

マイクロコンピュータ設計教育環境 City-1

～ FPGA コンピュータの自由な設計と製作～

高橋 隆一, 児島 彰, 上土井 陽子, 吉田 典可

広島市立大学 情報科学部 情報工学科

広島市立大学情報科学部情報工学科においては、3年前期の「情報工学実験III」において、1人1台、小規模な8ビットコンピュータの自由な設計を行なわせている。この仕様は、さらに、2人1組になって製作する、FPGAとTTLを用いたスーパーセットのボード上に実現される。HDLによる記述を行ない、シミュレーション/シンセシス、さらに配置/配線を行なうトップダウンの設計である。必要最小限の周辺回路を指導するため、初年度は、どの組も2週間～1カ月程度でさほどの困難もなく製作を済ませ、履修した53人のうちの50人が、割算ができるという意味で、完全なコンピュータを完成させた。約半数の学生はクイックソートまでができるレベルの命令セットを実現し、3人は2～4段のパイプラインRISCプロセッサを稼働させた。

Microcomputer Design Educational Environment City-1

～ Unrestricted FPGA-computer Design and Fabrication ～

Ryuichi TAKAHASHI, Akira KOJIMA, Yoko KAMIDOI, Noriyoshi YOSHIDA

Faculty of Information Sciences
Hiroshima City Univ.

151-5 Ozuka, Numata-cho, Asaminami-ku
Hiroshima City, 731-31 Japan

In Hiroshima City Univ., every junior student designs his own small 8 bit computer. Every pair of students fabricates a "superset" board for their designs using FPGAs and TTLs. In the first phase of this course, students design appropriate instruction sets and micro-architectures according to the requirements that are given only by showing some application programs that should be run on their machines. These specifications are described in a HDL which are expected to be synthesized into gate level specifications for automatic placement and routing. Since a minimum set of I/Os is introduced, the fabrication takes only about from two weeks up to a month. In the case of 1996, 50 students, out of 53 students, succeeded to complete the entire phases. About half of them succeeded to run quick sort programs on their own machines. Three of the machines were pipelined RISC with interlocks that detect the hazards.

1 はじめに

数万～数十万ゲート規模が日常化した今日では、自動配置／配線はもちろん、ゲートレベルの仕様も、ハードウェア記述言語 (HDL) で記述したレジスタトランスファレベルの仕様から自動生成することが普通になってきている。従って、大学においても、回路図の入力から出発するスキマティックエントリではなく、レジスタトランスファレベルの仕様からロジックシンセシスによってネットリストを自動生成し、さらに自動配置配線を行なわせる授業が強く望まれる。実現には、数万ゲート規模が比較的安価に市販されている FPGA を用いることができる。

学部3年の前期は、専門教育の履修が集中している時期で、情報工学科の学生にとって、あらゆる部分を自分で設計したコンピュータをはじめ作るには非常によい時期だと考えられる。「自分のコンピュータを」という実感を味わうためには、アドレス空間の設計を含む、命令セットを自由に定めることを許し、「作った」という実感を味わうためには、FPGA や TTL を実装済みのボードを使うのではなく、ひとつひとつの部品を組み上げて、実際に製作させることが重要だと筆者らは考えた。

広島市立大学の情報科学部情報工学科においては、3年前期の「情報工学実験 III」において、1人1台、小規模な8ビットコンピュータを自由に設計し、さらに、2人1組になって、FPGA と TTL を用いたスーパーセットのボードを製作して実現する実験を行なっている。周辺回路は必要最小限を指導する。

初年度の平成8年度の場合、履修した53人のうちの50人が、割算ができるという意味で、完全なコンピュータを完成させた。製作の期間は1カ月以内であった。約半数の学生はクイックソートを走らせるまでの達成度を示し、3人は2～4段のパイプライン RISC プロセッサを稼働させた。

本稿では、RISC を、(1) 固定フォーマットの命令セットをもち、(2) 演算のオペランドとしてメモリ上のデータを指定することがなく、(3) 1クロックあたり1命令のスループットを有するコンピュータの意味に用いる。

2 授業計画

本設計教育環境 City-1 は次のような点を特徴としている：

(i) 「クイックソートなどの適当なアプリケーションが実行可能であること」という以上の制約は与えず、ひとりひとりに、命令セットアーキテクチャやアドレス空間を自由に設計させていること。

(ii) レジスタトランスファレベルでの HDL による記述を出発点として、ロジックシンセサイザ、自動配置配線を用いるトップダウン設計であること。

(iii) HDL によるマイクロアーキテクチャの設計／記述に際しては、モジュールリティ (モジュール分割の大切さ) を明確に意識させていること。

(iv) 制御信号の発生のための符号化に際して、結線論理制御方式とマイクロプログラム制御方式の区別をさせていないこと。

(v) 必要最小限の周辺回路について指導することで学生の負担を減らし、自ら製作までを行なわせることで「自分で作った」という実感を抱かせていること。

入出力には、トグルスイッチと LED を用いている。FPGA に設計データを書き込めば自立して動くことができる。

以下では履修の背景や目的、授業の進め方や環境などについて詳述する。「モジュールリティを明確に意識させていること」、「制御信号の発生のための符号化」については3で詳述する。

2.1 履修の背景

広島市立大学の情報科学部情報工学科においては、情報工学実験 III に直接関連のある講義として、表1に示すような科目を事前に、あるいは並行して履修する。

「論理数学」と「論理回路」によって、論理回路設計の基本的な知識はすでに有していることを前提にできる [7],[6]。「情報工学実験 I」では、既成のマイコンによってアセンブラを学び、カウンタ程度の簡単な回路の設計は経験済みである。「情報工学実験 II」の内容には、スタックマシンを前提に、yacc と lex を

表 1: 情報工学実験 III に関連する講義

1 年後期	論理数学, 離散数学
2 年前期	論理回路, オートマトンと形式言語, データ構造とアルゴリズム, 情報工学実験 I
2 年後期	コンパイラ, コンピュータシステム I, 情報工学実験 II
3 年前期	論理シミュレーション, オペレーティングシステム, コンピュータシステム II

用いてコンパイラを製作することが含まれる。情報工学実験 III では、0～4 アドレス方式について説明して、「スタック」は 0 アドレス方式とみなせる [3] ことを説明している。yacc と lex の使用経験があるため、余裕のある学生には、「オートマトンと形式言語」などで学んだ有限状態機械やプッシュダウンオートマトンの能力を思い出させ、「自分のコンピュータ用のアセンブラを作ってみてははどうか」と助言している。

並行して履修する講義のうち、特に「論理シミュレーション」では、情報工学実験 III で使用するシミュレータやロジックシンセサイザを具体例として、CAD を用いたデジタルシステムの構築技術を指導している。

2.2 履修の目的

コンピュータを設計し、製作し、課題として与えられたプログラムを、自分が定めた機械語で書き、自分のコンピュータの上で走らせることによって、「コンピュータの動作原理と設計方法の基本を学ぶ」ことが目的である。

結果的に、HDL による記述や CAD ツールの使い方に習熟するが、そのこと自体は目的ではない。

2.3 授業の進め方

情報工学実験 III のシラバスはトップダウンの設計／製造過程を忠実に反映したものになっている：

- [1] システム分析 (商品企画)
- [2] 方式設計
- [3] 命令セットアーキテクチャ
- [4] 機能設計／モジュール分割
- [5] データバスの設計
- [6] データバスの検証

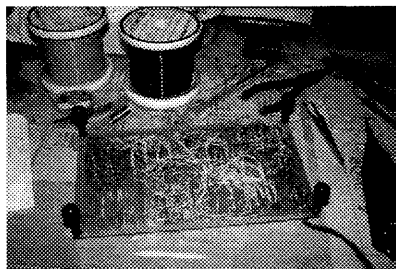


図 1: ラッピングによる製作の一例

- [7] 制御部の設計
- [8] 制御部の検証
- [9] コンピュータボードの製作
- [10] コンピュータボードの動作確認
- [11] FPGA の実装
- [12] システムの動作確認
- [13] システムのデバッグ
- [14] システムのテスト (出荷検査)
- [15] 終了判定

期間は 15 週、週 1 日、3 時限 (4.5 時間)

だが、毎週、別にもう 1 日、3 時限の予備日が設けられているほか、設計／製作環境の空き時間にも自由な履修を許している。

配線にはラッピングを用いている (図 1)。バスを含む、殆んどの配線を自由に行なわせていることは、アドレス空間を自由に定め、メモリバンド幅を工夫することに役立っていると考えられる。

並行して行なわれる「論理シミュレーション」の講義では、CAD を用いて、フリップフロップ、加算器や ALU、カウンタの設計方法を順を追って指導している。

2.4 進め方の工夫

情報工学実験 III は、学生たちにとっては、自分のコンピュータを設計し、製作するはじめての経験であり、将来に大きな影響を与える。このため、まずはできるだけ多くの学生が自分のコンピュータを完成させることを重視した。

学生の能力には差がある。著しく進まない学生を作らないよう、提出物の期限は、順を追って、小刻みに指定するよう配慮した。初年度履修した 53 人のうちの 50 人が、割算ができるという意味で、完全なコンピュータを稼働させたという実績の背景のひとつはここにある。

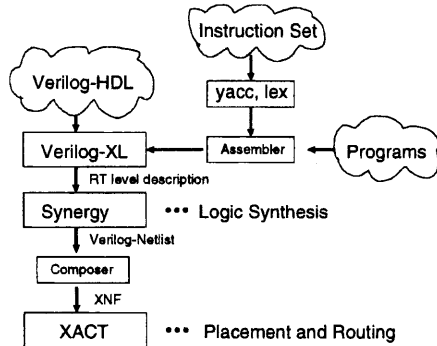


図 2：トップダウン設計の流れ

2.5 CAD 環境

HDL としては Verilog-HDL を用いている。Verilog-HDL を選んだ理由は次の通りである：

- (1) 18 の半導体ベンダーから 188 のライブラリが供給されていた。
- (2) 1995 年の時点で 2 万品種という、実績があった。
- (3) 事実上の業界標準であり、IEEE も IEEE1364 として標準に指定している。
- (4) 基本的に信号線 (wire) と記憶素子 (reg) の 2 種類の変数しかなく、入出力が input, output, inout の 3 種類しかないにもかかわらず広範な記述能力を有する。
- (5) C 言語に似ていて親しみ易い。

現在では、49 の半導体ベンダーから 400 ~ 500 のライブラリが供給されている。

Verilog-HDL に対しては、高速な論理シミュレータが多く供給されている。この実験で用いたのは Verilog-XL である。ロジックシンセサイザは Synergy を用いた。これらは HDL desktop というランチャーから起動して用いた。

Synergy が生成する Verilog-HDL のネットリストは Composer というツールを介して XACT という自動配置配線プログラムに送った。ターゲットは Xilinx 社の XC4000 シリーズとした。

図 2 に全体の流れを示す。XNF は Xilinx 社のネットリストである。アセンブラを開発するのは余裕のある学生に限られる。

2.6 計算機環境

上述した CAD ツールは学部環境と学科環境に分散させて導入している (図 3)。ライセンス数は上流工程のためのそれである。

情報科学部には 4 学科共有の資産として Sun

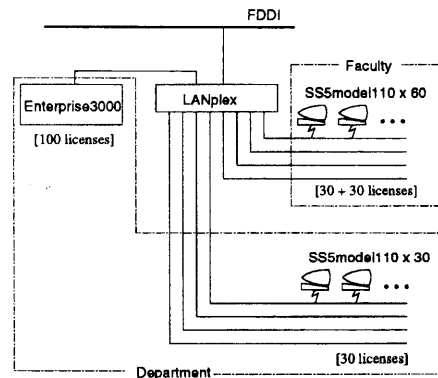


図 3：ネットワーク構成

WS が学生用に 60 台、教官用に 1 台 (SS20) 導入されており、学生用 WS の 2 台がそれぞれ 30 ずつのライセンスを管理している。OS は Sun OS 4.1.3.U1 である。ホームディレクトリは FDDI 上の 2 台のファイルサーバーに置かれている。

情報工学科は独自の資産として Sun WS が学生用に 30 台、教官用に 1 台を導入しており、教官用の WS が 30 ライセンスを管理している。教官用のマシン (SS20) の OS は Sun OS 4.1.3.U1 である。教官用マシンは別に、自動配置配線のための XACT の 30 ライセンスを管理している。学生用 WS の OS は Solaris2.5.1 である。情報工学科では、学部の 2 台のファイルサーバーとは別に、Sun の Enterprise3000 を導入しており、上流工程のため、その他研究用のツールのための 100 ライセンスを管理している。OS は Solaris2.5.1 である。

環境を分散させていることは、ひとつには危険分散、さらには CAD ツール更新のための便宜に貢献している。

3 指導内容の詳細

ここでは教育環境 City-1 の特徴である「モジュラリティの認識」と「制御信号の発生のための符号化の方法」を含む、指導内容の詳細について述べる。

3.1 シンセシスを前提とする記述

Verilog-HDL によって、合成を考慮して組合せ論理回路を記述する方法は基本的には 3 通

りある：(1) and, or, nand のようなプリミティブを用いた構造記述 (2) 継続的代入文 (assign 文を使う組合せ論理回路の記述) (3) 入力をすべて列挙した手続き的な記述。

これらは、学生が並行して履修する「論理シミュレーション」の中で指導している。「論理シミュレーション」の講義で示した記述例は、合成できない例を示すときを除いて、すべて合成可能な記述であるよう配慮した。記憶素子として宣言されている、組合せ論理回路で実現される場合があることは、「情報工学実験 III」で、ALUなどを合成しようとする過程を通して経験的にも学習させた。

3.2 モジュール性の認識

現状では、ひとりでプロセッサ全体を設計することはなく、適当な分業で協力して設計を行なうのが普通である。モジュールに分割して設計することは階層的な設計という観点からも重要である。

「情報工学実験 III」では、学生がデータパスの設計を学んだ時点で、レジスタユニット、演算ユニット、命令ユニット、制御ユニットのようなモジュール性を有し、合成はできるが、まだ全く不完全な記述を与えて、「これをもとに記述を完成させよ」という指示を与えている。ある程度完結した記述を与えてしまうことは、2年目以降、上級生が記述を提供してしまうことを防ぐことにも役立つと考えている。シフトなどは function 文を用いて記述できることなども指導している。

パイプライン化を行なおうとする学生には、「計算の本質は命令語間のデータ依存関係にある」[10]ことを指導している。「実行順序関係を崩すことにはハザードと呼ばれる危険を伴い、これを検出して回避する機構が(パイプライン)インターロックである」[1], [10]とまで説明すると、指導を受けた学生は、特に指示をしなかったにもかかわらず、「インターロックユニット」の記述を始めた。

3.3 制御信号の発生

データバスに送る制御信号の符号化に際して、結線論理制御方式とマイクロプログラム制

御方式の間に本質的な差はなく、これらは一括して扱うことができる [8],[9]。

情報工学実験 III では、結線論理制御方式の制御部の設計に、従来、水平型のマイクロプログラム制御方式に用いられてきた符号化をまず指導した。以下に記述例を示す：

```
reg [39:0] STATE;
// STATES
'define I_FETCH_1 40'b0100 *** 1000010010
'define I_FETCH_2 40'b0100 *** 1000010001
'define LOADREGO 40'b0100 *** 0100000000
    :
assign {FUNC_CTL_A,IR_LEFT,IR_RIGHT}=STATE[2:0]
    :
    {MEM_WRITE,MEM_READ}=STATE[39:38];
always @(posedge CLK1)
begin : state_machine
case (STATE)
'I_FETCH_1: STATE <= 'I_FETCH_2;
    begin
case (DATA_BUS)
'I_FETCH_2: .
```

Moore 型を指導すると「分かりやすい」と好評であったが、状態遷移関数を実現するための回路が大きくなるなどの問題が起こるため、Mealy 型の指導も自然に行なえた。

パイプライン化を行なおうとする学生には、制御部を分散するよう指導した。全ユニットがとりうる状態のすべてを把握する、いわゆる「タイムステーションナリー」な制御を行なうことはない判断したためである。

3.4 CISC か RISC か

学生に渡す、全体の、不完全な記述は、2 アドレス方式の CISC である。

1990 年代の前半は WS 市場において RISC が優位に立ったといえる。しかし、Intel 社の PentiumPro は 86 アーキテクチャでありながら 366 SPECint92 であり、R10000, PowerPC620, Alpha21164, PA-8000 のどれにも勝っている。

文献 [11] には CISC か RISC かについて興味深い議論が見られる。この文献では、CAD の視点からの議論には特に結論をあたえていないが、上述したような制御部の記述を行なった場合、結線論理制御方式とマイクロプログラム制御方式の間には、記述の負担という点では差がないといえる。メモリはクロック信号を使っていないという点からも、組合せ論理回路

だとみなすことができ、結線論理制御方式とマイクロプログラム制御方式の間に、本質的な差はない。しかし、歴史的には常に、CISCの世界においても、結線論理制御方式が勝ってきた。

City-1が指導している符号化の方法は、CISCに対し、結線論理制御方式を広範に使用する道を開いているといえる。

特に Moore 型の場合は、マイクロプログラムであるかのような順序回路を記述すると、状態遷移関数のための回路が大きな負担となることがある。しかしこのことは、状態遷移先を default で指定することで回避できることを学生が見出した。

4 履修の状況

4.1 命令セットアーキテクチャ

学生に渡した、全体の不完全な記述は、2 アドレス方式の CISC であったため、殆んどどの学生がこれを踏襲していた。しかしレジスタ構成にはバラエティーが見られた。

割算のプログラムのステップ数は 9 ~ 68 命令と様々であった。引き算で割算を行なっている学生がいる一方で、正負を扱い、シフトによって割算を行なう学生がいたためである。クイックソートのためのステップ数は 61 ~ 110 命令であった。RISC を設計した 3 人のうちひとり、CISC を設計した 50 人のうちの 2 人の計 3 人の学生が、前記の簡単な指導だけで、自分のコンピュータのためのアセンブラを開発して利用していた。

1 バイト命令と 2 バイト命令を混在させていた学生の多くが、レポートに、「固定フォーマットにすればもっと性能がでると思う」と書いていたことは特筆に値する。これに「主要な演算をレジスタ上で行なえば」という考え方を加え、高速化を考えれば RISC の発想になる。

4.2 メモリアーキテクチャ

殆んどどの学生が 8 ビットのアドレス空間を用いており、9 ビットを用いた学生は数名、2 ビットのレジスタと接続するなどして 2k バイトを確保した学生は 1 人だった。

パイプライン化を行なったうちの 2 人とパイプライン化をしていない 1 人の、計 3 人が、データとプログラムで個別のバスを備えた、ハーバードアーキテクチャを採用していた。

4.3 回路規模と性能

表 2 にロジックシンセサイザの Synergy が出力した回路規模と動作可能クロック周波数 (MHz) の平均を示す。左の「GRm」, 「SPn」は「汎用レジスタ数が m」, 「スタックポイント数が n」であることを表している。ビット数はアドレス空間を表す。セル数は、and, xor のような基本的な素子の数である。

表 2: 回路規模と動作可能周波数

レジスタ構成	人数	セル数	周波数 (MHz)
GR4	11	1062	42.9
GR4,SP1	19	1145	52.5
GR4,SP2	10	1398	46.5
GR6	1	1743	50.0
GR8	3	1089	56.7
GR4,9bit	1	1335	52.6
GR8,9bit	2	1234	42.3
GR4,11bit Bonjour	1	1111	69.6
GR10, 9bit T^2	1	1708	34.4
GR8, 10bit SCEPTER	1	1717	27.0

4.4 パイプライン化

もとの仕様が複数のクロックサイクルを前提に考えられていたならば、1 命令を 1 クロックで実行できるようにするパイプライン化は容易で、数倍の性能向上が期待できる。しかし、1 命令を 1 クロックで実行できている場合には、パイプラインの段数を増やすことはクロック周波数を高めることが目的になる。

表 2 の「Bonjour」, 「 T^2 」, 「SCEPTER」は 2,3,4 段のパイプライン化を行なった RISC であった。これらは、この順に、ステップ数が少ないという意味で、命令セットが強くなっていた。割算のプログラムの場合で、「SCEPTER」の所要ステップ数に対し、「 T^2 」と「Bonjour」は、同等な処理を行なうために、1.57 倍、1.86 倍の命令数を必要としていた。「SCEPTER」

の命令セットは SPARC Version7 を参考に作られていた。

しかし、命令セットが強くなっていったが故に、動作可能周波数は「Bonjour」, 「 T^2 」, 「SCEPTER」の順に遅くなっている。そこで、仮に、「SCEPTER」の性能を1とすると、「 T^2 」と「Bonjour」の相対性能は、それぞれ0.81, 1.39であるという結果になる。

パイプラインの性能がパイプライン化するための負担によって頭打ちになるとするならば、想定している命令セットに対して、最適な段数というものが存在すると考えられる。上記の簡単な議論で見ると、「 T^2 」, 「SCEPTER」は命令を強くしたことにみあう、十分なパイプライン化を行なっていなかったといえる。

4.5 デバッグ

ボードのデバッグにはロジックアナライザが多用された。バグの殆んどは、「MSB と LSB の位置のとり違い」, 「信号線のつなぎ忘れ」, 「TTL-IC の差し間違い」といった、見つけやすいものだった。図4にボードのデバッグ風景を示す。「Bonjour」と「 T^2 」は、この写真の、同じボード上に実装された。



図4：パイプライン RISC のデバッグ風景

コンピュータの仕組みを理解させる上で重要な事柄は、「コンピュータはデータバスと制御部から構成されていること」, 「データバスはプログラマから見える計算の経過という意味の状態を載せていること」, 「制御部は、さらに細かい単位で状態を変化させながら制御信号を発生し、命令が求める結果をデータバス上に実現させていること」だと考えられる [10].

これらは、Verilog-XL を用いた、レジスタトランスファレベルのビヘイビアの記述をデバッグする過程を通して、十分に理解された。

4.6 所要日数

表3に、設計と製作に要した日数を示す。「IS」, 「DP」などは、それぞれ、「命令セット」, 「データバス」, 「制御部」, 「全体」の設計, 「記述の手直し」, ボードの「製作」, 「デバッグ」に要した日数の平均である。「手直し」は、ロジックシンセサイザや自動配置配線にかけることが目的であった。

表3：設計と製作に要した平均日数

レジスタ構成	IS	DP	CTL	W	FIX	FAB	test
GR4	6.5	7.3	10.5	21.6	12	16.3	12.6
GR4,SP1	7.9	7.8	9.8	22.2	14.6	13.8	8.7
GR4,SP2	8.1	5.6	9.7	18.1	10.7	13.5	8.5
GR6	3	3	5	6	1	30	2
GR8	5.7	6	11.3	26.6	9.7	3.2	6.7
GR4,9bit	2	2	7	16	5	9	5
GR8,9bit	25	20	10	60	8	25	5
GR4,11bit Bonjour	21	21	21	30	14	28	14
GR10,9bit T^2	14	7	7	14	14	28	14
GR8,10bit SCEPTER	30	10	2	24	2	25	7

正規の授業日数は1日4.5時間、週2日、15週であったが、情報工学科の実験室は、いつでも利用できるように開放していた。特にデバッグの過程では、課外利用申請書に印がほしいと、連日、求められた。図5にこの時期の履修風景を示す。

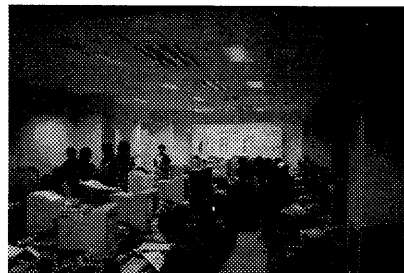


図5：履修風景（情報工学科実験室）

5 背景となる研究活動

情報工学実験 III は「論理回路学講座」が主担当している。論理回路学講座では City-1 と同様な考え方の環境を前提として、VLSI-CAD

技術を中心に、高速、高信頼なデジタルシステムを構築するための基盤技術の研究を行なっている。ここでは、このような活動の一部を簡単に紹介する。これらの成果は City-1 を用いた設計教育にも反映させていく考えである。

5.1 非ノイマン型演算ユニットの開発

Intel 社の Pentium Pro の場合、リオーダーバッファのエントリ数は 40 で、20～30 個の命令間のデータ依存関係の解析を行なっているが、同時発行可能な命令数は 3～5 個程度である。論理回路学講座では、フォンノイマン型コンピュータの演算ユニットを高速化する目的で、データフローなどの非ノイマン型コンピュータ構築技術を利用することを考えている。

IBM 社の 360/91 は、Tomasulo の技術を用いており [1],[2]、演算ユニットは事実上のデータフローコンピュータだとみなすことができる。

5.2 デジタル信号処理技術

Intel 社による MMX の出荷に象徴されるように、MPU の設計においても現在では、いわゆるマルチメディア市場におけるアプリケーションの重要性が強く認識されるようになった。論理回路学講座では、CAD 技術を前提に、マルチメディア指向の MPU と DSP の両面から、音響を対象とする、信号処理に関する研究を行なっている。

三菱と富士通からは、それぞれ D10V, MMA という VLIW の技術を用いた、DSP が発表されている。信号処理にアプリケーションを限れば上記の非ノイマン型コンピュータ構築技術がひきだせる並列度も高めることができる。

6 まとめ

本稿では、広島市立大学情報科学部情報工学科において、3 年前期の情報工学実験 III で用いている FPGA コンピュータの自由な設計と製作のための環境 City-1 を示して、CAD 技術、FPGA 利用技術を用いた、大学での教育研究状況を紹介した。

3 年前期は「自分のコンピュータをはじめて作る」という経験をする最良の機会であり、

自由な設計、製作環境を提供することの意義は大である。

3 年後期の情報工学実験 IV においては、本実験で学んだ知識を前提に、より発展的な授業が組まれている。

本研究の一部は、広島市立大学平成 6,7 年度特定研究「VLSI 設計教育に関する研究」

(弘中哲夫助教授代表) によるものである。

本研究で用いている CAD システムの、超高速論理シミュレータ、ロジックシンセサイザのライセンスは CADENCE 社のアカデミックプログラム、自動配置配線プログラムのライセンスは Xilinx 社のユニバーシティ・プログラムによるものである。

参考文献

- [1] Hennessy J. L. and Patterson D. A.: Computer Architecture A Quantitative Approach, Morgan Kaufmann Publishers, Inc.(1990)
- [2] Hennessy J. L. and Patterson D. A.: Computer Architecture A Quantitative Approach 2nd edition, Morgan Kaufmann Publishers, Inc.(1996)
- [3] 相磯秀夫, 飯塚 肇, 元岡 達, 田中英彦: 計算機アーキテクチャ, 岩波書店 (1982)
- [4] 村岡洋一: コンピュータアーキテクチャ, 近代科学社 (1981)
- [5] Hwang, K.: Computer Arithmetic, John Wiley & Sons(1979)
- [6] 当麻喜弘: スイッチング回路理論, コロナ社 (1986)
- [7] 吉田典可: 論理数学 II, 同 III 共立出版 (1978)
- [8] Takahashi R., Yoshimura T. and Goto S.: A VLSI Architecture Evaluation System, Proc. ICCD'86 pp.60-63 (1986)
- [9] 高橋隆一, 吉村 猛: ハイレベルシンセシスの動向, 信学論 A, Vol.J74-A, No.2, pp143-151 (1991)
- [10] 高橋隆一: システム開発と設計, サイエンス社 (1996)
- [11] 稲吉秀夫, 富田真治, 日比野 靖, 平山正治, 山本昌弘, 飯塚肇: パネル討論会「RISC は CISC に勝るか」, 情報処理 vol.30, No.11, pp.1376-1394 (1989)