

(19) 日本国特許庁(JP)

## (12) 特許公報(B2)

(11) 特許番号

特許第6724316号  
(P6724316)

(45) 発行日 令和2年7月15日(2020.7.15)

(24) 登録日 令和2年6月29日(2020.6.29)

(51) Int.Cl.

F 1

G 1 O H 7/02 (2006.01)  
G 1 O H 1/24 (2006.01)G 1 O H 7/02  
G 1 O H 1/24

請求項の数 8 (全 28 頁)

(21) 出願番号	特願2015-175809 (P2015-175809)
(22) 出願日	平成27年9月7日 (2015.9.7)
(65) 公開番号	特開2017-53898 (P2017-53898A)
(43) 公開日	平成29年3月16日 (2017.3.16)
審査請求日	平成30年8月30日 (2018.8.30)

(73) 特許権者	000001443 カシオ計算機株式会社 東京都渋谷区本町1丁目6番2号
(74) 代理人	100074099 弁理士 大菅 義之
(74) 代理人	100121083 弁理士 青木 宏義
(74) 代理人	100138391 弁理士 天田 昌行
(72) 発明者	佐藤 博毅 東京都羽村市栄町3丁目2番1号 カシオ 計算機株式会社羽村技術センター内

審査官 岩田 淳

最終頁に続く

(54) 【発明の名称】波形書き込み装置、方法、プログラム、及び電子楽器

## (57) 【特許請求の範囲】

## 【請求項 1】

楽音の音色毎に設けられ、少なくとも音域及びベロシティ域の一方が異なる複数のグループそれぞれに対応した音色波形データを記憶している二次記憶装置から、前記複数のグループそれぞれに対応する音色波形データを順次読み出し、前記読み出した音色波形データを一次記憶装置に順次書き込む書き込み処理と、

演奏情報が入力された際に前記演奏情報が示す音高及びベロシティが属するグループに對応する音色波形データの前記一次記憶装置への書き込みが完了していない場合は、前記一次記憶装置への書き込みが完了している他のグループに對応する音色波形データを読み出す読み出し処理と、を実行する処理部、10

を備えた波形書き込み装置。

## 【請求項 2】

前記処理部は、音色選択情報の入力に応答して、前記二次記憶装置から、前記入力した音色選択情報が示す音色に對応する複数の音色波形データを順次読み出し、前記読み出した音色波形データを一次記憶装置に順次書き込む書き込み処理を実行する、請求項 1 に記載の波形書き込み装置。

## 【請求項 3】

前記処理部は、前記演奏情報の入力に応答して、前記演奏情報が示す音高及びベロシティが属するグループに對応する音色波形データが前記一次記憶装置に書き込みが完了されているか否か判別し、前記書き込みが完了していると判別された場合は、前記演奏情報が20

示す音高及びペロシティが属するグループに対応する音色波形データを前記一次記憶装置から読み出すとともに、前記書き込みが完了していないと判別された場合は、前記一次記憶装置への読み込みが完了している他のグループに対応する音色波形データを読み出す処理を実行する、請求項1又は2に記載の波形書き込み装置。

**【請求項4】**

前記処理部が実行する前記読み出し処理において、前記他のグループに対応する音色波形データは、前記入力された演奏情報が示す音高より高い音高が属するグループに対応する音色波形データ及び前記入力された演奏情報が示すペロシティより大きいペロシティが属するグループに対応する音色波形データのいずれか一方である、請求項1乃至3のいずれかに記載の波形書き込み装置。 10

**【請求項5】**

前記一次記憶装置は、ランダムアクセスメモリを備え、前記二次記憶装置は、前記一次記憶装置より容量の大きいフラッシュメモリを備えた、請求項1乃至4のいずれかに記載の波形書き込み装置。

**【請求項6】**

処理部を有する波形書き込み装置に用いられる波形書き込み方法であって、前記処理部が、

楽音の音色毎に設けられ、少なくとも音域及びペロシティ域の一方が異なる複数のグループそれぞれに対応した音色波形データを記憶している二次記憶装置から、前記複数のグループそれぞれに対応する音色波形データを順次読み出し、前記読み出した音色波形データを一次記憶装置に順次書き込み。 20

前記演奏情報が入力された際に前記演奏情報が示す音高及びペロシティが属するグループに対応する音色波形データの前記一次記憶装置への書き込みが完了していない場合は、前記一次記憶装置への書き込みが完了している他のグループに対応する音色波形データを読み出す、

波形読み込み方法。

**【請求項7】**

波形書き込み装置として用いられるコンピュータに、

楽音の音色毎に設けられ、少なくとも音域及びペロシティ域の一方が異なる複数のグループそれぞれに対応した音色波形データを記憶している二次記憶装置から、前記複数のグループそれぞれに対応する音色波形データを順次読み出し、前記読み出した音色波形データを一次記憶装置に順次書き込む書き込みステップと。 30

前記演奏情報が入力された際に前記演奏情報が示す音高及びペロシティが属するグループに対応する音色波形データの前記一次記憶装置への書き込みが完了していない場合は、前記一次記憶装置への書き込みが完了している他のグループに対応する音色波形データを読み出す読み出しステップと、

を実行させるプログラム。

**【請求項8】**

請求項1乃至5のいずれかに記載の波形書き込み装置と、

演奏情報を入力する演奏操作子と、

複数種の音色波形データが記憶された前記二次記憶装置と、

前記一次記憶装置と、

前記演奏操作子からの演奏情報により前記一次記憶装置から読み出された音色波形データに基づいた音色の楽音を生成する音源と、

を有する電子楽器。 40

**【発明の詳細な説明】**

**【技術分野】**

**【0001】**

本発明は、波形書き込み装置、方法、プログラム、及びその装置を用いた電子楽器に関する。 50

**【背景技術】****【0002】**

波形読み込み方式により楽音波形を発生する音源装置では、より多数の、より長時間の波形データを利用できるようにするために、使用しない波形データはフラッシュメモリやハードディスク等の大容量の補助記憶装置(2次記憶装置)に保存しておき、使用する波形データのみを音源装置が直接アクセスできる波形メモリ(1次記憶装置)に転送して発音させるというシステムを採用するものがある。

つまり、高価な波形メモリ(1次記憶装置)が有する記憶容量以上の記憶容量の波形データを安価な補助記憶装置(2次記憶装置)に保持しておき、必要な場合のみ移動して発音に使用するというコスト的には効率的な方法であると言える。 10

**【0003】**

一従来技術として、次のような技術が知られている(例えば特許文献1に記載の技術)。この技術において、ROMには、音色毎に1つ以上の波形データが格納されている。音源LSI(大規模集積回路)は、指定された曲の曲データを参照して、それらの波形データのなかで楽音の発音に必要なものを特定し、必要と特定した波形データはそのなかで必要な部分を更に特定する。それにより、楽音の発音に必要な波形データはその必要な部分のみをROMから読み出してRAMに転送し格納させる。これにより、発音させるべき楽音の波高值生成用に波形データをワークメモリに格納する場合に、そのデータ量をより抑えられることが可能な楽音発生装置を提供するものである。 20

**【0004】**

他の従来技術として、次のような技術が知られている(例えば特許文献2に記載の技術)。この技術は、第1の記憶手段に記憶された各種データのなかで楽音発生装置の起動時に転送すべきデータ群を第2の記憶手段に転送する場合に、そのデータ群を構成する波形データのなかで予め定めた音色の波形データを優先して転送し、その転送状況に応じて、楽音発生装置の動作を制限する。例えば、同時発音可能な楽音数をより少なくする動作制御をかける。その動作制限により、そのときの転送状況で可能な動作に限定して楽音発生装置を使用させることができるようにし、転送の完了を待つ場合よりも転送開始(起動)からより短時間で演奏可能とさせるものである。 30

**【先行技術文献】****【特許文献】****【0005】**

【特許文献1】特開2007-271827号公報

【特許文献2】特許第4475323号公報

**【発明の概要】****【発明が解決しようとする課題】****【0006】**

しかしながら、上述した従来技術では依然として、補助記憶装置から波形メモリ等への波形データの移動に時間がかかるため、演奏に支障を来す場合がある。具体的には、演奏者が多数の音色の中から好みの音色を選ぶような際には、音色の切り替えの度に一定時間待たされて発音されない時間が存在するため、演奏者はストレスを感じるといった課題があった。 40

**【0007】**

そこで、本発明は、複数波形の転送を伴う音色切り替えの際に、波形メモリに所望の波形が存在しないために発音できないという無音時間を大幅に短縮することを目的とする。

**【課題を解決するための手段】****【0008】**

態様の一例では、楽音の音色毎に設けられ、少なくとも音域及びベロシティ域の一方が異なる複数のグループそれぞれに対応した音色波形データを記憶している二次記憶装置から、前記複数のグループそれぞれに対応する音色波形データを順次読み出し、前記読み出した音色波形データを一次記憶装置に順次書き込む書き込み処理と、演奏情報が入力され 50

た際に前記演奏情報が示す音高及びベロシティが属するグループに対応する音色波形データの前記一次記憶装置への書き込みが完了していない場合は、前記一次記憶装置への書き込みが完了している他のグループに対応する音色波形データを読み出す読み出し処理と、を実行する処理部、を備える。

**【発明の効果】**

**【0009】**

本発明によれば、複数波形の転送を伴う音色切り替えの際に、波形メモリに所望の波形が存在しないために発音できないという無音時間を大幅に短縮することが可能となる。

**【図面の簡単な説明】**

**【0010】**

10

【図1】本発明による電子鍵盤楽器の実施形態の外観図である。

【図2】電子鍵盤楽器の実施形態のハードウェア構成例を示す図である。

【図3】音色波形スプリットの説明図である。

【図4】フラッシュメモリ音色情報テーブルのデータ構成例、及び大容量フラッシュメモリから波形メモリへの波形データの読み動作を示す図である。

【図5】波形データの代用によりカバー範囲を拡張する動作概念の説明図である。

【図6】主要変数／定数の一覧を示す図である。

【図7】制御処理の全体処理の例を示すメインルーチンのフローチャートである。

【図8】初期化処理の詳細例を示すフローチャートである。

【図9】音色波形情報転送サブルーチンの詳細処理例を示すフローチャートである。

20

【図10】音源発音処理の詳細例を示すフローチャートである。

【図11】スプリット波形検索サブルーチンの詳細処理例を示すフローチャートである。

【図12】音色切替え処理の詳細例を示すフローチャートである。

【図13】波形読み込み処理の詳細例を示すフローチャート（その1）である。

【図14】波形読み込み処理の詳細例を示すフローチャート（その2）である。

【図15】最優先読み込み波形検索サブルーチンの詳細処理例を示すフローチャートである。

【図16】実効スプリット情報設定サブルーチンの詳細処理例を示すフローチャートである。

【図17】カバー範囲のベロシティ域上限設定ルーチンの詳細処理例を示すフローチャートである。

30

【図18】カバー範囲のベロシティ域下限設定ルーチンの詳細処理例を示すフローチャートである。

【図19】カバー範囲の鍵域上限設定ルーチンの詳細処理例を示すフローチャートである。

【図20】カバー範囲の鍵域下限設定ルーチンの詳細処理例を示すフローチャートである。

**【発明を実施するための形態】**

**【0011】**

40

以下、本発明を実施するための形態（以下「本実施形態」と記載する）について図面を参照しながら詳細に説明する。本実施形態は、例えば電子鍵盤楽器に適用される、音高（鍵域）や音量（ベロシティ：打鍵の速さ）などの演奏情報によって音色が変化するのを再現するために、大容量の2次記憶装置（例えばフラッシュメモリ）から1次記憶装置（例えばRAMによる波形メモリ）に、音高又は音量毎の波形データ（以下これを「スプリット波形」と呼ぶ）を読み込む楽音発生装置を対象とする。本実施形態は、このような楽音発生装置において、音域や強さによってスプリットされた複数波形に依って構成される音色選択が発生した際に、その音色において、鍵域や演奏の強さ（ベロシティ）などの使用頻度や、各鍵域、強さの波形の汎用性を考慮し、予め波形メモリに波形データをロードする順序を音色毎に決定しておいたり、あるいは実際の演奏を解析して各波形の使用頻度を求めておりして、その順序に従って波形メモリへの波形データの読みを行い、読み込みが完了

50

した波形を代用して、読み込み中もしくはこれから読み込みを行う波形の鍵域、強さ(ベロシティ域)を読み込みが完了するまで一時的にカバーして発音させることで、音色切替えに伴う波形入替えによる無音状態の時間を最初の波形の読み込み時間のみに短縮することを可能とする電子鍵盤楽器を実現するものである。

#### 【0012】

図1は、本発明による電子鍵盤楽器の実施形態の外観図である。本実施形態は、電子鍵盤楽器100として実施される。電子鍵盤楽器100は、演奏操作子としての複数の鍵からなる鍵盤101と、音色選択操作子としての音色選択を行うための音色選択ボタン102及び音色以外の各種機能選択を行う機能選択ボタン103からなるスイッチ・パネルと、ピッチベンドやトレモロ、ビブラート等の各種モジュレーション(演奏効果)を付加するベンダ／モジュレーション・ホイール104、音色や音色以外の各種設定情報を表示するLCD(Liquid Crystal Display：液晶ディスプレイ)105等を備える。また、電子鍵盤楽器100は、特には図示しないが、演奏により生成された楽音を放音するスピーカを裏面部、側面部、又は背面部等に備える。

#### 【0013】

音色選択ボタン102は、図1に示されるように、ピアノ(図中「Piano」)、エレクトリックピアノ(図中「E. piano」)、オルガン(図中「Organ」)、ギター(図中「Guitar」)等の各種音色のカテゴリを選択するためのボタン群である。演奏者は、この音色選択ボタン102を押下することにより、例えば16音色のうちの何れかを選択することができる。

#### 【0014】

図2は、図1の電子鍵盤楽器100の実施形態のハードウェア構成例を示す図である。図2において、電子鍵盤楽器100は、CPU(中央演算処理装置)201、ROM(リードオンリーメモリ)202、RAM(ランダムアクセスメモリ)203、大容量フラッシュメモリ204、波形メモリ206が接続される音源LSI(大規模集積回路)205、図1の鍵盤105と図1の音色選択ボタン102及び機能選択ボタン103からなるスイッチ・パネルとが接続されるキー・スキヤナ207、図1のベンダ／モジュレーション・ホイール104が接続されるA/Dコンバータ208、図1のLCD105が接続されるLCDコントローラ209、及びMIDI(Musical Instrument Digital Interface)入力を受け付けるMIDI IF(インターフェース)213が、それぞれシステムバス214に接続される構成を備える。また、音源LSI205から出力されるデジタル楽音波形データは、D/Aコンバータ208によりアナログ楽音波形信号に変換され、アンプ211で増幅された後に、特には図示しないスピーカ又は出力端子から出力される。

#### 【0015】

CPU201は、RAM203をワークメモリとして使用しながらROM202に記憶された制御プログラムを実行することにより、図1の電子鍵盤楽器100の制御動作を実行する。また、ROM202は、上記制御プログラム及び各種固定データを記憶する。

#### 【0016】

大容量フラッシュメモリ204は、波形データ等の大容量データの格納領域であり、シーケンシャルアクセスにより順次アクセスされる。一方、音源LSI205には、波形データを展開するRAMで構成される波形メモリ206が接続され、発音される楽音の波形データは必ず、この波形メモリ206上に配置されている必要がある。CPU201は、大容量フラッシュメモリ204から波形データをシーケンシャルに読みだし、それを音源LSI205経由で波形メモリ206に転送することで、音色データを入れ替えることができる。

#### 【0017】

LCDコントローラ209は、LCD105を制御するIC(集積回路)である。キー・スキヤナ207は、鍵盤105や音色選択ボタン102又は機能選択ボタン103等のスイッチ・パネルの状態を走査して、CPU201に通知するICである。A/Dコンバ

10

20

30

40

50

ータ208は、ベンダ／モジュレーション・ホール104の操作位置を検出するICである。16bitフリーランニング・タイマカウンタ212は、イベントの時刻検出のための計時を行う。

#### 【0018】

本実施形態においては、鍵域やベロシティによって音量や音高のみならず音色が変化するのを再現するために、大容量フラッシュメモリ204から波形メモリ206に、音高又は音量毎の音色の波形データが読み込まれる。このように、一つの音色に対して、音高又は音量毎に波形データを管理する手法を「音色波形スプリット」と呼び、そのように管理される波形データを「スプリット波形」と呼ぶ。図3は、音色波形スプリットの説明図である。音色波形スプリットにおいては、図3(a)に例示されるように、図1の鍵盤105上で演奏者が演奏する鍵域(図3(a)の横軸の「Key」)毎にそれぞれ波形データが割り当てられ、また同じ鍵域であっても演奏の強さすなわちベロシティ(図3(a)の縦軸の「Velocity」)毎でもそれぞれ波形データが割り当てられている。  
10

#### 【0019】

波形メモリ206に記憶される波形データは、図3(b)に例示される音色波形スプリット情報テーブルに基づいて管理される。大容量フラッシュメモリ204から波形メモリ206に或る音色の波形データが演奏されるときに、CPU201が例えば図2のROM202からその音色に対応する音色波形スプリット情報テーブルのデータを読み出して音源LSI205経由で波形メモリ206内の波形データが記憶される領域とは別の管理領域に転送される。音色波形スプリット情報テーブルには、1つの音色に含まれる各波形データ毎に、その波形データの「波形番号」と、その波形データが発音されるべき鍵域及びベロシティの範囲を示す、「最低ベロシティ」、「最大ベロシティ」、「最低キーナンバ(最低キー番号)」、及び「最高キーナンバ(最高キー番号)」と、波形メモリ206に転送された1つの音色の記憶領域の先頭からのアドレスを示す「音色先頭からのアドレス」(「H」は16進数値を示す)と、その波形データのデータサイズを示す「波形サイズ」(「H」は16進数値を示す)の各項目値からなるエントリが登録される。音源LSI205は、鍵盤105上で演奏された鍵のキーナンバとベロシティを、波形メモリ206内の音色波形スプリット情報テーブルの各エントリの「最低ベロシティ」、「最大ベロシティ」、「最低キー(音高)番号」、及び「最高キー(音高)番号」の各項目の値と比較することにより、上記演奏された鍵のキーナンバとベロシティに合致するエントリを探し出す。そして、音源LSI205は、波形メモリ206に転送された対象となる音色の記憶領域の先頭から、上記探し出したエントリの「音色先頭からのアドレス」項目の値だけ進んだアドレスから、そのエントリの「波形サイズ」項目の値が示すサイズ分の波形データを演奏で指定された鍵の音高に対応する速度で読み出し、その読み出した波形データに対して演奏で指定されたベロシティの振幅エンベロープを付加し、その結果得られる波形データを出力楽音波形データとして出力する。  
20  
30

#### 【0020】

1つの音色に含まれる各波形データは、大容量フラッシュメモリ204上では、図3(c)に示されるように、1本の連続したデータの塊として記録されている。その配置状態は、その音色に対応してROM202に記憶されている図3(b)に例示される音色波形スプリット情報テーブルで定義される。すなわち、1つの音色に含まれる各波形データの配置状態は、大容量フラッシュメモリ204及び波形メモリ206ともに同じであり、1つの音色波形スプリット情報テーブルで管理される。  
40

#### 【0021】

音色毎の波形データの数は定められていない(可変である)が、最大数と合計の最大サイズは定められている。本実施例では、図3(c)の音色波形サイズ、図3(b)の音色波形スプリット情報テーブルに例示されるように、1つの音色を構成する波形の最大数は例えば32、合計の最大サイズは例えば1MB(メガバイト)である。図3(c)の例では、1つの音色が11個の波形データから構成されていることが示されている。

#### 【0022】

図4(a)は、フラッシュメモリ音色情報テーブルのデータ構成例を示す図である。このフラッシュメモリ音色情報テーブルは、図2のROM202に記憶されている。図4(a)に例示されるように、フラッシュメモリ音色情報テーブルの各音色毎のエントリには、音色番号を示す「音色番号」項目値と、「音色名」項目値と、大容量フラッシュメモリ204の波形記憶領域の先頭からの記憶アドレスのオフセット(16進数)を示す「波形アドレス」項目値と、その音色に含まれる波形データ群の全体の波形サイズ(16進数)を示す「波形サイズ」項目値が記憶されている。

#### 【0023】

発音を行うためには、CPU201が波形データを直接アクセスできる音源LSI205の波形メモリ206に読み込んでおく必要があるが、この波形メモリ206上の各波形エリアは、16音色のうちの何れか1音色の全波形データ(最大で32波形データ)を読み込むことができる容量、すなわち1メガバイトを有する。各波形データが波形メモリに読み込まれる順序は後述するように状況に応じて最適化されるが、最終的にすべての波形データが読み込まれた状態での各波形データの並び方は大容量フラッシュメモリ204上の並びと同じである。図4(b)は、大容量フラッシュメモリ204(図中では「Flashメモリ」と表示)から波形メモリ206への波形データの読み込み動作を示す図である。押鍵時にはスプリットを判断してどれか1つの波形データの読み出しによる発音を行うが、その波形データが波形メモリ206への読み込み転送の最中である可能性もあり、必ずしも所望の波形データが波形メモリ206に読み込まれているとは限らない。本来、もし演奏の鍵域や強さに対応する波形が音源の波形メモリに存在しない場合は発音を行うことが出来ない。

#### 【0024】

本実施形態では、波形メモリ206上に既に読みが完了した波形データだけで一時的にスプリットのカバー範囲が拡張され、全域の演奏に対応でき、必要な波形データが徐々に波形メモリ206上に揃うに従って、そのカバー範囲が狭められてゆき、最終的に本来想定されているカバー範囲に設定されるという制御が行われる。スプリット波形は、波形読み込みの途中段階では、以下のようなルールに基づいて想定範囲以上をカバーする。

#### 【0025】

・横軸が鍵域(Key)、縦軸が強さ(Velocity)という2次元の平面を考えた時、ある鍵域は常に縦割りの長方形で表すことができ、その中に複数の強さの領域を持つことができる。逆に強さの分割領域は1つの鍵域の中でしか横割りにはできない。

#### 【0026】

・波形メモリに鍵域の高い波形と低い波形が読み込まれている場合は、より高い波形が低い波形の範囲をカバーする(高音優先)。

#### 【0027】

・波形メモリに同じ鍵域で弱い波形と強い波形が読み込まれている場合は、より強い波形が弱い波形の範囲をカバーする(強音優先)。

#### 【0028】

図5は、上述の3つのルールに基く波形データの代用によりカバー範囲を拡張する動作概念の説明図である。まず本来、図3(a)で示されるようなスプリット配置を想定して設計された音色波形があるとすると、4C、2B、3B、4A、5Aのスプリット波形の順に波形メモリへの読み込みを行った場合、下記の様なカバー範囲が実現される。

#### 【0029】

(A) 4Cが読み込まれた状態

- ・4Cのスプリット波形だけで全領域をカバー。

#### 【0030】

(B) 4C、2Bが読み込まれた状態

- ・2Bによって、2Bより下の鍵域を全てカバー。

- ・4Cによって、4Cより上の鍵域と、2Bと4Cとの間の鍵域を全てカバー。

#### 【0031】

10

20

30

40

50

- ( C ) 4 C、2 B、3 B が読み込まれた状態
- ・ 2 B によって、2 B より下の鍵域を全てカバー。
  - ・ 3 B によって、3 B の鍵域を全てカバー。
  - ・ 4 C によって、4 C より上の鍵域を全てカバー。

**【 0 0 3 2 】**

- ( d ) 4 C、2 B、3 B、4 A が読み込まれた状態
- ・ 2 B によって、2 B より下の鍵域を全てカバー。
  - ・ 3 B によって、3 B の鍵域を全てカバー。
  - ・ 4 C によって、4 A より上の強さで、4 C より上の鍵域を全てカバー。
  - ・ 4 A によって、4 A の鍵域のみをカバー。

10

**【 0 0 3 3 】**

- ( E ) 4 C、2 B、3 B、4 A、5 A が読み込まれた状態
- ・ 2 B によって、2 B より下の鍵域を全てカバー。
  - ・ 3 B によって、3 B の鍵域を全てカバー。
  - ・ 4 C によって、4 A より上の強さで、4 C の鍵域を全てカバー。
  - ・ 4 A によって、4 A の鍵域のみをカバー。
  - ・ 5 A によって、5 A の鍵域を全てカバー。

**【 0 0 3 4 】**

本実施形態における波形データの読み出し順序は、例えば波形データの代用に依る音色の劣化が最小限と成るようなスプリット配置を考慮して、音色毎に最適値を固定的に持たせ、これに従うこととする。

20

**【 0 0 3 5 】**

また、本実施形態では、演奏やユーザインターフェースの処理が滞らないように波形読み込み処理がバックグラウンドで実行されている。前述の「読み込み優先度」情報は、この波形読み込み処理によって渡されて、この順序に従って大容量フラッシュメモリ 204 上の波形データが音源 LSI 205 の波形メモリ 206 に転送される。

**【 0 0 3 6 】**

図 6 は、図 2 の CPU 201 が実行する制御処理において使用する主要な変数 / 定数の一覧を示す図であり、図 6 ( a ) は RAM 203 上の主要な変数の一覧、図 6 ( b ) は ROM 202 上の主要な定数の一覧である。各変数 / 定数については、後述する制御処理の詳細説明において詳述する。

30

**【 0 0 3 7 】**

以下に、上述の動作を実現するために CPU 201 が実行する制御処理の詳細例について説明する。

**【 0 0 3 8 】**

図 7 は、CPU 201 が実行する制御処理の全体処理の例を示すメインルーチンのフローチャートである。この処理例は、CPU 201 が ROM 202 に記憶された制御プログラムを実行する処理例である。

**【 0 0 3 9 】**

CPU 201 は、まず RAM 203 の内容を初期化した後 (ステップ S701)、ステップ S702 から S711 の一連の処理で示される定常ループ処理に入る。

40

**【 0 0 4 0 】**

定常ループ処理では、CPU 201 はまず、ユーザインターフェース処理 (図中「ユーザ I / F」と表示) を実行する (ステップ S702)。ここでは、CPU 201 は、図 2 のキー・スキャナ 207 を介して図 1 の音色選択ボタン 102 の状態を取得する。

**【 0 0 4 1 】**

次に、CPU 201 は、ステップ S702 の処理の結果、演奏者が音色選択ボタン 102 を操作することにより音色選択イベントが発生したか否かを判定する (ステップ S703)。

**【 0 0 4 2 】**

50

そして、CPU201は、音色切り替えが発生した場合（ステップS703の判定がYESの場合）は、音色切替え処理を実行する（ステップS704）。ステップS703の判定がNOならば、CPU201は、ステップS704の処理はスキップする。

#### 【0043】

次に、CPU201は、鍵盤処理を実行する（ステップS705）。ここでは、CPU201は、図2のキー・スキャナ207を介して図1の鍵盤105の押鍵状態を取得する。

#### 【0044】

次に、CPU201は、ステップS705の処理の結果、演奏者が鍵盤105上の何れかの鍵を押鍵することにより押鍵イベントが発生したか否かを判定する（ステップS706）。

#### 【0045】

そして、CPU201は、押鍵イベントが発生した場合（ステップS706の判定がYESの場合）は、音源発音処理を実行する（ステップS707）。ステップS706の判定がNOならば、CPU201は、ステップS707の処理はスキップする。

#### 【0046】

続いて、CPU201は、ステップS705の処理の結果、演奏者が鍵盤105上の何れかの押鍵中の鍵を離鍵することにより離鍵イベントが発生したか否かを判定する（ステップS708）。

#### 【0047】

そして、CPU201は、離鍵イベントが発生した場合（ステップS708の判定がYESの場合）は、音源リリース処理を実行する（ステップS709）。ステップS708の判定がNOならば、CPU201は、ステップS709の処理はスキップする。

#### 【0048】

その後、CPU201は、音源定常サービス処理を実行する（ステップS710）。ここでは、例えば図1の機能選択ボタン103が押された場合に対応する処理や、図1のペンドル/モジュレーション・ホール104が操作された場合に対応する処理等の、電子鍵盤楽器100に対する一般的な処理が実行される。

#### 【0049】

最後に、CPU201は、波形読み込み処理を実行する（ステップS711）。

#### 【0050】

その後、CPU201は、ステップS702の定常ループ処理の先頭に戻る。

#### 【0051】

図8は、図7のステップS701の初期化処理の詳細例を示すフローチャートである。

#### 【0052】

まず、CPU201は、RAM203上の図6に示されるメモリバックアップされない「変数」（「メモリバックアップ」項目に「×」が記載されている変数）の値を、全て「0」にする（以上、ステップS801）。

#### 【0053】

次に、CPU201は、音色波形情報転送サブルーチンの処理を実行する（ステップS802）。図9は、図8のステップS802の音色波形情報転送サブルーチンの詳細処理例を示すフローチャートである。この処理では、ROM202に予め記憶されている図3（b）に例示される音色波形スプリット情報テーブルの各項目値（図6（b）のROM\_WWM\_ST構造体参照）が、変数Wが示す現在の波形番号値に対応するRAM203上の上記各項目値を格納する構造体要素に転送される。

#### 【0054】

CPU201はまず、読み込み音色の波形の大容量フラッシュメモリ204上での格納先頭アドレス値を保持するRAM203上の変数FLASH\_OFFSET（図6（a）参照）に、メモリバックアップされているRAM203上の変数C\_TONEが示す現在の音色番号に対応し、ROM202に記憶されている読み込み音色の波形の大容量フラッシュ

10

20

30

40

50

メモリ204上での格納先頭アドレス値ROM\_WAVE\_OFFSET[C\_TONE]（図6（b）の「ROM\_WAVE\_OFFSET」参照）を格納する（ステップS901）。

#### 【0055】

次に、CPU201は、波形メモリ206上の波形エリア毎の波形番号カウント用のRAM203上の変数Wの値について、ステップS902で「0」にリセットした後、ステップS904で+1ずつインクリメントしながら、ステップS905で32に達したと判定するまで、変数Wが示す現在の波形番号値に対応する図3（b）に例示される音色波形スプリット情報テーブルの各項目値を格納する各構造体要素に、ROM202に予め記憶されている各項目値を格納する処理（ステップS903）を、繰り返し実行する。即ち、変数Wが示す波形番号値の波形データが本来発音する範囲の最小ベロシティが格納される構造体要素WM\_ST[W].ORG\_VEL\_LOに、メモリバックアップされているRAM203上の変数C\_TONEが示す現在の音色番号と変数Wが示す現在の波形番号値に対応し、ROM202に記憶されている「最低ベロシティ」値ROM\_WM\_ST[C\_TONE][W].VEL\_LOの値が格納される。また、変数Wが示す波形番号値の波形データが本来発音する範囲の最大ベロシティが格納される構造体要素WM\_ST[W].ORG\_VEL\_HIに、変数C\_TONEが示す現在の音色番号と変数Wが示す現在の波形番号値に対応し、ROM202に記憶されている「最高ベロシティ」値ROM\_WM\_ST[C\_TONE][W].VEL\_HIの値が格納される。また、変数Wが示す波形番号値の波形データが本来発音する範囲の最低キーナンバが格納される構造体要素WM\_ST[W].KEY\_LOに、変数C\_TONEが示す現在の音色番号と変数Wが示す現在の波形番号値に対応し、ROM202に記憶されている「最低キーナンバ」値ROM\_WM\_ST[C\_TONE][W].KEY\_LOの値が格納される。また、変数Wが示す波形番号値の波形データが本来発音する範囲の最高キーナンバが格納される構造体要素WM\_ST[W].ORG\_KEY\_HIに、変数C\_TONEが示す現在の音色番号と変数Wが示す現在の波形番号値に対応し、ROM202に記憶されている「最高キーナンバ」値ROM\_WM\_ST[C\_TONE][W].KEY\_HIの値が格納される。更に、構造体要素WM\_ST[W].OFFSET\_ADDRESSに、変数C\_TONEが示す現在の音色番号と変数Wが示す現在の波形番号値に対応し、ROM202に記憶されている波形メモリ206に転送された1つの音色の記憶領域の先頭からのアドレスを示す「音色先頭からのアドレス」値ROM\_WM\_ST[C\_TONE][W].OFFSETの値が格納される。加えて、構造体要素WM\_ST[W].WAVE\_SIZEに、変数C\_TONEが示す現在の音色番号と変数Wが示す現在の波形番号値に対応し、ROM202に記憶されている波形メモリ206に転送された波形データのデータサイズを示す「波形サイズ」値ROM\_WM\_ST[C\_TONE][W].WAVE\_SIZEの値が格納される。また、図3（b）には記載していないが、構造体要素WM\_ST[W].PRIに、変数C\_TONEが示す現在の音色番号と変数Wが示す現在の波形番号値に対応し、ROM202に記憶されている波形メモリ206に転送される波形データの読み込み優先度値ROM\_WM\_ST[C\_TONE][W].PRIの値が格納される。

#### 【0056】

以上のステップS902からS905までの一連の繰り返し動作が、32の波形データについて繰り返し実効された後、CPU201は、図9のフローチャートで示される図8のステップS802の音色波形情報転送サブルーチンの処理を終了する。

#### 【0057】

ステップS802の後、CPU201は、現在読み込み中の波形の番号を示す変数READING\_WAVE（図6（a）参照）に、無効値「-1」を格納する（ステップS803）。その後、CPU201は、図8のフローチャートで例示される図7のステップS701の初期化処理を終了する。

#### 【0058】

図10は、図7のステップS707の音源発音処理の詳細例を示すフローチャートであ

10

20

30

40

50

る。

**【0059】**

CPU201はまず押鍵時に、スプリット波形検索サブルーチンにより、押鍵に対応する鍵域及びベロシティに基づいて、現在選択されている音色中でその押鍵により発音されるべき波形データの波形番号を割り出し、その波形番号をRAM203上の変数Wに格納する（ステップS1001）。

**【0060】**

次に、CPU201は、変数Wが示す現在の波形番号値に対応する波形メモリ206上での波形の有無を示す要素WM\_ST[W].WAVE（図6（a）参照）の値が「1」であるか否か、すなわち、変数Wが示す波形番号に対応する波形データが波形メモリ206の波形エリアに存在するかどうかを判定する（ステップS1002）。

10

**【0061】**

CPU201は、ステップS1002の判定がYesならば、その波形データに対する発音処理を、音源LSI205に対して指示する発音処理を実行する（ステップS1003）。CPU201は、ステップS1002の判定がNoならば、ステップS1003の発音処理はスキップする。

**【0062】**

図11は、図10のステップS1001のスプリット波形検索サブルーチンの詳細処理例を示すフローチャートである。

20

**【0063】**

CPU201はまず、波形メモリ206上の波形エリア毎の波形番号カウント用のRAM203上の変数Wについて、ステップS1101で波形番号カウンタWに値「0」を格納した後、ステップS1106で+1ずつインクリメントしながら、ステップS1107で32に達したと判定するまで、ステップS1102からS1105までの一連の処理を、繰り返し実行する。

30

**【0064】**

これらの一連の処理において、CPU201はまず、変数Kに格納された現在押鍵中のキーナンバの値が、変数Wが示す現在の波形番号値に対応する最低キーナンバ要素値WM\_ST[W].KEY\_LO（図6（a）参照）の値以上であるか否かを判定する（ステップS1102）。ここで、最低キーナンバ要素値WM\_ST[W].KEY\_LOとしては、変数C\_TONEが示す現在選択されている音色に対応するROM202内の音色波形スプリット情報テーブル（図3（b）参照）の「最低キーナンバ」項目値が予めコピーされている（前述した図8のステップS802内の図9のステップS903参照）。ステップS1102の判定がNoならば、変数Wが示す波形番号の波形データは現在押鍵中の鍵域及びベロシティに対応するスプリット波形ではないため、ステップS1106に移行する。

**【0065】**

ステップS1102の判定がYesならば、CPU201は、変数Kに格納された現在押鍵中のキーナンバの値が、変数Wが示す現在の波形番号値に対応する最高キーナンバ要素値WM\_ST[W].KEY\_HI（図6（a）参照）の値以下であるか否かを判定する（ステップS1103）。ここで、最高キーナンバ要素値WM\_ST[W].KEY\_HIとしては、変数C\_TONEが示す現在選択されている音色に対応するROM202内の音色波形スプリット情報テーブル（図3（b）参照）の「最高キーナンバ」項目値が予めコピーされている（前述した図8のステップS802内の図9のステップS903参照）。ステップS1103の判定がNoならば、変数Wが示す波形番号の波形データは現在押鍵中の鍵域及びベロシティに対応するスプリット波形ではないため、ステップS1106に移行する。

40

**【0066】**

ステップS1103の判定がYesならば、CPU201は、変数Vに格納された現在押鍵中のベロシティの値が、変数Wが示す現在の波形番号値に対応する最低ベロシティ要

50

素値WM\_ST[W].VEL\_L\_O(図6(a)参照)の値以上であるか否かを判定する(ステップS1104)。ここで、最低ベロシティ要素値WM\_ST[W].VEL\_L\_Oとしては、変数C\_TONEが示す現在選択されている音色に対応するROM202内の音色波形スプリット情報テーブル(図3(b)参照)の「最低ベロシティ」項目値が予めコピーされている(前述した図8のステップS802内の図9のステップS903参照)。ステップS1104の判定がNoならば、変数Wが示す波形番号の波形データは現在押鍵中の鍵域及びベロシティに対応するスプリット波形ではないため、ステップS1106に移行する。

#### 【0067】

ステップS1104の判定がYesならば、CPU201は、変数Vに格納された現在押鍵中のベロシティの値が、変数Wが示す現在の波形番号値に対応する最高ベロシティ要素値WM\_ST[W].VEL\_HI(図6(a)参照)の値以下であるか否かを判定する(ステップS1105)。ここで、最高ベロシティ要素値WM\_ST[W].VEL\_HIとしては、変数C\_TONEが示す現在選択されている音色に対応するROM202内の音色波形スプリット情報テーブル(図3(b)参照)の「最高ベロシティ」項目値が予めコピーされている(前述した図8のステップS802内の図9のステップS903参照)。ステップS1105の判定がNoならば、変数Wが示す波形番号の波形データは現在押鍵中の鍵域及びベロシティに対応するスプリット波形ではないため、ステップS1106に移行する。

#### 【0068】

ステップS1105の判定がYesならば、現在押鍵中の鍵域が変数Wが示す波形番号の波形データに設定されている最低キーナンバ以上最高キーナンバ以下の範囲にあり、かつ現在押鍵中のベロシティが上記波形データに設定されている最低ベロシティ以上最高ベロシティ以下の範囲にあり、現在の押鍵に対して発音処理されるべき波形データである。このため、CPU201は、変数Wが示す波形番号を戻り値として、図11のフローチャートで示される図10のステップS1001のスプリット波形検索サブルーチンを終了する。なお、このサブルーチンの戻り値は波形番号Wであるが、理論的にはスプリットの設定によっては発音しない鍵域を残すことも可能であり、その場合は波形番号として例えれば32を返す。

#### 【0069】

図12は、図7のステップS704の音色切替え処理の詳細例を示すフローチャートである。ここで、RAM203上の変数Tに演奏者によって選択された音色番号が設定されているとする。

#### 【0070】

CPU201はまず、変数Tに得られた音色番号が、変数C\_TONEに得られている現在の音色番号に一致するか否かを判定する(ステップS1201)。ステップS1201の判定がYesならば、音色を切り替える必要はないため、そのまま図12のフローチャートで示される図7のステップS704の音色切替え処理を終了する。

#### 【0071】

ステップS1201の判定がNoならば、CPU201はまず、図2の音源LSI205に対して現在の発音を停止させる(ステップS1202)。

#### 【0072】

次に、CPU201は、変数C\_TONEに、変数Tに新たに得られた音色番号の値を格納する(ステップS1203)。

#### 【0073】

次に、CPU201は、波形メモリ206上の波形エリア毎の波形番号カウント用のRAM203上の変数Wの値について、ステップS1204で「0」にリセットした後、ステップS1206で+1ずつインクリメントしながら、ステップS1207で32に達したと判定するまで、変数Wが示す現在の波形番号値に対応する波形メモリ206上での波形の有無を示す要素WM\_ST[W].WAVE(図6(a)参照)に無効値「0」を格

10

20

30

40

50

納して初期化する処理（ステップS1205）を、繰り返し実行する。

#### 【0074】

その後、CPU201は、図7のステップS701の初期化処理における図8のステップS802及びS803と同様の、音色波形情報転送サブルーチンの処理（ステップS1208）を実行する。ステップS1208の処理により、新たな音色に対して、ROM202上の音色波形スプリット情報テーブルの各項目値がRAM203に転送される。また、CPU201は、図7のステップS701の初期化処理における図8のステップS803と同様の、変数READING\_WAVEに無効値「-1」を格納する処理（ステップS1209）を実行する。

#### 【0075】

その後、CPU201は、図12のフローチャートで示される図7のステップS704の音色切替え処理を終了する。

#### 【0076】

図13及び図14は、図7のステップS711の波形読み込み処理の詳細例を示すフローチャートである。演奏による発音処理を止めないようにするために、図7に例示したメインループにおいて、ステップS711で波形読み込み処理だけに処理を集中させず、他の処理も並行に処理するために、ステップS711で波形読み込み処理の1回の実行においては、一定の容量のメモリ転送が完了した時点で処理を中断し、他のメインループの処理を行って再び波形転送プログラムが起動された時に処理が再開される。

#### 【0077】

CPU201はまず、RAM203上の変数である現在読み込み中の波形の番号READING\_WAVEの値が無効値「-1」であるか否か、すなわち、現在波形データの読み込みが行われていないか否かを判定する（図13のステップS1301）。

#### 【0078】

現在波形データの読み込みが行われていない（ステップS1301の判定がYesである）場合には、CPU201は、最優先読み込み波形検索サブルーチンを実行する（図13のステップS1302）。この処理では、RAM203の読み込み優先度を格納する構造体要素WP\_ST[W].PRI(0 W 31)（図5(b)及び図6(a)参照）の値が順次調べられ、波形転送が必要かつ、完了していないものの中で最も優先順位の高い波形データが選定される。図15は、図13のステップS1302の最優先読み込み波形検索サブルーチンの詳細処理例を示すフローチャートである。

#### 【0079】

図15のフローチャートにおいて、CPU201はまず、読み込み優先度の値が格納される変数Pの値を、ステップS1205で最低優先度より一つ低い値「32」に初期化する。また、CPU201は、波形番号カウンタとして動作するRAM203上の変数Xの値を「0」に初期化する。更に、CPU201は、最高優先度の波形番号を格納する変数Wとする（ステップS1501）。

#### 【0080】

その後、CPU201は、変数Xの値について、上述のようにステップS1501で「0」にリセットした後、ステップS1505で+1ずつインクリメントしながら、ステップS1506で32に達したと判定するまで、ステップS1502からS1504までの一連の処理を、繰り返し実行する。

#### 【0081】

これらの一連の処理において、CPU201はまず、変数Xが示す波形番号に対応する構造体WM\_ST[X].WAVE（図6(a)参照）の値が0であるか否か、即ち、変数Xが示す波形番号の波形データが波形メモリ206上に存在しないか否かを判定する（ステップS1502）。

#### 【0082】

当該波形データが波形メモリ206上に存在しない場合（ステップS1502の判定がYesの場合）、CPU201は、変数Xが示す波形番号に対応する構造体WM\_ST[

10

20

30

40

50

X] . P R I の値が変数 P の値よりも小さいか否か、即ち、変数 X が示す波形番号の波形データの読み込み優先度が、変数 P に設定されている現時点で最も高い優先度よりも高いか否かを判定する(ステップ S 1 5 0 3)。

#### 【0083】

ステップ S 1 5 0 3 の判定が Y e s ならば、 C P U 2 0 1 は、変数 P の値を変数 X が示す波形番号に対応する構造体 W M \_ S T [ X ] . P R I の値で置き換え、最高優先度の波形番号を格納する変数 W に変数 X が示す波形番号を格納する(ステップ S 1 5 0 4)。これにより、現在の変数 X が示す波形番号が最高優先度の波形番号として設定される。

#### 【0084】

当該波形データが波形メモリ 2 0 6 上に既に存在して新たに読み込む必要がない場合(ステップ S 1 5 0 2 の判定が N o の場合)、又は変数 X が示す波形番号の波形データの読み込み優先度が変数 P に設定されている優先度よりも高くはない場合(ステップ S 1 5 0 3 の判定が N o の場合)には、ステップ S 1 5 0 4 の処理はスキップされる。

#### 【0085】

以上のステップ S 1 5 0 2 から S 1 5 0 4 の処理が 1 X 3 1 の範囲で繰り返し実行された結果、ステップ S 1 5 0 6 の判定が Y e s になると、 C P U 2 0 1 は、図 1 5 のフローチャートで示される図 1 3 のステップ S 1 3 0 2 の最優先読み波形検索サブルーチンの処理を終了する。この結果、最終的に変数 W に、波形転送が必要かつ完了していないもの中で最も優先順位の高い波形データの波形番号が得られる。

#### 【0086】

ステップ S 1 3 0 2 の後、 C P U 2 0 1 は、現在読み込み中の波形の番号を保持する変数 R E A D I N G \_ W A V E (図 6 ( a ) 参照) に変数 W に格納されている波形番号を格納する。また、 C P U 2 0 1 は、変数 W が示す波形番号に対応する大容量フラッシュメモリ 2 0 4 から波形メモリ 2 0 6 へ前回までに転送済みの波形データのサイズを示す構造体要素 W M \_ S T [ W ] . R E A D \_ S I Z E に、未転送を示す値「 0 」を格納する(以上、図 1 3 のステップ S 1 3 0 3)。

#### 【0087】

一方、現在波形データの読み込みが継続している(ステップ S 1 3 0 1 の判定が N o である)場合には、 C P U 2 0 1 は、変数 W に変数 R E A D I N G \_ W A V E が保持する波形番号をセットする(図 1 3 のステップ S 1 3 0 4)。その後、 C P U 2 0 1 は、図 1 3 のステップ S 1 3 0 5 から図 1 4 のステップ S 1 3 0 6 以降の処理に移行し、変数 W に再度セットされた波形番号の残りの波形データを大容量フラッシュメモリ 2 0 4 から波形メモリ 2 0 6 へ転送する処理を実行する。

#### 【0088】

上記図 1 3 のステップ S 1 3 0 3 又は S 1 3 0 4 の処理の後、 C P U 2 0 1 は、 R A M 2 0 3 上の変数である今回読み込みサイズカウンタ変数 c に初期値「 0 」をセットする。また、 C P U 2 0 1 は、大容量フラッシュメモリ 2 0 4 上の読み込みアドレスを示す R A M 2 0 3 上の変数 r p に、図 9 のステップ S 8 0 2 内の図 9 のステップ S 9 0 1 で変数 F L A S H \_ O F F S E T に格納された読み込み音色の波形の大容量フラッシュメモリ 2 0 4 上での格納先頭アドレス値をセットする。更に、 C P U 2 0 1 は、波形メモリ 2 0 6 上の書き込みアドレスを示す R A M 2 0 3 上の変数 w p に、図 9 のステップ S 8 0 2 内の図 9 のステップ S 9 0 3 で変数 W の波形番号に対応する構造体要素 W M \_ S T [ W ] . O F F S E T \_ A D R S に格納された波形メモリ 2 0 6 上の音色先頭からのアドレス値をセットする。加えて、 C P U 2 0 1 は、大容量フラッシュメモリ 2 0 4 から波形メモリ 2 0 6 へ今回までに転送済みの波形データのサイズを示す R A M 2 0 3 上の変数 r s に、変数 W が示す波形番号に対応する前回までに転送済みの波形データのサイズを示す構造体要素 W M \_ S T [ W ] . R E A D \_ S I Z E (後述する図 1 4 のステップ S 1 3 1 3 参照) の値を格納する(以上、図 1 3 のステップ S 1 3 0 5)。

#### 【0089】

その後、 C P U 2 0 1 は、図 1 4 のステップ S 1 3 0 6 以降の処理に移行し、変数 W に

10

20

30

40

50

得られた波形番号の波形データを大容量フラッシュメモリ204から波形メモリ206へ転送する処理を開始する。

#### 【0090】

CPU201はまず、大容量フラッシュメモリ204からの読み込み処理を実行する（図14のステップS1306）。即ち、CPU201は、図13のステップS1305で変数r\_pにセットされた大容量フラッシュメモリ204上での波形格納先頭アドレス値に、同じくステップS1305で今回までに転送済みの波形データのサイズを示す変数r\_sにセットされた変数Wが示す波形番号に対応する前回までに転送済みの波形データのサイズを加算して得られるアドレスから、1ワード分の波形データを読み込み、RAM203上の変数dに格納する。

10

#### 【0091】

次に、CPU201は、波形メモリ206への書き込み処理を実行する（図14のステップS1307）。即ち、CPU201は、図13のステップS1305で波形メモリ206上の書き込みアドレスを示す変数w\_pにセットされた音色先頭からのアドレス値に、同じくステップS1305で今回までに転送済みの波形データのサイズを示す変数r\_sにセットされた変数Wが示す波形番号に対応する前回までに転送済みの波形データのサイズを加算して得られるアドレスに、変数dに格納されている1ワード分の波形データを、音源LSI205経由で書き込む。

#### 【0092】

その後、CPU201は、今回までに転送済みの波形データのサイズを示す変数r\_sの値を+1インクリメントし、今回読み込みサイズカウンタ変数cの値を+1インクリメントする（図14のステップS1308）。

20

#### 【0093】

そして、CPU201は、今回までに転送済みの波形データのサイズを示す変数r\_sの値が、カウンタ変数Wが示す波形番号に対応する波形データのサイズ要素WM\_ST[W].WAVE\_SIZEの値に等しくなったか否かを判定する（図14のステップS1309）。

#### 【0094】

ステップS1309の判定がNoならば、CPU201は、今回読み込みサイズカウンタ変数cの値が100000バイトに等しくなったか否か、すなわち、今回転送されたサイズが100KB（キロバイト）に達したか否かを判定する（図14のステップS1310）。

30

#### 【0095】

ステップS1310の判定がNoならば、CPU201は、ステップS1306の処理に戻って転送を続行する。

#### 【0096】

ステップS1309の判定がYesになると、CPU201は、変数Wが示す現在の波形番号値に対応する波形メモリ206上での波形の有無を示す要素WM\_ST[W].WAVE（図6（a）参照）に、波形有りを示す値「1」をセットする。また、CPU201は、現在読み込み中の波形の番号を示す変数READING\_WAVE（図6（a）参照）に、無効値「-1」を格納する（以上、図14のステップS1311）。

40

#### 【0097】

続いて、CPU201は、実効スプリット情報設定サブルーチンの処理を実行する（図14のステップS1312）。CPU201は、この処理において、現在、音源LSI205の波形メモリ206に読み込まれている波形データを調査し、その波形データのスプリットカバー範囲を設定する。CPU201は、全ての波形データが揃っていない状態であっても、1つ以上の波形データが存在する場合に、全鍵域、全ベロシティ域をカバーするよう読み込まれている波形データの一時的な実効スプリットポイントを設定する。この処理の詳細については、後述する。

#### 【0098】

50

ステップ S 1312 の処理の後、又はステップ S 1310 の判定が Yes となったら、CPU201 は、変数 W が示す波形番号に対応する大容量フラッシュメモリ 204 から波形メモリ 206 へ前回までに転送済みの波形データのサイズを示す構造体要素 WM\_ST [W] . READ\_SIZE に、今回までに転送済みの波形データのサイズを示す変数 rs の値を格納する（図 14 のステップ S 1313）。

#### 【0099】

その後、CPU201 は、図 13 及び図 14 のフローチャートで示される図 7 のステップ S 711 の波形読み込み処理を終了する。

#### 【0100】

このように、本実施形態では、一定の容量（例えば 100KB）のメモリ転送が完了した時点で処理が中断され、他のメインループの処理を行って再び波形転送プログラムが起動された時に図 13 のステップ S 1302 から S 1303 を介して、処理が再開される。このとき、変数 W の波形番号に対応する前回までに転送済みのデータサイズを示す要素値 WM\_ST [W] . READ\_SIZE には、図 14 のステップ S 1313 の処理によって前回までに転送が完了したサイズが格納されているため、図 13 のステップ S 1305 、図 14 のステップ S 1306 及び S 1307 のアドレス演算処理によって、前回転送が完了したアドレスの次のアドレスから転送を再開することができる。

#### 【0101】

図 16 は、図 14 のステップ S 1312 の実効スプリット情報設定サブルーチンの詳細処理例を示すフローチャートである。CPU201 は、波形メモリ 206 上の波形エリア毎の波形番号カウント用の RAM 203 上の変数 W の値について、ステップ S 1601 で「0」にリセットした後、ステップ S 1608 で +1 ずつインクリメントしながら、ステップ S 1609 で 32 に達したと判定するまで、ステップ S 1602 から S 1607 の一連の処理を、繰り返し実行する。

#### 【0102】

これら一連の処理において、CPU201 はまず、変数 W が示す波形番号に対応する構造体 WM\_ST [W] . WAVE（図 6 (a) 参照）の値が 0 であるか否か、即ち、変数 W が示す波形番号の波形データが波形メモリ 206 上に存在しないか否かを判定する（ステップ S 1602）。

#### 【0103】

変数 W が示す波形番号の波形データが波形メモリ 206 上に存在しない（ステップ S 1602 の判定が Yes）ならば、CPU201 は、変数 W が示す現在の波形番号値に対応する最高キーナンバ要素値 WM\_ST [W] . KEY\_HI、最低キーナンバ要素値 WM\_ST [W] . KEY\_LO、最高ベロシティ要素値 WM\_ST [W] . VEL\_HI、最低ベロシティ要素値 WM\_ST [W] . VEL\_LO（図 6 (a) 参照）に、値「0」を格納する（ステップ S 1607）。この結果、前述した図 7 のステップ S 707 の音源発音処理内の図 10 のステップ S 1001 内の図 11 のスプリット波形検索サブルーチンにおいて、変数 W が示す波形番号に対応する波形エリアは選択されることはない。

#### 【0104】

変数 W が示す波形番号の波形データが波形メモリ 206 上に存在する（ステップ S 1602 の判定が No）ならば、CPU201 は、カバー範囲のベロシティ域上限設定ルーチン（ステップ S 1603）、カバー範囲のベロシティ域下限設定ルーチン（ステップ S 1604）、カバー範囲の鍵域上限設定ルーチン（ステップ S 1605）、及びカバー範囲の鍵域下限設定ルーチン（ステップ S 1606）を順次実行する。その後、CPU201 は、ステップ S 1608 に移行する。

#### 【0105】

図 17 は、図 16 のステップ S 1603 のカバー範囲のベロシティ域上限設定ルーチンの詳細処理例を示すフローチャートである。

#### 【0106】

図 17 のフローチャートにおいて、CPU201 はまず、波形メモリ 206 上の波形工

10

20

30

40

50

リア毎の波形番号カウント用のRAM203上の変数Cの値について、ステップS1701で「0」にリセットした後、ステップS1704で+1ずつインクリメントしながら、ステップ1705で32に達したと判定するまで、S1702及びS1703の処理を、繰り返し実行することにより、波形メモリ206に読み込み済みの調査対象波形よりも上のベロシティ域に配置される波形データを検索する。以下に示すように、波形データが存在すれば、変数Cの波形番号が示す比較対象の波形データが上のベロシティレンジをカバーするので拡張はしない。波形データが存在しなければ、変数Wの波形番号が示す調査対象の波形データがベロシティレンジ127までカバーする。

#### 【0107】

まずステップS1702において、CPU201は、変数Cが示す波形番号に対応する構造体WM\_ST[C].WAVE(図6(a)参照)の値が0であるか否か、即ち、変数Cが示す波形番号の波形データが波形メモリ206上に存在しないか否かを判定する(ステップS1702)。

#### 【0108】

変数Cが示す波形番号の波形データが波形メモリ206上に存在する(ステップS1702の判定がNoである)場合には、ステップS1703において、CPU201は、変数Wが示す調査対象となる波形番号の波形データが本来発音する範囲の最大ベロシティが格納される構造体要素WM\_ST[W].ORG\_VEL\_HIの値よりも、変数Cが示す比較対象となる波形番号の波形データが本来発音する範囲の最小ベロシティが格納される構造体要素WM\_ST[C].ORG\_VEL\_LOの値のほうが大きいか否かを判定する。ステップS1703の判定がNoならば、CPU201は、ステップS1704に移行して、次の比較対象について調査する。変数Cが示す波形番号の波形データが波形メモリ206上に存在しない(ステップS1702の判定がYesである)場合には、CPU201は、ステップS1703の判定処理はスキップする。なお特に図示しないが、CPU201は、C=Wの場合も、ステップS1703の判定処理はスキップする。

#### 【0109】

1つでもステップS1703の判定がYesになると、CPU201は、変数Wが示す調査対象となる波形番号の波形データが現在発音する範囲の最大ベロシティが格納される構造体要素WM\_ST[W].VEL\_HIに、変数Cが示す比較対象となる波形番号の波形データが本来発音する範囲の最大ベロシティが格納される構造体要素WM\_ST[C].ORG\_VEL\_HIの値を格納する(ステップS1707)。その後、CPU201は、図17のフローチャートで示される図16のステップS1603のカバー範囲のベロシティ域上限設定ルーチンを終了する。

#### 【0110】

ステップS1703の判定がNoのまま、全ての32の比較対象について調査が終了しステップS1705の判定がYesになると、CPU201は、変数Wが示す調査対象となる波形番号の波形データが現在発音する範囲の最大ベロシティが格納される構造体要素WM\_ST[W].VEL\_HIに、ベロシティ最大値「127」を格納する。その後、CPU201は、図17のフローチャートで示される図16のステップS1603のカバー範囲のベロシティ域上限設定ルーチンを終了する。

#### 【0111】

図18は、図16のステップS1604のカバー範囲のベロシティ域下限設定ルーチンの詳細処理例を示すフローチャートである。

#### 【0112】

図18のフローチャートにおいて、CPU201はまず、波形メモリ206上の波形エリア毎の波形番号カウント用のRAM203上の変数Cの値について、ステップS1801で「0」にリセットした後、ステップS1806で+1ずつインクリメントしながら、ステップ1807で32に達したと判定するまで、S1802からS1805の一連の処理を、繰り返し実行することにより、調査対象波形よりも下のベロシティ域に配置される波形データを検索する。以下に示すように、波形データが存在すれば、変数Cの波形番号

10

20

30

40

50

が示す比較対象の波形データの中で最も近くに配置されるベロシティ域の上限の1つ上までを変数Wの波形番号が示す調査対象の波形データがカバーする。波形データが1つも存在しなければ、変数Wの波形番号が示す調査対象の波形データがベロシティ域0までカバーする。また、CPU201は、ステップS1801において、RAM203上の変数VELO\_L0に初期値「0」を格納する。

#### 【0113】

これら一連の処理において、CPU201はまず、変数Cが示す波形番号に対応する構造体WM\_ST[C].WAVE(図6(a)参照)の値が0であるか否か、即ち、変数Cが示す波形番号の波形データが波形メモリ206上に存在しないか否かを判定する(ステップS1802)。

10

#### 【0114】

変数Cが示す波形番号の波形データが波形メモリ206上に存在する(ステップS1802の判定がNoである)場合には、CPU201は、変数Cが示す比較対象となる波形番号の波形データが本来発音する範囲の最大ベロシティが格納される構造体要素WM\_ST[C].ORG\_VEL\_HIの値よりも、変数Wが示す調査対象となる波形番号の波形データが本来発音する範囲の最小ベロシティが格納される構造体要素WM\_ST[W].ORG\_VEL\_LOの値のほうが大きいか否かを判定する(ステップS1803)。

#### 【0115】

ステップS1803の判定がYesならば、CPU201は更に、変数VELO\_L0の値よりも変数Cが示す比較対象となる波形番号の波形データが本来発音する範囲の最大ベロシティが格納される構造体要素WM\_ST[C].ORG\_VEL\_HIの値のほうが大きいか否かを判定する(ステップS1804)。

20

#### 【0116】

ステップS1804の判定もYesならば、CPU201は、変数VEL\_E\_L0に、変数Cが示す比較対象となる波形番号の波形データが本来発音する範囲の最大ベロシティが格納される構造体要素WM\_ST[C].ORG\_VEL\_HIの値を格納する(ステップS1805)。

#### 【0117】

変数Cが示す波形番号の波形データが波形メモリ206上に存在しない(ステップS1802の判定がYesである)場合、ステップS1803の判定がNoである場合、又はステップS1804の判定がNoである場合には、CPU201は、ステップS1805の処理はスキップする。なお特には図示しないが、CPU201は、C=Wの場合も、ステップS1805の判定処理はスキップする。

30

#### 【0118】

以上の調査が32の比較対象全てについて終了し、ステップS1807の判定がYesとなると、CPU201は、変数Wが示す調査対象となる波形番号の波形データが現在発音する範囲の最小ベロシティが格納される構造体要素WM\_ST[W].VEL\_L0に、変数VELO\_L0の値が格納される(ステップS1808)。その後、CPU201は、図18のフローチャートで示される図16のステップS1604のカバー範囲のベロシティ域下限設定ルーチンを終了する。

40

#### 【0119】

図19は、図16のステップS1605のカバー範囲の鍵域上限設定ルーチンの詳細処理例を示すフローチャートである。

#### 【0120】

図19のフローチャートにおいて、CPU201はまず、波形メモリ206上の波形エリア毎の波形番号カウント用のRAM203上の変数Cの値について、ステップS1901で「0」にリセットした後、ステップS1904で+1ずつインクリメントしながら、ステップ1705で32に達したと判定するまで、S1902及びS1903の処理を、繰り返し実行することにより、調査対象波形よりも上の鍵域に配置される波形データを検索する。以下に示すように、波形データが存在すれば、変数Cの波形番号が示す比較対象

50

の波形データが上のキーレンジをカバーするので拡張はしない。波形データが存在しなければ、変数Wの波形番号が示す調査対象の波形データがキーレンジ127までカバーする。

#### 【0121】

まずステップS1902において、CPU201は、変数Cが示す波形番号に対応する構造体WM\_ST[C].WAVE(図6(a)参照)の値が0であるか否か、即ち、変数Cが示す波形番号の波形データが波形メモリ206上に存在しないか否かを判定する(ステップS1902)。

#### 【0122】

変数Cが示す波形番号の波形データが波形メモリ206上に存在する(ステップS1902の判定がNoである)場合には、ステップS1903において、CPU201は、変数Wが示す調査対象となる波形番号の波形データが本来発音する範囲の最高キーナンバが格納される構造体要素WM\_ST[W].ORG\_KEY\_HIの値よりも、変数Cが示す比較対象となる波形番号の波形データが本来発音する範囲の最低キーナンバが格納される構造体要素WM\_ST[C].ORG\_KEY\_LOの値のほうが大きいか否かを判定する。ステップS1903の判定がNoならば、CPU201は、ステップS1904に移行して、次の比較対象について調査する。変数Cが示す波形番号の波形データが波形メモリ206上に存在しない(ステップS1902の判定がYesである)場合には、CPU201は、ステップS1903の判定処理はスキップする。なお特には図示しないが、CPU201は、C=Wの場合も、ステップS1903の判定処理はスキップする。

#### 【0123】

1つでもステップS1903の判定がYesになると、CPU201は、変数Wが示す調査対象となる波形番号の波形データが現在発音する範囲の最高キーナンバが格納される構造体要素WM\_ST[W].KEY\_HIに、変数Cが示す比較対象となる波形番号の波形データが本来発音する範囲の最高キーナンバが格納される構造体要素WM\_ST[C].ORG\_KEY\_HIの値を格納する(ステップS1907)。その後、CPU201は、図19のフローチャートで示される図16のステップS1605のカバー範囲の鍵域上限設定ルーチンを終了する。

#### 【0124】

ステップS1903の判定がNoのまま、全ての32の比較対象について調査が終了しステップS1905の判定がYesになると、CPU201は、変数Wが示す調査対象となる波形番号の波形データが現在発音する範囲の最高キーナンバが格納される構造体要素WM\_ST[W].KEY\_HIに、キーナンバ最大値「127」を格納する。その後、CPU201は、図19のフローチャートで示される図16のステップS1605のカバー範囲の鍵域上限設定ルーチンを終了する。

#### 【0125】

図20は、図16のステップS1606のカバー範囲の鍵域下限設定ルーチンの詳細処理例を示すフローチャートである。

#### 【0126】

図20のフローチャートにおいて、CPU201はまず、波形メモリ206上の波形エリア毎の波形番号カウント用のRAM203上の変数Cの値について、ステップS2001で「0」にリセットした後、ステップS2006で+1ずつインクリメントしながら、ステップS2007で32に達したと判定するまで、S2002からS2005の一連の処理を、繰り返し実行することにより、調査対象波形よりも下の鍵域に配置される波形データを検索する。以下に示すように、波形データが存在すれば、変数Cの波形番号が示す比較対象の波形データの中で最も近くに配置される鍵域の上限の1つ上までを変数Wの波形番号が示す調査対象の波形データがカバーする。波形データが1つも存在しなければ、変数Wの波形番号が示す調査対象の波形データが鍵域0までカバーする。また、CPU201は、ステップS2001において、RAM203上の変数KEY\_LOに初期値「0」を格納する。

10

20

30

40

50

**【 0 1 2 7 】**

これら一連の処理において、CPU201はまず、変数Cが示す波形番号に対応する構造体WM\_ST[C].WAVE(図6(a)参照)の値が0であるか否か、即ち、変数Cが示す波形番号の波形データが波形メモリ206上に存在しないか否かを判定する(ステップS2002)。

**【 0 1 2 8 】**

変数Cが示す波形番号の波形データが波形メモリ206上に存在する(ステップS2002の判定がNoである)場合には、CPU201は、変数Cが示す比較対象となる波形番号の波形データが本来発音する範囲の最高キーナンバが格納される構造体要素WM\_ST[C].ORG\_KEY\_HIの値よりも、変数Wが示す調査対象となる波形番号の波形データが本来発音する範囲の最低キーナンバが格納される構造体要素WM\_ST[W].ORG\_KEY\_LOの値のほうが大きいか否かを判定する(ステップS2003)。

10

**【 0 1 2 9 】**

ステップS2003の判定がYesならば、CPU201は更に、変数KEY\_LOの値よりも変数Cが示す比較対象となる波形番号の波形データが本来発音する範囲の最高キーナンバが格納される構造体要素WM\_ST[C].ORG\_KEY\_HIの値のほうが大きいか否かを判定する(ステップS2004)。

**【 0 1 3 0 】**

ステップS2004の判定もYesならば、CPU201は、変数KEY\_LOに、変数Cが示す比較対象となる波形番号の波形データが本来発音する範囲の最高キーナンバが格納される構造体要素WM\_ST[C].ORG\_KEY\_HIの値を格納する(ステップS2005)。

20

**【 0 1 3 1 】**

変数Cが示す波形番号の波形データが波形メモリ206上に存在しない(ステップS2002の判定がYesである)場合、ステップS2003の判定がNoである場合、又はステップS2004の判定がNoである場合には、CPU201は、ステップS2005の処理はスキップする。なお特には図示しないが、CPU201は、C=Wの場合も、ステップS2005の判定処理はスキップする。

**【 0 1 3 2 】**

以上の調査が32の比較対象全てについて終了し、ステップS2007の判定がYesとなると、CPU201は、変数Wが示す調査対象となる波形番号の波形データが現在発音する範囲の最低キーナンバが格納される構造体要素WM\_ST[W].KEY\_LOに、変数KEY\_LOの値が格納される(ステップS2008)。その後、CPU201は、図20のフローチャートで示される図16のステップS1606のカバー範囲の鍵域下限設定ルーチンを終了する。

30

**【 0 1 3 3 】**

以上説明したように、本実施形態では、複数波形に依って構成される音色選択が発生した際に、その音色において、鍵域や演奏の強さ(ベロシティ)などの使用頻度や、各鍵域、強さの波形の汎用性を考慮し、予め波形メモリ206に波形データをロードする順序を音色毎に決定しておいたり、あるいは実際の演奏を解析して各波形の使用頻度を求めたりして、その順序に従って波形メモリ206への波形データの読み込みを行い、読み込みが完了した波形を代用して、読み込み中もしくはこれから読み込みを行う波形の鍵域、強さ(ベロシティ域)を読み込みが完了するまで一時的にカバーして発音させることが可能となる。この結果として、音色切替えに伴う波形入替えによる無音状態の時間を最初の波形の読み時間のみに短縮することができる。例えば最初に読み込まれる波形データが全体の波形の10%の容量であれば、最後に読み込まれる波形データに対応した演奏を行った場合の発音可能になるまでの待ち時間は10分の1と大幅に短縮できることになる。これにより、演奏者のストレスを軽減、あるいは全く感じさせないレベルに抑制することができる。

40

**【 0 1 3 4 】**

以上の実施形態に関して、更に以下の付記を開示する。

50

## (付記 1 )

樂音の音色毎でかつ、複数種の演奏情報夫々に対応した複数種の音色波形データを記憶する二次記憶装置から、前記複数種の音色波形データを順次読み出し、前記読み出した音色波形データを二次記憶装置に順次書き込む波形書き込み処理と、

前記演奏情報が入力した際に前記演奏情報に対応する音色波形データの書き込みが完了していない場合は、前記一次記憶装置への読み込みが完了している他の音色波形データを読み出す読み出し処理と、を実行する処理部と、

を備えた波形書き込み装置。

## (付記 2 )

前記処理部は、前記書き込み処理として、音色選択情報の入力に応答して、樂音の音色毎でかつ、複数種の演奏情報夫々に対応した複数種の音色波形データを記憶する二次記憶装置から、前記入力した音色選択情報に対応する複数種の音色波形データを順次読み出し、前記読み出した音色波形データを二次記憶装置に順次書き込む処理を実行する、付記 1 に記載の波形書き込み装置。 10

## (付記 3 )

前記処理部は、前記読み出し処理として、前記演奏情報の入力に応答して、前記演奏情報に対応する音色波形データが前記二次記憶装置に書き込みが完了されているか否か判別し、前記書き込みが完了していると判別された場合は、前記演奏情報に対応する音色波形データを前記二次記憶装置から読み出すとともに、前記書き込みが完了していないと判別された場合は、前記一次記憶装置への読み込みが完了している他の音色波形データを読み出す処理を実行する、付記 1 又は 2 に記載の波形書き込み装置。 20

## (付記 4 )

前記演奏情報は、発生すべき樂音の音高情報とベロシティ情報とを少なくとも含み、

前記複数種の音色波形データ夫々は、前記音高情報及びベロシティ情報の少なくとも一方が異なる演奏情報夫々に対応する、付記 1 乃至 3 のいずれかに記載の波形書き込み装置。

## (付記 5 )

前記処理部は、前記読み出し処理において、前記他の波形データは、前記入力された演奏情報の音高情報より低い音高情報を有する演奏情報に対応する音色波形データ及び前記入力された演奏情報のベロシティ情報より小さいベロシティ情報を有する演奏情報に対応する音色波形データのいずれか一方である、付記 4 に記載の波形書き込み装置。 30

## (付記 6 )

前記二次記憶装置は、ランダムアクセスメモリを備え、前記一次記憶装置は、前記二次記憶装置より容量の大きいフラッシュメモリを備えた、付記 1 乃至 4 のいずれかに記載の波形書き込み装置。

## (付記 7 )

処理部を有する波形書き込み装置に用いられる波形書き込み方法であって、前記処理部が、

樂音の音色毎でかつ、複数種の演奏情報夫々に対応した複数種の音色波形データを記憶する二次記憶装置から、前記複数種の音色波形データを順次読み出し、前記読み出した音色波形データを二次記憶装置に順次書き込み、 40

前記演奏情報が入力した際に前記演奏情報に対応する音色波形データの書き込みが完了していない場合は、前記一次記憶装置への読み込みが完了している他の音色波形データを読み出す、波形読み込み方法。

## (付記 8 )

波形書き込み装置として用いられるコンピュータに、

樂音の音色毎でかつ、複数種の演奏情報夫々に対応した複数種の音色波形データを記憶する二次記憶装置から、前記複数種の音色波形データを順次読み出し、前記読み出した音色波形データを二次記憶装置に順次書き込む波形書き込みステップと、

前記演奏情報が入力した際に前記演奏情報に対応する音色波形データの書き込みが完了 50

していない場合は、前記一次記憶装置への読み込みが完了している他の音色波形データを読み出す読み出しステップと、  
を実行させるプログラム。

## (付記 9 )

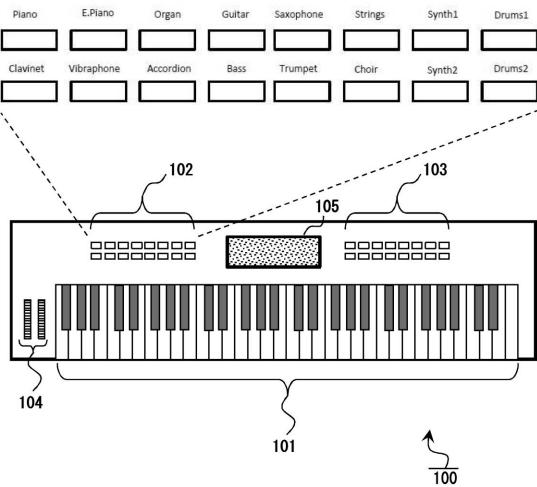
付記 1 乃至 6 のいずれかに記載の波形書き込み装置と、  
演奏情報を入力する演奏操作子と、  
複数種の音色波形データが記憶された一次記憶装置と、  
二次記憶装置と、  
前記演奏操作子からの演奏情報により前記二次記憶装置から読み出された音色波形データに基づいた音色の楽音を生成する音源と、  
を有する電子楽器。 10

## 【符号の説明】

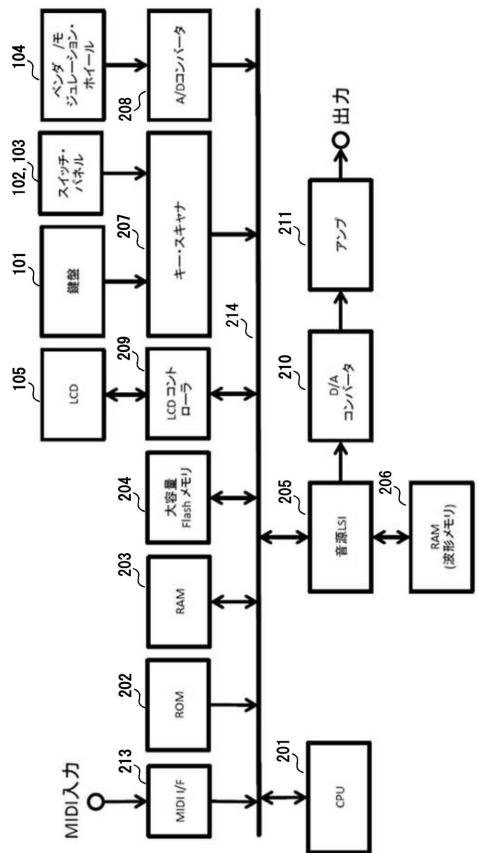
## 【0 1 3 5】

1 0 0	電子鍵盤楽器	
1 0 1	鍵盤	
1 0 2	音色選択ボタン	
1 0 3	機能選択ボタン	
1 0 4	ベンダ / モジュレーション・ホイール	
1 0 5	L C D	
2 0 1	C P U	20
2 0 2	R O M	
2 0 3	R A M	
2 0 4	大容量フラッシュメモリ	
2 0 5	音源 L S I	
2 0 6	波形メモリ	
2 0 7	キー・スキャナ	
2 0 8	A / D コンバータ	
2 0 9	L C D コントローラ	
2 1 0	D / A コンバータ	
2 1 1	アンプ	30
2 1 3	M I D I I / F	
2 1 4	システムバス	

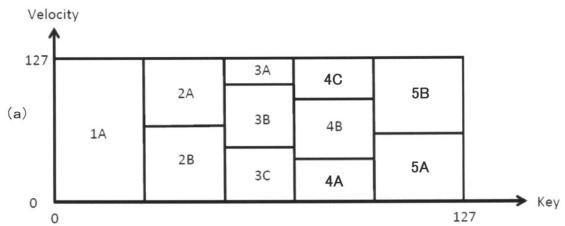
【図1】



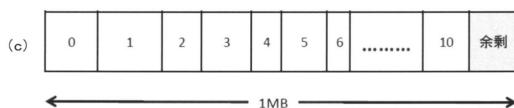
【図2】



【図3】



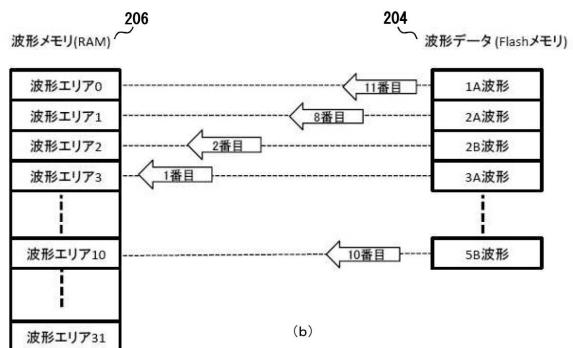
波形番号	最低ペロシティ	最大ペロシティ	最低キー番号	最高キー番号	音色先頭からのアドレス	波形サイズ
0 (1A)	0	127	0	30	00000H	0A00H
1 (2A)	61	127	30	50	00400H	1100H
:	:					
10 (5B)	80	127	100	127	F2000H	0800H
:	:					
31 (不使用)	0	0	0	0	FA000H	0



【図4】

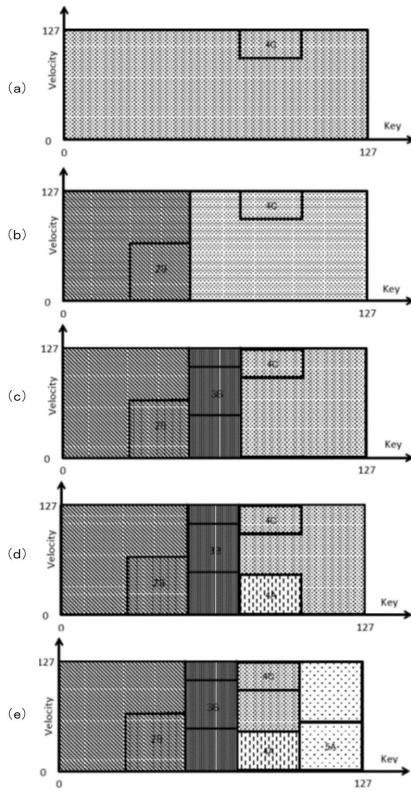
音色番号	音色名	波形アドレス	波形サイズ
0	Piano	0000000H	0FA00H
1	E.Piano	000FA000H	0E000H
2	Organ	001DA000H	0D800H
3	Guitar	002B2000H	05F00H
4	Saxophone	00311000H	0CC00H
5	Strings	003DD000H	09200H
6	Synth 1	0046F000H	0E000H
7	Drums 1	0054F000H	08700H
8	Clavinet	005D6000H	08900H
:	:		
15	Drums 2	00B47000H	0FC00H

(a)



(b)

【図5】



【図6】

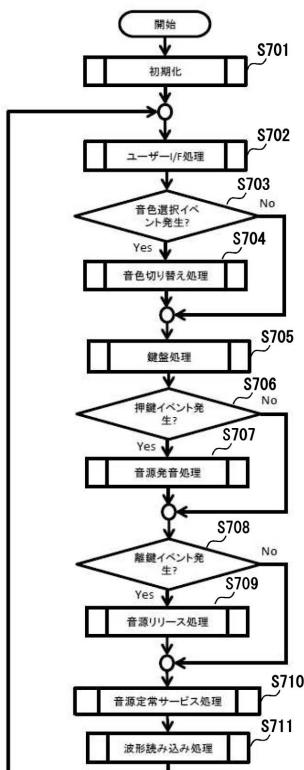
上位構造体 変数 [配列数]	変数 [配列数]	サイズ	域 (****は16進数)	初期 値	説明
-	C_TONE	8bit	0-15	0	現在の音色番号
-	FLASH_OFFSET	32bit	0-FFFFFFFFH	0	読み込み音色の波形のフラッシュメモリ上の 格納先頭アドレス
-	READING_WAVE	8bit	0-31, -1	-1	現在読み込み中の波形の番号 (-1の時は読み込み中の波形無し)
WM_ST[32] RAM波形メモ リスデータス (32波形部分)	WAVE	1bit	0-1	0	波形メモリ上の波形の有無 0:無し, 1:有り(読み込み済み)
	PRI	8bit	0-31, -1	-1	波形読み込み優先順位
	ORG_VEL_LO	8bit	0-127	0	本来、該当波形が発音する範囲の最小ペロシティ
	ORG_VEL_HI	8bit	0-127	127	本来、該当波形が発音する範囲の最大ペロシティ
	ORG_KEY_LO	8bit	0-127	0	本来、該当波形が発音する範囲の最低キーナンバ
	ORG_KEY_HI	8bit	0-127	127	本来、該当波形が発音する範囲の最高キーナンバ
	VEL_LO	8bit	0-127	0	現在、該当波形が発音する範囲の最小ペロシティ
	VEL_HI	8bit	0-127	127	現在、該当波形が発音する範囲の最大ペロシティ
	KEY_LO	8bit	0-127	0	現在、該当波形が発音する範囲の最低キーナンバ
	KEY_HI	8bit	0-127	127	現在、該当波形が発音する範囲の最高キーナンバ
OFFSET	OFFSET	32bit	0-100000H	0	音色の持つ波形データの先頭からの アドレスオフセット
	WAVE_SIZE	32bit	0-100000H	0	該当する波形データのサイズ(バイト) 0の時は波形なし
	READ_SIZE	32bit	0-100000H	0	前回の波形読み込みまでの 転送済みの波形データのサイズ(バイト)

(a)

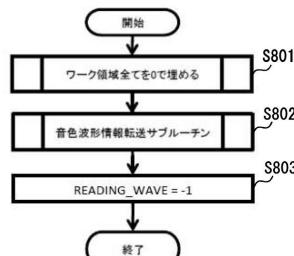
上位構造体 変数 [配列数]	変数 [配列数]	サイズ	域 (****は16進数)	説明
ROM_WAVE_OFFSET[16] ROM_WAVE_OFFSET[32] RAM波形メモ リスデータス (16音色 × 32波形部分)	ROM_WAVE_OFFSET[16]	32bit	0-FFFFFFFFFFH	音色波形のフラッシュメモリ上の先頭アドレス
	VEL_LO	8bit	0-127	該当波形が発音する範囲の本来の最小ペロシティ
	VEL_HI	8bit	0-127	該当波形が発音する範囲の本来の最大ペロシティ
	KEY_LO	8bit	0-127	該当波形が発音する範囲の本来の 最低キーナンバ
	KEY_HI	8bit	0-127	該当波形が発音する範囲の本来の 最高キーナンバ
	OFFSET	32bit	0-100000H	音色の持つ波形データの先頭からの アドレスオフセット
	WAVE_SIZE	32bit	0-100000H	該当する波形データのサイズ(バイト) 0の時は波形なし
	PRI	8bit	0-31, -1	波形読み込み順位 読み込み不要ならば-1

(b)

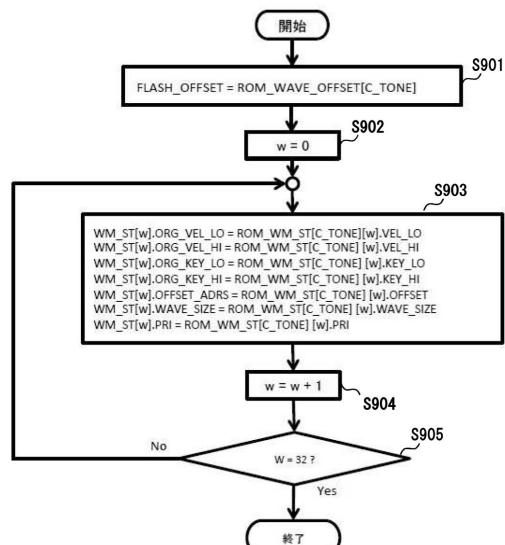
【図7】



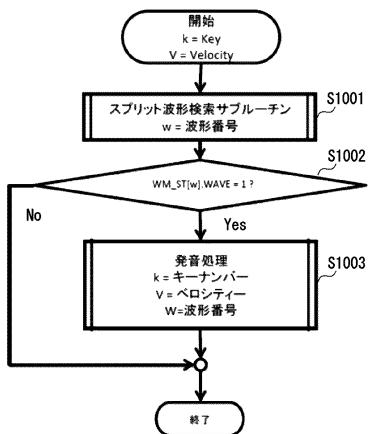
【図8】



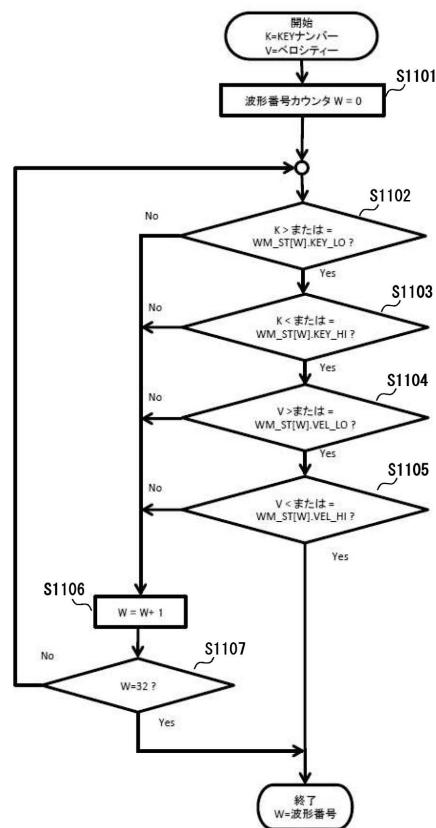
【図9】



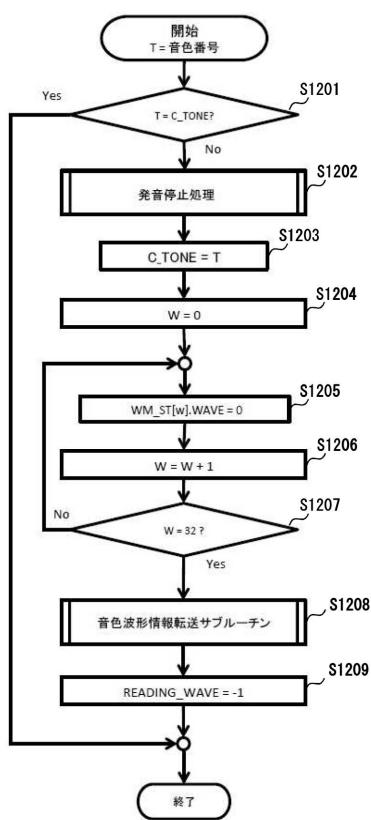
【図10】



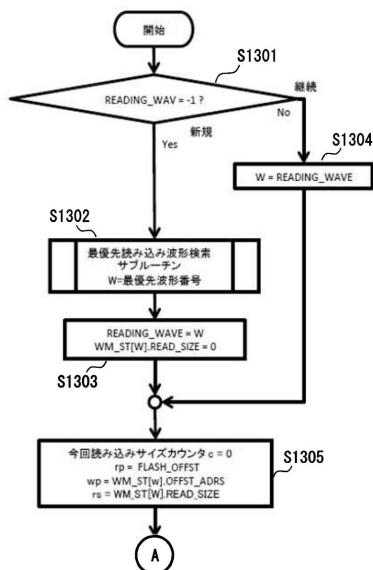
【図11】



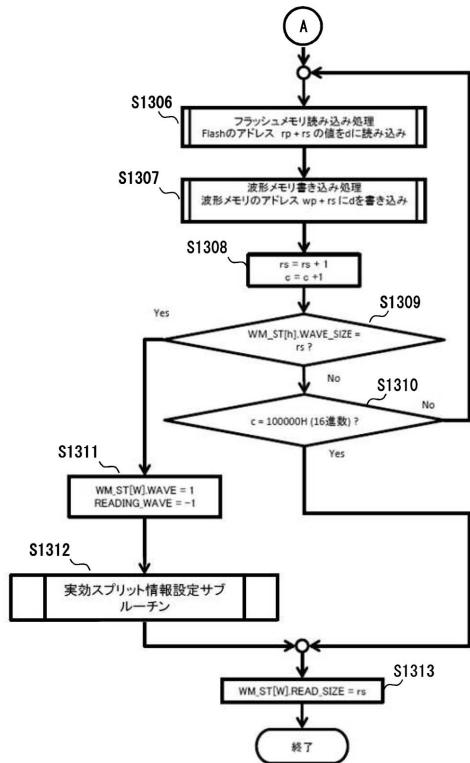
【図12】



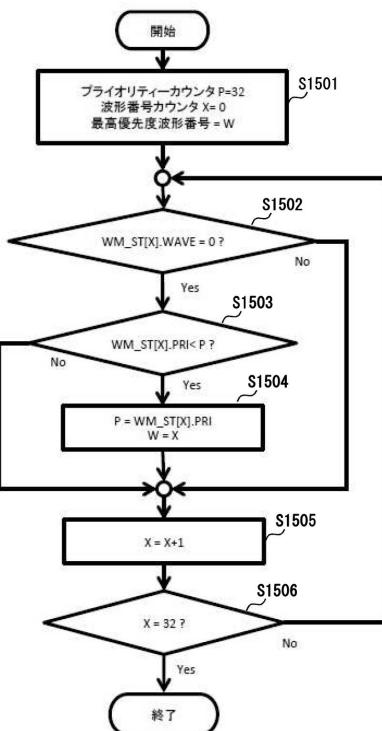
【図13】



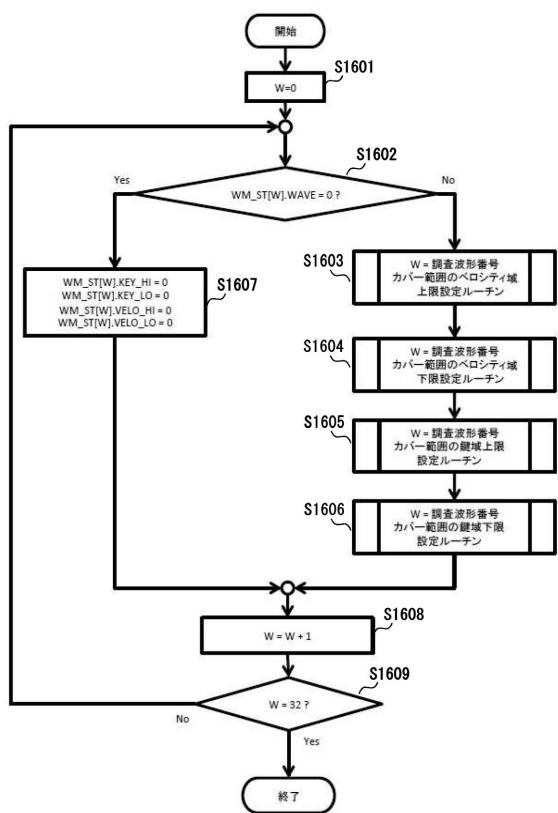
【図14】



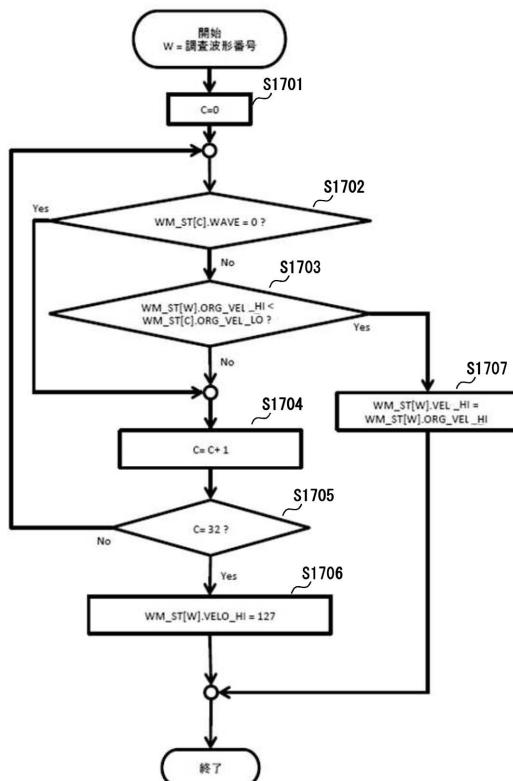
【図15】



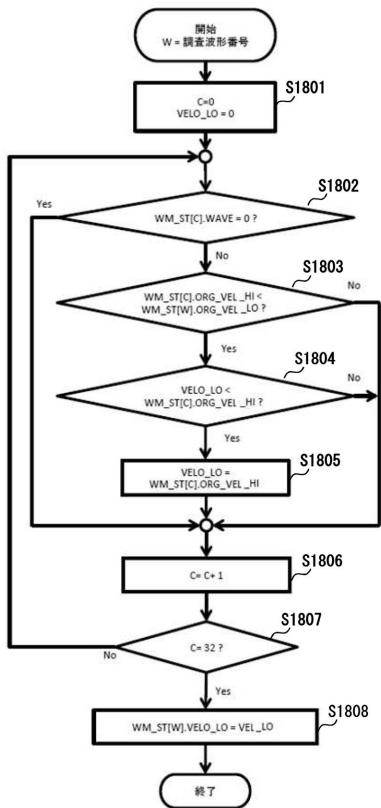
【図16】



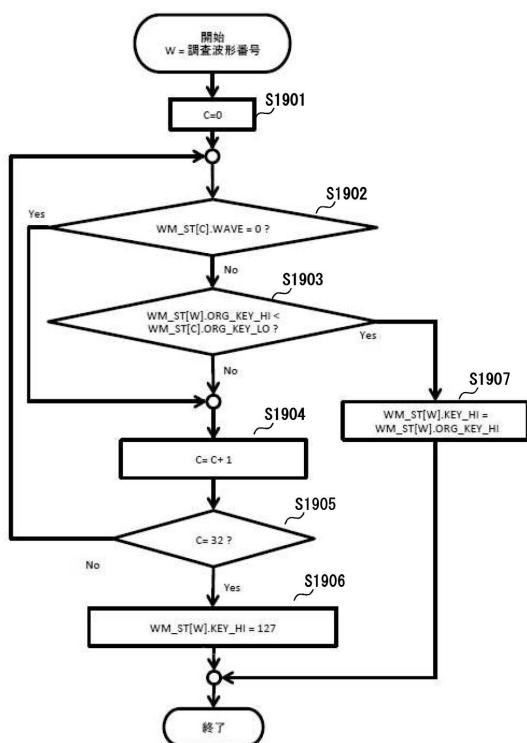
【図17】



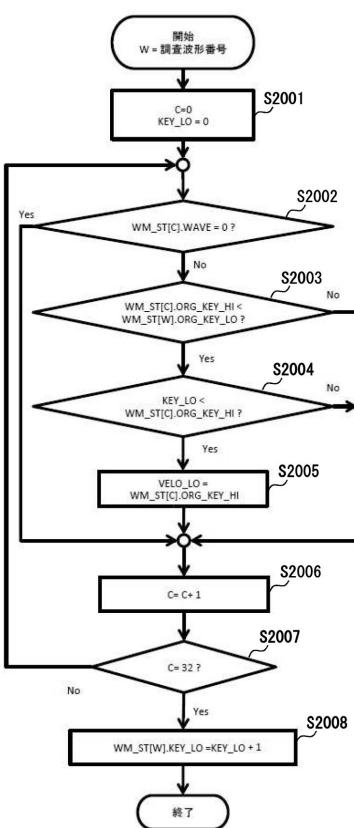
【図18】



【図19】



【図20】



---

フロントページの続き

(56)参考文献 特開2003-208181(JP,A)  
特開2000-276172(JP,A)

(58)調査した分野(Int.Cl., DB名)

G10H 1/00 - 7/12