

ASIC 設計手法を用いたプロセッサのビット幅拡張と並列処理による高速化

鈴木 貴斗† 本田 越次†† 宮崎 光平†† 江口 一彦† 五島 敬史郎†
 †愛知工業大学大学院 電気電子工学専攻 ††愛知工業大学 電子情報工学専攻

1. はじめに

半導体業界には半導体チップの集積度は 18 ヶ月で 2 倍になるというムーアの法則が 1 つの指標としてある。この指標をもとに半導体業界の技術は発達し微細かつ複雑な集積回路を製造できるようになってきた。一般に ASIC の手法では開発費用と開発期間を多く必要とする。そのため、それより短期間で開発できる FPGA が開発され普及してきた。しかし FPGA は大規模になるにつれ、消費電力や費用が増大するという欠点があるため、最近 ASIC が見直されている。その理由として、集積回路の動作速度向上により FPGA で扱うことのできないアナログの要素が必要になったことや、設計ツールの発達により開発コストが低減されたこと、回路の最適化を行うことで動作速度が速くなり、消費電力の低減が可能ということが挙げられる。

このような利点があることから本テーマでは ASIC の設計手法を用いて昨年本研究室で作成した CPU をベースにビット幅拡張と並列処理の機能を追加し設計を行う。

2. ASIC 設計手法

本研究室では大規模集積システム設計教育センターを通して各設計ツールを入手し設計を行った。また製造に関してはフェニテックセミコンダクター社の 0.6 μ m のプロセスで行う。以下に設計工程と各工程で使用するツールを示す。

設計工程	使用ツール
RTL設計	IcarusVerilog
構造設計	DesignCompiler (synopsys社)
物理設計	IC Compiler (synopsys社)
	α -SX Ismo(jedat社)
動作検証	Stylus(Inovys社)

表 1 設計工程及び使用ツール

3. 設計対象プロセッサ

設計対象は昨年本研究室で作成した 4 ビット CPU

をベースとし、そこにビット幅を 8 ビットに拡張しパイプライン処理の機能を追加して設計を行う。

3.1 ビット幅拡張

4 ビットでは扱うことのできるデータ幅や実行できる命令数に限りがある。そのため、ビット幅の拡張を行う。

3.2 パイプライン処理

昨年本研究室で作成した 4 ビット CPU は 1 クロックで 1 つの命令を実行する逐次方式であったが、それでは複数の命令を処理する場合時間がかかる。それを改善するためパイプライン処理の機能を追加を行う。これにより各命令の実行時間は変わらないが、スループットが向上するため全体的な処理の実行が速くなる。各命令は実行順に、命令フェッチ(以下 IF)、命令デコード(以下 ID)、命令実行(以下 EX)、メモリアクセス(以下 MA)、ライトバック(WB)と 5 つの処理に分けられる。以下にパイプライン処理と逐次方式の時間遷移と実行処理との関係の概要を示す。

クロック	1	2	3	4	5	6	7	8	9	10
命令1	IF	ID	EX	MEM	WB					
命令2		IF	ID	EX	MEM	WB				

図 1 パイプライン処理の時間遷移と実行処理

クロック	1	2	3	4	5	6	7	8	9	10
命令1	IF	ID	EX	MEM	WB					
命令2						IF	ID	EX	MEM	WB

図 2 逐次処理の時間遷移と実行処理

4 課題

現在は設計工程中の仕様設計が終わり VerilogHDL を用いて RTL 設計を行っている段階である。パイプライン処理機能の追加は各制御信号伝達のためのレジスタの追加や、ハザードの対策など昨年までとは処理の仕方が変わるためそれらに対応させることが課題である。

参考文献

[1] David A. Patterson, John L. Hennessy “コンピュータの構成と設計 上 –ハードウェアとソフトウェアのインターフェース”日経 BP 社, 2011

[2] 岩出秀平, 清水徹 “実用プロセッサ技術” ムイスイ出版株式会社, 2009