【公報種別】特許法第１７条の２の規定による補正の掲載
【部門区分】第７部門第３区分
【発行日】平成21年8月20日(2009.8.20)

【公開番号】特開2007-37115(P2007-37115A)
【公開日】平成19年2月8日(2007.2.8)
【年通号数】公開・登録公報2007-005
【出願番号】特願2006-189358(P2006-189358)
【国際特許分類】
Ｈ０３Ｍ　7/30　(2006.01)
【ＦＩ】
Ｈ０３Ｍ　7/30　Ｚ

【手続補正書】
【提出日】平成21年7月3日(2009.7.3)
【手続補正１】
【補正対象書類名】明細書
【補正対象項目名】００２５
【補正方法】変更
【補正の内容】
【００２５】
パラレルエンコーダ３００のヘッド制御３０１は入力データをデータのブロックに分割またはフォーマットしてもよく、データの各々のブロックは他のすべてのブロックから独立して圧縮される。ヘッド制御３０１は、さらに、各々のブロックが符号化のためにどのシリアルエンコーダに送られるかを選択してもよい。各々のブロックは異なるシリアルエンコーダにルーティングされてもよい。ヘッド制御３０１は、データの所与のブロックのためにどのシリアルエンコーダが使用されるべきであるかを判断するためのいくつかの異なるアルゴリズム、たとえばラウンドロビンおよび最も空いているノードスキームを使用してもよい。各々のシリアルエンコーダ２００－１から２００－ｎは、入力データをステージするためにその独自のヘッドＦＩＦＯを有してもよい。各々のシリアルエンコーダ２００－１から２００－ｎは、さらに、圧縮されたデータストリームが装置に出力されるときに、圧縮されたセグメントの組立順序が確実に維持されるのを助けるようにその独自の末尾ＦＩＦＯを有してもよい。ヘッド制御３０１は、任意のヘッドＦＩＦＯまたは末尾ＦＩＦＯを含むシリアルエンコーダ２００－１から２００－ｎを通されるデータの各々のブロック上にあるセグメント制御フィールドを変更されないまま維持してもよい。セグメント制御フィールドは、シリアルエンコーダからのいずれの圧縮されたデータストリームの前にも、末尾ＦＩＦＯに書込まれてもよい。セグメント制御フィールドは、シーケンシャル番号付または主なＣＲＣの使用を含むいくつかの異なるフォーマットを有してもよい。ヘッド制御３０１は、渡されるデータレコードサイズを使用してもよく、符号化データのブロック内のどこに第１のデータレコードが位置するかを判断するために復号中に使用され得る第１のレコードポインタを生成してもよい。
【手続補正２】
【補正対象書類名】明細書
【補正対象項目名】００２６
【補正方法】変更
【補正の内容】
【００２６】
パラレルエンコーダの末尾制御３０２は、各々のシリアルエンコーダ２００－１から２００－ｎの末尾ＦＩＦＯを監視してもよい。末尾制御３０２は、各々のシリアルエンコーダの末尾ＦＩＦＯに存在するセグメント制御フィールドを調べることによって、どのシリ

アルエンコーダが次のシーケンシャルセグメントを含むかを判断し得る。末尾制御３０２は、データの対応するブロックがシリアルエンコーダ２００－１から２００－ｎに入ったのと同じ順序で、符号化データのブロックが送出されることを確実にすることができる。末尾制御３０２は、コード化データのストリームとしてデータを出力する前に、符号化データのブロックからセグメント制御フィールドを取除いてもよい。

【手続補正３】
【補正対象書類名】明細書
【補正対象項目名】００３６
【補正方法】変更
【補正の内容】
【００３６】
履歴バッファ４００に置かれる各々の新しいバイトごとに、対応するビットが位置付メモリ５００においてセットされる。結果として生じる位置付メモリ５００は、いずれか１つの値が履歴バッファ４００の中のどこに位置するかをすばやく判断するために使用されてもよい。たとえば、０ｘ５５という１６進数の値すなわち８５という１０進数の値が履歴バッファ４００の中のどこに位置するかを判断するために、ストリングマッチング論理２０２は位置付メモリ５００のアドレス８５にアクセスしてもよい。アドレス８５を有するメモリにおいて１にセットされる各々のビットは、１０進数の値８５を含む履歴バッファ４００での位置を表わす。

【手続補正４】
【補正対象書類名】明細書
【補正対象項目名】００４１
【補正方法】変更
【補正の内容】
【００４１】
図７は、この発明の実施例に従うマッチレジスタ６００の使用を示す。マッチレジスタ６００は、一致が存在するかどうかおよびその一致までのオフセットを判断するために使用され得る。マッチレジスタ６００は０に初期化されてもよい。以下に記載されるように、処理後、マッチレジスタ６００は０ではない値を有してもよい。マッチレジスタ６００における０ではない値は、以前の１つ以上の文字との一致が検出されたことを示す。示される例では、現在の位置７００はビット７にあり、これはアドレス７において履歴バッファ４００の中に置かれる新しいバイトに対応する。現在の位置７００から、最も近いセットビット７０１までのビットの数は一致オフセットを示す。ここでは、現在の位置７００からビット位置０にある第１のセットビット７０１までに７ビット存在する。したがって、一致が存在し、一致オフセットは７である。