

# C-1 COMET-II 互換プロセッサによる CPU 設計演習環境の開発

## Development of the CPU Design Training Environment using the COMET-II Compatible Processor

河合 一慶† 宮内 新† 荒井 秀一†  
Kazuyoshi Kawai Arata Miyauchi Syuichi Arai

### 1. はじめに

COMET-II とは経済産業省が実施している情報処理技術者試験センターで採用されている仮想プロセッサである。本学のハードウェア教育として、以前より COMET 互換プロセッサを実装した COMET ボードによる演習を行ってきた。本学では学部 1 年次においてプロセッサを外側から見たときの動作をシミュレータを用いて教育を行っている。2 年次においては、プロセッサの内部がどのようなになっているのかを理解する為に、レジスタトランスファ論理(RTL)設計を行っている。そして、3 年次においては実際に CAD ツールを用いて RTL によるプロセッサの内部動作を記述することで、実機による演習を行っている。本稿で述べる COMET-II ボードは、この演習で使用するものである。

平成 13 年度から COMET の仕様に変更になった[1]。そこで、新たに教育用プロセッサとしての COMET-II 互換プロセッサを開発する。また、より良い演習環境の構築の為に演習用ボードの開発を行った。

本稿では、仮想プロセッサ COMET-II を教育用マイクロプロセッサとして VHDL により実現する。その際必要となる仕様上の変更点、演習環境の実現方法について述べる。また、プロセッサを実装した演習用ボードの開発およびそれを用いて学部演習を行い、CPU 設計演習環境としての評価を行う。

### 2. COMET-II 互換プロセッサの開発

#### 2.1 COMET-II アーキテクチャ

COMET-II の基本仕様は 1 語 16 ビット、主記憶容量は 65536 語、命令語長は 1 語長又は 2 語長である。汎用レジスタ (GR) は 8 個で、スタックポインタ (SP)、プログラムレジスタ (PR)、フラグレジスタ (FR) を持ち、基本命令は 28 種類である [1]。

教育用マイクロプロセッサである為には、構成が簡素で理解しやすいアーキテクチャでなければならない。また、演習者が内部を容易に変更できるようにする為に、VHDL により設計を行い FPGA 上に実装を行った。

COMET-II の仕様は、COMET に比べレジスタの個数や命令の種類が増加しているため、回路の実現を従来より効率的にするべきである。そこで、設計上以下のような工夫を行った。

- COMET-II アーキテクチャでは、図 1 に示すようなアーキテクチャ構成とした。特徴としてはバスの利用により実行ステップ数を減らしている。
- VHDL の記述において必要のないレジスタや変数を全て削除し、演習者への配慮をしている。

- VHDL に階層を設けることで、演習でのプロセッサの改造を効率よく行えるようにした。特に実行ステップ数については、従来我々が開発した COMET アーキテクチャ[2]が平均 10.5 ステップであるのに対し、COMET-II では平均 4.2 ステップまで減らす事ができた。

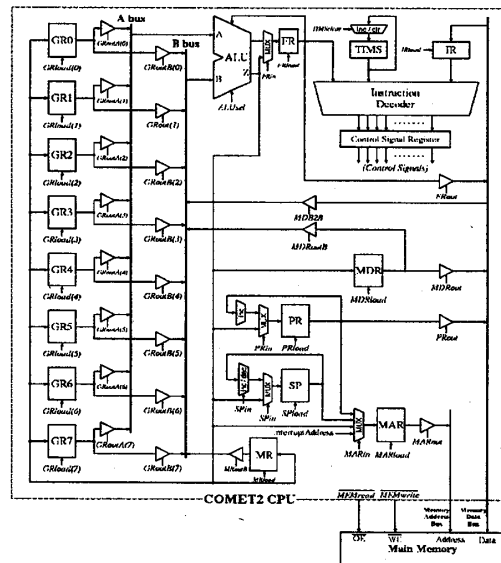


図 1 COMET-II アーキテクチャ

#### 2.2 割り込み機能の追加実装

COMET-II では割り込みに関する明確な仕様は定義されていないので、機能として追加した。割り込み要求は 8 本とした。割り込み要因はマスク可能であり、割り込みが受け付けられると割り込み要因に対応した割り込みベクタを求める。命令の開始時に割り込みは検査され、割り込み許可状態であるならば GR0, FR, PR をスタックに退避後割り込み禁止状態にする。割り込みベクタで指定された番地にあるアドレスに分岐すると共に、受け付けた割り込み要因に対して応答を返し割り込み信号をクリアする。

#### 2.3 機能追加による命令拡張

割り込み機能の追加、OS の実装等を行い演習用プロセッサとして実際に実機上で実装する為に、COMET-II 命令セットに以下の拡張を行った。

- 割り込み許可 (EI)、禁止 (DI) 命令
- 割り込み処理ルーチンからの復帰命令 (RETI)
- マスク読み込み (LDM)、書き込み (STM) 命令
- スタックポインタ読み込み (LSP)、書き込み (SSP) 命令
- スーパーバイザーコール (SVC)

特にスタック操作命令はシステム起動時におけるスタックポインタの初期化、OS の実装の為に必要であった。

† 武蔵工業大学, Musashi Institute of Technology

SVC 命令については COMET の仕様変更により仕様が以下のように改定になった[1].

**COMET** GR0,FR,PC をスタックに退避後 6 番地で示されるアドレスに分岐する.命令実行後,直ちに割込み禁止状態にする.

**COMET-II** 実効アドレスを引数として割込みを行う.レジスタ,FR の値は不定となる.

実効アドレスはスタックに積むことにした. GR0,FR,PR は外部割込みと同様にスタックに退避する.そして RETI 命令によってそれらを復帰するようにした.仕様上はレジスタ,FR は不定になる事とした.

### 3. 演習環境の開発

#### 3.1 COMET-II ボードの開発

VHDL で設計した COMET-II を実装する為のボード(図 2)は,64K 語のメモリを搭載する SRAM ボード(写真上),BIOS を搭載した ROM ボード(写真左),COMET-II を実装する FPGA ボード(写真中央),後述するモニタプログラム等を操作する為に主に使用する BIOS console ボード(写真下),ユーザ用キーボードや 7 セグメント LED を搭載したユーザ console ボード(写真右)の 5 枚の機能ボードから構成される.ROM ボード以外の各ボードは三菱電機マイコンソフトウェア株式会社製 PowerMedusa[3]を使用し,ROM ボードは本研究で開発した.機能ごとにボードを分ける事で,演習者がプロセッサやメモリ,その他の I/O との関係を視覚的に理解することができるようになる.ボード上には 7 セグメント LED やブザー,シリアルポート等の様々な入出力装置が搭載されている.演習ではこれらを使用してブザーによりメロディを流したり,シリアル通信によるロボット制御等の演習を行うことができる.COMET-II ボードでは入出力制御方式にメモリマップド I/O を採用しており,#FF00 番地から約 20 個の入出力装置に対し I/O アドレスが割り当てられている.こうする事で COMET-II の既存のロードストア命令により入出力制御を行うことができる.

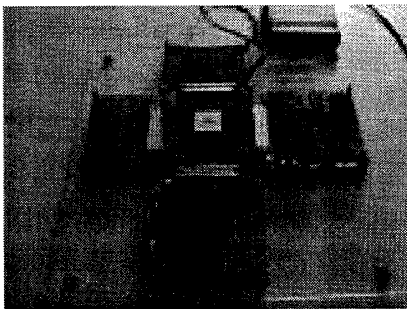


図 2 COMET-II ボード

#### 3.2 BIOS の開発

COMET-II BIOS は上述の ROM ボードに実装され,ボードが起動する際に起動プログラムにより SRAM へ転送される仕組みである.転送後は完全にプロセッサと SRAM 間での動作とすることが可能となる.

BIOS は主に以下のような処理を行う.

- スタックポインタ,システム用変数の初期化
- 割込みに対する処理ルーチンの選択や制御
- 機能キーを使用したモニタプログラム

#### ● BIOS ファンクションコール

モニタプログラムとは,20 個のキーボードを押すことによるキー割込みによって起動されるプログラムであり,演習用として以下の機能がある.

- 指定番地からのプログラム実行
- 指定番地のメモリ内容の参照
- 指定番地へのデータ書込み
- 指定番地からのシングルステップ実行
- レジスタ値の参照

COMET-II には IN 命令,OUT 命令という入出力装置を制御する為のマクロ命令が定義されている.COMET-II ボードでは小型ロボットの制御を行うので,IN 命令 OUT 命令によってシリアルポートを利用した通信が行えるようにした.

#### 3.2 サポートツールの開発

サポートツールとして,CASL2 アセンブラ及びプログラム転送ツールを開発した.プログラムは PC 上でアセンブリされ,シリアル通信によってボード上のメモリへ転送される.

### 4. CPU 設計演習環境としての評価

COMET-II ボードを使用した演習を行った.演習内容は以下の通りである.

1. 小型ロボットの制御
2. ロジックアナライザによるバスの実波形観測
3. シミュレータによる COMET-II の理解
4. CAD ツールを使用した COMET-II のアーキテクチャ改造及び命令セット拡張

1,2 の演習では主にプロセッサを外側から見た場合の動作理解を目的とした演習である.3,4 の演習は CPU 内部の動作の理解を目的とした演習である.プログラムのアセンブル,ボードへの転送,プログラムの実行及び確認等はスムーズに行うことができた.また,VHDL によるプロセッサの命令セットの追加変更においてもアーキテクチャの簡素化及びソースの階層化により理解が容易になった.従来の演習課題では掛け算命令の追加,レジスタ間演算の追加,レジスタの追加[3]であったが,今回はさらに PR 相対分岐命令を課題として与えた.しかし,ほとんどの班が課題を達成することができ,満足のいく結果であった.

### 5. 今後の課題および展開

本研究により COMET-II ボード上での CPU 設計演習環境が実現された.演習者は,実際のハードウェアを扱う事で十分な理解を得ることができたと言える.しかしデバッグ環境が充実していない点が今後の課題としてあげられる.今後デバッグ環境を構築し,リアルタイム OS を実装すると共に OS の演習ができる環境の構築を目指している.

#### 参考文献

- [1] 財団法人日本情報処理開発協会情報処理技術者試験センター:情報処理技術者試験案内書・願書,(2001)
- [2] 吉沢,尾崎,片山,石川,宮内:COMET 互換プロセッサによる設計演習環境の提案と実現,情報処理学会全国大会,1G-10,(1997)
- [3] 三菱電機マイコン機器ソフトウェア株式会社:PowerMedusa ユーザーズマニュアル,(2002)