

(19)日本国特許庁(JP)

(12)特許公報(B2)

(11)特許番号
特許第7061209号
(P7061209)

(45)発行日 令和4年4月27日(2022.4.27)

(24)登録日 令和4年4月19日(2022.4.19)

(51)国際特許分類	F I			
A 6 3 F 5/04 (2006.01)	A 6 3 F	5/04	6 1 1 B	
	A 6 3 F	5/04	6 1 2	

請求項の数 1 (全269頁)

(21)出願番号	特願2021-5808(P2021-5808)	(73)特許権者	598098526
(22)出願日	令和3年1月18日(2021.1.18)		株式会社ユニバーサルエンターテインメント
(62)分割の表示	特願2019-192433(P2019-192433)の分割		東京都江東区有明三丁目7番26号 有明フロンティアビルA棟
原出願日	平成28年8月15日(2016.8.15)	(74)代理人	110000925
(65)公開番号	特開2021-49477(P2021-49477A)		特許業務法人信友国際特許事務所
(43)公開日	令和3年4月1日(2021.4.1)	(72)発明者	大石 康幸
審査請求日	令和3年1月18日(2021.1.18)		東京都江東区有明3丁目7番26号
		審査官	櫻井 茂樹

最終頁に続く

(54)【発明の名称】 遊技機

(57)【特許請求の範囲】

【請求項1】

遊技動作を制御するための演算処理を行う演算処理手段と、
前記演算処理手段による前記演算処理の実行に必要な情報が記憶された第1記憶手段と、
前記演算処理手段による前記演算処理の実行に必要な情報が記憶される第2記憶手段と、
を備え、
前記演算処理手段は、
演算処理の実行時にデータが格納される複数の汎用レジスタと、
演算処理の実行時に格納されたデータにより指定可能なメモリ空間のアドレスの一部を指定する1バイトの拡張レジスタと、
内部当籤役を決定する内部抽籤手段と、を有し、
前記内部当籤役の期待値を調整するための設定値を記憶する設定値記憶領域が、前記第2記憶手段に含まれ、
前記演算処理手段は、前記第2記憶手段の2バイトのアドレスの上位側1バイトの値を、
前記拡張レジスタを用いて指定する複数種類の命令を実行可能であり、
前記内部抽籤手段は、抽籤処理内において、前記設定値により抽籤確率が異なる内部当籤役を抽籤する場合、前記設定値記憶領域に対して前記拡張レジスタを用いた加算命令を実行して判定データを算出し、算出した前記判定データに基づいて当該内部当籤役を抽籤するための抽籤値を取得し、
前記判定データは、前記取得された抽籤値が格納されたアドレスである

ことを特徴とする遊技機。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、遊技機に関する。

【背景技術】

【0002】

従来、複数の図柄がそれぞれの表面に設けられた複数のリールと、スタートスイッチと、ストップスイッチと、各リールに対応して設けられたステッピングモータと、制御部とを備えた、パチスロと呼ばれる遊技機が知られている。スタートスイッチは、メダルやコインなどの遊技媒体が遊技機に投入された後、スタートレバーが遊技者により操作されたこと（以下、「開始操作」ともいう）を検出し、全てのリールの回転の開始を要求する信号を出力する。ストップスイッチは、各リールに対応して設けられたストップボタンが遊技者により押されたこと（以下、「停止操作」ともいう）を検出し、該当するリールの回転の停止を要求する信号を出力する。ステッピングモータは、その駆動力を対応するリールに伝達する。また、制御部は、スタートスイッチ及びストップスイッチにより出力された信号に基づいて、ステッピングモータの動作を制御し、各リールの回転動作及び停止動作を行う。

10

【0003】

このような遊技機では、開始操作が検出されると、プログラム上で乱数を用いた抽籤処理（以下、「内部抽籤処理」という）が行われ、その抽籤の結果（以下、「内部当籤役」という）と停止操作のタイミングとに基づいてリールの回転の停止を行う。そして、全てのリールの回転が停止され、入賞の成立に係る図柄の組合せ（表示役）が表示されると、その図柄の組合せに対応する特典が遊技者に付与される。なお、遊技者に付与される特典の例としては、遊技媒体（メダル等）の払い出し、遊技媒体を消費することなく再度、内部抽籤処理を行う再遊技（以下、「リプレイ」ともいう）の作動、遊技媒体の払い出し機会が増加するボーナスゲームの作動等を挙げることができる。

20

【0004】

また、従来、上記構成の遊技機において、特定の小役（遊技媒体の払出に係る役）の成立をランプ等でナビゲートする機能、すなわち、アシストタイム（以下、「AT」という）の機能を備える遊技機が開発されている。また、従来、特定の図柄組合せが表示された場合にリプレイの当籤確率が通常時より高い遊技状態が作動する機能、すなわち、リプレイタイム（以下、「RT」という）の機能を備える遊技機も開発されている。さらに、従来、ATとRTとが同時に作動するアシストリプレイタイム（以下、「ART」という）の機能を備えたパチスロが開発されている。

30

【0005】

上述した遊技機は、通常、内部当籤役の決定、各リールの回転及び停止、入賞の有無の判定等の遊技機の主な遊技動作を制御する回路（主制御回路）が実装された主制御基板と、映像の表示等による演出動作を制御する回路（副制御回路）が実装された副制御基板とを備える。そして、遊技動作は、主制御回路に搭載されたCPU（Central Processing Unit）により制御される。この際、CPUの制御により、主制御回路のROM（Read Only Memory）に記憶されたプログラム及び各種テーブルデータ等が主制御回路のRAM（Random Access Memory）に展開され、各種遊技動作に関する処理が実行される。

40

【0006】

また、従来、上述した構成の遊技機において、スタックポインタを操作命令で使用して、数値データを処理する主制御装置が搭載された遊技機が知られている（例えば、特許文献1参照）。

【先行技術文献】

【特許文献】

【0007】

50

【文献】特開 2 0 0 5 - 2 3 7 7 3 7 号公報

【発明の概要】

【発明が解決しようとする課題】

【0008】

ところで、従来、上述した遊技機特有の制限として、主制御回路のプログラム容量が、規則により小容量に制限されている。さらに、近年、遊技性の複雑化により主制御回路のROMの容量が圧迫されており、主制御回路で管理する処理プログラムやテーブルなどの容量削減が求められている。

【0009】

本発明は、上記課題を解決するためになされたものであり、本発明の目的は、主制御回路で管理する処理プログラムやテーブルなどの容量を削減して主制御回路のROMの空き容量を増やし、該増えた容量分のROMの空き領域を利用して遊技性を高めることが可能な遊技機を提供することである。

10

【課題を解決するための手段】

【0010】

上記課題を解決するために、本発明では、以下のような構成の遊技機を提供する。

【0011】

遊技動作を制御するための演算処理を行う演算処理手段（例えば、後述のメインCPU101）と、

前記演算処理手段による前記演算処理の実行に必要な情報が記憶された第1記憶手段（例えば、後述のメインROM102）と、

20

前記演算処理手段による前記演算処理の実行に必要な情報が記憶される第2記憶手段（例えば、後述のメインRAM103）と、を備え、

前記演算処理手段は、

演算処理の実行時にデータが格納される複数の汎用レジスタと、

演算処理の実行時に格納されたデータにより指定可能なメモリ空間のアドレスの一部を指定する1バイトの拡張レジスタ（例えば、後述のQレジスタ）と、

内部当籤役を決定する内部抽籤手段（例えば、後述の内部抽籤処理）と、を有し、

前記内部当籤役の期待値を調整するための設定値を記憶する設定値記憶領域が、前記第2記憶手段に含まれ、

30

前記演算処理手段は、前記第2記憶手段の2バイトのアドレスの上位側1バイトの値を、前記拡張レジスタを用いて指定する複数種類の命令（例えば、後述の「LDQ」命令、「SETQ」命令、「ADDQ」命令等）を実行可能であり、

前記内部抽籤手段は、抽籤処理内において、前記設定値により抽籤確率が異なる内部当籤役を抽籤する場合、前記設定値記憶領域に対して前記拡張レジスタを用いた加算命令（例えば、後述の「ADDQ」命令）を実行して判定データを算出し、算出した前記判定データに基づいて当該内部当籤役を抽籤するための抽籤値を取得し、

前記判定データは、前記取得された抽籤値が格納されたアドレスである

ことを特徴とする遊技機。

【発明の効果】

40

【0012】

上記構成の本発明の遊技機によれば、主制御回路で管理する処理プログラムやテーブルなどの容量を削減して主制御回路のROM（第1記憶手段）の空き容量を増やし、該増えた容量分のROMの空き領域を利用して遊技性を高めることができる。

【図面の簡単な説明】

【0013】

【図1】本発明の一実施形態における遊技機の機能フローを説明するための図である。

【図2】本発明の一実施形態における遊技機の外觀構造を示す斜視図である。

【図3】本発明の一実施形態における遊技機の内部構造を示す図である。

【図4】本発明の一実施形態における遊技機の内部構造を示す図である。

50

【図 5】本発明の一実施形態のサブ表示装置に表示される各種表示画面の概略構成を示す図である。

【図 6】本発明の一実施形態におけるサブ表示装置の表示画面の遷移例を示す図である。

【図 7】本発明の一実施形態の遊技機が備える回路の全体構成を示すブロック図である。

【図 8】本発明の一実施形態における主制御回路の内部構成を示すブロック図である。

【図 9】本発明の一実施形態におけるマイクロプロセッサの内部構成を示すブロック図である。

【図 10】本発明の一実施形態における副制御回路の内部構成を示すブロック図である。

【図 11】本発明の一実施形態におけるメイン CPU が有する各種レジスタの構成図である。

10

【図 12】本発明の一実施形態における主制御回路のメモリマップを示す図である。

【図 13】本発明の一実施形態におけるパチスロのボーナス状態及び非ボーナス状態間における遊技状態の遷移フローを示す図である。

【図 14】本発明の一実施形態におけるパチスロの ART 遊技状態、非 ART 遊技状態及びボーナス状態間における遊技状態の遷移フローを示す図である。

【図 15】本発明の一実施形態における図柄配置テーブルの一例を示す図である。

【図 16】本発明の一実施形態における内部抽籤テーブルの一例を示す図である。

【図 17】本発明の一実施形態における内部抽籤テーブルの一例を示す図である。

【図 18】本発明の一実施形態における図柄組合せ決定テーブルの一例を示す図である。

【図 19】本発明の一実施形態における図柄組合せ決定テーブルの一例を示す図である。

20

【図 20】本発明の一実施形態における図柄組合せ決定テーブルの一例を示す図である。

【図 21】本発明の一実施形態における図柄組合せ決定テーブルの一例を示す図である。

【図 22】本発明の一実施形態における図柄組合せ決定テーブルの一例を示す図である。

【図 23】本発明の一実施形態における図柄組合せ決定テーブルの一例を示す図である。

【図 24】本発明の一実施形態における内部当籤役と停止図柄組合せとの対応関係を示す図である。

【図 25】本発明の一実施形態における内部当籤役と停止図柄組合せとの対応関係を示す図である。

【図 26】本発明の一実施形態におけるリール停止初期設定テーブルの一例を示す図である。

30

【図 27】本発明の一実施形態における引込優先順位テーブルの一例を示す図である。

【図 28】本発明の一実施形態における当り要求フラグ格納領域、入賞作動フラグ格納領域の構成（その 1）を示す図である。

【図 29】本発明の一実施形態における当り要求フラグ格納領域、入賞作動フラグ格納領域の構成（その 2）を示す図である。

【図 30】本発明の一実施形態における当り要求フラグ格納領域、入賞作動フラグ格納領域の（その 3）を示す図である。

【図 31】本発明の一実施形態における持越役格納領域の構成を示す図である。

【図 32】本発明の一実施形態における遊技状態フラグ格納領域の構成を示す図である。

【図 33】本発明の一実施形態における作動ストップボタン格納領域の構成を示す図である。

40

【図 34】本発明の一実施形態における押下順序格納領域の構成を示す図である。

【図 35】本発明の一実施形態における図柄コード格納領域の構成を示す図である。

【図 36】本発明の一実施形態における内部当籤役とサブフラグとの対応表（その 1）を示す図である。

【図 37】本発明の一実施形態における内部当籤役とサブフラグとの対応表（その 2）を示す図である。

【図 38】本発明の一実施形態の遊技機において、サブフラグ EX「3 連チリリブ」又は「リーチ目リブ」が当籤した際の報知動作を説明するための図である。

【図 39】本発明の一実施形態における一般遊技状態中の遊技の流れを説明するための図

50

である。

【図 4 0】本発明の一実施形態における通常中高確率抽籤テーブルの一例を示す図である。

【図 4 1】本発明の一実施形態における C Z 抽籤テーブルの一例を示す図である。

【図 4 2】本発明の一実施形態における C Z 1 中モードアップ抽籤テーブルの一例を示す図である。

【図 4 3】本発明の一実施形態における C Z 2 中ポイント抽籤テーブルの一例を示す図である。

【図 4 4】本発明の一実施形態における C Z 中 A R T 抽籤テーブル (C Z 1 , C Z 2 用) の一例を示す図である。

【図 4 5】本発明の一実施形態における C Z 中 A R T 抽籤テーブル (C Z 3 用) の一例を示す図である。

10

【図 4 6】本発明の一実施形態における通常 A R T 中の遊技の流れを説明するための図である。

【図 4 7】本発明の一実施形態における A R T 中フラグ変換抽籤テーブルの一例を示す図である。

【図 4 8】本発明の一実施形態における A R T レベル決定テーブルの一例を示す図である。

【図 4 9】本発明の一実施形態における通常 A R T 中高確率抽籤テーブルの一例を示す図である。

【図 5 0】本発明の一実施形態における A R T 中 C T 抽籤テーブルの一例を示す図である。

【図 5 1】本発明の一実施形態における通常 A R T 中上乗せ抽籤テーブルの一例を示す図である。

20

【図 5 2】本発明の一実施形態における C T 状態中の遊技の流れを説明するための図である。

【図 5 3】本発明の一実施形態における C T 中テーブル抽籤テーブルの一例を示す図である。

【図 5 4】本発明の一実施形態における C T 中フラグ変換抽籤テーブルの一例を示す図である。

【図 5 5】本発明の一実施形態における C T 中上乗せ抽籤テーブルの一例を示す図である。

【図 5 6】本発明の一実施形態における C T 中セット数上乗せ抽籤テーブルの一例を示す図である。

30

【図 5 7】本発明の一実施形態におけるボーナス状態中の遊技の流れを説明するための図である。

【図 5 8】本発明の一実施形態におけるボーナス種別抽籤テーブルの一例を示す図である。

【図 5 9】本発明の一実施形態におけるボーナス中 A R T ゲーム数上乗せ抽籤テーブルの一例を示す図である。

【図 6 0】本発明の一実施形態におけるボーナス終了時 C T 抽籤テーブルの一例を示す図である。

【図 6 1】本発明の一実施形態における一般遊技状態中の遊技 (その他) の流れを説明するための図である。

【図 6 2】本発明の一実施形態における非 A R T 中フラグ変換抽籤テーブルの一例を示す図である。

40

【図 6 3】本発明の一実施形態におけるメイン側ナビデータとサブ側ナビデータとの対応関係を示す図である。

【図 6 4】本発明の一実施形態における遊技機の主制御回路により実行される電源投入 (リセット割込み) 時処理の例を示すフローチャートである。

【図 6 5】本発明の一実施形態における電源投入時処理のフローチャート中の各種処理を実行するためのソースプログラムの一例を示す図である。

【図 6 6】本発明の一実施形態における遊技復帰処理の例を示すフローチャートである。

【図 6 7】本発明の一実施形態における遊技復帰処理のフローチャート中の各種処理を実行するためのソースプログラムの一例を示す図である。

50

【図 6 8】本発明の一実施形態における設定変更確認処理の例を示すフローチャートである。

【図 6 9】本発明の一実施形態における設定変更確認処理のフローチャート中の各種処理を実行するためのソースプログラムの一例を示す図である。

【図 7 0】本発明の一実施形態における設定変更コマンド生成格納処理の例を示すフローチャートである。

【図 7 1】本発明の一実施形態における設定変更コマンド生成格納処理のフローチャート中の各種処理を実行するためのソースプログラムの一例を示す図である。

【図 7 2】本発明の一実施形態における通信データ格納処理の例を示すフローチャートである。

【図 7 3】本発明の一実施形態における通信データ格納処理のフローチャート中の各種処理を実行するためのソースプログラムの一例を示す図である。

【図 7 4】本発明の一実施形態における通信データポインタ更新処理の例を示すフローチャートである。

【図 7 5】本発明の一実施形態における通信データポインタ更新処理のフローチャート中の各種処理を実行するためのソースプログラムの一例を示す図である。

【図 7 6】本発明の一実施形態における電断時（外部）処理の例を示すフローチャートである。

【図 7 7】本発明の一実施形態におけるチェックサム生成処理（規定外）の例を示すフローチャートである。

【図 7 8】本発明の一実施形態におけるチェックサム生成処理のフローチャート中の各種処理を実行するためのソースプログラムの一例、並びに、チェックサム生成処理で実行されるスタックポインタの更新動作及びレジスタへのデータの読み出し動作の様子を示す図である。

【図 7 9】本発明の一実施形態におけるサムチェック処理（規定外）の例を示すフローチャートである。

【図 8 0】本発明の一実施形態におけるサムチェック処理（規定外）の例を示すフローチャートである。

【図 8 1】本発明の一実施形態におけるサムチェック処理のフローチャート中の各種処理を実行するためのソースプログラムの一例を示す図である。

【図 8 2】本発明の一実施形態における遊技機の主制御回路により実行されるメイン処理（主要動作処理）の例を示すフローチャートである。

【図 8 3】本発明の一実施形態におけるメダル受付・スタートチェック処理の例を示すフローチャートである。

【図 8 4】本発明の一実施形態におけるメダル受付・スタートチェック処理のフローチャート中の各種処理を実行するためのソースプログラムの一例を示す図である。

【図 8 5】本発明の一実施形態におけるメダル投入処理の例を示すフローチャートである。

【図 8 6】本発明の一実施形態におけるメダル投入処理のフローチャート中の各種処理を実行するためのソースプログラムの一例を示す図である。

【図 8 7】本発明の一実施形態におけるメダル投入チェック処理の例を示すフローチャートである。

【図 8 8】本発明の一実施形態におけるメダル投入チェック処理のフローチャート中の各種処理を実行するためのソースプログラムの一例を示す図である。

【図 8 9】本発明の一実施形態におけるエラー処理の例を示すフローチャートである。

【図 9 0】本発明の一実施形態におけるエラー処理のソースプログラム上で、実際に参照されるエラーテーブルの構成の一例を示す図である。

【図 9 1】本発明の一実施形態における乱数取得処理の例を示すフローチャートである。

【図 9 2】本発明の一実施形態における内部抽籤処理の例を示すフローチャートである。

【図 9 3】本発明の一実施形態における内部抽籤処理のフローチャート中の各種処理を実行するためのソースプログラムの一例を示す図である。

10

20

30

40

50

【図 9 4】本発明の一実施形態における内部抽籤処理のソースプログラム上で、実際に参照される内部抽籤テーブル（一般遊技用）の構成の一例を示す図である。

【図 9 5】本発明の一実施形態における内部抽籤処理のソースプログラム上で、実際に参照される R T 状態別抽籤値選択テーブルの構成の一例を示す図である。

【図 9 6】本発明の一実施形態における内部抽籤処理のソースプログラム上で、実際に参照される、内部抽籤値テーブル選択テーブル、1 バイト内部抽籤値テーブル、2 バイト内部抽籤値テーブル、1 バイト設定別内部抽籤値テーブル及び 2 バイト設定別内部抽籤値テーブルの構成の一例を示す図である。

【図 9 7】本発明の一実施形態における図柄設定処理の例を示すフローチャートである。

【図 9 8】本発明の一実施形態における特賞（ボーナス）当籤番号及び小役当籤番号と、内部当籤役との対応を示す図である。

10

【図 9 9】本発明の一実施形態における図柄設定処理のフローチャート中の各種処理を実行するためのソースプログラムの一例を示す図である。

【図 1 0 0】本発明の一実施形態における図柄設定処理のソースプログラム上で、実際に参照される当り要求フラグテーブルの構成の一例を示す図である。

【図 1 0 1】本発明の一実施形態における圧縮データ格納処理の例を示すフローチャートである。

【図 1 0 2】本発明の一実施形態における第 2 インターフェースボード制御処理（規定外）の例を示すフローチャートである。

【図 1 0 3】本発明の一実施形態における第 2 インターフェースボード出力処理の例を示すフローチャートである。

20

【図 1 0 4】本発明の一実施形態における状態別制御処理の例を示すフローチャートである。

【図 1 0 5】本発明の一実施形態におけるサブフラグ変換処理の例を示すフローチャートである。

【図 1 0 6】本発明の一実施形態におけるサブフラグ変換処理のフローチャート中の各種処理を実行するためのソースプログラムの一例を示す図である。

【図 1 0 7】本発明の一実施形態におけるサブフラグ変換処理のソースプログラム上で、実際に参照されるサブフラグ変換テーブルの構成の一例を示す図である。

【図 1 0 8】本発明の一実施形態におけるナビセット処理の例を示すフローチャートである。

30

【図 1 0 9】本発明の一実施形態におけるナビセット処理のフローチャート中の各種処理を実行するためのソースプログラムの一例を示す図である。

【図 1 1 0】本発明の一実施形態におけるナビセット処理のソースプログラム上で、実際に参照されるナビデータテーブルの構成の一例を示す図である。

【図 1 1 1】本発明の一実施形態におけるフラグ変換処理の例を示すフローチャートである。

【図 1 1 2】本発明の一実施形態における通常中スタート時処理の例を示すフローチャートである。

【図 1 1 3】本発明の一実施形態における C Z 中スタート時処理の例を示すフローチャートである。

40

【図 1 1 4】本発明の一実施形態における C Z 1（C Z 2）中処理の例を示すフローチャートである。

【図 1 1 5】本発明の一実施形態における C Z 1（C Z 2）中処理の例を示すフローチャートである。

【図 1 1 6】本発明の一実施形態における C Z 3 中処理の例を示すフローチャートである。

【図 1 1 7】本発明の一実施形態における通常 A R T 中スタート時処理の例を示すフローチャートである。

【図 1 1 8】本発明の一実施形態における C T 中スタート時処理の例を示すフローチャートである。

50

【図 1 1 9】本発明の一実施形態における C T 中 C T 抽籤処理の例を示すフローチャートである。

【図 1 2 0】本発明の一実施形態におけるテーブルデータ取得処理の例を示すフローチャートである。

【図 1 2 1】本発明の一実施形態におけるテーブルデータ取得処理のフローチャート中の各種処理を実行するためのソースプログラムの一例を示す図である。

【図 1 2 2】本発明の一実施形態におけるテーブルデータ取得処理のソースプログラム上で、実際に参照される C T 中 C T 抽籤テーブルの構成の一例を示す図である。

【図 1 2 3】本発明の一実施形態における 1 バイト抽籤処理の例を示すフローチャートである。

10

【図 1 2 4】本発明の一実施形態における 1 バイト抽籤処理のフローチャート中の各種処理を実行するためのソースプログラムの一例を示す図である。

【図 1 2 5】本発明の一実施形態における B B 中スタート時処理の例を示すフローチャートである。

【図 1 2 6】本発明の一実施形態における引込優先順位格納処理の例を示すフローチャートである。

【図 1 2 7】本発明の一実施形態における引込優先順位格納処理のフローチャート中の各種処理を実行するためのソースプログラムの一例を示す図である。

【図 1 2 8】本発明の一実施形態における図柄コード取得処理の例を示すフローチャートである。

20

【図 1 2 9】本発明の一実施形態における図柄コード取得処理のフローチャート中の各種処理を実行するためのソースプログラムの一例を示す図である。

【図 1 3 0】本発明の一実施形態における図柄コード取得処理のソースプログラム上で、実際に参照される第 1 回胴（左リール）図柄配置テーブル、及び、第 1 回胴図柄配置テーブルセット時に参照される図柄対応入賞作動テーブルの構成の一例を示す図である。

【図 1 3 1】本発明の一実施形態における図柄コード取得処理のソースプログラム上で、実際に参照される第 2 回胴（中リール）図柄配置テーブル、及び、第 2 回胴図柄配置テーブルセット時に参照される図柄対応入賞作動テーブルの構成の一例を示す図である。

【図 1 3 2】本発明の一実施形態における図柄コード取得処理のソースプログラム上で、実際に参照される第 3 回胴（右リール）図柄配置テーブル、及び、第 3 回胴図柄配置テーブルセット時に参照される図柄対応入賞作動テーブルの構成の一例を示す図である。

30

【図 1 3 3】本発明の一実施形態における論理積演算処理の例を示すフローチャートである。

【図 1 3 4】本発明の一実施形態における引込優先順位取得処理の例を示すフローチャートである。

【図 1 3 5】本発明の一実施形態における引込優先順位取得処理の例を示すフローチャートである。

【図 1 3 6】本発明の一実施形態における引込優先順位取得処理のフローチャート中の各種処理を実行するためのソースプログラムの一例を示す図である。

【図 1 3 7】本発明の一実施形態における引込優先順位取得処理のソースプログラム上で、実際に参照される引込優先順位テーブルの構成の一例を示す図である。

40

【図 1 3 8】本発明の一実施形態におけるリール停止制御処理の例を示すフローチャートである。

【図 1 3 9】本発明の一実施形態におけるリール停止制御処理のフローチャート中の各種処理を実行するためのソースプログラムの一例を示す図である。

【図 1 4 0】本発明の一実施形態におけるリール停止制御処理のフローチャート中の各種処理を実行するためのソースプログラムの一例を示す図である。

【図 1 4 1】本発明の一実施形態におけるリール停止可能信号 O F F 処理（規定外）の例を示すフローチャートである。

【図 1 4 2】本発明の一実施形態におけるリール停止可能信号 O N 処理（規定外）の例を

50

示すフローチャートである。

【図 1 4 3】本発明の一実施形態における規定外ポート出力処理の例を示すフローチャートである。

【図 1 4 4】本発明の一実施形態における規定外ポート出力処理のフローチャート中の各種処理を実行するためのソースプログラムの一例を示す図である。

【図 1 4 5】本発明の一実施形態における入賞検索処理の例を示すフローチャートである。

【図 1 4 6】本発明の一実施形態における入賞検索処理のフローチャート中の各種処理を実行するためのソースプログラムの一例を示す図である。

【図 1 4 7】本発明の一実施形態における入賞検索処理のソースプログラム上で、実際に参照される払出枚数データテーブルの構成の一例を示す図である。

10

【図 1 4 8】本発明の一実施形態におけるイリーガルヒットチェック処理の例を示すフローチャートである。

【図 1 4 9】本発明の一実施形態におけるイリーガルヒットチェック処理のフローチャート中の各種処理を実行するためのソースプログラムの一例を示す図である。

【図 1 5 0】本発明の一実施形態における入賞チェック・メダル払出処理の例を示すフローチャートである。

【図 1 5 1】本発明の一実施形態における入賞チェック・メダル払出処理のフローチャート中の各種処理を実行するためのソースプログラムの一例を示す図である。

【図 1 5 2】本発明の一実施形態におけるメダル払出枚数チェック処理の例を示すフローチャートである。

20

【図 1 5 3】本発明の一実施形態におけるメダル払出枚数チェック処理のフローチャート中の各種処理を実行するためのソースプログラムの一例を示す図である。

【図 1 5 4】本発明の一実施形態における B B チェック処理の例を示すフローチャートである。

【図 1 5 5】本発明の一実施形態における R T チェック処理の例を示すフローチャートである。

【図 1 5 6】本発明の一実施形態における R T チェック処理の例を示すフローチャートである。

【図 1 5 7】本発明の一実施形態における C Z ・ A R T 終了時処理の例を示すフローチャートである。

30

【図 1 5 8】本発明の一実施形態における遊技機の主制御回路により実行される割込処理の例を示すフローチャートである。

【図 1 5 9】本発明の一実施形態における 7 セグ L E D 駆動処理の例を示すフローチャートである。

【図 1 6 0】本発明の一実施形態における 7 セグ L E D 駆動処理のフローチャート中の各種処理を実行するためのソースプログラムの一例を示す図である。

【図 1 6 1】本発明の一実施形態における 7 セグ表示データ生成処理の例を示すフローチャートである。

【図 1 6 2】本発明の一実施形態における 7 セグ表示データ生成処理のフローチャート中の各種処理を実行するためのソースプログラムの一例を示す図である。

40

【図 1 6 3】本発明の一実施形態における 7 セグ表示データ生成処理のソースプログラム上で、実際に参照される 7 セグカソードテーブルの構成の一例を示す図である。

【図 1 6 4】本発明の一実施形態におけるタイマー更新処理の例を示すフローチャートである。

【図 1 6 5】本発明の一実施形態におけるタイマー更新処理のフローチャート中の各種処理を実行するためのソースプログラムの一例を示す図である。

【図 1 6 6】本発明の一実施形態における試射試験信号制御処理（規定外）の例を示すフローチャートである。

【図 1 6 7】本発明の一実施形態における回胴制動信号生成処理の例を示すフローチャートである。

50

【図168】本発明の一実施形態における特賞信号制御処理の例を示すフローチャートである。

【図169】本発明の一実施形態における条件装置信号制御処理の例を示すフローチャートである。

【図170】本発明の一実施形態における条件装置信号制御処理の例を示すフローチャートである。

【図171】本発明の一実施形態における遊技機の副制御回路により実行されるサブ側ナビ制御処理の例を示すフローチャートである。

【図172】本発明の一実施形態における遊技者登録処理の例を示すフローチャートである。

【図173】本発明の一実施形態における履歴管理処理の例を示すフローチャートである。

【図174】本発明の変形例1におけるCT前兆中の遊技の流れを示す図である。

【図175】本発明の変形例2における内部当籤役と停止図柄組合せとの対応関係を示す図である。

【図176】本発明の変形例3におけるメイン側ナビデータとサブ側ナビデータとの対応関係を示す図である。

【図177】本発明の変形例5における押し順とロック状態との対応関係を示す図である。

【発明を実施するための形態】

【0014】

以下、本発明の一実施形態に係る遊技機としてパチスロを例に挙げ、図面を参照しながら、その構成及び動作について説明する。なお、本実施形態では、ボーナス作動機能及びART機能を備えたパチスロについて説明する。

【0015】

<機能フロー>

まず、図1を参照して、パチスロの機能フローについて説明する。本実施形態のパチスロでは、遊技を行うための遊技媒体としてメダルを用いる。なお、遊技媒体としては、メダル以外にも、例えば、コイン、遊技球、遊技用のポイントデータ又はトークン等を適用することもできる。

【0016】

遊技者によりパチスロにメダルが投入され、スタートレバーが操作されると、予め定められた数値範囲（例えば、0～65535）の乱数から1つの値（以下、乱数値という）が抽出される。

【0017】

内部抽籤手段は、抽出された乱数値に基づいて抽籤を行い、内部当籤役を決定する。この内部抽籤手段は、後述の主制御回路が備える各種処理手段（処理機能）の一つである。内部当籤役の決定により、後述の有効ライン（入賞判定ライン）に沿って表示を行うことを許可する図柄の組合せが決定される。なお、図柄の組合せの種別としては、メダルの払い出し、再遊技（リプレイ）の作動、ボーナスの作動等といった特典が遊技者に与えられる「入賞」に係るものと、それ以外のいわゆる「はずれ」に係るものとが設けられる。なお、以下では、メダルの払い出しに係る役を「小役」と称し、再遊技（リプレイ）の作動に係る役を「リプレイ役」と称する。また、ボーナスの作動（ボーナスゲーム）に係る役を「ボーナス役」ともいう。

【0018】

また、スタートレバーが操作されると、複数のリールの回転が行われる。その後、遊技者により所定のリールに対応するストップボタンが押されると、リール停止制御手段は、内部当籤役とストップボタンが押されたタイミングとに基づいて、該当するリールの回転を停止する制御を行う。このリール停止制御手段は、後述の主制御回路が備える各種処理手段（処理機能）の一つである。

【0019】

パチスロでは、基本的に、ストップボタンが押されたときから規定時間（190 msec

10

20

30

40

50

）内に、該当するリールの回転を停止する制御が行われる。本実施形態では、この規定時間内にリールの回転に伴って移動する図柄の数を「滑り駒数」という。そして、本実施形態では、規定期間が190 msecである場合には、滑り駒数の最大数（最大滑り駒数）を図柄4個分に定める。

【0020】

リール停止制御手段は、入賞に係る図柄の組合せ表示を許可する内部当籤役が決定されているときは、通常、190 msec（図柄4駒分）の規定時間内に、その図柄の組合せが有効ラインに沿って極力表示されるようにリールの回転を停止させる。また、リール停止制御手段は、規定時間を利用して、内部当籤役によってその表示が許可されていない図柄の組合せが有効ラインに沿って表示されないようにリールの回転を停止させる。

10

【0021】

このようにして、複数のリールの回転がすべて停止されると、入賞判定手段は、有効ラインに沿って表示された図柄の組合せが、入賞に係るものであるか否かの判定を行う。この入賞判定手段もまた、後述の主制御回路が備える各種処理手段（処理機能）の一つである。そして、表示された図柄の組合せが、入賞判定手段により入賞に係るものであると判定されると、メダルの払い出し等の特典が遊技者に与えられる。パチスロでは、以上のような一連の流れが1回の遊技（単位遊技）として行われる。

【0022】

また、パチスロでは、前述した一連の遊技動作の流れの中で、表示装置などによる映像の表示、各種ランプによる光の出力、スピーカによる音の出力、或いは、これらの組合せを利用して様々な演出が行われる。

20

【0023】

具体的には、スタートレバーが操作されると、上述した内部当籤役の決定に用いられた乱数値とは別に、演出用の乱数値が抽出される。演出用の乱数値が抽出されると、演出内容決定手段は、内部当籤役に対応づけられた複数種類の演出内容の中から今回実行する演出を抽籤により決定する。この演出内容決定手段は、後述の副制御回路が備える各種処理手段（処理機能）の一つである。

【0024】

次いで、演出内容決定手段により演出内容が決定されると、演出実行手段は、リールの回転開始時、各リールの回転停止時、入賞の有無の判定時等の各契機に連動させて対応する演出を実行する。このように、パチスロでは、例えば、内部当籤役に対応づけられた演出内容を実行することによって、決定された内部当籤役（言い換えると、狙うべき図柄の組合せ）を知る機会又は予想する機会が遊技者に提供され、遊技者の興味の向上を図ることができる。

30

【0025】

<パチスロの構造>

次に、図2～図4を参照して、本発明の一実施形態に係るパチスロの構造について説明する。

【0026】

[外観構造]

図2は、パチスロ1の外部構造を示す斜視図である。

40

【0027】

パチスロ1は、図2に示すように、外装体（遊技機本体）2を備える。外装体2は、リールや回路基板等を収容するキャビネット2aと、キャビネット2aの開口を開閉可能に取り付けられるフロントドア2bとを有する。

【0028】

キャビネット2aの内部には、3つのリール3L、3C、3R（変動表示手段、表示列）が横一列に並べて設けられている。以下、各リール3L、3C、3R（メインリール）を、それぞれ左リール3L、中リール3C、右リール3Rともいう。各リール3L、3C、3Rは、円筒状に形成されたリール本体と、リール本体の周面に装着された透光性のシー

50

ト材を有する。そして、シート材の表面には、複数（例えば20個）の図柄が周方向（リールの回転方向）に沿って所定の間隔をあけて描かれている。

【0029】

フロントドア2bは、ドア本体9と、フロントパネル10と、腰部パネル12と、台座部13とを備える。ドア本体9は、ヒンジ（不図示）を用いてキャビネット2aに開閉可能に取り付けられる。ヒンジは、パチスロ1の前方側（遊技者側）から見て、ドア本体9の左側の側端部に設けられる。

【0030】

フロントパネル10は、ドア本体9の上部に設けられている。このフロントパネル10は、開口10aを有する枠状部材で構成される。フロントパネル10の開口10aは、表示装置カバー30によって塞がれ、表示装置カバー30は、キャビネット2aの内部に配置された後述の表示装置11と対向して配置される。

10

【0031】

表示装置カバー30は、黒色の半透明な合成樹脂により形成される。それゆえ、遊技者は、後述の表示装置11により表示された映像（画像）を、表示装置カバー30を介して視認することができる。また、本実施形態では、表示装置カバー30を黒色の半透明な合成樹脂で形成することにより、キャビネット2a内への外光の入り込みを抑制して、表示装置11により表示された映像（画像）を鮮明に視認できるようにしている。

【0032】

フロントパネル10には、ランプ群21が設けられている。ランプ群21は、例えば、遊技者側から見て、フロントパネル10の上部に設けられたランプ21a、21bを含む。ランプ群21を構成する各ランプは、LED（Light Emitting Diode）等で構成され（後述の図7中のLED群85参照）、演出内容に対応するパターンで、光を点灯及び消灯する。

20

【0033】

腰部パネル12は、ドア本体9の略中央部に設けられる。腰部パネル12は、任意の画像が描かれた装飾パネルと、この装飾パネルを背面側から照明するための光を出射する光源（後述のLED群85に含まれるLED）とを有する。

【0034】

台座部13は、フロントパネル10と腰部パネル12との間に設けられる。台座部13には、図柄表示領域4と、遊技者による操作の対象となる各種装置（メダル投入口14、MAXベットボタン15a、1ベットボタン15b、スタートレバー16、3つのストップボタン17L、17C、17R、精算ボタン（不図示）等）とが設けられる。

30

【0035】

図柄表示領域4は、正面から見て、3つのリール3L、3C、3Rに重畳する領域で、かつ、3つのリール3L、3C、3Rより遊技者側の位置に配置されており、3つのリール3L、3C、3Rを視認可能にするサイズを有する。この図柄表示領域4は、表示窓としての機能を果たすものであり、その背後に設けられた各リール3L、3C、3Rを視認することが可能な構成になっている。以下、図柄表示領域4を、リール表示窓4という。

【0036】

リール表示窓4は、その背後に設けられた3つのリール3L、3C、3Rの回転が停止されたとき、各リールの周面に設けられた複数の図柄のうち、連続して配置された3つの図柄がその枠内に表示されるように構成されている。すなわち、3つのリール3L、3C、3Rの回転が停止されたとき、リール表示窓4の枠内には、リール毎に上段、中段及び下段の各領域にそれぞれ1個の図柄（合計で3個）が表示される（リール表示窓4の枠内には、3行×3列の態様で図柄が表示される）。そして、本実施形態では、リール表示窓4の枠内において、左リール3Lの中段領域、中リール3Cの中段領域、及び、右リール3Rの中段領域を結び擬似的なライン（センターライン）を、入賞か否かの判定を行う有効ラインとして定義する。

40

【0037】

50

リール表示窓 4 は、台座部 1 3 に設けられた枠部材 3 1 の開口により形成される。また、リール表示窓 4 を画成する枠部材 3 1 の下方には、略水平面の台座領域が設けられる。そして、遊技者側から見て、台座領域の右側にはメダル投入口 1 4 が設けられ、左側には MAX ベットボタン 1 5 a 及び 1 ベットボタン 1 5 b が設けられる。

【 0 0 3 8 】

メダル投入口 1 4 は、遊技者によって外部からパチスロ 1 に投下されるメダルを受け入れるために設けられる。メダル投入口 1 4 から受け入れられたメダルは、予め設定された所定枚数（例えば 3 枚）を上限として 1 回の遊技に使用され、所定枚数を越えたメダルの枚数分は、パチスロ 1 の内部に預けることができる（いわゆるクレジット機能（遊技媒体貯留手段））。

10

【 0 0 3 9 】

MAX ベットボタン 1 5 a 及び 1 ベットボタン 1 5 b は、キャビネット 2 a の内部に預けられているメダルから 1 回の遊技に使用する枚数を決定するために設けられる。なお、MAX ベットボタン 1 5 a の内部には、メダル投入が可能に点灯するベットボタン LED（不図示）が設けられている。また、精算ボタンは、パチスロ 1 の内部に預けられているメダルを外部に引き出す（排出する）ために設けられる。

【 0 0 4 0 】

なお、遊技者が MAX ベットボタン 1 5 a を押下操作すると、単位遊技のベット枚数（3 枚）のメダルが投入され、有効ラインが有効化される。一方、1 ベットボタン 1 5 b が 1 回、押下操作される度に 1 枚のメダルが投入される。1 ベットボタン 1 5 b が 3 回操作されると、単位遊技のベット枚数（3 枚）のメダルが投入され、有効ラインが有効化される。

20

【 0 0 4 1 】

なお、以下では、MAX ベットボタン 1 5 a の操作、1 ベットボタン 1 5 b の操作及びメダル投入口 1 4 にメダルを投入する操作（遊技を行うためにメダルを投入する操作）をいづれも「投入操作」という。

【 0 0 4 2 】

スタートレバー 1 6 は、全てのリール（3 L, 3 C, 3 R）の回転を開始するために設けられる。ストップボタン 1 7 L, 1 7 C, 1 7 R は、それぞれ、左リール 3 L、中リール 3 C、右リール 3 R に対応づけて設けられ、各ストップボタンは対応するリールの回転を停止するために設けられる。以下、ストップボタン 1 7 L, 1 7 C, 1 7 R を、それぞれ左ストップボタン 1 7 L、中ストップボタン 1 7 C、右ストップボタン 1 7 R ともいう。

30

【 0 0 4 3 】

また、リール表示窓 4 の下方の略水平面の台座領域の略中央には、情報表示器 6 が設けられる。なお、情報表示器 6 は、透明の窓カバー（不図示）によって覆われている。

【 0 0 4 4 】

情報表示器 6 には、特典として遊技者に対して払い出されるメダルの枚数（以下、「払出枚数」という）の情報を遊技者に対してデジタル表示（報知）するための 2 桁の 7 セグメント LED（以下、「7 セグ LED」という）や、パチスロ 1 の内部に預けられているメダルの枚数（以下、「クレジット枚数」という）などの情報を遊技者に対してデジタル表示（報知）するための 2 桁の 7 セグメント LED が設けられる。なお、本実施形態では、メダルの払出枚数表示用の 2 桁の 7 セグメント LED は、エラー発生及びエラー種別の情報を遊技者に対してデジタル表示（報知）するための 2 桁の 7 セグメント LED としても用いられる。それゆえ、エラー発生時には、メダルの払出枚数表示用の 2 桁の 7 セグメント LED の表示態様は、払出枚数の表示態様からエラー種別の情報の表示態様に切り替わる。

40

【 0 0 4 5 】

さらに、情報表示器 6 には、内部当籤役として決定された役に応じた図柄組合せを有効ラインに沿って表示するために必要な停止操作の情報を報知する指示モニタ（不図示）が設けられている。指示モニタ（指示表示器）は、例えば、2 桁の 7 セグメント LED により構成される。そして、指示モニタでは、報知する停止操作の情報と一義的に対応する態様で、2 桁の 7 セグメント LED が点灯、点滅又は消灯することにより、遊技者に対して必要な停

50

止操作の情報を報知する。

【 0 0 4 6 】

なお、ここでいう、報知する停止操作の情報と一義的に対応する態様とは、例えば、押し順「1st（第1停止操作を左リール3Lに対して行うこと）」を報知する場合には指示モニタに数値「1」を表示し、押し順「2nd（第1停止操作を中リール3Cに対して行うこと）」を報知する場合には指示モニタに数値「2」を表示し、押し順「3rd（第1停止操作を右リール3Rに対して行うこと）」を報知する場合には指示モニタに数値「3」を表示するなどの態様のことである。なお、指示モニタにおける停止操作の情報の報知態様（後述のメイン側で決定されるナビデータ）については、後述の図63を参照しながら後で詳述する。

10

【 0 0 4 7 】

情報表示器6は、後述の図7に示すように、ドア中継基板68及び遊技作動表示基板81を介して主制御基板71に電氣的に接続され、情報表示器6の表示動作は、主制御基板71内の後述の主制御回路90により制御される。また、上述した各種7セグLEDの制御方式は、ダイナミック点灯制御である。

【 0 0 4 8 】

なお、本実施形態のパチスロ1では、主制御基板71により制御される指示モニタに加えて、副制御基板72により制御される他の手段を用いて停止操作の情報を報知する構成を設ける。具体的には、後述のプロジェクタ機構211及び表示ユニット212（図3及び後述の図7参照）により構成される後述の表示装置11により停止操作の情報を報知する。

20

【 0 0 4 9 】

このような構成を適用した場合、指示モニタにおける報知の態様と、副制御基板72により制御されるその他の手段における報知の態様とは、互いに異なる態様であってもよい。すなわち、指示モニタでは、報知する停止操作の情報と一義的に対応する態様で報知すればよく、必ずしも、停止操作の情報を直接的に報知する必要はない（例えば、指示モニタにおいて数値「1」が表示されたとしても、遊技者によっては報知内容を特定できない可能性もあり、直接的な報知とは言えない）。一方、後述の表示装置11等のその他の手段によるサブ側（副制御基板側）での報知では、停止操作の情報を直接的に報知してもよい。例えば、押し順「1st」を報知する場合、指示モニタでは報知する押し順と一義的に対応する数値「1」を表示するが、その他の手段（例えば、表示装置11等）では、左リール3Lに対して第1停止操作を行わせるための指示情報を直接的に報知してもよい。

30

【 0 0 5 0 】

このような構成のパチスロ1では、副制御基板72の制御だけでなく、主制御基板71の制御によっても、内部当籤役に応じた必要な停止操作の情報を報知することができる。また、このような停止操作の情報の報知の有無は、遊技状態に応じて制御されるようにしてもよい。例えば、後述の一般遊技状態（非ART遊技状態）では停止操作の情報を報知せずに、後述のART遊技状態（後述の図14参照）において停止操作の情報を報知するようにしてもよい。

【 0 0 5 1 】

また、遊技者側から見て、リール表示窓4の左方には、サブ表示装置18が設けられる。サブ表示装置18は、図2に示すように、ドア本体9の前面部のうち、台座部13の略水平面の台座領域から略垂直に立設するように設けられる。サブ表示装置18は、液晶ディスプレイや有機EL（Electro-Luminescence）ディスプレイで構成され、各種情報を表示する。

40

【 0 0 5 2 】

また、サブ表示装置18の表示面上には、タッチセンサ19が設けられている（後述の図7参照）。タッチセンサ19は、静電容量方式などの所定の動作原理に従い動作し、遊技者の操作を受け付けると、タッチ入力情報として当該操作に応じた信号を出力する。そして、本実施形態のパチスロ1は、タッチセンサ19を介して受け付けた遊技者の操作（タッチセンサ19から出力されるタッチ入力情報）に応じて、サブ表示装置18の表示を切

50

り替え可能にする機能を有する。なお、サブ表示装置 18 は、タッチセンサ 19 から出力されるタッチ入力情報に基づいて後述の副制御基板 72（後述の図 7 参照）により制御される。

【0053】

ドア本体 9 の下部には、メダル払出口 24、メダル受皿 25、2 つのスピーカ用孔 20L、20R 等が設けられる。メダル払出口 24 は、後述のメダル払出装置 51 の駆動により排出されるメダルを外部に導く。メダル受皿 25 は、メダル払出口 24 から排出されたメダルを貯める。また、2 つのスピーカ用孔 20L、20R からは、演出内容に対応する効果音や楽曲等の音声が出力される。

【0054】

[内部構造]

次に、パチスロ 1 の内部構造を、図 3 及び図 4 を参照しながら説明する。図 3 は、キャビネット 2a の内部構造を示す図であり、図 4 は、フロントドア 2b の裏面側の内部構造を示す図である。

【0055】

キャビネット 2a は、図 3 に示すように、上面板 27a と、底面板 27b と、左右の側面板 27c、27d と、背面板 27e とを有する。そして、キャビネット 2a 内の上部には、表示装置 11 が配設される。

【0056】

表示装置 11 は、プロジェクタ機構 211 と、プロジェクタ機構 211 から投射された映像光が投影される箱状の被投影部材 212a とを有し、プロジェクションマッピングによる映像表示を行う。具体的には、表示装置 11 では、立体物となる被投影部材 212a の位置（投影距離や角度など）や形状に基づいて映像光を生成し、その映像光が、プロジェクタ機構 211 により被投影部材 212a の表面に投影される。このような演出機能を設けることにより、高度で且つ迫力のある演出を行うことができる。また、図 3 には示さないが、箱状の被投影部材 212a の裏側には、表示面が湾曲した別の被投影部材が設けられ、遊技状態に応じて、どちらか一方の被投影部材が、映像光が投影されるスクリーンとして使用される。それゆえ、キャビネット 2a 内は、遊技状態に応じて、被投影部材を切り換える機能（不図示）も設けられる。

【0057】

キャビネット 2a 内の下部には、メダル払出装置（以下、ホッパー装置という）51 と、メダル補助収納庫 52 と、電源装置 53 とが配設される。

【0058】

ホッパー装置 51 は、キャビネット 2a における底面板 27b の中央部に取り付けられる。このホッパー装置 51 は、多量のメダルを収容可能で、それらを 1 枚ずつ排出可能な構造を有する。ホッパー装置 51 は、貯留されたメダルが例えば 50 枚を超えたとき、又は、精算ボタンが押下されてメダルの精算が実行されるときに、メダルを払い出す。そして、ホッパー装置 51 によって払い出されたメダルは、メダル払出口 24（図 2 参照）から排出される。

【0059】

メダル補助収納庫 52 は、ホッパー装置 51 から溢れ出たメダルを収納する。このメダル補助収納庫 52 は、キャビネット 2a 内部を正面から見て、ホッパー装置 51 の右側に配置される。また、メダル補助収納庫 52 は、キャビネット 2a の底面板 27b に対して着脱可能に取り付けられている。

【0060】

電源装置 53 は、電源スイッチ 53a と、電源基板 53b（電源供給手段）とを有している（後述の図 7 参照）。この電源装置 53 は、キャビネット 2a 内部を正面から見て、ホッパー装置 51 の左側に配置されており、左側面板 27c に取り付けられている。電源装置 53 は、サブ電源装置（不図示）から供給された交流電圧 100V の電力を各部で必要な直流電圧の電力に変換して、変換した電力を各部へ供給する。

10

20

30

40

50

【 0 0 6 1 】

また、キャビネット 2 a 内の電源装置 5 3 の上方には、副制御基板 7 2（後述の図 7 参照）を収容する副制御基板ケース 5 7 が配設される。副制御基板ケース 5 7 に収納された副制御基板 7 2 には、後述の副制御回路 2 0 0（後述の図 1 0 参照）が搭載されている。この副制御回路 2 0 0 は、映像の表示等による演出の実行を制御する回路である。副制御回路 2 0 0 の具体的な構成については後述する。

【 0 0 6 2 】

キャビネット 2 a 内の副制御基板ケース 5 7 の上方には、副中継基板 6 1 が配設される。この副中継基板 6 1 は、副制御基板 7 2 と後述の主制御基板 7 1 とを接続する配線が実装された中継基板である。また、副中継基板 6 1 は、副制御基板 7 2 と副制御基板 7 2 の周辺に配設された基板や各種装置部（ユニット）などを接続する配線が実装された中継基板である。

10

【 0 0 6 3 】

また、図 3 には示さないが、キャビネット 2 a 内には、キャビネット側中継基板 4 4（後述の図 7 参照）が配設される。このキャビネット側中継基板 4 4 は、主制御基板 7 1（後述の図 7 参照）と、ホッパー装置 5 1、遊技メダル補助収納庫スイッチ 7 7（後述の図 7 参照）及びメダル払出カウントスイッチ（不図示）のそれぞれとを接続する配線が実装された中継基板である。

【 0 0 6 4 】

フロントドア 2 b の裏面側の中央部には、図 4 に示すように、ミドルドア 4 1 が、配設され、リール表示窓 4（図 2 参照）を裏側から開閉可能に取り付けられている。また、図 4 には示さないが、ミドルドア 4 1 のリール表示窓 4 側には、3 つのリール 3 L, 3 C, 3 R が取り付けられ、ミドルドア 4 1 のリール表示窓 4 側とは反対側には、主制御基板 7 1（後述の図 7 参照）が収納された主制御基板ケース 5 5 が取り付けられている。なお、3 つのリール 3 L, 3 C, 3 R には、所定の減速比をもったギアを介してステッピングモータ（不図示）が接続されている。

20

【 0 0 6 5 】

主制御基板ケース 5 5 に収納された主制御基板 7 1 は、後述する主制御回路 9 0（後述の図 9 参照）を有する。主制御回路 9 0（主制御手段）は、内部当籤役の決定、各リール 3 L, 3 C, 3 R の回転及び停止、入賞の有無の判定といった、パチスロ 1 における遊技の主な流れを制御する回路である。また、本実施形態では、例えば、ART の決定の有無の抽籤処理、ナビ情報の指示モニタへの表示処理、各種試験信号の送信処理などの制御も主制御回路 9 0 により行われる。なお、主制御回路 9 0 の具体的な構成は後述する。

30

【 0 0 6 6 】

フロントドア 2 b の裏面側において、ミドルドア 4 1 の下方には、スピーカ 6 5 L, 6 5 R が配設される。スピーカ 6 5 L, 6 5 R は、それぞれスピーカ用孔 2 0 L, 2 0 R（図 2 参照）と対向する位置に配置されている。

【 0 0 6 7 】

また、スピーカ 6 5 L の上方には、セレクトタ 6 6 と、ドア開閉監視スイッチ 6 7 とが配設される。セレクトタ 6 6 は、メダルの材質や形状等が適正であるか否かを選別する装置であり、メダル投入口 1 4 に投入された適正なメダルをホッパー装置 5 1 へ案内する。セレクトタ 6 6 内においてメダルが通過する経路上には、適正なメダルが通過したことを検出するメダルセンサ（遊技媒体検出手段：不図示）が設けられている。

40

【 0 0 6 8 】

ドア開閉監視スイッチ 6 7 は、フロントドア 2 b を裏面側から見て、セレクトタ 6 6 の左斜め下に配置される。このドア開閉監視スイッチ 6 7 は、フロントドア 2 b の開閉を報知するためのセキュリティ信号をパチスロ 1 の外部に出力する。

【 0 0 6 9 】

また、図 4 には示さないが、フロントドア 2 b を裏面において、ミドルドア 4 1 により開閉された領域であり且つリール表示窓 4 の下方には、ドア中継端子板 6 8 が配設される（

50

後述の図7参照)。このドア中継端子板68は、主制御基板ケース55内の主制御基板71と、各種のボタンやスイッチ、副中継基板61、セクタ66、遊技動作表示基板81、試験機用第1インターフェースボード301及び試験機用第2インターフェースボード302のそれぞれとを接続する配線が実装された中継基板である。なお、各種のボタン及びスイッチとしては、例えば、MAXベットボタン15a、1ベットボタン15b、ドア開閉監視スイッチ67、後述のBETスイッチ77、スタートスイッチ79等が挙げられる。

【0070】

<サブ表示装置の表示例>

ここで、図5A～図5Eを参照して、サブ表示装置18に表示される各種表示画面について説明する。なお、図5Aは、サブ表示装置18に表示されるトップ画面221を示す図であり、図5Bは、サブ表示装置18に表示されるメニュー画面222を示す図である。また、図5C～図5Eは、サブ表示装置18に表示される遊技情報画面223, 224, 225を示す図である。

10

【0071】

サブ表示装置18には、遊技者のタッチ操作により様々な表示画面が表示され、図5A～図5Eに示すように、トップ画面221、メニュー画面222及び遊技情報画面223, 224, 225を含む各種表示画面が表示される。これらの表示画面は、タッチセンサ19を介して受け付けた遊技者の操作信号に基づいて切り替えられる。

【0072】

トップ画面221は、サブ表示装置18に表示される表示画面のうちの初期画面であり、トップ画面221では、「MENU」ボタン221aと、概要遊技履歴221bとが表示される。「MENU」ボタン221aは、図5Bに示すメニュー画面222を呼び出すための操作ボタンであり、「MENU」ボタン221aに対して遊技者による所定操作(例えばタップ)が行われると、メニュー画面222が呼び出される。また、トップ画面221では、概要遊技履歴221bとして、パチスロ1の一部の遊技履歴(概要遊技履歴)を表示する。本実施形態では、概要遊技履歴221bとして、例えば、ボーナス回数、ART回数及びゲーム数(遊技回数)が表示される。

20

【0073】

メニュー画面222は、サブ表示装置18で表示可能なメニューを表示する画面であり、メニュー画面222では、「戻る」ボタン222a、「登録」ボタン222b、「説明」ボタン222c、「配列配当」ボタン222d、「リーチ目」ボタン222e、「WEBサイト」ボタン222f及び「音量」ボタン222gが表示される。「戻る」ボタン222aは、トップ画面221を呼び出すための操作ボタンであり、「戻る」ボタン222aに対して遊技者による操作が行われると、トップ画面221が呼び出される。また、「登録」ボタン222b～「音量」ボタン222gは、対応するメニュー内容の表示画面を呼び出すための操作ボタンであり、各ボタンに対して遊技者による操作が行われると、対応するメニュー内容の表示画面が呼び出される。

30

【0074】

例えば、メニュー画面222において「登録」ボタン222bが遊技者により操作された場合、遊技中の遊技者を登録するための登録画面(不図示)がサブ表示装置18の表示画面に呼び出される。近年のパチスロでは、機種ごとに遊技者を登録しておき、当該遊技者のこれまでの遊技履歴から、定められたミッションの達成状況などの様々な情報を管理するサービスが広く行われている。「登録」ボタン222bにより呼び出される登録画面は、このサービスの提供を受ける際に遊技者を登録するための表示画面である。

40

【0075】

また、例えば、メニュー画面222において「説明」ボタン222cが遊技者により操作された場合、パチスロ1の説明画面(不図示)がサブ表示装置18の表示画面に呼び出される。説明画面で表示される情報には、例えば、設定値ごとのボーナス当籤確率やART当籤確率などのパチスロ1の仕様に関する説明や、パチスロ1の演出に登場するキャラク

50

タの紹介説明などが含まれる。

【 0 0 7 6 】

また、例えば、メニュー画面 2 2 2 において「配列配当」ボタン 2 2 2 d が遊技者により操作された場合、パチスロ 1 の配列配当画面（不図示）がサブ表示装置 1 8 の表示画面に呼び出される。配列配当画面には、例えば、パチスロ 1 において入賞と判定される図柄の組合せと、入賞と判定された際の特典との対応関係（配当表）や、各リール 3 L , 3 C , 3 R に描かれた図柄列（リール配列）などが表示される。

【 0 0 7 7 】

また、例えば、メニュー画面 2 2 2 において「リーチ目」ボタン 2 2 2 e が遊技者により操作された場合、パチスロ 1 のリーチ目画面（不図示）がサブ表示装置 1 8 の表示画面に呼び出される。リーチ目画面には、パチスロ 1 で設定されている「リーチ目」と称される図柄組合せがの情報が表示される。なお、「リーチ目」と称する図柄組合せは、該図柄組合せが有効ラインに沿って表示されることにより、特別な特典が付与される図柄組合せであり、本実施形態のパチスロ 1 では、後述の図 2 8 ~ 図 3 0 の入賞作動フラグ格納領域の内容欄に示す略称「リーチ目リブ」に対応する図柄組合せが該当する。そして、本実施形態では、「リーチ目リブ」に係る図柄組合せが有効ラインに沿って表示された場合、その後、遊技者にとって有利な状態（例えば、ボーナス状態、通常 A R T 又は C T（後述の図 1 4 参照））に移行することが確定する。

10

【 0 0 7 8 】

また、例えば、メニュー画面 2 2 2 において「WEB サイト」ボタン 2 2 2 f が遊技者により操作された場合、パチスロ 1 のWEB 紹介画面（不図示）がサブ表示装置 1 8 の表示画面に呼び出される。WEB 紹介画面には、例えば、パチスロ 1 の機種ごとに設けられた特設WEB サイトやパチスロ 1 のメーカーのWEB サイトなど任意のWEB サイトのURL を示す二次元コード（例えば、QR コード（登録商標））が表示される。遊技者は、携帯電話などでWEB 紹介画面に表示される二次元コードを読み込むことにより、対応するWEB サイトにアクセスすることができる。

20

【 0 0 7 9 】

また、例えば、メニュー画面 2 2 2 において「音量」ボタン 2 2 2 g が遊技者により操作された場合、スピーカ 6 5 L , 6 5 R から出力する音の音量を調整することが可能な音量調整画面（不図示）がサブ表示装置 1 8 の表示画面に呼び出される。遊技者は、音量調整画面を介してパチスロ 1 の演出音の音量を調整することができる。

30

【 0 0 8 0 】

なお、サブ表示装置 1 8 は、上述した表示装置 1 1（プロジェクタ機構 2 1 1 及び表示ユニット 2 1 2）とは別体に設けられるため、表示装置 1 1 とは別個に制御することができる。それゆえ、本実施形態のパチスロ 1 では、遊技中（表示装置 1 1 による演出の実行中）であっても、サブ表示装置 1 8 の表示画面を遊技者の操作により切り替えることができる。その結果、例えば、遊技者が、表示装置 1 1 の演出において登場するキャラクターのことを知りたいと思った場合、遊技者は、「説明」ボタン 2 2 2 c を操作して説明画面を呼び出すことにより、キャラクター間の関係性などの情報を遊技中に把握することができる。また、例えば、遊技者が、遊技中に、いわゆる「レア役」が当籤した場合のリール回転中にレア役を入賞させるために目安とすべき図柄を把握したいと思った場合、遊技者は、「配列配当」ボタン 2 2 2 d を操作して配列配当画面を呼び出すことにより、リール配列を把握することができる。

40

【 0 0 8 1 】

遊技情報画面 2 2 3 , 2 2 4 , 2 2 5 は、パチスロ 1 の遊技履歴のうちのトップ画面 2 2 1 に表示する概要遊技履歴を含む詳細遊技履歴情報を表示する表示画面である。

【 0 0 8 2 】

遊技情報画面 2 2 3 には、「戻る」ボタン 2 2 3 a と、「MENU」ボタン 2 2 3 b と、「前へ」ボタン 2 2 3 c と、「次へ」ボタン 2 2 3 d と、遊技履歴 2 2 3 e とが表示される。「戻る」ボタン 2 2 3 a 及び「MENU」ボタン 2 2 3 b は、それぞれトップ画面 2

50

2 1 及びメニュー画面 2 2 2 をサブ表示装置 1 8 の表示画面に呼び出すための操作ボタンであり、各ボタンを遊技者が操作することにより、対応する表示画面が呼び出される。また、「前へ」ボタン 2 2 3 c 及び「次へ」ボタン 2 2 3 d は、遊技情報画面を所定の順序で切り替えるための操作ボタンであり、「前へ」ボタン 2 2 3 c が遊技者により操作されると、表示画面が遊技情報画面 2 2 3 から遊技情報画面 2 2 5 に切り替わり、「次へ」ボタン 2 2 3 d が遊技者により操作されると、表示画面が遊技情報画面 2 2 3 から遊技情報画面 2 2 4 に切り替わる。また、遊技履歴 2 2 3 e としては、図 5 C に示すように、ゲーム数（遊技回数）、ボーナス回数、ART 回数及び CZ（チャンスゾーン）回数が表示される。

【 0 0 8 3 】

遊技情報画面 2 2 4 には、「戻る」ボタン 2 2 4 a と、「MENU」ボタン 2 2 4 b と、「前へ」ボタン 2 2 4 c と、「次へ」ボタン 2 2 4 d と、遊技履歴 2 1 4 e とが表示される。「戻る」ボタン 2 2 4 a 及び「MENU」ボタン 2 2 4 b は、それぞれトップ画面 2 2 1 及びメニュー画面 2 2 2 をサブ表示装置 1 8 の表示画面に呼び出すための操作ボタンであり、各ボタンを遊技者が操作することにより、対応する表示画面が呼び出される。また、「前へ」ボタン 2 2 4 c 及び「次へ」ボタン 2 2 4 d は、遊技情報画面を所定の順序で切り替える操作ボタンであり、「前へ」ボタン 2 2 4 c が遊技者に操作されると、表示画面が遊技情報画面 2 2 4 から遊技情報画面 2 2 3 に切り替わり、「次へ」ボタン 2 2 4 d が遊技者により操作されると、表示画面が遊技情報画面 2 2 4 から遊技情報画面 2 2 5 に切り替わる。また、遊技履歴 2 2 4 e としては、後述の CZ（チャンスゾーン）の突入回数及び成功回数が表示される。なお、後述するように、本実施形態では、CZ として、CZ 1, CZ 2, CZ 3 の 3 種類の CZ が設けられる。それゆえ、遊技履歴 2 2 4 e としては、図 5 D に示すように、CZ 1 ~ CZ 3 のそれぞれの突入回数及び成功回数が表示される。

【 0 0 8 4 】

遊技情報画面 2 2 5 には、「戻る」ボタン 2 2 5 a と、「MENU」ボタン 2 2 5 b と、「前へ」ボタン 2 2 5 c と、「次へ」ボタン 2 2 5 d と、遊技履歴 2 2 5 e とが表示される。「戻る」ボタン 2 2 5 a 及び「MENU」ボタン 2 2 5 b は、それぞれトップ画面 2 2 1 及びメニュー画面 2 2 2 をサブ表示装置 1 8 の表示画面に呼び出すための操作ボタンであり、各ボタンを遊技者が操作することにより、対応する表示画面が呼び出される。また、「前へ」ボタン 2 2 5 c 及び「次へ」ボタン 2 2 5 d は、所定の順序で遊技情報画面を切り替えるための操作ボタンであり、「前へ」ボタン 2 2 5 c が遊技者に操作されると、表示画面が遊技情報画面 2 2 5 から遊技情報画面 2 2 4 に切り替わり、「次へ」ボタン 2 2 5 d が遊技者により操作されると、表示画面が遊技情報画面 2 2 5 から遊技情報画面 2 2 3 に切り替わる。また、遊技履歴 2 2 5 e としては、図 5 E に示すように、小役の当籤回数及び当籤確率（分子が 1 の分数）が表示される。

【 0 0 8 5 】

なお、サブ表示装置 1 8 に表示される表示画面の切り替え手法としては、例えば、それぞれの表示画面に表示される操作ボタンに対するタップ操作に基づいて切り替える手法を採用してもよいし、また、例えば、表示画面に対するスワイプ操作に基づいて切り替える手法を採用してもよい。

【 0 0 8 6 】

< サブ表示装置の表示画面の各種切り替え機能 >

次に、本実施形態のパチスロ 1 におけるサブ表示装置 1 8 の表示画面の各種切り替え機能について説明する。

【 0 0 8 7 】

[サブ表示装置の表示画面の遷移例]

まず、図 6 A 及び 6 B を参照して、本実施形態のパチスロ 1 におけるサブ表示装置 1 8 の表示画面の遷移例（切り替え態様）について説明する。なお、図 6 A は、遊技者登録状態がセットされていない状況におけるサブ表示装置 1 8 の表示画面の遷移例を示す図であり

10

20

30

40

50

、図6Bは、遊技者登録状態がセットされている状況におけるサブ表示装置18の表示画面の遷移例を示す図である。

【0088】

遊技者登録状態がセットされていない状況では、図6Aに示すように、サブ表示装置18の表示画面は、トップ画面221とメニュー画面222との間、並びに、メニュー画面222とメニュー画面222から遷移可能な各種表示画面との間でのみ遷移可能であり、これらの表示画面間の遷移は副制御基板72（後述のサブCPU201）により制御される。例えば、副制御基板72は、タッチセンサ19を介して取得したタッチ操作（例えば、所定のボタンに対するタップ操作や、表示画面上におけるスワイプ操作）に基づいて、トップ画面221及びメニュー画面222間で、表示画面を切り替える。しかしながら、遊技者登録状態がセットされていない状況では、副制御基板72は、遊技情報画面223，224，225の表示が不可能となるように制御する。すなわち、遊技者登録状態がセットされていない状況では、遊技者は、サブ表示装置18に遊技情報画面223，224，225を表示することができない。

10

【0089】

一方、遊技者登録状態がセットされている状況では、図6Bに示すように、サブ表示装置18の表示画面は、トップ画面221とメニュー画面222との間、並びに、メニュー画面222とメニュー画面222から遷移可能な各種表示画面との間に加え、メニュー画面222と遊技情報画面223，224，225との間においても遷移可能となり、これらの表示画面間の遷移は副制御基板72（後述のサブCPU201）により制御される。すなわち、副制御基板72は、タッチセンサ19を介して取得したタッチ操作に基づいて、トップ画面221とメニュー画面222との間だけでなく、トップ画面221及びメニュー画面222のそれぞれと、遊技情報画面223，224，225との間においても表示画面を切り替えることができる。それゆえ、本実施形態において、遊技者登録状態がセットされている場合、遊技者は、サブ表示装置18に遊技情報画面223，224，225を表示することができる。

20

【0090】

なお、図6Bに示すように、トップ画面221、メニュー画面222及び遊技情報画面223，224，225間における表示画面の遷移順序は任意である。それゆえ、例えば、トップ画面221から遊技情報画面223，224，225に直接遷移可能となる構成にしてもよいし、トップ画面221からメニュー画面222を介してのみ遊技情報画面223，224，225に遷移可能となる構成にしてもよい。

30

【0091】

本実施形態のパチスロ1では、トップ画面221からメニュー画面222を介してのみ遊技情報画面223，224，225に遷移可能な構成（トップ画面221から遊技情報画面223，224，225に直接遷移できない構成）を採用している。なお、本実施形態のパチスロ1では、メニュー画面222において、遊技者が表示画面に対してスワイプ操作（メニュー選択操作ではない）を行うことにより、表示画面をメニュー画面222から遊技情報画面223，224，225に遷移させることができる。

【0092】

なお、本実施形態では、遊技情報画面223，224，225は、メニュー画面222とは完全に独立して設けられた表示画面である。すなわち、本実施形態のパチスロ1では、遊技履歴という、遊技者が遊技中に強い関心を抱く遊技の結果を示す情報を、配当配列や音量調節などの遊技の結果とは関係のない情報として独立して表示する。そして、本実施形態では、遊技者登録状態がセットされている状況において、メニュー画面222に対して遊技者がメニュー選択操作を行うことなく、遊技情報画面223，224，225を表示可能にしている。それゆえ、本実施形態では、遊技者登録状態がセットされている場合、遊技者が所望する情遊技履歴情報へのアクセスを容易に行うことができる。

40

【0093】

また、本実施形態では、メニュー画面222に対するメニュー選択操作では、表示画面を

50

遊技情報画面 2 2 3 , 2 2 4 , 2 2 5 に遷移させることができず、メニュー画面 2 2 2 に対してスワイプ操作（メニュー表示では指定されていない操作）を行わなければ、表示画面を遊技情報画面 2 2 3 , 2 2 4 , 2 2 5 に遷移させることができない。それゆえ、本実施形態のパチスロ 1 では、表示画面を遊技情報画面 2 2 3 , 2 2 4 , 2 2 5 に遷移させるためのスワイプ操作を、遊技者登録状態がセットされている状況における隠しコマンドとして扱うことができる。この場合、遊技者にとってみれば、パチスロ 1 に対する自身の知識により、知識の少ない他の遊技者では見ることで見ることができない、より詳細な遊技履歴情報を見ることができるため、当該他の遊技者よりも有利に遊技を行うことができ、結果、遊技者が積極的に遊技を行うことを期待することができる。

【 0 0 9 4 】

[遊技情報画面からトップ画面への表示切り替え機能]

本実施形態のパチスロ 1 は、サブ表示装置 1 8 の表示画面を、遊技情報画面 2 2 3 , 2 2 4 , 2 2 5 から、遊技者の手動により、又は、自動的に、トップ画面 2 2 1 に遷移させる機能を有する。具体的には、本実施形態では、遊技情報画面 2 2 3 , 2 2 4 , 2 2 5 において「戻る」ボタンが操作されると、表示画面が遊技情報画面 2 2 3 , 2 2 4 , 2 2 5 からトップ画面 2 2 1 に遷移する（手動遷移機能）。また、本実施形態では、遊技情報画面 2 2 3 , 2 2 4 , 2 2 5 が表示されている状態において所定の条件を満たした場合には、遊技者の操作とは関係なく自動的に表示画面がトップ画面 2 2 1 に遷移する（自動遷移機能）。

【 0 0 9 5 】

より具体的には、パチスロ 1 では、遊技情報画面 2 2 3 , 2 2 4 , 2 2 5 が表示されている状態において、投入操作（MAXベットボタン 1 5 a への操作、1ベットボタン 1 5 b への操作及びメダル投入口 1 4 にメダルを投入する操作）が行われると、サブ表示装置 1 8 の表示画面が自動的にトップ画面 2 2 1 に遷移する。なお、ART遊技状態のように、リプレイ役が内部当籤役として決定される確率が高い遊技状態（高リプ状態）では、リプレイ役入賞に伴う再遊技の作動によりメダルが自動的に投入されてしまう結果、高リプ状態では、遊技情報画面 2 2 3 , 2 2 4 , 2 2 5 を表示する機会が制限されてしまう可能性がある。そこで、本実施形態のパチスロ 1 では、再遊技の作動によりメダルが自動的に投入された場合には、メダルの投入操作ではなく、開始操作を契機として、自動的に表示画面が遊技情報画面 2 2 3 , 2 2 4 , 2 2 5 からトップ画面 2 2 1 に遷移する。

【 0 0 9 6 】

すなわち、本実施形態では、再遊技が作動し、かつ、遊技情報画面 2 2 3 , 2 2 4 , 2 2 5 が表示されている場合には、開始操作を契機として、自動的に表示画面が遊技情報画面 2 2 3 , 2 2 4 , 2 2 5 からトップ画面 2 2 1 に遷移する。一方、再遊技の作動が行われていない場合には、投入操作を契機として、自動的に表示画面が遊技情報画面 2 2 3 , 2 2 4 , 2 2 5 からトップ画面 2 2 1 に遷移する。

【 0 0 9 7 】

[メニュー内容表示画面からトップ画面（又はメニュー画面）への表示切り替え機能]

本実施形態のパチスロ 1 は、サブ表示装置 1 8 の表示画面を、メニュー画面 2 2 2 に対するメニュー選択操作により遷移可能な各種メニュー内容表示画面（登録画面、説明画面、配列配当画面、リーチ目画面、WEB紹介画面及び音量調整画面）から、遊技者の手動により、又は、自動的に、トップ画面 2 2 1（又はメニュー画面 2 2 2）に遷移させる機能を有する。具体的には、本実施形態では、メニュー内容表示画面において所定のボタン（例えば、「TOPへ戻る」ボタン）が操作されると、表示画面が当該メニュー内容表示画面からトップ画面 2 2 1 に遷移する（手動遷移機能）。また、本実施形態では、メニュー内容表示画面において特定のボタン（例えば、「戻る」ボタン）が操作されると、表示画面が当該メニュー内容表示画面からメニュー画面 2 2 2 に表示画面を遷移する（手動遷移機能）。

【 0 0 9 8 】

さらに、本実施形態では、メニュー内容表示画面が表示されている状態において所定の時

10

20

30

40

50

間が経過すると、遊技者の操作とは関係なく、自動的に表示画面がトップ画面 2 2 1（又はメニュー画面 2 2 2）に遷移する（自動遷移機能）。なお、この際、トップ画面 2 2 1（又はメニュー画面 2 2 2）に自動遷移する契機となる所定の時間は、現在表示しているメニュー内容表示画面の種類に応じて異なる。例えば、パチスロ 1 から出力する音量の調整を行う音量調整画面を長時間表示していると、音量が誤操作により意図しない音量に調整されてしまうおそれがあるだけでなく、誤操作により他の遊技者を不快にしまうおそれもある。それゆえ、音量調整画面では、他のメニュー内容表示画面よりも短い時間で、自動的にトップ画面 2 2 1（又はメニュー画面 2 2 2）に遷移するように設定されている。一方、登録画面は、遊技者の登録を行い易くするために、他のメニュー内容表示画面よりも長い時間で、自動的にトップ画面 2 2 1（又はメニュー画面 2 2 2）に遷移するよ

10

【 0 0 9 9 】

すなわち、各メニュー内容表示画面には、トップ画面 2 2 1（又はメニュー画面 2 2 2）に自動遷移する契機となる経過時間（自動遷移時間）が、当該メニュー内容表示画面の種類に応じて適宜設定されており、音量調整画面には、他のメニュー内容表示画面よりも短い自動遷移時間が設定され、登録画面には、他のメニュー内容表示画面よりも長い自動遷移時間が設定されている。

【 0 1 0 0 】

[メニューの操作可否の選択機能]

本実施形態のパチスロ 1 では、サブ表示装置 1 8 の表示画面を、メニュー画面 2 2 2 から、登録画面、説明画面、配列配当画面、リーチ目画面、WEB 紹介画面及び音量調整画面に遷移させることにより、遊技者が、これらのメニュー内容表示画面に応じた各種操作を行うことができ、また、各種情報を確認することができる。なお、このような遊技者がメニュー選択できる機能を遊技店側の設定に応じて制限できるような機能（メニューの操作可否の選択機能）を設けてもよい。

20

【 0 1 0 1 】

例えば、遊技店側の設定により、表示画面をメニュー画面 2 2 2 から音量調節画面に遷移不可能にする（例えば、メニュー画面 2 2 2 に「音量」ボタン 2 2 2 g を表示しない）ようにしてもよい。この場合、遊技者による音量調節を不可能にすることができる。

【 0 1 0 2 】

< パチスロが備える制御系 >

次に、パチスロ 1 が備える制御系について、図 7 を参照して説明する。図 7 は、パチスロ 1 の制御系の構成を示す回路ブロック図である。

30

【 0 1 0 3 】

パチスロ 1 は、ミドルドア 4 1 に設けられた主制御基板 7 1 と、フロントドア 2 b に設けられた副制御基板 7 2 とを有する。また、パチスロ 1 は、主制御基板 7 1 に接続された、リール中継端子板 7 4、設定用鍵型スイッチ 5 4（設定スイッチ）及びキャビネット側中継基板 4 4 を有する。さらに、パチスロ 1 は、キャビネット側中継基板 4 4 を介して主制御基板 7 1 に接続された外部集中端子板 4 7、ホッパー装置 5 1、メダル補助収納庫スイッチ 7 5、リセットスイッチ 7 6 及び電源装置 5 3 を有する。なお、ホッパー装置 5 1 の構成については上述したので、ここでは、その説明を省略する。

40

【 0 1 0 4 】

リール中継端子板 7 4 は、各リール 3 L, 3 C, 3 R のリール本体の内側に配設されている。リール中継端子板 7 4 は、各リール 3 L, 3 C, 3 R のステッピングモータ（不図示）に電氣的に接続されており、主制御基板 7 1 からステッピングモータに出力される信号を中継する。

【 0 1 0 5 】

設定用鍵型スイッチ 5 4 は、主制御基板ケース 5 5 に設けられる。設定用鍵型スイッチ 5 4 は、パチスロ 1 の設定（設定 1 ~ 設定 6）を変更するとき、もしくは、パチスロ 1 の設定を確認するとき使用される。

50

【 0 1 0 6 】

キャビネット側中継基板 4 4 は、主制御基板 7 1 と、外部集中端子板 4 7、ホッパー装置 5 1、メダル補助収納庫スイッチ 7 5、リセットスイッチ 7 6 及び電源装置 5 3 のそれぞれとを接続する配線が実装された中継基板である。外部集中端子板 4 7 は、メダル投入信号、メダル払出信号及びセキュリティ信号などの信号をパチスロ 1 の外部へ出力するために設けられる。メダル補助収納庫スイッチ 7 5 は、メダル補助収納庫 5 2 に設けられ、メダル補助収納庫 5 2 がメダルで満杯になっているか否かを検出する。リセットスイッチ 7 6 は、例えば、パチスロ 1 の設定を変更する際に用いられる。

【 0 1 0 7 】

電源装置 5 3 は、電源基板 5 3 b と、電源基板 5 3 b に接続された電源スイッチ 5 3 a とを有する。電源スイッチ 5 3 a は、パチスロ 1 に必要な電源を供給するときには押下される。電源基板 5 3 b は、キャビネット側中継基板 4 4 を介して主制御基板 7 1 に接続されるとともに、副中継基板 6 1 を介して副制御基板 7 2 にも接続される。

10

【 0 1 0 8 】

また、パチスロ 1 は、ドア中継端子板 6 8、並びに、該ドア中継端子板 6 8 を介して、主制御基板 7 1 に接続された、セレクト 6 6、ドア開閉監視スイッチ 6 7、B E T スイッチ 7 7、精算スイッチ 7 8、スタートスイッチ 7 9、ストップスイッチ基板 8 0、遊技動作表示基板 8 1、副中継基板 6 1、試験機用第 1 インターフェースボード 3 0 1 及び試験機用第 2 インターフェースボード 3 0 2 を有する。なお、セレクト 6 6、ドア開閉監視スイッチ 6 7 及び副中継基板 6 1 については、上述したので、ここでは、それらの説明を省略する。

20

【 0 1 0 9 】

B E T スイッチ 7 7 (投入操作検出手段) は、M A X ベットボタン 1 5 a 又は 1 ベットボタン 1 5 b が遊技者により押下されたことを検出する。精算スイッチ 7 8 は、精算ボタン (不図示) が遊技者により押下されたことを検出する。スタートスイッチ 7 9 (開始操作検出手段) は、スタートレバー 1 6 が遊技者により操作されたこと (開始操作) を検出する。

【 0 1 1 0 】

ストップスイッチ基板 8 0 (停止操作検出手段) は、回転しているメインリールを停止させるための回路と、停止可能なメインリールを L E D などにより表示するための回路とを備える。また、ストップスイッチ基板 8 0 には、ストップスイッチ (不図示) が設けられる。ストップスイッチは、各ストップボタン 1 7 L , 1 7 C , 1 7 R が遊技者により押下されたこと (停止操作) を検出する。

30

【 0 1 1 1 】

遊技動作表示基板 8 1 は、情報表示器 (7 セグ表示器) 6 及び L E D 8 2 に接続される。L E D 8 2 には、例えば、今回の遊技に投入されたメダルの枚数 (以下、「投入枚数」という) に対応して点灯する、メダル投入枚数表示用の 3 つの L E D (以下、「第 1 L E D」~「第 3 L E D」という) や、遊技動作表示基板 8 1 から入力される信号に基づいて、メダル投入が可能であることを表示するマーク、遊技開始を表示するマーク、再遊技を行うマークなどを点灯させる L E D などが含まれる。第 1 L E D ~ 第 3 L E D (表示手段) では、メダルが 1 枚投入されると、第 1 L E D が点灯し、メダルが 2 枚投入されると、第 1 及び第 2 L E D が点灯し、メダルが 3 枚 (遊技開始可能枚数) 投入されると、第 1 L E D ~ 第 3 L E D が点灯する。なお、情報表示器 6 については、上述したので、ここでは、それらの説明を省略する。

40

【 0 1 1 2 】

試験機用第 1 インターフェースボード 3 0 1 及び試験機用第 2 インターフェースボード 3 0 2 はともに、パチスロ 1 の検定試験 (試射試験) において、遊技に関する各種信号を試験機に出力する際に用いられる中継基板である (なお、販売用のリリース製品としてのパチスロ 1 にはこれらの中継基板は搭載されていないので、販売用の主制御基板 7 1 の主制御回路 9 0 には、試験機用第 1 インターフェースボード 3 0 1 及び試験機用第 2 インター

50

フェースボード 302 に接続するために必要な各種電子部品もまた実装されていない)。例えば、遊技に係る主要な動作(例えば、内部抽籤、リール停止制御等)を制御するための試験信号は、試験機用第1インターフェースボード 301 を介して出力され、例えば、主制御基板 71 で決定された押し順ナビに係る試験信号などは、試験機用第2インターフェースボード 302 を介して出力される。

【0113】

副制御基板 72 は、ドア中継端子板 68 及び副中継基板 61 を介して主制御基板 71 に接続される。また、パチスロ 1 は、副中継基板 61 を介して副制御基板 72 に接続された、スピーカ群 84、LED 群 85、24h ドア開閉監視ユニット 63、タッチセンサ 19 及び表示ユニット 212 を有する。なお、タッチセンサ 19 については、上述したので、ここでは、その説明を省略する。

10

【0114】

スピーカ群 84 は、スピーカ 65L、65R や図示しない各種スピーカを含んで構成される。LED 群 85 は、フロントパネル 10 に設けられたランプ群 21 や、腰部パネル 12 の装飾パネルを背面側から照明するための光を出射する光源などを含んで構成される。24h ドア開閉監視ユニット 63 は、ミドルドア 41 の開閉の履歴情報を保存する。また、24h ドア開閉監視ユニット 63 は、ミドルドア 41 が開放されたときに、表示装置 11 によりエラー表示を行うための信号を副制御基板 72 (副制御回路 200) に出力する。表示ユニット 212 は、例えば、表示装置 11 を構成する被投影部材 212a、及び、被投影部材 212a の裏側に設けられた表示面が湾曲した別の被投影部材を含んで構成される。

20

【0115】

また、パチスロ 1 は、副制御基板 72 に接続された、ロムカートリッジ基板 86 及び液晶中継基板 87 を有する。なお、ロムカートリッジ基板 86 及び液晶中継基板 87 は、副制御基板 72 とともに副制御基板ケース 57 に収納されている。

【0116】

ロムカートリッジ基板 86 は、サブCPU 102 により実行される各種制御プログラムと、演出用の画像(映像)、音声(スピーカ群 84)、光(LED 群 85)及び通信のデータを管理するための基板である。液晶中継基板 87 は、副制御基板 72 と、表示装置 11 を構成するプロジェクタ機構 211、及び、サブ表示装置 18 との間の接続配線の中継する基板である。なお、プロジェクタ機構 211 及びサブ表示装置 18 については、上述したので、ここでは、それらの説明を省略する。

30

【0117】

<主制御回路>

次に、図8を参照して、主制御基板 71 に実装される主制御回路 90 の構成について説明する。図8は、パチスロ 1 の主制御回路 90 の構成例を示すブロック図である。

【0118】

主制御回路 90 は、マイクロプロセッサ 91 と、クロックパルス発生回路 92 と、電源管理回路 93 と、スイッチングレギュレータ 94 (電源供給手段)とを備える。

【0119】

マイクロプロセッサ 91 は、遊技機用のセキュリティ機能付きマイクロプロセッサである。なお、本実施形態のマイクロプロセッサ 91 では、後述するように、ソースプログラム上で規定可能な該マイクロプロセッサ 91 に特有の様々な命令コード(例えば、後述の「LDQ」命令等:以下、「メインCPU 101 専用命令コード」という)が設けられている。本実施形態では、このメインCPU 101 専用命令コードを用いることにより、処理の効率化やプログラム容量の削減などを実現している。マイクロプロセッサ 91 の内部構成については、後述の図9を参照して詳述し、マイクロプロセッサ 91 に設けられているメインCPU 101 専用命令コードについては、後述の主制御回路が実行する各種処理において詳述する。

40

【0120】

50

クロックパルス発生回路 9 2 は、メイン CPU 作動用のクロックパルス信号を生成し、該生成したクロックパルス信号をマイクロプロセッサ 9 1 に出力する。マイクロプロセッサ 9 1 は、入力されたクロックパルス信号に基づいて、制御プログラムを実行する。

【 0 1 2 1 】

電源管理回路 9 3 は、電源基板 5 3 b (図 7 参照) から供給される直流 1 2 V の電源電圧の変動を管理する。そして、電源管理回路 9 3 は、例えば、電源が投入された際 (電源電圧が 0 V から起動電圧値 (1 0 V) を上回った際) には、リセット信号をマイクロプロセッサ 9 1 の「 X S R S T 」端子に出力し、電断が発生した際 (電源電圧が 1 2 V から停電電圧値 (1 0 . 5 V) を下回った際) には、電断検知信号をマイクロプロセッサ 9 1 の「 X I N T 」端子に出力する。すなわち、電源管理回路 9 3 は、電源投入時に、マイクロプロセッサ 9 1 にリセット信号 (起動信号) を出力する手段 (起動手手段) 、及び、電断発生時に、マイクロプロセッサ 9 1 に電断検知信号 (停電信号) を出力する手段 (停電手段) も兼ねる。

10

【 0 1 2 2 】

スイッチングレギュレータ 9 4 は、 D C / D C 変換回路であり、マイクロプロセッサ 9 1 の直流駆動電圧 (直流 5 V の電源電圧) を生成し、該生成した直流駆動電圧をマイクロプロセッサ 9 1 の「 V C C 」端子に出力する。

【 0 1 2 3 】

< マイクロプロセッサ >

次に、図 9 を参照して、マイクロプロセッサ 9 1 の内部構成について説明する。図 9 は、マイクロプロセッサ 9 1 の内部構成を示すブロック図である。

20

【 0 1 2 4 】

マイクロプロセッサ 9 1 は、メイン CPU 1 0 1 と、メイン ROM 1 0 2 (第 1 記憶手段) と、メイン RAM 1 0 3 (第 2 記憶手段) と、外部バスインターフェース 1 0 4 と、クロック回路 1 0 5 と、リセットコントローラ 1 0 5 と、演算回路 1 0 7 と、乱数回路 1 1 0 と、パラレルポート 1 1 1 と、割込みコントローラ 1 1 2 と、タイマー回路 1 1 3 と、第 1 シリアル通信回路 1 1 4 と、第 2 シリアル通信回路 1 1 5 と、を有する。そして、マイクロプロセッサ 9 1 を構成するこれらの各部は信号バス 1 1 6 を介して互いに接続されている。

【 0 1 2 5 】

メイン CPU 1 0 1 は、クロック回路 1 0 5 で生成されたクロックパルスに基づいて、各種制御プログラムを実行して、遊技動作全般に係る制御を行う。ここで、メイン CPU 1 0 1 の制御動作の一例としてリール停止制御について説明する。

30

【 0 1 2 6 】

メイン CPU 1 0 1 は、リールインデックスを検出してから各リール 3 L , 3 C , 3 L (メインリール) のステッピングモータに対してパルスを出力した回数をカウントする。これにより、メイン CPU 1 0 1 は、各リールの回転角度 (主に、リールが図柄何個分だけ回転したか) を管理する。なお、リールインデックスとは、リールが一回転したことを示す情報である。このリールインデックスは、例えば、発光部及び受光部を有する光センサと、各リールの所定の位置に設けられ、各メインリールの回転により発光部と受光部との間に介在される検知片とを備えたリール位置検出部 (不図示) により検出される。

40

【 0 1 2 7 】

ここで、各リール 3 L , 3 C , 3 L (メインリール) の回転角度の管理について、具体的に説明する。ステッピングモータに対して出力されたパルスの数は、メイン RAM 1 0 3 に設けられたパルスカウンタによって計数される。そして、図柄 1 つ分の回転に必要な所定回数のパルスの出力がパルスカウンタで計数される毎に、メイン RAM 1 0 3 に設けられた図柄カウンタが 1 ずつ加算される。図柄カウンタは、各リールに応じて設けられている。図柄カウンタの値は、リール位置検出部 (不図示) によってリールインデックスが検出されるとクリアされる。

【 0 1 2 8 】

50

すなわち、本実施形態では、図柄カウンタを管理することにより、リールインデックスが検出されてから図柄何個分の回転が行われたのかを管理する。したがって、各リールの各図柄の位置は、リールインデックスが検出される位置を基準として検出される。

【0129】

メインROM102には、メインCPU101により実行される各種制御プログラム、各種データテーブル、副制御回路200に対して各種制御指令(コマンド)を送信するためのデータ等が記憶される。メインRAM103には、制御プログラムの実行により決定された内部当籤役等の各種データを格納する格納領域が設けられる。なお、メインROM102及びメインRAM103の内部構成(メモリーマップ)については、後述の図12を参照して詳述する。

10

【0130】

外部バスインターフェース104は、マイクロプロセッサ91の外部に設けられた各種構成部(例えば、各リール等)が接続された外部信号バス(不図示)と、マイクロプロセッサ104とを電気的に接続するためのインターフェース回路である。クロック回路105は、例えば分周器(不図示)等を含んで構成され、クロックパルス発生回路92から入力されたCPU作動用のクロックパルス信号を、その他の構成部(例えば、タイマー回路113)で使用される周波数のクロックパルス信号に変換する。なお、クロック回路105で生成されたクロックパルス信号は、リセットコントローラ106にも出力される。

【0131】

リセットコントローラ106は、電源管理回路93から入力されたリセット信号に基づいて、IAT(Illegal Address Trap)やWDT(watchdog timer)のリセットを行う。演算回路107は、乗算回路及び除算回路を含んで構成される。例えば、ソースプログラム上において、後述する「MUL(乗算)」命令(後述の図93B参照)を実行するときには、演算回路107がこの「MUL」命令に基づく乗算処理を実行する。

20

【0132】

乱数回路110は、予め定められた範囲の乱数(例えば、0~65535又は0~255)を発生させる。また、図示しないが、乱数回路110は、2バイトのハードラッチ乱数を得るための乱数レジスタ0と、2バイトのソフトラッチ乱数を得るための乱数レジスタ1~3と、1バイトのソフトラッチ乱数を得るための乱数レジスタ4~7とで構成されている。なお、メインCPU101は、乱数回路110で発生させた所定範囲の乱数の中から1つの値を、例えば内部抽籤用の乱数値として抽出する。パラレルポート111は、マイクロプロセッサ91と、マイクロプロセッサ91の外部に設けられた各種回路(例えば、電源管理回路93等)との間で入出力される信号のポート(メモリーマップI/O)である。また、パラレルポート111は、乱数回路110及び割込みコントローラ112にも接続される。スタートスイッチ79はパラレルポート111のPI0~PI4のいずれかの入力ポートに接続され、スタートスイッチ79がオン状態になったタイミング(オンエッジ)で、パラレルポート111から乱数回路110の乱数レジスタ0へラッチ信号が出力される。そして、乱数回路110では、ラッチ信号が入力されることにより乱数レジスタ0がラッチされ、2バイトのハードラッチ乱数が取得される。

30

【0133】

割込みコントローラ112は、パラレルポート111を介して電源管理回路93から入力される電断検知信号、又は、タイマー回路113から1.1172ms周期で入力されるタイムアウト信号に基づいて、メインCPU101による割込処理の実行タイミングを制御する。電源管理回路93から電断検知信号が入力された場合、又は、タイマー回路113からタイムアウト信号が入力された場合には、割込みコントローラ112は、割込処理開始指令を示す割込要求信号をメインCPU101に出力する。メインCPU101は、タイマー回路103からのタイムアウト信号に応じて割込みコントローラ112から入力される割込要求信号に基づいて、入力ポートチェック処理、リール制御処理、通信データ送信処理、7セグLED駆動処理、タイマー更新処理等の各種割込処理(後述の図158参照)を行う。

40

50

【 0 1 3 4 】

タイマー回路 1 1 3 (P T C) は、クロック回路 1 0 5 で生成されたクロックパルス信号 (メイン C P U 作動用のクロックパルス信号を分周器 (不図示) で分周された周波数のクロックパルス信号) で動作する (経過時間をカウントする) 。そして、タイマー回路 1 1 3 は、 1 . 1 1 7 2 m s e c の周期で割込みコントローラ 1 1 2 にタイムアウト信号 (トリガー信号) を出力する。

【 0 1 3 5 】

第 1 シリアル通信回路 1 1 4 は、主制御基板 7 1 から副制御基板 7 2 にデータ (各種制御指令 (コマンド)) を送信する際のシリアル送信動作を制御する回路である。第 2 シリアル通信回路 1 1 5 は、主制御基板 7 1 から試験機用第 2 インターフェースボード 3 0 2 にデータを送信する際のシリアル送信動作を制御する回路である。

10

【 0 1 3 6 】

< 副制御回路 >

次に、図 1 0 を参照して、副制御基板 7 2 に実装される副制御回路 2 0 0 (副制御手段) の構成について説明する。図 1 0 は、パチスロ 1 の副制御回路 2 0 0 の構成例を示すブロック図である。

【 0 1 3 7 】

副制御回路 2 0 0 は、主制御回路 9 0 と電氣的に接続されており、主制御回路 9 0 から送信されるコマンドに基づいて演出内容の決定や実行等の処理を行う。副制御回路 2 0 0 は、基本的に、サブ C P U 2 0 1、サブ R A M 2 0 2、レンダリングプロセッサ 2 0 3、描画用 R A M 2 0 4、ドライバ 2 0 5 を含んで構成される。

20

【 0 1 3 8 】

なお、サブ C P U 2 0 1 は、ロムカートリッジ基板 8 6 に接続される。ドライバ 2 0 5 は、液晶中継基板 8 7 に接続される。すなわち、ドライバ 2 0 5 は、液晶中継基板 8 7 を介してプロジェクタ機構 2 1 1 及びサブ表示装置 1 8 に接続される。

【 0 1 3 9 】

サブ C P U 2 0 1 は、主制御回路 9 0 から送信されたコマンドに応じて、ロムカートリッジ基板 8 6 に記憶されている制御プログラムに従い、映像、音、光の出力の制御を行う。ロムカートリッジ基板 8 6 は、基本的に、プログラム記憶領域とデータ記憶領域とによって構成される。

30

【 0 1 4 0 】

プログラム記憶領域には、サブ C P U 2 0 1 が実行する制御プログラムが記憶される。例えば、制御プログラムには、主制御回路 9 0 との通信を制御するための主基板通信タスクや、演出用の乱数値を抽出し、演出内容 (演出データ) の決定及び登録を行うための演出登録タスクを実行するための各種プログラムが含まれる。また、制御プログラムには、決定した演出内容に基づいて表示装置 1 1 による映像の表示を制御する描画制御タスク、 L E D 群 8 5 等の光源による光の出力を制御するランプ制御タスク、スピーカ群 8 4 による音の出力を制御する音声制御タスク等を実行するための各種プログラムも含まれる。

【 0 1 4 1 】

データ記憶領域には、各種データテーブルを記憶する記憶領域、各演出内容を構成する演出データを記憶する記憶領域、映像の作成に関するアニメーションデータを記憶する記憶領域が含まれる。また、データ記憶領域には、 B G M や効果音に関するサウンドデータを記憶する記憶領域、光の点消灯のパターンに関するランプデータを記憶する記憶領域等も含まれる。

40

【 0 1 4 2 】

サブ R A M 2 0 2 には、決定された演出内容や演出データを登録する格納領域や、主制御回路 9 0 から送信されるサブフラグ (内部当籤役) 等の各種データを格納する格納領域が設けられる。

【 0 1 4 3 】

サブ C P U 2 0 1、レンダリングプロセッサ 2 0 3、描画用 R A M (フレームバッファを

50

含む) 204及びドライバ205は、演出内容により指定されたアニメーションデータに従って映像を作成し、作成した映像を表示装置11(プロジェクタ機構211)及び/又はサブ表示装置18に表示させる。なお、表示装置11(プロジェクタ機構211)及びサブ表示装置18は、副制御基板72により、それぞれ個別に制御される。

【0144】

また、サブCPU201は、演出内容により指定されたサウンドデータに従ってBGMなどの音をスピーカ群84により出力させる。また、サブCPU201は、演出内容により指定されたランプデータに従ってLED群85の点灯及び消灯を制御する。

【0145】

<メインCPUが有する各種レジスタ>

次に、図11を参照しながら、メインCPU101が有する各種レジスタについて説明する。なお、図11は、メインCPU101に含まれる各種レジスタの概略構成図である。

【0146】

メインCPU101は、メイン・レジスタとして、アキュムレータA(以下、「Aレジスタ」という)、フラグ・レジスタF(フラグレジスタ)、汎用レジスタB(以下、「Bレジスタ」という)、汎用レジスタC(以下、「Cレジスタ」という)、汎用レジスタD(以下、「Dレジスタ」という)、汎用レジスタE(以下、「Eレジスタ」という)、汎用レジスタH(以下、「Hレジスタ」という)及び汎用レジスタL(以下、「Lレジスタ」という)を有する。また、メインCPU101は、サブ・レジスタとして、アキュムレータA、フラグ・レジスタF、汎用レジスタB、汎用レジスタC、汎用レジスタD、汎用レジスタE、汎用レジスタH及び汎用レジスタLを汎用レジスタとして有する。なお、各レジスタは、1バイトのレジスタで構成される。

【0147】

また、本実施形態では、BレジスタとCレジスタとをペアレジスタ(以下、「BCレジスタ」という)として用い、DレジスタとEレジスタとをペアレジスタ(以下、「DEレジスタ」という)として用いる。さらに、本実施形態では、HレジスタとLレジスタとをペアレジスタ(以下、「HLレジスタ」という)として用いる。

【0148】

フラグ・レジスタF、Fの各ビットには、図11に示すように、演算処理の結果等を示す所定のフラグ情報がセットされる。例えばビット6(D6)には、演算結果の判定処理において演算結果が「0」であるか否かを示すデータ(ゼロフラグ)がセットされる。具体的には、演算結果が「0」である場合、ビット6にデータ「1」がセットされ、演算結果が「0」でない場合には、ビット6にデータ「0」がセットされる。そして、演算結果の判定処理では、メインCPU101は、ビット6のデータ「0」/「1」を参照して判定(YES/NO)を行う。

【0149】

また、メインCPU101は、拡張レジスタQ(以下、「Qレジスタ」という)を有する。Qレジスタは、1バイトのレジスタで構成される。なお、本実施形態では、後述の各種処理フローの中で説明するように、ソースプログラム上において、このQレジスタを用いてアドレス指定を行う各種メインCPU101専用命令コードが設けられており、この命令コードの使用により、処理の効率化やメインROM102の容量削減などを実現している。なお、Qレジスタを用いてアドレス指定を行う各種メインCPU101専用命令コードでは、Qレジスタには、アドレスの上位側のアドレスデータ(アドレス値)が格納される。なお、Qレジスタには、メインCPU101のリセット直後に、初期値として「F0H」がセットされる。また、Qレジスタを用いた「LDQ, n(8ビットデータ)」命令において、「n」に任意の1バイトのデータをセットして該命令を実行することにより、Qレジスタの値を変更することができる。

【0150】

さらに、メインCPU101は、1バイトのレジスタで構成された、インタラプト・ページアドレス・レジスタI及びメモリ・リフレッシュ・レジスタR、並びに、2バイトのレ

10

20

30

40

50

ジスタで構成された、インデックス・レジスタIX、インデックス・レジスタIY、スタックポインタSP及びプログラムカウンタPCを専用レジスタとして有する。

【0151】

<メインROM及びメインRAMの内部構成(メモリマップ)>

次に、図12A～図12Cを参照しながら、主制御回路90(マイクロプロセッサ91)に含まれるメインROM102及びメインRAM103の内部構成(以下「メモリマップ」という)について説明する。なお、図12Aは、メモリ全体のメモリマップを示す図であり、図12Bは、メインROM102のメモリマップを示す図であり、図12Cは、メインRAM103のメモリマップを示す図である。

【0152】

主制御回路90(マイクロプロセッサ91)が備えるメモリ全体のメモリマップでは、図12Aに示すように、アドレスの先頭(0000H)側から、メインROM102のメモリ領域、メインRAM103のメモリ領域、内蔵レジスタエリア及びXCSデコードエリアが、不使用領域を間に挟んでこの順で、それぞれ所定のアドレスに配置される。

【0153】

メインROM102のメモリマップでは、図12Bに示すように、メインROM102のアドレスの先頭(0000H)側から、プログラムエリア、データエリア、規定外エリア、商標記録エリア、プログラム管理エリア及びセキュリティ設定エリアが、この順で、それぞれ所定のアドレスに配置される。

【0154】

なお、プログラムエリアには、遊技者により実施される遊技の遊技性に関連する各種制御処理において、メインCPU101により実行される各種処理の制御プログラムが記憶される。データエリアには、遊技者により実施される遊技の遊技性に関連する各種制御処理において、メインCPU101により使用される各種データ(例えば、内部抽籤テーブル等のデータテーブル、副制御回路42に対して各種制御指令(コマンド)を送信するためのデータ等)が記憶される。すなわち、プログラムエリアとデータエリアとからなる遊技用ROM領域(遊技用記憶領域)には、遊技店で遊技者が実際に行う遊技の遊技性に関連する制御処理(遊技性に関する処理)に必要な各種プログラム及び各種データが格納される。

【0155】

また、規定外エリアには、遊技者により実施される遊技の遊技性に直接関与しない各種処理(遊技性に影響を与えない処理)の制御プログラム及びデータが記憶される。例えば、パチスロ1の検定試験(試射試験)で使用されるプログラム及びデータ、電断時のチェックサム生成処理や電源復帰時のサムチェック処理などで使用される制御プログラム及びデータ、並びに、不正対策プログラム及びそれに必要なデータ等が、規定外エリアに格納される。

【0156】

メインRAM103のメモリマップでは、図12Cに示すように、メインRAM103のアドレスの先頭(F000H)側から、遊技用RAM領域(所定格納領域、遊技用一時記憶領域)及び規定外RAM領域(規定外一時記憶領域)が、この順で、それぞれ所定のアドレスに配置される。

【0157】

遊技用RAM領域には、遊技者により実施される遊技の遊技性に関連する制御プログラムの実行により決定された例えば内部当籤役等の各種データを一時的に格納する作業領域及びスタックエリアが設けられる。そして、各種データのそれぞれは、遊技用RAM領域内の所定アドレスの作業領域に格納される。

【0158】

また、規定外RAM領域には、遊技者により実施される遊技の遊技性に直接関与しない各種処理の作業領域となる規定外作業領域と、規定外スタックとが設けられる。本実施形態では、この規定外RAM領域を使用して、例えばサムチェック処理等の遊技者により実施

10

20

30

40

50

される遊技の遊技性に直接関与しない各種処理が実行される。

【 0 1 5 9 】

上述のように、本実施形態のパチスロ 1 では、メイン ROM 1 0 2 内において、遊技者により実施される遊技の遊技性に直接関与しない各種処理に使用される各種プログラム及び各種データ（テーブル）を、遊技用 ROM 領域とは異なるアドレスに配置された規定外 ROM 領域（規定外記憶領域）に格納する。また、そのような遊技者により実施される遊技の遊技性に直接関与しない各種処理は、メイン RAM 1 0 3 内において、遊技用 RAM 領域とは異なるアドレスに配置された規定外 RAM 領域を使用して行われる。

【 0 1 6 0 】

このようなメイン ROM 1 0 2 の構成では、従来の規則上においてプログラム等の配置不可とされていた ROM 領域（規定外 ROM 領域）に、遊技者が実際に行う遊技そのものには不要なプログラム及びデータを配置することができる。それゆえ、本実施形態では、遊技用 ROM 領域の容量の圧迫を回避することができる。

10

【 0 1 6 1 】

< 遊技状態の遷移フロー >

次に、図 1 3 及び図 1 4 を参照しながら、本実施形態のパチスロ 1 の主制御回路 9 0（メイン CPU 1 0 1）により管理される各種遊技状態及びその遷移フローについて説明する。なお、図 1 3 A は、パチスロ 1 の基本的な遊技状態の遷移フロー図であり、図 1 3 B は、その遊技状態の移行条件をまとめた表である。また、図 1 4 A は、報知（ART）機能の作動の有無を考慮した遊技状態の遷移フロー図であり、図 1 4 B は、その遊技状態の移行条件をまとめた表である。

20

【 0 1 6 2 】

[基本的な遊技状態の遷移フロー]

本実施形態のパチスロ 1 では、ボーナスゲームの種類として、ビッグボーナス（以下、「BB」と記す）が設けられる。BB は、第 1 種特別役物と呼ばれるレギュラーボーナス（以下、「RB」と記す）に係る役物連続作動装置であり、RB を連続して作動させる。

【 0 1 6 3 】

それゆえ、本実施形態では、主制御回路 9 0 は、ボーナス役の当籤 / 作動（入賞）の有無に基づいて遊技状態を管理する。具体的には、図 1 3 A に示すように、主制御回路 9 0 は、ボーナス役（後述の名称「F__BB1」, 「F__BB2」の内部当籤役）の当籤 / 作動（入賞）の有無に基づいて、「ボーナス非当籤状態」、「フラグ間状態」及び「ボーナス状態」と称する 3 種類の遊技状態を管理する。

30

【 0 1 6 4 】

なお、ボーナス非当籤状態は、ボーナスに非当籤であり、かつ、ボーナスが作動（入賞）していない状態であり、ボーナス状態は、ボーナスが作動している状態である。また、本実施形態では、ボーナス役が内部当籤役として決定されると、ボーナスが入賞するまで複数回の遊技に渡りボーナス役が内部当籤役として持ち越された状態が発生する。フラグ間状態は、ボーナス役が内部当籤役として持ち越されている状態、すなわち、ボーナス役が当籤し、かつ、ボーナスが作動していない状態である。

【 0 1 6 5 】

なお、ボーナス役の当籤の有無は、メイン RAM 1 0 3 に設けられる後述の当り要求フラグ格納領域（後述の図 2 8 ~ 図 3 0 参照）及び持越役格納領域（後述の図 3 1 参照）に格納されるデータに基づいて管理される。また、ボーナスの作動（入賞）の有無は、メイン RAM 1 0 3 に設けられる後述の遊技状態フラグ格納領域（後述の図 3 2 参照）に格納されるデータに基づいて管理される。

40

【 0 1 6 6 】

また、本実施形態では、図 1 3 A に示すように、ボーナスが作動していない遊技状態（ボーナス非当籤状態及びフラグ間状態）において、リプレイに係る内部当籤役の種別及びその当籤確率が互いに異なる、RT 0 遊技状態 ~ RT 5 遊技状態の 6 種類の状態（以下、それぞれ「RT 0 状態」 ~ 「RT 5 状態」という）が設けられる。なお、RT 0 状態、RT

50

2 状態及び R T 5 状態は、リプレイ役が内部当籤役として決定される確率が低確率となる遊技状態であり、R T 1 状態はリプレイ役が内部当籤役として決定される確率が中程度の中確率となる遊技状態である。また、R T 3 状態及び R T 4 状態は、リプレイ役が内部当籤役として決定される確率が高確率となる遊技状態である。なお、本実施形態では、ボーナス非当籤状態の R T 状態は、R T 0 状態 ~ R T 4 状態のいずれかとなり、フラグ間状態の R T 状態は R T 5 状態となる。

【 0 1 6 7 】

それゆえ、本実施形態では、主制御回路 9 0 は、ボーナスが作動していない遊技状態（ボーナス非当籤状態及びフラグ間状態）において、さらに、リプレイに係る内部当籤役の種類及びその当籤確率に基づいて、R T 1 状態 ~ R T 5 状態の 6 種類の状態も管理する。

10

【 0 1 6 8 】

なお、R T 0 状態 ~ R T 5 状態は、メイン R A M 1 0 3 に設けられる後述の遊技状態フラグ格納領域（後述の図 3 2 参照）に格納されるデータに基づいて管理される。具体的には、本実施形態のパチスロ 1 では、R T 1 状態フラグ ~ R T 5 状態フラグの 5 つの R T 状態を示すフラグが設けられ、これらのフラグのオン / オフ状態をメイン R A M 1 0 3 により管理することにより R T 状態が管理される。そして、主制御回路 9 0 は、オン状態である R T 状態フラグに対応する R T 状態を現在の R T 状態として特定する。なお、全ての R T 状態フラグがオフ状態である場合には、主制御回路 9 0 は、現在の R T 状態が R T 0 状態であると特定する。

【 0 1 6 9 】

20

図 1 3 A 及び 1 3 B に示すように、ボーナス非当籤状態においてボーナス役（後述の名称「F _ B B 1」, 「F _ B B 2」の内部当籤役）が内部当籤役として決定されると（移行条件（1）が成立すると）、主制御回路 9 0 は、遊技状態をボーナス非当籤状態からフラグ間状態に移行させる。また、フラグ間状態においてボーナス役が入賞すると（移行条件（2）が成立すると）、主制御回路 9 0 は、遊技状態をフラグ間状態からボーナス状態に移行させる。

【 0 1 7 0 】

また、ボーナス状態において規定枚数（2 1 6 枚）を超えるメダルが払い出され、ボーナス状態が終了すると（移行条件（3）が成立すると）、主制御回路 9 0 は、遊技状態をボーナス状態から R T 1 状態（ボーナス非当籤状態）に移行させる。

30

【 0 1 7 1 】

R T 1 状態において、2 0 ゲームが経過すると（移行条件（4）が成立すると）、主制御回路 9 0 は、遊技状態を R T 1 状態から R T 0 状態に移行させる。また、R T 1 状態において、2 0 ゲームが経過する前に、略称「ベルこぼし目」に係る図柄組合せ（後述の図 2 8 参照）が有効ライン上に表示されると（移行条件（5）が成立すると）、主制御回路 9 0 は、遊技状態を R T 1 状態から R T 2 状態に移行させる。

【 0 1 7 2 】

R T 0 状態において、略称「ベルこぼし目」に係る図柄組合せが有効ライン上に表示されると（移行条件（5）が成立すると）、主制御回路 9 0 は、遊技状態を R T 0 状態から R T 2 状態に移行させる。R T 2 状態において、略称「R T 3 移行リブ」に係る図柄組合せ（後述の図 2 8 参照）が有効ライン上に表示されると（移行条件（6）が成立すると）、主制御回路 9 0 は、遊技状態を R T 2 状態から R T 3 状態に移行させる。

40

【 0 1 7 3 】

R T 3 状態において、略称「R T 4 移行リブ」に係る図柄組合せ（後述の図 2 8 参照）が有効ライン上に表示されると（移行条件（7）が成立すると）、主制御回路 9 0 は、遊技状態を R T 3 状態から R T 4 状態に移行させる。また、R T 3 状態において、略称「ベルこぼし目」又は「R T 2 移行リブ」に係る図柄組合せ（後述の図 2 8 参照）が有効ライン上に表示されると（移行条件（8）が成立すると）、主制御回路 9 0 は、遊技状態を R T 3 状態から R T 2 状態に遊技状態を移行させる。さらに、R T 4 状態において、略称「ベルこぼし目」又は「R T 2 移行リブ」に係る図柄組合せが有効ライン上に表示されると（

50

移行条件(8)が成立すると)、主制御回路90は、遊技状態をRT4状態からRT2状態に遊技状態を移行させる。

【0174】

なお、略称「ベルこぼし目」に係る図柄組合せは、後述の名称「F__3択ベル__1st」、「F__3択ベル__2nd」又は「F__3択ベル__3rd」に係る内部当籤役(小役)が決定され、かつ、停止操作の順序が該小役の種別ごとに定められた押し順に対して不正解であるときに表示される図柄の組合せである(後述の図24参照)。略称「RT2移行リップ」に係る図柄組合せは、後述の名称「F__維持リップ__1st」、「F__維持リップ__2nd」又は「F__維持リップ__3rd」に係る内部当籤役(リプレイ役)が決定され、かつ、停止操作の順序が該リプレイ役の種別ごとに定められた押し順に対して不正解であるとき

10

【0175】

略称「RT3移行リップ」に係る図柄組合せは、後述の名称「F__RT3リップ__1st」、「F__RT3リップ__213」、「F__RT3リップ__231」又は「F__RT3リップ__3rd」に係る内部当籤役(リプレイ役)が決定され、かつ、停止操作の順序が該リプレイ役の種別ごとに定められた押し順に対して正解であるときに表示される図柄の組合せである。また、略称「RT4移行リップ」に係る図柄組合せは、後述の名称「F__RT4リップ__123」、「F__RT4リップ__132」、「F__RT4リップ__2nd」又は「F__RT4リップ__3rd」に係る内部当籤役(リプレイ役)が決定され、かつ、停止操作の順序が該リプレイ役の種別ごとに定められた押し順に対して正解であるときに表示される図柄の組合

20

【0176】

[報知(ART)機能の作動の有無を考慮した遊技状態の遷移フロー]

本実施形態では、主制御回路90(メインCPU101)により、遊技者にとって有利な停止操作を報知する機能(ART機能)の作動の有無が決定される。それゆえ、本実施形態では、ボーナス非作動状態においてART機能の作動/非作動状態も遊技状態として管理する。

【0177】

本実施形態のパチスロ1では、図14Aに示すように、主制御回路90は、非ボーナス作動状態において、報知(ART)の有無に基づいて「一般遊技状態」と「ART遊技状態」とを別個の遊技状態として管理する。すなわち、報知(ART)の有無を考慮した遊技状態の管理では、図14Aに示すように、主制御回路90は、大きな分類として、「ボーナス状態」、「一般遊技状態」及び「ART遊技状態」の3種類の遊技状態を管理する。

30

【0178】

なお、一般遊技状態は、基本的には、遊技者にとって有利な停止操作の情報を報知しない遊技状態(非ART)であり、遊技者にとって不利な遊技状態である。また、一般遊技状態は、RT0~RT4状態のいずれかの状態であり、かつ、ART非当籤の遊技状態である。

【0179】

一方、ART遊技状態は、遊技者にとって有利な停止操作の情報を報知する遊技状態であり、遊技者にとって有利な遊技状態である。また、ART遊技状態は、基本的には、RT4状態であり、かつ、ART当籤中の遊技状態である。なお、本実施形態では、ART当籤後、RT状態がRT4状態まで移行すると、ART遊技が開始される。

40

【0180】

また、本実施形態では、図14Aに示すように、一般遊技状態として、「通常遊技状態」及び「CZ(チャンスゾーン)」と称する2種類の状態が設けられる。

【0181】

通常遊技状態は、遊技者にとって最も不利な遊技状態であるが、通常遊技状態の遊技ではCZへの移行抽籤を行っている。そして、図14A及び14Bに示すように、通常遊技状態の遊技において、CZへの移行抽籤に当籤すると(移行条件(A)が成立すると)、主

50

制御回路 90 は、遊技状態を、通常遊技状態から CZ に移行させる。

【0182】

CZ は、ART 遊技状態への移行に対する期待度が高い遊技状態（チャンスゾーン）であり、CZ 中の遊技では ART への移行抽籤が行われている。そして、図 14A 及び 14B に示すように、CZ 中の遊技において、ART への移行抽籤に非当籤である場合には（移行条件（B）が成立すると）、主制御回路 90 は、遊技状態を、CZ から通常遊技状態に移行させる。一方、CZ 中の遊技において、ART への移行抽籤に当籤すると（移行条件（C）が成立すると）、主制御回路 90 は、遊技状態を、CZ から ART 遊技状態に移行させる。この際、図 14A には示さないが、主制御回路 90 は、遊技状態を、CZ から後述の ART 準備状態を経由して ART 遊技状態（後述の通常 ART 又は CT）に移行させる。

10

【0183】

ART 遊技状態は、上述のように、ART 当籤後に RT 状態が RT4 状態まで移行すると開始される。なお、図 13A で示したように、RT4 状態は、RT0 ~ RT2 状態から RT3 状態を経由して移行するので、ART 当籤後であってもすぐに ART 遊技状態が開始されない。そこで、本実施形態のパチスロ 1 では、ART 当籤後から RT 状態が RT4 状態に移行するまでの期間の遊技状態を ART 準備状態とする。そして、この ART 準備状態の遊技では、RT 状態を RT4 状態に移行させるために必要な停止操作の情報が報知される。

【0184】

また、本実施形態では、図 14A に示すように、ART 遊技状態として、遊技性が互いに異なる、「通常 ART」及び「CT（上乘せチャンス）」と称する 2 種類の状態が設けられる。

20

【0185】

通常 ART は、所定ゲーム数の期間、遊技者にとって有利な停止操作（例えば、払い出されるメダルの枚数が多い図柄組合せを表示させるための停止操作や、RT4 状態を維持するために必要な停止操作）が報知される遊技状態である。また、通常 ART 中の遊技では、CT への移行抽籤が行われる。

【0186】

CT は、遊技者にとって有利な停止操作が報知されるとともに、特定期間（1 セット 8 ゲームの期間）、通常 ART の継続期間を上乘せすることが可能となる遊技状態であり、上乘せチャンスゾーンとして機能する遊技状態である。また、CT 中では、通常 ART の継続期間を消化せずに遊技が行われる。なお、CT 中の遊技性については、後述の図 52A ~ 52C を参照して後で詳述する。

30

【0187】

図 14A 及び 14B に示すように、通常 ART 中の遊技において、CT への移行抽籤に当籤すると（移行条件（D）が成立すると）、主制御回路 90 は、遊技状態を、通常 ART から CT に遊技状態を移行させる。また、通常 ART において、当該通常 ART の継続期間が終了すると（移行条件（E）が成立すると）、主制御回路 90 は、遊技状態を、通常 ART から一般遊技状態（通常遊技状態又は CZ）に移行させる。なお、本実施形態では、ゲーム数により通常 ART の継続期間を管理するが、本発明はこれに限定されず、通常 ART の継続期間の管理方法は任意である。例えば、通常 ART の継続期間を、通常 ART 中に払い出されるメダルの枚数や差枚数により管理してもよいし、通常 ART 中にメダルの払い出しに影響を与える報知を行った回数（ナビ回数）により管理してもよい。

40

【0188】

図 14A 及び 14B に示すように、CT 中の遊技において、CT の継続期間（1 セット 8 ゲーム）が終了すると（移行条件（F）が成立すると）、主制御回路 90 は、遊技状態を、CT から通常 ART に移行させる。

【0189】

また、図 14A に示すように、一般遊技状態（通常遊技状態又は CZ）又は ART 遊技状

50

態（通常ART又はCT）において、ボーナス役が入賞すると（図13A及び13B中で説明した移行条件（2）が成立すると）、主制御回路90は、遊技状態を、一般遊技状態又はART遊技状態からボーナス状態に移行させる。

【0190】

ボーナス状態の遊技では、上述のように、ARTへの移行抽籤を行っており、ボーナス状態の遊技において、ARTへの移行抽籤が非当籤である場合には（移行条件（G）が成立すると）、主制御回路90は、遊技状態を、ボーナス状態から一般遊技状態（通常遊技状態又はCZ）に移行させる。ただし、ART遊技状態（通常ART又はCT）からボーナス状態に移行していた場合には、ボーナス状態の遊技においてARTの移行抽籤に非当籤であっても、主制御回路90は、遊技状態を、ボーナス状態からART遊技状態（通常ART又はCT）に移行させる。一方、ボーナス状態の遊技において、ARTへの移行抽籤に当籤すると（移行条件（H）が成立すると）、主制御回路90は、遊技状態を、ボーナス状態からART遊技状態（通常ART又はCT）に移行させる。なお、上述のように、ボーナス状態の終了時には、RT状態がRT1状態に移行するので、ボーナス状態からART遊技状態に遊技状態を移行する場合には、主制御回路90は、遊技状態を、ART準備状態を経由してART遊技状態に移行させる。

10

【0191】

<メインROMに記憶されているデータテーブルの構成>

次に、図15～図27を参照して、メインROM102に記憶されている各種データテーブルの構成について説明する。なお、一般遊技状態中及びART遊技状態中の遊技性（CZ、通常ART、CTの遊技性）に係る各種抽籤で用いられる各種データテーブルについては、別途、各遊技性の説明と一緒に後述する。

20

【0192】

[図柄配置テーブル]

まず、図15を参照して、図柄配置テーブルについて説明する。図柄配置テーブルは、左リール3L、中リール3C及び右リール3Rのそれぞれの回転方向における各図柄の位置と、各位置に配置された図柄の種類を特定するデータ（以下、図柄コード（図15中の図柄コード表を参照）という）との対応関係を規定する。

【0193】

図柄配置テーブルでは、リールインデックスが検出されたときに、リール表示窓4の枠内における各リールの中段領域に位置する図柄の位置を「0」と規定する。そして、各リールにおいて、図柄位置「0」を基準としてリールの回転方向（図15中の図柄位置「19」から図柄位置「0」に向かう方向）に進む順に、図柄カウンタの値に対応する「0」～「19」が、図柄位置として、各図柄に割り当てられる。

30

【0194】

すなわち、図柄カウンタの値（「0」～「19」）と、図柄配置テーブルとを参照することにより、リール表示窓4の枠内における各リールの上段領域、中段領域及び下段領域に表示されている図柄の種類を特定することができる。なお、本実施形態では、図柄として、「白7」、「青7」、「チリ上1」、「チリ上2」、「チリ下」、「リプレイ」、「帽子」、「サポテン1」、「サポテン2」及び「サポテン3」の10種類の図柄を用いる。

40

【0195】

また、本実施形態では、図柄コード表に示すように、図柄「白7」（図柄コード1）には、データとして「00000001」が割り当てられ、図柄「青7」（図柄コード2）には、データとして「00000010」が割り当てられている。図柄「チリ上1」（図柄コード3）には、データとして「00000011」が割り当てられ、図柄「チリ上2」（図柄コード4）には、データとして「00000100」が割り当てられている。

【0196】

図柄「チリ下」（図柄コード5）には、データとして「00000101」が割り当てられ、図柄「リプレイ」（図柄コード6）には、データとして「00000110」が割り当てられている。図柄「帽子」（図柄コード7）には、データとして「00000111

50

」が割り当てられ、図柄「サボテン1」（図柄コード8）には、データとして「00001000」が割り当てられている。また、図柄「サボテン2」（図柄コード9）には、データとして「00001001」が割り当てられ、図柄「サボテン3」（図柄コード10）には、データとして「00001010」が割り当てられている。

【0197】

[内部抽籤テーブル]

次に、図16及び図17を参照して、内部当籤役を決定する際に参照される内部抽籤テーブルについて説明する。なお、図16は、RT0状態～RT4状態のそれぞれにおいて参照される内部抽籤テーブルである。また、図17Aは、RT5状態において参照される内部抽籤テーブルであり、図17Bは、ボーナス状態において参照される内部抽籤テーブルである。

10

【0198】

内部抽籤テーブルは、遊技状態毎に設けられ、各種内部当籤役と、各内部当籤役が決定される際の抽籤値との対応関係を規定する。なお、抽籤値は、予め設定されたボーナス役や小役等の内部当籤の期待値を調整するための設定（設定1～6）毎に規定される。この設定は、例えば、リセットスイッチ76及び設定用鍵型スイッチ54（図7参照）を用いて変更される。

【0199】

本実施形態の内部抽籤処理では、まず、乱数回路110の乱数レジスタ0により、予め定められた数値の範囲（例えば、0～65535）から抽出される乱数値を、各内部当籤役に対応して規定された抽籤値で順次加算する。次いで、抽籤結果（抽籤値+乱数値）が65535を超えたか否か（抽籤結果がオーバーフローしたか否か）の判定を行う。そして、所定の内部当籤役において、抽籤結果が65535を超えた場合、該内部当籤役が当籤したと判定される。なお、本実施形態の内部抽籤処理では、抽出した乱数値に抽籤値を加算して抽籤を行う例を説明したが、本発明はこれに限定されず、乱数値から抽籤値を減算して、減算結果（抽籤結果）が「0」を下回ったか否か（抽籤結果がアンダーフローしたか否か）を判定して、内部抽籤の当籤/非当籤を決定してもよい。

20

【0200】

それゆえ、本実施形態の内部抽籤処理では、抽籤値として規定されている数値が大きい内部当籤役ほど、決定される確率が高い。なお、各内部当籤役の当籤確率は、「各当籤番号に規定された抽籤値/抽出される可能性のある全ての乱数値の個数（乱数分母：65536）」によって表すことができる。

30

【0201】

RT0状態～RT4状態のそれぞれにおいて参照される内部抽籤テーブルでは、図16に示すように、基本的には、RT状態の種別に応じて、内部当籤役として決定されるリプレイ役の種別及び当籤確率が変化する。例えば、名称「F__チリリプ（No.25）」～「F__リーチ目リプD（No.31）」に係るリプレイ役は、RT0状態～RT3状態では内部当籤役として決定されることなく、RT4状態でのみ内部当籤役として決定される。なお、本実施形態のパチスロ1では、RT4状態中に、名称「F__チリリプ（No.25）」～「F__リーチ目リプD（No.31）」に係るリプレイ役が内部当籤役として決定された場合、特有の制御（後述のフラグ変換）を行う。このフラグ変換については、後で詳述する。

40

【0202】

また、図16に示すように、RT0状態～RT3状態では、名称「F__リーチ目リプA」～「F__リーチ目リプD」のそれぞれの内部当籤役は、名称「F__BB1」又は「F__BB2」に係るボーナス役と重複して決定されることはあるが（No.3～6、15～18参照）、名称「F__リーチ目リプA」～「F__リーチ目リプD」のそれぞれの内部当籤役（リプレイ役）が単独で内部当籤役として決定されることはない。それゆえ、本実施形態において、RT0状態～RT3状態中に名称「F__リーチ目リプA」～「F__リーチ目リプD」に係るリプレイ役が内部当籤役として決定された場合（遊技者からすると名称「F

50

「__リーチ目リブA」～「F__リーチ目リブD」に係るリプレイ役に応じた図柄組合せが表示された場合)、ボーナス役(名称「F__BB1」又は「F__BB2」)が同時に内部当籤役として決定されていることになる。

【0203】

また、フラグ間状態であるRT5状態は、上述のようにボーナス役を内部当籤役として持ち越す遊技状態である。それゆえ、図17Aに示すように、RT5状態において参照される内部抽籤テーブルでは、持ち越しているボーナス役が必ず内部当籤役として決定されるようになっている。また、図17Bに示すように、ボーナス状態において参照される内部抽籤テーブルでは、名称「F__RB役1」～「F__RB役4」のいずれかに係る内部当籤役が必ず当籤する構成になっている(「はずれ」が当籤することはない)。

10

【0204】

[内部当籤役と図柄組合せ(入賞役)との対応表(図柄組合せ決定テーブル)]

次に、図18～図23を参照して、内部当籤役と図柄組合せとの対応表(図柄組合せ決定テーブル)について説明する。図柄組合せ決定テーブルは、各種内部当籤役と、各内部当籤役に対応付けられた、有効ライン(センターライン)上に表示可能な図柄組合せ(コンビネーション)との対応関係を規定する。すなわち、内部当籤役が決定されると、有効ライン上に表示可能な図柄組合せの種別(入賞可能な表示役の種別)が一義的に決定される。

【0205】

各図柄組合せ決定テーブル中の図柄組合せ欄に記載の各種データは、左リール3L、中リール3C及び右リール3Rに渡って設定された有効ラインに沿って表示を許可する図柄組合せを識別するためのデータである。なお、図柄組合せ(表示役)欄に記載の各名称と、具体的な図柄組合せとの関係は、後述の図28～図30の入賞作動フラグ格納領域に示す。

20

【0206】

また、図柄組合せ決定テーブル中に記載の「」印は、決定された内部当籤役において、有効ライン上に表示可能な図柄組合せ(コンビネーション)、すなわち、入賞可能となる表示役を示す。例えば、内部当籤役「F__チリリブ」が決定された場合、図18及び図19に示すように、コンビネーション名称「C__維持リブA__01」～「C__維持リブG__01」、「C__チリリブA__01」～「C__チリリブD__01」に係る図柄組合せが停止表示可能となる。なお、図柄組合せ決定テーブルには、「内部当籤役」が「はずれ」となる場合が規定されていないが、これは、図18～図23に示した図柄組合せテーブルにより規定されている全ての図柄組合せの表示が許可されないことを示す。

30

【0207】

本実施形態のパチスロ1では、主制御回路90(メインCPU101)は、内部当籤役及び遊技状態に応じて停止制御を異ならせ、所定の役が内部当籤役として決定された場合に、図18～図23に示す対応関係の図柄組合せ(コンビネーション)を表示可能とするように左リール3L、中リール3C及び右リール3Rの回転停止制御を行う。なお、図18～図23に示す対応表では、決定された内部当籤役に対して表示可能な全ての図柄組合せを「」印で列挙しているが、「」印が付された図柄組合せであっても、表示されないことがある。

【0208】

本実施形態では、停止表示可能な図柄組合せや現在の遊技状態に応じて停止制御(例えば、優先して引き込む図柄)を異ならせる機能を有し、優先して引き込む図柄の関係上、「」印が付された図柄組合せであっても表示されないことがある。内部当籤役の種別と実際に表示される図柄組合せとの対応関係については、後述の図24及び図25を参照して説明する。

40

【0209】

[非フラグ間状態中の当籤役と停止表示される図柄組合せとの対応関係]

ここで、図24を参照して、フラグ間状態を除く遊技状態(非フラグ間状態)における内部当籤役と停止表示される図柄組合せとの対応関係について説明する。なお、図24は、非フラグ間状態において決定され得る各種内部当籤役と、各内部当籤役決定時に停止表示

50

される図柄組合せ（略称）との対応関係（一部の役については省略）を示す図である。なお、図 2 4 中に記載の図柄組合せの名称は、後述の図 2 8 ~ 図 3 0 の入賞作動フラグ格納領域に示す内容欄に記載の「略称」である。

【 0 2 1 0 】

本実施形態のパチスロ 1 では、遊技者の停止操作の順序（押し順）に応じて表示される図柄組合せが異なる役、いわゆる「押し順役」を設ける。なお、図 2 4 に記載の「押し順正解」に対応付けられた図柄組合せは、押し順に応じて表示される図柄組合せのうち、遊技者にとって有利な図柄組合せであり、「押し順不正解」に対応付けられた図柄組合せは、押し順に応じて表示される図柄組合せのうち、遊技者にとって不利な図柄組合せである。遊技者にとって有利な停止操作を報知する場合、正解となる押し順が報知され、その報知に従って停止操作が行われれば、「押し順正解」に対応付けられた図柄組合せが表示される。また、ART 遊技状態であっても、不正解となる押し順が報知されることもあるが、その内容については、後で詳述する。

10

【 0 2 1 1 】

なお、本実施形態では、押し順役の一部に対しては、その名称の末尾に、正解となる押し順を示す。具体的には、内部当籤役の名称の末尾「1st」は、正解となる押し順が、第 1 停止操作（1 番目に行われる停止操作）が左リール 3 L に対するものであることを意味し、内部当籤役の名称の末尾「2nd」は、正解となる押し順が、第 1 停止操作が中リール 3 C に対するものであることを意味し、内部当籤役の名称の末尾「3rd」は、正解となる押し順が、第 1 停止操作が右リール 3 R に対するものであることを意味する。また、内部当籤役の名称の末尾「123」は、正解となる押し順が「左、中、右」の順であることを意味し、内部当籤役の名称の末尾「132」は、正解となる押し順が「左、右、中」の順であることを意味し、内部当籤役の名称の末尾「213」は、正解となる押し順が「中、左、右」の順であることを意味し、内部当籤役の名称の末尾「231」は、正解となる押し順が「左、右、中」の順であることを意味する。

20

【 0 2 1 2 】

また、以下では、第 1 停止操作が左リール 3 L に対して行われた場合の停止操作順序、具体的には、「左、中、右」及び「左、右、中」の押し順を「順押し」ともいう。さらに、以下では、第 1 停止操作が中リール 3 C 又は右リール 3 R に対して行われた場合の停止操作順序、具体的には、「中、左、右」、「中、右、左」、「右、中、左」、及び、「右、左、中」の押し順を、「変則押し」ともいう。

30

【 0 2 1 3 】

本実施形態では、図 2 4 に示すように、内部当籤役「F__チリリブ」は、押し順に応じて表示される図柄組合せが異なる押し順役であり、押し順が正解である場合には、略称「チリリブ」に係る図柄組合せ（後述の図 2 8 参照）のうちの図 1 8 ~ 図 2 3 に示した表示可能な図柄組合せのいずれかが有効ラインに沿って表示される。一方、押し順が正解でない場合には、略称「リプレイ」に係る図柄組合せ（後述の図 2 8 参照）のうちの図 1 8 ~ 図 2 3 に示した表示可能な図柄組合せのいずれかが有効ラインに沿って表示される。なお、内部当籤役「F__チリリブ」が決定された場合、図 1 8 ~ 図 2 3 に示すように、コンビネーション名称「C__チリリブA__01」、「C__チリリブB__01」又は「C__チリリブC__01」（略称「単チリリブ」又は「2 連チリリブ」：後述の図 2 8 中の略称「チリリブ（否 3 連）」に対応）に係る図柄組合せは表示できるが、コンビネーション名称「C__チリリブD__01」~「C__1 確チリリブD__01」（略称「3 連チリリブ」：後述の図 2 8 中の略称「チリリブ（3 連）」に対応）に係る図柄組合せを表示できない。すなわち、内部当籤役「F__チリリブ」は、略称「3 連チリリブ」に係る図柄組合せを表示できない役である。

40

【 0 2 1 4 】

また、内部当籤役「F__確チリリブ」及び「F__1 確チリリブ」はともに、押し順に応じて表示される図柄組合せが異なる押し順役であり、押し順が正解である場合には、略称「チリリブ」に係る図柄組合せ（後述の図 2 8 参照）のうちの図 1 8 ~ 図 2 3 に示した表示

50

可能な図柄組合せのいずれかが有効ラインに沿って表示される。一方、押し順が正解でない場合には、略称「リプレイ」に係る図柄組合せ（後述の図 2 8 参照）のうちの図 1 8 ~ 図 2 3 に示した表示可能な図柄組合せのいずれかが有効ラインに沿って表示される。なお、内部当籤役「F__確チリリブ」又は「F__1 確チリリブ」が決定された場合、図 1 8 ~ 図 2 3 に示すように、略称「3 連チリリブ」に係る図柄組合せを表示できる。すなわち、内部当籤役「F__確チリリブ」及び「F__1 確チリリブ」は、略称「3 連チリリブ」に係る図柄組合せを表示できる役である。

【 0 2 1 5 】

また、内部当籤役「F__リーチ目リブ A」~「F__リーチ目リブ D」は、押し順に応じて表示される図柄組合せが異なる押し順役であり、押し順が正解である場合には、略称「リーチ目リブ」に係る図柄組合せ（後述の図 2 8 及び図 2 9 参照）のうちの図 1 8 ~ 図 2 3 に示した表示可能な図柄組合せのいずれかが有効ラインに沿って表示される。一方、押し順が正解でない場合には、略称「リプレイ」に係る図柄組合せ（後述の図 2 8 参照）のうちの図 1 8 ~ 図 2 3 に示した表示可能な図柄組合せのいずれかが有効ラインに沿って表示される。

10

【 0 2 1 6 】

なお、本実施形態では、内部当籤役「F__チリリブ」、「F__確チリリブ」、「F__1 確チリリブ」及び「F__リーチ目リブ A」~「F__リーチ目リブ D」の当籤時における正解の押し順は、左リール 3 L に対して第 1 停止操作を行うものである。それゆえ、例えば、内部当籤役「F__リーチ目リブ A」が決定されている遊技において、遊技者が左リール 3 L に対して第 1 停止操作を行った場合には、略称「リーチ目リブ」に係る図柄組合せが停止表示される。なお、本発明はこれに限定されず、内部当籤役「F__チリリブ」、「F__確チリリブ」、「F__1 確チリリブ」及び「F__リーチ目リブ A」~「F__リーチ目リブ D」の当籤時における正解の押し順は、任意に設定することができる。

20

【 0 2 1 7 】

また、内部当籤役「F__維持リブ A」及び「F__維持リブ B」はともに、押し順役ではなく、押し順に関わらず略称「リプレイ」に係る図柄組合せ（後述の図 2 8 参照）のうちの図 1 8 ~ 図 2 3 に示した表示可能な図柄組合せのいずれかが有効ラインに沿って表示される。

【 0 2 1 8 】

また、内部当籤役「F__維持リブ__1 s t」~「F__維持リブ__3 r d」はいずれも、押し順に応じて表示される図柄組合せが異なる押し順役であり、押し順が正解である場合には、略称「リプレイ」に係る図柄組合せ（後述の図 2 8 参照）のうちの図 1 8 ~ 図 2 3 に示した表示可能な図柄組合せのいずれかが有効ラインに沿って表示される。一方、押し順が正解でない場合には、略称「RT 2 移行リブ」に係る図柄組合せ（後述の図 2 8 参照）のうちの図 1 8 ~ 図 2 3 に示した表示可能な図柄組合せのいずれかが有効ラインに沿って表示される。

30

【 0 2 1 9 】

また、内部当籤役「F__RT 3 リブ__1 s t」~「F__RT 3 リブ__3 r d」はいずれも、押し順に応じて表示される図柄組合せが異なる押し順役であり、押し順が正解である場合には、略称「RT 3 移行リブ」に係る図柄組合せ（後述の図 2 8 参照）が有効ラインに沿って表示される。一方、押し順が正解でない場合には、略称「リプレイ」に係る図柄組合せ（後述の図 2 8 参照）のうちの図 1 8 ~ 図 2 3 に示した表示可能な図柄組合せのいずれかが有効ラインに沿って表示される。

40

【 0 2 2 0 】

また、内部当籤役「F__RT 4 リブ__1 2 3」~「F__RT 4 リブ__3 r d」はいずれも、押し順に応じて表示される図柄組合せが異なる押し順役であり、押し順が正解である場合には、略称「RT 4 移行リブ」に係る図柄組合せ（後述の図 2 8 参照）が有効ラインに沿って表示される。一方、押し順が正解でない場合には、略称「リプレイ」に係る図柄組合せ（後述の図 2 8 参照）のうちの図 1 8 ~ 図 2 3 に示した表示可能な図柄組合せのい

50

れかが有効ラインに沿って表示される。

【 0 2 2 1 】

また、内部当籤役「F__3 択ベル__1 s t」～「F__3 択ベル__3 r d」はいずれも、押し順に応じて表示される図柄組合せが異なる押し順役であり、押し順が正解である場合には、略称「ベル」に係る図柄組合せ（後述の図 1 9 参照）が有効ラインに沿って表示される。一方、押し順が正解でない場合には、略称「ベルこぼし目」に係る図柄組合せ（後述の図 2 8 参照）、又は、略称「1 枚出目」に係る図柄組合せ（後述の図 3 0 参照）が表示される。

【 0 2 2 2 】

また、内部当籤役「F__共通ベル」は、押し順役ではなく、押し順に関わらず略称「ベル」に係る図柄組合せ（後述の図 2 9 参照）のうちの図 1 8 ～図 2 3 に示した表示可能な図柄組合せのいずれかが有効ラインに沿って表示される。また、内部当籤役「F__サボ 1」及び「F__サボ 2」はいずれも、押し順役ではなく、押し順に関わらず略称「サボテン」に係る図柄組合せ（後述の図 3 0 参照）のうちの図 1 8 ～図 2 3 に示した表示可能な図柄組合せのいずれかが有効ラインに沿って表示される。

10

【 0 2 2 3 】

また、内部当籤役「弱チェリー」は、押し順役ではなく、押し順に関わらず略称「弱チェリー」に係る図柄組合せ（後述の図 3 0 参照）のうちの図 1 8 ～図 2 3 に示した表示可能な図柄組合せのいずれかが有効ラインに沿って表示される。また、内部当籤役「F__強チリ 1」及び「F__強チリ 2」はいずれも、押し順役ではなく、押し順に関わらず略称「強チェリー」に係る図柄組合せ（後述の図 3 0 参照）のうちの図 1 8 ～図 2 3 に示した表示可能な図柄組合せのいずれかが有効ラインに沿って表示される。

20

【 0 2 2 4 】

[フラグ間状態中の当籤役と停止表示される図柄組合せとの対応関係]

次に、図 2 5 を参照して、フラグ間状態における、内部当籤役と停止表示される図柄組合せとの対応関係について説明する。なお、図 2 5 は、フラグ間状態における、内部当籤役と停止表示される図柄組合せとの対応関係（一部の役については省略）を示す図であり、特に、フラグ間状態中にボーナス役（BB 役）に係る図柄組合せ（コンビネーション名称「C__BB 1」又は「C__BB 2」）を表示可能であるか否を示す図である。

【 0 2 2 5 】

図 2 5 の対応表中の「BB の成立可否」欄に記載の「」印は、BB 役に係る図柄組み合わせが表示可能であることを示し、「x」印は、BB 役に係る図柄組み合わせが表示不可能であることを示す。なお、BB 役に係る図柄組み合わせが表示不可能である場合には、内部当籤役としてボーナス役と重複して決定されている役に係る図柄組み合わせが表示される。例えば、内部当籤役「F__BB 1 + F__チリリップ」が当籤した場合（内部当籤役「F__BB 1」と、内部当籤役「F__チリリップ」とが重複当籤した場合）、図 2 5 に示すように、内部当籤役「F__BB 1」に係る図柄組み合わせを停止表示することはできず、内部当籤役「F__チリリップ」に係る図柄組み合わせが停止表示される。

30

【 0 2 2 6 】

また、フラグ間状態中において、BB 役に係る図柄組み合わせが表示不可能であり、かつ、ボーナス役と重複して決定されている役に係る図柄組み合わせが表示される場合、図 2 4 で説明した押し順正解時の図柄組合せのみを表示可能にしてもよいし、押し順不正解時の図柄組合せのみを表示可能にしてもよい。

40

【 0 2 2 7 】

例えば、内部当籤役「F__BB 1 + F__3 択ベル__1 s t」が当籤した場合、図 2 5 に示すように、内部当籤役「F__BB 1」に係る図柄組み合わせを停止表示することはできないので、内部当籤役「F__3 択ベル__1 s t」に係る図柄組み合わせが停止表示されるが、この際、押し順正解時に表示される略称「ベル」に係る図柄組合せのみを表示可能にし、押し順不正解時に表示される略称「ベルこぼし目」又は「1 枚出目」に係る図柄組み合わせを表示不可能にしてもよい（図 2 4 参照）。また、例えば、内部当籤役「F__BB 1 + F__RT 3

50

リブ__1 s t」が当籤した場合に、押し順不正解時に表示される略称「リプレイ」に係る図柄組合せのみを表示可能にし、押し順正解時に表示される略称「RT3移行リブ」に係る図柄組合せを表示不可能にしてもよい(図24参照)。

【0228】

なお、フラグ間状態では、図25に示すように、ボーナス役(BB役)と、内部当籤役「はずれ」、「F__特殊1」、「F__特殊2」及び「F__特殊3」のいずれかとが重複して決定された場合、BB役に係る図柄組合せを停止表示することができる。

【0229】

[リール停止初期設定テーブル]

次に、図26を参照して、リール停止初期設定テーブルについて説明する。リール停止初期設定テーブルは、内部当籤役と、後述のリール停止制御処理で用いられる各種データとの対応関係を規定する。

【0230】

図26に示すリール停止初期設定テーブルは、内部当籤役(小役当籤番号)と、引込優先順位テーブル選択テーブル番号、引込優先順位テーブル番号及び停止テーブル番号との対応関係を規定する。なお、図26には、参照される遊技状態、及び、内部当籤役の名称も併せて記載する。

【0231】

引込優先順位テーブル選択テーブル番号、及び、引込優先順位テーブル番号は、引込優先順位テーブルの選択処理で用いられるデータである。例えば、リール停止初期設定テーブルにおいて、停止テーブル番号に対応する引込優先順位テーブル番号が規定されていれば、引込優先順位テーブル(後述の図27参照)に規定された引込優先順位テーブル番号に対応する表示役の優先順位に関するデータを取得することができる。一方、リール停止初期設定テーブルにおいて、停止テーブル番号に対応する引込優先順位テーブル番号が規定されていなければ、引込優先順位テーブル選択テーブル(不図示)を参照して、引込優先順位テーブル選択テーブル番号に対応する引込優先順位テーブル番号が決定される。

【0232】

ここで、本実施形態のパチスロ1におけるリールの停止制御(停止図柄位置の決定手法)について簡単に説明する。本実施形態では、ストップスイッチにより停止操作が検出された後、該当するリールの回転が190ms以内(以下、「190ms以内」)に停止するようにリールの停止制御が行われる。具体的には、停止操作が検出されたときの該当リールに応じた図柄カウンタの値に、滑り駒数「0」～「4」のうちの何れかを加算し、得られた値に対応する図柄位置を、リールの回転が停止する図柄位置(以下、「停止予定位置」という)として決定する。なお、停止操作が検出されたときの該当リールに応じた図柄カウンタの値に対応する図柄位置は、リールの回転の停止が開始される図柄位置(以下、「停止開始位置」という)である。

【0233】

すなわち、滑り駒数は、ストップスイッチにより停止操作が検出されてから該当するリールの回転が停止するまでのリールの回転量である。言い換えれば、ストップスイッチにより停止操作が検出されてから該当するリールの回転が停止するまでの期間において、リール表示窓4の該当するリールの中段領域を通過する図柄の数である。これは、ストップスイッチにより停止操作が検出されてから更新された図柄カウンタの値により把握される。

【0234】

図示しない停止テーブルを参照すると、各リールの停止開始位置に応じて滑り駒数が取得される。なお、本実施形態では、停止テーブルに基づいて滑り駒数が取得されるが、これは仮のものであり、取得した滑り駒数が直ちにリールの停止予定位置が決定されるものではない。本実施形態では、停止テーブルに基づいて取得された滑り駒数(以下、「滑り駒数決定データ」という)より適切な滑り駒数が存在する場合には、後述する引込優先順位テーブル(後述の図27参照)を参照して滑り駒数を変更する。そして、滑り駒数決定データは、停止開始位置から最大滑り駒数である4個先の図柄位置までの各図柄について、

10

20

30

40

50

優先順位の比較を行う際の検索順序を決定するために参照される。

【 0 2 3 5 】

[引込優先順位テーブル]

次に、図 2 7 を参照して、引込優先順位テーブルについて説明する。引込優先順位テーブルは、引込優先順位テーブル番号「 0 0 」～「 0 5 」のそれぞれにおける、後述の入賞作動フラグ格納領域（後述の図 2 8 ～図 3 0 参照）の種別毎の引込データ（入賞作動フラグデータ）と、予め定められたその優先順位との対応関係を規定する。

【 0 2 3 6 】

引込優先順位テーブルは、停止テーブル（不図示）に基づいて得られた滑り駒数の他に、より適切な滑り駒数が存在するか否かを検索するために使用される。優先順位は、入賞に係る図柄組合せ（入賞作動フラグ）の種別間で優先的に停止表示される（引き込まれる）順位を規定するデータである。また、図 2 7 では、説明の便宜上、引込データ（入賞作動フラグデータ）の欄には、入賞作動フラグのコンビネーション名称を記載するが、実際の引込優先順位テーブルでは、各引込データは、後述の入賞作動フラグ格納領域（後述の図 2 8 ～図 3 0 参照）に示すように、1 バイトのデータで表され、該 1 バイトデータ中の各ビットに対して固有の図柄組合せ（入賞作動フラグ）が割り当てられる。

10

【 0 2 3 7 】

本実施形態のリール停止制御では、まず、停止テーブル（不図示）に基づいて滑り駒数が取得される。しかしながら、優先順位に基づいて、この滑り駒数の他に、より適切な滑り駒数が存在する場合には、その適切な滑り駒数に変更する。すなわち、本実施形態では、停止テーブルにより取得された滑り駒数に関係なく、内部当籤役によって停止表示を許可する図柄組合せの優先順位に基づいて、より適切な滑り駒数を決定する。

20

【 0 2 3 8 】

本実施形態では、優先順位が上位である図柄組合せの停止表示（引き込み）が、優先順位が下位である図柄組合せの停止表示よりも優先的に行われる。

【 0 2 3 9 】

また、本実施形態では、図 2 7 に示すように、引込優先順位テーブル番号に応じて図柄組合せ（入賞作動フラグ）の優先順位が異なるだけでなく、優先順位の区分数も異なる。具体的には、引込優先順位テーブル番号が「 0 0 」である場合には、優先順位の区分数を 5 とし、引込優先順位テーブル番号が「 0 1 」又は「 0 4 」である場合には、優先順位の区分数を 4 とする。また、引込優先順位テーブル番号が「 0 2 」又は「 0 3 」である場合には、優先順位の区分数を 2 とし、引込優先順位テーブル番号が「 0 5 」である場合には、優先順位の区分数を 3 とする。

30

【 0 2 4 0 】

ここでは、引込優先順位テーブル番号が「 0 0 」である場合の優先順位について説明し、それ以外の引込優先順位テーブル番号における優先順位の説明は省略する。引込優先順位テーブル番号が「 0 0 」である場合の優先順位「 1 」（最上位の優先順位）には、コンビネーション名称「 C __ 9 枚 A __ 0 1 」、 「 C __ 1 確チリリプ C __ 0 1 」、 「 C __ 1 確チリリプ D __ 0 1 」及び「 C __ R T 3 リプ __ 0 1 」に対応する引込データが規定される。

【 0 2 4 1 】

引込優先順位テーブル番号が「 0 0 」である場合の優先順位「 2 」には、コンビネーション名称「 C __ 強 2 枚 C __ 0 1 」～「 C __ 強 2 枚 C __ 0 9 」、 「 C __ 弱 2 枚 B __ 0 1 」～「 C __ 弱 2 枚 B __ 0 3 」、 「 C __ 3 枚 E __ 0 1 」、 「 C __ 3 枚 E __ 0 2 」、 「 C __ 9 枚 F __ 0 1 」～「 C __ 9 枚 F __ 0 3 」、 「 C __ 1 確チリリプ B __ 0 1 」、 「 C __ チリリプ D __ 0 1 」及び「 C __ チリリプ C __ 0 1 」に対応する引込データが規定される。

40

【 0 2 4 2 】

引込優先順位テーブル番号が「 0 0 」である場合の優先順位「 3 」には、コンビネーション名称「 C __ 1 確チリリプ A __ 0 1 」、 「 C __ チリリプ A __ 0 1 」、 「 C __ チリリプ B __ 0 1 」及び「 C __ 維持リプ E __ 0 1 」～「 C __ 維持リプ E __ 0 4 」に対応する引込データが規定される。

50

【0243】

引込優先順位テーブル番号が「00」である場合の優先順位「4」には、コンビネーション名称「C__SP1__01」、「C__SP2__01」、「C__リーチ目リブP__01」、「C__リーチ目リブP__02」、「C__リーチ目リブO__01」、「C__リーチ目リブO__02」、「C__リーチ目リブN__01」、「C__リーチ目リブN__02」、「C__リーチ目リブM__01」、「C__リーチ目リブM__02」、「C__リーチ目リブL__01」～「C__リーチ目リブL__03」、「C__リーチ目リブK__01」～「C__リーチ目リブK__03」、「C__リーチ目リブJ__01」、「C__リーチ目リブI__01」～「C__リーチ目リブI__09」、「C__リーチ目リブH__01」～「C__リーチ目リブH__03」、「C__リーチ目リブG__01」、「C__リーチ目リブF__01」、「C__リーチ目リブF__02」、「C__リーチ目リブE__01」、「C__リーチ目リブD__01」、「C__リーチ目リブD__02」、「C__リーチ目リブC__01」～「C__リーチ目リブC__03」、「C__リーチ目リブB__01」、「C__リーチ目リブB__02」、「C__リーチ目リブA__01」、「C__維持リブF__01」、「C__維持リブF__02」、「C__維持リブD__01」～「C__維持リブD__04」、「C__維持リブC__01」～「C__維持リブC__03」、「C__維持リブB__01」、「C__維持リブB__02」及び「C__維持リブA__01」に対応する引込データが規定される。

10

【0244】

また、引込優先順位テーブル番号が「00」である場合の優先順位「5」（最下位の優先順位）には、コンビネーション名称「C__BB1」及び「C__BB2」に対応する引込データが規定される。

20

【0245】

<メインRAMに設けられている格納領域の構成>

次に、図28～図35を参照して、メインRAM103に設けられる各種格納領域の構成について説明する。

【0246】

[当り要求フラグ格納領域及び入賞作動フラグ格納領域]

まず、図28～図30を参照して、当り要求フラグ格納領域（内部当籤役格納領域）及び入賞作動フラグ格納領域（表示役格納領域）の構成について説明する。なお、本実施形態では、当り要求フラグ格納領域（フラグデータ格納領域、当籤フラグデータ格納領域）と、入賞作動フラグ格納領域（入賞フラグデータ格納領域）とは、互いに同じ構成を有する。

30

【0247】

本実施形態では、当り要求フラグ格納領域は、それぞれ1バイトのデータにより表される当り要求格納領域0～11で構成され、入賞作動フラグ格納領域は、それぞれ1バイトのデータにより表される入賞作動格納領域0～11で構成される。なお、当り要求フラグ格納領域及び入賞作動フラグ格納領域の各格納領域に格納されるデータは、図28～図30中の「データ」欄の1バイトデータのみであるが、図28～図30では、説明の便宜上、各格納領域のビットに対応付けられた、各リールの図柄組合せ（図中では、左リール3Lの図柄、中リール3Cの図柄及び右リール3Rの図柄の順で記載）、その名称（コンビネーション名称）及び略称、並びに、メダルの払出枚数も併せて記載する。

40

【0248】

当り要求フラグ格納領域0～11のそれぞれにおいて、所定のビットに「1」が格納されているとき、その所定のビットに対応する内部当籤役が内部当籤したことを示す。また、入賞作動格納領域0～11のそれぞれにおいて、所定のビットに「1」が格納されているとき、その所定のビットに対応する表示役（入賞作動フラグ）が入賞したことを示す。すなわち、所定のビットに「1」が格納されているとき、その所定のビットに対応する内部当籤役の各種図柄組合せが有効ライン上に表示されたことを示す。

【0249】

また、当り要求フラグ格納領域及び入賞作動フラグ格納領域では、図28～図30に示すように、各格納領域内の一つのビット（フラグ）に対して、複数の図柄組合せ（コンビネ

50

ーション)が割り当てられているものもある。すなわち、そのようなフラグに対しては、停止表示可能な図柄組合せ(入賞可能なコンビネーション)が複数存在することを意味する。

【0250】

例えば、当り要求格納領域5及び入賞作動格納領域5のビット5には、図柄組合せ「サポテン2」-「白7」-「帽子」(コンビネーション名称「C__維持リプC__01」)、図柄組合せ「サポテン2」-「チリ上1」-「帽子」(コンビネーション名称「C__維持リプC__02」)、及び、図柄組合せ「サポテン2」-「サポテン2」-「帽子」(コンビネーション名称「C__維持リプC__03」)の3つの図柄組合せが割り当てられている。それゆえ、当り要求格納領域5のビット5に「1」が格納されている場合には、この3つの図柄組合せが有効ライン上に停止表示可能であることを示す。また、入賞作動格納領域5のビット5に「1」が格納されている場合には、この3つの図柄組合せのいずれかが有効ライン上に表示されたことを示す。

10

【0251】

[持越役格納領域]

次に、図31を参照して、持越役格納領域の構成について説明する。本実施形態では、持越役格納領域は、1バイトのデータ格納領域で構成される。

【0252】

内部抽籤の結果、内部当籤役「F__BB1」又は「F__BB2」が決定されたときには、その内部当籤役(BB役)は、持越役として持越役格納領域に格納される。持越役格納領域に格納された持越役は、対応する図柄組合せが有効ライン上に表示されるまでクリアされずに保持される。また、持越役格納領域に持越役が格納されている間、内部抽籤によって決定された内部当籤役に加えて、持越役が当り要求格納領域に格納される。

20

【0253】

[遊技状態フラグ格納領域]

次に、図32を参照して、遊技状態フラグ格納領域の構成について説明する。遊技状態フラグ格納領域は、1バイトのデータ格納領域で構成される。本実施形態では、図32に示すように、遊技状態フラグ格納領域の各ビットに対して固有のボーナスの種別又はRTの種別が割り当てられる。

【0254】

遊技状態フラグ格納領域において、所定のビットに「1」が格納されているとき、その所定のビットに該当するボーナスゲーム又はRTの作動が行われていることを示す。例えば、遊技状態フラグ格納領域のビット0に「1」が格納されているときには、ビッグボーナス「BB」の作動が行われており、遊技状態がBB遊技状態であることを示す。また、例えば、遊技状態フラグ格納領域のビット3に「1」が格納されているときは、遊技状態がRT3状態であることを示す。

30

【0255】

[作動ストップボタン格納領域]

次に、図33を参照して、作動ストップボタン格納領域の構成について説明する。作動ストップボタン格納領域は、1バイトのデータ格納領域で構成され、1バイトからなる作動ストップボタンフラグを格納する。作動ストップボタンフラグにおいて、各ビットには、ストップボタンの操作状態が割り当てられる。

40

【0256】

例えば、左ストップボタン17Lが今回押されたストップボタン、つまり、作動ストップボタンである場合には、作動ストップボタン格納領域のビット0に「1」が格納される。また、例えば、左ストップボタン17Lが未だに押されていないストップボタン、つまり、有効ストップボタンである場合には、ビット4に「1」が格納される。メインCPU101は、作動ストップボタン格納領域に格納されているデータに基づいて、今回押されたストップボタンと未だに押されていないストップボタンとを識別する。

【0257】

50

[押下順序格納領域]

次に、図 3 4 を参照して、押下順序格納領域の構成について説明する。押下順序格納領域は、1 バイトのデータ格納領域で構成され、1 バイトからなる押下順序フラグを格納する。

【 0 2 5 8 】

押下順序フラグにおいて、各ビットには、ストップボタンの押下順序の種別が割り当てられる。例えば、ストップボタンの押下順序が「左、中、右」である場合には、押下順序格納領域のビット 0 に「1」が格納される。

【 0 2 5 9 】

[図柄コード格納領域]

次に、図 3 5 を参照して、図柄コード格納領域の構成について説明する。本実施形態では、図柄コード格納領域は、それぞれ 1 バイトのデータにより表される図柄コード格納領域 0 ~ 1 1 で構成される。なお、図柄コード格納領域は、当り要求フラグ格納領域及び入賞作動フラグ格納領域（図 2 8 ~ 図 3 0 参照）と同様の構成となる。

【 0 2 6 0 】

図柄コード格納領域では、有効ライン上に停止可能な図柄組合せ（コンビネーション）に対応するビットに「1」が格納される。なお、全てのリールが停止後、図柄コード格納領域 0 ~ 1 1 には、表示役（入賞作動フラグ）に対応する図柄コードが格納される。

【 0 2 6 1 】

[内部当籤役と各種サブフラグとの関係]

一般遊技状態や A R T 遊技状態において、主制御回路 9 0 による各種抽籤では各種データテーブルを参照するが、この際に用いるパラメータとして、本実施形態では、内部当籤役だけでなく、内部当籤役に対応する別の名称の各種パラメータ（以下、「サブフラグ（第 1 のサブフラグ）」、「サブフラグ E X（第 2 のサブフラグ）」及び「サブフラグ D」という）も用いる。それゆえ、本実施形態では、主制御回路 9 0 により、内部当籤役を各種サブフラグに変換する処理を行う（後述の図 1 0 4 中のサブフラグ変換処理、フラグ変換処理、サブフラグ圧縮処理参照）。なお、本実施形態では、内部当籤役に関する情報（通信パラメータ）として、サブフラグがスタートコマンドにセットされ、主制御回路 9 0 から副制御回路 2 0 0 に送信される。

【 0 2 6 2 】

ここで、図 3 6 及び図 3 7 を参照して、内部当籤役と各種サブフラグとの対応関係について説明する。図 3 6 は、内部当籤役（小役当籤番号）と各種サブフラグとの対応関係を示す図であり、図 3 7 は、内部当籤役（特賞当籤番号）とサブフラグとの対応関係を示す図である。

【 0 2 6 3 】

本実施形態のフラグ変換処理では、まず、同じ種別に属する複数の内部当籤役を一つのサブフラグにまとめる。本実施形態では、このフラグ変換処理により、図 3 6 に示すように、小役及びリプレイ役に関する 3 2 種類の内部当籤役（小役当籤番号）が、1 8 種類のサブフラグ（「0 1」~「1 8」：フラグデータ）に変換される。例えば、内部当籤役「F__維持リブ__1 s t（1 0：小役当籤番号）」~「F__維持リブ__3 r d（1 2）」は、サブフラグ「押し順リブ 1（0 9：フラグデータ）」にまとめられる。なお、内部当籤役「はずれ」に対しては、サブフラグ「ハズレ（0 0）」が割り当てられる。

【 0 2 6 4 】

また、本実施形態のフラグ変換処理では、図 3 6 に示すように、サブフラグ「ハズレ（0 0）」を含む 1 9 種類のサブフラグ（「0 0」~「1 8」）が、9 種類のサブフラグ E X（「0 0」~「0 8」：フラグデータ）に変換される。それゆえ、この変換処理では、サブフラグデータをさらに圧縮することができる。なお、この際、本実施形態では、抽籤（フラグ変換抽籤）によりサブフラグをサブフラグ E X に変換する。具体的には、次のように変換される。

【 0 2 6 5 】

サブフラグ「ハズレ（0 0）」は、フラグ変換抽籤の結果に関係なく、サブフラグ E X 「

10

20

30

40

50

ハズレ(00)」に変換され、サブフラグ「2連チリリブ(01)」は、フラグ変換抽籤の結果に関係なく、サブフラグEX「リプレイ(01)」に変換される。

【0266】

サブフラグ「3連チリリブA(02)」及びサブフラグ「3連チリリブB(03)」は、フラグ変換抽籤に当籤した場合(後述の「変換有り」の場合)、サブフラグEX「確定役(06)」又は「3連チリリブ(07)」に変換され、フラグ変換抽籤に非当籤であった場合(後述の「変換無し」の場合)には、サブフラグEX「リプレイ(01)」に変換される。

【0267】

サブフラグ「リーチ目リブ1(04)」～「リーチ目リブ4(07)」は、フラグ変換抽籤に当籤した場合、サブフラグEX「確定役(06)」又は「リーチ目リブ(08)」に変換され、フラグ変換抽籤に非当籤であった場合には、サブフラグEX「リプレイ(01)」に変換される。

10

【0268】

サブフラグ「リプレイ(08)」及び「押し順リブ1(09)」～「押し順リブ3(11)」は、フラグ変換抽籤の結果に関係なく、サブフラグEX「リプレイ(01)」に変換され、サブフラグ「押し順ベル(12)」及び「共通ベル(13)」は、フラグ変換抽籤の結果に関係なく、サブフラグEX「ベル(02)」に変換される。

【0269】

サブフラグ「サボテン(14)」、「弱チェリー(15)」及び「強チェリー(16)」は、フラグ変換抽籤の結果に関係なく、それぞれサブフラグEX「サボテン(03)」、「弱チェリー(04)」及び「強チェリー(05)」に変換される。また、サブフラグ「リーチ目1(17)」及び「リーチ目2(18)」は、フラグ変換抽籤の結果に関係なく、サブフラグEX「ハズレ(00)」に変換される。

20

【0270】

上述のように、本実施形態では、実質、サブフラグ「3連チリリブA(02)」、「3連チリリブB(03)」及び「リーチ目リブ1(04)」～「リーチ目リブ4(07)」のみがフラグ変換抽籤の対象となる。なお、上述したフラグ変換抽籤に用いられる抽籤テーブルについては、後で詳述する。

【0271】

さらに、本実施形態のフラグ変換処理では、図36に示すように、9種類のサブフラグEX(「00」～「08」)が7種類のサブフラグD(「00」～「06」)に変換される。それゆえ、この変換処理では、より一層、サブフラグデータを圧縮することができる。なお、この変換処理では抽籤を行わず、次のようにして、サブフラグEXとサブフラグDとを対応付けて変換を行う。

30

【0272】

サブフラグEX「ハズレ(00)」、「リプレイ(01)」及び「ベル(02)」は、サブフラグD「ハズレ(00)」に変換される。サブフラグEX「サボテン(03)」は、サブフラグD「サボテン(01)」に変換され、サブフラグEX「弱チェリー(04)」は、サブフラグD「弱チェリー(02)」に変換され、サブフラグEX「強チェリー(05)」は、サブフラグD「強チェリー(03)」に変換される。

40

【0273】

また、サブフラグEX「確定役(06)」は、サブフラグD「確定役(04)」に変換され、サブフラグEX「3連チリリブ(07)」は、サブフラグD「3連チリリブ(05)」に変換され、サブフラグEX「リーチ目リブ(08)」は、サブフラグD「リーチ目リブ(06)」に変換される。

【0274】

また、本実施形態のフラグ変換処理では、図37に示すように、内部当籤役「F__BB1(01:特賞当籤番号)」及び「F__BB2(02)」はいずれも、サブフラグ「BB」に変換される。

50

【 0 2 7 5 】

[サブフラグ E X 変換時の遊技性]

ここで、上述した内部当籤役をサブフラグ及びサブフラグ E X に変換する処理の過程、及び、サブフラグ E X 変換時の遊技性の一例を、図 3 8 A 及び 3 8 B を参照して説明する。図 3 8 A は、内部当籤役「 F __ 確チリリブ 」又は「 F __ 1 確チリリブ 」が決定された場合のフラグ変換過程を示す図であり、図 3 8 B は、内部当籤役「 F __ リーチ目リブ A 」～「 F __ リーチ目リブ D 」のいずれかが決定された場合のフラグ変換過程を示す図である。

【 0 2 7 6 】

なお、本実施形態のパチスロ 1 では、 R T 4 遊技状態中に内部当籤役「 F __ 確チリリブ 」、 「 F __ 1 確チリリブ 」及び「 F __ リーチ目リブ A 」～「 F __ リーチ目リブ D 」のいずれかが単独で内部当籤役として決定されると、フラグ変換抽籤を行う。そして、本実施形態では、このフラグ変換抽籤に当籤した場合、特別な特典（例えば、 A R T ゲーム数の上乘せや C T 当籤）が付与される。

【 0 2 7 7 】

例えば、内部当籤役「 F __ 確チリリブ 」又は「 F __ 1 確チリリブ 」が決定された場合、図 3 8 A に示すように、内部当籤役「 F __ 確チリリブ 」及び「 F __ 1 確チリリブ 」は、それぞれサブフラグ「 3 連チリリブ A (0 2) 」及び「 3 連チリリブ B (0 3) 」に変換される。

【 0 2 7 8 】

次いで、サブフラグ「 3 連チリリブ A (0 2) 」及び「 3 連チリリブ B (0 3) 」は、フラグ変換抽籤に当籤すると、サブフラグ E X 「 3 連チリリブ (0 7) 」又は「 確定役 (0 6) 」に変換される。一方、フラグ変換抽籤に非当籤であった場合には、サブフラグ「 3 連チリリブ A (0 2) 」及び「 3 連チリリブ B (0 3) 」はともに、サブフラグ E X 「 リプレイ (0 1) 」に変換される。

【 0 2 7 9 】

なお、図 2 4 で説明したように、内部当籤役「 F __ 確チリリブ 」又は「 F __ 1 確チリリブ 」が当籤した場合、押し順正解時には略称「 3 連チリリブ 」に係る図柄組合せが表示され、押し順不正解時には略称「 リプレイ 」に係る図柄組合せが表示される。それゆえ、本実施形態では、内部当籤役「 F __ 確チリリブ 」又は「 F __ 1 確チリリブ 」が決定され、かつ、フラグ変換抽籤に当籤した場合、内部当籤役「 F __ 確チリリブ 」及び「 F __ 1 確チリリブ 」はいずれも、サブフラグ E X 「 3 連チリリブ (0 7) 」又は「 確定役 (0 6) 」の役として扱われる。

【 0 2 8 0 】

そして、このフラグ変換過程によって内部当籤役「 F __ 確チリリブ 」又は「 F __ 1 確チリリブ 」がサブフラグ E X 「 3 連チリリブ (0 7) 」又は「 確定役 (0 6) 」に変換されると、略称「 3 連チリリブ 」に係る図柄組合せを表示するための情報が報知される（例えば、遊技者に対して順押しでチリ図柄を狙わせる旨の情報が報知される）。一方、このフラグ変換過程において、フラグ変換抽籤が非当籤となり、内部当籤役「 F __ 確チリリブ 」又は「 F __ 1 確チリリブ 」がサブフラグ E X 「 リプレイ (0 1) 」に変換されると、略称「 リプレイ 」に係る図柄組合せを表示するための情報が報知される（例えば、順押し以外の押し順（変則押し）が報知される）。

【 0 2 8 1 】

また、例えば、内部当籤役「 F __ リーチ目リブ A 」～「 F __ リーチ目リブ D 」のいずれかが決定された場合、図 3 8 B に示すように、内部当籤役「 F __ リーチ目リブ A 」～「 F __ リーチ目リブ D 」は、それぞれサブフラグ「 リーチ目リブ 1 (0 4) 」～「 リーチ目リブ 4 (0 7) 」に変換される。

【 0 2 8 2 】

次いで、サブフラグ「 リーチ目リブ 1 (0 4) 」～「 リーチ目リブ 4 (0 7) 」は、フラグ変換抽籤に当籤すると、サブフラグ E X 「 リーチ目リブ (0 8) 」又は「 確定役 (0 6) 」に変換される。一方、フラグ変換抽籤に非当籤であった場合には、サブフラグ「 リー

10

20

30

40

50

チ目リブ1(04)」～「リーチ目リブ4(07)」は、サブフラグEX「リブレイ(01)」に変換される。

【0283】

なお、図24で説明したように、内部当籤役「F__リーチ目リブA」～「F__リーチ目リブD」のいずれかが当籤した場合、押し順正解時には略称「リーチ目リブ」に係る図柄組合せが表示され、押し順不正解時には略称「リブレイ」に係る図柄組合せが表示される。それゆえ、本実施形態では、内部当籤役「F__リーチ目リブA」～「F__リーチ目リブD」のいずれかが決定され、かつ、フラグ変換抽籤に当籤した場合、内部当籤役「F__リーチ目リブA」～「F__リーチ目リブD」はいずれも、サブフラグEX「リーチ目リブ(08)」又は「確定役(06)」の役として扱われる。

10

【0284】

そして、このフラグ変換過程によって内部当籤役「F__リーチ目リブA」～「F__リーチ目リブD」が例えばサブフラグEX「リーチ目リブ(08)」又は「確定役(06)」に変換されると、略称「リーチ目リブ」に係る図柄組合せを表示するための情報が報知される(例えば、遊技者に対して順押しで図柄「白7」を狙わせる旨の情報が報知される)。一方、このフラグ変換過程において、フラグ変換抽籤が非当籤となり、内部当籤役「F__リーチ目リブA」～「F__リーチ目リブD」がサブフラグEX「リブレイ(01)」に変換されると、略称「リブレイ」に係る図柄組合せを表示するための情報が報知される(例えば、順押し以外の押し順(変則押し)が報知される)。

【0285】

また、本実施形態では、図38A又は38Bに示すフラグ変換過程において、フラグ変換抽籤に当籤して報知に従い遊技者が停止操作を行うと、略称「3連チリリブ」又は「リーチ目リブ」に係る図柄組合せが有効ライン上に停止表示され、特別な特典が付与される。この付与処理は、実質的には処理上において、パチスロ1がフラグ変換抽籤に当籤したことに応じて特別な特典を遊技者に付与することになるが、遊技者に対しては、略称「3連チリリブ」に係る図柄組合せが表示されたことにより、特別な特典が付与されたと感じさせることができる。

20

【0286】

パチスロの遊技性を高めるためには、特典が付与される図柄組合せの出現頻度が一定であるよりも、状態に応じて異なる方が好ましい場合がある。停止制御(表示される図柄組合せ)は、内部当籤役の種類によって異なるため、特典が付与される図柄組合せの出現頻度を状態に応じて異ならせる手法としては、内部当籤役の当籤確率を異ならせる手法も考えられる(パチスロ1では、内部当籤役の当籤確率は、ボーナスの作動の有無やRT状態に応じて異ならせることができるため、例えば、ART遊技状態に対応するRT状態として、RT4状態だけでなく、RT6状態やRT7状態などの他のRT状態を設けるという手法も考えられる)。しかしながら、内部当籤役の当籤確率を異ならせる契機(RT状態の移行契機)は限定されているため、遊技性(興趣)の向上という観点では、この手法は柔軟性に欠けている。

30

【0287】

それに対して、本実施形態のパチスロ1では、内部当籤役の当籤確率を変えずに、内部当籤役を決定するための内部抽籤に加え、フラグ変換抽籤及びその抽籤結果に基づく報知を行うことにより、特典が付与される図柄組合せの出現頻度を状態に応じて柔軟に異ならせることができる。すなわち、フラグ変換抽籤に当籤し易い状態では、特典が付与される図柄組合せの出現頻度を上げることができ、逆に、フラグ変換抽籤に当籤し難い状態では、特典が付与される図柄組合せの出現頻度を下げることができる。

40

【0288】

<一般遊技状態中の遊技性>

次に、図39A～39Cを参照して、一般遊技状態中の遊技の流れについて説明する。本実施形態のパチスロ1では、一般遊技状態中において、遊技状態が通常遊技状態からCZに移行し、その後、遊技状態がCZからART遊技状態に移行することにより、一般遊技

50

状態（非ART遊技状態）からART遊技状態への移行が行われる（図14A及び14B参照）。

【0289】

図39Aは、一般遊技状態中において、遊技状態が通常遊技状態からCZに移行する際の遊技の流れを示す図である。通常遊技状態は、図39Aに示すように、CZの抽籤状態として低確率状態と高確率状態とを有する。この低確率状態及び高確率状態は、通常遊技状態中に行われるCZ抽籤に当籤する期待度が互いに異なる状態であり、低確率状態はCZ抽籤に当籤し難い状態であり、高確率状態はCZ抽籤に当籤し易い状態である。そして、通常遊技状態中の遊技において行われるCZ抽籤に当籤した場合には、遊技状態が通常遊技状態からCZに移行する。

10

【0290】

なお、本実施形態のパチスロ1では、CZ（チャンスゾーン）として、「CZ1」、「CZ2」及び「CZ3」の複数のチャンスゾーンを設ける。CZ1～CZ3は、CZ中の遊技で行われるART抽籤に当籤する期待度が互いに異なるチャンスゾーンであり、CZ3は、ART抽籤に必ず当籤するチャンスゾーンであり、CZ1及びCZ2は、所定の確率でART抽籤に当籤するチャンスゾーンである。通常遊技状態中の遊技で行われるCZ抽籤では、CZの当籤/非当籤だけでなく、当籤時に移行するCZの種別（CZ1～CZ3のいずれか）も決定される（後述の図41参照）。

【0291】

図39Bは、遊技状態が一般遊技状態のCZ1及びCZ2からART遊技状態に移行する際の遊技の流れを示す図である。CZ1及びCZ2はともに、前半部と後半部とから構成される。前半部は、CZ中の遊技で行われるART抽籤に当籤する期待度のランクを昇格させる期間であり、後半部は、ランクに基づくART抽籤の抽籤結果を所定の演出（本実施形態では、キャラクタによるバトル演出）により報知する期間である。

20

【0292】

CZ1中では、ランクとして6段階のモード（モード1～6）が用意され、モードが上がるほど、ART抽籤に当籤する期待度が高くなる。CZ1の前半部では、第1の所定ゲーム数（例えば、最大で12ゲーム）の期間、継続して遊技が行われ、内部当籤役に基づいてモードの昇格抽籤が行われる。そして、CZ1の後半部の1ゲーム目では、前半部で昇格させたモード（前半部終了時点のモード）に基づいてART抽籤が行われる。

30

【0293】

また、CZ2中では、ランクとして10段階のポイントが用意され、ポイントが上がるほど、ART抽籤に当籤する期待度が高くなる。CZ2の前半部では、第2の所定ゲーム数（例えば、最大で15ゲーム）の期間、継続して遊技が行われ、内部当籤役に基づいてポイントの昇格抽籤が行われる。そして、CZ2の後半部の1ゲーム目では、前半部で昇格させたポイント（前半部終了時点のポイント）に基づいてART抽籤が行われる。

【0294】

CZ1の後半部では、味方キャラクタと敵キャラクタAとが対戦するバトル演出が行われ、CZ2の後半部では、味方キャラクタと敵キャラクタBとが対戦するバトル演出が行われる。このバトル演出は、第3の所定ゲーム数（例えば、最大で4ゲーム）の期間の遊技に渡って行われる。また、バトル演出の勝敗は、ART抽籤の結果に基づいて管理（決定）され、ART抽籤に当籤している場合には、バトル演出で味方キャラクタが勝利し、非当籤である場合には、バトル演出で敵キャラクタが勝利する。

40

【0295】

また、CZ1及びCZ2の各後半部（バトル演出中）では、毎ゲーム、内部当籤役に基づいてART抽籤が行われる。そして、このART抽籤に当籤すると、バトル演出の結果が書き換えられる。例えば、バトル演出中にいわゆる「レア」役が内部当籤役として決定されると、ART抽籤が行われ、その結果に基づいてバトル演出の結果が書き換えられる。

【0296】

CZ1及びCZ2において、ARTに非当籤の場合には、後半部のバトル演出で敗北し、

50

基本的には、その後、遊技状態が通常遊技状態に移行する。一方、C Z 1 及び C Z 2 において、A R T に当籤している場合には、後半部のバトル演出で勝利し、その後、遊技状態が C Z から A R T 準備状態を経由して通常 A R T に移行する。なお、本実施形態では、C Z 1 及び C Z 2 の前半部の遊技において、フリーズが発生する場合があります。その場合には、遊技状態が C Z から A R T 準備状態を経由して、通常 A R T ではなく C T (上乗せチャンスゾーン) に移行する。

【 0 2 9 7 】

図 3 9 C は、遊技状態が一般遊技状態の C Z 3 から A R T 遊技状態に移行する際の遊技の流れを示す図である。C Z 3 は、第 4 の所定ゲーム数 (例えば、最大で 1 7 ゲーム) の期間、継続して遊技が行われる。そして、C Z 3 では、毎ゲーム、内部当籤役に基づいて A R T 抽籤が行われる。

10

【 0 2 9 8 】

C Z 3 は、A R T 抽籤に当籤した時点で終了し、その次のゲーム以降、遊技状態が C Z 3 から A R T 準備状態を経由して C T (上乗せチャンスゾーン) に移行する。また、C Z 3 では、フリーズが発生する場合があります。その場合にも、次ゲーム以降、遊技状態が C Z 3 から A R T 準備状態を経由して C T (上乗せチャンスゾーン) に移行する。一方、C Z 3 において、A R T 抽籤に当籤せずに C Z 3 の遊技期間 (第 4 の所定ゲーム数) が経過した場合、遊技状態が C Z 3 から A R T 準備状態を経由して通常 A R T に移行する。すなわち、本実施形態では、C Z 3 は、A R T 遊技状態への移行が確定しているチャンスゾーンである。

20

【 0 2 9 9 】

< 一般遊技状態中に用いる各種データテーブル >

続いて、図 4 0 ~ 図 4 5 を参照して、一般遊技状態中に行われる遊技性に関する抽籤処理で用いられる各種データテーブルについて説明する。なお、以下に説明する各種データテーブルは、メイン R O M 1 0 2 に格納される。

【 0 3 0 0 】

また、以下に示す各種データテーブルでは、抽籤値の情報を概念的に示す。データテーブル中の「 0 」は、当籤確率「 0 % 」に相当する抽籤値が規定されていることを意味し、「極々低」は、当籤確率「 0 % ~ 1 % 未満」に相当する抽籤値が規定されていることを意味し、「極低」は、当籤確率「 1 % ~ 1 0 % 未満」に相当する抽籤値が規定されていることを意味する。また、データテーブル中の「低」は、当籤確率「 1 0 % ~ 3 0 % 未満」に相当する抽籤値が規定されていることを意味し、「中」は、当籤確率「 3 0 % ~ 6 0 % 未満」に相当する抽籤値が規定されていることを意味し、「高」は、当籤確率「 6 0 % ~ 8 0 % 未満」に相当する抽籤値が規定されていることを意味する。さらに、データテーブル中の「極高」は、当籤確率「 8 0 % ~ 9 9 % 未満」に相当する抽籤値が規定されていることを意味し、「極々高」は、当籤確率「 9 9 % ~ 1 0 0 % 未満」に相当する抽籤値が規定されていることを意味し、「確定」は、当籤確率「 1 0 0 % 」に相当する抽籤値が規定されていることを意味する。

30

【 0 3 0 1 】

そして、以下に示す各種データテーブルでは、乱数回路 1 1 0 の乱数レジスタ 1 により、予め定められた数値の範囲 (0 ~ 6 5 5 3 5) から抽出される抽籤用乱数値を、規定された抽籤値で順次減算し、減算の結果が負となったか否か (いわゆる「桁かり」が生じたか否か) の判定を行うことによって内部的な抽籤が行われる。なお、本実施形態では、一般遊技状態中に行われる遊技性に関する抽籤処理において抽籤用乱数値から抽籤値を減算して当籤 / 非当籤を判定する例を説明したが、本発明はこれに限定されず、抽出した抽籤用乱数値に抽籤値を加算し、加算結果が 6 5 5 3 6 を超えたか否か (いわゆる「桁あふれ」が生じたか否か) を判定して、当籤 / 非当籤を決定してもよい。

40

【 0 3 0 2 】

[通常中高確率抽籤テーブル]

まず、図 4 0 A 及び 4 0 B を参照して、C Z の抽籤状態 (低確率及び高確率) の移行抽籤

50

で用いられる通常中高確率抽籤テーブルについて説明する。なお、本実施形態のパチスロ1では、毎ゲーム、内部当籤役に基づいてC Zの抽籤状態の移行抽籤が行われるだけでなく、例えばボーナス終了時やC Z, A R T終了時などの場合にもC Zの抽籤状態の移行抽籤が行われる。図40Aは、通常遊技状態中に毎ゲーム参照される通常中高確率抽籤テーブルの構成図であり、図40Bは、例えば設定変更時、ボーナス終了時又はC Z, A R T終了時等に参照される通常中高確率抽籤テーブルの構成図である。なお、図40Aに示す内部当籤役の名称は、上述したサブフラグの名称に対応する。

【0303】

図40Aに示す通常中高確率抽籤テーブルは、現在のC Zの抽籤状態と内部当籤役との各組合せと、移行後のC Zの抽籤状態の抽籤結果（低確率/高確率）と、各抽籤結果に対応付けられた抽籤値の情報との対応関係を規定する。

10

【0304】

図40Aに示す通常中高確率抽籤テーブルから明らかなように、現在のC Zの抽籤状態が低確率である場合には、内部当籤役がサブフラグ「弱チェリー」に対応する役であるときに、C Zの抽籤状態が高確率に移行し易くなる。一方、現在のC Zの抽籤状態が高確率である場合には、内部当籤役がサブフラグ「共通ベル」、「サポテン」、「弱チェリー」及び「強チェリー」のいずれかに対応する役であるときに、C Zの抽籤状態が高確率に維持される。

【0305】

図40Bに示す通常中高確率抽籤テーブルは、該テーブルを参照する際の各状況と、移行後のC Zの抽籤状態の抽籤結果（低確率/高確率）と、各抽籤結果に対応付けられた抽籤値の情報との対応関係を規定する。図40Bに示す通常中高確率抽籤テーブルから明らかなように、ボーナス終了時にはC Zの抽籤状態が必ず高確率に移行する。

20

【0306】

[C Z抽籤テーブル]

次に、図41A及び41Bを参照して、C Z抽籤で用いられるC Z抽籤テーブルについて説明する。図41Aは、通常遊技状態中に内部当籤役に基づいてC Z抽籤を行う際に用いられるC Z抽籤テーブルの構成図であり、図41Bは、例えばC Z失敗時やA R T終了時などにおいて、C Zの引き戻しを行うか否かのC Z抽籤を行う際に用いられるC Z抽籤テーブルの構成図である。なお、図41Aに示す内部当籤役の名称は、上述したサブフラグの名称に対応する。

30

【0307】

図41Aに示すC Z抽籤テーブルは、現在のC Zの抽籤状態と内部当籤役との各組合せと、C Z 1、C Z 2、C Z 3の当籤/非当籤（抽籤結果）と、各抽籤結果に対応付けられた抽籤値の情報との対応関係を規定する。図41Aに示すC Z抽籤テーブルから明らかなように、現在のC Zの抽籤状態が高確率中である場合には、現在のC Zの抽籤状態が低確率中である場合よりも、C Z抽籤に当籤する確率が高くなる。

【0308】

図41Bに示すC Z抽籤テーブルは、C Z失敗時やA R T終了時における、C Z 1、C Z 2、C Z 3の当籤/非当籤（抽籤結果）と、各抽籤結果に対応付けられた抽籤値の情報との対応関係を規定する。C Z失敗時（C Z 1、C Z 2中のA R T抽籤に非当籤時）やA R T遊技状態の終了時には、このC Z抽籤テーブルを用いてC Zの引き戻し抽籤が行われる。

40

【0309】

[C Z 1中モードアップ抽籤テーブル]

次に、図42を参照して、C Z 1の前半部において行われるC Z 1のモードアップ抽籤で用いられるC Z 1中モードアップ抽籤テーブルについて説明する。図42は、C Z 1中モードアップ抽籤テーブルの構成図である。なお、図42に示す内部当籤役の名称は、上述したサブフラグの名称に対応する。

【0310】

C Z 1中モードアップ抽籤テーブルは、現在のモードと内部当籤役との各組合せと、モー

50

ドアップ抽籤の結果（当籤／非当籤）と、各抽籤結果に対応付けられた抽籤値の情報との対応関係を規定する。後述の図 4 4 A に示すように、C Z 1 では、モードが上がる（モードの値が高くなる）ほど A R T 抽籤に当籤する確率が上がり、モードがモード 6 まで上がると、A R T 抽籤に必ず当籤する。

【 0 3 1 1 】

なお、図 4 2 中の抽籤結果「モード 1 U P」とは、C Z 1 のモードが 1 段階上がることを意味し、抽籤結果「モード 2 U P」とは、C Z 1 のモードが 2 段階上がることを意味する。それゆえ、例えば、現在のモードがモード 2 である状況において、抽籤結果「モード 2 U P」に当籤すると、C Z 1 のモードはモード 2 からモード 4 に上がる。また、例えば、抽籤結果「モード 6 U P__フリーズ発生」に当籤すると、フリーズが発生し、A R T 抽籤の当籤及び C T の付与が決定される。

10

【 0 3 1 2 】

[C Z 2 中ポイント抽籤テーブル]

次に、図 4 3 を参照して、C Z 2 の前半部において行われる C Z 2 のポイントアップ抽籤で用いられる C Z 2 中ポイント抽籤テーブルについて説明する。図 4 3 は、C Z 2 中ポイント抽籤テーブルの構成図である。なお、図 4 3 に示す内部当籤役の名称は、上述したサブフラグの名称に対応する。

【 0 3 1 3 】

C Z 2 中ポイント抽籤テーブルは、現在のポイントと内部当籤役との各組合せと、ポイントアップ抽籤の結果（当籤／非当籤）と、各抽籤結果に対応付けられた抽籤値の情報との対応関係を規定する。後述の図 4 4 B に示すように、C Z 2 では、ポイントが上がるほど A R T 抽籤に当籤する確率が上がり、ポイントが「ポイント 1 0」まで上がると、A R T 抽籤に必ず当籤する。なお、図 4 3 中の抽籤結果「ポイント 2 U P」とは、現在の C Z 2 のポイントに「2」が加算されることを意味し、例えば、現在のポイントが「2」である状況において、抽籤結果「ポイント 2 U P」に当籤すると、C Z 2 のポイントは「2」から「4」に上がる。また、例えば、抽籤結果の「ポイント 1 0 U P__フリーズ発生」に当籤すると、フリーズが発生し、A R T 抽籤の当籤及び C T の付与が決定される。

20

【 0 3 1 4 】

[C Z 中 A R T 抽籤テーブル]

次に、図 4 4 A ~ 4 4 C 及び図 4 5 を参照して、C Z 中に実行される A R T 抽籤で用いられる C Z 中 A R T 抽籤テーブルについて説明する。なお、図 4 4 A は、C Z 1 の後半部の 1 ゲーム目で用いられる C Z 中 A R T 抽籤テーブル（C Z 1 用）の構成図であり、図 4 4 B は、C Z 2 の後半部の 1 ゲーム目で用いられる C Z 中 A R T 抽籤テーブル（C Z 2 用）の構成図であり、図 4 4 C は、C Z 1 , C Z 2 の後半部で用いられる C Z 中 A R T 抽籤テーブル（C Z 1 , C Z 2 共通 後半バトル中用）の構成図である。また、図 4 5 は、C Z 3 中に実行される A R T 抽籤で用いられる C Z 中 A R T 抽籤テーブル（C Z 3 用）の構成図である。なお、図 4 4 C 及び図 4 5 に示す内部当籤役の名称は、上述したサブフラグの名称に対応する。

30

【 0 3 1 5 】

図 4 4 A に示す C Z 中 A R T 抽籤テーブル（C Z 1 用）は、現在のモードと、A R T 抽籤の結果（当籤の有無）と、各抽籤結果に対応付けられた抽籤値の情報との対応関係を規定する。また、図 4 4 B に示す C Z 中 A R T 抽籤テーブル（C Z 2 用）は、現在のポイントと、A R T 抽籤の結果（当籤の有無）と、各抽籤結果に対応付けられた抽籤値の情報との対応関係を規定する。

40

【 0 3 1 6 】

C Z 中 A R T 抽籤テーブル（C Z 1 用）及び C Z 中 A R T 抽籤テーブル（C Z 2 用）から明らかのように、C Z 1 及び C Z 2 では前半部のランク（モード又はポイント）が上がるほど、A R T 抽籤に当籤し易くなる。

【 0 3 1 7 】

図 4 4 C に示す C Z 中 A R T 抽籤テーブル（C Z 1 , C Z 2 共通 後半バトル中用）は、内

50

部当籤役と、ART抽籤の結果（当籤の有無）と、各抽籤結果に対応付けられた抽籤値の情報との対応関係を規定する。CZ中ART抽籤テーブル（CZ1，CZ2共通後半バトル中用）から明らかのように、CZ1及びCZ2の後半部において、レア役（サブフラグ「弱チェリー」、「サボテン」又は「強チェリー」に対応する役）が内部当籤役として決定されると、所定の確率でART抽籤に当籤する。

【0318】

図45に示すCZ中ART抽籤テーブル（CZ3用）は、CZ3の消化ゲーム数と内部当籤役との各組合せと、ART抽籤の結果（当籤の有無）と、各抽籤結果に対応付けられた抽籤値の情報との対応関係を規定する。CZ中ART抽籤テーブル（CZ3用）から明らかのように、本実施形態では、CZ3中においてART抽籤に当籤すると必ずCTにも当籤する。

10

【0319】

<通常ART中の遊技性>

次に、図46A及び46Bを参照して、遊技ART中の遊技の流れについて説明する。本実施形態のパチスロ1では、上述のように、ART遊技状態として、通常ARTとCTとが設けられ（図14A及び14B参照）、CT中を上乗せチャンスゾーンとしている。それゆえ、本実施形態では、遊技者は、通常ART中の遊技において、CTへの移行を目指して遊技を行うことになる。

【0320】

[通常ARTからCTへの移行態様]

図46Aは、通常ARTからCTへの遊技状態の移行態様を示す図である。本実施形態のパチスロ1では、図46Aに示すように、通常ART中に行われるCT抽籤に当籤した場合、遊技状態が通常ARTからCTに移行する。なお、本実施形態のパチスロ1は、図46Aに示すように、通常ART中に行われる様々な抽籤に影響を与えるパラメータとして、ARTレベル及びCT抽籤状態が設けられる。

20

【0321】

ARTレベルとしては、レベル1～レベル4の4段階のレベルが設けられ、このARTレベルは、主に通常ART中の継続（消化）ゲーム数に基づいて制御（決定）される。そして、ARTレベルは、CT抽籤状態の決定や後述する通常ART中のフラグ変換抽籤などに対して影響を与える。

30

【0322】

CT抽籤状態としては、低確率、通常、高確率及び超高確率の4段階の状態が設けられ、CT抽籤状態は、主に、ARTレベルや通常ART中の内部当籤役などに基づいて制御（決定）される。そして、CT抽籤状態は、通常ART中に行うCT抽籤や後述する通常ART中のフラグ変換抽籤などに対して影響を与える。

【0323】

[通常ART中のフラグ変換]

上述のように、本実施形態のパチスロ1では、RT4状態中、すなわち、ART遊技状態中に、内部当籤役「F__確チリリブ」、「F__1確チリリブ」及び「F__リーチ目リブA」～「F__リーチ目リブD」のいずれかが単独で内部当籤役として決定されると、フラグ変換抽籤を行い、その抽籤結果に応じて特別な特典（例えば、ARTゲーム数の上乗せやCT当籤）を付与する。図46Bは、通常ART中に行われるフラグ変換抽籤の手法の概要を示す図である。

40

【0324】

本実施形態では、図46Bに示すように、通常ART中において、ARTレベル及びCT抽籤状態を参照して、フラグ変換抽籤が行われる。その結果、フラグ変換抽籤に当籤した場合には、特別な特典を付与するとともに、略称「3連チリリブ」に係る図柄組合せや略称「リーチ目リブ」に係る図柄組合せなどを有効ライン上に停止表示させるためのナビ（例えば、順押しで所定の図柄を狙わせる旨の情報の報知）が行われる。一方、フラグ変換抽籤に非当籤であった場合には、略称「リプレイ」に係る図柄組合せを有効ライン上に停

50

止表示させるためのナビ（例えば、順押し以外の押し順の報知）が行われる。

【0325】

そして、遊技者がこの報知（ナビ）に従い停止操作を行うと、報知内容に応じた図柄組合せが有効ライン上に停止表示される。具体的には、フラグ変換抽籤に当籤した場合には略称「3連チリリブ」に係る図柄組合せや略称「リーチ目リブ」に係る図柄組合せなどが有効ライン上に停止表示され、フラグ変換抽籤に非当籤であった場合には略称「リブレイ」に係る図柄組合せが有効ライン上に停止表示される。

【0326】

<通常ART中に用いる各種データテーブル>

次に、図47～図51を参照して、通常ART中の抽籤処理で用いられる各種データテーブルについて説明する。なお、以下に説明する各種データテーブルは、メインROM102に格納される。

10

【0327】

[ART中フラグ変換抽籤テーブル]

図47A及び47Bは、通常ART中に行われるフラグ変換抽籤で用いられるART中フラグ変換抽籤テーブルの構成図である。

【0328】

本実施形態に係るパチスロ1では、通常ART中のフラグ変換抽籤を2段階で行う。具体的には、内部当籤役「F__確チリリブ」又は「F__1確チリリブ」が当籤した場合、まず、1段階目のフラグ変換抽籤が行われ、この1段階目のフラグ変換抽籤に当籤すると、その後、2段階目のフラグ変換抽籤が行われる。そして、この2段階目のフラグ変換抽籤に当籤すると、内部当籤役「F__確チリリブ」又は「F__1確チリリブ」がサブフラグEX「3連チリリブ」に変換される。一方、1段階目のフラグ変換抽籤又は2段階目のフラグ変換抽籤が非当籤であった場合には、内部当籤役「F__確チリリブ」又は「F__1確チリリブ」はサブフラグEX「リブレイ」に変換される（通常のリブレイ役として扱う）。なお、内部当籤役「F__リーチ目リブA」～「F__リーチ目リブD」のいずれかが当籤した場合には、2段階目のフラグ変換抽籤のみが行われる。

20

【0329】

図47Aは、1段階目のフラグ変換抽籤で用いられるART中フラグ変換抽籤テーブルの構成図であり、図47Bは、2段階目のフラグ変換抽籤で用いられるART中フラグ変換抽籤テーブルの構成図である。

30

【0330】

図47Aに示すART中フラグ変換抽籤テーブルは、内部当籤役（「F__確チリリブ」又は「F__1確チリリブ」）と、1段階目のフラグ変換抽籤の抽籤結果（変換無し/変換有り（仮））と、各抽籤結果に対応付けられた抽籤値の情報との対応関係を規定する。

【0331】

図47Bに示すART中フラグ変換抽籤テーブルは、内部当籤役とARTレベルとCT抽籤状態との各組合せと、2段階目のフラグ変換抽籤の抽籤結果（変換無し/変換有り）と、各抽籤結果に対応付けられた抽籤値の情報との対応関係を規定する。なお、通常ARTにおいて1回、CTに当籤するまでの遊技では、図47B中の項目「ARTレベル」の「初回（一度CTに当籤するまで）」欄のテーブルが参照される。

40

【0332】

本実施形態において、図47A及び47Bに示すように、ART中フラグ変換抽籤テーブルのそれぞれを用いた段階目及び2段階目のフラグ変換抽籤では、確率分母が「256」となる乱数値（0～255）を用いて抽籤が行われる。それゆえ、本実施形態では、上述した2段階のフラグ変換抽籤は、確率分母が「65536」となる乱数値を用いて一回抽籤を行う場合と実質同一の抽籤であるとみなすことができる。

【0333】

近年のパチスロでは、従来、副制御基板72側（以下、「サブ側」という）で行っていた出玉に関する抽籤（ART抽籤など）を主制御基板71側（以下、「メイン側」という）

50

で行うことが求められている。しかしながら、メイン側の記憶手段（メインROM102）の容量が小容量に制限されているため、処理容量の増加を抑えつつ遊技性を損なうことのない抽籤を可能にする仕組みが求められている。

【0334】

この点に関して、本実施形態のパチスロ1では、確率分母が「256」となる抽籤を2段階で行うことにより、確率分母が「65536」となる抽籤を行うことができるので、抽籤処理に係るメイン側の容量の増加を抑えることができる。また、2段階目の抽籤では、ARTレベルやCT抽籤状態などを参照するので、内部当籤役だけでなく現在の状態に応じたフラグ変換抽籤を行うことができ、その結果、多様な遊技性を持ったフラグ変換抽籤を行うことができる。

10

【0335】

[ARTレベル決定テーブル]

図48A及び48Bは、ARTレベルを決定する際に用いられるARTレベル決定テーブルの構成図である。なお、ARTレベルの決定処理は、ART遊技状態への移行が決まったART当籤時、及び、通常ART中に行われる。図48Aは、ART当籤時に用いられるARTレベル決定テーブルの構成図であり、図48Bは、通常ART中に用いられるARTレベル決定テーブルの構成図である。

【0336】

図48Aに示すARTレベル決定テーブルは、ARTレベル1～4（抽籤結果）と、各ARTレベルに対応付けられた抽籤値の情報との対応関係を規定する。なお、本実施形態では、ART当籤時にフリーズが発生している場合には、ARTレベルとしてARTレベル2が決定される。

20

【0337】

図48Bに示すARTレベル決定テーブルは、現在のARTレベルと通常ARTの経過（消化）ゲーム数との各組合せと、移行先の各種ARTレベルと、移行先の各ARTレベルに対応付けられた抽籤値の情報との対応関係を規定する。また、図48Bに示すARTレベル決定テーブルは、現在のARTレベルとCT突入時の通常ARTの経過ゲーム数との各組合せと、移行先の各種ARTレベルと、移行先の各ARTレベルに対応付けられた抽籤値の情報との対応関係を規定する。すなわち、通常ART中では、通常ARTの経過（消化）ゲーム数が所定ゲーム数に到達したタイミングでARTレベルが移行可能となるだけでなく、通常ART中においてCTに突入したタイミングにおいてもARTレベルが移行可能となる。

30

【0338】

[通常ART中高確率抽籤テーブル]

図49は、通常ART中においてCT抽籤状態を決定する際に用いられる通常ART中高確率抽籤テーブルの構成図である。

【0339】

通常ART中高確率抽籤テーブルは、現在のCT抽籤状態と内部当籤役との各組合せと、移行先の各種CT抽籤状態と、各CT抽籤状態に対応付けられた抽籤値の情報との対応関係を規定する。なお、図49に示す内部当籤役の名称は、上述したサブフラグの名称に対応する。

40

【0340】

通常ART中高確率抽籤テーブルから明らかなように、サブフラグ「3連チリリプ（3連チリリプA及び3連チリリプB）」やサブフラグ「リーチ目リプ（リーチ目リプ1～4）」に対応する内部当籤役が当籤している場合、CT抽籤状態が「低確率」に移行（転落）し易くなる。ただし、後述の図50に示すように、サブフラグ「3連チリリプ」や「リーチ目リプ」に対応する内部当籤役が当籤している場合には、CT抽籤状態が転落してしても、CT抽籤に必ず当籤する構成になっている。

【0341】

[ART中CT抽籤テーブル]

50

図50は、通常ART中に行われるCT抽籤で用いられるART中CT抽籤テーブルの構成図である。

【0342】

ART中CT抽籤テーブルは、現在のCT抽籤状態と内部当籤役との各組合せと、CT抽籤の各種抽籤結果（非当籤/通常CT/高確率CT）と、各抽籤結果に対応付けられた抽籤値の情報との対応関係を規定する。なお、図50に示す内部当籤役の名称は、上述したサブフラグの名称に対応する。

【0343】

本実施形態において、内部当籤役として、サブフラグ「サポテン」、「弱チェリー」、「強チェリー」、「3連チリリブ（3連チリリブA及び3連チリリブB）」、「リーチ目リブ（リーチ目リブ1～4）」又は「BB」に対応する役が決定されている場合、ART中CT抽籤テーブルを用いたCT抽籤処理では、確率分母が「256」となる範囲の乱数値を用いたCT抽籤が行われる。また、内部当籤役としてこれらの役以外の内部当籤役（例えば、サブフラグ「リプレイ」、「共通ベル」、「押し順ベル」などに対応する役）が決定されている場合には、ART中CT抽籤テーブルを用いたCT抽籤処理において、確率分母が「65536」となる範囲の乱数値を用いたCT抽籤が行われる。

10

【0344】

なお、本実施形態のパチスロ1では、CTとして「通常CT」及び「高確率CT」と称する2種類のCTを設ける。通常CTと高確率CTとでは、CT（上乘せチャンスゾーン）中に上乘せされるARTゲーム数の期待度が互いに異なり、高確率CTは、通常CTに比べて多くのARTゲーム数が上乘せされ易いCTである（後述の図55参照）。

20

【0345】

[通常ART中上乘せ抽籤テーブル]

図51は、通常ART中に行われるARTゲーム数の上乘せ抽籤で用いられる通常ART中上乘せ抽籤テーブルの構成図である。なお、図51に示す内部当籤役の名称は、上述したサブフラグの名称に対応する。

【0346】

通常ART中上乘せ抽籤テーブルは、内部当籤役と、上乘せ抽籤の各種抽籤結果（非当籤/上乘せ10G～300G）と、各抽籤結果に対応付けられた抽籤値の情報との対応関係を規定する。

30

【0347】

<CT中の遊技性>

次に、図52A～52Cを参照して、CT中の遊技の流れについて説明する。なお、図52A及び52Bは、主に、サブフラグEX「3連チリリブ」当籤時におけるCT中の遊技フローの概要を示す図であり、図52Cは、CT中に行われるフラグ変換処理の概要を示す図である。

【0348】

[CT中の遊技内容]

本実施形態のパチスロ1において、CTでは、1セット8回（8ゲーム）の遊技が行われる。CT期間中には、毎ゲーム、内部当籤役に基づいて、ARTゲーム数の上乘せ抽籤が行われる。そして、その上乘せ抽籤に当籤した場合には、CT遊技の単位遊技数（ゲーム数）の減算は行われず、一方、上乘せ抽籤が非当籤であった場合には、CT遊技の単位遊技数（ゲーム数）の減算が行われる。それゆえ、CT期間中において、ARTゲーム数が上乘せされた遊技では、CTが終了することなく、同一のセット内でARTゲーム数が上乘せされない遊技が8回実施されると、CTが終了する。

40

【0349】

また、本実施形態では、図52A及び52Bに示すように、CT期間中にサブフラグEX「3連チリリブ」が当籤した場合、すなわち、内部当籤役「F__確チリリブ」又は「F__1確チリリブ」が当籤し、かつ、フラグ変換抽籤に当籤した場合、1セット8回のCT遊技が再セット（ストック）される。そして、この再セット（ストック）されたCT遊技の

50

セットは、当該 C T 遊技のセットが終了した後に開始される。

【 0 3 5 0 】

例えば、同一セット内で A R T ゲーム数の上乗せ抽籤に非当籤である単位遊技が 7 回行われた後、A R T ゲーム数が上乗せされない C T 遊技が 1 回行われると C T が終了するが、このゲームにおいてサブフラグ E X 「 3 連チリリブ 」 が当籤していると、C T 遊技の再セットが行われる。その結果、C T 遊技が再セットされた後、A R T ゲーム数の上乗せ抽籤に非当籤である単位遊技が 8 回行われるまで C T が終了しないことになる。それゆえ、C T の遊技期間は、サブフラグ E X 「 3 連チリリブ 」 が当籤するほど長くなる。

【 0 3 5 1 】

[C T 中のフラグ変換]

次に、図 5 2 C を参照して、C T 中に行われるフラグ変換抽籤の手法について説明する。上述のように、本実施形態では、C T 期間中にサブフラグ E X 「 3 連チリリブ 」 が当籤すると（内部当籤役「 F __ 確チリリブ 」又は「 F __ 1 確チリリブ 」が当籤し、かつ、フラグ変換抽籤に当籤すると）、C T が再セット（ストック）される。また、後述の図 5 4 の C T 中フラグ変換抽籤テーブルに示すように、C T 中に内部当籤役「 F __ 確チリリブ 」又は「 F __ 1 確チリリブ 」が当籤すると、フラグ変換抽籤に必ず当籤する（サブフラグ E X 「 3 連チリリブ 」に必ず変換される）。すなわち、本実施形態では、C T 中において、内部当籤役「 F __ 確チリリブ 」又は「 F __ 1 確チリリブ 」が当籤すると、C T が必ず再セットされる。

【 0 3 5 2 】

また、内部当籤役「 F __ リーチ目リブ A 」～「 F __ リーチ目リブ D 」のいずれかが当籤した場合におけるフラグ変換抽籤では、3 種類のフラグ変換テーブル（テーブル 0 ～ 2 ）に基づいて、フラグ変換抽籤の当籤確率が制御される。具体的には、図 5 2 C に示すように、テーブル 0 はサブフラグ E X 「 リーチ目リブ 」に変換される確率が最も低いフラグ変換テーブルであり、テーブル 1 はサブフラグ E X 「 リーチ目リブ 」に変換される確率が次に低いフラグ変換テーブルであり、テーブル 2 はサブフラグ E X 「 リーチ目リブ 」に変換される確率が最も高いフラグ変換テーブルである。なお、C T 中にサブフラグ E X 「 リーチ目リブ 」に当籤すると、後述の図 5 6 の C T 中セット数上乗せ抽籤テーブルに示すように、C T が新たに付与される。

【 0 3 5 3 】

また、本実施形態において、通常 C T では、図 5 2 C に示すように、A R T レベルに基づいてフラグ変換テーブルが決定される。一方、高確率 C T では、A R T レベルに関係なく、フラグ変換テーブルとして、テーブル 0 が必ず決定される。

【 0 3 5 4 】

< C T 中に用いる各種データテーブル >

次に、図 5 3 ～ 図 5 6 を参照して、C T 中に行われる抽籤処理で用いられる各種データテーブルについて説明する。なお、以下に説明する各種データテーブルは、メイン R O M 1 0 2 に格納される。

【 0 3 5 5 】

[C T 中テーブル抽籤テーブル]

図 5 3 は、3 段階のフラグ変換テーブル（テーブル 0 ～ 2 ）の中からフラグ変換抽籤に用いるテーブルを決定する際に用いられる C T 中テーブル抽籤テーブルの構成図である。

【 0 3 5 6 】

C T 中テーブル抽籤テーブルは、A R T レベルやこれから実行する C T の種別などの各状態と、フラグ変換テーブル（テーブル 0 ～ 2 ）の種別と、各種別に対応付けられた抽籤値の情報との対応関係を規定する。なお、C T 中テーブル抽籤テーブルは、C T 抽籤に当籤して C T に移行することが決定された時、又は、C T の開始時に参照される。

【 0 3 5 7 】

[C T 中フラグ変換抽籤テーブル]

図 5 4 は、C T 中に行われるフラグ変換抽籤で用いられる C T 中フラグ変換抽籤テーブル

10

20

30

40

50

の構成図である。

【0358】

CT中フラグ変換抽籤テーブルは、内部当籤役（「F__確チリリブ」、「F__1確チリリブ」及び「F__リーチ目リブA」～「F__リーチ目リブD」のいずれか）と、各フラグ変換テーブル（テーブル0～2）におけるフラグ変換抽籤の抽籤結果（変換無し/変換有り）と、各抽籤結果に対応付けられた抽籤値の情報との対応関係を規定する。CT中フラグ変換抽籤テーブルから明らかなように、CT中に内部当籤役「F__確チリリブ」又は「F__1確チリリブ」が当籤すると、フラグ変換抽籤に必ず当籤する（サブフラグEX「3連チリリブ」に必ず変換される）。

【0359】

[CT中上乘せ抽籤テーブル]

図55は、CT中に行われるARTゲーム数の上乘せ抽籤で用いられるCT中上乘せ抽籤テーブルの構成図である。

【0360】

CT中上乘せ抽籤テーブルは、現在のCT状態と内部当籤役との各組合せと、上乘せ抽籤の各種抽籤結果（非当籤/上乘せ10ゲーム/.../上乘せ300ゲーム）と、各抽籤結果に対応付けられた抽籤値の情報との対応関係を規定する。なお、図55に示す内部当籤役の名称は、上述したサブフラグの名称に対応し、図55に示す内部当籤役（サブフラグ）以外の役が内部当籤した場合には、CT中の上乗せ抽籤に当籤することはない。

【0361】

また、本実施形態の通常CT中における上乘せ抽籤では、サブフラグ「3連チリリブ」（内部当籤役「F__確チリリブ」又は「F__1確チリリブ」）の当籤回数に応じて上乘せゲーム数の付与形態が変化する。

【0362】

具体的には、同一のCTセット中におけるサブフラグ「3連チリリブ」の当籤回数が1～8回である場合には、図55中に示す抽籤結果「上乘せ__10G」及び「上乘せ__20G」で付与される上乘せゲーム数はそれぞれ10ゲーム及び20ゲームとなる。それゆえ、この場合には、ARTの上乗せゲーム数として10ゲームが決定され易くなる。また、同一のCTセット中におけるサブフラグ「3連チリリブ」の当籤回数が9～16回である場合には、図55中に示す抽籤結果「上乘せ__10G」及び「上乘せ__20G」で付与される上乘せゲーム数はともに20ゲームとなる。すなわち、図55中のサブフラグ「3連チリリブ」の抽籤値「極高」に対応する上乘せゲーム数（抽籤結果）が20ゲームに昇格する。それゆえ、この場合には、ARTの上乗せゲーム数として20ゲームが決定され易くなる。

【0363】

また、同一のCTセット中におけるサブフラグ「3連チリリブ」の当籤回数が17～24回である場合には、図55中に示す抽籤結果「上乘せ__10G」及び「上乘せ__20G」で付与される上乘せゲーム数はともに30ゲームとなる。すなわち、図55中のサブフラグ「3連チリリブ」の抽籤値「極高」及び「極低」に対応する上乘せゲーム数（抽籤結果）が30ゲームに昇格する。それゆえ、この場合には、ARTの上乗せゲーム数として「30ゲーム」が決定され易くなる。さらに、同一のCTセット中におけるサブフラグ「3連チリリブ」の当籤回数が25回以上である場合には、図55中に示す抽籤結果「上乘せ__10G」及び「上乘せ__20G」で付与される上乘せゲーム数はともに50ゲームとなる。すなわち、図55中のサブフラグ「3連チリリブ」の抽籤値「極高」及び「極低」に対応する上乘せゲーム数（抽籤結果）が50ゲームに昇格する。それゆえ、この場合には、ARTの上乗せゲーム数として「50ゲーム」が決定され易くなる。

【0364】

上述のように、本実施形態のパチスロ1では、CT中のサブフラグ「3連チリリブ」の当籤回数に応じて1回の上乗せ抽籤により上乘せできるARTゲーム数を増やすことが可能になる。また、上述のように、本実施形態では、ARTゲーム数の上乘せが行われている

10

20

30

40

50

限り、CTは終了することなく、さらに、サブフラグEX「3連チリリブ」に当籤するとCTの再セット（ストック）が行われる。それゆえ、本実施形態では、遊技者に対して、CTが継続するほど、1ゲーム当りの上乗せ量の増加に対する期待を抱かせることができ、CT中の興趣を向上させることができる。また、1ゲーム当りの上乗せ量を増やす契機となるサブフラグ「3連チリリブ」の当籤回数は、CT1セット分の基本遊技回数（8回）よりも多い回数（9回以上）であるため、遊技者に対して過大な利益を与えてしまうことを防止でき、遊技者及び遊技店間において、利益のバランスをとることができる。

【0365】

なお、本実施形態では、上述したサブフラグ「3連チリリブ」（内部当籤役「F__確チリリブ」又は「F__1確チリリブ」）の当籤回数は、同一のCTセット中において計数された回数とするが、本発明はこれに限定されない。例えば、CT中に行われるセット数上乗せ抽籤に当籤した場合に付与される新たなCTも「同一のCTセット中」に含めるようにしてもよい。

10

【0366】

[CT中セット数上乗せ抽籤テーブル]

図56は、CT中に行われるCTセットの上乗せ抽籤で用いられるCT中セット数上乗せ抽籤テーブルの構成図である。

【0367】

CT中セット数上乗せ抽籤テーブルは、現在のCT状態と内部当籤役（サブフラグ「リーチ目リブ（リーチ目リブ1~4）」）との各組合せと、上乗せ抽籤の各種抽籤結果（非当籤/通常CT当籤/高確率CT当籤）と、各抽籤結果に対応付けられた抽籤値の情報との対応関係を規定する。

20

【0368】

CT中セット数上乗せ抽籤テーブルから明らかなように、CT中に内部当籤役「F__リーチ目リブA」~「F__リーチ目リブD」のいずれかに当籤すると、CTセットの上乗せ抽籤に必ず当籤する（CTのセットが必ずストックされる）。なお、ストックされたCTのセットは、現在作動中のCTのセットが終了した後に開始される。

【0369】

<ボーナス状態中の遊技性>

次に、図57A~57Cを参照して、ボーナス状態中の遊技の流れについて説明する。図57Aは、一般遊技状態（ART非当籤）中に遊技状態がボーナス状態に移行した場合における遊技の流れを示す図であり、図57Bは、通常ART中に遊技状態がボーナス状態に移行した場合における遊技の流れを示す図であり、図57Cは、CT中に遊技状態がボーナス状態に移行した場合における遊技の流れを示す図である。

30

【0370】

なお、本実施形態のパチスロ1では、図57A~57Cに示すように、遊技性の面において、ボーナスの種別として通常BBと特殊BBとを設け、ボーナス状態への移行時にこのボーナスの種別が決定される。この際、特殊BBが決定された場合には、ボーナス状態の終了後、遊技状態はART準備状態を経由してCTに移行する。一方、通常BBが決定された場合には、移行先の遊技状態は、ボーナス状態に移行する前の状態に応じて異なる。

40

【0371】

遊技状態が一般遊技状態から通常BBに移行した場合、図57Aに示すように、通常BB中の遊技では、内部当籤役に基づいてART抽籤が行われる。そして、このART抽籤に当籤すると、ボーナス状態の終了後、遊技状態がART準備状態を経由して通常ARTに移行する。なお、この場合、ART抽籤に当籤した後のボーナス状態中の遊技では、ARTゲーム数の上乗せ抽籤が行われる。

【0372】

遊技状態が通常ARTから通常BBに移行した場合、図57Bに示すように、通常BBの終了時にCT抽籤が行われる。このCT抽籤の当籤確率は50%であり、当籤するとボーナス状態の終了後に、遊技状態はART準備状態を経由してCTに移行する。一方、CT

50

抽籤に非当籤である場合には、ボーナス状態の終了後に、遊技状態はART準備状態を経由して通常ARTに移行する。なお、通常ARTから移行したボーナス状態中の遊技では、ARTゲーム数の上乗せ抽籤も行われる。

【0373】

遊技状態がCTから通常BB又は特殊BBに移行した場合、図57Cに示すように、ボーナス状態の終了後に、遊技状態はART準備状態を経由してCTに移行する。なお、CTから移行したボーナス状態中の遊技では、ARTゲーム数の上乗せ抽籤も行われる。

【0374】

<ボーナス状態中の遊技で用いる各種データテーブル>

続いて、図58～図60を参照して、ボーナス状態中の遊技で行われる抽籤処理で用いられる各種データテーブルについて説明する。なお、以下に説明する各種データテーブルは、メインROM102に格納される。

【0375】

[ボーナス種別抽籤テーブル]

図58は、ボーナス種別(通常BB、特殊BB)を決定する際に用いられるボーナス種別抽籤テーブルの構成図である。

【0376】

ボーナス種別抽籤テーブルは、ボーナス状態に移行する前の各遊技状態(CT及びそれ以外)と、各種抽籤結果(ボーナス種別:通常BB/特殊BB)と、各抽籤結果に対応付けられた抽籤値の情報との対応関係を規定する。なお、ボーナス種別抽籤テーブルを参照したボーナス種別の決定処理は、ボーナス状態の開始時に行われる。

【0377】

[ボーナス中ARTゲーム数上乗せ抽籤テーブル]

図59は、ボーナス状態中の遊技で行われるART抽籤及びARTゲーム数の上乗せ抽籤で用いられるボーナス中ARTゲーム数上乗せ抽籤テーブルの構成図である。

【0378】

ボーナス中ARTゲーム数上乗せ抽籤テーブルは、現在のボーナス種別と内部当籤役との各組合せと、上乗せ抽籤の各種抽籤結果(非当籤/5ゲーム/.../300ゲーム)と、各抽籤結果に対応付けられた抽籤値の情報との対応関係を規定する。

【0379】

本実施形態では、ART非当籤の状態(一般遊技状態から移行した通常BBにおいて、ART抽籤に当籤するまでの状態)では、ボーナス中ARTゲーム数上乗せ抽籤テーブルは、ART抽籤に用いられる。具体的には、ART非当籤の状態において、ボーナス中ARTゲーム数上乗せ抽籤テーブルを用いた抽籤により1ゲーム以上(図59に示す例では50ゲーム以上)の上乗せゲーム数が決定されると、ART抽籤に当籤するとともに、対応するゲーム数がARTゲーム数として付与される。一方、ART当籤後の状態では、ボーナス中ARTゲーム数上乗せ抽籤テーブルは、ARTゲーム数の上乗せ抽籤のみに用いられる。

【0380】

[ボーナス終了時CT抽籤テーブル]

図60は、ボーナス状態の終了時に行われるCT抽籤で用いられるボーナス終了時CT抽籤テーブルの構成図である。

【0381】

ボーナス終了時CT抽籤テーブルは、ボーナス種別(通常BB、特殊BB)とボーナス状態に移行する前の遊技状態(通常CT中、高確率CT中)との各組合せと、CT抽籤の各種抽籤結果(非当籤/通常CT当籤/高確率CT当籤)と、各抽籤結果に対応付けられた抽籤値の情報との対応関係を規定する。ボーナス終了時CT抽籤テーブルから明らかのように、例えば、通常ART中に通常BBが行われると、ボーナス状態の終了時に50%の確率でCTに当籤する。

【0382】

10

20

30

40

50

< 一般遊技状態中の例外的な遊技性 >

次に、図 6 1 を参照して、一般遊技状態中の例外的な遊技の流れについて説明する。

【 0 3 8 3 】

本実施形態のパチスロ 1 における基本的な遊技状態の流れでは、一般遊技状態中に遊技状態が通常遊技状態から C Z に移行し、C Z において A R T 抽籤に当籤することにより遊技状態が A R T 遊技状態に移行する。そして、本実施形態では、R T 4 状態において報知を行うことにより A R T 遊技状態を実現している。また、本実施形態では、図 6 1 に示すように、停止表示される図柄組合せに応じて R T 状態の移行制御を行う。

【 0 3 8 4 】

なお、R T 状態を移行させるための図柄組合せは、遊技者の停止操作の順序（押し順）に応じて停止表示されるもの（図 2 4 参照）であるので、報知が行われない場合であっても偶然、R T 状態が R T 4 状態に移行することもある。また、R T 4 状態では、内部当籤役「F__リーチ目リブ A」～「F__リーチ目リブ D」のいずれかが決定される可能性があるため、一般遊技状態（非 A R T）中であっても、特別な特典が付与されるリーチ目（略称「リーチ目リブ」に係る図柄組合せ）を表示することができる。

【 0 3 8 5 】

そこで、本実施形態のパチスロ 1 では、図 6 1 に示すように、一般遊技状態（非 A R T）中に偶然、R T 状態が R T 4 状態に移行し、略称「リーチ目リブ」に係る図柄組合せが表示可能な状態になると、C Z を経由することなく、遊技状態を A R T 遊技状態（通常 A R T）に移行可能にする。

【 0 3 8 6 】

より具体的には、一般遊技状態で、かつ、R T 4 状態中に内部当籤役「F__リーチ目リブ A」～「F__リーチ目リブ D」のいずれかが決定されると、フラグ変換抽籤を行い、このフラグ変換抽籤に当籤すると、略称「リーチ目リブ」に係る図柄組合せを表示するための報知（ナビ）が行われるとともに、A R T の権利が付与される。一方、一般遊技状態で、かつ、R T 4 状態中に内部当籤役「F__リーチ目リブ A」～「F__リーチ目リブ D」のいずれかが決定された場合において、フラグ変換抽籤に非当籤となると、略称「リプレイ」に係る図柄組合せを表示するための報知が行われ、略称「リーチ目リブ」に係る図柄組合せが表示されないような制御が行われる。

【 0 3 8 7 】

< 一般遊技状態中の例外的な遊技制御で用いる各種データテーブル >

次に、図 6 2 を参照して、上述した一般遊技状態中の例外的な遊技制御で行われる抽籤処理で用いるデータテーブルについて説明する。なお、以下に説明するデータテーブルは、メイン R O M 1 0 2 に格納される。

【 0 3 8 8 】

[非 A R T 中フラグ変換抽籤テーブル]

図 6 2 は、一般遊技状態で、かつ、R T 4 状態中の遊技で行われるフラグ変換抽籤で用いられる非 A R T 中フラグ変換抽籤テーブルの構成図である。

【 0 3 8 9 】

非 A R T 中フラグ変換抽籤テーブルは、内部当籤役（「F__リーチ目リブ A」～「F__リーチ目リブ D」）と、フラグ変換抽籤の抽籤結果（変換無し / 変換あり）と、各抽籤結果に対応付けられた抽籤値の情報との対応関係を規定する。

【 0 3 9 0 】

< メイン側の制御による報知機能 >

従来のパチスロでは、A R T 中にサブ（副制御基板 7 2）側の制御により、リールの停止操作の情報（押し順など）の報知（ナビ）を行っていた。しかしながら、この報知の有無が遊技者の利益（いわゆる、出玉）に影響を与えるため、近年では、遊技者の利益を管理するメイン（主制御基板 7 1）側で報知を行うことが求められている。そこで、本実施形態のパチスロ 1 では、上述のように、メイン側で制御される情報表示器 6 に停止操作の情報を報知するための指示モニタ（不図示）を設け、メイン側の制御により、リールの停止

10

20

30

40

50

操作の情報を報知する機能が設けられている。

【0391】

ここで、図63A～63Dに、本実施形態のパチスロ1において、メイン側で行う報知(ナビ)とサブ側で行う報知(ナビ)との対応関係を示す。なお、図63Aは、ART準備状態における、メイン側で行う報知(ナビ)とサブ側で行う報知(ナビ)との対応関係を示す図であり、図63Bは、ART(通常ART又はCT)中における、メイン側で行う報知(ナビ)とサブ側で行う報知(ナビ)との対応関係を示す図である。また、図63Cは、RT5状態中(BB1フラグ間)における、メイン側で行う報知(ナビ)とサブ側で行う報知(ナビ)との対応関係を示す図であり、図63Dは、RT5状態中(BB2フラグ間)における、メイン側で行う報知(ナビ)とサブ側で行う報知(ナビ)との対応関係を示す図である。

10

【0392】

本実施形態では、図63A～63Dに示すように、メイン(主制御基板71)側では、指示モニタに「1」～「11」の数値を表示することにより、リールの停止操作の情報を報知する。なお、指示モニタに表示されるこの「1」～「11」の数値は、それぞれが報知する停止操作の内容に一義的に対応している。

【0393】

具体的には、数値「1」～「3」はそれぞれ、第1停止操作を行うリールの種別を示しており、数値「1」は第1停止操作を左リール3Lに対して行うことを意味し、数値「2」は第1停止操作を中リール3Cに対して行うことを意味し、数値「3」は第1停止操作を右リール3Rに対して行うことを意味する。

20

【0394】

また、数値「4」～「9」はそれぞれ、報知する押し順を示しており、数値「4」は押し順が「左、中、右」の順であることを意味し、数値「5」は押し順が「左、右、中」の順であることを意味し、数値「6」は押し順が「中、左、右」の順であることを意味し、数値「7」は押し順が「中、右、左」の順であることを意味し、数値「8」は押し順が「右、左、中」の順であることを意味し、数値「9」は押し順が「右、中、左」の順であることを意味する。

【0395】

また、数値「10」及び「11」はそれぞれ、ボーナス役を報知するものであり、数値「10」は、コンビネーション名称「C__BB1」に係る図柄組合せ(図柄「白7」-図柄「白7」-図柄「白7」)を意味し、数値「11」は、コンビネーション名称「C__BB2」に係る図柄組合せ(図柄「青7」-図柄「青7」-図柄「青7」)を意味する。

30

【0396】

なお、メイン側(指示モニタ)で報知する数値「1」～「11」は、報知する停止操作の内容に一義的に対応しているものの、全ての遊技者が、その数値に基づいて、明確に報知内容を把握できるとは限らない。例えば、メイン側で指示モニタに数値「6」を表示しただけでは、遊技者によっては報知内容を把握できない可能性もある。

【0397】

そこで、本実施形態のパチスロ1では、メイン側の報知と併せてサブ側でもストップボタンの停止操作に係る情報を報知する。具体的には、サブ側で制御される表示措置11(プロジェクタ機構211及び表示ユニット212)を用いて、サブ側の制御により停止操作に係る情報の報知を行う。

40

【0398】

例えば、第1停止操作を左リール3Lに対して行う押し順を報知する場合、メイン側で指示モニタに数値「1」を表示するとともに、サブ側では、表示装置11の表示画面内の左リール3Lの上方に数値「1」と表示し、左リール3Lが第1停止操作の対象であることを報知する。また、押し順「中、左、右」を報知する場合、メイン側で指示モニタに数値「6」を表示するとともに、サブ側では、表示装置11の表示画面内の中リール3Cの上方に数値「1」を表示し、左リール3Lの上方に数値「2」を表示し、右リール3Rの上

50

方に数値「3」を表示し、この表示により押し順が「中、左、右」の順であることを報知する。また、内部当籤役「F__BB1」が決定されている場合、メイン側で指示モニタに数値「10」を表示するとともに、サブ側では、表示装置11の表示画面に「白7」 - 「白7」 - 「白7」の図柄組合せに関する情報を表示し、遊技者に対して狙うべき図柄を報知する。

【0399】

なお、メイン側で報知を行うタイミングは、少なくとも報知を行う一遊技の期間であれば任意のタイミングに設定することができる。例えば、遊技者の開始操作を検知した（受け付けた）タイミングでメイン側の報知を行ってもよいし、リールの回転開始時にメイン側の報知を行ってもよいし、第1停止操作～第3停止操作のいずれかを検知したタイミングでメイン側の報知を行ってもよい。一方、サブ側で報知を行うタイミングは、少なくとも第1停止操作よりも前のタイミングであることが好ましい。それゆえ、本実施形態のパチスロ1では、開始操作を検知したタイミング、又は、リールの回転開始時のタイミングで、メイン側及びサブ側の双方において報知（ナビ）を行う。これにより、遊技者が停止操作を行う前に、メイン側の指示モニタ及びサブ側の表示装置11の双方において停止操作の情報が報知される。

【0400】

ART準備状態では、図63Aに示すように、メイン側の制御により、「ベルナビ」、「維持リブナビ」、「RT3移行リブナビ」及び「RT4移行リブナビ」と称する報知（ナビ）が行われる。「ベルナビ」では、内部当籤役「F__3択ベル__1st」～「F__3択ベル__3rd」が決定された際に、略称「ベル」に係る図柄組合せ（図29参照）を有効ライン上に停止表示させるための押し順が報知される。「維持リブナビ」では、内部当籤役「F__維持リブ__1st」～「F__維持リブ__3rd」が決定された際に、略称「リブレイ」に係る図柄組合せ（図28参照）を有効ライン上に停止表示させるための押し順が報知される。「RT3移行リブナビ」では、内部当籤役「F__RT3移行リブ__1st」～「F__RT3移行リブ__3rd」が決定された際に、略称「RT3移行リブ」に係る図柄組合せ（図28参照）を有効ライン上に停止表示させるための押し順が報知される。また、「RT4移行リブナビ」では、内部当籤役「F__RT4移行リブ__123」～「F__RT4移行リブ__3rd」が決定された際に、略称「RT4移行リブ」に係る図柄組合せ（図28参照）を有効ライン上に停止表示させるための押し順が報知される。

【0401】

また、ART遊技状態（通常ART又はCT）中では、図63Bに示すように、メイン側の制御により、「ベルナビ」、「維持リブナビ」、「RT3移行リブナビ」及び「RT4移行リブナビ」と称する報知（ナビ）が行われる。なお、ART遊技状態（RT4状態）中の遊技は、フラグ変換抽籤が行われ、この抽籤結果に基づいて略称「3連チリリブ」、「リーチ目リブ」又は「リブレイ」に係る図柄組合せを表示させるための押し順が報知されるが、この報知は、サブ側のみで行われ、メイン側では行われない。

【0402】

上述のように、略称「3連チリリブ」又は「リーチ目リブ」に係る図柄組合せは、特別な特典の付与に関係しているため、報知の有無が遊技者の利益（出玉）に影響を与えるように見えるが、実際には、本実施形態のパチスロ1では、特別な特典は、フラグ変換抽籤の抽籤結果に基づいて付与されるものであるため、表示される図柄組合せは付与する特典に対して影響を与えない。それゆえ、例えば、フラグ変換抽籤に当籤している状態において、仮に、略称「リブレイ」に係る図柄組合せが停止表示されてしまっても、特別な特典が付与される。一方、フラグ変換抽籤に当籤していない状態において、仮に、略称「3連チリリブ」又は「リーチ目リブ」に係る図柄組合せを停止表示できたとしても、特別な特典は付与されない。本実施形態のパチスロ1では、このように表示される図柄組合せが遊技者の利益（出玉）に影響を与えない場合、メイン側の指示モニタでの報知を行わずに、サブ側で制御される表示装置11でのみ報知を行う。

【0403】

10

20

30

40

50

また、RT5状態(フラグ間状態)中では、図63C及び63Dに示すように、内部当籤役として持ち越されているボーナス役に係る図柄組合せを遊技者に狙わせる旨の情報が報知される。例えば、内部当籤役「F__BB1」が持ち越されている場合には、図63Cに示すように、メイン側の制御により、「白7ナビ」と称する報知(ナビ)が行われ、内部当籤役「F__BB2」が持ち越されている場合には、図63Dに示すように、メイン側の制御により、「青7ナビ」と称する報知(ナビ)が行われる。

【0404】

「白7ナビ」では、内部当籤役「F__BB1」に対応する図柄組合せ、すなわち、コンビネーション名称「C__BB1」に係る図柄組合せ(「白7」-「白7」-「白7」:図28参照)を有効ライン上に停止表示させるための停止操作の情報が報知される。また、「青7ナビ」では、内部当籤役「F__BB2」に対応する図柄組合せ、すなわち、コンビネーション名称「C__BB2」に係る図柄組合せ(「青7」-「青7」-「青7」:図28参照)を有効ライン上に停止表示させるための停止操作の情報が報知される。

10

【0405】

フラグ間状態において、ボーナス役と、内部当籤役「はずれ」、「F__特殊1」、「F__特殊2」及び「F__特殊3」のいずれかが決定されている場合、図24で説明したように、ボーナス役(BB役)に係る図柄組合せを有効ライン上に停止表示することができる。しかしながら、内部当籤役「はずれ」、「F__特殊1」、「F__特殊2」及び「F__特殊3」以外の内部当籤役とボーナス役とが当籤している場合には、ボーナス役に係る図柄組合せを有効ライン上に停止表示することができない。それゆえ、本実施形態では、図63C及び63Dに示すように、持ち越されているボーナス役と、内部当籤役「はずれ」、「F__特殊役1」、「F__特殊役2」及び「F__特殊役3」のいずれかが当籤としている場合に限り、メイン側の制御による「白7ナビ」又は「青7ナビ」と称する報知が行われる。それゆえ、本実施形態では、メイン側の制御による「白7ナビ」又は「青7ナビ」と称する報知(ナビ)を、ボーナス役を入賞させることのできる適切なタイミングで行うことができる。

20

【0406】

なお、本実施形態のパチスロ1には、例えば、ボーナス確定画面を表示することや、ボーナス確定ランプを点灯させることなどにより、ボーナス告知を行う機能も設けられている。そこで、メイン側では、ボーナス役が内部当籤役として決定されていることを告知(ボーナス告知)した後にのみ、「白7ナビ」又は「青7ナビ」と称するナビを行うようにしてもよい。

30

【0407】

ボーナス告知としては、例えば、複数回の遊技期間に渡って行われる演出(いわゆる連続演出)を行い、この連続演出の結果に応じてボーナス確定画面を表示するような演出が一般的に行われている。このような連続演出の最中にメイン側で「白7ナビ」などを行うと、連続演出の結果が途中で分かってしまうので、興趣を損ねてしまう可能性がある。そこで、本実施形態では、主制御基板71は、ボーナス告知が行われた後に、メイン側の制御による「白7ナビ」又は「青7ナビ」と称する報知(ナビ)が行われる。

【0408】

なお、ボーナス告知が行われたタイミングをメイン側で把握可能にする手法は任意である。その一手法として、ボーナス役が内部当籤役として決定されると、主制御基板71がボーナス告知終了までに要するゲーム数を決定し、このゲーム数の遊技を消化した後に、「白7ナビ」又は「青7ナビ」と称する報知(ナビ)を行う手法が考えられる。より具体的には、主制御基板71は、ボーナス告知終了までに要するゲーム数を決定すると、このゲーム数を副制御基板72に通知する。副制御基板72は、このゲーム数に従い演出の制御を行い、該ゲーム数の遊技が消化されたタイミングでボーナス確定画面を表示することにより、メイン側においてボーナス告知が行われたタイミングを把握することができる。すなわち、主制御基板71は、ボーナス役が持ち越されていない状態でボーナス役を内部当籤役として決定してからの単位遊技の回数を計数し、その計数結果が所定回数に達した後

40

50

、ボーナス役と、内部当籤役「はずれ」、「F__特殊役1」、「F__特殊役2」及び「F__特殊役3」のいずれかが内部当籤役として決定された場合に、「白7ナビ」又は「青7ナビ」を行う。

【0409】

また、他の手法としては、ボーナス告知をサブ側ではなくメイン側において制御する手法が考えられる。より具体的には、主制御基板71は、ボーナス役が持ち越されていない状態でボーナス役を内部当籤役として決定すると、表示装置11で実行する演出（少なくとも演出に要するゲーム数）を決定し、副制御基板72に通知する。副制御基板72が通知された演出を実行し、ボーナス確定画面を表示することにより、メイン側においてボーナス告知が行われたタイミングを把握することができる。

10

【0410】

なお上述した2つの手法以外の他の手法によりボーナス告知が行われたタイミングをメイン側で把握可能にする構成にしてもよい。この場合、主制御基板71は、副制御基板72などからの信号を受け付けることができないため、主制御基板71が受け付け可能な信号に基づいてボーナス告知が行われたタイミングを把握する必要がある。例えば、停止操作に伴う信号は、主制御基板71が受け付け可能であるため、ボーナス役が内部当籤役として決定されている状態で、所定の停止操作（例えば、順押し以外）が行われた場合に、ボーナス告知を行う手法も考えられる。具体的には、副制御基板72は、主制御基板71から内部当籤役に関する情報と停止操作に関する情報とを取得し、これらの情報の組合せが所定の組合せである場合にボーナス告知を行う。このようなボーナス告知の手法を採用することにより、ボーナス告知の契機を主制御基板71でも把握することができるため、メイン側においてボーナス告知が行われたタイミングを把握することができる。

20

【0411】

<主制御回路の動作説明>

次に、図64～図170を参照して、主制御回路90のメインCPU101が、プログラムを用いて実行する各種処理の内容について説明する。

【0412】

[電源投入（リセット割込）時処理]

まず、メインCPU101の制御により行われるパチスロ1の電源投入（リセット割込）時処理を、図64及び図65を参照して説明する。図64は、電源投入（リセット割込）時処理の手順を示すフローチャートであり、図65A～65Cは、それぞれ、該フローチャート中のS2、S7及びS8、並びに、S13の処理を実行するためのソースプログラムの一例を示す図である。なお、図64に示す電源投入（リセット割込）時処理は、電源管理回路93が、マイクロプロセッサ91に電源電圧の供給が開始されたことを検知した際に、リセット信号をマイクロプロセッサ91の「XSRST」端子に出力し、それにより、マイクロプロセッサ91の割込みコントローラ112からメインCPU101に出力される割込要求信号に基づいて、実行される。

30

【0413】

まず、メインCPU101は、電源監視ポート（電源監視手段）がオン状態であるか否かを判別する（S1）。

40

【0414】

S1において、メインCPU101が、電源監視ポートがオン状態であると判別したとき（S1がYES判定の場合）、メインCPU101は、S1の処理を繰り返す。なお、ここでいう、電源監視ポートがオン状態とは、メインCPU101に供給されている電源電圧（DC+5V）が安定していない状態のことである。

【0415】

一方、S1において、メインCPU101が、電源監視ポートがオン状態でないと判別したとき（S1がNO判定の場合）、メインCPU101は、タイマー回路113（PTC）の初期化処理を行う（S2）。この処理では、メインCPU101は、タイマー回路113の初期設定を行う。具体的には、メインCPU101は、タイマー用プリスケラレ

50

ジスタ（不図示）に分周比をセットし、タイマー用制御レジスタ（不図示）に割り込み可等の設定を行い、タイマー用カウンタ（不図示）の初期カウント値を設定する。

【0416】

次いで、メインCPU101は、主制御回路90及び副制御回路200間用の第1シリアル通信回路114（SCU1）の初期化处理、及び、第2インターフェースボード用の第2シリアル通信回路115（SCU2）の初期化处理を行う（S3）。次いで、メインCPU101は、乱数回路110（RDG）の初期化处理を行う（S4）。次いで、メインCPU101は、メインRAM103の書き込みテストを行う（S5）。

【0417】

次いで、メインCPU101は、書き込みテストの結果、メインRAM103への書き込みが正常に行われたか否かを判別する（S6）。 10

【0418】

S6において、メインCPU101が、メインRAM103への書き込みが正常に行われなかったと判別したとき（S6がNO判定の場合）、メインCPU101は、後述のS13の処理を行う。一方、S6において、メインCPU101が、メインRAM103への書き込みが正常に行われたと判別したとき（S6がYES判定の場合）、メインCPU101は、タイマー回路113のタイマー用制御レジスタ（不図示）の状態を取得する（S7）。

【0419】

次いで、メインCPU101は、取得したタイマー用制御レジスタの状態に基づいて、現在の状態が割込処理の発生タイミングであるか否かを判別する（S8）。具体的には、メインCPU101は、取得したタイマー用制御レジスタの状態に基づいて、タイマーカウンタ開始後から1.1172ms経過したか否かを判別する。 20

【0420】

なお、本実施形態では、S2のタイマー回路113の初期化处理によりタイマー時間1.1172msがセットされると、CPU内蔵タイマーのカウント処理が開始される。その後、タイマー用制御レジスタ（不図示）の情報を読み込むことによりタイマー回路113のステータスを取得することができる。そして、本実施形態では、タイマー用制御レジスタに、現在の状態が割込処理の発生タイミングであるか否か（タイマー割込状態であるか否か）を判別（参照）可能なビット（判別ビット）が設けられる。 30

【0421】

それゆえ、上記S7の処理では、メインCPU101は、タイマー用制御レジスタ（不図示）の情報を読み込み、上記S8の処理では、メインCPU101は、タイマー用制御レジスタ内の判別ビットのオン/オフ状態（「1」/「0」）を参照することにより、現在の状態が割込処理の発生タイミングであるか否かを判別する。なお、タイマー回路113によるカウント開始から1.1172ms経過したとき（タイマー回路113のカウント値が0であれば）、該判別ビットはオン状態となる。

【0422】

S8において、メインCPU101が、現在の状態が割込処理の発生タイミングでないと判別したとき（S8がNO判定の場合）、メインCPU101は、処理をS7の処理に戻し、S7以降の処理を繰り返す。 40

【0423】

一方、S8において、メインCPU101が、現在の状態が割込処理の発生タイミングであると判別したとき（S8がYES判定の場合）、メインCPU101は、サムチェック処理（規定外）を行う（S9）。この処理では、メインCPU101は、メインRAM103のサムチェック処理を行うが、この処理の作業は、メインRAM103内の規定外作業領域（図12C参照）で行われる。また、このサムチェック処理で用いられるプログラムはメインROM102内の規定外エリアに格納されている（図12B参照）。なお、サムチェック処理の詳細については、後述の図79及び図80を参照しながら後で説明する。

【0424】

また、S 8において、メインCPU 101が、現在の状態が割込処理の発生タイミングであると判別したとき（S 8がYES判定の場合）には、メインCPU 101は、S 9の処理の前に、後述の割込処理（後述の図158参照）を実行する。そして、この割込処理により、主制御回路90（主制御基板71）から副制御回路200（副制御基板72）には、無操作コマンドが送信される。

【0425】

S 9の処理後、メインCPU 101は、設定用鍵型スイッチ54がオン状態であるか否かを判別する（S 10）。

【0426】

S 10において、メインCPU 101が、設定用鍵型スイッチ54がオン状態であると判別したとき（S 10がYES判定の場合）、メインCPU 101は、後述のS 15の処理を行う。一方、S 10において、メインCPU 101が、設定用鍵型スイッチ54がオン状態でないと判別したとき（S 10がNO判定の場合）、メインCPU 101は、S 9のサムチェック処理の結果に基づいて、サムチェック判定結果が正常であったか否かを判別する（S 11）。

10

【0427】

S 11において、メインCPU 101が、サムチェック判定結果が正常でないと判別したとき（S 11がNO判定の場合）、メインCPU 101は、後述のS 13の処理を行う。一方、S 11において、メインCPU 101が、サムチェック判定結果が正常であると判別したとき（S 11がYES判定の場合）、メインCPU 101は、遊技復帰処理を行う（S 12）。この処理では、メインCPU 101は、遊技の状態を電断検知前の状態に戻す処理を行う。なお、遊技復帰処理の詳細については、後述の図66を参照しながら後で説明する。

20

【0428】

S 6又はS 11がNO判定の場合、メインCPU 101は、情報表示器6（7セグLED表示器）に、エラー発生を意味する文字列「rr」を表示する（S 13）。その後、メインCPU 101は、WDTのクリア処理を繰り返す（S 14）。

【0429】

ここで再度、S 10の処理に戻って、S 10がYES判定の場合、メインCPU 101は、設定変更確認処理を行う（S 15）。この処理では、メインCPU 101は、主に、設定変更開始時における設定変更コマンドの生成格納処理を行う。なお、設定変更確認処理の詳細については、後述の図68を参照しながら後で説明する。

30

【0430】

次いで、メインCPU 101は、RAM初期化処理を行う（S 16）。この処理では、メインCPU 101は、図12Cに示すメインRAM 103の遊技用RAM領域内の「RAM異常時又は設定変更開始時」のアドレスを、初期化開始の先頭アドレスとして設定し、該先頭アドレスから遊技用RAM領域の最終アドレスまでの情報を消去（クリア）する。そして、S 16の処理後、メインCPU 101は、後述のメイン処理（後述の図82参照）を開始する。

【0431】

本実施形態では、上述のようにして電源投入（リセット割込）時処理が行われる。そして、上述した電源投入（リセット割込）時処理中のS 2の処理は、メインCPU 101が、図65Aのソースプログラムで規定されている各ソースコードを順次実行することにより行われる。

40

【0432】

図65Aのソースプログラムでは、CPUクロックとして10MHz（供給クロックは20MHz）が設定され、プリスケアラレジスタ設定値として228が設定され、初期カウント値として49が設定される。この結果、割り込み処理のタイマー時間（実行周期）として、 $1.1172\text{ms} (= 1 / (10\text{MHz} / 228) \times 49)$ が算出される。

【0433】

50

また、上述した電源投入（リセット割込）時処理中のS 7及びS 8の処理は、メインCPU 101が、図65Bのソースプログラムで規定されている各ソースコードを順次実行することにより行われる。具体的には、S 7のタイマー用制御レジスタの状態の取得処理は、図65B中の「LD」命令により実行され、S 8の判定処理は、図65B中の「JBIT」命令により実行される。そして、タイマー回路113によるカウント開始から1.1172ms経過するまで、図65B中の「LD」命令及び図65B中の「JBIT」命令が繰り返し行われ、タイマー回路113によるカウント開始から1.1172ms経過すると、タイマー回路113から割込みコントローラ112を介してメインCPU 101に割込要求信号が出力される。メインCPU 101は、この割込要求信号の入力を契機として、電源復帰後の最初の1.1172ms周期の割込処理を開始する。

10

【0434】

なお、この電源復帰直後（電源投入時の初期化後）の最初の1.1172ms周期の割込処理では、遊技動作に関するコマンドはセットされていないので、主制御回路90から副制御回路200には無操作コマンドが送信される。このように電源復帰直後に割込処理を許可することにより、電源復帰後、最短時間で無操作コマンドが送信され、主制御回路90及び副制御回路200間の通信接続を確立することができ、主制御回路90及び副制御回路200間の通信動作を安定化させることができる。

【0435】

また、この通信動作で送信される無操作コマンドを構成する通信パラメータ1～5には、電源復帰時に、それぞれLレジスタ、Hレジスタ、Eレジスタ、Dレジスタ及びCレジスタに格納されているデータがセットされる。それゆえ、本実施形態では、電源復帰後の最初の割込処理で送信される無操作コマンドの通信パラメータ1～5にそれぞれセットされるデータを、電源復帰毎に異ならせる（不定にする）ことができる。すなわち、電源復帰直後（電源投入時の初期化後）の割込処理で送信される無操作コマンドのサム値（CCC）を、電源復帰毎に異ならせることができる。この場合、ゴト等の不正行為を抑制することができる。

20

【0436】

また、上述した電源投入（リセット割込）時処理中のS 13の処理（割込み禁止処理）は、メインCPU 101が、図65Cのソースプログラムで規定されている各ソースコードを順次実行することにより行われる。そして、エラーコード「rr」を、情報表示器6内の2桁の7セグLEDに表示する際の制御は、図65Cに示すように、一つのソースコード「LDW HL, 100H * c ZCHRAR + c BX__PAYSEG」により実行され、2桁の7セグLEDへの7セグコモン出力データの出力動作と7セグカソード出力データの出力動作とが同時に行われる。

30

【0437】

すなわち、本実施形態のパチスロ1では、2桁の7セグLEDをダイナミック点灯制御する際に、7セグコモン出力データと、7セグカソード出力データとが同時に出力される。この場合、ソースプログラム上において、7セグLEDのダイナミック点灯制御に必要な命令コード数を減らすことができ、ソースプログラムの容量（メインROM 102の使用容量）を低減することができる。それゆえ、本実施形態では、メインROM 102において、空き容量を確保する（増大させる）ことができ、増えた空き容量を活用して、遊技性を高めることができる。

40

【0438】

なお、ここでいう、「7セグコモン出力データ」は、7セグLEDをダイナミック制御する際に、7セグLEDのコモン（共通）端子に出力されるLED駆動データであり、「7セグカソード出力データ」は、7セグLEDをダイナミック点灯制御する際に、7セグLEDの各カソード端子に出力されるLED駆動データである。

【0439】

[遊技復帰処理]

次に、図66及び図67を参照して、電源投入（リセット割込）時処理（図64参照）中

50

の S 1 2 で行う遊技復帰処理について説明する。なお、図 6 6 は、遊技復帰処理の手順を示すフローチャートであり、図 6 7 は、該フローチャート中の S 2 5 ~ S 3 2 の処理を実行するためのソースプログラムの一例を示す図である。

【 0 4 4 0 】

まず、メイン CPU 1 0 1 は、スタックポインタ (S P) に、電断時のスタックポインタをセットする (S 2 1)。次いで、メイン CPU 1 0 1 は、入力ポートの 1 割込処理前のオンエッジデータ、及び、現在セットされているオンエッジデータをクリア (オフ) する (S 2 2)。次いで、メイン CPU 1 0 1 は、出力ポートのバックアップデータを出力ポートにセットする (S 2 3)。次いで、メイン CPU 1 0 1 は、入力ポートのデータを読み込み、該データを、入力ポートの現在及び 1 割込処理前のデータ格納領域に保存する (S 2 4)。

10

【 0 4 4 1 】

次いで、メイン CPU 1 0 1 は、回胴制御データ格納領域のアドレスをセットする (S 2 5)。次いで、メイン CPU 1 0 1 は、チェックするリール数 (本実施形態では「 3 」) をセットする (S 2 6)。

【 0 4 4 2 】

次いで、メイン CPU 1 0 1 は、セットされた回胴制御データ格納領域のアドレスに基づいて、所定のリールのリール制御管理情報 (電断発生時の表示列の変動制御に関するデータ) を取得する (S 2 7)。なお、リール制御管理情報 (表示列の変動制御管理情報) は、各リールの制御状態 (回転状況) に関する情報であり、電断時には、バックアップされて保存される。

20

【 0 4 4 3 】

次いで、メイン CPU 1 0 1 は、リール制御管理情報がリールの加速中、定速待ち又は定速中の回転状況に対応する情報であるか否かを判別する (S 2 8)。

【 0 4 4 4 】

S 2 8 において、メイン CPU 1 0 1 が、S 2 8 の条件を満たさないと判別したとき (S 2 8 が N O 判定の場合)、メイン CPU 1 0 1 は、後述の S 3 1 の処理を行う。一方、S 2 8 において、メイン CPU 1 0 1 が、S 2 8 の条件を満たすと判別したとき (S 2 8 が Y E S 判定の場合)、メイン CPU 1 0 1 は、回胴制御データ (リール制御管理情報) をクリアする (S 2 9)。この処理により、遊技復帰後、リールの回転制御が加速処理から開始される。次いで、メイン CPU 1 0 1 は、リールの作動タイミング値 (回胴制御データの実行開始タイミング「 1 」) をセットする (S 3 0)。なお、リールの作動タイミングに「 1 」がセットされると、リール制御処理 (後述の図 1 5 8 中の S 9 0 3 参照) 内で、励磁変更タイミングとなるため、メイン CPU 1 0 1 は、リールの回転制御を加速処理から開始する。

30

【 0 4 4 5 】

S 3 0 の処理後又は S 2 8 が N O 判定の場合、メイン CPU 1 0 1 は、リール数の値を 1 減算する (S 3 1)。次いで、メイン CPU 1 0 1 は、減算後のリール数の値が「 0 」であるか否かを判別する (S 3 2)。

【 0 4 4 6 】

S 3 2 において、メイン CPU 1 0 1 が、減算後のリール数の値が「 0 」でないと判別したとき (S 3 2 が N O 判定の場合)、メイン CPU 1 0 1 は、チェック対象のリールを変えて、処理を S 2 7 の処理に戻し、S 2 7 以降の処理を繰り返す。

40

【 0 4 4 7 】

一方、S 3 2 において、メイン CPU 1 0 1 が、減算後のリール数の値が「 0 」であると判別したとき (S 3 2 が Y E S 判定の場合)、メイン CPU 1 0 1 は、R A M 初期化処理を行う (S 3 3)。この処理では、メイン CPU 1 0 1 は、図 1 2 C に示すメイン R A M 1 0 3 の遊技用 R A M 領域内の「電源復帰時」のアドレスを、初期化開始の先頭アドレスとして設定し、該先頭アドレスから遊技用 R A M 領域の最終アドレスまでの情報を消去 (クリア) する。

50

【 0 4 4 8 】

次いで、メインCPU101は、電断検知時に退避させた全てのレジスタのデータを全てのレジスタに復帰させる(S34)。そして、S34の処理後、メインCPU101は、遊技復帰処理を終了し、処理を電断検知時の処理に戻す。

【 0 4 4 9 】

本実施形態では、上述のようにして遊技復帰処理が行われる。本実施形態の遊技復帰処理では、上述のように、電断発生時の各ポートの入出力状態を電源復帰時に担保するとともに、電断時にリール回転中の場合には、電源復帰時にリール制御管理情報を取得してリールの再回転開始に必要な処理も行う(S25～S32の処理参照)。それゆえ、本実施形態では、回胴回転中の電断から復帰したときであっても、安定して、リールの再回転制御を行うことができ、遊技者に不快感を与えることが無くなる。

10

【 0 4 5 0 】

また、上述した遊技復帰処理中のS25～S32の処理は、メインCPU101が、図67のソースプログラムで規定されている各ソースコードを順次実行することにより行われる。なお、上述のように、本実施形態のパチスロ1で用いられている遊技機用セキュリティ機能付きのマイクロプロセッサ91では、メインCPU101専用の各種命令コードが設けられている。例えば、図67中の「LDQ」命令(所定の読み出し命令)は、メインCPU101専用命令コードの一つである。

【 0 4 5 1 】

ソースプログラム上において、例えば、ソースコード「LDQ HL, k」が実行されると、Qレジスタの内容(格納データ)と、1バイトの整数k(直値)とで指定されたアドレスが、HLレジスタにロードされる。この際、Qレジスタの内容が指定先アドレスの上位側のアドレス値となり、整数k(直値)が指定先アドレスの下位側のアドレス値となる。それゆえ、図67中のソースコード「LDQ HL, .LOW.WR1_CTRL」が実行されると、Qレジスタの内容(回胴制御データ格納領域のアドレスの上位側のアドレス値)と、整数値「.LOW.WR1_CTRL」(回胴制御データ格納領域のアドレスの下位側のアドレス値)とで指定されるアドレス(回胴制御データ格納領域のアドレス)が、HLレジスタにロードされる。なお、「.LOW.」は、実際の命令ではなく、擬似命令と呼ばれるものである。この疑似命令の機能では、「.LOW.」に続いて規定される格納領域のアドレスの下位側アドレスのみが有効にされる。また、疑似命令は、実際のROMに格納される命令ではなく、ソースファイルをROMに格納するための形式に変換する際に、変換プログラム(アセンブラ)が参照するための命令である。

20

30

【 0 4 5 2 】

上述のように、本実施形態では、Qレジスタ(拡張レジスタ)を用いてアドレス指定を行うメインCPU101専用命令コードを使用することにより、直値により、メインROM102、メインRAM103やメモリーマップI/Oにアクセスすることができる。この場合、ソースプログラム上において、アドレス設定に係る命令コードを省略することができる。その分、ソースプログラムの容量(メインROM102の使用容量)を低減することができる。この結果、本実施形態では、メインROM102において、空き容量を確保する(増大させる)ことができ、増えた空き容量を活用して、遊技性を高めることが可能になる。

40

【 0 4 5 3 】

[設定変更確認処理]

次に、図68及び図69を参照して、電源投入(リセット割込)時処理(図64参照)中のS15で行う設定変更確認処理について説明する。なお、図68は、設定変更確認処理の手順を示すフローチャートであり、図69Aは、該フローチャート中のS44～S47の処理を実行するためのソースプログラムの一例を示す図であり、図69Bは、該フローチャート中のS57の処理を実行するためのソースプログラムの一例を示す図である。

【 0 4 5 4 】

まず、メインCPU101は、メインRAM103内の規定外RAM領域の初期化処理を

50

行う（S41）。次いで、メインCPU101は、1割り込み待ち処理を行う（S42）。この処理では、メインCPU101は、割込処理による無操作コマンドの副制御回路200への送信処理が終了するまで待機する。

【0455】

次いで、メインCPU101は、RAM初期化処理を行う（S43）。この処理では、メインCPU101は、図12Cに示すメインRAM103の遊技用RAM領域内の「RAM異常時又は設定変更開始時」のアドレスを、初期化開始の先頭アドレスとして設定し、該先頭アドレスから遊技用RAM領域の最終アドレスまでの情報を消去（クリア）する。

【0456】

次いで、メインCPU101は、設定用鍵型スイッチ54がオン状態であるか否かを判別する（S44）。なお、設定用鍵型スイッチ54に差し込まれる設定キー（不図示）は、パチスロ1の設定（設定1～6）を操作するための操作キーであり、設定キーがオンされていると、設定用鍵型スイッチ54がオン状態となる。

10

【0457】

S44において、メインCPU101が、設定用鍵型スイッチ54がオン状態でないと判別したとき（S44がNO判定の場合）、メインCPU101は、設定変更確認処理を終了し、処理を電源投入（リセット割込）時処理（図64参照）のS16の処理に移す。一方、S44において、メインCPU101が、設定用鍵型スイッチ54がオン状態であると判別したとき（S44がYES判定の場合）、メインCPU101は、メダル受付禁止の処理を行う（S45）。この処理により、セレクト66（図7参照）のソレノイドの駆動が行われず、投入されたメダルがメダル払出口24（図2参照）から排出される。

20

【0458】

次いで、メインCPU101は、Lレジスタに設定変更開始又は設定確認開始の情報（005H：第1の値）をセットし、設定変更コマンド（設定変更/設定確認開始）の生成格納処理を行う（S46）。この処理では、メインCPU101は、設定変更処理又は設定確認処理の開始時に主制御回路90から副制御回路200に送信される設定変更コマンドデータ（第1のコマンドデータ）を生成し、該コマンドデータをメインRAM103に設けられた通信データ格納領域に保存する。なお、設定変更コマンド生成格納処理の詳細については、後述の図70を参照しながら後で説明する。また、通信データ格納領域に保存された設定変更コマンド（設定変更/設定確認開始）は、後述の図158で説明する割込処理内の通信データ送信処理により、主制御回路90から副制御回路200に送信される。

30

【0459】

次いで、メインCPU101は、エラーカウントリレーをオン状態にセットする（S47）。次いで、メインCPU101は、設定変更及び設定確認のいずれが行われたかを判別する（S48）。

【0460】

S48において、メインCPU101が、設定変更が行われていない（設定確認が行われた）と判別したとき（S48がNO判定の場合）、メインCPU101は、後述のS55の処理を行う。

【0461】

一方、S48において、メインCPU101が、設定変更が行われた（設定確認が行われていない）と判別したとき（S48がYES判定の場合）、メインCPU101は、設定値の更新処理を行う（S49）。次いで、メインCPU101は、設定値の7セグ表示設定処理を行う（S50）。この処理により、更新後の設定値が情報表示器6内の7セグLEDで表示可能になる。

40

【0462】

次いで、メインCPU101は、リセットスイッチ76がオン状態であるか否かを判別する（S51）。

【0463】

S51において、メインCPU101が、リセットスイッチ76がオン状態であると判別

50

したとき（S 5 1 が Y E S 判定の場合）、メイン C P U 1 0 1 は、処理を S 4 9 の処理に戻し、S 4 9 以降の処理を繰り返す。一方、S 5 1 において、メイン C P U 1 0 1 が、リセットスイッチ 7 6 がオン状態でないと判別したとき（S 5 1 が N O 判定の場合）、メイン C P U 1 0 1 は、スタートスイッチ 7 9 がオン状態であるか否かを判別する（S 5 2）。

【 0 4 6 4 】

S 5 2 において、メイン C P U 1 0 1 が、スタートスイッチ 7 9 がオン状態でないと判別したとき（S 5 2 が N O 判定の場合）、メイン C P U 1 0 1 は、処理を S 5 1 の処理に戻し、S 5 1 以降の処理を繰り返す。一方、S 5 2 において、メイン C P U 1 0 1 が、スタートスイッチ 7 9 がオン状態であると判別したとき（S 5 2 が Y E S 判定の場合）、メイン C P U 1 0 1 は、メイン R A M 1 0 3 に設けられた設定値格納領域（不図示）に設定値を格納する（S 5 3）。

10

【 0 4 6 5 】

次いで、メイン C P U 1 0 1 は、設定用鍵型スイッチ 5 4 がオフ状態であるか否かを判別する（S 5 4）。

【 0 4 6 6 】

S 5 4 において、メイン C P U 1 0 1 が、設定用鍵型スイッチ 5 4 がオフ状態でないと判別したとき（S 5 4 が N O 判定の場合）、メイン C P U 1 0 1 は、S 5 4 の処理を繰り返す。一方、S 5 4 において、メイン C P U 1 0 1 が、設定用鍵型スイッチ 5 4 がオフ状態であると判別したとき（S 5 4 が Y E S 判定の場合）、メイン C P U 1 0 1 は、後述の S 5 5 の処理を行う。

20

【 0 4 6 7 】

S 4 8 が N O 判定の場合又は S 5 4 が Y E S 判定の場合、メイン C P U 1 0 1 は、設定変更及び設定確認のいずれが行われたか否かを判別する（S 5 5）。

【 0 4 6 8 】

S 5 5 において、メイン C P U 1 0 1 が、設定変更が行われていない（設定確認が行われた）と判別したとき（S 5 5 が N O 判定の場合）、メイン C P U 1 0 1 は、後述の S 5 7 の処理を行う。一方、S 5 5 において、メイン C P U 1 0 1 が、設定変更が行われた（設定確認が行われていない）と判別したとき（S 5 5 が Y E S 判定の場合）、メイン C P U 1 0 1 は、R A M 初期化処理を行う（S 5 6）。この処理では、メイン C P U 1 0 1 は、図 1 2 C に示すメイン R A M 1 0 3 の遊技用 R A M 領域内の図示しない「設定変更終了時」のアドレス（設定値格納領域の次のアドレス）を、初期化開始の先頭アドレスとして設定し、該先頭アドレスから遊技用 R A M 領域の最終アドレスまでの情報を消去（クリア）する。

30

【 0 4 6 9 】

S 5 6 の処理後又は S 5 5 が N O 判定の場合、メイン C P U 1 0 1 は、L レジスタに設定変更終了又は設定確認終了の情報（0 0 4 H：第 2 の値）をセットし、設定変更コマンド（設定変更 / 設定確認終了）の生成格納処理を行う（S 5 7）。この処理では、メイン C P U 1 0 1 は、設定変更処理又は設定確認処理の終了時に主制御回路 9 0 から副制御回路 2 0 0 に送信される設定変更コマンドデータ（第 2 のコマンドデータ）を生成し、該コマンドデータをメイン R A M 1 0 3 に設けられた通信データ格納領域に保存する。なお、設定変更コマンド生成格納処理の詳細については、後述の図 7 0 を参照しながら後で説明する。また、通信データ格納領域に保存された設定変更コマンド（設定変更 / 設定確認終了）は、後述の図 1 5 8 で説明する割込処理内の通信データ送信処理により、主制御回路 9 0 から副制御回路 2 0 0 に送信される。そして、S 5 7 の処理後、メイン C P U 1 0 1 は、設定変更確認処理を終了し、処理を電源投入（リセット割込）時処理（図 6 4 参照）の S 1 6 の処理に移す。

40

【 0 4 7 0 】

本実施形態では、上述のようにして、設定変更確認処理が行われる。上述した設定変更確認処理中の S 4 4 ~ S 4 7 の処理は、メイン C P U 1 0 1 が、図 6 9 A のソースプログラムで規定されている各ソースコードを順次実行することにより行われる。その中で、例え

50

ば、S 4 4 の設定キーの状態判定処理は、図 6 9 A 中のソースコード「B I T Q 7 , (. L O W . (w I B U F + 4)) 」により実行され、S 4 7 のエラーカウントリレーをオン状態にセットする処理は、図 6 9 A 中のソースコード「S E T Q 1 , (. L O W . w E C R R E Q) 」により実行される。

【 0 4 7 1 】

「B I T Q」命令及び「S E T Q」命令はともに、Qレジスタ（拡張レジスタ）を用いてアドレス指定を行うメインCPU 1 0 1 専用命令コードである。

【 0 4 7 2 】

ソースプログラム上において、例えば、ソースコード「B I T Q b , (k) 」が実行されると、Qレジスタの格納データ（上位側アドレス値）と、1バイトの整数値k（直値：下位側アドレス値）とで指定されるアドレスのメモリのビットbがチェックされ、該ビットbに「1」が格納されていれば、フラグ・レジスタFのゼロフラグ（ビット6：図11参照）に「0」がセットされ、該ビットbに「0」が格納されていれば、フラグ・レジスタFのゼロフラグ（所定のビット領域）に「1」がセットされる。それゆえ、図69A中のソースコード「B I T Q 7 , (. L O W . (w I B U F + 4)) 」が実行されると、Qレジスタの格納データと、整数値「. L O W . (w I B U F + 4) 」とで指定されるアドレスのメモリのビット7がチェックされ、該ビット7に「1」が格納されていれば、フラグ・レジスタFのゼロフラグに「0」がセットされ、該ビット7に「0」が格納されていれば、フラグ・レジスタFのゼロフラグに「1」がセットされる。

【 0 4 7 3 】

また、ソースプログラム上において、例えば、ソースコード「S E T Q b , (k) 」が実行されると、Qレジスタの格納データ（上位側アドレス値）と、1バイトの整数値k（直値：下位側アドレス値）とで指定されるアドレスのメモリのビットbに「1」がセットされる。それゆえ、図69A中のソースコード「S E T Q 1 , (. L O W . w E C R R E Q) 」が実行されると、Qレジスタの格納データと、整数値「. L O W . w E C R R E Q 」とで指定されるアドレスのメモリのビット1に「1」がセットされる。

【 0 4 7 4 】

すなわち、本実施形態の設定変更確認処理では、上述のようなQレジスタ（拡張レジスタ）を用いた各種メインCPU 1 0 1 専用命令コードが使用されており、これらのメインCPU 1 0 1 専用命令コードの使用により、直値で、メインROM 1 0 2、メインRAM 1 0 3 やメモリーマップ I / O にアクセスすることができる。この場合、アドレス設定に係る命令コードを省略することができ、その分、ソースプログラムの容量（メインROM 1 0 2 の使用容量）を低減することができる。この結果、本実施形態では、メインROM 1 0 2 の空き容量を増やすことが可能となるとともに、処理の高速化も図ることができる。

【 0 4 7 5 】

また、上述した設定変更確認処理中のS 4 6 の設定変更 / 設定確認開始時に行う設定変更コマンド（初期化コマンド）の生成格納処理は、メインCPU 1 0 1 が図69A中の「C A L L F」命令を実行することにより行われ、上述したS 5 7 の設定変更 / 設定確認終了時に行う設定変更コマンド（初期化コマンド）の生成格納処理は、メインCPU 1 0 1 が図69B中の「C A L L F」命令を実行することにより行われる。なお、「C A L L F」命令もまた、メインCPU 1 0 1 専用命令コードである。

【 0 4 7 6 】

ソースプログラム上において、例えば、ソースコード「C A L L F m n」が実行されると、現在のPCレジスタ（プログラム・カウンタPC：図11参照）の値（格納データ）がスタックポインタ（SP）で指定されているメモリに保存され、スタックポインタが-2更新され、「m n」がPCレジスタに格納されて、「m n」で指定されているアドレスに処理がジャンプする。ただし、「C A L L F」命令は、2バイト命令であり、ジャンプできるアドレス範囲は、0 0 0 0 H ~ 1 1 F F H の範囲となる。それゆえ、例えば、図69A中のソースコード「C A L L F S B _ P C I N I T _ 0 0」が実行されると、現在のPCレジスタの値がスタックポインタ（SP）で指定されているメモリに保存され、ス

10

20

30

40

50

タックポインタが - 2 更新され、「S B _ P C I N I T _ 0 0」のアドレスが P C レジスタに格納されて、「S B _ P C I N I T _ 0 0」で指定されているアドレスに処理がジャンプする。

【0477】

なお、本実施形態では、「C A L L F」命令と同種の命令コードとして、「C A L L」命令と呼ばれる命令コードも用意されている。そして、ソースプログラム上において、例えば、ソースコード「C A L L m n」が実行されると、「C A L L F」命令と同様に、現在の P C レジスタ（プログラム・カウンタ P C：図 11 参照）の値（格納データ）がスタックポインタ（S P）で指定されているメモリに保存され、スタックポインタが - 2 更新され、「m n」が P C レジスタに格納されて、「m n」で指定されているアドレスに処理がジャンプする。ただし、「C A L L」命令は、3 バイト命令であり、ジャンプできるアドレス範囲が、「C A L L F」命令のそれと異なり、ジャンプできるアドレス範囲は、0 0 0 0 H ~ F F F F H の範囲である。なお、「C A L L F」命令は、「C A L L」命令に比べてバイト数の少ない命令コードであるので、ソースプログラムの容量（メイン R O M 1 0 2 の使用容量）を低減することができるとともに、処理の効率化も図ることができる。

10

【0478】

また、本実施形態の設定変更確認処理では、図 6 9 A 及び 6 9 B に示すように、S 4 6 の「C A L L F」命令で指定するジャンプ先のアドレス「S B _ P C I N I T _ 0 0」は、S 5 7 の「C A L L F」命令で指定するジャンプ先のアドレスと同じである。すなわち、本実施形態では、設定変更時（遊技機起動時）、設定確認開始時（通常動作中）及び設定確認終了時に副制御回路 2 0 0 に送信する設定変更コマンド（初期化コマンド）の生成格納処理を実行するためのソースプログラムが、互いに同じであり、S 4 6 及び S 5 7 の両処理において用いられる、設定変更コマンド生成格納処理のソースプログラムが共有化（モジュール化）されている。

20

【0479】

この場合、S 4 6 及び S 5 7 の両処理において、それぞれ別個に設定変更コマンド生成格納処理のソースプログラムを設ける必要が無くなるので、その分、ソースプログラムの容量（メイン R O M 1 0 2 の使用容量）を低減することができる。この結果、本実施形態では、メイン R O M 1 0 2 において、空き容量を確保する（増大させる）ことができ、増えた空き容量を活用して、遊技性を高めることが可能になる。

30

【0480】

[設定変更コマンド生成格納処理]

次に、図 7 0 及び図 7 1 を参照して、設定変更確認処理（図 6 8 参照）中の S 4 6 及び S 5 7 で行う設定変更コマンド生成格納処理について説明する。なお、図 7 0 は、設定変更コマンド生成格納処理の手順を示すフローチャートであり、図 7 1 は、設定変更コマンド生成格納処理を実行するためのソースプログラムの一例を示す図である。

【0481】

まず、メイン C P U 1 0 1 は、設定値（1 ~ 6）の情報を E レジスタにセットする（S 6 1）。次いで、メイン C P U 1 0 1 は、R T 状態の情報を C レジスタにセットする（S 6 2）。次いで、メイン C P U 1 0 1 は、設定変更コマンドのコマンド種別情報（0 2 H）を A レジスタにセットする（S 6 3）。

40

【0482】

次いで、メイン C P U 1 0 1 は、通信データ格納処理を行う（S 6 4）。この処理では、メイン C P U 1 0 1 は、S 6 1 ~ S 6 3 で各レジスタにセットされた情報と、S 4 6 又は S 5 7（図 6 8 参照）で D レジスタにセットされた情報（設定ステータスである設定変更開始 / 設定変更終了 / 設定確認開始 / 設定確認終了）とを用いて、設定変更コマンドデータを生成し、該生成されたコマンドデータを通信データ格納領域に保存する。なお、通信データ格納処理の詳細については、後述の図 7 2 を参照しながら後で説明する。

【0483】

S 6 4 の処理後、メイン C P U 1 0 1 は、設定変更コマンド生成格納処理を終了する。な

50

お、設定変更確認処理（図 6 8 参照）中の S 4 6 で行う設定変更コマンド生成格納処理を終了する際には、メイン CPU 1 0 1 は、S 6 4 の処理後、処理を設定変更確認処理（図 6 8 参照）の S 4 7 の処理に移す。また、設定変更確認処理（図 6 8 参照）中の S 5 7 で行う設定変更コマンド生成格納処理を終了する際には、メイン CPU 1 0 1 は、S 6 4 の処理後、設定変更コマンド生成格納処理を終了するとともに、設定変更確認処理（図 6 8 参照）も終了する。

【 0 4 8 4 】

本実施形態では、上述のようにして設定変更コマンド生成格納処理が行われる。なお、上述した設定変更コマンド生成格納処理は、メイン CPU 1 0 1 が、図 7 1 のソースプログラムで規定されている各ソースコードを順次実行することにより行われる。

10

【 0 4 8 5 】

上述のように、設定変更コマンド生成格納処理では、設定変更コマンド生成格納処理が実行される直前に設定ステータスが通信パラメータ 4 として D レジスタに格納され、設定変更コマンド生成格納処理の実行中に設定値が通信パラメータ 3 として E レジスタに格納され、RT 情報が通信パラメータ 5 として C レジスタに格納される。すなわち、設定変更コマンド（初期化コマンド）を構成する通信パラメータ 1 ~ 5 のうち、通信パラメータ 3 ~ 5 は副制御回路 2 0 0 側で使用（解析）される通信パラメータ（使用パラメータ）であり、これらの通信パラメータには新たな情報がセットされる。一方、設定変更コマンド（初期化コマンド）を構成するその他の通信パラメータ 1 及び 2 は、副制御回路 2 0 0 側で使用（解析）されない通信パラメータ（未使用パラメータ）であり、通信パラメータ 1 及び 2 に対しては、現時点で L レジスタ及び H レジスタにそれぞれ格納されている値がセットされる。それゆえ、設定変更コマンド（初期化コマンド）送信時における通信パラメータ 1 及び 2 の値は不定値となる。この場合、設定変更コマンドのサム値（BCC）を送信毎に不定値にすることができ、ゴト等の不正行為を抑制することができる。

20

【 0 4 8 6 】

[通信データ格納処理]

次に、図 7 2 及び図 7 3 を参照して、例えば、設定変更コマンド生成格納処理（図 7 0 参照）中の S 6 4 で行う通信データ格納処理について説明する。なお、通信データ格納処理は、設定変更コマンド生成時だけでなく、他のコマンド生成時にも実行される。図 7 2 は、通信データ格納処理の手順を示すフローチャートであり、図 7 3 は、通信データ格納処理中の S 7 1 ~ S 7 6 の処理を実行するためのソースプログラムの一例を示す図である。

30

【 0 4 8 7 】

まず、メイン CPU 1 0 1 は、A レジスタにセットされているデータを通信コマンド種別のデータとして、メイン RAM 1 0 3 内の通信データ一時格納領域（不図示）に格納する（S 7 1）。次いで、メイン CPU 1 0 1 は、H レジスタ及び L レジスタにセットされているデータを、それぞれ通信コマンドのパラメータ 1 及び 2 として、メイン RAM 1 0 3 内の通信データ一時格納領域（所定の格納領域）に格納する（S 7 2）。

【 0 4 8 8 】

次いで、メイン CPU 1 0 1 は、D レジスタ及び E レジスタにセットされているデータを、それぞれ通信コマンドのパラメータ 3 及び 4 として、メイン RAM 1 0 3 内の通信データ一時格納領域に格納する（S 7 3）。次いで、メイン CPU 1 0 1 は、B レジスタ及び C レジスタにセットされているデータを、それぞれ通信コマンドのパラメータ 5 及び RT 状態のデータとして、メイン RAM 1 0 3 内の通信データ一時格納領域に格納する（S 7 4）。

40

【 0 4 8 9 】

次いで、メイン CPU 1 0 1 は、A レジスタ ~ L レジスタにセットされているデータ値から通信コマンドの BCC データ（サム値）を生成する（S 7 5）。次いで、メイン CPU 1 0 1 は、生成した BCC データをメイン RAM 1 0 3 内の通信データ一時格納領域に格納する（S 7 6）。

【 0 4 9 0 】

50

S 7 6 の処理後、メインCPU 1 0 1 は、メインRAM 1 0 3 内の通信データ格納領域に空きがあるか否かを判別する(S 7 7)。なお、本実施形態では、通信データ格納領域に最大9個のコマンドデータが格納可能である(後述の図7 5 B参照)。

【0 4 9 1】

S 7 7 において、メインCPU 1 0 1 が、通信データ格納領域に空きがないと判別したとき(S 7 7 がNO判定の場合)、メインCPU 1 0 1 は、通信データ格納処理を終了するとともに、例えば、設定変更コマンド生成格納処理(図7 0参照)も終了する。

【0 4 9 2】

一方、S 7 7 において、メインCPU 1 0 1 が、通信データ格納領域に空きがあると判別したとき(S 7 7 がYES判定の場合)、メインCPU 1 0 1 は、上述したS 7 1 ~ S 7 6 の処理により通信データ一時格納領域に格納された通信データを通信コマンドデータとして、通信データ格納領域に格納する(S 7 8)。

10

【0 4 9 3】

次いで、メインCPU 1 0 1 は、通信データポインタ更新処理を行う(S 7 9)。この処理では、メインCPU 1 0 1 は、主に、通信データ格納領域内における通信データの格納アドレスを示す通信データポインタの更新処理を行う。なお、通信データポインタ更新処理の詳細については、後述の図7 4を参照しながら後で説明する。

【0 4 9 4】

そして、S 7 9 の処理後、メインCPU 1 0 1 は、通信データ格納処理を終了するとともに、例えば、設定変更コマンド生成格納処理(図7 0参照)も終了する。

20

【0 4 9 5】

本実施形態では、上述のようにして通信データ格納処理が行われる。なお、上述した通信データ格納処理中のS 7 1 ~ S 7 6 の処理は、メインCPU 1 0 1 が、図7 3のソースプログラムで規定されている各ソースコードを順次実行することにより行われる。そして、この一連の処理において、コマンドデータに含まれる、通信コマンドの種別データ、各種通信パラメータ、遊技状態フラグデータ及びBCCデータの格納処理は、図7 3に示すように、ソースプログラム上では、Qレジスタ(拡張レジスタ)を用いてアドレス指定を行うメインCPU 1 0 1 専用命令コードである、「LDQ」命令を用いて実行される。

【0 4 9 6】

具体的には、ソースコード「LDQ (.LOW.(WPDT__TMP+0)),A」の実行により、Aレジスタに格納された通信コマンドの種別データが、Qレジスタの格納データ(上位側アドレス値)と1バイトの整数値「.LOW.(WPDT__TMP+0)」(下位側アドレス値)とで指定されたアドレスの通信データ一時格納領域に格納される。また、ソースコード「LDQ (.LOW.(WPDT__TMP+1)),HL」の実行により、Lレジスタに格納された通信パラメータ1が、Qレジスタの格納データと1バイトの整数値「.LOW.(WPDT__TMP+1)」とで指定されたアドレスの通信データ一時格納領域に格納され、Hレジスタに格納された通信パラメータ2が、その次のアドレスの通信データ一時格納領域に格納される。また、ソースコード「LDQ (.LOW.(WPDT__TMP+3)),DE」の実行により、Eレジスタに格納された通信パラメータ3が、Qレジスタの格納データと1バイトの整数値「.LOW.(WPDT__TMP+3)」とで指定されたアドレスの通信データ一時格納領域に格納され、Dレジスタに格納された通信パラメータ4が、その次のアドレスの通信データ一時格納領域に格納される。そして、ソースコード「LDQ (.LOW.(WPDT__TMP+5)),BC」の実行により、Cレジスタに格納された通信パラメータ5が、Qレジスタの格納データと1バイトの整数値「.LOW.(WPDT__TMP+5)」とで指定されたアドレスの通信データ一時格納領域に格納され、Bレジスタに格納された遊技状態フラグデータが、その次のアドレスの通信データ一時格納領域に格納される。

30

40

【0 4 9 7】

さらに、通信データ格納処理でセットされたコマンドデータのサム値となるBCCデータは、一連のソースコード「ADD(加算命令コード)A,H」~「ADD A,B」の实

50

行により算出され、Aレジスタに格納される。そして、ソースコード「LDQ (.LOW.(wPDT__TMP+7)),A」の実行により、Aレジスタに格納されたBCCデータが、Qレジスタの格納データと1バイトの整数値「.LOW.(wPDT__TMP+7)」とで指定されたアドレスの通信データ一時格納領域に格納される。

【0498】

上述のように、本実施形態では、1パケット(8バイト)の通信データ(コマンドデータ)を作成する際に、各種パラメータをレジスタから転送して通信データ一時格納領域(通信バッファ)に格納する。このようなコマンドデータの作成手法では、コマンド生成時に各レジスタに格納されているデータがそのままコマンドデータの各種パラメータとして通信データ一時格納領域に格納される。それゆえ、未使用パラメータを含むコマンドデータを作成した時には、作成時毎に、未使用パラメータの値が不定値となる。この場合、同じ種別のコマンドデータあり、かつ、使用パラメータの値が同一であっても、コマンド作成毎に、コマンドデータのサム値(BCCデータ)が可変可能となる。また、本実施形態では、誤り符号の一つであるサム値の計算をADD(加算命令コード)により算出したが、加算命令コードに換えて、SUB(減算命令コード)、XOR(排他的論理和命令コード)により誤り符号を算出しても同様の効果が得られる。さらに、メインCPU101専用命令である、MUL(乗算命令コード)又はDIV(除算命令コード)を使用して誤り符号を算出しても同様の効果が得られる。

10

【0499】

それゆえ、本実施形態では、未使用パラメータを不定値とすることにより、通信データの解析を困難にしてゴト等の不正行為を抑止することができるとともに、不必要なゴト対策処理を加える必要がないため、ゴト対策処理の追加による、主制御回路90のプログラム容量の圧迫を抑制することができる。

20

【0500】

[通信データポインタ更新処理]

次に、図74及び図75を参照して、通信データ格納処理(図72参照)中のS79で行う通信データポインタ更新処理について説明する。なお、図74は、通信データポインタ更新処理の手順を示すフローチャートであり、図75Aは、通信データポインタ更新処理を実行するためのソースプログラムの一例を示す図であり、図75Bは、通信データポインタ更新処理のソースプログラム上で実際にセットされる通信データ格納領域の構成を示す図である。

30

【0501】

まず、メインCPU101は、現在、セットされている通信データポインタの値を取得する(S81)。

【0502】

次いで、メインCPU101は、通信データポインタの値を1パケット分(8バイト)加算更新する(S82)。なお、この処理において、更新後の通信データポインタの値が、通信データ格納領域(図75B参照)の上限サイズ以上となる場合には、メインCPU101は、更新後の通信データポインタの値を「0」にセットし、これにより、通信データ格納領域に格納されているコマンドデータを全て無効にする(破棄した状態と同様の状態にする)。

40

【0503】

本実施形態では、1回の送信動作で送信されるデータ量(1パケット)は8バイトである。すなわち、本実施形態では、一つの送信動作で一つのコマンドデータを送信することができる。また、本実施形態では、通信データ格納領域に最大9個のコマンドデータを格納可能であるので(図75B参照)、通信データ格納領域の上限サイズは、72バイト(=8バイト×9)となる。それゆえ、本実施形態では、通信データポインタの範囲を「0」~「71」とし、S82の処理において、更新後(通信データポインタを+8更新した場合)の通信データポインタの値が「71(上限値)」を超えるような値となる場合には、更新後の通信データポインタの値を「0」にセットして(通信データの格納先のアドレス

50

を先頭アドレスに戻して)、通信データ格納領域に格納されているコマンドデータを全て無効にする(破棄した状態と同様の状態にする)。なお、通信データポインタの値を「0」にセットすると、次にコマンドデータを通信データ格納領域に格納する場合には、通信データ格納領域の先頭アドレスから格納されるので、その前に格納されていたコマンドデータは新たなコマンドデータで上書きされることになる。それゆえ、本実施形態では、通信データポインタの値が「71(上限値)」を超えた場合に、通信データ格納領域を初期化(クリア)する必要はない。

【0504】

そして、S82の処理後、メインCPU101は、通信データポインタ更新処理を終了するとともに、通信データ格納処理(図72参照)も終了する。

10

【0505】

本実施形態では、上述のようにして通信データポインタ更新処理が行われる。そして、上述した通信データポインタ更新処理は、メインCPU101が、図75Aのソースプログラムで規定されている各ソースコードを順次実行することにより行われる。その中で、S82の通信データポインタの更新処理は、図75A中の「ADD」命令及び「ICPLD」命令(所定の更新命令)により実行されるが、この「ICPLD」命令もまた、メインCPU101専用命令コードである。

【0506】

ソースプログラム上において、例えば、ソースコード「ICPLD A, n」が実行されると、Aレジスタの内容(格納データ)と整数nとが比較され、Aレジスタの内容が整数n未満である場合には、Aレジスタの内容に「1」が加算され、Aレジスタの内容が整数n以上である場合には、Aレジスタに「0」がセットされる。

20

【0507】

それゆえ、S82の通信データポインタの更新処理を実行する場合、図75Aのソースプログラム上では、まず、ソースコード「ADD A, 7」が実行され、Aレジスタの内容(更新前の通信データポインタの値)に「7」が加算され、該加算結果がAレジスタに格納される。次いで、ソースコード「ICPLD A, 71」が実行され、Aレジスタの内容(7加算後の通信データポインタの値)と整数「71」とを比較し、Aレジスタの内容が整数「71」未満である場合には、Aレジスタの内容に「1」を加算し、Aレジスタの内容が整数「71」以上である場合には、Aレジスタに「0」をセットする。すなわち、S82の処理において、通信データポインタの値を+7更新したときに、更新後の通信データポインタの値が上限値「71」を超えるような場合には、通信データポインタをゼロクリアする処理(通信データの格納アドレスを通信データ格納領域の先頭アドレスに戻す処理)が行われる。一方、更新後の通信データポインタの値が上限値「71」を超えない場合には、「ICPLD」命令でさらに通信データポインタに「1」を加算することにより、トータルで通信データポインタの値を+8更新する。

30

【0508】

上述のように、本実施形態では、通信データポインタ更新処理において、一つの「ICPLD」命令コード(送信バッファの上限判定命令と、判断分岐命令とが一体になっている命令コード)により、通信データポインタの更新(1加算)処理、更新後の通信データポインタの判定チェック処理及び通信データポインタのクリア処理をまとめて実行することができる。この場合、各処理を別個に実行するための命令コードを設ける必要がなくなる。例えば、更新後の通信データポインタの値がその上限値「71」を超えるか否かの判断分岐命令コードを省略することができる。

40

【0509】

それゆえ、通信データポインタ更新処理等において、メインCPU101専用の「ICPLD」命令コードを用いることにより、ソースプログラムの容量(メインROM102の使用容量)を低減することができる。この結果、本実施形態では、メインROM102において、空き容量を確保する(増大させる)ことができ、増えた空き容量を活用して、遊技性を高めることが可能になる。

50

【 0 5 1 0 】

[電断時 (外部) 処理]

次に、メインCPU101の制御により行われるパチスロ1の電断時(外部)処理を、図76を参照して説明する。図76は、電断時(外部)処理の手順を示すフローチャートである。なお、図76に示す電断時(外部)処理は、電源管理回路93が、マイクロプロセッサ91に供給される電源電圧の低下(電断)を検知した際に、電断検知信号をマイクロプロセッサ91の「XINT」端子に出力し、これにより、マイクロプロセッサ91の割込みコントローラ112からメインCPU101に出力される割込要求信号に基づいて、実行される。

【 0 5 1 1 】

まず、メインCPU101は、全てのレジスタにセットされているデータを退避させる(S91)。次いで、メインCPU101は、電断検知ポートにセットされているデータを読み込む(S92)。

【 0 5 1 2 】

次いで、メインCPU101は、電断検知ポートがオン状態であるか否かを判別する(S93)。

【 0 5 1 3 】

S93において、メインCPU101が、電断検知ポートがオン状態でないと判別したとき(S93がNO判定の場合)、メインCPU101は、割込処理許可をセットする(S94)。そして、S94の処理後、メインCPU101は、電断時(外部)処理を終了する。なお、S93がNO判定である場合に行われるこれらの処理は、電源管理回路93が瞬間的に電断を検知した場合等に発生する瞬停対策の処理に対応する。

【 0 5 1 4 】

一方、S93において、メインCPU101が、電断検知ポートがオン状態であると判別したとき(S93がYES判定の場合)、メインCPU101は、メダル投入不可を設定し、ホッパー装置51の停止を設定する(S95)。

【 0 5 1 5 】

次いで、メインCPU101は、現在セットされているスタックポインタ(SP)の値をメインRAM103内の遊技用RAM領域のスタックエリアに保存する(S96)。

【 0 5 1 6 】

次いで、メインCPU101は、メインRAM103のチェックサム生成処理を行う(S97)。なお、この処理は、メインRAM103内の規定外作業領域(図12C参照)で行われる。また、このチェックサム生成処理で用いられるプログラムはメインROM102内の規定外エリアに格納されている(図12B参照)。なお、チェックサム生成処理の詳細については、後述の図77を参照しながら後で説明する。

【 0 5 1 7 】

次いで、メインCPU101は、メインRAM103へのアクセス禁止を設定する(S98)。そして、S98の処理後、電源が停止するまで(電源電圧が、メインCPU101が動作できない電圧に達するまで)無限ループ処理が行われる。

【 0 5 1 8 】

[チェックサム生成処理 (規定外)]

次に、図77及び図78を参照して、電断時(外部)処理(図76参照)中のS97で行うチェックサム生成処理について説明する。なお、図77は、チェックサム生成処理の手順を示すフローチャートであり、図78Aは、チェックサム生成処理を実行するためのソースプログラムの一例を示す図であり、図78Bは、チェックサム生成処理で実行されるスタックポインタの更新動作及びメインRAM103からレジスタへのデータの読み出し動作の様子を示す図である。

【 0 5 1 9 】

まず、メインCPU101は、現在のスタックポインタ(SP)の値(遊技用RAM領域のスタックエリアの使用アドレス)をメインRAM103の規定外RAM領域の規定外

10

20

30

40

50

スタックエリアに保存する (S101)。次いで、メインCPU101は、スタックポインタに規定外スタックエリアのアドレスをセットする (S102)。次いで、メインCPU101は、RAMアドレス (規定外スタックエリアのアドレス) の上位側のアドレス値 (F0H) をQレジスタにセットする (S103)。次いで、メインCPU101は、電断発生フラグを設定する (S104)。

【0520】

次いで、メインCPU101は、スタックポインタに、遊技用RAM領域内のサム値の計算開始アドレスをセットし、サム算出カウンタに、サム値の算出対象格納領域のバイト数を「2」で除算した値をセットする (S105)。なお、サム算出カウンタは、サム値算出の終了契機を判定するためのカウンタであり、メインRAM103に設けられる。そして、S105で設定されたサム算出カウンタが「0」になれば、メインRAM103の遊技用RAM領域のサム値算出処理を終了する。

10

【0521】

次いで、メインCPU101は、HLレジスタを0クリア (値「0」をセット) する (S106)。この処理により、サム値の初期値「0」がセットされる。

【0522】

次いで、メインCPU101は、「POP命令」 (特定の命令) と呼ばれる命令コード (図78A中に記載のソースコード「POP DE」) を実行し、スタックポインタ (SP) にセットされたメインRAM103の格納領域のアドレスから2バイト分の領域のデータ (保存値) をDEレジスタに読み出す (S107)。

20

【0523】

なお、「POP」命令が実行されると、スタックポインタで指定されたアドレスの1バイト領域に保存されているデータ (メモリ内容) が、ペアレジスタの下位側のレジスタにロードされ、スタックポインタで指定されたアドレスを1更新したアドレスの1バイト領域に保存されているデータ (メモリ内容) が、ペアレジスタの上位側のレジスタにロードされる。また、「POP」命令が実行されると、スタックポインタ (SP) にセットされたアドレスに対して2バイト分のアドレス更新処理 (アドレスを「2」加算する処理) が行われる。

【0524】

それゆえ、S107の処理では、スタックポインタで指定されたアドレスに保存されているデータ (メモリ内容) がEレジスタにロードされ、スタックポインタで指定されたアドレスに「1」を加算したアドレスに保存されているデータ (メモリ内容) がDレジスタにロードされる。

30

【0525】

図78Bには、「POP」命令実行時における、DEレジスタへのデータの読み込み動作、及び、スタックポインタにセットされるアドレスの更新動作の様子を示す。サム値の算出開始時に、スタックポインタ (SP) にセットされているアドレスが「F010h」である場合には、アドレス「F010h」に保存されているデータ (メモリ内容) がEレジスタにロードされ、アドレス「F011h」に保存されているデータ (メモリ内容) がDレジスタにロードされる。また、この際、スタックポインタ (SP) にセットされているアドレスに2加算する更新処理が行われ、スタックポインタ (SP) にセットされているアドレスが「F010h」から「F012h」に変更される。次いで、再度、「POP」命令が実行されると、アドレス「F012h」に保存されているデータ (メモリ内容) がEレジスタにロードされ、アドレス「F013h」に保存されているデータ (メモリ内容) がDレジスタにロードされる。また、この際、スタックポインタ (SP) にセットされているアドレスの更新処理が行われ、スタックポインタ (SP) にセットされているアドレスが「F012h」から「F014h」に変更される。その後、「POP」命令が実行される度に上述した、DEレジスタへのデータの読み込み動作及びスタックポインタにセットされるアドレスの更新動作が繰り返される。

40

【0526】

50

S 1 0 7の処理後、メインCPU 1 0 1は、サム値の算出処理を行う(S 1 0 8)。具体的には、メインCPU 1 0 1は、HLレジスタに格納されている値にDEレジスタに格納されている値を加算し、該加算された値をサム値としてHLレジスタに格納する。

【0527】

次いで、メインCPU 1 0 1は、サム算出カウンタの値を1減算する(S 1 0 9)。次いで、メインCPU 1 0 1は、更新後のサム算出カウンタの値が「0」であるか否かを判別する(S 1 1 0)。

【0528】

S 1 1 0において、メインCPU 1 0 1が、サム算出カウンタの値が「0」でないと判別したとき(S 1 1 0がNO判定の場合)、メインCPU 1 0 1は、処理をS 1 0 7の処理に戻し、S 1 0 7以降の処理を繰り返す。すなわち、メインRAM 1 0 3の遊技用RAM領域のサム値算出処理が終了するまで、S 1 0 7～S 1 1 0の処理が繰り返される。

10

【0529】

一方、S 1 1 0において、メインCPU 1 0 1が、サム算出カウンタの値が「0」であると判別したとき(S 1 1 0がYES判定の場合)、メインCPU 1 0 1は、DEレジスタに、メインRAM 1 0 3内の規定外RAM領域のサム値の計算開始アドレスをセットし、サム算出カウンタに、規定外用サムカウンタ値をセットする(S 1 1 1)。なお、規定外用サムカウンタ値は、規定外用格納領域のバイト数となる。それゆえ、S 1 1 1で設定されたサム算出カウンタが「0」になれば、メインRAM 1 0 3の規定外RAM領域のサム値算出処理、すなわち、メインRAM 1 0 3全体のサム値算出処理が終了する。

20

【0530】

次いで、メインCPU 1 0 1は、DEレジスタにセットされた規定外RAM領域のアドレスから1バイト分の領域のデータ(保存値)をAレジスタに読み出す(S 1 1 2)。

【0531】

次いで、メインCPU 1 0 1は、サム値の算出処理を行う(S 1 1 3)。具体的には、メインCPU 1 0 1は、HLレジスタに格納されている値にAレジスタに格納されている値を加算し、該加算された値をサム値としてHLレジスタに格納する。

【0532】

次いで、メインCPU 1 0 1は、DEレジスタに格納されているアドレスを1加算し、サム算出カウンタの値を1減算する(S 1 1 4)。次いで、メインCPU 1 0 1は、更新後のサム算出カウンタの値が「0」であるか否かを判別する(S 1 1 5)。

30

【0533】

S 1 1 5において、メインCPU 1 0 1が、サム算出カウンタの値が「0」でないと判別したとき(S 1 1 5がNO判定の場合)、メインCPU 1 0 1は、処理をS 1 1 2の処理に戻し、S 1 1 2以降の処理を繰り返す。すなわち、メインRAM 1 0 3の規定外RAM領域のサム値を遊技用RAM領域のサム値に加算する処理が終了するまで、S 1 1 2～S 1 1 5の処理が繰り返される。

【0534】

一方、S 1 1 5において、メインCPU 1 0 1が、サム算出カウンタの値が「0」であると判別したとき(S 1 1 5がYES判定の場合)、メインCPU 1 0 1は、HLレジスタに格納されている値を電断発生時のサム値として、メインRAM 1 0 3内のサム値格納領域(不図示)に保存する(S 1 1 6)。次いで、メインCPU 1 0 1は、S 1 0 1で規定外スタックエリアに保存されたスタックポインタ(SP)の値をスタックポインタにセットする(S 1 1 7)。そして、S 1 1 7の処理後、メインCPU 1 0 1は、チェックサム生成処理を終了し、処理を電断時(外部)処理(図76参照)のS 9 8の処理に移す。

40

【0535】

本実施形態では、上述のようにしてチェックサム生成処理が行われる。そして、上述したチェックサム生成処理は、メインCPU 1 0 1が、図78Aのソースプログラムで規定されている各ソースコードを順次実行することにより行われる。上述のように、本実施形態では、電断発生時のメインRAM 1 0 3のチェックサムは、加算式で算出される。この際

50

、遊技用RAM領域のサム値算出では、2バイト単位で加算処理（図78A中のソースコード「ADD HL, DE」参照）が行われ、規定外RAM領域では、1バイト単位で加算処理（図78A中のソースコード「ADDWB HL, A」参照）が行われる。

【0536】

[サムチェック処理（規定外）]

次に、図79～図81を参照して、電源投入時処理（図64参照）中のS9で行うサムチェック処理について説明する。なお、図79及び図80は、サムチェック処理の手順を示すフローチャートであり、図81は、サムチェック処理中のS122～S132の処理を実行するためのソースプログラムの一例を示す図である。

【0537】

まず、メインCPU101は、現在のスタックポインタ（SP）の値を規定外スタックエリアに保存する（S121）。次いで、メインCPU101は、スタックポインタにサム値格納領域のアドレスをセットし、サム算出カウンタに、サム値の算出対象格納領域のバイト数を「2」で除算した値をセットする（S122）。なお、ここでセットされるサム算出カウンタは、サム値算出（サム値の減算処理）の終了契機を判定するためのカウンタであり、メインRAM103に設けられる。次いで、メインCPU101は、サム値格納領域からサム値（チェックサム）を取得する（S123）。この処理により、電断発生時に生成されたチェックサム（減算前の初期値）がHLレジスタに格納される。

【0538】

次いで、メインCPU101は、「POP」命令を実行し、スタックポインタ（SP）にセットされたメインRAM103の格納領域のアドレスから2バイト分の領域のデータ（保存値）をDEレジスタに読み出す（S124）。なお、この際、「POP」命令の実行により、スタックポインタで指定されたアドレスの1バイト領域に保存されているデータ（メモリ内容）が、Eレジスタにロードされ、スタックポインタで指定されたアドレスを1更新したアドレスの1バイト領域に保存されているデータ（メモリ内容）が、Dレジスタにロードされる（図78B参照）。また、「POP」命令が実行されると、スタックポインタ（SP）にセットされたアドレスに対して2バイト分のアドレス更新処理（アドレスを2加算する処理）が行われる。

【0539】

次いで、メインCPU101は、サム値の算出（減算）処理を行う（S125）。具体的には、メインCPU101は、HLレジスタに格納されている値（サム値の初期値又は前回の減算処理後のサム値）からDEレジスタに格納されている値を減算し、該減算された値をサム値としてHLレジスタに格納する。

【0540】

次いで、メインCPU101は、サム算出カウンタの値を1減算する（S126）。次いで、メインCPU101は、更新後のサム算出カウンタの値が「0」であるか否かを判別する（S127）。

【0541】

S127において、メインCPU101が、サム算出カウンタの値が「0」でないと判別したとき（S127がNO判定の場合）、メインCPU101は、処理をS124の処理に戻し、S124以降の処理を繰り返す。すなわち、メインRAM103の遊技用RAM領域の全域に渡ってサム値の減算処理が終了するまで、S124～S127の処理が繰り返される。

【0542】

一方、S127において、メインCPU101が、サム算出カウンタの値が「0」であると判別したとき（S127がYES判定の場合）、メインCPU101は、DEレジスタに、メインRAM103内の規定外RAM領域のサム値の計算開始アドレスをセットし、サム算出カウンタに、規定外用サムカウント値をセットする（S128）。なお、規定外用サムカウント値は、規定外RAM領域のバイト数となる。

【0543】

10

20

30

40

50

次いで、メインCPU101は、DEレジスタにセットされた規定外RAM領域のアドレスから1バイト分の領域のデータ(保存値)をAレジスタに読み出す(S129)。

【0544】

次いで、メインCPU101は、サム値の算出(減算)処理を行う(S130)。具体的には、メインCPU101は、HLレジスタに格納されている値からAレジスタに格納されている値を減算し、該減算された値をサム値としてHLレジスタに格納する。

【0545】

次いで、メインCPU101は、DEレジスタに格納されているアドレスを1加算し、サム算出カウンタの値を1減算する(S131)。次いで、メインCPU101は、更新後のサム算出カウンタの値が「0」であるか否かを判別する(S132)。

10

【0546】

S132において、メインCPU101が、サム算出カウンタの値が「0」でないと判別したとき(S132がNO判定の場合)、メインCPU101は、処理をS129の処理に戻し、S129以降の処理を繰り返す。すなわち、メインRAM103の規定外RAM領域の全域に渡ってサム値の減算処理が終了するまで、S129~S132の処理が繰り返される。

【0547】

一方、S132において、メインCPU101が、サム算出カウンタの値が「0」であると判別したとき(S132がYES判定の場合)、メインCPU101は、サムチェック処理の判定結果に「サム異常」をセットする(S133)。次いで、メインCPU101は、算出されたサム値が「0」であるか否かを判別する(S134)。

20

【0548】

なお、この処理では、メインCPU101は、フラグ・レジスタFのゼロフラグ(ビット6)の状態(1/0)を参照して、サム値が「0」であるか否かを判別する。本実施形態では、S128でセットされたサム算出カウンタの値が「0」になった時点、すなわち、メインRAM103の全域に渡ってサム値の減算処理が終了した時点において、サム値が「0」である場合には、フラグ・レジスタFのゼロフラグには「1」がセットされ、サム値が「0」でない場合には、フラグ・レジスタFのゼロフラグには「0」がセットされている。それゆえ、S134の処理の時点において、フラグ・レジスタFのゼロフラグに「1(オン状態)」がセットされていれば、メインCPU101はサム値が「0」であると判定する。

30

【0549】

S134において、メインCPU101が、算出されたサム値が「0」でないと判別したとき(S134がNO判定の場合)、メインCPU101は、後述のS139の処理を行う。一方、S134において、メインCPU101が、算出されたサム値が「0」であると判別したとき(S134がYES判定の場合)、メインCPU101は、判定結果に「電断異常」をセットする(S135)。

【0550】

次いで、メインCPU101は、電断発生フラグを取得する(S136)。次いで、メインCPU101は、電断発生フラグが電断なしの状態(オフ状態)であるか否かを判別する(S137)。

40

【0551】

S137において、メインCPU101が、電断発生フラグが電断なしの状態であると判別したとき(S137がYES判定の場合)、メインCPU101は、後述のS139の処理を行う。一方、S137において、メインCPU101が、電断発生フラグが電断なしの状態でないと判別したとき(S137がNO判定の場合)、メインCPU101は、判定結果に「正常」をセットする(S138)。

【0552】

S138の処理後、S134がNO判定の場合、又は、S137がYES判定の場合、メインCPU101は、サムチェック判定結果に判定結果を保存し、電断発生フラグをクリ

50

ア(オフ)する(S139)。次いで、メインCPU101は、S121で規定外スタックエリアに保存されたスタックポインタ(SP)の値をスタックポインタにセットする(S140)。そして、S140の処理後、メインCPU101は、サムチェック処理を終了し、処理を電源投入時処理(図64参照)のS10の処理に移す。

【0553】

本実施形態では、上述のようにしてサムチェック処理が行われる。そして、上述したサムチェック処理中のS122～S132の処理は、メインCPU101が、図81のソースプログラムで規定されている各ソースコードを順次実行することにより行われる。

【0554】

上述のように、本実施形態における、メインRAM103のサムチェックの判定処理では、まず、電源発生時に生成されたチェックサムの値を、電源復帰時のメインRAM103に格納されたデータで順次減算する。この際、遊技用RAM領域では、2バイト単位で減算処理(図81中のソースコード「SUB HL, DE」参照)が行われ、規定外RAM領域では、1バイト単位で減算処理(図81中のソースコード「SUBWB HL, A」参照)が行われる。次いで、最終的な減算結果が「0」であるか否か(ゼロフラグがオン状態であるか否か)に基づいて、異常の発生の有無を判定する。そして、減算結果が「0」である場合(ゼロフラグがオン状態である場合)には、正常と判定され、減算結果が「0」でない場合(ゼロフラグがオフ状態である場合)には、異常と判定される。

【0555】

すなわち、本実施形態では、電源発生時のチェックサムの生成処理は加算方式で行われ、電源復帰時のチェックサムの判定処理は減算方式で行われる。そして、チェックサムの最終的な減算結果に基づいて、正常/異常の判定が行われる。このようなチェックサムの生成処理及び判定処理を行った場合、電源復帰時に再度チェックサムを生成して、該チェックサムを電源発生時のチェックサムと照合する処理が不要となる。この場合、ソースプログラム上において、照合命令コードを省略することができ、ソースプログラムの容量を低減することができる。この結果、本実施形態では、メインROM102において、照合命令コードの省略分に対応する空き容量を確保することができ、増えた空き容量を活用して、遊技性を高めることが可能になる。なお、上述の電源発生時のチェックサムの生成処理、及び、電源復帰時のチェックサムの判定処理で実行される「POP」命令は、スタックポインタ(SP)操作専用命令であり、ソースコード「POP DE」以外にもソースコード「POP HL」、「POP AF」等が存在する。

【0556】

[メインCPUの制御によるパチスロのメイン処理]

次に、図82を参照して、メインCPU101の制御により実行されるパチスロ1のメイン処理(主要動作処理)について説明する。なお、図82は、メイン処理の手順を示すフローチャート(以下、メインフローという)である。

【0557】

まず、メインCPU101は、RAM初期化処理を行う(S201)。この処理では、メインCPU101は、図12Cに示すメインRAM103の遊技用RAM領域内の「一遊技終了時」のアドレスを、初期化開始の先頭アドレスとして設定し、該先頭アドレスから遊技用RAM領域の最終アドレスまでの情報を消去(クリア)する。なお、この範囲の格納領域は、例えば、内部当籤役格納領域や表示役格納領域などの1回の単位遊技(ゲーム)ごとにデータの消去が必要な格納領域である。

【0558】

次いで、メインCPU101は、メダル受付・スタートチェック処理を行う(S202)。この処理では、メインCPU101は、メダルセンサ(不図示)やスタートスイッチ79などの入力チェック処理等を行う。なお、メダル受付・スタートチェック処理の詳細については、後述の図83を参照しながら後で説明する。

【0559】

次いで、メインCPU101は、乱数取得処理を行う(S203)。この処理では、メイ

10

20

30

40

50

ンCPU101は、内部当籤役抽籤用の乱数値（0～65535：ハードラッチ乱数となる乱数回路110の乱数レジスタ0の値）やART関連の各種抽籤で用いられる演出用乱数値（0～65535：ソフトラッチ乱数となる乱数回路110の乱数レジスタ1～3の各値、0～255：ソフトラッチ乱数となる乱数回路110の乱数レジスタ4～7の各値）などを抽出し、該抽出した各種乱数値をメインRAM103に設けられた乱数値格納領域（不図示）に格納する。なお、乱数取得処理の詳細については、後述の図91を参照しながら後で説明する。

【0560】

次いで、メインCPU101は、内部抽籤処理を行う（S204）。この処理では、メインCPU101は、S203で抽出した乱数値（ハードラッチ乱数）に基づいた抽籤により内部当籤役の決定処理を行う。なお、内部抽籤処理の詳細については、後述の図92を参照しながら後で説明する。

10

【0561】

次いで、メインCPU101は、図柄設定処理を行う（S205）。この処理では、メインCPU101は、例えば、当り要求フラグステータス（フラグステータス情報）から内部当籤役を生成する処理、当り要求フラグデータの展開処理、当り要求フラグデータを当り要求フラグ格納領域を格納する処理等を行う。なお、図柄設定処理の詳細については、後述の図97を参照しながら後で説明する。

【0562】

次いで、メインCPU101は、スタートコマンド生成格納処理を行う（S206）。この処理では、メインCPU101は、副制御回路200に送信するスタートコマンドのデータを生成し、該コマンドデータをメインRAM103に設けられた通信データ格納領域（図75B参照）に保存する。通信データ格納領域に保存されたスタートコマンドは、後述の図158で説明する割込処理内の通信データ送信処理により、主制御回路90から副制御回路200に送信される。なお、スタートコマンドは、内部当籤役等を特定するパラメータ（サブフラグ等）を含んで構成される。

20

【0563】

次いで、メインCPU101は、第2インターフェースボード制御処理を行う（S207）。なお、第2インターフェースボード制御処理は、メインRAM103の規定外用作業領域で実行される。第2インターフェースボード制御処理の詳細については、後述の図102を参照しながら後で説明する。

30

【0564】

次いで、メインCPU101は、状態別制御処理を行う（S208）。この処理では、メインCPU101は、主に、遊技状態に応じた遊技開始時処理（スタート処理）を行う。なお、状態別制御処理の詳細については、後述の図104を参照しながら後で説明する。

【0565】

次いで、メインCPU101は、リール停止初期設定処理を行う（S209）。この処理では、メインCPU101は、リール停止初期設定テーブル（図26参照）を参照し、内部当籤役及び遊技状態に基づいて、引込優先順位テーブル選択テーブル番号、引込優先順位テーブル番号、停止テーブル番号を取得する処理や、ストップボタン未作動カウンタに「3」を格納する処理などを行う。

40

【0566】

次いで、メインCPU101は、リール回転開始処理を行う（S210）。この処理では、メインCPU101は、全リールの回転開始を要求する。そして、全リールの回転開始が要求されると、一定の周期（1.1172 msec）で実行される後述の割込処理（後述の図158参照）により、3つのステップモータ（不図示）の駆動が制御され、左リール3L、中リール3C及び右リール3Rの回転が開始される。次いで、各リールは、その回転速度が定速度に達するまで加速制御され、その後、該定速度が維持されるように制御される。

【0567】

50

次いで、メインCPU101は、リール回転開始コマンド生成格納処理を行う(S211)。この処理では、メインCPU101は、副制御回路200に送信するリール回転開始コマンドのデータを生成し、該コマンドデータをメインRAM103に設けられた通信データ格納領域(図75B参照)に保存する。通信データ格納領域に保存されたリール回転開始コマンドは、後述の図158で説明する割込処理内の通信データ送信処理により、主制御回路90から副制御回路200に送信される。なお、リール回転開始コマンドは、リールの回転開始動作開始されたことを示すパラメータを含んで構成される。

【0568】

次いで、メインCPU101は、引込優先順位格納処理を行う(S212)。この処理では、メインCPU101は、引込優先順位データを取得して、引込優先順位データ格納領域に格納する。なお、引込優先順位格納処理の詳細については、後述の図126を参照しながら後で説明する。

10

【0569】

次いで、メインCPU101は、リール停止制御処理を行う(S213)。この処理では、メインCPU101は、左ストップボタン17L、中ストップボタン17C及び右ストップボタン17Rがそれぞれ押されたタイミングと内部当籤役とに基づいて該当するリールの回転の停止制御を行う。なお、リール停止制御処理の詳細については、後述の図138を参照しながら後で説明する。

【0570】

次いで、メインCPU101は、入賞検索処理を行う(S214)。この処理では、メインCPU101は、図柄コード格納領域(図35参照)のデータを入賞作動フラグ格納領域(図28~図30参照)に格納する。また、この処理では、メインCPU101は、有効ラインに表示役が表示されたか否かを判定し、その判定結果に基づいて、メダルの払出枚数をセットする。なお、入賞検索処理の詳細については、後述の図145を参照しながら後で説明する。

20

【0571】

次いで、メインCPU101は、イリーガルヒットチェック処理を行う(S215)。この処理では、メインCPU101は、当り要求フラグ(内部当籤役)と入賞作動フラグ(表示役)とを合成し、その合成結果に基づいてイリーガルヒットエラーの有無を判定する。なお、イリーガルヒットチェック処理の詳細については、後述の図148を参照しながら後で説明する。

30

【0572】

次いで、メインCPU101は、入賞チェック・メダル払出処理を行う(S216)。この処理では、メインCPU101は、入賞作動コマンドの生成処理を行う。また、この処理では、メインCPU101は、S214において決定された表示役の払出枚数に基づいて、ホッパー装置51の駆動やクレジット枚数の更新を行い、メダルの払い出し処理を行う。なお、入賞チェック・メダル払出処理の詳細については、後述の図150を参照しながら後で説明する。

【0573】

次いで、メインCPU101は、BBチェック処理を行う(S217)。この処理では、メインCPU101は、ボーナス状態の作動及び終了を制御する。なお、BBチェック処理の詳細については、後述の図154を参照しながら後で説明する。

40

【0574】

次いで、メインCPU101は、RTチェック処理を行う(S218)。この処理では、メインCPU101は、有効ライン上に停止表示された図柄組合せに基づいてRT状態の移行制御を行う。なお、RTチェック処理の詳細については、後述の図155及び図156を参照しながら後で説明する。

【0575】

次いで、メインCPU101は、CZ・ART終了時処理を行う(S219)。この処理では、メインCPU101は、主に、CZの引き戻し抽籤処理を行う。なお、CZ・AR

50

T終了時処理の詳細については、後述の図157を参照しながら後で説明する。そして、S219の処理後（一遊技終了後）、メインCPU101は、処理をS201の処理に戻す。

【0576】

[メダル受付・スタートチェック処理]

次に、図83及び図84を参照して、メインフロー（図82参照）中のS202で行うメダル受付・スタートチェック処理について説明する。なお、図83は、メダル受付・スタートチェック処理の手順を示すフローチャートであり、図84は、メダル受付・スタートチェック処理中のS231～S233の処理を実行するためのソースプログラムの一例を示す図である。

10

【0577】

まず、メインCPU101は、自動投入メダルカウンタの値が「0」であるか否か（自動投入要求はあるか否か）を判別する（S221）。なお、この処理において、自動投入メダルカウンタが「1」以上であるときは、メインCPU101は、自動投入要求があると判別する。また、自動投入メダルカウンタは、前回の単位遊技において再遊技（リプレイ）に係る表示役が成立したか否かを識別するためのデータである。再遊技に係る表示役が成立したときには、前回の単位遊技において投入された枚数分のメダルが自動投入メダルカウンタに自動的に投入される。

【0578】

S221において、メインCPU101が、自動投入メダルカウンタの値が「0」であると判別したとき（S221がYES判定の場合）、メインCPU101は、後述のS225の処理を行う。

20

【0579】

一方、S221において、メインCPU101が、自動投入メダルカウンタの値が「0」でないと判別したとき（S221がNO判定の場合）、メインCPU101は、メダル投入処理を行う（S222）。この処理では、メインCPU101は、メダル投入コマンドの生成格納処理やメダル投入枚数のLED点灯制御処理などを行う。なお、メダル投入処理の詳細については、後述の図85を参照しながら後で説明する。

【0580】

次いで、メインCPU101は、自動投入メダルカウンタの値を1減算する（S223）。次いで、減算後の自動投入メダルカウンタの値が「0」であるか否かを判別する（S224）。

30

【0581】

S224において、メインCPU101が、自動投入メダルカウンタの値が「0」でないと判別したとき（S224がNO判定の場合）、メインCPU101は、処理をS222の処理に戻し、S222以降の処理を繰り返す。

【0582】

一方、S224において、メインCPU101が、自動投入メダルカウンタの値が「0」とであると判別したとき（S224がYES判定の場合）、又は、S221がYES判定の場合、メインCPU101は、メダル補助収納庫スイッチチェック処理を行う（S225）。この処理では、メインCPU101は、メダル補助収納庫スイッチ75のオン/オフ状態に基づいて、メダル補助収納庫52がメダルで満杯になっているか否かを検出する。

40

【0583】

次いで、メインCPU101は、メダル投入状態チェック処理を行う（S226）。次いで、メインCPU101は、メダル投入状態チェック処理の結果に基づいて、メダル投入可能な状態であるか否かを判別する（S227）。

【0584】

S227において、メインCPU101が、メダル投入可能な状態でないと判別したとき（S227がNO判定の場合）、メインCPU101は、後述のS231の処理を行う。

【0585】

50

一方、S 2 2 7において、メインCPU 1 0 1が、メダル投入可能な状態であると判断したとき（S 2 2 7がYES判定の場合）、メインCPU 1 0 1は、メダル投入チェック処理を行う（S 2 2 8）。この処理では、メインCPU 1 0 1は、例えば、メダルセンサ入力状態に基づいて、メダルが正常に通過したか否かの判定処理や、規定数を超えてメダル投入が行われた場合に該メダルをクレジットする処理などをを行う。なお、メダル投入チェック処理の詳細については、後述の図 8 7を参照しながら後で説明する。

【 0 5 8 6 】

次いで、メインCPU 1 0 1は、メダル投入チェック処理の結果に基づいて、メダル投入又はクレジット可能な状態であるか否かを判断する（S 2 2 9）。

【 0 5 8 7 】

S 2 2 9において、メインCPU 1 0 1が、メダル投入又はクレジット可能な状態であると判断したとき（S 2 2 9がYES判定の場合）、メインCPU 1 0 1は、後述のS 2 3 1の処理を行う。一方、S 2 2 9において、メインCPU 1 0 1が、メダル投入又はクレジット可能な状態でないと判断したとき（S 2 2 9がNO判定の場合）、メインCPU 1 0 1は、メダル受付禁止の処理を行う（S 2 3 0）。この処理により、セレクタ 6 6（図 4 参照）のソレノイドの駆動が行われず、投入されたメダルがメダル払出口 2 4から排出される。

【 0 5 8 8 】

S 2 3 0の処理後、S 2 2 7がNO判定の場合、又は、S 2 2 9がYES判定の場合、メインCPU 1 0 1は、現在のメダルの投入枚数が遊技可能開始枚数であるか否かを判断する（S 2 3 1）。なお、本実施形態では、遊技開始可能枚数は3枚である（図 2 8 ~ 図 3 0 参照）。

【 0 5 8 9 】

S 2 3 1において、メインCPU 1 0 1が、現在のメダルの投入枚数が遊技可能開始枚数であると判断したとき（S 2 3 1がYES判定の場合）、メインCPU 1 0 1は、後述のS 2 3 4の処理を行う。一方、S 2 3 1において、メインCPU 1 0 1が、現在のメダルの投入枚数が遊技可能開始枚数でないと判断したとき（S 2 3 1がNO判定の場合）、メインCPU 1 0 1は、メダル投入があるか否かを判断する（S 2 3 2）。

【 0 5 9 0 】

S 2 3 2において、メインCPU 1 0 1が、メダル投入があると判断したとき（S 2 3 2がYES判定の場合）、メインCPU 1 0 1は、処理をS 2 2 6に戻し、S 2 2 6以降の処理を繰り返す。一方、S 2 3 2において、メインCPU 1 0 1が、メダル投入がないと判断したとき（S 2 3 2がNO判定の場合）、メインCPU 1 0 1は、図 6 8で説明した設定変更確認処理を行う（S 2 3 3）。この処理では、メインCPU 1 0 1は、設定確認開始時の設定変更コマンドの生成格納処理などを行う。

【 0 5 9 1 】

S 2 3 3の処理後又はS 2 3 1がYES判定の場合、メインCPU 5 1は、スタートスイッチ 7 9がオン状態であるか否かを判断する（S 2 3 4）。

【 0 5 9 2 】

S 2 3 4において、メインCPU 1 0 1が、スタートスイッチ 7 9がオン状態でないと判断したとき（S 2 3 4がNO判定の場合）、メインCPU 1 0 1は、処理をS 2 2 6に戻し、S 2 2 6以降の処理を繰り返す。

【 0 5 9 3 】

一方、S 2 3 4において、メインCPU 1 0 1が、スタートスイッチ 7 9がオン状態であると判断したとき（S 2 3 4がYES判定の場合）、メインCPU 1 0 1は、メダル受付禁止の処理を行う（S 2 3 5）。この処理により、セレクタ 6 6（図 4 参照）のソレノイドの駆動が行われず、投入されたメダルがメダル払出口 2 4から排出される。そして、S 2 3 5の処理後、メインCPU 1 0 1は、メダル受付・スタートチェック処理を終了し、処理をメインフロー（図 8 2 参照）のS 2 0 3に移す。

【 0 5 9 4 】

10

20

30

40

50

本実施形態では、上述のようにしてメダル受付・スタートチェック処理が行われる。そして、上述したメダル受付・スタートチェック処理中のS 2 3 1～S 2 3 3の処理は、メインCPU 1 0 1が、図 8 4のソースプログラムで規定されている各ソースコードを順次実行することにより行われる。その中でも、S 2 3 3の処理では、メインCPU 1 0 1専用命令コードである「CALLF」命令により、処理を設定変更確認処理の実行プログラムのアドレス「SB__WVSC__00」にジャンプさせ、図 6 8及び図 6 9で説明した設定変更確認処理を行う。

【0595】

そして、上述したメダル受付・スタートチェック処理中のS 2 3 3の処理、すなわち、設定変更確認処理は、遊技状態に関係なく実行される。それゆえ、本実施形態では、遊技状態に関係なく、すなわち、遊技状態がボーナス状態（特賞作動状態）であっても、設定値及びホールメニュー（各種履歴データ（エラー、電断履歴等））を確認することができ、ゴト等の不正行為を抑制することができる。

10

【0596】

[メダル投入処理]

次に、図 8 5及び図 8 6を参照して、メダル受付・スタートチェック処理（図 8 3参照）中のS 2 2 2で行うメダル投入処理について説明する。なお、図 8 5は、メダル投入処理の手順を示すフローチャートであり、図 8 6は、メダル投入処理を実行するためのソースプログラムの一例を示す図である。

【0597】

まず、メインCPU 1 0 1は、メダルカウンタの値に「1」を加算する（S 2 4 1）。なお、メダルカウンタは、メダルの投入枚数をカウント（計数）するためのカウンタであり、メインRAM 1 0 3に設けられる。

20

【0598】

次いで、メインCPU 1 0 1は、メダル投入コマンド生成格納処理を行う（S 2 4 2）。この処理では、メインCPU 1 0 1は、副制御回路 2 0 0に送信するメダル投入コマンドのデータを生成し、該コマンドデータをメインRAM 1 0 3に設けられた通信データ格納領域（図 7 5 B参照）に保存する。通信データ格納領域に保存されたメダル投入コマンドは、後述の図 1 5 8で説明する割込処理内の通信データ送信処理により、主制御回路 9 0から副制御回路 2 0 0に送信される。すなわち、メダル投入コマンドは、メダルが1枚投入される度に、主制御回路 9 0から副制御回路 2 0 0に送信される。なお、メダル投入コマンドは、投入枚数等を特定するためのパラメータを含んで構成される。

30

【0599】

次いで、メインCPU 1 0 1は、LED 8 2（図 7参照）に含まれるメダル投入枚数表示用の第1～第3LEDを消灯させる（S 2 4 3）。次いで、メインCPU 1 0 1は、メダル投入枚数（メダルカウンタの値）に基づいて、該メダル投入枚数に対応するLED点灯データ（点灯制御データ）を算出する（S 2 4 4）。この処理において、例えば、メダル投入枚数が1枚である場合には、メダル投入枚数表示用の第1LEDのみを点灯させるLED点灯データが算出され、また、例えば、メダル投入枚数が3枚である場合には、メダル投入枚数表示用の第1～第3LEDの全てを点灯させるLED点灯データが算出される。なお、このLED点灯データの算出手法については、後で詳述する。

40

【0600】

次いで、メインCPU 1 0 1は、算出されたLED点灯データを用いて、対応するメダル投入枚数表示用のLEDを点灯させる（S 2 4 5）。そして、S 2 4 5の処理後、メインCPU 1 0 1は、メダル投入処理を終了し、処理をメダル受付・スタートチェック処理（図 8 3参照）のS 2 2 3に移す。

【0601】

本実施形態では、上述のようにしてメダル投入処理が行われる。なお、上述したメダル投入処理は、メインCPU 1 0 1が、図 8 6のソースプログラムで規定されている各ソースコードを順次実行することにより行われる。その中でS 2 4 4の処理では、メダル投入枚

50

数表示用のLED点灯データがテーブルを参照したループ処理でなく、演算処理により生成される。この演算処理は、メインCPU101が図86に示すソースプログラム中のソースコード「LD A, L」～「OR L」を順次実行することにより行われる。

【0602】

この演算処理では、まず、ソースコード「LD A, L」の実行により、Lレジスタに格納されたメダル投入枚数のデータがAレジスタに格納される。例えば、メダル投入枚数が3枚である場合には、「00000011B」（10進数で「3」）がAレジスタに格納される。なお、本実施形態において、1バイトデータを「*****B」と記すが、最後の文字「B」は、文字「B」の前に示された「0」又は「1」がビットデータであることを意味する。

10

【0603】

次いで、ソースコード「ADD A, A」の実行により、Aレジスタに格納されたデータにAレジスタに格納されたデータが加算され、該加算結果がAレジスタに格納される。例えば、この「ADD」命令の実行前の時点でAレジスタに格納されているデータが「00000011B」である場合（メダル投入枚数が3枚である場合）、この「ADD」命令により、加算結果となる「00000110B」がAレジスタに格納される。

【0604】

次いで、ソースコード「DEC A」の実行により、Aレジスタに格納されているデータが1減算され、該減算結果がAレジスタに格納される。例えば、この「DEC」命令の実行前の時点でAレジスタに格納されているデータが「00000110B」である場合、この「DEC」命令により、減算結果となる「00000101B」がAレジスタに格納される。

20

【0605】

次いで、ソースコード「OR L」の実行により、Lレジスタに格納されているデータ（メダル投入枚数）とAレジスタに格納されているデータとの論理和演算が行われ、該演算結果がAレジスタに格納される。例えば、この「OR」命令の実行前の時点でAレジスタに格納されているデータが「00000101B」であり、Lレジスタに格納されているデータが「00000011B」である場合（メダル投入枚数が3枚である場合）、この「OR」命令により、両データの論理和演算の結果となる「00000111B」がAレジスタに格納される。そして、「OR」命令の実行によりAレジスタに格納されたデータが、メダル投入枚数表示用のLED点灯データ（メダル投入LEDの点灯状態を示すデータ）となる。

30

【0606】

例えば、メダル投入枚数が3枚である場合には、上述のように、S244のメダル投入枚数表示用のLED点灯データの算出処理により、最終的な算出結果「00000111B」がメダル投入枚数表示用のLED点灯データとなる。そして、本実施形態では、最終的に算出されたLED点灯データのビット0の「1/0」が1枚目のメダル投入枚数表示用のLED（第1LED）への出力ポートの「オン/オフ」状態に対応し、ビット1の「1/0」が2枚目のメダル投入枚数表示用のLED（第2LED）への出力ポートの「オン/オフ」状態に対応し、ビット2の「1/0」が3枚目のメダル投入枚数表示用のLED（第3LED）への出力ポートの「オン/オフ」状態に対応する。それゆえ、メダル投入枚数が3枚である場合には、上述のように、LED点灯データとして「00000111B」が生成されるので、メダル投入枚数表示用の第1～第3LEDの全ての出力ポートがオン状態にセットされ、メダル投入枚数表示用の第1～第3LEDが全て点灯状態となる。

40

【0607】

上述のようにしてメダル投入枚数表示用のLED点灯データを演算処理により生成した場合、メダル投入枚数表示用のLED点灯データを生成する際に参照するテーブルデータが不要となるのでメインROM102のテーブル領域の空き容量を増やすことができるとともに、プログラムの容量増を最小限に抑えることができる。すなわち、本実施形態の上述したメダル投入処理では、メダル投入LED表示の処理を効率化することができるとも

50

に、メインROM 102の空き容量を確保し(増やし)、該増えた空き領域を活用して、遊技性を高めることが可能になる。

【0608】

[メダル投入チェック処理]

次に、図87及び図88を参照して、メダル受付・スタートチェック処理(図83参照)中のS228で行うメダル投入チェック処理について説明する。なお、図87は、メダル投入チェック処理の手順を示すフローチャートであり、図88は、メダル投入チェック処理中のS255~S258の処理を実行するためのソースプログラムの一例を示す図である。

【0609】

まず、メインCPU101は、再遊技中であるか否かを判別する(S251)。

【0610】

S251において、メインCPU101が、再遊技中であると判別したとき(S251がYES判定の場合)、メインCPU101は、メダル投入チェック処理を終了し、処理をメダル受付・スタートチェック処理(図83参照)のS229に移す。

【0611】

一方、S251において、メインCPU101が、再遊技中でないと判別したとき(S251がNO判定の場合)、メインCPU101は、メダル受付許可を行う(S252)。この処理では、セクタ66(図4参照)のソレノイドの駆動が行われ、メダル投入口24から投入されたメダルが受け入れられる。受け入れられたメダルは計数されてからホッパー装置51へ案内される。

【0612】

次いで、メインCPU101は、ベットボタンチェック処理を行う(S253)。この処理では、メインCPU101は、BETスイッチ77のオン/オフ状態に基づいて、ベットボタン(MAXベットボタン15a又は1ベットボタン15b)の操作が行われたか否かを判別する。次いで、メインCPU101は、S253のベットボタンチェック処理の結果に基づいて、ベット動作が完了したか否かを判別する(S254)。

【0613】

S254において、メインCPU101が、ベット動作が完了したと判別したとき(S254がYES判定の場合)、メインCPU101は、メダル投入チェック処理を終了し、処理をメダル受付・スタートチェック処理(図83参照)のS229に移す。

【0614】

一方、S254において、メインCPU101が、ベット動作が完了していないと判別したとき(S254がNO判定の場合)、メインCPU101は、現処理時のメダルセンサ入力状態(遊技媒体の受付状態)と、前回処理時のメダルセンサ入力状態とを取得する(S255)。なお、メダルセンサ入力状態は、メダル投入口24に受け入れられたメダルのセクタ66内の通過状況を示す情報であり、セクタ66に入口及び出口に設けられた各メダルセンサ(不図示)の検知結果により生成される。

【0615】

本実施形態では、メダルセンサ入力状態は、1バイト(8ビット)のデータで表され、セクタ66の出口にメダルの通過方向に並んで設けられた上流側の第1メダルセンサ(不図示)の検知結果がビット0の情報(「0」又は「1」)に対応し、下流側の第2メダルセンサ(不図示)の検知結果がビット1の情報(「0」又は「1」)に対応する。第1メダルセンサによりメダルの通過が検知された場合には、ビット0に「1」がセットされ、第2メダルセンサによりメダルの通過が検知された場合には、ビット1に「1」がセットされる。それゆえ、メダルセンサ入力状態「00000000B」は、メダル通過前又は通過後(通過時)の状態を示し、メダルセンサ入力状態「00000001B」は、メダル通過開始時の状態を示し、メダルセンサ入力状態「00000011B」は、メダル通過中の状態を示し、メダルセンサ入力状態「00000010B」は、メダル通過完了直前の状態を示す。

10

20

30

40

50

【0616】

次いで、メインCPU101は、現処理時のメダルセンサ入力状態が前回処理時のメダルセンサ入力状態から変化したか否かを判別する(S256)。

【0617】

S256において、メインCPU101が、現処理時のメダルセンサ入力状態が前回処理時のメダルセンサ入力状態から変化していないと判別したとき(S256がNO判定の場合)、メインCPU101は、後述のS261の処理を行う。

【0618】

一方、S256において、メインCPU101が、現処理時のメダルセンサ入力状態が前回処理時のメダルセンサ入力状態から変化したと判別したとき(S256がYES判定の場合)、メインCPU101は、前回処理時のメダルセンサ入力状態に基づいて、演算処理により、現処理時で得られるメダルセンサ入力状態の正常値(正常変化値)を生成する(S257)。

10

【0619】

なお、この処理において、前回処理時のメダルセンサ入力状態が「00000000B」である場合(第1及び第2メダルセンサがともにメダル未検知である場合)には、メダルセンサ入力状態の正常変化値として「00000001B」(第1メダルセンサがメダル検知であり、第2メダルセンサがメダル未検知である場合)が生成され、前回処理時のメダルセンサ入力状態が「00000001B」である場合には、メダルセンサ入力状態の正常変化値として「00000011B」(第1及び第2メダルセンサがともにメダル検知である場合)が生成される。また、この処理において、前回処理時のメダルセンサ入力状態が「00000011B」である場合には、メダルセンサ入力状態の正常変化値として「00000010B」(第1メダルセンサがメダル未検知であり、第2メダルセンサがメダル検知である場合)が生成され、前回処理時のメダルセンサ入力状態が「000000010B」である場合には、メダルセンサ入力状態の正常変化値として「00000000B」(第1及び第2メダルセンサがともにメダル未検知である場合)が生成される。なお、メダルセンサ入力状態の正常変化値の生成(算出)手法については後で詳述する。

20

【0620】

次いで、メインCPU101は、現処理時のメダルセンサ入力状態がS257で生成された正常変化値と同じであるか否かを判別する(S258)。なお、この判定処理では、メダル逆行エラーの発生の有無が判定され、S258の判定条件が満たされない場合には、メインCPU101は、メダル逆行エラーが発生したと判定する。

30

【0621】

S258において、メインCPU101が、現処理時のメダルセンサ入力状態がS257で生成された正常変化値と同じでないと判別したとき(S258がNO判定の場合)、メインCPU101は、後述のS262の処理を行う。

【0622】

一方、S258において、メインCPU101が、現処理時のメダルセンサ入力状態がS257で生成された正常変化値と同じであると判別したとき(S258がYES判定の場合)、メインCPU101は、現処理時のメダルセンサ入力状態がメダル通過時の状態(「00000000B」)であるか否かを判別する(S259)。S259において、メインCPU101が、現処理時のメダルセンサ入力状態がメダル通過時の状態であると判別したとき(S259がYES判定の場合)、メインCPU101は、後述のS263の処理を行う。

40

【0623】

S259において、メインCPU101が、現処理時のメダルセンサ入力状態がメダル通過時の状態でないと判別したとき(S259がNO判定の場合)、メインCPU101は、メダル通過チェックタイマーをセットする(S260)。この処理でメダル通過チェックタイマーにセットされる時間は、メダルがセクタ66を通過したか否かを判別可能な時間であれば、任意の時間に設定することができる。また、この処理でセットされるタイ

50

マー値は、例えば、現処理時のメダルセンサ入力状態に応じて変化させてもよい。

【0624】

S260の処理後又はS256がNO判定の場合、メインCPU101は、現処理時のメダルセンサ入力状態がメダル通過中の状態（「00000011B」）であり、かつ、メダル通過チェックタイマーが停止しているか否かを判別する（S261）。この判定処理では、メダル通過エラー（投入メダル通過時間エラー）の発生の有無が判定され、S261の判定条件が満たされた場合、メインCPU101は、メダル通過エラーが発生したと判定する。

【0625】

S261において、メインCPU101が、S261の判定条件が満たされないと判別したとき（S261がNO判定の場合）、メインCPU101は、処理をS253の処理に戻し、S253以降の処理を繰り返す。

10

【0626】

一方、S261において、メインCPU101が、S261の判定条件が満たされると判別したとき（S261がYES判定の場合）、又は、S258がNO判定の場合、すなわち、メダル通過エラー又はメダル逆行エラーが発生したと判定された場合、メインCPU101は、エラー処理を行う（S262）。この処理では、メインCPU101は、例えば、エラーコマンド生成格納処理等のエラー発生時の各種処理を行う。なお、エラー処理の詳細については、後述の図89を参照しながら後で説明する。そして、S262の処理後、メインCPU101は、処理をS253の処理に戻し、S253以降の処理を繰り返す。

20

【0627】

ここで再度、S259の処理に戻って、S259がYES判定の場合、メインCPU101は、規定数（本実施形態では3枚）のメダルが投入済みの状態であるか否かを判別する（S263）。

【0628】

S263において、メインCPU101が、規定数のメダルが投入済みの状態でないと判別したとき（S263がNO判定の場合）、メインCPU101は、図85で説明したメダル投入処理を行う（S264）。そして、S264の処理後、メインCPU101は、処理をS253の処理に戻し、S253以降の処理を繰り返す。

30

【0629】

一方、S263において、メインCPU101が、規定数のメダルが投入済みの状態であると判別したとき（S263がYES判定の場合）、メインCPU101は、クレジットカウンタの値に「1」を加算する（S265）。次いで、メインCPU101は、メダル投入コマンド生成格納処理を行う（S266）。この処理では、メインCPU101は、副制御回路200に送信するメダル投入コマンドのデータを生成し、該コマンドデータをメインRAM103に設けられた通信データ格納領域（図75B参照）に保存する。通信データ格納領域に保存されたメダル投入コマンドは、後述の図158で説明する割込処理内の通信データ送信処理により、主制御回路90から副制御回路200に送信される。

【0630】

次いで、メインCPU101は、クレジットカウンタの値に基づいて、メダルのクレジット枚数が上限値（本実施形態では50枚）であるか否かを判別する（S267）。

40

【0631】

S267において、メインCPU101が、メダルのクレジット枚数が上限値でないと判別したとき（S267がNO判定の場合）、メインCPU101は、処理をS253の処理に戻し、S253以降の処理を繰り返す。一方、S267において、メインCPU101が、メダルのクレジット枚数が上限値であると判別したとき（S267がYES判定の場合）、メインCPU101は、メダル投入チェック処理を終了し、処理をメダル受付・スタートチェック処理（図83参照）のS229に移す。

【0632】

50

本実施形態では、上述のようにしてメダル投入チェック処理が行われる。そして、上述したメダル投入チェック処理中のS255～S258の処理は、メインCPU101が、図88のソースプログラムで規定されている各ソースコードを順次実行することにより行われる。その中で、S257のメダルセンサ入力状態の正常変化値の生成処理は、テーブルを参照して取得する処理ではなく、演算処理により行われる。具体的には、正常変化値の生成処理は、メインCPU101が図88に示すソースプログラム中のソースコード「RLA」及び「AND cBX_MDISW」をこの順で実行することにより行われる。

【0633】

「RLA」命令は、Aレジスタに格納された1バイトのデータを、左(ビット0からビット7に向かう方向)に1回(1ビット分)シフトさせる命令コードである。図88に示す例では、「RLA」命令より前に実行されるソースコード「LD A, B」によりAレジスタに格納された前回のメダルセンサ入力状態を示す1バイトのデータが、「RLA」命令により、左に1回シフトされる。この際、ビット0に新たに格納されるビットデータは、「RLA」命令より前に実行されるソースコード「CP cBX_MDISW2」の実行結果に基づいて決定される。

10

【0634】

「CP」命令は比較動作を実行する命令コードである。また、「cBX_MDISW2」は、1バイトのデータであり、本実施形態では「00000010B」である。ソースコード「CP cBX_MDISW2」が実行されると、Aレジスタに格納された前回のメダルセンサ入力状態を示す1バイトのデータが、「cBX_MDISW2(00000010B)」と比較される。

20

【0635】

そして、ソースコード「CP cBX_MDISW2」を実行した結果、前回のメダルセンサ入力状態を示す1バイトのデータが「cBX_MDISW2(00000010B)」未満であるという結果が得られた場合にはフラグレジスタFのキャリーフラグ(図11参照)に「1」がセットされ、「RLA」命令の実行時に、Aレジスタのビット0にフラグレジスタFのキャリーフラグの「1」が格納される。一方、ソースコード「CP cBX_MDISW2」を実行した結果、前回のメダルセンサ入力状態を示す1バイトのデータが「cBX_MDISW2(00000010B)」以上であるという結果が得られた場合にはフラグレジスタFのキャリーフラグ(図11参照)に「0」がセットされ、「RLA」命令の実行時により、Aレジスタのビット0にフラグレジスタFのキャリーフラグの「0」が格納される。

30

【0636】

それゆえ、例えば、前回のメダルセンサ入力状態を示す1バイトのデータが「00000000B(メダル通過前又は通過後(通過時)の状態)」(<「cBX_MDISW2」)であれば、「RLA」命令の実行により、「00000001B」が生成され、前回のメダルセンサ入力状態を示す1バイトのデータが「00000001B(メダル通過開始時の状態)」(<「cBX_MDISW2」)であれば、「RLA」命令の実行により、「00000011B」が生成される。一方、例えば、前回のメダルセンサ入力状態を示す1バイトのデータが「00000011B(メダル通過中の状態)」(>「cBX_MDISW2」)であれば、「RLA」命令の実行により、「00000110B」が生成され、前回のメダルセンサ入力状態を示す1バイトのデータが「00000010B(メダル通過完了直前の状態)」(=「cBX_MDISW2」)であれば、「RLA」命令の実行により、「00000100B」が生成される。

40

【0637】

次いで、ソースコード「AND cBX_MDISW」が実行されると、「RLA」命令の実行により生成された1バイトデータ(Aレジスタの格納データ)が、1バイトのデータ「cBX_MDISW」と論理積され、メダルセンサ入力状態の正常変化値が算出される。なお、1バイトのデータ「cBX_MDISW」は、本実施形態では「00000011B」である。それゆえ、ソースコード「AND cBX_MDISW」が実行

50

されれば、Aレジスタの格納データ中のビット0及びビット1のデータだけがマスクされ、その他のビットデータが「0」になる。

【0638】

その結果、例えば、前回のメダルセンサ入力状態を示す1バイトのデータが「00000000B（メダル通過前の状態）」であれば、メダルセンサ入力状態の正常変化値として「00000001B（メダル通過開始時の状態）」が生成され、前回のメダルセンサ入力状態を示す1バイトのデータが「00000001B（メダル通過開始時の状態）」であれば、メダルセンサ入力状態の正常変化値として「00000011B（メダル通過中の状態）」が生成される。一方、例えば、前回のメダルセンサ入力状態を示す1バイトのデータが「00000011B（メダル通過中の状態）」であれば、メダルセンサ入力状態の正常変化値として「00000010B（メダル通過完了直前の状態）」が生成され、前回のメダルセンサ入力状態を示す1バイトのデータが「00000010B」（メダル通過完了直前の状態）であれば、メダルセンサ入力状態の正常変化値として「00000000B（メダル通過後（通過時）の状態）」が生成される。

10

【0639】

上述のようにして、メダルセンサ入力状態の変化態様の検知処理をテーブル参照処理から演算処理に変更することにより、メインROM102のテーブル格納領域の空き容量を増やすことができるとともに、プログラムの容量増を最小限に抑えることができる。それゆえ、上述した手法を採用することにより、メダル投入センサ状態の検知処理を効率化することができることにより、メインROM102において増えた空き容量を活用して、遊技性を高めることが可能となる。

20

【0640】

[エラー処理]

次に、図89及び図90を参照して、例えば、メダル投入チェック処理（図87参照）中のS262で行うエラー処理について説明する。図89は、エラー処理の手順を示すフローチャートであり、図90は、エラー処理のソースプログラム上で、実際に参照されるエラーテーブルの構成の一例を示す図である。

【0641】

なお、図90に示すエラーテーブルでは、エラー要因の種別を示すポートのオン/オフ状態を表す1バイトデータ（図90中の例えばアドレス「dERR_HE」に格納されている1バイトデータ等）毎に、エラー表示データ（図90中の例えばアドレス「dERR_HE+1」及び「dERR_HE+2」に格納されている1バイトデータ等）が規定される。このエラー表示データは、情報表示器6に含まれる2桁の7セグLED（払出枚数表示用及びエラー表示用兼用）に出力される。

30

【0642】

まず、メインCPU101は、メダルソレノイドのオフ処理を行う（S271）。具体的には、メインCPU101は、セクタ66（図7参照）のソレノイドの駆動を停止する。次いで、メインCPU101は、メダルの払出枚数表示データの退避処理を行う（S272）。

【0643】

次いで、メインCPU101は、エラーテーブルのセット処理を行う（S273）。この処理により、図90に示すエラーテーブルの先頭アドレスがソースプログラム上にセットされる。

40

【0644】

次いで、メインCPU101は、エラー要因を取得する（S274）。なお、この処理で取得されるエラー要因は、現在処理中のエラー処理を読み出した処理に応じて変化する。なお、本実施形態で対象とするエラー要因としては、図90に示すように、「ホッパーエンptyエラー」、「ホッパージャムエラー」、「投入メダル通過カウントエラー」、「投入メダル通過チェックエラー」、「投入メダル通過チェックエラー」、「投入メダル通過時間エラー」、「投入メダル逆行エラー」、「投入メダル補助収納庫満杯エラー」、「

50

イリーガルヒットエラー」が規定される。例えば、メダル投入チェック処理中の S 2 5 8 の処理後にエラー処理が読み出された場合には、この処理において、エラー要因として図 9 0 中の「投入メダル逆行エラー (C r)」が取得される。また、例えば、メダル投入チェック処理中の S 2 6 1 の処理後にエラー処理が読み出された場合には、この処理において、エラー要因として図 9 0 中の「投入メダル通過時間エラー (C E)」が取得される。

【 0 6 4 5 】

次いで、メイン CPU 1 0 1 は、エラーテーブルとエラー要因とから、エラー表示データを取得する (S 2 7 5)。例えば、エラー要因が「投入メダル逆行エラー (C r)」である場合、この処理において、2 桁の 7 セグ LED のうち、上位桁の 7 セグ LED に出力するエラー表示データとして、図 9 0 に示すエラーテーブル中のアドレス「 d E R R _ C R + 1 」に格納されている 1 バイトデータ「 0 1 0 0 1 1 1 0 B 」が取得され、下位桁の 7 セグ LED に出力するエラー表示データとして、アドレス「 d E R R _ C R + 2 」に格納されている 1 バイトデータ「 0 0 0 0 1 0 0 1 B 」が取得される。この場合、2 桁の 7 セグ LED には、「 C r 」の 2 文字がエラー情報として表示される。

10

【 0 6 4 6 】

次いで、メイン CPU 1 0 1 は、エラーコマンド (発生) 生成格納処理を行う (S 2 7 6)。この処理では、メイン CPU 1 0 1 は、副制御回路 2 0 0 に送信する、エラー発生時のエラーコマンドのデータを生成し、該コマンドデータをメイン RAM 1 0 3 に設けられた通信データ格納領域 (図 7 5 B 参照) に保存する。通信データ格納領域に保存されたエラー発生時のエラーコマンドは、後述の図 1 5 8 で説明する割込処理内の通信データ送信処理により、主制御回路 9 0 から副制御回路 2 0 0 に送信される。なお、エラー発生時のエラーコマンドには、エラー発生を示すパラメータを含んで構成される。

20

【 0 6 4 7 】

次いで、メイン CPU 1 0 1 は、1 割込時間 (1 . 1 1 7 2 m s) の待機処理を行う (S 2 7 7)。次いで、メイン CPU 1 0 1 は、エラーが解除されたか否かを判別する (S 2 7 8)。

【 0 6 4 8 】

S 2 7 8 において、メイン CPU 1 0 1 が、エラーが解除されていないと判別したとき (S 2 7 8 が N O 判定の場合)、メイン CPU 1 0 1 は、処理を S 2 7 7 の処理に戻し、S 2 7 7 以降の処理を繰り返す。

30

【 0 6 4 9 】

一方、S 2 7 8 において、メイン CPU 1 0 1 が、エラーが解除されたと判別したとき (S 2 7 8 が Y E S 判定の場合)、メイン CPU 1 0 1 は、エラー要因のクリア処理を行う (S 2 7 9)。なお、この処理は、メイン RAM 1 0 3 の規定外作業領域で行われる。次いで、メイン CPU 1 0 1 は、S 2 7 2 で退避させたメダルの払出枚数表示データの復帰処理を行う (S 2 8 0)。

【 0 6 5 0 】

次いで、メイン CPU 1 0 1 は、エラーコマンド (解除) 生成格納処理を行う (S 2 8 1)。この処理では、メイン CPU 1 0 1 は、副制御回路 2 0 0 に送信する、エラー解除時のエラーコマンドのデータを生成し、該コマンドデータをメイン RAM 1 0 3 に設けられた通信データ格納領域 (図 7 5 B 参照) に保存する。通信データ格納領域に保存されたエラー解除時のエラーコマンドは、後述の図 1 5 8 で説明する割込処理内の通信データ送信処理により、主制御回路 9 0 から副制御回路 2 0 0 に送信される。なお、エラー解除時のエラーコマンドには、エラー解除を示すパラメータを含んで構成される。そして、S 2 8 1 の処理後、メイン CPU 1 0 1 は、エラー処理を終了し、処理を例えばメダル投入チェック処理 (図 8 7 参照) 中の S 2 5 3 に移す。なお、エラー解除では、発生したエラー要因が解除され、リセットスイッチ 7 6 が押下されることにより、エラー状態が解除される。

40

【 0 6 5 1 】

[乱数取得処理]

次に、図 9 1 を参照して、メインフロー (図 8 2 参照) 中の S 2 0 3 で行う乱数取得処理

50

について説明する。なお、図 9 1 は、乱数取得処理の手順を示すフローチャートである。

【 0 6 5 2 】

まず、メイン CPU 1 0 1 は、乱数回路の乱数レジスタ 0 のハードラッチ乱数 (0 ~ 6 5 5 3 5) を取得し、取得した乱数値を内部当籤役抽籤用の乱数値として、メイン RAM 1 0 3 内の乱数値格納領域 (不図示) に保存する (S 2 9 1) 。

【 0 6 5 3 】

次いで、メイン CPU 1 0 1 は、乱数回路の乱数レジスタ 1 ~ 7 のソフトラッチ乱数 (0 ~ 6 5 5 3 5 : A R T 関連の抽籤処理で用いられる演出用乱数値、 0 ~ 2 5 5 : 1 バイト抽籤処理で乱数値) を生成するためのソフトラッチ乱数取得レジスタのセット処理を行う (S 2 9 2) 。次いで、メイン CPU 1 0 1 は、ソフトラッチ乱数の取得個数 (例えば、 7) をセットする (S 2 9 3) 。

10

【 0 6 5 4 】

次いで、メイン CPU 1 0 1 は、取得個数分のソフトラッチ乱数を一括で取得し、取得個数分のソフトラッチ乱数を乱数値格納領域に保存する (S 2 9 4) 。なお、この際、乱数回路 1 1 0 の乱数レジスタ 1 から取得されるソフトラッチ乱数 (演出用乱数値、 2 バイト乱数値) は、乱数値格納領域内において、乱数回路の乱数レジスタ 0 から取得されるハードラッチ乱数 (内部当籤役抽籤用の乱数値) が格納された領域とは異なる領域に保存される。そして、S 2 9 4 の処理後、メイン CPU 1 0 1 は、乱数取得処理を終了し、処理をメインフロー (図 8 2 参照) の S 2 0 4 に移す。なお、本実施形態では、4 つの 2 バイト乱数と、4 つの 1 バイト乱数を格納するために、メイン RAM 1 0 3 に 1 2 バイトの格納領域が乱数格納領域として割り当てられている。

20

【 0 6 5 5 】

[内部抽籤処理]

次に、図 9 2 ~ 図 9 6 を参照して、メインフロー (図 8 2 参照) 中の S 2 0 4 で行う内部抽籤処理について説明する。なお、図 9 2 は、内部抽籤処理の手順を示すフローチャートであり、図 9 3 A は、内部抽籤処理中の S 3 0 2 ~ S 3 0 5 の処理を実行するためのソースプログラムの一例を示す図であり、図 9 3 B は、内部抽籤処理中の S 3 0 8 ~ S 3 0 9 の処理を実行するためのソースプログラムの一例を示す図である。

【 0 6 5 6 】

また、図 9 4 は、内部抽籤処理のソースプログラム上で、実際に参照される内部抽籤テーブル (一般遊技中用) の構成の一例を示す図であり、図 9 5 は、内部抽籤処理のソースプログラム上で、実際に参照される R T 状態別抽籤値選択テーブルの構成の一例を示す図である。さらに、図 9 6 は、内部抽籤処理のソースプログラム上で、実際に参照される、内部抽籤値テーブル選択テーブル、1 バイト内部抽籤値テーブル、2 バイト内部抽籤値テーブル、1 バイト設定別内部抽籤値テーブル及び 2 バイト設定別内部抽籤値テーブルの構成の一例を示す図である。なお、本実施形態では、R B (B B) 中用の内部抽籤テーブルも設けられているが、ここでは、内部抽籤処理のソースプログラム上で参照される R B 中用の内部抽籤テーブルの構成の図示は省略する。

30

【 0 6 5 7 】

まず、メイン CPU 1 0 1 は、設定値・メダル投入枚数チェック処理を行う (S 3 0 1) 。この処理では、メイン CPU 1 0 1 は、現遊技の設定値 (1 ~ 6 のいずれか) 及びメダル投入枚数 (本実施形態では 3 枚) のチェック処理を行う。

40

【 0 6 5 8 】

次いで、メイン CPU 1 0 1 は、一般遊技中用の内部抽籤テーブル及び抽籤回数 (本実施形態では 5 3 回) をセットする (S 3 0 2) 。この処理では、図 9 4 に示す内部抽籤テーブル (一般遊技中用) 中の「特賞当籤番号 + 小役当籤番号」の値 (当り要求フラグステータス) が C レジスタにセットされ、「抽籤値選択テーブル or 抽籤係数テーブル」の値 (判定データ : アドレスに関するデータ) が A レジスタにセットされる。なお、当り要求フラグステータスは、図 9 4 に示すように、特賞当籤番号 (「 0 0 H (はずれ) 」、 「 0 1 H (B B 1) 」、 「 0 2 H (B B 2) 」 : 1 0 進数で 0、1、2) に 「 2 5 H (1 6 進数

50

: 10進数では37(後述の特賞番号)が乗算された値に、小役当籤番号(「00H」~「24H」:10進数で0~36)を加算した値である。

【0659】

次いで、メインCPU101は、RB作動中であるか否かを判別する(S303)。S303において、メインCPU101が、RB作動中ではないと判別したとき(S303がNO判定の場合)、メインCPU101は、後述のS305の処理を行う。

【0660】

一方、S303において、メインCPU101が、RB作動中であると判別したとき(S303がYES判定の場合)、メインCPU101は、RB中用の内部抽籤テーブル及び抽籤回数(本実施形態では5回)をセットする(S304)。この処理では、S302で

10

【0661】

S304の処理後又はS303がNO判定の場合、メインCPU101は、セットされている内部抽籤テーブルから抽籤対象役の判定データ(アドレスに関するデータ)を取得し、抽籤テーブルアドレスを更新する(S305)。

【0662】

次いで、メインCPU101は、判定データがRT状態別データであるか否かを判別する(S306)。この処理では、メインCPU101は、現在取得されている抽籤対象役がRT状態に応じて抽籤値が変化する内部当籤役であるか否かを判別する。具体的には、メインCPU101は、現在取得されている抽籤対象役の判定データに規定されているアドレスが、図95に示すRT状態別抽籤値選択テーブル内のアドレスであるか否かを判別する。

20

【0663】

例えば、図94の内部抽籤テーブルにおいて内部当籤役「F__チリリブ」に対応付けられている判定データ「(dRPPTRO1-dRTRB__SEL)*2+001H」では、図95に示すRT状態別抽籤値選択テーブル内の内部当籤役「F__チリリブ」のアドレス「dRPPTRO1」が規定されているので、内部当籤役「F__チリリブ」に対応付けられている判定データは、RT状態別データに対応する。それゆえ、現在取得されている抽籤対象役が内部当籤役「F__チリリブ」である場合には、S306の処理において、メインCPU101は、判定データがRT状態別データであると判定する。

30

【0664】

S306において、メインCPU101が、判定データがRT状態別データでないと判別したとき(S306がNO判定の場合)、メインCPU101は、後述のS308の処理を行う。一方、S306において、メインCPU101が、判定データがRT状態別データであると判別したとき(S306がYES判定の場合)、メインCPU101は、判定データに基づいて、図95に示すRT状態抽籤値選択テーブルから選択データを取得し、該取得した選択データを判定データにセットする(S307)。

【0665】

S307の処理後又はS306がNO判定の場合、メインCPU101は、抽籤対象役の判定データが設定別データであるか否かを判別する(S308)。この処理では、メインCPU101は、現在取得されている抽籤対象役が、設定値に応じて抽籤値が変化する内部当籤役であるか否かを判別する。具体的には、メインCPU101は、現在取得されている抽籤対象役の判定データに規定されているアドレスが、図96に示す1バイト設定別内部抽籤値テーブル又は2バイト設定別内部抽籤値テーブル内のアドレスであるか否かを判別する。

40

【0666】

例えば、図94の内部抽籤テーブルにおいて内部当籤役「F__強チリ1」に対応付けられている判定データ「((dNMLB00F289-dPRB__DB__WV)/06H)*2+080H」では、図96に示す1バイト設定別内部抽籤値テーブル内の内部当籤役「

50

「F__強チリ1」のアドレス「dNMLB00F289」が規定されているので、内部当籤役「F__強チリ1」に対応付けられている判定データは、設定別データに対応する。それゆえ、現在取得されている抽籤対象役が内部当籤役「F__強チリ1」である場合には、S308の処理において、メインCPU101は、判定データが設定別データであると判定する。

【0667】

S308において、メインCPU101が、判定データが設定別データでないと判別したとき（S308がNO判定の場合）、メインCPU101は、後述のS310の処理を行う。一方、S308において、メインCPU101が、判定データが設定別データであると判別したとき（S308がYES判定の場合）、メインCPU101は、判定データに設定値データ（0～5にいずれか）を加算し、該加算した値を判定データにセットする（S309）。なお、この処理で判定データに加算される設定値データは、設定値に対応付けられたデータであるが、設定値そのものの値ではなく、設定値データ「0」～「5」は、それぞれ「設定1」～「設定6」に対応するデータである。

10

【0668】

S309の処理後又はS308がNO判定の場合、メインCPU101は、セットされている判定データ（アドレスデータ）に基づいて、抽籤対象役の抽籤値が格納された領域のアドレスを算出し、該アドレスに格納された抽籤値を取得する（S310）。

【0669】

S310の処理において、例えば、抽籤対象役が、その抽籤値がRT状態及び設定値の両方に依存しない内部当籤役「F__サボ2」である場合には、図96に示す1バイト内部抽籤値テーブルからアドレス「dNM__B00F26」に格納された抽籤値「128」が取得される。また、例えば、抽籤対象役が、その抽籤値がRT状態により変化する内部当籤役「F__RT3リプ__1st」であり、RT状態がRT2状態である場合には、図96に示す2バイト内部抽籤値テーブルからアドレス「dRT2B00F13456」に格納された抽籤値「1800」が取得される。

20

【0670】

また、S310の処理において、例えば、抽籤対象役が、その抽籤値が設定値により変化する内部当籤役「F__強チリ1」である場合には、図96に示す1バイト設定別内部抽籤値テーブルに規定されている6種類の抽籤値「150（設定1）、150（設定2）、150（設定3）、150（設定4）、160（設定5）、170（設定6）」の中から設定値に対応する抽籤値が取得される。この際、設定値に対応する抽籤値の取得は、判定データに設定値データを加算（S309の処理）して求められたアドレスを指定することにより取得される。それゆえ、例えば、抽籤対象役が「F__強チリ1」であり、設定値が「6」（設定値データが「5」）である場合には、アドレス「dNMLB00F289+5」に格納された抽籤値「170」が取得される。

30

【0671】

なお、本実施形態では、例えば内部当籤役「F__維持リプA」のように、その抽籤値がRT状態及び設定値の両方に依存する役の場合には、内部抽籤値テーブル及び設定別内部抽籤値テーブルの両方を参照して、抽籤値が取得される。

40

【0672】

次いで、メインCPU101は、乱数格納領域に格納された内部当籤役抽籤用の乱数値（0～65535のいずれか）を取得する（S311）。

【0673】

次いで、メインCPU101は、抽籤実行処理を行う（S312）。この処理では、メインCPU101は、S310で取得された抽籤値に、S311で取得された乱数値を加算し、その加算結果を抽籤結果（抽籤対象役の当籤/非当籤）とする。なお、この抽籤実行処理において、抽籤値と乱数値との和が65535を超えた場合（オーバーフローした場合）、抽籤対象役が当籤した（抽籤対象役が内部当籤役として決定された）と判定される。

【0674】

50

次いで、メインCPU101は、乱数値に抽籤値を加算した値（抽籤実行後の乱数値）を新たな乱数値として、乱数格納領域に保存する（S313）。次いで、メインCPU101は、抽籤実行処理で当籤したか否か（オーバーフローが発生したか否か）を判別する（S314）。

【0675】

S314において、メインCPU101が、抽籤実行処理で当籤したと判別したとき（S314がYES判定の場合）、メインCPU101は、内部抽籤テーブルを参照して当籤した内部当籤役に対応する当り要求フラグステータス（「特賞当籤番号+小役当籤番号」の値）を取得する（S315）。例えば、一般遊技中において、抽籤対象役が「F_確チリリブ」であるときの抽籤実行処理で当籤した場合、S315の処理では、当り要求フラグステータス「(00H*25H)+02H」（特賞当籤番号=0、小役当籤番号=2）が取得される。そして、S315の処理後、メインCPU101は、内部抽籤処理を終了し、処理をメインフロー（図82参照）のS205に移す。

10

【0676】

一方、S314において、メインCPU101が、抽籤実行処理で当籤していないと判別したとき（S314がNO判定の場合）、メインCPU101は、内部抽籤テーブルにおいて抽籤対象役を次の役に更新し、抽籤回数を1減算する（S316）。次いで、メインCPU101は、減算後の抽籤回数が「0」であるか否かを判別する（S317）。

【0677】

S317において、メインCPU101が、減算後の抽籤回数が「0」でないと判別したとき（S317がNO判定の場合）、メインCPU101は、処理をS305の処理に戻し、S305以降の処理を繰り返す。

20

【0678】

一方、S317において、メインCPU101が、減算後の抽籤回数が「0」であると判別したとき（S317がYES判定の場合）、すなわち、内部当籤役が「はずれ」である場合、メインCPU101は、ハズレステータスをセットする（S318）。なお、「ハズレステータス」は、特賞当籤番号及び小役当籤番号のいずれもが「0」となる当り要求フラグステータスに対応する。そして、S318の処理後、メインCPU101は、内部抽籤処理を終了し、処理をメインフロー（図82参照）のS205に移す。

【0679】

本実施形態では、上述のようにして内部抽籤処理が行われる。なお、上述した内部抽籤処理中のS302～S305の処理は、メインCPU101が、図93Aのソースプログラムで規定されている各ソースコードを順次実行することにより行われる。その中でも、S305の判定データの取得処理は、図93A中のソースコード「LDIN AC, (HL)」により実行される。

30

【0680】

ソースプログラム上において、例えば、ソースコード「LDIN ss, (HL)」が実行されると、HLレジスタ（ペアレジスタ）にセットされたアドレス及び該アドレスに「1」を加算したアドレスで指定されるメモリの内容（データ）が、ss（BC、DE、AC、AE又はBD）ペアレジスタにロードされるとともに、HLレジスタにセットされているアドレスが+2更新（2加算）される。それゆえ、図93A中のソースコード「LDIN AC, (HL)」が実行されると、HLレジスタ（ペアレジスタ）にセットされたアドレス及び該アドレスに1加算したアドレスで指定されるメモリの内容（データ）が、ACレジスタにロードされるとともに、HLレジスタにセットされているアドレスが+2更新（2加算）される。なお、S305の判定データの取得処理では、上述のように、この「LDIN」命令（所定の読み出し命令）により、Aレジスタに、判定データ（「抽籤値選択テーブルor抽籤係数テーブル」の値）が格納され、Cレジスタに当り要求フラグステータス（「特賞当籤番号+小役当籤番号」の値）が格納される。

40

【0681】

上述のように、内部抽籤処理中のS305の判定データの取得処理では、一つの命令コー

50

ド(「LDIN」命令)により、データのロード処理及びアドレスの更新処理の両方を行うことができる。この場合、ソースプログラム上において、アドレス設定に係る命令コードを省略することができ、その分、ソースプログラムの容量(メインROM102の使用容量)を低減することができる。この結果、本実施形態では、メインROM102において、空き容量を確保する(増大させる)ことができ、増えた空き容量を活用して、遊技性を高めることが可能になる。

【0682】

また、上述した内部抽籤処理中のS308及びS309の処理は、メインCPU101が、図93Bのソースプログラムで規定されている各ソースコードを順次実行することにより行われる。その中で、S309の設定値データ(0~5のいずれか)の加算処理は、メインCPU101が図93B中のソースコード「MUL A, 6」及び「ADDQ A, (.LOW.wWAVENUM)」をこの順で実行することにより行われる。なお、「MUL」命令及び「ADDQ」命令はともに、メインCPU101専用命令コードであり、「ADDQ」命令は、Qレジスタ(拡張レジスタ)を用いてアドレス指定を行うメインCPU101専用命令コードである。

10

【0683】

ソースプログラム上において、例えば、ソースコード「MUL A, n」が実行されると、Aレジスタの格納データと、1バイトの整数nとを乗算し、その乗算結果をAレジスタに格納する。それゆえ、図93B中のソースコード「MUL A, 6」では、Aレジスタの内容(格納データ)に、1バイトの整数6が乗算され、その乗算結果がAレジスタに格納される。なお、この乗算処理は、マイクロプロセッサ91に含まれる演算回路107(図9参照)により実行される。すなわち、本実施形態のパチスロ1では、ソースプログラム上における乗算処理及び除算処理を実行するための演算専用回路(演算回路107)が設けられているので、乗算処理及び除算処理の効率化を図ることができる。

20

【0684】

また、ソースプログラム上において、例えば、ソースコード「ADDQ r, (k)」が実行されると、Qレジスタの格納データ(上位側アドレス値)及び1バイトの整数k(直値:下位側アドレス値)で指定されたアドレスのメモリの内容(格納データ)に、rレジスタ(A、B、C、D、E、H又はLレジスタ)の格納データが加算され、該加算結果がrレジスタに格納される。それゆえ、図93B中のソースコード「ADDQ A, (.LOW.wWAVENUM)」が実行されると、Qレジスタの格納データ及び1バイトの整数値「.LOW.wWAVENUM」で指定されたアドレスのメモリの内容(設定値データ)にAレジスタの内容(格納データ)が加算され、該加算結果がAレジスタに格納される。

30

【0685】

すなわち、図93Bに示す例では、S309の設定値の加算処理において、抽籤テーブル選択用相対値に係数「6」を乗算して、その乗算値に設定値データを加算することにより、抽籤対象役の抽籤値が格納された抽籤テーブルのアドレスを算出している。

【0686】

上述のように、本実施形態では、内部抽籤処理において、Qレジスタ(拡張レジスタ)を用いたメインCPU101専用命令コード(「ADDQ」命令)が用いられており、この命令コードを用いれば、直値により、メインROM102、メインRAM103やメモリアクセスすることができる。それゆえ、内部抽籤処理のソースプログラム上において、アドレス設定に係る命令を省略することができ、その分、ソースプログラムの容量(メインROM102の使用容量)を低減することができる。この結果、本実施形態では、メインROM102において、空き容量を確保する(増大させる)ことができ、増えた空き容量を活用して、遊技性を高めることが可能になる。

40

【0687】

[図柄設定処理]

次に、図97~図100を参照して、メインフロー(図82参照)中のS205で行う図

50

柄設定処理について説明する。

【0688】

図97は、図柄設定処理の手順を示すフローチャートである。図98は、特賞（ボーナス）当籤番号及び小役当籤番号と、内部当籤役との対応表である。なお、図98では、「はずれ（00）」に対応する特賞当籤番号及び小役当籤番号の図示は省略している。また、図99は、図柄設定処理中のS324～S330の処理を実行するためのソースプログラムの一例を示す図であり、図100は、図柄設定処理のソースプログラム上で、実際に参照される当り要求フラグテーブル（フラグデータテーブル、当籤フラグテーブルデータ）の構成の一例を示す図である。

【0689】

まず、メインCPU101は、内部抽籤処理で取得された当り要求フラグステータスに基づいて、特賞当籤番号及び小役当籤番号を抽出し、該抽出された特賞当籤番号及び小役当籤番号をメインRAM103内の当籤番号格納領域（不図示）に保存する（S321）。

【0690】

本実施形態では、図98に示すように、特賞（ボーナス）当籤番号「1」及び「2」には、それぞれ、内部当籤役「F__BB1」及び「F__BB2」が対応付けられている。また、小役当籤番号「1」～「36」には、それぞれ、内部当籤役「F__チリリブ」～「F__RB役4」が対応付けられている。そして、図94で説明したように、当り要求フラグステータスの値は、特賞当籤番号に特賞番号（本実施形態では「37（16進数では25H）」）を乗算した値に、小役当籤番号を加算した値である。それゆえ、S321の処理において、当り要求フラグステータスの値から特賞当籤番号及び小役当籤番号を抽出するため、本実施形態では、メインCPU101は、当り要求フラグステータスの値を特賞番号（「37」）で除算する。その結果、除算処理により生成された、商の値が特賞当籤番号（10進数で0～2のいずれか）となり、余りの値が小役当籤番号（10進数で0～36のいずれか）となる。

【0691】

次いで、メインCPU101は、抽出された小役当籤番号に基づいて、小役が当籤したか否かを判別する（S322）。この処理において、小役当籤番号が1～36のいずれかである場合には、メインCPU101は、小役が当籤したと判定し、小役当籤番号が0である場合には、メインCPU101は、小役が当籤しなかったと判定する。

【0692】

S322において、メインCPU101が、小役が当籤していないと判別したとき（S322がNO判定の場合）、メインCPU101は、後述のS331の処理を行う。一方、S322において、メインCPU101が、小役が当籤したと判別したとき（S322がYES判定の場合）、メインCPU101は、小役当籤番号を減算結果の初期値としてセットする（S323）。

【0693】

次いで、メインCPU101は、当り要求フラグテーブル（図100参照）をセットする（S324）。次いで、メインCPU101は、減算結果を1減算し、該減算結果を更新する（S325）。次いで、メインCPU101は、減算結果が「0」未満であるか否かを判別する（S326）。

【0694】

S326において、メインCPU101が、減算結果が「0」未満でないと判別したとき（S326がNO判定の場合）、メインCPU101は、ビット数算出処理を行う（S327）。なお、S327のビット数算出処理では、当り要求フラグテーブルに規定されている、小役当籤番号に対応する当り要求フラグデータの格納領域のブロック数を取得する。

【0695】

なお、本実施形態では、当り要求フラグ格納領域（内部当籤役格納領域）において、当り要求格納領域0～7のブロックと、当り要求格納領域8～11のブロックとが設けられている。それゆえ、S327のビット数算出処理で取得される当り要求フラグデータの格納

10

20

30

40

50

領域のブロック数の最大値は「2」となる。例えば、内部当籤役が「F__確チリリブ」である場合には、当り要求フラグテーブル（図100参照）に示すように、当り要求格納領域0～7のブロックに含まれる格納領域7と、当り要求格納領域8～11のブロックに含まれる格納領域9にそれぞれ当り要求フラグデータが規定されているので、S327のビット数算出処理で取得される当り要求フラグデータの格納領域のブロック数は「2」となる。

【0696】

次いで、メインCPU101は、ビット数算出処理を行う（S328）。なお、S328のビット数算出処理では、当り要求フラグテーブル（図100参照）において規定されるブロック単位の当り要求フラグデータのバイト数を算出する。例えば、内部当籤役が「F__確チリリブ」である場合には、当り要求フラグテーブル（図100参照）に示すように格納領域7及び格納領域9ではともに1バイトの当り要求フラグデータが格納されるので、S328のビット数算出処理で取得されるブロック単位の当り要求フラグデータのバイト数は1バイトとなる。なお、図100に記載のテーブルにおいて、格納領域7に格納される当り要求フラグデータには「10000000B | 01000000B」と記載されているが、これは、格納領域7に格納される当り要求フラグデータが「10000000B」又は（「|」は論理和の記号）「01000000B」であることを意味する。

10

【0697】

なお、上述したS325～S328の処理は、小役当籤番号の回数だけ繰り返される。例えば、内部当籤役が「F__確チリリブ」（小役当籤番号が「2」）である場合には、上述したS325～S328の処理は、2回繰り返される。また、S325～S328の処理が複数回繰り返される場合には、S327及びS328のビット数算出処理でそれぞれ取得されるブロック数及びブロック単位の当り要求フラグデータのバイト数は、別の格納領域に保存される。また、上述したS325～S328の処理により得られたブロック数及びブロック単位の当り要求フラグデータのバイト数は、当り要求フラグデータの格納先を指定する情報（オンビット情報）となる。

20

【0698】

ここで再度、S326の処理に戻って、S326において、メインCPU101が、減算結果が「0」未満であると判別したとき（S326がYES判定の場合）、メインCPU101は、当り要求フラグ格納領域（内部当籤役格納領域）のセット処理を行う（S329）。この際、メインCPU101は、上述したS325～S328の処理により得られたブロック数及びブロック単位の当り要求フラグデータのバイト数（オンビット情報）に基づいて、チェック（更新）対象となる当り要求フラグ格納領域のみをセットする。具体的には、チェック（更新）対象となる当り要求フラグ格納領域のアドレスをDレジスタに格納する（図99参照）。

30

【0699】

次いで、メインCPU101は、圧縮データ格納処理を行う（S330）。この処理では、メインCPU101は、主に、当り要求フラグデータをチェック（更新）対象となる当り要求フラグ格納領域内の所定の格納領域に転送（展開）する処理を行う。圧縮データ格納処理の詳細については、後述の図101を参照しながら後で説明する。

40

【0700】

S330の処理後又はS322がNO判定の場合、メインCPU101は、持越役格納領域（図31参照）を参照して、持越役があるか否かを判別する（S331）。S331において、メインCPU101が、持越役があると判別したとき（S331がYES判定の場合）、メインCPU101は、後述のS334の処理を行う。

【0701】

一方、S331において、メインCPU101が、持越役がないと判別したとき（S331がNO判定の場合）、メインCPU101は、S321の処理で抽出された特賞当籤番号に基づいて、ボーナス役（BB1又はBB2）が当籤したか否かを判別する（S332）。

50

【0702】

S332において、メインCPU101が、ボーナス役が当籤していないと判別したとき（S332がNO判定の場合）、メインCPU101は、図柄判定処理を終了し、処理をメインフロー（図82参照）のS206に移す。

【0703】

一方、S332において、メインCPU101が、ボーナス役が当籤したと判別したとき（S332がYES判定の場合）、メインCPU101は、当籤した特賞当籤番号を持越役格納領域に格納する（S333）。

【0704】

S333の処理後又はS331がNO判定の場合、メインCPU101は、特賞当籤番号を当籤番号格納領域（不図示）にセットし、当り要求フラグ格納領域に当り要求フラグデータをセットし、RT状態をRT5状態にセットし、RT遊技数（RT1状態の消化ゲーム数）をクリア（「0」）する（S334）。そして、S334の処理後、メインCPU101は、図柄設定処理を終了し、処理をメインフロー（図82参照）のS206に移す。

10

【0705】

本実施形態では、上述のようにして図柄設定処理が行われる。上述した図柄設定処理中のS324～S330の処理（入賞に係るデータの圧縮・展開処理）は、メインCPU101が、図99のソースプログラムで規定されている各ソースコードを順次実行することにより行われる。その中で、S330の圧縮データ格納処理は、メインCPU101が図99中のソースコード「CALLF SB__BTEP__00」を実行することにより行われる。

20

【0706】

「CALLF」命令は、上述のようにメインCPU101専用の2バイト命令コードであり、図99中のソースコード「CALLF SB__BTEP__00」が実行されると、「SB__BTEP__00」で指定されているアドレスに、処理をジャンプさせ、圧縮データ格納処理が開始される。なお、S330の圧縮データ格納処理では、上述のように、当り要求フラグテーブルに格納された当り要求フラグデータ（圧縮データ）が、対応する当り要求フラグ格納領域に展開（コピー）される。

【0707】

また、上述した図柄設定処理中のS329の当り要求フラグ格納領域のアドレスのセット処理は、メインCPU101が図99中のソースコード「LDQ DE, .LOW.wWAVEBIT」を実行することにより行われる。すなわち、図柄設定処理中のS329の処理は、Qレジスタ（拡張レジスタ）を用いてアドレス指定を行うメインCPU101専用の「LDQ」命令により行われる。この場合、図柄設定処理のソースプログラム上において、アドレス設定に係る命令コードを省略することができ、その分、図柄設定処理のソースプログラムの容量（メインROM102の使用容量）を低減することができる。この結果、本実施形態では、メインROM102において、空き容量を確保する（増大させる）ことができ、増えた空き容量を活用して、遊技性を高めることが可能になる。

30

【0708】

さらに、本実施形態では、上述した図柄設定処理中のS324～S330で説明した処理手順で入賞に係るデータの圧縮・展開処理を行い、かつ、その処理の中で上述したメインCPU101専用命令コードを用いることにより、入賞に係るデータの圧縮・展開処理の効率化を図ることができるとともに、限られたメインRAM103の容量を有効活用することができる。

40

【0709】

[圧縮データ格納処理]

次に、図101を参照して、例えば、図柄判定処理（図97参照）中のS330で行う圧縮データ処理について説明する。図101は、圧縮データ格納処理の手順を示すフローチャートである。

【0710】

50

なお、図 101 に示す圧縮データ格納処理は、図柄判定処理（図 97 参照）中の S330 だけでなく、後述の図柄コード取得処理（後述の図 128 参照）中の S649 においても実行される。図柄判定処理（図 97 参照）中の S330 で実行される圧縮データ格納処理では、処理対象となるフラグデータは当り要求フラグデータ（当籤役に係るフラグデータ）となるが、後述の図柄コード取得処理（後述の図 128 参照）中の S649 で実行される圧縮データ格納処理では、処理対象となるフラグデータは入賞作動フラグデータ（入賞役に係るフラグデータ）である。そして、処理対象となるフラグデータの種別が異なること以外は、両者の処理は同じ処理になる。

【0711】

それゆえ、図 101 のフローチャートでは、処理対象とするフラグデータを「処理対象フラグデータ」と記し、処理対象となるフラグテーブルを「処理対象フラグテーブル」と記す。また、この記載に合わせて、以下の圧縮データ格納処理の説明においても、当り要求フラグデータ又は入賞作動フラグデータを「処理対象フラグデータ」と称し、当り要求フラグテーブル（図 100 参照）又は後述の図柄対応入賞作動テーブル（例えば、後述の図 130A 等参照）を「処理対象フラグテーブル」と称す。

【0712】

まず、メイン CPU 101 は、格納先チェックビットをセットする（S341）。この処理では、格納先チェックビットは A レジスタ以外のレジスタに格納される。

【0713】

格納先チェックビットは、処理対象フラグデータの格納先（転送先）となるブロックを指定するための 1 バイトのデータである。本実施形態では、当り要求フラグ格納領域及び入賞作動フラグ格納領域はともに、2 つのブロック（格納領域 0 ~ 7 のブロック及び格納領域 8 ~ 11 のブロック）で構成される。そして、例えば、内部当籤役「F_確チリリブ」が決定された場合には、図 100 の当り要求フラグテーブルに示すように格納領域 7 及び格納領域 9 のそれぞれに当り要求フラグデータが格納されるので（格納先のブロック数が「2」になるので）、S341 の処理では、格納先チェックビットとして、「00000011B」がセットされる。なお、この 1 バイトのデータのビット 0 の値（1/0）が格納領域 0 ~ 7 のブロック内の格納先の有無に対応し、ビット 1 の値（1/0）が格納領域 8 ~ 11 のブロック内の格納先の有無に対応する。

【0714】

次いで、メイン CPU 101 は、バイト単位の転送カウンタの値を「8」にセットする（S342）。本実施形態では、各ブロックのバイト数が「8」であるので、転送カウンタの初期値には「8」がセットされる。

【0715】

次いで、格納先チェックビットから転送指示ビットの値を抽出する（S343）。なお、転送指示ビットは、格納先チェックビット内のビット 0 のデータに対応し、S343 の処理では、1 バイトのレジスタに格納されている格納先チェックビットを 1 回（1 ビット分）右シフトすることにより、転送指示ビットが抽出される。具体的には、格納先チェックビットが格納された 1 バイトのレジスタ（A レジスタ以外のレジスタ）を 1 回右シフトすると、ビット 7 ~ ビット 1 に格納されているデータがそれぞれビット 6 ~ ビット 0 に移動するとともに、シフト前のビット 0 のデータが出力される。そして、このシフト処理により出力されたデータが転送指示ビットの値となる。

【0716】

次いで、メイン CPU 101 は、抽出された転送指示ビットの値に基づいて、転送指示があるか否かを判別する（S344）。この処理では、メイン CPU 101 は、抽出された転送指示ビットの値が「1」である場合に転送指示があると判定する。例えば、格納先チェックビットとして、「00000011B」がセットされた場合、1 回目（格納領域の 1 ブロック目に対応）及び 2 回目（格納領域の 2 ブロック目に対応）の S344 の判定処理では、転送指示ありの判定となるが、3 回目以降の S344 の判定処理では、転送指示なしの判定となる。

10

20

30

40

50

【 0 7 1 7 】

S 3 4 4 において、メイン CPU 1 0 1 が、転送指示がないと判別したとき (S 3 4 4 が NO 判定の場合)、メイン CPU 1 0 1 は、後述の S 3 5 4 の処理を行う。

【 0 7 1 8 】

一方、S 3 4 4 において、メイン CPU 1 0 1 が、転送指示があると判別したとき (S 3 4 4 が YES 判定の場合)、メイン CPU 1 0 1 は、処理対象フラグテーブルからバイト単位格納先指定情報を取得する (S 3 4 5)。この処理では、バイト単位格納先指定情報として、処理対象フラグテーブル内の処理対象役 (当籤役又は入賞役) のフラグデータが格納された領域の先頭アドレスに格納されている、転送先を示す 1 バイトのデータが取得される。例えば、内部当籤役が「 F __ 確チリリブ」である場合には、図 1 0 0 に示す当り要求フラグテーブル内の「 F __ 確チリリブ」のフラグデータが格納された領域の先頭アドレスに格納されている、格納領域 7 を転送先として指定する 1 バイトデータ「 1 0 0 0 0 0 0 0 B 」がバイト単位格納先指定情報として取得される。

10

【 0 7 1 9 】

次いで、メイン CPU 1 0 1 は、処理対象フラグテーブル内で参照するアドレスの更新処理 (アドレスを 1 加算する処理) を行う (S 3 4 6)。また、この処理では、メイン CPU 1 0 1 は、処理対象フラグデータの格納 (転送) 先となるブロックの先頭格納領域を指定するアドレスを初期アドレスとしてセットする。例えば、1 ブロック目の処理では、S 3 4 6 の処理において、初期アドレスとして格納領域 0 のアドレスがセットされ、2 ブロック目の処理では、S 3 4 6 の処理において、初期アドレスとして格納領域 8 のアドレスがセットされる。

20

【 0 7 2 0 】

次いで、メイン CPU 1 0 1 は、バイト単位格納先指定情報から転送指示ビットの値を抽出する (S 3 4 7)。なお、ここでいう転送指示ビットは、バイト単位格納先指定情報のビット 0 に対応し、S 3 4 7 の処理では、1 バイトのレジスタに格納されているバイト単位格納先指定情報を 1 回右シフトすることにより、転送指示ビットの値を抽出する (ビット 0 のデータを出力する)。

【 0 7 2 1 】

次いで、メイン CPU 1 0 1 は、S 3 4 7 の処理で抽出された転送指示ビットの値に基づいて、転送指示があるか否かを判別する (S 3 4 8)。この処理では、メイン CPU 1 0 1 は、抽出された転送指示ビットの値が「 1 」である場合、転送指示があると判定する。例えば、バイト単位格納先指定情報として、「 0 0 0 0 0 0 1 0 B 」がセットされた場合、2 回目 (1 ブロック目の格納領域 1 又は 2 ブロック目の格納領域 9) の S 3 4 7 の処理でビット 1 のデータ「 1 」が転送指示ビットの値として出力され転送指示ありの判定となるが、1 回目及び 3 ~ 8 回目の S 3 4 7 の処理では、転送指示なしの判定となる。

30

【 0 7 2 2 】

S 3 4 8 において、メイン CPU 1 0 1 が、転送指示がないと判別したとき (S 3 4 8 が NO 判定の場合)、メイン CPU 1 0 1 は、後述の S 3 5 1 の処理を行う。

【 0 7 2 3 】

一方、S 3 4 8 において、メイン CPU 1 0 1 が、転送指示があると判別したとき (S 3 4 8 が YES 判定の場合)、メイン CPU 1 0 1 は、現在セットされている処理対象フラグテーブル内のアドレスに格納されている処理対象フラグデータ (当り要求フラグデータ又は入賞作動フラグデータ) を、指定された格納領域に転送 (コピー) する (S 3 4 9)。

40

【 0 7 2 4 】

例えば、内部当籤役が「 F __ 確チリリブ」であり、現在の処理が 1 ブロック目の格納領域 (格納領域 0 ~ 7) に対して行われている場合には、バイト単位格納先指定情報が「 1 0 0 0 0 0 0 0 B 」 (格納領域 7 を格納先として指定するデータ) となるので、8 回目の S 3 4 7 の処理で転送指示があると判定され、その後の S 3 4 9 の処理で、当り要求フラグデータ「 1 0 0 0 0 0 0 0 B 」、「 0 1 0 0 0 0 0 0 B 」、「 0 0 1 0 0 0 0 0 B 」及び「 0 0 0 1 0 0 0 0 B 」のいずれかが、当り要求フラグ格納領域の格納領域 7 に転送 (コ

50

ピー)される。

【0725】

次いで、メインCPU101は、処理対象フラグテーブル内で参照するアドレスの更新処理(アドレスを1加算する処理)を行う(S350)。

【0726】

S350の処理後又はS348がNO判定の場合、メインCPU101は、処理対象フラグデータの格納先となる格納領域を指定するアドレスの更新処理(アドレスを1加算する処理)を行う(S351)。次いで、メインCPU101は、転送カウンタの値を1減算する(S352)。

【0727】

次いで、メインCPU101は、転送カウンタの値が「0」であるか否かを判別する(S353)。S353において、メインCPU101が、転送カウンタの値が「0」でないと判別したとき(S353がNO判定の場合)、メインCPU101は、処理をS347の処理に戻し、S347以降の処理を繰り返す。

【0728】

一方、S353において、メインCPU101が、転送カウンタの値が「0」であると判別したとき(S353がYES判定の場合)、メインCPU101は、現在の格納先チェックビットに転送指示対象が残っているか否かを判別する(S354)。この処理では、メインCPU101は、現処理時点において、格納先チェックビット内に「1」が格納されているビットが残っているか否かを判別する。そして、メインCPU101は、格納先チェックビット内に「1」が格納されているビットが残っている場合、すなわち、処理対象となるブロックが存在する場合には、現在の格納先チェックビットに転送指示対象が残っていると判定する。

【0729】

S354において、メインCPU101が、現在の格納先チェックビットに転送指示対象が残っていると判別したとき(S354がYES判定の場合)、メインCPU101は、処理をS342の処理に戻し、S342以降の処理を繰り返す。一方、S354において、メインCPU101が、現在の格納先チェックビットに転送指示対象が残っていないと判別したとき(S354がNO判定の場合)、メインCPU101は、圧縮データ格納処理を終了し、処理を例えば図柄判定処理(図97参照)中のS331に移す。

【0730】

[第2インターフェースボード制御処理(規定外)]

次に、図102を参照して、メインフロー(図82参照)中のS207で行う第2インターフェースボード制御処理について説明する。図102は、第2インターフェースボード制御処理の手順を示すフローチャートである。なお、この処理は、メインRAM103内の規定外作業領域(図12C参照)で行われる。また、この第2インターフェースボード制御処理で用いられるプログラムはメインROM102内の規定外エリアに格納されている(図12B参照)。

【0731】

まず、メインCPU101は、スタックポインタ(SP)にセットされているメインRAM103内のスタックエリアのアドレスデータを退避させる(S361)。次いで、メインCPU101は、メインRAM103内の規定外スタックエリアのアドレスデータをスタックポインタ(SP)にセットする(S362)。

【0732】

次いで、メインCPU101は、ナビデータを取得する(S363)。次いで、メインCPU101は、ナビ変換テーブルをメインRAM103内の規定外作業領域にセットする(S364)。

【0733】

次いで、メインCPU101は、ナビ変換テーブルを参照して第2インターフェース用押し順番号を取得する(S365)。次いで、メインCPU101は、取得した第2インタ

10

20

30

40

50

ーフェース用押し順番号を、規定外作業領域に設けられた規定外押し順番号格納領域（不図示）に格納する（S366）。次いで、メインCPU101は、規定外作業領域に設けられた押下位置テーブル選択カウンタの値に「0」をセットする（S367）。

【0734】

次いで、メインCPU101は、取得したナビデータが押し順ナビ（押し順小役用のナビデータ）であるか否かを判別する（S368）。S368において、メインCPU101が、取得したナビデータが押し順ナビであると判別したとき（S368がYES判定の場合）、メインCPU101は、後述のS372の処理を行う。

【0735】

一方、S368において、メインCPU101が、取得したナビデータが押し順ナビでない
と判別したとき（S368がNO判定の場合）、メインCPU101は、取得したナビ
データがBB1停止操作用のナビデータ（10）であるか否かを判別する（S369）。 10

【0736】

S369において、メインCPU101が、取得したナビデータがBB1停止操作用のナビ
データであると判別したとき（S369がYES判定の場合）、メインCPU101は
、後述のS371の処理を行う。一方、S369において、メインCPU101が、取得
したナビデータがBB1停止操作用のナビデータでない
と判別したとき（S369がNO
判定の場合）、メインCPU101は、押下位置テーブル選択カウンタの値に「1」を加
算する（S370）。S370において、押下位置テーブル選択カウンタの値に「1」を
加算する処理は、押下位置テーブル（不図示）からBB2の押下位置を取得するために行
われる処理である。 20

【0737】

S370の処理後又はS369がYES判定の場合、メインCPU101は、押下位置テ
ーブル選択カウンタの値に「1」を加算する（S371）。S371において、押下位置
テーブル選択カウンタの値に「1」を加算する処理は、押下位置テーブル（不図示）から
BB1又はBB2の押下位置を取得するために行われる処理である。

【0738】

S371の処理後又はS368がYES判定の場合、メインCPU101は、押下位置テ
ーブル選択カウンタの値に基づいて、押下位置テーブル（不図示）を選択する（S372
）。次いで、メインCPU101は、選択した押下位置テーブルを参照して、3リール分
（左リール3L、中リール3C及び右リール3R）の押下位置データを取得する（S37
3）。次いで、メインCPU101は、取得した押下位置データを規定外作業領域に設け
られた規定外押下位置格納領域（不図示）に格納する（S374）。S367～S371
の処理により、ナビデータが押し順ナビであれば、押下位置テーブル選択カウンタの値は
「0」となり、ナビデータがBB1停止操作用のナビデータであれば、押下位置テ
ーブル選択カウンタの値は「1」となり、ナビデータがBB2停止操作用のナビデータ
であれば、押下位置テーブル選択カウンタの値は「2」となる。すなわち、ナビデータに基づいて
、押下位置データが取得される。

【0739】

次いで、メインCPU101は、第2インターフェースボード出力処理を行う（S375
）。なお、第2インターフェースボード出力処理の詳細については、後述の図103を参
照しながら後で説明する。 40

【0740】

次いで、メインCPU101は、全レジスタの復帰処理を行う（S376）。次いで、メ
インCPU101は、S361で退避させたスタックエリアのアドレスデータをスタック
ポインタ（SP）にセットする（S377）。そして、S377の処理後、メインCPU
101は、第2インターフェースボード制御処理を終了し、処理をメインフロー（図82
参照）のS208に移す。

【0741】

[第2インターフェースボード出力処理]

次に、図 103 を参照して、第 2 インターフェースボード制御処理（図 102 参照）中の S 375 で行う第 2 インターフェースボード出力処理について説明する。図 103 は、第 2 インターフェースボード出力処理の手順を示すフローチャートである。なお、この第 2 インターフェースボード出力処理は、メイン RAM 103 の規定外作業領域で行われる。

【0742】

まず、メイン CPU 101 は、第 2 インターフェース用シリアル回線（第 2 シリアル通信回路 115：SCU2）を介して送信動作が行われているか否かを判別する（S 381）。S 381 において、メイン CPU 101 が、第 2 インターフェース用シリアル回線を介して送信動作が行われていると判別したとき（S 381 が YES 判定の場合）、メイン CPU 101 は、第 2 インターフェースボード出力処理を終了し、処理を第 2 インターフェースボード制御処理（図 102 参照）の S 376 に移す。

10

【0743】

一方、S 381 において、メイン CPU 101 が、第 2 インターフェース用シリアル回線を介して送信動作が行われていないと判別したとき（S 381 が NO 判定の場合）、メイン CPU 101 は、規定外作業領域に設けられたループカウンタの値に「3」（リールの個数）をセットし、シリアル通信用サム値に初期値「1」をセットする（S 382）。次いで、メイン CPU 101 は、第 2 インターフェース用シリアル回線（第 2 シリアル通信回路 115：SCU2）を介して、送信開始データを送信する（S 383）。

【0744】

次いで、メイン CPU 101 は、所定のリール（回胴）の規定外押下位置格納領域を参照し、所定のリールの押下位置データを取得する（S 384）。次いで、メイン CPU 101 は、参照する規定外押下位置格納領域を次の対象リール（回胴）のそれに更新する（S 385）。

20

【0745】

次いで、メイン CPU 101 は、パルス変換データ（不図示）及び取得した押下位置データに基づいて、押下位置データ（図柄位置）に対応するパルス数データを取得する（S 386）。なお、押下位置データ（図柄位置）とパルス数データとの対応関係の詳細については省略するが、例えば、取得した押下位置データ（図柄位置）が「3」（左リール 3L では図柄「白 7」）である場合には、パルス数データとして「38」が取得され、押下位置データ（図柄位置）が「10」（左リール 3L では図柄「リプレイ」）である場合には、パルス数データとして「155」が取得される。また、例えば、取得した押下位置データ（図柄位置）が「12」（左リール 3L では図柄「青 7」）である場合には、パルス数データとして「189」が取得され、押下位置データ（図柄位置）が「15」（左リール 3L では図柄「リプレイ」）である場合には、パルス数データとして「239」が取得される。

30

【0746】

次いで、メイン CPU 101 は、取得したパルス数データを第 2 インターフェース用シリアル回線（第 2 シリアル通信回路 115：SCU2）を介して送信する（S 387）。次いで、メイン CPU 101 は、シリアル通信用サム値にパルス数データを加算する（S 388）。次いで、メイン CPU 101 は、ループカウンタの値を 1 減算する（S 389）。

40

【0747】

次いで、メイン CPU 101 は、ループカウンタの値が「0」であるか否かを判別する（S 390）。S 390 において、メイン CPU 101 が、ループカウンタの値が「0」でないと判別したとき（S 390 が NO 判定の場合）、メイン CPU 101 は、対象リールを次のリールに変更するとともに、処理を S 384 に戻し、S 384 以降の処理を繰り返す。

【0748】

一方、S 390 において、メイン CPU 101 が、ループカウンタの値が「0」であると判別したとき（S 390 が YES 判定の場合）、メイン CPU 101 は、シリアル通信用サム値を第 2 インターフェース用シリアル回線（第 2 シリアル通信回路 115：SCU2

50

)を介して送信する(S 3 9 1)。そして、S 3 9 1の処理後、メインCPU 1 0 1は、第2インターフェースボード出力処理を終了し、処理を第2インターフェースボード制御処理(図1 0 2参照)のS 3 7 6に移す。

【0 7 4 9】

[状態別制御処理]

次に、図1 0 4を参照して、メインフロー(図8 2参照)中のS 2 0 8で行う状態別制御処理について説明する。図1 0 4は、状態別制御処理の手順を示すフローチャートである。

【0 7 5 0】

まず、メインCPU 1 0 1は、サブフラグ変換処理を行う(S 4 0 1)。この処理では、メインCPU 1 0 1は、内部当籤役をサブフラグ(図3 6及び図3 7参照)に変換する処理を行う。なお、サブフラグ変換処理の詳細については、後述の図1 0 5を参照しながら後で説明する。

10

【0 7 5 1】

次いで、メインCPU 1 0 1は、ナビセット処理を行う(S 4 0 2)。この処理では、メインCPU 1 0 1は、RT状態、遊技状態及び小役当籤番号に基づいてナビデータを取得する。なお、ナビセット処理の詳細については、後述の図1 0 8を参照しながら後で説明する。

【0 7 5 2】

次いで、メインCPU 1 0 1は、現在のRT状態がRT 4状態であるか否かを判別する(S 4 0 3)。S 4 0 3において、メインCPU 1 0 1が、現在のRT状態がRT 4状態でないとは判別したとき(S 4 0 3がNO判定の場合)、メインCPU 1 0 1は、後述のS 4 0 6の処理を行う。

20

【0 7 5 3】

一方、S 4 0 3において、メインCPU 1 0 1が、現在のRT状態がRT 4状態であると判別したとき(S 4 0 3がYES判定の場合)、メインCPU 1 0 1は、フラグ変換処理を行う(S 4 0 4)。この処理では、メインCPU 1 0 1は、サブフラグをサブフラグEX(図3 6参照)に変換するためのフラグ変換抽籤処理(サブフラグデータの圧縮処理)を行う。このフラグ変換処理により、1 9種類(ハズレも含む)のサブフラグが、9種類(ハズレも含む)のサブフラグEXに変換(圧縮)される。なお、フラグ変換処理の詳細については、後述の図1 1 1を参照しながら後で説明する。

30

【0 7 5 4】

次いで、メインCPU 1 0 1は、サブフラグ圧縮処理を行う(S 4 0 5)。この処理では、メインCPU 1 0 1は、サブフラグEXをサブフラグD(図3 6参照)に変換し、サブフラグデータのさらなる圧縮処理を行う。このサブフラグ圧縮処理により、9種類(ハズレも含む)のサブフラグEXが、7種類(ハズレも含む)のサブフラグDに変換(圧縮)される。

【0 7 5 5】

S 4 0 5の処理後又はS 4 0 3がNO判定の場合、メインCPU 1 0 1は、現在の遊技状態が通常遊技状態であるか否かを判別する(S 4 0 6)。

【0 7 5 6】

S 4 0 6において、メインCPU 1 0 1が、現在の遊技状態が通常遊技状態であると判別したとき(S 4 0 6がYES判定の場合)、メインCPU 1 0 1は、通常中スタート時処理を行う(S 4 0 7)。なお、通常中スタート時処理の詳細については、後述の図1 1 2を参照しながら後で説明する。そして、S 4 0 7の処理後、メインCPU 1 0 1は、状態別制御処理を終了し、処理をメインフロー(図8 2参照)のS 2 0 9に移す。

40

【0 7 5 7】

一方、S 4 0 6において、メインCPU 1 0 1が、現在の遊技状態が通常遊技状態でないと判別したとき(S 4 0 6がNO判定の場合)、メインCPU 1 0 1は、現在の遊技状態がC Zであるか否かを判別する(S 4 0 8)。

【0 7 5 8】

50

S 4 0 8において、メインCPU 1 0 1が、現在の遊技状態がC Zであると判別したとき（S 4 0 8がYES判定の場合）、メインCPU 1 0 1は、C Z中スタート時処理を行う（S 4 0 9）。なお、C Z中スタート時処理の詳細については、後述の図 1 1 3を参照しながら後で説明する。そして、S 4 0 9の処理後、メインCPU 1 0 1は、状態別制御処理を終了し、処理をメインフロー（図 8 2 参照）のS 2 0 9に移す。

【 0 7 5 9 】

一方、S 4 0 8において、メインCPU 1 0 1が、現在の遊技状態がC Zでないと判別したとき（S 4 0 8がNO判定の場合）、メインCPU 1 0 1は、現在の遊技状態が通常ARTであるか否かを判別する（S 4 1 0）。

【 0 7 6 0 】

S 4 1 0において、メインCPU 1 0 1が、現在の遊技状態が通常ARTであると判別したとき（S 4 1 0がYES判定の場合）、メインCPU 1 0 1は、通常ART中スタート時処理を行う（S 4 1 1）。なお、通常ART中スタート時処理の詳細については、後述の図 1 1 7を参照しながら後で説明する。そして、S 4 1 1の処理後、メインCPU 1 0 1は、状態別制御処理を終了し、処理をメインフロー（図 8 2 参照）のS 2 0 9に移す。

【 0 7 6 1 】

一方、S 4 1 0において、メインCPU 1 0 1が、現在の遊技状態が通常ARTでないと判別したとき（S 4 1 0がNO判定の場合）、メインCPU 1 0 1は、現在の遊技状態がCTであるか否かを判別する（S 4 1 2）。

【 0 7 6 2 】

S 4 1 2において、メインCPU 1 0 1が、現在の遊技状態がCTであると判別したとき（S 4 1 2がYES判定の場合）、メインCPU 1 0 1は、CT中スタート時処理を行う（S 4 1 3）。なお、CT中スタート時処理の詳細については、後述の図 1 1 8を参照しながら後で説明する。そして、S 4 1 3の処理後、メインCPU 1 0 1は、状態別制御処理を終了し、処理をメインフロー（図 8 2 参照）のS 2 0 9に移す。

【 0 7 6 3 】

一方、S 4 1 2において、メインCPU 1 0 1が、現在の遊技状態がCTでないと判別したとき（S 4 1 2がNO判定の場合）、メインCPU 1 0 1は、現在の遊技状態がボーナス状態であるか否かを判別する（S 4 1 4）。

【 0 7 6 4 】

S 4 1 4において、メインCPU 1 0 1が、現在の遊技状態がボーナス状態であると判別したとき（S 4 1 4がYES判定の場合）、メインCPU 1 0 1は、BB中スタート時処理を行う（S 4 1 5）。なお、BB中スタート時処理の詳細については、後述の図 1 2 5を参照しながら後で説明する。そして、S 4 1 5の処理後、メインCPU 1 0 1は、状態別制御処理を終了し、処理をメインフロー（図 8 2 参照）のS 2 0 9に移す。

【 0 7 6 5 】

一方、S 4 1 4において、メインCPU 1 0 1が、現在の遊技状態がボーナス状態でないと判別したとき（S 4 1 4がNO判定の場合）、メインCPU 1 0 1は、その他処理を行う（S 4 1 6）。この処理では、メインCPU 1 0 1は、上記各種判定処理で対象となった遊技状態以外の遊技状態に応じた処理を行う。例えば、現在の遊技状態がART準備状態である場合には、ART準備状態に対応した処理を行う。そして、S 4 1 6の処理後、メインCPU 1 0 1は、状態別制御処理を終了し、処理をメインフロー（図 8 2 参照）のS 2 0 9に移す。

【 0 7 6 6 】

[サブフラグ変換処理]

次に、図 1 0 5 ~ 図 1 0 7を参照して、状態別制御処理（図 1 0 4 参照）中のS 4 0 1で行うサブフラグ変換処理について説明する。図 1 0 5は、サブフラグ変更処理の手順を示すフローチャートである。また、図 1 0 6は、サブフラグ変更処理を実行するためのソースプログラムの一例を示す図であり、図 1 0 7は、サブフラグ変換処理のソースプログラム上で、実際に参照されるサブフラグ変換テーブル（変換テーブル）の構成の一例を示す

10

20

30

40

50

図である。

【0767】

まず、メインCPU101は、小役当籤番号(0~36)を取得する(S421)。次いで、メインCPU101は、現在、ボーナス作動中であるか否かを判別する(S422)。

【0768】

S422において、メインCPU101が、現在、ボーナス作動中であると判別したとき(S422がYES判定の場合)、メインCPU101は、小役当籤番号をボーナス作動中のサブフラグに変換して保存する(S423)。この小役当籤番号をボーナス作動中のサブフラグに変換する処理は、メインCPU101が図106中のソースコード「SUB(減算命令) cNHT_RBST - c7HT1_FLA」を実行することにより行われる。そして、本実施形態では、この「SUB」命令の実行により、一律、小役当籤番号をサブフラグ「サボテン(14)」に変換する。そして、S423の処理後、メインCPU101は、サブフラグ変換処理を終了し、処理を状態別制御処理(図104参照)のS402に移す。

10

【0769】

一方、S422において、メインCPU101が、現在、ボーナス作動中でないと判別したとき(S422がNO判定の場合)、メインCPU101は、図107に示すサブフラグ変換テーブルをセットする(S424)。この処理では、判定対象とするサブフラグの初期値を「ハズレ(00)」にセットするとともに参照対象とする図107に示すサブフラグ変換テーブル内のブロックの初期アドレスとして、サブフラグ「ハズレ(00)」が格納されているアドレス(「dSBCVTB+1」)をセットする。

20

【0770】

次いで、メインCPU101は、現在、参照対象となっているサブフラグ変換テーブル内のブロックに規定されている小役当籤番号のデータが、現ゲームで取得された小役当籤番号に対応するデータであるか否かを判別する(S425)。

【0771】

S425において、メインCPU101が、参照対象となっているサブフラグ変換テーブル内のブロックに規定されている小役当籤番号のデータが、現ゲームで取得された小役当籤番号に対応するデータでないと判別したとき(S425がNO判定の場合)、メインCPU101は、参照対象とするサブフラグ変換テーブル内のブロックを次のアドレスのブロックに更新する(S426)。次いで、メインCPU101は、サブフラグの値に「1」を加算する(S427)。そして、S427の処理後、メインCPU101は、処理をS425の処理に戻し、S425以降の処理を繰り返す。

30

【0772】

一方、S425において、メインCPU101が、参照対象となっているサブフラグ変換テーブル内のブロックに規定されている小役当籤番号のデータが、現ゲームで取得された小役当籤番号に対応するデータであると判別したとき(S425がYES判定の場合)、メインCPU101は、図107に示すサブフラグ変換テーブルを参照して、小役当籤番号に対応付けられたサブフラグ変換制御データ(小役当籤番号のアドレスの次のアドレスに格納された1バイトデータ)を取得し、該サブフラグ変換制御データをメインRAM103に設けられたサブフラグ変換制御データ格納領域(不図示)に格納する(S428)。この処理において、例えば、現ゲームで取得された小役当籤番号が「03」(内部当籤役「F_3連チリリプ」)である場合には、図107に示すサブフラグ変換テーブルを参照して、サブフラグ変換制御データ「00000011B」が取得される。そして、S428の処理後、メインCPU101は、サブフラグ変換処理を終了し、処理を状態別制御処理(図104参照)のS402に移す。

40

【0773】

本実施形態では、上述のようにしてサブフラグ変換処理が行われる。なお、上述したサブフラグ変換処理は、メインCPU101が、図106のソースプログラムで規定されている各ソースコードを順次実行することにより行われる。また、サブフラグ変換処理のソ-

50

スプログラム上で実際に参照される、図107に示すサブフラグ変換テーブルでは、各サブフラグに対してサブフラグ変換制御データ（制御ステータス）が対応付けられている。この際、同種のサブフラグに対しては、同じサブフラグ変換制御データ（制御ステータス）が対応付けられている。

【0774】

例えば、サブフラグ「3連チリリブA」及び「3連チリリブB」に対しては、サブフラグ変換制御データ（制御ステータス）「00000011B」が共通して割り付けられている。また、例えば、サブフラグ「リーチ目リブ1」～「リーチ目リブ4」に対しては、サブフラグ変換制御データ（制御ステータス）「00000001B」が共通して割り付けられている。そして、上述した内部当籤役（サブフラグ）をサブフラグEXに変換する際のフラグ変換抽籤処理では、サブフラグ変換制御データ格納領域に格納されたサブフラグ変換制御データ（制御ステータス）に基づいて、抽籤が行われる。

10

【0775】

メイン側で管理するフラグ（内部当籤役）をサブ側で管理可能なフラグに変換するためのサブフラグ変換テーブルにおいて、同種の内部当籤役（サブフラグ）に対して共通のサブフラグ変換制御データを設けることにより、該変換テーブルの汎用性が高くなり、機種変更に伴う変換プログラムの変更も軽微な変更で対応可能となるので、開発コストの増大を抑制することができる。

【0776】

[ナビセット処理]

次に、図108～図110を参照して、状態別制御処理（図104参照）中のS402で行うナビセット処理について説明する。図108は、ナビセット処理の手順を示すフローチャートである。また、図109は、ナビセット処理中の後述のS434～S436の処理を実行するためのソースプログラムの一例を示す図であり、図110は、ナビセット処理のソースプログラム上で、実際に参照されるナビデータテーブルの構成の一例を示す図である。

20

【0777】

まず、メインCPU101は、サブフラグ変換制御データ格納領域（不図示）にナビセットフラグがセットされているか否かを判別する（S431）。具体的には、メインCPU101は、サブフラグ変換制御データ格納領域を参照し、セットされているサブフラグ変換制御データが、押し順ナビを発生させる小役当籤番号（10～23）に対応するデータであるか否かを判別する。S431において、メインCPU101が、サブフラグ変換制御データ格納領域にナビセットフラグがセットされていないと判別したとき（S431がNO判定の場合）、メインCPU101は、ナビセット処理を終了し、処理を状態別制御処理（図104参照）のS403に移す。

30

【0778】

一方、S431において、メインCPU101が、サブフラグ変換制御データ格納領域にナビセットフラグがセットされていると判別したとき（S431がYES判定の場合）、メインCPU101は、RT状態がRT0又はRT1状態であるか否かを判別する（S432）。S432において、メインCPU101が、RT状態がRT0又はRT1状態でないと判別したとき（S432がNO判定の場合）、メインCPU101は、ナビセット処理を終了し、処理を状態別制御処理（図104参照）のS403に移す。

40

【0779】

一方、S432において、メインCPU101が、RT状態がRT0又はRT1状態であると判別したとき（S432がYES判定の場合）、メインCPU101は、遊技状態が一般遊技状態であるか否かを判別する（S433）。S433において、メインCPU101が、遊技状態が一般遊技状態であると判別したとき（S433がYES判定の場合）、メインCPU101は、ナビセット処理を終了し、処理を状態別制御処理（図104参照）のS403に移す。

【0780】

50

一方、S 4 3 3において、メインCPU 1 0 1が、遊技状態が一般遊技状態でないと判別したとき（S 4 3 3がNO判定の場合）、メインCPU 1 0 1は、小役当籤番号を取得する（S 4 3 4）。次いで、メインCPU 1 0 1は、図 1 1 0 に示すナビデータテーブルを参照し、小役当籤番号に基づいて、ナビデータ（1～9のいずれか）を取得する（S 4 3 5）。

【0781】

次いで、メインCPU 1 0 1は、取得したナビデータ（複数の表示列の変動表示の停止操作に関する情報）をメインRAM 1 0 3内の図示しないナビデータ格納領域（停止操作指示情報格納領域）に格納する（S 4 3 6）。そして、S 4 3 6の処理後、メインCPU 1 0 1は、ナビセット処理を終了し、処理を状態別制御処理（図 1 0 4 参照）のS 4 0 3に移す。

10

【0782】

本実施形態では、上述のようにしてナビセット処理が行われる。なお、上述したナビセット処理中のS 4 3 4～S 4 3 6の処理は、メインCPU 1 0 1が、図 1 0 9 のソースプログラムで規定されている各ソースコードを順次実行することにより行われる。この一連の処理では、図 1 0 9 に示すように、ソースプログラム上において、Qレジスタ（拡張レジスタ）を用いてアドレス指定を行うメインCPU 1 0 1専用の「LDQ」命令が用いられる。

【0783】

ソースプログラム上において、例えば、ソースコード「LDQ A, (k)」が実行されると、Qレジスタの格納データ（上位側アドレス値）と、1バイトの整数k（直値：下位側アドレス値）とで指定されたアドレスのメモリの内容（格納データ）がAレジスタにロードされる。それゆえ、例えば、図 1 0 9 中のソースコード「LDQ A, (WHITFR T)」が実行されると、Qレジスタの格納データと、整数値「WHITFR T」とで指定されるアドレスのメモリの内容がAレジスタにロードされる。

20

【0784】

また、ソースプログラム上において、例えば、ソースコード「LDQ (k), A」が実行されると、Aレジスタの格納データが、Qレジスタの格納データ（上位側アドレス値）と、1バイトの整数k（直値：下位側アドレス値）とで指定されたアドレスのメモリにロードされる。それゆえ、例えば、図 1 0 9 中のソースコード「LDQ (wNAV IPT N), A」の実行により、Aレジスタに格納されたデータ（ナビデータ）が、Qレジスタの格納データ（上位側アドレス値）と1バイトの整数値「wNAV IPT N」（下位側アドレス値）とで指定されたアドレスのナビデータ格納領域に格納される。

30

【0785】

上述のように、本実施形態では、ナビセット処理において、Qレジスタ（拡張レジスタ）を用いたメインCPU 1 0 1専用命令コードが用いられ、直値により、メインROM 1 0 2、メインRAM 1 0 3やメモリーマップI/Oにアクセスすることができる。この場合、ナビセット処理のソースプログラム上において、アドレス設定に係る命令を省略することができるので、その分、ソースプログラムの容量（メインROM 1 0 2の使用容量）を低減することができる。この結果、本実施形態では、メインROM 1 0 2において、空き容量を確保する（増大させる）ことができ、増えた空き容量を活用して、遊技性を高めることが可能になる。

40

【0786】

[フラグ変換処理]

次に、図 1 1 1 を参照して、状態別制御処理（図 1 0 4 参照）中のS 4 0 4で行うフラグ変換処理について説明する。なお、図 1 1 1 は、フラグ変換処理の手順を示すフローチャートである。

【0787】

まず、メインCPU 1 0 1は、CT開始時であるか否かを判別する（S 4 4 1）。

【0788】

50

S 4 4 1において、メインCPU101が、CT開始時でないと判別したとき(S 4 4 1がNO判定の場合)、メインCPU101は、後述のS 4 4 3の処理を行う。一方、S 4 4 1において、メインCPU101が、CT開始時であると判別したとき(S 4 4 1がYES判定の場合)、メインCPU101は、CT中のフラグ変換抽籤に用いるフラグ変換抽籤テーブルのテーブル番号を抽籤で決定し、セットする(S 4 4 2)。

【0789】

S 4 4 2の処理後又はS 4 4 1がNO判定の場合、メインCPU101は、現在の状態に応じたフラグ変換抽籤テーブルをセットする(S 4 4 3)。例えば、現在の状態が非ART中のRT4状態である場合には、非ART中フラグ変換抽籤テーブル(図62参照)がセットされ、現在の状態が通常ART中のRT4状態である場合には、ART中フラグ変換抽籤テーブル(図47A及び47B参照)がセットされ、現在の状態がCT中のRT4状態である場合には、CT中フラグ変換抽籤テーブル(図54参照)がセットされる。

10

【0790】

次いで、メインCPU101は、セットされたフラグ変換抽籤テーブルを参照し、内部当籤役に基づいてフラグ変換抽籤処理を行う(S 4 4 4)。なお、実際、この処理では、メインCPU101は、内部当籤役に対応するサブフラグに基づいて、図107に示すサブフラグ変換テーブルから取得されるサブフラグ変換制御データを用いてフラグ変換抽籤処理を行う。

【0791】

次いで、メインCPU101は、S 4 4 4のフラグ変換抽籤に当籤したか否かを判別する(S 4 4 5)。

20

【0792】

S 4 4 5において、メインCPU101が、フラグ変換抽籤に当籤したと判別したとき(S 4 4 5がYES判定の場合)、メインCPU101は、サブフラグ変換処理を行う(S 4 4 6)。この処理において、例えば、内部当籤役が「F_1確チリリブ」である場合、すなわち、サブフラグが「3連チリリブB(03)」である場合、フラグ変換抽籤処理に当籤すると、S 4 4 6のサブフラグ変換処理により、サブフラグ「3連チリリブB(03)」が、サブフラグEX「確定役(06)」又はサブフラグEX「3連チリリブ(07)」に変換される(図36参照)。

【0793】

S 4 4 6の処理後、メインCPU101は、現在の遊技状態が非ART状態であるか否かを判別する(S 4 4 7)。S 4 4 7において、メインCPU101が、現在の遊技状態が非ART状態でないと判別したとき(S 4 4 7がNO判定の場合)、メインCPU101は、フラグ変換処理を終了し、処理を状態別制御処理(図104参照)のS 4 0 5に移す。

30

【0794】

一方、S 4 4 7において、メインCPU101が、現在の遊技状態が非ART状態であると判別したとき(S 4 4 7がYES判定の場合)、メインCPU101は、ARTセット数に「1」を加算する(S 4 4 8)。次いで、メインCPU101は、次遊技の遊技状態にART準備状態をセットする(S 4 4 9)。そして、S 4 4 9の処理後、メインCPU101は、フラグ変換処理を終了し、処理を状態別制御処理(図104参照)のS 4 0 5に移す。

40

【0795】

ここで再度、S 4 4 5の処理に戻って、S 4 4 5において、メインCPU101が、フラグ変換抽籤に当籤しなかったと判別したとき(S 4 4 5がNO判定の場合)、メインCPU101は、サブフラグ維持処理を行う(S 4 5 0)。この処理において、例えば、内部当籤役が「F_1確チリリブ」である場合、すなわち、サブフラグが「3連チリリブB(03)」である場合、フラグ変換抽籤に非当籤であると、S 4 5 0のサブフラグ維持処理により、サブフラグ「3連チリリブB(03)」が、サブフラグEX「リプレイ(01)」に変換(維持)される。そして、S 4 5 0の処理後、メインCPU101は、フラグ変換処理を終了し、処理を状態別制御処理(図104参照)のS 4 0 5に移す。

50

【 0 7 9 6 】

[通常中スタート時処理]

次に、図 1 1 2 を参照して、状態別制御処理（図 1 0 4 参照）中の S 4 0 7 で行う通常中スタート時処理について説明する。なお、図 1 1 2 は、通常中スタート時処理の手順を示すフローチャートである。

【 0 7 9 7 】

まず、メイン CPU 1 0 1 は、C Z 抽籤テーブル（図 4 1 A 参照）を参照し、現在の C Z の抽籤状態及び内部当籤役（サブフラグ）に基づいて C Z 抽籤処理を行う（S 4 6 1）。次いで、メイン CPU 1 0 1 は、S 4 6 1 の C Z 抽籤に当籤したか否かを判別する（S 4 6 2）。

10

【 0 7 9 8 】

S 4 6 2 において、メイン CPU 1 0 1 が、C Z 抽籤に当籤しなかったと判別したとき（S 4 6 2 が NO 判定の場合）、メイン CPU 1 0 1 は、後述の S 4 6 5 の処理を行う。

【 0 7 9 9 】

一方、S 4 6 2 において、メイン CPU 1 0 1 が、C Z 抽籤に当籤したと判別したとき（S 4 6 2 が YES 判定の場合）、メイン CPU 1 0 1 は、次遊技の遊技状態に当籤した種別の C Z をセットする（S 4 6 3）。次いで、メイン CPU 1 0 1 は、当籤した種別の C Z ゲーム数を C Z ゲーム数カウンタにセットする（S 4 6 4）。なお、C Z ゲーム数カウンタは、C Z の継続期間を計数するカウンタであり、メイン RAM 1 0 3 に設けられる。S 4 6 4 の処理において、例えば、C Z 1 が当籤している場合には、C Z ゲーム数カウンタ（前半部）に第 1 の所定ゲーム数（例えば、「1 2」）がセットされ、C Z 2 が当籤している場合には、C Z ゲーム数カウンタ（前半部）に第 2 の所定ゲーム数（例えば、「1 5」）がセットされ、C Z 3 が当籤している場合には、C Z ゲーム数カウンタに第 4 の所定ゲーム数（例えば、「1 7」）がセットされる。

20

【 0 8 0 0 】

S 4 6 4 の処理後又は S 4 6 2 が NO 判定の場合、メイン CPU 1 0 1 は、通常中高確率抽籤テーブル（図 4 0 A 参照）を参照し、内部当籤役（サブフラグ）に基づいて C Z の抽籤状態の移行抽籤を行う（S 4 6 5）。次いで、メイン CPU 1 0 1 は、移行抽籤の結果に基づいて、C Z の抽籤状態を更新する（S 4 6 6）。そして、S 4 6 6 の処理後、メイン CPU 1 0 1 は、通常中スタート時処理を終了するとともに、状態別制御処理（図 1 0 4 参照）も終了する。

30

【 0 8 0 1 】

[C Z 中スタート時処理]

次に、図 1 1 3 を参照して、状態別制御処理（図 1 0 4 参照）中の S 4 0 9 で行う C Z 中スタート時処理について説明する。なお、図 1 1 3 は、C Z 中スタート時処理の手順を示すフローチャートである。

【 0 8 0 2 】

まず、メイン CPU 1 0 1 は、現在の遊技状態が C Z 1 であるか否かを判別する（S 4 7 1）。

【 0 8 0 3 】

S 4 7 1 において、メイン CPU 1 0 1 が、現在の遊技状態が C Z 1 であると判別したとき（S 4 7 1 が YES 判定の場合）、メイン CPU 1 0 1 は、C Z 1（C Z 2）中処理を行う（S 4 7 2）。なお、C Z 1（C Z 2）中処理の詳細については、後述の図 1 1 4 及び図 1 1 5 を参照しながら後で説明する。そして、S 4 7 2 の処理後、メイン CPU 1 0 1 は、C Z 中スタート時処理を終了するとともに、状態別制御処理（図 1 0 4 参照）も終了する。

40

【 0 8 0 4 】

一方、S 4 7 1 において、メイン CPU 1 0 1 が、現在の遊技状態が C Z 1 でないと判別したとき（S 4 7 1 が NO 判定の場合）、メイン CPU 1 0 1 は、現在の遊技状態が C Z 2 であるか否かを判別する（S 4 7 3）。

50

【 0 8 0 5 】

S 4 7 3において、メインCPU 1 0 1が、現在の遊技状態がC Z 2であると判別したとき（S 4 7 3がYES判定の場合）、メインCPU 1 0 1は、C Z 1（C Z 2）中処理を行う（S 4 7 4）。C Z 1（C Z 2）中処理の詳細については、後述の図 1 1 4 及び図 1 1 5を参照しながら後で説明する。そして、S 4 7 4の処理後、メインCPU 1 0 1は、C Z中スタート時処理を終了するとともに、状態別制御処理（図 1 0 4参照）も終了する。なお、本実施形態では、C Z 1中処理とC Z 2中処理との間ではART抽籤に当籤する期待度を示すランク（モード又はポイント）が異なるだけであり、基本的な処理内容は同じである。そこで、本実施形態では、C Z 1中処理及びC Z 2中処理をC Z 1（C Z 2）中処理として一つの処理で説明する。

10

【 0 8 0 6 】

一方、S 4 7 3において、メインCPU 1 0 1が、現在の遊技状態がC Z 2でないと判別したとき（S 4 7 3がNO判定の場合）、メインCPU 1 0 1は、C Z 3中処理を行う（S 4 7 5）。なお、C Z 3中処理の詳細については、後述の図 1 1 6を参照しながら後で説明する。そして、S 4 7 5の処理後、メインCPU 1 0 1は、C Z中スタート時処理を終了するとともに、状態別制御処理（図 1 0 4参照）も終了する。

【 0 8 0 7 】

[C Z 1（C Z 2）中処理]

次に、図 1 1 4 及び図 1 1 5を参照して、C Z中スタート時処理（図 1 1 3参照）中のS 4 7 2又はS 4 7 4で行うC Z 1（C Z 2）中処理について説明する。なお、図 1 1 4 及び図 1 1 5は、C Z 1（C Z 2）中処理の手順を示すフローチャートである。

20

【 0 8 0 8 】

まず、メインCPU 1 0 1は、現遊技がC Z 1（又はC Z 2）の前半部の遊技であるか否かを判別する（S 4 8 1）。S 4 8 1において、メインCPU 1 0 1が、現遊技がC Z 1（又はC Z 2）の前半部の遊技でないと判別したとき（S 4 8 1がNO判定の場合）、メインCPU 1 0 1は、後述のS 4 9 0の処理を行う。

【 0 8 0 9 】

一方、S 4 8 1において、メインCPU 1 0 1が、現遊技がC Z 1の前半部の遊技であると判別したとき（S 4 8 1がYES判定の場合）、メインCPU 1 0 1は、C Z 1中モードアップ抽籤テーブル（図 4 2参照）を参照し、内部当籤役（サブフラグ）に基づいてモードアップ抽籤処理を行う（S 4 8 2）。また、S 4 8 1において、メインCPU 1 0 1が、現遊技がC Z 2の前半部の遊技であると判別したとき（S 4 8 1がYES判定の場合）、メインCPU 1 0 1は、C Z 2中ポイント抽籤テーブル（図 4 3参照）を参照し、内部当籤役（サブフラグ）に基づいてポイントアップ抽籤を行う（S 4 8 2）。

30

【 0 8 1 0 】

次いで、メインCPU 1 0 1は、S 4 8 2の抽籤結果に基づいて、ランク（モード又はポイント）を更新する（S 4 8 3）。次いで、メインCPU 1 0 1は、S 4 8 2の抽籤においてフリーズに当籤したか否かを判別する（S 4 8 4）。

【 0 8 1 1 】

S 4 8 4において、メインCPU 1 0 1が、フリーズに当籤したと判別したとき（S 4 8 4がYES判定の場合）、メインCPU 1 0 1は、遊技の進行を一時的に停止するフリーズ処理を行うとともに、ARTセット数及びCTセット数に「1」を加算する（S 4 8 5）。また、この処理では、メインCPU 1 0 1は、ARTレベル決定テーブル（図 4 8 A参照）を参照してARTレベルを決定し、セットする。なお、フリーズ発生時には、ARTレベルとして「ARTレベル2」が決定される。

40

【 0 8 1 2 】

次いで、メインCPU 1 0 1は、次遊技の遊技状態にART準備状態をセットする（S 4 8 6）。そして、S 4 8 6の処理後、メインCPU 1 0 1は、C Z 1（C Z 2）中処理を終了するとともに、C Z中スタート時処理（図 1 1 3参照）も終了する。

【 0 8 1 3 】

50

ここで再度、S 4 8 4 の処理に戻って、S 4 8 4 において、メインCPU 1 0 1 が、フリーズに当籤しなかったと判別したとき (S 4 8 4 がNO判定の場合)、メインCPU 1 0 1 は、C Z ゲーム数カウンタ (前半部) の値を1減算する (S 4 8 7)。次いで、メインCPU 1 0 1 は、C Z ゲーム数カウンタ (前半部) の値が「0」であるか否かを判別する (S 4 8 8)。

【0814】

S 4 8 8 において、メインCPU 1 0 1 が、C Z ゲーム数カウンタ (前半部) の値が「0」でないと判別したとき (S 4 8 8 がNO判定の場合)、メインCPU 1 0 1 は、C Z 1 (C Z 2) 中処理を終了するとともに、C Z 中スタート時処理 (図 1 1 3 参照) も終了する。

10

【0815】

一方、S 4 8 8 において、メインCPU 1 0 1 が、C Z ゲーム数カウンタ (前半部) の値が「0」であると判別したとき (S 4 8 8 がYES判定の場合)、メインCPU 1 0 1 は、次遊技の遊技状態にC Z 1 又はC Z 2 の後半部をセットする (S 4 8 9)。そして、S 4 8 9 の処理後、メインCPU 1 0 1 は、C Z 1 (C Z 2) 中処理を終了するとともに、C Z 中スタート時処理 (図 1 1 3 参照) も終了する。

【0816】

ここで再度、S 4 8 1 の処理に戻って、S 4 8 1 がNO判定の場合、メインCPU 1 0 1 は、現ゲームがC Z 1 又はC Z 2 の後半部の1ゲーム目であるか否かを判別する (S 4 9 0)。S 4 9 0 において、メインCPU 1 0 1 が、現ゲームがC Z 1 又はC Z 2 の後半部の1ゲーム目でないと判別したとき (S 4 9 0 がNO判定の場合)、メインCPU 1 0 1 は、後述のS 4 9 5 の処理を行う。

20

【0817】

一方、S 4 9 0 において、メインCPU 1 0 1 が、現ゲームがC Z 1 又はC Z 2 の後半部の1ゲーム目であると判別したとき (S 4 9 0 がYES判定の場合)、メインCPU 1 0 1 は、C Z 中ART抽籤テーブル (図 4 4 A 及び 4 4 B 参照) を参照し、前半部のランク (モード又はポイント) に基づいてART抽籤処理を行う (S 4 9 1)。

【0818】

次いで、メインCPU 1 0 1 は、ART抽籤に当籤したか否かを判別する (S 4 9 2)。S 4 9 2 において、メインCPU 1 0 1 が、ART抽籤に当籤しなかったと判別したとき (S 4 9 2 がNO判定の場合)、メインCPU 1 0 1 は、後述のS 4 9 4 の処理を行う。

30

【0819】

一方、S 4 9 2 において、メインCPU 1 0 1 が、ART抽籤に当籤したと判別したとき (S 4 9 2 がYES判定の場合)、メインCPU 1 0 1 は、ARTセット数に「1」を加算し、ARTレベル決定テーブル (図 4 8 A 参照) を参照してARTレベルの抽籤を行い、その抽籤結果をセットする (S 4 9 3)。

【0820】

S 4 9 3 の処理後又はS 4 9 2 がNO判定の場合、メインCPU 1 0 1 は、C Z ゲーム数カウンタ (後半部) に所定値をセットする (S 4 9 4)。なお、S 4 9 4 の処理において、例えば、ART抽籤に当籤している場合には、C Z ゲーム数カウンタ (後半部) に「4」がセットされ、ART抽籤に非当籤である場合には、C Z ゲーム数カウンタ (後半部) に「3」がセットされる。

40

【0821】

S 4 9 4 の処理後又はS 4 9 0 がNO判定の場合、メインCPU 1 0 1 は、C Z 中ART抽籤テーブル (図 4 4 C 参照) を参照して、内部当籤役 (サブフラグ) に基づいてART抽籤処理を行う (S 4 9 5)。

【0822】

次いで、メインCPU 1 0 1 は、ART抽籤に当籤したか否かを判別する (S 4 9 6)。S 4 9 6 において、メインCPU 1 0 1 が、ART抽籤に当籤しなかったと判別したとき (S 4 9 6 がNO判定の場合)、メインCPU 1 0 1 は、後述のS 4 9 8 の処理を行う。

50

【0823】

一方、S496において、メインCPU101が、ART抽籤に当籤したと判別したとき（S496がYES判定の場合）、メインCPU101は、ARTセット数に「1」を加算し、ARTレベル決定テーブル（図48A参照）を参照してARTレベルの抽籤を行い、その抽籤結果をセットする（S497）。

【0824】

S497の処理後又はS496がNO判定の場合、メインCPU101は、CZゲーム数カウンタ（後半部）の値を1減算する（S498）。次いで、メインCPU101は、CZゲーム数カウンタ（後半部）の値が「0」であるか否かを判別する（S499）。

【0825】

S499において、メインCPU101が、CZゲーム数カウンタ（後半部）の値が「0」でないと判別したとき（S499がNO判定の場合）、メインCPU101は、CZ1（CZ2）中処理を終了するとともに、CZ中スタート時処理（図113参照）も終了する。

【0826】

一方、S499において、メインCPU101が、CZゲーム数カウンタ（後半部）の値が「0」であると判別したとき（S499がYES判定の場合）、メインCPU101は、ARTセット数が「1」以上であるか否かを判別する（S500）。

【0827】

S500において、メインCPU101が、ARTセット数が「1」以上であると判別したとき（S500がYES判定の場合）、メインCPU101は、次遊技の遊技状態にART準備状態をセットする（S501）。そして、S501の処理後、メインCPU101は、CZ1（CZ2）中処理を終了するとともに、CZ中スタート時処理（図113参照）も終了する。

【0828】

一方、S500において、メインCPU101が、ARTセット数が「1」以上でないと判別したとき（S500がNO判定の場合）、メインCPU101は、次遊技の遊技状態にCZ失敗時の状態をセットする（S502）。そして、S502の処理後、メインCPU101は、CZ1（CZ2）中処理を終了するとともに、CZ中スタート時処理（図113参照）も終了する。

【0829】

[CZ3中処理]

次に、図116を参照して、CZ中スタート時処理（図113参照）中のS475で行うCZ3中処理について説明する。なお、図116は、CZ3中処理の手順を示すフローチャートである。

【0830】

まず、メインCPU101は、CZ中ART抽籤テーブル（図45参照）を参照し、内部当籤役（サブフラグ）に基づいてART抽籤処理を行う（S511）。

【0831】

次いで、メインCPU101は、ART抽籤に当籤したか否かを判別する（S512）。S512において、メインCPU101が、ART抽籤に当籤しなかったと判別したとき（S512がNO判定の場合）、メインCPU101は、後述のS518の処理を行う。

【0832】

一方、S512において、メインCPU101が、ART抽籤に当籤したと判別したとき（S512がYES判定の場合）、メインCPU101は、ARTセット数に「1」を加算し、CTセット数に「1」を加算する（S513）。次いで、メインCPU101は、S512のART抽籤においてフリーズに当籤したか否かを判別する（S514）。

【0833】

S514において、メインCPU101が、フリーズに当籤しなかったと判別したとき（S514がNO判定の場合）、メインCPU101は、後述のS516の処理を行う。一

10

20

30

40

50

方、S 5 1 4において、メインCPU 1 0 1が、フリーズに当籤したと判別したとき（S 5 1 4がYES判定の場合）、メインCPU 1 0 1は、遊技の進行を一時的に停止するフリーズ処理を行う（S 5 1 5）。

【0834】

S 5 1 5の処理後又はS 5 1 4がNO判定の場合、メインCPU 1 0 1は、ARTレベル決定テーブル（図48A参照）を参照してARTレベルの抽籤処理を行い、その抽籤結果（ARTレベル）をセットする（S 5 1 6）。次いで、メインCPU 1 0 1は、次遊技の遊技状態にART準備状態をセットする（S 5 1 7）。そして、S 5 1 7の処理後、メインCPU 1 0 1は、C Z 3中処理を終了するとともに、C Z中スタート時処理（図113参照）も終了する。

10

【0835】

ここで再度、S 5 1 2の処理に戻って、S 5 1 2がNO判定の場合、メインCPU 1 0 1は、C Zゲーム数カウンタの値を1減算する（S 5 1 8）。次いで、メインCPU 1 0 1は、C Zゲーム数カウンタの値が「0」であるか否かを判別する（S 5 1 9）。

【0836】

S 5 1 9において、メインCPU 1 0 1が、C Zゲーム数カウンタの値が「0」でないと判別したとき（S 5 1 9がNO判定の場合）、メインCPU 1 0 1は、C Z 3中処理を終了するとともに、C Z中スタート時処理（図113参照）も終了する。

【0837】

一方、S 5 1 9において、メインCPU 1 0 1が、C Zゲーム数カウンタの値が「0」であると判別したとき（S 5 1 9がYES判定の場合）、メインCPU 1 0 1は、ARTセット数に「1」を加算し、ARTレベル決定テーブル（図48A参照）を参照してARTレベルの抽籤を行い、その抽籤結果をセットする（S 5 2 0）。次いで、メインCPU 1 0 1は、次遊技の遊技状態にART準備状態をセットする（S 5 2 1）。そして、S 5 2 1の処理後、メインCPU 1 0 1は、C Z 3中処理を終了するとともに、C Z中スタート時処理（図113参照）も終了する。

20

【0838】

[通常ART中スタート時処理]

次に、図117を参照して、状態別制御処理（図104参照）中のS 4 1 1で行う通常ART中スタート時処理について説明する。なお、図117は、通常ART中スタート時処理の手順を示すフローチャートである。

30

【0839】

まず、メインCPU 1 0 1は、ART継続ゲーム数カウンタの値に「1」を加算する（S 5 3 1）。なお、ART継続ゲーム数カウンタは、通常ARTが継続したゲーム数（消化ゲーム数）を計数するカウンタである。また、本実施形態では、ART継続ゲーム数カウンタの他に、通常ARTが継続可能なゲーム数を計数するART終了ゲーム数カウンタも設ける。そして、本実施形態のパチスロ1では、ART継続ゲーム数カウンタの値とART終了ゲーム数カウンタの値とを比較し、ART継続ゲーム数カウンタの値がART終了ゲーム数カウンタの値に到達すると、ART遊技状態が終了する。

【0840】

次いで、メインCPU 1 0 1は、ART中CT抽籤テーブル（図50参照）を参照し、現在のCT抽籤状態及び内部当籤役（サブフラグ）に基づいてCT抽籤処理を行う（S 5 3 2）。次いで、メインCPU 1 0 1は、CT抽籤に当籤したか否かを判別する（S 5 3 3）。S 5 3 3において、メインCPU 1 0 1が、CT抽籤に当籤しなかったと判別したとき（S 5 3 3がNO判定の場合）、メインCPU 1 0 1は、後述のS 5 3 6の処理を行う。

40

【0841】

一方、S 5 3 3において、メインCPU 1 0 1が、CT抽籤に当籤したと判別したとき（S 5 3 3がYES判定の場合）、メインCPU 1 0 1は、CTセット数に「1」を加算し、CTゲーム数カウンタの値に「8」をセットする（S 5 3 4）。次いで、メインCPU 1 0 1は、次遊技の遊技状態に当籤した種別のCTをセットする（S 5 3 5）。

50

【 0 8 4 2 】

S 5 3 5 の処理後又は S 5 3 3 が N O 判定の場合、メイン C P U 1 0 1 は、A R T レベル決定テーブル（図 4 8 B 参照）を参照し、A R T 継続ゲーム数カウンタの値に基づいて A R T レベルを抽籤し、その抽籤結果をセットする（S 5 3 6）。次いで、メイン C P U 1 0 1 は、通常 A R T 中高確率抽籤テーブル（図 4 9 参照）を参照し、現在の C T 抽籤状態及び内部当籤役（サブフラグ）に基づいて、移行先の C T 抽籤状態を抽籤し、その抽籤結果をセットする（S 5 3 7）。

【 0 8 4 3 】

次いで、メイン C P U 1 0 1 は、通常 A R T 中上乘せ抽籤テーブル（図 5 1 参照）を参照し、内部当籤役（サブフラグ）に基づいて A R T ゲーム数の上乘せ抽籤処理を行う（S 5 3 8）。次いで、メイン C P U 1 0 1 は、上乘せ抽籤に当籤したか否かを判別する（S 5 3 9）。

10

【 0 8 4 4 】

S 5 3 9 において、メイン C P U 1 0 1 が、上乘せ抽籤に当籤しなかったと判別したとき（S 5 3 9 が N O 判定の場合）、メイン C P U 1 0 1 は、後述の S 5 4 1 の処理を行う。一方、S 5 3 9 において、メイン C P U 1 0 1 が、上乘せ抽籤に当籤したと判別したとき（S 5 3 9 が Y E S 判定の場合）、メイン C P U 1 0 1 は、当籤結果を A R T 終了ゲーム数カウンタに加算する（S 5 4 0）。

【 0 8 4 5 】

S 5 4 0 の処理後又は S 5 3 9 が N O 判定の場合、メイン C P U 1 0 1 は、A R T 継続ゲーム数カウンタの値が、A R T 終了ゲーム数カウンタの値に達したか否かを判定する（S 5 4 1）。S 5 4 1 において、メイン C P U 1 0 1 が、A R T 継続ゲーム数カウンタの値が、A R T 終了ゲーム数カウンタの値に達していないと判別したとき（S 5 4 1 が N O 判定の場合）、メイン C P U 1 0 1 は、通常 A R T 中スタート時処理を終了するとともに、状態別制御処理（図 1 0 4 参照）も終了する。

20

【 0 8 4 6 】

一方、S 5 4 1 において、メイン C P U 1 0 1 が、A R T 継続ゲーム数カウンタの値が、A R T 終了ゲーム数カウンタの値に達したと判別したとき（S 5 4 1 が Y E S 判定の場合）、メイン C P U 1 0 1 は、A R T セット数を 1 減算する（S 5 4 2）。次いで、メイン C P U 1 0 1 は、A R T 終了時の状態をセットする（S 5 4 3）。そして、S 5 4 3 の処理後、メイン C P U 1 0 1 は、通常 A R T 中スタート時処理を終了するとともに、状態別制御処理（図 1 0 4 参照）も終了する。

30

【 0 8 4 7 】

[C T 中スタート時処理]

次に、図 1 1 8 を参照して、状態別制御処理（図 1 0 4 参照）中の S 4 1 3 で行う C T 中スタート時処理について説明する。なお、図 1 1 8 は、C T 中スタート時処理の手順を示すフローチャートである。

【 0 8 4 8 】

まず、メイン C P U 1 0 1 は、C T 中上乘せ抽籤テーブル（図 5 5 参照）を参照し、内部当籤役（サブフラグ）に基づいて A R T ゲーム数の上乘せ抽籤を行う（S 5 5 1）。次いで、メイン C P U 1 0 1 は、上乘せ抽籤に当籤したか否かを判別する（S 5 5 2）。

40

【 0 8 4 9 】

S 5 5 2 において、メイン C P U 1 0 1 が、上乘せ抽籤に当籤しなかったと判別したとき（S 5 5 2 が N O 判定の場合）、メイン C P U 1 0 1 は、後述の S 5 5 6 の処理を行う。一方、S 5 5 2 において、メイン C P U 1 0 1 が、上乘せ抽籤に当籤したと判別したとき（S 5 5 2 が Y E S 判定の場合）、メイン C P U 1 0 1 は、当籤結果を A R T 終了ゲーム数カウンタに加算する（S 5 5 3）。なお、上述したように、本実施形態のパチスロ 1 では、同一の C T 中にサブフラグ「3 連チリリップ（3 連チリリップ A 又は 3 連チリリップ B）」に当籤した回数が増えるほど、1 回の抽籤当りの上乘せ量が増える。

【 0 8 5 0 】

50

S 5 5 3 の処理後、メイン CPU 1 0 1 は、内部当籤役がサブフラグ EX 「 3 連チリリブ 」 (又は 「 確定役 」) に対応する役であるか否か、すなわち、内部当籤役 「 F __ 確チリリブ 」 又は 「 F __ 1 確チリリブ 」 であり、かつ、図 1 1 1 中の S 4 4 4 のフラグ変換抽籤処理に当籤したか否かを判別する (S 5 5 4) 。

【 0 8 5 1 】

S 5 5 4 において、メイン CPU 1 0 1 が、内部当籤役がサブフラグ EX 「 3 連チリリブ 」 に対応する役でないと判別したとき (S 5 5 4 が NO 判定の場合) 、メイン CPU 1 0 1 は、CT 中スタート時処理を終了するとともに、状態別制御処理 (図 1 0 4 参照) も終了する。

【 0 8 5 2 】

一方、S 5 5 4 において、メイン CPU 1 0 1 が、内部当籤役がサブフラグ EX 「 3 連チリリブ 」 に対応する役であると判別したとき (S 5 5 4 が YES 判定の場合) 、メイン CPU 1 0 1 は、CT ゲーム数カウンタの値に 「 1 」 を加算する (S 5 5 5) 。そして、S 5 5 5 の処理後、メイン CPU 1 0 1 は、CT 中スタート時処理を終了するとともに、状態別制御処理 (図 1 0 4 参照) も終了する。

【 0 8 5 3 】

ここで再度、S 5 5 2 の処理に戻って、S 5 5 2 が NO 判定の場合、メイン CPU 1 0 1 は、CT 中 CT 抽籤処理を行う (S 5 5 6) 。この処理では、メイン CPU 1 0 1 は、主に、CT セット数の上乘せ抽籤を行う。なお、CT 中 CT 抽籤処理の詳細については、後述の図 1 1 9 を参照しながら後で説明する。

【 0 8 5 4 】

次いで、メイン CPU 1 0 1 は、CT ゲーム数カウンタの値を 1 減算する (S 5 5 7) 。次いで、メイン CPU 1 0 1 は、CT ゲーム数カウンタの値が 「 0 」 であるか否かを判別する (S 5 5 8) 。

【 0 8 5 5 】

S 5 5 8 において、メイン CPU 1 0 1 が、CT ゲーム数カウンタの値が 「 0 」 でないと判別したとき (S 5 5 8 が NO 判定の場合) 、メイン CPU 1 0 1 は、CT 中スタート時処理を終了するとともに、状態別制御処理 (図 1 0 4 参照) も終了する。一方、S 5 5 8 において、メイン CPU 1 0 1 が、CT ゲーム数カウンタの値が 「 0 」 であると判別したとき (S 5 5 8 が YES 判定の場合) 、メイン CPU 1 0 1 は、CT セット数が 「 1 」 以上であるか否かを判別する (S 5 5 9) 。

【 0 8 5 6 】

S 5 5 9 において、メイン CPU 1 0 1 が、CT セット数が 「 1 」 以上であると判別したとき (S 5 5 9 が YES 判定の場合) 、メイン CPU 1 0 1 は、CT セット数を 1 減算する (S 5 6 0) 。次いで、メイン CPU 1 0 1 は、CT ゲーム数カウンタの値に 「 8 」 をセットする (S 5 6 1) 。そして、S 5 6 1 の処理後、メイン CPU 1 0 1 は、CT 中スタート時処理を終了するとともに、状態別制御処理 (図 1 0 4 参照) も終了する。

【 0 8 5 7 】

一方、S 5 5 9 において、メイン CPU 1 0 1 が、CT セット数が 「 1 」 以上でないと判別したとき (S 5 5 9 が NO 判定の場合) 、メイン CPU 1 0 1 は、次遊技の遊技状態に通常 ART をセットする (S 5 6 2) 。そして、S 5 6 2 の処理後、メイン CPU 1 0 1 は、CT 中スタート時処理を終了するとともに、状態別制御処理 (図 1 0 4 参照) も終了する。

【 0 8 5 8 】

[CT 中 CT 抽籤処理]

次に、図 1 1 9 を参照して、CT 中スタート時処理 (図 1 1 8 参照) 中の S 5 5 6 で行う CT 中 CT 抽籤処理について説明する。なお、図 1 1 9 は、CT 中 CT 抽籤処理の手順を示すフローチャートである。

【 0 8 5 9 】

まず、メイン CPU 1 0 1 は、CT 中 CT 抽籤テーブルをセットする (S 5 7 1) 。なお

10

20

30

40

50

、ここで、セットされるCT中CT抽籤テーブルは、上記図56で説明したCT中セット数上乘せ抽籤テーブルであるが、ソースプログラム上で実際にセットされるCT中CT抽籤テーブル（CT中セット数上乘せ抽籤テーブル）の構成については、後述の図122を参照しながら後で説明する。

【0860】

次いで、メインCPU101は、テーブルデータ取得処理を行う（S572）。この処理では、メインCPU101は、CT中CT抽籤処理で参照する抽籤テーブルのアドレスを取得する。なお、テーブルデータ取得処理の詳細については、後述の図120を参照しながら後で説明する。

【0861】

次いで、メインCPU101は、1バイト抽籤処理を行う（S573）。この処理では、メインCPU101は、CTセットの上乗せ抽籤を行う。なお、1バイト抽籤処理の詳細については、後述の図123を参照しながら後で説明する。

【0862】

次いで、メインCPU101は、1バイト抽籤処理に当籤したか否かを判別する（S574）。S574において、メインCPU101が、1バイト抽籤処理に当籤しなかったと判別したとき（S574がNO判定の場合）、メインCPU101は、CT中CT抽籤処理を終了し、処理をCT中スタート時処理（図118参照）のS557に移す。

【0863】

一方、S574において、メインCPU101が、1バイト抽籤処理に当籤したと判別したとき（S574がYES判定の場合）、メインCPU101は、CTセット数に「1」を加算する（S575）。そして、S575の処理後、メインCPU101は、CT中CT抽籤処理を終了し、処理をCT中スタート時処理（図118参照）のS557に移す。

【0864】

[テーブルデータ取得処理]

次に、図120～図122を参照して、CT中CT抽籤処理（図119参照）中のS572で行うテーブルデータ取得処理について説明する。図120は、テーブルデータ取得処理の手順を示すフローチャートである。また、図121は、テーブルデータ取得処理を実行するためのソースプログラムの一例を示す図であり、図122は、CT中CT抽籤処理及びテーブルデータ取得処理のソースプログラム上で、実際に参照されるCT中CT抽籤テーブル（図56で説明したCT中セット数上乘せ抽籤テーブルに対応）の構成の一例を示す図である。なお、図122に示すCT中CT抽籤テーブルに格納されている具体的な各種抽籤値は一例である。

【0865】

なお、本実施形態において、CT中CT抽籤処理で参照する抽籤値を取得する際、2段階のアドレス算出処理（1段階目及び2段階目のテーブルデータ取得処理）を経て、抽籤値が格納されているアドレスを算出する。まず、1段階目のテーブルデータ取得処理（後述のS582及びS583の処理）では、内部当籤役（実際にはサブフラグD）に対応付けられた「選択値（1バイト）」が取得される。なお、選択値には、内部当籤役の種別毎に設けられ、内部当籤役が抽籤対象であるか否かが判別可能であり且つ内部当籤役に対応付けられた抽籤テーブルの配置先を指定可能な値（相対値）が規定される。また、本実施形態では、CT中CT抽籤処理の抽籤結果が非当籤となる内部当籤役（実際にはサブフラグD）に対して予め選択値「0」を規定し、それらの内部当籤役を1段階目のテーブルデータ取得処理の時点で「ハズレ」として扱う。そして、2段階目のテーブルデータ取得処理（後述のS585～S587の処理）では、「0」以外の選択値が規定された内部当籤役の抽籤値が格納されたアドレスが算出される（抽籤テーブルの基準アドレス（2バイト）から相対値（選択値）を加算したアドレスが算出される）。

【0866】

まず、メインCPU101は、CT中CT抽籤選択テーブル（不図示）を参照して、CT中CT抽籤テーブルのアドレスを算出するための1段階目及び2段階目の加算選択データ

10

20

30

40

50

のアドレス、並びに、CT抽籤の抽籤回数（本実施形態では、2回）を取得する（S581）。次いで、メインCPU101は、1段階目の加算選択データのアドレスをCT中CT抽籤テーブルのアドレスに加算して、1段階目の選択アドレスを算出する（S582）。

【0867】

次いで、メインCPU101は、算出した1段階目の選択アドレスに格納されている選択値を取得する（S583）。次いで、メインCPU101は、選択値が「0」であるか否かを判別する（S584）。

【0868】

S584において、メインCPU101が、選択値が「0」であると判別したとき（S584がYES判定の場合）、メインCPU101は、テーブルデータ取得処理を終了し、処理をCT中CT抽籤処理（図119参照）のS573に移す。

10

【0869】

一方、S584において、メインCPU101が、選択値が「0」でないと判別したとき（S584がNO判定の場合）、メインCPU101は、選択アドレスに選択値を加算して、2段階目の選択アドレスを算出する（S585）。次いで、メインCPU101は、2段階目の選択アドレスに2段階目の加算選択データのアドレスを加算して、選択アドレスを算出する（S586）。

【0870】

次いで、メインCPU101は、S586で算出した選択アドレスに格納されている選択値を取得し、該選択値を選択アドレスに加算して、CT中CT抽籤テーブル内において参照するアドレスを算出する（S587）。そして、S587の処理後、メインCPU101は、テーブルデータ取得処理を終了し、処理をCT中CT抽籤処理（図119参照）のS573に移す。

20

【0871】

本実施形態では、上述のようにしてテーブルデータ取得処理が行われる。そして、上述したテーブルデータ取得処理は、メインCPU101が、図121のソースプログラムで規定されている各ソースコードを順次実行することにより行われる。

【0872】

その中で、1段階目のテーブルデータ取得処理（S581～S584の処理）では、図122に示すCT中CT当籤抽籤テーブル中のアドレス「dCTCTSTTB」～「dCTCTS__RER-1」までの領域に格納されたテーブル（当籤役別テーブル選択相対テーブル）が参照される。また、当籤役別テーブル選択相対テーブル（抽籤テーブル選択テーブル）内において、CT当籤がハズレとなる各役（サブフラグD「ハズレ」、「サボテン」、「弱チェリー」、「強チェリー」、「確定役」、「3連チリリブ」）の選択テーブルのアドレスには、上述した選択値（相対値）として「0」が格納される。そして、1段階目のテーブルデータ取得処理（アドレス算出処理）では、選択値が「0」である場合には、抽籤結果を「ハズレ」にする（上記S584の判定処理参照）。

30

【0873】

上記構成のCT中CT当籤抽籤テーブルでは、1段階目のテーブルデータ取得処理で参照する当籤役別テーブル選択相対テーブルにおいて、役の種別だけで「ハズレ」を設定することができるので、抽籤テーブルに「ハズレ」役の抽籤値を規定する必要がなくなる。それゆえ、本実施形態では、CT中CT当籤抽籤テーブルにおいて、「ハズレ」役の抽籤値データ（「0」）を格納する必要がなくなり、メインROM102のテーブル領域の容量を節約することができる。

40

【0874】

また、2段階目（サブフラグD「リーチ目リブ」取得時）のテーブルデータ取得処理（S585～S587の処理）中のS587の処理では、算出された抽籤テーブルアドレスに基づいて、図122に示すCT中CT当籤抽籤テーブル（図56のCT中セット数上乘せ抽籤テーブルに対応）内から、通常CT状態時に用いる抽籤テーブル（先頭アドレス「dNMCTCTS__RER」）又は高確率CT状態の抽籤テーブル（先頭アドレス「dSP

50

「CTCTS__RER」)の一方が選択される。

【0875】

高確率CT状態時に用いる抽籤テーブルでは、図122に示すように、先頭アドレス「dSPCTCTS__RER」の次のアドレス領域に1バイトデータからなる「判定ビット」(判定データ)が格納される。判定ビットには、抽籤対象の抽籤値が格納されたアドレスの範囲を示すデータが格納される。図122に示す例のように、高確率CT状態の抽籤テーブルの「判定ビット」の格納領域の次のアドレスにのみ抽籤対象(高確CT)の抽籤値が格納されている場合には、判定ビットの格納領域には、ビット0にのみ「1」が格納された1バイトデータ「00000001B」が判定ビットとして格納される。一方、通常CT状態時に用いる抽籤テーブルのように、判定ビットの格納領域の次のアドレス及び次々アドレスに抽籤対象(高確CT及び通常CT)の抽籤値が格納されている場合には、図122に示すように、判定ビットの格納領域には、ビット0及び1にのみ「1」が格納された1バイトデータ「00000011B」が判定ビットとして格納される。このような判定ビットを設けた場合、抽籤テーブルにおいて、判定ビットにおいてビットデータが「0」となるアドレスの領域に抽籤対象外の抽籤値データを格納する必要が無くなる。

10

【0876】

さらに、高確率CT状態時に用いる抽籤テーブルでは、図122に示すように、「判定ビット」の次のアドレス「dSPCTCTS__RER+2」に高確率CT当籤の抽籤値が格納され、高確率CT当籤の抽籤値としてアドレス「cABS__HIT」に規定されたデータが格納される。本実施形態では、このアドレス「cABS__HIT」に規定されているデータは、当籤確定(100%当籤)を示すデータ(以下、「確定データ」という)である。また、本実施形態では、CT中CT当籤抽籤テーブルにおいてハズレ用の抽籤値データ(「0」)を設ける必要がないので、図122に示すように、アドレス「cABS__HIT」に規定されている確定データに「0」を規定することができる。すなわち、上記構成のCT中CT当籤抽籤テーブルでは、抽籤値「0」を確定データとして使用することができる。

20

【0877】

なお、図56のCT中セット数上乘せ抽籤テーブルに説明したように、本実施形態では、高確率CT状態時のCT中セット数上乘せ抽籤では、必ず、「高確率CT当籤」が決定される。それゆえ、本実施形態では、ソースプログラム上では、図122に示すCT中CT当籤抽籤テーブルにおいて、高確率CT当籤の抽籤値として確定データを格納することができる。

30

【0878】

上述した2段階目(サブフラグD「リーチ目リブ」取得時)の抽籤テーブルのように、判定ビットを構成する各ビットデータの値により、CT抽籤の抽籤対象役及び抽籤対象外の役(サブフラグD)を判別することにより、抽籤対象外の役の抽籤値データ(ハズレデータ)をテーブルに格納する必要が無くなる。また、抽籤対象役の当籤確率が100%である確定データとしては、抽籤値「0」を用いることができる。これらのことから、本実施形態では、CT中CT当籤抽籤テーブル(CT中セット数上乘せ抽籤テーブル)の容量を圧縮することができ、メインROM102において、空き容量を確保する(増大させる)ことができ、増えた空き容量を活用して、遊技性を高めることが可能になる。また、本実施形態では、この技術をCT中CT当籤処理で使用する例を説明したが、本発明はこれに限定されず、ART抽籤(図115参照)、ARTゲーム数上乘せ抽籤(図117参照)、後述のCT抽籤(後述の図154参照)及び後述のCZの引き戻し抽籤(後述の図157参照)等で使用してもよい。

40

【0879】

[1バイト抽籤処理]

次に、図123及び図124を参照して、CT中CT抽籤処理(図119参照)中のS573で行う1バイト抽籤処理について説明する。図123は、1バイト抽籤処理の手順を示すフローチャートである。また、図124は、1バイト抽籤処理を実行するためのソー

50

プログラムの一例を示す図である。

【0880】

まず、メインCPU101は、メインRAM103内の乱数格納領域（不図示）に格納されているCT中CT抽籤用の1バイト乱数値（0～255：乱数回路110の乱数レジスタ4のソフトラッチ乱数）をセットする（S591）。次いで、メインCPU101は、テーブルデータ取得処理中のS587で算出したアドレスに基づいて、CT中CT抽籤テーブルから抽籤判定データ（図122の2段階目に規定されている抽籤テーブル中の判定ビット）を取得する（S592）。また、この処理では、メインCPU101は、抽籤回数の初期値として、判定ビットのビット数「8」をセットする。

【0881】

次いで、メインCPU101は、抽籤判定データが抽籤対象であるか否かを判別する（S593）。この判定処理では、メインCPU101は、現在の抽籤回数に対応付けられた判定ビット内のビットデータを参照し、該ビットデータが「1」であれば、抽籤対象であると判定する。なお、本実施形態では、判定ビット内のビット0～ビット7が、抽籤回数「8」～「1」にそれぞれ対応付けられている。

【0882】

S593において、メインCPU101が、抽籤判定データが抽籤対象でないと判別したとき（S593がNO判定の場合）、メインCPU101は、後述のS599の処理を行う。一方、S593において、メインCPU101が、抽籤判定データが抽籤対象であると判別したとき（S593がYES判定の場合）、メインCPU101は、CT中CT抽籤テーブルから抽籤値を取得する（S594）。

【0883】

次いで、メインCPU101は、抽籤値が「0」（当籤確定データ）であるか否かを判別する（S595）。

【0884】

S595において、メインCPU101が、抽籤値が「0」とであると判別したとき（S595がYES判定の場合）、メインCPU101は、1バイト抽籤処理を終了し、処理をCT中CT抽籤処理（図119参照）のS574に移す。一方、S595において、メインCPU101が、抽籤値が「0」でないと判別したとき（S595がNO判定の場合）、メインCPU101は、CT抽籤（CTセット数の上乘せ抽籤）処理を行う（S596）。具体的には、メインCPU101は、乱数値（1バイト乱数値）から抽籤値を減算し、その減算結果を乱数値とする。

【0885】

次いで、メインCPU101は、S596のCT抽籤に当籤したか否かを判別する（S597）。なお、S596のCT抽籤では、メインCPU101は、S596の減算結果が「0」以下となった場合（いわゆる「桁かり」が生じた場合）に、当籤したと判定する。

【0886】

S597において、メインCPU101が、CT抽籤に当籤したと判別したとき（S597がYES判定の場合）、メインCPU101は、1バイト抽籤処理を終了し、処理をCT中CT抽籤処理（図119参照）のS574に移す。一方、S597において、メインCPU101が、CT抽籤に当籤しなかったと判別したとき（S597がNO判定の場合）、メインCPU101は、CT中CT抽籤テーブル内において参照する抽籤値の格納アドレス（抽籤アドレス）を次の抽籤アドレスに更新する（S598）。

【0887】

S598の処理後又はS593がNO判定の場合、メインCPU101は、抽籤回数を1減算する（S599）。次いで、メインCPU101は、抽籤回数が「0」であるか否かを判別する（S600）。

【0888】

S600において、メインCPU101が、抽籤回数が「0」でないと判別したとき（S600がNO判定の場合）、メインCPU101は、処理をS593に戻し、S593以

10

20

30

40

50

降の処理を繰り返す。一方、S 6 0 0において、メインCPU 1 0 1が、抽籤回数が「0」であると判別したとき（S 6 0 0がYES判定の場合）、メインCPU 1 0 1は、1バイト抽籤処理を終了し、処理をCT中CT抽籤処理（図1 1 9参照）のS 5 7 4に移す。

【0 8 8 9】

本実施形態では、上述のようにして1バイト抽籤処理が行われる。なお、上述した1バイト抽籤処理は、メインCPU 1 0 1が、図1 2 4のソースプログラムで規定されている各ソースコードを順次実行することにより行われる。

【0 8 9 0】

1バイト抽籤処理中のS 5 9 1の乱数取得処理では、図1 2 4に示すように、ソースプログラム上において、Qレジスタ（拡張レジスタ）を用いてアドレス指定を行うメインCPU 1 0 1専用命令コードである「LDQ」命令が用いられる。それゆえ、本実施形態では、1バイト抽籤処理においても、Qレジスタ（拡張レジスタ）を用いた命令コードが用いられることにより、直値により、メインROM 1 0 2、メインRAM 1 0 3やメモリーマップE / Oにアクセスすることができるので、アドレス設定に係る命令コードを省略することができる。この結果、本実施形態では、メインROM 1 0 2において、空き容量を確保する（増大させる）ことができ、増えた空き容量を活用して、遊技性を高めることが可能になる。

【0 8 9 1】

[BB中スタート時処理]

次に、図1 2 5を参照して、状態別制御処理（図1 0 4参照）中のS 4 1 5で行うBB中スタート時処理について説明する。なお、図1 2 5は、BB中スタート時処理の手順を示すフローチャートである。

【0 8 9 2】

まず、メインCPU 1 0 1は、ボーナス中ARTゲーム数上乘せ抽籤テーブル（図5 9参照）を参照し、内部当籤役に基づいてARTゲーム数の上乘せ抽籤処理を行う（S 6 1 1）。次いで、メインCPU 1 0 1は、上乘せ抽籤に当籤したか否かを判別する（S 6 1 2）。

【0 8 9 3】

S 6 1 2において、メインCPU 1 0 1が、上乘せ抽籤に当籤しなかったと判別したとき（S 6 1 2がNO判定の場合）、メインCPU 1 0 1は、BB中スタート時処理を終了するとともに、状態別制御処理（図1 0 4参照）も終了する。一方、S 6 1 2において、メインCPU 1 0 1が、上乘せ抽籤に当籤したと判別したとき（S 6 1 2がYES判定の場合）、メインCPU 1 0 1は、当籤結果（上乘せゲーム数）をART終了ゲーム数カウンタに加算する（S 6 1 3）。

【0 8 9 4】

次いで、メインCPU 1 0 1は、ARTセット数が「0」であるか否かを判別する（S 6 1 4）。S 6 1 4において、メインCPU 1 0 1が、ARTセット数が「0」でないと判別したとき（S 6 1 4がNO判定の場合）、メインCPU 1 0 1は、BB中スタート時処理を終了するとともに、状態別制御処理（図1 0 4参照）も終了する。

【0 8 9 5】

一方、S 6 1 4において、メインCPU 1 0 1が、ARTセット数が「0」であると判別したとき（S 6 1 4がYES判定の場合）、メインCPU 1 0 1は、ARTセット数に「1」を加算する（S 6 1 5）。そして、S 6 1 5の処理後、メインCPU 1 0 1は、BB中スタート時処理を終了するとともに、状態別制御処理（図1 0 4参照）も終了する。

【0 8 9 6】

[引込優先順位格納処理]

次に、図1 2 6及び図1 2 7を参照して、メインフロー（図8 2参照）中のS 2 1 2で行う引込優先順位格納処理について説明する。図1 2 6は、引込優先順位格納処理の手順を示すフローチャートである。また、図1 2 7は、引込優先順位格納処理中の後述のS 6 2

10

20

30

40

50

5 及び S 6 2 6 の処理を実行するためのソースプログラムの一例を示す図である。

【 0 8 9 7 】

まず、メイン CPU 1 0 1 は、検索ルール数に「 3 」をセットする (S 6 2 1)。次いで、メイン CPU 1 0 1 は、引込優先順位テーブル選択処理を行う (S 6 2 2)。この処理では、内部当籤役及び作動ストップボタンに基づいて、引込優先順位テーブル (図 2 7 参照) が選択される。

【 0 8 9 8 】

次いで、メイン CPU 1 0 1 は、引込優先順位格納領域選択処理を行う (S 6 2 3)。この処理では、検索対象のルールの引込優先順位データ格納領域が選択される。次いで、メイン CPU 1 0 1 は、図柄チェック数 (回数) として「 2 0 」をセットする (S 6 2 4)。 10

【 0 8 9 9 】

次いで、メイン CPU 1 0 1 は、図柄コード取得処理を行う (S 6 2 5)。この処理では、図柄チェック数に対応した入賞作動フラグ格納領域及び図柄コード格納領域を参照して、図柄コードを取得する。なお、図柄コード取得処理の詳細については、後述の図 1 2 8 を参照しながら後で説明する。

【 0 9 0 0 】

次いで、メイン CPU 1 0 1 は、論理積演算処理を行う (S 6 2 6)。この処理では、メイン CPU 1 0 1 は、入賞作動フラグデータの生成処理を行う。論理積演算処理の詳細については、後述の図 1 3 3 を参照しながら後で説明する。

【 0 9 0 1 】

次いで、メイン CPU 1 0 1 は、引込優先順位取得処理を行う (S 6 2 7)。この処理では、メイン CPU 1 0 1 は、入賞作動フラグ (入賞役) 格納領域 (図 2 8 ~ 図 3 0 参照) 内においてビットが「 1 」にセットされており、かつ、当り要求フラグ格納領域でビットが「 1 」にされている役について、引込優先順位テーブル (図 2 7 参照) を参照して、引込優先順位データを取得する。なお、引込優先順位取得処理の詳細については、後述の図 1 3 4 及び図 1 3 5 を参照しながら後で説明する。 20

【 0 9 0 2 】

次いで、メイン CPU 1 0 1 は、取得した引込優先順位データをメイン RAM 1 0 3 内の引込優先順位データ格納領域 (不図示) に格納する (S 6 2 8)。この際、引込優先順位データは、各優先順位の値と、格納領域のビットとが対応するように引込優先順位データ格納領域に格納される。 30

【 0 9 0 3 】

なお、引込優先順位データ格納領域には、メインルールの種類毎に優先順位データの格納領域が設けられる。各引込優先順位データ格納領域には、対応するメインルールの各図柄位置「 0 」 ~ 「 1 9 」に応じて決定された引込優先順位データが格納される。本実施形態では、この引込優先順位データ格納領域を参照することにより、停止テーブルに基づいて決定された滑り駒数の他に、より適切な滑り駒数が存在するか否かを検索する。

【 0 9 0 4 】

引込優先順位データ格納領域に格納される優先順位引込データの内容は、引込優先順位データを決定する際に参照された引込優先順位テーブル内の引込優先順位テーブル番号の種類によって異なる。また、引込優先順位データは、その値が大きいほど優先順位が高いことを表す。引込優先順位データを参照することにより、メインルールの周面に配された各図柄間における優先順位の相対的な評価が可能となる。すなわち、引込優先順位データとして最も大きい値が決定されている図柄が最も優先順位の高い図柄となる。したがって、引込優先順位データは、メインルールの周面に配された各図柄間の順位を示すものともいえる。なお、引込優先順位データの値が等しい図柄が複数存在する場合には、優先順序テーブルが規定する優先順序に従って 1 つの図柄が決定される。 40

【 0 9 0 5 】

次いで、メイン CPU 1 0 1 は、引込優先順位格納領域の更新処理を行う (S 6 2 9)。この処理では、メイン CPU 1 0 1 は、次のチェック図柄の引込優先順位データ格納領域 50

をセットする。次いで、メインCPU101は、図柄チェック数を1減算する(S630)。次いで、メインCPU101は、図柄チェック数が「0」であるか否かを判別する(S631)。

【0906】

S631において、メインCPU101が、図柄チェック数が「0」でないと判別したとき(S631がNO判定の場合)、メインCPU101は、処理をS625の処理に戻し、S625以降の処理を繰り返す。一方、S631において、メインCPU101が、図柄チェック数が「0」であると判別したとき(S631がYES判定の場合)、メインCPU101は、検索対象リールの変更処理を行う(S632)。

【0907】

次いで、メインCPU101は、検索リール数を1減算する(S633)。次いで、メインCPU101は、検索リール数が「0」であるか否か、すなわち、全てのメインリールに対して上述した一連の処理が行われたか否かを判別する(S634)。

【0908】

S634において、メインCPU101が、検索リール数が「0」でないと判別したとき(S634がNO判定の場合)、メインCPU101は、処理をS622の処理に戻し、S622以降の処理を繰り返す。一方、S634において、メインCPU101が、検索リール数が「0」であると判別したとき(S634がYES判定の場合)、メインCPU101は、引込優先順位格納処理を終了し、処理をメインフロー(図82参照)のS213に移す。

【0909】

本実施形態では、上述のようにして引込優先順位格納処理が行われる。上述した引込優先順位格納処理中のS625及びS626の処理は、メインCPU101が、図127のソースプログラムで規定されている各ソースコードを順次実行することにより行われる。

【0910】

その中で、S626の論理積演算処理は、メインCPU101が図127中のソースコード「CALLF SB__DAND__00」を実行することにより行われる。「CALLF」命令は、上述のようにメインCPU101専用の2バイト命令コードであり、図127中のソースコード「CALLF SB__DAND__00」が実行されると、「SB__DAND__00」で指定されているアドレスに、処理をジャンプさせ、論理積演算処理が開始される。

【0911】

[図柄コード取得処理]

次に、図128～図132を参照して、引込優先順位格納処理(図126参照)中のS625で行う図柄コード取得処理について説明する。図128は、図柄コード取得処理の手順を示すフローチャートであり、図129は、図柄コード取得処理を実行するためのソースプログラムの一例を示す図である。図130Aは、図柄コード取得処理のソースプログラム上で、実際に参照される第1リール(左リール)図柄配置テーブルの構成の一例を示す図であり、図130Bは、第1リール図柄配置テーブルセット時に参照される図柄対応入賞作動テーブルの構成の一例を示す図である。図131Aは、図柄コード取得処理のソースプログラム上で、実際に参照される第2リール(中リール)図柄配置テーブルの構成の一例を示す図であり、図131Bは、第2リール図柄配置テーブルセット時に参照される図柄対応入賞作動テーブルの構成の一例を示す図である。また、図132Aは、図柄コード取得処理のソースプログラム上で、実際に参照される第3リール(右リール)図柄配置テーブルの構成の一例を示す図であり、図132Bは、第3リール図柄配置テーブルセット時に参照される図柄対応入賞作動テーブルの構成の一例を示す図である。

【0912】

まず、メインCPU101は、入賞作動フラグ格納領域のクリア処理を行う(S641)。この処理では、メインCPU101は、入賞作動フラグ格納領域(図28～図30参照)内の全ての格納領域に「0」をセットする。次いで、メインCPU101は、第1リール

10

20

30

40

50

ル図柄配置テーブル(図130A参照)をセットする(S642)。

【0913】

次いで、メインCPU101は、第1リール(左リール3L)の停止時であるか否かを判別する(S643)。

【0914】

S643において、メインCPU101が、第1リール(左リール3L)の停止時であると判別したとき(S643がYES判定の場合)、メインCPU101は、後述のS647の処理を行う。一方、S643において、メインCPU101が、第1リール(左リール3L)の停止時でないと判別したとき(S643がNO判定の場合)、メインCPU101は、第2リール図柄配置テーブル(図131A参照)をセットする(S644)。この処理では、S642の処理でセットされた第1リール図柄配置テーブルが、第2リール図柄配置テーブルで上書きされる。

10

【0915】

次いで、メインCPU101は、第2リール(中リール3C)の停止時であるか否かを判別する(S645)。

【0916】

S645において、メインCPU101が、第2リール(中リール3C)の停止時であると判別したとき(S645がYES判定の場合)、メインCPU101は、後述のS647の処理を行う。一方、S645において、メインCPU101が、第2リール(中リール3C)の停止時でないと判別したとき(S645がNO判定の場合)、メインCPU101は、第3リール図柄配置テーブル(図132A参照)をセットする(S646)。この処理では、S644の処理でセットされた第2リール図柄配置テーブルが、第3リール図柄配置テーブルで上書きされる。

20

【0917】

S646の処理後、又は、S643或いはS645がYES判定の場合、メインCPU101は、停止制御対象のリールに対する停止操作実行時の図柄チェック処理を行い、図柄チェック処理により取得された図柄に対応する図柄対応入賞作動テーブルを取得する(S647)。例えば、第1リール(左リール3L)停止時であり、停止操作時に有効ライン上に位置する図柄が「白7」である場合、メインCPU101は、図130B中のアドレス「dR1__SVN1」~アドレス「dR1__SVN2-1」の範囲のブロックに規定された図柄対応入賞作動テーブルの先頭アドレスを取得する。

30

【0918】

次いで、メインCPU101は、入賞作動フラグ格納領域をセットする(S648)。次いで、メインCPU81は、図101で説明した圧縮データ格納処理を行う(S649)。この処理では、メインCPU101は、主に、図柄対応入賞作動テーブルに格納された入賞可能な入賞作動フラグデータを、入賞作動フラグ格納領域内の対応する格納領域に転送(展開)する処理を行う。

【0919】

例えば、第1リール(左リール3L)停止時であり、停止操作時に有効ライン上に位置する図柄が「白7」である場合には、入賞可能な図柄組合せ(コンビネーション)は、図28~図30に示すように、第2格納領域に規定されるコンビネーション名称「C__2nd__A__01」、「C__2nd__A__01」及び「C__SP1__01」、第3格納領域に規定されるコンビネーション名称「C__9枚C__01」~「C__9枚C__03」、「C__9枚C__07」~「C__9枚C__09」及び「C__9枚E__01」、第4格納領域に規定されるコンビネーション名称「C__RB役A__01」、「C__RB役A__02」、「C__RB役B__01」~「C__RB役B__04」、「C__RB役C__01」及び「C__RB役C__02」、第6格納領域に規定されるコンビネーション名称「C__リーチ目リプC__01」~「C__リーチ目リプC__03」、「C__リーチ目リプD__01」、「C__リーチ目リプD__02」及び「C__リーチ目リプE__01」、並びに、第10格納領域に規定されるコンビネーション名称「C__BB1」である。

40

50

【0920】

この場合、図柄対応入賞作動テーブルに格納された入賞可能な入賞作動フラグデータの第1ブロック（第0～第7格納領域）の格納先は、図130Bに示すテーブル内のアドレス「dR1__SVN1+1」に格納されている1バイトの指定データ「01011100B」により指定される（図130B中のコメント「格納領域 +2, +3, +4, +6」欄参照）。また、図柄対応入賞作動テーブルに格納された入賞可能な入賞作動フラグデータの第2ブロック（第8～第11格納領域）の格納先は、図130Bに示すテーブル内のアドレス「dR1__SVN1+6」に格納されている1バイトの指定データ「10000000B」により指定される（図130B中のコメント「格納領域 +10」欄参照）。

【0921】

なお、本実施形態では、第1ブロックの指定データのビット0～ビット7が、格納先として、それぞれ第1ブロックの第0～第7格納領域を指定するビットであり、第2ブロックの指定データのビット0～ビット3が、格納先として、それぞれ第2ブロックの第8～第11格納領域を指定するビットである。そして、各ブロックの1バイトの指定データでは、入賞作動フラグデータの格納先となる入賞作動フラグ格納領域内の格納領域に対応するビットに「1」が格納される。

【0922】

それゆえ、例えば、第1リール（左リール3L）停止時であり、停止操作時に有効ライン上に位置する図柄が「白7」である場合、S649の処理において、図130Bに示すテーブル内のアドレス「dR1__SVN1+2」に格納されている入賞作動フラグデータ「00010000B」又は「00000100B」が、図柄対応入賞作動テーブルから入賞作動フラグ格納領域内の第2格納領域に転送され、アドレス「dR1__SVN1+3」に格納されている入賞作動フラグデータ「00100000B」又は「00000100B」が、図柄対応入賞作動テーブルから入賞作動フラグ格納領域内の第3格納領域に転送される。また、この場合、S649の処理において、図130Bに示すテーブル内のアドレス「dR1__SVN1+4」に格納されている入賞作動フラグデータ「10000000B」、「01000000B」又は「00100000B」が、図柄対応入賞作動テーブルから入賞作動フラグ格納領域内の第4格納領域に転送され、アドレス「dR1__SVN1+5」に格納されている入賞作動フラグデータ「00100000B」、「00010000B」又は「00001000B」が、図柄対応入賞作動テーブルから入賞作動フラグ格納領域内の第6格納領域に転送される。さらに、この場合、S649の処理において、図130Bに示すテーブル内のアドレス「dR1__SVN1+8」に格納されている入賞作動フラグデータ「10000000B」が、図柄対応入賞作動テーブルから入賞作動フラグ格納領域内の第10格納領域に転送される。

【0923】

S649の処理後、メインCPU101は、圧縮データ格納処理により更新された入賞作動フラグ格納領域をセットし、図柄コード格納領域をセットし、入賞作動フラグ格納領域のデータ長（本実施形態では12バイト）をセットする（S650）。そして、S650の処理後、メインCPU101は、図柄コード取得処理を終了し、処理を引込優先順位格納処理（図126参照）のS626に移す。

【0924】

本実施形態では、上述のようにして図柄コード取得処理が行われる。なお、上述した図柄コード取得処理は、メインCPU101が、図129のソースプログラムで規定されている各ソースコードを順次実行することにより行われる。その中でも、S649の圧縮データ格納処理は、メインCPU101が図129中のソースコード「CALLF SB__BTEP__00」を実行することにより行われる。

【0925】

「CALLF」命令は、上述のようにメインCPU101専用の2バイト命令コードであり、図129中のソースコード「CALLF SB__BTEP__00」が実行されると、「SB__BTEP__00」で指定されているアドレスに、処理をジャンプさせ、圧縮デー

10

20

30

40

50

タ格納処理が開始される。そして、この圧縮データ格納処理では、上述のように、各リールの図柄対応入賞作動フラグテーブルに格納された入賞作動フラグデータ（圧縮データ）が、入賞作動フラグ格納領域に展開（コピー）される。

【0926】

なお、本実施形態では、上述した図柄コード取得処理中のS647～S649で説明した処理手順で入賞に係るデータの圧縮・展開処理を行い、かつ、その処理の中で上述したメインCPU101専用命令コードを用いることにより、入賞に係るデータの圧縮・展開処理の効率化を図ることができるとともに、限られたメインRAM103の容量を有効活用することができる。

【0927】

また、本実施形態では、図柄コード取得処理中のS649の圧縮データ格納処理において、「CALLF」命令で指定するジャンプ先のアドレス「SB__BTEP__00」は、図97で説明した図柄設定処理中のS330の圧縮データ格納処理において、「CALLF」命令で指定するジャンプ先のアドレスと同じである。すなわち、本実施形態では、図柄コード取得処理で行う圧縮データ格納処理を実行するためのソースプログラムが、図柄設定処理で行う圧縮データ格納処理を実行するためのソースプログラムと同じであり、S649及びS330の両処理において、圧縮データ格納処理のソースプログラムが共有化（モジュール化）されている。この場合、S649及びS330の両処理において、それぞれ別個に圧縮データ格納処理のソースプログラムを設ける必要が無くなるので、その分、ソースプログラムの容量（メインROM102の使用容量）を低減することができる。この結果、本実施形態では、メインROM102において、空き容量を確保する（増大させる）ことができ、増えた空き容量を活用して、遊技性を高めることが可能になる。

【0928】

[論理積演算処理]

次に、図133を参照して、例えば、引込優先順位格納処理（図126参照）中のS626で行う論理積演算処理について説明する。図133は、論理積演算処理の手順を示すフローチャートである。なお、図133に示す論理積演算処理は、引込優先順位格納処理（図126参照）中のS626だけでなく、後述の引込優先順位取得処理（後述の図134及び図135参照）中のS687においても実行される。

【0929】

引込優先順位格納処理（図126参照）中のS626で実行される論理積演算処理において、論理積演算される2つのデータは、上述した図柄コード取得処理中のS650でセットされた入賞作動フラグ格納領域のデータ、及び、図柄コード格納領域のデータである。そして、前者のデータが後述の「論理積先データ」に対応し、後者のデータが後述の「論理積元データ」に対応する。また、この場合、上述した図柄コード取得処理中のS650でセットされたデータ長（12バイト）のバイト数「12」が後述の「論理積回数」に対応する。

【0930】

一方、後述の引込優先順位取得処理（後述の図134及び図135参照）中のS687で実行される論理積演算処理において、論理積演算される2つのデータは、当り（引込）要求フラグ格納領域のデータ、及び、入賞作動フラグ格納領域のデータである。そして、前者のデータが後述の「論理積先データ」に対応し、後者のデータが後述の「論理積元データ」に対応する。また、この場合、後述の図136B中に記載のRT作動組み合わせ表示フラグのデータ長（1バイト）のバイト数「1」が後述の「論理積回数」に対応する。

【0931】

まず、メインCPU101は、論理積元データ（例えば、図柄コード格納領域のデータ）を取得する（S661）。次いで、メインCPU101は、論理積元データと論理積先データ（例えば、入賞作動フラグ格納領域のデータ）との論理積演算を行い、その演算結果を論理積先データとして保存する（S662）。

【0932】

10

20

30

40

50

次いで、メインCPU101は、取得する論理積元データのアドレスを1加算する(S663)。次いで、メインCPU101は、参照する論理積先データのアドレスを1加算する(S664)。

【0933】

次いで、メインCPU101は、論理積回数を1減算する(S665)。次いで、メインCPU101は、論理積回数が「0」であるか否かを判別する(S666)。

【0934】

S666において、メインCPU101が、論理積回数が「0」でないと判別したとき(S666がNO判定の場合)、メインCPU101は、処理をS661の処理に戻し、S661以降の処理を繰り返す。一方、S666において、メインCPU101が、論理積回数が「0」であると判別したとき(S666がYES判定の場合)、メインCPU101は、論理積演算処理を終了し、処理を例えば引込優先順位格納処理(図126参照)のS627に移す。

【0935】

[引込優先順位取得処理]

次に、図134～図137を参照して、引込優先順位格納処理(図126参照)中のS627で行う引込優先順位取得処理について説明する。なお、図134及び図135は、引込優先順位取得処理の手順を示すフローチャートである。図136Aは、引込優先順位取得処理中の後述のS680～S683の処理を実行するためのソースプログラムの一例を示す図であり、図136Bは、引込優先順位取得処理中の後述のS686の処理を実行するためのソースプログラムの一例を示す図であり、図136Cは、引込優先順位取得処理中の後述のS687の処理を実行するためのソースプログラムの一例を示す図である。また、図137は、引込優先順位取得処理のソースプログラム上で、実際に参照される引込優先順位テーブルの構成の一例を示す図である。

【0936】

まず、メインCPU101は、右ルール3R(特定の表示列)のチェック時であるか否かを判別する(S671)。

【0937】

S671において、メインCPU101が、右ルール3Rのチェック時でないと判別したとき(S671がNO判定の場合)、メインCPU101は、後述のS674の処理を行う。一方、S671において、メインCPU101が、右ルール3Rのチェック時であると判別したとき(S671がYES判定の場合)、メインCPU101は、内部当籤役に係る図柄組合せ(入賞役)に「ANY役」(所定の図柄の組合せ)が含まれるか否かを判別する(S672)。なお、ここでいう「ANY役」とは、少なくとも右ルール3Rの停止図柄に関係なく入賞が確定する役(少なくとも右ルール3Rの停止図柄が任意の図柄である入賞役)のことをいう。

【0938】

S672において、メインCPU101が、内部当籤役に係る図柄組合せに「ANY役」が含まれないと判別したとき(S672がNO判定の場合)、メインCPU101は、後述のS674の処理を行う。一方、S672において、メインCPU101が、内部当籤役に係る図柄組合せに「ANY役」が含まれると判別したとき(S672がYES判定の場合)、メインCPU101は、入賞作動フラグ格納領域内の「ANY役」に対応する格納領域をマスクする(S673)。具体的には、メインCPU101は、入賞作動フラグ格納領域内の「ANY役」に対応するビットに「1」をセットする。

【0939】

S673の処理後、又は、S671或いはS672がNO判定の場合、メインCPU101は、入賞作動フラグ格納領域(図28～図30参照)のアドレスとして、その最後尾の格納領域のアドレスに「1」を加算したアドレスをセットし、停止禁止データをセットし、入賞作動フラグデータ長(入賞作動フラグ格納領域のデータ長:本実施形態では、12バイト)をセットする(S674)。次いで、メインCPU101は、ストックボタン作

10

20

30

40

50

動カウンタの値、及び、ストップボタン作動状態を取得する（S 6 7 5）。なお、ストップボタン作動カウンタは、停止操作が検出されているストップボタンの数を管理するためのカウンタである。また、ストップボタン作動状態は、作動ストップボタン格納領域（図 3 3 参照）を参照することにより取得される。

【 0 9 4 0 】

次いで、メインCPU 1 0 1 は、セットされている入賞作動フラグ格納領域のアドレスを 1 減算（- 1 更新）する（S 6 7 6）。次いで、メインCPU 1 0 1 は、セットされている入賞作動フラグ格納領域とそれに対応する当り要求フラグ格納領域（図 2 8 ~ 図 3 0 参照）とから当り要求フラグデータを生成し、該生成された当り要求フラグデータに基づいて禁止入賞作動位置を生成する（S 6 7 7）。

10

【 0 9 4 1 】

次いで、メインCPU 1 0 1 は、停止操作位置が禁止入賞作動位置であるか否かを判別する（S 6 7 8）。

【 0 9 4 2 】

S 6 7 8 において、メインCPU 1 0 1 が、停止操作位置が禁止入賞作動位置でないと判別したとき（S 6 7 8 がNO判定の場合）、メインCPU 1 0 1 は、後述のS 6 8 4 の処理を行う。一方、S 6 7 8 において、メインCPU 1 0 1 が、停止操作位置が禁止入賞作動位置であると判別したとき（S 6 7 8 がYES判定の場合）、メインCPU 1 0 1 は、ストップボタン作動カウンタの値が第3停止の値であるか否かを判別する（S 6 7 9）。

【 0 9 4 3 】

S 6 7 9 において、メインCPU 1 0 1 が、ストップボタン作動カウンタの値が第3停止の値であると判別したとき（S 6 7 9 がYES判定の場合）、メインCPU 1 0 1 は、後述のS 7 0 5 の処理を行う。一方、S 6 7 9 において、メインCPU 1 0 1 が、ストップボタン作動カウンタの値が第3停止の値でないと判別したとき（S 6 7 9 がNO判定の場合）、メインCPU 1 0 1 は、ストップボタン作動カウンタの値が第2停止の値であるか否かを判別する（S 6 8 0）。

20

【 0 9 4 4 】

S 6 8 0 において、メインCPU 1 0 1 が、ストップボタン作動カウンタの値が第2停止の値でないと判別したとき（S 6 8 0 がNO判定の場合）、メインCPU 1 0 1 は、後述のS 6 8 4 の処理を行う。一方、S 6 8 0 において、メインCPU 1 0 1 が、ストップボタン作動カウンタの値が第2停止の値であると判別したとき（S 6 8 0 がYES判定の場合）、メインCPU 1 0 1 は、右リール3 R の停止後であるか否かを判別する（S 6 8 1）。

30

【 0 9 4 5 】

S 6 8 1 において、メインCPU 1 0 1 が、右リール3 R の停止後であると判別したとき（S 6 8 1 がYES判定の場合）、メインCPU 1 0 1 は、後述のS 6 8 4 の処理を行う。一方、S 6 8 1 において、メインCPU 1 0 1 が、右リール3 R の停止後でないと判別したとき（S 6 8 1 がNO判定の場合）、メインCPU 1 0 1 は、当り要求フラグが「ANY 役」の干渉を受ける可能性があるフラグでないか否か（内部当籤役に係る図柄組合せ（入賞役）に「ANY 役」が含まれないか否か）を判別する（S 6 8 2）。

40

【 0 9 4 6 】

S 6 8 2 において、メインCPU 1 0 1 が、当り要求フラグが「ANY 役」の干渉を受ける可能性があるフラグでないと判別したとき（S 6 8 2 がYES判定の場合）、メインCPU 1 0 1 は、後述のS 6 8 4 の処理を行う。一方、S 6 8 2 において、メインCPU 1 0 1 が、当り要求フラグが「ANY 役」の干渉を受ける可能性があるフラグであると判別したとき（S 6 8 2 がNO判定の場合）、メインCPU 1 0 1 は、現チェックが「ANY 役」を含む当り要求フラグのチェック時であるか否かを判別する（S 6 8 3）。

【 0 9 4 7 】

S 6 8 3 において、メインCPU 1 0 1 が、現チェックが「ANY 役」を含む当り要求フラグのチェック時であると判別したとき（S 6 8 3 がYES判定の場合）、メインCPU

50

101は、後述のS705の処理を行う。

【0948】

一方、S683において、メインCPU101が、現チェックが「ANY役」を含む当り要求フラグのチェック時でないとは判断したとき（S683がNO判定の場合）、S678或いはS680がNO判定の場合、又は、S681或いはS682がYES判定の場合、メインCPU101は、入賞作動フラグデータ長を1減算する（S684）。次いで、メインCPU101は、入賞作動フラグデータ長が「0」であるか否かを判断する（S685）。

【0949】

S685において、メインCPU101が、入賞作動フラグデータ長が「0」でないと判断したとき（S685がNO判定の場合）、メインCPU101は、処理をS676の処理に戻し、S676以降の処理を繰り返す。

10

【0950】

一方、S685において、メインCPU101が、入賞作動フラグデータ長が「0」であると判断したとき（S685がYES判定の場合）、メインCPU101は、停止制御用引込要求フラグ設定処理を行う（S686）。この処理は、メインCPU101により、図136Bのソースプログラムで規定されている各処理が順次実行されることにより行われる。それゆえ、この処理の中では、図133で説明した論理積演算処理が行われる。なお、S686の処理内で実行される論理積演算処理では、上述のように、当り（引込）要求フラグ格納領域のデータが「論理積先データ」にセットされ、入賞作動フラグ格納領域のデータが「論理積元データ」にセットされ、「論理積回数」には、RT作動組み合わせ表示フラグのデータ長（1バイト）のバイト数「1」がセットされる。RT作動組み合わせ表示フラグは、入賞作動フラグ格納領域において、RT移行に係る図柄組合せが規定された格納領域のことであり、本実施形態では、図28～図30に示すように格納領域11のみとなる。

20

【0951】

次いで、メインCPU101は、引込優先順位テーブルアドレス格納領域を参照して、引込優先順位テーブルを取得する（S687）。この処理は、メインCPU101により、図136Cのソースプログラムで規定されている各処理が順次実行されることにより行われる。それゆえ、この処理では、現在セットされているアドレスに、引込優先順位データの初期値「1（001H）」が設定されるとともに、図137に示す、先頭アドレスが「dPLVLTB00」～「dPLVLTB05」のいずれかとなるブロックに格納された引込優先順位テーブルが取得される。なお、図137に示す、先頭アドレスが「dPLVLTB00」～「dPLVLTB05」となるブロックに格納された引込優先順位テーブルは、それぞれ、図27に記載の引込優先順位テーブル番号「00」～「05」の引込優先順位テーブルに対応する。

30

【0952】

次いで、メインCPU101は、現在セットされているアドレスに格納されている引込優先順位テーブルのデータが、エンドコード（000H）であるか否かを判断する（S688）。

40

【0953】

S688において、メインCPU101が、現在セットされているアドレスに格納されている引込優先順位テーブルのデータが、エンドコードであると判断したとき（S688がYES判定の場合）、メインCPU101は、後述のS705の処理を行う。一方、S688において、メインCPU101が、現在セットされているアドレスに格納されている引込優先順位テーブルのデータが、エンドコードでないと判断したとき（S688がNO判定の場合）、メインCPU101は、入賞作動フラグ格納領域をセットする（S689）。

【0954】

次いで、メインCPU101は、現在セットされているアドレスに基づいて、引込優先順

50

位テーブルから引込優先順位データを取得する（S 6 9 0）。次いで、メインCPU 1 0 1は、引込優先順位テーブルのブロックカウンタをセットする（S 6 9 1）。本実施形態では、この処理において、メインCPU 1 0 1は、引込優先順位テーブルのブロックカウンタの値に「2」をセットする。

【0 9 5 5】

次いで、メインCPU 1 0 1は、引込優先順位テーブルのチェック回数をセットし、参照する引込優先順位テーブルのアドレスを1加算（+1更新）する（S 6 9 2）。本実施形態では、この処理において、メインCPU 1 0 1は、引込優先順位テーブルのチェック回数に「8」（図1 3 7に示す引込優先順位テーブルに規定されているチェックデータのビット数）をセットする。

10

【0 9 5 6】

次いで、メインCPU 1 0 1は、更新された引込優先順位テーブルのアドレスに基づいて、チェックデータ（図1 3 7参照）を取得し、チェックデータからチェックビットを抽出する（S 6 9 3）。なお、本実施形態では、ここで抽出するチェックビットは、チェックデータのビット0に対応する。例えば、S 6 9 0の処理において、先頭アドレスが「d P L V L T B 0 0」であるブロックに規定された引込優先順位テーブルから引込優先順位データ「0 3 E H」が取得された場合、S 6 9 3の処理では、チェックデータとして「1 0 0 0 1 0 0 0 B」が取得され、チェックビットの値として「0」が抽出される。

【0 9 5 7】

次いで、メインCPU 1 0 1は、抽出されたチェックビットの値が「1」であるか否かを判別する（S 6 9 4）。

20

【0 9 5 8】

S 6 9 4において、メインCPU 1 0 1が、抽出されたチェックビットの値が「1」でないと判別したとき（S 6 9 4がNO判定の場合）、メインCPU 1 0 1は、後述のS 6 9 9の処理を行う。一方、S 6 9 4において、メインCPU 1 0 1が、抽出されたチェックビットの値が「1」であると判別したとき（S 6 9 4がYES判定の場合）、メインCPU 1 0 1は、参照する引込優先順位テーブルのアドレスを1加算（+1更新）し、更新後のアドレスに基づいて、引込優先順位テーブルから判定データ（図1 3 7中の「フラグ判定データ」）を取得する（S 6 9 5）。

【0 9 5 9】

次いで、メインCPU 1 0 1は、S 6 9 5で取得した判定データに基づいて、現在取得されている入賞作動フラグデータが判定対象であるか否かを判別する（S 6 9 6）。この処理では、メインCPU 1 0 1は、現在取得されている入賞作動フラグデータと、判定データとを比較し、前者が後者に対応するものであるか否かを判定し、前者が後者に対応するものである場合には、現在取得されている入賞作動フラグデータが判定対象であると判定する。

30

【0 9 6 0】

S 6 9 6において、メインCPU 1 0 1が、入賞作動フラグデータが判定対象でないと判別したとき（S 6 9 6がNO判定の場合）、メインCPU 1 0 1は、後述のS 6 9 9の処理を行う。一方、S 6 9 6において、メインCPU 1 0 1が、入賞作動フラグデータが判定対象であると判別したとき（S 6 9 6がYES判定の場合）、メインCPU 1 0 1は、引込優先順位データの更新処理を行う（S 6 9 7）。この処理では、メインCPU 1 0 1は、S 6 9 7で取得した判定データに対応付けられた引込優先順位データで、現在セットされている引込優先順位データを更新（上書き）する。

40

【0 9 6 1】

次いで、メインCPU 1 0 1は、チェックデータの更新処理を行う（S 6 9 8）。この処理では、メインCPU 1 0 1は、チェックデータを1ビットだけ右方向（ビット7からビット0に向かう方向）にシフトする。なお、この処理において、シフト後のチェックデータのビット7には、「0」がセットされる。

【0 9 6 2】

50

S 6 9 8 の処理後、又は、S 6 9 4 或いは S 6 9 6 が N O 判定の場合、メイン C P U 1 0 1 は、チェックデータにチェック対象のビット（「1」がセットされているビット）があるか否かを判別する（S 6 9 9）。

【0963】

S 6 9 9 において、メイン C P U 1 0 1 が、チェックデータにチェック対象のビットがないと判別したとき（S 6 9 9 が N O 判定の場合）、メイン C P U 1 0 1 は、後述の S 7 0 2 の処理を行う。一方、S 6 9 9 において、メイン C P U 1 0 1 が、チェックデータにチェック対象のビットがあると判別したとき（S 6 9 9 が Y E S 判定の場合）、メイン C P U 1 0 1 は、チェックする入賞作動フラグ格納領域のアドレスを1加算（+1更新）し、チェック回数を1減算する（S 7 0 0）。

10

【0964】

次いで、メイン C P U 1 0 1 は、チェック回数が「0」であるか否かを判別する（S 7 0 1）。S 7 0 1 において、メイン C P U 1 0 1 が、チェック回数が「0」でないと判別したとき（S 7 0 1 が N O 判定の場合）、メイン C P U 1 0 1 は、処理を S 6 9 8 の処理に戻し、S 6 9 8 以降の処理を繰り返す。

【0965】

一方、S 7 0 1 において、メイン C P U 1 0 1 が、チェック回数が「0」であると判別したとき（S 7 0 1 が Y E S 判定の場合）、メイン C P U 1 0 1 は、現在参照している入賞作動フラグ格納領域のアドレスにチェック回数の初期値「8」を加算して入賞作動フラグ格納領域のアドレスを更新し、ブロックカウンタの値を1減算する（S 7 0 2）。次いで、メイン C P U 1 0 1 は、ブロックカウンタの値が「0」であるか否かを判別する（S 7 0 3）。

20

【0966】

S 7 0 3 において、メイン C P U 1 0 1 が、ブロックカウンタの値が「0」でないと判別したとき（S 7 0 3 が N O 判定の場合）、メイン C P U 1 0 1 は、処理を S 6 9 2 の処理に戻し、S 6 9 2 以降の処理を繰り返す。

【0967】

一方、S 7 0 3 において、メイン C P U 1 0 1 が、ブロックカウンタの値が「0」であると判別したとき（S 7 0 3 が Y E S 判定の場合）、メイン C P U 1 0 1 は、参照する引込優先順位テーブルのアドレスを1加算（+1更新）する（S 7 0 4）。例えば、現在参照している引込優先順位テーブルが、先頭アドレスが「d P L V L T B 0 0」であるブロックに規定された引込優先順位テーブルである場合、この処理により、参照する引込優先順位テーブルが、先頭アドレスが「d P L V L T B 0 1」であるブロックに規定された引込優先順位テーブルに変更される。そして、S 7 0 4 の処理後、メイン C P U 1 0 1 は、処理を S 6 8 8 の処理に戻し、S 6 8 8 以降の処理を繰り返す。

30

【0968】

ここで再度、S 6 7 9、S 6 8 3 又は S 6 8 8 の処理に戻って、S 6 7 9、S 6 8 3 又は S 6 8 8 が Y E S 判定の場合、メイン C P U 1 0 1 は、この時点でセットされている引込優先順位データを、最終的な引込優先順位データとしてセットする（S 7 0 5）。なお、S 6 7 9 又は S 6 8 3 が Y E S 判定の場合、メイン C P U 1 0 1 は、最終的な引込優先順位データとして「0（00H）」をセットする。この場合、引込優先順位データ「0（00H）」にはエンドコードが割り付けられているので、引込データ無し（停止禁止）がセットされる。そして、S 7 0 5 の処理後、メイン C P U 1 0 1 は、引込優先順位取得処理を終了し、処理を引込優先順位格納処理（図126参照）の S 6 2 8 に移す。

40

【0969】

本実施形態では、上述のようにして引込優先順位取得処理が行われる。なお、上述した引込優先順位取得処理中の S 6 8 0 ~ S 6 8 3 の「A N Y 役」の引込優先対応処理は、メイン C P U 1 0 1 が、図136Aのソースプログラムで規定されている各ソースコードを順次実行することにより行われる。その中で、例えば、S 6 8 3 の判定処理は、ソースプログラム上において、「J C P」命令（所定の判定命令）により実行される。なお、「J C

50

「P」命令は、比較命令相当の動作を実行する命令であり、メインCPU101専用命令コードである。

【0970】

ソースプログラム上において、例えば、ソースコード「JCP cc, A, n, e」が実行されると、Aレジスタの内容（格納データ）と、整数nとを比較し、その比較結果が、ccの条件となれば、処理をeで指定されるアドレスにジャンプさせる。なお、「JCP」命令の「ccの条件」には、フラグ・レジスタF内のキャリーフラグの状態及びゼロフラグの状態の一方が指定される（図11参照）。例えば、ccに「C」が指定されていれば、ccの条件はキャリーフラグが「1」（オン状態）であることを意味し、ccに「NC」が指定されていれば、ccの条件はキャリーフラグが「0」（オフ状態）であることを意味する。また、例えば、ccに「Z」が指定されていれば、ccの条件はゼロフラグが「1」（オン状態）であることを意味し、ccに「NZ」が指定されていれば、ccの条件はゼロフラグが「0」（オフ状態）であることを意味する。

10

【0971】

「ANY役」の引込優先対応処理のソースプログラム上において、図136Aに示すように、「JCP」命令を用いた場合、アドレス設定に係る命令を省略することができる（アドレス設定に係る命令を別途設ける必要がなくなる）ので、「ANY役」の引込優先対応処理の処理効率を高めることができるとともに、ソースプログラムの容量（メインROM102の使用容量）を低減することができる。

【0972】

また、上述した引込優先順位取得処理中のS686の停止制御用引込要求フラグ設定処理は、メインCPU101が、図136Bのソースプログラムで規定されている各ソースコードを順次実行することにより行われる。S686の停止制御用引込要求フラグ設定処理では、図136Bに示すように、メインCPU101専用命令コードである、Qレジスタ（拡張レジスタ）を用いてアドレス指定を行う「LDQ」命令、及び、「CALLF」命令が利用される。

20

【0973】

それゆえ、S686の停止制御用引込要求フラグ設定処理において、Qレジスタ（拡張レジスタ）を用いた「LDQ」命令を用いることにより、直値により、メインROM102、メインRAM103やメモリーマップI/Oにアクセスすることができる。この場合、ソースプログラム上において、アドレス設定に係る命令を省略することができ、ソースプログラムの容量（メインROM102の使用容量）を低減することができる。また、「CALLF」命令は、上述のように、2バイトの命令コードである。それゆえ、停止制御用引込要求フラグ設定処理において、これらのメインCPU101専用命令コードを使用することにより、処理の効率化を図ることができ、限られたメインRAM103の容量を有効活用することができる。

30

【0974】

さらに、本実施形態では、優先引込順位取得処理中のS686の停止制御用引込要求フラグ設定処理において、「CALLF」命令で指定するジャンプ先の論理積演算処理のアドレス「SB_DAND_00」は、上記図126で説明した引込優先順位格納処理中のS626の論理積演算処理において「CALLF」命令で指定するジャンプ先のアドレスと同じである（図127参照）。すなわち、本実施形態では、優先引込順位取得処理中のS686の停止制御用引込要求フラグ設定処理で行う論理積演算処理を実行するためのソースプログラムが、引込優先順位格納処理中のS626で行う論理積演算処理を実行するためのソースプログラムと同じであり、S686及びS626の両処理において、論理積演算処理のソースプログラムが共有化（モジュール化）されている。この場合、S686及びS626の両処理において、それぞれ別個に論理積演算処理のソースプログラムを設ける必要が無くなるので、その分、ソースプログラムの容量（メインROM102の使用容量）を低減することができる。この結果、本実施形態では、メインROM102において、空き容量を確保する（増大させる）ことができ、増えた空き容量を活用して、遊技性を

40

50

高めることが可能になる。

【0975】

また、上述した引込優先順位取得処理中のS687の引込優先順位テーブル（図137参照）の取得処理は、メインCPU101が、図136Cのソースプログラムで規定されている各ソースコードを順次実行することにより行われる。そして、S687の引込優先順位テーブルの取得処理では、図136Cに示すように、メインCPU101専用命令コードである、Qレジスタ（拡張レジスタ）を用いてアドレス指定を行う「LDQ」命令が利用される。

【0976】

それゆえ、S687の引込優先順位テーブルの取得処理においても、「LDQ」命令の使用により、ソースプログラム上において、アドレス設定に係る命令を省略することができる。その結果、引込優先順位テーブルの取得処理の効率化を図ることができるとともに、ソースプログラムの容量（メインROM102の使用容量）を低減することができる。

10

【0977】

上述のように、本実施形態の優先引込順位取得処理中の上記各種処理では、上述したメインCPU101専用の各種命令コードが適宜用いられ、対応する処理の効率化及びソースプログラムの容量の削減を実現している。その結果、本実施形態では、主制御回路90のプログラム処理速度の効率化と容量の削減とを図ることができ、削減した容量に対応する空き領域を活用して、遊技性を高めることが可能となる。

【0978】

[リール停止制御処理]

次に、図138～図140を参照して、メインフロー（図82参照）中のS213で行うリール停止制御処理について説明する。なお、図138は、リール停止制御処理の手順を示すフローチャートである。図139は、リール停止制御処理中の後述のS711～S716の処理を実行するためのソースプログラムの一例を示す図であり、図140は、リール停止制御処理中の後述のS726の処理を実行するためのソースプログラムの一例を示す図である。

20

【0979】

まず、メインCPU101は、リール停止可能信号OFF処理を行う（S711）。この処理では、メインCPU101は、主に、リール停止可能信号OFFデータのポート出力処理を行う。また、この処理は、メインRAM103の規定外作業領域を使用して行われる。なお、リール停止可能信号OFF処理の詳細については、後述の図141を参照しながら後で説明する。

30

【0980】

次いで、メインCPU101は、全リールの回転速度が所定の一定速度に到達したか否か（「定速」になったか否か）を判別する（S712）。S712において、メインCPU101が、全リールの回転速度が「定速」になっていないと判別したとき（S712がNO判定の場合）、メインCPU101は、S712の処理を繰り返す。

【0981】

一方、S712において、メインCPU101が、全リールの回転速度が「定速」になったと判別したとき（S712がYES判定の場合）、メインCPU101は、リール停止可能信号ON処理を行う（S713）。この処理では、メインCPU101は、主に、リール停止可能信号ONデータのポート出力処理を行う。また、この処理は、メインRAM103の規定外作業領域を使用して行われる。なお、リール停止可能信号ON処理の詳細については、後述の図142を参照しながら後で説明する。

40

【0982】

次いで、メインCPU101は、有効なストップボタンが押されたか否かを判別する（S714）。

【0983】

S714において、メインCPU101が、有効なストップボタンが押されていないと判

50

別したとき（S714がNO判定の場合）、メインCPU101は、処理をS713の処理に戻し、S713以降の処理を繰り返す。一方、S714において、メインCPU101が、有効なストップボタンが押されたと判別したとき（S714がYES判定の場合）、メインCPU101は、作動ストップボタン格納領域（図33参照）を更新し、ストップボタン未作動カウンタの値を1減算する（S715）。

【0984】

次いで、メインCPU101は、作動ストップボタンから検索対象リールを決定する（S716）。また、この処理では、検索対象リールのリール制御管理情報が格納される回胴制御データ格納領域のアドレス（先頭アドレス）セット処理も行われる（図139中のソースコード「LDQ IX, WR1_CTRL - (WR2_CTRL - WR1_CTRL)」参照）。

10

【0985】

次いで、メインCPU101は、リール停止可能信号OFF処理を行う（S717）。この処理は、上記S711と同様に、メインRAM103の規定外作業領域を使用して行われる。なお、リール停止可能信号OFF処理の詳細については、後述の図141を参照しながら後で説明する。次いで、メインCPU101は、図柄カウンタの値に基づいて停止開始位置をメインRAM103に格納する（S718）。

【0986】

次いで、メインCPU101は、リール停止選択処理を行う（S719）。詳細な説明は省略するが、この処理では、メインCPU101は、滑り駒数の選択処理を行う。

20

【0987】

次いで、メインCPU101は、停止開始位置と、S719で決定された滑り駒数とに基づいて停止予定位置を決定し、該決定した停止予定位置をメインRAM103に格納する（S720）。この処理では、メインCPU101は、停止開始位置に滑り駒数を加算し、その加算結果を停止予定位置とする。

【0988】

次いで、メインCPU101は、図柄コード格納処理を実行する（S721）。この処理では、停止予定位置に対応する図柄コードが図柄コード格納領域に格納される。次いで、メインCPU101は、制御対象のリールが最終停止（第3停止）のリールであるか否かを判別する（S722）。この処理では、メインCPU101は、ストップボタン未作動カウンタの値に基づいて、制御対象のリールが最終停止（第3停止）のリールであるか否かを判別し、ストップボタン未作動カウンタの値が「0」であるときには、制御対象のリールが最終停止のリールであると判定する。

30

【0989】

S722において、メインCPU101が、制御対象のリールが最終停止のリールでないと判別したとき（S722がNO判定の場合）、メインCPU101は、制御変更処理を行う（S723）。この処理では、特定の停止位置にあった場合に、リールの停止に用いる停止情報群が更新される。次いで、メインCPU101は、図126で説明した引込優先順位格納処理を行う（S724）。

【0990】

次いで、メインCPU101は、停止間隔残時間待機処理を行う（S725）。この処理では、メインCPU101は、予め設定された所定のリール停止間隔時間が経過するまで、待機処理を行う。そして、S725の処理後、メインCPU101は、処理をS711の処理に戻し、S711以降の処理を繰り返す。

40

【0991】

ここで再度、S722の処理に戻って、S722において、メインCPU101が、制御対象のリールが最終停止のリールであると判別したとき（S722がYES判定の場合）、メインCPU101は、全リールの励磁が停止状態であるか否かを判別する（S726）。S726において、メインCPU101が、全リールの励磁が停止状態でないと判別したとき（S726がNO判定の場合）、メインCPU101は、S726の処理を繰り返す。

50

返す。

【0992】

一方、S726において、メインCPU101が、全リールの励磁が停止状態であると判別したとき（S726がYES判定の場合）、メインCPU101は、第3停止操作されたストップボタンがオン状態のままである（ストップボタンが放されていない）か否かを判別する（S727）。S727において、メインCPU101が、第3停止操作されたストップボタンがオン状態のままであると判別したとき（S727がYES判定の場合）、メインCPU101は、S727の処理を繰り返す。一方、S727において、メインCPU101が、第3停止操作されたストップボタンがオン状態のままでないとして判別したとき（S727がNO判定の場合）、メインCPU101は、リール停止制御処理を終了し、処理をメインフロー（図82参照）のS214に移す。

10

【0993】

本実施形態では、上述のようにしてリール停止制御処理が行われる。なお、上述したリール停止制御処理中のS711～S716の処理は、メインCPU101が、図139のソースプログラムで規定されている各ソースコードを順次実行することにより行われる。図139に示すように、本実施形態のリール停止制御処理のソースプログラムでは、メインCPU101専用命令コードである、例えば、Qレジスタ（拡張レジスタ）を用いてアドレス指定を行う「LDQ」命令や、「CALLF」命令が用いられる。

【0994】

それゆえ、リール停止制御処理において、このようなメインCPU101専用命令コードを用いることにより、リール制御処理のソースプログラムの容量を削減することができる。すなわち、本実施形態では、主制御回路90におけるプログラム処理速度の効率化と容量の削減とを行うことが可能となり、削減した容量に応じて増加したメインROM102の空き領域を活用して、遊技性を高めることが可能となる。

20

【0995】

また、上述したリール停止制御処理中のS726の判定処理は、メインCPU101が、図140のソースプログラムで規定されている各ソースコードを順次実行することにより行われる。この処理は、図140に示すように、ソースコード上では、「LDQ」命令、「ORQ」命令（所定の論理和演算命令）を用いて実行される。

30

【0996】

なお、「ORQ」命令は、論理和演算を行う命令コードであり、Qレジスタ（拡張レジスタ）を用いてアドレス指定を行うメインCPU101専用命令コードである。そして、ソースプログラム上において、例えば、ソースコード「ORQ（k）」が実行されると、Qレジスタの格納データ（上位側アドレス値）及び1バイトの整数k（直値：下位側アドレス値）で指定されたアドレスのメモリの内容（格納データ）と、Aレジスタの内容（格納データ）との論理和演算が行われ、その演算結果がAレジスタに記憶される。

【0997】

それゆえ、リール停止制御処理中のS726の判定処理において、まず、図140中のソースコード「LDQ A, (.LOW.wR1__TIM)」が実行されると、Qレジスタの格納データと、整数値「.LOW.wR1__TIM」とで指定されるアドレスのメモリの内容（第1リールの励磁タイマー値）がAレジスタにロードされる。なお、本実施形態では、メインRAM103内における第1リールの励磁タイマー値が格納された領域のアドレスは、「F032h」である。そして、上述したS726の判定処理では、LDQ命令実行時に予めQレジスタに、アドレス「wR1__TIM(F032h)」の上位側アドレス値「F0h」がセットされ、kの値（直値）には、下位側アドレス値（「.LOW.wR2__TIM」=32h）が代入される。

40

【0998】

次いで、図140中のソースコード「ORQ(.LOW.wR2__TIM)」が実行されると、Qレジスタの格納データ（F0h）と、第2リールの励磁タイマー値が格納され

50

た領域のアドレス「wR2__TIM(F03Dh)」の下位側アドレス値(3Dh)で指定されたアドレスのメモリの内容(第2リールの励磁タイマー値)と、Aレジスタの内容(第1リールの励磁タイマー値)との論理和演算が行われ、その演算結果(第1リールの励磁タイマー値と第2リールの励磁タイマー値との合成結果)がAレジスタに記憶される。次いで、図140中のソースコード「ORQ(.LOW.wR3__TIM)」が実行されると、Qレジスタの格納データ(F0h)と、第3リールの励磁タイマー値が格納された領域のアドレス「wR3__TIM(F048h)」の下位側アドレス値(48h)で指定されたアドレスのメモリの内容(第3リールの励磁タイマー値)と、Aレジスタの内容(第1リールの励磁タイマー値と第2リールの励磁タイマー値との合成結果)との論理和演算が行われ、その演算結果(第1～第3リールの励磁タイマー値の合成結果)がAレジスタに記憶される。

10

【0999】

上述のように、本実施形態では、リール(回胴)の停止状態のチェック処理において、Qレジスタ(拡張レジスタ)を用いた各種メインCPU101専用命令コードが用いられる。それゆえ、これらのメインCPU101専用命令コードを用いることにより、直値により、メインROM102、メインRAM103やメモリーマップI/Oにアクセスすることができ、ソースプログラム上において、アドレス設定に係る命令を省略することができ、その分、ソースプログラムの容量(メインROM102の使用容量)を低減することができる。この結果、本実施形態では、メインROM102において、空き容量を確保する(増大させる)ことができ、増えた空き容量を活用して、遊技性を高めることが可能になる。

20

【1000】

上述のように、本実施形態のリール停止制御処理中の上記各種処理では、上述したメインCPU101専用の各種命令コードが適宜用いられ、対応する処理の効率化及びソースプログラムの容量の削減を実現している。その結果、本実施形態では、主制御回路90のプログラム処理速度の効率化と容量の削減とを図ることができ、削減した容量に対応する空き領域を活用して、遊技性を高めることが可能となる。

【1001】

[リール停止可能信号OFF処理]

次に、図141を参照して、リール停止制御処理(図138参照)中のS711又はS717で行うリール停止可能信号OFF処理について説明する。なお、図141は、リール停止可能信号OFF処理の手順を示すフローチャートである。

30

【1002】

まず、メインCPU101は、スタックポインタ(SP)にセットされているメインRAM103のスタックエリア(図12C参照)のアドレスを退避させる(S731)。次いで、メインCPU101は、スタックポインタ(SP)に規定外スタックエリアのアドレスをセットする(S732)。

【1003】

次いで、メインCPU101は、全てのレジスタにセットされているデータを退避させる(S733)。次いで、メインCPU101は、リール停止可能信号OFFデータのセット処理を行う(S734)。

40

【1004】

次いで、メインCPU101は、規定外ポート出力処理を行う(S735)。この処理では、メインCPU101は、リール停止可能信号OFFデータに基づいて、後述のOFF出力データ(出力オフモードデータ)の生成及び出力処理を行う。なお、この処理は、メインRAM103の規定外作業領域を使用して行われる。規定外ポート出力処理の詳細については、後述の図143を参照しながら後で説明する。

【1005】

次いで、メインCPU101は、S733で退避させた全レジスタのデータを復帰させる(S736)。次いで、メインCPU101は、S731で退避させたスタックエリアの

50

アドレスをスタックポインタ (S P) にセットする (S 7 3 7) 。

【 1 0 0 6 】

そして、 S 7 3 7 の処理後、メイン C P U 1 0 1 は、リール停止可能信号 O F F 処理を終了する。この際、実行したリール停止可能信号 O F F 処理がリール停止制御処理 (図 1 3 8 参照) 中の S 7 1 1 の処理である場合には、メイン C P U 1 0 1 は、処理をリール停止制御処理中の S 7 1 2 の処理に移す。一方、実行したリール停止可能信号 O F F 処理がリール停止制御処理 (図 1 3 8 参照) 中の S 7 1 7 の処理である場合には、メイン C P U 1 0 1 は、処理をリール停止制御処理中の S 7 1 8 の処理に移す。

【 1 0 0 7 】

[リール停止可能信号 O N 処理]

次に、図 1 4 2 を参照して、リール停止制御処理 (図 1 3 8 参照) 中の S 7 1 3 で行うリール停止可能信号 O N 処理について説明する。なお、図 1 4 2 は、リール停止可能信号 O N 処理の手順を示すフローチャートである。

【 1 0 0 8 】

まず、メイン C P U 1 0 1 は、スタックポインタ (S P) にセットされているメイン R A M 1 0 3 のスタックエリア (図 1 2 C 参照) のアドレスを退避させる (S 7 4 1) 。次いで、メイン C P U 1 0 1 は、スタックポインタ (S P) に規定外スタックエリアのアドレスをセットする (S 7 4 2) 。

【 1 0 0 9 】

次いで、メイン C P U 1 0 1 は、全てのレジスタにセットされているデータを退避させる (S 7 4 3) 。次いで、メイン C P U 1 0 1 は、作動ストップボタン格納領域 (図 3 3 参照) を参照し、ストップボタン状態を取得する (S 7 4 4) 。次いで、メイン C P U 1 0 1 は、リール停止可能信号 O N データのセット処理を行う (S 7 4 5) 。

【 1 0 1 0 】

次いで、メイン C P U 1 0 1 は、規定外ポート出力処理を行う (S 7 4 6) 。この処理では、メイン C P U 1 0 1 は、リール停止可能信号 O N データに基づいて、後述の O N 出力データ (出力オンモードデータ) の生成及び出力処理を行う。なお、この処理は、メイン R A M 1 0 3 の規定外作業領域を使用して行われる。規定外ポート出力処理の詳細については、後述の図 1 4 3 を参照しながら後で説明する。

【 1 0 1 1 】

次いで、メイン C P U 1 0 1 は、 S 7 4 3 で退避させた全レジスタのデータを復帰させる (S 7 4 7) 。次いで、メイン C P U 1 0 1 は、 S 7 4 1 で退避させたスタックエリアのアドレスをスタックポインタ (S P) にセットする (S 7 4 8) 。そして、 S 7 4 8 の処理後、メイン C P U 1 0 1 は、リール停止可能信号 O N 処理を終了し、処理をリール停止制御処理 (図 1 3 8 参照) 中の S 7 1 4 の処理に移す。

【 1 0 1 2 】

[規定外ポート出力処理]

次に、図 1 4 3 及び図 1 4 4 を参照して、リール停止可能信号 O F F 処理 (図 1 4 1 参照) 中の S 7 3 5 及びリール停止可能信号 O N 処理 (図 1 4 2 参照) 中の S 7 4 6 で行う規定外ポート出力処理について説明する。なお、図 1 4 3 は、規定外ポート出力処理の手順を示すフローチャートである。また、図 1 4 4 は、規定外ポート出力処理を実行するためのソースプログラムの一例を示す図である。

【 1 0 1 3 】

まず、メイン C P U 1 0 1 は、ポート出力設定が O N 出力モードであるか否かを判別する (S 7 5 1) 。この処理において、メイン C P U 1 0 1 は、リール停止可能信号 O N データがセットされている場合には、ポート出力設定が O N 出力モードであると判定し、リール停止可能信号 O F F データがセットされている場合には、ポート出力設定が O N 出力モードでないと判定する。

【 1 0 1 4 】

S 7 5 1 において、メイン C P U 1 0 1 が、ポート出力設定が O N 出力モードであると判

10

20

30

40

50

別したとき（S751がYES判定の場合）、メインCPU101は、後述のS753の処理を行う。一方、S751において、メインCPU101が、ポート出力設定がON出力モードでないと判別したとき（S751がNO判定の場合）、メインCPU101は、OFF出力データ（出力オフモードデータ）の生成処理を行う（S752）。この処理では、現在、出力オン状態となっているポート（ビット）のうち、オフ状態にしたいポート（ビット）をオフ状態にするとともに、現在、出力オフ状態となっているポート（ビット）をオフ状態に維持するためのOFF出力データが生成される。

【1015】

S752の処理後又はS751がNO判定の場合、メインCPU101は、ON出力データ（出力オンモードデータ）の生成処理を行う（S753）。この処理では、現在、出力オフ状態となっているポート（ビット）のうち、オン状態にしたいポート（ビット）をオン状態にするとともに、現在、出力オン状態となっているポート（ビット）をオン状態に維持するためのON出力データが生成される。次いで、メインCPU101は、生成された出力データを指定ポートから出力する（S754）。

10

【1016】

そして、S754の処理後、メインCPU101は、規定外ポート出力処理を終了する。この際、実行した規定外ポート出力処理がリール停止可能信号OFF処理（図141参照）中のS735の処理である場合には、メインCPU101は、処理をリール停止可能信号OFF処理中のS736の処理に移す。一方、実行した規定外ポート出力処理がリール停止可能信号ON処理（図142参照）中のS746の処理である場合には、メインCPU101は、処理をリール停止可能信号ON処理中のS747の処理に移す。

20

【1017】

本実施形態では、上述のようにして規定外ポート出力処理が行われる。そして、上述した規定外ポート出力処理は、メインCPU101が、図144のソースプログラムで規定されている各ソースコードを順次実行することにより行われる。

【1018】

その中で、上述した規定外ポート出力処理中のS752のOFF出力データの生成処理は、図144中のソースコード「XOR（HL）」及び「AND（HL）」をこの順で実行することに行われる。また、上述した規定外ポート出力処理中のS753のON出力データの生成処理は、図144中のソースコード「OR（HL）」を実行することに行われる。

30

【1019】

ソースプログラム上において、このような各出力データの生成処理を行うことにより、S752のOFF出力データの生成処理後に、S753のON出力データの生成処理が行われてもS752で生成されたOFF出力データは変化しない。

【1020】

例えば、ポート出力設定がOFF出力モードであり、今回の処理でオフ状態にしたい規定外ポートを示す出力データが「00010111」（「1」がオフ状態にしたいビット）であり、現在、規定外ポートに出力されている出力データ（バックアップデータ）が「01010011」（「1」が現在、オン状態のビット）である場合、バックアップデータのビット0、ビット1及びビット5のデータを「1」から「0」にするためのOFF出力データが生成される。この場合、まず、図144中のソースコード「XOR（HL）」が実行されると、出力データ「00010111」と、バックアップデータ「01010011」との排他的論理和演算が行われ、演算結果として「01000100」が得られる。次いで、図144中のソースコード「AND（HL）」が実行されると、演算結果「01000100」とバックアップデータ「01010011」との論理積演算が行われ、演算結果「01000000」がOFF出力データとして生成される。

40

【1021】

その後、S753のON出力データの生成処理が（図144中のソースコード「OR（HL）」）が実行されると、演算結果「01000000」（OFF出力データ）と、今

50

回の処理でオン状態にしたい規定外ポートを示す出力データ「00000000」（ポート出力設定がOFF出力モードであるので、出力データの各ビットには「0」がセットされる）との論理和演算が行われ、演算結果として「01000000」が得られ、OFF出力データは変化しない。定性的には、ポート出力設定がOFF出力モードである場合、S753のON出力データの生成処理では、OFF出力データにおいて出力オン状態となっているポート（ビット）をオン状態に維持するための出力データが生成されるので、S752のOFF出力データの生成処理後に、S753のON出力データの生成処理が行われてもS752で生成されたOFF出力データは変化しない。

【1022】

[入賞検索処理]

次に、図145～図147を参照して、メインフロー（図82参照）中のS214で行う入賞検索処理について説明する。なお、図145は、入賞検索処理の手順を示すフローチャートである。図146は、入賞検索処理を実行するためのソースプログラムの一例を示す図である。また、図147は、入賞検索処理のソースプログラム上で、実際に参照される払出枚数データテーブルの構成の一例を示す図である。

【1023】

まず、メインCPU101は、図柄コード格納領域（図35参照）に格納された各格納領域のデータを、入賞作動フラグ格納領域（図28～図30参照）の対応する格納領域に転送して保存する（S761）。そして、この処理終了時点では、DEレジスタに入賞作動フラグ格納領域の最後尾のアドレスがセットされる。

【1024】

次いで、メインCPU101は、払出枚数データテーブルのアドレス（図147に示す払出枚数データテーブルの先頭アドレス「dPAYNUMTB」）をHLレジスタにセットする（S762）。次いで、メインCPU101は、払出枚数テーブル数（本実施形態では「5」）を入賞検索カウンタの初期値とし、該初期値をBレジスタにセットする（S763）。

【1025】

次いで、メインCPU101は、HLレジスタにセットされたアドレスに基づいて、メダルの払出枚数（本実施形態では、1枚、2枚、3枚及び9枚のいずれか）のデータをCレジスタにセットし、判定対象データをAレジスタにセットし、HLレジスタにセットされているアドレスに「2」を加算（+2更新）する（S764）。なお、図147に示す払出枚数データテーブルにおいて、メダルの払出枚数のデータは、「払出枚数（1, 2, 3又は9）* 2 + 0」であり、判定対象データは、払出枚数のデータの次のアドレスに格納されている1バイトのデータ（例えば「11111000B」等）である。また、以下では、Cレジスタにセットされたメダルの払出枚数のデータ「払出枚数（1, 2, 3又は9）* 2 + 0」内のデータ「0」を「判定ビット」という。この判定ビットは入賞検索の判定対象ブロックであるか否かを示す情報である。

【1026】

次いで、メインCPU101は、Cレジスタにセットされたメダルの払出枚数のデータから判定ビットの値を抽出する（S765）。次いで、メインCPU101は、抽出した判定ビットの値に基づいて、判定対象ブロックであるか否かを判別する（S766）。この処理において、メインCPU101は、抽出した判定ビットの値が「1」である場合に、判定対象ブロックであると判定する。なお、本実施形態では、図147に示すように、メダルの払出枚数に関係なく、判定ビットの値は、常に「0」であるので、S766の処理は必ずNO判定となる。

【1027】

S766において、メインCPU101が、判定対象ブロックでないと判別したとき（S766がNO判定の場合）、メインCPU101は、後述のS768の処理を行う。一方、S766において、メインCPU101が、判定対象ブロックであると判別したとき（S766がYES判定の場合）、メインCPU101は、DEレジスタにセットされてい

10

20

30

40

50

る入賞作動フラグ格納領域のアドレスを1減算(-1更新)する(S767)。

【1028】

S767の処理後又はS766がNO判定の場合、メインCPU101は、Dレジスタにセットされた入賞作動フラグ格納領域のアドレスで指定される格納領域のデータを判定データとして抽出する(S768)。

【1029】

次いで、メインCPU101は、S764でAレジスタにセットされた判定対象データと、S768で抽出した判定データとに基づいて、判定の結果が入賞であるか否かを判別する(S769)。この処理において、メインCPU101は、S764でAレジスタにセットされた判定対象データが、S768で抽出した判定データと同じであれば、判定の結果が入賞であると判定する。

10

【1030】

S769において、メインCPU101が、判定の結果が入賞でないと判別したとき(S769がNO判定の場合)、メインCPU101は、後述のS776の処理を行う。一方、S769において、メインCPU101が、判定の結果が入賞であると判別したとき(S769がYES判定の場合)、メインCPU101は、現遊技が3枚遊技(メダルのベット枚数が3枚である遊技)であるか否かを判別する(S770)。

【1031】

S770において、メインCPU101が、現遊技が3枚遊技であると判別したとき(S770がYES判定の場合)、メインCPU101は、後述のS772の処理を行う。一方、S770において、メインCPU101が、現遊技が3枚遊技でないと判別したとき(S770がNO判定の場合)、メインCPU101は、2枚遊技(メダルのベット枚数が2枚である遊技)の払出枚数(2枚)をCレジスタにセットする(S771)。

20

【1032】

S771の処理後又はS770がYES判定の場合、メインCPU101は、払出枚数の更新処理を行う(S772)。具体的には、メインCPU101は、現在の入賞枚数カウンタの値に、Cレジスタにセットされたメダルの払出枚数を加算し、加算後の値を払出枚数にセットする。

【1033】

次いで、メインCPU101は、払出枚数の値が最大払出枚数「10」未満であるか否かを判別する(S773)。

30

【1034】

S773において、メインCPU101が、払出枚数の値が最大払出枚数「10」未満であると判別したとき(S773がYES判定の場合)、メインCPU101は、後述のS775の処理を行う。一方、S773において、メインCPU101が、払出枚数の値が最大払出枚数「10」未満でないと判別したとき(S773がNO判定の場合)、メインCPU101は、払出枚数に最大払出枚数「10」をセットする(S774)。

【1035】

S774の処理後又はS773がYES判定の場合、メインCPU101は、払出枚数を入賞枚数カウンタに保存する(S775)。

40

【1036】

S775の処理後又はS769がNO判定の場合、メインCPU101は、他の入賞があるか否かを判別する(S776)。S776において、メインCPU101が、他の入賞があると判別したとき(S776がYES判定の場合)、メインCPU101は、処理をS769の処理に戻し、S769以降の処理を繰り返す。

【1037】

一方、S776において、メインCPU101が、他の入賞がないと判別したとき(S776がNO判定の場合)、メインCPU101は、入賞検索カウンタの値を1減算(-1更新)する(S777)。なお、本実施形態のように、有効ラインが1本である場合には、複数の小役が重複して入賞することがないので、S776の判定処理は必ずNO判定と

50

なる。

【1038】

次いで、メインCPU101は、入賞検索カウンタの値が「0」であるか否かを判別する(S778)。

【1039】

S778において、メインCPU101が、入賞検索カウンタの値が「0」でないと判別したとき(S778がNO判定の場合)、メインCPU101は、処理をS764の処理に戻し、S764以降の処理を繰り返す。一方、S778において、メインCPU101が、入賞検索カウンタの値が「0」であると判別したとき(S778がYES判定の場合)、メインCPU101は、入賞検索処理を終了し、処理をメインフロー(図82参照)中のS215の処理に移す。

10

【1040】

本実施形態では、上述のようにして入賞検索処理が行われる。そして、上述した入賞検索処理は、メインCPU101が、図146のソースプログラムで規定されている各ソースコードを順次実行することにより行われる。中でも、S764の払出枚数及び判定対象データのセット処理は、メインCPU101がソースコード「LDIN AC, (HL)」を実行することにより行われる。

【1041】

ソースプログラム上において、例えば、ソースコード「LDIN ss, (HL)」が実行されると、HLレジスタ(ペアレジスタ)にセットされたアドレス及び該アドレスに1加算したアドレスで指定されるメモリの内容(データ)がss(BC、DE、AC、AE又はBD)ペアレジスタにロードされるとともに、HLレジスタにセットされているアドレスが+2更新(2加算)される。それゆえ、図146中のソースコード「LDIN AC, (HL)」が実行されると、HLレジスタ(ペアレジスタ)にセットされたアドレス及び該アドレスに1加算したアドレスで指定されるメモリの内容(払出枚数及び判定対象データ)が、ACレジスタにロードされるとともに、HLレジスタにセットされているアドレスが+2更新(2加算)される。なお、S764の処理では、この「LDIN」命令により、Aレジスタに払出枚数のデータが格納され、Cレジスタに判定対象データが格納され、HLレジスタにセットされているアドレスが+2更新される。

20

【1042】

上述のように、本実施形態の入賞検索処理では、一つの「LDIN」命令により、データのロード処理及びアドレスの更新処理の両方を行うことができる。この場合、ソースプログラム上において、アドレス設定に係る命令を省略することができ、その分、ソースプログラムの容量(メインROM102の使用容量)を低減することができる。この結果、本実施形態では、メインROM102において、空き容量を確保する(増大させる)ことができ、増えた空き容量を活用して、遊技性を高めることが可能になる。

30

【1043】

また、上述した入賞検索処理中のS770の判定処理で参照するメダルカウンタの値の取得処理、S772の処理で参照する入賞枚数カウンタの値の取得処理、及び、S775の処理で行う入賞枚数カウンタの保存(更新)処理はいずれも、図146に示すように、Qレジスタ(拡張レジスタ)を用いてアドレス指定を行う「LDQ」命令(メインCPU101専用命令コード)により実行される。それゆえ、本実施形態の入賞検索処理では、「LDQ」命令を用いることにより、直値により、メインROM102、メインRAM103やメモリーマップI/Oにアクセスすることができるので、ソースプログラム上において、アドレス設定に係る命令を省略することができ(アドレス設定に係る命令を別途設ける必要がなくなる)、その分、ソースプログラムの容量(メインROM102の使用容量)を低減することができる。この結果、本実施形態では、メインROM102において、空き容量を確保する(増大させる)ことができ、増えた空き容量を活用して、遊技性を高めることが可能になる。

40

【1044】

50

また、上述した入賞検索処理中の S 7 6 9 の判定処理は、図 1 4 6 に示すように、ソースプログラム上において、「J S L A A」命令（所定の判定命令）により実行される。なお、「J S L A A」命令は、左シフト（S L A）命令相当の動作を実行する命令である。

【 1 0 4 5 】

ソースプログラム上において、例えば、ソースコード「J S L A A c c , e」が実行されると、c c の条件が成立すれば、処理を e で指定されるアドレスにジャンプさせる。なお、「J S L A A」命令で規定される「c c の条件」には、フラグ・レジスタ F 内のキャリーフラグの状態が指定される。例えば、c c に「C」が指定されていれば、c c の条件はキャリーフラグが「1」（オン状態）であることを意味し、c c に「N C」が指定されていれば、c c の条件はキャリーフラグが「0」（オフ状態）であることを意味する。それゆえ、図 1 4 6 中のソースコード「J S L A A N C , M N _ C K L N _ 0 6」では、キャリーフラグが「0」（オフ状態）であれば、「M N _ C K L N _ 0 6」で指定されるアドレスに処理がジャンプする。

10

【 1 0 4 6 】

また、上述した入賞検索処理中の S 7 7 0 及び S 7 7 3 の判定処理は、図 1 4 6 に示すように、ソースプログラム上において、「J C P」命令により実行される。なお、「J C P」命令は、上述のように、比較命令相当の動作を実行する命令であり、メイン C P U 1 0 1 専用命令コードである。

【 1 0 4 7 】

それゆえ、入賞検索処理のソースプログラム上において、上述した「J S L A A」命令及び「J C P」命令を用いた場合、アドレス設定に係る命令を省略することができ（アドレス設定に係る命令を別途設ける必要がなくなる）、その分、ソースプログラムの容量（メイン R O M 1 0 2 の使用容量）を低減することができる。この結果、本実施形態では、メイン R O M 1 0 2 において、空き容量を確保する（増大させる）ことができ、増えた空き容量を活用して、遊技性を高めることが可能になる。

20

【 1 0 4 8 】

[イリーガルヒットチェック処理]

次に、図 1 4 8 及び図 1 4 9 を参照して、メインフロー（図 8 2 参照）中の S 2 1 5 で行うイリーガルヒットチェック処理について説明する。なお、図 1 4 8 は、イリーガルヒットチェック処理の手順を示すフローチャートである。また、図 1 4 9 は、イリーガルヒットチェック処理を実行するためのソースプログラムの一例を示す図である。なお、イリーガルヒットとは、内部抽籤処理（図 9 2 参照）で抽籤され、図柄設定処理（図 9 7 参照）で当籤番号格納領域に格納された B B 当籤番号及び小役当籤番号（内部当籤役）に基づいて、左リール 3 L、中リール 3 C 及び右リール 3 R が、成立しえない図柄の組合せで有効ライン上に停止（図柄組合せ不成立）したことを示す用語である。

30

【 1 0 4 9 】

まず、メイン C P U 1 0 1 は、入賞作動フラグ格納領域（図 2 8 ~ 図 3 0 参照）のアドレスをセットする（S 7 8 1）。次いで、メイン C P U 1 0 1 は、入賞作動フラグ格納領域のサイズ（バイト数、本実施形態では「1 2」）を、チェックカウンタの値にセットする（S 7 8 2）。

40

【 1 0 5 0 】

次いで、メイン C P U 1 0 1 は、現在セットされている入賞作動フラグ格納領域のアドレスに基づいて、該アドレスに対応する当り要求フラグ格納領域（内部当籤役格納領域）内の格納領域に格納された内部当籤役のデータ（当り要求フラグデータ）を取得する（S 7 8 3）。次いで、メイン C P U 1 0 1 は、現在セットされている入賞作動フラグ格納領域のアドレスに格納された入賞役のデータ（入賞作動フラグデータ）と、内部当籤役のデータ（当り要求フラグデータ）とを合成する（S 7 8 4）。

【 1 0 5 1 】

なお、この合成処理では、まず、メイン C P U 1 0 1 は、入賞役のデータ（入賞作動フラグデータ）と内部当籤役のデータ（当り要求フラグデータ）との排他的論理和を求める（

50

図149に示すソースプログラム中のソースコード「XOR (HL)」)。次いで、メインCPU101は、求められた排他的論理和の算出結果と入賞役のデータ(入賞作動フラグデータ)との論理積を求め(図149に示すソースプログラム中のソースコード「AND (HL)」)、論理積の算出結果を合成結果とする。なお、イリーガルヒットエラーが発生していない場合、この合成結果の値は「0」となる。

【1052】

次いで、メインCPU101は、S784の合成処理の結果に基づいて、イリーガルヒットエラーが発生しているか否かを判別する(S785)。

【1053】

S785において、メインCPU101が、イリーガルヒットエラーが発生していないと判別したとき(S785がNO判定の場合)、メインCPU101は、参照する入賞作動フラグ格納領域のアドレスを+1更新する(S786)。次いで、メインCPU101は、チェックカウンタの値を1減算する(S787)。次いで、メインCPU101は、チェックカウンタの値が「0」であるか否かを判別する(S788)。

10

【1054】

S788において、メインCPU101が、チェックカウンタの値が「0」でないと判別したとき(S788がNO判定の場合)、メインCPU101は、処理をS783の処理に戻し、S783以降の処理を繰り返す。一方、S788において、メインCPU101が、チェックカウンタの値が「0」であると判別したとき(S788がYES判定の場合)、メインCPU101は、イリーガルヒットチェック処理を終了し、処理をメインフロー(図82参照)中のS216の処理に移す。

20

【1055】

ここで再度、S785の処理に戻って、S785において、メインCPU101が、イリーガルヒットエラーが発生していると判別したとき(S785がYES判定の場合)、メインCPU101は、図89で説明したエラー処理を行う(S789)。この処理により、情報表示器6に含まれる2桁の7セグLED(払出枚数表示用及びエラー表示用兼用)に、イリーガルヒットエラーの発生を示す2文字「EE」をエラー情報として表示するためのエラー表示データが出力される。なお、イリーガルヒットエラーの発生状態(エラー状態)は、リセットスイッチ76(図7参照)を押下することにより解除される。

【1056】

次いで、メインCPU101は、入賞枚数カウンタの値及び当り要求フラグ格納領域のデータをクリアする(S790)。そして、S790の処理後、メインCPU101は、イリーガルヒットチェック処理を終了し、処理をメインフロー(図82参照)中のS216の処理に移す。

30

【1057】

本実施形態では、上述のようにしてイリーガルヒットチェック処理が行われる。そして、上述したイリーガルヒットチェック処理は、メインCPU101が、図149のソースプログラムで規定されている各ソースコードを順次実行することにより行われる。

【1058】

なお、本実施形態では、図28~図30に示すように、入賞作動フラグ格納領域(表示役格納領域)の構成が当り要求フラグ格納領域(内部当籤役格納領域)のそれと同じであるので、入賞作動フラグ格納領域の役と内部当籤役との合成処理時にメインRAM103に配置される当り要求フラグ格納領域と入賞作動フラグ格納領域とを同一構成にすることができる。それゆえ、本実施形態のイリーガルヒットチェック処理におけるS784の演算結果(入賞役のデータと内部当籤役のデータとを合成結果)は、上述のように、ソースプログラム上において、入賞役のデータと内部当籤役のデータとを単純に論理積(「AND」命令で実行する)することにより求められる。その結果、本実施形態では、イリーガルヒットチェック処理を効率化及び簡略化することができ、主制御プログラムの空き容量を確保する(増やす)ことができ、増えた空き容量を使用して遊技性を高めることが可能になる。

40

50

【 1 0 5 9 】

[入賞チェック・メダル払出処理]

次に、図 1 5 0 及び図 1 5 1 を参照して、メインフロー（図 8 2 参照）中の S 2 1 6 で行う入賞チェック・メダル払出処理について説明する。なお、図 1 5 0 は、入賞チェック・メダル払出処理の手順を示すフローチャートである。また、図 1 5 1 は、入賞チェック・メダル払出処理中の後述の S 8 0 4 ~ S 8 0 8 の処理を実行するためのソースプログラムの一例を示す図である。

【 1 0 6 0 】

まず、メイン CPU 1 0 1 は、入賞作動コマンド生成処理を行う（S 8 0 1）。この処理では、メイン CPU 1 0 1 は、副制御回路 2 0 0 に送信する入賞作動コマンドに含まれる、種別データおよび各種通信パラメータを生成する。なお、入賞作動コマンドは、入賞作動フラグ（表示役）等を特定するパラメータを含んで構成される。

10

【 1 0 6 1 】

次いで、メイン CPU 1 0 1 は、図 7 2 で説明した通信データ格納処理を行う（S 8 0 2）。この処理により、入賞作動コマンドデータがメイン RAM 1 0 3 に設けられた通信データ格納領域（図 7 5 B 参照）に保存される。なお、入賞作動コマンドは、後述の図 1 5 8 で説明する割込処理内の通信データ送信処理により、主制御回路 9 0 から副制御回路 2 0 0 に送信される。

【 1 0 6 2 】

次いで、メイン CPU 1 0 1 は、入賞枚数カウンタの値が「0」であるか否かを判別する（S 8 0 3）。S 8 0 3 において、メイン CPU 1 0 1 が、入賞枚数カウンタの値が「0」であると判別したとき（S 8 0 3 が YES 判定の場合）、メイン CPU 1 0 1 は、入賞チェック・メダル払出処理を終了し、処理をメインフロー（図 8 2 参照）中の S 2 1 7 の処理に移す。

20

【 1 0 6 3 】

一方、S 8 0 3 において、メイン CPU 1 0 1 が、入賞枚数カウンタの値が「0」でないと判別したとき（S 8 0 3 が NO 判定の場合）、メイン CPU 1 0 1 は、メダルのクレジット枚数（貯留枚数）がその上限枚数（本実施形態では 5 0 枚）以上であるか否かを判別する（S 8 0 4）。

【 1 0 6 4 】

S 8 0 4 において、メイン CPU 1 0 1 が、メダルのクレジット枚数とその上限枚数以上でないと判別したとき（S 8 0 4 が NO 判定の場合）、メイン CPU 1 0 1 は、クレジットカウンタの値に「1」を加算（+1更新）する（S 8 0 5）。加算されたクレジットカウンタの値は、情報表示器 6 に含まれる貯留枚数表示用の 2 桁の 7 セグ LED（不図示）により表示される。次いで、メイン CPU 1 0 1 は、メダル払出枚数チェック処理を行う（S 8 0 6）。なお、メダル払出枚数チェック処理の詳細については、後述の図 1 5 2 を参照しながら後で説明する。

30

【 1 0 6 5 】

次いで、メイン CPU 1 0 1 は、メダルの払い出しが終了したか否かを判別する（S 8 0 7）。S 8 0 7 において、メイン CPU 1 0 1 が、メダルの払い出しが終了したと判別したとき（S 8 0 7 が YES 判定の場合）、メイン CPU 1 0 1 は、入賞チェック・メダル払出処理を終了し、処理をメインフロー（図 8 2 参照）中の S 2 1 7 の処理に移す。

40

【 1 0 6 6 】

一方、S 8 0 7 において、メイン CPU 1 0 1 が、メダルの払い出しが終了していないと判別したとき（S 8 0 7 が NO 判定の場合）、メイン CPU 1 0 1 は、払出間隔待機処理を行う（S 8 0 8）。この処理では、メイン CPU 1 0 1 は、予め設定されたメダル払出間隔時間（本実施形態では 6 0 . 3 3 m s e c : 後述の図 1 5 8 で説明する割込処理（1 . 1 1 7 2 m s e c 周期）の 5 4 周期分）が経過するまでウェイトする。そして、S 8 0 8 の処理後、メイン CPU 1 0 1 は、処理を S 8 0 3 の処理に戻し、S 8 0 3 以降の処理を繰り返す。

50

【1067】

ここで再度、S804の処理に戻って、S804において、メインCPU101が、メダルのクレジット枚数がある上限枚数(50枚)以上であると判別したとき(S804がYES判定の場合)、メインCPU101は、メダルの払出処理を行う(S809)。この処理により、メダルが1枚、払い出される。そして、S809の処理後、メインCPU101は、入賞チェック・メダル払出処理を終了し、処理をメインフロー(図82参照)中のS217の処理に移す。

【1068】

本実施形態では、上述のようにして入賞チェック・メダル払出処理が行われる。なお、上述した入賞チェック・メダル払出処理中のS804～S808の処理は、メインCPU101が、図151のソースプログラムで規定されている各ソースコードを順次実行することにより行われる。

10

【1069】

なお、本実施形態では、クレジットカウンタの更新(+1)後、払出動作を継続する場合、メインCPU101は、S808の処理で60.33ms間のウェイト(払出間隔待ち)処理を行うが、この処理は、ソースプログラム上では、メインCPU101がソースコード「LD BC, cTM_PAYC」及び「RST SB_W1BC_00」をこの順で実行することにより実現されている。このように、入賞チェック・メダル払出処理において、クレジットカウンタの更新(+1)後、払出動作を継続するときに60.33ms間のウェイト(払出間隔待ち)を行った場合、無駄な待ち時間を減らすことができ、遊技者の精神的負担を軽減することができる。

20

【1070】

[メダル払出枚数チェック処理]

次に、図152及び図153を参照して、入賞チェック・メダル払出処理(図150参照)中のS806で行うメダル払出枚数チェック処理について説明する。なお、図152は、メダル払出枚数チェック処理の手順を示すフローチャートである。また、図153Aは、メダル払出枚数チェック処理中の後述のS811～S814の処理を実行するためのソースプログラムの一例を示す図であり、図153Bは、メダル払出枚数チェック処理中の後述のS816及びS817の処理を実行するためのソースプログラムの一例を示す図である。

30

【1071】

まず、メインCPU101は、メダルOUTカウンタの値に「1」を加算(+1更新)する(S811)。なお、メダルOUTカウンタは、メダルの払出回数を計数するためのカウンタである。次いで、メインCPU101は、払出枚数カウンタの値に「1」を加算(+1更新)する(S812)。なお、払出枚数カウンタは、メダルの払出枚数を計数するためのカウンタである。

【1072】

次いで、メインCPU101は、払出枚数7SEG表示処理を行う(S813)。この処理では、メインCPU101は、払出枚数カウンタの値を、情報表示器6に含まれる払出枚数表示用の2桁の7セグLED(不図示)により表示させる制御処理を行う。

40

【1073】

次いで、メインCPU101は、役連終了枚数カウンタの更新処理を行う(S814)。なお、役連終了枚数カウンタは、入賞役に対応するメダルの払出枚数の残り枚数を計数するためのカウンタである。この処理では、メインCPU101は、役連終了枚数カウンタの値とその下限値「0」とを比較し、役連終了枚数カウンタの値が下限値「0」より大きい場合には、役連終了枚数カウンタの値を1減算(-1更新)し、役連終了枚数カウンタの値が下限値「0」以下である場合には、役連終了枚数カウンタの値を「0」に保持する。

【1074】

次いで、メインCPU101は、入賞枚数カウンタの値を1減算(-1更新)する(S815)。

50

【1075】

次いで、メインCPU101は、クレジット情報コマンド生成処理を行う(S816)。この処理では、メインCPU101は、副制御回路200に送信するクレジット情報コマンドに含まれる、種別データ及び各種通信パラメータを生成する。なお、クレジット情報コマンドは、メダルのクレジット枚数を特定するパラメータを含んで構成される。

【1076】

次いで、メインCPU101は、図72で説明した通信データ格納処理を行う(S817)。この処理により、クレジット情報コマンドデータがメインRAM103に設けられた通信データ格納領域(図75B参照)に保存される。なお、クレジット情報コマンドは、後述の図158で説明する割込処理内の通信データ送信処理により、主制御回路90から副制御回路200に送信される。そして、S817の処理後、メインCPU101は、メダル払出枚数チェック処理を終了し、処理を入賞チェック・メダル払出処理(図150参照)中のS807の処理に移す。

【1077】

本実施形態では、上述のようにしてメダル払出枚数チェック処理が行われる。なお、上述したメダル払出枚数チェック処理中のS811～S814の処理は、メインCPU101が、図153Aのソースプログラムで規定されている各ソースコードを順次実行することにより行われる。その中で、S814の役連終了枚数カウンタの更新処理は、図153A中の「DCPLD」命令(所定の更新命令)により実行される。なお、「DCPLD」命令は、メインCPU101専用命令コードである。

【1078】

ソースプログラム上において、例えば、ソースコード「DCPLD (HL), n」が実行されると、HLレジスタで指定されたアドレスのメモリの内容(格納データ)と整数nとが比較され、メモリの内容が整数nより大きい場合には、メモリの内容が1減算され、メモリの内容が整数n以下である場合には、HLレジスタで指定されたアドレスのメモリに整数nが格納される。それゆえ、図153A中のソースコード「DCPLD (HL), 0」が実行されると、HLレジスタで指定されたアドレスのメモリの内容(役連終了枚数カウンタの値)と整数0(下限値)とが比較され、メモリの内容(役連終了枚数カウンタの値)が整数0より大きい場合には、メモリの内容が1減算され、メモリの内容が整数0以下である場合には、メモリの内容(役連終了枚数カウンタの値)に「0」がセットされる。すなわち、現時点の役連終了枚数カウンタの値が「0」より大きい場合には、役連終了枚数カウンタの更新処理が行われ、現時点の役連終了枚数カウンタの値が「0」以下であれば、役連終了枚数カウンタの値を「0」に保持する処理が行われる。

【1079】

上述のように、メダル払出枚数チェック処理中のS814の処理では、一つの「DCPLD」命令(枚数管理カウンタの下限判定命令と、判断分岐命令が一体になっている命令)により、役連終了枚数カウンタの更新(減算)処理及び連終了枚数カウンタの値を「0」に保持する処理の両方を実行することができる。この場合、両処理を別個に実行するための命令コードを設ける必要がなくなる。例えば、連終了枚数カウンタの値が「0」であるか否かを判別するための判断分岐命令コードを省略することができる。それゆえ、本実施形態のメダル払出枚数チェック処理では、ソースプログラムの容量(メインROM102の使用容量)を低減することができ、メインROM102において、空き容量を確保する(増大させる)ことができ、増えた空き容量を活用して、遊技性を高めることが可能になる。

【1080】

また、上述したメダル払出枚数チェック処理中のS816及びS817の処理は、メインCPU101が、図153Bのソースプログラムで規定されている各ソースコードを順次実行することにより行われる。

【1081】

その中で、S816の処理では、図153Bに示すように、クレジット情報コマンドの通

10

20

30

40

50

信パラメータ 1 には L レジスタを介して払出枚数カウンタの値がセットされ、通信パラメータ 5 には C レジスタを介してクレジットカウンタの値がセットされる。しかしながら、クレジット情報コマンドを構成するその他の通信パラメータ 2 ~ 4 には、現時点において H レジスタ、E レジスタ及び D レジスタにそれぞれ格納されている値（不定値）がセットされる。それゆえ、クレジット情報コマンド送信時における通信パラメータ 2 ~ 4 の値は不定値となる。その結果、本実施形態では、クレジット情報コマンドのサム値（B C C）を送信毎に不定値にすることができ、ゴト等の不正行為を抑制することができる。

【1082】

[B B チェック処理]

次に、図 154 を参照して、メインフロー（図 82 参照）中の S 217 で行う B B チェック処理について説明する。なお、図 154 は、B B チェック処理の手順を示すフローチャートである。

10

【1083】

まず、メイン CPU 101 は、現在の遊技状態がボーナス状態であるか否かを判別する（S 821）。S 821 において、メイン CPU 101 が、現在の遊技状態がボーナス状態でないと判別したとき（S 821 が NO 判定の場合）、メイン CPU 101 は、後述の S 832 の処理を行う。

【1084】

一方、S 821 において、メイン CPU 101 が、現在の遊技状態がボーナス状態であると判別したとき（S 821 が YES 判定の場合）、メイン CPU 101 は、ボーナス状態中に払い出し可能なメダルの枚数を計数するための B B 中払出枚数カウンタの値から、入賞チェック・メダル払出処理において払い出されたメダルの払出枚数を減算する（S 822）。

20

【1085】

次いで、メイン CPU 101 は、B B 中払出枚数カウンタの値が「0」未満であるか否かを判別する（S 823）。S 823 において、メイン CPU 101 が、B B 中払出枚数カウンタの値が「0」未満でないと判別したとき（S 823 が NO 判定の場合）、メイン CPU 101 は、B B チェック処理を終了し、処理をメインフロー（図 82 参照）中の S 218 の処理に移す。

【1086】

一方、S 823 において、メイン CPU 101 が、B B 中払出枚数カウンタの値が「0」未満であると判別したとき（S 823 が YES 判定の場合）、メイン CPU 101 は、ボーナス終了時処理を行う（S 824）。この処理では、メイン CPU 101 は、ボーナス状態中の各種情報をクリアするとともに、RT 1 状態フラグをオン状態にセットする。

30

【1087】

次いで、メイン CPU 101 は、ボーナス終了時 C T 抽籤テーブル（図 60 参照）を参照して、ボーナス終了時の C T 抽籤を行う（S 825）。次いで、メイン CPU 101 は、ボーナス終了時の C T 抽籤に当籤したか否かを判別する（S 826）。

【1088】

S 826 において、メイン CPU 101 が、C T 抽籤に当籤しなかったと判別したとき（S 826 が NO 判定の場合）、メイン CPU 101 は、後述の S 828 の処理を行う。一方、S 826 において、メイン CPU 101 が、C T 抽籤に当籤したと判別したとき（S 826 が YES 判定の場合）、メイン CPU 101 は、C T セット数に「1」を加算する（S 827）。なお、ART セット数が「0」であるときに C T 抽籤に当籤した場合には、S 827 の処理において、C T セット数に「1」を加算するとともに、ART セット数にも「1」を加算する。

40

【1089】

S 827 の処理後又は S 826 が NO 判定の場合、メイン CPU 101 は、ART セット数又は C T セット数が「1」以上であるか否かを判別する（S 828）。

【1090】

50

S 8 2 8 おいて、メイン CPU 1 0 1 が、ART セット数又は CT セット数が「1」以上であると判別したとき (S 8 2 8 が YES 判定の場合)、メイン CPU 1 0 1 は、次遊技の遊技状態に ART 準備状態をセットする (S 8 2 9)。そして、S 8 2 9 の処理後メイン CPU 1 0 1 は、BB チェック処理を終了し、処理をメインフロー (図 8 2 参照) 中の S 2 1 8 の処理に移す。

【1 0 9 1】

一方、S 8 2 8 において、メイン CPU 1 0 1 が、ART セット数又は CT セット数が「1」以上でないと判別したとき (S 8 2 8 が NO 判定の場合)、メイン CPU 1 0 1 は、次遊技の遊技状態に通常遊技状態をセットする (S 8 3 0)。次いで、メイン CPU 1 0 1 は、通常中高確率抽籤テーブル (図 4 0 B 参照) を参照して、C Z の抽籤状態を抽籤し、抽籤結果をセットする (S 8 3 1)。そして、S 8 3 1 の処理後、メイン CPU 1 0 1 は、BB チェック処理を終了し、処理をメインフロー (図 8 2 参照) 中の S 2 1 8 の処理に移す。

10

【1 0 9 2】

ここで再度、S 8 2 1 の処理に戻って、S 8 2 1 が NO 判定の場合、メイン CPU 1 0 1 は、BB 役に係る図柄組合せ (コンビネーション「C__BB1」又は「C__BB2」の図柄組合せ) が表示されたか否かを判定する (S 8 3 2)。S 8 3 2 において、メイン CPU 1 0 1 が、BB 役に係る図柄組合せが表示されなかったと判別したとき (S 8 3 2 が NO 判定の場合)、メイン CPU 1 0 1 は、BB チェック処理を終了し、処理をメインフロー (図 8 2 参照) 中の S 2 1 8 の処理に移す。

20

【1 0 9 3】

一方、S 8 3 2 において、メイン CPU 1 0 1 が、BB 役に係る図柄組合せが表示されたと判別したとき (S 8 3 2 が YES 判定の場合)、メイン CPU 1 0 1 は、ボーナス種別抽籤テーブル (図 5 8 参照) を参照して、ボーナス種別を抽籤し、抽籤結果をセットする (S 8 3 3)。次いで、メイン CPU 1 0 1 は、BB 中払出枚数カウンタの値に所定値 (ボーナス終了契機となる払出枚数：本実施形態では、「2 1 6」) をセットする (S 8 3 4)。

【1 0 9 4】

次いで、メイン CPU 1 0 1 は、ボーナス開始時処理を行う (S 8 3 5)。この処理では、メイン CPU 1 0 1 は、例えば、次遊技の遊技状態にボーナス状態をセットするなどのボーナスの作動開始に必要な各種処理を行う。そして、S 8 3 5 の処理後、メイン CPU 1 0 1 は、BB チェック処理を終了し、処理をメインフロー (図 8 2 参照) 中の S 2 1 8 の処理に移す。

30

【1 0 9 5】

[RT チェック処理]

次に、図 1 5 5 及び図 1 5 6 を参照して、メインフロー (図 8 2 参照) 中の S 2 1 8 で行う RT チェック処理について説明する。なお、図 1 5 5 及び図 1 5 6 は、RT チェック処理の手順を示すフローチャートである。

【1 0 9 6】

まず、メイン CPU 1 0 1 は、RT 状態が RT 5 状態であるか否かを判別する (S 8 4 1)。S 8 4 1 において、メイン CPU 1 0 1 が、RT 状態が RT 5 状態であると判別したとき (S 8 4 1 が YES 判定の場合)、メイン CPU 1 0 1 は、RT チェック処理を終了し、処理をメインフロー (図 8 2 参照) 中の S 2 1 9 の処理に移す。

40

【1 0 9 7】

一方、S 8 4 1 において、メイン CPU 1 0 1 が、RT 状態が RT 5 状態でないと判別したとき (S 8 4 1 が NO 判定の場合)、メイン CPU 1 0 1 は、RT 状態が RT 0 状態であるか否かを判別する (S 8 4 2)。S 8 4 2 において、メイン CPU 1 0 1 が、RT 状態が RT 0 状態でないと判別したとき (S 8 4 2 が NO 判定の場合)、メイン CPU 1 0 1 は、後述の S 8 4 5 の処理を行う。

【1 0 9 8】

50

一方、S 8 4 2において、メインCPU 1 0 1が、RT状態がRT 0状態であると判別したとき（S 8 4 2がYES判定の場合）、メインCPU 1 0 1は、略称「ベルこぼし目」の図柄組合せ（図 2 8 参照）が表示されたか否かを判別する（S 8 4 3）。S 8 4 3において、メインCPU 1 0 1が、略称「ベルこぼし目」の図柄組合せが表示されなかったと判別したとき（S 8 4 3がNO判定の場合）、メインCPU 1 0 1は、RTチェック処理を終了し、処理をメインフロー（図 8 2 参照）中のS 2 1 9の処理に移す。

【 1 0 9 9 】

一方、S 8 4 3において、メインCPU 1 0 1が、略称「ベルこぼし目」の図柄組合せが表示されたと判別したとき（S 8 4 3がYES判定の場合）、メインCPU 1 0 1は、RT 2状態フラグをオン状態にセットする（S 8 4 4）。この処理により、RT状態がRT 0状態からRT 2状態に移行する。そして、S 8 4 4の処理後、メインCPU 1 0 1は、RTチェック処理を終了し、処理をメインフロー（図 8 2 参照）中のS 2 1 9の処理に移す。

10

【 1 1 0 0 】

ここで再度、S 8 4 2の処理に戻って、S 8 4 2がNO判定の場合、メインCPU 1 0 1は、RT状態がRT 1状態であるか否かを判別する（S 8 4 5）。S 8 4 5において、メインCPU 1 0 1が、RT状態がRT 1状態でないと判別したとき（S 8 4 5がNO判定の場合）、メインCPU 1 0 1は、後述のS 8 5 0の処理を行う。

【 1 1 0 1 】

一方、S 8 4 5において、メインCPU 1 0 1が、RT状態がRT 1状態であると判別したとき（S 8 4 5がYES判定の場合）、メインCPU 1 0 1は、略称「ベルこぼし目」の図柄組合せが表示されたか否かを判別する（S 8 4 6）。

20

【 1 1 0 2 】

S 8 4 6において、メインCPU 1 0 1が、略称「ベルこぼし目」の図柄組合せが表示されたと判別したとき（S 8 4 6がYES判定の場合）、メインCPU 1 0 1は、RT 1状態フラグをオフ状態にセットするとともに、RT 2状態フラグをオン状態にセットする（S 8 4 7）。この処理により、RT状態がRT 1状態からRT 2状態に移行する。そして、S 8 4 7の処理後、メインCPU 1 0 1は、RTチェック処理を終了し、処理をメインフロー（図 8 2 参照）中のS 2 1 9の処理に移す。

【 1 1 0 3 】

一方、S 8 4 6において、メインCPU 1 0 1が、略称「ベルこぼし目」の図柄組合せが表示されなかったと判別したとき（S 8 4 6がNO判定の場合）、メインCPU 1 0 1は、RT 1状態の遊技が2 0ゲーム経過したか否かを判別する（S 8 4 8）。

30

【 1 1 0 4 】

S 8 4 8において、メインCPU 1 0 1が、RT 1状態の遊技が2 0ゲーム経過していないと判別したとき（S 8 4 8がNO判定の場合）、メインCPU 1 0 1は、RTチェック処理を終了し、処理をメインフロー（図 8 2 参照）中のS 2 1 9の処理に移す。一方、S 8 4 8において、メインCPU 1 0 1が、RT 1状態の遊技が2 0ゲーム経過したと判別したとき（S 8 4 8がYES判定の場合）、メインCPU 1 0 1は、RT 1状態フラグをオフ状態にセットする（S 8 4 9）。この処理により、RT状態がRT 1状態からRT 0状態に移行する。そして、S 8 4 9の処理後、メインCPU 1 0 1は、RTチェック処理を終了し、処理をメインフロー（図 8 2 参照）中のS 2 1 9の処理に移す。

40

【 1 1 0 5 】

ここで再度、S 8 4 5の処理に戻って、S 8 4 5がNO判定の場合、メインCPU 1 0 1は、RT状態がRT 2状態であるか否かを判別する（S 8 5 0）。S 8 5 0において、メインCPU 1 0 1が、RT状態がRT 2状態でないと判別したとき（S 8 5 0がNO判定の場合）、メインCPU 1 0 1は、後述のS 8 5 3の処理を行う。

【 1 1 0 6 】

一方、S 8 5 0において、メインCPU 1 0 1が、RT状態がRT 2状態であると判別したとき（S 8 5 0がYES判定の場合）、メインCPU 1 0 1は、略称「RT 3移行リブ

50

」の図柄組合せ（図 28 参照）が表示されたか否かを判別する（S 8 5 1）。S 8 5 1 において、メイン CPU 1 0 1 が、略称「RT3 移行リブ」の図柄組合せが表示されなかったと判別したとき（S 8 5 1 が NO 判定の場合）、メイン CPU 1 0 1 は、RT チェック処理を終了し、処理をメインフロー（図 8 2 参照）中の S 2 1 9 の処理に移す。

【1 1 0 7】

一方、S 8 5 1 において、メイン CPU 1 0 1 が、略称「RT3 移行リブ」の図柄組合せが表示されたと判別したとき（S 8 5 1 が YES 判定の場合）、メイン CPU 1 0 1 は、RT 2 状態フラグをオフ状態にセットするとともに、RT 3 状態フラグをオン状態にセットする（S 8 5 2）。この処理により、RT 状態が RT 2 状態から RT 3 状態に移行する。そして、S 8 5 2 の処理後、メイン CPU 1 0 1 は、RT チェック処理を終了し、処理をメインフロー（図 8 2 参照）中の S 2 1 9 の処理に移す。

10

【1 1 0 8】

ここで再度、S 8 5 0 の処理に戻って、S 8 5 0 が NO 判定の場合、メイン CPU 1 0 1 は、RT 状態が RT 3 状態であるか否かを判別する（S 8 5 3）。S 8 5 3 において、メイン CPU 1 0 1 が、RT 状態が RT 3 状態でないと判別したとき（S 8 5 3 が NO 判定の場合）、メイン CPU 1 0 1 は、後述の S 8 6 2 の処理を行う。

【1 1 0 9】

一方、S 8 5 3 において、メイン CPU 1 0 1 が、RT 状態が RT 3 状態であると判別したとき（S 8 5 3 が YES 判定の場合）、メイン CPU 1 0 1 は、略称「ベルこぼし目」又は「RT 2 移行リブ」の図柄組合せが表示されたか否かを判別する（S 8 5 4）。

20

【1 1 1 0】

S 8 5 4 において、メイン CPU 1 0 1 が、略称「ベルこぼし目」又は「RT 2 移行リブ」の図柄組合せが表示されたと判別したとき（S 8 5 4 が YES 判定の場合）、メイン CPU 1 0 1 は、RT 3 状態フラグをオフ状態にセットするとともに、RT 2 状態フラグをオン状態にセットする（S 8 5 5）。この処理により、RT 状態が RT 3 状態から RT 2 状態に移行する。そして、S 8 5 5 の処理後、メイン CPU 1 0 1 は、RT チェック処理を終了し、処理をメインフロー（図 8 2 参照）中の S 2 1 9 の処理に移す。

【1 1 1 1】

一方、S 8 5 4 において、メイン CPU 1 0 1 が、略称「ベルこぼし目」又は「RT 2 移行リブ」の図柄組合せが表示されていないと判別したとき（S 8 5 4 が NO 判定の場合）、メイン CPU 1 0 1 は、略称「RT 4 移行リブ」の図柄組合せ（図 28 参照）が表示されたか否かを判別する（S 8 5 6）。

30

【1 1 1 2】

S 8 5 6 において、メイン CPU 1 0 1 が、略称「RT 4 移行リブ」の図柄組合せが表示されていないと判別したとき（S 8 5 6 が NO 判定の場合）、メイン CPU 1 0 1 は、RT チェック処理を終了し、処理をメインフロー（図 8 2 参照）中の S 2 1 9 の処理に移す。一方、S 8 5 6 において、メイン CPU 1 0 1 が、略称「RT 4 移行リブ」の図柄組合せが表示されたと判別したとき（S 8 5 6 が YES 判定の場合）、メイン CPU 1 0 1 は、RT 3 状態フラグをオフ状態にセットするとともに、RT 4 状態フラグをオン状態にセットする（S 8 5 7）。この処理により、RT 状態が RT 3 状態から RT 4 状態に移行する。

40

【1 1 1 3】

S 8 5 7 の処理後、メイン CPU 1 0 1 は、遊技状態が ART 準備状態であるか否かを判別する（S 8 5 8）。S 8 5 8 において、メイン CPU 1 0 1 が、遊技状態が ART 準備状態でないと判別したとき（S 8 5 8 が NO 判定の場合）、メイン CPU 1 0 1 は、RT チェック処理を終了し、処理をメインフロー（図 8 2 参照）中の S 2 1 9 の処理に移す。

【1 1 1 4】

一方、S 8 5 8 において、メイン CPU 1 0 1 が、遊技状態が ART 準備状態であると判別したとき（S 8 5 8 が YES 判定の場合）、メイン CPU 1 0 1 は、CT セット数が「1」以上であるか否かを判別する（S 8 5 9）。

50

【 1 1 1 5 】

S 8 5 9において、メインCPU 1 0 1が、CTセット数が「1」以上であると判別したとき（S 8 5 9がYES判定の場合）、メインCPU 1 0 1は、次遊技の遊技状態にCTをセットし、CTゲーム数カウンタに「8」をセットする（S 8 6 0）。そして、S 8 6 0の処理後、メインCPU 1 0 1は、RTチェック処理を終了し、処理をメインフロー（図8 2参照）中のS 2 1 9の処理に移す。

【 1 1 1 6 】

一方、S 8 5 9において、メインCPU 1 0 1が、CTセット数が「1」以上でないと判別したとき（S 8 5 9がNO判定の場合）、メインCPU 1 0 1は、次遊技の遊技状態に通常ARTをセットし、ART終了ゲーム数カウンタに所定値をセットする（S 8 6 1）。そして、S 8 6 1の処理後、メインCPU 1 0 1は、RTチェック処理を終了し、処理をメインフロー（図8 2参照）中のS 2 1 9の処理に移す。

【 1 1 1 7 】

ここで再度、S 8 5 3の処理に戻って、S 8 5 3がNO判定の場合、メインCPU 1 0 1は、略称「ベルこぼし目」又は「RT 2 移行リップ」の図柄組合せが表示されたか否かを判別する（S 8 6 2）。S 8 6 2において、メインCPU 1 0 1が、略称「ベルこぼし目」又は「RT 2 移行リップ」の図柄組合せが表示されていないと判別したとき（S 8 6 2がNO判定の場合）、メインCPU 1 0 1は、RTチェック処理を終了し、処理をメインフロー（図8 2参照）中のS 2 1 9の処理に移す。

【 1 1 1 8 】

一方、S 8 6 2において、メインCPU 1 0 1が、略称「ベルこぼし目」又は「RT 2 移行リップ」の図柄組合せが表示されたと判別したとき（S 8 6 2がYES判定の場合）、メインCPU 1 0 1は、RT 4 状態フラグをオフ状態にセットするとともに、RT 2 状態フラグをオン状態にセットする（S 8 6 3）。この処理により、RT 状態がRT 4 状態からRT 2 状態に移行する。そして、S 8 6 3の処理後、メインCPU 1 0 1は、RTチェック処理を終了し、処理をメインフロー（図8 2参照）中のS 2 1 9の処理に移す。

【 1 1 1 9 】

[CZ・ART終了時処理]

次に、図1 5 7を参照して、メインフロー（図8 2参照）中のS 2 1 9で行うCZ・ART終了時処理について説明する。なお、図1 5 7は、CZ・ART終了時処理の手順を示すフローチャートである。

【 1 1 2 0 】

まず、メインCPU 1 0 1は、現在の遊技状態がCZ失敗時及びART終了時のいずれかであるか否かを判別する（S 8 7 1）。S 8 7 1において、メインCPU 1 0 1が、現在の遊技状態がCZ失敗時及びART終了時のいずれかでないと判別したとき（S 8 7 1がNO判定の場合）、メインCPU 1 0 1は、CZ・ART終了時処理を終了し、処理をメインフロー（図8 2参照）中のS 2 0 1の処理に移す。

【 1 1 2 1 】

一方、S 8 7 1において、メインCPU 1 0 1が、現在の遊技状態がCZ失敗時及びART終了時のいずれかであると判別したとき（S 8 7 1がYES判定の場合）、メインCPU 1 0 1は、CZ抽籤テーブル（図4 1 B参照）を参照して、CZの引き戻し抽籤を行う（S 8 7 2）。次いで、メインCPU 1 0 1は、CZの引き戻し抽籤に当籤したか否かを判別す（S 8 7 3）。

【 1 1 2 2 】

S 8 7 3において、メインCPU 1 0 1が、CZの引き戻し抽籤に当籤したと判別したとき（S 8 7 3がYES判定の場合）、メインCPU 1 0 1は、次遊技の遊技状態に当籤した種別のCZをセットする（S 8 7 4）。次いで、メインCPU 1 0 1は、当籤した種別のCZに応じた値をCZゲーム数カウンタにセットする（S 8 7 5）。そして、S 8 7 5の処理後、メインCPU 1 0 1は、CZ・ART終了時処理を終了し、処理をメインフロー（図8 2参照）中のS 2 0 1の処理に移す。

10

20

30

40

50

【 1 1 2 3 】

一方、S 8 7 3において、メインCPU 1 0 1が、C Zの引き戻し抽籤に当籤しなかったと判別したとき（S 8 7 3がNO判定の場合）、メインCPU 1 0 1は、次遊技の遊技状態に通常遊技状態をセットする（S 8 7 6）。次いで、メインCPU 1 0 1は、通常中高確率抽籤テーブル（図 4 0 B 参照）を参照して、C Zの抽籤状態を抽籤し、抽籤結果をセットする（S 8 7 7）。そして、S 8 7 7の処理後、メインCPU 1 0 1は、C Z・ART終了時処理を終了し、処理をメインフロー（図 8 2 参照）中のS 2 0 1の処理に移す。

【 1 1 2 4 】

[メインCPUの制御による割込処理（1.1172 msec）]

次に、図 1 5 8を参照して、1.1172 msec周期で、メインCPU 1 0 1が行う割込処理について説明する。なお、図 1 5 8は、割込処理の手順を示すフローチャートである。1.1172 msec周期で繰り返し実行される割込処理は、タイマー回路 1 1 3（PTC）の初期化処理（図 6 4 中のS 2 参照）で設定されたタイマー回路 1 1 3のタイムアウト信号の出力タイミングに基づいて発生する割込みコントローラ 1 1 2からの割込要求信号がメインCPU 1 0 1に輸入された際に実行される処理である。

【 1 1 2 5 】

まず、メインCPU 1 0 1は、レジスタの退避処理を行う（S 9 0 1）。次いで、メインCPU 1 0 1は、入力ポートチェック処理を行う（S 9 0 2）。この処理では、ストップスイッチ等の各種スイッチから入力される信号がチェックされる。

【 1 1 2 6 】

次いで、メインCPU 1 0 1は、リール制御処理を行う（S 9 0 3）。この処理では、メインCPU 1 0 1は、全リールの回転開始が要求されたときに、左リール 3 L、中リール 3 C及び右リール 3 Rの回転を開始し、その後、各リールが一定速度で回転するように、3つのステップモータを駆動制御する。また、滑り駒数が決定されたときは、メインCPU 1 0 1は、該当するリールの図柄カウンタを滑り駒数分だけ更新する。そして、メインCPU 1 0 1は、更新された図柄カウンタが停止予定位置に対応する値に一致する（停止予定位置の図柄が表示窓の有効ライン上の領域に到達する）のを待って、該当するリールの回転の減速及び停止が行われるように、対応するステップモータを駆動制御する。

【 1 1 2 7 】

次いで、メインCPU 1 0 1は、通信データ送信処理を行う（S 9 0 4）。この処理では、主に、通信データ格納領域に格納された各種コマンドを主制御回路 9 0の第1シリアル通信回路 1 1 4（図 9 参照）を介して副制御回路 2 0 0に送信する。メインCPU 1 0 1は、副制御回路 2 0 0にコマンドを送信した後、通信データポインタを1パケット分減算更新し（不図示）、通信データ格納領域の送信済みのコマンドデータをクリアする。なお、通信データ格納領域に複数のコマンドデータが格納されている場合には、格納された古い順で、コマンドデータを副制御回路 2 0 0に送信する。また、通信データ格納領域にコマンドデータが格納されていない場合、すなわち、通信データポインタの値が「0」である場合には、無操作コマンドを生成して副制御回路 2 0 0に送信する。次いで、メインCPU 1 0 1は、投入メダル通過チェック処理を行う（S 9 0 5）。この処理では、メインCPU 1 0 1は、メダルセンサ（不図示）の検出結果（メダルセンサ入力状態）に基づいて、投入メダルがセクタ 6 6を通過したか否かのチェック処理を行う。次いで、メインCPU 1 0 1は、WDTのリスタート処理を行う（S 9 0 6）。

【 1 1 2 8 】

次いで、メインCPU 1 0 1は、7セグLED駆動処理を行う（S 9 0 7）。この処理では、メインCPU 1 0 1は、情報表示器 6に含まれる各種7セグLEDを駆動制御して、例えば、メダルの払出枚数やクレジット枚数、ストップボタンの押し順データなどを表示する。なお、7セグLED駆動処理の詳細については、後述の図 1 5 9を参照しながら後で説明する。

【 1 1 2 9 】

10

20

30

40

50

次いで、メインCPU101は、タイマー更新処理を行う(S908)。この処理では、メインCPU101は、セットされた各種タイマーのカウント(減算)処理を行う。なお、タイマー更新処理の詳細については、後述の図164を参照しながら後で説明する。

【1130】

次いで、メインCPU101は、エラー検知処理を行う(S909)。次いで、メインCPU101は、ドア開閉チェック処理を行う(S910)。ドア開閉チェック処理では、メインCPU101は、ドア開閉監視スイッチ67のオン(ドア閉)/オフ(ドア開)状態をチェックすることにより、フロントドア2b(図2参照)の開閉状態をチェックする。

【1131】

次いで、メインCPU101は、試射試験信号制御処理を行う(S911)。この処理では、第2インターフェースポート等を介して試験機に各種試験信号の出力する際の制御処理が行われる。また、この処理は、メインRAM103の規定外作業領域(図12C参照)を用いて実行される。なお、本実施形態では、この処理は、試射試験時以外のとき(パチスロ1が遊技店に設置された後)にも行われるが、この時には、主制御基板71が第2インターフェースポート等を介して試験機に接続されていないので、各種試験信号は生成されても出力はされない。試射試験信号制御処理の詳細については、後述の図166を参照しながら後で説明する。

10

【1132】

次いで、メインCPU101は、レジスタの復帰処理を行う(S912)。そして、S912の処理後、メインCPU101は、割込処理を終了する。

20

【1133】

[7セグLED駆動処理]

次に、図159及び図160を参照して、割込処理(図158参照)中のS907で行う7セグLED駆動処理について説明する。なお、図159は、7セグLED駆動処理の手順を示すフローチャートである。また、図160Aは、7セグLED駆動処理中の後述のS923~S925の処理を実行するためのソースプログラムの一例を示す図であり、図160Bは、7セグLED駆動処理中の後述のS931~S936の処理を実行するためのソースプログラムの一例を示す図である。

【1134】

まず、メインCPU101は、割込カウンタの値に「1」を加算(+1更新)する(S921)。次いで、メインCPU101は、割込カウンタの値が奇数であるか否かを判別する(S922)。

30

【1135】

S922において、メインCPU101が、割込カウンタの値が奇数でないと判別したとき(S922がNO判定の場合)、メインCPU101は、7セグLED駆動処理を終了し、処理を割込処理(図158参照)中のS908の処理に移す。すなわち、本実施形態では、2回の割込周期毎に、7セグLED駆動処理が行われる。なお、本実施形態では、7セグLED駆動処理を割込みカウンタの値が偶数の場合に実行する例を説明したが、本発明はこれに限定されず、割込みカウンタの値が奇数の場合に7セグLED駆動処理を実行してもよいし、また、任意の整数で割込みカウンタの値を除算したときの商又は余りを用いて、7セグLED駆動処理の実行タイミングを決定してもよい。

40

【1136】

一方、S922において、メインCPU101が、割込カウンタの値が奇数であると判別したとき(S922がYES判定の場合)、メインCPU101は、ナビデータ格納領域からナビデータを取得する(S923)。次いで、メインCPU101は、7セグLEDの各カソードに出力される押し順表示データを格納するための押し順表示データ格納領域のアドレスをセットする(S924)。

【1137】

次いで、メインCPU101は、7セグ表示データ生成処理を行う(S925)。この処理では、メインCPU101は、ナビデータに基づいて、押し順表示データ(7セグ表示

50

データ)を作成し、生成された押し順表示データを押し順表示データ格納領域に格納する。なお、7セグ表示データ生成処理の詳細については、後述の図161を参照しながら後で説明する。

【1138】

次いで、メインCPU101は、クレジットカウンタの値を取得する(S926)。次いで、メインCPU101は、7セグLEDの各カソードに出力されるクレジット表示データを格納するためのクレジット表示データ格納領域のアドレスをセットする(S927)。

【1139】

次いで、メインCPU101は、7セグ表示データ生成処理を行う(S928)。この処理では、メインCPU101は、クレジットカウンタの値に基づいて、クレジット表示データ(7セグ表示データ)を生成し、生成されたクレジット表示データをクレジット表示データ格納領域に格納する。なお、7セグ表示データ生成処理の詳細については、後述の図161を参照しながら後で説明する。

10

【1140】

次いで、メインCPU101は、後述の7セグコモンカウンタの値を格納するための7セグコモンカウンタ格納領域のアドレスをセットする(S929)。次いで、メインCPU101は、7セグコモンカウンタの値に「1」を加算(+1更新)する(S930)。なお、この処理において、更新後の7セグコモンカウンタの値が「8」となった場合には、メインCPU101は、7セグコモンカウンタの値に「0」をセットする。本実施形態では、7セグLEDをダイナミック制御するため、8回周期で7セグコモンカウンタの値が更新される。

20

【1141】

次いで、メインCPU101は、7セグコモンカウンタの値に基づいて、コモン選択データを作成し、対象のカソードデータ格納領域(押し順表示データ格納領域又はクレジット表示データ格納領域内の対象格納領域)のアドレスをセットする(S931)。次いで、メインCPU101は、7セグLEDのカソードにクリアデータを出力する(S932)。この処理は、7セグLEDを一旦消灯して、残像の影響を無くすために行われる。

【1142】

次いで、メインCPU101は、対象のカソードデータ格納領域から7セグカソード出力データを取得してセットする(S933)。次いで、メインCPU101は、7セグコモンバックアップデータとコモン選択データとから、7セグコモン出力データを生成する(S934)。

30

【1143】

次いで、メインCPU101は、7セグコモンバックアップデータ及び7セグカソードバックアップデータにそれぞれ7セグコモン出力データ及び7セグカソード出力データを保存する(S935)。次いで、メインCPU101は、7セグカソード出力データ及び7セグコモン出力データを出力する(S936)。そして、S936の処理後、メインCPU101は、7セグLED駆動処理を終了し、処理を割込処理(図158参照)中のS908の処理に移す。

【1144】

本実施形態では、上述のようにして7セグLED駆動処理が行われる。なお、上述した7セグLED駆動処理中のS923~S925の処理は、メインCPU101が、図160Aのソースプログラムで規定されている各ソースコードを順次実行することにより行われる。また、上述した7セグLED駆動処理中のS931~S936の処理は、メインCPU101が、図160Bのソースプログラムで規定されている各ソースコードを順次実行することにより行われる。

40

【1145】

その中で、S936の各7セグ出力データの出力処理は、図160Bに示すように、一つのソースコード「LD(cPA_SEG COM), BC」により実行される。それゆえ、本実施形態の7セグLED駆動処理では、2桁の7セグLEDをダイナミック点灯制御

50

する際に、7セグコモン出力（選択）データと、7セグカソード出力データとが同時に出力される。すなわち、指示モニタで押し順ナビを実施する際の7セグLEDをダイナミック点灯制御、及び、2桁の7セグLEDでクレジット情報を表示する際の7セグLEDをダイナミック点灯制御では、7セグコモン出力（選択）データと、7セグカソード出力データとが同時に出力される。

【1146】

この場合、ソースプログラム上において、7セグLEDのダイナミック点灯制御に必要な命令コード数を減らすことができる。それゆえ、本実施形態では、ソースプログラムの容量（メインROM102の使用容量）を低減することができ、メインROM102において、空き容量を確保する（増大させる）ことができ、増えた空き容量を活用して、遊技性を高めることが可能になる。また、本実施形態では、7セグLED駆動処理を行う7セグ駆動回路（不図示）をカソードコモン回路で構成し、カソードで制御する例を説明したが、本発明はこれに限定されず、7セグ駆動回路をアノードコモン回路で構成し、アノードで7セグLEDの制御を行ってもよい。

10

【1147】

また、上述した7セグLED駆動処理中のS923のナビデータの取得処理及びS924の押し表示データ格納領域のアドレスセット処理はいずれも、図160Aに示すように、Qレジスタ（拡張レジスタ）を用いてアドレス指定を行う「LDQ」命令（メインCPU101専用命令コード）により実行される。それゆえ、本実施形態の7セグLED駆動処理では、「LDQ」命令を用いることにより、直値により、メインROM102、メインRAM103やメモリーマップI/Oにアクセスすることができるので、ソースプログラム上において、アドレス設定に係る命令を省略することができ（アドレス設定に係る命令を別途設ける必要がなくなる）、その分、ソースプログラムの容量（メインROM102の使用容量）を低減することができる。この結果、本実施形態では、メインROM102において、空き容量を確保する（増大させる）ことができ、増えた空き容量を活用して、遊技性を高めることが可能になる。

20

【1148】

[7セグ表示データ生成処理]

次に、図161～図163を参照して、7セグLED駆動処理（図159参照）中のS925及びS928で行う7セグ表示データ生成処理について説明する。なお、図161は、7セグ表示データ生成処理の手順を示すフローチャートである。図162は、7セグ表示データ生成処理を実行するためのソースプログラムの一例を示す図である。また、図163は、7セグ表示データ生成処理のソースプログラム上で、実際に参照される7セグカソードテーブルの構成の一例を示す図である。

30

【1149】

なお、7セグLED駆動処理（図159参照）中のS925で行われる7セグ表示データ生成処理で生成される後述の「表示データ」は押し順表示データに対応し、7セグLED駆動処理（図159参照）中のS928で行われる7セグ表示データ生成処理で生成される後述の「表示データ」はクレジット表示データに対応する。

【1150】

まず、メインCPU101は、カソードデータ格納領域にセットされた表示データを「10」で除算し、その除算結果の商の値を、2桁の7セグLEDの上位桁の表示データとして取得し、除算結果の余の値を下位桁の表示データとして取得する（S941）。次いで、メインCPU101は、取得した上位桁の表示データに基づいて、上位桁表示を行うか否かを判別する（S942）。

40

【1151】

S942において、メインCPU101が、上位桁表示を行うと判別したとき（S942がYES判定の場合）、メインCPU101は、後述のS944の処理を行う。一方、S942において、メインCPU101が、上位桁表示を行わないと判別したとき（S942がNO判定の場合）、メインCPU101は、上位桁の表示無しをセットする（S94

50

3)。

【1152】

S943の処理後又はS942がYES判定の場合、メインCPU101は、7セグカソードテーブル(図163参照)を参照して、上位桁の表示データを取得する(S944)。次いで、メインCPU101は、上位桁の表示データ格納領域(不図示)に取得した上位桁の表示データを保存する(S945)。

【1153】

次いで、メインCPU101は、7セグカソードテーブル(図163参照)を参照して、下位桁の表示データを取得する(S946)。次いで、メインCPU101は、下位桁の表示データ格納領域(不図示)に取得した下位桁の表示データを保存する(S947)。

10

【1154】

そして、S947の処理後、メインCPU101は、7セグ表示データ生成処理を終了する。この際、実行した7セグ表示データ生成処理が7セグLED駆動処理(図158参照)中のS925の処理である場合には、メインCPU101は、処理を7セグLED駆動処理中のS926の処理に移す。一方、実行した7セグ表示データ生成処理が7セグLED駆動処理(図158参照)中のS928の処理である場合には、メインCPU101は、処理を7セグLED駆動処理中のS929の処理に移す。

【1155】

本実施形態では、上述のようにして7セグ表示データ生成処理が行われる。なお、上述した7セグ表示データ生成処理は、メインCPU101が、図162のソースプログラムで規定されている各ソースコードを順次実行することにより行われる。

20

【1156】

[タイマー更新処理]

次に、図164及び図165を参照して、割込処理(図158参照)中のS908で行うタイマー更新処理について説明する。なお、図164は、タイマー更新処理の手順を示すフローチャートである。また、図165は、タイマー更新処理中の後述のS951~S954の処理を実行するためのソースプログラムの一例を示す図である。

【1157】

まず、メインCPU101は、HLレジスタに2バイトタイマー格納領域(不図示)の更新開始アドレスをセットし、Bレジスタに2バイトタイマー数をセットする(S951)。

30

【1158】

次いで、メインCPU101は、2バイトタイマー値とその下限値「0」とを比較し、2バイトタイマー値が下限値「0」より大きい場合には、2バイトタイマー値を1減算(-1更新)し、2バイトタイマー値が下限値「0」以下である場合には、2バイトタイマー値を「0」に保持する(S952)。さらに、S952の処理では、メインCPU101は、HLレジスタにセットされている2バイトタイマー格納領域の更新開始アドレスを2減算(-2更新)する。

【1159】

次いで、メインCPU101は、Bレジスタにセットされた2バイトタイマー数を1減算(-1更新)する(S953)。次いで、メインCPU101は、Bレジスタにセットされた2バイトタイマー数が「0」であるか否かを判別する(S954)。

40

【1160】

S954において、メインCPU101が、Bレジスタにセットされた2バイトタイマー数が「0」でないと判別したとき(S954がNO判定の場合)、メインCPU101は、処理をS952の処理に戻し、S952以降の処理を繰り返す。

【1161】

一方、S954において、メインCPU101が、Bレジスタにセットされた2バイトタイマー数が「0」であると判別したとき(S954がYES判定の場合)、メインCPU101は、HLレジスタに1バイトタイマー格納領域の更新開始アドレスをセットし、Bレジスタに1バイトタイマー数をセットする(S955)。

50

【 1 1 6 2 】

次いで、メインCPU101は、1バイトタイマー値とその下限値「0」とを比較し、1バイトタイマー値が下限値「0」より大きい場合には、1バイトタイマー値を1減算（-1更新）し、1バイトタイマー値が下限値「0」以下である場合には、1バイトタイマー値を「0」に保持する（S956）。さらに、S956の処理では、メインCPU101は、HLレジスタにセットされている1バイトタイマー格納領域の更新開始アドレスを1減算（-1更新）する。

【 1 1 6 3 】

次いで、メインCPU101は、Bレジスタにセットされた1バイトタイマー数を1減算（-1更新）する（S957）。次いで、メインCPU101は、Bレジスタにセットされた1バイトタイマー数が「0」であるか否かを判別する（S958）。 10

【 1 1 6 4 】

S958において、メインCPU101が、Bレジスタにセットされた1バイトタイマー数が「0」でないと判別したとき（S958がNO判定の場合）、メインCPU101は、処理をS956の処理に戻し、S956以降の処理を繰り返す。

【 1 1 6 5 】

一方、S958において、メインCPU101が、Bレジスタにセットされた1バイトタイマー数が「0」であると判別したとき（S958がYES判定の場合）、メインCPU101は、電磁カウンタ制御処理を行う（S959）。この処理では、メダルのIN/OUTを示す信号を外部集中端子板47に出力する際の出力制御処理が行われる。そして、S959の処理後、メインCPU101は、タイマー更新処理を終了し、処理を割込処理（図158参照）中のS909の処理に移す。 20

【 1 1 6 6 】

本実施形態では、上述のようにしてタイマー更新処理が行われる。なお、上述したタイマー更新処理中のS951～S954の処理（2バイトタイマーの更新処理）は、メインCPU101が、図165のソースプログラムで規定されている各ソースコードを順次実行することにより行われる。

【 1 1 6 7 】

その中で、S952の処理（2バイトタイマーの更新処理）は、図165中の「DCPWL D」命令（所定の更新命令）により実行される。なお、「DCPWL D」命令は、メインCPU101専用命令コードである。 30

【 1 1 6 8 】

ソースプログラム上において、例えば、ソースコード「DCPWL D（HL），n」が実行されると、HLレジスタで指定されたアドレスから2バイト分のメモリの内容（格納データ）と整数nとが比較され、2バイト分のメモリの内容が整数nより大きい場合には、2バイト分のメモリの内容が1減算され、2バイト分のメモリの内容が整数n以下である場合には、HLレジスタで指定されたアドレスから2バイト分のメモリに整数nが格納される。

【 1 1 6 9 】

それゆえ、図165中のソースコード「DCPWL D（HL），0」では、HLレジスタで指定されたアドレスから2バイト分のメモリの内容（2バイトタイマー値）と整数「0」（下限値）とが比較され、2バイト分のメモリの内容が整数「0」より大きい場合には、2バイト分のメモリの内容が1減算され、2バイト分のメモリの内容が整数「0」以下である場合には、2バイト分のメモリの内容に「0」がセットされる。すなわち、現時点の2バイトタイマー値が「0」より大きい場合には、2バイトタイマーの更新処理が行われ、現時点の2バイトタイマー値が「0」以下であれば、2バイトタイマー値が「0」に保持される。 40

【 1 1 7 0 】

上述のように、本実施形態のタイマー更新処理では、メインCPU101専用命令コードである「DCPWL D」命令により、タイマー値の更新（減算）処理及びタイマー値を「 50

0」に保持する処理の両方を実行することができる。この場合、両処理を別個に実行するための命令コードを設ける必要がなくなる。また、タイマー値が「0」であるか否かを判別するための判断分岐命令コードも省略することができる。それゆえ、本実施形態では、ソースプログラムの容量（メインROM102の使用容量）を低減することができ、メインROM102において、空き容量を確保する（増大させる）ことができ、増えた空き容量を活用して、遊技性を高めることが可能になる。なお、本実施形態では、2バイトタイマーの更新処理においてのみ「DCPWD」命令を使用する例を説明したが、本発明はこれに限定されず、1バイトタイマーの更新処理においても「DCPWD」命令を使用してもよい。

【1171】

[試射試験信号制御処理（規定外）]

次に、図166を参照して、割込処理（図158参照）中のS911で行う試射試験信号制御処理について説明する。なお、図166は、試射試験信号制御処理の手順を示すフローチャートである。

【1172】

まず、メインCPU101は、メインRAM103のスタックエリアのアドレスを退避させる（S961）。次いで、メインCPU101は、スタックポインタ（SP）に規定外スタックエリアのアドレスをセットする（S962）。次いで、メインCPU101は、全レジスタのデータを退避させる（S963）。

【1173】

次いで、メインCPU101は、回胴制動信号生成処理を行う（S964）。この処理では、メインCPU101は、第2インターフェースポート等を介して試験機に出力される、各リールの回転制御信号（駆動信号）の生成及び出力処理を行う。なお、回胴制動信号生成処理の詳細については、後述の図167を参照しながら後で説明する。

【1174】

次いで、メインCPU101は、特賞信号制御処理を行う（S965）。この処理では、メインCPU101は、試験機に出力される、ボーナス（特賞）のON/OFF信号（試験信号）の出力処理を行う。なお、特賞信号制御処理の詳細については、後述の図168を参照しながら後で説明する。

【1175】

次いで、メインCPU101は、条件装置信号制御処理を行う（S966）。この処理では、メインCPU101は、条件装置信号制御フラグの状態に対応する制御信号の出力処理を行う。なお、条件装置信号制御処理の詳細については、後述の図169及び図170を参照しながら後で説明する。

【1176】

次いで、メインCPU101は、S963の処理で退避させた全レジスタのデータの復帰処理を行う（S967）。次いで、メインCPU101は、S961の処理で退避させたスタックエリアのアドレスをスタックポインタ（SP）にセットする（S968）。そして、S968の処理後、メインCPU101は、試射試験信号制御処理を終了し、処理を割込処理（図158参照）中のS912の処理に移す。

【1177】

[回胴制動信号生成処理]

次に、図167を参照して、試射試験信号制御処理（図166参照）中のS964で行う回胴制動信号生成処理について説明する。なお、図167は、回胴制動信号生成処理の手順を示すフローチャートである。

【1178】

まず、メインCPU101は、規定外作業領域に回胴制御データ格納領域（不図示）をセットする（S971）。次いで、メインCPU101は、リール数に「3」をセットし、回胴制御信号及びその生成状態（1バイトデータ）をクリアする（S972）。

【1179】

10

20

30

40

50

次いで、メインCPU101は、回胴制御データが「停止中未満」のデータであるか否かを判別する(S973)。なお、ここでいう「停止中未満」の回胴制御データとは、リールを停止するための回胴制御データ以外の回胴制御データ、すなわち、リールを回転駆動するための回胴制御データ(加速準備、加速中、定速待ち、定速中及び停止開始位置待ちのいずれかの状態)のことである。

【1180】

S973において、メインCPU101が、回胴制御データが「停止中未満」のデータであると判別したとき(S973がYES判定の場合)、メインCPU101は、後述のS975の処理を行う。一方、S973において、メインCPU101が、回胴制御データが「停止中未満」のデータでないと判別したとき(S973がNO判定の場合)、メインCPU101は、回胴制御データが「静定ホールド制御終了」のデータであるか否かを判別する(S974)。なお、ここでいう「静定ホールド制御終了」の回胴制御データとは、リールの全相全停止状態を示す回胴制御データのことである。

10

【1181】

S974において、メインCPU101が、回胴制御データが「静定ホールド制御終了」のデータであると判別したとき(S974がYES判定の場合)、メインCPU101は、後述のS976の処理を行う。一方、S974において、メインCPU101が、回胴制御データが「静定ホールド制御終了」のデータでないと判別したとき(S974がNO判定の場合)、又は、S973がYES判定の場合、メインCPU101は、回胴制御信号の生成状態(1バイトデータ)のビット3をオン状態(「1」)にする(S975)。

20

【1182】

S975の処理後又はS974がNO判定の場合、メインCPU101は、生成状態の各ビットのデータを1ビット分、右(ビット7からビット0に向かう方向)にシフトする(S976)。次いで、メインCPU101は、回胴制御データ格納領域のアドレスを次の制御対象のリールのアドレスに更新する(S977)。

【1183】

次いで、メインCPU101は、リール数を1減算する(S978)。次いで、メインCPU101は、リール数が「0」であるか否かを判別する(S979)。

【1184】

S979において、メインCPU101が、リール数が「0」でないと判別したとき(S979がNO判定の場合)、メインCPU101は、処理をS973の処理に戻し、S973以降の処理を繰り返す。

30

【1185】

一方、S979において、メインCPU101が、リール数が「0」であると判別したとき(S979がYES判定の場合)、メインCPU101は、生成状態のデータを回胴制動信号出力ポートを介して試験機用第1インターフェースボード301(図7参照)へ出力する(S980)。そして、S980の処理後、メインCPU101は、回胴制動信号生成処理を終了し、処理を試射試験信号制御処理(図166参照)中のS965の処理に移す。

【1186】

[特賞信号制御処理]

次に、図168を参照して、試射試験信号制御処理(図166参照)中のS965で行う特賞信号制御処理について説明する。なお、図168は、特賞信号制御処理の手順を示すフローチャートである。

40

【1187】

まず、メインCPU101は、遊技状態フラグ格納領域(図32参照)を参照して、遊技状態フラグを取得する(S991)。次いで、メインCPU101は、遊技状態がRB遊技状態であるか否かを判別する(S992)。

【1188】

S992において、メインCPU101が、遊技状態がRB遊技状態であると判別したと

50

き（S 9 9 2 が Y E S 判定の場合）、メイン C P U 1 0 1 は、試験信号用の R B 中信号ポートから O N 信号を試験機用第 1 インターフェースボード 3 0 1（図 7 参照）へ出力する（S 9 9 3）。一方、S 9 9 2 において、メイン C P U 1 0 1 が、遊技状態が R B 遊技状態でないと判別したとき（S 9 9 2 が N O 判定の場合）、メイン C P U 1 0 1 は、試験信号用の R B 中信号ポートから O F F 信号を試験機用第 1 インターフェースボード 3 0 1（図 7 参照）へ出力する（S 9 9 4）。

【 1 1 8 9 】

S 9 9 3 又は S 9 9 4 の処理後、メイン C P U 1 0 1 は、遊技状態フラグ格納領域（図 3 2 参照）を参照して、遊技状態が B B 遊技状態であるか否かを判別する（S 9 9 5）。

【 1 1 9 0 】

S 9 9 5 において、メイン C P U 1 0 1 が、遊技状態が B B 遊技状態であると判別したとき（S 9 9 5 が Y E S 判定の場合）、メイン C P U 1 0 1 は、試験信号用の B B 中信号ポートから O N 信号を試験機用第 1 インターフェースボード 3 0 1（図 7 参照）へ出力する（S 9 9 6）。一方、S 9 9 5 において、メイン C P U 1 0 1 が、遊技状態が B B 遊技状態でないと判別したとき（S 9 9 5 が N O 判定の場合）、メイン C P U 1 0 1 は、試験信号用の B B 中信号ポートから O F F 信号を試験機用第 1 インターフェースボード 3 0 1（図 7 参照）へ出力する（S 9 9 7）。

【 1 1 9 1 】

そして、S 9 9 6 又は S 9 9 7 の処理後、メイン C P U 1 0 1 は、特賞信号制御処理を終了し、処理を試射試験信号制御処理（図 1 6 6 参照）中の S 9 6 6 の処理に移す。

【 1 1 9 2 】

[条件装置信号制御処理]

次に、図 1 6 9 及び図 1 7 0 を参照して、試射試験信号制御処理（図 1 6 6 参照）中の S 9 6 6 で行う条件装置信号制御処理について説明する。なお、図 1 6 9 及び図 1 7 0 は、条件装置信号制御処理の手順を示すフローチャートである。

【 1 1 9 3 】

まず、メイン C P U 1 0 1 は、条件装置信号制御フラグが初期状態であるか否かを判別する（S 1 0 0 1）。S 1 0 0 1 において、メイン C P U 1 0 1 が、条件装置信号制御フラグが初期状態であると判別したとき（S 1 0 0 1 が Y E S 判定の場合）、メイン C P U 1 0 1 は、条件装置信号制御処理を終了し、処理を試射試験信号制御処理（図 1 6 6 参照）中の S 9 6 7 の処理に移す。

【 1 1 9 4 】

一方、S 1 0 0 1 において、メイン C P U 1 0 1 が、条件装置信号制御フラグが初期状態でないと判別したとき（S 1 0 0 1 が N O 判定の場合）、メイン C P U 1 0 1 は、条件装置信号制御フラグが再遊技状態識別信号のオン状態を示すものであるか否かを判別する（S 1 0 0 2）。

【 1 1 9 5 】

S 1 0 0 2 において、メイン C P U 1 0 1 が、条件装置信号制御フラグが再遊技状態識別信号のオン状態を示すものであると判別したとき（S 1 0 0 2 が Y E S 判定の場合）、メイン C P U 1 0 1 は、条件装置信号制御状態に役物条件装置信号のオン状態をセットする（S 1 0 0 3）。次いで、メイン C P U 1 0 1 は、条件装置 1 ~ 6 信号ポートから R T 状態の情報を試験機用第 1 インターフェースボード 3 0 1（図 7 参照）へ出力する（S 1 0 0 4）。そして、S 1 0 0 4 の処理後、メイン C P U 1 0 1 は、条件装置信号制御処理を終了し、処理を試射試験信号制御処理（図 1 6 6 参照）中の S 9 6 7 の処理に移す。

【 1 1 9 6 】

一方、S 1 0 0 2 において、メイン C P U 1 0 1 が、条件装置信号制御フラグが再遊技状態識別信号のオン状態を示すものでないと判別したとき（S 1 0 0 2 が N O 判定の場合）、メイン C P U 1 0 1 は、条件装置信号制御フラグが再遊技状態識別信号のオフ状態を示すものであるか否かを判別する（S 1 0 0 5）。

【 1 1 9 7 】

10

20

30

40

50

S 1 0 0 5において、メインCPU 1 0 1が、条件装置信号制御フラグが再遊技状態識別信号のオフ状態を示すものであると判別したとき（S 1 0 0 5がYES判定の場合）、メインCPU 1 0 1は、条件装置1～8信号ポートからOFF信号を試験機用第1インターフェースボード301（図7参照）へ出力する（S 1 0 0 6）。そして、S 1 0 0 6の処理後、メインCPU 1 0 1は、条件装置信号制御処理を終了し、処理を試射試験信号制御処理（図166参照）中のS 9 6 7の処理に移す。

【1198】

一方、S 1 0 0 5において、メインCPU 1 0 1が、条件装置信号制御フラグが再遊技状態識別信号のオフ状態を示すものでないと判別したとき（S 1 0 0 5がNO判定の場合）、メインCPU 1 0 1は、条件装置信号制御状態が役物条件装置信号のオン状態であるか否かを判別する（S 1 0 0 7）。

10

【1199】

S 1 0 0 7において、メインCPU 1 0 1が、条件装置信号制御状態が役物条件装置信号のオン状態であると判別したとき（S 1 0 0 7がYES判定の場合）、メインCPU 1 0 1は、条件装置1～8信号ポートから特賞当籤番号の情報を試験機用第1インターフェースボード301（図7参照）へ出力し、この際、条件装置8信号ポートからON信号を試験機用第1インターフェースボード301（図7参照）へ出力する（S 1 0 0 8）。次いで、メインCPU 1 0 1は、条件装置信号出力待ちタイマーに所定の待ち時間（本実施形態では、24.58ms）をセットする（S 1 0 0 9）。次いで、メインCPU 1 0 1は、条件装置信号制御状態に条件装置信号出力待ちの状態をセットする（S 1 0 1 0）。そして、S 1 0 1 0の処理後、メインCPU 1 0 1は、条件装置信号制御処理を終了し、処理を試射試験信号制御処理（図166参照）中のS 9 6 7の処理に移す。

20

【1200】

一方、S 1 0 0 7において、メインCPU 1 0 1が、条件装置信号制御状態が役物条件装置信号のオン状態でないと判別したとき（S 1 0 0 7がNO判定の場合）、メインCPU 1 0 1は、条件装置信号制御状態が条件装置信号出力待ちの状態であるか否かを判別する（S 1 0 1 1）。

【1201】

S 1 0 1 1において、メインCPU 1 0 1が、条件装置信号制御状態が条件装置信号出力待ちの状態であると判別したとき（S 1 0 1 1がYES判定の場合）、メインCPU 1 0 1は、条件装置信号出力待ちタイマーの値が「0」であるか否かを判別する（S 1 0 1 2）。

30

【1202】

S 1 0 1 2において、メインCPU 1 0 1が、条件装置信号出力待ちタイマーの値が「0」でないと判別したとき（S 1 0 1 2がNO判定の場合）、メインCPU 1 0 1は、条件装置信号制御処理を終了し、処理を試射試験信号制御処理（図166参照）中のS 9 6 7の処理に移す。一方、S 1 0 1 2において、メインCPU 1 0 1が、条件装置信号出力待ちタイマーの値が「0」であると判別したとき（S 1 0 1 2がYES判定の場合）、メインCPU 1 0 1は、条件装置信号制御状態に小役条件装置信号のオン状態又は条件装置信号のオフ状態をセットする（S 1 0 1 3）。そして、S 1 0 1 3の処理後、メインCPU 1 0 1は、条件装置信号制御処理を終了し、処理を試射試験信号制御処理（図166参照）中のS 9 6 7の処理に移す。

40

【1203】

ここで再度、S 1 0 1 1の処理に戻って、S 1 0 1 1において、メインCPU 1 0 1が、条件装置信号制御状態が条件装置信号出力待ちの状態でないと判別したとき（S 1 0 1 1がNO判定の場合）、メインCPU 1 0 1は、条件装置信号制御状態が小役条件装置信号のオン状態であるか否かを判別する（S 1 0 1 4）。

【1204】

S 1 0 1 4において、メインCPU 1 0 1が、条件装置信号制御状態が小役条件装置信号のオン状態であると判別したとき（S 1 0 1 4がYES判定の場合）、メインCPU 1 0

50

1は、条件装置1～8信号ポートから小役当籤番号の情報を試験機用第1インターフェースボード301（図7参照）へ出力し、この際、条件装置7信号ポートからON信号を試験機用第1インターフェースボード301（図7参照）へ出力する（S1015）。次いで、条件装置信号出力待ちタイマーに所定の待ち時間（本実施形態では、24.58ms）をセットする（S1016）。次いで、メインCPU101は、条件装置信号制御状態に条件装置信号出力待ちの状態をセットする（S1017）。そして、S1017の処理後、メインCPU101は、条件装置信号制御処理を終了し、処理を試射試験信号制御処理（図166参照）中のS967の処理に移す。

【1205】

一方、S1014において、メインCPU101が、条件装置信号制御状態が小役条件装置信号のオン状態でないと判別したとき（S1014がNO判定の場合）、メインCPU101は、条件装置1～8信号ポートからOFF信号を試験機用第1インターフェースボード301（図7参照）へ出力する（S1018）。そして、S1018の処理後、メインCPU101は、条件装置信号制御処理を終了し、処理を試射試験信号制御処理（図166参照）中のS967の処理に移す。

【1206】

<副制御回路の動作説明>

次に、図171～図173を参照して、副制御回路200のサブCPU201が、プログラムを用いて実行する各種処理の内容について説明する。

【1207】

[サブ側ナビ制御処理]

最初に、図171を参照して、サブ側ナビ制御処理について説明する。なお、図171は、サブ側ナビ制御処理の手順を示すフローチャートである。

【1208】

まず、サブCPU201は、ナビデータを取得したか否かを判定する（S1101）。サブCPU201は、主制御基板71から受信したスタートコマンドデータの中から主制御基板71で決定されたナビデータを取得する。それゆえ、S1101の処理では、サブCPU201は、受信したスタートコマンドデータの中にナビデータが含まれていたか否かを判定する。

【1209】

S1101において、サブCPU201が、ナビデータを取得したと判別したとき（S1101がYES判定の場合）、サブCPU201は、ナビデータに応じたサブ側ナビデータをセットする（S1102）。例えば、サブCPU201がナビデータ「4」を取得した場合、図63に示すように、サブ側ナビデータとして押し順「左、中、右」を報知するためのナビデータが、この処理でセットされる。この結果、メイン側及びサブ側の双方において停止操作の内容を報知することができる。そして、S1102の処理後、サブCPU201は、サブ側ナビ制御処理を終了する。

【1210】

一方、S1101において、サブCPU201が、ナビデータを取得していないと判別したとき（S1101がNO判定の場合）、サブCPU201は、ナビ（停止操作の報知）の必要があるか否かを判定する（S1103）。本実施形態では、サブCPU201は、例えば、主制御基板71においてフラグ変換抽籤が行われた場合や、主制御基板71において所定の役が内部当籤役として決定された場合に、ナビの必要があると判定する。なお、フラグ変換抽籤の結果や、内部当籤役の種別は、スタートコマンドデータに含まれている。それゆえ、S1103の処理では、サブCPU201は、スタートコマンドデータに含まれるこれらの各種情報に基づいて、ナビの必要があるか否かを判定する。

【1211】

S1103において、サブCPU201が、ナビの必要がないと判別したとき（S1103がNO判定の場合）、サブCPU201は、サブ側ナビ制御処理を終了する。

【1212】

10

20

30

40

50

一方、S 1 1 0 3において、サブCPU 2 0 1が、ナビの必要があると判別したとき（S 1 1 0 3がYES判定の場合）、サブCPU 2 0 1は、各種抽籤結果に応じたサブ側ナビデータをセットする（S 1 1 0 4）。例えば、内部当籤役「F__確チリリプ」が決定され、かつ、フラグ変換抽籤に当籤している場合、サブCPU 2 0 1は、この処理において、略称「3連チリリプ」に係る図柄組合せを表示するためのナビデータ（例えば、順押しでチリ図柄を狙わせるナビデータ）をセットする。また、例えば、内部当籤役「F__確チリリプ」が決定され、かつ、フラグ変換抽籤に非当籤であった場合、サブCPU 2 0 1は、この処理において、略称「リプレイ」に係る図柄組合せを表示するためのナビデータ（例えば、順押し以外の押し順を示すナビデータ）をセットする。これらの処理により、メイン側で停止操作の内容を報知しない場合であっても、サブ側単独で停止操作の内容を報知

10

【1213】

[遊技者登録処理]

次に、図172を参照して、遊技者登録処理について説明する。なお、図172は、遊技者登録処理の手順を示すフローチャートである。

【1214】

まず、サブCPU 2 0 1は、登録操作を受け付けたか否かを判別する（S 1 1 1 1）。例えば、サブ表示装置18のメニュー画面222（図5B参照）において登録ボタン222bの操作を受け付け、その場合に表示される登録画面（不図示）において所定の操作を受け付けると、サブCPU 2 0 1は、登録操作を受け付けたと判定する。

20

【1215】

S 1 1 1 1において、サブCPU 2 0 1が、登録操作を受け付けたと判別したとき（S 1 1 1 1がYES判定の場合）、サブCPU 2 0 1は、遊技者登録状態をセットする（S 1 1 1 2）。なお、遊技者登録状態がセットされている状況では、サブCPU 2 0 1は、サブ表示装置18に遊技情報画面223, 224, 225（図5C～5E参照）が表示可能となるようにサブ表示装置18の表示画面を制御する。そして、S 1 1 1 2の処理後、サブCPU 2 0 1は、遊技者登録処理を終了する。

【1216】

一方、S 1 1 1 1において、サブCPU 2 0 1が、登録操作を受け付けていないと判別したとき（S 1 1 1 1がNO判定の場合）、サブCPU 2 0 1は、登録削除操作を受け付けたか否かを判別する（S 1 1 1 3）。例えば、サブ表示装置18の登録画面において特定の操作を受け付けると、サブCPU 2 0 1は、登録削除操作を受け付けたと判定する。

30

【1217】

S 1 1 1 3において、サブCPU 2 0 1が、登録削除操作を受け付けていないと判別したとき（S 1 1 1 3がNO判定の場合）、サブCPU 2 0 1は、遊技者登録処理を終了する。

【1218】

一方、S 1 1 1 3において、サブCPU 2 0 1が、登録削除操作を受け付けたと判別したとき（S 1 1 1 3がYES判定の場合）、サブCPU 2 0 1は、遊技者登録状態をクリアする（S 1 1 1 4）。なお、遊技者登録状態がクリアされている状況では、サブCPU 2 0 1は、サブ表示装置18に遊技情報画面223, 224, 225（図5C～5E参照）が表示不可能となるようにサブ表示装置18の表示画面を制御する。そして、S 1 1 1 4の処理後、サブCPU 2 0 1は、遊技者登録処理を終了する。

40

【1219】

[履歴管理処理]

次に、図173を参照して、履歴管理処理について説明する。なお、図173は、履歴管理処理の手順を示すフローチャートである。

【1220】

まず、サブCPU 2 0 1は、主制御基板71から受信した各種コマンドデータから遊技結果を取得する（S 1 1 2 1）。例えば、サブCPU 2 0 1は、この処理において、スター

50

トコマンドデータから内部当籤役として決定された役の種類を把握することができる。また、例えば、サブCPU201は、この処理において、入賞作動コマンドデータから表示された図柄組合せ（すなわち、内部当籤役として決定された役の入賞の有無）を把握することができる。さらに、例えば、サブCPU201は、この処理において、スタートコマンドデータなどから現在の遊技状態や遊技状態の移行状況を把握することができる。

【1221】

次いで、サブCPU201は、取得した遊技結果に基づいて、遊技履歴の更新処理を行う（S1122）。この処理により、サブCPU201は、各種コマンドデータから取得した遊技結果に基づいて、例えば、ボーナス回数、ART回数、ゲーム数（遊技回数）、CZ回数、CZ成功回数、それぞれの役の当籤回数及び当籤確率などの様々な遊技履歴を管理することができる。そして、S1122の処理後、サブCPU201は、履歴管理処理を終了する。

10

【1222】

<各種効果>

本実施形態のパチスロ1では、その遊技性において、次のような各種効果が得られる。

【1223】

[CT中の継続期間の管理]

本実施形態のパチスロ1では、通常ARTの継続期間を延長可能な上乘せチャンスゾーンとしてCTを設け、このCT中の内部当籤役（サブフラグ）に基づいてARTゲーム数の上乘せを行う。なお、CTでは、1セット8回の遊技が行われるが、CT中にARTゲーム数の上乘せを行うことができた場合には遊技回数の減算を行わずに、上乘せできない場合に限り遊技回数を減算する。それゆえ、遊技者にとってみれば、CTがいつまで続くか分からず、また、上乘せが行われている限りCTが終了することがないため、CT中の遊技の興趣を高めることができる。

20

【1224】

また、本実施形態において、CT中にサブフラグEX「3連チリリブ」に当籤（サブフラグ変換抽籤に当籤）して上乘せが行われた場合には、1セット8回のCT遊技も再セット（ストック）される。この場合、例えば、CTの遊技期間が終了直前になっても、サブフラグEX「3連チリリブ」当籤時には、上乘せが行われ、CTが初めから再開されることになるので、CT中の遊技に対して強い関心を抱くことになり、退屈することなく遊技を継続できるとともに、CT中の遊技の興趣を高めることができる。

30

【1225】

また、サブフラグEX「3連チリリブ」当籤時の上乘せは、内部当籤役「F__確チリリブ」又は「F__1確チリリブ」が内部当籤役として決定され、かつ、フラグ変換抽籤に当籤した場合に限り行われる。それゆえ、遊技者に対して過大な利益を与えてしまうことを防止することができ、遊技者と遊技店との間の利益のバランスをとることができる。なお、上記実施形態では、CT中に、内部当籤役「F__確チリリブ」又は「F__1確チリリブ」が決定された際にはフラグ変換抽籤に必ず当籤する例（図54参照）を説明したが、本発明はこれに限定されず、フラグ変換抽籤の当籤確率は、遊技者及び遊技店間の利益のバランスに応じて適宜設定することができる。

40

【1226】

[CT中の「3連チリリブ」当籤時の上乘せゲーム数]

上記実施形態のパチスロ1では、CT中にサブフラグ「3連チリリブ」に当籤し、該サブフラグ「3連チリリブ」に基づく上乘せが行われた回数が所定回数を超えると、1回の上乗せ抽籤あたりに当籤するARTの上乗せゲーム数が増加する（図55参照）。また、上述のように、ARTゲーム数の上乘せが行われている限り、CTは終了することなく、また、サブフラグEX「3連チリリブ」に当籤するとCTの再セットが行われる。それゆえ、遊技者からすると、CTが継続するほど1回（1ゲーム）当りの上乘せ量が増えることについての期待を持つことができ、CT中の興趣が向上する。さらに、1回（1ゲーム）当りの上乘せ量を増やす契機となる回数は、CTの1セット分の基本遊技回数（8回）よ

50

りも多い回数（9回以上）であるため、遊技者に対して過大な利益を与えてしまうことを防止でき、遊技者及び遊技店間の利益のバランスをとる（良好に保つ）ことができる。

【1227】

[メイン側で行うボーナス報知]

上記実施形態のパチスロ1では、情報表示器6の指示モニタ（不図示）に、ボーナス役（BB役）に係る図柄組合せを表示させるための停止操作の情報に一義的に対応付けた数値「10」（又は「11」）を表示することにより、メイン側でボーナス報知を行う（図63参照）。しかしながら、通常、パチスロでは、有効ライン上に引き込む（停止表示する）図柄の優先順位が定められており、ボーナス役とその他の役とが重複して内部当籤役として決定されている場合、優先順位により、ボーナス役に係る図柄組合せを引き込めることもあれば、引き込めないこともある。

10

【1228】

例えば、本実施形態のパチスロ1において、ボーナス役と、内部当籤役「はずれ」、「F__特殊役1」、「F__特殊役2」及び「F__特殊役3」のいずれかとが重複して決定されている場合には、ボーナス役に係る図柄組合せを引き込むことができるが、ボーナス役と、内部当籤役「はずれ」、「F__特殊役1」、「F__特殊役2」及び「F__特殊役3」以外の役とが重複して決定されている場合には、ボーナス役に係る図柄組合せを引き込むことができない。それゆえ、本実施形態において、ボーナス役と、内部当籤役「はずれ」、「F__特殊役1」、「F__特殊役2」及び「F__特殊役3」のいずれかとが重複して決定されている場合に限り、指示モニタに数値「10」（又は「11」）を表示する。この場合、メイン側のナビ（ボーナス報知）を、ボーナス役を入賞させることのできる適切なタイミングで行うことができる。

20

【1229】

また、本実施形態のパチスロ1において、ボーナス役と、内部当籤役「はずれ」、「F__特殊役1」、「F__特殊役2」及び「F__特殊役3」のいずれかとが重複して決定されたとしても、ボーナス役が最初に当籤してから所定回数の遊技が経過するまでは、指示モニタに数値「10」（又は「11」）を表示せず（ナビをせず）、所定回数の遊技が経過したことを条件に、指示モニタに数値「10」（又は「11」）を表示する。

【1230】

なお、例えば、ボーナス役に当籤したことを契機として複数回の遊技にわたり行われる演出（いわゆる連続演出）が行われた場合、この連続演出の最中に、指示モニタに数値「10」（又は「11」）を表示すると、連続演出の意味が薄れてしまい、興趣を損ねてしまう可能性がある。それゆえ、本実施形態のパチスロ1では、所定回数の遊技が経過するまでは、指示モニタによる表示を行わず、所定回数の遊技が経過した後に、指示モニタによる表示を行う。その結果、演出効果を損ねることなく、メイン側でのボーナス報知を行うことができる。

30

【1231】

[メイン側及びサブ側の両方で行う報知とサブ側単独で行う報知]

本実施形態のパチスロ1では、停止操作の態様（押し順）に応じて表示される図柄組合せが異なる役を複数種類設けるとともに（図24参照）、これら複数種類の役には、表示される図柄組合せによって異なる特典が付与される役と、表示される図柄組合せが異なったとしても同一の特典が付与される役とが含まれる。

40

【1232】

例えば、略称「ベル」に係る図柄組合せが表示された場合と略称「ベルこぼし目」に係る図柄組合せが表示された場合とでは払い出されるメダルの枚数が異なり、これらの役は、表示される図柄組合せによって異なる特典が付与される役である。また、略称「リプレイ」に係る図柄組合せが表示された場合と略称「RT2移行リプ」に係る図柄組合せが表示された場合とでは再遊技の作動に加えてRT状態の移行が行われるか否かが異なるため、内部当籤役「F__3択ベル__1st」や「F__維持リプ__1st」もまた、表示される図柄組合せによって異なる特典が付与される役である。

50

【 1 2 3 3 】

一方、略称「3連チリリブ」に係る図柄組合せが表示された場合と略称「リプレイ」に係る図柄組合せが表示された場合とは、両者ともに再遊技の作動が行われるだけである。それゆえ、内部当籤役「F__確チリリブ」又は「F__1確チリリブ」は、表示される図柄組合せが異なったとしても同一の特典が付与される役である。なお、内部当籤役「F__確チリリブ」又は「F__1確チリリブ」は、上述のように、フラグ変換抽籤の結果に応じて付与する特典が異なる役であるが、表示される図柄組合せが付与する特典に影響を与える役ではない。

【 1 2 3 4 】

本実施形態のパチスロ1では、表示される図柄組合せによって異なる特典が付与される役に対して、メイン側及びサブ側の両方で報知を行うが、表示される図柄組合せが異なったとしても同一の特典が付与される役に対しては、メイン側の指示モニタでは報知を行わずに、サブ側の表示装置11のみで報知を行う。このような報知機能を設けることにより、特典に影響する報知は、特典を管理するメイン側の指示モニタで適切に行いつつ、特典に影響しない報知は、サブ側の表示装置11で多様性のあるナビで行うことができる。

10

【 1 2 3 5 】

[遊技履歴の表示機能]

本実施形態のパチスロ1では、表示装置11（プロジェクタ機構211及び表示ユニット212）とは別にサブ表示装置18を設け、このサブ表示装置18により遊技者に役立つ様々な情報を表示する。例えば、図5A～5Eに示すように、概要遊技履歴を表すトップ画面221、パチスロ1に対する様々な操作が可能なメニュー画面222、詳細遊技履歴を表す遊技情報画面223、224、225をサブ表示装置18に表示することができる。

20

【 1 2 3 6 】

また、本実施形態のパチスロ1は、サブ表示装置18を介して、遊技を行う遊技者を登録可能にする機能、及び、登録された遊技者に対して固有のサービスを提供する機能を備える。例えば、本実施形態では、遊技者の登録を受け付けていない場合には、詳細遊技履歴を表す遊技情報画面223、224、225をサブ表示装置18に表示不可能にするが、遊技者の登録を受け付けている場合には、詳細遊技履歴を表す遊技情報画面223、224、225をサブ表示装置201に表示可能にする。より詳細な遊技履歴を確認できるようにすることは、遊技者の利便性の向上につながるため、本実施形態では、遊技者の登録を受け付けている場合に、利便性を向上させることができる。

30

【 1 2 3 7 】

また、本実施形態のパチスロ1では、サブ表示装置18は、演出を行う表示装置11とは別体に設けられる。サブ表示装置18は、液晶中継基板87を介して副制御基板72（サブCPU201）により制御される。また、表示装置11を構成する表示ユニット212は、役物中継基板（不図示）を介して副制御基板72（サブCPU201）により制御される。それゆえ、本実施形態では、サブ表示装置18を、表示装置11とは別個に制御することができる。具体的には、遊技中（すなわち、表示装置11による演出の実行中）であっても、サブ表示装置18の表示画面を切り替えることができる。それゆえ、遊技中であっても、表示装置11により実行されている演出を邪魔することなく、サブ表示装置18の表示を切り替えることにより、遊技者は様々な情報を取得することができる。

40

【 1 2 3 8 】

また、本実施形態のパチスロ1の表示装置11では、プロジェクタ機構211からの照射光の照射により映像を出現させる複数のスクリーン機構（表示ユニット212）を切り替えることにより、平面状の映像表示を用いた演出、奥行き感（立体感）のある映像表示を用いた演出、及び、湾曲した映像表示を用いた演出を実行する場合、演出効果を著しく高めることができる。しかしながら、このような情報の表示形態は、演出中に遊技履歴などの演出とは関係ない情報を表示することには適さない。それゆえ、本実施形態のパチスロ1では、表示装置11とは別個に設けられたサブ表示装置18に演出とは関係ない情報を表示することができるので、演出効果を損なうことなく、かつ、遊技履歴などの各種情報

50

を適切に表示することができる。

【 1 2 3 9 】

ところで、一般的なパチスロでは、遊技者側から見て、台座部 1 3 の右側にメダル投入口 1 4 が設けられ、台座部 1 3 の左側にベットボタン 1 5 a , 1 5 b やスタートレバー 1 6 が設けられる。それゆえ、通常、遊技を進行させる際、遊技者は台座部 1 3 の右側又は左側（側方）の操作部を操作することになる。

【 1 2 4 0 】

それに対して、本実施形態のパチスロ 1 では、台座部 1 3 から略垂直に立設する面の側方（左側）にサブ表示装置 1 8 を設け、このサブ表示装置 1 8 の画面上にサブ表示装置 1 8 の表示画面を切り替えるためのタッチセンサ 1 9 が設けられる。それゆえ、本実施形態では、遊技中に遊技者の手が位置する場所にサブ表示装置 1 8 やその表示を制御する入力装置（タッチセンサ 1 9 ）が設けられることになるので、遊技者の操作性を向上させることができる。特に、本実施形態のように、タッチセンサ 1 9 付きのサブ表示装置 1 8 を、台座部 1 3 の水平面から立設する面に設けた場合には、遊技者は、台座部 1 3 に自身の手を置きながら、サブ表示装置 1 8 を操作することができる。この場合、遊技者の操作性が向上するだけでなく、操作に伴う遊技者の疲労も軽減することができ、この結果、稼働率の向上も期待することができる。

10

【 1 2 4 1 】

[規定外 R O M 領域及び規定外 R A M 領域]

本実施形態のパチスロ 1 では、図 1 2 B に示すように、遊技者により実施される遊技の遊技性に直接関与しない各種処理（遊技性に影響を与えない各種処理）に使用される各種プログラム及び各種データ（テーブル）を、メイン R O M 1 0 2 内において、遊技用 R O M 領域とは異なるアドレスに配置された規定外 R O M 領域に格納する。

20

【 1 2 4 2 】

このようなメイン R O M 1 0 2 の構成では、従来の規則上では、プログラム等の配置不可とされていた R O M 領域（規定外 R O M 領域）に、遊技者が実際に行う遊技そのものに不要なプログラム及びデータを配置することができる。それゆえ、本実施形態では、主制御基板 7 1 のメイン R O M 1 0 2 内において、遊技用 R O M 領域の容量の圧迫を回避することができるとともに、メイン R O M 1 0 2 内におけるプログラム及びテーブルの拡張性を高めることができる。

30

【 1 2 4 3 】

[電源投入（リセット割込）時処理により得られる効果]

本実施形態のパチスロ 1 の電源投入（リセット割込）時処理では、図 6 4 に示すように、電源復帰直後（サムチェック前）に最初の 1 . 1 1 7 2 m s 周期の割込処理を行い（ S 7 及び S 8 ）、主制御回路 9 0 から副制御回路 2 0 0 に無操作コマンドが送信される。このように電源復帰直後に割込処理を許可することにより、電源復帰後、最短時間で無操作コマンドが送信され、主制御回路 9 0 及び副制御回路 2 0 0 間の通信接続を確立することができ、主制御回路 9 0 及び副制御回路 2 0 0 間の通信動作を安定化させることができる。

【 1 2 4 4 】

また、電源復帰直後に送信される無操作コマンドを構成する通信パラメータ 1 ~ 5 には、電源復帰時に、それぞれ L レジスタ、H レジスタ、E レジスタ、D レジスタ及び C レジスタに格納されているデータがセットされる。それゆえ、本実施形態では、電源復帰直後の割込処理で送信される無操作コマンドのサム値（ B C C ）を、電源復帰毎に異ならせることができ、ゴト等の不正行為を抑制することができる。

40

【 1 2 4 5 】

さらに、電源投入（リセット割込）時処理中の S 1 3 の処理において行われる、エラーコード「 r r 」を情報表示器 6 内の 2 桁の 7 セグ L E D に表示する際の制御は、一つの「 L D W 」命令（所定の読み出し命令）により実行され、2 桁の 7 セグ L E D への 7 セグコモン出力（選択）データの出力動作と 7 セグカソード出力データの出力動作とが同時に行われる（図 6 5 C 参照）。すなわち、本実施形態のパチスロ 1 では、電源投入（リセット割

50

込)時処理において、2桁の7セグLEDをダイナミック点灯制御する際に、7セグコモン出力(選択)データと、7セグカソード出力データとが同時に出力される。

【1246】

この場合、ソースプログラム上において、7セグLEDのダイナミック点灯制御に必要な命令コード数を減らすことができる。それゆえ、本実施形態では、ソースプログラムの容量(メインROM102の使用容量)を低減することができ、メインROM102において、空き容量を確保する(増大させる)ことができ、増えた空き容量を活用して、遊技性を高めることができる。

【1247】

[遊技復帰処理で得られる効果]

本実施形態のパチスロ1の遊技復帰処理では、図66に示すように、電断発生時の各ポートの入出力状態を電源復帰時に担保するとともに、電断時に回胴回転中の場合には、電源復帰時にリール制御管理情報を取得してリールの再回転開始に必要な処理も行う(S25~S32参照)。それゆえ、本実施形態では、リール回転中の電断から復帰したときであっても、安定して、リールの再回転制御を行うことが可能となり、遊技者に不快感を与えることが無くなる。

【1248】

また、本実施形態のパチスロ1は、上述のように、遊技機用のセキュリティ機能付きマイクロプロセッサ91を備える。そして、このマイクロプロセッサ91には、ソースプログラム上において規定可能な該マイクロプロセッサ91に特有の命令コード(メインCPU101専用命令コード)が各種設けられており、このメインCPU101専用命令コードを各種処理において用いることにより、処理の効率化やプログラム容量の削減などを可能にしている。

【1249】

例えば、遊技復帰処理では、図67に示すように、ソースプログラム上において、メインCPU101専用命令コードの一つである「LDQ」命令が用いられる。「LDQ」命令は、Qレジスタ(拡張レジスタ)を用いてアドレス指定を行う命令コードであり、上述のように、直値でメインROM102、メインRAM103やメモリーマップI/Oにアクセスすることができる。この場合、アドレス設定に係る命令コードを省略することができ、その分、遊技復帰処理のソースプログラムの容量(メインROM102の使用容量)を低減することができる。この結果、本実施形態では、メインROM102において、空き容量を確保する(増大させる)ことができ、増えた空き容量を活用して、遊技性を高めることができる。

【1250】

[設定変更確認処理で得られる効果]

本実施形態のパチスロ1の設定変更確認処理では、図69Aに示すように、ソースプログラム上において、メインCPU101専用命令コードである、「BITQ」命令及び「SETQ」命令(所定の命令)が用いられる。「BITQ」命令及び「SETQ」命令はいずれも、Qレジスタ(拡張レジスタ)を用いてアドレス指定を行う命令コードであり、これらの命令コードを使用した場合、上述のように、直値でメインRAM103やメモリーマップI/Oにアクセスすることができる。この場合、アドレス設定に係る命令コードを省略することができ、その分、ソースプログラムの容量(メインROM102の使用容量)を低減することができる。この結果、本実施形態では、メインROM102の空き容量を増やすことが可能となるとともに、処理の高速化も図ることができる。

【1251】

また、設定変更確認処理中のS46の設定変更/設定確認開始時及びS57の設定変更/設定確認終了時で行う設定変更コマンド(初期化コマンド)の生成格納処理は、図69A及び69Bに示すように、ソースプログラム上において、メインCPU101専用命令コードである「CALLF」命令により実行される。そして、S46の「CALLF」命令で指定するジャンプ先のアドレスは、S57の「CALLF」命令で指定するジャンプ先

10

20

30

40

50

のアドレスと同じである。すなわち、本実施形態では、設定変更時（遊技機起動時）、設定確認開始時（通常動作中）及び設定確認終了時に副制御回路 200 に送信する設定変更コマンド（初期化コマンド）の生成格納処理を実行するためのソースプログラムが、互いに同じであり、S46 及び S57 の両処理において、そのソースプログラムが共有化（モジュール化）されている。

【1252】

この場合、S46 及び S57 の両処理において、それぞれ別個に設定変更コマンドの生成格納処理のソースプログラムを設ける必要が無くなるので、その分、ソースプログラムの容量（メインROM102の使用容量）を低減することができる。この結果、本実施形態では、メインROM102において、空き容量を確保する（増大させる）ことができ、増えた空き容量を活用して、遊技性を高めることができる。

10

【1253】

[設定変更コマンド生成格納処理で得られる効果]

本実施形態のパチスロ1の設定変更コマンド生成格納処理では、図70に示すように、設定値が通信パラメータ3としてEレジスタに格納され、RT情報が通信パラメータ5としてCレジスタに格納される。すなわち、設定変更コマンド（初期化コマンド）を構成する通信パラメータ1～5のうち、通信パラメータ3及び5は副制御回路200側で使用（解析）される通信パラメータ（使用パラメータ）であり、これらの通信パラメータには新たな情報がセットされる。一方、設定変更コマンド（初期化コマンド）を構成するその他の通信パラメータ1, 2及び4は、副制御回路200側で使用（解析）されない通信パラメータ（未使用パラメータ）であり、通信パラメータ1, 2及び4に対しては、現時点でLレジスタ、Hレジスタ及びDレジスタにそれぞれ格納されている値がセットされる。それゆえ、設定変更コマンド（初期化コマンド）送信時における通信パラメータ1, 2及び4の値は不定値となる。この場合、設定変更コマンドのサム値（BC C）を送信毎に不定値にすることができ、ゴト等の不正行為を抑制することができる。

20

【1254】

[通信データ格納処理で得られる効果]

本実施形態のパチスロ1の通信データ格納処理では、図72に示すように、Aレジスタに格納されたデータが通信コマンドの種別データとしてセットされ、Lレジスタ、Hレジスタ、Eレジスタ、Dレジスタ及びCレジスタに格納されたデータがそれぞれ通信コマンドの通信パラメータ1～5としてセットされ、Bレジスタに格納されたデータが通信コマンドの遊技状態フラグデータとしてセットされる。すなわち、本実施形態では、1パケット（8バイト）の通信データ（コマンドデータ）を作成する際に、各種パラメータをレジスタから転送して通信データ一時格納領域（通信バッファ）に格納する。

30

【1255】

この場合、未使用パラメータを含むコマンドデータを作成した時には、作成時毎に、未使用パラメータの値が不定値となる。すなわち、未使用パラメータを含むコマンドデータでは、同じ種別のコマンドデータあり、かつ、使用パラメータの値が同一であっても、コマンド作成毎に、コマンドデータのサム値（BC Cデータ）が可変可能となる。それゆえ、本実施形態では、未使用パラメータを不定値とすることにより、通信データの解析を困難にしてゴト等の不正行為を抑止することができるとともに、不必要なゴト対策処理を加える必要がないため、ゴト対策処理の追加による、主制御回路90のプログラム容量の圧迫を抑制することができる。

40

【1256】

[通信データポインタ更新処理で得られる効果]

本実施形態のパチスロ1の通信データポインタ更新処理では、図75Aに示すように、ソースプログラム上において、メインCPU101専用命令コードである、「ICPLD」命令が用いられる。

【1257】

通信データポインタ更新処理において、「ICPLD」命令は、送信バッファの上限判定

50

命令と、判断分岐命令とが一体になっている命令コードであるので、各命令処理を別個に実行するための命令コードを設ける必要がなくなる。それゆえ、「ICPLD」命令を用いることにより、通信データポインタ更新処理のソースプログラムの容量（メインROM 102の使用容量）を低減することができる。この結果、本実施形態では、メインROM 102において、空き容量を確保する（増大させる）ことができ、増えた空き容量を活用して、遊技性を高めることができる。

【1258】

[チェックサム生成処理及びサムチェック処理で得られる効果]

本実施形態のパチスロ1において、電断時に行われるチェックサム生成処理（規定外）では、図77に示すように、メインRAM103のデータを順次加算することにより、チェックサムが算出される。一方、電源投入時（電源復帰時）に行われるサムチェック処理（規定外）では、図79に示すように、電断発生時に生成されたチェックサムの値を、電源復帰時のメインRAM103に格納されたデータで順次減算し、最終的な減算結果が「0」であるか否かに基づいて、異常の発生の有無を判定する。すなわち、本実施形態では、電断発生時のチェックサムの生成処理は加算方式で行われ、電源復帰時のチェックサムの判定処理は減算方式で行われる。

10

【1259】

このようなチェックサムの生成処理及び判定処理を採用した場合、電源復帰時に再度チェックサムを生成して、該チェックサムを電断発生時のチェックサムと照合する処理が不要となる。この場合、ソースプログラム上において、照合命令コードを省略することができ、ソースプログラムの容量を低減することができる。この結果、本実施形態では、メインROM102において、照合命令コードの省略分に対応する空き容量を確保する（増やす）ことができ、増えた空き容量を活用して、遊技性を高めることが可能になる。

20

【1260】

[メダル受付・スタートチェック処理で得られる効果]

本実施形態のパチスロ1のメダル受付・スタートチェック処理では、図83に示すように、設定変更確認処理（S233の処理）が行われるが、この処理は、遊技状態に関係なく実行される。それゆえ、本実施形態では、遊技状態がボーナス状態（特賞作動状態）であっても、設定値及びホールメニュー（各種履歴データ（エラー、電断履歴等））を確認することができ、ゴト等の不正行為を抑制することができる。

30

【1261】

[メダル投入処理で得られる効果]

本実施形態のパチスロ1のメダル投入処理では、図85に示すように、S244の処理において、メダル投入枚数表示用のLED点灯データが、テーブルを参照したループ処理でなく、演算処理により生成される。具体的には、図86に示すソースプログラム中の一連のソースコード「LDA, L」～「OR, L」が順次実行されるにより、メダル投入枚数表示用のLED点灯データが生成される。

【1262】

メダル投入枚数表示用のLED点灯データを演算処理により生成した場合、メインROM102のテーブル領域の空き容量を増やすことができるとともに、プログラムの容量増を最小限に抑えることができる。すなわち、本実施形態のメダル投入処理では、メダル投入LED表示の処理を効率化できるとともに、メインROM102の空き容量を確保し（増やし）、増えた空き領域を活用して、遊技性を高めることができる。

40

【1263】

[メダル投入チェック処理で得られる効果]

本実施形態のパチスロ1のメダル投入チェック処理（図87参照）において、S257のメダルセンサ入力状態の正常変化値の生成処理は、テーブルを参照して取得する処理ではなく、演算処理により行われる。具体的には、図88に示すソースプログラム中のソースコード「RLA」及び「AND, cBX, MDINSW」が順次実行されることにより、メダルセンサ入力状態正常変化値が算出される。

50

【 1 2 6 4 】

メダルセンサ入力状態の変化態様の検知処理をテーブル参照処理から演算処理に変更することにより、メインROM 102のテーブル格納領域の空き容量を増やすことができるとともに、プログラムの容量増を最小限に抑えることができる。それゆえ、上述した処理手法を採用することにより、メダル投入センサ状態の変化態様の検知処理を効率化することができるとともに、メインROM 102において増えた空き容量を活用して、遊技性を高めることができる。

【 1 2 6 5 】

[内部抽籤処理で得られる効果]

本実施形態のパチスロ1の内部抽籤処理(図92参照)において、S305の判定データの取得処理は、図93A中のソースコード「LDIN AC, (HL)」により実行される。この「LDIN」命令の実行により、S305の処理では、Aレジスタに、判定データ(「抽籤値選択テーブルor抽籤計数テーブル」の値)が格納され、Cレジスタに当り要求フラグステータス(「特賞当籤番号+小役当籤番号」の値)が格納される。また、「LDIN」命令の実行により、HLレジスタにセットされているアドレスが+2更新(2加算)される。

【 1 2 6 6 】

すなわち、内部抽籤処理中のS305の判定データの取得処理では、一つの命令コード(「LDIN」命令)により、データのロード処理及びアドレスの更新処理の両方を行うことができる。この場合、ソースプログラム上において、アドレス設定に係る命令コードを省略することができ、その分、ソースプログラムの容量(メインROM 102の使用容量)を低減することができる。この結果、本実施形態では、メインROM 102において、空き容量を確保する(増大させる)ことができ、増えた空き容量を活用して、遊技性を高めることができる。

【 1 2 6 7 】

また、内部抽籤処理のS309の設定値データ(0~5のいずれか)の加算処理は、メインCPU101が図93B中のソースコード「MUL A, 6」及び「ADDQ A, (.LOW.WWAVENUM)」をこの順で実行することにより行われる。

【 1 2 6 8 】

「MUL」命令は、メインCPU101専用の乗算処理の命令コードであり、この命令の実行は、マイクロプロセッサ91に含まれる演算回路107(図9参照)により実行される。すなわち、本実施形態のパチスロ1では、ソースプログラム上における乗算処理及び除算処理を実行するための演算専用回路(演算回路107)が設けられているので、乗算処理及び除算処理の効率化を図ることができる。

【 1 2 6 9 】

また、「ADDQ」命令(所定の加算命令)は、メインCPU101専用命令コードであり、Qレジスタ(拡張レジスタ)を用いてアドレス指定を行う命令コードである。そして、この「ADDQ」命令を用いれば、直値により、メインROM 102、メインRAM 103やメモリーマップI/Oにアクセスすることができる。それゆえ、「ADDQ」命令の使用により、内部抽籤処理のソースプログラム上において、アドレス設定に係る命令を省略することができ、その分、ソースプログラムの容量(メインROM 102の使用容量)を低減することができる。この結果、本実施形態では、メインROM 102において、空き容量を確保する(増大させる)ことができ、増えた空き容量を活用して、遊技性を高めることができる。

【 1 2 7 0 】

[図柄設定処理で得られる効果]

本実施形態のパチスロ1の図柄設定処理(図97参照)において、S330の圧縮データ格納処理は、メインCPU101が図99中のソースコード「CALLF SB__BTEP__00」を実行することにより行われる。「CALLF」命令は、上述のようにメインCPU101専用の2バイト命令コードであり、図99中のソースコード「CALLF

10

20

30

40

50

「SB__BTEP__00」が実行されると、「SB__BTEP__00」で指定されているアドレスに、処理をジャンプさせ、圧縮データ格納処理が開始される。

【1271】

また、図柄設定処理中のS329の当り要求フラグ格納領域のアドレスのセット処理は、メインCPU101が図99中のソースコード「LDQ DE, .LOW.wWAVEBIT」を実行することにより行われる。すなわち、S329の処理は、Qレジスタ（拡張レジスタ）を用いたメインCPU101専用の「LDQ」命令により行われる。この場合、図柄設定処理のソースプログラム上において、アドレス設定に係る命令コードを省略することができ、その分、図柄設定処理のソースプログラムの容量（メインROM102の使用容量）を低減することができる。この結果、本実施形態では、メインROM102において、空き容量を確保する（増大させる）ことができ、増えた空き容量を活用して、遊技性を高めることができる。

10

【1272】

また、本実施形態では、上述した図柄設定処理中のS324～S330で説明した処理手順で入賞に係るデータの圧縮・展開処理を行い、かつ、その処理の中で上述したメインCPU101専用命令コードを用いることにより、入賞に係るデータの圧縮・展開処理の効率化を図ることができるとともに、限られたメインRAM103の容量を有効活用することができる。

【1273】

[サブフラグ変換処理で得られる効果]

本実施形態のパチスロ1において、サブフラグ変換処理のソースプログラム上で実際に参照される、図107に示すサブフラグ変換テーブルでは、各サブフラグに対してサブフラグ変換制御データ（制御ステータス）が対応付けられている。この際、同種のサブフラグに対しては、同じサブフラグ変換制御データ（制御ステータス）が対応付けられている。

20

【1274】

例えば、サブフラグ「3連チリリプA」及び「3連チリリプB」に対しては、サブフラグ変換制御データ（制御ステータス）「00000011B」が共通して割り付けられている。そして、内部当籤役（サブフラグ）をサブフラグEXに変換する際のフラグ変換抽籤処理では、サブフラグに対応付けられたサブフラグ変換制御データ（制御ステータス）に基づいて、抽籤が行われる。

30

【1275】

このように、メイン側で管理するサブフラグ変換テーブルにおいて、同種の内部当籤役（サブフラグ）に対して共通のサブフラグ変換制御データを設けることにより、該変換テーブルの汎用性が高くなり、機種変更に伴う変換プログラムの変更も軽微な変更で対応可能となるので、開発コストの増大を抑制することができる。

【1276】

[ナビセット処理で得られる効果]

本実施形態のパチスロ1のナビセット処理では、図109に示すように、ソースプログラム上において、Qレジスタ（拡張レジスタ）を用いてアドレス指定を行う「LDQ」命令（メインCPU101専用命令コード）が用いられる。

40

【1277】

それゆえ、本実施形態のナビセット処理では、ソースプログラム上において、アドレス設定に係る命令を省略することができ、その分、ソースプログラムの容量（メインROM102の使用容量）を低減することができる。この結果、本実施形態では、メインROM102において、空き容量を確保する（増大させる）ことができ、増えた空き容量を活用して、遊技性を高めることができる。

【1278】

[テーブルデータ取得処理で得られる効果]

本実施形態のパチスロ1のテーブルデータ取得処理（図120参照）において、S581～S584の1段階目のテーブルデータ取得処理では、CT中CT当籤抽籤テーブル（図

50

122参照)中の当籤役別テーブル選択相対テーブルが参照される。そして、この1段階目のテーブルデータ取得処理で参照する当籤役別テーブル選択相対テーブルでは、内部当籤役(サブフラグD)の種別毎に設けられた選択値によりCT抽籤の「ハズレ」が設定されているので、抽籤テーブルに「ハズレ」役の抽籤値を規定する必要がなくなる。それゆえ、本実施形態では、CT中CT当籤抽籤テーブルにおいて、「ハズレ」役の抽籤値データを格納する必要がなくなり、メインROM102のテーブル領域の容量を節約することができる。

【1279】

また、CT中CT当籤抽籤テーブル内の2段階目(サブフラグD「リーチ目リブ」取得時)の抽籤テーブルでは、判定ビットを構成する各ビットの値により抽籤対象役又は抽籤対象外の役を判別することができ、抽籤対象外の役に対して抽籤値データ(ハズレデータ)をテーブルに格納する必要がなくなる。さらに、CT中CT当籤抽籤テーブル内では、抽籤対象役の当籤確率が100%である確定データとしては、抽籤値「0」を用いることができる。これらのことから、本実施形態では、CT中CT当籤抽籤テーブル(CT中セット数上乘せ抽籤テーブル)の容量を圧縮することができ、メインROM102において、空き容量を確保する(増大させる)ことができ、増えた空き容量を活用して、遊技性を高めることができる。

10

【1280】

[図柄コード取得処理で得られる効果]

本実施形態のパチスロ1の図柄コード取得処理(図128参照)では、S647~S649で説明した処理手順で入賞に係るデータの圧縮・展開処理を行い、かつ、その処理の中のメインCPU101専用命令コード(「CALLF」命令等)を用いることにより、入賞に係るデータの圧縮・展開処理の効率化を図ることができるとともに、限られたメインRAM103の容量を有効活用することができる。

20

【1281】

また、本実施形態では、図柄コード取得処理中のS649の圧縮データ格納処理において、「CALLF」命令で指定するジャンプ先のアドレスは、図柄設定処理(図97参照)中のS330の圧縮データ格納処理において、「CALLF」命令で指定するジャンプ先のアドレスと同じである。すなわち、本実施形態では、図柄コード取得処理及び図柄設定処理の両処理において、圧縮データ格納処理を実行するためのソースプログラムが共有化(モジュール化)されている。この場合、各処理において、それぞれ別個に圧縮データ格納処理のソースプログラムを設ける必要がなくなるので、その分、ソースプログラムの容量(メインROM102の使用容量)を低減することができる。この結果、本実施形態では、メインROM102において、空き容量を確保する(増大させる)ことができ、増えた空き容量を活用して、遊技性を高めることができる。

30

【1282】

[引込優先順位取得処理で得られる効果]

本実施形態のパチスロ1の引込優先順位取得処理(図134及び図135参照)において、「ANY役」の引込優先対応処理中のS683の判定処理は、ソースプログラム上において、メインCPU101専用命令コードである「JCP」命令(比較命令)により実行される(図136A参照)。

40

【1283】

「ANY役」の引込優先対応処理のソースプログラム上において、「JCP」命令を用いた場合、上述のように、アドレス設定に係る命令を省略することができるので、「ANY役」の引込優先対応処理の処理効率を高めることができるとともに、ソースプログラムの容量(メインROM102の使用容量)を低減することができる。

【1284】

また、引込優先順位取得処理中のS686の停止制御用引込要求フラグ設定処理では、ソースプログラム上において、図136Bに示すように、メインCPU101専用命令コードである、Qレジスタ(拡張レジスタ)を用いてアドレス指定を行う「LDQ」命令、及

50

び、「CALLF」命令が利用される。

【1285】

それゆえ、S686の停止制御用引込要求フラグ設定処理では、「LDQ」命令を用いることにより、ソースプログラム上において、アドレス設定に係る命令を省略することができる。また、「CALLF」命令は、上述のように、2バイトの命令コードである。それゆえ、停止制御用引込要求フラグ設定処理において、これらのメインCPU101専用命令コードを使用することにより、処理の効率化を図ることができ、限られたメインRAM103の容量を有効活用することができる。

【1286】

さらに、本実施形態では、優先引込順位取得処理中のS686の停止制御用引込要求フラグ設定処理において、「CALLF」命令で指定するジャンプ先の論理積演算処理のアドレスは、引込優先順位格納処理(図126参照)中のS626の論理積演算処理において「CALLF」命令で指定するジャンプ先のアドレスと同じである(図127参照)。すなわち、本実施形態では、優先引込順位取得処理及び引込優先順位格納処理の両処理において、論理積演算処理を実行するためのソースプログラムが共有化(モジュール化)されている。

【1287】

この場合、優先引込順位取得処理及び引込優先順位格納処理の両処理において、それぞれ別個に論理積演算処理のソースプログラムを設ける必要が無くなるので、その分、ソースプログラムの容量(メインROM102の使用容量)を低減することができる。この結果、本実施形態では、メインROM102において、空き容量を確保する(増大させる)ことができ、増えた空き容量を活用して、遊技性を高めることができる。

【1288】

また、引込優先順位取得処理中のS687の引込優先順位テーブル(図137参照)の取得処理では、図136Cに示すように、「LDQ」命令(メインCPU101専用命令コード)が利用される。それゆえ、S687の引込優先順位テーブルの取得処理においても、ソースプログラム上において、アドレス設定に係る命令を省略することができる。その結果、引込優先順位テーブルの取得処理の効率化を図ることができるとともに、ソースプログラムの容量(メインROM102の使用容量)を低減することができる。

【1289】

[リール停止制御処理で得られる効果]

本実施形態のパチスロ1のリール停止制御処理(図138参照)において、S711~S715の処理では、図139に示すように、ソースプログラム上において、Qレジスタ(拡張レジスタ)を用いてアドレス指定を行う「LDQ」命令、及び、「CALLF」命令が利用される。

【1290】

それゆえ、本実施形態では、これらのメインCPU101専用命令コードを用いることにより、リール制御処理のソースプログラムの容量を削減するとともに、リール停止制御処理の処理効率を向上させることができる。すなわち、本実施形態では、主制御回路90におけるプログラム処理速度の効率化と容量の削減を行うことが可能となり、削減した容量に応じて増加したメインROM102の空き領域を活用して、遊技性を高めることができる。

【1291】

また、リール停止制御処理中のS726の判定処理(リール(回胴)の停止状態のチェック処理)では、図140に示すように、ソースプログラム上において、「LDQ」命令及び「ORQ」命令(Qレジスタ(拡張レジスタ)を用いてアドレス指定を行うメインCPU101専用命令コード)が用いられる。

【1292】

それゆえ、本実施形態では、リール停止制御処理のソースプログラム上において、アドレ

10

20

30

40

50

ス設定に係る命令を省略することができ、その分、ソースプログラムの容量（メインROM 102の使用容量）を低減することができる。この結果、本実施形態では、メインROM 102において、空き容量を確保する（増大させる）ことができ、増えた空き容量を活用して、遊技性を高めることができる。

【1293】

[入賞検索処理で得られる効果]

本実施形態のパチスロ1の入賞検索処理（図145参照）において、S764の払出枚数及び判定対象データのセット処理では、図146に示すように、ソースプログラム上において、「LDIN」命令が用いられる。

【1294】

それゆえ、本実施形態の入賞検索処理では、一つの「LDIN」命令により、データのロード処理及びアドレスの更新処理の両方を行うことができる。この場合、入賞検索処理のソースプログラム上において、アドレス設定に係る命令を省略することができ、その分、ソースプログラムの容量（メインROM 102の使用容量）を低減することができる。この結果、本実施形態では、メインROM 102において、空き容量を確保する（増大させる）ことができ、増えた空き容量を活用して、遊技性を高めることができる。

【1295】

また、入賞検索処理中のS770の判定処理で参照するメダルカウンタの値の取得処理、S772の処理で参照する入賞枚数カウンタの値の取得処理、及び、S775の処理で行う入賞枚数カウンタの保存（更新）処理では、いずれも、図146に示すように、Qレジスタ（拡張レジスタ）を用いてアドレス指定を行う「LDQ」命令が用いられる。それゆえ、本実施形態の入賞検索処理では、ソースプログラム上において、アドレス設定に係る命令を省略することができ、その分、ソースプログラムの容量（メインROM 102の使用容量）を低減することができる。この結果、本実施形態では、メインROM 102において、空き容量を確保する（増大させる）ことができ、増えた空き容量を活用して、遊技性を高めることができる。

【1296】

さらに、入賞検索処理中のS769の判定処理では、図146に示すように、ソースプログラム上において、「JSLAA」命令が用いられ、S770及びS773の判定処理では、「JCP」命令が用いられる。入賞検索処理のソースプログラム上において、「JSLAA」命令及び「JCP」命令を用いた場合、上述のように、アドレス設定に係る命令を省略することができ、その分、ソースプログラムの容量（メインROM 102の使用容量）を低減することができる。この結果、本実施形態では、メインROM 102において、空き容量を確保する（増大させる）ことができ、増えた空き容量を活用して、遊技性を高めることができる。

【1297】

[イリーガルヒットチェック処理で得られる効果]

本実施形態では、図28～図30に示すように、入賞作動フラグ格納領域（表示役格納領域）の構成が当り要求フラグ格納領域（内部当籤役格納領域）のそれと同じである。それゆえ、本実施形態のイリーガルヒットチェック処理におけるS784の演算処理では、ソースプログラム（図149参照）上において、入賞役のデータと内部当籤役のデータとを単純に論理積（「AND」命令で実行する）するだけで、入賞役のデータと内部当籤役のデータとの合成結果を得ることができる。

【1298】

それゆえ、本実施形態では、イリーガルヒットチェック処理を効率化及び簡略化することができ、その結果、主制御プログラムの空き容量を確保することができ、該空き容量を使用して遊技性を高めることができる。

【1299】

[入賞チェック・メダル払出処理で得られる効果]

本実施形態のパチスロ1の入賞チェック・メダル払出処理（図150参照）では、クレジ

10

20

30

40

50

ットカウンタの更新(+1)後、払出動作を継続する場合、S808の処理において、60.33ms間のウェイト(払出間隔待ち)処理が行われる。この場合、無駄な待ち時間を減らすことができ、遊技者の精神的負担を軽減することができる。

【1300】

[メダル払出枚数チェック処理で得られる効果]

本実施形態のパチスロ1のメダル払出枚数チェック処理(図152参照)中のS814の役連終了枚数カウンタの更新処理では、図153Aに示すように、ソースプログラム上において、メインCPU101専用命令コードである「DCPLD」命令が用いられる。

【1301】

S814の処理において、「DCPLD」命令は、枚数管理カウンタの下限判定命令と、判断分岐命令とが一体になった命令コードでなるので、役連終了枚数カウンタの更新(減算)処理及び連終了枚数カウンタの値を「0」に保持する処理の両方を実行することができる。この場合、両処理を別個に実行するための命令コードを設ける必要がなくなる。それゆえ、本実施形態のメダル払出枚数チェック処理では、ソースプログラムの容量(メインROM102の使用容量)を低減することができ、メインROM102において、空き容量を確保する(増大させる)ことができ、増えた空き容量を活用して、遊技性を高めることができる。

【1302】

また、メダル払出枚数チェック処理中のS816の処理では、図153Bに示すように、クレジット情報コマンドの通信パラメータ1には払出枚数カウンタの値がセットされ、通信パラメータ5にはクレジットカウンタの値がセットされる。しかしながら、クレジット情報コマンドを構成するその他の通信パラメータ2~4(未使用パラメータ)には、現時点においてHレジスタ、Eレジスタ及びDレジスタにそれぞれ格納されている値がセットされる。それゆえ、クレジット情報コマンド送信時における通信パラメータ2~4の値は不定値となる。その結果、本実施形態では、クレジット情報コマンドのサム値(BCC)を送信毎に不定値にすることができ、ゴト等の不正行為を抑制することができる。

【1303】

[7セグLED駆動処理で得られる効果]

本実施形態のパチスロ1の7セグLED駆動処理(図159参照)中のS936で行われる7セグコモン出力(選択)データ及び7セグカソード出力データの出力処理は、図160Bに示すように、一つのソースコード「LD(cPA__SEGC0M),BC」により実行される。すなわち、本実施形態では、7セグLED駆動処理において、2桁の7セグLEDをダイナミック点灯制御する際に、7セグコモン出力データと、7セグカソード出力データとが同時に出力される。この出力制御は、情報表示器6内の指示モニタに押し順表示データを表示する際にも行われる。

【1304】

この場合、7セグLED駆動処理のソースプログラム上において、7セグLEDのダイナミック点灯制御に必要な命令コード数を減らすことができる。それゆえ、本実施形態では、ソースプログラムの容量(メインROM102の使用容量)を低減することができ、メインROM102において、空き容量を確保する(増大させる)ことができ、増えた空き容量を活用して、遊技性を高めることができる。

【1305】

[タイマー更新処理で得られる効果]

本実施形態のパチスロ1のタイマー更新処理(図164参照)中のS952の処理(2バイトタイマーの更新処理)では、図165に示すように、ソースプログラム上において、メインCPU101専用命令コードである「DCPWD」命令が用いられる。

【1306】

タイマー更新処理において、「DCPWD」命令を実行した場合、上述のように、タイマー値(2バイトタイマー値)の更新(減算)処理及びタイマー値を「0」に保持する処理の両方を実行することができる。この場合、両処理を別個に実行するための命令コード

10

20

30

40

50

を設ける必要がなくなる。それゆえ、本実施形態では、ソースプログラムの容量（メインROM102の使用容量）を低減することができ、メインROM102において、空き容量を確保する（増大させる）ことができ、増えた空き容量を活用して、遊技性を高めることができる。

【1307】

<各種変形例>

以上、本発明に係る遊技機の構成及び動作について、その作用効果も含めて説明した。しかしながら、本発明は、上述した実施形態に限定されるものではなく、特許請求の範囲に記載した本発明の要旨を逸脱しない限り、その他の種々の実施形態及び変形例が含まれる。

【1308】

[変形例1：通常ART中のCT前兆遊技と報知抽籤]

上記実施形態のパチスロ1では、理解を容易にするために、遊技者にとって有利な状態（例えば、CT）に当籤した場合に、遊技状態を次遊技（次ゲーム）から有利な状態に移行させる例を説明したが、本発明はこれに限定されない。例えば、遊技者に対して有利な状態の遊技制御を行う場合に、いわゆる「前兆遊技」などと呼ばれている、所定回数の遊技を行った後に当該有利な状態の遊技制御を行ってもよい。

【1309】

ここでは、図174A及び174Bを参照して、その一例として、遊技状態が、通常ARTから所定回数の前兆遊技を介してCTに移行する例を説明する。なお、図174Aは、変形例1におけるCT抽籤当籤時の遊技フローを示す図であり、図174Bは、前兆遊技中に行われるフラグ変換抽籤で用いられるフラグ変換抽籤テーブルの構成図である。

【1310】

この例の通常ART中の遊技では、まず、図174Aに示すように、上記実施形態と同様に、ART中CT抽籤テーブル（図50参照）を用い、内部当籤役（サブフラグ）に基づいてCT抽籤を行う。このCT抽籤に当籤すると、遊技状態が、CT（上乘せチャンスゾーン）という遊技者にとって有利な状態に移行することが決定されるので、CTに当籤するまでの期間において、CT抽籤のための各種抽籤は、メイン側（主制御基板71）で行われる。例えば、主制御基板71（メインCPU101）は、内部当籤役を決めるための内部抽籤を行うとともに、内部当籤役として内部当籤役「F__確チリリプ」、「F__1確チリリプ」及び「F__リーチ目リプA」～「F__リーチ目リプD」のいずれかが決定された場合には、ART中フラグ変換抽籤テーブル（図47A及び47B参照）を用いてフラグ変換抽籤を行う。

【1311】

次いで、通常ART中の遊技においてCT抽籤に当籤すると、この例では、CT移行前に所定期間の前兆遊技（CT前兆遊技）が行われる。なお、この例では、CT前兆遊技において、例えば、フラグ変換抽籤により決定されたサブフラグEX（「3連チリリプ」、「リーチ目リプ」、「リプレイ」等）に基づくCT抽籤などの遊技者にとって特典を付与するような抽籤を行わないこととする。しかしながら、この例では、CT前兆遊技において、例えば、図174Bに示すフラグ変換抽籤テーブルを用いたフラグ変換抽籤をサブ側（副制御基板72側）で行い、このサブ側で行ったフラグ変換抽籤の結果に基づいて、サブ側で行う報知内容が制御（決定）される。例えば、サブ側で行ったフラグ変換抽籤に当籤した場合には、略称「リーチ目リプ」に係る図柄組合せを表示するための情報を表示装置11（プロジェクト機構211及び表示ユニット212）で報知し、フラグ変換抽籤に非当籤であった場合には、略称「リプレイ」に係る図柄組合せを表示するための情報を表示装置11で報知する。

【1312】

上述のように近年のパチスロでは、遊技者の利益（出玉）に影響を与える抽籤をメイン側で行うことが求められているが、この例のパチスロ1では、フラグ変換抽籤の抽籤結果が遊技者の利益に何ら影響を与えない期間（CT前兆遊技の期間）を設け、この期間に限り、サブ側でフラグ変換抽籤を行う。それゆえ、この例では、例えば、CT前兆中という特

10

20

30

40

50

典の付与が決まっている状況において、例えば、略称「リーチ目リップ」に係る図柄組合せ等の特別な図柄組合せを表示する機会を増やすことができるとともに、図柄組合せの見せ方のバリエーションも増加させることができる。その結果、この例の構成によれば、遊技性をより向上させることが可能になる。

【1313】

なお、図174Bに示す抽籤テーブルは、サブ側で行うフラグ変換抽籤に用いられるCT前兆中のフラグ変換抽籤テーブルであり、ロムカートリッジ基板86に記憶されている。CT前兆中のフラグ変換抽籤テーブルは、内部当籤役「F__リーチ目リップA」～「F__リーチ目リップD」と、サブ側で行うフラグ変換抽籤の抽籤結果（非当籤/当籤）と、各抽籤結果に対応付けられた抽籤値の情報との対応関係を規定する。

10

【1314】

図174Bに示すフラグ変換抽籤テーブルから明らかなように、この例では、CT前兆遊技において、内部当籤役「F__リーチ目リップA」～「F__リーチ目リップD」が非常に高い確率でサブフラグEX「リーチ目リップ」に変換される（フラグ変換抽籤に当籤する）。すなわち、CT前兆遊技において、内部当籤役「F__リーチ目リップA」～「F__リーチ目リップD」のいずれかが決定されると、高い確率で略称「リーチ目リップ」の図柄組合せが停止表示されることになり、遊技者に対して、現在の遊技がCT前兆遊技中であることを示唆することができる。この際、上述のように、CT前兆遊技においてサブ側で行うフラグ変換抽籤に当籤した場合、略称「リーチ目リップ」に係る図柄組合せを表示するための報知は行われるものの、特典を付与することはない。

20

【1315】

なお、この例において、CT前兆遊技中のフラグ変換抽籤をメイン側で行ってもよい。しかしながら、図174A及び174Bで説明した例のように、遊技者の利益に何ら影響を与えない期間における抽籤をサブ側で行うことにより、メイン側のデータ容量や処理負荷を軽減することができる。

【1316】

[変形例2：3連チリリップ表示用の押し順の別例]

上記実施形態のパチスロ1では、内部当籤役「F__確チリリップ」又は「F__1確チリリップ」が決定された場合、ストップボタンの押し順が正解であると、略称「3連チリリップ」に係る図柄組合せが表示され、押し順が不正解であると、略称「リプレイ」に係る図柄組合せが表示される例を説明した（図24参照）。また、上記実施形態では、内部当籤役「F__確チリリップ」及び「F__1確チリリップ」に対応付けられた正解の押し順が「順押し」である例を説明した。しかしながら、本発明はこれに限定されない。

30

【1317】

例えば、内部当籤役「F__確チリリップ」又は「F__1確チリリップ」が決定された場合における、押し順が不正解である場合に表示される図柄組合せの種類を増やすとともに、内部当籤役「F__確チリリップ」が決定された際の正解の押し順を、内部当籤役「F__1確チリリップ」が決定された場合のそれと異ならせてもよい。その一例（変形例2）を、図175A及び175Bを参照して説明する。なお、図175Aは、変形例2における、内部当籤役と停止表示される図柄組合せ（停止図柄（略称））との対応関係を示す図であり、図175Bは、変形例2における、通常ART中のフラグ変換抽籤の結果とサブ側で行われるナビ種別との対応関係を示す図である。

40

【1318】

この例では、図175Aに示すように、内部当籤役「F__確チリリップ」が決定された場合における、正解の押し順は「順押し（左リール3Lを第1停止）」とし、不正解の押し順は「中リール3Cを第1停止（以下「中押し」という）又は右リール3Rを第1停止（以下、「逆押し」という）」、すなわち、「変則押し」とする。そして、この例では、内部当籤役「F__確チリリップ」が決定された場合に、順押しされると、略称「3連チリリップ」に係る図柄組合せが表示され、中押しされると、略称「リプレイ」に係る図柄組合せが表示され、逆押しされると、略称「2連チリリップ」に係る図柄組合せが表示される。

50

【 1 3 1 9 】

また、この例では、内部当籤役「F__1確チリリブ」が決定された場合における、正解の押し順は逆押しとし、不正解の押し順は順押し及び中押しとする。そして、この例では、内部当籤役「F__1確チリリブ」が決定された場合に、逆押しされると、略称「3連チリリブ」に係る図柄組合せが表示され、中押しされると、略称「リプレイ」に係る図柄組合せが表示され、順押しされると、略称「2連チリリブ」に係る図柄組合せが表示される。

【 1 3 2 0 】

この例のパチスロ1では、内部当籤役「F__確チリリブ」及び「F__1確チリリブ」のいずれかが決定された場合、順押しすると、内部当籤役の種類に応じて、略称「3連チリリブ」又は「2連チリリブ」に係る図柄組合せが表示される。また、内部当籤役「F__確チリリブ」及び「F__1確チリリブ」のいずれかが決定された場合に、逆押しすると、内部当籤役の種類に応じて、略称「3連チリリブ」又は「2連チリリブ」に係る図柄組合せが表示される。さらに、この例では、内部当籤役「F__確チリリブ」及び「F__1確チリリブ」のいずれかが決定された場合に、中押しすると、内部当籤役の種類に関係なく、略称「リプレイ」に係る図柄組合せが表示される。

【 1 3 2 1 】

内部当籤役と停止表示される図柄組合せ（停止図柄（略称））との対応関係を図175Aに示す関係に設定すると、例えば、内部当籤役「F__確チリリブ」又は「F__1確チリリブ」が決定された場合に、遊技者にチリ図柄を狙わせるための様々なナビ（報知）を実施することが可能になる。

【 1 3 2 2 】

例えば、「順押しでチリ図柄を狙わせるナビ」と「逆押しでチリ図柄を狙わせるナビ」との双方を行うことができる。なお、内部当籤役「F__確チリリブ」が決定されている場合には、「順押しでチリ図柄を狙わせるナビ」は略称「3連チリリブ」に係る図柄組合せを表示させるためのナビとなり、「逆押しでチリ図柄を狙わせるナビ」は略称「3連チリリブ」に係る図柄組合せを表示させないためのナビ（略称「2連チリリブ」に係る図柄組合せが表示されるナビ）となる。一方、内部当籤役「F__1確チリリブ」が決定されている場合には、「順押しでチリ図柄を狙わせるナビ」は略称「3連チリリブ」に係る図柄組合せを表示させないためのナビ（略称「2連チリリブ」に係る図柄組合せが表示されるナビ）となり、「逆押しでチリ図柄を狙わせるナビ」は略称「3連チリリブ」に係る図柄組合せを表示させるためのナビとなる。

【 1 3 2 3 】

この例において、上述したナビを行うか否かの決定は、メイン側で行うフラグ変換抽籤により管理され、このメイン側のフラグ変換抽籤の結果に基づいて、サブ側の制御によりナビが実行される。この際、通常ART中にメイン側で行われるフラグ変換抽籤の抽籤結果と、サブ側で制御されるナビ種別との対応関係は、図175Bに示す対応関係となる。

【 1 3 2 4 】

この例においても、通常ART中の遊技において内部当籤役「F__確チリリブ」又は「F__1確チリリブ」が決定されると、主制御基板71（メインCPU101）は、2段階のフラグ変換抽籤を行う（図47参照）。一方、サブ側では、この2段階のフラグ変換抽籤の結果を、スタートコマンドデータから取得し、2段階のフラグ変換抽籤の結果に基づいて、表示装置11で行うナビを決定する。

【 1 3 2 5 】

そこで、この例において、1段階目のフラグ変換抽籤の時点で抽籤結果が非当籤であった場合には、副制御基板72（サブCPU201）は、図175Bに示すように、「リプレイナビ」と称するナビを行う。なお、「リプレイナビ」では、遊技者に対して中押しするように指示する情報が報知される。この例では、図174Aに示すように、内部当籤役が「F__確チリリブ」及び「F__1確チリリブ」のいずれであっても、中押し時には略称「リプレイ」に係る図柄組合せが表示されることになる。

【 1 3 2 6 】

10

20

30

40

50

また、この例において、1段階目のフラグ変換抽籤に当籤し、かつ、2段階目のフラグ変換抽籤の抽籤結果が非当籤である場合には、副制御基板72（サブCPU201）は、図175Bに示すように、「チリリブ煽りナビ」と称するナビを行う。なお、「チリリブ煽りナビ」では、内部当籤役「F__確チリリブ」が決定されている場合、遊技者に対して逆押しでチリ図柄を狙わせるような指示情報が報知される。一方、「チリリブ煽りナビ」では、内部当籤役「F__1確チリリブ」が決定されている場合、遊技者に対して順押しでチリ図柄を狙わせるような指示情報が報知される。そして、このような「チリリブ煽りナビ」では、図174Aに示すように、内部当籤が「F__確チリリブ」である場合には、逆押し時に略称「2連チリリブ」に係る図柄組合せが表示され、内部当籤役が「F__1確チリリブ」である場合には、順押し時に略称「2連チリリブ」に係る図柄組合せが表示されることになる。

10

【1327】

また、この例において、1段階目及び2段階目のフラグ変換抽籤の両方に当籤した場合には、副制御基板72（サブCPU201）は、図175Bに示すように、「チリリブ揃いナビ」と称するナビを行う。なお、「チリリブ揃いナビ」では、内部当籤役「F__確チリリブ」が決定されている場合、遊技者に対して順押しでチリ図柄を狙わせるような指示情報が報知される。一方、「チリリブ揃いナビ」では、内部当籤役「F__1確チリリブ」が決定されている場合、遊技者に対して逆押しでチリ図柄を狙わせるような指示情報が報知される。そして、このような「チリリブ揃いナビ」では、図174Aに示すように、内部当籤が「F__確チリリブ」である場合には、順押し時に略称「3連チリリブ」に係る図柄組合せが表示され、内部当籤役が「F__1確チリリブ」である場合には、逆押し時に略称「3連チリリブ」に係る図柄組合せが表示されることになる。

20

【1328】

この例における、上述したフラグ変換抽籤の抽籤結果に基づく報知は、利益に影響を与えるものではない（フラグ変換抽籤自体は利益に影響を与えるものの、結果として表示される図柄組合せは利益に影響を与えない）ので、この例の上記報知動作は、メイン側（指示モニタ）では行わず、サブ側（表示装置11）でのみ行う。また、この例では、上述のように、「チリリブ揃いナビ」だけでなく、「チリリブ煽りナビ」も併せて行うことで、利益に影響しないナビを多様な態様でサブ側により制御することができる。その結果、遊技の興趣を向上させることができる。

30

【1329】

[変形例3：フラグ間中及び非フラグ間中のベルナビ態様の別例]

上記実施形態のパチスロ1では、上述のように、リールの停止操作の情報と一義的に対応する数値を、指示モニタ（不図示）に表示することにより、メイン側での報知を行う。この際、図63A～63Dで説明したナビデータの対応関係が参照される。そして、上記実施形態では、BBフラグ間状態（RT5状態）中に「白7ナビ」や「青7ナビ」を行うが、「ベルナビ」などを行わない例を説明したが、本発明はこれに限定されない。BBフラグ間状態（RT5）中に、「白7ナビ」や「青7ナビ」以外に「ベルナビ」などを行う構成にしてもよい。

【1330】

この場合、指示モニタに表示する数値は、BBフラグ間状態中と非BBフラグ間状態中とで互いに異ならせることとしてもよい。ここで、図176A及び176Bに、変形例3における、BBフラグ間状態中にメイン側で行う報知（ナビ）とサブ側で行う報知（ナビ）との対応関係を示す。なお、図176Aは、RT5状態中（BB1フラグ間）における、メイン側で行う報知（ナビ）とサブ側で行う報知（ナビ）との対応関係を示す図であり、図176Bは、RT5状態中（BB2フラグ間）における、メイン側で行う報知（ナビ）とサブ側で行う報知（ナビ）との対応関係を示す図である。

40

【1331】

また、この例において、非BBフラグ間状態中にメイン側で行う報知（ナビ）とサブ側で行う報知（ナビ）との対応関係は、上記実施形態と同様である（図63A及び63B参照

50

）。それゆえ、非 B B フラグ間状態において「ベルナビ」が行われる場合、指示モニタには「1」～「3」の数値（押し順役第 2 指示情報）が表示される。

【 1 3 3 2 】

この例では、図 1 7 6 A 及び 1 7 6 B に示すように、B B フラグ間状態において「ベルナビ」が行われる場合、指示モニタに「1 2」～「1 4」の数値（押し順役第 1 指示情報）が表示される。なお、本発明はこれに限定されず、B B フラグ間状態において「ベルナビ」が行われる場合に、指示モニタに表示される数値は適宜変更することができる。なお、この例において、表示装置 1 1 を用いて行うサブ側のナビは、B B フラグ間状態及び非 B B フラグ間状態の両方において共通のものが設けられていてもよい。

【 1 3 3 3 】

[変形例 4 : 特典付与の別例]

上記実施形態のパチスロ 1 では、メイン側で行われるフラグ変換抽籤の結果に基づいて報知内容が制御されるので、報知に従い停止操作を行った場合には表示される図柄組合せが異なる。すなわち、上記実施形態では、メイン側で行われるフラグ変換抽籤の結果に基づいて特典を付与するか否かが決定される例を説明したが、本発明はこれに限定されない。例えば、メイン側（主制御基板 7 1）において、実際に表示された図柄組合せに基づいて特典を付与する構成にしてもよい。

【 1 3 3 4 】

この場合、主制御基板 7 1（メイン CPU 1 0 1）は、フラグ変換抽籤の結果に応じて行われる報知に従って所定の図柄組合せが表示された場合には、特典を付与し、報知に従わなかった場合には、所定の図柄組合せが表示されても特典を付与しない構成にしてもよい。上記実施形態のパチスロ 1 では、押し順に応じて表示される図柄組合せが異なるが、遊技者が報知を無視して停止操作を行ってしまった場合にも、略称「3 連チリリブ」に係る図柄組合せ等の特別な図柄組合せが表示されてしまう可能性がある。それゆえ、この例では、報知を無視して特別な図柄組合せが表示されたとしても、特典を付与することなく、報知に従って特別な図柄組合せが表示された場合に限り、特典を付与するようにしてもよい。

【 1 3 3 5 】

[変形例 5 : 利益（特典）に影響を与えない報知制御の別例]

上記変形例 1（図 1 7 4 A 及び 1 7 4 B 参照）のパチスロ 1 では、利益に影響を与える報知を行うか否かは、メイン側での報知抽籤（フラグ変換抽籤）により決定され、利益に影響を与えない報知を行うか否かは、サブ側での報知抽籤で決定される例を説明した。そして、この利益に影響を与えない報知の一例として、前兆遊技中に略称「リーチ目リブ」に係る図柄組合せを表示するための報知を行う例を説明したが、利益に影響を与えない報知は、この例に限定されるものではない。

【 1 3 3 6 】

近年のパチスロでは、停止操作の順序に応じて遊技ロックを行い易い（難しい）状態に遷移させることがある。例えば、「左、中、右」の押し順である場合に遊技ロックを行い易い状態に遷移し、「右、中、左」の押し順である場合に遊技ロックを行い難い状態に遷移することがある。このようなパチスロでは、押し順を報知することで、遊技ロックを行い易い（難しい）状態に遷移させることができる。しかしながら、遊技ロックを行うか否かが利益に対して影響を与える場合には、この報知をメイン側で制御する必要がある。なお、遊技ロックを行うか否かが利益に対して影響を与えない場合には、この報知をサブ側で制御してもよい。

【 1 3 3 7 】

また、遊技ロックを行うか否かが利益に対して影響を与える場合としては、例えば、遊技ロックが行われることで、ART 抽籤に当籤するといった場合が考えられる。一方、遊技ロックを行うか否かが利益に対して影響を与えない場合は、遊技ロックを演出として行う場合である。例えば、利益（特典）を付与することが決まっている前兆遊技中に遊技ロックを頻繁に行うようにすることにより、この後の遊技において利益が付与されることを演

10

20

30

40

50

出で示すことが可能になる。

【 1 3 3 8 】

ここで、図 1 7 7 A 及び 1 1 7 B を参照して、変形例 5 における、押し順とロック状態とを対応付ける構成例を説明する。なお、図 1 7 7 A は、押し順とロック状態との対応関係を示す図であり、図 1 7 7 B は、遊技ロックによる利益への影響の有無と、報知態様との関係を示す図である。

【 1 3 3 9 】

図 1 7 7 A に示す例では、内部当籤役「F__維持リブA」が決定された時に「左、中、右」の押し順で停止操作が行われると、ロック状態が「0」（ロックし難い状態）から「1」（ロックし易い状態）に遷移し、「左、右、中」、「中、左、右」又は「中、右、左」の押し順で停止操作が行われると、現在のロック状態が維持され、「右、左、中」又は「右、中、左」の押し順で停止操作が行われると、ロック状態が「1」から「0」に遷移する。また、内部当籤役「F__維持リブB」が決定された時には、「中、左、右」の押し順で停止操作が行われると、ロック状態が「0」（ロックし難い状態）から「1」（ロックし易い状態）に遷移し、それ以外の押し順で停止操作が行われると、現在のロック状態が維持される。

10

【 1 3 4 0 】

なお、図 1 1 7 A に示す例では、説明を単純にするために、ロック状態を「0（ロックし難い状態）」及び「1（ロックし易い状態）」の2段階にする例を説明するが、本発明はこれに限定されない。例えば、3段階以上のロック状態を設けてもよい。また、この場合には、ロック状態が1段階ずつ遷移するのではなく、多段階を1回で遷移する構成にしてもよい。

20

【 1 3 4 1 】

そして、この例では、図 1 1 7 B に示すように、遊技ロックが利益に影響を与える場合には、報知を行うか否かの報知抽籤をメイン（主制御基板 7 1）側で行うとともに、その抽籤結果に基づいてメイン側及びサブ側の両方で所定の押し順を報知する。一方、遊技ロックが利益に影響を与えない場合には、報知を行うか否かの報知抽籤をサブ（副制御基板 7 2）側で行うとともに、その抽籤結果に基づいてサブ側で所定の押し順を報知する。

【 1 3 4 2 】

なお、パチスロとしては、遊技の全期間にわたり遊技ロックが利益に影響を与えるパチスロもあれば、遊技の全期間にわたり遊技ロックが利益に影響を与えないパチスロもある。また、パチスロとしては、遊技の所定期間では遊技ロックが利益に影響を与えるが、遊技の特定期間では遊技ロックが利益に影響を与えないパチスロもある。それゆえ、遊技ロックが利益に影響を与える期間では、報知を行うか否かの報知抽籤をメイン側で行い、その抽籤結果に基づいてメイン側及びサブ側の両方で所定の押し順を報知し、一方、遊技ロックが利益に影響を与えない期間では、報知を行うか否かの報知抽籤をサブ側で行い、その抽籤結果に基づいてサブ側で所定の押し順を報知するようにしてもよい。

30

【 1 3 4 3 】

なお、遊技ロックの制御は、通常、メイン側（主制御基板 7 1）で行われるので、図 1 1 7 A に示す押し順に応じてロック状態を遷移する機能は、主制御基板 7 1 に設けられる。すなわち、主制御基板 7 1 は、検出した停止操作の順序に基づいてロック状態を設定するロック状態設定手段、ロック状態設定手段が設定したロック状態に応じた確率で遊技ロックを行うか否かを抽籤により決定する遊技ロック判定手段、及び、遊技ロック判定手段が遊技ロックを行うと判定すると、遊技の進行を一時的に停止するロック実行手段、としても機能する。また、主制御基板 7 1 及び/又は副制御基板 7 2 は、遊技ロックし易い（し難い）ロック状態をロック状態設定手段が設定する際に、押し順を報知するか否かの報知抽籤を行う報知抽籤手段、及び、報知抽籤手段の抽籤結果に基づいて所定の押し順を報知する報知手段としても機能する。

40

【 1 3 4 4 】

[変形例 6 : 通常 A R T や C T の終了条件の別例]

50

通常ARTやCTの終了条件は、上記実施形態で説明した例に限定されず、任意の終了条件を採用することができる。例えば、通常ART中やCT中に付与されたメダルの枚数、通常ART中やCT中の単位遊技の消化回数、通常ART中やCT中に行われた遊技者にとって有利な情報の報知の回数、所定ゲーム数（例えば50ゲーム）を1セットとする場合のセット数、1セット終了時の継続率等の終了条件を採用することができる。

【1345】

また、通常ART中やCT中に付与されたメダルの枚数の計数手法としては、例えば、単位遊技において払い出されたメダルの枚数を計数する手法を採用してもよいし、単位遊技において払い出されたメダルの枚数から当該単位遊技に用いられたメダルのベット（掛け）枚数を減算した差枚数（純増枚数）を計数する手法を採用してもよい。また、通常ART中やCT中に付与されたメダルの枚数の計数手法としては、実際に増加したメダルに基づき算出する手法（実値による算出）を採用してもよいし、実際に増加したか否かに関わらず、報知に従った場合に増加する予定のメダルの枚数に基づき算出する手法（理想値による算出）を採用してもよい。

10

【1346】

また、内部当籤役の種別によって、付与されたメダルの枚数を増加しない構成、すなわち、付与されたメダルの枚数の終了条件となるメダルの枚数又は差枚数にはカウントしない構成を採用してもよい。例えば、内部当籤役として決定される確率が低い一部の役（レア役）や、停止操作のタイミングに応じて図柄組合せの表示／非表示が切り替わる役などが内部当籤役として決定されたとしても、付与されたメダルの枚数の増減（カウント）を行わないようにしてもよい。

20

【1347】

[変形例7：その他]

通常ART中やCT中に行われる報知の内容は上述した例に限定されず、任意である。例えば、遊技者にとって有利な状態となる特別な図柄組合せが表示される停止操作の順序（押し順）を報知してもよいし、当該図柄組合せが表示されるために必要な停止操作のタイミング（狙うべき図柄）を報知するようにしてもよい。

【1348】

遊技者にとって有利な状態としては、再遊技に係る内部当籤役の当籤確率は変化しない（又は遊技性に影響を与えない程度の範囲で変化する）が、遊技者にとって有利な停止操作の態様を報知する機能、すなわち、AT機能が作動する遊技状態であってもよい。また、遊技者にとって有利な状態としては、再遊技に係る内部当籤役の当籤確率が高くなる再遊技高確率状態（リプレイタイム）が作動するとともに、遊技者にとって有利な停止操作の態様を報知する機能が作動する、すなわち、ART機能が作動する遊技状態であってもよい。

30

【1349】

また、上記実施形態のパチスロ1では、遊技者側から見て、リール表示窓4の左側に設けたサブ表示装置18において各種表示画面を表示する例を説明したが、本発明はこれに限定されない。例えば、遊技者側から見て、リール表示窓4の右側にも別のサブ表示装置を設け、このサブ表示装置においても各種表示画面を表示する構成にしてもよい。この場合、リール表示窓4の右側に設けられたサブ表示装置の表示面上にタッチセンサを設け、このタッチセンサから出力されるタッチ入力情報に基づいて、当該サブ表示装置の表示画面を切り替える構成にしてもよい。

40

【1350】

また、上記実施形態及び各種変形例では、遊技機としてパチスロを例に挙げて説明したが、本発明はこれに限定されない。本発明のリール制御に係る特徴や設定変更及び確認に係る特徴などのパチスロ1に特有の特徴以外の特徴は、「パチンコ」と呼ばれる遊技機にも適用可能であり、同様の効果が得られる。例えば、チェックサムの生成及び判定処理、メインCPU101専用命令コードを使用した各種処理（Qレジスタを用いたアドレスの指定処理、ソフトタイマーの更新処理、7セグLED駆動処理、通信データの生成格納処理

50

等)、規定外ROM領域及び規定外RAM領域を使用した各種処理などの特徴は、「パチンコ」にも適用可能である。

【1351】

<主制御基板及び副制御基板が有する各種機能>

以上、本発明に係るパチンコ1の実施形態及び各種変形例について説明した。ここで、本発明に係るパチンコ1の主制御基板71(主制御回路90,メインCPU101)及び副制御基板72(副制御回路200,サブCPU201)が有する各種機能をまとめて説明する。

【1352】

主制御基板71は、スタートスイッチ79やストップスイッチ基板80と接続され、図1に示す遊技の進行を制御する。それゆえ、主制御基板72は、開始操作検出手段、図柄変動手段、内部当籤役決定手段、停止操作検出手段、リール停止制御手段(停止制御手段)及び入賞判定手段として機能する。

10

【1353】

また、主制御基板71は、通常ART中に内部当籤役「F__確チリリブ」又は「F__1確チリリブ」が決定されると、フラグ変換抽籤を行い、このフラグ変換抽籤の結果やその他の内部当籤役に基づくCT抽籤に当籤すると、通常ARTの継続期間を延長するCTを開始する。それゆえ、主制御基板71は、変換抽籤手段及び上乗せ遊技開始手段としても機能する。

【1354】

また、主制御基板71は、CT中に内部当籤役に応じて通常ARTの継続期間を示すARTゲーム数を上乗せ(延長)する。具体的には、主制御基板71は、サブフラグ「サポテン」、「弱チェリー」又は「強チェリー」に対応する内部当籤役が決定された場合に、通常ARTのARTゲーム数を所定量上乗せし、サブフラグ「3連チリリブ(3連チリリブA又は3連チリリブB)」に対応する内部当籤役が決定された場合に、通常ARTのARTゲーム数を特定量上乗せする。さらに、主制御基板71は、サブフラグ「3連チリリブ」に基づくARTゲーム数の上乗せが行われた回数が、CTの1セットの基本ゲーム数である8回を超える9回以上になると、上乗せ1回あたりの上乗せ量を増加する。それゆえ、主制御基板71は、上乗せ制御手段としても機能する。

20

【1355】

また、主制御基板71は、CT中にARTゲーム数が上乗せされない遊技の回数をCTゲーム数カウンタを用いて計数し、CTゲーム数カウンタが「8」を計数すると、CTを終了する。このとき、サブフラグEX「3連チリリブ」が当籤(すなわち、フラグ変換抽籤に当籤)すると、主制御基板71は、1セット8回(8ゲーム)のCT遊技を再セットする(CTゲーム数カウンタの値を初期値に戻す)。それゆえ、主制御基板71は、計数手段及び上乗せ遊技終了手段としても機能する。

30

【1356】

また、主制御基板71は、ボーナス役(内部当籤役「F__BB1」、「F__BB2」)が内部当籤役として決定されると、遊技状態をBBフラグ間状態(RT5状態)に移行させるとともに、BBフラグ間状態ではボーナスが作動するまで(ボーナス役が入賞するまで)ボーナス役を内部当籤役として持ち越す。それゆえ、主制御基板71は、ボーナス持ち越手段としても機能する。また、主制御基板71は、BBフラグ間状態においてボーナス役が入賞すると、ボーナスを作動させ、遊技状態をボーナス状態に移行させる。それゆえ、主制御基板71は、ボーナス開始手段及び有利遊技手段としても機能する。

40

【1357】

また、主制御基板71は、図25に示すように、BBフラグ間状態において、ボーナス役と所定の内部当籤役(「はずれ」、「F__特殊1」~「F__特殊3」)とが重複して決定されている場合には、ボーナス役に係る図柄組合せ(「C__BB1」、「C__BB2」)を表示可能となるように停止制御を行い、ボーナス役と所定の内部当籤役以外の内部当籤役とが重複して決定されている場合には、ボーナス役に係る図柄組合せを表示不可能と

50

なるように停止制御を行う。そして、ＢＢフラグ間状態中にボーナス役に係る図柄組合せが表示可能な場合には、主制御基板 7 1 は、情報表示器 6 の指示モニタ（不図示）を制御して、ボーナス役を入賞させるための停止操作の態様を一義的に示す数値「10」又は「11」の表示してボーナス指示情報を報知する。一方、ＢＢフラグ間状態中にボーナス役に係る図柄組合せが表示不可能な場合には、主制御基板 7 1 は、指示モニタに数値「10」及び「11」を表示（報知）しない。それゆえ、主制御基板 7 1 及び情報表示器 6 の指示モニタは、指示情報報知手段として機能する。

【1358】

また、この際、主制御基板 7 1 は、ボーナス告知をした後に限り指示モニタを介して数値「10」又は「11」を報知する。具体的には、主制御基板 7 1 は、ボーナス役を持ち越していない状態でボーナス役を内部当籤役として決定すると、その後の遊技の回数を計数して、当該計数結果が所定回数になった後にボーナス役と所定の内部当籤役（「はずれ」、「F__特殊1」～「F__特殊3」）とが重複して決定されると、指示モニタを介して数値「10」又は「11」を報知する。それゆえ、主制御基板 7 1 は、ボーナス役を内部当籤役として決定してからの単位遊技の回数を計数する計数手段としても機能する。

10

【1359】

副制御基板 7 2 は、主制御基板 7 1 から受信した各種コマンドデータに基づいて遊技履歴を管理するとともに、遊技者からの登録操作を受け付けると、遊技を行う遊技者の登録を受け付ける。それゆえ、副制御基板 7 2 は、履歴管理手段及び登録受付手段として機能する。

20

【1360】

また、本発明に係るパチスロ 1 は、遊技の進行に応じた演出を行う表示装置 1 1 と、表示装置 1 1 とは別個に設けられ、トップ画面 2 2 1 や遊技情報画面 2 2 3 , 2 2 4 , 2 2 5 などを含む複数の表示画面を表示するサブ表示装置 1 8 と、サブ表示装置 1 8 の表示部上に設けられたタッチセンサ 1 9 とを有し、副制御基板 7 2 は、表示装置 1 1 及びサブ表示装置 1 8 の動作を制御する。具体的には、副制御基板 7 2 は、遊技者の登録を受け付けている場合には、サブ表示装置 1 8 に遊技情報画面 2 2 3 , 2 2 4 , 2 2 5 を表示可能となるようにサブ表示装置 1 8 を制御し、遊技者の登録を受け付けていない場合には、サブ表示装置 1 8 に遊技情報画面 2 2 3 , 2 2 4 , 2 2 5 を表示不可能とするよう制御する。それゆえ、副制御基板 7 2 は、制御手段としても機能する。

30

【1361】

また、主制御基板 7 1 は、遊技の結果に応じて様々な特典を付与するため、特典付与手段としても機能する。具体的には、主制御基板 7 1 は、有効ラインに沿って表示された図柄組合せに応じて特典を付与する。

【1362】

例えば、内部当籤役「F__3 択ベル__1 s t」が決定された遊技において、略称「ベル」に係る図柄組合せが有効ライン上に表示されると、主制御基板 7 1 は、9 枚のメダルを付与し、略称「ベルこぼし目」に係る図柄組合せが有効ライン上に表示されると、主制御基板 7 1 は、0 枚のメダルを付与する。また、例えば、内部当籤役「F__確チリリプ」が決定された遊技において、略称「3 連チリリプ」又は略称「リプレイ」に係る図柄組合せが有効ライン上に表示されると、主制御基板 7 1 は、再遊技の作動という同一の特典を付与する。また、主制御基板 7 1 は、有効ラインに沿って表示された図柄組合せではなく、フラグ変換抽籤の結果に基づいても特典を付与する。例えば、主制御基板 7 1 は、内部当籤役「F__確チリリプ」が決定されるとフラグ変換抽籤を行い、フラグ変換抽籤に当籤すると、ＣＴ抽籤に当籤するなどの特典を付与する。

40

【1363】

また、副制御基板 7 2 は、表示装置 1 1 を介して停止操作の態様に応じた演出を実行する。それゆえ、副制御基板 7 2 及び表示装置 1 1 は、演出実行手段としても機能する。この際、有効ラインに沿って表示された図柄組合せによって付与する特典が異なる場合には、主制御基板 7 1 は、指示モニタを介して遊技者にとって有利な停止操作の態様を一義的に

50

示す情報を報知し、有効ラインに沿って表示された図柄組合せによって付与する特典が同一である場合には、遊技者にとって有利な停止操作の態様を報知しない。これに対して、副制御基板 7 2 は、表示された図柄組合せによって付与する特典が同一 / 異なるかに関係なく、停止操作の順序を示す演出を実行する。

【 1 3 6 4 】

また、主制御基板 7 1 は、C T を開始するか否かの C T 抽籤を行い、C T 抽籤に当籤した場合には、当籤してから例えば所定回数の前兆遊技が行われた後に C T を開始する。それゆえ、主制御基板 7 1 は、有利状態抽籤手段及び有利状態開始手段としても機能する。さらに、この際、C T 前兆遊技中に内部当籤役「F_確チリリップ」又は「F_1 確チリリップ」が決定されると、副制御基板 7 2 は、サブ側でのフラグ変換抽籤を行う。それゆえ、副制御基板 7 2 は、報知抽籤手段としても機能する。

10

【 1 3 6 5 】

また、本発明に係るパチスロ 1 では、フラグ変換抽籤の結果に基づいて、主制御基板 7 1 は特典を付与するとともに、主制御基板 7 1 及び副制御基板 7 2 は、指示モニタ及び表示装置 1 1 を介して押し順を報知する。それゆえ、主制御基板 7 1 は、特典付与手段としても機能する。また、主制御基板 7 1、副制御基板 7 2、指示モニタ及び表示装置 1 1 は、報知手段としても機能する。

【 1 3 6 6 】

さらに、本発明に係るパチスロ 1 では、主制御基板 7 1 (主制御回路 9 0、メイン CPU 1 0 1) は、遊技動作全般に渡って、各種制御処理を行う。それゆえ、主制御基板 7 1 は演算制御手段としても機能する。また、本発明に係るパチスロ 1 では、主制御基板 7 1 (主制御回路 9 0、メイン CPU 1 0 1) は、以下に示す各種処理の実行手段としても機能する。

20

【 1 3 6 7 】

主制御基板 7 1 (主制御回路 9 0、メイン CPU 1 0 1) は、電断発生時のチェックサム生成処理 (図 7 7 参照) 及び電源復帰時のサムチェック処理 (図 7 9 及び図 8 0 参照) を行う。それゆえ、主制御基板 7 1 はサム値算出手段、サム値減算手段及びサム値判定手段としても機能する。

【 1 3 6 8 】

主制御基板 7 1 (主制御回路 9 0、メイン CPU 1 0 1) は、入賞検索処理 (図 1 4 5 参照) を行う。それゆえ、主制御基板 7 1 は、特典付与判定手段及び入賞役決定手段としても機能する。

30

【 1 3 6 9 】

主制御基板 7 1 (主制御回路 9 0、メイン CPU 1 0 1) は、通信データ送信処理 (図 1 5 8 の割込処理中の S 9 0 4) を行う。それゆえ、主制御基板 7 1 は、データ送信手段としても機能する。

【 1 3 7 0 】

主制御基板 7 1 (主制御回路 9 0、メイン CPU 1 0 1) は、設定変更確認処理 (図 6 8 参照) を行う。それゆえ、主制御基板 7 1 は設定変更確認手段、開始時コマンド生成手段及び終了時コマンド生成手段としても機能する。

40

【 1 3 7 1 】

主制御基板 7 1 (主制御回路 9 0、メイン CPU 1 0 1) は、通信データ格納処理 (図 7 2 参照) 及び通信データポイント更新処理 (図 7 4 参照) を行う。それゆえ、主制御基板 7 1 は、通信データ生成手段及び通信データ生成格納手段としても機能する。

【 1 3 7 2 】

主制御基板 7 1 (主制御回路 9 0、メイン CPU 1 0 1) は、7 セグ LED 駆動処理 (図 1 5 9 参照) を行う。それゆえ、主制御基板 7 1 は、7 セグ LED 駆動手段、LED 駆動制御手段としても機能する。

【 1 3 7 3 】

主制御基板 7 1 (主制御回路 9 0、メイン CPU 1 0 1) は、遊技復帰処理 (図 6 8 参照

50

）を行う。それゆえ、主制御基板 7 1 は、遊技復帰手段としても機能する。

【 1 3 7 4 】

主制御基板 7 1 (主制御回路 9 0、メイン CPU 1 0 1) は、メダル受付・スタートチェック処理 (図 8 3 参照) を行う。それゆえ、主制御基板 7 1 は、遊技開始判定手段及び設定確認手段 (S 2 3 3) としても機能する。

【 1 3 7 5 】

主制御基板 7 1 (主制御回路 9 0、メイン CPU 1 0 1) は、メダル投入チェック処理 (図 8 7 参照) を行う。それゆえ、主制御基板 7 1 は、遊技媒体受付状態判別手段 (S 2 5 5 ~ S 2 5 8) としても機能する。

【 1 3 7 6 】

主制御基板 7 1 (主制御回路 9 0、メイン CPU 1 0 1) は、1 . 1 1 7 2 m s e c 周期で割込処理 (図 1 5 8 参照) を繰り返し実行する。それゆえ、主制御基板 7 1 は、割込処理実行手段、定周期処理手段としても機能する。

【 1 3 7 7 】

主制御基板 7 1 (主制御回路 9 0、メイン CPU 1 0 1) は、電源投入時処理 (図 6 4 参照) を行う。それゆえ、主制御基板 7 1 は、電源復帰処理実行手段としても機能する。

【 1 3 7 8 】

主制御基板 7 1 (主制御回路 9 0、メイン CPU 1 0 1) は、内部抽籤処理 (図 9 2 参照) を行う。それゆえ、主制御基板 7 1 は、内部抽籤手段としても機能する。

【 1 3 7 9 】

主制御基板 7 1 (主制御回路 9 0、メイン CPU 1 0 1) は、図柄設定処理 (図 9 7 参照) を行う。それゆえ、主制御基板 7 1 は、内部当籤役生成手段 (S 3 2 1)、フラグテーブル展開手段、当籤フラグテーブル展開手段 (S 3 2 4)、フラグ格納領域指定手段、当籤フラグ格納領域指定手段 (S 3 2 9) 及びフラグデータ格納手段、当籤フラグデータ格納手段 (S 3 3 0) としても機能する。

【 1 3 8 0 】

主制御基板 7 1 (主制御回路 9 0、メイン CPU 1 0 1) は、図柄コード取得処理 (図 2 8 参照) を行う。それゆえ、主制御基板 7 1 は、入賞フラグ格納領域指定手段 (S 6 4 8) 及び図柄コード格納領域設定手段 (S 6 5 0) としても機能する。

【 1 3 8 1 】

主制御基板 7 1 (主制御回路 9 0、メイン CPU 1 0 1) は、リール停止制御処理 (図 1 3 9 参照) を行う。それゆえ、主制御基板 7 1 は、停止操作検出結果取得手段 (S 7 1 4) 及び停止制御データ格納領域設定手段 (図 1 3 9 中のソースコード「LDQ IX, w R 1 _ _ C T R L - (w R 2 _ _ C T R L - w R 1 _ _ C T R L)」) としても機能する。

【 1 3 8 2 】

主制御基板 7 1 (主制御回路 9 0、メイン CPU 1 0 1) は、引込優先順位取得処理 (図 1 3 4 及び図 1 3 5 参照) を行う。それゆえ、主制御基板 7 1 は、優先停止図柄決定手段、任意役対応処理手段 (S 6 8 0 ~ S 6 8 3)、当籤フラグ格納領域指定手段 (S 6 8 6)、入賞フラグ格納領域指定手段 (S 6 8 6)、論理積演算手段 (S 6 8 6) 及び優先順位データテーブル取得手段 (S 6 8 7) としても機能する。

【 1 3 8 3 】

主制御基板 7 1 (主制御回路 9 0、メイン CPU 1 0 1) は、イリーガルヒットチェック処理 (図 1 4 8 参照) を行う。それゆえ、主制御基板 7 1 は、エラー検出手段、エラー処理手段、入賞フラグ格納領域指定手段 (S 7 8 1)、論理積演算手段 (S 7 8 4) 及びエラー判定手段 (S 7 8 5) としても機能する。

【 1 3 8 4 】

主制御基板 7 1 (主制御回路 9 0、メイン CPU 1 0 1) は、入賞チェック・メダル払出処理 (図 1 5 0 参照) を行う。それゆえ、主制御基板 7 1 は、遊技媒体払出手段、遊技媒体加算手段 (S 8 0 5)、払出終了判定手段 (S 8 0 7) 及びウェイト発生手段 (S 8 0 8) としても機能する。

10

20

30

40

50

【 1 3 8 5 】

主制御基板 7 1 (主制御回路 9 0、メイン CPU 1 0 1) は、CT 中 CT 抽籤処理 (図 1 1 9 参照) を行う。それゆえ、主制御基板 7 1 は、特典付与決定手段としても機能する。

【 1 3 8 6 】

主制御基板 7 1 (主制御回路 9 0、メイン CPU 1 0 1) は、サブフラグ変換処理 (図 1 0 5 参照) を行う。それゆえ、主制御基板 7 1 は、第 1 サブフラグ変換手段としても機能する。

【 1 3 8 7 】

また、主制御基板 7 1 (主制御回路 9 0、メイン CPU 1 0 1) は、フラグ変換処理 (図 1 1 1 参照) を行う。それゆえ、主制御基板 7 1 は、第 2 サブフラグ変換手段としても機能する。

10

【 1 3 8 8 】

< 付記 (本発明のまとめ) >

[第 1 の遊技機]

従来、上述した構成の遊技機において、電断時に RAM に記憶されているデータのチェックサムを求め、電源復帰時に、電断時に求めたチェックサムの判定処理を行う遊技機が知られている (例えば、特開 2 0 0 9 - 0 1 1 3 7 5 号公報参照)。特開 2 0 0 9 - 0 1 1 3 7 5 号公報の遊技機では、電源復帰時のチェックサムの判定処理において、電源復帰時に求めたチェックサムが電断時に求めたチェックサムが一致しない場合にエラー報知が行われる。

20

【 1 3 8 9 】

ところで、従来、上述した遊技機特有の制限として、主制御回路のプログラム容量が、規則により小容量に制限されている。さらに、近年、遊技性の複雑化により主制御回路の ROM の容量が圧迫されており、主制御回路で管理する遊技性以外の処理プログラムやテーブルなどの容量削減が求められている。

【 1 3 9 0 】

本発明は、上記第 1 の課題を解決するためになされたものであり、本発明の第 1 の目的は、主制御回路で管理する遊技性以外の処理プログラムやテーブルなどの容量を削減して主制御回路の ROM の空き容量を増やし、該増えた容量分の ROM の空き領域を利用して遊技性を高めることが可能な遊技機を提供することである。

30

【 1 3 9 1 】

上記第 1 の課題を解決するために、本発明では、以下のような構成の第 1 の遊技機を提供する。

【 1 3 9 2 】

遊技動作を制御するための演算処理を行う演算処理手段 (例えば、メイン CPU 1 0 1) と、

前記演算処理手段による前記演算処理の実行に必要な情報が記憶された第 1 記憶手段 (例えば、メイン ROM 1 0 2) と、

前記演算処理手段による前記演算処理の実行に必要な情報が記憶される第 2 記憶手段 (例えば、メイン RAM 1 0 3) と、

40

電源電圧を供給する電源供給手段 (例えば、電源基板 5 3 b 及びスイッチングレギュレータ 9 4) と、

前記電源電圧が予め定められた起動電圧値 (例えば、1 0 V) を上回った場合に、前記演算処理手段に起動信号を出力する起動手段 (例えば、電源管理回路 9 3 のリセット信号の出力処理) と、

前記電源電圧が予め定められた停電電圧値 (例えば、1 0 . 5 V) を下回った場合に、前記演算処理手段に停電信号を出力する停電手段 (例えば、電源管理回路 9 3 の電断検知信号の出力処理) と、を備え、

前記演算処理手段は、

演算処理の結果に対応するデータを格納するフラグレジスタ (例えば、フラグ・レジスタ

50

F)と、

前記停電手段が前記停電信号を出力したことを契機として、前記第2記憶手段内の所定格納領域(例えば、遊技用RAM領域)に記憶された全ての情報を累積加算してサム値を算出するサム値算出手段(例えば、チェックサム生成処理)と、

前記起動手段が前記起動手信号を出力したことを契機として、直近の電断発生時に前記サム値算出手段により生成された前記サム値から、前記所定格納領域に記憶された情報を順次減算するサム値減算手段(例えば、サムチェック処理中のS122~S131)と、

前記所定格納領域に記憶された全ての情報に対して、前記サム値減算手段による減算処理が終了したときに、前記フラグレジスタ内の所定のビット領域(例えば、ゼロフラグ)にセットされた減算結果に対応するデータに基づいて、異常の発生の有無を判定するサム値判定手段(例えば、サムチェック処理中のS134)と、を有する

10

ことを特徴とする遊技機。

【1393】

また、前記本発明の第1の遊技機では、前記サム値算出手段は、前記所定格納領域に記憶された情報を加算するときに、特定の命令(例えば、POP命令)を実行することにより、連続して記憶された2バイト分の情報を取得して加算するとともに、前記情報の読み出し開始アドレスの情報を2バイト分更新し、

前記サム値減算手段は、電断発生時に生成された前記サム値から前記所定格納領域に記憶された情報を減算するときに、前記特定の命令を実行することにより、連続して記憶された2バイト分の情報を取得して減算するとともに、前記情報の読み出し開始アドレスの情報を2バイト分更新するようにしてもよい。

20

【1394】

さらに、前記本発明の第1の遊技機では、前記演算処理手段は、前記第2記憶手段の前記所定格納領域のアドレスを設定可能なスタックポインタを有し、

前記サム値算出手段が前記所定格納領域に記憶された情報を加算するときに実行する前記特定の命令は、前記スタックポインタを操作するための専用命令(例えば、POP命令)であるようにしてもよい。

【1395】

上記構成の本発明の第1の遊技機によれば、遊技性以外の処理プログラムやテーブルなどの容量を削減して主制御回路のROM(第1記憶手段)の空き容量を増やし、該増えた容量分のROMの空き領域を利用して遊技性を高めることができる。

30

【1396】

[第2~第5の遊技機]

従来、上述した構成の遊技機において、スタックポインタを操作命令で使用して、数値データを処理する主制御装置が搭載された遊技機が提案されている(例えば、特開2005-237737号公報参照)。

【1397】

ところで、従来、上述した遊技機特有の制限として、主制御回路のプログラム容量が、規則により小容量に制限されている。さらに、近年、遊技性の複雑化により主制御回路のROMの容量が圧迫されており、主制御回路で管理する処理プログラムやテーブルなどの容量削減が求められている。

40

【1398】

本発明は、上記第2の課題を解決するためになされたものであり、本発明の第2の目的は、主制御回路で管理する処理プログラムやテーブルなどの容量を削減して主制御回路のROMの空き容量を増やし、該増えた容量分のROMの空き領域を利用して遊技性を高めることが可能な遊技機を提供することである。

【1399】

上記第2の課題を解決するために、本発明では、以下のような構成の第2の遊技機を提供する。

【1400】

50

遊技動作を制御するための演算処理を行う演算処理手段（例えば、メインCPU101）と、

前記演算処理手段による前記演算処理の実行に必要な情報が記憶された第1記憶手段（例えば、メインROM102）と、

前記演算処理手段による前記演算処理の実行に必要な情報が記憶される第2記憶手段（例えば、メインRAM103）と、を備え、

前記演算処理手段は、

前記演算処理手段による前記演算処理の実行時にデータが格納される汎用レジスタと、

前記演算処理手段による前記演算処理の実行時にデータが格納される専用レジスタと、

前記演算処理手段による前記演算処理の実行時にデータが格納され、該格納されたデータにより前記第1記憶手段又は前記第2記憶手段内のアドレスの一部を指定可能な拡張レジスタ（例えば、Qレジスタ）と、を有し、

前記演算処理手段は、前記拡張レジスタを用いて前記第2記憶手段内のアドレス指定を行うことができる所定の命令（例えば、「BITQ」命令、「SETQ」命令、「LDQ」命令等）を実行可能である

ことを特徴とする遊技機。

【1401】

また、前記本発明の第2の遊技機では、前記拡張レジスタで指定可能な前記アドレスの一部が、前記アドレスを構成する上位側のアドレス値であるようにしてもよい。

【1402】

また、上記第2の課題を解決するために、本発明では、以下のような構成の第3の遊技機を提供する。

【1403】

遊技媒体の投入操作を検出する投入操作検出手段（例えば、BETスイッチ77）と、

前記投入操作検出手段による投入操作の検出された後、遊技者による開始操作を検出する開始操作検出手段（例えば、スタートスイッチ79）と、

前記開始操作検出手段による開始操作の検出に基づいて予め定められた確率で内部当籤役を決定する内部当籤役決定手段（例えば、内部抽籤処理）と、

複数の表示列を含み、前記開始操作検出手段による開始操作の検出に基づいて、各表示列に設けられた図柄を変動表示する変動表示手段（例えば、3つのリール3L, 3C, 3R）と、

遊技者による停止操作の検出を行う停止操作検出手段（例えば、ストップスイッチ基板80）と、

前記内部当籤役決定手段の決定結果と前記停止操作検出手段による停止操作の検出とに基づいて、前記図柄の変動表示を停止させる停止制御手段（例えば、リール停止制御処理）と、

前記停止制御手段による前記表示列の停止動作を制御するための演算処理を行う演算処理手段（例えば、メインCPU101）と、

前記演算処理手段による前記演算処理の実行に必要な情報が記憶された第1記憶手段（例えば、メインROM102）と、

前記演算処理手段による前記演算処理の実行に必要な情報が記憶される第2記憶手段（例えば、メインRAM103）と、を備え、

前記演算処理手段は、

前記演算処理手段による前記演算処理の実行時にデータが格納される汎用レジスタと、

前記演算処理手段による前記演算処理の実行時にデータが格納される専用レジスタと、

前記演算処理手段による前記演算処理の実行時にデータが格納され、該格納されたデータにより前記第1記憶手段又は前記第2記憶手段内のアドレスの一部を指定可能な拡張レジスタ（例えば、Qレジスタ）と、を有し、

前記演算処理手段は、

前記表示列の停止状態をチェックする処理において、

10

20

30

40

50

前記拡張レジスタを用いて前記第2記憶手段内のアドレス指定を行うことができる所定の読み出し命令（例えば、「LDQ」命令）を実行して、前記第2記憶手段に記憶されている所定の前記表示列の変動表示の状態を示す情報を読み出し、
 次に、前記拡張レジスタを用いて前記第2記憶手段内のアドレス指定を行うことができる所定の論理和演算命令（例えば、「ORQ」命令）を実行して、前記第2記憶手段に記憶されている他の一つの表示列の変動表示の状態を示す情報を読み出すとともに、当該情報と、前記所定の前記表示列の変動表示の状態を示す情報との論理和演算を行い、
 その後、前記所定の論理和演算命令の実行を繰り返して、論理和演算の結果と、残りの各表示列の変動表示の状態を示す情報との論理和演算を繰り返し、全ての表示列に対する論理和演算が終了した際に得られる演算和演算の結果に基づいて、全ての表示列が停止状態にあるか否かを判定することを特徴とする遊技機。

10

【1404】

また、前記本発明の第3の遊技機では、前記拡張レジスタで指定可能な前記アドレスの一部が、前記アドレスを構成する上位側のアドレス値であるようにしてもよい。

【1405】

また、上記第2の課題を解決するために、本発明では、以下のような構成の第4の遊技機を提供する。

【1406】

遊技媒体の投入操作を検出する投入操作検出手段（例えば、BETスイッチ77）と、
 前記投入操作検出手段による投入操作の検出された後、遊技者による開始操作を検出する開始操作検出手段（例えば、スタートスイッチ79）と、
 前記開始操作検出手段による開始操作の検出に基づいて予め定められた確率で内部当籤役を決定する内部当籤役決定手段（例えば、内部抽籤処理）と、
 複数の表示列を含み、前記開始操作検出手段による開始操作の検出に基づいて、各表示列に設けられた図柄を変動表示する変動表示手段（例えば、3つのリール3L, 3C, 3R）と、

20

遊技者による停止操作の検出を行う停止操作検出手段（例えば、ストップスイッチ基板80）と、

前記内部当籤役決定手段の決定結果と前記停止操作検出手段による停止操作の検出とに基づいて、前記図柄の変動表示を停止させる停止制御手段（例えば、リール停止制御処理）と、

30

前記内部当籤役決定手段による前記内部当籤役の決定動作を制御するための演算処理を行う演算処理手段（例えば、メインCPU101）と、

前記演算処理手段による前記演算処理の実行に必要な情報が記憶された第1記憶手段（例えば、メインROM102）と、

前記演算処理手段による前記演算処理の実行に必要な情報が記憶される第2記憶手段（例えば、メインRAM103）と、を備え、

前記演算処理手段は、

前記演算処理手段による前記演算処理の実行時にデータが格納される汎用レジスタと、

40

前記演算処理手段による前記演算処理の実行時にデータが格納される専用レジスタと、

前記演算処理手段による前記演算処理の実行時にデータが格納され、該格納されたデータにより前記第1記憶手段又は前記第2記憶手段内のアドレスの一部を指定可能な拡張レジスタ（例えば、Qレジスタ）と、を有し、

前記内部当籤役決定手段により決定される前記内部当籤役には、特典付与に係る内部当籤役が決定される期待値を調整するための設定値に応じて当籤確率が変化する設定別内部当籤役が設けられ、

前記演算処理手段は、

前記内部当籤役を決定する処理において、

前記拡張レジスタを用いて前記第2記憶手段内のアドレス指定を行うことができる所定の

50

加算命令（例えば、「ADDQ」命令）を実行することにより、抽籤対象となる前記設定別内部当籤役の設定値毎の抽籤値が格納された領域の先頭アドレスに現在の設定値を加算して、現在の設定値に対応付けられた前記設定別内部当籤役の抽籤値が格納されたアドレスを指定し、当該抽籤値を取得することを特徴とする遊技機。

【1407】

また、前記本発明の第4の遊技機では、前記拡張レジスタで指定可能な前記アドレスの一部が、前記アドレスを構成する上位側のアドレス値であるようにしてもよい。

【1408】

また、上記第2の課題を解決するために、本発明では、以下のような構成の第5の遊技機を提供する。

【1409】

遊技媒体の投入操作を検出する投入操作検出手段（例えば、BETスイッチ77）と、前記投入操作検出手段による投入操作の検出された後、遊技者による開始操作を検出する開始操作検出手段（例えば、スタートスイッチ79）と、前記開始操作検出手段による開始操作の検出に基づいて予め定められた確率で内部当籤役を決定する内部当籤役決定手段（例えば、内部抽籤処理）と、複数の表示列を含み、前記開始操作検出手段による開始操作の検出に基づいて、各表示列に設けられた図柄を変動表示する変動表示手段（例えば、3つのリール3L, 3C, 3R）と、

遊技者による停止操作の検出を行う停止操作検出手段（例えば、ストップスイッチ基板80）と、

前記内部当籤役決定手段の決定結果と前記停止操作検出手段による停止操作の検出とに基づいて、前記図柄の変動表示を停止させる停止制御手段（例えば、リール停止制御処理）と、

前記停止制御手段により前記複数の表示列の変動表示が停止された場合に、前記複数の表示列に跨って設定された判定ライン上に、前記遊技媒体の払い出しに係る内部当籤役に対応する図柄の組合せが停止表示されたか否かを判定する特典付与判定手段（例えば、入賞検索処理）と、

前記特典付与判定手段による判定動作を制御するための演算処理を行う演算処理手段（例えば、メインCPU101）と、

前記演算処理手段による前記演算処理の実行に必要な情報が記憶された第1記憶手段（例えば、メインROM102）と、

前記演算処理手段による前記演算処理の実行に必要な情報が記憶される第2記憶手段（例えば、メインRAM103）と、を備え、

前記演算処理手段は、

前記演算処理手段による前記演算処理の実行時にデータが格納される汎用レジスタを有し、前記演算処理手段は、

前記遊技媒体の払い出しに係る内部当籤役に対応する図柄の組合せが前記判定ライン上に停止表示されたか否かを判定する処理において、

所定の読み出し命令（例えば、「LDIN」命令）を実行して、前記遊技媒体の払い出しに係る内部当籤役に対応する図柄の組合せが前記判定ライン上に停止表示された場合に付与され得る前記遊技媒体の払出数のデータと、該払出数のデータに対応付けられた、前記遊技媒体の払い出しに係る内部当籤役に対応する図柄の組合せの種別を示す判定データとを同時に取得する

ことを特徴とする遊技機。

【1410】

また、前記本発明の第5の遊技機では、前記演算処理手段は、

前記遊技媒体の払い出しに係る内部当籤役に対応する図柄の組合せが前記判定ライン上に停止表示されたか否かを判定する処理において、

10

20

30

40

50

判定命令、処理のジャンプ先アドレスの指定命令及び処理のジャンプ動作命令を一つの命令で実行可能な所定の判定命令（例えば、「J S L A A」命令）を実行して、前記遊技媒体の払い出しに係る内部当籤役に対応する図柄の組合せが停止表示されたか否かを判定するようにしてもよい。

【 1 4 1 1 】

上記構成の本発明の第 2 ～ 第 5 の遊技機によれば、主制御回路で管理する処理プログラムやテーブルなどの容量を削減して主制御回路の R O M（第 1 記憶手段）の空き容量を増やし、該増えた容量分の R O M の空き領域を利用して遊技性を高めることができる。

【 1 4 1 2 】

[第 6 及び第 7 の遊技機]

従来、上述した構成の遊技機において、メイン制御部の R A M に記憶されているデータに異常が生じた場合に、R A M 異常エラー状態に制御され、ゲームの進行が不能化されるとともに、設定変更モードに移行し、設定変更操作に基づいて設定値が新たに選択・設定されたときには、ゲームの進行の不能化状態を解除し、ゲームの進行が可能な状態にする遊技機が提案されている（例えば、特開 2 0 0 7 - 2 0 9 8 1 0 号公報参照）。

【 1 4 1 3 】

ところで、従来、上述した遊技機特有の制限として、主制御回路のプログラム容量が、規則により小容量に制限されている。さらに、近年、遊技性の複雑化により主制御回路の R O M の容量が圧迫されており、主制御回路で管理する処理プログラムやテーブルなどの容量削減が求められている。

【 1 4 1 4 】

本発明は、上記第 3 の課題を解決するためになされたものであり、本発明の第 3 の目的は、主制御回路で管理する処理プログラムやテーブルなどの容量を削減して主制御回路の R O M の空き容量を増やし、該増えた容量分の R O M の空き領域を利用して遊技性を高めることが可能な遊技機を提供することである。

【 1 4 1 5 】

上記第 3 の課題を解決するために、本発明では、以下のような構成の第 6 の遊技機を提供する。

【 1 4 1 6 】

遊技媒体の投入操作を検出する投入操作検出手段（例えば、B E T スイッチ 7 7 ）と、
前記投入操作検出手段による投入操作の検出された後、遊技者による開始操作を検出する開始操作検出手段（例えば、スタートスイッチ 7 9 ）と、
前記開始操作検出手段による開始操作の検出に基づいて予め定められた確率で内部当籤役を決定する内部当籤役決定手段（例えば、内部抽籤処理）と、
複数の表示列を含み、前記開始操作検出手段による開始操作の検出に基づいて、各表示列に設けられた図柄を変動表示する変動表示手段（例えば、3 つのリール 3 L , 3 C , 3 R ）と、
遊技者による停止操作の検出を行う停止操作検出手段（例えば、ストップスイッチ基板 8 0 ）と、
前記内部当籤役決定手段の決定結果と前記停止操作検出手段による停止操作の検出とに基づいて、前記図柄の変動表示を停止させる停止制御手段（例えば、リール停止制御処理）と、
前記遊技動作に関するコマンドデータを送信するデータ送信手段（例えば、割込処理中の S 9 0 4（通信データ送信処理））と、
特典付与に係る内部当籤役が決定される期待値を調整するための設定値の変更処理及び設定値の確認処理を実行可能な設定変更確認手段（例えば、設定変更確認処理）と、
前記設定変更確認手段による設定値の変更動作又は確認動作を制御するための演算処理を行う演算処理手段（例えば、メイン C P U 1 0 1 ）と、
前記演算処理手段による前記演算処理の実行に必要な情報が記憶された第 1 記憶手段（例えば、メイン R O M 1 0 2 ）と、

10

20

30

40

50

前記演算処理手段による前記演算処理の実行に必要な情報が記憶される第2記憶手段（例えば、メインRAM103）と、を備え、

前記演算処理手段は、

前記設定値の変更処理又は設定値の確認処理の開始時において、第1のコマンドデータを生成する開始時コマンド生成手段（例えば、設定変更確認処理中のS43）と、

前記設定値の変更処理又は設定値の確認処理の終了時において、第2のコマンドデータを生成する終了時コマンド生成手段（例えば、設定変更確認処理中のS57）と、を有し、

前記開始時コマンド生成手段により実行される前記第1のコマンドデータの生成処理と、前記終了時コマンド生成手段により実行される前記第2のコマンドデータの生成処理とは、共有化されている

10

ことを特徴とする遊技機。

【1417】

また、前記本発明の第6の遊技機では、前記演算処理手段は、前記演算処理手段による前記演算処理の実行時にデータが格納される汎用レジスタを有し、

前記演算処理手段は、

前記第1のコマンドデータを生成する場合には、前記汎用レジスタの所定のレジスタ（例えば、Lレジスタ）に第1の値（例えば、「005H」）を設定して、前記生成処理を実行し、

前記第2のコマンドデータを生成する場合には、前記汎用レジスタの所定のレジスタに第2の値（例えば、「004H」）を設定して、前記生成処理を実行するようにしてもよい。

20

【1418】

また、上記第3の課題を解決するために、本発明では、以下のような構成の第7の遊技機を提供する。

【1419】

遊技媒体の投入操作を検出する投入操作検出手段（例えば、BETスイッチ77）と、

前記投入操作検出手段による投入操作の検出された後、遊技者による開始操作を検出する開始操作検出手段（例えば、スタートスイッチ79）と、

前記開始操作検出手段による開始操作の検出に基づいて予め定められた確率で内部当籤役を決定する内部当籤役決定手段（例えば、内部抽籤処理）と、

複数の表示列を含み、前記開始操作検出手段による開始操作の検出に基づいて、各表示列に設けられた図柄を変動表示する変動表示手段（例えば、3つのリール3L, 3C, 3R）と、

30

遊技者による停止操作の検出を行う停止操作検出手段（例えば、ストップスイッチ基板80）と、

前記内部当籤役決定手段の決定結果と前記停止操作検出手段による停止操作の検出とに基づいて、前記図柄の変動表示を停止させる停止制御手段（例えば、リール停止制御処理）と、

前記遊技動作に関するコマンドデータを送信するデータ送信手段（例えば、割込処理中のS904（通信データ送信処理））と、

前記コマンドデータを作成する通信データ生成手段（例えば、設定変更コマンド生成格納処理及び通信データ格納処理）と、

40

特典付与に係る内部当籤役が決定される期待値を調整するための設定値の変更処理及び設定値の確認処理を実行可能な設定変更確認手段（例えば、設定変更確認処理）と、

前記通信データ生成手段によるコマンドデータの生成動作、及び、前記設定変更確認手段による設定値の変更動作又は確認動作を制御するための演算処理を行う演算処理手段（例えば、メインCPU101）と、

前記演算処理手段による前記演算処理の実行に必要な情報が記憶された第1記憶手段（例えば、メインROM102）と、

前記演算処理手段による前記演算処理の実行に必要な情報が記憶される第2記憶手段（例えば、メインRAM103）と、を備え、

50

前記演算処理手段は、
 前記設定値の変更処理又は設定値の確認処理の開始時において、開始時コマンドデータを生成する開始時コマンド生成手段（例えば、設定変更確認処理中のS43）と、
 前記設定値の変更処理又は設定値の確認処理の終了時において、終了時コマンドデータを生成する終了時コマンド生成手段（例えば、設定変更確認処理中のS57）と、を有し、
 前記開始時コマンド生成手段により実行される前記開始時コマンドデータの生成処理と、
 前記終了時コマンド生成手段により実行される前記終了時コマンドデータの生成処理とは、
 共有化されており、

前記演算処理手段は、前記演算処理手段による前記演算処理の実行時に複数種のデータがそれぞれ格納される複数の汎用レジスタを有し、

10

前記演算処理手段は、
 前記コマンドデータの生成処理において、
 前記コマンドデータを構成する複数種の通信パラメータのうち、使用される通信パラメータを、前記複数の汎用レジスタのうちの対応する汎用レジスタにセットし、
 前記汎用レジスタにセットされた使用される通信パラメータを、前記第2記憶手段内の所定の格納領域（例えば、通信データ一時格納領域）に格納し、
 前記複数種の通信パラメータのうち、使用されない通信パラメータがある場合には、前記コマンドデータの生成時に当該未使用の通信パラメータに対応付けられた汎用レジスタに格納されているデータを通信パラメータとして前記所定の格納領域に格納し、
 通信パラメータに対応付けられた前記汎用レジスタに格納されたデータに基づいて、前記コマンドデータのサム値を生成することを特徴とする遊技機。

20

【1420】

また、前記本発明の第7の遊技機において、前記コマンドデータの生成処理では、前記汎用レジスタのアクュームレータに格納された値に、通信パラメータに対応付けられた前記汎用レジスタに格納された値を加算することにより、前記コマンドデータのサム値を生成し、該生成したサム値を前記所定の格納領域に格納するようにしてもよい。

【1421】

上記構成の本発明の第6及び第7の遊技機によれば、主制御回路で管理する処理プログラムやテーブルなどの容量を削減して主制御回路のROMの空き容量を増やし、該増えた容量分のROMの空き領域を利用して遊技性を高めることができる。

30

【1422】

[第8及び第9の遊技機]

従来、上述した構成の遊技機において、遊技制御基板（主制御回路）から演出制御基板（副制御回路）にコマンドを送信する遊技機が知られている（例えば、特開2002-360766号公報参照）。

【1423】

ところで、従来、上述した遊技機特有の制限として、主制御回路のプログラム容量が、規則により小容量に制限されている。さらに、近年、遊技性の複雑化により主制御回路のROMの容量が圧迫されており、主制御回路で管理する処理プログラムやテーブルなどの容量削減が求められている。

40

【1424】

本発明は、上記第4の課題を解決するためになされたものであり、本発明の第4の目的は、主制御回路で管理する処理プログラムやテーブルなどの容量を削減して主制御回路のROMの空き容量を増やし、該増えた容量分のROMの空き領域を利用して遊技性を高めることが可能な遊技機を提供することである。

【1425】

上記第4の課題を解決するために、本発明では、以下のような構成の第8の遊技機を提供する。

【1426】

50

遊技動作を制御するための演算処理を行う演算処理手段（例えば、メインCPU101）と、

前記演算処理手段による前記演算処理の実行に必要な情報が記憶された第1記憶手段（例えば、メインROM102）と、

前記演算処理手段による前記演算処理の実行に必要な情報が記憶される第2記憶手段（例えば、メインRAM103）と、

遊技動作に関する通信データを送信するデータ送信手段（例えば、割込処理中のS904（通信データ送信処理））と、

前記通信データを作成して、該生成した通信データを前記第2記憶手段内の所定のアドレス範囲に設けられた通信データ格納領域に格納する通信データ生成格納手段（例えば、通信データ格納処理及び通信データポインタ更新処理）と、を備え、

10

前記通信データ生成格納手段は、

前記所定のアドレス範囲の先頭アドレスの格納領域から最後尾アドレスの格納領域に向かって順次、前記通信データを前記通信データ格納領域に格納する際に、更新命令、上限判定命令及び判断分岐命令を一つの命令で実行可能な所定の更新命令（例えば、「ICPLD」命令）を実行することにより、前記通信データ格納領域内のアドレス指定に関するパラメータである通信データポインタの現在の値と、前記最後尾アドレスに対応する前記通信データポインタの上限値とを比較するとともに、現在の前記通信データポインタが前記上限値未満であれば、前記通信データポインタを加算更新し、現在の前記通信データポインタが前記上限値以上であれば、前記通信データポインタを前記先頭アドレスに対応する前記通信データポインタの下限値に変更する

20

ことを特徴とする遊技機。

【1427】

また、前記本発明の第8の遊技機では、前記通信データ生成格納手段は、前記通信データポインタを前記先頭アドレスに対応する前記通信データポインタの下限値に変更した場合に、前記通信データ格納領域に格納された前記通信データを無効にするようにしてもよい。

【1428】

また、上記第4の課題を解決するために、本発明では、以下のような構成の第9の遊技機を提供する。

【1429】

30

遊技動作を制御するための演算処理を行う演算処理手段（例えば、メインCPU101）と、

前記演算処理手段による前記演算処理の実行に必要な情報が記憶された第1記憶手段（例えば、メインROM102）と、

前記演算処理手段による前記演算処理の実行に必要な情報が記憶される第2記憶手段（例えば、メインRAM103）と、を備え、

前記演算処理手段は、

遊技動作の進行に関連する所定のデータの値の計数処理（例えば、メダル払出枚数チェック処理）において、

前記所定のデータの値を更新する際に、更新命令、下限判定命令及び判断分岐命令を一つの命令で実行可能な所定の更新命令（例えば、「DCPLD」命令）を実行することにより、現在の前記所定のデータの値と前記所定のデータの値の下限値とを比較するとともに、現在の前記所定のデータの値が前記所定のデータの値の下限値より大きければ、前記所定のデータの値を減算更新し、現在の前記所定のデータの値が前記所定のデータの値の下限値以下であれば、前記所定のデータの値を前記下限値に保持する

40

ことを特徴とする遊技機。

【1430】

また、前記本発明の第9の遊技機では、前記所定のデータが、前記遊技媒体の払出数であるようにしてもよい。

【1431】

50

上記構成の本発明の第 8 及び第 9 の遊技機によれば、主制御回路で管理する処理プログラムやテーブルなどの容量を削減して主制御回路の ROM（第 1 記憶手段）の空き容量を増やし、該増えた容量分の ROM の空き領域を利用して遊技性を高めることができる。

【 1 4 3 2 】

[第 1 0 の遊技機]

従来、上述した構成の遊技機において、ソフトウェアによるタイマー減算処理で制御される遊技機が知られている（例えば、特開 2 0 0 4 - 0 4 1 2 6 1 号公報参照）。

【 1 4 3 3 】

ところで、従来、上述した遊技機特有の制限として、主制御回路のプログラム容量が、規則により小容量に制限されている。さらに、近年、遊技性の複雑化により主制御回路の ROM の容量が圧迫されており、主制御回路で管理する処理プログラムやテーブルなどの容量削減が求められている。

10

【 1 4 3 4 】

本発明は、上記第 5 の課題を解決するためになされたものであり、本発明の第 5 の目的は、主制御回路で管理する処理プログラムやテーブルなどの容量を削減して主制御回路の ROM の空き容量を増やし、該増えた容量分の ROM の空き領域を利用して遊技性を高めることが可能な遊技機を提供することである。

【 1 4 3 5 】

上記第 5 の課題を解決するために、本発明では、以下のような構成の第 1 0 の遊技機を提供する。

20

【 1 4 3 6 】

遊技動作を制御するための演算処理を行う演算処理手段（例えば、メイン CPU 1 0 1 ）と、

前記演算処理手段による前記演算処理の実行に必要な情報が記憶された第 1 記憶手段（例えば、メイン ROM 1 0 2 ）と、

前記演算処理手段による前記演算処理の実行に必要な情報が記憶される第 2 記憶手段（例えば、メイン RAM 1 0 3 ）と、を備え、

前記演算処理手段は、

ソフトタイマーのタイマー値の計数処理（例えば、タイマー更新処理）において、更新命令、下限判定命令及び判断分岐命令を一つの命令で実行可能な所定の更新命令（例えば、「DCPWL D」命令）を実行することにより、現在の前記ソフトタイマーのタイマー値と前記タイマー値の下限値とを比較するとともに、現在の前記ソフトタイマーのタイマー値が前記下限値より大きければ、前記ソフトタイマーのタイマー値を減算更新し、現在の前記ソフトタイマーのタイマー値が前記下限値以下であれば、前記ソフトタイマーのタイマー値を前記下限値に保持することを特徴とする遊技機。

30

【 1 4 3 7 】

前記本発明の第 1 0 の遊技機では、前記ソフトタイマーが、2 バイトのソフトタイマーであるようにしてもよい。

【 1 4 3 8 】

40

また、前記本発明の第 1 0 の遊技機では、前記演算処理手段は、一定の周期で処理を行う定周期処理手段（例えば、1 . 1 1 7 2 m s e c 周期で繰り返し実行される割込処理）を有し、

前記ソフトタイマーによる前記タイマー値の計数処理は、前記定周期処理手段により実行され、

前記定周期処理手段が処理を行う周期と前記タイマー値とに基づいて、前記ソフトタイマーの経過時間が決定されるようにしてもよい。

【 1 4 3 9 】

さらに、前記本発明の第 1 0 の遊技機では、前記定周期処理手段による処理は、前記演算処理手段に内蔵されたタイマー機能（例えば、タイマー回路 1 1 3 ）が発生する割込信号

50

に基づいて実行されるようにしてもよい。

【1440】

上記構成の本発明の第10の遊技機によれば、主制御回路で管理する処理プログラムやテーブルなどの容量を削減して主制御回路のROM（第1記憶手段）の空き容量を増やし、該増えた容量分のROMの空き領域を利用して遊技性を高めることが可能な遊技機を提供することができる。

【1441】

[第11及び第12の遊技機]

従来、上述した構成の遊技機において、主制御回路の制御により内部抽籤結果を7セグメントLEDで表示する遊技機が知られている（例えば、特開2008-237337号公報参照）。

10

【1442】

ところで、従来、上述した遊技機特有の制限として、主制御回路のプログラム容量が、規則により小容量に制限されている。さらに、近年、遊技性の複雑化により主制御回路のROMの容量が圧迫されており、主制御回路で管理する処理プログラムやテーブルなどの容量削減が求められている。

【1443】

本発明は、上記第6の課題を解決するためになされたものであり、本発明の第6の目的は、主制御回路で管理する処理プログラムやテーブルなどの容量を削減して主制御回路のROMの空き容量を増やし、該増えた容量分のROMの空き領域を利用して遊技性を高めることが可能な遊技機を提供することである。

20

【1444】

上記第6の課題を解決するために、本発明では、以下のような構成の第11の遊技機を提供する。

【1445】

遊技動作を制御するための演算処理を行う演算処理手段（例えば、メインCPU101）と、

遊技に関する特定の情報を報知する複数の7セグLEDと、

前記複数の7セグLEDを駆動する7セグLED駆動手段（例えば、7セグLED駆動処理）と、

30

前記7セグLED駆動手段による前記複数の7セグLEDの駆動動作の制御を行うLED駆動制御手段と、

前記演算処理手段による前記演算処理の実行に必要な情報が記憶された第1記憶手段（例えば、メインROM102）と、

前記演算処理手段による前記演算処理の実行に必要な情報が記憶される第2記憶手段（例えば、メインRAM103）と、を備え、

前記LED駆動制御手段は、

前記複数の7セグLEDに対してダイナミック駆動制御を行い、所定の読み出し命令（例えば、「LDW」命令）を実行して、前記複数の7セグLEDに対してコモン選択データ及びカソードデータを同時に出力する

40

ことを特徴とする遊技機。

【1446】

また、前記本発明の第11の遊技機では、前記演算処理手段は、一定の周期で処理を行う定周期処理手段（例えば、1.1172msec周期で繰り返し実行される割込処理）を有し、

前記定周期処理手段は、前記定周期処理手段による処理の実行回数をカウントし、

前記カウントされた値が偶数である場合に、前記LED駆動制御手段による制御処理が実行されるようにしてもよい。

【1447】

また、上記第6の課題を解決するために、本発明では、以下のような構成の第12の遊技

50

機を提供する。

【 1 4 4 8 】

遊技媒体の投入操作を検出する投入操作検出手段（例えば、B E Tスイッチ77）と、
 前記投入操作検出手段による投入操作の検出された後、遊技者による開始操作を検出する
 開始操作検出手段（例えば、スタートスイッチ79）と、
 前記開始操作検出手段による開始操作の検出に基づいて予め定められた確率で内部当籤役
 を決定する内部当籤役決定手段（例えば、内部抽籤処理）と、
 複数の表示列を含み、前記開始操作検出手段による開始操作の検出に基づいて、各表示列
 に設けられた図柄を変動表示する変動表示手段（例えば、3つのリール3L, 3C, 3R
 ）と、
 遊技者による停止操作の検出を行う停止操作検出手段（例えば、ストップスイッチ基板8
 0）と、
 前記内部当籤役決定手段の決定結果と前記停止操作検出手段による停止操作の検出とに基
 づいて、前記図柄の変動表示を停止させる停止制御手段（例えば、リール停止制御処理）
 と、
 前記複数の表示列の変動表示の停止操作に関する情報を報知する複数の7セグLEDを含
 む指示表示器（例えば、指示モニタ）と、
 前記複数の7セグLEDを駆動する7セグLED駆動手段（例えば、7セグLED駆動処
 理）と、
 前記7セグLED駆動手段による前記複数の7セグLEDの駆動動作を制御するための演
 算処理を行う演算処理手段（例えば、メインCPU101）と、
 前記演算処理手段による前記演算処理の実行に必要な情報が記憶された第1記憶手段（例
 えば、メインROM102）と、
 前記演算処理手段による前記演算処理の実行に必要な情報が記憶される第2記憶手段（例
 えば、メインRAM103）と、を備え、
 前記演算処理手段は、
 前記複数の7セグLEDに対してダイナミック駆動制御を行い、所定の読み出し命令（例
 えば、「LDW」命令）を実行して、前記複数の7セグLEDに対してコモン選択データ
 及びカソードデータを同時に出力する
 ことを特徴とする遊技機。

10

20

30

【 1 4 4 9 】

また、前記本発明の第12の遊技機では、前記演算処理手段は、
 前記演算処理手段による前記演算処理の実行時にデータが格納される汎用レジスタと、
 前記演算処理手段による前記演算処理の実行時にデータが格納され、該格納されたデータ
 により前記第2記憶手段内のアドレスの一部を指定可能な拡張レジスタ（例えば、Qレジ
 スタ）と、有し、
 前記演算処理手段は、前記拡張レジスタを用いて前記第2記憶手段内のアドレス指定を行
 うことができる所定の読み出し命令（例えば、「LDQ」命令）を実行することにより、
 前記第2記憶手段内に配置され且つ前記複数の表示列の変動表示の停止操作に関する情報
 を格納する停止操作指示情報格納領域（例えば、ナビデータ格納領域）のアドレスを指定
 するとともに、前記複数の表示列の変動表示の停止操作に関する情報を前記停止操作指示
 情報格納領域に格納するようにしてもよい。

40

【 1 4 5 0 】

上記構成の本発明の第11及び第12の遊技機によれば、主制御回路で管理する処理プロ
 グラムやテーブルなどの容量を削減して主制御回路のROM（第1記憶手段）の空き容量
 を増やし、該増えた容量分のROMの空き領域を利用して遊技性を高めることができる。

【 1 4 5 1 】

[第13の遊技機]

従来、上述した構成の遊技機において、遊技制御基板（主制御基板）が、リールユニット
 を制御する遊技機が知られている（例えば、特開2012-034914号公報参照）。

50

【 1 4 5 2 】

ところで、リール（回胴）の回転制御において、リールの回転動作の安定感の欠如は、遊技者（特に熟練者）にとって不快感を与えることとなり、不快感から遊技の興味が削がれる可能性がある。この場合、遊技店の不利益となるとともに、遊技機自体の販売にも影響を及ぼす恐れがある。

【 1 4 5 3 】

本発明は、上記第 7 の課題を解決するためになされたものであり、本発明の第 7 の目的は、リールの回転動作の安定感の欠如を抑制し、遊技者に不快感を与えないようにすることが可能な遊技機を提供することである。

【 1 4 5 4 】

上記第 7 の課題を解決するために、本発明では、以下のような構成の第 1 3 の遊技機を提供する。

【 1 4 5 5 】

遊技媒体の投入操作を検出する投入操作検出手段（例えば、BETスイッチ 77）と、前記投入操作検出手段による投入操作の検出された後、遊技者による開始操作を検出する開始操作検出手段（例えば、スタートスイッチ 79）と、複数の表示列を含み、前記開始操作検出手段による開始操作の検出に基づいて、各表示列に設けられた図柄を変動表示する変動表示手段（例えば、3つのリール 3L, 3C, 3R）と、

遊技者による停止操作の検出を行う停止操作検出手段（例えば、ストップスイッチ基板 80）と、

遊技動作を制御するための演算処理を行う演算処理手段（例えば、メインCPU 101）と、

前記演算処理手段による前記演算処理の実行に必要な情報が記憶された第 1 記憶手段（例えば、メインROM 102）と、

前記演算処理手段による前記演算処理の実行に必要な情報が記憶される第 2 記憶手段（例えば、メインRAM 103）と、

前記第 2 記憶手段の所定の領域を初期化するための設定スイッチ（例えば、設定用鍵型スイッチ 54）と、を備え、

前記演算処理手段は、

前記開始操作検出手段による開始操作の検出に基づいて予め定められた確率で内部当籤役を決定する内部当籤役決定手段（例えば、内部抽籤処理）と、

前記内部当籤役決定手段の決定結果と前記停止操作検出手段による停止操作の検出とに基づいて、前記図柄の変動表示を停止させる停止制御手段（例えば、リール停止制御処理）と、

電源復帰時に電断発生時の入力ポートの入力状態及び出力ポートの出力状態をセットするとともに、電断発生時に前記表示列が変動中であった場合には、前記第 2 記憶手段に格納された電断発生時の前記表示列の変動制御管理情報（例えば、リール制御管理情報）をクリアするとともに、前記表示列の変動制御管理情報に前記表示列の変動開始を指示する情報をセットする遊技復帰手段（例えば、遊技復帰処理）と、を有する

ことを特徴とする遊技機。

【 1 4 5 6 】

また、前記本発明の第 1 3 の遊技機では、前記演算処理手段は、

前記設定スイッチがオフ状態である場合には、前記遊技復帰手段による処理を実行し、

前記設定スイッチがオン状態である場合には、前記遊技復帰手段による処理を実行することなく、前記第 2 記憶手段の所定の領域を初期化するようにしてもよい。

【 1 4 5 7 】

上記構成の本発明の第 1 3 の遊技機によれば、リールの回転動作（表示列の変動表示動作）の安定感の欠如を抑制し、遊技者に不快感を与えないようにすることができる。

【 1 4 5 8 】

10

20

30

40

50

[第 1 4 の遊技機]

従来、上述した構成の遊技機において、設定変更スイッチを操作することにより、設定値（1～6）を表示可能な遊技機が知られている（例えば、特開 2 0 0 8 - 2 4 5 7 0 4 号公報及び特開 2 0 1 3 - 0 4 2 8 7 0 号公報参照）。

【 1 4 5 9 】

ところで、従来、例えばボーナス遊技中、ART遊技中等の、遊技者に有利な遊技状態で遊技が行われている最中では、設定値の確認が行えない遊技機が主流である。このような遊技機では、「ゴト」と呼ばれる不正行為直後にボーナス遊技やART遊技が開始された場合、その不正行為を確認することができず、遊技店に不利益を与えてしまう可能性がある。

10

【 1 4 6 0 】

本発明は、上記第 8 の課題を解決するためになされたものであり、本発明の第 8 の目的は、遊技者に有利な遊技状態で遊技が行われている最中であっても、設定値等の情報を確認することができ、ゴト等の不正行為を抑止することが可能な遊技機を提供することである。

【 1 4 6 1 】

上記第 8 の課題を解決するために、本発明では、以下のような構成の第 1 4 の遊技機を提供する。

【 1 4 6 2 】

遊技機本体内部の所定の位置に配置された設定スイッチと、
遊技動作を制御するための演算処理を行う演算処理手段（例えば、メインCPU101）と、
前記演算処理手段による前記演算処理の実行に必要な情報が記憶された第 1 記憶手段（例えば、メインROM102）と、
前記演算処理手段による前記演算処理の実行に必要な情報が記憶される第 2 記憶手段（例えば、メインRAM103）と、
遊技媒体の投入操作を検出する投入操作検出手段（例えば、BETスイッチ77）と、
前記投入操作検出手段による投入操作の検出された後、遊技者による開始操作を検出する開始操作検出手段（例えば、スタートスイッチ79）と、
前記開始操作検出手段による開始操作の検出に基づいて予め定められた確率で内部当籤役を決定する内部当籤役決定手段（例えば、内部抽籤処理）と、
前記内部当籤役決定手段により決定された内部当籤役に基づいて、遊技者にとって有利な遊技状態を実行する有利遊技手段（例えば、後述のボーナスゲーム）と、
前記設定スイッチの操作に応じて、内部当籤役が決定される確率に係る設定値を変更又は確認可能な設定変更確認手段（例えば、メダル受付・スタートチェック処理中のS233）と、
複数の表示列を含み、前記開始操作検出手段による開始操作の検出に基づいて、各表示列に設けられた図柄を変動表示する変動表示手段（例えば、3つのリール3L, 3C, 3R）と、
遊技者による停止操作の検出を行う停止操作検出手段（例えば、ストップスイッチ基板80）と、
前記内部当籤役決定手段の決定結果と前記停止操作検出手段による停止操作の検出とに基づいて、前記図柄の変動表示を停止させる停止制御手段（例えば、リール停止制御処理）と、
遊技が開始可能であるか否かを判定する遊技開始判定手段（例えば、メダル受付・スタートチェック処理）と、を備え、
前記設定変更確認手段は、電源投入時に前記設定スイッチの操作に応じて、内部当籤役が決定される確率に係る設定値を変更可能であり、
前記遊技開始判定手段により遊技が開始可能であると判定され且つ前記設定スイッチを操作された場合には、遊技状態に関係なく、内部当籤役が決定される確率に係る設定値を前記設定変更確認手段による処理により確認可能にする

20

30

40

50

ことを特徴とする遊技機。

【 1 4 6 3 】

また、前記本発明の第 1 4 の遊技機では、前記設定変更確認手段は、前記設定スイッチが操作された状態で遊技機に電源投入された場合には、前記第 2 記憶手段の所定の記憶領域を初期化するとともに、前記設定値を変更し、該変更した前記設定値を前記第 2 記憶手段に格納するようにしてもよい。

【 1 4 6 4 】

上記構成の本発明の第 1 4 の遊技機によれば、例えばボーナス遊技中、ART 遊技中等の、遊技者に有利な遊技状態で遊技が行われている最中であっても、設定値等の情報を確認することができ、ゴト等の不正行為を抑止することができる。

10

【 1 4 6 5 】

[第 1 5 及び第 1 6 の遊技機]

従来、上述した構成の遊技機において、メダルの投入枚数を表示するための表示装置を備えた遊技機が知られている（例えば、特開 1 9 9 9 - 1 7 8 9 8 3 号公報参照）。

【 1 4 6 6 】

ところで、従来、上述した遊技機特有の制限として、主制御回路のプログラム容量が、規則により小容量に制限されている。さらに、近年、遊技性の複雑化により主制御回路の ROM の容量が圧迫されており、主制御回路で管理する処理プログラムやテーブルなどの容量削減が求められている。

【 1 4 6 7 】

本発明は、上記第 9 の課題を解決するためになされたものであり、本発明の第 9 の目的は、主制御回路で管理する処理プログラムやテーブルなどの容量を削減して主制御回路の ROM の空き容量を増やし、該増えた容量分の ROM の空き領域を利用して遊技性を高めることが可能な遊技機を提供することである。

20

【 1 4 6 8 】

上記第 9 の課題を解決するために、本発明では、以下のような構成の第 1 5 の遊技機を提供する。

【 1 4 6 9 】

遊技媒体の投入操作を検出する投入操作検出手段（例えば、BET スイッチ 7 7）と、前記投入操作検出手段による投入操作の検出された後、遊技者による開始操作を検出する開始操作検出手段（例えば、スタートスイッチ 7 9）と、前記開始操作検出手段による開始操作の検出に基づいて予め定められた確率で内部当籤役を決定する内部当籤役決定手段（例えば、内部抽籤処理）と、複数の表示列を含み、前記開始操作検出手段による開始操作の検出に基づいて、各表示列に設けられた図柄を変動表示する変動表示手段（例えば、3 つのリール 3 L , 3 C , 3 R）と、

30

遊技者による停止操作の検出を行う停止操作検出手段（例えば、ストップスイッチ基板 8 0）と、

前記内部当籤役決定手段の決定結果と前記停止操作検出手段による停止操作の検出とに基づいて、前記図柄の変動表示を停止させる停止制御手段（例えば、リール停止制御処理）と、

40

前記遊技媒体の受付状態（例えば、メダルセンサ入力状態）を検出する遊技媒体検出手段（例えば、メダルセンサ）と、

前記遊技媒体検出手段により検出されている現在の前記遊技媒体の受付状態の変化態様が正常であるか否かを、前回処理で検出された前記遊技媒体の受付状態に基づいて、演算処理により判別する遊技媒体受付状態判別手段（例えば、メダル投入チェック処理中の S 2 5 5 ~ S 2 5 8）と、を備える

遊技機。

【 1 4 7 0 】

前記本発明の第 1 5 の遊技機では、前記遊技媒体受付状態判別手段は、前回処理で検出さ

50

れた前記遊技媒体の受付状態に基づいて今回処理で検出され得る前記遊技媒体の受付状態の正常値を論理演算で算出し、前記正常値と、現在の前記遊技媒体の受付状態とを比較して、前記遊技媒体の受付状態の変化態様が正常であるか否かを判別するようにしてもよい。

【1471】

また、前記本発明の第15の遊技機では、前記遊技媒体受付状態判別手段は、1バイトの情報内の2ビットにより、前記遊技媒体の受付状態の変化態様が正常である否かを判別するようにしてもよい。

【1472】

上記第9の課題を解決するために、本発明では、以下のような構成の第16の遊技機を提供する。

【1473】

遊技媒体の投入操作を検出する投入操作検出手段（例えば、BETスイッチ77）と、前記投入操作検出手段による投入操作の検出された後、遊技者による開始操作を検出する開始操作検出手段（例えば、スタートスイッチ79）と、

前記開始操作検出手段による開始操作の検出に基づいて予め定められた確率で内部当籤役を決定する内部当籤役決定手段（例えば、内部抽籤処理）と、

複数の表示列を含み、前記開始操作検出手段による開始操作の検出に基づいて、各表示列に設けられた図柄を変動表示する変動表示手段（例えば、3つのリール3L, 3C, 3R）と、

遊技者による停止操作の検出を行う停止操作検出手段（例えば、ストップスイッチ基板80）と、

前記内部当籤役決定手段の決定結果と前記停止操作検出手段による停止操作の検出とに基づいて、前記図柄の変動表示を停止させる停止制御手段（例えば、リール停止制御処理）と、

前記遊技媒体の投入数を示す情報を、前記遊技媒体の投入数に対応する表示態様で報知する表示手段（例えば、第1LED～第3LED）と、

前記表示手段による報知動作を制御するための演算処理を行う演算処理手段（例えば、メインCPU101、メダル投入処理）と、

前記演算処理手段による前記演算処理の実行に必要な情報が記憶された第1記憶手段（例えば、メインROM102）と、

前記演算処理手段による前記演算処理の実行に必要な情報が記憶される第2記憶手段（例えば、メインRAM103）と、を備え、

前記演算処理手段は、

前記遊技媒体の投入数を示す情報を前記遊技媒体の投入数に対応する表示態様で報知するための点灯制御データを、前記遊技媒体の投入数に基づいて論理演算処理により生成することを特徴とする遊技機。

【1474】

また、前記本発明の第16の遊技機において、前記論理演算処理では、前記遊技媒体の点灯制御データは、1バイト内の情報の3ビットのデータから生成されるようにしてもよい。

【1475】

上記構成の本発明の第15及び第16の遊技機によれば、主制御回路で管理する処理プログラムやテーブルなどの容量を削減して主制御回路のROMの空き容量を増やし、該増えた容量分のROMの空き領域を利用して遊技性を高めることができる。

【1476】

[第17及び第18の遊技機]

従来、上述した構成の遊技機において、メイン基板（主制御基板）とサブ基板（副制御基板）とが通信により接続され、メイン基板からサブ基板へコマンドが送信される遊技機が知られている（例えば、特許第5725590号公報参照）。

【1477】

ところで、従来、遊技に係る制御を行う主制御基板と演出に係る制御を行う副制御基板と

10

20

30

40

50

の間における通信は、主制御基板から副制御基板への一方向通信であり、また、電源投入時における主制御基板の起動時間と副制御基板のそれとの間には大きな隔りがある。それゆえ、電源投入時における主制御基板及び副制御基板間の通信接続が不安定になる恐れがある。

【 1 4 7 8 】

本発明は、上記第 1 0 の課題を解決するためになされたものであり、本発明の第 1 0 の目的は、電源投入時における主制御基板（主制御手段）及び副制御基板（副制御手段）間の通信を安定動作させることが可能な遊技機を提供することである。

【 1 4 7 9 】

上記第 1 0 の課題を解決するために、本発明では、以下のような構成の第 1 7 の遊技機を提供する。

10

【 1 4 8 0 】

遊技動作に関する通信データを送信する主制御手段（例えば、主制御回路 9 0 ）を備え、前記主制御手段は、

前記主制御手段による制御処理中に、所定周期で割込処理を実行し、割込処理実行時に、遊技動作に関する通信データが無い場合には、無操作コマンドを送信する割込処理実行手段（例えば、割込処理）と、

電源復帰時に所定の電源復帰処理を実行する電源復帰処理実行手段（例えば、電源投入時処理）と、を有し、

前記電源復帰処理実行手段は、前記電源復帰処理開始後、前記割込処理実行手段による前記割込処理の実行が可能になるまで待機する処理（例えば、電源投入時処理中の S 7 及び S 8 ）を行い、

20

前記割込処理実行手段は、前記電源復帰処理実行手段による前記待機の終了時に、前記割込処理を実行して前記無操作コマンドを前記副制御手段に送信する

ことを特徴とする遊技機。

【 1 4 8 1 】

また、前記本発明の第 1 7 の遊技機では、前記主制御手段は、

前記所定周期を計測するタイマー回路（例えば、タイマー回路 1 1 3 ）と、

前記タイマー回路のタイムアウト信号に基づいて前記割込処理を実行させるための割込信号を発生させる割込回路（例えば、割込みコントローラ 1 1 2 ）と、を有し、

30

前記電源復帰処理実行手段は、

前記所定周期で前記タイムアウト信号を発生させるための初期設定を前記タイマー回路に設定した後、前記割込処理実行手段による前記割込処理の実行が可能になるまで待機する処理を行い、

前記待機の終了を、前記タイマー回路の前記タイムアウト信号の発生の有無に基づいて判断するようにしてもよい。

【 1 4 8 2 】

また、上記第 1 0 の課題を解決するために、本発明では、以下のような構成の第 1 8 の遊技機を提供する。

【 1 4 8 3 】

40

遊技動作を制御するための演算処理を行う演算処理手段（例えば、メイン CPU 1 0 1 ）と、

遊技動作に関するコマンドデータを作成する通信データ生成手段（例えば、設定変更コマンド生成格納処理及び通信データ格納処理）と、

前記演算処理手段による制御処理中に、所定周期で割込処理を実行し、割込処理実行時に、遊技動作に関するコマンドデータが無い場合には、無操作コマンドを送信する割込処理実行手段（例えば、割込処理）と、

電源復帰時に所定の電源復帰処理を実行する電源復帰処理実行手段（例えば、電源投入時処理）と、

前記演算処理手段による前記演算処理の実行に必要な情報が記憶された第 1 記憶手段（例

50

えば、メインROM102)と、
 前記演算処理手段による前記演算処理の実行に必要な情報が記憶される第2記憶手段(例えば、メインRAM103)と、を備え、
 前記電源復帰処理実行手段は、前記電源復帰処理開始後、前記割込処理実行手段による前記割込処理の実行が可能になるまで待機する処理(例えば、電源投入時処理中のS7及びS8)を行い、
 前記割込処理実行手段は、前記電源復帰処理実行手段による前記待機の終了時に、前記割込処理を実行して前記無操作コマンドを送信し、
 前記演算処理手段は、
 前記演算処理手段による前記演算処理の実行時に複数種のデータがそれぞれ格納される複数の汎用レジスタを有し、
 前記コマンドデータは、コマンド種別と複数の通信パラメータとサム値とにより構成され、
 複数の前記通信パラメータは、それぞれ複数の前記汎用レジスタに割り当てられ、
 前記通信データ生成手段は、
 前記コマンドデータの前記コマンド種別に応じて、前記通信パラメータに割り当てられた前記汎用レジスタに通信パラメータをセットした後、前記汎用レジスタにセットされた通信パラメータを、前記第2記憶手段内の所定の格納領域(例えば、通信データ一時格納領域)に格納し、
 前記複数の通信パラメータのうち、前記コマンドデータの前記コマンド種別に基づいて使用されない通信パラメータがある場合にも、当該未使用の通信パラメータに対応付けられた汎用レジスタに格納されているデータを通信パラメータとして前記所定の格納領域に格納し、
 前記コマンド種別と、通信パラメータに割り当てられた前記汎用レジスタに格納されているデータとに基づいて、前記コマンドデータのサム値を生成することを特徴とする遊技機。

10

20

【1484】

また、前記本発明の第18の遊技機では、電源電圧の状態を監視する電源監視手段(例えば、電源監視ポート)を備え、
 前記演算処理手段は、
 前記電源電圧の状態が安定していない場合には、前記電源電圧の状態が安定するまで待機し、
 前記電源電圧の状態が安定していることを条件に、前記演算処理手段のタイマー回路、シリアル通信回路及び乱数回路を初期化した後、前記第2記憶手段の所定の領域を使用して前記第2記憶手段が正常であるか否かを確認し、
 前記第2記憶手段が正常であることを条件に、前記無操作コマンドを送信するようにしてもよい。

30

【1485】

上記構成の本発明の第17及び第18の遊技機によれば、電源投入時における主制御基板及び副制御基板間の通信を安定動作させることができる。

【1486】

[第19及び第20の遊技機]

従来、上述した構成の遊技機において、抽籤結果に基づいて、例えば、リールの入賞図柄の引き込み制御と呼ばれる制御や、リールの入賞図柄の蹴飛ばし制御と呼ばれる制御を実行する遊技機が知られている(例えば、特開2002-219213号公報参照)。

【1487】

ところで、従来、上述した遊技機特有の制限として、主制御回路のプログラム容量が、規則により小容量に制限されている。さらに、近年、遊技性が複雑になるとともに、成立役(入賞役)の図柄組合せパターンが増大したため、主制御回路のRAM容量が圧迫されており、主制御回路で実行する処理の効率化を図り、主制御回路の限られたRAM容量を有効活用可能な技術の開発が求められている。

40

50

【 1 4 8 8 】

本発明は、上記第 1 1 の課題を解決するためになされたものであり、本発明の第 1 1 の目的は、主制御回路で実行する処理の効率化を図り、主制御回路の R A M 容量を有効活用することが可能な遊技機を提供することである。

【 1 4 8 9 】

上記第 1 1 の課題を解決するために、本発明では、以下のような構成の第 1 9 の遊技機を提供する。

【 1 4 9 0 】

遊技動作を制御するための演算処理を行う演算処理手段（例えば、メイン C P U 1 0 1 ）と、

前記演算処理手段による前記演算処理の実行に必要な情報が記憶された第 1 記憶手段（例えば、メイン R O M 1 0 2 ）と、

前記演算処理手段による前記演算処理の実行に必要な情報が記憶される第 2 記憶手段（例えば、メイン R A M 1 0 3 ）と、を備え、

前記演算処理手段は、

予め定められた確率で、内部当籤役に関するフラグステータス情報（例えば、当り要求フラグステータス）を決定する内部抽籤手段（例えば、内部抽籤処理）と、

前記内部抽籤手段により取得された前記フラグステータス情報から内部当籤役を生成する内部当籤役生成手段（例えば、図柄設定処理中の S 3 2 1 ）と、

前記内部当籤役と、それに対応するフラグデータと、前記第 2 記憶手段内の前記フラグデータの格納先を指定する格納先データとの対応関係を規定し且つ前記第 1 記憶手段に記憶されたフラグデータテーブル（例えば、当り要求フラグテーブル）を前記第 2 記憶手段内に展開するフラグテーブル展開手段（例えば、図柄設定処理中の S 3 2 4 ）と、

前記第 2 記憶手段内に配置され且つ前記フラグデータを格納するフラグデータ格納領域（例えば、当り要求フラグ格納領域）のアドレスを指定するフラグ格納領域指定手段（例えば、図柄設定処理中の S 3 2 9 ）と、

前記内部当籤役生成手段により生成された前記内部当籤役に対応する前記フラグデータを、対応する前記格納先データに基づいて、前記フラグテーブルから前記フラグデータ格納領域内に転送して格納するフラグデータ格納手段（例えば、図柄設定処理中の S 3 3 0 ）と、を有する

ことを特徴とする遊技機。

【 1 4 9 1 】

また、前記本発明の第 1 9 の遊技機では、前記演算処理手段は、前記フラグデータテーブルの前記格納先データを示すオンビット情報を算出して、転送対象となる前記オンビット情報に設定されている前記フラグデータを前記フラグデータ格納領域に転送するようにしてもよい。

【 1 4 9 2 】

また、上記第 1 1 の課題を解決するために、本発明では、以下のような構成の第 2 0 の遊技機を提供する。

【 1 4 9 3 】

遊技媒体の投入操作を検出する投入操作検出手段（例えば、B E T スイッチ 7 7 ）と、前記投入操作検出手段による投入操作の検出された後、遊技者による開始操作を検出する開始操作検出手段（例えば、スタートスイッチ 7 9 ）と、

複数の表示列を含み、前記開始操作検出手段による開始操作の検出に基づいて、各表示列に設けられた図柄を変動表示する変動表示手段（例えば、3 つのリール 3 L , 3 C , 3 R ）と、

遊技者による停止操作の検出を行う停止操作検出手段（例えば、ストップスイッチ基板 8 0 ）と、

遊技動作を制御するための演算処理を行う演算処理手段（例えば、メイン C P U 1 0 1 ）と、

10

20

30

40

50

前記演算処理手段による前記演算処理の実行に必要な情報が記憶された第 1 記憶手段（例えば、メインROM 102）と、

前記演算処理手段による前記演算処理の実行に必要な情報が記憶される第 2 記憶手段（例えば、メインRAM 103）と、を備え、

前記演算処理手段は、

前記演算処理手段による前記演算処理の実行時にデータが格納される汎用レジスタと、

前記演算処理手段による前記演算処理の実行時にデータが格納され、該格納されたデータにより前記第 1 記憶手段又は前記第 2 記憶手段内のアドレスの一部を指定可能な拡張レジスタ（例えば、Qレジスタ）と、

予め定められた確率で、内部当籤役に関するフラグステータス情報（例えば、当り要求フラグステータス）を決定する内部抽籤手段（例えば、内部抽籤処理）と、

前記内部抽籤手段により取得された前記フラグステータス情報から内部当籤役を生成する内部当籤役生成手段（例えば、図柄設定処理中の S 3 2 1）と、

前記内部当籤役と、それに対応する当籤フラグデータと、前記第 2 記憶手段内の前記当籤フラグデータの格納先を指定する格納先データとの対応関係を規定した当籤フラグデータテーブル（例えば、当り要求フラグテーブル）を前記第 2 記憶手段内に展開する当籤フラグテーブル展開手段（例えば、図柄設定処理中の S 3 2 4）と、

前記拡張レジスタを用いて前記第 2 記憶手段内のアドレス指定を行うことができる所定の読み出し命令（例えば、「LDQ」命令）を実行することにより、前記第 2 記憶手段内に配置され且つ前記当籤フラグデータを格納する当籤フラグデータ格納領域（例えば、当り要求フラグ格納領域）のアドレスを指定する当籤フラグ格納領域指定手段（例えば、図柄設定処理中の S 3 2 9）と、

前記内部当籤役生成手段により生成された前記内部当籤役に対応する前記当籤フラグデータを、対応する前記格納先データに基づいて、前記当籤フラグテーブルから前記当籤フラグデータ格納領域内に転送して格納する当籤フラグデータ格納手段（例えば、図柄設定処理中の S 3 3 0）と、

前記内部抽籤手段の決定結果と前記停止操作検出手段による停止操作の検出とに基づいて、前記図柄の変動表示を停止させる停止制御手段（例えば、リール停止制御処理）と、

前記停止制御手段により前記複数の表示列の変動表示が停止された場合に、前記複数の表示列に跨って設定された判定ライン上に、前記内部当籤役に対応する図柄の組合せが停止表示されたか否かを判定する入賞役決定手段（例えば、入賞検索処理）と、

前記所定の読み出し命令を実行することにより、前記第 2 記憶手段内に配置され且つ前記判定ライン上に停止表示された図柄の組合せに対応する入賞フラグデータを格納する入賞フラグデータ格納領域（例えば、入賞作動フラグ格納領域）のアドレスを指定する入賞フラグ格納領域指定手段（例えば、図柄コード取得処理中の S 6 4 8）と、

前記所定の読み出し命令を実行することにより、前記第 2 記憶手段内に配置され且つ前記判定ライン上に停止表示された図柄の組合せに対応する図柄コードデータを格納する図柄コード格納領域のアドレスを指定する図柄コード格納領域設定手段（例えば、図柄コード取得処理中の S 6 5 0）と、を有する

ことを特徴とする遊技機。

【 1 4 9 4 】

また、前記本発明の第 20 の遊技機では、前記拡張レジスタで指定可能な前記アドレスの一部が、前記アドレスを構成する上位側のアドレス値であるようにしてもよい。

【 1 4 9 5 】

上記構成の本発明の第 19 及び第 20 の遊技機によれば、主制御回路で実行する処理の効率化を図り、主制御回路の RAM（第 2 記憶手段）容量を有効活用することができる。

【 1 4 9 6 】

[第 21 及び第 22 の遊技機]

従来、上述した構成の遊技機において、内部当籤役と遊技者の停止操作とに基づきリール図柄の可変表示の停止制御を行い、図柄の組合せにより入賞判定を行い、入賞判定の結果

10

20

30

40

50

に基づいてホッパー（メダル払出装置）を制御して、メダルの払出制御を行う遊技機が知られている（例えば、特開2008-119498号公報参照）。

【1497】

ところで、従来、上述した遊技機特有の制限として、主制御回路のプログラム容量が、規則により小容量に制限されている。さらに、近年、遊技性の複雑化により主制御回路のROMの容量が圧迫されており、主制御回路で管理する処理プログラムやテーブルなどの容量削減が求められている。

【1498】

本発明は、上記第12の課題を解決するためになされたものであり、本発明の第12の目的は、主制御回路で管理する処理プログラムやテーブルなどの容量を削減して主制御回路のROMの空き容量を増やし、該増えた容量分のROMの空き領域を利用して遊技性を高めることが可能な遊技機を提供することである。

【1499】

上記第12の課題を解決するために、本発明では、以下のような構成の第21の遊技機を提供する。

【1500】

遊技媒体の投入操作を検出する投入操作検出手段（例えば、BETスイッチ77）と、前記投入操作検出手段による投入操作の検出された後、遊技者による開始操作を検出する開始操作検出手段（例えば、スタートスイッチ79）と、

前記開始操作検出手段による開始操作の検出に基づいて予め定められた確率で内部当籤役を決定する内部当籤役決定手段（例えば、内部抽籤処理）と、

複数の表示列を含み、前記開始操作検出手段による開始操作の検出に基づいて、各表示列に設けられた図柄を変動表示する変動表示手段（例えば、3つのリール3L, 3C, 3R）と、

遊技者による停止操作の検出を行う停止操作検出手段（例えば、ストップスイッチ基板80）と、

前記内部当籤役決定手段の決定結果と前記停止操作検出手段による停止操作の検出とに基づいて、前記図柄の変動表示を停止させる停止制御手段（例えば、リール停止制御処理）と、

前記停止制御手段による前記表示列の停止動作を制御するための演算処理を行う演算処理手段（例えば、メインCPU101）と、

前記演算処理手段による前記演算処理の実行に必要な情報が記憶された第1記憶手段（例えば、メインROM102）と、

前記演算処理手段による前記演算処理の実行に必要な情報が記憶される第2記憶手段（例えば、メインRAM103）と、を備え、

前記演算処理手段は、

前記演算処理手段による前記演算処理の実行時にデータが格納される汎用レジスタと、

前記演算処理手段による前記演算処理の実行時にデータが格納され、該格納されたデータにより前記第2記憶手段内のアドレスの一部を指定可能な拡張レジスタ（例えば、Qレジスタ）と、

前記拡張レジスタを用いて前記第2記憶手段内のアドレス指定を行うことができる所定の読み出し命令（例えば、「LDQ」命令）を実行することにより、前記停止操作検出手段による停止操作の検出結果を取得する停止操作検出結果取得手段（例えば、リール停止制御処理中のS714）と、

前記停止操作検出手段により所定の表示列に対する遊技者の停止操作が検出された場合に、前記所定の読み出し命令を実行することにより、前記第2記憶手段内に配置され且つ前記所定の表示列の変動表示の停止制御データを格納する停止制御データ格納領域のアドレスを指定する停止制御データ格納領域設定手段（例えば、図139中のソースコード「LDQ IX, WR1__CTRL - (WR2__CTRL - WR1__CTRL)」）と、を有する

10

20

30

40

50

ことを特徴とする遊技機。

【 1 5 0 1 】

また、前記本発明の第 2 1 の遊技機では、前記拡張レジスタで指定可能な前記アドレスの一部が、前記アドレスを構成する上位側のアドレス値であるようにしてもよい。

【 1 5 0 2 】

また、上記第 1 2 の課題を解決するために、本発明では、以下のような構成の第 2 2 の遊技機を提供する。

【 1 5 0 3 】

遊技媒体の投入操作を検出する投入操作検出手段（例えば、B E T スイッチ 7 7 ）と、
前記投入操作検出手段による投入操作の検出された後、遊技者による開始操作を検出する
開始操作検出手段（例えば、スタートスイッチ 7 9 ）と、

10

前記開始操作検出手段による開始操作の検出に基づいて予め定められた確率で内部当籤役
を決定する内部当籤役決定手段（例えば、内部抽籤処理）と、

複数の表示列を含み、前記開始操作検出手段による開始操作の検出に基づいて、各表示列
に設けられた図柄を変動表示する変動表示手段（例えば、3 つのリール 3 L , 3 C , 3 R
）と、

遊技者による停止操作の検出を行う停止操作検出手段（例えば、ストップスイッチ基板 8
0 ）と、

前記内部当籤役決定手段の決定結果と前記停止操作検出手段による停止操作の検出とに基
づいて、前記図柄の変動表示を停止させる停止制御手段（例えば、リール停止制御処理）

20

と、

前記内部当籤役決定手段により複数種の内部当籤役が決定された場合に、前記複数種の内
部当籤役にそれぞれ対応する複数種の図柄の組合せの中なら、前記複数の表示列に跨って
設定された判定ライン上に、優先して停止表示させる図柄の組合せを決定する優先停止図
柄決定手段（例えば、引込優先順位取得処理）と、

前記停止制御手段による前記表示列の停止動作、及び、前記優先停止図柄決定手段による
停止表示させる図柄の組合せの決定動作を制御するための演算処理を行う演算処理手段（
例えば、メイン C P U 1 0 1 ）と、

前記演算処理手段による前記演算処理の実行に必要な情報が記憶された第 1 記憶手段（例
えば、メイン R O M 1 0 2 ）と、

30

前記演算処理手段による前記演算処理の実行に必要な情報が記憶される第 2 記憶手段（例
えば、メイン R A M 1 0 3 ）と、を備え、

前記演算処理手段は、

前記演算処理手段による前記演算処理の実行時にデータが格納される汎用レジスタと、

前記演算処理手段による前記演算処理の実行時にデータが格納され、該格納されたデータ
により前記第 2 記憶手段内のアドレスの一部を指定可能な拡張レジスタ（例えば、Q レジ
スタ）と、

前記拡張レジスタを用いて前記第 2 記憶手段内のアドレス指定を行うことができる所定の
読み出し命令（例えば、「L D Q」命令）を実行することにより、前記停止操作検出手段
による停止操作の検出結果を取得する停止操作検出結果取得手段（例えば、リール停止制
御処理中の S 7 1 4 ）と、

40

前記停止操作検出手段により所定の表示列に対する遊技者の停止操作が検出された場合に
、前記所定の読み出し命令を実行することにより、前記第 2 記憶手段内に配置され且前記
所定の表示列の変動表示の停止制御データを格納する停止制御データ格納領域のアドレス
を指定する停止制御データ格納領域設定手段（例えば、図 1 3 9 中のソースコード「L D
Q I X , w R 1 _ C T R L - (w R 2 _ C T R L - w R 1 _ C T R L)」）と、

前記優先停止図柄決定手段により、優先して、前記判定ライン上に停止表示させる図柄の
組合せを決定する処理において、比較判定命令、処理のジャンプ先アドレスの指定命令及
び処理のジャンプ動作命令を一つの命令で実行可能な所定の判定命令（例えば、「J C P
」命令）を実行することにより、判定対象となる前記内部当籤役に対応する図柄の組合せ

50

に、現在判定対象中の特定の表示列（例えば、右リール 3 R）における前記判定ライン上の停止図柄が任意となる所定の図柄の組合せ（例えば、「ANY」役）が含まれるか否かを判定するとともに、判定対象となる前記内部当籤役に対応する図柄の組合せに前記所定の図柄の組合せが含まれると判定された場合に、判定対象となる前記内部当籤役に対応する図柄の組合せの停止表示を禁止する任意役対応処理手段（例えば、引込優先順位取得処理中の S 6 8 3）と、を有する

ことを特徴とする遊技機。

【1504】

また、前記本発明の第 2 2 の遊技機では、前記拡張レジスタで指定可能な前記アドレスの一部が、前記アドレスを構成する上位側のアドレス値であるようにしてもよい。

10

【1505】

上記構成の本発明の第 2 1 及び第 2 2 の遊技機によれば、主制御回路で管理する処理プログラムやテーブルなどの容量を削減して主制御回路の ROM の空き容量を増やし、該増えた容量分の ROM の空き領域を利用して遊技性を高めることができる。

【1506】

[第 2 3 ~ 第 2 5 の遊技機]

従来、上述した構成の遊技機において、図柄の停止制御時に引込優先順位テーブルを使用して図柄の停止制御を行う遊技機が知られている（例えば、特開 2 0 0 7 - 1 7 5 4 5 0 号公報参照）。

【1507】

20

ところで、従来、上述した遊技機特有の制限として、主制御回路のプログラム容量が、規則により小容量に制限されている。さらに、近年、遊技性の複雑化により主制御回路の ROM の容量が圧迫されており、主制御回路で管理する処理プログラムやテーブルなどの容量削減が求められている。

【1508】

本発明は、上記第 1 3 の課題を解決するためになされたものであり、本発明の第 1 3 の目的は、主制御回路で管理する処理プログラムやテーブルなどの容量を削減して主制御回路の ROM の空き容量を増やし、該増えた容量分の ROM の空き領域を利用して遊技性を高めることが可能な遊技機を提供することである。

【1509】

30

上記第 1 3 の課題を解決するために、本発明では、以下のような構成の第 2 3 の遊技機を提供する。

【1510】

遊技媒体の投入操作を検出する投入操作検出手段（例えば、BET スイッチ 7 7）と、前記投入操作検出手段による投入操作の検出された後、遊技者による開始操作を検出する開始操作検出手段（例えば、スタートスイッチ 7 9）と、

前記開始操作検出手段による開始操作の検出に基づいて予め定められた確率で内部当籤役を決定する内部当籤役決定手段（例えば、内部抽籤処理）と、

複数の表示列を含み、前記開始操作検出手段による開始操作の検出に基づいて、各表示列に設けられた図柄を変動表示する変動表示手段（例えば、3 つのリール 3 L, 3 C, 3 R）と、

40

遊技者による停止操作の検出を行う停止操作検出手段（例えば、ストップスイッチ基板 8 0）と、

前記内部当籤役決定手段の決定結果と前記停止操作検出手段による停止操作の検出とに基づいて、前記図柄の変動表示を停止させる停止制御手段（例えば、リール停止制御処理）と、

前記内部当籤役決定手段により複数種の内部当籤役が決定された場合に、前記複数種の内部当籤役にそれぞれ対応する複数種の図柄の組合せの中なら、前記複数の表示列に跨って設定された判定ライン上に、優先して停止表示させる図柄の組合せを決定する優先停止図柄決定手段（例えば、引込優先順位取得処理）と、

50

前記優先停止図柄決定手段による優先して停止表示させる図柄の組合せの決定動作を制御するための演算処理を行う演算処理手段（例えば、メインCPU101）と、
 前記演算処理手段による前記演算処理の実行に必要な情報が記憶された第1記憶手段（例えば、メインROM102）と、
 前記演算処理手段による前記演算処理の実行に必要な情報が記憶される第2記憶手段（例えば、メインRAM103）と、を備え、
 前記演算処理手段は、
 前記演算処理手段による前記演算処理の実行時にデータが格納される汎用レジスタと、
 前記演算処理手段による前記演算処理の実行時にデータが格納され、該格納されたデータにより前記第2記憶手段内のアドレスの一部を指定可能な拡張レジスタ（例えば、Qレジスタ）と、を有し、
 前記優先停止図柄決定手段により、優先して停止表示させる図柄の組合せを決定する処理において、
 前記演算処理手段は、
 前記拡張レジスタを用いて前記第2記憶手段内のアドレス指定を行うことができる所定の読み出し命令（例えば、「LDQ」命令）を実行することにより、前記第2記憶手段内に配置され且つ前記内部当籤役に対応する当籤フラグデータを格納する当籤フラグデータ格納領域（例えば、当り要求フラグ格納領域）のアドレスを指定する当籤フラグ格納領域指定手段（例えば、引込優先順位取得処理中のS686）と、
 前記所定の読み出し命令を実行することにより、前記第2記憶手段内に配置され且つ前記判定ライン上に停止表示された図柄の組合せに対応する入賞フラグデータを格納する入賞フラグデータ格納領域（例えば、入賞作動フラグ格納領域）のアドレスを指定する入賞フラグ格納領域指定手段（例えば、引込優先順位取得処理中のS686）と、
 前記当籤フラグデータとそれに対応する前記入賞フラグデータとの論理積演算を行う論理積演算手段（例えば、引込優先順位取得処理中のS686）と、を有することを特徴とする遊技機。

【1511】

また、前記本発明の第23の遊技機では、前記優先停止図柄決定手段により、優先して停止表示させる図柄の組合せを決定する処理において、判定対象となる前記内部当籤役に対応する図柄の組合せに、現在判定対象中の特定の表示列（例えば、右リール3R）における前記判定ライン上の停止図柄が任意となる所定の図柄の組合せ（例えば、「ANY」役）が含まれる場合には、
 前記演算処理手段は、前記当籤フラグ格納領域指定手段による前記当籤フラグデータ格納領域のアドレス指定処理、前記入賞フラグ格納領域指定手段による前記入賞フラグデータ格納領域のアドレス指定処理、及び、前記論理積演算手段による論理積演算処理が実行されないようにしてもよい。

【1512】

また、上記第13の課題を解決するために、本発明では、以下のような構成の第24の遊技機を提供する。

【1513】

遊技媒体の投入操作を検出する投入操作検出手段（例えば、BETスイッチ77）と、
 前記投入操作検出手段による投入操作の検出された後、遊技者による開始操作を検出する開始操作検出手段（例えば、スタートスイッチ79）と、
 前記開始操作検出手段による開始操作の検出に基づいて予め定められた確率で内部当籤役を決定する内部当籤役決定手段（例えば、内部抽籤処理）と、
 複数の表示列を含み、前記開始操作検出手段による開始操作の検出に基づいて、各表示列に設けられた図柄を変動表示する変動表示手段（例えば、3つのリール3L, 3C, 3R）と、
 遊技者による停止操作の検出を行う停止操作検出手段（例えば、ストップスイッチ基板80）と、

10

20

30

40

50

前記内部当籤役決定手段の決定結果と前記停止操作検出手段による停止操作の検出とに基づいて、前記図柄の変動表示を停止させる停止制御手段（例えば、リール停止制御処理）と、

前記内部当籤役決定手段により複数種の内部当籤役が決定された場合に、前記複数種の内部当籤役にそれぞれ対応する複数種の図柄の組合せの中なら、前記複数の表示列に跨って設定された判定ライン上に、優先して停止表示させる図柄の組合せを決定する優先停止図柄決定手段（例えば、引込優先順位取得処理）と、

前記優先停止図柄決定手段による優先して停止表示させる図柄の組合せの決定動作を制御するための演算処理を行う演算処理手段（例えば、メインCPU101）と、

前記演算処理手段による前記演算処理の実行に必要な情報が記憶された第1記憶手段（例えば、メインROM102）と、

10

前記演算処理手段による前記演算処理の実行に必要な情報が記憶される第2記憶手段（例えば、メインRAM103）と、を備え、

前記演算処理手段は、

前記演算処理手段による前記演算処理の実行時にデータが格納される汎用レジスタと、

前記演算処理手段による前記演算処理の実行時にデータが格納され、該格納されたデータにより前記第1記憶手段又は前記第2記憶手段内のアドレスの一部を指定可能な拡張レジスタ（例えば、Qレジスタ）と、を有し、

前記優先停止図柄決定手段により、優先して停止表示させる図柄の組合せを決定する処理において、

20

前記演算処理手段は、

前記拡張レジスタを用いて前記第2記憶手段内のアドレス指定を行うことができる所定の読み出し命令（例えば、「LDQ」命令）を実行することにより、前記第2記憶手段内に配置され且つ前記内部当籤役に対応する当籤フラグデータを格納する当籤フラグデータ格納領域（例えば、当り要求フラグ格納領域）のアドレスを指定する当籤フラグ格納領域指定手段（例えば、引込優先順位取得処理中のS686）と、

前記所定の読み出し命令を実行することにより、前記第2記憶手段内に配置され且つ前記判定ライン上に停止表示された図柄の組合せに対応する入賞フラグデータを格納する入賞フラグデータ格納領域（例えば、入賞作動フラグ格納領域）のアドレスを指定する入賞フラグ格納領域指定手段（例えば、引込優先順位取得処理中のS686）と、

30

前記当籤フラグデータとそれに対応する前記入賞フラグデータとの論理積演算を行う論理積演算手段（例えば、引込優先順位取得処理中のS686）と、

前記所定の読み出し命令を実行することにより、前記複数種の図柄の組合せ間における、停止表示させる図柄の組合せの優先順位を規定したデータテーブルを取得する優先順位データテーブル取得手段（例えば、引込優先順位取得処理中のS687）と、を有することを特徴とする遊技機。

【1514】

また、前記本発明の第24の遊技機では、前記優先停止図柄決定手段により、優先して停止表示させる図柄の組合せを決定する処理において、判定対象となる前記内部当籤役に対応する図柄の組合せに、現在判定対象中の特定の表示列（例えば、右リール3R）における前記判定ライン上の停止図柄が任意となる所定の図柄の組合せ（例えば、「ANY」役）が含まれる場合には、

40

前記演算処理手段は、前記当籤フラグ格納領域指定手段による前記当籤フラグデータ格納領域のアドレス指定処理、前記入賞フラグ格納領域指定手段による前記入賞フラグデータ格納領域のアドレス指定処理、及び、前記論理積演算手段による論理積演算処理が実行されないようにしてもよい。

【1515】

また、上記第13の課題を解決するために、本発明では、以下のような構成の第25の遊技機を提供する。

【1516】

50

遊技媒体の投入操作を検出する投入操作検出手段（例えば、BETスイッチ77）と、前記投入操作検出手段による投入操作の検出された後、遊技者による開始操作を検出する開始操作検出手段（例えば、スタートスイッチ79）と、前記開始操作検出手段による開始操作の検出に基づいて予め定められた確率で内部当籤役を決定する内部当籤役決定手段（例えば、内部抽籤処理）と、複数の表示列を含み、前記開始操作検出手段による開始操作の検出に基づいて、各表示列に設けられた図柄を変動表示する変動表示手段（例えば、3つのリール3L, 3C, 3R）と、遊技者による停止操作の検出を行う停止操作検出手段（例えば、ストップスイッチ基板80）と、前記内部当籤役決定手段の決定結果と前記停止操作検出手段による停止操作の検出とに基づいて、前記図柄の変動表示を停止させる停止制御手段（例えば、リール停止制御処理）と、イリーガルヒットエラーの発生の有無を判定するエラー検出手段（例えば、イリーガルヒットチェック処理）と、前記エラー検出手段による判定動作を制御するための演算処理を行う演算処理手段（例えば、メインCPU101）と、前記演算処理手段による前記演算処理の実行に必要な情報が記憶された第1記憶手段（例えば、メインROM102）と、前記演算処理手段による前記演算処理の実行に必要な情報が記憶される第2記憶手段（例えば、メインRAM103）と、を備え、前記演算処理手段は、前記演算処理手段による前記演算処理の実行時にデータが格納される汎用レジスタと、前記演算処理手段による前記演算処理の実行時にデータが格納され、該格納されたデータにより前記第2記憶手段内のアドレスの一部を指定可能な拡張レジスタ（例えば、Qレジスタ）と、を有し、前記エラー検出手段により、イリーガルヒットエラーの発生の有無を判定処理において、前記演算処理手段は、前記拡張レジスタを用いて前記第2記憶手段内のアドレス指定を行うことができる所定の読み出し命令（例えば、「LDQ」命令）を実行することにより、前記第2記憶手段内に配置され且つ前記複数の表示列に跨って設定された判定ライン上に停止表示された図柄の組合せに対応する入賞フラグデータを格納する入賞フラグデータ格納領域（例えば、入賞作動フラグ格納領域）のアドレスを指定する入賞フラグ格納領域指定手段（例えば、イリーガルヒットチェック処理中のS781）と、前記入賞フラグデータ格納領域に格納された前記入賞フラグデータと、それに対応する前記内部当籤役を示す当籤フラグデータとの論理積演算を行う論理積演算手段（例えば、イリーガルヒットチェック処理中のS784）と、前記論理積演算手段による演算結果に基づいて、イリーガルヒットエラーの発生の有無を判定するエラー判定手段（例えば、イリーガルヒットチェック処理中のS785）と、を有し、前記当籤フラグデータが格納される当籤フラグデータ格納領域（例えば、当り要求フラグ格納領域）の構成が、前記入賞フラグデータ格納領域の構成と同一であることを特徴とする遊技機。

【1517】

また、前記本発明の第25の遊技機では、前記拡張レジスタで指定可能な前記アドレスの一部が、前記アドレスを構成する上位側のアドレス値であるようにしてもよい。

【1518】

さらに、前記本発明の第25の遊技機では、前記演算処理手段は、前記エラー検出手段により、イリーガルヒットエラーがあると判定された場合には、エラー処理を行うエラー処理手段を有し、

10

20

30

40

50

前記エラー処理手段は、前記イリーガルヒットエラーを視認可能に報知するとともに、前記イリーガルヒットエラーが解除されるまで、遊技を停止するようにしてもよい。

【1519】

上記構成の本発明の第23～第25の遊技機によれば、主制御回路で管理する処理プログラムやテーブルなどの容量を削減して主制御回路のROMの空き容量を増やし、該増えた容量分のROMの空き領域を利用して遊技性を高めることができる。

【1520】

[第26の遊技機]

従来、上述した構成の遊技機において、遊技制御基板（主制御基板）から演出制御基板（副制御基板）にコマンドデータを送信する遊技機が知られている（例えば、特開2013-027655号公報及び特開2014-033751号公報参照）。

10

【1521】

ところで、近年、遊技性の多様化に伴い、副制御回路による演出制御において遊技性に係る処理が増える傾向がある。そのため、主制御回路から副制御回路に送信される通信データを改ざんする、「ゴト」と呼ばれる不正行為が発生しており、遊技店に損害を与えている。それに対して、従来、例えば上記特開2014-033751号公報等で提案されているように、主制御回路において送信データにゴト防止処理を施す対策も実施されている。しかしながら、このようなゴト防止処理の追加により、主制御回路のプログラム容量が圧迫されるという課題が発生している。

【1522】

本発明は、上記第14の課題を解決するためになされたものであり、本発明の第14の目的は、送信データに不正（ゴト）防止処理を施すことなく、不正行為を抑止するとともに、不正防止処理による主制御回路のプログラム容量の圧迫を無くすことが可能な遊技機を提供することである。

20

【1523】

上記第14の課題を解決するために、本発明では、以下のような構成の第26の遊技機を提供する。

【1524】

前記遊技動作に関する通信データを送信するデータ送信手段（例えば、主制御回路90）と、

30

前記通信データを作成する通信データ生成手段（例えば、通信データ格納処理）と、
前記通信データ生成手段による通信データの生成動作を制御するための演算処理を行う演算処理手段（例えば、メインCPU101）と、

前記演算処理手段による前記演算処理の実行に必要な情報が記憶された第1記憶手段（例えば、メインROM102）と、

前記演算処理手段による前記演算処理の実行に必要な情報が記憶される第2記憶手段（例えば、メインRAM103）と、を備え、

前記演算処理手段は、

前記演算処理手段による前記演算処理の実行時に複数種のデータがそれぞれ格納される複数の汎用レジスタを有し、

40

前記演算処理手段は、

前記通信データ生成手段による通信データの生成処理において、

前記通信データを構成する複数種の通信パラメータをそれぞれ前記複数の汎用レジスタに割り当て、

前記通信データを構成する複数種の通信パラメータのうち、前記通信データの種別に基づいて通信パラメータを、前記複数の汎用レジスタのうちの対応する汎用レジスタにセットし、

前記汎用レジスタにセットされた通信パラメータを、前記第2記憶手段内の所定の格納領域（通信データ一時格納領域）に格納し、

前記複数種の通信パラメータのうち、通信データの生成時に未使用の通信パラメータに対

50

応付けられた汎用レジスタに格納されているデータを通信パラメータとして前記所定の格納領域に格納し、

前記複数種の通信パラメータに応じて前記複数の汎用レジスタにセットされたデータに基づいて、前記通信データのサム値を生成する

ことを特徴とする遊技機。

【1525】

また、前記本発明の第26の遊技機では、前記通信データ生成手段は、前記第2記憶手段内の所定の格納領域に空きが無い場合には、前記複数の汎用レジスタにセットされた前記通信パラメータを前記第2記憶手段内の所定の格納領域に格納することなく通信データの生成処理を終了するようにしてもよい。

【1526】

上記構成の本発明の第26の遊技機によれば、送信データ（通信データ）に不正防止処理を施すことなく、不正行為を抑止することができるとともに、不正防止処理による主制御回路のプログラム容量の圧迫も無くすることができる。

【1527】

[第27の遊技機]

従来、上述した構成の遊技機において、主制御回路の制御により払い出されたメダルの枚数に応じて、クレジット数を更新する機能を備えた遊技機が知られている（例えば、特開2009-077977号公報参照）。

【1528】

ところで、従来、ART中やボーナス中の遊技では、メダルの払い出しが発生する頻度が高くなるが、この際、メダル1枚単位の払出間隔は一律（一定）で制御されることが多い。この場合、メダル払出期間において無駄な待ち時間が発生する。それゆえ、例えば、ARTのように長時間の遊技では、遊技期間が無駄に長くなり、遊技者にとって精神的負担となる可能性がある。

【1529】

本発明は、上記第15の課題を解決するためになされたものであり、本発明の第15の目的は、メダル払出期間において、無駄な待ち時間を減らし、遊技者の精神的負担を軽減することが可能な遊技機を提供することである。

【1530】

上記第15の課題を解決するために、本発明では、以下のような構成の第27の遊技機を提供する。

【1531】

遊技媒体の投入操作を検出する投入操作検出手段（例えば、BETスイッチ77）と、前記投入操作検出手段による投入操作の検出された後、遊技者による開始操作を検出する開始操作検出手段（例えば、スタートスイッチ79）と、

前記開始操作検出手段による開始操作の検出に基づいて予め定められた確率で内部当籤役を決定する内部当籤役決定手段（例えば、内部抽籤処理）と、

複数の表示列を含み、前記開始操作検出手段による開始操作の検出に基づいて、各表示列に設けられた図柄を変動表示する変動表示手段（例えば、3つのリール3L, 3C, 3R）と、

遊技者による停止操作の検出を行う停止操作検出手段（例えば、ストップスイッチ基板80）と、

前記内部当籤役決定手段の決定結果と前記停止操作検出手段による停止操作の検出とに基づいて、前記図柄の変動表示を停止させる停止制御手段（例えば、リール停止制御処理）と、

前記停止制御手段により前記複数の表示列の変動表示が停止された場合に、前記複数の表示列に跨って設定された判定ライン上に、前記遊技媒体の払い出しに係る内部当籤役に対応する図柄の組合せが停止表示されたか否かを判定する特典付与判定手段（例えば、入賞検索処理）と、

10

20

30

40

50

前記特典付与判定手段により前記遊技媒体の払い出しに係る内部当籤役に対応する図柄の組合せが前記判定ライン上に停止表示されたと判定された場合に、前記遊技媒体を払い出す遊技媒体払出手段（例えば、入賞チェック・メダル払出処理）と、前記遊技媒体を貯留する遊技媒体貯留手段（例えば、クレジット機能）と、前記遊技媒体払出手段による前記遊技媒体の払出動作を制御するための演算処理を行う演算処理手段（例えば、メインCPU101）と、を備え、

前記演算処理手段は、

前記特典付与判定手段により前記遊技媒体の払い出しに係る内部当籤役に対応する図柄の組合せが前記判定ライン上に停止表示されたと判定され、且つ、前記遊技媒体貯留手段に貯留されている前記遊技媒体の貯留数がある上限値未満である場合には、前記遊技媒体の貯留数に1を加算する遊技媒体加算手段（例えば、入賞チェック・メダル払出処理中のS805）と、

前記遊技媒体加算手段により前記遊技媒体の貯留数に前記遊技媒体が1加算された後、前記遊技媒体の払い出しに係る内部当籤役に対応する図柄の組合せが停止表示された際に付与される前記遊技媒体の全払出数の払い出しが終了したか否かを判定する払出終了判定手段（例えば、入賞チェック・メダル払出処理中のS807）と、

前記払出終了判定手段により前記遊技媒体の全払出数の払い出しが終了していないと判定された場合に、所定期間、遊技に関する操作が無効になるウェイトを発生させるウェイト発生手段（例えば、入賞チェック・メダル払出処理中のS808）と、を有する

ことを特徴とする遊技機。

【1532】

また、前記本発明の第27の遊技機では、前記演算処理手段は、所定期間で割込処理を実行する割込処理実行手段を有し、

前記ウェイト発生手段による遊技に関する操作が無効となる前記ウェイトでは、前記割込処理実行手段による割込処理が予め定められた回数実行されるようにしてもよい。

【1533】

上記構成の本発明の第27の遊技機によれば、メダル（遊技媒体）払出期間において、無駄な待ち時間を減らし、遊技者の精神的負担を軽減することができる。

【1534】

[第28及び第29の遊技機]

従来、上述した構成の遊技機において、内部当籤役やAT遊技を決定するための抽籤テーブルがROMに記憶された遊技機が知られている（例えば、特開2009-125459号公報参照）。

【1535】

ところで、上記特開2009-125459号公報に記載の遊技機では、AT遊技の抽籤テーブルや抽籤処理プログラムは、記憶容量に余裕のある、副制御基板に設けられたROMに保存されている。しかしながら、近年の遊技機業界に特有の理由から、AT遊技の抽籤に係るテーブル及びプログラムも主制御基板に設けられたROMに記憶する必要がある。このため、小容量に制限されている主制御基板のROM容量が圧迫されることになる。

【1536】

本発明は、上記第16の課題を解決するためになされたものであり、本発明の第16の目的は、主制御回路の処理で用いるデータの容量を削減し、主制御回路のROMの空き容量を増やすことが可能な遊技機を提供することである。

【1537】

上記第16の課題を解決するために、本発明では、以下のような構成の第28の遊技機を提供する。

【1538】

遊技者による開始操作を検出する開始操作検出手段（例えば、スタートスイッチ79）と、前記開始操作検出手段による開始操作の検出に基づいて予め定められた確率で内部当籤役を決定する内部当籤役決定手段（例えば、内部抽籤処理）と、

10

20

30

40

50

前記内部当籤役決定手段により決定された内部当籤役に基づいて、遊技者にとって有利な遊技状態を特典として付与するか否かを抽籤により決定する特典付与決定手段（例えば、C T 中 C T 抽籤処理）と、

前記特典付与決定手段により参照される抽籤テーブル（例えば、C T 中 C T 当籤抽籤テーブル）と、を備え、

前記抽籤テーブルでは、

抽籤対象となる前記内部当籤役の種別を指定する判定データ（例えば、判定ビット）と、

前記判定データで指定される抽籤対象の前記内部当籤役の抽籤値とが規定され、

抽籤対象外となる前記内部当籤役の抽籤値は規定されず、

当籤確率が 1 0 0 % 未満である抽籤対象の前記内部当籤役の抽籤値には 0 以外の値が規定され、当籤確率が 1 0 0 % である抽籤対象の前記内部当籤役の抽籤値には 0 が規定されている

10

ことを特徴とする遊技機。

【 1 5 3 9 】

また、前記本発明の第 2 8 の遊技機では、前記特典付与決定手段は、

ソフトラッチ乱数から 1 バイトの乱数値を取得し、

取得した前記乱数値と前記抽籤値とを用いて抽籤することにより、前記遊技者にとって有利な遊技状態を特典として付与するか否かを決定するようにしてもよい。

【 1 5 4 0 】

上記第 1 6 の課題を解決するために、本発明では、以下のような構成の第 2 9 の遊技機を提供する。

20

【 1 5 4 1 】

遊技者による開始操作を検出する開始操作検出手段（例えば、スタートスイッチ 7 9 ）と、前記開始操作検出手段による開始操作の検出に基づいて予め定められた確率で内部当籤役を決定する内部当籤役決定手段（例えば、内部抽籤処理）と、

前記内部当籤役決定手段により決定された内部当籤役に基づいて、遊技者にとって有利な遊技状態を特典として付与するか否かを抽籤により決定する特典付与決定手段（例えば、C T 中 C T 抽籤処理）と、

前記特典付与決定手段により参照され、前記内部当籤役の種別毎に設けられた抽籤テーブル（例えば、C T 中 C T 当籤抽籤テーブル）と、

30

現在決定されている内部当籤役に対応付けられた前記抽籤テーブルを選択するための抽籤テーブル選択テーブル（例えば、当籤役別テーブル選択相対テーブル）と、を備え、

前記抽籤テーブル選択テーブルでは、

前記内部当籤役の種別毎に、該種別の内部当籤役が抽籤対象であるか否かを判別可能であり且つ該種別の内部当籤役に対応付けられた前記抽籤テーブルの配置先を指定可能な選択値が規定され、

抽籤対象外となる前記内部当籤役の前記選択値には、前記特典付与決定手段による抽籤結果をハズレ扱いとする 0 が規定され、

抽籤対象となる前記内部当籤役の前記選択値には、0 以外のデータが規定されている

ことを特徴とする遊技機。

40

【 1 5 4 2 】

また、前記本発明の第 2 9 の遊技機では、前記抽籤テーブル選択テーブルの前記選択値は、選択する前記抽籤テーブルまでの相対値であり、該相対値が 1 バイトの値であるようにしてもよい。

【 1 5 4 3 】

上記構成の本発明の第 2 8 及び第 2 9 の遊技機によれば、主制御回路の処理で用いるデータの容量を削減し、主制御回路の R O M の空き容量を増やすことができる。

【 1 5 4 4 】

[第 3 0 の遊技機]

従来、上述した構成の遊技機では、主制御基板に搭載された R O M 内の決まった領域にブ

50

プログラム及びテーブルがそれぞれ配置されている（例えば、特開 2 0 1 2 - 1 1 0 6 3 5 号公報参照）。

【 1 5 4 5 】

ところで、上記特開 2 0 1 2 - 1 1 0 6 3 5 号公報に記載の遊技機のように、主制御基板の R O M に配置可能なプログラム及びテーブルは、遊技機業界の規則上、制限されており、主制御基板の R O M 内におけるプログラム及びテーブルの拡張性は著しく乏しい。

【 1 5 4 6 】

本発明は、上記第 1 7 の課題を解決するためになされたものであり、本発明の第 1 7 の目的は、主制御基板の R O M 内におけるプログラム及びテーブルの拡張性を高めることが可能な遊技機を提供することである。

10

【 1 5 4 7 】

上記第 1 7 の課題を解決するために、本発明では、以下のような構成の第 3 0 の遊技機を提供する。

【 1 5 4 8 】

遊技動作を制御するための演算処理を行う演算処理手段（例えば、メイン C P U 1 0 1 ）と、

前記演算処理手段による前記演算処理の実行に必要な情報が記憶された第 1 記憶手段（例えば、メイン R O M 1 0 2 ）と、

前記演算処理手段による前記演算処理の実行に必要な情報が記憶される第 2 記憶手段（例えば、メイン R A M 1 0 3 ）と、を備え、

20

前記第 1 記憶手段には、遊技者により実施される遊技の遊技性に関与する処理の実行に必要な情報が記憶された遊技用記憶領域（例えば、遊技用 R O M 領域）と、前記遊技用記憶領域とは異なる領域に配置され且つ遊技者により実施される遊技の遊技性に関与しない処理の実行に必要な情報が記憶された規定外記憶領域（例えば、規定外 R O M 領域）とが設けられている

ことを特徴とする遊技機。

【 1 5 4 9 】

また、前記本発明の第 3 0 の遊技機では、前記第 2 記憶手段には、遊技者により実施される遊技の遊技性に関与する処理の実行に必要な情報が一時的に記憶される遊技用作業領域及び遊技用スタック領域を含む遊技用一時記憶領域（例えば、遊技用 R A M 領域）と、前記遊技用一時記憶領域とは異なる領域に配置され、且つ、遊技者により実施される遊技の遊技性に関与しない処理の実行に必要な情報が一時的に記憶される規定外作業領域及び規定外スタック領域を含む規定外一時記憶領域（例えば、規定外 R A M 領域）とが設けられているようにしてもよい。

30

【 1 5 5 0 】

さらに、前記本発明の第 3 0 の遊技機では、前記演算処理手段は、専用レジスタとしてスタックポインタを有し、

前記演算処理手段は、

前記第 1 記憶手段の前記規定外記憶領域に記憶されたプログラムを実行する場合には、前記スタックポインタに前記第 2 記憶手段の規定外スタック領域のアドレスを設定し、

40

前記第 1 記憶手段の前記規定外記憶領域に記憶されたプログラムを終了する場合には、前記規定外記憶領域に記憶されたプログラムを実行するときに退避させた前記第 2 記憶手段の前記遊技用スタック領域のアドレスを前記スタックポインタに設定して、前記第 1 記憶領域の前記遊技用記憶領域に記憶されたプログラムに戻すようにしてもよい。

【 1 5 5 1 】

上記構成の本発明の第 3 0 の遊技機によれば、主制御基板の R O M 内におけるプログラム及びテーブルの拡張性を高めることができる。

【 1 5 5 2 】

[第 3 1 の遊技機]

従来、上述した構成の遊技機において、主制御回路で決定された内部当籤役の情報を副制

50

御回路の演出制御で用いるために、内部当籤役を、その種別をグループ単位でまとめるサブフラグに変換する機能を備えた遊技機が知られている（例えば、特開 2010-051471 号公報参照）。

【1553】

上記特開 2010-051471 号公報に記載の遊技機では、内部当籤役をサブフラグに変換するための変換テーブルや変換処理プログラムは、記憶容量に余裕のある、副制御基板に設けられた ROM に保存されている。しかしながら、近年の遊技機業界に特有の理由から、主制御回路で決定された内部当籤役の情報を副制御回路に送信できなくなり、サブフラグに係る変換テーブルや変換処理プログラムも主制御基板に設けられた ROM に記憶する必要がある。この場合、これらのテーブルやプログラムにより、小容量に制限されている主制御基板の ROM 容量が圧迫されることになる。そこで、サブフラグに係る変換テーブルや変換処理プログラムによる主制御基板の ROM 容量への圧迫を抑制するとともに、遊技機の機種、企画、仕様等の変更への対応を効率良く行える技術の開発が求められている。

10

【1554】

本発明は、上記第 18 の課題を解決するためになされたものであり、本発明の第 18 の目的は、主制御基板の ROM 容量の圧迫を抑制するとともに、遊技機の機種、企画、仕様等の変更への対応を効率良く行えることが可能な遊技機を提供することである。

【1555】

上記第 18 の課題を解決するために、本発明では、以下のような構成の第 31 の遊技機を提供する。

20

【1556】

遊技者による開始操作を検出する開始操作検出手段（例えば、スタートスイッチ 79）と、前記開始操作検出手段による開始操作の検出に基づいて予め定められた確率で内部当籤役を決定する内部当籤役決定手段（例えば、内部抽籤処理）と、

前記内部当籤役決定手段により決定された内部当籤役を、変換テーブルを参照して、第 1 のサブフラグ（例えば、サブフラグ）に変換する第 1 サブフラグ変換手段（例えば、サブフラグ変換処理）と、

前記第 1 サブフラグ変換手段による変換処理で得られた前記第 1 のサブフラグを、抽籤テーブル（例えば、各種フラグ変換抽籤テーブル）を参照して、抽籤により第 2 のサブフラグ（例えば、サブフラグ EX）に変換する第 2 サブフラグ変換手段（例えば、フラグ変換処理）と、を備え、

30

前記変換テーブルでは、前記内部当籤役と、前記内部当籤役に対応付けられた制御ステータスデータと、前記第 1 のサブフラグとの対応関係が規定され、同じ種別の前記第 1 のサブフラグに対して、同じ値の前記制御ステータスデータが割り付けられ、

前記第 2 サブフラグ変換手段による抽籤は、前記制御ステータスデータに基づいて行われることを特徴とする遊技機。

【1557】

また、前記本発明の遊技機では、前記遊技動作に関するコマンドデータを送信するデータ送信手段を備え、

40

前記データ送信手段は、前記開始操作検出手段が開始操作を検出したことを契機として、前記第 1 のサブフラグをコマンドデータの通信パラメータとして送信するようにしてもよい。

【1558】

上記構成の本発明の第 31 の遊技機によれば、主制御基板の ROM 容量の圧迫を抑制するとともに、遊技機の機種、企画、仕様等の変更への対応を効率良く行えることができる。

【符号の説明】

【1559】

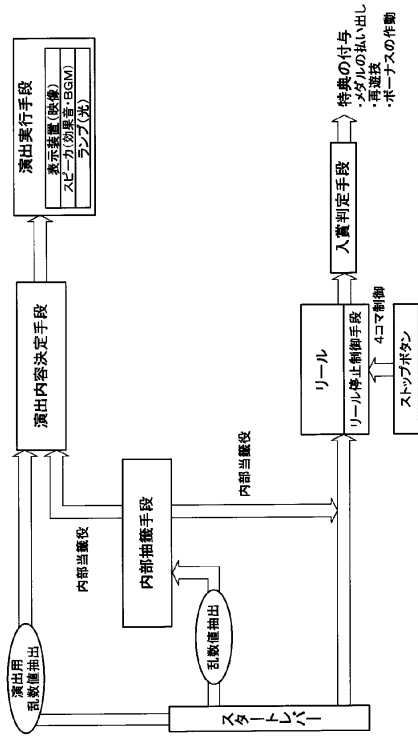
1 ... パチスロ、3L, 3C, 3R ... リール、4 ... リール表示窓、6 ... 情報表示器、11 ... 表示装置、17L, 17C, 17R ... ストップボタン、18 ... サブ表示装置、71 ... 主制

50

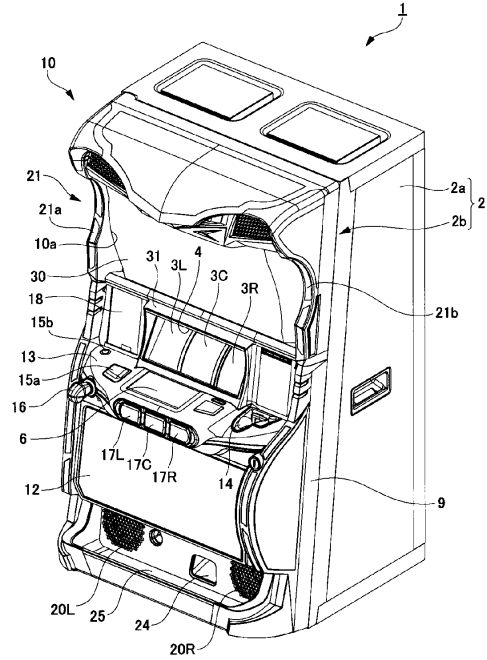
御基板、72...副制御基板、90...主制御回路、91...マイクロプロセッサ、101...メインCPU、102...メインROM、103...メインRAM、107...演算回路、114...第1シリアル通信回路、115...第2シリアル通信回路、200...副制御回路、201...サブCPU201、301...第1インターフェースボード、302...第2インターフェースボード

【図面】

【図1】



【図2】



10

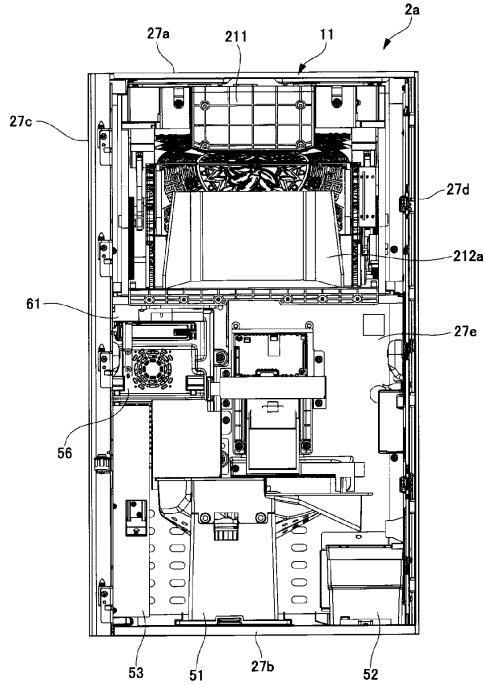
20

30

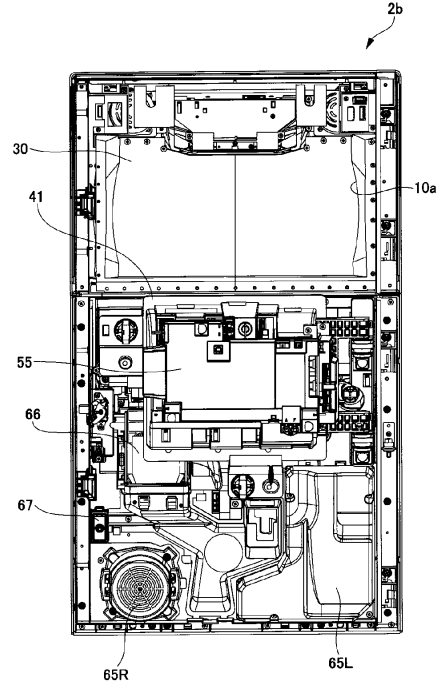
40

50

【図3】



【図4】

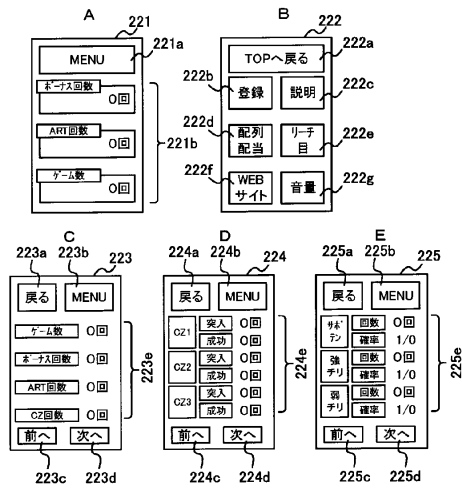


10

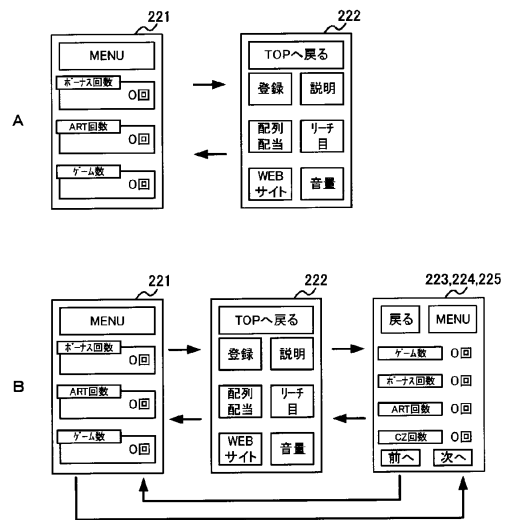
20

【図5】

サブ表示装置の表示画面



【図6】

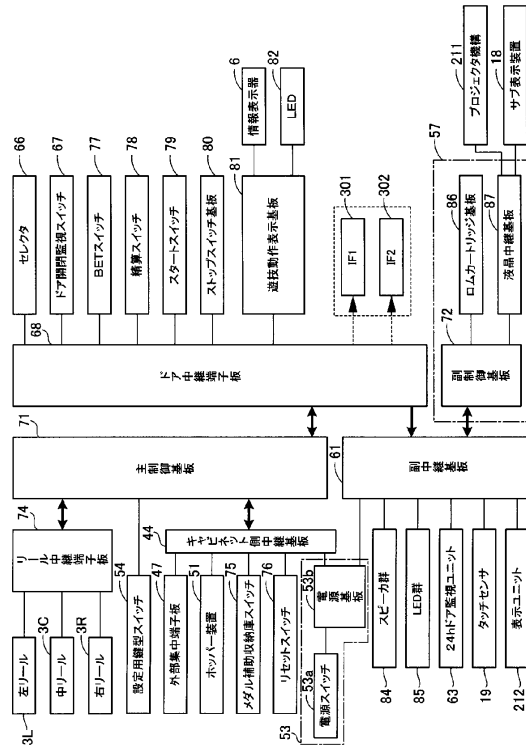


30

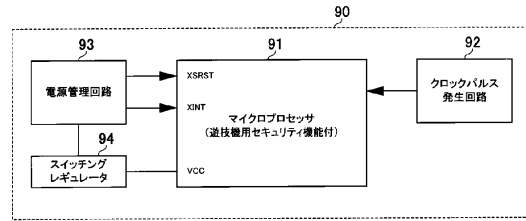
40

50

【図 7】



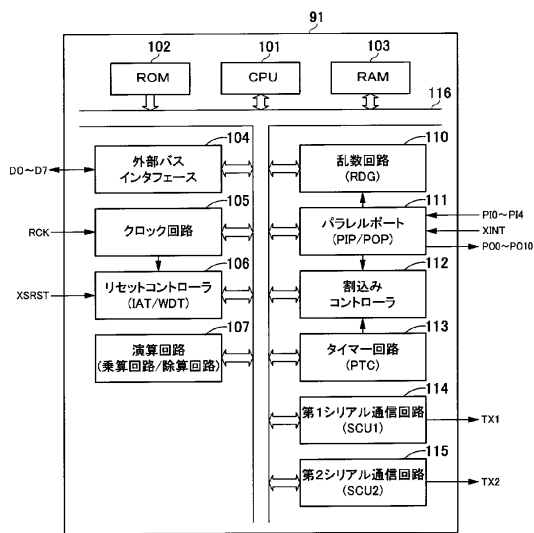
【図 8】



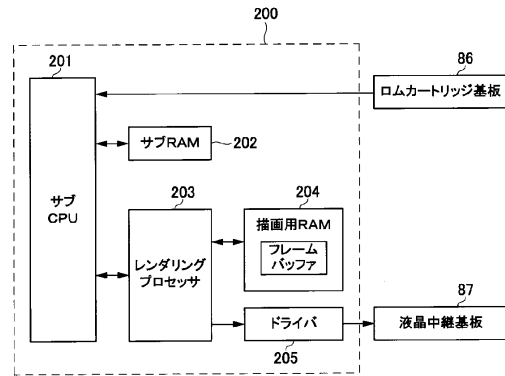
10

20

【図 9】



【図 10】

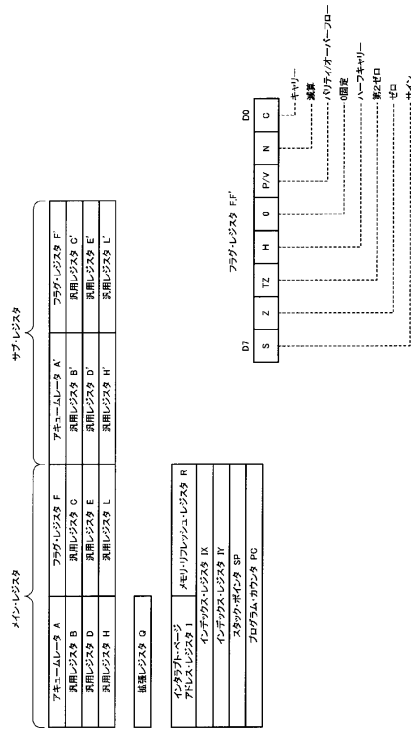


30

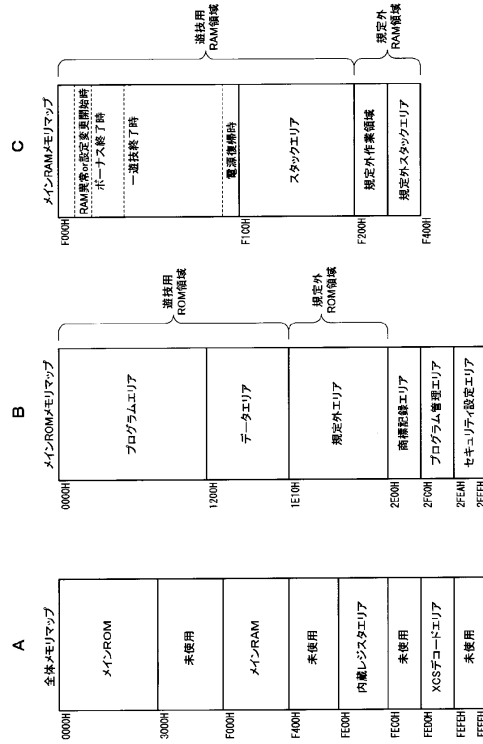
40

50

【図 1 1】



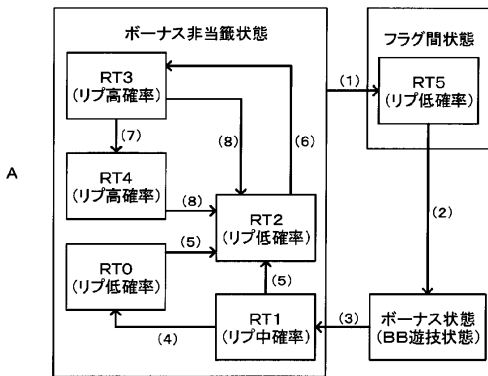
【図 1 2】



10

20

【図 1 3】

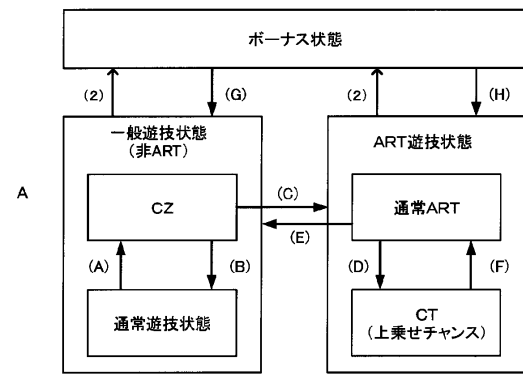


A

移行条件	内容
(1)	F.BB1.2に内部当籤
(2)	F.BB1.2入賞(C.BB1.2が表示)
(3)	ボーナス終了(216枚を超える払出)
(4)	20ゲーム経過
(5)	ベルこぼし目表示
(6)	RT3移行リプレイ表示
(7)	RT4移行リプレイ表示
(8)	ベルこぼし目又はRT2移行リプレイ表示

B

【図 1 4】



A

移行条件	内容
(A)	通常遊技状態中のCZ抽籤に当籤
(B)	CZ中のART抽籤に非当籤
(C)	CZ中のART抽籤に当籤
(D)	通常ART中のCT抽籤に当籤
(E)	通常ART終了
(F)	CT終了
(G)	ボーナス状態中のART抽籤に非当籤
(H)	ボーナス状態中のART抽籤に当籤

B

30

40

50

【図 2 3】

当職役	図柄組合せ					
	C 強 2 枚 0 2	C 強 2 枚 0 3	C 強 2 枚 0 5	C 強 2 枚 0 6	C 強 2 枚 0 7	C 強 2 枚 0 9
F_BB1						
F_BB2						
F_チリッ						
F_確チリッ						
F_1確チリッ						
F_リーチ目リッA						
F_リーチ目リッB						
F_リーチ目リッC						
F_リーチ目リッD						
F_維持リッA						
F_維持リッB						
F_維持リッ_1st						
F_維持リッ_2nd						
F_維持リッ_3rd						
F_RT3リッ_1st						
F_RT3リッ_213						
F_RT3リッ_231						
F_RT3リッ_3rd						
F_RT4リッ_123						
F_RT4リッ_132						
F_RT4リッ_2nd						
F_RT4リッ_3rd						
F_3択へル_1st						
F_3択へル_2nd						
F_3択へル_3rd						
F_共通へル						
F_サホ1						
F_サホ2						
F_弱チリ						
F_強チリ1	○	○	○	○	○	○
F_強チリ2	○	○	○	○	○	○
F_特殊1						
F_特殊2						
F_特殊3						
F_RB役1						
F_RB役2						
F_RB役3						
F_RB役4	○	○	○	○	○	○

【図 2 4】

内部当職役と停止図柄組合せ(略称)との対応関係

当職役	停止操作	
	押し順 正解	押し順 不正解
F_チリッ	チリッ(否3連)	リプレイ
F_確チリッ	チリッ(3連)	リプレイ
F_1確チリッ	チリッ(3連)	リプレイ
F_リーチ目リッA	リーチ目リッ	リプレイ
F_リーチ目リッB	リーチ目リッ	リプレイ
F_リーチ目リッC	リーチ目リッ	リプレイ
F_リーチ目リッD	リーチ目リッ	リプレイ
F_維持リッA		リプレイ
F_維持リッB		リプレイ
F_維持リッ_1st	リプレイ	RT2移行リッ
F_維持リッ_2nd	リプレイ	RT2移行リッ
F_維持リッ_3rd	リプレイ	RT2移行リッ
F_RT3リッ_1st	RT3移行リッ	リプレイ
F_RT3リッ_213	RT3移行リッ	リプレイ
F_RT3リッ_231	RT3移行リッ	リプレイ
F_RT3リッ_3rd	RT3移行リッ	リプレイ
F_RT4リッ_123	RT4移行リッ	リプレイ
F_RT4リッ_132	RT4移行リッ	リプレイ
F_RT4リッ_2nd	RT4移行リッ	リプレイ
F_RT4リッ_3rd	RT4移行リッ	リプレイ
F_3択へル_1st	ベル	ベルこぼし目 or 1枚出目
F_3択へル_2nd	ベル	ベルこぼし目 or 1枚出目
F_3択へル_3rd	ベル	ベルこぼし目 or 1枚出目
F_共通へル		ベル
F_サホ1		サボテン
F_サホ2		サボテン
F_弱チリ		弱チエリー
F_強チリ1		強チエリー
F_強チリ2		強チエリー

10

20

【図 2 5】

内部当職役と停止図柄組合せとの対応関係

当職役	BB(C, BB1) の成立可否	当職役	BB(C, BB2) の成立可否
はずれ(F_BB1)	○	はずれ(F_BB2)	○
F_BB1+F_チリッ	×	F_BB2+F_チリッ	×
F_BB1+F_確チリッ	×	F_BB2+F_確チリッ	×
F_BB1+F_1確チリッ	×	F_BB2+F_1確チリッ	×
F_BB1+F_リーチ目リッA	×	F_BB2+F_リーチ目リッA	×
F_BB1+F_リーチ目リッB	×	F_BB2+F_リーチ目リッB	×
F_BB1+F_リーチ目リッC	×	F_BB2+F_リーチ目リッC	×
F_BB1+F_リーチ目リッD	×	F_BB2+F_リーチ目リッD	×
F_BB1+F_維持リッA	×	F_BB2+F_維持リッA	×
F_BB1+F_維持リッB	×	F_BB2+F_維持リッB	×
F_BB1+F_維持リッ_1st	×	F_BB2+F_維持リッ_1st	×
F_BB1+F_維持リッ_2nd	×	F_BB2+F_維持リッ_2nd	×
F_BB1+F_維持リッ_3rd	×	F_BB2+F_維持リッ_3rd	×
F_BB1+F_RT3リッ_1st	×	F_BB2+F_RT3リッ_1st	×
F_BB1+F_RT3リッ_213	×	F_BB2+F_RT3リッ_213	×
F_BB1+F_RT3リッ_231	×	F_BB2+F_RT3リッ_231	×
F_BB1+F_RT3リッ_3rd	×	F_BB2+F_RT3リッ_3rd	×
F_BB1+F_RT4リッ_123	×	F_BB2+F_RT4リッ_123	×
F_BB1+F_RT4リッ_132	×	F_BB2+F_RT4リッ_132	×
F_BB1+F_RT4リッ_2nd	×	F_BB2+F_RT4リッ_2nd	×
F_BB1+F_RT4リッ_3rd	×	F_BB2+F_RT4リッ_3rd	×
F_BB1+F_3択へル_1st	×	F_BB2+F_3択へル_1st	×
F_BB1+F_3択へル_2nd	×	F_BB2+F_3択へル_2nd	×
F_BB1+F_3択へル_3rd	×	F_BB2+F_3択へル_3rd	×
F_BB1+F_共通へル	×	F_BB2+F_共通へル	×
F_BB1+F_サホ1	×	F_BB2+F_サホ1	×
F_BB1+F_サホ2	×	F_BB2+F_サホ2	×
F_BB1+F_弱チリ	×	F_BB2+F_弱チリ	×
F_BB1+F_強チリ1	×	F_BB2+F_強チリ1	×
F_BB1+F_強チリ2	×	F_BB2+F_強チリ2	×
F_BB1+F_特殊1	○	F_BB2+F_特殊1	○
F_BB1+F_特殊2	○	F_BB2+F_特殊2	○
F_BB1+F_特殊3	○	F_BB2+F_特殊3	○

【図 2 6】

ルール停止初期設定テーブル

内部当職役 (小役 当職番号)	引込優先順位 テーブル 番号	引込優先順位 テーブル番号	停止テーブル 番号	遊技状態	備考
0	—	00	0	全遊技状態	はずれ
1	0	—	1	全遊技状態	F_チリッ
2	1	—	2	全遊技状態	F_確チリッ
3	1	—	3	全遊技状態	F_1確チリッ
4	2	—	4	一般遊技中	F_リーチ目リッA
	3	—		フラグ開状態	
5	4	—	5	一般遊技中	F_リーチ目リッB
	3	—		フラグ開状態	
6	4	—	6	一般遊技中	F_リーチ目リッC
	3	—		フラグ開状態	
7	4	—	7	一般遊技中	F_リーチ目リッD
	3	—		フラグ開状態	
8	0	—	8	全遊技状態	F_維持リッA
9	5	—	9	全遊技状態	F_維持リッB
10	6	—	10	全遊技状態	F_維持リッ_1st
11	7	—	11	全遊技状態	F_維持リッ_2nd
12	8	—	12	全遊技状態	F_維持リッ_3rd
13	9	—	13	全遊技状態	F_RT3リッ_1st
14	10	—	14	全遊技状態	F_RT3リッ_213
15	11	—	15	全遊技状態	F_RT3リッ_231
16	12	—	16	全遊技状態	F_RT3リッ_3rd
17	13	—	17	全遊技状態	F_RT4リッ_123
18	14	—	18	全遊技状態	F_RT4リッ_132
19	15	—	19	全遊技状態	F_RT4リッ_2nd
20	16	—	20	全遊技状態	F_RT4リッ_3rd
21	17	—	21	全遊技状態	F_3択へル_1st
22	18	—	22	全遊技状態	F_3択へル_2nd
23	19	—	23	一般遊技中	F_3択へル_3rd
	20	—		フラグ開状態	
24	21	—	24	全遊技状態	F_共通へル
25	21	—	25	全遊技状態	F_サホ1
26	21	—	26	全遊技状態	F_サホ2
27	21	—	27	全遊技状態	F_弱チリ
28	21	—	28	全遊技状態	F_強チリ1
29	21	—	29	全遊技状態	F_強チリ2
30	—	00	30	全遊技状態	F_特殊1
31	—	00	31	全遊技状態	F_特殊2
32	0	—	32	全遊技状態	F_特殊3
33	22	—	33	全遊技状態	F_RB役1
34	22	—	34	全遊技状態	F_RB役2
35	23	—	35	全遊技状態	F_RB役3
36	—	00	36	全遊技状態	F_RB役4

30

40

50

【 図 3 1 】

持越役格納領域			
格納領域種別	データ		内容
持越役 格納領域	ビット7	0	未使用
	ビット6	0	未使用
	ビット5	0	未使用
	ビット4	0	未使用
	ビット3	0	未使用
	ビット2	0	未使用
	ビット1	0 or 1	BB2
	ビット0	0 or 1	BB1

【 図 3 2 】

遊技状態フラグ格納領域			
格納領域種別	データ		内容
遊技状態 フラグ 格納領域	ビット7	0	未使用
	ビット6	0 or 1	RT5状態
	ビット5	0 or 1	RT4状態
	ビット4	0 or 1	RT3状態
	ビット3	0 or 1	RT2状態
	ビット2	0 or 1	RT1状態
	ビット1	0 or 1	RB遊技状態
	ビット0	0 or 1	BB遊技状態

【 図 3 3 】

作動ストップボタン格納領域			
格納領域種別	データ		内容
作動ストップボタン 格納領域	ビット7	0	未使用
	ビット6	0 or 1	右ストップボタン操作有効
	ビット5	0 or 1	中ストップボタン操作有効
	ビット4	0 or 1	左ストップボタン操作有効
	ビット3	0	未使用
	ビット2	0 or 1	右ストップボタン操作
	ビット1	0 or 1	中ストップボタン操作
	ビット0	0 or 1	左ストップボタン操作

※ビット0～2は、「0」で操作なし、「1」で操作あり
 ※ビット4～6は、「0」で無効、「1」で有効

【 図 3 4 】

押下順序格納領域			
格納領域種別	データ		内容
押下順序格納領域	ビット7	0	未使用
	ビット6	0	未使用
	ビット5	0 or 1	右→中→左
	ビット4	0 or 1	右→左→中
	ビット3	0 or 1	中→右→左
	ビット2	0 or 1	中→左→右
	ビット1	0 or 1	左→右→中
	ビット0	0 or 1	左→中→右

※ビット0～5は、「0」で無効、「1」で有効

10

【 図 3 5 】

図柄コード格納領域			
格納領域種別	データ		内容
図柄コード 格納領域 11	ビット7	0 or 1	R_RT移行目A.01
			R_RT移行目A.02
			R_RT移行目A.03
			R_RT移行目A.04
	ビット6	0 or 1	R_RT移行目B.01
			R_RT移行目B.02
			未使用
			未使用
	ビット5	0	未使用
			未使用
未使用			
未使用			
ビット4	0	未使用	
		未使用	
		未使用	
		未使用	
ビット3	0	未使用	
		未使用	
		未使用	
		未使用	
ビット2	0	未使用	
		未使用	
		未使用	
		未使用	
ビット1	0	未使用	
		未使用	
		未使用	
		未使用	
ビット0	0	未使用	
		未使用	
		未使用	
		未使用	
図柄コード 格納領域 0	ビット7	0 or 1	C_弱2枚A.01
			C_弱2枚B.01
			C_弱2枚B.02
			C_弱2枚B.03
	ビット6	0 or 1	C_強2枚A.01
			C_強2枚A.02
			C_強2枚A.03
			C_強2枚B.01
	ビット5	0 or 1	C_強2枚B.02
			C_強2枚C.01
			C_強2枚C.02
			C_強2枚C.03
	ビット4	0 or 1	C_強2枚C.04
			C_強2枚C.05
			C_強2枚C.06
			C_強2枚C.07
ビット3	0 or 1	C_強2枚C.08	
		C_強2枚C.09	
		未使用	
		未使用	
ビット2	0	未使用	
		未使用	
		未使用	
		未使用	
ビット1	0	未使用	
		未使用	
		未使用	
		未使用	
ビット0	0	未使用	
		未使用	
		未使用	
		未使用	

【 図 3 6 】

内部当籤役とサブフラグとの対応表(その1)			
内部当籤役 (小役当籤番号)	サブフラグ (フラグデータ)	サブフラグEX (フラグデータ)	サブフラグD (フラグデータ)
はずれ(00)	はずれ(00)	はずれ(00)	はずれ(00)
F_チリリブ(01)	2連チリリブ(01)	リプレイ(01)	はずれ(00)
F_確チリリブ(02)	3連チリリブA(02)	リプレイ(01), 確定役(06)	はずれ(00), 確定役(04), リーチ目リブ(06)
F_1確チリリブ(03)	3連チリリブB(03)	3連チリリブ(07)	3連チリリブ(05)
F_リーチ目リブA(04)	リーチ目リブ1(04)	リプレイ(01), 確定役(06)	はずれ(00), 確定役(04), リーチ目リブ(06)
F_リーチ目リブB(05)	リーチ目リブ2(05)	リプレイ(01), 確定役(06)	はずれ(00), 確定役(04), リーチ目リブ(06)
F_リーチ目リブC(06)	リーチ目リブ3(06)	リプレイ(01), 確定役(06)	はずれ(00), 確定役(04), リーチ目リブ(06)
F_リーチ目リブD(07)	リーチ目リブ4(07)	リプレイ(01), 確定役(06)	はずれ(00), 確定役(04), リーチ目リブ(06)
F_維持リブA(08)	リプレイ(08)	リプレイ(01)	はずれ(00)
F_維持リブB(09)	リプレイ(08)	リプレイ(01)	はずれ(00)
F_維持リブ_1st(10)	押し順リブ1(09)	リプレイ(01)	はずれ(00)
F_維持リブ_2nd(11)	押し順リブ2(10)	リプレイ(01)	はずれ(00)
F_維持リブ_3rd(12)	押し順リブ3(11)	リプレイ(01)	はずれ(00)
F_RT3リブ_1st(13)	押し順リブ2(10)	リプレイ(01)	はずれ(00)
F_RT3リブ_213(14)	押し順リブ2(10)	リプレイ(01)	はずれ(00)
F_RT3リブ_231(15)	押し順リブ2(10)	リプレイ(01)	はずれ(00)
F_RT3リブ_3rd(16)	押し順リブ2(10)	リプレイ(01)	はずれ(00)
F_RT4リブ_123(17)	押し順リブ3(11)	リプレイ(01)	はずれ(00)
F_RT4リブ_132(18)	押し順リブ3(11)	リプレイ(01)	はずれ(00)
F_RT4リブ_2nd(19)	押し順リブ3(11)	リプレイ(01)	はずれ(00)
F_RT4リブ_3rd(20)	押し順リブ3(11)	リプレイ(01)	はずれ(00)
F_3択ベル_1st(21)	押し順ベル(12)	ベル(02)	はずれ(00)
F_3択ベル_2nd(22)	押し順ベル(12)	ベル(02)	はずれ(00)
F_3択ベル_3rd(23)	押し順ベル(12)	ベル(02)	はずれ(00)
F_共通ベル(24)	共通ベル(13)	ベル(02)	はずれ(00)
F_サボ1(25)	サボテン(14)	サボテン(03)	サボテン(01)
F_サボ2(26)	サボテン(14)	サボテン(03)	サボテン(01)
F_弱チリ(27)	弱チェリー(15)	弱チェリー(04)	弱チェリー(02)
F_強チリ1(28)	強チェリー(16)	強チェリー(05)	強チェリー(03)
F_強チリ2(29)	強チェリー(16)	強チェリー(05)	強チェリー(03)
F_特殊1(30)	リーチ目1(17)	はずれ(00)	はずれ(00)
F_特殊2(31)	リーチ目1(17)	はずれ(00)	はずれ(00)
F_特殊3(32)	リーチ目2(18)	はずれ(00)	はずれ(00)

20

30

40

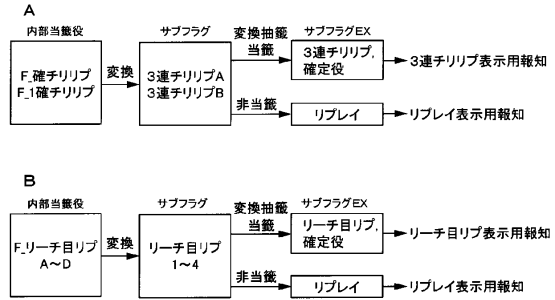
50

【 図 3 7 】

内部当籤役とサブフラグとの対応表(その2)

内部当籤役 (特賞当籤番号)	サブフラグ
F.BB1(01)	BB
F.BB2(02)	

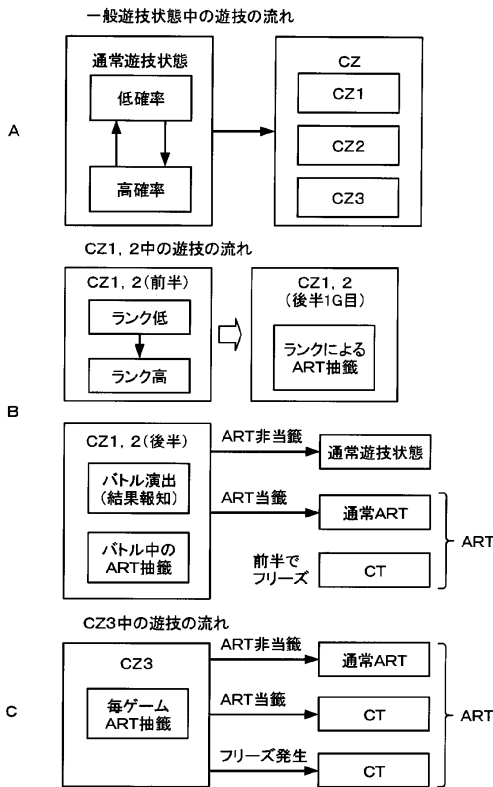
【 図 3 8 】



※ 変換抽籤はRT4中のみ行う

10

【 図 3 9 】



【 図 4 0 】

通常中高確率抽籤テーブル(毎ゲーム参照)

現在 の状態	移行後 の状態	当籤役				
		共通ベル	サポテン	弱チェリー	強チェリー	それ以外
低確率中	低確率	極高	極高	高	極高	確定
	高確率	極低	低	中	低	0
高確率中	低確率	0	0	0	0	低
	高確率	確定	確定	確定	確定	極高

通常中高確率抽籤テーブル(状況によって参照)

移行後 の状態	状況		
	設定変更時	ボーナス 終了時	CZ, ART 終了時
低確率	極高	0	極高
高確率	低	確定	低

確率分母256
※ 0<極々低<極低<低<中<高<極高<極々高<確定(=100%)

20

30

40

50

【 図 4 1 】

CZ抽籤テーブル(初当たり用)

現在の状態	内容	当籤役			
		サブポテン	弱チェリー	強チェリー	BB
A 低確率中	非当籤	極高	確定	高	高
	CZ1当籤	極低	0	低	低
	CZ2当籤	極低	0	極低	極低
	CZ3当籤	0	0	0	0
高確率中	非当籤	高	極高	中	中
	CZ1当籤	中	0	中	中
	CZ2当籤	低	0	低	低
	CZ3当籤	極低	極低	極低	極低

CZ抽籤テーブル(引き戻し用)

状況	内容	抽籤値
B CZ失敗時 ART終了時	非当籤	極高
	CZ1当籤	低
	CZ2当籤	極低
	CZ3当籤	0

※ 確率分母256

【 図 4 2 】

CZ1中モードアップ抽籤テーブル

現在のモード	内容	当籤役			
		リプレイ	共通ベル	弱チェリー サブポテン	強チェリー
モード1 モード2	非当籤	極高	0	0	0
	モード1UP	低	確定	極高	0
	モード2UP	0	0	極低	0
	モード3UP	0	0	極低	極高
	モード4UP	0	0	0	0
	モード5UP	0	0	0	0
	モード6UP	0	0	極低	低
モード3	モード6UPフリーズ発生	0	0	0	0
	非当籤	極高	0	0	0
	モード1UP	低	確定	極高	0
	モード2UP	0	0	極低	0
	モード3UP	0	0	極低	極高
	モード4UP	0	0	0	0
	モード5UP	0	0	0	0
モード4	モード6UP	0	0	極低	低
	モード6UPフリーズ発生	0	0	0	0
	非当籤	極高	0	0	0
	モード1UP	低	確定	極高	0
	モード2UP	0	0	極低	0
	モード3UP	0	0	極低	極高
	モード4UP	0	0	0	0
モード5 モード6	モード5UP	0	0	0	0
	モード6UP	0	0	極低	低
	モード6UPフリーズ発生	0	0	0	0
	非当籤	極高	0	0	0
	モード1UP	低	確定	極高	0
	モード2UP	0	0	極低	0
	モード3UP	0	0	極低	極高
CZ2開始 2G以内	モード4UP	0	0	0	0
	モード5UP	0	0	0	0
	モード6UP	0	0	0	0
	モード6UPフリーズ発生	0	0	0	0
	非当籤	極高	0	0	0
	モード1UP	低	確定	極高	0
	モード2UP	0	0	低	0

※ 確率分母256

10

20

【 図 4 3 】

CZ2中ポイント抽籤テーブル

現在のポイント	内容	当籤役			
		リプレイ	共通ベル	弱チェリー サブポテン	強チェリー
ポイント 0~9	非当籤	0	0	0	0
	ポイント1UP	極高	中	0	0
	ポイント2UP	0	中	高	0
	ポイント3UP	極低	極低	低	0
	ポイント4UP	極低	極低	極低	高
	ポイント5UP	極低	極低	極低	中
	ポイント10UP	0	0	極低	0
CZ2開始 2G以内	ポイント10UPフリーズ発生	0	0	0	0
	非当籤	0	0	0	0
	ポイント1UP	極高	中	0	0
	ポイント2UP	0	中	高	0
	ポイント3UP	極低	極低	低	0
	ポイント4UP	極低	極低	極低	0
	ポイント5UP	0	0	0	0
ポイント10UP	0	0	0	0	
ポイント10UPフリーズ発生	0	0	0	確定	

※ 確率分母256

【 図 4 4 】

CZ中ART抽籤テーブル(CZ1用)		現在のモード											
内容	モード1	モード2	モード3	モード4	モード5	モード6	モード1	モード2	モード3	モード4	モード5	モード6	モード10
非当籤	極高	極高	極高	極高	極高	極高	極高	極高	極高	極高	極高	極高	極高
ART当籤	極低	極低	極低	極低	極低	極低	極低	極低	極低	極低	極低	極低	極低

CZ中ART抽籤テーブル(CZ2用)		現在のモード											
内容	モード1	モード2	モード3	モード4	モード5	モード6	モード1	モード2	モード3	モード4	モード5	モード6	モード10
非当籤	極高	極高	極高	極高	極高	極高	極高	極高	極高	極高	極高	極高	極高
ART当籤	極低	極低	極低	極低	極低	極低	極低	極低	極低	極低	極低	極低	極低

CZ中ART抽籤テーブル(CZ1, 2共通 後半ハハル中用)		当籤役		
内容	弱チェリー サブポテン	強チェリー	BB	中
非当籤	極高	極高	極高	中
ART当籤	極低	極低	極低	中

※ 確率分母256

30

40

50

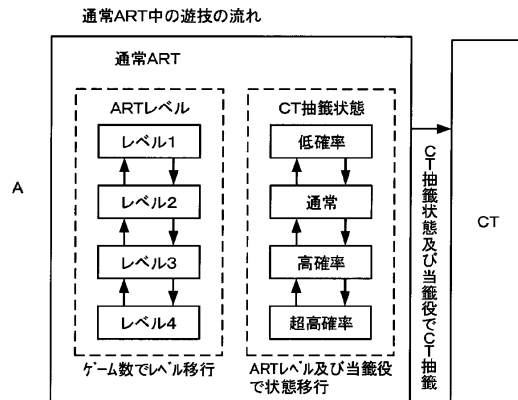
【図 4 5】

CZ中ART抽籤テーブル(CZ3用)

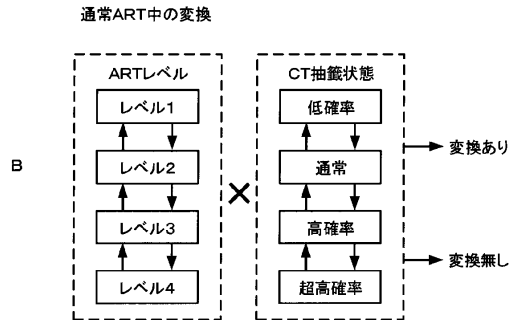
状態	内容	当籤役				
		リプレイ	共通ベル	弱チャエリー サポテン	強チャエリー	それ以外
CZ3開始 2G以内	非当籤	極高	確定	中	0	確定
	ART当籤+通常CT	0	0	0	0	0
	ART当籤+高確CT	0	0	0	0	0
	フリーズ+通常CT	低	0	中	確定	0
CZ3開始 3G~17 G	非当籤	極高	極高	中	0	極高
	ART当籤+通常CT	極低	低	中	中	極低
	ART当籤+高確CT	0	0	0	0	0
	フリーズ+通常CT	極低	0	極低	中	0
	フリーズ+高確CT	0	0	0	0	0

※ 確率分母256

【図 4 6】



10



20

【図 4 7】

ART中フラグ変換抽籤テーブル(第1段階用)

内容	当籤役	
	F. 確りリプレイ or F. 1 確りリプレイ	種々高
変換無し	種々高	
変換あり(仮)	種々低	

※ 確率分母256

ART中フラグ変換抽籤テーブル(第2段階用(第1段階で変換あり(仮)に当籤))

ARTレベル	CT抽籤状態	内容	当籤役	
			F. 確りリプレイ or F. 1 確りリプレイ	F. リーチ目リプレイA~D
初回 (一度CTに当籤 するまで)	低確率	変換無し	0	極高
		変換有り	確定	低
	通常	変換無し	0	極高
		変換有り	確定	低
高確率	変換無し	0	0	
	変換有り	確定	確定	
ARTレベル1	低確率	変換無し	種々高	種々高
		変換有り	種々低	種々低
	通常	変換無し	種々高	種々高
		変換有り	種々低	種々低
高確率	変換無し	種々高	高	
	変換有り	種々低	中	
超確率	変換無し	種々高	0	
	変換有り	種々低	確定	
ARTレベル 2~4	低確率	変換無し	種々高	種々高
		変換有り	低	種々低
	通常	変換無し	種々高	種々高
		変換有り	低	種々低
高確率	変換無し	0	高	
	変換有り	確定	中	
超確率	変換無し	0	0	
	変換有り	確定	確定	

※ 確率分母256

【図 4 8】

ARTレベル決定テーブル(ART当籤時用)

ARTレベル	抽籤値
ARTレベル1	低
ARTレベル2	高
ARTレベル3	極低
ARTレベル4	極高

※ ART当籤時にフリーズ発生している場合は、ARTレベル2が決定
※ 確率分母256

ARTレベル決定テーブル(通常ART中用)

経過ゲーム数	移行先のARTレベル	現在のARTレベル		
		ARTレベル1	ARTレベル2	ARTレベル3
50ゲーム経過	ARTレベル1	確定	0	0
	ARTレベル2	0	確定	0
	ARTレベル3	0	0	確定
100ゲーム経過	ARTレベル1	確定	低	0
	ARTレベル2	0	高	0
	ARTレベル3	0	0	確定
200ゲーム経過	ARTレベル1	確定	極高	0
	ARTレベル2	0	低	0
	ARTレベル3	0	0	確定
300ゲーム経過	ARTレベル1	確定	種々高	中
	ARTレベル2	0	種々低	中
	ARTレベル3	0	0	中
500ゲーム経過	ARTレベル1	確定	種々高	0
	ARTレベル2	0	種々低	種々高
	ARTレベル3	0	0	低
1000ゲーム経過	ARTレベル1	確定	種々高	種々高
	ARTレベル2	0	種々低	種々高
	ARTレベル3	0	0	種々低
CT突入時 (0~100ゲーム)	ARTレベル1	中	低	0
	ARTレベル2	中	高	0
	ARTレベル3	0	0	確定
CT突入時 (101~500ゲーム)	ARTレベル1	種々高	中	0
	ARTレベル2	種々低	中	0
	ARTレベル3	0	0	確定
CT突入時 (501ゲーム以上)	ARTレベル1	種々高	種々高	0
	ARTレベル2	種々低	種々低	中
	ARTレベル3	0	0	中

※ 確率分母256

30

40

50

【図 4 9】

通常ART中高確率抽籤テーブル

現在の状態	移行後の状態	当籤役					
		押し順ベル共通ベル	サボテン	弱チェリー	強チェリー	3連チリリブ リーチ目リブ	それ以外
低確率 (ARTレベル1)	低確率	極高	中	中	極高	極高	確定
	通常	極低	中	中	0	0	0
	高確率	0	極低	極低	低	0	0
	超高確率	0	0	0	0	極低	0
低確率 (ARTレベル2~4)	低確率	極高	中	中	極高	極高	確定
	通常	極低	中	中	0	0	0
	高確率	0	極低	極低	低	0	0
	超高確率	0	0	0	0	極低	0
通常 (ARTレベル1)	低確率	0	0	0	0	極高	中
	通常	極高	中	中	高	0	中
	高確率	極低	中	中	中	0	0
	超高確率	0	0	0	極低	極低	0
通常 (ARTレベル2~4)	低確率	0	0	0	0	極高	低
	通常	極高	中	中	高	0	高
	高確率	低	中	中	中	0	0
	超高確率	0	0	0	極低	極低	0
高確率	低確率	0	0	0	0	極高	中
	通常	0	0	0	0	0	0
	高確率	確定	極高	極高	極高	0	中
	超高確率	0	極低	極低	極低	極低	0
超高確率	低確率	0	0	0	0	0	0
	通常	0	0	0	0	0	0
	高確率	0	0	0	0	低	0
	超高確率	確定	確定	確定	高	確定	確定

※ 確率分母256

【図 5 0】

ART中CT抽籤テーブル

現在の状態	内容	当籤役					
		サボテン	弱チェリー	強チェリー	3連チリリブ リーチ目リブ	BB	それ以外
低確率	非当籤	極高	極高	極高	0	確定	極々高
	通常CT	極低	極低	極低	確定	0	0
	高確率CT	0	0	0	0	0	極々低
通常	非当籤	極高	極高	極高	0	確定	極々高
	通常CT	極低	極低	低	確定	0	0
	高確率CT	0	0	0	0	0	極々低
高確率	非当籤	極高	極高	中	0	確定	極々高
	通常CT	低	極低	中	確定	0	極々低
	高確率CT	0	0	0	0	0	極々低
超高確率	非当籤	極高	極高	0	0	確定	極高
	通常CT	低	極低	確定	確定	0	極低
	高確率CT	0	0	0	0	0	極々低

※ サボテン、弱チェリー、強チェリー、3連チリリブ、リーチ目リブ、BBは、確率分母256
 それ以外は、確率分母65536
 ※ BBは内部当籤時にはCTの抽籤を行わず(必ず非当籤)、ボーナス状態終了時に抽籤する。
 ※ 確率分母256

10

20

【図 5 1】

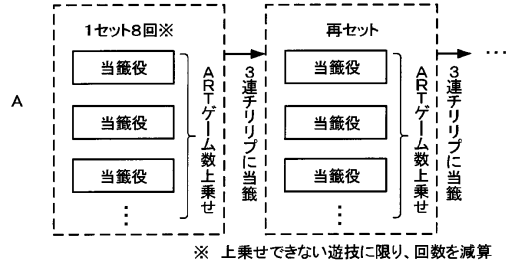
通常ART中上乗せ抽籤テーブル

内容	当籤役		
	サボテン	弱チェリー	強チェリー
非当籤	極高	極々高	0
上乗せ 10G	0	0	中
上乗せ 20G	0	0	中
上乗せ 30G	低	0	低
上乗せ 50G	極低	0	極低
上乗せ 100G	極低	極々低	極低
上乗せ 300G	極々低	極々低	極々低

※ 確率分母65536

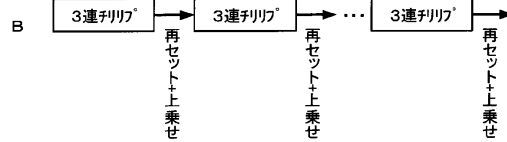
【図 5 2】

CT中の遊技の流れ1

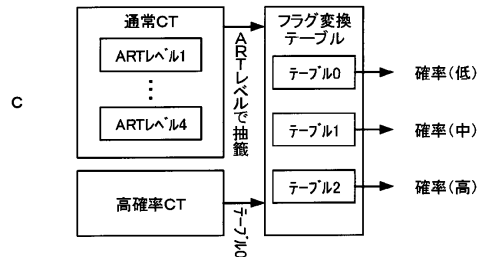


30

CT中の遊技の流れ2



CT中の変換



40

50

【 図 5 3 】

CT中テーブル抽籤テーブル

内容	状態				高確率CT
	ARTレベル1 (通常CT)	ARTレベル2 (通常CT)	ARTレベル3,4 (通常CT)	初回CT	
テーブル0	極高	高	高	高	確定
テーブル1	極低	低	低	低	0
テーブル2	極々低	極々低	極々低	極々低	0

※ 確率分母256

【 図 5 4 】

CT中フラグ変換抽籤テーブル

フラグ 変換テーブル	内容	当籤役	
		F. 確チリブ or F. 1確チリブ	F. リーチ目リブA~D
テーブル0	変換無し	0	高
	変換有り	確定	低
テーブル1	変換無し	0	低
	変換有り	確定	高
テーブル2	変換無し	0	0
	変換有り	確定	確定

※ 確率分母256

10

【 図 5 5 】

CT中上乗せ抽籤テーブル

状態	内容	当籤役			
		サボテン	弱チエリー	強チエリー	3連チリブ
通常CT	非当籤	0	0	0	0
	上乗せ 10G	極高	極高	0	極高 ※1
	上乗せ 20G	極低	極低	0	極低 ※2
	上乗せ 30G	0	0	極高	0
	上乗せ 50G	極々低	極々低	極々低	極々低
	上乗せ 100G	極々低	極々低	極々低	極々低
	上乗せ 300G	極々低	極々低	極々低	極々低
高確率CT	非当籤	0	0	0	0
	上乗せ 10G	0	0	0	0
	上乗せ 20G	0	0	0	0
	上乗せ 30G	0	0	0	0
	上乗せ 50G	極高	極高	高	極高
	上乗せ 100G	極低	極低	低	極低
	上乗せ 300G	極々低	極々低	極々低	極々低

※ 同一のCT中に3連チリブが当籤した回数に応じて、上乗せゲーム数が変わる。
 1~8回: ※1は10ゲーム、※2は20ゲーム
 9~16回: ※1, 2ともに20ゲーム
 17~24回: ※1, 2ともに30ゲーム
 25回以上: ※1, 2ともに50ゲーム
 ※ 確率分母65536
 ※ その他の役については非当籤

【 図 5 6 】

CT中セット数上乗せ抽籤テーブル

状態	内容	当籤役
		リーチ目リブ
通常CT	非当籤	0
	通常CT当籤	極々高
	高確率CT当籤	極々低
高確率CT	非当籤	0
	通常CT当籤	0
	高確率CT当籤	確定

※ 確率分母256

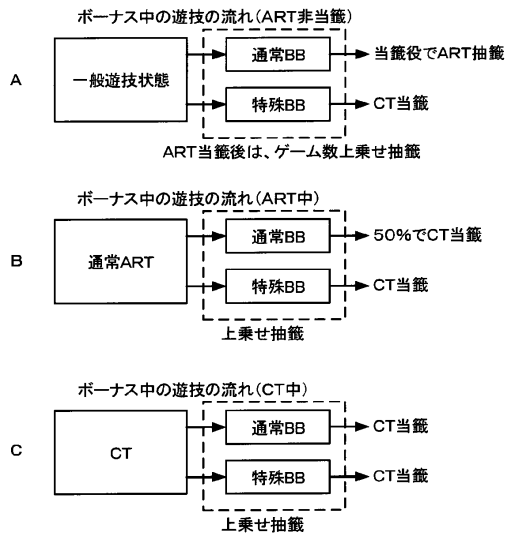
20

30

40

50

【 図 5 7 】



【 図 5 8 】

ボーナス種別抽籤テーブル

状態	内容	抽籤値
CT中	通常BB	極々高
	特殊BB	極々低
それ以外	通常BB	極々高
	特殊BB	極々低

※ 確率分母256

10

20

【 図 5 9 】

ボーナス中ARTゲーム数上乘せ抽籤テーブル

ボーナス種別	内容	当籤役		
		F.RB役1,2	F.RB役3	F.RB役4
通常BB	非当籤	0	確定	確定
	5G	0	0	0
	10G	0	0	0
	20G	0	0	0
	30G	0	0	0
	50G	極高	極高	0
	100G	極低	極低	0
特殊BB	非当籤	0	0	0
	5G	0	0	高
	10G	0	極高	低
	20G	0	極低	極低
	30G	0	極低	極低
	50G	0	0	極々低
	100G	極高	0	極々低
300G	低	0	極々低	

※ 確率分母256

【 図 6 0 】

ボーナス終了時CT抽籤テーブル

内容	ボーナス種別など			
	通常BB (一般遊技状態)	通常BB (通常ART)	特殊BB (一般・通常ART)	通常CT中 高確率CT中 (通常or特殊BB)
非当籤	確定	0	0	0
通常CT当籤	0	50%	確定	50%
高確率CT当籤	0	0	0	50%

※ 確率分母256

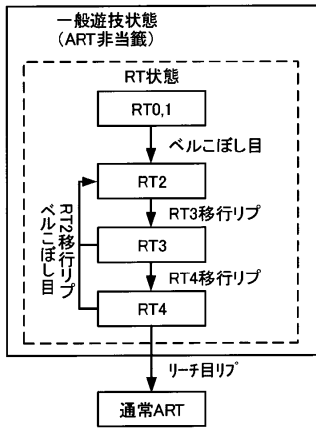
30

40

50

【 図 6 1 】

一般遊技状態中の遊技の流れ(その他)



【 図 6 2 】

非ART中フラグ変換抽籤テーブル

内容	当籤役
	F.リーチ目リブA~D
変換無し	極高
変換あり	極低

※ 確率分母256

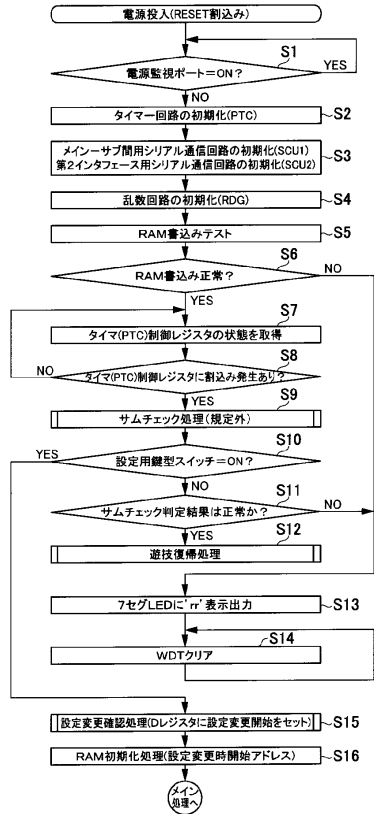
10

【 図 6 3 】

ナビ子-タの対応関係

内容	1	2	3	4	5	6	7	8	9	10	11
ART通常中	左	中	右	左-中-右	サブ脚ナビ	中-左-右	中-左-右	中-左-右	中-左-右	中-左-右	白7
ナビ子A	F.RT307.1sk	F.RT307.2sk	F.RT307.3sk	F.RT307.4sk	F.RT307.5sk	F.RT307.6sk	F.RT307.7sk	F.RT307.8sk	F.RT307.9sk	F.RT307.10sk	F.RT307.11sk
ナビ子B	F.RT307.1sk	F.RT307.2sk	F.RT307.3sk	F.RT307.4sk	F.RT307.5sk	F.RT307.6sk	F.RT307.7sk	F.RT307.8sk	F.RT307.9sk	F.RT307.10sk	F.RT307.11sk
ナビ子C	F.RT307.1sk	F.RT307.2sk	F.RT307.3sk	F.RT307.4sk	F.RT307.5sk	F.RT307.6sk	F.RT307.7sk	F.RT307.8sk	F.RT307.9sk	F.RT307.10sk	F.RT307.11sk
ナビ子D	F.RT307.1sk	F.RT307.2sk	F.RT307.3sk	F.RT307.4sk	F.RT307.5sk	F.RT307.6sk	F.RT307.7sk	F.RT307.8sk	F.RT307.9sk	F.RT307.10sk	F.RT307.11sk
ナビ子E	F.RT307.1sk	F.RT307.2sk	F.RT307.3sk	F.RT307.4sk	F.RT307.5sk	F.RT307.6sk	F.RT307.7sk	F.RT307.8sk	F.RT307.9sk	F.RT307.10sk	F.RT307.11sk
ナビ子F	F.RT307.1sk	F.RT307.2sk	F.RT307.3sk	F.RT307.4sk	F.RT307.5sk	F.RT307.6sk	F.RT307.7sk	F.RT307.8sk	F.RT307.9sk	F.RT307.10sk	F.RT307.11sk
ナビ子G	F.RT307.1sk	F.RT307.2sk	F.RT307.3sk	F.RT307.4sk	F.RT307.5sk	F.RT307.6sk	F.RT307.7sk	F.RT307.8sk	F.RT307.9sk	F.RT307.10sk	F.RT307.11sk
ナビ子H	F.RT307.1sk	F.RT307.2sk	F.RT307.3sk	F.RT307.4sk	F.RT307.5sk	F.RT307.6sk	F.RT307.7sk	F.RT307.8sk	F.RT307.9sk	F.RT307.10sk	F.RT307.11sk
ナビ子I	F.RT307.1sk	F.RT307.2sk	F.RT307.3sk	F.RT307.4sk	F.RT307.5sk	F.RT307.6sk	F.RT307.7sk	F.RT307.8sk	F.RT307.9sk	F.RT307.10sk	F.RT307.11sk
ナビ子J	F.RT307.1sk	F.RT307.2sk	F.RT307.3sk	F.RT307.4sk	F.RT307.5sk	F.RT307.6sk	F.RT307.7sk	F.RT307.8sk	F.RT307.9sk	F.RT307.10sk	F.RT307.11sk
ナビ子K	F.RT307.1sk	F.RT307.2sk	F.RT307.3sk	F.RT307.4sk	F.RT307.5sk	F.RT307.6sk	F.RT307.7sk	F.RT307.8sk	F.RT307.9sk	F.RT307.10sk	F.RT307.11sk
ナビ子L	F.RT307.1sk	F.RT307.2sk	F.RT307.3sk	F.RT307.4sk	F.RT307.5sk	F.RT307.6sk	F.RT307.7sk	F.RT307.8sk	F.RT307.9sk	F.RT307.10sk	F.RT307.11sk
ナビ子M	F.RT307.1sk	F.RT307.2sk	F.RT307.3sk	F.RT307.4sk	F.RT307.5sk	F.RT307.6sk	F.RT307.7sk	F.RT307.8sk	F.RT307.9sk	F.RT307.10sk	F.RT307.11sk
ナビ子N	F.RT307.1sk	F.RT307.2sk	F.RT307.3sk	F.RT307.4sk	F.RT307.5sk	F.RT307.6sk	F.RT307.7sk	F.RT307.8sk	F.RT307.9sk	F.RT307.10sk	F.RT307.11sk
ナビ子O	F.RT307.1sk	F.RT307.2sk	F.RT307.3sk	F.RT307.4sk	F.RT307.5sk	F.RT307.6sk	F.RT307.7sk	F.RT307.8sk	F.RT307.9sk	F.RT307.10sk	F.RT307.11sk
ナビ子P	F.RT307.1sk	F.RT307.2sk	F.RT307.3sk	F.RT307.4sk	F.RT307.5sk	F.RT307.6sk	F.RT307.7sk	F.RT307.8sk	F.RT307.9sk	F.RT307.10sk	F.RT307.11sk
ナビ子Q	F.RT307.1sk	F.RT307.2sk	F.RT307.3sk	F.RT307.4sk	F.RT307.5sk	F.RT307.6sk	F.RT307.7sk	F.RT307.8sk	F.RT307.9sk	F.RT307.10sk	F.RT307.11sk
ナビ子R	F.RT307.1sk	F.RT307.2sk	F.RT307.3sk	F.RT307.4sk	F.RT307.5sk	F.RT307.6sk	F.RT307.7sk	F.RT307.8sk	F.RT307.9sk	F.RT307.10sk	F.RT307.11sk
ナビ子S	F.RT307.1sk	F.RT307.2sk	F.RT307.3sk	F.RT307.4sk	F.RT307.5sk	F.RT307.6sk	F.RT307.7sk	F.RT307.8sk	F.RT307.9sk	F.RT307.10sk	F.RT307.11sk
ナビ子T	F.RT307.1sk	F.RT307.2sk	F.RT307.3sk	F.RT307.4sk	F.RT307.5sk	F.RT307.6sk	F.RT307.7sk	F.RT307.8sk	F.RT307.9sk	F.RT307.10sk	F.RT307.11sk
ナビ子U	F.RT307.1sk	F.RT307.2sk	F.RT307.3sk	F.RT307.4sk	F.RT307.5sk	F.RT307.6sk	F.RT307.7sk	F.RT307.8sk	F.RT307.9sk	F.RT307.10sk	F.RT307.11sk
ナビ子V	F.RT307.1sk	F.RT307.2sk	F.RT307.3sk	F.RT307.4sk	F.RT307.5sk	F.RT307.6sk	F.RT307.7sk	F.RT307.8sk	F.RT307.9sk	F.RT307.10sk	F.RT307.11sk
ナビ子W	F.RT307.1sk	F.RT307.2sk	F.RT307.3sk	F.RT307.4sk	F.RT307.5sk	F.RT307.6sk	F.RT307.7sk	F.RT307.8sk	F.RT307.9sk	F.RT307.10sk	F.RT307.11sk
ナビ子X	F.RT307.1sk	F.RT307.2sk	F.RT307.3sk	F.RT307.4sk	F.RT307.5sk	F.RT307.6sk	F.RT307.7sk	F.RT307.8sk	F.RT307.9sk	F.RT307.10sk	F.RT307.11sk
ナビ子Y	F.RT307.1sk	F.RT307.2sk	F.RT307.3sk	F.RT307.4sk	F.RT307.5sk	F.RT307.6sk	F.RT307.7sk	F.RT307.8sk	F.RT307.9sk	F.RT307.10sk	F.RT307.11sk
ナビ子Z	F.RT307.1sk	F.RT307.2sk	F.RT307.3sk	F.RT307.4sk	F.RT307.5sk	F.RT307.6sk	F.RT307.7sk	F.RT307.8sk	F.RT307.9sk	F.RT307.10sk	F.RT307.11sk

【 図 6 4 】



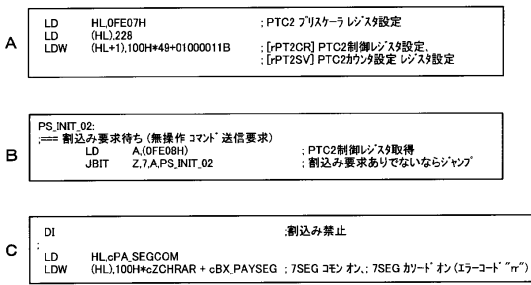
20

30

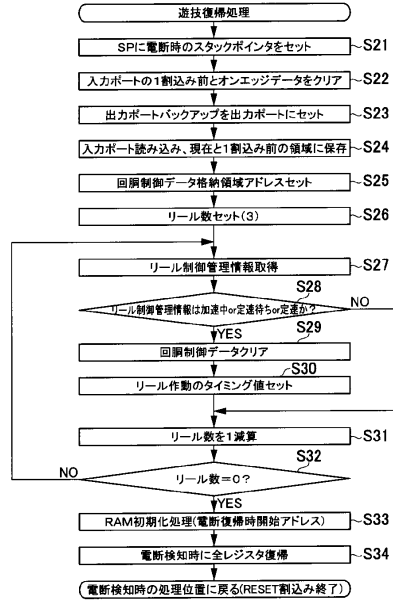
40

50

【図 6 5】



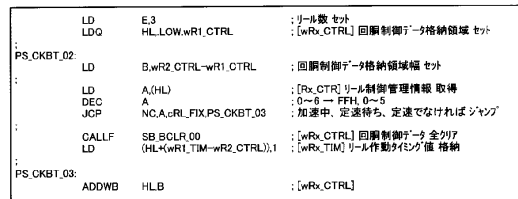
【図 6 6】



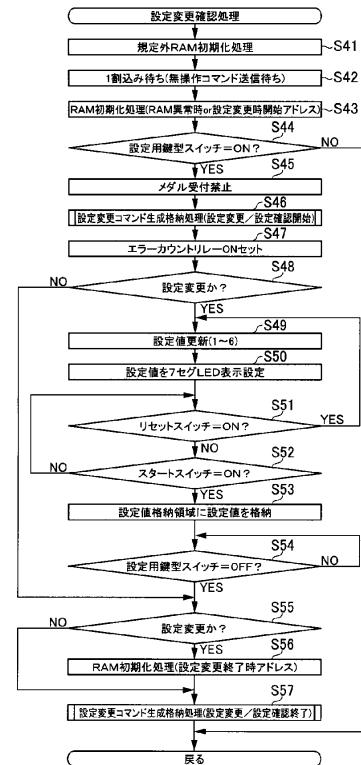
10

20

【図 6 7】



【図 6 8】



30

40

50

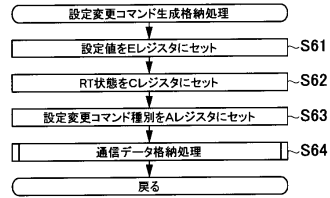
【 図 69 】

```

SB_WVSC_00:
  BITQ 7,(LOW.wBUF+4) ; キーオフ オンか?
  RET 2 ; 違うなら リターン
:
SB_WVSC_10:
  CALLF SB_MOFF_00 ; 9% 受付禁止処理
:
;== エラー カウントリミット 出力要求
  SETO 1,(LOW.wECRREQ) ; エラー カウントリミット 要求 セット
:
;== 初期化 コマンド 送信 (設定変更又は設定確認)
  LD L005H ; ハラマー01 セット (設定確認、キーオフ)
  CALLF SB_PCINIT_00 ; 初期化コマンド 送信処理
:
;== 設定確認終了 コマンド 送信
  LD L004H ; ハラマー01 セット (設定確認終了、キーオフ)
  CALLF SB_PCINIT_00 ; 初期化コマンド 送信処理
  RET

```

【 図 70 】



10

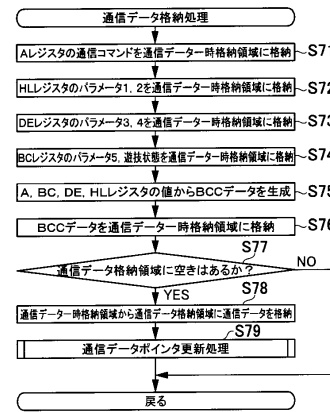
【 図 71 】

```

SB_PCINIT_00:
  CALLF SB_WVCK_00 ; 設定値 チェック処理
  LD E:A ; パラメータ3 セット (設定値)
  LDG C,(LOW.wRTGMSTS) ; ハラマー05 セット (RT遊技状態)
  LD A:02H ; 初期化コマンド セット
  RST SB_OUTP_00 ; 通信データ格納処理

```

【 図 72 】



20

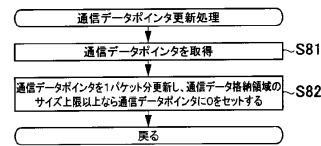
【 図 73 】

```

SB_OUTP_00:
  DI ; レジスタ 遊選
  PUSH GPR ; 遊技状態フラグ 取得
  LDQ B,(LOW.wGAMEST) ; 遊技状態フラグ 取得
  LDQ (LOW.wPDT_TMP+0):A ; 通信コマンド 格納
  LDQ (LOW.wPDT_TMP+1):HL ; ハラマー01, 2 格納
  LDQ (LOW.wPDT_TMP+3):DE ; ハラマー03, 4 格納
  LDQ (LOW.wPDT_TMP+5):BC ; ハラマー05, 遊技状態フラグ 格納
:
  ADD A:H ; BCCデータ 生成
  ADD A:L
  ADD A:D
  ADD A:E
  ADD A:C
  ADD A:B
  LDQ (LOW.wPDT_TMP+7):A ; BCCデータ 格納

```

【 図 74 】

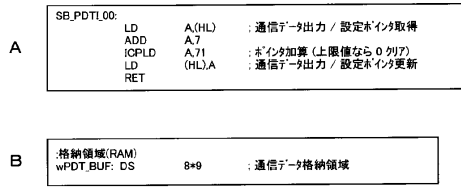


30

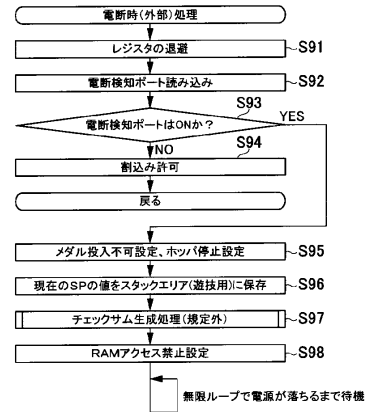
40

50

【 図 7 5 】



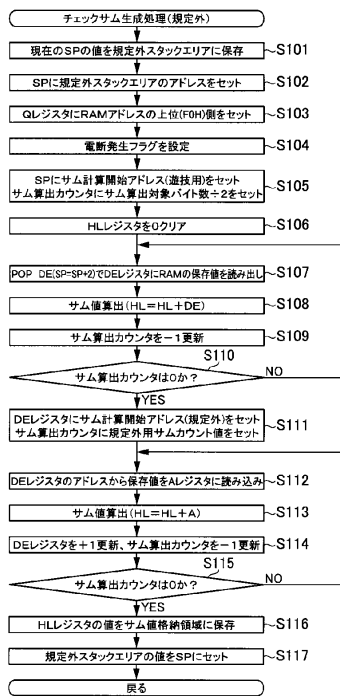
【 図 7 6 】



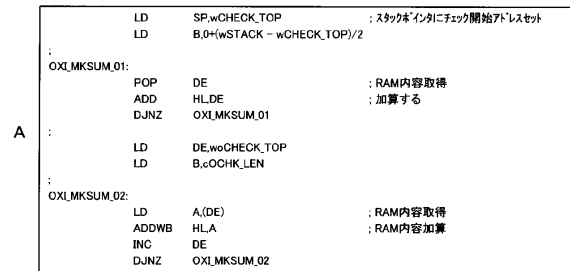
10

20

【 図 7 7 】

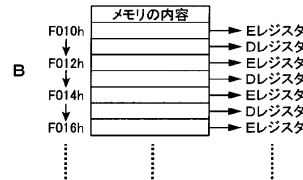


【 図 7 8 】



30

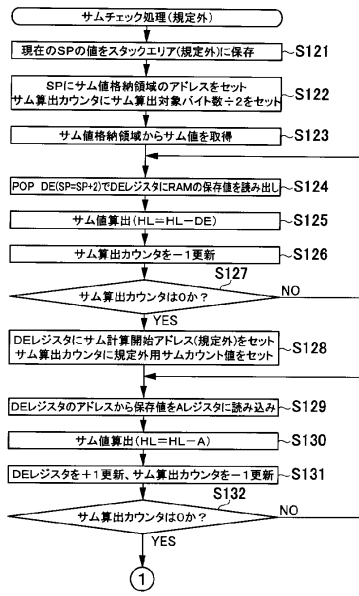
POP DE命令実行時のスタックポインタ及びRAM間のデータの流れ



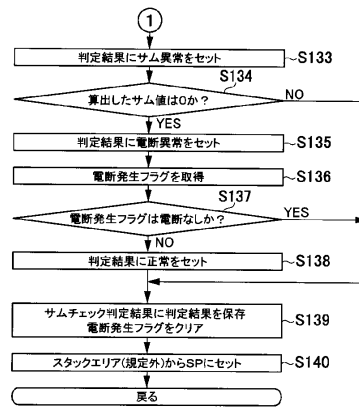
40

50

【 図 7 9 】



【 図 8 0 】



10

20

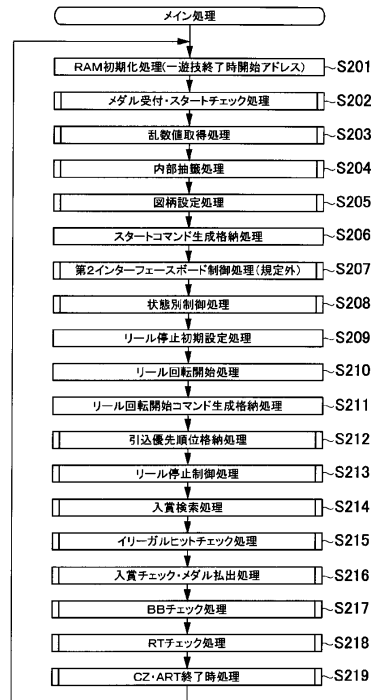
【 図 8 1 】

```

LD SP,wCHECK_TOP ; RAMフェックサム格納領域セット
LD B,0*(wSTACK - wCHECK_TOP)/2
LD HL,(wCHECKSUM) ; RAMフェックサム取得
:
OT_CKSUM_01:
POP DE ; RAM内容取得
SUB HL,DE ; RAMフェック (ZRFラグセット)
DJNZ OT_CKSUM_01 ; RAMフェック終了時 SP = F200H
:
LD DE,wCHECK_TOP ; 規定外RAMフェックサム格納領域セット
LD B,cOCHK_LEN
:
OT_CKSUM_02:
LD A,(DE) ; RAM内容取得
SUBWB HL,A ; RAMフェック (ZRFラグセット)
INC DE
DJNZ OT_CKSUM_02

```

【 図 8 2 】

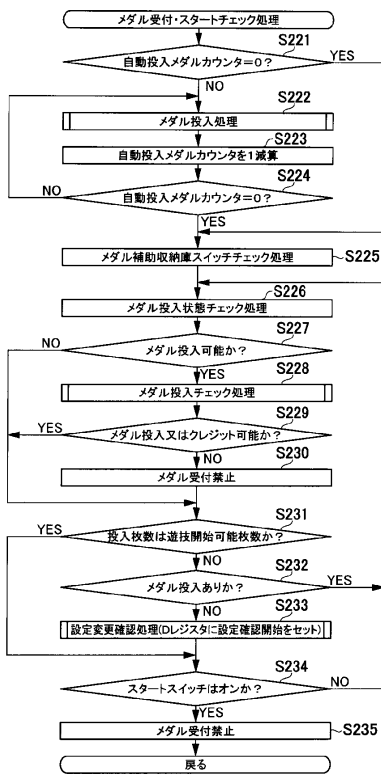


30

40

50

【 図 8 3 】



【 図 8 4 】

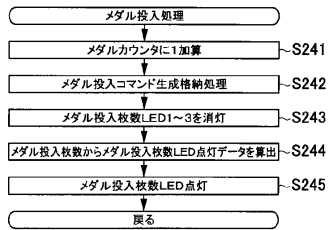
```

CALLF SB_MDCK_00 ;メダル投入枚数チェック処理
JR     NC,MN_GMST_08 ;遊技可能確定ならジャンプ
JR     NZ,MN_GMST_03 ;メダルが投入されているならジャンプ
LD     D,cBX_INT_WCK |cBX_INT_KEY ;設定確認入りフラグセット
CALLF SB_WVSC_00 ;設定変更確認処理
  
```

10

20

【 図 8 5 】



【 図 8 6 】

```

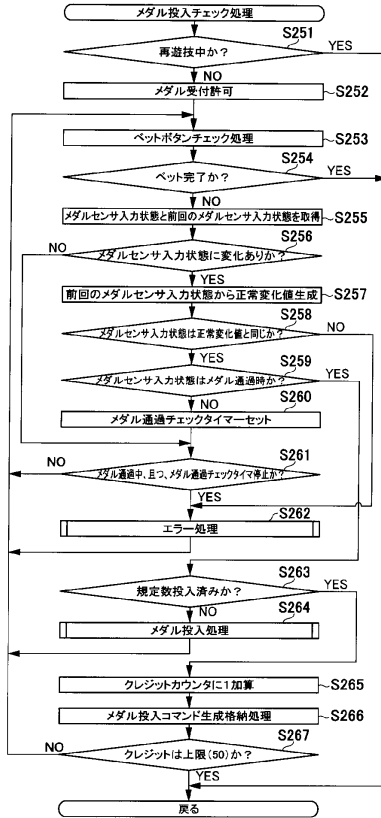
;メダル投入処理
_GMSB_MDIN_00 INQO (wMEDLCTR) ;メダルカウンタ加算更新
;
;== 投入枚数(1~3)コマンド送信
;
CALLF SB_PCMDIN_00 ;遊技メダル投入コマンド送信処理
;
;== メダル投入枚数LED点灯
;
CALLF SB_MDLLDF_00 ;メダル投入枚数LED消灯処理
;
LD     AL ;メダル投入枚数セット
ADD   AA @0 1:010, 2:100, 3:110
DEC   A @0 1:001, 2:011, 3:101
OR    L @0 1:001, 2:011, 3:111
LD     BA ;メダル投入枚数LED点灯データセット
LD     C,cI0SET+ePR_MDLLD*2
RST   SB_OUTX_00 ;メダル投入枚数LED点灯
RET
  
```

30

40

50

【 図 8 7 】



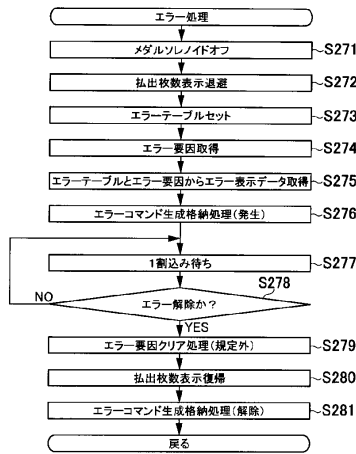
【 図 8 8 】

_GMST_MDIC_01:			
LD	E_cERR_CR	: 投入メダル逆行エラーコードセット (cERR_CR)	
LDQ	HL_wMINLVST	: メダル受付状態ステータス格納領域セット	
LDQ	A_wIBUF+cI_NOW+cPL_MDINSW	: 投入メダル入力情報取得	
AND	cBX_MDINSW	: 投入メダル状態抽出	
CP	(HL)	: 前回と同じ状態か?	
JR	Z_GMST_MDIC_02	: そうならシフト	
;== 状態変化時のチェック			
LD	B,(HL)	: メダル受付状態ステータス取得	
LD	(HL),A	: 新しい状態格納	
:			
LD	A,B	: 古い状態セーブ	
CP	cBX_MDINSW2	: 正常変化値生成	
HLA	AND	cBX_MDINSW	: @@ 1: 00B → 01B
			: @@ 2: 01B → 11B
			: @@ 3: 11B → 10B
			: @@ 4: 10B → 00B
CP	(HL)	: 正常な変化の仕方か?	
JR	NZ_GMST_MDIC_03	: 違うならシフト	

10

20

【 図 8 9 】



【 図 9 0 】

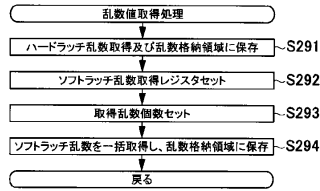
dERRTB	EQU	\$: 本ソフトエラーなし
dERR.HE	DB	00000000B	: 本ソフトエラーなし
DB	00110111B,01001111B	: ホール-エンブレムエラー "HE"	
dERR.HJ	DB	01000000B	: ホール-カウントスイッチ
DB	00110111B,00111000B	: ホール-ジムエラー "HJ"	
dERR.CC	DB	00100000B	: メダル通過チェックスイッチ2
DB	01001110B,01001110B	: 投入メダル通過カウントエラー "CC"	
dERR.CJ	DB	00000100B	: メダル通過チェックスイッチ1
DB	01001110B,00111000B	: 投入メダル通過チェックエラー "CJ"	
dERR.CE	DB	00000111B	: 投入メダル1, 2
DB	01001110B,01001111B	: 投入メダル通過時間エラー "CE"	
dERR.CR	DB	00000011B	: 投入メダル逆行エラー "Cr"
DB	01001110B,00001001B	: 投入メダル逆行エラー "Cr"	
dERR.CO	DB	10000000B	: 補助収納庫スイッチ
DB	01001110B,00011101B	: 遊技メダル補助収納庫満杯エラー "Co"	
dERR.EE	DB	00000000B	: 本ソフトエラーなし
DB	01001111B,01001111B	: イレガリティエラー "EE"	

30

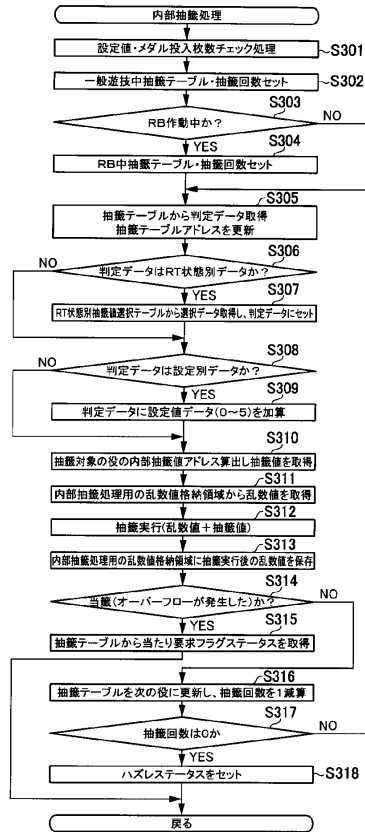
40

50

【図91】



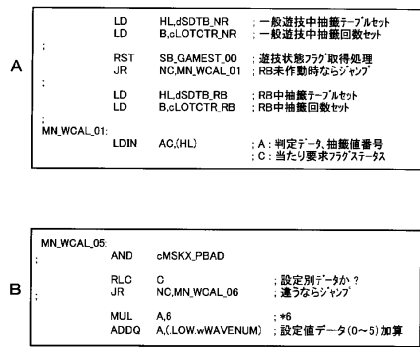
【図92】



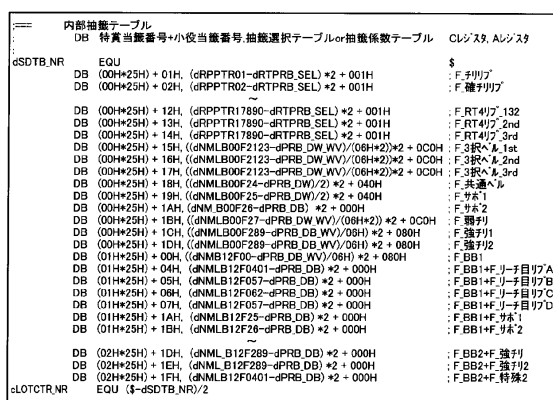
10

20

【図93】



【図94】



30

40

50

【図 9 5】

```

==== RT状態別抽籤値選択テーブル
dRTPRE_SEL EQU $
;
dRPPTR01: ; F.リリブ
DB 00010000B
DB ((dRT4B00F01-dPRB_DW)/2)*2 +040H ; RT4 作動中
;
dRPPTR02: ; F.維持リブ
DB 00010000B
DB ((dRT4B00F02-dPRB_DW_WV)/(06H*2))*2 +0C0H ; RT4 作動中
;
dRPPTR03: ; F.1確リリブ
DB 00010000B
DB ((dRT4B00F03-dPRB_DW)/2)*2 +040H ; RT4 作動中
;
;
dRPPTR08: ; F.維持リブA
DB 00110111B
DB ((dRT0B00F06-dPRB_DW)/2)*2 +040H ; RT未作動時
DB ((dRT1B00F06-dPRB_DW)/2)*2 +040H ; RT1 作動中
DB ((dRT2B00F06-dPRB_DW)/2)*2 +040H ; RT2 作動中
DB ((dRT4B00F08-dPRB_DW_WV)/(06H*2))*2 +0C0H ; RT4 作動中
DB ((dRT5B00F08-dPRB_DW)/2)*2 +040H ; RT5 作動中
;
;
dRPPTR09: ; F.維持リブB
DB 00010110B
DB ((dRT124B00F09-dPRB_DW)/2)*2 +040H ; RT1 作動中
DB ((dRT124B00F08-dPRB_DW)/2)*2 +040H ; RT2 作動中
DB ((dRT124B00F09-dPRB_DW)/2)*2 +040H ; RT4 作動中
;
;
dRPPTR1012: ; F.維持リブ1st.F.維持リブ2nd.F.維持リブ3rd
DB 00111000B
DB ((dRT3B00F1012-dPRB_DW)/2)*2 +040H ; RT3 作動中
DB ((dRT4B00F1012-dPRB_DW)/2)*2 +040H ; RT4 作動中
DB ((dRT5B00F1012-dPRB_DW)/2)*2 +040H ; RT5 作動中
;
;
dRPPTR17890: ; F.RT4リブ123.F.RT4リブ132.F.RT4リブ2nd.F.RT4リブ3rd
DB 00001000B
DB ((dRT3B00F17890-dPRB_DW)/2)*2 +040H ; RT3 作動中

```

【図 9 6】

```

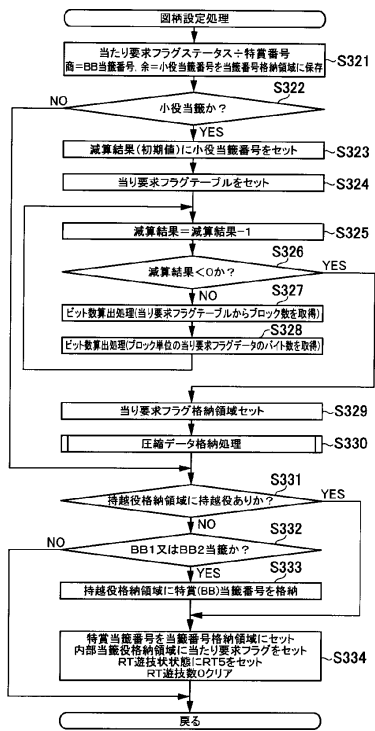
==== 内部抽籤値テーブル 選択テーブル
dPRB_SEL DB dPRB_DB - dPRB_SEL ; [0] 1バイト内部抽籤値テーブル
DB dPRB_DW - dPRB_SEL ; [1] 2バイト内部抽籤値テーブル
DB dPRB_DB_WV - dPRB_SEL ; [2] 1バイト設定別内部抽籤値テーブル
DB dPRB_DW_WV - dPRB_SEL ; [3] 2バイト設定別内部抽籤値テーブル
;
==== 内部抽籤値テーブル
;==== 1バイト内部抽籤値テーブル
dPRB_DB:
dNMLB12F0401: DB 5 ; F.BB1+F.リブ目リブA.F.BB2+F.リブ目リブA.F.BB1+F.特殊1.F.BB2+F.特殊2
;
dNMLB00F28: DB 128 ; F.リブ2
dNMLB00F34: DB 100 ; F.リブ役2
;
;==== 2バイト内部抽籤値テーブル
dPRB_DW:
dRT0B00F08: DW 8900 ; [RT0] F.維持リブA
dRT1B00F08: DW 37800 ; [RT1] F.維持リブA
dRT124B00F08: DW 330 ; [RT1,RT2,RT4] F.維持リブB
;
;==== [RT2]
dRT2B00F08: DW 1300 ; F.維持リブA
dRT2B00F13456: DW 1800 ; F.RT3リブ1st.F.RT3リブ213.F.RT3リブ231.F.RT3リブ3rd
;
;==== [RT5]
dRT5B00F1012: DW 1640 ; F.維持リブ1st.F.維持リブ2nd.F.維持リブ3rd
dRT5B00F13456: DW 410 ; F.RT3リブ1st.F.RT3リブ213.F.RT3リブ231.F.RT3リブ3rd
dRT5B00F04: DW 4100 ; F.リブ目リブA
dRT5B00F05: DW 1650 ; F.リブ目リブB
;
;==== [RT3]
dNMLB00F33: DW 300 ; F.RB役1
dNMLB00F35: DW 7300 ; F.RB役3
dNMLB00F36: DW 57836 ; F.RB役4
;
;==== 1バイト設定別内部抽籤値テーブル
dPRB_DB_WV:
dNMLB00F289: DB 150,150,150,150,160,170 ; F.強リ1.F.強リ2
dNMLB1F00: DB 20,21,22,23,24,26 ; F.BB1.F.BB2
dNMLB12F27: DB 5,7,9,11,13,15 ; F.BB1+F.弱リ.F.BB2+F.弱リ
;
;==== 2バイト設定別内部抽籤値テーブル
dPRB_DW_WV:
dRT4B00F02: DW 13400,11000,13400,11000,13400,11000 ; [RT4] F.維持リブ
dRT4B00F08: DW 11800,14100,11680,13870,11530,13720 ; [RT4] F.維持リブA
dNMLB00F2123: DW 5250,5260,5230,5260,5190,5210 ; F.3択へリ1st.F.3択へリ2nd.F.3択へリ3rd
dNMLB00F27: DW 970,900,930,960,990,1020 ; F.弱リ

```

10

20

【図 9 7】



【図 9 8】

当籤番号一覧(定義)			
特賞番号(ボーナス番号)			
cNHT_BB01	EQU	01	@@ 01_F_BB1
cNHT_BB02	EQU	02	@@ 02_F_BB2
小役当籤番号			
cNHT_FC01	EQU	01	@@ 01_F_リリブ
cNHT_FC02	EQU	02	@@ 02_F_維持リブ
cNHT_FC03	EQU	03	@@ 03_F_1確リリブ
cNHT_FC04	EQU	04	@@ 04_F_リブ目リブ A
cNHT_FC05	EQU	05	@@ 05_F_リブ目リブ B
cNHT_FC06	EQU	06	@@ 06_F_リブ目リブ C
cNHT_FC07	EQU	07	@@ 07_F_リブ目リブ D
cNHT_FC08	EQU	08	@@ 08_F_維持リブ A
cNHT_FC09	EQU	09	@@ 09_F_維持リブ B
cNHT_FC10	EQU	10	@@ 10_F_維持リブ_1st
cNHT_FC11	EQU	11	@@ 11_F_維持リブ_2nd
cNHT_FC12	EQU	12	@@ 12_F_維持リブ_3rd
cNHT_FC13	EQU	13	@@ 13_F_RT3リブ_1st
cNHT_FC14	EQU	14	@@ 14_F_RT3リブ_213
cNHT_FC15	EQU	15	@@ 15_F_RT3リブ_231
cNHT_FC16	EQU	16	@@ 16_F_RT3リブ_3rd
cNHT_FC17	EQU	17	@@ 17_F_RT4リブ_123
cNHT_FC18	EQU	18	@@ 18_F_RT4リブ_132
cNHT_FC19	EQU	19	@@ 19_F_RT4リブ_2nd
cNHT_FC20	EQU	20	@@ 20_F_RT4リブ_3rd
cNHT_FC21	EQU	21	@@ 21_F_3択へリ_1st
cNHT_FC22	EQU	22	@@ 22_F_3択へリ_2nd
cNHT_FC23	EQU	23	@@ 23_F_3択へリ_3rd
cNHT_FC24	EQU	24	@@ 24_F_共通へリ
cNHT_FC25	EQU	25	@@ 25_F_リブA
cNHT_FC26	EQU	26	@@ 26_F_リブB
cNHT_FC27	EQU	27	@@ 27_F_弱リ
cNHT_FC28	EQU	28	@@ 28_F_強リ1
cNHT_FC29	EQU	29	@@ 29_F_強リ2
cNHT_FC30	EQU	30	@@ 30_F_特殊1
cNHT_FC31	EQU	31	@@ 31_F_特殊2
cNHT_FC32	EQU	32	@@ 32_F_特殊3
cNHT_FC33	EQU	33	@@ 33_F_RB役1
cNHT_FC34	EQU	34	@@ 34_F_RB役2
cNHT_FC35	EQU	35	@@ 35_F_RB役3
cNHT_FC36	EQU	36	@@ 36_F_RB役4

30

40

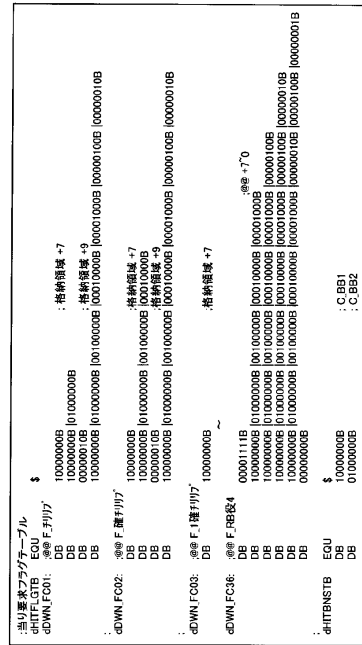
50

【 9 9 】

```

==小役当り要求 フラグテーブル 検索処理
LD HL,dHTFLGTB ;小役当り要求フラグテーブル セット
MN_FGST_03:
DEC D
JR Z,MN_FGST_05
;
CALLF SB,BTNM_00 ;ビット数算出処理
CALLF SB,BTNM_00 ;ビット数算出処理
JR MN_FGST_03
==小役当り要求 フラグテーブル展開処理
MN_FGST_05:
LDG DE,LOWwWAVEBIT ;当り要求フラグ格納領域 セット
CALLF SB,BTEP_00 ;圧縮テーブル格納処理
    
```

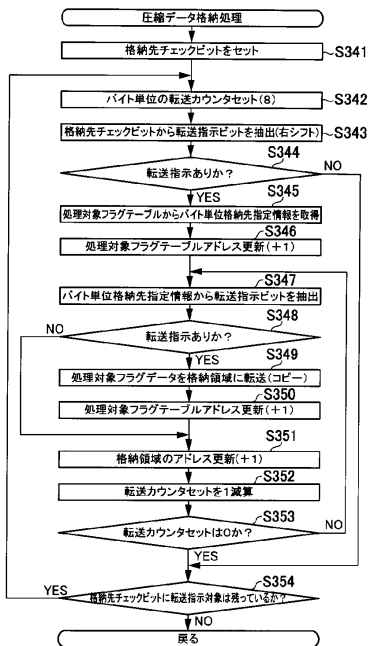
【 1 0 0 】



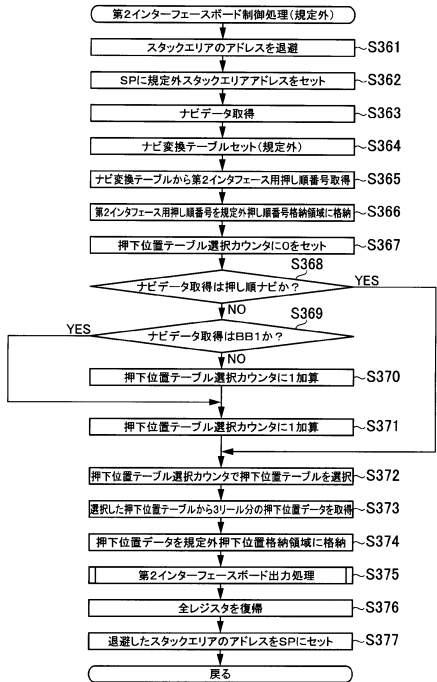
10

20

【 1 0 1 】



【 1 0 2 】

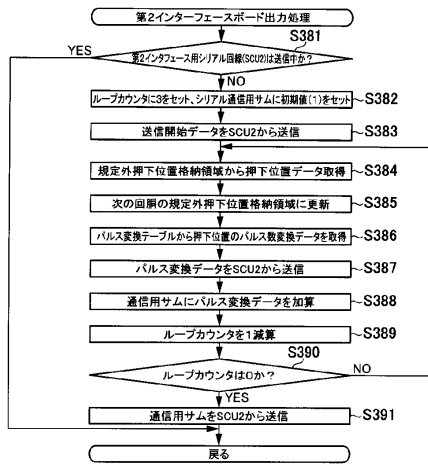


30

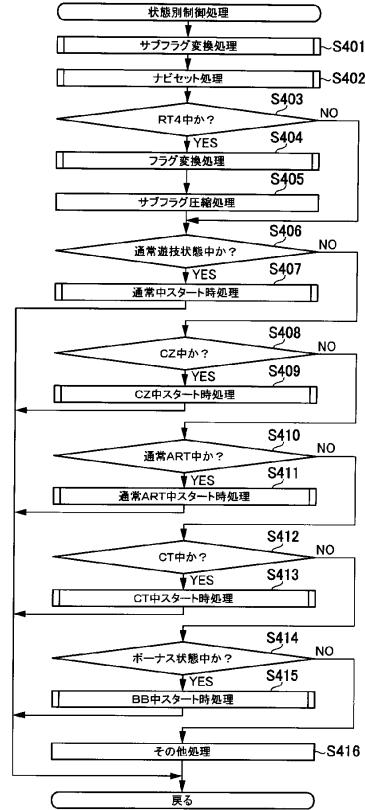
40

50

【 図 1 0 3 】



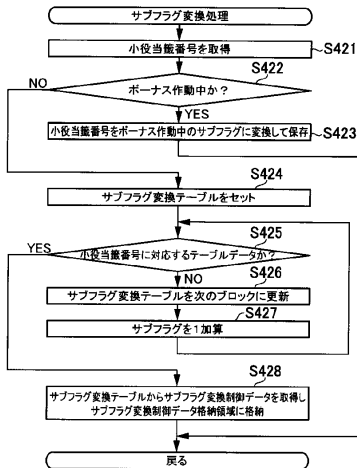
【 図 1 0 4 】



10

20

【 図 1 0 5 】



【 図 1 0 6 】

```

EF.SFCV.00 サブフラグ変換処理
EF.SFCV.00:
RST SB_GAMEST_00 : 遊技状態フラグ取得処理
LDQ A(wHITFRIT) : 小役番号取得
LDQ DE.wSUBFLG : サブフラグ格納領域
JR NC.EF.SFCV_01 : 特賞作動中でないならジャンプ
SUB @NHT_RBST-c7HT1_FLG : (RB中フラグ判定番号(33) - 7番い(19))=サボラン(14)
LD (DE)A : サブフラグ格納
JR EF.SFCV_04

EF.SFCV_01:
LD HL.dSBCVTB : サブフラグ変換テーブルセット
EF.SFCV_02:
CP (HL) : 対応フラグか?
INC HL : 
JR C.EF.SFCV_03 : そうならジャンプ
INC HL : 
INC (DE) : [wSUBFLG] サブフラグ+1
JR EF.SFCV_02

EF.SFCV_03:
LDQ (wSFCONDA).(HL) : サブフラグ制御データ格納
EF.SFCV_04:
RET
  
```

30

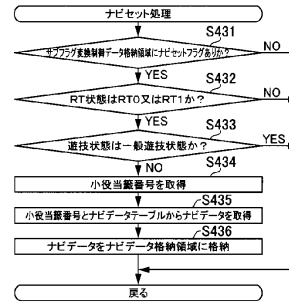
40

50

【 107 】

サブフラグ変換テーブル			
dSBCVTB	EQU	\$	
DB	00+1	00010000B	:00:ハズレ
DB	01+1	00000000B	:01:2選リリブ
DB	02+1	00000011B	:02:3選リリブA
DB	03+1	00000010B	:03:3選リリブB
DB	04+1	00000001B	:04:リ子目リブ1
DB	05+1	00000000B	:05:リ子目リブ2
DB	06+1	00000001B	:06:リ子目リブ3
DB	07+1	00000000B	:07:リ子目リブ4
DB	08+1	00000000B	:08:グレイ
DB	12+1	01000000B	:09:押し順リブ1
DB	16+1	01000000B	:10:押し順リブ2
DB	20+1	01000000B	:11:押し順リブ3
DB	23+1	01000100B	:12:押し順ヘル
DB	24+1	00000000B	:13:共通ヘル
DB	26+1	00000000B	:14:サボテン
DB	27+1	00000000B	:15:弱チリ
DB	29+1	00000000B	:16:強チリ
DB	31+1	00011000B	:17:リ子目1
DB	32+1	00011000B	:18:リ子目2

【 108 】



10

【 109 】

LDO	A,(WHITRT)	:小役番号格納領域(0 ~ 36)
LD	HL,dNAVIPTNB-10	:ナビデータテーブル(小役番号10 ~ 23)
RST	SB,ADRA_00	
LDO	(wNAVIPTNB),A	:ナビセット

【 110 】

dNAVIPTNB	EQU	\$	
DB	7		:10:F 維持リブ_1st
DB	8		:11:F 維持リブ_2nd
DB	9		:12:F 維持リブ_3rd
DB	1		:13:F RT3リブ_1st
DB	3		:14:F RT3リブ_213
DB	4		:15:F RT3リブ_231
DB	6		:16:F RT3リブ_3rd
DB	1		:17:F RT4リブ_123
DB	2		:18:F RT4リブ_132
DB	3		:19:F RT4リブ_2nd
DB	6		:20:F RT4リブ_3rd
DB	7		:21:F 3択ヘル_1st
DB	8		:22:F 3択ヘル_2nd
DB	9		:23:F 3択ヘル_3rd

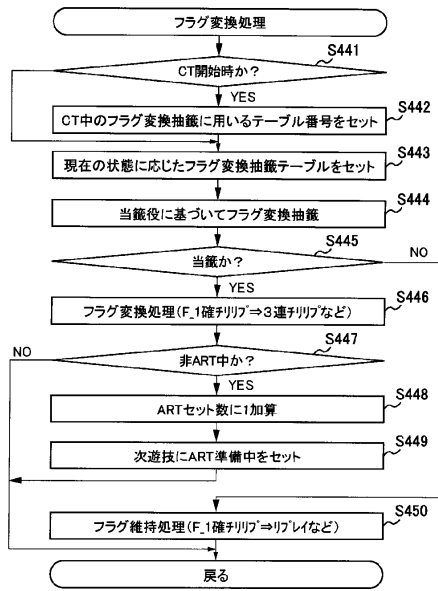
20

30

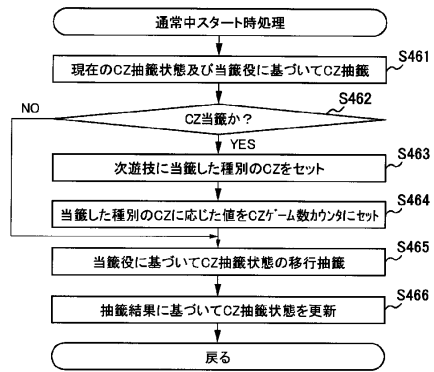
40

50

【図 1 1 1】



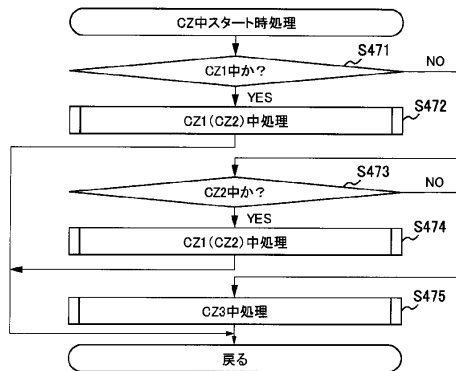
【図 1 1 2】



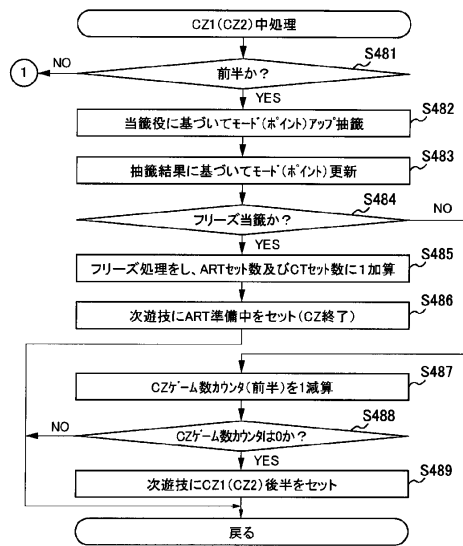
10

20

【図 1 1 3】



【図 1 1 4】

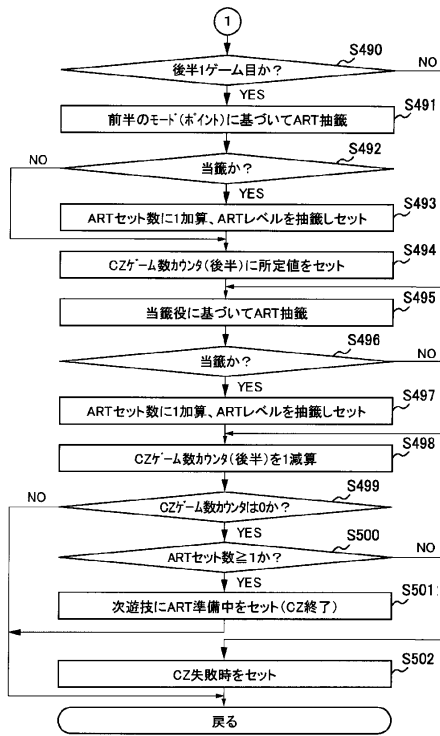


30

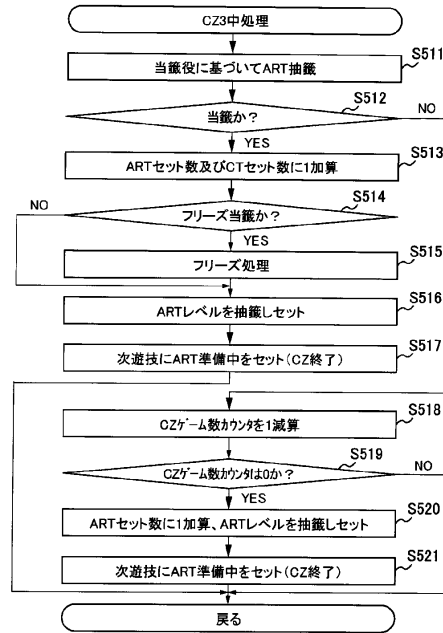
40

50

【図 1 1 5】



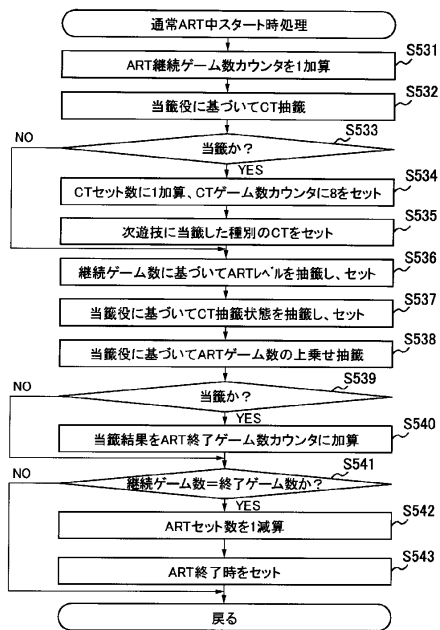
【図 1 1 6】



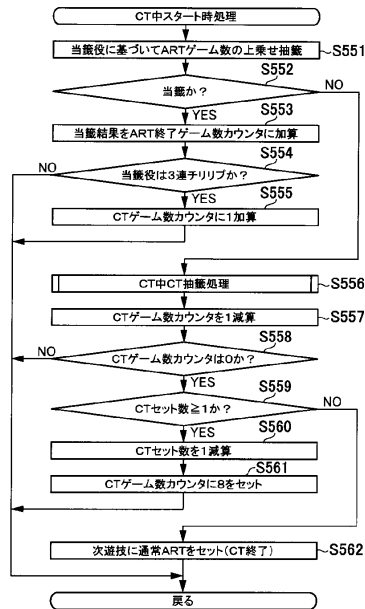
10

20

【図 1 1 7】



【図 1 1 8】

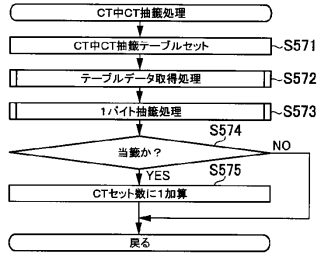


30

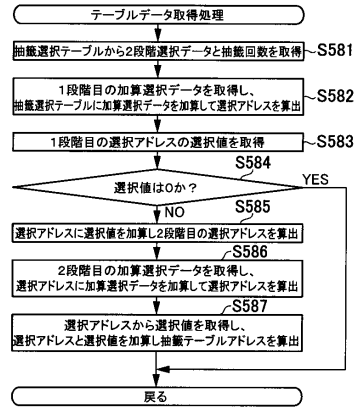
40

50

【 図 1 1 9 】



【 図 1 2 0 】



10

【 図 1 2 1 】

```

;== テーブルデータ取得処理
ESB_GTD2_00:
    LDIN DE,(HL) ;加算RAMアドレス取得 E:1回目 D:2回目
    LD B,(HL) ;抽籤回数取得
    INC HL
    LDQ A,(E)
    RST SB_ADRA_00
    RT ZA
    ADDWB HLA
ESB_GTD2_20:
    LDQ A,(D)
    RST SB_ADRA_00
    RST SB_ADRA_00
    RET
  
```

【 図 1 2 2 】

```

;== CT中CT当籤抽籤テーブル
cABS_HIT EQU 0
dCTCTSTTB EQU $ ;当籤役別テーブル選択相対テーブル
DB LOW.wSUBFLGD ;サブフラグD
DB LOW.wCT_KIND ;CT種別
DB ? ;抽籤回数
DB 0,0,0,0,0 ;ハシレ,弱チリ,弱チリ,強チリ,確定役,3選チリリブ
DB dCTCTS_RER-$ ;リブ目リブ
dCTCTS_RER: DB dNMCTCTS_RER-$ ;選券リブ目リブ
DB dSPCTCTS_RER-$ ;高確リブ目リブ
dNMCTCTS_RER: DB 00000011B ;判定ピット
DB 1 ;高確CT
DB 255 ;選券CT
DB ;ハシレ
dSPCTCTS_RER: DB ;高確リブ目リブ
DB 00000001B ;判定ピット
DB cABS_HIT ;高確CT
DB ;選券CT
DB ;ハシレ
  
```

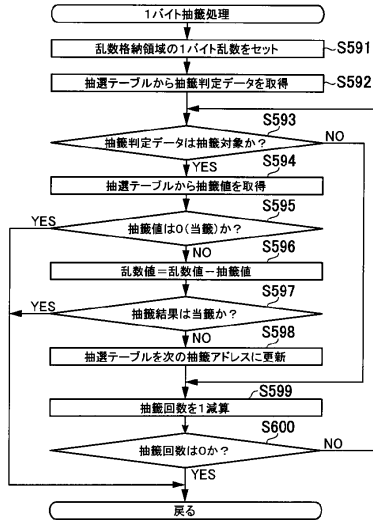
20

30

40

50

【図 1 2 3】



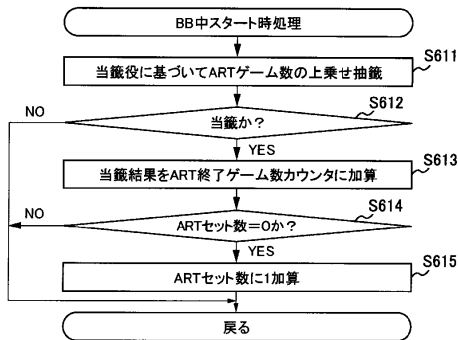
【図 1 2 4】

```

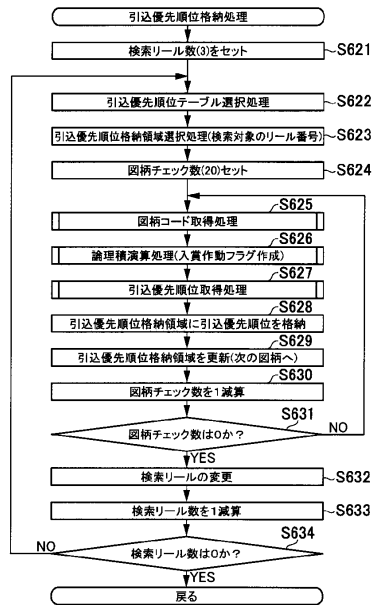
;ESB_LOTP_00 1バイト抽籤処理
ESB_LOTP_00:
LDQ DF,wRANDOMP ;乱数値P
ESB_LOT1_00:
LD C(HL) ;抽籤判定データ取得
INC HL
ESB_LOT1_01:
SRL C
JR NO,ESB_LOT1_02 ;抽籤対象でなければジャンプ
JT Z,(HL),ESB_LOT1_03 ;0なら当籤
;
LD A,(DE) ;抽籤実行
SUB (HL)
LD (DE),A
JR C,ESB_LOT1_03 ;当籤ならジャンプ
;
INC HL ;次のデータへ
ESB_LOT1_02:
DJNZ ESB_LOT1_01
ESB_LOT1_03:
LD A,B
RET
  
```

10

【図 1 2 5】



【図 1 2 6】



20

30

40

50

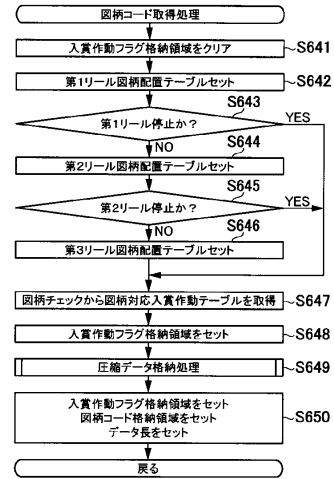
【図 1 2 7】

```

LD   A,cREEL_ZONT
SUB  B           ; 図柄コード取得処理
CALLF ,STSB_RLTB_00 ; @@ DE = [wHITFLAG] 入賞作動フラグ格納領域
                           ; @@ HL = [wXARAY] 図柄コード格納領域
                           ; @@ B = cARYSIZE
:
;== 1 ラインに対する入賞作動フラグ作成
:
EX   DF,HL
CALLF SB,DAND_00 ; 論理積演算処理

```

【図 1 2 8】



10

【図 1 2 9】

```

STSB_RLTB_00:
CALLF SB,HTFLGOR_00 ; 入賞作動フラグクリア
:
LD   HL,dREELTB1 ; 第一回胴図柄配置テーブル
JBIT NZ,0,C,STSB_RLTB_01 ; 第一回胴ならジャンプ
LD   HL,dREELTB2 ; 第二回胴図柄配置テーブル
JBIT NZ,1,C,STSB_RLTB_01 ; 第二回胴ならジャンプ
LD   HL,dREELTB3 ; 第三回胴図柄配置テーブル
:
STSB_RLTB_01:
RST  SB,ADRA_00 ; 図柄コード相対アドレス取得
ADDWB H,LA ; 図柄コードアドレス取得
:
LDQ  DE,wHITFLAG ; 入賞作動フラグデータ格納領域処理
CALLF SB,BTEP_00
:
LDQ  DE,wHITFLAG ; 入賞作動フラグ格納領域セット
LDQ  HL,wXARAY ; 図柄コード格納領域セット
LD   B,cARYSIZE ; データ長セット
RET

```

【図 1 3 0】

```

;== 第一回胴 図柄配置テーブル
dREELTB1 EQU $
DB dR1_REP - $ ; 図柄位置 0
DB dR1_BEL - $ ; 1
DB dR1_SVN1 - $ ; 2
~
DB dR1_REP - $ ; 18
DB dR1_CCT2 - $ ; 19

```

```

;== 図柄対応入賞作動テーブル(第一回胴)
dR1_SVN1:
DB 0101100B ; 白7 (第一回胴)
DB 00010000B | 00000100B ; 格納領域 +2, +3, +4, +6 ; フラグ
DB 00100000B | 00001000B
DB 10000000B | 01000000B | 00100000B
DB 00100000B | 00010000B | 00001000B
DB 00000100B ; 格納領域 +10
:C.BB1
DB 10000000B
dR1_SVN2:
DB 00100101B ; 青7 (第一回胴)
;以下省略

```

20

30

40

50

【 図 1 3 1 】

A

```

:= 第二回胴 図柄配置テーブル
dREELTB2 EQU $
DB dR2_BEL_-$ : 図柄位置 0
DB dR2_REP_-$ : 1
DB dR2_SVN1_-$ : 2
~
DB dR2_REP_-$ : 18
DB dR2_CCT2_-$ : 19

```

B

```

:= 図柄対応入賞作動テーブル(第二回胴)
dR2_SVN1: ; 白7 (第二回胴)
DB 01111001B : 格納領域 +0, +3, +4, +5, +6
DB 00100000B
DB 01000000B [00000001B
DB 10000000B [00100000B [00001000B
DB 10000000B
DB 00010000B [00001000B
DB 00000110B : 格納領域 +9, +10
DB 10000000B [00100000B
DB 10000000B
dR2_SVN2: ; 青7 (第二回胴)
DB 00011101B : 格納領域 +0, +2, +3, +4
-以下省略-

```

【 図 1 3 2 】

A

```

:= 第三回胴 図柄配置テーブル
dREELTB3 EQU $
DB dR3_REP_-$ : 図柄位置 0
DB dR3_BEL_-$ : 1
DB dR3_SVN1_-$ : 2
~
DB dR3_CH2_-$ : 18
DB dR3_CCT1_-$ : 19

```

B

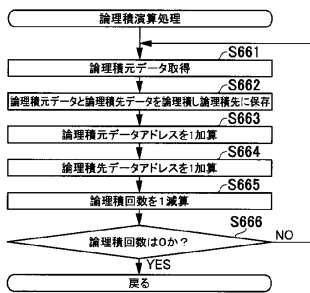
```

:= 図柄対応入賞作動テーブル(第三回胴)
dR3_SVN1: ; 白7 (第三回胴)
DB 01111001B : 格納領域 +0, +3, +4, +5, +6
DB 00001000B
DB 00100000B [00010000B [00000100B
DB 00010000B
DB 10000000B [00100000B [00010000B
DB 00100000B [00000001B
DB 00000101B : 格納領域 +8, +10
DB 00001000B [00000100B
DB 10000000B
dR3_SVN2: ; 青7 (第三回胴)
DB 01111001B : 格納領域 +0, +3, +4, +5, +6
-以下省略-

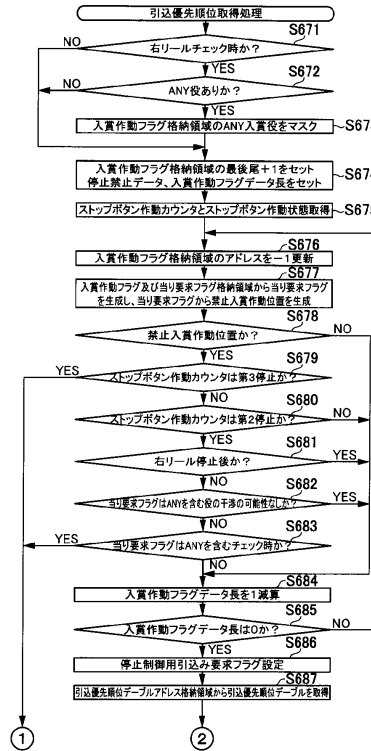
```

10

【 図 1 3 3 】



【 図 1 3 4 】



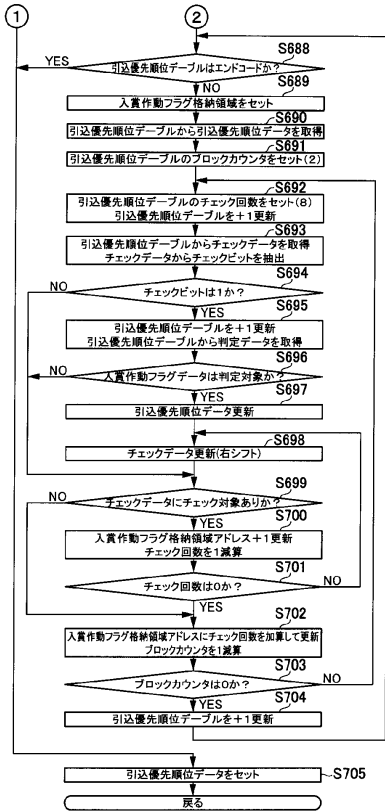
20

30

40

50

【図 1 3 5】



【図 1 3 6】

```

; @>>> ANY 役対応処理
DEC D ; 第二停止チェック時か?
JR NZ, STSR, GTST_03 ; 違うならジャンプ
;
;
JBIT NZ, 2.E, STSB, GTST_03 ; 第三回期停止後ならジャンプ
;
;
JAND Z, cHT, ANY3W14, STSB, GTST_03 ; ANYを含む役が干渉の可能性なしならジャンプ
;
LD A, B ; 入賞作動フラグ格納領域の位置
JCP Z, A, cANY, WPOS+1, STSB, GTST_10 ; ANYを含む入賞作動フラグチェック時ならジャンプ(停止禁止位置)
; @<<< ANY 役対応処理

;== 停止制御用引込みフラグ設定
LDQ HL, wHITFLAG+11 ; 入賞作動フラグ格納領域
LDO DE, wRTDPBIT ; 引込み要求フラグ格納領域
LD B, cRTDPLENG ; RT作動組合せ表示フラグデータ長(1)
CALLF SB, DAND, D0 ; 論理積演算処理

;== 引込優先順位テーブルの生成
LD D, 1 ; 引込み優先順位データ初期値 セット(1)
LDQ HL, (wPLVLADR) ; 引込み優先順位テーブルセル(PLVLTB00~dPLVLTB05)

```

10

20

【図 1 3 7】

```

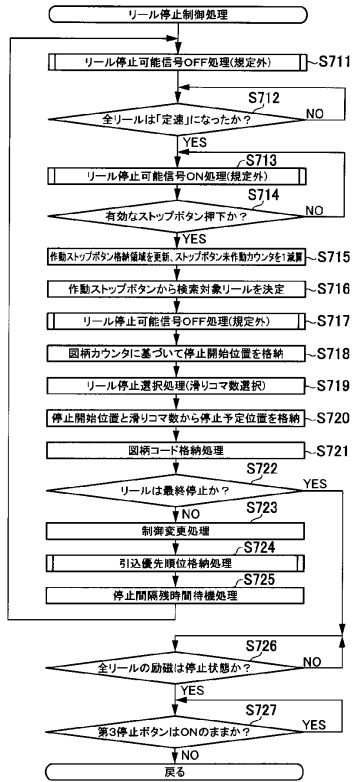
; 引込優先順位テーブル
dPLVLTB00 EQU $
DB 03EH ; 引込優先順位データ
DB 10001000B ; チェックデータ
DB 080H, 003H ; フラグ判定データ
DB 00000001B ; チェックデータ
DB 040H ; フラグ判定データ
;
DB 01EH ; 引込優先順位データ
DB 10001011B ; チェックデータ
DB 048H, 008H, 004H, 034H ; フラグ判定データ
DB 00000000B ; チェックデータ
;
DB 002H ; 引込優先順位データ
DB 00000000B ; チェックデータ
DB 00000100B ; チェックデータ
DB 0C0H ; フラグ判定データ
;
DB 000H ; エンドコード

dPLVLTB01 EQU $
DB 01EH ; 引込優先順位データ

dPLVLTB05 EQU $
DB 00EH ; 引込優先順位データ
DB 00010000B ; チェックデータ
DB 004H ; フラグ判定データ
DB 00000000B ; チェックデータ
;
DB 006H ; 引込優先順位データ
DB 00001000B ; チェックデータ
DB 080H ; フラグ判定データ
DB 00000000B ; チェックデータ
;
DB 002H ; 引込優先順位データ
DB 00010000B ; チェックデータ
DB 002H ; フラグ判定データ
DB 00000000B ; チェックデータ
;
DB 000H ; エンドコード

```

【図 1 3 8】



30

40

50

【図 1 3 9】

```

LD      HLOT_SLEDOF_00      ; リールストップ可能信号OFF処理(規格外メニュー)
MN_STOP_01:
;==== スタック許可のフラグ
CALLF  SB_OCALK_00        ; 領域外呼び出し処理
CALLF  SB_FXCHK_00        ; 戻還フラグ処理
JR      CMM_STOP_00        ; 戻還領域でないならジャンプ
LDQ    HL_wSLAMPBIT        ; スタック番号と番号格納格納領域セット
LDO    A,(WRBUF+4,ONE+eFLSTPBTN) ; 入力ポートオンオフデータ取得(ストップメニュー入力)
AND    (HL)                ; 有効な回転中のストップメニュー入力?
JR      NZ,MN_STOP_02      ; そうならジャンプ (停止制御開始)
LD      HLOT_SLEDON_00     ; リールストップ可能信号ON処理(規格外メニュー)
JR      MN_STOP_01
;==== スタック開始時の初期化 (同時降下時は左の回転を最優先)
MN_STOP_02:
LDQ    (wR1,CTRL-(wR2,CTRL-wR1,CTRL)) ; 第一回周 制御時→格納領域セット
LDO    C,00BH              ; スタックフラグ初期化(セット)
LD

```

【図 1 4 0】

```

MN_STOP_05
:
;==== 回調制御終了のフラグ
LDQ    A,(LOW.wR1,TIM)
ORQ    (LOW.wR2,TIM)
ORQ    (LOW.wR3,TIM)        ; 全回調の動磁がマージ合成
JR      NZ,MN_STOP_05      ; 停止終了でないならジャンプ

```

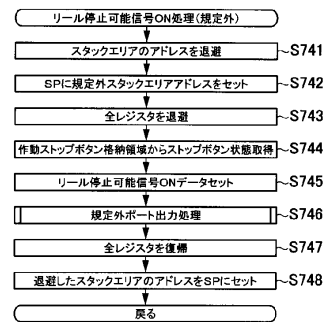
10

20

【図 1 4 1】



【図 1 4 2】

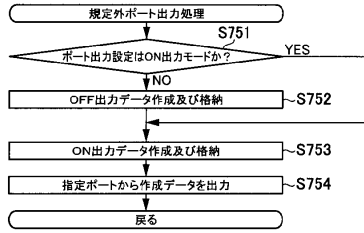


30

40

50

【 図 1 4 3 】



【 図 1 4 4 】

```

OSB_OUTX_00:
LD   A,C           ;出力モード、出力ポートデータセット
:
PUSH GPR           ;レジスタ退避
SRL  C             ;出力ポートデータ抽出
:
LD   HL,woXOUT0
ADDWB HL,C         ;領域外出力ポートバックアップ領域セット
RRCA                ;出力モード抽出(CV7ラフセット/リセット)
LD   A,B           ;出力制御データセット
== 出力オン/オフチェック
JR   C,OSB_OUTX_01 ;出力オンモードならジャンプ
:
XOR  (HL)          ;出力オフモードデータ作成
AND  (HL)
LD   (HL),A
OSB_OUTX_01:
OR   (HL)          ;出力オンモードデータ作成
LD   (HL),A
== メモリマップT/I/O出力
:
CALL OI1_OUTX_00  ;試験信号出力処理
:
POP  GPR           ;レジスタ復帰
XOR  A             ;クリア(兼 CV7ラフクリア)
:
RET
  
```

OFF出力データ出力作成の一例

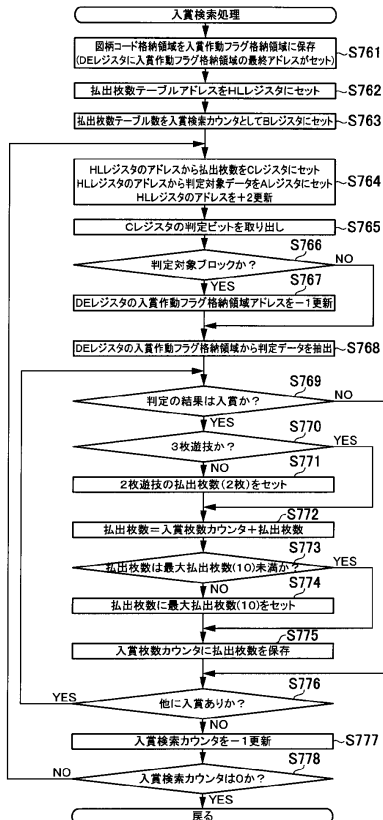
```

00010111 出力データ
01010011  バックアップデータ
01000100  XORした結果      OFF出力データ作成
01000000  ANDした結果
01000000  ORした結果       ON出力データ作成
  
```

10

20

【 図 1 4 5 】



【 図 1 4 6 】

```

MN_CKLN_00:
LDQ  HL,LOW.wXARAY ;図柄コード格納領域セット
LDQ  DE,LOW.wHITFLAG ;入賞作動フラグ格納領域セット
LD   B,12
LDIRS                ;図柄コードを入賞作動フラグに転送
:
LD   HL,dPAYNUMTB
LD   B,5             ;払出枚数データテーブル  
;払出枚数データテーブルサイズ
MN_CKLN_01:
LDIN AC,(HL)        ;C = 払出枚数, A = 判定データ
SRL  C              ;バイト変換タミングか?
JR   NC,MN_CKLN_02 ;違うならジャンプ
MN_CKLN_02:
DEC  DE             ;判定ビットを抽出
AND  (DE)
MN_CKLN_03:
JSLAA NG,MN_CKLN_06 ;入賞なしならジャンプ
== 入賞 3枚 加算(重複入賞 対応)
PUSH AF
LDQ  A,(LOW.wMEDLCTR) ;3枚カウンタ取得
JCP  Z,A,3,MN_CKLN_04 ;3枚遊技時ならジャンプ
LD   C,2            ;2枚遊技の払い出し枚数セット
MN_CKLN_04:
LDQ  A,(LOW.wHITCTR) ;入賞枚数カウンタ取得
ADD  A,C            ;払い出し枚数加算
:
JCP  C,A,9,MN_CKLN_05 ;獲得枚数の最大値未満ならジャンプ
LD   A,9            ;獲得枚数の最大値セット
MN_CKLN_05:
LDQ  (LOW.wHITCTR),A ;入賞枚数カウンタ保存
POP  AF
MN_CKLN_06:
JR   NZ,MN_CKLN_03 ;他に入賞ありならジャンプ
:
DJNZ RET            ;MN_CKLN_01
  
```

30

40

50

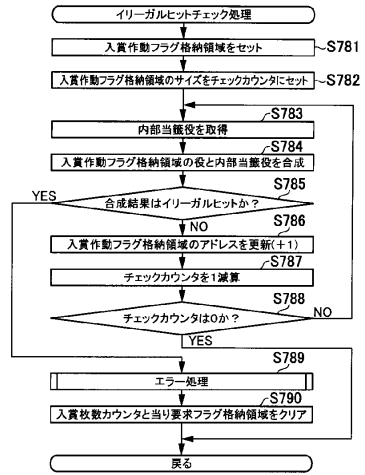
【図 147】

```

: 払出枚数データテーブル
oPAYNUMTB EQU $
DB 9*2 + 0, 111111110B : @@ F.RB 役1~4
DB 9*2 + 0, 111111111B : @@ F.3 振へル, 1st, F.3 振へル, 2nd, F.3 振へル, 3rd, F.共済へル
DB 1*2 + 0, 111111110B : @@ F.5*1, F.5*2
DB 3*2 + 0, 111111110B : @@ F.5*1, F.5*2
DB 2*2 + 0, 111110000B : @@ 弱子か, 強子か, F.特殊1~3
oPAYNUMTBNUM EQU ($-oPAYNUMTB)/2

```

【図 148】



10

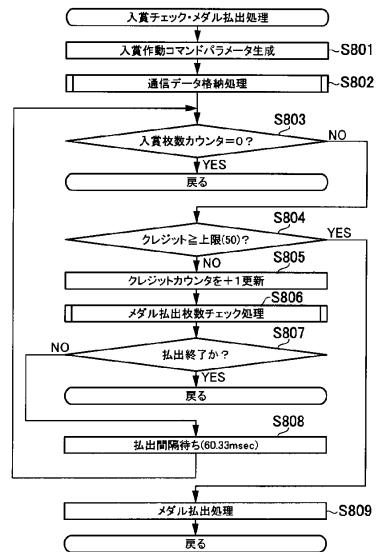
【図 149】

```

:イリーガルチェック処理
MN_ILCK_00:
LDQ HL, LOW.wHITFLAG : 入賞作動フラグ格納領域セット
LD B, WV LENG : 入賞作動フラグデータ長セット
:
MN_ILCK_01:
LD A, (HL+(wWAVEBIT-wHITFLAG)) : 当り要求フラグ取得
XOR (HL) : イリーガルヒットはあるか?
AND (HL) : 入賞作動フラグと合成
JR NZ, MN_ILCK_02 : そうならジャンプ
:
INC HL
DJNZ MN_ILCK_01
RET
:
--- イリーガルヒットエラーの表示
MN_ILCK_02:
LD E, 7 : イリーガルヒットエラーコードセット
CALLF SB_ERRS_00 : エラー処理

```

【図 150】



20

30

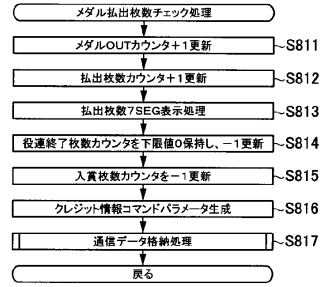
40

50

【 151 】

MN HCPS.02			
== 払出の制御			
CALLF	SB.CDMX.00	:	クレジットは上限に達しているか?
LD	A.SB.PAY.HIT	:	入金残行一対セット
JR	NC.SB.MPAY.00	:	そうならメダル払出処理へ
:			
INC	(HL)	:	[wCREDIT] クレジットカウンター加算更新
:			
CALLF	SB.MPCK.00	:	メダル払出枚数チェック処理
RET	Z	:	払出終了ならリターン
:			
LD	BC.TM.PAYC	:	
RST	SB.WIBC.00	:	払出間隔待ち
:			
JR	MN.HCPS.02	:	払出処理継続

【 152 】

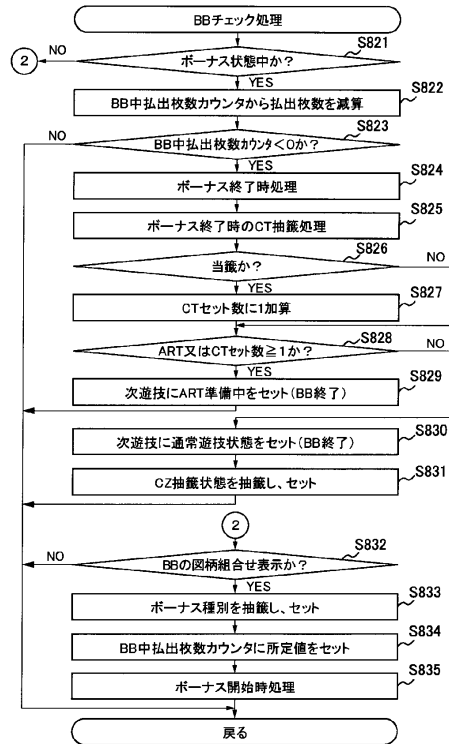


10

【 153 】

A			
INCQ	(LOW.wCTR.PAY)	:	メダルOUTカウンタ要求加算更新
DEC	HL	:	[wPAYNUM]
INC	(HL)	:	払出枚数カウンタ加算更新
:			
LD	A.(HL)	:	払出枚数表示データを取得
CALLF	SR.PAY7.00	:	払出枚数7SEG表示処理
:			
LDQ	HL.LOW.wGETCTR	:	役達終了枚数管理カウンタセット
DQPLD	(HL),0	:	役達終了枚数管理カウンタ更新(下限值なら0保持)
:			
B			
LDQ	HL.(LOW.wPAYNUM)	:	パラメータ1(払出枚数カウンタ)
LDQ	C.(LOW.wCREDIT)	:	パラメータ5セット(クレジットカウンタ)
:			
LD	A.wPC.CRST	:	クレジット情報コマンドセット
RST	SB.OUTP.00	:	通信データ格納処理

【 154 】



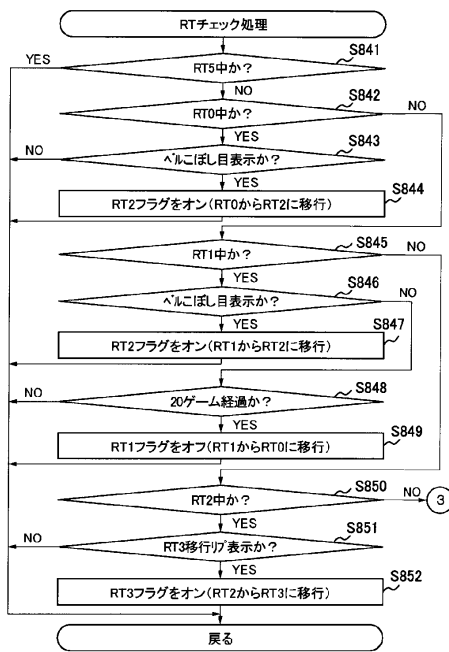
20

30

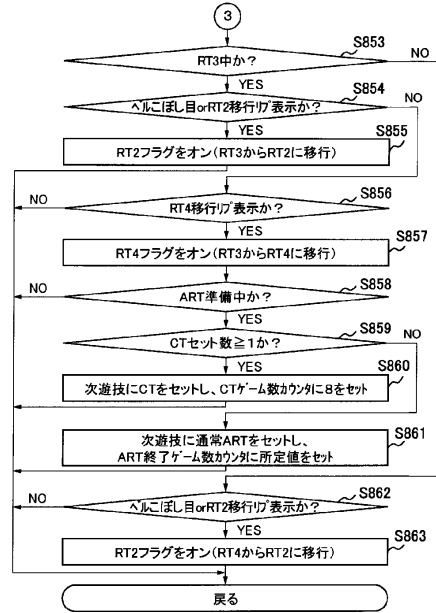
40

50

【図 155】



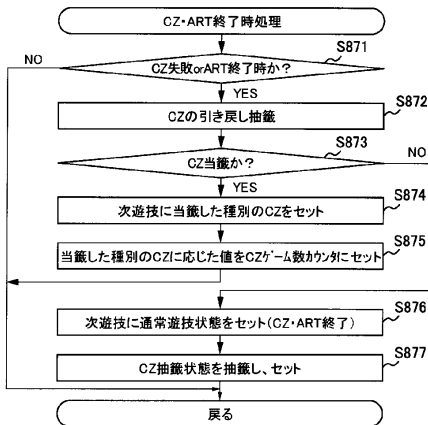
【図 156】



10

20

【図 157】



【図 158】

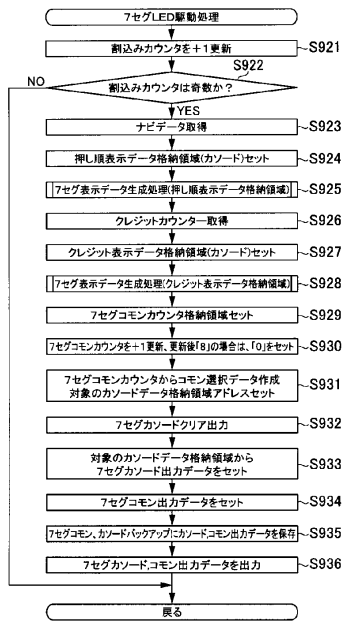


30

40

50

【図 159】



【図 160】

```

A
H,7SEG,00:
LDQ A,(wNAVPTN) ; 7セグカソード格納領域
LDQ DE,wLED_POD_M ; 押し順表示データ格納領域セット
CALLF SB_DSP7_00 ; 7SEG表示処理

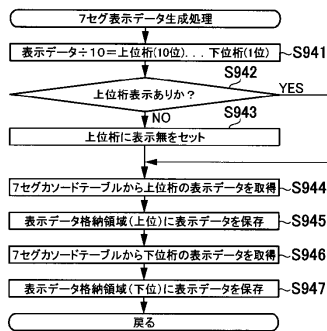
B
;== 7SEGカソード割付
;
LDQ DE,LOW(wXOUT0+ePR_SEGCOM) ; 7SEGカソードバックアップ格納領域セット
LD A,(DE) ; 出力ポートバックアップ取得
AND eBX_SEGMSK ; 7SEGカソード割付
LD (ePA_SEGCOM),A

;== 7SEGカソード出力
;
OR C ; 7SEGカソード出力データセット
LD C,A ; 7SEGカソード出力データセット
LD B,(HL) ; 7SEGカソード出力データセット
LD (DE),BC ; 7SEGカソードバックアップにカソード共通出力データを保存
LD (ePA_SEGCOM),BC ; 7SEGカソード出力
  
```

10

20

【図 161】



【図 162】

```

SB_DSP7_00:
DIV C,A,10 ; A=(商)上位表示データ
; C=(余)下位表示データ
; 上位表示ありならジャンプ
JT NZ,A,SB_DSP7_01
;
LD A,10 ; 上位に無表示データ[10]セット
SB_DSP7_01:
CALLF SB_DCD7_00 ; 7SEGコード処理(上位)
LD AC ; 7SEGコードデータ(下位)セット
SB_DCD7_00:
LD HL,dSEQPARAM ; 7SEGカソードテーブルセット
RST SB_ADRA_00 ; 表示データ格納
LD (DE),A ; 表示データ格納
JNC DE ; 次の格納領域へ
RET
  
```

30

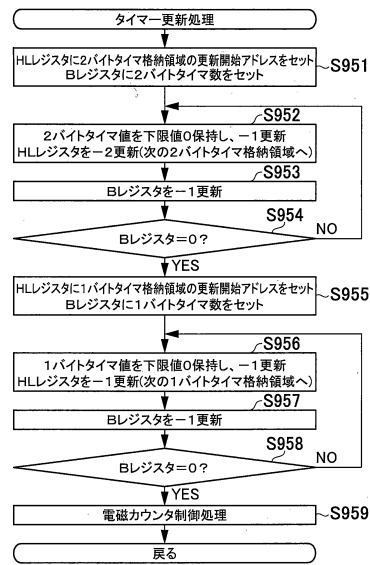
40

50

【図 163】

JSEGPARM	EQU	\$	
DB	01111110B	:	"0" 表示
DB	00110000B	:	"1" 表示
DB	01101101B	:	"2" 表示
DB	01111001B	:	"3" 表示
DB	00110011B	:	"4" 表示
DB	01011011B	:	"5" 表示
DB	01011111B	:	"6" 表示
DB	01110000B	:	"7" 表示
DB	01111111B	:	"8" 表示
DB	01111011B	:	"9" 表示
DB	00000000B	:	表示なし(テラ)

【図 164】



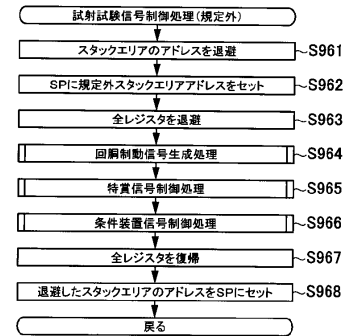
10

20

【図 165】

i1_TIME_00:	LD	HL, LOW(wXTIMER2BS-2)	: [wLOFF_TMR]
:	LD	B, cTIMER2BL	
i1_TIME_01:	DCPWLD	(HL), 0	: タイマ更新 (下限値なら0保持)
:	DEC	HL	: 次のタイマ格納領域へ
:	DEC	HL	
:	DJNZ	i1_TIME_01	

【図 166】

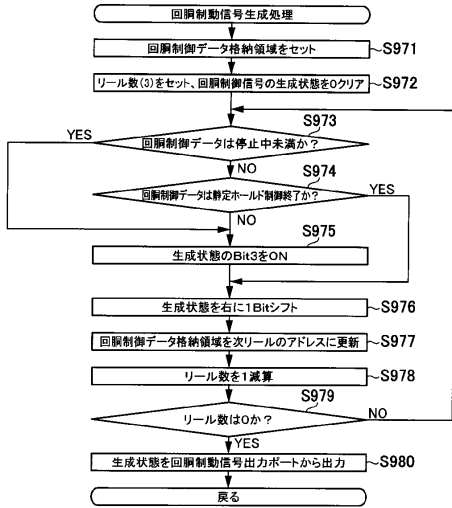


30

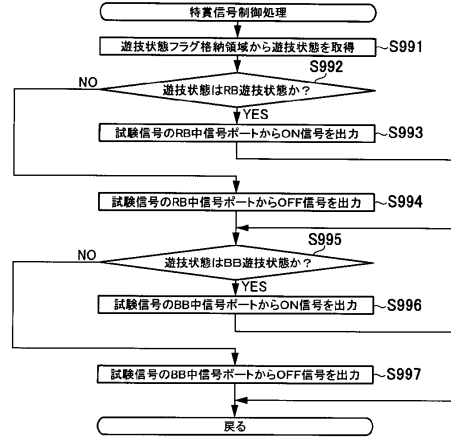
40

50

【 図 1 6 7 】



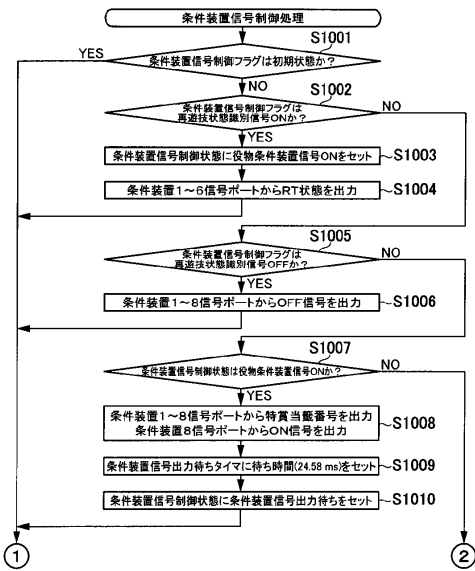
【 図 1 6 8 】



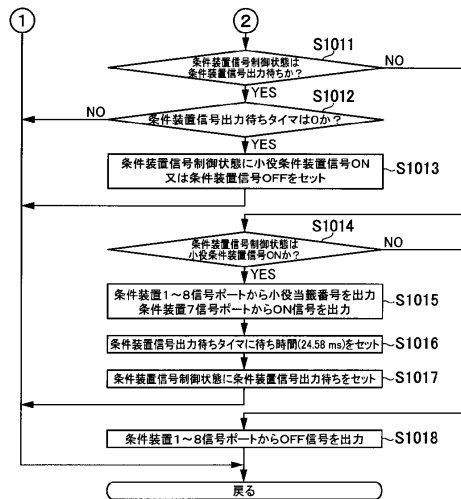
10

20

【 図 1 6 9 】



【 図 1 7 0 】

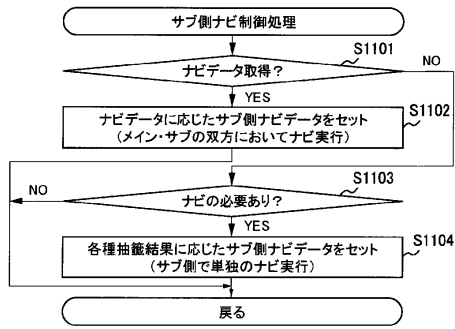


30

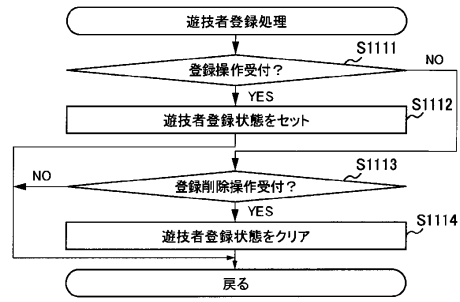
40

50

【図 171】

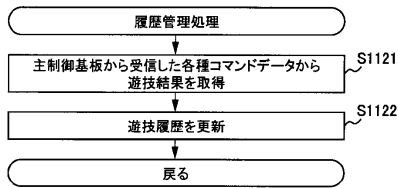


【図 172】

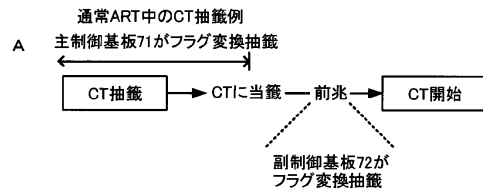


10

【図 173】



【図 174】



20

B CT前兆中のフラグ変換抽籤テーブル

内容	当籤役
	非当籤
当籤	極高

※ 確率分母256

30

40

50

【 図 1 7 5 】

内部当籤役と停止図柄組合せ(略称)との対応関係

当籤役	第1停止操作		
	左(1st)	中(2nd)	右(3rd)
F.確リリア	チリリ(3連)	リプレイ	チリリ(2連)
F.1確リリア	チリリ(2連)	リプレイ	チリリ(3連)

通常ART中のフラグ変換抽籤とサブ側のナビとの関係

当籤役	フラグ変換抽籤		
	1段階目非当籤	1段階目当籤 2段階目非当籤	1段階目当籤 2段階目当籤
F.確リリア	リプレイナビ (第1停止操作中)	チリリ編リナビ (第1停止操作右)	チリリ揃いナビ (第1停止操作左)
F.1確リリア	リプレイナビ (第1停止操作中)	チリリ編リナビ (第1停止操作左)	チリリ揃いナビ (第1停止操作右)

【 図 1 7 6 】

ナビゲームの対応関係

RT5中(8B1フラグ間)

内容	メイン側ナビ															
	1	2	3	...	10	11	12	13	14	...	10	11	12	13	14	...
白7ナビ
ベルナビ

RT5中(8B2フラグ間)

内容	メイン側ナビ															
	1	2	3	...	10	11	12	13	14	...	10	11	12	13	14	...
青7ナビ
ベルナビ

10

20

【 図 1 7 7 】

押し順とロック状態との関係

当籤役	押し順					
	左-中-右	左-右-中	中-左-右	中-右-左	右-左-中	右-中-左
F.維持リプA	0⇒1	維持	維持	維持	維持	1⇒0
F.維持リプB	維持	維持	0⇒1	維持	維持	維持
...

※ ロック状態0: ロックし難い状態
 ロック状態1: ロックし易い状態

ロック状態を移行するための報知

ロックによる利益	報知抽籤	報知手段	
		指示モニタ	表示装置
あり	メイン側で抽籤	○ 報知	○ 報知
なし	サブ側で抽籤	× 報知せず	○ 報知

30

40

50

フロントページの続き

- (56)参考文献 特開 2011 - 098151 (JP, A)
特開 2013 - 180074 (JP, A)
特開 2014 - 230748 (JP, A)
- (58)調査した分野 (Int.Cl., DB名)
A63F5/04、7/02