

## 5T-6 ATTEMPT における IEEE Futurebus インタフェース

呉燦文、天野英晴

慶応義塾大学理工学部

### 1. はじめに

将来、数プロセッサによる小規模マルチプロセッサ構成が一般的になる可能性が高いと言われているが、この時内部で用いられるバスは現在の VMEbus、Multibus ではその機能が不足する。IEEE Futurebus<sup>[1]</sup>はこのような必要性に答えるべく、検討が進められてきたマルチプロセッサシステム用の標準化バスである。Futurebus のプロトコルは以下のように定義されている。

**P896.1** 電気的特性、信号線等ハードウェアについてのプロトコル

**P896.2** キャッシュの制御等ファームウェアについてのプロトコル

**P896.3** さらに上位のソフトウェアに関するプロトコル

このうち P896.2、P896.3 は検討されている段階であるが、P896.1 はすでに決定され、1988 年 12 月に正式の仕様書が配布された。テストベットとしての拡張性を高めるため、ATTEMPT は商用機としてはおそらく世界で初めてこのバスを用いることにした。本報告ではこのバスについて簡単に紹介し、実装における問題点を述べる。

### 2. Futurebus の特徴

従来のバスと比較したときの Futurebus の主な特徴を以下に示す。

1. 完全な非同期バスである。
2. アドレスと、データは、32bit のライン上でマルチプレクスされる。
3. 全てのラインはオープンコレクタである。アービトレーションもオープンコレクタを利用して行われる。
4. データのブロック転送、ブロードキャストが重視されている。
5. 耐故障性について配慮されている。

1 回のアービトレーションで決定されたマスタは、BUS TENURE といわれる一連のシーケンスを確保する。各シーケンスは基本的にブロック転送の形式を取る。すなわち、まずアドレスを送り、次にデータが連続して送られる。もちろんアドレスだけ、データ 1 個だけの転送もこの中に含まれる。

非同期バスであるため、1 回のデータ転送毎にハンドシェイクをとる必要がある。ここで、ブロードキャストかどうかでハンドシェイクの方法は違ってくる。ストロブ信号と同時にマスタはコマンド情報を用いて、ブロードキャストかどうかをスレーブに知らせ

る。ブロードキャストでない場合、転送は応答信号のみを用いた 2 線ハンドシェイクで行われ、そうでなければ、反転応答信号をこれに加えた 3 線ハンドシェイクによって行われる。後者の場合、オープンコレクタバスの特徴が生かされる。すなわち、全スレーブが反転応答信号を H レベルを出力した時、はじめて信号線は実際に H レベルになり、マスタはスレーブ全ての準備ができたことを検出できる。

このようなハンドシェイクにより、転送効率は同期バスに比べ低下する。このことを防ぐため、Futurebus ではブロック転送用に連続転送モードを用意している。この方法は 2 エッジハンドシェイクといい、ストロブ信号と応答信号の上下エッジが交互に用いられ、データは連続転送される。このバスを実装する上での問題点は以下の 2 点である。

1. 制御線がオープンコレクタであるためこのライン上のノイズ、特にワイヤド OR グリッジと呼ばれる現象が誤動作の原因になる。
2. バス制御、特にアービトレーションの制御が複雑でインタフェースの実装が難しい。

以下、これらの問題点について検討する。

### 3. バスのアナログ的問題

#### 3.1 バックプレーンについて

Futurebus の電気的特性は、転送能力を高めるために振幅を 1V-2V の範囲で狭くとり、特殊なドライバが必要である。これには NS 社の DS3896、DS3897 が使用可能である。これらのドライバは 2V の安定化電源に対し終端抵抗を介して接続される。現在 Futurebus 専用のバックプレーンは市販されていないが、基板および筐体のサイズは VMEbus と互換性があるため流用が可能である。この二種類のバックプレーンについてテストした結果、下のような無負荷時特性が得られ、厳密には規格に一致しないものの使用に耐えることが明らかになった。

- 無負荷時伝搬遅延時間: 5.54nsec/20 スロット
- 負荷時伝搬遅延時間: 1 ドライバ当たり約 1nsec 増加
- 無負荷時線路インピーダンス: 180Ω

#### 3.2 ワイヤド OR グリッジについて

Futurebus において 1 本の制御線上でオープンコレクタを使って論理 OR を取る必要がある。今、制御線の両端にそれぞれ 1 つのトランジスタをつなぎ、共にオンの状態にあり、終端抵抗により均等に電流が流れているとする(図 1)。この時に B 点のトランジスタをオフにすると、B 点に流れていた電流が A 点に対して流れ込むため、線上のレベルが一時的に上がり、グリッジを生ずる。これがワイヤドグリッジと呼ばれる現象である。この現象を確認するため先に述べたバックプレーンを用いて実験を行なった。

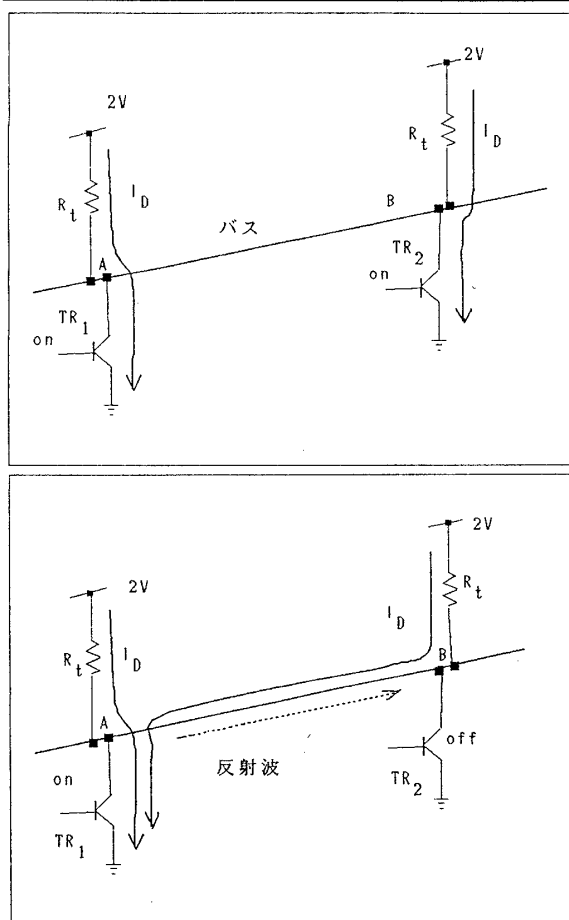


図1 ワイヤドORグリッジの発生原理

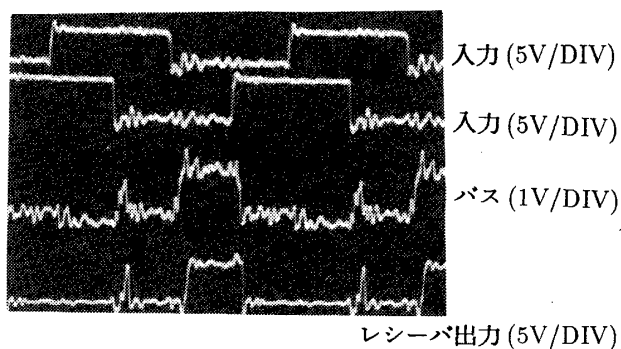
ここで、終端抵抗の整合が取れていない場合には、かなり激しくこの現象が起ることが確かめられた(写真1)。これを抑えるために、レシーバの出力側にCR積分器を入れて整形してみた。終端抵抗の不整合がはなはだしくなれば、簡単なCR積分器で除去できることがわかった(写真2)。写真2における積分器により約17nsecの余分な遅延が生ずるが、この程度は許容範囲内である。積分器は全ての制御線およびアービトレーション用データ線のレシーバに必要である。

#### 4. インタフェースの設計

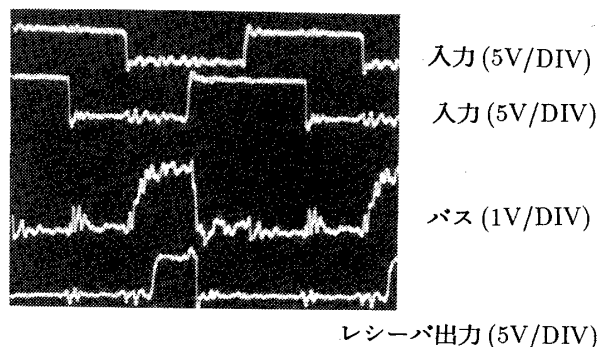
ATTEMPTにおいてはキャッシュ、同期機構はそれぞれコントローラを持っており、これがFuturebusのブロック転送等の制御を行なうため、インタフェースの機能は以下の点に限定されている。

- 電気的バッファ
- アドレス、データの保持とパリティチェック
- コマンド、ステータスの生成、解読
- スレープ時の応答信号の発生
- アービトレーション

このうち最も複雑なのはアービトレーションである。Futurebusのアービタはオープンコレクタの性質を利用した巧妙な方法であるが、非同期バスであるため



入力周波数 2MHz

写真1 ワイヤドORグリッジの例  
(終端:47Ω)

入力周波数 2MHz

写真2 ワイヤドORグリッジの例  
(終端:180Ω、積分器 220Ω,110pF)

全ボードが同期をとることが難しい。このためAP, AQ, ARの3本の制御線をラウンドロビン方式で線ハンドシェイクを取り、アービトレーション用シーケンスをつくる。ATTEMPTにおいてはPLAを用いたコントローラでこの制御を行なっている。全体で約50個のTTLが必要で特にアドレス、データ保持用のレジスタ、パリティチェッカがかなりの面積をしめる。

#### 5. 終わりに

Futurebusは非常に優れた性能を持つ一方、インタフェースにかなりのハードウェアを要する。このバスが一般化するためには少なくとも今回設計したインタフェース程度の機能を持つコントローラチップが必要であろう。とくにアービトレーション部のチップ化はATTEMPTにとってもこのバスの普及にとっても早急に解決しなければならない課題である。

#### 参考文献

- [1] IEEE P896 Working Group, "Futurebus P896.1 A Backplane Bus Specification for Multiprocessor Architectures," Draft 7.5a June, 1987.