

32ビット RISC マイクロプロセッサ DLX-FPGA の 設計教育フィジビリティ・スタディ

井上 弘士^{†1} 飯田 全広^{†2} 大内 正英^{†3} 久我 守弘^{†4} 末吉 敏則^{†5}

†九州工業大学 情報工学部 知能情報工学科
‡九州工業大学 マイクロ化総合技術センター

〒820 福岡県飯塚市川津 680-4

E-mail : {¹kouji,²iida,³oouchi}@mickey.ai.kyutech.ac.jp
⁴kuga@cms.kyutech.ac.jp
⁵sueyoshi@ai.kyutech.ac.jp

上級コース向き教育用マイクロプロセッサ DLX-FPGA を利用した設計教育の実現可能性について調査した。DLX-FPGA は、命令パイプラインや割込み等を備えた本格的な教育用マイクロプロセッサであり、実装デバイスとして書換え可能な FPGA を使用している。そのため、DLX-FPGA は教育方針やカリキュラムに応じた様々な教育を支援することができる。本稿では、教育用マイクロプロセッサ DLX-FPGA と大規模 FPGA を利用した DLX-FPGA 教材ボードについて紹介する。また、大学院生を対象に予備実験を行うことで DLX-FPGA の回路規模および開発時間を調査し、DLX-FPGA を用いた設計教育のカリキュラム案について検討した結果を報告する。

A Feasibility Study for Design Education Using 32bit RISC Microprocessor DLX-FPGA

Kouji Inoue^{†1} Masahiro Iida^{†2} Masahide Ouchi^{†3}
Morihiko Kuga^{†4} and Toshinori Sueyoshi^{†5}

† Department of Artificial Intelligence, Kyushu Institute of Technology

‡ Center for Microelectronic Systems, Kyushu Institute of Technology

680-4 Kawazu, Iizuka City, Fukuoka, 820 Japan

We studied the possibility of realizing an education system using the educational microprocessor DLX-FPGA which can be used for advanced computer architecture and LSI design education. The DLX-FPGA is a full-scale educational microprocessor, implemented using reconfigurable FPGAs, has pipeline, interrupt, and other features. The DLX-FPGA can support various types or levels of education depending on the educational aims and curriculum requirements. In this paper, we describe the educational microprocessor DLX-FPGA and the DLX-FPGA Board. We also have investigated the number of gates in DLX-FPGA, the period of development with preliminary results as implemented to advanced course students, and then describe the curriculum plan for design education utilizing DLX-FPGA.

1 はじめに

集積回路技術の進歩により集積度はめざましく向上し、高性能かつ高機能な LSI が実現されるようになった。また、ハードウェアやソフトウェアの大規模化・複雑化が進み、これらの開発を短期間で実現するための様々な設計支援環境が構築されてきた。今後、さらなる技術進歩により、高性能かつ大規模な新しいシステムの実現が予測される。そのため、ハードウェア/ソフトウェアのトレードオフの決定から LSI 設計までシステム全体を見渡すことのできる次世代の優秀な技術者や研究者の育成が重要となる。

そこで我々は、計算機工学教育や LSI 設計教育における上級コースでの利用を目的とする教材として、32 ビット教育用 RISC マイクロプロセッサ DLX FPGA の開発を行った [1]。DLX FPGA は、計算機工学の基礎教育を終えた学習者を対象とする教育用マイクロプロセッサであり、実用レベルに近いシステム設計教育を支援する教材として利用できる。ハードウェア教育においては、講義だけでなく、実験・演習として学習者に本格的なマイクロプロセッサの設計・動作確認を体験させることで高い学習効果を期待できる。一方、システムソフトウェア教育では、自ら設計したマイクロプロセッサを利用してオペレーティングシステムや最適化コンパイラなどの開発実験を行うことで、ハードウェア/ソフトウェアのトレードオフについて学習できる。また、LSI 設計教育においては、DLX-FPGA を利用して論理設計やテストベクタ生成、タイミング解析などを行い、学習者に実際の ASIC 開発工程を体験させることで効果的な教育を行うことができる。

本稿では、ハードウェア記述言語を利用して教育用 RISC マイクロプロセッサ DLX-FPGA の設計を行い、回路規模および開発時間を調査した。また、これらの調査結果を基にして、DLX-FPGA を利用した設計教育の実現可能性について検討した。以下、第 2 章では DLX-FPGA の概要と設計仕様を示し、第 3 章で DLX-FPGA の実装環境である DLX-FPGA 教材ボードを紹介する。第 4 章では、2 つの設計事例に基づき DLX-FPGA を利用した設計教育の実現可能性について考察し、教育目的に応じた DLX-FPGA 開発実験のカリキュラム案を提案する。最後に第 5 章で簡単なまとめと今後の課題を述べる。

2 上級コース向き教育用マイクロプロセッサ DLX-FPGA

2.1 DLX-FPGA の概要

DLX-FPGA は、実用レベルに近いシステム設計教育を支援する教材として開発した 32 ビット RISC タイプのマイクロプロセッサである。

DLX-FPGA は、基本となるプロセッサモデルとして DLX アーキテクチャを採用している。DLX は文献 [2] の中で教材として利用されるマイクロプロセッサであり、商用の RISC マイクロプロセッサの多くを平均化したようなアーキテクチャを有する。そのため、計算機の性能を向上させる様々な要素技術(命令パイプラインなど)が反映されており、かつ、理解しやすい構成となっている。また、DLX-FPGA の実装デバイスとしては、書換え可能な FPGA を利用している。そのため、学習者は論理回路の設計完了後、手元で短時間に LSI 化することが可能と

なる。また、書換えの回数に制限がなく、設計のやり直しや機能変更・機能拡張を何度でも行うことができる。

2.2 設計仕様

DLX-FPGA の設計仕様は、基本的に DLX アーキテクチャに準拠している。しかし、文献 [2] では、DLX の基本仕様のみが定義されており、オペレーションコードの割当てや割込み処理などの詳細な部分は未定義である。そこで、これらの未定義部分の仕様については独自に決定を行った。

命令セット: 命令は全て 32 ビット固定長であり、3 つの命令フォーマットに分類される。命令セットとしては、データ転送命令、算術論理演算命令、制御転送命令、浮動小数点命令の全 75 命令をサポートする。ただし、DLX の命令セットでは、単精度と倍精度の浮動小数点命令が定義されているが、実装規模や設計負荷を考慮して単精度浮動小数点命令に限定した。

レジスタセット: レジスタセットとしては、整数用・浮動小数点用レジスタファイルとして、それぞれ 32 個のレジスタを搭載している。また、特殊レジスタとして命令レジスタ、プログラムカウンタ、割込みアドレスレジスタ、データメモリ・アドレスレジスタ、ストアメモリ・データレジスタ、浮動小数点ストアメモリ・データレジスタを備えている。さらに、文献 [2] では DLX の実行状態を示すレジスタとして状態レジスタおよび浮動小数点状態レジスタが存在するが、各レジスタの構成は未定義であるため独自に定義している。

割込み: 割込みとしては、10 レベルの例外と 7 レベルの内部割込み、および、7 レベルの外部割込みをサポートしており、正確な割込み (precise interrupt) を保証する。

命令パイプライン: 全ての命令は 5 段のパイプライン・ステージ(命令フェッチ、命令デコード、実行、メモリアクセス、レジスタ書込み)を処理することで実行を完了する。DLX-FPGA の命令パイプラインは、図 1 に示すように、整数パイプラインと浮動小数点パイプラインで構成されており、整数命令と浮動小数点命令のオーバーラップ実行が可能である。また、浮動小数点パイプラインにおいては、浮動小数点加減算ユニット (FP-ADD Unit)、浮動小数点/整数乗算ユニット (FP/Integer-MULT Unit)、浮動小数点/整数除算ユニット (FP/Integer-DIV Unit) を搭載しており、浮動小数点加減算ユニット、浮動小数点/整数乗算ユニットは 3 段のパイプライン構成となっている。ただし、浮動小数点/整数除算ユニットは、回路規模を考慮しパイプライン化しておらず、マルチサイクル実行により除算を行う。

2.3 DLX-FPGA のラビッドプロトタイプリング

複数 FPGA を搭載した米国 Aptix 社の FPCB(Field Programmable Circuit Board)[3]を利用して、DLX-FPGA のラビッドプロトタイプリングを行った [4]。本 FPCB には、12 個の FPGA(約 13,000 ゲート相当の論理回路を実現可能)と 3 個の配線用デバイスである FPID(Field Programmable Interconnect Device)が搭載されており、約 156,000 ゲート相当の論理回路が実装可能である。また、FPID に構成データをダウンロードすることで、FPGA 間の配線を短時間で自由に変更できる。

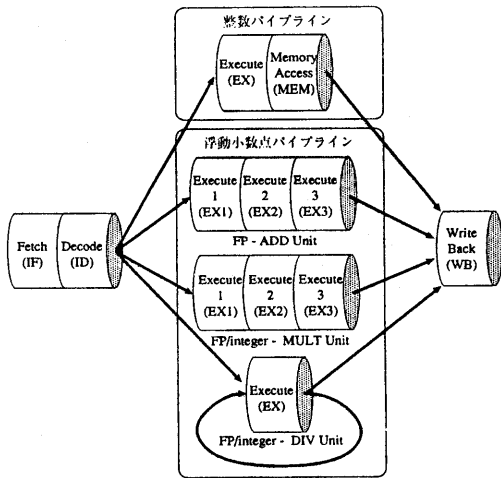


図 1: DLX FPGA 命令パイプラインの構成

ラビッドプロトタイピングを行った結果、回路図入力による設計では9個(整数パイプラインに4個, 浮動小数点パイプラインに5個), ハードウェア記述言語(VHDLを使用)による設計では12個(整数パイプラインに4個, 浮動小数点パイプラインに8個)のFPGAを利用することでDLX-FPGAの実装を完了した。また, 実装後, いくつかのアプリケーションプログラムを実行させ, その正確な動作を確認した。

3 大規模FPGAを利用したDLX-FPGA実装支援環境

3.1 DLX-FPGA 開発実験の導入における問題点

大学などの教育機関において, LSIを利用したマイクロプロセッサの開発実験を実施するためには, 厳しい時間的・予算的制約を満たさなければならない。しかし, 浮動小数点命令や乗除算命令をサポートするDLX-FPGAを設計対象とする場合, 以下のような問題点があげられる。

- DLX-FPGA全体を設計させる場合, 設計教育の教材としては大規模な構成となるため, 学習者に対して重い設計負荷がかかる。また, DLX-FPGAの設計完了後, 複数FPGAへ分割実装しなければならず, 論理回路の分割といった設計教育において本質的でない部分に多くの開発時間を費やしてしまう恐れがある。
- DLX-FPGA全体を実装するためには, 複数個のFPGAと配線用デバイスであるFPIDを搭載したFPCBのような, 大規模回路を実装できる環境が必要である。しかし, 現状では, このような実装環境を学生実験向けに低コストで実現することは難しい。

3.2 DLX-FPGA 教材ボードの開発

3.1節で述べた問題点を考慮すると, DLX-FPGAが設計教育に広く利用されるには, 1) 開発時間の短縮, 2) 低

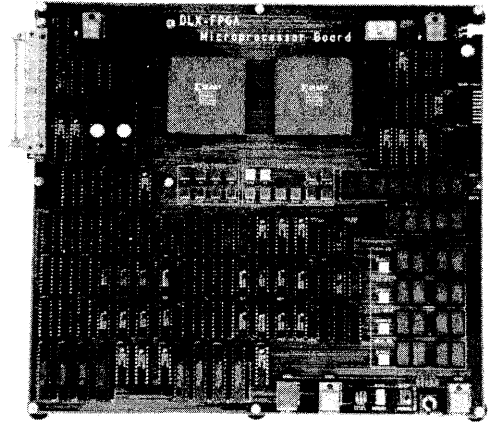


図 2: DLX-FPGA 教材ボード

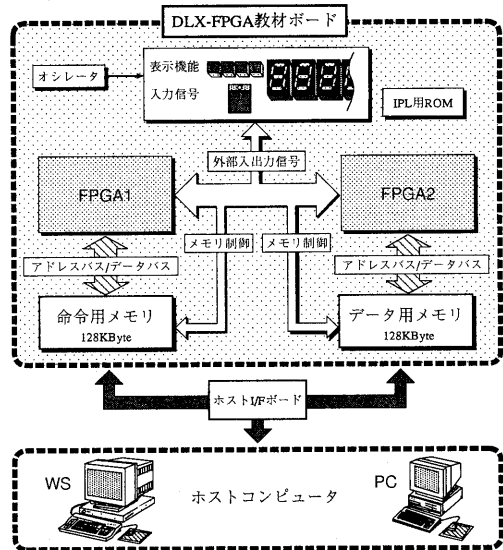


図 3: DLX-FPGA 整数パイプライン実装環境

コストの実装, を実現できる教育環境を提供しなければならない。そこで, 我々は, 設計対象をDLX-FPGAのサブセットである整数パイプライン(浮動小数点命令, 乗除算命令を未実装とする)に限定し, このDLX-FPGA整数パイプライン用の実装環境としてDLX-FPGA教材ボードを開発した[5]。

DLX-FPGA教材ボードならびにDLX-FPGA整数パイプラインの実装環境を図2, 図3に示す。DLX-FPGA整数パイプラインは, ハードウェア記述言語による設計

例では換算ゲート数にして約 28,000 ゲートであった [1]。学習者の能力の違いや論理合成ツールの性能による設計結果のばらつき、さらには、機能拡張などによる回路規模の増大を考慮し、DLX-FPGA 教材ボードには約 25,000 ゲート相当の論理回路が実現可能な FPGA(米国 Xilinx 社の XC4025[6]) を 2 個搭載した。

図3で示すように、各 FPGA には独立にアクセス可能な命令用メモリとデータ用メモリが接続されている。これらのメモリは、ホスト I/F ボードを介してホストコンピュータとのデータ転送が可能である (SCSI および RS232 インタフェースをサポート)。また、IPL 用 ROM を利用して、命令メモリの初期化を行うことができる。さらに、FPGA 初期化用 ROM を搭載しており、DLX FPGA 教材ボード単体での動作も可能である。

表示機能としては、DLX FPGA 実行の様子を表す複数の LED を搭載しており、各パイプライン・ステージにおいて現在実行中の命令とその無効化信号、状態レジスタのフラグ、命令用/データ用メモリに対する制御信号などをソフトウェアの存在なしに観測できる。また、8 個の汎用 LED を利用して、観測すべき信号線を設計者が自由に選択することも可能である。FPGA への入力信号としては、7 レベルの外部割込み要求信号、リセット信号、クロック信号などがある。また、DLX FPGA の実行モードとして、通常モード/ステップ実行モードを選択可能であり、通常モードではクロックスピードを 0.1Hz から 250KHz まで変更できる。さらに、別搭載のオシレータを利用して DLX FPGA に高速クロックを供給することも可能である。

4 DLX-FPGA の設計教育フィジビリティ・スタディ

4.1 DLX-FPGA を利用した設計教育

設計教育の教材として DLX FPGA を利用することで、学習者は図 4 に示すような項目を学習できる。

システム設計 (方式設計) の教育では、ハードウェア教育を対象として、高速化技法の基礎技術であるパイプライン処理やロード/ストア・アーキテクチャ、命令のオーバラップ実行により発生するハザード問題とその解決法 (フォワードイング、ロード遅延、遅延分岐など) を学習できる。また、システムソフトウェア教育を対象とした際には、自ら設計した DLX-FPGA を利用して最適化コ

教育内容	DLX-FPGA を利用した学習項目
システム設計 (方式設計)	<ul style="list-style-type: none"> ● HW/SW トレードオフの決定 ● アーキテクチャの理解 ● システムソフトウェア (最適化コンパイラ、OS など)
論理設計	<ul style="list-style-type: none"> ● トップダウン設計 ● ハードウェア記述言語による設計 ● HDL シミュレータを利用した機能検証 ● 論理合成・最適化 ● テスタータ生成
レイアウト設計	<ul style="list-style-type: none"> ● 配置 ● ハードマクロ (相対配置マクロ) ● タイミング解析

図 4: DLX-FPGA を利用した学習項目

ンパイラや OS の開発を行うことが可能である。この場合、マイクロプロセッサの機能や内部構成を十分把握した上でシステムソフトウェアの開発演習を実施できるため、高い学習効果を期待できる。

論理設計の教育では、学習者が実際にマイクロプロセッサを設計することで、トップダウン設計手法を学習できる。また、ハードウェア記述言語による設計、HDL シミュレータを利用した機能検証、論理合成・最適化、テストベクタ作成等を行うことで、実際の ASIC 設計における作業の流れを体験することが可能である。

レイアウト設計の教育では、自ら設計した論理回路のタイミング解析、レイアウトを行うことができる。ただし、レイアウトの演習では、実装デバイスとして利用する FPGA の論理セルの配置を指定し自動配線を行うため、一部 FPGA 特有のレイアウト作業となることがある。

4.2 ハードウェア記述言語による DLX-FPGA の設計事例

DLX-FPGA が設計教育の教材として広く利用されるには、様々な教育目的 (どの教育項目に重点を置くか) やカリキュラムによる時間的制約に対し柔軟に対応できなければならない。そこで、回路規模および開発期間を調査するため、大学院生 2 名を対象とした DLX-FPGA 整数パイプライン開発の予備実験を行った。

4.2.1 設計条件

設計事例 A では、教育目的として計算機工学指向の設計教育を想定し、仕様書のみを与えて内部構成の検討・設計・機能テストプログラム作成・シミュレーション・実装・動作確認までを行った。ただし、設計者は、過去にハードウェア記述言語 (Verilog-HDL) の利用経験があり、今回の設計においても Verilog-HDL を使用した。一方、設計事例 B では、教育目的として LSI 工学指向の設計教育を想定し、論理設計およびレイアウト設計に重点を置くものとする。そこで、システム設計における作業を低減するため、あらかじめ仕様書だけでなく内部構成例も提供した。設計者は内部構成例に基づいて設計し、機能テストプログラム作成・LSI テストベクタ作成・シミュレーション・タイミング解析・実装・動作確認を行った。ただし、設計者は、回路図入力によるマイクロプロセッサの開発経験はあるが、ハードウェア記述言語 (VHDL を利用) による設計は今回が初めてである。

4.2.2 実装結果

設計事例 A・設計事例 B における DLX-FPGA 整数パイプラインの実装結果を図 5 に示す。図中の () の値は、実装デバイスである FPGA の論理セル (CLB) 利用率を表す。ただし、レジスタファイルの実装においては、内部資源の RAM マクロを利用することで回路規模および配線数の増大を回避した。設計事例 A・設計事例 B ともに、50% 以下の内部資源利用率で DLX-FPGA 整数パイプラインの実装を完了している。従って、トップダウン設計やハードウェア記述言語に不慣れな学習者を教育対象とする際に生じる、回路規模のばらつきにも十分対応することができる。

4.2.3 開発時間

図 6 に、設計事例 A・設計事例 B における DLX-FPGA 整数パイプラインの開発所要時間を示す。

	設計事例A		設計事例B	
	FPGA1	FPGA2	FPGA1	FPGA2
Packed CLBs	383(37%)	502(49%)	428(41%)	457(44%)
Flip Flops	276(13%)	212(10%)	295(14%)	196(9%)
Bonded I/O Pins	239(93%)	232(90%)	240(93%)	222(86%)

図 5: DLX-FPGA 整数パイプラインの回路規模

方式設計	作業工程		事例A(時間/人)	事例B(時間/人)
方式設計	仕様理解		6	6
	内部構成の検討		4	0
論理設計	HDLソース作成(DLX-FPGAの記述)		20	52
	HDLソース作成(テスト環境の記述)		4	11
	機能テストプログラム作成		12	34
	テストベクタ生成		0	22
	機能検証		12	27
	論理合成・配置配線		43	5
	タイミング解析		0	2
	実装・デバッグ		17	1
合計開発時間			128	160

図 6: DLX FPGA 整数パイプライン開発の所要時間

方式設計の工程では、設計者 A は仕様の理解から内部構成の検討・決定までに 20 時間要している。これに対し、設計者 B はあらかじめ与えられた内部構成例に従った設計を行ったため、方式設計に要した時間は仕様理解の 6 時間のみである。

論理設計の工程において、設計事例 A では、設計者がハードウェア記述言語の使用経験者であったため HDL ソースの作成(DLX FPGA のソースとテスト環境のソース)を 24 時間で完了した。また、機能テストプログラム作成および HDL シミュレータを利用した機能検証には 24 時間を要している。機能テストプログラムは全てアセンブリ言語で記述しており、基本命令テスト 1.192 クロック、例外・割込みテスト 762 クロック、ハードウェア機能テスト 81 クロックである。しかし、機能検証において論理回路の誤りを十分に検出できなかったため、論理合成から実装・デバッグの工程で 60 時間費やしている。

一方、設計事例 B では、設計者がハードウェア記述言語の学習と DLX FPGA の設計を同時進行了ため、HDL ソースの作成には 63 時間を費やしている。また、機能テストプログラムおよびテストベクタは全てアセンブリ言語で記述しており、これらのテストソース作成には 56 時間を要している。機能テストプログラムは、基本命令テスト 1,822 クロック、例外・割込みテスト 7,332 クロック、ハードウェア機能テスト 486 クロックである。テストベクタの実行には、テストベクタ実行環境を利用した。このテストベクタ実行環境は、設計を完了した DLX FPGA のネットリストをゲートレベルの HDL 記述に変換し、HDL シミュレータ上でテストベクタを実行させる。また、実行結果としては、セル稼働率および活性化していないネット名を出力する。テストベクタを実行した結果、98%以

上のセル稼働率を達成した。

4.3 DLX-FPGA を利用したカリキュラム案

DLX FPGA を用いた設計教育を行う場合、当該学科の教育理念やカリキュラムによる時間的制約などに応じて、4.1 節で示した教育内容の中から選択することができる。本節では、4.2 節で示した設計事例を基に、DLX-FPGA を利用したカリキュラム案を紹介する。なお、カリキュラムの時間的制約として、一学期 15 週の 67.5 時間(週 4.5 時間)で講義・演習を行う場合を想定している。

4.3.1 計算機工学指向の設計教育

4.2 節における設計事例 A は、計算機工学指向の設計教育を想定しており、内部構成の決定を設計者自ら行っている。総開発時間は 128 時間となり、学習者 1 人での演習を想定した場合には、一学期間週 2 回あるいは通年の授業で DLX-FPGA 開発実験を実施せざるを得ない。もし一学期の授業時間内で DLX-FPGA の開発実験を実施するのであれば、論理設計における作業工程を軽減させる必要がある。設計事例 A では、HDL シミュレータによる機能検証が不十分であったため、総開発時間の約 50%を論理合成・配置配線および実装・デバッグに費やしている。そこで、テスト環境(テスト環境作成のための HDL ソースと機能テストプログラム)をあらかじめ学習者に提供する。これにより、HDL ソース作成や機能テストプログラム作成に要する時間を削減できるだけでなく、十分な機能検証を行わせることで、実装後のデバッグに伴う論理合成・配置配線のやり直しに要する時間も短縮できる。すなわち、方式設計および HDL ソース作成には設計事例 A より 40 時間(仕様理解に 6 時間、内部構成の検討に 14 時間、DLX FPGA の HDL ソース作成に 20 時間)、機能検証・論理合成・配置配線・実装・デバッグには設計事例 B より推定して 33 時間(機能検証に 27 時間、論理合成・配置配線に 5 時間、実装・デバッグに 1 時間)の合計 73 時間で開発実験を完了できる。また、時間的余裕がない場合には、処理負荷の重い論理合成や配置配線などに授業以外の時間を利用することが考えられる。

前学期で DLX FPGA の設計および実装を完了した場合、後学期において DLX-FPGA の仕様や内部構成を再検討し、設計のやり直しを行わせることもできる。DLX-FPGA で利用する FPGA は書換えの回数に制限がなく、何度でも論理回路の実装が可能である。また、設計事例 A・設計事例 B ともに半分以下の資源利用率で DLX-FPGA 整数パイプラインの実装を完了している。そこで、残りの内部資源を利用して DLX-FPGA の機能拡張や仕様変更を行い、学習者独自のハードウェア機構(アイデア)を盛り込むことが可能である。例えば、DLX-FPGA の基本仕様において未実装である分岐予測機能の採用(基本仕様では分岐不成立予測法を採用)や遅延分岐法の利用、仮想記憶をサポートするための MMU の追加などがある。その後、最適化コンパイラや OS などのシステムソフトウェア開発実験を行うことで、学習者は計算機の性能向上を実現するために重要な事柄となる、ハードウェア/ソフトウェアのトレードオフについて学習できる。

4.3.2 LSI 工学指向の設計教育

LSI 工学の設計教育では、論理設計や回路設計から機能検証、さらにはテストベクタ作成など実際の ASIC 開発工程を体験させることを目的とする場合があげられる。4.2 節の設計事例 B では、LSI 工学指向の設計教育を想

定しており、仕様書だけでなく内部構成例をあらかじめ提供することで、方式設計に要する時間を短縮している。しかし、総開発時間は160時間と長く、HDLソース作成からテストベクタ作成、実装・デバッグまでの全工程を学習者1人に行わせる場合には、一学期間週2回もしくは通年の実験・演習にせざるを得ない。

設計事例Bにおいて、設計者はハードウェア記述言語の使用経験がなかったため、HDLソースの作成に総開発時間の約40%を費やしている。そこで、一学期の授業時間内で実験・演習を実施するために、演習付き講義を行うことで開発時間を短縮する方法が考えられる。この演習付き講義を利用した場合のDLX-FPGA開発実験例を図7に示す。まず、講義においてDLX-FPGA設計仕様および内部構成の説明を行う。次に、ハードウェア記述言語の説明と同時に、演習例題としてDLX-FPGAの構成要素となる部品(ALUなど)を設計させる。各部品の設計完了後、HDLシミュレータを利用して動作確認を行う。これにより、シミュレータの利用法も同時に習得できる。時間的余裕がない場合は、あらかじめ各部品のHDLソースを用意し、一部穴埋め形式問題などを解決させることで時間短縮を図ることも可能である。

あらかじめHDLの講義・演習を行った場合、学習者全員がDLX-FPGA構成要素の各部品に対する知識を習得できる。そのため、複数人で構成されるチーム単位でのDLX-FPGAの開発が行い易くなる。そこで、1チーム3人程度のグループを編成し、各チームごとにDLX-FPGAを設計する方法が考えられる。DLX-FPGAの記述およびテスト環境の記述を行う際には、演習付き講義で作成したHDLソースを利用する。また、各チームのメンバーは作業を分担し、HDLソース作成、機能テストプログラム作成、テストベクタ作成を並行して進めることができる。そのため、これらの作業時間を29時間(HDLソース作成に30時間+機能テストプログラム作成に34時間+テストベクタ作成に22時間/3人)とすると、図7より合計70時間程度で実験・演習を完了できる。

一方、学習者全員に対し、個別にDLX-FPGAの開発を行わせる場合、HDLソースおよびテストプログラムのサンプルをあらかじめ提供することで開発時間を短縮することもできる。まず、演習付き講義においてDLX-FPGA構成要素の部品だけでなく、DLX-FPGA全体の設計まで完了させる。次に、学習者は、あらかじめ与え

られたサンプルプログラムを参考にして機能テストプログラムやテストベクタを作成する。例えば、数命令しか使用していないテストプログラムやセル稼働率の低いテストベクタを提供し、学習者はこれらを拡張することでテスト用のプログラムを完成させる。また、演習付き講義でDLX-FPGAのHDLソースを完成させるため、機能検証に要する時間も短縮できる。各作業に要する時間を図7で示すように仮定すると一学期(63時間程度)でDLX-FPGA開発実験を完了する。

5 おわりに

本稿では、FPGAを利用した上級コース向き32ビット教育用RISCマイクロプロセッサDLX-FPGAとその実装環境であるDLX-FPGA教材ボードを紹介した。また、DLX-FPGAを利用した設計教育の実現可能性を検討するため回路規模および開発時間を調査し、DLX-FPGAを利用したカリキュラム案を提示した。

今回、本学の大学院生2名を対象にして、DLX-FPGAを利用した設計教育における予備実験を行った。しかし、実際の大学院教育では、やはり多数の受講者を前提とした設計教育の実施に伴う様々な問題点も明らかにする必要がある。そこで、本稿の議論に基づき、現在本学大学院の講義(後学期)において博士前期課程(修士課程)一年生14名(知能情報工学コース7名、電子情報工学コース6名、機械システム工学1名)を対象とした、チーム編成に基づくトップダウン設計によるDLX-FPGA開発の実験演習を実施中である。なお、本稿では基本的に一学期を通じての講義・演習を仮定したが、学習者の専門的バックグラウンドや能力の違いによって、カリキュラム時間内にはDLX-FPGA開発実験を完了できない場合もあり得る。これに対処するため、実際のフィールド実験で得られた知見や対処策に基づいて、さらに教材やカリキュラムの改善を図り完成度を高めていく予定である。

謝辞

日頃御討論頂く、本学情報工学研究科の奥村勝氏、横尾徳保氏に感謝いたします。

なお、本研究は一部、文部省科学研究費補助金試験研究(B)(課題番号06558043)ならびに九州計測器株式会社との共同研究による。

参考文献

- [1] Nakagaki, K., Onchi, M., Inoue, K., Apduhan, B.O., Kuga, M., Suceyoshi, T.: "Design and Implementation of the Educational Microprocessor DLX-FPGA Using VHDL," in Proc. of the 2nd Asian Pacific Conference on Hardware Description Languages, pp.147-150, 1994.
- [2] Hennessy, J.L., and Patterson, D.A., Computer Architecture: A Quantitative Approach, Morgan Kaufmann Publishers, Inc., 1990.
- [3] Aptix Corporation: System Data Book, 1993.
- [4] 井上, 中垣, 大内, 久我, 末吉: "教育用RISC型マイクロプロセッサDLX-FPGAとそのラビッドシステムプロトタイプリング," 信学技報CPSY95-20, 1995.
- [5] 末吉, 井上, 奥村, 久我: "教育用32ビットRISCマイクロプロセッサDLX-FPGAと教材ボードの開発," 第3回FPGA/PLD Design Conference & Exhibit 論文集, pp.579-588, 1995.
- [6] Xilinx Inc.: The Programmable Logic Data Book, 1994.

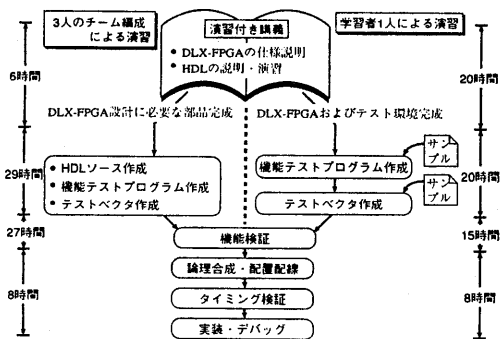


図7: 演習付き講義を利用したDLX-FPGA開発実験(案)