

【公報種別】特許法第 17 条の 2 の規定による補正の掲載

【部門区分】第 1 部門第 2 区分

【発行日】平成21年6月4日(2009.6.4)

【公開番号】特開2008-43797(P2008-43797A)

【公開日】平成20年2月28日(2008.2.28)

【年通号数】公開・登録公報2008-008

【出願番号】特願2007-282550(P2007-282550)

【国際特許分類】

A 6 3 F 7/02 (2006.01)

【F I】

A 6 3 F 7/02 3 3 4

A 6 3 F 7/02 3 2 6 Z

【手続補正書】

【提出日】平成21年4月20日(2009.4.20)

【手続補正 1】

【補正対象書類名】特許請求の範囲

【補正対象項目名】全文

【補正方法】変更

【補正の内容】

【特許請求の範囲】

【請求項 1】

乱数カウンタと、その乱数カウンタの値を所定の範囲内で更新する第 1 更新手段と、所定の契機に基づいて前記乱数カウンタの値を読み出す読出手段と、その読出手段により読み出された前記乱数カウンタの値が予め定められた値と一致する場合に遊技者に所定の遊技価値を付与する制御手段とを備えた遊技機において、

前記乱数カウンタの値は、前記第 1 更新手段により所定回更新されることで一周するものであり、

前記第 1 更新手段は、前記乱数カウンタの値が一周すると、前記所定の範囲内のいずれかの値を更新の初期値として次の週の更新を行うものであり、

前記制御手段は、

前記乱数カウンタの値が一周する毎に、前記第 1 更新手段の更新の初期値を変更する変更手段と、

前記所定の範囲と同じ範囲で更新され、前記変更手段が初期値の変更に使用する初期値カウンタと、

その初期値カウンタの値を更新する第 2 更新手段とを備え、

定期的な割込信号に基づいて定期処理を行うとともに、その定期処理の終了の後、次の定期処理が行われるまでの期間に繰り返し所定の処理を行うものであり、

その所定の処理および前記定期処理において、前記第 2 更新手段による更新が行われ、

前記第 1 更新手段による前記乱数カウンタの更新、および、前記読出手段による前記乱数カウンタの値の読み出しは、前記定期処理において行われ、

電源投入後最初の前記読出手段による前記乱数カウンタの値の読み出しが行われるより前に、前記第 1 更新手段による前記乱数カウンタの更新、および、前記第 2 更新手段による前記初期値カウンタの更新が行われ、

さらに、電源投入後最初の前記第 1 更新手段による前記乱数カウンタの更新が行われるより前に、前記第 2 更新手段による前記初期値カウンタの更新が行われることを特徴とする遊技機。

【請求項 2】

前記遊技機は、パチンコ機であることを特徴とする請求項 1 記載の遊技機。

## 【手続補正２】

【補正対象書類名】明細書

【補正対象項目名】全文

【補正方法】変更

【補正の内容】

【発明の詳細な説明】

【発明の名称】遊技機

【技術分野】

【０００１】

本発明は、パチンコ遊技機などに代表される遊技機に関し、特に、「ぶら下げ基板」等による不正行為を防止することができる遊技機に関するものである。

【背景技術】

【０００２】

この種のパチンコ遊技機は、複数種類の図柄を変動表示可能な表示装置を備えており、遊技領域に打ち込まれた打球が図柄作動ゲートを通過すると、変動表示を開始するように構成されている。この変動表示が予め定められた図柄の組み合わせと一致して停止すると、大当たりとなって、遊技者に所定の遊技価値が付与され、大量の遊技球が払出可能な状態となる。

【０００３】

かかる大当たりの発生の有無は、打球が図柄作動ゲートを通過するタイミングで決定される。即ち、１カウントずつ定期的に一定の範囲で（例えば、１カウントずつ、２ｍｓ毎に、０から６３０の範囲で）更新されるカウンタを備え、打球が図柄作動ゲートを通過したときに、そのカウンタの値を読み出して、読み出されたカウンタの値が、例えば「７」などの所定値と一致する場合に、大当たりを発生するようにしている。

【発明の開示】

【発明が解決しようとする課題】

【０００４】

ところが、最近、「ぶら下げ基板」と呼ばれる不正な基板を使用した不正行為が報告されている。この不正行為は、制御基板と表示装置の表示用基板等との間に、不正な基板をぶら下げて（不正な「ぶら下げ基板」を取り付けて）、不当に大当たりを発生させるというものである。具体的には、前記したパチンコ遊技機に設けられる大当たりを決定するためのカウンタと同様の働きをするカウンタ（１カウントずつ定期的に一定の範囲で更新されるカウンタ）を「ぶら下げ基板」内に設け、そのカウンタの値をパチンコ遊技機の電源投入等に合わせてリセット（０クリア）することにより、「ぶら下げ基板」内で大当たりの発生タイミングを把握するのである。そして、その把握した大当たりの発生タイミングに合わせて、「ぶら下げ基板」内で打球の図柄作動ゲート通過信号を不正に生成し、これをパチンコ遊技機の制御基板へ出力して、不当に大当たりを発生させるというものである。遊技場などでは、この「ぶら下げ基板」を用いた不正行為により、多大な被害を被っているという問題点があった。

【０００５】

本発明は上記例示した問題点等を解決するためになされたものであり、大当たり等の発生タイミングの把握を不可能にして、「ぶら下げ基板」等を用いた不正行為を防止することができる遊技機を提供することを目的としている。

【課題を解決するための手段】

【０００６】

この目的を達成するために請求項１記載の遊技機は、乱数カウンタと、その乱数カウンタの値を所定の範囲内で更新する第１更新手段と、所定の契機に基づいて前記乱数カウンタの値を読み出す読出手段と、その読出手段により読み出された前記乱数カウンタの値が予め定められた値と一致する場合に遊技者に所定の遊技価値を付与する制御手段とを備えており、前記乱数カウンタの値は、前記第１更新手段により所定回更新されることで一周

するものであり、前記第 1 更新手段は、前記乱数カウンタの値が一周すると、前記所定の範囲内のいずれかの値を更新の初期値として次の周の更新を行うものであり、前記制御手段は、前記乱数カウンタの値が一周する毎に、前記第 1 更新手段の更新の初期値を変更する変更手段と、前記所定の範囲と同じ範囲で更新され、前記変更手段が初期値の変更に使用する初期値カウンタと、その初期値カウンタの値を更新する第 2 更新手段とを備え、定期的な割込信号に基づいて定期処理を行うとともに、その定期処理の終了の後、次の定期処理が行われるまでの期間に繰り返し所定の処理を行うものであり、その所定の処理および前記定期処理において、前記第 2 更新手段による更新が行われ、前記第 1 更新手段による前記乱数カウンタの更新、および、前記読出手段による前記乱数カウンタの値の読み出しは、前記定期処理において行われ、電源投入後最初の前記読出手段による前記乱数カウンタの値の読み出しが行われるより前に、前記第 1 更新手段による前記乱数カウンタの更新、および、前記第 2 更新手段による前記初期値カウンタの更新が行われ、さらに、電源投入後最初の前記第 1 更新手段による前記乱数カウンタの更新が行われるより前に、前記第 2 更新手段による前記初期値カウンタの更新が行われる。

【 0 0 0 7 】

請求項 2 記載の遊技機は、請求項 1 記載の遊技機において、前記遊技機は、パチンコ機である。

【 0 0 0 8 】

本発明の遊技機によれば、乱数カウンタの値は、第 1 更新手段により更新されると共に、所定の契機に基づいて読出手段によって読み出される。読み出された乱数カウンタの値が予め定められた値と一致する場合に、遊技者に所定の遊技価値が付与される。

【 0 0 0 9 】

乱数カウンタの更新の初期値は、初期値カウンタの値に基づいて変更される。このように、乱数カウンタの更新の初期値は、固定値ではなく、定期的に変更される値であるので、「ぶら下げ基板」等が大当たりの発生タイミングを把握することを防止することができるのである。

【 0 0 1 0 】

【 0 0 1 1 】

【 0 0 1 2 】

【 0 0 1 3 】

【発明の効果】

【 0 0 1 4 】

本発明の遊技機によれば、大当たり等を決定するための乱数カウンタの更新の初期値は、固定値ではなく、定期的に変更される値であるので、「ぶら下げ基板」等による大当たりの発生タイミングの把握を不可能にして、「ぶら下げ基板」等による不正行為を防止することができるという効果がある。

【発明を実施するための最良の形態】

【 0 0 1 5 】

以下、本発明の好ましい実施例について、添付図面を参照して説明する。本実施例では、弾球遊技機の一例としてパチンコ遊技機、特に、第 1 種パチンコ遊技機を用いて説明する。なお、本発明を第 3 種パチンコ遊技機や他の弾球遊技機に用いることは、当然に可能である。

【 0 0 1 6 】

図 1 は、第 1 実施例におけるパチンコ遊技機 P の遊技盤の正面図である。遊技盤 1 の周囲には、打球が入賞することにより 5 個から 15 個の遊技球が払い出される複数の入賞口 2 が設けられている。また、遊技盤 1 の中央には、複数種類の識別情報としての図柄などを表示する液晶 (LCD) ディスプレイ 3 が設けられている。この LCD ディスプレイ 3 の表示画面は横方向に 3 分割されており、3 分割された各表示領域において、それぞれ図柄の変動表示が行われる。

【 0 0 1 7 】

L C Dディスプレイ3の下方には、図柄作動ゲート(第1種始動口)4が設けられ、打球がこの図柄作動ゲート4を通過することにより、前記したL C Dディスプレイ3の変動表示が開始される。図柄作動ゲート4の下方には、特定入賞口(大入賞口)5が設けられている。この特定入賞口5は、L C Dディスプレイ3の変動後の表示結果が予め定められた図柄の組み合わせの1つと一致する場合に、大当たりとなって、打球が入賞しやすいように所定時間(例えば、30秒経過するまで、あるいは、打球が10個入賞するまで)開放される入賞口である。この特定入賞口5内には、Vゾーン5aが設けられており、特定入賞口5の開放中に、打球がVゾーン5a内を通過すると、継続権が成立して、特定入賞口5の閉鎖後、再度、その特定入賞口5が所定時間(又は、特定入賞口5に打球が所定個数入賞するまで)開放される。この特定入賞口5の開閉動作は、最高で16回(16ラウンド)繰り返し可能にされており、開閉動作の行われ得る状態が、いわゆる所定の遊技価値の付与された状態(特別遊技状態)である。

#### 【0018】

図2は、かかるパチンコ遊技機Pの電氣的構成を示したブロック図である。パチンコ遊技機Pの制御部Cは、演算装置であるC P U 1 1と、そのC P U 1 1により実行される各種の制御プログラムや固定値データを記憶したR O M 1 2と、各種のデータ等を一時的に記憶するためのメモリであるR A M 1 3とを備えている。図3から図5に示すフローチャートのプログラムは、制御プログラムの一部としてR O M 1 2内に記憶されている。

#### 【0019】

C P U 1 1は、演算を行うA L Uのほか、アキュムレータ(以下「A c c」と称す)1 1 aや複数の内部レジスタ1 1 b、フラグレジスタ1 1 cを備えている。R A M 1 3内に設けられるカウンタ等の値は、一旦、C P U 1 1の内部レジスタ1 1 bへロードされ(書き込まれ)、その内部レジスタ1 1 b内で更新された後に、R A M 1 3の元のカウンタ内へセーブされて(書き込まれて)、更新される。

#### 【0020】

なお、68系の8ビットC P U 1 1では、ペアになっている2バイト(16ビット)の内部レジスタ1 1 bの値を、連続したアドレスの2バイトのメモリ(R A M 1 3内)へ1命令でセーブする(書き込む)ことができる。この場合の書き込みは、バスライン14のデータバスは8ビットで構成されるので、上位バイト、下位バイトの順に行われる。また、80系の8ビットC P Uでは、68系のC P U 1 1とは逆に、ペアになっている2バイト(16ビット)の内部レジスタの値を、連続したアドレスの2バイトのメモリへ、下位バイト上位バイトの順に1命令でセーブすることができる。

#### 【0021】

R A M 1 3は、乱数カウンタ1 3 aと、初期値カウンタ1 3 bと、初期値メモリ1 3 cとを備えている。乱数カウンタ1 3 aは、大当たりの発生を決定するためのカウンタであり、図4の乱数更新処理(S7)によって、「0~630(0~276h)」の範囲で、2ms毎に1カウントずつ更新される。このため乱数カウンタ1 3 aは2バイトで構成されている。打球が図柄作動ゲート4を通過したときに取得した乱数カウンタ1 3 aの値が例えば「7」であると、大当たりが発生する。大当たりが発生すると、大当たりコマンドが制御部Cから後述する表示装置Dへ送られる。表示装置Dは、この大当たりコマンドに基づいて、L C Dディスプレイ3の変動表示を大当たりの状態に制御する。

#### 【0022】

初期値カウンタ1 3 bは、乱数カウンタ1 3 aの更新の初期値をカウントするためのカウンタであり、乱数カウンタ1 3 aと同様に2バイトで構成されている。この初期値カウンタ1 3 bの値は、図5の初期値カウンタ更新処理(S6, S22)によって、乱数カウンタ1 3 aの更新範囲と同じ「0~630(276h)」の範囲で、1カウントずつ更新される。

#### 【0023】

図5の初期値カウンタ更新処理は、図3のリセット割込処理における残余時間の間、即ち、効果音処理(S20)の終了後、次のリセット割込処理が発生するまでの間に、繰り

返し実行される ( S 2 2 )。リセット割込処理は 2 m s 毎に実行されるが、1 回のリセット割込処理において実行される S 1 から S 2 0 の各処理の処理時間は遊技の状況に応じて変化するので、リセット割込処理の残余時間は、一定な時間ではなく、遊技の状況に応じて変化する不定な時間となる。「ぶら下げ基板」ではこの不定な時間を把握することはできないので、かかる不定な時間内に繰り返し更新される初期値カウンタ 1 3 b の値を乱数カウンタ 1 3 a の更新の初期値として使用することにより、「ぶら下げ基板」による大当たり発生のタイミングの把握を不可能にしている。

#### 【 0 0 2 4 】

なお、図 5 の説明で後述するように、初期値カウンタ 1 3 b の値は、C P U 1 1 の内部レジスタ 1 1 b を介して更新される。内部レジスタ 1 1 b から初期値カウンタ 1 3 b への書き込みは 6 8 系 C P U 1 1 の 1 命令によって、上位バイト、下位バイトの順に行われる。よって、例えば更新前の初期値カウンタ 1 3 b の値が「 1 F F h 」の場合、内部レジスタ 1 1 b 内で「 2 0 0 h 」に更新された後に、更新後の値が上位バイト、下位バイトの順に初期値カウンタ 1 3 b へ書き込まれる ( S 4 5 )。即ち、初期値カウンタ 1 3 b の値は、「 1 F F h 」の状態から内部レジスタ 1 1 b の上位バイトが書き込まれることにより一旦「 2 F F h 」となり、その後、内部レジスタ 1 1 b の下位バイトが書き込まれて「 2 0 0 h 」に更新されるのである。

#### 【 0 0 2 5 】

ところが、図 5 の初期値カウンタ更新処理はリセット割込処理の残余時間の間に繰り返し実行されるので、更新後の内部レジスタ 1 1 b の上位バイトを初期値カウンタ 1 3 b へ書き込んだ後であって内部レジスタ 1 1 b の下位バイトの書き込み前に ( S 4 5 の処理の途中に)、次のリセット割込処理が発生する場合がある。リセット割込は、割込の発生を禁止することができないノンマスカプブルな割込であると共に、割込の優先順位が最も高く、C P U 1 1 の命令の実行途中であっても強制的に開始される割込である。よって、かかるタイミングに ( 更新後の内部レジスタ 1 1 b の上位バイトを初期値カウンタ 1 3 b へ書き込んだ後であって内部レジスタ 1 1 b の下位バイトの書き込み前に)、リセット割込が発生すると、初期値カウンタ 1 3 b の値は、本来の更新範囲である「 0 ~ 6 3 0 ( 0 ~ 2 7 6 h ) 」の範囲を超えた「 7 6 7 ( 2 F F h ) 」になってしまうことがある。

#### 【 0 0 2 6 】

乱数カウンタ 1 3 a の値の更新範囲も「 0 ~ 6 3 0 ( 0 ~ 2 7 6 h ) 」なので、この初期値カウンタ 1 3 b の値 ( 7 6 7 ( 2 F F h ) ) を乱数カウンタ 1 3 a の初期値とすると、乱数カウンタ 1 3 a の更新周期が変動して大当たりの発生確率を設定値と異なった確率にしまったり、或いは、乱数カウンタ 1 3 a の更新の初期値を以降は変更できなくしてしまうという不具合が生じる。図 4 の乱数更新処理 ( S 7 ) では、乱数カウンタ 1 3 a の値は「 6 3 1 ( 2 7 7 h ) 」以上にはなり得ないので ( S 3 3 : Y e s , S 3 4 )、一旦、乱数カウンタ 1 3 a の更新の範囲外の値である「 6 3 1 ( 2 7 7 h ) 」以上の値が、乱数カウンタ 1 3 a の更新の初期値として初期値メモリ 1 3 c に書き込まれると ( S 3 6 , S 3 7 )、以降は S 3 5 の処理において、Y e s の分岐が生じ得ないからである。

#### 【 0 0 2 7 】

ここで、初期値カウンタ 1 3 b の値が初期値として乱数カウンタ 1 3 a へ書き込まれるのは乱数更新処理である ( S 7 の S 3 6 ~ S 3 8 )。一方、本来の更新範囲外の値となった初期値カウンタ 1 3 b の値は、図 5 の初期値カウンタ更新処理を再実行することにより、本来の更新範囲内の値に戻される。「 6 3 1 ( 2 7 7 h ) 」以上の初期値カウンタ 1 3 b の値は「 0 」クリアされるからである ( S 4 3 : Y e s , S 4 4 , S 4 5 )。よって、本実施例では、上記不具合の発生を回避するために、図 5 の初期値カウンタ更新処理を、リセット割込処理の残余時間の間に繰り返し実行するだけでなく ( S 2 2 )、乱数更新処理 ( S 7 ) の実行前にも少なくとも 1 回実行している ( S 6 )。

#### 【 0 0 2 8 】

初期値メモリ 1 3 c は、乱数カウンタ 1 3 a の更新の初期値を記憶するためのメモリであり、乱数カウンタ 1 3 a と同様に 2 バイトで構成されている。本実施例では、乱数カウ

ンタ 1 3 a の更新の初期値は、乱数カウンタの一回りの更新毎に変更される。よって、更新された乱数カウンタ 1 3 a の値が初期値メモリ 1 3 c の値と一致すると、乱数カウンタ 1 3 a の一回りの更新が終了したことになるので、両値 1 3 a , 1 3 c の一致を契機として、そのときの初期値カウンタ 1 3 b の値が乱数カウンタ 1 3 a および初期値メモリ 1 3 c に書き込まれて、乱数カウンタ 1 3 a の更新の初期値が変更される。従って、乱数カウンタ 1 3 a の更新の初期値を変更しても、乱数の一様性（連続で取得した場合に同じ値を取ることがなく、しかも、すべての値が同じ確率で取り出せること）のある乱数値を得ることができるのである。

【 0 0 2 9 】

これらの CPU 1 1、ROM 1 2、RAM 1 3 は、バスライン 1 4 を介して互いに接続されており、バスライン 1 4 は、また、入出力ポート 1 5 にも接続されている。この入出力ポート 1 5 は表示装置 D や他の入出力装置 1 6 と接続されている。制御部 C は、入出力ポート 1 5 を介して、表示装置 D や他の入出力装置 1 6 へ動作コマンドを送り、それら各装置を制御する。LCD ディスプレイ 3 の変動表示や特定入賞口 5 の開閉動作も、この動作コマンドに基づいて制御される。

【 0 0 3 0 】

表示装置 D は、CPU 2 1 と、プログラム ROM 2 2 と、ワーク RAM 2 3 と、ビデオ RAM 2 4 と、キャラクタ ROM 2 5 と、画像コントローラ 2 6 と、入出力ポート 2 7 と、LCD ディスプレイ 3 とを備えている。表示装置 D の CPU 2 1 は、制御部 C から出力される動作コマンドに応じて、LCD ディスプレイ 3 の表示制御（変動表示）を行うものであり、プログラム ROM 2 2 には、この CPU 2 1 により実行されるプログラムが記憶されている。ワーク RAM 2 3 は、CPU 2 1 によるプログラムの実行時に使用されるワークデータが記憶されるメモリである。

【 0 0 3 1 】

ビデオ RAM 2 4 は、LCD ディスプレイ 3 に表示されるデータが記憶されるメモリであり、このビデオ RAM 2 4 の内容を書き換えることにより、LCD ディスプレイ 3 の表示内容が変更される。即ち、各表示領域における図柄の変動表示は、ビデオ RAM 2 4 の内容が書き換えられることにより行われる。キャラクタ ROM 2 5 は、LCD ディスプレイ 3 に表示される図柄などのキャラクタデータを記憶するメモリである。画像コントローラ 2 6 は、CPU 2 1、ビデオ RAM 2 4、入出力ポート 2 7 のそれぞれのタイミングを調整して、データの読み書きを介在するとともに、ビデオ RAM 2 4 に記憶される表示データをキャラクタ ROM 2 5 を参照して所定のタイミングで LCD ディスプレイ 3 に表示させるものである。

【 0 0 3 2 】

次に、上記のように構成されたパチンコ遊技機 P で実行される各処理を、図 3 から図 5 のフローチャートを参照して説明する。図 3 は、パチンコ遊技機 P の制御部 C において、2 m s 毎に実行されるリセット割込処理のフローチャートである。パチンコ遊技機 P の主な制御は、このリセット割込処理によって実行される。

【 0 0 3 3 】

リセット割込処理では、まず、スタックポインタを設定し（S 1）、RAM 1 3 の所定エリアに書き込まれているパターンのチェックを行う（S 2）。チェックの結果、所定エリアに所定のパターンが書き込まれていれば、RAM 1 3 に異常はなく正常であるので（S 2：正常）、処理を S 3 へ移行する。一方、S 2 のチェックの結果、所定エリアに所定のパターンが書き込まれていなければ、電源投入後最初に行われるリセット割込処理であるか、或いは、RAM 1 3 に異常があるので（S 2：異常）、この場合には処理を S 2 3 へ移行して、一旦、RAM 1 3 の内容をクリアした後、RAM 1 3 内へ初期値を書き込んで（S 2 3）、次のリセット割込処理の発生を待機する。

【 0 0 3 4 】

S 3 の処理ではタイマ割込の設定を行う（S 3）。ここで設定されるタイマ割込としては、LCD ディスプレイ 3 の表示を制御するコマンドを表示装置 D へ送信するためのスト

ローブ信号を発生させるタイマ割込などがある。タイマ割込の設定後は、各割込を許可状態とする（S 4）。割込の許可後は、特別図柄変動処理（S 1 6）や、表示データ作成処理（S 1 8）、ランプ・情報処理（S 1 9）などにより、前回のリセット割込処理において更新された出力データを一度に各ポートへ出力するポート出力処理を実行する（S 5）。その後、後述する初期値カウンタ更新処理（S 6）を実行して、初期値カウンタ 1 3 b を加算方向へ「+ 1」更新すると共に、乱数更新処理（S 7）を実行して、乱数カウンタ 1 3 a の値を「+ 1」更新し、更に、記憶タイマ減算処理を実行する（S 8）。記憶タイマ減算処理は、大当たり判定の保留球が所定数以上あり、且つ、LCDディスプレイ 3 において図柄の変動表示中である場合に、図柄の変動表示時間の短縮を行うものである。

#### 【0035】

スイッチ読込処理（S 9）は、各スイッチの値を読み込むことにより、遊技領域 1 へ打ち込まれた打球の入賞口 2 や大入賞口 5（Vゾーン 5 a を含む）への入賞、図柄作動ゲート 4 の通過、更には賞球や貸球を検出するための処理である。カウント異常監視処理（S 1 0）は、S 9 のスイッチ読込処理によって読み込まれたスイッチデータに異常があるか否かを監視するための処理である。例えば、大入賞口 5 が開放され、打球の Vゾーン 5 a の通過を検出する V カウントスイッチで打球が検出されたにも拘わらず、Vゾーン 5 a 以外の大入賞口 5 への入賞を検出する 1 0 カウントスイッチで 1 球の打球も検出できない場合には、1 0 カウントスイッチが抜き取られるなどして、1 0 カウントスイッチに何らかの異常が発生している。また、賞球を払い出すモータを駆動したにも拘わらず、1 球の賞球も払い出されない場合には、賞球の払出装に何らかの異常が発生している。このようにカウント異常監視処理（S 1 0）では、スイッチ読込処理（S 9）によって読み込まれたスイッチデータに基づいて、上記のような異常の有無を監視している。

#### 【0036】

図柄カウンタ更新処理（S 1 1）では、LCDディスプレイ 3 で行われる変動表示の結果、停止表示される図柄を決定するためのカウンタの更新処理が行われる。また、図柄チェック処理（S 1 2）では、図柄カウンタ更新処理（S 1 1）で更新されたカウンタの値に基づいて、特別図柄変動処理（S 1 6）で使用される大当たり図柄や、はずれ図柄、更にはリーチ図柄などが決定される。

#### 【0037】

S 3 から S 1 2 までの処理において、エラーが発生していなければ（S 1 3：正常）、普通図柄変動処理（S 1 4）によって、7 セグメント LED の変動表示を行うと共に、その変動表示の結果、当たりが発生した場合には普通電動役物（図示せず）を所定時間開放する当たり処理を実行する。その後、状態フラグをチェックし（S 1 5）、LCDディスプレイ 3 の図柄の変動表示中であれば（S 1 5：図柄変動中）、特別図柄変動処理（S 1 6）によって、打球が図柄作動ゲート 4 を通過するタイミングで読みとられた乱数カウンタ 1 3 a の値に基づいて、大当たりか否かの判定が行われると共に、LCDディスプレイ 3 の表示図柄の変動処理を実行する。一方、状態フラグをチェックした結果、大当たり中であれば（S 1 5：大当たり中）、大入賞口 5 を開放するなどの大当たり処理（S 1 7）を実行する。更に、状態フラグをチェックした結果、図柄の変動中でも大当たり中でもなければ（S 1 5：その他）、S 1 6 及び S 1 7 の処理をスキップして、S 1 8 の表示データ作成処理へ移行する。なお、S 1 3 の処理において、エラーが確認された場合には（S 1 3：エラー）、S 1 4～S 1 7 の各処理をスキップして、S 1 8 の表示データ作成処理へ移行する。

#### 【0038】

表示データ作成処理（S 1 8）では、図柄の変動表示以外に LCDディスプレイ 3 に表示されるデモデータや、7 セグメント LED の表示データなどが作成され、ランプ・情報処理（S 1 9）では、保留球のランプデータをはじめ、各種のランプデータが作成される。効果音処理（S 2 0）では、遊技の状況に応じた効果音データが作成される。なお、これらの表示データおよび効果音データは、前記したポート出力処理（S 5）やタイマ割込処理によって各装置へ出力される。

## 【 0 0 3 9 】

効果音処理 ( S 2 0 ) の終了後は、次のリセット割込処理が発生するまでの残余時間の間、 S 1 1 と同一の処理である図柄カウンタ更新処理 ( S 2 1 ) と、 S 6 と同一の処理である初期値カウンタ更新処理 ( S 2 2 ) とを繰り返し実行する。 S 1 ~ S 2 0 の各処理の実行時間は遊技の状態に応じて変化するので、次のリセット割込処理が発生するまでの残余時間は、一定の時間ではなく、遊技の状態に応じて変化する。よって、かかる残余時間を使用して図柄カウンタ更新処理 ( S 2 1 ) を繰り返し実行することにより、停止図柄をランダムに変更することができる。また、かかる残余時間を使用して初期値カウンタ更新処理 ( S 2 2 ) を繰り返し実行することにより、乱数カウンタ 1 3 a の更新の初期値となる初期値カウンタ 1 3 b の値を「ぶら下げ基板」により把握不可能とすることができる。

## 【 0 0 4 0 】

図 4 は、乱数更新処理のフローチャートである。乱数更新処理 ( S 7 ) では、 C P U 1 1 の内部レジスタ 1 1 b を介して、乱数カウンタ 1 3 a の値を「 0 ~ 6 3 0 ( 0 ~ 2 7 6 h ) 」の範囲内で「 + 1 」ずつ更新すると共に、制御部 C で使用される他の乱数の更新を行っている。

## 【 0 0 4 1 】

まず、 2 バイトで構成される乱数カウンタ 1 3 a の値を 2 バイトの内部レジスタ 1 1 b へ書き込む ( S 3 1 ) 。内部レジスタ 1 1 b の値を 1 加算し ( S 3 2 ) 、加算後の内部レジスタ 1 1 b の値が「 6 3 1 」以上であるか否か、即ち、乱数カウンタ 1 3 a の更新範囲の値を超えている否かを調べる ( S 3 3 ) 。加算後の内部レジスタ 1 1 b の値が「 6 3 1 」以上であれば ( S 3 3 : Y e s ) 、更新範囲の値を超えているので、内部レジスタ 1 1 b の値を「 0 」クリアする ( S 3 4 ) 。一方、加算後の内部レジスタ 1 1 b の値が「 6 3 0 」以下であれば ( S 3 3 : N o ) 、更新範囲内の値であるので、 S 3 4 の処理をスキップして、 S 3 5 の処理へ移行する。

## 【 0 0 4 2 】

S 3 5 の処理では、更新後の内部レジスタ 1 1 b の値と初期値メモリ 1 3 c の値とが比較される。初期値メモリ 1 3 c には乱数カウンタ 1 3 a の更新の初期値が記憶されているので、両値が等しい場合には ( S 3 5 : Y e s ) 、乱数カウンタ 1 3 a の更新は一回り終了したということである。よって、かかる場合には、 2 バイトの初期値カウンタ 1 3 b の値を内部レジスタ 1 1 b へ書き込み ( S 3 6 ) 、その内部レジスタ 1 1 b の値を初期値メモリ 1 3 c 及び乱数カウンタ 1 3 a へ書き込んで ( S 3 7 , S 3 8 ) 、乱数カウンタ 1 3 a の更新の初期値を変更する。

## 【 0 0 4 3 】

一方、更新後の内部レジスタ 1 1 b の値と初期値メモリ 1 3 c の値とが等しくない場合には ( S 3 5 : N o ) 、乱数カウンタ 1 3 a の更新は未だ一回り終了していないので、 S 3 6 及び S 3 7 の処理をスキップして、 S 3 2 から S 3 4 の処理で更新された内部レジスタ 1 1 b の値を乱数カウンタ 1 3 a へ書き込み ( S 3 8 ) 、乱数カウンタ 1 3 a の更新を行う。その後は、制御部 C で使用される他の乱数の更新処理を行って ( S 3 9 ) 、この乱数更新処理を終了する。

## 【 0 0 4 4 】

図 5 は、初期値カウンタ更新処理のフローチャートである。初期値カウンタ更新処理 ( S 6 , S 2 2 ) では、 C P U 1 1 の内部レジスタ 1 1 b を介して、乱数カウンタ 1 3 a の更新の初期値をカウントする初期値カウンタ 1 3 b の値を、乱数カウンタ 1 3 a の更新範囲の「 0 ~ 6 3 0 ( 0 ~ 2 7 6 h ) 」の範囲内で「 + 1 」ずつ更新する。

## 【 0 0 4 5 】

まず、 2 バイトで構成される初期値カウンタ 1 3 b の値を 2 バイトの内部レジスタ 1 1 b へ書き込む ( S 4 1 ) 。内部レジスタ 1 1 b の値を 1 加算し ( S 4 2 ) 、加算後の内部レジスタ 1 1 b の値が「 6 3 1 」以上であるか否か、即ち、乱数カウンタ 1 3 a の更新範囲の値を超えている否かを調べる ( S 4 3 ) 。加算後の内部レジスタ 1 1 b の値が「 6 3 1 」以上であれば ( S 4 3 : Y e s ) 、乱数カウンタ 1 3 a の更新範囲の値を超えている



ので、内部レジスタ 11b の値を「0」クリアする (S44)。一方、加算後の内部レジスタ 11b の値が「630」以下であれば (S43: No)、乱数カウンタ 13a の更新範囲内の値であるので、S44 の処理をスキップして、S45 の処理へ移行する。

【0046】

S45 の処理では、68系 CPU 11 の 2 バイト書き込み命令によって、1 命令で、2 バイトの内部レジスタ 11b の値が、上位バイト、下位バイトの順に初期値カウンタ 13b へ書き込まれて (S45)、初期値カウンタ 13b の更新が行われる。なお、80系 CPU では、更新された 2 バイトの内部レジスタの値が 1 バイトずつ、上位バイト、下位バイトの順に 2 命令で初期値カウンタ 13b へ書き込まれる。

【0047】

前記した通り、初期値カウンタ更新処理は、リセット割込処理において、次のリセット割込が発生するまでの残余時間の間に繰り返し実行される (S22)。このため S45 の処理において、内部レジスタ 11b の上位バイトが初期値カウンタ 13b へ書き込まれた後であって下位バイトの書き込み前に、次のリセット割込が発生すると、内部レジスタ 11b の下位バイトの値を初期値カウンタ 13b へ書き込むことなく、図 3 の S1 の処理が実行される。

【0048】

例えば、初期値カウンタ 13b の値が「1FFh」であれば、S41 ~ S44 の処理によって、内部レジスタ 11b の値は「200h」とされる。S45 の処理によって、この内部レジスタ 11b の上位バイトが初期値カウンタ 13b へ書き込まれると、初期値カウンタ 13b の値は、一旦「2FFh」となり、乱数カウンタ 13a の更新範囲外の値となる。本来ならば、その直後に、内部レジスタ 11b の下位バイトが初期値カウンタ 13b へ書き込まれ、初期値カウンタ 13b の値は「200h」の値になるのだが、前記したように、下位バイトの書き込み前に次のリセット割込が発生すると、下位バイトの書き込みは中止され、初期値カウンタ 13b の値は「2FFh」のままになってしまう。

【0049】

かかる初期値カウンタ 13b の値を、乱数更新処理 (S7) の S36 の処理で使用すると、乱数カウンタ 13a の更新の初期値を更新の範囲外の値にしてしまう。しかし、本実施例では、初期値カウンタ更新処理を、乱数更新処理 (S7) の前にも実行するようにしている (S6)、初期値カウンタ 13b の値が乱数カウンタ 13a の更新範囲外の値となった場合にも、その値を乱数更新処理 (S7) の前に、乱数カウンタ 13a の更新範囲内の値に戻すことができる。初期値カウンタ更新処理では、「631」以上の値は「0」クリアされるからである (S43: Yes, S44)。よって、乱数カウンタ 13a の値を常に所定の更新範囲内で更新することができるのである。

【0050】

また、本実施例のように、初期値カウンタ更新処理を残余時間のみならず (S22)、乱数更新処理 (S7) の前にも実行することにより (S6)、残余時間が不足する場合であっても、初期値カウンタ 13b の更新を確実に行うことができるのである。

【0051】

図 6 は、第 2 実施例における初期値カウンタ更新処理のフローチャートである。第 1 実施例の初期値カウンタ更新処理が、初期値カウンタ 13b の値を加算方向に「+1」ずつ更新し、且つ、更新後の値を上位バイト下位バイトの順に初期値カウンタ 13b へ書き込むのに対し、第 2 実施例の初期値カウンタ更新処理では、初期値カウンタ 13b の値を減算方向に「-1」ずつ更新し、且つ、更新後の値を下位バイト上位バイトの順に初期値カウンタ 13b へ書き込んでいる。なお、前記した第 1 実施例と同一の部分には同一の符号を付し、その説明は省略する。

【0052】

まず、2 バイトで構成される初期値カウンタ 13b の値を 2 バイトの内部レジスタ 11b へ書き込む (S51)。内部レジスタ 11b の値を 1 減算し (S52)、減算後の内部レジスタ 11b の値が「631 (277h)」以上であるか否かを調べる (S53)。減

算前の内部レジスタ11bの値が「0」であれば、減算によってその値は「0 F F F F h」となる。よって、減算後の内部レジスタ11bの値が「0 F F F F h」も含めた「6 3 1 ( 2 7 7 h )」以上であれば ( S 5 3 : Y e s )、乱数カウンタ13aの更新範囲の最大値である「6 3 0 ( 2 7 6 h )」を内部レジスタ11bへ書き込む ( S 5 4 )。一方、減算後の内部レジスタ11bの値が「6 3 0 ( 2 7 6 h )」以下であれば ( S 5 3 : N o )、S 5 4 の処理をスキップして、S 5 5 の処理へ移行する。

【 0 0 5 3 】

S 5 5 の処理では、減算方向に更新された2バイトの内部レジスタ11bの下位バイトの値を2バイトの初期値カウンタ13bの下位バイトへ書き込み ( S 5 5 )、次に、2バイトの内部レジスタ11bの上位バイトの値を2バイトの初期値カウンタ13bの上位バイトへ書き込む ( S 5 6 )。即ち、更新された2バイトの内部レジスタ11bの値を1バイトずつ、下位バイト、上位バイトの順に2命令で初期値カウンタ13bへ書き込むのである。なお、80系CPUでは、2バイト書き込み命令によって、1命令で、2バイトの内部レジスタの値を、下位バイト、上位バイトの順に初期値カウンタ13bへ書き込むことができる。

【 0 0 5 4 】

前記した通り、初期値カウンタ更新処理は、リセット割込処理において、次のリセット割込が発生するまでの残余時間の間に繰り返し実行される ( S 2 2 )。このため、S 5 5 の処理によって内部レジスタ11bの下位バイトが初期値カウンタ13bへ書き込まれた後であって、S 5 6 の処理による上位バイトの書き込み前に、次のリセット割込が発生する可能性がある。リセット割込は、割込の優先順位が最も高く、割込処理の開始を禁止できないノンマスカブルな割込であるので、かかる場合には、S 5 6 の処理が行われないうちに、初期値カウンタ更新処理が強制終了され、図3のS 1 の処理が実行される。これにより、書き込み途中の値が更新された初期値カウンタ13bの値になってしまう。

【 0 0 5 5 】

例えば、初期値カウンタ13bの値が「2 0 0 h」である場合、S 5 1 からS 5 3 の処理によって、CPU11の内部レジスタ11bの値は「1 F F h」に更新される。更新後の値は、下位バイト上位バイトの順に内部レジスタ11bから初期値カウンタ13bへ書き込まれるので、初期値カウンタ13bの値は下位バイトの書き込みにより一旦「2 F F h」となり ( S 5 5 )、その後、上位バイトの書き込みにより「1 F F h」となって ( S 5 6 )、初期値カウンタ13bの更新が完了する。よって、次のリセット割込が、初期値カウンタ13bの下位バイトへの書き込み後であって上位バイトへの書き込み前に発生すると、初期値カウンタ13bの値は「2 F F h」となり、本来の更新範囲の値である「0 ~ 6 3 0 ( 0 ~ 2 7 6 h )」の範囲外の値になってしまう。

【 0 0 5 6 】

この初期値カウンタ13bの値を、乱数更新処理 ( S 7 ) のS 3 6 の処理で使用すると、乱数カウンタ13aの更新の初期値を更新の範囲外の値にしてしまう。しかし、本実施例では、初期値カウンタ更新処理を、乱数更新処理 ( S 7 ) の前にも実行するようにしているので ( S 6 )、初期値カウンタ13bの値が乱数カウンタ13aの更新範囲外の値となった場合にも、その値を乱数更新処理 ( S 7 ) の前に、乱数カウンタ13aの更新範囲内の値に戻すことができる。初期値カウンタ更新処理では、「6 3 1 ( 2 7 7 h )」以上の値は乱数カウンタ13aの更新範囲の最大値である「6 3 0 ( 2 7 6 h )」に戻されるからである ( S 5 3 : Y e s , S 5 4 )。よって、乱数カウンタ13aの値を常に所定の更新範囲内で更新することができる。

【 0 0 5 7 】

なお、上記各実施例において、請求項1記載の定期処理としてはノンマスカブルなりセット割込処理のうちS 1 からS 2 0 の処理が該当し、所定の処理としてはS 2 1 及びS 2 2 の処理が該当する。第1更新手段としては図4の乱数更新処理 ( S 7 ) のS 3 1 からS 3 4 及びS 3 8 の処理が該当する。また、第2更新手段としてはS 6 及びS 2 2 の初期値カウンタ更新処理が該当し、変更手段としてはS 3 5 からS 3 8 の処理が該当する。

## 【 0 0 5 8 】

以上、実施例に基づき本発明を説明したが、本発明は上記実施例に何ら限定されるものではなく、本発明の趣旨を逸脱しない範囲内で種々の改良変形が可能であることは容易に推察できるものである。

## 【 0 0 5 9 】

例えば、本実施例では、乱数カウンタ 1 3 a の更新を行う乱数更新処理 ( S 7 ) の前に、初期値カウンタ更新処理 ( S 6 ) を実行して、初期値カウンタ 1 3 b の値を乱数カウンタ 1 3 a の更新範囲内の値に復帰させていた。しかしながら、必ずしも係る方式に限られるものではなく、例えば、初期値カウンタ 1 3 b の値を乱数カウンタ 1 3 a および初期値メモリ 1 3 c へ書き込む前に ( S 3 6 の処理の前に )、その初期値カウンタ 1 3 b の値をチェックして、その値が乱数カウンタ 1 3 a の更新範囲外の「 6 3 1 」以上である場合には、初期値カウンタ 1 3 b へ「 0 」または「 6 3 0 」を書き込み、初期値カウンタ 1 3 b の値を乱数カウンタ 1 3 a の更新範囲内の値に復帰させ、その復帰した値を乱数カウンタ 1 3 a および初期値メモリ 1 3 c へ書き込むようにしても良いのである。

## 【 0 0 6 0 】

また、初期値カウンタ 1 3 b の値の更新は、その値を C P U 1 1 の内部レジスタ 1 1 b へ一旦読み込んだ後に行われたが、内部レジスタ 1 1 b を介さずに、C P U 1 1 内の A L U へ直接読み込んで更新するようにしても良い。

## 【 0 0 6 1 】

以下に本発明の変形例を示す。請求項 1 または 2 に記載の弾球遊技機において、前記復帰手段は前記初期値カウンタの値を前記乱数カウンタの更新の範囲内の値で更新する第 3 更新手段によって構成されると共に、その第 3 更新手段は前記第 1 更新手段の実行前に少なくとも 1 回実行されることを特徴とする弾球遊技機 1。

## 【 0 0 6 2 】

請求項 1 に記載の弾球遊技機または弾球遊技機 1 において、前記第 2 更新手段による更新後の値の前記初期値カウンタへの書き込みは、上位バイト下位バイトの順に 2 バイトの書き込みが 1 命令で行われる書き込み命令により実行されることを特徴とする弾球遊技機 2。

## 【 0 0 6 3 】

請求項 1 若しくは 2 に記載の弾球遊技機または弾球遊技機 1 若しくは 2 において、前記初期値カウンタの値は、前記乱数カウンタの値が前記初期値メモリの値と一致する場合に、前記乱数カウンタおよび初期値メモリに書き込まれることを特徴とする弾球遊技機 3。

## 【 図面の簡単な説明 】

## 【 0 0 6 4 】

【 図 1 】 本発明の第 1 実施例であるパチンコ遊技機の遊技盤の正面図である。

【 図 2 】 パチンコ遊技機の電氣的構成を示したブロック図である。

【 図 3 】 リセット割込処理を示したフローチャートである。

【 図 4 】 乱数更新処理を示したフローチャートである。

【 図 5 】 初期値カウンタ更新処理を示したフローチャートである。

【 図 6 】 第 2 実施例の初期値カウンタ更新処理を示したフローチャートである。

## 【 符号の説明 】

## 【 0 0 6 5 】

1 1	制御部の C P U
1 1 b	制御部の C P U の内部レジスタ
1 3	制御部の R A M
1 3 a	乱数カウンタ
1 3 b	初期値カウンタ
1 3 c	初期値メモリ
C	制御部 ( 制御手段 )
P	パチンコ遊技機 ( 遊技機 )