

仮想 CPU 作成ツールの開発

秋和潤^{†1} 佐々木純一^{†2} 原美智子^{†3} 西田誠幸^{†4} 原田紀夫^{†5}
 拓殖大学工学部情報工学科^{††}

1. はじめに

計算機アーキテクチャの教育現場において、CPUの構成と動作は最も重要な学習項目の一つである。その導入教育支援の一つとしてCity-1[2]等のCPUの設計演習を通して学習を行う方法がある。これは、ハードウェア記述言語を用いて作成された機能記述をFPGAに送り込みボード上に独自のCPUを完成させるというハードウェア演習である。この演習で、学生は、CPUの構成と動作について学習することができる。

さらに、これらの演習をソフトウェア上で行うことができれば、使用者は、参考書等に取り挙げられたCPUを手軽に再現し、動作確認することができる。また、演習に要する時間だけでなく、コストの節約にも繋がる。

そこで、本稿ではGUIを用いた仮想CPU作成ツールの開発について述べる。これは使用者自身の意図する機能を持った独自のCPUを作成することで計算機アーキテクチャの学習を支援するツールである。

またこのツールを用いて作成されたCPUは、当研究室で開発が進められている専用のシミュレータソフトによって動作確認することができる。

2. 従来のCPU学習環境

従来のCPU学習環境で、設計を通して行うハードウェア演習では、HDL等のハードウェア記述言語を用いて作成した機能記述をFPGAに送り込みボード上にCPUを完成させるというスタイルである。演習の初期段階でハードウェア記述言語を習得することが必須となるが、これはCPUの学習とは直接関係ないもので学生にとっ

ては大きな負担となる。

また、あらかじめ構成の定められたCPUでその構成と動作について学ぶことができる教材として手動CPU[1]がある。

手動CPUは、ボタンやスイッチを操作することで制御信号を送り、データの流れや命令実行の一連の動作やCPUを構成する各回路の働きを学習することができる教材である。これは一昨年、当研究室で開発された。

3. 本ソフトウェアの特徴

本ソフトウェアは、使用者に対し従来のようなハードウェア記述言語等の予備知識を必要とせず、CPUの手軽な学習環境を提供することができ、かつ能動的な学習を促すことのできるツールである。また使用者が種々のCPUの構成を機能回路単位で作成可能なため直感的な理解を助けることができる。ただしコストについての評価はこのツールでは行えない。

4. 本ソフトウェアで作成できるCPU

4.1 命令セットアーキテクチャ

作成可能なCPUは、演算命令に用いられる明示的オペランド数が1つのアキュムレータアーキテクチャ、もしくは2つの汎用レジスタアーキテクチャである。また明示的オペランドはメモリまたはレジスタを直接指定するものとし、メモリ参照は直接アドレッシング方式のみ可能である。命令セットで用意される演算操作は使用者が候補の中から選択的に採用できるようになっている。語の構成は最大16ビットまでとし、その範囲内において自由に設定可能である。

4.2 マイクロアーキテクチャ

機能回路については、プログラムカウンタ、インストラクションレジスタ、メモリ、ALU、レジスタ、マルチプレクサの6種類を用意している。各機能回路はデータパスで接続可能であり、それぞれのデータパスにはビット幅を設定できる。さらに命令取り出しや定義済み命令セット、それ

Design and Implementation of a Development Environment for CPU Specification with Datapath Elements as the Unit

^{†1}Jun AKIWA (jakiwa@cs.takushoku-u.ac.jp)

^{†2}Junichi SASAKI (j-sasaki@cs.takushoku-u.ac.jp)

^{†3}Michiko HARA (mhara@cs.takushoku-u.ac.jp)

^{†4}Seikoh NISHITA (snishita@cs.takushoku-u.ac.jp)

^{†5}Norio HARADA (nharada@cs.takushoku-u.ac.jp)

^{††}Dept. of Computer Science, Faculty of Engineering, Takushoku University

それぞれに対応した CPU 機能回路の制御信号を定義することができる。

5. CPU 作成支援ソフトの構成

5.1 本ソフトが提供する CPU 作成手順

本ソフトによる作成は以下の手順で行う。

命令語の仕様を決定 本ソフトウェアではまず、明示的オペランド数を決定させる。さらに命令語の長さとしてデータ長をそれぞれ最大 16 ビットまでの範囲において選択させる。内部ビット構成は、使用者自ら決めた命令語の長さの範囲内で、命令部、アドレス部をそれぞれ決定させる。これらは全ての命令セットに対し固定のものとなる。

命令セットを定義 次に、オペランドの種類をいくつか提示し、使用者に選択させることで命令語を定義する。この際、命令毎に機械語とモニタリングを自由に設定させる。

CPU の構成を決定 さらに、使用可能な CPU の機能回路を提示し、使用者に選択、配置させる。また、各回路間の接続関係とデータパスの名称、転送幅について、使用者が自由に設定できる環境を提供する。機能回路として、プログラムカウンタ、インストラクションレジスタ、メモリを各 1 つと、レジスタ、マルチプレクサ、ALU の使用を認める。また、ALU の機能は実際の CPU で用いられる演算を全て用意し、使用者がその中から選択し利用できるようにする。レジスタは保持するデータのフラグ機能を設定できるようにする。フラグには、0 フラグと負フラグの 2 種類を用意する。

制御部を設計 最後に使用者に、自ら設計した CPU の命令取り出しと、その他の命令の実行時における各機能回路毎のタイムチャートを作成させる。CPU は、その情報を制御部とみなし動作するものとする。

5.2 シミュレータ側へ出力するデータ

出力するデータは主に、データ長、命令語の仕様と命令セットの情報、使用した各機能回路の配置情報と配線情報と制御信号の情報である。

6. シミュレータソフトについて

シミュレータソフトは、本ソフトウェアによって作成された CPU を動作確認するためのもので

ある。これは、使用者が作成した CPU 仕様記述を読み込むことで動作するが、本ソフトウェアの出力データを利用することも可能である。

一般的なシミュレータソフトと異なる点は、設計の段階で間違いがあった場合にも、可能な限り動作させるという点である。つまり、作成ツールで CPU を作成後、シミュレータソフトで動作確認しシミュレータソフトが動作不能になった時点で作成ツール側に戻って誤りを訂正するという、主体的な学習を行わせることができる。

7. 本ソフトウェアの実装

本ソフトウェアは、Java1.4 を用いて実装した。実行画面を図 1 に示す。

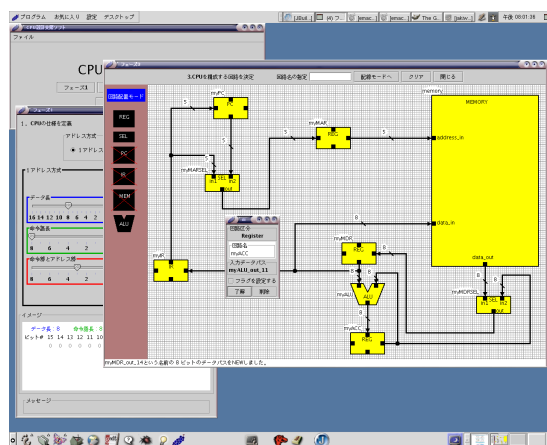


図 1: 仮想 CPU 作成ツールの実行の様子

8. おわりに

本稿では、CPU の構成と動作についての初心者への学習支援を目的とした仮想 CPU 作成ツールについて述べた。

本ソフトウェアの特徴として、CPU 機能回路を単位としたマイクロアーキテクチャの指定、タイムチャートによる制御回路の指定を行うことができる。

参考文献

- [1] 佐々木純一，西田誠幸，原田紀夫：“CPU の構成と動作を学習するための実験教材の作成” 情報処理学会第 65 回全国大会講演論文集 (2003-3)。
- [2] 高橋隆一，児島彰，上土井陽子，吉田典可：“マイクロコンピュータ設計教育環境 City-1”