

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第4829968号
(P4829968)

(45) 発行日 平成23年12月7日(2011.12.7)

(24) 登録日 平成23年9月22日(2011.9.22)

(51) Int.Cl.

F I

H04B 1/709 (2011.01)

H04J 13/00 420

請求項の数 3 (全 14 頁)

(21) 出願番号	特願2008-523848 (P2008-523848)	(73) 特許権者	501263810
(86) (22) 出願日	平成17年7月26日(2005.7.26)		トムソン ライセンシング
(65) 公表番号	特表2009-503993 (P2009-503993A)		Thomson Licensing
(43) 公表日	平成21年1月29日(2009.1.29)		フランス国, 92130 イッシー レ
(86) 国際出願番号	PCT/US2005/026453		ムーリノー, ル ジャンヌ ダルク,
(87) 国際公開番号	W02007/018497		1-5
(87) 国際公開日	平成19年2月15日(2007.2.15)		1-5, rue Jeanne d' A
審査請求日	平成20年7月17日(2008.7.17)		rc, 92130 ISSY LES
			MOULINEAUX, France
		(74) 代理人	110001243
			特許業務法人 谷・阿部特許事務所
		(72) 発明者	リトウィン ルイス ロバート
			アメリカ合衆国, ニュージャージー O
			8820, エディソン, ギスボロー
			ウェイ 15番地
			最終頁に続く

(54) 【発明の名称】 メモリ・アーキテクチャを用いたプライマリ・セルサーチのための相関器

(57) 【特許請求の範囲】

【請求項 1】

第1段階相関器から入力データを受信する第2段階相関器を備えている装置であって、
 前記第2段階相関器は、メモリアーキテクチャを備え、
 前記第2段階相関器は、1組の第2段階相関器を備え、
 前記1組の第2段階相関器のうちの第1のものは、入力データの実部の値を受信して処理し、前記1組の第2段階相関器のうちの第2のものは、入力データの虚部の値を受信して処理し、

前記1組の第2段階相関器のそれぞれはさらに、
 入力データを受信する第1のマルチプレクサと、
 前記入力データの第1の部分を受信して保存する第1の保存レジスタと、
 前記入力データの第2の部分を受信して保存する第2の保存レジスタと、
 入力データの前記第1の部分と、入力データの前記第2の部分を連結するための連結器と、

書き込みポイントにより書き込まれ、読み出しポイントにより読み出されるデュアルポートメモリであって、前記連結された入力データを受信して保存するメモリと、
 前記メモリの読み出し/書き込みポイント値を生成する、読み出し/書き込みアドレス生成部と、

前記連結された入力データを読み出して保存する読み出しレジスタと、
 前記連結された入力データの、2つの異なる部分を解析する手段と、

10

20

複数の相関レジスタと、
前記複数の相関レジスタのうちの１つのデータに前記解析されたデータを加算すること、及び前記複数の相関レジスタうちの１つのデータから、前記解析されたデータを減算することにより、相関を実行する加算及び符号変換器と、
前記加算及び符号変換器から、同時に相関を処理する前記複数の相関レジスタへの出力を多重する第２のマルチプレクサと、
前記加算及び符号変換器から前記複数の相関レジスタへの出力を多重する第３のマルチプレクサと、
前記複数の相関レジスタに保存された相関値を積算して、積算された相関値を生成する前記加算器と、
複数の相関出力レジスタと、
前記積算された相関値を、前記複数の相関出力レジスタの１つに多重する第４のマルチプレクサと、
前記相関出力レジスタから、前記積算された相関値を出力する第５のマルチプレクサと、
プライマリ同期符号のインデックスを生成するインデックス生成器と、
前記複数の相関レジスタのうちの１つのデータに前記解析されたデータを加算するか、又は前記複数の相関レジスタのうちの１つのデータから前記解析されたデータを減算するかを決定するために、前記加算及び符号変換器に利用可能なプライマリ同期シーケンスを作成するプライマリ同期保存部であって、前記プライマリ同期シーケンスを保存するプライマリ同期保存部と、
相関処理を制御する制御部と、
前記積算された相関値の絶対値計算を実行する手段と
 を備える
 ことを特徴とする装置。

10

20

【請求項２】

前記１組の第２段階相関器のそれぞれは、同じ機能である、請求項１に記載の装置。

【請求項３】

前記装置は、移動装置である請求項１に記載の装置。

30

【発明の詳細な説明】

【技術分野】

【０００１】

本発明は、移動端末に関し、特に、プライマリ・セルサーチに用いる相関器に関する。

【背景技術】

【０００２】

UMTS無線信号の時間の基本単位は、１０ミリ秒（ms）の無線フレームであり、無線フレームは、それぞれが２５６０チップの１５スロットに分割される。セル（又は基地局）からUMTS受信機へのUMTS無線信号は、“下りリンク信号”であり、反対方向の無線信号は、“上りリンク信号”と呼ばれる。

40

【０００３】

UMTS（Universal mobile Telecommunication System）WCDMA（Wideband Code-Division Multiple Access）標準の物理層は、３．８４Mcpsのチップ・レートでの直接シーケンス・スペクトラム拡散（DSSS）変調を使用する。周波数分割複信（FDD）モードは、上りリンクチャネル及び下りリンクチャネルを、それぞれが５MHzである異なる周波数帯域で伝送する。このモードは、時分割複信（TDD）モードより多数のユーザを収容できるため、典型的には、広い屋外セルに使用される。TDDモードにおいては、同じ上りリンクチャネルと下りリンクチャネルが、異なるタイムスロットで伝送される。TDDモードは、FDDモード程、多くのユーザを収容しないので、TDDモードは

50

、より小さなセルに適している。TDDモードは、FDDモードと比較し、非対称トラフィックの伝送により適している。

【0004】

例えば、CDMA移動受信機といった、UMTSネットワーク内の受信機により実行される重要な手順は、セルサーチ動作である。セルサーチは、典型的には、受信機の一部として組み込まれているセルサーチ・システムにより実行される。セルサーチ・システムは、受信機が位置しているセルに関する同期情報を判定するため、受信機の電源投入後に活性化される。セルサーチ動作は、3段階の処理である。つまり、セルサーチ・システムは、スロット同期（第1の同期）と、フレーム同期及びスクランブル符号グループ判定（第2の同期）と、スクランブル符号判定とを実行する。

10

【0005】

電源投入後、移動端末（MT）は、音声／データ通信の開始が可能になる前に、幾つかの動作を行わなければならない。第1に、受信機は、受信信号電力を調整し、アナログデジタル変換器でのクリップを防ぐために、自動利得制御（AGC）を実行する必要がある。この処理は、まず、同期チャンネル（SCH）上で実行され、セルのスクランブル符号を獲得すると、ディスクリンブルされた共通パイロットチャンネル（CPICH）が使用される。

【0006】

次に、受信機は、タイミング同期を獲得する必要がある。タイミング同期は、SCHチャンネルにより達成できる。MTは、見つけることができる最も強いSCH信号を探索し、その信号で、MTは通信を開始するセルを決定する。SCHチャンネルは周期的なので、受信機は、タイミング誤差を導出するために、プライマリSCHに対して相関を取ることができる。このチャンネルに基づき、受信機は、チップ、シンボル及びスロット同期を達成することができる。

20

【0007】

プライマリSCHは、システムの総てのセルで同じ信号を伝送する。セカンダリSCHは、セルごとに異なり、フレーム毎に繰り返すセカンダリ同期符号（SSC）のパターンを伝送する。MTがこのシーケンスを受信すると、フレーム同期が確立される。

【0008】

セルサーチの実行において、セルサーチ・システムは、受信無線信号の同期チャンネル（SCH）と共通パイロットチャンネル（CPICH）にアクセスする。SCHは、プライマリSCHとセカンダリSCHから形成される複合チャンネルである。各スロット内で、プライマリSCHは、プライマリ同期符号（PSC）を指定する。プライマリSCHは、しかしながら、各2560チップ・スロットの最初の256チップの間のみデータを含んでいる。公知の様に、“チップ”又は“チップ・レート”は、CDMA通信システム内の拡散符号のレートを参照している。

30

【0009】

更に、パターンは、現在のセルのスクランブル符号が属するスクランブル符号グループを特定する。64個のスクランブル符号グループがあり、各グループは、8個のスクランブル符号を含んでいる。MTが現在のセルのスクランブル符号グループを判定すると、現在のセルのスクランブル符号は、このグループの8つの符号に絞り込まれる。

40

【0010】

典型的な、キャリアベース受信機の獲得処理は以下の通りである。

プライマリ・セルサーチ

セカンダリ・セルサーチ

スクランブル符号決定

マルチパス探索

フィンガ割当て

符号トラック及び自動周波数制御（AFC）ループのロック

フィンガ出力の最大比合成（MRC）

50

受信機ロックが獲得され、データが上位レイヤに送信可能となる。
この獲得処理は長く、完了するまで数秒のオーダがかかり得る。

【 0 0 1 1 】

取り組む問題は、3 G W C D M A 受信機でのプライマリ・セルサーチ処理の第 2 段階のための面積効率の良い相関ブロックをどの様に実装するかである。プライマリ・セルサーチ処理の第 1 段階は、1 6 連続サンプルの相関をとることを含み、1 6 チップ毎に相関出力を生成する。この様に、第 1 段階の相関器は、ある相関のために、同時に 1 6 チップを保存することのみ必要であり、これは実行することが比較的簡単である。1 チップ当たり 4 サンプルを使用する受信機でさえ、連続する 2 5 6 サンプルのみを保存すれば良い。これは、第 1 段階の相関器は、サンプルの連続するグループを、到着した時に処理することを意味する。

10

【 0 0 1 2 】

第 2 段階の処理における各相関も、1 6 チップを必要とする。しかしながら、3 G W C D M A 標準で使用されている階層的ゴレー符号の性質により、これら 1 6 チップのそれぞれは、1 6 チップだけ離れて位置している。この様に、1 チップ当たり 4 サンプルを使用する受信機は、連続していない 2 5 6 チップを処理する必要がある。それどころか、ある相関のために、 $16 \times 4 = 64$ サンプルだけ離れた 2 5 6 チップを必要とする。ある第 2 段階の相関に必要な総てのサンプルを保存するために、受信機は、1 0 2 4 位置のタップ遅延線を必要とする (1 6 チップだけ離れた 1 6 チップで 2 5 6 チップ、そして、1 チップ当たり 4 サンプルで、1 0 2 4 サンプル)。従来技術は、第 2 段階の相関を、レジスタに基づく設計により実装している。この数のレジスタ (例えば、1 0 2 4) は、A S I C 上での広いダイ・スペースを消費するので、A S I C デザインに実用的ではない。この様に、より面積効率の良い手法が有利である。

20

【 発明の開示 】

【 発明が解決しようとする課題 】

【 0 0 1 3 】

本発明は、3 G W C D M A 受信機のプライマリ・セルサーチ処理で使用する相関器の第 2 の階層的段階のアーキテクチャである。使用されるアーキテクチャは、メモリに基づくものであり、A S I C 上で利用可能なダイ・スペースの面積効率の良い設計を可能にする。

30

【 課題を解決するための手段 】

【 0 0 1 4 】

ある位置の数に対して、メモリはレジスタより効率的であるため、本発明は、メモリに基づく手法を使用する。しかしながら、デュアル・ポート R A M メモリブロックの特性は、あるクロック・サイクルで実行可能なメモリ読み出し / 書き込みが、1 サイクル当たり 1 回の読み出しと 1 回の書き込みに制限されることを意味している。これは、受信機の、1 チップ当たり 3 2 クロック・サイクルという制限内で、総ての処理を可能にするために十分な読み出し及び書き込みを可能としないので、メモリブロック設計の課題を示している。1 チップ当たり 3 2 クロック・サイクル内で所望の処理を達成するために、1 クロック・サイクル当たり単一の読み出し及び書き込みを使用するための幾つの特徴が、アーキテクチャに加えられる。

40

【 0 0 1 5 】

第 1 段階相関器から入力データを受信する第 2 段階相関器を備えており、第 2 段階相関器は、メモリ・アーキテクチャを利用する装置が記載される。読み出しポインタ及び書き込みポインタをリセットし、1 組の保存レジスタの 1 つに入力データを交互に多重し、1 組の保存レジスタの内容を連結し、連結された内容を、書き込みポインタに従いメモリに書き込み、連結された内容を、読み出しポインタに従い、メモリから読み出しレジスタに出力し、読み出しアドレス・ポインタを更新し、書き込みアドレス・ポインタを更新する、データの第 2 段階相関を行う方法も記載される。

【 発明を実施するための最良の形態 】

50

【 0 0 1 6 】

本発明は、以下の図面を用いた詳細な説明から良く理解できる。図において、同様の要素には、同様の参照番号を表す。

【 0 0 1 7 】

セルサーチは、移動端末で実行される。セルサーチ処理の最上位レベルのブロック図である図 1 を参照すると、本発明は、プライマリ・セルサーチの第 2 段階で使用する相関器 1 2 5、1 3 0 を含み、これらは、プライマリ・セルサーチの第 1 段階相関器 1 0 5、1 1 0 から、実部入力 1 1 5 と虚部入力 1 2 0 を受信する。第 1 段階プライマリ・セルサーチ相関器 1 0 5、1 1 0 の出力は、第 2 段階相関器 1 2 5、1 3 0 に入力される。第 2 段階相関器 1 2 5、1 3 0 の出力は、非コヒーレント結合器 1 3 5 に出 force され、非コヒーレント結合器 1 3 5 は、フレーム・バッファ 1 4 0 に入力を与える。フレーム・バッファ 1 4 0 は、セルサーチ結果を与える。

10

【 0 0 1 8 】

図 2 は、本発明のアーキテクチャのブロック図である。特に、図 2 は、プライマリ・セルサーチの第 2 段階で使用される、本発明の相関器のアーキテクチャである。本発明の相関器は、A S I C のダイ・スペースの点で、面積効率が良いという利点を持つ、メモリ・アーキテクチャを使用する。図 2 のメモリ読み出し / 書き込みアドレス生成ブロック 2 3 5 は、読み出し / 書き込みポインタ値を生成する (図 3 にも示されている)。本発明の第 2 段階相関器 1 2 3 は、実際には、機能的には同一 / 等価である 1 組の第 2 段階相関器 1 2 5、1 3 0 である。第 2 段階相関器 1 2 5 と 1 3 0 の違いは、各第 2 段階相関器が受信する入力データ (実部と虚部) である。

20

【 0 0 1 9 】

第 1 段階相関器 (図 1 に表示) の相関出力 (実部と虚部) は、図 2 のマルチプレクサ 2 0 5 に到達する。これらサンプルは、1 チップ当たり 4 回到達し、それらは、交互に多重されて、最初のものは、storage_low_reg (ロー保存レジスタ) 2 1 0 に、次のものは、storage_high_reg (ハイ保存レジスタ) 2 1 5 に、以下、交互に、保存レジスタに出力される。以下にて詳細に記述するロジックに基づき、ロー及びハイ保存レジスタ値 (それぞれ 1 6 ビット幅) は、単一の 3 2 ビット値を形成するためにブロック 2 2 5 で連結され、所定のクロック・サイクルでメモリ 2 3 0 に書き込まれる。この手法は、クロック・サイクル当たり 1 回のメモリ書き込みのみという制限のため使用し、2 つのサンプルを 1 つの値として保存することで、この設計は、各クロック・サイクルにおいてメモリ 2 3 0 に 2 つのサンプルを保存することを可能にする。この点でのメモリ 2 3 0 の使用は、チップのダイ・スペースを節約する。従来技術での実装は、メモリの代わりに、レジスタバンクを使用している。

30

【 0 0 2 0 】

値が、メモリの所定位置から読み出され、read_reg (リード・レジスタ) 2 4 0 に保存される。この点において、ビットは、上位及び下位の値に対応して解析され、2 つの異なるサンプルとして処理される。インデックス生成器 2 4 5 は、P S C インデックス / シーケンスを生成する。read_reg 2 4 0 からサンプルを取得し、取得したサンプルを、保存された P S C シーケンス・ブロック 2 5 0 の符号に基づき、corr_reg (相関レジスタ) のサンプルに加算、あるいは、corr_reg のサンプルから減算することで、面積を必要とする乗算器を用いることなく、ブロック 2 5 5 において相関が実行される (P S C シーケンスが + 1 であれば、値は加算され、P S C シーケンスが - 1 であれば、値は減算される。)。corr_reg は corr_reg 0 [0] ~ corr_reg 0 [3] 2 7 0 a、corr_reg 1 [0] ~ corr_reg 1 [3] 2 7 0 b、corr_reg 2 [0] ~ corr_reg 2 [3] 2 7 0 c、corr_reg 3 [0] ~ corr_reg 3 [3] 2 7 0 d の、1 6 個あることに留意すべきである。これは、4 つの並列ブロックそれぞれにおいて、4 つの相関を同時に保存して処理することを可能にするためである。各組のレジスタは、3 2 クロック・サイクルの 8 クロック・サイクルを使用し、1 組のレジスタのみが同時に使用される。ブロック 2 5 5 の出

40

50

力は、マルチプレクサ260、265で多重化され、関連レジスタ270a~270dに出力される。

【0021】

ある関連の16の値の総てが加算ブロック275で積算された後、corr_regに保存された値は、4つの対応するcorr_out(関連出力)レジスタ285a~285dの1つに、マルチプレクサ280経由で転送される。つまり、corr_out[0]=corr_reg0[0]+corr_reg1[0]+corr_reg2[0]+corr_reg3[0]285aである。

【0022】

corr_outレジスタの出力は、図1の非コヒーレント結合器135に向けて多重化される。corr_outレジスタの内容の絶対値(ABS)を取ることも必要である。このブロックは、図2には示していないが、その機能は、corr_outレジスタ又はマルチプレクサ290の下流に追加するブロックにより実行される。

10

【0023】

テーブル1に示す擬似コードは、どの様にアーキテクチャが動作するかについてのより詳細を与える。図2の制御ブロック220は、本発明の関連器の構成要素と機能を、調整し制御する。左の数字は、クロック・サイクルを示す。本発明のアーキテクチャは、1つのサンプル当たり32クロックのクロック・サイクル構造に基づいている。

【0024】

擬似コード

```
rp=3           // 読み出しポインタ - 9ビット数
wp=1           // 書き込みポインタ - 9ビット数
corr_reg[0..3]=0
corr_reg[0..3]=0
```

10

サンプル出力コード

メモリ入力／出力コード*

【 0 0 2 5 】

【表 2】

相関及びメモリアンタフェースコード

0	corr_reg3[0]及びcorr_reg3[1]をread_reg(上位及び下位)のサンプルで更新 2クロックサイクル先のため”読み出し”アドレスをメモリに書き込み - リードアドレスはrp	
1	corr_reg3[2]及びcorr_reg3[3]をread_reg(上位及び下位)のサンプルで更新 2クロックサイクル先のため”読み出し”アドレスをメモリに書き込み - リードアドレスはrp-1 rp=rp+32	
2, 4, 6, 8	corr_reg0[0]及びcorr_reg0[1]をread_reg(上位及び下位)のサンプルで更新 2クロックサイクル先のため”読み出し”アドレスをメモリに書き込み - リードアドレスはrp	10
3, 5, 7, 9	corr_reg0[2]及びcorr_reg0[3]をread_reg(上位及び下位)のサンプルで更新 2クロックサイクル先のため”読み出し”アドレスをメモリに書き込み - リードアドレスはrp-1 rp=rp+32	
10, 12, 14, 16	corr_reg1[0]及びcorr_reg1[1]をread_reg(上位及び下位)のサンプルで更新 2クロックサイクル先のため”読み出し”アドレスをメモリに書き込み - リードアドレスはrp	
11, 13, 15, 17	corr_reg1[2]及びcorr_reg1[3]をread_reg(上位及び下位)のサンプルで更新 2クロックサイクル先のため”読み出し”アドレスをメモリに書き込み - リードアドレスはrp-1 rp=rp+32	20
18, 20, 22, 24	corr_reg2[0]及びcorr_reg2[1]をread_reg(上位及び下位)のサンプルで更新 2クロックサイクル先のため”読み出し”アドレスをメモリに書き込み - リードアドレスはrp	
19, 21, 23, 25	corr_reg2[2]及びcorr_reg2[3]をread_reg(上位及び下位)のサンプルで更新 2クロックサイクル先のため”読み出し”アドレスをメモリに書き込み - リードアドレスはrp-1 rp=rp+32	
26, 28, 30	corr_reg3[0]及びcorr_reg3[1]をread_reg(上位及び下位)のサンプルで更新 2クロックサイクル先のため”読み出し”アドレスをメモリに書き込み - リードアドレスはrp	30
27, 29, 31	corr_reg3[2]及びcorr_reg3[3]をread_reg(上位及び下位)のサンプルで更新 2クロックサイクル先のため”読み出し”アドレスをメモリに書き込み - リードアドレスはrp-1 クロックサイクルが31でなければ、 rp=rp+32 クロックサイクルが31であれば、 rp=rp+482	

【0026】

擬似コードのリセットコードは、他の総ての処理の開始前に、共に9ビットの数である読み出しポインタ(rp)及び書き込みポインタ(wp)を初期化する。相関レジスタ(corr_reg)及び相関出力レジスタ(corr_out)もまた初期化される。 40

【0027】

擬似コードの相関出力コードは、クロック・サイクル0において、corr_out[0]及び[1]レジスタを、corr_regレジスタの内容に設定し、クロック・サイクル1において、corr_out[2]及び[3]レジスタを、corr_regレジスタの内容に設定する。

【0028】

擬似コードのサンプル出力コードは、クロック・サイクル7で、corr_out[0]レジスタの絶対値(abs)の出力サンプル(samp_out)を与える。擬似コー 50

ドのサンプル出力コードは、クロック・サイクル15で、`corr_out[1]`レジスタの絶対値(`abs`)の出力サンプル(`sample_out`)を与える。擬似コードのサンプル出力コードは、クロック・サイクル23で、`corr_out[2]`レジスタの絶対値(`abs`)の出力サンプル(`sample_out`)を与える。擬似コードのサンプル出力コードは、クロック・サイクル31で、`corr_out[3]`レジスタの絶対値(`abs`)の出力サンプル(`sample_out`)を与える。

【0029】

クロック・サイクル0及び16で、擬似コードのメモリ入力/出力コードは、`storage_low_reg`を、入力サンプル(`sample_in`)の値に設定する。クロック・サイクル8及び24で、擬似コードのメモリ入力/出力コードは、`storage_high_reg`を、入力サンプル(`sample_in`)の値に設定する。更に、クロック・サイクル8及び24で、メモリ書き込みアドレスに、書き込みポインタ(`wp`)の値が設定され、アドレス内のメモリデータは、`storage_high_reg`と`storage_low_reg`を連結した値に設定され、書き込みポインタ(`wp`)の値が減ぜられる。各クロック・サイクルにおいて、`read_reg`は、メモリ読み出し/書き込みアドレス生成ブロック235が生成する読み出しアドレスに従いメモリから読み出される`data_out`に設定される。

【0030】

擬似コードの相関及びメモリインタフェースコードは、以下の通り機能する。

【0031】

クロック・サイクル0で、`corr_reg3[0]`及び`corr_reg3[1]`は、`read_reg`の上位及び下位サンプルに更新される。2クロック・サイクル先のため“読み出し”アドレスはメモリに書き込まれ、“読み出し”アドレスは`rp`に等しくされる。

【0032】

クロック・サイクル1で、`corr_reg3[2]`及び`corr_reg3[3]`は、`read_reg`の上位及び下位サンプルに更新される。2クロック・サイクル先のため“読み出し”アドレスはメモリに書き込まれ、“読み出し”アドレスは`rp - 1`に等しくされる。リードポインタは、その後、32だけ増加される。

【0033】

クロック・サイクル2、4、6及び8で、`corr_reg0[0]`及び`corr_reg0[1]`は、`read_reg`の上位及び下位サンプルに更新される。2クロック・サイクル先のため“読み出し”アドレスはメモリに書き込まれ、“読み出し”アドレスは`rp`に等しくされる。

【0034】

クロック・サイクル3、5、7及び9で、`corr_reg0[2]`及び`corr_reg0[3]`は、`read_reg`の上位及び下位サンプルに更新される。2クロック・サイクル先のため“読み出し”アドレスはメモリに書き込まれ、“読み出し”アドレスは`rp - 1`に等しくされる。リードポインタは、その後、32だけ増加される。

【0035】

クロック・サイクル10、12、14及び16で、`corr_reg1[0]`及び`corr_reg1[1]`は、`read_reg`の上位及び下位サンプルに更新される。2クロック・サイクル先のため“読み出し”アドレスはメモリに書き込まれ、“読み出し”アドレスは`rp`に等しくされる。

【0036】

クロック・サイクル11、13、15及び17で、`corr_reg1[2]`及び`corr_reg1[3]`は、`read_reg`の上位及び下位サンプルに更新される。2クロック・サイクル先のため“読み出し”アドレスはメモリに書き込まれ、“読み出し”アドレスは`rp - 1`に等しくされる。リードポインタは、その後、32だけ増加される。

【0037】

クロック・サイクル18、20、22及び24で、`corr_reg2[0]`及び`corr_reg2[1]`は、`read_reg`の上位及び下位サンプルに更新される。2クロック・サイクル先のため“読み出し”アドレスはメモリに書き込まれ、“読み出し”アドレスは`rp`に等しくされる。

【0038】

クロック・サイクル19、21、23及び25で、`corr_reg2[2]`及び`corr_reg2[3]`は、`read_reg`の上位及び下位サンプルに更新される。2クロック・サイクル先のため“読み出し”アドレスはメモリに書き込まれ、“読み出し”アドレスは`rp-1`に等しくされる。リードポインタは、その後、32だけ増加される。

【0039】

クロック・サイクル26、28及び30で、`corr_reg3[0]`及び`corr_reg3[1]`は、`read_reg`の上位及び下位サンプルに更新される。2クロック・サイクル先のため“読み出し”アドレスはメモリに書き込まれ、“読み出し”アドレスは`rp`に等しくされる。

【0040】

クロック・サイクル27、29及び31で、`corr_reg3[2]`及び`corr_reg3[3]`は、`read_reg`の上位及び下位サンプルに更新される。2クロック・サイクル先のため“読み出し”アドレスはメモリに書き込まれ、“読み出し”アドレスは`rp-1`に等しくされる。クロック・サイクル31でなければ、リードポインタは、その後、32だけ増加される。クロック・サイクル31であれば、リードポインタは、482だけ減ぜられる。

【0041】

図3に関し、書き込みポインタ(`wp`)は値1に初期化され、各32クロック・サイクル周期内で2回減ぜられる(モジュロ512)。読み出しポインタ(`rp`)は値3に初期化され、各32クロック・サイクル周期内で15回、32だけ増加され、32クロック・サイクル周期ごとに1回、482(512-30)だけ減ぜられる。本発明のデュアル・ポート・メモリ及びその利用は、読み出し及び書き込みポインタが同じメモリを異なるときにアドレス指定するパッファ又はスライディングウインドウの様なものである。つまり、読み出されるメモリ位置と、書き込まれるメモリ位置のオーバーラップは無い。これは、クロック・サイクル当たり、1回の読み出し及び1回の書き込みのみであるからである。読み出し及び書き込みポインタのインデックスと、増加及び減少の値は、サンプル/チップの数が増加又は減少すると、変化する。具体的には、本例の512の位置を有するデュアル・ポート・メモリを示す図3を参照すると、各位置は32ビットであり、リセット時、書き込みポインタ(`wp`)は1に初期化され、読み出しポインタ(`rp`)は3に初期化される。最初の32クロック・サイクルの後、書き込みポインタ(`wp`)は511であり、読み出しポインタ(`rp`)は1である。

【0042】

本発明の第2段階相関器の動作のフローチャートである図4を参照する。ステップ405で、サンプルは、交互に、`storage_reg_low`及び`storage_reg_high`に多重化される。ステップ410で、`storage_reg_low`及び`storage_reg_high`の内容が連結され、メモリ読み出し/書き込みアドレス生成ブロック235によって指定される書き込みポインタ(`wp`)に従い、メモリに単一の値として書き込まれる。ステップ415で、クロック・サイクルごとに、メモリ230からのサンプルは、メモリ読み出し/書き込みアドレス生成ブロック235によって指定される読み出しポインタ(`rp`)に従い、`read_reg240`に出力される。ステップ420で、ブロック245により生成され、ブロック250に保存されたPSCインデックス/シーケンスの符号に基づき、(+/-)`read_reg240`の値を、対応する`corr_reg270a~270d`の値に加算することにより相関処理が実行される。ステップ425で、16個の積算処理の後、`corr_reg`の値は、加算器275及びマルチプレクサ280経由で、対応する`corr_out285a~285d`に保存

10

20

30

40

50

され、4つの相関の並行処理が効果的に完了する。corr_out285a~285dレジスタの値の絶対値(abs)が、corr_out285a~285dレジスタで、あるいは、ステップ430での相関値の出力前に、corr_out285a~285dレジスタの出力が(図示しない)絶対値ブロックに多重化されることで、求められる。

【0043】

本発明は、例えば、移動端末、アクセスポイント又は移動体通信網内において、種々のタイプのハードウェア、ソフトウェア、ファームウェア、特別な目的のプロセッサ、又は、それらの組合せで実現可能であることが理解される。好ましくは、本発明は、ハードウェアとソフトウェアの組合せとして実装される。更に、ソフトウェアは、好ましくは、プログラム保存装置上で明白に具現化されたアプリケーションプログラムとして実装される。アプリケーションプログラムは、任意の好ましいアーキテクチャの装置にアップロードされて、その装置により実行される。好ましくは、装置は、1つ以上の中央処理部(CPU)、ランダムアクセスメモリ(RAM)及び入出力(I/O)インタフェースの様なハードウェアを有するコンピュータプラットフォーム上で実装される。コンピュータプラットフォームは、オペレーティングシステム及びマイクロ命令コードも含んでいる。上述した種々の処理及び機能は、オペレーティングシステム経由で実行されるマイクロ命令コードの部分又はアプリケーションプログラムの部分(又は、それらの組合せ)であるかもしれない。更に、追加のデータ保存装置やプリント装置の様な、種々の他の周辺機器がコンピュータプラットフォームに接続可能である。

【0044】

図を用いて説明した、システムを構成する要素と、方法ステップの幾つかは、好ましくは、ソフトウェアで実装され、システム要素(又は、方法ステップ)は、本発明のプログラムの方法により異なるかもしれない。この教示により、当業者は、本発明の同様の実装や構成を考えることができる。

【図面の簡単な説明】

【0045】

【図1】セルサーチ処理の最上位レベルのブロック図である。

【図2】本発明のアーキテクチャのブロック図である。

【図3】本発明の原理による、メモリの読み出し/書き込みポインタの利用の実施形態である。

【図4】本発明の原理によるフローチャートである。

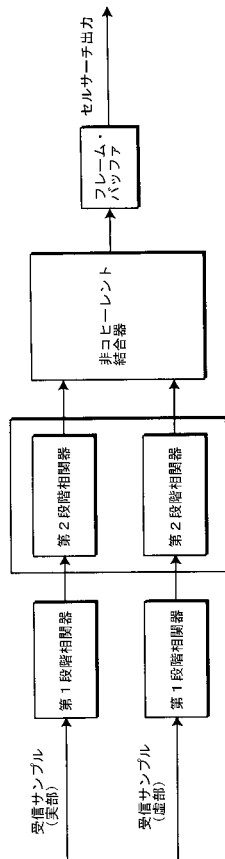
【符号の説明】

【0046】

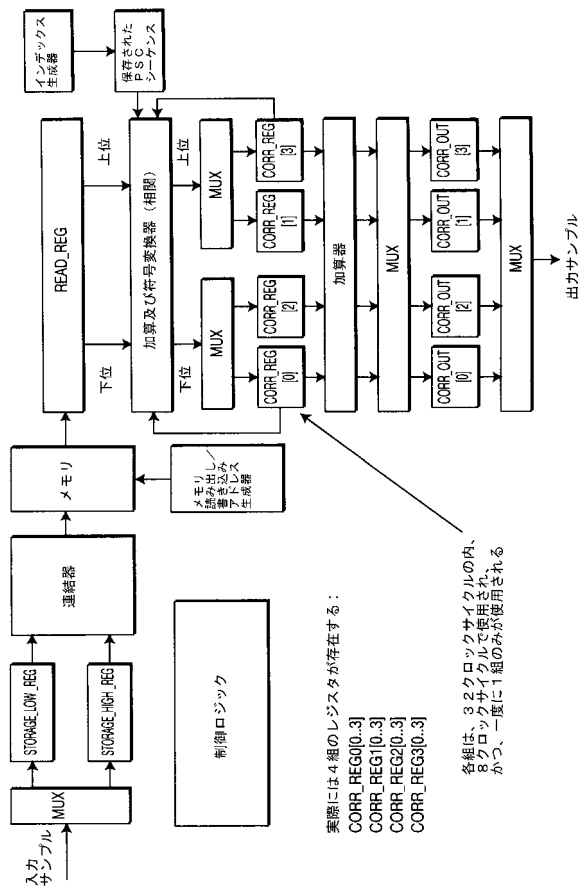
- 105、110 第1段階相関器
- 123、125、130 第2段階相関器
- 135 非コヒーレント結合器
- 140 フレーム・バッファ
- 210 ロー保存レジスタ
- 215 ハイ保存レジスタ
- 220 制御ブロック
- 225 連結ブロック
- 230 メモリ
- 235 メモリ読み出し/書き込みアドレス生成ブロック
- 240 リード・レジスタ
- 245 インデックス生成器
- 250 PSCシーケンス・ブロック
- 255 相関ブロック
- 260、265、280、290 マルチプレクサ
- 270a~270d 相関レジスタ
- 275 加算ブロック

285a ~ 285d 関連出力レジスタ

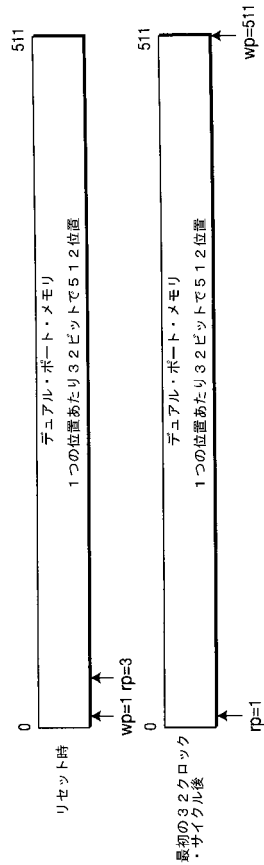
【図1】



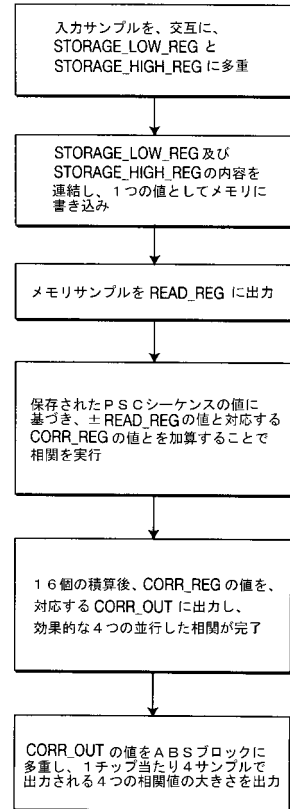
【図2】



【図 3】



【図 4】



フロントページの続き

審査官 橘 均憲

(56)参考文献 特開 2 0 0 3 - 1 5 2 6 0 0 (J P , A)
特開 2 0 0 1 - 1 3 6 1 0 3 (J P , A)
特開 2 0 0 1 - 0 9 4 4 6 8 (J P , A)
特開 2 0 0 5 - 1 1 7 2 6 5 (J P , A)
特開 2 0 0 4 - 2 2 2 1 5 2 (J P , A)
欧州特許出願公開第 0 1 3 9 1 9 9 9 (E P , A 1)
特開 2 0 0 3 - 2 1 8 8 3 5 (J P , A)

(58)調査した分野(Int.Cl. , D B 名)

H04B 1/69- 1/719

H04J 13/00-13/22