

(19)日本国特許庁(JP)

## (12)特許公報(B2)

(11)特許番号  
特許第7072545号  
(P7072545)

(45)発行日 令和4年5月20日(2022.5.20)

(24)登録日 令和4年5月12日(2022.5.12)

(51)国際特許分類

A 6 3 F 7/02 (2006.01)

F I

A 6 3 F

7/02

3 2 6 Z

A 6 3 F

7/02

3 2 0

請求項の数 1 (全78頁)

(21)出願番号 特願2019-164663(P2019-164663)  
 (22)出願日 令和1年9月10日(2019.9.10)  
 (65)公開番号 特開2021-40884(P2021-40884A)  
 (43)公開日 令和3年3月18日(2021.3.18)  
 審査請求日 令和2年10月26日(2020.10.26)

(73)特許権者 391010943  
 株式会社藤商事  
 大阪府大阪市中央区内本町一丁目1番4号  
 (74)代理人 100100376  
 弁理士 野中 誠一  
 (74)代理人 100142077  
 弁理士 板谷 真之  
 (74)代理人 100143199  
 弁理士 磯邊 毅  
 (72)発明者 柴田 伸美  
 大阪府大阪市中央区内本町一丁目1番4号 株式会社藤商事内  
 審査官 金子 和孝

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】

VDP (Video Display Processor) と、前記VDPを制御するCPUが機能して、画像演出と音声演出を含んだ所定の演出動作が実行される遊技機であって、前記CPUがアクセス可能なメモリ空間には、前記CPUを含むCPU回路の外部に位置して、データバス幅やメモリ種別を各々規定可能な複数のアドレス空間が含まれ、前記アドレス空間は、所定のキャッシュメモリを経由して、アクセス可能な第1空間と、前記キャッシュメモリを経由することなく、アクセス可能な第2空間と、に区分され、前記アドレス空間には、アドレス情報の一部を構成する特殊ビットの相違により区別される、N個の第1空間SP1iと、N個の第2空間SP2iとが設けられ(1 ≤ i ≤ N)、前記第1空間SP1i又は前記第2空間SP2iにアクセスする場合に、前記特殊ビット以外が同一である限り、実際のメモリデバイスにおけるアクセス先が共通するよう構成されてい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、遊技動作に起因する抽選処理を行い、その抽選結果に対応する画像演出を実行する遊技機に関し、特に、迫力ある画像演出を安定して実行できる遊技機に関する。

【背景技術】

【0002】

パチンコ機などの弾球遊技機は、遊技盤に設けた図柄始動口と、複数の表示図柄による一連の図柄変動態様を表示する図柄表示部と、開閉板が開閉される大入賞口などを備えて構成されている。そして、図柄始動口に設けられた検出スイッチが遊技球の通過を検出すると入賞状態となり、遊技球が賞球として払出された後、図柄表示部では表示図柄が所定時間変動される。その後、7・7・7などの所定の態様で図柄が停止すると大当たり状態となり、大入賞口が繰返し開放されて、遊技者に有利な遊技状態を発生させている。

【0003】

このような遊技状態を発生させるか否かは、図柄始動口に遊技球が入賞したことを条件に実行される大当たり抽選で決定されており、上記の図柄変動動作は、この抽選結果を踏まえたものとなっている。例えば、抽選結果が当選状態である場合には、リーチアクションなどと称される演出動作を20秒前後実行し、その後、特別図柄を整列させている。一方、ハズレ状態の場合にも、同様のリーチアクションが実行されることがあり、この場合には、遊技者は、大当たり状態になることを強く念じつつ演出動作の推移を注視することになる。そして、図柄変動動作の終了時に、停止ラインに所定図柄が揃えば、大当たり状態であることが遊技者に保証されたことになる。

【先行技術文献】

【特許文献】

【0004】

【文献】特開2017-093633号公報

特開2017-093632号公報

特開2016-159030号公報

特開2016-159029号公報

【発明の概要】

【発明が解決しようとする課題】

【0005】

この種の遊技機では、各種の演出を複雑化かつ豊富化したいところ、特に、画像演出については、その要請が高い。そこで、出願人は、各種の提案をしているが（引用文献1～引用文献4）、画像演出の更なる高度化や、画像演出制御を中心とした各種の演出制御動作の更なる改善が望まれるところである。

【0006】

本発明は、上記の課題に鑑みてなされたものであり、画像演出制御を中心とした各種の演出制御動作が更に改善された遊技機を提供することを目的とする。

【課題を解決するための手段】

【0007】

上記の目的を達成するため、本発明に係る遊技機は、VDP（Video Display Processor）と、前記VDPを制御するCPUが機能して、画像演出と音声演出を含んだ所定の演出動作が実行される遊技機であって、前記CPUがアクセス可能なメモリ空間には、前記CPUを含むCPU回路の外部に位置して、データバス幅やメモリ種別を各々規定可能な複数のアドレス空間が含まれ、前記アドレス空間は、所定のキャッシュメモリを経由して、アクセス可能な第1空間と、前記キャッシュメモリを経由することなく、アクセス可能な第2空間と、に区分され、前記アドレス空間には、アドレス情報の一部を構成する特殊ビットの相違により区別される、N個の第1空間SP1*i*と、N個の第2空間SP2*i*とが設けられ（ $1 \leq i \leq N$ ）、前記第1空間SP1*i*又は前記第2空間SP2*i*にアクセスする場合に、前記特殊ビット以外が同一である限り、実際のメモリデバイスにおけるアクセス先が共通するよう構成されている。

【発明の効果】

【0008】

上記した本発明によれば、第1空間をアクセスしてキャッシュメモリを活用することで、外部メモリをアクセスする場合に、制御処理の迅速化が図れる。また、キャッシュ効果が望めない場合や、キャッシュ処理の弊害が懸念される場合には、第2空間をアクセスする

10

20

30

40

50

ことで適切に対処できる。すなわち、キャッシュ無効設定や、キャッシュ有効設定のプログラム処理を繰り返す必要がない。

【図面の簡単な説明】

【0009】

【図1】本実施例のパチンコ機を示す斜視図である。

【図2】図1の遊技機の遊技領域を示す正面図である。

【図3】図1の遊技機の全体回路構成を示すブロック図である。

【図4】図1の遊技機について、演出制御部の回路構成をやや詳細に示すブロック図である。

【図5】演出制御部を構成する複合チップを説明する図面である。

10

【図6】図4に示すCPU回路の内部構成を示すブロック図である。

【図7】CPU回路の内蔵CPU（演出制御CPU）のメモリマップを図示したものである。

【図8】DMACについて、各種の転送動作モード（a）～（b）と、転送動作手順（c）～（e）を説明する図面である。

【図9】インデックス空間、インデックステーブル、仮想描画空間、及び、描画領域について説明する図面である。

【図10】データ転送回路の内部構成を、関連する回路構成と共に記載したブロック図である。

【図11】表示回路の内部構成を、関連する回路構成と共に記載したブロック図である。

20

【図12】CPUリセット後のパワーリセット動作を説明するフローチャートである。

【図13】図12の一部であるメモリセクション初期化処理を説明するフローチャートである。

【図14】図12の一部であるメイン制御処理と、割込み処理を説明するフローチャートである。

【図15】メイン制御処理の一部であるCGROMの初期化処理を説明するフローチャートである。

【図16】別の割込み処理について、処理内容の一部を説明するフローチャートである。

【図17】プリローダを使用しない場合について、演出制御CPU63の制御動作を説明するフローチャートである。

30

【図18】ディスプレイリストの構成を説明する図面である。

【図19】ディスプレイリストDLを発行するDL発行処理を示すフローチャートである。

【図20】図19の動作にDMACが関与する場合の動作を説明するフローチャートである。

【図21】図20の処理に続く動作を説明するフローチャートである。

【図22】プリローダを使用する場合について、演出制御CPU63の制御動作を説明するフローチャートである。

【図23】図22の一部を説明するフローチャートである。

【図24】図22の別の一部を説明するフローチャートである。

【図25】プリローダを使用しない実施例について、VDP各部の動作を示すタイムチャートである。

40

【図26】プリローダを使用する実施例について、VDP各部の動作を示すタイムチャートである。

【図27】別の実施例について全体回路構成を示すブロック図である。

【図28】図27の一部をやや詳細に示すブロック図である。

【図29】別の実施例について、動作内容を説明するフローチャートである。

【図30】更に別の実施例を説明する図面である。

【図31】設定値を繰り返し設定する実施例を説明する図面である。

【図32】内蔵された音声回路を使用する実施例の回路構成を説明する図面である。

【図33】音声回路の初期設定動作を説明するフローチャートである。

50

【図 3 4】CPUリセット後のパワーリセット動作について別の実施例を説明する図面である。

【図 3 5】メモリREAD動作とメモリWRITE 動作の一例を示すタイムチャートである。

【発明を実施するための形態】

【0010】

以下、実施例に基づいて本発明を詳細に説明する。図 1 は、本実施例のパチンコ機 GM を示す斜視図である。このパチンコ機 GM は、島構造体に着脱可能に装着される矩形枠状の木製外枠 1 と、外枠 1 に固着されたヒンジ 2 を介して開閉可能に枢着される前枠 3 とで構成されている。この前枠 3 には、遊技盤 5 が、裏側からではなく、表側から着脱自在に装着され、その前側には、ガラス扉 6 と前面板 7 とが夫々開閉自在に枢着されている。

10

【0011】

ガラス扉 6 の外周には、LED ランプなどによる電飾ランプが、略 C 字状に配置されている。一方、ガラス扉 6 の上部左右位置と下側には、全 3 個のスピーカが配置されている。上部に配置された 2 個のスピーカは、各々、左右チャンネル R, L の音声を出力し、下側のスピーカは低音を出力するよう構成されている。

【0012】

前面板 7 には、発射用の遊技球を貯留する上皿 8 が装着され、前枠 3 の下部には、上皿 8 から溢れ出し又は抜き取った遊技球を貯留する下皿 9 と、発射ハンドル 10 とが設けられている。発射ハンドル 10 は発射モータと連動しており、発射ハンドル 10 の回動角度に応じて動作する打撃槌によって遊技球が発射される。

20

【0013】

上皿 8 の外周面には、チャンスボタン 11 が設けられている。このチャンスボタン 11 は、遊技者の左手で操作できる位置に設けられており、遊技者は、発射ハンドル 10 から右手を離すことなくチャンスボタン 11 を操作できる。このチャンスボタン 11 は、通常時には機能していないが、ゲーム状態がボタンチャンス状態となると内蔵ランプが点灯されて操作可能となる。なお、ボタンチャンス状態は、必要に応じて設けられるゲーム状態である。

【0014】

また、チャンスボタン 11 の下方には、ロータリースイッチ型の音量スイッチ V L S W が配置されており、遊技者が音量スイッチ V L S W を操作することで、無音レベル (= 0) から最高レベル (= 7) まで、スピーカ音量を 8 段階に調整できるようになっている。なお、スピーカの音量は、係員だけが操作可能な設定スイッチ (不図示) によって初期設定されており、遊技者が音量スイッチ V L S W を操作しない限り、初期設定音量が維持される。また、異常事態が発生したことを報知する異常報知音は、係員による初期設定音量や、遊技者の設定音量に拘らず最高音量で放音される。

30

【0015】

上皿 8 の右部には、カード式球貸し機に対する球貸し操作用の操作パネル 12 が設けられ、カード残額を 3 桁の数字で表示する度数表示部と、所定金額分の遊技球の球貸しを指示する球貸しスイッチと、ゲーム終了時にカードの返却を指令する返却スイッチとが設けられている。

40

【0016】

図 2 に示すように、遊技盤 5 の表面には、金属製の外レールと内レールとからなるガイドレール 13 が環状に設けられ、その略中央には、中央開口 H O が設けられている。そして、中央開口 H O の下方には、不図示の可動演出体が隠蔽状態で収納されており、可動予告演出時には、その可動演出体が上昇して露出状態となることで、所定の信頼度の予告演出を実現している。ここで、予告演出とは、遊技者に有利な大当たり状態が招来することを不確定に報知する演出であり、予告演出の信頼度とは、大当たり状態が招来する確率を意味している。

【0017】

中央開口 H O には、大型 (例えば、横 1280 × 縦 1024 ピクセル) の液晶カラーディ

50

スプレイ（ＬＣＤ）で構成されたメイン表示装置ＤＳ１が配置され、メイン表示装置ＤＳ１の右側には、小型（例えば、横４８０×縦８００ピクセル）の液晶カラーディスプレイで構成された可動式のサブ表示装置ＤＳ２が配置されている。メイン表示装置ＤＳ１は、大当たり状態に係わる特定図柄を変動表示すると共に背景画像や各種のキャラクタなどをアニメーション的に表示する装置である。この表示装置ＤＳ１は、中央部に特別図柄表示部Ｄａ～Ｄｃと右上部に普通図柄表示部１９とを有している。そして、特別図柄表示部Ｄａ～Ｄｃでは、大当たり状態の招来を期待させるリーチ演出が実行されることがあり、特別図柄表示部Ｄａ～Ｄｃ及びその周りでは、適宜な予告演出などが実行される。

【００１８】

サブ表示装置ＤＳ２は、通常時には、その表示画面が遊技者に見やすい角度に傾斜した静止状態で画像情報を表示している。但し、所定の予告演出時には、遊技者に見やすい角度に傾斜角度を変えながら、図示の左側に移動する共に、所定の予告画像を表示するようになっている。

10

【００１９】

すなわち、実施例のサブ表示装置ＤＳ２は、単なる表示装置ではなく、予告演出を実行する可動演出体としても機能している。ここで、サブ表示装置ＤＳ２による予告演出は、その信頼度が高く設定されており、遊技者は、大きな期待感をもってサブ表示装置ＤＳ２の移動動作に注目することになる。

【００２０】

ところで、遊技球が落下移動する遊技領域には、第１図柄始動口１５ａ、第２図柄始動口１５ｂ、第１大入賞口１６ａ、第２大入賞口１６ｂ、普通入賞口１７、及び、ゲート１８が配設されている。これらの入賞口１５～１８は、それぞれ内部に検出スイッチを有しており、遊技球の通過を検出できるようになっている。

20

【００２１】

第１図柄始動口１５ａの上部には、導入口ＩＮから進入した遊技球がシーソー状又はルーレット状に移動した後に、第１図柄始動口１５に入賞可能に構成された演出ステージ１４が配置されている。そして、第１図柄始動口１５に遊技球が入賞すると、特別図柄表示部Ｄａ～Ｄｃの変動動作が開始されるよう構成されている。

【００２２】

第２図柄始動口１５ｂは、左右一対の開閉爪を備えた電動式チューリップで開閉されるように構成され、普通図柄表示部１９の変動後の停止図柄が当り図柄を表示した場合には、所定時間だけ、若しくは、所定個数の遊技球を検出するまで、開閉爪が開放されるようになっている。

30

【００２３】

なお、普通図柄表示部１９は、普通図柄を表示するものであり、ゲート１８を通過した遊技球が検出されると、普通図柄が所定時間だけ変動し、遊技球のゲート１８の通過時点において抽出された抽選用乱数値により決定される停止図柄を表示して停止する。

【００２４】

第１大入賞口１６ａは、前後方向に進退するスライド盤を有して構成され、第２大入賞口１６ｂは、下端が軸支されて前方に開放する開閉板を有して構成されている。第１大入賞口１６ａや第２大入賞口１６ｂの動作は、特に限定されないが、この実施例では、第１大入賞口１６ａは、第１図柄始動口１５ａに対応し、第２大入賞口１６ｂは、第１図柄始動口１５ｂに対応するよう構成されている。

40

【００２５】

すなわち、第１図柄始動口１５ａに遊技球が入賞すると、特別図柄表示部Ｄａ～Ｄｃの変動動作が開始され、その後、所定の大当たり図柄が特別図柄表示部Ｄａ～Ｄｃに整列すると、第１大当たりたる特別遊技が開始され、第１大入賞口１６ａのスライド盤が、前方に開放されて遊技球の入賞が容易化される。

【００２６】

一方、第２図柄始動口１５ｂへの遊技球の入賞によって開始された変動動作の結果、所定

50

の大当り図柄が特別図柄表示部 D a ~ D c に整列すると、第 2 大当りたる特別遊技が開始され、第 2 大入賞口 1 6 b の開閉板が開放されて遊技球の入賞が容易化される。特別遊技（大当り状態）の遊技価値は、整列する大当り図柄などに対応して種々相違するが、何れの遊技価値が付与されるかは、遊技球の入賞タイミングに応じた抽選結果に基づいて予め決定される。

#### 【0027】

典型的な大当り状態では、大入賞口 1 6 の開閉板が開放された後、所定時間が経過し、又は所定数（例えば 1 0 個）の遊技球が入賞すると開閉板が閉じる。このような動作は、最大で例えば 1 5 回まで継続され、遊技者に有利な状態に制御される。なお、特別図柄表示部 D a ~ D c の変動後の停止図柄が特別図柄のうちの特定図柄であった場合には、特別遊技の終了後のゲームが高確率状態（確変状態）となるという特典が付与される。

#### 【0028】

図 3 は、上記した各動作を実現するパチンコ機 G M の全体回路構成を示すブロック図であり、図 4（a）はその一部を詳細に図示したものである。

#### 【0029】

図 3 に示す通り、このパチンコ機 G M は、A C 2 4 V を受けて各種の直流電圧や、電源異常信号 A B N 1、A B N 2 を出力する電源基板 2 0 と、遊技制御動作を中心統括的に担う主制御基板 2 1 と、音声演出用の回路素子 S N D を搭載した演出インタフェース基板 2 2 と、主制御基板 2 1 から受けた制御コマンド C M D に基づいてランプ演出、音声演出、及び画像演出を統一的に実行する演出制御基板 2 3 と、演出制御基板 2 3 と表示装置 D S 1、D S 2 の間に位置する液晶インタフェース基板 2 4 と、主制御基板 2 1 から受けた制御コマンド C M D ' に基づいて払出モータ M を制御して遊技球を払い出す払出制御基板 2 5 と、遊技者の操作に応答して遊技球を発射させる発射制御基板 2 6 と、を中心に構成されている。

#### 【0030】

本実施例の場合、演出インタフェース基板 2 2 と、演出制御基板 2 3 と、液晶インタフェース基板 2 4 とは、配線ケーブルを経由することなく、雄型コネクタと雌型コネクタとを直結されている。そのため、各電子回路の回路構成を複雑高度化しても基板全体の収納空間を最小化できると共に、接続ラインを最短化することで耐ノイズ性を高めることができる。

#### 【0031】

図示の通り、主制御基板 2 1 が出力する制御コマンド C M D ' は、主基板中継基板 3 3 を経由して、払出制御基板 2 5 に伝送される。一方、主制御基板 2 1 が出力する制御コマンド C M D は、演出インタフェース基板 2 2 を経由して演出制御基板 2 3 に伝送される。制御コマンド C M D、C M D ' は、何れも 1 6 b i t 長であるが、8 b i t 長毎に 2 回に分けて平行送信される。

#### 【0032】

主制御基板 2 1 と払出制御基板 2 5 には、ワンチップマイコンを含むコンピュータ回路が搭載されている。また、演出制御基板 2 3 には、V D P 回路（Video Display Processor）5 2 や内蔵 C P U 回路 5 1 などのコンピュータ回路が内蔵された複合チップ 5 0 が搭載されている。そこで、これらの制御基板 2 1、2 5、2 3 と、演出インタフェース基板 2 2 や液晶インタフェース基板 2 4 に搭載された回路、及びその回路によって実現される動作を機能的に総称して、本明細書では、主制御部 2 1、演出制御部 2 3、及び払出制御部 2 5 と言うことがある。なお、主制御部 2 1 に対して、演出制御部 2 3 と、払出制御部 2 5 がサブ制御部となる。

#### 【0033】

また、このパチンコ機 G M は、図 3 の破線で囲む枠側部材 G M 1 と、遊技盤 5 の背面に固定された盤側部材 G M 2 とに大別されている。枠側部材 G M 1 には、ガラス屏 6 や前面板 7 が枢着された前枠 3 と、その外側の木製外枠 1 とが含まれており、機種の変更に拘わらず、長期間にわたって遊技ホールに固定的に設置される。一方、盤側部材 G M 2 は、機種

10

20

30

40

50

変更に対応して交換され、新たな盤側部材GM2が、元の盤側部材の代わりに枠側部材GM1に取り付けられる。なお、枠側部材1を除く全てが、盤側部材GM2である。

【0034】

図3の破線枠に示す通り、枠側部材GM1には、電源基板20と、払出制御基板25と、発射制御基板26と、枠中継基板36とが含まれており、これらの回路基板が、前枠3の適所に各々固定されている。一方、遊技盤5の背面には、主制御基板21、演出制御基板23が、表示装置DS1、DS2やその他の回路基板と共に固定されている。そして、枠側部材GM1と盤側部材GM2とは、一箇所に集中配置された接続コネクタC1～C4によって電氣的に接続されている。

【0035】

電源基板20は、接続コネクタC2を通して、主基板中継基板33に接続され、接続コネクタC3を通して、電源中継基板34に接続されている。そして、電源基板20には、交流電源の投入と遮断とを監視する電源監視部MNTが設けられている。電源監視部MNTは、交流電源の遮断を検知すると、電源異常信号ABN1、ABN2を、直ちにLレベルに遷移させる。なお、電源異常信号ABN1、ABN2は、電源投入後に速やかにHレベルとなる。

【0036】

主基板中継基板33は、電源基板20から出力される電源異常信号ABN1、バックアップ電源BAK、及びDC5V、DC12V、DC35Vを、そのまま主制御部21に出力している。また、電源中継基板34は、電源基板20から受けた交流及び直流の電源電圧DC5V、DC12V、DC35Vを、そのまま演出インタフェース基板22に出力している。

【0037】

図示の通り、演出インタフェース基板22には、音声プロセッサ27などの音声回路SNDが搭載され、演出制御基板23には、VDP回路52や内蔵CPU回路51などのコンピュータ回路が内蔵された複合チップ50が搭載されている。以下、内蔵CPU回路をCPU回路と略称することがある。

【0038】

演出インタフェース基板22には、電源投入時に、電源電圧の上昇を検知して各種のリセット信号RT3、RT4を生成するリセット回路RST3、RST4が搭載されている。まず、リセット回路RST3は、電源基板20から配電された直流電圧12Vと5Vに基づいて、リセット信号RT3を生成している。そして、リセット信号RT3は、音声メモリ28だけを電源リセットして、そのまま演出制御基板23に伝送される。

【0039】

演出制御基板23に伝送されたリセット信号RT3は、図4(a)に示す通り、ANDゲートG1において、WDT(Watch Dog Timer)回路58の出力とAND演算され、システムリセット信号SYSとして、CPU回路51やVDP回路52を電源リセットしている(図4(a)及び図4(d)参照)。

【0040】

リセット回路RST3が生成するリセット信号RT3は、電源投入後、電源リセット信号として所定時間Lレベルを維持した後、Hレベルに上昇する。しかし、その後、直流電圧12V又は直流電圧5Vの何れか一以上が降下した場合(通常は電源遮断時)には、リセット信号RT3のレベル降下に対応して、システムリセット信号SYSもLレベルに降下するので、演出制御基板23のCPU回路51とVDP回路52は動作停止状態となる。

【0041】

このシステムリセット信号SYSは、WDT回路58の出力(正常時にはHレベル)に基づいても変化するので、リセット信号RT3=Hの状態、プログラム暴走時などに起因して、WDT回路58の出力がLレベルに降下することに対応して、システムリセット信号SYSもLレベルに変化して、CPU回路51やVDP回路52を異常リセットする(図4(d)参照)。

10

20

30

40

50

## 【0042】

一方、リセット回路RST4は、電源基板20から配電された5Vを降下して生成された3.3Vに基づいて、リセット信号RT4を生成している。このリセット信号RT4は、電源投入時の電源リセット信号として、音声プロセッサ27を電源リセットしている。

## 【0043】

図示の通り、リセット回路RST4には、演出制御基板23から返送されたシステムリセット信号SYSも供給されているので、CPU回路51やVDP回路52の異常リセット時には、これらの回路の異常リセットに同期して、音声プロセッサ27も異常リセットされる。この結果、音声演出は、画像演出やランプ演出と共に初期状態に戻ることであり、不自然な音声演出が継続するおそれがない。

10

## 【0044】

次に、枠側部材GM1たる払出制御基板25は、中継基板を介することなく、電源基板20に直結されて、主制御部21が受けると同様の電源異常信号ABN2や、バックアップ電源BAKを、その他の電源電圧と共に受けている。また、主制御部21と払出制御部25には、各々、リセット回路RST1、RST2が搭載されており、電源投入時に電源リセット信号が生成され、各コンピュータ回路が電源リセットされるよう構成されている。

## 【0045】

このように、本実施例では、主制御部21と、払出制御部25と、演出インタフェース基板22に、各々、リセット回路RST1～RST4を配置しており、システムリセット信号SYSが回路基板間で伝送されることがない。すなわち、システムリセット信号SYSを伝送する配線ケーブルが存在しないので、配線ケーブルに重畳するノイズによって、コンピュータ回路が異常リセットされるおそれが解消される。

20

## 【0046】

但し、主制御部21や払出制御部25に設けられたリセット回路RST1、RST2は、各々ウォッチドッグタイマを内蔵しており、各制御部21、25のCPUから、定期的なクリアパルスを受けない場合には、各CPUは強制的にリセットされる。

## 【0047】

また、主制御部21には、係員が操作可能な初期化スイッチSWが配置されており、電源投入時、初期化スイッチSWがON操作されたか否かを示すRAMクリア信号CLRが出力されるよう構成されている。このRAMクリア信号CLRは、主制御部21と払出制御部25のワンチップマイコンに伝送され、各制御部21、25のワンチップマイコンの内蔵RAMの全領域を初期設定するか否かを決定している。

30

## 【0048】

また、主制御部21及び払出制御部25は、電源基板20から電源異常信号ABN1、ABN2を受けることによって、停電や営業終了に先立って、必要な終了処理を開始するようになっている。また、バックアップ電源BAKは、営業終了や停電により交流電源24Vが遮断された後も、主制御部21と払出制御部25のワンチップマイコンの内蔵RAMのデータを保持するDC5Vの直流電源である。したがって、主制御部21と払出制御部25は、電源遮断前の遊技動作を電源投入後に再開できることになる（電源バックアップ機能）。このパチンコ機では少なくとも数日は、各ワンチップマイコンのRAMの記憶内容が保持されるよう設計されている。

40

## 【0049】

図3に示す通り、主制御部21は、払出制御部25から、遊技球の払出動作を示す賞球計数信号や、払出動作の異常に係わるステイタス信号CONや、動作開始信号BGNを受信している。ステイタス信号CONには、例えば、補給切れ信号、払出不足エラー信号、下皿満杯信号が含まれる。動作開始信号BGNは、電源投入後、払出制御部25の初期動作が完了したことを主制御部21に通知する信号である。

## 【0050】

また、主制御部21は、遊技盤中継基板32を経由して、遊技盤5の各遊技部品に接続されている。そして、遊技盤上の各入賞口16～18に内蔵された検出スイッチのスイッチ

50

信号を受ける一方、電動式チューリップなどのソレノイド類を駆動している。ソレノイド類や検出スイッチは、主制御部 21 から配電された電源電圧  $V_B$  (12 V) で動作するように構成されている。また、図柄始動口 15 への入賞状態などを示す各スイッチ信号は、電源電圧  $V_B$  (12 V) と電源電圧  $V_{cc}$  (5 V) とで動作するインタフェイス IC で、TTL レベル又は CMOS レベルのスイッチ信号に変換された上で、主制御部 21 に伝送される。

#### 【0051】

先に説明した通り、演出インタフェイス基板 22 と演出制御基板 23 と液晶インタフェイス基板 24 とはコネクタ連結によって一体化されており、演出インタフェイス基板 22 は、電源中継基板 34 を経由して、電源基板 20 から各レベルの直流電圧 (5 V, 12 V, 35 V) を受けている (図 3 及び図 4 (a) 参照)。直流電圧 12 V は、デジタルアンプ 29 の電源電圧であると共に、LED ランプなどの駆動電圧として使用される。また、直流電圧 35 V は、モータの駆動電圧として使用される。

10

#### 【0052】

一方、直流電圧 5 V は、演出インタフェイス基板 22 各所の回路素子の電源電圧として供給されると共に、2つの DC/DC コンバータ DC1, DC2 に供給されて 3.3 V と 1.0 V が生成される (図 4 (a) 参照)。生成された直流電圧 3.3 V と 1.0 V は、各々、I/O (入出力) 用と、チップコア用の電源電圧として音声プロセッサ 27 に供給される。また、直流電圧 3.3 V は、リセット回路 RST4 が生成する電源リセット信号 RST4 の基礎電圧となる。

20

#### 【0053】

演出インタフェイス基板 22 に配電された直流電圧 5 V は、DC/DC コンバータ DC1 で生成された 3.3 V と共に、演出制御基板 23 に配電される。そして、演出制御基板 23 に配電された直流電圧 3.3 V は、電源電圧として、複合チップ 50 や、PROM 53 及び CGROM 55 に供給される。

#### 【0054】

図 4 (a) に示す通り、演出制御基板 23 には、2つの DC/DC コンバータ DC3, DC4 が配置されており、各々に供給される直流電圧 5 V に基づいて、1.5 V と 1.05 V を生成している。ここで、直流電圧 1.05 V は、複合チップ 50 のチップコア用の電源電圧であり、直流電圧 1.5 V は、DRAM 54 との I/O (入出力) 用の電源電圧である。したがって、直流電圧 1.5 V は、電源電圧として、DRAM 54 にも供給される。

30

#### 【0055】

図 3 に示す通り、演出インタフェイス基板 22 は、主制御部 21 から制御コマンド CMD とストローク信号 STB とを受けて、演出制御基板 23 に転送している。より詳細には、図 4 (a) に示す通りであり、制御コマンド CMD とストローク信号 STB は、入力バッファ 40 を経由して、演出制御基板 23 の複合チップ 50 (CPU 回路 51) に転送される。ここで、ストローク信号 STB は、受信割込み信号 IRQ\_CMD であり、演出制御 CPU 63 は、受信割込み信号 IRQ\_CMD を受けて起動される割込み処理プログラム (割込みハンドラ) に基づいて、制御コマンド CMD を取得している。

#### 【0056】

図 4 (a) に示す通り、演出インタフェイス基板 22 の入力バッファ 44 は、枠中継基板 35, 36 からチャンスボタン 11 や音量スイッチ VLSW のスイッチ信号を受け、各スイッチ信号を演出制御基板 23 の CPU 回路 51 に伝送している。具体的には、音量スイッチ VLSW の接点位置 (0~7) を示すエンコーダ出力の 3 bit 長と、チャンスボタン 11 の ON/OFF 状態を示す 1 bit 長を CPU 回路 51 に伝送している。

40

#### 【0057】

また、演出インタフェイス基板 22 には、ランプ駆動基板 30 やモータランプ駆動基板 31 が接続されると共に、枠中継基板 35, 36 を経由して、ランプ駆動基板 37 にも接続されている。図示の通り、ランプ駆動基板 30 に対応して、出力バッファ 42 が配置され、モータランプ駆動基板 31 に対応して、入力バッファ 43a と出力バッファ 43b が配

50

置されている。なお、図4(a)では、便宜上、入力バッファ43aと出力バッファ43bを総称して、入出力バッファ43と記載している。入力バッファ43aは、可動演出体たる役物の現在位置(演出モータM1~Mnの回転位置)を把握する原点センサの出力SN0~SNnを受けて、演出制御基板23のCPU回路51に伝送している。

#### 【0058】

ランプ駆動基板30、モータランプ駆動基板31、及び、ランプ駆動基板37には、同種のドライバICが搭載されており、演出インタフェース基板22は、演出制御基板23から受けるシリアル信号を、各ドライバICに転送している。シリアル信号は、具体的には、ランプ(モータ)駆動信号SDATAとクロック信号CKであり、駆動信号SDATAがクロック同期方式で各ドライバICに伝送され、多数のLEDランプや電飾ランプによるランプ演出や、演出モータM1~Mnによる役物演出が実行される。

10

#### 【0059】

本実施例の場合、ランプ演出は、三系統のランプ群CH0~CH2によって実行されており、ランプ駆動基板37は、枠中継基板35、36を経由して、CH0のランプ駆動信号SDATA0を、クロック信号CK0に同期して受けている。なお、シリアル信号として伝送される一連のランプ駆動信号SDATA0は、動作制御信号ENABLE0がアクティブレベルに変化したタイミングで、ドライバICからランプ群CH0に出力されることで一斉に点灯状態が更新される。

#### 【0060】

以上の点は、ランプ駆動基板30についても同様であり、ランプ駆動基板30のドライバICは、ランプ群CH1のランプ駆動信号SDATA1を、クロック信号CK1に同期して受け、動作制御信号ENABLE1がアクティブレベルに変化したタイミングで、ランプ群CH1の点灯状態を一斉に更新している。

20

#### 【0061】

一方、モータランプ駆動基板31に搭載されたドライバICは、クロック同期式で伝送されるランプ駆動信号を受けてランプ群CH2を駆動すると共に、クロック同期式で伝送されるモータ駆動信号を受けて、複数のステッピングモータで構成された演出モータ群M1~Mnを駆動している。なお、ランプ駆動信号とモータ駆動信号は、一連のシリアル信号SDATA2であって、クロック信号CK1に同期してシリアル伝送され、これを受けたドライバICは、動作制御信号ENABLE2がアクティブレベルに変化するタイミングで、ランプ群CH2やモータ群M1~Mnの駆動状態を更新する。

30

#### 【0062】

続いて、音声回路SNDについて説明する。図4(a)に示す通り、演出インタフェース基板22には、演出制御基板23のCPU回路51(演出制御CPU63)から受ける指示に基づいて音声信号を再生する音声プロセッサ(音声合成回路)27と、再生される音声信号の元データである圧縮音声データなどを記憶する音声メモリ28と、音声プロセッサ27から出力される音声信号を受けるデジタルアンプ29と、が搭載されている。

#### 【0063】

音声プロセッサ27は、内部回路の異常動作時に、内部回路の設定値を自動的にデフォルト値(初期値)にリセットするWDT回路と、音声制御レジスタSRGとを内蔵して構成されている。そして、音声プロセッサ27は、演出制御CPU63から音声制御レジスタSRGに受ける動作パラメータ(音声コマンドによる設定値)に基づいて、音声メモリ28をアクセスして、必要な音声信号を再生して出力している。

40

#### 【0064】

図4(a)に示す通り、音声プロセッサ27と、音声メモリ28とは、26bit長の音声アドレスバスと、16bit長の音声データバスで接続されている。そのため、音声メモリ28には、1Gbit(=2<sup>26</sup>\*16)のデータが記憶可能となる。

#### 【0065】

音声制御レジスタSRGは、レジスタバンク1~レジスタバンク6に区分され、各々、00H~FFHのレジスタ番号で特定される。したがって、所定の設定動作は、レジスタバ

50

ンクを特定した上で、演出制御CPU63が、所定のレジスタ番号（1バイト長）の音声制御レジスタSRGに、1バイト長の動作パラメータを書込むことで実現される。

#### 【0066】

本実施例の場合、音声制御レジスタSRGのレジスタ番号（00H～FFH）は、演出制御CPU63のアドレス空間CS3に対応しており、例えば、レジスタ番号XXHの音声制御レジスタSRGに、動作パラメータYYHを設定する場合には、演出制御CPU63は、アドレス空間CS3のゼロ番地にXXHを書込み、次に、1番地にYYHを書込むことになる。すなわち、演出制御CPU63は、そのデータバスにXXHとYYHを、この順番に書き出すことになる。なお、本明細書において、添え字Hや、0X/0xの接頭記号は、数値が16進数表示であることを示している。

10

#### 【0067】

また、本明細書において、アドレス空間CS0～CS7とは、揮発性の有無を含むメモリ種別や、データバス幅（8/16/32ビット）を、各々、規定可能なCPU回路51にとって外部メモリを意味する（内蔵メモリを除く）。このアドレス空間CS0～CS7は、異なるチップセレクト信号CS0～CS7で選択され、READ/WRITE アクセス時に機能するREAD/WRITE 制御信号がメモリ種別に対応して最適化できるよう設定可能に構成されている。なお、この設定動作は、バスステートコントローラ66に対して実行される。

#### 【0068】

図4（e）は、演出制御CPU63による音声レジスタSRGへの設定動作を図示したものであり、2bit長のアドレスバスA1-A0と、1バイト長のデータバスD7-D0の内容が示されている。なお、本実施例では、チップセレクト信号CS3は、アドレス空間CS3をアクセスする場合に、自動的にアクティブになるよう、電源投入時に設定されるが、この点は図6や図12に関して後述する。

20

#### 【0069】

何れにしても、本実施例の場合、音声メモリ28に記憶された圧縮音声データは、13bit長のフレーズ番号NUM（000H～1FFFH）で特定されるフレーズ（phrase）圧縮データであり、一連の背景音楽の一曲分（BGM）や、ひと纏まりの演出音（予告音）などが、最高8192種類（=2<sup>13</sup>）、各々、フレーズ番号NUMに対応して記憶されている。そして、このフレーズ番号NUMは、演出制御CPU63から音声プロセッサ27の音声制御レジスタSRGに伝送される音声コマンドの設定値（動作パラメータ）によって特定される。

30

#### 【0070】

前記の通り、上記の構成を有する音声メモリ28は、リセット信号RT3で電源リセットされ、音声プロセッサ27は、リセット信号RT4で電源リセットされる。図4（c）に示す通り、リセット信号RT4は、電源投入後、所定のアサート期間ASRT（Lレベル区間）を経て、Hレベルに立ち上がるが、本実施例では、その後、音声プロセッサ27の内部回路が自動的に機能して、初期化シーケンス処理が実行されるよう構成されている。なお、この初期化シーケンス処理は、所定の手順で実行される内部動作であり、初期化シーケンス処理の動作中は、演出制御CPU63が音声レジスタSRGをアクセスすることはできない。

40

#### 【0071】

そして、内部動作たる初期化シーケンス処理が完了すると、CPU回路51に対する割込み信号IRQ\_SNDがLレベルに変化し、CPU回路51（演出制御CPU63）は、割込み信号IRQ\_SNDに基づき割込み処理プログラムを実行する。そして、所定の命令に基づいて割込み信号IRQ\_SNDがHレベルに戻されるが、その詳細については、図14（c）を参照して更に後述する。

#### 【0072】

図4（a）に示す通り、演出制御部23のCPU回路51のデータバスとアドレスバスは、液晶インタフェース基板24に搭載された時計回路（real time clock）38と演出データメモリ39にも及んでいる。時計回路38は、CPU回路51のアドレスバスの下位

50

4 b i t と、データバスの下位 4 b i t に接続されており、チップセレクト信号 C S 4 で時計回路 3 8 が選択された状態では、C P U 回路 5 1 が、(4 b i t 長アドレス値を有する) 内部レジスタを任意にアクセスできるよう構成されている。

【0073】

また、演出データメモリ 3 9 は、高速アクセス可能なメモリ素子 S R A M (Static Random Access Memory) であって、C P U 回路 5 1 のアドレスバスの 1 6 b i t と、データバスの下位 1 6 b i t に接続されており、チップセレクト信号 C S 4 でチップ選択された状態では、S R A M (演出データメモリ) 3 9 に記憶されている遊技実績情報その他が、C P U 回路 5 1 から適宜に R / W アクセスされるようになっている。なお、チップセレクト信号 C S 4 で選択されるアドレス空間 C S 4 において、0 番地から 1 5 番地までは時計回路 3 8 に付番されているので、S R A M 3 9 では使用しない。

10

【0074】

時計回路 3 8 と演出データメモリ 3 9 は、不図示の二次電池で駆動されており、この二次電池は、遊技動作中、電源基板 2 0 からの給電電圧によって適宜に充電される。そのため、電源遮断後も、時計回路 3 8 の計時動作が継続され、また、演出データメモリ 3 9 に記憶された遊技実績情報が、永続的に記憶保持されることになる(不揮発性を付与)。なお、時計回路(RTC) 3 8 は、C P U 回路 5 1 に対して、割込み信号 I R Q \_ R T C を出力可能に構成されている(RTC 割込み)。この R T C 割込みには、日、曜日、時、分、秒が特定可能なアラーム割込みと、所定時間経過後に起動されるタイマ割込みが存在するが、本実施例では、毎日の営業終了時に、日々の遊技実績情報を更新するアラーム割込みを活用している。

20

【0075】

図 4 (a) の右側に示す通り、演出制御基板 2 3 には、C P U 回路 5 1 や V D P 回路 5 2 を内蔵する複合チップ 5 0 と、C P U 回路 5 1 の制御プログラムを記憶する制御メモリ(PROM) 5 3 と、大量のデータを高速にアクセス可能な D R A M (Dynamic Random Access Memory) 5 4 と、演出制御に必要な大量の C G データを記憶する C G R O M 5 5 と、が搭載されている。

【0076】

図 7 に関して後述するように、制御メモリ(PROM) 5 3 は、本実施例では、チップセレクト信号 C S 0 で選択されるアドレス空間 C S 0 に位置付けされている。また、D D R (Double-Data-Rate 3) で構成される D R A M (Dynamic Random Access Memory) 5 4 は、チップセレクト信号 C S 5 で選択されるアドレス空間 C S 5 に位置付けされている。

30

【0077】

図 5 (a) は、演出制御部 2 3 を構成する複合チップ 5 0 について、関連する回路素子も含めて図示した回路ブロック図である。図示の通り、実施例の複合チップ 5 0 には、所定時間毎にディスプレイリスト D L を発行する C P U 回路 5 1 と、発行されたディスプレイリスト D L に基づいて画像データを生成して表示装置 D S 1 , D S 2 を駆動する V D P 回路 5 2 とが内蔵されている。そして、C P U 回路 5 1 と V D P 回路 5 2 とは、互いの送受信データを中継する C P U I F 回路 5 6 を通して接続されている。

40

【0078】

なお、V D P 回路 5 2 には、音声プロセッサ 2 7 と同等の機能を発揮する音声回路 S N D が内蔵されているが、これから説明する最初の実施例では、音声回路 S N D を活用していない。但し、最後に説明する実施例のように、V D P 回路 5 2 に内蔵された音声回路 S N D を活用すれば、音声メモリ 2 8 や音声プロセッサ 2 7 の配置が不要となる。

【0079】

まず、C P U 回路 5 1 は、発振器 O S C 1 の発振出力(例えば 1 0 0 / 3 M H z) を H C L K I 端子に受けて、これを周波数逓倍(例えば 8 逓倍)して、2 6 6 . 7 M H z 程度の C P U 動作クロックとしている。ここで、発振器 O S C 1 は、スペクトラムス拡散波を出力するように構成されることで、電波障害／電磁妨害を防止する E M I (Electromagnetic Int

50

erference) 対策を図っている。

#### 【0080】

一方、VDP回路52は、発振器OSC2の発振出力（例えば40MHz）をPLLREF端子に受け、PLL（Phase Locked Loop）回路で、適宜に周波数逡倍した上で、VDP回路52のシステムクロック、表示装置用の表示クロック（ドットクロックなど）、及び、外付けDRAM54のDDRクロックとして使用している。すなわち、発振器OSC2の出力は、VDP回路52全体のリファレンスクロックとして機能している。なお、PLL回路の周波数逡倍比は、所定の設定端子への設定値で規定される。

#### 【0081】

そこで、このリファレンスクロックの重要性を考慮して、本実施例では、発振器OSC2をVDP回路52と同じ電源電圧3.3Vで動作させると共に、出力イネーブル端子OEがHレベル（=3.3V）であることを条件に、リファレンスクロックを発振出力するよう構成されている。そして、万一、電源電圧3.3Vが所定レベル以下に低下した場合には、その後、正常な演出動作は望めないもので、マスク不能の割込み（NMI）が生じるよう構成されている。

#### 【0082】

また、複合チップ50には、HBTS端子が設けられ、HBTS端子の論理レベルに基づいて、電源投入（CPUリセット）後に実行されるブートプログラム（初期設定プログラム）が、CGROM55に記憶されているか（HBTS=H）、それ以外のメモリに記憶されているか（HBTS=L）を特定している。図示の通り、この実施例では、HBTS=Lレベルに設定されており、演出制御CPU63のアドレス空間CS0のゼロ番地が、CGROM以外に割り当てられ、具体的には、アドレス空間CS0は、制御メモリ53に割り当てられている。

#### 【0083】

一方、HBTS端子=Hレベルに設定されている場合（破線参照）は、演出制御CPU63のアドレス空間CS0のゼロ番地が、CGROM55に割り当てられる。この場合は、CGROM55のメモリ種別と、バス幅（64/32/16bit）とが、2bit長のHBTBWD端子と、4bit長のHBTRMSL端子への入力値に基づいて各々特定されるようになっている。なお、これらの点は、図34に基づいて更に後述する。

#### 【0084】

続いて、CPU回路51とVDP回路52について、互いの送受信データを中継するCPUIF回路56について説明する。図5（a）に示す通り、CPUIF回路56には、制御プログラムや必要な制御データを不揮発的に記憶する制御メモリ（PROM）53と、2Mバイト程度の記憶容量を有するワークメモリ（RAM）57とが接続され、各々、CPU回路51からアクセス可能に構成されている。先に説明した通り、制御メモリ（PROM）53は、チップセレクト信号CS0で選択されるアドレス空間CS0に位置付けられ、ワークメモリ（RAM）57は、チップセレクト信号CS6で選択されるアドレス空間CS6に位置付けられている。

#### 【0085】

このワークメモリ（RAM）57には、表示装置DS1，DS2の各一フレームを特定する一連の指示コマンドが記載されたディスプレイリストDLを、一次的に記憶するDLバッファBUFが確保されている。本実施例の場合、一連の指示コマンドには、CGROM55から画像素材（テクスチャ）を読み出してデコード（展開）するためのTXLOADコマンドなどのテクスチャロード系コマンドと、デコード（展開）先のVRAM領域（インデックス空間）を予め特定するなどの機能を有するSETINDEXコマンドなどのテクスチャ設定系コマンドと、デコード（展開）後の画像素材を仮想描画空間の所定位置に配置するためのSPRITEコマンドなどのプリミティブ描画系コマンドと、描画系コマンドによって仮想描画空間に描画された画像のうち、実際に表示装置に描画する描画領域を特定するためのSETDAVR コマンドやSETDAVF コマンドなどの環境設定コマンドと、インデックス空間を管理するインデックステーブルIDXTBLに関するインデックステーブル制御系コマン

10

20

30

40

50

ド (WRIDXTBL) が含まれる。

【0086】

なお、図9(c)には、仮想描画空間(水平X方向±8192:垂直Y方向±8192)と、仮想描画空間の中で任意に設定可能な描画領域と、表示装置DS1, DS2に出力する画像データを一次保存するフレームバッファFBa, FBbにおける実描画領域と、の関係が図示されている。

【0087】

次に、CPU回路51は、汎用のワンチップマイコンと同等の性能を有する回路であり、制御メモリ53の制御プログラムに基づき画像演出を統括的に制御する演出制御CPU63と、プログラムが暴走状態になるとCPUを強制リセットするウォッチドッグタイマ(WDT)と、16kバイト程度の記憶容量を有してCPUの作業領域として使用される内蔵RAM59と、CPU63を経由しないでデータ転送を実現するDMAC(Direct Memory Access Controller)60と、複数の入力ポートSi及び出力ポートSoを有するシリアル入出力ポート(SIO)61と、複数の入力ポートPi及び出力ポートPoを有するパラレル入出力ポート(PIO)62と、前記各部の動作を制御するべく設定値が設定される動作制御レジスタREGなどを有して構成されている。但し、外付けのWDT回路58を設けた本実施例では、CPU回路51に内蔵されたウォッチドッグタイマ(WDT)を活用していない。

【0088】

なお、本明細書では、便宜上、入出力ポートとの表現を使用するが、演出制御部23において、入出力ポートには、独立して動作する入力ポートと出力ポートとが含まれている。この点は、以下に説明する入出力回路64pや入出力回路64sについても同様である。

【0089】

パラレル入出力ポート62は、入出力回路64pを通して外部機器(演出インタフェース基板22)に接続されており、演出制御CPU63は、入力回路64pを経由して、音量スイッチVLSWのエンコーダ出力3bitと、チャンスボタン11のスイッチ信号と、制御コマンドCMDと、割込み信号STBと、を受信するようになっている。エンコーダ出力3bitと、スイッチ信号1bitは、入出力回路64pを経由して、パラレル入出力ポート(PIO)62に供給されている。

【0090】

同様に、受信した制御コマンドCMDは、入出力回路64pを経由して、パラレル入出力ポート(PIO)62に供給されている。また、ストローブ信号STBは、入出力回路64pを経由して、演出制御CPU63の割込み端子に供給されることで、受信割込み処理を起動させている。したがって、受信割込み処理に基づいて、制御コマンドCMDを把握した演出制御CPU63は、演出抽選などを経て、この制御コマンドCMDに対応する音声演出、ランプ演出、モータ演出、及び画像演出を統一的に制御することになる。

【0091】

特に限定されないが、本実施例では、ランプ演出とモータ演出のために、VDP回路52のSMC部(Serial Management Controller)78を使用している。SMC部78は、LEDコントローラとMotorコントローラと、を内蔵した複合コントローラであり、クロック同期方式でシリアル信号を出力できるよう構成されている。また、Motorコントローラは、所定の制御レジスタ70への設定値に基づき、任意のタイミングでラッチパルスを出力可能に構成され、また、クロック同期方式でシリアル信号を入力可能に構成されている。

【0092】

そこで、本実施例では、クロック信号に同期してモータ駆動信号やLED駆動信号を、SMC部78から出力させる一方、適宜なタイミングで、ラッチパルスを、動作制御信号ENABLEとして出力するようにしている。また、演出モータ群M1~Mnからの原点センサ信号SN0~SNnをクロック同期方式でシリアル入力するよう構成されている。

【0093】

図4 (a) に関して説明した通り、クロック信号CK0～CK2、駆動信号SDATA0～SDATA2、及び、動作制御信号ENABLE0～ENABLE2は、出力バッファ41～43を経由して、所定の駆動基板30, 31, 37に伝送される。また、原点センサ信号SN0～SNnは、モータランプ駆動基板31から入出力バッファ43を経由して、SMC部78にシリアル入力される。

【0094】

但し、本実施例において、SMC部78を使用することは必須ではない。すなわち、CPU回路51には、汎用のシリアル入出力ポートSIO61が内蔵されているので、これらを使用して、ランプ演出とモータ演出を実行することもできる。

【0095】

具体的には、図5 (a) の破線に示す通りであり、破線で示す構成では、シリアル入出力ポートSIO61に内部接続されている入出力回路64sを経由して、クロック信号CK0～CK2、駆動信号SDATA0～SDATA2が出力され、入出力回路64pを経由して動作制御信号ENABLE0～ENABLE2が出力される。なお、便宜上、入出力ポートや入出力回路と表現するが、実際に機能するのは、出力ポートや出力回路である。

【0096】

ここで、シリアル出力ポートSOは、16段のFIFOレジスタを内蔵して構成されている。そして、DMAC回路60は、演出制御CPU63から動作開始指示(図17 (b) ST18参照)を受けて起動し、ランプ／モータ駆動テーブル(図17 (b) 参照)から、必要な駆動データを順番に読み出し、シリアル出力ポートSOのFIFOレジスタにDMA転送するよう構成されている。FIFOレジスタに蓄積された駆動データは、クロック同期方式でシリアル出力ポートSOからシリアル出力される。なお、DMAC回路には、複数(例えば7)のDMAチャンネルが存在するが、優先度に劣る第3のDMAチャンネルでランプ駆動データをDMA転送し、最優先度の第1のDMAチャンネルでモータ駆動データをDMA転送するよう構成されている。

【0097】

CPU回路51に内蔵された動作制御レジスタREGは、レジス番号(アドレス値)が0xFF40000以降に付番された8bit、16bit、又は32bit長のレジスタであり、演出制御CPU63から適宜にWRITE/READアクセス可能に構成されている(図7参照)。そのためノイズなどの影響で、動作制御レジスタREGに、不合理な値に設定される可能性がある。

【0098】

但し、例えば、意図的に無限ループ処理を実行させて外付けのWDT回路58を起動させることで、複合チップ50を異常リセットすることができる。この場合、動作制御レジスタREGの値が、電源投入後と同じデフォルト値(初期値)に戻され、且つ、VDP回路52についても、VDPレジスタRGijの値が、デフォルト値(初期値)に戻されることで異常状態が解消される。

【0099】

図4 (b) は、このリセット動作に関連する回路構成であって、本実施例に特徴的なリセット機構を説明する図面である。なお、本明細書において、RGijと表記するVDPレジスタは、CPU回路51に内蔵された動作制御レジスタREGではなく、VDP回路52の内部動作を制御する制御レジスタ群70(図7参照)の何れかを意味する。また、図4 (b) に示すシステム制御回路520とは、VDPレジスタRGij(図7の制御レジスタ群70の何れか)への設定値に基づいて機能するVDP回路52の内部制御回路を意味する(図4 (a) 参照)。なお、VDPレジスタRGijは、演出制御CPU63のアドレスマップにおいて、チップセレクト信号CS7で選択されるアドレス空間CS7に位置付けられる。

【0100】

以上を踏まえてリセット機構について説明すると、図4 (b) に示す通り、複合チップ50は、論理反転されたシステムリセット信号SYSバーを受ける3個のORゲートG2～

10

20

30

40

50

G 4を経由して、内部回路がリセット可能に構成されている。但し、本実施例では、破線で示すように、内蔵WDTを有効化しないので、ORゲートG 2の入端子と出力端子は、直結状態となる。

【0101】

何れにしても、CPU回路51とVDP回路52の間に、パターンチェック回路CHKが設けられ、パターンチェック回路CHKは、パラレル入出力ポート（PIO）62から、所定のキーワード列（リセット用の暗号列）を受けることを条件に、リセット信号RSTを出力するよう構成されている。

【0102】

そして、複合チップ50の内部回路は、（1）CPU回路51と、（2）VDP回路52の表示回路74と、（3）VDP回路52における表示回路以外とに三分されて、各々、ORゲートG 2～G 4から第1リセット経路～第3リセット経路のリセット信号を受けるよう構成されている。

【0103】

まず、入出力端子が直結状態のORゲートG 2は、第1リセット経路に関連しており、システムリセット信号SYSバーに基づき、CPU回路51の全体をシステムリセットされるように構成されている。また、ORゲートG 3は、第2リセット経路に関連しており、システムリセット信号SYSバーと、パターンチェック回路CHKからのリセット信号RSTとを受けて、OR論理に基づき、VDP回路52全体をリセット可能に構成されている。

【0104】

この第2リセット経路は、電源投入時の電源リセット動作だけでなく、所定の異常を検出した演出制御CPU63が、VDP回路52の全体を異常リセットして初期状態に戻す用途で使用される。具体的には、VDP回路52の内部動作を示す所定のステータスレジスタRGijに基づき、重大な異常が発生していると判断される場合には、パターンチェック回路CHKからリセット信号RSTを発生させることで、VDP回路52の全体を異常リセットしている。なお、表示回路74は、ORゲートG 4を経由して、第2リセット経路→第3リセット経路で異常リセットされる。

【0105】

一方、VDP回路52に内蔵された内部回路は、第4リセット経路で、必要時に個々のリセットすることも可能に構成されている。個々のリセット可能な内部回路には、図5（a）に示すインデックステーブルIDXTBL、データ転送回路72、プリローダ73、表示回路74、描画回路76、SMC回路78、及び、音声回路SNDや、図10に示すICM回路が含まれている。

【0106】

個別的なリセット動作を実現する手法は、図4（b）の下部に記載の通りであり、例えば、表示回路74は、所定のVDPレジスタRGij（システムコマンドレジスタ）に、第1リセット値を書き込むことで、第4リセット経路4A→第3リセット経路を経てリセットされる。

【0107】

また、VDP回路52の各内部回路（72，73，74，76，SND，・・・）は、（1）第1のVDPレジスタRGij（リセットRQレジスタ）に、対象回路を特定する設定値を書き込んだ後、（2）所定のVDPレジスタRGij（システムコマンドレジスタ）に、第2リセット値を書き込むことで、個々のリセットされる（第4リセット経路4B）。なお、この実施例では使用しないが、音声回路SNDは、第4リセット経路4Bによるリセットだけでなく、所定のVDPレジスタ（回路設定コマンドレジスタ）に、リセット値を書き込むことでリセット可能である（第4リセット経路4C）。

【0108】

本実施例は、上記の構成を有するので、電源投入時やプログラム暴走時に、VDP回路52全体が自動的に初期状態に戻るだけでなく、必要に応じて、各部を初期状態に戻して異

10

20

30

40

50

常事態の回復を図ることができる。例えば、一定期間、内蔵 V R A M 7 1 に対して R E A D / W R I T E アクセスがない描画回路 7 6 のフリーズ時には、第 4 リセット経路 4 B を経由して描画回路 7 6 が個別的に初期化される（図 1 7 ( d ) の S T 1 6 a 参照）。プリローダ 7 3 やデータ転送回路 7 2 についても、ほぼ同様であり、所定の異常時には、第 4 リセット経路 4 B を経由してプリローダ 7 3 が初期化され（図 2 4 の S T 2 7 参照）、第 4 リセット経路 4 B を経由してデータ転送回路 7 2 が初期化される（図 1 9 や図 2 4 の S T 2 7 参照）。

#### 【 0 1 0 9 】

また、表示回路 7 4 については、1 / 6 0 秒毎の表示タイミングに、表示データの生成が間に合わないアンダーラン（Underrun）異常が続くような場合に、第 4 リセット経路 4 A 又は第 4 リセット経路 4 B を経由して、表示回路 7 4 が個別的に初期化される（図 1 7 の S T 1 0 c 参照）。なお、これら個別的なリセット動作については、図 1 7 以降に記載したプログラム処理に関して更に後述する。

10

#### 【 0 1 1 0 】

以上、本実施例に特徴的なリセット機構について説明したが、何れかのリセット経路 1 ~ 4 が機能して、複合チップ 5 0 の内部回路がリセットされると、その内部回路に対応する V D P レジスタ R G i j の設定値は、電源投入後と同じデフォルト値に戻る。

#### 【 0 1 1 1 】

続いて、C P U 回路 5 1 の内部構成に戻って、特徴的な回路構成の説明を続ける。図 6 は、C P U 回路 5 1 の内部構成をやや詳細に示すブロック図である。C P U 回路 5 1 は、先に説明した内蔵 R A M 5 9、D M A C 回路 6 0、S I O 6 1、P I O 6 2、W D T 以外にも、多くの特徴的な回路を含んで構成されている。

20

#### 【 0 1 1 2 】

まず第 1 に、C P U 回路 5 1 は、命令用の C P U フェッチバスと、データ用の C P U メモリアクセスバスとを別々に有してハーバード・アーキテクチャを実現している。そのため、C P U コア（演出制御 C P U）6 3 が命令をメモリから読むフェッチ動作と、メモリアクセス動作とが競合せず、フェッチ動作を連続させることで高速処理を実現している。

#### 【 0 1 1 3 】

また、C P U コア 6 3 は、複数個（例えば 1 5 個）のレジスタバンク R B 0 ~ R B 1 4 を有して構成されており、その使用の有無を選択できるよう構成されている。そして、レジスタバンク R B i の使用を許可した動作状態では、割込み処理の開始時に、C P U の内蔵レジスタ（例えば 1 9 個）のレジスタ値（各 3 2 b i t 長）が、空き状態のレジスタバンク R B i に自動的に退避される。

30

#### 【 0 1 1 4 】

また、割込み処理の終了時に所定の復帰命令を実行すると、例えば 1 9 個の退避データが、対応する内蔵レジスタに自動的に復帰される。したがって、通常の構成のように、割込み処理の開始時に P U S H 命令を 1 9 回実行し、割込み処理の終了時に P O P 命令を 1 9 回実行する手間が不要となり、高速処理が実現される。

#### 【 0 1 1 5 】

また、実施例の C P U 回路 5 1 は、命令キャッシュメモリ 6 7 と、オペランドキャッシュメモリ 8 9 と、キャッシュコントローラ 6 9 とを設けることで、ハーバードキャッシュ動作を実現しており、同一アドレスをアクセスする場合に、キャッシュ済みのデータを活用することでプログラム処理の更なる高速化を図っている。なお、バスブリッジ 6 5 と、周辺バス(1)用のコントローラ、周辺バス(2)用のコントローラ、及び、周辺バス(3)用のコントローラとが設けられることで、内部バスと、周辺バス(1)、周辺バス(2)、及び周辺バス(3)とを適宜に接続している。

40

#### 【 0 1 1 6 】

次に、図 6 の回路構成において、バスステートコントローラ 6 6 は、動作制御レジスタ R E G への適宜な設定値に基づいて動作して、C P U 回路 5 1 に接続された各種メモリデバイスとのメモリ R E A D 動作やメモリ W R I T E 動作を最適化する部分である。メモリ R E A D 動

50

作やメモリWRITE動作は、例えば、図35に例示した動作タイミングで実行されるが、アドレスバス(28Bit)から出力されるアドレスデータと、READデータバス(32Bit)に読み出されるREADデータと、WRITEデータバス(32Bit)に書き出されるWRITEデータと、チップセレクト信号CS0～CS7などの制御信号との動作タイミングが、動作制御レジスタREGへの設定値に基づいて、各メモリデバイスの特性に対応して適宜に規定される。

#### 【0117】

READデータバスとWRITEデータバスが別々に設けられているので、上記したハード・アーキテクチャによる高速動作が実現される。なお、本明細書では、アドレスバス(28Bit)、READデータバス(32Bit)、及び、WRITEデータバス(32Bit)について、図6に示す内部バスや、周辺バス(1)～周辺バス(3)などと区別する意味で、外部バスと総称することがある。

#### 【0118】

図7は、チップセレクト信号CS0～CS7によって選択されるアドレス空間CS0～CS7を図示したものであり、バスステートコントローラ66を経由してアクセスされる演出制御CPU63にとってのアドレスマップを図示したものである。先ず、各アドレス空間CS0～CS7は、何れも、最大64Mバイトに(=0x4000000H=67108864)に規定されている。

#### 【0119】

先に説明した通り、アドレス空間CS0～CS7とは、揮発性の有無を含むメモリ種別や、データバス幅(8/16/32ビット)を、各々、規定可能なCPU回路51にとって外部メモリを意味する。そして、本実施例では、図6(b)や図7に示す通り、制御メモリ(PROM)53がアドレス空間CS0、音声プロセッサ27の音声制御レジスタSRGがアドレス空間CS3、時計回路38の内部レジスタやSRAM39がアドレス空間CS4、外付けDRAM(DDR)54がアドレス空間CS5、ワークメモリ57がアドレス空間CS6、VDPLレジスタRGijがアドレス空間CS7に位置付けられている。なお、アドレス空間CS1、CS2についての説明は省略する。

#### 【0120】

ところで、図7から確認される通り、アドレス空間CS0～CS7は、アドレス値0x00000000～0x1FFFFFFF(キャッシュ有効空間)だけでなく、アドレス値0x20000000～0x3FFFFFFF(キャッシュ無効空間)にも確保されている。これは、アドレスビットA29=1のときには、CPU回路51の内部動作に基づいて、キャッシュ無効とする一方、アドレスビットA29=0のときにキャッシュ有効とすることで、キャッシュ機能の活用を任意選択できるようにしたものである。

#### 【0121】

そのため、本実施例では、全32bitのアドレス情報(ビットA31～A0)のうち、ビットA29の値が1又は0の何れであっても、残り31bit(ビットA31～A30とビットA28～A0)の値が同じであれば、同一のメモリの同一番地を指示することになる。例えば、0x18000000番地をREADアクセスしても、0x38000000番地をREADアクセスしても、ワークメモリ57のゼロ番地から同一データが読み出されることになる。なお、0x18000000番地をREADアクセスした場合には、読み出したデータがキャッシュに保存されるが、図6(b)は、キャッシュ有効/無効のアクセス動作を図示している。

#### 【0122】

もっとも、所定の動作制御レジスタREGへの設定値に基づいて、命令キャッシュ及び/又はオペランドキャッシュについて、キャッシュ動作を無効化することもできる。但し、本実施例では、電源投入後、命令キャッシュ及びオペランドキャッシュについて、キャッシュ動作を有効化した上で、必要に応じて、キャッシュ無効空間をアクセスすることで、キャッシュ動作を無効化している。

#### 【0123】

図7のメモリマップについて説明を続けると、0x40000000番地以降は、バスステートコ

10

20

30

40

50

ントローラ 66 が機能しない内部メモリ空間であって、0xF0000000番地～0xFF3FFFFF番地は、キャッシュのアドレスレイ空間に割り当てられている。また、0xFF400000番地～0xFFFF7FFFF番地と、0xFFFC0000番地～0xFFFFFFFF番地は、内蔵周辺モジュールに割り当てられ、具体的には、CPU回路の動作制御レジスタ REG に割り当てられている。なお、内蔵 RAM 59 のアドレス範囲は、0xFFFF80000～0xFFFFBFFFFである。

#### 【0124】

CPU回路51の内部構成について説明を続けると、コンペアマッチタイマCMTと、マルチファンクションタイマユニットMTUは、CPU回路51に供給される外部信号をカウントしたり、或いは、内部クロックを逡倍又は分周した計数クロックをカウントして、カウント結果が所定値に達すると、割込み信号などを発生する回路である。特に限定されないが、本実施例では、マルチファンクションタイマユニットMTUを活用して、1mS割込み信号と、20μS割込み信号を発生させている。

#### 【0125】

次に、割込みコントローラINTCは、VDP回路52やDMAC回路60やマルチファンクションタイマユニットMTUなどからの内部割込みと、IRQ\_CMD、IRQ\_SND、IRQ\_RCTなどの外部割込みを受けて、予め規定されている優先順位に基づいて、割込み処理（割込みハンドラ）を起動させる回路である。ここで、IRQ\_CMDは、制御コマンドCMDを受信すべきコマンド受信割込み信号、IRQ\_SNDは、音声プロセッサ27が初期化シーケンスを終えたことを示す終了割込み信号、IRQ\_RCTは、アラーム割込み信号である。

#### 【0126】

そして、本実施例では、割込み優先度は、コマンド受信割込みIRQ\_CMDが最高レベルであり、以下、20μS割込み→1mS割込み→VDP回路からの割込み（IRQ0，IRQ1，IRQ2，IRQ3）→DMAC割込み→IRQ\_SND→IRQ\_RCTの順になっている（図14（d）参照）。なお、これらは何れも、マスク可能な割込みであり、マスク不能な割込みNMIは、先に説明した通り、発振器OSC2からリファレンスクロックが出力されていない場合に演出制御CPU63に出力される。

#### 【0127】

そして、何れの割込み処理でも、CPUの複数の内蔵レジスタのレジスタ値（各32bit長）は、空き状態の何れかのレジスタバンクRBiに、自動的に退避される。そして、割込み処理の最後に所定の復帰命令を実行すると、退避データが、対応する内蔵レジスタに自動的に復帰される。

#### 【0128】

続いて、DMAC回路60について説明する。実施例のDMAC回路60は、所定の動作制御レジスタREGへの設定値に基づいて、転送元（Source）から転送先（Destination）に対して、所定のDMA転送モードで、所定のデータ転送単位毎に、所定回数、データ転送を繰り返す回路である。なお、同一の内部構成を有する複数チャネルのDMAC0～DMACnが用意されており、並列的に動作可能となっている。但し、優先度が決まっており（チャネル0>・・・>チャネルn）、チャネル調停動作モードの並列動作時には、所定タイミングでのチャネル調停によって優先度の高いDMACiの動作が優先される。

#### 【0129】

DMAC回路60の活用としては、例えば、シリアル出力ポートSOが機能する実施例（図7（a）破線部参照）では、CPU回路51の動作制御レジスタREGには、ランプ／モータ駆動テーブルの先頭アドレス（転送元アドレスの先頭値）と、シリアル出力ポートSOの入力レジスタのアドレス（転送先アドレスの固定値）と、データ転送単位（8bit）と、転送回数と、が指定される。そして、所定の動作制御レジスタREGに動作開始指示を受けたDMAC回路60は、転送元アドレスを更新しつつ、所定の転送先アドレスに駆動データをDMA転送する。そして、全てのDMA転送が終われば、DMAC割込み（動作終了割込み）が生じるよう構成されている。

#### 【0130】

この点は、ディスプレイリストDLをDMAC回路60が発行する実施例（図20、図24（c））の場合もほぼ同様である。すなわち、演出制御CPU63は、CPU回路51の所定の動作制御レジスタREGに、転送元（DLバッファBUF）の先頭アドレスと、転送先（転送ポートTR\_PORT）のアドレスと、DMA転送モードと、データ転送単位と、転送回数、その他の条件を設定することになる。なお、これらの点は、図20に関して更に後述する。

#### 【0131】

ところで、一般に、DMA転送モードには、DMA転送の単位動作（R動作／W動作）の途中でバス制御権を開放するなど、DMA動作がメモリバスを占有しないサイクルスチール転送モードと、複数のR動作やW動作を連続させるなど、指定された転送回数が完了するまでバス制御権を解放しないバースト転送（パイプライン転送）モードと、他のデバイスから受けるDMA転送要求（デマンド）がアクティブの間はDMA動作を継続するデマンド転送モードなどが考えられる。しかし、本実施例のDMAC回路60は、DMA転送時のリードアクセス起動（R動作）とライトアクセス起動（W動作）の間に、少なくとも1サイクルのメモリ開放期間を設けたサイクルスチール転送モードで機能することで、演出制御CPU63の動作に支障が出ないようにしている。

#### 【0132】

図8は、サイクルスチール転送動作（a1）と、パイプライン転送（a2）とを説明する図面である。図8（a1）に示す通り、サイクルスチール転送モードで機能するDMAC回路60は、1データ転送のリードアクセス起動（R）とライトアクセス起動（W）の間に、少なくとも1サイクル空けて動作しており、この空いたサイクルでは、演出制御CPU63のバス使用が可能となる。図8（a1）と図8（a2）の対比関係から明らかなように、パイプライン転送では、一サイクル（一オペランド転送）が終わるまでは、バスがCPUに開放されないのに対して、サイクルスチール転送モードでは、リードアクセス毎に、バスがCPUに開放されるので、CPUの動作が大きく遅れることがない。

#### 【0133】

そして、例えば、ディスプレイリストDLのVDP回路52への発行時に、DMAC回路60を使用する実施態様では、一サイクルのデータ転送単位（1オペランド）を、 $32 \times 2 \text{ bit}$ に設定し、ディスプレイリストDLが格納されている内蔵RAM59のソースアドレスを適宜に増加しつつ（1オペランド転送毎に+8）、固定アドレスで特定されるデータ転送回路72の転送ポートレジスタTR\_PORT（図10参照）に対して、DMA転送動作を実行している。

#### 【0134】

後述するように、実施例では、ディスプレイリストDLに、必要個数のNOP（no operation）コマンドを付加することで、全体のデータサイズを、固定値（例えば、 $4 \times 64 = 256$ バイト、又はその整数倍）に調整しており、 $32 \text{ bit} \times 2$ 回のオペランド転送を32回（又はその整数倍）繰り返すことで、ディスプレイリストDLの発行を完了させている。なお、描画回路76がNOPコマンドを実行しても、事実上、何の変化も生じない。

#### 【0135】

また、DMA転送条件に関して動作モードを分類すると、一般に、単一オペランド転送（図8（b1）参照）と、連続オペランド転送（図8（b2）参照）と、ノンストップ転送（図8（b3）参照）とが考えられる。

#### 【0136】

ここで、単一オペランド転送とは、図8（b1）に示すように、DMA転送要求が与えられるたびに、1オペランドの転送を繰り返し、転送バイト数をカウントするバイトカウンタがゼロになった時点で、DMA割込み要求が生じる動作モードを意味する。次に、連続オペランド転送とは、図8（b2）に示すように、1回のDMA要求で、バイトカウンタがゼロになるまでDMA転送を繰り返す動作モードを意味する。

#### 【0137】

これら、連続オペランド転送（b 2）や単一オペランド転送（b 1）では、1 オペランド転送が終了するごとにチャンネル調停が行われ、優先順位の高いチャンネルのDMA 要求がないことを条件に、現在のチャンネルの転送が継続される（チャンネル調停動作モード）。そこで、本実施例では、ディスプレイリストDLのVDP 回路への発行や、ランプ駆動データやモータ駆動データのDMA 転送は、単一オペランド転送方式を採っている。そして、並列動作時には、例えば、モータデータ>ディスプレイリストDL>ランプデータの優先度のチャンネル調停となるよう、最適チャンネルのDMA Ci を使用している。

【0138】

一方、ノンストップ転送とは、チャンネル調停が実行されない動作モードであって、図8（b 3）に記載の通り、1 回のDMA 要求で、バイトカウントがゼロになるまで連続的にDMA 転送が繰り返される。本実施例では、電源投入時のメモリセクション初期化処理（図12のSP8）では、ノンストップ転送でプログラムやデータをDMA 転送している。

【0139】

以上、CPU 回路51 について説明したので、次に、VDP 回路52 について説明すると、VDP 回路52 には、画像演出を構成する静止画や動画の構成要素となる圧縮データを記憶するCGROM 55 と、4 Gbit 程度の記憶容量を有する外付けDRAM（Dynamic Random Access Memory）54 と、メイン表示装置DS1 と、サブ表示装置DS2 とが接続されている。なお、DRAM 54 は、好適にはDDR3（Double-Data-Rate3 SDRAM）で構成される。

【0140】

特に限定するものではないが、この実施例では、CGROM 55 は、62 Gbit 程度の記憶容量のNAND 型フラッシュメモリで構成されたフラッシュSSD（solid state drive）で構成されており、シリアル伝送によって必要な圧縮データを取得するよう構成されている。そのため、パラレル伝送において不可避免的に生じるスキュー（ビットデータ毎の伝送速度の差）の問題が解消され、極限的な高速伝送動作が可能となる。特に限定されないが、本実施例では、Serial ATA に準拠したHSS（High Speed Serial）方式で、CGROM 55 を高速アクセスしている。

【0141】

なお、Serial ATA に準拠したHSS 方式を採るか否かに拘らず、NAND 型のフラッシュメモリは、ハードディスクより機械的に安定であり、且つ高速アクセスが可能である一方で、シーケンシャルアクセスメモリであるため、DRAM やSRAM（Static Random Access Memory）に比較すると、ランダムアクセス性に問題がある。そこで、本実施例では、一群の圧縮データ（CGデータ）を、描画動作に先行してDRAM 54 に読み出ししておくプリロード動作を実行することで、描画動作時におけるCGデータの円滑なランダムアクセスを実現している。ちなみに、アクセス速度は、内蔵VRAM>外付けDRAM>CGROMの順番に遅くなる。

【0142】

VDP 回路52 は、詳細には、VDP（Video Display Processor）の動作を規定する各種の動作パラメータが演出制御CPU 63 によって設定可能な制御レジスタ群70 と、表示装置DS1，DS2 に表示すべき画像データの生成時に使用される48 Mバイト程度の内蔵VRAM（video RAM）71 と、チップ内部の各部間のデータ送受信及びチップ外部とのデータ送受信を実行するデータ転送回路72 と、内蔵VRAM 71 に関して、Source やDestination のアドレス情報を特定可能なインデックステーブルIDXTBL と、描画動作に先行してCGROM 55 をREAD アクセスするプリロード動作が実行可能なプリローダ73 と、CGROM 55 から読み出した圧縮データをデコード（復号伸長／展開）するグラフィックスデコーダ（GDEC）75 と、デコード（展開）後の静止画データや動画データを適宜に組み合わせて表示装置DS1，DS2 の各一フレーム分の画像データを生成する描画回路76 と、描画回路76 の動作の一部として、適宜な座標変換によって立体画像を生成するジオメトリエンジン77 と、描画回路76 が生成したフレームバッファFBa，FBb の画像データを読み出して、適宜な画像処理を並列的に実行可能な3 系統

(A/B/C)の表示回路74A~74Cと、3系統(A/B/C)の表示回路74の出力を適宜に選択出力する出力選択部79と、出力選択部79が出力する画像データをLVDS信号に変換するLVDS部80と、シリアルデータ送受信可能なSMC部78と、CPUIF回路56とのデータ送受信を中継するCPUIF部81と、CGROM55からのデータ受信を中継するCGバスIF部82と、外付けDRAM54とのデータ送受信を中継するDRAMIF部83と、内蔵VRAM71とのデータ送受信を中継するVRAMIF部84と、を有して構成されている。なお、音声回路SNDも内蔵されている。

#### 【0143】

図5(b)には、CPUIF部81、CGバスIF部82、DRAMIF部83、及び、VRAMIF部84と、制御レジスタ群70、CGROM55、DRAM54、及び内蔵VRAM71との関係が図示されている。図示の通り、CGROM55から取得したCGデータは、例えば、プリロードデータとして、データ転送回路72及びDRAMIF部83を経由して、外付けDRAM54のプリロード領域に転送される。

#### 【0144】

但し、上記したプリロード動作は、何ら必須動作ではなく、また、データ転送先も、外付けDRAM54に限定されず、内蔵VRAM71であっても良い。したがって、例えば、プリロード動作を実行しない実施例では、CGデータは、データ転送回路72及びVRAMIF部84を経由して、内蔵VRAM71に転送される(図5(b))。

#### 【0145】

ところで、本実施例では、内蔵VRAM71には、CGROM55から読み出した圧縮データの展開領域、表示装置のW×H個の表示ピクセルの各ARGB情報(32bit=8×4)を特定する画像データを格納するフレームバッファ領域、及び、各表示ピクセルの深度情報を記憶するZバッファ領域などが必要となる。なお、ARGB情報において、Aは、8bitのαプレーンデータ、RGBは三原色の8bitデータを意味する。

#### 【0146】

ここで、内蔵VRAM71の上記した各領域は、演出制御CPU63がディスプレイリストDLに記載した各種の指示コマンド(前記したテクスチャやSPRITEなど)に基づいて間接的にアクセスされるが、そのREAD/WRITEアクセスにおいて、一々、内蔵VRAM71のDestinationアドレスや、Sourceアドレスを特定するのでは煩雑である。そこで、本実施例では、CPUリセット後の初期処理において、描画動作で必要となる一次元または二次元の論理アドレス空間(以下、インデックス空間という)を確保して、各インデックス空間にインデックス番号を付与することで、インデックス番号に基づくアクセスを可能にしている。

#### 【0147】

具体的には、CPUリセット後、内蔵VRAM71を3種類のメモリ領域に大別すると共に、各メモリ領域に、必要数のインデックス空間を確保している。そして、インデックス空間とインデックス番号とを紐付けて記憶するインデックステーブルIDXTBL(図9(a)参照)を構築することで、その後のインデックス番号に基づく動作を実現している。

#### 【0148】

このインデックス空間は、(1)初期処理後に追加することや、逆に、(2)開放することも必要となる。そこで、これら追加/開放の演出制御CPU63の動作時に、追加/開放の処理が可能なタイミングか否か、また、追加/開放などの処理が実際に完了したか否か、などを判定可能なフラグ領域FGをインデックステーブルIDXTBLに設けている。なお、内蔵VRAM71は、以下に説明する2つのAAC領域(a1,a2)と、ページ領域(b)と、任意領域(c)の3種類のメモリ領域に大別され、この3種類のメモリ領域(a1,a2)(b)(c)に対応して、インデックステーブルIDXTBLが3区分されている(図9(a))。図示の通り、この実施例では、AAC領域(a)として、第一AAC領域(a1)と第二AAC領域(a2)が確保されているが、特に限定されるものではなく、何れか一方だけでも良い。なお、以下の説明では、第一と第二のAAC領域(a1,a2)を総称する場合には、AAC領域(a)と称する場合がある。

10

20

30

40

50

## 【0149】

本実施例の場合、内蔵VRAM71は、(a) インデックス空間とそのインデックス番号が内部処理によって自動付与され、且つメモリキャッシュ機能を有するAAC領域と、(b) 例えば4096bit×128ラインの二次元空間を単位空間として、その整数倍の範囲でインデックス空間が確保可能なページ領域と、(c) 先頭アドレス（空間先頭アドレス）STxと水平サイズHxが任意に設定できる任意領域と、に区分可能に構成されている（図9（b）参照）。但し、VDP回路52の内部動作を円滑化するため、任意領域(c)において任意設定されるインデックス空間の空間先頭アドレスSTxは、その下位11bitが0であって、所定ビット（2048bit＝256バイト）単位とする必要がある。

## 【0150】

そして、CPUリセット後、各々に必要なアドレス空間の最大値と、領域先頭アドレス（下位11bit＝0）を規定して、AAC領域(a1)と、第二AAC領域(a2)と、ページ領域(b)とが確保され、その残りのメモリ領域が任意領域(c)となる。VDP回路52の内部動作を円滑化するため、AAC領域のアドレス空間の最大値は、2048bit単位で規定され、ページ領域のアドレス空間の最大値は、上記した4096bit×128ラインの単位空間の整数倍とされる。

## 【0151】

次に、このように確保された各領域(a1,a2)(b)(c)に必要個数のインデックス空間が設定される。なお、任意領域(c)を使用する場合、VDP回路52の内部動作を円滑化するため、二次元データを扱うインデックス空間の水平サイズHxは、256bitの倍数として、任意に設定可能である一方、その垂直サイズは固定値（例えば、2048ライン）となっている。

## 【0152】

何れにしても、第一と第二のAAC領域(a1,a2)は、VDP回路52によって、インデックス空間とインデックス番号が自動的に付与されるので、例えば、テクスチャ設定系コマンドのSETINDEXコマンドによって、デコード先をAAC領域(a)に指定すれば、CGROM55からCGデータを読み出すTXLOAD（テクスチャロード）コマンドでは、CGROM55のSourceアドレスと、展開（デコード）後の水平・垂直サイズなどを指定するだけで足りることになる。そこで、本実施例では、予告演出時などに一時的に出現するキャラクターなどの静止画（テクスチャ）や、エストリーム動画については、そのデコード先をAAC領域(a)にしている。

## 【0153】

このAAC領域(a)は、いずれも、メモリキャッシュ機能が付与されているので、例えば、CGROM55の同一のテクスチャを複数回、AAC領域(a)に読み出すような場合には、二度目以降は、AAC領域(a)にキャッシュされているデコードデータが活用可能となり、余分なREADアクセスとデコード処理が抑制可能となる。もっとも、AAC領域(a)を使い切った場合には、古いデータが自動的に破壊されるので、本実施例では、AAC領域(a)を使用する場合、原則として第一AAC領域(a1)を使用することとし、繰り返し使用する特定のテクスチャだけを第二AAC領域(a2)に取得するようにしている。

## 【0154】

繰り返し使用するテクスチャとして、例えば、所定の予告演出時に繰り返し出現するキャラクターや、背景画面を静止画で構築する場合の背景画などを例示することができる。このような場合、テクスチャ設定系コマンドのSETINDEXコマンドによって、デコード先を第二AAC領域(a2)に設定し、TXLOADコマンドによって、キャラクターや背景画などのテクスチャを第二AAC領域(a2)にデコードした後は、第二AAC領域(a2)を使用しないことで、デコード結果を保護する。

## 【0155】

そして、その後、SETINDEXコマンドによって、デコード先を第二AAC領域(a2)に指定した上で、取得済みのテクスチャを再取得する同一のTXLOADコマンドを実行させると、取得済みのテクスチャがキャッシュヒットするので、CGROM55へのREADアクセス

10

20

30

40

50

と、デコード処理に要する時間を削除することができる。後述するように、このようなキャッシュヒット機能は、プリロード領域に先読みされたプリロードデータでも発揮されるが、プリロード領域でキャッシュヒットするプリロードデータは、デコード前の圧縮データであるのに対して、AAC領域でキャッシュヒットするのはデコード後の展開データである点に意義がある。

【0156】

ところで、テクスチャ(texture)とは、一般に、物の表面の質感・手触りなどを指す概念であるが、本明細書では、静止画を構成するスプライト画像データや、動画一フレームを構成する画像データや、三角形や四角形などの描画プリミティブ(primitive)に貼り付ける画像データだけでなく、デコード後の画像データも含む概念として使用している。そして、内蔵VRAM71の内部で、画像データをコピーする(以下、便宜上、移動と称する)場合には、テクスチャ設定系コマンドのSETINDEXコマンドによって、移動元の画像データをテクスチャとして設定した上で、SPRITEコマンドを実行することになる。

【0157】

なお、SPRITEコマンドの実行により、移動元のSource画像データが、形式上は、図9(c)に示す仮想描画空間に描画されるが、表示装置に実際に描画される仮想描画空間内の描画領域と、フレームバッファとなるインデックス空間との対応関係を、予め環境設定コマンド(SETDAVR, SETDAVF)や、テクスチャ設定系コマンド(SETINDEX)によって設定しておけば、例えば、SPRITEコマンドによる仮想描画空間への描画により、所定のインデックス空間(フレームバッファ)には、移動元のSource画像データが描画されることになる(図9(c)参照)。

【0158】

何れにしても、本実施例では、内蔵VRAM71が、AAC領域(a1,a2)とページ領域(b)と任意領域(c)に大別され、各々に、適当数のインデックス空間を確保することができ、各インデックス空間は、各領域(a)(b)(c)ごとに独立のインデックス番号によって特定される。インデックス番号は、例えば、1バイト長であり、(内部回路によって自動付与されるAAC領域(a)を除いた)ページ領域(b)と任意領域(c)については、0~255の範囲で演出制御CPU63が、インデックス番号を自由に付与することができる。

【0159】

そこで、本実施例では、図9(a)に示す通り、表示装置DS1用として、任意領域(c)に、一対のフレームバッファF Baを確保して、ダブルバッファ構造の双方に、インデックス番号255, 254を付与している。すなわち、メイン表示装置DS1用のフレームバッファF Baとして、トグル的に切り換えて使用されるインデックス空間255と、インデックス空間254を確保している。特に限定されないが、このインデックス空間255, 254は、表示装置DS1の横方向ピクセル数に対応して、水平サイズ1280としている。なお、各ピクセルは、ARGB情報32bitで特定されるので、水平サイズ1280は、 $32 \times 1280 = 40960 \text{ bit}$  (256bitの倍数)を意味する。

【0160】

また、表示装置DS2用として、任意領域(c)に、別の一対のフレームバッファF B bを確保して、ダブルバッファ構造の双方にインデックス番号252, 251を付与している。すなわち、サブ表示装置DS2用のフレームバッファF B bとして、インデックス空間252と、インデックス空間251を確保している。このインデックス空間252, 251は、表示装置DS2の横方向ピクセル数に対応して、水平サイズ480としている。この場合も、各ピクセルは、ARGB情報32bitで特定されるので、水平サイズ480は、 $32 \times 480 = 15360 \text{ bit}$  (256bitの倍数)を意味する。

【0161】

なお、フレームバッファF Ba, F B bを任意領域(c)に確保するのは、任意領域(c)には、32バイト(=256bit=8ピクセル分)の倍数として、任意の水平サイズに設定することができ、上記のように、表示装置DS1, DS2の水平ピクセル数に一致させれば、確保領域に無駄が生じないからである。一方、ページ領域(b)には、128ピクセル

× 1 2 8 ラインの単位空間の整数倍の水平／垂直サイズしか設定できない。

【 0 1 6 2 】

但し、任意領域 (c) に確保される二次元のインデックス空間は、その垂直サイズが固定値 (例えば、2 0 4 8 ライン) となっている。そのため、フレームバッファ F B a において、水平サイズ 1 2 8 0 × 垂直サイズ 1 0 2 4 の領域だけが、メイン表示装置 D S 1 にとって有効データ領域となる。この点は、サブ表示装置 D S 2 についても同様であり、フレームバッファ F B b において、水平サイズ 4 8 0 × 垂直サイズ 8 0 0 の領域だけが、サブ表示装置 D S 2 にとって有効データ領域となる (図 9 (c) , 図 1 7 (e) 参照)。

【 0 1 6 3 】

上記の点は更に後述するが、何れにしても、フレームバッファ F B a , F B b は、描画回路 7 6 にとっての描画領域として、各ダブルバッファ (255/254 , 252/251 ) が交互に使用され、また、表示回路 7 4 A , 7 4 B にとっての表示領域として、各ダブルバッファ (255/254 , 252/251 ) が交互に使用される。なお、本実施例では、表示ピクセルの深度情報を記憶する Z バッファを使用しないので欠番 (2 5 3) が生じるが、Z バッファを使用する場合には、任意領域 (c) におけるインデックス番号 2 5 3 , 2 5 0 のインデックス空間 2 5 3 , 2 5 0 が、表示装置 D S 1 と表示装置 D S 2 のための Z バッファとなる。

【 0 1 6 4 】

また、本実施例では、フレームバッファ F B a , F B b が確保された任意領域 (c) に、追加のインデックス空間 (メモリ領域) を確保する場合には、0 から始まるインデックス番号を付与するようにしている。何ら限定されないが、本実施例では、キャラクタやその他の静止画で構成された演出画像を、必要に応じて、適宜な回転姿勢で表示画面の一部に出現させる予告演出用の作業領域として、任意領域 (c) に、インデックス空間 (0) を確保している。

【 0 1 6 5 】

但し、作業領域の使用は必須ではなく、また、任意領域 (c) に代えて、ページ領域 (b) に作業領域としてのインデックス空間を確保しても良い。ページ領域 (b) を使用すれば、水平サイズ 1 2 8 (= 4 0 9 6 b i t) × 垂直サイズ 1 2 8 の正方形の単位空間の倍数寸法のインデックス空間を確保できるので、小型の演出画像を扱うには好適である。

【 0 1 6 6 】

ところで、本実施例では、背景画も含め動画で構成されており、画像演出は、ほぼ動画のみで実現されている。特に、変動演出時には、多数 (通常 1 0 個以上) の動画が同時に描画されている。これらの動画は、何れも、一連の動画フレームとして、圧縮状態で C G R O M 5 5 に格納されているが、I フレームのみで構成された I ストリーム動画と、I フレームと P フレームとで構成された I P ストリーム動画とに区分される。ここで、I フレーム (Intra coded frame) とは、他画面とは独立して、入力画像をそのまま圧縮するフレームを意味する。一方、P フレーム (Predictive coded frame) とは、前方向予測符号化を行うフレームを意味し、時間的に過去に位置する I フレームまたは P フレームが必要となる。

【 0 1 6 7 】

そこで、本実施例では、I P ストリーム動画については、旧データの破壊が懸念される A A C 領域 (a) ではなく、ページ領域 (b) に展開している。すなわち、水平サイズ 1 2 8 × 垂直サイズ 1 2 8 の倍数寸法のインデックス空間を確保可能なページ領域 (b) に、多数のインデックス空間 (I D X 0 ~ I D X N) を確保して、一連の動画フレームは、各動画 M V i に対応する、常に同一のインデックス空間 I D X i を使用してデコードするようにしている。すなわち、動画 M V 1 はインデックス空間 I D X 1 に展開され、動画 M V 2 はインデックス空間 I D X 2 に展開され、以下同様に、動画 M V i はインデックス空間 I D X i に展開されるよう構成されている。

【 0 1 6 8 】

動画 M V i について、更に具体的に説明すると、SETINDEX コマンドによって、「I P ス

10

20

30

40

50

トリーム動画MV i のデコード先は、ページ領域(b)におけるインデックス番号 i のインデックス空間 (i) である」と予め指定した上で、IPストリーム動画MV i の動画フレームを取得するTXLOADコマンドを実行させている。

【0169】

すると、TXLOADコマンドが特定するCGROM55上の動画フレーム（一連の動画フレームの何れか）が、先ず、AAC領域(a)に取得され、その後、自動的に起動するGDEC（グラフィックスデコーダ）75によって、ページ領域(b)のインデックス空間 (i) に、取得した動画フレームがデコードされて展開されることになる。

【0170】

一方、本実施例では、IPストリーム動画については、静止画と同一扱いとしており、SETINDEXコマンドによって、「IPストリーム動画MV j のデコード先は、第一AAC領域(a1)である」と指定して、TXLOADコマンドを実行させる。その結果、動画フレームは第一AAC領域(a1)に取得され、その後、自動的に起動するGDEC75が、第一AAC領域(a1)にデコードデータを展開している。先に説明した通り、AAC領域(a)のインデックス空間は、自動的に生成されるので、インデックス番号を指定する必要はない。なお、インデックス空間に必要な展開ボリューム、つまり、デコードされたテクスチャ（動画フレーム）の水平サイズと垂直サイズは、展開先がAAC領域(a)か、ページ領域(b)かに拘らず、TXLOADコマンドによって特定される。

【0171】

ところで、IPストリーム動画MV i やIPストリーム動画MV j は、一般にN枚の動画フレーム（IフレームやPフレーム）で構成されている。そのため、TXLOADコマンドでは、例えば、k枚目 ( $1 \leq k \leq N$ ) の動画フレームが記憶されているCGROM55のSourceアドレスと、展開後の水平・垂直サイズなどを指定することになる。何ら限定されないが、静止画を殆ど使用しない実施例では、内蔵VRAM71のアドレス空間48Mバイトの大部分（30Mバイト程度）をページ領域(b)に割り当てている。そして、静止画を殆ど使用しない実施例では、AAC領域として、第一AAC領域(a1)だけを確保し、第二AAC領域(a2)を確保せず、また、前記したAAC領域のキャッシュヒット機能も活用しない。

【0172】

なお、圧縮動画データのデコード処理を高速化するため、専用のGDEC（グラフィックスデコーダ）回路を設けることも考えられる。そして、専用のGDEC回路をVDP回路52に内蔵させれば、N枚の圧縮動画フレームで構成された圧縮動画データのデコード処理において、動画圧縮データの先頭アドレスをGDEC回路に指示すれば足りるので、N枚の圧縮動画フレームについて、1枚ごとに先頭アドレスを指定する必要がなくなる。

【0173】

しかし、このような専用のGDEC回路を、圧縮アルゴリズム毎に複数個内蔵させるのでは、VDP回路52の内部構成が更に複雑化する。そこで、本実施例では、ソフトウェアGDECとし、IPストリーム動画、IPストリーム動画、静止画、その他α値などのデータについて、各圧縮アルゴリズムに対応するソフトウェア処理によってデコード処理を実現している。なお、ハードウェア処理とソフトウェア処理の処理時間差は、あまり問題にならず、処理時間が問題になるのは、もっぱら、CGROM55からのアクセス（READ）タイムである。

【0174】

続いて、図5(a)に戻って説明を続けると、データ転送回路72は、VDP回路内部のリソース（記憶媒体）と外部記憶媒体を、転送元ポート又は転送先ポートとして、これらの間でDMA（Direct Memory Access）的にデータ転送動作を実行する回路である。図10は、このデータ転送回路72の内部構成を、関連する回路構成と共に記載したブロック図である。

【0175】

図10に示す通り、データ転送回路72は、ルータ機能を有する統合接続バスICMを経

10

20

30

40

50

由して、CGROM55、DRAM54、及び、内蔵VRAM71とデータを送受信するよう構成されている。なお、CGROM55とDRAM54は、CGバスIF部82や、DMAMIF部83を経由してアクセスされる。

【0176】

一方、CPU回路51は、データ転送回路72に内蔵された転送ポートレジスタTR\_PORTを経由して、描画回路76やプリローダ73にディスプレイリストDLを発行している。なお、CPU回路51とデータ転送回路72は、双方向に接続されているが、ディスプレイリストDLの発行時には、転送ポートレジスタTR\_PORTは、ディスプレイリストDLを構成する一単位のデータを受け入れるデータ書き込みポートとして機能する。なお、転送ポートレジスタTR\_PORTの書き込み単位（一単位データ長）は、CPUバス制御部72dのFIFO構造に対応して32bitとなる。

10

【0177】

図示の通り、演出制御CPU63は、CPUIF部81を経由して、転送ポートレジスタTR\_PORTをWRITEアクセスできる一方、DMAC回路60を活用する場合には、DMAC回路60が、転送ポートレジスタTR\_PORTを直接的にWRITEアクセスすることになる。そして、転送ポートレジスタTR\_PORTに書込まれた一連の指示コマンド（つまり、ディスプレイリストDLを構成する指示コマンド列）は、32bit単位で、FIFO構造（32bit×130段）のFIFOバッファを内蔵したCPUバス制御部72dに、自動蓄積されるよう構成されている。

【0178】

また、このデータ転送回路72は、3チャンネルChA～ChCの伝送経路で、データの送受信動作を実行しており、FIFO構造（64bit×N段）のFIFOバッファを有するChA制御回路72a（N=130段）と、ChB制御回路72b（N=1026段）と、ChC制御回路72c（N=130段）と、を有している。

20

【0179】

そして、CPUバス制御部72dに蓄積された指示コマンド列（ディスプレイリストDL）は、演出制御CPU63によるデータ転送レジスタRGij（各種制御レジスタ70の一種）への設定値に基づき、描画回路76か、又はプリローダ73に転送される。矢印で示す通り、ディスプレイリストDLは、CPUバス制御部72dから、ChB制御回路72bのFIFOバッファを経由して描画回路76に転送され、ChC制御回路72cのFIFOバッファを経由してプリローダ73に転送されるよう構成されている。

30

【0180】

なお、本実施例では、ChB制御回路72bと、ChC制御回路72bは、ディスプレイリストDLの転送動作に特化されており、CPUバス制御部72dのFIFOバッファに蓄積されたデータは、ChB制御回路72bか、ChC制御回路72cのFIFOバッファを経由して、各々、ディスプレイリストDLの一部として、描画回路76かプリローダ73のディスプレイリストアナライザ（Display List Analyzer）に転送される。

【0181】

そして、描画回路76は、転送されたディスプレイリストDLに基づいた描画動作を開始する。一方、プリローダ73は、転送されたディスプレイリストDLに基づき、必要なプリロード動作を実行する。プリロード動作によってCGROM55のCGデータが、DRAM54に確保されたプリロード領域に先読みされ、TXLOADコマンドなどに関して、テクスチャのSourceアドレスを変更したディスプレイリストDL（以下、書換えリストDL'という）が、DRAM54に確保されたDLバッファ領域BUF'に保存される。

40

【0182】

一方、CGROM55、DRAM54、及び、内蔵VRAM71などの記憶媒体の間のデータ転送には、ChA制御回路72aと、接続バスアクセス調停回路72eとが機能する。また、インデックステーブルIDXTBLのアドレス情報が必要になる内蔵VRAM71のアクセス時には、IDXTBLアクセス調停回路72fが機能する。具体的に確認すると、ChA制御回路72aは、例えば、（a）CGROM55の圧縮データを内蔵VRAM71

50

に転送する場合や、(b) CGROM55の圧縮データをプリロード(先読み)して外付けDRAM54に転送する場合や、(c)プリロード領域の先読みデータを、内蔵VRAM71に転送する場合に機能する。

#### 【0183】

ここで、ChA制御回路72aは、ChB制御回路72bやChC制御回路72cと並行して動作可能に構成されており、上記した(a)～(c)の動作は、ディスプレイリストDLの発行動作(図17のST8, 図22のPT11)や、書換えリストDL'の転送動作(図22のPT10)と並行して実行可能となる。また、ChB制御回路72bとChC制御回路72cも、同時実行可能であり、例えば、ChB制御回路72bが機能する図22のステップPT10の処理と、ChC制御回路72cが機能するステップPT11の処理は並行して実行可能である。但し、転送ポートレジスタTR\_PORTは単一であるので、何れか一方(72b/72c)が転送ポートレジスタTR\_PORTを使用しているタイミングでは、他方(72c/72b)は、転送ポートレジスタTR\_PORTをアクセスすることはできない。

#### 【0184】

なお、ChA制御回路72aの動作時に、接続バスアクセス調停回路72eは、統合接続バスICMを経由する各記憶素子(CGROM55、DRAM54)とのデータ伝送を調停(Arbitration)している。一方、IDXTBLアクセス調停回路72fは、インデックステーブルIDXTBLに基づいてChA制御回路72aを制御することで、内蔵VRAM71とのデータ交信を調停している。なお、プリローダ73が機能する実施例の場合、DRAM54のDLバッファ領域BUF'に保存された書換えリストDL'は、接続バスアクセス調停回路72eと、ChB制御回路72bを経由して描画回路76に転送されることになる(図23(b)参照)。

#### 【0185】

上記の通り、本実施例のデータ転送回路72は、各種の記憶リソース(Resource)から任意に選択されたデータ転送元と、各種の記憶リソース(Resource)から任意に選択されたデータ転送先との間で、高速のデータ転送を実現している。図10から確認される通り、データ転送回路72が機能する記憶リソースには、内蔵VRAM71だけでなく、CPUIF部56、CGバスIF部82、DRAMIF部83を経由する外部デバイスも含まれる。

#### 【0186】

そして、CGROM55から1回に取得すべきデータ量(メモリシーケンシャルREAD)のように、ChA制御回路72aが機能する外部デバイスとのデータ転送量は、ChB制御回路72bやChC制御回路72cが機能するディスプレイリストDLの場合と比較して膨大であり、互いに、データ転送量が大きく相違する。

#### 【0187】

ここで、これら各種のデータ転送について、単位データ量や総転送データ量を、細かく設定可能に構成することも考えられるが、これでは、VDP内部の制御動作が煩雑化し、円滑な転送動作が阻害される。そこで、本実施例では、データ転送の最低データ量Dminを一意に規定すると共に、総転送データ量を、最低データ量DTminの整数倍となるよう制限することで、高速で円滑なデータ転送動作を実現している。特に限定されないが、実施例のデータ転送回路72では、最低データ量Dmin(単位データ量)を、256バイトとし、総転送データ量を、この整数倍に制限することにしている。

#### 【0188】

したがって、32bit毎にCPUバス制御部72dのFIFOバッファに蓄積されたディスプレイリストDLの指示コマンド列は、その総量が最低データ量Dminに達したタイミングで、ChB制御回路72bやChC制御回路72bに転送され、各々のFIFOバッファに蓄積されることになる。

#### 【0189】

ディスプレイリストDLは、一連の指示コマンドで構成されているが、本実施例では、転

10

20

30

40

50

送ポートレジスタTR\_PORT の書込み単位 (32 bit) に対応して、ディスプレイリストDLは、コマンド長が、32 bitの整数N倍 (N>0) の指示コマンドのみで構成されている。したがって、データ転送回路72を経由して、ディスプレイリストDLの指示コマンドを受ける描画回路76やプリローダ73は、素早く円滑にコマンド解析処理 (DL analyze) を開始することができる。なお、32 bitの整数N倍のコマンド長は、その全てが有意ビットとは限らず、無意ビット (Don't care bit) も含んで、32 bitの整数N倍という意味である。

#### 【0190】

次に、プリローダ73について説明する。先に概略説明した通り、プリローダ73は、データ転送回路72 (ChC制御回路72b) から転送されたディスプレイリストDLを解釈して、TXLOADコマンドが参照しているCGROM55上のCGデータを、予め、DRAM54のプリロード領域に転送する回路である。また、プリローダ73は、このTXLOADコマンドに関し、CGデータの参照先を、転送後のアドレスに書換えた書換えリストDL'を、DRAM54のDLバッファBUF'に記憶する。なお、DLバッファBUF'や、プリロード領域は、CPUリセット後の初期処理時 (図17のST3) に、予め確保されている。

#### 【0191】

そして、書換えリストDL'は、描画回路76の描画動作の開始時に、データ転送回路72の接続バスアクセス調停回路72eや、ChB制御回路72bを経由して、描画回路76のディスプレイリストアナライザ (DL Analyzer) に転送される。そして、描画回路76は、書換えリストDL'に基づいて、描画動作を実行する。したがって、TXLOADコマンドなどに基づき、本来は、CGROM55から取得すべきCGデータが、プリロード領域に先読みされているプリロードデータとして、DRAM54のプリロード領域から取得される。この場合、プリロードデータは、上書き消去されない限り、繰り返し使用可能であり、プリロード領域にキャッシュヒットしたプリロードデータは、繰り返し再利用される。

#### 【0192】

本実施例では、十分な記憶容量を有する外付けDRAM54にプリロード領域を設定しているので、上記のキャッシュヒット機能が有効に機能する。また、外付けDRAM54の記憶容量が大きいので、例えば、複数フレーム分のCGデータを一気にプリロードする多重プリロードも可能である。すなわち、プリローダ73の動作期間に関し、CGデータの先読み動作を含んだ一連のプリロード動作の動作期間を、VDP回路52の間欠動作時の動作周期 $\delta$ の整数倍の範囲内で、適宜に設定することで多重プリロードが実現される。

#### 【0193】

但し、以下の説明では、便宜上、多重プリロードのない実施例について説明するので、実施例のプリローダ73は、一動作周期 ( $\delta$ ) の間に、一フレーム分のプリロード動作を完了することにする。なお、図17に関し後述するように、本実施例では、VDP回路52の間欠動作時の動作周期 $\delta$ は、表示装置DS1の垂直同期信号の2倍周期である1/30秒である。

#### 【0194】

次に、描画回路76は、データ転送回路72を経由して転送されたディスプレイリストDLや書換えリストDL'の指示コマンド列を順番に解析して、グラフィックスデコーダ75やジオメトリエンジン77などと協働して、VRAM71に形成されたフレームバッファに、各表示装置DS1, DS2の一フレーム分の画像を描画する回路である。

#### 【0195】

上記の通り、プリローダ73を機能させる実施例では、書換えリストDL'のCGデータの参照先は、CGROM55ではなく、DRAM54に設定されたプリロード領域である。そのため、描画回路76による描画の実行中に生じるCGデータへのシーケンシャルアクセスを迅速に実行することができ、動きの激しい高解像度の動画についても問題なく描画することができる。すなわち、本実施例によれば、CGROM55として、安価なSATAMジュールを活用しつつ、複雑高度な画像演出を実行することができる。

## 【0196】

ところで、プリローダ73を機能させるか否かに拘らず、ディスプレイリストDLや書換えリストDL'の転送時に、仮にデータ化けが発生しても、描画回路76は、これを検出することはできない。また、ノイズなどの影響で、描画回路76がフリーズして、内蔵VRAM71のREAD/WRITEアクセスが異常停止することもあり得る。そこで、本実施例では、描画回路76が不合理な指示コマンド(analyze 不能のビット並び)を検出した場合や、一定期間、内蔵VRAM71に対してREAD/WRITEアクセスがない場合には、描画異常割込みを発生させるよう構成されている(描画異常割込みが許可状態)。なお、この点は、図17(d)に関して後述する。

## 【0197】

次に、図9に関して説明した通り、VRAM71の任意領域(c)に確保されたフレームバッファFBは、描画領域と読出領域に区分されたダブルバッファであり、2つの領域を、交互に用途を切り替えて使用する。また、本実施例では、2つの表示装置DS1, DS2が接続されているので、図9に示す通り、2区画のフレームバッファFba/Fbbが確保されている。したがって、描画回路76は、表示装置DS1用のフレームバッファFbaの描画領域(書込み領域)に、一フレーム分の画像データを描画すると共に、表示装置DS2用のフレームバッファFbaの描画領域(書込み領域)に、一フレーム分の画像データを描画することになる。なお、描画領域に、画像データが書込まれているとき、表示回路74は、他方の読出領域(表示領域)の画像データを読み出して、各表示装置DS1, DS2に出力する。

## 【0198】

表示回路74は、フレームバッファFba, Fbbの画像データを読み出して、最終的な画像処理を施した上で出力する回路である(図11)参照)。最終的な画像処理には、例えば、画像を拡大/縮小するスケーラのスケーリング処理、微妙なカラー補正処理、画像全体の量子化誤差が最小化するディザリング処理が含まれている。そして、これらの画像処理を経たデジタルRGB信号(合計24bit)が、水平同期信号や垂直同期信号と共に出力される。図11に示す通り、本実施例では、上記の動作を並列的に実行する3系統の表示回路A/B/Cが設けられており、各表示回路74A~74Cは、各々に対応するフレームバッファFba/Fbb/Fbcの画像データを読み出して、上記の最終画像処理を実行する。但し、本実施例では、表示装置は2個であるので、フレームバッファFbcは確保されておらず、表示回路74Cが機能することもない。

## 【0199】

この動作に関連して、この実施例の出力選択部79は、表示回路74Aの出力信号を、LVDS部80aに伝送し、表示回路74Bの出力信号を、LVDS部80bに伝送している(図11)。そして、LVDS部80aは、画像データ(合計24bitのデジタルRGB信号)をLVDS信号に変換して、クロック信号を伝送する一対を加えて、全五対の差動信号としてメイン表示装置DS1に出力している。なお、メイン表示装置DS1には、LVDS信号の変換受信部RVが内蔵されており、LVDS信号からRGB信号を復元して、表示回路74Aの出力に対応する画像を表示している。

## 【0200】

この点は、LVDS部80bも同様であり、各8bitのデジタルRGB信号の合計24bitについて、クロック信号を伝送する一対を加えて、全五対の差動信号として変換受信部RVに出力し、サブ表示装置DS2が変換受信部RVから受ける合計24bitのRGB信号による画像表示を実現している。そのため、サブ表示装置DS2と、メイン表示装置DS1は、 $28 \times 28 \times 28$ の解像度を有することになる。

## 【0201】

なお、必ずしもLVDS信号とする必要は無く、例えば伝送距離が短い場合には、デジタルRGB信号を、デジタルRGB部80cを経由して、そのまま表示装置に伝送するか、或いは、伝送距離が長い場合には、デジタルRGB信号を、変換送信部TR'において、V-by-one(登録商標)信号に変換して変換受信部RV'に伝送した後、変換受信部R

10

20

30

40

50

V'においてデジタルRGB信号に戻すのも好適である。なお、図11の破線は、この動作態様を示しているが、出力選択部79の動作を適宜に設定することで、表示回路74A～74Cの何れの出力信号であっても上記の動作が可能となる。

#### 【0202】

ところで、本実施例の場合、各表示回路74A～74Bには、表示タイミングに対して、表示データの生成が間に合わなかったUnderrun異常をカウントするアンダーランカウンタURCNTa～URCNTcが設けられている（図11参照）。そして、このアンダーランカウンタURCNTa～URCNTcのカウント値は、アンダーラン異常が発生するとVBLANK毎に自動的に加算されるよう構成されている。

#### 【0203】

次に、SMC部78（Serial Management Controller）は、LEDコントローラとMotorコントローラとを内蔵した複合コントローラである。そして、外部基板に搭載したLED/Motorドライバ（シフトレジスタを内蔵するドライバIC）に対して、クロック信号に同期してLED駆動信号やモータ駆動信号を出力する一方、適宜なタイミングで、ラッチパルスを出力可能に構成されている。

#### 【0204】

上記したVDP回路52の内部回路及びその動作に関し、内部回路が実行すべき動作内容は、演出制御CPU63が、制御レジスタ群70に設定する動作パラメータ（設定値）で規定され、VDP回路52の実行状態は、制御レジスタ群70の動作ステータス値をREADすることで特定できるようになっている。制御レジスタ群70は、演出制御CPU63のメモリマップ上、1Mバイト程度のアドレス空間（0～FFFFFFH）にマッピングされた多数のVDPレジスタRGijを意味し、演出制御CPU63は、CPUIF部81を経由して動作パラメータのWRITE（設定）動作と、動作ステータス値のREAD動作を実行するようになっている（図5（b）参照）。

#### 【0205】

制御レジスタ群70（VDPレジスタRGij）には、割り込み動作などシステム動作に関する初期設定値が書込まれる「システム制御レジスタ」と、内蔵VRAMにAAC領域(a)やページ領域(b)を確定する共に、インデックステーブルIDXTBLを構築又は変更などに関する「インデックステーブルレジスタ」と、演出制御CPU63とVDP回路52の内部回路との間のデータ転送回路72によるデータ転送処理に関する設定値などが書込まれる「データ転送レジスタ」と、グラフィックスデコーダ75の実行状況を特定する「GDECレジスタ」と、指示コマンドや描画回路76に関する設定値が書込まれる「描画レジスタ」と、プリローダ73の動作に関する設定値が書込まれる「プリローダレジスタ」と、表示回路74の動作に関する設定値が書込まれる「表示レジスタ」と、LEDコントローラ（SMC部78）に関する設定値が書込まれる「LED制御レジスタ」と、Motorコントローラ（SMC部78）に関する設定値が書込まれる「モータ制御レジスタ」と、音声回路SNDに関する設定値が書込まれる「音声制御レジスタSRG」と、が含まれている。但し、本実施例では、音声回路SNDを活用していない。

#### 【0206】

何れにしても、以下の説明では、制御レジスタ群70に含まれる一又は複数のレジスタRGijを、上記した個別名称で呼ぶ場合と、VDPレジスタRGijと総称することがあるが、何れにしても、演出制御CPU63は、所定のVDPレジスタRGijに、適宜な設定値を書込むことで、VDP回路52の内部動作を制御している。具体的には、演出制御CPU63は、適宜な時間間隔で更新するディスプレイリストDLと、所定のVDPレジスタRGijへの設定値に基づいて、所定の画像演出を実現している。なお、この実施例では、ランプ演出やモータ演出も含め、演出制御CPU63が担当するので、VDPレジスタRGijには、LED制御レジスタやモータ制御レジスタも含まれる。

#### 【0207】

続いて、上記したCPU回路51とVDP回路52とを内蔵した複合チップ50によって実現される、画像演出、音声演出、モータ演出、及び、ランプ演出の統一的な演出制御動

10

20

30

40

50

作について説明する。

#### 【0208】

本実施例の場合、複合チップ50の動作は、電源投入や異常リセットによるパワーオンリセット動作（図12（a）参照）によって開始され、初期設定プログラム（ブートプログラム）Pinitによる初期設定処理（SP1～SP9）を経て、演出制御プログラムMain及び割込み処理プログラム（ベクタハンドラ）Voptによるメイン制御処理（SP10）に移行するよう構成されている。メイン制御処理については、図14（a）に、その導入部の処理内容が記載されており、本体部の処理内容が図17（a）に記載されている。なお、図14のステップSP27の処理は、図17（a）のステップST1～ST3の処理を含んでいる。

10

#### 【0209】

以上を踏まえて、パワーオンリセット動作について図12（a）に基づいて説明する。電源投入時など、システムリセット信号SYSが所定の期間（アサート期間）Lレベルを維持すると、全ての動作制御レジスタREGや、全てのVDPレジスタRGijは、所定のデフォルト値に自動設定される。そして、その後、システムリセット信号SYSがHレベル（ネゲートレベル）に変化すると、本実施例では、最初に、アドレス空間CS0の先頭番地からの32bitデータが、演出制御CPU63のプログラムカウンタPCに設定され、これに続く32bitデータが、スタックポインタSPに設定されるよう構成されている。なお、図7や図13（c）では、プログラムカウンタPCやスタックポインタSPの初期値を記憶するメモリの先頭領域をベクタテーブルVECTと称している。

20

#### 【0210】

図12（b）に示す通り、このベクタテーブルVECTには、優先度と割込み要因などを特定するベクタ番号と、アドレス情報とが対応して記憶されている。ベクタ番号は、その番号が小さいほど優先度が高いが、例えば、ベクタ番号11は、マスク不能割込み（NMI）であって、アドレス情報として、NMI割込み時に実行される割込み処理プログラムの先頭アドレスが記憶されている。また、ベクタ番号64は、VDPからの内部割込み（VDP\_IRQ0）であって、アドレス情報として、VDP\_IRQ0割込み時に実行される割込み処理プログラムの先頭アドレスが記憶されている。

#### 【0211】

割込み優先度は、図14（d）に示す通りであるので、ベクタ番号64より小さいベクタ番号の欄には、制御コマンド受信割込みIRQ\_CMDと、20μsタイマ割込みと、1msタイマ割込みについて、割込み処理プログラムの先頭アドレスが各々記憶されていることになる。一方、ベクタ番号64より大きいベクタ番号の欄には、VDP\_IRQ1より優先度が低い割込み処理プログラム（IRQ\_SND，IRQ\_RTCなど）の先頭アドレスが各々記憶されている。

30

#### 【0212】

また、ベクタテーブルVECTにおいて、ベクタ番号0とベクタ番号1は、パワーオンリセット時にCPUのプログラムカウンタと、スタックポインタに自動設定されるべき、設定値が規定されている。図12（b）に示す通り、この実施例では、パワーオンリセット時（リセットアサート期間）の内部動作として、4バイトデータ「\*\*\*\*」が、プログラムカウンタPCに設定され、4バイトデータ「++++」がスタックポインタSPに設定される。なお、「\*\*\*\*」は、アドレス空間CS0に不揮発的に記憶されている初期設定プログラムPinit（図12のSP1～SP9）の先頭アドレス値であり、「++++」は、内蔵RAM59に確保された、LIFO（Last-In First-Out）方式で機能するスタック領域の先端又は終端のアドレス値である。

40

#### 【0213】

なお、本実施例では、レジスタバンクRBiを有効活用するので、割込み処理時に、スタック領域が消費されることなく、それほどのメモリ容量は必要とされない。すなわち、本実施例では、スタック領域は、専ら、関数処理や、サブルーチン処理において活用される。

#### 【0214】

50

以上の動作の結果、その後、演出制御CPU63は、アドレス値「\*\*\*\*」以降に記載された初期設定プログラムPinitを実行することになる。但し、アドレス空間CS0のメモリREAD動作は、バスステートコントローラ66（図6）の動作を規定する動作制御レジスタREGのデフォルト値（初期値）に基づいて実行される。この動作制御レジスタREGの初期値は、リセットアサート期間（システムリセット信号SYSがLレベルを維持する図4（d）に示す期間）に自動的に設定される値であり、アドレス空間CS0を、如何なるメモリデバイスで構成しても、問題なくREADアクセスできるよう、最遅READアクセス動作（デフォルトアクセス動作）に設定されている。

#### 【0215】

そこで、このデフォルトアクセス動作を、最適なアクセス動作に変更するべく、最初に、アドレス空間CS0に対するバスステートコントローラ66（図6）の動作を規定する所定の動作制御レジスタREGに最適値を設定する（SP1）。すなわち、初期設定プログラムPinit（SP1～SP9）、演出制御プログラムMainB（SP10）、定数データなどを格納したPROM53をアクセスする場合のメモリREAD動作を、メモリデバイスに合わせて最適化するべく、バス幅やページアクセスの有無を設定すると共に、チップセレクト信号CS0や、READ制御信号や、WRITE 制御信号その他の動作タイミングを最適設定する（図35参照）。

#### 【0216】

以上の設定の結果、ステップSP2以降の処理は、アドレス空間CS0に記憶されているプログラムを、最適にメモリREADして実行されることになる。そこで、次に、演出制御CPU63が、VDPレジスタRGijをアクセスする場合のREAD／WRITE アクセス動作を最適化するべく、VDPレジスタRGijに対するバスステートコントローラ66（図6）の動作を規定する所定の動作制御レジスタREGに最適値を設定する（SP2）。

#### 【0217】

先に説明した通り、本実施例では、VDPレジスタRGijは、演出制御CPU63のアドレス空間CS7に位置付けられているので、チップセレクト信号CS7や、その他の制御信号の動作タイミングを最適設定するべく所定の動作制御レジスタREGに所定値を書込むことになる。

#### 【0218】

続いて、特定のVDPレジスタRGijのレジスタ値を読み出して、その値が所定値（デバイスコード）か否かを判定する（SP3）。これは、VDP回路52のシステムクロックが安定化したことの確認判定である。すなわち、VDP回路52は、PLLREF端子に供給される発振器OSC2の発振出力に基づいて動作するが、このVDP回路52が、CPU回路51からの指令（つまり、VDPレジスタRGijへの設定など）を正常に受け付け可能か否かの判定である。

#### 【0219】

そして、デバイスコードの読出し処理（SP3）によって、システムクロックが安定化したことが確認できれば、その後は、VDP回路52の正常動作を期待できるので、所定のVDPレジスタRGijに対する設定処理を実行する（SP4～SP6）。具体的には、先ず、演出制御CPU63からVDPレジスタRGijをアクセスする場合のエンディアン設定（ビッグ／リトル）や、データバス幅を設定する（SP4）。

#### 【0220】

なお、本実施例では、設定値の最上位ビット(Most significant Bit)を、VDPレジスタRGijの最上位ビットに格納するビックエンディアンに設定し、データ32バス幅を32bitに設定するが、これらの設定値が、仮に、デフォルト値と同じであれば、これらの設定処理を省略することもできる（以下の処理も同様）。

#### 【0221】

次に、VDP回路からCPU回路への内部割込み（VDP\_IRQ0，VDP\_IRQ1，VDP\_IRQ2，VDP\_IRQ3）について、割込み有意レベル（H／L）を設定し、PLLREF端子（図5（a）参照）へのクロック信号（リファレンスクロック）に基づいてDDR（DRAM54

10

20

30

40

50

）を機能させる旨を設定する（SP4）。なお、PLLREF端子に、発振器OSC2のリファレンスクロックが供給されることは図5（a）に関して説明した通りである。

#### 【0222】

続いて、図7に示すメモリマップを実現するべく、アドレス空間CS1～CS6を定義する（SP5）。先に説明した通り、アドレス空間CS3は、音声プロセッサ27の内部レジスタに付与され、アドレス空間CS4は、RTC38の内部レジスタやSRAM39のアドレス空間に付与され、アドレス空間CS5は、外付けDRAM（DDR）54に付与され、アドレス空間CS6は、内蔵CPUのワークメモリ57に付与される。

#### 【0223】

なお、VDPレジスタRGijが、アドレス空間CS7に割り当てられることは固定的に規定されているので、アドレス空間CS7の定義処理は不要である。また、アドレス空間CS0は、CPU回路51のメモリマップ0x00000000番地以降であることは予め固定的に規定されており、この規定を前提として、アドレス空間CS0が、CGROM55に確保されているか、その他のメモリデバイスに付与されるかは、HBTS端子のH/Lレベルで規定される。

#### 【0224】

先に説明した通り、本実施例では、HBTS端子=Lとなっており、CGROM55以外にアドレス空間CS0が定義されていることが示されている。そして、CGROM55以外である制御メモリ53の具体的なバス幅や、最適なアクセス動作については、ステップSP1において設定済みであるので、アドレス空間CS0についても、ステップSP5の処理は不要である。

#### 【0225】

続いて、ステップSP5の処理で定義されたアドレス空間CS1～CS6について、各アドレス空間CSiをアクセスする場合のバス幅やページアクセスの有無について、所定の動作制御レジスタREGに所定値を書込む（SP6）。また、チップセレクト信号CSiその他を最適設定するべく、所定の動作制御レジスタREGに所定値を書込む（SP6）。これらの処理は、ステップSP1やSP2の処理と同様の内容であり、バスステートコントローラ66（図6）の動作を規定する動作制御レジスタへの書込み処理によって、チップセレクト信号CSi、READ制御信号、WRITE制御信号、その他の動作タイミングが最適に設定される。

#### 【0226】

続いて、WDT回路58にクリア信号を出力することで、異常リセットを回避する（SP7）。これは、電源投入後、WDT回路58が自動的に動作を開始することを考慮したものであり、この後も、繰り返し同様の処理が実行される。なお、ステップSP9の処理は、サブルーチンSP7として制御メモリ53に格納されているが、ステップSP9の終了時までは、制御メモリ53のサブルーチンSP7が呼び出され、ステップSP9の終了後は、外付けDRAM54に転送された別のサブルーチンSP7'が呼び出されて実行される。

#### 【0227】

続いて、アドレス空間CS0に格納されているプログラムやデータのうち、図12（b）や図13（c）に示すベクタハンドラVopt（割り込み処理プログラム）、エラー復帰処理プログラムPiram、演出制御プログラムMainB、初期値有り変数D、及び、定数データCを、外付けDRAM54や、内蔵RAM59に転送する（SP8）。なお、初期値有り変数Dとは、所定の変数領域に記憶されている初期値データを意味する。このメモリセクションの初期化処理（SP8）は、演出制御処理の高速化を図るため、プログラムやデータを転送する処理であり、アクセス速度に劣るROMへのアクセスを回避するための処理である。

#### 【0228】

そして、次に、レジスタバンクRBiを使用する旨の設定をする（SP9）。そのため、その後は、割り込み処理時に、レジスタバンクRB0～RB14が機能することになり、割

10

20

30

40

50

込み処理が迅速化されると共に、スタック領域の消費が緩和される。

【0229】

以上の処理は、アドレス空間CS0である制御メモリ53に格納されている「初期設定プログラムPinit」の実行によって実現される(図13(c)参照)。そして、この初期設定プログラムPinitの実行が終われば、続いて、演出制御プログラムMainによるメイン制御処理を実行する(SP10)。ここで、メイン制御処理の実行とは、ステップSP8の転送処理によって、制御メモリ53から外付けDRAM54に転送された「演出制御プログラムMain」の実行を意味する(図12(b)参照)。

【0230】

メイン制御処理(演出制御プログラムMain)の具体的な内容については、図14(a)や、図17(a)に基づいて説明するが、それに先行して、メモリセクションの初期化処理(SP8)について説明する。図13(a)に示す通り、メモリセクションの初期化処理(SP8)では、最初に複数チャンネルのDMACを動作停止状態に初期設定する。なお、この処理は、念のための形式的な処理に過ぎない。

【0231】

以上の処理が終われば、所定チャンネルのDMACiを起動させて、制御メモリ53の記憶されているベクタハンドラVopt(割込み処理プログラム)を、内蔵RAM59に、ノンストップ転送方式(図8(b3)参照)でDMA転送する。本実施例では、割込み処理プログラムVoptを内蔵RAM59に転送するので、外付けDRAM54の異常時においても、適切な異常対応処理が可能となる。

【0232】

その後の処理も同じであり、所定チャンネルのDMACiを使用して、ノンストップ転送方式で実行され、エラー復帰処理プログラムPiramを内蔵RAM59にDMA転送する(SP62)。本実施例では、エラー復帰処理プログラムPiramを内蔵RAM59に転送するので、エラー復帰処理において、周辺回路を確実にリセット状態にすることができる。例えば、エラー復帰処理プログラムPiramを、内蔵RAM59以外の例えば外付けDRAM54に転送すると、エラー復帰処理時に、外付けDRAM54をリセット処理できないことになる。

【0233】

次に、演出制御プログラムMainを、外付けDRAM54にDMA転送し(SP63)、定数データCを、外付けDRAM54にDMA転送する(SP64)。定数データには、演出抽選に使用する抽選データや、図17(b)に示す各種の駆動データテーブルにおける、ランプ駆動データやモータ駆動データが含まれる。また、初期値の有る変数Dを、外付けDRAM54にDMA転送するが(SP65)、これら何れも、所定チャンネルのDMACiを使用したノンストップ転送方式で実行される。

【0234】

最後に、外付けDRAMの変数領域Bの先頭にクリアデータを書込む(SP66)。この先頭アドレスを、仮にADbとすると、その後のDMA転送処理では、転送元アドレスをADbとし、転送先アドレスをADb+1と初期設定した後、各アドレス値ADb、ADb+1をインクリメント処理しつつ、このクリアデータを拡散させることで、変数領域Bのクリア処理を実行することになる(SP67)。

【0235】

以上説明したステップSP61~SP66、及びステップSP67の処理は、何れも、類似の動作であり図13(b)に示す通りである。すなわち、まず、所定チャンネルのDMACiに関し、DMA転送条件として、(1)サイクルスチール転送モード、(2)ノンストップ転送方式を採り、(3)SourceとDestinationのアドレス値をincrement更新すると、設定する(SP68)。

【0236】

次に、転送元Sourceアドレスと、転送先Destinationアドレスの初期値を設定し(SP69)、転送サイズを設定し、割込み禁止などに設定した上で(SP70)、DMA転送

10

20

30

40

50

の動作を開始させる（SP71）。なお、ステップSP68～SP71の設定は、何れも所定の動作制御レジスタREGへの設定動作によって実現される。

#### 【0237】

このメモリセクションの初期化処理では、DMA転送終了の割込みを禁止設定しているので（SP70）、DMA転送の動作を開始させた後は、所定の動作制御レジスタREGのステータスフラグを、繰り返しREADアクセスして、DMA転送の終了を待つ（SP72）。但し、動作終了までの処理時間を考慮して、WDT回路58に対して、クリア信号を繰り返し出力する（SP73）。そして、DMA転送の終了時には、所定の動作制御レジスタREGへの設定動作に基づいてDMACiを停止設定する。

#### 【0238】

続いて、メイン制御処理の動作内容について図14～図17に基づいて説明する。先に説明した通り、メイン制御処理については、図14（a）に、その導入部（SP20～SP27）の処理内容が記載されており、本体部（ST4～ST14）の処理内容が図17（a）に記載されている。なお、図14のステップSP27の処理は、図17（a）のステップST1～ST3の処理を含んでいる。

#### 【0239】

図14（a）に示す通り、メイン制御処理（導入部）では、最初に、CGROM55について、そのバス幅やROMデバイスの種別を特定する（SP20）。具体的には、図15（a）に示す通りであり、CGROM55とのインタフェースを司るCGバスの動作状態を特定する所定のVDPLレジスタRGij（例えばCGバスStatusレジスタ）をREADアクセスして（SP80）、CGバスについて動作設定が可能か否かを判定する（SP81）。

#### 【0240】

ここで、CGバスStatusレジスタの値が1であれば、CGバスの内部回路がリセット動作中であることを意味し、VDPLレジスタRGijへの設定値を受け付けることができないことを意味する。そこで、CGバスStatusレジスタの値が1から0に変化していることを確認した上で（SP81）、CGROMを構成するメモリデバイスに対応して規定可能なデバイス区間（SPA0～SPAN）毎に（1）各デバイス区間SPAiの有効／無効、（2）ROMデバイスの種別、（3）データバス幅などの動作パラメータを、所定のVDPLレジスタRGijに設定する（SP82）。

#### 【0241】

図14（a）に示す通り、この実施例では、CGROM55を複数領域（デバイス区間）に区分できるようになっており、例えば、デバイス区間（SPA0～SPAN）毎に、メモリデバイスや、データバス幅を選択可能に構成されている。メモリデバイスとしては、例えば、（1）本実施例で採用するSATAモジュール（AHSE／F）、（2）パラレルI／F（Interface）形式を採るメモリ素子、（3）シーケンシャルI／F形式を採るメモリ素子などに大別されるが、大別されたメモリデバイスごとに、メモリデバイスを具体的に選択可能であり、且つ、データバス幅などを任意に規定できるようになっている。

#### 【0242】

次に、デバイス区間（SPA0～SPAN）毎に選択されたメモリデバイスとのメモリREAD動作を最適化すべく、所定の動作パラメータを、所定のVDPLレジスタRGijに設定する（SP83）。動作パラメータには、チップセレクト信号と、その他の制御信号（READ制御信号など）との動作タイミングを規定する設定値が含まれている。また、シーケンシャルI／F形式を採るメモリ素子が選択された場合には、図15（b）も示す動作を実現すべく、アドレスラッチの出力タイミングや、読出しクロック数なども特定される。

#### 【0243】

したがって、種類の異なるメモリデバイスを組合せてCGROM55を構成することもできる。但し、本実施例では、SATAモジュールだけを使用してCGROM55を構成し、デバイス区間（SPA0）だけを有効化し、他のデバイス区間（SPA1～SPAN）

10

20

30

40

50

を無効化している。

【0244】

何れにしても、ステップSP82～SP83の設定処理が終われば、その設定処理の実効化を図るべく、所定のVDPレジスタRGijに所定値を書込む（SP84）。これは、CGバスの内部回路がステップSP82～SP83の設定処理に対応して動作できるまでに所定の時間を要することを考慮したものであり、内部回路の動作中は、前記したCGバスStatusレジスタ（SP80参照）の値が0となる。

【0245】

したがって、その後は、CGバスStatusレジスタを繰り返しREADアクセスして（SP85）、Statusレジスタの値が1から0に戻ることを確認して処理を終える（SP86）。なお、所定回数の判定に拘らず、Statusレジスタの値が1から0に戻らない場合に、ステップSP66の処理を終えても良い。但し、その場合CGROMが正常にアクセスできない状態で遊技処理が始まるので、その後、何れかのタイミングでWDT回路58が起動して複合チップ50が異常リセット状態になる。そして、この場合は、再度、パワーオンリセット動作が実行されることになる。

【0246】

一方、図14のステップSP20の処理が、正常に実行された後は、割込みコントローラINTCや、DMAC回路60や、マルチファンクションタイマユニットMTUなど、CPU回路51の内蔵回路をソフトウェア処理によって個々の初期化する（SP21）。

【0247】

次に、マルチファンクションタイマユニットMTUについて、所定のタイマ計測動作を開始させた後（SP22）、内部割込み及び内部割込みについて、所定の動作制御レジスタREGに許可設定値を書込んで割込み許可状態に設定する（SP23）。

【0248】

その結果、その後は、図14（d）に示す各種の割込みが生じ得ることになる。通常、このタイミングでは、音声プロセッサ27は、その初期化シーケンスを終えているので、図4（c）に示す通り、終了割込み信号IRQ\_SNDはLレベルに降下している筈である。そのため、図14（c）に示す割込み処理が起動され、演出制御CPU63は、エラーフラグERRを1に初期設定すると共に、アドレス空間CS3をREADアクセスして（SP30）、音声プロセッサ27の所定の音声レジスタSRGの値を取得して、初期化シーケンスが正常に終了しているか否かを判定する（SP31）。

【0249】

そして、万一、初期化シーケンスが正常に終了していない場合には、演出制御CPU63は、音声プロセッサ27の所定の音声レジスタSRGにリセットコマンドを書込むと共に（SP32）、1に初期設定されているエラーフラグERRを2にセットする（SP33）。このエラーフラグERRは、音声プロセッサ初期化処理（SP26）を実行するか否かを規定しており、エラーフラグERR=1がステップSP26の実行条件となっている。

【0250】

一方、音声プロセッサ27は、リセットコマンドを受けたことに対応して、終了割込み信号IRQ\_SND=Hレベルの状態、再度、初期化シーケンスを開始し、初期化シーケンスが終われば、終了割込み信号IRQ\_SNDはLレベルに降下させる。この結果、図14（c）の処理が再実行されることになる。

【0251】

以上、初期化シーケンスが正常に終了していない例外的な場合について説明したが、通常は、ステップSP31に続いて、ステップSP32の処理が実行され、演出制御CPU63は、所定の音声レジスタSRGに、所定値を書込むことで終了割込み信号IRQ\_SNDを、LレベルからHレベルに復帰させる（SP34）。

【0252】

そして、最後に、所定の音声レジスタSRGに所定値を書込むことで、全ての音声レジスタSRGへのREAD／WRITEアクセスを許可する（SP35）。この処理の結果、その後

10

20

30

40

50

の音声プロセッサ初期化処理（SP26）では、必要な設定処理を実行できることになる。

【0253】

以上、ステップSP23の割込み許可設定に対応するMaskable Interruptの一例について説明したが、発振器OSC2の発振停止に基づくマスク不能割込み（Non Maskable Interrupt）は、任意のタイミングで起動可能である。先に説明した通り、内蔵CPU（演出制御CPU63）以外の回路の動作クロックは、発振器OSC2の出力クロックをPLL（Phase Locked Loop）で周波数逡倍して生成されており、発振器OSC2の発振が停止されれば、その後のVDP回路52の正常動作は不可能である。

【0254】

一方、演出制御CPU63の動作クロックは、発振器OSC1の出力クロックをPLLで逡倍して生成されており、プログラム処理は継続可能である。しかも、割込み処理プログラムは、内蔵RAM59に格納されている。そこで、演出制御CPU63は、異常事態の発生を音声やランプによって異常報知すると共に（SP28）、WDT回路58にクリア信号を出力し続ける（SP29）。異常報知は、例えば「異常事態が発生しました。至急、係員に連絡して下さい」との音声報知となる。なお、WDT回路58にクリア信号を出力し続けるのは、異常リセット動作を回避するためである。すなわち、発振器OSC1が動作を停止する重大な異常時には、仮に、異常リセット処理を繰り返しても、機器の正常復帰が望めないと考えられるからである。

【0255】

以上、図14（b）と、図14（c）について説明したので、図14（a）に戻って説明を続ける。ステップSP24では、外付けDRAMのプログラム領域を保護するため、必要領域を書込み禁止に設定する。次に、電源遮断時に電池で駆動されている時計回路38について、電源遮断時の正常動作を確認すると共に、念のためアラーム割込みについて再設定する（SP25）。

【0256】

そして、エラーフラグERR=1であることを条件に、音声プロセッサ27の内蔵レジスタ（音声レジスタSRG）に、必要な設定値を書込んで初期化処理を実行する（SP26）。なお、エラーフラグERR=0の場合は、所定時間、エラーフラグERR=1となるまで待機するが、限界時間を超える場合には、WDT回路58を起動させるべく無限ループ処理に移行する。

【0257】

次に、VDPレジスタRGijに、必要な設定値を書込むことで、VDP回路52の初期化処理を実行する（SP27）。なお、ステップSP27の処理には、図17のST1～ST3の処理が含まれている。

【0258】

以上、音声プロセッサから終了割込み信号IRQ\_SNDを受ける実施例について説明したが、図14（c）の割込み処理を省略するのも好適である。図16は、変形実施例を示しており、終了割込み信号IRQ\_SNDに代えて、マルチファンクションタイマユニットMTUが生成する1msタイマ割込信号を活用している。

【0259】

図16は、1msタイマ割込処理の一部を図示したものであり、初期状態がゼロである動作管理フラグFLAGの値（0／1／2／3）に基づいて、4段階の動作を実現している。なお、音声プロセッサ27のIRQ\_SND出力端子は、開放状態とし、CPU回路51のIRQ\_SND入力端子は、Hレベルに固定されている。

【0260】

1msタイマ割込み処理において、まず、ステップSP42の処理で、動作管理フラグFLAG=0と判定される場合には、音声プロセッサ27の初期化シーケンスが正常終了していることを確認する（SP43）。そして、正常終了している場合には、所定の音声レジスタSRGに所定値を書込むことで割込み信号（IRQ\_SND）をクリアさせ（SP46）、動作管理フラグFLAGを1にする（SP47）。なお、ステップSP43とSP46の

10

20

30

40

50

処理は、図 1 4 (c) のステップ S P 3 1 と S P 3 4 の処理と同じである。

【0 2 6 1】

一方、初期化シーケンスが正常終了していない場合には、所定の音声レジスタ S R G にリセットコマンドを書込むことで、音声プロセッサ 2 7 に初期化シーケンスを起動させ (S P 4 4)、動作管理フラグ F L G をゼロに戻す (S P 4 5)。なお、ステップ S P 4 4 の処理は、図 1 4 (c) のステップ S P 3 2 の処理に対応している。

【0 2 6 2】

通常は、ステップ S P 4 7 の処理を経て動作管理フラグ F L G = 1 となるので、次の 1 m s タイマ割り込みでは、所定の音声レジスタに所定値を書込むことで全ての音声レジスタへのアクセスを許可し (S P 4 8)、動作管理フラグ F L G = 2 に設定する (S P 4 9)。

10

【0 2 6 3】

次に、動作管理フラグ F L G = 2 の 1 m s タイマ割り込みでは、図 1 4 (a) のステップ S P 2 6 の場合と同様に、音声プロセッサ 2 7 の内蔵レジスタ (音声レジスタ S R G) に、必要な設定値を書込んで初期化処理を実行し (S P 5 0)、動作管理フラグ F L G = 3 に設定する。

【0 2 6 4】

動作管理フラグ F L G = 3 は、通常の音声制御状態を意味し、必要な音声レジスタ S R G に、必要な動作パラメータを設定することで、音声制御を進行させる (S P 5 2)。

【0 2 6 5】

20

以上、音声プロセッサ 2 7 の初期化シーケンスの正常終了を、割り込み信号 (I R Q \_ S N D) に起因する割り込み処理で確認する方法 (図 1 4 (c) の S P 3 1) と、1 m S タイマ割り込み処理で確認する方法 (図 1 6 の S P 4 3) について説明したが、これらの方法に、何ら限定されるものではない。例えば、図 1 4 のステップ S P 2 6 の処理の一部として、音声プロセッサ 2 7 の初期化シーケンスが正常に終了したか否かを判定するのも好適である。

【0 2 6 6】

以上、メイン制御処理の導入部 (図 1 4 の S P 2 0 ~ S P 2 7) について説明したので、以下、図 1 7 に基づいて、メイン制御処理の本体部の動作を説明する。図 1 7 に示す通り、演出制御 C P U 6 3 の動作は、メイン制御処理 (a) と、1 m S 毎に起動するタイマ割り込み処理 (b) と、制御コマンド C M D を受けて起動する受信割り込み処理 (不図示) と、表示装置 D S 1 の V ブランク (垂直帰線期間) の開始タイミングに生じる VBLANK 信号を受けて起動する VBLANK 割り込み処理 (c) と、動作フリーズ時や不合理な指示コマンド検出時に生じる描画異常割り込み処理 (d) と、を含んで構成されている。なお、2 0 μ S 割り込み処理については説明を省略する。

30

【0 2 6 7】

受信割り込み処理では、主制御部 2 1 から受けた制御コマンド C M D を、メイン制御処理 (S T 1 3) において参照できるように、所定の受信バッファに記憶して処理を終える。また、VBLANK 割り込み処理 (図 1 7 (b)) では、VBLANK 割り込み毎に、割り込みカウンタ V C N T をインクリメントし (S T 1 5)、メイン制御処理の開始タイミングでは、割り込みカウンタ V C N T の値に基づいて、1 / 3 0 秒の動作開始タイミングを把握した上で、割り込みカウンタ V C N T をゼロクリアしている (S T 4)。

40

【0 2 6 8】

一方、タイマ割り込み処理には、図 1 7 (b) に示す通り、ランプ演出やモータ演出の進行処理 (S T 1 8) と、原点センサ信号 S N 0 ~ S N n 信号や、チャンスボタン信号などを取得するセンサ信号取得処理 (S T 1 9) とが含まれている。ランプ演出やモータ演出は、全ての演出動作を一元管理する演出シナリオに基づいて制御されており、演出カウンタ E N が管理する演出開始時に達すれば、演出シナリオ更新処理 (S T 1 1) において、モータ駆動テーブルやランプ駆動テーブルが特定されるようになっている。

【0 2 6 9】

そして、その後は、特定されたモータ駆動テーブルに基づいてモータ演出が進行し、特定

50

されたモータ駆動テーブルに基づいてランプ演出が進行することになる。先に説明した通り、ステップST18の動作時に、DMAC回路（第1と第2のDMAチャンネル）60が機能する実施例もある。なお、モータ演出は、1mS毎に進行するが、ランプ演出は、1mSより長い適宜なタイミングで進行する。

#### 【0270】

一方、図17（d）に示す通り、描画異常割込み処理では、描画回路76の動作状態を示すステータスレジスタRGijをREADアクセスして、割込み原因を特定する。具体的には、(1) 異常な指示コマンドの検出（ビット化け）による描画異常割込みか、(2) 描画回路76の動作異常（フリーズ）による描画異常割込みかを特定する（ST16a）。そして、異常な指示コマンドの検出に基づく描画異常割込みである場合には、所定のシステム制御レジスタRGijに、所定値を書き込むことで、描画回路76を初期化する（ST16b）。この動作は、図4（b）に示すリセット経路4Bの個別リセット動作に他ならない。

10

#### 【0271】

次に、個別リセット動作の正常終了を、所定のステータスレジスタRGijで確認した後、描画回路76の動作を規定する一群の動作パラメータを所定の描画レジスタRGijに再設定して処理を終える（ST16c）。そして、戻り先アドレスを記憶するスタック領域を調整した後（割込み処理後の戻り先アドレスを消去する開放処理）、ステップST13の処理に移行させる（ST16c）。

#### 【0272】

一方、描画回路76の動作異常に基づく描画異常割込みの場合には、無限ループ処理に移行させることで（ST16d）、WDT回路58を起動させ、複合チップ50全体をリセットする。なお、CPU回路51をリセットしたくない場合には、所定のキーワード列をパターンチェック回路CHKに出力して、リセット信号RSTによってVDP回路52だけをリセットしても良い（図4（b）参照）。この場合には、VDP回路52のリセット動作の正常終了を確認した後、ステップST4やST13の処理に移行させる。なお、可能な限り制御コマンドCMDの読み落しを回避するためには、他の場合も含め、ステップST4より、ステップST13に移行される方が良い。

20

#### 【0273】

複合チップ50全体をリセットすると、それまでの演出が消滅して、演出制御が完全に初期状態（電源投入状態）に戻るが、VDP回路52だけをリセットする場合には、VDP回路52のリセット動作が完了するまで、所定の待機時間は生じるものの、一連の演出制御を継続させることができる。なお、演出制御CPU63は、画像演出、ランプ演出、及び、音声演出を統一的に制御しているので、各演出に不自然なズレが生じることもない。

30

#### 【0274】

続いて、プリローダを機能しない実施例について、メイン制御処理（a）について説明する。図17（a）に示す通り、メイン制御処理は、CPUリセット後に実行される導入初期処理（ST1～ST3）と、その後、1／30秒毎に繰り返し実行される定常処理（ST4～ST14）とに区分される。なお、初期処理（ST1～ST3）は、メイン制御処理の導入部の一部であり、定常処理がメイン制御処理の本体部を意味する。

40

#### 【0275】

そして、定常処理は、割込みカウンタVCNTが、 $VCNT \geq 2$ となったタイミングで開始されるので（ST4）、定常処理の動作周期 $\delta$ は、1／30秒となる。この動作周期 $\delta$ は、演出制御CPU63の制御に基づいて間欠動作するVDP回路52について、その実質的な動作周期 $\delta$ に他ならない。なお、判定条件を、 $VCNT \geq 2$ とするのは、定常処理（ST4～ST14）が異常に長引いて、 $VCNT = 2$ のタイミングを見逃す可能性を考慮したものであるが、 $VCNT = 3$ となる事態が発生しないよう設計されている。

#### 【0276】

以上を踏まえてメイン制御処理（図17（a））の説明を続けると、本実施例では、初期処理において、記憶容量48Mバイトの内蔵VRAM71を、適切な記憶容量を有するA

50

ＣＣ領域(a)と、ページ領域(b)と、任意領域(c)と、に適宜に切り分ける（ＳＴ１）。具体的には、ＡＣＣ領域(a1,a2)と、ページ領域(b)について、各々の領域先頭アドレスと必要な総データサイズを、所定のインデックステーブルレジスタRGijに設定する（ＳＴ１）。すると、確保されたＡＣＣ領域(a1,a2)と、ページ領域(b)には含まれない残余領域が任意領域(c)となる。

【０２７７】

ここで、第一と第二のＡＣＣ領域(a1,a2)と、ページ領域(b)の領域先頭アドレスは、各々の下位１１bitが０でなくてはならないが、２０４８bit単位で任意に選択可能である（１番地＝１バイトとして、２５６番地ごとの選択）。また、総データサイズも、単位サイズの整数倍の範囲で任意に選択される。特に限定されないが、ＡＣＣ領域(a)の単位サイズは、２０４８bit、ページ領域(b)の単位サイズは、５１２kbitである。

【０２７８】

このように本実施例では、ＡＣＣ領域(a1,a2)と、ページ領域(b)の領域設定に一定の条件を設けるが、それは、メモリ容量が限られている内蔵ＶＲＡＭ７１について、可能な限り無駄領域を排除する一方で、ＶＤＰ回路５２の内部動作の円滑化を図るためである。すなわち、内蔵ＶＲＡＭ７１の記憶容量を無闇に増加させると、製造コストの高騰やチップ面積の大型化が懸念される一方、無駄領域を完全に排除するような自由な領域設定を認めると、内部処理が煩雑化して、ＶＲＡＭアクセスの処理時間を短縮化できないためである。なお、以下に説明するインデックス空間の確保に、一定の制約を設けるのも同じ理由による。

【０２７９】

以上を踏まえて説明を続けると、ステップＳＴ１の処理に続いて、ページ領域(b)と、任意領域(c)について、必要なインデックス空間IDXiを確保する（ＳＴ２）。具体的には、所定のインデックステーブルレジスタRGijに、必要な情報を設定することで、各領域(b)(c)のインデックス空間IDXiを確保する。

【０２８０】

例えば、ページ領域(b)にインデックス空間IDXiを設ける場合には、任意のインデックス番号iに対応して、任意の水平サイズHxと、任意の垂直サイズWxの倍数情報（単位空間に対する縦横の倍数情報）が、所定のインデックステーブルレジスタRGijに設定される（ＳＴ２）。

【０２８１】

先に説明した通り、ページ領域(b)のインデックス空間IDXiは、水平サイズ１２８×垂直サイズ１２８ラインを単位空間としており、また、１ピクセルは３２bitの情報で特定されるので、水平サイズHxと垂直サイズWxの設定に基づいて、データサイズ（bit長）＝３２×１２８×Hx×１２８×Wxのインデックス空間IDXiが確保されたことになる。なお、ページ領域(b)のインデックス空間IDXiの先頭アドレス（空間先頭アドレス）は、内部的に自動付与される。

【０２８２】

また、任意領域(c)にインデックス空間IDXiを設ける場合には、任意のインデックス番号iに対応して、任意の先頭アドレス（空間先頭アドレス）STxと、任意の水平サイズHxの倍数情報が、所定のインデックステーブルレジスタRGijに設定される（ＳＴ２）。ここで、任意とは、所定条件を前提とするもので、水平サイズHxは２５６bit単位で任意決定され、先頭アドレスSTxの下位１１bitは０であって、２０４８bit単位で任意決定される。先に説明した通り、任意領域の垂直サイズは、２０４８ラインに固定化されるので、水平サイズHxの設定に基づいて、先頭アドレスSTx以降には、データサイズ（bit長）＝２０４８×Hxのインデックス空間が確保されたことになる。

【０２８３】

具体的には、メイン表示装置DS1のフレームバッファFBAとして、水平サイズ１２８０×垂直ライン２０４８の一对のインデックス空間が、各々インデックス番号を特定して、一又は複数の所定のインデックステーブルレジスタRGijに設定され、サブ表示装置

10

20

30

40

50

D S 2 のフレームバッファ F B b として、水平サイズ 4 8 0 × 垂直ライン 2 0 4 8 の一対のインデックス空間が、各々インデックス番号を特定して、一又は複数の所定のインデックステーブルレジスタ R G i j に設定される。なお、もし、表示装置の水平ピクセル数が、 $256 \text{ bit} / 32 \text{ bit}$  の整数倍に一致しない場合には、各インデックス空間の水平サイズを、その表示装置の水平ピクセル数より大きく、且つ、 $256 / 32 = 8$  の整数倍となる値に設定して、無駄なメモリ領域の発生を最小限に抑制する。

#### 【0284】

以上のように、ページ領域(b) と、任意領域(c) について、必要なサイズ情報やアドレス情報を所定のインデックステーブルレジスタ R G i j に各々設定することで、必要個数のインデックス空間 I D X i が生成される (S T 2)。そして、この設定処理 (S T 2) に対応して、各インデックス空間 I D X i のアドレス情報やサイズ情報を特定するインデックステーブル I D X T B L が自動的に構築される。図 9 (a) に示す通り、インデックステーブル I D X T B L には、各インデックス空間 I D X i の先頭アドレスが、その他の必要情報と共に記憶されており、V D P 回路 5 2 内部でのデータ転送時や、外部記憶リソース (Resource) からのデータ取得時に参照される (図 10 参照)。なお、A A C 領域(a) のインデックス空間 I D X i は、必要時に自動生成され、自動消滅するので、ステップ S T 2 の設定処理は不要である。

#### 【0285】

図 9 (a) (b) に示す通り、任意領域(c) には、各一対のフレームバッファ F B a と F B b が確保され、各々、インデックス番号が付与されている。Z バッファを使用しない実施例では、フレームバッファ F B a として、インデックス番号 2 5 5, 2 5 4 が付与された、一対のインデックス空間 2 5 5, 2 5 4 が確保される。また、フレームバッファ F B b として、インデックス番号 2 5 2, 2 5 1 が付与された、一対のインデックス空間 2 5 2, 2 5 1 が確保される。なお、本実施例では、任意領域(c) に、インデックス番号 0 の作業領域 (インデックス空間 0) も確保されている。

#### 【0286】

また、本実施例では、ページ領域(a) に、I P ストリーム動画のデコード領域となる必要個数のインデックス空間 I D X i を確保し、インデックス番号 i を付与することになっている。但し、初期的には、背景動画 (I P ストリーム動画) のためのインデックス空間 I D X 0 だけを確保している。そして、画像演出 (変動演出や予告演出) における必要性に応じて、インデックステーブルレジスタ R G i j への設定処理や、ディスプレイリスト D L の指示コマンドに基づいて、ページ領域(a) のインデックス空間 I D X j を増やし、その後、不要になれば、そのインデックス空間 I D X j を開放するようにしている。すなわち、図 9 (a) は、定常動作時のインデックステーブル I D X T B L を示している。

#### 【0287】

なお、A C C 領域(a) のインデックス空間は、ディスプレイリスト D L に記載されている指示コマンドに基づいて、必要時に自動的に生成され、インデックステーブル I D X T B L には、自動生成されたインデックス空間 I D X j の先頭アドレスや、その他の必要情報が自動設定される。本実施例では、この A A C 領域(a) を、静止画その他のテクスチャのデコード領域として使用している。

#### 【0288】

インデックス空間を確保する上記の動作は、もっぱら、制御レジスタ群 7 0 に含まれるインデックステーブルレジスタ R G i j への設定動作によって実現されるが、ステップ S T 1 ~ S T 2 の処理に続いて、他の V D P レジスタ R G i j に、必要な設定動作を実行することで、図 2 5 ~ 図 2 6 に示す V D P 回路 5 2 の定常動作 (間欠動作) を可能にしている。

#### 【0289】

例えば、表示回路 7 4 の動作を規定する所定の表示レジスタ R G i j に、所定の動作パラメータ (ライン数と画素数) を書込むことで、各表示装置 D S 1, S D 2 について表示ライン数と水平画素数を設定している (S S 3 0)。その結果、各フレームバッファ F B a, F B b において、表示回路 7 4 が READ アクセスすべき有効データ領域 (図 1 7 (e))

の破線部)の縦横寸法が、特定されることになる。

#### 【0290】

次に、所定の表示レジスタRGijに、所定の動作パラメータ(アドレス値)を書込んで、各フレームバッファFBA, FBBについて、垂直表示開始位置と水平表示開始位置を特定する(SS31)。その結果、ステップSS30の処理で縦横寸法が特定された有効データ領域が、フレームバッファFBA, FBB上に確定されることになる。ここで、垂直表示開始位置と水平表示開始位置は、各インデックス空間における相対アドレス値であって、図17(e)に示す実施例では、表示開始位置は(0, 0)となっている。

#### 【0291】

続いて、メイン表示装置DS1を駆動する表示回路74Aに関する表示レジスタRGij(DSPAINDEX)と、サブ表示装置DS2を駆動する表示回路74Bに関する表示レジスタRGij(DSPBINDEX)に、各々、「表示領域(0)」と「表示領域(1)」を設定して、各表示領域を定義している(SS32)。

#### 【0292】

ここで、「表示領域」とは、表示回路74A, 74Bが、表示装置DS1, DS2を駆動するために、画像データを読み出すべきインデックス空間(フレームバッファFBA, FBB)を意味し、各々ダブルバッファ構造であるフレームバッファFBA, FBBにおけるダブルバッファの何れか一方を意味する。もっとも、表示回路74A, 74Bが、実際に画像データを読み出すのは、表示領域(0)又は表示領域(1)における、ステップSS30~SS31で特定された「有効データ領域」に限定される。

#### 【0293】

何ら限定されないが、本実施例では、フレームバッファFBAについて、VRAM任意領域(c)におけるインデックス番号254のインデックス空間254を「表示領域(0)」と定義し、VRAM任意領域(c)におけるインデックス番号255のインデックス空間255を、「表示領域(1)」と定義している(SS32)。

#### 【0294】

また、フレームバッファFBBについて、VRAM任意領域(c)におけるインデックス番号251のインデックス空間251を「表示領域(0)」とし、VRAM任意領域(c)におけるインデックス番号252のインデックス空間252を「表示領域(1)」としている(SS32)。なお、「表示領域」を初期処理(SS3)において定義することは、特に限定されず、動作周期δ毎に、表示回路74が画像データをREADアクセスすべきインデックス空間(表示領域)をトグル的に切替えても良い。

#### 【0295】

本実施例では、以上の初期処理(SS30~SS32)が終われば、次に、所定のシステム制御レジスタRGijへの設定値が、その後、ノイズなどの影響で変更されないよう、第1種の禁止設定レジスタRGijに、所定の禁止値を設定している(第1の禁止設定SS33)。

#### 【0296】

ここで、今後の書込みが禁止される設定値には、(1)表示装置DS1, DS2の表示クロックに関する設定値、(2)LVDSのサンプリングクロックに関する設定値、(3)出力選択回路79の選択動作に関する設定値、(4)複数の表示回路DS1, DS2の同期関係(表示回路74Bが表示回路74Aの動作周期に従属すること)などが含まれている。なお、第1の禁止設定を解除するソフトウェア処理は存在するが、本実施例では使用していない。但し、必要に応じて使用するのも好適である。

#### 【0297】

次に、第2種の禁止設定レジスタRGijに、所定の禁止値を設定することで、初期設定系のVDPレジスタRGijについて書込み禁止設定をしている(第2の禁止設定SS34)。ここで、禁止設定されるレジスタには、ステップSS30~SS32に係るVDPレジスタRGijが含まれている。

#### 【0298】

一方、第3種の禁止設定レジスタRGijに、所定の禁止値を設定することで、ステップST1～ST3の設定処理に関するVDPレジスタを含んだ、多数のVDPレジスタへの禁止設定も可能である（第3の禁止設定）。但し、本実施例では原則として使用しない。何れにしても、第2の禁止設定や、第3の禁止設定は、所定の解除レジスタRGijに、解除値を書込むことで任意に解除可能であり、定常動作中に設定値を変更することも可能となる。

#### 【0299】

なお、以上説明したステップST1～ST3の初期設定処理は、VDPレジスタRGijのレジスタアドレス値と、そのレジスタRGijへの設定値とを対応させた初期値設定テーブルSETTABLE（図31参照）に基づいて実行される。以上、初期設定処理について説明したので、次に、定常処理（ST4～ST14）を説明する前に、演出制御CPU63によって制御されるVDP回路52の定常動作（間欠動作）について図25（a）及び図26（b）に基づいて概略的に説明しておく。

#### 【0300】

VDP回路52の間欠動作は、図25や図26に示す通りであり、プリローダ73を使用しない実施例では、図25（a）に示すように、演出制御CPU63が完成させたディスプレイリストDLiは、その動作周期（T1）で、描画回路76に発行され、描画回路76はディスプレイリストDLiに基づく描画動作によって、フレームバッファFba、Fbbに、画像データを完成させる。そして、フレームバッファFba、Fbbに完成された画像データは、次の動作周期T1+δに、表示回路74が表示装置DS1、DS2に出力することで、その後の、表示装置DS1、DS2の描画動作に基づき、遊技者が感知する表示画面となる。

#### 【0301】

一方、プリローダ73を使用する実施例では、図26（a）に示すように、演出制御CPU63が完成させたディスプレイリストDLiは、その動作周期（T1）で、プリローダ73に発行され、プリローダ73は、ディスプレイリストDLiを解釈して、必要な先読み動作を実行すると共に、ディスプレイリストDLiの一部を書き換えて、書換えリストDL'を完成させる。なお、先読みされたCGデータと書換えリストDL'は、DRAM54の適所に格納される。

#### 【0302】

次に、描画回路76は、その次の動作周期（T1+δ）で、DRAM54から書換えリストDL'を取得し、書換えリストDL'に基づく描画動作によって、フレームバッファFba、Fbbに、画像データを完成させる。そして、フレームバッファFba、Fbbに完成された画像データは、更にその次の動作周期（T1+2δ）で、表示回路74が表示装置DS1、DS2に出力することで、その後の表示装置DS1、DS2の描画動作に基づき、遊技者が感知する表示画面となる。

#### 【0303】

以上、VDP回路52の間欠動作について概略的に説明したが、上記した図25～図26の動作を実現するため、演出制御CPU63は、初期処理（ST1～ST3）の後、割込みカウンタVCNTの値を繰り返し参照して、動作開始タイミングに達するのを待ち、動作開始タイミング（一つ飛びのVblank開始タイミング）に達すれば、割込みカウンタVCNTをゼロクリアする（ST4）。

#### 【0304】

その後、定常動作を開始するが、本実施例では、最初に、定常動作を開始すべき動作開始条件を満たしているか否かを判定する（ST5）。なお、この判定タイミングは、図25～図26に記載のT1、T1+δ、T1+2δ、・・・のタイミング、つまり、表示装置DS1の垂直帰線期間（VBLANK）の開始タイミングである。なお、表示装置DS2の表示タイミングは、表示装置DS1の表示タイミングに従属するよう、初期設定（ST3）時に設定されている。

#### 【0305】

垂直帰線期間（VBLANK）の開始タイミングで判定される動作開始条件は、プリローダ73を活用するか否かで異なるので、先ず、プリローダ73を活用しない実施例（図17）について説明する。この場合は、本来、図25（a）のタイムチャートに示す通りにVDPの内部動作が進行するよう、回路構成やプログラムが設計されている。すなわち、動作周期（ $T1$ ）で完成されたディスプレイリストDL1に基づき、描画回路76は、その動作周期中（ $T1 \sim T1 + \delta$ ）に、描画動作を終える筈である。しかし、例えば、図25（a）の動作周期（ $T1 + 2\delta$ ）で完成されたディスプレイリストDL3のように、その動作周期中（ $T1 + 2\delta \sim T1 + 3\delta$ ）に、描画動作を終わらない場合も無いとは言えない。また、表示回路74に関して、表示タイミングに対して、表示データの生成が間に合わないUnderrun異常が生じている可能性も無くはない。

10

## 【0306】

ステップST5の判定処理は、かかる事態を考慮したのであり、演出制御CPU63は、描画回路76の動作状態を示すステータスレジスタRGij（制御レジスタ群70の一種）をアクセスして、ステップST5のタイミングで、描画回路76が、必要な動作を終えているか否かと、Underrun異常の有無を判定する。なお、Underrun異常の有無は、アンダーランカウンタURCNTa～URCNTcに基づいて判定される。また、プリローダ73を活用しない実施例では、例えば、図25（a）のタイミング $T1 + \delta$ では、描画回路76に関する描画レジスタのステータス情報をREADアクセスして、ディスプレイリストDL1に基づく描画動作が終わっていることを確認する。

## 【0307】

そして、動作開始条件を満たさない場合（異常／不適合）には、異常回数をカウントする異常フラグERをインクリメントして、ステップST6～ST8処理をスキップする。異常フラグERは、その他の重大異常フラグABNと共に、ステップST9やST10の処理で判定され、重大異常フラグABNがリセット状態である前提において、連続異常回数が多い場合（ $ER \leq 2$ ）には、正常時と同様に、演出コマンド解析処理を実行する（ST13）。

20

## 【0308】

Underrun異常時の場合も、同様に、ステップST6～ST8処理をスキップする。そして、所定のシステム制御レジスタRGijに、所定のクリア値を書込むことで、表示クロック（周波数）と表示回路74を初期化する（ST10c）。そして、この初期化処理の正常終了を確認した後、表示クロックの周波数や、表示回路74の動作を規定する一群のシステム制御レジスタRGijの値を、規定値に再設定した上で（ST10c）、演出コマンド解析処理を実行する（ST13）。

30

## 【0309】

演出コマンド解析処理（ST13）では、主制御基板21から制御コマンドCMDを受けているか否かを判定し、制御コマンドCMDを受けた場合には、その制御コマンドCMDを解析して必要な処理を実行する（ST13）。ここで、必要な処理には、変動演出の開始を指示する制御コマンドCMDに基づく新規の変動演出の開始準備処理や、エラー発生を示す制御コマンドCMDに基づくエラー報知の開始処理が含まれる。続いて、WDT回路にクリアパルスを出力して（ST14）、ステップST4の処理に戻る。

40

## 【0310】

以上、軽微なUnderrun異常時や、動作開始条件が不適合の場合であって、異常フラグERが $ER \leq 2$ である場合について説明したが、このような場合には、その動作周期では、表示回路74が読み出す表示領域をトグル切替える処理（ST6）や、ディスプレイリストの作成処理（ST7）がスキップされ、且つ、演出シナリオが進行しないことになる（ST8～ST12参照）。これは、不完全な状態のフレームバッファFBa、FBbの画像データを出力させないためである。そのため、例えば、図25（a）の動作周期（ $T1 + 3\delta$ ）では、画像演出が進行せず、元の画面（DL2に基づく画面）が再表示されるフレーム落ちが生じる。

## 【0311】

50

ここで、フレーム落ちを回避するため、動作開始条件が成立するまで待機する構成も考えられる。しかし、演出制御CPU63が実行すべき制御処理(ST6~ST12)は数多く、各々の処理時間を確保する必要があるので、本実施例では、動作開始条件を満たさない場合にフレーム落ちを生じさせている。

#### 【0312】

但し、フレーム落ちが生じたとしても、割込み処理(図17(b))によって進行するランプ演出やモータ演出と比較して、1/30~2/30秒程度、画像演出の進行が遅れるだけであり、これに遊技者が気付くことはない。しかも、フレーム落ち時には、演出カウンタENの更新処理を含んだ演出シナリオ処理(ST11)や、音声進行処理(ST12)も合わせてスキップされるので、その後を開始されるリーチ演出や予告演出や役物演出において、画像演出、音声演出、ランプ演出、及びモータ演出などの開始タイミングがずれるおそれはない。

10

#### 【0313】

すなわち、演出シナリオでは、画像演出、音声演出、ランプ演出、モータ演出の開始タイミングと、その後実行すべき演出内容を一元的に管理しており、正常時に限り更新される演出カウンタENによって、開始タイミングを制御しているので、各種の演出の同期が外れることはない。例えば、爆発音と、爆発画像と、役物移動と、ランプフラッシュ動作を複合した演出動作がある場合、フレーム落ちが生じた後であっても、上記した各演出動作は正しく同期して開始される。

#### 【0314】

以上、比較的軽微な異常時について説明したが、重大異常フラグABNがセット状態である場合や、連続異常回数が多い場合(ER>2)や、繰り返しUnderrun異常が生じる場合には、ステップST10の判定の後、無限ループ状態としている(ST10b)。その結果、WDT回路58の計時動作が進行して、演出制御CPU63を含んだ複合チップ50は、異常リセットされ、その後、初期処理(ST1~ST3)が再実行されることで、異常事態発生の根本原因の解消が期待される。

20

#### 【0315】

なお、このリセット動作は、WDT回路58が起動して実行されるので、CPU回路51も含め複合チップ50全体がリセット状態となる(図4(b))。そこで、CPU回路51のリセットを回避するべく、演出制御CPU63が、所定のキーワード列(例えば1バイトデータ3個)をパターンチェック回路CHKに出力して、リセット信号RSTをVDP回路52に出力するのも好適である(図31のST100参照)。この場合も、VDP回路52のリセット動作の正常終了を確認した後(ST101)、ステップST4やST13の処理に移行させることになる。

30

#### 【0316】

何れにしても、この異常時には、音声回路SNDも合わせ異常リセットされるので、画像演出、音声演出、ランプ演出、モータ演出は、全て初期状態に戻るようになる。但し、これらのリセット動作は、主制御部21や払出制御部25には、何の影響も与えななので、大当り状態の消滅や、賞球の消滅のような事態が発生するおそれはない。

#### 【0317】

以上、異常事態について説明したが、実際には、軽微な場合も含め上記した異常が発生することは殆どなく、ステップST5の処理の後、所定の表示レジスタRGij(DSPACTL/DSPBCTL)への設定に基づき、表示回路74Aと表示回路74Bが読み出すべき画像データを記憶するフレームバッファFba, Fbbの「表示領域」をトグル的に切り換える(ST6)。先に説明した通り、「表示領域(0)」と「表示領域(1)」は、予め初期処理において定義されているので(ST3)、ステップST6の処理では、フレームバッファFba, Fbbについて、今回の「表示領域」が、表示領域(0)/表示領域(1)の何れであるかを特定する。

40

#### 【0318】

このステップST6が実行されることで、表示回路74Aは、インデックス空間254(

50

表示領域（０））と、インデックス空間２５５（表示領域（１））から、動作周期δ毎に、交互に画像データを読み出して表示装置ＤＳ１を駆動することになる。同様に、表示回路７４Ｂは、インデックス空間２５１（表示領域（０））と、インデックス空間２５２（表示領域（１））から、動作周期δ毎に、交互に画像データを読み出してサブ表示装置ＤＳ２を駆動することになる。なお、表示回路７４が実際にREADアクセスするのは、表示領域（０）／表示領域（１）における有効データ領域に限定されるのは先に説明した通りである。

#### 【０３１９】

何れにしても、本実施例では、動作周期毎に「表示領域」が切り替わるので、表示回路７４Ａ，７４Ｂは、直前の動作周期で描画回路７６が完成させた画像データについて、表示装置ＤＳ１，ＤＳ２への出力処理を開始することになる。但し、ステップＳＴ５の処理は、メイン表示装置ＤＳ１の垂直帰線期間（Ｖブランク）の開始時から開始されるので、実際には、垂直帰線期間が完了してから画像データの出力処理が開始されることになる。図２５（ａ）において、表示回路の欄に示す矢印は、この出力処理の動作周期を示している。

#### 【０３２０】

以上のような意義を有するステップＳＴ６の処理が終われば、演出制御ＣＰＵ６３は、続いて、次の動作周期で、表示回路７４が表示装置に出力すべき画像データを特定したディスプレイリストＤＬを完成させる（ＳＴ７）。特に限定されないが、この実施例では、ＲＡＭ５９のリストバッファ領域（ＤＬバッファＢＵＦ）を確保し、そこにディスプレイリストＤＬを完成させている（図１０参照）。

#### 【０３２１】

ディスプレイリストＤＬは、一連の指示コマンドを、適宜な順番で列記して構成され、ＥＯＤＬ（End Of DL）コマンドを記載して終わるよう構成されている。そして、本実施例では、データ転送回路７２、描画回路７６、プリローダ７３の円滑な動作を実現するべく、ＥＯＤＬコマンドを含む全ての指示コマンドを、コマンド長が３２ｂｉｔの整数Ｎ倍（ $N > 0$ ）の指示コマンドだけに限定している。なお、３２ｂｉｔの整数Ｎ倍で構成された指示コマンドに、無意ビット（Don't care bit）も含んで良いことは先に説明した通りである。

#### 【０３２２】

このように、実施例のディスプレイリストＤＬは、コマンド長が３２ｂｉｔの整数Ｎ倍（ $N > 0$ ）の指示コマンドだけで構成されているので、ディスプレイリストＤＬ全体のデータボリューム値（データ総量）は、必ず、コマンド長の最小単位（３２ｂｉｔ＝４バイト）の整数倍となる。更に、本実施例では、データ転送回路７２の最低データ量 $D_{min}$ を考慮して、ディスプレイリストＤＬのデータボリューム値を、最低データ量 $D_{min}$ の整数倍（１以上）であって、且つ、指示コマンドの最小単位（４バイト）の整数倍となるよう調整している。例えば、 $D_{min} = 256$ バイトであれば、ディスプレイリストＤＬのデータボリューム値は、２５６バイト、５１２バイト・・・の何れかの値に調整される。

#### 【０３２３】

ここで、演出内容の複雑さに応じて、適宜に、２５６バイトか、又は５１２バイトに調整するのも好適であるが、本実施例では、表示装置が二個であり、サブ表示装置ＤＳ２はそれほど複雑な画像演出を実行させないことを考慮して、ディスプレイリストＤＬのデータボリューム値を、常に、２５６バイトに調整している。

#### 【０３２４】

もっとも、この手法は、何ら限定されず、表示装置が三個以上になる場合や、サブ表示装置ＤＳ２も含め複雑な画像演出を実行する遊技機の場合には、５１２バイト又は、７６８バイトに調整される。また、通常の演出時は、ディスプレイリストＤＬのデータボリューム値を２５６バイトに調整し、特別な演出を実行する場合に限り、ディスプレイリストＤＬのデータボリューム値を、５１２バイト又は、７６８バイトに調整するのも好適である。

#### 【０３２５】

但し、本実施例の場合には、ディスプレイリストＤＬのデータボリューム値は、各動作周期δにおいて、予め規定された所定バイト長（２５６バイト）に調整される。調整手法と

10

20

30

40

50

しては、32bit長のEODLコマンドの後に、不足領域を補填する32bit長のNOP (No Operation) コマンドを埋める簡易手法 (A) か、或いは、不足領域を32bit長のNOP コマンドで埋めた後、最後に32bit長のEODLコマンドを記載する標準手法 (B) が考えられる。なお、ディスプレイリストDLのデータボリューム値 (データ総量) を全く調整することなくEODLコマンドで終結させ、データ転送回路72の動作時に、ダミーデータを付加的に転送して、最低データ量Dminの整数倍の転送量を確保する無調整手法 (C) も考えられる。

#### 【0326】

ここで、標準手法 (B) を採る場合には、最初、コマンドカウンタCNTを規定値 (256バイトに対応する64-1) に初期設定し、DLバッファ領域BUFに、有意な指示コマンドを書き込むごとに、コマンドカウンタCNTを適宜に減算し、一連の有意な指示コマンドの書き込みが終われば、コマンドカウンタCNTがゼロになるまで、NOP コマンドを記載し、最後にEODLコマンドを記載する手法が考えられる。本実施例の場合、指示コマンドは、そのコマンド長が32bitの整数N倍 (N>0) のものに限定されているので、上記の処理は容易であり、コマンドカウンタCNTの減算処理は、整数Nに対応した減算処理となる。

#### 【0327】

一方、簡易手法 (A) を採る場合には、ディスプレイリストDLの作成時、最初に、リストバッファ領域 (DLバッファBUF) の全てをNOP コマンドで埋めれば足りるので、一見、標準手法 (B) より優れているように思われる。また、簡易性の観点では、無調整手法 (C) も優れているように思われる。しかし、本実施例では、基本的に標準手法 (B) を採っており、ディスプレイリストDLの先頭からEODLコマンドまでの実データ量、つまり、EODLコマンドまでのデータ量が、常に、データ転送回路72の最低データ量Dminの整数倍となるよう調整している。

#### 【0328】

これは、プリローダ73を活用する実施例を考慮したものであり、もし、簡易手法 (A) や無調整手法 (C) を採用すると、EODLコマンドまでのディスプレイリストDLの実データ量が、ランダムな値となり、プリローダ73が書き換えた書換えリストDL'のDRAM54への転送時や、DRAM54から描画回路76への書換えリストDL'の転送時に支障が生じるからである。なお、書換えリストDL'のDRAM54への転送時には、データ転送回路72のChA制御回路72aが機能し、書換えリストDL'の描画回路76への転送時には、ChB制御回路72bが機能するが (図23参照)、何れの場合もEODLコマンドまでの書換えリストDL'しか転送しないことになる。

#### 【0329】

以上、ディスプレイリストDLのデータボリューム値を調整する標準手法 (B) の利点を説明したが、プリローダ73を使用しない実施例では、発行されたディスプレイリストDLは、描画回路76によって処理されるだけであるので、簡易手法 (A) や無調整手法 (C) の使用が何ら禁止されない。

#### 【0330】

但し、以下の説明では、プリローダ73の使用の有無に拘らず、原則として標準手法 (B) を採ることを前提に、図18に基づいて、ディスプレイリストDLの詳細について説明する。

#### 【0331】

特に限定されないが、本実施例では、ディスプレイリストDLに、先ず、メイン表示装置DS1に関する指示コマンド列 (L11~L16) を記載し、その後、サブ表示装置DS2に関する指示コマンド列 (L17~L20) を記載するようにしている。また、標準手法 (B) を採用して、ディスプレイリストDLのデータボリューム値を固定長 (256バイト) に調整している。なお、図18は、事実上、演出制御CPU63が、RAM59のリストバッファ領域に、指示コマンドを書き込む手順や、ディスプレイリストDLに基づく描画回路76の動作を示したものともなっている。

10

20

30

40

50

## 【0332】

図18に示す通り、ディスプレイリストDLの先頭では、環境設定系の指示コマンド（SETDAVR）を記載して、表示装置DS1のフレームバッファFBAについて、インデックス空間IDX上の左上基点アドレス（X，Y）を規定する（L11）。図9（a）に関して説明した通り、本実施例では、表示装置DS1用として、任意領域(c)に、一対のフレームバッファFBAが確保されている。そして、通常は、表示回路74にとっての有効データ領域に対応して、基点アドレス（X，Y）＝（0，0）とすることで、フレームバッファFBAの先頭位置から描画回路76に活用される。

## 【0333】

図9（c）では、その下方左側の実描画領域にL11と付しているが、これは、指示コマンドL11によって、フレームバッファFBA上の実描画領域が、フレームバッファFBAの基点アドレス（0，0）位置から始まると特定されたことを意味している。ただし、実描画領域の縦横寸法や、その実描画領域を具体的に特定するインデックス番号は、未だ未確定であり、後述する指示コマンド（SETINDEX）L13によって確定する。なお、指示コマンドL11ではZバッファの使用の有無も指定される。

10

## 【0334】

次に、環境設定系の指示コマンド（SETDAVF）によって、仮想描画空間上に、左上基点座標（Xs，Ys）と、右下対角点座標（Xe，Ye）を設定して、W×H寸法の描画領域を定義する（L12）。ここで、仮想描画空間とは、描画用の指示コマンド（SPRITEコマンドなど）によって描画可能な、X方向±8192、Y方向±8192の仮想的な二次元空間である（図9（c）参照）。

20

## 【0335】

この指示コマンドL12（SETDAVF）によって、仮想描画空間は、描画内容が実際に表示装置DS1に反映される描画領域と、その他の非描画領域に区分される。また、指示コマンドL12（SETDAVF）は、指示コマンドL11で開始位置（基点アドレス）が規定された実描画領域と、仮想描画空間上の描画領域とを対応付けることになる。

## 【0336】

この点を言い換えると、指示コマンドL12によって、（インデックス空間は未定の）フレームバッファFBAには、仮想描画空間上の描画領域に対応する、基点アドレスから始まるW×Hの実描画領域が定義されることになる。したがって、指示コマンドL12で指定する描画領域は、フレームバッファFBAの水平サイズと同一か、それ以下とする必要がある。通常、描画領域や実描画領域は、表示回路74にとっての有効データ領域（図17（e））と同寸法となるよう定義される。

30

## 【0337】

そして、描画回路76が指示コマンドL11，L12を実行した後は、仮想描画空間に描画された描画内容のうち、描画領域に含まれるものだけが、フレームバッファFBAの実描画領域に反映されることになる。したがって、描画領域からはみ出した部分や、図9（c）において作業領域と記載された部分の描画内容は、そのままでは、フレームバッファに反映されることはない。なお、仮想描画空間に作業領域を確保する場合には、仮想描画空間の非描画領域が使用される。

40

## 【0338】

次に、今回の動作周期において、描画回路76が、これから完成させるディスプレイリストDLに基づいて描画する描画内容を何処に描画すべきかを規定する（L13）。具体的には、ダブルバッファ構成の表示装置DS1のフレームバッファFBAについて、今回のディスプレイリストDLに基づく描画内容の「書込み領域」となるインデックス空間IDXが特定される（L13）。具体的には、テクスチャ設定系のコマンドであるSETINDEXコマンドによって、(1) フレームバッファFBAは、任意領域に確保されていること、及び、(2) 「書込み領域」となるインデックス空間IDXNの任意領域上のインデックス番号Nが特定される。

## 【0339】

50

この指示コマンド L 1 3 によって、例えば、N = 2 5 5 と特定された場合には、仮想描画空間上に定義された描画領域に対応する実描画領域は、具体的には、ダブルバッファ構造のフレームバッファ F B a におけるインデックス空間 I D X 2 5 5 であると定義されたことになる。

【0340】

本実施例の場合、フレームバッファ F B a のインデックス番号は、2 5 5 又は 2 5 4 であり（図 9（a））、トグル的に切り換えた何れかが指定される（L 1 3）。なお、このインデックス番号は、メイン制御処理のステップ S T 6 で指定された表示領域（0）／（1）ではない方のインデック番号である。例えば、ステップ S T 6 の処理において、表示回路 7 4 に対して、表示領域（0）が指定されている場合には、表示領域（1）が、描画回路 7 6 にとっての「書込み領域」となる。

10

【0341】

以上の通り、指示コマンド L 1 1 と指示コマンド L 1 2 とで、実描画領域（W×H の論理空間）と描画領域（W×H の仮想空間）との対応関係が、一般的に定義された後、インデックス空間 I D X を具体的に特定する指示コマンド L 1 3（SETINDEX）によって、W×H の仮想空間が、特定のインデックス空間 I D X における W×H の論理空間であるに対応付けられたことになる。

【0342】

この点を言い換えると、今後、一連の指示コマンドに基づいて、W×H の仮想空間に仮想的に描画される内容は、仮想空間と内蔵 V R A M 7 1 の実アドレスとの対応関係を規定する V D P 内部の変換テーブルに基づいて、内蔵 V R A M 7 1（フレームバッファ）の画像データとなる。

20

【0343】

続いて、「書込み領域」として、特定されたインデックス空間 I D X を、例えば、黒色で塗りつぶすフレームバッファ・クリア処理を実行する指示コマンドが記載される（L 1 4，L 1 5）。これは、二動作期間前にフレームバッファ F B a に書き込まれた画像データの消去処理に他ならない。

【0344】

具体的には、環境設定コマンドの一種である SETFCOLOR コマンドによって、例えば黒色を選択し、プリミティブ描画系コマンドである RECTANGLE コマンドによって矩形領域を塗り潰すべく規定する。なお、RECTANGLE コマンドでは、仮想描画空間に設定された描画領域（フレームバッファ F B a に対応する仮想空間）について、その左上端点と、右下端点の X Y 座標が指定される（図 9（c）参照）。

30

【0345】

以上の処理によって、描画準備処理が完了するので、次に、静止画や動画一フレームなど、適宜なテクスチャを、仮想描画空間に描画するための指示コマンドを列記する。典型的には、先ず、テクスチャの展開先となるインデックス空間 I D X を、テクスチャ設定系の SETINDEX コマンドで特定した上で、テクスチャロード系の指示コマンドである TXLOAD コマンドを記載して、C G R O M 5 5 から読み出す所定のテクスチャを、所定のインデックス空間 I D X に展開するようディスプレイリスト D L に記載する。

40

【0346】

先に説明した通り、本実施例では、背景動画が、I P ストリーム動画で構成されている。そこで、例えば、背景動画について、これを展開すべきインデックス空間 I D X を、テクスチャ設定系の SETINDEX コマンドで、ページ領域 (b) のインデックス空間 I D X 0 と特定した上で、テクスチャロード系の TXLOAD コマンドを記載する。なお、TXLOAD コマンドでは、今回 LOAD すべき動画フレームについて、C G R O M 5 5 の先頭アドレス（テクスチャの Source アドレス）と、展開後のデータサイズ（水平×垂直）を特定する必要がある。

【0347】

V D P 回路 5 2 において、上記の TXLOAD コマンドが実行されると、背景動画の一動画フ

50

レーム（テクスチャ）は、まず、A A C 領域(a) に取得され、その後、自動的に起動する G D E C 7 5 によって、ページ領域(b) のインデックス空間 I D X 0 に展開される。次に、この一動画フレームを仮想描画空間に描画することになる。この場合に、SETINDEX コマンド（テクスチャ設定系）によって、「ページ領域(b) のインデックス空間 I D X 0 が、その後の処理対象のテクスチャである」と設定しても良いが、TXLOAD コマンドに連続して処理する場合には、このSETINDEX コマンドの記載を省略することができる。

【0348】

何れにしても、「ページ領域(b) のインデックス空間 I D X 0 が、その後の処理対象のテクスチャである」と特定されている状態で、次に、 $\alpha$  ブレンド処理のためのパラメータを設定するなど、適宜な描画間演算系の指示コマンドを記載する。なお、 $\alpha$  ブレンド処理とは、既に描画領域（フレームバッファ F B a）に記載されている画像と、これから上書きする画像との透明化／半透明化処理に関するものある。したがって、背景動画の動画フレームのように、第一枚目の描画動作では、描画間演算系の指示コマンドの使用は不要である。

【0349】

続いて、プリミティブ描画系の指示コマンドであるSPRITE コマンドによって、「ページ領域(b) のインデックス空間 I D X 0 のテクスチャ（背景動画の一動画フレーム）」を、仮想描画空間の適所（矩形のDestination 領域）に描画するべくSPRITE コマンドを記載する。なお、SPRITE コマンドには、仮想描画空間のDestination 領域について、その左上端点と、右下端点を特定する必要がある。

【0350】

このDestination 領域は、予め、指示コマンド L 1 1, L 1 2 によって、実描画領域（F B a）に対応付けられた描画領域（仮想描画空間上に定義された仮想空間）の全体又はその一部である。但し、背景動画は、通常、表示画面全体に描画するので、このような場合のDestination 領域は、描画領域の全体又はそれ以上となる。なお、Destination 領域が、描画領域の全体より大きい場合とは、例えば、背景動画がズームアップされる場合である。

【0351】

以上の処理によって、背景動画の動画フレームの描画が終わったので、続いて、テクスチャロード系、テクスチャ設定系、描画間演算系、プリミティブ描画系コマンドなどの指示コマンドを適宜な順番で列記して、背景動画に重ねて、各種のテクスチャを描画するべくディスプレイリスト D L を構成することになる。先に説明したように、変動演出時では、多数の動画が必要となるので、その場合には、内蔵 V R A M 7 1 のページ領域(b) について、インデックス空間 I D X を増加するべく、インデックステーブル制御系の指示コマンド（NEWPIX）を記載することになる。

【0352】

例えば、二つ目の I P ストリーム動画に関し、NEWPIX コマンドによって、ページ領域(b) に、追加のインデックス空間 I D X 1 を確保した後、このインデックス空間 I D X 1 を特定して（SETINDEX）、二つ目の動画の一フレームの展開を指示し（TXLOAD）、展開したテクスチャを描画領域の適所に配置する（SPRITE）。通常、この場合のDestination 領域は、描画領域の一部となる。

【0353】

以下、同様であり、NEWPIX コマンドによって、次々、インデックス空間 I D X k を確保した後、適宜な $\alpha$  ブレンド処理を実行しつつ、複数の I P ストリームを描画領域に描画すれば、描画領域への描画内容は、実描画領域であるフレームバッファ F B a に画像データとして順次蓄積されることになる。複数 N 個の I P ストリーム動画が描画されている演出時には、ページ領域(b) において、複数 N 個のインデックス空間が機能している。

【0354】

そして、一連の変動演出が終了したような場合には、ページ領域(b) に確保した多数のインデックス空間 I D X 1 ~ I D X k のうち、不要と思われるインデックス空間 I D X を開

10

20

30

40

50

放するべく、DELPPIXコマンドによって不要なインデックス空間 I D X を削除すれば良い。

【0355】

なお、静止画やストリーム動画を描画する場合には、SETINDEXコマンドによって、これらのテクスチャのデコード先が、A A C 領域(a) であると指定した上で、TXLOADコマンドを実行させれば、A A C 領域(a) に取得されたテクスチャは、その後、自動的に起動するG D E C 7 5によってA C C 領域(a) に展開される。そして、展開されたテクスチャは、SPRITEコマンドによって、描画領域の適所に描画すれば良い。なお、キャッシュヒット機能を活用するか否かに応じて、第一A A C 領域(a1)か、第二A A C 領域(a2)が使用される。

【0356】

ここまでの説明では、各テクスチャは、直接的に、メイン表示装置用D S 1の描画領域に描画されるが、必ずしも、このような動作に限定されない。例えば、既に表示装置D S 1用に確保されている描画領域に重複しない状態で、適宜な描画領域を設け（図9（c））、この描画領域を内蔵V R A M 7 1の作業領域に対応付ければ、中間的な描画領域を構築して、適宜な演出画像を完成させることができる。ここで、表示装置D S 1用の描画領域と重複しない状態とするのは、重複領域については、後の対応付け設定が優先され、その領域への描画内容がフレームバッファF B aに反映されないからである。

【0357】

図9（c）に示す通り、本実施例の作業領域は、任意領域(c)におけるインデックス空間 I D X 0である。そして、この作業領域を使用する演出タイミングでは、先行して、演出画像用の描画領域（図9（c）参照）を、作業領域（インデックス空間 I D X 0の実描画領域）に対応付けるための指示コマンド列（SETDAVR，SETDAVF，SETINDEX）を記載しておく。図9（c）に示す通り、演出画像用の描画領域は、メイン表示装置D S 1用の描画領域に含まれない領域に確保される。

【0358】

そして、その後は、フレームバッファF B aに関する指示コマンド列L 1 6と同様の指示コマンドを列記して、インデックス空間 I D X 0に、適宜な演出画像を完成させれば良い。本実施例の場合、演出画像は、静止画で構成されるので、デコードデータは第一A A C 領域(a1)に展開されるよう指示コマンド（SETINDEX）が記載され、次に、インデックス空間 I D X 0の描画領域の適所をDestination とするプリミティブ描画系の指示コマンド（SPRITE）が使用されることになる。なお、このような動作は、演出内容に応じて、一回又は複数回繰り返される。

【0359】

そして、演出画像を完成させたインデックス空間 I D X 0をテクスチャと位置付けた後（SETINDEX）、SPRITEコマンドによって、メイン表示装置用D S 1の描画領域の適所に、インデックス空間 I D X 0の演出画像（テクスチャ）を描画すれば良い。このような場合、インデックス空間 I D X 0の演出画像を、三角形の描画プリミティブ（primitive）に分解し、適宜な角度に回転させた上で、描画領域に描画することが考えられる。なお、テクスチャの回転角度は、例えば、予告演出の信頼度などに対応付けられる。

【0360】

以上、メイン表示装置D S 1の一フレームを完成させるための指示コマンド列（L 1 1～L 1 6）について説明したが、サブ表示装置D S 2の一フレームを完成させるための指示コマンド列（L 1 7～L 1 2）についても、同様である。すなわち、フレームバッファF B bの開始X Y座標を特定し（L 1 7）を定義し（通常はX = 0，Y = 0）、図9（c）に示す仮想描画空間上に、サブ表示装置D S 2のための描画領域を定義する（L 1 8）。

【0361】

ところで、本実施例では、メイン表示装置D S 1用の画像データの生成を終えた後、サブ表示装置D S 2用の生成処理に移行するので、サブ表示装置D S 2用の描画領域が、メイン表示装置D S 1用の描画領域と重複しても何の問題もなく、描画領域を自由に設定することができる。そのため、ディスプレイリストD Lの生成プログラムの開発時、例えば、S

10

20

30

40

50

PRITEコマンドで、新規に設定された描画領域に適宜なテクスチャを貼り付けるような場合、SPRITEコマンドの動作パラメータ（Destination 領域）の設定その他を、ある程度、定型化することができる。

#### 【0362】

このような任意の描画領域の定義が終われば（L18）、次に、ダブルバッファ構成の表示装置DS2のフレームバッファFBbについて、今回のディスプレイリストDLに基づく描画内容の「書込み領域」となるインデックス空間IDXを特定する（L19）。このインデックス空間IDXのインデックス番号は、フレームバッファFBbに関し、メイン制御処理のステップST6で指定された表示領域（0）／（1）に対応しない方のインデックス番号である。

10

#### 【0363】

そして、その後、サブ表示装置DS2についての指示コマンド列L20～L22が、メイン表示装置DS1に関する指示コマンド列L14～L16と同様に列記される。また、インデックス空間IDX0に完成させた演出画像を使用することもできる。

#### 【0364】

以上、ディスプレイリストDLを構成するL11～L22の指示コマンドは、本実施例では、全て、コマンド長が32ビットの整数倍のものに限定されている。そして、先に説明した通り、本実施例のディスプレイリストDLのデータボリューム値（データ総量）を、固定長（256バイト）に調整しており、ダミーコマンドたる必要数のNOP コマンド（L23）を付加した上で、EODLコマンド（L24）で終結させている。すなわち、図18の実施例では、前記した標準手法（B）を採っている。

20

#### 【0365】

但し、標準手法（B）を採る場合でも、全ての動作周期において、ディスプレイリストDLのデータ総量を256バイトと固定化することは必ずしも必須ではない。すなわち、別の実施例では、NOP コマンドを除くディスプレイリストDLのデータ総量が、256バイトを超える場合（例えば、特別な演出期間）には、ディスプレイリストDLのデータ総量は、NOP コマンドを付加することで、512バイト又はそれ以上の $N \times 256$ バイトに調整される。なお、標準手法（B）を採る場合、 $N \times 256$ バイトの最後はEODLコマンドで終端されることは先に説明した通りである。

#### 【0366】

以上、ディスプレイリストDLの構成について詳細に説明したが、演出制御CPU63は、完成させた固定バイト長のディスプレイリストDLをVDP回路に発行することになる（ST7～ST8）。図19は、演出制御CPU63が、転送回路72の転送ポートレジスタTR\_PORTを直接WRITE アクセスして、描画回路76にディスプレイリストDLを発行するDL発行処理（図17のST8）を説明するフローチャートである。なお、転送ポートレジスタTR\_PORTは、データ転送回路72の動作内容を規定するデータ転送レジスタRGi jの一種である。

30

#### 【0367】

DL発行処理を実現するには、まず、データ転送回路72の動作内容を規定する複数のデータ転送レジスタRGi jに、必要な設定値を設定する必要がある。具体的には、データ転送回路72の転送動作態様と、データ転送回路72内部の伝送経由と、を所定のデータ転送レジスタRGi jに特定する。設定内容は、特に限定されないが、ここでは、CPU IF部56からChB制御回路72bを経由すること、及び、CPUバス制御部72dに関し、そのFIFOバッファの残量をチェックしながらデータ転送動作を実行すると設定する（ST20）。なお、以下の説明では、ChB制御回路72bを、便宜上、「転送回路ChB」と略すことがある。

40

#### 【0368】

次に、転送総サイズを、所定のデータ転送レジスタRGi jに設定する。先に説明した通り、本実施例では、ディスプレイリストDLのデータ総量を256バイトの整数倍に調整しているので、その値を設定する。なお、データ総量 $= 256 \times N$ は、データ転送回路7

50

2の最低データ量Dminの整数N倍にもなっている。通常、倍数Nは、1又は2であるが、以下の説明では、N=1として説明することにする。

#### 【0369】

ここで、転送ポートレジスタTR\_PORT（以下、転送ポートと略すことがある）は、32bit長のレジスタであるので、演出制御CPU63は、32bit毎に、転送ポートTR\_PORTに対して、レジスタWRITE動作を実行することになる。そこで、レジスタWRITE回数を管理する管理カウンタCNの値を64の初期設定する（ST21）。なお、無調整手法（C）を採る場合は、このタイミングで、最低データ量Dminの整数倍のデータ転送量を決定して、管理カウンタCNを設定することになる。

#### 【0370】

以上の処理で初期設定が完了するので、次に、転送回路ChBを経由するデータ転送動作を開始状態に設定すると共に（ST22）、描画回路76の動作を規定する所定の描画レジスタRGijへの設定値に基づいて、描画動作を開始させる（ST23）。この結果、その後、演出制御CPU63が、転送ポートTR\_PORTにレジスタWRITE動作する指示コマンド列について、描画回路76（ディスプレイリストアナライザ）による迅速かつ円滑なAnalyze処理が担保される。

#### 【0371】

なお、迅速かつ円滑なAnalyze処理には、ディスプレイリストDLに列記する指示コマンドが、コマンド長32bit整数倍の指示コマンドに限定されている点も有効に寄与する。図25（a）におけるタイミングt1、t2、t3、t4は、ステップST23の動作タイミングを示している。なお、ディスプレイリストDLの発行処理（ST8）は、素早く終わるので、図25～図26では発行処理の要する時間幅を記載していない。

#### 【0372】

続いて、ステップST22の設定が機能したか否かを確認する（ST24）。これは、データ転送回路72の各部の初期設定は、演出制御CPU63によるレジスタWRITE動作（設定動作）より処理時間がかかるので、不完全な状態のデータ転送回路72に対して、その後の指示を与えないためである。そして、万一、所定時間、待機しても動作開始状態にならない場合には、重大異常フラグABNをセットしてDL発行処理を終える（ST25）。その結果、その後、WDT回路58が機能して、複合チップ50は異常リセットされる（ST10）。

#### 【0373】

なお、演出制御CPU63は、CPU回路51のリセットを回避するべく、所定のキーワード列をパターンチェック回路CHKに出力して、リセット信号RSTに基づいてVDP回路52だけを異常リセットしても良いのは前述した通りである。

#### 【0374】

但し、通常は、ステップST22の設定は、迅速に完了するので、続いて、CPUバス制御部72dのFIFOバッファ（32bit×130段）について、FIFOバッファが満杯でないことを確認した上で（ST26）、ディスプレイリストDLを構成する先頭行から順番に、一行ごとに転送ポートTR\_PORTに指示コマンドを書込む（ST28）。

#### 【0375】

そして、管理カウンタCNをデクリメントしつつ（ST29）、管理カウンタCNがゼロになるまで、ステップST26～ST29の処理を繰り返す（ST30）。この実施例の場合、データ転送回路72には、最低データ量Dminが規定されているので、FIFOバッファに最低データ量Dminが蓄積されたタイミングで、データ転送動作が実行されることになり、間欠的な転送動作となる。

#### 【0376】

何れにしても、本実施例では、迅速にDL発行処理（ST28）が完了するが、万一、ノイズなどの影響でVDPレジスタRGijへの設定内容が矛盾したような場合には、ステップST26の判定において、所定時間待機してもFIFOバッファFullの状態が解消されない場合もあり得る。そして、そのような場合には、所定のVDPレジスタRGijに

10

20

30

40

50

初期化データをセットして、描画回路 7 6 とデータ転送回路 7 2 を初期化した上で、重大異常フラグ A B N をセットして D L 発行処理を終える ( S T 2 7 ) 。

【 0 3 7 7 】

ところで、このタイミングでは、データ転送回路 7 2 や、描画回路 7 6 は、既に動作を開始しており、ある程度の処理を終えているので、描画回路 7 6 の初期化処理には、描画レジスタ R G i j の内容を維持した状態で、(1) ディスプレイリスト D L によって設定される可能性のある全ての内部パラメータを初期値に設定すること、(2) 全ての内部制御回路を初期状態に設定すること、(3) G D E C 7 5 を初期化すること、(4) A A C 領域のキャッシュ状態を初期化することが含まれている。同様に、データ転送回路 7 2 の初期化処理には、F I F O バッファのクリアなど、それまでのデータ転送全体の初期化処理が含まれている。この結果、データ転送回路 7 2 の動作状態を示すステータス情報が所定値 ( データ転送全体初期化中を示す値 ) に変化する。

10

【 0 3 7 8 】

なお、上記したステップ S T 2 7 の初期化処理では、描画レジスタ R G i j の内容を維持したが、所定の描画レジスタについては、その内容を初期化しても良い。初期値にクリアされる所定の描画レジスタには、(a) 描画実行開始を設定する実行制御レジスタ ( 図 1 9 の S T 2 3 参照 )、(b) 描画回路 7 6 の実行状況を示すステータスレジスタ、及び、(c) 現在処理しているディスプレイリストの位置を特定するステータスレジスタが含まれる。

【 0 3 7 9 】

何れにしても、重大異常フラグ A B N をセットした結果、その後、W D T 回路 5 8 や演出制御 C P U 6 3 が機能して、複合チップ 5 0 か、又は V D P 回路 5 2 が異常リセットされるので ( S T 1 0 a )、描画回路 7 6 やデータ転送回路 7 2 を初期化する処理は必ずしも必須ではない。一方、描画回路 7 6 やデータ転送回路 7 2 を初期化する場合には、その結果、異常回復が期待できるので、重大異常フラグ A B N をセットすることなく、ステップ S T 2 0 の処理に戻って D L 発行処理を再実行するのも好適である。

20

【 0 3 8 0 】

この点は、ステップ S T 2 5 の処理においても同様であり、データ転送回路 7 2 や描画回路 7 6 を初期化した上で、重大異常フラグ A B N をセットすることなく、ステップ S T 2 0 の処理に戻るもの好適である。但し、このような場合には、D L 発行処理の再実行回数をカウントし、再実行回数が限界値を越えれば、重大異常フラグ A B N をセットして D L 発行処理を終えることになる。

30

【 0 3 8 1 】

図 1 9 ( b ) は、正常な動作状態について、確認的に図示したものである。図示の通り、発行されたディスプレイリスト D L は、列記された指示コマンドの順番に、描画回路 7 6 ( ディスプレイリストアナライザ ) によって解析され、各指示コマンドに基づく動作が実行される。この動作は、ディスプレイリスト D L の発行処理や、データ転送回路 7 2 のデータ転送動作 ( S T 2 6 ~ S T 3 0 ) に並行して実行される。

【 0 3 8 2 】

例えば、指示コマンド ( T X L O A D ) が実行されることで、C G R O M 5 5 から必要なテクスチャが読み出されて A A C 領域 ( a ) に取得され、その後、G D E C 7 5 が自動的に起動してデコード動作が実行され、デコード後のデータが所定のインデックス空間に展開される。また、指示コマンドによっては、ジオメトリエンジン 7 7 その他が機能するが、何れにしても、描画回路 7 6 の各部が協働することで、ディスプレイリスト D L に対応する画像データがフレームバッファ F B a , F B b に完成されることになる。

40

【 0 3 8 3 】

続いて、D M A C 回路 6 0 を介在させてディスプレイリスト D L を発行する場合を、図 2 0 に基づいて説明する。何ら限定されないが、D M A C 回路 6 0 に内蔵された第 1 ~ 第 4 の D M A チャンネルのうち、第 3 の D M A チャンネルを使用することにする。

【 0 3 8 4 】

図 2 0 の実施例では、まず、所定のデータ転送レジスタ R G i j と、所定の描画レジスタ

50

R G i j に各々クリア値を設定して、データ転送回路 7 2 と、描画回路 7 6 を初期化する ( S T 2 0 )。この処理は、図 1 9 のステップ S T 2 7 のエラー処理と同じであり、F I F O バッファを含んだデータ転送回路 7 2 の内部回路が初期化され、データ転送の進行状態を示すデータ転送レジスタのステータスビットが初期値となり、データ転送全体を初期化中であることを示すビットが所定値となる。

#### 【 0 3 8 5 】

描画回路 7 6 についても同様であり、上記した(1) 内部パラメータを初期値に設定すること、(2) 内部制御回路を初期状態に設定すること、(3) G D E C 7 5 を初期化すること、(4) A A C 領域のキャッシュ状態を初期化する処理が含まれている。また、描画回路の初期化処理 ( 図 2 0 の S T 2 0 ) においても、前記した所定の描画レジスタ R G i j を初期化しても良い。なお、図 1 9 の処理において、このような初期化処理を最初に実行しても良い。

10

#### 【 0 3 8 6 】

図 2 0 の処理では、次に、初期化処理が正常に完了したことを、データ転送回路 7 2 と描画回路 7 6 の動作状態を特定する所定のステータスレジスタ R G i j を R E A D して確認する ( S T 2 1 )。そして、万一、初期化できない場合には、重大異常フラグ A B N をセットして処理を終える ( S T 2 2 )。但し、このような事態は、実際にはほぼ発生しない。

#### 【 0 3 8 7 】

次に、データ転送回路 7 2 の転送動作態様と、データ転送回路 7 2 内部の伝送経由とを、所定のデータ転送レジスタ R G i j に設定する。設定内容は、特に限定されないが、ここでは、C P U I F 部 5 6 から C h B 制御回路 7 2 b を経由すること、及び、C P U バス制御部 7 2 d への転送プロトコルに関し、D M A C 回路 6 0 への設定に従うと設定する ( S T 2 3 )。

20

#### 【 0 3 8 8 】

次に、転送総サイズを、所定のデータ転送レジスタ R G i j に設定する。図 1 9 の場合と同様、データ総量 = 2 5 6 となる。なお、無調整手法 ( C ) を採る場合は、このタイミングで、最低データ量 D m i n の整数倍の転送総サイズを決定して設定することになる。次に、所定の描画レジスタ R G i j への設定値に基づいて、描画回路 7 6 の描画動作を開始させる ( S T 2 5 )。図 2 5 ( a ) におけるタイミング t 1 , t 2 , t 3 , t 4 は、ステップ S T 2 5 の動作タイミングでもある。そして次に、D M A C 回路 6 0 の動作を開始させた上で ( S T 2 6 )、データ転送回路 7 2 のデータ転送動作を開始させる ( S T 2 7 )。

30

#### 【 0 3 8 9 】

D M A C 回路 6 0 の動作を開始処理は、図 2 0 ( b ) に示す通りであり、先ず、D M A C 転送を禁止した状態で、一サイクルのデータ転送単位 ( 1 オペランド ) の転送が完了するのを待つ ( S T 4 0 )。詳細な動作内容は、図 2 1 に示す処理と同じであり、D M A C 転送を禁止設定する処理 ( S T 5 3 ) と、その後の待機処理 ( S T 5 4 ) に区分される。

#### 【 0 3 9 0 】

このような処理を設けるのは、(1) 他の実施例では、メイン制御処理やタイマ割込み処理 ( 図 1 7 ) で、D M A C 回路 6 0 ( 第 3 の D M A チャンネル ) を使用する可能性があること、及び、(2) 図 1 7 のステップ S T 5 の処理を設けない他の実施例において、ディスプレイリスト D L の発行を開始した D M A C 回路 6 0 が、その動作周期 (  $\delta$  ) 内では D L 発行動作を終了できない場合もあり得ることなどを考慮したものである。

40

#### 【 0 3 9 1 】

上記のような例外事態において、動作中の D M A C 回路 6 0 に対して、新規の設定値 ( 矛盾する設定値など ) を追加的に設定すると、正常な D M A 動作が全く担保されず、深刻なトラブルが懸念されるが、ステップ S T 4 0 の処理を設けることで、その後の設定値に基づく正常動作が担保される。すなわち、本実施例を一部変更した変更実施例でも、先行するトラブルに拘らず、その後の正常な D M A 動作を実現することができる。

#### 【 0 3 9 2 】

以上のような意義を有するステップ S T 4 0 の処理を実行すれば、次に、D M A C 回路 6

50

0の動作条件を設定する(ST41)。具体的には、図6に示す通り、サイクルスチール転送モードを選択し、一オペランド転送を32bit転送×2回とする。また、Sourceアドレスは、RAM59のリストバッファ領域(DLバッファBUF)のアドレスであるので順次増加と認識すべきこと、一方、Destinationアドレスは、転送ポートTR\_PORTであるため固定値とすべきことを規定する。

#### 【0393】

次に、RAM59のDLバッファBUFの先頭アドレスを、DMAC回路60の動作を規定する所定の動作制御レジスタREGに設定すると共に(ST42)、転送先アドレスである転送ポートTR\_PORTのアドレスを設定する(ST43)。また、転送総サイズ、つまり、ディスプレイリストDLのデータ総量を256バイトに設定した上で(ST44)、DMAC回路60のDMA動作を開始させる(ST45)。

10

#### 【0394】

ところで、ここまでの説明は、指示コマンドの実質ビット長が、全て32bitの整数倍であることを前提にした。しかし、ディスプレイリストDLや指示コマンドの構成は必ずしも限定されないので、以下、このような場合について説明する。

#### 【0395】

例えば、前記した無調整手法(C)を採る場合も含め、ディスプレイリストDLのデータ総量Xが、32bitの整数倍ではない任意値Xである場合には、ステップST44の処理では、この任意値Xを、適切な転送量MODに調整した上で、転送総サイズの設定処理を実行する。ここで、適切な転送量MODは、一オペランド転送についての設定内容と、データ転送回路72の最低データ量Dmin(バイト)とに基づいて規定される。

20

#### 【0396】

具体的には、一オペランド転送設定がNバイト×M回であれば、転送量MODは、N×M(バイト)の整数倍であって、且つ、Dmin(バイト)の整数倍の値に調整される。例えば、N×M=8×4、Dmin=256であれば、任意値X(=300)バイトは、転送量MOD(=512)バイトに調整される。

#### 【0397】

以上、一般論も含め説明したが、DMAC回路60のDMA動作は、図6に示すようなサイクルスチール転送動作が開始され、CPUの動作を特に阻害することなく、ディスプレイリストDLが、実施例の場合には32ビット毎に、転送ポートTR\_PORTに転送される。そして、転送されたデータは、転送回路ChBを経由して、描画回路76に転送される。

30

#### 【0398】

このような動作を実現するため、本実施例では、ステップST45の処理に続いて、データ転送回路72の転送動作を開始させて処理を終える(ST27)。その後、データ転送回路72は、DMAC回路60から最低データ量Dminを一単位として、ディスプレイリストDLの指示コマンド列を受け、これを描画回路76に転送する。そして、描画回路76は、ディスプレイリストDLの指示コマンドに基づいて描画動作を実行する。したがって、ステップST27の処理の後、演出制御CPU63は、図17のステップST11の処理を開始することができ、VDP回路52による描画動作(DMAC回路60によるDL発行処理)と並行して、音声演出やランプ演出やモータ演出を制御することができる。

40

#### 【0399】

図20(c)は、この動作内容を図示したものである。DMA転送に先行して、描画回路の動作を開始しており(ST25)、描画回路76のディスプレイリストアナライザは、迅速かつ円滑にAnalyze処理を実行し、その他、GDEC75やジオメトリエンジン77などの動作に基づき、フレームバッファFBa、FBbには、各表示装置DS1、DS2について、各一フレーム分の画像データが生成される。

#### 【0400】

ところで、DL発行処理をステップST27の処理で終える図20の構成は、必ずしも、限定されない。例えば、図27～図28のように、音声演出、ランプ演出、及びモータ演出を、他のCPUが制御する場合には、ステップST27の処理の後、DMAC回路60

50

やデータ転送回路 7 2 の正常動作を確認するのが好ましい。図 2 1 は、図 2 0 のステップ S T 2 7 に続く動作であり、正常動作の確認処理を説明するフローチャートである。

【0 4 0 1】

まず、所定のステータスレジスタを参照して、D M A C 回路 6 0 の転送動作が正常に終了していることを確認する ( S T 5 0 ) 。また、データ転送回路 7 2 が転送動作を終了していることを確認する ( S T 5 1 ) 。通常、このような経路で、図 2 0 の D L 発行処理が完了する。

【0 4 0 2】

一方、所定時間待機しても、D M A C 回路 6 0 の動作が完了していない場合、或いは、データ転送回路 7 2 が転送動作を完了していない場合には、描画回路 7 6 とデータ転送回路 7 2 について、所定の V D P レジスタ R G i j にクリア値を設定して、D L 発行処理を初期化する ( S T 5 2 ) 。これは、ディスプレイリスト D L の発行処理が正常に終了していないことに基づく動作であり、具体的には、図 1 9 のステップ S T 2 7 のエラー処理や、図 2 0 のステップ S T 2 0 の初期処理と同じ内容である。

【0 4 0 3】

すなわち、この場合も、描画回路 7 6 は、既に動作を開始しており、ある程度の処理を終えているので、描画回路 7 6 の初期化処理には、(1) ディスプレイリスト D L によって設定される可能性のある全ての内部パラメータを初期値に設定すること、(2) 全ての内部制御回路を初期状態に設定すること、(3) G D E C 7 5 を初期化すること、(4) A A C 領域のキャッシュ状態を初期化することが含まれる。

【0 4 0 4】

次に、新規の D M A 転送動作を禁止した上で ( S T 5 3 ) 、実行中の一オペランドの転送動作が終わるのを待つ ( S T 5 4 ) 。先に説明した通り、本実施例では、3 2 b i t 転送 × 2 回を一オペランドとしており、動作中の D M A C 回路 6 0 をいきなり初期化することを避けるためである。

【0 4 0 5】

そして、この準備作業が終われば、D M A C 回路 6 0 の動作を規定する所定の動作制御レジスタ R E G にクリア値を設定して、D M A C 回路 6 0 を初期化する ( S T 5 2 ) 。そして、重大異常フラグ A B N をセットして D L 発行処理を終える。なお、この場合、ステップ S T 5 2 や S T 5 5 の処理によって、異常回復が期待できるので、重大異常フラグ A B N をセットすることなく、図 2 0 のステップ S T 2 0 に戻って、D L 発行処理を再実行するのも好適である。但し、D L 発行処理 ( S T 2 3 ~ S T 2 7 ) の再実行回数をカウントし、再実行回数が限界値を越えれば、重大異常フラグ A B N をセットして D L 発行処理を終える必要がある。

【0 4 0 6】

続いて、プリローダ 7 3 を使用する場合のメイン制御処理について、図 2 2 に基づいて説明する。図 2 2 の処理は、図 1 7 の処理に類似しているが、まず、開始条件判定 ( S T 5 ' ) の内容が相違する。すなわち、プリローダを使用する実施例では、各動作周期の開始時に、描画回路 7 6 とプリローダ 7 3 のステータス情報を READ アクセスして、ディスプレイリスト D L 1 に基づく描画動作が終わっていること、及び、ディスプレイリスト D L 2 に基づくプリロード動作が終わっていること確認する ( S T 5 ' ) 。

【0 4 0 7】

図 2 6 ( a ) のタイムチャートに示す通り、プリローダ 7 6 は、例えば、動作周期 ( T 1 ) に発行されたディスプレイリスト D L 1 に基づき、その動作周期中 ( T 1 ~ T 1 + δ ) に、先読み動作 (プリロード動作) を終えている筈である。また、描画回路 7 6 は、例えば、動作周期 ( T 1 + δ ) で指示された動作開始指令に基づき、その動作周期中 ( T 1 + δ ~ T 1 + 2 δ ) に、ディスプレイリスト D L 1 に基づく描画動作を終えている筈である。

【0 4 0 8】

そこで、( S T 5 ' ) では、描画回路 7 6 とプリローダ 7 3 に関する V D P レジスタ R G i j のステータス情報を READ アクセスして、上記の正常動作を確認するのである。図 2 6

10

20

30

40

50

(a) には、動作周期  $T1$  ,  $T1 + \delta$  ,  $T1 + 2\delta$  ,  $T1 + 4\delta$  の判定タイミングでは正常動作が確認されるが、動作周期  $T1 + 3\delta$  の判定タイミングでは、プリロード動作が終了していない状態が示されている。

【0409】

そして、このような異常時には、異常フラグ  $ER$  をインクリメントした上で ( $ER = ER + 1$ )、ステップ  $ST9$  の処理に移行させている。そのため、図17の実施例の場合と同様に、フレーム落ちが生じる。すなわち、表示領域の切換え処理 ( $ST6$ ) がスキップされるので、同じ画面が再表示される。図25(a)に示す動作期間 ( $T1 + 3\delta \sim T1 + 4\delta$ ) は、その動作状態を示している。

【0410】

また、ステップ  $ST5'$  の判定において、開始条件を満たさない場合には、描画回路76に対して、書換えリスト  $DL'$  に基づく描画動作の開始指示 ( $PT10$ ) が実行されないことで、描画回路76は非動作状態であり、また、新規のディスプレイリストが生成されることもない。なお、図26(a)において、タイミング  $t0$  ,  $t2$  ,  $t4$  は、描画動作の開始指示 ( $PT10$ ) の動作タイミング、より正確には、図23のステップ  $ST26$  のタイミングを示している。

【0411】

以上、ステップ  $ST5'$  の判定が不適合の場合を説明したが、通常の場合は、フレームバッファ  $Fb_a$  ,  $Fb_b$  の表示領域をトグル的に切換えた後 ( $ST6$ )、描画回路76に対して、書換えリスト  $DL'$  に基づく描画動作を開始させる ( $PT10$ )。具体的な内容は、図23に示す通りであり、描画回路76は、演出制御  $CPU63$  の制御に基づき、データ転送回路72 (転送回路  $ChB$ ) を経由して、外付け  $DRAM54$  の  $DL$  バッファ  $BUF'$  から書換えリスト  $DL'$  を取得して描画動作を実行することになる。

【0412】

この動作を実現する図23のフローチャートを説明することに先行して、プリローダ73の動作を確認すると、プリローダ73は、一動作周期前に取得したディスプレイリスト  $DL$  に基づき、 $CGROM55$  の先読み動作 (プリロード) を完了しており、先読みされたデータは、外付け  $DRAM54$  に確保されたプリロード領域に既に格納済みである。また、ディスプレイリスト  $DL$  に記載されているテクスチャロード系のコマンド ( $TXLOAD$ ) については、その  $Source$  アドレスがプリロード領域のアドレスに書換えられ、書換えリスト  $DL'$  として、外付け  $DRAM54$  の  $DL$  バッファ  $BUF'$  に格納されている。

【0413】

なお、この書換え処理において、ディスプレイリスト  $DL$  のデータ総量に変化はなく、書換えリスト  $DL'$  のデータ総量は、ディスプレイリスト  $DL$  と同じである。また、ディスプレイリスト  $DL$  は、標準手法 (B) で作成されており、書換えリスト  $DL'$  の最後は、ディスプレイリスト  $DL$  の場合と同様に  $EODL$  コマンドである。

【0414】

以上を踏まえて、図23について説明すると、演出制御  $CPU63$  は、まず、所定のデータ転送レジスタ  $RGij$  と、所定の描画レジスタ  $RGij$  に各々クリア値を設定して、データ転送回路72と、描画回路76を初期化する ( $ST20$ )。この処理は、図20の  $ST20$  の処理と同一内容である。次に、この初期化処理が正常に終了したことを確認し ( $ST21$ )、万一、所定時間経過しても初期化が完了しない場合には、重大異常フラグ  $ABN$  をセットして処理を終える ( $ST22$ )。

【0415】

通常は、データ転送回路72と描画回路76の初期化は正常に終了するので、続いて、データ転送回路72内部の伝送経路を、所定のデータ転送レジスタ  $RGij$  に設定する ( $ST23$ )。具体的には、外付け  $DRAM54$  から、 $ChB$  制御回路72bを経由して描画回路76にデータを転送すると設定する ( $ST23$ )。次に、書換えリスト  $DL'$  が格納されている外付け  $DRAM54$  の  $DL$  バッファ  $BUF'$  について、その先頭アドレスを、所定のデータ転送レジスタ  $RGij$  に設定する ( $ST24$ )。

10

20

30

40

50

## 【0416】

また、この書換えリストDL'について、転送総サイズを、所定のデータ転送レジスタRGijに設定する(ST25)。先に説明した通り、書換えリストDL'のデータ総量は、ディスプレイリストDLのデータ総量と同じであり、具体的には、例えば、256バイトである。

## 【0417】

次に、所定の描画レジスタRGijへの設定値に基づいて、描画回路76の描画動作を開始させる(ST26)。図25(a)におけるタイミングt1, t2, t3, t4は、ステップST26の動作タイミングでもある。そして、次に、所定のデータ転送レジスタRGijへの設定値に基づいて、データ転送回路60の動作を開始させて処理を終える(ST27)。その後、演出制御CPU63は、データ転送回路72や描画回路の動作に特に  
10 関与せず、次の動作周期で実効化されるディスプレイリストの生成処理(ST7)に移行する。

## 【0418】

一方、ステップST26のタイミングで動作を開始する描画回路76は、書換えリストDL'に基づいた描画動作を実行して、フレームバッファFba, Fbbに、書換えリストDL'に基づいた画像データを生成する。なお、この動作において、描画回路76は、CGROM55をREADアクセスすることなく、もっぱら、プリロード領域をREADアクセスするので、一連の描画動作を迅速に完了することができる。

## 【0419】

以上、ステップPT10の処理内容を説明したので、図22に戻って説明を続けると、ステップPT11の処理の後、プリローダ73を活用する実施例では、次サイクルで実効化されるディスプレイリストDLを、標準手法(B)に基づいて作成する(ST7)。例えば、図26(a)に示す動作周期(T1)では、次サイクルである動作周期(T1+δ)において、描画回路76に参照されるディスプレイリストDLを作成する。  
20

## 【0420】

次に、演出制御CPU63は、作成後のディスプレイリストDLを、描画回路76ではなく、プリローダ73に発行する(PT11)。具体的な動作内容は、図24に示す通りである。先に、プリローダ73を使用しない実施例(図17)に関し、演出制御CPU63が、ディスプレイリストDLを、直接的に、描画回路76に発行する場合(図19)と、DMA C回路60を経由して発行する場合(図20)を示しているが、図24には、発行先がプリローダ73である点を除いて、ほぼ同じ動作が図24(b)と図24(c)に示されている。  
30

## 【0421】

図24(a)は、図24(b)の動作を説明するフローチャートであり、図19のフローチャートとほぼ同じである。但し、CPUIF部56からChC制御回路72cを経由すること、及び、CPUバス制御部72dに関し、そのFIFOバッファの残量をチェックしながらデータ転送動作を実行すると設定する(ST20)。なお、以下の説明では、ChC制御回路72cを、便宜上、「転送回路ChC」と略すことがある。

## 【0422】

次に、転送総サイズ(標準手法(B)で調整された例えば256バイト)を、所定のデータ転送レジスタRGij設定し、管理カウンタCNを64に初期設定する(ST21)。次に、転送回路ChCを経由するデータ転送動作を開始状態に設定すると共に(ST22)、プリローダ73の動作を規定するプリロードレジスタRGijへの設定値に基づいて、プリロード動作を開始させる(ST23)。  
40

## 【0423】

この結果、その後、プリローダ73は、演出制御CPU63が、転送ポートTR\_PORTに書込む指示コマンド毎に、必要な解析(Analyze)処理を実行し、CGROM55をREADアクセスすべき指示コマンド(TXLOAD)を検出すると、そのテクスチャをプリロードして、DRAM54のプリロード領域に保存する。また、テクスチャのSourceアドレスを  
50

変更した書き換えリストDL'をDRAM54のDLバッファ領域BUF'に保存する。

【0424】

なお、図26(a)におけるタイミング $t_1$ 、 $t_3$ 、 $t_5$ は、事実上、図24のステップST23の動作タイミングを示している。但し、この実施例においても、ディスプレイリストDLの発行処理の途中で、何らかの異常が発生すれば、ステップST25やステップST27の処理を実行する。具体的には、データ転送回路72や、プリローダ73の動作を初期化して、可能な範囲でディスプレイリストDLの発行処理(ST20~ST30)を再実行する。プリローダ73の初期化処理には、未完成状態の書き換えリストDL'の消去や、新規にプリロードデータを記憶したプリロード領域のクリア処理が含まれる。

【0425】

以上、プリローダ73を使用する場合と、使用しない場合について詳細に説明したが、具体的な動作内容は、特に限定されない。図25(b)は、演出制御CPU63が生成したディスプレイリストを、生成したその動作周期ではなく、一動作周期 $\delta$ 遅れて描画回路76に発行する実施例を示している。このような実施例の場合には、描画回路76は、一動作周期( $\delta$ )のほぼ全時間を使用できるので、フレーム落ちの可能性が低減される。

【0426】

また、図26(b)は、演出制御CPU63が生成したディスプレイリストを、生成したその動作周期ではなく、一動作周期遅れてプリローダ73に発行する実施例を示している。この場合は、プリローダ73は、一動作周期( $\delta$ )のほぼ全時間を使用してプリロード動作を実行することができるので、この場合も、フレーム落ちの可能性が低減される。

【0427】

なお、ここまでの説明では、複合チップ50を使用することになっているが、必ずしも、演出制御CPU63とVDP回路52を、一素子に集積化する必要はない。更にまた、上記の実施例では、演出制御全体を、単一のCPU(演出制御CPU63)で制御しているが、上流側のCPUと、下流側の演出制御CPU63が互いに協働して、演出制御動作を実行しても良い。

【0428】

図27~図28は、このような実施例を示すブロック図である。図示の通り、この実施例では、上流側の演出制御CPUが、音声演出と、ランプ演出と、モータ演出を制御している。一方、下流側のCPU回路51は、演出制御CPUから受ける制御コマンドCMD'に基づいて、画像演出だけを制御している。

【0429】

このような構成を採る場合には、CPU回路51は、図17(a)のステップST12の処理と、図17(b)の処理を実行する必要がなく、十分に時間をかけて複雑なディスプレイリストDLを生成することができ、より複雑で高度な3D(Dimension)などの画像演出を実現することができる。このような場合には、ディスプレイリストが大型化するので、その場合には、ディスプレイリストDLのデータ総量は、ダミーコマンドを付加することで、512バイト又はそれ以上の $N \times 256$ バイトに調整される。

【0430】

また、下流側のCPU回路51の動作は、画像演出制御に特化されるので、ディスプレイリストDLの発行後、描画動作が完了するのを確認することもできる。図19の下方は、この場合の動作制御例を示しており、限界時間を超えても描画動作が完了しない場合には、重大異常フラグABNをセットして処理を終える(ST32)。なお、下流側のCPU回路51の処理は、画像演出制御だけであるので、簡易的には、描画動作の完了を無限ループ状に待機するのでも良い。

【0431】

このような構成を採る場合、図17(a)の開始条件判定(ST5)を所定時間繰り返すことができる。このように構成しても、描画動作完了の遅れが、それほど長くなければ、表示領域(0)と表示領域(1)の切り換えが遅れるだけの問題しか生じない。すなわち、図29(a)に示す動作周期 $T_1 + 3\delta$ のように、表示動作が二回繰り返される一動作

10

20

30

40

50

周期の中で、前半だけ、フレーム落ち状態となり、後半は、正常なフレームが表示される。

【0432】

この点は、プリローダを使用する場合も同様であり、図22(a)の開始条件判定(ST5')を所定時間繰り返すことができる。そして、多少の遅れであれば、図29(b)に示す動作周期 $T1 + 3\delta$ のように、前半だけ、フレーム落ち状態となり、後半は、正常なフレームが表示される。但し、描画動作の完了が大幅に遅れると、図25(a)の動作周期 $T1 + 3\delta$ と同様に、完全なフレーム落ちが生じることになり、万一、このような事態が継続すると、WD T回路58が起動することになる。この点は、プリローダを使用しない場合も同様である。

【0433】

また、CPU回路51の制御動作が画像演出制御に特化される場合、DMA転送を採用する実施例では、図21の下方に示す通り、描画回路76の描画動作の完了と、データ転送回路72の動作完了と、DMAC回路60の動作完了が判定される(ST50'~ST52')。そして、何れかの動作が正常に終了しない場合には、データ転送回路72と、描画回路76の動作を初期化し、ステップST53~ST55の処理と同様の処理(ST55'~ST57')が実行される。なお、この場合も、所定回数だけ、DL発行処理を再実行するのが好適である。

【0434】

以上、メイン表示装置DS1及びサブ表示装置DS2のフレームバッファFBa, FBbとして、各表示装置の水平ピクセル数に完全一致する水平サイズのインデックス空間を構築する実施例について説明した。図30(a)は、この関係を確認的に図示したものであり、仮想描画空間上の描画領域( $W \times H$ )と、インデックス空間上の有効データ領域(実描画領域 $W \times H$ )とが、何れも、表示装置の水平/垂直ピクセル数に一致する場合を示している。

【0435】

このような対応関係において、ディスプレイリストDLによる仮想描画空間への描画動作は、必ずしも、描画領域( $W \times H$ )に限定されないもので、例えば、図30(a)上部の左傾斜線で示すように、描画領域( $W \times H$ )を超える描画画像( $W' \times H'$ )について、その描画位置を時間的に移動させることで、図30(a)下部の右傾斜線で示す実描画領域 $W \times H$ への描画内容を、縦/横/斜めに適宜に移動させることが可能となる。

【0436】

また、このような演出を実行するため、例えば、図30(b)に示すように、表示装置の水平ピクセル数より大きい水平サイズ $W$ のインデックス空間を設けても良い。この場合には、ディスプレイリストDLの指示コマンドL12(SETDAVF)で定義される仮想描画空間上の描画領域 $W \times H$ は、表示装置の水平/垂直ピクセル数に対応する実描画領域 $w \times h$ より大きく設定される。なお、図30(b)の下部には、実描画領域 $w \times h$ が、右傾斜線で示されている。

【0437】

そして、実描画領域 $w \times h$ の縦横寸法は、図17のステップSS30の処理で、表示装置の表示ライン数と水平画素数として特定され、また、実描画領域 $w \times h$ の左上端点は、図17のステップSS31の処理で、垂直/水平表示開始位置として、所定の表示レジスタに設定される。

【0438】

一方、インデックス空間における基点アドレス( $X, Y$ )は、ディスプレイリストの指示コマンドL11によって、所定の描画レジスタに設定される。先に説明した通り、具体的には、環境設定系の指示コマンドL11(SETDAVR)によって、インデックス空間IDX上の左上基点アドレスが、例えば(0, 0)と規定される。そして、実描画領域 $w \times h$ の左上端点を、定常処理において適宜に移動させれば、図30(b)下部の右傾斜線で示す実描画領域 $W \times H$ の描画内容は、縦/横/斜めに適宜に移動することになる。

【0439】

図 1 7 に関して説明した通り、ステップ S S 3 0 ~ S S 3 2 に係る V D P レジスタ R G i j については、初期設定後に、書き込み禁止設定がされているが（第 2 の禁止設定 S S 3 4 ）、上記の演出を実行するタイミングでは、所定の V D P レジスタ R G i j に解除値を書込むことで、この禁止設定が解除される。

#### 【 0 4 4 0 】

ところで、上記の実施例では、第 1 種や第 2 種の禁止設定レジスタを活用して、所定のシステム制御レジスタ R G i j や初期設定系の所定の V D P レジスタ R G i j を、画一的に書き込み禁止状態に設定して（図 1 7 や図 2 2 の S S 3 3 , S S 3 4 参照）、これらのレジスタへの設定値が、その後、ノイズなどの影響で変更されないようにした。しかし、このような書き込み禁止設定をすることなく、重要なシステム制御レジスタ R G i j の設定値については、所定時間毎に設定処理を繰り返すのも好適である。

10

#### 【 0 4 4 1 】

図 3 1 は、このような場合の処理を説明する図面であり、初期設定処理（S T 3）において設定すべき設定値は、制御メモリ 5 3（PROGMEM）に記憶された設定値テーブル SETTABLE に纏められている。なお、図 1 7 のステップ S T 3 では、説明を省略しているが、(a) 初期値設定テーブル SETTABLE に基づいて初期設定処理を実行すること、及び、(b) 初期値設定テーブル SETTABLE の内容については、図 1 7 の実施例も、以下に説明する内容と実質的に同じである。

#### 【 0 4 4 2 】

そして、何れの実施例でも、設定値テーブル SETTABLE は、V D P レジスタ R G i j のレジスタアドレス値と、そのレジスタ R G i j への設定値を一組みとした複数組（N 組）で構成されている。特に限定されないが、レジスタアドレス値は 1 6 b i t 長、設定値は 3 2 b i t 長に固定化されており、各々が固定長であることで、初期値設定テーブル SETTABLE のデータ容量は、 $6 \times N$  バイト（ $= 4 8 t \times N b i t$ ）長、V D P レジスタ R G i j は N 個となっている。

20

#### 【 0 4 4 3 】

但し、図 1 7 の実施例では、初期値設定テーブル SETTABLE が一回だけ READ アクセスされ、全 N 個の V D P レジスタ R G i j が一回だけ初期設定されるのに対して、図 3 1 の実施例では、全 N 個の V D P レジスタ R G i j は、一回だけ初期設定される N 1 個の V D P レジスタ R G i j と、一回目の初期設定の後、 $1 / 3 0$  秒毎に繰り返し初期設定される N 2 個の V D P レジスタ R G i j とに区分される。

30

#### 【 0 4 4 4 】

そして、図 3 1 の実施例では、繰り返し初期設定される設定値には、（1）DMA 転送動作についての設定値、（2）V R A M に対する設定値、（3）割込みに関する設定値、（4）表示回路 7 4 に関する設定値、及び、（5）描画回路 7 6 に関する設定値が含まれている。

#### 【 0 4 4 5 】

（1）DMA 転送動作に関する設定値は、例えば、ステップ S T 4 1 で規定する動作条件の前提条件となる設定値であり、図 2 0（c）や図 2 4（c）における動作条件の違いに拘わらず固定的に適用される基本設定値である。具体的には、(a) D A M C 回路 6 0 に内蔵された F I F O バッファ（N 段）がどれだけ開放されると、転送元に転送要求をするかの閾値（例えば全体の  $1 / 2$  段）、(b) 転送先や転送元とハンドシェイク動作をするか否か（例えば No）などの設定値が含まれる。

40

#### 【 0 4 4 6 】

また、（2）V R A M の設定値には、リフレッシュ動作のリフレッシュ周期が含まれている。内蔵 V R A M 7 1 は、このリフレッシュ周期で動作することで、記憶データの自然放電が防止される。次に、（3）割込みに関する設定値には、割込み要求要因となるエラー種別や、割込み信号の出力端子（内蔵 C P U の内部端子）を特定する値であり、例えば、(a) 描画回路 7 6 がフリーズすれば C P U 回路 5 1 に対して描画異常割込みが発生すること（割込み許可状態、図 1 7（d）参照）、(b) 表示装置 D S 1 の V B L A N K 開始時に、C P

50

U回路51に対してVBLANK開始割込みが発生すること（図17（c）参照）などの設定値が含まれている。

【0447】

なお、本実施例は、CPU回路51とVDP回路52とが統合された複合チップ50を使用するが、別チップとする場合には、VDP回路52が割込み信号を出力する出力端子は、CPU回路51の外部割込み入力端子に接続される。

【0448】

また、（4）表示回路に関する設定値には、（a）各フレームバッファの水平／垂直開始位置（SS31参照）、（b）各表示装置の水平同期信号に関する設定値、（c）各表示装置の垂直同期信号に関する設定値、（d）スケラについての設定値、（e）各表示装置の水平画素数と表示ライン数の設定値（SS30）などが含まれている。

10

【0449】

（5）描画回路76に関する設定値には、描画異常割込みが発生するまでのフリーズ時間の設定値が含まれている。この設定値は、例えば、垂直同期信号の周期の整数倍として設定される。図17（d）において説明した通り、ここで規定されたフリーズ期間、描画回路76がVRAMをアクセスしない場合には、描画回路76が個別的にリセットされ（ST16b）、描画回路76に対する動作パラメータが再設定される（ST16c）。

【0450】

上記の通り、この実施例では、重要な設定値については、所定時間毎に繰り返し再設定されるので、ノイズなどの影響で、万一、設定値のビット化けが生じて、その異常が直ちに回復される。また、この実施例では、図17の実施例の場合のように、第1種や第2種の禁止設定レジスタRGijを書込み禁止状態に禁止設定しないので、やや煩雑な禁止解除処理を経ることなく、自由に書換え処理を実行することができる。

20

【0451】

以上、ここまで実施例では、（1a）所定のフリーズ時間を経過する描画回路76の動作フリーズ状態か、（1b）描画回路76が、ディスプレイリストDLに不合理な指示コマンドを検出した場合には、VDP回路52の描画回路76からCPU回路51に対して、描画異常割込みが生じる構成について説明した（図17（d）参照）。そして、描画異常割込み時には、割込み原因を判定した上で（図17（d）のST16a）、その判定結果に応じた処理を実行する構成（ST16c～ST16d）を採った。

30

【0452】

しかし、本発明者の実験によれば、ノイズの多い過酷な動作条件下でも、描画異常割込みが生じることは殆ど無い。そこで、制御負担を軽減化するためには、割込み原因判定処理（ST16a）を設けることなく、画一的に無限ループ処理に移行させるか（図22（b）参照）、パターンチェック回路CHK（図4（b）参照）を機能させるのも好適である（図22（c）のST17a参照）。

【0453】

この場合には、その後、所定時間後にWDT回路58が起動して、複合チップ50全体がリセットされるか、或いは、その後、直ちにVDP回路52だけがリセットされることになる（図4（b）参照）。なお、リセットキーワードの出力処理（ST17a）に基づき、VDP回路52がリセットされた場合には、そのリセット動作の正常終了を確認すると共に、戻り番地を記憶するスタック領域を整理した後（ST17b）、例えば、ステップST4かST13の処理に移行させることになる。

40

【0454】

また、本実施例では、異常判定処理（図17や図22のST5）を設けて描画回路76の動作完了を1／30秒毎に判定しているので、更に制御負担を軽減するべく、実質的に何も実行しない描画異常割込み処理（図22（d））を設けても良い。図22（d）に示す通り、この構成では、描画異常割込み時、直ちにIRET（Interrupt Return）命令を実行してメイン制御処理に戻るので、描画回路76のフリーズ状態などは、そのまま継続されることになる。しかし、本実施例では、図17や図22のステップST5の処理で、フ

50

レーム落ち回数を、異常フラグ E R でカウントしており、いずれ W D T 回路 5 8 か、パターンチェック回路 C H K が起動するので、図 2 2 ( d ) の構成は、図 2 2 ( b ) や図 2 2 ( c ) の構成と実質的に同じである。

#### 【 0 4 5 5 】

また、制御負担を更に軽減化するため、初期設定時（図 1 7 や図 2 2 のステップ S T 3 参照）に、V D P 回路 5 2 を描画異常割込み禁止状態に設定するのも好適である。なお、電源投入時のデフォルト状態が、描画異常割込み禁止状態である構成を採る場合には、(a) 異常割込みの許可／禁止を規定する許否値を設定すべき所定のシステム制御レジスタ R G i j を書き込み禁止状態に設定するか、或いは、(b) 所定時間毎に、前記のシステム制御レジスタ R G i j に禁止値を繰り返し書込むことになる。

10

#### 【 0 4 5 6 】

この構成は、一見、図 2 2 ( b ) や図 2 2 ( b ) の構成より優れているようにも思われる。しかし、この種の遊技機の全機種について、(a) 描画異常割込みを、画一的に禁止状態に設定する構成より、(b) 画一的に許可状態に設定した上で、機種毎に図 1 7 ( d ) の構成を採るか、図 2 2 ( b ) ～ ( d ) の何れかの構成を採るかを選択する方が、制御プログラムの汎用化の観点からは優れている。なお、前者の構成(a) では、機種毎に、初期設定ルーチン（図 1 7 や図 2 2 のステップ S T 3 参照）などを変える必要性（煩雑さ）が生じる。

#### 【 0 4 5 7 】

なお、更なる変更実施例として、複合チップ 5 0 に内蔵された音声回路 S N D を活用するのも好適である。図 3 2 は、このような実施例を示すブロック図である。図 3 2 を、図 4 と対比すれば明らかな通り、この実施例では、音声プロセッサ 2 7 と、音声メモリ 2 8 が不要となり、且つ、C P U 回路 5 1 のデータバス（8 b i t 分）とアドレスバス（2 b i t 分）について、音声回路への外部配線が不要となる。また、アンダーフロー信号 U F の伝送線も存在しないので、この U F 伝送線に重畳するノイズによって、複合チップが誤って異常リセットされるおそれも回避される。

20

#### 【 0 4 5 8 】

また、この実施例では、音声メモリ 2 8 を排除することに対応して、音声メモリ 2 8 に記憶すべき音声データを、C G R O M 5 3 に格納している。図 3 3 ( d ) は、C G R O M 5 3 の記憶内容を図示したものであり、C G R O M 5 3 には、サウンドROMヘッダ情報と、フレーズヘッダ情報 H D と、一群の音声データを圧縮した多数のフレーズデータ P H と、音声回路 S N D の動作を規定する多数のサウンドコマンド S C M D と、が固定的に記憶されている。

30

#### 【 0 4 5 9 】

図示の通り、サウンドROMヘッダ情報は、先頭番地 S N D s t から記憶されており、これに続いて、データサイズ H D v l のフレーズヘッダ情報 H D が、先頭番地 H D s t から記憶されている。また、データサイズ P H v l のフレーズデータ P H が、先頭番地 P H s t から記憶され、データサイズ S C M D v l のサウンドコマンド S C M D が、先頭番地 S C M D s t から記憶されている。

#### 【 0 4 6 0 】

ここで、サウンドROMヘッダ情報とは、具体的には、フレーズヘッダ H D 領域の先頭アドレス H D s t と、フレーズヘッダ H D 領域のデータサイズ H D v l と、フレーズデータ領域 P H の先頭アドレス P H s t と、フレーズデータ領域 P H のデータサイズ P H v l と、サウンドコマンド領域 S C M の先頭アドレス S C M D s t と、サウンドコマンド領域 S C M D のデータサイズ S C M D v l と、を意味する。そして、これらの情報は、電源投入時に音声回路 S N D の内部回路に取得されるようになっている（ステップ S D 4 参照）。

40

#### 【 0 4 6 1 】

また、フレーズヘッダ情報 H D とフレーズデータ P H は、電源投入時に、外付け D R A M 5 4 に転送することで、その後の R E A D アクセスの迅速化を図っている（ステップ S D 6 ）。このように、本実施例では、音声プロセッサ 2 7 と、音声メモリ 2 8 を排除して、小

50

型化と製造コストの抑制を図ると共に、安価で大容量化が容易である一方、アクセス速度の遅いCGROM53の弱点を克服している。

#### 【0462】

以上を踏まえて、電源投入時の初期設定処理について図33(a)に基づいて説明する。なお、これらの処理は、図17や図22のステップST3の処理の一部として実行される。

#### 【0463】

図4(b)に関して説明した通り、電源投入時やWDT58が起動する異常リセット時には、リセット経路2の経路で音声回路SNDがハードウェアリセットされる(ステップSD1)。また、演出制御CPU63が、音声回路SNDの異常を検出した場合には、リセット経路4B又は4Cの経路で音声回路SNDがハードウェアリセットされる(ステップSD1)。なお、演出制御CPU63がパターンチェック回路CHKを機能させることで、音声回路SNDが、他の回路(72, 73, 74...)と共にハードウェアリセットされる場合もある(ステップSD1)。

#### 【0464】

これら何れの場合でも、次に、演出制御CPU63は、リセット動作が正常に完了したことを確認した上で(ステップSD2)、最初に、サウンドデータ領域の先頭アドレスSNDstを、音声回路SNDのシステム制御レジスタRGijに設定する(ステップSD3)。次に、所定のシステム制御レジスタに所定値を設定することで、サウンドROMヘッダ情報HDを内部回路に記憶させる。なお、サウンドROMヘッダ情報HDは、前記した6要素(HDst, HDvl, PHst, PHvl, SCMDst, SCMDvl)であり、図33(c)に記載の通りである。

#### 【0465】

そして、ここまでの処理が正常に動作したことを確認し、万一、正常に終了できない場合には、リセット経路4B又は4Cの経路で、音声回路を個別的にリセットする。但し、通常は、正常終了を確認できるので、続いて、データ転送回路72を利用して、フレーズヘッダ情報HDと、フレーズデータPHを外付けDRAM54に転送する(ステップSD6)。なお、データ転送回路72には、転送先の先頭アドレスBGNと、転送元の先頭アドレスHDstと、転送データ総量HDvl+FDvlなどが適宜に指定される。

#### 【0466】

次に、フレーズヘッダ情報HDとフレーズデータPHは、CGROM55ではなく、外付けDRAM54に存在することを、所定のシステム制御レジスタRGijに設定した上で(ステップSD7)、外付けDRAM54に転送した一群のデータの先頭アドレスBGN(サウンドRAM先頭アドレス)を、所定のシステム制御レジスタに設定する(ステップSD8)。その後、その他の初期設定処理を終えることで(ステップSD9)、音声制御動作が可能となる。

#### 【0467】

先に説明した通り、サウンドROMヘッダ情報、つまり、6個の情報(HDst, HDvl, PHst, PHvl, SCMDst, SCMDvl)は、音声回路SNDの内部回路に保存されているので(ステップSD4)、その後、演出制御CPU63は、フレーズデータなどの必要な情報を、サウンドRAM先頭アドレスBGNとの相対値で指示することができ、この指示を受けた音声回路SNDは、相対アドレス値を絶対アドレス値に変換して、必要な音声処理を実行することになる。フレーズデータなどの音声データは、CGROM55ではなく、外付けDRAM54からREADアクセスされるので、複雑高度な音声演出であっても、これを円滑に実現することができる。

#### 【0468】

以上、各種の実施例について詳細に説明したが、弾球遊技機や回胴遊技機などに限定されないだけでなく、具体的な記載内容についても、何ら本発明を限定しない。例えば、図12に示すパワーオンリセット動作は、制御メモリ53の0x00000000番地以降に確保されたベクタテーブルVECTの情報に基づいて起動されたが、HBTS端子=Hレベルに設定すると共に、CGROM55の先頭領域にベクタテーブルVECTを配置するのも好適

10

20

30

40

50

である。図34(a)や図34(b)は、このような場合のアドレスマップを図示したものであり、演出制御CPU63のアドレス空間CS0は、CGROM55の一部(先頭領域)に確保されている。

#### 【0469】

なお、CGROM55の本体部は、演出制御CPU63からアクセスされることはなく(アクセス不可)、専らVDP回路52からアクセスされるので、アドレス空間CSiに位置付けられることない。先に説明した通り、CGROM55の本体部を、複数のメモリデバイスで構成することもでき、このような場合には、図14(a)のステップSP20の処理によって、SPA0~SPA1のデバイス区間に区分することで、メモリデバイスの特性に適合した最適なREADアクセスが可能となる。

10

#### 【0470】

何れにしても、HBTSLS端子=Hレベルに設定されている場合には、これに対応して、CGROM55のメモリ種別と、バス幅(64/32/16bit)とが、2bit長のHBTBWDS端子と、4bit長のHBTRMSLS端子への固定的な入力値に基づいて予め特定されている必要がある。

#### 【0471】

そして、この実施例では、ベクタテーブルVECTに続いて、CGROM55からのREADアクセスを最適化するためのバスパラメータをCGROM55の先頭領域に記憶しておく必要がある。なお、必須ではないが、演出制御プログラムの違法解析を困難にするため難読化した場合に、それを解読するための難読化パラメータを記憶させておくのも好適である。

20

#### 【0472】

このような構成を採った場合には、電源リセット後、リセットアサート期間に、以下の動作1~動作4がプログラム処理を経ることなく自動的に実行される。まず、HBTRMSLS端子への入力値に基づいてアドレス空間CS0のバス幅が特定され、また、BTBWD端子への入力値に基づいてメモリ種別が自動的に特定され、所定のVDPレジスタRGijに各々設定される(動作1)。この場合のメモリ種別は、パラレルI/F(Interface)形式を採るメモリ素子、シーケンシャルI/F形式を採るメモリ素子に大別される。

#### 【0473】

次に、CGROM55に記憶されている難読化パラメータをロードし、難読化を解除するために必要な情報が内部回路内に自動設定される(動作2)。また、CGROM55に記憶されているバスパラメータが自動的にVDPレジスタRGijに取得される(動作3)。なお、この動作3は、図15のステップSP63のプログラム処理に対応する動作であり、内部回路によって自動的に実行される。

30

#### 【0474】

そして最後に、動作1~3で設定されたバスパラメータを実効化するべく、図15のステップSP64のプログラム処理に対応する動作が内部回路によって、自動的に実行される(動作4)。そして、バスパラメータの設定が実効化されたタイミングで、ベクタテーブルの情報に基づいてプログラムカウンタPCと、スタックポインタSPの値が自動的に設定され、ブートプログラム(初期設定プログラム)の実行が開始される。

40

#### 【0475】

この図34(a)に示す構成によれば、図12(a)のステップSP1のプログラム処理も不要となり、自動的に動作1~動作4が実行されるので、プログラム処理負担が大きく軽減される。そして、この場合も、初期設定プログラムPinItの動作に基づいて、ベクタハンドラVopt以降のプログラムやデータが適宜なRAM領域に転送される。

#### 【0476】

なお、ベクタハンドラVopt以降のプログラムやデータは、必ずしもCGROM55の先頭領域に記憶しておく必要はなく、例えば、制御メモリ53に記憶しておくのでも良い(図34(b))。また、ベクタハンドラVopt以降のプログラムやデータは、必ずしもRAM領域に転送する必要はなく、転送しない場合には、初期化設定プログラムに

50

おけるメモリセクション初期化处理（図 12 の S P 8）が不要となる。

【符号の説明】

【 0 4 7 7 】

G M 遊技機

6 3 C P U

5 1 C P U回路

C S 0 ～ C S 7 アドレス空間

10

20

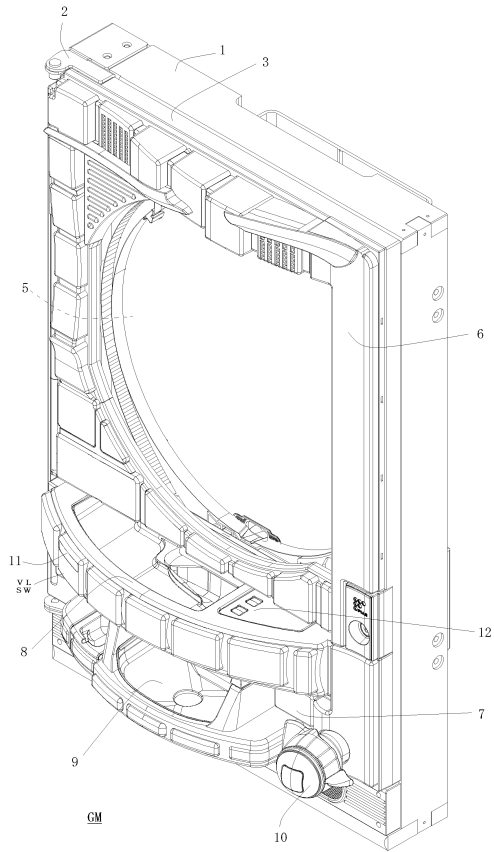
30

40

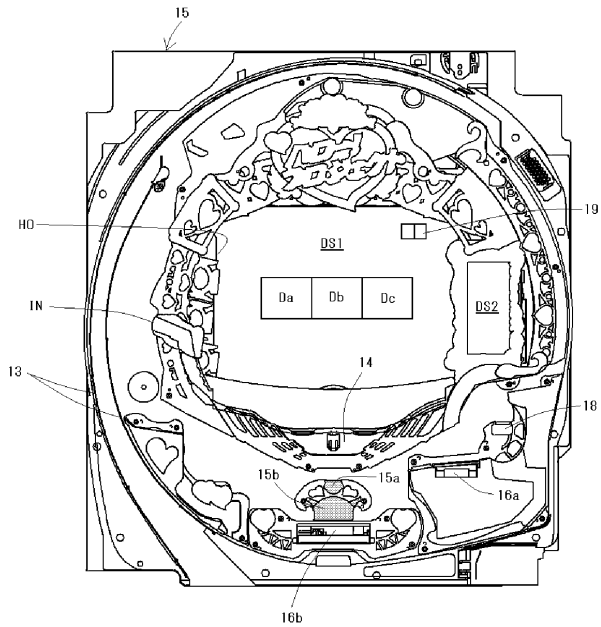
50

【図面】

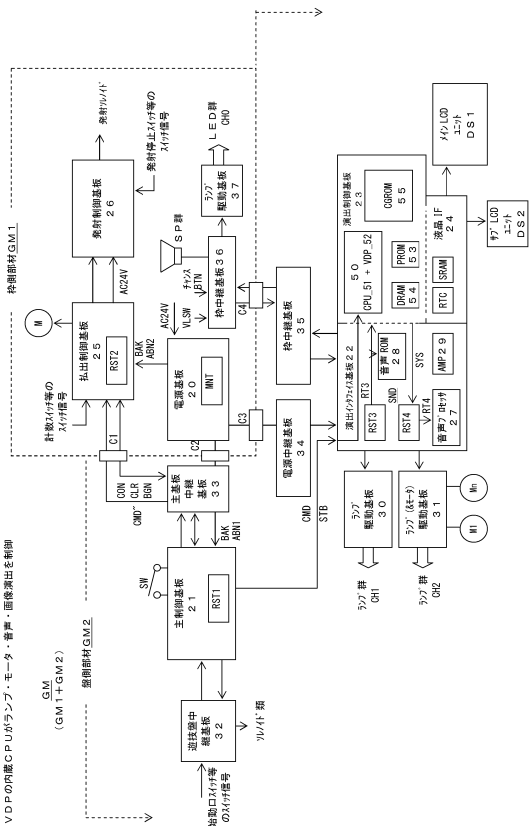
【図 1】



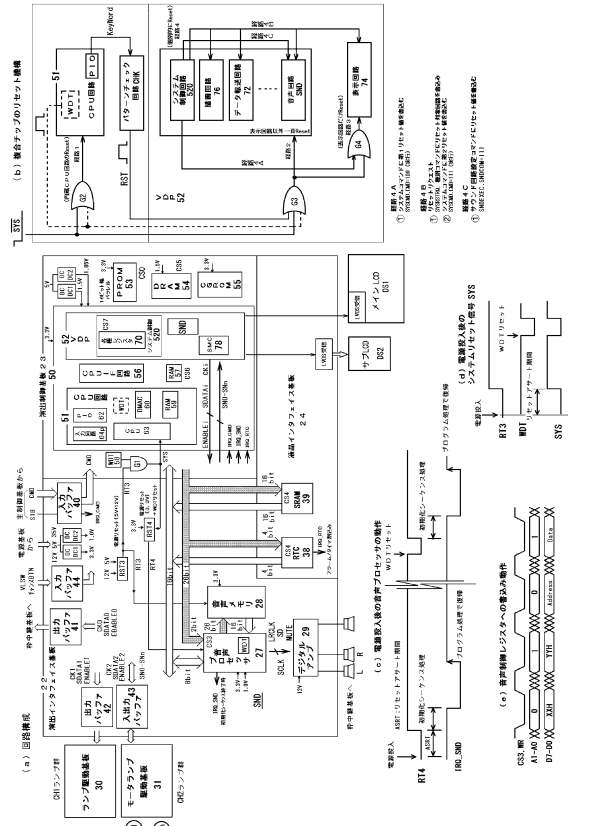
【図 2】



【図 3】



【図 4】



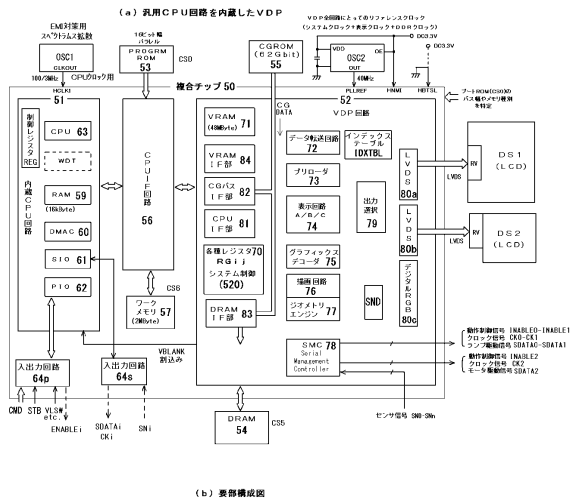
【図 5】



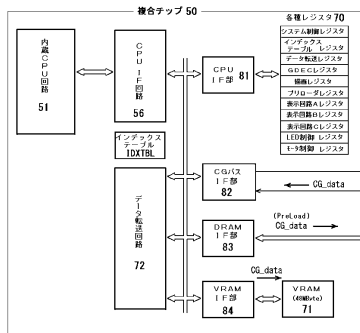
【図 6】



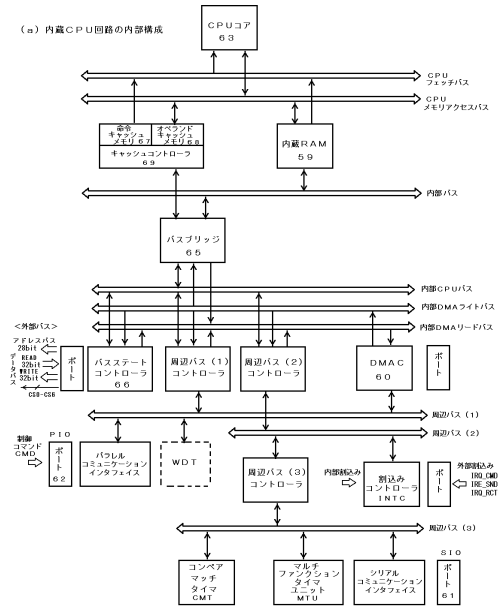
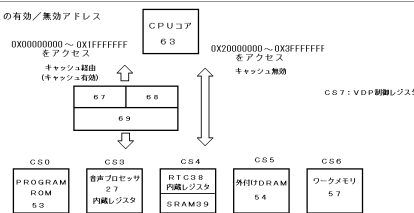
【図 5】



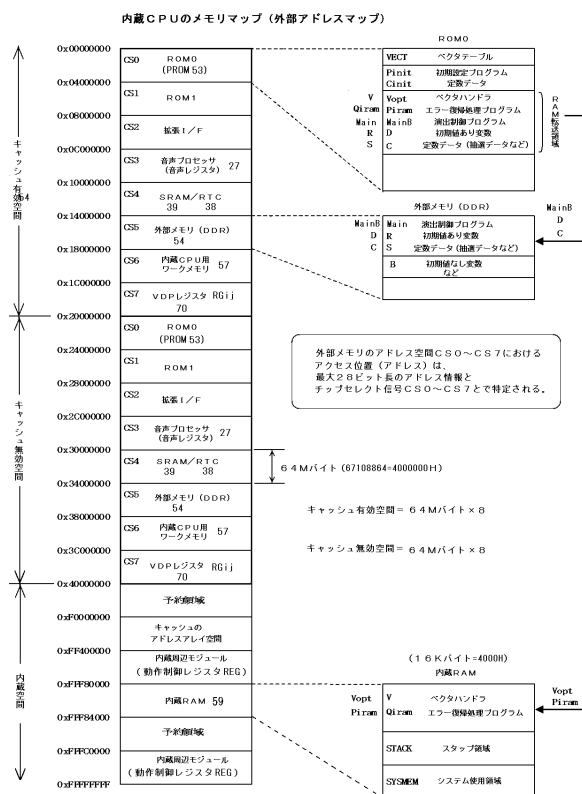
(b) 要部構成図



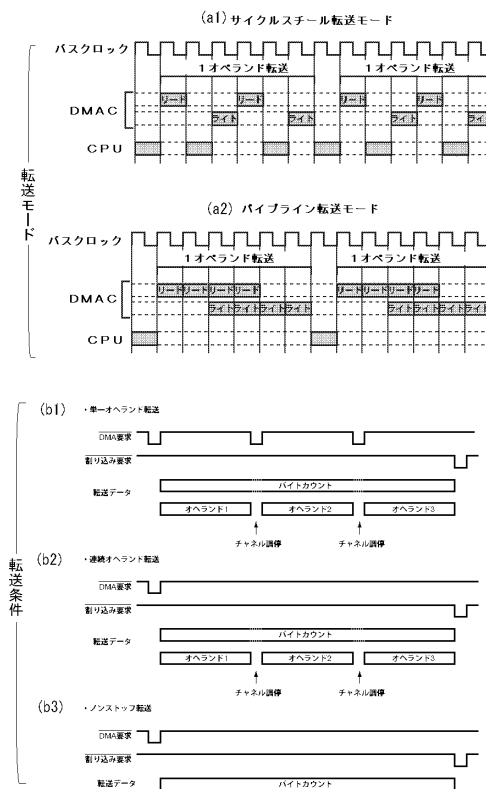
【図 6】

(b) キャッシュの有効/無効アドレス CRU32

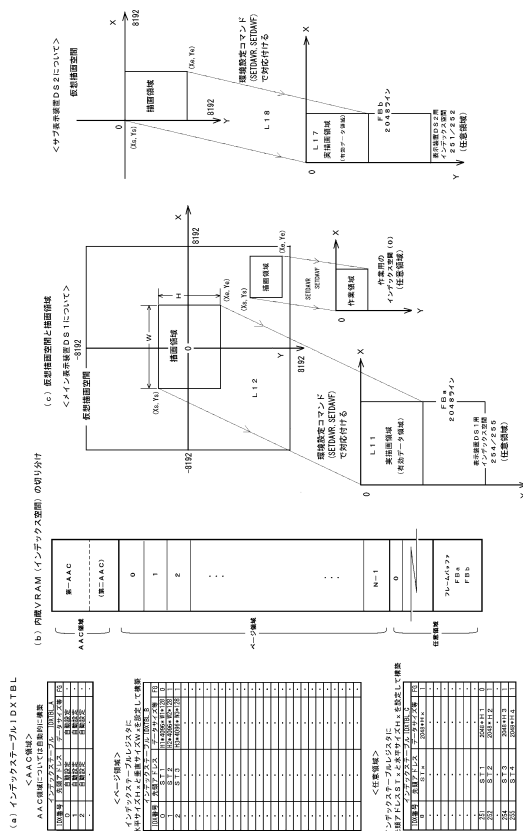
【圖 7】



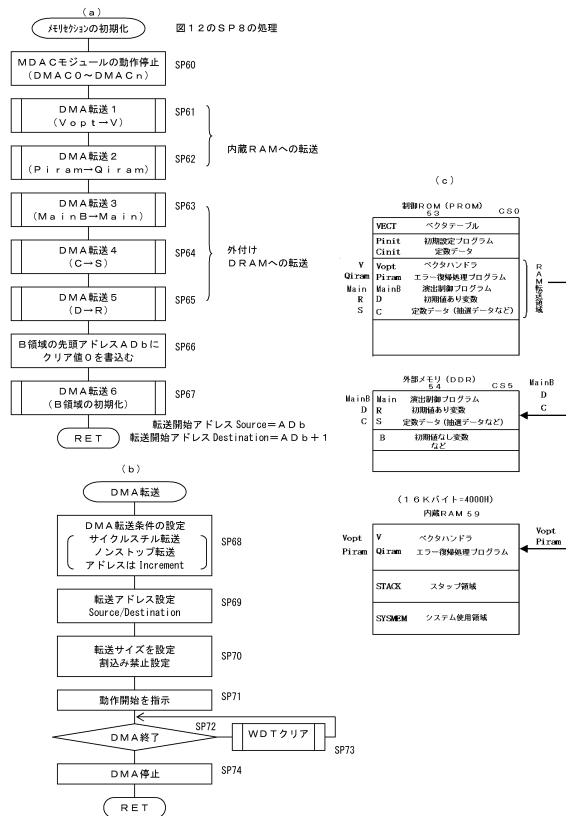
【圖 8】



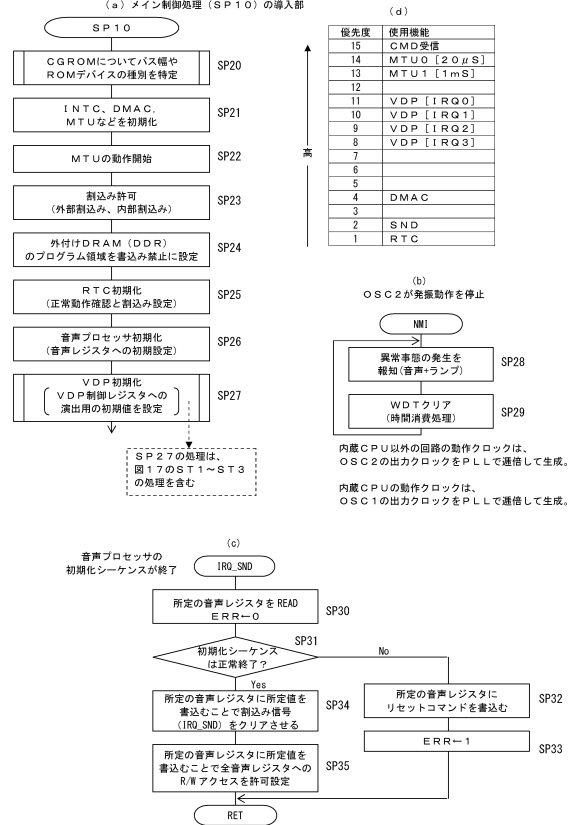
【図 9】



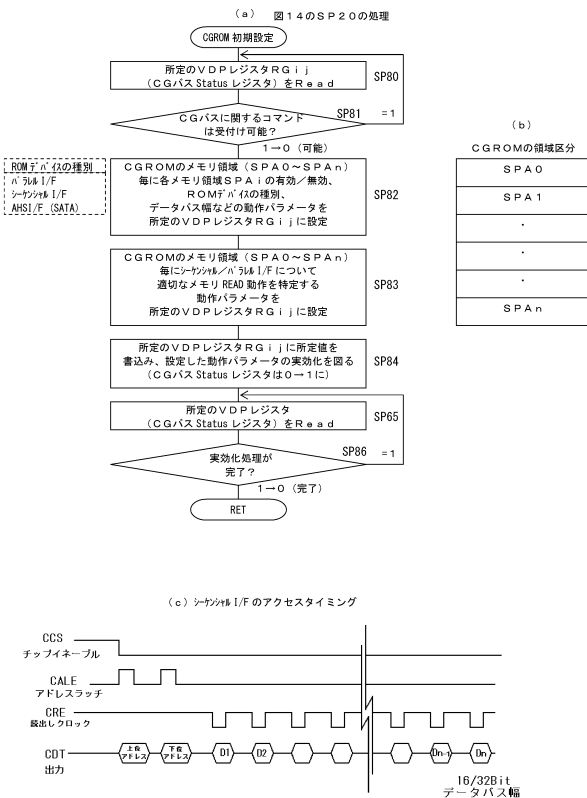
【図 13】



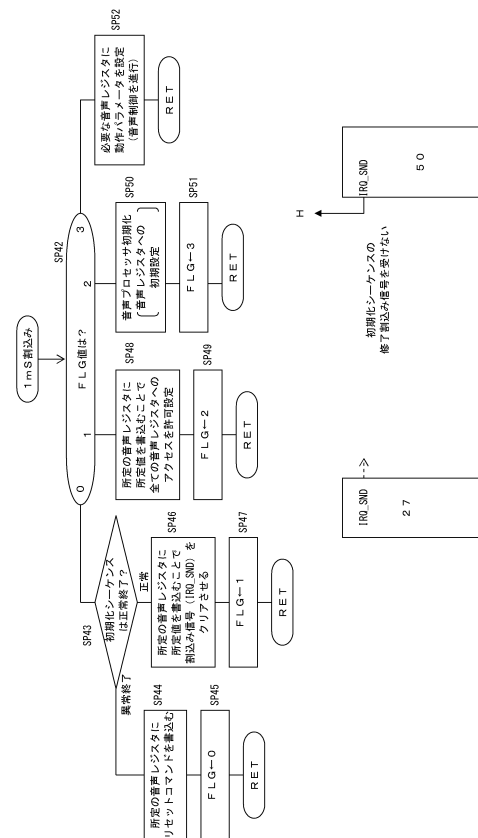
【図 14】



【図 15】



【図 16】



10

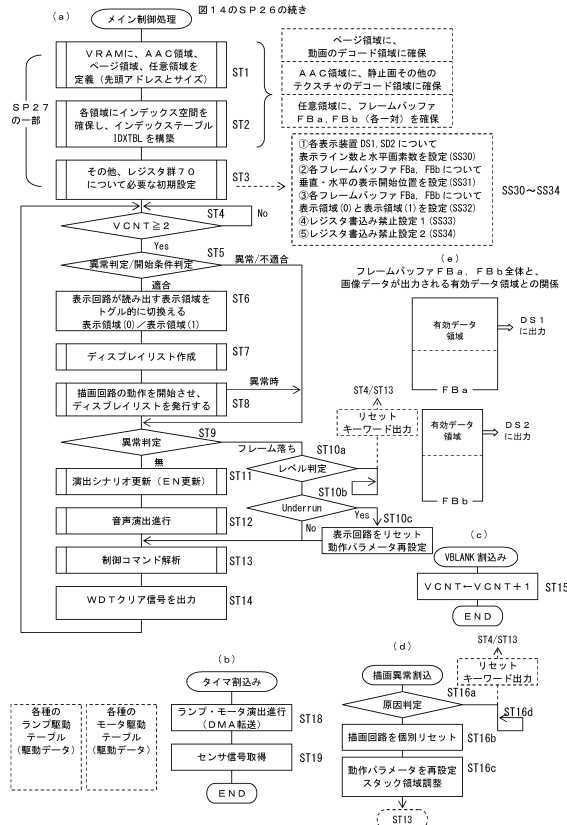
20

30

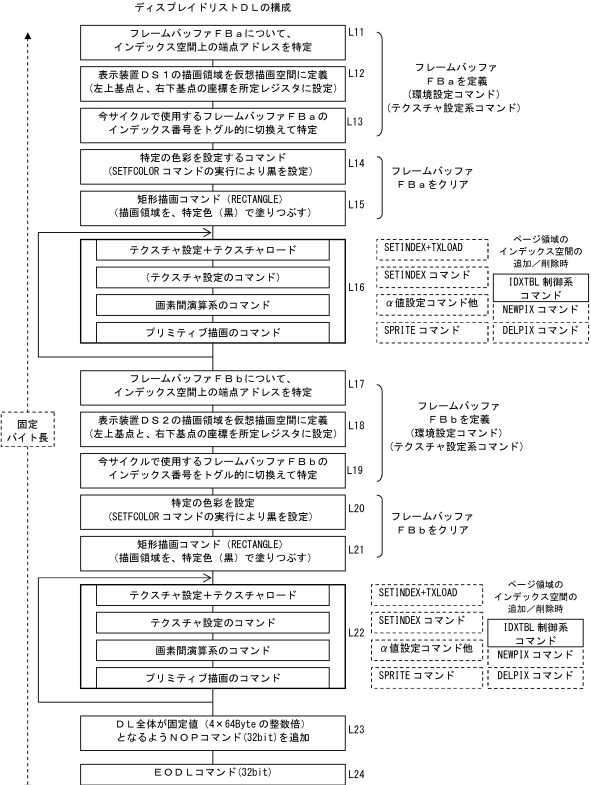
40

50

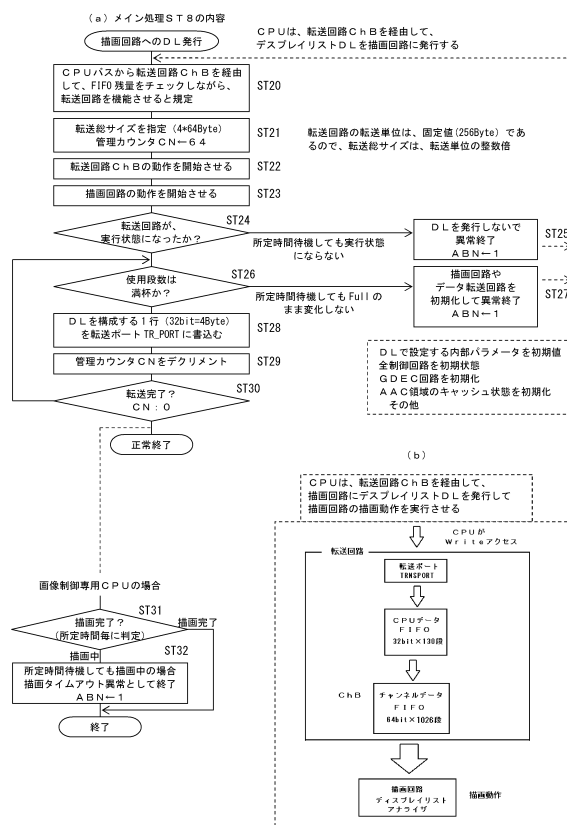
【図 17】



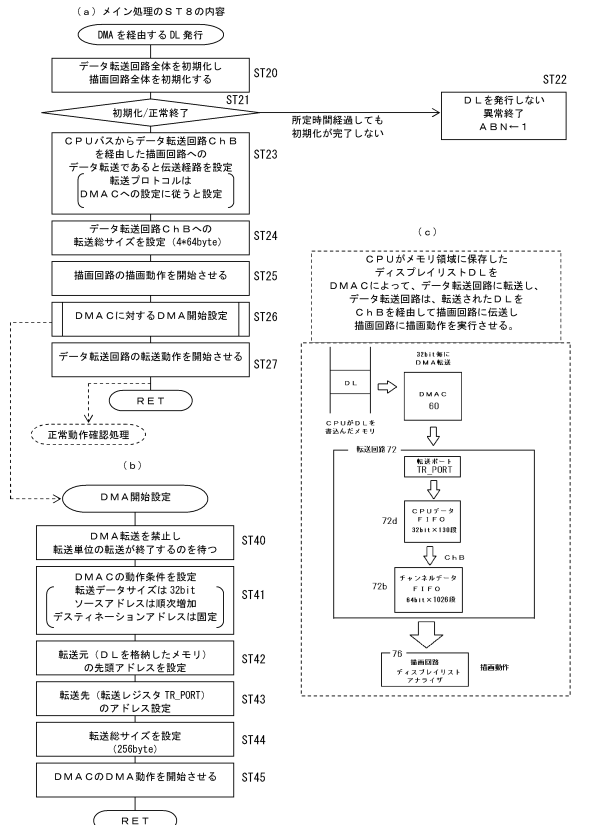
【圖 18】



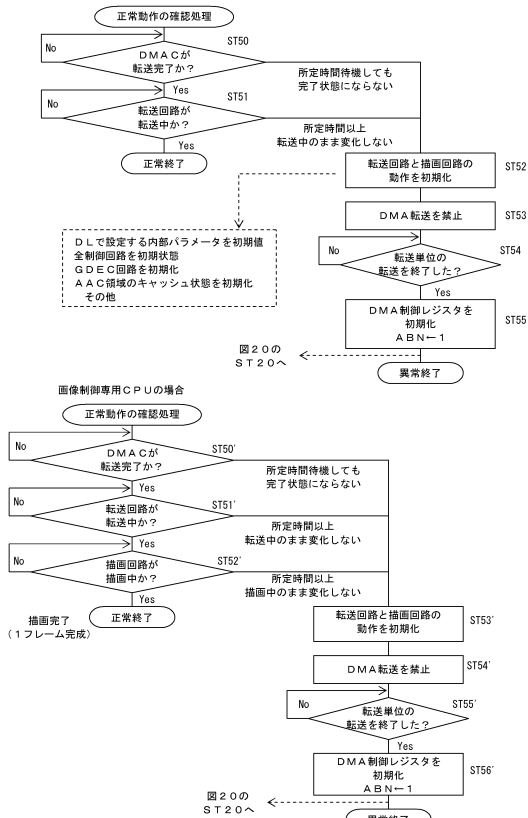
【図 19】



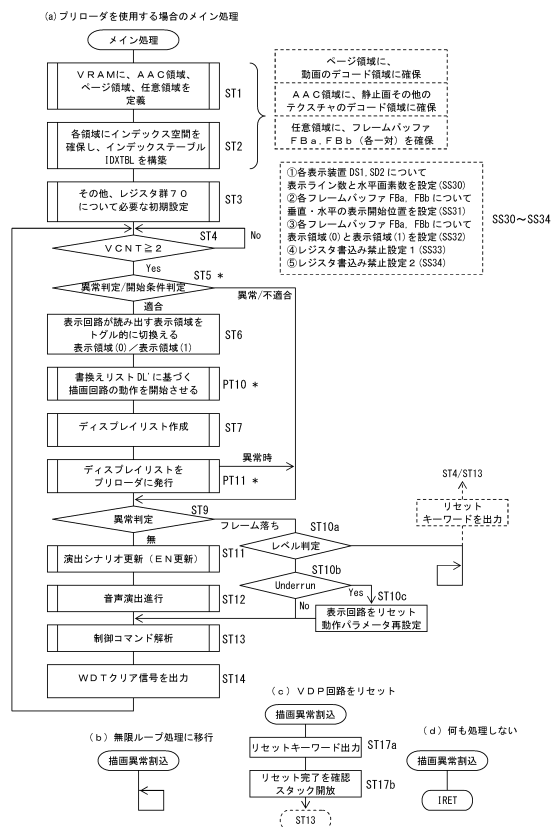
【圖 20】



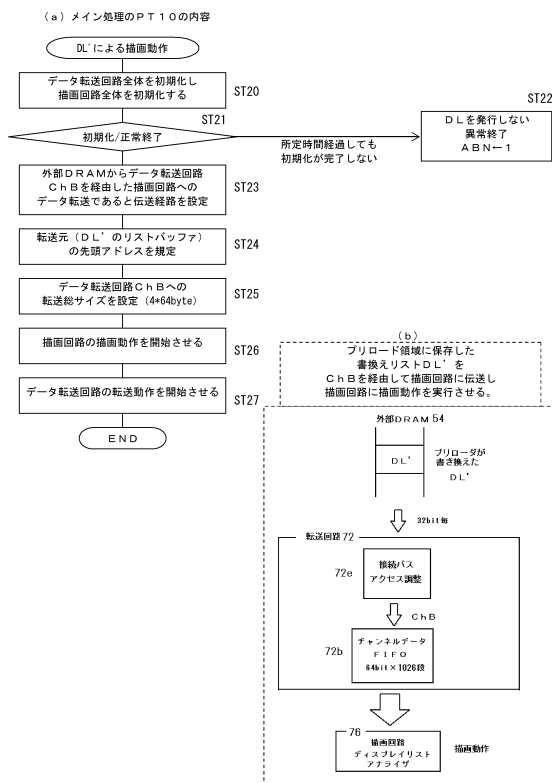
【図 2 1】



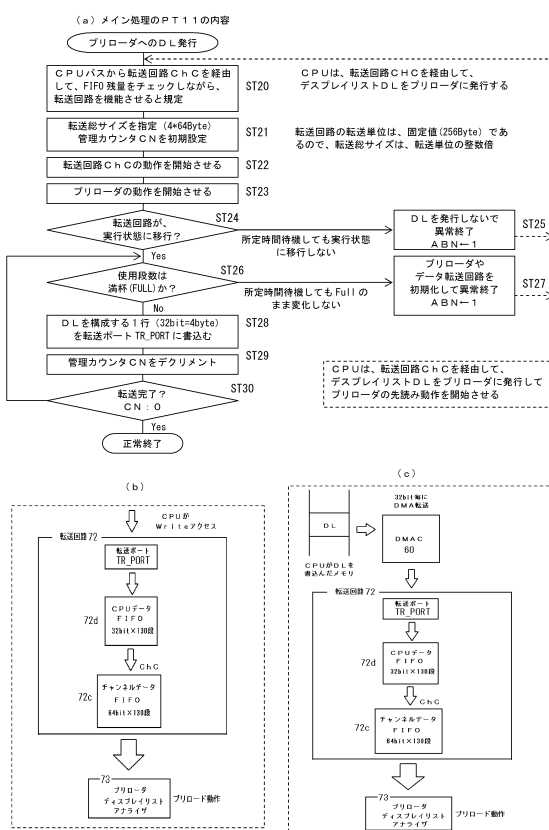
【圖 22】



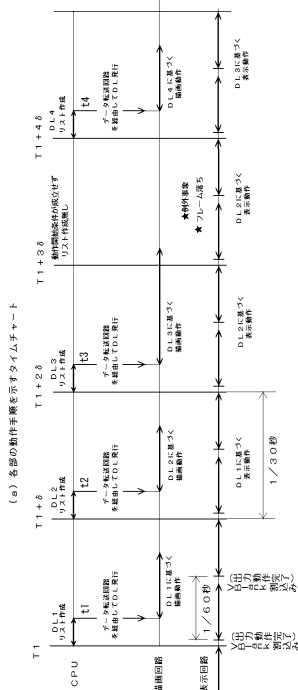
【圖 23】



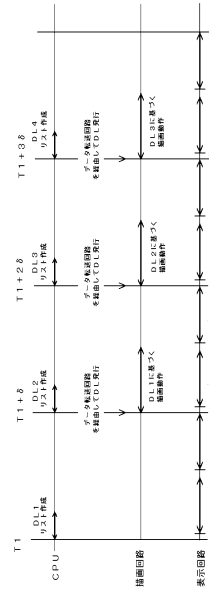
【圖 24】



【図 25】

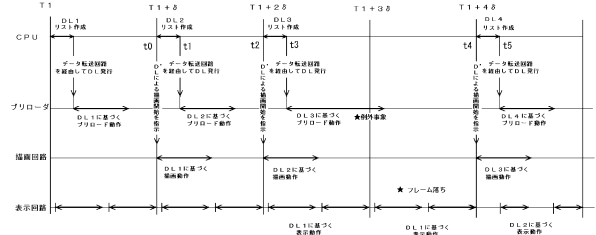


(b) 各部の動作手順を示すタイムチャート (描画回路の動作)

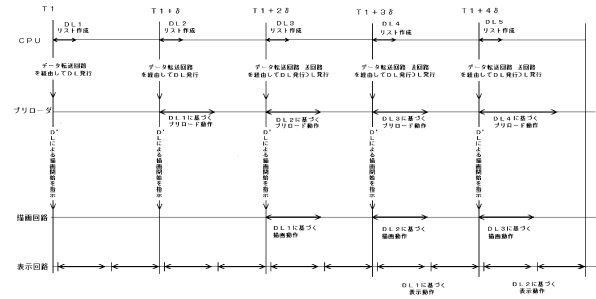


【図 26】

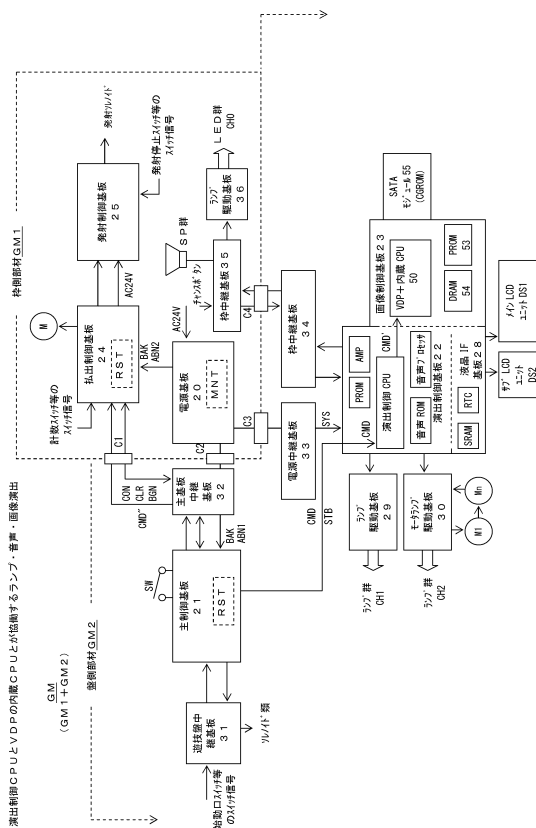
(a) 各部の動作手順を示すタイムチャート (遅延なし)



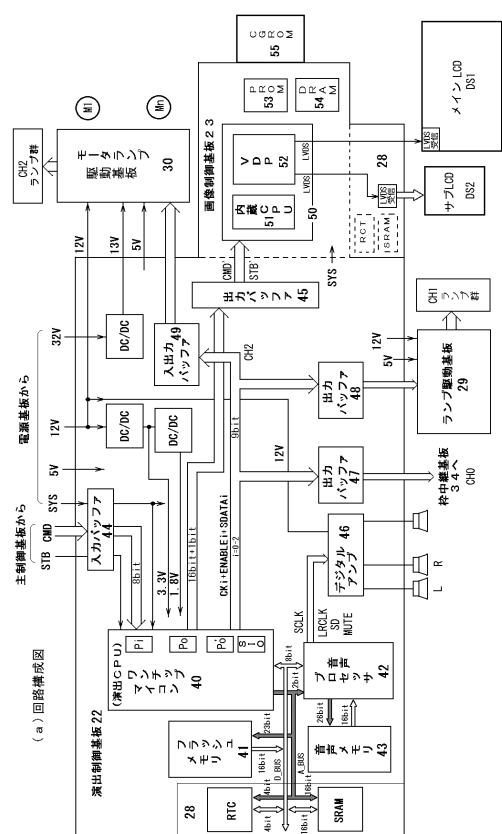
(b) 各部の動作手順を示すタイムチャート (遅延あり)



【図 27】



【図 28】



10

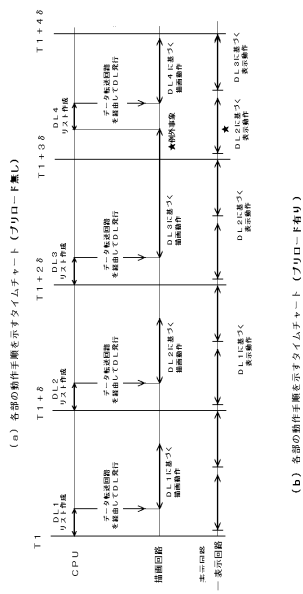
20

30

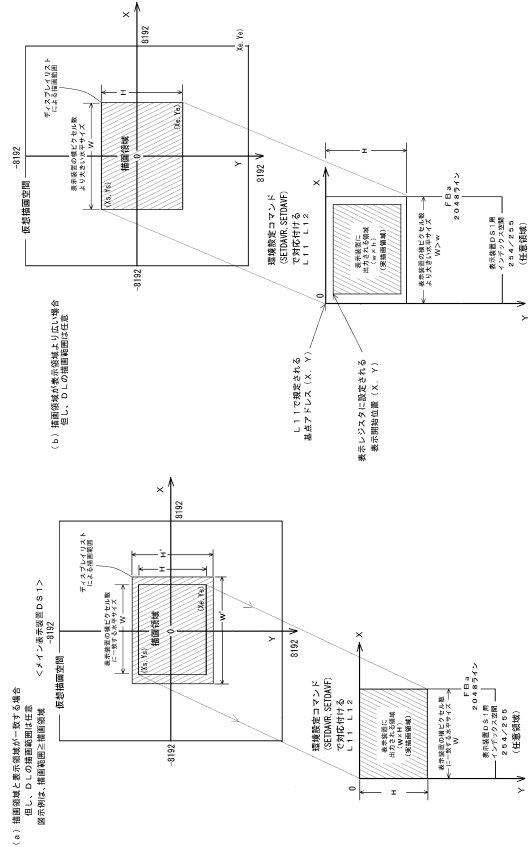
40

50

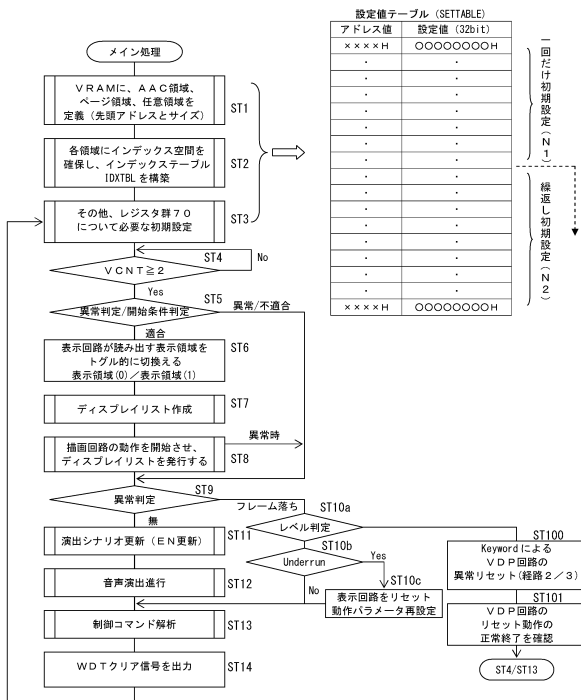
【図 29】



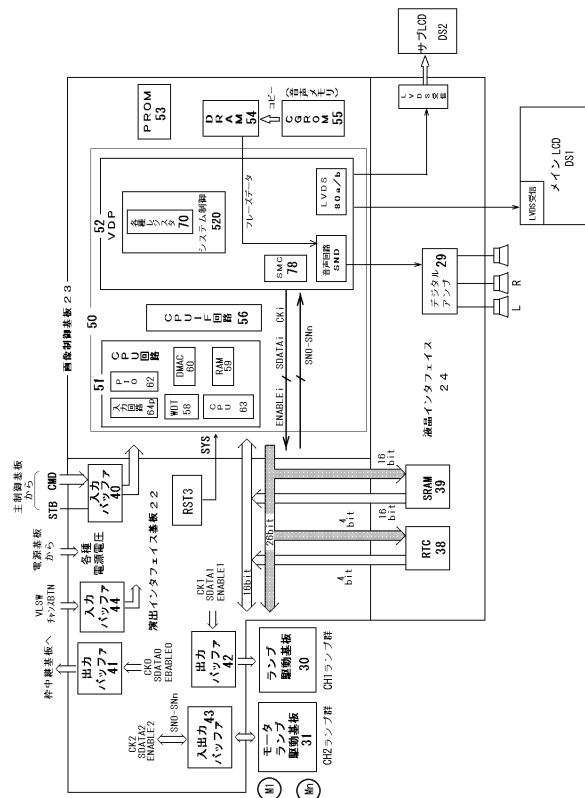
【図 30】



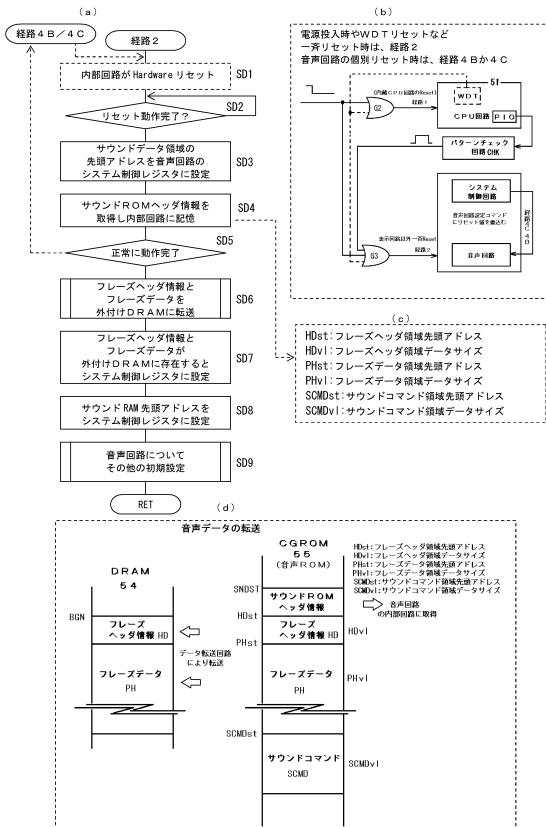
【図 3 1】



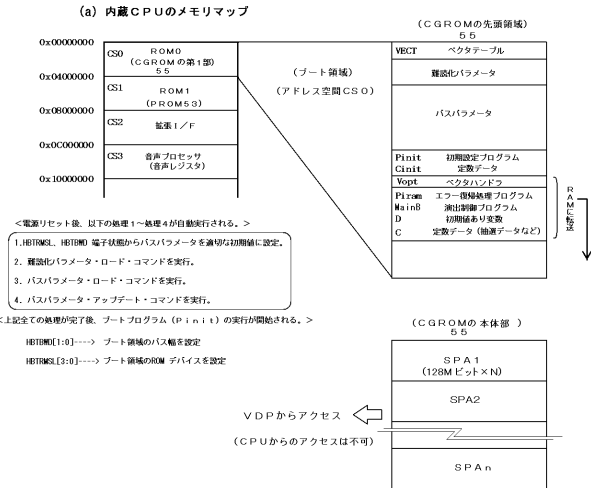
【圖 3 2】



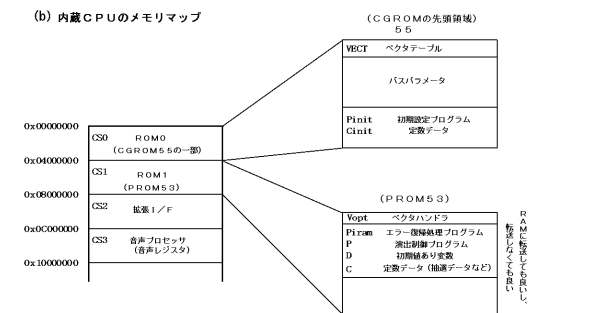
【図 3 3】



【図 3 4】

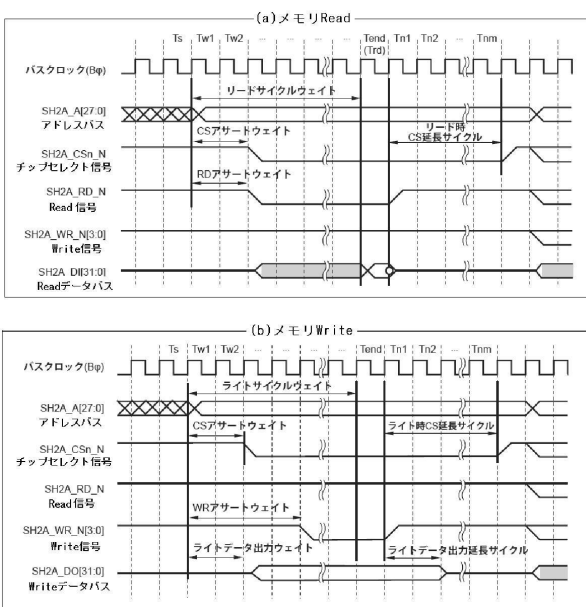


10



20

【図 3 5】



30

40

50

---

フロントページの続き

- (56)参考文献      特開２０１５－０１３１７９（ＪＰ，Ａ）  
                    特開２０１２－１８７２２２（ＪＰ，Ａ）  
                    特開２００３－０４４３５８（ＪＰ，Ａ）  
                    特開平１１－０６５９２５（ＪＰ，Ａ）  
                    特開２００７－３１７２３２（ＪＰ，Ａ）  
                    特開２００２－１４９４８８（ＪＰ，Ａ）  
                    特開平０９－３１９６５４（ＪＰ，Ａ）
- (58)調査した分野 (Int.Cl., DB名)  
                    A 63 F      7／02