以上である場合に、上述の図255（a）の大当たり判定テーブル、同図（b）の高確率状態移行判定テーブル、同図（c）のタイマ番号決定テーブルなどを使用した各種抽選のうち、最初に大当たり判定を行う。具体的には、ステップS9203で乱数値記憶領域に記憶した特図当選乱数値が、図255（a）に示す大当たり判定テーブルの第1特図始動口用抽選データの数値範囲であるか否かを判定し、特図当選乱数値が第1特図始動口用抽選データの数値範囲である場合には、特図変動遊技の当選と判定してRAM308に設けた大当たりフラグの格納領域に大当たりとなることを示す情報を設定する（ここで、大当たりの情報をRAM308に設定することを大当たりフラグをオンに設定するという）。一方、特図当選乱数値が第1特図始動口用抽選データの数値範囲以外である場合には、特図変動遊技の外れと判定してRAM308に設けた大当たりフラグの格納領域に外れとなることを示す情報を設定する（ここで、外れの情報をRAM308に設定することを大当たりフラグをオフに設定するという）。なお、保留している特図変動遊技の数は、RAM308に設けた特図保留数記憶領域に記憶するようにしており、当り判定をするたびに、保留している特図変動遊技の数から1を減算した値を、この特図保留数記憶領域に記憶し直すようにしている。また、当り判定に使用した乱数値を消去する。

## 【1908】

具体例としては、遊技状態が低確率状態であり、第1特図始動口230への球入賞の検出に基づいて取得した特図当選乱数値が10100の場合は大当たりフラグをオンに設定し、特図当選乱数値が10200の場合は大当たりフラグをオフに設定する。また、第2特図始動口232への球入賞の検出に基づいて取得した特図当選乱数値が20100の場合は大当たりフラグをオンに設定し、特図当選乱数値が20200の場合は大当たりフラグをオフに設定する。

## 【1909】

大当たりフラグにオンを設定した場合には、次に確変移行判定を行う。具体的には、ステップS9203で乱数値記憶領域に記憶した特図乱数値が、図255（b）に示す移行判定乱数の数値範囲であるか否かを判定し、特図乱数値が抽選データの数値範囲である場合には、RAM308に設けた確変（確率変動）フラグの格納領域に、特別大当たり遊技を開始することを示す情報を設定する。（ここで、特別大当たり遊技開始の情報をRAM308に設定することを確変フラグをオンに設定するという）。一方、特図乱数値が抽選データの数値範囲以外である場合には、上述の確変フラグの格納領域に、大当たり遊技を開始することを示す情報を設定する（ここで、大当たり遊技開始の情報をRAM308に設定することを確変フラグをオフに設定するという）。例えば、取得した特図乱数値が20の場合には確変フラグをオンに設定する。一方、取得した特図乱数値が特図乱数値が80の場合に

は確変フラグをオフに設定する。

#### 【1910】

大当たり判定の結果に関わらず、次にタイマ番号を決定する処理を行う。具体的には、上述の特図タイマ乱数値生成用の乱数カウンタの値の特図タイマ乱数値として取得する。大当たりフラグの値、および取得した特図タイマ乱数値を含む図255(c)に示すタイマ乱数の数値範囲に対応するタイマ番号を選択し、RAM308に設けた所定のタイマ番号格納領域に記憶する。さらに、そのタイマ番号に対応する変動時間を、特図変動表示時間として、上述の特図表示図柄更新タイマに記憶し、コマンド設定送信処理(ステップS9316)で一般コマンド回転開始設定送信処理を実行させるために上述の送信情報記憶領域に01Hを送信情報(一般情報)として追加記憶してから処理を終了する。

10

#### 【1911】

例えば、大当たりフラグがオフで、取得した特図タイマ乱数値が50000の場合には、特図タイマ乱数値は0~60235の範囲であることから、タイマ番号決定テーブルのそれらの条件に対応する1行目に記憶しているタイマ番号を示すタイマ1、および変動時間を示す5を選択し、RAM308に設けたそれぞれの記憶領域に記憶する。一方、大当たりフラグがオンで、取得した特図タイマ乱数値が64000の場合には、特図タイマ乱数値は0~15535の範囲ではないことからタイマ2は選択せず、15536~24535ではないことからタイマ3は選択せず、24536~62535ではないことからタイマ4は選択しないが、62536~65535の範囲内であることから、タイマ番号決定テーブルのそれらの条件に対応する8行目に記憶しているタイマ番号を示すタイマ5、および変動時間を示す50を選択し、RAM308に設けたそれぞれの記憶領域に記憶する。なお、割り込み処理の開始周期である2msを考慮して、選択した変動時間の値に500(1000ms/2ms)を掛けた値を変動時間記憶領域にセットする。例えば、変動時間が5秒の場合には、変動時間記憶領域には2500の値を初期値としてセットし、ステップS9308のタイマ更新処理を実行する度に、この変動時間記憶領域の値を1だけ減算するようにすることで、割り込み処理の実行回数により時間の経過を計測できるようにしている。また、複数回(例えば5回)のタイマ割込処理の実行ごと(例えば2ms周期)に変動時間記憶領域の値を減算する場合には、変動時間が10秒の場合であれば、10秒が10000msであることから周期(2ms×5)で割り算して1000を変動時間記憶領域に設定する。

20

30

#### 【1912】

ステップS9316では、コマンド設定送信処理を行う(詳細は後述する)。なお、演出制御部350に送信する出力予定情報は16ビットで構成しており、ビット15はストロープ情報(オンの場合、データをセットしていることを示す)、ビット11~14はコマンド種別(00Hの場合は基本コマンド、01Hの場合は図柄変動開始コマンド、04Hの場合は図柄変動停止コマンド、05Hの場合は入賞演出開始コマンド、06Hの場合は終了演出開始コマンド、07Hの場合は大当たりラウンド数指定コマンド、0EHの場合は復電コマンド、0FHの場合はRAMクリアコマンドをそれぞれ示すなどコマンドの種類を特定可能な情報)、ビット0~10はコマンドデータ(コマンド種別に対応する所定の情報)で構成している。

40

#### 【1913】

具体的には、ストロープ情報は上述のコマンド送信処理でオン、オフするようにしている。また、コマンド種別が図柄変動開始コマンドの場合であればコマンドデータに、大当たりフラグの値、確変フラグの値、特図関連抽選処理で選択したタイマ番号などを示す情報を含み、図柄変動停止コマンドの場合であれば、大当たりフラグの値、確変フラグの値などを含み、入賞演出コマンドおよび終了演出開始コマンドの場合であれば、確変フラグの値などを含み、大当たりラウンド数指定コマンドの場合であれば確変フラグの値、大当たりラウンド数などを含むようにしている。コマンド種別が基本コマンドを示す場合は、コマンドデータにデバイス情報、第1特図始動口230への入賞の有無、第2特図始動口232への入賞の有無、可変入賞口234への入賞の有無などを含む。

50

## 【1914】

また、上述の一般コマンド回転開始設定送信処理では、コマンド種別に01H、コマンドデータにRAM308に記憶している大当りフラグの値、確変フラグの値、特図関連抽選処理で選択したタイマ番号、保留している特図変動遊技の数などを示す情報を設定する。上述の一般コマンド回転停止設定送信処理では、コマンド種別に04H、コマンドデータにRAM308に記憶している大当りフラグの値、確変フラグの値などを示す情報を設定する。上述の一般コマンド入賞演出設定送信処理では、コマンド種別に05H、コマンドデータにRAM308に記憶している入賞演出期間中に装飾図柄表示装置208・各種ランプ418・スピーカ120に出力する演出制御情報、確変フラグの値、保留している特図変動遊技の数などを示す情報を設定する。上述の一般コマンド終了演出設定送信処理では、コマンド種別に06H、コマンドデータにRAM308に記憶している演出待機期間中に装飾図柄表示装置208・各種ランプ418・スピーカ120に出力する演出制御情報、確変フラグの値、保留している特図変動遊技の数などを示す情報を設定する。上述の一般コマンド大入賞口開放設定送信処理では、コマンド種別に07H、コマンドデータにRAM308に記憶している大当りラウンド数、確変フラグの値、保留している特図変動遊技の数などを示す情報を設定する。上述の一般コマンド大入賞口閉鎖設定送信処理では、コマンド種別に08H、コマンドデータにRAM308に記憶している大当りラウンド数、確変フラグの値、保留している特図変動遊技の数などを示す情報を設定する。第1副制御部400では、受信した出力予定情報に含まれるコマンド種別により、主制御部300における遊技制御の変化に応じた演出制御の決定が可能になるとともに、出力予定情報に含まれているコマンドデータの情報に基づいて、演出制御内容を決定することができるようになる。

10

20

## 【1915】

ステップS9317では、外部出力信号設定処理を行う。この外部出力信号設定処理では、RAM308に記憶している遊技情報を、情報出力回路334を介してパチンコ機100とは別体の情報入力回路350に出力する。

## 【1916】

ステップS9318では、機能限定状態終了条件が成立しているか否か（本実施例では、上述の機能限定タイマ記憶領域から機能限定タイマの値を読み出して1を減算し、減算後の機能限定タイマの値が0になっているか否か）を判定する。そして、機能限定状態終了条件が成立している場合（本実施例では、減算後の機能限定タイマが0である場合）はステップS9319に進み、機能限定状態終了条件が成立していない場合（本実施例では、減算後の機能限定タイマが0でない場合）はステップS9319の処理を行うことなくステップS9321に進む。なお、本実施例では、上述の機能限定ウェイト処理において機能限定タイマの初期値として3を設定しているため、機能限定ウェイト処理において初期値を設定し、且つ、主制御部タイマ割り込みの割り込みを許可してから、最大で約6ms（＝約2ms（主制御部タイマ割り込みの周期）×3）後、最小で約4ms（＝約2ms（主制御部タイマ割り込みの周期）×2）後に機能限定タイマの減算結果が0になる（機能限定を解除する）。

30

## 【1917】

ステップS9319では、電源ステータス、チェックサムなど所定の電断情報などを用いて、電断前の状態に復帰したこと、さらに入力ポートの履歴情報の更新が正常に終了したことを示す復帰コマンドを第1副制御部400に送信する復帰コマンド設定送信処理を行う（詳細は後述する）。また、その処理に続いて、後述する基本コマンド設定送信処理を実行して、遊技の進行状況を示す様々な情報を含んだ基本コマンドを送信するようにしている。

40

## 【1918】

ステップS9320では、デバイス監視処理を行う。このデバイス監視処理では、ステップS9303において信号状態記憶領域に記憶した各種センサの信号状態を読み出して、ガラス枠開放エラーの有無、前枠開放エラーの有無、または下皿満タンエラーの有無な

50

どを監視し、ガラス枠開放エラー、前枠開放エラー、または下皿満タンエラーを検出した場合に、演出制御部 350 に送信すべき送信情報に、ガラス枠開放エラーの有無、前枠開放エラーの有無、下皿満タンエラーの有無を示すデバイス情報を設定する。また、各種ソレノイド 332 を駆動して第 2 特図始動口 232 や、可変入賞口 234 の開閉を制御したり、表示回路を介して普図表示装置 210、特図表示装置 212、214、各種状態表示部 328 などに出力する表示データを、I/O 310 の出力ポートに設定する。また、後述の払出要求数送信処理（ステップ S9311）のステップ S9415 で設定した出力予定情報を出力ポート 310 を介して第 1 副制御部 400 に出力する。

#### 【1919】

ステップ S9321 では、低電圧信号がオンであるか否かを監視する。そして、低電圧信号がオンの場合（電源の遮断を検知した場合）にはステップ S9323 に進み、低電圧信号がオフの場合（電源の遮断を検知していない場合）にはステップ S9322 に進む。

#### 【1920】

ステップ S9322 では、タイマ割り込みエンド処理を行う。このタイマ割り込みエンド処理では、ステップ S9301 で一時的に退避した各レジスタの値を元の各レジスタに設定したり、割り込み許可の設定などを行う。

#### 【1921】

ステップ S9323 では、上述の電源ステータス記憶領域に記憶した電源ステータスの情報を読み出し、読み出した電源ステータスの情報が正常状態を示す情報であるか否かを判定する。そして、電源ステータスの情報が正常状態を示す情報の場合にはステップ S9325 に進み、そうでない場合にはステップ S9324 に進む。

#### 【1922】

ステップ S9324 では、電断時処理 1 を行う。この電断時処理 1 では、上述のスタックポインタ退避領域に現在のスタックポインタの値を記憶し、上述の電源ステータス記憶領域にサスペンドを示す情報を設定する。また、RAM 308 の所定の領域（例えば全ての領域）に記憶している 1 バイトデータを初期値が 0 である 1 バイト構成のレジスタに全て加算し、チェックサム算出用数値記憶領域に記憶している値からその加算した結果を減算した値をチェックサム（電断時チェックサム）として算出し、算出した電断時チェックサムを上述のチェックサム算出用数値記憶領域に記憶し、RAM 308 への書き込みを禁止する設定を行った後、無限ループとなる。

#### 【1923】

ステップ S9325 では、電断時処理 2 を行う。この電断時処理 2 では、上述のスタックポインタ退避領域に現在のスタックポインタの値を記憶することなく、上述の電源ステータス記憶領域にサスペンドを示す情報を設定する。また、RAM 308 に記憶している数値を読み出してチェックサム（電断時チェックサム）を算出し、算出した電断時チェックサムを上述のチェックサム算出用数値記憶領域に記憶し、RAM 308 への書き込みを禁止する設定を行った後、無限ループとなる。

#### 【1924】

＜払出要求数送信処理＞

#### 【1925】

次に、図 260 を用いて、上述の主制御部タイマ割り込み処理における払出要求数送信処理（ステップ S9311）について説明する。なお、同図は払出要求数送信処理の流れを示すフローチャートである。

#### 【1926】

ステップ S9401 では、RAM 308 に設けた払出要求数記憶領域に払出要求数として 0 を設定する。

#### 【1927】

ステップ S9402 では、RAM 308 に設けた出力予定情報記憶領域から出力予定情報（I/O ポート 310 から出力する予定の情報）を読み出し、出力予定情報に情報が設定されているか否かを判定する。そして、出力予定情報に情報が設定されている場合には

10

20

30

40

50

ステップ S 9 4 1 5 に進み、出力予定情報に情報が設定されていない場合にはステップ S 9 4 0 3 に進む。

【1928】

ステップ S 9 4 0 3 では、電源投入後の最初の送信処理か否かを示す情報を R A M 3 0 8 に設けた所定の記憶領域から読み出して、電源投入後の最初の送信処理か否かを判定する。そして、電源投入後の最初の送信処理の場合にはステップ S 9 4 1 2 に進み、そうでない場合はステップ S 9 4 0 4 に進む。

【1929】

ステップ S 9 4 0 4 では、最初に、入賞により賞球の払出しを行う各入賞口（本実施例では、一般入賞口 2 2 6、第 1、第 2 特図始動口 2 3 0、2 3 2、および可変入賞口 2 3 4）を、処理対象として順次選択する。次に、各処理対象に対応して R A M 3 0 8 に設けられた払出カウンタ記憶領域から、各処理対象の払出カウンタの値を読み出し、払出カウンタの値が 0 であるか否かを判定する。そして、処理対象の払出カウンタが 0 の場合はステップ S 9 4 0 5 に進み、処理対象の払出カウンタが 0 以外の場合はステップ S 9 4 0 6 に進む。

10

【1930】

ステップ S 9 4 0 5 では、全ての処理対象の払出カウンタの処理が終了したか否かを判定する。そして、全ての処理対象の払出カウンタの処理が終了した場合には主制御部タイマ割り込み処理に復帰し、全ての処理対象の払出カウンタの処理が終了していない場合はステップ S 9 4 0 4 に戻り、残りの処理対象の払出カウンタの処理を継続する。

20

【1931】

ステップ S 9 4 0 6 では、上述の払出カウンタ記憶領域から処理対象の払出カウンタの値を読み出して 1 を減算した後、減算後の値を新たな払出カウンタの値として元の払出カウンタ記憶領域に記憶する。

【1932】

ステップ S 9 4 0 7 では、上述の図 2 5 6（a）の払出要求数テーブルを参照し、処理対象に対応する払出要求情報（払出要求数）を取得して、R A M 3 0 8 の所定記憶領域に記憶する。例えば、処理対象が第 1 特図始動口 2 3 0 の場合には、払出要求数として払出要求テーブルの 1 行目のデータ、すなわち 3 を選択する。他の処理対象も同様にして、処理対象が第 2 特図始動口 2 3 2 の場合には払出要求テーブルの 2 行目のデータである 4 を、処理対象が一般入賞口 2 2 6 の場合には払出要求テーブルの 3 行目のデータである 1 0 を、処理対象が可変入賞口 2 3 4 の場合には払出要求テーブルの 4 行目のデータである 1 5 をそれぞれ選択する。

30

【1933】

ステップ S 9 4 0 8 では、ハードウェア乱数カウンタとして使用している、基本回路 3 0 2 に設けている 1 または 2 の値をランダムに出力する乱数生成回路（図示せず）から取得した乱数値と、前回の送信に使用した加工種別（前回加工種別）に基づいて（過去の加工種別の履歴に基づいて）、新しい加工種別（今回加工種別）を選択する。具体的には、R A M 3 0 8 に設けた前回加工種別記憶領域から前回の送信時に選択し、記憶した前回加工種別を読み出し、この前回加工種別に、取得した乱数値を加算する。そして、加算結果が 4 未満の場合には加算結果を R A M 3 0 8 に設けた今回加工種別記憶領域に今回加工種別として記憶し、加算結果が 4 以上の場合には加算結果から 4 を減算し、この減算結果を今回加工種別記憶領域に今回加工種別として記憶する。

40

【1934】

ステップ S 9 4 0 9 では、ステップ S 9 4 0 8 で今回加工種別記憶領域に記憶した今回加工種別を読み出すと共に、上述の図 2 5 6（b）の払出加工テーブルを参照して、今回加工種別に対応する加工データを選択する。例えば、今回加工種別が 0 の場合には、加工データとして払出加工テーブルの 1 行目のデータ、すなわち 0 0 H を選択する。他の加工データも同様にして、今回加工種別が 1 の場合には払出加工テーブルの 2 行目のデータである 1 3 H を、今回加工種別が 2 の場合には払出加工テーブルの 3 行目のデータである 2

50



E Hを、今回加工種別が3の場合には払出加工テーブルの4行目のデータである39Hをそれぞれ選択する。

【1935】

ステップS9410では、ステップS9408で今回加工種別記憶領域に記憶した今回加工種別を読み出して払出要求情報に追加する。

【1936】

ステップS9411では、ステップS9409で選択した加工データを用いて払出要求情報を加工する。本実施例では、払出要求数と加工データの排他的論理和（E X O R）を算出し、その算出結果を、加工後の払出要求数として払出要求情報に設定する。例えば、払出要求数が3（00000011B（数値の後ろのBは数値が2進数であることを示す。以下同じ））、加工データが00H（00000000B）の場合、払出要求数と加工データの排他的論理和は3であるから、加工後の払出要求数は3である。また、払出要求数が3、加工データが13H（00010011B）の場合、払出要求数と加工データの排他的論理和は10H（00010000B）であるから、加工後の払出要求数は10Hであり、払出要求数が3、加工データが2EH（00101110B）の場合、払出要求数と加工データの排他的論理和は2DH（00101100B）であるから、加工後の払出要求数は2DHであり、払出要求数が3、加工データが39H（00111001B）の場合、払出要求数と加工データの排他的論理和は3AH（00111010B）であるから、加工後の払出要求数は3AHである。

【1937】

ステップS9412では、払出要求情報に初期化情報（後述する電源投入情報のビットにオン）をセットする。

【1938】

ステップS9413では、ステップS9411またはステップS9412で設定した払出要求情報をI/Oポート310から払出制御部600に出力する。

【1939】

ステップS9414では、払出要求情報にストロープ情報を追加する。

【1940】

ステップS9415では、払出要求情報を、RAM308に設けた出力予定情報記憶領域に出力予定情報として記憶する。ステップS9401で払出要求情報をクリアした場合には、後続のデバイス監視処理（ステップS9302）において、全てオフを示す情報をI/Oポート310から払出制御部600に出力する。また、ステップS9414でストロープ情報を追加した場合には、後続のデバイス監視処理（ステップS9302）において、ストロープ情報を追加した払出要求情報をI/Oポート310から払出制御部600に出力する。

【1941】

この払出要求数送信処理のステップS9413の処理ではストロープ情報を含まず、払出要求情報を含む信号をI/Oポート310から払出制御部600に出力する。この払出要求数送信処理ではRAM308に設けた出力予定情報に記憶領域の出力予定情報にストロープ情報、および払出要求情報を記憶させたところで処理を終了し、この出力予定情報をI/Oポート310から信号として出力することはない。次にデバイス監視処理（ステップS9320）でストロープ情報、および払出要求情報、すなわち出力予定情報を含む信号をI/Oポート310から払出制御部600に出力する。これにより払出制御部600の基本回路が入力しているストロープ信号も立ち下がる。さらに次に実行されるタイマ割り込み処理における払出要求数送信処理で、上述の出力予定情報記憶領域を初期化（全信号オフとなる情報を記憶）し、この全オフとなった出力予定情報をI/Oポート310から信号として出力することはない。次に実行するデバイス監視処理（ステップS9320）で全信号オフとすることを示す出力予定情報を含む信号をI/Oポート310から払出制御部600に出力する。ここでストロープ情報もオフにしていることから払出制御部600の基本回路が入力しているストロープ信号も立ち上がる。

## 【1942】

<タイマ更新処理>

## 【1943】

次に、図261を用いて、上述の主制御部タイマ割り込み処理におけるタイマ更新処理（ステップS9308）について説明する。なお、同図はタイマ更新処理の流れを示すフローチャートである。なお、後述する普図汎用タイマは、上述の普図表示図柄更新タイマ、普図停止時間管理用タイマ、羽根開放時間管理用タイマ、羽根閉鎖時間管理用タイマを兼用した1つのタイマであり、特図汎用タイマは、上述の特図表示図柄更新タイマ、特図停止時間管理用タイマ、特図待機時間管理用タイマ、扉開放時間管理用タイマ、扉開放時間管理用タイマ、扉閉鎖時間管理用タイマ、および演出待機時間管理用タイマを兼用した1つのタイマである。

10

## 【1944】

ステップS9501では、上述の図256(c)のタイマ制御データテーブルを参照して、タイマ種類数、タイマ個数、および処理対象のタイマを取得する。本実施例では、タイマ種類数としてタイマ制御データテーブルの1行目のデータである2、タイマ個数としてタイマ制御データテーブルの2行目のデータである9、処理対象のタイマとしてタイマ1（特図表示図柄更新タイマ）をそれぞれ取得する。

## 【1945】

ステップS9502では、処理対象のタイマに対応してRAM308に設けたタイマ値記憶領域からタイマ値を読み出して、そのタイマ値が0であるか否かを判定し、タイマ値が0でない場合はステップS9503でタイマ値から1を減算し、減算結果を元のタイマ値記憶領域に記憶する。一方、タイマ値が0の場合はステップS9503の処理を飛ばしてステップS9504に進む。

20

## 【1946】

ステップS9504では、タイマ個数（本実施例では9）から1を減算し、ステップS9505では、減算後のタイマ個数が0であるか否かを判定する。そして、減算後のタイマ個数が0の場合にはステップS9507に進み、減算後のタイマ個数が0でない場合はステップS9506に進む。

## 【1947】

ステップS9506では、処理対象とするタイマを更新し、次の処理対象のタイマのタイマ値を減算すべくステップS9502に戻る。本実施例では、処理対象のタイマを、タイマ制御データテーブルの3行目に格納アドレスを記憶した特図表示図柄更新タイマ→タイマ制御データテーブルの4行目に格納アドレスを記憶した普図表示図柄更新タイマ→・・・タイマ制御データテーブルの10行目に格納アドレスを記憶した普図汎用タイマ→タイマ制御データテーブルの11行目に格納アドレスを記憶した10.08msタイマの順で更新する。

30

## 【1948】

ステップS9507では、タイマ種類数（本実施例では初期値は2）から1を減算し、ステップS9508では、減算後のタイマ種類数が0であるか否かを判定する。そして、減算後のタイマ種類が0の場合にはタイマ割り込み処理に復帰し、減算後のタイマ種類が0でない場合はステップS9509に進む。

40

## 【1949】

ステップS9509では、処理対象のタイマ（本実施例では、タイマ制御データテーブルの11行目に格納アドレスを記憶した10.08msタイマ）に対応してRAM308に設けたタイマ値記憶領域からタイマ値を読み出して、そのタイマ値が0であるか否かを判定し、タイマ値が0でない場合はタイマ割り込み処理に復帰し、タイマ値が0の場合はステップS9510に進む。

## 【1950】

ステップS9510では、タイマ制御データテーブルからタイマ値の初期値（本実施例では、タイマ制御データテーブルの12行目のタイマ更新周期に対応する数値の5）を取

50

得し、取得した初期値を処理対象のタイマに設定する。

#### 【1951】

ステップS9511では、タイマ制御データテーブルからタイマ個数を取得し（本実施例では、タイマ個数としてタイマ制御データテーブルの13行目のデータである1を取得し）、処理対象のタイマを次のタイマ（本実施例では、タイマ制御データテーブルの14行目に格納アドレスを記憶した特図汎用タイマ）に更新した後、更新後のタイマのタイマ値を減算すべくステップS9502に戻る。

#### 【1952】

このような構成により、本実施例では、タイマ制御データテーブルの14行目に対応するタイマは約10ms（約2ms（主制御部タイマ割り込みの割り込み周期）×5（タイマ制御データテーブルの12行目のタイマ更新周期））毎に更新し、タイマ制御データテーブルの3～11行目に対応するタイマは約2ms（主制御部タイマ割り込みの割り込み周期）ごとに更新する。なお、タイマ制御データテーブルの14行目に対応するタイマを更新する条件となるタイマ（ここではタイマ制御データテーブルの11行目に対応するタイマ）は適宜箇所で初期化しておく必要がある。本実施例では、このタイマに対応する特図表示図柄更新タイマの初期化は、特図変動表示時間として、上述の特図関連抽選処理（ステップS9315）の特図表示図柄更新タイマにタイマ番号に対応する変動時間を記憶すると同時期（同一割り込み）に行うのが最適である。

#### 【1953】

＜コマンド設定送信処理＞

#### 【1954】

次に、図262を用いて、上述の主制御部タイマ割り込み処理におけるコマンド設定送信処理（ステップS9316）について説明する。なお、同図はコマンド設定送信処理の流れを示すフローチャートである。

#### 【1955】

ステップS9601では、第1副制御部400に送信すべき送信情報に、上述の入賞受付情報、デバイス情報などの汎用情報が含まれているか否かを判定し、送信情報に汎用情報が含まれている場合にはステップS9602で基本コマンド設定送信処理（詳細は後述する）を行い、送信情報に汎用情報が含まれていない場合にはステップS9603に進む。

#### 【1956】

ステップS9603では、第1副制御部400に送信すべき情報があるか無いか（送信情報記憶領域に送信情報がセットされているかどうか）を判定し、送信すべき情報がある場合にはステップS9604に進み、そうでない場合は主制御部タイマ割り込み処理に戻る。

#### 【1957】

ステップS9604では、上述の図256(d)のジャンプテーブルを参照し、RAM308に設けた送信情報記憶領域に記憶した送信情報に基づいてジャンプ先アドレスを選択する。例えば、送信情報が01Hの場合には、ジャンプ先アドレスとして、ジャンプテーブルの1行目の一般コマンド回転開始設定送信処理（の先頭アドレス）を選択する。送信情報が01H以外の場合も同様にして、送信情報が02Hの場合には、ジャンプテーブルの2行目の一般コマンド回転停止設定送信処理（の先頭アドレス）を、送信情報が04Hの場合には、ジャンプテーブルの3行目の一般コマンド入賞演出設定送信処理（の先頭アドレス）を、送信情報が08Hの場合には、ジャンプテーブルの4行目の一般コマンド終了演出設定送信処理（の先頭アドレス）を、送信情報が10Hの場合には、ジャンプテーブルの5行目の一般コマンド大入賞口開放設定送信処理（の先頭アドレス）を、送信情報が20Hの場合には、ジャンプテーブルの6行目の一般コマンド大入賞口閉鎖設定送信処理（の先頭アドレス）を選択する。各処理の処理内容は後述する。

#### 【1958】

ステップS9605では、ステップS9604で選択したジャンプ先アドレスをPC（

10

20

30

40

50

プログラムカウンタ)に設定してジャンプ先アドレスに処理を移し、ジャンプ先アドレス以降に記憶した制御プログラムに従って各種の処理を行う。ジャンプ先での処理が終了すると一旦コマンド設定送信処理に復帰し、直ちに主制御部タイマ割り込み処理に復帰するようにしている。

#### 【1959】

＜基本コマンド設定送信処理＞

#### 【1960】

次に、図263を用いて、上述のコマンド設定送信処理における基本コマンド設定送信処理（ステップS9602）について説明する。なお、同図は基本コマンド設定送信処理の流れを示すフローチャートである。

10

#### 【1961】

ステップS9701では、第1副制御部400に送信すべき基本コマンド情報を、出力予定情報に設定する。

#### 【1962】

ステップS9702およびステップS9703では、上述の汎用情報に含まれる入賞受付情報およびデバイス情報を、出力予定情報に含める。

#### 【1963】

ステップS9704では、コマンド送信処理（詳細は後述する）を行ってから、送信情報記憶領域の送信情報より汎用情報を削除し、処理を終了する。なお、本実施例1では、可変入賞装置内に遊技媒体が進入したことなどを示す情報を含む上述の入賞受付情報と、前面枠扉106の開閉状態などを示す情報を含む上述のデバイス情報と、を同時に第1副制御部400に送信する。なお、始動口や可変入賞装置内に遊技媒体が進入したことなどを示す情報を含む上述の入賞受付情報を先に第1副制御部400に送信し、同一割り込み内で前面枠扉106の開閉状態などを示す情報を含む上述のデバイス情報を第1副制御部400に送信するようにしてもよい。また、逆に前面枠扉106の開閉状態などを示す情報を先に第1副制御部400に送信し、同一割り込み内で始動口や可変入賞装置内に遊技媒体が進入したことなどを示す情報を含む上述の入賞受付情報を先に第1副制御部400に送信するようにしてもよい。

20

#### 【1964】

＜コマンド送信処理＞

30

#### 【1965】

次に、図264を用いて、上述のコマンド設定送信処理におけるコマンド送信処理（ステップS9704）について説明する。なお、同図はコマンド送信処理の流れを示すフローチャートである。

#### 【1966】

ステップS9801では、上述の出力予定情報記憶領域に記憶した出力予定情報を読み出して、この出力予定情報のデータをI/Oポート310から第1副制御部400に出力する。

#### 【1967】

ステップS9802では、データ保持前時間を設定し、ステップS9803では、ステップS9802で設定したデータ保持前時間を減算し、データ保持前時間が0になったか否か（データ保持前時間が経過したか否か）を判定する。そして、データ保持前時間が経過している場合にはステップS9804に進み、データ保持前時間が経過していない場合にはステップS9803においてデータ保持前時間が経過するのを待つ。

40

#### 【1968】

ステップS9804では、出力予定情報にストロブ情報を追加して出力ポート310から第1副制御部400に出力する。

#### 【1969】

ステップS9805では、データ保持前時間を設定し、ステップS9806では、ステップS9805で設定したデータ保持時間を減算し、データ保持時間が0になったか否か

50

(データ保持時間が経過したか否か)を判定する。そして、データ保持時間が経過している場合にはステップS9807に進み、データ保持時間が経過していない場合にはステップS9806においてデータ保持時間が経過するのを待つ。

#### 【1970】

ステップS9807では、出力予定情報記憶領域に記憶した出力予定情報をクリアし、ステップS9808では、出力予定情報記憶領域に記憶した出力予定情報を読み出して、この出力予定情報をI/Oポート310から出力する。

#### 【1971】

コマンド設定送信処理(ステップS9316)の開始時における送信情報に汎用情報が含まれている場合にはまず、基本コマンド設定送信処理から分岐したコマンド送信処理のステップS9801でストロブ情報を含まず、汎用情報を含む信号を出力ポート310を介して第1副制御部400に出力し、所定期間後に開始するステップS9804でストロブ情報、および汎用情報を含む信号を出力ポート310を介して第1副制御部400に出力する。これにより第1副制御部400の基本回路402が入力しているストロブ信号も立ち下がる。さらに所定期間後にステップS9808で全信号がオフである信号を出力ポート310を介して第1副制御部400に出力するようにしている。ここでストロブ情報もオフにしていることから第1副制御部400の基本回路402が入力しているストロブ信号も立ち上がる。次にステップS9605からジャンプした先の処理から分岐したコマンド送信処理のステップS9801でストロブ情報を含まず、一般情報を含む信号を出力ポート310を介して第1副制御部400に出力し、所定期間後に開始するステップS9804でストロブ情報、および一般情報を含む信号を出力ポート310を介して第1副制御部400に出力する。これにより第1副制御部400の基本回路402が入力しているストロブ信号も立ち下がる。さらに所定期間後にステップS9808で全信号がオフである信号を出力ポート310を介して第1副制御部400に出力するようにしている。ここでストロブ情報もオフにしていることから第1副制御部400の基本回路402が入力しているストロブ信号も立ち上がる。

#### 【1972】

一方、コマンド設定送信処理(ステップS9316)の開始時における送信情報に汎用情報が含まれていない場合には汎用情報を出力せず、ステップS9605からジャンプした先の処理から分岐したコマンド送信処理のステップS9801でストロブ情報を含まず、一般情報を含む信号を出力ポート310を介して第1副制御部400に出力し、所定期間後に開始するステップS9804でストロブ情報、および一般情報を含む信号を出力ポート310を介して第1副制御部400に出力する。これにより第1副制御部400の基本回路402が入力しているストロブ信号も立ち下がる。さらに所定期間後にステップS9808で全信号がオフである信号を出力ポート310を介して第1副制御部400に出力するようにしている。ここでストロブ情報もオフにしていることから第1副制御部400の基本回路402が入力しているストロブ信号も立ち上がる。

#### 【1973】

<復帰コマンド設定送信処理>

#### 【1974】

次に、図265を用いて、上述の主制御部タイマ割り込み処理における復帰コマンド設定送信処理(ステップS9319)について説明する。なお、同図は復帰コマンド設定送信処理の流れを示すフローチャートである。

#### 【1975】

ステップS9901では、復電時であるか否かを判定し、復電時であればステップS9902に進み、そうでなければステップS9903に進む。

#### 【1976】

ステップS9902では、出力予定情報記憶領域に記憶した出力予定情報に復電を示す情報を設定してステップS9904に進む。

#### 【1977】

ステップS 9 9 0 3では、出力予定情報記憶領域に記憶した出力予定情報にR A Mクリアを示す情報を設定してステップS 9 9 1 3に進む。

【1 9 7 8】

ステップS 9 9 0 4では、大当りの表示確定後であるか否かを判定し、大当りの表示後確定後でなければステップS 9 9 0 5に進み、大当りの表示後確定後であればステップS 9 9 0 9に進む。

【1 9 7 9】

ステップS 9 9 0 5～ステップS 9 9 0 8では、出力予定情報記憶領域に記憶した出力予定情報に、遊技状態、特図作動状態、特図抽選結果、保留している特図変動遊技の数および確変抽選結果を追加する。本実施例では、現在の遊技状態に基づいて、通常状態を示す情報、時短状態を示す情報、確変状態を示す情報のいずれかを遊技状態として追加する。また、特図変動遊技が作動している場合には、特図作動状態として作動中を示す情報を追加し、特図変動遊技が作動していない場合には、特図作動状態として停止中を示す情報を追加する。また、大当りフラグをオンに設定している場合には、特図抽選結果として大当りを示す情報を追加し、大当りフラグをオフに設定している場合には、特図抽選結果としてハズレを示す情報を追加する。また、確変フラグをオンに設定している場合には、確変抽選結果として確変を示す情報を追加し、確変フラグをオフに設定している場合には、確変抽選結果としてハズレを示す情報を追加する。

【1 9 8 0】

ステップS 9 9 0 9～ステップS 9 9 1 2では、出力予定情報記憶領域に記憶した出力予定情報に、大当り中か否かを示す情報、特図保留数情報、大入賞口開放回数、確変フラグの値および大入賞口開閉状態を追加する。本実施例では、大当りフラグをオンに設定している場合には大当りを示す情報を追加する。また、特図保留数情報として、特図遊技の保留数（本実施例では0～4）に対応する保留数データを追加する。また、大入賞口開放回数として、可変入賞口2 3 4を開放する回数を示す情報を追加する。また、可変入賞口2 3 4が開放中の場合には、大入賞口開閉状態として可変入賞口2 3 4が開放中であることを示す情報を追加し、可変入賞口2 3 4が閉鎖中の場合には、大入賞口開閉状態として大入賞口が閉鎖中であることを示す情報を追加する。

【1 9 8 1】

ステップS 9 9 1 3では、上述のコマンド送信処理を行う。

【1 9 8 2】

<払出制御部リセット割り込み処理>

【1 9 8 3】

次に、図2 6 6を用いて、払出制御部6 0 0のC P Uが実行する払出制御部リセット割り込み処理について説明する。なお、同図は払出制御部リセット割り込み処理の流れを示すフローチャートである。

【1 9 8 4】

払出制御部6 0 0には、電源が投入されるとリセット信号を出力するリセット信号出力回路を設けている。このリセット信号を入力した基本回路のC P Uは、リセット割り込みによりリセットスタートしてR O Mに予め記憶している制御プログラムに従って処理を実行する。

【1 9 8 5】

ステップS 1 1 0 0 1では、初期設定1を行う。この初期設定1では、C P Uのスタックポインタ（S P）へのスタック初期値の設定等を行う。

【1 9 8 6】

ステップS 1 1 0 0 2では、低電圧信号がオンであるか否か、すなわち、電圧監視回路が、電源制御部6 6 0から払出制御部6 0 0に供給している電源の電圧値が所定の値（本実施例では9 V）未満である場合に電圧が低下したことを示す低電圧信号を出力しているか否かを監視する。そして、低電圧信号がオンの場合（電源の遮断を検知した場合）にはステップS 1 1 0 0 1の処理を繰り返し実行し、低電圧信号がオフの場合（電源の遮断を

10

20

30

40

50

検知していない場合)にはステップS 1 1 0 0 3に進む。

【1 9 8 7】

ステップS 1 1 0 0 3では、初期設定2を行う。この初期設定2では、後述する払出制御部タイマ割り込み処理を定期毎に実行するための周期を決める数値をカウンタ・タイマに設定する処理、RAMへの書き込みを許可する設定、I/Oポートの初期設定等を行う。

【1 9 8 8】

ステップS 1 1 0 0 4では、電源の遮断前(電断前)の状態に復帰するか否かの判定を行い、電断前の状態に復帰しない場合(パチンコ機100を初期状態にする場合)にはステップS 1 1 0 0 6に進み、電断前の状態に復帰する場合にはステップS 1 1 0 0 5に進む。

10

【1 9 8 9】

具体的には、最初に、電源基板に設けた操作部を遊技店の店員などが操作した場合に送信されるRAMクリア信号がオン(操作があったことを示す)であるか否か、すなわちRAMクリアが必要であるか否かを判定し、RAMクリア信号がオンの場合(RAMクリアが必要な場合)には、パチンコ機100を初期状態にすべくステップS 1 1 0 0 6に進む。一方、RAMクリア信号がオフの場合(RAMクリアが必要でない場合)は、RAM408に設けた電源ステータス記憶領域に記憶した電源ステータスの情報を読み出し、この電源ステータスの情報がサスペンドを示す情報であるか否かを判定する。そして、電源ステータスの情報がサスペンドを示す情報でない場合には、パチンコ機100を初期状態にすべくステップS 1 1 0 0 6に進み、電源ステータスの情報がサスペンドを示す情報である場合には、RAMの所定の領域(例えば全ての領域)に記憶している1バイトデータを初期値が0である1バイト構成のレジスタに全て加算することによりチェックサムを算出し、算出したチェックサムの結果が特定の値(例えば0)であるか否か(チェックサムの結果が正常であるか否か)を判定する。そして、チェックサムの結果が0の場合(チェックサムの結果が正常である場合)には電断前の状態に復帰すべくステップS 1 1 0 0 5に進み、チェックサムの結果が0以外である場合(チェックサムの結果が異常である場合)には、パチンコ機100を初期状態にすべくステップS 1 1 0 0 6に進む。同様に電源ステータスの情報が「サスペンド」以外の情報を示している場合にもステップS 1 1 0 0 6に進む。

20

30

【1 9 9 0】

ステップS 1 1 0 0 5では、復電時処理を行う。この復電時処理では、RAMの記憶領域のうち、復電時にクリアすべき記憶領域(コマンドを格納するためのコマンドバッファ、エラー状態を記憶するためのエラーステータスなどを除く記憶領域)の初期化などを行う。

【1 9 9 1】

ステップS 1 1 0 0 6では、初期化処理を行う。この初期化処理では、割り込み禁止の設定、スタックポインタへのスタック初期値の設定、RAMの所定の領域(例えば、全ての記憶領域)の初期化などを行う。

【1 9 9 2】

ステップS 1 1 0 0 7では、初期設定3を行う。この初期設定3では、RAMに設けたエラーステータス記憶領域に記憶したエラーステータスのうち、不正払出エラーと払出超過エラー以外の情報をクリアしたり、割り込み許可の設定などを行う。

40

【1 9 9 3】

ステップS 1 1 0 0 8では、主制御部300から入力したデータの中に未解析データがあるか無いかを判定し、未解析データがある場合にはステップS 1 1 0 0 9でコマンド解析処理(詳細は後述する)を行い、未解析データがない場合にはステップS 1 1 0 1 0に進む。

【1 9 9 4】

ステップS 1 1 0 1 0では、低電圧信号がオフであるか否かを監視し、低電圧信号がオ

50

フの場合（電源の遮断を検知していない場合）にはステップ S 1 1 0 0 8 に戻り、低電圧信号がオンの場合（電源の遮断を検知した場合）にはステップ S 1 1 0 1 1 に進む。

#### 【1995】

ステップ S 1 1 0 1 1 では、電断時処理を行う。この電断時処理では、R A M に設けたスタックポインタ退避領域に現在のスタックポインタの値を記憶し、上述の電源ステータス記憶領域にサスペンドを示す情報を設定する。また、R A M の所定の領域（例えば全ての領域）に記憶している 1 バイトデータを初期値が 0 である 1 バイト構成のレジスタに全て加算し、チェックサム算出用数値記憶領域に記憶している値からその加算した結果を減算した値をチェックサム（電断時チェックサム）として算出し、算出した電断時チェックサムを上述のチェックサム算出用数値記憶領域に記憶し、R A M への書き込みを禁止する設定などを行う。

10

#### 【1996】

ステップ S 1 1 0 1 2 では、低電圧信号がオンであるか否かを監視し、低電圧信号がオンの場合（電源の遮断を検知した場合）にはステップ S 1 1 0 1 2 の処理を繰返し実行して低電圧信号がオフになるのを待ち、低電圧信号がオフの場合（電源の遮断を検知していない場合）にはステップ S 1 1 0 0 1 に戻り、払出制御部リセット割り込み処理を最初から開始する。すなわち、このステップ S 1 1 0 1 2 では、ステップ S 1 1 0 1 0 で低電圧信号を入力した後（電源の遮断を検知した後）で、この低電圧信号の出力が停止したことを検知した場合（電源の復帰を検知した場合）に払出制御回路を初期化する初期化処理を行う。

20

#### 【1997】

<コマンド解析処理>

#### 【1998】

次に、図 2 6 7 を用いて、上述の払出制御部リセット割り込み処理におけるコマンド解析処理（ステップ S 1 1 0 0 9）について説明する。なお、同図はコマンド解析処理の流れを示すフローチャートである。

#### 【1999】

ステップ S 1 1 1 0 1 では、R A M に設けた賞球数記憶領域に未解析データとして記憶した賞球数（払出要求数）と、R A M に設けた今回加工種別記憶領域に未解析データとして記憶した加工種別を取得する。

30

#### 【2000】

ステップ S 1 1 1 0 2 では、ステップ S 1 1 1 0 1 で取得した今回加工種別、および R A M に設けた前回加工種別記憶領域から取得した、コマンド解析処理で前回に使用した前回加工種別の両者を比較する。そして、今回加工種別と前回加工種別が同一であるか否かを判定し、両者が同一の場合は処理を終了し、両者が異なる場合はステップ S 1 1 1 0 3 に進む。

#### 【2001】

ステップ S 1 1 1 0 3 では、今回加工種別の値を、前回加工種別として前回加工種別記憶領域に記憶する。

#### 【2002】

ステップ S 1 1 1 0 4 では、上述の図 2 5 6（e）の復元データテーブルを参照し、今回加工種別に基づいて復元データを選択した後、復元データを用いて賞球数の加工を解除する。本実施例では、加工された賞球数と復元データの排他的論理和（E X O R）を算出し、その算出結果の下位 4 ビットを賞球数として賞球数記憶領域に記憶する。例えば、加工された賞球数が 3（0 0 0 0 0 0 1 1 B）、復元データが 0 0 H（0 0 0 0 0 0 0 0 B）の場合、排他的論理和の算出結果は 0 3 H（0 0 0 0 0 0 1 1 B）であるから、賞球数はその下位 4 ビットの 3 である。また、加工された賞球数が 1 0 H（0 0 0 1 0 0 0 0 B）、復元データが 0 3 H（0 0 0 0 0 0 1 1 B）の場合、排他的論理和の算出結果は 1 3 H（0 0 0 1 0 0 1 1 B）であるから、賞球数はその下位 4 ビットの 3 であり、加工された賞球数が 2 D H（0 0 1 0 1 1 0 0 B）、復元データが 0 E H（0 0 0 0 1 1 1 0 B）

40

50



の場合、排他的論理和の算出結果は23H(00100011B)であるから、賞球数はその下位4ビットの3であり、加工された賞球数が3AH(00111010B)、復元データが09H(00001001B)の場合、排他的論理和の算出結果は33H(00110011B)であるから、賞球数はその下位4ビットの3である。

【2003】

ステップS11105では、RAMに設けた次賞球要求数記憶領域に記憶した次賞球要求数に、ステップS11104で取得した賞球数を加算して処理を終了する。

【2004】

＜コマンド受信割り込み処理＞

【2005】

次に、図268を用いて、払出制御部600のCPUが実行するコマンド受信割り込み処理について説明する。なお、同図はコマンド受信割り込み処理の流れを示すフローチャートである。

【2006】

払出制御部600は、主制御部300が出力するストロブ信号の立ち下がりエッジを検出した場合に、このコマンド受信割り込み処理を実行する。主制御部300でストロブ情報を含めた払出要求信号を出力すると、それまでHighレベルを保持していた払出制御部600のストロブ信号がLowレベルに立ち下がるような回路構成にしている。

【2007】

ステップS11201では、受信割り込みスタート処理を行う。この受信割り込みスタート処理では、CPUの各レジスタの値をスタック領域に一時的に退避する処理などを行う。

【2008】

ステップS11202では、主制御部300から入力する払出要求情報をI/Oポートから入力し、ステップS11203では、払出要求情報があるか無いかを判定する。そして、払出要求情報がある場合にはステップS11204に進み、払出要求情報がない場合にはステップS11209に進む。

【2009】

ステップS11204では、払出要求情報のストロブ情報があるか無いか(オンかオフか)を判定し、ストロブ情報がある場合にはステップS11205に進み、ストロブ情報が無い場合にはステップS11209に進む。

【2010】

ステップS11205では、ステップS11202で入力した払出要求情報から加工種別を取得する。

【2011】

ステップS11206では、払出要求情報に初期化情報があるか無いか(オンかオフか)を判定する。そして、初期化情報がある場合にはステップS11208に進み、初期化情報が無い場合にはステップS11207に進む。

【2012】

ステップS11207では、払出要求情報から賞球数を取得し、上述の賞球数記憶領域に記憶するとともに、RAMに設けた今回加工種別記憶領域に、払出要求情報から取得した加工種別を未解析データとして記憶する。

【2013】

ステップS11208では、ステップS11205で取得した加工種別を、上述の前回加工種別記憶領域に記憶する。

【2014】

ステップS11209では、受信割り込みエンド処理を行う。この受信割り込みエンド処理は、ステップS11201で一時的に退避した各レジスタの値を元の各レジスタに設定したり、割り込み許可の設定などを行う。

【2015】

10

20

30

40

50

< 払出制御部タイマ割り込み処理 >

【2016】

次に、図269を用いて、払出制御部600のCPUが実行する払出制御部タイマ割り込み処理について説明する。なお、同図は払出制御部タイマ割り込み処理の流れを示すフローチャートである。

【2017】

払出制御部600は、所定の周期（本実施例では1msに1回）でタイマ割り込みを発生するカウンタ・タイマ412を備えており、このタイマ割り込みを契機として払出制御部タイマ割り込み処理を所定の周期で開始する。

【2018】

ステップS11301では、タイマ割り込みスタート処理を行う。このタイマ割り込みスタート処理では、CPUの各レジスタの値をスタック領域に一時的に退避する処理などを行う。

【2019】

ステップS11302では、ポート入力管理処理を行う。このポート入力管理処理では、I/Oポートの値を取得して、各種センサの状態などを検出する。

【2020】

ステップS11303では、タイマ更新管理処理を行う。このタイマ更新管理処理では、払出報知用LEDの点灯／消灯時間、モータ駆動／非駆動時間などを計時するためのタイマなどを含む各種タイマを更新する。

【2021】

ステップS11304では、エラー管理処理を行う。このエラー管理処理では、I/Oポートに入力する皿満杯信号を検出して皿満杯信号がオンであるか否かを判定し、皿満杯信号がオンの場合（下皿126が球で一杯になっている場合）には、RAMに設けたエラーステータス記憶領域に皿満杯エラーを示す情報を記憶し、皿満杯信号がオフの場合（下皿150に球を貯留する空きができた場合）には、エラーステータス記憶領域に皿満杯エラーの解除を示す情報を記憶する。また、主制御部300と払出制御部600との間の通信回線が断線などにより通信可能かどうかを検出し、通信可能な場合には、RAMに設けたエラーステータス記憶領域に通信可能であることを示す情報を、また通信不可能な場合には通信不可能であることを示す情報を記憶する。また、エラー管理処理では、I/Oポートに入力するエラー解除スイッチ信号を検出してエラー解除信号がオンであるか否かを判定し、エラー解除信号がオンである場合には、エラーステータス記憶領域に記憶している不正払出エラーの情報、または、払出超過エラーの情報を初期化して、これらのエラーを解除する。

【2022】

ステップS11305では、CRユニット通信管理処理を行う。このCRユニット通信管理処理では、CRインターフェース部606から遊技媒体貸出信号を受信して遊技媒体貸出信号センサ信号がオンであるか否かを判定し、遊技媒体貸出信号がオンの場合（CRインターフェース部606からの球貸要求を入力した場合）には、RAMに設けた遊技媒体貸出情報記憶領域に遊技媒体の貸出要求があったことを示す情報を記憶する。

【2023】

ステップS11306では、払出管理処理を行う。この払出管理処理では、上述のエラーステータス記憶領域から不正払出エラーの情報、および払出超過エラーの情報を読み出し、いずれのエラーも発生していない場合に、センサ回路を介して入力する払出センサ604の信号（以下、払出センサ信号と称する場合がある）に基づいて払出個数の監視を行う。すなわち、所定のエラー（ここでは、いずれかのエラー）が発生している場合にはモータの駆動、すなわち払出装置からの賞媒体（例えば遊技球）の払出を停止するようにしている。具体的には、払出センサ604の信号を検出して払出センサ信号がオンであるか否かを判定し、払出センサ信号がオンの場合（払出センサを球が通過した場合）には払出完了数チェックに1を加算して払出完了数チェック記憶領域に記憶する。また、賞球およ

10

20

30

40

50

び貸出球の要求が無いときに払出センサ信号がオンになった場合には、上述のエラーステータス記憶領域に不正払出エラーを示す情報を設定し、賞球数または貸出球数が各々の要求数を超え、その超過数が所定数以上になった場合には、上述のエラーステータス記憶領域に払出超過エラーを示す情報を設定する。

#### 【2024】

また、上述のエラーステータス記憶領域から皿満杯エラーの情報、不正払出エラーの情報、および払出超過エラーの情報を読み出し、いずれのエラーも発生していない場合に、払出開始監視処理、初期位置検索動作処理、通常払出動作処理、リトライ動作処理、逆回転動作処理のいずれかの処理を行う。

#### 【2025】

払出開始監視処理では、貸出要求数、および賞球要求数が0であり、次賞球要求数が0以外の場合には、賞球要求数に次賞球要求数をセットし、次賞球要求数をクリアする。また、スプロケットを駆動する払出モータ602の位置が不確定の場合（動作モードが初期位置検索動作モードの場合）には、払出完了数チェックから1を減算して払出完了数チェック記憶領域に記憶し、スプロケットを駆動する払出モータ602の位置が確定している場合（動作モードが通常払出動作モードの場合）には、払出完了数チェックとして払出完了数チェック記憶領域に0を設定する。また、賞球要求数を、スプロケットの払出モータ602を駆動する量（モータ駆動量）に変換し、これをRAMに設けたモータ駆動量記憶領域に記憶すると共に、RAMに設けたモータ制御データテーブルを参照してモータ駆動量に対応するモータ駆動制御データを選択し、正転を示すモータ駆動制御データをI/Oポートを介してモータ制御回路に出力する。これにより、モータ制御回路はスプロケットのモータの励磁位置を所定回変化してスプロケットを正方向に回転駆動する。

#### 【2026】

初期位置検索動作処理および通常払出動作処理では、モータの駆動終了後に、払出完了数チェック記憶領域から払出完了チェックを読み出し、払出完了チェックが0の場合には、払出開始監視処理を実行する準備を行い、払出完了チェックが0以外の場合には、エラーステータス記憶領域に払出装置エラーを示す情報を設定すると共に、リトライ動作処理を実行する準備を行う。

#### 【2027】

リトライ動作処理では、所定の時間が経過するのを待ち（リトライ動作開始待ちタイマが0になるのを待ち）、リトライ動作開始待ちタイマが0になった場合には、逆回転動作処理を実行する準備を行う。逆回転動作処理では、上述のモータ制御データテーブルを参照してモータ駆動量に対応するモータ駆動制御データを選択し、逆転を示すモータ駆動制御データをI/Oポートを介してモータ制御回路に出力する。これにより、モータ制御回路はスプロケットの払出モータ602の励磁位置を所定回変化してスプロケットを逆回転駆動する。また、逆回転動作処理では、モータの駆動終了後に払出開始監視処理を実行する準備を行う。

#### 【2028】

ステップS11307では、モータ駆動管理処理を行う。このモータ駆動管理処理では、駆動開始監視処理、加速駆動処理、定速駆動処理、ブレーキ駆動処理、駆動終了処理のいずれかの処理を行う。

#### 【2029】

駆動開始監視処理では、上述のエラーステータス記憶領域から皿満杯エラーの情報、不正払出エラーの情報、および払出超過エラーの情報を読み出し、いずれのエラーも発生していない場合に、上述のモータ制御データテーブルを参照してモータ駆動量に対応するモータ駆動制御データを選択し、正転を示すモータ駆動制御データをI/Oポートを介してモータ制御回路に出力する。これにより、モータ制御回路はスプロケットの払出モータ602の励磁位置を所定回変化してスプロケットを正方向に回転駆動する。

#### 【2030】

加速駆動処理および定速駆動処理では、スプロケットが初期位置検索動作中、または、

10

20

30

40

50

逆回転動作中の場合を除き、モータの励磁位置を16回変化させるごとに払出完了数チェックから1を減算して払出完了数チェック記憶領域に記憶する。また、更新後の払出完了数チェックが-4未満になった場合には、ブレーキ駆動処理を実行する準備を行う。さらに、上述の遊技媒体貸出情報記憶領域から遊技媒体貸出情報を読み出して、遊技媒体の貸出要求があったことを示す情報の有無を判定し、遊技媒体の貸出要求があったことを示す情報がある場合（賞球の払出中にC R インターフェース部606からの球貸要求を入力した場合）にも、ブレーキ駆動処理を実行する準備を行う。

#### 【2031】

ブレーキ駆動処理では、所定の時間が経過するのを待ち（モータ駆動管理タイマが0になるのを待ち）、モータ駆動管理タイマが0になった場合には、駆動終了処理を実行する準備を行い、駆動終了処理では、モータ駆動の後処理を行う。

10

#### 【2032】

ステップS11308では、LED管理処理を行う。このLED管理処理では、エラーステータス記憶領域の不正払出エラー情報が不正払出エラーが発生中であることを示している場合には、不正払出エラーが発生していることを遊技者に報知するためのLEDを点灯させるとともに、不正払出エラーが発生していないことを示している場合にはそのLEDを消灯させる。また、エラーステータス記憶領域の払出超過エラー情報が払出超過エラーが発生中であることを示している場合には、払出超過エラーが発生していることを遊技者に報知するためのLEDを点灯させるとともに、払出超過エラーが発生していないことを示している場合にはそのLEDを消灯させる。

20

#### 【2033】

ステップS11309では、信号出力管理処理を行う。この信号出力管理処理では、RAMに記憶している遊技情報（例えば払出センサ信号を入力するたびに出力する賞球信号）を、情報出力回路（図示せず）を介してパチンコ機100とは別体の情報入力回路350に出力する。

#### 【2034】

ステップS11310では、タイマ割り込みエンド処理を行う。このタイマ割り込みエンド処理では、ステップS11301で一時的に退避した各レジスタの値を元の各レジスタに設定したり、割り込み許可の設定などを行う。

#### 【2035】

以上説明したように、本実施例1に係るパチンコ機100は、遊技媒体を所定の遊技領域124内で転動させる遊技の制御用に動作するCPU304を含む制御回路（本実施例では基本回路）302と、CPU304による演算結果を記憶する演算記憶部（本実施例ではRAM）308と、起動信号を制御回路302に出力する起動信号出力回路340と、特定の周期で定期信号を制御回路302に出力する定期信号出力回路（本実施例ではカウンタ・タイマ）312と、遊技領域124内の所定の位置を遊技媒体が通過した場合に、検出信号を出力する検出回路（本実施例では球検出センサ）と、を備え、定期信号を入力した場合に開始する複数の定期処理（本実施例では主制御部タイマ割り込み処理内の各処理）と、複数の定期処理のうちの一つの処理であり、検出回路からの検出信号の有無を監視し、演算記憶部308に記憶する検出監視処理（本実施例では入力ポート状態更新処理）と、複数の検出監視処理における監視結果に基づいて、遊技領域124内の所定の位置を遊技媒体が通過したと判定する通過判定処理と、所定の時期（本実施例では電源投入後）から特定の周期ごとに開始させた複数の検出監視処理における監視結果を演算記憶部308に記憶するまで、通過判定処理による判定結果を用いる処理を行わずに後続の処理に分岐させる処理と、を制御回路302に含むことを特徴とする、遊技台である。

30

40

#### 【2036】

換言すれば、本実施例1に係るパチンコ機100は、遊技動作のプログラムを記憶するプログラム記憶部（本実施例ではROM）306、および起動信号を入力した場合に前記プログラムを読み出して動作するCPU304を含む制御回路302と、CPU304による演算結果を記憶する演算記憶部308と、起動信号を制御回路302に出力する起

50

動信号出力回路 338 と、特定の周期で定期信号を制御回路 302 に出力する定期信号出力回路 312 と、遊技領域 124 内の所定の位置を遊技媒体が通過した場合に、検出信号を出力する検出回路と、を備え、定期信号を入力した場合に開始する複数の定期処理と、複数の定期処理のうちの一つの処理であり、検出回路からの検出信号の有無を監視し、演算記憶部 308 に記憶する検出監視処理と、複数回の検出監視処理における監視結果に基づいて、遊技領域 124 内の所定の位置を遊技媒体が通過したと判定する通過判定処理と、所定の時期から特定の周期ごとに開始させた複数回の検出監視処理における監視結果を演算記憶部 308 に記憶するまで、通過判定処理による判定結果を用いる処理を行わずに後続の処理に分岐させる処理と、を前記プログラムに含むことを特徴とする、遊技台である。

10

#### 【2037】

本実施例 1 に係るパチンコ機 100 によれば、遊技の進行中の電源異常による電断・復電が原因の誤動作の発生を防ぐことができる場合があり、また、遊技台にノイズを印加して制御回路を暴走させ、不当な利益を得るような不正行為を未然に防止することができる場合がある。検出監視処理における監視結果を演算記憶部に記憶するまで、通過判定処理による判定結果を用いる処理を行わないように構成しているので、所定の時期から通過判定処理による判定結果を用いる処理の開始までの期間を最短にできるとともに、通過判定に必要な情報は、確実に所定の時期からの情報を使用することができるので、遊技台に電源が投入されているにも関わらず、遊技制御を行っていない期間を最低限にして、遊技を中断している遊技者の不快感および遊技台の稼働の低下を減少することが可能になる場合がある。また品質の高い情報を用いた遊技制御を行うことができる場合がある。

20

#### 【2038】

また、起動信号を入力した場合に、遊技媒体を所定の遊技領域内で転動させる遊技の制御用に動作する CPU を含む制御回路と、前記 CPU による演算結果を記憶する演算記憶部と、前記起動信号を前記制御回路に出力する起動信号出力回路と、特定の周期で定期信号を前記制御回路に出力する定期信号出力回路と、前記遊技領域内の所定の位置を遊技媒体が通過した場合に、検出信号を出力する検出回路と、を備え、前記定期信号を入力した場合に開始する複数の定期処理と、前記複数の定期処理のうちの一つの処理であり、前記検出回路からの検出信号の有無を監視し、前記演算記憶部に記憶する検出監視処理と、複数回の前記検出監視処理における監視結果に基づいて、前記遊技領域内の所定の位置を遊技媒体が通過したと判定する通過判定処理と、前記起動信号を入力してから、前記特定の周期ごとに開始させた前記複数回の前記検出監視処理における監視結果を前記演算記憶部に記憶するまで、前記通過判定処理による判定結果を用いる処理を行わずに後続の処理に分岐させる処理と、を前記制御回路に含む構成とすれば、検出監視処理における監視結果を演算記憶部に記憶するまで、通過判定処理による判定結果を用いる処理を行わないように構成しているので、起動信号を入力した時期から通過判定処理による判定結果を用いる処理の開始までの期間を最短にできるとともに、通過判定に必要な情報は、確実に起動信号を入力した時期からの情報を使用することができるので、遊技台に電源が投入されているにも関わらず、遊技制御を行っていない期間を最低限にして、遊技を中断している遊技者の不快感および遊技台の稼働の低下を減少することが可能になる場合がある。また品質の高い情報を用いた遊技制御をおこなうことができる場合がある。

30

40

#### 【2039】

また、電源回路（本実施例では電源制御部）660 から制御回路 302 に供給している電圧が特定の値未満である場合に、低電圧信号を出力する電圧監視回路 338 と、電源回路 660 から CPU 304 への電力供給を停止している期間中も演算記憶部 308 に電力を供給するバックアップ電源回路（本実施例では蓄電装置）と、を更に備え、所定の時期（本実施例では電源投入後）から、複数の定期処理を開始させた回数が特定の回数に到達するまでの期間中に低電圧信号を入力した場合に、制御回路 302 の設定のうちの所定の第 1 の情報を演算記憶部 308 に記憶する第 1 の電断時処理（本実施例では電断時処理 1）と、起動信号を入力してから、複数の定期処理を開始させた回数が特定の回数に到達し

50

た後に低電圧信号を入力した場合に、制御回路302の設定のうちの第1の情報とは別の第2の情報を演算記憶部308に記憶する第2の電断時処理（本実施例では電断時処理2）と、起動信号を入力した場合に、第1電断時処理または第2の電断時処理が記憶させた情報に基づいて、制御回路302の設定の一部を更新する復電時処理と、を制御回路302に含むように構成すれば、動作が安定する前に記憶した信頼性の低い情報に基づいて遊技制御を行うことがなく、遊技の進行中の電源異常による電断・復電が原因の誤動作の発生をより確実に防ぐことができる場合がある。所定の条件が成立しているかどうかによって、異なる電断時処理のうちの1つを実行するように構成しているので、柔軟性の高い遊技制御を行うことができる。例えば遊技状態が第1の状態（例えば大当たり中、エラー発生中など）には所定の電断時処理（例えばRAM308の全ての領域に関するチェックサムの取得、電源ステータスの格納の実行、スタックポインタへのレジスタ退避）を行い、遊技状態が第2の状態（例えば図柄の変動開始を一切保留しておらず、図柄の変動を行っていない状態で、なおかつ入賞口への入賞の記憶が全くない）の場合には、該所定の電断時処理を実行しない、または一部の電断時処理を省略するようにできる。こうすることで、不要な場合には電断時処理を素早く終了することが出来るようになり、供給される電力が不安定な中での制御部、特に基本回路の動作を少なくすることで、遊技台が予測していない動作を行うといった不具合（例えば暴走、RAM破壊）をより抑制することができる場合がある。ここで所定の条件の成立として、上述の起動信号を入力してから、上述の複数の定期処理を開始させた回数が特定の回数に到達した後に前記低電圧信号を入力した場合を適用するのが好適である。

10

20

#### 【2040】

なお、上述の実施例における各種記憶領域は必ずしもRAMに設ける必要はなく、例えば、情報を一時的に記憶すれば足りるような場合には各制御部のCPUのレジスタなどに記憶してもよい。

#### 【2041】

また、遊技状態格納領域には、単一の情報を記憶可能なだけでなく、複数の遊技状態に関する情報を並列的に記憶可能に構成してもよい。すなわち、複数単位の記憶領域により構成してもよい。

#### 【2042】

また、処理は1つの命令、複数の一連の命令、および一連ではない複数の命令などを示し、サブルーチン全体、分岐から復帰までの一連の命令に限定しているものではない。

30

#### 【2043】

また、ステップS9101では、WDT314の初期値として32.8msに相当する数値を設定したが、WDT314に設定する初期値はこれに限定されるものではない。また、ステップS9104では、主制御部タイマ割り込みの割り込み周期として約2msに相当する数値を設定し、ステップS11003では、払出制御部タイマ割り込みの割り込み周期として約2msに相当する数値を設定したが、タイマ割り込みの周期はこれに限定されるものではない。

#### 【2044】

また、ステップS9201では、機能限定タイマの初期値として3を設定したが、本発明はこれに限定されず、例えば、主制御部タイマ割り込みの周期を1msに設定し、機能限定タイマの初期値として6を設定すれば、機能限定ウェイト処理において初期値を設定し、且つ、主制御部タイマ割り込みの割り込みを許可してから、最大で6ms（＝1ms（主制御部タイマ割り込みの周期）×6）後、最小で5ms（＝1ms（主制御部タイマ割り込みの周期）×5）後に機能限定を解除することになり、機能限定解除の時間の誤差（ゆらぎ）を小さくすることができる。

40

#### 【2045】

また、ステップS9303の入力ポート状態更新処理では、複数回の監視結果に基づいて、遊技領域124内の所定の位置を遊技媒体が通過したと判定するように構成したが、本発明はこれに限定されず、この判定処理を行わなくてもよい。この場合、上述の実施例

50

では、ステップ S 9 3 0 7 において制御状態を機能限定状態であると判定すれば、入力ポート状態更新処理による判定結果を用いる処理を行わずに後続の処理に分岐させることができる。

#### 【2046】

また、ステップ S 9 3 1 0 の入賞受付処理では、前々回検出信号記領域、前回検出信号記領域、および今回検出信号記領域の各記憶領域に記憶した各々の球検出センサの検出信号の有無の情報を比較し、各々の球検出センサにおける過去 3 回分の検出信号の有無の情報が一致するか否かを判定したが、いずれかの検出信号の有無の情報をマスクしてもよく、例えば、前回検出信号記領域、および今回検出信号記領域の各記憶領域に記憶した各々の球検出センサの検出信号の有無の情報を比較し、各々の球検出センサにおける過去 2 回分の検出信号の有無の情報が一致するか否かを判定してもよい。

10

#### 【2047】

また、主制御部タイマ割り込み処理では、ステップ S 9 3 0 7 において制御状態が機能限定状態ではないと判定するまで、入力ポート状態更新処理による判定結果を用いる処理を行わずに後続の処理に分岐させるように構成したが、スキップする処理はステップ S 9 3 0 8 ～ステップ S 9 3 1 7 の処理に限定されるものではなく、例えば、制御状態が機能限定状態の場合にステップ S 9 3 1 7 の外部出力信号設定処理を行ってもよい。

#### 【2048】

また、ステップ S 9 4 1 1 では、払出要求数と加工データの排他的論理和 (E X O R) を算出することで払出要求情報を加工を行ったが、例えば、払出要求数と加工データの論理積 (A N D) や論理和 (O R) を算出することで払出要求情報を加工を行ってもよく、この場合、払出制御部では否定論理積 (N A N D) や否定論理和 (N O R) を用いて加工の解除を行えばよい。

20

#### 【2049】

また、タイマ更新処理では、タイマ 1 ～タイマ 9 の 9 種類のタイマを約 2 m s 毎に更新し、タイマ 1 0 を約 1 0 m s 毎に更新したが、本発明はこれに限定されず、例えば、タイマ制御データテーブルの 1 2 行目のタイマ更新周期を 1 0 に設定すれば、タイマ 1 ～タイマ 9 の 9 種類のタイマを約 2 m s 毎に更新し、タイマ 1 0 を約 2 0 m s 毎に更新することができる。また、タイマ制御データテーブルの 2 行目のタイマ個数を 5 に設定し、その次にタイマ 1 ～5 の格納アドレスを記憶し、その次にタイマ更新周期として 5、タイマ個数として 5 を記憶し、その次にタイマ 6 ～1 0 の格納アドレスを記憶すれば、タイマ 1 ～タイマ 5 の 5 種類のタイマを約 2 m s 毎に更新し、タイマ 6 ～1 0 の 5 種類のタイマを約 1 0 m s 毎に更新することもできる。

30

#### 【2050】

また、実施例では、電源回路 6 6 0 から遊技制御回路 3 0 2 に供給している電圧が特定の値未満である場合に、第 1 の低電圧信号を出力する第 1 の電圧監視回路 3 3 8 と、電源回路 6 6 0 から払出制御回路 6 0 0 に供給している電圧が特定の値未満である場合に、第 2 の低電圧信号を出力する第 2 の電圧監視回路の 2 つの電圧監視回路を設けたが、本発明はこれに限定されず、電源回路 6 6 0 から遊技制御回路 3 0 2 および払出制御回路 4 0 2 に供給している電圧の両者を 1 つの電圧監視回路で監視し、第 1 の低電圧信号または／および第 2 の低電圧信号を出力するように構成してもよい。

40

#### 【2051】

また、実施例では、主制御部 3 0 0 に電圧監視回路 3 3 8 を 1 つだけ設け、また払出制御部 6 0 0 に電圧監視回路を 1 つだけ設けているが、これに限定されず、所定の制御部 (主制御部 3 0 0、払出制御部 6 0 0 など) に電圧監視回路を複数設け、それら複数の電圧監視回路のそれぞれが別々の電力線を監視し、これら複数の電力線のうちの 1、複数または全部の電力線の電圧値が所定の値未満になったことを検出した場合に、低電圧信号を該所定の制御部に設けた基本回路に出力するようにしてもよい。このように構成することで、所定の制御部による制御を確実におこなうことができ、遊技台が予測していない動作を行うといった不具合 (例えば暴走、R A M 破壊) を抑制することができる場合がある。こ

50

ここで複数の電力線のうち1、複数または全部の電力線が同じ電圧で電力を配電するようにしてもよいし、全てが別の値の電圧となるように電力を配電するようにしてもよい。

#### 【2052】

また、実施例では、所定の制御部（主制御部300、払出制御部600）の基本回路（302、402）で動作する処理のうち、遊技制御の開始を待機させる処理（ステップS9103、ステップS11002）が遊技制御の開始を待機させる条件としている低電圧信号がオンであること、および遊技制御を終了させる処理を実行するかどうかの判定処理（ステップS9323、ステップS11010）で、該遊技制御を終了させる処理を実行する条件としている低電圧信号がオンであることの2つに関して、同一の低電圧信号を用いているが、これに限定されない。例えば、所定の制御部の所定の電力線を複数の電圧監視回路で監視し、第1の監視回路は第1の電圧値よりもその所定の電力線の電圧値が低い場合に第1の低電圧信号を、また複数の電圧監視回路のうち第1の電圧監視回路とは別の第2の電圧監視回路は第1の電圧値とは別の第2の電圧値よりもその所定の電力線の電圧値が低い場合に第2の低電圧信号を、それぞれ所定の制御部の基本回路に出力するように構成し、上述の遊技制御の開始を待機させる条件を第1の低電圧信号がオン（第1の電圧監視回路が第1の低電圧信号を出力している状態）であること、また上述の該遊技制御を終了させる処理を実行する条件を第2の低電圧信号がオン（第2の電圧監視回路が第2の低電圧信号を出力している状態）であること、などとそれぞれの条件を異ならせても良い。この場合第1の監視回路と第2の監視回路は同一基体内に集積するように構成してもよい。これらの場合、第1の電圧値と第2の電圧値との関係は、第1の電圧値を第2の電圧値よりも高くして適用した場合であれば、遊技制御の開始を所定の制御部に供給される電力の電圧値が十分に高い電圧値になるまで待機するので、所定の制御部、特に基本回路が不安定な動作を実行することをより確実に防止することができ、また遊技台が予測していない動作をおこなうといった不具合（例えば暴走、RAM破壊）を抑制することができる場合があると同時に、遊技制御を終了させる処理の実行は、所定の制御部に供給される電力の電圧値が第1の電圧値よりも低い第2の電圧値になるまで遊技制御を継続するので、制御期間を長く継続でき、また、遊技制御の終了処理から所定の制御部、特に基本回路の電源が完全に遮断されるまでの期間を短くすることができ、所定の制御部、特に基本回路が不安定な動作を実行することをより確実に防止することができ、また遊技台が予測していない動作をおこなうといった不具合（例えば暴走、RAM破壊）をより抑制することができる場合がある。逆に第2の電圧値を第1の電圧値よりも高くして適用した場合であれば、遊技台に電源供給を開始してから遊技制御の開始を迅速におこなうことができる場合があると同時に、遊技制御を終了させる処理の実行は、所定の制御部に供給される電力の電圧値が遊技制御の終了処理から所定の制御部、特に基本回路の電源が完全に遮断されるまでの期間を十分に長く維持することができるので、電断時の所定の制御部、特に基本回路の制御状態を確実に保存することができる場合がある。

#### 【2053】

また、実施例に挙げた払出制御部リセット割り込み処理などについては、遊技制御の終了処理の後でさらに電圧を監視し、所定の低電圧信号がオフになると所定の処理（例では起動信号を入力してから最初に開始する処理）に移行するように構成しているが、この所定の処理に移行する条件を上述の遊技制御を終了させる処理を実行する条件で使用している第2の低電圧信号がオフ（第2の電圧監視回路が第2の低電圧信号を出力していない状態）としてもよい。所定の制御部への電力供給が断たれた場合であれば、遊技制御を終了させる処理を実行する条件で使用している第2の低電圧信号がオンになった後で、遊技制御を終了させる処理を実行するので、さらに電圧は低下しているはずであるにも関わらず電圧値が下がっていない、または上昇している場合には、ノイズ、電力の不安定な変動などが原因で第2の低電圧信号を基本回路は入力し、遊技制御を終了させる処理を実行させた可能性が高い。そのような場合を考えるとこの構成は所定の処理に早く移行することができるので、ノイズ、電力の不安定な供給が原因で遊技を中断する期間を少なくすることができ、遊技を中断している遊技者の不快感および遊技台の稼働の低下を減少することが



可能になる場合がある。またこの場合、所定の処理に移行する条件として、上述の遊技制御の開始を待機させる条件で使用している第1の低電圧信号がオフになった場合を適用しても同様の効果を得られる場合がある。

#### 【2054】

また、上述の複数の電圧監視回路のうち第1・第2の電圧監視回路とは別の第3の電圧監視回路を設け、その第3の電圧監視回路は第1・第2の電圧値とは別の第3の電圧値よりもその所定の電力線の電圧値が低い場合に第3の低電圧信号を、それぞれ所定の制御部の基本回路に出力するように構成し、上述の所定の処理に移行する条件として、上述の第3の低電圧信号がオフ（第3の電圧監視回路が第3の低電圧信号を出力していない状態）にである場合としてもよい。このように構成することで各条件に応じた電圧値を適宜設定することができる場合があり、所定の制御部、特に基本回路が不安定な動作を実行することをより確実に防止することができ、また遊技台が予測していない動作を行うといった不具合（例えば暴走、RAM破壊）を抑制することができる場合がある。この場合第3の電圧値を第2の電圧値よりも低く設定することで、遊技制御を終了させる処理の実行で減少する電力を加味した最適な形式とすることができる場合がある。

10

#### 【2055】

また、電圧監視回路（336、426）、第1～第3の電圧監視回路を主制御部および払出制御部の両方に備える例を挙げたが、これに限定されず、主制御部に低電圧信号を出力する電圧監視回路および払出制御部に低電圧信号を出力する電圧監視回路を主制御部、払出制御部、または電源管理部のいずれか1つに設けても良く、1つの電圧監視回路が主制御部および払出制御部の両方に低電圧信号を出力するように構成し、この電圧監視回路を主制御部、払出制御部、または電源管理部のいずれか1つに設けても良い。

20

#### 【2056】

また、各々の球検出センサにおける過去3回分の検出信号の有無の情報が予め定めた入賞判定パターン情報（実施例では、前々回検出信号無し、前回検出信号有り、今回検出信号有りであることを示す情報）と一致した場合にその球検出センサに対応する入賞口などへの入賞と判定するとともに、機能限定状態を継続する期間の長さをこの過去の検出信号の有無の情報の個数分に対応させる例を挙げたがこれに限定されず、機能限定状態を継続する期間の長さをこの過去の検出信号の有無の情報の個数分よりも長くするようにしてもよい。また、入賞判定パターンも予め定めていればどのようなものでもよく、各々の球検出センサにおける過去2回分の検出信号の有無の情報が予め定めた入賞判定パターン情報（例えば、前回検出信号無し、今回検出信号有りであることを示す情報）と一致した場合にその球検出センサに対応する入賞口などへの入賞と判定するようにしても良いし、各々の球検出センサにおける過去4回分の検出信号の有無の情報が予め定めた入賞判定パターン情報（例えば、3回前検出信号無し、前々回検出信号無し、前回検出信号有り、今回検出信号有りであることを示す情報）と一致した場合にその球検出センサに対応する入賞口などへの入賞と判定するようにしても良い。また複数の球検出センサの全てがオフ（通過非検出）の場合にオンに設定し、1つでもオン（通過検出）の場合にオフに設定するフラグをRAM308に設け、このフラグがオンに設定されたタイマ割り込み処理の次回以降の所定回数（例えば、1回、2回）、連続してオン（通過検出）を出力した球検出センサ（上述複数の球検出センサに含まれるもの）に対応する入賞口などへの入賞があったと判定するようにしてもよい。

30

40

#### 【2057】

また、入賞判定パターン情報と一致しているかどうかの判定に使用する検出信号の有無の情報を生成するための過去の所定回数分（例えば過去3回分）の検出結果が、今回の検出結果を含んでいるようにしてもよい。

#### 【2058】

また、払出制御部リセット割り込み処理（図266）では、低電圧信号がオンになった場合に電断時処理に移行し、その後低電圧信号がオフに変化した場合に、所定の処理（このフローでは初期設定1）に処理を移行し、さらに低電圧信号がオフになるまで所定の制

50

御（このフローでは払出制御、復電制御）に移行しないように待機している。通常、電源オフなどにより電力の供給が停止した場合には、その後なだらかに電力（電圧）が低下していくような回路構成にしているが、この低下していく際に電圧値は波打ちながら低下していく場合があり、低電圧信号がオンとオフを繰り返す現象が発生する場合がある。そのため、この払出制御部リセット割り込み処理のように、電断時処理のあとで低電圧信号がオフになった場合に、直ちに所定の制御（このフローでは払出制御、復電制御）に移行しないように、ステップS11002で低電圧信号がオフになるまで制御の進行を待機させることで、所定の制御部（例えば払出制御部600）、特に基本回路402が不安定な動作を実行することをより確実に防止することができ、また遊技台が予測していない動作を行うといった不具合（例えば暴走、RAM408の破壊）を抑制することができる場合がある。このような処理を他の制御部（例えば主制御部300、第1副制御部400）に適用するようにしてもよい。

10

#### 【2059】

また、各々の球検出センサにおける過去3回分の検出信号の有無の情報が予め定めた入賞判定パターン情報（例では、前々回検出信号無し、前回検出信号有り、今回検出信号有りであることを示す情報）と一致した場合にその球検出センサに対応する入賞口などへの入賞と判定するとともに、機能限定状態を継続する期間の長さをこの過去の検出信号の有無の情報の個数分に対応させる例を挙げたが、さらに所定の装置（例えばガラス枠151）の状態（例えば開放状態、閉鎖状態）を確認するために、該所定の装置が第1の状態（例えば開放状態）である場合にオン信号を出力し、第1の状態とは別の第2の状態（例えば閉鎖状態）の場合にオフ信号を出力する所定の検出装置を備えて、所定の回数（例えば2回）分の検出装置からの信号の有無（オン、オフ）の情報が予め定めた入賞判定パターン情報（例えば、前回検出信号有り、回検出信号有りであることを示す情報）と一致した場合にその所定の装置の状態が第1の状態と判定し、一致しなかった場合には第2の状態と判定するようにしてもよい。このように第1の検出装置からの第1の回数分の信号に基づく第1の判定と、第2の検出装置からの第2の回数分の信号に基づく第2の判定と、について第1の回数と第2の回数が異なる場合には、第1の回数と第2の回数とのうち一方に対応する回数に対応する期間に亘って、この第1の判定および第2の判定をおこなわない期間限定処理を継続するようにしても良い。例えば第1の回数の方が第2の回数よりも多い場合には、第1の回数に対応する期間に亘って、この第1の判定および第2の判定をおこなわない期間限定処理を継続するようにしても良いし、第2の回数に対応する期間に亘って、この第1および第2の判定をおこなわない第1の期間限定処理と、該第1の期間限定処理の終了から第1の判定処理をおこなわずに、第2の判定をおこなう第2の期間限定処理を継続するようにしてもよい。例えば第1の判定として上述の入賞口などへの球通過判定を適用し、第2の判定としてガラス枠151の上述の状態判定を適用すると、電源投入から2回のタイマ割り込み処理では上述の球通過判定および上述のガラス枠151の状態判定の両方をおこなわず、3回目のタイマ割り込み処理では上述の球通過判定をおこなわず、上述のガラス枠151の状態判定はおこない、4回目移行のタイマ割り込み処理では上述の球通過判定および上述のガラス枠151の状態判定の両方の判定をおこなうようにしてもよい。こうすることで処理時間を短縮することができ、処理の負荷を低減させられる場合がある。

20

30

40

#### 【2060】

また、実施例では、払出制御部600のRAMのエラーステータス記憶領域に記憶している各装置のエラー情報（例えば皿満杯エラーを示す情報など）のうち、所定のエラーが発生していることを示す情報がオンの場合には、払出用スプロケットを回転させるためのモータの駆動、すなわち払出装置からの賞媒体（例えば遊技球）の払出を停止するようにするとともに、所定の期間（例えば上述のモータ駆動中）主制御部300から受信した払出要求情報に含まれている賞球数を次賞球要求数に加算・記憶しておき、該所定の期間の終了後に、その次賞球要求数分の賞球を連続的に払い出すように、払出用スプロケットを回転させるためのモータを駆動するようにしているが、このような構成とすることで、遊

50

技中は常に皿を満杯（球抜きレバーを操作しない）にしておくことで皿満杯エラーをわざと発生させて払出を停止し、かつ賞球数を次賞球要求数に加算・記憶しておき、遊技の終了時に球抜きレバーを操作して皿満杯エラーを解消させて、全ての賞球を一度に払出させるようなことをする遊技者に対しても、払出用スプロケットを回転させるためのモータの駆動が従来の機械のような間欠的なものではなく、連続的に払出する構成であることから払出時間を短縮でき、遊技者を待たせる時間を減少させられるとともに、待ち時間が長い場合に生じる遊技者のイライラ感・不快感を減らすことができる場合がある。

#### 【2061】

また、基本コマンドに、遊技情報（特図抽選状態が低確率なのか、高確率なのか、特図変動時間短縮状態か通常状態か、第2始動口開放時間延長状態か通常状態かなど）を含めるようにしてもよく、復帰コマンド、基本コマンドなどに、大当たりフラグの値、確変フラグの値を含めるようにしてもよい。

10

#### 【2062】

また、本実施例における普通図柄変動遊技は、普図始動口228に球が入賞したことを所定の球検出センサが検出した場合に、ステップS9310の入賞受付処理で、普図当選乱数値を取得し、取得した値をRAM308の所定の記憶領域に記憶するところから開始し、ステップS9313の普図関連抽選処理でRAM308の所定の記憶領域に記憶している乱数カウンタの値を普図タイマ乱数値として取得し、RAM308の所定の記憶領域に記憶してある普図当選乱数値に基づいて当否判定し、その判定結果と取得した普図タイマ乱数値に基づいて普図の変動時間の決定をおこない、その変動時間の間に亘って普図を

20

#### 【2063】

また、本実施例における特別図柄変動遊技は、第1特図始動口230または第2特図始動口232に球が入賞したことを所定の球検出センサが検出した場合に、ステップS9310の入賞受付処理で、特図当選乱数値および特図乱数値を取得し、取得した値をRAM308の所定の記憶領域にそれぞれ記憶するところから開始し、ステップS9315の特図抽選処理でRAM308の所定の記憶領域に記憶している乱数カウンタの値を特図タイマ乱数値として取得し、RAM308の所定の記憶領域に記憶してある特図当選乱数値に基づいて当否判定し、RAM308の所定の記憶領域に記憶してある特図乱数値に基づいて

30

#### 【2064】

また、本実施例における入賞には、遊技盤200に設けた入賞口や始動口に球が入球した場合のほか、遊技盤200に設けた通過領域を球が通過した場合（例えば、普図始動口124を球が通過した場合）も含まれる。また、始動口は、必ずしも本実施例で示される位置に設ける必要は無く、例えば、特定の入賞口内の特定の領域を始動口として機能させてもよい。

40

#### 【2065】

また、可変入賞口234は、球が入賞不可能な閉状態と、入賞可能な開状態の2種類の開閉状態に変化するものに限定されず、球が入賞可能な第1の開状態と、第1の開状態よりも入賞が容易な第2の開状態の2種類の開状態に変化するものも含まれる。

#### 【2066】

また、本実施例では、大当たりに対応する図柄組合せ（大当たり図柄組合せ）を装飾図柄表示装置208に停止表示した場合に大当たり遊技の開始となるが、大当たり遊技の開始条件はこれに限定されるものではない。したがって、例えば、大当たり図柄組合せを停止表示した後で、盤面の特定のゲートを遊技球が通過した場合、大当たり図柄組合せを停止表示した後で、盤面の特定の入賞口に遊技球が通過した場合、大当たり図柄組合せを停止表示した後で

50

、大入賞口（可変入賞口）以外の入賞口内の特定の領域を遊技球が通過した場合、特定の入賞口に遊技球が入賞したことを条件に開放した大入賞口（可変入賞口）内の特定の領域を遊技球が通過した場合、などを大当り遊技の開始条件としてもよい。

#### 【2067】

また、本実施例における特図は、特図表示装置212、214が変動表示および停止表示する図柄全てを示しており、変動表示する図柄のみ、または、停止表示する図柄のみを示すものではない。さらに、本実施例における普図や装飾図柄も同様に、普図表示装置210や装飾図柄表示装置208が変動表示および停止表示する図柄全てを示しており、変動表示する図柄のみ、または、停止表示する図柄のみを示すものではない。

#### 【2068】

また、大当りフラグおよび確変フラグの両方がオンの場合には上述の特図A等、大当りフラグがオンで確変フラグがオフの場合には上述の特図B等、大当りフラグおよび確変フラグの両方がオフの場合には上述の特図I等を停止表示するようにしたが、確変フラグの値によらず大当りフラグがオンの場合には上述の特図B等、大当りフラグがオフの場合には特図C等を停止表示し、確変フラグがオンの場合には高確中ランプ222を点灯させるようにしてもよい。もちろん、大当りフラグがオンで確変フラグがオフの場合に停止表示する特図の停止態様を1または複数種類、大当りフラグおよび確変フラグの両方がオンの場合に停止表示する特図の停止態様を1または複数種類、および大当りフラグおよび確変フラグの両方がオフの場合に停止表示する特図の停止態様を1または複数種類用意しておいてもよい。このように確変移行判定の結果を特図表示装置212、214が報知するようにしてもよい。

#### 【2069】

なお、遊技状態として高確率状態、低確率状態、大当り状態の3つを例示したが、これら以外にも、第2特別始動口232に設けた羽根の開放継続時間が所定の時間である通常開放状態と、その所定の時間よりも長い開放延長状態と、を設け、この2つの状態についても遊技状態に含めてもよい。また、普図表示装置210の変動時間を所定の変動時間とする通常変動状態（普図）と、該所定の変動時間よりも短い変動時間とする短縮変動状態（普図）と、を設け、この2つの状態についても遊技状態に含めてもよい。また、特図表示装置212、214の変動時間を所定の変動時間とする通常変動状態（特図）と、該所定の変動時間よりも短い変動時間とする短縮変動状態（特図）と、を設け、この2つの状態についても遊技状態に含めてもよい。また、普図変動遊技の結果を当選とするか否かの抽選における当選確率を所定の確率とする低確率状態（普図）と該所定の確率よりも高い高確率状態（普図）とを設け、この2つの状態についても遊技状態に含めてもよいし、これら複数の状態を複合して発生させるようにしてもよい。例えば、上述の特図1で大当りした場合には大当りの終了後に特図の変動開始および停止を所定回数（例えば50回）繰り返すまでの間、高確率状態、高確率状態（普図）、開放延長状態、短縮変動状態（普図）、短縮変動状態（特図）の5つの状態を複合して遊技者に有利な状態を発生させ、その所定回数の経過、または上述の特図2を停止表示しての大当りが終了した場合には、次に大当りを開始するまでの間、低確率状態、低確率状態（普図）、通常開放状態、通常変動状態（普図）、通常変動状態（特図）の5つの状態を複合して遊技者に不利な状態を発生させ、上述の特図1で大当りした場合には大当りの終了後に次に大当りを開始するまでの間、高確率状態、高確率状態（普図）、開放延長状態、短縮変動状態（普図）、短縮変動状態（特図）の5つの状態を複合して遊技者に最も有利な状態発生させるようにしてもよい。

#### 【2070】

また、確変フラグがオンであり、装飾図柄表示装置208に装飾2—装飾2—装飾2など大当り終了後に確率変動となることを確定的に報知しない大当り図柄の組合せを停止表示した場合には、大当り開始前、大当り中、大当り終了後のいずれかのタイミングで確変となること、または確変となったことを装飾図柄表示装置208による特定の画像表示で報知する場合には、その特定の画像表示で報知すること、報知する特定画像の種類、報知

10

20

30

40

50

するタイミングなどのうち1、複数又は全部に基づいて、演出のパターンを選択するようにしてもよい。ここで確変となること、または確変となったことの特定の画像による報知は、装飾図柄表示装置208に装飾2—装飾2—装飾2など大当たり終了後に確率変動となることを確定的に報知しない大当たり図柄の組合せを停止表示したが、大当たり終了後に確変となることを昇格、確変、または確率変動などといった文字情報を含んだ画像を装飾図柄表示装置208などに表示することなどにより確定的に報知することを示している（これを昇格演出と呼ぶ）。

#### 【2071】

また、上述の実施例では、特図表示装置212、214に特図1または特図2を停止表示した場合に、大当たり遊技または特別大当たり遊技を開始すると記載したが、特図表示装置212、214に特図1または特図2を停止表示した場合に確変フラグが示す値によらず大当たり遊技を開始し、その大当たり終了後に遊技状態を高確状態にするようにしてもよいし、特図表示装置212、214に特図1または特図2を停止表示した場合に確変フラグがオンの場合には大当たり遊技中に可変入賞口234を15回開閉（15ラウンドの大当たり）し、確変フラグがオフの場合には大当たり遊技中に可変入賞口234を5回開閉（5ラウンドの大当たり）するようにしてもよい。

#### 【2072】

また、上述の実施例では、第1始動口用抽選データと第2始動口用抽選データを別データとした例を示したが、これに限定されず、第1始動口に遊技球が入賞したことを検出した場合、および第2始動口に遊技球が入賞したことを検出した場合に行う大当たり判定の両方で、単一の所定の抽選データを参照するようにしてもよい。

#### 【2073】

また、上述の実施例においては、遊技台として、遊技球を遊技媒体としたパチンコ機の例を示したが、本発明に係る遊技台はこれに限定されるものではなく、例えば、メダル（コイン）を遊技媒体としたパチスロ機（スロットマシン）などにも適用可能である。

#### 【2074】

さらに、本発明は、前述のパチスロ機やパチンコ遊技機等の実機その他、これらのパチスロ機等の実機の動作を家庭用ゲーム機用として擬似的に実行するようなゲームプログラムにおいても、本発明を適用してゲームを実行することができる。その場合、ゲームプログラムを記録する記録媒体は、DVD-ROM、CD-ROM、FD（フレキシブルディスク）、その他任意の記録媒体を利用できる。

#### 【2075】

以上、本発明の実施例を説明したが、発明の実施の形態に記載された、作用及び効果は、本発明から生じる最も好適な作用及び効果を列挙したに過ぎず、本発明による作用及び効果は、本発明の実施の形態に記載されたものに限定されるものではない。

#### 【2076】

次に、以上説明した実施形態6によるパチンコ機100の特徴的構成について説明する。

（1）本実施の形態によるパチンコ機100は、

16ビット乱数を少なくとも生成可能な乱数生成手段と、

前記16ビット乱数を少なくとも記憶可能な記憶手段と、

複数の遊技制御処理のそれぞれを、メイン制御および割込み周期ごとに行われる割込み制御のうちの、少なくともいずれかの制御において実行可能な遊技制御手段と、

を備えた遊技台であって、

前記遊技制御手段は、前記16ビット乱数を一命令で前記記憶手段に転送可能であり、

前記乱数生成手段の特定フラグは前記命令の実行前後で変化しないこと

を特徴とする。

#### 【2077】

ハード乱数取得の流れの一例として、「ラッチレジスタに値をセット→乱数値レジスタに乱数を取り込まれる→ラッチフラグレジスタにフラグがセットされる→該フラグを参照

10

20

30

40

50

し、フラグが立っている場合に乱数を取得→フラグクリア」がある。この場合、ラッチ信号の出力からラッチ完了信号（セット信号）あるいはリード完了信号（クリア信号）のタイミングで乱数取得処理の実行時期が推測され易いという課題があるが、上記構成を備えたパチンコ機100によれば、16ビット乱数を一命令で転送可能であり、マイコン内の特定フラグは該命令実行前後で変化しないため、特定の処理の実行時期を判別困難にでき、これらの特定の処理の狙い撃ちを防止することができる場合がある。なお、特定フラグは例えば、乱数ソフトラッチフラグレジスタの所定のビットのことである。

#### 【2078】

16ビット乱数は複数の番地（レジスタ）に跨って格納され、一命令で下位バイトのレジスタを指定するだけでリード可能である。当否判定乱数は16ビット乱数で、必ずこの一命令で乱数を取得するようにしてもよい。また、16ビット乱数のみをこの一命令で取得し、当否判定乱数が8ビット乱数の場合は他の命令で取得するようにしてもよい。

10

#### 【2079】

乱数リードを許可する処理（従来はラッチ信号が出力されていたが、本実施形態ではいわゆるプログラム上のフラグを立てるだけなので信号出力はない）と、乱数をリードする処理（リード信号出力あり）を別モジュールで実行するようにしてもよい。従来はラッチ信号出力のタイミングから乱数をリードする処理を推測し易かったが、本実施形態では推測困難にできる場合がある。

#### 【2080】

さらに、乱数リードを許可する処理とリードする処理の間に別の処理があり、この別の処理の実行時間は種々の判定処理があるので可変とすることができる。処理の推測をより困難にできる場合がある。

20

#### 【2081】

リードを許可する処理は一割込み内で複数回（特図1、特図2、普図の組合せ）実行可能としてもよい。さらに一のモジュール内で実行可能としてもよい。一のモジュールは、主制御部タイマ割込み処理の例えば入力ポート状態更新処理（ステップS205）であってもよい。リードを許可する処理が一割込み内で実行される回数は可変なので乱数をリードする処理の実行時期を推測困難にできる場合がある。

#### 【2082】

乱数をリードする処理も一割込み内で複数回実行可能としてもよい。この場合の複数回数には、特図1、特図2、普図の組合せだけに限らず、ハード乱数（16ビット乱数生成回路で生成される乱数）とソフト乱数（ユーザプログラムより生成される乱数）の組合せでもよい。さらにこれらのリード処理を一のモジュール内で実行可能としてもよい。例えば、入賞判定パターンに一致した場合に乱数をリードする処理を一モジュール内で繰り返し行うようにしてもよい。あるいは、別モジュールにして、さらに別の処理（別のモジュール）が挟まれているようにしてもよい。モジュールの先頭アドレスはRST命令に対応するアドレスとは異なってもよいし、一致していてもよい。

30

#### 【2083】

乱数の取込方法は3種類ある。上述の通り、ソフトウェアによる取込みとして、ソフトラッチ乱数値レジスタの下位バイトを読み込み、続いて上位バイトをリードした場合にソフトラッチフラグレジスタをリセットする取込み方法がある。また、一命令である第三の命令を用いてソフトラッチ乱数値レジスタの下位、上位バイトをRAMに直接取込み、ソフトラッチフラグレジスタの状態に変更を加えない取込み方法がある。さらに、外部端子入力による乱数取込みも行うことができる。

40

#### 【2084】

また、16ビット乱数にソフト乱数を組み合わせたり、さらに初期値更新乱数を組み合わせたりしてもよい。あるいは、これらを組み合わせなくてもよい。例えば、変動時間決定用乱数にハード乱数を使う場合、この乱数をペアレジスタに取り込まず直接にRAMに読み込むようにしてもよい。逆に当否判定乱数は必ずペアレジスタに取り込むようにしてもよい。

50

## 【2085】

また、最大値設定しない乱数生成回路と最大値設定をする乱数生成回路があり、電源投入後に自動で起動する乱数生成回路もある。また、手動起動させる乱数生成回路もある。磁界検知信号、磁気検知信号、電波検知信号、振動検知信号等が所定の端子に入力されている場合であっても乱数の最大値設定を実行して乱数生成回路を起動させるようにしてもよい。

## 【2086】

機能限定タイマの構成として、乱数を取得して抽選を行う処理は主制御部タイマ割込み処理で行い、機能限定タイマを作動させている間は主制御部タイマ割込み処理に移行するが抽選処理には移行しないようにしてもよい。この場合でもハード乱数は更新されている。乱数を取得して抽選を行う処理は主制御部タイマ割込み処理で実行し、機能限定タイマ作動中は主制御部タイマ割込み処理の一部のみを実行する。抽選取得しないのに最大値設定で乱数生成回路を起動しておくことにより処理を推測困難にできる場合がある。

10

## 【2087】

また、複数の乱数の起動タイミングを異ならせるようにしてもよい。また、最大値が大きいものが常に最初に起動されるようにしてもよい。電源投入後の最初のモジュール（スタックポインタを設定する処理など）とは異なるモジュールで、乱数値の最大値を設定するようにしてもよい。WDT起動後に所定の乱数生成回路に最大値を設定して起動させるようにしてもよい。

## 【2088】

20

乱数最大値設定処理の位置は、例えば以下のタイミングが挙げられる。

(a) 電源投入後の低電圧信号出力判定ループ処理より前、当該処理中、または当該処理より後、のいずれか。

(b) 内蔵レジスタに所定のデータを設定する処理より前、当該処理中、または当該処理より後、のいずれか。

(c) 電源ステータスを参照する処理（分岐する処理）より前、当該処理中、または当該処理より後、のいずれか。

(d) チェックSUMのループ処理より前、当該処理中、または当該処理より後、のいずれか。

(e) 内蔵RAMアクセス許可より前、当該処理中、または当該処理より後、のいずれか。

30

(f) 復帰起動または初期起動より前、当該処理中、または当該処理より後、のいずれか。

(g) 割り込み許可より前、当該処理中、または当該処理より後、のいずれか。

(h) サブ起動待ちより前、当該処理中、または当該処理より後、のいずれか。

(i) ソフト乱数の更新開始より前、当該処理中、または当該処理より後、のいずれか。

(j) サブへの通信確認コマンド（サブ起動開始コマンド、完了コマンド）、払出への通信確認コマンドなどのコマンド送出より前、当該処理中、または当該処理より後、のいずれか。

(k) WDT手動起動より前、当該処理中、または当該処理より後、のいずれか。

40

## 【2089】

(2) 上記パチンコ機100であって、

前記遊技制御手段を少なくとも有するマイクロプロセッサを備え、

前記マイクロプロセッサは、

少なくとも、セキュリティチェックを行うことが可能なものであり、該セキュリティチェックの後に前記遊技制御が行われるユーザモードへ移行するまでの時間をランダムに変動させるランダム延長処理が少なくとも行われる、セキュリティモードに滞在した後、該ユーザモードへ移行するものであることを特徴とする。

## 【2090】

当該構成を備えたパチンコ機100によれば、ユーザモードに移行する前にランダム延

50

長処理を実行するので、遊技制御プログラムの実行開始タイミングをずらすことが可能なため、遊技制御処理の特定の処理（例えばソフト乱数の更新処理、初期値更新処理等）の実行時期を判別困難にでき、これらの特定の処理の狙い撃ちを防止することができる場合がある。

#### 【2091】

（3）上記パチンコ機100であって、前記ランダム延長処理の開始前に、前記マイクロプロセッサの所定の出力端子から所定の信号が出力されることを特徴とする。

（4）上記パチンコ機100であって、前記所定の端子は、前記マイクロプロセッサ以外の外部機器に対してリセット信号を出力可能なりセット端子であることを特徴とする。

#### 【2092】

当該構成を備えたパチンコ機100によれば、ランダム延長処理の前に外部機器に対してリセット信号を出力し、ランダム延長処理の後に遊技制御プログラムの実行を開始することができるので、遊技制御処理の特定の処理（例えばソフト乱数の更新処理、初期値更新処理等）の実行時期を判別困難にでき、これらの特定の処理の狙い撃ちを防止することができる場合がある。

#### 【2093】

（5）上記パチンコ機100であって、前記遊技制御手段から送出されるコマンドに基づいて演出制御を行う副制御手段を有することを特徴とする。

#### 【2094】

##### <実施形態7>

以下、図面を用いて、本発明の実施形態7に係るパチンコ機（遊技台）について詳細に説明する。従来、乱数を用いて遊技制御を行う遊技台が知られている。しかし、不正者によって遊技台を解析されて不正に利益を得られる不正行為が行われてしまうという問題があった。

#### 【2095】

##### <乱数回路>

まず、図270を用いて、本実施形態に係る遊技台に用いられる16ビット乱数回路について説明する。図270は、16ビット乱数回路の内部構成図である。

#### 【2096】

16ビット乱数回路9000は、それぞれ異なる乱数を生成する4つの乱数生成チャンネルCH0～CH3を備えている。乱数生成チャンネルCH0～CH3は、それぞれ個別に16ビット乱数を発生させることができる。なお、各チャンネル回路の内部構成は同一であるため、図270では、1つの乱数生成チャンネルCH0を示し、残りの乱数生成チャンネルCH1～CH3については、図示を簡略化している。以下の説明では、この乱数生成チャンネルCH0を中心に16ビット乱数回路9000について説明する。

#### 【2097】

プログラム管理エリア内の16ビット乱数初期値設定部9002には、RCK端子からの外部クロック信号9004aと内部システムクロック9004bのうち、いずれのクロック信号を乱数生成回路9006で使用するかを定めるクロック選択情報が、プログラム管理エリア内の情報に基づいて設定される。RCK端子からの外部クロック信号9004aと内部システムクロック9004bとは、乱数生成回路9006の手前に設けられたマルチプレクサに入力される。16ビット乱数初期値設定部9002が上記クロック選択情報に従って更新クロック選択信号をこのマルチプレクサに入力することにより、この更新クロック選択信号によって選択されたクロック信号が乱数生成回路9006に入力されるようになっている。なお、外部クロック信号9004aが選択されている場合、所定の分周比（この例では、1/2）で分周されたクロック信号が乱数生成回路9006に入力される。なお、分周されたクロック信号が、内部システムクロック9004bよりも低い周波数でない場合には使用することができない。なお、本実施形態では、外部クロック信号9004aが選択されているものとして説明を続ける。

#### 【2098】



乱数生成回路 9006 による乱数値更新を開始する起動方法は、最大値設定信号による起動又はユーザモード起動信号 9008 による起動の 2 つの方法から選択することができる。最大値設定で起動を選択した場合には、生成される乱数値の最大値が最大値設定レジスタ 9010 に設定された後、乱数生成回路 9006 が起動される。ユーザモード起動信号 9008 による起動を選択すると、ユーザモード移行で自動的に乱数生成回路 9006 が起動され、最大値の設定をすることができないようになっている。乱数値の最大値は、乱数生成チャンネル CH0 ～ CH3 でそれぞれ個別に設定することができる。

#### 【2099】

16 ビット乱数初期値設定部 9002 には、いずれの方法で乱数生成回路 9006 を起動するかを決める選択情報が、プログラム管理エリア内の情報に基づいて設定される。最大値設定で起動する起動信号又はユーザモード起動 9008 で起動する起動信号とが乱数生成回路 9006 の手前に設けられたマルチプレクサに入力される。16 ビット乱数初期値設定部 9002 が上記クロック選択情報に従っていずれの起動信号を選択するかを選択信号をこのマルチプレクサに入力することにより、選択された起動信号で乱数生成回路 9006 が起動するようになっている。

#### 【2100】

乱数生成回路 9006 は、16 ビット乱数初期値設定部 9002 によって選択されたクロック信号が入力され、このクロック信号の周期に従って乱数を更新する。乱数生成回路 9006 には、更新クロック信号の 1 クロック毎に 1 が加算され、乱数が更新されるようになっている。16 ビット乱数回路 9000 の乱数生成回路 9006 では、0 から 65535 までの乱数を生成することができる。

#### 【2101】

また、乱数生成回路 9006 には、更新監視回路（不図示）が設けられている。乱数生成回路 9006 が更新した乱数の値は、更新監視回路に入力される。更新監視回路は、乱数値の入力に基づいて、乱数が正常に更新されているか否かを監視する。乱数の更新に異常がある場合、この乱数の更新に異常があることを示す情報を、リセットコントローラ 9060 に送信する。

#### 【2102】

乱数生成回路 9006 の 1 周目からの乱数スタート値は、例えば、「0001H」、「システムリセット毎の変更値」又は「ID ナンバーを基にした値」のいずれかに、16 ビット乱数初期値設定部 9002 によって設定される。乱数スタート値が「0001H」に設定されると、乱数生成回路 9006 の起動時（更新開始時）のスタート値が 0001H に設定される。また、乱数スタート値が「システムリセット毎の変更値」に設定されると、乱数生成回路 9006 の起動時（更新開始時）のスタート値がシステムリセット毎に変動する値に設定される。また、乱数スタート値が「ID ナンバーを基にした値」に設定されると、乱数生成回路 9006 の起動時（更新開始時）のスタート値がチップ個別に異なる値に設定される。

#### 【2103】

乱数スタート値選択回路 9012 には、16 ビット乱数初期値設定部 9002 からいずれの乱数スタート値で乱数生成回路 9006 が乱数更新を開始するか設定する信号が入力される。乱数スタート値選択回路 9012 は、入力された信号に基づいて、乱数生成回路 9006 に選択された乱数スタート値を出力する。乱数生成回路 9006 は、入力された乱数スタート値から乱数更新を開始する。

#### 【2104】

乱数生成回路 9006 が生成する乱数列は、所定の数値範囲（例えば、0 ～ 65535）で更新される。16 ビット乱数回路 9000 では、この乱数列の並びを変更することができる。乱数列の変更方法は、ソフトウェアによる変更、2 周目から自動的に変更または 1 周目から自動的に変更のいずれかを選択することができる。ソフトウェアによる変更を選択した場合には、ユーザプログラムにより乱数列が変更される。また、この場合には、乱数生成回路 9006 が生成する乱数値が一巡し、次の周期の開始後から乱数列が変更さ

れる。なお、乱数列変更選択回路 9016 が乱数生成回路 9006 から乱数一巡信号を受けた時点で、一つの乱数列を一周したと判定される。

#### 【2105】

ソフトウェアによる変更を選択した場合にのみ有効な乱数列変更レジスタ 9014 が設けられている。乱数列変更レジスタ 9014 は、乱数生成回路 9006 が生成した乱数列の並びを変更するために設けられている。

#### 【2106】

乱数列変更選択回路 9016 には、16ビット乱数初期値設定部 9002 により乱数列を変更するか否かの選択信号が入力される。乱数列変更選択回路 9016 は、入力された選択信号に基づいて、乱数列の変更を行うか否かを選択する。乱数列を変更しない場合には、乱数列変更選択回路 9016 は、「変更せず」という信号を選択し、乱数列変更回路 9018 に出力する。乱数列を変更する場合には、乱数列変更選択回路 9016 は、いずれの変更方法で乱数列を変更するかという信号を乱数列変更回路 9018 に出力する。ソフトウェアによる変更が選択された場合には、乱数列変更選択回路 9016 は、乱数列変更レジスタ 9014 からの信号に基づいて、乱数列変更回路 9018 に乱数列の変更情報を出力する。

#### 【2107】

乱数列変更回路 9018 は、乱数列を変更する場合には、乱数生成回路 9006 が生成した乱数列を変更し、変更後の乱数列を最大値比較回路 9020 に出力し、乱数列を変更しない場合には、乱数生成回路 9006 が生成した乱数列をそのまま最大値比較回路 9020 に出力する。

#### 【2108】

最大値を設定して乱数生成回路 9006 を起動した場合には、最大値設定レジスタ 9010 に設定された最大値が格納されている。最大値設定レジスタ 9010 に最大値が設定されている場合には、最大値比較回路 9020 は、入力される乱数値と最大値との比較を行う。最大値比較回路 9020 は、当該乱数値が最大値以下であると判断すると、後述するハードラッチ乱数値レジスタおよびソフトラッチ乱数値レジスタに乱数値を出力し、当該乱数値が最大値より大きいと判断すると、ハードラッチ乱数値レジスタおよびソフトラッチ乱数値レジスタに乱数値を出力しないようになっている。

#### 【2109】

16ビット乱数回路 9000 には、乱数生成チャンネル CH0 用のハードラッチ選択レジスタとして、CH0 ハードラッチ選択レジスタ 9022a、9022b の2つのレジスタが設けられている。また、乱数生成チャンネル CH1～CH3 用のハードラッチ選択レジスタとして、CH1 ハードラッチ選択レジスタ 9024、CH2 ハードラッチ選択レジスタ 9026、CH3 ハードラッチ選択レジスタ 9028 がそれぞれ設けられている。これらのハードラッチ選択レジスタには、パラレル入力ポートを介して P I 0～P I 5 信号がそれぞれ入力される。各ハードラッチ選択レジスタは、P I 0～P I 5 信号の入力に基づいて、対応するハードラッチ乱数値レジスタにラッチ信号を出力する。

#### 【2110】

ソフトラッチ選択レジスタ 9030 には、後述するソフトラッチ乱数値レジスタ 9040～9046 のそれぞれが乱数をラッチすることを示す情報が C P U 304 からの指示によって適宜選択される。この情報が、ラッチ信号としてソフトラッチ乱数値レジスタ 9040～9046 のそれぞれに出力される。

#### 【2111】

乱数生成チャンネル CH0 用のハードラッチ乱数値レジスタとして、CH0 ハードラッチ乱数値レジスタ 9032a、9032b、9032c、9032d の4つのレジスタが設けられている。乱数生成チャンネル CH1 用のハードラッチ乱数値レジスタとして、乱数生成チャンネル CH1 ハードラッチ乱数値レジスタ 9034a、9034b の2つのレジスタが設けられている。乱数生成チャンネル CH2 用のハードラッチ乱数値レジスタとして、CH2 ハードラッチ乱数値レジスタ 9036a、9036b の2つのレジスタが設

けられている。乱数生成チャンネルCH 3用のハードラッチ乱数値レジスタとして、CH 3ハードラッチ乱数値レジスタ9038a、9038bの2つのレジスタが設けられている。

#### 【2112】

これらのハードラッチ乱数値レジスタ9032a～9032d、9034a～9034b、9036a～9036b、9038a～9038bには、3つの信号が入力される。1つ目の信号は、対応する乱数生成チャンネルの乱数発生回路により生成された乱数を表す信号である。2つ目の信号は、ハードラッチ選択レジスタから出力された乱数ラッチ信号（ハードラッチ信号）である。3つ目の信号は、乱数の読み取りを示すリード信号である。

#### 【2113】

ハードラッチ乱数値レジスタ9032a～9032d、9034a～9034b、9036a～9036b、9038a～9038bには、対応する乱数生成チャンネルの乱数発生回路によって更新された乱数を示す信号が常に入力されている。ここで、ラッチ信号が入力されると、この入力タイミングにおける乱数がハードラッチ乱数値レジスタ内にラッチ（保持）される。このとき、ハードラッチ乱数値レジスタからは、乱数がラッチされていることを示すセット信号が、ハードラッチフラグレジスタ9048a、9048bに出力される。このときCPU304は、ラッチされている乱数を取得することができる。なお、CPU304により乱数が取得されると、ハードラッチ乱数値レジスタにリード信号が入力される。この信号により新たなラッチ信号が入力された場合に乱数をラッチすることを許容する許容状態となる。換言すれば、一度乱数をラッチすると、リード信号が入力されるまで新たに乱数をラッチすることができない非許容状態となる。なお、リード信号が入力されてもラッチされている乱数は保持し続けるため、CPU304はラッチされている同一のタイミングでラッチされた乱数を何度でも取得することができる。このように構成することで乱数ラッチ信号を出力するセンサ回路におけるチャタリングの影響を押さえることができる。なお、リード信号が入力されたことを示すクリア信号がハードラッチフラグレジスタ9048a、9048bに出力される。また、図270に示すように、ハードラッチ乱数値レジスタが複数あることにより、同じ乱数生成回路から生成された乱数を様々なタイミングで取得することができる。

#### 【2114】

ハードラッチフラグレジスタ9048a、9048bには、ハードラッチ乱数値レジスタに乱数がラッチされているか否かを示す情報が記憶される。なお、ハードラッチフラグレジスタ9048a、9048bは、ハードラッチ乱数レジスタにラッチされた乱数（16ビット乱数の全てまたは一部）が読み出されたときに自動的にクリアされるように構成してもよいし、CPU304によってクリアするように構成してもよい。

#### 【2115】

割込み制御レジスタ9052a、9052bには、ハードラッチ乱数値レジスタに乱数が保持された際に割込み制御回路に割込みを発生させるか否かを示す情報が設定される。この情報はハードラッチ乱数値レジスタ毎に設定することができる。例えば、第1特図始動口230への入球により乱数生成チャンネルCH 1の乱数がラッチされた場合には、割込みを発生させ、第2特図始動口232への入球により乱数生成チャンネルCH 2の乱数がラッチされた場合には、割込みを発生させない、というような設定をすることができる。

#### 【2116】

また、乱数生成チャンネルCH 0～CH 3用のソフトラッチ乱数値レジスタとして、CH 0ソフトラッチ乱数値レジスタ9040、CH 1ソフトラッチ乱数値レジスタ9042、CH 2ソフトラッチ乱数値レジスタ9044、CH 3ソフトラッチ乱数値レジスタ9046が乱数生成チャンネルCH 0～CH 3に対してそれぞれ1つずつ設けられている。

#### 【2117】

これらのソフトラッチ乱数値レジスタ9040～9046には、3つの信号が入力され

10

20

30

40

50

る。1つ目の信号は、対応する乱数生成チャンネルの乱数発生回路により生成された乱数を表す信号である。2つ目の信号は、ソフトラッチ選択レジスタ9030から出力された乱数ラッチ信号（ソフトラッチ信号）である。3つ目の信号は、乱数の読み取りを示すリード信号である。

#### 【2118】

ソフトラッチ乱数値レジスタ9040～9046には、対応する乱数生成チャンネルの乱数発生回路によって更新された乱数を示す信号が常に入力されている。ここで、ラッチ信号が入力されると、この入力タイミングにおける乱数がソフトラッチ乱数値レジスタ内にラッチ（保持）される。このとき、ソフトラッチ乱数値レジスタからは、乱数がラッチされていることを示すセット信号が、ソフトラッチフラグレジスタ9050に出力される。このときCPU304は、ラッチされている乱数を取得することができる。なお、CPU304により乱数が取得されると、ソフトラッチ乱数値レジスタにリード信号が入力される。この信号により新たなラッチ信号が入力された場合に乱数をラッチすることを許容する許容状態となる。換言すれば、一度乱数をラッチすると、リード信号が入力されるまで新たに乱数をラッチすることができない非許容状態となる。なお、リード信号が入力されてもラッチされている乱数は保持し続けるため、CPU304はラッチされている同一のタイミングでラッチされた乱数を何度でも取得することができる。このように構成することで乱数ラッチ信号を出力するセンサ回路におけるチャタリングの影響を押さえることができる。なお、リード信号が入力されたことを示すクリア信号がソフトラッチフラグレジスタ9050に出力される。

#### 【2119】

ソフトラッチフラグレジスタ9050には、ソフトラッチ乱数値レジスタ9040～9046に乱数がラッチされているか否かを示す情報が記憶される。なお、ソフトラッチフラグレジスタ9050は、ソフトラッチ乱数レジスタ9040～9046にラッチされた乱数（16ビット乱数の全てまたは一部）が読み出されたときに自動的にクリアされるように構成してもよいし、CPU304によってクリアするように構成してもよい。

#### 【2120】

割込みコントローラ9070は、外部入力や内部状態の変化に応じて適宜処理を実行させるために割込みを発生させる。この割込み処理には、例えば外部からの入力（センサによる信号）を受け付けた場合に実行する処理がある。

#### 【2121】

上記説明した16ビット乱数生成回路9000を用いて、本実施形態では、普図始動口282、第1特図始動口230、第2特図始動口232のそれぞれに入球したタイミングで乱数を取得する。なお、任意のタイミングでCPU304に乱数を取得させるように、プログラムを実行させることもできる。さらに異なる乱数生成チャンネルを使用してそれぞれ独立の乱数を取得し、それぞれのチャンネル毎に乱数生成範囲の最大値を設定することができる。

#### 【2122】

なお、本実施形態では、個別のチャンネルに対して乱数生成範囲を変更することができ、設定されていない場合には所定の最大値（例えば65535）が適用される例について説明したが、これ以外に例えば、乱数生成範囲を変更する場合に全てのチャンネルに対して乱数生成範囲を設定することを必要とする構成であってもよい。

#### 【2123】

なお、普図始動口282に入球したタイミングで取得された乱数は、後述する普図当選乱数として用いられる。また、第1特図始動口230に入球したタイミングで取得された乱数は、加工（例えば、乱数生成回路とは異なる回路（例えば、基本回路302やカウンタ回路312）でカウントアップ（ダウン）される値を乱数に加算（減算）する処理など）されて、後述する特図1当選乱数として用いられる。さらに、第2特図始動口232に入球したタイミングで取得された乱数は、加工されて、後述する特図2当選乱数として用いられる。

## 【2124】

なお、乱数の加工は、第1特図始動口230および第2特図始動口232に入球したタイミングで取得される乱数に限らず、他の契機（例えば、普図始動口282へに入球したタイミングなど）によって取得される乱数に対して行ってもよい。さらに、上述した乱数生成範囲の最大値を設定する場合においては、設定した最大値に応じた加工（例えば、この最大値を超えない値を乱数に加算する処理など）すれば良い。

## 【2125】

また、乱数の加工は必ずしも行う必要はないが、乱数の加工を行うことは乱数の狙いうちの防止に有効に働く。例えば加工した後の乱数が更新されたか否かをCPU304によって監視した場合、乱数生成回路318内での乱数の更新が正常に行われているか否かを正確に判断することができない。このように、乱数を加工する場合には、上述の乱数監視回路3184による乱数の更新の監視がより有効に働くこととなる。

## 【2126】

図271は、8ビット乱数回路10000の内部構成図である。図271に示すように、8ビット乱数回路10000は、それぞれ異なる乱数を生成する4つの乱数生成チャンネルCH0～CH3を備えている。乱数生成チャンネルCH0～CH3は、それぞれ個別に8ビット乱数（0～256）を発生させることができる。なお、各チャンネル回路の内部構成は同一であるため、図271では、1つの乱数生成チャンネルCH0を示し、残りの乱数生成チャンネルCH1～CH3については、図示を簡略化している。なお、8ビット乱数回路10000については、図270に示す16ビット乱数回路9000とほぼ同様であるため、詳細な説明を省略する。

## 【2127】

## &lt;主制御部メイン処理&gt;

次に、図272を用いて、主制御部300のCPU304が実行する主制御部メイン処理について説明する。なお、同図は主制御部メイン処理の流れを示すフローチャートである。

## 【2128】

上述したように、主制御部300には、電源が投入されると起動信号（リセット信号）を出力する起動信号出力回路（リセット信号出力回路）340を設けている。この起動信号を入力した基本回路302のCPU304は、リセット割込によりリセットスタートしてROM306に予め記憶している制御プログラムに従って図272に示す主制御部メイン処理を実行する。

## 【2129】

まず、主制御部メイン処理では、図272（a）に示す、プログラム起動設定処理を行う。プログラム起動設定処理のステップS15101では、スタックポインタ設定を行う。このスタックポインタ設定では、CPU304のスタックポインタ（SP）へのスタック初期値の設定（仮設定）を行う。

## 【2130】

次いで、図272（b）に示す、電源投入時処理を行う。電源投入時処理のステップS15103では、低電圧信号がオンであるか否か、すなわち、電圧監視回路338が、電源制御部660から主制御部300に供給している電源の電圧値が所定の値（本実施形態では9V）未満である場合に電圧が低下したことを示す低電圧信号を出力しているか否かを監視する。そして、低電圧信号がオンの場合（CPU304が電源の遮断を検知した場合）にはステップS15103の処理を繰り返し実行し、低電圧信号がオフの場合（CPU304が電源の遮断を検知していない場合）にはステップS15105に進む。なお、電源が投入された直後で未だ上記所定の値（9V）に達しない場合にも、供給電圧がその所定の値以上になるまで、ステップS15103の処理を繰り返し実行する。

## 【2131】

ステップS15105では、初期設定2を実行する（詳細は後述）。

## 【2132】

ステップ S 1 5 1 0 7 では、電断前（電源の遮断前）の状態に復帰するか否かの判定を行い、電断前の状態に復帰しない場合（主制御部 3 0 0 の基本回路 3 0 2 を初期状態にする場合）には初期化処理（ステップ S 1 5 1 1 3）に進む。

#### 【 2 1 3 3 】

具体的には、最初に、電源基板に設けた R W M クリアスイッチ 1 8 0 を遊技店の店員などが操作した場合に送信される R A M クリア信号がオン（操作があったことを示す）であるか否か、すなわち R A M クリアが必要であるか否かを判定し、R A M クリア信号がオンの場合（R A M クリアが必要な場合）には、基本回路 3 0 2 を初期状態にすべくステップ S 1 5 1 1 3 に進む。一方、R A M クリア信号がオフの場合（R A M クリアが必要でない場合）には、R A M 3 0 8 に設けた電源ステータス記憶領域に記憶した電源ステータスの情報を読み出し、この電源ステータスの情報がサスペンドを示す情報であるか否かを判定する。そして、電源ステータスの情報がサスペンドを示す情報でない場合には、基本回路 3 0 2 を初期状態にすべくステップ S 1 5 1 1 3 に進み、電源ステータスの情報がサスペンドを示す情報である場合には、R A M 3 0 8 の所定の領域（例えば全ての領域）に記憶している 1 バイトデータを初期値が 0 である 1 バイト構成のレジスタに全て加算することによりチェックサムを算出し、算出したチェックサムの結果が特定の値（例えば 0）であるか否か（チェックサムの結果が正常であるか否か）を判定する。そして、チェックサムの結果が特定の値（例えば 0）の場合（チェックサムの結果が正常である場合）には、電断前の状態に復帰すべくステップ S 1 5 1 0 9 に進んでから、その後の処理を遅延させるべくステップ S 1 5 1 1 1 に進む。チェックサムの結果が特定の値（例えば 0）以外である場合（チェックサムの結果が異常である場合）には、パチンコ機 1 0 0 を初期状態にすべくステップ S 1 5 1 1 3 に進む。同様に電源ステータスの情報が「サスペンド」以外の情報を示している場合にもステップ S 1 5 1 1 3 に進む。

#### 【 2 1 3 4 】

ステップ S 1 5 1 0 9 では、復帰処理（復電時処理）を行う。この復帰処理では、電断時に R A M 3 0 8 に設けられたスタックポインタ退避領域に記憶しておいたスタックポインタの値を読み出し、スタックポインタに再設定（本設定）する。また、電断時に R A M 3 0 8 に設けられたレジスタ退避領域に記憶しておいた各レジスタの値を読み出し、各レジスタに再設定した後、割込許可の設定を行う。以降、C P U 3 0 4 が、再設定後のスタックポインタやレジスタに基づいて制御プログラムを実行する結果、パチンコ機 1 0 0 は電源断時の状態に復帰する。すなわち、電断直前にタイマ割込処理（後述）に分岐する直前に行った（ステップ S 1 5 1 1 7 ～ S 1 5 1 2 1 のいずれか）命令の次の命令から処理を再開する。また、図 4 に示す主制御部 3 0 0 における基本回路 3 0 2 に搭載されている R A M 3 0 8 には、送信情報記憶領域が設けられている。このステップ S 1 5 1 0 9 では、その送信情報記憶領域に、復電コマンドをセットする。この復電コマンドは、電源断時の状態に復帰したことを表すコマンドであり、後述する、主制御部 3 0 0 のタイマ割込処理におけるステップ S 1 5 2 3 3 において、第 1 副制御部 4 0 0 へ送信される。

#### 【 2 1 3 5 】

ステップ S 1 5 1 1 1 では、遅延処理を実行する（詳細は後述）。

#### 【 2 1 3 6 】

ステップ S 1 5 1 1 3 では、初期化処理を行う。この初期化処理では、割込禁止の設定、スタックポインタへのスタック初期値の設定（本設定）、R A M 3 0 8 の全ての記憶領域の初期化などを行う。さらにここで、主制御部 3 0 0 の R A M 3 0 8 に設けられた送信情報記憶領域に正常復帰コマンドをセットする。この正常復帰コマンドは、主制御部 3 0 0 の初期化処理（ステップ S 1 5 1 1 3）が行われたことを表すコマンドであり、復電コマンドと同じく、主制御部 3 0 0 のタイマ割込処理におけるステップ S 1 5 2 3 3 において、第 1 副制御部 4 0 0 へ送信される。

#### 【 2 1 3 7 】

ステップ S 1 5 1 1 5 では、遅延処理を実行する（詳細は後述）。

#### 【 2 1 3 8 】

ステップ S 1 5 1 1 7 では、割込禁止の設定を行った後、基本乱数初期値更新処理を行う。この基本乱数初期値更新処理では、普図当選乱数カウンタ、および特図乱数値カウンタの初期値をそれぞれ生成するための 2 つの初期値生成用乱数カウンタと、普図タイマ乱数値、および特図タイマ乱数値それぞれを生成するための 2 つの乱数カウンタを更新する。例えば、普図タイマ乱数値として取り得る数値範囲が 0 ～ 1 0 0 とすると、R A M 3 0 8 に設けた普図タイマ乱数値を生成するための乱数カウンタ記憶領域から値を取得し、取得した値に 1 を加算してから元の乱数カウンタ記憶領域に記憶する。このとき、取得した値に 1 を加算した結果が 1 0 1 であれば 0 を元の乱数カウンタ記憶領域に記憶する。他の初期値生成用乱数カウンタ、乱数カウンタもそれぞれ同様に更新する。なお、初期値生成用乱数カウンタは、後述するステップ S 1 5 2 0 7 でも更新する。なお、ステップ S 1 5 1 1 7 では、基本乱数初期値更新処理の他に後述する基本乱数更新処理を行ってもよいし、基本乱数更新処理のみを行ってもよい。

10

**【 2 1 3 9 】**

ステップ S 1 5 1 1 9 では、コマンド送信処理を行い、各種のコマンドが第 1 副制御部 4 0 0 に送信される。

**【 2 1 4 0 】**

ステップ S 1 5 1 2 1 では、割り込み許可の設定を行った後、その他判定乱数更新処理を行う。この判定乱数では、特図変動時間決定用乱数、普図変動時間決定用乱数を生成するための乱数カウンタが更新される。

20

**【 2 1 4 1 】**

主制御部 3 0 0 は、所定の周期ごとに開始するタイマ割込処理を行っている間を除いて、ステップ S 1 5 1 1 7 ～ S 1 5 1 2 1 の処理を繰り返し実行する。

**【 2 1 4 2 】****< 初期設定 2 >**

次に、図 2 7 2 ( c ) を用いて、主制御部 3 0 0 の C P U 3 0 4 が実行する初期設定 2 について説明する。なお、同図は主制御部メイン処理の電源投入時処理における初期設定 2 の流れを示すフローチャートである。

**【 2 1 4 3 】**

ステップ S 1 5 1 3 1 では、W D T を起動させる処理を行う。ここでは、W D T の起動許可及び初期値の設定等を行う。なお、本実施形態では、W D T に、初期値として 3 2 . 8 m s に相当する数値を設定する。

30

**【 2 1 4 4 】**

ステップ S 1 5 1 3 3 では、乱数生成回路初期設定処理を行い、最大値設定で起動される乱数生成チャンネルの乱数生成回路が起動される。最大値設定レジスタに生成する乱数値の最大値を設定した後、乱数生成回路が起動される。起動された乱数生成回路は、所定の初期値から最大値までの範囲内で乱数値の更新を開始する。

**【 2 1 4 5 】**

ステップ S 1 5 1 3 5 では、後述する主制御部タイマ割込処理を定期毎に実行するための周期を決める数値をタイマ回路に設定する処理を行う。

**【 2 1 4 6 】**

ステップ S 1 5 1 3 7 では、その他の設定処理を行う。その他の設定処理後に、初期設定 2 を終了する。

40

**【 2 1 4 7 】**

初期設定 2 で行われるステップ S 1 5 1 3 1 ～ S 1 5 1 3 7 の各処理の順番は任意に設定することができる。なお、初期設定 2 で行う処理の一部または全部を、ステップ S 1 5 1 0 1 で行うようにしてもよい。

**【 2 1 4 8 】****< 遅延処理 >**

次に、図 2 7 3 を用いて、主制御部 3 0 0 の C P U 3 0 4 が実行する遅延処理について説明する。なお、同図は主制御部メイン処理における遅延処理の流れを示すフローチャ

50

トである。

【2149】

ステップS15141では、I/Oの所定の出力ポートからクリア信号を送信する処理を実行する。

【2150】

ステップS15143では、第1副制御部400の起動を待つ周期を決める副制御部起動待ちカウンタを初期設定（例えば90を設定）する。

【2151】

ステップS15145では、割込み機能限定カウンタを初期設定（例えば250を設定）する。割込み機能限定カウンタは、4ms毎のタイマ割込み処理で減算される。

10

【2152】

ステップS15147では、基本乱数初期値更新処理を実行する（詳細は後述）。なお、ステップS15147では、基本乱数初期値更新処理の他に後述する基本乱数更新処理を行ってもよいし、基本乱数更新処理のみを行ってもよい。

【2153】

ステップS15149では、コマンド送信処理を行い、各種のコマンドが第1副制御部400に送信される。

【2154】

ステップS15151では、割込み機能限定カウンタが0であるか否かの判定を行い、割込み機能限定カウンタが0以外の場合、ステップS15147の処理に戻る。一方、割込み機能限定カウンタが0の場合、すなわち、ステップS15145において割込み機能限定カウンタに250が設定され、4ms毎のタイマ割込み処理で減算されて1000ms = 1秒経過した場合、ステップS15153に進む。

20

【2155】

ステップS15153では、副制御部起動待ちカウンタを1減算する。

【2156】

ステップS15155では、副制御部起動待ちカウンタが0であるか否かの判定を行い、副制御部起動待ちカウンタが0以外の場合、ステップS15145に戻る。一方、副制御部起動待ちカウンタが0の場合、すなわち、ステップS15143において副制御部起動待ちカウンタに90が設定され、90秒経過した場合、ステップS15157に進む。

30

【2157】

ステップS15157では、その他の処理を実行する。

【2158】

<主制御部タイマ割込処理>

次に、図274を用いて、主制御部300のCPU304が実行する主制御部タイマ割込処理について説明する。なお、同図は主制御部タイマ割込処理の流れを示すフローチャートである。

【2159】

主制御部300は、所定の周期（本実施形態では約2msに1回）でタイマ割込信号を発生するカウンタタイマ312を備えており、このタイマ割込信号を契機として主制御部タイマ割込処理を所定の周期で開始する。

40

【2160】

ステップS15201では、タイマ割込開始処理を行う。このタイマ割込開始処理では、CPU304の各レジスタの値をスタック領域に一時的に退避する処理などを行う。

【2161】

ステップS15203では、WDT314のカウント値が初期設定値（本実施形態では32.8ms）を超えてWDT割込が発生しないように（処理の異常を検出しないように）、WDTを定期的に（本実施形態では、主制御部タイマ割込の周期である約2msに1回）リスタートを行う。

【2162】

50



ステップ S 1 5 2 0 5 では、入力ポート状態更新処理を行う。この入力ポート状態更新処理では、I / O 3 1 0 の入力ポートを介して、上述の前面枠扉開放センサや内枠開放センサや下皿満タンセンサ、各種の球検出センサを含む各種センサ 3 2 0 の検出信号を入力して検出信号の有無を監視し、R A M 3 0 8 に各種センサ 3 2 0 ごとに区画して設けた信号状態記憶領域に記憶する。球検出センサの検出信号を例にして説明すれば、前々回のタイマ割込処理（約 4 m s 前）で検出した各々の球検出センサの検出信号の有無の情報を、R A M 3 0 8 に各々の球検出センサごとに区画して設けた前回検出信号記憶領域から読み出し、この情報を R A M 3 0 8 に各々の球検出センサごとに区画して設けた前々回検出信号記憶領域に記憶し、前回のタイマ割込処理（約 2 m s 前）で検出した各々の球検出センサの検出信号の有無の情報を、R A M 3 0 8 に各々の球検出センサごとに区画して設けた  
10 今回検出信号記憶領域から読み出し、この情報を上述の前回検出信号記憶領域に記憶する。また、今回検出した各々の球検出センサの検出信号を、上述の今回検出信号記憶領域に記憶する。

#### 【 2 1 6 3 】

また、ステップ S 1 5 2 0 5 では、上述の前々回検出信号記憶領域、前回検出信号記憶領域、および今回検出信号記憶領域の各記憶領域に記憶した各々の球検出センサの検出信号の有無の情報を比較し、各々の球検出センサにおける過去 3 回分の検出信号の有無の情報が入賞判定パターン情報と一致するか否かを判定する。一個の遊技球が一つの球検出センサを通過する間に、約 2 m s という非常に短い間隔で起動を繰り返すこの主制御部タイマ割込処理は何回か起動する。このため、主制御部タイマ割込処理が起動する度に、上述の  
20 ステップ S 1 5 2 0 5 では、同じ遊技球が同じ球検出センサを通過したことを表す検出信号を確認することになる。この結果、上述の前々回検出信号記憶領域、前回検出信号記憶領域、および今回検出信号記憶領域それぞれに、同じ遊技球が同じ球検出センサを通過したことを表す検出信号が記憶される。すなわち、遊技球が球検出センサを通過し始めたときには、前々回検出信号無し、前回検出信号有り、今回検出信号有りになる。本実施形態では、球検出センサの誤検出やノイズを考慮して、検出信号無しの後に検出信号が連続して 2 回記憶されている場合には、入賞があったと判定する。図 4 に示す主制御部 3 0 0 の R O M 3 0 6 には、入賞判定パターン情報（本実施形態では、前々回検出信号無し、前回検出信号有り、今回検出信号有りであることを示す情報）が記憶されている。このステップ  
30 S 1 5 2 0 5 では、各々の球検出センサにおいて過去 3 回分の検出信号の有無の情報が、予め定めた入賞判定パターン情報（本実施形態では、前々回検出信号無し、前回検出信号有り、今回検出信号有りであることを示す情報）と一致した場合に、一般入賞口 2 2 6 、可変入賞口 2 3 4 、第 1 特図始動口 2 3 0 、および第 2 特図始動口 2 3 2 への入球、または普図始動口 2 2 8 の通過があったと判定する。すなわち、これらの入賞口 2 2 6 、2 3 4 やこれらの始動口 2 3 0 、2 3 2 、2 2 8 への入賞があったと判定する。例えば、一般入賞口 2 2 6 への入球を検出する一般入賞口センサにおいて過去 3 回分の検出信号の有無の情報が上述の入賞判定パターン情報と一致した場合には、一般入賞口 2 2 6 へ入賞があったと判定し、以降の一般入賞口 2 2 6 への入賞に伴う処理を行うが、過去 3 回分の検出信号の有無の情報が上述の入賞判定パターン情報と一致しなかった場合には、以降の一般入賞口 2 2 6 への入賞に伴う処理を行わずに後続の処理に分岐する。なお、主制御部 3 0  
40 0 の R O M 3 0 6 には、入賞判定クリアパターン情報（本実施形態では、前々回検出信号有り、前回検出信号無し、今回検出信号無しであることを示す情報）が記憶されている。入賞が一度あったと判定した後は、各々の球検出センサにおいて過去 3 回分の検出信号の有無の情報が、その入賞判定クリアパターン情報に一致するまで入賞があったとは判定せず、入賞判定クリアパターン情報に一致すれば、次からは上記入賞判定パターン情報に一致するか否かの判定を行う。

#### 【 2 1 6 4 】

ステップ S 1 5 2 0 7 およびステップ S 1 5 2 0 9 では、基本乱数初期値更新処理および基本乱数更新処理を行う。これらの基本乱数初期値更新処理および基本乱数更新処理では、上記ステップ S 1 5 1 1 7 で行った初期値生成用乱数カウンタの値の更新を行い、次  
50

に主制御部 300 で使用する、普図当選乱数値、特図 1 乱数値、および特図 2 乱数値をそれぞれ生成するための 2 つの乱数カウンタを更新する。例えば、普図当選乱数値として取り得る数値範囲が 0 ～ 100 とすると、RAM 308 に設けた普図当選乱数値を生成するための乱数カウンタ記憶領域から値を取得し、取得した値に 1 を加算してから元の乱数カウンタ記憶領域に記憶する。このとき、取得した値に 1 を加算した結果が 101 であれば 0 を元の乱数カウンタ記憶領域に記憶する。また、取得した値に 1 を加算した結果、乱数カウンタが一周していると判定した場合にはそれぞれの乱数カウンタに対応する初期値生成用乱数カウンタの値を取得し、乱数カウンタの記憶領域にセットする。例えば、0 ～ 100 の数値範囲で変動する普図当選乱数値生成用の乱数カウンタから値を取得し、取得した値に 1 を加算した結果が、RAM 308 に設けた所定の初期値記憶領域に記憶している  
10  
前回設定した初期値と等しい値（例えば 7）である場合に、普図当選乱数値生成用の乱数カウンタに対応する初期値生成用乱数カウンタから値を初期値として取得し、普図当選乱数値生成用の乱数カウンタにセットすると共に、普図当選乱数値生成用の乱数カウンタが次に 1 周したことを判定するために、今回設定した初期値を上述の初期値記憶領域に記憶しておく。また、普図当選乱数値生成用の乱数カウンタが次に 1 周したことを判定するための上述の初期値記憶領域とは別に、特図乱数生成用の乱数カウンタが 1 周したことを判定するための初期値記憶領域を RAM 308 に設けている。なお、本実施形態では特図 1 の乱数値を取得するためのカウンタと特図 2 の乱数値を取得するためのカウンタとを別に設けたが、同一のカウンタを用いてもよい。

#### 【2165】

ステップ S15211 では、演出乱数更新処理を行う。この演出乱数更新処理では、主制御部 300 で使用する演出用乱数値を生成するための乱数カウンタを更新する。

#### 【2166】

ステップ S15212 では、割込み機能限定カウンタが 0 であるか否かの判定を行い、割込み機能限定カウンタが 0 以外の場合、ステップ S15213 に進む。一方、割込み機能限定カウンタが 0 の場合、ステップ S15214 に進む。

#### 【2167】

ステップ S15213 では、割込み機能限定カウンタを 1 減算し、その後、ステップ S15239 に進む。

#### 【2168】

ステップ S15214 では、特定異常検出フラグ（詳しくは後述する。）が ON であるか否かの判定を行い、特定異常検出フラグが ON の場合、ステップ S15239 に進む。一方、特定異常検出フラグが OFF の場合、ステップ S15215 に進む。

#### 【2169】

ステップ S15215 では、タイマ更新処理を行う。このタイマ更新処理では、普通図柄表示装置 210 に図柄を変動・停止表示する時間を計時するための普図表示図柄更新タイマ、第 1 特別図柄表示装置 212 に図柄を変動・停止表示する時間を計時するための特図 1 表示図柄更新タイマ、第 2 特図表示装置 214 に図柄を変動・停止表示する時間を計時するための特図 2 表示図柄更新タイマ、所定の入賞演出時間、所定の開放時間、所定の閉鎖時間、所定の終了演出期間などを計時するためのタイマなどを含む各種タイマを更新  
40  
する。

#### 【2170】

ステップ S15216 では、入賞口カウンタ更新処理を行う。この入賞口カウンタ更新処理では、入賞口 226、234 や始動口 230、232、228 に入賞があった場合に、RAM 308 に各入賞口ごと、あるいは各始動口ごとに設けた賞球数記憶領域の値を読み出し、1 を加算して、元の賞球数記憶領域に設定する。

#### 【2171】

また、ステップ S15217 では、入賞受付処理を行う。この入賞受付処理では、第 1 特図始動口 230、第 2 特図始動口 232、普図始動口 228 および可変入賞口 234 へ  
50  
の入賞があったか否かを判定する。ここでは、ステップ S15205 における入賞判定パ

ターン情報と一致するか否かの判定結果を用いて判定する。第1特図始動口230へ入賞があった場合且つRAM308に設けた対応する保留数記憶領域が満タンでない場合、カウンタ回路318の当選用カウンタ値記憶用レジスタから値を特図1当選乱数値として取得するとともに特図1乱数値生成用の乱数カウンタから値を特図1乱数値として取得して対応する乱数値記憶領域に格納する。第2特図始動口232へ入賞があった場合且つRAM308に設けた対応する保留数記憶領域が満タンでない場合、カウンタ回路318の当選用カウンタ値記憶用レジスタから値を特図2当選乱数値として取得するとともに特図2乱数値生成用の乱数カウンタから値を特図2乱数値として取得して対応する乱数値記憶領域に格納する。普図始動口228へ入賞があった場合且つRAM308に設けた対応する保留数記憶領域が満タンでない場合、普図当選乱数値生成用の乱数カウンタから値を普図当選乱数値として取得して対応する乱数値記憶領域に格納する。可変入賞口234へ入賞があった場合には、可変入賞口用の入賞記憶領域に、可変入賞口234に球が入球したことを示す情報を格納する。

10

#### 【2172】

ステップS15219では、払出要求数送信処理を行う。なお、払出制御部600に出力する出力予定情報および払出要求情報は、例えば1バイトで構成しており、ビット7にストロブ情報（オンの場合、データをセットしていることを示す）、ビット6に電源投入情報（オンの場合、電源投入後一回目のコマンド送信であることを示す）、ビット4～5に暗号化のための今回加工種別（0～3）、およびビット0～3に暗号化加工後の払出要求数を示すようにしている。

20

#### 【2173】

ステップS15221では、普図状態更新処理を行う。この普図状態更新処理は、普図の状態に対応する複数の処理のうちの1つの処理を行う。例えば、普図変動表示の途中（上述する普図表示図柄更新タイマの値が1以上）における普図状態更新処理では、普通図柄表示装置210を構成する7セグメントLEDの点灯と消灯を繰り返す点灯・消灯駆動制御を行う。この制御を行うことで、普通図柄表示装置210は普図の変動表示（普図変動遊技）を行う。

#### 【2174】

また、普図変動表示時間が経過したタイミング（普図表示図柄更新タイマの値が1から0になったタイミング）における普図状態更新処理では、当りフラグがオンの場合には、当り図柄の表示態様となるように普通図柄表示装置210を構成する7セグメントLEDの点灯・消灯駆動制御を行い、当りフラグがオフの場合には、外れ図柄の表示態様となるように普通図柄表示装置210を構成する7セグメントLEDの点灯・消灯駆動制御を行う。また、主制御部300のRAM308には、普図状態更新処理に限らず各種の処理において各種の設定を行う設定領域が用意されている。ここでは、上記点灯・消灯駆動制御を行うとともに、その設定領域に普図停止表示中であることを示す設定を行う。この制御を行うことで、普通図柄表示装置210は、当り図柄（図5（c）に示す普図A）および外れ図柄（図5（c）に示す普図B）いずれか一方の図柄の確定表示を行う。さらにその後、所定の停止表示期間（例えば500m秒間）、その表示を維持するためにRAM308に設けた普図停止時間管理用タイマの記憶領域に停止期間を示す情報を設定する。この設定により、確定表示された図柄が所定期間停止表示され、普図変動遊技の結果が遊技者に報知される。

30

40

#### 【2175】

また、普図変動遊技の結果が当りであれば、後述するように、普図当りフラグがオンされる。この普図当りフラグがオンの場合には、所定の停止表示期間が終了したタイミング（普図停止時間管理用タイマの値が1から0になったタイミング）における普図状態更新処理では、RAM308の設定領域に普図作動中を設定するとともに、所定の開放期間（例えば2秒間）、第2特図始動口232の羽根部材232aの開閉駆動用のソレノイド（332）に、羽根部材232aを開放状態に保持する信号を出力するとともに、RAM308に設けた羽根開放時間管理用タイマの記憶領域に開放期間を示す情報を設定する。

50

## 【2176】

また、所定の開放期間が終了したタイミング（羽根開放時間管理用タイマの値が1から0になったタイミング）で開始する普図状態更新処理では、所定の閉鎖期間（例えば500m秒間）、羽根部材の開閉駆動用のソレノイド332に、羽根部材を閉鎖状態に保持する信号を出力するとともに、RAM308に設けた羽根閉鎖時間管理用タイマの記憶領域に閉鎖期間を示す情報を設定する。

## 【2177】

また、所定の閉鎖期間が終了したタイミング（羽根閉鎖時間管理用タイマの値が1から0になったタイミング）で開始する普図状態更新処理では、RAM308の設定領域に普図非作動中を設定する。さらに、普図変動遊技の結果が外れであれば、後述するように、  
10 普図外れフラグがオンされる。この普図外れフラグがオンの場合には、上述した所定の停止表示期間が終了したタイミング（普図停止時間管理用タイマの値が1から0になったタイミング）における普図状態更新処理でも、RAM308の設定領域に普図非作動中を設定する。普図非作動中の場合における普図状態更新処理では、何もせずに次のステップS15223に移行するようにしている。

## 【2178】

ステップS15223では、普図関連抽選処理を行う。この普図関連抽選処理では、普図変動遊技および第2特図始動口232の開閉制御を行っておらず（普図の状態が非作動中）、且つ、保留している普図変動遊技の数が1以上である場合に、上述の乱数値記憶領域に記憶している普図当選乱数値に基づいた乱数抽選により普図変動遊技の結果を当選とするか、不当選とするかを決定する当り判定をおこない、当選とする場合にはRAM308に設けた当りフラグにオンを設定する。不当選の場合には、当りフラグにオフを設定する。また、当り判定の結果に関わらず、次に上述の普図タイマ乱数値生成用の乱数カウンタの値を普図タイマ乱数値として取得し、取得した普図タイマ乱数値に基づいて複数の変動時間のうちから普図表示装置210に普図を変動表示する時間を1つ選択し、この変動表示時間を、普図変動表示時間として、RAM308に設けた普図変動時間記憶領域に記憶する。なお、保留している普図変動遊技の数は、RAM308に設けた普図保留数記憶領域に記憶するようにしており、当り判定をするたびに、保留している普図変動遊技の数から1を減算した値を、この普図保留数記憶領域に記憶し直すようにしている。また当り判定に使用した乱数値を消去する。  
20  
30

## 【2179】

ステップS15224では、特図先読み処理を行う。この先読み処理では、まず、RAM308に設けられた特図1の保留記憶部内の特図1当選乱数を先読みするか、あるいは特図2の保留記憶部内の特図2当選乱数を先読みする。なお、ここでの先読みとは始動情報を当否判定（本抽選）の前に先に読むことを意味するが、以降の先読み処理では、先読みという言葉を用いて、先（当否判定（本抽選）の結果）を読むという意味で使うことがある。このステップS15224では、後述する特図2関連抽選処理、特図1関連抽選処理（ステップS15229、ステップS15231）で用いる特図抽選テーブルの内容と同じ内容の事前判定用テーブルを用い、先読みした特図当選乱数に基づく当否判定の事前判定を行う。なお、特図2関連抽選処理、特図1関連抽選処理では、特図抽選テーブルを用いて特図変動遊技の当否判定を改めて行い、ここでの判定結果は、あくまで事前判定の結果になる。当否判定の事前判定では、「大当り」という結果か、あるいは「大当り」以外という結果が導出され、「大当り」という結果の場合には、RAM308に設けられた特図1の保留記憶部内の大当り用特図1乱数を先読みするか、あるいは特図2の保留記憶部内の大当り用特図2乱数を先読みする。続いて、特図2関連抽選処理、特図1関連抽選処理（ステップS15229、ステップS15231）の特図関連処理で用いる停止図柄抽選テーブルの内容と同じ内容の事前判定用テーブルを用い、先読みした大当り用特図乱数に基づく特図の停止図柄の事前判定を行う。  
40

## 【2180】

なお、特図2関連抽選処理、特図1関連抽選処理では、停止図柄抽選テーブルを用いて  
50

特図の停止図柄の抽選を改めて行い、ここでの判定結果は、あくまで事前判定の結果になる。こうして、特図の停止図柄を事前判定すると、先読み予告の実行可否抽選を行う。この先読み予告は、特図2関連抽選処理、特図1関連抽選処理（ステップS15229、ステップS15231）が実行される前、すなわち当否判定が行われる前に、特図2関連抽選処理、特図1関連抽選処理で行われる当否判定の結果が大当たり（ここではより限定して15Rの大当たり（15R特別大当たりか15R大当たり））になることを予告するための報知である。ここでの先読み予告には、停止図柄の事前判定結果が15Rの大当たり図柄（特図Aまたは特図B）でなくても、15Rの大当たりになるかのように偽りで予告する偽の先読み予告も含まれる。すなわち、先読み予告は、当否判定の結果が15R大当たりになる可能性があることを表したり、遊技者に示唆する事前報知、あるいは当否判定の結果が15R大当たりになることを遊技者に期待させる事前報知であるといえる。先読み予告の実行可否抽選を行うタイミングで、RAM308に設けられた演出乱数カウンタから演出乱数（例えば、取り得る範囲は0～99）を取得し、取得した演出乱数に基づいて実行可否抽選を行う。なお、停止図柄の事前判定結果を第1副制御部400に送信し、この実行可否抽選は、第1副制御部400が行うようにしてもよい。

10

#### 【2181】

次いで、特図1および特図2それぞれについての特図状態更新処理を行うが、最初に、特図2についての特図状態更新処理（特図2状態更新処理）を行う（ステップS15225）。この特図2状態更新処理は、特図2の状態に応じて、次の8つの処理のうちの1つの処理を行う。例えば、特図2変動表示の途中（上述の特図2表示図柄更新タイマの値が1以上）における特図2状態更新処理では、第2特別図柄表示装置214を構成する7セグメントLEDの点灯と消灯を繰り返す点灯・消灯駆動制御を行う。この制御を行うことで、第2特別図柄表示装置214は特図2の変動表示（特図2変動遊技）を行う。

20

#### 【2182】

また、コマンド設定送信処理（ステップS15233）で回転開始設定送信処理を実行させることを示す所定の送信情報を上述の送信情報記憶領域に追加記憶してから処理を終了する。

#### 【2183】

また、主制御部300のRAM308には、15R大当たりフラグ、2R大当たりフラグ、第1小当たりフラグ、第2小当たりフラグ、第1はずれフラグ、第2はずれフラグ、特図確率変動フラグ、および普図確率変動フラグそれぞれのフラグが用意されている。特図2変動表示時間が経過したタイミング（特図2表示図柄更新タイマの値が1から0になったタイミング）で開始する特図2状態更新処理では、15R大当たりフラグはオン、特図確率変動フラグもオン、普図確率変動フラグもオンの場合には図5（a）に示す特図A、15R大当たりフラグはオン、特図確率変動フラグはオフ、普図確率変動フラグはオンの場合には特図B、2R大当たりフラグはオン、特図確率変動フラグもオン、普図確率変動フラグもオンの場合には特図C、2R大当たりフラグはオン、特図確率変動フラグはオフ、普図確率変動フラグはオンの場合には特図D、2R大当たりフラグはオン、特図確率変動フラグもオン、普図確率変動フラグはオンの場合には特図E、2R大当たりフラグはオン、特図確率変動フラグはオフ、普図確率変動フラグもオフの場合には特図F、第1小当たりフラグがオンの場合には特図G、第2小当たりフラグがオンの場合には特図H、第1はずれフラグがオンの場合には特図I、第2はずれフラグがオンの場合には特図Iそれぞれの態様となるように、第2特別図柄表示装置214を構成する7セグメントLEDの点灯・消灯駆動制御を行い、RAM308の設定領域に特図2停止表示中であることを表す設定を行う。この制御を行うことで、第2特別図柄表示装置214は、15R特別大当たり図柄（特図A）、15R大当たり図柄（特図B）、突然確変図柄（特図C）、突然時短図柄（特図D）、隠れ確変図柄（特図E）、突然通常図柄（特図F）、第1小当たり図柄（特図G）、第2小当たり図柄（特図H）、第1はずれ図柄（特図I）、および第2はずれ図柄（特図J）のいずれか一つの図柄の確定表示を行う。さらにその後、所定の停止表示期間（例えば500m秒間）その表示を維持するためにRAM308に設けた特図2停止時間管理用タイマの記憶領域に

30

40

50

停止期間を示す情報を設定する。この設定により、確定表示された特図 2 が所定期間停止表示され、特図 2 変動遊技の結果が遊技者に報知される。また、RAM 308 に設けられた時短回数記憶部に記憶された時短回数が 1 以上であれば、その時短回数から 1 を減算し、減算結果が 1 から 0 となった場合は、特図確率変動中（詳細は後述）でなければ、時短フラグをオフする。さらに、大当たり遊技中（特別遊技状態中）にも、時短フラグをオフする。

#### 【2184】

また、コマンド設定送信処理（ステップ S 15233）で回転停止設定送信処理を実行させることを示す所定の送信情報を上述の送信情報記憶領域に追加記憶するとともに、変動表示を停止する図柄が特図 2 であることを示す特図 2 識別情報を、後述するコマンドデータに含める情報として RAM 308 に追加記憶してから処理を終了する。

10

#### 【2185】

また、特図 2 変動遊技の結果が大当たりであれば、後述するように、大当たりフラグがオンされる。この大当たりフラグがオンの場合には、所定の停止表示期間が終了したタイミング（特図 2 停止時間管理用タイマの値が 1 から 0 になったタイミング）における特図 2 状態更新処理では、RAM 308 の設定領域に特図 2 作動中を設定するとともに、所定の入賞演出期間（例えば 3 秒間）すなわち装飾図柄表示装置 208 による大当たりを開始することを遊技者に報知する画像を表示している期間待機するために RAM 308 に設けた特図 2 待機時間管理用タイマの記憶領域に入賞演出期間を示す情報を設定する。また、コマンド設定送信処理（ステップ S 15233）で入賞演出設定送信処理を実行させることを示す所定の送信情報を上述の送信情報記憶領域に追加記憶する。

20

#### 【2186】

また、所定の入賞演出期間が終了したタイミング（特図 2 待機時間管理用タイマの値が 1 から 0 になったタイミング）で開始する特図 2 状態更新処理では、所定の開放期間（例えば 29 秒間、または可変入賞口 234 に所定球数（例えば 10 球）の遊技球の入賞を検出するまで）可変入賞口 234 の扉部材 234a の開閉駆動用のソレノイド（332）に、扉部材 234a を開放状態に保持する信号を出力するとともに、RAM 308 に設けた扉開放時間管理用タイマの記憶領域に開放期間を示す情報を設定する。また、コマンド設定送信処理（ステップ S 15233）で大入賞口開放設定送信処理を実行させることを示す所定の送信情報を上述の送信情報記憶領域に追加記憶する。

30

#### 【2187】

また、所定の開放期間が終了したタイミング（扉開放時間管理用タイマの値が 1 から 0 になったタイミング）で開始する特図 2 状態更新処理では、所定の閉鎖期間（例えば 1.5 秒間）可変入賞口 234 の扉部材 234a の開閉駆動用のソレノイド（332）に、扉部材 234a を閉鎖状態に保持する信号を出力するとともに、RAM 308 に設けた扉閉鎖時間管理用タイマの記憶領域に閉鎖期間を示す情報を設定する。また、コマンド設定送信処理（ステップ S 15233）で大入賞口閉鎖設定送信処理を実行させることを示す所定の送信情報を上述の送信情報記憶領域に追加記憶する。

#### 【2188】

また、この扉部材の開放・閉鎖制御を所定回数（本実施例では 15 ラウンドか 2 ラウンド）繰り返し、終了したタイミングで開始する特図 2 状態更新処理では、所定の終了演出期間（例えば 3 秒間）すなわち装飾図柄表示装置 208 による大当たりを終了することを遊技者に報知する画像を表示している期間待機するように設定するために RAM 308 に設けた演出待機時間管理用タイマの記憶領域に演出待機期間を示す情報を設定する。また、普図確率変動フラグがオンに設定されていれば、この大当たり遊技の終了と同時に、RAM 308 に設けられた時短回数記憶部に時短回数 100 回をセットするとともに、RAM 308 に設けられた時短フラグをオンする。なお、その普図確率変動フラグがオフに設定されていれば、時短回数記憶部に時短回数をセットすることなく、また時短フラグをオンすることもない。ここにいう時短とは、特図変動遊技における大当たりを終了してから、次の大当たりを開始するまでの時間を短くするため、パチンコ機が遊技者にとって有利な状態に

40

50

なることをいう。この時短フラグがオンに設定されていると、普図高確率状態である。普図高確率状態では普図低確率状態に比べて、普図変動遊技に大当たりする可能性が高い。また、普図高確率状態の方が、普図低確率状態に比べて普図変動遊技の変動時間および特図変動遊技の変動時間は短くなる。さらに、普図高確率状態では普図低確率状態に比べて、第2特別始動口232の一对の羽根部材232aの1回の開放における開放時間が長くなりやすい。加えて、普図高確率状態では普図低確率状態に比べて、一对の羽根部材232aは多く開きやすい。また、上述のごとく、時短フラグは、大当たり遊技中（特別遊技状態中）にはオフに設定される。したがって、大当たり遊技中には、普図低確率状態が維持される。これは、大当たり遊技中に普図高確率状態であると、大当たり遊技中に可変入賞口234に所定の個数、遊技球が入球するまでの間に第2特図始動口232に多くの遊技球が入球し、大当たり中に獲得することができる遊技球の数が多くなってしまい射幸性が高まってしまうという問題があり、これを解決するためのものである。

#### 【2189】

さらに、コマンド設定送信処理（ステップS15233）で終了演出設定送信処理を実行させることを示す所定の送信情報を上述の送信情報記憶領域に追加記憶する。

#### 【2190】

また、所定の終了演出期間が終了したタイミング（演出待機時間管理用タイマの値が1から0になったタイミング）で開始する特図2状態更新処理では、RAM308の設定領域に特図2非作動中を設定する。さらに、特図2変動遊技の結果が外れであれば、後述するように、はずれフラグがオンされる。このはずれフラグがオンの場合には、上述した所定の停止表示期間が終了したタイミング（特図2停止時間管理用タイマの値が1から0になったタイミング）における特図2状態更新処理でも、RAM308の設定領域に特図2非作動中を設定する。特図2非作動中の場合における特図2状態更新処理では、何もせずに次のステップS15227に移行するようにしている。

#### 【2191】

続いて、特図1についての特図状態更新処理（特図1状態更新処理）を行う（ステップS15227）。この特図1状態更新処理では、特図1の状態に応じて、上述の特図2状態更新処理で説明した各処理を行う。この特図1状態更新処理で行う各処理は、上述の特図2状態更新処理で説明した内容の「特図2」を「特図1」と読み替えた処理と同一であるため、その説明は省略する。なお、特図2状態更新処理と特図1状態更新処理の順番は逆でもよい。

#### 【2192】

ステップS15225およびステップS15227における特図状態更新処理が終了すると、今度は、特図1および特図2それぞれについての特図関連抽選処理を行う。ここでも先に、特図2についての特図関連抽選処理（特図2関連抽選処理）を行い（ステップS15229）、その後で、特図1についての特図関連抽選処理（特図1関連抽選処理）を行う（ステップS15231）。これらの特図関連抽選処理についても、主制御部300が特図2関連抽選処理を特図1関連抽選処理よりも先に行うことで、特図2変動遊技の開始条件と、特図1変動遊技の開始条件が同時に成立した場合でも、特図2変動遊技が先に変動中となるため、特図1変動遊技は変動を開始しない。また、装飾図柄表示装置208による、特図変動遊技の大当たり判定の結果の報知は、第1副制御部400によって行われ、第2特図始動口232への入賞に基づく抽選の抽選結果の報知が、第1特図始動口230への入賞に基づく抽選の抽選結果の報知よりも優先して行われる。

#### 【2193】

ステップS15232では、デバイス監視処理を実行する（詳細は後述）。

#### 【2194】

ステップS15233では、コマンド設定送信処理を行い、各種のコマンド（例えば、ステップS15232のデバイス監視処理で特定異常があった場合は、特定異常を示す信号）が第1副制御部400に送信される。なお、第1副制御部400に送信する出力予定情報は例えば16ビットで構成しており、ビット15はストロープ情報（オンの場合、デ

ータをセットしていることを示す)、ビット11~14はコマンド種別(本実施形態では、基本コマンド、図柄変動開始コマンド、図柄変動停止コマンド、入賞演出開始コマンド、終了演出開始コマンド、大当たりラウンド数指定コマンド、復電コマンド、F R A M (登録商標)クリアコマンドなどコマンドの種類を特定可能な情報)、ビット0~10はコマンドデータ(コマンド種別に対応する所定の情報)で構成している。

#### 【2195】

具体的には、ストロブ情報は上述のコマンド送信処理でオン、オフするようにしている。また、コマンド種別が図柄変動開始コマンドの場合であればコマンドデータに、15R大当たりフラグや2R大当たりフラグの値、特図確率変動フラグの値、特図関連抽選処理で選択したタイマ番号などを示す情報を含み、図柄変動停止コマンドの場合であれば、15R大当たりフラグや2R大当たりフラグの値、特図確率変動フラグの値などを含み、入賞演出コマンドおよび終了演出開始コマンドの場合であれば、特図確率変動フラグの値などを含み、大当たりラウンド数指定コマンドの場合であれば特図確率変動フラグの値、大当たりラウンド数などを含むようにしている。コマンド種別が基本コマンドを示す場合は、コマンドデータにデバイス情報、第1特図始動口230への入賞の有無、第2特図始動口232への入賞の有無、可変入賞口234への入賞の有無などを含む。

#### 【2196】

また、上述の回転開始設定送信処理では、コマンドデータにR A M 3 0 8に記憶している、15R大当たりフラグや2R大当たりフラグの値、特図確率変動フラグの値、特図1関連抽選処理および特図2関連抽選処理で選択したタイマ番号、保留している第1特図変動遊技または第2特図変動遊技の数などを示す情報を設定する。上述の回転停止設定送信処理では、コマンドデータにR A M 3 0 8に記憶している、15R大当たりフラグや2R大当たりフラグの値、特図確率変動フラグの値などを示す情報を設定する。上述の入賞演出設定送信処理では、コマンドデータに、R A M 3 0 8に記憶している、入賞演出期間中に装飾図柄表示装置208・各種ランプ418・スピーカ120に出力する演出制御情報、特図確率変動フラグの値、保留している第1特図変動遊技または第2特図変動遊技の数などを示す情報を設定する。上述の終了演出設定送信処理では、コマンドデータに、R A M 3 0 8に記憶している、演出待機期間中に装飾図柄表示装置208・各種ランプ418・スピーカ120に出力する演出制御情報、特図確率変動フラグの値、保留している第1特図変動遊技または第2特図変動遊技の数などを示す情報を設定する。上述の大入賞口開放設定送信処理では、コマンドデータにR A M 3 0 8に記憶している大当たりラウンド数、特図確率変動フラグの値、保留している第1特図変動遊技または第2特図変動遊技の数などを示す情報を設定する。上述の大入賞口閉鎖設定送信処理では、コマンドデータにR A M 3 0 8に記憶している大当たりラウンド数、特図確率変動フラグの値、保留している第1特図変動遊技または第2特図変動遊技の数などを示す情報を設定する。また、このステップS15233では一般コマンド特図保留増加処理も行われる。この一般コマンド特図保留増加処理では、コマンドデータにR A M 3 0 8の送信用情報記憶領域に記憶している特図識別情報(特図1または特図2を示す情報)、予告情報(事前予告情報、偽事前予告情報、または事前予告無情報のいずれか)を設定する。

#### 【2197】

第1副制御部400では、受信した出力予定情報に含まれるコマンド種別により、主制御部300における遊技制御の変化に応じた演出制御の決定が可能になるとともに、出力予定情報に含まれているコマンドデータの情報に基づいて、演出制御内容を決定することができるようになる。

#### 【2198】

ステップS15235では、外部出力信号設定処理を行う。具体的には、外部端子板信号や試射試験信号などを設定する。また、ステップS15232のデバイス監視処理で特定異常があった場合は、特定異常を示す信号を設定する。この外部出力信号設定処理では、R A M 3 0 8に記憶している遊技情報を、情報出力回路336を介してパチンコ機100とは別体の情報入力回路350に出力する。



## 【2199】

ステップS15239では、低電圧信号がオンであるか否かを監視する。そして、低電圧信号がオンの場合（電源の遮断を検知した場合）にはステップS15243に進み、低電圧信号がオフの場合（電源の遮断を検知していない場合）にはステップS15241に進む。

## 【2200】

ステップS15241では、タイマ割込終了処理を行う。このタイマ割込終了処理では、ステップS15201で一時的に退避した各レジスタの値を元の各レジスタに設定したり、割込許可の設定などを行い、その後、図272に示す主制御部メイン処理に復帰する。

10

## 【2201】

一方、ステップS15243では、復電時に電断時の状態に復帰するための特定の変数やスタックポインタを復帰データとしてRAM308の所定の領域に退避し、入出力ポートの初期化等の電断処理を行い、その後、上述の主制御部メイン処理に復帰する。

## 【2202】

なお、デバイス監視処理（ステップS15232）で監視するエラーとしては、次に示すようなものが考えられる。

## 【2203】

エラー種別として、下受け皿満タンエラー（下皿満タンエラー）、払出装置エラー、払出超過エラー、不正払出エラー、払出個数スイッチエラー、主制御通信エラー、CRユニット未接続エラー、CRユニット通信エラー、磁気異常エラー、外部クロック異常エラー、磁界異常エラー、スイッチレベル異常エラー、枠開放エラー（前面枠扉開放エラー）、スイッチ未接続エラー、衝撃センサエラーなどがある。

20

## 【2204】

下受け皿満タンエラーとは、下皿満タンスイッチ信号のオン状態が、メイン基板の所定の入力ポートによってタイマ割り込みでオンを1回読み込んだとき（オフ状態からオン状態に切り換わるとき）にメインから払出に出力されるものである。

## 【2205】

払出装置エラーとは、払出モータが払出要求数分の駆動を終了後（実際は+ブレーキ期間+無励磁期間経過後）の次の割り込み内で、払出数（払出個数スイッチのカウント数）が払出要求数よりも少ない場合に払出装置エラーの発生を検知するものである。1個ずつ遊技球を払い出すリトライ処理が完了することにより、当該エラー状態が解除される。

30

## 【2206】

払出超過エラーとは、払出要求数に対して実際に払い出された遊技球が10個多く検知されたとき払出動作を停止するものである。この払出超過エラーは、エラー解除スイッチ168によって解除するエラーであり、エラー解除スイッチ168を操作するか、RAMクリアスイッチ180による初期化（RAMクリア）によって当該エラー状態が解除される。

## 【2207】

不正払出エラーとは、払出要求が発生していない状態で遊技球の払出を検知（払出個数スイッチがカウント）したとき払出動作を停止するものである。この不正払出エラーは、エラー解除スイッチ168によって解除するエラーであり、エラー解除スイッチ168を操作するか、RAMクリアスイッチ180による初期化（RAMクリア）によって当該エラー状態が解除される。

40

## 【2208】

払出個数スイッチエラーとは、払出基板170に制御される遊技球の払出個数スイッチ（不図示）のエラーであり、払出個数スイッチの接続異常を検知したとき（メイン基板は関係なし）払出動作を停止するものである。この払出個数スイッチエラーは、エラー解除スイッチ168によって解除するエラーであり、スイッチの接続が正しいか確認した後、エラー解除スイッチ168を操作するか、RAMクリアスイッチ180による初期化（R

50

A Mクリア)によって当該エラー状態が解除される。

#### 【2209】

主制御通信エラーとは、主制御接続確認信号のオフ状態を検知した場合、または主制御動作確認信号のオン状態を2割り込み継続して検知した場合に発生するエラーであり、メイン基板からのコマンド受信が不能になるものである。但し、払出要求数が残存している場合は、その分までは払い出す。主制御接続確認信号のオン状態を検知した場合で、かつ、主制御動作確認信号のオフ状態を2割り込み継続して検知した場合に当該エラー状態が解除される。

#### 【2210】

C Rユニット未接続エラーとは、C Rユニット未接続エラーが解除状態のときに、C Rユニット接続信号(V L信号)のオフ状態を検知し、16ms継続した場合に発生するものであり、貸出停止(賞球の払出には影響なし)となるものである。C Rユニット接続信号のON状態を検知し、16ms継続した場合に当該エラー状態が解除される。なお、エラー解除した後、C RユニットREADY信号(B R D Y信号)およびC Rユニット貸出要求完了確認信号(B R Q信号)がオフ状態となるまでC Rユニットとの通信が無効状態になる。

#### 【2211】

ここで、C Rユニットとインタフェースの接続信号関係について説明する。各種信号が、C Rユニットからインタフェースを介して払出基板へと伝達される場合、C Rユニット接続信号とは、C Rユニット内の+18V電源からなるV L信号である。C RユニットR E A D Y信号とは、C Rユニットが貸し出しの処理中であることを伝達するB R D Y信号である。C Rユニット貸出要求確認信号とは、C Rユニットが基本単位分25個の貸出要求と貸出指示を伝達するB R Q信号である。

#### 【2212】

C Rユニット通信エラーとは、貸出要求発生前にB R Q信号のオン状態を検知した場合に発生するものである(なお、他の条件もあるが、ここでは特に関係がないため割愛する)。遊技機側から通信異常を通知した後、B R D Y信号およびB R Q信号をオフ状態にすることで当該エラー状態が解除される。

#### 【2213】

磁気異常エラーとは、所定量を超える磁力を検出したとき主制御動作を停止するものである。外部クロック異常エラーとは、乱数更新用クロックに異常を検出した場合に発生するエラーである。磁界異常エラーとは、所定量を超える磁界を検出した場合に発生するエラーである。

#### 【2214】

スイッチレベル異常エラーとは、入賞スイッチを検出し、1s継続した場合に発生するエラーである。枠開放エラーとは、ガラス枠または内枠の開放を検出した場合に発生するエラーである。スイッチ未接続エラーとは、入賞スイッチ(不図示)のエラーであり、入賞スイッチの接続異常を検出した場合に発生するエラーである。

#### 【2215】

衝撃センサエラーとは、衝撃センサが所定量を超える衝撃を検出したとき電源断まで継続してエラーを報知するものである。

#### 【2216】

<デバイス監視処理>

次に、図275を用いて、主制御部300のC P U 304が実行するデバイス監視処理について説明する。なお、同図は主制御部タイマ割込処理におけるデバイス監視処理の流れを示すフローチャートである。

#### 【2217】

ステップS15251では、上述のエラーの有無を示すデバイス情報を初期設定する。

#### 【2218】

ステップS15253では、信号状態記憶領域に記憶した各種センサの信号状態を読み

10

20

30

40

50

出して、上述のエラーの有無を監視し、当該エラーを検出した場合に、ステップ S 1 5 2 5 5 に進む。一方、上述のエラーを検出しなかった場合には、当該デバイス監視処理を終了する。

【2 2 1 9】

ステップ S 1 5 2 5 5 では、第 1 副制御部 4 0 0 に送信すべき送信情報に、ステップ S 1 5 2 5 3 で検出したエラーを示す異常情報（デバイス情報）を設定する。

【2 2 2 0】

ステップ S 1 5 2 5 7 では、ステップ S 1 5 2 5 5 で設定した異常情報の中に、特定異常情報があるか否かを判定し、特定異常情報がある場合に、ステップ S 1 5 2 5 9 に進む。一方、特定異常情報がない場合に、ステップ S 1 5 2 6 1 に進む。なお、特定異常情報は、上述のエラーについての異常情報の中から、予め設定されている 1 つまたは複数のエラーの異常情報のことである。例えば、磁気異常エラーの異常情報を特定異常情報として予め設定しておいてもよい。

10

【2 2 2 1】

ステップ S 1 5 2 5 9 では、特定異常検出フラグをオンにする。

【2 2 2 2】

ステップ S 1 5 2 6 1 では、デバイスコマンド送信要求を設定する。

【2 2 2 3】

次に、図 2 7 6 を用いて本実施形態の遊技台で使用される乱数と、その主な導出源について説明する。同図は、本実施形態の遊技台で使用される乱数の導出源を表で示す図である。

20

【2 2 2 4】

特図 1 当選乱数は、1 6 ビット乱数回路 C H で生成されハードラッチ乱数値レジスタ 0 でラッチされた乱数値と 1 6 ビット乱数回路 C H 1 で生成されソフトラッチ乱数値レジスタでラッチされた乱数値とを合成したものが基になる。これらの乱数値が生成される乱数生成回路は、外部クロック信号に基づいて更新される。また、生成される乱数値の最大値設定は無しに設定されている。

【2 2 2 5】

特図 2 当選乱数は、1 6 ビット乱数回路 C H 0 で生成されハードラッチ乱数値レジスタ 1 でラッチされた乱数値と 1 6 ビット乱数回路 C H 2 で生成されソフトラッチ乱数値レジスタでラッチされた乱数値とが合成されたものが基になる。これらの乱数値が生成される乱数生成回路は、外部クロック信号に基づいて更新される。また、生成される乱数値の最大値設定は無しに設定されている。

30

【2 2 2 6】

特図 1 当選乱数と特図 2 当選乱数とは、同じ乱数生成チャンネルを使用し、ハードラッチするハードラッチ乱数値レジスタを特図 1 当選乱数と特図 2 当選乱数とで異ならせてもよい。またソフトラッチする乱数生成チャンネルを特図 1 当選乱数と特図 2 当選乱数とで異ならせてもよい。

【2 2 2 7】

表示図柄乱数は、1 6 ビット乱数回路 C H 3 で生成されハードラッチ乱数値レジスタ 0 でラッチされた乱数値が基になる。当該乱数値が生成される乱数生成回路は、外部クロック信号に基づいて更新される。また、生成される乱数値の最大値設定は有り（例えば、0 ～ 1 9 9 ）に設定されている。

40

【2 2 2 8】

普図当選乱数は、8 ビット乱数回路 C H 0 で生成されハードラッチ乱数値レジスタ 0 でラッチされた乱数値が基になる。当該乱数値が生成される乱数生成回路は、外部クロック信号に基づいて更新される。また、生成される乱数値の最大値設定は無しに設定されている。

【2 2 2 9】

大当り用特図乱数、小当り用特図乱数、およびはずれ用特図乱数は、上述の主制御部タ

50

イマ割込み処理において生成される。すなわちこれらの乱数は、所謂ソフトウェア乱数である。なお、これらの乱数を生成する際に用いられる初期値生成用乱数は、主制御部メイン処理および主制御部タイマ割込み処理で生成される。なお、大当り用特図乱数、小当り用特図乱数、およびはずれ用特図乱数は、ハードラッチされた乱数を用いてもよい。

#### 【2230】

特図変動時間決定用乱数、および普図変動時間決定用乱数は、カウンタ回路312の値を乱数として用いる。本実施形態のカウント回路312は、マイクロプロセッサ3000のシステムクロックの他、上記タイマ回路からのタイムアウト信号、メモリの読み書き信号、メモリリクエスト信号、外部入出力の信号等もカウンタの対象として用いることができる。このため、これらを組み合わせることで規則性のない値を導出させ、上記の乱数に使用している。演出用乱数は、主制御部メイン処理および主制御部タイマ割込み処理で生成される。

10

#### 【2231】

次に、図277を用いて本実施形態の遊技台（スロットマシン）で使用される乱数と、その主な導出源について説明する。同図は、本実施形態の遊技台で使用される乱数の導出源を表で示す図である。

#### 【2232】

役抽選乱数は、16ビット乱数回路CH0で生成されソフトラッチ乱数値レジスタでラッチされた乱数値が基になる。当該乱数値が生成される乱数生成回路は、内部クロック信号に基づいて更新される。また、生成される乱数値の最大値設定は無しに設定されている。

20

#### 【2233】

演出抽選乱数は、16ビット乱数回路CH1で生成されソフトラッチ乱数値レジスタでラッチされた乱数値が基になる。当該乱数値が生成される乱数生成回路は、内部クロック信号に基づいて更新される。また、生成される乱数値の最大値設定は無しに設定されている。

#### 【2234】

左リール遅延時間抽選乱数は、8ビット乱数回路CH0で生成されソフトラッチ乱数値レジスタでラッチされた乱数値が基になる。当該乱数値が生成される乱数生成回路は、内部クロック信号に基づいて更新される。また、生成される乱数値の最大値設定は有りに設定されている。

30

#### 【2235】

中リール遅延時間抽選乱数は、8ビット乱数回路CH1で生成されソフトラッチ乱数値レジスタでラッチされた乱数値が基になる。当該乱数値が生成される乱数生成回路は、内部クロック信号に基づいて更新される。また、生成される乱数値の最大値設定は有りに設定されている。

#### 【2236】

右リール遅延時間抽選乱数は、8ビット乱数回路CH2で生成されソフトラッチ乱数値レジスタでラッチされた乱数値が基になる。当該乱数値が生成される乱数生成回路は、内部クロック信号に基づいて更新される。また、生成される乱数値の最大値設定は有りに設定されている。

40

#### 【2237】

スロットマシンでは、外部クロック信号に基づいて乱数生成回路を更新しないため、最大値設定前に取り込まれた乱数のリード処理を行わないようにしてもよい。

#### 【2238】

次に、以上説明した実施形態7によるパチンコ機100の特徴的構成について再度図270乃至図277を参照しつつ説明する。

(1) 本実施の形態によるパチンコ機100は、

乱数を少なくとも導出可能な乱数回路（例えば、16ビット乱数回路9000、8ビット乱数回路10000）と、

50

複数の遊技制御処理のそれぞれを、メイン制御（例えば、主制御部メイン処理）および割込み周期ごとに行われる割込み制御（例えば、主制御部タイマ割込み処理）のうちの、少なくともいずれかの制御において実行可能な遊技制御手段（例えば、主制御部 3 0 0）と、

を備えた遊技台であって、

前記乱数回路は、第一の数値範囲内で更新される第一の乱数（例えば、最大値設定で起動される乱数生成回路が生成する乱数値）を導出可能なものであり、

前記乱数回路は、第二の数値範囲内で更新される第二の乱数（例えば、ユーザモード起動で起動される乱数生成回路が生成する乱数値）を導出可能なものであり、

前記第一の乱数の起動時期と前記第二の乱数の起動時期（例えば、乱数生成チャンネル C H 0 ～ C H 3 の乱数生成回路の起動時期）は異なるものであること

10

を特徴とする。

#### 【 2 2 3 9 】

当該構成を備えたパチンコ機 1 0 0 によれば、第一の乱数と第二の乱数とを生成する乱数生成回路の起動時期が異なるため、不正者による 2 つの乱数生成回路の起動時期の解析を困難にすることができ、安定した遊技制御を行うことができる場合がある。

#### 【 2 2 4 0 】

( 2 ) 上記パチンコ機 1 0 0 であって、

第一の処理（例えば、主制御部メイン処理のステップ S 1 5 1 0 3 ）を前記遊技制御処理は少なくとも含むものであり、

20

前記第一の乱数の起動時期と前記第二の乱数の起動時期のうちの一方は、前記第一の処理よりも後であること

を特徴とする。

#### 【 2 2 4 1 】

当該構成を備えたパチンコ機 1 0 0 によれば、第一の乱数または第二の乱数を生成する乱数生成回路の起動時期が第一の処理後であるため、不正者による不正をより困難することができ、安定した遊技制御を行うことができる場合がある。

#### 【 2 2 4 2 】

( 3 ) 上記パチンコ機 1 0 0 であって、

前記第一の処理は、制御の進行を保留可能なものであり、

30

前記第一の処理は、制御進行条件が成立した場合に（例えば、主制御部メイン処理のステップ S 1 5 1 0 3 の Y e s ）、前記保留を解除可能なものであり、

前記制御進行条件は、第一の信号が第一の状態から第二の状態に変化した場合に成立するものであること

を特徴とする。

#### 【 2 2 4 3 】

当該構成を備えたパチンコ機 1 0 0 によれば、第一の信号の状態の変化時期に応じて乱数生成回路の起動時期が変化するため、不正者による不正をより困難にでき、安定した遊技制御を行うことができる場合がある。

#### 【 2 2 4 4 】

40

( 4 ) 上記パチンコ機 1 0 0 であって、

前記乱数回路に前記数値範囲の設定を指示可能な第二の処理（例えば、主制御部メイン処理のステップ S 1 5 1 3 3 ）と、

前記乱数回路によって導出された乱数を取得可能な第三の処理（例えば、主制御部タイマ割込処理のステップ S 1 5 2 1 7 ）と、

前記第三の処理によって取得された乱数に基づく判定を実行可能な第四の処理（例えば、主制御部タイマ割込処理のステップ S 1 5 2 2 3 、ステップ S 1 5 2 2 9 、ステップ S 1 5 2 3 1 ）と、を前記遊技制御処理は少なくとも含むものであり、

前記第一の数値範囲は、前記第二の処理によって設定されるものであり、

前記第二の数値範囲は、前記第二の処理によらずに設定されるものであり、

50

前記第一の数値範囲が設定された場合に、前記第一の乱数の起動が開始可能なものであること

を特徴とする。

【2245】

当該構成を備えたパチンコ機100によれば、数値範囲設定後に乱数生成回路を起動（更新を開始）するので数値範囲外の数値を更新することがなく、安定した遊技制御を行うことができる場合がある。

【2246】

第一の処理は、1又は複数の命令であってもよい。

第一の処理は、分岐命令であってもよい。

第一の乱数の更新周期と第二の乱数の更新周期が同じであってもよい。

第一の乱数と第二の乱数が同一の乱数回路（チャンネル）で生成されるものであってもよい。

第一の乱数の起動時期が、第二の乱数の起動時期よりも後であってもよい。

第一の乱数の起動時期は、第二の乱数が更新を複数回行った後であってもよい。

第一の数値範囲は、第二の数値範囲よりも小さいものであってもよい。

第二の乱数の起動時期は、プログラムの開始時期と同時またはプログラムの開始時期よりも前であってもよい。

第一の乱数と第二の乱数のうち、一方は遊技者の利益に関わるものであってもよく、他方は遊技者の利益に関わらないものであってもよい。

第一の乱数と第二の乱数は遊技者の利益に関わるものであってもよく、第一の乱数と第二の乱数のうちの一方は直接的に遊技者の利益に関わるものであってもよく、他方は間接的に遊技者の利益に関わるものであってもよい。

第一の乱数と第二の乱数は、遊技者の利益に関わるものであってもよい。

第一の乱数と第二の乱数は、遊技者の利益に関わらないものであってもよい。

第一の乱数の起動時期は、スタックポインタの設定の後、直後、前、または直前の何れかであってもよい。

第一の乱数の起動時期は、ジャンプ処理の後、直後、前、または直前の何れかであってもよい。

第一の乱数の起動時期は、第一の処理の後、直後、前、または直前の何れかであってもよい。

第一の乱数の起動時期は、カウンタ回路設定処理の後、直後、前、または直前の何れかであってもよい。

第一の乱数の起動時期は、クリア信号出力処理の後、直後、前、または直前の何れかであってもよい。

前記乱数回路は、第三の数値範囲内で更新される第三の乱数を導出可能なものであり、第三の数値範囲は、前記第二の処理によって設定されるものであり、第一の数値範囲と第三の数値範囲の設定値を少なくとも記憶可能な記憶手段を備え、第一の数値範囲と第三の数値範囲の設定は、記憶手段に記憶された設定値が格納されるアドレスの小さいものから設定されるものであってもよい。

第二の乱数は遊技者の利益に関わるものであってもよく、第一の乱数は間接的に遊技者の利益に関わるものであってもよく、第三の乱数は遊技者の利益に関わらないものであってもよい。

第三の乱数の起動時期は、第一の乱数の起動時期の後、直後、前、または直前の何れかであってもよい。

乱数回路は、複数の乱数回路を含み、複数の乱数回路はそれぞれ、ハードラッチ乱数値レジスタとソフトラッチ乱数値レジスタを有してもよい。

複数の乱数回路は、16ビット乱数回路と8ビット乱数回路を含んでもよい。

16ビット乱数回路は、ハードラッチ乱数値レジスタを複数有してもよい。

8ビット乱数回路は、ハードラッチ乱数値レジスタを一つのみ有してもよい。

10

20

30

40

50

１６ビット乱数回路は、第一の数のハードラッチ乱数値レジスタを有する第一の１６ビット乱数回路と、第一の数よりも小さい第二の数のハードラッチ乱数値レジスタを有する第二の１６ビット乱数回路を含んでもよい。

１６ビット乱数回路は、ソフトラッチ乱数値レジスタを一つのみ有してもよい。

８ビット乱数回路は、ソフトラッチ乱数値レジスタを一つのみ有してもよい。

１６ビット乱数回路の数値範囲と８ビット乱数回路の数値範囲を設定する場合、１６ビット乱数回路の数値範囲を８ビット乱数回路の数値範囲よりも先に設定してもよい。

１６ビット乱数回路の数値範囲と８ビット乱数回路の数値範囲を設定する場合、８ビット乱数回路の数値範囲を１６ビット乱数回路の数値範囲よりも先に設定してもよい。

第一の乱数の起動時期は、払出制御手段が初期設定処理を行っている間であってもよい。

第一の乱数の起動時期は、払出制御手段が主制御手段からの信号を待っている間であってもよい。

第一の乱数の起動時期は、副制御手段が起動処理を行っている間であってもよい。

第一の乱数の起動時期は、画像表示制御手段が画像表示手段に表示する表示データを展開している間であってもよい。

乱数回路は、複数のチャネル（第一のチャネル、第一のチャネルを含む）を備え、チャネル毎に乱数の更新が可能であり、チャネル毎に乱数値を発生させることが可能であり、チャネル毎に起動することが可能であり、チャネル毎に発生させる数値範囲の設定が可能である。

「第一の数値範囲の第一の乱数値」は、「第一の最小値以上であり、かつ第一の最大値以下である第一の乱数値」であってもよい。

「第二の数値範囲の第二の乱数値」は、「第二の最小値以上であり、かつ第二の最大値以下である第二の乱数値」であってもよい。

#### 【２２４７】

##### <実施形態８>

次に、上述の実施形態５の主制御部５３００の基本回路５３０２が搭載されるパッケージのさらに他の変形例としての実施形態８について詳細に説明する。図２７８（ａ）は、基本回路５３０２が搭載されるパッケージ１０１００ａおよびパッケージ１０１００ｂの平面図である。パッケージ１０１００ａは、ほぼ直方体形状を有し、短手方向両側には、パッケージ１０１００ａの長手方向に沿って複数の端子が配置されている。パッケージ１０１００ｂはパッケージ１０１００ａと同一形状を備えている。パッケージ１０１００ａおよびパッケージ１０１００ｂの内部にそれぞれ設けられているＩＣは異なる機能を奏するように用いられ、組み合わせることにより基本回路５３０２を構成できるようになっている。パッケージ１０１００ａの平面視中央には四角柱状の凸部１０１０１ａが形成され、パッケージ１０１００ｂの平面視中央には四角柱状の凸部１０１０１ｂが形成されている。

#### 【２２４８】

同図（ｂ）は、パッケージ１０１００ａおよびパッケージ１０１００ｂを接続部材１０１００ｃを用いて接続した形状を示す平面図である。同図（ｃ）は、同側面図である。また、同図（ｄ）は、同斜視図である。接続部材１０１００ｃは平板棒状の細長い直方体形状を有している。接続部材１０１００ｃの薄板部両端側には四角柱状の突起１０１０１ａまたは突起１０１０１ｂを嵌め込むことができる凹部１０１０１ｃがそれぞれ設けられている。接続部材１０１００ｃの短手方向の長さは、パッケージ１０１００ａおよびパッケージ１０１００ｂの短手方向の長さより短い。

#### 【２２４９】

同図（ｂ）、（ｃ）および（ｄ）は、接続部材１０１００ｃの２つの凹部１０１０１ｃにパッケージ１０１００ａおよびパッケージ１０１００ｂの凸部１０１０１ａおよび凸部１０１０１ｂがそれぞれ嵌め込まれて、パッケージ１０１００ａおよびパッケージ１０１００ｂの複数の端子の配設方向を長手方向に揃えて一体化した状態を示している。このよ

うに2つのパッケージ10100aおよびパッケージ10100bを接続部材10100cで接続して一体化することにより、基板上で近接させて2つのパッケージ10100aおよびパッケージ10100bを容易に配置することができる場合がある。

#### 【2250】

図279(a)は、基本回路5302が搭載されるパッケージ10120aおよびパッケージ10120bを接続配線部10120cで接続した形状を示す平面図である。同図(b)は、同側面図である。同図(c)は、同斜視図である。パッケージ10120aは、ほぼ直方体形状を有し、短手方向両側には、パッケージ10120aの長手方向に沿って複数の端子が配置されている。パッケージ10120bはパッケージ10120aと同一形状を備えている。パッケージ10120aおよびパッケージ10120bの内部にそれぞれ設けられているICは異なる機能を奏するように用いられ、組み合わせることにより基本回路5302を構成できるようになっている。

10

#### 【2251】

パッケージ10120b内の回路は、複数の配線が内蔵された不透明の接続配線部10120cを介してパッケージ10120a内の回路と電氣的に接続できるようになっている。接続配線部10120cは可撓性を備えた配線基板を有し、配線基板上で並列する複数の配線を有している。接続配線部10120cの両配線端部がパッケージ10120aおよびパッケージ10120bの短手側の側面部中央に設けられた端子部にそれぞれ接続されている。接続配線部10120cの複数の配線の延伸方向にはほぼ直交し、複数の配線が連設されている方向の長さは、パッケージ10120aおよびパッケージ10120bの短手方向の長さより短い。上述の図278に示すパッケージ10100aおよびパッケージ10100bは互いに電氣的に直接的に接続される構成になっていないが、図279に示すパッケージ10120aおよびパッケージ10120bは接続配線部10120cを介して互いに電氣的に直接的に接続される構成になっている。パッケージ10120aおよびパッケージ10120bを組み合わせてマイコンとして用いることも可能であり、入出力端子が余らないように組み合わせることによりセキュリティ性を向上させることができる場合がある。

20

#### 【2252】

図280は、図278(b)、(c)、(d)に示すパッケージ10100aおよびパッケージ10100bを接続部材10100cで一体化して収容ケース10103で覆った状態を示す図である。図280(a)は平面図、(b)は側面図、(c)は斜視図である。収容ケース10103は上ケース10103aと下ケース10103bとに分離可能である。収容ケース10103は例えば不透明な絶縁性樹脂で成型されている。上ケース10103aと下ケース10103bとを組み合わせた状態で全体として収容ケース10103は薄い直方体形状を呈している。上ケース10103aと下ケース10103bとでパッケージ10100aおよびパッケージ10100b、接続部材10100cを挟み込むことにより、パッケージ10100aおよびパッケージ10100bの複数の端子だけが突出してその他の部材が収容ケース10103内に収容された一体化パッケージ10100を得ることができる。このように収容ケース10103でパッケージ10100aおよびパッケージ10100b、接続部材10100cを覆うことにより、パッケージ10100aおよびパッケージ10100b、接続部材10100cが分離してしまうことを未然に防ぐことができる場合がある。

30

40

#### 【2253】

図281は、図279(a)、(b)、(c)に示すパッケージ10120aおよびパッケージ10120bを接続配線部10120cで一体化して収容ケース10104で覆った状態を示す図である。図281(a)は平面図、(b)は側面図、(c)は斜視図である。収容ケース10104は上ケース10104aと下ケース10104bとに分離可能である。収容ケース10104は例えば不透明な絶縁性樹脂で形成されている。上ケース10104aと下ケース10104bとを組み合わせた状態で全体として収容ケース10103は3つの薄い直方体形状をつなぎ合わせた形状を呈している。具体的には、収容

50



ケース１０１０４は、パッケージ１０１２０aの外形に倣うように形成された箱状の第一領域Aと、パッケージ１０１２０bの外形に倣うように形成された箱状の第三領域Cと、第一領域Aと第三領域Cを接続して接続配線部１０１２０cを収容する第二領域Bとで構成されている。上ケース１０１０４aと下ケース１０１０４bとでパッケージ１０１２０a、１０１２０bおよび接続配線部１０１２０cを挟み込むことにより、パッケージ１０１２０aおよびパッケージ１０１２０bの複数の端子だけがケース１０１０４の第一側面および第一側面に対向する第二側面から突出してその他の部材が収容ケース１０１０４内に収容された一体化パッケージ１０１２０を得ることができる。収容ケース１０１０４の短手方向の長さ（第一側面と第二側面との距離）に関し、第一領域Aと第三領域Cはほぼ同じ長さであり、第二領域Bは第一領域Aより短い長さを有している。このように収容ケース１０１０４でパッケージ１０１２０a、１０１２０bおよび接続配線部１０１２０cを覆うことにより、パッケージ１０１２０aおよびパッケージ１０１２０b、接続配線部１０１２０cが分離してしまうことを未然に防ぐことができる場合がある。

10

#### 【２２５４】

また、一体化パッケージ１０１２０は２つのパッケージ１０１２０aとパッケージ１０１２０bとを接続配線部１０１２０cで電氣的に接続しているため、接続配線部１０１２０cを介して不正な電子部品を接続する不正行為が行われてしまうことが想定される。不正な電子部品は発見を困難にするため一体化パッケージ１０１２０の裏面側に装着される可能性が高い。このため、図２８０に示すような全体で直方体形状の収容ケース１０１０３よりも、図２８１に示すような中央部の幅が狭い収容ケース１０１０４の形状の方が一体化パッケージ１０１２０の裏面側を観察し易いので好適である。第二領域Bの短手方向の幅が第一及び第三領域A、Cの短手方向の幅より狭いことにより収容ケース１０１０４中央部にできた凹部を利用して観察することにより、一体化パッケージ１０１２０の裏側や、一体化パッケージ１０１２０と基板（またはICソケット）の隙間などに不正な回路を後付けするような不正行為を容易に発見することができ、不正行為を未然に防止できる場合がある。

20

#### 【２２５５】

次に、以上説明した実施形態８によるパチンコ機１００の特徴的構成について説明する。

。

（１）本実施の形態によるパチンコ機１００は、

30

遊技制御を行うマイクロコンピュータを備えた遊技台であって、

前記遊技台は、ぱちんこ機またはスロットマシンであり、

前記マイクロコンピュータは、DIP型のパッケージ（例えば、一体化パッケージ１０１２０）に収容され、

前記パッケージは、第一領域（例えば、第一領域A）、第二領域（例えば、第二領域B）、および第三領域（例えば、第三領域C）がこの順に一体的に設けられており、

前記第一領域および前記第三領域の第一側面、および該第一側面に対向する第二側面の各々に複数の端子が設けられており、

前記第二領域の前記第一側面および前記第二側面には端子が設けられておらず、

前記第二領域の前記第一側面および前記第二側面間の幅は、前記第一領域の前記第一側面および前記第二側面間の幅より短いこと

40

を特徴とする。

#### 【２２５６】

当該構成を備えたパチンコ機１００によれば、不正な回路を後付けするような不正行為を容易に発見することができ、不正行為を未然に防止できる場合がある。

#### 【２２５７】

<実施形態９>

以下、図面を用いて、本発明の実施形態９に係るパチンコ機（遊技台）について詳細に説明する。

#### 【２２５８】

50

図 2 8 2 は、特図始動口ポートの状態と乱数取得との関係を示している。同図 (a)、(b) は、I/O 3 1 0 の特図始動口ポートに出力される特図始動口（ここでは例えば特図 2 始動口 2 3 2 とする）の球検出センサの検出信号を示している。横軸は時間の経過を示し、縦軸は信号レベルを示している。また、図中 (1) ~ (4) に示す時点は、主制御部 3 0 0 でのタイマ割込み処理の開始タイミングを示している。本例では、タイマ割込みの周期は 4 m s であるものとする。また、図中破線の枠で示す領域は、乱数がラッチまたは取得される時期 A、B、C を示している。時期 A は、例えば 1 6 ビット乱数回路 9 0 0 0 で生成された乱数が所定のソフトラッチ乱数値レジスタ（例えば、C H 0 ソフトラッチ乱数値レジスタ 9 0 4 0）にラッチされてタイマ割込み処理で乱数値（以下、ハード乱数（ソフトラッチ）と称する）が取得される時期を示している。時期 B は、ユーザプログラムで生成しているソフトウェア乱数（以下、ソフト乱数と称する）を取得する時期を示している。時期 C は、1 6 ビット乱数回路 9 0 0 0 の所定のハードラッチ乱数値レジスタ（例えば、C H 0 ハードラッチ乱数値レジスタ (0) 9 0 3 2 a）に乱数値（以下、ハード乱数（ハードラッチ）と称する）がラッチされる時期を示している。これらの時期 A、B、C はそれぞれラッチや取得の時期が変動したり不定であったりするので時間的に幅を持って示している。

#### 【2 2 5 9】

図 2 8 2 (a) に示す例では、時点 (1) での特図始動口ポートの状態はオフ、続く時点 (2) での特図始動口ポートの状態はオン、続く時点 (3) での特図始動口ポートの状態もオンであるため（以下、オフーオンーオンのように略称する場合がある）、入力ポート状態更新処理（ステップ S 2 0 5）において、過去 3 回分の検出信号の有無の情報が入賞判定パターン情報と一致すると判定される。これにより、時点 (3) で開始されたタイマ割込み処理の入賞受付処理（ステップ S 2 1 7）において、特図 2 始動口 2 3 2 への入賞があったと判定される。主制御部 3 0 0 は、時点 (3) を含む時期 A で 1 6 ビット乱数回路 9 0 0 0 の所定のソフトラッチ乱数値レジスタからハード乱数（ソフトラッチ）を取得するとともに、時点 (3) を含む時期 B でソフト乱数を取得する。次いで、ハード乱数（ソフトラッチ）にソフト乱数を加算する等の乱数の加工処理を実行する。加工された乱数値は当否判定用乱数として特図 2 関連抽選処理（ステップ S 2 2 9）で使用される。

#### 【2 2 6 0】

図 2 8 2 (a) の右方に示すテーブルは乱数取得タイミングをまとめたものであり、左から順に、時期、取得タイミング、乱数の種類を示している。図 2 8 2 (a) に示す例では、取得タイミングが時点 (3) を含む時期 A でハード乱数（ソフトラッチ）が取得され、取得タイミングが時点 (3) を含む時期 B でソフト乱数が取得される。なお、厳密には、入賞受付処理（ステップ S 2 1 7）は時点 (3) から所定期間後になる。また、入賞受付処理の前に種々の判定処理があるのでハード乱数（ソフトラッチ）のラッチ時期やソフト乱数の取得時期をランダムに変化させることができる場合がある。また、乱数取得は時期 (4) で実行してもよい。この場合は、入賞判定パターン情報として、オンーオンーオフ（あるいは、オンーオンーオンーオフ）の情報をうい、時期 (4) で開始されたタイマ割込み処理の入力ポート状態更新処理（ステップ S 2 0 5）で、過去 3 回分の検出信号の有無の情報が入賞判定パターン情報と一致すると判定される。また、時点 (4) で開始されたタイマ割込み処理の入賞受付処理（ステップ S 2 1 7）で特図 2 始動口 2 3 2 への入賞があったと判定されて乱数取得が実行される。こうすることにより、乱数取得の時期をさらに変動させることができる場合がある。また、入賞判定から乱数取得までの期間を乱数列が一巡する周期よりも確実に長くすることができる。なお、過去 2 回分の検出信号の有無の情報として例えばオフーオンを入賞判定パターン情報とする場合には、時点 (1) での特図始動口ポートの状態はオフ、続く時点 (2) での特図始動口ポートの状態がオンなので時期 (2) で開始されたタイマ割込み処理で乱数取得を実行することができる。

#### 【2 2 6 1】

次に、図 2 8 2 (b) に示す例について説明する。例えば、図 2 7 0 に示す乱数回路 9 0 0 0 は、外部端子入力による乱数取込みができるようになっている。1 6 ビット乱数回

10

20

30

40

50

路 9 0 0 0 では予め入力ポート制御レジスタで設定した期間以上のアクティブ信号が入力されると所定のハードラッチ乱数値レジスタに乱数値がラッチされるようになっている。このため、特図始動口ポートの状態を示す信号を乱数回路 9 0 0 0 の外部端子入力に入力することにより、ハード乱数（ハードラッチ）を取得することができる。

#### 【2 2 6 2】

図 2 8 2 (b) に示すように時点 (1) と時点 (2) の間で特図始動口ポートの状態がオフからオンに変化すると、時期 C でハード乱数（ハードラッチ）がラッチされる。より具体的には、特図始動口ポートの状態がオフからオンに変化する波形の立ち上がりエッジから数  $\mu s$  後にハードラッチが実行される。つまり、球検出センサを通過してから所定期間後にハードラッチが行われる。

#### 【2 2 6 3】

次いで、時点 (1) での特図始動口ポートの状態はオフ、続く時点 (2) での特図始動口ポートの状態はオン、続く時点 (3) での特図始動口ポートの状態はオンであるため、時点 (3) での入力ポート状態更新処理（ステップ S 2 0 5）において、過去 3 回分の検出信号の有無の情報が入賞判定パターン情報と一致すると判定される。これにより、時点 (3) で開始されたタイマ割込み処理の入賞受付処理（ステップ S 2 1 7）において、特図 2 始動口 2 3 2 への入賞があったと判定される。主制御部 3 0 0 は、時点 (3) を含む時期 A で 1 6 ビット乱数回路 9 0 0 0 の所定のソフトラッチ乱数値レジスタからハード乱数（ソフトラッチ）を取得するとともに、予め時期 C でラッチされていたハード乱数（ハードラッチ）を取得する。次いで、ハード乱数（ソフトラッチ）にハード乱数（ハードラッチ）を加算する等の乱数の加工処理を実行する。加工された乱数値は当否判定用乱数として特図 2 関連抽選処理（ステップ S 2 2 9）で使用される。

#### 【2 2 6 4】

図 2 8 2 (b) の右方に示すテーブルは乱数取得タイミングをまとめたものであり、左から順に、時期、取得タイミング、乱数の種類を示している。図 2 8 2 (b) に示す例では、取得タイミングが時点 (3) を含む時期 A でハード乱数（ソフトラッチ）および時期 C でラッチされたハード乱数（ハードラッチ）が取得される。なお、厳密には、入賞受付処理（ステップ S 2 1 7）は時点 (3) から所定期間後になる。また、ハードラッチは入賞のタイミングに依存しており、主制御部 3 0 0 のタイマ割込み処理とは無関係に発生する。つまりハード乱数（ハードラッチ）をラッチするタイミングは偶発的なものなので、ユーザプログラムの開始位置からラッチの時期を特定することはできない場合がある。また、入賞受付処理の前に種々の判定処理があるのでハード乱数（ソフトラッチ）のラッチ時期をランダムに変化させることができる場合がある。また、上述のように乱数取得を時期 (4) で開始されるタイマ割込み処理で実行してもよい。こうすることにより、乱数取得の時期をさらに変動させることができる場合がある。また、入賞判定から乱数取得までの期間を乱数列が一巡する周期よりも確実に長くすることができる。なお、上述のように入賞判定パターン情報を変更して、時点 (1) での特図始動口ポートの状態はオフ、続く時点 (2) での特図始動口ポートの状態はオンで入賞と判定して時期 (2) で開始されるタイマ割込み処理で乱数取得を実行することもできる。なお、過去複数回分の検出信号の有無の情報が入賞判定パターン情報と一致しないと判定された場合は、ハードラッチに係るハード乱数は取込まない。

#### 【2 2 6 5】

図 2 8 3 は、図 2 8 2 (b) に示すものと同様だが、ハード乱数のハードラッチのタイミングがずれた場合について示している。ノイズフィルタ 3 1 8 5 の設定や、特図 2 始動口 2 3 2 への入球のタイミングによっては、ハードラッチのタイミングは特図始動口ポートの波形の立ち上がりエッジおよびタイマ割込み処理の開始時点 (2) より後の時刻  $t$  になる場合がある。つまり、タイマ割込みタイミングが特図始動口ポートの波形の立ち上がりエッジとハードラッチのタイミング（時刻  $t$ ）とで挟まれている。

#### 【2 2 6 6】

また、図 2 8 3 は、時点 (4) で開始されるタイマ割込み処理でハード乱数をソフトラ

10

20

30

40

50

ッチする場合を示している。16ビット乱数値（0～65535）が一巡する周期は、例えば内部システムクロック（例えば、10MHz）を用いた場合、約6.6msとなる。一方、時点（2）近傍の時刻tでハード乱数（ハードラッチ）がラッチされ、それよりほぼ2割込み分の時間経過後の時点（4）以降でハード乱数（ソフトラッチ）がラッチされるので、ハードラッチとソフトラッチの間隔を最短でも2割込み分（約8ms）程度あけることができる。このため、ハードラッチによるハード乱数のラッチ後に確実に乱数列を一巡させた後にソフトラッチによるハード乱数をラッチすることができる。つまり、乱数取得の時期のランダム性を増大させることができる。時点（4）で開始されるタイマ割込み処理で入賞判定処理を可能にするには、入賞判定パターン情報を前々回がオン、前回がオン、今回がオフとすればよい。こうすることにより、時点（2）での特図始動口ポートの状態はオン、続く時点（3）での特図始動口ポートの状態はオン、続く時点（4）での特図始動口ポートの状態はオフであるため、時点（4）で開始されるタイマ割込み処理の入力ポート状態更新処理（ステップS205）において、過去3回分の検出信号の有無の情報が入賞判定パターン情報と一致すると判定される。16ビット乱数回路9000に最大値設定をした場合には、確実に乱数列を一巡以上させることができるようになる場合がある。また、確実に乱数列を一巡させたい場合や、2割込みでソフトラッチをしたい場合等では、最大値を低く設定することにより実現できる。

#### 【2267】

図282および図283に示した乱数取込みにおけるソフト乱数の取込みは、ソフトラッチに代えて実施形態6で図246等を用いて説明した第三の命令を用いた乱数値直接取込みを用いてもよい。また、上記実施形態では、ハードラッチの設定において、ラッチした乱数値をリードしなくても次のラッチを受け付けるようにしているが、ラッチした乱数値をリードしないと次のラッチを受け付けない設定にしてももちろんよい。この場合には、過去3回分の検出信号の有無の情報が入賞判定パターン情報と一致しないと判定された場合であってもハードラッチに係るハード乱数を一旦取込み、次いで当該乱数を破棄するようにすればよい。

#### 【2268】

また、上記実施形態では、ハードラッチとソフトラッチの組合せによる乱数値を用いるものとして当否判定用乱数を例に挙げたが、例えば、図柄決定用乱数にはハードラッチとソフトラッチの組合せによる乱数値は使わないようにしてもよい。

また、上記実施形態では、16ビット乱数回路9000に最大値設定をしない場合で説明したが、最大値設定をしてもよい。また、8ビット乱数回路10000に対し上記と同様にして8ビット乱数を取得するようにしてもよい。さらに、16ビット乱数回路9000と8ビット乱数回路10000を組合わせて上記と同様にして乱数を取得するようにしてもよい。またさらに、ハードラッチとソフトラッチの組合せにさらにソフト乱数を組合わせたり、他の数値を組合わせたりしてももちろんよい。また、ハードラッチと別のハードラッチとを組合せてもよい。また、上記実施形態では、図270に示す16ビット乱数回路9000を用いた乱数値取得を例に挙げたが、図284に示すカウンタ回路312を用いた乱数値取得でハードラッチやソフトラッチを行わせるようにしてもよい。

#### 【2269】

また、遊技球が球検出センサを通過するタイミングとハードラッチのタイミングが、別々の割込み周期になる場合がある。例えば上述のように1回目の割込みによる4ms間に遊技球が球検出センサを通過し、次の割込みによる4ms間にハードラッチが行われる場合がある。なお、ハードラッチによりハードラッチ乱数値レジスタに乱数値がラッチされるとラッチフラグレジスタに「1」がセットされる。このとき、すでに開始されている割込み処理での例えば特図関連抽選処理等でハード乱数（ハードラッチ）をCPUコアレジスタやRAMへ転送する場合があるが、次の割込み処理での特図関連抽選処理等でハード乱数（ハードラッチ）を転送する場合ももちろんある。ハード乱数（ハードラッチ）が取得されるとラッチフラグレジスタに「0」がセットされる。ハード乱数（ソフトラッチ）は、割込み処理での入力ポート状態更新処理等で実行され、ソフトラッチ選択レジスタに

10

20

30

40

50

「1」がセットされると、所定のソフトラッチ乱数値レジスタに乱数値がラッチされるとともにソフトラッチフラグレジスタに「1」がセットされる。次いで、特図関連抽選処理等で、ハード乱数値（ソフトラッチ）がCPUコアレジスタやRAMの所定領域に書き込まれる。また、ソフトラッチフラグレジスタが「0」にセットされる（クリアされる）。

なお、特図関連抽選処理において、取得した乱数値に基づいて特図停止図柄を決定し、例えば、大当たり遊技のラウンド数を決定したり、大当たり遊技後の遊技状態を確変状態や通常状態にするかを決定しているが、それらの決定において別々の乱数を用いてもよい。また、種々の組合せが可能であり、例えば、特図停止図柄が決まると遊技状態だけは決まるようにしてもよいし、特図停止図柄が決まるとラウンド数だけは決まるようにしてもよい。

10

#### 【2270】

次に、以上説明した実施形態9によるパチンコ機100の特徴的構成について説明する。

(1) 本実施の形態によるパチンコ機100は、

所定の数値範囲内で更新される乱数を導出する乱数生成手段（例えば、16ビット乱数回路9000、8ビット乱数回路10000）と、

複数の遊技制御処理のそれぞれを、メイン制御（例えば、主制御部メイン処理）および割込み周期ごとに行われる割込み制御（例えば、主制御部タイマ割り込み処理）のうちの、少なくともいずれかの制御において実行可能な遊技制御手段（例えば、主制御部300）と

20

を備えた遊技台であって、

前記乱数生成手段は、

外部端子入力に入力された信号に基づいてラッチされた第一の乱数（例えば、ハード乱数（ハードラッチ））を生成可能であり、前記遊技制御手段の制御に基づいて第二の乱数（例えば、ハード乱数（ソフトラッチ）や、乱数値直接取込みにより取得されるハード乱数）を生成可能であり、

前記遊技制御手段は、

前記第一の乱数がラッチされるタイミングと異なるタイミングで前記第二のハード乱数を取得可能であり、

取得した前記第一の乱数と前記第二の乱数とを組み合わせることで所定の乱数を生成することを特徴とする。

30

#### 【2271】

従来はユーザプログラムでチャタリングの除去を制御できるソフトラッチで取得した乱数を使っていた。ラッチしたハード乱数を無駄にしないため、所定の入賞判定パターン情報を利用してハード乱数のラッチが行えるのが望ましい。しかしながら、回路内部にラッチ信号が出力されるので、このラッチ信号を手がかりに不正者に乱数の解析や当否判定のタイミングが把握されてしまうおそれがある。そこで、ハード乱数のラッチ後に、ユーザプログラムで更新するソフト乱数を組み合わせることで使うことにより、少なくとも乱数の解析を困難にしていた。しかしながら、ソフト乱数を取得するタイミングはハード乱数のラッチのタイミングに近いのと、ソフト乱数の乱数更新周期が長いので解読されやすいという課題を有している。ハード乱数のラッチ時期とソフト乱数の取得時期とが近いのが問題であれば、ソフト乱数を取るタイミングを、例えば特図始動口ポートに入力される信号の立ち上がりエッジに同期させればよいとも考えられる。しかしながら、ソフト乱数の更新周期が長いので解読されやすいという課題は相変わらず残る。

40

#### 【2272】

そこで、本実施形態では、ソフト乱数を組み合わせるのではなく、ハードラッチのハード乱数を取得するようにして、別の方法で取得したハード乱数を組み合わせることで例えば当否判定用の乱数を生成するようにする。上記パチンコ機によれば、ハードラッチによるハード乱数とは異なるハード乱数を、ハードラッチのタイミングとは異なるタイミングで取得し、取得されたハード乱数（ハードラッチ）とハードラッチ以外で取得したハード乱数を

50

組み合わせるようにした。これにより、ソフト乱数ではタイマ割込み基準で処理の実行タイミングが把握されるおそれがあったのに対し、ハードラッチでは偶発的でラッチのタイミングが不定なので特定され難くすることができる場合がある。

(2) 上記パチンコ機100であって、

前記遊技制御手段を少なくとも有するマイクロプロセッサを備え、

前記マイクロプロセッサは、

少なくとも、セキュリティチェックを行うことが可能なものであり、該セキュリティチェックの後に前記遊技制御が行われるユーザモードへ移行するまでの時間をランダムに変動させるランダム延長処理が少なくとも行われる、セキュリティモードに滞在した後、該ユーザモードへ移行するものであることを特徴とする。

10

【2273】

当該構成を備えたパチンコ機100によれば、ユーザモードに移行する前にランダム延長処理を実行するので、遊技制御プログラムの実行開始タイミングをずらすことが可能なため、遊技制御処理の特定の処理（例えばソフト乱数の更新処理、初期値更新処理等）の実行時期を判別困難にでき、これらの特定の処理の狙い撃ちを防止することができる場合がある。

【2274】

(3) 上記パチンコ機100であって、前記ランダム延長処理の開始前に、前記マイクロプロセッサの所定の出力端子から所定の信号が出力されることを特徴とする。

(4) 上記パチンコ機100であって、前記所定の端子は、前記マイクロプロセッサ以外の外部機器に対してリセット信号を出力可能なリセット端子であることを特徴とする。

20

【2275】

当該構成を備えたパチンコ機100によれば、ランダム延長処理の前に外部機器に対してリセット信号を出力し、ランダム延長処理の後に遊技制御プログラムの実行を開始することができるので、遊技制御処理の特定の処理（例えばソフト乱数の更新処理、初期値更新処理等）の実行時期を判別困難にでき、これらの特定の処理の狙い撃ちを防止することができる場合がある。

【2276】

(5) 上記パチンコ機100であって、前記遊技制御手段から送出されるコマンドに基づいて演出制御を行う副制御手段を有することを特徴とする。

30

【2277】

＜実施形態7の変形例＞

次に、図284を用いて、実施形態7の変形例に係るパチンコ機（遊技台）に用いられるカウンタ回路312について説明する。図284は、カウンタ回路312を示すブロック図である。

【2278】

カウンタ回路312は、8ビットのプログラマブルカウンタを4チャンネル内蔵している。すなわち、図284に示すように、4つのカウンタ回路（カウンタ回路0～3）が用意されている。各カウンタ回路は、アップカウンタ回路であって、それぞれ独立した動作モードの設定が可能である。これら4つのカウンタ回路の構成はいずれも同じであるため、ここではカウンタ回路0を用いて説明する。

40

【2279】

カウンタ回路は、制御レジスタ3121、8ビットカウンタ3122、カウンタ設定レジスタ3123、およびカウンタレジスタ3124を有する。制御レジスタ3121は、内部バスを流れる各種の信号を取得可能であり、図284には、制御レジスタ3121が、8種類の信号を取得可能であることが示されている。これら8種類の信号は、システムクロック信号（SCLK）、図132に示すタイマ回路0～2それぞれからのタイムアウト信号、CPUコアのメモリリクエスト信号（XMREQ信号）、CPUコアのライト信号（XWR信号）、CPUコアのI/Oリクエスト信号（XIORD信号）、およびCPUコアの割込み要求応答信号（マスカブル割込みアクノリッジサイクル；XINTACK信

50

号)である。制御レジスタ3121は、8ビットカウンタ3122を動作させるためのクロック源を選択する8ビットのレジスタである。すなわち、8本の信号線から入力されるいずれの信号をクロック源にするかを選択するものである。また、制御レジスタ3121は、8ビットカウンタ3122の許可や禁止を設定するレジスタでもある。

#### 【2280】

また、カウンタ回路312は、更新源としてXWR信号、XMREQ信号、XIORQ信号、タイマ回路nのタイムアウト信号のうちの少なくとも2以上の更新源を設定可能であってもよい。換言すると、カウンタ回路は、複数の更新源のうちの第一の更新源を除く第二の更新源と第三の更新源を共に設定可能であってもよい。

#### 【2281】

制御レジスタ3121の詳細な構成については、図134に示す例と同じであるので説明を省略する。

#### 【2282】

また、カウンタ設定レジスタ3123は、アップカウンタ回路である8ビットカウンタ3122のカウント値(最大値)を設定する8ビットのレジスタである。本実施形態では、0~255のカウント値を設定することができ、「255」を書き込んだ場合には、カウント値として最大値の255が設定される。この場合には0~255が所定の数値範囲になる。なお、設定値に「00」を設定した場合には、8ビットカウンタの動作を停止するようにしてもよい。なお、各カウンタ回路は、ダウンカウンタ回路であってもよく、ダウンカウンタ回路である場合には、カウンタ設定レジスタ3123は、最小値を設定するレジスタになる。

#### 【2283】

カウンタレジスタ3124は、CPUレジスタリード信号の入力タイミングに合わせて8ビットカウンタ3122の値を読み出すためのレジスタである。

#### 【2284】

以上説明した、各レジスタの設定値は、内蔵ROM306のプログラムデータエリアに記憶されている。プログラム管理エリアに記憶されている各設定値は、図272に示す主制御部メイン処理における初期設定2(ステップS15105)において、プログラムデータエリアから呼び出され、各レジスタに設定されるか、あるいは主制御部メイン処理や図274に示す主制御部割込処理において呼び出され、各レジスタに設定される。カウンタ設定レジスタ3123にカウント値が書き込まれると、8ビットカウンタ3122の値が「00H」にクリアされ、その後、制御レジスタ3121の4ビット目に「1」が書き込まれると、選択されたクロック源で8ビットカウンタ3122のアップカウントが開始される。

#### 【2285】

カウンタ回路312には、カウンタ回路0、1用のハードラッチ選択レジスタとして、CH0、1ハードラッチ選択レジスタ9022、9024が設けられている。図284では、CH0、1ハードラッチ選択レジスタ9022、9024を1つにまとめて図示している。また、カウンタ回路2、3用のハードラッチ選択レジスタとして、CH2、3ハードラッチ選択レジスタ9026、9028が設けられている。図284では、CH2、3ハードラッチ選択レジスタ9026、9028を1つにまとめて図示している。これらのハードラッチ選択レジスタ9022、9024、9026、9028には、パラレル入力ポートを介してPI0~PI5信号がそれぞれ入力される。各ハードラッチ選択レジスタは、PI0~PI5信号の入力に基づいて、対応するハードラッチ乱数値レジスタにラッチ信号を出力する。

#### 【2286】

カウンタ回路0用のハードラッチ乱数値レジスタとして、CH0ハードラッチ乱数値レジスタ9032aが設けられている。また、カウンタ回路1用のハードラッチ乱数値レジスタとして、CH1ハードラッチ乱数値レジスタ9034aが設けられている。また、カウンタ回路2用のハードラッチ乱数値レジスタとして、CH2ハードラッチ乱数値レジスタ

10

20

30

40

50

タ 9 0 3 6 a が設けられている。また、カウンタ回路 3 用のハードラッチ乱数値レジスタとして、CH 3 ハードラッチ乱数値レジスタ 9 0 3 8 a が設けられている。

#### 【 2 2 8 7 】

これらのハードラッチ乱数値レジスタ 9 0 3 2 a、9 0 3 4 a、9 0 3 6 a、9 0 3 8 a には、3 つの信号が入力される。1 つ目の信号は、対応するカウンタ回路 0 ~ 3 の 8 ビットカウンタにより生成された乱数を表す信号である。2 つ目の信号は、ハードラッチ選択レジスタ 9 0 2 2、9 0 2 4、9 0 2 6、9 0 2 8 からそれぞれ出力された乱数ラッチ信号（ハードラッチ信号）である。3 つ目の信号は、乱数の読み取りを示すリード信号である。

#### 【 2 2 8 8 】

10

ハードラッチ乱数値レジスタ 9 0 3 2 a、9 0 3 4 a、9 0 3 6 a、9 0 3 8 a には、対応するカウンタ回路 0 ~ 3 の 8 ビットカウンタにより生成された乱数を表す信号が常に入力されている。ここで、ラッチ信号が入力されると、この入力タイミングにおける乱数がハードラッチ乱数値レジスタ内にラッチ（保持）される。このとき、ハードラッチ乱数値レジスタからは、乱数がラッチされていることを示すセット信号が、ハードラッチフラグレジスタ 9 0 4 8 a、9 0 4 8 b に出力される。このとき CPU 3 0 4 は、ラッチされている乱数を取得することができる。なお、CPU 3 0 4 により乱数が取得されると、ハードラッチ乱数値レジスタにリード信号が入力される。この信号により新たなラッチ信号が入力された場合に乱数をラッチすることを許容する許容状態となる。換言すれば、一度乱数をラッチすると、リード信号が入力されるまで新たに乱数をラッチすることができない非許容状態となる。なお、リード信号が入力されてもラッチされている乱数は保持し続けるため、CPU 3 0 4 はラッチされている同一のタイミングでラッチされた乱数を何度でも取得することができる。このように構成することで乱数ラッチ信号を出力するセンサ回路におけるチャタリングの影響を押さえることができる。なお、リード信号が入力されたことを示すクリア信号がハードラッチフラグレジスタ 9 0 4 8 a、9 0 4 8 b に出力される。

20

#### 【 2 2 8 9 】

ハードラッチフラグレジスタ 9 0 4 8 a、9 0 4 8 b には、ハードラッチ乱数値レジスタに乱数がラッチされているか否かを示す情報が記憶される。なお、ハードラッチフラグレジスタ 9 0 4 8 a、9 0 4 8 b は、ハードラッチ乱数レジスタにラッチされた乱数（8

30

#### 【 2 2 9 0 】

本変形例によるカウンタ回路 3 1 2 のカウンタ回路 0 ~ 3 に設けられている 8 ビットカウンタの起動時期は、カウント値を設定しない場合と、カウント値を設定する場合とで異ならせることができる。カウント値を設定しない場合の 8 ビットカウンタは、主制御部メイン処理のプログラム起動設定処理中に起動される。また、カウント値を設定する場合の 9 ビットカウンタは、主制御部メイン処理の初期設定 2（ステップ S 1 5 1 0 5）内でカウント値を設定した後に起動される。カウンタ回路 0 ~ 3 の 8 ビットカウンタの起動時期はそれぞれ個別に設定することができる。

40

#### 【 2 2 9 1 】

以上説明したカウンタ回路 3 1 2 は、数値更新手段の一例に相当する。

#### 【 2 2 9 2 】

本変形例によるパチンコ機 1 0 0 で用いられるカウンタ回路 3 1 2 の 8 ビットカウンタは、ハード乱数カウンタである。

ハード乱数カウンタは、内部システムクロック（S C L K）または外部クロック（R C K）で更新される。

ハード乱数カウンタの更新周期は、ソフト乱数の更新周期よりも短い。

ハード乱数カウンタの更新周期は、割込み処理の割込み周期よりも短い。

ハード乱数カウンタの更新時期は、プログラム処理に依存しないが、最大値設定時は更

50



新開始時期（起動時期）のみがプログラム処理に依存する

なお、ソフト乱数カウンタは、プログラム処理内の更新処理毎に更新される。

ソフト乱数の更新時期は、プログラム処理に依存する。

【2293】

次に、以上説明した実施形態7の変形例によるパチンコ機100の特徴的構成について再度図284を参照しつつ説明する。

（1）本実施の形態によるパチンコ機100は、

乱数を少なくとも導出可能なハード乱数カウンタ（例えば、8ビットカウンタ3122）と、

複数の遊技制御処理のそれぞれを、メイン制御（例えば、主制御部メイン処理）および割込み周期ごとに行われる割込み制御（例えば、主制御部タイマ割込処理）のうちの、少なくともいずれかの制御において実行可能な遊技制御手段（例えば、主制御部300）と、を備えた遊技台であって、

前記ハード乱数カウンタは、第一の数値範囲内で更新される第一の乱数（例えば、最大値設定で起動される乱数生成回路が生成する乱数値）を導出可能なものであり、

前記ハード乱数カウンタは、第二の数値範囲内で更新される第二の乱数（例えば、ユーザモード起動で起動される乱数生成回路が生成する乱数値）を導出可能なものであり、

前記第一の乱数の起動時期と前記第二の乱数の起動時期（例えば、カウンタ回路0～3の8ビットカウンタの起動時期）は異なるものであること

を特徴とする

【2294】

当該構成を備えたパチンコ機100によれば、2つのハード乱数カウンタの起動時期が異なるため、不正者による2つの乱数の解析を困難にでき、安定した遊技制御を行うことができる場合がある。

【2295】

（2）上記パチンコ機100であって、

第一の処理（例えば、主制御部メイン処理のステップS15103）を前記遊技制御処理は少なくとも含むものであり、

前記第一の乱数の起動時期と前記第二の乱数の起動時期のうちの一方は、前記第一の処理よりも後であること

を特徴とする

【2296】

当該構成を備えたパチンコ機100によれば、ハード乱数カウンタの起動時期が第一の処理後であるため、不正者による不正をより困難にでき、安定した遊技制御を行うことができる場合がある。

【2297】

（3）上記パチンコ機100であって、

前記第一の処理は、制御の進行を保留可能なものであり、

前記第一の処理は、制御進行条件が成立した場合に（例えば、主制御部メイン処理のステップS15103のYes）、前記保留を解除可能なものであり、

前記制御進行条件は、第一の信号が第一の状態から第二の状態に変化した場合に成立するものであること

を特徴とする。

【2298】

当該構成を備えたパチンコ機100によれば、第一の信号の状態の変化時期に応じて起動時期が変化するため、不正者による不正をより困難にでき、安定した遊技制御を行うことができる場合がある。

【2299】

（4）上記パチンコ機100であって、

前記ハード乱数カウンタに前記数値範囲の設定を指示可能な第二の処理（例えば、主制

10

20

30

40

50

御部メイン処理のステップS 1 5 1 3 3)と、

前記ハード乱数カウンタによって導出された乱数を取得可能な第三の処理(例えば、主制御部タイマ割込処理のステップS 1 5 2 1 7)と、

前記第三の処理によって取得された乱数に基づく判定を実行可能な第四の処理(例えば、主制御部タイマ割込処理のステップS 1 5 2 2 3、ステップS 1 5 2 2 9、ステップS 1 5 2 3 1)と、を前記遊技制御処理は少なくとも含むものであり、

前記第一の数値範囲は、前記第二の処理によって設定されるものであり、

前記第二の数値範囲は、前記第二の処理によらずに設定されるものであり、

前記第一の数値範囲が設定された場合に、前記第一の乱数の起動が開始可能なものであること

10

を特徴とする。

#### 【2300】

当該構成を備えたパチンコ機100によれば、数値範囲設定後に更新するので数値範囲外の数値を更新することがなく、安定した遊技制御を行うことができる場合がある。

#### 【2301】

以上説明した本願発明によるパチンコ機100における上述の<第4特殊命令/RST命令>についての特徴的構成について以下に付記する。

##### (付記1)

CPUを内蔵するマイクロコンピュータを備えた遊技台であって、

前記遊技台は、ぱちんこ機またはスロットマシンであり、

20

前記CPUは、特定のジャンプ命令を受け付けたことに基づいて、ジャンプ先アドレスに記憶されている命令を受け付けるように構成されており、

前記特定のジャンプ命令には、アドレス情報が含まれているものであり、

前記アドレス情報として特定のアドレス情報が含まれている場合には、前記ジャンプ先アドレスは、当該アドレス情報と特定の値を乗算した場合に得られる値と一致しない所定の値により示される所定のアドレスとなるものであり、

前記アドレス情報として前記特定のアドレス情報とは異なるアドレス情報が含まれている場合には、前記ジャンプ先アドレスは、当該アドレス情報と前記特定の値を乗算した場合に得られる値と一致する値により示されるアドレスとなるものである、

ことを特徴とする遊技台。

30

##### (付記2)

付記1に記載の遊技台において、

前記特定のアドレス情報が0であることから、前記CPUは、前記特定のジャンプ命令の次に0番地に記憶されている命令を受け付けることが禁止されているものである、

ことを特徴とする遊技台。

##### (付記3)

付記1または2に記載の遊技台であって、

前記CPUは、少なくとも特定のレジスタを備え、

前記CPUは、前記特定のレジスタに値をセットする機能のうち、ロード命令を受けたことに基づいて行われるものとしては、直値により値をセットする機能のみを有することを特徴とする遊技台。

40

##### (付記4)

付記3に記載の遊技台であって、

前記CPUは、前記特定のレジスタに値をセットする機能のうち、前記ロード命令を受けたことに基づいて行われるもの以外のものとして、所定の値を初期値としてセットする機能を少なくとも有することを特徴とする遊技台。

##### (付記5)

付記1～4のいずれかに記載の遊技台であって、

遊技制御を行う遊技制御手段と、

演出制御を行う演出制御手段と、を備え、

50

前記CPUを内蔵した前記マイクロコンピュータは、前記演出制御手段に搭載されず、前記遊技制御手段にのみ搭載されることを特徴とする遊技台。

(付記6)

付記1～4のいずれかに記載の遊技台であって、  
遊技制御を行う遊技制御手段と、  
演出制御を行う演出制御手段と、  
払出制御を行う払出制御手段と、を備え、

前記CPUを内蔵した前記マイクロコンピュータは、前記演出制御手段に搭載されず、前記遊技制御手段および前記払出制御手段のうちの少なくとも一方に搭載されることを特徴とする遊技台。

10

(付記7)

CPUを少なくとも内蔵するマイクロコンピュータを備えた遊技台であって、  
前記遊技台は、ぱちんこ機またはスロットマシンであり、  
前記CPUは、第一のジャンプ命令を受け付けたことに基づいて、第一のジャンプ先アドレスに記憶されている命令を受け付けることが可能なものであり、  
前記CPUは、第二のジャンプ命令を受け付けたことに基づいて、第二のジャンプ先アドレスに記憶されている命令を受け付けることが可能なものであり、  
前記第一のジャンプ命令には、3ビットの第一の識別情報が少なくとも含まれており、  
前記第二のジャンプ命令には、3ビットの第二の識別情報が少なくとも含まれており、  
前記第一のジャンプ先アドレスは、第一の値により示されるアドレスであり、  
前記第二のジャンプ先アドレスは、第二の値により示されるアドレスであり、  
前記第一の値は、前記第一の識別情報の8倍の値であり、  
前記第二の値は、前記第二の識別情報の8倍の値とは異なり、3ビットで示すことが可能な最大値を1だけ超えた値の8倍の値である、  
ことを特徴とする遊技台。

20

(付記8)

付記7に記載の遊技台であって、  
前記第二の識別情報は0である、  
ことを特徴とする遊技台。

(付記9)

付記7または8に記載の遊技台であって、  
前記第一の識別情報は1から7のいずれかの値である、  
ことを特徴とする遊技台。

30

(付記10)

付記9に記載の遊技台であって、  
前記第二のジャンプ先アドレスは、前記第一の識別情報の最大値である7を1だけ超えた値である8の8倍の値である64である、  
ことを特徴とする遊技台。

(付記11)

請求項7乃至10のいずれかに記載の遊技台であって、  
前記CPUは、前記第一のジャンプ命令および前記第二のジャンプ命令とは異なるジャンプ命令を受け付けることも可能なものである、  
ことを特徴とする遊技台。

40

【2302】

以下、図285および図286を用いて、本発明の実施形態に係るパチンコ機（遊技台）について説明する。図285は、CPU304によって実行される処理A～処理Dを示している。処理A（例えば、定期割り込み処理、メインループ等）は、ROM306のアドレスA～アドレスBまでの間に記憶された処理であり、処理a1、先頭アドレスがアドレスCである処理を呼び出すことが指定された呼出命令A（例えば、RST命令等）、処理a2、先頭アドレスがアドレスEである処理を呼び出すことが指定された呼出命令A（例

50

えば、R S T命令等）、処理a 3、先頭アドレスがアドレスFである処理を呼び出すことが指定された呼出命令B（例えば、E X E S U B命令等）、および処理a 4の順で各処理・各命令がC P U 3 0 4によって実行されることを示している。

#### 【2303】

処理B（例えば、データ処理1等）は、R O M 3 0 6のアドレスC～アドレスDまでの間に記憶された処理であり、処理b 1、先頭アドレスがアドレスFである処理を呼び出すことが指定された呼出命令B（例えば、E X E S U B命令等）、N O P処理（例えば、一つのN O P命令）、処理b 2、および処理Cの順で各処理・各命令がC P U 3 0 4によって実行されることを示している。

#### 【2304】

処理C（例えば、データ処理2等）は、R O M 3 0 6のアドレスE～アドレスDまでの間に記憶された処理である。

処理D（例えば、データ処理3等）は、R O M 3 0 6のアドレスF～アドレスGまでの間に記憶された処理である。

#### 【2305】

図285および図286における”命令”は、一つの命令操作部（オペコード）と当該命令操作部に基いてC P U 3 0 4が動作する際に必要な数のオペランド部（オペランド）からなるC P U 3 0 4が受け付け可能な単一セットの指示を指す用語である。

#### 【2306】

図285および図286における”処理”は、データ処理を指す用語であり、一つの命令から構成されるものや複数の命令から構成されるもの、入力データ等によってC P Uによって実行される命令の数が変化するものを含むものである。

#### 【2307】

C P U 3 0 4は、R S T命令（図65等参照）を受け付けた場合に、予め定められた複数のアドレス（例えば、08H、10H、18H、20H、28H、30H、38H、40Hの8つのアドレス等）のうちの当該R S T命令のオペランド部で示されるアドレスを先頭アドレスとする処理を実行するものであり、C P U 3 0 4は、当該処理の実行を終了（例えばR E T命令により復帰）した場合は、当該R S T命令の次に記憶されている命令を読み出して実行するように構成されている。

#### 【2308】

C P U 3 0 4は、E X E S U B命令（図25～図28等を参照）を受け付けた場合に、予め定められたアドレス範囲（例えば0000H～11FFHの4キロバイト）のうちの当該E X E S U B命令のオペランド部で示されるアドレスを先頭アドレスとする処理を実行するものであり、C P U 3 0 4は、当該処理の実行を終了（例えばR E T命令により復帰）した場合は、当該E X E S U B命令の次に記憶されている命令を読み出して実行するように構成されている。

#### 【2309】

これらの構成によりC P U 3 0 4は、処理a 1、先頭アドレスがアドレスCである処理を呼び出すことが指定された呼出命令A（例えば、R S T命令等）、処理b 1、先頭アドレスがアドレスFである処理を呼び出すことが指定された呼出命令B（例えば、E X E S U B命令等）、N O P処理、処理b 2、処理C、処理a 2、先頭アドレスがアドレスEである処理を呼び出すことが指定された呼出命令A（例えば、R S T命令等）、処理C、処理a 3、先頭アドレスがアドレスFである処理を呼び出すことが指定された呼出命令B（例えば、E X E S U B命令等）、処理D、および処理a 4の順で各処理・各命令を実行する。

#### 【2310】

ここでN O P処理は、N O P命令を少なくとも含む処理であり、一つのN O P命令のみから構成されるもの、複数のN O P命令のみから構成されるもの、一または複数のN O P命令と一または複数の他の命令によって構成されたもの等を適用してもよい。

#### 【2311】

CPU 304は、NOP命令を受け付けた場合には、何もせずに、またはNOP命令を受け付けた場合特有の動作をせずにNOP命令時間の経過後に当該NOP命令の次に記憶されている命令を読み出して実行するように構成されている。またNOP命令はオペランド部を必要とせず、所定のステート数（例えば4ステート）、所定の命令長（例えば1バイト）で構成されている。

#### 【2312】

しかし図285で示される処理は、遊技機（スロットマシン、ぱちんこ機、遊技媒体封入式のスロットマシン、遊技媒体封入式のぱちんこ機等）にとっては、NOP処理が不正者によって他の不正処理に置き換えられてしまうといった非常に大きな問題を有している。ROM 306に記憶されている情報が点検者にとって確認しがたい数値（命令を示す数値、すなわち命令操作部・オブジェクト部を構成する数値）情報で構成されていることも不正者に不正利用される要因になっている。

10

#### 【2313】

例えばROM 306のアドレスHから始まる領域に不正処理を記憶する改造が不正者によって行われ、かつNOP処理を特定の呼出処理（不正な処理）に置き換える改造が不正者によって行われてしまったとする。ここで特定の呼出処理は、当該アドレスHから始まる処理を呼び出すための呼び出し処理だとする。このような改造が行われると、CPU 304は処理Bの実行時に不正者による改造によって記憶された不正処理を呼び出し、実行してしまうことになる。ここでは、正規の遊技機に取り付けられていたROM 306の記憶内容を記憶し直す改造を例示したが、不正者によっては、正規の遊技機に取り付けられていたROM 306を上述のような特定の呼出処理・不正処理が記憶された不正なROMに交換する手口や、正規の遊技機に取り付けられていたマイクロプロセッサ3000（ROM 306を含む（例えば内蔵する）もの）を不正なマイクロプロセッサ（上述のような特定の呼出処理・不正処理が記憶された不正なROMを内蔵するもの）に交換する手口なども使われる。

20

#### 【2314】

また不正処理は、不正な入力（例えばマイクロプロセッサに入力される所定の信号を監視し、当該信号の変化パターンが特定の変化パターンである場合にオンと判定される信号）があった場合に、所定の不正RAM設定処理（例えばRAM 308の遊技状態を示すフラグ（ぱちんこ機の場合の大当たりフラグ、ぱちんこ機の場合の確変フラグ、スロットマシンの場合のBB・RB中かどうかを示すフラグ等）に、不正者の都合のよい状態を示す値（ぱちんこ機の場合の大当たり中であることを示す値、ぱちんこ機の場合の確変中であることを示す値、スロットマシンの場合のBB・RB中であることを示す値等）を設定する）を行うような処理が一般的だが、これに限らず、単に時間を費やすだけの処理など不正者による様々な処理のうちの一つまたは複数である場合があり得る。

30

そこで、このNOP処理を使わずに遊技制御を実現したものを図286に示す。

#### 【2315】

図286は、CPU 304によって実行される処理A～処理Dを示している。図286における処理・命令のうち図285と同じ処理・命令は、同じ名称としている。

40

図286の処理A（例えば、定期割り込み処理、メインループ等）は、ROM 306のアドレスA～アドレスBまでの間に記憶された処理であり、処理a1、先頭アドレスがアドレスCである処理を呼び出すことが指定された呼出命令A（例えば、RST命令等）、処理a2、先頭アドレスがアドレスEである処理を呼び出すことが指定された呼出命令A（例えば、RST命令等）、処理a3、先頭アドレスがアドレスFである処理を呼び出すことが指定された呼出命令B（例えば、EXESUB命令等）、および処理a4の順で各処理・各命令がCPU 304によって実行されることを示している。

#### 【2316】

処理B（例えば、データ処理1等）は、ROM 306のアドレスC～アドレスDまでの間に記憶された処理であり、処理b1、先頭アドレスがアドレスFである処理を呼び出すことが指定された呼出命令C（例えば、CALL命令等）、処理b2、および処理Cの順で

50

各処理・各命令がCPU 304によって実行されることを示している。

【2317】

処理C(例えば、データ処理2等)は、ROM 306のアドレスE～アドレスDまでの間に記憶された処理である。

処理D(例えば、データ処理3等)は、ROM 306のアドレスF～アドレスGまでの間に記憶された処理である。

【2318】

CPU 304は、CALL命令を受け付けた場合に、当該CALL命令のオペランド部で示されるアドレスを先頭アドレスとする処理を実行するものであり、CPU 304は、当該処理の実行を終了(例えばRET命令により復帰)した場合は、当該CALL命令の次に記憶されている命令を読み出して実行するように構成されている。

10

【2319】

これらの構成によりCPU 304は、処理a1、先頭アドレスがアドレスCである処理を呼び出すことが指定された呼出命令A(例えば、RST命令等)、処理b1、先頭アドレスがアドレスFである処理を呼び出すことが指定された呼出命令C(例えば、CALL命令等)、処理b2、処理C、処理a2、先頭アドレスがアドレスEである処理を呼び出すことが指定された呼出命令A(例えば、RST命令等)、処理C、処理a3、先頭アドレスがアドレスFである処理を呼び出すことが指定された呼出命令B(例えば、EXESUB命令等)、処理D、および処理a4の順で各処理・各命令を実行する。

【2320】

EXESUB命令およびCALL命令の両方の処理は、処理Dを呼び出すことが可能な処理であるが、EXESUB命令のサイズは2バイト、ステート数は13であり、一方CALL命令のサイズは3バイト、ステート数は16であり、処理Dの先頭アドレスであるアドレスFが予め定められたアドレス範囲内であれば、CALL命令を用いる場合よりもEXESUB命令を用いる場合の方が、サイズのなメリット、ステート数(処理速度)的なメリットを有することになる。

20

【2321】

しかし、RST命令が上述のとおり予め定められた複数のアドレスのいずれかのアドレスしか呼び出せないことにより、第一の処理(ここではアドレスCからアドレスEに記憶されたNOP以外の複数の命令)のサイズが所定のサイズ(例えば、7バイト、7バイト以下等)である場合は、第二の処理(ここでは処理C)を単独で呼び出したり、当該第二の処理を含む第三の処理(ここでは処理B(第一の処理および第二の処理の両方からなる処理))を呼び出したりすることがNOP処理を含めなければできなくなってしまう。

30

【2322】

そこで第一の処理では呼出命令B(ここではEXESUB命令)を用いず、代わりに呼出命令C(ここではCALL命令)を用いることで、サイズ・ステート数の両面で不利(サイズの大きいことで不利、処理速度的に遅いことで不利)にはなるが、当該第一の処理をアドレスC(例えば20H)からアドレスE(例えば28H)の間に収めることができる、すなわち当該第一の処理のサイズをNOP処理を用いずに8バイトにすることができるので、不正者が不正な処理を当該NOP処理と置き換えるようなことを防止でき、不正に強い遊技制御プログラムをROM 306に記憶することが可能となる。

40

【2323】

ここでは図285と図286を用いて、呼出命令Bの代わりに、命令(処理)のサイズ・命令の処理時間は異なるが処理の結果が当該呼出命令Bと同じとなる呼出命令Cを用いることで、NOP処理を使わない例を示したが、これに限定されず、第二の命令の代わりに、命令(処理)のサイズ・命令の処理時間は異なるが処理の結果が当該第二の命令と同じとなる第一の命令を用いることで、NOP処理を使わないような処理であればよい。ここでは第二の命令と第一の命令の例として、EXESUB命令とCALL命令を示したが、その他の例としては、JR命令とJP命令、RST命令とEXESUB命令、RST命令とCALL命令などマイクロプロセッサ3000が受け付け可能な命令のうちの「第二

50

の命令」と「命令（処理）のサイズ・命令の処理時間は異なるが処理の結果が当該第二の命令と同じとなる第一の命令」を適用するようにすればよい。ここでは第一・第二の命令の例として分岐命令を示したが、第一・第二の命令の両方を転送命令、サーチ命令、算術演算処理命令、論理演算処理命令、汎用算術機能、MPU制御命令、ローテイト命令、シフト命令、ビット処理命令、ジャンプ命令、コール命令、シフト命令、および入出力命令のうちのいずれかの命令としてもよい。また「第一の複数の命令のサイズの合計・処理時間の合計」と「複数の命令のサイズの合計・複数の命令の処理時間の合計は異なるが処理の結果が当該第一の複数の命令と同じとなる第二の複数の命令」に適用してもよいし、「第二の命令のサイズ・処理時間」と「複数の命令のサイズの合計・複数の命令の処理時間の合計は異なるが処理の結果が当該第二の命令と同じとなる第二の複数の命令」に適用してもよいし、「第一の複数の命令のサイズの合計・処理時間の合計」と「命令のサイズ・命令の処理時間は異なるが処理の結果が当該第一の複数の命令と同じとなる第一の命令」に適用してもよい。ここで第一の複数の命令に含まれる命令および第二の複数の命令に含まれる命令は、マイクロプロセッサ3000が受け付け可能な命令であれば、どのような命令を適用してもよい。また「命令のサイズ・処理時間が異なる」または「命令のサイズの合計・処理時間の合計が異なる」と記載してきたが、少なくとも「命令のサイズ」または「命令のサイズの合計」と置き換えても同様の効果が得られる。第一の命令が実行された後のフラグレジスタの値と第二の命令のフラグレジスタの値については、同一であっても同様の効果が得られる場合がある。もちろん第一の命令が実行された後のフラグレジスタの値と第二の命令のフラグレジスタの値については、異なっている場合もある。

#### 【2324】

なお、「処理Bを構成する複数の命令」のうち「処理Cを構成する複数の命令」以外の命令（図286の場合であれば、処理b1を構成する一または複数の命令、b2を構成する一または複数の命令、および呼出命令C）にNOP命令が含まれていてもよく、NOP命令が含まれていたとしても呼出命令CによりNOP命令の数が減少しているので、不正がし難くなるという効果を奏する場合がある。

#### 【2325】

NOP命令は上述の「処理Bを構成する複数の命令」のうち「処理Cを構成する複数の命令」以外の命令に一切含まれていなくてもよく、またROM306に記憶され、CPU304によって実行される全ての命令をNOP以外の命令としてもよい。

ここで不正者による不正行為に利用される命令としてNOP命令を例示してきたが、「LD A, A」のようなCPUによって受け付けられ実行されても、当該実行の前後で何もかわらないような処理をNOP命令と同様に考えるようにしてもよい。

#### 【2326】

またアドレスEからアドレスCを差し引いた値は、処理Bのサイズから処理Cのサイズを差し引いた値としてもよい場合がある。アドレスEからアドレスCを差し引いた値よりも、処理Bのサイズから処理Cのサイズを差し引いた値の方が小さくてもよい場合がある。

#### 【2327】

図286について、処理a1、処理a2、処理a3、処理a4、処理b1、処理b2、および処理Dのうちの一または複数の処理を省略するようにしてもよい。また呼出命令Bについても省略してもよい。

#### 【2328】

同一の処理（サブルーチン）の複数の命令に、先頭アドレスがアドレスCである処理Bを呼び出すことが指定された呼出命令Aと、アドレスEである処理Cを呼び出すことが指定された呼出命令Aと、が含まれていなくても、処理Aにおける複数の命令に先頭アドレスがアドレスCである処理Bを呼び出すことが指定された呼出命令Aが含まれ、処理Aとは別の処理Eにおける複数の命令に先頭アドレスがアドレスEである処理Cを呼び出すことが指定された呼出命令Aが含まれていてもよい。

## 【2329】

処理のサイズについては、当該処理から他の処理を呼び出す呼出命令のサイズは含むようにしてもよく、当該呼出命令によって呼び出された処理のサイズは含まないものとしてもよい。

## 【2330】

処理のステート数（処理時間）については、当該処理から他の処理を呼び出す呼出命令のステート数（処理時間）は含むようにしてもよく、また当該呼出命令によって呼び出された処理のステート数（処理時間）も含むようにしてもよい。

## 【2331】

第一の命令（例えばCALL命令）は、ROM306に記憶された全ての遊技制御プログラムのうち、上述の呼出命令Aから呼び出されるサブルーチン内にのみ用いられるように構成されていてもよい。もちろん上述の呼出命令Aによって当該サブルーチンが呼び出されてから終了するまでの間にだけ用いられるように構成されていてもよく、例えば当該サブルーチンから呼び出されるサブルーチン内にのみ用いられているように構成されていてもよい。また上述の呼出命令Aから呼び出されるサブルーチン以外で用いられていてもよい

10

## 【2332】

第二の命令（例えばEXESUB命令）は、ROM306に記憶された全ての遊技制御プログラムのうち、上述の呼出命令Aから呼び出されるサブルーチン内にのみ用いられるように構成されていてもよい。もちろん上述の呼出命令Aによって当該サブルーチンが呼び出されてから終了するまでの間にだけ用いられるように構成されていてもよく、例えば当該サブルーチンから呼び出されるサブルーチン内にのみ用いられているように構成されていてもよい。また上述の呼出命令Aから呼び出されるサブルーチン以外で用いられていてもよい

20

## 【2333】

なお、本実施例で例示した命令として他の実施例で例示した命令を適用してもよいし、逆に他の実施例で例示した命令の一形態として本実施例で例示した命令を適用してもよい。例えば第一、第二の呼出命令のうちの少なくとも一方の命令に、他の実施例で例示した”RST 40H”を適用しても良いし、第一、第二の呼出命令のうちの少なくとも一方の命令に、他の実施例で例示した「”RST 40H”以外の7つのRST命令のいずれか」を適用してもよい。

30

## 【2334】

これらを包含するように記載すると、

CPUを内蔵するマイクロプロセッサを備えた遊技台Aであって、

上述の遊技台Aは、ぱちんこ機またはスロットマシンであり、

上述のCPUは、第一の命令（例えばCALL命令）を受け付けた場合に第一の動作を少なくとも実行可能なものであり、

上述のCPUは、第二の命令（例えばEXESUB命令）を受け付けた場合に第二の動作を少なくとも実行可能なものであり、

上述の第一の命令のサイズは、上述の第二の命令のサイズよりも大きいものであり、

40

上述のCPUは、第一の呼出命令（例えば、”RST 38H”、図64における命令コード”FFH”の命令等）を受け付けた場合に、第一のサブルーチンに含まれる第一の複数の命令を少なくとも受付可能なものであり、

上述のCPUは、第二の呼出命令（例えば、”RST 40H”、図64における命令コード”C7H”の命令等）を受け付けた場合に、第二のサブルーチンに含まれる第二の複数の処理を少なくとも受付可能なものであり、

上述の第一の複数の命令は、第三の複数の命令および上述の第二の複数の命令から構成されたものである、

ことを特徴とする遊技台A。なお、この記載の一部の構成を省略してもよいし、複数の構成を省略してもよい。また命令コードの値は一例であり、他の値であってもよい。

50



## 【2335】

遊技台 A について、

上述の CPU は、NOP 命令を受け付けた場合に、当該 NOP 命令の次の命令を受け付けるものであり、

上述の第三の複数の命令は、NOP 命令を含まず、上述の第一の命令を少なくとも含むものであってもよい。

## 【2336】

遊技台 A について、

上述の CPU は、NOP 命令を受け付けた場合に、当該 NOP 命令を受け付けた場合にのみ実行される動作は行わず、上述の次の命令を受け付けるものであってもよい。

10

## 【2337】

遊技台 A について、

上述の第二のサブルーチンの先頭アドレス（例えば 40 H 等）から上述の第一のサブルーチンの先頭アドレス（例えば 38 H 等）を引いた値（ここでは 8 バイト）は、上述の第三の複数の命令のサイズを示す値と等しいものであってもよい。

## 【2338】

遊技台 A について、

上述の CPU は、呼出命令を受け付けた場合に、当該呼出命令によって指定されたアドレスから始まるサブルーチンに含まれる複数の命令を少なくとも実行可能なものであり、当該複数の命令のうちの最後の RET 命令を受け付けた場合に、当該呼出命令の次の命令を受け付けて少なくとも実行可能なものであってもよい。

20

## 【2339】

遊技台 A について、

上述の CPU は、呼出命令を受け付けた場合に、当該呼出命令によって指定されたアドレスから始まるサブルーチンに含まれる複数の命令を少なくとも実行可能なものであり、当該複数の命令のうちの最後の RET 命令を受け付けた場合に、当該呼出命令の次の命令を受け付けて少なくとも実行可能なものであってもよい。

## 【2340】

遊技台 A について、

上述の第二の複数の命令における最後の命令を、上述の RET 命令としてもよい、

30

## 【2341】

遊技台 A について、

上述の第一の複数の命令における最後の命令を、上述の RET 命令としてもよい、

上述の第一の複数の命令には、上述の RET 命令が一つのみ含まれているものとしてもよい。

## 【2342】

遊技台 A について、

上述の第二の複数の命令は、上述の第三の複数の命令に続いて ROM に記憶されたものであり、

上述の第三の複数の命令には、上述の第二のサブルーチンを呼び出す呼出命令が含まれていないものとしてもよい。

40

## 【2343】

第二のサブルーチンの先頭アドレスとして 40 H、第一のサブルーチンの先頭アドレスとして 38 H 等、第二のサブルーチンの先頭アドレスから第一のサブルーチンの先頭アドレスを引いた値が 08 H（8 バイト）となる例を示したが、これに限定されず、第二のサブルーチンの先頭アドレスとして 38 H、第一のサブルーチンの先頭アドレスとして 018 H 等としてもよい。第一のサブルーチンの先頭アドレスを RST 命令で呼び出せるアドレスのうちのいずれかとしてもよいし、第二のサブルーチンの先頭アドレスを RST 命令で呼び出せるアドレスのうちのいずれかとしてもよい。

## 【2344】

50

なお、第一、第二の呼出命令のうちの少なくとも一方の命令に、他の実施例で例示した「RST 40H」（第二のジャンプ命令）を適用しても良いし、第一、第二の呼出命令のうちの少なくとも一方の命令に、他の実施例で例示した「RST 40H」以外の7つのRST命令のいずれか（第一のジャンプ命令）」を適用してもよい。例えば、第一の呼出命令に第二のジャンプ命令を適用し、第二の呼出命令に第一のジャンプ命令を適用してもよい。第一、第二の呼出命令やRST命令に限らず、本実施例で例示した命令として他の実施例で例示した命令を適用してもよいし、逆に他の実施例で例示した命令の一形態として本実施例で例示した命令を適用してもよい。

#### 【2345】

上記実施形態では、内蔵デバイスが出力する割込みベクタとIレジスタとで作成したジャンプ先アドレスのデータとその次のアドレスのデータとを用いて割込み要求が発生した場合の処理の先頭アドレスを決めることを説明した。また、＜第4特殊命令／RST命令＞の項では、図64および図65を用いてRST命令について説明した。さらに、ここではそれらに関連する事項について図287および図288を用いて説明する。

#### 【2346】

まず、内蔵デバイスが出力する割込みベクタとIレジスタとで作成される割込み要求発生時のジャンプ先アドレスの決め方について説明する。Iレジスタは電源投入後に操作していないので00Hになっている。つまりジャンプ先アドレスの上位バイトは「00H」になっている。

図287は割込み初期設定レジスタの設定例を示している。割込み初期設定レジスタはプログラム管理エリア内の所定位置に設定される。図287に示すように、割込み初期設定レジスタのビット7～4には、割込みベクタの上位4ビットが設定される。割込みベクタの上位4ビットはユーザが任意に決められる値である。以下、割込みベクタの上位4ビットを「XH」と表記する。ビット3はノンマスカブル割込み中の多重受付の禁止・許可をするためにあり、多重受付禁止の場合は「0」をセットし、多重受付許可の場合は「1」をセットする。ビット2～0はマスカブル割込み要因の優先度の組合せを設定する。割込み要因にはタイマ回路のチャンネル0のタイムアウトによる割込み（PTOI）やその他の種々の割込み要因（図示の通り7個の割込み要因）がある。本例では、タイマ回路のチャンネル0（例えば、図132のタイマ回路0）による割込み（PTOI）のみを有効にするものとする。なお、他の割込みを無効にしていれば、いずれの設定値でも問題ないが設定ミスを考えてタイマ回路が優先されている。タイマ回路のチャンネル0のタイムアウトによる割込み（PTOI）は図示の通り、00H～02Hのいずれかが選択可能である。タイマ回路以外の割込みを無効にしているのであれば、00H～02Hのいずれでもよいが、デフォルトの00Hから他の値に変えるとコーディングミスにつながる可能性があるので00Hを設定することが好ましい。このため、多重受付禁止としてビット3に「0」を設定した場合は割込みベクタの下位4ビットは0000Bとなり、多重受付許可としてビット3に「1」を設定した場合は割込みベクタの下位4ビットは1000Bとなる。これにより、割込みベクタの下位4ビットの設定値が16進数で0Hまたは8Hとなるのでコーディングや、チェックがし易くなる場合がある。遊技制御を安定して行うためにも割込みベクタの下位4ビットは0000Bか1000Bにするのが好ましい。こうすることにより、ジャンプ先アドレスの下位バイトは「X0H」または「X8H」になり、ジャンプ先アドレスは、「00X0H」または「00X8H」となる。なお、Iレジスタにデータを設定する場合には、「XXX0H」や「XXX8H」も設定可能である。

#### 【2347】

ところで、RST命令では8種類のアドレス（0008H、0010H、0018H、0020H、0028H、0030H、0038H、0040H）だけにジャンプすることが可能となっている。つまり、（RST 40H）命令以外のRST命令で組み込めるモジュールの容量は8バイト未満であることが望ましい。仮に、ジャンプ先アドレスが0008Hから8バイトを超える大きさのモジュールを組み込むと次のジャンプ先アドレスの0010HのRST命令が実質的に使えなくなる。そこで、8バイトを超える大きさの

モジュールを(RST 40H)命令により「0040H」がジャンプ先アドレスとなる領域に配置する場合がある。なお、RST 40Hで呼び出すモジュールのサイズが8バイトを超えるものを前提で説明するが、RST 40Hで呼び出すモジュールのサイズが8バイト未満であってももちろんよい。また、RST命令の8種類の各アドレスで呼び出すモジュールは、全てが8バイト丁度であってもよいし、全てが8バイト未満であってもよいし、また、一部のみが8バイト丁度であってもよいし、一部のみ8バイト未満であってもよい。また、RST 40Hで呼び出すモジュールのサイズのみが、8バイト丁度であってもよいし、8バイト未満であってもよい。このように、RST 08~40で呼び出すモジュールは8バイトでなくてもよく、一部を8バイト超にして一部を8バイト未満にしてもよい。あるいは、8バイト超のモジュールは、アドレスが「0038H」で用いてもよく、あるいは、「0040H」を超えないようにしてもよい。

#### 【2348】

図288は、ROM領域内のアドレス0040Hからの一部領域を示している。同図(a)は、アドレス「0040H」をRST命令のジャンプ先アドレスとし、「0050H」を割込み要求発生時のジャンプ先アドレスとした場合を例示している。この場合には、8バイトを超える大きさのモジュールAが配置でき、残りは余り領域Bとなっている。本例で余り領域は、図285に示すNOP処理に相当している。同図(b)は、モジュールAの次のアドレスから他のモジュールB(例えば、EXSUB命令等によって呼び出されるモジュールである。なお、このモジュール内では、2バイトのジャンプ命令に代えてあえて3バイトのジャンプ命令を使用したりしてもよい)を組み込んで余り領域Dを余り領域Bより小さくしている。同図(c)は、「0060H」を割込み要求発生時のジャンプ先アドレスとした場合を例示している。この場合にはモジュールAの次のアドレスからモジュールAよりサイズの大きな他のモジュールEを組み込むことができる。また、余り領域Fが生じている。このように、割込み要求発生時のジャンプ先アドレスを調整することにより、アドレス0040Hから割込み要求発生時のジャンプ先アドレスまでのメモリ領域を有効に使うことができる場合がある。

また、余り領域を現実的に0にするのは困難を伴うので、不正者による不正使用を極力排除するために、余り領域のサイズを例えば5バイトから1~3バイト程度に小さくすることで、実質的に不正プログラムの挿入を困難にしている。また、割込みベクタテーブルのアドレス(本例では0050H)よりも前に、RST命令で呼び出されるモジュールの他、複数のモジュールから読み出されるモジュールを配置することで、プログラムソースのチェックが容易に行えるようになる場合がある。いわゆる共通モジュールをチェックする場合は、0008Hから割込みベクタテーブルのアドレス(本例では0050H)の間をチェックすればよい。

#### 【2349】

次に、上記実施形態8の変形例について図289を用いて説明する。同図(a)は、上記実施形態8の一体化パッケージ10120を示しており、同図(b)、(c)はその変形例を示している。同図(a)に示すように、一体化パッケージ10120は、端子のない中央部の幅x2が端子のある両側の幅x1より狭くなっている。同図(b)に示す一体化パッケージ10120は、端子のない中央部の幅x2が端子のある両側の幅x1より狭くなっている点は同図(a)のものと同じだが、さらに、パッケージ10120a側の角部に切欠き部が形成されており、幅x1、x2と同方向に測った切欠き部近傍を含む幅x3は、幅x1より短く、幅x2より長くなっている。同図(c)に示す一体化パッケージ10120は、パッケージ10120a側の角部に切欠き部が形成されており、幅x1、x2と同方向に測った切欠き部近傍の幅x3は、幅x1より短く、幅x2より長くなっている点は同図(b)のものと同じだが、さらに、パッケージ10120a先端面に凹部が形成されており、幅x1に直交する長手方向の幅に関して、凹部も切欠き部も含まない領域の幅y1に対し、凹部を含む領域の幅y2は幅y1より短くなっている。また、切欠き部近傍を含む幅y3も幅y1より短くなっている。

#### 【2350】

なお、本発明に係る遊技台は、封入式パチンコ機やメダルレススロットマシンに適用することもできる。また、主制御部、第1副制御部、および第2副制御部をワンチップで構成してもよいし、主制御部と第1副制御部で双方向の通信が可能に構成してもよい。また、主制御部と第1副制御部で双方向の通信を可能とする一方で、第1副制御部から第2副制御部への通信は一方方向の通信としてもよい。

#### 【2351】

また、本発明の実施の形態に記載された作用および効果は、本発明から生じる最も好適な作用および効果を列挙したに過ぎず、本発明による作用および効果は、本発明の実施の形態に記載されたものに限定されるものではない。また、実施形態に記載した複数の構成のうち、1つの構成に記載している内容を、他の構成に適用することでより遊技の幅を広げられる場合がある。したがって、例えば、WDTに基づくリセットに関する記載において、WDTタイムアウト信号（WDT起動信号）を指定エリア外走行禁止信号に読み替えて適用してもよい。

#### 【2352】

上記実施形態に基づく本発明は以下のように記述することもできる。

<付記A>

<付記A1>

CPUと、

16ビットの乱数値を少なくとも発生可能な乱数回路と、

前記CPUを少なくとも搭載したマイクロプロセッサと、

取り込み条件の成立があった場合に、前記乱数値を少なくとも取り込み可能な乱数値レジスタと、

を少なくとも備えた遊技台であって、

前記遊技台は、ぱちんこ機またはスロットマシンであり、

前記マイクロプロセッサは、前記乱数回路を少なくとも内蔵するものであり

前記乱数回路は、複数のレジスタによって少なくとも制御されるものであり、

前記乱数値レジスタは、前記複数のレジスタのうちの少なくとも一つのレジスタであり

、  
前記CPUは、乱数値格納条件の成立があった場合に、前記乱数値レジスタに取り込まれている前記乱数値をレジスタペアに少なくとも格納可能なものであり、

前記レジスタペアは、前記CPUの汎用レジスタである第一および第二のレジスタの組み合わせによって少なくとも構成されるものであり、

前記CPUは、16ビットロード命令を少なくとも受け付け可能なものであり、

前記乱数値格納条件は、前記CPUが前記16ビットロード命令を受け付けたことを少なくとも含むものである、

ことを特徴とする遊技台。

付記A1に記載の遊技台において、「CPU」は、「CPUコア」を示すものとしてもよい。「乱数回路」が発生する乱数は擬似乱数であってもよい（上記実施形態における8ビット乱数および16ビット乱数は、8ビット擬似乱数、16ビット擬似乱数としてもよい）。

<付記A2>

付記A1に記載の遊技台であって、

前記マイクロプロセッサは、遊技制御プログラムが少なくとも記憶されているROMを少なくとも内蔵するものであり、

前記マイクロプロセッサは、少なくともランダム延長機能を有し、

前記ランダム延長機能は、前記遊技制御プログラムの実行開始タイミングをランダムに変化させることが可能なものである、

ことを特徴とする遊技台。

<付記A3>

付記A2に記載の遊技台であって、

前記マイクロプロセッサは、信号出力端子を少なくとも備えるものであり、

前記信号出力端子からの出力は、少なくとも特定のタイミングで第一のレベルから該第一のレベルよりも高い第二のレベルに変化するものであり、

前記特定のタイミングは、前記ランダム延長機能の実行前である、  
ことを特徴とする遊技台。

<付記 A 4>

付記 A 2 または A 3 に記載の遊技台であって、

少なくとも、セキュリティモードおよびユーザモードを備え、

前記遊技制御プログラムは、前記ユーザモードで実行されるユーザプログラムである、  
ことを特徴とする遊技台。

10

<付記 A 5>

付記 A 2 から A 4 のいずれか一項に記載の遊技台であって、

前記遊技制御プログラムの実行開始は、システムリセットおよびユーザリセットのうちの少なくともいずれか一方を受け付けた後でおこなわれるものである、  
ことを特徴とする遊技台。

【2353】

<付記 B>

<付記 B 1>

乱数値を用いた制御を少なくとも実行可能な遊技制御手段と、

前記乱数値を少なくとも発生可能な第一の乱数手段と、

前記乱数値を少なくとも発生可能な第二の乱数手段と、  
を少なくとも備えた遊技台であって、

20

前記遊技台は、ぱちんこ機またはスロットマシンであり、

前記第一の乱数手段は、第一の数値範囲から第一の乱数値を少なくとも発生可能なものであり、

前記第二の乱数手段は、第二の数値範囲から第二の乱数値を少なくとも発生可能なものであり、

前記第一の乱数手段の更新開始時期は、前記第二の乱数手段の更新開始時期とは異なるものである、

ことを特徴とする遊技台

30

<付記 B 2>

付記 B 1 に記載の遊技台であって、

第一の処理を少なくとも前記遊技制御は含むものであり、

前記第一の乱数手段の更新開始時期は、前記遊技制御手段による前記第一の処理の実行よりも後である、

ことを特徴とする遊技台。

<付記 B 3>

付記 B 2 に記載の遊技台であって、

前記第二の乱数手段の更新開始時期は、前記遊技制御手段による前記第一の処理の実行よりも前である、

40

ことを特徴とする遊技台。

<付記 B 4>

付記 B 2 または B 3 に記載の遊技台であって、

前記第一の処理は、該第一の処理よりも後に実行される第二の処理の実行時期を変化させることが可能なものである、

ことを特徴とする遊技台。

<付記 B 5>

付記 B 1 ～ B 4 の何れかに記載の遊技台であって、

前記第一の乱数手段と前記第二の乱数手段の少なくとも何れかに前記数値範囲の設定を少なくとも指示可能な第三の処理と、

50

前記第一の乱数手段と前記第二の乱数手段の少なくとも何れかによって導出された値を少なくとも取得可能な第四の処理と、

前記第四の処理によって取得された値に基づく判定を少なくとも実行可能な第五の処理と、を少なくとも前記遊技制御は含むものであり、

前記第一の数値範囲は、前記第三の処理によって設定されるものであり、

前記第二の数値範囲は、前記第三の処理によらずに設定されるものであり、

前記第一の数値範囲が設定された場合に、前記第一の乱数手段の更新が開始可能であることを特徴とする遊技台。

なお、本件の「第一の乱数手段」は、ソフトウェアによる乱数カウンタにより構成されていてもよい。また、本件の「第二の乱数手段」は、ソフトウェアによる乱数カウンタにより構成されていてもよい。

10

【2 3 5 4】

<付記 C>

<付記 C 1>

CPUと、

乱数値を少なくとも発生可能な乱数回路と、

前記乱数値を用いた制御を少なくとも実行可能な遊技制御手段と、

前記CPUを少なくとも搭載したマイクロプロセッサと、

を少なくとも備えた遊技台であって、

前記遊技台は、ぱちんこ機またはスロットマシンであり、

20

前記マイクロプロセッサは、前記乱数回路を少なくとも内蔵するものであり

前記乱数回路の第一のチャンネルは、第一の数値範囲から第一の乱数値を少なくとも発生可能なものであり、

前記乱数回路の第二のチャンネルは、第二の数値範囲から第二の乱数値を少なくとも発生可能なものであり、

前記第一のチャンネルの更新開始時期は、前記第二のチャンネルの更新開始時期とは異なるものである、

ことを特徴とする遊技台。

<付記 C 2>

付記 C 1 に記載の遊技台であって、

30

第一の処理を少なくとも前記遊技制御は含むものであり、

前記第一のチャンネルの更新開始時期は、前記遊技制御手段による前記第一の処理の実行よりも後である、

ことを特徴とする遊技台。

<付記 C 3>

付記 C 2 に記載の遊技台であって、

前記第二のチャンネルの更新開始時期は、前記遊技制御手段による前記第一の処理の実行よりも前である、

ことを特徴とする遊技台。

<付記 C 4>

40

付記 C 2 または C 3 に記載の遊技台であって、

前記第一の処理は、該第一の処理よりも後に実行される第二の処理の実行時期を変化させることが可能なものである、

ことを特徴とする遊技台。

<付記 C 5>

付記 C 1 ～ C 4 の何れかに記載の遊技台であって、

前記乱数回路に前記数値範囲の設定を少なくとも指示可能な第三の処理と、

前記乱数回路によって導出された値を少なくとも取得可能な第四の処理と、

前記第四の処理によって取得された値に基づく判定を少なくとも実行可能な第五の処理と、を少なくとも前記遊技制御は含むものであり、

50

前記第一の数値範囲は、前記第三の処理によって設定されるものであり、  
前記第二の数値範囲は、前記第三の処理によらずに設定されるものであり、  
前記第一の数値範囲が設定された場合に、前記第一のチャンネルの更新が開始可能であることを特徴とする遊技台。

【2355】

<付記D>

<付記D1>

ウォッチドッグタイマが少なくとも内蔵されたマイクロプロセッサと、  
低電圧信号出力条件の成立があった場合に、低電圧信号を少なくとも出力可能な電圧監視手段と、

を備えた遊技台であって、

前記低電圧信号出力条件は、監視している電源ラインの電圧が基準電圧よりも低いことを少なくとも含むものであり、

前記遊技台は、ぱちんこ機またはスロットマシンであり、

前記ウォッチドッグタイマは、起動条件の成立があった場合に、少なくとも起動可能なものであり、

前記ウォッチドッグタイマは、前記電圧監視手段による前記低電圧信号の出力が停止された後で、起動されるものである、

ことを特徴とする遊技台。

<付記D2>

付記D1に記載の遊技台であって、

前記マイクロプロセッサは、遊技制御プログラムが少なくとも記憶されているROMを少なくとも備え、

前記マイクロプロセッサは、少なくともランダム延長機能を有し、

前記ランダム延長機能は、前記遊技制御プログラムの実行開始タイミングをランダムに変化させることが可能なものである、

ことを特徴とする遊技台。

<付記D3>

付記D2に記載の遊技台であって、

前記マイクロプロセッサは、信号出力端子を少なくとも備え、

前記信号出力端子からの出力は、少なくとも特定のタイミングで第一のレベルから該第一のレベルよりも高い第二のレベルに変化するものであり、

前記特定のタイミングは、前記ランダム延長機能の実行前である、

ことを特徴とする遊技台。

<付記D4>

付記D2またはD3に記載の遊技台であって、

少なくとも、セキュリティモードおよびユーザモードを備え、

前記遊技制御プログラムは、前記ユーザモードで実行されるユーザプログラムである、  
ことを特徴とする遊技台。

<付記D5>

付記D2からD4のいずれか一項に記載の遊技台であって、

前記遊技制御プログラムの実行開始は、システムリセットおよびユーザリセットのうちの少なくともいずれか一方を受け付けた後でおこなわれるものである、

ことを特徴とする遊技台。

【2356】

<付記E>

<付記E1>

マイクロプロセッサを備えた遊技台であって、

前記遊技台は、ぱちんこ機またはスロットマシンであり、

前記マイクロプロセッサのパッケージは、DIPパッケージであり、

前記パッケージは、第一の領域と第二の領域を少なくとも含んで構成されたものであり

、

前記第二の領域の幅は、前記第一の領域の幅より狭いことを特徴とする遊技台。

<付記 E 2>

付記 E 1 に記載の遊技台であって、

前記第一の領域の幅 (x 1) と前記第二の領域の幅 (x 2) は、前記第一の領域から前記第二の領域に続く方向に直交する方向の幅であること  
を特徴とする遊技台。

<付記 E 3>

付記 E 1 または E 2 に記載の遊技台であって、

前記第一の領域の第一の側面、および該第一の側面に対向する第二の側面の各々に複数の端子が設けられており、  
前記第二の領域の前記第一の側面および前記第二の側面には端子が設けられていないこと

を特徴とする遊技台。

<付記 E 4>

付記 E 1 から E 3 のいずれか一項に記載の遊技台であって、

前記パッケージは、前記第一の領域、前記第二の領域および第三の領域を少なくとも含んで構成されたものであり、

前記第二の領域の幅は、前記第三の領域の幅より狭いことを特徴とする遊技台。

<付記 E 5>

付記 E 4 に記載の遊技台であって、

前記第二の領域は、前記第一の領域および前記第二の領域の間に設けられていることを特徴とする遊技台。

【2 3 5 7】

<付記 F>

<付記 F 1>

遊技制御を少なくとも実行可能な遊技制御手段と、  
を備えた遊技台であって、

前記遊技台は、ぱちんこ機またはスロットマシンであり、

前記遊技制御手段は、第一の数値取得条件の成立があった場合に、前記第一の数値を少なくとも取得可能なものであり、

前記遊技制御手段は、第二の数値取得条件の成立があった場合に、前記第二の数値を少なくとも取得可能なものである、  
ことを特徴とする遊技台。

<付記 F 2>

付記 F 1 に記載の遊技台であって、

前記遊技制御手段は、当否判定条件の成立があった場合に、当否判定処理を少なくとも実行可能なものであり、

前記当否判定処理は、前記第一の数値および前記第二の数値を少なくとも用いて実行されるものである、  
ことを特徴とする遊技台。

<付記 F 3>

付記 F 1 または F 2 に記載の遊技台であって、

クロックを少なくとも生成可能なクロック生成回路と、

前記クロックの入力に応じて乱数値を少なくとも更新可能な乱数値更新回路と、

取り込み条件の成立があった場合に、前記乱数値を少なくとも取り込み可能な乱数値レジスタと、

10

20

30

40

50



を備え、

前記第一の数値および前記第二の数値のうちの少なくとも一方は、前記乱数値レジスタに取り込まれた前記乱数値である、  
ことを特徴とする遊技台。

<付記 F 4>

付記 F 3 に記載の遊技台であって、  
前記乱数値は、16 ビット乱数である、  
ことを特徴とする遊技台。

<付記 F 5>

付記 F 1 から F 4 のいずれか一項に記載の遊技台であって、  
前記遊技制御手段は、遊技制御処理を少なくとも実行可能な主制御マイクロプロセッサを少なくとも含むものである、  
ことを特徴とする遊技台。

10

【2358】

<付記 G>

<付記 G 1>

所定のセンサと、  
制御プログラムデータで示される命令に基づいて複数の遊技制御処理を実行する CPU と、

所定の数値範囲内で数値を更新するとともに、該数値を前記 CPU へ送信可能に構成された乱数回路と、

20

制御プログラムデータを記憶した ROM と、を備え、

前記乱数回路は、

保持指示を受けたことに基づいて更新されている前記数値を保持し、

送信指示を受けたことに基づいて保持されている該数値を送信するものであり、

前記 CPU は、

前記複数の遊技制御処理それぞれをメイン制御および所定の周期ごとに実行される割り込み制御のうちの少なくともいずれか一方の制御において実行し、

前記所定のセンサから出力される信号を監視する処理、該所定のセンサの監視結果の履歴が予め定められた所定のパターンと一致した否かを判定する第一の判定処理、および該第一の判定処理によって該所定のセンサの監視結果の履歴が該所定のパターンと一致したことに基づいて前記乱数回路に対して前記保持指示を行う処理を該割り込み制御において実行し、

30

該乱数回路に対して前記送信指示を行う処理を該メイン制御または該割り込み制御において実行し、

該乱数回路から送信された前記数値を用いた当否判定処理を該メイン制御または該割り込み制御において実行し、

演算処理、該演算処理の結果を用いて所定の条件が成立したか否かを判定する第二の判定処理、該第二の判定処理の判定結果に応じた処理に分岐させる分岐処理、および該分岐処理によって分岐された先の処理である分岐先処理を、少なくとも該割り込み制御において実行し、

40

前記制御プログラムデータで示される単一の命令である特定の命令に基づいて、該演算処理、該第二の判定処理および該分岐処理を実行し、

該特定の命令に基づく処理に要する時間を同一にすることで、該第二の判定処理によって該所定の条件が成立したと判定された場合、および該判定処理によって該所定の条件が成立したと判定されなかった場合のいずれの場合であっても、同一のタイミングで該分岐先処理の実行を開始するものであることを特徴とする遊技台。

<付記 G 2>

所定のセンサと、

制御プログラムデータで示される命令に基づいて複数の遊技制御処理を実行する CPU

50

と

所定の数値範囲内で数値を更新するとともに、該数値を前記CPUへ送信可能に構成された乱数回路と、

前記制御プログラムデータを記憶したROMと、を備え、

前記乱数回路は、

送信指示を受けたことに基づいて更新されている前記数値を送信するものであり、

前記CPUは、

前記複数の遊技制御処理それぞれをメイン制御および所定の周期ごとに実行される割込み制御のうちの少なくともいずれか一方の制御において実行し、

前記所定のセンサから出力される信号を監視する処理、該所定のセンサの監視結果の履歴が予め定められた所定のパターンと一致した否かを判定する第一の判定処理、該第一の判定処理によって該所定のセンサの監視結果の履歴が該所定のパターンと一致したことに基づいて前記乱数回路に対して前記送信指示を行う処理を、該割込み制御において実行し、

該乱数回路から送信された前記数値を用いた当否判定処理を該メイン制御または該割込み制御において実行し、

演算処理、該演算処理の結果を用いて所定の条件が成立したか否かを判定する第二の判定処理、該第二の判定処理の判定結果に応じた処理に分岐させる分岐処理、および該分岐処理によって分岐された先の処理である分岐先処理を、少なくとも該割込み制御において実行し、

前記制御プログラムデータで示される単一の命令である特定の命令に基づいて、該演算処理、該第二の判定処理および該分岐処理を実行し、

該特定の命令に基づく処理に要する時間を同一にすることで、該第二の判定処理によって該所定の条件が成立したと判定された場合、および該判定処理によって該所定の条件が成立したと判定されなかった場合のいずれの場合であっても、同一のタイミングで該分岐先処理の実行を開始するものであることを特徴とする遊技台。

#### 【2359】

<付記H>

<付記H1>

複数の遊技制御処理を実行するCPUと、

所定の数値範囲内で数値を更新するとともに、該数値を前記CPUへ送信可能に構成された乱数回路と、を備え、

前記乱数回路は、

前記所定の数値範囲を設定可能に構成され、

更新範囲設定指示を受けたことに基づいて該所定の数値範囲を設定し、

保持指示を受けたことに基づいて更新されている前記数値を保持し、

送信指示を受けたことに基づいて保持されている該数値を送信するものであり、

前記CPUは、

前記複数の遊技制御処理それぞれをメイン制御および所定の周期ごとに実行される割込み制御のうちの少なくともいずれか一方の制御において実行し、

前記乱数回路に対して前記保持指示を行う処理を該割込み制御において実行し、

該乱数回路に対して前記送信指示を行う処理を該メイン制御または該割込み制御において実行し、

該乱数回路から送信された前記数値を用いた当否判定処理を該メイン制御または該割込み制御において実行し、

該複数の遊技制御処理が最初から実行される場合において、該乱数回路に対して前記更新範囲設定指示を行う処理、および該割込み制御を許可する処理それぞれを、該メイン制御において実行し、

該割込み制御が禁止されている状態において該乱数回路に対して該更新範囲設定指示を行う処理を実行し、

該乱数回路に対して該更新範囲設定指示を行う処理を実行した後に該割込み制御を許可する処理を実行するものであることを特徴とする遊技台。

<付記 H 2>

複数の遊技制御処理を実行する CPU と、

所定の数値範囲内で数値を更新するとともに、該数値を前記 CPU へ送信可能に構成された乱数回路と、を備え、

前記乱数回路は、

前記所定の数値範囲を設定可能に構成され、

更新範囲設定指示を受けたことに基づいて該所定の数値範囲を設定し、

送信指示を受けたことに基づいて更新されている前記数値を送信するものであり、

10

前記 CPU は、

前記複数の遊技制御処理それぞれをメイン制御および所定の周期ごとに実行される割込み制御のうちの少なくともいずれか一方の制御において実行し、

前記乱数回路に対して前記送信指示を行う処理を該割込み制御において実行し、

該乱数回路から送信された前記数値を用いた当否判定処理を該メイン制御または該割込み制御において実行し、

該複数の遊技制御処理が最初から実行される場合において、該乱数回路に対して前記更新範囲設定指示を行う処理、および該割込み制御を許可する処理それぞれを、該メイン制御において実行し、

該割込み制御が禁止されている状態において該乱数回路に対して該更新範囲設定指示を行う処理を実行し、

20

該乱数回路に対して該更新範囲設定指示を行う処理を実行した後に該割込み制御を許可する処理を実行するものであることを特徴とする遊技台。

<付記 H 3>

付記 H 1 または H 2 に記載の遊技台であって、

前記複数の遊技制御処理を正常に復帰させるためのリセット回路を備え、

前記リセット回路は、

前記複数の遊技制御処理の進行に関する異常が発生したか否かを判定するとともに、該判定によって該異常が発生したと判定されたことに基づいて復帰指示を前記 CPU に対して行うものであり、

30

前記 CPU は、

前記復帰指示を受けた場合において前記複数の遊技制御処理を最初から実行するものであることを特徴とする遊技台。

<付記 H 4>

付記 H 3 に記載の遊技台であって、

前記 CPU、前記乱数回路および前記リセット回路を少なくとも有するマイクロプロセッサを備え、

前記マイクロプロセッサは、

少なくとも前記リセット回路によって前記異常が発生したと判定されたことに基づいて、セキュリティチェックが行われるセキュリティモードに所定の期間に亘り滞在した後、前記 CPU によって前記複数の遊技制御処理が実行されるユーザモードへ移行するものであることを特徴とする遊技台。

40

【2360】

<付記 I>

<付記 I 1>

複数の遊技制御処理を実行する CPU と、

前記複数の遊技制御処理を正常に復帰させるためのリセット回路と、

遊技に関するデータを記憶する RWM と、を備え、

前記 RWM は、

前記複数の遊技制御処理を正常に復帰させるための復帰データを記憶し、

50

電源ステータスの情報を記憶する所定の記憶領域を有するものであり、  
前記リセット回路は、  
開始指示を受けたことに基づいて経過時間の計測を開始し、  
再開始指示を受けたことに基づいて計測している該経過時間を初期化するとともに、初期化された該経過時間の計測を再び開始し、  
該経過時間が予め定められた所定の時間となったか否かを判定し、  
該経過時間が該所定の時間となったと判定したに基づいて前記CPUに対して復帰指示を行うものであり、  
前記CPUは、  
前記複数の遊技制御処理それぞれをメイン制御および所定の周期ごとに実行される割込み制御のうちの少なくともいずれか一方の制御において実行し、  
電源が遮断される場合において該電源が遮断されることを示す所定のデータを前記所定の記憶領域に記憶させる処理を実行し、  
前記復帰指示を受けた場合には該所定のデータを該所定の記憶領域に記憶させる処理を実行せず、  
該電源が投入された場合または前記復帰指示を受けた場合において該複数の遊技制御処理を最初から実行し、  
該複数の遊技制御処理を最初から実行する場合において該所定の記憶領域に該所定のデータが記憶されているか否かを判定する判定処理を実行し、  
該判定処理によって該所定の記憶領域に該所定のデータが記憶されていると判定された場合には、前記復帰データを用いた復帰処理を実行することが可能であり  
該判定処理によって該所定の記憶領域に該所定のデータが記憶されていないと判定された場合には、該復帰処理を実行することが不可能であり、  
前記再開始指示を行う処理を前記割込み制御において実行し、  
少なくとも該復帰処理を実行した後であり、かつ最初の該割込み制御が実行される前に、前記リセット回路に対して前記開始指示を行う処理を実行するものであることを特徴とする遊技台。

<付記 I 2>  
付記 I 1 に記載の遊技台であって、  
前記CPU、前記リセット回路および前記RWMを少なくとも有するマイクロプロセッサを備え、  
前記マイクロプロセッサは、  
少なくとも前記リセット回路によって前記経過時間が前記所定の時間となったと判定されたに基づいて、セキュリティチェックが行われるセキュリティモードに所定の期間に亘り滞在した後、前記CPUによって前記複数の遊技制御処理が実行されるユーザーモードへ移行するものであることを特徴とする遊技台。

<付記 I 3>  
複数の遊技制御処理を実行するCPUと、  
前記複数の遊技制御処理を正常に復帰させるためのリセット回路と、  
遊技に関するデータを記憶するRWMと、備え、  
前記RWMは、  
前記複数の遊技制御処理を正常に復帰させるための復帰データを記憶し、  
電源ステータスの情報を記憶する所定の記憶領域を有するものであり、  
前記リセット回路は、開始指示を受けたことに基づいて経過時間の計測を開始し、  
再開始指示を受けたことに基づいて計測している該経過時間を初期化するとともに、初期化された該経過時間の計測を再び開始し、  
該経過時間が予め定められた所定の時間となったか否かを判定し、  
該経過時間が該所定の時間となったと判定したに基づいて前記CPUに対して復帰指示を行うものであり、  
前記CPUは、

前記複数の遊技制御処理それぞれをメイン制御および所定の周期ごとに実行される割込み制御のうちの少なくともいずれか一方の制御において実行し、

電源が遮断される場合において該電源が遮断されることを示す所定のデータを前記所定の記憶領域に記憶させる処理を実行し、

前記復帰指示を受けた場合には該所定のデータを該所定の記憶領域に記憶させる処理を実行せず、

該電源が投入された場合または前記復帰指示を受けた場合において該複数の遊技制御処理を最初から実行し、

該複数の遊技制御処理を最初から実行する場合において該所定の記憶領域に該所定のデータが記憶されているか否かを判定する判定処理を実行し、

該判定処理によって該所定の記憶領域に該所定のデータが記憶されていると判定された場合には、前記復帰データを用いた復帰処理を実行することが可能であり、

該判定処理によって該所定の記憶領域に該所定のデータが記憶されていないと判定された場合には、前記復帰データを初期化する初期化処理を実行し、

前記再開始指示を行う処理を前記割込み制御において実行し、

少なくとも該初期化処理を実行した後であり、かつ最初の該割込み制御が実行される前に、前記リセット回路に対して前記開始指示を行う処理を実行するものであることを特徴とする遊技台。

#### 【2361】

<付記J>

<付記J1>

CPUを有し、遊技制御を行う遊技制御手段を備えた遊技台であって、

前記遊技台は、ぱちんこ機またはスロットマシンであり、

前記CPUは、マイクロコンピュータに内蔵され、

前記CPUは、少なくとも特定レジスタを備え、

前記CPUは、前記特定レジスタに値をセットする機能のうち、ロード命令を受けたことに基づいて行われるものとしては、直値により値をセットする機能のみを有することを特徴とする遊技台。

<付記J2>

付記J1に記載の遊技台であって、

前記CPUは、前記特定レジスタに値をセットする機能のうち、ロード命令を受けたことに基づいて行われるものの以外のもので、所定の値を初期値としてセットする機能を少なくとも有することを特徴とする遊技台。

<付記J3>

付記J1またはJ2に記載の遊技台であって、

演出制御を行う演出制御手段を備え、

前記CPUを内蔵した前記マイクロコンピュータは、前記演出制御手段に搭載されず、前記遊技制御手段にのみ搭載されることを特徴とする遊技台。

<付記J4>

付記J1またはJ2に記載の遊技台であって、

演出制御を行う演出制御手段と、

払出制御を行う払出制御手段と、を備え、

前記CPUを内蔵した前記マイクロコンピュータは、前記演出制御手段に搭載されず、前記遊技制御手段および前記払出制御手段のうちの少なくとも一方に搭載されることを特徴とする遊技台。

#### 【2362】

<付記K>

<付記K1>

CPUと、

遊技制御プログラムが記憶されたROMと、

データを一時記憶可能な R A M と、  
を内蔵するマイクロコンピュータを備えた遊技台であって、  
前記遊技台は、ぱちんこ機またはスロットマシンであり、  
前記 C P U は、少なくとも特定のレジスタを有し、  
前記 R A M の先頭アドレスの上位バイトは、前記 R O M の先頭アドレスの上位バイトよりも大きく、

前記 C P U は、前記特定のレジスタに値をセットする機能のうち、ロード命令を受けたことに基づいて行われるものとしては、直値により値をセットする機能のみを有し、

前記 C P U は、前記特定のレジスタに値をセットする機能のうち、前記ロード命令を受けたことに基づいて行われるもの以外のものとしては、前記 R A M の先頭アドレスの上位  
10  
バイトと同じ値を初期値としてセットする機能を少なくとも有する、  
ことを特徴とする遊技台。

<付記 K 2>

付記 K 1 に記載の遊技台であって、

前記マイクロコンピュータは、リセット信号を入力してから、前記 C P U による前記遊  
技制御プログラムの実行が開始されるまでの間に、該遊技制御プログラムの認証を行うセ  
キュリティチェックを少なくとも実行するセキュリティモードに移行可能であり、

該セキュリティモードの時間は、ランダムな時間分の延長が可能である、  
ことを特徴とする遊技台。

<付記 K 3>

20

付記 K 2 に記載の遊技台であって、

前記初期値のセットは、前記 C P U が前記遊技制御プログラムの実行開始前に該遊技制  
御プログラムによらずに行うものであることを特徴とする遊技台。

<付記 K 4>

付記 K 2 または K 3 に記載の遊技台であって、

前記遊技制御プログラムの実行開始は、前記リセット信号の入力によるシステムリセッ  
トおよびユーザーリセットのうちの少なくともいずれか一方を受け付けた後で行われるこ  
とを特徴とする遊技台。

<付記 K 5>

30

付記 K 2 ～ K 4 のいずれかに記載の遊技台であって、

少なくとも、前記セキュリティモードと、ユーザーモードを備え、

前記遊技制御プログラムは、前記ユーザーモードで実行されるユーザープログラムであ  
ることを特徴とする遊技台。

<付記 K 6>

付記 K 1 ～ K 5 のいずれかに記載の遊技台であって、

遊技制御を行う遊技制御手段と、

演出制御を行う演出制御手段と、を備え、

前記マイクロコンピュータは、前記演出制御手段に搭載されず、前記遊技制御手段にの  
み搭載されることを特徴とする遊技台。

<付記 K 7>

40

付記 K 1 ～ K 5 のいずれかに記載の遊技台であって、

遊技制御を行う遊技制御手段と、

演出制御を行う演出制御手段と、

払出制御を行う払出制御手段と、を備え、

前記マイクロコンピュータは、前記演出制御手段に搭載されず、前記遊技制御手段およ  
び前記払出制御手段のうちの少なくとも一方に搭載されることを特徴とする遊技台。

【2 3 6 3】

<付記 L>

<付記 L 1>

複数のアドレスそれぞれで示される記憶領域に制御プログラムデータおよび該制御プロ  
50

グラムデータに基づいて参照される参照データを含む複数種類のデータを記憶したROMと、

前記ROMに記憶された前記制御プログラムデータおよび前記参照データに基づいて所定の周期毎に実行される割込み処理を含む複数種類の遊技制御処理を実行するCPUと、を備え、

前記ROMは、

1又は複数の前記制御プログラムデータであって、前記CPUが実行する複数種類の命令それぞれに対応した命令データと、

1又は複数の前記制御プログラムデータであって、前記CPUが該命令を実行するために必要な補足データと、を記憶し、

前記命令データであって、特定のアドレスを識別可能にする特定識別情報の一部である第1の識別情報を示す第1のアドレスデータおよび該第1のアドレスデータと異なる別データで構成され、該特定のアドレスで示される記憶領域に記憶されたデータを前記CPUに読み込ませる特定命令に対応した特定命令データと、

前記補足データであって、前記特定識別情報から前記第1の識別情報を除いた情報である第2の識別情報を示す第2のアドレスデータで構成され、前記CPUが該特定命令を実行するために必要な特定補足データと、を前記割込み処理で実行される命令に対応するように記憶し、

前記特定命令の実行により前記CPUが読み込み可能な特定の記憶領域に全ての前記制御プログラムデータを記憶し、

全ての制御プログラムデータが記憶された前記特定の記憶領域とは異なる記憶領域に、前記参照データを記憶していることを特徴とする遊技台。

#### 【2364】

<付記M>

<付記M1>

少なくとも第一から第四のレジスタを備え、上位レジスタおよび下位レジスタからなるレジスタペアを用いた処理を実行可能であるCPUと、

遊技制御プログラムを記憶するROMと、を内蔵するマイクロコンピュータを備えた遊技台であって、

前記遊技台は、ぱちんこ機またはスロットマシンであり、

前記CPUは、第一の命令と第二の命令を含む複数の命令を実行可能であり、

少なくとも前記第一のレジスタと前記第二のレジスタの二つのレジスタの組み合わせは、前記レジスタペアのうちの所定のレジスタペアであり、

前記第三のレジスタと前記第四のレジスタの少なくともいずれかは、前記上位レジスタであり、

前記第一の命令が実行されたことに基づいて、前記所定のレジスタペアに格納された値によって示される第一のアドレスが示す領域に格納された第一の値が前記第四のレジスタにセットされ、該第一のアドレスと連続する第二のアドレスが示す領域に格納された第二の値が前記第三のレジスタにセットされ、該所定のレジスタペアの値が該第二のアドレスと連続するアドレスを示す値になり、

前記第二の命令が実行されたことに基づいて、前記所定のレジスタペアに格納された値によって示される第一のアドレスと連続する第二のアドレスが示す領域に格納された第二の値が前記第四のレジスタにセットされ、該第二のアドレスと連続する第三のアドレスが示す領域に格納された第三の値が前記第三のレジスタにセットされ、該所定のレジスタペアの値が該第三のアドレスを示す値になり、

前記CPUは、前記レジスタペアとは異なる前記第三のレジスタと前記第四のレジスタの二つのレジスタの組み合わせに前記ROMに格納された値を一命令でセットすることが可能であるとともに、前記所定のレジスタペアの値を変化させる所定の機能としては、前記第一の命令を受けたことに基づいて値をセットする第一の機能と前記第二の命令を受けたことに基づいて値をセットする第二の機能の二つを少なくとも有し、

前記CPUは、少なくとも特定レジスタを備え、

前記CPUは、前記特定レジスタに値をセットする機能のうち、ロード命令を受けたことに基づいて行われるものとしては、直値により値をセットする機能のみを有する、ことを特徴とする遊技台。

<付記M2>

付記M1に記載の遊技台であって、

前記レジスタペアは、スタック領域に16ビットの値をセットするプッシュ命令において用いられることが可能な二つのレジスタの組み合わせであることを特徴とする遊技台。

<付記M3>

付記M1またはM2に記載の遊技台であって、

10

前記CPUは、前記特定レジスタに値をセットする機能のうち、前記ロード命令を受けたことに基づいて行われるもの以外のものとして、所定の値を初期値としてセットする機能を少なくとも有することを特徴とする遊技台。

<付記M4>

付記M1～M3のいずれかに記載の遊技台であって、

演出制御を行う演出制御手段を備え、

前記CPUを内蔵した前記マイクロコンピュータは、前記演出制御手段に搭載されず、前記遊技制御手段にのみ搭載されることを特徴とする遊技台。

<付記M5>

付記M1～M3のいずれかに記載の遊技台であって、

20

演出制御を行う演出制御手段と、

払出制御を行う払出制御手段と、を備え、

前記CPUを内蔵した前記マイクロコンピュータは、前記演出制御手段に搭載されず、前記遊技制御手段および前記払出制御手段のうちの少なくとも一方に搭載されることを特徴とする遊技台。

【2365】

<付記N>

<付記N1>

8ビット長の値を記憶可能な第一から第三のレジスタを少なくとも備えるCPUを備え、

30

前記CPUは、16ビット長の値を記憶可能な第四のレジスタを少なくとも備え、

前記CPUは、前記第一のレジスタおよび前記第二のレジスタからなるレジスタペアを用いた処理を実行可能であり、

前記CPUを内蔵するマイクロコンピュータを備えた遊技台であって、

前記遊技台は、ぱちんこ機またはスロットマシンであり、

前記CPUは、前記第四のレジスタの値に前記第三のレジスタの値を加算して該第四のレジスタに加算結果をセットすることが可能な第一の命令を実行可能であり、

前記第一の命令は、前記レジスタペアの値に前記第三のレジスタの値を加算して該レジスタペアに加算結果をセットすることが可能であり、

前記CPUは、前記レジスタペアの値から前記第三のレジスタの値を減算して該レジスタペアに減算結果をセットする第二の命令を実行可能であり、

40

前記CPUにおける前記16ビット長の前記レジスタペアに対して前記8ビット長の値を加減算する機能は、前記第一の命令および前記第二の命令の両方によって実現されるが、前記CPUにおける前記16ビット長の前記第四のレジスタに対して前記8ビット長の値を加算する機能は、前記第一の命令によって少なくとも実現され、

前記CPUは、少なくとも特定レジスタを備え、

前記CPUは、前記特定レジスタに値をセットする機能のうち、ロード命令を受けたことに基づいて行われるものとしては、直値により値をセットする機能のみを有する、

ことを特徴とする遊技台。

<付記N2>

50



付記 N 1 に記載の遊技台であって、

前記レジスタペアは、スタック領域に 16 ビットの値をセットするプッシュ命令において用いられることが可能な二つのレジスタの組み合わせであることを特徴とする遊技台。

< 付記 N 3 >

付記 N 1 または N 2 に記載の遊技台であって、

前記 CPU は、前記特定レジスタに値をセットする機能のうち、前記ロード命令を受けたことに基づいて行われるもの以外のものとして、所定の値を初期値としてセットする機能を少なくとも有することを特徴とする遊技台。

< 付記 N 4 >

付記 N 1 ～ N 3 のいずれかに記載の遊技台であって、

演出制御を行う演出制御手段を備え、

前記 CPU を内蔵した前記マイクロコンピュータは、前記演出制御手段に搭載されず、前記遊技制御手段にのみ搭載されることを特徴とする遊技台。

< 付記 N 5 >

付記 N 1 ～ N 3 のいずれかに記載の遊技台であって、

演出制御を行う演出制御手段と、

払出制御を行う払出制御手段と、を備え、

前記 CPU を内蔵した前記マイクロコンピュータは、前記演出制御手段に搭載されず、前記遊技制御手段および前記払出制御手段のうちの少なくとも一方に搭載されることを特徴とする遊技台。

【 2 3 6 6 】

< 付記 O >

< 付記 O 1 >

制御プログラムデータを記憶した ROM と、

前記制御プログラムデータで示される命令に基づいて、遊技に関する複数の処理それぞれをメイン制御および所定の周期ごとに実行される割り込み制御の少なくともいずれか一方で実行する CPU と、を備えた遊技台であって、

前記 CPU は、

少なくとも前記割り込み制御において、演算処理、該演算処理の結果を用いて所定条件の成立を判定する判定処理、該判定処理の判定結果に応じた処理に分岐させる分岐処理、該分岐処理によって分岐された先の処理である分岐先処理、の順に処理を実行し、前記制御プログラムデータで示される単一の命令である特定命令に基づいて、前記演算処理、前記判定処理および前記分岐処理を実行し、

前記特定命令に基づく処理に要する時間を同一にすることで、前記判定処理によって前記所定条件が成立したと判定された場合、および前記判定手段によって前記所定条件が成立したと判定されなかった場合のいずれの場合であっても、同一のタイミングで前記分岐先処理の実行を開始する、

ことを特徴とする遊技台。

< 付記 O 2 >

付記 O 1 に記載の遊技台であって、

遊技媒体を検出する検出センサを備え、

前記 CPU は、

前記割り込み制御において、前記検出センサの状態を監視する検出センサ監視処理、該検出センサ監視処理の監視結果に基づく所定の処理、の順に処理を実行する、

ことを特徴とする遊技台。

【 2 3 6 7 】

< 付記 P >

< 付記 P 1 >

遊技制御を行うマイクロコンピュータを備えた遊技台であって、

前記遊技台は、ぱちんこ機またはスロットマシンであり、

10

20

30

40

50

前記マイクロコンピュータは、第一の側面、および該第一の側面に対向する第二の側面の各々に複数の端子が設けられたDIP型のパッケージに收容され、

前記パッケージは、非透明部材からなり、

前記パッケージは、二本の端子が第一の隙間を空けて設けられた第一の部位を有し、

前記パッケージは、二本の端子を前記第一の隙間よりも広い第二の隙間を空けて設け、該第二の隙間に前記端子が設けられていないことによって前記第一の部位よりも特定の領域を視認容易にする第二の部位を有し、

前記特定の領域には、前記パッケージの裏面における第一の領域、該裏面に対向する基板の表面における第二の領域、および前記裏面と前記表面との間の空間における第三の領域のうちの少なくとも一つが含まれ、

10

前記パッケージにおける前記第一の側面および前記第二の側面の少なくとも一方に、前記第二の部位が配置されるとともに、

前記パッケージにおける前記第一の側面の端部または前記第二の側面の端部の一方に、前記端子を設けない切欠き部を形成することによって、前記第一の部位よりも前記特定の領域を視認容易にする第三の部位が配置されている、

ことを特徴とする遊技台。

<付記P2>

付記P1に記載の遊技台であって、

前記第一の側面の長手方向中央部または／および前記第二の側面の長手方向中央部に、前記第二の部位が配置されている、

20

ことを特徴とする遊技台。

<付記P3>

付記P1またはP2に記載の遊技台であって、

前記第一の側面と前記第二の側面の両方に、前記第二の部位が配置されている、

ことを特徴とする遊技台。

<付記P4>

付記P3に記載の遊技台であって、

前記第一の側面と前記第二の側面の両方に、前記第二の部位が対向して配置されている、

、

ことを特徴とする遊技台。

30

【2368】

<付記Q>

<付記Q1>

所定の数値範囲内で更新される数値を導出するものであって、該所定の数値範囲を設定可能な乱数生成手段と、

前記所定の数値範囲を設定するための指示を前記乱数生成手段に対して行う数値範囲設定処理、前記乱数生成手段によって導出された数値を取得する数値取得処理、および該数値取得処理によって取得された数値に基づく抽選処理を少なくとも含む、遊技制御処理それぞれを、メイン制御および所定の割込み周期ごとに行われる割込み制御のうちの、少なくともいずれか一方の制御において行う遊技制御手段と、

40

前記遊技制御処理の進行に関する異常を検出する処理を実行するとともに、該異常を検出した場合には、前記遊技制御手段に前記遊技制御処理を最初から行わせて前記遊技制御処理を正常に復帰させるための復帰指示を行う、異常検出手段と、を備え、

前記遊技制御手段は、

前記数値範囲設定処理、および前記遊技制御処理であって前記割込み制御を許可する割込み許可処理を、前記メイン制御において行い、

前記数値取得処理を、前記割込み制御において行い、

前記メイン制御において、前記数値範囲設定処理が行われた後に、前記割込み許可処理を行い、

少なくとも前記復帰指示を受けた場合には、前記割込み制御が禁止されている割込み禁

50

止状態において前記数値範囲設定処理を行うものであることを特徴とする遊技台。

<付記 Q 2>

付記 Q 1 に記載の遊技台であって、

前記乱数生成手段、前記遊技制御手段、および前記異常検出手段を少なくとも有する、マイクロプロセッサを備え、

前記マイクロプロセッサは、

少なくとも前記異常検出手段によって前記異常が検出された場合には、セキュリティチェックが行われるセキュリティモードに所定期間に亘り滞在した後、前記遊技制御手段によって前記遊技制御処理が行われるユーザモードへ移行するものであることを特徴とする遊技台。

10

<付記 Q 3>

付記 Q 1 または Q 2 に記載の遊技台であって、

前記マイクロプロセッサは、

遊技に関する情報を一時的に記憶する R A M と、

電源が投入されたことに基づいて、前記遊技制御手段に前記遊技制御処理を最初から行わせるための起動指示を行う起動指示手段と、を備え、

前記遊技制御手段は、

前記遊技制御処理であり、かつ電源が遮断される場合に実行される処理であって、前記 R A M の所定領域に当該処理が行われたことを示す所定のデータを記憶させる処理を含む電断時処理を行い、

20

前記遊技制御処理を最初から行う場合において、前記所定のデータが記憶されている場合には、前記遊技制御処理であり、かつ前記 R A M の領域であって前記乱数生成手段によって導出された数値を記憶する領域が少なくとも初期化される初期化処理を行うことなく前記遊技制御処理を復帰させることが可能なものである一方、前記遊技制御処理を最初から行う場合において、前記所定のデータが記憶されていない場合には、該初期化処理を行うことなく前記遊技制御処理を復帰させることが不可能なものであり、

前記復帰指示を受けた場合には、前記電断時処理を行わないものであることを特徴とする遊技台。

【 2 3 6 9 】

<付記 R>

30

<付記 R 1>

C P U と、

遊技制御プログラムを少なくとも記憶する R O M と、

を少なくとも内蔵するマイクロコンピュータを備えた遊技台であって、

前記遊技台は、ぱちんこ機またはスロットマシンであり、

前記 C P U は、所定のフラグを少なくとも有するフラグレジスタを少なくとも有し、

前記 C P U は、所定の命令を少なくとも実行可能であり、

前記遊技制御プログラムは、メイン処理と、タイマ割込によって起動される割込処理と、該割込処理の実行中に自身以外の処理から少なくとも呼び出される複数のサブ処理を少なくとも含み、

40

前記複数のサブ処理は、第一のサブ処理と第二のサブ処理を少なくとも含み、

前記所定の命令は、前記第二のサブ処理で少なくとも実行され、

前記所定の命令は、所定の移行条件が成立した場合には、該命令を実行した前記第二のサブ処理から前記第一のサブ処理に移行させることが可能で、かつ、所定の変化条件が成立した場合には、該命令の実行に基づいて前記所定のフラグをセットまたはクリアすることが少なくとも可能な命令であり、

前記 C P U は、前記所定の命令の実行後の前記第一のサブ処理または前記第二のサブ処理において、前記所定の命令の実行に基づいてセットまたはクリアされた前記所定のフラグを参照しない処理と、前記所定の命令以外の命令の実行に基づいてセットまたはクリアされた前記所定のフラグを参照する処理を実行する、

50

ことを特徴とする遊技台。

<付記 R 2>

付記 R 1 に記載の遊技台であって、

前記 C P U は、前記フラグレジスタと所定のレジスタを少なくとも有し、

前記所定の移行条件が成立した場合とは、前記所定のレジスタが所定の条件を満たす場合である、

ことを特徴とする遊技台。

<付記 R 3>

付記 R 2 に記載の遊技台であって、

前記所定の命令以外の命令には、算術論理演算命令が少なくとも含まれ、

前記所定の变化条件が成立した場合とは、前記所定のレジスタを用いた前記算術論理演算命令が実行されたときに、該算術論理演算命令の実行後の前記所定のレジスタが前記所定の条件を満たす場合であり、

前記所定の移行条件が成立した場合とは、前記所定のレジスタを用いた算術論理演算命令が実行されたときに、該算術論理演算命令の実行後の前記所定のレジスタが前記所定の条件を満たした結果、前記所定のフラグがセットまたはクリアされた場合である、

ことを特徴とする遊技台。

<付記 R 4>

付記 R 1 ～ R 3 のいずれかに記載の遊技台であって、

前記 C P U は、前記所定の命令の実行に基づいてセットまたはクリアされた前記所定のフラグを参照しないで分岐先を決定する分岐処理と、前記所定の命令以外の命令の実行に基づいてセットまたはクリアされた前記所定のフラグを参照して分岐先を決定する分岐処理を実行する、

ことを特徴とする遊技台。

<付記 R 5>

付記 R 1 ～ R 4 のいずれかに記載の遊技台であって、

前記 C P U は、前記所定の命令の実行に基づいてセットまたはクリアされた前記所定のフラグを参照しない処理を行った後に、前記所定の命令以外の命令の実行に基づいて変化した前記所定のフラグを参照する処理を続けて行う、

ことを特徴とする遊技台。

<付記 R 6>

付記 R 1 ～ R 5 のいずれかに記載の遊技台であって、

前記第二のサブ処理は、前記第一のサブ処理から呼び出される処理である、

ことを特徴とする遊技台。

<付記 R 7>

付記 R 1 ～ R 6 のいずれかに記載の遊技台であって、

前記所定のフラグは、ゼロフラグである、

ことを特徴とする遊技台。

【2370】

<付記 S>

<付記 S 1>

C P U を少なくとも内蔵するマイクロコンピュータを備えた遊技台であって、

前記遊技台は、ぱちんこ機またはスロットマシンであり、

前記 C P U は、第一のジャンプ命令を受け付けたことに基づいて、第一のジャンプ先アドレスに記憶されている命令を受け付けることが可能なものであり、

前記 C P U は、第二のジャンプ命令を受け付けたことに基づいて、第二のジャンプ先アドレスに記憶されている命令を受け付けることが可能なものであり、

前記第一のジャンプ命令には、3 ビットの第一の識別情報が少なくとも含まれており、

前記第二のジャンプ命令には、3 ビットの第二の識別情報が少なくとも含まれており、

前記第一のジャンプ先アドレスは、第一の値により示されるアドレスであり、

前記第二のジャンプ先アドレスは、第二の値により示されるアドレスであり、  
前記第一の値は、前記第一の識別情報の 8 倍の値であり、  
前記第二の値は、前記第二の識別情報の 8 倍の値とは異なり、3 ビットで示すことが可能な最大値を 1 だけ超えた値の 8 倍の値である、  
ことを特徴とする遊技台。

< 付記 S 2 >

付記 S 1 に記載の遊技台であって、  
前記第二の識別情報は 0 である、  
ことを特徴とする遊技台。

< 付記 S 3 >

付記 S 1 または S 2 に記載の遊技台であって、  
前記第一の識別情報は 1 から 7 のいずれかの値である、  
ことを特徴とする遊技台。

< 付記 S 4 >

付記 S 3 に記載の遊技台であって、  
前記第二のジャンプ先アドレスは、前記第一の識別情報の最大値である 7 を 1 だけ超えた値である 8 の 8 倍の値である 64 である、  
ことを特徴とする遊技台。

< 付記 S 5 >

付記 S 1 乃至 S 4 のいずれかに記載の遊技台であって、  
前記 CPU は、前記第一のジャンプ命令および前記第二のジャンプ命令とは異なるジャンプ命令を受け付けることも可能なものである、  
ことを特徴とする遊技台。

【2371】

< 付記 T >

< 付記 T 1 >

遊技に関する複数種類の遊技制御処理それぞれを、メイン制御および所定の割込み周期ごとに行われる割込み制御のうちの、少なくともいずれか一方の制御において実行する遊技制御手段と、

遊技を復帰させるための復帰情報を含む遊技に関する情報を記憶する記憶手段と、  
開始指示を受けたことに基づいて経過時間の計測を開始し、該経過時間が特定時間を超えたか否かを判定するとともに、該経過時間が特定時間を超えたと判定したことに基づいて前記遊技制御手段に対して復帰指示を行い、前記遊技制御手段から初期化指示を受けたことに基づいて該経過時間を初期化する復帰指示手段と、を備えた遊技台であって、

前記記憶手段は、

電源ステータスの情報を記憶する電源ステータス記憶領域を有し、

前記遊技制御手段は、

前記メイン制御において、第一の処理を実行するとともに、該第一の処理の後に、繰り返し実行される第二の処理を実行し、

前記割込み制御において、第三の処理を実行し、

電源が投入された場合、および前記復帰指示手段から前記復帰指示を受けた場合に、前記第一の処理の実行を開始し、

前記第三の処理において、電源が遮断される場合に、電源の遮断に関する所定のデータを前記電源ステータス記憶領域に記憶させる処理を含む電断時処理を実行し、

前記第一の処理において、前記電源ステータス記憶領域に前記所定のデータが記憶されている場合には、前記復帰情報に基づいて前記第二の処理へ復帰させるための復帰処理を実行することが可能であり、

前記第一の処理において、前記電源ステータス記憶領域に前記所定のデータが記憶されていない場合には、前記復帰処理を実行することが不可能なものであり、

この遊技台は、前記復帰指示手段から前記復帰指示を受けた場合には、前記電断時処理

10

20

30

40

50

が行われないことで、前記電源ステータス記憶領域に前記所定のデータが記憶されず、  
前記遊技制御手段は、

前記第一処理において、前記復帰処理を実行した後に、前記開始指示を前記復帰指示手段に対して行う処理を実行するものであることを特徴とする遊技台。

<付記 T 2>

付記 T 1 に記載の遊技台であって、

前記遊技制御手段、前記記憶手段、および前記復帰指示手段を、少なくとも有する、マイクロプロセッサを備え、

前記マイクロプロセッサは、

少なくとも、前記復帰指示手段によって、前記経過時間が前記特定時間を超えたと判定された場合には、セキュリティチェックが行われるとともに、該セキュリティチェックの後に前記遊技制御が行われるユーザモードへ移行するまでの時間をランダムに変動させるランダム延長が少なくとも行われる、セキュリティモードに滞在した後、該ユーザモードへ移行するものであることを特徴とする遊技台。

10

【産業上の利用可能性】

【2372】

本発明の遊技台は、回胴遊技機（スロットマシン）や弾球遊技機（パチンコ等）に代表される遊技台の分野で特に利用することができる。

【符号の説明】

【2373】

20

- 100 パチンコ機
- 208 装飾図柄表示装置
- 208d 演出表示領域
- 212 第1特図表示装置
- 214 第2特図表示装置
- 230 第1特図始動口
- 232 第2特図始動口
- 2321 羽根部材
- 234 可変入賞口
- 2341 扉部材
- 300 主制御部
- 304 CPU
- 306 ROM
- 308 RAM
- 310 I/O
- 311 タイマ回路
- 312 カウンタ回路
- 314 リセット制御回路
- 318 乱数生成回路
- 400 第1副制御部
- 404 CPU
- 406 ROM
- 408 RAM
- 500 第2副制御部
- 600 払出制御部
- 1100 スロットマシン
- 1300 主制御部
- 1304 CPU
- 1306 ROM
- 1308 RAM

30

40

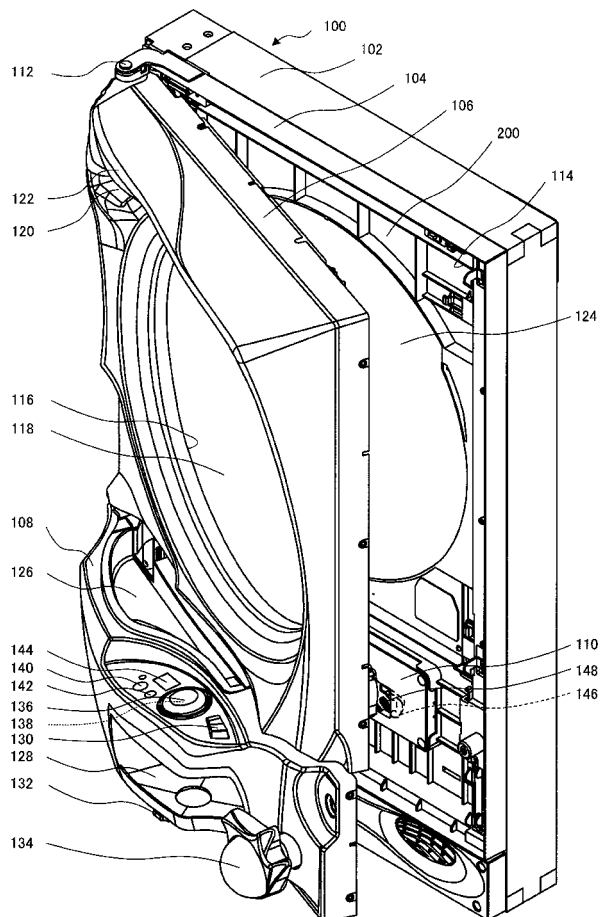
50

- 1 3 1 0 I / O
- 1 4 0 0 第 1 副制御部
- 1 5 0 0 第 2 制御部
- 5 1 0 0 パチンコ機
- 5 1 0 2 遊技盤
- 5 1 0 4 遊技領域
- 5 1 1 0 装飾図柄表示装置
- 5 1 1 2 普図表示装置
- 5 1 1 4 特図表示装置
- 5 1 2 2 一般入賞口
- 5 1 2 4 普図始動口
- 5 1 2 6 第 1 特図始動口
- 5 1 2 8 第 2 特図始動口
- 5 1 3 0 可変入賞口
- 5 3 0 0 主制御部
- 5 3 0 2 基本回路
- 5 4 0 0、5 0 0 副制御部
- 5 6 8 0 パッケージ
- 5 6 8 0 a 視認容易部
- 5 6 8 0 b 視認非容易部
- 5 8 5 0 I C ソケット

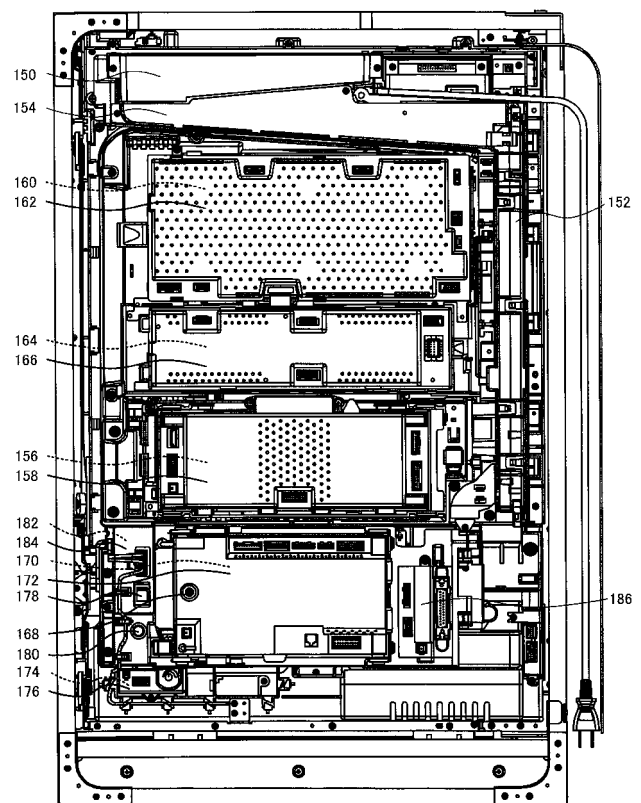
10

20

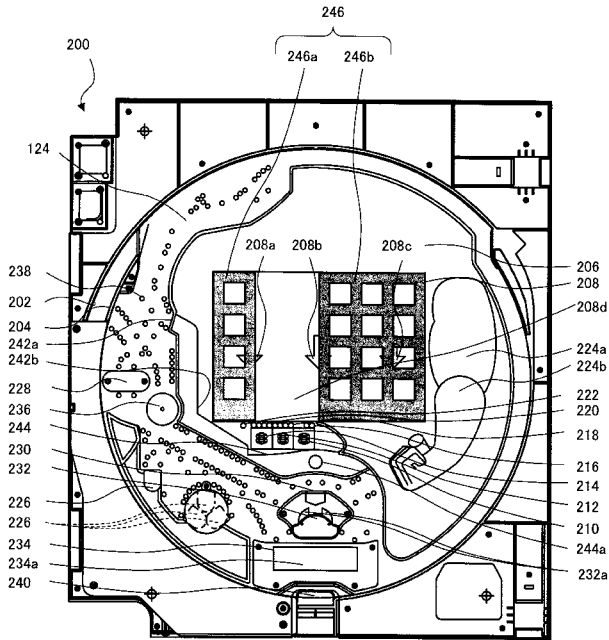
【図 1】



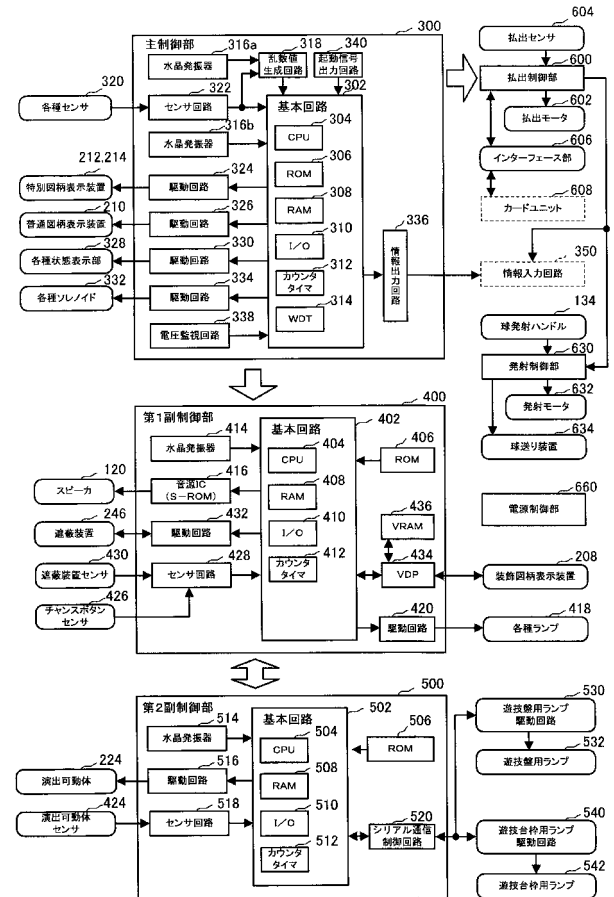
【図 2】



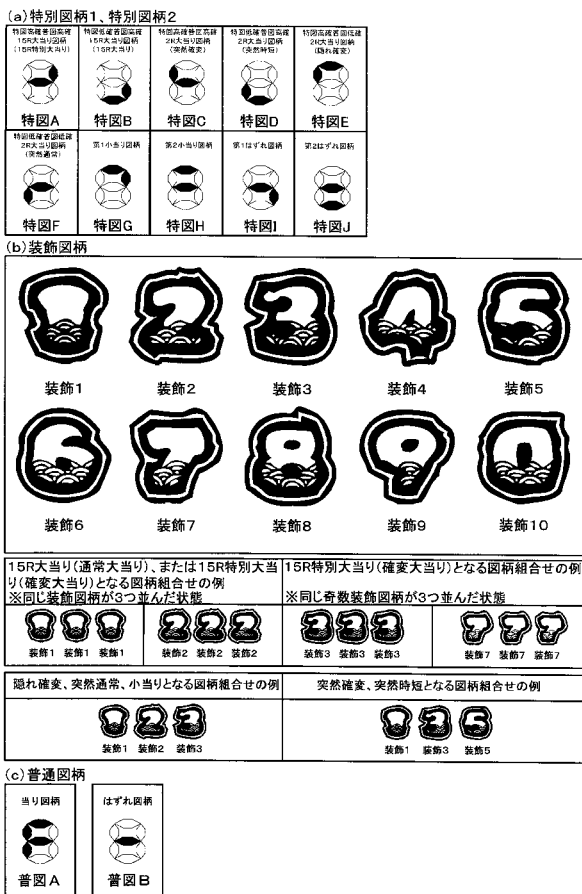
【図 3】



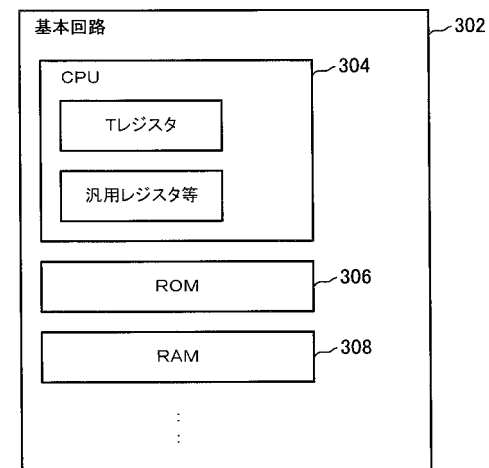
【図 4】



【図 5】

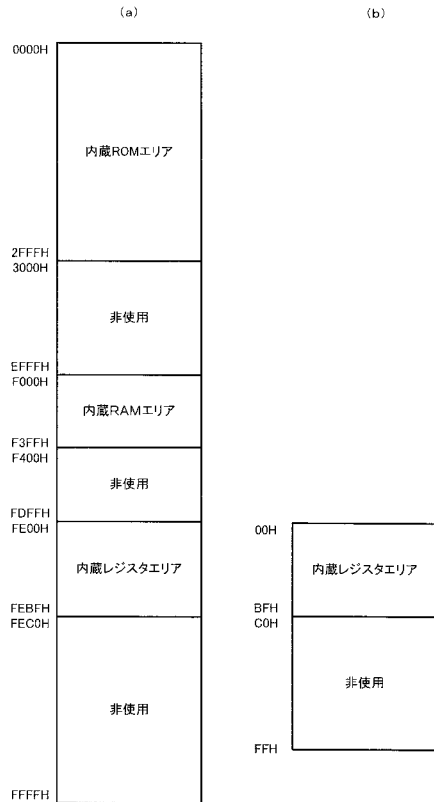


【図 6】

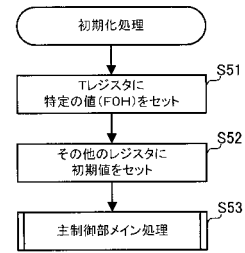




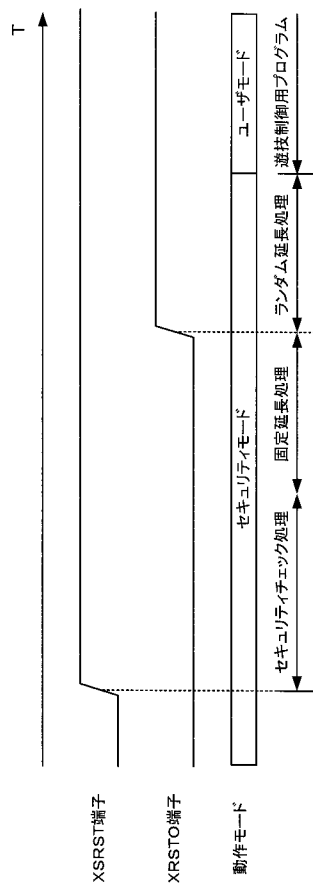
【図 7】



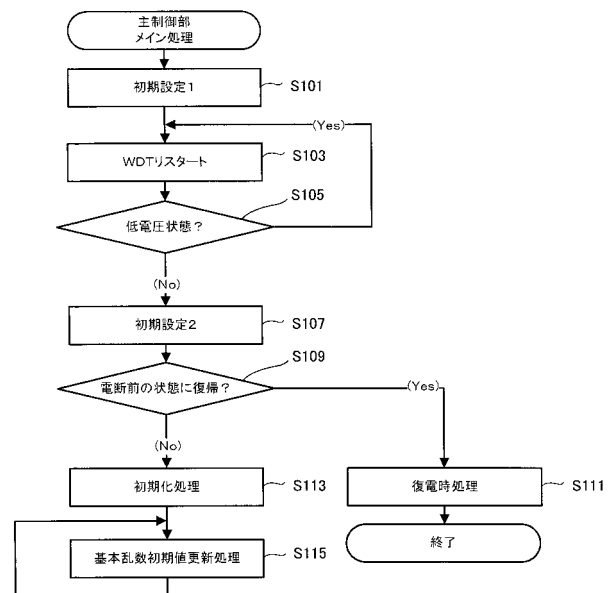
【図 8】



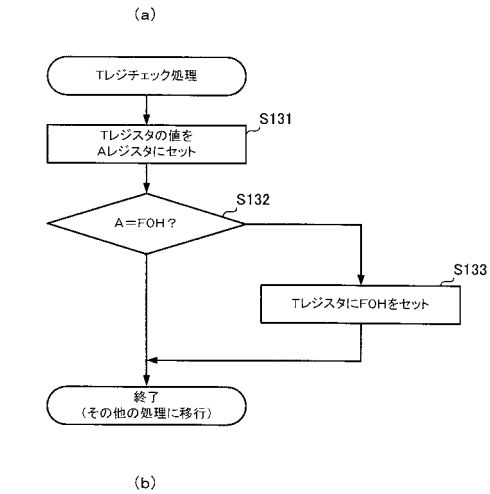
【図 9】



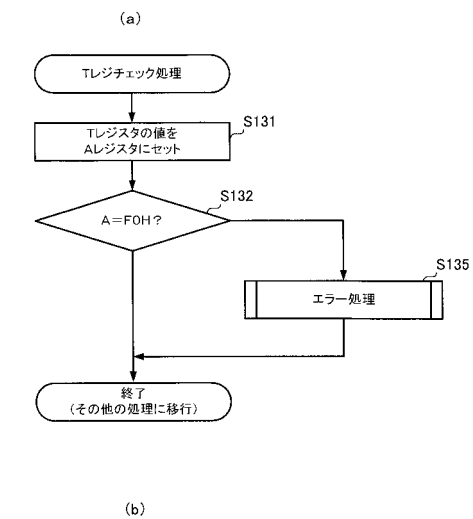
【図 10】



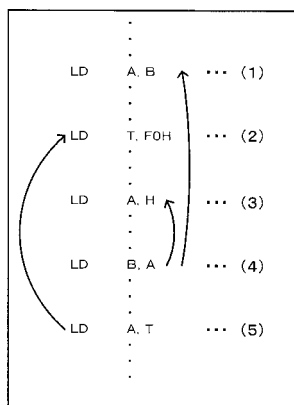
【図 1 1】



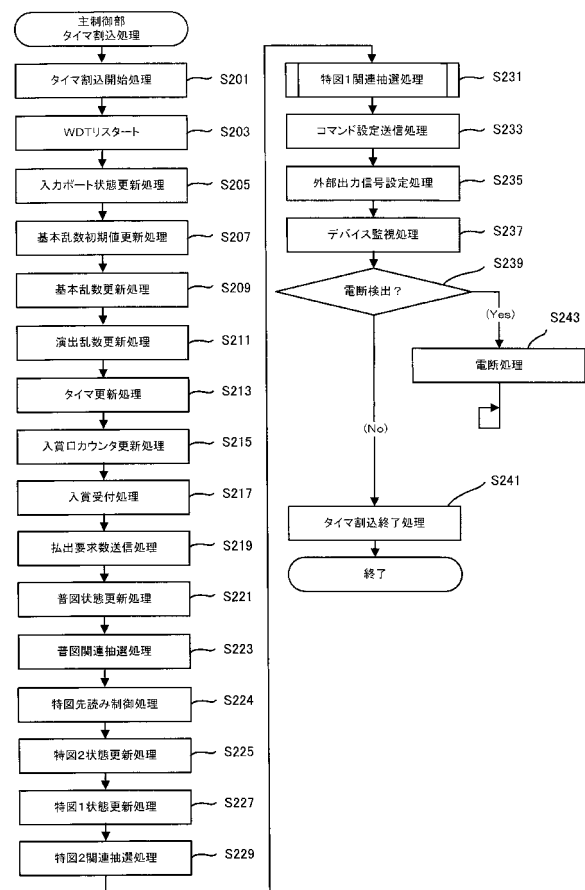
【図 1 2】



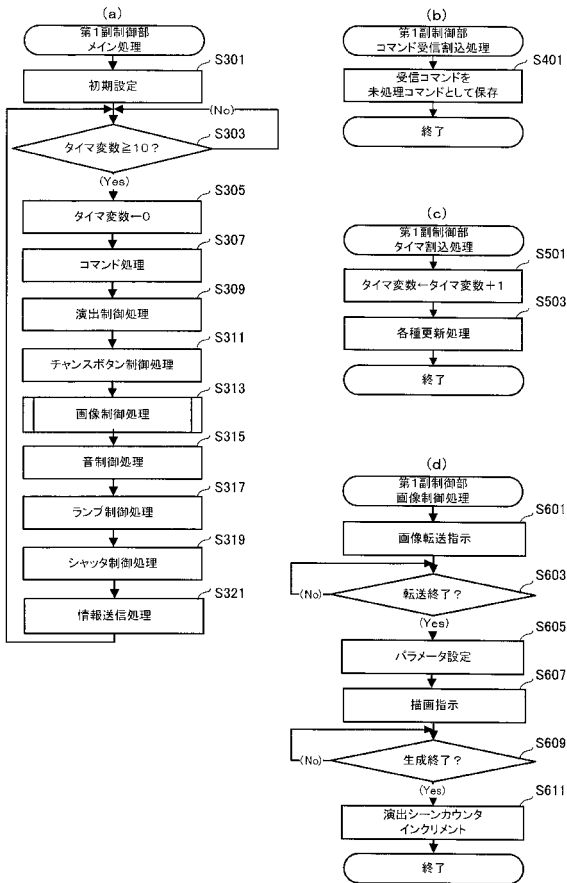
【図 1 3】



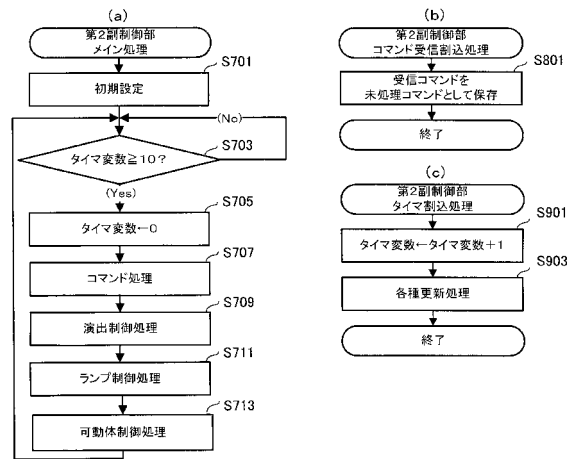
【図 1 4】



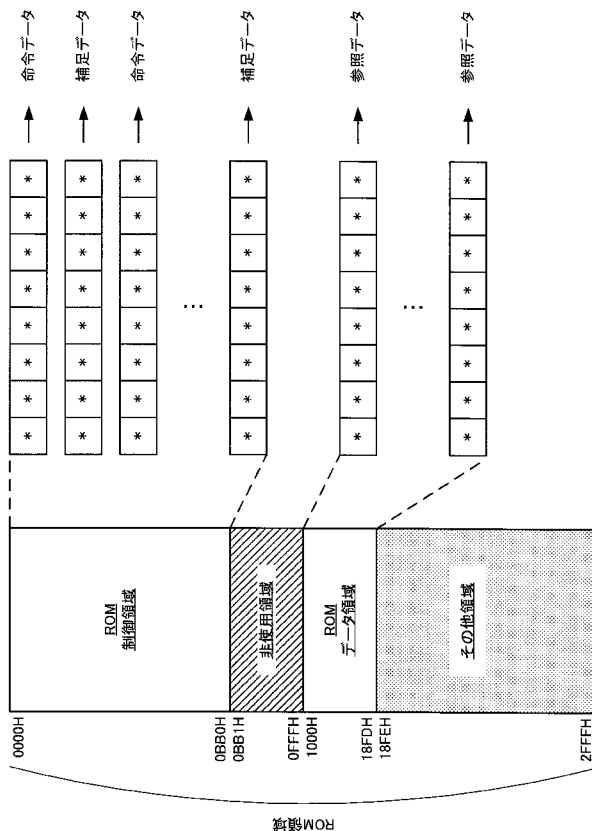
【図 15】



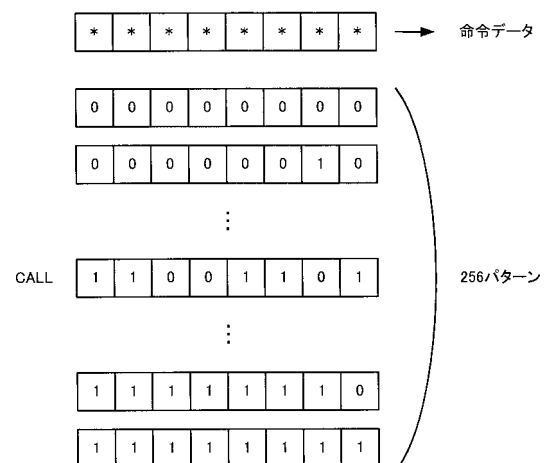
【図 16】



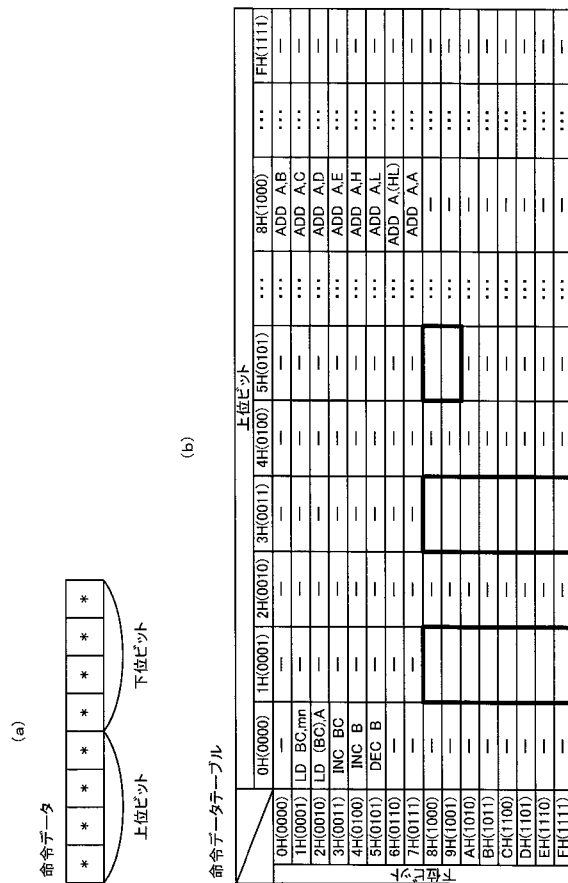
【図 17】



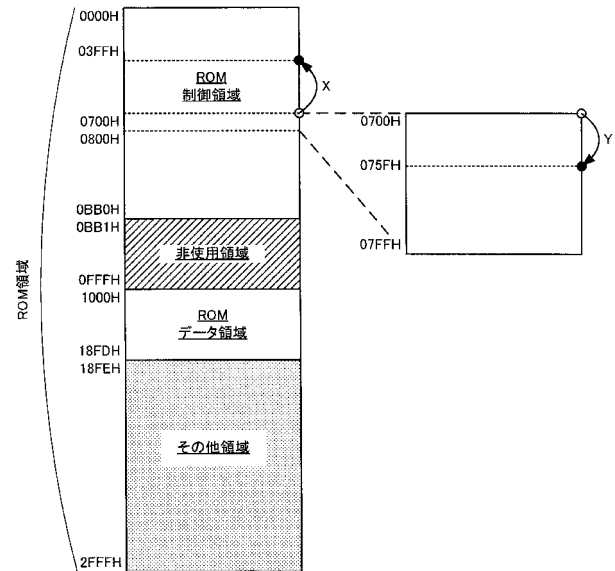
【図 18】



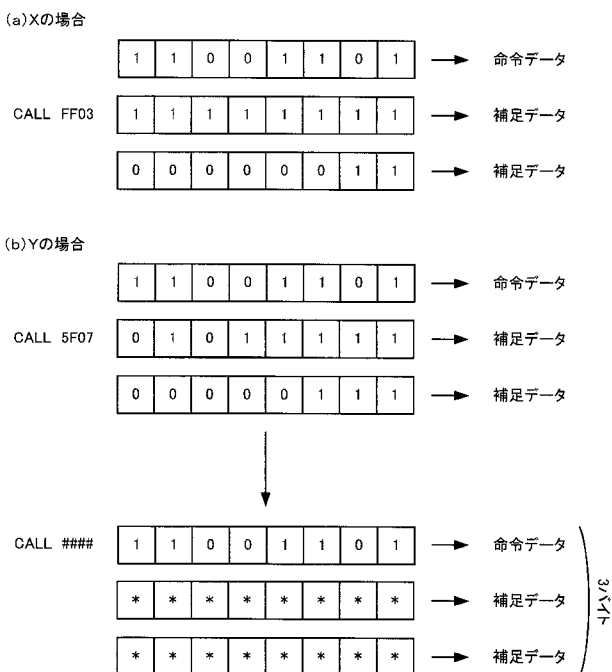
【図 19】



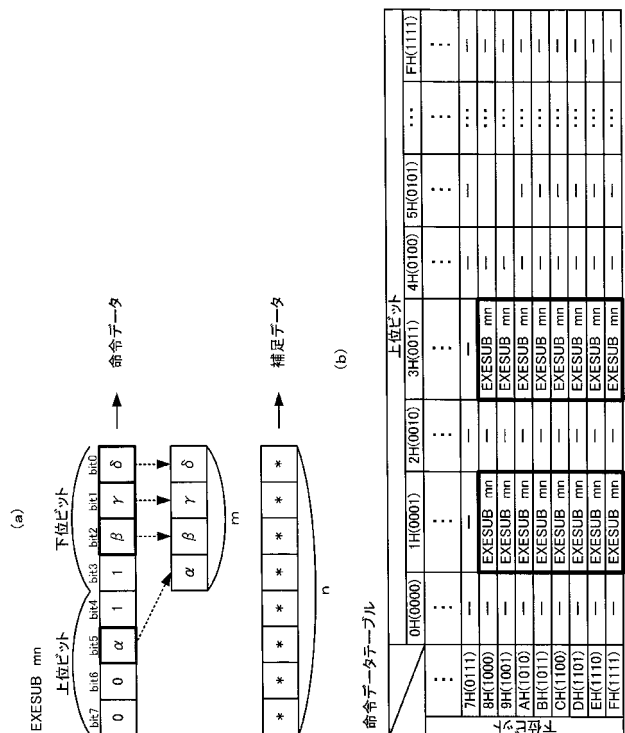
【図 20】



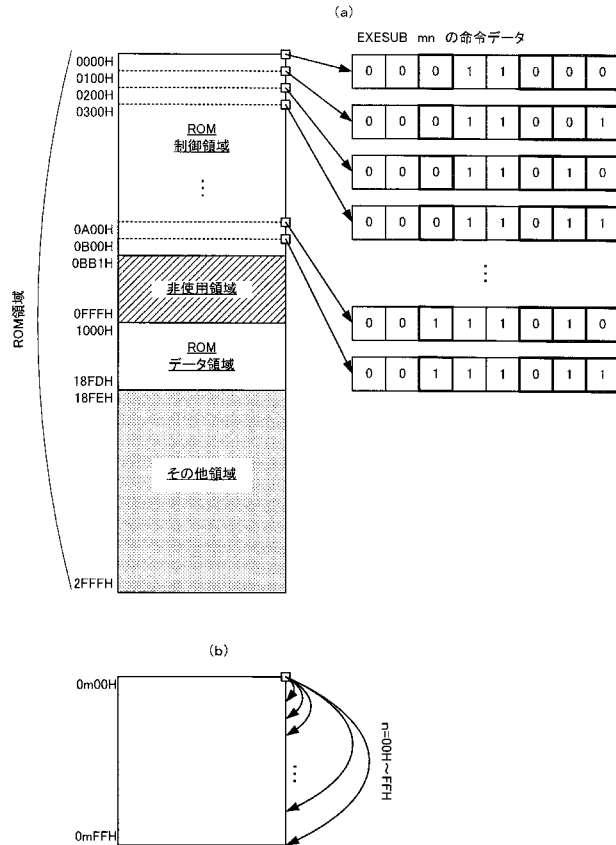
【図 21】



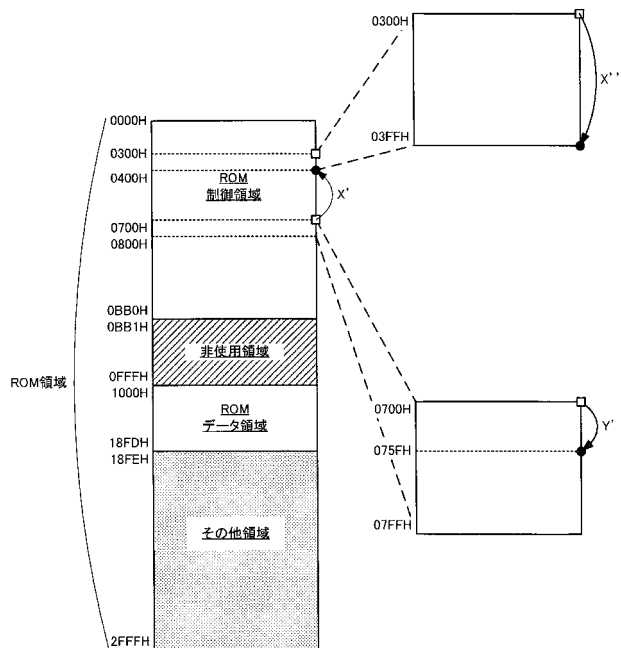
【図 22】



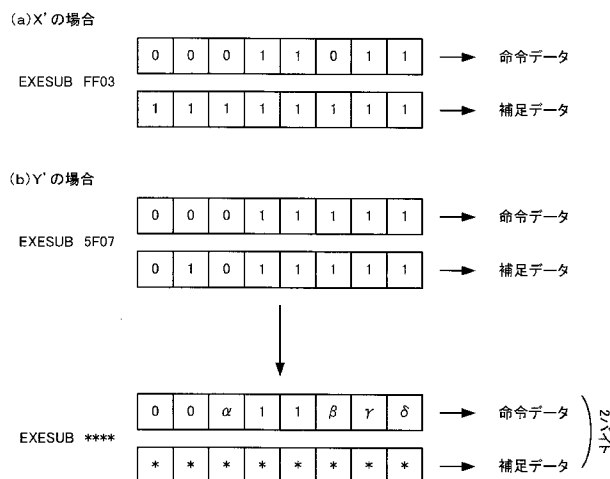
【図 2 3】



【図 2 4】



【図 2 5】



【図 2 6】

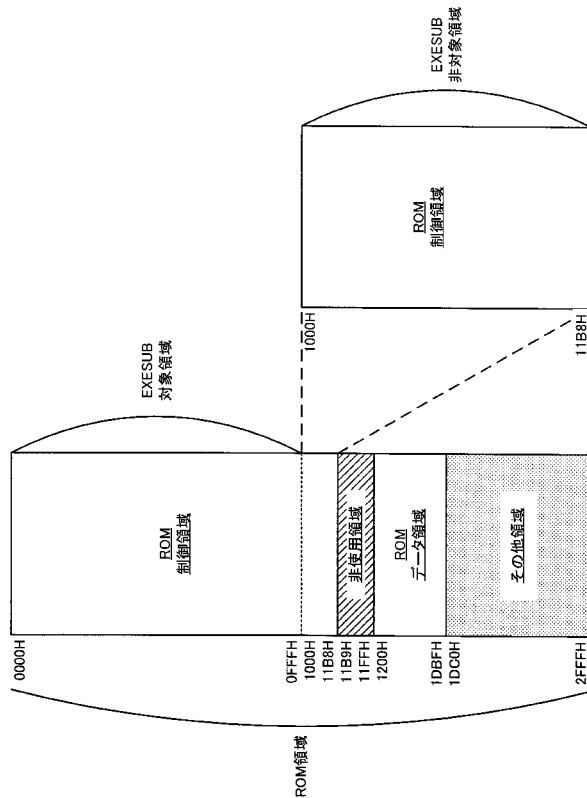
主制御部タイマ割込み処理の制御プログラム

```

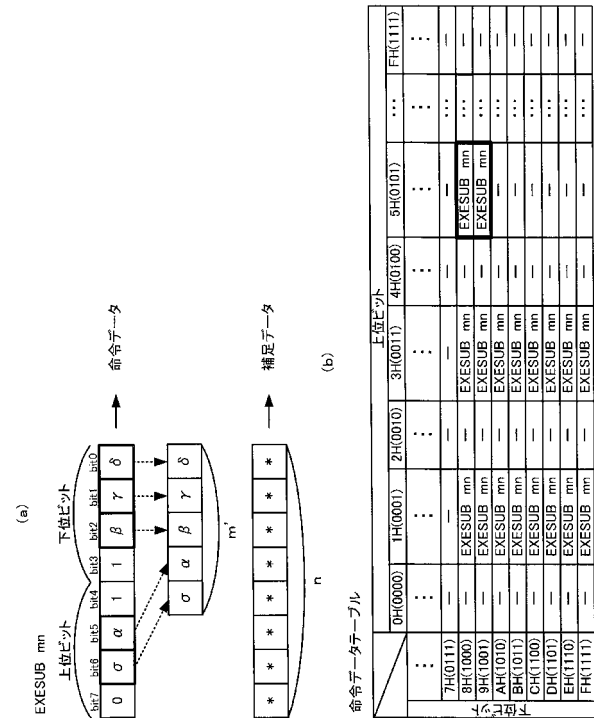
.....タイマ割込開始処理
.....WDTリスタート
.....
0424 EXESUB ** ... (1) 入力ポート状態更新処理
0426 EXESUB ** ... (2) 基本乱数初期値更新処理
0428 EXESUB ** ... (3) 基本乱数更新処理
0430 EXESUB ** ... (4) 演出乱数更新処理
0432 EXESUB ** ... (5) タイマ更新処理
0434 EXESUB ** ... (6) 入賞ロカウンタ更新処理
0436 EXESUB ** ... (7) 入賞受付処理
0438 EXESUB ** ... (8) 払出要求数送信処理
0440 EXESUB ** ... (9) 普図状態更新処理
0442 EXESUB ** ... (10) 普図関連抽選処理
0444 EXESUB ** ... (11) 特図先読み制御処理
0446 EXESUB ** ... (12) 特図2状態更新処理
0448 EXESUB ** ... (13) 特図1状態更新処理
0450 EXESUB ** ... (14) 特図2関連抽選処理
0452 EXESUB ** ... (15) 特図1関連抽選処理
0454 EXESUB ** ... (16) コマンド設定送信処理
0456 EXESUB ** ... (17) 外部出力信号設定処理
0458 EXESUB ** ... (18) デバイス監視処理
.....その他処理

```

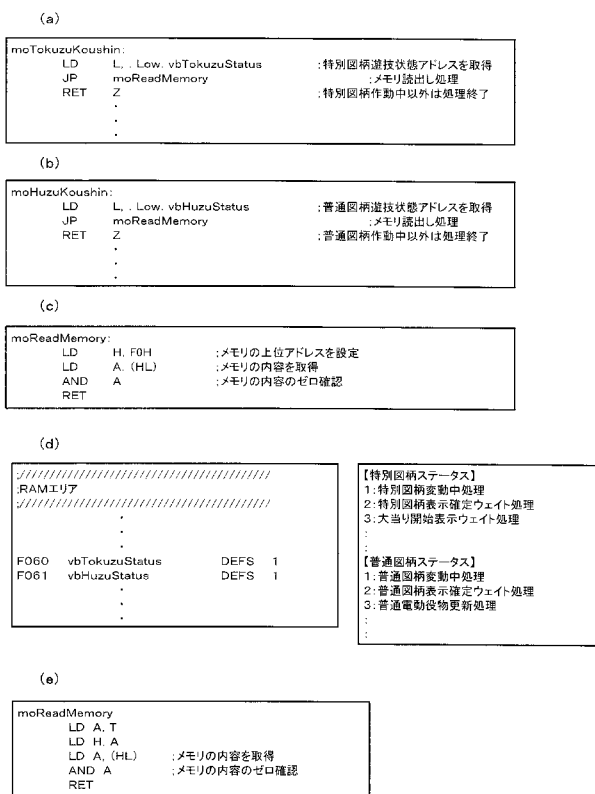
【図 27】



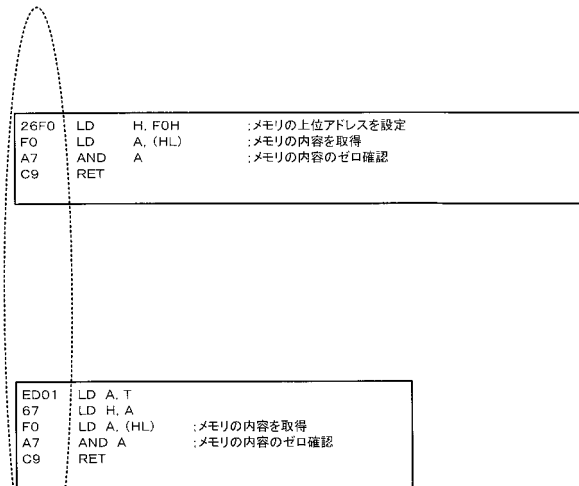
【図 28】



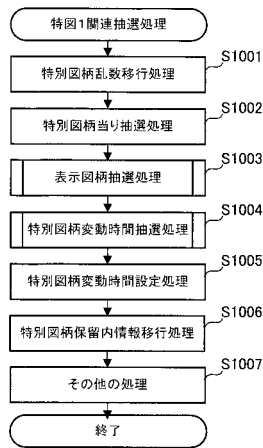
【図 29】



【図 30】



【図 3 1】



【図 3 2】

(a) テーブル選択用テーブル (tableSelectTable)

| 保留  | アドレス  | 飛先アドレス |
|-----|-------|--------|
| 0個用 | 1600H | 174FH  |
| 1個用 | 1602H | 174FH  |
| 2個用 | 1604H | 174FH  |
| 3個用 | 1606H | 1755H  |

(b) 第1変動パターン選択テーブル (timerSelectTable1)

| アドレス  | オフセット | 乱数の総数 |
|-------|-------|-------|
| 1750H | AEH   | 20    |
| 1752H | AEH   | 228   |
| 1754H | B0H   | 8     |
| 1756H | AAH   | 248   |
| 1758H | ACH   | 8     |

(c) 第2変動パターン選択テーブル (timerSelectTable2)

| アドレス  | パターン番号 | 乱数の個数 |
|-------|--------|-------|
| 1800H | 01H    | 255   |
| 1802H | 02H    | 128   |
| 1804H | 03H    | 128   |
| 1806H | 04H    | 128   |
| 1808H | 05H    | 32    |
| 180AH | 06H    | 32    |
| 180CH | 07H    | 32    |
| 180EH | 08H    | 33    |

(d) RAM

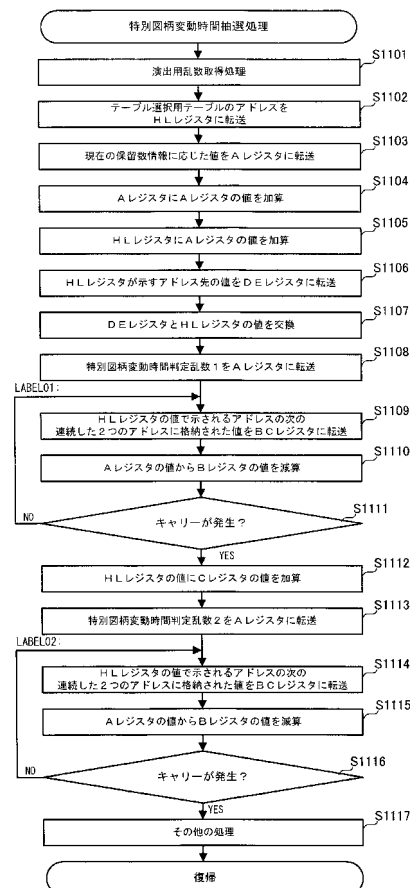
| 領域  | アドレス  | 値   |
|-----|-------|-----|
| 保留数 | F040H | *** |
| 乱数1 | F041H | *** |
| 乱数2 | F042H | *** |

【図 3 3】

```

////////////////////////////////////////////////////
//ROMエリア
////////////////////////////////////////////////////
.
.
.
1600    tableSelectTable:
1600        DEFB    174FH
1602        DEFB    174FH
1604        DEFB    174FH
1606        DEFB    1755H
.
.
.
1750    timerSelectTable1:
1750        DEFB    AEH,    20
1752        DEFB    AEH,    228
1754        DEFB    B0H,    8
1756        DEFB    AAH,    248
1758        DEFB    ACH,    8
.
.
.
1800    timerSelectTable2:
1800        DEFB    01H,    255
1802        DEFB    02H,    128
1804        DEFB    03H,    128
1806        DEFB    04H,    128
1808        DEFB    05H,    32
180A        DEFB    06H,    32
180C        DEFB    07H,    32
180E        DEFB    08H,    33
.
.
.
//RAMエリア
////////////////////////////////////////////////////
.
.
.
F040    vbHoryuuSP1    DEFS    1
F041    vbRndHendou1    DEFS    1
F042    vbRndHendou2    DEFS    1
.
.
.
  
```

【図 3 4】

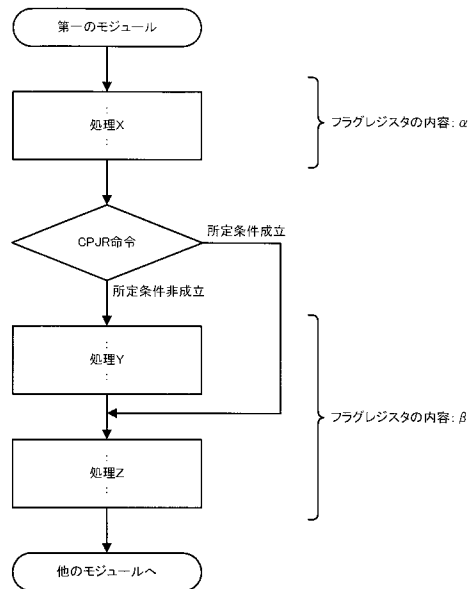




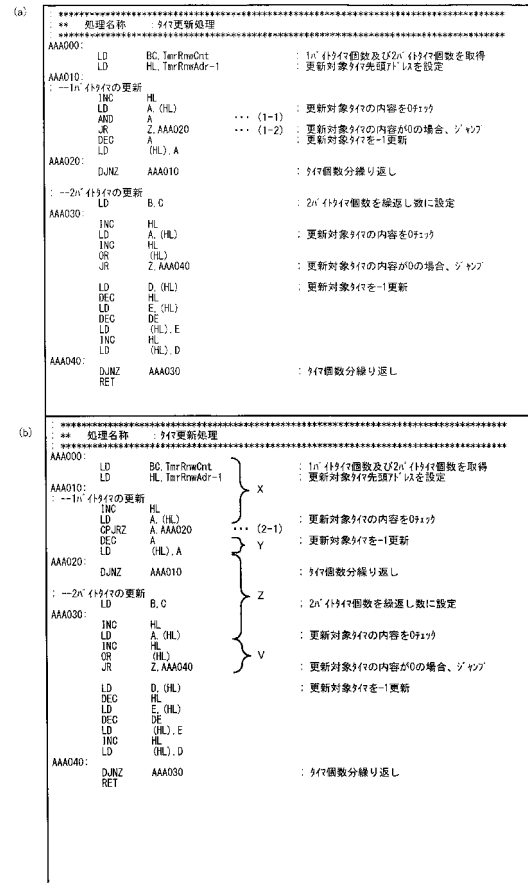




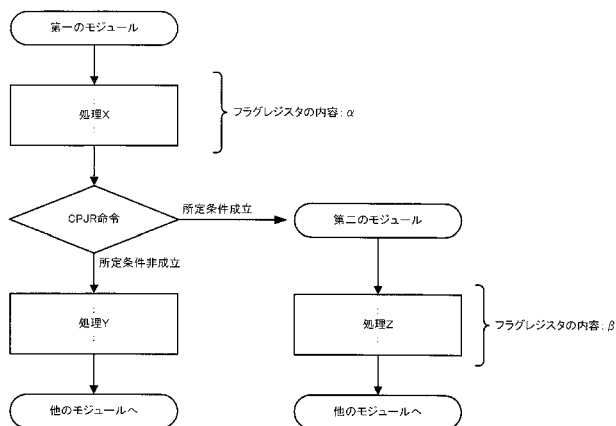
【図 4 3】



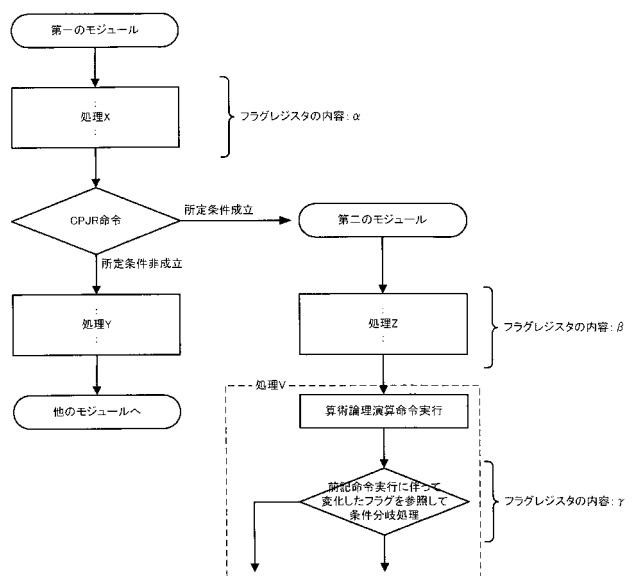
【図 4 4】



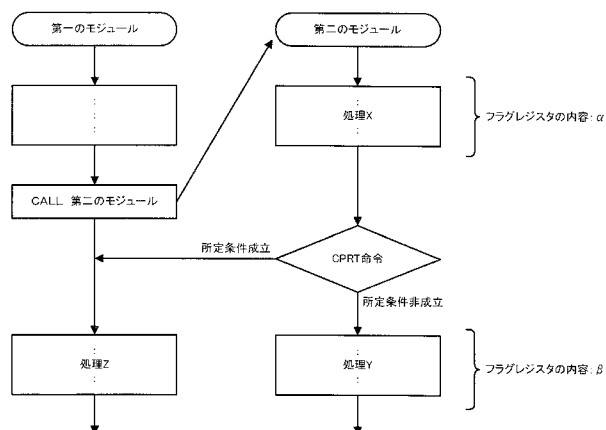
【図 4 5】



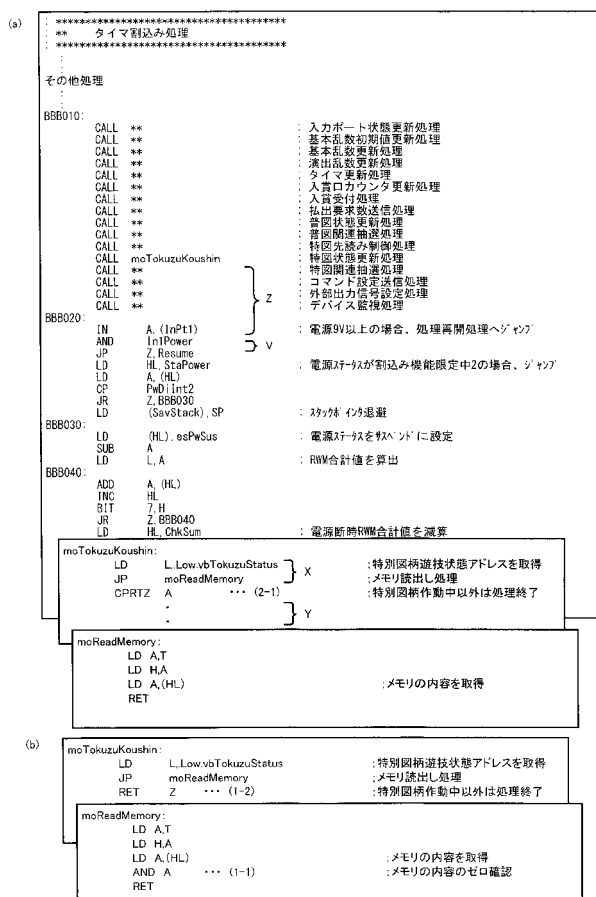
【図 4 6】



【图 4 7】



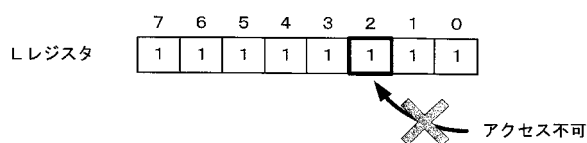
【图 48】



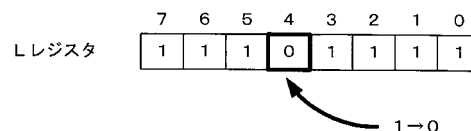
【图 4 9】

|                                                     |                                                              |
|-----------------------------------------------------|--------------------------------------------------------------|
| <p>RES0Z n、r</p> <p>* n=1~7</p> <p>* r=レジスタ</p>     | <p>RES0Z n、(rr)</p> <p>* n=1~7</p> <p>* rr=任意のペアレジスタ</p>     |
| <p>RES1Z n、r</p> <p>* n=0、2~7</p> <p>* r=レジスタ</p>   | <p>RES1Z n、(rr)</p> <p>* n=0、2~7</p> <p>* rr=任意のペアレジスタ</p>   |
| <p>RES2Z n、r</p> <p>* n=0~1、3~7</p> <p>* r=レジスタ</p> | <p>RES2Z n、(rr)</p> <p>* n=0~1、3~7</p> <p>* rr=任意のペアレジスタ</p> |
| <p>⋮</p>                                            | <p>⋮</p>                                                     |
| <p>RES7Z n、r</p> <p>* n=0~6</p> <p>* r=レジスタ</p>     | <p>RES7Z n、(rr)</p> <p>* n=0~6</p> <p>* rr=任意のペアレジスタ</p>     |

【図 50】



(b) [RES2Z 4、L] の場合



【図 5 1】

(a) [RES2Z 2, (HL)] の場合  
(HL=F010H)

|       |   |   |   |   |   |   |   |   |
|-------|---|---|---|---|---|---|---|---|
|       | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Hレジスタ | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 |

|       |   |   |   |   |   |   |   |   |
|-------|---|---|---|---|---|---|---|---|
|       | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Lレジスタ | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |

|       |   |
|-------|---|
| F000H | : |
| :     | : |
| F010H | : |
| F011H | : |
| :     | : |
| F3FFH | : |

|       |   |   |   |   |   |   |   |   |
|-------|---|---|---|---|---|---|---|---|
|       | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| F010H | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

アクセス不可

(b) [RES2Z 4, (HL)] の場合  
(HL=F010H)

|       |   |   |   |   |   |   |   |   |
|-------|---|---|---|---|---|---|---|---|
|       | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| F010H | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 |

1→0

【図 5 3】

RES1Z n, r  
\* n=1~7  
\* r=レジスタ

RES1Z n, (rr)  
\* n=1~7  
\* rr=任意のペアレジスタ

RES2Z n, r  
\* n=0, 2~7  
\* r=レジスタ

RES2Z n, (rr)  
\* n=0, 2~7  
\* rr=任意のペアレジスタ

RES3Z n, r  
\* n=2~7  
\* r=レジスタ

RES3Z n, (rr)  
\* n=2~7  
\* rr=任意のペアレジスタ

RES4Z n, r  
\* n=0~1, 3~7  
\* r=レジスタ

RES4Z n, (rr)  
\* n=0~1, 3~7  
\* rr=任意のペアレジスタ

⋮

⋮

RES254Z n, r  
\* n=0  
\* r=レジスタ

RES254Z n, (rr)  
\* n=0  
\* rr=任意のペアレジスタ

【図 5 2】

| 上8ビット         | 4 (0100)      | 3 (0011)      | 2 (0010)      | 1 (0001)      | 下8ビット         |
|---------------|---------------|---------------|---------------|---------------|---------------|
| RES2Z 6, L    | RES2Z 4, L    | RES2Z 5, L    | RES2Z 3, L    | RES2Z 1, L    | RES2Z 0, L    |
| RES2Z 6, (HL) | RES2Z 4, (HL) | RES2Z 5, (HL) | RES2Z 3, (HL) | RES2Z 1, (HL) | RES2Z 0, (HL) |
| RES2Z 7, L    | RES2Z 6, L    | RES2Z 5, L    | RES2Z 4, L    | RES2Z 3, L    | RES2Z 2, L    |
| RES2Z 7, (HL) | RES2Z 6, (HL) | RES2Z 5, (HL) | RES2Z 4, (HL) | RES2Z 3, (HL) | RES2Z 2, (HL) |

【図 5 4】

(a) [RES2Z 4, L] の場合

RES2Z ...

|  |   |   |   |   |   |   |   |   |
|--|---|---|---|---|---|---|---|---|
|  | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|  | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |

|       |   |   |   |   |   |   |   |   |
|-------|---|---|---|---|---|---|---|---|
|       | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Lレジスタ | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 |

アクセス不可

(b) [RES3Z 4, L] の場合

RES3Z ...

|  |   |   |   |   |   |   |   |   |
|--|---|---|---|---|---|---|---|---|
|  | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|  | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |

|       |   |   |   |   |   |   |   |   |
|-------|---|---|---|---|---|---|---|---|
|       | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Lレジスタ | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 |

アクセス不可



【図 59】

| 上位バイト |          | 1 (0001)  | 2 (0010)  | 3 (0011)  | 4 (0100)  |
|-------|----------|-----------|-----------|-----------|-----------|
| 下位バイト | 5 (0101) | RESZ 0, L | RESZ 8, L | RESZ 4, L | RESZ 6, L |
|       | D (1101) | RESZ 1, L | RESZ 3, L | RESZ 5, L | RESZ 7, L |
|       |          |           |           |           |           |
|       |          |           |           |           |           |

【図 60】

|           |                |
|-----------|----------------|
| (a)       |                |
| RESZ n, r | RESZ n, (rr)   |
| * n=0~7   | * n=0~7        |
| * r=レジスタ  | * rr=任意のペアレジスタ |
| ⋮         | ⋮              |
| (b)       |                |
| RESZ n, r | RESZ n, (rr)   |
| * n=0~7   | * n=0~7        |
| * r=レジスタ  | * rr=任意のペアレジスタ |
| ⋮         | ⋮              |

【図 61】

| 上位バイト |          | 4 (0100)  | 5 (0101)  | 6 (0110)  | 7 (0111)  |
|-------|----------|-----------|-----------|-----------|-----------|
| 下位バイト | 5 (0101) | RESZ 0, L | RESZ 2, L | RESZ 4, L | RESZ 6, L |
|       | D (1101) | RESZ 1, L | RESZ 3, L | RESZ 5, L | RESZ 7, L |
|       |          |           |           |           |           |
|       |          |           |           |           |           |

【図 62】

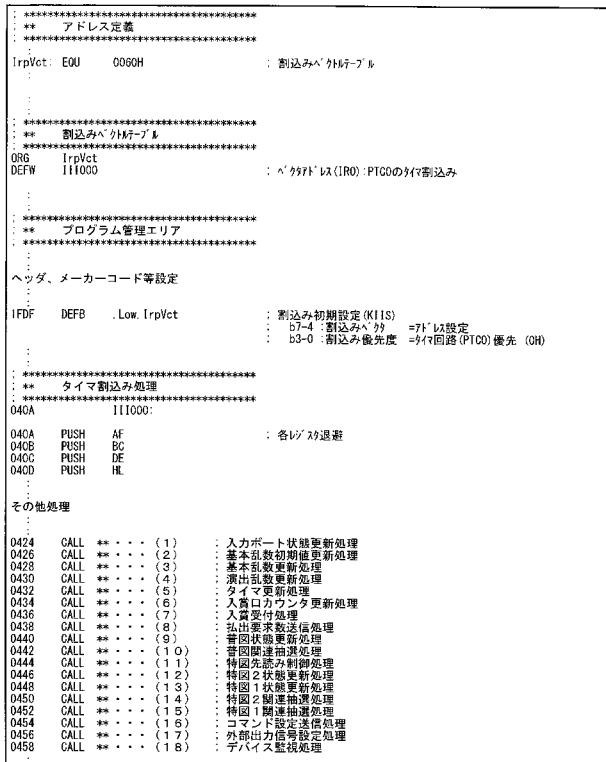
| 上位バイト |          | 1 (0001)  | 2 (0010)  | 3 (0011)  | 4 (0100)  |
|-------|----------|-----------|-----------|-----------|-----------|
| 下位バイト | 5 (0101) | RESZ 0, L | RESZ 8, L | RESZ 4, L | RESZ 6, L |
|       | D (1101) | RESZ 1, L | RESZ 3, L | RESZ 5, L | RESZ 7, L |
|       |          |           |           |           |           |
|       |          |           |           |           |           |

|         |                  |
|---------|------------------|
| (a)     |                  |
| *****   | *****            |
| ** 処理名称 | : プログラム起動設定処理    |
| *****   | *****            |
| 0000    | GEG000:          |
| 0000    | LD SP, Stk       |
| 0003    | LD 2             |
| 0005    | JP HH000         |
|         | : スタックイン設定       |
|         | : 寄込みモードをモード2に設定 |
|         | : 電源投入時処理へ移行     |

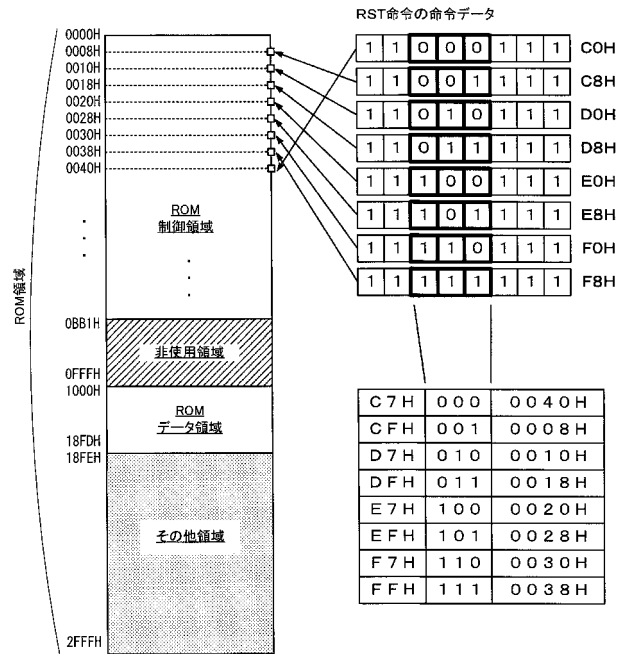
| 上位バイト |          | 1 (0001)  | 2 (0010)  | 3 (0011)  | 4 (0100)  |
|-------|----------|-----------|-----------|-----------|-----------|
| 下位バイト | 5 (0101) | RESZ 8, L | RESZ 2, L | RESZ 4, L | RESZ 6, L |
|       | D (1101) | RESZ 1, L | RESZ 3, L | RESZ 5, L | RESZ 7, L |
|       |          |           |           |           |           |
|       |          |           |           |           |           |

|         |                |
|---------|----------------|
| (b)     |                |
| *****   | *****          |
| ** 処理名称 | : 電源投入時設定処理    |
| *****   | *****          |
| 0062    | HH000:         |
| 0062    | : 電源監視信号確認     |
| 0062    | LD A, 05H      |
| 0062    | OUT (WCL), A   |
| 0064    | RLCA           |
| 0066    | OUT (WCL), A   |
| 0067    |                |
| 0069    | : 電源監視信号確認     |
| 0069    | IN A, (IptCS1) |
| 0069    | LD C, A        |
| 006C    | AND CSIPwrChk  |
| 006E    | JR NZ, HH010   |
|         |                |
|         |                |
|         |                |
|         |                |
|         |                |

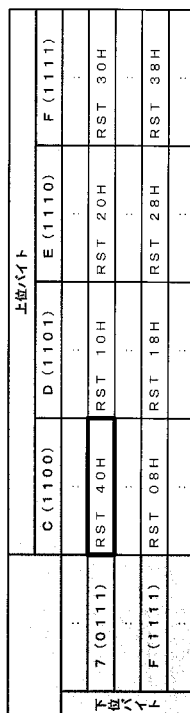
【図 6 3】



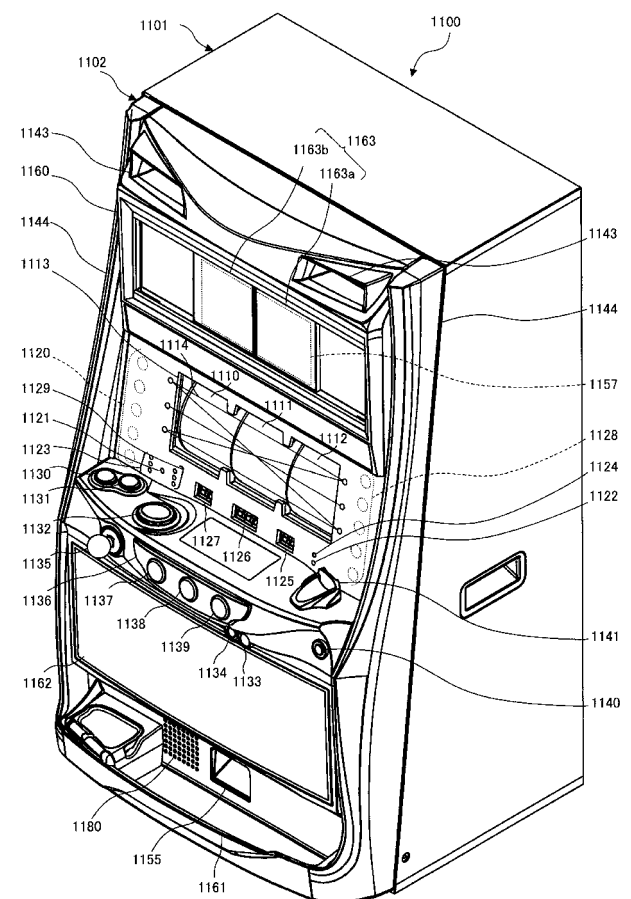
【図 6 4】



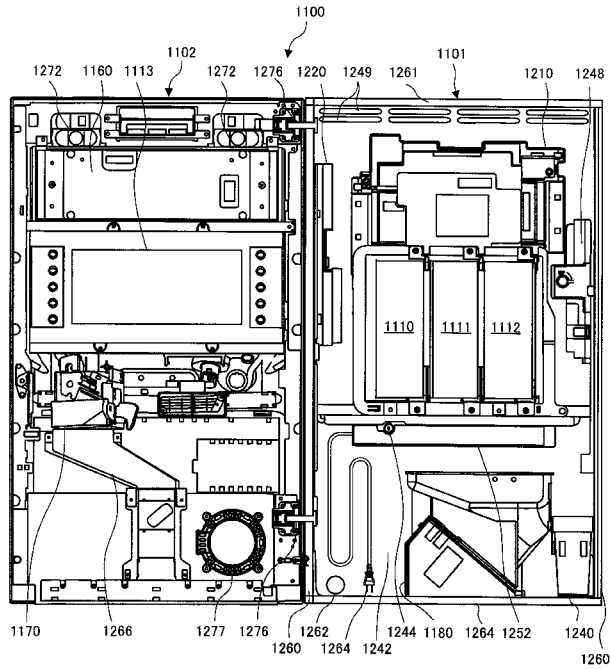
【図 6 5】



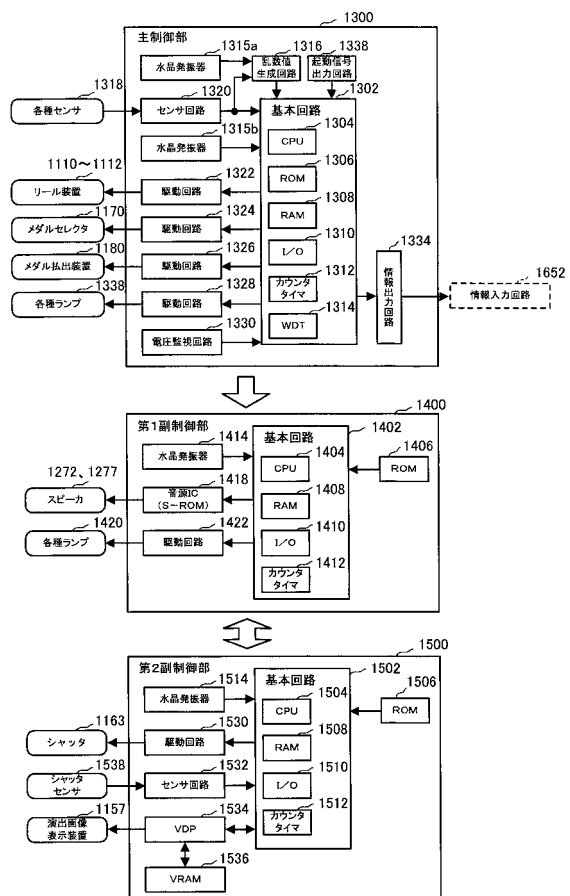
【図 6 6】



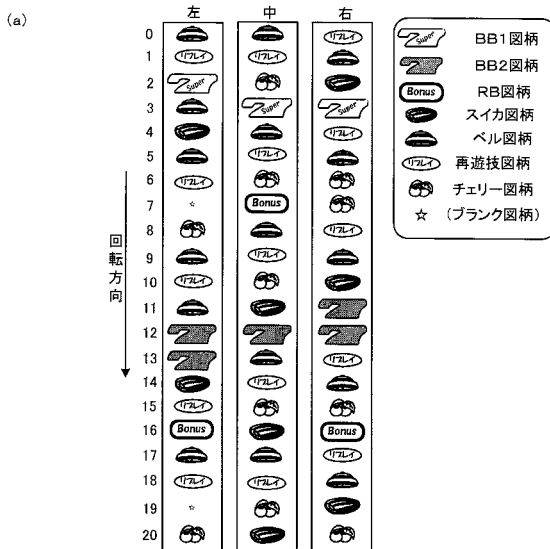
【図 67】



【図 68】



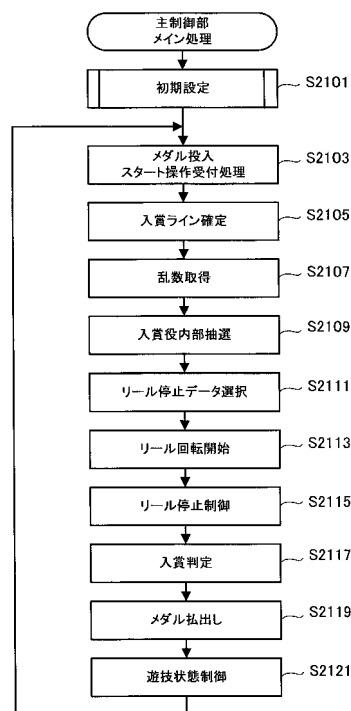
【図 69】



(b)

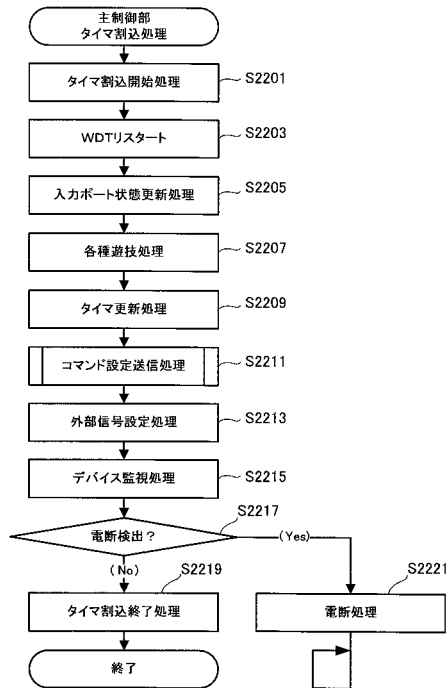
| 入賞役の種類、図柄組合せ、払出/作動 |         |                  |
|--------------------|---------|------------------|
| 入賞役<br>※作動役も含む     | 図柄組合せ   | 払出 (作動)          |
| BB 1               |         | 0枚 / (ビッグボーナス)   |
| BB 2               |         | 0枚 / (ビッグボーナス)   |
| RB                 |         | 0枚 / (レギュラーボーナス) |
| チェリー               | ANY ANY | 4枚               |
| スイカ                |         | 12枚              |
| ベル                 |         | 8枚               |
| 再遊技                |         | (再遊技)            |

【図 70】

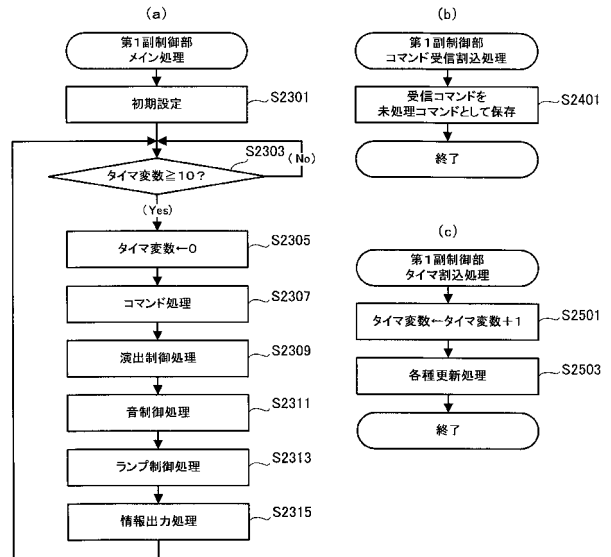




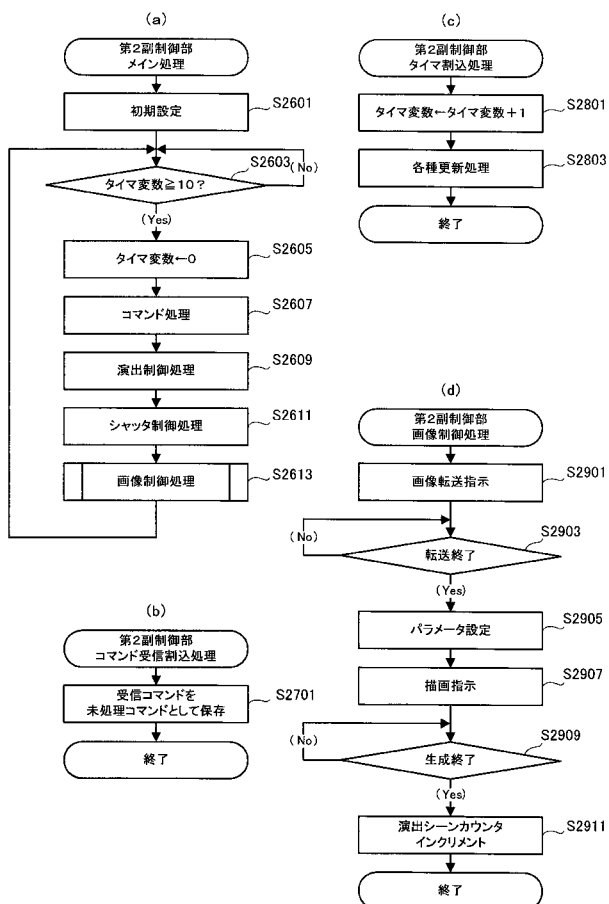
【図 7 1】



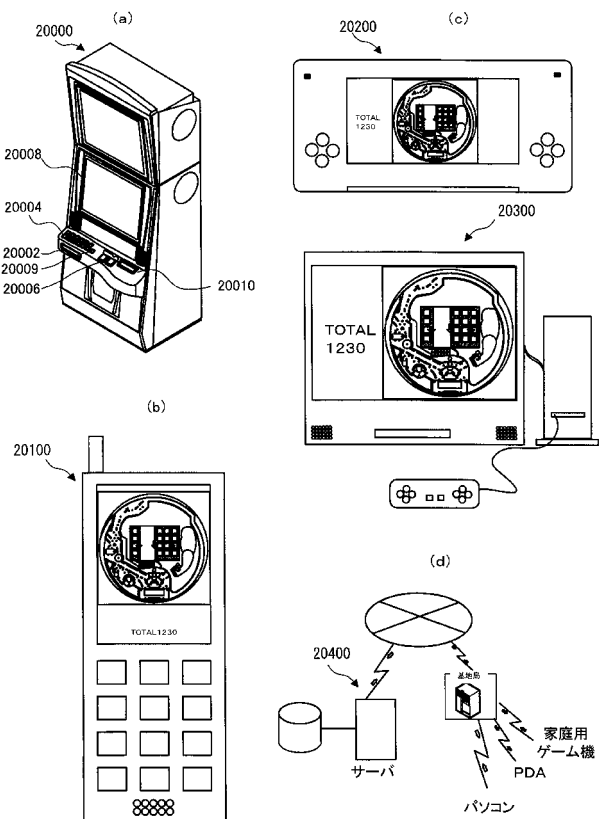
【図 7 2】



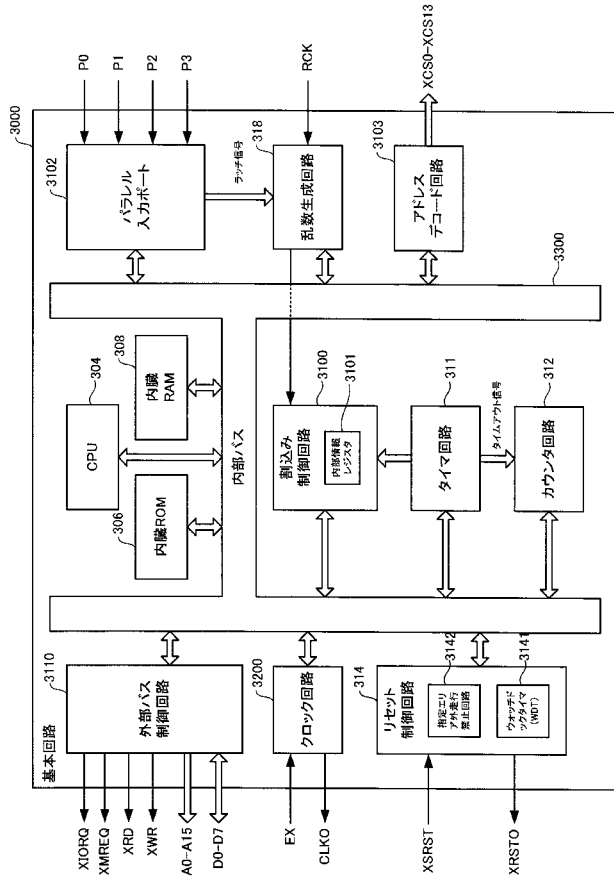
【図 7 3】



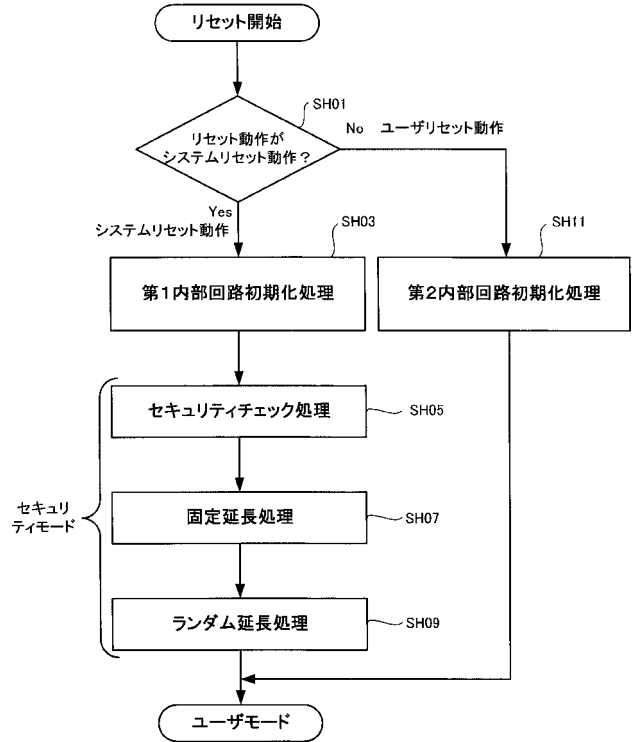
【図 7 4】



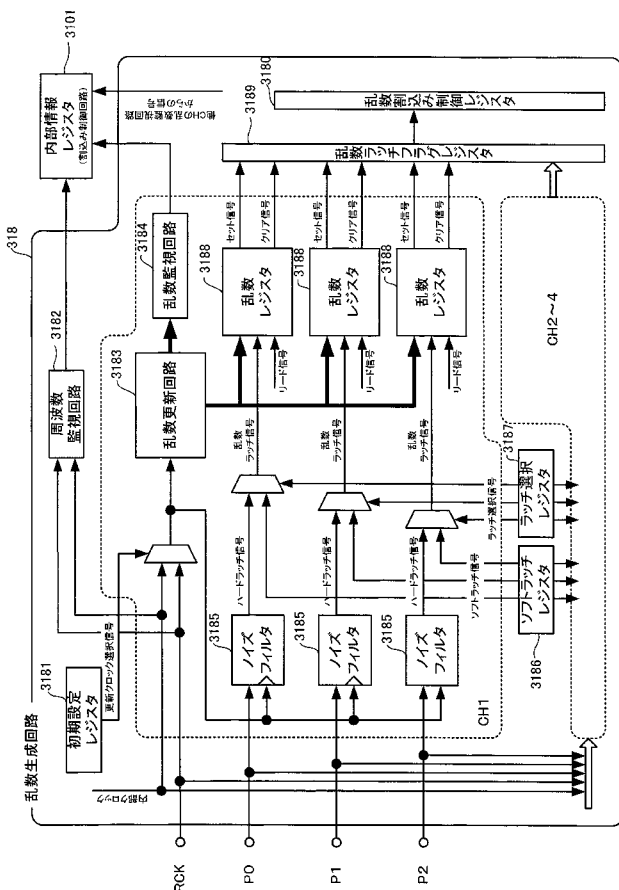
【図 75】



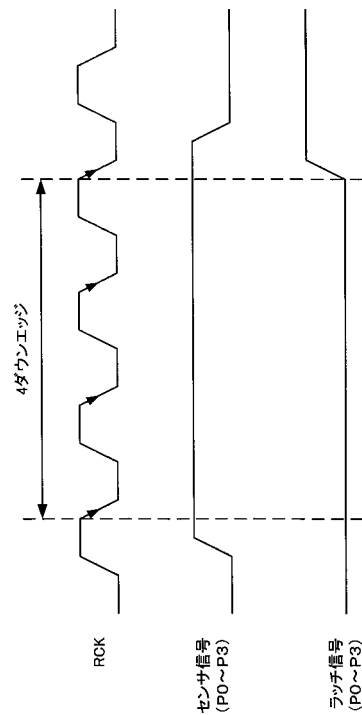
【図 76】



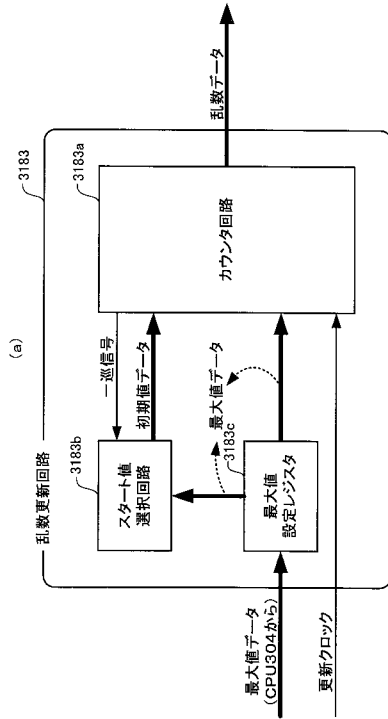
【図 77】



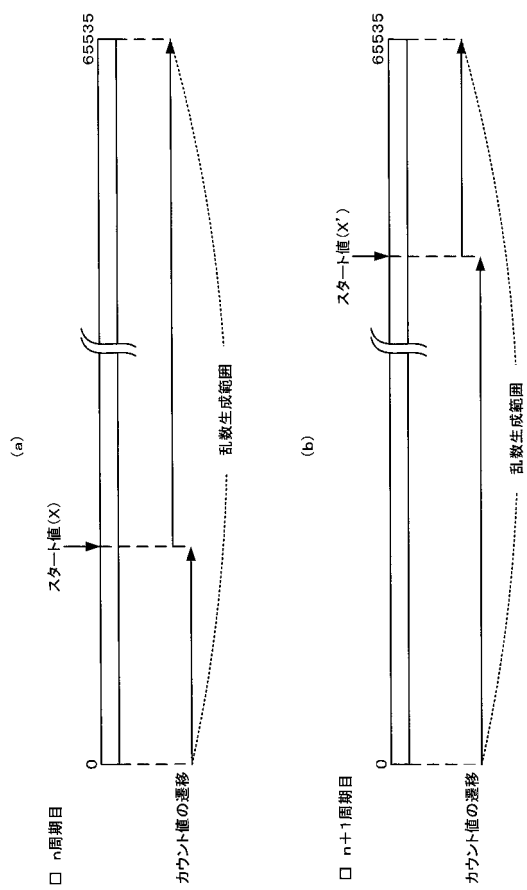
【図 78】



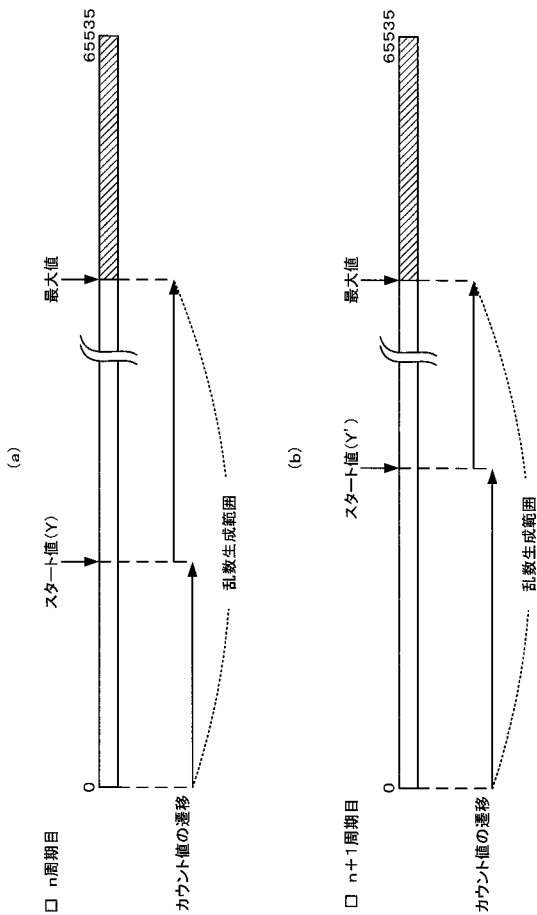
【図 79】



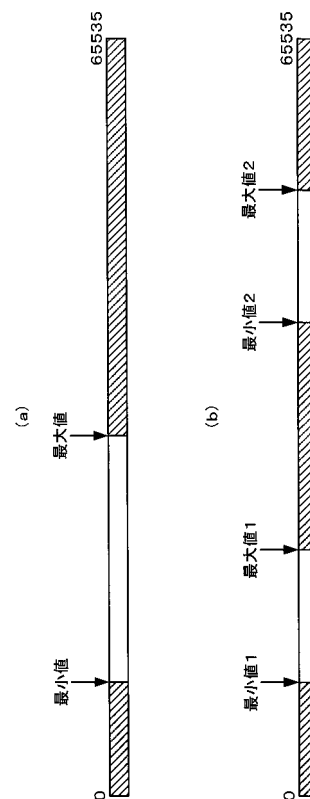
【図 80】



【図 81】



【図 82】



【図 8 3】

| 乱数                                                      | 更新場所                                                      | 備考                                                        |
|---------------------------------------------------------|-----------------------------------------------------------|-----------------------------------------------------------|
| 特図1当選乱数<br>(非確率変動中)                                     | 乱数生成回路                                                    | 乱数生成回路のCH1                                                |
| 特図1当選乱数<br>(確率変動中)                                      | 乱数生成回路                                                    | 乱数生成回路のCH2                                                |
| 特図2当選乱数<br>(非確率変動中)                                     | 乱数生成回路                                                    | 乱数生成回路のCH1                                                |
| 特図2当選乱数<br>(確率変動中)                                      | 乱数生成回路                                                    | 乱数生成回路のCH2                                                |
| 普図当選乱数<br>(普図低確率状態)                                     | 乱数生成回路                                                    | 乱数生成回路のCH3                                                |
| 普図当選乱数<br>(普図高確率状態)                                     | 乱数生成回路                                                    | 乱数生成回路のCH4                                                |
| 大当り用特図乱数<br>(特図1、2共通)                                   | 主制御部タイマ割込み処理                                              | 主制御部タイマ割込み処理                                              |
| 小当り用特図乱数<br>(特図1、2共通)                                   | 主制御部タイマ割込み処理                                              | 主制御部タイマ割込み処理                                              |
| はずれ用特図乱数<br>(特図1、2共通)                                   | 主制御部タイマ割込み処理                                              | 主制御部タイマ割込み処理                                              |
| 初期値生成用乱数<br>(大当り、小当り、はずれ共通)<br>特図変動時間決定用乱数<br>(特図1、2共通) | 主制御部メイン処理<br>主制御部タイマ割込み処理<br>主制御部タイマ割込み処理<br>主制御部タイマ割込み処理 | 主制御部メイン処理<br>主制御部タイマ割込み処理<br>主制御部タイマ割込み処理<br>主制御部タイマ割込み処理 |
| 普図変動時間決定用乱数                                             | カウンタ回路                                                    | カウンタ回路のCHO                                                |
| 演出用乱数                                                   | カウンタ回路<br>主制御部メイン処理<br>主制御部タイマ割込み処理                       | カウンタ回路のCHO<br>主制御部メイン処理<br>主制御部タイマ割込み処理                   |

【図 8 4】

(a)

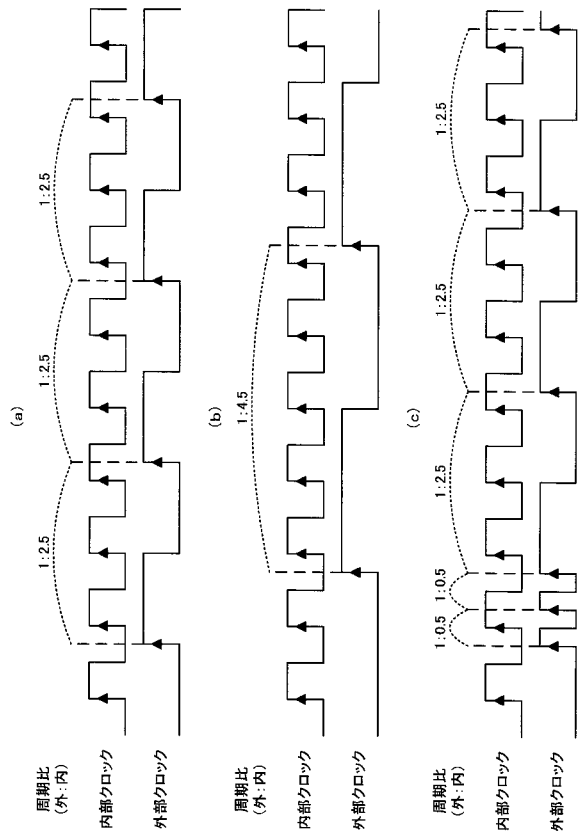
内部情報レジスタ

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 0 | 0 | 0 | — | — | — | — | — |

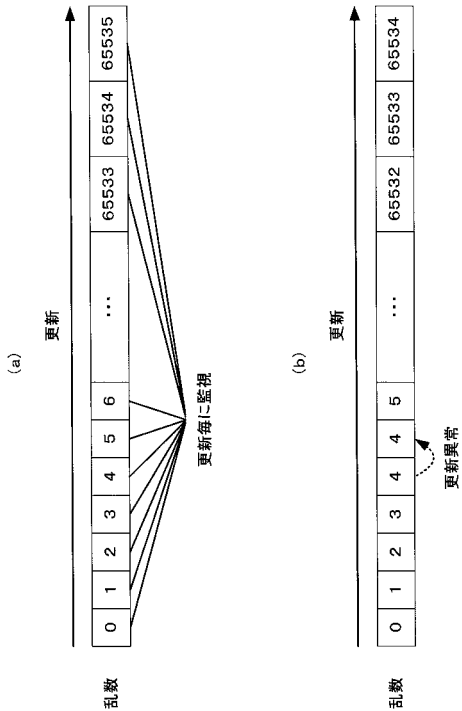
(b)

| ビット番号 | ビット名            | 詳細               |
|-------|-----------------|------------------|
| 0     | 更新クロック異常(全CH共通) | 0:異常なし<br>1:異常あり |
| 1     | 乱数更新異常(CH1)     | 0:異常なし<br>1:異常あり |
| 2     | 乱数更新異常(CH2)     | 0:異常なし<br>1:異常あり |
| 3     | 乱数更新異常(CH3)     | 0:異常なし<br>1:異常あり |
| 4     | 乱数更新異常(CH4)     | 0:異常なし<br>1:異常あり |
| 5     | —               | 0固定              |
| 6     | —               | 0固定              |
| 7     | —               | 0固定              |

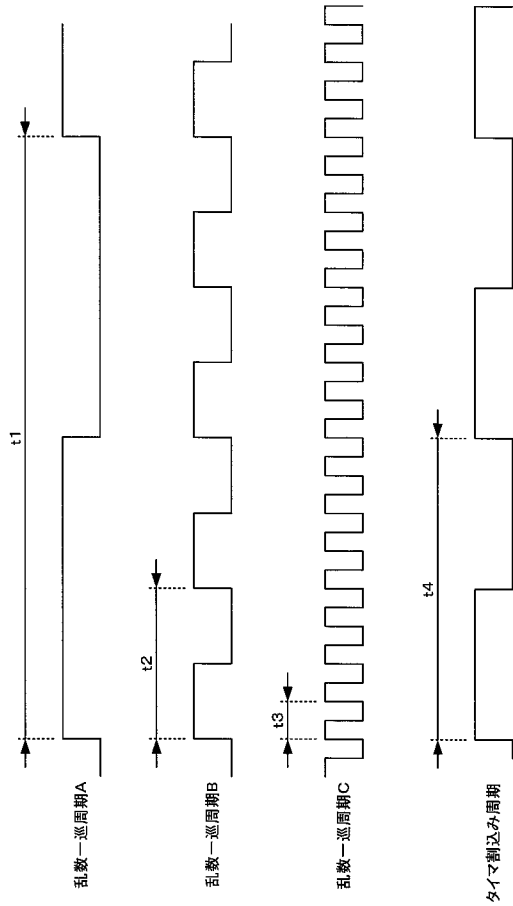
【図 8 5】



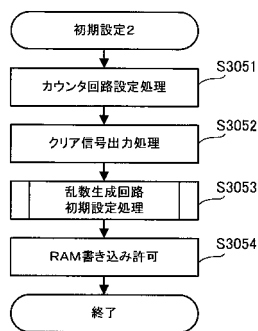
【図 8 6】



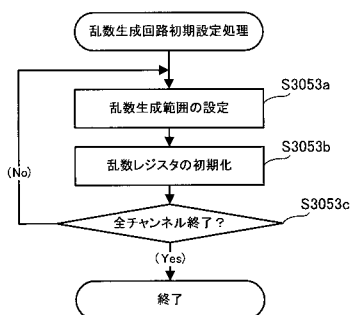
【図 87】



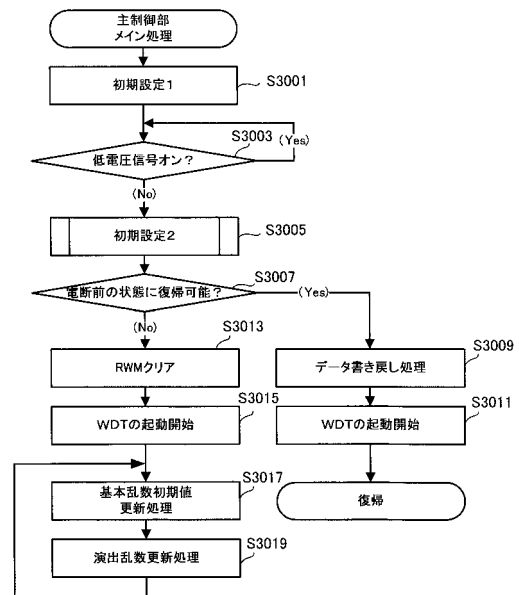
【図 89】



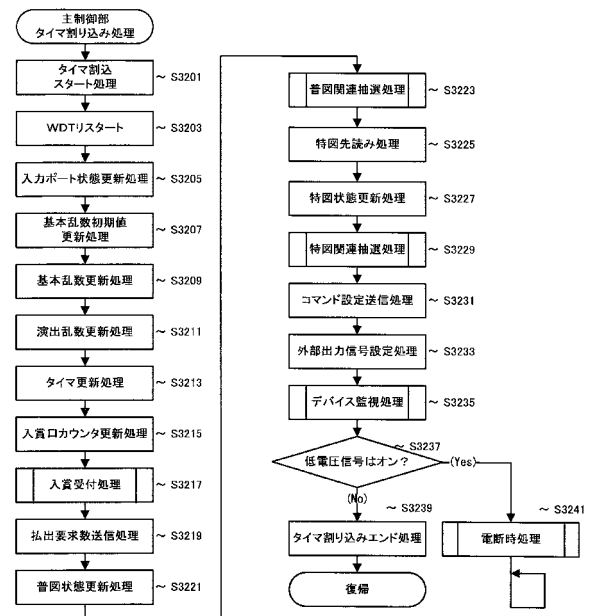
【図 90】



【図 88】

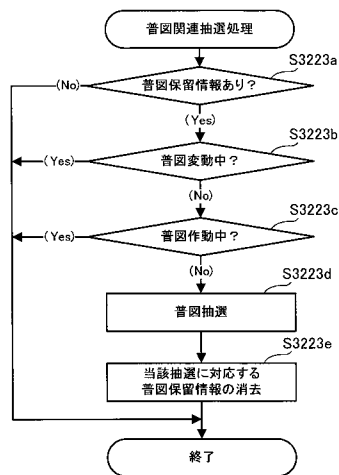


【図 91】

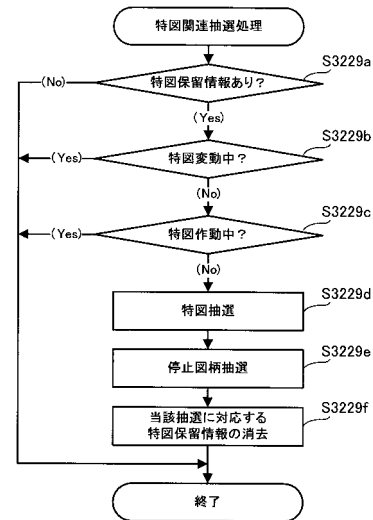


【図 9 2】

(a)



【図 9 3】



(b) 普図抽選テーブル(普図低確率状態/普図高確率状態)

|         | 普図当選データ | 乱数生成範囲設定 | 当選確率  | 対応CH |
|---------|---------|----------|-------|------|
| 普図低確率状態 | 9       | 0~999    | 1/100 | CH3  |
| 普図高確率状態 | 9       | 0~9      | 1/1   | CH4  |

【図 9 4】

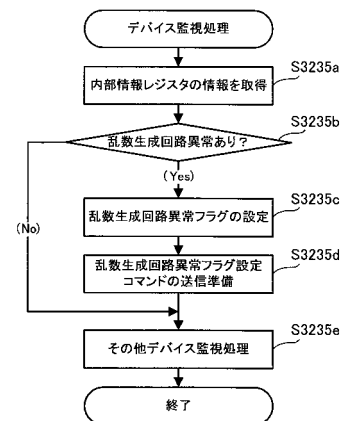
(a) 特図抽選テーブル(非確率変動中/確率変動中)

|        | 特図大当り当選データ | 乱数生成範囲  | 当選確率  | 対応CH |
|--------|------------|---------|-------|------|
| 非確率変動中 | 124        | 0~49999 | 1/400 | CH1  |
| 確率変動中  | 124        | 0~4999  | 1/40  | CH2  |

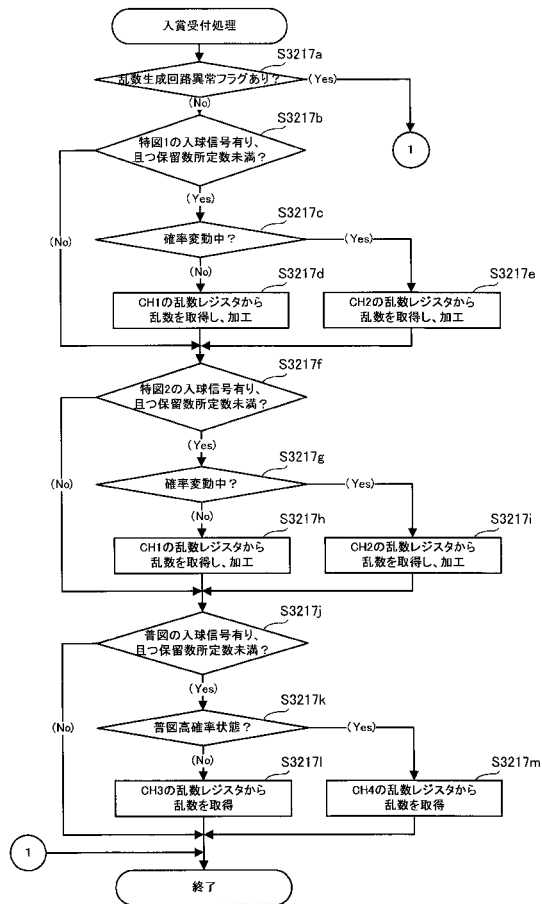
(b) 停止図柄抽選テーブル(乱数生成範囲:0~99)

| 当否判定結果 | 停止図柄 | 図柄当選データ | 当選確率   |
|--------|------|---------|--------|
| 大当り    | 特図A  | 50      | 50/100 |
|        | 特図B  | 10      | 10/100 |
|        | 特図C  | 10      | 10/100 |
|        | 特図D  | 10      | 10/100 |
|        | 特図E  | 10      | 10/100 |
|        | 特図F  | 10      | 10/100 |
| 小当たり   | 特図G  | 50      | 50/100 |
|        | 特図H  | 50      | 50/100 |
| はずれ    | 特図I  | 50      | 50/100 |
|        | 特図J  | 50      | 50/100 |

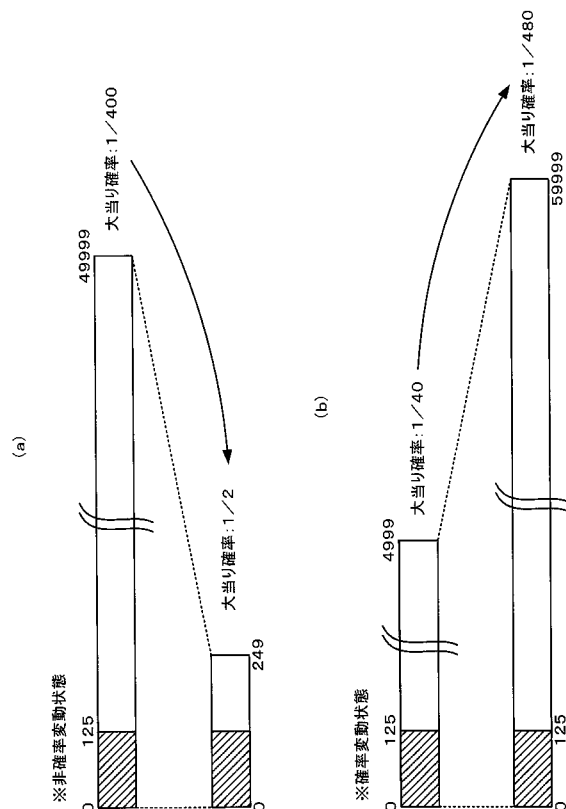
【図 9 5】



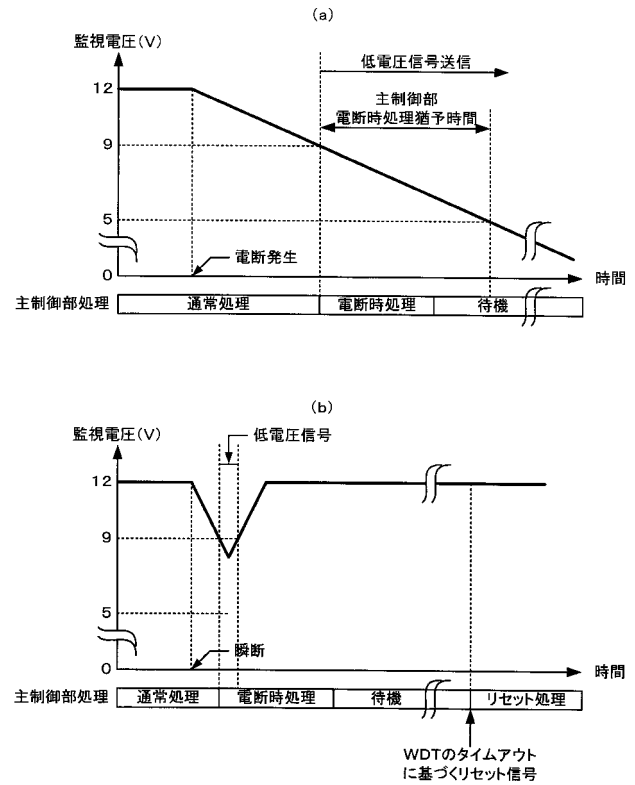
【図 96】



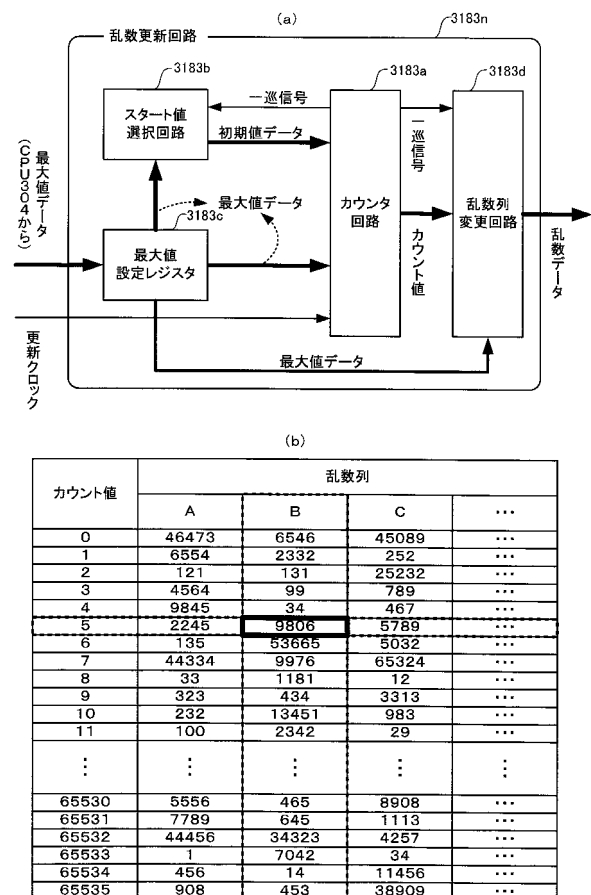
【図 98】



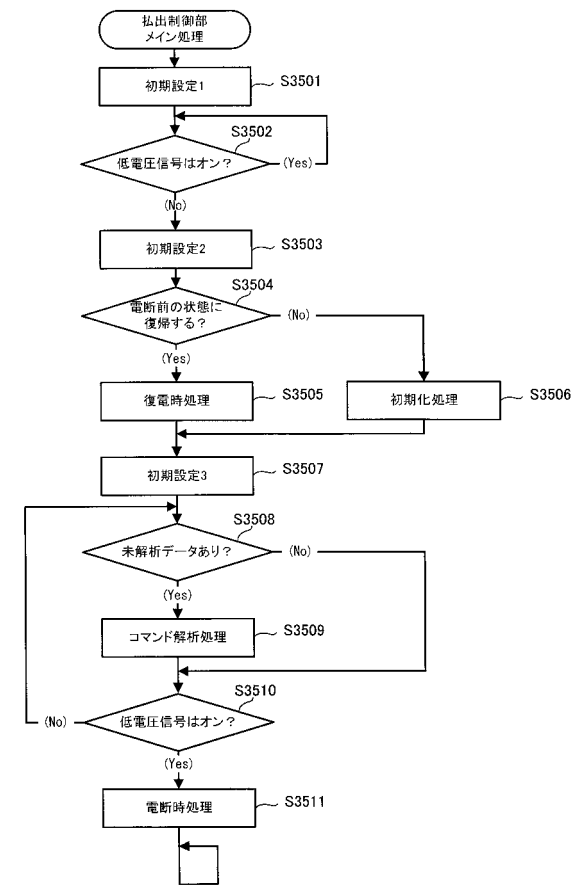
【図 97】



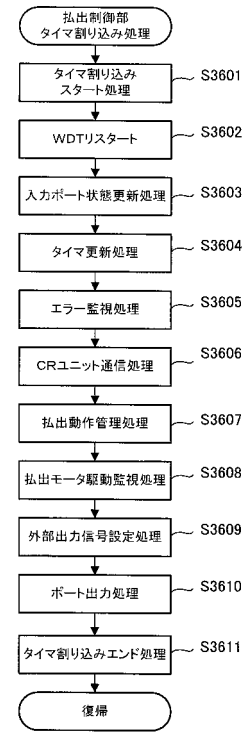
【図 99】



【図 1 0 0】



【図 1 0 1】



【図 1 0 2】

|            |   |   |   |   |   |   |   |   |
|------------|---|---|---|---|---|---|---|---|
| 第2内部情報レジスタ | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|------------|---|---|---|---|---|---|---|---|

| ビット番号 | 意味                                                                                                  |
|-------|-----------------------------------------------------------------------------------------------------|
| 2ビット  | 直前に発生したリセット要因が、システムリセットであることを示すビット<br>0: システムリセット未発生<br>1: システムリセット発生                               |
| 1ビット  | 直前に発生したリセット要因が、WDTタイムアウト信号によるリセットであることを示すビット<br>0: WDTタイムアウト信号によるリセット未発生<br>1: WDTタイムアウト信号によるリセット発生 |
| 0ビット  | 直前に発生したリセット要因が、IAT発生信号によるリセットであることを示すビット<br>0: IAT発生信号によるリセット未発生<br>1: IAT発生信号によるリセット発生             |

【図 1 0 3】

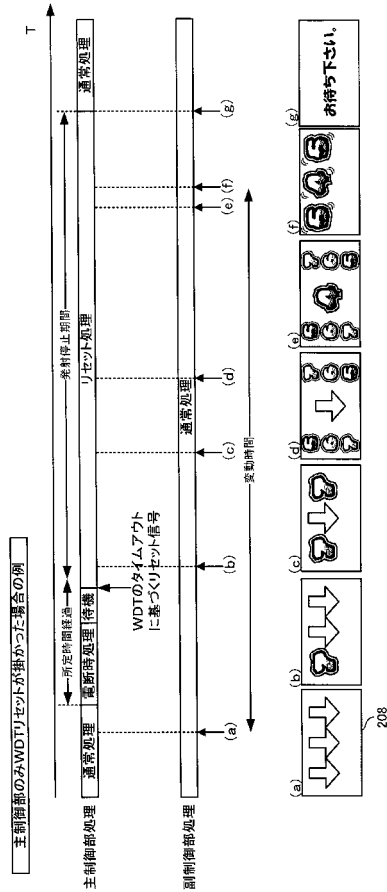
|            |   |   |   |   |   |   |   |   |
|------------|---|---|---|---|---|---|---|---|
| 第3内部情報レジスタ | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|------------|---|---|---|---|---|---|---|---|

| ビット番号 | 意味                                                           |
|-------|--------------------------------------------------------------|
| 1ビット  | WDTタイムアウト信号によるリセットの動作を示すビット<br>0: ユーザリセット動作<br>1: システムリセット動作 |
| 0ビット  | IAT発生信号によるリセットの動作を示すビット<br>0: ユーザリセット動作<br>1: システムリセット動作     |

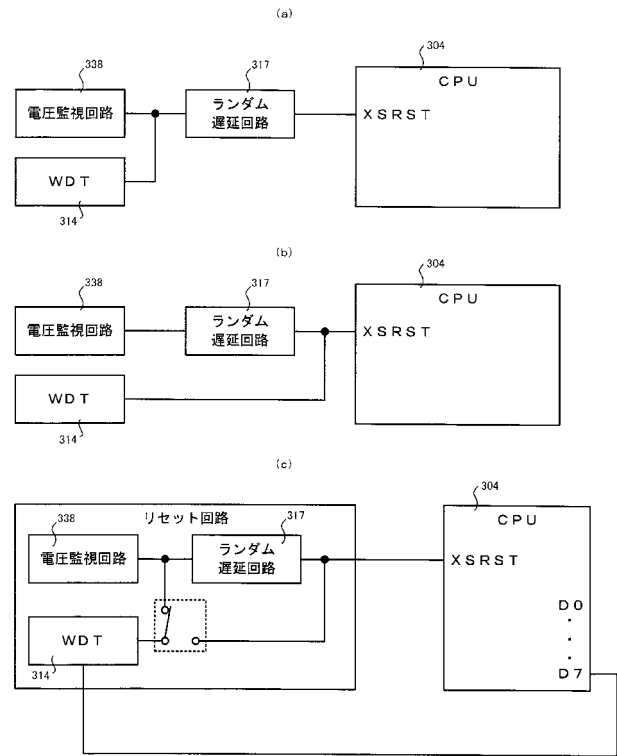




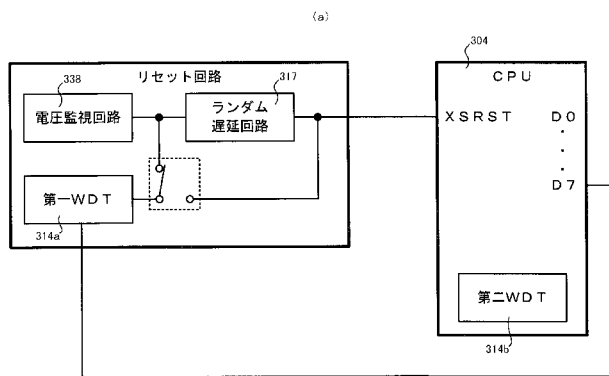
【図108】



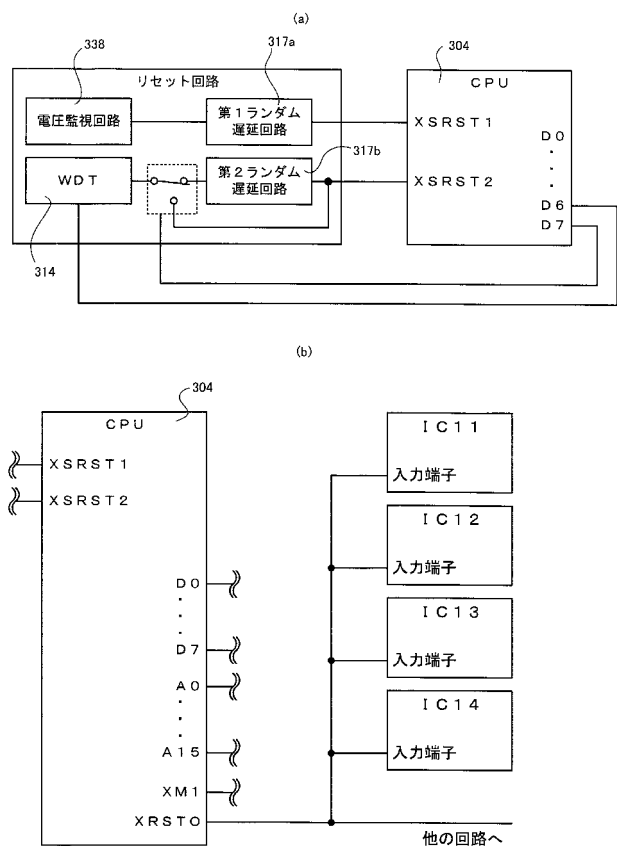
【図109】



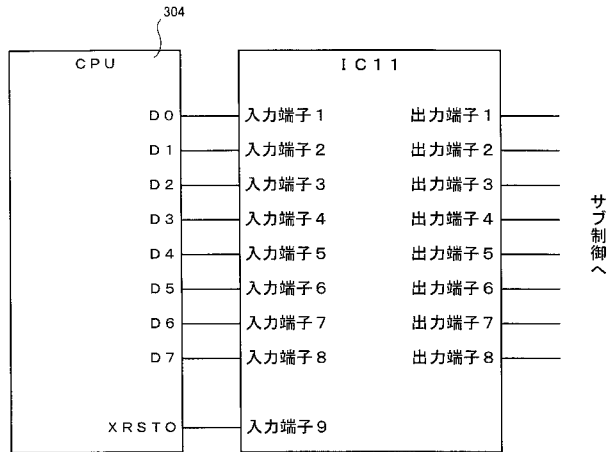
【図110】



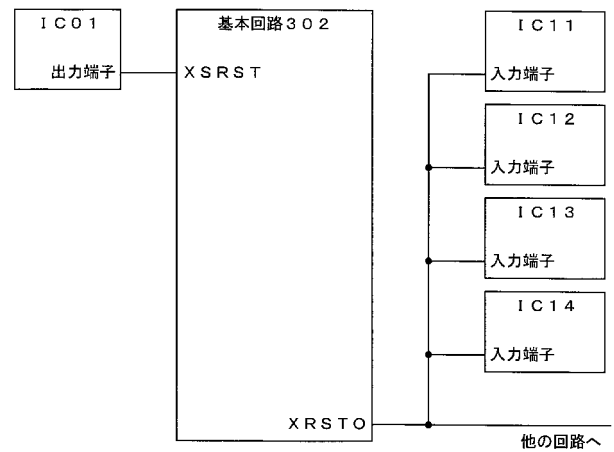
【図111】



【図 1 1 2】



【図 1 1 3】



【図 1 1 4】

(a) 固定延長時間の設定例

| 設定値 | 延長時間        |              |              |
|-----|-------------|--------------|--------------|
|     | SCLK = 8MHz | SCLK = 10MHz | SCLK = 12MHz |
| 000 | 延長しない       |              |              |
| 001 | 約525ms      | 約420ms       | 約350ms       |
| 010 | 約1050ms     | 約840ms       | 約700ms       |
| 011 | 約1575ms     | 約1260ms      | 約1050ms      |
| 100 | 約2100ms     | 約1680ms      | 約1400ms      |
| 101 | 約2625ms     | 約2100ms      | 約1750ms      |
| 110 | 約3150ms     | 約2520ms      | 約2100ms      |
| 111 | 約30000ms    | 約30000ms     | 約30000ms     |

(b) ランダム延長時間の設定例

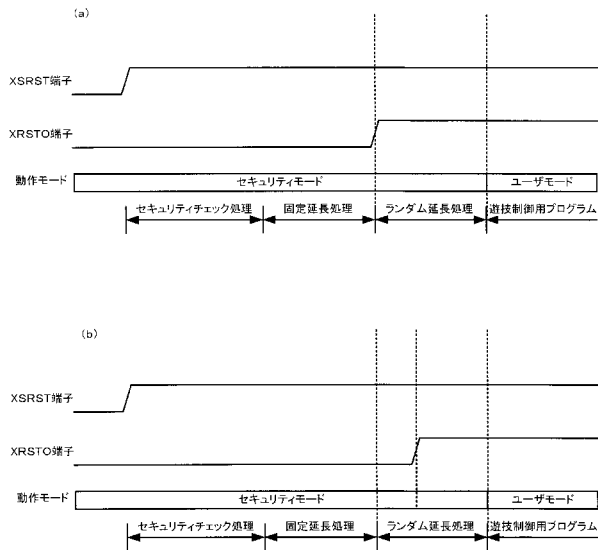
| 設定値 | 延長時間の範囲     |              |              |
|-----|-------------|--------------|--------------|
|     | SCLK = 8MHz | SCLK = 10MHz | SCLK = 12MHz |
| 00  | 延長しない       |              |              |
| 01  | 0~0.5ms     | 0~0.4ms      | 0~0.3ms      |
| 11  | 0~250.0ms   | 0~200.0ms    | 0~150.0ms    |

【図 1 1 5】

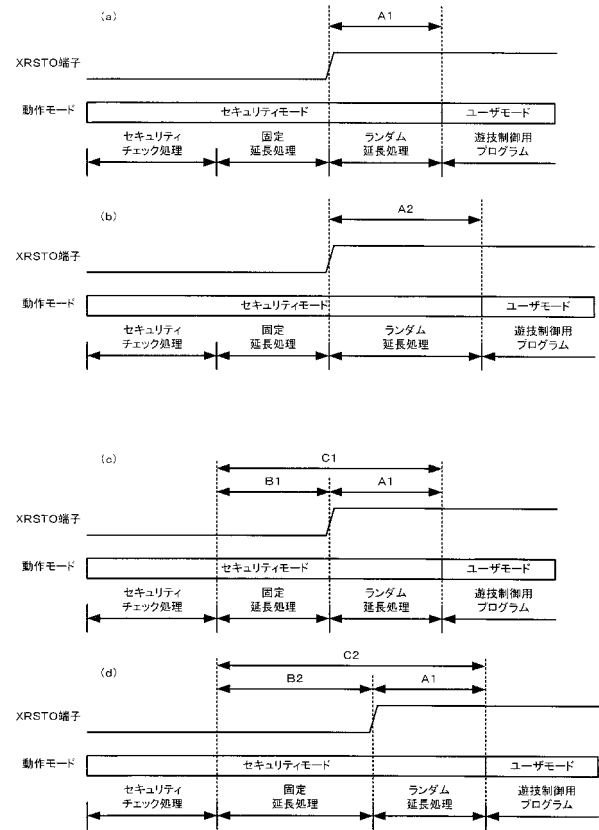
WDTタイムアウト時間の設定例

| 設定値  | 延長時間        |              |              |
|------|-------------|--------------|--------------|
|      | SCLK = 8MHz | SCLK = 10MHz | SCLK = 12MHz |
| 0000 | 禁止する(使用しない) |              |              |
| 0001 | 約65ms       | 約52ms        | 約44ms        |
| 0010 | 約525ms      | 約420ms       | 約350ms       |
| 0011 | 約4250ms     | 約3400ms      | 約2800ms      |
| 0100 | 約29375ms    | 約23500ms     | 約22400ms     |
| 0101 | 約122ms      | 約98ms        | 約82ms        |
| 0110 | 約982ms      | 約786ms       | 約655ms       |
| 0111 | 約7875ms     | 約6300ms      | 約5200ms      |
| 1000 | 約82500ms    | 約50000ms     | 約42000ms     |

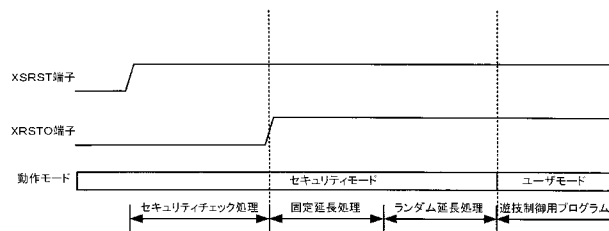
【図 1 1 6】



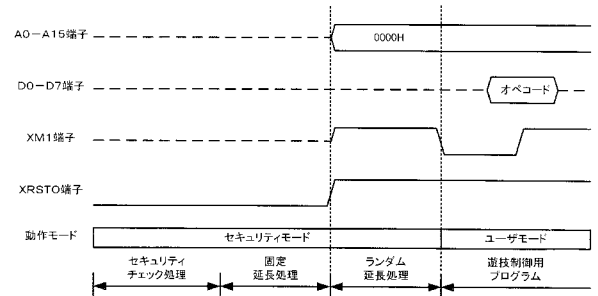
【図 1 1 7】



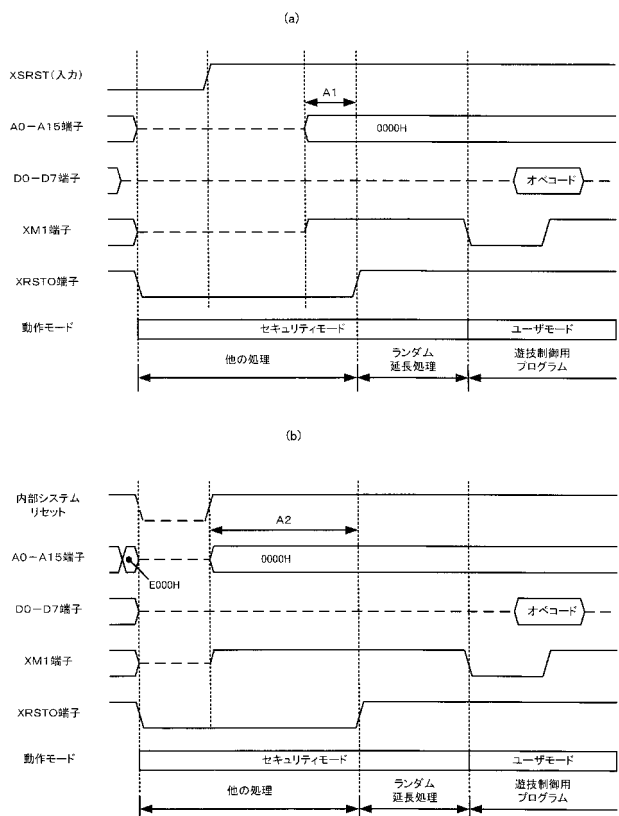
【図 1 1 8】



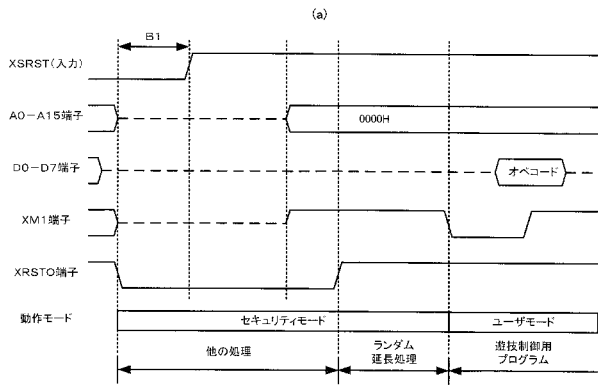
【図 1 1 9】



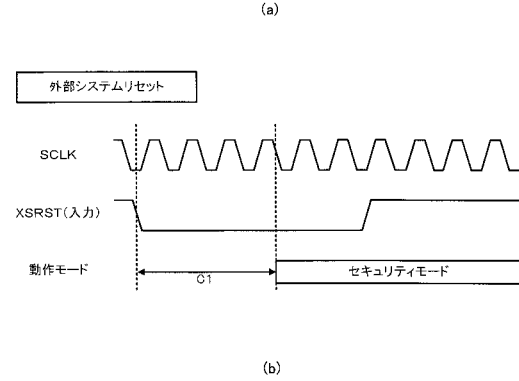
【図 1 2 0】



【図 1 2 1】



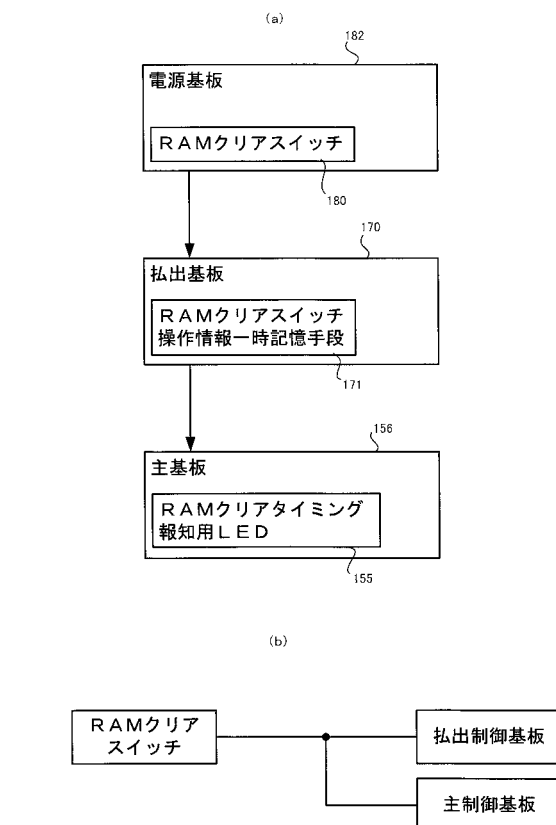
【図 1 2 2】



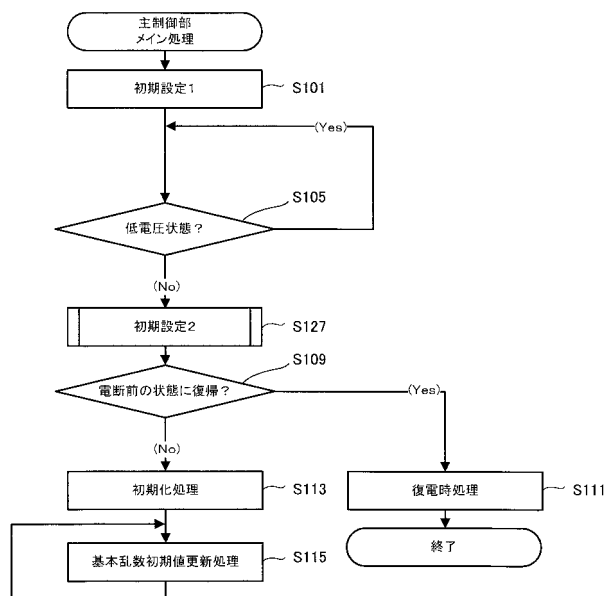
【図 1 2 3】



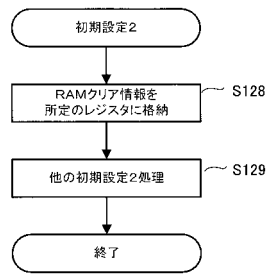
【図 1 2 4】



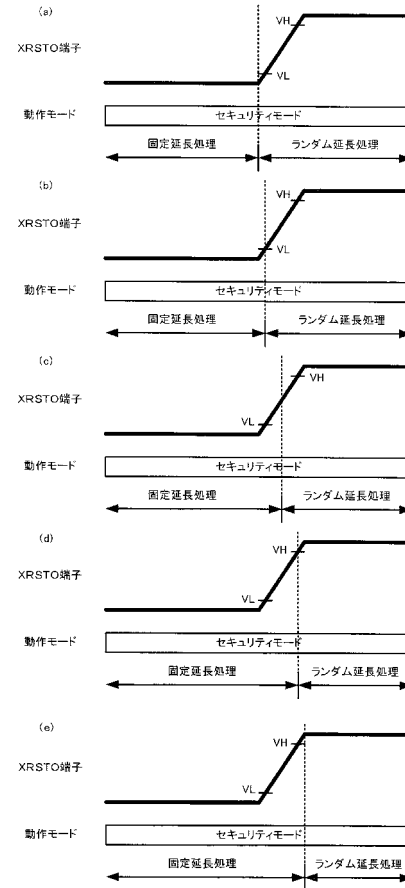
【図 1 2 5】



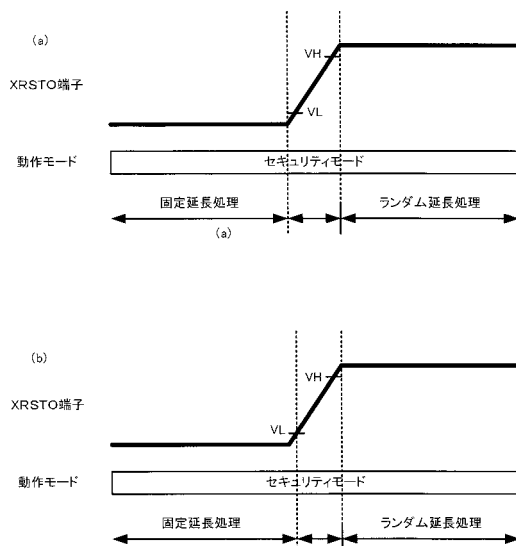
【図 1 2 6】



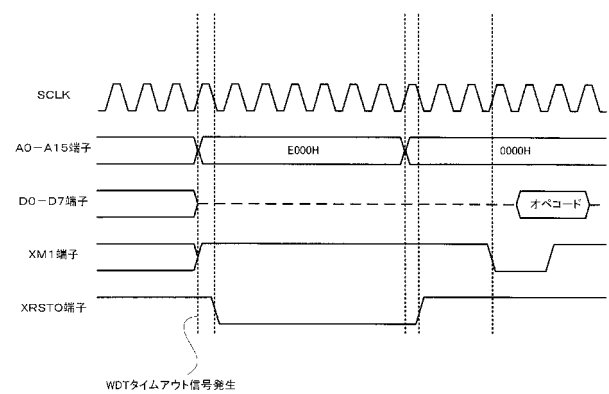
【図 1 2 7】



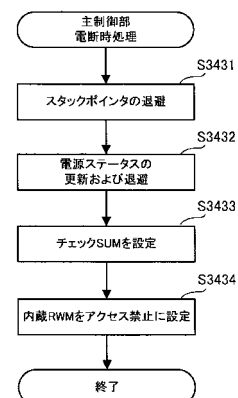
【図 1 2 8】



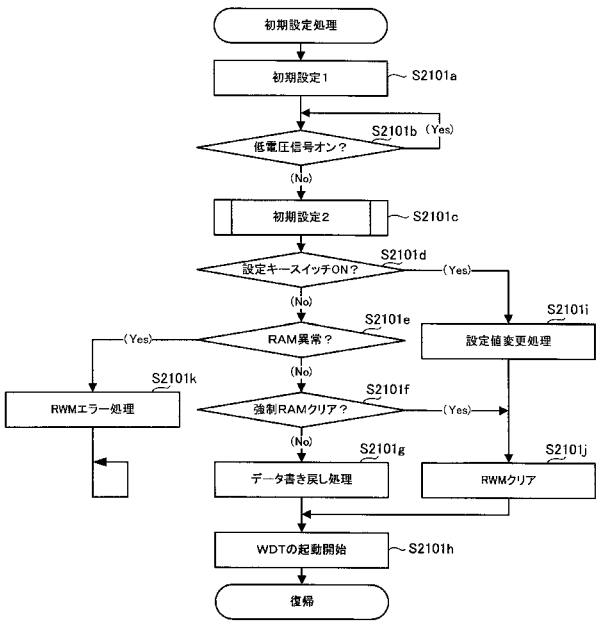
【図 1 2 9】



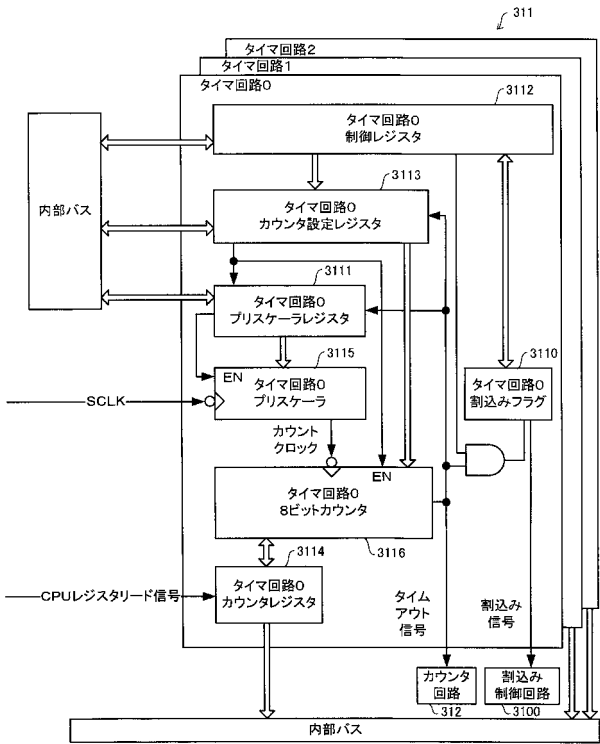
【図 1 3 0】



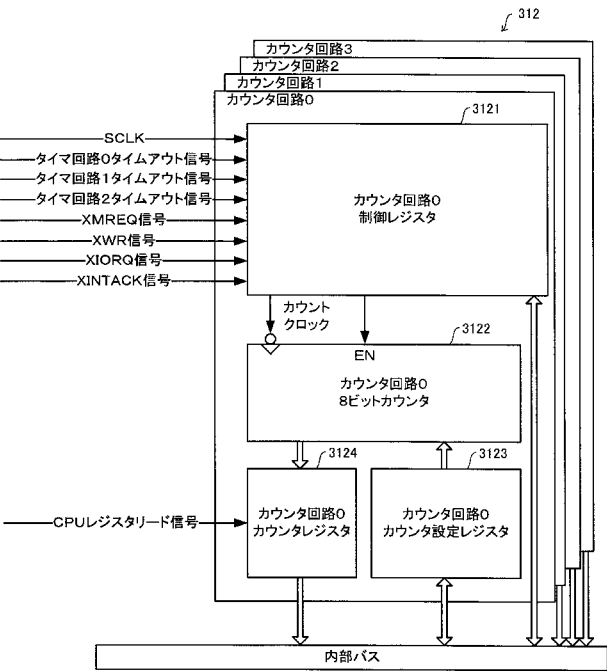
【図 1 3 1】



【図 1 3 2】



【図 1 3 3】

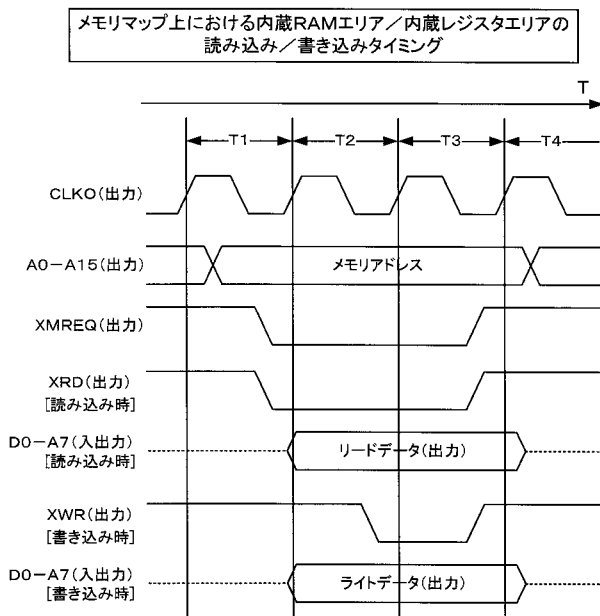


【図 1 3 4】

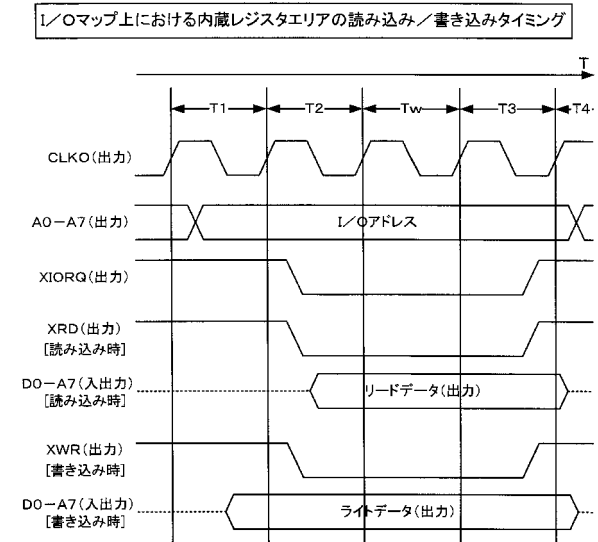
|               |   |   |   |   |   |   |   |   |
|---------------|---|---|---|---|---|---|---|---|
| カウンタ回路0制御レジスタ | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| カウンタ回路1制御レジスタ | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| カウンタ回路2制御レジスタ | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| カウンタ回路3制御レジスタ | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

| ビット番号  | 意味                                                                                                                                                                                                                      |
|--------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 4ビット   | カウンタ回路n 8ビットカウンタを動作開始、停止させるためのビット<br>0:動作停止(初期値)<br>1:動作開始                                                                                                                                                              |
| 3ビット   | カウンタ回路n 8ビットカウンタをカスケード接続設定するビット<br>0:カスケード接続未設定(初期値)<br>1:カスケード接続設定<br>※カスケード接続設定は、カウンタ回路の0と1、2と3をペアとする                                                                                                                 |
| 2-0ビット | カウンタ回路n 8ビットカウンタのカウンタクロックを設定するビット<br>000:SCLK(初期値)<br>001:タイマ回路0 8ビットカウンタタイマのタイムアウト信号<br>010:タイマ回路1 8ビットカウンタタイマのタイムアウト信号<br>011:タイマ回路2 8ビットカウンタタイマのタイムアウト信号<br>100:XWR信号<br>101:XMREQ信号<br>110:XIORQ信号<br>111:XINTACK信号 |

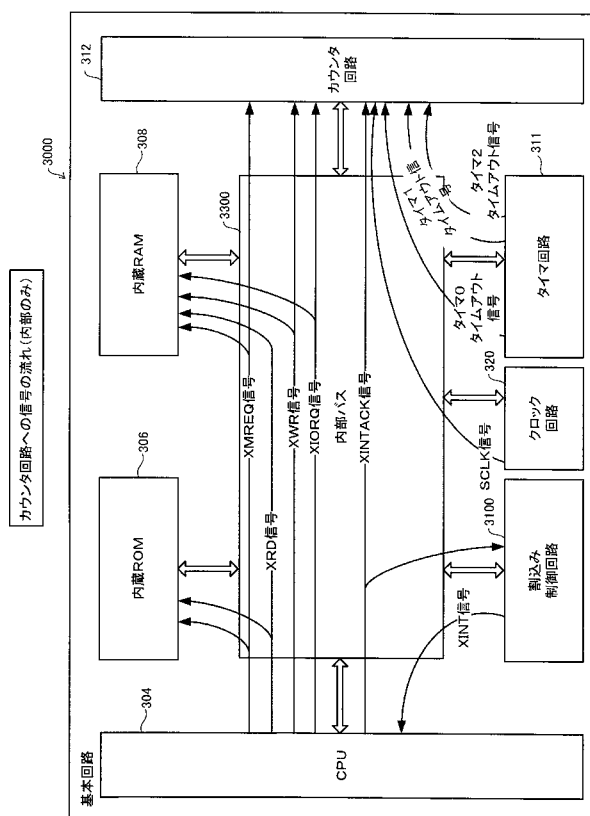
【図 1 3 5】



【図 1 3 6】

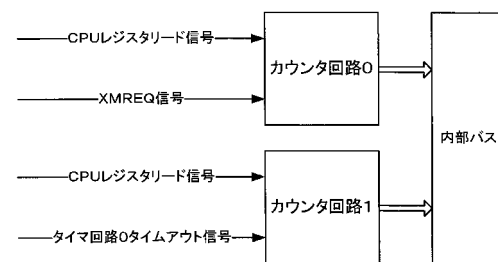


【図 1 3 7】



【図 1 3 8】

カウンタ回路0を不定期な更新とする例

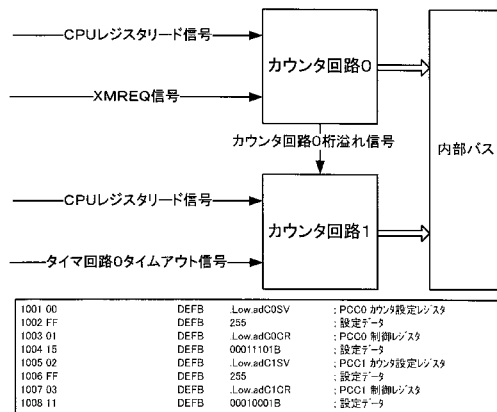


|         |      |             |                           |
|---------|------|-------------|---------------------------|
| 1001 00 | DEFB | .Low.adC0SV | : PCC0 カウンタ設定レジスタ         |
| 1002 FF | DEFB | 255         | : 設定データ                   |
| 1003 01 | DEFB | .Low.adC0CR | : PCC0 制御レジスタ             |
| 1004 15 | DEFB | 00010101B   | : 設定データ                   |
| 1005 02 | DEFB | .Low.adC1SV | : PCC1 カウンタ設定レジスタ         |
| 1006 FF | DEFB | 255         | : 設定データ                   |
| 1007 03 | DEFB | .Low.adC1CR | : PCC1 制御レジスタ             |
| 1008 11 | DEFB | 00010001B   | : 設定データ                   |
| 1009 0C | DEFB | .Low.adT0PR | : PTC0 プリスケールレジスタ         |
| 100A F0 | DEFB | 240         | : 設定データ                   |
| 100B 0D | DEFB | .Low.adT0CR | : PTC0 制御レジスタ             |
| 100C 41 | DEFB | 01000001B   | : 設定データ                   |
| 100D    |      |             | : 前込み許可、タイムアウト転送、ビットモード   |
| 100E 0E | DEFB | .Low.adT0SV | : PTC0 カウンタ設定レジスタ         |
| 100F C8 | DEFB | 200         | : 設定データ                   |
| 100F    |      |             | : 周期 4ms (=240*200/12MHz) |



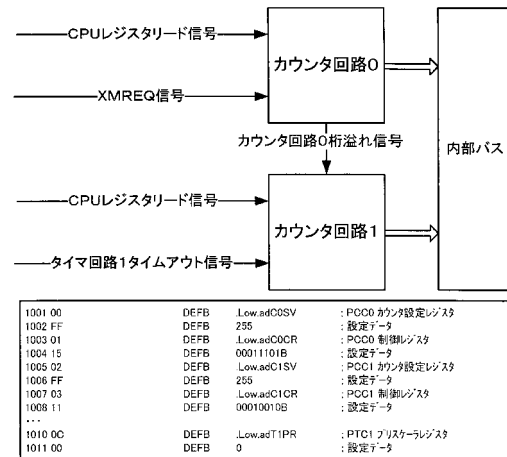
【図 1 3 9】

変形例1: カウンタ回路0をカウンタ回路1の更新ソースとする例(カスケード接続)



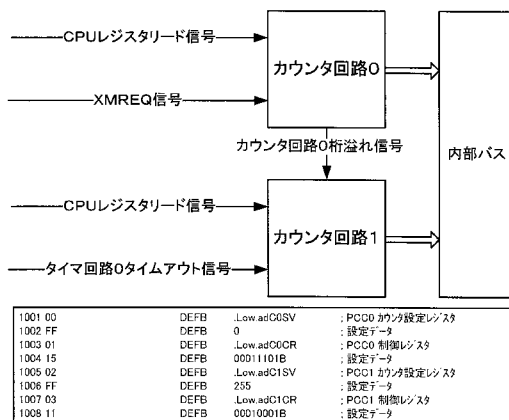
【図 1 4 0】

変形例2: カウンタ回路0とカウンタ回路1を2バイトカウンタとする例



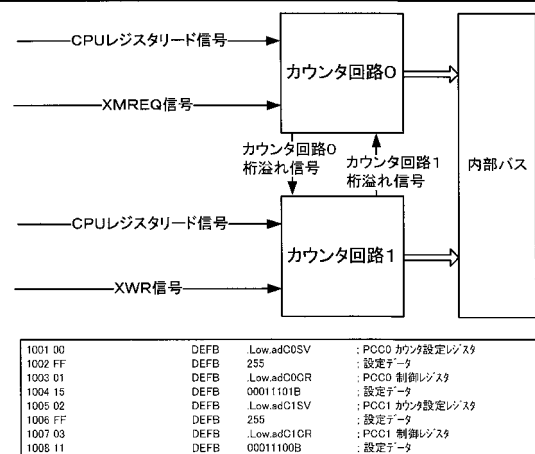
【図 1 4 1】

変形例3: カウンタ回路0の更新ソースをカウンタ回路1のソースとする例

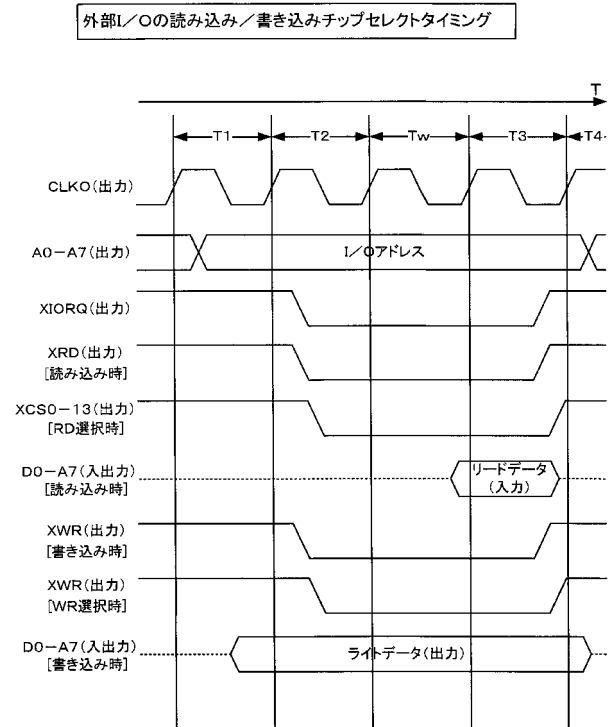


【図 1 4 2】

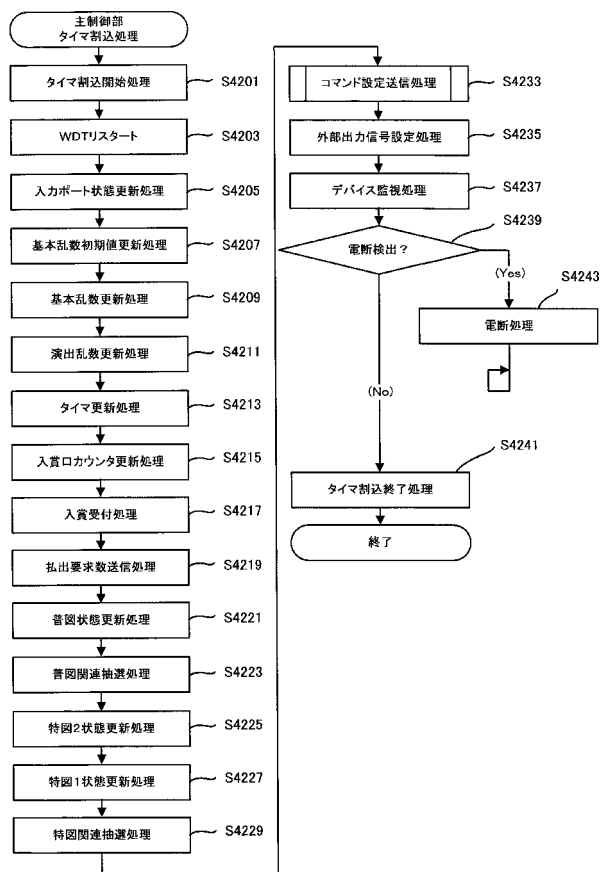
変形例4: カウンタ回路0とカウンタ回路1を相互にカスケード接続する例



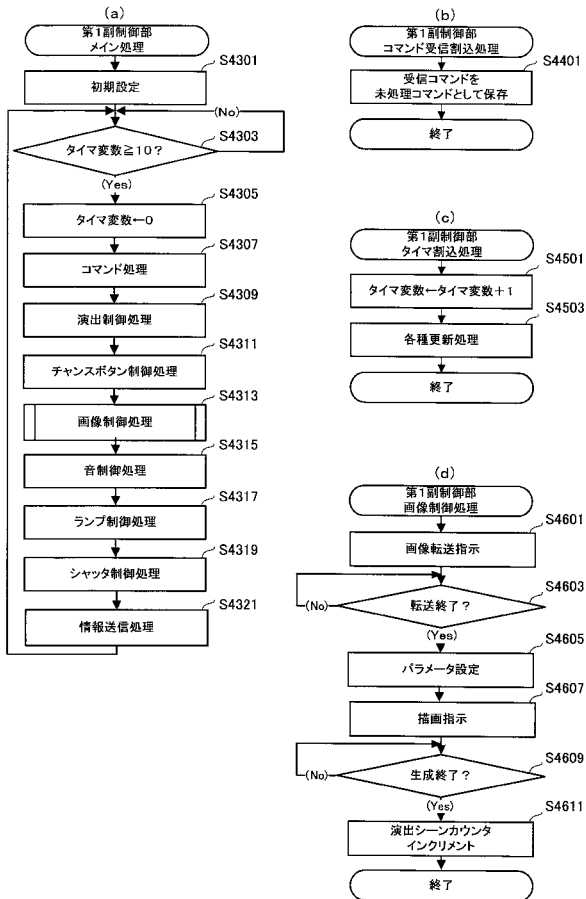
【図 1 4 4】



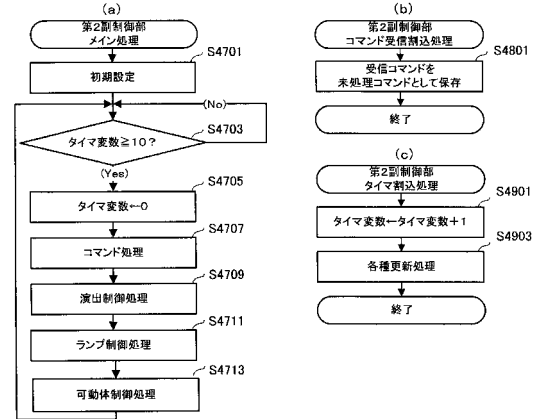
【图 1 4 6】



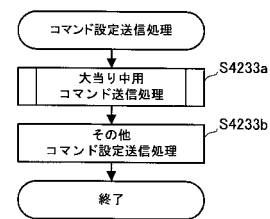
【図 147】



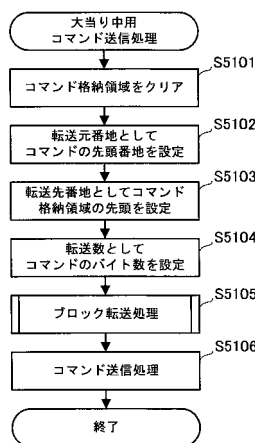
【図 148】



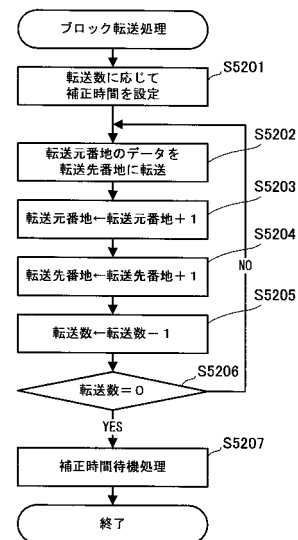
【図 149】



【図 150】



【図 151】



【図 1 5 2】

(a) 大当り開始コマンド格納領域

| コマンドの名称    | 先頭番地 | バイト数 | 1バイト目 | 2バイト目 |
|------------|------|------|-------|-------|
| 大当り開始コマンド1 | 100  | 2    | 11H   | 01H   |
| 大当り開始コマンド2 | 102  | 2    | 11H   | 02H   |
| ⋮          | ⋮    | ⋮    | ⋮     | ⋮     |
| 大当り開始コマンド6 | 110  | 2    | 11H   | 06H   |

(b) ラウンド開始コマンド格納領域

| コマンドの名称      | 先頭番地 | バイト数 | 1バイト目 | 2バイト目 | 3バイト目 |
|--------------|------|------|-------|-------|-------|
| ラウンド開始コマンド1  | 200  | 3    | 12H   | 01H   | 01H   |
| ラウンド開始コマンド2  | 203  | 3    | 12H   | 02H   | 01H   |
| ⋮            | ⋮    | ⋮    | ⋮     | ⋮     | ⋮     |
| ラウンド開始コマンド6  | 215  | 3    | 12H   | 06H   | 01H   |
| ラウンド開始コマンド7  | 218  | 3    | 12H   | 01H   | 02H   |
| ラウンド開始コマンド8  | 221  | 3    | 12H   | 02H   | 02H   |
| ⋮            | ⋮    | ⋮    | ⋮     | ⋮     | ⋮     |
| ラウンド開始コマンド12 | 233  | 3    | 12H   | 06H   | 02H   |
| ラウンド開始コマンド13 | 236  | 3    | 12H   | 01H   | 03H   |
| ラウンド開始コマンド14 | 239  | 3    | 12H   | 02H   | 03H   |
| ⋮            | ⋮    | ⋮    | ⋮     | ⋮     | ⋮     |
| ラウンド開始コマンド28 | 311  | 3    | 12H   | 01H   | 0FH   |
| ラウンド開始コマンド29 | 314  | 3    | 12H   | 02H   | 0FH   |

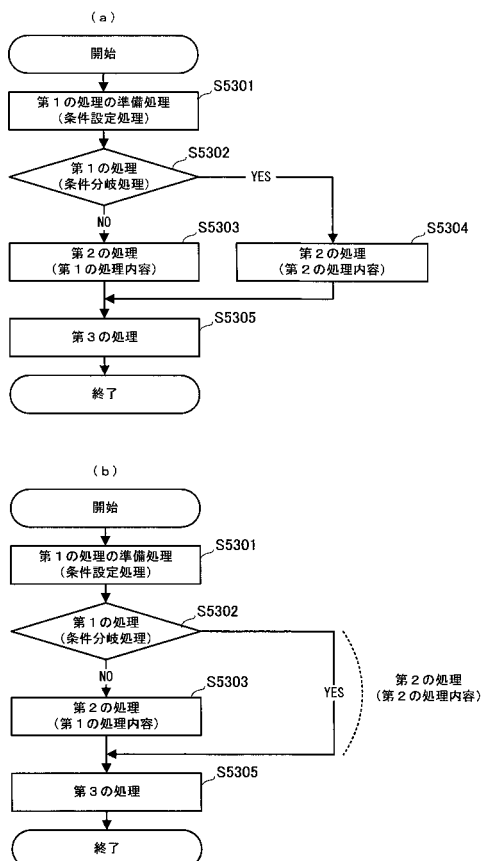
(c) ラウンド終了コマンド格納領域

| コマンドの名称      | 先頭番地 | バイト数 | 1バイト目 | 2バイト目 |
|--------------|------|------|-------|-------|
| ラウンド終了コマンド1  | 500  | 2    | 13H   | 01H   |
| ラウンド終了コマンド2  | 502  | 2    | 13H   | 02H   |
| ⋮            | ⋮    | ⋮    | ⋮     | ⋮     |
| ラウンド終了コマンド15 | 528  | 2    | 13H   | 0FH   |

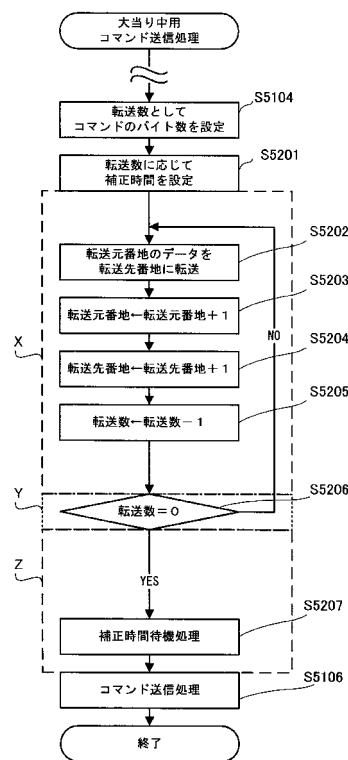
(d) 大当り終了コマンド格納領域

| コマンドの名称    | 先頭番地 | バイト数 | 1バイト目 |
|------------|------|------|-------|
| 大当り終了コマンド1 | 600  | 1    | 14H   |

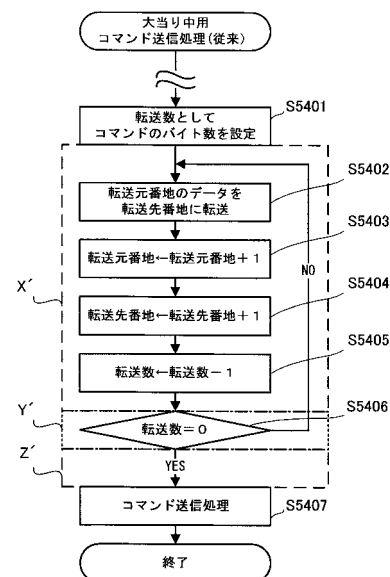
【図 1 5 4】



【図 1 5 3】



【図 1 5 5】



【図 1 5 6】

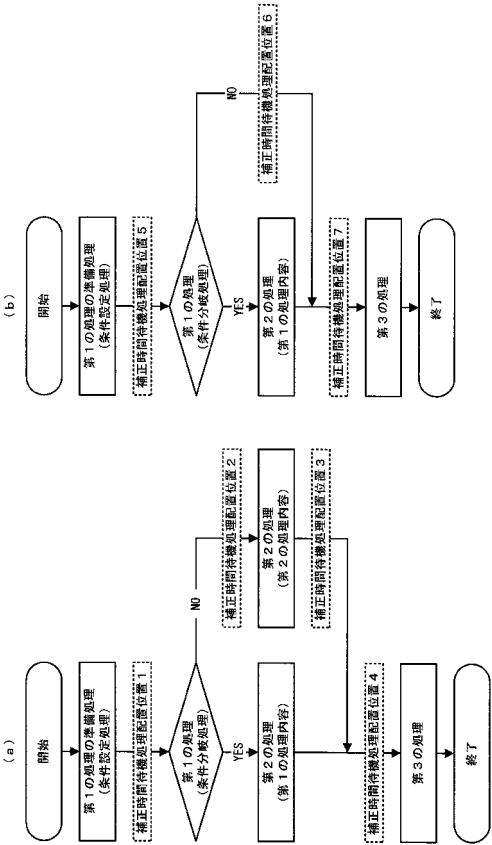
(a)

| 転送バイト数 | 所要時間          |               |                             | 合計                          |  |
|--------|---------------|---------------|-----------------------------|-----------------------------|--|
|        | 第 1 の処理       | 第 2 の処理       |                             |                             |  |
|        |               | 基本処理          | 補正時間待機処理                    |                             |  |
| 1      | T3            | -             | $T3 \times 2 + T4 \times 2$ | $T3 \times 3 + T4 \times 2$ |  |
| 2      | $T3 \times 2$ | T4            | $T3 + T4$                   | $T3 \times 3 + T4 \times 2$ |  |
| 3      | $T3 \times 3$ | $T4 \times 2$ | -                           | $T3 \times 3 + T4 \times 2$ |  |

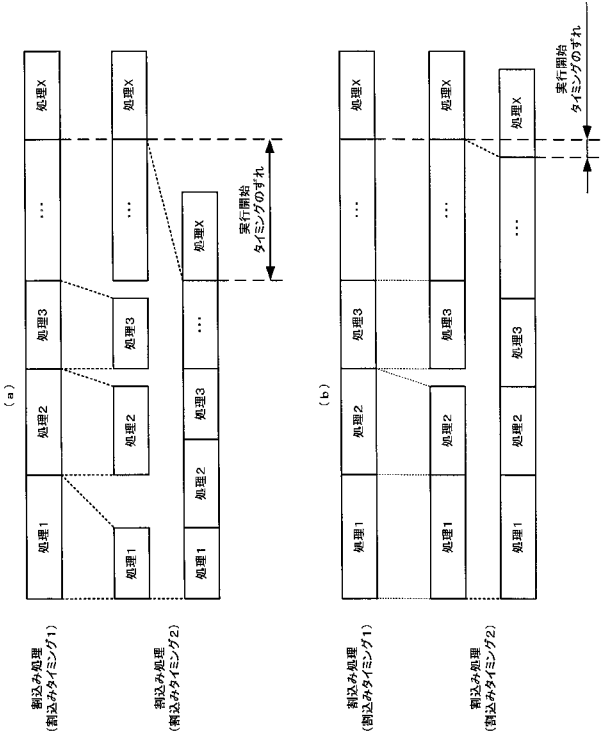
(b)

| 転送バイト数 | 所要時間          |               | 合計                          |
|--------|---------------|---------------|-----------------------------|
|        | 第 1 の処理       | 第 2 の処理       |                             |
|        |               |               |                             |
| 1      | T1            | -             | T1                          |
| 2      | $T1 \times 2$ | T2            | $T1 \times 2 + T2$          |
| 3      | $T1 \times 3$ | $T2 \times 2$ | $T1 \times 3 + T2 \times 2$ |

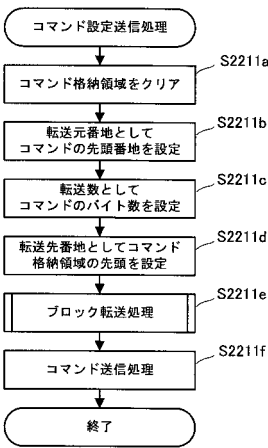
【図 1 5 7】



【図 1 5 8】



【図 1 5 9】



## 【図 160】

(a) 遊技開始コマンド格納領域

| コマンドの名称   | 先頭番地 | バイト数 | 1バイト目 | 2バイト目 |
|-----------|------|------|-------|-------|
| 遊技開始コマンド1 | 1000 | 2    | A1H   | 01H   |
| 遊技開始コマンド2 | 1002 | 2    | A1H   | 02H   |

(b) レバー受付コマンド格納領域

| コマンドの名称     | 先頭番地 | バイト数 | 1バイト目 | 2バイト目 | 3バイト目 |
|-------------|------|------|-------|-------|-------|
| レバー受付コマンド1  | 1200 | 3    | A3H   | 01H   | 01H   |
| レバー受付コマンド2  | 1203 | 3    | A3H   | 01H   | 02H   |
| レバー受付コマンド3  | 1206 | 3    | A3H   | 01H   | 03H   |
| レバー受付コマンド4  | 1209 | 3    | A3H   | 01H   | 04H   |
| レバー受付コマンド5  | 1212 | 3    | A3H   | 01H   | 05H   |
| レバー受付コマンド6  | 1215 | 3    | A3H   | 02H   | 01H   |
| レバー受付コマンド7  | 1218 | 3    | A3H   | 02H   | 02H   |
| レバー受付コマンド8  | 1221 | 3    | A3H   | 03H   | 01H   |
| レバー受付コマンド9  | 1224 | 3    | A3H   | 03H   | 02H   |
| レバー受付コマンド10 | 1227 | 3    | A3H   | 04H   | 01H   |
| レバー受付コマンド11 | 1230 | 3    | A3H   | 04H   | 02H   |
| レバー受付コマンド12 | 1233 | 3    | A3H   | 04H   | 03H   |

(c) 左リール停止コマンド格納領域

| コマンドの名称     | 先頭番地 | バイト数 | 1バイト目 |
|-------------|------|------|-------|
| 左リール停止コマンド1 | 1300 | 1    | A4H   |

(d) 中リール停止コマンド格納領域

| コマンドの名称     | 先頭番地 | バイト数 | 1バイト目 |
|-------------|------|------|-------|
| 中リール停止コマンド1 | 1400 | 1    | A5H   |

(e) 右リール停止コマンド格納領域

| コマンドの名称   | 先頭番地 | バイト数 | 1バイト目 |
|-----------|------|------|-------|
| 右リールコマンド1 | 1500 | 1    | A6H   |

(f) 全リール停止コマンド格納領域

| コマンドの名称     | 先頭番地 | バイト数 | 1バイト目 | 2バイト目 |
|-------------|------|------|-------|-------|
| 全リール停止コマンド1 | 1600 | 2    | A7H   | 01H   |
| 全リール停止コマンド2 | 1602 | 2    | A7H   | 02H   |
| 全リール停止コマンド3 | 1604 | 2    | A7H   | 03H   |
| 全リール停止コマンド4 | 1606 | 2    | A7H   | 04H   |
| 全リール停止コマンド5 | 1608 | 2    | A7H   | 05H   |

## 【図 162】

| 命令                                       | バイト数 | ステート数 |
|------------------------------------------|------|-------|
| JP Z,mn<br>JP NZ,mn                      | 3    | 10    |
| JP SZ,mn<br>JP NSZ,mn                    | 4    | 14    |
| JR Z,e<br>JR NZ,e<br>JR SZ,e<br>JR NSZ,e | 2    | 8/7   |
| CALL Z,mn<br>CALL NZ,mn                  | 3    | 16/10 |
| CALL SZ,mn<br>CALL NSZ,mn                | 4    | 20/14 |
| RET Z<br>RET NZ<br>RET SZ<br>RET NSZ     | 1    | 10/4  |

## 【図 161】

| 命令                                                                                                                                          | 説明                                                                                                                                                                                             |
|---------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| WARU r,A,n<br>(17ステート、3バイト)<br>WARU Ar<br>(14ステート、2バイト)<br>WARU ss,HL<br>WARU HL,ss<br>(17ステート、2バイト)<br>[r=B, C, D, E, H, L]<br>[ss=BC, DE] | 最後のオペランドの値で最後から2番目のオペランドの値を除外し、最初のオペランドで指定されるレジスタに商を格納し、最初から2番目のオペランドで指定されるレジスタに余りを格納<br><br>Zフラグ・SZフラグは、余り無しの時に1、余り有りの時に0<br>Cフラグは、除数が0の時に1、除数が0以外の場合に0<br>除数が0のとき、商、余りともにFFFF(2バイトならFFFFFFh) |
| JP SZ,mn<br>(14ステート、4バイト)                                                                                                                   | SZフラグが1のとき、mnが示すアドレスにジャンプ<br>SZフラグが0のとき、処理を継続                                                                                                                                                  |
| JP NSZ,mn<br>(14ステート、4バイト)                                                                                                                  | SZフラグが0のとき、mnが示すアドレスにジャンプ<br>SZフラグが1のとき、処理を継続                                                                                                                                                  |
| JR SZ,e<br>(8/7ステート、2バイト)                                                                                                                   | SZフラグが1のとき、PC+eが示すアドレスにジャンプ<br>SZフラグが0のとき、処理を継続                                                                                                                                                |
| JR NSZ,e<br>(8/7ステート、2バイト)                                                                                                                  | SZフラグが0のとき、PC+eが示すアドレスにジャンプ<br>SZフラグが1のとき、処理を継続                                                                                                                                                |
| CALL SZ,mn<br>(20/14ステート、4バイト)                                                                                                              | SZフラグが1のとき、PCHを(SP-1)、PCLを(SP-2)にストアした後にmnにジャンプ<br>SP-2<br>SZフラグが0のとき、処理を継続                                                                                                                    |
| CALL NSZ,mn<br>(20/14ステート、4バイト)                                                                                                             | SZフラグが0のとき、PCHを(SP-1)、PCLを(SP-2)にストアした後にmnにジャンプ<br>SP-2<br>SZフラグが1のとき、処理を継続                                                                                                                    |
| RET SZ<br>(10/4ステート、1バイト)                                                                                                                   | SZフラグが1のとき、PCL←(SP)<br>PCH←(SP+1)<br>SP←SP+2<br>SZフラグが0のとき、処理を継続                                                                                                                               |
| RET NSZ<br>(10/4ステート、1バイト)                                                                                                                  | SZフラグが0のとき、PCL←(SP)<br>PCH←(SP+1)<br>SP←SP+2<br>SZフラグが1のとき、処理を継続                                                                                                                               |

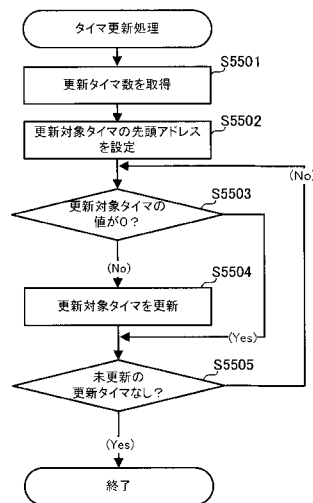
## 【図 163】

| ZフラグとSZフラグを同じ値に変化させる命令の例                                                                                                                                                                                    | 変化条件                    |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------|
| 16ビットADD命令・16ビットINC命令・16ビットDEC命令を除く<br>算術演算命令<br>(8ビットADD命令・ADDTWOONE命令・ADC命令・SUB命令・SBC命令・AND命令・OR命令・XOR命令・CP命令・8ビットINC命令・8ビットDEC命令)<br>RLCA命令・RLA命令・RRCA命令・RRA命令を除く<br>ローテートシフト命令<br>「LD A,I」命令、「LD A,R」命令 | 第1オペランドの値が0のとき1、0以外の場合0 |
| WARU命令                                                                                                                                                                                                      | 第2オペランドの値が0のとき1、0以外の場合0 |
| INI命令・INIR命令・IND命令・INDR命令<br>OUTI命令・OTIR命令・OUTD命令・OTDR命令                                                                                                                                                    | Bレジスタの値が0のとき1、0以外の場合0   |

| Zフラグを変化させずSZフラグを変化させる命令の例                                                             | 変化条件                                      |
|---------------------------------------------------------------------------------------|-------------------------------------------|
| 「LD A,I」命令、「LD A,R」命令を除くロード命令<br>16ビットADD命令<br>16ビットINC命令<br>16ビットDEC命令<br>即値を入れるIN命令 | 第1オペランドの値が0のとき1、0以外の場合0                   |
| INCTENSOU命令<br>(例: INCTENSOU AC,(HL))<br>TENSOUINC命令<br>(例: TENSOUINC AC,(HL))        | 第1オペランド(例の場合はAレジスタとCレジスタ)の値が0のとき1、0以外の場合0 |
| RLCA命令<br>RLA命令<br>RRCA命令<br>RRA命令                                                    | Aレジスタの値が0のとき1、0以外の場合0                     |

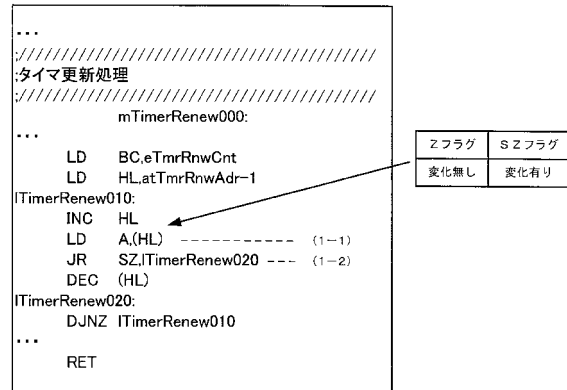
| Zフラグを変化させSZフラグを変化させない命令の例          | 変化条件                  |
|------------------------------------|-----------------------|
| CPI命令<br>CPJR命令<br>CPD命令<br>CPDR命令 | Aレジスタの値が0のとき1、0以外の場合0 |

【図 164】

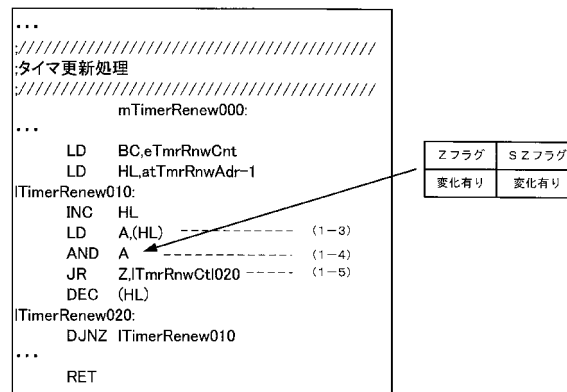


【図 165】

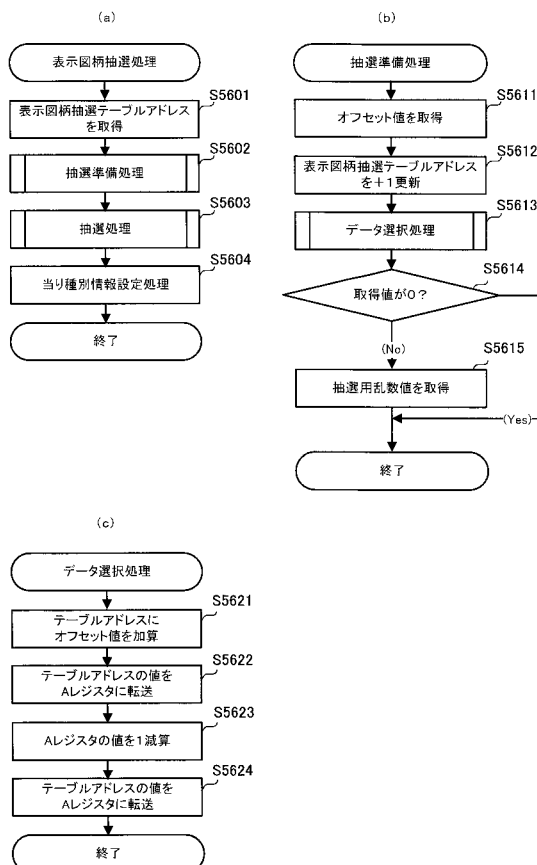
(a) 本発明のタイマ更新処理



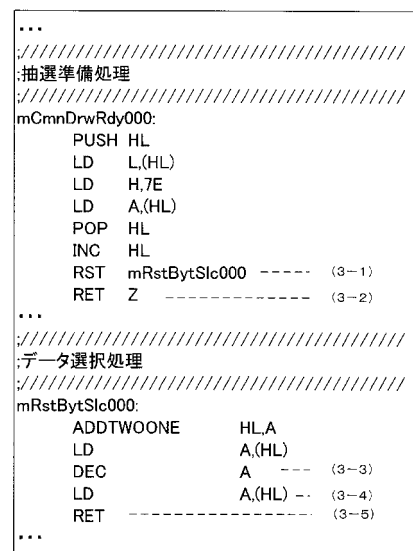
(b) 従来のタイマ更新処理



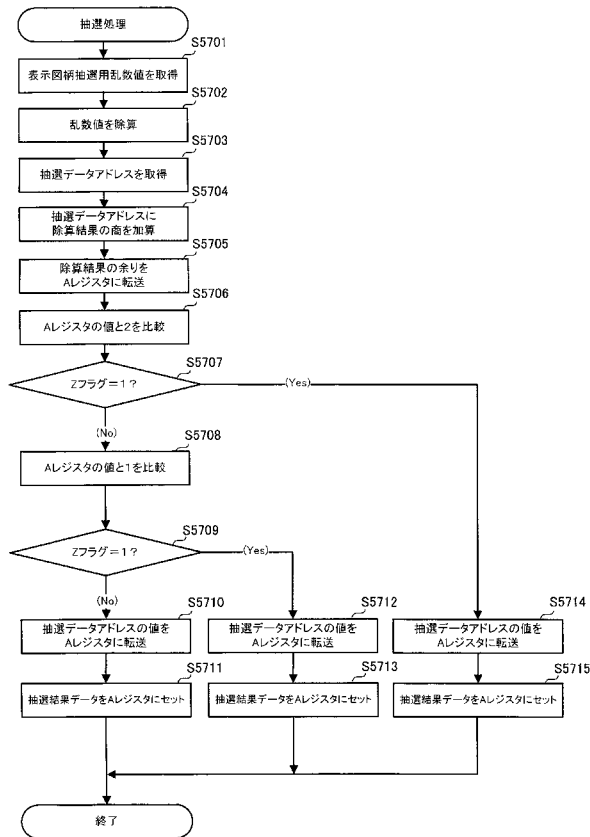
【図 166】



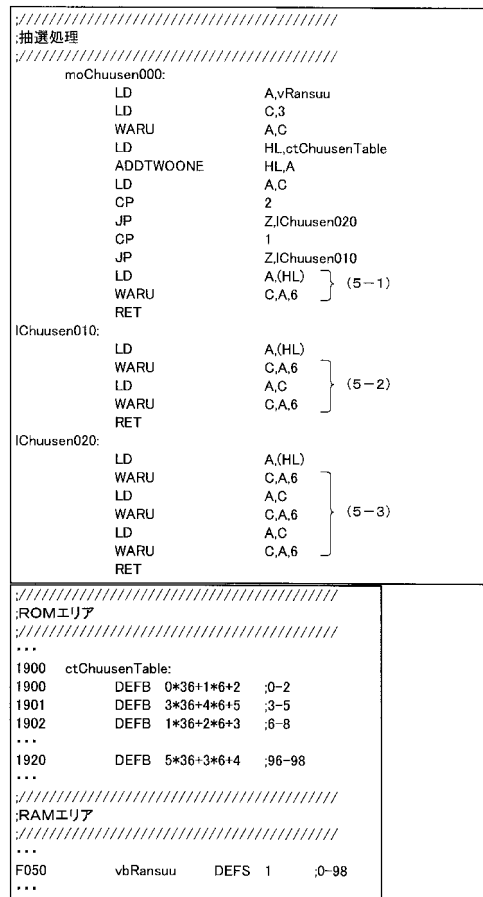
【図 167】



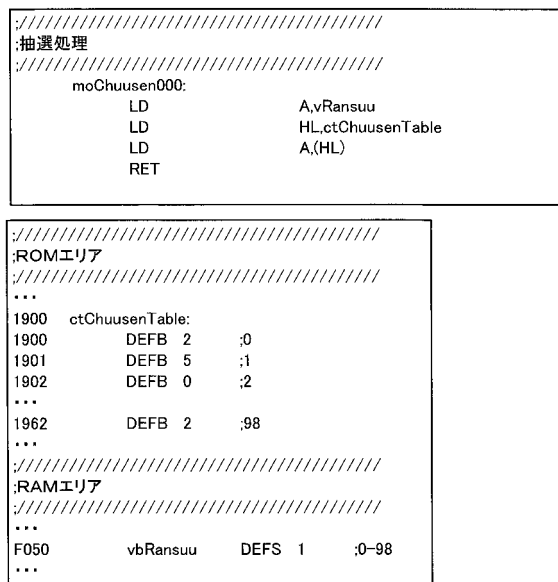
【図 168】



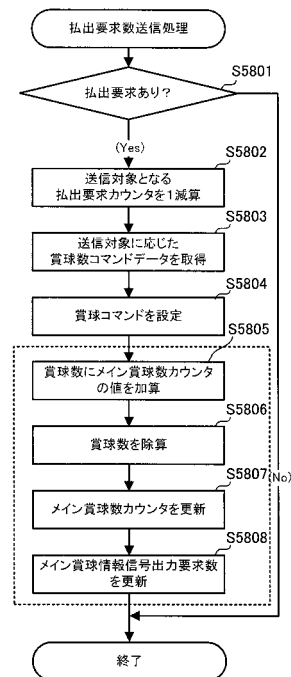
【図 169】



【図 170】



【図 171】





【図 172】

```

////////////////////////////////////
;払出要求数送信処理
////////////////////////////////////
mSCUPayCmd000:
...
    AND    0FH
    LD     HL,vSigPayCnt  ----- (5-1)
    ADD    A,(HL)         ----- (5-2)
    WARU   C,A,10         ----- (5-3)
    LD     (HL),A         ----- (5-4)
    INC    HL             ----- (5-5)
    LD     A,C            ----- (5-6)
    ADD    A,(HL)         ----- (5-7)
    LD     (HL),A         ----- (5-8)
    RET

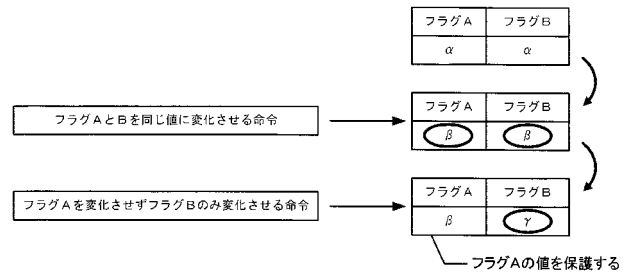
```

```

////////////////////////////////////
;RAMエリア
////////////////////////////////////
...
F060      vSigPayCnt:      DEFS  1
F061      vSigPayReq:      DEFS  1
...

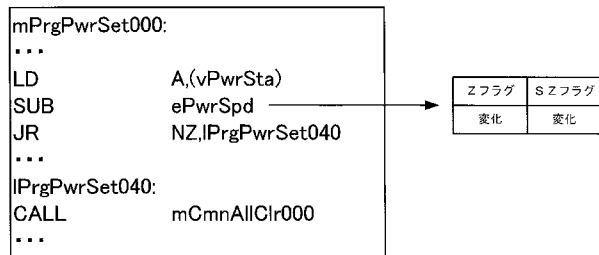
```

【図 173】

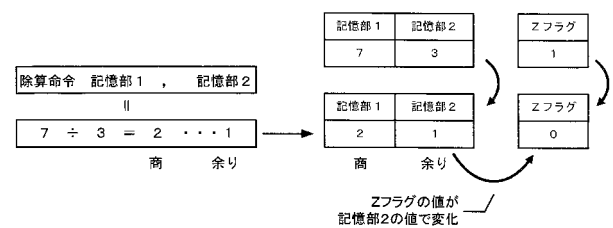


【図 174】

(a) 電源ステータスがサスペンドか否か判定する処理

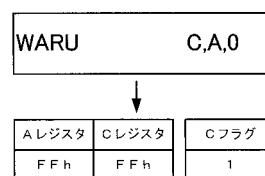


【図 175】

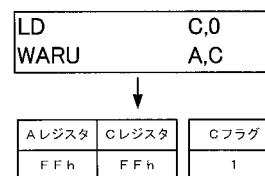


【図 176】

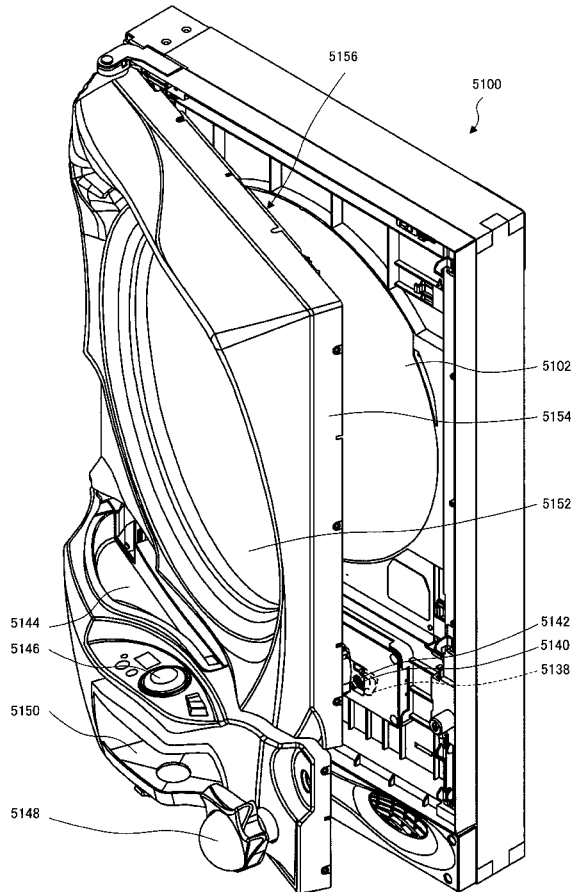
(a) WARU 命令の他の例 1



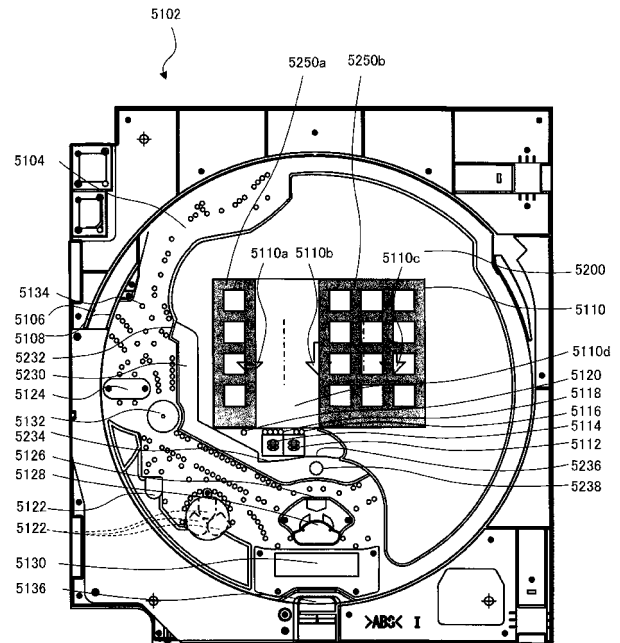
(b) WARU 命令の他の例 2



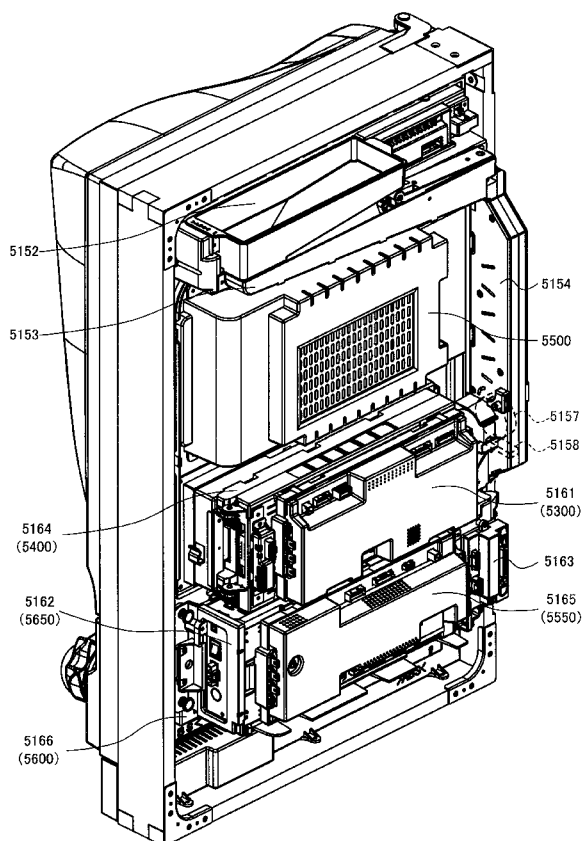
【図 177】



【図 178】

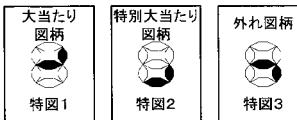


【図 179】



【図 180】

(a) 特別図柄



(b) 装飾図柄



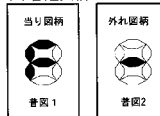
大当たりとなる図柄組合せ



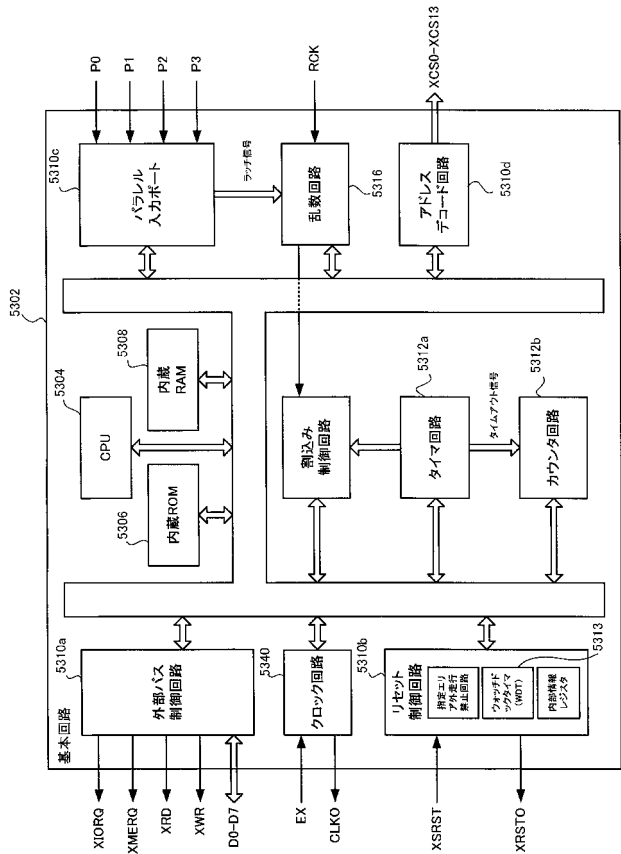
特別大当たりとなる図柄組合せ



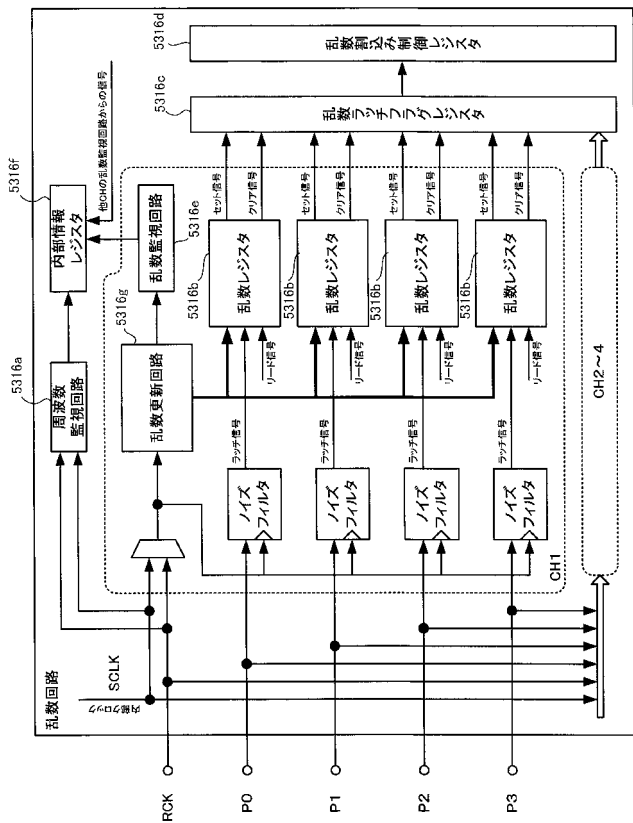
(c) 普通図柄



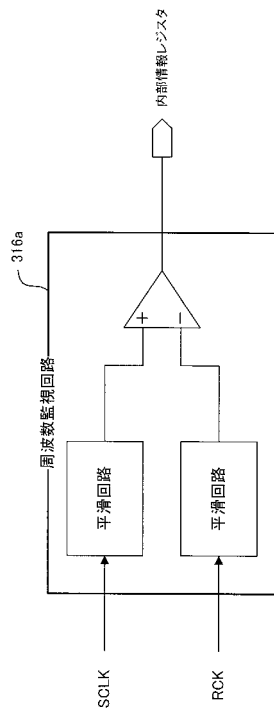
【图 1 8 2】



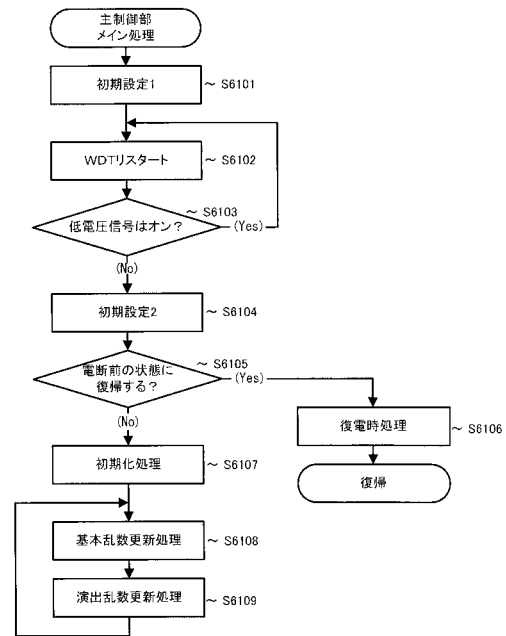
【図 1 8 4】



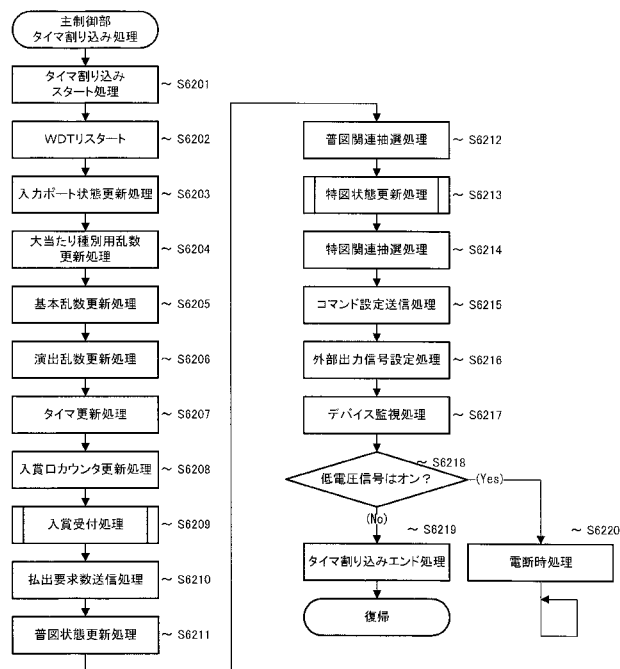
【図185】



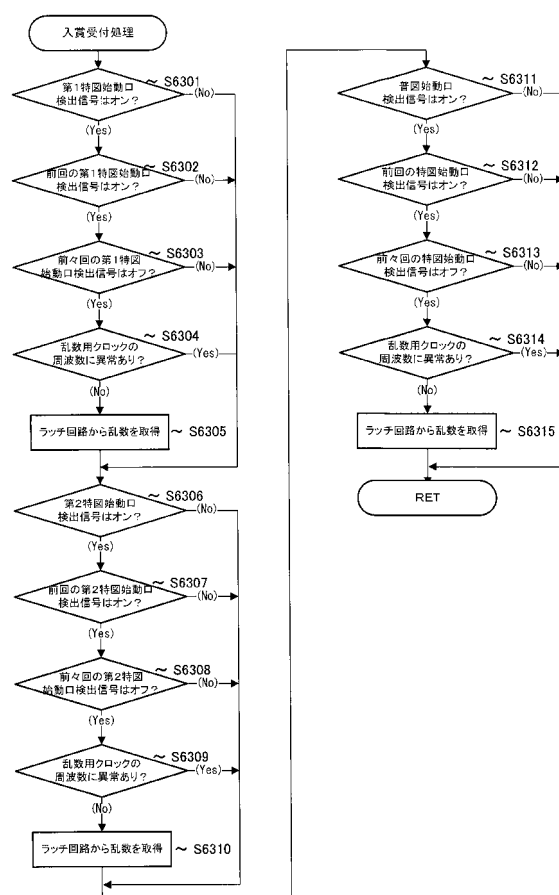
【図186】



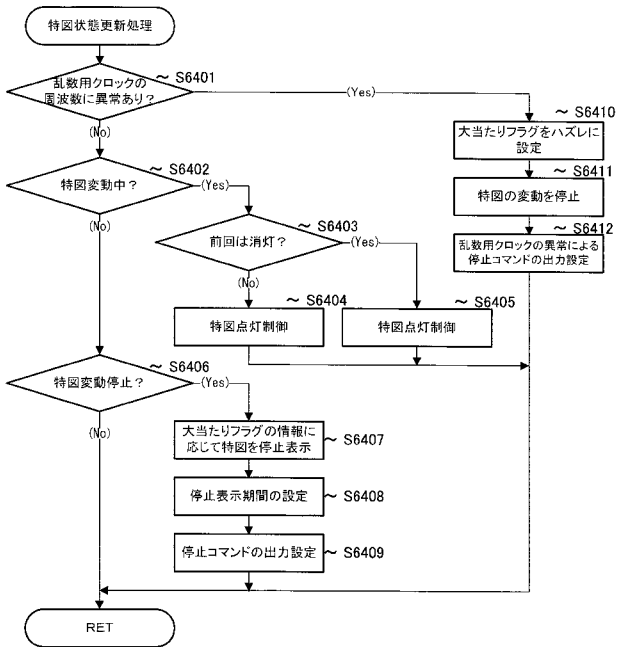
【図187】



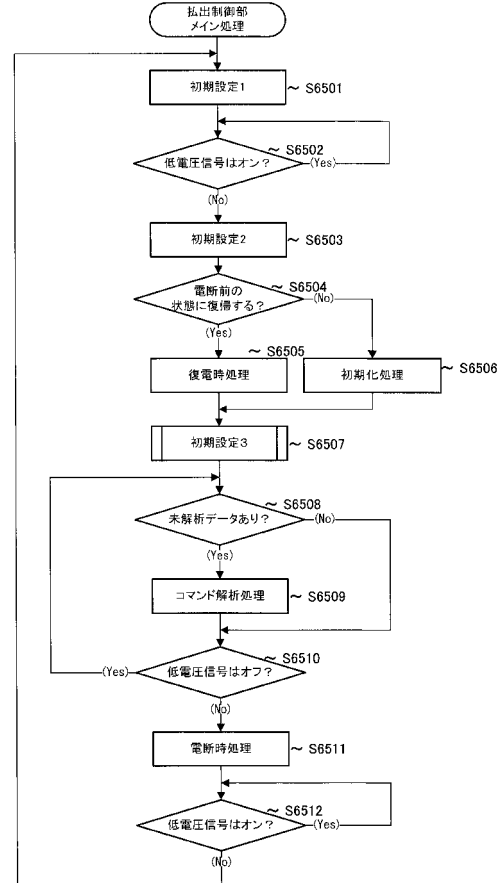
【図188】



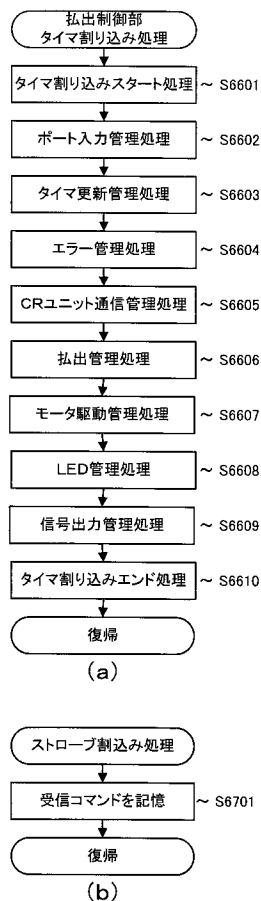
【図189】



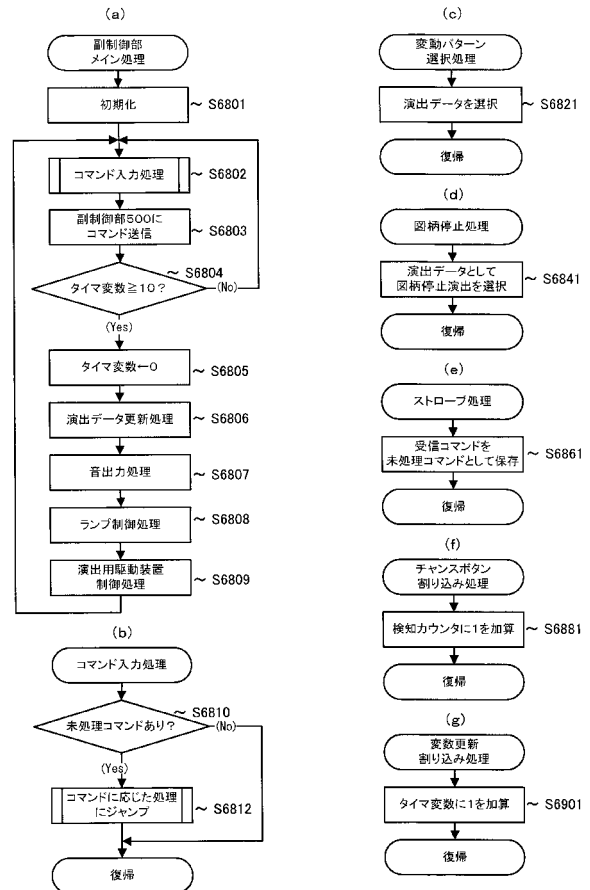
【図190】



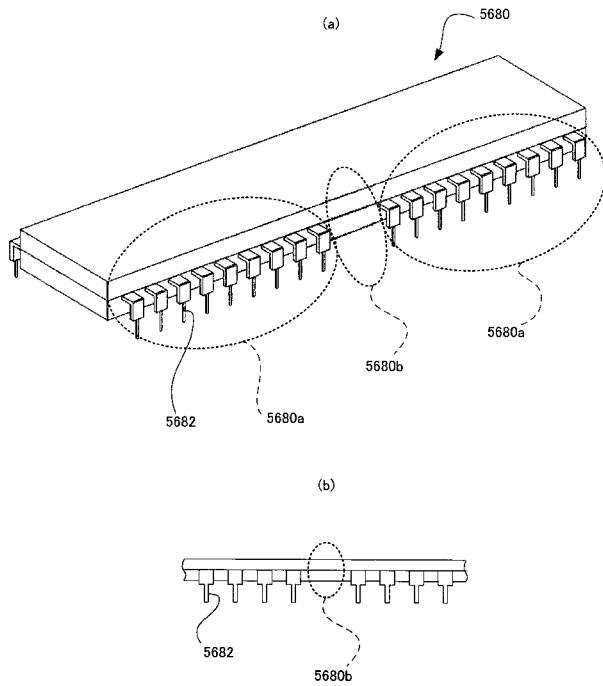
【図191】



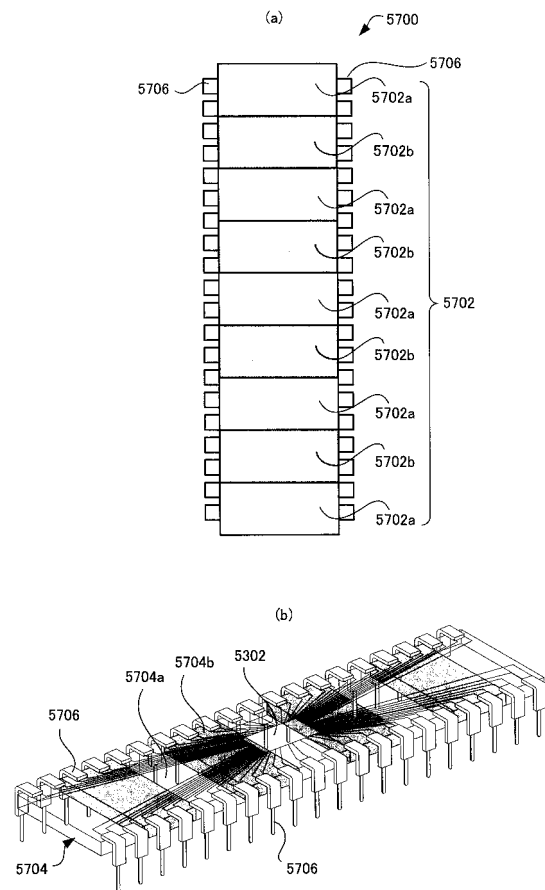
【図192】



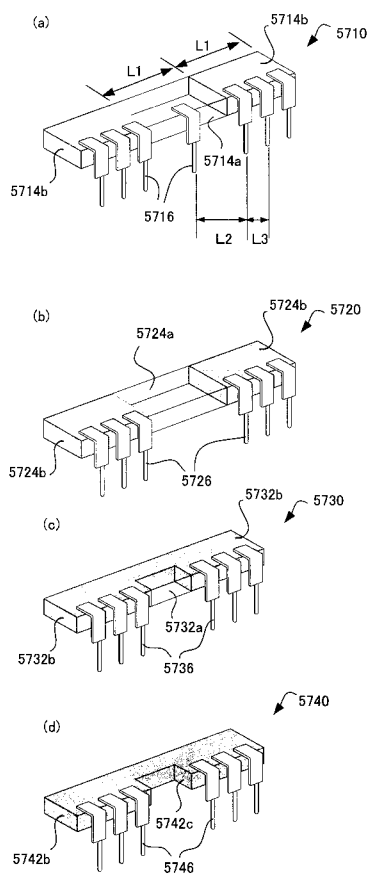
【図 193】



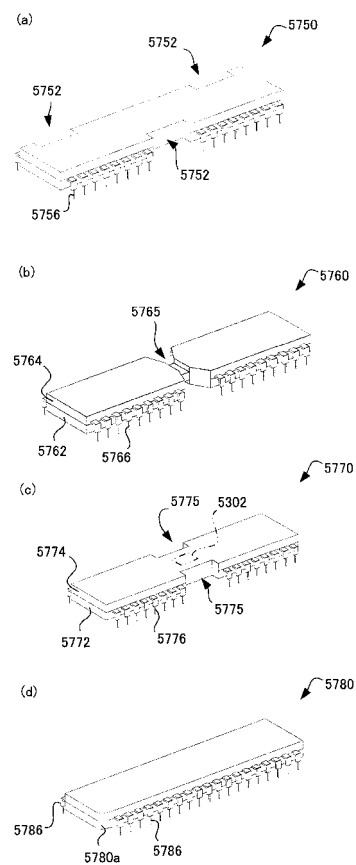
【図 194】



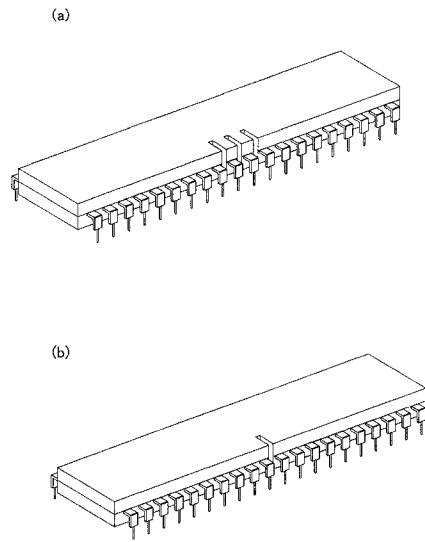
【図 195】



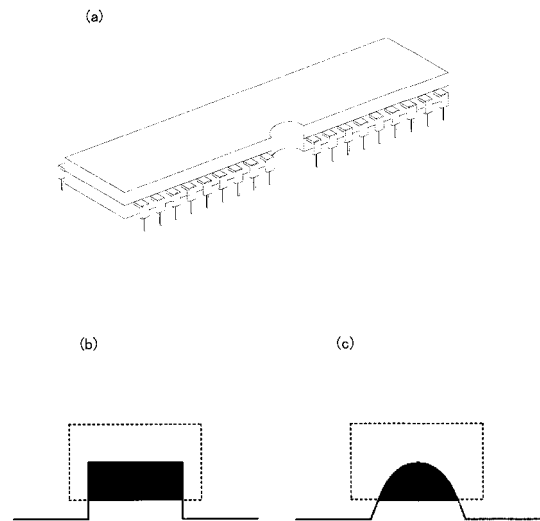
【図 196】



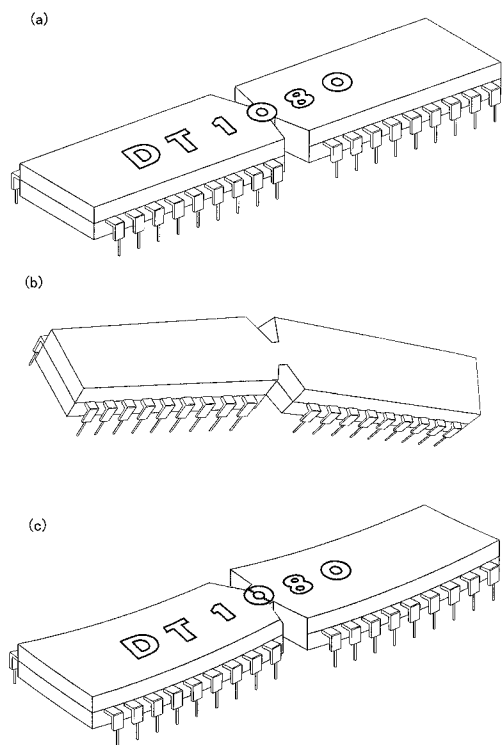
【図 197】



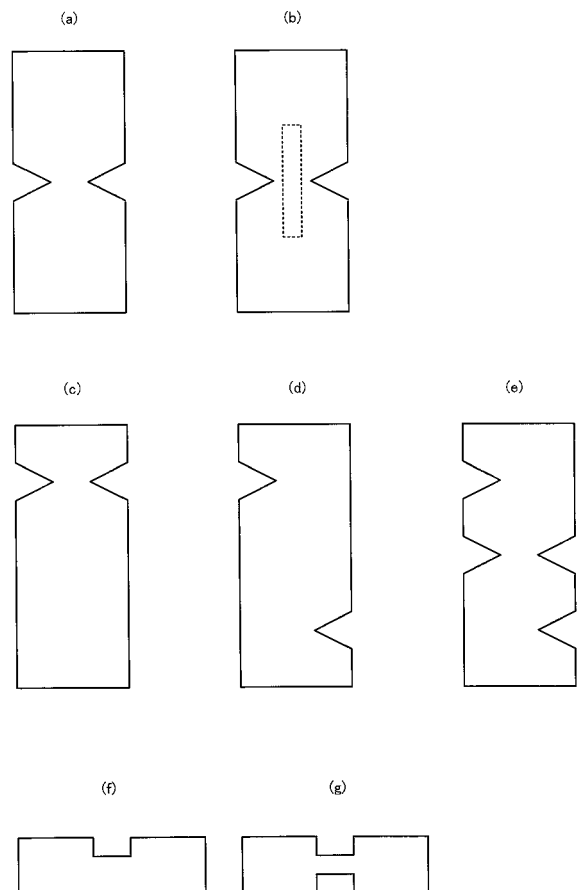
【図 198】



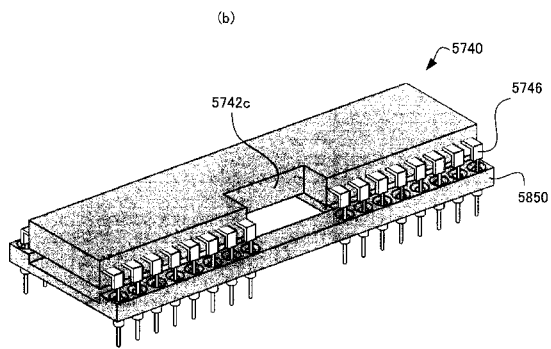
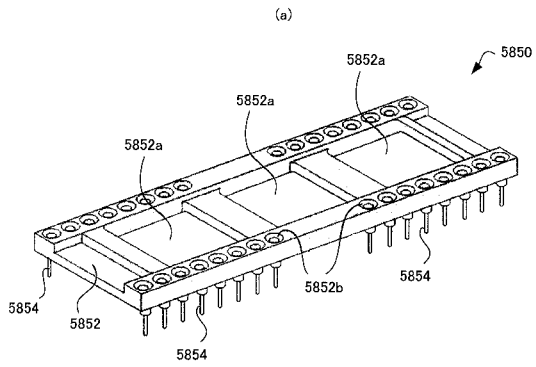
【図 199】



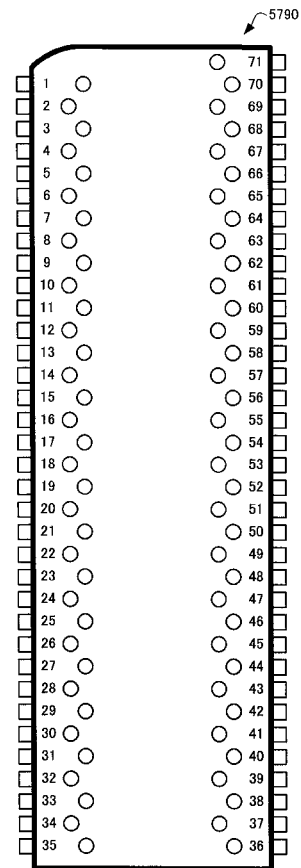
【図 200】



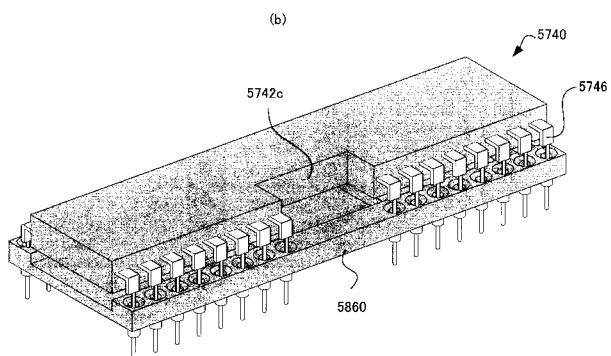
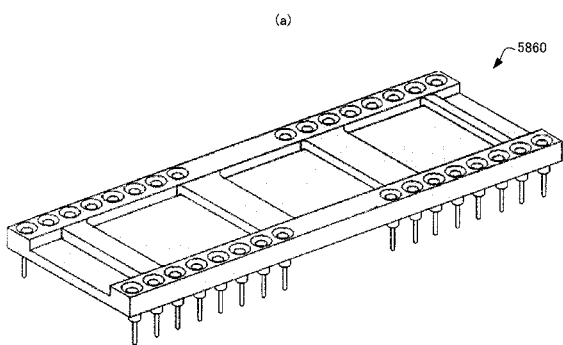
【図 201】



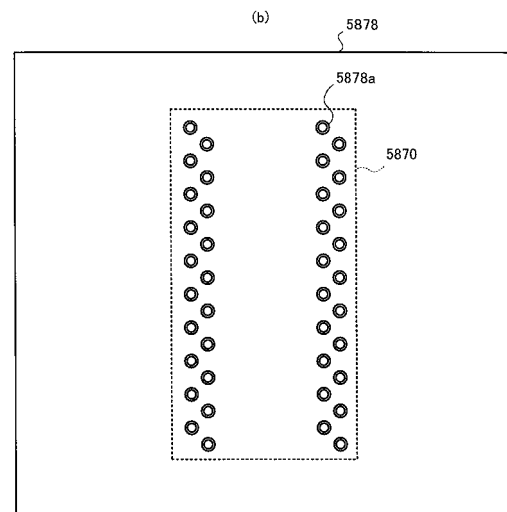
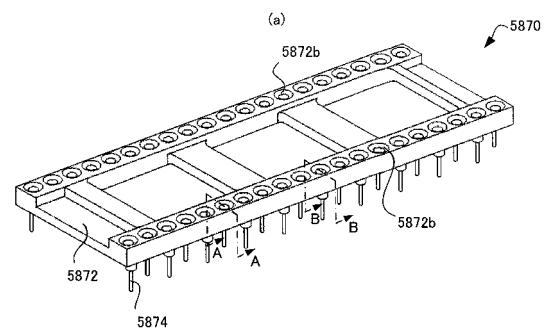
【図 202】



【図 203】

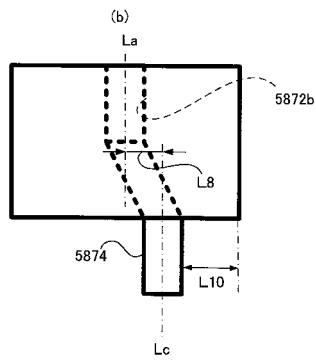
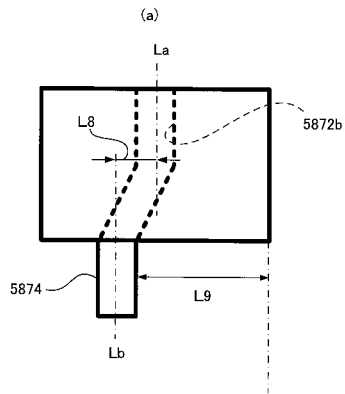


【図 204】

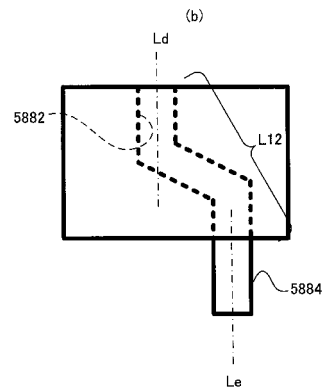
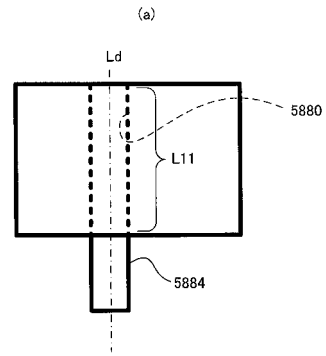




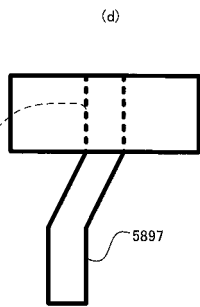
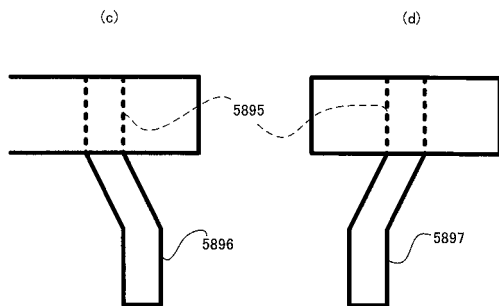
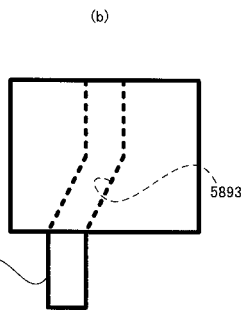
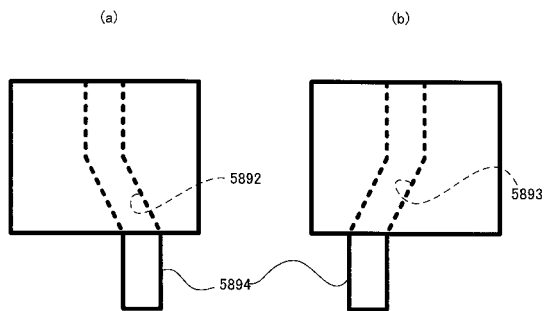
【図 205】



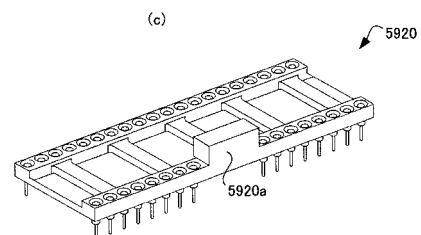
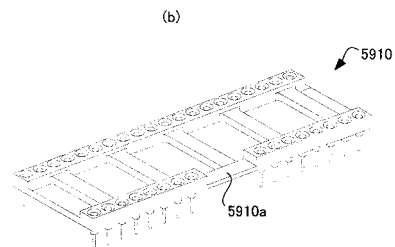
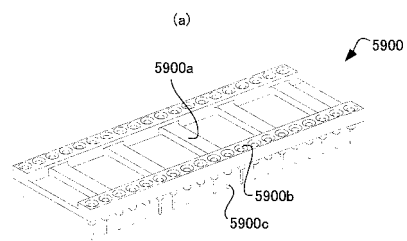
【図 206】



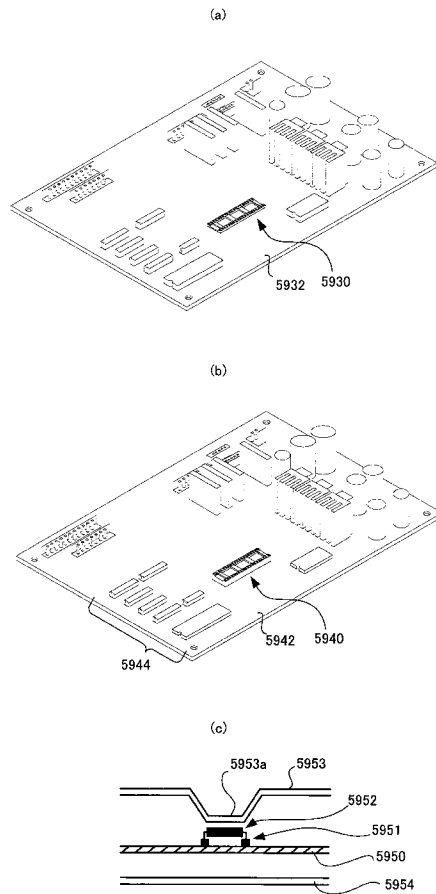
【図 207】



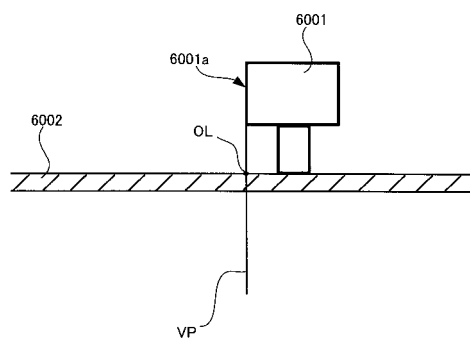
【図 208】



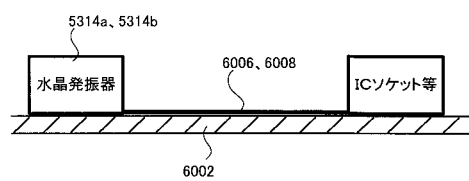
【図 2 0 9】



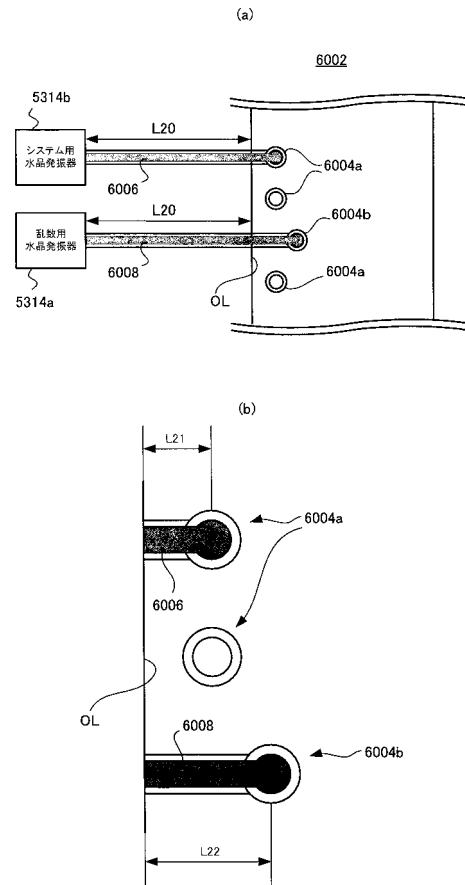
【図 2 1 1】



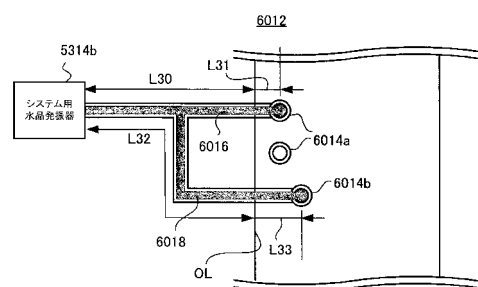
【図 2 1 2】



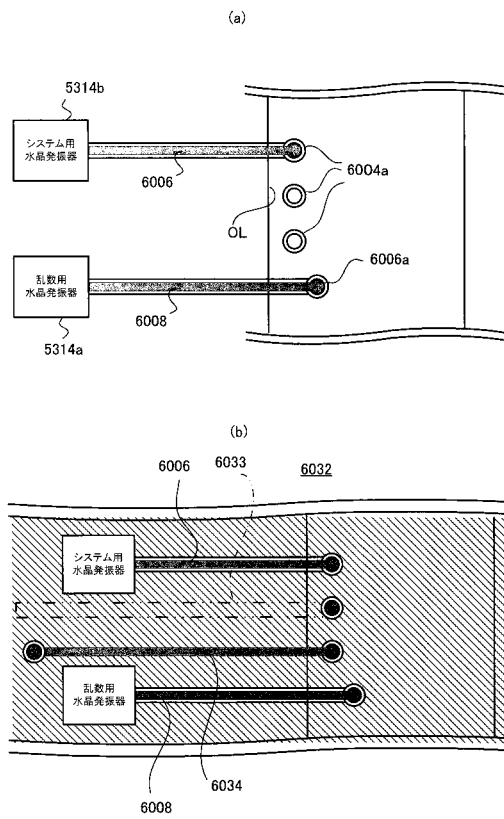
【図 2 1 0】



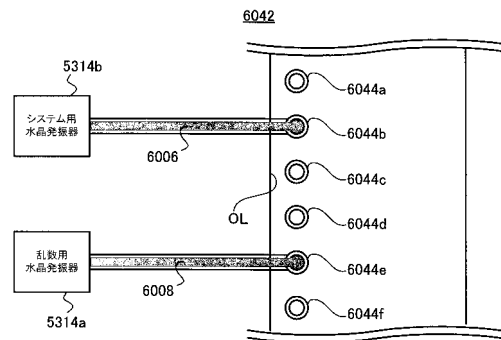
【図 2 1 3】



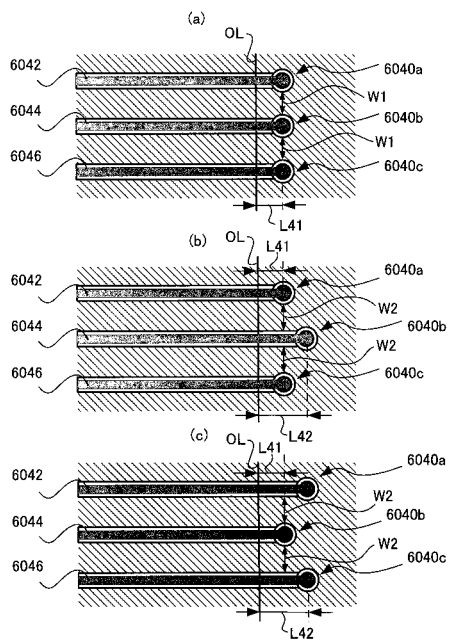
【図 2 1 4】



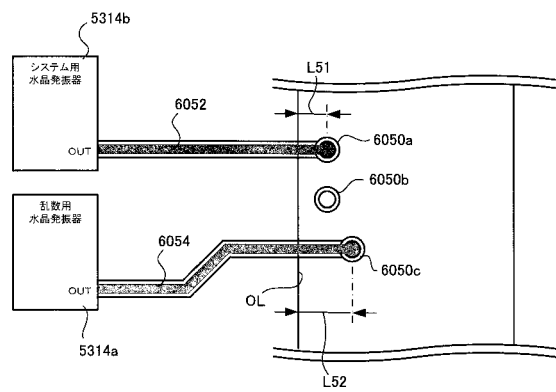
【図 2 1 5】



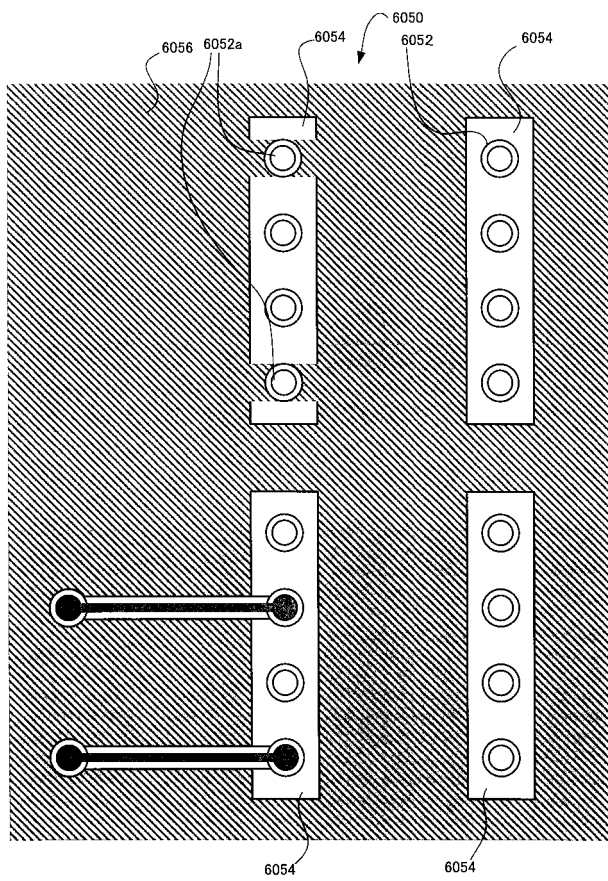
【図 2 1 6】



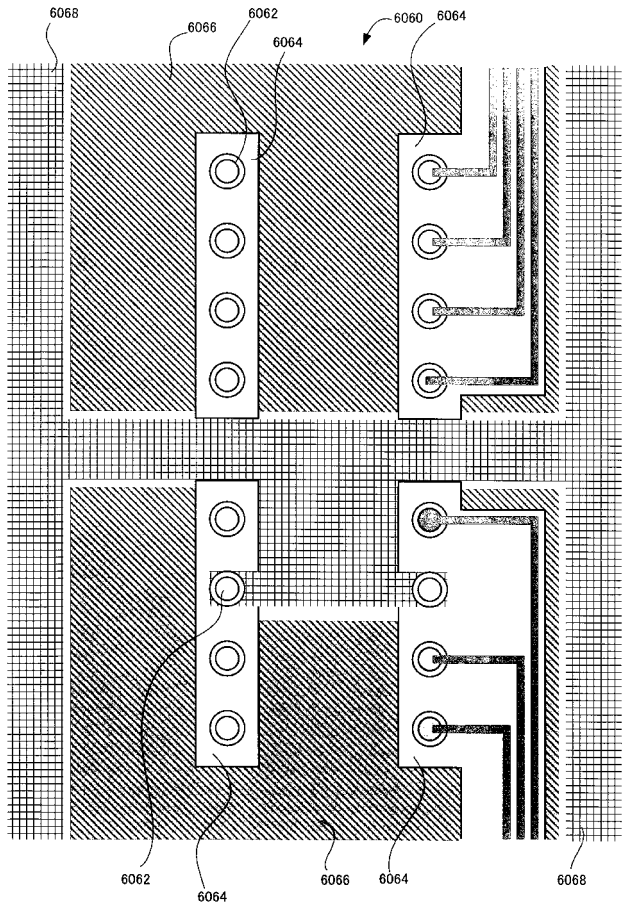
【図 2 1 7】



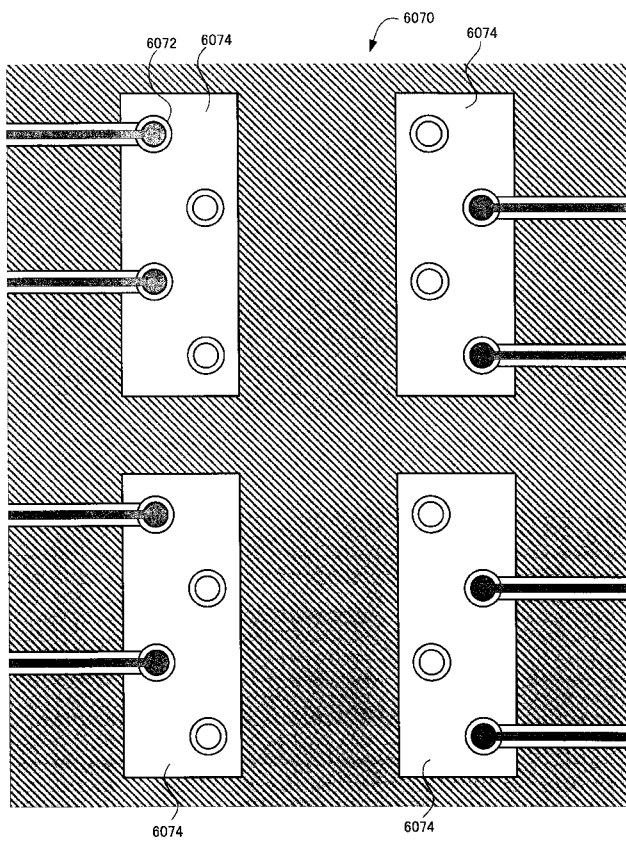
【図 2 1 8】



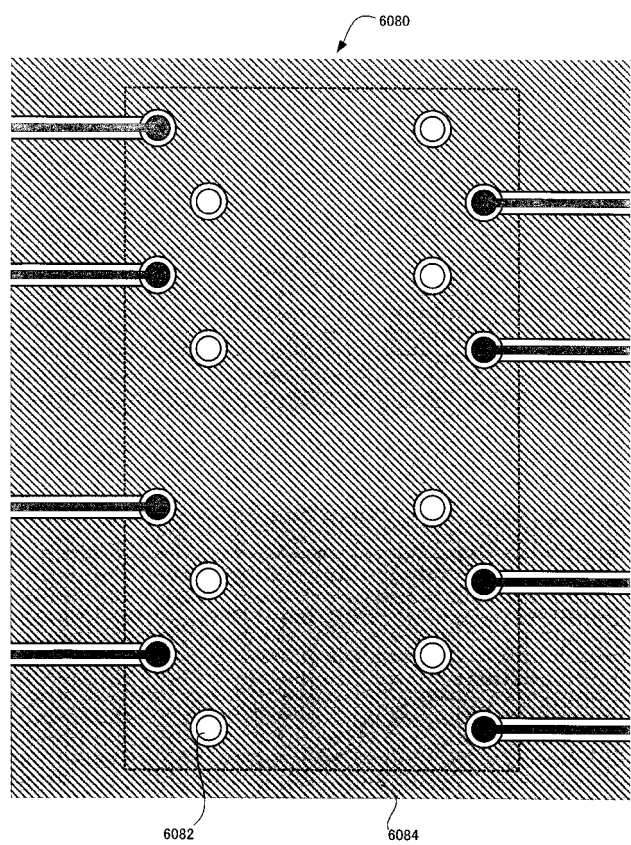
【図 2 1 9】



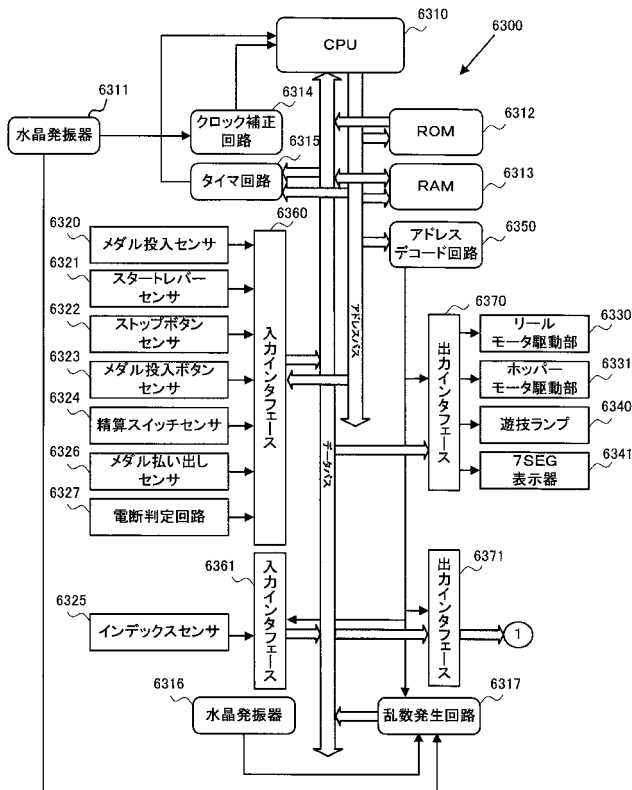
【図 2 2 0】



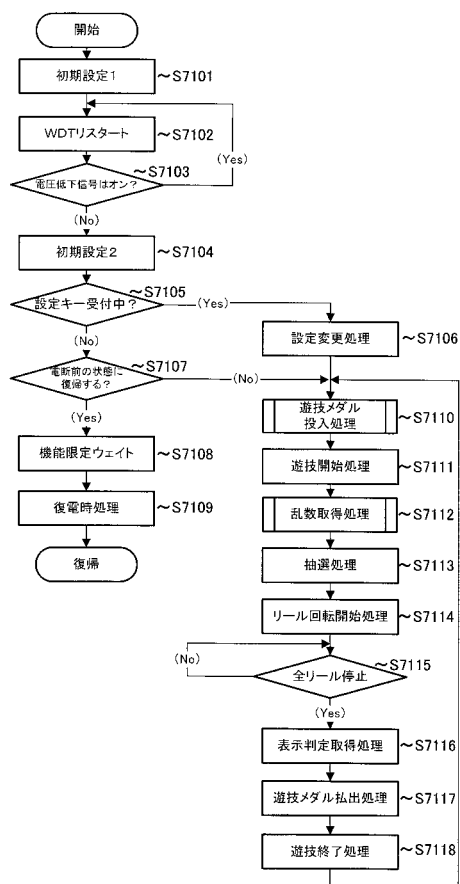
【図 2 2 1】



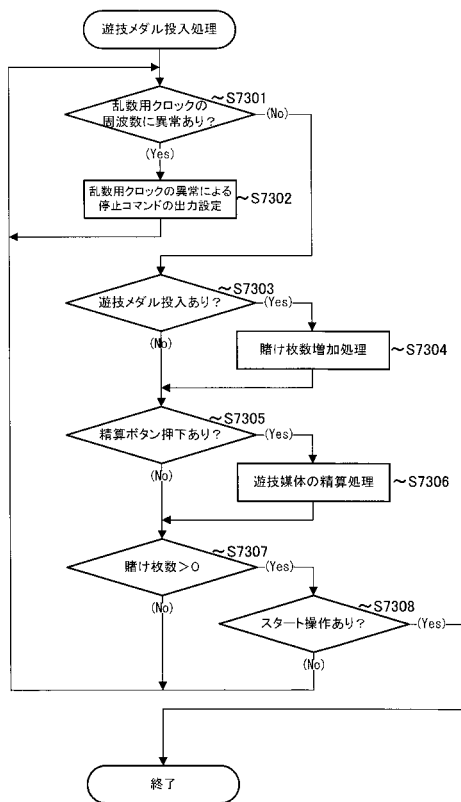
【 図 2 2 3 】



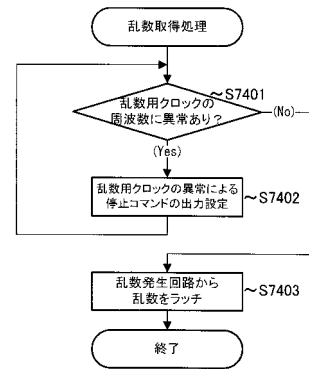
【图 2 2 5】



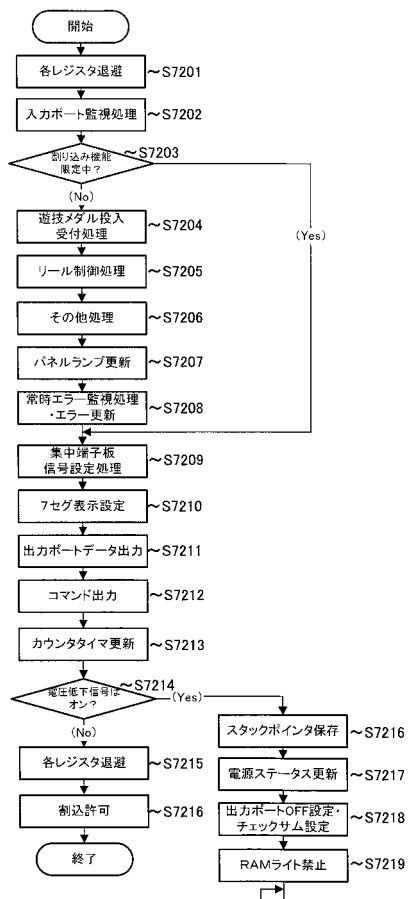
【図 2 2 6】



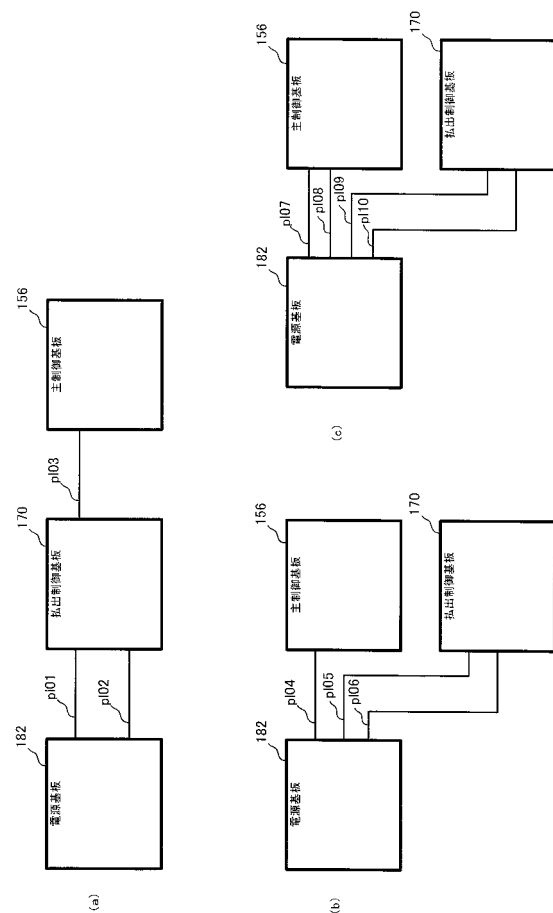
【図 2 2 7】



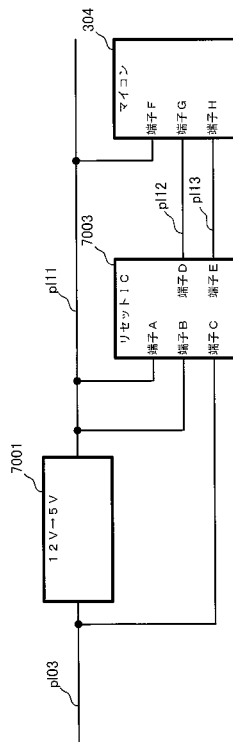
【図 2 2 8】



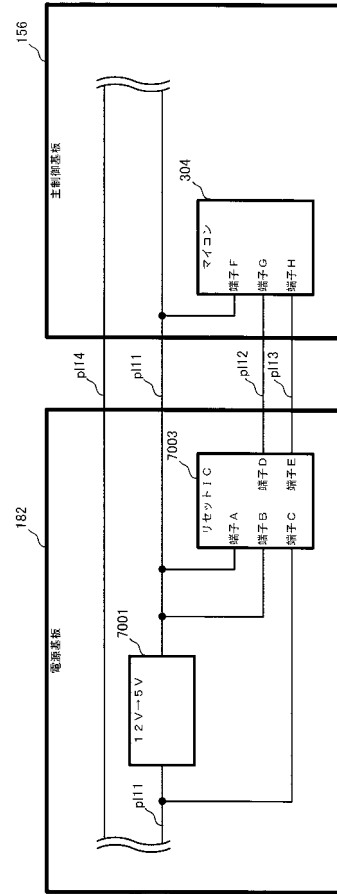
【図 2 2 9】



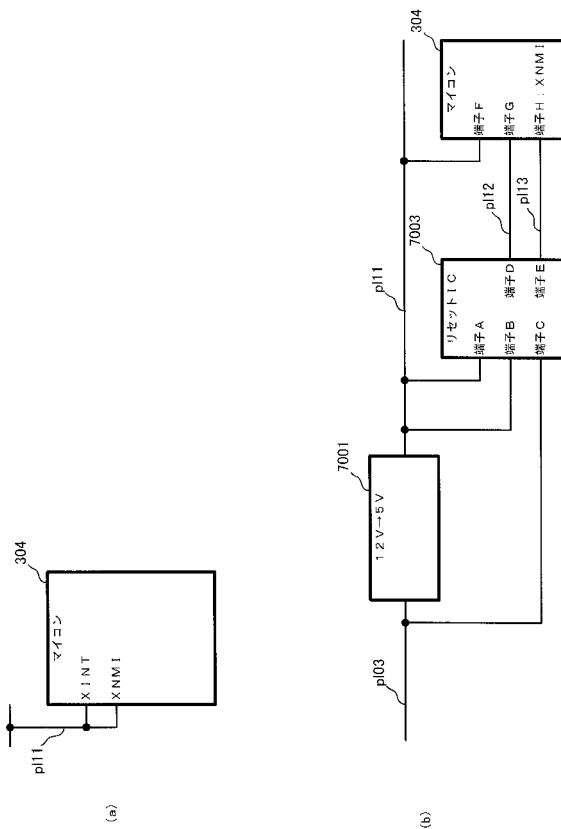
【 図 2 3 0 】



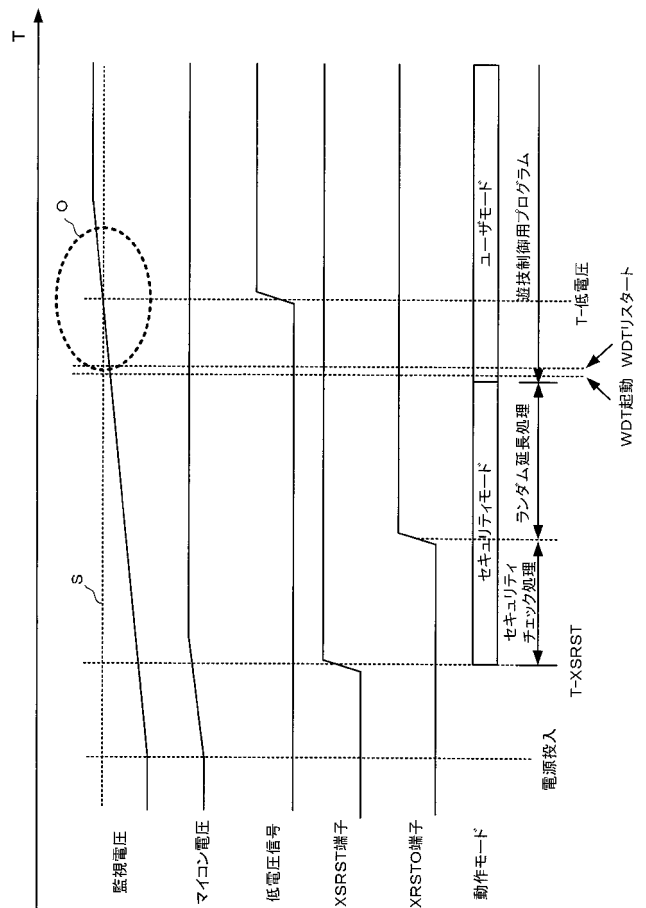
【图 2 3 1】



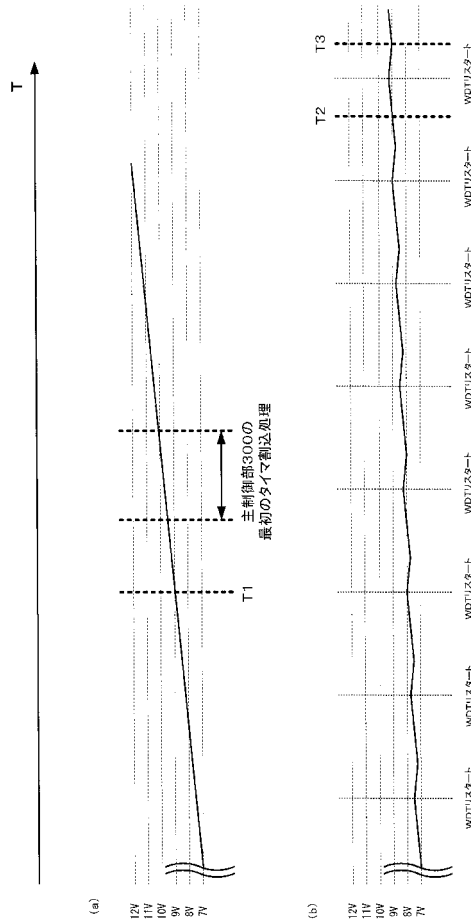
【 図 2 3 2 】



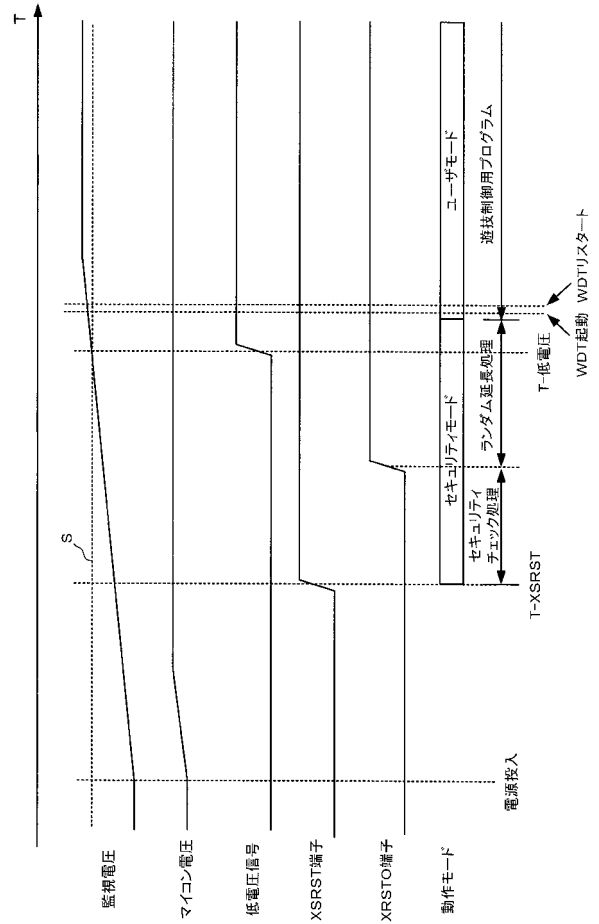
【图 2 3 3】



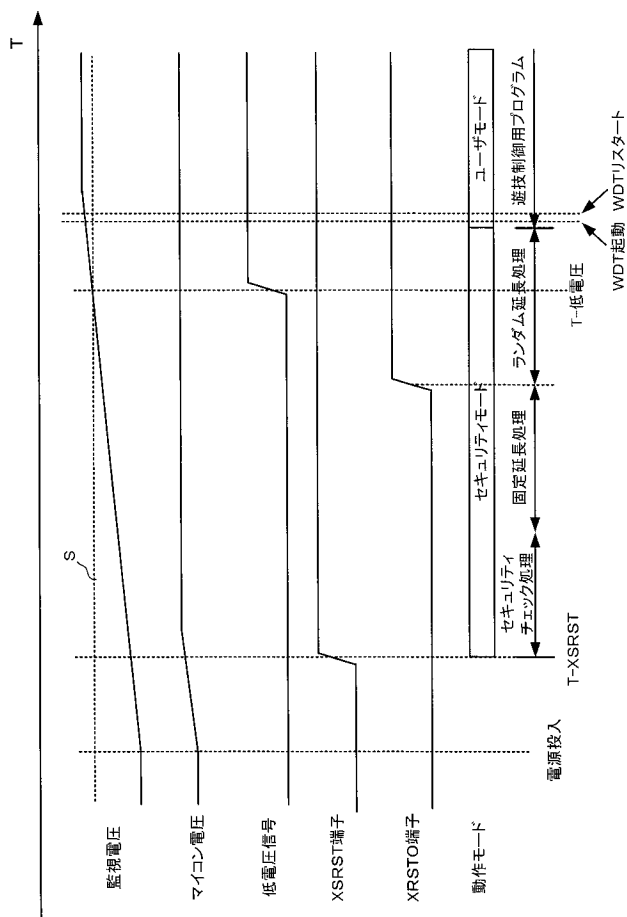
【図 234】



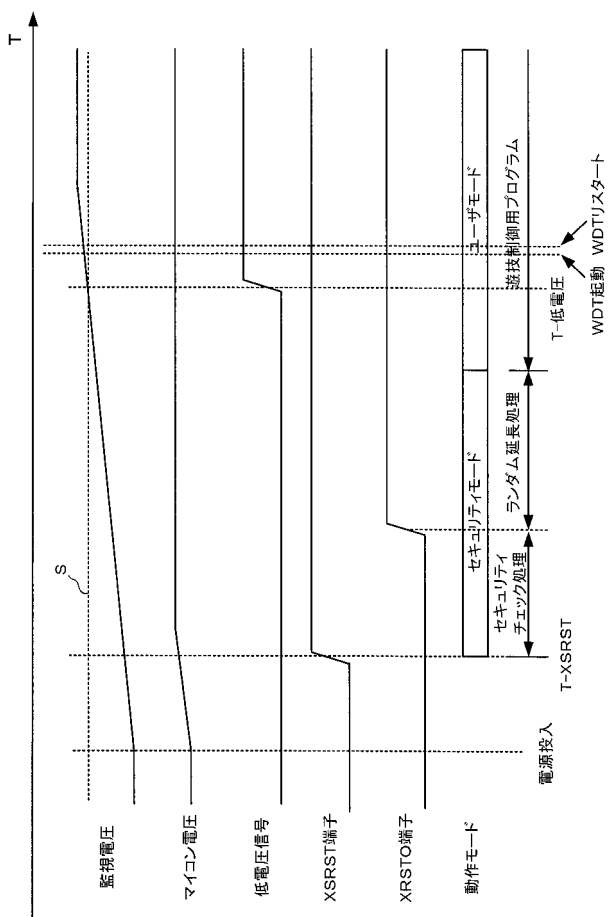
【図 235】



【図 236】

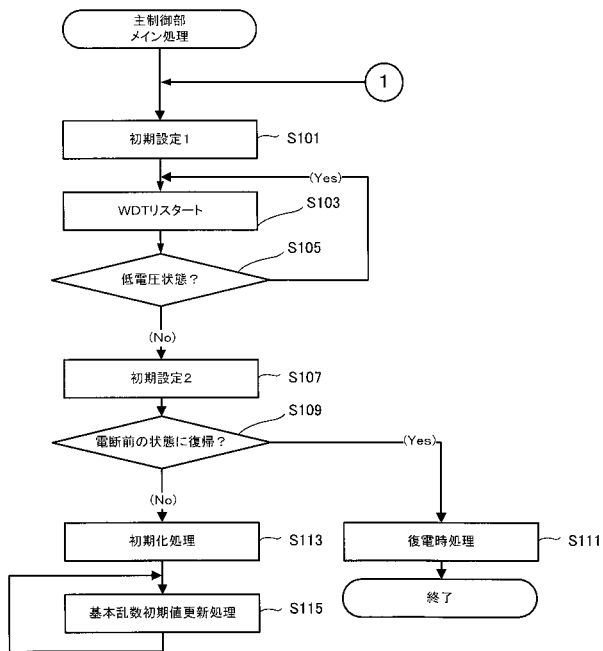


【図 237】

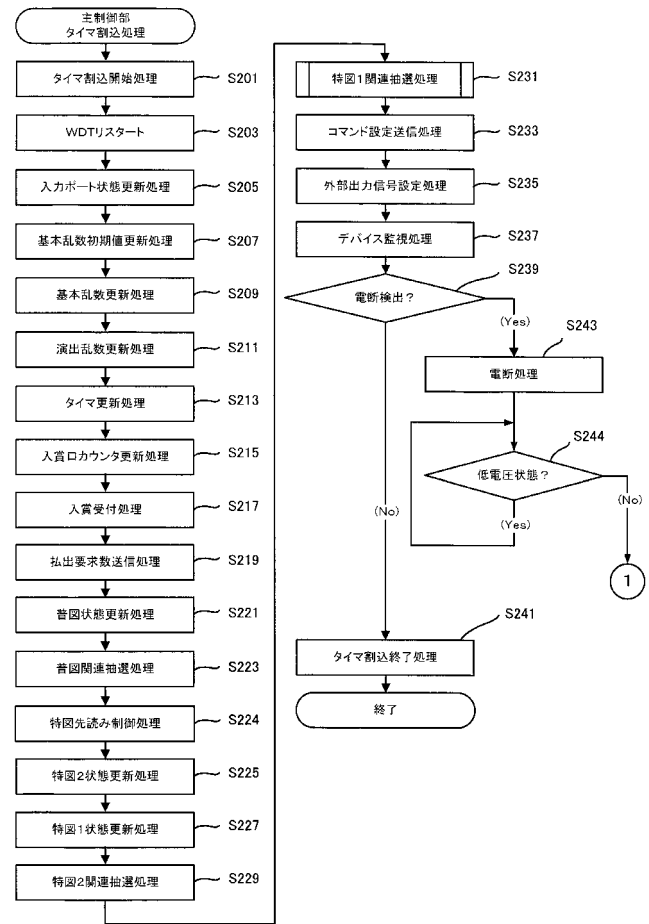




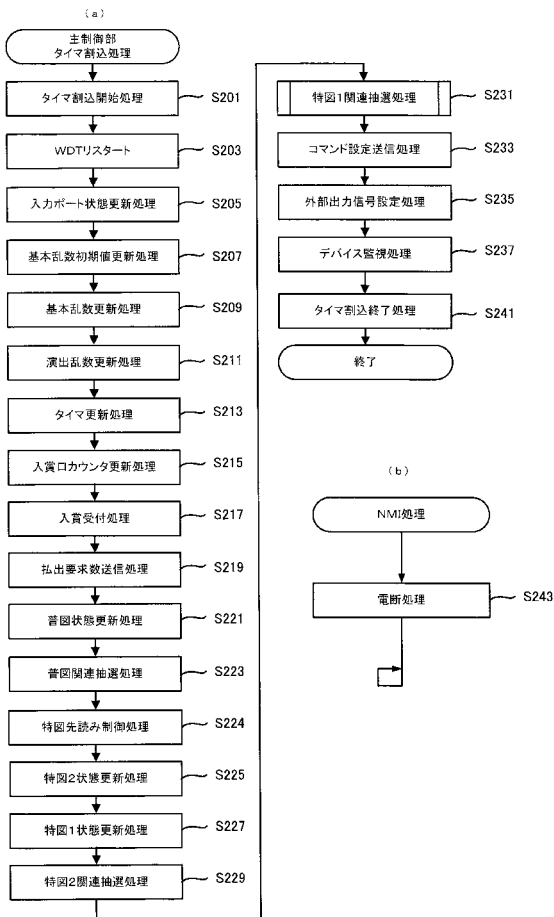
【図 238】



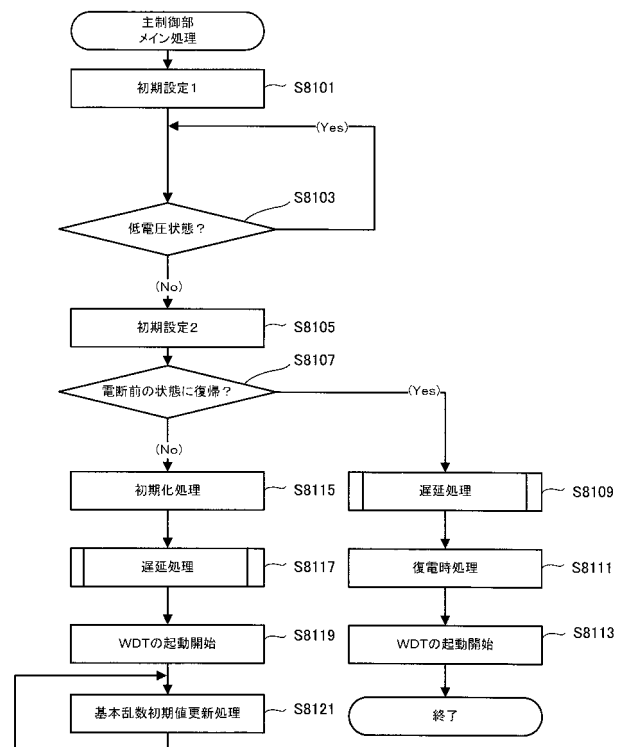
【図 239】



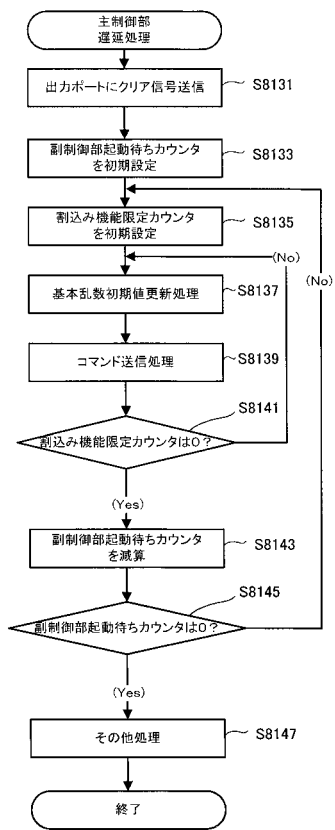
【図 240】



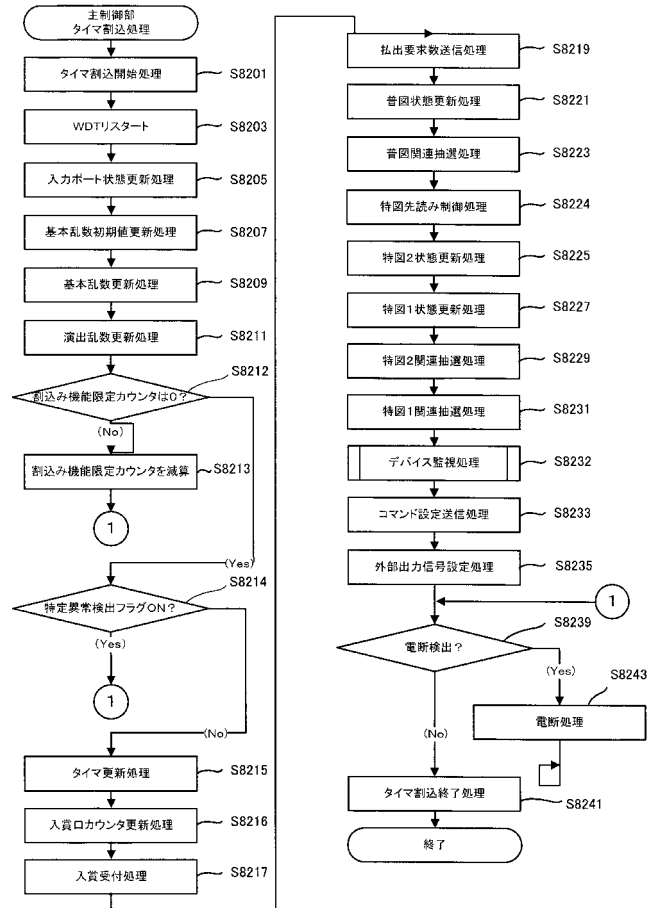
【図 241】



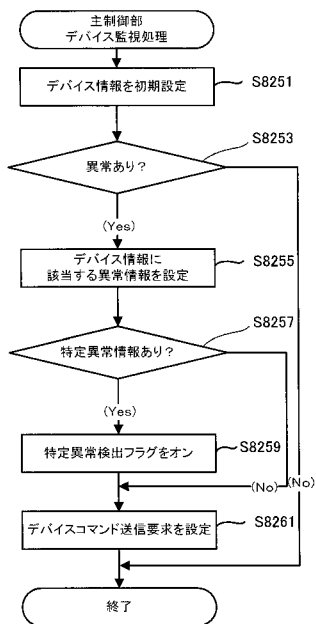
【図 2 4 2】



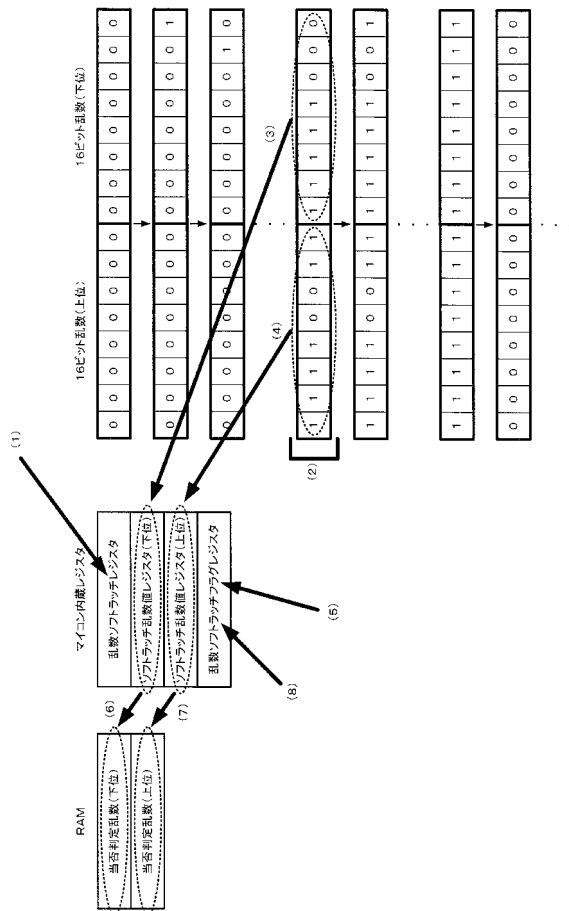
【図 2 4 3】



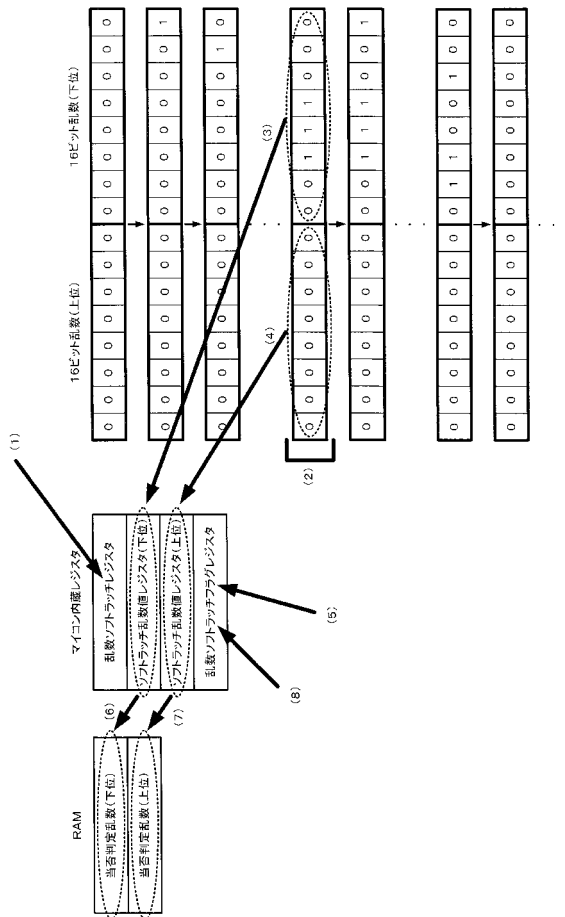
【図 2 4 4】



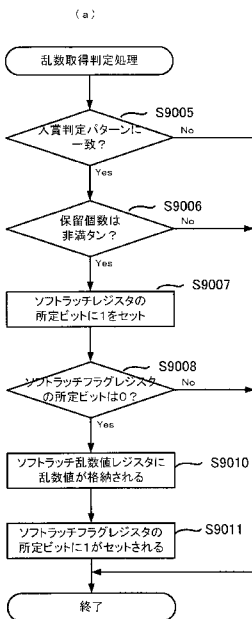
【図 2 4 5】



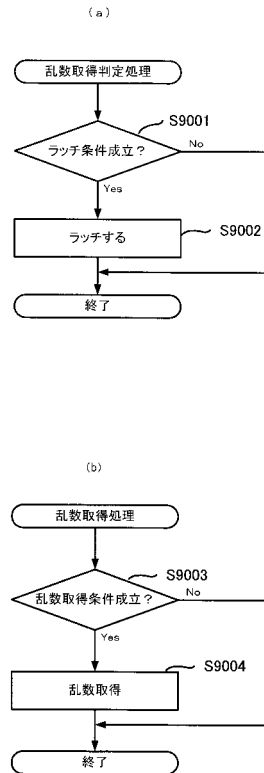
【図 2 4 6】



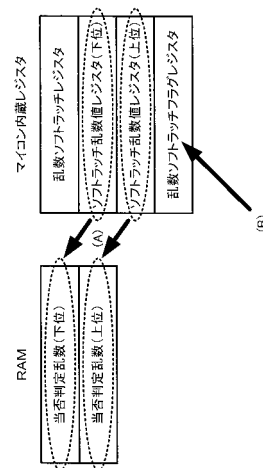
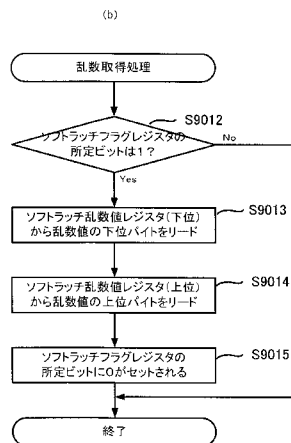
【図 2 4 8】



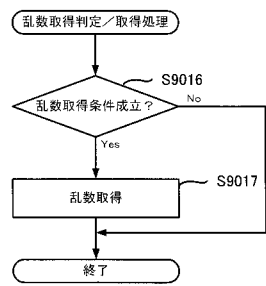
【図 2 4 7】



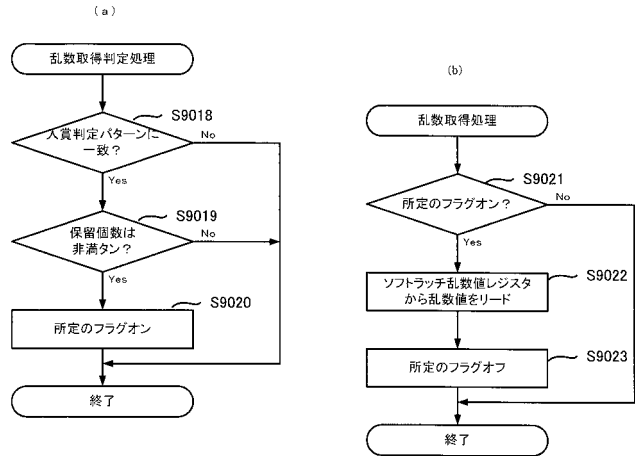
【図 2 4 9】



【図 250】



【図 251】



【図 252】

```
;;内蔵レジスタ
RANLAT EQU 2030H ; 乱数ソフトラッチレジスタ
RANLAT_FLG EQU 2031H ; 乱数ソフトラッチフラグレジスタ
RANSU_REGO_L EQU 2032H ; ソフトラッチ乱数値レジスタ(下位)
RANSU_REGO_H EQU 2033H ; ソフトラッチ乱数値レジスタ(上位)

RAMIUT
F080 RANSUO_L DEFS 1
F081 RANSUO_H DEFS 1

ROMIUT
LD (RANLAT), 01H ; 乱数ソフトラッチレジスタに「1」をセット
LD A, (RANLAT_FLG) ; 乱数ソフトラッチフラグレジスタの値を取得
AND 01H ; 取得した乱数ソフトラッチフラグレジスタの値を「1」と比較
NZ, moRNDGET ; ラッチフラグレジスタの値が「1」でなければ、下方のモジュールにジャンプ

moRNDGET:
LD HL, RANSUO_L ; ソフトラッチ乱数値レジスタ(下位)のアドレスを取得
LD A, (HL) ; ソフトラッチ乱数値レジスタ(下位)の値を取得
LD (RANSUO_L), A ; ソフトラッチ乱数値レジスタ(下位)の値をRANSUに読み込む
LD HL, RANSUO_H ; ソフトラッチ乱数値レジスタ(上位)のアドレスを取得
LD A, (HL) ; ソフトラッチ乱数値レジスタ(上位)の値を取得
LD (RANSUO_H), A ; ソフトラッチ乱数値レジスタ(上位)の値をRANSUに読み込む
```

【図 253】

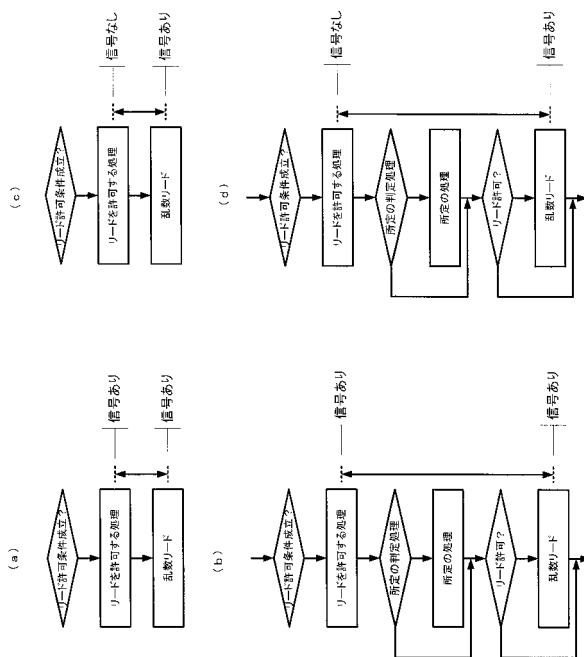
```
;;内蔵レジスタ
RANLAT EQU 2030H
RANLAT_FLG EQU 2031H
RANSU_REGO_L EQU 2032H
RANSU_REGO_H EQU 2033H

RAMIUT
F080 RANSUO_L DEFS 1
F081 RANSUO_H DEFS 1

ROMIUT
LD HL, (RANLAT)-01H
LD A, (RANLAT_FLG)-01H
AND NZ, moRNDGET

moRNDGET:
LDW (RANSUO_L), RANSU_REGO_L
```

【図 2 5 4】



【図 2 5 5】

(a) 大当たり判定テーブル (乱数範囲: それぞれ0~65535)

| 遊技状態  | 第1特図始動口       | 第2特図始動口       |
|-------|---------------|---------------|
| 低確率状態 | 10001 - 10187 | 20001 - 20187 |
| 高確率状態 | 30001 - 31310 | 40001 - 41310 |

(b) 高確率状態移行判定テーブル (乱数範囲: 0~127)

| 移行判定<br>乱数の範囲 |
|---------------|
| 11 - 74       |

(c) タイマ番号決定テーブル (乱数範囲: 0~65535)

| 大当りフラグ | タイマ<br>乱数の範囲 | 演出    |             |
|--------|--------------|-------|-------------|
|        |              | タイマ番号 | 変動時間<br>[s] |
| オフ     | 60236        | タイマ 1 | 5           |
|        | 4250         | タイマ 2 | 10          |
|        | 800          | タイマ 3 | 20          |
|        | 250          | タイマ 4 | 40          |
| オン     | 15536        | タイマ 2 | 10          |
|        | 9000         | タイマ 3 | 20          |
|        | 38000        | タイマ 4 | 40          |
|        | 3000         | タイマ 5 | 50          |

【図 2 5 6】

(a) 払出要求数テーブル

| データ | 備考                    |
|-----|-----------------------|
| 3   | 払出要求カウンタに対応 (第1特図始動口) |
| 4   | 払出要求カウンタに対応 (第2特図始動口) |
| 10  | 払出要求カウンタに対応 (一般入賞口用)  |
| 15  | 払出要求カウンタに対応 (可変入賞口用)  |

(b) 払出加工テーブル

| データ | 備考 (Bit0~3: マスク, Bit4~5: 加工種別) |
|-----|--------------------------------|
| 00H | 加工種別 = 0 (2進数表記: 0000 0000B)   |
| 13H | 加工種別 = 1 (2進数表記: 0001 0011B)   |
| 2EH | 加工種別 = 2 (2進数表記: 0010 1110B)   |
| 39H | 加工種別 = 3 (2進数表記: 0011 1001B)   |

(c) タイマ制御データテーブル

| タイマ種類数              | 2    | タイマ名                |
|---------------------|------|---------------------|
| タイマ個数               | 9    |                     |
| 各タイマの RAM<br>使用バイト数 | タイマ① | 1 特図表示図柄更新タイマ       |
|                     | タイマ② | 1 普通表示図柄更新タイマ       |
|                     | タイマ③ | 1 始動口1検知信号送信タイマ     |
|                     | タイマ④ | 1 始動口2検知信号送信タイマ     |
|                     | タイマ⑤ | 1 ガラス枠開放エラー検出・解除タイマ |
|                     | タイマ⑥ | 1 前枠開放エラー検出・解除タイマ   |
|                     | タイマ⑦ | 1 下皿満カエラー検出・解除タイマ   |
|                     | タイマ⑧ | 2 普通汎用タイマ           |
|                     | タイマ⑨ | 1 10.08ms基本タイマ      |
| タイマ更新周期             | 5    |                     |
| タイマ個数               | 1    |                     |
| タイマ⑧の RAM 使用バイト数    | 2    | 特図汎用タイマ             |

(d) ジャンプテーブル

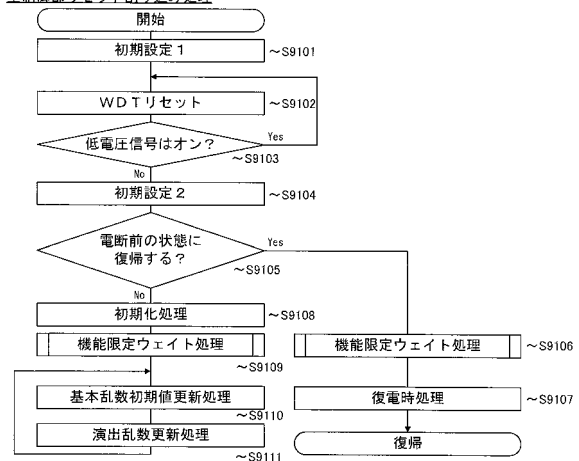
| 送信情報 | ジャンプ先              |
|------|--------------------|
| 01H  | 一般コマンド回転開始設定送信処理   |
| 02H  | 一般コマンド回転停止設定送信処理   |
| 04H  | 一般コマンド入賞演出設定送信処理   |
| 08H  | 一般コマンド終了演出設定送信処理   |
| 10H  | 一般コマンド大入賞口開放設定送信処理 |
| 20H  | 一般コマンド大入賞口閉鎖設定送信処理 |

(e) 復元データテーブル

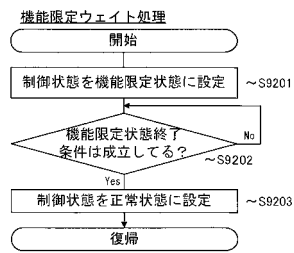
| 値   | 備考                           |
|-----|------------------------------|
| 00H | 加工種別 = 0 (2進数表記: 0000 0000B) |
| 03H | 加工種別 = 1 (2進数表記: 0000 0011B) |
| 0EH | 加工種別 = 2 (2進数表記: 0000 1110B) |
| 09H | 加工種別 = 3 (2進数表記: 0000 1001B) |

【図 2 5 7】

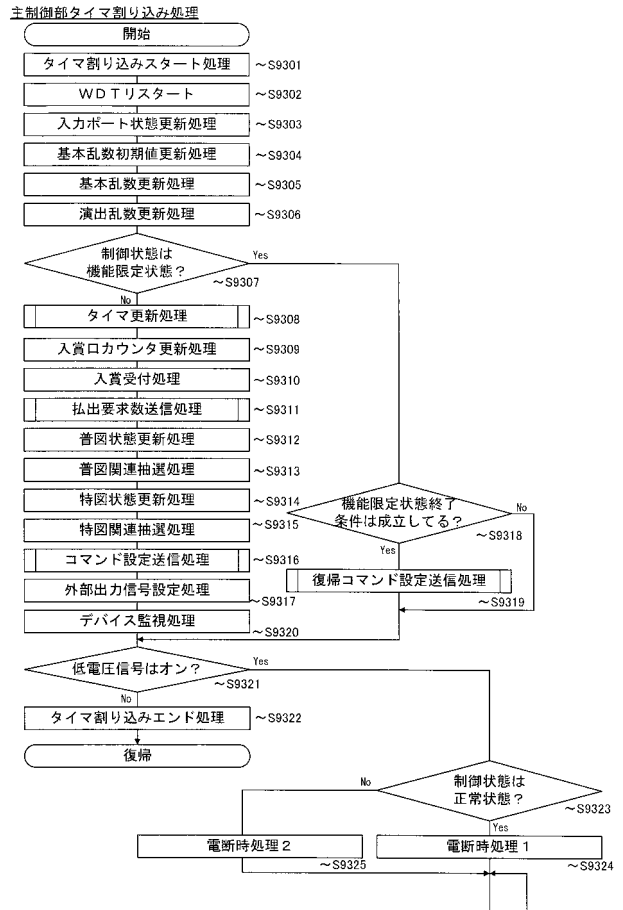
主制御部リセット割り込み処理



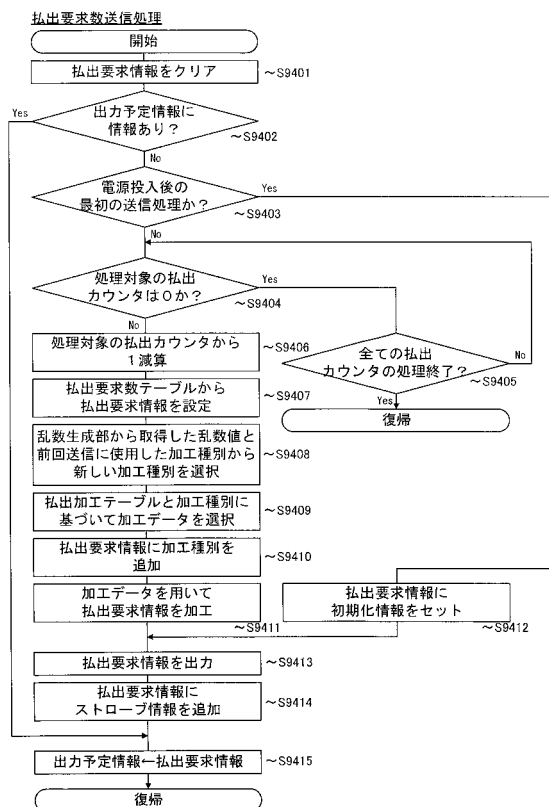
【図 258】



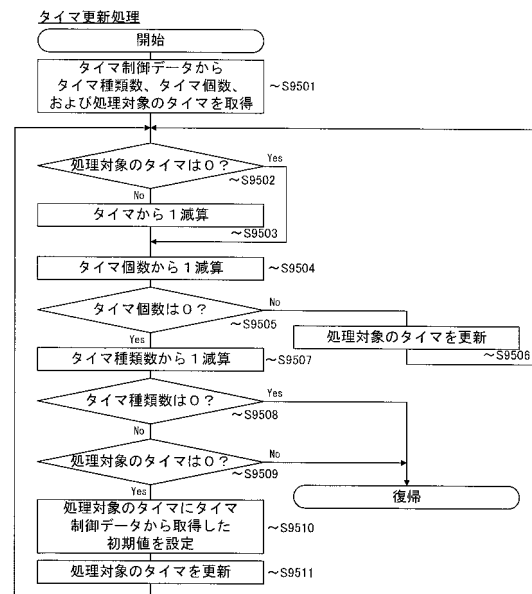
【図 259】



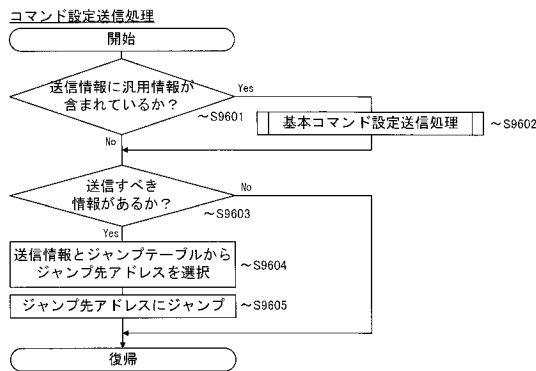
【図 260】



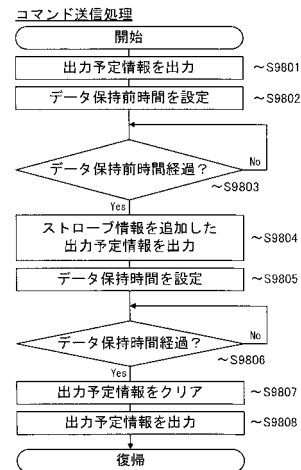
【図 261】



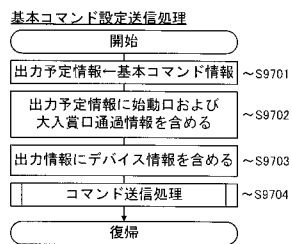
【図 262】



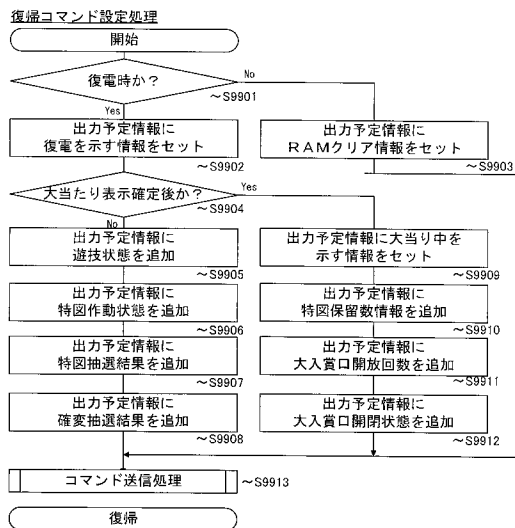
【図 264】



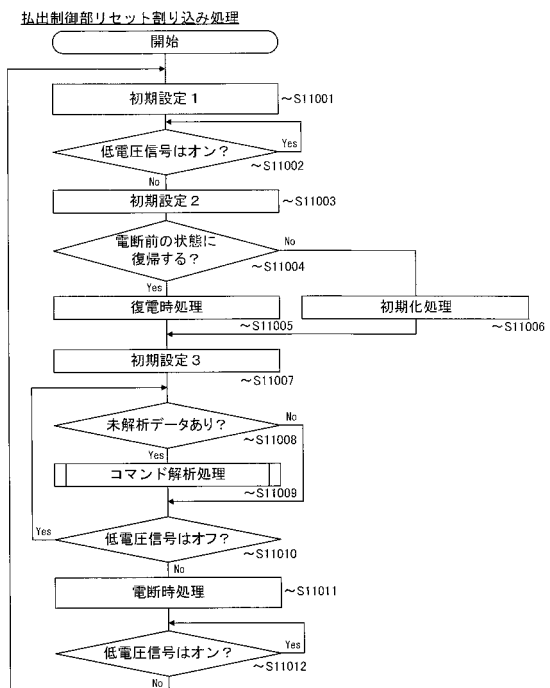
【図 263】



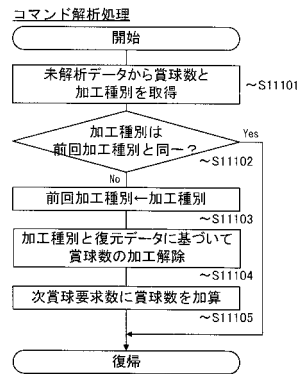
【図 265】



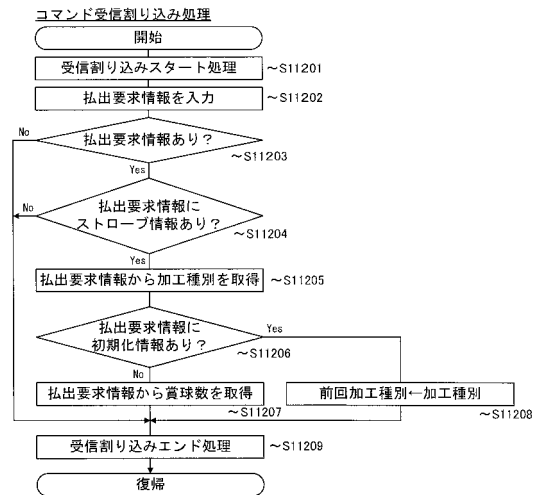
【図 266】



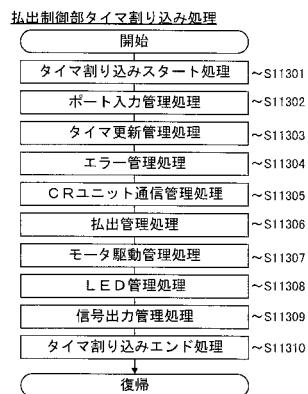
【図 267】



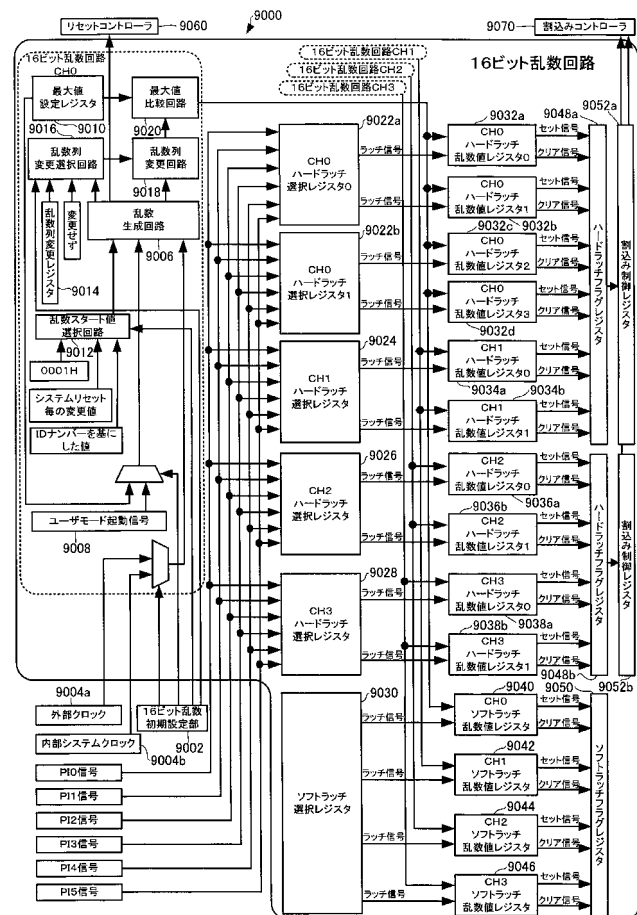
【図 268】



【図 269】

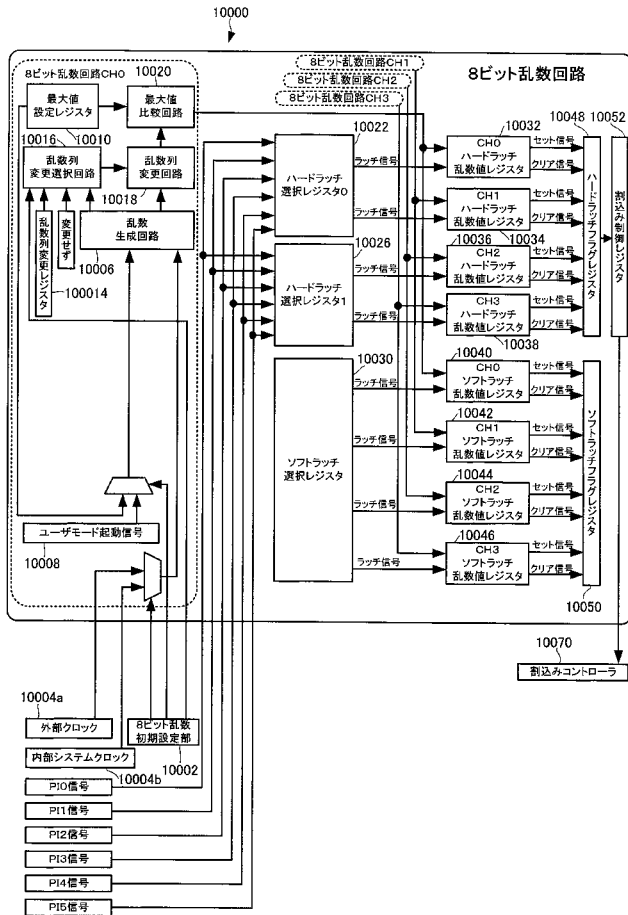


【図 270】

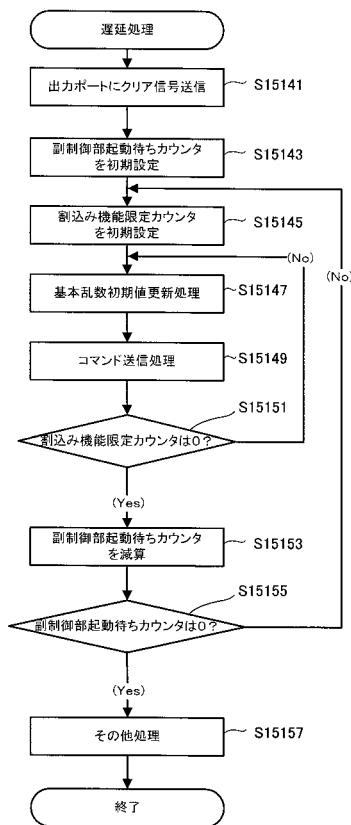




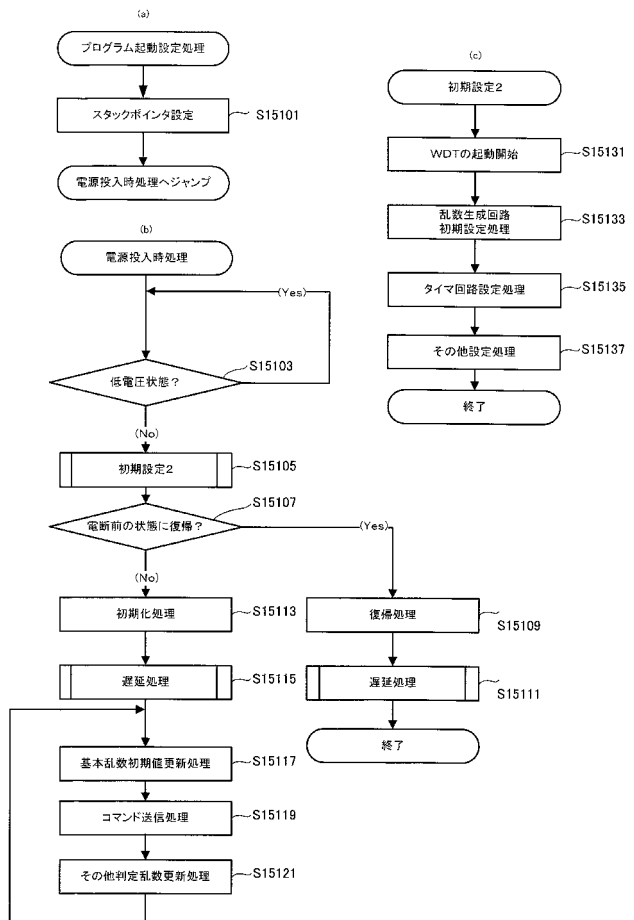
【図 271】



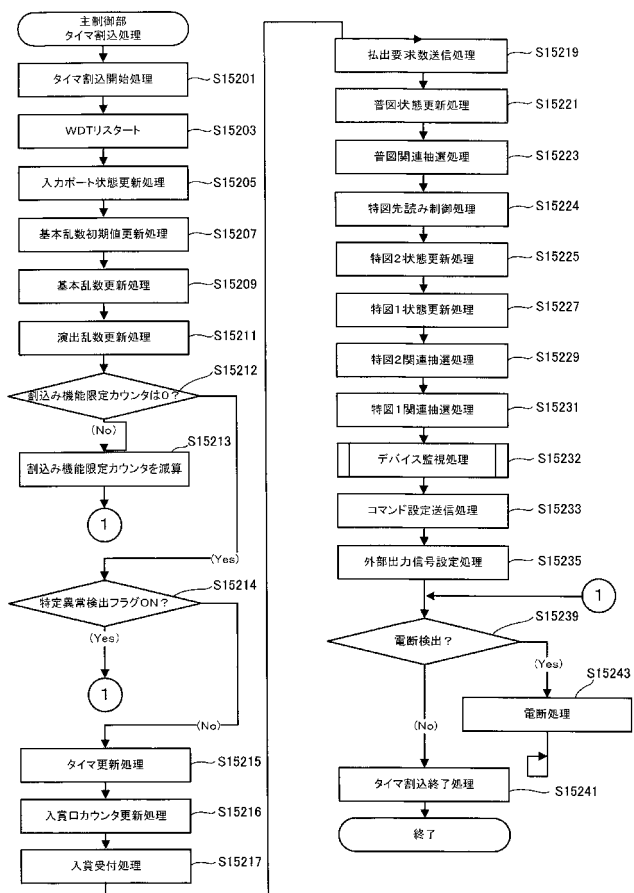
【図 273】



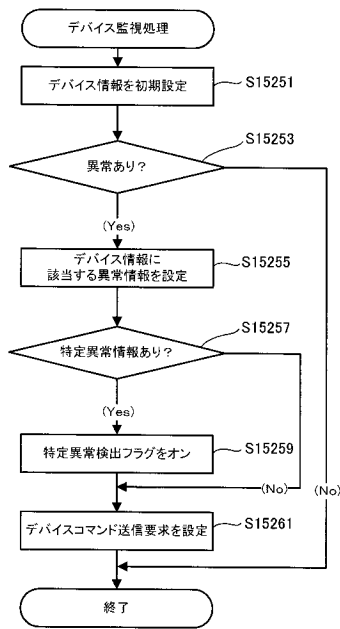
【図 272】



【図 274】



【図 275】



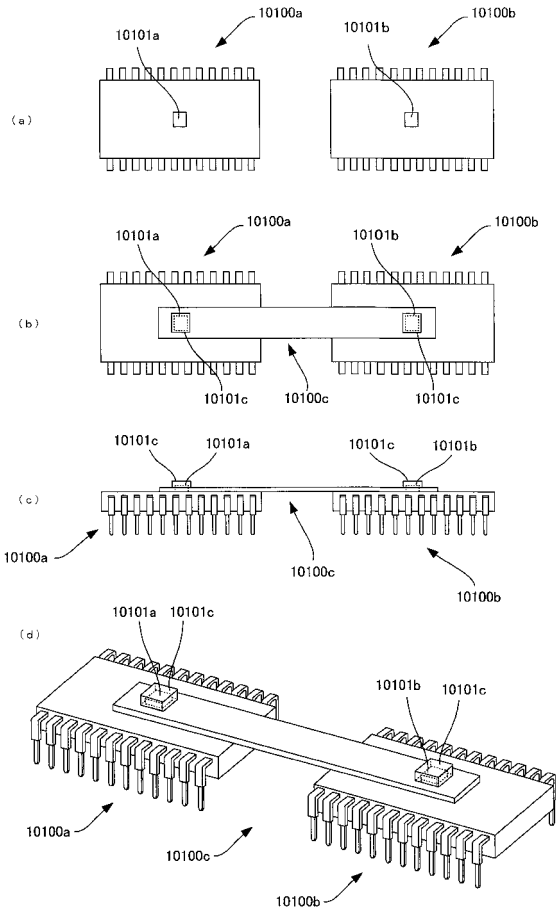
【図 276】

| 乱数                          | 更新場所                       | 備考                                        |
|-----------------------------|----------------------------|-------------------------------------------|
| 特図1 当選乱数                    | 乱数回路                       | 16ビット乱数回路CH0のハードラッチレジスタ0/外部クロック更新/最大値設定無し |
|                             |                            | 16ビット乱数回路CH1のソフトラッチレジスタ/外部クロック更新/最大値設定無し  |
|                             |                            | 16ビット乱数回路CH0のハードラッチレジスタ1/外部クロック更新/最大値設定無し |
|                             |                            | 16ビット乱数回路CH2のソフトラッチレジスタ/外部クロック更新/最大値設定無し  |
| 特図2 当選乱数                    | 乱数回路                       | 16ビット乱数回路CH3のハードラッチレジスタ0/外部クロック更新/最大値設定有り |
|                             |                            | 8ビット乱数回路CH0のハードラッチレジスタ0/外部クロック更新/最大値設定無し  |
| 表示図柄乱数                      | 乱数回路                       | 主制御部タイマ割込み処理                              |
| 普通図当選乱数                     | 主制御部タイマ割込み処理               | 主制御部タイマ割込み処理                              |
| 大当り用特図乱数<br>(特図1、2未選)       | 主制御部タイマ割込み処理               | 主制御部タイマ割込み処理                              |
| 小当り用特図乱数<br>(特図1、2未選)       | 主制御部タイマ割込み処理               | 主制御部タイマ割込み処理                              |
| はずれ用特図乱数<br>(特図1、2未選)       | 主制御部タイマ割込み処理               | 主制御部タイマ割込み処理                              |
| 初期値生成用乱数<br>(大当り、小当り、はずれ共通) | 主制御部タイマ割込み処理               | 主制御部タイマ割込み処理                              |
| 特図変動時間決定用乱数<br>(特図1、2未選)    | カウンタ回路                     | カウンタ回路のCH0                                |
| 普通変動時間決定用乱数                 | カウンタ回路                     | カウンタ回路のCH0                                |
| 演出用乱数                       | 主制御部メイン処理<br>/主制御部タイマ割込み処理 | 主制御部メイン処理<br>/主制御部タイマ割込み処理                |

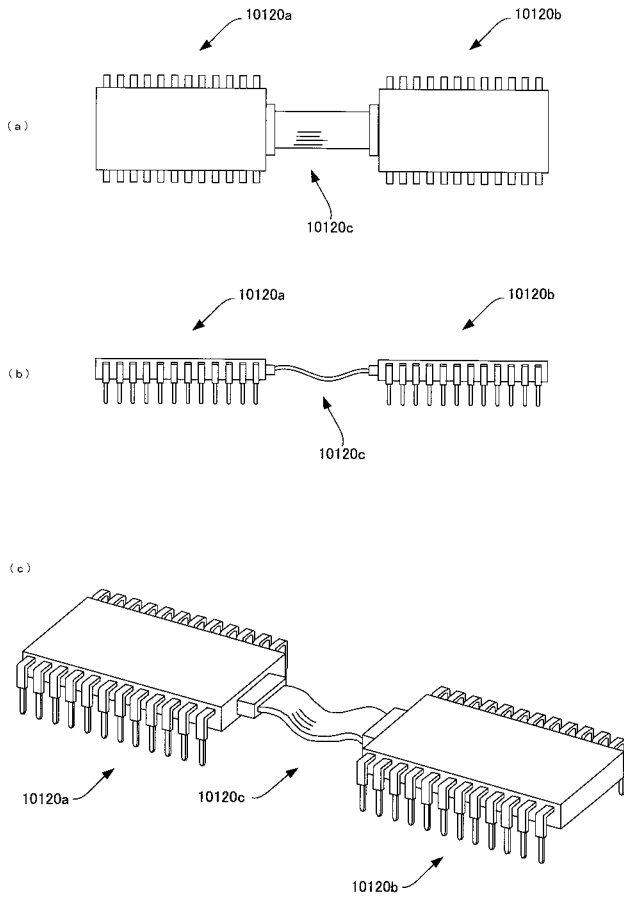
【図 277】

| 乱数           | 更新場所 | 備考                                       |
|--------------|------|------------------------------------------|
| 役抽選乱数        | 乱数回路 | 16ビット乱数回路CH0のソフトラッチレジスタ/内部クロック更新/最大値設定無し |
| 演出抽選乱数       | 乱数回路 | 16ビット乱数回路CH1のソフトラッチレジスタ/内部クロック更新/最大値設定無し |
| 左リール遅延時間抽選乱数 | 乱数回路 | 8ビット乱数回路CH0のソフトラッチレジスタ/内部クロック更新/最大値設定有り  |
| 中リール遅延時間抽選乱数 | 乱数回路 | 8ビット乱数回路CH1のソフトラッチレジスタ/内部クロック更新/最大値設定有り  |
| 右リール遅延時間抽選乱数 | 乱数回路 | 8ビット乱数回路CH2のソフトラッチレジスタ/内部クロック更新/最大値設定有り  |

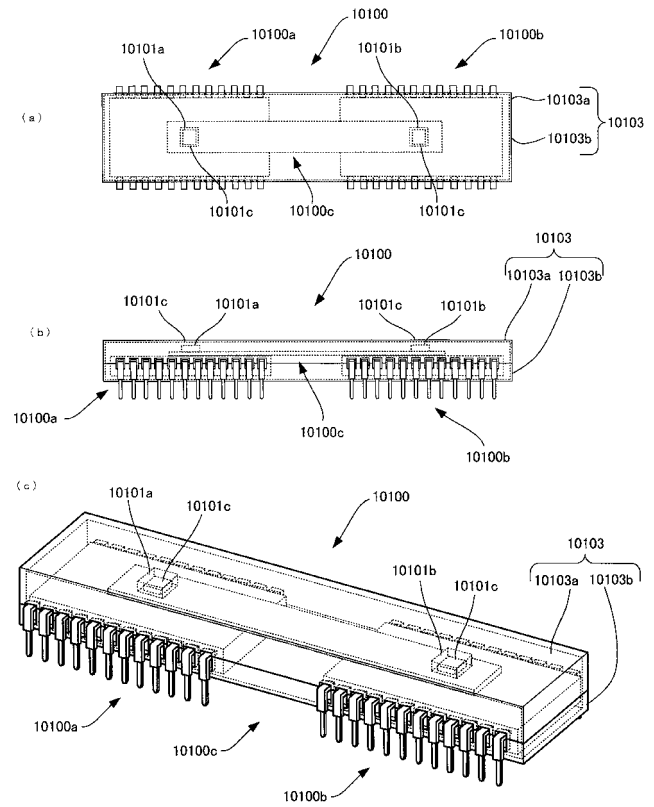
【図 278】



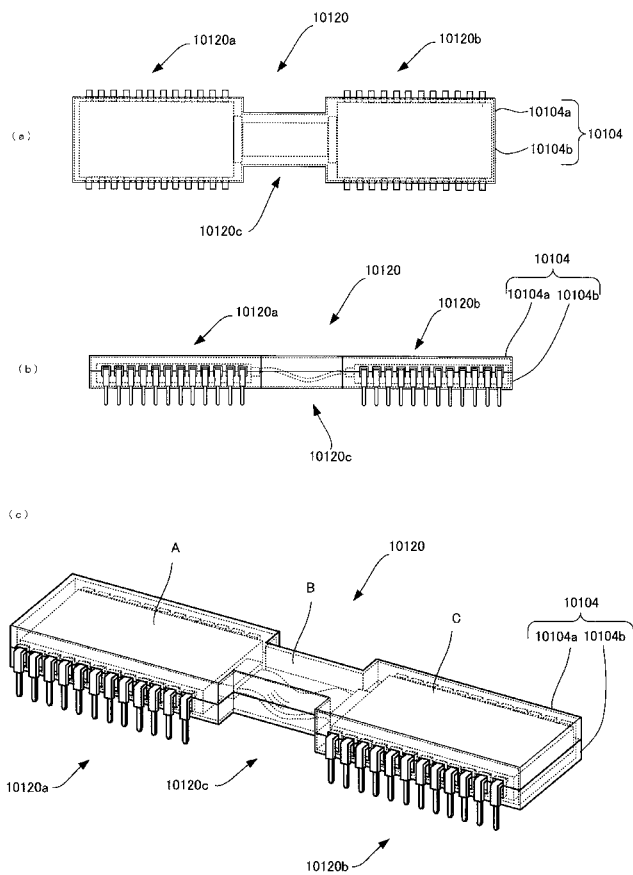
【図 279】



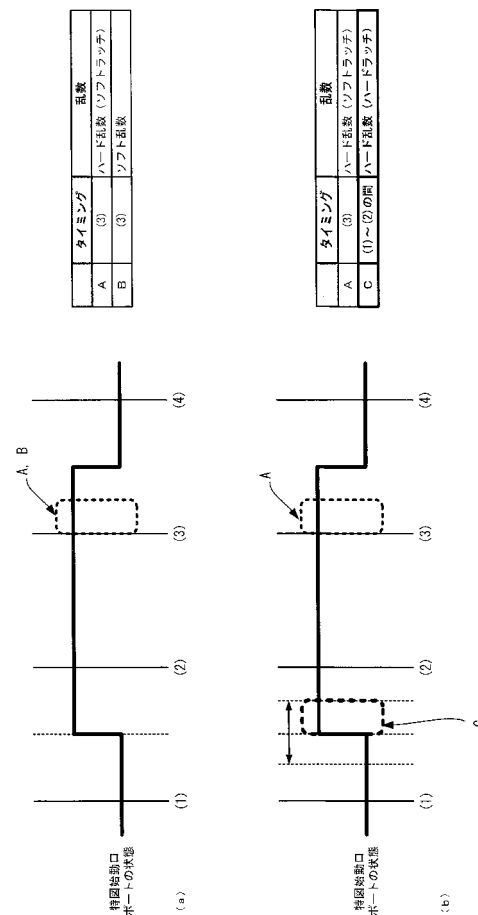
【図 280】



【図 281】



【図 282】



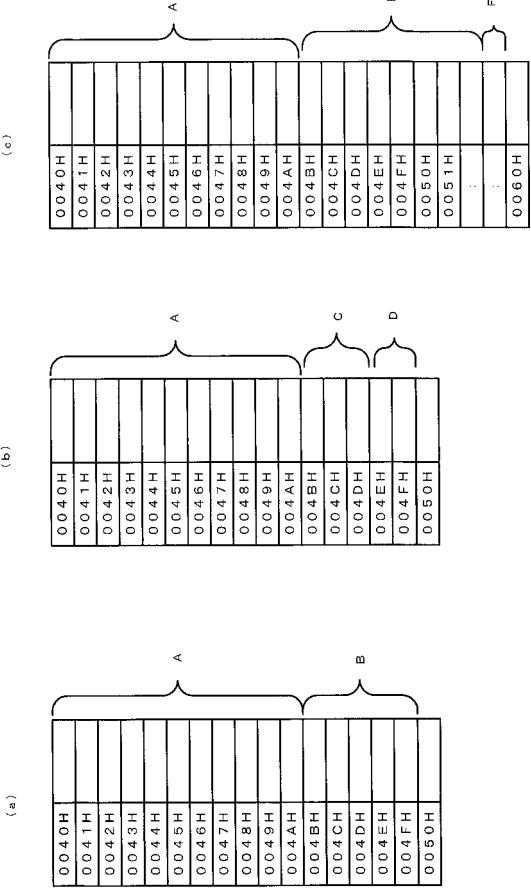


【図 287】

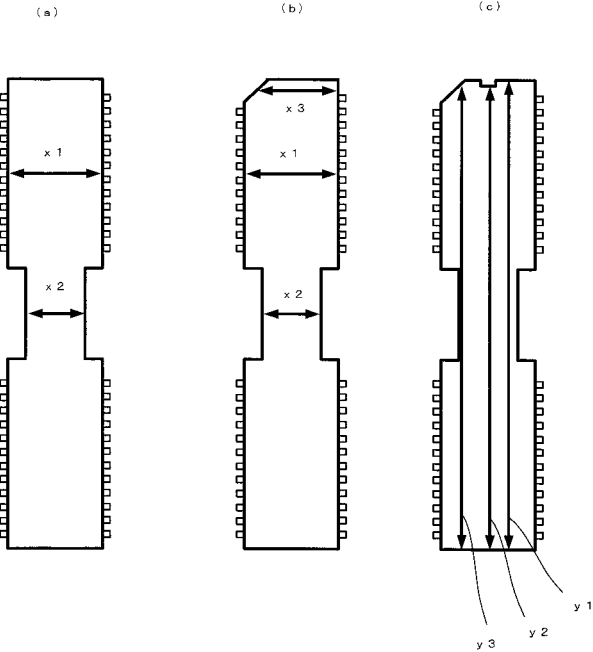
割込み初期設定レジスタの一例（プログラム管理エリア内）

| ビット番号 |  | 設定内容                       |  |        |      |      |       |      |      |       |      |
|-------|--|----------------------------|--|--------|------|------|-------|------|------|-------|------|
| 7 - 4 |  | 割込みベクタの上位4ビットを設定するビット      |  |        |      |      |       |      |      |       |      |
|       |  | No                         |  | A15-A8 |      |      | A7-A4 |      |      | A3-A0 |      |
|       |  | 0                          |  |        |      |      |       |      |      | 00H   |      |
|       |  | 1                          |  |        |      |      |       |      |      | 02H   |      |
|       |  | 2                          |  |        |      |      |       |      |      | 04H   |      |
|       |  | 3                          |  |        |      |      |       |      |      | 06H   |      |
|       |  | 4                          |  |        |      |      |       |      |      | 08H   |      |
|       |  | 5                          |  |        |      |      |       |      |      | 0AH   |      |
|       |  | 6                          |  |        |      |      |       |      |      | 0CH   |      |
|       |  | 設定値                        |  |        |      |      |       |      |      |       |      |
|       |  | セットされたレジスタの値               |  |        |      |      |       |      |      |       |      |
| 3     |  | ノンマスクカプル割込み中の多重受付の禁止・許可を設定 |  |        |      |      |       |      |      |       |      |
|       |  | 0：多重受付禁止<br>1：多重受付許可       |  |        |      |      |       |      |      |       |      |
| 2 - 0 |  | マスクカプル割込み要因の優先度の組み合わせを設定   |  |        |      |      |       |      |      |       |      |
|       |  | 設定値                        |  | 00H    | 01H  | 02H  | 03H   | 04H  | 05H  | 06H   |      |
|       |  | 高                          |  | PT0I   | PT0I | PT0I | RX0I  | RX0I | RX0I | RX0I  | RDGI |
|       |  | 1                          |  | PT1I   | PT1I | PT1I | TX0I  | TX0I | TX0I | TX0I  | RX0I |
|       |  | 2                          |  | PT2I   | PT2I | PT2I | RDGI  | PT0I | PT1I | PT1I  | TX0I |
|       |  | 3                          |  | RDGI   | XINT | RX0I | PT0I  | PT1I | PT2I | PT1I  | TX0I |
|       |  | 4                          |  | RX0I   | RDGI | TX0I | PT1I  | PT2I | RX0I | PT1I  | TX0I |
|       |  | 5                          |  | TX0I   | RX0I | RDGI | PT2I  | RDGI | TX0I | PT2I  | TX0I |
|       |  | 6                          |  | XINT   | TX0I | XINT | XINT  | XINT | XINT | XINT  | XINT |
|       |  | 低                          |  |        |      |      |       |      |      |       |      |

【図 288】



【図 289】



---

フロントページの続き

Fターム(参考) 2C082 AA02 AB03 AB12 AB16 AB25 BA02 BA22 BA35 BB02 BB13  
BB14 BB23 BB24 BB33 BB46 BB52 BB78 BB83 BB85 BB93  
BB94 CA02 CA23 CA24 CA27 CA29 CB04 CB23 CB33 CB42  
CC01 CC13 CC51 CD03 CD06 CD11 CD12 CD18 CD31 CD41  
CD51 CD54 CE12 CE13 CE14 CE23 DA19 DA29 DA52 DA55  
DA58 DA63 DA66 DA80  
2C088 BA37 BA56 BC22 BC25 BC47 BC49 BC58 CA27 EA10 EB14  
EB28 EB53 EB56 EB58