

ハード/ソフト・コラーニングシステムにおける FPGA ボードコンピュータの設計

池田修久[†] 中村浩一郎[†] 大八木睦[†] Hoang Anh Tuan[†] 山崎 勝弘[†] 小柳 滋[†]

立命館大学大学院 理工学研究科[†]

1. はじめに

システムレベルでの最適化が望まれるシステム LSI 設計において、ハードウェアとソフトウェアのトレードオフ、つまりコデザインがシステム設計の鍵を握る。そして、大学におけるコンピュータアーキテクチャ教育の重要性は、その社会における重要性に比例して今もなお増加している。システム設計手法に様々な言語選択が存在する今だからこそ、大学教育ではハードウェアとソフトウェアの関係を深く理解すべき必要があると考える。そこで我々はコンピュータアーキテクチャにおけるマイクロプロセッサ(ハードウェア)とプログラム(ソフトウェア)の関係を学習するハード/ソフト・コラーニングシステムを構築する。本稿ではハード/ソフト・コラーニングシステムの概要とシステムの構成要素である FPGA ボードコンピュータの設計について述べる。

2. ハード/ソフト・コラーニングシステムの設計

2.1 システムの概要

コラーニングシステムとは、プロセッサアーキテクチャが選択可能な命令セットシミュレータを用いて、()プログラムと命令セットの評価、及び最適化コンパイラ設計を行うソフトウェア開発、()HDL を用いて MPU コアを設計し FPGA へ実装するハードウェア設計、を融合した教育システムである。仕様通りのソフトウェアを開発するのみならず、プロセッサアーキテクチャを理解した上でソフトウェア開発が行えるようになることが本システムの最大の目標である。本システムはアーキテクチャ学習に使用する応用プログラム開発ツール(命令セットシミュレータ、アセンブラ、コンパイラ)[1]、及び実機検証に使用する MPU コアとその周辺モジュールからなる FPGA ボードコンピュータによって構成される。

2.2 システムの特徴

プロセッサアーキテクチャが選択可能な教育用シミュレータを用いて、ソフトウェアプログラミングとプロセッサアーキテクチャを体系的に学習できることが本システムの特徴である。またプロセッサを設計し FPGA を用いてエミュレーションすることでコンピュータアーキテクチャに対する理解も深まる。

2.3 学習方法

コラーニングシステムにおける学習はソフトウェア学習とハードウェア学習から構成される。ソフトウェア学習では命令セットの評価とプロセッサアーキテクチャの

理解を目標として、教育用シミュレータを使用してアセンブリ言語によるプログラミング・デバッグ、プロセッサアーキテクチャの変更によるプログラムの比較・評価、及び最適化コンパイラ設計などを学習する。ハードウェア学習では実際に MPU コア部を設計し FPGA に実装することで、命令セットとプロセッサアーキテクチャに対する理解をより確かなものとする。図 1 にコラーニングシステムの全体像を示す。図 1 の左側が教育用シミュレータを用いたソフトウェア学習であり、右側が FPGA ボードコンピュータを設計するハードウェア学習となる。

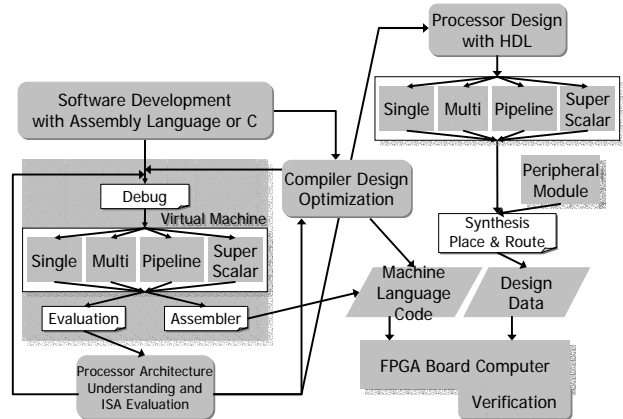


図 1: ハード/ソフト・コラーニングシステム

3. FPGA ボードコンピュータの実装

3.1 システム構成

ボードコンピュータはコラーニングシステムにおいて、命令セットとプロセッサアーキテクチャ理解の確認と実践、及びコンピュータアーキテクチャの理解を目的として使用される。FPGA 1 チップで MPU と周辺回路を実現し、メモリ以外は IP を使わず VerilogHDL による RTL 設計を行う。本システムで使用する FPGA ボードは Celoxica 社の RC100 ボードで、FPGA は Xilinx 社 Spartan2(xc2s200-5fg456)が搭載されている。ボードコンピュータのブロック図を図 2 に示す。

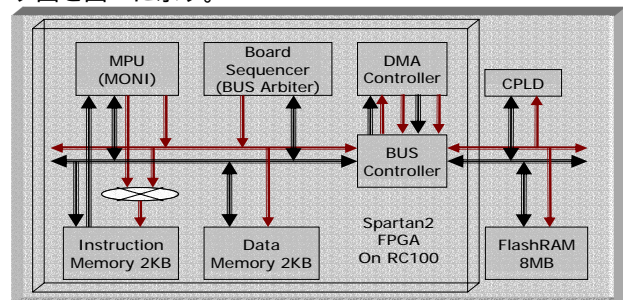


図 2: ボードコンピュータのブロック図

Design of a FPGA Board Computer in a Hardware/Software Co-Learning System

Nobuhisa Ikeda[†] Koichiro Nakamura[†] Mutsumi Oyagi[†]

Hoang Anh Tuan[†] Katsuhiko Yamazaki[†] and Shigeru Oyanagi[†]

[†]Department of Computer Science, Faculty of Science and Engineering, Ritsumeikan University

ボードコンピュータは、FPGA 内に MPU、 Board Sequencer(システムのシーケンス制御など)、 DMA Controller(DMA 転送制御)、 BUS Controller(アドレスデコーダ、バスブリッジ)、 命令メモリ・データメモリ(FPGA 内の BlockRAM)、及びボード上に FlashRAM(データ格納用)、CPLD(ホスト PC との転送制御)の 7 個のモジュールから構成される。本システムでは 24 ビットのアドレス空間を共有し、DMA 転送、MPU 演算時には図 3に示すシステムアドレス空間が使用される。

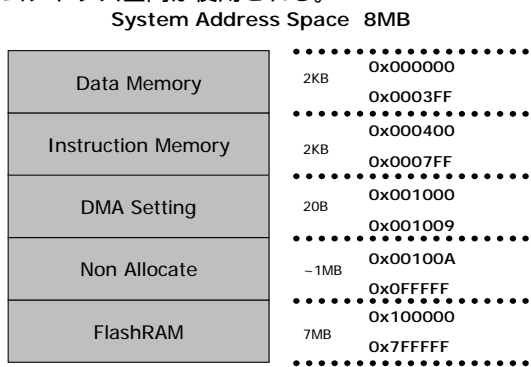


図 3：システムアドレス空間

データメモリ・命令メモリは共に 2KB の容量を持つ。DMA 転送時は 0x001000 ~ 0x001009 にメモリマップされたアドレスに適切な値を設定して DMA 転送を行う。

3.2 16 ビット命令セット MONI の定義

基本となる 16 ビット命令セット MONI は、教育システムにおけるアセンブリプログラミングのしやすさを考慮して定義した MIPS のサブセットで全 43 命令から成る[2]。MONI は 4 つの命令形式(R、I5、I8、J)を持ち、加減算、論理演算、シフト、セット命令に即値アドレッシングとレジスタ・アドレッシングの両方を備え、メモリアクセスはレジスタ間接アドレッシングを採用する。レジスタ間演算は 3 オペランド方式を採用する。

3.3 MONI 命令セットにおけるプロセッサ設計

本システムで設計するプロセッサは、異なる命令レベル並列処理における 4 種類のアーキテクチャである。単一サイクル(1 命令を 1 クロックで実行)、マルチサイクル(1 命令を 3~5 クロックで実行)、パイプライン(5 ステージ)、スーパースカラ(In-order の 3 命令同時発行)。

3.4 DMA 転送と実行プログラムの生成

DMA 転送を行う際は Board Sequencer によって DMA Controller に転送情報が設定される。この際、TDI(Transfer Data Information)と名付けた 64 ビットのレジスタを用いる。図 4に TDI のビット割り当てを示す。

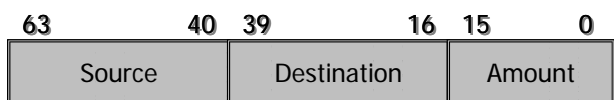


図 4：TDI ビット割り当て

本システムにおける DMA 転送は システム起動直後のデ

ータメモリ設定(FlashRAM からデータメモリ)、 システム起動直後の命令メモリ設定(FlashRAM から命令メモリ)、 MPU 演算処理後の結果の確認(データメモリから FlashRAM)、が考えられる。よって DMA 転送における最大転送量は命令メモリまたはデータメモリサイズに依存する。この 3 種類の TDI はシステム起動前に生成し、FlashRAM の所定番地に格納しておく必要がある。TDI、機械語及び変数データの生成とそれらを格納する FlashRAM の番地を図 5に示す。

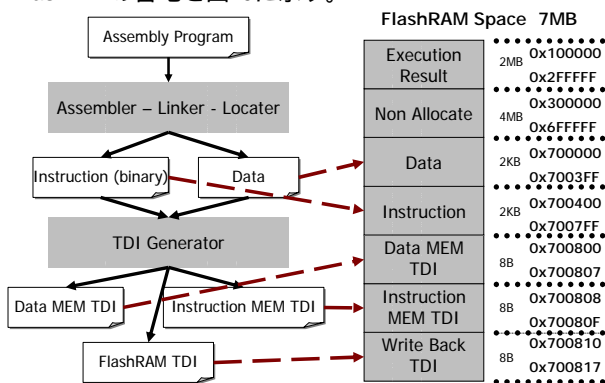


図 5：実行ファイル生成

図 5に示す様に、FlashRAM 内の 0x700000 ~ 0x70080B 番地にシステム起動に必要なファイルを格納しておく。システムを起動すると 0x100000 ~ 0x2FFFFFFF 番地に MPU による演算結果が格納され、ユーザーはホスト PC から上記の番地を読み出すことで結果の成否を確認する。

4. FPGA ボードコンピュータの性能

ボードコンピュータは Xilinx 社の FoundationISE を用いて設計を行った。論理合成ツールは XST Verilog を使用し、エリア重視で論理合成を掛けた際の FPGA リソース使用率は、49%であった(MPU は単一サイクル)。ボードコンピュータの動作周波数は FlashRAM に依存しており、ボードから供給される 80MHz のクロックを内部で 5MHz と 10MHz に分周して動作させている。

5. おわりに

ハード/ソフト・カラーリングシステムの概要と FPGA ボードコンピュータの設計について述べた。現在までに単一サイクル、マルチサイクル、及びパイプラインの MPU コアでの実装が完了した。今後は教育用シミュレータとともにシステムの完成を目指す。そして将来的には命令セットの定義と命令セットシミュレータを自動生成し、教育システムに留まらずプロセッサ合成における空間探索として使用可能なシステムの構築を目指す。

参考文献

[1]大八木睦他:ハード/ソフト・カラーリングシステムにおけるアーキテクチャ選択可能なプロセッサシミュレータの設計,情報処理学会第 66 回全国大会論文集,2004.
 [2]John L. Hennessy, David A. Patterson 著, 成田光彰 訳:コンピュータの構成と設計(上)(下),日経 BP 社,1999.
 [3]三木良雄:システム LSI 設計特論第 1 章講義資料,立命館大学大学院スターク寄附講座,2003.
 [4]今井正春:高位言語ベースデザイン設計特論第 4 章講義資料,立命館大学大学院スターク寄附講座,2003.