

可変長文字列検索連想メモリによる LZ77 高速データ圧縮 LSI (2)

6P-7

佐藤 証 新島 秀人

日本アイ・ビー・エム株式会社 東京基礎研究所

1. はじめに

LZ77 符号はデータ圧縮時において可変長文字列検索に膨大な処理時間を要していたが、連想メモリを用いることで高速に実行することが可能となった。本稿では高速データ圧縮 LSI に採用された圧縮フォーマットを示し、また可変長文字列連想メモリを用いた圧縮 LSI の動作や注意点をフローチャートを交えて解説する。

2. 圧縮データフォーマット

LZ77 符号は入力データをバッファに貯め、繰り返し現れる文字列をバッファ内の“アドレス m から x 文字分と一致”といった符号に変換する。図 1 は今回のデータ圧縮 LSI で用いた圧縮フォーマットであり、これは 1/4 インチテープカートリッジの標準フォーマット QIC-154 として採用されている。1 文字は 8 ビットでバッファサイズは 512, 1024, 2048 文字分の 3 種類があるので一致位置 m を示すのに 9~11 ビットを割り当てる。一致長 l は固定長ではなく、表 1 に示すように高い頻度で現れる短い繰り返しには少ないビットを、低頻度の長い繰り返しには多くのビットを割り当てている。2 文字以上の一致で圧縮を行い、1 文字だけの一致は不一致と同様に扱い文字をそのまま出力するが、圧縮データと区別するために非圧縮データはフラグビット 0 を、圧縮データはフラグビット 1 を先頭に付加する。

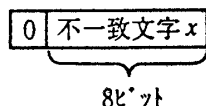
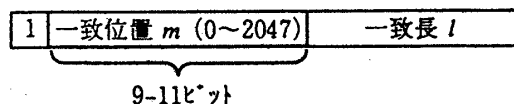
● 非圧縮データ ($l \leq 1$)● 圧縮データ ($l \geq 2$)

図 1 QIC-154 フォーマット

QIC-154 以前に標準フォーマットとして採用されている QIC-122 は一致位置 l に相対距離を用いており、現在の入力から 127 文字前以内に一致したときには 8 ビットが、128~2048 文字前には 12 ビットが一致位置 l に割り当てられる。これは繰り返しが局所的に発生する可能性

が高いためである。これに対して QIC-154 で絶対アドレスを採用したのは、複数の一致アドレスから 1 つを選択して出力するプライオリティーエンコーダ回路において、選択されるアドレスは低位側を優先するように固定されており、入力された文字から近いアドレスを選択することが難しいためである。しかしながら QIC-154 の一致長は $l=4$ を除く全てにおいて QIC-122 と同じか少ないビット数で済むため、結果的に圧縮率は優れている。

表 1 QIC-122 と QIC-154 の一致長符号

l	QIC-122	QIC-154
2	00	00
3	01	01
4	10	1 000
5	1100	1 001
6	1101	1 010
7	1110	1 011
8	1111 0000	11 0000
9	1111 0001	11 0001
⋮	⋮	⋮
15	1111 0111	11 0111
16	1111 1000	111 00000
17	1111 1001	111 00001
⋮	⋮	⋮
22	1111 1110	111 00110
23	1111 1111 0000	111 00111
24	1111 1111 0001	111 01000
⋮	⋮	⋮
31	1111 1111 1000	111 01111
32	1111 1111 1001	1111 000000
33	1111 1111 1010	1111 000001
⋮	⋮	⋮

3. データ圧縮回路における連想メモリの制御

データ圧縮回路は入力データをバイト文字に区切りながら連想メモリに入力し、その一致結果に応じて上記のフォーマットに従った圧縮データを出力する。

ところで圧縮の初期においては連想メモリに比較すべき有効なデータが存在せず、入力が無効データと一致してしまつては正しい圧縮が行えない。そこで図 2 に示す様に、連想メモリの各行にはデータの有効・無効を示す 1 ビットのレジスタが設けられている。レジスタは回路の初期化時に 0 にリセットされ、一旦その行にデータが書き込まれた後はデータ入力終了するまで 1 を保持する。そして各マッチ線はこのレジスタ出力と AND され、無効データがマスクされた後 ML レジスタにラッチされる。またデータの書き込みと検索は並列に実行されるため、そのままでは書き込みアドレスは常に一致してしまうことになる。これを避けるために、ワード線がアクティヴである書き込みアドレスに対しては、有効・無効ビットの値にかかわらずマッチ線をマスクしている。

