

高速シリアルリンク結合に基づくアーキテクチャの ボードレベルシミュレーション

新野 竜太[†] 松本 尚^{††} 平木 敬[†]

本研究では高速シリアルリンクをインタフェースにもつアーキテクチャのハードウェアシミュレーション環境を考察する。主要ブロック制御部は再構成可能デバイスにマッピングして、測定対象の仕様に応じて内部回路を再構成する。シミュレータ内の各ブロックをターゲットの仕様に対し一律にスケールダウンすることで、大規模な性能評価を正確かつ高速に行う。本稿ではこの特性に基づいた並列計算機プロトタイプお茶の水7号を構築した。評価対象は高速シリアルリンクでメモリ、プロセッサチップを結合した並列アーキテクチャであり、トポロジーは必要なチップ間バンド幅、メモリ量によって可変である。想定されたターゲットの性能に応じ、基板内のパラメータと回路を変更し、シリアルリンク部の評価を行なう。

Board-Level Simulation of Architecture Based with High-Speed Serial Links

RYUTA NIINO[†], TAKASHI MATSUMOTO^{††} and KEI HIRAKI[†]

We examine hardware simulation environment for the architecture based on linkage with high-speed serial links. Reconfigurable devices emulate major blocks such as memory controllers or link controllers. We reconstruct internal circuits of the devices according to specifications of the simulation targets. They simulate the target blocks by scaling down all the specification at a constant rate. We simulate architecture with OCHA-7, a parallel computer prototype based on the above-mentioned architecture. The target is parallel architecture that high-speed serial line linked between memory chips and processor chips. We re-compose its topology according to required bandwidth and memories. We modify the internal parameters and circuits on boards according to specification of the target, and evaluate the serial link blocks.

1. はじめに

半導体技術の進歩により、1チップに埋め込めるゲート数は急激に増加し、動作検証、機能検証に設計時間の多くを要求される。更にキャッシュアーキテクチャなどの有効性を検証する際に大規模なベンチマークを走らせる必要がある。

ターゲットの環境に近い性能評価方法として、プロトタイプチップ上の検証がある。非常に高速な検証が可能だが、作成に時間とコストを要する、パラメータの変更が困難または不可能である等大きな欠点がある。ターゲットの仕様がほぼ固定後の最終段階でのハードウェア性能検証、機能検証には用いられるが、未定のパラメータで最適な数値を選ぶために複数のモデルを作成し、比

較検証する目的には向かない。さらに将来実現される仕様の検証は不可能である。

一方、ソフトウェアによるシミュレーションは柔軟だが非常に遅い。高速なサイクルベースシミュレータでも1kHzにも満たない。さらに回路規模が増えるにつれシミュレーション時間も増加する。専用のハードウェアエンジンがソフトウェアを支援することで高速化したシミュレーション環境がある。特にFPGA(Field Programmable Gate Array)、FPIC(Field Programmable Interconnect Chip)を用いた専用ボード上でのシミュレーションはチップ、アーキテクチャの検証に広く使われている[2][5][15]。この方法は従来のソフトウェアのみのシミュレーションに比べ、約 10^6 倍速いという報告がある[14]。しかしこれらのソフトウェアシミュレータエンジンは同期回路の検証に機能を限定して高速化を行っている。シリアル制御部等の非同期要素やアナログ要素を持つ回路の細部の正確な評価は、1ns未満のスケールを要するシミュレーションが必要となり、計算時間が膨大となる。

[†] 東京大学大学院理学系研究科情報科学専攻
Department of Information Science, Faculty of Science,
University of Tokyo

^{††} 科学技術振興事業団さきがけ研究 21「情報と知」領域
PRESTO, Japan Science and Technology Corporation

我々が高速シリアルリンクで結合したアーキテクチャをシミュレートする方法は、全てのターゲット回路をFPGAにマッピングしない。プロセッサ部のシミュレーションは汎用プロセッサモジュールを用い、ネイティブコードを実行する。シリアルリンク部はASICを用い、非同期要素をシミュレートする。これらに隣接する再構成可能素子から細粒度にタイミングを同期させることで、ターゲットの仕様に相似な環境を作成する。ターゲットの仕様に依り制御部の内部回路を変更し、パラメータを変更したシミュレーションを行う。これらのブロック全体を1枚のPCBで実装し、ターゲットの1個のチップ、またはチップ内の機能ブロック1個を表すことでボードレベルシミュレーション環境を作成する。

2章ではシミュレーション対象のアーキテクチャの性質を述べ、3章では並列計算機プロトタイプお茶の水7号の仕様を説明する。4章ではシミュレーション方法を説明し、5章ではシリアルリンク部のスケールダウンを実際に行い評価を行う。

2. Target Architecture

近い将来プロセッサ、メモリ間のレイテンシ上昇、バンド幅の不足が深刻になると言われている[6]。これらの解決策にいくつものアーキテクチャが提案されてきた。その主なものにメモリ混載プロセッサがある[12][13]。しかしこれらの多くは混載メモリを越えるワーキングセットを持つプログラムに対して、有効な解決手段を提示していない。メモリにDirect Rambus DRAMを用い、メモリ内センスアンプのキャッシュとアドレスパケットのパイプラインを利用してバンド幅を上げる提案もある[4]。他の提案に多重化された高速シリアルリンクをインタフェースを持つアーキテクチャがある[3]。memory string architecture[1]は規格が統一された高速シリアルリンクによってIO、メモリチップ、プロセッサチップを結合したシステムである。シリアルリンクは1本辺りのバンド幅を上げるため、アドレス、データのほかにフロー制御、クロック情報も多重化し、1ピン当りのバンド幅を増やし、基本データ転送単位を大きくする。このアーキテクチャに使われるメモリチップはMSAM(Multi-ported Serial-Access Memory)、プロセッサチップはMSP(Multi-ported Serial-access Processor)である。

MSAMはVRAM(Video RAM)のようにメモリを1row単位でバッファに読み込み、シフトレジスタとシリアルリンクを介してアクセスできる。VRAMと異なる点はパラレルポートを持たず、シリアルポートを複数持つことである。高速化のためにシリアルリンクからのアクセスはメモリの1row単位またはその公約数のうち大きな境界単位でしか認めない。シリアルポートを2対持つMSAMのダイアグラムを図1に示す。

MSPは、プロセッサにMSAMと同様のシリアル

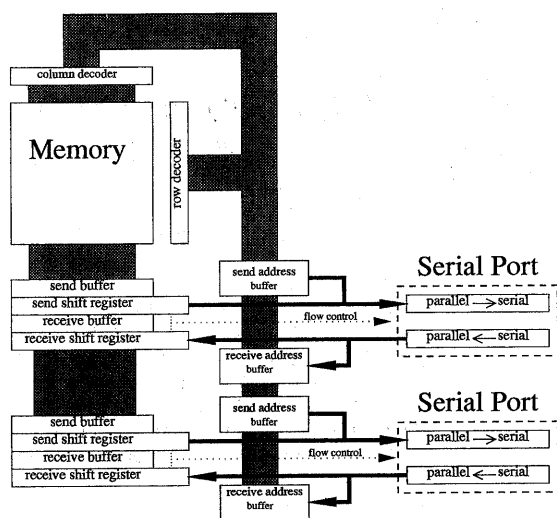


図1 MSAMのブロックダイアグラム(シリアルリンクを2対持つ場合)

リンクとメモリを混載する。プロセッサ側からチップ内メモリへのインタフェースは通常のパラレルポートがシリアル側とは別にある。MSP内に混載されたメモリは主記憶としても使えるが、主に大容量キャッシュとして用いる。

MSAM, MSP間でのトポロジーは可変である。また、送り側ノードと受け側ノードの間は直接リンクでつながっている必要はない。中間のノードがデータを中継する機能があれば、フォワーディングをし、レイテンシは増えるが隣接していないノードへアクセスが可能である。

3. お茶の水7号

我々はmemory string architectureの性能検証を行うため、並列計算機プロトタイプお茶の水7号を作成した。お茶の水7号は、MSAMまたはMSP 1chipをシミュレートするプロセッサボード、ホストマシンとのインタフェースを担当するホストボード、そしてボード間を結ぶシリアルリンクから構成される。ホストマシンとホストボードはSCSIで結ばれるが、ボード間の結合は全てシリアルリンクである。以下ではプロセッサボードの構成について述べる。

図2、図3はそれぞれプロセッサボードのブロックダイアグラムと外観である。メモリチップをシミュレートする場合は、プロセッサ部、MBus controller部は機能しない。

プロセッサボードはシリアルポートを4つ持っており、この唯一の外部インタフェースはIEEE1394[8]である。IEEE1394の物理層、リンク層のコントローラに

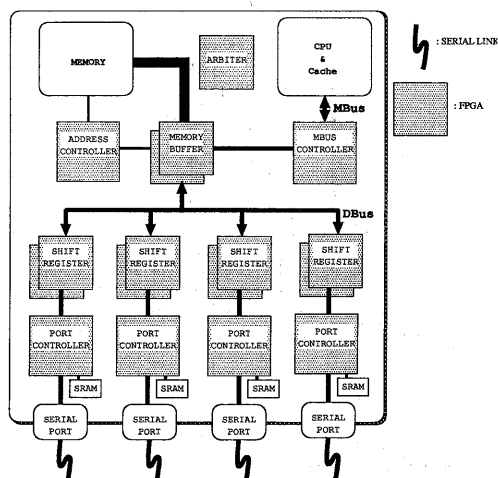


図2 プロセッサボードのブロックダイアグラム

はそれぞれ TI 社の TSB11C01 [9]、TSB12C01 [10] を使用する。プロセッサには SuperSparc Module [11] を使用する。

プロセッサモジュールとシリアルポート以外の主なブロックは SRAM ベースの FPGA(Xilinx 社の XC4000E [7] シリーズ) を使用している(図2 参照)。ピン数、ゲート数を多く必要とする **SHIFT REGISTER**、**MEMORY BUFFER** 部は XC4025E-3 を 2 chip、他はそれぞれ XC4013E-3 を 1 chip 使用している。各シリアルポートには DBus→シリアルポート用、シリアルポート→DBus 用の 2つのシフトレジスタが用意されている。ポートコントローラ (**PORT CONTROLLER**) は、シリアルポートの初期化、フロー制御、シリアルポートとの通信で用いるパケットのエンコード、デコードを行なう。各ポートコントローラには SRAM とつながっている。SRAM 内にはトポロジー情報、シリアルリンクの内部パラメータ情報といったシミュレーションで変更されやすいデータを格納する。**MBus controller** は MBus を介してプロセッサからの要求を処理する。**Arbiter** はボード内のバスの調停、システムのリセット、他の FPGA のコンフィギュレーション制御を行う。メモリブロックはデータ幅が 32 bit の SIMM が 8 枚並んでいる。これらの SIMM はデータ線は共通になっておらず、並列にアクセス可能である。

お茶の水 7 号の構成例を図 4 に示す。この例ではプロセッサボード(図中の MSP) は 1 枚で、ポート 4 がホストボードとつながっている。他のプロセッサボード(図中の **MSAM-{a-e}**) はプロセッサブロックは使わず、メモリボードとして MSAM のシミュレーションを行う。プロセッサボードがメモリボードへアクセスするとき、直接つながっていないボードへのアクセスは、

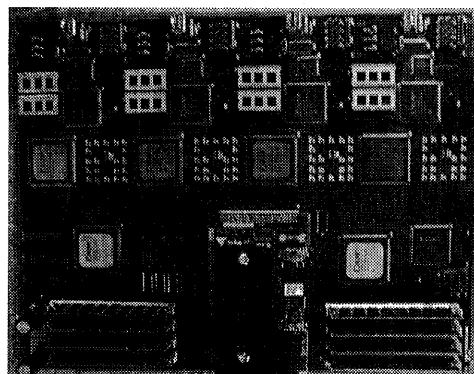


図3 プロセッサボードの外観

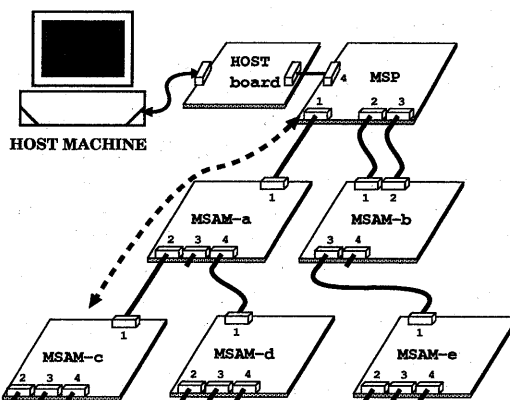


図4 お茶の水 7 号の構成例

中間のプロセッサボードが中継を行う。例えば、図 4 で MSP が **MSAM-c** へリードリクエストを出した場合、中継する **MSAM-a** のポート 1 から 2 へ、ボード内の DBus を介してフォワーディングされる。また、ボード間でリンク 1 本分より太いバンド幅を要求するときは、**MSP,MSAM-c** 間の様に複数のポートを同一ボードセット間に張ることができる。

トポロジー情報は各ポートコントローラに隣接した SRAM が保持し、これに従いパケットの分割、フォワードが行われる。例えば、図 4 中の MSP ポート 1 に接続された SRAM に、静的に **MSAM-{a,c,d}** 行きのパケットを拾うように指示されており、シミュレータ起動時にこの情報を **SHIFT REGISTER** が受取り DBus に流れるパケットの取捨選択を行う。

4. お茶の水 7号でのシミュレーション方法

4.1 トポロジーの変更

memory string architecture は、チップ間のトポロジーは可変である。バンド幅やメモリ量を変更してシミュレーションを行うために、高速シリアルリンクを容易につなぎかえることが必要である。これを本シミュレーション環境でサポートするためにシリアルリンクに IEEE1394 を採用した。この規格はリンクにつながったポートが電源投入時にネゴシエーションを行い ID、最高バンド幅を自動調整するため、トポロジーの変更に伴う設定変更が容易となる。ただし、3章で述べたように上位層でのトポロジー情報は別にシリアルポートのコントローラが保持する必要がある。トポロジー変更はポートコントローラに隣接した SRAM に反映させればよく、FPGA の回路データをトポロジーごとに用意する必要はない。

4.2 性能仕様の変更

我々のシミュレーション環境は PCB 上の実装であるため、部品のピン数、動作周波数などのハードウェア制約がある。これらの制限を越えたターゲットを機能検証する場合は、シミュレーション環境全体の性能を一律にスケールダウンし、仕様においてターゲットと相似なモデルを作成する。ハードウェア・シミュレーションではソフトウェアによるそれと異なり、現在の技術レベルでは実装できない、将来のスペックを想定したチップを性能検証できないが、我々の環境では仕様においてターゲットを一律にスケールダウンしたモデルを作成することで検証可能である。

スケールダウンの比率は、アーキテクチャ全体の各ブロックでターゲットと実機のパフォーマンス比率を出し、その中で最大値を選択する。以下ではスケールダウンの例を挙げる。簡単のため 1枚のプロセッサボードのプロセッサ部、メモリのバンド幅、シリアルリンクのみに着目する。ターゲットの仕様が

- メモリバス: 4096bit 幅、動作周波数はプロセッサボードと同一
- プロセッサ動作速度: 1GHz
- シリアルリンク: 2Gbps

と仮定する。一方、お茶の水 7号ではメモリバスが 256bit、プロセッサが 60MHz、シリアルリンクが 100Mbps である。 $\max(4096/256 \times 2 = 32^*, 1G/60M \approx 16.6, 2G/100M = 20) = 32$ の比率に合わせて全体をスケールダウンする。

FPGA 制御部は、バンド幅は単純にクロックを下げ、レイテンシはラッチを入れることで比較的単純に与えられた比率に動作をスケールダウンできる。しかし再構成可能素子でないプロセッサブロックとシリアルポ

* MSP では Column Address は必要ないのでレイテンシは半分となる。

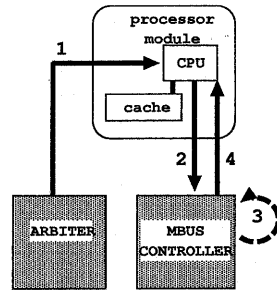


図5 プロセッサ部のスケールダウン方法

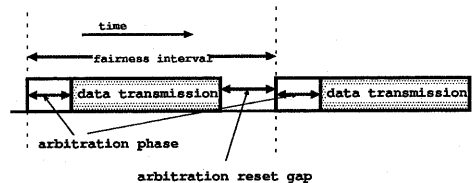


図6 IEEE1394で1ノードが連続してパケットを送り出すタイミング

トブロックにはハードウェアの制御に制限があり、一律に動作周波数を下げる等の処理ができない。そこで隣接する制御ブロックから間接的にスケールダウンを行う。

まずプロセッサ部の問題点は、動作周波数は 60MHz から変更できないことである。スケールダウンの値に合わせるために、タイマーを持つ FPGA (ARBITER) から一定時間ごとにプロセッサに割り込みをかける (図5(1))。プロセッサはこの割り込みを受けるとノンキャッチャブルな命令発行する (2)。このリクエストを MBus Controller が受けると応答時間を調整して (3) 制御をプロセッサに返す (4) ことで、プロセッサモジュール外からスケールダウンの制御が可能である。

シリアルリンクは 100Mbps で動作するが、これを 30MHz で動作する隣接した FPGA から正確に制御することは不可能である。しかし本環境で採用した IEEE1394 では上位層から一部の物理層のパラメータを変更でき、正確な制御が行える。図6は IEEE1394 のデータ転送の例である。 arbitration phase でバスを勝ち取り、 data transmission でデータを転送した後、 arbitration reset gap の間バスにリクエストを出さない***。この一連の周期を fairness interval と呼ぶ。この arbitration reset gap は外部から変更可能なパラメータであり***、目的の比率に合わせ、シ

** 本来はバス調停で starvation を起こさないために用いられる機能である

*** IEEE1394 ではバス制御層が直接物理層のパラメータを読み書きできる。PORT CONTROLLER はバス制御層も担当する

bandwidth	gap_count	1quad	gap_count	2quad	gap_count	4quad
original	1	5.0 (μs)	1	5.4 (μs)	1	6.3 (μs)
1/2	17	10.3	18	11.1	21	12.9
1/3	32	15.5	34	16.6	40	19.6
1/4	47	20.8	50	22.3	60	26.8

表 1 fairness interval

アルリンク部の正確なスケールダウンを行える。5章では実際にお茶の水7号上でリンク部のバンド幅を変更する。

5. 評 価

IEEE1394 チップのパラメータを制御するのが、図 2 の **PORT CONTROLLER** であり、FPGA XC4013E が 30MHz で動作している。

実験ではデータ部の長さが 1quad(32bit)、2quad、4quad の write パケットに対してバンド幅の変更を行う。

まず、基本バンド幅と arbitration reset gap を測定する。バンド幅から得られる fairness arbitration から、後者はパケット長*から計算した data transmission time と arbitration phase time を引いて求める。そして元のバンド幅に対して 1/2, 1/3, 1/4 になるような arbitration reset gap を求め、パラメータをチップに入れて実測してみる。ロジックアナライザで **PORT CONTROLLER** の信号を拾い、時間を計測した。結果は表 1 である。100 packet の送信を測定し、1 packet 当りの fairness interval(つまり、band 幅の逆数)の値を示している。データ長が 1, 2, 4quad のどの場合でも、バンド幅は目標の比率に調整できている。gap_count は arbitration reset gap を決定するパラメータである**。

6. おわりに

本稿では、高速シリアルリンクに基づくアーキテクチャをボードレベルで実装した。非同期要素を持つシリアルリンクブロックは ASIC で実装する。プロセッサ部は汎用モジュールで実装し、ネイティブコードを実行する。主要制御ブロックは再構成可能素子で実装する。ターゲットの仕様を一律にスケールダウンし相似なモデルをつくることで非同期要素を持つシステムのシミュレーションも高速に行える。相似の比率はシステム内の各ブロックでターゲットとシミュレーション環境の仕様の比率を求め、その最大値を用いる。高速性を要するブ

ロック、非同期要素を持つブロックは、隣接する再構成可能なブロックから細粒度でスケールダウンの比率を制御し、ターゲットのパラメータを変更したシミュレーションを行う。我々は高速シリアルリンクで結合されたシステム memory string architecture をシミュレートする環境を実装した。この環境上で隣接する FPGA からシリアルリンク部の物理層を制御し、正確な相似モデルを実装するための細かい制御を行えることを示した。

謝辞

なお、本研究は Mentor Graphics 社と Synopsys 社の University Program を用いた。両社に深く感謝する。

参 考 文 献

- 1) 松本 尚, 平木 敬 : Memory String Architecture — メモリウォールを越えて — 情報処理学会研究報告 96-ARC-120 Vol.96, No.106, pp.1-6 (1996)
- 2) William H. Mangione-Smith, Brad Hutchings, David Andrews, et al. : "Seeking Solutions in Configurable Computing" *IEEE Computer* Vol. 30, No. 12 pp. 38-43 (1997)
- 3) D.Litaize, et. al. : Multiprocessor with a Serial Multiport Memory and a Pseudo Crossbar of Serial Links Used as a Processor-Memory Switch. *ACM Computer Architecture News* Vol.17, No.6, pp.8-21 (1989)
- 4) Sung I.Hong, Sally A McKee, Maximo H. Aslinas, et al. Access Order and Effective Bandwidth for Streams on a Direct Rambus Memory *The Fifth International Symposium on High-Performance Computer Architecture* pp.80-89 (1999)
- 5) Suganuma N., Murata Y., Nakata S., Nagata S., Tomita M., and Hirano K. : Reconfigurable Machine and Its Application to Logic Diagnosis, *IEEE/ACM International Conf. on Computer-Aided Design* pp. 373-376 (1992)
- 6) Wm.A.Wulf and S.A.McKee : Hitting the Memory Wall: Implications of the Obvious. *ACM Computer Architecture News*, Vol.23,

* 例えば複数 quad の write block packet では、データ部以外に header と CRC で 6quad からなる。

** arbitration reset gap = (51 + gap_count * 32) / BASER-ATE. [8] 参照

- No.1, pp.20-24 (1995)
- 7) XILINX, INC. : The Programmable logic Data Book : (1997)
 - 8) IEEE Standards Board : IEEE Standard for a High Performance Serial Bus. P1394/Draft7.1 Vol.1 (1994)
 - 9) TEXAS INSTRUMENTS : TSB11C01 IEEE1394-1995 TRIPLE-CABLE TRANSCEIVER/ ARBITER (1996)
 - 10) TEXAS INSTRUMENTS : TSB12C01 Data Manual (1994)
 - 11) TEXAS INSTRUMENTS : SuperSPARC User's Guide. (1992)
 - 12) Murakami K., Shirakawa S. and Miyajima H. Parallel Processing RAM Chip with 256Mb DRAM and Quad Processor *IEEE International Solid-State Circuits Conference* pp. 228-229 (1997)
 - 13) David Patterson, Thomas Anderson, Neal Cardwell, Richard Fromm, Kimberly Keeton, Christoforos Kozyrakis, Randis Thomas, Katherine Yelick : A Case for Intelligent RAM: IRAM *IEEE/ACM International Symposium on Microarchitecture* pp. 34-44 (April 1997)
 - 14) J.Kumar, N.Strader, J.Freeman, and M.Miller: Emulation Verification of the Motorola 68060 *International Conference on Computer Design Proceeding* pp. 150-158 (1995)
 - 15) Jerry Bauer, Michael Bershteyn, Ian Kaplan and Paul Vyedin : A Reconfigurable Logic Machine for Fast Event-Driven Simulation *Design Automation Conference Proceedings* pp. 668-671 (June 1998)