

バス制御を伴う分散型非同期式アービタの構成法

小田桐 由樹^{*†}, 赤利 昌紀[†], 近藤 真史^{††}, 横川 智教[†], 佐藤 洋一郎[†], 有本 和民[†]
 (†岡山県立大学, ††川崎医療福祉大学)

1. まえがき

環状に配置された複数のバスを動的に接続する形態であるリングセグメントバス (RSB) は、優れた速度性能を期待できる一方、バスの制御をグローバルアービタ (*GArb*) が集中的に行っているため、拡張性が著しく低いという問題がある。そこで本研究では、バスの制御を分散的に行う (分散制御型) *GArb* の構成法について検討している。

2. 分散制御型リングセグメントバス

分散制御型 *GArb* に基づく RSB の構成を図 1 に示す。クロックドメイン (*CD*) は、共通のクロックで動作する回路要素の集合であり、同期的なデータ転送を行う。一方、*CD* 間での非同期的なデータ転送は、セグメントブリッジ (*SBrg*) を介してそれに至るまでのセグメント (*Seg*) を接続する必要がある。環状に配置された *GCell* がその処理を担う。

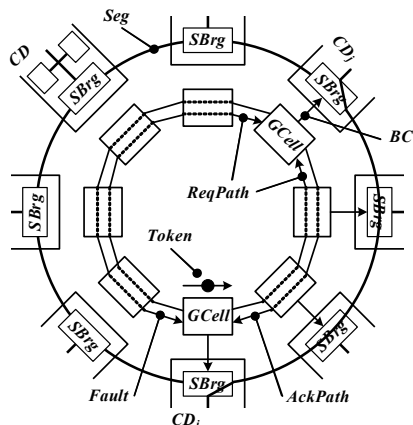


図 1: 分散制御型 *GArb* に基づく RSB の構成

転送元及び転送先 *CD* をそれぞれ CD_i 及び CD_j とすると、データ転送経路は次の手順で構築できる。まず、各 *GCell* にはトークンリング方式に基づいて単一のトークン (*Token*) が循環しており、トークンを獲得した *GCell* のみが以降の処理を行うことができる。 CD_i の *GCell* がトークンを獲得すると、双方向へ構築要求 (*ReqPath*) を送信する。また、途上の *GCell* は、*ReqPath* を後続の *GCell* に伝搬する。そして、 CD_j の *GCell* は、先に *ReqPath* が到達した経路に対して構築承認 (*AckPath*) を、他方の経路に対して *Fault* をそれぞれ返送する。*AckPath* の返送を受けた *GCell* は、*SBrg* に対して転送経路の構築信号 (*BC*) を送信する。

3. *GCell* の回路構成

GCell の動作は、属する *CD* の状況によって、送信動作 (転送元 *CD*)、受信動作 (転送先 *CD*)、伝搬動作 (途上の *CD*) に大別され、それぞれの動作は排他的か

つ独立に処理される。したがって、これらの動作を単位として *GCell* は分割可能であり、属する *CD* の状況に応じて各動作に対応する回路機構を切り替えればよい。これに基づいた *GCell* の構成を図 2 に示す。

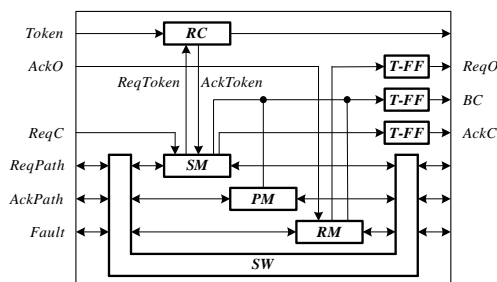


図 2: *GCell* の構成

RC: トークンの授受を行う回路であり、要求 (*ReqToken*) が生起している場合はその承認 (*AckToken*) を返し、生起していない場合はトークンを後続の *GCell* へ送る。

SM: 送信動作を担う回路機構であり、転送元 *CD* からバスの使用要求 (*ReqC*) を受けるとトークンを要求し、その承認を経て *ReqPath* を送信する。*AckPath* が返送されると、*BC* を更新するとともに、使用承認 (*AckC*) を送信する。

PM: 伝搬動作を担う回路機構であり、前段の *GCell* から受信した信号を後段の *GCell* に送信する。また、*AckPath* が返送されると *BC* を更新する。対応する *Seg* が使用中の場合、*Fault* を返送する。

RM: 受信動作を担う回路機構であり、*ReqPath* を受信すると、転送先 *CD* に対して *ReqO* の使用要求 (*ReqO*) を送信する。その承認 (*AckO*) を受信すると、先に *ReqPath* が到達した経路に対して *AckPath* を、他方の経路に対して *Fault* をそれぞれ返送するとともに *BC* を更新する。

SW: 属する *CD* の状況に応じ、*ReqPath*, *AckPath*, *Fault* を各回路機構に接続する回路である。

上記の構成で重要な点は、*GCell* が発する信号の制御に *T-FF* を用いる点である。これは、転送経路の構築・解放処理がトグル動作を行う点に着目し、これらを共通の回路機構で処理し、その制御構造を簡略化するためである。そして、以上に基づいた *GArb* を *Petrify4.2* により非同期式回路として設計・合成を行った結果、論理レベルで所望の動作が実現できたことを確認した。

4. あとがき

本研究で提案した *GArb* に依れば、設計時に配線遅延を考慮する必要がなく、回路実装時の配線遅延に基づいて最短となる経路が構築されるため、更に優れた速度性能と拡張性を実現できるものと考えられる。