

アーキテクチャ教育における FPGA を用いた組み込み設計と実装

青山 義弘

福井工業高等専門学校 電子情報工学科(〒916-8507 鯖江市下司町)

yfa@fukui-nct.ac.jp

System design and implementation using an FPGA kit in architectural education

Yoshihiro AOYAMA

National Institute of Technology, Fukui College,
Department of Electronics and Information Engineering
(Geshi, Sabae, Fukui 916-8507, Japan)

(Received October 26, 2015; Accepted December 25, 2015)

Abstract

In this study, we integrated into the FPGA kit a unique design circuit that was described in VHDL. We designed a new circuit with a combination IP core and tried an integrated software, SOPC Builder, and a program development software, NiosII-EDS, for the IP core. Using the FPAG implementation test, we were able to synthesize the schematic design and the logic of the LED blinking circuit by introducing the NiosII processor to prepare a use case diagram according the UML of the LED blinking system.

Key words: FPGA kit, VHDL, LED blinking system, NiosII processor

1. はじめに

コンピュータのシステム設計におけるアーキテクチャ教育は、ハードウェアの視点から極めて重要である。実際にコンピュータシステムを構築することによりその動作、性能等を体感することはアーキテクチャ教育を行う上で非常に有効である。また、そのハードウェアに関しても昨今のパソコン自作で行われているボードや入出力機器(デバイス)の接続、結線といったレベルから、CPU 内部の構成を設計するレベルまでが想定される。どのレベルの教育を行うか目的に応じて変更することができる

CPU やメモリ、各種 I/O 部品の接続レベルのコンピュ

ータシステム設計によるアーキテクチャ教育を行うために、これまで研究室レベルでは CPU ボード上にそれぞれの部品を置き、半田付け等でシステムを作成してきた。しかし、今回は FPGA(Field Programmable Gate Array)内にこれらの部品を実装することにより、システムオンチップ (SoC : System on a Chip : IC チップ上にコンピュータシステムを構築すること、市販のマイコンがこれにあたる) 開発を体験させ、マイコンによる簡単な入出力制御の実験を導入したので、そのアーキテクチャ教育の実装成果[1]を報告する。

2. FPGA 評価キット

FPGA は内部回路をプログラミングできる LSI であり、近年の技術発展により高集積化・高性能化・低消費電力化・低コスト化が進み、通信基地局・大規模ルータなどに幅広く採用されている。これまでは、FPGA 教材として豊田高専の仲野巧教授らと共同研究「高専における LSI 設計教育カリキュラムに関する研究」で教育用 FPGA 実験ボードによる実験を行ってきた[2,3,4]。今回は、より多機能な FPGA 評価キットである Altera 社が開発した FPGA の CycloneIII ファミリの 3C25F324 を搭載したキット NEEK (NiosII Embedded Evaluation Kit: NiosII 組み込み評価キット)を用いることとした (図 1)。NEEK はスターター基板とマルチメディア基板が HSMC コネクタ (High Speed Mezzanine Card Connector)でつながった構成になっている。スターター基板には FPGA として CycloneIII ファミリの 3C25F324(25000 ロジック・エレメント、18×18 ビット乗算器 66 個、600K ビット専用メモリ)が搭載されており、各種メモリ(16M バイト・フラッシュ ROM、1M バイト同期型 SRAM、32M バイト DDR-SDRAM)および押しボタン・スイッチ、LED が搭載されている。FPGA には NiosII プロセッサが実装され、その動作確認を容易に行えるハードウェアとなっている。また、JTAG-USB インタフェースである USB-Blaster 機能を持った MAX CPLD も搭載しており、パソコンとは USB ケーブルを接続するだけですべての開発が行うことができる。



図 1 NEEK キットの外観

マルチメディア基板には、タッチ・スクリーン付きの高精細 TFT LCD モジュール(840×480 ピクセル)に加え、オーディオ入出力、コンポジット・ビデオ入力、VGA ビデオ出力、RS-232-C、PS/2 コネクタ、10/100Base Ethernet、SD カード・スロットと豊富な入出力インタフェースが用意されている。

FPGA を用いることにより、マイコンや各種部品をボード上で組み合わせハードウェア設計を行う実験に代わり、コンピュータ上の EDA (Electronic Design Automation) ツールを用いることにより、ハンダ付けやワイヤラッピング等の結線作業無しに FPGA 上でほとんどの部品の結合を行い、人間の作業における単純なエラーを無くすることができる。

また、本研究の主旨は、これまでの部品に対応する IP コア(Intellectual Property core)による回路設計であり、これを組み込みシステム設計と呼ぶ。そして、この組み込みシステムを設計するために、本来はソフトウェア設計時にその内部使用を記述するためのフローチャート記述より、さらに高度な記述を行う際に用いるモデル記述手法 UML (Unified Modeling Language)をハードウェア設計にも用いて SoC の仕様を図 2 のように作成する[5,6]。

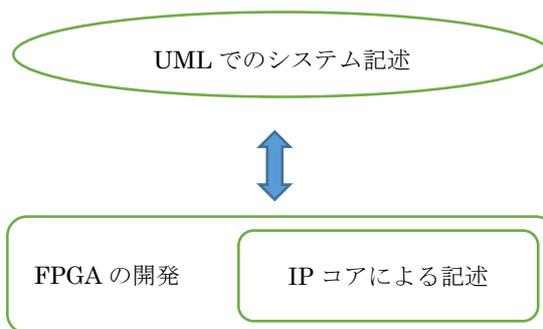


図 2 本研究の主旨

3. ハードウェアの設計

IP コアの利用を目的とするため、簡単な LED 明滅回路を設計した。また、開発ツールの使用方法を学ぶため SOPCBuilder を使用しない QuartusII のみでのブロック図をベースとした回路も並行して設計した。設計した回路を図 3 に示す。

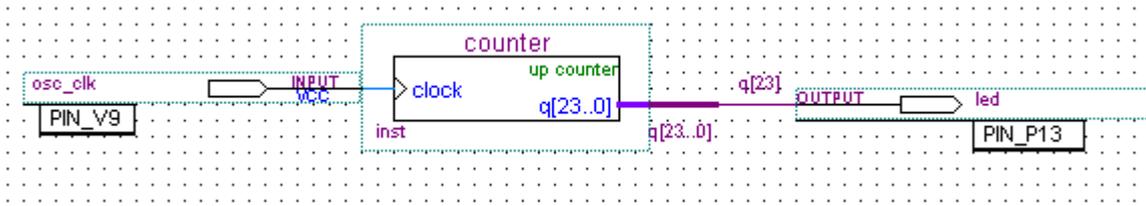


図 3 LED 明滅回路のブロック図

LED 明滅回路は、スターター基板に搭載されているオシレーター(50MHz)をクロックとして 24bit カウンターを動作させ、最下位ビットを出力として LED を明滅させるものである。使用している素子は、INPUT、COUNTER、OUTPUT の 3 つのみで COUNTER についてはメガファンクションと呼ばれる IP コアであり、その設定が必要である。設定は、LPM_COUNTER で図 4 のように行う。

カウンターにより 1 と 0 の反転周期を遅延させ出力とした。この時の LED の明滅は $0.02 \mu \times 2^{23}$ 秒である。また、FPGA への実装にはピンアサインメントという INPUT や OUTPUT へのピン番号の割り当てが必要である(図 5)。このピン番号は製品によって異なるので製品のマニュアルを参照して設定する。NEEK に搭載されている CycloneIII EP3C25F324C6 の場合はオシレーターのピン番号が V9 であり、LED のピン番号が P13 であるため、そのように設定する。

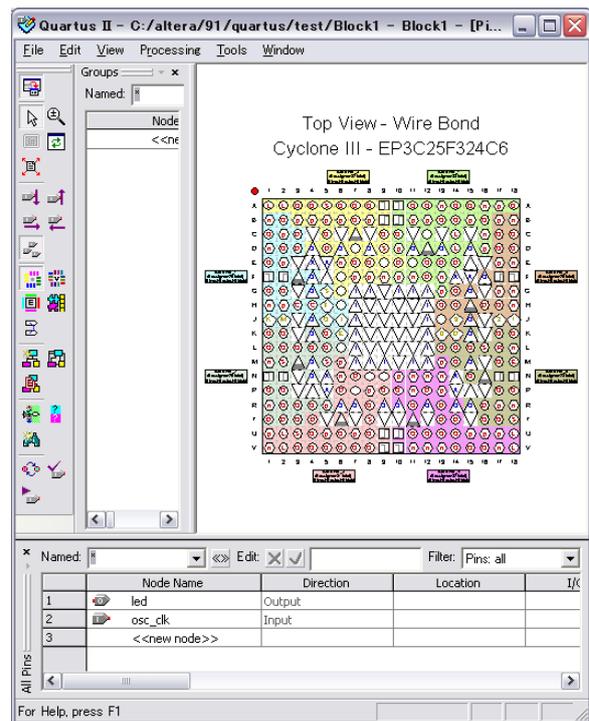


図 5 ピンのアサイン画面

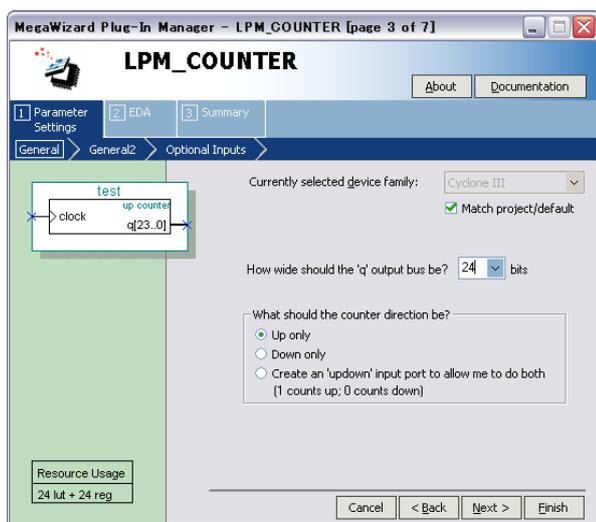


図 4 COUNTER の設計画面

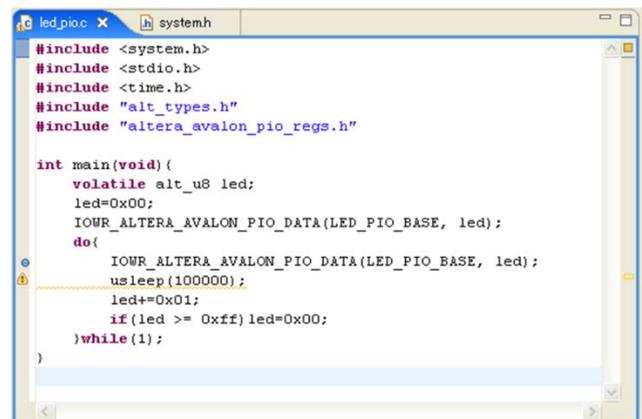


図 6 LED 明滅回路のプログラムと NiosII 画面

開発ツールとして使用した NiosII-EDS では、SOPCBuilder で Generate した際に生成されるファイル(.ptf)が必要であり、システムで定義された情報や定義から

生成されたファイルの情報が記されている。NiosII では C 言語や C++ でプログラムを記述できる。作成した C プログラムを図 6 に示す。

図 6 に示したプログラムは led という 8bit 幅の unsigned char を定義し初期値 0x00(16 進数で 00)を設定する。それを IOWR_ALTERA_AVALON_PIO_DATA (LED_PIO_BASE,led);で LED_pio のアドレスの初めに led を入力、としている。その後無限ループで、「led を入力、0.1 秒待つ、led の値を 1 増やす、led の値が 0xff になったら 0x00 に戻す」ことを繰り返している。

QuartusII のみで作成するブロックをベースとした回路のクラス図を図 7 に示す。oscillator と 24bit counter と

LED というクラスがある。oscillator は 24bit counter から見ればクロックであり、逆に 24bit counter は oscillator から見ればカウンターである。

統合システムという箱を図 7 の 24bit counter に置き換えるとほとんど同じ構成であることがわかる。図 8 の統合システムというパッケージは SOPCBuilder で統合したシステムを指す。また、cpu と memory 間に示される名前は、その 2 つのプログラム関連を示した。cpu と PIO(PIO は output only)間には使用とあり、図 8 中の黒三角が PIO に向いているが、これは cpu が PIO を使用するという意味である。



図 7 QuartusII のみで作成するブロックをベースとした回路のクラス図

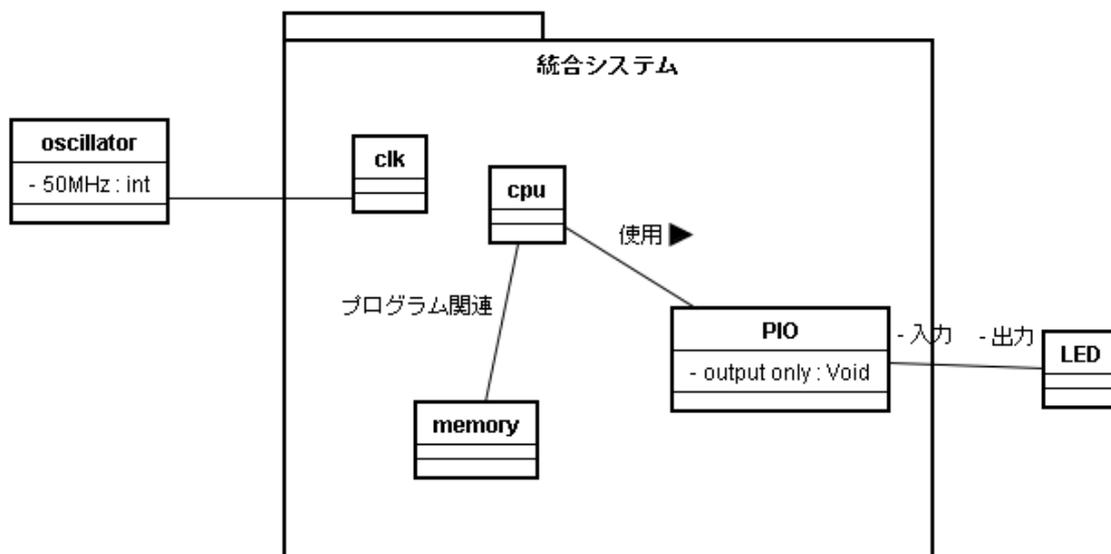


図 8 SOPCBuilder を用いた IP コアでのシステムのクラス図

表 1 コンパイル時間とピン数・ロジックエレメンツ(LE)数の関係

	LE 数	レジスタ数	ピン数	コンパイル時間	SOPCBuilder の構築時間
図 3 で作成した回路と同等のもの	1	1	2	1 分程度	
SOPCBuilder 作成の回路と同等のもの	1157	557	6	2 分程度	2 分程度
サンプルデザイン(ピクチャビューア)	10249	6492	133	26 分程度	7 分程度

4. FPGA 実装の評価

作成した回路やシステムのコンパイルが success した場合、拡張子(.sof)で実装用の情報が記されたファイルが生成される。QuartusII に付属する Programmer を使って NEEK へコンフィギュレーションを行う。コンフィギュレーションとは実際の FPGA のピンにピン割当てを設定することである。

Hardware Setup は USBBlaster を選択し、Auto Detect でコンパイルした回路のファイル(.sof)が現れる。その後、パソコンの USB と NEEK をケーブルでつなぎ、NEEK を起動し画面の Start を押すことで NEEK へコンフィギュレーションされる。

実装後、LED が明滅するのを確認した。ハードウェア環境は、Windows XP SP2、Pentium® D CPU 3.00GHz、2.99GHz 0.99GB RAM、QuartusII ver.9.1、NiosII-EDS ver.9.1 で FPGA 実装を評価した。評価データとして、コンパイル時間とピン数、ロジックエレメント(LE)数の関係を表 1 に示す。

5. まとめ

本研究では、独自に設計した回路を VHDL で記述し FPGA に実装して動作および性能試験を行うだけでなく、IP コアを組み合わせて新しい回路を設計し FPGA に実装し動作および性能試験を行った。その際、仕様の記述方法に UML 記述を用いること、およびこれまで実験等で使用していた FPGA 実装ツール以外に新規に IP コアによる統合システム生成用ソフトウェア SOPCBuilder や、そこで生成したプロセッサ用のプログラム開発ソフトウェア NiosII-EDS の試用を行った。研究成果を以下に箇条書きする。

- 1) プロセッサなしの LED 明滅回路の仕様記述として UML によるユースケース図と各クラス図の作成
- 2) LED 明滅回路のスキマティックによる設計と論理合成及び FPAG 実装テスト
- 3) NiosII プロセッサを導入し統合した LED 明滅システムの仕様記述として UML によるユースケース図と各クラス図の作成

- 4) 上記の組込みシステムの IP コアを用いたハードウェア設計とソフトウェア設計および論理合成及び FPAG 実装テスト

今回は、開発ツールと IP コアによる設計を目的としていたため簡単な回路(システム)を設計し開発したが、FPGA の規模や NEEK の機能を考えるともっと大きなシステムを設計するべきである。

FPGA での開発は、電子回路の知識や FPGA 開発の把握、IP コアをどのようなシステムに組み込むかを理解している必要がある。これは FPGA での開発をいくつか行うなどの活用をすることが重要である。

さらに、UML 記述について考察すると、実際のモデルを UML で記述することによって、整理した表現ができ、実際の仕様書への適用が可能であることがわかった。そして UML をハードウェア設計に適用することで、要求仕様や機能仕様、検証仕様の可読性を上げることが考えられる。

また、ハードウェア設計における UML の利点としては、統一された表記で表現された図を多用することで複雑な動作を視覚的に表現できるので有用である。

しかし、もともと UML 記述はオブジェクト指向ソフトウェア開発のモデリング手法として考えられている。関数内をブラックボックス化し、データの入出力、メソッドの指定をするという点では、ハードウェアと非常に似ており、これを活用することは有用であり、それ以上の記述表現がある。視覚的に表現することは、文書の理解を早め、早期に問題点を洗い出すことにつながっていくので良いが、統一された表記を学び、表現のバラつきを抑えることにはかなり熟練する必要があることがわかった。

今後は、ハードウェア仕様記述用に簡略化した UML 記述を考え教育に应用することを考える必要がある。さらに、ハードウェア仕様だけに UML 記述を利用するのではなく、さらにモデルを分析／詳細化していくことで、RTL (register transfer level) レベルの設計へ援用の可能性も考えられる。

参考文献

- 1) 川崎基輝、“IP コアによる FPGA 開発”，平成 21 年度卒業論文(2009)
- 2) 仲野巧、“教育用マイクロプロセッサと組み込みシステム用教育ボード”，システム LSI 琵琶湖ワークショップ論文集，第 4 号，275-278 (2000)
- 3) 青山義弘 他、“FPGA を用いた組み込みシステム設計によるアーキテクチャ教育の提案”，福井工業高等専門学校研究紀要，No. 45，13—17 (2011)
- 4) 月本功 他、“マイコン機能を搭載した教育・研究用 FPGA ボードの開発”，詫間電波工業高等専門学校研究紀要，No.36，57—60 (2008)
- 5) 狩野大樹 他、“UML 図を用いたハードウェア設計手法による実証実験と評価”，情報処理学会研究報告，Vol.2011-EVA-36，No.2 (2011)
- 6) “SoC 設計にモデリング手法を導入する”；
http://www.itmedia.co.jp/keywords/soc_model.html