

(19) 日本国特許庁(JP)

(12) 特許公報(B2)

(11) 特許番号

特許第5800525号
(P5800525)

(45) 発行日 平成27年10月28日(2015.10.28)

(24) 登録日 平成27年9月4日(2015.9.4)

(51) Int.Cl.

G06K 7/14 (2006.01)

F 1

G06K 7/14 O 6 O

請求項の数 14 (全 44 頁)

(21) 出願番号 特願2011-35162 (P2011-35162)
 (22) 出願日 平成23年2月21日 (2011.2.21)
 (65) 公開番号 特開2012-173973 (P2012-173973A)
 (43) 公開日 平成24年9月10日 (2012.9.10)
 審査請求日 平成25年12月12日 (2013.12.12)

(73) 特許権者 391062872
 株式会社オプトエレクトロニクス
 埼玉県蕨市塚越4丁目12番17号
 (74) 代理人 110001209
 特許業務法人山口国際特許事務所
 (72) 発明者 木村 一人
 埼玉県蕨市塚越4-12-17 株式会社
 オプトエレクトロニクス内
 (72) 発明者 早川 浩
 埼玉県蕨市塚越4-12-17 株式会社
 オプトエレクトロニクス内
 審査官 岡北 有平

最終頁に続く

(54) 【発明の名称】光学情報読取装置、光学情報読取方法、コンピュータが読み取り可能なプログラム及び記録媒体

(57) 【特許請求の範囲】

【請求項 1】

1 キャラクタ内で隣接するバー及びスペースを含む n (n ≥ 2) 本のエレメントの幅を加算して 2 ~ n 本加算エレメント幅を各々求め、1 キャラクタ幅に対する前記 2 ~ n 本加算エレメント幅の各々の比率を算出する演算手段と、

前記演算手段によって算出された前記比率から、前記 2 ~ n 本加算エレメント幅の各々のモジュール数が誤差 1 以内の整数となる対象キャラクタを絞り込んで候補キャラクタを抽出する抽出手段と、

前記抽出手段によって抽出された各々の前記 2 ~ n 本加算エレメント幅のモジュール数の候補キャラクタと、当該候補キャラクタを評価するための期待値キャラクタとを比較する比較手段と、

前記候補キャラクタと最も相関が強い期待値キャラクタを検索する検索手段とを備え、前記演算手段は、

当該対象キャラクタの端の前記 2 ~ n 本加算エレメント幅を各々求める際に、

前記対象キャラクタの前後のキャラクタのバー及びスペースを含むエレメントの幅を 1 本以上 n - 1 本以下加算範囲を拡張することを特徴とする光学情報読取装置。

【請求項 2】

前記抽出手段は、

当該対象キャラクタの端の前記 2 ~ n 本加算エレメント幅を各々求める際に、

前記 1 キャラクタ分の n 本の幅データと、当該対象キャラクタの前後の 1 本以上 n - 1

10

20

本以下の幅データとを抽出することを特徴とする請求項 1 に記載の光学情報読取装置。

【請求項 3】

前記検索手段は、

前記抽出手段から得られた複数の前記対象キャラクタの各々の前記 2 ~ n 本加算エレメント幅に対して 2 種類の前記対象キャラクタの何れかと一致する対象キャラクタを検索し、前記候補キャラクタの数を絞り込むことを特徴とする請求項 1 に記載の光学情報読取装置。

【請求項 4】

前記検索手段は、

前記候補キャラクタのモジュール数の実測値と、前記期待値キャラクタのモジュール数の理想値との間の誤差を算出し、前記誤差を合計して得られる誤差値が最小となる前記候補キャラクタと最も相関が強い期待値キャラクタを検索することを特徴とする請求項 1 に記載の光学情報読取装置。

10

【請求項 5】

前記検索手段は、

前記誤差値が最小となる前記候補キャラクタと一致する前記期待値キャラクタを検索することを特徴とする請求項 4 に記載の光学情報読取装置。

【請求項 6】

前記 1 キャラクタ内で隣接するバー及びスペースを含む n (n - 2) 本のエレメントの幅を加算した合計値と、前記 1 キャラクタの全幅を示す全幅値とを比較し、比較結果に基づいて当該 1 キャラクタ分の幅データが適切か否かを判別する比較手段を備えることを特徴とする請求項 1 に記載の光学情報読取装置。

20

【請求項 7】

光学情報読取装置が、

1 キャラクタ内で隣接するバー及びスペースを含む n (n - 2) 本のエレメントの幅を加算して 2 ~ n 本加算エレメント幅を各々求めるステップと、

求めた前記 2 ~ n 本加算エレメント幅の各々の、 1 キャラクタ幅に対する比率を算出するステップと、

算出した前記比率から、前記 2 ~ n 本加算エレメント幅の各々のモジュール数が誤差 1 以内の整数となる対象キャラクタを絞り込んで候補キャラクタを抽出するステップと、

30

抽出された各々の前記 2 ~ n 本加算エレメント幅のモジュール数の候補キャラクタと、当該候補キャラクタを評価するための期待値キャラクタとを比較し、当該候補キャラクタと最も相関が強い期待値キャラクタを検索するステップとを実行し、

当該対象キャラクタの端の前記 2 ~ n 本加算エレメント幅を各々求める際に、

前記対象キャラクタの前後のキャラクタのバー及びスペースを含むエレメントの幅を 1 本以上 ~ n - 1 本以下加算範囲を拡張することを特徴とする光学情報読取方法。

【請求項 8】

当該対象キャラクタの前記 2 ~ n 本加算エレメント幅を各々求める際に、

前記 1 キャラクタ分の n 本の幅データと、当該対象キャラクタの前後の 1 本以上 n - 1 本以下の幅データとを抽出することを特徴とする請求項 7 に記載の光学情報読取方法。

40

【請求項 9】

抽出された複数の前記対象キャラクタの各々の前記 2 ~ n 本加算エレメント幅に対して 2 種類の前記対象キャラクタの何れかと一致する対象キャラクタを検索し、前記候補キャラクタの数を絞り込むことを特徴とする請求項 8 に記載の光学情報読取方法。

【請求項 10】

前記候補キャラクタのモジュール数の実測値と、前記期待値キャラクタのモジュール数の理想値との間の誤差を算出し、前記誤差を合計して得られる誤差値が最小となる前記候補キャラクタと最も相関が強い期待値キャラクタを抽出することを特徴とする請求項 7 に記載の光学情報読取方法。

【請求項 11】

50

前記誤差値が最小となる前記候補キャラクタと一致する前記期待値キャラクタを検出する
ことを特徴とする請求項 10 に記載の光学情報読み取り方法。

【請求項 12】

前記 1 キャラクタ内で隣接するバー及びスペースを含む n (n ≥ 2) 本のエレメントの
幅を加算した合計値と、前記 1 キャラクタの全幅を示す全幅値とを比較し、比較結果に基
づいて当該 1 キャラクタ分の幅データが適切か否かを判別することを特徴とする請求項 7
に記載の光学情報読み取り方法。

【請求項 13】

請求項 7 乃至 12 に記載の光学情報読み取り方法を実行するための指令を組み合わせたこと
を特徴とするコンピュータが読み取り可能なプログラム。

10

【請求項 14】

請求項 7 乃至 12 に記載の光学情報読み取り方法を実行するためのプログラムが記述されて
いることを特徴とするコンピュータが読み取り可能な記録媒体。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、バーコード記号を読み取ってバーコードデータを出力するバーコード記号読み取り装置に適用可能な光学情報読み取り装置、光学情報読み取り方法、コンピュータが読み取り可能なプログラム及び記録媒体に関するものである。

【背景技術】

【0002】

従来から、物品流通システムにおいて、流通物品にバーコード記号を貼付又は印刷し、この種のバーコード記号をバーコードリーダー（以下光学情報読み取り装置という）等により読み取って情報処理するシステムが利用されている。光学情報読み取り装置によれば、バーコード記号 1 を読み取って、そのスタートマージンを検索し、更に、スタートマージンに続くキャラクタをデコード処理してバーコードデータを出力するようになされる。

20

【0003】

ところで、バーコード記号の規格には、CODE 128 や、CODE 39、EAN/U
 PC (EAN / Universal Product Code)、インターリーブド 2 オブ 5、コーダバー、
 RSS、RSSリミテッド、RSSエクスパンデド等の各々の方式がある。一方、バーコード記号は流通物品に対して、バーコード記号標を貼付し、又は、パッケージに直接印刷して使用される場合が多い。しかし、これらの印刷技術や、貼付技術等のバラツキや、流通物品の取り扱い方法によって流通物品に貼付又は印刷されたバーコード記号を正しく読み取れない場合が想定される。バーコード記号を正しく読み取れない原因として、バーコード記号を構成するバーやスペースの欠落や、傷等が挙げられる。

30

【0004】

この種の光学情報読み取り装置に関する特許文献 1 にはバーコード記号読み取り装置、それによるスタートマージン候補検出方法並びにバーコード記号読み取り方法が開示されている。このバーコード記号読み取り装置を応用して、バーコード記号を構成するバーやスペースに不本意に生じた、欠落や傷等を含む難読キャラクタをデコード処理する場合について、図 1
 4 ~ 図 17 を参照してその説明をする。

40

【0005】

図 14 は、従来例に係る光学情報読み取り装置 20 の構成例を示すブロック図である。図 1
 4 に示す光学情報読み取り装置 20 は、光学読み取り部 2、クロック発生部 3、m 段データバッファ 5、x 倍器 7、セレクタ部 6、比較器 8、スタートマージン格納部 9、インターフェース部 11、デコードバッファ 25、カウントデータ値格納部 30、CPU 33 及び、バーコードデータ値用のカウンタ部 40 を有して構成される。

【0006】

光学読み取り部 2 は、読み取り対象であるバーコード記号 1 を読み取って、コード読み取りデータ D 2 を発生する。例えば、光学読み取り部 2 からバーコード記号 1 に対して、発光ダイオード

50

ド(L E D)やレーザダイオード(L D)等の光源による光を照射し、その反射光をフォトダイオードやフォトトランジスタ等の受光素子で受光して電気信号に変換し、その電気信号を2値化してバーコード記号1の濃淡、すなわち、黒部分のバー又は白部分のスペースを示すデジタルのコード読取データD2に変換する。

【0007】

光学読取部2はカウンタ部40に接続される。カウンタ部40にはクロック発生部3が接続される。クロック発生部3は、所定周波数のサンプリングクロック信号(以下単にCLK信号という)を発生して、CLK信号をカウンタ部40に出力する。クロック発生部3には発振器が使用される。

【0008】

バーコードデータ値用のカウンタ部40は、光学読取部2から出力されるコード読取データD2を入力して、クロック発生部3から出力されるCLK信号を入力し、当該CLK信号に基づいてコード読取データD2をカウントし、コード読取データD2を白部分のスペースおよび黒部分のバーを含むエレメントの幅を示すカウントデータD4(バーコードデータ値:数値)に変換する。

【0009】

カウンタ部40にはm段データバッファ5、 \times 倍器7及びカウントデータ値格納部30が接続される。m段データバッファ5はカウンタ部40から出力されるカウントデータD4を、m段分(mは任意の整数)だけ一時格納する。m段は、バーコード記号1の印刷状態により任意に設定される。m段データバッファ5はフリップフロップ回路やメモリ等の記憶回路から構成される。

【0010】

\times 倍器7は、カウントデータD4からスタートマージンを検索するに当たり、カウントデータD4を何倍かを示す「 \times 」が予め設定され、その \times 倍の演算を行って、 \times 倍データD7を出力する。 \times 倍器7はカウントデータD4を \times 倍する演算回路によって構成される。CPU33は、バーコード記号1の印刷状態に対応して、カウントデータD4を何倍にするかを \times 倍器7に設定する。

【0011】

m段データバッファ5にはセレクタ部6が接続される。セレクタ部6は、m段データバッファ5に一時格納されたm段分のカウントデータD4のうち、比較対象とするデータを選択する。セレクタ部6はCPU33等の制御を受けて、m段分のカウントデータD4のうち、どのカウントデータD4を比較するかを選択するようになされる。

【0012】

CPU33は、m段データバッファ5中の何段(k段)前までのカウントデータD4を比較対象として選択するかを、バーコード記号1の印刷状態、印刷汚れ、外的要因による汚れ等の状態に準じてセレクタ部6に設定される。この例では、m段データバッファ5から出力されるカウントデータD4が、今、カウンタ部40から出力されたカウントデータD4の値の何倍より大きいことをもってスタートマージン候補とするかが予め設定されている。

【0013】

上述のセレクタ部6及び \times 倍器7には比較器8が接続される。比較器8はセレクタ部6が選択した、m段データバッファ5中のk段前までの各カウントデータD4と、 \times 倍器7から出力される \times 倍データD7とを比較する。比較結果によって \times 倍器7からの \times 倍データD7の方が小さいときは、スタートマージン候補があつたと判定し、スタートマージン候補検索フラグ(以下でSMFデータD9という)を発生する。SMFデータD9はスタートマージン格納部9に出力される。

【0014】

比較器8にはスタートマージン格納部9が接続される。スタートマージン格納部9にはフリップフロップ回路のような記憶機能を有して構成される。スタートマージン格納部9には比較器8から出力されるSMFデータD9、そのとき比較対象としたカウントデータ

10

20

30

40

50

値 = バーコードデータ値 D 3 0 とが対応させて格納される。どのカウントデータ D 4 によってスタートマージン候補となったかが判るようにするためである。スタートマージン候補は、バーコード記号 1 におけるバーの手前に、ある一定以上の幅の白バーが存在しているときに発生される。

【 0 0 1 5 】

カウントデータ値格納部 3 0 は、カウンタ部 4 0 から順次出力されるカウントデータ D 4 を、時系列的にすべて格納して保存する。カウントデータ値格納部 3 0 は、フリップフロップ回路や、D R A M 等のような記憶回路から構成される。スタートマージン格納部 9 及びカウントデータ値格納部 3 0 にはインターフェース部 1 1 が接続される。

【 0 0 1 6 】

インターフェース部 1 1 は、スタートマージン格納部 9 から読み出される S M F データ D 9 や、カウントデータ値格納部 3 0 から読み出した各キャラクタのバーコードデータ値 D 3 0 を C P U 3 3 に入力する。インターフェース部 1 1 にはデコードバッファ 2 5 が接続され、C P U 3 3 から出力されるバーコードデータ D 1 2 が格納される。

【 0 0 1 7 】

デコードバッファ 2 5 はカウントデータ値格納部 3 0 から出力された各キャラクタのバーコードデータ値 D 3 0 の演算途中のデータを格納もなされる。また、インターフェース部 1 1 には出力端子 OUT が接続され、出力端子 OUT にはデコード処理後のバーコードデータ D 1 2 が出力される。バーコードデータ D 1 2 は、例えば、デコードバッファ 2 5 からインターフェース部 1 1 を介して出力端子 OUT へ読み出される。

【 0 0 1 8 】

図 1 5 A ~ E は、従来例に係る 2 本加算データの算出例を示す説明図である。図 1 5 A に示す C L K 信号は図 1 4 に示したクロック発生部 3 からカウンタ部 4 0 へ出力される。図 1 5 B に示すバーコード記号 1 は、例えば、C O D E 1 2 8 規格のものである。C O D E 1 2 8 規格によれば、図 1 5 B に示す 1 キャラクタについて、3 本の黒部分のバー (1) , (3) , (5) 及び 3 本の白部分のスペース (2) , (4) , (6) から構成される。1 キャラクタの全幅は 1 1 モジュールから構成される。当該キャラクタ C 1 の黒部分のバー (1) は 4 モジュールから成り、バー (3) 及びバー (5) は各々 1 モジュールから成る。白部分のスペース (2) は 1 モジュールから成り、スペース (4) 及び (6) は各々 2 モジュールから成る。

【 0 0 1 9 】

図 1 5 C に示すバーコードデータ値 D 3 0 は、図 1 4 に示した光学読取部 2 から出力されるコード読取データ D 2 を、クロック発生部 3 から出力される C L K 信号に基づいてサンプリングし、当該 C L K 信号に基づいてコード読取データ D 2 をカウントし、コード読取データ D 2 を白部分のスペースおよび黒部分のバーを含むエレメントの幅を示すカウントデータ D 4 に変換することで得られる。

【 0 0 2 0 】

図 1 5 D に示す W 1 1 はキャラクタ C 1 の黒部分のバー (1) の幅である。W 1 2 はバー (1) に続くスペース (2) の幅である。W 1 3 はキャラクタ C 1 のスペース (2) に続くバー (3) の幅である。W 1 4 はキャラクタ C 1 のバー (3) に続くスペース (4) の幅である。W 1 5 はスペース (4) に続くバー (5) の幅である。W 1 6 はキャラクタ C 1 のバー (5) に続くスペース (6) の幅である。上述の 6 種類の幅を示すデータは 1 キャラクタのバーコードデータ値 D 3 0 を構成する。

【 0 0 2 1 】

図 1 5 E に示す W 2 1 はキャラクタ C 1 のバー (1) とスペース (2) の各々が C P U 3 3 によって加算された幅 (以下バー (1) + スペース (2) と記す) である。バー (1) + スペース (2) の加算幅を示すカウントデータ値は幅 W 2 1 データを成す。W 2 2 はキャラクタ C 1 のバー (1) に続くスペース (2) 及びバー (3) の各々が C P U 3 3 によって加算された幅である。スペース (2) + バー (3) の加算幅を示すカウントデータ値は幅 W 2 2 データを成す。

10

20

30

40

50

【0022】

W23はキャラクタC1のスペース(2)に続くバー(3)及びスペース(4)の各々がC P U 3 3によって加算された幅である。バー(3)+スペース(4)の加算幅を示すカウントデータ値は幅W23データを成す。W24はキャラクタC1のバー(3)に続くスペース(4)及びバー(5)の各々がC P U 3 3によって加算された幅である。スペース(4)+バー(5)の加算幅を示すカウントデータ値は幅W24データを成す。W25はキャラクタC1のスペース(4)に続くバー(5)及びスペース(6)の各々がC P U 3 3によって加算された幅である。バー(5)+スペース(6)の加算幅を示すカウントデータ値は幅W25データを成す。上述の5種類の幅W21データ～幅W25データは1キャラクタの2本加算方式のデータであり、バーコードデータ値D30を構成する。 10

【0023】

図16及び図17は、従来例に係る難読キャラクタのデコード例(その1,2)を示すフローチャートである。光学情報読取装置20では、C O D E 1 2 8規格に基づくバーコード記号1を読み取って、そのスタートマージンを検索し、更に、スタートマージンに続くキャラクタを2本加算方式によりデコード処理してバーコードデータD12を出力する場合を例に挙げる。その場合のバーコード記号1のバー及びスペースの各エレメントの誤差許容度(最大誤差範囲)が、例えば、±0.5モジュールに設定される。

【0024】

これらをデコード条件にして、図16に示すフローチャートのステップS T81でC P U 3 3はスタートマージンを検索する。このとき、光学読取部2は、バーコード記号1を読み取って、コード読取データD2を発生する。コード読取データD2は、光学読取部2からカウンタ部40へ出力される。クロック発生部3は、所定周波数のサンプリング用のC L K信号をカウンタ部40に出力する。 20

【0025】

カウンタ部40は光学読取部2から出力されるコード読取データD2を入力すると共に、クロック発生部3から出力されるC L K信号を入力し、C L K信号に基づいてコード読取データD2をカウントし、コード読取データD2を白部分のスペースおよび黒部分のバーを含むエレメントの2本加算幅を示すカウントデータD4に変換する。カウントデータD4は、バーコード記号1における各バー(黒部分)の幅およびスペース(白部分)の幅を示す数値データ列となる。カウントデータD4は、カウントデータ値格納部30に保存される。 30

【0026】

m段データバッファ5はカウンタ部40から出力されるカウントデータD4を、m段分(mは任意の整数)だけ一時格納する。m段は、バーコード記号1の印刷状態により任意に設定されている。x倍器7は、カウントデータD4からスタートマージンを検索するに当たり、カウントデータD4を何倍かを示す「x」が予め設定され、そのx倍の演算を行って、x倍データD7を出力する。 40

【0027】

セレクタ部6は、C P U 3 3の選択制御を受けて、m段データバッファ5に一時格納されたm段分のカウントデータD4のうち、比較対象とするデータを選択する。比較器8はセレクタ部6が選択した、m段データバッファ5中のk段前までの各カウントデータD4と、x倍器7から出力されるx倍データD7とを比較する。

【0028】

比較結果によってx倍器7からのx倍データD7の方が小さいときは、スタートマージン候補があったと判定し、スタートマージン候補検索フラグ(以下でS M FデータD9という)を発生する。S M FデータD9はスタートマージン格納部9に出力される。スタートマージン格納部9には比較器8から出力されるS M FデータD9、そのとき比較対象としたカウントデータ値=D10とが対応させて格納される。

【0029】

ステップS T82でC P U 3 3は1キャラクタ分の幅データ(6本)を抽出する。図15 50

に示したキャラクタC 1によれば、黒部分のバー(1)の幅W 1 1がカウンタ部4 0により計数され、そのカウントデータ値が幅W 1 1データとなる。バー(1)に続く白部分のスペース(2)の幅W 1 2もカウンタ部4 0により計数され、そのカウントデータ値が幅W 1 2データとなる。スペース(2)に続くバー(3)の幅W 1 3はカウンタ部4 0により計数され、そのカウントデータ値が幅W 1 3データとなる。

【0030】

このバー(3)に続くスペース(4)の幅W 1 4もカウンタ部4 0により計数され、そのカウントデータ値は幅W 1 4データとなる。スペース(4)に続くバー(5)の幅W 1 5はカウンタ部4 0により計数され、そのカウントデータ値が幅W 1 5データとなる。バー(5)に続くスペース(6)の幅W 1 6もカウンタ部4 0により計数され、そのカウントデータ値が幅W 1 6データとなる。上述の6種類の幅W 1 1データ～幅W 1 6データはバーコードデータ値D 3 0を構成する。バーコードデータ値D 3 0はインターフェース部1 1を介してCPU 2 3に出力される。

10

【0031】

ステップST83でCPU 3 3はキャラクタC 1の1モジュール幅を算出する。この例では、CPU 3 3は、1キャラクタ分の6本の幅W 1 1データ～幅W 1 6データのバーコードデータ値D 3 0をキャラクタのモジュール数に換算して2本加算パターンのモジュール数を求めている。図15Eに示したように、カウンタ部4 0のカウントデータ値から、5種類の幅W 2 1データ～幅W 2 5データがCPU 3 3に順次出力される。

20

【0032】

例えば、CPU 3 3は幅W 2 1データを入力し、キャラクタC 1のバー(1)とスペース(2)とを加算した加算幅W 2 1のモジュール数を求める。モジュール数は全キャラクタ幅 = 1 1モジュールに対する加算幅W 2 1の比率、例えば、2.4を算出する。この場合、加算幅W 2 1のモジュール数は、バーやスペース等の各エレメントの最大誤差が0.5以内となるように、当該モジュール数を四捨五入して最も近い整数になされる。例えば、W 2 1のモジュール数が2.4の場合、W 2 1のモジュール数は「2」である。モジュール数が2.6の場合、W 2 1のモジュール数は「3」である。

20

【0033】

ステップST84でCPU 3 3は1キャラクタ分のバーコードデータ値D 3 0を2本加算パターンに変換する。例えば、図15Eに示したキャラクタC 1のバー(1)とスペース(2)の各々がCPU 3 3によって加算された加算幅W 2 1を示すカウントデータ値が幅W 2 1データとなる。キャラクタC 1のバー(1)に続くスペース(2)及びバー(3)の各々がCPU 3 3によって加算された加算幅W 2 2を示すカウントデータ値が幅W 2 2データとなる。

30

【0034】

キャラクタC 1のスペース(2)に続くバー(3)及びスペース(4)の各々がCPU 3 3によって加算された加算幅W 2 3を示すカウントデータ値が幅W 2 3データとなる。キャラクタC 1のバー(3)に続くスペース(4)及びバー(5)の各々がCPU 3 3によって加算された加算幅W 2 4を示すカウントデータ値が幅W 2 4データとなる。キャラクタC 1のスペース(4)に続くバー(5)及びスペース(6)の各々がCPU 3 3によって加算された加算幅W 2 5を示すカウントデータ値が幅W 2 5データとなる。

40

【0035】

その後、ステップST85でCPU 3 3は、先に変換した2本加算パターンが一致するキャラクタを検索する。この例では、図14に示したカウンタ部4 0から出力されるバーコードデータ値D 3 0に基づいて、CPU 3 3が5種類の幅W 2 1データ～幅W 2 5データ(2本加算データ)に関して、先に変換されたモジュール値と、期待値キャラクタの理論値とが一致するキャラクタを検索する。従来方式では、この段階でキャラクタが決定する。

【0036】

ステップST86でCPU 3 3は一致するキャラクタの存在有無に基づいて制御を分岐す

50

る。一致するキャラクタが存在する場合は、ステップ S T 87で 2 本加算パターンが一致したキャラクタをデコード結果とする。その後、図 17 に示すステップ S T 88に移行する。

【 0 0 3 7 】

上述のステップ S T 87でデコード結果となされたキャラクタについては、ステップ S T 88で、C P U 3 3 はマージン直後のキャラクタか否かに基づいて制御を分岐する。デコード結果となされたキャラクタが、マージン直後のキャラクタではない場合は、ステップ S T 89に移行して C P U 3 3 はデコード結果がストップキャラクタか否かに基づいて制御を分岐する。

【 0 0 3 8 】

デコード結果がストップキャラクタではない場合は、ステップ S T 90に移行して C P U 3 3 は、誤り検知（パリティチェック）を実行して、チェックデジットが正しいか否かに基づいて制御を分岐する。チェックデジットが正しい場合は、ステップ S T 91で C P U 3 3 は読み取りを完了し、デコードバッファ 2 5 のバーコードデータ D 1 2 を出力し、デコードバッファ 2 5 をクリアする（正常終了）。その後、ステップ S T 95に移行する。

【 0 0 3 9 】

なお、ステップ S T 88でマージン直後のキャラクタである場合は、ステップ S T 92に移行して C P U 3 3 は、デコード結果がスタートキャラクタか否かに基づいて制御を分岐する。上述のステップ S T 89でデコード結果がストップキャラクタである場合及び、ステップ S T 92でデコード結果がスタートキャラクタである場合は、図 16 に示すステップ S T 83に移行する。ステップ S T 83で C P U 3 3 は、次キャラクタ（6 本後ろのバーコードデータ値 D 3 0 ）のデコード処理を実行する。その後、ステップ S T 81に戻る。

【 0 0 4 0 】

また、図 16 に示したステップ S T 86で一致するキャラクタが存在しない場合、図 17 に示したステップ S T 92でデコード結果がスタートキャラクタではない場合及び、ステップ S T 90でチェックデジットが誤っている場合は、ステップ S T 94に移行して C P U 3 3 は、読み取りエラー処理を実行する。読み取りエラー処理では次のマージンを検索する等が行われる。

【 0 0 4 1 】

その後、ステップ S T 95に移行して、C P U 3 3 は終了を判別する。例えば、当該光学情報読取装置 2 0 に対して読み取り制御を終了する。読み取り制御が検出され、読み取り終了指令が検出された場合は、バーコード記号 1 の読み取り制御を終了する。読み取り終了指令が検出されていない場合は、バーコード記号 1 の読み取り制御を継続するべく、ステップ S T 81に戻って、上述した内容を繰り返すようになる。

【 先行技術文献 】

【 特許文献 】

【 0 0 4 2 】

【 特許文献 1 】特許 4 5 5 5 9 5 2 号公報（第 4 頁 図 1 ）

【 発明の概要 】

【 発明が解決しようとする課題 】

【 0 0 4 3 】

ところで、従来例に係る光学情報読取装置 2 0 によれば、特許文献 1 に見られるようなバーコード記号読取装置の構成に、1 キャラクタに関して、2 本幅加算方式を導入して難読キャラクタをデコード可能な光学情報読取装置を作成しようとした場合、次のような問題がある。

i . 従来方式の光学情報読取装置 2 0 では、キャラクタパターン算出時、バー及びスペースを含む各エレメントの誤差許容度が ± 0.5 モジュール以内に設定され、1 キャラクタ分の 6 本の幅 W 1 1 データ～幅 W 1 6 データのバーコードデータ値 D 3 0 がキャラクタのモジュール数に換算されて、2 本加算パターンのモジュール数が求められている。しかも、バーやスペース等の各エレメントの最大誤差が 0.5 以内となるように、2 本加算パターンのモジュール数が四捨五入して最も近い整数になされている。

10

20

30

40

50

【0044】

従って、加算幅W21のキャラクタのモジュール数の計数値が2.4の場合、本来、整数の候補として「2」と「3」となるにも関わらず、モジュール数が四捨五入されて最も近い整数として、モジュール数が「2」となされてしまう（ステップS T83参照）。

【0045】

因みに、期待値キャラクタとして、モジュール数が「3」であったような場合であって、印刷技術や貼付技術等によって、バーコード記号を構成するバーやスペースに、欠落や傷等が介在して、モジュール数の計数値が2.4となった場合に、±1.0の誤差が介入する結果となって、バーコード記号を正確に読み取れないという問題がある。誤差が±0.5モジュールを越えるエレメントが存在した場合は、読み取りエラーとなっていた（ステップS T94参照）。

【0046】

ii. また、キャラクタパターン検索時、5種類の幅W21データ～幅W25データ（2本加算データ）に対して一致するキャラクタを検索しているが、誤差が±0.5モジュールを越えるエレメントを含むキャラクタは検索対象から除外されてしまう。従って、ステップS T83で四捨五入によって誤差を含んだキャラクタを含むキャラクタがデコード結果と決定される場合が想定される。この結果、従来方式のバーコード読み取り方法による、バーやスペース等のエレメントが傷やかされた場合に、バーコードを誤読してしまい、バーコードデータを正確に output することが困難となるという問題がある。

【0047】

そこで、本発明は上述した課題を解決したものであって、従来方式のバーコード読み取り方法に比べてバーコードの誤読取を防止できるようにすると共に、エレメントが傷やかされた場合であっても、正確なバーコードデータを出力できるようにした光学情報読み取り装置、光学情報読み取り方法、コンピュータが読み取り可能なプログラム及び記録媒体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】

【0048】

上述の課題を解決するために、請求項1に記載の光学情報読み取り装置は、1キャラクタ内で隣接するバー及びスペースを含むn(n-2)本のエレメントの幅を加算して2～n本加算エレメント幅を各々求め、1キャラクタ幅に対する前記2～n本加算エレメント幅の各々の比率を算出する演算手段と、前記演算手段によって算出された前記比率から、前記2～n本加算エレメント幅の各々のモジュール数が誤差1以内の整数となる対象キャラクタを絞り込んで候補キャラクタを抽出する抽出手段と、前記抽出手段によって抽出された各々の前記2～n本加算エレメント幅のモジュール数の候補キャラクタと、当該候補キャラクタを評価するための期待値キャラクタとを比較する比較手段と、前記候補キャラクタと最も相関が強い期待値キャラクタを検索する検索手段とを備え、前記演算手段は、当該対象キャラクタの端の前記2～n本加算エレメント幅を各々求める際に、前記対象キャラクタの前後のキャラクタのバー及びスペースを含むエレメントの幅を1本以上n-1本以下加算範囲を拡張することを特徴とするものである。

【0049】

請求項1に係る光学情報読み取り装置によれば、バーコード記号を読み取ってそのデコード結果を出力する場合に、演算手段は、1キャラクタ内で隣接するバー及びスペースを含むn(n-2)本のエレメントの幅を加算して2～n本加算エレメント幅を各々求め、この2～n本加算エレメント幅の各々を1キャラクタのモジュール数に換算して当該2～n本加算エレメント幅のモジュール数を求める。抽出手段は、演算手段によって求められた2～n本加算エレメント幅の各々のモジュール数が誤差1以内の整数となるキャラクタを候補キャラクタとして抽出する。これらを前提にして、検索手段が抽出手段によって抽出された各々の2～n本加算エレメント幅のモジュール数の候補キャラクタと、当該候補キャラクタを評価するための期待値キャラクタとを比較し、当該候補キャラクタと最も相関が強い期待値キャラクタを検索するようになる。

10

20

30

40

50

【0050】

この検索によって、複数の候補キャラクタから抽出された候補キャラクタと最も相関が強い期待値キャラクタのバーコードデータを当該バーコード記号の読み取り時のデコード結果として出力できるようになるので、従来方式のバーコード読み取り方法に比べて大幅にバーコードの誤読み取りを防止できるようになる。

【0052】

請求項2に記載の光学情報読み取り装置は、請求項1において、前記抽出手段は、当該対象キャラクタの端の前記2～n本加算エレメント幅を各々求める際に、前記1キャラクタ分のn本の幅データと、当該対象キャラクタの前後の1本以上n-1本以下の幅データとを抽出することを特徴とするものである。

10

【0053】

請求項3に記載の光学情報読み取り装置は、請求項1において、前記検索手段は、前記抽出手段から得られた複数の前記対象キャラクタの各々の前記2～n本加算エレメント幅に対して2種類の前記対象キャラクタの何れかと一致する対象キャラクタを検索し、前記候補キャラクタの数を絞り込むことを特徴とするものである。

【0054】

請求項4に記載の光学情報読み取り装置は、請求項1において、前記検索手段は、前記候補キャラクタのモジュール数の実測値と、前記期待値キャラクタのモジュール数の理想値との間の誤差を算出し、前記誤差を合計して得られる誤差値が最小となる前記候補キャラクタと最も相関が強い期待値キャラクタを検索することを特徴とするものである。

20

【0055】

請求項5に記載の光学情報読み取り装置は、請求項4において、前記検索手段は、前記誤差値が最小となる前記候補キャラクタと一致する前記期待値キャラクタを検出することを特徴とするものである。

【0056】

請求項6に記載の光学情報読み取り装置は、請求項1において、前記1キャラクタ内で隣接するバー及びスペースを含むn(n-2)本のエレメントの幅を加算した合計値と、前記1キャラクタの全幅を示す全幅値とを比較し、比較結果に基づいて当該1キャラクタ分の幅データが適切か否かを判別する比較手段を備えることを特徴とするものである。

30

【0057】

請求項7に記載の光学情報読み取り方法は、光学情報読み取り装置が、1キャラクタ内で隣接するバー及びスペースを含むn(n-2)本のエレメントの幅を加算して2～n本加算エレメント幅を各々求めるステップと、求めた前記2～n本加算エレメント幅の各々の、1キャラクタ幅に対する比率を算出するステップと、算出した前記比率から、前記2～n本加算エレメント幅の各々のモジュール数が誤差1以内の整数となる対象キャラクタを絞り込んで候補キャラクタを抽出するステップと、抽出された各々の前記2～n本加算エレメント幅のモジュール数の候補キャラクタと、当該候補キャラクタを評価するための期待値キャラクタとを比較し、当該候補キャラクタと最も相関が強い期待値キャラクタを検索するステップとを実行し、当該対象キャラクタの端の前記2～n本加算エレメント幅を各々求める際に、前記対象キャラクタの前後のキャラクタのバー及びスペースを含むエレメントの幅を1本以上n-1本以下加算することを特徴とするものである。

40

【0059】

請求項8に記載の光学情報読み取り方法は、請求項7において、前記2～n本加算エレメント幅を各々求める際に、前記1キャラクタ分のn本の幅データと、当該キャラクタの前後の1本以上n-1本以下の幅データとを抽出することを特徴とするものである。

【0060】

請求項9に記載の光学情報読み取り方法は、請求項8において、抽出された複数の前記対象キャラクタの各々の前記2～n本加算エレメント幅に対して2種類の前記対象キャラクタの何れかと一致する対象キャラクタを検索し、前記候補キャラクタの数を絞り込むことを特徴とするものである。

50

【0061】

請求項10に記載の光学情報読取方法は、請求項7において、前記候補キャラクタのモジュール数の実測値と、前記期待値キャラクタのモジュール数の理想値との間の誤差を算出し、前記誤差を合計して得られる誤差値が最小となる前記候補キャラクタと最も相関が強い期待値キャラクタを抽出することを特徴とするものである。

【0062】

請求項11に記載の光学情報読取方法は、請求項10において、前記誤差値が最小となる前記候補キャラクタと一致する前記期待値キャラクタを検出することを特徴とするものである。

【0063】

請求項12に記載の光学情報読取方法は、請求項7において、前記1キャラクタ内で隣接するバー及びスペースを含むn(n-2)本のエレメントの幅を加算した合計値と、前記1キャラクタの全幅を示す全幅値とを比較し、比較結果に基づいて当該1キャラクタ分の幅データが適切か否かを判別することを特徴とするものである。

【0064】

請求項13に記載のコンピュータが読み取り可能なプログラムは、請求項7乃至12に記載の光学情報読取方法を実行するための指令を組み合わせたことを特徴とするものである。

【0065】

請求項14に記載のコンピュータが読み取り可能な記録媒体は、請求項7乃至12に記載の光学情報読取方法を実行するためのプログラムが記述されていることを特徴とするものである。

【発明の効果】

【0066】

請求項1に係る光学情報読取装置及び、請求項7に係る光学情報読取方法によれば、2~n本加算エレメント幅の各々のモジュール数が誤差1以内の整数となる対象キャラクタを絞り込んで候補キャラクタが抽出され、ここに抽出された各々の2~n本加算エレメント幅のモジュール数の候補キャラクタと、当該候補キャラクタを評価するための期待値キャラクタとを比較し、当該候補キャラクタと最も相関が強い期待値キャラクタを検索する検索手段を備えるものである。

【0067】

この構成によって、複数の候補キャラクタから抽出された候補キャラクタと最も相関が強い期待値キャラクタのバーコードデータを当該バーコード記号の読取時のデコード結果として出力できるようになるので、従来方式のバーコード読み取り方法に比べて大幅にバーコードの誤読取を防止できるようになる。

【0068】

更に、当該対象キャラクタの2~n本加算エレメント幅を各々求める際に、対象キャラクタの前後のキャラクタのバー及びスペースを含むエレメントの幅を加算するので、複数の候補キャラクタから抽出された候補キャラクタと最も相関が強い期待値キャラクタのバーコードデータを当該バーコード記号の読取時のデコード結果として出力できるようになる。

【0069】

請求項2に記載の光学情報読取装置及び、請求項8に記載の光学情報読取方法によれば、当該対象キャラクタの前記2~n本加算エレメント幅を各々求める際に、前記1キャラクタ分のn本の幅データと、当該対象キャラクタの前後の1本以上n-1本以下の幅データとを抽出するので、複数の候補キャラクタから抽出された候補キャラクタと最も相関が強い期待値キャラクタのバーコードデータを当該バーコード記号の読取時のデコード結果として出力できるようになる。

【0070】

また、請求項2に係る光学情報読取装置及び請求項8に係る光学情報読取方法によれば

10

20

30

40

50

、1キャラクタ分のn本の幅データと、当該キャラクタの前後の1本以上n-1本以下の幅データとを抽出するので、2~n本加算エレメント幅の各々のモジュール数が誤差1以内の整数となる候補キャラクタを従来方法に比べてより多く抽出できるようになり、第1番目に相関が強いものをデコード結果とするだけでなく、第2番目、第3番目・・・に相関が強いものがデコード結果となる確率が高くなるというようにデコード結果の候補キャラクタを複数出力できるようになる。

【0071】

請求項3に記載の光学情報読取装置及び、請求項9に記載の光学情報読取方法によれば、抽出された複数の前記対象キャラクタの各々の前記2~n本加算エレメント幅に対して2種類の前記対象キャラクタの何れかと一致する対象キャラクタを検索し、候補キャラクタの数を絞り込むので、従来方式のバーコード読み取り方法に比べて大幅にバーコードの誤読取を防止できるようになる。10

【0072】

請求項4に記載の光学情報読取装置及び、請求項10に記載の光学情報読取方法によれば、誤差値が最小となる候補キャラクタと最も相関が強い期待値キャラクタを検索するので、従来方式のバーコード読み取り方法に比べて大幅にバーコードの誤読取を防止できるようになる。

【0073】

請求項5に記載の光学情報読取装置及び、請求項11に記載の光学情報読取方法によれば、誤差値が最小となる候補キャラクタと一致する期待値キャラクタを検出するので、従来方式のバーコード読み取り方法に比べて大幅にバーコードの誤読取を防止できるようになる。20

【0074】

請求項6に記載の光学情報読取装置及び、請求項12に記載の光学情報読取方法によれば、当該1キャラクタ分の幅データが適切か否かの判別結果に基づいてキャラクタのバーの消失や分割等を判断できるようになる。複数のスキャンのデコード結果を組み合わせることが可能となり、汚れや、かすれがひどいバーコードにおいても読み取りが可能となる。

【0075】

請求項13に係るコンピュータが読み取り可能なプログラム及び請求項14に係る記録媒体によれば、本発明に係る光学情報読取方法を実行するための指令を組み合わせたので、複数の候補キャラクタから抽出された候補キャラクタと最も相関が強い期待値キャラクタのバーコードデータを当該バーコード記号の読み取り時のデコード結果として出力できるようになる。従って、従来方式のバーコード読み取り方法に比べて大幅にバーコードの誤読取を防止できるようになる。30

【図面の簡単な説明】

【0076】

【図1】本発明に係る実施形態としてのバーコード記号読取装置100の構成例を示すブロック図である。

【図2】(A)~(G)は第1の実施例に係る1本データ、2~6本加算データの算出例を示す説明図である。40

【図3】第1の実施例に係る難読キャラクタのデコード例(その1)を示すフローチャートである。

【図4】難読キャラクタのデコード例(その2)を示すフローチャートである。

【図5】(A)~(G)は、第1の実施例に係る1本データ、2本~6本加算データの他の算出例(その1)を示す説明図である。

【図6】(A)~(G)は、第1の実施例に係る1本データ、2本~6本加算データの他の算出例(その2)を示す説明図である。

【図7】(A)~(C)は、CODE128のバーコード記号例、その欠損例及びその再構成例を示す説明図である。50

【図8】第2の実施例に係る難読キャラクタのデコード例(その1)を示すフローチャートである。

【図9】難読キャラクタのデコード例(その2)を示すフローチャートである。

【図10】難読キャラクタのデコード例(その3)を示すフローチャートである。

【図11】第3の実施例に係る難読キャラクタのデコード例(その1)を示すフローチャートである。

【図12】難読キャラクタのデコード例(その2)を示すフローチャートである。

【図13】難読キャラクタのデコード例(その3)を示すフローチャートである。

【図14】従来例に係る光学情報読取装置20の構成例を示すプロック図である。

【図15】(A)～(D)は、2本加算データの算出例を示す説明図である。

10

【図16】難読キャラクタのデコード例(その1)を示すフローチャートである。

【図17】難読キャラクタのデコード例(その2)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】

【0077】

以下、図面を参照しながら、本発明に係る光学情報読取装置、光学情報読取方法、コンピュータが読み取り可能なプログラム及び記録媒体について説明する。図1に示すバーコード記号読取装置100は光学情報読取装置の一例を構成し、バーコード記号1を読み取って、そのスタートマージンを検索し、更に、スタートマージンに続くキャラクタをデコード処理してバーコードデータD12を出力するものである。

【0078】

20

バーコード記号読取装置100は、光学読取部2、クロック発生部3、カウンタ部4、m段データバッファ5、×倍器7、セレクタ部6、比較器8、スタートマージン格納部9、幅データ格納部10、インターフェース部11、マイクロプロセッサ12及びデコードバッファ25を有して構成される。なお、従来例と同じ名称及び同じ符号のものは同じ機能を有するのでその説明を省略する。

【0079】

カウンタ部4は光学読取部2から出力されるコード読取データD2を入力すると共に、クロック発生部3から出力されるCLK信号を入力し、当該CLK信号に基づいてコード読取データD2をカウントし、コード読取データD2を白部分のスペースおよび黒部分のバーを含むエレメントの1本幅や、n本加算幅を示すカウントデータD4(バーコードデータ値：数値)に変換する。

30

【0080】

カウンタ部4には幅データ格納部10が接続される。幅データ格納部10は、カウンタ部4から順次出力される1本幅、n本加算幅の対象キャラクタに係るカウントデータD4を、時系列的にすべて格納して保存する。幅データ格納部10は、フリップフロップ回路や、DRAM等のような記憶回路から構成される。幅データ格納部10は、1本幅、n本加算幅の対象キャラクタに係るカウントデータD4を各キャラクタの幅データD10(バーコードデータ値D30に対応)として出力する。スタートマージン格納部9及び幅データ格納部10にはインターフェース部11が接続される。

【0081】

40

マイクロプロセッサ12は、インターフェース部11を介して入力されるSMFデータD9や、各キャラクタの幅データD10によって、バーコードの認識処理を実行する。マイクロプロセッサ12は、プログラム格納用の読み出し専用メモリ(Read Only Memory:以下ROM21という)、ワークメモリ用の隨時情報の書き込み読み出し可能なメモリ(Random Access Memory:以下RAM22という)、及び中央処理ユニット(Central Processing Unit:以下CPU23という)及び期待値格納部24を含むマイクロコンピュータを構成している。

【0082】

ROM21は記録媒体の一例を構成する。ROM21にはコンピュータが読み取り可能なプログラムであって、本発明に係る光学情報読取方法を実行するための指令を組み合わ

50

せたプログラムデータ D p が格納される。その内容は、バーコード記号読取装置 100 が、1 キャラクタ内で隣接するバー及びスペースを含む n (n ≥ 2) 本のエレメントの幅を加算して 2 ~ n 本加算エレメント幅を各々求めるステップと、求めた 2 ~ n 本加算エレメント幅の各々を 1 キャラクタのモジュール数に換算して当該 2 ~ n 本加算エレメント幅のモジュール数を求めるステップと、ここで求めた 2 ~ n 本加算エレメント幅の各々のモジュール数が誤差 1 以内の整数となる対象キャラクタを絞り込んで候補キャラクタを抽出するステップと、抽出された各々の 2 ~ n 本加算エレメント幅のモジュール数の候補キャラクタと、当該候補キャラクタを評価するための期待値キャラクタとを比較し、当該候補キャラクタと最も相関が強い期待値キャラクタを検索するステップとを実行するプログラムである。ここに候補キャラクタとは対象キャラクタを絞り込んだキャラクタをいう。プログラムデータ D p には、システム起動用のシステムプログラムデータも含まれる。10

【0083】

R A M 2 2 にはバーコード記号読取装置 100 を制御するためのシステムプログラムの他に、光学情報読取方法を実行するための演算や、抽出、検索、比較等の制御を実行する際の制御情報が一時格納される。例えば、電源がオンされると、電源オン情報を検出した C P U 2 3 は、R O M 2 1 からシステムプログラムを読み出して R A M 2 2 に展開し、システムを起動して、当該バーコード記号読取装置 100 全体を制御するようになれる。

【0084】

この例で、C P U 2 3 は、少なくとも、演算部 2 3 a や、抽出部 2 3 b 、比較部 2 3 c 、検索部 2 3 d 等の 4 つの機能を有している。演算部 2 3 a は演算手段の一例を構成し、各対象キャラクタの幅データ D 1 0 を入力し、幅データ D 1 0 に基づいて 1 キャラクタ内で隣接するバー及びスペースを含む n (n ≥ 2) 本のエレメントの幅を加算して 2 ~ n 本加算エレメント幅を各々求める。演算部 2 3 a は、求めた 2 ~ n 本加算エレメント幅の各々を 1 キャラクタのモジュール数に換算して当該 2 ~ n 本加算エレメント幅のモジュール数を求める。20

【0085】

抽出部 2 3 b は、演算部 2 3 a によって求められた 2 ~ n 本加算エレメント幅（以下で 2 ~ n 本加算パターンともいう）の各々のモジュール数が誤差 1 以内の整数となる対象キャラクタを絞り込んで候補キャラクタを抽出する。抽出部 2 3 b は、2 ~ n 本加算パターンを各々求める際に、1 キャラクタ分の n 本の幅データ D 1 0 と、対象キャラクタの前後の 1 本以上の幅データ D 1 0 とを抽出する。30

【0086】

この例では、対象キャラクタの抽出基準を従来方式の誤差 0.5 以内の整数から誤差 1 以内の整数へと範囲を拡大し、1 キャラクタ分の n 本の幅データ D 1 0 と、対象キャラクタの前後の 1 本以上の幅データ D 1 0 とを抽出するようにした。従って、2 ~ n 本加算エレメント幅の各々のモジュール数が誤差 1 以内の整数となる対象キャラクタを従来方法に比べて、より多く抽出できるようになる。しかも、第 1 番目に相関が強いものをデコード結果とするだけでなく、第 2 番目、第 3 番目・・・に相関が強いものがデコード結果となる確率が高くなるというようにデコード結果の候補キャラクタを複数出力できるようになる。これにより、従来方式のバーコード読み取り方法に比べて大幅にバーコードの誤読取を防止できるようになる。40

【0087】

比較部 2 3 c は、抽出部 2 3 b によって抽出された各々の 2 ~ n 本加算エレメント幅のモジュール数の候補キャラクタと、当該候補キャラクタを評価するための期待値キャラクタ（理論値）とを比較する。比較部 2 3 c は判別機能を有しており、1 キャラクタ内で隣接するバー及びスペースを含む n (n ≥ 2) 本のエレメントの幅を加算した合計値と、1 キャラクタの全幅を示す全幅値とを比較し、比較結果に基づいて当該 1 キャラクタ分の幅データ D 1 0 が適切か否かを判別する。

【0088】

この判別によって、キャラクタのバーの消失や分割等を判断できるようになる。また、50

複数のスキャンのデコード結果を組み合わせることが可能となり、汚れや、かすれがひどいバーコードにおいても読み取りが可能となる。また、バーの消失や分割等の判断結果に基づいて対象キャラクタを絞ることができ、計算量の軽減が図れ、処理時間の高速化につながる。

【0089】

検索部 23d 検索部 23d は、候補キャラクタと最も相関が強い期待値キャラクタを検索する。例えば、上述の演算部 23a が候補キャラクタのモジュール数の実測値と、期待値キャラクタのモジュール数の理想値との間の誤差を算出すると、検索部 23d が、演算部 23a によって算出された誤差を合計して得られる誤差値が最小となる候補キャラクタと最も相関が強い期待値キャラクタを検索する。この例では、誤差値が最小となる候補キャラクタと最も相関が強い期待値キャラクタを検索するので、従来方式のバーコード読み取り方法に比べて大幅にバーコードの誤読み取りを防止できるようになる。

10

【0090】

マイクロプロセッサ 12 には期待値格納部 24 が接続される。期待値格納部 24 には期待値キャラクタ（理論値）が格納されている。期待値キャラクタは CODE 128 規格のキャラクタの各エレメントを構成するモジュール数（理論値）である。これらにより、バーコード記号読み取り装置 100 を構成する。CPU 23 内の演算部 23a、抽出部 23b、比較部 23c 及び、検索部 23d はソフトウェアでも、ハードウェアでもどちらで構築してもよい。

20

【実施例 1】

【0091】

続いて、図 2A～G を参照して、第 1 の実施例に係る 1 本データ、2～6 本加算データの算出例について説明する。この実施例では、CODE 128 規格のバーコード記号 1 について説明する。CODE 128 規格によれば、図 2A に示す 1 キャラクタは、3 本の黒部分のバー (1), (3), (5) 及び 3 本の白部分のスペース (2), (4), (6) から構成される。図中、C1 は対象キャラクタであり、C0 は対象キャラクタ C1 の 1 つ前のキャラクタであり、C2 はその 1 つ後ろのキャラクタである。

【0092】

1 キャラクタの全幅は 11 モジュールから構成される。この例で、黒部分のバー (1) は 4 モジュールから成り、バー (3) 及びバー (5) は各々 1 モジュールから成る。白部分のスペース (2) は 1 モジュールから成り、スペース (4) 及び (6) は各々 2 モジュールから成る。

30

【0093】

スタートコードは 3 種類あり、それぞれ 103 種類のコードパターンが表される。CODE 128 規格では、アスキー 128 文字を表現できるようになっている。スタートコード C を使用すれば、1 キャラクタで数字 2 衔を表現でき、非常に高い印字密度で表現できる。CODE 128 のバーコード記号 1 は 8 キャラクタから構成される（図 5A 参照）。バーに用いるモジュール数の和は偶数パリティであり、スペースに用いるモジュール数の和は奇数パリティとなっている。偶数パリティ及び奇数パリティを利用した自己機能を持っている。

40

【0094】

図 2B に示す W11 は対象キャラクタ C1 の黒部分のバー (1) の幅である。バー (1) の幅はカウンタ部 4 により計数され、そのカウントデータ値は幅 W11 データを成す。W12 はバー (1) に続くスペース (2) の幅である。スペース (2) の幅もカウンタ部 4 により計数され、そのカウントデータ値は幅 W12 データを成す。W13 は対象キャラクタ C1 のスペース (2) に続くバー (3) の幅である。バー (3) の幅はカウンタ部 4 により計数され、そのカウントデータ値は幅 W13 データを成す。

【0095】

W14 は対象キャラクタ C1 のバー (3) に続くスペース (4) の幅である。スペース (4) の幅もカウンタ部 4 により計数され、そのカウントデータ値は幅 W14 データを成

50

す。W15はスペース(4)に続くバー(5)の幅である。バー(5)の幅はカウンタ部4により計数され、そのカウントデータ値は幅W15データを成す。W16は対象キャラクタC1のバー(5)に続くスペース(6)の幅である。スペース(6)の幅もカウンタ部4により計数され、そのカウントデータ値は幅W16データを成す。上述の6種類の幅W11データ～幅W16データは1キャラクタの1本データであり、幅データD10を構成する。

【0096】

図2Cに示すW21は対象キャラクタC1のバー(1)とスペース(2)の各々が演算部23aによって加算された幅(以下バー(1)+スペース(2)と記す)である。バー(1)+スペース(2)の加算幅を示すカウントデータ値は幅W21データを成す。W22は対象キャラクタC1のバー(1)に続くスペース(2)及びバー(3)の各々が演算部23aによって加算された幅である。スペース(2)+バー(3)の加算幅を示すカウントデータ値は幅W22データを成す。

【0097】

W23は対象キャラクタC1のスペース(2)に続くバー(3)及びスペース(4)の各々が演算部23aによって加算された幅である。バー(3)+スペース(4)の加算幅を示すカウントデータ値は幅W23データを成す。W24は対象キャラクタC1のバー(3)に続くスペース(4)及びバー(5)の各々が演算部23aによって加算された幅である。スペース(4)+バー(5)の加算幅を示すカウントデータ値は幅W24データを成す。W25は対象キャラクタC1のスペース(4)に続くバー(5)及びスペース(6)の各々が演算部23aによって加算された幅である。バー(5)+スペース(6)の加算幅を示すカウントデータ値は幅W25データを成す。上述の5種類の幅W21データ～幅W25データは1キャラクタの2本加算データであり、幅データD10を構成する。

【0098】

図2Dに示すW31は対象キャラクタC1のバー(1)、スペース(2)及びバー(3)の各々が演算部23aによって加算された幅である。バー(1)+スペース(2)+バー(3)の加算幅を示すカウントデータ値は幅W31データを成す。W32は対象キャラクタC1のバー(1)に続くスペース(2)、バー(3)及びスペース(4)の各々が演算部23aによって加算された幅である。スペース(2)+バー(3)+スペース(4)の加算幅を示すカウントデータ値は幅W32データを成す。

【0099】

W33は対象キャラクタC1のスペース(2)に続くバー(3)、スペース(4)及びバー(5)の各々が演算部23aによって加算された幅である。バー(3)+スペース(4)+バー(5)の加算幅を示すカウントデータ値は幅W33データを成す。W34は対象キャラクタC1のバー(3)に続くスペース(4)、バー(5)及びスペース(6)の各々が演算部23aによって加算された幅である。スペース(4)+バー(5)+スペース(6)の加算幅を示すカウントデータ値は幅W34データを成す。上述の4種類の幅W31データ～幅W34データは1キャラクタの3本加算データであり、幅データD10を構成する。

【0100】

図2Eに示すW41は対象キャラクタC1のバー(1)、スペース(2)、バー(3)及びスペース(4)の各々が演算部23aによって加算された幅である。バー(1)+スペース(2)+バー(3)+スペース(4)の加算幅を示すカウントデータ値は幅W41データを成す。W42の対象キャラクタC1はバー(1)に続くスペース(2)、バー(3)、スペース(4)及びバー(5)の各々が演算部23aによって加算された幅である。スペース(2)+バー(3)+スペース(4)+バー(5)の加算幅を示すカウントデータ値は幅W42データを成す。

【0101】

W43は対象キャラクタC1のスペース(2)に続くバー(3)、スペース(4)、バー(5)及びスペース(6)の各々が演算部23aによって加算された幅である。バー(

10

20

30

40

50

3) + スペース (4) + バー (5) + スペース (6) の加算幅を示すカウントデータ値は幅 W 4 3 データを成す。上述の 3 種類の幅 W 4 1 データ ~ 幅 W 4 3 データは 1 キャラクタの 4 本加算データであり、幅データ D 1 0 を構成する。

【 0 1 0 2 】

図 2 F に示す W 5 1 は対象キャラクタ C 1 のバー (1) 、スペース (2) 、バー (3) 、スペース (4) 及びバー (5) の各々が演算部 2 3 a によって加算された幅である。バー (1) + スペース (2) + バー (3) + スペース (4) + バー (5) の加算幅を示すカウントデータ値は幅 W 5 1 データを成す。

【 0 1 0 3 】

W 5 2 は対象キャラクタ C 1 のバー (1) に続くスペース (2) 、バー (3) 、スペース (4) 、バー (5) 及びスペース (6) の各々が演算部 2 3 a によって加算された幅である。スペース (2) + バー (3) + スペース (4) + バー (5) + スペース (6) の加算幅を示すカウントデータ値は幅 W 5 2 データを成す。上述の 2 種類の幅 W 5 1 データ及び幅 W 5 2 データは 1 キャラクタの 5 本加算データであり、幅データ D 1 0 を構成する。

【 0 1 0 4 】

図 2 G に示す W 6 1 は対象キャラクタ C 1 のバー (1) 、スペース (2) 、バー (3) 、スペース (4) 、バー (5) 及びスペース (6) の各々が演算部 2 3 a によって加算された幅である。バー (1) + スペース (2) + バー (3) + スペース (4) + バー (5) + スペース (6) の加算幅を示すカウントデータ値は幅 W 6 1 データを成す。1 種類の幅 W 6 1 データは 1 キャラクタの 6 本加算データであり、幅データ D 1 0 を構成する。

【 0 1 0 5 】

上述のカウンタ部 4 から出力されるカウントデータ値から、6 種類の幅 W 1 1 データ ~ 幅 W 1 6 データ (1 本データ) 、5 種類の幅 W 2 1 データ ~ 幅 W 2 5 データ (2 本加算データ) 、4 種類の幅 W 3 1 データ ~ 幅 W 3 4 データ (3 本加算データ) 、3 種類の幅 W 4 1 データ ~ 幅 W 4 3 データ (4 本加算データ) 、2 種類の幅 W 5 1 データ及び幅 W 5 2 データ (5 本加算データ) 及び、1 種類の幅 W 6 1 データ (6 本加算データ) の合計で、21 種類の 1 本データ、2 ~ 6 本加算データ が得られる。

【 0 1 0 6 】

続いて、本発明に係る光学情報読取方法について、図 3 及び図 4 を参照し、第 1 の実施例に係る難読キャラクタのデコード例を説明する。この実施例では、C O D E 1 2 8 規格に基づくバーコード記号 1 を読み取って、そのスタートマージンを検索し、更に、スタートマージンに続く対象キャラクタをデコード処理してバーコードデータ D 1 2 を出力する場合を例に挙げる。その場合のバーコード記号 1 のバー及びスペースの各エレメントの誤差許容度 (最大誤差範囲) が、± 1 . 0 モジュールに設定される。バーコード記号 1 については図 2 に示した対象キャラクタ C 1 について説明する (キャラクタデコードアルゴリズム) 。

【 0 1 0 7 】

これらをデコード条件にして、まず、図 3 に示すフローチャートのステップ S T 1 でマイクロプロセッサ 1 2 はスタートマージンを検索する。このとき、光学読取部 2 は、バーコード記号 1 を読み取って、コード読取データ D 2 を発生する。コード読取データ D 2 は、光学読取部 2 からカウンタ部 4 へ出力される。クロック発生部 3 は、所定周波数のサンプリング用の C L K 信号をカウンタ部 4 に出力する。

【 0 1 0 8 】

カウンタ部 4 は光学読取部 2 から出力されるコード読取データ D 2 を入力すると共に、クロック発生部 3 から出力される C L K 信号を入力し、C L K 信号に基づいてコード読取データ D 2 をカウントし、コード読取データ D 2 を白部分のスペースおよび黒部分のバーを含むエレメントの 1 本幅及び 6 本加算幅を示すカウントデータ D 4 に変換する。カウントデータ D 4 は、バーコード記号 1 における各バー (黒部分) の幅およびスペース (白部分) の幅を示す数値データ列となる。カウントデータ D 4 は、幅データ格納部 1 0 に保存される。

10

20

30

40

50

【0109】

m段データバッファ5はカウンタ部4から出力されるカウントデータD4を、m段分(mは任意の整数)だけ一時格納する。m段は、バーコード記号1の印刷状態により任意に設定されている。×倍器7は、カウントデータD4からスタートマージンを検索するに当たり、カウントデータD4を何倍かを示す「×」が予め設定され、その×倍の演算を行って、×倍データD7を出力する。

【0110】

セレクタ部6は、マイクロプロセッサ12の選択制御を受けて、m段データバッファ5に一時格納されたm段分のカウントデータD4のうち、比較対象とするデータを選択する。比較器8はセレクタ部6が選択した、m段データバッファ5中のk段前までの各カウントデータD4と、×倍器7から出力される×倍データD7とを比較する。

10

【0111】

比較結果によって×倍器7からの×倍データD7の方が小さいときは、スタートマージン候補があったと判定し、スタートマージン候補検索フラグ(以下でSMFデータD9という)を発生する。SMFデータD9はスタートマージン格納部9に出力される。スタートマージン格納部9には比較器8から出力されるSMFデータD9、そのとき比較対象としたカウントデータ値=D10とが対応させて格納される。

【0112】

ステップST2でCPU23は1キャラクタ分の幅データD10(6本)と、その前後の5本ずつ幅データD10を抽出する。図2Bに示した対象キャラクタC1によれば、黒部分のバー(1)の幅W11がカウンタ部4により計数され、そのカウントデータ値が幅W11データとなる。バー(1)に続く白部分のスペース(2)の幅W12もカウンタ部4により計数され、そのカウントデータ値が幅W12データとなる。スペース(2)に続くバー(3)の幅W13はカウンタ部4により計数され、そのカウントデータ値が幅W13データとなる。

20

【0113】

このバー(3)に続くスペース(4)の幅W14もカウンタ部4により計数され、そのカウントデータ値は幅W14データとなる。スペース(4)に続くバー(5)の幅W15はカウンタ部4により計数され、そのカウントデータ値が幅W15データとなる。バー(5)に続くスペース(6)の幅W16もカウンタ部4により計数され、そのカウントデータ値が幅W16データとなる。上述の6種類の幅W11データ~幅W16データは幅データD10を構成する。幅データD10はインターフェース部11を介してCPU23に出力される。

30

【0114】

CPU23では、演算部23aが各対象キャラクタの幅データD10を入力し、幅データD10に基づいて1キャラクタ内で隣接するバー及びスペースを含むn(n-2)本のエレメントの幅を加算して2~n本加算パターンを各々求める。この例ではn=6である。2~n本加算エレメント幅を各々求める際に、1キャラクタ分の6本の幅データD10と、対象キャラクタの前後の5本の幅データD10とを抽出する。

40

【0115】

例えば、図2Cに示した対象キャラクタC1のバー(1)とスペース(2)の各々が演算部23aによって加算され、その加算幅W21を示すカウントデータ値が幅W21データとなる。対象キャラクタC1のバー(1)に続くスペース(2)及びバー(3)の各々が演算部23aによって加算され、その加算幅W22を示すカウントデータ値が幅W22データとなる。対象キャラクタC1のスペース(2)に続くバー(3)及びスペース(4)の各々が演算部23aによって加算され、その加算幅W23を示すカウントデータ値が幅W23データとなる。

【0116】

対象キャラクタC1のバー(3)に続くスペース(4)及びバー(5)の各々が演算部23aによって加算され、その加算幅W24を示すカウントデータ値が幅W24データと

50

なる。対象キャラクタC1のスペース(4)に続くバー(5)及びスペース(6)の各々が演算部23aによって加算され、その加算幅W25を示すカウントデータ値が幅W25データとなる。上述の5種類の幅W21データ～幅W25データは幅データD10を構成する。

【0117】

図2Dに示した対象キャラクタC1のバー(1)、スペース(2)及びバー(3)の各々が演算部23aによって加算され、その加算幅W31を示すカウントデータ値が幅W31データとなる。対象キャラクタC1のバー(1)に続くスペース(2)、バー(3)及びスペース(4)の各々が演算部23aによって加算され、その加算幅W32を示すカウントデータ値が幅W32データとなる。

10

【0118】

対象キャラクタC1のスペース(2)に続くバー(3)、スペース(4)及びバー(5)の各々が演算部23aによって加算され、その加算幅W33を示すカウントデータ値が幅W33データとなる。対象キャラクタC1のバー(3)に続くスペース(4)、バー(5)及びスペース(6)の各々が演算部23aによって加算され、その加算幅W34を示すカウントデータ値が幅W34データとなる。上述の4種類の幅W31データ～幅W34データは幅データD10を構成する。

【0119】

図2Eに示した対象キャラクタC1のバー(1)、スペース(2)、バー(3)及びスペース(4)の各々が演算部23aによって加算され、その加算幅W41を示すカウントデータ値が幅W41データとなる。対象キャラクタC1はバー(1)に続くスペース(2)、バー(3)、スペース(4)及びバー(5)の各々が演算部23aによって加算され、その加算幅W42を示すカウントデータ値が幅W42データとなる。

20

【0120】

対象キャラクタC1のスペース(2)に続くバー(3)、スペース(4)、バー(5)及びスペース(6)の各々が演算部23aによって加算され、その加算幅W43を示すカウントデータ値が幅W43データとなる。上述の3種類の幅W41データ～幅W43データは幅データD10を構成する。

【0121】

図2Fに示した対象キャラクタC1のバー(1)、スペース(2)、バー(3)、スペース(4)及びバー(5)の各々が演算部23aによって加算され、その加算幅W51を示すカウントデータ値が幅W51データとなる。対象キャラクタC1のバー(1)に続くスペース(2)、バー(3)、スペース(4)、バー(5)及びスペース(6)の各々が演算部23aによって加算され、その加算幅W52を示すカウントデータ値が幅W52データを成す。上述の2種類の幅W51データ及び幅W52データは幅データD10を構成する。

30

【0122】

図2Gに示した対象キャラクタC1のバー(1)、スペース(2)、バー(3)、スペース(4)、バー(5)及びスペース(6)の各々が演算部23aによって加算され、その加算幅W61を示すカウントデータ値が幅W61データとなる。1種類の幅W61データは幅データD10を構成する。

40

【0123】

上述のカウンタ部4から出力されるカウントデータ値から、6種類の幅W11データ～幅W16データ(1本データ)、5種類の幅W21データ～幅W25データ(2本加算データ)、4種類の幅W31データ～幅W34データ(3本加算データ)、3種類の幅W41データ～幅W43データ(4本加算データ)、2種類の幅W51データ及び幅W52データ(5本加算データ)及び、1種類の幅W61データ(6本加算データ)の合計で、21種類の1本データ、2～6本加算データが得られる。

【0124】

ステップST3でCPU23は対象キャラクタC1の1モジュール幅を算出する。この

50

例では、演算部 2 3 a は、先に求めた 2 ~ 6 本加算パターン（エレメント幅）の各々を 1 キャラクタのモジュール数に換算して 2 ~ 6 本加算パターンのモジュール数を求める。図 2 B ~ G に示したように、カウンタ部 4 から出力されるカウントデータ値から、6 種類の幅 W 1 1 データ ~ 幅 W 1 6 データ、5 種類の幅 W 2 1 データ ~ 幅 W 2 5 データ、4 種類の幅 W 3 1 データ ~ 幅 W 3 4 データ、3 種類の幅 W 4 1 データ ~ 幅 W 4 3 データ、2 種類の幅 W 5 1 データ及び幅 W 5 2 データ及び、1 種類の幅 W 6 1 データが C P U 2 3 に順次出力される。

【 0 1 2 5 】

例えれば、演算部 2 3 a は幅 W 2 1 データを入力し、対象キャラクタ C 1 のバー（1）とスペース（2）とを加算した加算幅 W 2 1 のモジュール数を求める。モジュール数は全キャラクタ幅 = 1 1 モジュールに対する加算幅 W 2 1 の比率、例えば、2 . 4 を算出する。この場合、加算幅 W 2 1 のモジュール数は、バーやスペース等の各エレメントの最大誤差が 1 . 0 以内の整数となる対象キャラクタを絞り込んで候補キャラクタを抽出するので、加算幅 W 2 1 のモジュール数は「2」又は「3」となる。

【 0 1 2 6 】

この例では、各エレメントの最大誤差範囲を $\pm 1 . 0$ モジュール以内に拡張したので、各エレメントの誤差許容度が 1 . 0 以内まで許容量を増加できるようになる。許容量を増加したので、対象キャラクタ一致を探る条件を増加できる。これにより、バーコード記号 1 の誤読みを防止できるようになる。

【 0 1 2 7 】

ステップ S T 4 で C P U 2 3 は抽出された 2 1 種類の幅データ D 1 0 を 1 ~ 6 本加算パターン（エレメント幅）に変換する。この例では、6 種類の幅 W 1 1 データ ~ 幅 W 1 6 データが 1 キャラクタの 1 本データに変換され、1 本データに基づく対象キャラクタ C 1 の被認識パターンとして、1 本パターンが作成される。5 種類の幅 W 2 1 データ ~ 幅 W 2 5 データは 1 キャラクタの 2 本加算データに変換され、2 本加算データに基づく対象キャラクタ C 1 の 2 本加算パターン（被認識パターン）が作成される。4 種類の幅 W 3 1 データ ~ 幅 W 3 4 データは 1 キャラクタの 3 本加算データに変換され、3 本加算データに基づく対象キャラクタ C 1 の 3 本加算識パターンが作成される。

3 種類の幅 W 4 1 データ ~ 幅 W 4 3 データは 1 キャラクタの 4 本加算データに変換され、4 本加算データに基づく対象キャラクタ C 1 の 4 本加算パターンが作成される。2 種類の幅 W 5 1 データ及び幅 W 5 2 データは 1 キャラクタの 5 本加算データに変換され、5 本加算データに基づく対象キャラクタ C 1 の 5 本加算パターンが作成される。1 種類の幅 W 6 1 データは 1 キャラクタの 6 本加算データに変換され、6 本加算データに基づく対象キャラクタ C 1 の 6 本加算パターンが作成される。

【 0 1 2 8 】

ステップ S T 5 で C P U 2 3 は対象キャラクタを絞り込むために、1 本パターン、2 ~ 6 本加算パターンの全てに対してモジュール数の誤差が 1 以内の対象キャラクタを検索して候補キャラクタを決定する。このとき、C P U 2 3 では抽出部 2 3 b が、演算部 2 3 a によって求められた 1 本パターン、2 ~ 6 本加算パターンの各々のモジュール数が誤差 1 以内の整数となる対象キャラクタを絞り込んで候補キャラクタを抽出する。その際に、抽出部 2 3 b は、演算部 2 3 a によって求められた 1 本パターン、2 ~ 6 本加算パターンの 1 キャラクタ分の 6 本の幅データ D 1 0 と、対象キャラクタの前後の (6 - 1) 本の幅データ D 1 0 とを抽出する。

【 0 1 2 9 】

ステップ S T 6 で C P U 2 3 は、候補キャラクタが 1 つ以上存在するか否かに基づいて制御を分岐する。対象キャラクタが 1 つ以上（複数）存在する場合は、ステップ S T 7 で 1 ~ 6 本加算パターンの実測値とその理想値との間の誤差を算出し、この誤差の合計が最小となった候補キャラクタをデコード結果とする。この誤差値が最小となる候補キャラクタは、期待値キャラクタと最も相関が強いキャラクタである。この例では、総合的な誤差が最も少ない対象キャラクタを候補キャラクタとして選択できるようになる。

10

20

30

40

50

【0130】

このとき、C P U 2 3 では、期待値キャラクタ（理論値）が期待値格納部24からC P U 2 3 の比較部23cへ読み出される。比較部23cは、抽出部23bによって抽出された各々の2～6本加算パターン（エレメント幅）のモジュール数の候補キャラクタと、候補キャラクタを評価するための期待値キャラクタ（理論値）とを比較する。この比較結果は検索部23dへ出力される。

【0131】

検索部23dでは、候補キャラクタと最も相関が強い期待値キャラクタが検索される。例えば、上述の演算部23aが候補キャラクタのモジュール数の実測値と、期待値キャラクタのモジュール数の理想値との間の誤差を算出すると、検索部23dが、演算部23aによって算出された誤差を合計して得られる誤差値が最小となる候補キャラクタと最も相関が強い期待値キャラクタを検索する。この例では、誤差値が最小となる候補キャラクタと最も相関が強い期待値キャラクタを検索するので、従来方式のバーコード読み取り方法に比べて大幅にバーコードの誤読み取りを防止できるようになる。

10

【0132】

以後は従来方式と同じ内容となる。例えば、図4に示すステップS T 8でC P U 2 3 は、デコード結果の候補キャラクタがマージン直後の対象キャラクタか否かに基づいて制御を分岐する。デコード結果の候補キャラクタがマージン直後の対象キャラクタである場合は、ステップS T 9に移行してC P U 2 3 はデコード結果の候補キャラクタがストップキャラクタか否かに基づいて制御を分岐する。

20

【0133】

デコード結果の候補キャラクタがストップキャラクタではない場合は、ステップS T 10に移行してC P U 2 3 は誤り検知（パリティチェック）を実行し、チェックデジットが正しいか否かに基づいて制御を分岐する。チェックデジットが正しい場合は、読み取りを完了し、ステップS T 11でC P U 2 3 はデコード結果となされた候補キャラクタのバーコードデータD 1 2 を出力する。

【0134】

なお、ステップS T 8でC P U 2 3 はデコード結果の候補キャラクタがマージン直後の対象キャラクタでない場合は、ステップS T 12に移行して、C P U 2 3 はデコード結果の候補キャラクタがスタートキャラクタか否かに基づいて制御を分岐する。

30

【0135】

上述のステップS T 9でC P U 2 3 はデコード結果の候補キャラクタがストップキャラクタである場合及び、ステップS T 12でデコード結果の候補キャラクタがスタートキャラクタである場合は、図3に示すステップS T 13に移行する。ステップS T 13で、C P U 2 3 は次キャラクタ（6本後ろのデータ）のデコード処理を実行する。その後、ステップS T 2 に戻る。

【0136】

また、ステップS T 12でデコード結果の候補キャラクタがスタートキャラクタでない場合及び、ステップS T 10でチェックデジットが誤っている場合は、ステップS T 14に移行して、C P U 2 3 は読み取りエラー処理を実行する。読み取りエラー処理では次のマージンを検索する等が行われる。キャラクタ認識がエラーとなった場合に、エラーキャラクタを仮に配置してバーコードデータを作成するようにしてもよい。

40

【0137】

その後、ステップS T 15に移行して、C P U 2 3 は終了を判別する。例えば、当該バーコード記号読み取り装置100に対して読み取り終了指令が検出され、読み取り終了指令が検出された場合は、バーコード記号1の読み取り制御を終了する。読み取り終了指令が検出されていない場合は、バーコード記号1の読み取り制御を継続するべく、ステップS T 1 に戻って、上述した内容を繰り返すようになされる。

【0138】

このように第1の実施例に係るバーコード記号読み取り装置100によれば、バーコード記

50

号1を読み取ってそのデコード結果を出力する場合に、演算部23aは、1キャラクタ内で隣接するバー及びスペースを含む $n = 6$ ($n \geq 2$) 本のエレメントの幅を加算して2~6本加算パターン(エレメント幅)を各々求める。演算部23aは、2~6本加算パターンの各々を1キャラクタのモジュール数に換算して当該2~6本加算パターンのモジュール数を求める。

【0139】

抽出部23bは、演算部23aによって求められた2~6本加算パターンの各々のモジュール数が誤差1以内の整数となる対象キャラクタを絞り込んで候補キャラクタを抽出する。これらを前提にして、検索部23dが抽出部23bによって抽出された各々の2~6本加算パターンのモジュール数の候補キャラクタと、当該候補キャラクタを評価するための期待値キャラクタとを比較し、当該候補キャラクタと最も相関が強い期待値キャラクタを検索するようになる。

10

【0140】

この検索によって、複数の対象キャラクタから抽出された候補キャラクタと最も相関が強い期待値キャラクタのバーコードデータD12を当該バーコード記号1の読み取り時のデコード結果として出力できるようになるので、従来方式のバーコード読み取り方法に比べて大幅にバーコード記号1の誤読み取りを防止できるようになる。

【0141】

しかも、誤差の範囲を従来方式の ± 0.5 から ± 1.0 に拡大し、候補キャラクタを増加させている。そして、1キャラクタ分の6本の幅データD10と、当該キャラクタC1の前後の5本の幅データD10とを抽出するので、2~6本加算パターンの各々のモジュール数が誤差1以内の整数となる対象キャラクタを絞り込んだ候補キャラクタを従来方法に比べてより多く抽出できるようになる。

20

【0142】

また、第1番目に相関が強い候補キャラクタをデコード結果とするだけでなく、第2番目、第3番目・・・に相関が強い候補キャラクタがデコード結果となる確率が高くなるというようにデコード結果の候補キャラクタを複数出力できるようになる。これにより、従来方式のバーコード読み取り方法に比べて大幅にバーコードの誤読み取りを防止できるようになる。これにより、従来方式のバーコード読み取り方法に比べて大幅にバーコードの誤読み取りを防止できるようになる。なお、誤差の範囲を設定する際の閾値を可変することで、幅データD10の信頼性を選択することも可能となる。

30

【0143】

続いて、図5A~Gを参照して、第1の実施例に係る1本データ、2本加算データの他の算出例(その1)について説明する。この例では、対象キャラクタC1の両側のキャラクタC0, C2まで加算範囲を拡張したものである。対象キャラクタC1と隣接する1つ前のキャラクタC0のスペース(6)及びバー(5)の2エレメントまで加算範囲を拡張し、1つ後ろのキャラクタC2のバー(1)及びスペース(2)の2エレメントまで加算範囲を拡張する場合を前提とする。

【0144】

図5Aに示す対象キャラクタC1の1つ前にはキャラクタC0が設けられ、対象キャラクタC1の1つ後ろにはキャラクタC2が設けられる。各々のキャラクタC0~C2は、その各々が全幅11モジュールから構成される。3個のバー(1), (3), (5)及び3個のスペース(2), (4), (6)から構成される。バー(1)のモジュール数は「4」であり、バー(3), (5)のモジュール数は「1」である。また、スペース(2)のモジュール数は「1」であり、スペース(4), (6)のモジュール数は「2」である。これらのキャラクタC0~C2を有したバーコードの場合を例に挙げて、対象キャラクタの作成する場合を説明する。

40

【0145】

図5Bに示す対象キャラクタC1のW11は、黒部分のバー(1)の幅である。バー(1)の幅はカウンタ部4により計数され、そのカウントデータ値が幅W11データを成す

50

。W12は対象キャラクタC1のバー(1)に続くスペース(2)の幅である。スペース(2)の幅もカウンタ部4により計数され、そのカウントデータ値が幅W12データを成す。W13は対象キャラクタC1のスペース(2)に続くバー(3)の幅である。バー(3)の幅はカウンタ部4により計数され、そのカウントデータ値が幅W13データを成す。

【0146】

W14は対象キャラクタC1のバー(3)に続くスペース(4)の幅である。スペース(4)の幅もカウンタ部4により計数され、そのカウントデータ値が幅W14データを成す。W15は対象キャラクタC1のスペース(4)に続くバー(5)の幅である。バー(5)の幅はカウンタ部4により計数され、そのカウントデータ値が幅W15データを成す。W16は対象キャラクタC1のバー(5)に続くスペース(6)の幅である。スペース(6)の幅もカウンタ部4により計数され、そのカウントデータ値が幅W16データを成す。上述の6種類の幅W11データ～幅W16データは1キャラクタの1本データであり、幅データD10を構成する。」

図5Cに示すW2-1は、1つ前の対象キャラクタC0のスペース(6)と対象キャラクタC1のバー(1)の各々が演算部23aによって加算された幅である。スペース(6)+バー(1)の加算幅を示すカウントデータ値が幅W2-1データを成す。W21は1つ前のキャラクタC0のスペース(6)に続く対象キャラクタC1のバー(1)とスペース(2)の各々が演算部23aによって加算された幅(以下バー(1)+スペース(2)と記す)である。バー(1)+スペース(2)の加算幅を示すカウントデータ値が幅W21データを成す。W22は対象キャラクタC1のバー(1)に続くスペース(2)及びバー(3)の各々が演算部23aによって加算された幅である。スペース(2)+バー(3)の加算幅を示すカウントデータ値が幅W22データを成す。

【0147】

W23は対象キャラクタC1のスペース(2)に続くバー(3)及びスペース(4)の各々が演算部23aによって加算された幅である。バー(3)+スペース(4)の加算幅を示すカウントデータ値が幅W23データを成す。W24は対象キャラクタC1のバー(3)に続くスペース(4)及びバー(5)の各々が演算部23aによって加算された幅である。スペース(4)+バー(5)の加算幅を示すカウントデータ値が幅W24データを成す。

【0148】

W25は対象キャラクタC1のスペース(4)に続くバー(5)及びスペース(6)の各々が演算部23aによって加算された幅である。バー(5)+スペース(6)の加算幅を示すカウントデータ値が幅W25データを成す。W26はバー(5)に続くスペース(6)及び1つ後ろのキャラクタC2のバー(1)の各々が演算部23aによって加算された幅である。スペース(6)+バー(1)の加算幅を示すカウントデータ値が幅W26データを成す。上述の7種類の幅W2-1データ～幅W26データは1キャラクタの2本加算データであり、幅データD10を構成する。

【0149】

図5Dに示すW3-2は、1つ前のキャラクタC0のバー(5)、スペース(6)及び、対象キャラクタC1のバー(1)の各々が演算部23aによって加算された幅である。バー(5)+スペース(6)+バー(1)の加算幅を示すカウントデータ値が幅W3-2データを成す。W3-1は、1つ前のキャラクタC0のスペース(6)、対象キャラクタC1のバー(1)及び、スペース(2)の各々が演算部23aによって加算された幅である。スペース(6)+バー(1)+スペース(2)の加算幅を示すカウントデータ値が幅W3-1データを成す。

【0150】

W31は対象キャラクタC1のバー(1)、スペース(2)及びバー(3)の各々が演算部23aによって加算された幅である。バー(1)+スペース(2)+バー(3)の加算幅を示すカウントデータ値が幅W31データを成す。W32は対象キャラクタC1のバ

10

20

30

40

50

-(1)に続くスペース(2)、バー(3)及びスペース(4)の各々が演算部23aによって加算された幅である。スペース(2)+バー(3)+スペース(4)の加算幅を示すカウントデータ値が幅W32データを成す。

【0151】

W33は対象キャラクタC1のスペース(2)に続くバー(3)、スペース(4)及びバー(5)の各々が演算部23aによって加算された幅である。バー(3)+スペース(4)+バー(5)の加算幅を示すカウントデータ値が幅W33データを成す。W34は対象キャラクタC1のバー(3)に続くスペース(4)、バー(5)及びスペース(6)の各々が演算部23aによって加算された幅である。スペース(4)+バー(5)+スペース(6)の加算幅を示すカウントデータ値が幅W34データを成す。

10

【0152】

W35は、対象キャラクタC1のスペース(4)に続くバー(5)、スペース(6)、1つ後ろのキャラクタC2のバー(1)の各々が演算部23aによって加算された幅である。バー(5)+スペース(6)+バー(1)の加算幅を示すカウントデータ値が幅W35データを成す。W36は、対象キャラクタC1のバー(3)に続くスペース(6)の1つ後ろのキャラクタC2のバー(1)及びスペース(2)の各々が演算部23aによって加算された幅である。スペース(6)+バー(1)+スペース(2)の加算幅を示すカウントデータ値が幅W36データを成す。上述の8種類の幅W3-2データ～幅W36データは1キャラクタの3本加算データであり、幅データD10を構成する。

【0153】

20

図5Eに示すW4-2は、1つ前のキャラクタC0のバー(5)、スペース(6)、対象キャラクタC1のバー(1)及びスペース(2)の各々が演算部23aによって加算された幅である。バー(5)+スペース(6)+バー(1)+スペース(2)の加算幅を示すカウントデータ値が幅W4-2データを成す。W4-1は、1つ前のキャラクタC0のスペース(6)、対象キャラクタC1のバー(1)、スペース(2)及びバー(3)の各々が演算部23aによって加算された幅である。スペース(6)+バー(1)+スペース(2)+バー(3)の加算幅を示すカウントデータ値が幅W4-1データを成す。

【0154】

W41は対象キャラクタC1のバー(1)、スペース(2)、バー(3)及びスペース(4)の各々が演算部23aによって加算された幅である。バー(1)+スペース(2)+バー(3)+スペース(4)の加算幅を示すカウントデータ値が幅W41データを成す。W42は対象キャラクタC1のバー(1)に続くスペース(2)、バー(3)、スペース(4)及びバー(5)の各々が演算部23aによって加算された幅である。スペース(2)+バー(3)+スペース(4)+バー(5)の加算幅を示すカウントデータ値が幅W42データを成す。

30

【0155】

W43は対象キャラクタC1のスペース(2)に続くバー(3)、スペース(4)、バー(5)及びスペース(6)の各々が演算部23aによって加算された幅である。バー(3)+スペース(4)+バー(5)+スペース(6)の加算幅を示すカウントデータ値が幅W43データを成す。W44は、対象キャラクタC1のバー(3)に続くスペース(4)、バー(5)、スペース(6)、1つ後ろのキャラクタC2のバー(1)の各々が演算部23aによって加算された幅である。スペース(4)+バー(5)+スペース(6)+バー(1)の加算幅を示すカウントデータ値が幅W44データを成す。

40

【0156】

W45は、対象キャラクタC1の白のスペース(4)に続くバー(5)、スペース(6)、1つ後ろのキャラクタC2のバー(1)及び、スペース(2)の幅である。バー(5)+スペース(6)+バー(1)+スペース(2)の加算幅を示すカウントデータ値が幅W45データを成す。上述の7種類の幅W4-2データ～幅W45データは、1キャラクタの4本加算データであり、幅データD10を構成する。

【0157】

50

図 5 F に示す W5-2 は、1 つ前のキャラクタ C 0 のバー (5)、スペース (6)、対象キャラクタ C 1 のバー (1)、スペース (2) 及びバー (3) の各々が演算部 23a によって加算された幅である。バー (5) + スペース (6) + バー (1) + スペース (2) + バー (3) の加算幅を示すカウントデータ値が幅 W5-2 データを成す。W5-1 は、1 つ前のキャラクタ C 0 のスペース (6) + キャラクタ C 1 のバー (1) + スペース (2) + バー (3) + スペース (4) の各々が演算部 23a によって加算された幅である。スペース (6) + バー (1) + スペース (2) + バー (3) + スペース (4) の加算幅を示すカウントデータ値が幅 W5-1 データを成す。

【0158】

W5-1 は対象キャラクタ C 1 のバー (1)、スペース (2)、バー (3)、スペース (4) 及びバー (5) の各々が演算部 23a によって加算された幅である。バー (1) + スペース (2) + バー (3) + スペース (4) + バー (5) の加算幅を示すカウントデータ値が幅 W5-1 データを成す。W5-2 は対象キャラクタ C 1 のバー (1) に続くスペース (2)、バー (3)、スペース (4)、バー (5) 及びスペース (6) の各々が演算部 23a によって加算された幅である。スペース (2) + バー (3) + スペース (4) + バー (5) + スペース (6) の加算幅を示すカウントデータ値が幅 W5-2 データを成す。

【0159】

W5-3 は、対象キャラクタ C 1 のスペース (2) に続くバー (3)、スペース (4)、バー (5)、スペース (6) 及び 1 つ後ろのキャラクタ C 2 のバー (1) の各々が演算部 23a によって加算された幅である。バー (3) + スペース (4) + バー (5) + スペース (6) + バー (1) の加算幅を示すカウントデータ値が幅 W5-3 データを成す。W5-4 は、対象キャラクタ C 1 のバー (3) に続くスペース (4)、バー (5)、スペース (6)、1 つ後ろのキャラクタ C 2 のバー (1) 及びスペース (2) の各々が演算部 23a によって加算された幅である。スペース (4) + バー (5) + スペース (6) + バー (1) + スペース (2) の加算幅を示すカウントデータ値が幅 W5-4 データを成す。上述の 6 種類の幅 W5-2 データ及び幅 W5-4 データは 1 キャラクタの 5 本加算データであり、幅データ D10 を構成する。

【0160】

図 5 G に示す W6-2 は、1 つ前のキャラクタ C 0 のバー (5)、スペース (6)、対象キャラクタ C 1 のバー (1)、スペース (2)、バー (3) 及びスペース (4) の各々が演算部 23a によって加算された幅である。バー (5) + スペース (6) + キャラクタ C 1 のバー (1) + スペース (2) + バー (3) + スペース (4) の加算幅を示すカウントデータ値が幅 W6-2 データを成す。

【0161】

W6-1 は、1 つ前のキャラクタ C 0 のスペース (6)、対象キャラクタ C 1 のバー (1)、スペース (2)、バー (3)、スペース (4) 及びバー (5) の各々が演算部 23a によって加算された幅である。スペース (6) + バー (1) + スペース (2) + バー (3) + スペース (4) + バー (5) の加算幅を示すカウントデータ値が幅 W6-1 データを成す。

【0162】

W6-1 は対象キャラクタ C 1 のバー (1)、スペース (2)、バー (3)、スペース (4)、バー (5) 及びスペース (6) の各々が演算部 23a によって加算された幅である。バー (1) + スペース (2) + バー (3) + スペース (4) + バー (5) + スペース (6) の加算幅を示すカウントデータ値が幅 W6-1 データを成す。

【0163】

W6-2 は対象キャラクタ C 1 のバー (1) に続くスペース (2)、バー (3)、スペース (4)、バー (5)、スペース (6) 及び 1 つ後ろのキャラクタ C 2 のバー (1) の各々が演算部 23a によって加算された幅である。スペース (2) + バー (3) + スペース (4) + バー (5) + スペース (6) + バー (1) の加算幅を示すカウントデータ値が幅 W6-2 データを成す。

10

20

30

40

50

【0164】

W63は対象キャラクタC1のスペース(2)に続くバー(3)、スペース(4)、バー(5)、スペース(6)、1つ後ろのキャラクタC2のバー(1)及びスペース(2)の各々が演算部23aによって加算された幅である。バー(3)+スペース(4)+バー(5)+スペース(6)+バー(1)+スペース(2)の加算幅を示すカウントデータ値が幅W63データを成す。上述の5種類の幅W6-2データ及び幅W63データは1キャラクタの6本加算データであり、幅データD10を構成する。

【0165】

上述のカウンタ部4から出力されるカウントデータ値に基づいて、6種類の幅W11データ～幅W16データ(1本データ)、7種類の幅W2-1データ～幅W26データ(2本加算データ)、8種類の幅W3-2データ～幅W36データ(3本加算データ)、7種類の幅W4-2データ～幅W45データ(4本加算データ)、6種類の幅W5-2データ及び幅W54データ(5本加算データ)及び、5種類の幅W6-2データ及び幅W63データ(6本加算データ)の合計で、39種類の1本データ、2～6本加算データが得られる。

10

【0166】

続いて、図6A～Gを参照して、第1の実施例に係る1本データ、2本～6本加算データの他の算出例(その2)について説明する。この例では、対象キャラクタC1の両側のキャラクタC0、C2まで加算範囲を拡張する場合であって、図6Aに示す対象キャラクタC1と隣接する1つ前のキャラクタC0のスペース(6)、バー(5)、スペース(4)、バー(3)及びスペース(2)の5エレメントまで加算範囲を拡張し、かつ、1つ後ろのキャラクタC2のバー(1)、スペース(2)、バー(3)、スペース(4)及びバー(5)の5エレメントまで加算範囲を拡張する場合を前提とする。これらのキャラクタC0～C2を有したバーコード記号1の場合を例に挙げて、対象キャラクタを作成する場合を説明する。

20

【0167】

図6Bに示すW11は対象キャラクタC1の黒部分のバー(1)の幅である。バー(1)の幅はカウンタ部4により計数され、そのカウントデータ値が幅W11データを成す。W12は対象キャラクタC1のバー(1)に続くスペース(2)の幅である。スペース(2)の幅もカウンタ部4により計数され、そのカウントデータ値が幅W12データを成す。W13はスペース(2)に続くバー(3)の幅である。バー(3)の幅はカウンタ部4により計数され、そのカウントデータ値が幅W13データを成す。

30

【0168】

W14は対象キャラクタC1のバー(3)に続くスペース(4)の幅である。スペース(4)の幅もカウンタ部4により計数され、そのカウントデータ値が幅W14データを成す。W15は対象キャラクタC1のスペース(4)に続くバー(5)の幅である。バー(5)の幅はカウンタ部4により計数され、そのカウントデータ値が幅W15データを成す。W16は対象キャラクタC1のバー(5)に続くスペース(6)の幅である。スペース(6)の幅もカウンタ部4により計数され、そのカウントデータ値が幅W16データを成す。上述の6種類の幅W11データ～幅W16データは1キャラクタの1本データであり、幅データD10を構成する。

40

【0169】

図6Cに示すW2-1は、1つ前のキャラクタC0のスペース(6)と対象キャラクタC1のバー(1)の各々が演算部23aによって加算された幅である。スペース(6)+バー(1)の加算幅を示すカウントデータ値が幅W2-1データを成す。W21は1つ前のキャラクタC0のスペース(6)に続くバー(1)とスペース(2)の各々が演算部23aによって加算された幅(以下バー(1)+スペース(2)と記す)である。バー(1)+スペース(2)の加算幅を示すカウントデータ値が幅W21データを成す。W22は対象キャラクタC1のバー(1)に続くスペース(2)及びバー(3)の各々が演算部23aによって加算された幅である。スペース(2)+バー(3)の加算幅を示すカウントデータ値が幅W22データを成す。

50

【0170】

W23は対象キャラクタC1のスペース(2)に続くバー(3)及びスペース(4)の各々が演算部23aによって加算された幅である。バー(3)+スペース(4)の加算幅を示すカウントデータ値が幅W23データを成す。W24は対象キャラクタC1のバー(3)に続くスペース(4)及びバー(5)の各々が演算部23aによって加算された幅である。スペース(4)+バー(5)の加算幅を示すカウントデータ値が幅W24データを成す。W25は対象キャラクタC1のスペース(4)に続くバー(5)及びスペース(6)の各々が演算部23aによって加算された幅である。バー(5)+スペース(6)の加算幅を示すカウントデータ値が幅W25データを成す。W26は対象キャラクタC1のバー(5)に続くスペース(6)及び1つ後ろのキャラクタC2のバー(1)の各々が演算部23aによって加算された幅である。スペース(6)+バー(1)の加算幅を示すカウントデータ値が幅W26データを成す。上述の7種類の幅W2-1データ～幅W26データは1キャラクタの2本加算データであり、幅データD10を構成する。10

【0171】

図6Dに示すW3-2は、1つ前のキャラクタC0のバー(5)、スペース(6)及び、対象キャラクタC1のバー(1)の各々が演算部23aによって加算された幅である。バー(5)+スペース(6)+バー(1)の加算幅を示すカウントデータ値が幅W3-2データを成す。W3-1は、1つ前のキャラクタC0のスペース(6)対象キャラクタC1のバー(1)及び、スペース(2)の各々が演算部23aによって加算された幅である。スペース(6)+バー(1)+スペース(2)の加算幅を示すカウントデータ値が幅W3-1データを成す。20

【0172】

W31は対象キャラクタC1のバー(1)、スペース(2)及びバー(3)の各々が演算部23aによって加算された幅である。バー(1)+スペース(2)+バー(3)の加算幅を示すカウントデータ値が幅W31データを成す。W32は対象キャラクタC1のバー(1)に続くスペース(2)、バー(3)及びスペース(4)の各々が演算部23aによって加算された幅である。スペース(2)+バー(3)+スペース(4)の加算幅を示すカウントデータ値が幅W32データを成す。

【0173】

W33は対象キャラクタC1のスペース(2)に続くバー(3)、スペース(4)及びバー(5)の各々が演算部23aによって加算された幅である。バー(3)+スペース(4)+バー(5)の加算幅を示すカウントデータ値が幅W33データを成す。W34はバー(3)に続くスペース(4)、バー(5)及びスペース(6)の各々が演算部23aによって加算された幅である。スペース(4)+バー(5)+スペース(6)の加算幅を示すカウントデータ値が幅W34データを成す。30

【0174】

W35は、対象キャラクタC1のスペース(4)に続くバー(5)、スペース(6)、1つ後ろのキャラクタC2のバー(1)の各々が演算部23aによって加算された幅である。バー(5)+スペース(6)+バー(1)の加算幅を示すカウントデータ値が幅W35データを成す。W36は、対象キャラクタC1のバー(3)に続くスペース(6)の1つ後ろのキャラクタC2のバー(1)及びスペース(2)の各々が演算部23aによって加算された幅である。スペース(6)+バー(1)+スペース(2)の加算幅を示すカウントデータ値が幅W36データを成す。上述の8種類の幅W3-2データ～幅W36データは1キャラクタの3本加算データであり、幅データD10を構成する。40

【0175】

図6Eに示すW4-3は、1つ前のキャラクタC0のスペース(4)、バー(5)、スペース(6)及び、バー(1)の各々が演算部23aによって加算された幅である。スペース(4)+バー(5)+スペース(6)+バー(1)の加算幅を示すカウントデータ値が幅W4-3データを成す。W4-2は、1つ前のキャラクタC0のバー(5)、スペース(6)、対象キャラクタC1のバー(1)及びスペース(2)の各々が演算部23aによって加50

算された幅である。バー(5) + スペース(6) + バー(1) + スペース(2)の加算幅を示すカウントデータ値が幅W4-2データを成す。

【0176】

W4-1は、1つ前のキャラクタC0のスペース(6)、対象キャラクタC1のバー(1)、スペース(2)及びバー(3)の各々が演算部23aによって加算された幅である。スペース(6) + バー(1) + スペース(2) + バー(3)の加算幅を示すカウントデータ値が幅W4-1データを成す。W4-1は対象キャラクタC1のバー(1)、スペース(2)、バー(3)及びスペース(4)の各々が演算部23aによって加算された幅である。バー(1) + スペース(2) + バー(3) + スペース(4)の加算幅を示すカウントデータ値が幅W4-1データを成す。W4-2は対象キャラクタC1のバー(1)に続くスペース(2)、バー(3)、スペース(4)及びバー(5)の各々が演算部23aによって加算された幅である。スペース(2) + バー(3) + スペース(4) + バー(5)の加算幅を示すカウントデータ値が幅W4-2データを成す。

【0177】

W4-3は対象キャラクタC1のスペース(2)に続くバー(3)、スペース(4)、バー(5)及びスペース(6)の各々が演算部23aによって加算された幅である。バー(3) + スペース(4) + バー(5) + スペース(6)の加算幅を示すカウントデータ値が幅W4-3データを成す。W4-4は、対象キャラクタC1のバー(3)に続くスペース(4)、バー(5)、スペース(6)、1つ後ろのキャラクタC2のバー(1)の各々が演算部23aによって加算された幅である。スペース(4) + バー(5) + スペース(6) + バー(1)の加算幅を示すカウントデータ値が幅W4-4データを成す。

【0178】

W4-5は、対象キャラクタC1の白のスペース(4)に続くバー(5)、スペース(6)、1つ後ろのキャラクタC2のバー(1)及び、スペース(2)の各々が演算部23aによって加算された幅である。バー(5) + スペース(6) + バー(1) + スペース(2)の加算幅を示すカウントデータ値が幅W4-5データを成す。W4-6は、対象キャラクタC1のバー(5)に続くスペース(6)、1つ後ろのキャラクタC2のバー(1)、スペース(2)及びバー(3)の各々が演算部23aによって加算された幅である。スペース(6) + バー(1) + スペース(2) + バー(3)の加算幅を示すカウントデータ値が幅W4-6データを成す。上述の9種類の幅W4-3データ～幅W4-6データは1キャラクタの4本加算データであり、幅データD10を構成する。

【0179】

図6Fに示すW5-4は、1つ前のキャラクタC0のバー(3)、スペース(4)、バー(5)、スペース(6) + キャラクタC1のバー(1)の各々が演算部23aによって加算された幅である。バー(3) + スペース(4) + バー(5) + スペース(6) + バー(1)の加算幅を示すカウントデータ値が幅W5-4データを成す。W5-3は、1つ前のキャラクタC0のスペース(4)、バー(5)、スペース(6)、対象キャラクタC1のバー(1)及びスペース(2)の各々が演算部23aによって加算された幅である。スペース(4) + バー(5) + スペース(6) + バー(1) + スペース(2)の加算幅を示すカウントデータ値が幅W5-3データを成す。

【0180】

W5-2は、1つ前のキャラクタC0のバー(5)、スペース(6)、対象キャラクタC1のバー(1)、スペース(2)及びバー(3)の各々が演算部23aによって加算された幅である。バー(5) + スペース(6) + バー(1) + スペース(2) + バー(3)の加算幅を示すカウントデータ値が幅W5-2データを成す。W5-1は、1つ前のキャラクタC0のスペース(6)、対象キャラクタC1のバー(1)、スペース(2)、バー(3)及びスペース(4)の各々が演算部23aによって加算された幅である。スペース(6) + バー(1) + スペース(2) + バー(3) + スペース(4)の加算幅を示すカウントデータ値が幅W5-1データを成す。

【0181】

10

20

30

40

50

W51は対象キャラクタC1のバー(1)、スペース(2)、バー(3)、スペース(4)及びバー(5)の各々が演算部23aによって加算された幅である。バー(1)+スペース(2)+バー(3)+スペース(4)+バー(5)の加算幅を示すカウントデータ値が幅W51データを成す。W52は対象キャラクタC1のバー(1)に続くスペース(2)、バー(3)、スペース(4)、バー(5)及びスペース(6)の各々が演算部23aによって加算された幅である。スペース(2)+バー(3)+スペース(4)+バー(5)+スペース(6)の加算幅を示すカウントデータ値が幅W52データを成す。

【0182】

W53は、対象キャラクタC1のスペース(2)に続くバー(3)、スペース(4)、バー(5)、スペース(6)及び1つ後ろのキャラクタC2のバー(1)の各々が演算部23aによって加算された幅である。バー(3)+スペース(4)+バー(5)+スペース(6)+バー(1)の加算幅を示すカウントデータ値が幅W53データを成す。W54は、対象キャラクタC1のバー(3)に続くスペース(4)、バー(5)、スペース(6)、1つ後ろのキャラクタC2のバー(1)及びスペース(2)の各々が演算部23aによって加算された幅である。スペース(4)+バー(5)+スペース(6)+バー(1)+スペース(2)の加算幅を示すカウントデータ値が幅W54データを成す。

【0183】

W55は、対象キャラクタC1のスペース(4)に続くバー(5)、スペース(6)、1つ後ろのキャラクタC2のバー(1)、スペース(2)及び、バー(3)の各々が演算部23aによって加算された幅である。バー(5)+スペース(6)+バー(1)+スペース(2)+バー(3)の加算幅を示すカウントデータ値が幅W55データを成す。W56は、対象キャラクタC1のバー(5)に続くスペース(6)、1つ後ろのキャラクタC2のバー(1)、スペース(2)、バー(3)及びスペース(4)の各々が演算部23aによって加算された幅である。スペース(6)+バー(1)+スペース(2)+バー(3)+スペース(4)の加算幅を示すカウントデータ値が幅W56データを成す。上述の10種類の幅W5-4データ及び幅W56データは1キャラクタの5本加算データであり、幅データD10を構成する。

【0184】

図6Gに示すW6-5は、1つ前のキャラクタC0のスペース(2)、バー(3)、スペース(4)、バー(5)、スペース(6)及び、対象キャラクタC1のバー(1)の各々が演算部23aによって加算された幅である。スペース(2)+バー(3)+スペース(4)+バー(5)+スペース(6)+バー(1)の加算幅を示すカウントデータ値が幅W6-5データを成す。W6-4は、1つ前のキャラクタC0のバー(3)、スペース(4)、バー(5)、スペース(6)、対象キャラクタC1のバー(1)及びスペース(2)の各々が演算部23aによって加算された幅である。バー(3)+スペース(4)+バー(5)+スペース(6)+バー(1)+スペース(2)の加算幅を示すカウントデータ値が幅W6-4データを成す。

【0185】

W6-3は、1つ前のキャラクタC0のスペース(4)、バー(5)、スペース(6)、対象キャラクタC1のバー(1)、スペース(2)及びバー(3)の各々が演算部23aによって加算された幅である。スペース(4)+バー(5)+スペース(6)+バー(1)+スペース(2)+バー(3)の加算幅を示すカウントデータ値が幅W6-3データを成す。W6-2は、1つ前のキャラクタC0のバー(5)+スペース(6)+キャラクタC1のバー(1)+スペース(2)+バー(3)+スペース(4)の各々が演算部23aによって加算された幅である。バー(5)+スペース(6)+キャラクタC1のバー(1)+スペース(2)+バー(3)+スペース(4)の加算幅を示すカウントデータ値が幅W6-2データを成す。

【0186】

W6-1は、1つ前のキャラクタC0のスペース(6)+キャラクタC1のバー(1)+スペース(2)+バー(3)+スペース(4)+バー(5)の各々が演算部23aによつ

10

20

30

40

50

て加算された幅である。スペース(6) + キャラクタC1のバー(1) + スペース(2) + バー(3) + スペース(4) + バー(5)の加算幅を示すカウントデータ値が幅W6-1データを成す。

【0187】

W61は対象キャラクタC1のバー(1)、スペース(2)、バー(3)、スペース(4)、バー(5)及びスペース(6)の各々が演算部23aによって加算された幅である。バー(1) + スペース(2) + バー(3) + スペース(4) + バー(5) + スペース(6)の加算幅を示すカウントデータ値が幅W61データを成す。

【0188】

W62は対象キャラクタC1のバー(1)に続くスペース(2)、バー(3)、スペース(4)、バー(5)、スペース(6)及び、1つ後ろのキャラクタC2のバー(1)の各々が演算部23aによって加算された幅である。スペース(2) + バー(3) + スペース(4) + バー(5) + スペース(6) + バー(1)の加算幅を示すカウントデータ値が幅W62データを成す。

10

W63は対象キャラクタC1のスペース(2)に続くバー(3)、スペース(4)、バー(5)、スペース(6)、1つ後ろのキャラクタC2のバー(1)及びスペース(2)の各々が演算部23aによって加算された幅である。バー(3) + スペース(4) + バー(5) + スペース(6) + バー(1) + スペース(2)の加算幅を示すカウントデータ値が幅W63データを成す。

20

【0189】

W64は、対象キャラクタC1のバー(3)に続くスペース(4)、バー(5)、スペース(6)、1つ後ろのキャラクタC2のバー(1)、スペース(2)及びバー(3)の各々が演算部23aによって加算された幅である。スペース(4) + バー(5) + スペース(6) + 1つ後ろのキャラクタC2のバー(1) + スペース(2) + バー(3)の加算幅を示すカウントデータ値が幅W64データを成す。

【0190】

W65は、対象キャラクタC1のスペース(4)に続くバー(5)、スペース(6)、1つ後ろのキャラクタC2のバー(1)、スペース(2)、バー(3)及びスペース(4)の各々が演算部23aによって加算された幅である。バー(5) + スペース(6) + 1つ後ろのキャラクタC2のバー(1) + スペース(2) + バー(3) + スペース(4)の加算幅を示すカウントデータ値が幅W65データを成す。

30

【0191】

W66は、対象キャラクタC1のバー(5)に続くスペース(6)、1つ後ろのキャラクタC2のバー(1)、スペース(2)、バー(3)、スペース(4)及びバー(5)の各々が演算部23aによって加算された幅である。スペース(6) + 1つ後ろのキャラクタC2のバー(1) + スペース(2) + バー(3) + スペース(4) + バー(5)の加算幅を示すカウントデータ値が幅W66データを成す。上述の11種類の幅W6-5データ及び幅W66データは1キャラクタの6本加算データであり、幅データD10を構成する。

【0192】

40

この例では、図1に示したカウンタ部4から出力されるカウントデータ値に基づいて、6種類の幅W11データ～幅W16データ(1本データ)、7種類の幅W2-1データ～幅W26データ(2本加算データ)、8種類の幅W3-2データ～幅W36データ(3本加算データ)、9種類の幅W4-3データ～幅W46データ(4本加算データ)、10種類の幅W5-4データ及び幅W56データ(5本加算データ)及び、11種類の幅W6-5データ及び幅W66データ(6本加算データ)の合計で、51種類の1本データ、2～6本加算データが得られる。

【実施例2】

【0194】

続いて、図7～図10を参照して、第2の実施例に係るデコード例について説明する。

50

ここで、図 7 A～C を参照して、CODE128 のバーコード記号例、その欠損例及びその再構成例について説明をする。図 7 A に示す CODE128 のバーコード記号 1 は、バーやスペース等の欠落や傷等を生じていないものである。バーコード記号 1 は 8 キャラクタから構成され、例えば、バーコードデータ D12 は、数字の「12345678」を表現する。

【0195】

図 7 B に示す CODE128 のバーコード記号 1' は、バーやスペース等に欠落や傷等が生じているものである。バーコード記号 1' は図 7 A に示した CODE128 のバーコード記号 1 に生じたものと仮定する。図中の 3 本の横向き破線は、閾値 $th_1 \sim th_3$ である。各々の閾値 $th_1 \sim th_3$ は、スキャン方向と直交する上・中・下段に設定されている。 10

【0196】

閾値 th_1 を基準にしてバーコード記号 1' をスキャンして、データを読み込むと、傷等によりキャラクタ「3」及び「4」が欠落している。閾値 th_2 を基準にしてバーコード記号 1' をスキャンしてデータを読み込むと、傷等によりキャラクタ「3」～「6」が欠落している。閾値 th_3 を基準にしてバーコード記号 1' をスキャンしてデータを読み込むと、傷等によりキャラクタ「7」及び「8」が欠落している。

【0197】

図 7 C に示すように閾値 th_1 を基準にしたデコード結果は、「12??5678」を示すバーコードデータ D12 が outputされる。「3」及び「4」がデコードされていない。閾値 th_2 を基準にしたデコード結果は、「12????78」を示すバーコードデータ D12 が outputされる。「3」～「6」がデコードされていない。閾値 th_3 を基準にしたデコード結果は、「123456??」を示すバーコードデータ D12 が outputされる。「7」及び「8」がデコードされていない。 20

【0198】

第 2 の実施例では、図 7 B に示したようにバーやスペース等に欠落や傷等が生じている場合であっても、「12??5678」や、「12????78」、「123456??」等を「12345678」を表現するバーコードデータ D12 に再構成するようになされる。

【0199】

この例では、スキャン方向と直交する上・中・下段に次々とスキャンを行って、エラーキャラクタを補正し、エラーキャラクタが無くなった時点で CD (チェックデジット) の照合を行って、バーコードデータ D12 を作成するようにした。これにより、かすれ、汚れ等により、エレメントの一部が欠落しているバーコード記号 1 であってもその読み取りが可能となる。 30

【0200】

また、第 1 の実施例では 1 キャラクタ内の幅データ D10 のみを利用した場合について説明したが、1 キャラクタ内の端部の幅データ D10 が利用される回数が少ない。例えば、対象キャラクタ C1 のバー (1) の幅 W11 データは 2 本加算パターンの算出時、加算幅 W21 でのみ利用されるが、スペース (2) の幅 W12 データは、2 本加算パターンの算出時、加算幅 W21 と加算幅 W22 とで利用されている。そこで、第 2 の実施例では当該キャラクタ C1 に隣接するキャラクタ C0, C2 も含めたエレメント部分も含めて計算を行うようにした。 40

【0201】

続いて、図 8～図 10 を参照して、第 2 の実施例に係る難読キャラクタのデコード例 (その 1～3) について説明する。この実施例では、パターン認識する加算範囲を両側の 2 キャラクタのエレメントまで拡張した。加算範囲は、図 7 A に示した対象キャラクタと隣接する 1 つ前のキャラクタ C0 のスペース (6)、バー (5)、スペース (4)、バー (3) 及びスペース (2) の 5 エレメントまで拡張し、かつ、1 つ後ろのキャラクタ C2 のバー (1)、スペース (2)、バー (3)、スペース (4) 及びバー (5) の 5 エレメントまで拡張する場合を前提とする。この加算範囲の拡張によって、51 種類の 1 本データ 50

、2～6本加算データからデコード結果を得る場合を前提とする（マルチスキャンリコントラクション）。

【0202】

これらをデコード条件にして、図8に示すフローチャートのステップS T21でCPU23はスタートマージンを検索する。このとき、光学読取部2は、第1の実施例と同様にして、バーコード記号1を読み取って、コード読取データD2を発生する。コード読取データD2は、光学読取部2からカウンタ部4へ出力される。クロック発生部3は、所定周波数のサンプリング用のCLK信号をカウンタ部4に出力する。

【0203】

カウンタ部4は光学読取部2から出力されるコード読取データD2を入力すると共に、クロック発生部3から出力されるCLK信号を入力し、CLK信号に基づいてコード読取データD2をカウントし、コード読取データD2を白部分のスペースおよび黒部分のバーを含むエレメントの1本幅及び2本加算幅を示すカウントデータD4に変換する。カウントデータD4は、バーコード記号1における各バー（黒部分）の幅およびスペース（白部分）の幅を示す数値データ列となる。カウントデータD4は、幅データ格納部10に保存される。なお、スタートマージンの検索については、その説明を省略する。

【0204】

ステップS T22でCPU23は、1キャラクタ分の幅データ（6本）を抽出する。図15に示したキャラクタC1によれば、黒部分のバー（1）の幅W11がカウンタ部4により計数され、そのカウントデータ値が幅W11データとなる。バー（1）に続く白部分のスペース（2）の幅W12もカウンタ部4により計数され、そのカウントデータ値が幅W12データとなる。スペース（2）に続くバー（3）の幅W13はカウンタ部4により計数され、そのカウントデータ値が幅W13データとなる。

【0205】

このバー（3）に続くスペース（4）の幅W14もカウンタ部4により計数され、そのカウントデータ値は幅W14データとなる。スペース（4）に続くバー（5）の幅W15はカウンタ部4により計数され、そのカウントデータ値が幅W15データとなる。バー（5）に続くスペース（6）の幅W16もカウンタ部4により計数され、そのカウントデータ値が幅W16データとなる。上述の6種類の幅W11データ～幅W16データはバーコードデータ値D30を構成する。バーコードデータ値D30はインターフェース部11を介してCPU23に出力される。

【0206】

ステップS T23でCPU23はバーの分割、消失の有無に基づいて制御を分岐する。バーの分割、消失の有無については、6本の幅の合計と1キャラクタ幅とを比較し、1キャラクタ分の幅データD10が適切かを判断する。ここでは、1キャラクタ内で隣接するバー及びスペースを含む6本のエレメントの幅を加算した合計値と、1キャラクタの全幅を示す全幅値とを比較し、比較結果に基づいて1キャラクタ分の幅データD10が適切か否かを判別する。

【0207】

この例で、CPU23では比較部23cが1キャラクタ内で隣接する6本のバー（1）、スペース（2）、バー（3）、スペース（4）、バー（5）及びスペース（6）の幅W1～W6の合計と、1キャラクタの全幅を示す全幅値とを比較する。比較部23cによる比較結果によって、1キャラクタ分の幅データD10が適切か否かが判別でき、この判別に基づいてキャラクタのバーの消失や分割等を判断できるようになる。また、複数のスキャンのデコード結果を組み合わせることが可能となり、汚れや、かすれがひどいバーコードにおいても読み取りが可能となる。

【0208】

また、当該キャラクタC1の前後のキャラクタC0、C2の全幅からキャラクタフレームを判定できるので、ボイドやスポット等によるエレメントに過不足が有っても、デコードを継続できるようになる。また、バーの消失や分割等の判断結果に基づいて対象キャラ

10

20

30

40

50

クタを絞ることができ、計算量の軽減が図れ、処理時間の高速化につながる。

【0209】

バーの分割、消失が無い場合は、ステップS T24でC P U 2 3は1モジュール幅を算出し、1キャラクタ幅を記憶する。このとき、演算部2 3 aは、6種類の幅W 1 1データ～幅W 1 6データに基づいて当該キャラクタC 1の1モジュール幅を算出する。1キャラクタ幅は、図7Bに示した1キャラクタ内で隣接する6本のバー(1)、スペース(2)、バー(3)、スペース(4)、バー(5)及びスペース(6)の幅W 1～W 6を合計して求められる。

【0210】

ステップS T25でC P U 2 3は1キャラクタ分の幅データD 1 0を2本加算パターンに変換する。例えば、図7Cに示した1つ前のキャラクタC 0のスペース(6)と対象キャラクタC 1のバー(1)の各々が演算部2 3 aによって加算された加算幅W2-1を示すカウントデータ値が幅W2-1データとなる。

10

【0211】

同様にして、1つ前のキャラクタC 0のスペース(6)に続くバー(1)とスペース(2)の各々が演算部2 3 aによって加算された加算幅W 2 1を示すカウントデータ値が幅W 2 1データとなる。対象キャラクタC 1のバー(1)に続くスペース(2)及びバー(3)の各々が演算部2 3 aによって加算された加算幅W 2 2を示すカウントデータ値が幅W 2 2データとなる。

【0212】

20

対象キャラクタC 1のスペース(2)に続くバー(3)及びスペース(4)の各々が演算部2 3 aによって加算された加算幅W 2 3を示すカウントデータ値が幅W 2 3データとなる。対象キャラクタC 1のバー(3)に続くスペース(4)及びバー(5)の各々が演算部2 3 aによって加算された加算幅W 2 4を示すカウントデータ値が幅W 2 4データとなる。

【0213】

対象キャラクタC 1のスペース(4)に続くバー(5)及びスペース(6)の各々が演算部2 3 aによって加算された加算幅W 2 5を示すカウントデータ値が幅W 2 5データとなる。対象キャラクタC 1のバー(3)に続くスペース(4)及び1つ後ろのキャラクタC 2のバー(1)の各々が演算部2 3 aによって加算された加算幅W 2 6を示すカウントデータ値が幅W 2 6データとなる。上述の7種類の幅W2-1データ～幅W 2 6データは1キャラクタの2本加算データであり、幅データD 1 0から構成される。

30

【0214】

その後、ステップS T26でC P U 2 3は、先に変換した2本加算パターンが一致するキャラクタを検索する。この例では、図1に示したカウンタ部4から出力されるカウントデータ値に基づいて、検索部2 3 dが、6種類の幅W 1 1データ～幅W 1 6データ(1本データ)、7種類の幅W2-1データ～幅W 2 6データ(2本加算データ)、8種類の幅W3-2データ～幅W 3 6データ(3本加算データ)、9種類の幅W4-3データ～幅W 4 6データ(4本加算データ)、10種類の幅W5-4データ及び幅W 5 6データ(5本加算データ)及び、11種類の幅W6-5データ及び幅W 6 6データ(6本加算データ)の51種類の1本データ、2～6本加算データC 1に対して一致するキャラクタを検索する。その際に各エレメントの幅に対して2種類の対象キャラクタのどちらかと一致する対象キャラクタを検索し、対象キャラクタを絞り込む。この絞り込みにより、およそ1～4程度の候補キャラクタが検索される。

40

【0215】

ステップS T27でC P U 2 3は一致するキャラクタの存在有無に基づいて制御を分岐する。一致するキャラクタが存在する場合は、ステップS T28で2本加算パターンが一致した対象キャラクタをデコード結果とする。その後、図9に示すステップS T31に移行する。

【0216】

50

なお、ステップ S T 23でバーの分割、消失が有る場合、及び、ステップ S T 27で一致するキャラクタが存在しない場合は、ステップ S T 29で C P U 2 3 は現在のキャラクタをエラーキャラクタとする。その後、ステップ S T 30で C P U 2 3 は次のキャラクタ(1キャラクタ幅分後ろのデータ)のデコード処理に移行する。その後、ステップ S T 22に移行して、上述した内容を繰り返す。

【0217】

上述のステップ S T 28でデコード結果となされた候補キャラクタについては、図9に示すステップ S T 31で C P U 2 3 はマージン直後のキャラクタか否かに基づいて制御を分岐する。デコード結果となされた候補キャラクタが、マージン直後のキャラクタではない場合は、ステップ S T 32に移行して C P U 2 3 はデコード結果がストップキャラクタか否かに基づいて制御を分岐する。デコード結果がストップキャラクタである場合は、ステップ S T 33に移行して C P U 2 3 は、デコード結果にエラーキャラクタが存在しないか否かに基づいて制御を分岐する。

10

【0218】

デコード結果にエラーキャラクタが存在しない場合は、ステップ S T 34に移行して、 C P U 2 3 はデコードバッファ25に現在(今)のスキャンのデコード結果を上書きする。その後、ステップ S T 35に移行して C P U 2 3 は誤り検知(パリティチェック)を実行して、チェックデジットが正しいか否かに基づいて制御を分岐する。チェックデジットが正しい場合は、ステップ S T 36で C P U 2 3 は読み取りを完了し、デコードバッファ25のバーコードデータD12を出力し、デコードバッファ25をクリアする(正常終了)。その後、ステップ S T 45に移行する。

20

【0219】

なお、ステップ S T 31でマージン直後のキャラクタである場合は、ステップ S T 37に移行して C P U 2 3 は、デコード結果がスタートキャラクタか否かに基づいて制御を分岐する。上述のステップ S T 37でデコード結果がスタートキャラクタでない場合及び、ステップ S T 32でデコード結果がストップキャラクタでない場合は、ステップ S T 38に移行する。ステップ S T 38で C P U 2 3 は、次キャラクタ(6本後ろの幅データD10)のデコード処理を実行する。その後、ステップ S T 22に戻る。

【0220】

また、ステップ S T 33でデコード結果にエラーキャラクタが存在している場合は、図10に示すステップ S T 39で C P U 2 3 はデコードバッファ25の長さとデコードデータ長とが等しいか否かに基づいて制御を分岐する。デコードバッファ25の長さとデコードデータ長とが等しい場合は、ステップ S T 41で、デコードバッファ25のエラーキャラクタ部分のデコードが成功していたらデータを書き換える。その後、ステップ S T 42に移行して、デコードバッファ25にエラーキャラクタが存在しない否かに基づいて制御を分岐する。デコードバッファ25にエラーキャラクタが存在していない場合は、ステップ S T 35に戻る。

30

【0221】

上述のステップ S T 35でチェックデジットが誤っている場合は、図10に示すステップ S T 43でデコードバッファ25をクリアした後、図9に示したステップ S T 37でデコード結果がスタートキャラクタではない場合、図10に示したステップ S T 39でデコードバッファ25の長さとデコードデータ長とが異なる場合は、ステップ S T 40でデコードバッファ25をクリアした後、及び、ステップ S T 42でデコードバッファ25にエラーキャラクタが存在する場合は、ステップ S T 44に移行して C P U 2 3 は、読み取りエラー処理を実行する。読み取りエラー処理では次のマージンを検索する等が行われる。

40

【0222】

また、キャラクタ認識がエラーとなった場合に、エラーキャラクタを仮に配置してバーコードデータを作成するようにしてもよい。その後、図9に示すステップ S T 45に移行して、 C P U 2 3 は終了を判別する。例えば、当該バーコード記号読取装置100に対して読み取終了指令が検出され、読み取終了指令が検出された場合は、バーコード記号1の読み取

50

り制御を終了する。読み取り終了指令が検出されていない場合は、バーコード記号1の読み取り制御を継続するべく、ステップS T 21に戻って、上述した内容を繰り返すようになされる。

【0223】

このように第2の実施例に係るバーコード記号読み取り装置100によれば、パターン認識する加算範囲を対象キャラクタの両側の2キャラクタC0, C2のエレメントまで拡張した。すなわち、加算範囲に関して、図7Aに示した対象キャラクタと隣接する1つ前のキャラクタC0のスペース(6)、バー(5)、スペース(4)、バー(3)及びスペース(2)の5エレメントまで拡張し、かつ、1つ後ろのキャラクタC2のバー(1)、スペース(2)、バー(3)、スペース(4)及びバー(5)の5エレメントまで拡張したので、51種類の対象キャラクタの中からデコード結果となる候補キャラクタを決定できるようになる。10

【0224】

従って、キャラクタ内の幅データだけでパターン認識を行う従来方式に比べて、対象キャラクタの両側の2キャラクタC0, C2のエレメントも含めた部分についても計算に参画されるので、対象キャラクタの端部に、欠落、傷等の誤差が多く含まれるパターンであっても、誤読し難くなつた。これにより、バー、スペース等のエレメント(エレメント)の幅の最大誤差が±0.5モジュール以上あっても、キャラクタとして認識できる可能性が高くなつた。しかも、対象キャラクタを絞ることによって、計算量の軽減が図れ、パターン認識処理に係る時間が短縮され、データ処理の高速化につながる。20

【実施例3】

【0225】

続いて、図11～図13を参照して、第3の実施例に係る難読キャラクタのデコード例(その1～3)について説明する。この実施例では、第1の実施例で説明したキャラクタデコードアルゴリズムと、第2の実施例で説明したマルチスキヤンリコントラクションとを併合した処理が行われる。これらをデコード条件にして、図11に示すフローチャートのステップS T 51でスタートマージンを検索する。このとき、光学読み取り部2は、第1の実施例と同様にして、バーコード記号1を読み取って、コード読み取りデータD 2を発生する。コード読み取りデータD 2は、光学読み取り部2からカウンタ部4へ出力される。クロック発生部3は、所定周波数のサンプリング用のCLK信号をカウンタ部4に出力する。30

【0226】

カウンタ部4は光学読み取り部2から出力されるコード読み取りデータD 2を入力すると共に、クロック発生部3から出力されるCLK信号を入力し、CLK信号に基づいてコード読み取りデータD 2をカウントし、コード読み取りデータD 2を白部分のスペースおよび黒部分のバーを含むエレメントの1本幅及び2本加算幅を示すカウントデータD 4に変換する。カウントデータD 4は、バーコード記号1における各バー(黒部分)の幅およびスペース(白部分)の幅を示す数値データ列となる。カウントデータD 4は、幅データ格納部10に保存される。なお、スタートマージンの検索については、その説明を省略する。

【0227】

ステップS T 52でCPU23は6本の幅の合計と1キャラクタ幅とを比較し、1キャラクタ分の幅データD 10が適切かを判断する。ここでは、1キャラクタ内で隣接するバー及びスペースを含む6本のエレメントの幅を加算した合計値と、1キャラクタの全幅を示す全幅値とを比較し、比較結果に基づいて1キャラクタ分の幅データD 10が適切か否かを判別する。40

【0228】

この例で、CPU23では比較部23cが1キャラクタ内で隣接する6本のバー(1)、スペース(2)、バー(3)、スペース(4)、バー(5)及びスペース(6)の幅W1～W6の合計と、1キャラクタの全幅を示す全幅値とを比較する。比較部23cによる比較結果によって、1キャラクタ分の幅データD 10が適切か否かが判別でき、この判別に基づいてキャラクタ候補を絞ることができ、計算量の軽減が図れ、処理時間の高速化に50

つながる。

【0229】

ステップS T53でC P U 2 3はバーの分割、消失の有無に基づいて制御を分岐する。バーの分割、消失の有無については、比較部2 3 cによる比較結果を利用する。対象キャラクタC 1の前後のキャラクタC 0, C 2の全幅からキャラクタフレームを判定できるので、ボイドやスポット等によるエレメントに過不足が有っても、デコードを継続できるようになる。

【0230】

バーの分割、消失が無い場合は、ステップS T54でC P U 2 3は1モジュール幅を算出し、1キャラクタ幅を記憶する。このとき、演算部2 3 aは、6種類の幅W 1 1データ～幅W 1 6データに基づいて対象キャラクタC 1の1モジュール幅を算出する。1キャラクタ幅は、図7Bに示した1キャラクタ内で隣接する6本のバー(1)、スペース(2)、バー(3)、スペース(4)、バー(5)及びスペース(6)の幅W 1～W 6を合計して求められる。

10

【0231】

ステップS T55でC P U 2 3は抽出された21種類の幅データD 1 0を1～6本加算パターン(エレメント幅)に変換する。この例では、6種類の幅W 1 1データ～幅W 1 6データが1キャラクタの1本データに変換され、1本データに基づく対象キャラクタC 1の被認識パターンとして、1本パターンが作成される。5種類の幅W 2 1データ～幅W 2 5データは1キャラクタの2本加算データに変換され、2本加算データに基づく対象キャラクタC 1の2本加算パターン(被認識パターン)が作成される。4種類の幅W 3 1データ～幅W 3 4データは1キャラクタの3本加算データに変換され、3本加算データに基づく対象キャラクタC 1の3本加算識パターンが作成される。

20

3種類の幅W 4 1データ～幅W 4 3データは1キャラクタの4本加算データに変換され、4本加算データに基づく対象キャラクタC 1の4本加算パターンが作成される。2種類の幅W 5 1データ及び幅W 5 2データは1キャラクタの5本加算データに変換され、5本加算データに基づく対象キャラクタの5本加算パターンが作成される。1種類の幅W 6 1データは1キャラクタの6本加算データに変換され、6本加算データに基づく対象キャラクタC 4の6本加算パターンが作成される。

30

【0232】

ステップS T56でC P U 2 3はキャラクタを絞り込むために、1本パターン、2～6本加算パターンの全てに対してモジュール数の誤差が1以内のキャラクタを検索して候補キャラクタを決定する。このとき、C P U 2 3では抽出部2 3 bが、演算部2 3 aによって求められた1本パターン、2～6本加算パターンの各々のモジュール数が誤差1以内の整数となる対象キャラクタC 1等を絞り込んで候補キャラクタを抽出する。その際に、抽出部2 3 bは、演算部2 3 aによって求められた1本パターン、2～6本加算パターンの1キャラクタ分の6本の幅データD 1 0と、対象キャラクタの前後の(6-1)本の幅データD 1 0とを抽出する。

【0233】

ステップS T57でC P U 2 3は、候補キャラクタが1つ以上存在するか否かに基づいて制御を分岐する。候補キャラクタが1つ以上(複数)存在する場合は、ステップS T58で1～6本加算パターンの実測値とその理想値との間の誤差を算出し、この誤差の合計が最小となった候補キャラクタをデコード結果とする。この誤差値が最小となる候補キャラクタは、期待値キャラクタと最も相関が強いキャラクタである。この例では、総合的な誤差が最も少ない対象キャラクタC 1等を候補キャラクタとして選択できるようになる。

40

【0234】

このとき、C P U 2 3では、期待値キャラクタ(理論値)が期待値格納部2 4からC P U 2 3の比較部2 3 cへ読み出される。比較部2 3 cは、抽出部2 3 bによって抽出された各々の2～6本加算パターン(エレメント幅)のモジュール数の候補キャラクタと、候補キャラクタを評価するための期待値キャラクタ(理論値)とを比較する。この比較結果

50

は検索部 23d へ出力される。

【0235】

検索部 23d では、候補キャラクタと最も相関が強い期待値キャラクタが検索される。例えば、上述の演算部 23a が候補キャラクタのモジュール数の実測値と、期待値キャラクタのモジュール数の理想値との間の誤差を算出すると、検索部 23d が、演算部 23a によって算出された誤差を合計して得られる誤差値が最小となる候補キャラクタと最も相関が強い期待値キャラクタを検索する。

【0236】

この例では、誤差値が最小となる候補キャラクタと最も相関が強い期待値キャラクタを検索するので、従来方式のバーコード読み取り方法に比べて大幅にバーコードの誤読取を防止できるようになる。なお、ステップ S T 57 で候補キャラクタが存在しない場合は、ステップ S T 59 で現在のキャラクタをエラーキャラクタとする。その後、ステップ S T 60 で次のキャラクタ（キャラクタ幅分後ろのデータ）のデコード処理に移行する。

10

【0237】

上述のステップ S T 58 でデコード結果となされた候補キャラクタについては、図 12 に示すステップ S T 61 で C P U 23 はマージン直後の対象キャラクタか否かに基づいて制御を分岐する。デコード結果となされた候補キャラクタが、マージン直後の対象キャラクタではない場合は、ステップ S T 62 に移行して C P U 23 はデコード結果がストップキャラクタか否かに基づいて制御を分岐する。デコード結果がストップキャラクタである場合は、ステップ S T 63 に移行して C P U 23 は、デコード結果にエラーキャラクタが存在しないか否かに基づいて制御を分岐する。

20

【0238】

デコード結果にエラーキャラクタが存在しない場合は、ステップ S T 64 に移行して、C P U 23 はデコードバッファ 25 に現在（今）のスキャンのデコード結果を上書きする。その後、ステップ S T 65 に移行して C P U 23 は誤り検知（パリティチェック）を実行して、チェックデジットが正しいか否かに基づいて制御を分岐する。チェックデジットが正しい場合は、ステップ S T 66 で C P U 23 は読み取りを完了し、デコードバッファ 25 のバーコードデータ D 12 を出力し、デコードバッファ 25 をクリアする（正常終了）。その後、ステップ S T 65 に移行する。

30

【0239】

なお、ステップ S T 61 でマージン直後の対象キャラクタである場合は、ステップ S T 67 に移行して C P U 23 は、デコード結果がスタートキャラクタか否かに基づいて制御を分岐する。上述のステップ S T 67 でデコード結果がスタートキャラクタである場合及び、ステップ S T 62 でデコード結果がストップキャラクタである場合は、ステップ S T 68 に移行する。ステップ S T 68 で C P U 23 は、次キャラクタ（6 本後ろの幅データ D 10）のデコード処理を実行する。その後、ステップ S T 52 に戻る。

【0240】

また、ステップ S T 63 でデコード結果にエラーキャラクタが存在している場合は、図 13 に示すステップ S T 69 で C P U 23 はデコードバッファ 25 の長さとデコードデータ長とが等しいか否かに基づいて制御を分岐する。デコードバッファ 25 の長さとデコードデータ長とが等しい場合は、ステップ S T 71 で、デコードバッファ 25 のエラーキャラクタ部分のデコードが成功していたらデータを書き換える。その後、ステップ S T 72 に移行して、デコードバッファ 25 にエラーキャラクタが存在しない否かに基づいて制御を分岐する。デコードバッファ 25 にエラーキャラクタが存在していない場合は、ステップ S T 65 に戻る。

40

【0241】

上述のステップ S T 65 でチェックデジットが誤っている場合は、図 13 に示すステップ S T 73 でデコードバッファ 25 をクリアした後、図 12 に示したステップ S T 67 でデコード結果がスタートキャラクタではない場合、図 13 に示したステップ S T 69 でデコードバッファ 25 の長さとデコードデータ長とが異なる場合は、ステップ S T 70 でデコードバッ

50

ファ 2 5 をクリアした後、及び、ステップ S T 72 でデコードバッファ 2 5 にエラー キャラクタが存在する場合は、ステップ S T 74 に移行して C P U 2 3 は、読み取りエラー処理を実行する。読み取りエラー処理では次のマージンを検索する等が行われる。

【 0 2 4 2 】

また、キャラクタ認識がエラーとなった場合に、エラー キャラクタを仮に配置してバーコードデータを作成するようにしてもよい。その後、図 1 2 に示すステップ S T 75 に移行して、C P U 2 3 は終了を判別する。例えば、当該バーコード記号読取装置 1 0 0 に対して読取終了指令が検出され、読取終了指令が検出された場合は、バーコード記号 1 の読み取り制御を終了する。読取終了指令が検出されていない場合は、バーコード記号 1 の読み取り制御を継続するべく、ステップ S T 51 に戻って、上述した内容を繰り返すようになされる。

【 0 2 4 3 】

このように、第 3 の実施例に係るバーコード記号読取装置 1 0 0 によれば、バーコード記号 1 を読み取ってそのデコード結果を出力する場合に、第 1 の実施例で説明したキャラクタデコードアルゴリズムと、第 2 の実施例で説明したマルチスキヤンリコントラクションとを併合した処理が行われる。これを前提にして、検索部 2 3 d が抽出部 2 3 b によって抽出された各々の 2 ~ 6 本加算パターンのモジュール数の候補キャラクタと、当該候補キャラクタを評価するための期待値キャラクタとを比較し、当該候補キャラクタと最も相関が強い期待値キャラクタを検索するようになる。

【 0 2 4 4 】

この検索によって、複数の対象キャラクタから抽出された候補キャラクタと最も相関が強い期待値キャラクタのバーコードデータ D 1 2 を当該バーコード記号 1 の読取時のデコード結果として出力できるようになるので、従来方式のバーコード読み取り方法に比べて大幅にバーコード記号 1 の誤読取を防止できるようになる。

【 0 2 4 5 】

しかも、誤差の範囲を従来方式の ± 0 . 5 から ± 1 . 0 に拡大し、対象キャラクタを増加させている。そして、1 キャラクタ分の 6 本の幅データ D 1 0 と、対象キャラクタの前後の 5 本の幅データ D 1 0 とを抽出するので、2 ~ 6 本加算パターンの各々のモジュール数が誤差 1 以内の整数となる対象キャラクタを絞り込んだ候補キャラクタを従来方法に比べてより多く抽出できるようになる。また、候補キャラクタを絞ることによって、計算量の軽減が図れ、パターン認識処理に係る時間が短縮され、データ処理の高速化につながる。これにより、従来方式のバーコード読み取り方法に比べて大幅にバーコードの誤読取を防止できるようになる。なお、誤差の範囲を設定する際の閾値を可変することで、幅データ D 1 0 の信頼性を選択することも可能となる。

【 0 2 4 6 】

この例では、バーコード記号の規格に関して、C O D E 1 2 8 の場合について説明したが、これに限られることはなく、C O D E 3 9 、E A N / U P C 、インターリーブド 2 オブ 5 、コードバー、R S S 、R S S リミテッド、R S S エクスパンデド等の各々の方式についても、本発明を同様にして適用できるようになる。

【 産業上の利用可能性 】

【 0 2 4 7 】

本発明は、バーコード記号を読み取ってバーコードデータを出力するバーコード記号読取装置に適用して極めて好適である。

【 符号の説明 】

【 0 2 4 8 】

2 . . . 光学情報読取部、3 . . . クロック発生部、4 . . . カウンタ部、5 . . . m 段データバッファ、6 . . . セレクタ部、7 . . . x 倍器、8 . . . 比較器、. . . スタートマージン格納部、1 0 . . . 幅データ格納部、1 1 . . . インターフェース部、1 2 . . . マイクロプロセッサ、2 1 . . . R O M 、2 2 . . . R A M 、2 3 . . . C P U 、2 3 a . . . 演算部（演算手段）、2 3 b . . . 抽出部（抽出手段）、2 3 c . . .

10

20

30

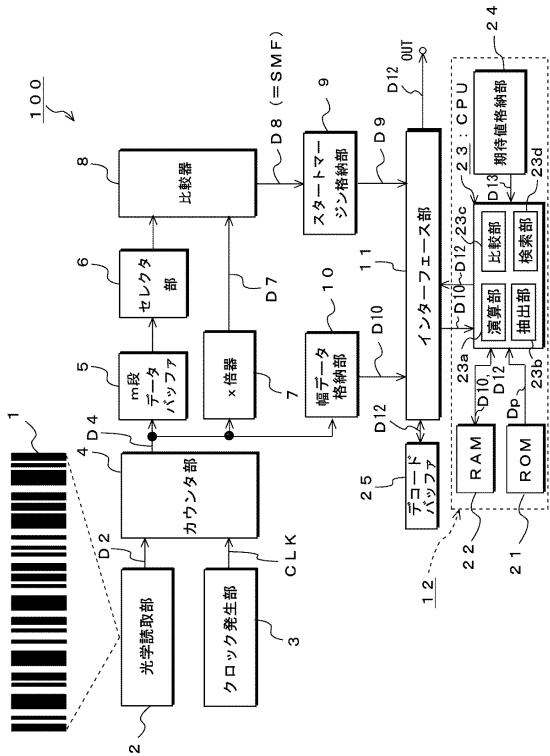
40

50

比較部、23d・・・検索部、24・・・期待値格納部、25・・・デコードバッファ及び、100・・・バーコード記号読取装置（光学情報読取装置）

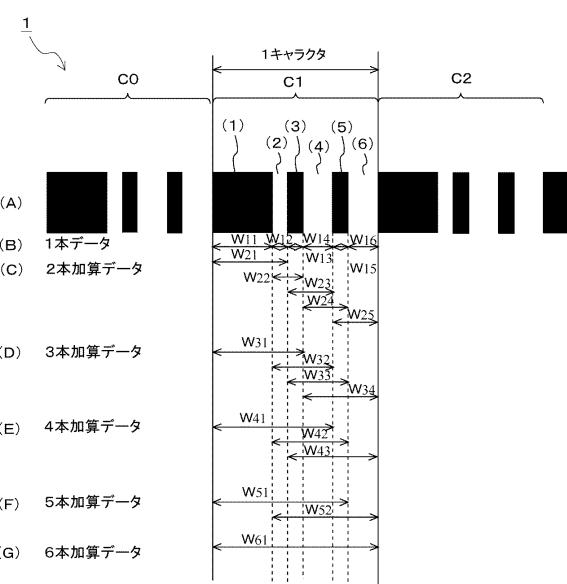
【 図 1 】

実施形態としてのバーコード記号読取装置100の構成例



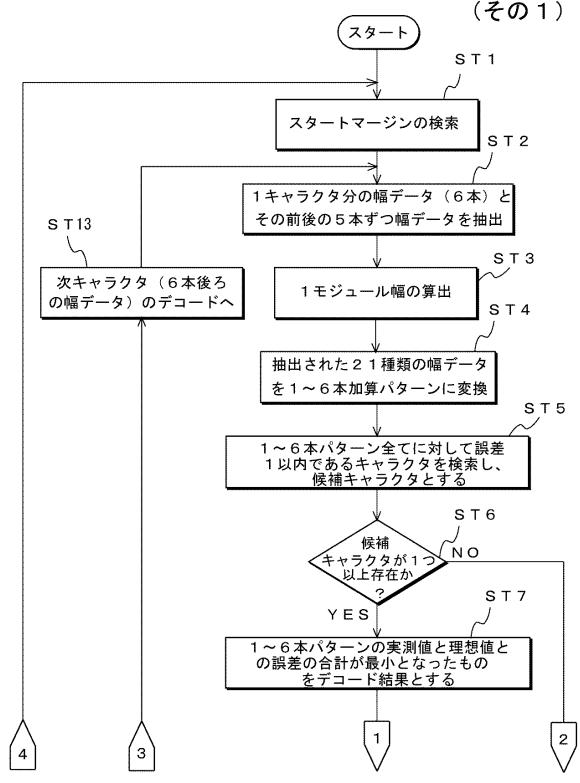
【 図 2 】

第1の実施例に係る1本、2~6本加算データの算出例



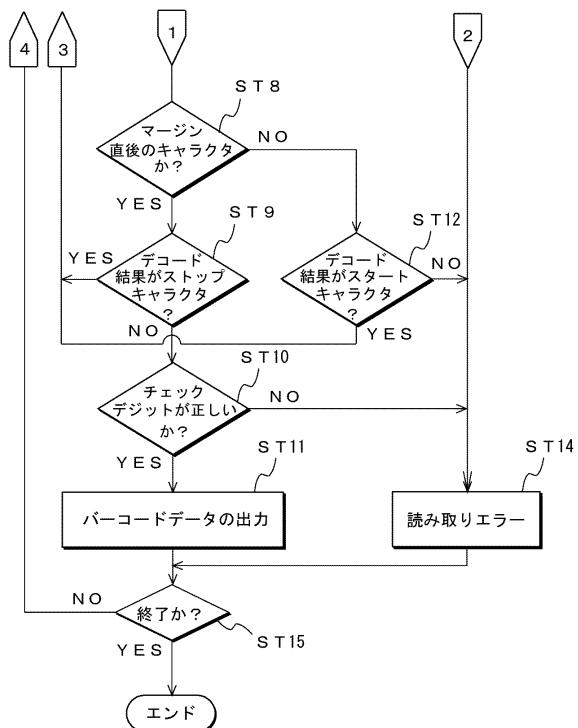
【図3】

第1の実施例に係る難読キャラクタの解読例
(その1)



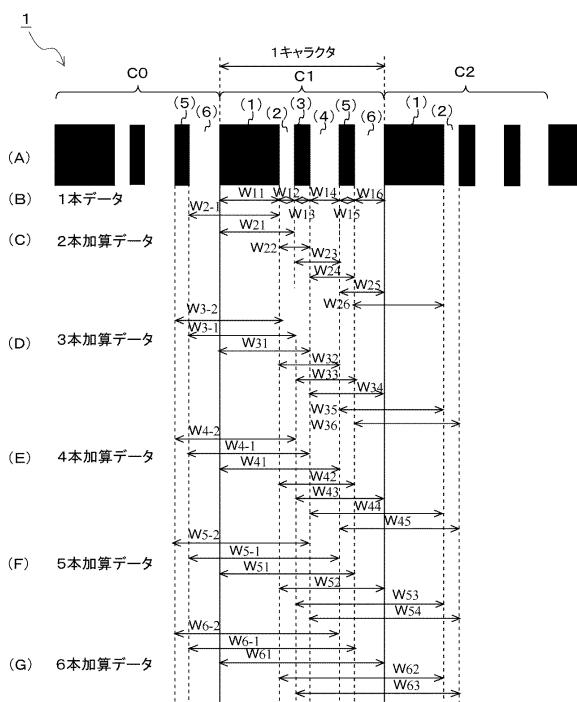
【図4】

第1の実施例に係る難読キャラクタの解読例 (その2)



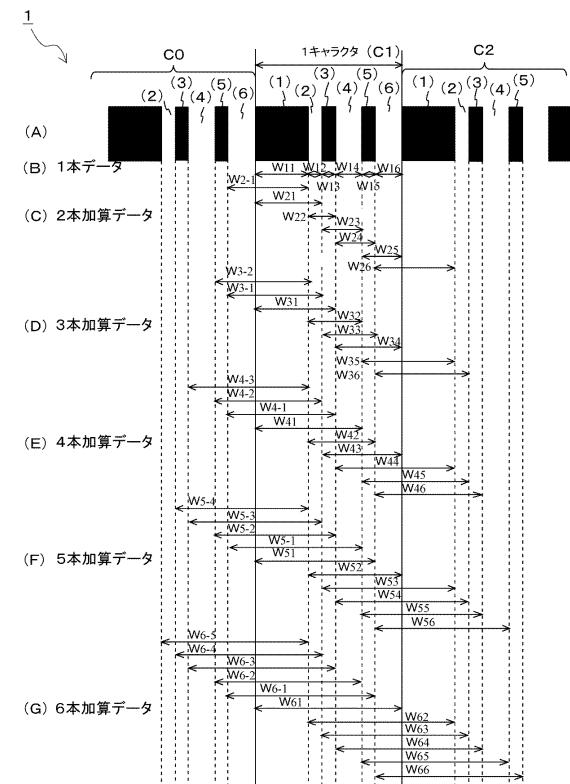
【図5】

1本データ、2~6本加算データの他の算出例(その1)



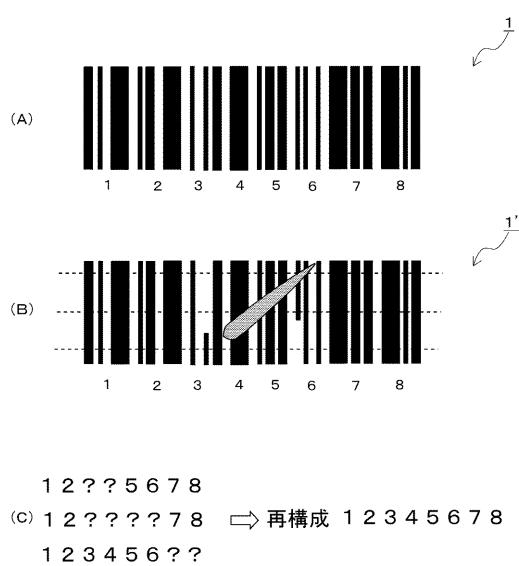
【図6】

1本データ、2~6本加算データの他の算出例(その2)



【図7】

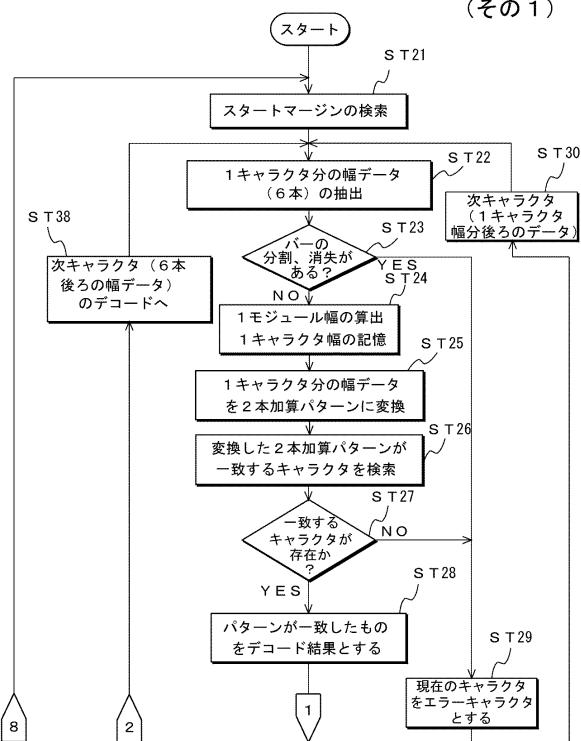
第2の実施例に係るバーコードデータの再生例



【図8】

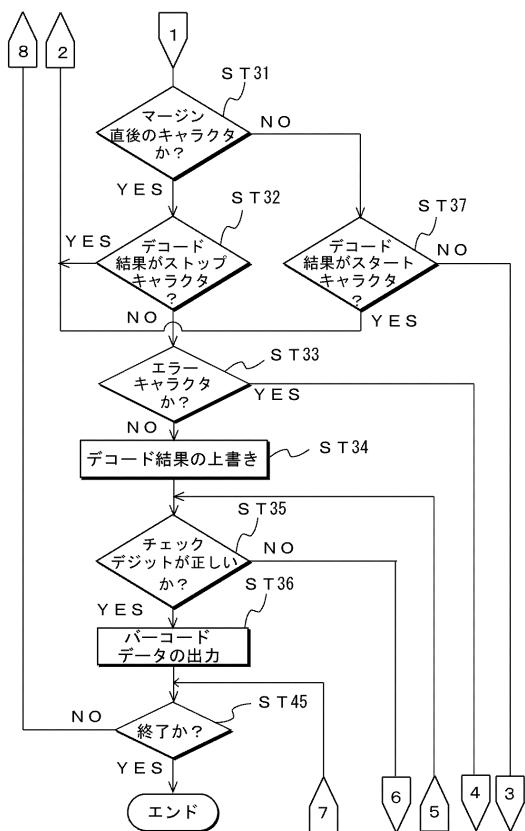
第2の実施例に係る難読キャラクタの解読例

(その1)



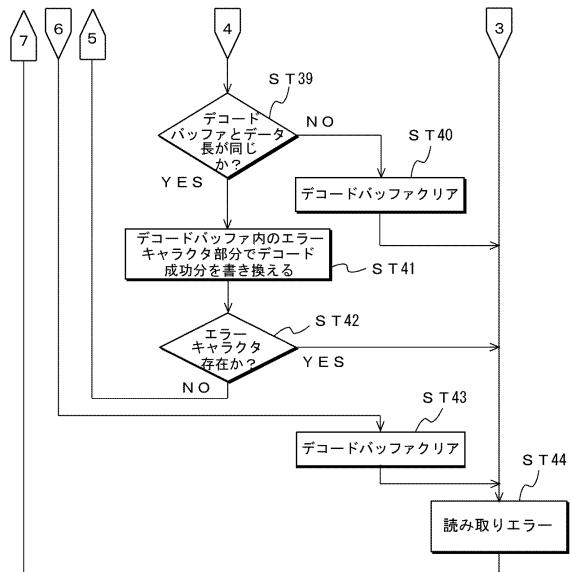
【図9】

第2の実施例に係る難読キャラクタの解読例 (その2)



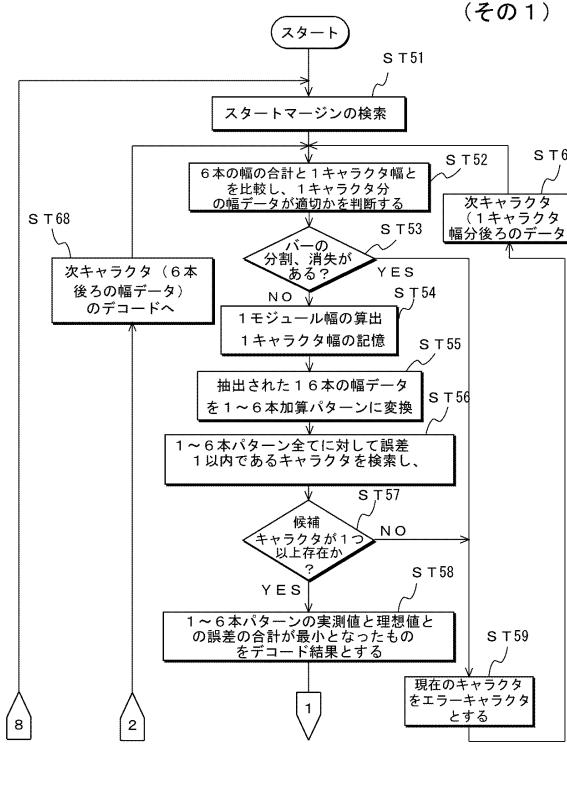
【図10】

第2の実施例に係る難読キャラクタの解読例 (その3)



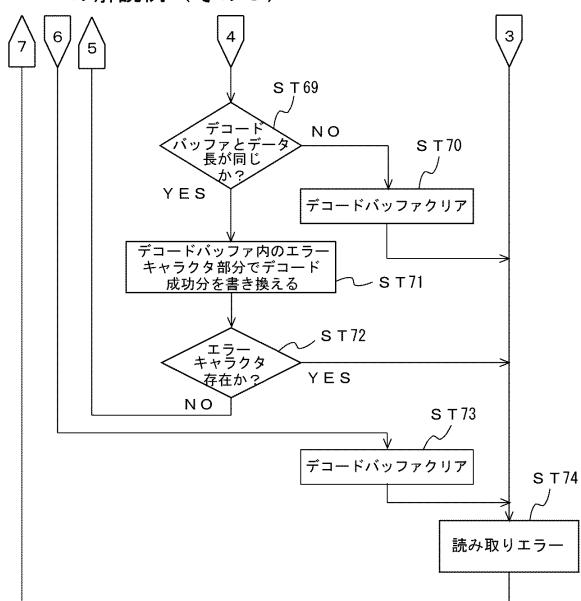
【図11】

第3の実施例に係る難読キャラクタの解読例
(その1)



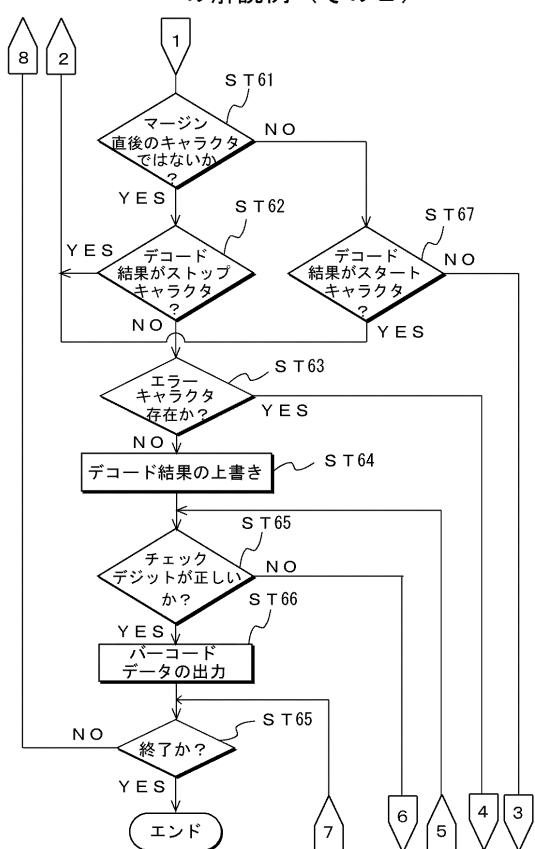
【図13】

第2の実施例に係る難読キャラクタの解読例
(その3)



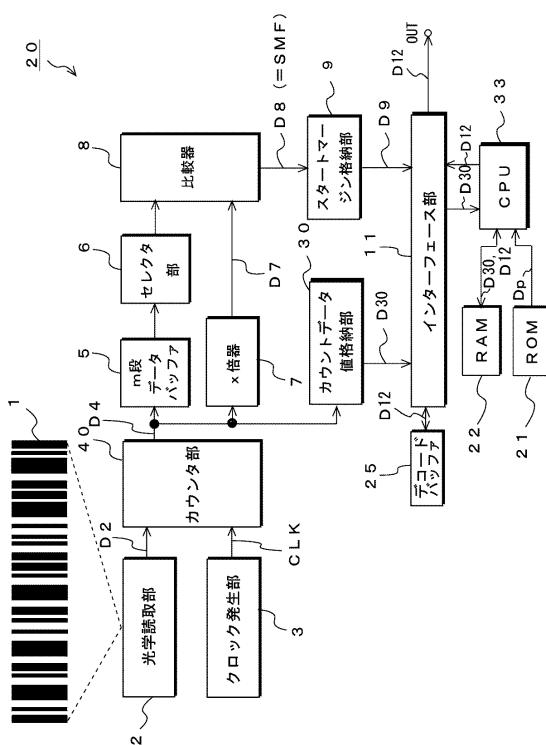
【図12】

第3の実施例に係る難読キャラクタの解読例
(その2)



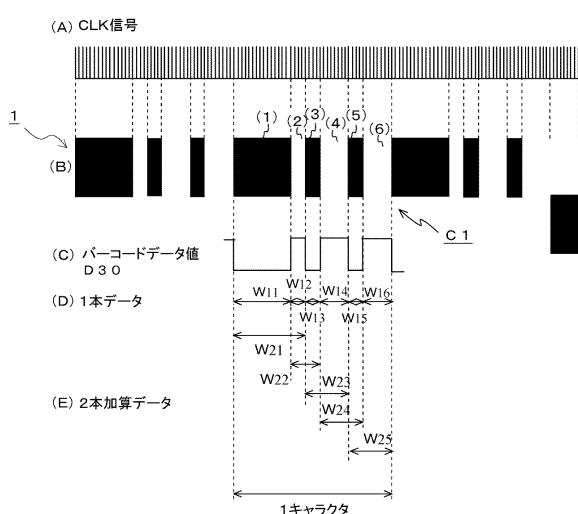
【図14】

従来例に係る光学情報読取装置20の構成例



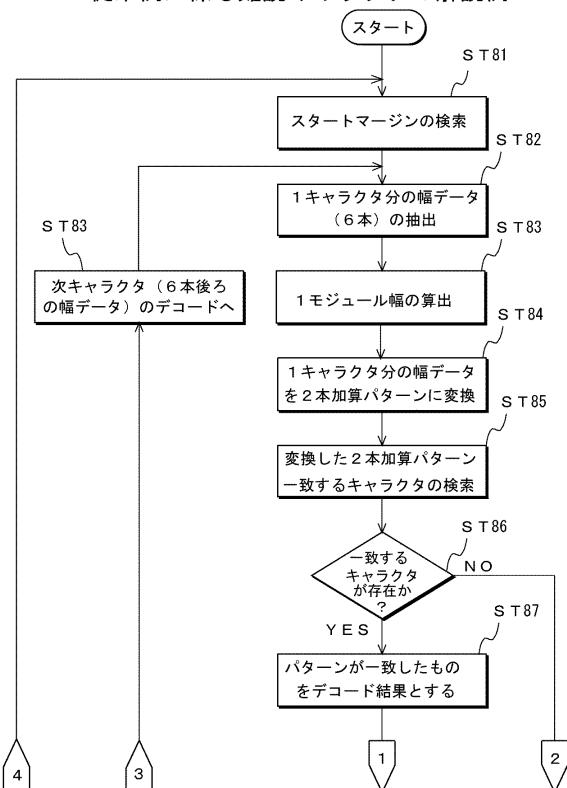
【図15】

従来例に係る2本加算データの算出例



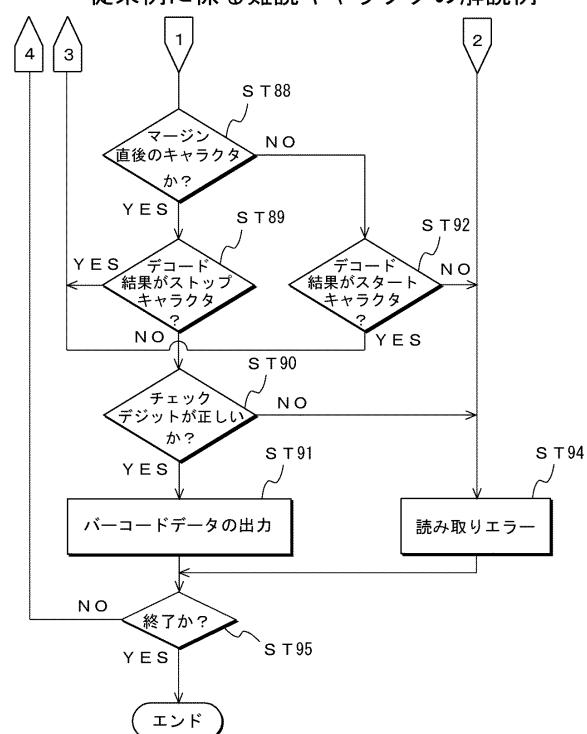
【図16】

従来例に係る難読キャラクタの解読例



【図17】

従来例に係る難読キャラクタの解読例



フロントページの続き

(56)参考文献 特開平10-187874(JP,A)
特開2005-128626(JP,A)
特開平06-036065(JP,A)
特開昭63-155381(JP,A)

(58)調査した分野(Int.Cl., DB名)
G 06 K 7/00 - 7/14