

## FPGA を利用した計算機システムの学習環境の実現

2R-04

下川智士† 早川栄一‡ 高橋延匡‡  
 † 拓殖大学大学院工学研究科 ‡ 拓殖大学工学部

## 1. はじめに

情報工学において、ハードウェアとソフトウェアを分けて考える傾向がある。ハードとソフトを区別せず、一つの計算機システムとして理解ができる学習環境があれば、一貫した計算機システムの理解に役立つはずである。

本研究室では、OS、コンパイラ、ハードウェアアーキテクチャに着目したシステムソフトウェア教育支援環境の開発プロジェクトの下、現在は OS の教育支援システムの開発[1]を行っている。本研究の位置付けとしては、システムソフトウェアと CPU アーキテクチャの関係を学習するものである。本システムを利用することで、計算機システムにおけるプロセス動作理解の補助ができる。

## 2. 設計方針

## (1) FPGA の利用

FPGA を利用することで、プロセッサを短時間で作成、動作確認を繰り返し行うことができる。また、制御などの実時間性を持つものについても実機環境で学習を可能にする。

## (2) FPGA・PC 間で通信し可視化する

FPGA で作った CPU の動作情報を外部マシンへ送る。それを可視化することで、ユーザに見やすく、動作学習がしやすい環境を提供する。

## (3) ハードウェアモニタの組み込み

FPGA 上で CPU を作成する際、処理を外部に出力する回路を組み込む。これによって CPU が実際に行った動作を外部でトレースすることができる。

## (4) CPU 動作学習環境の提供

FPGA から得た CPU 動作データを分かりやすく表現し、学習者に動作の理解をさせる。手段として回路の可視化とプログラムのトレース環境を提供する。

Realization of a environment using FPGA for learning computer system

Satoshi Shimokawa †, Eiichi Hayakawa and Nobumasa Takahashi ‡

† Graduate school of Engineering, Takushoku University

‡ Faculty of Engineering, Takushoku University

## 3. 全体構成

本システムの構成図を図 1 に示す。ここでは図 1 の①～⑤について述べる。

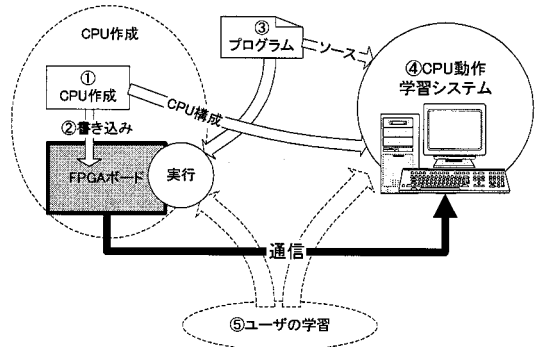


図 1 構成図

## ① CPU 作成

HDL で記述したプログラムを論理合成し、FPGA に書き込むことで CPU を作成する。HDL には Verilog-HDL を利用する。

## ② 書き込み

作成した HDL を論理合成し FPGA に書き込む。このとき CPU の機能と共に、外部通信回路を加えることで FPGA 上に実現された CPU が外部と通信できるようになる。

## ③ プログラム

作成した CPU 上でプログラムを実行させる。また、このプログラムを CPU 動作学習システムにも送る。

## ④ CPU 動作学習システム

FPGA の動作に連動して動くシステム。ユーザは主にこのシステムにより CPU 動作を学習する。

## ⑤ ユーザの学習

ユーザは FPGA 上と外部マシンの CPU 動作学習システムにより、動作の学習を行う。

## 4. 設計

## 4.1 通信部分

FPGA ボードから USB を使って計算機と通信を行う。通信で扱うデータを次に述べる。

- ・ プログラムカウンタ (PC)
- ・ インストラクションレジスタ (IR)
- ・ ストラレジスタ (SR)
- ・ フラグレジスタ (FR)
- ・ 汎用レジスタ (GR)

#### 4. 2 動作学習

本システムでは①実測モードと②ステップ実行モードの2モードの学習環境を提供する。

##### ① 実速モード

FPGA上でプログラムを実行するとき、LEDを点灯させることでCPUの動作速度を体感させる。複数のLEDをそれぞれ2進数の1桁に割り当て、プログラムカウンタの動作ごとにインクリメントする。

##### ② ステップ実行モード

FPGAと外部PCを通信させ、ステップごとにFPGAから外部PCにデータを送る。外部PCは受け取ったデータを基に動作を表示させる。

表示させるデータを次に述べる。

- ・ 回路図 (あらかじめ外部マシンに渡す)
- ・ レジスタ
- ・ メモリ
- ・ プログラム (あらかじめ外部マシンに渡す)

また、本システムではFPGAの動作ログファイル情報をPC上へダウンロードできる。このデータを処理することで、命令頻度、パターンの算出が可能になり、学習の際にundo機能を実現することができる。実行画面を図2に示す。

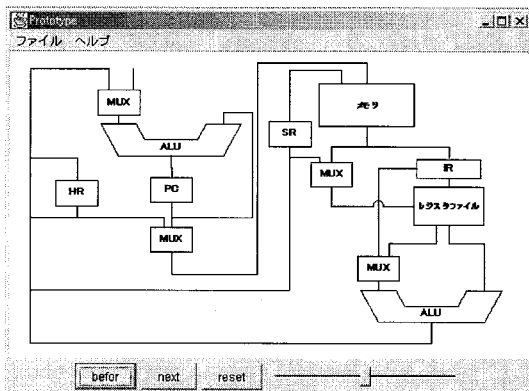


図2 ステップ実行モード画面

図の before、next、reset ボタンがそれぞれステップ実行の前、後、クリアに対応している。

データの受け渡し部分はバスの色を変えることで表現し、それぞれのレジスタの中身を数値や言葉で

表現する。表1にレジスタの可視化の内容を示す。

表1 レジスタ表示

要素	表示法
PC	数値
IR	具体的な命令
SR	数値
FR	状態 (正、負、零)
GR	数値

プログラムとメモリは別ウィンドウで表示を行う。また、メモリの内容は規模が大きいため、レジスタの通信回路とは別のルートからデータを取得し、ステップ実行を行っているときだけFPGAと連動して動作する。

#### 4. 3 ソフトウェアの実行環境

本システムは「HDLによるVLSI設計」[2]を参考にCPUを改良、作成したものであり、作成したCPUでソフトウェアを実行させるためにアセンブラを作成し、FPGA上に組み込んである。現在アセンブリ言語のプログラムが実行可能である。

#### 5. コースウェア

本学習環境では作成したFPGA上のCPUでソフトウェアを実行させる。ユーザはFPGA上のLEDでCPU速度を目で確認することができる。次に外部マシン上で命令とCPUの動作関係や処理の流れを学習することができる。これにより、ハードウェア・ソフトウェア間の協調動作の理解をさせることを想定している。

#### 6. おわりに

本原稿では、FPGAを利用した計算機システムの学習環境の実現について述べた。現在ソフトの実行環境にはアセンブリ言語にしか対応していないので、コンパイラをどうするか考えている。今後は、他のシステムソフトウェア学習環境と連動し、OSなどの動作も学習できるようにしていく予定である。

#### 参考文献

- [1] 西野洋介、早川栄一、高橋延匡：「OS概念の教育支援システムの開発と評価」第63回情報処理学会全国大会 3T-1 (2001)
- [2] 「HDLによるVLSI設計」深山正幸・北川章夫・秋田純一・鈴木政國／著 共立出版株式会社 (1999)