

(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2013-233297

(P2013-233297A)

(43) 公開日 平成25年11月21日(2013.11.21)

(51) Int.Cl. F I テーマコード (参考)  
**A 6 3 F 7/02 (2006.01)** A 6 3 F 7/02 3 2 6 Z 2 C 0 8 8

審査請求 未請求 請求項の数 5 O L (全 112 頁)

(21) 出願番号	特願2012-107599 (P2012-107599)	(71) 出願人	000132747 株式会社ソフィア 群馬県桐生市境野町7丁目201番地
(22) 出願日	平成24年5月9日(2012.5.9)	(74) 代理人	100075513 弁理士 後藤 政喜
		(74) 代理人	100120260 弁理士 飯田 雅昭
		(74) 代理人	100142468 弁理士 高山 裕志
		(72) 発明者	園田 欽章 群馬県太田市吉沢町990番地 株式会社 ソフィア内
		(72) 発明者	亀井 欽一 群馬県太田市吉沢町990番地 株式会社 ソフィア内

最終頁に続く

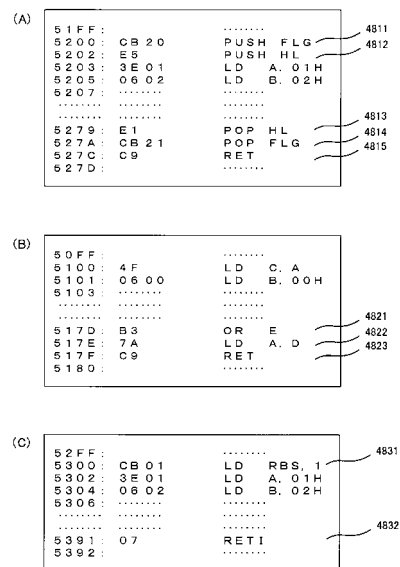
(54) 【発明の名称】 遊技機

(57) 【要約】

【課題】遊技制御プログラムのコード量を削減する。

【解決手段】遊技制御装置を備えた遊技機において、遊技制御プログラム記憶手段と、演算処理を行う演算処理手段と、演算処理を行う際にデータが記憶されるレジスタと、演算処理手段によって更新される情報が記憶される更新情報記憶手段とを備え、更新情報記憶手段にはスタック領域が設定され、レジスタには演算処理手段による演算結果を記憶するためのフラグレジスタが含まれ、遊技制御プログラムには、呼出元ルーチンと、呼出元ルーチンから呼び出される第1呼出先ルーチンと、所定の割込信号によって呼び出される第2呼出先ルーチンと、が含まれ、演算処理手段は、第1呼出先ルーチンが呼び出された場合は、戻りアドレスをスタック領域に格納し、フラグレジスタを格納せずに処理を開始し、第2呼出先ルーチンが呼び出された場合は、戻りアドレスとフラグレジスタをスタック領域に格納して処理を開始する。

【選択図】 図48



**【特許請求の範囲】****【請求項 1】**

所定の始動条件の成立により変動表示ゲームが実行され、該変動表示ゲームが特別な結果となる場合に遊技者に特典を付与する特別遊技状態を発生させる遊技機において、

前記特別遊技状態を発生させる制御を行う遊技制御装置と、

該遊技制御装置からの指令により前記変動表示ゲームに関わる表示の制御を行う表示制御装置と、

を備えるとともに、

前記遊技制御装置には、

遊技制御プログラムを記憶する遊技制御プログラム記憶手段と、

前記遊技制御プログラムにより所要の演算処理を行う演算処理手段と、

前記演算処理を行う際にデータが記憶されるレジスタと、

前記演算処理手段によって更新される情報が記憶される更新情報記憶手段と、

を備え、

前記更新情報記憶手段にはスタック領域が設定され、

前記レジスタには、前記演算処理手段による演算結果を記憶するためのフラグレジスタが含まれ、

前記遊技制御プログラムには、

呼出元の処理が記述された呼出元ルーチンと、

前記呼出元ルーチンに記述された呼出の命令が実行された場合に呼び出される処理が記述された第 1 の呼出先ルーチンと、

前記呼出元ルーチンに記述された命令の実行中において、所定の割込信号が発生した場合に呼び出される処理が記述された第 2 の呼出先ルーチンと、が含まれ、

前記演算処理手段は、

前記第 1 の呼出先ルーチンが呼び出された場合は、前記呼出元ルーチンに含まれる戻りアドレスの値を前記スタック領域に格納し、且つ前記フラグレジスタの値を前記スタック領域に格納しない状態で、前記第 1 の呼出先ルーチンの処理の実行を開始する一方で、

前記第 2 の呼出先ルーチンが呼び出された場合は、前記呼出元ルーチンに含まれる戻りアドレスの値と前記フラグレジスタの値の各々を前記スタック領域に格納した状態で、前記第 2 の呼出先ルーチンの処理の実行を開始することを特徴とする遊技機。

**【請求項 2】**

前記演算処理手段は、前記第 2 の呼出先ルーチンが呼び出された場合は、前記フラグレジスタの値を前記スタック領域に格納した後に、前記呼出元ルーチンに含まれる戻りアドレスの値を前記スタック領域に格納することを特徴とする請求項 1 に記載の遊技機。

**【請求項 3】**

前記フラグレジスタの容量は、前記戻りアドレスを格納するために必要な容量と異なることを特徴とする請求項 2 に記載の遊技機。

**【請求項 4】**

前記レジスタは、第 1 のレジスタと前記第 1 のレジスタと同じ構成を有する第 2 のレジスタを含み、

前記演算処理手段は、

前記第 1 のレジスタ及び前記第 2 のレジスタのいずれか一方にアクセスし、

アクセス可能なレジスタを切り替えるバンク切替手段を備えることを特徴とする請求項 3 に記載の遊技機。

**【請求項 5】**

前記第 2 の呼出先ルーチンには、前記補助遊技に関連する乱数及びタイマを更新する処理が含まれることを特徴とする請求項 4 に記載の遊技機。

**【発明の詳細な説明】****【技術分野】****【0001】**

10

20

30

40

50

本発明は、レジスタを備えたプロセッサによって制御を行う遊技機に関する。

【背景技術】

【0002】

従来よりも、プログラムのメンテナンス性および処理速度を維持しつつ、不正改造抑制および不正改造発見容易性向上を実現するために、ROMの0008H番地から000FH番地には、変動停止コマンド送信処理のプログラムを格納し、ROMの0020H番地から0027H番地には、賞球数コマンド送信処理のプログラムを格納するとともに、これらのプログラムをRST命令を用いて呼び出すようにした遊技機が知られている（例えば、特許文献1）。

【0003】

この特許文献1の遊技機の構成では、特許文献1の図22に示すように、RST命令を実行すると、呼び出し元のモジュールの戻りアドレスの値をスタックに格納して、呼び出し先のサブモジュールに移行する。その後、呼び出し先のサブモジュールの終了時にRET命令を実行することで、スタックに格納してあった戻りアドレスの値をプログラムカウンタに戻し、呼び出し元のモジュールに復帰する。なお、特許文献1の段落[0148]に示すように、CALL命令を用いても同様の処理が行われる。

【0004】

なお、特許文献1のようにRST命令やCALL命令を実行しなくても、タイマ割込等の割込が発生した場合に、呼び出し元のモジュールの戻りアドレスの値をスタックに格納して、呼び出し先のサブモジュール（タイマ割込みの処理モジュール）に移行する処理を行う遊技機もある。

【0005】

例えば、特許文献2に開示された遊技機では、特許文献2の段落[0071]～[0073]に示すように、タイマ割込等の割込が発生すると、呼び出し元のモジュールの戻りアドレスの値をスタックに格納して、呼び出し先のサブモジュール（タイマ割込みの処理モジュール）に移行する。その後、呼び出し先のサブモジュールの終了時にRETI命令を実行することで、スタックに格納してあった戻りアドレスの値をプログラムカウンタに戻し、呼び出し元のモジュールに復帰する。

【0006】

このようなタイマ割込みの処理を行うときは、特許文献3の図5や特許文献4の図6（c）に示すように、処理の先頭にて、フラグレジスタの値をAFレジスタとしてスタックに格納することが通常行われる。なお、このスタックに格納されたフラグレジスタの値は、特許文献3の図5に示されるように、タイマ割込みの処理を終了してRETI命令を実行する際に元のフラグレジスタへ復帰するようになっている。

【先行技術文献】

【特許文献】

【0007】

【特許文献1】特開2008-080004号公報

【特許文献2】特開2001-212330号公報

【特許文献3】特開2004-089476号公報

【特許文献4】特開2006-068568号公報

【発明の概要】

【発明が解決しようとする課題】

【0008】

呼び出し元のモジュールで用いられていたレジスタの値が、呼び出し先のサブモジュールで変更されてしまうと、CPUの演算処理に支障を来す。そのため、従来の遊技機では、特許文献3の図5のS11に示すように、処理の先頭でPUSH命令を用いてレジスタの値をスタックに退避する処理が通常行われ、特許文献3の図5のS25に示すように、処理の最後でPOP命令を用いてレジスタの値をスタックから復帰させる処理が行われる。このとき、特許文献4の図6（c）に示すように、スタックには、様々なレジスタの

10

20

30

40

50

値が一時的に格納される。そのため、呼び出し先のモジュール内にて、PUSH命令やPOP命令を記述する必要があり、プログラム容量が増えてしまう要因となっていた。

【0009】

このようなPUSH命令及びPOP命令を用いて、レジスタの値をスタックから出し入れする処理は、特許文献2に開示されたタイマ割り込みの処理に限らず、特許文献1に示したようなRST命令（或いはCALL命令）を用いて呼び出されるサブモジュール内でも、頻繁に行われている。そのため、このようなサブモジュール内にもPUSH命令やPOP命令を記述する必要があり、プログラム容量が増えてしまう要因となっていた。

【0010】

なお、前者のタイマ割り込みの処理では、フラグレジスタの値を無条件にスタックに格納する必要があったのに対して、後者のRST命令（或いはCALL命令）を用いて呼び出されるサブモジュールの処理では、フラグレジスタを退避すべきか否かは、モジュールの目的によって一意に定まっていなかった。そこで、このような条件を満たしつつも、プログラム容量を削減するための工夫が必要とされていた。

【0011】

本発明は、上記のような問題点を解決するためになされたもので、フラグレジスタを退避すべきか否かをプログラムによって決定できる構成としながらも、遊技制御プログラム全体のコード量を削減することを目的とする。

【課題を解決するための手段】

【0012】

本発明の代表的な一形態では、所定の始動条件の成立により変動表示ゲームが実行され、該変動表示ゲームが特別な結果となる場合に遊技者に特典を付与する特別遊技状態を発生させる遊技機において、前記特別遊技状態を発生させる制御を行う遊技制御装置と、該遊技制御装置からの指令により前記変動表示ゲームに関わる表示の制御を行う表示制御装置と、を備え、るとともに、前記遊技制御装置には、遊技制御プログラムを記憶する遊技制御プログラム記憶手段と、前記遊技制御プログラムにより所要の演算処理を行う演算処理手段と、前記演算処理を行う際にデータが記憶されるレジスタと、前記演算処理手段によって更新される情報が記憶される更新情報記憶手段と、を備え、前記更新情報記憶手段にはスタック領域が設定され、前記レジスタには、前記演算処理手段による演算結果を記憶するためのフラグレジスタが含まれ、前記遊技制御プログラムには、呼出元の処理が記述された呼出元ルーチンと、前記呼出元ルーチンに記述された呼出の命令が実行された場合に呼び出される処理が記述された第1の呼出先ルーチンと、前記呼出元ルーチンに記述された命令の実行中において、所定の割込信号が発生した場合に呼び出される処理が記述された第2の呼出先ルーチンと、が含まれ、前記演算処理手段は、前記第1の呼出先ルーチンが呼び出された場合は、前記呼出元ルーチンに含まれる戻りアドレスの値を前記スタック領域に格納し、且つ前記フラグレジスタの値を前記スタック領域に格納しない状態で、前記第1の呼出先ルーチンの処理の実行を開始する一方で、前記第2の呼出先ルーチンが呼び出された場合は、前記呼出元ルーチンに含まれる戻りアドレスの値と前記フラグレジスタの値の各々を前記スタック領域に格納した状態で、前記第2の呼出先ルーチンの処理の実行を開始する。

【0013】

本明細書における実施形態では、「遊技機」をパチンコ機として説明しているが、スロットマシンなどの回胴式遊技機であってもよいし、スマートボールであってもよい。これらのパチンコ機以外の遊技機であっても、前述した課題はパチンコ機の場合と同様に発生する課題である。

【0014】

「所定の始動条件」とは、パチンコ機であれば、例えば、遊技球が始動入賞口に入賞する場合である。また、スロットマシンであれば、例えば、遊技媒体のベットをした後、遊技者によってスタートレバーが操作される場合である。

【0015】

「補助遊技」とは、例えば、表示画面の画像を表示可能な領域（表示領域）設けられた複数の変動表示領域において、各変動表示領域に識別情報などが表示され、各識別図柄が変動表示する変動表示ゲームである。変動表示ゲームにおける「特別な結果」とは、例えば、変動表示する識別図柄がすべて同じ図柄で停止した場合である。

【0016】

「遊技制御プログラム」とは、遊技の進行制御を行うプログラムや、演出制御を行うプログラムが含まれる。「遊技制御プログラム」は、「遊技制御プログラム記憶手段」に記憶され、「演算処理手段」によって実行される。「遊技制御プログラム記憶手段」は、読み書き可能（揮発性）であってもよいし、読み出し専用（不揮発性）であってもよい。「演算処理手段」は、いわゆるCPU（若しくはCPUに含まれる命令解釈実行回路）である。

10

【0017】

「レジスタ」は、「演算処理手段」によってアクセスされるデータが格納される。「遊技制御プログラム記憶手段」や「更新情報記憶手段」に格納されたデータを「演算処理手段」がアクセスする場合には「レジスタ」を介するようにしてもよいし、直接アクセス可能としてもよい。

【0018】

「フラグレジスタ」とは、CPU（演算処理手段）の演算状態を示す値が格納されるレジスタである。例えば、演算結果が正であるか否かを示すフラグや、0であるか否かを示すフラグなどが含まれる。「演算処理手段」は「フラグレジスタ」を参照して処理を分岐させたり、処理の実行順序を制御したりする。

20

【0019】

「所定の割込信号」とは、例えば、所定の時間間隔で出力する場合（タイマ割込）や停電発生時に出力する場合などがある。遊技制御装置（遊技制御手段）は、割込信号を受信すると、割込信号に対応する割り込み処理（第2の呼出先ルーチン）を実行する。

【0020】

「スタック領域」は、実行中の処理から他の処理を呼び出す場合や実行中の処理を中断して割り込み処理を実行する場合に、元の処理に復帰するために必要なデータを一時的に格納する領域である。呼び出された順序が新しい順にスタック領域に格納されたデータを取り出してレジスタや更新情報記憶手段に戻す。

30

【発明の効果】

【0021】

本発明の一形態によれば、割り込みが発生した場合など、フラグレジスタを確実に退避させる必要がある場合にはフラグレジスタの値をスタック領域に格納してから処理を実行する。一方、必ずしもフラグレジスタの値を退避させる必要がない場合には、フラグレジスタの値をスタック領域に格納せずに処理を実行することによって呼び出し先の処理の種類に応じた処理を実行することができる。

【図面の簡単な説明】

【0022】

【図1】本発明の第1の実施の形態の遊技装置の構成を説明する図である。

40

【図2】本発明の第1の実施の形態の遊技機の背面図である。

【図3】本発明の第1の実施の形態の遊技盤の正面図である。

【図4】本発明の第1の実施の形態の遊技装置のブロック図である。

【図5】本発明の第1の実施の形態の遊技用演算処理装置（アミューズチップ）のブロック図である。

【図6】本発明の第1の実施の形態の遊技制御装置におけるシリアル送信回路の構成例を示すブロック図である。

【図7】本発明の第1の実施の形態の送信シリアルチャンネル設定レジスタの構成例を示す図である。

【図8】本発明の第1の実施の形態の送信制御レジスタの構成例を示す図である。

50

【図 9】本発明の第 1 の実施の形態の送信データステータスレジスタの構成例を示す図である。

【図 10】本発明の第 1 の実施の形態の送信データレジスタ（1 段分）の構成例を示す図である。

【図 11】本発明の第 1 の実施の形態の遊技制御装置に備わる遊技用演算処理装置（アミューズチップ）とその周辺のブロック図である。

【図 12】本発明の第 1 の実施の形態の CPU コアの内部構成を説明するブロック図である。

【図 13】本発明の第 1 の実施の形態のフラグレジスタの構成を説明する図である。

【図 14】本発明の第 1 の実施の形態のユーザワーク RAM の一例を示す図である。

【図 15】本発明の第 1 の実施の形態のユーザワーク RAM に割り当てられたワークエリアに格納されるデータの具体例を説明する図であり、保留カウンタ、各種乱数の生成領域及び普図乱数の保存領域を示す図である。

【図 16】本発明の第 1 の実施の形態のユーザワーク RAM に割り当てられたワークエリアに格納されるデータの具体例を説明する図であり、第 1 特図変動表示ゲームを実行するために必要な各種乱数の保存領域を示す図である。

【図 17】本発明の第 1 の実施の形態のユーザワーク RAM に割り当てられたワークエリアに格納されるデータの具体例を説明する図であり、第 2 特図変動表示ゲームを実行するために必要な各種乱数の保存領域を示す図である。

【図 18】本発明の第 1 の実施の形態のスタック領域の一例を示す図である。

【図 19】本発明の第 1 の実施の形態のスタック領域の一例を示す図である。

【図 20】本発明の第 1 の実施の形態のスタック領域の変形例を示す図である。

【図 21】本発明の第 1 の実施の形態の CPU コアによって実行される遊技制御プログラムを記述するためのアセンブリ言語の命令を説明する図であり、変換後のコードデータにアドレス部を含まない命令（変換後のコードデータが命令コード部のみで構成される命令）を示す図である。

【図 22】本発明の第 1 の実施の形態の CPU コアによって実行される遊技制御プログラムを記述するためのアセンブリ言語の命令を説明する図であり、変換後のコードデータに命令コード部とアドレス部の各々を含む命令を示す図である。

【図 23】本発明の第 1 の実施の形態の各装置（遊技制御装置、払出制御装置、及び演出制御装置）の電源投入時処理のフローチャートである。

【図 24】本発明の第 1 の実施の形態の遊技制御装置プログラム開始準備処理を説明するフローチャートである。

【図 25】本発明の第 1 の実施の形態の遊技制御装置メイン処理の前半部のフローチャートである。

【図 26】本発明の第 1 の実施の形態の遊技制御装置メイン処理の後半部のフローチャートである。

【図 27】本発明の第 1 の実施の形態の遊技制御装置メイン処理におけるステップ 2501 からステップ 2502 B までの処理を説明する図である。

【図 28】本発明の第 1 の実施の形態のディレイ処理を説明する図である。

【図 29】本発明の第 1 の実施の形態のタイマ割込処理を示すフローチャートである。

【図 30】本発明の第 1 の実施の形態の遊技制御装置から、演出制御装置及び払出制御装置に初期化指令信号を送信する初期化指令送信処理の手順を示すフローチャートである。

【図 31】本発明の第 1 の実施の形態の遊技制御装置から、演出制御装置及び払出制御装置にコマンドを送信するためのコマンド送信処理の手順を示すフローチャートである。

【図 32】本発明の第 1 の実施の形態の電源投入時の遊技制御装置、払出制御装置、及び演出制御装置が行う処理、並びに、遊技制御装置に備わるシリアル送信回路の状態のタイミングチャートである。

【図 33】本発明の第 1 の実施の形態の遊技制御装置から払出制御装置に送信される排出指令の一例を示す図である。

10

20

30

40

50

【図 3 4】本発明の第 1 の実施の形態のスイッチの立ち上がりを検出する手順を示すタイムチャートである。

【図 3 5】本発明の第 1 の実施の形態の遊技制御装置から演出制御装置に送信される演出制御コマンドの一例を示す図である。

【図 3 6】本発明の第 1 の実施の形態の遊技制御装置から演出制御装置に送信される送信データの構成を示す説明図であり、( a ) は送信データの概略構成を示し、( b ) は送信データの詳細構成を示している。

【図 3 7】本発明の第 1 の実施の形態の特図ゲーム処理の手順を示すフローチャートである。

【図 3 8】本発明の第 1 の実施の形態の特図ゲーム処理におけるゲーム処理番号に基づいた分岐の処理を説明する図である。

【図 3 9】本発明の第 1 の実施の形態の始動口 S W 監視処理の手順を示すフローチャートである。

【図 4 0】本発明の第 1 の実施の形態の特図始動口 S W 共通処理の手順を示すフローチャートである。

【図 4 1】本発明の第 1 の実施の形態の特図保留情報判定処理の手順を示すフローチャートである。

【図 4 2】本発明の第 1 の実施の形態の特図大当たり判定処理の手順を示すフローチャートである。

【図 4 3】本発明の第 1 の実施の形態の後半変動パターンテーブルである。

【図 4 4】本発明の第 1 の実施の形態の前半変動パターンテーブルである。

【図 4 5】本発明の第 1 の実施の形態の特図普段処理の手順を示すフローチャートである。

【図 4 6】本発明の第 1 の実施の形態の特図 1 / 特図 2 変動開始処理の手順を示すフローチャートである。

【図 4 7 A】本発明の第 1 の実施の形態における始動記憶をシフトさせる処理のプログラムの一例を示す図であり、K レジスタを利用して記憶容量を削減したプログラムである。

【図 4 7 B】本発明の第 1 の実施の形態における始動記憶をシフトさせる処理のプログラムの一例を示す図であり、従来のプログラムである。

【図 4 8】本発明の第 1 の実施の形態のサブルーチンを構成するプログラムの例を説明する図である。

【図 4 9】本発明の第 1 の実施の形態の変動表示ゲームにおける画面遷移を説明する図である。

【図 5 0】本発明の第 1 の実施の形態の変動表示ゲームにおける画面遷移を説明する図である。

【図 5 1 A】本発明の第 1 の実施の形態において、戻りアドレス値の後にフラグレジスタの値をスタック領域に待避させた状態を説明する図である。

【図 5 1 B】本発明の第 1 の実施の形態において、戻りアドレス値の前にフラグレジスタの値をスタック領域に待避させた状態を説明する図である。

【図 5 1 C】本発明の第 1 の実施の形態において、戻りアドレス値とフラグレジスタの値とが同一のバイト数で構成されている状態を説明する図である。

【図 5 1 D】本発明の第 1 の実施の形態において、戻りアドレス値とフラグレジスタの値とが異なるバイト数で構成されている状態を説明する図である。

【図 5 2】本発明の第 1 の実施の形態の変形例のユーザワーク R A M の一例を示す図である。

【図 5 3 A】本発明の第 2 の実施の形態の特図変動表示ゲームを実行するために必要な各種乱数の保留 1 から 4 までの保存領域を示す図である。

【図 5 3 B】本発明の第 2 の実施の形態の特図変動表示ゲームを実行するために必要な各種乱数の保留 5 から 8 までの保存領域を示す図である。

【図 5 4】本発明の第 2 の実施の形態の特図保留情報判定処理の手順を示すフローチャー

10

20

30

40

50

トである。

【図 5 5】本発明の第 2 の実施の形態の特図普段処理の手順を示すフローチャートである。

【発明を実施するための形態】

【0023】

以下、本発明の実施の形態について、図面を参照しながら説明する。

【0024】

なお、以下の実施の形態の説明における前後左右とは、遊技者から見た、つまり遊技盤（遊技機）に向かって見た方向を指すものとする。

【0025】

（第 1 の実施の形態）

図 1 は、本発明の第 1 の実施の形態の遊技装置 6 の構成を説明する図である。

【0026】

遊技装置 6 は、有価価値を記憶する記憶媒体が挿入されるカードユニット 7 0 及び実際に遊技を行い、遊技媒体を払出可能な遊技機 1 を備える。

【0027】

まず、遊技機 1 について説明する。

【0028】

遊技機 1 の前面枠 3 は、本体枠（外枠）2 にヒンジ 4 によって開閉回動可能に組み付けられる。遊技盤 5（図 3 参照）は前面枠 3 の表側に形成された収納部（図示省略）に収装される。また、前面枠 3 には、遊技盤 5 の前面を覆うカバーガラス（透明部材）を備えたガラス枠 1 8 が取り付けられている。

【0029】

ガラス枠 1 8 のカバーガラスの周囲には、装飾光が発光される装飾部材 9 が備えられている。この装飾部材 9 の内部にはランプや LED 等からなる装飾装置が備えられている。この装飾装置を所定の発光態様によって発光させることによって、装飾部材 9 が所定の発光態様で発光する。

【0030】

ガラス枠 1 8 の左右には、音響（例えば、効果音）を発するスピーカ 3 0 が備えられている。また、ガラス枠 1 8 の上方には照明ユニット 1 0 が備えられている。照明ユニット 1 0 の内部には、装飾装置が備えられている。

【0031】

照明ユニット 1 0 の右側には、遊技機 1 のエラー発生や前面枠 3 の開放をホール店員に通知するためのエラー報知 LED 2 9 が備えられている。

【0032】

前面枠 3 の下部の開閉パネル 2 0 には図示しない打球発射装置に遊技球を供給する上皿 2 1 が備えられている。さらに、固定パネル 2 2 には灰皿 1 5、下皿 2 3 及び打球発射装置の操作部 2 4 等が備えられている。下皿 2 3 には、下皿 2 3 に貯まった遊技球を排出するための下皿球抜き機構 1 6 が備えられる。前面枠 3 下部右側には、ガラス枠 1 8 を施錠するための鍵 2 5 が備えられている。

【0033】

また、遊技者が操作部 2 4 を回動操作することによって、打球発射装置は、上皿 2 1 から供給される遊技球を発射する。

【0034】

また、上皿 2 1 の上縁部には、遊技者からの操作入力を受け付けるためのセレクトスイッチ 4 0 及び操作スイッチ 4 1 が備えられている。

【0035】

遊技者がセレクトスイッチ 4 0 を操作することによって、表示装置 8（図 3 参照）における変動表示ゲームの演出内容を選択することができる。また、遊技者が操作スイッチ 4 1 を操作することによって、表示装置 8 における変動表示ゲームに、遊技者の操作を介入

10

20

30

40

50



させた演出を行うことができる。

【0036】

上皿21の右上部には、遊技者が遊技媒体を借りる場合に操作する球貸ボタン26、及び、カードユニット70からプリペイドカードを排出させるために操作される排出ボタン27が設けられている。これらのボタン26、27の間には、プリペイドカードの残高を表示する残高表示部28が設けられる。

【0037】

次に、カードユニット70について説明する。

【0038】

カードユニット70の下部には、プリペイドカード又は会員カード等のカードを挿入可能なカード挿入口71が設けられる。

【0039】

プリペイドカード又は会員カード等のカードには、当該カードの一意的識別子、当該カードの所有者(遊技者)の会員情報、及び残高等が記憶されている。会員情報には、カードの所有者の住所、氏名、年齢、及び職業等が含まれる。

【0040】

カード挿入口71にプリペイドカード又は会員カード等のカードが挿入された場合、図示しないカードリーダー・ライターによって、カードに記憶された情報が読み出される。そして、当該カードに記憶された残高が、遊技機1の残高表示部28及びカードユニット70の中央付近に設けられた残高表示部72に表示される。

【0041】

残高表示部72の上方には、紙幣を挿入可能な紙幣挿入口73が設けられる。紙幣挿入口73に挿入された紙幣の有価価値は、カードに残高として記憶される。

【0042】

紙幣挿入口73の上方には、動作表示部74が設けられる。動作表示部74は、カードユニット70の動作に対応した色で点灯する。

【0043】

次に、図2を参照しながら遊技機1の裏面側について説明する。図2は、本発明の第1の実施の形態の遊技機1の背面図である。

【0044】

遊技機1の裏面側、具体的には、前面枠3の裏面側には、中央に略正方形の開口部を有する枠状の裏機構盤310が取り付けられる。

【0045】

裏機構盤310の上部には、島設備に設けられた補給装置(図示省略)から補給された遊技球を貯留すると共に、貯留した遊技球を流下させる球貯留ユニット320が配設される。

【0046】

裏機構盤310の側部(図2中右側)には、球貯留ユニット320から流下してきた遊技球を、遊技機前面に配設された上皿21及び下皿23に払い出す球排出ユニット330が配設される。

【0047】

裏機構盤310の中央部には、遊技を統括的に制御する遊技制御装置100と、遊技制御装置100から送信される演出制御指令に基づいて変動表示ゲームの演出を制御する演出制御装置150とが配設される。

【0048】

遊技制御装置100には、図示しない検査装置に接続される検査装置接続端子107が配設される。

【0049】

裏機構盤310の下部には、遊技制御装置100から送信されるデータに基づいて球排出ユニット330の動作を制御し、遊技者に賞球を付与する払出制御装置(付与制御装置

10

20

30

40

50

) 210と、電源装置160とが配設される。

【0050】

払出制御装置210には、図示しない検査装置に接続される検査装置接続端子217及び払出制御装置210に発生したエラーの種類を数字で表示するエラーナンバー表示器222が配設される。

【0051】

また、電源装置160の右側の裏機構盤310には、遊技機1をカードユニット70に接続するためのカードユニット接続端子340が配設される。

【0052】

次に、遊技盤5について、図3を参照しながら説明する。図3は、本発明の第1の実施の形態の遊技盤5の正面図である。

10

【0053】

遊技盤5の表面には、ガイドレール55で囲われた略円形状の遊技領域51が形成される。遊技領域51は、遊技盤5の四方に各々設けられた樹脂製のサイドケース52及びガイドレール55によって構成される。遊技領域51の右下側のサイドケース52は、前面の中央部が黒色透明の証紙プレート53で覆われている。

【0054】

遊技領域51には、ほぼ中央に表示装置8が設けられるセンターケース300が配置される。表示装置8はセンターケース300に設けられた凹部に、センターケース300の前面より奥まった位置に取り付けられている。すなわち、センターケース300は表示装置8の表示領域の周囲を囲い、表示装置8の表示領域から突出して設けられている。

20

【0055】

また、遊技領域51の右下の領域には、図4で後述する特図表示器120及び普図表示器121を一体化した、図柄表示ユニット(一括表示器)45が備えられる。

【0056】

表示装置8は、例えば、LCD(液晶表示器)、CRT(ブラウン管)等で表示画面が構成されている。表示画面の画像を表示可能な領域(表示領域)には、複数の変動表示領域が設けられており、各変動表示領域に識別情報(特別図柄)や特図(特別図柄)変動表示ゲームを演出するキャラクタが表示される。表示画面の変動表示領域には、識別情報として割り当てられた三つの特別図柄が変動表示(可変表示)して特図変動表示ゲームが行われる。その他、表示画面には遊技の進行に基づく画像(例えば、大当り表示、ファンファーレ表示、エンディング表示等)が表示される。

30

【0057】

センターケース300の左側には、普通図柄始動ゲート31が設けられる。センターケース300の左下側には、三つの一般入賞口32が備えられ、センターケース300の右下側には、一つの一般入賞口32が備えられている。

【0058】

センターケース300の下方には、第1始動入賞口37と、第1始動入賞口37の直下に配設され、上部に逆「八」の字状に開いて遊技球が流入し易い状態に変換する一對の開閉部材を含む普通変動入賞装置33を備える第2始動入賞口34とが配置される。そして、遊技球が第1始動入賞口37又は第2始動入賞口34に入賞した場合には、補助遊技として特図変動表示ゲームが実行される。すなわち、第1始動入賞口37又は第2始動入賞口34に遊技球が入賞することが特図変動表示ゲームを開始するための始動条件となっており、第1始動入賞口37及び第2始動入賞口34が当該始動条件を発生させる変動始動入賞装置をなしている。なお、第1始動入賞口37に遊技球が入賞したことによって実行される特図変動表示ゲームを第1特図変動表示ゲームとし、第2始動入賞口34に遊技球が入賞したことによって実行される特図変動表示ゲームを第2特図変動表示ゲームとする。

40

【0059】

また、センターケース300に設けられた始動入賞口の下方には、表示装置8の作動結

50

果（特図変動表示ゲームの結果）によって遊技球を受け入れない状態と受け入れ易い状態とに変換可能な特別変動入賞装置（大入賞口）36が配設される。

【0060】

遊技機1では、図示しない発射装置から遊技領域51に向けて遊技球（パチンコ球）が打ち出されることによって遊技が行われる。打ち出された遊技球は、遊技領域51内の各所に配置された釘や風車等の方向転換部材によって転動方向を変えながら遊技領域51を流下する。そして、普通図柄始動ゲート31、一般入賞口32、第1始動入賞口37、第2始動入賞口34、又は特別変動入賞装置36に入賞するか、遊技領域51の最下部に設けられたアウト口39から排出される。

【0061】

また、第2始動入賞口34の状態には、普通変動入賞装置33の開閉によって、遊技球が入賞しやすい状態（入賞容易状態）と遊技球が入賞しにくい状態（非入賞容易状態）とがある。

【0062】

通常、普通変動入賞装置33が閉状態の場合には、第2始動入賞口34は、遊技球が入賞しにくい状態となる。普通図柄始動ゲート31を遊技球が通過することによって、普通図柄変動表示ゲームが実行され、普通図柄変動表示ゲームの結果が当たりとなると、普通変動入賞装置33が開状態に変換され、第2始動入賞口34は遊技球が入賞し易い状態となる。

【0063】

一般入賞口32への遊技球の入賞は、一般入賞口32に備えられた入賞口SW（スイッチ）32A～32N（図4参照）によって検出される。

【0064】

第1始動入賞口37への遊技球の入賞は第1特図始動SW（スイッチ）37A（図4参照）によって検出される。この遊技球の通過タイミングによって抽出された各種乱数は、遊技制御装置100内の特図1始動記憶領域に特別図柄入賞記憶として所定回数（例えば、最大で4回分）を限度に記憶される。

【0065】

また、第2始動入賞口34への遊技球の入賞は第2特図始動SW（スイッチ）34A（図4参照）によって検出される。この遊技球の通過タイミングによって抽出された各種乱数は、遊技制御装置100内の特図2始動記憶領域に特別図柄入賞記憶として所定回数（例えば、最大で4回分）を限度に記憶される。

【0066】

遊技制御装置100は、これらの特別図柄入賞記憶に基づき、図柄表示ユニット45を形成する特図表示器120（図4参照）にて、第1特図変動表示ゲーム及び第2特図変動表示ゲームを行う。

【0067】

ここで特図表示器120について説明する。特図表示器120は、図示しない第1特図図柄表示部、第1特図入賞記憶数表示部、第2特図図柄表示部、及び第2特図入賞記憶数表示部を有する。

【0068】

そして、遊技球が第1始動入賞口37に入賞して特別図柄入賞記憶が発生すると、第1特図入賞記憶数表示部に特別図柄入賞記憶の記憶数が表示され、第1特図図柄表示部にて第1特図変動表示ゲームが実行される。同様に、遊技球が第2始動入賞口34に入賞して特別図柄入賞記憶が発生すると、第2特図入賞記憶数表示部に特別図柄入賞記憶の記憶数が表示され、第2特図図柄表示部にて第2特図変動表示ゲームが実行される。

【0069】

なお、第1始動入賞口37への入賞が所定のタイミングでなされたとき（具体的には、入賞検出時の当り乱数値が当り値であるとき）には、第1特図変動表示ゲームの結果として表示図柄により特別結果態様が導出される。具体的には、第1図柄表示部にて、対応する特定の図柄（例えば、「1」～「9」までのいずれかの数字等）が停止表示される。な

10

20

30

40

50

お、第1特図変動表示ゲームがはずれの場合には、第1特図図柄表示部にて、はずれに対応する図柄（例えば「0」等）が停止表示される。

【0070】

同様に、第2始動入賞口34への入賞が所定のタイミングでなされたとき（具体的には、入賞検出時の当り乱数値が当り値であるとき）には、第2特図変動表示ゲームの結果として表示図柄により特別結果態様が導出される。具体的には、第2特図図柄表示部にて、対応する特定の図柄（例えば、「1」～「9」までのいずれかの数字等）が停止表示される。なお、第2特図変動表示ゲームがはずれの場合には、第2特図図柄表示部にて、はずれに対応する図柄（例えば「0」等）が停止表示される。

【0071】

そして、第1特図変動表示ゲームの結果として特別結果態様が導出された場合、あるいは、第2特図変動表示ゲームの結果として特別結果態様が導出された場合には、大当たり状態（特別遊技状態）が発生する。このとき、特別変動入賞装置36は、大入賞口ソレノイド38（図4参照）への通電によって、所定の時間（例えば、30秒）だけ、遊技球を受け入れない閉状態から遊技球を受け入れやすい開状態に変換される。すなわち、特別変動入賞装置36が所定の時間又は所定数の遊技球が入賞するまで大きく開くので、この間遊技者は多くの遊技球を獲得することができるという特典が付与される。

【0072】

なお、表示装置8においても、図柄表示ユニット45の特図表示器120で実行される第1特図変動表示ゲームと第2特図変動表示ゲームに同期して、飾り特図変動表示ゲームが実行される。このとき、前述した数字等で構成される特別図柄（識別情報）が左（第一特別図柄）、右（第二特別図柄）、中（第三特別図柄）の順に変動表示を開始して、飾り特図変動表示ゲームに関する画像が表示される。

【0073】

第1特図変動表示ゲーム若しくは第2特図変動表示ゲームが特別結果態様を導出する場合は、表示装置8でも特定態様の図柄が導出されて停止表示する。具体的には、表示装置8では、三つの特別図柄が揃った状態（大当たり図柄）で停止する。第1特図変動表示ゲーム及び第2特図変動表示ゲームが特別結果態様を導出しない場合は、表示装置8でも特定態様の図柄を停止表示しないように制御される。

【0074】

なお、本実施形態では、図柄表示ユニット45で実行される第1特図変動表示ゲーム及び第2特図変動表示ゲームと、表示装置8で実行される飾り特図変動表示ゲームのいずれをも特図変動表示ゲームとして扱うものとする。

【0075】

特別変動入賞装置36への遊技球の入賞は、カウントSW（スイッチ）36A（図4参照）によって検出される。

【0076】

普通図柄始動ゲート31への遊技球の通過は、普図始動SW（スイッチ）31A（図4参照）で検出される。この遊技球の通過タイミングによって抽出された普図乱数は、遊技制御装置100内の普図始動記憶領域に普通図柄入賞記憶として所定回数（例えば、最大で4回分）を限度に記憶される。遊技制御装置100は、この普通図柄入賞記憶に基づき、図柄表示ユニット45を形成する普図表示器121（図4参照）にて、普図変動表示ゲームを行う。

【0077】

ここで普図表示器121について説明する。普図表示器121は、図示しない普図図柄表示部と普図入賞記憶数表示部を有する。前述したように、遊技球が普通図柄始動ゲート31を通過して普通図柄入賞記憶が発生すると、この普図図柄表示部にて普図変動表示ゲームが実行される。また、普図入賞記憶数表示部には、普通図柄入賞記憶の記憶数が表示される。

【0078】

10

20

30

40

50

そして、遊技球が普通図柄始動ゲート 3 1 を通過して普通図柄入賞記憶が発生すると、普通図柄入賞記憶数表示部に普通図柄入賞記憶の記憶数が表示され、普通図柄表示部にて普通変動表示ゲームが実行される。

【 0 0 7 9 】

なお、普通図柄始動ゲート 3 1 の通過が所定のタイミングでなされたとき（具体的には、通過検出時の当り乱数値が当り値であるとき）には、普通図柄表示部に表示される普通図柄が当り状態で停止し、普通変動表示ゲームが当りとなる。このとき、普通変動入賞装置 3 3 は、普電ソレノイド 9 0（図 4 参照）への通電により、第 2 始動入賞口 3 4 への入口が所定の時間（例えば、0.5 秒～2.9 秒の範囲内で予め定められた時間）だけ開放するように変換され、遊技球の第 2 始動入賞口 3 4 への入賞が許容される。これによって、遊技球が第 2 始動入賞口 3 4 へ入賞しやすくなり、第 2 特図変動表示ゲームの始動が容易となる。

10

【 0 0 8 0 】

このようにして、一般入賞口 3 2、第 1 始動入賞口 3 7、第 2 始動入賞口 3 4、又は特別変動入賞装置 3 6 に遊技球が入賞すると、入賞した入賞口の種類に応じた数の賞球が払出制御装置 2 1 0 によって制御される払出ユニットから、前面枠 3 の上皿 2 1 又は下皿 2 3 に排出される。

【 0 0 8 1 】

なお、本実施形態のパチンコ遊技機は、特図変動表示ゲームの結果に対応して（厳密には、特図変動表示ゲームの結果として特図表示器 1 2 0 に停止表示される図柄に対応して）、以後の特図変動表示ゲームの当り確率が変化する場合があり、遊技状態は、常時、特図変動表示ゲームが低確率で大当りとなる低確率状態か、当該低確率状態よりも特図変動表示ゲームが大当りとなる確率の高い高確率状態のいずれかに設定されている。なお、低確率状態を通常状態や非確変遊技状態と称したり、高確率状態を確変遊技状態（確変状態）と称したりする場合もある。

20

【 0 0 8 2 】

さらに、本実施形態のパチンコ遊技機の遊技状態は、特図変動表示ゲームの結果に対応して、普通変動入賞装置 3 3 の開放頻度が変化する場合があり、遊技状態は、常時、普通変動入賞装置 3 3 の開放頻度が低い入賞抑制状態か、当該入賞抑制状態よりも普通変動入賞装置 3 3 の開放頻度が高い入賞促進状態のいずれかに設定されている。なお、入賞抑制状態を通常状態や非時短遊技状態と称したり、入賞促進状態を時短遊技状態（時短状態）と称したりする場合もある。

30

【 0 0 8 3 】

この入賞促進状態においては、普通変動表示ゲームの実行時間が入賞抑制状態における実行時間より短くなるように制御される（例えば、入賞抑制状態で 1 0 秒に対し、入賞促進状態で 1 秒）。これによって、単位時間当りの普通変動入賞装置 3 3 の開放回数が実質的に多くなるように制御される。

【 0 0 8 4 】

また、入賞促進状態においては、普通変動表示ゲームが当り結果となって普通変動入賞装置 3 3 が開放される場合に、開放時間が通常遊技状態の開放時間より長くなるように制御されてもよい（例えば、入賞抑制状態で 0.5 秒に対し、入賞促進状態で 2.9 秒）。また、入賞促進状態においては、普通図柄変動表示ゲームの 1 回の当り結果に対して、普通変動入賞装置 3 3 が 1 回ではなく、複数回（例えば、2 回）開放してもよい。さらに、入賞促進状態においては、普通変動表示ゲームの結果が当りとなる確率が入賞抑制状態より高くなるように制御してもよい。すなわち、入賞促進状態では、入賞抑制状態よりも普通変動入賞装置 3 3 の開放頻度が増加し、普通変動入賞装置 3 3 に遊技球が入賞しやすくなり、特図変動表示ゲームの始動が容易となる特典が付与される。

40

【 0 0 8 5 】

図 4 は、本発明の第 1 の実施の形態の遊技装置 6 のブロック図である。

【 0 0 8 6 】

50

遊技制御装置100は、遊技用マイコン(遊技用演算処理装置600)101、入力I/F(Interface)105、出力I/F(Interface)106及び検査装置接続端子107を備える。

【0087】

遊技用マイコン101は、CPU102、ROM(Read Only Memory)103及びRAM(Random Access Memory)104を備える。

【0088】

CPU102は、遊技を統括的に制御する主制御装置であって、遊技制御を行う。ROM103は、遊技制御のための不変の情報(プログラム、データ等)を記憶する。RAM104は、遊技制御時にワークエリアとして利用される。

10

【0089】

遊技制御装置100には、遊技用マイコン101に一意に設定された識別番号を出力することが可能な検査装置接続端子107が設けられている。検査装置接続端子107に図示しない検査装置を接続すると、検査装置は遊技機1を識別することができる。

【0090】

CPU102は、入力I/F105を介して各種検出器(第1特図始動SW37A、第2特図始動SW34A、普図始動SW31A、カウントSW36A、及び入賞口SW32A~入賞口SW32N、オーバーフローSW(スイッチ)109、球切れSW(スイッチ)110、及び枠開放SW(スイッチ)111)からの検出信号を受けて、大当り抽選等、種々の処理を行う。

20

【0091】

オーバーフロースイッチ109は、下皿23に遊技球が所定数以上貯留されていることを検出する。球切れスイッチ110は、球貯留ユニット320に配設され、球貯留ユニット320に貯留される遊技球が所定数以下になることを検出する。枠開放スイッチ111は、前面枠3の開放を検出する。

【0092】

また、CPU102は、出力I/F106を介して、普図表示器121、特図表示器120、普電SOL(ソレノイド)90、大入賞口SOL(ソレノイド)38、払出制御装置210及び演出制御装置150に指令信号を送信し、遊技を統括的に制御する。

【0093】

普図表示器121は、前述したように、普図変動表示ゲームが表示される。特図表示器120は、前述したように、第1特図変動表示ゲームと第2特図変動表示ゲームとが表示される。

30

【0094】

普電SOL90は、第2始動入賞口34に遊技球が入賞可能となるように、第2始動入賞口34に備えられた開閉部材で構成された普通変動入賞装置33を所定の時間だけ開放させる。

【0095】

大入賞口SOL38は、特別変動入賞装置36の大入賞口を所定の時間だけ、遊技球を受け入れない閉状態(遊技者に不利な状態)から遊技球を受け入れやすい開状態(遊技者に有利な状態)にする。

40

【0096】

また、遊技制御装置100は、遊技機1に関する情報を、外部情報端子108を介して、遊技店に設置された情報収集端末や遊技場内部管理装置(図示省略)に出力する。

【0097】

遊技制御装置100は、変動開始コマンド、客待ちデモコマンド、ファンファーレコマンド、確率情報コマンド、及びエラー指定コマンド等を、演出制御指令信号として、演出制御装置150へ送信する。

【0098】

次に、払出制御装置210及び演出制御装置150について説明する。

50

## 【 0 0 9 9 】

演出制御装置（表示制御装置）150は、遊技制御装置100から入力される各種信号に基づいて、エラー報知LED29、スピーカ30、発光により遊技演出を行う装飾部材9（図1）及び表示装置8を制御する。

## 【 0 1 0 0 】

演出制御装置150は、遊技用マイコン（遊技用演算処理装置600）151、ドライバ155、音回路156、及びVDP157を備える。

## 【 0 1 0 1 】

遊技用マイコン151は、CPU152、ROM153及びRAM154を備える。

## 【 0 1 0 2 】

CPU152は、演出制御を行う制御装置である。ROM153は、演出制御に必要な不変の情報（プログラム、データ等）を記憶している。RAM154は、演出制御時にワークエリアとして利用される。

## 【 0 1 0 3 】

ドライバ155は、CPU152からの指令により、エラー報知LED29及び装飾部材9を制御する。音回路156は、CPU152からの指令により、効果音を生成してスピーカ30から出力する。VDP157は、CPU152からの指令により、画像データを生成して表示装置8へ出力する。

## 【 0 1 0 4 】

払出制御装置210は、遊技制御装置100からの賞球指令信号に基づいて、払出装置の払出モータ220を駆動させ、賞球を払い出させるための制御を行う。また、払出制御装置210は、カードユニット70からの貸球要求信号に基づいて、遊技制御装置100が送信する排出指令信号に基づいて、払出装置の払出モータ220を駆動させ、貸球を払い出させるための制御を行う。

## 【 0 1 0 5 】

払出制御装置210は、遊技用マイコン（遊技用演算処理装置600）211、入力I/F（Interface）215、入出力I/F（Interface）216及び検査装置接続端子217を備える。

## 【 0 1 0 6 】

遊技用マイコン211は、CPU212、ROM213及びRAM214を備える。

## 【 0 1 0 7 】

CPU212は、払い出しを統括的に制御する制御装置であって、払出制御を司る。ROM213は、払出制御のための不変の情報（プログラム、データ等）を記憶している。RAM214は、払出制御時にワークエリアとして利用される。

## 【 0 1 0 8 】

CPU212は、入力I/F215を介して払出球検出スイッチ112、オーバーフロースイッチ109、球切れスイッチ110、エラー解除スイッチ223、税率設定スイッチ226、及び貸出料金設定スイッチ227からの入力を受ける。

## 【 0 1 0 9 】

エラー解除スイッチ223は、払出制御装置210にエラーが発生した場合に、遊技店の店員等が発生したエラーの原因を解消した際に、遊技店の店員等によって操作され、エラー状態を解除するためのスイッチである。

## 【 0 1 1 0 】

税率設定スイッチ226は、遊技球の貸し出しに対して課税される間接税の税率を設定するスイッチである。貸出料金設定スイッチ227は、貸し出される遊技球の有価価値を設定するためのスイッチである。

## 【 0 1 1 1 】

また、CPU212は、入出力I/F216を介して、払出モータ220、発射制御装置221、エラーナンバー表示器222、税率表示器224及び貸出料金表示器225に指令信号を送信する。また、CPU212は、入出力I/F216を介して遊技制御装置

10

20

30

40

50

100から送信された各種信号を受信する。

【0112】

払出モータ220は、実際に払出装置で遊技球を払い出すために駆動されるモータである。具体的には、払出モータ220は、1個の遊技球を貯留可能な凹部を所定個数有するスプロケットを回転させることによって、遊技球を払い出す。

【0113】

発射制御装置221は、遊技球を遊技盤5に発射するための発射装置を制御する。エラーナンバー表示器222は、払出制御装置210の裏面側に配設され、払出制御装置210で発生したエラーの種類を特定可能に表示する。

【0114】

税率表示器224は、払出制御装置210の裏面側に配設され、税率設定スイッチ226によって設定された間接税の税率を表示する。貸出料金表示器225は、払出制御装置210の裏面側に配設され、貸出料金設定スイッチ227によって設定された貸し出される遊技球の有価価値を表示する。

【0115】

電源装置160は、バックアップ電源161、RAMクリアスイッチ162を備える。遊技制御装置100、演出制御装置150、及び払出制御装置210は、電源装置160に接続される。

【0116】

バックアップ電源161は、停電時においても、遊技制御装置100、演出制御装置150、及び払出制御装置210に電源を供給する。なお、演出制御装置150には必ずしも電源を供給しなくてもよく、停電復帰後、遊技制御装置100からコマンドを送信するようにしてもよい。

【0117】

RAMクリアスイッチ162は、遊技制御装置100に備わるRAM104及び払出制御装置210に備わるRAM214に記憶されている情報を初期化するスイッチである。

【0118】

また、遊技機1に備わる球貸ボタン26が操作されると、カードユニット70は、プリペイドカード又は会員カード等のカードに記憶されている有価価値から貸し出される遊技球分の有価価値を減算して、減算した有価価値の値を遊技機1の残高表示部28に表示する。また、遊技機1に備わる排出ボタン27が操作されると、カードユニット70は、カード挿入口71に挿入されたカードを排出する。

【0119】

遊技制御装置100に備わる遊技用マイコン101と払出制御装置210に備わる遊技用マイコン211とは、暗号化された暗号化信号(暗号化データ)を双方向通信可能に接続される。また、暗号化されない非暗号化信号(平文データ)については、遊技制御装置100に備わる遊技用マイコン101から払出制御装置210に備わる遊技用マイコン211に単方向通信が可能に接続される。

【0120】

また、遊技制御装置100に備わる遊技用マイコン101と演出制御装置150に備わる遊技用マイコン151とは、暗号化されない非暗号化信号(平文データ)を遊技制御装置100から演出制御装置150への単方向で通信可能に接続される。

【0121】

なお、遊技制御装置100に備わる遊技用マイコン101、演出制御装置150に備わる遊技用マイコン151及び払出制御装置210に備わる遊技用マイコン211は、これらの接続に必要なポートを備えている。

【0122】

次に、遊技制御装置100に備わる遊技用マイコン101、演出制御装置150に備わる遊技用マイコン151及び払出制御装置210に備わる遊技用マイコン211(以下、総称して遊技用演算処理装置600という)について、図5を用いて詳細に説明する。

10

20

30

40

50



## 【 0 1 2 3 】

図 5 は、本発明の第 1 の実施の形態の遊技用演算処理装置（アミューズチップ）600 のブロック図である。

## 【 0 1 2 4 】

遊技用演算処理装置 600 はいわゆるアミューズチップ用の IC として製造され、遊技制御を行う遊技領域部 600A と情報管理を行う情報領域部 600B とに区分される。

## 【 0 1 2 5 】

まず、遊技領域部 600A は、CPU コア 601、ユーザプログラム ROM 602、HW パラメータ ROM 603、ユーザワーク RAM 604、ミラード RAM 605、外部バスインターフェース（I/F）606、バス切替回路 607、乱数生成回路 608、クロック生成回路 609、割込制御回路 610A、リセット回路 610B、アドレスデコーダ 611、出力制御回路 612、ブートブロック 613、復号化・ROM 書込回路 614、シリアル送信回路 615A、シリアル送信回路 615B、シリアル受信回路 625、暗号化送受信回路 616、及びバス 617 によって構成される。なお、シリアル送信回路 615A 及びシリアル送信回路 615B を総称して、シリアル送信回路 615 という。

10

## 【 0 1 2 6 】

CPU コア 601 は、図 4 の CPU 102、CPU 152 又は CPU 212 に相当する。ユーザプログラム ROM 602 は、図 4 の ROM 103、ROM 153 又は ROM 213 に相当する。また、ユーザプログラム ROM 602 及び HW パラメータ ROM 603 を総称して、ROM（不揮発性記憶手段）という。

20

## 【 0 1 2 7 】

ユーザワーク RAM 604 は、図 4 の RAM 104、RAM 154 又は RAM 214 に相当する。また、ユーザワーク RAM 604 及びミラード RAM 605 を総称して、RAM（揮発性記憶手段）という。

## 【 0 1 2 8 】

CPU コア 601 は、遊技制御のための演算処理を行う演算処理手段として機能する。ユーザプログラム ROM 602 は、制御プログラムを格納する。制御プログラムは、遊技用演算処理装置 600 が遊技制御装置 100 に備わる遊技用マイコン 101 である場合には、遊技の制御を行うための遊技制御プログラムである。また、遊技用演算処理装置 600 が払出制御装置 210 に備わる遊技用マイコン 211 である場合には、遊技球の払い出しを行うための払出制御プログラムである。さらに、遊技用演算処理装置 600 が演出制御装置 150 に備わる遊技用マイコン 151 である場合には、演出の制御を行うための演出制御プログラムである。

30

## 【 0 1 2 9 】

HW パラメータ ROM 603 は、正当性確認情報を格納する。正当性確認情報とは、遊技用演算処理装置 600 の正当性の簡易チェックを行う場合の情報であり、例えば、遊技機 1 の一意な識別子を示す固有 ID、メーカーコード（遊技機 1 の製造メーカー毎に割り振られた固有の製造メーカーの一意な識別子）、遊技機 1 のランク（1 種、2 種等）を示すランクコード、製造メーカーが遊技機 1 の種類に設定する機種コード、検査番号を示す検査コード、電源投入時に RAM をバックアップするか否かを示す RAM バックアップコード、税率設定スイッチ 226 によって設定された税率、貸出料金設定スイッチ 227 によって設定された貸出料金等である。また、HW パラメータ ROM 603 には、最初に貸出情報要求を送信した検査装置の一意な識別子である固有 ID が一つのみ記憶される。

40

## 【 0 1 3 0 】

第三者機関又は遊技機 1 の製造メーカーがユーザプログラム ROM 602 にプログラムを書き込む際に、正当性確認情報が HW パラメータ ROM 603 に書き込まれる。遊技用演算処理装置 600 は、電源立ち上がり時に、ユーザプログラム ROM 602 に書き込まれたプログラムが正当であるか否かについて簡易チェックを行うことができる。具体的には、遊技用演算処理装置 600 の電源立ち上がり時に、遊技用演算処理装置 600 自身が演算した演算値と、正当性確認情報（すなわち、第三者機関等によって予め設定された結果

50

値)とを比較判定することで、簡易的な遊技用演算処理装置600のチェックを行うことが可能になっている。

【0131】

ユーザワークRAM604は、遊技領域部600Aにおけるプログラムに基づく処理を実行する際にワークエリア(作業領域)として用いられるものである。このユーザワークRAM604には、バックアップ電源161(図4)からのバックアップ電源が供給されているので、遊技機1への電源供給が途絶えても、記憶データが保持されるように構成されている。ミラードRAM605は、クロックの立ち上がり時にユーザワークエリアに記憶された情報を複製し、複製した情報を記憶する(CPUコアがZ80の場合には、クロックの立ち上がり時に処理を実行するため、同期して動くことがないようにしている)。

10

【0132】

外部バスインターフェース606は、メモリリクエスト信号MREQ、入出力リクエスト信号IORQ、メモリ書き込み信号WR、メモリ読み出し信号RD及びモード信号MODEなどのインターフェースであり、また、バス切替回路607は、16ビットのアドレス信号A0~A15や8ビットのデータ信号D0~D7のインターフェースである。

【0133】

例えば、MODE信号をハイレベルにした状態で、アドレス信号A0~A15を順次にインクリメントしながら、データ信号D0~D7を加えると、ユーザプログラムROM602への書き込みモードとなって遊技機1の製造メーカ又は第三者機関によるプログラムの書き込みが可能になる。なお、書き込みモードはプログラムの書き込みを可能にするものであり、ブートブロック613に記憶されるブートプログラムを書き込みできるようにするものではない。

20

【0134】

また、ユーザプログラムROM602へのプログラムの書き込みが終了すると、HWパラメータROM603の所定領域に書込終了コードが記録(例えば、所定のコード若しくは所定ビットを物理的に切断することで記録)されるようになっており、HWパラメータROM603に書込終了コードが記録されている場合には、ユーザプログラムROM602への新たなプログラムの書き込みができないようになっている。

【0135】

乱数生成回路608は遊技の実行過程において遊技価値(例えば、大当り)を付加するか否か等に係わる乱数(乱数は、大当りの決定や停止時の図柄の決定等に使用)を生成するもので、一様性乱数を生成する数学的手法(例えば、合同法又はM系列法等)を利用している。なお、遊技用演算処理装置600が払出制御装置210に備わる遊技用マイコン211である場合には、乱数生成回路608は必要ない。

30

【0136】

クロック生成回路609は、遊技用演算処理装置600の外部から入力されたシステムクロック信号を分周して、遊技制御に必要な各種の周期的な信号を生成する。具体的には、CPUコア102を動作させるためのクロック信号、乱数生成回路608の乱数カウンタ値を更新するためのクロック信号、及びシリアル送受信を行う際のタイミング信号などを生成する。なお、クロック生成回路609は、CTC(Counter Timer Circuit)の機能を有しており、CPUコア102にて所定周期(例えば、4ミリ秒)のタイマ割込処理を行わせるためのタイマ割込信号も生成している。

40

【0137】

割込制御回路610Aは、所定の割り込み条件の成立に基づいて、割り込みの発生をCPUコア102に知らせる。なお、所定の割り込み条件は、遊技用演算処理装置600の外部から割込要求信号(INT信号、NMI信号)が入力された場合や、クロック生成回路609のCTCからタイマ割込信号が出力された場合などが相当する。

【0138】

CPUコア102は、INT信号やNMI信号が入力された場合には、現在実行中の処理を中断して予め定められた割込処理を実行する。なお、INT信号による割込(マスク

50

ブル割込)は、プログラムの設定により禁止できるが、NMI信号による割込(ノンマスクブル割込)はプログラムの設定により禁止できない。また、CPUコア102は、タイマ割込信号が出力された場合には、図29に示すタイマ割込処理を実行する。

**【0139】**

また、リセット回路610Bは、外部から入力されたリセット信号(RST0端子からの信号RST0)を検出すると、遊技用演算処理装置600の内部に備えられた各回路とCPUコア102にリセット信号を伝達する。

**【0140】**

なお、リセット回路610Bは、セキュリティ回路630(図11参照)としても機能する。このセキュリティ回路630は、遊技用演算処理装置600の内部に備えられた各回路に対して異常なアクセスが発生したことを検出する機能を有しており、異常を検出するとCPUコア102にリセット信号を伝達する。

10

**【0141】**

アドレスデコーダ611は、内蔵デバイス及び内蔵コントロール/ステータスレジスタ群のロケーションをメモリマップドI/O方式及びI/OマップドI/O方式によりデコードする。

**【0142】**

出力制御回路612は、アドレスデコーダ611からの信号制御を行って外部端子より8ビットのチップセレクト信号(CS0~CS7)を外に出力するとともに、遊技用演算処理装置600の内部に備えた回路を選択するチップセレクト信号を発生する機能を有する。ブートブロック613は、ブートプログラムを記憶し、電源投入時に遊技用演算処理装置600の初期化に係わる処理を行う。

20

**【0143】**

復号化・ROM書込回路614は、ユーザプログラムROM602及びHWパラメータROM603への書込みモードの際に使用されるもので、モード信号MODEが[H]レベルになっている間、バス切替回路607を介してアドレス信号A0~A15やデータ信号D0~D7を取り込み、そのデータ信号D0~D7に含まれる情報(暗号化されたプログラム及び暗号化された変更後の固有ID)を復号化処理した後、バス617を介してユーザプログラムROM602及びHWパラメータROM603に出力する(書き込む)。

**【0144】**

シリアル送信回路615A及びシリアル送信回路615Bは、暗号化されていない平文データを送信するための回路である。シリアル送信回路615Aは、SIO TX0端子を介して演出制御装置150に接続される。また、シリアル送信回路615Bは、SIO TX1端子を介して払出制御装置210に接続される。シリアル受信回路625は、暗号化されていない平文データを、SIO RX端子を介して受信するための回路である。

30

**【0145】**

暗号化送受信回路616は、NJLINK端子を介して暗号化された暗号化データを送受信する回路である。例えば、遊技制御装置100から払出制御装置210にデータを送信する場合に使用される、また、遊技制御装置100と払出制御装置210の間は、NJLINK接続で接続され、暗号化送受信回路616には、NJLINK信号線が接続される。暗号化送受信回路616は、NJLINK信号線を介してデータを送受信する。

40

**【0146】**

バス617はデータバス(図11のデータバス660)、アドレスバス(図11のアドレスバス650)及び制御バスを含むものであり、情報領域部600Bまで延びている。

**【0147】**

次に、遊技用演算処理装置600における情報管理を行う情報領域部600Bは、HPGプログラムROM618、IDプロパティメモリ619、バスモニタ回路620、HPGワークRAM621、制御回路622、外部通信制御回路623、バス624、及び遊技領域部600Aから延びるバス617の一部を含んで構成される。

**【0148】**

50

H P GプログラムROM 6 1 8には、各種検査動作を行うH P Gプログラムが格納される。

【0 1 4 9】

I Dプロパティメモリ 6 1 9には、図示しない検査装置から外部通信制御回路 6 2 3を介して受信した要求に基づいて、H WパラメータROM 6 0 3に記憶されている情報を図示しない検査装置にすぐに出力できるように、遊技用演算処理装置 6 0 0の電源投入時（システムリセット時）にH Wパラメータに記憶されている情報を複製して記憶する。なお、I Dプロパティメモリ 6 1 9は、遊技領域部 6 0 0 A側及び情報領域部 6 0 0 B側の双方よりアクセスが可能な構成になっている。

【0 1 5 0】

バスモニタ回路 6 2 0は、情報領域部 6 0 0 B側より遊技領域部 6 0 0 A側のバス 6 1 7の状態監視及び制御を行う。ここでの制御とは、H WパラメータROM 6 0 3の内容をI Dプロパティメモリ 6 1 9に複写する際のタイミング制御や、ユーザプログラムROM 6 0 2に格納されたプログラムを外部に出力する際（遊技領域部 6 0 0 A側のバス 6 1 7を開放してユーザプログラムROM 6 0 2からプログラムを読み込んで情報領域部 6 0 0 B側より外部に出力する際）のタイミング制御である。なお、プログラムは、外部通信制御回路 6 2 3で暗号化されてから出力される。

【0 1 5 1】

H P GワークRAM 6 2 1は、情報領域部 6 0 0 Bにおけるプログラムに基づく処理を実行する際にワークエリア（作業領域）として用いられるものである。

【0 1 5 2】

制御回路 6 2 2は情報領域部 6 0 0 B側を制御するもので、バッファメモリを有している。制御回路 6 2 2は、例えば、バスモニタ回路 6 2 0を介してC P Uコア 1 0 2の動作を監視し、非動作中に遊技領域部 6 0 0 AのユーザワークRAM 6 0 4に記憶された内容をミラードRAM 6 0 5へコピーする。また、図示しない検査装置からの要求に应答して情報領域部 6 0 0 BのI Dプロパティメモリ 6 1 9の内容を外部へ転送したり、プログラム要求に应答してバスモニタ回路 6 2 0を介してユーザプログラムROM 6 0 2内のプログラムを外部へ転送したりする。制御回路 6 2 2のメモリは、転送時のタイミング調節のために用いられる。

【0 1 5 3】

外部通信制御回路 6 2 3は図示しない検査装置との通信を行うもので、例えば、外部からの指令に基づいて遊技用演算処理装置 6 0 0内に格納されている情報（例えば、固有I D、プログラム、実払出数等）を暗号化した後、外部へ転送する等の処理を行う。

【0 1 5 4】

遊技用演算処理装置 6 0 0では、遊技領域部 6 0 0 Aと情報領域部 6 0 0 Bがバスモニタ回路 6 2 0を介して独立して動作する。すなわち、情報領域部 6 0 0 B側は遊技領域部 6 0 0 AにおけるC P Uコア 1 0 2の作動に関係なく（プログラム実行に関係なく）動作可能である。

【0 1 5 5】

なお、図 5では図示されていないが、遊技用演算処理装置 6 0 0には、図 1 1にて後述するRAMアクセス規制回路（更新規制手段） 6 4 0を備えている。

【0 1 5 6】

図 6は、本発明の第 1の実施の形態の遊技制御装置 1 0 0におけるシリアル送信回路 6 1 5の構成例を示すブロック図である。

【0 1 5 7】

シリアル送信回路 6 1 5は、送信シリアルチャンネル設定レジスタ 6 3 3、送信データステータスレジスタ 6 3 1、送信制御レジスタ 6 3 2、送信データレジスタ 6 3 5（送信データバッファレジスタ 6 3 5 A、送信データシフトレジスタ 6 3 5 B）、ポーレート生成回路（送信速度設定手段） 6 3 4を含んで構成される。

【0 1 5 8】

10

20

30

40

50

シリアル送信回路 6 1 5 は、出力制御回路 6 1 2 から入力された選択信号に基づいて、送信先を選択する。また、リセット信号の入力を受け付けると、各種レジスタに設定された値を 0 クリアする。

【 0 1 5 9 】

送信シリアルチャンネル設定レジスタ 6 3 3 は、データ送信時の通信速度及び通信フォーマットを指定するレジスタである。送信シリアルチャンネル設定レジスタ 6 3 3 には、後述するメイン処理におけるシリアル通信設定処理によって各値が設定される。

【 0 1 6 0 】

図 7 は、本発明の第 1 の実施の形態の送信シリアルチャンネル設定レジスタ 6 3 3 の構成例を示す図である。図 7 に示すように、送信シリアルチャンネル設定レジスタ 6 3 3 は、16 ビットで構成されており、ビット 0 ~ 15 はすべて書き込み / 読み出し可能とされる。

10

【 0 1 6 1 】

送信シリアルチャンネル設定レジスタ 6 3 3 において、ビット 0 ~ 12 には、送信ボーレート（通信速度）を算出するためのボーレート設定値（例えば、分周比）が設定される。

【 0 1 6 2 】

具体的には、遊技用演算処理装置 6 0 0 に入力されるシステムクロック（MCLK）の周波数（単位 Hz：ヘルツ）を 32 で除した値を、さらに、このボーレート設定値で除した値が、送信ボーレート（1 秒間に送信されるデータのビット数）として設定される。例えば、ボーレート設定値として 100 を設定すると、システムクロックの周波数が 20 MHz であった場合には、送信ボーレートは、 $20,000,000 \div 32 \div 100 = 6250$  (bps) となる。

20

【 0 1 6 3 】

ビット 13 には、送信データ長を 8 ビットとする場合に “0” が設定され、9 ビットとする場合に “1” が設定される。

【 0 1 6 4 】

ビット 14 には、送信データにパリティを付加しない場合に “0” が設定され、パリティを付加する場合に “1” が設定される。ビット 15 には、送信データに付加するパリティを偶数パリティとする場合に “0” が設定され、奇数パリティとする場合に “1” が設定される。なお、ビット 15 は、ビット 14 に “1”（パリティ有り）が設定されている場合に有効となる。

30

【 0 1 6 5 】

なお、遊技用演算処理装置 6 0 0 にリセット信号（RST0）が入力されると、シリアル送信回路 6 1 5 がリセットされ、送信シリアルチャンネル設定レジスタ 6 3 3 の全ビットが “0” に設定される（図 6 等も参照）。

【 0 1 6 6 】

図 6 の説明に戻り、送信制御レジスタ 6 3 2 は、シリアル送信回路 6 1 5 の動作を制御するためのレジスタである。

【 0 1 6 7 】

図 8 は、本発明の第 1 の実施の形態の送信制御レジスタ 6 3 2 の構成例を示す図である。図 8 に示すように、送信制御レジスタ 6 3 2 は、例えば 8 ビットで構成され、ビット 6 は読み出し専用とされ、他のビット 0、4、5、7 は書き込み / 読み出し可能とされる。なお、本実施形態では送信制御レジスタ 6 3 2 のビット 1 ~ 3 は未使用としているため、図 8 では記載を省略している。

40

【 0 1 6 8 】

送信制御レジスタ 6 3 2 のビット 0 には、送信回路（シリアル送信回路 6 1 5）を初期化する場合に “1” が設定される。送信回路が初期化されると送信データレジスタ 6 3 5 のデータも含めて全てのレジスタが初期値とされる。

【 0 1 6 9 】

50

ビット4には、送信データレジスタ635（送信データシフトレジスタ635B）からのデータ送信を禁止する場合に“0”が設定され、データ送信を許可する場合に“1”が設定される。

【0170】

ビット5には、送信データレジスタ635が空になったときに送信割り込みを要求しない場合に“0”が設定され、送信割り込みを要求する場合に“1”が設定される。

【0171】

ビット6には、送信割り込み要求が発生しているか否か（送信割り込み状態）を示す値が設定される。ビット6に“0”が設定されている場合には送信割り込みを要求していない状態であることを示し、“1”が設定されている場合には送信割り込みを要求している状態であることを示す。

10

【0172】

ビット7には、送信データレジスタのビット8の値（データ長が9ビットの場合）が設定される。

【0173】

なお、遊技用演算処理装置600にリセット信号（RST0）が入力されると、シリアル送信回路615がリセットされ、送信制御レジスタ632の全ビットが“0”に設定される（図6等も参照）。その結果、送信制御レジスタ632のビット4が“0”になるので、送信データレジスタ635（送信データシフトレジスタ635B）からのデータ送信が禁止され、データ出力がオフされた状態になる。

20

【0174】

図6の説明に戻り、送信データステータスレジスタ631は、送信データレジスタ635の状態を示すレジスタである。CPU102は、送信データステータスレジスタ631の設定値によって、送信データレジスタ635の状態を確認することができる。

【0175】

図9は、本発明の第1の実施の形態の送信データステータスレジスタ631の構成例を示す図である。送信データステータスレジスタ631は、例えば8ビットで構成され、ビット0～5、7はすべて読み出し専用とされる。なお、本実施形態では送信データステータスレジスタ631のビット6は未使用としているため、図9では省略している。

【0176】

送信データステータスレジスタ631において、ビット0～5には、送信データの残量を示す値が設定される。例えば、ビット0～5に“00h”（16進数の“0”）が設定されていると送信データがないことを示し、“01h”が設定されていると送信データが1バイト残っていることを示し、“20h”が設定されていると送信データが32バイト残っていることを示す。

30

【0177】

ビット7には、送信データレジスタ635におけるデータの送信状態を示す値が設定される。ビット7に“1”が設定されているとデータを送信していない状態であることを示し、“0”が設定されているとデータを送信している状態であることを示す。

【0178】

なお、遊技用演算処理装置600にリセット信号（RST0）が入力されると、シリアル送信回路615がリセットされ、送信データステータスレジスタ631の全ビットが“0”に設定される（図6等も参照）。その結果、送信データステータスレジスタ631のビット0～5が“00h”となり、送信データの残量がない状態となる。

40

【0179】

図6の説明に戻り、送信データレジスタ635は、シリアル送信回路615が送信するデータを格納するレジスタである。送信データレジスタ635は、例えば、1段の送信データシフトレジスタ635Bと、31段の送信データバッファレジスタ635Aで構成される。

【0180】

50

図10は、本発明の第1の実施の形態の送信データレジスタ635（1段分）の構成例を示す図である。1段の送信データレジスタ635は、例えば8ビットで構成され、ビット0～7はすべて書き込み専用とされる。

【0181】

この送信データレジスタ635には、タイマ割込処理で生成される制御指令データが格納され、送信制御レジスタのビット4に“1”（送信許可）が設定されていれば、格納された制御指令データは自動的に演出制御装置150に送信される。

【0182】

制御指令データは、例えば、1バイトのモードデータと1バイトのアクションデータの2バイトで構成されるので、2段の送信データレジスタ635に1つの制御指令データが格納されることとなる。そして、本実施形態では、送信データレジスタ635を32段で構成しているので、1回のタイマ割込処理で最大16の制御指令データが生成される場合、これをすべて送信データレジスタ635に格納することができる。

【0183】

ただし、CPU102によって、新たな送信データを送信データレジスタ635に格納できるのは、送信データステータスレジスタ631のビット0～5の値（送信データの残量を示す値）が“00h”～“1Fh”の場合（送信データレジスタ635に、0～31バイトの未送信データが残っている場合）に限られる。

【0184】

送信データステータスレジスタ631のビット0～5の値が“20h”の場合は、送信データレジスタ635に空きがないので、CPU102によって送信データレジスタ635に書き込もうとされたデータは廃棄される。これにより、送信データレジスタ635が満杯のときは、誤ってCPU102によるデータ書き込みが発生しても、既に格納されている送信データレジスタ635のデータが破壊されないようになっている。

【0185】

なお、遊技用演算処理装置600にリセット信号（RST0）が入力されると、シリアル送信回路615がリセットされ、送信データレジスタ635の全ビットが“0”に設定される（図6等も参照）。

【0186】

図6の説明に戻り、ポーレート生成回路634は、クロック生成回路609から分周回路629を介して出力されるクロック信号（遊技用演算処理装置600に入力されるシステムクロック（MCLK）を分周した信号）及び送信シリアルチャンネル設定レジスタ633に設定されている設定値（ポーレート設定値）に基づいて、シリアル送信回路615が用いる送信ポーレートを生成する。このとき、ポーレート生成回路634は、クロック信号及びポーレート設定値に基づいて、前述の計算式を用いて送信ポーレートを求める。また、分周回路629は、CPU102にも分周されたクロック信号を入力する。なお、分周回路629は、クロック生成回路609に含まれるように構成してもよい。

【0187】

シリアル送信回路615では、送信許可の設定（送信制御レジスタ632のビット4を“1”）がなされた後、送信するデータを送信データレジスタ635（送信データバッファレジスタ635A）に書き込む、又は、送信するデータを送信データレジスタ635に書き込んだ後、送信許可の設定がなされると、自動的に送信が開始される。送信が開始されると、送信データバッファレジスタ635Aのデータが送信データシフトレジスタ635Bに転送され、送信データシフトレジスタ635Bからシリアル変換されて、最下位ビット（ビット0）から1ビットずつ順次出力される。そして、データの送信が完了すると送信データシフトレジスタ635Bは空になるので、送信データバッファレジスタ635Aに書き込まれている次のデータが送信データシフトレジスタ635Bに転送され、出力される。

【0188】

したがって、シリアル送信回路615では、送信データレジスタ635（送信データシ

10

20

30

40

50

フトレジスタ635B、送信データバッファレジスタ635A)に書き込まれたデータ(制御指令データ)が、演出制御装置150に1ビットずつ順次送信されることとなる。

【0189】

このように、シリアル送信回路(制御指令送信手段)615は、送信データ(例えば、制御指令データ)を格納する送信データレジスタ635を備え、送信データレジスタ635に送信データが格納されると、遊技制御装置100から演出制御装置150へ向かう方向に、格納された送信データを1ビットずつ順次送信する(いわゆるシリアル通信)ように構成されている。

【0190】

具体的には、送信データレジスタ635は、格納されたデータをすぐに送信する送信データシフトレジスタ635Bと、格納されたデータを保持するとともに、送信データシフトレジスタ635Bがデータを格納可能な状態(データの送信が完了した状態)となったときに、保持しているデータを送信データシフトレジスタ635Bに転送する送信データバッファレジスタ635Aと、で構成される。

10

【0191】

これにより、従来のパラレル通信では必須とされていたタイマ割込処理における制御指令データの送信処理を省略できるので、CPU102の負担を軽減することができる。

【0192】

また、シリアル通信とすることで、制御指令データを送信するための配線本数を比較的少なくすることができる。

20

【0193】

また、遊技制御装置100と演出制御装置150との間の通信は、遊技制御装置100から演出制御装置150へのみデータを送信可能な単方向通信とされ、遊技制御装置100にデータは入力されないので、不正が行われることを防止できる。

【0194】

なお、本実施形態においては、送信データレジスタ635に最大で32バイトの送信データが格納可能であるが、この32バイトのデータが、1回のタイマ割込処理において全て出力できるように、ボーレート設定値(送信シリアルチャンネル設定レジスタ633のビット0~12)の値が設定されている。

【0195】

具体的には、送信データレジスタ635から出力される1バイトあたりのデータ送信に必要な時間 $T_b$ と、タイマ割込信号の発生周期 $F$ と、送信データレジスタ635に格納できるデータの上限バイト数 $B$ との関係が、 $F/B > T_b$ となるように、ボーレート設定値を設定して送信の速度を決めればよい。

30

【0196】

例えば、タイマ割込信号の発生周期 $F = 4$ ミリ秒で、送信データレジスタ635に格納できるデータの上限バイト数 $B = 32$ バイトであれば、 $F/B = 4000/32 = 125$ マイクロ秒よりも短くなるように $T_b$ の値を決定し、遊技用演算処理装置600に入力されるシステムクロック(MCLK)の周波数を考慮したうえで、ボーレート設定値の値を定めればよい。

40

【0197】

このような構成とすることで、シリアル送信回路(制御指令送信手段、送信手段)615は、1回のタイマ割込処理において生成される一連の制御指令データをすべて格納することが可能となり、タイマ割り込み毎に生成される制御指令データを確実に送信することができる。

【0198】

図11は、本発明の第1の実施の形態の遊技制御装置100に備わる遊技用演算処理装置(アミューズチップ)600とその周辺のブロック図である。

【0199】

遊技用演算処理装置600は、セキュリティ回路630、CPUコア102(図11で

50



は601)、RAMアクセス規制回路640、ユーザワークRAM104(図11では604)、アドレスデコーダ611、出力制御回路612、及び、ユーザプログラムROM103(図11では602)を備える。

【0200】

なお、遊技用演算処理装置600に備わるこれらの回路等は、アドレスバス650及びデータバス660を介して接続されている。アドレスバス650は、A0~A15の16ビットの信号線によって構成され、データバス660は、D0~D7の8ビットの信号線によって構成される。

【0201】

また、遊技制御装置100は、演出制御装置150に接続されるシリアル送信回路615A、及び、払出制御装置210に接続されるシリアル送信回路615Bを備える。

10

【0202】

なお、演出制御装置150及び払出制御装置210には、シリアル受信回路625(図5)に相当する受信回路が備えられていて、シリアル送信回路615A、615Bから出力されたシリアルのデータを、それぞれが受信する構成となっている。

【0203】

遊技用演算処理装置600に電源が投入される際には、RST0端子(図5)を介して電源装置160からリセット信号(起動信号)が入力され、リセット回路610B(図5)が作動する。

【0204】

セキュリティ回路630は、このリセット信号が入力されるとHWパラメータROM603に記憶された正当性確認情報を用いて、セキュリティチェック処理を実行する。このセキュリティチェック処理は、ユーザプログラムROM103に記憶されたプログラムの正当性の判定を行う処理である。

20

【0205】

セキュリティ回路630は、このセキュリティチェック処理を実行している間は、CPUコア102のリセット端子(RES0(負論理))にリセット信号を継続して出力することで、CPUコア102の起動を待機させる。

【0206】

CPUコア102は、前述のリセット端子(RES0(負論理))と、書込指令出力端子(WR(負論理))、及び読出指令出力端子(RD(負論理))を備える。リセット端子はセキュリティ回路630に接続されており、遊技用演算処理装置600にリセット信号が入力されると、前述のように、セキュリティチェック処理を実行している間、CPUコア102に対するリセット信号がリセット端子に入力される。

30

【0207】

CPUコア102のリセット端子にリセット信号が入力されると、CPUコア102は、CPUコア102に備わるレジスタ(REG)を初期化する。

【0208】

また、CPUコア102がユーザワークRAM104にデータの書き込みを指令する書込指令を出力する場合には、CPUコア102の書込指令出力端子からは所定値よりも低い電圧のローレベルの信号が出力される。同様に、CPUコア102がユーザワークRAM104からデータの読み出しを指令する読出指令を出力する場合には、CPUコア102の読出指令出力端子からは所定値よりも低い電圧のローレベルの信号が出力される。

40

【0209】

つまり、書込指令出力端子及び読出指令出力端子は、通常電圧がハイレベルに維持されており、ユーザワークRAM104への読み書きを行うときにのみ電圧がローレベルになる。

【0210】

また、CPUコア102は、アドレスバス650に指定したアドレスを出力し、データバス660を介して指定したアドレスに格納されたデータを入出力する。

50

## 【0211】

まず、ユーザワークRAM104のデータの読み出しについて説明する。

## 【0212】

CPUコア102から、ユーザワークRAM104の読出指令入力端子(RD(負論理))に読出指令が入力されると、アドレスバス650及びデータバス660を介してCPUコア102に読出データが出力される。

## 【0213】

このとき、CPUコア102からアドレスバス650へは、ユーザワークRAM104のアドレスが出力され、アドレスデコード611からユーザワークRAM104のチップ選択端子(所謂CS端子に相当、図示は略)に選択信号が入力されることによって、ユーザワークRAM104が選択される。次いで、選択されたユーザワークRAM104は、アドレスバス650が指定する記憶領域のデータをデータバス660へ出力する。次いで、CPUコア102は、データバス660へ出力されたデータを内部へ取り込む。このような手順により、CPUコア102はユーザワークRAM104からデータを読み出す。

10

## 【0214】

次に、ユーザワークRAM104へのデータの書き込みについて説明する。

## 【0215】

CPUコア102に備わる書込指令出力端子は、RAMアクセス規制回路640のORゲート回路642に備わる二つの入力端子のうち一方の入力端子に接続される。ORゲート回路642の他方の入力端子は、RAMアクセス規制回路640のフリップフロップ回路641の出力端子(Q(負論理))に接続され、ORゲート回路642の出力端子は、ユーザワークRAM104の書込指令入力端子(WR(負論理))に接続されている。

20

## 【0216】

また、ユーザワークRAM104の書込指令入力端子に所定値以下の電圧であるローレベルの信号が入力されると、ユーザワークRAM104への書き込みが許容される。

## 【0217】

このため、ORゲート回路642の二つの入力端子にそれぞれローレベルの信号が入力されなければ、ユーザワークRAM104への書き込みが許容されない。言い換えれば、ORゲート回路642の少なくとも一方の入力端子にハイレベルの信号が入力されていると、ユーザワークRAM104への書き込みが規制(禁止)される。

30

## 【0218】

ここで、RAMアクセス規制回路640のフリップフロップ回路641について説明する。

## 【0219】

フリップフロップ回路641は、D型のフリップフロップ回路であり、入力端子として、データ端子(D)、リセット端子(R(負論理))、及び出力イネーブル端子(OE(負論理))を備えるとともに、出力端子(Q(正論理)、Q(負論理))を備える。

## 【0220】

データ端子には、データバス660を構成する信号線D0~D7のうち所定の一本の信号線(例えば、D0)が接続されている。

40

## 【0221】

リセット端子には電源装置160からリセット信号線が接続され、リセット信号が入力されるとリセット端子はローレベルとなる。このときフリップフロップ回路641は、出力端子Q(正論理)からローレベルの信号を出力させ、出力端子Q(負論理)からハイレベルの信号を出力させる。出力端子Q(正論理)からの出力と、出力端子Q(負論理)からの出力は、相互に反転するレベルとなっている。

## 【0222】

また、出力イネーブル端子は、出力制御回路612から送信された出力イネーブル信号が入力される。出力イネーブル信号がハイレベルの場合には、出力端子から信号の出力が可能な状態となる。

50

## 【0223】

このフリップフロップ回路641に備えた出力端子Q（負論理）からの信号レベルは、CPUコア102によって、自在に設定できるようになっている。この設定は、CPUコア102が、フリップフロップ回路641に割り当てられたアドレスの記憶領域に所定のデータを書き込むことで実現される。

## 【0224】

具体的には、CPUコア102によってフリップフロップ回路641に割り当てられたアドレスの記憶領域にデータを書き込む処理が行われると、CPUコア102からアドレスバス650へは、フリップフロップ回路641のアドレスが出力される。次に、アドレスデコーダ611から、出力制御回路612を介して、フリップフロップ回路641の出カインーブル端子にクロック信号が入力され、出カインーブル端子の電圧レベルが立ち上がり、ハイレベルとなる。

10

## 【0225】

このときフリップフロップ回路641は、データ端子に入力されている信号を取り込んで、取り込んだ信号を出力端子Q（正論理）から出力し、取り込んだ信号の反転値を出力端子Q（負論理）から出力する。

## 【0226】

また、フリップフロップ回路641は、出力制御回路612がクロック信号の入力を終了した場合には、出カインーブル端子の電圧レベルは立ち下がりローレベルとなり、出力端子Q（正論理）及び出力端子Q（負論理）の電圧レベルを保持する。

20

## 【0227】

また、出力端子Q（負論理）は、ORゲート回路642の入力端子に信号を出力する。出力端子Q（正論理）には何も接続されない。

## 【0228】

次に、フリップフロップ回路641の入力状態に応じた各種動作について説明する。

## 【0229】

フリップフロップ回路641は、前述したように、出カインーブル端子の電圧レベルの立ち上がり、つまり出カインーブル信号の入力開始時に、データ端子の電圧レベルを読み取り、読み取った電圧レベルの反転値を出力端子Q（負論理）から出力する。

## 【0230】

30

一方、フリップフロップ回路641は、出カインーブル端子の電圧レベルの立ち下がり、つまり、出カインーブル信号の入力終了時に、出カインーブル端子の電源レベルの立ち上がり時の出力端子Q（負論理）からの出力を保持する。

## 【0231】

出力端子Q（負論理）からハイレベルの信号がORゲート回路642の入力端子に出力されていると、ORゲート回路642の他方の入力端子にローレベル及びハイレベルのいずれの信号が入力されても、ORゲート回路642の出力端子からはハイレベルの信号が出力される。

## 【0232】

このため、フリップフロップ回路641の出力端子Q（負論理）からハイレベルの信号が出力されていれば、ORゲート回路642の他方の入力端子に書込指令信号が入力されても（当該他方の入力端子にローレベルの信号が入力されても）、ユーザワークRAM104の書込指令入力端子にはローレベルが入力されなくなり、RAM書込禁止状態が発生する。

40

## 【0233】

RAMアクセス規制回路640をRAM書込禁止状態にするかRAM書込許可状態にするかは、クロック信号がフリップフロップ回路641に入力されたときのフリップフロップ回路641のデータ端子に入力される電圧レベル、又はリセット信号の入力の有無に基づく。

## 【0234】

50

前述のようにCPUコア102は、出力制御回路612を制御してクロック信号の出力を制御でき、データバス660の信号線の出力も制御できるので、フリップフロップ回路641の出力端子Q(負論理)から出力される信号は、CPUコア102によって制御可能である。言い換えると、CPUコア102は、データバス660の信号レベルを制御することによってRAMアクセス規制回路640の書込状態を制御できる。

【0235】

さらに、前述のようにフリップフロップ回路641のリセット端子にリセット信号が入力された場合には、フリップフロップ回路641は、出力端子Qの電圧レベルをローにするため、出力端子Q(負論理)の電圧レベルはハイになる。このため、フリップフロップ回路641にリセット信号が入力された場合には、RAMアクセス規制回路640では、RAM書込禁止状態が発生することになる。

10

【0236】

前述のように、出力制御回路612が払出制御装置210に接続されるシリアル送信回路615Bにクロック信号を入力すると、シリアル送信回路615Bは、クロック信号が入力されたタイミングで、データバス617からデータを読み取り、読み取ったデータを送信データバッファレジスタ635Aに格納する。そして、送信データバッファレジスタ635Aに格納されたデータを送信データシフトレジスタ635Bに格納し、払出制御装置210に順次出力する。

【0237】

なお、前述したセキュリティ回路630、RAMアクセス規制回路640、及びシリアル送信回路615の起動(リセット)は、電源装置160からのリセット信号を、前述のリセット回路610B(図5)を介して受け入れた場合に実行される。ただし、電源装置160からのリセット信号は、必ずしもリセット回路610Bを介して各回路に入力される必要はなく、リセット回路610Bを経由しない別個の信号線を介して各回路に入力されるような構成でもよい。

20

【0238】

なお、シリアル送信回路615(615A、615B)においては、図7~図10の説明で前述したように、リセット信号によって、演出制御装置150や払出制御装置210へのデータ出力がオフ状態となる。

【0239】

また、払出制御装置210は、シリアル送信回路615を備えてはいない点が、図6に示した遊技制御装置100と異なっている。その他の構成は、図11に示した遊技制御装置100と同じ構成である。

30

【0240】

また、演出制御装置150は、シリアル送信回路615を備えてはいない点、さらに、遊技用演算処理装置600にRAMアクセス規制回路640を備えていない点が、図6に示した遊技制御装置100と異なっている。その他の構成については、図11に示した遊技制御装置100と同じ構成である。

【0241】

なお、払出制御装置210及び演出制御装置150に備えたシリアル受信回路625は、払出制御装置210(又は演出制御装置150)のCPU102が起動した後であれば、遊技制御装置100のシリアル送信回路615からの信号を受け入れられる状態となっている。なお、払出制御装置210(又は演出制御装置150)のシリアル受信回路625とCPU102とは、データバス617によって相互に接続される構成となっている。

40

【0242】

図12は、本発明の第1の実施の形態のCPUコア102の内部構成を説明するブロック図である。CPUコア102はZ80系のCPUとして構成されている。

【0243】

図12に示すCPUコア102は、それぞれ8ビットの幅を有する、Wレジスタ1201A、Aレジスタ1202A、Bレジスタ1204A、Cレジスタ1205A、Dレジス

50

タ1207A、Eレジスタ1208A、Hレジスタ1210A、Lレジスタ1211Aを備えている。

【0244】

これらの汎用レジスタは、Wレジスタ1201AとAレジスタ1202Aとを組み合わせ、16ビットの幅を有するWAレジスタ1203Aとして使用することも可能である。同様に、Bレジスタ1204AとCレジスタ1205Aとを組み合わせたBCレジスタ1206A、Dレジスタ1207AとEレジスタ1208Aとを組み合わせたDEレジスタ1209A、Hレジスタ1210AとLレジスタ1211Aとを組み合わせたHLレジスタ1212Aを使用することも可能である。

【0245】

なお、これらの汎用レジスタは、1つの汎用レジスタ群（バンク0のレジスタ群）1220Aを形成している。一方、CPUコア102は、バンク0のレジスタ群1220Aに含まれる汎用レジスタと同様の構成を有する、もう1つの汎用レジスタ群（バンク1のレジスタ群）1220Bを備えている。

【0246】

このバンク1のレジスタ群1220Bには、バンク0のWレジスタ1201A～Lレジスタ1211Aと同一の機能を有する、Wレジスタ1201B～Lレジスタ1211Bを備えている。これらのレジスタも、バンク0同様に、WAレジスタ1203B～HLレジスタ1212Bとして、16ビットのレジスタとして使用することが可能である。

【0247】

さらに、CPUコア102は、8ビットの幅を有するフラグレジスタ1200を備えている。

【0248】

フラグレジスタ1200は、図13で詳細を説明するが、レジスタを用いた演算結果が格納される。また、フラグレジスタ1200の、レジスタバンクセレクタ（RBS）1301（図13で後述）によって、2つの汎用レジスタ群1220A、1220Bのうちのいずれを、演算対象として用いるかが選択される。

【0249】

レジスタバンクセレクタ（RBS）1301により選択されたレジスタ群に属する各レジスタは、後述の命令解釈実行回路1242によって演算に用いられる。一方、選択されていないレジスタ群に属する各レジスタは、レジスタバンクセレクタ（RBS）1301の値が変更されて選択対象となるまでは、値を保持する。

【0250】

また、CPUコア102は、8ビットの幅を有して、上位アドレス指定レジスタとして機能するKレジスタ1230を備えている。このレジスタは、後述の命令解釈実行回路1242が、所定の16ビットのアドレスに記憶されたデータにアクセスする際の上位8ビットのアドレスを指定する上位アドレス指定レジスタである。

【0251】

さらに、CPUコア102は、それぞれ16ビットの幅を有する、IXレジスタ1231、IYレジスタ1232、スタックポインタとして機能するSPレジスタ1233、及びプログラムカウンタとして機能するPCレジスタ1234を備えている。

【0252】

IXレジスタ1231、IYレジスタ1232は、後述の命令解釈実行回路1242がデータをアクセスする際のインデックスとして用いられる。スタックポインタ1233は、スタックエリア（図18で後述）にデータを格納する（又はデータを取り出す）際の領域の位置を示す。プログラムカウンタ1234は、後述の命令解釈実行回路1242で実行されている命令が格納されているアドレスを示している。

【0253】

命令解釈実行回路1242は、遊技制御プログラムを実行して、CPUコア102内部の各レジスタを用いた演算処理を行う。具体的には、ユーザプログラムROM103にて

10

20

30

40

50

、プログラムカウンタ 1 2 3 4 に示されるアドレスに記憶されたデータを読み出すとともに、読み出したデータをコードと見なして、コードに対応する命令を実行する。

【 0 2 5 4 】

故に、本実施形態においては、CPUコア 1 0 2 自体を演算処理手段として例示しているが、CPUコア 1 0 2 の内部では、命令解釈実行回路 1 2 4 2 が主体となって演算処理手段の機能を果たしている。

【 0 2 5 5 】

なお、命令解釈実行回路 1 2 4 2 は、遊技制御プログラムの命令に対応して、アクセス回路 1 2 4 3、アドレスバス 6 5 0、及びデータバス 6 6 0 を介して、CPUコア 1 0 2 外部のユーザプログラム ROM 1 0 3、ユーザワーク RAM 6 0 4、及び他の回路との間で、データの授受を行う場合もある。

10

【 0 2 5 6 】

また、命令解釈実行回路 1 2 4 2 は、ユーザプログラム ROM 1 0 3 の命令を 1 つずつ実行する毎に、次の命令が格納されているアドレスをプログラムカウンタ 1 2 3 4 に格納する。このようにして命令の実行と、プログラムカウンタ 1 2 3 4 の更新を繰り返すことで、遊技制御プログラムが順次実行される。なお、割込制御回路 6 1 0 A からの割込信号を受け付けると、プログラムカウンタ 1 2 3 4 の値は、予め設定された割込処理のアドレスの値に切り替えられる。

【 0 2 5 7 】

この命令解釈実行回路 1 2 4 2 及び CPU コア 1 0 2 に備える各レジスタは、内部バス 1 2 3 5 によって、データが授受される。

20

【 0 2 5 8 】

初期値設定回路 1 2 4 1 は、CPU コア 1 0 2 に備える各レジスタに初期値をハード的に設定する回路である。

【 0 2 5 9 】

内蔵リセット回路 1 2 4 0 は、セキュリティ回路 6 3 0 からのリセット信号を受信すると、初期値設定回路 1 2 4 1 を起動させ、CPU コア 1 0 2 に備える各レジスタに初期値を設定させたのちに、命令解釈実行回路 1 2 4 2 を起動させる。

【 0 2 6 0 】

図 1 3 は、本発明の第 1 の実施の形態のフラグレジスタ 1 2 0 0 の構成を説明する図である。フラグレジスタ 1 2 0 0 の各ビットの値は、図 1 2 における命令解釈実行回路 1 2 4 2 によって設定される。

30

【 0 2 6 1 】

割込マスタ許可フラグ ( I M F ) 1 3 0 0 は、割込信号による割込処理の許否を設定するフラグであり、セット ( 値が「 1 」 ) で許可、クリア ( 値が「 0 」 ) で禁止となる。

【 0 2 6 2 】

レジスタバンクセレクタ ( R B S ) 1 3 0 1 は、図 1 2 における命令解釈実行回路 1 2 4 2 が演算処理を行う際に、2 つの汎用レジスタ群 1 2 2 0 A、1 2 2 0 B のうちのいずれを用いるかを選択するフラグであり、セット ( 値が「 1 」 ) でバンク 1 のレジスタ群 1 2 2 0 B、クリア ( 値が「 0 」 ) でバンク 0 のレジスタ群 1 2 2 0 A が選択される。

40

【 0 2 6 3 】

オーバーフローフラグ ( V F ) 1 3 0 2 は、所定の演算によって演算対象の汎用レジスタに桁溢れ ( オーバーフロー ) が発生するとセット ( 値が「 1 」 ) され、他の場合はクリア ( 値が「 0 」 ) される。

【 0 2 6 4 】

サインフラグ ( S F ) 1 3 0 3 は、所定の演算によって演算対象の汎用レジスタの最上位ビットが「 1 」になるとセット ( 値が「 1 」 ) され、他の場合はクリア ( 値が「 0 」 ) される。

【 0 2 6 5 】

ハーフキャリーフラグ ( H F ) 1 3 0 4 は、8 ビット演算を行った結果として演算対象

50

の汎用レジスタの4ビット目に桁上がり（キャリー）や桁借り（ボロー）が発生するとセット（値が「1」）され、他の場合はクリア（値が「0」）される。

【0266】

キャリーフラグ（CF）1305は、所定の演算によって桁上がり（キャリー）や桁借り（ボロー）が発生するとセット（値が「1」）され、他の場合はクリア（値が「0」）される。

【0267】

ゼロフラグ（ZF）1306は、所定の演算結果が「0」となった場合にセット（値が「1」）され、他の場合はクリア（値が「0」）される。

【0268】

ジャンプステータスフラグ（JF）1307は、キャリーフラグ（CF）1305又はゼロフラグ（ZF）1306の少なくとも一方がセットされた場合にはセット（値が「1」）される。若しくは、演算処理を行わない場合であっても、汎用レジスタの値が「00H」の値になった場合は、セット（値が「1」）される。このような条件をいずれも満たさない場合は、クリア（値が「0」）される。

【0269】

なお、CPUコア102がリセットされた場合には、フラグレジスタの全bitが0に設定される。

【0270】

図14は、本発明の第1の実施の形態のユーザワークRAM104の一例を示す図である。

【0271】

ユーザワークRAM104は、第1停電復旧領域701、ワークエリア702、第2停電復旧領域703、チェックサム領域704、使用禁止領域（アクセス禁止領域）705、及びスタック領域706を有する。

【0272】

ユーザワークRAM104には、アドレス「2800H」～「29FFH」が割り当てられており、第1停電復旧領域701にはアドレス「2800H」が割り当てられ、ワークエリア702にはアドレス「2801H」～「2917H」が割り当てられ、第2停電復旧領域703にはアドレス「2918H」が割り当てられ、チェックサム領域704にはアドレス「2919H」が割り当てられ、使用禁止領域705にはアドレス「291AH」～「297FH」が割り当てられ、スタック領域706にはアドレス「2980H」～「29FFH」が割り当てられる。

【0273】

ユーザワークRAM104の各領域について説明する。

【0274】

第1停電復旧領域701及び第2停電復旧領域703は、遊技機1への電源供給開始時に参照される情報が格納されており、直前の電源供給停止のとき（停電発生や遊技機1の電源スイッチをオフにしたとき）に、電源遮断の処理が正しく実行されていたか否かを示す情報（電源遮断確認フラグ）が格納されている。

【0275】

ワークエリア702には、遊技制御で必要な変数等が格納され、図25及び図26に示す遊技制御装置メイン処理並びに図29に示すタイマ割込処理等で、これらの変数が更新される。具体的には、特図変動表示ゲーム及び普図変動表示ゲームの保留数を格納する保留カウンタ、特図変動表示ゲームを実行するために必要な各種乱数の生成領域、普図変動表示ゲームを抽選するために必要な乱数（普図乱数）の生成領域、普図乱数の保存領域、特図変動表示ゲームを実行するために必要な各種乱数の保存領域などがワークエリア702に記憶される。これらの詳細については、図15から図17を参照して後述する。

【0276】

チェックサム領域704には、停電発生時に算出されたユーザワークRAM104の第

10

20

30

40

50

1 停電復旧領域 701、ワークエリア 702、及び第 2 停電復旧領域 703 のチェックサムが格納される。

【0277】

使用禁止領域 705 は使用されない記憶領域であり、当該領域へのアクセスがあると、セキュリティ回路 630 (図 6) によって、CPU コア 102 がリセットされるようになっている。

【0278】

スタック領域 706 には、CPU コア 102 で演算されているデータの一部を一時的に退避させる場合に、退避データが格納される。また、割込みが発生した場合の戻りアドレスや、サブルーチンや関数を呼び出す場合の戻りアドレスも格納される。

10

【0279】

なお、本実施形態では、ワークエリア 702 のうち遊技制御に用いられる各種乱数の生成及び保存領域の上位アドレスと、スタック領域 706 が割り当てられている領域の上位アドレスが異なるように設定されている。具体的には、ワークエリア 702 の各種乱数の生成及び保存領域の上位アドレスが「28H」(第 1 記憶領域)、スタック領域の上位アドレスが「29H」(第 2 記憶領域)となっており、乱数更新時にスタック領域の値を誤って更新してしまったり、スタック領域更新時に乱数の値を誤って更新してしまったりすることを防止している。

【0280】

さらに、上位アドレスが誤って設定されてしまった場合であっても誤作動を生じさせないように、各種乱数の生成及び保存領域の下位アドレスがスタック領域の下位アドレスと重ならないように設定している。

20

【0281】

具体的には、図 15 から図 17 で後述する、特図変動表示ゲームを実行するための各種乱数(始動記憶)の生成領域(2820H~2827H)、普図乱数の生成領域(2828H)及び普図乱数の保存領域(282CH~282FH)、第 1 特図変動表示ゲームを実行するための各種乱数(始動記憶)の保存領域(2830H~284FH)、及び、第 2 特図変動表示ゲームを実行するための各種乱数(始動記憶)の保存領域(2850H~286FH)などが該当する。これらの領域の上位アドレスを「28H」から「29H」に置換した領域(即ち 2920H~296FH の領域)は、図 14 に示すように、使用禁止領域 705 として設定されている。

30

【0282】

図 15 から図 17 は、本発明の第 1 の実施の形態のユーザワーク RAM 104 に割り当てられたワークエリア 702 に格納されるデータの具体例を説明する図である。図 15 は保留カウンタ、各種乱数の生成領域及び普図乱数の保存領域を示す。図 16 は第 1 特図変動表示ゲームを実行するために必要な各種乱数の保存領域、図 17 は第 2 特図変動表示ゲームを実行するために必要な各種乱数の保存領域を示す。

【0283】

図 15 を参照すると、保留カウンタを格納する領域として 281DH から 281FH が割り当てられている。具体的には、第 1 特図変動表示ゲームの保留記憶数を格納する第 1 特図保留カウンタ(281DH)、第 2 特図変動表示ゲームの保留記憶数を格納する第 1 特図保留カウンタ(281EH)、普図変動表示ゲームの保留記憶数を格納する普図保留カウンタ(281FH)となっている。さらに、特図変動表示ゲームを実行するための各種乱数(始動記憶)の生成領域(2820H~2827H)、普図乱数の生成領域(2828H)及び普図乱数の保存領域(282CH~282FH)が割り当てられている。

40

【0284】

特図変動表示ゲームを実行するための各種乱数には、特図変動表示ゲームの大当りを決定するための大当り乱数(2バイト)、特図変動表示ゲームの大当り図柄を決定するための大当り図柄乱数、特図変動表示ゲームの変動パターンを決定するための第 1 変動 P (パターン)乱数(2バイト)、第 2 変動 P (パターン)乱数、第 3 変動 P (パターン)乱数

50



(2バイト)が含まれる。

【0285】

図16を参照すると、第1特図変動表示ゲームを実行するための各種乱数を格納する領域が割り当てられている。本実施形態では、第1特図変動表示ゲームの始動記憶(特図1始動記憶)を最大4回分保持可能となっている。そして、これら乱数値は、特図始動口SW共通処理(図40)において設定される。

【0286】

具体的には、記憶された特図1始動記憶の保留順に領域が割り当てられる。具体的には、保留1は2830Hから2837H、保留2は2838Hから283FH、保留3は2840Hから2847H及び保留4は2848Hから284FHの各領域が割り当てられている。また、各領域には、前述した特図変動表示ゲームを実行するための各種乱数に応じた領域がさらに割り当てられている。

10

【0287】

そして、保留1個目の特図1始動記憶に対応する第1特図変動表示ゲームが実行される場合には、後述する特図普段処理の特図1変動開始処理(図45)において、保留1個目の各乱数を取得(一時的に退避)した後に、保留2個目~4個目の特図1始動記憶に対応する領域に記憶された各乱数を保留1個目~3個目の特図1始動記憶に対応する領域にシフトする。

【0288】

そして、遊技制御装置100は、後述する特図大当たり判定処理(図42)において、取得した各乱数(一時的に退避していた保留1個目の各乱数)の値と、予め定義された特図1変動表示ゲーム用の大当たり判定値とを比較して、特図1変動表示ゲームが大当たりか否かを判定し、特図1始動記憶数(第1特図保留カウンタに記憶された値)を1減算する。

20

【0289】

図17を参照すると、図16に示した第1特図変動表示ゲームの場合と同様に、第2特図変動表示ゲームを実行するための各種乱数(特図2始動記憶)を格納する領域が割り当てられている。特図2始動記憶に含まれる各種乱数は、特図1始動記憶と同様に処理される。

【0290】

図18及び図19は、本発明の第1の実施の形態のスタック領域706の一例を示す図である。図18及び図19では、スタック領域706に戻りアドレスや所定のレジスタの値が格納される場合について説明する。

30

【0291】

まず、図18(A)は、遊技機1に電源が投入された直後の状態であり、スタック領域706に何もデータが格納されていない状態を示している。この場合、スタックポインタ(SP)1233は、スタック領域の最終領域(29FFH)をスタックポインタ初期値として示している。

【0292】

図18(B)は、スタック領域706に退避データが格納されたり、割り込み発生やサブルーチン呼び出しによって、スタック領域706に戻りアドレスなどのデータが格納された状態を示している。この場合、最後にデータ(又はアドレス)が格納された領域の1つ上のアドレスを、スタックポインタ1233によって示すことになる。

40

【0293】

なお、スタック領域706に戻りアドレスが格納される場合としては、割り込みが発生した場合と、後述する「CALL」命令を実行して通常サブルーチン呼び出しを行った場合とがあるが、本実施形態では、前者の場合と後者の場合とで、スタック領域706に格納されるデータが異なる。

【0294】

図19(C)は、呼出元のルーチンにて後述する「CALL」命令を実行して通常サブルーチン呼び出しを行った場合において、スタック領域706にデータが退避する様子

50

を示している。なお、この図は、図 18 (B) の状態にて「CALL」命令を実行した直後の様子を示している。

【0295】

図 18 (B) において、「CALL」命令を実行すると、スタック領域 706 に、呼出元のルーチンの戻りアドレスが上位バイトと下位バイトに分離して格納される。このとき、最後にアドレスが格納された領域の 1 つ上のアドレスが、スタックポインタ 1233 によって示される。

【0296】

なお、呼出先のサブルーチンの処理の最後にて、後述する「RET」命令が実行されると、この戻りアドレスがプログラムカウンタ 1234 に戻されるとともに、スタックポインタ 1233 の値も、「CALL」命令実行前の値に戻される (図 18 (B) の状態に戻る)。これにより、呼出元のルーチンに処理を戻すことができる。

10

【0297】

一方、図 19 (D) は、呼出元のルーチンにて割込みが発生して、呼出先の割込処理のルーチンが実行される場合において、スタック領域 706 にデータが退避する様子を示している。なお、図 19 (D) は、図 18 (B) の状態にて割込みが発生した直後の様子を示している。

【0298】

図 18 (B) において、割込みが発生すると、スタック領域 706 に、まず、呼出元のルーチンで設定されていたフラグレジスタ 1200 の値が格納され、次に、呼出元のルーチン戻りアドレスが上位バイトと下位バイトに分離して格納される。このとき、最後にアドレスが格納された領域の 1 つ上のアドレスが、スタックポインタ 1233 によって示される。

20

【0299】

なお、呼出先の割込処理ルーチンの最後にて、後述する「RETI」若しくは「RET N」命令が実行されると、スタック領域 706 に格納していた戻りアドレスがプログラムカウンタ 1234 に戻され、スタック領域 706 に格納していたフラグレジスタの値も、フラグレジスタ 1200 に戻される。さらに、スタックポインタ 1233 の値も、割込みが発生する前の値に戻される (図 18 (B) の状態に戻る)。これにより、呼出元のルーチンに処理を戻すことができる。

30

【0300】

このようにして、スタック領域 706 に格納された戻りアドレス等のデータは、後に格納された戻りアドレスから先に読み出される。

【0301】

なお、後述する「PUSH」命令を実行すると、命令で指定されたレジスタの値がスタック領域 706 に格納され、スタックポインタ 1233 の値も、図 19 (C) や図 19 (D) と同様に、最後にデータが格納された領域の 1 つ上のアドレスを示すことになる。

【0302】

一方、後述する「POP」命令を実行すると、スタック領域 706 に格納されていた値を取り出して、命令で指定されたレジスタに格納され、スタックポインタ 1233 の値も、図 19 (C) から図 18 (B) に変化するように、最後にデータが取り出された領域のアドレスを示すことになる。

40

【0303】

このように、本実施形態では、スタック領域 706 において最後にデータが格納されたアドレスの 1 つ上のアドレス (言い換えれば、最後に取り出されたデータのアドレス) を、スタックポインタ 1233 によって示している。すなわち、現時点でのスタックポインタ 1233 は、次のスタック領域 706 に格納されるデータのアドレスを示している。

【0304】

なお、別の方法として、現時点でのスタックポインタ 1233 によって、次にスタック領域 706 から取り出されるデータのアドレスを示すような方法も考えられる。このよう

50

な変形例を図20に示す。

【0305】

図20(E)は、遊技機1に電源が投入された直後の状態であり、スタック領域706に何もデータが格納されていない状態を示している。この場合、スタックポインタ(SP)1233は、スタック領域の最終領域(29FFH)の1つ下のアドレス(2A00H)をスタックポインタ初期値として示している。なお、このスタックポインタ初期値が示す領域は、スタック領域には含まれない領域(本実施形態では、ユーザワークRAM104の記憶領域にも含まれていない領域)である。

【0306】

図20(F)は、スタック領域706に退避データが格納されたり、割り込み発生やサブルーチン呼び出しによって、スタック領域706に戻りアドレスなどのデータが格納された状態を示している。この場合、最後にデータ(又はアドレス)が格納された領域を、スタックポインタ1233によって示すことになる。スタック領域706からデータを取り出す場合は、現時点でスタックポインタ1233が示しているアドレスからデータを取り出せばよい。

【0307】

なお、スタック領域706の割り当てが、上位アドレスが同じ領域に限定されている(例えば、2900H~29FFHの領域に割り当てられることが限定されている)のであれば、スタックポインタ1233は下位のアドレスのみを指定するだけで機能を発揮する。このような構成においては、スタックポインタ1233を1バイトのレジスタで構成することが可能であり、図20の方法よりも、図18及び図19の方法にてスタックポインタ1233によるアドレス指定を行う構成が効果的である。

【0308】

図21及び図22は、本発明の第1の実施の形態のCPUコア102によって実行される遊技制御プログラムを記述するためのアセンブリ言語の命令を説明する図である。各命令は、対応するコードデータに変換されて、ユーザプログラムROM103の所定のアドレスに記憶されている。図21は、変換後のコードデータにアドレス部を含まない命令(変換後のコードデータが命令コード部のみで構成される命令)を示し、図22は、変換後のコードデータに命令コード部とアドレス部の各々を含む命令を示している。なお、ここで対象となるレジスタは、図12に示すレジスタである。

【0309】

図21において、命令2100は、指定するレジスタの値を、Aレジスタ1202Aに格納する命令である。例えば、「LD A, B」は、Bレジスタ1204Aの値を、Aレジスタ1202Aに格納する命令である。他の「LD A, C」~「LD A, L」の命令も同様に、対応するCレジスタ1205A~Lレジスタ1211Aから値を抽出して、Aレジスタ1202Aに格納する命令である。

【0310】

命令2101は、Aレジスタ1202Aの値を、指定するレジスタに格納する命令である。例えば、「LD B, A」は、Aレジスタ1202Aの値を、Bレジスタ1204Aの値に格納する命令である。他の「LD C, A」~「LD L, A」の命令も同様に、Aレジスタ1202Aから値を抽出して、対応するCレジスタ1205A~Lレジスタ1211Aに格納する命令である。

【0311】

命令2102は、Aレジスタ1202Aの値と指定するレジスタの値とで論理和演算を行い、演算結果をAレジスタ1202Aに格納する命令である。例えば、「OR B」は、Aレジスタ1202Aの値とBレジスタ1204Aの値とで、各ビット毎に論理和演算を行い、演算結果をAレジスタ1202Aに格納する命令である。他の「OR C」~「OR L」の命令も同様に、Aレジスタ1202Aの値と、対応するCレジスタ1205A~Lレジスタ1211Aの値とで論理和演算を行い、演算結果をAレジスタ1202Aに格納する命令である。

10

20

30

40

50

## 【0312】

命令2103の「ADD A, A」は、Aレジスタ1202Aの値に、同じAレジスタ1202Aの値を加算して、演算結果をAレジスタ1202Aに格納する命令である。実質的には、Aレジスタ1202Aの値が2倍される演算が行われる命令である。命令2104の「ADD HL, BC」は、HLレジスタ1212Aの値にBCレジスタ1206Aの値を加算して、演算結果をHLレジスタ1212Aに格納する命令である。

## 【0313】

命令2105の「PUSH HL」は、HLレジスタ1212Aの値を、スタックエリアに格納する命令である。命令2106の「POP HL」は、スタックエリアから取り出した値をHLレジスタ1212Aの値に格納する命令である。

10

## 【0314】

命令2107の「INC HL」は、HLレジスタ1212Aの値を「1」だけ加算する命令である。命令2108の「DEC HL」は、HLレジスタ1212Aの値を「1」だけ減算する命令である。

## 【0315】

命令2109の「JP (HL)」は、HLレジスタ1212Aの値をアドレスと見なして、そのアドレス先に分岐する命令である。例えば、HLレジスタ1212Aの値が「1234H」であれば、命令解釈実行回路1242は、プログラムカウンタ1234の値を「1234H」に変更する。これにより、次に、命令解釈実行回路1242が実行する命令は、アドレスが「1234H」となる領域に格納されているコードを変換した命令となる。以降、命令解釈実行回路1242は、変更後のアドレス以降の命令を順次実行する。

20

## 【0316】

命令2110は、スタックエリアに格納された値をプログラムカウンタ1234に戻すことで、呼び出し先の処理ルーチンから、呼び出し元の処理ルーチンに復帰する命令である。ここで、「RET」は、図22で後述する「CALL nn」命令の実行により呼び出された処理ルーチンから復帰する命令であり、「RETI」は、INT割込により呼び出された処理ルーチンから復帰する命令であり、「RETN」は、NMI割込により呼び出された処理ルーチンから復帰する命令である。

## 【0317】

命令2111は、割込を受け付けるか否かを設定する命令である。「DI」は割り込みの禁止を設定する命令であり、「EI」は割り込みの許可を設定する命令である。

30

## 【0318】

命令2112は、Kレジスタ1230に格納されている値を上位バイトとし、Lレジスタ1211Aに格納されている値を下位バイトとして合成した2バイトの値をアドレスと見なして、そのアドレスの領域に記憶されている値を、Aレジスタ1202Aに設定する命令である。具体的には、「2803H」のアドレスに記憶された値をAレジスタ1202Aに設定したいときは、予め、Kレジスタ1230に「28H」の値を格納するとともに、Lレジスタ1211Aに「03H」の値を格納し、その状態で「LDK A, (L)」という命令を実行する。

40

## 【0319】

なお、Aレジスタ1202Aの値を、所望のアドレスの領域に格納する場合は、予め格納先の上位アドレスをKレジスタ1230に設定し、下位アドレスをLレジスタ1211Aに設定した状態で、図に示す「LDK (L), A」という形式の命令を実行する。

## 【0320】

命令2113の「INC L」は、Lレジスタ1211Aに格納されている値を「1」だけ加算する命令である。ただし、Lレジスタ1211Aの値が「FFH」のときに、「INC L」命令を実行すると、命令実行後のLレジスタ1211Aの値が「00H」になる。

## 【0321】

50

命令 2114 の「DEC L」は、レジスタ 1211A に格納されている値を「1」だけ減算する命令である。ただし、レジスタ 1211A の値が「00H」のときに、「DEC L」命令を実行すると、命令実行後のレジスタ 1211A の値が「FFH」になる。

【0322】

命令 2115 は、図 13 で前述したフラグレジスタ 1200 の「レジスタバンクセクタ」のビットに値を設定するための命令である。「LD RBS, 0」は該当ビットに「0」の値を設定する命令であり、「LD RBS, 1」は該当ビットに「1」の値を設定する命令である。

【0323】

命令 2106 の「PUSH FLG」は、フラグレジスタ 1200 の値を、スタックエリアに格納する命令である。命令 2107 の「POP FLG」は、スタックエリアから取り出した値をフラグレジスタ 1200 に格納する命令である。

【0324】

図 21 に示す命令のうち、図に示すように、命令 2100 から命令 2114 までの命令は、ユーザプログラム ROM 103 に記憶される際に、1 バイトの命令コード部を構成するコードデータに変換される。例えば、「LD A, B」の命令は、「78H」という 1 バイトのコードデータに変換されて、ユーザプログラム ROM 103 の所定アドレスに記憶されている。

【0325】

一方、図 21 に示す命令のうち、命令 2115 から命令 2117 までの命令は、ユーザプログラム ROM 103 に記憶される際に、2 バイトの命令コード部を構成するコードデータに変換される。例えば、「LD RBS, 0」は、「CBH」「00H」という 2 バイトのコードデータに変換され、ユーザプログラム ROM 103 の連続する所定のアドレスに記憶されている。

【0326】

次に、図 22 において、命令 2200 は、指定するレジスタに任意の 1 バイトの値を設定する命令である。図中の「n」はレジスタに設定する値を示しており、例えば、A レジスタ 1202A に「42H」の値を設定したいときは、「LD A, 42H」という命令になる。

【0327】

なお、この命令 2200 は、ユーザプログラム ROM 103 に記憶される際に、1 バイトの命令コード部と、1 バイトのアドレス部とからなる計 2 バイトのコードデータに変換される。例えば、「LD A, 42H」の命令は、1 バイトの命令コード部のデータ「3EH」と、1 バイトのアドレス部のデータ「42H」とに変換され、ユーザプログラム ROM 103 の連続する所定のアドレスに「3EH」、「42H」の順で記憶されている。

【0328】

同様に、「LD B, n」は、B レジスタ 1204A に任意の 1 バイトの値 (n) を設定する命令であり、「LD K, n」は、K レジスタ 1230 に任意の 1 バイトの値 (n) を設定する命令である。これらの命令も、1 バイトの命令コード部と、1 バイトのアドレス部のデータに変換されて、ユーザプログラム ROM 103 の連続する所定のアドレスに記憶されている。

【0329】

命令 2201 は、任意のアドレスの領域に記憶されている値を、A レジスタ 1202A に設定する命令である。図中の「nn」は 2 バイトのアドレス値を示しており、例えば、「2901H」のアドレスに記憶された値を A レジスタ 1202A に設定したいときは、「LD A, (2901H)」という命令になる。

【0330】

なお、A レジスタ 1202A の値を、任意のアドレスの領域に格納する場合は、図に示すように「LD (nn), A」という形式の命令になる。例えば、A レジスタ 1202

10

20

30

40

50

A の値を「2901H」のアドレスの領域に格納したいときは、「LD (2901H), A」という命令になる。

【0331】

この命令2201は、ユーザプログラムROM103に記憶される際に、1バイトの命令コード部と、2バイトのアドレス部とからなる計3バイトのコードデータに変換される。例えば、「LD A, (2901H)」の命令は、1バイトの命令コード部のデータ「3AH」と、2バイトのアドレス部のデータ「01H」「29H」とに変換され、ユーザプログラムROM103の連続する所定のアドレスに、「3AH」、「01H」、「29H」の順で記憶されている。

【0332】

命令2202は、Kレジスタ1230に格納されている値を上位バイトとし、任意の1バイトの値で示される値を下位バイトとして合成した2バイトの値をアドレスと見なして、そのアドレスの領域に記憶されている値を、Aレジスタ1202Aに設定する命令である。この場合の任意の1バイトの下位アドレスは、図中の「n」に対応している。

【0333】

具体的には、「2803H」のアドレスに記憶された値をAレジスタ1202Aに設定したいときは、予めKレジスタ1230に「28H」の値を格納しておき、その状態で「LDK A, (03H)」という命令を実行する。

【0334】

なお、Aレジスタ1202Aの値を、所望のアドレスの領域に格納する場合は、予め格納先の上位アドレスをKレジスタ1230に設定した状態で、図に示す「LDK (n), A」という形式の命令を実行する。例えば、Aレジスタ1202Aの値を「2803H」のアドレスの領域に格納したいときは、Kレジスタ1230の値を「28H」に設定した状態で、「LDK (03H), A」という命令を実行する。

【0335】

この命令2202は、ユーザプログラムROM103に記憶される際に、1バイトの命令コード部と、1バイトのアドレス部とからなる計2バイトのコードデータに変換される。例えば、「LDK A, (03H)」の命令は、1バイトの命令コード部のデータ「3FH」と、1バイトのアドレス部のデータ「03H」とに変換され、ユーザプログラムROM103の連続する所定のアドレスに、「3FH」、「03H」の順で記憶されている。

【0336】

命令2203は、指定するレジスタに任意の2バイトの値を設定する命令である。図中の「nn」はレジスタに設定する2バイトの値を示しており、例えば、HLレジスタ1212Aに「5678H」の値を設定したいときは、「LD HL, 5678H」という命令になる。

【0337】

なお、この命令2203は、ユーザプログラムROM103に記憶される際に、1バイトの命令コード部と、2バイトのアドレス部とからなる計2バイトのコードデータに変換される。例えば、「LD HL, 5678H」の命令は、1バイトの命令コード部のデータ「21H」と、2バイトのアドレス部のデータ「78H」「56H」とに変換され、ユーザプログラムROM103の連続する所定のアドレスに「21H」、「78H」、「56H」の順で記憶されている。

【0338】

同様に、「LD SP, nn」は、スタックポインタ1233に任意の2バイトの値(n)を設定する命令である。これらの命令も、1バイトの命令コード部と、2バイトのアドレス部のデータに変換されて、ユーザプログラムROM103の連続する所定のアドレスに記憶されている。

【0339】

命令2204は、任意の連続するアドレスの領域に記憶されている2バイトの値を、H

10

20

30

40

50

Lレジスタ1212Aに設定する命令である。図中の「nn」は連続する小さいほうのアドレス値を示しており、例えば、「2901H」と「2902H」の各アドレスに記憶された値をHLレジスタ1212Aに設定したいときは、「LD HL, (2901H)」という命令になる。このとき、「2901H」のアドレスに格納されている値がLレジスタ1211Aに格納され、「2902H」のアドレスに格納されている値がHレジスタ1210Aに格納される。

**【0340】**

なお、HLレジスタ1212Aの値を、任意の連続するアドレスの領域に格納する場合は、図に示すように「LD (nn), HL」という形式の命令になる。例えば、HLレジスタ1212Aの値を、「2901H」と「2902H」の各アドレスの領域に格納したいときは、「LD (2901H), HL」という命令になる。このとき、Lレジスタ1211Aの値が「2901H」のアドレスに格納され、Hレジスタ1210Aの値が「2902H」のアドレスに格納される。

10

**【0341】**

この命令2204は、ユーザプログラムROM103に記憶される際に、1バイトの命令コード部と、2バイトのアドレス部とからなる計3バイトのコードデータに変換される。例えば、「LD HL, (2901H)」の命令は、1バイトの命令コード部のデータ「2AH」と、2バイトのアドレス部のデータ「01H」「29H」とに変換され、ユーザプログラムROM103の連続する所定のアドレスに、「2AH」、「01H」、「29H」の順で記憶されている。

20

**【0342】**

命令2205は、Kレジスタ1230に格納されている値を上位バイトとし、任意の1バイトの値で示される値を下位バイトとして合成した2バイトの値をアドレスと見なして、そのアドレスの領域に記憶されている値と、当該アドレスに続くアドレスに記憶されている値の各々を、HLレジスタ1212Aに設定する命令である。この場合の任意の1バイトの下位アドレスは、図中の「n」に対応している。

**【0343】**

具体的には、「2803H」及び「2804H」のアドレスに記憶された値をHLレジスタ1212Aに設定したいときは、予めKレジスタ1230に「28H」の値を格納しておき、その状態で「LDK HL, (03H)」という命令を実行する。これにより、「2803H」のアドレスに格納されている値がLレジスタ1211Aに格納され、「2804H」のアドレスに格納されている値がHレジスタ1210Aに格納される。

30

**【0344】**

なお、HLレジスタ1212Aの値を、所望のアドレスの領域に格納する場合は、予め格納先の上位アドレスをKレジスタ1230に設定した状態で、図に示す「LDK (n), HL」という形式の命令を実行する。例えば、HLレジスタ1212Aの値を「2803H」及び「2804H」のアドレスの領域に格納したいときは、Kレジスタ1230の値を「28H」に設定した状態で、「LDK (03H), HL」という命令を実行する。

**【0345】**

この命令2205は、ユーザプログラムROM103に記憶される際に、1バイトの命令コード部と、1バイトのアドレス部とからなる計2バイトのコードデータに変換される。例えば、「LDK HL, (03H)」の命令は、1バイトの命令コード部のデータ「2FH」と、1バイトのアドレス部のデータ「03H」とに変換され、ユーザプログラムROM103の連続する所定のアドレスに、「3FH」、「03H」の順で記憶されている。

40

**【0346】**

命令2206は、指定するアドレスに処理を分岐させる命令である。図中の「nn」は分岐先のアドレスを示しており、例えば、「5678H」のアドレスに処理に分岐させるときは、「JP 5678H」という命令になる。この命令を実行すると、命令解釈実行

50

回路 1 2 4 2 は、指定するアドレスの値をプログラムカウンタ 1 2 3 4 に設定する。

【 0 3 4 7 】

なお、この「JP nn」の形式で示される命令は、フラグレジスタ 1 2 0 0 の内容にかかわらず、指定するアドレスに処理を分岐する。フラグレジスタ 1 2 0 0 のゼロフラグ 1 3 0 6 ( 図 1 3 ) がセットされているときのみ分岐を行いたい場合は「JP Z, nn」の命令を用い、このゼロフラグ 1 3 0 6 がクリアされているときのみ分岐を行いたい場合は「JP NZ, nn」の命令を用いる。

【 0 3 4 8 】

なお、この命令 2 2 0 6 は、ユーザプログラム ROM 1 0 3 に記憶される際に、1 バイトの命令コード部と、2 バイトのアドレス部とからなる計 2 バイトのコードデータに変換される。例えば、「JP 5 6 7 8 H」の命令は、1 バイトの命令コード部のデータ「C 3 H」と、2 バイトのアドレス部のデータ「7 8 H」「5 6 H」とに変換され、ユーザプログラム ROM 1 0 3 の連続する所定のアドレスに「C 3 H」、「7 8 H」、「5 6 H」の順で記憶されている。

10

【 0 3 4 9 】

命令 2 2 0 7 の「CALL nn」は、指定するアドレスの処理を呼び出して、呼び出し先の処理ルーチンに一時的に分岐させる命令である。図中の「nn」は呼び出し先のアドレスを示しており、例えば、「5 6 7 8 H」のアドレスの処理を呼び出すときは、「CALL 5 6 7 8 H」という命令になる。この命令を実行すると、命令解釈実行回路 1 2 4 2 は、現在実行中の命令の次のステップに位置するアドレスの値を、戻り先のアドレスとしてスタックエリアに格納した上で、分岐先のアドレスの値をプログラムカウンタ 1 2 3 4 に設定する。

20

【 0 3 5 0 】

なお、呼び出し先の処理ルーチンの最終ステップでは、前述の「RET」命令を実行することで、プログラムカウンタ 1 2 3 4 には、スタックエリアに格納された戻り先のアドレスの値を復帰させることができる。これにより、呼び出し元の処理ルーチンのアドレスに処理を戻すことができる。

【 0 3 5 1 】

なお、この命令 2 2 0 7 は、ユーザプログラム ROM 1 0 3 に記憶される際に、1 バイトの命令コード部と、2 バイトのアドレス部とからなる計 2 バイトのコードデータに変換される。例えば、「CALL 5 6 7 8 H」の命令は、1 バイトの命令コード部のデータ「CDH」と、2 バイトのアドレス部のデータ「7 8 H」「5 6 H」とに変換され、ユーザプログラム ROM 1 0 3 の連続する所定のアドレスに「C 3 H」、「7 8 H」、「5 6 H」の順で記憶されている。

30

【 0 3 5 2 】

図 2 3 は、本発明の第 1 の実施の形態の各装置 ( 遊技制御装置 1 0 0 、 払出制御装置 2 1 0 、 及び演出制御装置 1 5 0 ) の電源投入時処理のフローチャートである。

【 0 3 5 3 】

具体的には、図 2 3 ( A ) は、遊技制御装置 1 0 0 の電源投入時処理のフローチャートであり、図 2 3 ( B ) は、払出制御装置 2 1 0 の電源投入時処理のフローチャートであり、図 2 3 ( C ) は、演出制御装置 1 5 0 の電源投入時処理のフローチャートである。

40

【 0 3 5 4 】

最初に、遊技制御装置 1 0 0 の電源投入時処理 ( 図 2 3 ( A ) ) から説明する。この電源投入時処理は、ユーザプログラム ROM 6 0 2 ( 図 5 参照 ) に記憶された遊技制御プログラムによって実行される処理ではなく、遊技制御装置 1 0 0 に備わる各種ハードウェア ( CPU コア 1 0 2 周辺のハードウェア ) によって実行される処理である。

【 0 3 5 5 】

まず、遊技制御装置 1 0 0 は、電源装置 1 6 0 から出力されたりセット信号が伝達される ( 2 3 0 1 ) 。

【 0 3 5 6 】

50



このリセット信号は、電源装置 160 から、セキュリティ回路 630 (図 11 参照)、RAM アクセス規制回路 640 のフリップフロップ回路 641 のリセット端子 (図 11 参照)、及びシリアル送信回路 615 のリセット端子に入力される。具体的には、これらのリセット端子には、電源が投入されると、所定時間、所定の電圧 (例えば、5V) 以下の電圧が印加されることによってリセット信号が入力され、所定時間経過後に所定の電圧が印加されることによって、リセット信号が入力されなくなる。

【0357】

なお、セキュリティ回路 630 は、電源装置 160 からリセット信号が入力されると、後述のセキュリティチェック処理が終了するまで CPU コア 102 のリセット端子にリセット信号を出力し続けて、CPU コア 102 の起動を待機させる。

10

【0358】

そして、シリアル送信回路 615 のリセット端子にリセット信号が入力されると、シリアル送信回路 615 の入力端子及び出力端子の電圧レベルがローに制御され、各種装置 (普電 SOL90、大入賞口 SOL38 等) に接続される出力 I/F 106 のポートをすべて 0 に設定することにより、シリアル送信回路 615、及び出力 I/F 106 がハードウェアにより初期化される (2302)。

【0359】

次に、RAM アクセス規制回路 640 によって、ユーザワーク RAM 104 への書き込み規制される RAM 書込禁止状態が発生する (2303)。

【0360】

20

具体的には、図 11 で説明したように、フリップフロップ回路 641 のクリア端子にはリセット信号が入力されるため、フリップフロップ回路 641 の出力端子 Q (負論理) からハイレベルの信号が出力される状態となる。これにより、OR ゲート回路 642 の他方の入力端子にハイレベルの信号が入力されても、ローレベルの信号が入力されても、ユーザワーク RAM 104 の書込指令入力端子にはハイレベルの信号が入力されることになるため、RAM 書込禁止状態が発生する。このように、電源投入時に RAM 書込禁止状態が発生させることによって、誤ったタイミングでユーザワーク RAM 104 が更新されることを防ぐことが可能となる。

【0361】

次に、リセット信号が入力された図 11 に示すセキュリティ回路 630 が自己診断処理を実行する (2304)。自己診断処理は、セキュリティ回路 630 が初期化されているか否かを判定する処理である。

30

【0362】

そして、自己診断処理によって、セキュリティ回路 630 が初期化されていると判定された場合には、セキュリティ回路 630 は、セキュリティチェック処理を実行する (2305)。セキュリティチェック処理は、図 11 で説明したように、HW パラメータ ROM 603 (図 5 参照) に記憶された正当性確認情報を用いて、ユーザプログラム ROM 602 (図 5 参照) に記憶されたプログラムの正当性の判定を行う処理である。

【0363】

ステップ 2305 の処理で、セキュリティチェック処理を実行すると、遊技制御装置プログラム開始準備処理へ移行する。このとき、セキュリティ回路 630 は、CPU コア 102 のリセット端子に出力していたリセット信号を停止することで、CPU コア 102 が起動する。このため、遊技制御装置プログラム開始準備処理は、CPU コア 102 によって実行される。遊技制御装置プログラム開始準備処理は図 24 にて後述する。

40

【0364】

次に、払出制御装置 210 の電源投入時処理 (図 23 (B)) を説明する。前述したように、払出制御装置 210 は、シリアル送信回路 615 を備えていない点を除き、図 11 に示した遊技制御装置 100 と同じ構成である。図 11 に示す遊技制御装置 100 の構成部と同じ構成部については、同じ符号を付与して説明する。

【0365】

50

まず、払出制御装置 210 は、電源装置 160 から出力されたりセット信号が伝達される (2311)。なお、ステップ 2311 の処理は、ステップ 2301 の処理と同じである。

【0366】

そして、払出制御装置 210 にリセット信号が入力されると、払出制御装置 210 の出力ポート (図 4 の入出力 I/F 216 に含まれる) の電圧レベルが 0 に設定され、各種装置 (払出モータ 220、及び発射制御装置 221 等) に接続される入出力 I/F 216 のポートがすべて 0 に設定され、入出力 I/F 216 がハードウェアにより初期化される (2312)。

【0367】

次に、払出制御装置 210 の RAM アクセス規制回路 640 によって、RAM 214 への書き込み規制される RAM 書込禁止状態が発生する (2313)。なお、ステップ 2313 の処理の具体的な説明は、ステップ 2303 の処理と同じである。

【0368】

次に、リセット信号が入力された払出制御装置 210 のセキュリティ回路 630 が自己診断処理を実行する (2314)。なお、ステップ 2314 の処理の具体的な説明は、ステップ 2304 の処理と同じである。

【0369】

そして、自己診断処理によって、セキュリティ回路 630 が初期化されていると判定された場合には、セキュリティ回路 630 は、セキュリティチェック処理を実行する (2315)。なお、ステップ 2315 の処理の具体的な説明は、ステップ 2305 の処理と同じである。

【0370】

そして、払出制御装置 210 は、電源投入時の初期化処理を実行する (2316)。電源投入時の初期化処理は、RAM 214 等を初期化する処理であって、CPU 212 によって実行される。また、RAM 214 を初期化する前に、ステップ 2313 の処理で発生した RAM 書込禁止状態が解除されて、RAM 214 は RAM 書込可能状態となる。

【0371】

次に、払出制御装置 210 は、遊技制御装置 100 からの指令を受信可能な状態を発生させる (2317)。そして、払出制御装置 210 の CPU 212 は、遊技制御装置 100 から送信された指令が初期化指令であるか否かを判定する (2319)。ステップ 2319 の処理で、遊技制御装置 100 から送信された指令が初期化指令でないと判定された場合には (2319 の結果が「N」)、初期化指令が取り込まれるまで待機する。

【0372】

一方、遊技制御装置 100 から送信された指令が初期化指令であると判定された場合には (2319 の結果が「N」)、払出制御装置 210 は通信開始時の初期化処理を実行し (2320)、払出制御装置メイン処理へ移行する。

【0373】

次に、演出制御装置 150 の電源投入時処理 (図 23 (C)) を説明する。前述したように、演出制御装置 150 は、シリアル送信回路 615 を備えていない点、及び、遊技用演算処理装置 600 が RAM アクセス規制回路 640 を備えていない点以外は、図 11 に示した遊技制御装置 100 と同じ構成である。図 11 に示す遊技制御装置 100 の構成部と同じ構成部については、同じ符号を付与して説明する。

【0374】

まず、演出制御装置 150 は、電源装置 160 から出力されたりセット信号が伝達される (2321)。なお、ステップ 2321 の処理は、ステップ 2301 の処理と同じである。

【0375】

そして、演出制御装置 150 にリセット信号が入力されると、演出制御装置 150 の出力ポートがハードウェアにより初期化される (2322)。

10

20

30

40

50

## 【0376】

そして、演出制御装置150は、電源投入時の初期化処理を実行する(2323)。電源投入時の初期化処理は、RAM154等を初期化する処理であって、CPU152によって実行される。

## 【0377】

次に、演出制御装置150は、遊技制御装置100からの指令を受信可能な状態を発生させる(2324)。そして、遊技制御装置100から送信された指令が初期化指令であるか否かを判定する(2326)。

## 【0378】

演出制御装置150は、遊技制御装置100から送信された指令が初期化指令でないと判定された場合には(2326の結果が「N」)、初期化指令が取り込まれるまで待機する。

10

## 【0379】

一方、演出制御装置150は、遊技制御装置100から送信された指令が初期化指令であると判定された場合(2326の結果が「Y」)、演出制御装置150は通信開始時の初期化処理を実行し(2327)、演出制御装置メイン処理へ移行する。

## 【0380】

図24は、本発明の第1の実施の形態の遊技制御装置プログラム開始準備処理を説明するフローチャートである。なお、遊技制御装置プログラム開始準備処理の前半(プログラムカウンタ1234にリセットアドレスを設定するまでの間)は、ユーザプログラムROM602(図5参照)に記憶された遊技制御プログラムによって実行される処理ではなく、CPUコア102に備わる初期値設定回路124(図12参照)によって実行される処理である。

20

## 【0381】

CPUコア102は、セキュリティ回路630からのリセット信号を受け入れると、内蔵リセット回路1240により、初期値設定回路1241を動作させる。このとき、内蔵リセット回路1240は、初期値設定回路1241が各レジスタに初期値を設定するまで、命令解釈実行回路1242の動作を待機させる。

## 【0382】

初期値設定回路1241が動作を開始すると、スタックポインタ1233に初期値「29FFH」を設定し、Kレジスタ1230に「28H」を設定し、HLレジスタ1212Aにリセットアドレスである「4000H」を設定し、他のレジスタには「00H」の値(16ビットレジスタには「0000H」の値)を設定する(2401)。なお、リセットアドレスは「0000H」としてもよい。

30

## 【0383】

次いで、初期値設定回路1241は、プログラムカウンタ1234に、リセットアドレスである「4000H」の値を設定する(2402)。これにより、各レジスタに初期値が設定される。次いで、内蔵リセット回路1240は、待機させていた命令解釈実行回路1242を動作させると、命令解釈実行回路1242は、プログラムカウンタ1234が示すアドレス(遊技制御プログラムが記憶されたユーザプログラムROM602のリセットアドレスに相当)からプログラムの実行を開始する。命令解釈実行回路1242がプログラムを実行開始した後の処理は、図25の遊技制御装置メイン処理にて説明する。

40

## 【0384】

次に、遊技制御装置100のCPU102によって実行される遊技制御装置メイン処理を、図25及び図26を用いて説明する。

## 【0385】

図25は、本発明の第1の実施の形態の遊技制御装置メイン処理の前半部のフローチャートであり、図26は、本発明の第1の実施の形態の遊技制御装置メイン処理の後半部のフローチャートである。なお、これらの処理は遊技制御装置100によって実行されるが、厳密には、命令解釈実行回路1242が、CPUコア102内部の各レジスタを参照し

50

ながら、遊技制御プログラムを実行することで実現される。

【0386】

まず、遊技制御装置100は、CPU102への割込みを禁止する(2501)。そして、遊技制御装置100は、スタックポインタ1233(図12)に初期値「29FFH」を設定する(2502A)。これにより、スタックポインタ1233の値が「29FFH」でなければ「29FFH」に変更される。なお、スタックポインタ1233に初期値「29FFH」が既に設定されていても、再度「29FFH」の値が設定される。

【0387】

次いで、遊技制御装置100は、Kレジスタ1230に「28H」の値を設定する(2502B)。これにより、Kレジスタ1230の値が「28H」でなければ「28H」に変更される。なお、Kレジスタ1230に初期値「28H」が既に設定されていても、再度「28H」の値が設定される。

10

【0388】

次いで、遊技制御装置100は、割込モードを設定する(2503)。割込モードは、CPU102が内蔵デバイスからの割込要求の処理を可能とし、また、プログラムにおいて割込要求の処理を実行する位置を設定することを可能とするものである。

【0389】

次に、遊技制御装置100は、入力I/F105からRAMクリアSW信号の状態を取り込み、取り込んだRAMクリアSW信号の状態をCPU102のレジスタに記憶する(2504)。

20

【0390】

そして、遊技制御装置100は、RAM104を使用しないディレイ処理を実行する(2505)。このディレイ処理は、所定時間、処理を待機させる処理であり、具体的には、チェックサムが算出されない記憶領域にて、所定の数になるまでデクリメントし続ける処理である。ディレイ処理は、この所定の数に待機させる時間に対応する時間に設定することによって、所定時間を計時するタイマ計時手段となる。なお、ディレイ処理の詳細については、図28にて後述する。

【0391】

次に、遊技制御装置100は、再度、入力I/F105からRAMクリアSW信号の状態を取り込み、取り込んだRAMクリアSW信号の状態をCPU102のレジスタに記憶する(2506)。なお、CPU102が二つのRAMクリア信号の状態を比較できるように、ステップ2504の処理でRAMクリアSW信号の状態を記憶するレジスタの領域、及び、ステップ2506の処理でRAMクリアSW信号の状態を記憶するレジスタの領域は、異なる領域である。

30

【0392】

次に、遊技制御装置100は、図23のステップ2303の処理で発生したRAM書込禁止状態をRAM書込可能状態にする(2507)。

【0393】

具体的には、CPU102の指令によって、フリップフロップ回路641のクロック端子にクロック信号を出力制御回路612から入力させ、かつ、フリップフロップ回路641のデータ端子に接続された信号線の信号レベルをハイレベルにする。これにより、フリップフロップ回路641の出力端子Q(正論理)からハイレベルの信号が出力され、出力端子Q(負論理)からローレベルの信号が出力されるため、ORゲート回路642の入力端子にローレベルの信号が入力されることにより、RAM書込可能状態になる。

40

【0394】

次に、遊技制御装置100は、スタック領域706を使用して、各種設定処理を実行する(2508)。この設定処理は、例えば、サブルーチンや関数を呼び出して、遊技制御に必要な各種記憶領域に初期データを設定する処理である。この設定処理において、CPU102により、図7~図9で前述した、送信シリアルチャンネル設定レジスタ633、送信制御レジスタ632、送信データステータスレジスタ631の各ビットの初期値が設

50

定されることで、送信ポーレート等の設定が行われる。

【0395】

これらのサブルーチンや関数は、遊技制御プログラムに記述した複数の箇所から呼び出される形態となっており、遊技制御プログラムの容量削減に貢献している。一方で、サブルーチンや関数を呼び出す際には、前述したように、戻りアドレスをスタック領域706に待避する処理を必要とする。

【0396】

そして、遊技制御装置100は、ステップ2504の処理でレジスタに記憶されたRAMクリアSW信号の状態とステップ2508の処理でレジスタに記憶されたRAMクリアSW信号の状態とを比較して、どちらのRAMクリアSW信号の状態も、RAMクリアSW162が操作されたことを示しているか否かを判定する(2509)。

10

【0397】

ステップ2509の処理では、異なるタイミングで取得したRAMクリア信号の状態に基づいてRAMクリアSW162が操作されたか否かを判定しているため、ノイズ等による誤判定を防止できる。

【0398】

ステップ2509の処理で、RAMクリアSW162が操作されたと判定された場合、遊技制御装置100は、ユーザワークRAM104のすべての記憶領域を初期化する(2510)。

【0399】

そして、遊技制御装置100は、初期化指令信号を払出制御装置210及び演出制御装置150へ送信し(2511)、図26に示すステップ2517の処理に進む。

20

【0400】

一方、ステップ2509の処理で、RAMクリアSW162が操作されていないと判定された場合、遊技制御装置100は、ユーザワークRAM104の第1停電復旧領域701及び第2停電復旧領域703に、電源遮断確認フラグが格納されているか(正確には、電源遮断確認フラグがオンとなっているか)を確認する(2512)。

【0401】

そして、遊技制御装置100は、直前の電源供給停止のときに、電源遮断の処理が正しく実行されていたか否かを判定する(2513)。具体的には、遊技制御装置100は、第1停電復旧領域701及び第2停電復旧領域703の両方に電源遮断確認フラグが格納されている場合には、電源遮断の処理が正しく実行されているものであると判定し、一方、第1停電復旧領域701及び第2停電復旧領域703の少なくとも一方に電源遮断確認フラグが格納されていない場合(少なくとも一方の電源遮断確認フラグがオフの場合)には、電源遮断の処理が正しく実行されていないと判定する。

30

【0402】

ステップ2513の処理で電源遮断の処理が正しく実行されていたと判定された場合には、遊技制御装置100は、ユーザワークRAM104の第1停電復旧領域701、ワークエリア702、及び第2停電復旧領域703を用いてチェックサムを算出して、算出したチェックサムがチェックサム領域704に格納されているチェックサムと一致するか否かを照合する(2514)。

40

【0403】

なお、チェックサム領域704に格納されているチェックサムは、停電検出時のユーザワークRAM104の第1停電復旧領域701、ワークエリア702、及び第2停電復旧領域703を用いてチェックサムを算出して、格納されたものである。

【0404】

つまり、ステップ2514の処理は、停電検出時のユーザワークRAM104に格納された情報と電源投入時のユーザワークRAM104に格納された情報とが一致するか否かを照合する処理である。

【0405】

50

そして、ステップ 2 5 1 4 の処理の照合結果が、算出したチェックサムとチェックサム領域 7 0 4 に格納されたチェックサムとが一致するものであるか否かを判定する ( 2 5 1 5 )。

【 0 4 0 6 】

ステップ 2 5 1 4 の処理で算出したチェックサムとチェックサム領域 7 0 4 に格納されたチェックサムとが一致しないとステップ 2 5 1 5 の処理で判定された場合、つまり、停電検出時のユーザワーク R A M 1 0 4 に格納された情報と電源投入時のユーザワーク R A M 1 0 4 に格納された情報とが一致しない場合には、遊技制御装置 1 0 0 は、ステップ 2 5 1 0 の処理に進み、ユーザワーク R A M 1 0 4 のすべての領域を初期化し、ステップ 2 5 1 1 の処理にて初期化指令を払出制御装置 2 1 0 及び演出制御装置 1 5 0 に送信する。

10

【 0 4 0 7 】

一方、ステップ 2 5 1 4 の処理で算出したチェックサムとチェックサム領域 7 0 4 に格納されたチェックサムとが一致するとステップ 2 5 1 5 の処理で判定された場合、つまり、停電検出時のユーザワーク R A M 1 0 4 に格納された情報と電源投入時のユーザワーク R A M 1 0 4 に格納された情報とが一致する場合には、遊技制御装置 1 0 0 は、遊技制御装置 1 0 0 の起動に必要な領域 ( ユーザワーク R A M 1 0 4 の一部の領域 ) を初期化する ( 2 5 1 6 )。このとき、ユーザワーク R A M 1 0 4 の第 1 停電復旧領域 7 0 1 及び第 2 停電復旧領域 7 0 3 の各々にて、電源遮断確認フラグが消去 ( 正確には、各領域にて電源遮断確認フラグがオフ ) される。そして、遊技制御装置 1 0 0 は、初期化指令を払出制御装置 2 1 0 及び演出制御装置 1 5 0 に送信する ( 2 5 1 1 )。

20

【 0 4 0 8 】

これらの処理が完了すると、遊技制御装置 1 0 0 に関する初期化処理が完了となる。次いで、図 2 6 に示すステップ 2 5 1 7 の処理に進む。

【 0 4 0 9 】

次に、ステップ 2 5 1 1 の処理で初期化指令が払出制御装置 2 1 0 及び演出制御装置 1 5 0 に送信された後、遊技制御装置 1 0 0 は、各種時間を計測やタイマ割込みを行うための C T C ( Counter Timer Circuit ) を起動し ( 2 5 1 7 )、遊技制御に関する乱数を生成する乱数生成回路 6 0 8 ( 図 5 ) を初期化する ( 2 5 1 8 )。そして、遊技制御装置 1 0 0 は、ステップ 2 5 0 1 の処理で禁止された C P U 1 0 2 への割込みを許可する ( 2 5 1 9 )。

30

【 0 4 1 0 】

次に、遊技制御装置 1 0 0 は、初期値乱数を更新する初期値乱数更新処理を実行する ( 2 5 2 0 )。初期値乱数とは、遊技制御に関する乱数のカウンタ ( 例えば、始動入賞口へ入賞したタイミングで取得される乱数のカウンタ ) が上限値に達した場合に初期値に戻るが、その初期値を決定するための乱数である。

【 0 4 1 1 】

そして、遊技制御装置 1 0 0 は、停電検出信号が入力されたか否かを確認し ( 2 5 2 1 )、ステップ 2 5 2 1 の処理での確認結果が、停電検出信号が入力されたことを示すか否かを判定する ( 2 5 2 2 )。

【 0 4 1 2 】

40

遊技制御装置 1 0 0 は、停電検出信号が入力されていないと判定した場合には ( ステップ 2 5 2 2 の結果が「 N 」)、停電は発生していないので、ステップ 2 5 2 0 の処理に戻る。

【 0 4 1 3 】

一方、ステップ 2 5 2 2 の処理で、停電検出信号が入力されたと判定された場合には ( ステップ 2 5 2 2 の結果が「 Y 」)、遊技制御装置 1 0 0 により停電処理が実行される。まず、遊技制御装置 1 0 0 は、C P U 1 0 2 への割込みを禁止し ( 2 5 2 3 )、出力 I / F 1 0 6 に備わる出力ポートの電圧レベルをローレベルに設定する ( 2 5 2 4 )。

【 0 4 1 4 】

次に、遊技制御装置 1 0 0 は、ユーザワーク R A M 1 0 4 の第 1 停電復旧領域 7 0 1 及

50

び第2停電復旧領域703に、電源遮断確認フラグを格納（正確には、各領域にて電源遮断確認フラグをオン）し（2525）、ユーザワークRAM104の第1停電復旧領域701、ワークエリア702、及び第2停電復旧領域703を用いてチェックサムを算出して、算出したチェックサムをチェックサム領域704に格納する（2526）。

【0415】

次に、遊技制御装置100は、RAMアクセス規制回路640によってユーザワークRAM104をRAM書込禁止状態にする（2527）。

【0416】

具体的には、CPU102の指令によって、フリップフロップ回路641のクロック端子にクロック信号を出力制御回路612から入力させ、かつ、フリップフロップ回路641のデータ端子に接続された信号線の信号レベルをローレベルにする。これにより、フリップフロップ回路641の出力端子Q（正論理）からローレベルの信号が出力され、出力端子Q（負論理）からハイレベルの信号が出力されるため、ORゲート回路642の入力端子にハイレベルの信号が入力されることにより、RAM書込禁止状態になる。

10

【0417】

そして、遊技制御装置100は、遊技機1の電源が切れるまで待機する（2528）。なお、遊技制御装置100には、バックアップ電源が接続されているので、停電が発生しても、すぐに電源が切れることはない。

【0418】

なお、本実施形態では、ステップ2514の処理で電源断時のユーザワークRAM104と電源投入時のユーザワークRAM104との正当性を判定する前のステップ2507の処理でRAM書込可能状態にしたが、RAM書込可能状態にするタイミングは、遅くともステップ2514の処理の正当性に依じて行われるステップ2510又は2516の処理におけるユーザワークRAM104の初期化処理の実行直前であればよい。

20

【0419】

このように、遊技機1にて電源供給が遮断した場合には、必要な電源遮断処理を実行した後は、ユーザワークRAM104をRAM書込禁止状態に設定し、遊技機1にて再度電源供給が復帰したときでも、すぐにユーザワークRAM104をRAM書込可能状態としないで、ハードウェアに関する初期化処理を一定時間実行し、ステップ2514の処理の正当性に依じて行われるステップ2510又は2516の処理におけるユーザワークRAM104の初期化処理の実行直前になって、ようやくRAM書込可能状態にすることによって、ユーザワークRAM104の初期化まで不用意なユーザワークRAM104の書き込みを防止できる。

30

【0420】

そのため、ステップ2514の処理における正当性判定が行われる直前には、RAM書込禁止状態になっているので、電源投入後にユーザワークRAM104に誤った書き込みがなされ、ステップ2514の処理で誤った判定がされることを防止できる。

【0421】

なお、本実施形態では、ステップ2508の処理でスタック領域706を用いた各種設定処理を実行するために、ステップ2514の処理における正当性判定処理の前のステップ2507の処理でRAM書込可能状態にしている。

40

【0422】

これによって、正当性判定を行う前に正当性判定の対象とはならないスタック領域706を用いた各種設定処理を行うことができるようになるため、遊技制御装置100の各種設定を早い段階で行うことができるので遊技制御装置100の起動を高速化でき、また、スタック領域706を用いるので処理プログラムが共通化でき、プログラム容量を削減できる。

【0423】

なお、図15では、ステップ2510又は2516の処理でユーザワークRAM104を初期化した後、ステップ2515の処理で初期化指令信号を送信しているが、ステップ

50

2514における正当性判定の実行前のステップ2508の処理の実行後に初期化指令信号を送信してもよい。

【0424】

この場合には、ステップ2514の処理における正当性判定の実行前であるので、正当性判定に寄与しないスタック領域706又はCPU102に備わるレジスタを用いて、初期化指令信号を送信する。

【0425】

なお、CPU102に備わるレジスタを用いなくても、例えば、ユーザワークRAM104とは別個に、遊技制御装置100に所定のタイマ回路などを設けて、このタイマ回路に備えられた記憶領域を更新させるような方法でも実現可能である。言い換えれば、正当性判定に影響のない記憶領域であれば、どのようなものを用いても適用が可能であり、好ましくは、ユーザワークRAM104と記憶領域とを分離できれば、CPU102のプログラムも簡素化されるということである。

【0426】

ステップ2510又は2516の処理では、RAM104の一部領域を初期化する処理であるステップ2516の処理が、RAM104の全領域を初期化する処理であるステップ2510の処理よりも実行時間が長いため、ステップ2510の処理を実行するかステップ2516の処理を実行するかによって、初期化指令信号が送信される時間が異なってしまう。

【0427】

ステップ2514の処理における正当性判定の実行前に初期化指令信号を送信することによって、ステップ2511の処理で初期化指令信号を送信するよりも早く初期化指令信号を送信できる。また、電源投入から一定時間で初期化指令信号を送信することができる。

【0428】

図27は、本発明の第1の実施の形態の遊技制御装置メイン処理(図25)におけるステップ2501からステップ2502Bまでの処理を説明する図である。ここでは、図12で前述した各種レジスタ、図21及び図22で前述したアセンブリ言語を用いて説明を行うことにする。

【0429】

なお、図27における列2710は、遊技制御プログラムが格納されているユーザプログラムROM103のアドレスを示し、列2720は、当該アドレスに格納されているデータを示している。例えば、ユーザプログラムROM103の「4000H」のアドレスには「F3H」のデータが格納され、続く「4001H」のアドレスには「31H」のデータが格納され、続く「4002H」のアドレスには「00H」のデータが格納され、続く「4003H」のアドレスには「2AH」のデータが格納されている。

【0430】

行2701の「DI」命令は、遊技制御装置メイン処理(図25)のステップ2501の割込禁止処理に対応し、この命令に対応するコード「F3H」の値がアドレス「4000H」の領域に格納される。

【0431】

行2702の命令は、同じくステップ2502Aの処理に対応し、スタックポインタ(SP)に対応するSPレジスタ1233に初期アドレスとして「29FFH」を設定する。

【0432】

行2703の命令は、同じくステップ2502Bの処理に対応し、Kレジスタ1230に、上位アドレスのデフォルトの初期値「28H」を設定する。このように、Kレジスタ1230に予め「28H」を設定しておくことによって、「28H」を上位バイトとし、さらに任意の1バイトの値で示される値を下位バイトとして合成した2バイトの値をアドレスと見なして、そのアドレスの領域に記憶されている値を所定のレジスタに設定するこ

10

20

30

40

50



とができる。例えば、「2803H」のアドレスに記憶された値をAレジスタ1202Aに設定したい場合には、「LDK A, (03H)」という命令を実行すればよい。こうすることによって、2バイトのアドレスの全てを直接指定する場合よりもコード量を1バイト削減することができる。特に、特定の領域を指定する頻度が大きいほど削減されるコード量を大きくすることができる。本実施形態では、例えば、変動表示ゲームが実行されるたびにシフトされ、アクセス頻度の多い始動記憶を格納するアドレスを指定する場合に使用される。また、Kレジスタを有効に活用するために、これらの始動記憶の上位アドレスは「28H」となるように配置されている。

#### 【0433】

なお、ユーザプログラムROM103の「4000H」のアドレスは、前述のように、リセットアドレスである。よって、リセット信号の発生時には、まず、この「4000H」に配置されたコードの命令が実行され、その後は、以降のアドレスに配置された命令が順次実行される。

10

#### 【0434】

ちなみに、リセット信号が発生すると、初期値設定回路1241によってSPレジスタ1233に「29FFH」がハード的に設定されるので、行2702の命令は、省略（言い換えれば、図25のステップ2502Aを省略）することも可能である。同様に、リセット信号が発生すると、初期値設定回路1241によってKレジスタ1230に「28H」がハード的に設定されるので、行2703の命令は、省略（言い換えれば、図25のステップ2502Bを省略）することも可能である。ただし、初期値設定回路1241が機能せずに、プログラムカウンタ1234の値のみがリセットアドレスの値に変更される事態を想定するのであれば、行2702及び行2703の命令を配置しておくことにも意味がある。

20

#### 【0435】

図28は、本発明の第1の実施の形態のディレイ処理を説明する図である。

#### 【0436】

図28に示すディレイ処理は、図25のステップ2505で実行されるが、当該ディレイ処理を実行している時点では、ユーザワークRAM104の値が更新できないようにRAM書込禁止状態となっている。これは、直前の停電発生時に格納されたチェックサムと、電源投入直後となる現時点でのチェックサムとの照合を行うためである。

30

#### 【0437】

このため、図25に示すステップ2505におけるディレイ処理では、正当性の判定が行われる記憶領域が含まれたユーザワークRAM104を用いずに、他の記憶領域（正当性判定の対象とならない判定対象外記憶領域）を用いてディレイ処理を実行しなければならない。そこで、本実施形態のディレイ処理は、CPUコア102に備わるレジスタ（汎用レジスタ）を用いて実行される。

#### 【0438】

以下に、判定対象外記憶領域を含むユーザワークRAM104の記憶領域をまったく利用せずに、レジスタを用いたディレイ処理を説明する。なお、CPUコア102として、Z80系のCPUを用いるものとするので、Z80系のCPUで使用されるレジスタ及びアセンブリ言語を用いて説明を行う。

40

#### 【0439】

なお、図28における列2810は、図27の列2710同様に遊技制御プログラムが格納されているユーザプログラムROM103のアドレスを示し、列2820は、図27の列2720同様に、当該アドレスに格納されているデータを示している。

#### 【0440】

まず、行2801は、当該ディレイ処理の最初の処理に相当し、CPUコア102のレジスタ（図11参照）のHレジスタ及びLレジスタを1つのペアとして構成したHLレジスタに、「0603H」をロードする。具体的には、Hレジスタに「06H」がロードされ、Lレジスタには「03H」がロードされる。

50

## 【0441】

次に、行2802を実行し、HLレジスタの値をデクリメント(1減算)する。したがって、行2802が最初に実行された後、HLレジスタの値は「0602H」となる。

## 【0442】

続いて、行2803を実行し、Hレジスタに格納された値をAレジスタにロードする。そして、行2804を実行し、AレジスタとLレジスタとの論理和を算出する。行2805では、行2804で算出された論理和がゼロでなければ(NZ)、行2802(アドレスが4023Hとなる行)に戻る。したがって、Hレジスタ及びLレジスタの両方が「00H」となるまで、行2802から2805までの処理を繰り返すことになる。

## 【0443】

つまり、図28では、維持タイマとして使用されるHレジスタ及びLレジスタに格納された「0603H」(=1539)が「0000H」になるまでデクリメントされるもので、合計1539回デクリメントが行われる。この間、図25に示す遊技制御装置メイン処理は、ステップ2505の処理で待機するため、遊技制御装置100の起動が遅延することとなる。

## 【0444】

ここで遅延時間を具体的に算出する。遅延時間は、行2802から2805までの処理を、繰り返し回数(「0603H」=1539)分だけ実行した時間となる。そこで、行2803から2805までの各行の実行時間を算出する。各行には実行される命令が定義されており、各命令には、実行に必要なCPUのクロックサイクル数(ステート数)が対応している。したがって、1ステート当りの時間を各命令に対応するステート数に乗じることによって各命令の実行時間を算出することができる。

## 【0445】

本発明の第1の実施の形態では、クロック数20MHzを2倍に分周した10MHzがCPUの動作クロックとなるため、 $1/10000000 = 100n$ 秒が1ステート当りの処理時間となる。以下、具体的に各行の処理時間を算出する。

## 【0446】

行2802で実行されるデクリメント「DEC」命令のステート(数)は6となっている。したがって、行2802の処理時間は $6 \times 100n$ 秒 =  $600n$ 秒となる。同様に、行2803で実行されるロード「LD」命令のステート(数)は4、及び、行2804で実行されるロード「OR」命令のステート(数)は4となっており、それぞれの処理時間は $4 \times 100n$ 秒 =  $400n$ 秒となっている。さらに、行2805で実行されるジャンプ「JP」命令のステート(数)は行2804の演算結果が非0の場合には12、0の場合には7となっている。0の場合は遅延時間終了時だけであるため、ステート数を12とすると、処理時間は $12 \times 100n$ 秒 =  $1200n$ 秒となる。

## 【0447】

以上より、1回の繰り返しにおける処理時間は、 $600n$ 秒 +  $400n$ 秒 +  $400n$ 秒 +  $1200n$ 秒 =  $2600n$ 秒となる。そして、繰り返し回数は、1539回であるため、 $2600n$ 秒  $\times$  1539 =  $4.0014m$ 秒となり、約4秒の遅延時間となる。したがって、この場合のディレイ処理は、4秒間を計時するタイマ計時手段となっている。

## 【0448】

また、このディレイ処理中は、ユーザワークRAM104へのアクセスが全く行われない。すなわち、正当性の判定が行われる記憶領域が含まれたユーザワークRAM104の値を書き換えることなく、ディレイ処理を実行することができる。

## 【0449】

図28で説明したように、本実施形態では、ハードウェアを用いずに、正当性判定に寄与しない、つまり、チェックサムを算出しない領域を用いてソフトウェアにより実現(維持タイマを計時)しているので、図25に示すステップ2514の正当性判定を正確に行うことができるとともに、ハードウェアでディレイ処理を実現するよりも安価に実現することができる。

10

20

30

40

50

## 【0450】

例えば、ハードウェアでディレイ処理を実現する遊技機として、特開2002-224394号公報に開示されるような技術が知られており、この遊技機では、電源が断たれた後の復帰時に、払出しの不都合な状態が解消するまで賞媒体の払出し動作を停止できるようにすること、さらに、賞媒体の払出しに関して遊技者とホール側とでトラブルが発生しないようにすることを目的として、停電からの復帰時に、払出し制御手段が主制御手段よりも先に起動して払出し制御が開始された場合、初期化スイッチが操作されていないため、払出し動作復帰処理が実行され、その後、主制御手段から払出し再開コマンドを受信するまで、払出し動作を停止して、払出し再開可能な状態で待機する構成となっている。

## 【0451】

さらに、この遊技機では、後から起動した主制御手段は補給切れ検出スイッチや満杯検出スイッチからの検出スイッチに基づいて払出しに関するエラーを検出しない場合に、主制御手段から払出し再開コマンドが送信されてくるので、払出し制御手段はその払出し再開コマンド受信をきっかけに払出し動作を再開する構成となっている。

## 【0452】

そして、この遊技機は、主制御手段を、払出し制御手段よりも遅延させて起動させるために、主制御手段（主制御基板39）に遅延回路90を設けて、リセット信号発生手段77からのリセット信号が、払出し制御手段（払出し制御基板46）に到達するよりも時間tだけ遅延して主制御手段に到達するように構成している（特開2002-224394号公報の段落[0051]～[0053]、図9、図11参照）、遅延回路90などのハードウェアが必要であるため、コストが高くなってしまいう問題があった。また、遅延回路90はハードウェアで構成されているため、遅延の時間値をプログラムで変更できないという問題もあった。

## 【0453】

この場合、遅延回路90に相当する機能を、主制御手段（主制御基板39）に設けたCPUを用いてソフトウェアによって実現すれば、コスト面での課題が解決するが、CPUを用いて遅延時間を計時するためには、主制御手段（主制御基板39）のバックアップ用メモリ39bを用いなければならず、この場合、主制御手段が起動後にバックアップ用メモリ39bの正当性を確認して、バックアップ用メモリ39bが使用可能な状態になってから遅延時間を計時するので、遊技機全体の起動が遅れてしまうという課題を残していた。そのため、ソフトウェアによって遊技制御装置の起動を従属制御装置の起動よりも遅延させることによってコストダウンを図りつつも、遊技機全体の起動が遅延してしまうことを防止する遊技機が提供されることが望まれていた。

## 【0454】

本実施形態に戻って、CPUコア102で利用できるレジスタの数が少ない場合には、スタック領域を利用してディレイ処理を実行するほうが有効である。ただし、正当性判定の対象となっている第1停電復旧領域701、ワークエリア702、第2停電復旧領域703、チェックサム領域704の各記憶領域を、ノイズ等によって書き換えてしまうことを極力防止したいのであれば、ディレイ処理中を通してユーザーワークRAM104をRAM書込禁止状態とし、図28に示したスタック領域を利用しないディレイ処理を実行するほうが有効ともいえる。

## 【0455】

また、動作クロック数を高く設定することによってCPUによる演算処理速度を高速化することが可能となるが、演算処理速度を高速化すると、ディレイ処理におけるループ回数（図28では「0603H」=1539回）を高速化した分だけ大きくする必要はある。しかしながら、演算処理速度を高速化しすぎると、ループ回数が大きくなりすぎてしまい、ループ回数を格納するレジスタのバイト数が所定バイト数（例えば、2バイト）を超えてしまい、プログラムの容量が大きくなってしまふ。そこで、図28にて説明したように、ループ回数を格納するレジスタのバイト数が2バイトに収まる（ループ回数が65536回を超えない）ようにCPU102の動作速度（クロック数）を設定することによ

10

20

30

40

50

て、高速通信を実現しながらもプログラム容量の増大を抑えることが可能となる。

【0456】

図29は、本発明の第1の実施の形態のタイマ割込処理を示すフローチャートである。このタイマ割込処理は、遊技制御装置100のCPUコア102によって実行される。

【0457】

遊技機の電源が投入されると、遊技制御装置メイン処理（図25及び図26参照）が実行される。そして、ステップ2517の処理で起動させたCTCによって、所定時間周期（例えば、4ミリ秒周期）でタイマ割込みが発生すると、遊技制御装置100のCPU102によって、タイマ割込処理が繰り返し実行される。ただし、これらの処理（2912～2922の処理）は、割り込み発生毎に必ずしもすべて行なわれなくてもよい。例えば、ステップ2912の入出力処理においては、毎回入力信号を監視するが、出力処理は割り込みの発生のおきにおきに実行されてもよい。つまり、1回の割込処理で一通りの処理をすべて完了するのではなく、この割込処理が複数回繰り返し実行されて一連の遊技制御処理が完了するようにしてもよい。

10

【0458】

本実施形態のタイマ割込処理において、遊技制御装置100は、まず、レジスタのデータをスタック領域706に退避する（2911）。ここでは、遊技制御プログラムに記述された命令を実行することにより、CPUコア102の内部のレジスタ（図12参照）の中から、タイマ割込処理が呼び出される直前の処理（呼出元処理）で使用されていたものを選択して、スタック領域706に退避させる処理が行われる。

20

【0459】

ただし、フラグレジスタ1200（図12参照）に関しては、図19の（D）に示すように、当該タイマ割込処理が開始する時点で既にスタック領域706に退避されているので、改めて命令を実行させる必要はない。

【0460】

次に、遊技制御装置100は、入出力処理を実行する（2912）。入出力処理は、入力処理と出力処理とを含む。入力処理は、入力I/F105を介して各種センサ（特図始動SW34A、普図始動SW31A、カウントSW36A、入賞口SW32A～32N、オーバーフローSW109、球切れSW110、枠開放SW111など）から入力される信号にチャタリング除去等の処理をし、入力情報を確定する処理である。

30

【0461】

出力処理は、出力I/F106を介して、特図ゲーム処理（2919）及び普図ゲーム処理（2920）にて設定されたパラメータに基づいて、特図表示器120、普図表示器121、普電SOL90、及び大入賞口SOL38を制御するための信号を出力する。

【0462】

なお、前述したように、入力処理と出力処理とは1回のタイマ割り込みで同時に実行されなくてもよい。

【0463】

次に、遊技制御装置100は、各種処理で送信バッファにセットされた（コマンド）を演出制御装置150及び払出制御装置210等に出力するコマンド送信処理を行う（2913）。具体的には、演出制御装置150に特別図柄変動表示ゲームに係わる演出指令信号（演出コマンド）を出力したり、払出制御装置210に排出指令信号（払出指令信号、払出コマンド）を出力したりする。コマンド送信処理の詳細については、図31にて後述する。なお、払出コマンドについては図33にて詳細を説明し、演出コマンドについては図35にて詳細を説明する。

40

【0464】

その後、遊技制御装置100は、特別図柄変動表示ゲームの当りはずれを判定するための大当り乱数カウンタ（図15の大当り乱数の生成領域）の値を1ずつ加算する乱数更新処理1を行う（2914）。なお、この乱数更新処理1では、特別図柄変動表示ゲームの停止図柄を決定する大当り図柄乱数カウンタ（図15の大当り図柄乱数の生成領域）の値

50

、普通図柄変動表示ゲームの当りはずれを判定するための普図乱数カウンタ（図 15 の普図乱数生成領域）にも 1 ずつ加算する。

【0465】

次に、遊技制御装置 100 は、乱数の初期値を更新し、乱数の時間的な規則性を崩すための初期値乱数更新処理を実行する（2915）。ステップ 2915 の初期値乱数更新処理は、図 26 に示す初期値乱数更新処理（2520）と同じなので、説明を省略する。

【0466】

そして、遊技制御装置 100 は、特別図柄変動表示ゲームに関連した飾り特別図柄変動表示ゲームにおける変動表示パターンを決定する乱数を更新するための変動表示パターン乱数カウンタ（図 15 の第 1 変動 P 乱数の生成領域、第 2 変動 P 乱数の生成領域、第 3 変動 P 乱数の生成領域）の値を 1 ずつ加算する乱数更新処理 2 を行う（2916）。 10

【0467】

次に、遊技制御装置 100 は、各入賞口に遊技球が入賞していないかを監視するために、入賞口監視処理を行う（2917）。具体的には、第 1 特図始動 SW37A、第 2 特図始動 SW34A、普図始動 SW31A、カウント SW36A、入賞口 SW32A ~ 32N、から信号の入力があるか否か（遊技球の検出を示す信号が入力されているか否か）を監視する。

【0468】

このとき、大当たりが発生して特別変動入賞装置 36 が開放中であるときには、カウント SW36A により遊技球の検出数を計数して、特別変動入賞装置 36 へ入賞した遊技球の数を計数する。この計数結果が所定数（例えば 10 個）になると、特別変動入賞装置 36 を閉止して、次のラウンドに更新する処理を行う。 20

【0469】

また、このとき、普図始動 SW31A による遊技球の検出があれば、普図保留カウンタの値が所定数未満（例えば、「4 個」未満）であることを条件に「1」だけ増加させ、普図乱数カウンタ値（図 15 の普図乱数生成領域）が、普図保留カウンタに対応する普図始動入賞記憶領域（図 15）に記憶される。

【0470】

なお、第 1 特図始動 SW37A による遊技球の検出があれば、後述する「始動口 SW 監視処理」にて、各種乱数値が、第 1 特図始動入賞記憶領域（図 16）に記憶される。同様に、第 2 特図始動 SW34A による遊技球の検出があれば、後述する「始動口 SW 監視処理」にて、各種乱数値が、第 2 特図始動入賞記憶領域（図 17）に記憶される。 30

【0471】

その後、遊技制御装置 100 は、排出球の球詰まりや、各種スイッチ、センサ等の異常などを監視するエラー監視処理を行う（2918）。

【0472】

その後、遊技制御装置 100 は、特別図柄変動表示ゲームに関する処理を行う特図ゲーム処理（2919）、普通図柄変動表示ゲームに関する処理を行う普図ゲーム処理（2920）を行う。

【0473】

特図ゲーム処理（2919）は、第 1 特図始動 SW37A 及び第 2 特図始動 SW34A で検出された始動入賞口への遊技球の入賞に基づいて抽出され、特別図柄始動入賞記憶に記憶された特別図柄乱数カウンタ値が当りが否か判定し、特図表示器 120 で特別図柄変動表示ゲームを実行する。なお、特図始動入賞記憶には、直ちに前記変動表示ゲームを実行することができない状態で始動入賞口に遊技球が入賞した場合に、抽出された乱数が始動入賞記憶として記憶される。なお、特図ゲーム処理の詳細は、図 37 にて後述する。 40

【0474】

また、特図ゲーム処理（2919）では、特図表示器 120 の表示に対応する識別情報の変動表示のための処理を行う。抽出された乱数が所定の値であれば、特別図柄に関する当り状態となり、識別情報の変動表示が当り図柄で停止する。また、当り状態になると、 50

特別変動入賞装置 36 に遊技球を受け入れやすい開状態になる。

【0475】

普図ゲーム処理(2920)は、普図始動SW31Aで検出された普通図柄始動ゲート31への遊技球の通過に基づいて抽出され、普通図柄始動入賞記憶に記憶された普図乱数カウンタ値(2917の処理で抽出・記憶された普通図柄変動表示ゲームの結果に関する乱数)が当りか否かを判定し、普図保留カウンタの値を1だけ減算して、普図表示器121で普通図柄の変動表示ゲームを実行する。普図乱数カウンタ値が所定の値であれば、普図に関する当り状態となり、普通図柄の変動表示が当り状態で停止するためのパラメータを設定する。

【0476】

次に、遊技制御装置100は、遊技機1に設けられ、遊技に関する各種情報を表示するセグメントLED(特図表示器120及び普図表示器121)に出力する信号を編集する処理を行う(2921)。具体的には、特別図柄変動表示ゲームが開始されると、今回開始した特別図柄変動表示ゲームの実行回数を減じた特別図柄入賞記憶数を特図表示器120の特図記憶表示部に表示するためのパラメータを編集する。同様に、普通図柄の変動表示ゲームが開始されると、今回開始した普通図柄変動表示ゲームの実行回数を減じた普通図柄入賞記憶数を普図表示器121の普図記憶表示器に表示するためのパラメータを編集する。

【0477】

その後、遊技制御装置100は、検査装置接続端子107を介して接続される管理用コンピュータに遊技機1の状態を出力するための外部情報を編集する外部情報編集処理を行う(2922)。外部情報には、図柄が確定したか、当りであるか、確率変動中であるか、変動時間短縮中であるか、変動表示ゲームのスタート等、変動表示ゲームの進行状態に関連する情報が含まれる。また、エラーが発生したことを示すエラー信号も含まれる。

【0478】

次に、遊技制御装置100は、タイマ割込処理の終了を宣言する(2923)。

【0479】

その後、遊技制御装置100は、スタック領域706に退避していたレジスタを復帰する復帰処理(2924)を行う。ここでは、遊技制御プログラムに記述された命令を実行することにより、S2911のステップにて退避したレジスタの値を復帰させる。次いで、禁止設定されていた割り込みの許可設定をする処理を行う(2925)。

【0480】

そして、タイマ割込処理を終了し、遊技制御装置メイン処理(図25及び図26)に戻る。ここでは、「RET」命令ではなく「RETI」命令を実行して、呼出元の処理に戻る。そのため、スタック領域706に退避されていたフラグレジスタ1200(図12参照)の値も復帰することになり、スタックポインタ1233が示す値も、図19の(D)から図18の(B)の様に変化する。

【0481】

なお、タイマ割込処理の先頭にて、図21に示す「レジスタバンクセクタ」を切り替える命令2115(「LD RBS, 0」又は「LD RBS, 1」)を実行することにより、演算対象として用いるレジスタ群(汎用レジスタ群1220A、1220B)を切り替えることも可能である。例えば、呼出元の処理では汎用レジスタ群1220Aのレジスタを用いる一方で、当該タイマ割込処理では汎用レジスタ群1220Bのレジスタを用いるように切り替えることも可能である。

【0482】

このようなレジスタバンクの切り換えを実行する場合は、タイマ割込処理の中では、呼出元の処理とは異なるレジスタ群のレジスタが使用されるため、S2911のステップにてレジスタを退避させるような処理や、S2911のステップにてレジスタを復帰させるような処理は不要である。

【0483】

10

20

30

40

50

なお、当該タイマ割込処理でも、最後に「RETI」命令を実行するので、スタック領域706に退避されていたフラグレジスタ1200(図12参照)の値が復帰することになり、フラグレジスタ1200のレジスタバンクセレクタ(RBS)1301(図13参照)の値も呼出前の状態に復帰する。そのため、図21に示す「レジスタバンクセレクタ」を切り替える命令2115(「LD RBS, 0」又は「LD RBS, 1」)は、タイマ割込処理の先頭で1回だけ実行すればよい。

【0484】

そして、次のタイマ割り込みが発生するまで初期値乱数更新処理等(図26のステップ2520~2522の処理)を繰り返す。

【0485】

図30は、本発明の第1の実施の形態の遊技制御装置100から、演出制御装置150及び払出制御装置210に初期化指令信号を送信する初期化指令送信処理の手順を示すフローチャートである。本処理は、図25のステップ2511の初期化指令送信処理に対応する。

【0486】

遊技制御装置100は、まず、演出指令及び排出指令の送信を禁止状態に設定する(3001)。具体的には、送信制御レジスタ632(図8)のビット4を“0”に設定して、送信データレジスタ635からの信号の出力を禁止した状態に設定する。

【0487】

次に、遊技制御装置100は、起動時の演出指令を送信データレジスタ635に格納する(3002)。そして、起動時の演出指令がすべて送信データレジスタ635に格納されるまで処理を継続する(3003)。

【0488】

遊技制御装置100は、すべての演出指令が送信データレジスタ635に格納されると(3003の結果が「N」)、起動時の排出指令を送信データレジスタ635に格納する(3004)。そして、起動時の排出指令がすべて送信データレジスタ635に格納されるまで処理を継続する(3005)。

【0489】

最後に、遊技制御装置100は、ステップ3001の処理で禁止状態に設定されていた演出指令及び排出指令の送信を許可状態に設定する(3006)。具体的には、送信制御レジスタ632(図8)のビット4を“1”に設定して、送信データレジスタ635からの信号の出力を許可した状態に設定する。

【0490】

図31は、本発明の第1の実施の形態の遊技制御装置100から、演出制御装置150及び払出制御装置210にコマンドを送信するためのコマンド送信処理の手順を示すフローチャートである。本処理は、図29のステップ2913のコマンド送信処理に対応する。

【0491】

遊技制御装置100は、初期化指令送信処理と同様に送信制御レジスタ632(図8)のビット4を“0”に設定して、まず、演出指令及び排出指令の送信を禁止状態に設定する(3101)。

【0492】

次に、遊技制御装置100は、送信待ちの演出指令が存在するか否か(今回のタイマ割込処理のタイミングで、演出制御装置150に対応する送信データレジスタ635に書き込むべきデータが存在するか否か)を判定する(3102)。送信待ちの演出指令が存在しない場合には(3102の結果が「N」)、ステップ3112以降の排出指令に関する処理を実行する。

【0493】

一方、遊技制御装置100は、送信待ちの演出指令が存在する場合には(3102の結果が「Y」)、送信待ちの演出指令に変動開始の演出指令が含まれているか否かを判定す

10

20

30

40

50

る(3103)。変動開始の演出指令とは、図35にて後述する「停止図柄指定コマンド(MODE=B0H)」と「飾り図柄変動パターン指定コマンド(MODE=B1H~BFH)」の各々に相当するコマンドであり、演出制御装置150は、このコマンドを受信したことを契機に、表示装置8にて変動表示ゲームの実行を開始する。

【0494】

遊技制御装置100は、送信待ちの演出指令に変動開始の演出指令が含まれている場合には(3103の結果が「Y」)、変動開始の演出指令を送信データレジスタに格納する(3104)。そして、送信待ちの演出指令に含まれているすべての変動開始の演出指令が送信データレジスタに格納されるまで処理を継続する(3105)。

【0495】

遊技制御装置100は、送信待ちの演出指令に変動開始の演出指令が含まれていなかった場合(3103の結果が「N」)、又は送信待ちの演出指令に含まれている変動開始の演出指令をすべて送信データレジスタに格納した場合には(3105の結果が「Y」)、送信待ちの演出指令に他の演出指令が含まれているか否かを判定する(3106)。

【0496】

遊技制御装置100は、送信待ちの演出指令に変動開始以外の演出指令が含まれていない場合には(3106の結果が「N」)、ステップ3111の処理を実行し、続いて、排出指令に関する処理を実行する。

【0497】

一方、遊技制御装置100は、送信待ちの演出指令に変動開始以外の演出指令が含まれている場合には(3106の結果が「Y」)、送信バッファ(送信データバッファレジスタ635A)に空きがあるか否かを判定する(3107)。具体的には、送信データステータスレジスタ631(図9)のビット0~5の値(送信データの残量を示す値)が“00h”~“1Fh”であれば、空きがあると判定される。

【0498】

送信バッファに空きがない場合には(3107の結果が「N」)、送信待ちの演出指令を次の送信タイミングに持ち越し(3108)、ステップ3111の処理を実行し、続いて、排出指令に関する処理を実行する。

【0499】

遊技制御装置100は、送信バッファに空きがある場合には(3107の結果が「Y」)、送信データレジスタに変動開始以外の演出指令を格納する(3109)。そして、送信バッファの空きが無くなるか、すべての演出指令が送信データレジスタに格納されるまで、ステップ3107から2110までの処理を継続する(3110)。

【0500】

遊技制御装置100は、送信待ちの演出指令を送信バッファに格納する処理が終了すると、送信制御レジスタ632(図8)のビット4を“1”に設定することで、ステップ3101の処理で禁止状態に設定されていた演出指令の送信を許可状態に設定する(3111)。

【0501】

以上のように、コマンド送信処理において演出指令を演出制御装置150に送信する場合、変動開始の演出指令を優先して送信することによって、遊技制御装置100における変動表示ゲームの進行状態と、演出制御装置150における変動表示ゲームの進行状態との時間差が常に固定されたものとなる。そのため、変動表示ゲームが開始される毎にこの時間差が変化するような不具合を防止できるようになり、遊技制御装置100と演出制御装置150とを同期させながら、表示装置8で実行される変動表示ゲームをより確実に実行させることができる。演出指令の送信が終了すると、続いて、排出指令を排出制御装置210に送信するための処理を実行する。

【0502】

遊技制御装置100は、まず、SW制御領域を検査し、賞球排出対象スイッチ(SW)の立ち上がりがあるか否かを監視する(3112)。賞球を排出する入賞口に遊技球が入

10

20

30

40

50



賞すると、賞球排出対象SWがオンに設定される。そして、遊技制御装置100は、賞球の排出に該当するスイッチが存在するか否かを判定する(3113)。

【0503】

なお、SW制御領域とは、遊技機に備えられた各種スイッチの検出状態を、タイマ割込毎に記憶しておく記憶領域のことである。ここでは、SW制御領域のうち、遊技球検出によって賞球が排出されるスイッチのみが対象とされ、これら対象となったスイッチのうちで、「立ち上がり情報」がオンとなっているものがあるか否かを判定している。

【0504】

遊技制御装置100は、賞球の排出に該当するスイッチが存在する場合(「立ち上がり情報」がオンとなっている賞球排出対象のSWが存在する場合)には(3113の結果が「Y」)、オンとなっている賞球排出対象SWの1つを選択し、選択されたSWに該当する排出指令を送信データレジスタに格納する(3114)。

10

【0505】

次に、その時点で「立ち上がり情報」がオンとなっている賞球排出対象のSWが、他にも存在するかを確認する。遊技制御装置100は、他の賞球排出対象のSWがオンになっていれば、オンとなっている賞球排出対象SWの1つをさらに選択し、選択されたSWに該当する排出指令を送信データレジスタに格納する。そして、すべての賞球を排出する指令が送信データレジスタに格納されるまで処理を継続する(3115)。

【0506】

遊技制御装置100は、賞球の排出に該当するスイッチが存在しない場合には(3113の結果が「N」)、又は賞球の排出に該当するスイッチに対応する排出指令をすべて送信データレジスタに格納した場合には、送信待ちの他の排出指令(払出制御装置210へエラー発生やエラー解除を指令するコマンドなど)が存在するか否かを判定する(3116)。

20

【0507】

遊技制御装置100は、送信待ちの他の排出指令が存在しない場合には(3116の結果が「N」)、排出指令に関する送信を許可状態に設定し(3121)、呼び出し元に戻る。

【0508】

一方、遊技制御装置100は、送信待ちの他の排出指令が存在する場合には(3116の結果が「Y」)、送信バッファに空きがあるか否かを判定する(3117)。具体的には、送信データステータスレジスタ631(図9)のビット0~5の値(送信データの残量を示す値)が“00h”~“1Fh”であれば、空きがあると判定される。

30

【0509】

送信バッファに空きがない場合には(3117の結果が「N」)、送信待ちの排出指令を次の送信タイミングに持ち越し(3118)、排出指令に関する送信を許可状態に設定し(3121)、呼び出し元に戻る。

【0510】

遊技制御装置100は、送信バッファに空きがある場合には(3117の結果が「Y」)、送信待ちの排出指令を送信データレジスタに格納する(3119)。そして、送信バッファの空きが無くなるか、すべての排出指令が送信データレジスタに格納されるまで、ステップ3117から3120までの処理を継続する(3120)。最後に、排出指令に関する送信を許可状態に設定し(3121)、呼び出し元に戻る。

40

【0511】

以上のように、本実施形態では、コマンド送信処理において排出指令を払出制御装置210に送信する場合に賞球排出指令を、その他の排出指令(エラー発生/解除の指令)よりも優先して送信することによって、賞球排出対象の複数のスイッチが、同一のタイマ割込周期内で同時にオンした場合であっても、確実に賞球を排出させるように構成されている。

【0512】

50

このように、演出指令や賞球排出指令などの制御指令を内容に応じて優先して送信して遊技が円滑に進行するように制御し、さらに、優先されなかった制御指令については次回割込発生時に送信することによって、送信漏れのない正確な指令送信を実現することが可能となる。

**【 0 5 1 3 】**

図 3 2 は、本発明の第 1 の実施の形態の電源投入時の遊技制御装置 1 0 0、払出制御装置 2 1 0、及び演出制御装置 1 5 0 が行う処理、並びに、遊技制御装置 1 0 0 に備わるシリアル送信回路 6 1 5 の状態のタイミングチャートである。

**【 0 5 1 4 】**

リセット信号が払出制御装置 2 1 0 に接続されるシリアル送信回路 6 1 5 B 及び演出制御装置 1 5 0 に接続されるシリアル送信回路 6 1 5 A に伝達されると、図 2 3 に示すステップ 2 3 0 2 の処理により、各シリアル送信回路 6 1 5 が不定状態 ( 3 2 0 1 ) から初期状態 ( 3 2 0 2 ) に移行する。

**【 0 5 1 5 】**

この不定状態では、シリアル送信回路 6 1 5 ( シリアル送信回路 6 1 5 A、6 1 5 B ) から出力される信号線のレベルは、ハイレベルであるのかローレベルであるのか保証されない状態である。一方、シリアル送信回路 6 1 5 がリセット信号により初期化されて初期状態に遷移すると、シリアル送信回路 6 1 5 B からの出力信号はオフを示すレベルに確定される。

**【 0 5 1 6 】**

シリアル送信回路 6 1 5 の初期状態は、遊技制御装置 1 0 0 が図 2 5 に示すステップ 2 5 1 1 の処理で初期化指令を送信するために、初期化指令が各シリアル送信回路 6 1 5 に設定されるまで ( 3 2 0 3 ) 継続する。

**【 0 5 1 7 】**

一方、遊技制御装置 1 0 0 のセキュリティ回路 6 3 0 にリセット信号が伝達されると、図 2 3 に示すステップ 2 3 0 4 の処理で自己診断処理を実行し、ステップ 2 3 0 5 の処理でセキュリティチェック処理を実行する ( 3 2 0 4 )。セキュリティチェック処理の実行後に CPU 1 0 2 が起動し、CPU 1 0 2 によって遊技制御装置メイン処理 ( 図 2 5 及び図 2 6 ) が実行される。

**【 0 5 1 8 】**

CPU 1 0 2 は、ディレイ処理の実行 ( 3 2 0 6 ) 前に 1 回目の RAM クリア信号の取り込み ( 3 2 0 5 ) と、ディレイ処理の実行後に 2 回目の RAM クリア信号の取り込み ( 3 2 0 7 ) と、を行う。言い換えると、1 回目の RAM クリア信号取り込み ( 3 2 0 5 ) と 2 回目の RAM クリア信号取り込み ( 3 2 0 7 ) とは、ディレイ処理 ( 3 2 0 6 ) を挟んで実行される。

**【 0 5 1 9 】**

このように、2 2 0 5 及び 2 2 0 7 の各時点で実行される RAM クリア信号取り込みの間に、ディレイ処理を実行するので、ディレイ処理の間に、1 回目の RAM クリア信号取り込みで取り込んだチャタリング除去等を行うことができる。

**【 0 5 2 0 】**

ディレイ処理 ( 3 2 0 6 ) で処理を待機させた後に、図 2 5 に示すステップ 2 5 1 6 及び 1 5 1 0 の処理で RAM 1 0 4 の初期化処理を行い ( 3 2 0 8 )、ステップ 2 5 1 1 の処理で初期化指令を送信してから、通常の遊技制御を行う ( 3 2 0 9 )。

**【 0 5 2 1 】**

なお、通常の遊技制御を実行すると、遊技状態に応じて、払出制御指令を払出制御装置 2 1 0 に送信するために、払出制御指令が払出制御装置 2 1 0 に接続されるシリアル送信回路 6 1 5 B に設定される ( 3 2 1 0 )。また、通常の遊技制御の実行中には、遊技状態に応じて、演出制御指令を演出制御装置 1 5 0 に送信するために、演出制御指令が演出制御装置 1 5 0 に接続されるシリアル送信回路 6 1 5 A に設定される ( 3 2 1 1 )。

**【 0 5 2 2 】**

一方で、払出制御装置 210 のセキュリティ回路にリセット信号が伝達されると、払出制御装置 210 のセキュリティ回路は、図 23 に示すステップ 2314 の処理で自己診断処理を実行し、ステップ 2315 の処理でセキュリティチェック処理を実行する(3212)。セキュリティチェック処理の実行後に CPU 212 が起動し、CPU 212 によって、図 14 のステップ 2316 の処理で電源投入時の初期化処理を実行する(3213)。払出制御装置 210 の初期化処理が実行されると、払出制御装置 210 のシリアル受信回路 625 を、遊技制御装置 100 からの指令を受信可能な状態にする(3214)。

#### 【0523】

また、演出制御装置 150 にリセット信号が伝達されると、演出制御装置 150 は、図 23 のステップ 2323 の処理で電源投入時の初期化処理を実行する(3215)。演出制御装置 150 の初期化処理が実行されると、演出制御装置 150 のシリアル受信回路 625 を、遊技制御装置 100 からの指令を受信可能な状態にする(3216)。

10

#### 【0524】

遊技制御装置 100 は、ディレイ処理を実行することで、RAM 104 の初期化処理の実行開始のタイミングを遅延させている。言い換えると、ディレイ処理によって、演出制御装置 150 や払出制御装置 210 へ初期化指令を送信するタイミングを遅延させている。

#### 【0525】

このため、ディレイ処理によって、払出制御装置 210 に接続されるシリアル送信回路 615B 及び演出制御装置 150 に接続されるシリアル送信回路 615A が初期状態を維持する時間を十分に確保し、その間に、払出制御装置 210 及び演出制御装置 150 は、初期化処理を実行し、自身のシリアル受信回路 625 を介して遊技制御装置 100 からの指令を受信可能な状態にすることができる。

20

#### 【0526】

したがって、ディレイ処理を設けることで、図 25 のように、リセット信号が、遊技制御装置 100、払出制御装置 210 及び演出制御装置 150 に同時に伝達される構成の遊技機であっても、ハードウェア等で構成した遅延回路を設けることなく、各制御装置が起動を開始するタイミングを適切に設定することができる。

#### 【0527】

よって、図 32 のように、まず、払出制御装置 210 に接続されるシリアル送信回路 615B 及び演出制御装置 150 に接続されるシリアル送信回路 615A が初期状態に維持され、その状態で、払出制御装置 210 及び演出制御装置 150 のシリアル受信回路 625 が指令受信可能状態になり、次いで、払出制御装置 210 及び演出制御装置 150 に初期化指令を送信させることを確実に実行できるようになる。

30

#### 【0528】

仮に、遊技機 1 への電源投入直後において、遊技制御装置 100 の払出制御装置 210 に接続されるシリアル送信回路 615B 及び演出制御装置 150 に接続されるシリアル送信回路 615A が初期状態に維持される以前に、払出制御装置 210 若しくは演出制御装置 150 のシリアル受信回路 625 が指令受信可能状態になると、払出制御装置 210 に接続されるシリアル送信回路 615B 及び演出制御装置 150 に接続されるシリアル送信回路 615A から出力される信号レベルが不安定であるから、払出制御装置 210 若しくは演出制御装置 150 にてこの不安定な信号レベルの情報を、正規な信号であると誤って受信するおそれがあり、誤作動を引き起こす可能性がある。

40

#### 【0529】

また、払出制御装置 210 若しくは演出制御装置 150 のシリアル受信回路 625 が指令受信可能状態になる前に、遊技制御装置 100 から、払出制御装置 210 若しくは演出制御装置 150 へ初期化指令を送信してしまうと、払出制御装置 210 や演出制御装置 150 で初期化指令を受信できなくなり、誤作動を引き起こす可能性がある。

#### 【0530】

特に、本実施形態の遊技機のように、遊技制御装置 100 から払出制御装置 210 へ単

50

方向で指令を送信する構成や、遊技制御装置 100 から演出制御装置 150 へ単方向で指令を送信する構成の場合には、指令された情報が正しく送信されているか否かを確認することが困難であるため、初期化時に処理を遅延させることが有効である。

【0531】

また、図 32 では、RAM クリア信号の取り込みが 2 回である例を示したが、複数回であればよい。この複数回の中にディレイ処理を実行することによって、ディレイ処理実行直前の RAM クリア信号取り込みのチャタリング除去等にかかる時間をディレイ処理による遅延時間と重複させることができるので、処理を効率化させることができる。

【0532】

図 33 は、本発明の第 1 の実施の形態の遊技制御装置 100 から払出制御装置 210 に送信される排出指令の一例を示す図である。

10

【0533】

払出制御装置 210 に送信される排出指令は、払出制御装置 210 を初期化する指令（初期化指令信号）と、賞球を排出する指令（排出指令信号）と、エラーの発生及び解除を通知する指令（エラー通知信号）があり、モード部及びアクション部からなる共通のフォーマットで送信される。

【0534】

まず、初期化指令信号について説明すると、初期化指令信号は、モード部が「80H」であり、アクション部は「00H～7FH」のいずれかの値となる。初期化指令信号のアクション部は、払出制御装置 210 に設定されている認証コードに対応する値（「00H～7FH」のいずれかの値）となる。この払出制御装置 210 に設定されている認証コードに対応する値は、例えば、RAM 104 に設定されているものとする。なお、モード部は「80H」以上の値、アクション部には「80H」未満の値が設定されており、モード部を受信する場合に「80H」未満の値が送信された場合には不正なコマンドが送信されてきたことを判別することが可能となっている。

20

【0535】

初期化指令信号の出力時期は、遊技制御装置 100 に電源投入時であり、具体的には、図 25 に示すステップ 2511 の処理である。

【0536】

次に、排出指令信号について説明する。払出制御装置 210 によって払い出される遊技媒体の個数に対応して、15 個の排出指令信号が用意されている。

30

【0537】

排出指令信号のモード部は「A1H～AFH」である。なお、このモード部の下位バイトは、排出指令信号が払い出しを指令する遊技媒体の個数と一致する。また、排出指令信号のアクション部は「5EH～50H」となる。このアクション部は、モード部の各ビットの論理を反転した値となっている。

【0538】

例えば、1 個の遊技媒体の払い出しを指令する排出指令信号のモード部は「A1H」であり、アクション部は「5EH」である。すなわち、排出指令信号は、モード部とアクション部とからなる 2 バイトのデータで構成されている。

40

【0539】

なお、排出指令信号の出力時期は、一般入賞口 32、第 1 始動入賞口 37、第 2 始動入賞口 34、特別変動入賞装置（大入賞口）36 に遊技球が入賞したタイミングで出力される。

【0540】

また、払出制御装置 210 は、排出指令信号を受信すると、受信した排出指令信号のモード部の各ビットの論理を反転した値が、アクション部の各ビットの値と一致しなければ、受信した排出指令信号に対応する個数の遊技媒体の払い出しを許可しない。

【0541】

最後に、エラー通知信号について説明する。排出指令がエラー発生通知の場合には、モ

50

ード部にエラーが発生したことを示す「C0H」が設定される。また、エラー通知信号のアクション部は、発生したエラーの種類に対応する値（「00H～7FH」のいずれかの値）が設定される。

【0542】

排出指令がエラー解除通知の場合には、モード部にエラーが解除されたことを示す「C1H」が設定される。また、エラー解除信号のアクション部は、エラー通知信号の場合と同様に、発生したエラーの種類に対応する値（「00H～7FH」のいずれかの値）が設定される。

【0543】

図34は、本発明の第1の実施の形態のスイッチの立ち上がりを検出する手順を示すタイムチャートである。なお、図中のfは割込周期であり、割込周期の先頭でタイマ割込が発生する。また、dは遅延時間を示す。スイッチの立ち上がり、すなわち、スイッチがオンになったか否かの判定は、図29に示したタイマ割込処理のステップ2912の入出力処理で行われる。なお、以下に示す、第1物理レベル、第2物理レベル、論理レベル、立上り情報は、タイマ割込が発生する毎に更新され、遊技機に備えられたスイッチ毎に整理されてSW制御領域（図31で前述）に記憶される。

10

【0544】

まず、スイッチの立ち上がり時（3401）及び立ち下がり時（3402）について説明する。CPU102は、タイマ割込発生時の入出力処理（図29のステップ2912）において、スイッチの検出信号のレベルが、前回設定された第1物理レベルと相違すると（ローレベルからハイレベル、又は、ハイレベルからローレベルに変化）、第1物理レベルを検出信号のレベルに新たに設定する。そして、所定の遅延時間が経過した後、スイッチの検出信号が、前回設定された第2物理レベルと相違する場合には、第2物理レベルを検出信号のレベルに設定する。なお、第1物理レベルと第2物理レベルとは、検出タイミングが異なるだけである。

20

【0545】

このとき、第1物理レベルと、第2物理レベルとが一致していれば、検出信号が変化したものと判断し、論理レベルに当該レベルを設定する。そして、論理レベルがローレベルからハイレベルに変化した場合には、次のタイマ割込発生から遅延時間が経過するまで、立ち上がり情報をオンに設定する。すなわち、立ち上がり情報をオンに設定した後、次のタイマ割込発生でオフに設定する。

30

【0546】

また、本発明の第1の実施の形態では、第1物理レベルと第2物理レベルの信号レベルが相違する場合、すなわち、3403に示すように、第1物理レベル検出時と、第2物理レベル検出時とで、スイッチの検出信号のレベルが相違する場合には、ノイズが発生したものと、論理レベルを変更しないように構成されている。このように構成することによって、ノイズ発生時に誤って立上り情報がオンに設定されることを防ぎ、後述するように、賞球排出指令が誤って払出制御装置210に送信されることを防ぐことができる。

【0547】

また、本発明の第1の実施の形態では、図34のタイミングチャートに示した立上り情報がオンとなったスイッチの中に賞球排出対象となるものが含まれている場合は、送信バッファ（図6の送信データバッファレジスタ635A）に賞球排出指令が格納されることで、遊技制御装置100から払出制御装置210に賞球排出指令が送信される。また、賞球排出指令は、賞球排出の対象となる入賞スイッチ（図4における、第1特図始動SW37A、第2特図始動SW34A、カウントSW36A、及び入賞口SW32A～入賞口SW32N）ごとに定義される。なお、本発明の第1の実施の形態では、賞球排出の対象となる入賞スイッチが16個備えられている。

40

【0548】

このとき、1回のタイマ割込周期内での送信で、送信バッファ内に格納されたすべての賞球排出指令が送信されないと、次のタイマ割込発生時に、賞球排出対象のスイッチの

50

立上り情報がオンとなって新たに発生した賞球排出指令を、送信バッファに取り込めないおそれがある。これを防止するには、新たに賞球排出指令が発生する度に、送信バッファに空きがあるか否かを確認して、空きがなければ次回送信時まで賞球排出指令を保持していなければならない、送信できない賞球排出指令を保持するための記憶領域（例えば、前述したSW制御領域の立ち上がり情報を一時的に退避させる領域など）を必要としてしまう。また、送信バッファに格納できなかった賞球排出指令を退避させる処理も必要となってしまう。

【0549】

例えば、賞球排出対象のスイッチが5個であり、賞球排出指令のサイズが2バイトであるならば、払出制御装置210に指令を送信するためのバッファ（送信データレジスタ635）に格納できるデータの最大バイト数を10バイトとしておけば、同一タイム割込周期内で賞球排出対象の全てのスイッチが同時にオンしたとしても、全ての賞球排出指令をバッファに取り込むことができる。しかしながら、賞球排出対象のスイッチが5個を超えた場合には、バッファにはより多くの容量を必要とすることになる。

10

【0550】

本発明の第1の実施の形態では、送信バッファの容量を32バイトに設定しているので、賞球排出対象のスイッチが16個以下であれば、1回の割り込み発生時にすべての賞球排出指令を遊技制御装置100から払出制御装置210に送信できるように構成されている。したがって、送信されなかった賞球排出指令を保持するための記憶領域を必要とせず、また、送信されていない賞球排出指令を退避させる処理も不要となるため、必要な記憶容量を削減し、遊技制御装置100の制御プログラムを簡略化することができる。

20

【0551】

図35は、本発明の第1の実施の形態の遊技制御装置100から演出制御装置150に送信される演出制御コマンドの一例を示す図である。

【0552】

演出制御装置150に送信されるコマンドは、初期化コマンドと通常時のコマンドである演出コマンドとがあり、これらのモード部及びアクション部によって構成される共通のフォーマットで送信される。

【0553】

まず、初期化コマンドについて説明する。

30

【0554】

初期化コマンドには、RAM104のすべての領域が初期化されたか否かを示す電源投入コマンドと、遊技機1のシリーズを特定するためのシリーズ機特定コマンドとが含まれる。また、直前の電源遮断時における遊技機1の遊技状態（低確率状態、高確率状態、入賞抑制状態、入賞促進状態）を通知するコマンドや直前の電源遮断時における特別図柄入賞記憶の数を通知するコマンドも初期化コマンドに含まれる。

【0555】

図35に示すように、RAM104のすべての領域が初期化されたことを示す電源投入コマンドAのモード部は「80H」であり、アクション部は「01H」である。RAM104のすべての領域が初期化されたこととは、図25に示すステップ2510の処理が実行されたことである。

40

【0556】

一方、RAM104のすべての領域が初期化されていないこと、つまり、RAM104の一部の領域が初期化されたことを示す電源投入コマンドBのモード部は「80H」であり、アクション部は「02H」である。RAM104のすべての領域が初期化されていないこと、つまり、RAM104の一部の領域が初期化されたこととは、図25に示すステップ2516の処理が実行されたことである。

【0557】

したがって、図25に示すステップ2510の処理が実行された場合には、ステップ2511の処理で、モード部が「80H」でアクション部が「01H」である電源投入コマ

50

ンド A が送信される。図 25 に示すステップ 2516 の処理が実行された場合には、ステップ 2511 の処理で、モード部が「80H」でアクション部が「02H」である電源投入コマンド A が送信される。

【0558】

演出制御装置 150 は、RAM 104 のすべての領域が初期化されたことを示す電源投入コマンド A を受信すると、RAM 104 のすべての領域が初期化されたことを表示装置 8 に表示する。

【0559】

また、演出制御装置 150 は、RAM 104 のすべての領域が初期化されていないことを示す電源投入コマンド B を受信すると、RAM 104 のすべての領域が初期化されていないことを表示装置 8 に表示する。

10

【0560】

また、シリーズ機特定コマンドのモード部は「81H」であり、アクション部は「01H～7FH」である。アクション部は、遊技機 1 のシリーズに対応する「01H」～「7FH」のいずれかの値である。なお、遊技機 1 のシリーズに対応する値は、ROM 103 に設定されている。

【0561】

また、遊技状態（低確率状態、高確率状態、入賞抑制状態、入賞促進状態）を通知するコマンドは、モード部が「90H」となっており、アクション部には、直前の電源遮断時における遊技状態別に対応付けられた値が格納される。例えば、低確率状態であればアクション部は「01H」であり、高確率状態であればアクション部は「02H」となる。演出制御装置 150 は、遊技状態を通知するコマンドを受信すると、遊技状態を報知するための演出を行う。

20

【0562】

また、特別図柄入賞記憶の数を通知する保留情報コマンドは、モード部が「A0H」となっており、アクション部は「00H～04H」のいずれかの値である。アクション部は、直前の電源遮断時における始動記憶数（0～4）に対応した値である。演出制御装置 150 は、保留情報コマンドを受信すると、表示装置 8 の記憶表示部（図 49 で後述する第 1 記憶表示部 4920 又は第 2 記憶表示部 4930）部に、受信した保留情報コマンドに対応する始動記憶数を表示する。

30

【0563】

これらのシリーズ機特定信号、遊技状態を通知するコマンド、及び特別図柄入賞記憶の数を通知するコマンドの出力時期は、電源投入時であり、図 25 に示すステップ 2511 の処理で送信される。なお、これらの各信号と電源投入コマンドの出力順序は、いずれが先であっても後であってもよい。さらに、電源投入時に、遊技制御装置 100 から演出制御装置 150 へ通知すべき情報が他にもあれば、初期化コマンドと一緒に送信してもよい。

【0564】

次に、各演出コマンドについて説明する。

【0565】

まず、表示装置 8 で実行される変動表示ゲームにおいて、始動口に遊技球が入賞したタイミングで出力され、図柄の変動開始前に実行される演出を指示する変動前演出指定コマンドについて説明する。

40

【0566】

変動前演出指定コマンドのモード部は「A1H～AFH」であり、アクション部は「01H～7FH」のいずれかの値である。モード部は前半変動パターン番号に対応し、アクション部は後半変動パターン番号に対応する。

【0567】

演出制御装置 150 は、変動前演出指定コマンドを受信すると、表示装置 8 において識別図柄の変動開始前の演出、例えば、先読み予告としてキャラクタが登場するなどの演出

50

が実行され、その後、図柄の変動表示を開始し、変動表示ゲームを開始する。

【0568】

表示装置8で実行される変動表示ゲームにおいて図柄の変動パターンを指示する飾り図柄変動パターン指定コマンドについて説明する。

【0569】

飾り図柄変動パターン指定コマンドのモード部は「B1H」であり、アクション部は「01H～7FH」のいずれかの値である。モード部は前半変動パターン番号に対応し、アクション部は後半変動パターン番号に対応する。

【0570】

演出制御装置150は、飾り図柄変動パターン指定コマンドを受信すると、表示装置8において図柄の変動表示を開始し、変動表示ゲームを開始する。

10

【0571】

飾り図柄変動パターン指定コマンドは、表示装置8において変動表示ゲームの図柄の変動表示を開始するタイミングで送信する。具体的には、表示装置8で変動表示ゲームが終了した場合に始動記憶がある場合、又は表示装置8で変動表示ゲームが実行されていない場合に始動入賞口に遊技球が入賞した場合である。

【0572】

次に、表示装置8における変動表示ゲームにおける停止図柄を指定する停止図柄指定コマンドについて説明する。

【0573】

20

停止図柄指定コマンドのモード部は「B0H」であり、アクション部は「01H～7FH」のいずれかの値である。アクション部は、停止図柄に対応する値である。

【0574】

演出制御装置150は、停止図柄指定コマンドを受信すると、受信した停止図柄指定コマンドに基づいて、表示装置8における変動表示ゲームの停止図柄を特定する。

【0575】

停止図柄指定コマンドは、表示装置8の変動表示ゲームの変動表示を開始するときに送信される。

【0576】

続いて、変動時間が経過し、変動表示中の図柄を停止するための図柄変動停止コマンドについて説明する。

30

【0577】

図柄変動停止コマンドのモード部は「C0H」であり、アクション部は「01H」である。

【0578】

演出制御装置150は、図柄変動停止コマンドを受信すると、表示装置8で変動表示している図柄を停止させる。

【0579】

図柄変動停止コマンドは、変動時間が経過したタイミングで送信される。

【0580】

40

続いて、特別遊技状態発生中に送信される大当たり関連コマンドについて説明する。

【0581】

大当たり関連演出指令信号のモード部は「D0H」であり、アクション部は「01H～7FH」のいずれかの値である。アクション部は、特別遊技状態の進行状況に応じた値である。

【0582】

演出制御装置150は、大当たり関連コマンドを受信すると、受信した大当たり関連コマンドに基づいて、特別遊技状態に関連する演出を行う。

【0583】

遊技機1においてエラーが発生した場合にエラーの発生を報知するためのエラー関連コ

50



マンドについて説明する。

【0584】

エラー関連コマンドのモード部は「70H」であり、アクション部は「01H～7FH」のいずれかの値である。アクション部は発生したエラーに対応した値である。

【0585】

演出制御装置150は、エラー関連コマンドを受信すると、エラー関連コマンドに基づいて、発生したエラーを報知するための演出を行う。

【0586】

エラー関連コマンドは、遊技制御装置100がエラーを検出したタイミングで送信される。

【0587】

なお、前述の遊技状態を通知するコマンド(モード部＝「90H」)は、電源投入時だけでなく、通常の遊技中において遊技状態が変化した場合にも送信される。例えば、遊技中において低確率状態が発生したときに、モード部＝「90H」かつアクション部＝「01H」のコマンドが送信され、遊技中において、高確率状態が発生したときに、モード部＝「90H」かつアクション部＝「02H」のコマンドが送信される。

【0588】

また、前述の特別図柄入賞記憶の数を通知するコマンド(モード部＝「A0H」)は、電源投入時だけでなく、通常の遊技中において始動入賞口に遊技球が入賞して始動記憶数が増加した場合にも送信される。例えば、遊技中において始動入賞口に遊技球が入賞して始動記憶数が「3」に変化したときには、モード部＝「A0H」かつアクション部＝「03H」の信号が送信される。

【0589】

したがって、これらの遊技状態を通知する信号、及び特別図柄入賞記憶の数を通知する信号は、演出コマンドとしても機能することになる。

【0590】

なお、前述したように、これらの信号のうち、変動前演出指定コマンドは、他の信号よりも優先して演出制御装置150へのデータ送信を行うためのバッファ(送信データレジスタ635)に取り込まれる。これにより、遊技制御装置100における変動表示ゲームの進行状態と、演出制御装置150における変動表示ゲームの進行状態との時間差を常に固定させる。

【0591】

図36は、本発明の第1の実施の形態の遊技制御装置100から演出制御装置150に送信される送信データの構成を示す説明図である。図36(a)は送信データの概略構成を示し、図36(b)は送信データの詳細構成を示している。なお、遊技制御装置100から演出制御装置210に送信される送信データの構成も同様である。

【0592】

図36(a)に示すように、1組の制御指令データは、コマンドの分類を識別するためのモードデータDCmと、実行されるコマンドの内容(機能)を示すアクションデータDCaで構成される。これらのモードデータDCm及びアクションデータDCaは、図33の賞球排出指令のMODE及びACTION、図35の演出制御コマンドのMODE及びACTIONに各々対応する。

【0593】

また、本実施形態では、送信バッファに設定されたすべての制御指令データ(最大32バイト)を1回の割込周期ですべて送信可能となるようにデータの送信速度が設定されている。したがって、遊技制御装置100の制御プログラムを複雑化させずに、演出制御装置150への指令送信を一時的に中断するなどの処理を必要とせず、また、指令送信の遅れなどによって、遊技制御装置100と演出制御装置150の各制御の進行状態がずれないようにすることが可能となる。

【0594】

10

20

30

40

50

また、図36(b)に示すように、1組の制御指令データを構成するモードデータDCm及びアクションデータDCaは、8ビットのコマンドデータに、1フレームの開始であることを示すスタートビット、1ビットのパリティデータ、1フレームの終わりであることを示すストップビットが付加され、1フレームを処理単位として送信される。従って、この1フレームの時間が、1バイトあたりのデータ送信時間(送信バッファに格納された制御指令データを1バイト送信するために必要な時間)となる。

【0595】

すなわち、本実施形態では、非同期方式(調歩同期式)によるシリアル通信を利用して制御指令データを送信する。

【0596】

演出制御装置150は、受信した1組の制御指令データ(モードデータDCm+アクションデータDCa)を解析し、表示装置8、音回路156、装飾制御装置等を制御するための制御データを生成する。

【0597】

本実施形態では、1回の割込処理で生成された一連の制御指令データを送信するときに、全ての制御指令データを、タイマ割込周期内で一時に送信するようにしている。すなわち、一連の制御指令データを全て送信データレジスタ635に格納した後で、シリアル送信回路615を送信許可状態として(送信制御レジスタ632の送信イネーブルを送信許可に設定)送信を開始する。

【0598】

従属制御装置としての演出制御装置150は、連続して受信したデータを、一連の制御指令データとして認識する。そして、この一連の制御指令データを処理単位として演出装置(例えば、表示装置8)を制御する。

【0599】

これにより、演出制御装置150は、タイマ割込処理毎に生成された一連の制御指令データを特定でき、1回のタイマ割込処理で生成された一連の制御指令データを処理単位として演出装置を制御するので、一連の制御指令データに基づく演出を連続して(タイムラグなく)実行することができる。

【0600】

次に、図37を参照して、図29のタイマ割込処理における特図ゲーム処理(2918)の詳細を説明する。図37は、本発明の第1の実施の形態の特図ゲーム処理の手順を示すフローチャートである。

【0601】

遊技制御装置100は、第1始動入賞口37(図3参照)及び普通変動入賞装置33(図3参照)の第2始動入賞口への遊技球の入賞を監視し、入賞に基づき各種乱数値の記憶を行う始動口SW監視処理を実行する(3700)。なお、始動口SW監視処理についての詳細は、図39にて後述する。

【0602】

遊技制御装置100は、カウントSW36A(図3参照)からの入力を監視するカウントスイッチ監視処理を行う(3701)。

【0603】

遊技制御装置100は、ステップ3706~3712の各処理で設定される特図ゲーム処理タイマが0であるか否かを判定する(3702)。

【0604】

特図ゲーム処理タイマが0でないと判定された場合には(3702の結果が「N」)、遊技制御装置100は、特図ゲーム処理タイマを-1更新し(3703)、更新された特図ゲーム処理タイマがタイムアップして0になったか否かを判定する(3704)。特図ゲーム処理タイマがタイムアップしていない場合には(3704の結果が「N」)、遊技制御装置100はステップ3714の処理を実行する。

【0605】

10

20

30

40

50

一方、特図ゲーム処理タイマが0であると判定された場合には(3702の結果が「Y」)、又は特図ゲーム処理タイマがタイムアップしたと判定された場合には(3704の結果が「Y」)、遊技制御装置100は、ゲーム処理番号を取得し、取得したゲーム処理番号に基づいて実行する処理を選択するゲーム分岐処理を行う(3705)。

#### 【0606】

ゲーム処理番号が0である場合には、遊技制御装置100は、特図変動表示ゲームの変動開始を監視し、特図変動表示ゲームの変動開始の設定や演出の設定、特図変動中処理を行うために必要な情報の設定を行う特図普段処理を実行する(3706)。特図普段処理では、特図変動表示ゲームの変動に係る情報の設定を行うようになっており、始動記憶に基づく特図変動表示ゲームの実行に伴って始動記憶を1デクリメントする処理や特図変動表示ゲームにおける変動時間の設定処理等を行う。なお、特図普段処理についての詳細は、図45にて後述する。

10

#### 【0607】

ゲーム処理番号が1である場合には、遊技制御装置100は、特図表示中処理(3708)を行うために必要な情報(特図変動表示ゲームにおける結果の停止表示時間等)の設定を行う特図変動中処理を実行する(3707)。

#### 【0608】

ゲーム処理番号が2である場合には、遊技制御装置100は、特図表示中処理を実行する(3708)。特図表示中処理では、特図変動表示ゲームの結果が大当たりである場合にファンファーレ・インターバル処理(3709)を行うために必要な情報が設定され、特図変動表示ゲームの結果がはずれである場合には特図普段処理(3706)を行うために必要な情報が設定される。

20

#### 【0609】

ゲーム処理番号が3である場合には、遊技制御装置100は、特別遊技状態における大入賞口の開放回数の更新等の処理を行うファンファーレ・インターバル処理を実行する(3709)。なお、この処理において、ラウンド開始コマンドが設定される。

#### 【0610】

ゲーム処理番号が4である場合には、遊技制御装置100は大入賞口開放中処理を行う(3710)。大入賞口開放中処理では、特別遊技状態が最終ラウンドでなければファンファーレ・インターバル処理(3709)を行うために必要な情報(インターバルコマンドを含む)が設定され、特別遊技状態が最終ラウンドであれば特別遊技状態の終了画面のコマンド(大当たり終了コマンド)の設定や大入賞口残存球処理(3711)を行うために必要な情報が設定される。

30

#### 【0611】

ゲーム処理番号が5である場合には、遊技制御装置100は大入賞口残存球処理を行う(3711)。大入賞口残存球処理では、特別遊技状態が最終ラウンドである場合に、大入賞口を閉鎖した後大入賞口内に残存する全ての遊技球がカウンタSW42cで検出されるまでの時間が設定される。

#### 【0612】

ゲーム処理番号が6である場合には、遊技制御装置100は大当たり終了処理を実行する(3712)。大当たり終了処理では、特別遊技状態を終了する処理が行われ、特図普段処理(3706)を行うために必要な情報が設定される。

40

#### 【0613】

ゲーム処理番号に基づくステップ3706~3712の処理を行った後、遊技制御装置100は、設定された各種データをセーブするテーブルデータセーブ処理を実行する(3713)。その後、遊技制御装置100は、特図1変動表示ゲームのゲーム処理タイマの更新等を行う特図1変動制御処理(3714)を実行し、特図2変動表示ゲームのゲーム処理タイマの更新等を行う特図2変動制御処理(3715)を実行して、特図ゲーム処理を終了する。

#### 【0614】

50

図38は、本発明の第1の実施の形態の特図ゲーム処理におけるゲーム処理番号に基づいた分岐の処理を説明する図である。この分岐の処理は、図37のステップ3705で実行され、行3801～行3807の命令が順に実行されることになる。ここでは、図12で前述した各種レジスタ、図21及び図22で前述したアセンブリ言語を用いて説明を行うことにする。

【0615】

なお、図38における列3821は、遊技制御プログラムが格納されているユーザプログラムROM103のアドレスを示し、列3822は、当該アドレスに格納されているデータを示している。例えば、ユーザプログラムROM103の「4800H」のアドレスには「F5H」のデータが格納され、続く「4801H」のアドレスには「0CH」のデータが格納され、続く「4802H」のアドレスには「48H」のデータが格納され、続く「4803H」のアドレスには「3AH」のデータが格納されている。以降の「4804H」～「480BH」のアドレスには、「10H」、「28H」、「87H」、「09H」、「E9H」のデータが格納されている。

10

【0616】

さらに、「480CH」と「480DH」のアドレスには「20H」と「48H」の値が格納されている。この2バイトのデータを16ビットのデータ(ただし、下位アドレス側と上位アドレス側を入れ替える)として表現したデータ「4820H」は、図37のステップ3706で実行される特図普段処理の先頭アドレスに相当する。同様に、「480EH」と「480FH」のアドレスに格納されるデータを16ビットで表現した「4840H」は、図37のステップ3707で実行される特図変動中処理の先頭アドレスに相当する。

20

【0617】

以下、「4810H」以降に格納される16ビットの各データ「4860H」、「4880H」、「48A0H」、は、図37のステップ3708、3709、3710、で実行される、特図表示中処理、ファンファーレ・インターバル処理、大入賞口開放処理、のそれぞれの先頭アドレスに相当する。このようにして、アドレス値480CH～4819Hの領域では、各処理の先頭アドレスが処理番号順に格納されることにより、所定の分岐アドレステーブルを構成している。

30

【0618】

行3801の命令は、HLレジスタ1212Aに、前述の分岐アドレステーブルの先頭となるアドレス値「480CH」をロードするものである。次に、行3802の命令が実行され、Aレジスタ1202Aにゲーム処理番号(図37の3705参照)がロードされる。次に、行3803の命令が実行され、Aレジスタ1202Aの値が2倍される。

【0619】

次に、行3804によりAレジスタ1202Aの値がCレジスタ1205Aにロードされ、行3805によりBレジスタ1204Aに「0」の値がロードされることで、BCレジスタ1206AにAレジスタ1202Aの値が設定される。

【0620】

次に、行3806によりHLレジスタ1212AにBCレジスタ1206Aの値が加算される。これにより、HLレジスタ1212Aには、ゲーム処理番号に対応する処理の分岐先のアドレスが設定される。最後に、行3807の「JP (HL)」命令を実行することで、HLレジスタ1212Aが示すアドレスに、処理を移行させる。

40

【0621】

ここで、行3801～行3807の各命令と、遊技制御プログラムが格納されているユーザプログラムROM103に記憶される列3722の各データとの関係について説明する。

【0622】

遊技制御プログラムを記述するアセンブリ言語の命令は、図21及び図22で前述したコードに変換されて、ユーザプログラムROM103に記憶されている。例えば、行38

50

01の「LD HL, 480CH」の命令は、「4800H」～「4803H」のアドレスに配置される「F5H」、「0CH」、「48H」の値に変換されて、ユーザプログラムROM103に記憶されている。行3802～行3807の各命令も、対応するコードに変換されて、ユーザプログラムROM103の「4803H」～「480BH」のアドレスに、順に記憶されている。

**【0623】**

なお、行3808以降の記述は、特図普段処理、特図変動中処理、特図表示中処理、ファンファーレ・インターバル処理、大入賞口開放処理、 の先頭アドレスの値をプログラム中に配置することを示しており、ユーザプログラムROM103の「480CH」以降のアドレスに、対応するアドレスデータが設定されることによって、前述の分岐アドレステーブルが構成されることになる。

10

**【0624】**

このように、HLレジスタに所定の分岐アドレステーブルの先頭アドレス（「480CH」）を設定し、処理番号（SEQNO）に応じてHLレジスタの値を更新することによって、処理番号に応じた処理を実行することができる。したがって、HLレジスタは、分岐アドレス設定レジスタとして機能する。

**【0625】**

なお、「JP (HL)」以外に、特定のレジスタが示す値をアドレスと見なして分岐する命令が存在すれば、そのレジスタは分岐アドレス設定レジスタとして機能することになる。例えば、図21には図示していないが、IXレジスタ1231が示す値のアドレスに処理を分岐させる命令「JP (IX)」を用いていれば、IXレジスタ1231は分岐アドレス設定レジスタとして機能していることとなる。

20

**【0626】**

また、「JP (HL)」命令を用いなくても、HLレジスタが示す値をアドレスと見なして、そのアドレスに処理を分岐させるような方法は、他にもある。例えば、「PUSH HL」命令を実行して、一旦スタック領域に706にHLレジスタの値を格納し、直後に「RET」命令を実行すれば、スタック領域に706に格納してあったHLレジスタの値がプログラムカウンタ1234に格納される。その結果、HLレジスタが示す値のアドレスに処理を分岐させることができる。

**【0627】**

このような方法を用いるのであれば、HLレジスタ1212A以外のレジスタを分岐アドレス設定レジスタとして機能させることも可能である。例えば、DEレジスタ1209Aをスタック領域に706に格納した直後に「RET」命令を実行すれば、DEレジスタ1209Aが示す値のアドレスに分岐するので、この場合はDEレジスタ1209Aが分岐アドレス設定レジスタとして機能することになる。

30

**【0628】**

次に、図39を参照して、図37の特図ゲーム処理における始動口SW監視処理（3700）の詳細を説明する。図39は、本発明の第1の実施の形態の始動口SW監視処理の手順を示すフローチャートである。

**【0629】**

始動口SW監視処理では、遊技制御装置100は、まず第1始動入賞があるか否かの判定を行う（3900）。第1始動入賞口37への入賞があるか否かは、第1始動入賞口37への入賞があった場合に第1特図始動SW37Aから出力される遊技球の検出信号の有無によって判定される。第1始動入賞がない場合には（3900の結果が「N」）、遊技制御装置100はステップ3901の処理を実行する。これに対して、第1始動入賞がある場合には（3900の結果が「Y」）、遊技制御装置100は特図始動口SW共通処理（3904）を実行し、始動口SW監視処理を終了する。なお、特図始動口SW共通処理についての詳細は図40にて後述する。

40

**【0630】**

遊技制御装置100は、普電作動中であるか否かを判定する（3901）。具体的には

50

、遊技制御装置 100 は、普電（普通電動役物）としての普通変動入賞装置 33 の開閉部材が作動し、遊技球の入賞が可能な開状態となっているか否かを判定する。開閉部材が開状態であって普電作動中である場合には（3901の結果が「Y」）、遊技制御装置 100 はステップ 3903 の処理を実行する。これに対して、開閉部材が開状態であって普電作動中でない場合には（3901の結果が「N」）、遊技制御装置 100 はステップ 3902 の処理を実行する。

#### 【0631】

遊技制御装置 100 は、不正入賞エラーが発生したか否かを判定する（3902）。具体的には、遊技制御装置 100 は、普通変動入賞装置 33 の開閉部材が開状態である状態で発生した不正な入賞があるか否かを判定する。なお、普通変動入賞装置 33 の開閉部材が開状態になった後の所定期間に検出された入賞は、不正入賞ではないと判定される。不正入賞エラーが発生している場合には（3902の結果が「Y」）、遊技制御装置 100 は始動口 SW 監視処理を終了する。これに対して、不正入賞エラーが発生していない場合には（3902の結果が「N」）、遊技制御装置 100 はステップ 3903 の処理を実行する。

10

#### 【0632】

遊技制御装置 100 は、第 2 始動入賞（普通変動入賞装置 33 の第 2 始動入賞口 34 への入賞）があるか否かを判定する（3903）。普通変動入賞装置 33 の第 2 始動入賞口 34 への入賞があるか否かは、第 2 始動入賞口 34 への入賞があった場合に第 2 特図始動 SW 34A から出力される遊技球の検出信号の有無によって判定される。第 2 始動入賞がない場合には（3903の結果が「N」）、遊技制御装置 100 は、始動口 SW 監視処理を終了する。これに対して、第 2 始動入賞がある場合には（3903の結果が「Y」）、遊技制御装置 100 は特図始動口 SW 共通処理（3904）を実行し、始動口 SW 監視処理を終了する。

20

#### 【0633】

図 40 を参照して、図 39 の始動口 SW 監視処理における特図始動口 SW 共通処理（3904）の詳細を説明する。図 40 は、本発明の第 1 の実施の形態の特図始動口 SW 共通処理の手順を示すフローチャートである。

#### 【0634】

特図始動口 SW 共通処理では、遊技制御装置 100 は、まず対応する始動入賞フラグを設定する処理を実行する（4000）。遊技制御装置 100 は、第 1 始動入賞口 37 と普通変動入賞装置 33 の第 2 始動入賞口 34 のいずれの始動口への入賞により特図始動口 SW 共通処理が開始されたかに基づき始動入賞フラグを設定する。始動入賞フラグとしては、第 1 始動入賞口 37 への入賞により特図始動口 SW 共通処理が開始された場合は特図 1 フラグが設定され、普通変動入賞装置 33 の第 2 始動入賞口 34 への入賞により特図始動口 SW 共通処理が開始された場合は特図 2 フラグが設定される。以下の処理では、特図 1 フラグが設定されていれば対応する処理として特図 1 始動記憶についての処理が行われ、特図 2 フラグが設定されていれば対応する処理として特図 2 始動記憶についての処理が行われる。

30

#### 【0635】

遊技制御装置 100 は、まず、遊技球が入賞した始動入賞口に対応するフラグを設定する（4000）。具体的には第 1 始動入賞口に遊技球が入賞した場合には第 1 始動入賞フラグを設定し、第 2 始動入賞口に遊技球が入賞した場合には第 2 始動入賞フラグを設定する。

40

#### 【0636】

次に、遊技制御装置 100 は、対応する保留の数が上限（例えば 4）であるか否かを判定する（4001）。対応する保留の数とは、第 1 始動入賞口 37 に遊技球が入賞した場合には第 1 特図保留カウンタの値が対象となり、第 2 始動入賞口 34 に遊技球が入賞した場合には第 2 特図保留カウンタの値が対象となる（ともに図 15 参照）。

#### 【0637】

50

このとき、対応する保留の数が上限となっていて、これ以上対応する始動記憶を記憶できない場合には(4001の結果が「Y」)、遊技制御装置100は本処理を終了して始動口SW監視処理を終了する。これに対して、対応する保留が上限でない場合には(4001の結果が「N」)、遊技制御装置100は、対応する始動記憶数(即ち、第1特図保留カウンタ又は第2特図保留カウンタのうち、入賞した始動口に対応する方)を1インクリメントする処理を実行し(4002)、その後、ステップ4003の処理を実行する。

#### 【0638】

遊技制御装置100は、保留数情報を設定する処理を実行する(4003)。遊技制御装置100は、対応する始動記憶数が1インクリメントされたことに基づき、始動記憶数の情報である保留数情報を設定する。なお、保留数情報は、始動記憶に関する情報を含む保留情報コマンド(図35参照)の一つとして後に演出制御装置150に送信されるようになっている。その後、遊技制御装置100は、対応する大当たり判定用乱数を取得する処理(4004)、対応する大当たり図柄乱数を取得する処理(4005)、対応する特図変動表示ゲームの変動パターン乱数を取得する処理(4006)を実行する。ここでは、図15に示される、大当たり乱数、大当たり図柄乱数、第1~第3変動パターン乱数の各生成領域から、乱数値を取得する。

10

#### 【0639】

遊技制御装置100は、ステップ4004~4006で取得した各種乱数値を用いて、特図保留情報判定処理を実行する(4007)。この特図保留情報判定処理の詳細は、図41を参照して後述する。

20

#### 【0640】

そして、遊技制御装置100は、対応する大当たり判定用乱数を記憶する処理(4008)、対応する大当たり図柄乱数を記憶する処理(4009)、対応する変動パターン乱数を記憶する処理を実行する(4010)。ここでは、遊技制御装置100は、ステップ4004~4006で取得した各種乱数値を、対応する乱数記憶領域に記憶する。

#### 【0641】

対応する乱数記憶領域とは、処理対象の始動口が第1始動入賞口37であれば、図16で前述した、第1特図変動表示ゲームを実行するために必要な各種乱数の保存領域のうち、第1特図保留カウンタに対応する領域が該当する。同様に、処理対象の始動口が第2始動入賞口34であれば、図17で前述した、第2特図変動表示ゲームを実行するために必要な各種乱数の保存領域のうち、第2特図保留カウンタに対応する領域が該当する。

30

#### 【0642】

例えば、第2始動入賞口34に遊技球が入賞した結果、ステップ4002の処理にて第2特図保留カウンタが1インクリメントされて「3」となった場合は、図17における「保留3の保存領域」に割り当てられた各種乱数の保存領域に、ステップ4004~4006で取得した各種乱数値を記憶することになる。

#### 【0643】

次いで、遊技制御装置100は、保留情報コマンドを設定する処理を実行し(4011)、本処理を終了する。保留情報コマンドは、始動口への入賞によって更新された後の保留数(第1特図保留カウンタ又は第2特図保留カウンタの値)を通知するコマンドであり、直後に演出制御装置150に送信される(正確には、次回のタイマ割込処理(図29)のコマンド送信処理(2912)が実行されるタイミングで送信される)ようになっている。

40

#### 【0644】

続いて、図41を参照して、図40の特図始動口SW共通処理における特図保留情報判定処理(4007)の詳細を説明する。図41は、本発明の第1の実施の形態の特図保留情報判定処理の手順を示すフローチャートである。

#### 【0645】

特図保留情報判定処理では、乱数記憶領域へ新たに始動記憶が記憶される際に、すなわち、第1始動入賞口37や第2始動入賞口34(普通変動入賞装置33)に遊技球が入賞

50

したタイミングで、当該始動記憶に対応する結果関連情報の判定が行われる。つまり、特図保留情報判定処理では、新たに記憶された始動記憶に基づく特図変動表示ゲームの開始タイミングより前に、当該始動記憶に対応した結果関連情報の判定を行う先読み処理が行われることになる。

【0646】

特図保留情報判定処理では、遊技制御装置100は、まず特図2始動記憶についての処理であるか否かを判定する(4100)。特図2始動記憶についての処理である場合には(4100の結果が「Y」)、常に先読み処理を実行するため、遊技制御装置100はステップ4103の処理を実行する。

【0647】

これに対して、特図2始動記憶についての処理でない場合(4100の結果が「N」)、つまり、特図1始動記憶についての処理である場合には、遊技制御装置100は、普通変動入賞装置33の作動状態が抑制状態であるか否かを判定する(4101)。普通変動入賞装置33の作動状態が抑制状態でない場合には(4101の結果が「N」)、遊技制御装置100は、特図1始動記憶に関連する先読み演出を実行することなく、特図保留情報判定処理を終了する。

【0648】

これに対して、普通変動入賞装置33が抑制状態である場合には、遊技制御装置100は、大当たり中であるか否かを判定する(4102)。遊技状態が特別遊技状態であって、大当たり中である場合には(4102の結果が「Y」)、遊技制御装置100は、特図1始動記憶に対する先読み演出をせずに、特図保留情報判定処理を終了する。

【0649】

これに対して、大当たり中でない場合には(4102の結果が「N」)、遊技制御装置100は、始動記憶に対して大当たり乱数が大当たり判定値と一致するかを判定し、大当たりである場合には大当たりの種類や変動パターンを決定する特図大当たり判定処理を実行する(4103)。なお、特図大当たり判定処理についての詳細は、図42にて後述する。

【0650】

遊技制御装置100は、ステップ4103の処理で一時的に記憶される大当たり種類を特定する情報及び変動パターンを特定する情報を用いて、変動前演出指定コマンドを設定する(4104)。変動前演出指定コマンドは、後述する図43及び図44の変動パターンテーブルを用いて設定され、一旦、遊技制御装置100のRAM104に記憶される。そして、次のタイマ割込処理(図29参照)が実行されて、変動前演出指定コマンドが遊技制御装置100から演出制御装置150に送信されると、演出制御装置150の変動パターン情報記憶領域に記憶される。

【0651】

図42を参照して、図41の特図保留情報判定処理における特図大当たり判定処理(4103)の詳細を説明する。なお、後述の図46の特図1変動開始処理(特図2変動開始処理)における特図大当たり判定処理(4103)を実行する際にも、図42の処理が行われる。図42は、本発明の第1の実施の形態の特図大当たり判定処理の手順を示すフローチャートである。

【0652】

特図大当たり判定処理では、判定対象の始動記憶の大当たり種類及び変動パターンを決定する処理が行われる。なお、図41の特図保留情報判定処理の特図大当たり判定処理を実行する場合は、乱数記憶領域へ新たに記憶された始動記憶が判定対象となるが、図46の特図1変動開始処理(特図2変動開始処理)の特図大当たり判定処理を実行する場合は、変動開始となる始動記憶が判定対象となる。すなわち、特図大当たり判定処理は、複数の呼び出し元から参照される共通モジュールとなっている。

【0653】

遊技制御装置100は、まず、特図確変フラグはオンになっているか否かを判定する(4200)。そして、特図確変フラグがオンの場合には(4200の結果が「Y」)、遊

10

20

30

40

50



技制御装置 100 は、確変状態における、すなわち特図確率が高確率時の大当たり判定を実行する(4201)。なお、特図確変フラグは、特図表示中処理(図37のステップ3708)においてオン/オフに設定される。

【0654】

特図確変フラグがオンでない場合には(4200の結果が「N」)、遊技制御装置100は、非確変状態、すなわち特図確率が低確率時の大当たり判定を実行する(4202)。

【0655】

次に、遊技制御装置100は、対応する始動記憶が大当たりであるか否かを判定する(4203)。すなわち、始動記憶の大当たり乱数の値が、大当たり判定値と一致するか否かを判定する。

【0656】

そして、大当たりである場合には(4203の結果が「Y」)、遊技制御装置100は、対応する始動記憶の大当たり図柄乱数を用いて、確変大当たりや通常大当たりといった大当たりの種類を決定する(4204)。

【0657】

これに対し、対応する始動記憶が大当たりではない場合には、遊技制御装置100は、大当たりの種類が「はずれ」となったことをRWMに一時的に記憶する(4205)。

【0658】

続いて、遊技制御装置100は、対応する始動記憶の特図変動表示ゲームが短時間変動となる条件が成立しているか否かを判定する(4206)。ここでは、判定対象が特図2変動表示ゲームの始動記憶である場合は、短時間変動となる条件が成立しているものと見なされる。あるいは、判定対象が特図1変動表示ゲームの始動記憶であっても、当該判定を行う時点における特図1変動表示ゲームの始動記憶数が上限数(ここでは4)となっているか、普通変動入賞装置33の作動状態が促進状態となっているかのいずれかの状態である場合は、短時間変動となる条件が成立しているものと見なされる。

【0659】

そして、短時間変動となる条件が成立する場合には(4206の結果が「Y」)、遊技制御装置100は、対応する始動記憶の第1~第3変動パターン乱数により短時間変動の変動パターンを決定する(4207)。

【0660】

これに対し、短時間変動となる条件が成立していない場合には(4206の結果が「N」)、遊技制御装置100は、対応する始動記憶の第1~第3変動パターン乱数により長時間変動の変動パターンを決定する(4208)。なお、変動パターン決定の仕方の詳細については図43及び図44にて後述する。

【0661】

最後に、遊技制御装置100は、はずれを含む大当たり種類を特定する情報と、変動パターンを特定する情報を一時的にRWMに記憶する(4209)。ここでは、特定する情報とは、変動パターン番号である。その後、遊技制御装置100は、特図大当たり判定処理を終了する。

【0662】

図43及び図44を参照して、図42の変動パターンの決定(ステップ4207及びステップ4208)やステップ4209における変動パターン番号の決定について説明する。図43は、本発明の第1の実施の形態の後半変動パターンテーブルである。また、図44は、本発明の第1の実施の形態の前半変動パターンテーブルである。

【0663】

図43(A)は、図42のステップ4203の処理において、対応する始動記憶が大当たりではないと判定された場合に選択されるテーブルであり、特図変動表示ゲームの後半の変動パターンが決定される。

【0664】

まず、対応する始動記憶の第1変動パターン乱数の値に応じて、グループ番号0~2の

10

20

30

40

50

いずれかが選択される。ここで、グループ番号「0」は「リーチなし」、番号「1」は「ノーマルリーチ」、番号「2」は「SPリーチ」を示す。「リーチなし」が選択される確率は $2701/3001$ 、「ノーマルリーチ」が選択される確率は $295/3001$ 、「SPリーチ」が選択される確率は $5/3001$ である。

【0665】

次に、対応する始動記憶の第2変動パターン乱数の値に応じて、後半変動パターン番号01h~23hのいずれかが選択される。ここで、後半変動パターン番号「01h」は「リーチなし変動」で変動時間2秒を示す。後半変動パターン番号11h~13hは「ノーマルリーチ」変動であり、番号「11h」は変動時間9秒の「ノーマルリーチ短変動」、番号「12h」は変動時間10秒の「ノーマルリーチ中変動」、番号「13h」は変動時間11秒の「ノーマルリーチ長変動」を示す。また、後半変動パターン番号21h~23hは「SPリーチ」変動であり、番号「21h」は変動時間37秒の「SPリーチ短変動」、番号「22h」は変動時間38秒の「SPリーチ中変動」、番号「23h」は変動時間39秒の「SPリーチ長変動」を示す。

10

【0666】

図43(B)は、図42のステップ4203において、対応する始動記憶が大当たりであると判定された場合に選択されるテーブルであり、特図変動表示ゲームの後半の変動パターンが決定される。ここでは、図43(A)と異なることを説明する。

【0667】

大当たりが発生する場合は、後半変動パターンにおいてグループ番号「0」の「リーチなし」が選択されることはない。「ノーマルリーチ」又は「SPリーチ」が選択される。「ノーマルリーチ」が選択される確率は $1501/3001$ 、「SPリーチ」が選択される確率は $1500/3001$ である。

20

【0668】

なお、「ノーマルリーチ」からさらに第2変動パターン乱数に応じて選択される後半変動パターン番号は、51h~53hとなるが、それぞれ11h~13hに対応しており、選択される確率も同じである。同様に、「SPリーチ」の後半変動パターン番号は、61h~63h(21h~23hに対応)となる。

【0669】

このように、変動パターンの決定には、まず、第1変動パターン乱数を用いて後半変動のリーチ種類(なしを含む)を決定し、次に第2変動パターン乱数を用いて後半変動パターン(変動時間)を決定する。そして、後半変動パターンを示す後半変動パターン番号と、第3変動パターン乱数とを用いて、前半変動パターンを決定する(図44参照)。

30

【0670】

続いて、図44を参照して、前半変動パターンテーブルについて説明する。

【0671】

図44では、後半変動パターン番号01h~63h毎に設定された振分率に応じて、第3変動パターン乱数を用いて前半変動パターンが決定される。

【0672】

前半変動パターンには、「前半キャラクタA出現変動」と、「前半キャラクタB出現変動」と、「前半キャラクタなし変動」の変動時間を短縮した「前半短縮変動」とがある。それぞれの変動時間は、10秒、10秒、10秒、1秒に設定されている。

40

【0673】

第1変動パターン乱数及び第2変動パターン乱数を用いた結果、後半変動パターンを「ノーマルリーチ変動」とする決定がなされている場合には、第3変動パターン乱数の値が0~200の場合に「前半キャラクタA出現変動」を選択し、第3変動パターン乱数の値が201~205の場合に「前半キャラクタB出現変動」を選択し、第3変動パターン乱数の値が206~400の場合に「前半キャラクタなし変動」を選択する。

【0674】

第1変動パターン乱数及び第2変動パターン乱数を用いた結果、後半変動パターンを「

50

「SPリーチ変動」とする決定がなされている場合には、必ずキャラクタが出現するように制御され、第3変動パターン乱数の値が0～200の場合に「前半キャラクタA出現変動」を選択し、第3変動パターン乱数の値が201～400の場合に「前半キャラクタB出現変動」を選択する。

#### 【0675】

第1変動パターン乱数及び第2変動パターン乱数を用いた結果、後半変動パターンを「リーチなし変動」とする決定がなされている場合（後半変動パターン番号＝01hの場合）には、第3変動パターン乱数の値が0～5のときに「前半キャラクタA出現変動」が選択される。第3変動パターン乱数の値が6～400の場合には、図42のステップ4206における短時間変動の条件が成立しているときに「前半短縮変動」が選択され、この条件が成立していないときは「前半キャラクタなし変動」が選択される。

10

#### 【0676】

本発明の第1の実施の形態では、後半変動パターン番号01hの「リーチなし変動」の場合にのみ、「前半短縮変動」が設定されている。そして、「リーチなし変動」は、大当りの種類が「はずれ」である場合に高い確率で選択される。また、「リーチなし変動」の場合には「前半キャラクタなし変動」が選択される確率は低く、「前半短縮変動」又は「前半キャラクタなし変動」が選択される確率が高い。

#### 【0677】

したがって、保留記憶数が多い場合でははずれのときは「リーチなし・前半短縮変動」が選択される確率が高いため、全体の平均変動時間を短縮する。そして、保留記憶数が少ない場合でははずれのときは「リーチなし・前半キャラクタなし変動」が選択される確率が高くなっている。

20

#### 【0678】

これに対して、はずれのときは、リーチを伴う変動（「ノーマルリーチ変動」及び「SPリーチ変動」）や、リーチを伴わない「リーチなし・前半キャラクタA出現変動」が選択される確率は低くなっている。なお、特図変動表示ゲームが大当たりとなる確率は、低確率時において1/301とし、高確率時において1/61とするものとする。

#### 【0679】

本実施形態では、リーチを伴う変動（「ノーマルリーチ変動」及び「SPリーチ変動」）や、「リーチなし・前半キャラクタA出現変動」を、保留記憶数の数に関係なく共通の変動時間が設定された共通変動態様とする。これらの変動態様は、特図の大当たり結果と、第1～第3変動パターン乱数とによって決定されるので、決定処理を行う時点における保留記憶数の数に依存することなく決定されるものである。つまり、同一の保留記憶であれば、特図保留情報判定処理（図41）の特図大当たり判定処理（4104）により決定される変動態様と、特図1（特図2）変動開始処理（図46）の特図大当たり判定処理（4606）による決定される変動態様とが一致する。

30

#### 【0680】

これに対して、「リーチなし・前半短縮変動」や「リーチなし・前半キャラクタなし変動」は、保留記憶数の数に基づいていずれかが選択される変動パターンであるので、固有変動態様とする。これらの変動態様は、特図の大当たり結果と、第1～第3変動パターン乱数のみならず、決定処理を行う時点における保留記憶数の数に依存して決定されるものである。つまり、同一の保留記憶であっても、特図保留情報判定処理（図41）の特図大当たり判定処理（4104）により決定される変動態様と、特図1（特図2）変動開始処理（図46）の特図大当たり判定処理（4606）により決定される変動態様とが一致するとは限らない。ただし、前者の処理で決定された変動態様が固有変動態様であれば、後者の処理で決定された変動態様も固有変動態様となるように設定されている。

40

#### 【0681】

なお、これらの前半変動パターンには、前半変動パターン番号が設けられているが、始動入賞時及び変動開始時のどちらのタイミングで決定された前半変動パターンかによって前半変動パターン番号が異なる。

50

## 【0682】

本テーブルを用いる図42の特図大当り判定処理は、遊技球が第1始動入賞口37又は普通変動入賞装置33の第2始動入賞口に入賞したときに実行される特図保留情報判定処理(図41)で対象の始動記憶の結果を先読みするために実行され、さらに、後述する特図1(特図2)変動開始処理(図46)でも実行される。

## 【0683】

これは、始動入賞時と変動開始時とでは、保留記憶数が異なったり、遊技状態が異なったりする。すなわち、始動入賞時と変動開始時とでは、図42の特図大当り判定処理における処理結果が異なる場合があり、選択される前半変動パターンも異なる場合がある。このため、始動入賞時及び変動開始時のどちらの場合に決定した前半変動パターン番号かが判別できるようにしている。

10

## 【0684】

ここでは、始動入賞時に決定した前半変動パターン番号の頭文字を「A」とし、変動開始時に決定した前半変動パターン番号の頭文字を「B」とする。そして、「前半キャラクタA出現変動」の場合はE/Cに続く番号を「1h」とし、「前半キャラクタB出現変動」の場合は「2h」とし、「前半キャラクタなし変動」の場合は「3h」とし、「前半短縮変動」の場合は「4h」とする。このように、互いに共通するモジュールを用いる。

## 【0685】

なお、図50で後述するが、本実施形態では、「前半キャラクタA出現変動」では「キャラクタA」が出現し、「前半キャラクタB出現変動」では「キャラクタB」が出現するようになっている。

20

## 【0686】

次に、図45を参照して、図37の特図ゲーム処理における特図普段処理(3706)の詳細を説明する。図45は、本発明の第1の実施の形態の特図普段処理の手順を示すフローチャートである。

## 【0687】

特図普段処理では、遊技制御装置100は、まず、保留された特図2始動記憶があるかを判定する(4500)。

## 【0688】

特図2始動記憶がある場合には(4500の結果が「Y」)、遊技制御装置100は、特図2変動開始処理を実行し(4501)、その後、図37に示した特図ゲーム処理で特図変動中処理が実行されるように特図ゲーム処理番号を1に設定する(4504)。

30

## 【0689】

さらに、遊技制御装置100は、特図2変動開始処理によって算出され、決定した前半変動パターン及び後半変動パターンに対応する合計変動時間を特図ゲーム処理タイマに設定する処理を実行する(ステップ4505)。その後、特図普段処理を終了する。なお、ステップ4505の処理で変動時間が特図ゲーム処理タイマに設定されるので、変動時間が経過するまでは、図37の特図変動中処理(3707)は実行されない。

## 【0690】

一方、特図2始動記憶がない場合には(4500の結果が「N」)、遊技制御装置100は、保留された特図1始動記憶があるかを判定する(4502)。特図1始動記憶がある場合には(4502の結果が「N」)、遊技制御装置100は、特図1変動開始処理を実行する(4503)。その後、遊技制御装置100は、特図ゲーム処理番号を1に設定する処理を実行する(4504)。さらに、特図1変動開始処理で算出され、決定した変動時間を特図ゲーム処理タイマに設定する処理を実行し(4505)、特図普段処理を終了する。

40

## 【0691】

特図1始動記憶がないと判定された場合には(4502の結果が「N」)、遊技制御装置100は、客待ちデモフラグの設定があるかを判定する(4506)。客待ちデモフラグの設定がある場合には(4506の結果が「Y」)、遊技制御装置100は特図普

50

段処理を終了する。これに対して、客待ちデモフラグの設定がない場合には（４５０６の結果が「N」）、遊技制御装置１００は、客待ちデモフラグをセットする処理を実行する（４５０７）。さらに、客待ちデモコマンドの設定を実行し（４５０８）、特図普段処理を終了する。

【０６９２】

なお、ステップ４５０７の処理でセットされた客待ちデモコマンドは、後に演出制御装置１５０に送信される。演出制御装置１５０は、この客待ちデモコマンドの受信に基づき、例えば表示装置８の表示部に客待ちデモ画像を表示する処理を実行する。

【０６９３】

図４５に示したように、本実施形態では、遊技制御装置１００は、特図２始動記憶がある場合に当該特図２始動記憶に基づく特図２変動表示ゲームを、特図１始動記憶に基づく特図１変動表示ゲームよりも優先的に実行する。

10

【０６９４】

次に、図４６を参照して、図４５の特図普段処理における特図１／特図２変動開始処理（ステップ４５０３、ステップ４５０１）の詳細を説明する。図４６は、本発明の第１の実施の形態の特図１／特図２変動開始処理の手順を示すフローチャートである。

【０６９５】

まず、特図１変動開始処理では、遊技制御装置１００は、第１特図変動表示ゲームに関する各種乱数の保存領域（図１６参照）のうち、次回の変動の対象となる領域、すなわち保留１個目に対応する保留乱数記憶領域を指定する（４６００）。次に、ステップ４６０

20

の処理で指定した記憶領域から大当り乱数、大当り図柄乱数、第１～第３変動パターン乱数を取得した後、保留１～３個目に対応する保留乱数記憶領域に、保留２～４個目に対応する保留乱数記憶領域（次回以降の変動に対応する記憶領域）の各乱数をシフトする（４６０１）。そして、特図１の保留数を１減算し（４６０２）、ステップ４６０６以降の処理を実行する。

【０６９６】

次いで、遊技制御装置１００は、ステップ４６０１の処理で取得した保留１個目の各種乱数を用いて、図４２に示した特図大当り判定処理を実行する（４６０６）。次に、図４２の特図大当り判定処理のステップ４２０９の処理において一時的に記憶した情報を用いて、演出制御装置１５０に送信する飾り図柄変動パターンコマンドを設定する（４６０

30

７）。飾り図柄変動パターンコマンドは、表示装置８で実行される飾り特図変動表示ゲームの内容を決めるコマンドである。

【０６９７】

さらに、遊技制御装置１００は、保留情報コマンドを設定する（４６０８）。前述したように、保留情報コマンドは特図１又は特図２の保留数を通知するコマンドであり、ここでは、特図１又は特図２のうち、変動対象の特図の保留数を設定する。そして、ステップ４６０９において図４５の特図普段処理のステップ４５０５の処理で設定した、決定した変動パターンに対応する変動時間を取得し（４６０９）、その後、特図１変動開始処理を終了する。

【０６９８】

同様に、特図２変動開始処理では、遊技制御装置１００は、前述の特図１変動開始処理のステップ４６００～４６０２の処理に対応してステップ４６０３～４６０５の処理を実行する。

40

【０６９９】

遊技制御装置１００は、第２特図変動表示ゲームに関する各種乱数の保存領域（図１７参照）のうち、次回の変動の対象となる領域、すなわち保留１個目に対応する保留乱数記憶領域を指定する（４６０３）。次に、ステップ４６０３の処理で指定した記憶領域から大当り乱数、大当り図柄乱数、第１～第３変動パターン乱数を取得した後、保留１～３個目に対応する保留乱数記憶領域に、保留２～４個目に対応する保留乱数記憶領域（次回以降の変動に対応する記憶領域）の各乱数をシフトする（４６０４）。そして、特図２の保

50

留数を1減算して(4605)、ステップ4606以降の処理を実行する。

【0700】

なお、ステップ4606では、ステップ4604の処理で取得した保留1個目の各種乱数を用いて、図42に示した特図大当り判定処理を実行することになる。

【0701】

このように、特図1又は特図2の変動が開始されるときにも特図大当り判定処理が実行されることで、飾り特図変動表示ゲームの表示内容が決定される。

【0702】

図47A及び図47Bは、本発明の第1の実施の形態における始動記憶をシフトさせる処理のプログラムの一例を示す図である。図47A及び図47Bに示すプログラムは、図46のステップ4601の処理に対応する。図47AはKレジスタを利用してプログラムのコード量を削減したプログラムであって、図47BはKレジスタを利用しないプログラムである。

10

【0703】

図47Aを参照すると、まず、行4701に示すように、Kレジスタに「28H」を設定し、その後、行4702に示すように、LDK命令によって指定された下位アドレスとKレジスタに設定された上位アドレスとを組み合わせたアドレスにアクセスする。このように、CPUコア102はKレジスタに固有の上位アドレスを設定する固有値設定手段をなす。

【0704】

前述のように、保留2に対応する始動記憶を保留1に対応する領域にシフトする場合には、図16に示したように、保留2に対応する始動記憶のアドレス「2838H」～「283FH」に格納されたデータを、保留1に対応する始動記憶にアドレス「2830H」～「2837H」に移動させる。行4702では、アドレス2838Hに格納されたデータをAレジスタに格納し、行4703でAレジスタに格納されたデータをアドレス2830Hに格納することによってデータを移動させている。同様に、「2839H」～「283FH」に格納されたデータを、「2831H」～「2837H」に移動させることによって保留2に対応する始動記憶を保留1に対応する領域にシフトさせる。

20

【0705】

なお、保留3に対応する始動記憶を保留2に対応する始動記憶にシフトさせる処理、及び保留4に対応する始動記憶を保留3に対応する始動記憶にシフトさせる処理も、図に示すように、後半のプログラムによって実行される。

30

【0706】

図47Bは、図47Aと同様に、始動記憶の消化にともなって、始動記憶をシフトさせる処理のプログラムを示しており、図47Aのプログラムを実行した場合と同じ処理が実行される。しかしながら、図47Bに示すプログラムでは、2バイトのアドレスを直接指定している(例えば、行4722の「2838H」)。

【0707】

以上のように、図47Aに示すプログラムでは、始動記憶が格納された領域のアドレスを指定するために1バイトの下位アドレスのみを指定すればよい(4710)。これに対し、図47Bに示すプログラムでは、アドレスの指定に2バイトの容量を要し(4730)、図47Aに示したプログラムよりも多くの容量を必要とする。したがって、始動記憶を格納する領域のように、連続した領域にデータを格納し、これらのデータにアクセスする場合には、図47Aに示したプログラムのように、上位アドレスをKレジスタに予め保持してデータにアクセスするようにすることによってプログラムの容量を削減することができる。

40

【0708】

また、前述のように、Kレジスタ(上位アドレス指定レジスタ)に固有のアドレス値(上位アドレス、例えば、「28H」)を予め設定しておくことによって、その後、下位アドレスのみ指定すれば所定の領域にアクセスできるため、CPUコア(演算処理手段)1

50

02の処理負担を軽減することもできる。

【0709】

図48は、本発明の第1の実施の形態のサブルーチン(サブモジュール)を構成するプログラムの一例を示す図である。(A)及び(B)は、プログラム中に記述されたサブルーチン呼び出しの命令(CALL命令又はRST命令)を実行した際に呼び出されるサブルーチンの一例を示している。これに対して(C)は、割込信号の発生により呼び出されるサブルーチンの一例を示している。

【0710】

まず(A)は、呼出元のルーチンで格納されたフラグレジスタの値が、呼出先のサブルーチンにおける処理過程で書き換えられないように構成したプログラムの例である。このサブルーチンは、5200Hのアドレスから処理を開始する例となっており、最初に「PUSH FLG」の命令を実行することで、フラグレジスタの値をスタック領域に退避している(行4811)。

10

【0711】

なお、呼出元のルーチンで格納されたフラグレジスタ以外のレジスタであっても、サブルーチン内で書き換えられないように構成する場合は、「PUSH」命令を用いて、該当するレジスタをスタック領域に退避することができる。例えば、行4812に示すように、「PUSH HL」の命令を実行することで、HLレジスタの値をスタック領域に退避することができる。

【0712】

そして、サブルーチンに記述された一連のプログラムを実行終了した場合は、「POP」命令を実行して、スタック領域に退避しておいたHLレジスタ及びフラグレジスタの値を復帰させてから(行4813、4814)、「RET」命令を実行して呼出元の処理に戻るようになっている(行4815)。

20

【0713】

一方(B)は、サブルーチンにおける処理過程にて演算結果をフラグレジスタに設定し、そのフラグレジスタの値を呼出元のルーチンに返すプログラムの一例である。このサブルーチンは、5100Hのアドレスから処理を開始する例となっており、(A)のように「PUSH」「POP」命令を用いて、フラグレジスタの値をスタック領域に出し入れするような処理を行っていない。

30

【0714】

そして、サブルーチンに記述された一連のプログラムを実行終了した場合は、AレジスタとEレジスタの値の論理和を演算することでフラグレジスタに演算結果が設定され(行4821)、さらに、Dレジスタの値をAレジスタにロードしてから(行4822)、「RET」命令を実行して呼出元に戻るようになっている(行4823)。

【0715】

この(B)のようなサブルーチンの構成は、サブルーチン内で所定の演算を実行するとともに、その演算結果をフラグレジスタに格納し、さらに格納されたフラグレジスタの値が呼出元のルーチンに戻されてからも活用されるような場合に適している。そのため、呼び出されるサブルーチンのプログラムには、フラグレジスタの値をスタック領域に出し入れする「PUSH」「POP」命令が記述されていない。

40

【0716】

このようなサブルーチンの構成は、予め共通化した処理を遊技制御プログラムの様々な箇所で行わせる場合に適しており、共通化した処理のブロックを予めサブルーチンとしてプログラム中に配置しておいて、呼出元となるプログラムでCALL命令(又はRST命令)を実行することで実現できる。

【0717】

ただし、プログラム製作者の立場を考慮すると、(A)のようにサブルーチンで用いたフラグレジスタと呼出元ルーチンで用いるフラグレジスタとを別個のものとして扱いたい場合もあれば、(B)のようにサブルーチンで用いたフラグレジスタと呼出元ルーチンで

50

用いるフラグレジスタとを同一のものとして扱いたい場合もある。

【0718】

そのため、サブルーチンを呼び出す目的に応じて、フラグレジスタの値を退避させるか否かの選択を行う必要がある。従って、サブルーチンを構成するプログラムに「PUSH」「POP」命令を記述するか否かによって、フラグレジスタをスタック領域に出し入れするか否かを選択できるように構成しておくことが好ましいことになる。

【0719】

一方(C)は、タイマ割込などの割込信号で呼び出されるサブルーチン(割込処理)を構成したプログラムの例である。このサブルーチンが呼び出されると、図19(D)に示すように、「PUSH」命令を実行しなくてもフラグレジスタの値がスタック領域に退避される。このとき、呼出元ルーチンで使用していたレジスタ群のバンクが0(表レジスタ)であれば、フラグレジスタのRBSのビットは「0」となっている。

10

【0720】

なお、このサブルーチンは、呼出元ルーチンで実行中の処理を任意のタイミングで中断して実行されるため、呼び出されたサブルーチンで用いられるフラグレジスタの値と、呼出元のルーチンで使用しているフラグレジスタの値には相関がない。そのため、呼出元のルーチンで使用しているフラグレジスタの値が、割込処理のサブルーチン内で変更されると、処理結果が予期しないものとなる。

【0721】

それ故に、割込処理のサブルーチンが呼び出されたときは、例外なくフラグレジスタの値をスタック領域に退避させる必要がある。(C)の構成であれば、フラグレジスタの値をスタック領域に退避させる「PUSH」命令のコードが不要となるので、割込処理のサブルーチンを記述するプログラムのコード量を削減することができる。

20

【0722】

ちなみに(C)の割込処理のサブルーチンは、5300Hのアドレスから処理を開始する例となっている。そして、行4831の「LD RBS, 1」を実行すると、フラグレジスタのRBSのビットが「1」となる。これにより、使用可能なレジスタ群が、呼出元ルーチンで使用していたバンク0(表レジスタ)から、バンク1(裏レジスタ)に切り替えられる。このとき、バンク0のレジスタ群の各レジスタに格納された値はそのまま保持され、割込処理のサブルーチンでは、バンク1のレジスタ群のレジスタを用いて実行される。

30

【0723】

さらに、割込処理のサブルーチンの最後で「RETI」を実行することによって、呼出元の処理に戻るようになっている(行4832)。このとき、スタック領域に退避してあったフラグレジスタの値が復帰することで、フラグレジスタのRBSのビットが「0」に復帰し、以降使用されるレジスタ群のバンクが0に戻される。

【0724】

なお(A)のサブルーチンにおいて、使用可能なレジスタ群をバンク1(裏レジスタ)に切り替える場合は、行4811の「PUSH FLG」を実行した後に、「LD RBS, 1」の命令を実行すればよい。これにより、フラグレジスタのRBSのビットが「1」となり、使用可能なレジスタ群が、呼出元ルーチンで使用していたバンク0(表レジスタ)から、バンク1(裏レジスタ)に切り替わる。この場合は、行4814の「POP FLG」を実行することで、フラグレジスタのRBSのビットを「0」に復帰させ、呼出元の処理に戻る直前にレジスタ群のバンクを0に戻すことになる。

40

【0725】

また(B)のサブルーチンにおいて、使用可能なレジスタ群をバンク1(裏レジスタ)に切り替える場合は、プログラム中で「LD RBS, 1」の命令を実行すればよいが、呼出元ルーチンに戻る直前で「LD RBS, 0」の命令を実行しておく必要がある。

【0726】

以上、図48の(A)~(C)に示したように、本実施形態では、呼出元のルーチンで

50



、プログラム中に記述されたサブルーチン呼び出しの命令（CALL命令又はRST命令）を実行した場合や、割込信号が発生した場合には、サブルーチンが呼び出され、その際に、呼出元のルーチンに含まれる戻りアドレスの値をスタック領域に格納する構成となっている。

【0727】

この場合において、割込信号は呼出元のルーチンの実行中のアドレスとは無関係に発生するものであるので、（C）に示すような割込処理のサブルーチンを呼び出す際は、割込信号が発生した時点におけるフラグレジスタ1200の値は、呼出先のルーチンの処理中で破壊されないように、スタック領域に退避しておくことが好ましい。

【0728】

そのため、図19の（D）に示すように、呼出元のルーチンに含まれる戻りアドレスの値だけでなく、フラグレジスタ1200の値もスタック領域に格納して、呼出先の処理に移行させる構成となっている。これにより、呼出先のルーチンのプログラム中にフラグレジスタをスタック領域に格納するための命令（「PUSH FLG」）を記述しなくても、フラグレジスタの値が自動的にスタック領域に格納されるため、プログラムのコード量を節約することができる。

【0729】

これに対して、呼出元のルーチンに記述されたCALL命令等の呼出の命令を実行する場合は、（A）に示すように、呼出元のルーチンで設定されたフラグレジスタ1200の値を呼出先のルーチンの処理中で破壊しないように構成すべき場合もある一方で、（B）に示すように、呼出先のルーチンの処理中で設定されたフラグレジスタ1200の値を、呼出元のルーチンに戻して演算に利用することもあり得る。つまり、呼出元のルーチンで設定されたフラグレジスタ1200の値をスタック領域に退避すべきか否かはケースバイケースである。

【0730】

そのため、図19の（C）に示すように、呼出元のルーチンに含まれる戻りアドレスの値のみを格納して、フラグレジスタ1200の値はスタック領域に格納しない状態で、呼出先の処理に移行させる構成となっている。そして、呼出先のルーチンのプログラム中に、プログラム中にフラグレジスタ1200の値をスタック領域に格納する命令（例えば、「PUSH FLG」）を記述するか否かにより、値を退避するか否かが選択できる構成となっている。

【0731】

さらに、図48の（A）～（C）に示したように、本実施形態では、汎用レジスタのバンクを切り変える機能を有しているので、呼出先のルーチン内でバンクを切り替えることにより、呼出元とは異なるレジスタを活用することができる。

【0732】

特に（C）に示した割込処理のサブルーチンでは、フラグレジスタの値を復帰させる命令を実行しなくても、処理の最後で「RETI」を実行することによって、スタック領域に退避してあったフラグレジスタの値が自動的に復帰する構成となっている。そのため、サブルーチン内で、フラグレジスタの値を復帰させる命令を実行しなくても、呼出元ルーチンに戻るときにバンクを戻すことができるので、よりプログラムのコード量を節約できるようになっている。

【0733】

図49及び図50は、本発明の第1の実施の形態において、表示装置8で実行される飾り特図変動表示ゲームの画面遷移を説明する図である。

【0734】

画面（a）は変動開始前の状態を示し、始動記憶が消化されると、画面（b）に示すように、図柄表示領域4910にて各識別図柄が変動を開始する。なお、図柄表示領域4910の下方には第1記憶表示部4920と第2記憶表示部4930とが設定される。これらの各記憶表示部には、保留記憶があることを示すマーク（通常の保留記憶表示）494

10

20

30

40

50

0 と、保留記憶がないことを示すマーク 4950 が表示される。その後、画面 (c) に示すように、左図柄、右図柄の順に識別図柄の変動が停止され、最終的には中図柄の変動が停止して、画面 (d) に示すように変動停止した状態となる。

【0735】

このとき、左図柄と右図柄が同一の図柄で停止した場合には、画面 (e) に示すようにリーチ状態となり、その後、中図柄の変動が継続される。そして、変動表示ゲームの結果がはずれの場合には、画面 (f) に示すように、左右の識別図柄とは異なる図柄で中図柄が停止する。また、変動表示ゲームの結果が大当たりとなる場合には、画面 (g) に示すように、左右の識別図柄と同じ識別図柄で中図柄が停止する。変動表示ゲームで大当たりとなると、前述したように特別遊技状態に遷移し、遊技者は多くの遊技価値を得ることが可能となる。

10

【0736】

また、画面 (e) で示したリーチは、通常のリーチであるが、画面 (e) の代わり又は画面 (e) に示したリーチの後に、画面 (h) に示すように、中図柄が回転して表示されるスーパー (SP) リーチが発生する場合がある。SP リーチが発生する場合には、通常のリーチと比較して変動表示ゲームの結果が大当たりとなる確率が大きくなるように設定されている。

【0737】

また、画面 (i) は、変動開始後の状態を示しているが、始動記憶のうちの一つに対して先読み報知が実行されている。具体的には、始動入賞時に設定された前半変動パターン番号がキャラクタ A 又はキャラクタ B が出現する前半変動を行うものであった場合に、先読み報知が実行される。このとき、記憶表示部にて、通常の保留記憶表示 4940 とは表示態様が異なる特定の保留記憶表示 5010 が表示されることで、先読みの報知が行われる。

20

【0738】

その後、始動記憶が消化され (画面 (j))、先読み報知がなされた始動記憶に基づく変動表示ゲームが開始されると (画面 (k))、キャラクタが出現する予告演出が実行される (画面 (l)、画面 (m))。

【0739】

画面 (l) ではキャラクタ A (第 1 のキャラクタ) 5020 が出現し、画面 (m) ではキャラクタ B (第 2 のキャラクタ) 5030 が出現している。これらのキャラクタ A と B は、互いに表示態様が異なるものが好ましい。ただし、互いに表示態様が同じものであっても、出現するキャラクタの大きさ、出現数、移動速度などをキャラクタ A と B とで各々異ならせるようにして、遊技者に各々が区別できるようにしているものであれば、特に限定はされない。

30

【0740】

本実施形態では、前半変動 (一部の識別図柄の変動が停止してリーチ状態が発生する前の状態) でキャラクタが出現するようになっている。その後、左右の識別図柄が停止し、リーチ状態が発生する (画面 (n))。このとき、スーパーリーチ若しくは通常のリーチが発生する。なお、このリーチ状態のときに、前述のキャラクタが出現するようによい。

40

【0741】

なお、本実施形態では、入賞口 SW 監視処理 (図 29 のステップ 2917) において、普図始動 SW 31A による遊技球の検出があったときに、普図乱数カウンタ値 (図 15 の普図乱数生成領域) の値を普図始動入賞記憶領域 (図 15) に記憶しているが、普図変動表示ゲームに関する乱数であれば、別の乱数を記憶するようによい。例えば、乱数生成回路 608 にて普図変動表示ゲームに関する乱数を生成 (ハード乱数を生成) しておき、普図始動 SW 31A による遊技球の検出があったときに、この生成した乱数を普図始動入賞記憶領域 (図 15) に記憶するようによい。

【0742】

50

また、本実施形態では、特図始動口SW共通処理(図40)のステップ4004~4006の処理において、図15に示される大当り乱数、大当り図柄乱数、第1~第3変動パターン乱数の各生成領域から乱数値を取得しているが、特図変動表示ゲームに関する乱数であれば、別の乱数を取得するようにしてもよい。例えば、乱数生成回路608にて、大当り乱数、大当り図柄乱数、第1~第3変動パターン乱数の全部又は一部を生成(ハード乱数を生成)しておき、ステップ4004~4006の処理の実行時に、この生成した乱数を取得するようにしてもよい。

**【0743】**

ここで、本実施形態において、割込み処理が呼び出された場合に、スタック領域に、フラグレジスタの値と、戻りアドレス値が格納される順序について説明する。

10

**【0744】**

まず、図51Aは、本発明の第1の実施の形態とは対照的に、割込み処理が呼び出された場合に、先に戻りアドレス値をスタック領域に格納し、その後、フラグレジスタの値をスタック領域に格納することを仮定した場合の図である。この図51Aの(A)~(C)により、割込み処理ルーチンの実行過程において、スタック領域に退避されたデータがどのように変化するかを説明する。

**【0745】**

なお、図51Aの「FLG」は1バイトのフラグレジスタの値を示している。また、「ADR\_H」は戻りアドレスの上位バイトの値を示し、「ADR\_L」は戻りアドレスの下位バイトの値を示しており、「ADR\_H」と「ADR\_L」を組み合わせることで、2バイトの戻りアドレス値が構成される。

20

**【0746】**

まず、図51Aの(A)は、割込み処理が発生して、戻りアドレス値及びフラグレジスタの値が、スタック領域に格納された直後の状態である。次いで、割込み処理ルーチン内で「PUSH」や「CALL」の命令が実行されると、図51Aの(B)に示すように、スタック領域に別の退避データが蓄積される。

**【0747】**

次いで、以降の割込み処理ルーチン内で「POP」命令等が実行されると、蓄積された退避データが徐々に復帰し、割込み処理ルーチンの最後の段階になると図51Aの(A)の状態に戻ることになる。そして、割込み処理ルーチンの最後で「RETI」命令を実行すると、スタック領域から「FLG」の値がフラグレジスタに復帰して格納されるとともに、「ADR\_H」と「ADR\_L」の戻りアドレス値が復帰してプログラムカウンタに格納されることで、割込信号が発生した時点に対応する呼出元ルーチンのアドレスに処理が移り、以降の処理を継続することになる。

30

**【0748】**

なお、「RETI」命令が実行されるまでは、この割込み処理ルーチンの実行中でのスタックポインタの値は、フラグレジスタの値が格納されている領域のアドレス(図では29F9H)よりも、常に小さな値(2980H~29F8H)の範囲で増減する。

**【0749】**

ただし、割込み処理ルーチンのプログラムにバグ等があるときは、スタックポインタの値が想定している範囲を外れてしまう恐れがある。例えば「POP」命令が、プログラム開発者が意図している回数よりも1回だけ余分に実行されると、スタックポインタの値が、フラグレジスタの値が格納されている領域のアドレス(29F9H)を一時的に指し示してしまうことになる。そして、この状態で「PUSH」命令を1回だけ実行すると、29F9Hのアドレスに退避されている「FLG」の値に、別の値が上書きされてしまうことになる。

40

**【0750】**

このようなバグが存在する場合は、割込み処理ルーチンの最後で「RETI」命令を実行する段階になったとき、図51Aの(C)に示すように、29FAH~29FBHのアドレスに退避された「ADR\_H」と「ADR\_L」を書き換えることなく、29F9H

50

のアドレスに退避された「FLG」のみを異常な値に書き換えるという状態が発生する恐れがある。

【0751】

この場合、復帰後のフラグレジスタに格納される値が異常値になっているため、以降の処理が正常に動作せずに障害が発生することが考えられる。しかしながら、プログラムカウンタに復帰する戻りアドレス値（「ADR\_H」と「ADR\_L」）は正常な値であるので、割り込み処理ルーチンの最後で「RETI」命令を実行すると、プログラムが暴走することなく呼出元のルーチンに戻って処理を継続することになる。

【0752】

そのため、遊技機の開発期間中では、開発者がプログラム中のバグの存在に気づかない可能性がある。仮に、障害の発生に気づいても、プログラムが暴走しない可能性が高いが故に、障害の発生要因を特定することが困難になるおそれがある。

【0753】

一方、図51Bは、本発明の第1の実施の形態と同様に、割り込み処理が呼び出された場合に、先にフラグレジスタの値をスタック領域に格納し、その後、戻りアドレス値をスタック領域に格納した場合の図である。この図51Bの(A)～(C)により、割り込み処理ルーチンの実行過程において、スタック領域に退避されたデータがどのように変化するかを説明する。なお、図51Bの「FLG」、「ADR\_H」、「ADR\_L」も、図51Aと同様の構成である。

【0754】

まず、図51Bの(A)は、割り込み処理が発生して、戻りアドレス値及びフラグレジスタの値が、スタック領域に格納された直後の状態である。次いで、割り込み処理ルーチン内で「PUSH」や「CALL」の命令を実行すると、図51Bの(B)に示すように、スタック領域に別の退避データが蓄積される。

【0755】

次いで、以降の割り込み処理ルーチン内で「POP」命令等が実行されると、蓄積された退避データが徐々に復帰し、割り込み処理ルーチンの最後の段階になると図51Bの(A)の状態に戻るようになる。そして、割り込み処理ルーチンの最後で「RETI」命令を実行すると、スタック領域から「FLG」の値がフラグレジスタに復帰して格納されるとともに、「ADR\_H」と「ADR\_L」の戻りアドレス値が復帰してプログラムカウンタに格納されることで、割り込み信号が発生した時点に対応する呼出元ルーチンのアドレスに処理が移り、以降の処理を継続することになる。

【0756】

なお、「RETI」命令が実行されるまでは、この割り込み処理ルーチンの実行中でのスタックポインタの値は、戻りアドレス値が格納されている領域のアドレス（図では29F9H）よりも、常に小さな値（2980H～29F8H）の範囲で増減する。

【0757】

このとき、前述の図51Aと同様に、割り込み処理ルーチンのプログラムにバグ等があると、スタックポインタの値が想定している範囲を外れてしまう恐れがある。この場合、スタック領域に待避されている「ADR\_L」、「ADR\_H」、「FLG」などの値が別の値が上書きされてしまう恐れがあり、割り込み処理ルーチンの最後で「RETI」命令を実行したときに、フラグレジスタの値が正常に復帰しなくなってしまう。

【0758】

しかしながら、前述の図51Aとは異なり、図51Bの構成であれば、スタック領域に退避された「ADR\_H」と「ADR\_L」の値を書き換えずに、スタック領域に退避された「FLG」のみを書き換えることは、あり得ない。その理由は、図51Bの(C)に示すように、スタック領域へのデータの出し入れの順序に特徴があるが故に、「FLG」が異常値に書き換えられるときには、その上方に待避されている「ADR\_H」と「ADR\_L」の値も、書き換えられていると考えられるからである。

【0759】

10

20

30

40

50

このようにして、割込み処理ルーチンの最後で図5 1 Bの(C)の状態となり、この状態で「RETI」命令を実行すると、29F9H~29FAHのアドレスに退避された異常なアドレス値がプログラムカウンタに格納されるので、呼出元のルーチンに戻れず、プログラムが暴走することになる。

【0760】

すなわち、本実施形態と同様の図5 1 Bの構成であれば、復帰後のフラグレジスタに格納される値が異常値になるような場合には、「RETI」命令実行後のプログラムカウンタにも異常なアドレス値が必ず格納される。そのため、割込み処理ルーチンにバグがあれば、遊技機の開発期間中にプログラムが暴走して、開発者が障害の発生に気づきやすい構成となるのである。従って、図5 1 Aの構成よりも、プログラムのバグが発見しやすい遊技機を実現することが可能となる。

10

【0761】

次に、本実施形態において、割込み処理が呼び出された場合にスタック領域に格納される、フラグレジスタの値のバイト数と、戻りアドレス値のバイト数との関係について説明する。

【0762】

まず、図5 1 Cは、本発明の第1の実施の形態とは対照的に、フラグレジスタの値のバイト数と、戻りアドレス値のバイト数とを、ともに2バイトで構成することを仮定した場合の図である。この図5 1 Cの(A)~(D)により、割込み処理の開始直後及び終了直前におけるスタック領域内の退避データの状態と、CALL命令により呼び出されたサブルーチンの開始直後及び終了直前におけるスタック領域内の退避データの状態について説明する。

20

【0763】

なお、図5 1 Cの「FLG」は2バイトのフラグレジスタの値を示している。また、「ADRW\_H」と「ADRW\_L」は、当該割込み処理の発生時に格納される戻りアドレス値の上位バイトと下位バイトの値を示している。また、「ADR2\_H」と「ADR2\_L」、及び「ADR1\_H」と「ADR1\_L」は、当該割込み処理が発生するよりも前に格納されたアドレス値の上位バイトと下位バイトである。なお、「ADR3\_H」と「ADR3\_L」は、割込み処理が発生することに替えて、CALL命令によるサブルーチンが開始された場合に格納される戻りアドレス値の上位バイトと下位バイトの値を示している。

30

【0764】

まず、図5 1 Cの(A)は、割込み処理が発生して、戻りアドレス値及びフラグレジスタの値が、スタック領域に格納された直後の状態である。この割込み処理の最後のステップにおいては、「RETI」命令が実行されることで、スタック領域から2バイトのフラグレジスタの値と、2バイトの戻りアドレス値が復帰する。なお、スタック領域へのデータの格納状況に対応してスタックポインタの値も増減するようになっており、「RETI」命令実行後に呼出元の処理に復帰した際には、スタック領域は図5 1 Cの(B)に示す状態となっている。

【0765】

一方、割込み処理が発生するのではなく、呼出元の処理でCALL命令を実行してサブルーチンを読み出した場合は、図5 1 Cの(C)に示すように、戻りアドレス値のみがスタック領域に格納され、フラグレジスタの値は格納されない。そして、このサブルーチンの最後のステップにおいては、「RET」命令が実行されることで、スタック領域から2バイトの戻りアドレス値のみが復帰する。なお、スタック領域へのデータの格納状況に対応してスタックポインタの値も増減するようになっており、「RET」命令実行後に呼出元の処理に復帰した際には、スタック領域は図5 1 Cの(B)に示す状態となっている。

40

【0766】

ただし、CALL命令で呼び出したサブルーチンのプログラムにバグがあり、サブルーチンの最後で、「RET」命令の代わりに、誤って「RETI」命令を実行してしまった

50

場合は、図51Cの(C)に示す状態から「ADR3\_\_H」と「ADR3\_\_L」の値を戻りアドレス値としてプログラムカウンタに復帰させるだけでなく、「ADR2\_\_H」と「ADR2\_\_L」の値を2バイトのフラグレジスタの値と見なして復帰させてしまうことになる。そのため、「RETI」命令実行後に呼出元の処理に復帰した際には、スタック領域は図51Cの(D)に示す状態となってしまう。

【0767】

その後、呼出元の処理でさらに「RET」命令を実行すると、図51Cの(D)の状態から、「ADR1\_\_H」と「ADR1\_\_L」の戻りアドレス値を取得して、プログラムカウンタに復帰させ、そのアドレスに移行してしまうことになる。これは、呼出元の処理で「RET」命令を実行した場合に取得されるべき本来の戻りアドレス値(図51Cの(B)の「ADR2\_\_H」と「ADR2\_\_L」)とは異なるアドレスに、処理が移行することに相当する。

10

【0768】

この場合、本来実行すべき処理とは異なる処理が行われるため、何らかの障害が発生するはずであるが、誤って取得した「ADR1\_\_H」と「ADR1\_\_L」の戻りアドレス値は、プログラム上にアドレス値として使用されている値である。故に、プログラムが暴走することなく呼出元のルーチンに戻って処理を継続することになる。

【0769】

そのため、遊技機の開発期間中では、開発者がプログラム中のバグの存在に気づかない可能性がある。仮に、障害の発生に気づいても、プログラムが暴走しない可能性が高いが故に、障害の発生要因を特定することが困難になるおそれがある。

20

【0770】

一方、図51Dは、本発明の第1の実施の形態と同様に、フラグレジスタの値のバイト数と、戻りアドレス値のバイト数とを、異ならせて構成した場合の図である。この図51Dの(A)~(C)により、割込み処理の開始直後及び終了直前におけるスタック領域内の退避データの状態と、CALL命令により呼び出されたサブルーチンの開始直後及び終了直前におけるスタック領域内の退避データの状態について説明する。

【0771】

なお、図51Dの「ADR1\_\_H」、「ADR1\_\_L」、「ADR2\_\_H」、「ADR2\_\_L」、「ADR3\_\_H」、「ADR3\_\_L」も、図51Cと同様の構成である。ただし、図51Dの「FLG」が1バイトで構成されている点は、図51Cと異なっている。

30

【0772】

まず、図51Dの(A)は、呼出元の処理でCALL命令を実行してサブルーチンを呼び出した場合において、戻りアドレス値がスタック領域に格納された直後の状態である。この場合は、戻りアドレス値のみがスタック領域に格納され、フラグレジスタの値は格納されない。そして、このサブルーチンの最後のステップにおいては、「RET」命令が実行されることで、スタック領域から2バイトの戻りアドレス値のみが復帰する。なお、スタック領域へのデータの格納状況に対応してスタックポインタの値も増減するようになっており、「RET」命令実行後に呼出元の処理に復帰した際には、スタック領域は図51Dの(B)に示す状態となっている。

40

【0773】

ただし、CALL命令で呼び出したサブルーチンのプログラムにバグがあり、サブルーチンの最後で、「RET」命令の代わりに、誤って「RETI」命令を実行してしまった場合は、図51Dの(A)に示す状態から「ADR3\_\_H」と「ADR3\_\_L」の値を戻りアドレス値としてプログラムカウンタに復帰させるだけでなく、「ADR2\_\_L」の値を1バイトのフラグレジスタの値と見なして復帰させてしまうことになる。そのため、「RETI」命令実行後に呼出元の処理に復帰した際には、スタック領域は図51Dの(C)に示す状態となってしまう。

【0774】

その後、呼出元の処理でさらに「RET」命令を実行すると、図51Dの(C)の状態

50

から、「ADR1\_\_L」と「ADR2\_\_H」の値を戻りアドレス値として取得してプログラムカウンタに格納し、さらにプログラムカウンタが示す場所へ移行してしまうことになる。これは、呼出元の処理で「RET」命令を実行した場合に取得されるべき本来の戻りアドレス値(図51Dの(A)の「ADR2\_\_H」と「ADR2\_\_L」)とは異なるアドレスに、処理が移行することに相当する。

【0775】

この場合、誤って取得したアドレスのうち、「ADR2\_\_H」がアドレスの下位バイトと認識され、「ADR1\_\_L」がアドレスの上位バイトと認識されるので、実際には正しい組み合わせになっていないアドレスが生成されることになり、プログラム上にアドレス値として使用されていないアドレス値になると考えられる。そのため、以降に「RET」命令を実行したときに、プログラムカウンタには異常なアドレス値が格納されることになり、それ故にCPUが暴走する可能性が高くなる。そのため、割込み処理ルーチンにバグがあれば、遊技機の開発期間中にプログラムが暴走して、開発者が障害の発生に気づきやすい構成となるのである。従って、図51Cの構成よりも、プログラムのバグが発見しやすい遊技機を実現することが可能となる。

10

【0776】

以上のように構成することによって、フラグレジスタ及び戻りアドレスをスタック領域に退避させる場合にノイズ等によってスタック領域に格納されたデータが書き換えられて障害が発生した場合であっても、障害を早期に発見可能となるため、障害発生時の原因特定が容易になり、開発効率を向上させることが可能となる。

20

【0777】

ここで、本発明とは対照的な従来技術と本発明との対比を行う。

【0778】

まず、特許文献Aとして特開2009-183500号公報に開示されるような遊技機が知られており、この遊技機では、従来よりもプログラムコードを削減し、記憶手段の記憶容量を削減するために、CPUからI/O空間を介してアクセス可能な記憶領域を少なくとも有する記憶手段を備えている。

【0779】

そして、この遊技機では、特許文献Aの段落[0137]~[0142]に記載があるように、LD命令、IN命令、OUT命令を適宜使い分けることで、プログラム全体のステート数やプログラムコード量を削減している。

30

【0780】

さらに、この遊技機では、特許文献Aの図13に記載があるように、指定したRAM領域からレジスタにデータを格納するLD命令や、逆に、レジスタから指定したRAM領域にデータを格納するLD命令を用いている。

【0781】

具体的には、RAM領域の2810H番地のデータをAレジスタに格納するための「LD A, (2810H)」、AレジスタのデータをRAM領域の2810H番地に格納するための「LD (2810H), A」、RAM領域の2810H番地のデータをHLレジスタに格納するための「LD HL, (2810H)」、HLレジスタのデータをRAM領域の2810H番地に格納するための「LD (2810H), HL」などが該当する。

40

【0782】

これらの命令は、特許文献Aの図13にも記載があるが、3バイトのプログラムコード量を要する命令である。なお、プログラムコードとは、任意の命令を、命令コード部(オペコード部)とアドレス部(オペランド部)との組み合わせを用いて所定バイト数の数値で表現したものである。

【0783】

命令コード部は、命令同士を識別するために必要なコードであり、例えば、「LD A, (2810H)」のような、指定されたRAM領域のデータをAレジスタに格納するた

50

めのLD命令であれば、「3AH」などの1バイトの値が割り当てられている。一方、アドレス部はRAM領域のアドレスを指定するためのコードであり、例えば、2810H番地へのアクセスを指定する場合は、「28H」「10H」という2バイトの値がプログラムにて設定される。

【0784】

このようなLD命令は、遊技プログラムにて頻繁に使用されるものであるから、プログラム全体のコード量を削減するためには、命令をできるだけ少ないバイト数で表現できることが好ましいのは言うまでもない。しかしながら、RAM領域のアドレスを指定する際には、アドレス部に少なくとも2バイトのコードを割り当てる必要がある上に、命令同士を識別するための命令コード部のコードが少なくとも1バイト必要となるため、このようなLD命令を表現するためのプログラムコードは、2バイト以下に削減することは不可能である。

10

【0785】

そこで本発明は、上記のような問題点を解決するために、遊技制御プログラム全体のコード量を削減することを目的としている。

【0786】

例えば、本発明の第1の実施の形態によれば、上位アドレス指定レジスタを用いることにより、RAM（更新情報記憶手段）にアクセスするための命令を、少ないバイト数のコードで実現することが可能となる。そのため、遊技制御プログラムに当該命令を用いることで、遊技制御プログラム全体のコード量を削減することが可能となる。

20

【0787】

なお、上位アドレス指定レジスタ自体は1バイトのレジスタであるので、特定アドレスの前後各128バイト（又は前後各127バイト）の範囲内からデータを取り出すことを目的として、当該特定アドレスを指定するためにわざわざ2バイトのレジスタを用いるような方法と比較すると、アドレス指定のためのレジスタ構成が簡素化されている。

【0788】

また、Kレジスタ（上位アドレス指定レジスタ）に固有のアドレス値（“28H”）を設定すること（固有値設定手段）によって、リセット（起動）信号が発生した際に、CPU（演算処理手段）によってKレジスタに固有のアドレス値を設定する処理を行う必要がなくなる。したがって、リセット後、遊技制御プログラムでKレジスタに固有のアドレス値を設定しなくても、RAMにデータを読み書きすることが可能となる。さらに、CPUの処理負担が軽減され、遊技制御プログラムのコード量を削減することも可能となる。

30

【0789】

また、RAMに記憶された情報の更新を規制するRAMアクセス規制回路を備え、所定の起動信号が発生するとRAMアクセス規制回路によりRAMの更新禁止状態となり、その後、CPUコアの動作が開始されてRAMの更新禁止状態が解除され、次いで、上位アドレス指定レジスタにより指定されるアドレスに記憶された情報を用いて遊技制御が開始されるので、誤ったタイミングでRAMの値が更新されることを防止できる。

【0790】

さらに、本発明の第1の実施の形態によれば、始動記憶などを保持するワークエリアと、スタック領域とが異なる上位アドレスとなる領域に保持されるため、乱数更新時などに誤ってスタック領域に格納されたデータを更新することを防止することができるため、誤動作を防ぐことができる。

40

【0791】

特に、始動記憶を保持する領域（例えば、図16、図17、図53A、図53Bに例示するような乱数記憶領域、あるいは図15に例示する普図乱数保存領域）は、変動表示ゲームを開始する際に、格納されている乱数がシフトされることになるが、これらの領域の上位アドレスはすべて同一（28H）になるように設定されているので、シフトの処理を行う際に上位アドレス指定レジスタの値を変更する必要がなく、処理が簡素化される。

【0792】

50



また、ワークエリアの下位アドレスが一致する領域であって、上位アドレスがスタック領域と同じ領域を使用禁止領域に設定することによって、上位アドレスの値を間違っている状態で始動記憶を更新しようするとCPUがリセットされ、誤作動を防止することができる。

【0793】

次に、特許文献Bとして特開平10-033804号公報に開示されるようなパチンコ機が知られており、パチンコ機の制御装置のプログラムにおいては、遊技状態に対応して異なる命令が実行されるように、プログラム内部で分岐の処理が行われている。この特許文献Bでは、段落[0072]~[0073]に記載があるように、プロセスフラグの値に応じて制御を分岐させる処理がなされ、その分岐の際には、基本回路30のROMに記憶されているプロセスジャンプテーブル(特許文献Bの図13)を参照して行なわれる。

10

【0794】

一方、パチンコ機の制御装置などに用いられるCPUとして、特許文献Cとして特開平11-232099号公報に開示されるZ80系のCPUが知られている。Z80系の命令には、特許文献Cの段落[0020]に示されるような「JP(HL)」の命令が用意されている。この命令は、HLレジスタに格納された値をアドレスと認識させて、そのアドレスで示されるプログラムの箇所に処理を分岐させるものである。なお、CPUが起動した時点では、HLレジスタの値は不定な値が格納されている。

【0795】

特許文献Bの従来技術の遊技機のプログラムにて分岐の処理を行う場合には、特許文献CのCPUの「JP(HL)」が用いられることがあり、この場合、HLレジスタに分岐先のアドレスの値を格納してから、「JP(HL)」の命令を実行する構成となる。

20

【0796】

しかしながら、特許文献Bの従来技術の遊技機などは、ノイズ環境に晒されているために、CPUが暴走する危険を有している。特に、特許文献Cのような従来のCPUを用いた場合は、遊技機の電源が投入されてCPUが起動を開始した直後では、HLレジスタの値が不定な値であるために、誤ってプログラム中の「JP(HL)」命令が実行された場合には、分岐先が不定なためにCPUが暴走するおそれがあった。

【0797】

そこで、本発明は、上記のような問題点を解決するために、ノイズ環境でも正常にCPUを動作させることを目的としている。

30

【0798】

例えば、本発明の第1の実施の形態によれば、固有値設定手段(図12の初期値設定回路)を備えており、起動信号(リセット信号)が発生すると、固有値設定手段により分岐アドレス設定レジスタに固有のアドレス値が設定されるので、遊技制御プログラム開始直後にノイズ等が発生して、誤って特定の命令を実行してしまった場合であっても、固有のアドレス値で示されるアドレスに処理を分岐させることができる。そのため、遊技制御プログラム開始直後にノイズ等が発生しても、演算処理手段の暴走を防止することができる。

【0799】

すなわち、固有のアドレス値で示されるアドレス(例えば、遊技制御プログラムのリセットアドレス)には、演算処理手段が暴走したときに実行すべきプログラムの命令(リセット処理等の命令)が予め配置されているので、遊技制御プログラム開始直後にノイズ等が発生しても対処することが可能となる。

40

【0800】

また、本発明の第1の実施の形態では、分岐アドレス設定レジスタを用いて、特図変動表示ゲームの制御状態(ゲーム処理番号に対応する各遊技状態)に対応した分岐処理を行っているので、制御状態が順次切り替わっても、その都度、切り換えられた制御状態に対応する分岐先の処理が行えるように、効率的な遊技制御プログラムが構成できるようになっている。なお、普図変動表示ゲームの制御状態に対応した分岐処理も、分岐アドレス設

50

定レジスタを用いて行うようにしてもよい。

【0801】

さらに、本発明の第1の実施の形態によれば、電源投入時の際に、遊技制御プログラムが開始される前に、初期値設定回路（固有値設定手段）1241によってCPUコア102に備えるスタックポインタに初期値がハード的に設定される。そのため、遊技制御プログラムの実行が開始された直後にノイズ等が原因で割込処理が発生しても、呼出元のルーチンの戻りアドレスをスタック領域に確実に格納することができ、割込処理から呼出元のルーチンに復帰する際にも問題なく処理を継続することができる。

【0802】

また、分岐アドレス設定レジスタ（HLレジスタ）にリセットアドレス値が格納された状態で、特定の命令（JP（HL））が実行された場合であっても、スタックポインタの値を設定できる。

【0803】

なお、遊技制御中に異常が発生して、リセット回路610B（セキュリティ回路630）によりCPUコア102がリセットされるような場合であっても、CPUコア102の処理がリセットアドレスに戻される。

【0804】

この場合、初期値設定回路1241によってスタックポインタに初期値がハード的に設定されなくても、リセットアドレスからの命令を実行した直後に、ソフトウェアによりスタックポインタに初期値を設定する（図25の遊技制御装置メイン処理のステップ2502A）ので、あらゆるリセット状態に対応してスタックポインタに初期値を設定することが可能となる。なお、同様の理由により、上位アドレス指定レジスタにもソフトウェアによって初期値を設定している（図25の遊技制御装置メイン処理のステップ2502B）。

【0805】

さらに、遊技制御プログラムは、呼出元のルーチンに記述されたCALL命令等の呼出の命令が実行された場合（第1の呼出条件が成立した場合）や、呼出元のルーチンの命令実行中に所定の割込信号が発生した場合（第2の呼出条件が成立した場合）には、呼出元のルーチンに含まれる戻りアドレスの値をスタック領域に格納して、呼出先の処理に移行するような構成となっている。

【0806】

この場合において、後者の所定の割込信号は、呼出元のルーチンの実行中のアドレスとは無関係に発生するものであるため、割込信号が発生した時点におけるフラグレジスタ1200の値は、呼出先のルーチンの処理中で破壊されないように、スタック領域に退避しておくことが好ましい。

【0807】

そこで、後者の第2の呼出条件が成立した場合は、図19の（D）に示すように、呼出元のルーチンに含まれる戻りアドレスの値だけでなく、フラグレジスタ1200の値もスタック領域に格納して、呼出先の処理に移行させている。そのため、後者の第2の呼出条件が成立した場合には、呼出先のルーチンのプログラム中にフラグレジスタをスタック領域に格納するための命令（「PUSH FLAG」）を記述しなくても、フラグレジスタの値が自動的にスタック領域に格納されるため、プログラムのコード量を節約することができる。

【0808】

これに対して、前者のように、呼出元のルーチンに記述されたCALL命令等の呼出の命令を実行する場合は、呼出元のルーチンで設定されたフラグレジスタ1200の値を呼出先のルーチンの処理中で破壊してはいけない場合もあるが、呼出先のルーチンの処理中で設定されたフラグレジスタ1200の値を、呼出元のルーチンに戻して演算に利用することもあり得る。よって、呼出元のルーチンで設定されたフラグレジスタ1200の値をスタック領域に退避すべきか否かはケースバイケースであることから、呼出先のルーチン

10

20

30

40

50

のプログラム中に、プログラム中にフラグレジスタ1200の値をスタック領域に格納する命令（例えば、「PUSH FLG」）を記述して、値を退避するか否かを決定したほうが都合が良い。

【0809】

故に、前者の第1の呼出条件が成立した場合は、図19の(C)に示すように、呼出元のルーチンに含まれる戻りアドレスの値はスタック領域に格納するものの、フラグレジスタ1200の値を自動的にスタック領域に格納することはない。

【0810】

なお、図19の(D)及び図51Bに示すように、スタック領域には、先にフラグレジスタの値を格納して、後から戻りアドレスの値を設定しているため、戻りアドレスの値を破壊することなくフラグレジスタの値のみを破壊するような事態は発生しない。そのため、図51Aで例示した構成の遊技機のように、呼出先の割込処理ルーチンでフラグレジスタの値が破壊された状態のまま、スムーズに呼出元のルーチンに戻って処理を継続してしまうことはない。

【0811】

また、図19の(D)及び図51Dに示すように、スタック領域に格納される戻りアドレスの値は2バイト構成であるのに対して、スタック領域に格納されるフラグレジスタの値は1バイト構成となっている。このように、スタック領域に格納される戻りアドレスとフラグレジスタとのバイト数を異ならせることで、呼出先のルーチンで誤ってフラグレジスタの値を戻りアドレスとして復帰させた場合には、図51Cで例示した構成の遊技機のような不具合が発生することがなく、以降の処理では戻りアドレスの取得が不可能となるので、プログラムのバグ等を発見しやすい。

【0812】

さらに、図21及び図48(C)に示すように、「LD RBS, 1」の命令（又は「LD RBS, 0」の命令）を実行することによって、汎用レジスタのバンクを切り替える機能を有しているため、呼出先のルーチン内でバンクを切り替えることにより、呼出元とは異なるレジスタを活用することができる。

【0813】

なお、図14に示したように、第1の実施の形態では、ユーザワークRAM104において上位アドレスの境界、すなわち、上位アドレスが「28H」の領域と「29H」の領域との境界を連続してアクセスすることが可能となっていたが、この境界に使用禁止領域を設けるようにしてもよい。

【0814】

さらに、特許文献Dとして特開2009-142566号公報に開示されるようなパチンコ機が知られており、遊技機を制御するためのプログラムの容量削減を図るために、パチンコ機の主制御基板において、CPUからIOにアクセスするためのイン/アウトコマンドが出力された時、エリア区分判定部が、アドレス値に応じて、アクセス先をIO、RAMのいずれかに切り換え、その結果に応じてIO識別信号又はチップセクタ信号をアクティブにする構成が開示されている。

【0815】

この特許文献Dの遊技機の構成では、イン/アウトコマンドを用いてRAMの一部（疑似RAM領域）にアクセス可能となり、この時、RAMの先頭アドレスを含む連続領域にワーク及び疑似RAM領域を割り当てるとともに、終端アドレスを開始点としてアドレスが小さくなる側に順次データが蓄積されるようにスタック領域を割り当てるように構成されている。

【0816】

一般に、スタックをRAMのどの領域に割り当てるかは、制御装置の設計者の裁量に任せられるが、設計の自由度が増すために都合が良い。そのため、CPUの起動時にはスタックポインタの初期値は不定な値にしておき、CPUがプログラムの実行を開始した後、ソフトウェアによりスタックポインタの初期値を設定する構成が好ましい。従って、こ

10

20

30

40

50

の特許文献Dの遊技機においても、CPUの起動時のスタックポインタの初期値は、不定な値となっているものと考えられる。

【0817】

しかしながら、スタックポインタの初期値が不定な状態で、ノイズ等により割込み信号が発生すると、不定な値で示された領域をスタック領域と見なしたままで、CPUが処理を行ってしまうおそれがあった。この場合でも、CPUは、割込み処理ルーチンに処理を移してしまうことになり、割込み処理ルーチンの最後にRETIなどの命令を実行しても、その時点でスタックポインタが指し示している領域には戻りアドレスの値が格納されている保障がないため、CPUが暴走するおそれもあった。その結果、想定外の大当たり等が発生するなどして正規な遊技を行えないおそれがあった。

10

【0818】

そのため、ノイズ環境に晒されているパチンコ機等の遊技機では、スタック領域を設定するための設計の自由度よりも、CPUが暴走しないように構成することを重視したほうがよいという課題があった。

【0819】

また、従来 of 遊技機においては、RAM領域内に割り当てられるスタック領域には、CPUが遊技プログラム実行中に所定のサブモジュールを呼び出した際に、呼び出し元のモジュールのアドレスの値が、戻りアドレス値として格納されるものがあった。そして、呼び出し先のサブモジュールの実行後に、スタック領域に格納してあった戻りアドレスの値をプログラムカウンタに戻すことで、CPUの処理が呼び出し元のモジュールに復帰するようになっていた。

20

【0820】

例えば、特許文献E（特開2008-080004号公報）に開示された遊技機では、特許文献Eの図22に示すように、RST命令を実行すると、呼び出し元のモジュールの戻りアドレスの値をスタックに格納して、呼び出し先のサブモジュールに移行する。その後、呼び出し先のサブモジュールの終了時にRET命令を実行することで、スタック領域に格納されていた戻りアドレスの値をプログラムカウンタに戻し、呼び出し元のモジュールに復帰する。なお、特許文献Eの段落[0148]に示すように、CALL命令を用いても同様の処理が行われる。

【0821】

また、特許文献Eでは、従来よりも、プログラムのメンテナンス性および処理速度を維持しつつ、不正改造抑制および不正改造発見容易性向上を実現するために、ROMの0008H番地から000FH番地には、変動停止コマンド送信処理のプログラムを格納し、ROMの0020H番地から0027H番地には、賞球数コマンド送信処理のプログラムを格納するとともに、これらのプログラムをRST命令を用いて呼び出すようにしている。

30

【0822】

なお、特許文献EのようにRST命令やCALL命令を実行しなくても、タイマ割込等の割込が発生した場合に、呼び出し元のモジュールの戻りアドレスの値をスタックに格納して、呼び出し先のサブモジュール（タイマ割込みの処理モジュール）に移行する処理を行う遊技機もある。

40

【0823】

例えば、特許文献F（特開2001-212330号公報）に開示された遊技機では、特許文献Fの段落[0071]～[0073]に示すように、タイマ割込等の割込が発生すると、呼び出し元のモジュールの戻りアドレスの値をスタックに格納して、呼び出し先のサブモジュール（タイマ割込みの処理モジュール）に移行する。その後、呼び出し先のサブモジュールの終了時にRETI命令を実行することで、スタックに格納してあった戻りアドレスの値をプログラムカウンタに戻し、呼び出し元のモジュールに復帰する。

【0824】

このようなタイマ割込みの処理を行うときは、特許文献G（特開2004-08947

50

6号公報)の図5や特許文献H(特開2006-068568号公報)の図6(c)に示すように、処理の先頭にて、フラグレジスタの値をAFレジスタとしてスタックに格納することが通常行われる。なお、このスタックに格納されたフラグレジスタの値は、特許文献Gの図5に示されるように、タイマ割込みの処理を終了してRETI命令を実行する際に元のフラグレジスタへ復帰するようになっている。

【0825】

しかし、呼び出し元のモジュールで用いられていたレジスタの値が、呼び出し先のサブモジュールで変更されてしまうと、CPUの演算処理に支障をきたしてしまう。そのため、従来の遊技機では、特許文献Gの図5のS11に示すように、処理の先頭でPUSH命令を用いてレジスタの値をスタック領域に退避する処理が通常行われ、処理の最後でPOP命令を用いてレジスタの値をスタック領域から復帰させる処理が行われる。このとき、特許文献Hの図6(c)に示すように、スタック領域には、様々なレジスタの値が一時的に格納される。そのため、呼び出し先のモジュール内にて、PUSH命令やPOP命令を記述する必要があり、プログラム容量が増えてしまう要因となっていた。

10

【0826】

このようなPUSH命令及びPOP命令を用いて、レジスタの値をスタックから出し入れする処理は、特許文献Eに開示されたタイマ割り込みの処理に限らず、特許文献Dに示したようなRST命令(あるいはCALL命令)を用いて呼び出されるサブモジュール内でも、頻繁に行われている。そのため、このようなサブモジュール内にもPUSH命令やPOP命令を記述する必要があり、プログラム容量が増えてしまう要因となっていた。

20

【0827】

なお、前者のタイマ割込みの処理では、フラグレジスタの値を無条件にスタックに格納する必要があったのに対して、後者のRST命令(あるいはCALL命令)を用いて呼び出されるサブモジュールの処理では、フラグレジスタを退避すべきか否かは、モジュールの目的によって一意に定まっていなかった。そこで、このような条件を満たしつつも、プログラム容量を削減するための工夫が必要とされていた。

【0828】

次に、図52は、本発明の第1の実施の形態の変形例のユーザワークRAM104の一例を示す図である。図52に示すユーザワークRAM104には、上位アドレスの境界(RAMのアドレスが「28FFH」から「2900H」に変化する境界線を含んだ領域)に使用禁止領域が設けられている。このように、上位アドレスが変化(「28H」から「29H」に変化)する境界に使用禁止領域を設けることによって、上位アドレスを変更せずに連続した領域にアクセスしようとするなどを防ぐことが可能となり、誤作動を防止することができる。

30

【0829】

例えば、このような境界線の前後のエリアに、Aレジスタ1202Aの値を一括して格納することを考える。この場合、上位アドレス指定レジスタであるKレジスタ1230の値を「28H」に固定した状態で、「INCL」命令(図21参照)を実行してLレジスタ1211Aの値を増加させながら、「LDK(L),A」の命令(図21参照)を実行させることになる。

40

【0830】

このとき「INCL」命令の実行により、Lレジスタ1211Aの値が「FFH」から「00H」に変化するタイミングがある。これは、Aレジスタ1202Aの値の格納先が境界線を跨ぐタイミングであるが、正しい格納先を指定するためには、以降のKレジスタ1230の値を「29H」に変更する必要がある。そのため、本来であれば、遊技プログラムにてKレジスタ1230の値を「29H」に変更する命令を実行しなければならないが、プログラム製作者が見落としてしまうおそれがある。

【0831】

なお、図52では、ワークエリアが、上位アドレスが「28H」と「29H」となる2つの領域に分散して設定されているが、ワークエリアを上位アドレス「28H」の領域に

50

限定してもよい。この場合であっても、遊技制御装置 100 の CPU 102 は、初期化処理を実行する際に、ユーザワーク RAM 104 の全域（上位のアドレスが互いに異なる領域、具体的には上位のアドレスが「28H」及び「29H」になる全ての領域）に対して、データの読み書きを行うことになる。これは、ワークエリア以外の領域（例えば、スタック領域など）でも、初期化を行う必要があるからである。

【0832】

そして、初期化処理が完了した時点で、上位アドレス指定レジスタである K レジスタ 1230 の値が「28H」に設定され、以後、停電処理が発生するまでの間は、K レジスタ 1230 の値を「28H」に維持したまま、CPU 102 によって遊技制御を継続する。

【0833】

このように構成することによって、K レジスタに設定された値を変更する頻度を大幅に低減させることができるので、レジスタの値を変更する処理の実行回数を削減することができる。したがって、プログラム容量をさらに削減することが可能となり、また、遊技制御処理の高速化を期待できる。

【0834】

（第2の実施の形態）

第1の実施の形態では、第2特図変動表示ゲームが第1特図変動表示ゲームよりも優先して実行されるように構成されていたが、第2の実施の形態では、第2特図変動表示ゲーム及び第1特図変動表示ゲームの区別なく、入賞した順序で特図変動表示ゲームが実行される。

【0835】

そこで、第1の実施の形態では、始動記憶に大当たり乱数、大当たり図柄乱数、変動パターン乱数が含まれていたが、第2の実施の形態では、さらに、第1始動入賞口37に遊技球が入賞したか、第2始動入賞口34に遊技球が入賞したかを区別する、すなわち、実行される特図変動表示ゲームが第1特図変動表示ゲームであるか、又は、第2特図変動表示ゲームであるかを区別するための特図識別 bit が始動記憶に含まれる。また、始動記憶の上限も実質的には8個となるので、ユーザワーク RAM 104 において、8ブロック分の乱数の保存領域を確保し、これを保留1～保留8として割り当てるものとする。

【0836】

図53A及び図53Bは、本発明の第2の実施の形態の特図変動表示ゲームを実行するために必要な各種乱数の保存領域を示す図であり、図53Aは保留1から4、図53Bは保留5から8に対応する各種乱数の保存領域を示す。

【0837】

本実施形態では、図53A及び図53Bに示すように、第3変動パターン乱数を格納する領域の先頭 bit を特図識別 bit とする。

【0838】

なお、図40の特図始動口SW共通処理を実行する際に、ステップ4004～4006で取得した各種乱数値を、ステップ4008～4010にて対応する乱数記憶領域に記させることになるが、対応する領域が第1の実施の形態とは異なっている。第2の実施形態においての対応する乱数記憶領域とは、第1特図保留カウンタと第2特図保留カウンタとの合算値に対応する領域である。

【0839】

例えば、第1始動入賞口37若しくは第2始動入賞口34に遊技球が入賞した結果、第1特図保留カウンタと第2特図保留カウンタとの合算値が「2」となった場合は、図53Aにおける「保留2の保存領域」に割り当てられた各種乱数の保存領域に、ステップ4004～4006で取得した各種乱数値を記憶することになる。このとき、第1始動入賞口37への入賞であれば「保留2の保存領域」の特図識別 bit には「0」が設定され、第2始動入賞口34への入賞であれば「保留2の保存領域」の特図識別 bit には「1」が設定される。

【0840】

10

20

30

40

50

同様に、第1始動入賞口37若しくは第2始動入賞口34に遊技球が入賞した結果、第1特図保留カウンタと第2特図保留カウンタとの合算値が「7」となった場合は、図53Bにおける「保留7の保存領域」に割り当てられた各種乱数の保存領域に、ステップ4004～4006で取得した各種乱数値を記憶することになる。このとき、第1始動入賞口37への入賞であれば「保留7の保存領域」の特図識別bitには「0」が設定され、第2始動入賞口34への入賞であれば「保留7の保存領域」の特図識別bitには「1」が設定される。

#### 【0841】

図54は、本発明の第2の実施の形態の特図保留情報判定処理の手順を示すフローチャートである。なお、第1の実施の形態の特図保留情報判定処理と共通の処理については、図41と同じ符号を割り当てて記載を省略する。

10

#### 【0842】

本発明の第2の実施の形態では、遊技制御装置100は、特図1と特図2とを区別せずに始動記憶を扱うので、第1の実施の形態におけるステップ4100～ステップ4102のような条件を設定せずに無条件で特図大当り判定処理(4103)を実行する。

#### 【0843】

図55は、本発明の第2の実施の形態の特図普段処理の手順を示すフローチャートである。なお、第1の実施の形態と共通の処理については、図45と同じ符号を割り当てて記載を省略する。

#### 【0844】

本発明の第2の実施の形態では、特図2始動記憶を優先して消化しない。そこで、まず、遊技制御装置100は、特図1及び特図2の保留数を合計し(5500)、合計保留数が0より大きいかが否か、すなわち、特図1又は特図2の保留があるかが否かを判定する(5501)。そして、保留がある場合には、ステップ5502において次の保留が特図1の保留であるかが否かを判定する。そして、保留が特図1である場合には、特図1変動開始処理(4503)を実行する。また、保留が特図1でなく特図2である場合には、特図2変動開始処理(4504)を実行する。

20

#### 【0845】

なお、特図1変動開始処理と特図2変動開始処理のいずれの処理を行ったかにかかわらず、図46の処理において、遊技制御装置100は、図53Aの保留1の保存領域を、次回の変動の対象として指定する(4600若しくは4603)。さらに、この指定された記憶領域から大当り乱数、大当り図柄乱数、第1～第3変動パターン乱数を取得した後、保留1～7個目に対応する保留乱数記憶領域(図53A及び図53B)に、保留2～8個目に対応する保留乱数記憶領域(次回以降の変動に対応する記憶領域:図53A及び図53B参照)の各乱数をシフトし(4601若しくは4604)、以降の処理(4602若しくは4605)の処理を実行することになる。

30

#### 【0846】

一方、ステップ5501の処理で合計保留数が0の場合には、遊技制御装置100は、ステップ4506以降の処理を実行する。

#### 【0847】

このように、本発明の第2の実施の形態によれば、第2特図変動表示ゲームを優先して実行しない場合であっても、第1の実施の形態と同様に、遊技制御プログラム全体のコード量を削減することが可能となる。

40

#### 【0848】

なお、今回開示した実施の形態における補助遊技は、例として、変動表示ゲームを開示しているが、これに限定されるものではない。例えば、遊技の実行中に遊技者にボタン等による選択操作をさせ、その操作入力に対応して遊技結果が導出されるような遊技も含まれる。あるいは、遊技の実行中に遊技球やメダルなどの遊技媒体が所定の領域に入賞するか否かの物理的な抽選を行わせ、その抽選結果に対応して遊技結果が導出されるような遊技も補助遊技に含まれる。この場合、遊技媒体が所定の領域に入賞することが「特別な結

50

果」となる。

【0849】

また、今回開示した実施の形態における遊技機には、パチンコ機以外の様々な遊技機が適用可能である。例えば、パチスロ機、アレンボール機、じゃん球遊技機などにも適用が可能である。

【0850】

なお、今回開示した実施の形態は、全ての点で例示であって制限的なものではない。また、本発明の範囲は前述した発明の説明ではなくて特許請求の範囲によって示され、特許請求の範囲と均等の意味及び内容の範囲での全ての変更が含まれることが意図される。

【産業上の利用可能性】

10

【0851】

以上のように、本発明は、レジスタを備えたプロセッサを含む遊技制御装置が遊技制御を行う遊技機に適用可能である。

【符号の説明】

【0852】

1	遊技機	
2	本体枠（外枠）	
3	前面枠	
5	遊技盤	
6	遊技装置	20
8	表示装置	
18	ガラス枠	
31	普通図柄始動ゲート	
32	一般入賞口	
33	普通変動入賞装置	
34	第2始動入賞口	
36	特別変動入賞装置	
37	第1始動入賞口	
51	遊技領域	
70	カードユニット	30
100	遊技制御装置	
101	遊技用マイコン	
102、601	CPUコア	
103、602	ユーザプログラムROM	
104、604	ユーザワークRAM	
150	演出制御装置（表示制御装置）	
160	電源装置	
161	バックアップ電源	
210	払出制御装置	
600	遊技用演算処理装置（アミューズチップ）	40
603	HWパラメータROM	
608	乱数生成回路	
609	クロック生成回路	
610A	割込制御回路	
610B	リセット回路	
611	アドレスデコーダ	
618	HPGプログラムROM	
619	IDプロパティメモリ	
621	HPGワークRAM	
629	分周回路	50

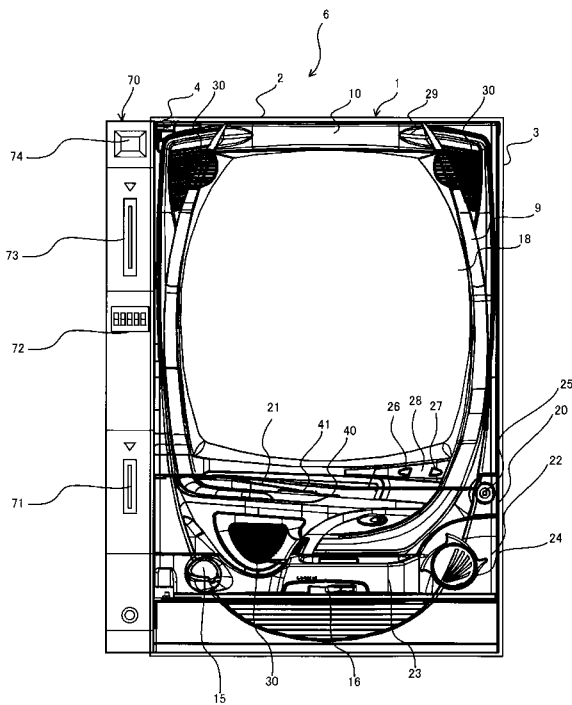


- 6 3 0     セキュリティ回路
- 6 4 0     RAMアクセス規制回路（更新規制手段）
- 6 4 1     フリップフロップ回路
- 6 4 2     ORゲート回路
- 6 5 0     アドレスバス
- 6 6 0     データバス
- 7 0 2     ワークエリア
- 7 0 5     使用禁止領域（アクセス禁止領域）
- 7 0 6     スタック領域
- 1 2 0 0    フラグレジスタ
- 1 2 0 3 A、1 2 0 3 B    WAレジスタ
- 1 2 0 6 A、1 2 0 6 B    BCレジスタ
- 1 2 0 9 A、1 2 0 9 B    DEレジスタ
- 1 2 1 2 A、1 2 1 2 B    HLレジスタ
- 1 2 2 0 A、1 2 2 0 B    汎用レジスタ群
- 1 2 3 0    Kレジスタ
- 1 2 3 3    スタックポインタ（SP）レジスタ
- 1 2 3 4    プログラムカウンタ（PC）レジスタ
- 1 2 4 0    内蔵リセット回路
- 1 2 4 1    初期値設定回路（固有値設定手段）
- 1 2 4 2    命令解釈実行回路
- 1 2 4 3    アクセス回路

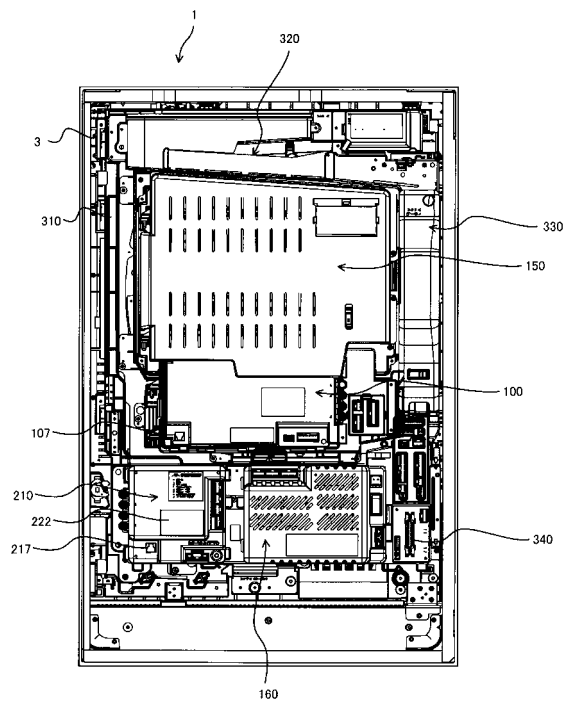
10

20

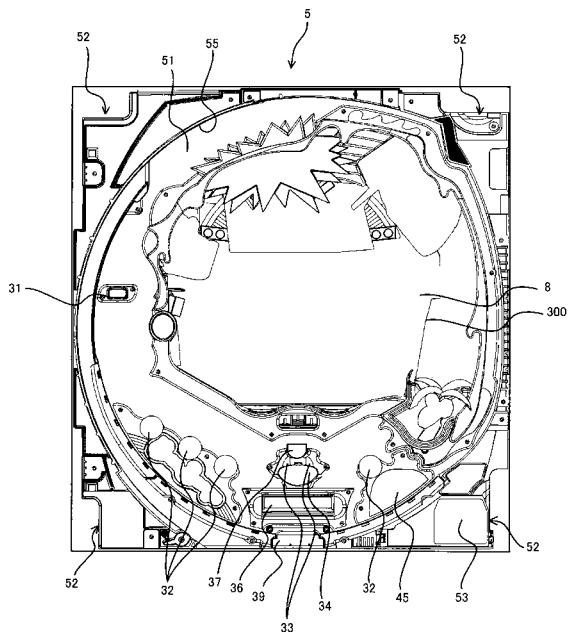
【 図 1 】



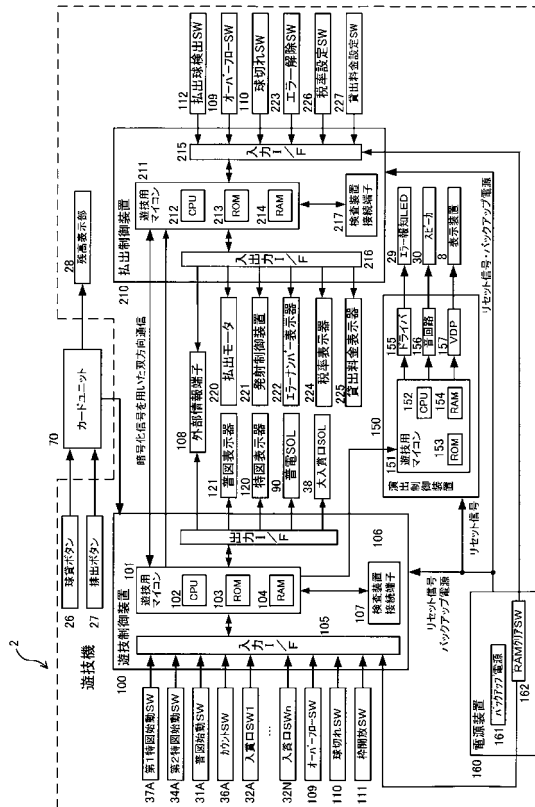
【 図 2 】



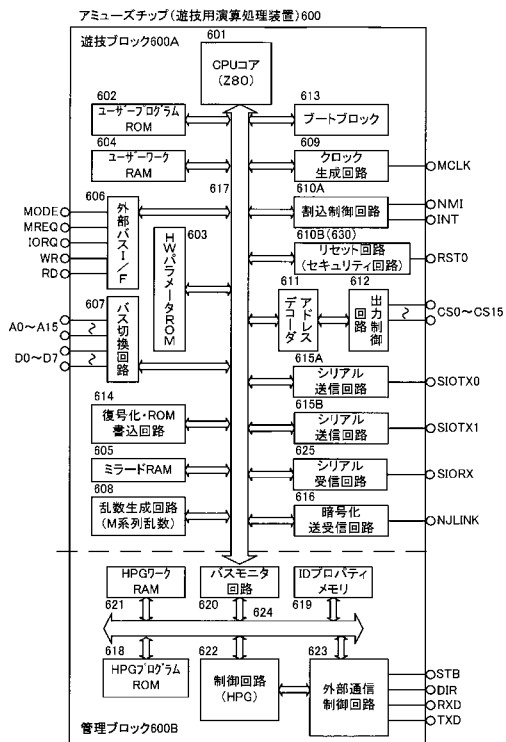
【図3】



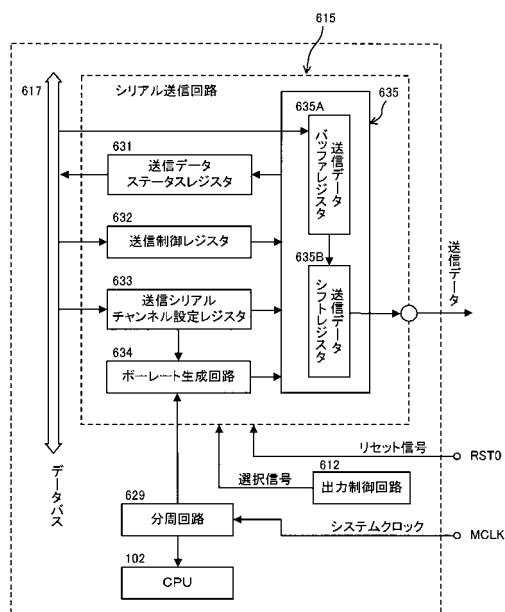
【図4】



【図5】



【図6】



【 図 7 】

シリアルチャンネル設定レジスタ 633

設定ビット	機能	設定	R/W
bit0~12	SIOxポーレート設定	0000h~1FFh: 分周比設定	R/W
bit13	SIOx送信モード	0: 8ビット 1: 9ビット	R/W
bit14	SIOxパリティ有無設定	0: パリティ無し 1: パリティ有り	R/W
bit15	SIOxパリティ有無設定	0: 偶数パリティ 1: 奇数パリティ	R/W

【 図 9 】

送信バッファステータスレジスタ 635

設定ビット	機能	設定	R/W
bit0~5	SIOx送信バッファステータス	00h: 送信データ無し 01h: 送信データ1バイト有り 02h: 送信データ2バイト有り ..... 20h: 送信データ32バイト有り	R
bit7	SIOx送信状態	0: 送信していない 1: 送信中	R

【 図 8 】

送信制御レジスタ 632

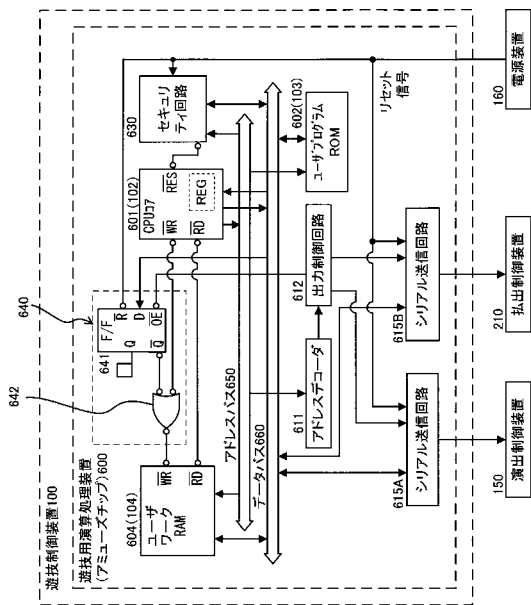
設定ビット	機能	設定	R/W
bit0	SIOx送信回路リセット	0: 何もしない 1: 送信回路を初期化	R/W
bit4	SIOxイネーブル	0: 送信禁止 1: 送信許可	R/W
bit5	SIOx送信割り込み要求	0: 割り込みを要求しない 1: 割り込みを要求する	R/W
bit6	SIOx送信割込状態	0: SIOx送信割り込み要求発生なし 1: SIOx送信割り込み要求発生あり	R
bit7	SIOxデータビット8の値	送信データのbit8 (9ビット時有効)	R/W

【 図 10 】

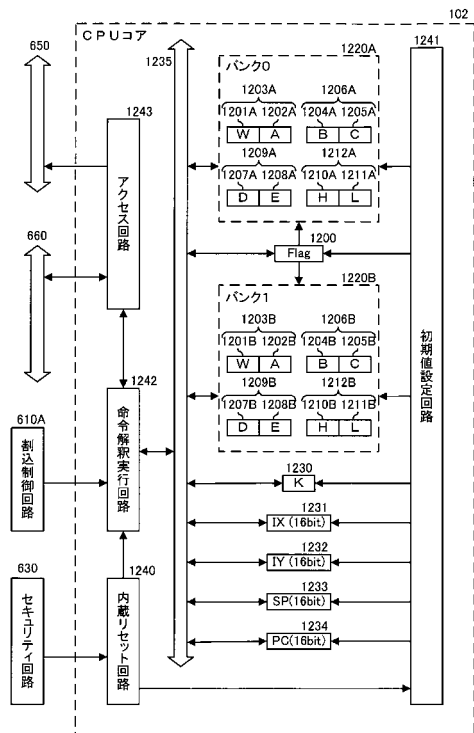
送信データレジスタ 636

設定ビット	機能	設定	R/W
bit0~7	SIOx送信バッファ	00h~FFh: 送信データ	W

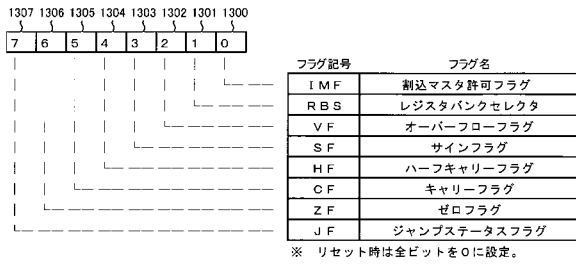
【 図 11 】



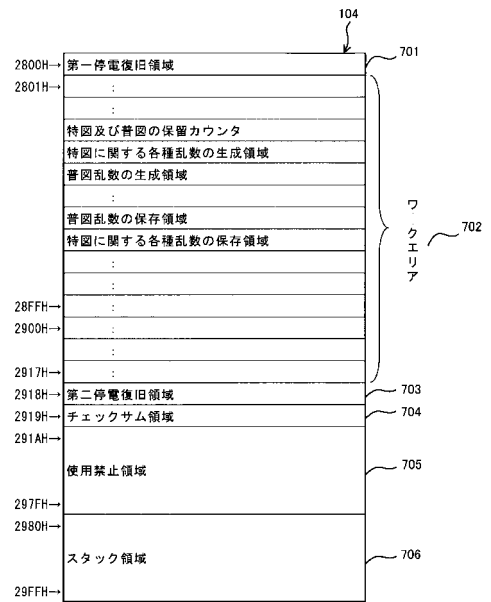
【 図 12 】



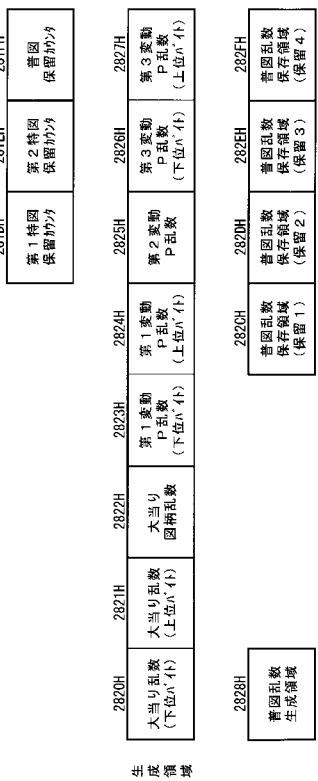
【図 13】



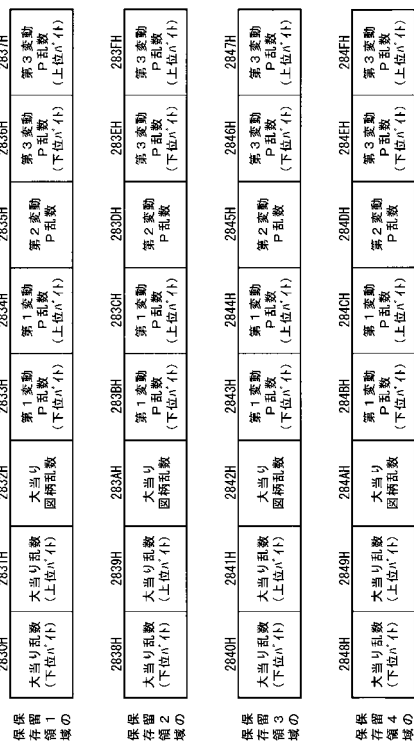
【図 14】



【図 15】



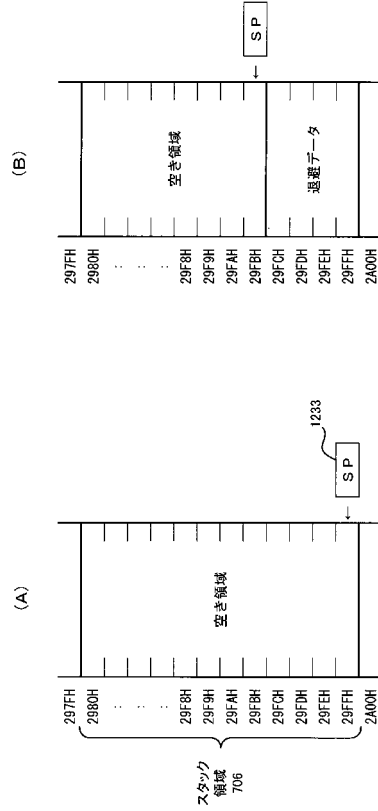
【図 16】



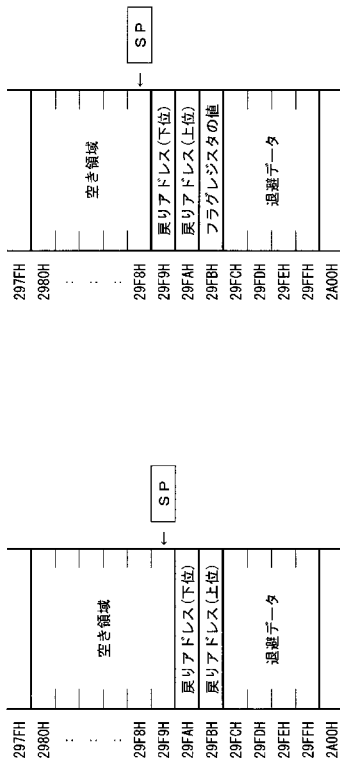
【図 17】

2850H	2851H	2852H	2853H	2854H	2855H	2856H	2857H
保 存 第 1 域 の 保 存 第 1 変 動 P 乱 数 (下位nビット)	大 当 り 乱 数 (上位nビット)	大 当 り 乱 数 (下位nビット)	第 1 変 動 P 乱 数 (上位nビット)	第 1 変 動 P 乱 数 (上位nビット)	第 2 変 動 P 乱 数 (下位nビット)	第 3 変 動 P 乱 数 (下位nビット)	第 3 変 動 P 乱 数 (上位nビット)
2858H	2859H	285AH	285BH	285CH	285DH	285EH	285FH
保 存 第 2 域 の 保 存 第 2 変 動 P 乱 数 (下位nビット)	大 当 り 乱 数 (上位nビット)	大 当 り 乱 数 (下位nビット)	第 1 変 動 P 乱 数 (上位nビット)	第 1 変 動 P 乱 数 (上位nビット)	第 2 変 動 P 乱 数 (下位nビット)	第 3 変 動 P 乱 数 (下位nビット)	第 3 変 動 P 乱 数 (上位nビット)
2860H	2861H	2862H	2863H	2864H	2865H	2866H	2867H
保 存 第 3 域 の 保 存 第 3 変 動 P 乱 数 (下位nビット)	大 当 り 乱 数 (上位nビット)	大 当 り 乱 数 (下位nビット)	第 1 変 動 P 乱 数 (上位nビット)	第 1 変 動 P 乱 数 (上位nビット)	第 2 変 動 P 乱 数 (下位nビット)	第 3 変 動 P 乱 数 (下位nビット)	第 3 変 動 P 乱 数 (上位nビット)
2868H	2869H	286AH	286BH	286CH	286DH	286EH	286FH
保 存 第 4 域 の 保 存 第 4 変 動 P 乱 数 (下位nビット)	大 当 り 乱 数 (上位nビット)	大 当 り 乱 数 (下位nビット)	第 1 変 動 P 乱 数 (上位nビット)	第 1 変 動 P 乱 数 (上位nビット)	第 2 変 動 P 乱 数 (下位nビット)	第 3 変 動 P 乱 数 (下位nビット)	第 3 変 動 P 乱 数 (上位nビット)

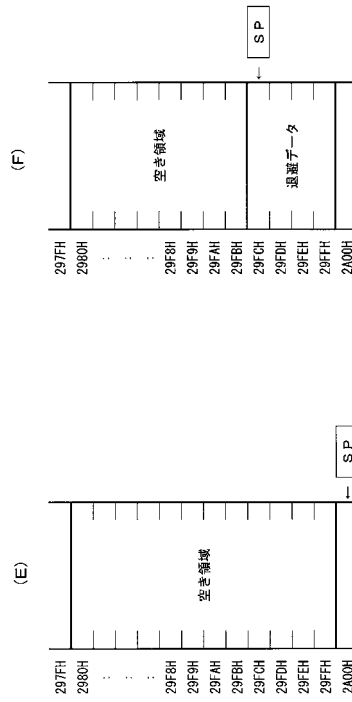
【図 18】



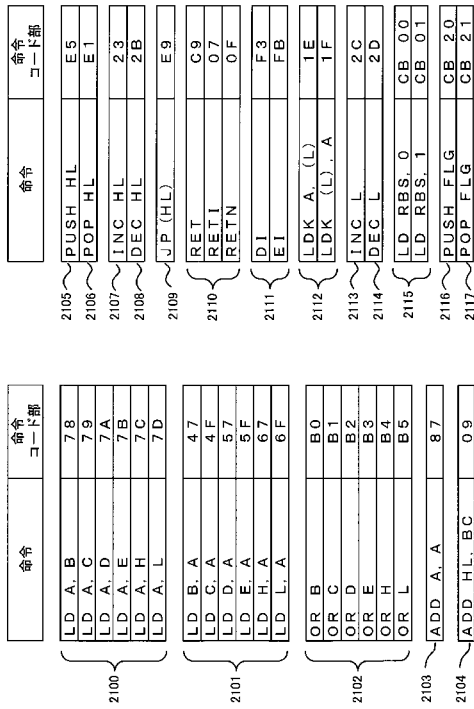
【図 19】



【図 20】



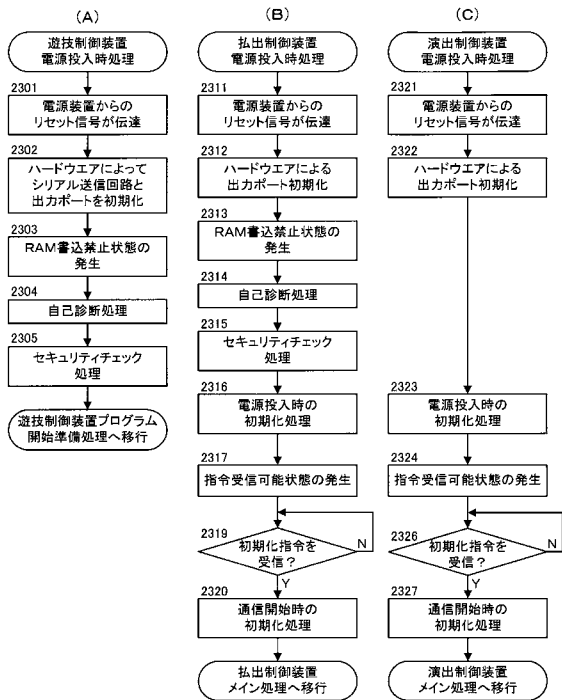
【 図 2 1 】



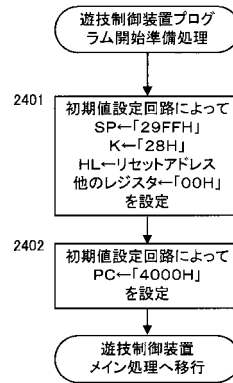
【 図 2 2 】



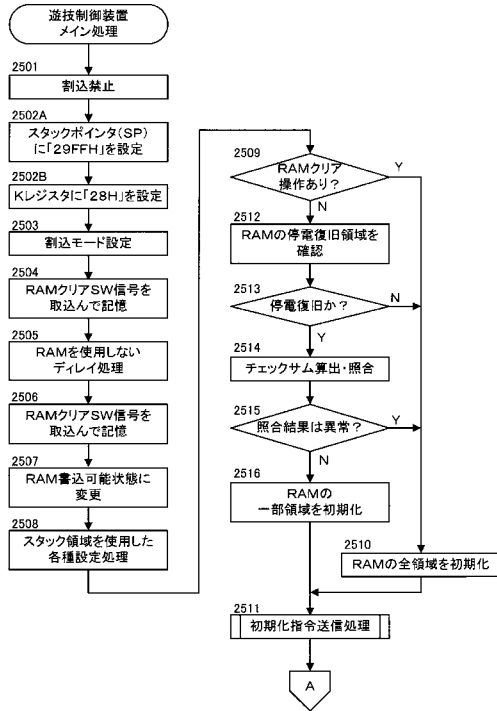
【 図 2 3 】



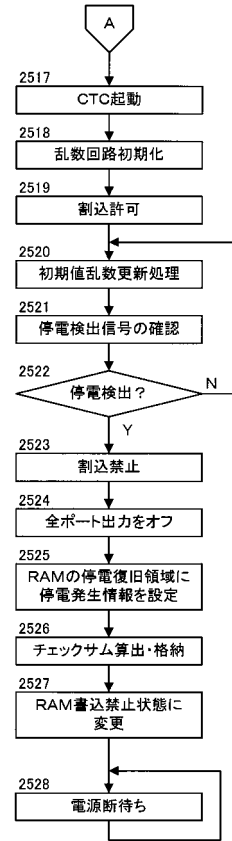
【 図 2 4 】



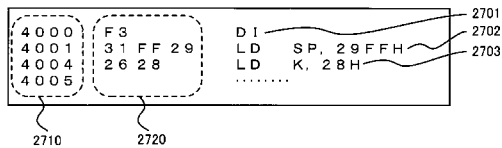
【 図 2 5 】



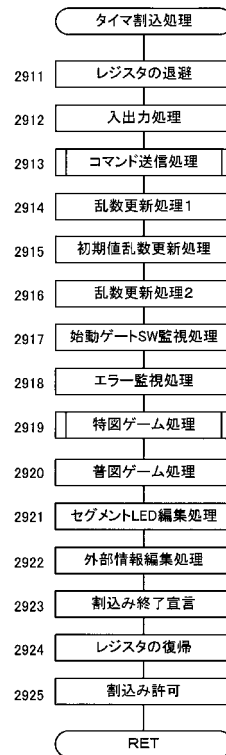
【 図 2 6 】



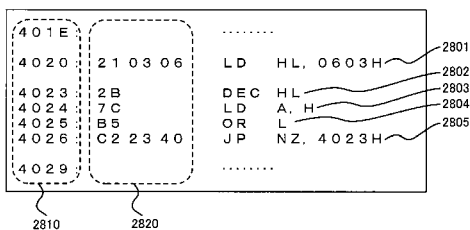
【 図 2 7 】



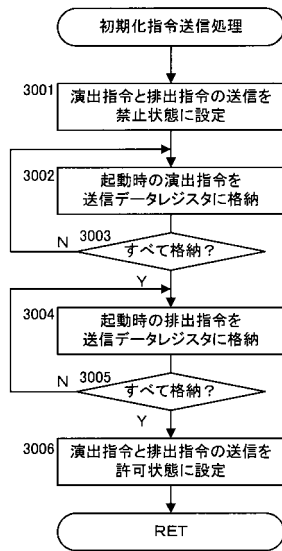
【 図 2 9 】



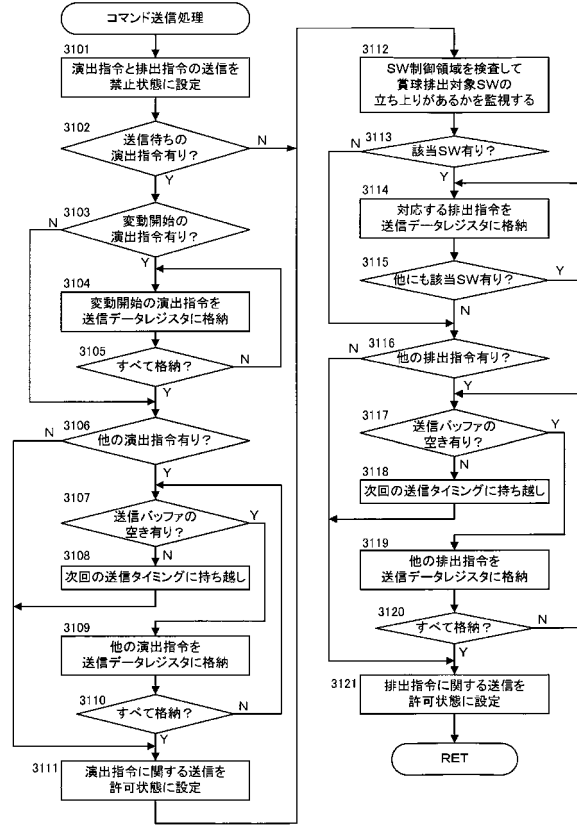
【 図 2 8 】



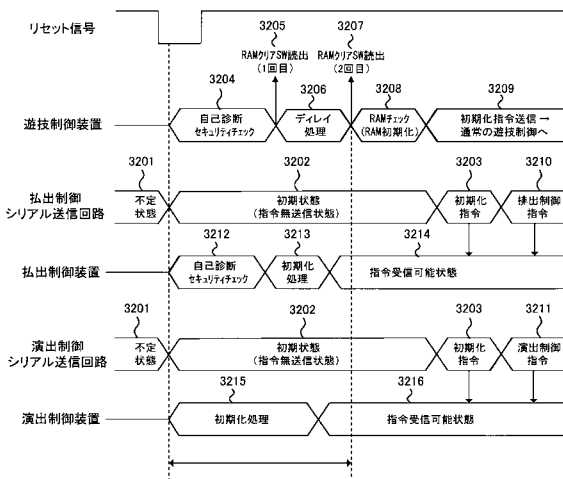
【図30】



【図31】



【図32】

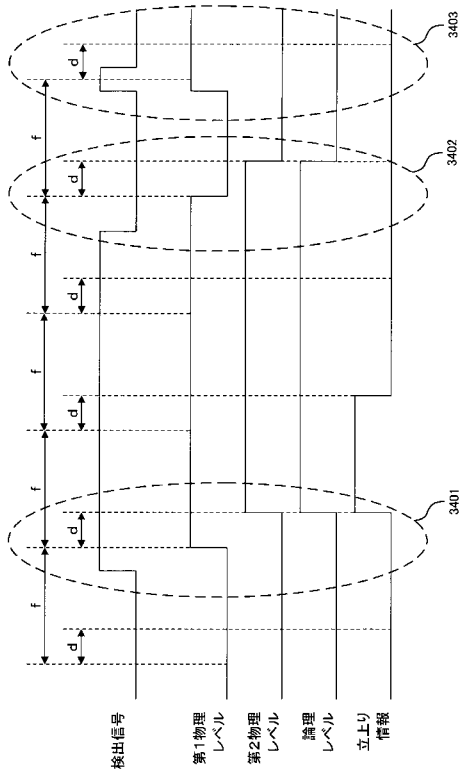


【図33】

MODE	ACTION	機能	出力時期
80H	00H~7FH	初期化指令 (ACTIONにて認証コード送信)	起動時
A1H	5EH	1個賞球排出	入賞時
A2H	5DH	2個賞球排出	入賞時
A3H	5CH	3個賞球排出	入賞時
A4H	5BH	4個賞球排出	入賞時
A5H	5AH	5個賞球排出	入賞時
A6H	59H	6個賞球排出	入賞時
A7H	58H	7個賞球排出	入賞時
A8H	57H	8個賞球排出	入賞時
A9H	56H	9個賞球排出	入賞時
AAH	55H	10個賞球排出	入賞時
ABH	54H	11個賞球排出	入賞時
ACH	53H	12個賞球排出	入賞時
ADH	52H	13個賞球排出	入賞時
AEH	51H	14個賞球排出	入賞時
AFH	50H	15個賞球排出	入賞時
C0H	00H~7FH	エラー発生通知 (ACTIONにてエラー種別を識別)	エラー発生時
C1H	00H~7FH	エラー解除通知 (ACTIONにてエラー種別を識別)	エラー解除時



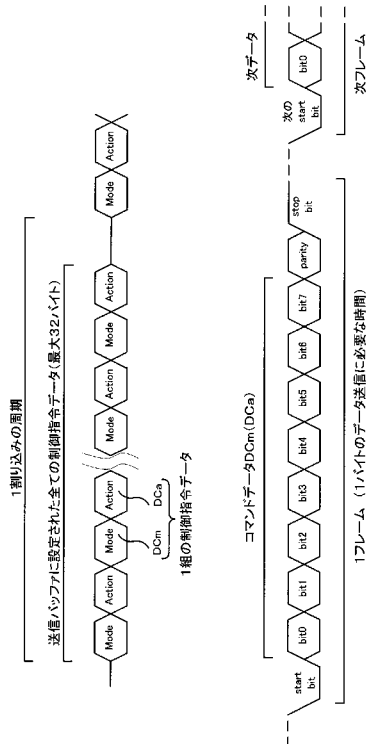
【図34】



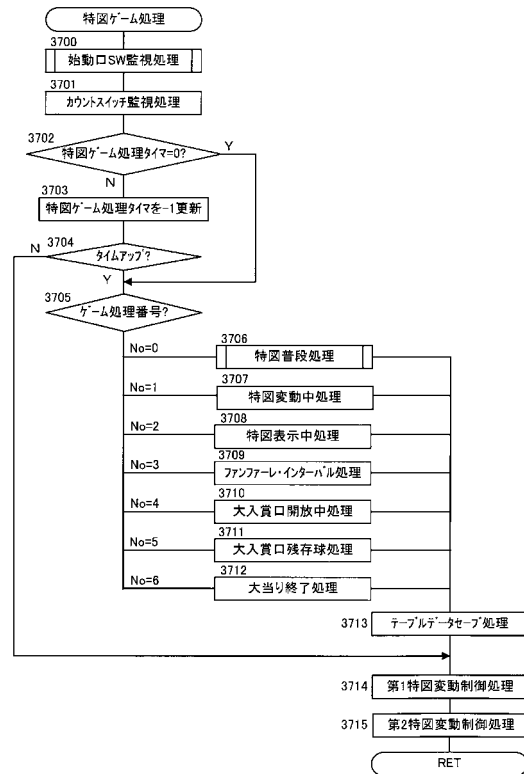
【図35】

MODE	ACTION	機能	出力時期
80H	01H	電源投入コマンドA (RAM初期化処理の実行有り)	起動時
80H	02H	電源投入コマンドB (RAM初期化処理の実行なし)	起動時
81H	01H~7FH	シリーズ機指定コマンド (ACTIONにて番号指定)	起動時
90H	01H	低確率状態発生コマンド	起動時及び確率状態の変化時
90H	02H	高確率状態発生コマンド	起動時及び確率状態の変化時
A0H	00H~04H	保留情報コマンド (ACTIONにて保留記憶数指定)	起動時、始動口入賞時及び変動開始時
A1H~AFH	01H~7FH	変動前演出指定コマンド (MODEにて前半変動パターン番号を指定) (ACTIONにて後半変動パターン番号を指定)	始動口入賞時
B0H	01H~7FH	停止図柄指定コマンド (ACTIONにて停止図柄指定)	変動開始時
B1H~BFH	01H~7FH	残り図柄変動パターン指定コマンド (MODEにて前半変動パターン番号を指定) (ACTIONにて後半変動パターン番号を指定)	変動開始時
C0H	01H	図柄変動停止コマンド	変動停止時
D0H	01H~7FH	大当たり関連コマンド	大当たり中
E0H	01H~7FH	エラー関連コマンド	エラーの発生時 エラー解除時

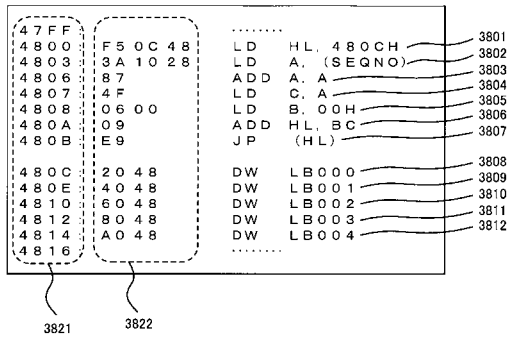
【図36】



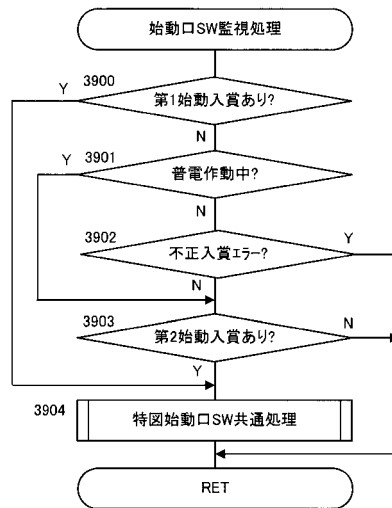
【図37】



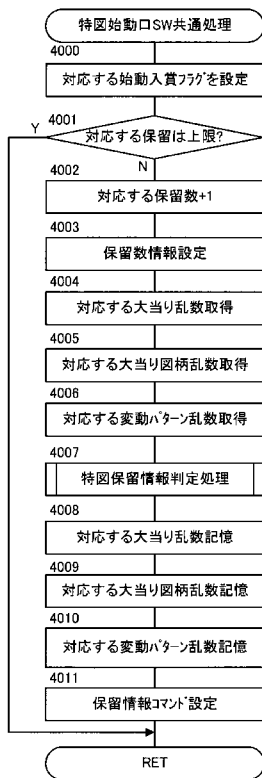
【図38】



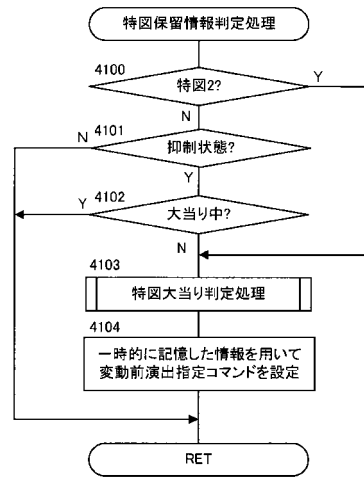
【図39】



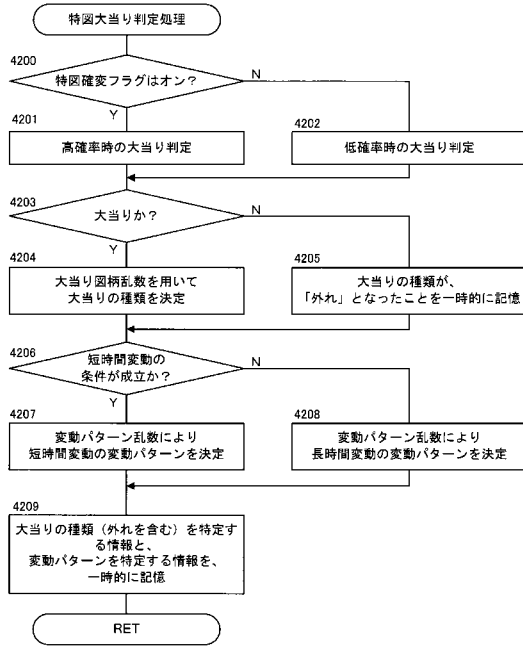
【図40】



【図41】



【図42】



【図43】

(A)

<はずれ時の後半変動パターンテーブル>

グループ番号	グループ名	選択条件	後半変動パターン番号	選択条件	後半変動パターン名	変動時間
0	リーチなし	0~2700	01h	0~260	リーチなし変動	2秒
			11h	0~80	ノーマルリーチ短変動	9秒
1	ノーマルリーチ	2701~2995	12h	81~170	ノーマルリーチ中変動	10秒
			13h	171~250	ノーマルリーチ長変動	11秒
			21h	0~80	SPリーチ短変動	37秒
2	SPリーチ	2996~3000	22h	81~170	SPリーチ中変動	38秒
			23h	171~250	SPリーチ長変動	39秒

(B)

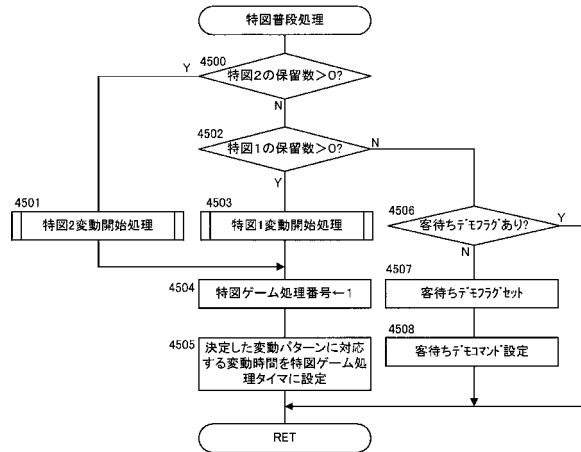
<大当り時の後半変動パターンテーブル>

グループ番号	グループ名	選択条件	後半変動パターン番号	選択条件	後半変動パターン名	変動時間
0	リーチなし	-	-	-	-	-
1	ノーマルリーチ	0~1500	51h	0~80	ノーマルリーチ短変動	9秒
			52h	81~170	ノーマルリーチ中変動	10秒
			53h	171~250	ノーマルリーチ長変動	11秒
2	SPリーチ	1501~3000	61h	0~80	SPリーチ短変動	37秒
			62h	81~170	SPリーチ中変動	38秒
			63h	171~250	SPリーチ長変動	39秒

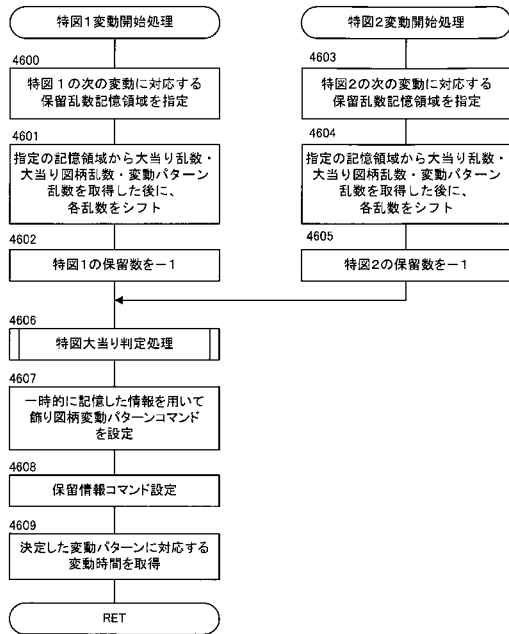
【図44】

後半変動パターン番号	後半変動パターン名	前半変動パターン名	前半変動パターン名	備考
01h	リーチなし変動	前半キャラクタA出現変動	前半キャラクタB出現変動	短縮時の差分 非短縮時の差分
11h/51h	ノーマルリーチ短変動	0~5	6~400	
12h/52h	ノーマルリーチ中変動	0~200	201~400	
13h/53h	ノーマルリーチ長変動	201~205	206~400	
21h/61h	SPリーチ短変動	201~205	206~400	
22h/62h	SPリーチ中変動	201~200	-	
23h/63h	SPリーチ長変動	201~400	-	
		前半変動パターン番号(変動開始時)	前半変動パターン番号(変動入賞時)	
		B1h 10秒	B4h 1秒	
		A1h 10秒	B3h 10秒	
		A2h 10秒	A3h 10秒	
		A1h 10秒	A4h 10秒	

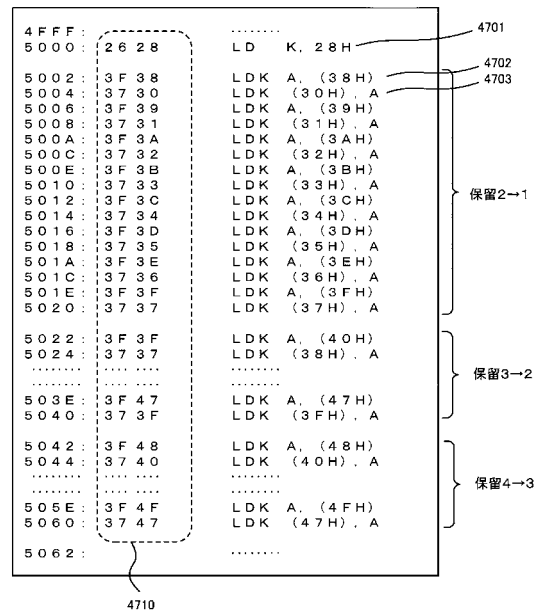
【図45】



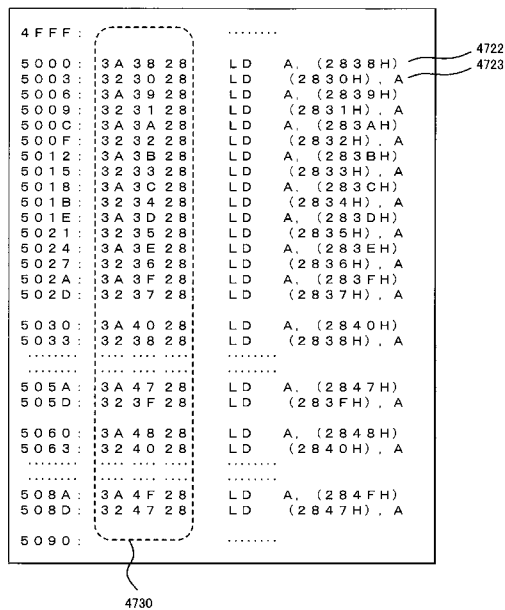
【図46】



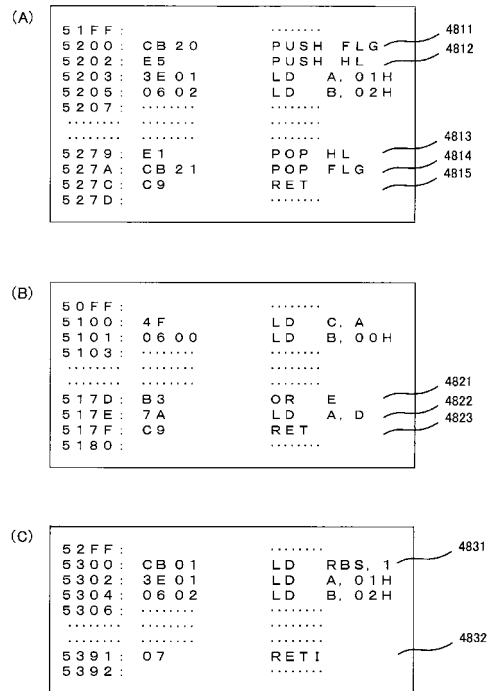
【図47A】



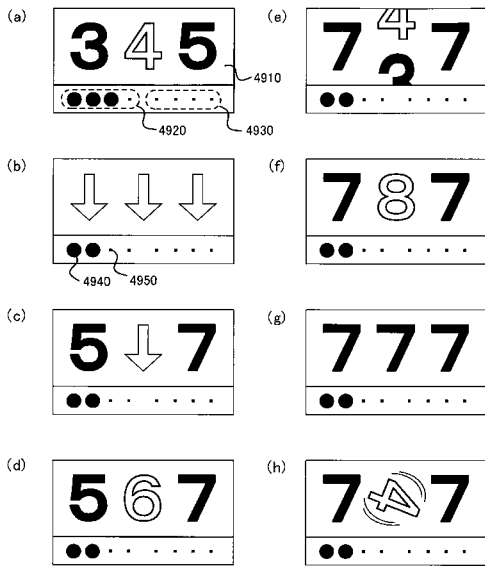
【図47B】



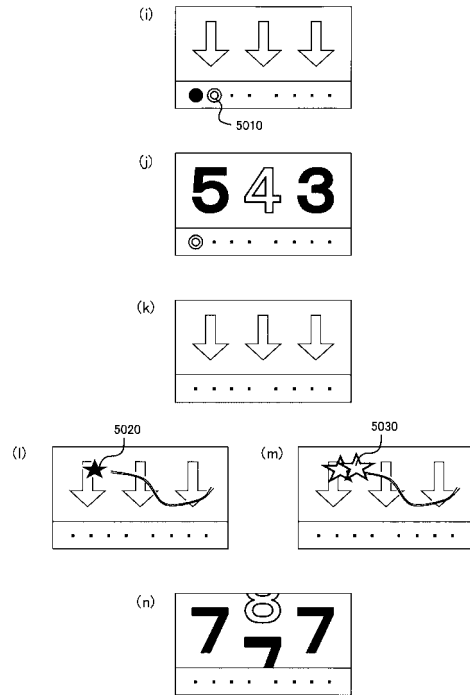
【図48】



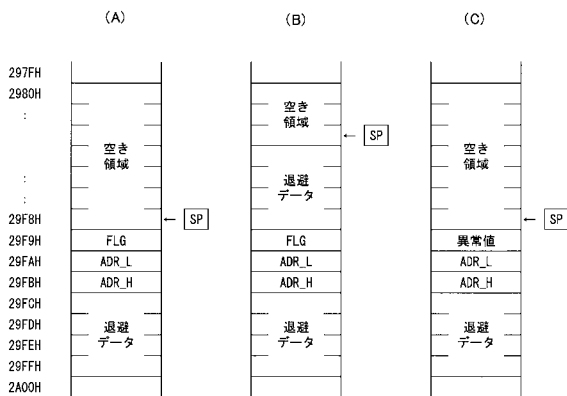
【 図 4 9 】



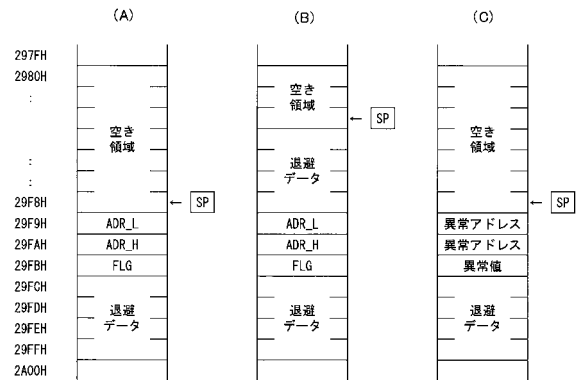
【 図 5 0 】



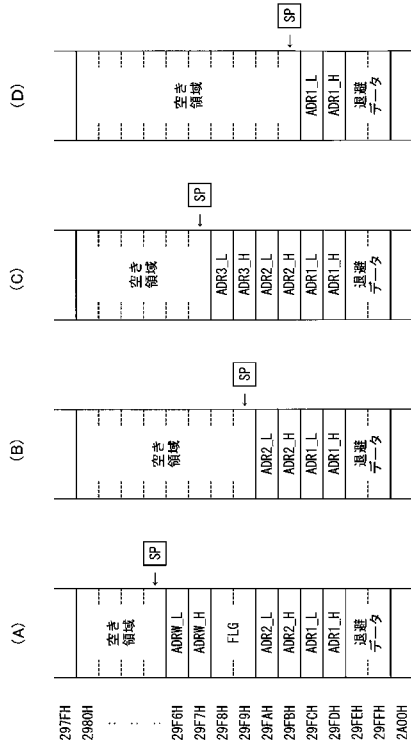
【 図 5 1 A 】



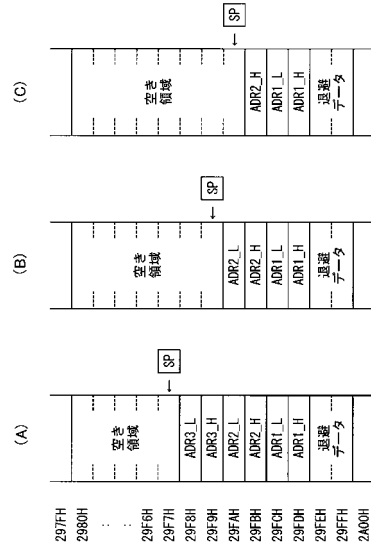
【 図 5 1 B 】



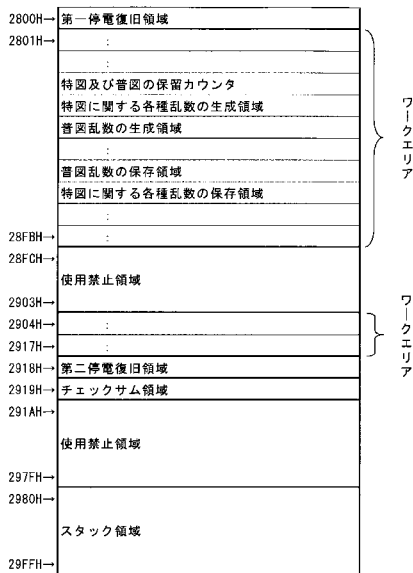
【図 5 1 C】



【図 5 1 D】



【図 5 2】



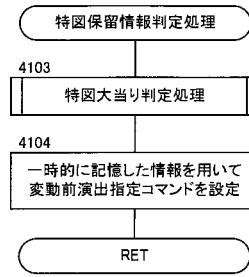
【図 5 3 A】

2830H	2831H	2832H	2833H	2834H	2835H	2836H	2837H
保留 領域 1	保留 領域 2	保留 領域 3	保留 領域 4	保留 領域 1	保留 領域 2	保留 領域 3	保留 領域 4
大当り乱数 (下位n'位)	大当り乱数 (上位n'位)	大当り 回帰乱数	第1変動 P乱数 (下位n'位)	第1変動 P乱数 (上位n'位)	第2変動 P乱数	第3変動 P乱数 (下位n'位)	第3変動 P乱数 (上位n'位)
2838H	2839H	283AH	283BH	283CH	283DH	283EH	283FH
保留 領域 2	保留 領域 3	保留 領域 4	保留 領域 1	保留 領域 2	保留 領域 3	保留 領域 4	保留 領域 1
大当り乱数 (下位n'位)	大当り乱数 (上位n'位)	大当り 回帰乱数	第1変動 P乱数 (下位n'位)	第1変動 P乱数 (上位n'位)	第2変動 P乱数	第3変動 P乱数 (下位n'位)	第3変動 P乱数 (上位n'位)
2840H	2841H	2842H	2843H	2844H	2845H	2846H	2847H
保留 領域 3	保留 領域 4	保留 領域 1	保留 領域 2	保留 領域 3	保留 領域 4	保留 領域 1	保留 領域 2
大当り乱数 (下位n'位)	大当り乱数 (上位n'位)	大当り 回帰乱数	第1変動 P乱数 (下位n'位)	第1変動 P乱数 (上位n'位)	第2変動 P乱数	第3変動 P乱数 (下位n'位)	第3変動 P乱数 (上位n'位)
2848H	2849H	284AH	284BH	284CH	284DH	284EH	284FH
保留 領域 4	保留 領域 1	保留 領域 2	保留 領域 3	保留 領域 4	保留 領域 1	保留 領域 2	保留 領域 3
大当り乱数 (下位n'位)	大当り乱数 (上位n'位)	大当り 回帰乱数	第1変動 P乱数 (下位n'位)	第1変動 P乱数 (上位n'位)	第2変動 P乱数	第3変動 P乱数 (下位n'位)	第3変動 P乱数 (上位n'位)

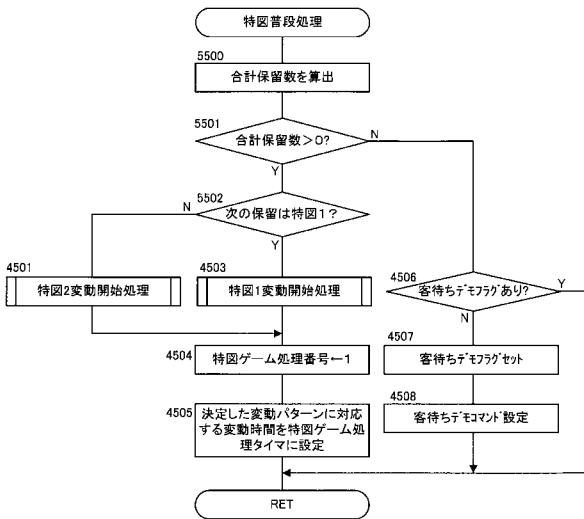
【 図 5 3 B 】

285DH 大当り乱数 (下位nビット)	285EH 大当り乱数 (上位nビット)	285FH 大当り 図柄乱数	285GH 第1変動 P乱数 (下位nビット)	285HH 第1変動 P乱数 (上位nビット)	285IH 第2変動 P乱数	285JH 第3変動 P乱数 (下位nビット)	285KH 第3変動 P乱数 (上位nビット)	285LH 特図識別ビット
285BH 大当り乱数 (下位nビット)	285CH 大当り乱数 (上位nビット)	285DH 大当り 図柄乱数	285EH 第1変動 P乱数 (下位nビット)	285FH 第1変動 P乱数 (上位nビット)	285GH 第2変動 P乱数	285IH 第3変動 P乱数 (下位nビット)	285JH 第3変動 P乱数 (上位nビット)	285KH 特図識別ビット
286DH 大当り乱数 (下位nビット)	286EH 大当り乱数 (上位nビット)	286FH 大当り 図柄乱数	286GH 第1変動 P乱数 (下位nビット)	286HH 第1変動 P乱数 (上位nビット)	286IH 第2変動 P乱数	286JH 第3変動 P乱数 (下位nビット)	286KH 第3変動 P乱数 (上位nビット)	286LH 特図識別ビット
286BH 大当り乱数 (下位nビット)	286CH 大当り乱数 (上位nビット)	286DH 大当り 図柄乱数	286EH 第1変動 P乱数 (下位nビット)	286FH 第1変動 P乱数 (上位nビット)	286GH 第2変動 P乱数	286IH 第3変動 P乱数 (下位nビット)	286JH 第3変動 P乱数 (上位nビット)	286KH 特図識別ビット

【 図 5 4 】



【 図 5 5 】



---

フロントページの続き

Fターム(参考) 2C088 AA17 AA35 AA36 AA42 BB11 BC15 BC22 BC23 BC58 CA27  
CA28 DA21 EA10 EB14 EB28 EB56 EB58