

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第4167732号
(P4167732)

(45) 発行日 平成20年10月22日(2008.10.22)

(24) 登録日 平成20年8月8日(2008.8.8)

(51) Int.Cl. F I
HO3M 5/12 (2006.01) HO3M 5/12
HO4L 25/49 (2006.01) HO4L 25/49 F

請求項の数 13 (全 21 頁)

(21) 出願番号	特願平8-163464	(73) 特許権者	591034154 フランス テレコム
(22) 出願日	平成8年6月24日(1996.6.24)		フランス国 パリ 75015 プラス ダルレ 6
(65) 公開番号	特開平9-18344	(73) 特許権者	594024774 ラ・ポスト
(43) 公開日	平成9年1月17日(1997.1.17)		フランス・92777・ブローニュ・ピラ ンクール・セデックス・クアイ・ドゥ・ブ ワン・ドゥ・ジュール・4
審査請求日	平成15年6月3日(2003.6.3)	(74) 代理人	100064908 弁理士 志賀 正武
審判番号	不服2006-4107(P2006-4107/J1)	(74) 代理人	100089037 弁理士 渡邊 隆
審判請求日	平成18年3月6日(2006.3.6)		
(31) 優先権主張番号	95 07491		
(32) 優先日	平成7年6月22日(1995.6.22)		
(33) 優先権主張国	フランス (FR)		

最終頁に続く

(54) 【発明の名称】 マンチェスター符号化復号化装置

(57) 【特許請求の範囲】

【請求項1】

受信されたマンチェスター符号化入力信号のタイミングを変更することなく動作するマンチェスター符号化復号化装置において、

前記符号化復号化装置は、

前記符号化入力信号によって動作し、かつ、符号化復号化装置のクロックに対する開始信号を提供する同期信号生成モジュールと、

伝送 / 符号化クロックと復号化クロックと低周波クロックとを生成するクロックの同期化および生成のためのモジュールと

を含み、

前記復号化クロックは、受信された符号化入力信号と同期化され、

前記符号化復号化装置は、

復号化クロックによって駆動されるフリップフロップを含む復号化モジュール

を含み、

復号化は、受信フェイズの間における、復号化されるべきマンチェスター符号化データの入力信号の簡単なサンプリングであり、

前記符号化復号化装置は、

マンチェスター符号内に符号化されるべきNRZ入力信号と伝送 / 符号化クロックとの間における“排他的OR”機能を実行する符号化モジュール

を含み、

同期信号生成モジュールは、マンチェスター符号化入力信号の第1の後縁の遷移に応答し、かつ、寄生パルスによるクロックの開始を回避するためにタイマを具備し、

前記低周波クロックは、読取器によって伝送される無線周波数信号の周波数でのクロックから得られ、

前記伝送 / 符号化クロックは、伝送速度を固定し、かつ、マンチェスター符号化で使用される

ことを特徴とするマンチェスター符号化復号化装置。

【請求項2】

伝送 / 符号化クロックおよび復号化クロックを所定の伝送周波数で作動させることを特徴とする請求項1に記載のマンチェスター符号化復号化装置。

10

【請求項3】

符号化クロックは、周波数分周器を使用して、高周波クロック信号から生成され、符号化クロックは、受信の間、符号化信号に再同期化され、符号化クロックの位相調整が、受信信号の第1の後縁において実行され、前記第1の後縁は、符号化クロックの生成のための時間ゼロとして取得され、復号化クロックは、符号化クロックを生成する同じ周波数分周器を使用して、高周波信号から生成され、

復号化クロックは、特定の遅延を有し、

復号化は、復号化クロックの前縁において入力信号をサンプリングすることによって生じる

20

ことを特徴とする請求項2に記載のマンチェスター符号化復号化装置。

【請求項4】

前記伝送 / 符号化クロックおよび前記復号化クロックが9600ポーで作動することを特徴とする請求項2に記載のマンチェスター符号化復号化装置。

【請求項5】

アナログヘッドと、メモリと、伝送 / 受信プロトコル及びメモリでの読取 / 書込を管理する論理部とを統合するASICにまた統合されるマクロセルの形式で作成されていることを特徴とする請求項1に記載のマンチェスター符号化復号化装置。

【請求項6】

プログラブル回路の形式で作成されていることを特徴とする請求項1に記載のマンチェスター符号化復号化装置。

30

【請求項7】

非接触カードの分野で使用されていることを特徴とする請求項1に記載のマンチェスター符号化復号化装置。

【請求項8】

電子ラベルの分野で使用されていることを特徴とする請求項1に記載のマンチェスター符号化復号化装置。

【請求項9】

同期信号生成器は、マンチェスター符号化信号の第1の後縁の遷移に応答し、かつ、寄生パルスにおける開始クロックを回避するためにタイマを具備し、

40

同期信号は、タイマによって定義される時間の後に生じる第1の後縁の遷移において生成される

ことを特徴とする請求項1に記載のマンチェスター符号化復号化装置。

【請求項10】

同一周波数を伴う2つのクロックが、周波数分周器を使用して、CLK信号から生成され、

第1のクロックは、符号化処理によって使用されている同期信号によって、受信されたマンチェスター符号化信号と再同期化され、

第2のクロックは、復号化処理において使用されている第1のクロックに対して特定の遅延を有し、

50

復号化は、復号化クロックの立ち上がりエッジにおいてマンチェスター符号化信号をサンプリングすることによって発生する

ことを特徴とする請求項 9 に記載のマンチェスター符号化復号化装置。

【請求項 11】

アナログフロントエンドと、メモリと、伝送 / 受信プロトコル及びメモリでの読取 / 書込動作を管理する論理部とを統合するASICに統合されるマクロセルの形式で作成されている

ことを特徴とする請求項 10 に記載のマンチェスター符号化復号化装置。

【請求項 12】

VHDL 高レベル記述言語を用いて設計され、かつ、標準セルライブラリにおいて合成されている

ことを特徴とする請求項 11 に記載のマンチェスター符号化復号化装置。

【請求項 13】

非接触カードおよび電子ラベルの分野で使用されていることを特徴とする請求項 12 に記載のマンチェスター符号化復号化装置。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

本発明は、マンチェスター符号化復号化装置に関するものである。

【0002】

【従来の技術】

本発明の分野は、全ての種類の非接触カードの分野である。すなわち、遠隔駆動又は電池駆動の、近接カード、中長距離カード、単周波数カード、多周波遠隔駆動カードである。本発明の分野もまた電子ラベルの分野である。交換プロトコルは、伝送情報の符号化とともに非接触カードとその読取器との間の伝送に使用されている。符号化の本質的な目的は、一連の“0”ビットの伝送を回避することにある。この場合、マンチェスター符号が最も使用される符号である。

【0003】

【発明が解決しようとする課題】

しかしこの符号は、カード側であろうが読取器側であろうがデータの復号化に関して特別の実践的な困難性を導入する。もしマンチェスター符号化の形成がありふれたものであるならば、それは“排他的OR”タイプの操作によって容易に実行されることができる。復号化操作は、非常に複雑で、復号化チェーンに遅延線の導入を必要とする。両方の伝送方向でカード側及び読取器側の伝送クロックの同期化に特別の注意をしなければならない。本発明の目的は、マンチェスター様式の符号化 / 復号化を実行する装置を提案することにある。大きな特徴は復号化部分にあり、この装置は、伝送クロックの生成及びその同期化の両方を確実にし、伝送クロックはアナログヘッドで再生されたHF周波数から生成されている。

【0004】

【課題を解決するための手段】

本発明のマンチェスター符号化復号化装置は、下記のモジュールを含んでいる。

同期信号生成モジュール： このモジュールは主に重要である。このモジュールは、クロック、特に復号器がT0として知られている基準時間に対してそれ自体を整列させることを可能にするクロックに対して開始信号を生じさせる。

クロックの同期・生成のためのモジュール： このモジュールは、伝送 / 符号化クロック及び復号化クロック並びに低周波クロックを生成する。これらのクロックは符号化復号化装置の内部で使用される。これらのクロックはまた、他の回路と符号化復号化装置とを同期化するために外部への出力を有する(受信、伝送等)。

復号化モジュール： マンチェスター復号化は、正確な復号化クロックを生成している前述のモジュールによって実行される。このクロックのエッジは、常に受信ビット上で

10

20

30

40

50

同一の場所に位置決めされている。この結果、復号化クロックによって駆動されるフリップフロップに減少される非常に簡単な復号器になる。要するに、復号化は、クロックの前縁で(間接符号のための後縁で)実行される受信フェーズ中、復号化されるべきマンチェスター符号化データの入力信号の簡単なサンプリングである。

符号化モジュール：このモジュールは、伝送フェーズ中にマンチェスター符号に符号化されるべきであるデータの入力信号と伝送クロックとの間の“排他的OR”機能を実行する。

【0005】

復号化されるべきデータだけで復号化には十分であることを特徴とし、符号化クロックに関するいかなる情報も復号化には必要でない。

本発明の符号化復号化装置は、所定の伝送周波数、例えば、9600ボーで2つのクロック、すなわち1つの符号化クロックと1つの復号化クロックによって作動する。

第1の符号化クロック(伝送中)は、分周器を使用してHF(高周波)信号から生成される。このクロックは、受信中符号化信号上で再同期化される。整相は、第2のクロックの形成のための時間ゼロ(T0)とみなされる受信信号の第1の後縁で実行される。

第2の復号化クロックはまた、符号化クロックを生成する同一分周器を使用してHF信号から生成され、このクロックは符号化クロックに対して、例えば1/4ビットの特定された遅延を有している。復号化は、復号化クロックの前縁で入力ラインをサンプリングすることによって生じる。間接マンチェスター符号の場合、遅延は $3T/4$ であることが有り得る。

【0006】

本発明の符号化復号化装置は、ASIC(“特定用途向けIC”)に統合されるマクロセルの形式で作成されている。本発明の目的であるモジュールとは別に、ASICは、アナログヘッドと、例えば、EEPROM(“電氣的消去プログラマブルROM)型式のメモリと、伝送/受信プロトコル及びメモリでの読み出し/書き込みを管理する論理部とを統合する。このASICは2つのアプリケーション、すなわち電子ラベル及び非接触カードで使用されている。

本発明のモジュールはまた、プログラマブル回路(FPGA:“フィールドプログラマブルゲートアレイ”又はプログラマブル構成要素)の形式で作成されている。この形式では、それは読取器側で使用されることができ。

確かに、本発明の符号化復号化装置は、データの2つのフェーズ符号化を使用するいかなる非接触システムに対しても関わりがある。その顕著な点は、下記である。

それは完全にデジタルである。その形成は非常に簡単である。伝送クロックの生成及び復号化は同一の資源を使用する。この結果、低電力消費を有する非常にコンパクトなモジュールを生じる。

非接触カード(又は電子ラベル)は常に読取器と同期化される。原理は復号化システムと同一である。

使用することが簡単である。システムの両側(読取器及びカード)のためのマイクロプロセッサ及びケーブル接続されたロジックによって同様に広範囲に使用されることができ。

【0007】

【発明の実施の形態】

図1は、本発明の符号化復号化装置の概略図を示す。図1は、8つの入出力を有する装置10を有する。すなわち、

読取器によって伝送される無線周波数信号の周波数すなわちRFでのクロックCLK。この信号は、アナログインタフェースによるRF信号の回復/再生の結果である。クロックCLKの周波数は例として示される場合では4.9MHzである。この周波数は非常に高いが、より低いこともまたあり得る。周波数制限は本発明の符号化復号化装置の形成のせいではない。このクロックCLKは、本発明の符号化復号化装置で使用される全ての他のクロックを生成するために使用される。すなわち、

受信フェーズ中及び復号化されることを予定されているマンチェスター符号化データの入力 S I。

- 伝送フェーズ中にマンチェスター符号に符号化されることを予定されているデータの入力 S O N R Z

【 0 0 0 8 】

入力 S I で受信され、本発明の符号化復号化装置によって復号化されたデータの出力 S I _ N R Z。

入力 S O _ N R Z で受信されたデータ及び本発明の符号化復号化装置によってマンチェスター符号化されたデータの出力 S O。

クロック C L K から生成され、論理の残りを作動するために使用されるクロック H _ S Y S T E M。このクロックの周波数は、クロック C L K の周波数に対して低い。このクロック C L K の周波数は、回路(その電力消費が周波数とともに増加する C M O S 技術で製造された回路)のエネルギーバランスを改善する。考察の場合では周波数は 1 5 0 K H z である。

クロック H _ S Y S T E M から生成された伝送クロック H _ T R A N S。それは、伝送速度(考察された場合では 9 6 0 0 ボー)並びにマンチェスター符号化をクロックするために使用される。

クロック H _ S Y S T E M から生成されたマンチェスター復号化クロック H _ D E C O R D I N G (考察される場合では 9 6 0 0 H z)。

【 0 0 0 9 】

本発明の符号化復号化装置の伝送速度は 9 6 0 0 ボーに制限されない。符号化復号化装置は、最も使用される伝送速度(9 6 0 0、 1 9 2 0 0 又はいかなる他の速度も)の範囲によって使用されることができる。

図 1 で示されるように、本発明のマンチェスター符号化復号化装置は、下記のモジュールを含んでいる。

・同期信号生成モジュール 1 1 : このモジュールは、最も重要性を有する。それは、クロック、特に、復号器が T 0 として既知の基準時間に対してそれ自体を整列させる復号化クロック H _ D E C O R D I N G のための開始信号を与える。

同期信号(R A Z _ _ H _ 9 6 0 0)を生成するために、このユニットはライン S I を連続的に監視する。それは“ 1 ”から“ 0 ”への遷移を検出する瞬間から、それはタイマをトリガし、例えば、 1 m s の間、スタンバイにそれ自体をリセットする。 1 m s 後に到達する“ 1 ”から“ 0 ”への第 1 の遷移は、基準時間 T 0 に対してそれを与える。

この 1 m s の待機時間は、寄生遷移で同期化することを回避することを可能にする。 : この時間遅延は調整可能であり、安定信号を確定する面からの復調システム性能に応じて、それはより大きくも、又はより小さくもあり得る。

【 0 0 1 0 】

・クロックの同期化及び生成のためのモジュール 1 2 : このモジュールは、伝送 / 符号化クロック及び復号化クロック並びに低周波クロックを生成する。これらのクロックは、符号化復号化装置の内部で使用される。これらのクロックはまた、符号化復号化装置とともに作動する他の回路を同期化するために外部への出力(受信、伝送等)を有する。すなわち、

【 0 0 1 1 】

簡単なカウンタ割算器(この場合では 3 2 による割算)によってクロック C L K から得られたクロック H _ S Y S T E M : それは信号 R E S E T による初期設定直後に開始する固定クロックである。

伝送クロック H _ T R A N S : それは、伝送速度を固定し、マンチェスター符号化で使用される。それは、第 1 のモジュール 1 1 によって生成された同期信号によってゼロにリセットされるカウンタ割算器(1 6 で割算)によってクロック H _ S Y S T E M から生成される(R A Z _ _ H 9 6 0 0 = “ 1 ” は、カウンタを強いて 0 にする。 R A Z _ _ H 9 6 0 0 が “ 0 ” になるや否やカウンタは解放される)。次に、クロック H _ T R A N S は、開始

10

20

30

40

50

時間T0から作動し、信号RAZ__H9600の各パルスで再同期化される。

復号化クロックH_DECORRING：それはマンチェスター復号化のためだけに使用される。それは、H_TRANSと同様に及び同一カウンタとともに生成される。パルスRAZ__H9600 = "1"であるとき、それはクロックH_TRANSと同時に使用不可能にされるが、信号RAZ__H9600が"0"になるとき、H_TRANSに対してある遅延で開始する。この遅延は、直接マンチェスター符号の場合では1ビットの持続期間の1/4である。それは、間接マンチェスター符号の場合では1ビットの持続期間の3/4である(9600ボーでの1ビットの持続期間は104μsである)。2つの前(後)縁を分離する遅延は常に同一である。クロックH_TRANSの周波数シフトもまた同一比率でクロックH_DECORRINGに影響を与える。

10

クロックCLKは、読取器の中及びカードの中の両方で使用可能である。符号化復号化装置が両方の側で使用されるならば、信号CLKから同様に生成されている伝送クロックH_TRANSは2つの装置では同期している。符号化復号化装置がカード側で使用されるだけである場合、カードクロックH_TRANSは読取器クロックによって制御される。両方の場合、符号化復号化装置側の同期は常に保証される。

【0012】

・復号化モジュール13：マンチェスター復号化は、正確な復号化クロックH_DECORRINGを生成する前述のモジュールによって広大な範囲まで実行される。このクロックのエッジは常に受信ビットの同一桁に位置決めされる。この結果、復号化クロックによって駆動されるフリップフロップに減少される非常に簡単な復号器となる。実際は、復号化は、クロックの前縁で(間接コードのための後縁で)実行される受信フェーズ中、復号化される予定のマンチェスター符号化データの入力信号SIの簡単なサンプリングである。

20

・符号化モジュール14：このモジュールは、伝送フェーズ中、マンチェスター符号に符号化されるべきであるデータの入力信号SONRZと伝送クロックH_TRANSとの間の“排他的OR”機能を実行する。

【0013】

本発明の符号化復号化装置10は、所定の伝送周波数、例えば、9600ボーで2つのクロック、1つの符号化クロックと1つの復号化クロックとで作動する。

第1のクロックは、符号化のために使用される(伝送中)。それは、分周器を使用してHF信号から生成される。このクロックは、受信中、符号化信号上に再同期化される。整相は、第2のクロックの形成のための時間ゼロ(T0)とみなされる受信信号の第1の後縁で実行される。

30

復号化クロックもまた、符号化クロックを生成する同一の分周器を使用してHF信号から生成され、このクロックは、例えば、符号化クロックに対して1/4ビットの特定遅延を有する。このクロックの整相は、時間T0+T/4で実行される。復号化は、図2に示されるような復号化クロックの前縁上で入力ラインをサンプリングすることによって生じる。

【0014】

本発明の符号化復号化装置の作成は非常に簡単である。そのことは、その独創性及びその重要性を符号化復号化装置に与えている。

40

本発明の符号化復号化装置は、ハードウェア記述言語VHDL(VERY High Speed Integrated Circuit Hardware Description Language“超高速集積回路ハードウェア記述言語”)で作成されている。その働きはシミュレートされ、その動作は確認されている。ハードウェアレベルでの実現は“標準セル”のライブラリ内の合成によって実行されている。図1のモジュールへの分割は、VHDLで記述された操作上の分割に正確に相当する。VHDL言語での記述は下記の付表に示されている。

【0015】

付表

VHDL言語による記述

I. 本発明の符号化／復号化の概要、および各モジュールへのブレイクダウン

```

library IEEE;
use IEEE.std_logic_1164.all;
use IEEE.std_logic_arith.all;
entity CODEC_M is
    Port (CLK          : in  STD_ULOGIC;
          H_TRANS     : out STD_ULOGIC;
          H_SYSTEM    : out STD_ULOGIC;
          H_DECODING  : out STD_ULOGIC;
          RESET       : in  STD_ULOGIC;
          SI_NRZ      : out STD_ULOGIC;
          SO_NRZ      : in  STD_ULOGIC;
          SI          : in  STD_ULOGIC;
          SO          : out STD_ULOGIC);
end CODEC_M
architecture SCHEMATIC of CODEC_M is
    signal H_TRANS_L      : STD_ULOGIC;
    signal RAZ_H_9600     : STD_ULOGIC;
    signal H_DECODING_L  : STD_ULOGIC;
    signal H_SYSTEM      : STD_ULOGIC;
component SYNCHRO
port (RESET          : in  STD_ULOGIC;
      H_SYSYSTEM    : in  STD_ULOGIC;
      SI            : in  STD_ULOGIC;
      RAZ_H_9600    : in  STD_ULOGIC;
end component
component CLOCKS

```

```

port (RESET      : in      STD_ULOGIC ;
      RAZ_H_9600 : in      STD_ULOGIC ;
      CLK        : in      STD_ULOGIC ;
      H_DECODING : out     STD_ULOGIC ;
      H_SYSTEM   : buffer  STD_ULOGIC ;
      H_TRANS    : out     STD_ULOGIC) ;
end component ;
component CODER
port (SO          : out    STD_ULOGIC ;
      SO_NRZ      : in     STD_ULOGIC ;
      H_TRANS     : in     STD_ULOGIC ;
      RESET       : in     STD_ULOGIC) ;
end component ;
component DECODER
port (SI          : in     STD_ULOGIC ;
      SI_NRZ      : out    STD_ULOGIC ;
      H_DECODING : in     STD_ULOGIC ;
      RESET       : in     STD_ULOGIC) ;
end component ;
begin
    H_TRANS    ← H_TRANS_L ;
    H_DECODING ← H_DECODING_L ;
    H_SYSTEM   ← H_SYSTEM_L ;
I__1 : SYNCHRO
    port map (RESET      → RESET ;
              H_SYSTEM   → H_SYSTEM_L ;
              SI         → SI ;
              RAZ_H_9600 → RAZ_H_9600) ;
I__2 : CLOCKS

```

```

    port map (RESET      →RESET;
              RAZ_H_9600 →RAZ_H_9600;
              CLK        →CLK;
              H_DECODING →H_DECODING_L;
              H_SYSTEM   →H_SYSTEM_L);
    H_TRANS →H_TRANS_L;
I__4:CODER
    port map (SO      →SO;
              SO_NRZ  →SI_NRZ;
              H_TRANS →H_TRANS_L;
              RESET   →RESET);
I__5:DECODER
    port map (SI      →SI;
              SI_NRZ  →SI_NRZ;
              H_DECODING →H_DECODING_L;
              RESET   →RESET);
end SCHEMATIC
-- pragma translate_off
configuration CFG_CODEC_M_BEHAVIORAL
  of DECORDER is for BEHAVIORAL
end for;
end CFG_CODEC_M_BEHAVIORAL;
[0016]

```

10

20

30

IIー同期信号生成モジュール (11)

```

library IEEE;
  use IEEE.std_logic_1164.all;
  use IEEE.std_logic_arith.all;
entity SYNCHRO is
  port (RESET      :in  STD_ULOGIC;
        H_SYSTEM   :in  STD_ULOGIC;
        SI         :in  STD_ULOGIC;
        RAZ_H_9600:out  STD_ULOGIC;
end SYNCHRO
architecture BEHAVIOAL of SYNCHRO is
  -----
P_synchro:process (RESET, H_SYSTEM, SI)
  type TYPE_STATE is (E_INIT,
  E_SYNC0, E_SYNC1, E_SYNC2, E_SYNC3,
  E_SYNC4, E_SYNC5,);
  constant K_1ms:INTEGER:=154;--delay
  of 1ms:1ms*153.6khz
  variable STATE :TYPE_STATE;
  variable K:INTEGER range 0 to 2**9-1;
  --Lmax>K_2_7ms
  begin
    if RESET=`0` then
      RAZ_H_9600 ← `1`
      STATE:=E_INIT
    elseif H_SYSTEM=`1` H_SYSTEM event
    then case STATE is
      when E_INIT → RAZ_H_9600 ← `0`;
                   STATE := E_SYNC0

```

```

;
_____DELAY FIRST TRANSITION TO `0`
OF SI

    when E_SYNC0 → if SI= `1` then
                        STATE := E_SYNC1;
                        end if;
    when E_SYNC1 → if SI=`0` then
                        K:=K_1ms;-delay t
o                        avoid triggering
                        -on parasitic transitions
                        STATE := E_SYNC2;
                        end if;
    when E_SYNC2 → if K= `0` then
                        STATE := E_SYNC3;
                        else
                        K:= K-1
                        end if;
    when E_SYNC3 → if SI= `1` then
                        STATE := E_SYNC4;
                        end if;
    when E_SYNC4 → if SI= `0` then
                        RAZ_H_9600 ← `1`
                        STATE := E_SYNC5;
                        end if
    when E_SYNC5 → RAZ_H_9600 ← `0`;
                        STATE := E_SYNC0
;    end case;
end if;

```

10

20

30

40

```
end process P_synchro
_____END PROCESS SYNCHRO_____
end BEHAVIORAL;
--pragma translate_off
configuration CFG_SYNCHRO_BEHAVIORAL
of SYNCHRO is for BEHAVIORAL
end for;
end CFG_CYNCHRO_BEHAVIORAL;
【0017】
```

III. クロック発生モジュール(12)

```

library IEEE;
    use IEEE.STD_LOGIC_1164.all;
entity CLOCK is
port (RESET      : in      STD_ULOGIC;
      RAZ_H_9600 : in      STD_ULOGIC;
      CLK        : in      STD_ULOGIC;
      H_DECODING : out     STD_ULOGIC;
      H_SYSTEM   : buffer  STD_ULOGIC;
      H_TRANS    : out     STD_ULOGIC);
end CLOCKS;
architecture BEHAVIORAL of CLOCKS
is
    constant K_BIT: Integer :=16;
    us*4.9152mhz:
        duration of 1 bit at 9600 bauds
    constant K_HOR: Integer :=32;
begin
P_H_SYSTEM : process (RESET, CLK)
    variable K_153KHZ: INTEGRER range
0 to K_HOR-1;
begin
    if RESET = `0` then
        K_153KHZ :=0;
        H_SYSTEM ← `1`;
    elsif CLK=`1` and CLK' event then
        if K_153KHZ=0 then
            H_SYSTEM←`0`
            K_153KHZ :=K_HOR-1;

```

```

e l s i f  K_153KHZ =K_HOR/2  t h e n
    H_SYSTEM ← `1`;
    K_153KHZ :=K_153KHZ-1;
e l s e
    K_153KHZ :=K_153KHZ-1;
e n d  i f ;
e n d  i f ;
e n d  p r o c e s s  P_H_SYSTEM
P_CLOCKS:process (RESET, H_SYSTEM,
RAZ_H_9600)
    variable K :Integer range 0 to
K_BIT-1;
b e g i n
    i f  RAZ_H_9600=`1` t h e n
        K:=K_BIT61;
        H_DECODING←`0`;
        H_TRANS ←`0`;
    e l s e i f  H_SYSTEM=`1` a n d  H_SYSTEM
e v e n t  t h e n
        i f  K=K_BIT-1  t h e n
            H_TRANS ←`1`; -start clock
h_9600
            H_DECODING←`0`;
            K :=K-1;
        e l s i f  K = 3*K_BIT/4+1  t h e n
            H_DECODING←`1`; -start decoding
c l o c k
            K :=K-1;

```

```

e l s i f K = K _ B I T / 2 - 1 t h e n
    H _ T R A N S   ← ` 0 ` ; - T / 2 H _ T R A N S
    K := K - 1 ;
e l s i f K = K _ B I T / 4 + 1 t h e n
    H _ D E C O D I N G ← ` 0 ` ; - T / 2 d e c o r d i n g
c l o c k
    K := K - 1 ;
e l s i f K = 0 t h e n
    K = K _ B I T - 1
e l s e
    K := K - 1 ;
e n d i f ;
e n d i f ;
e n d p r o c e s s P _ C l o c k s ;
e n d B E H A V I O R A L ;
-- p r a g m a t r a n s l a t e _ o f f
c o n f i g u r a t i o n C F G _ C L O C K S _ B E H A V I O R A L
o f C L O C K S i s f o r B E H A V I O R A L
e n d f o r ;
e n d C F G _ C L O C K S _ B E H A V I O R A L ;
[ 0 0 1 8 ]

```

10

20

30

IV. 復号化モジュール (13)

```

library IEEE;
    use IEEE.std_logic_1164.all;
    use IEEE.std_logic_arith.all;
entity DECODER is
    Port (SI      :IN  STD_ULOGIC;
          SI_NRZ   :OUT STD_ULOGIC;
          H_DECODING:IN  STD_ULOGIC;
          RESET    :IN  STD_ULOGIC);
end DECODER;
architecture BEHAVIORAL of DECODER
is begin

    --SI      :Input of coded data
    --SO_NRZ  :Output of decoded data
    --H_DECODING :Transmission clock
    adopted
    --RESET   :Initialisation of the
circuit

    =====Manchester Type Decoding

    -- The decoding is carried out by
    sampling the signal code SI on the
    leading
    -- edge of the decoding clock H_
DECODING
    P_DECODING :process
    begin

```

```
wait until H DECODING=`1` ;
SI NRZ ←SI
end process P DECODING;
end BEHAVIORAL;
--pragma translate_off
configuration CFG_DECORDER_
BEHAVIORAL of DECORDER is for
BEHAVIORAL
end for;
end CFG_DECORDER_BEHAVIORAL;
[0019]
```

V. 符号化モジュール(14)

```

library IEEE;
    use IEEE.std_logic_1164.all;
    use IEEE.std_logic_arith.all;
entity CODER is
Port (SO          :OUT  STD_ULOGIC;
      SO_NRZ      :IN   STD_ULOGIC;
      H_TRANS     :IN   STD_ULOGIC;
      RESET       :IN   STD_ULOGIC);
end CODER;
architecture BEHAVIORAL of CODER is
begin
--SO          :Output of coded data
--SO_NRZ      :Input of data to be
               encoded
--H_TRANS     :Transmission clock
               adopted
--RESET       :Initialisation of the
               circuit

=====Manchester Type Decoding

--encoding by an "OR EXCLUSIVE"
  function between the transmission
  clock
--and the signal to be encoded,
  the function "NOT" is used to
  implement
--the direct Manchester code

```

10

20

30

40

(a`0` is coded by "01" ; a`1` by "10"

```

SO←not (H_TRANS xor SO_NRZ) ;
end BEHAVIORAL ;
--pragma translate off
configuration CFG_CODER_BEHAVIORAL
of CODER is for BEHAVIORAL
end for ;
end CFG_CODER_BEHAVIORAL ;

```

10

【0020】

本発明の符号化復号化装置は、ASICに統合されるマクロセルの形式で作成されている。本発明の目的であるモジュールとは別に、ASICは、アナログヘッドと、例えば、EEPROM型式のメモリと、伝送/受信プロトコル及びメモリでの読み出し/書き込みを管理する論理部とを統合する。このASICは2つのアプリケーション、すなわち電子ラベル及び非接触カードで使用されている。

20

本発明のモジュールはまた、プログラマブル回路(FPGA)の形式で作成されている。この形式では、それは読取器側で使用されることができる。

作成された本発明の符号化復号化装置を作成する例では下記のパラメータを有する。

1 . 87MHzでのHFクロック。

9600Hzでの伝送クロック。

直接マンチェスター符号：“1”ビットは、2ビット“10”で符号化される。“0”ビットは2ビット“01”で符号化される。符号化マンチェスタービットの持続時間は、非符号化ビットの持続時間の半分に等しい。したがって、伝送の実際の速度は19200ボーである（他方では、間接マンチェスターは“1”ビットを“01”として符号化し、“0”を“10”として符号化する）。

30

【図面の簡単な説明】

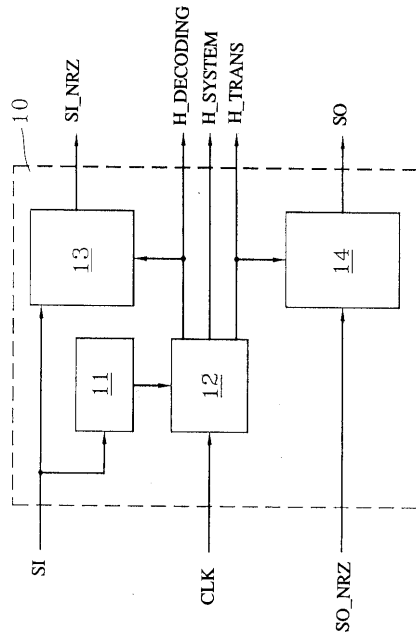
【図1】本発明の符号化復号化装置を示す。

【図2】符号化クロック及び復号化クロックのタイミング図を示す。

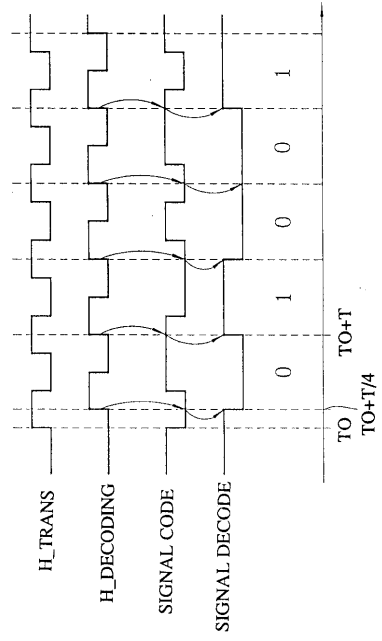
【符号の説明】

- 1 1 同期信号生成モジュール
- 1 2 クロックの同期及び生成のためのモジュール
- 1 3 復号化モジュール
- 1 4 符号化モジュール

【 図 1 】



【 図 2 】



フロントページの続き

(72)発明者 サルマン・アブー・ハッサン
フランス・14000・ケーン・リュ・ルシアン・ネル・29

合議体

審判長 石井 研一

審判官 阿部 弘

審判官 萩原 義則

(56)参考文献 特開平3-267821(JP,A)
特開昭63-139428(JP,A)
特開平6-96300(JP,A)
特開平6-11299(JP,A)
特開昭60-145745(JP,A)
特開昭61-208318(JP,A)
特開昭62-290228(JP,A)
特開平5-83237(JP,A)

(58)調査した分野(Int.Cl., DB名)

H03M 5/12

H04L 25/49