

### 各種演算機能をもたせた $0.18\mu\text{m}$ CMOS VLSI の設計と試作評価

藤田, 実 / SHIMIZU, Naoki / 米本, 友紀 / YONEMOTO,  
Tomonori / 青木, 勇樹 / AOKI, Yuki / 清水, 直樹 /  
Fujita, Minoru

---

(出版者 / Publisher)

法政大学工学部

(雑誌名 / Journal or Publication Title)

法政大学工学部研究集報 / 法政大学工学部研究集報

(巻 / Volume)

41

(開始ページ / Start Page)

11

(終了ページ / End Page)

14

(発行年 / Year)

2005-03

(URL)

<https://doi.org/10.15002/00003758>

# 各種演算機能をもたせた 0.18 $\mu$ m CMOS VLSI の設計と試作評価

## DESIGN OF 0.18 $\mu$ m CMOS VLSI WITH VARIOUS ARITHMETIC FUNCTIONS AND ITS EVALUATION

青木勇樹\*, 米本友紀\*, 清水直樹\*, 藤田 実\*\*

Yuki AOKI, Tomonori YONEMOTO, Naoki SHIMIZU and Minoru FUJITA

To cultivate the practical design skill of integrated circuits, very fine 0.18  $\mu$  m technology CMOS VLSI was designed, fabricated and evaluated. Using the hardware description language, various arithmetic logic units were designed and implemented to VLSI. The measurement of its operation was very informative to understand the characteristics of VLSI.

*Key Words* : HDL, Verilog, ALU, VLSI

### 1. はじめに

集積回路は現在のあらゆる電子システムの中核となって使われている。集積回路の知識を深め、創造的な技術力を養成するには、実際に VLSI を設計し、試作された VLSI を評価するという実践的な実験と研究をすることが非常に重要である。大学における VLSI 研究をサポートする体制としては、東京大学大規模集積システム設計教育研究センター(以下 VDEC と略)が設立されており、著者等はこれに参加して研究を行うことにした[1]。

実際に集積回路の設計と評価を行うには、実現したい機能を記述する言語の習得から始まり、広い分野にまたがる各種の設計ツールを選択して使い方をマスターしなければならない。本論文は、集積回路の機能設計から特性評価まで一貫して体験するのを可能にする設計フロー確立のための検討と各段階で行った研究及び実験の内容を報告するものである。

### 2. 目標機能の設定

今回の VLSI 設計は初めての試行であるので、機能が明確で確実に評価できるものを対象とすることが望ましい。このため、加算器を取り上げ、標準的なものを初めとして、著者等が考案した独自の機能を持つものをそれぞれが担当して設計した。以下に設計対象にした 4 種の加算器の機能を示す、

#### (1) 標準的な加算器

4ビットの2進数入力 a と b の加算を行い、結果 s を 5ビットで出力する。

#### (2) 小数点を持つ数値の加算器

入力する数値を 10 進数 2 桁とし、小数点の位置は任意とする。異なった小数点位置を持つ数値同士の加算を可能にする。結果が 3 桁以上になった場合は上位 2 桁のみを表示する。

#### (3) メモリ付き加算器

電卓のようなメモリ機能を持つ加算器を作る。通常の加算もでき、加算した値をメモリに保存し、スイッチ一つで呼び出せるようにした。メモリから値を呼び出し、次の入力した値との加算も出来るようにした。

#### (4) フィボナッチ数の計算器

1 から始まり、入力スイッチを押す毎にフィボナッチ数を計算して出力する。演算は 10 回まで行えるようにした。

今回設計する VLSI の信号ピン数は 64 ピンに限定されており、誤動作防止のため 4 つの機能間でピンを共有させないようにしたので、扱える桁数を 2 桁に限定した。しかし、これで目標とする機能の確認は出来るし、今後必要があれば、桁数を拡張すればよい。

### 3. VLSI の設計フローと内容

最初にターゲット技術を何にするか決める必要がある。VDEC には幾つかの技術が用意されているが、今回は日立の 0.18 $\mu$ m 技術を選んだ。VLSI の設計技術を一貫して体験するだけが目的であれば、必ずしも最先端技術を使う必要はないが、最先端加工技術の VLSI の性能を体験することは、今後の高性能な VLSI 設計の研究につなげるのに非常に役に立つと考えた。論理規模は小さいので、チップは 2.8mm 角のものにした。動作電圧は 1.8V である。

VLSI 設計の全体フローを図 1 に示した。以下に各段階で実行した内容を記す。

#### (1) 機能記述

機能記述言語には VHDL と Verilog HDL があるが、C 言語に近く取り扱いが容易な Verilog を選択した[2][3]。

\* 大学院情報電子工学専攻

\*\* 電子情報学科

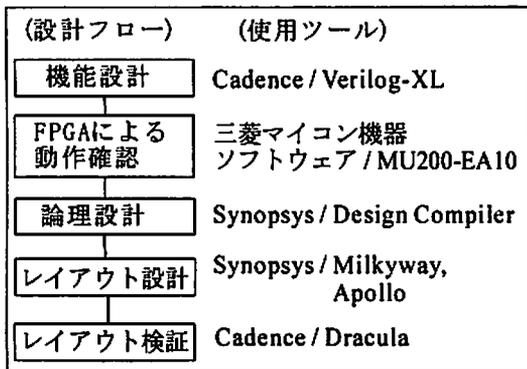


図1 VLSI設計の全体フロー

```

トップモジュールの記述
module TOP4(IN_f_a, IN_f_b, OUT_f_s, IN_a_ChipIn, CLK_a, RST_a,
  OUT_a_H, OUT_a_L, IN_y_in, IN_y_clk, IN_y_CE, IN_y_rst,
  OUT_y_hd, OUT_y_ld, IN_s_in, IN_s_clk, IN_s_rst,
  IN_s_plus, IN_s_m_in, IN_s_m_out, OUT_s_SEGH, OUT_s_SEGG);
  input CLK_a, RST_a, IN_y_in, IN_y_clk, IN_y_CE, IN_y_rst,
  IN_s_clk, IN_s_rst, IN_s_plus, IN_s_m_in, IN_s_m_out;
  input [3:0] IN_f_a, IN_f_b, IN_a_ChipIn, IN_s_in;
  output [3:0] OUT_s_SEGH, OUT_s_SEGG;
  output [4:0] OUT_f_s, OUT_a_H, OUT_a_L;
  output [6:0] OUT_y_hd, OUT_y_ld;
  adder I1(OUT_f_s, IN_f_a, IN_f_b);
  segouttop I0(.ChipIn(IN_a_ChipIn), .CLK(CLK_a), .RST(RST_a),
  .ChipOut0(OUT_a_L), .ChipOut1(OUT_a_H));
  fibotop I2(.y_in(IN_y_in), .CE(IN_y_CE), .hd(OUT_y_hd),
  .ld(OUT_y_ld), .y_clk(IN_y_clk), .y_rst(IN_y_rst));
  top I3(.in(IN_s_in), .SEG_H(OUT_s_SEGH),
  .SEG_G(OUT_s_SEGG), .clk(IN_s_clk), .rst(IN_s_rst),
  .plus(IN_s_plus), .m_in(IN_s_m_in),
  .m_out(IN_s_m_out));
endmodule

インスタンス I1 の adder の記述
module adder(s, a, b);
  output [4:0] s;
  input [3:0] a, b;
  assign s = a + b;
endmodule
    
```

図2 Verilogによるチップの階層記述

ソフトウェアはCadence社のVerilog-XLを用いた。チップは階層設計とし、各著者がそれぞれの機能をインスタンスとして設計し、トップレベルでまとめた。図2に最上階層と一番論理量の少ない標準的な加算器 adder のインスタンス記述を示す。動作機能はVerilogのシミュレーションで確かめた。

(2) FPGAによる動作確認

通常のVLSI設計ではこの工程はなく、上項の機能シミュレーションで済ましている。著者等はこれまでHDL記述の確認にFPGAボードを用いており、記述通りに正しく動作するかFPGAボードで確かめれば実感が得られるし安心にもなる。いわゆるラビッドプロトタイプングとして使った[4]。

表1 論理合成の結果

インスタンス	セル数	面積	最大遅延時間
I0: segouttop	414	25,283	4.20 ns
I1: adder	5	492	0.96 ns
I2: fibotop	270	16,589	5.07 ns
I3: top	322	18,900	4.90 ns

(3) 論理合成

設計ツールにはSynopsys社のDesign Compilerを用いた。初めての設計であり、タイミング情報を持ち合わせていないため、ぎりぎりのタイミングにするのではなく安全サイドの値でクロックや遅延等の設定をした。論理合成後の各機能ブロックのゲート数や最大遅延時間を表1に示す。

(4) レイアウト設計

レイアウト設計用のツールはデータベースの作成と配置配線から成り立っており、それぞれにSynopsys社のMilkywayとApolloを用いた。

論理合成後のゲート数は表1のように少ないので、2.8mm角のチップでも十分な余裕があり、むしろ空白部をどの様に埋めるかが課題となった。

スタンダードセル方式を採用し、それぞれの機能はブロックとして完全に独立させ、チップ上の別々の領域を指定し、そこにセル敷き詰め方式で配置した。これにより、ブロック相互の混同による配置配線の間違いを避けた。残りの部分はフィラーセルを埋め込んだ。

(5) レイアウトの検証

レイアウト後の検証としては、結線チェック(LVS)とパターンチェック(DRC)が必要である。ツールとしてはCadence社のDraculaを用いた。

実際に検証をしてみると、予想してなかったことが指摘された。それは、high levelとして論理値の1を用いて記述している箇所があったが、レイアウト段階ではそれが処理されていなく、検証でエラーとして指摘されることが判った。これについては、1の代わりにV<sub>DD</sub>と記述し直すことで、再度、配置配線を行って対策した。

(6) VDECへのデータ送付と試作依頼

設計データはVDECに送付しチップ試作を依頼した。出来上がったチップはVDECから送付してもらった[5]。

4. 試作結果と評価

(1) VLSIの試作品

完成したVLSIの外観を図3に示す。QFPの160ピンパッケージに収納されている。

チップの外観は図4の通りである。先にも述べたように、それぞれの機能はブロックとして、チップの離れた4



図3 完成したVLSI

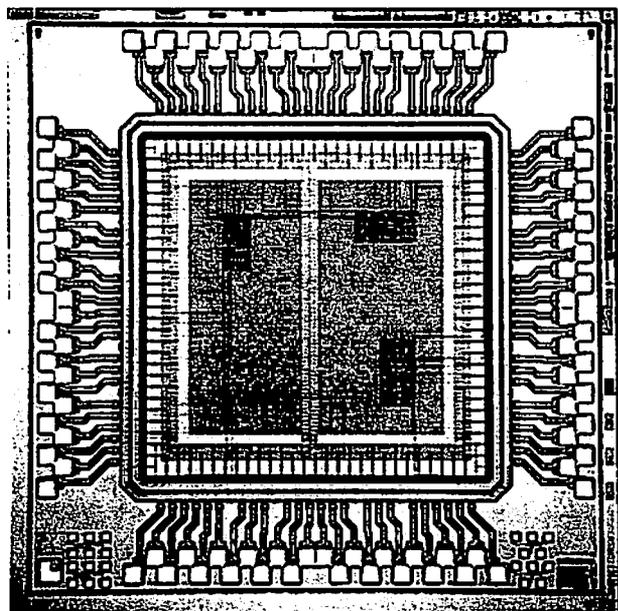


図4 チップの外観写真

箇所に配置してある。各設計部分を拡大して示したのが図5である。

(2) 動作評価

論理回路としての動作評価はデータジェネレータ、ロジックアナライザ、オシロスコープを用いて行った。消費電流はデジタルマルチメータを用いて測定した。

加算器の動作は表2のようになるはずで、表に示す入力波形 a, b をデータジェネレータで生成してVLSIに印加した。VLSIを1.8Vで動作させ、入出力波形をロジックアナライザで解析した結果は図6の通りであった。期待通りの動作をしていることが判る。入力 a, b の最下位ビットだけをを入力し、出力 s の下位2ビット波形をオシロスコープで観測したものを図7に示す。ここでも期待通りに動作していることが確認された。

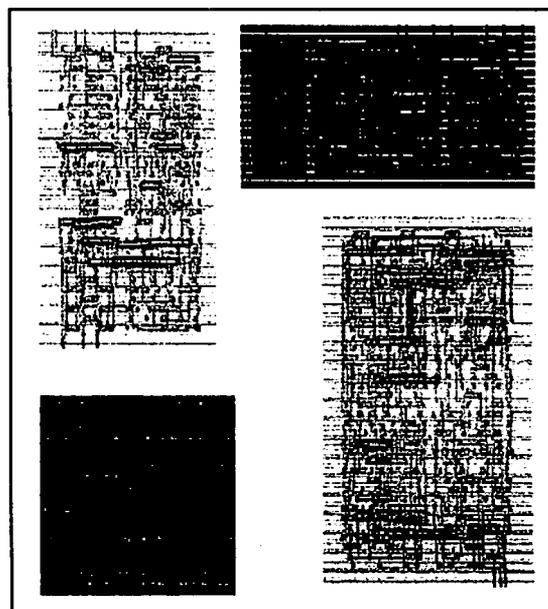


図5 各設計部分の拡大図

インスタンス(モジュール)名  
 右下: segouttop, 左下: adder,  
 右上: fibtop, 左上: top

表2 加算器の動作

	10ns	20ns	30ns	40ns	50ns	60ns	70ns	80ns	90ns	100ns
a[3:0]	0	1	1	3	3	A	A	C	C	5
b[3:0]	0	0	2	2	4	4	8	8	D	D
s[4:0]	0	1	3	5	7	14	18	20	25	18

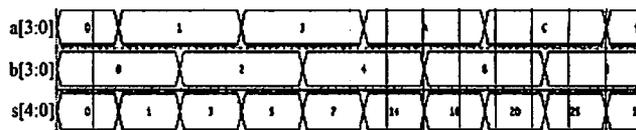


図6 ロジックアナライザによる動作波形解析結果

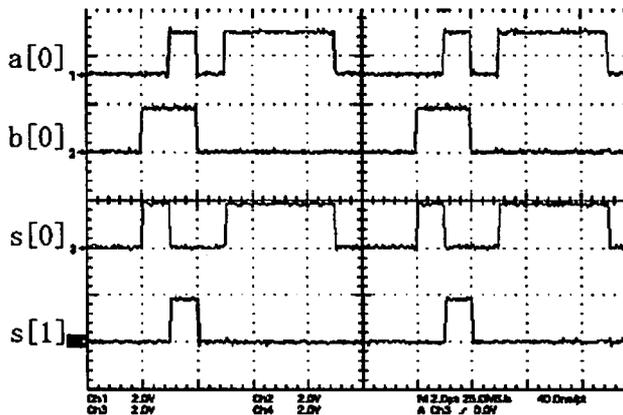


図7 オシロスコープによる動作波形観測結果

次に、消費電力を調べた。最初にスタティックな消費電流( $I_{stat}$ )を測定したが、結果は図8の通りで、電圧と共に消費電流がかなり増加することが判った。サブスレッショールド電流以外に、一部論理回路にショートがあるため、電流が増加した。これは設計ソフト間で大文字、小文字の取り扱いの違いがあるため発生したものである。

ダイナミック消費電流については、まず、電源電圧と、クロックおよび入力信号の周波数を変えて電流を測定した。ここで得られた電流は全電流( $I_{total}$ )であるため、以下の計算でダイナミック消費電流( $I_{dyn}$ )を求めた。

$$I_{dyn} = I_{total} - I_{stat} \quad (1)$$

結果は図9の通りであった。CMOS回路のダイナミック消費電流は、遷移確率を $\alpha$ 、負荷容量を $C_L$ 、周波数を $f$ とするなら、理論的には次式で表される。

$$I_{dyn} = \alpha C_L f V_{DD} \quad (2)$$

図9を見ると、周波数に対し消費電流が線形で増加しており、ほぼ理論通りの特性を示した。ただし、0.7Vの特性は直線からずれている。波形を観測したところ、遅延時間が大きくなったため、100MHzでは正常動作をしておらず、これが原因になっていた。100MHzで0.9V以上の領域では、電流は(2)式の通り電源電圧に線形に比例するのが確かめられた。0.8V以下では電流は(2)式よりも低くなっていく。これは応答速度が遅くなり、フルスイングで $C_L$ の充放電が行われなくなるためと考えられる。

### 5. 考察

今回の研究で非常に多くの設計ツールを使ったが、それに伴って課題があることが判った。以下の項目の重要度は高く、今後の設計で改善する必要があるのが判った。

#### (1) 精度の高いタイミングの検証

高速化のための、セットアップやホールドタイムの検証、クロックツリー生成とスキューの見積もり、バックアノテーションによる検証精度向上が今後求められる。

#### (2) フィラセルの改善

ライブラリに用意されたフィラセルにはゲートパターンがないため、ゲート密度が非常に低くなってしまった。今後は独自にフィラセルを設計して使っていくことが必要である。

#### (3) カスタム設計力

必要とするセルを自分で設計してライブラリに登録して使用するか、人手で任意の場所に配置配線する等、より高度な設計が出来るようにする必要がある。

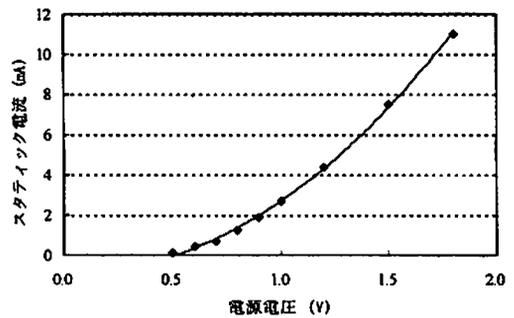


図8 スタティック消費電流特性

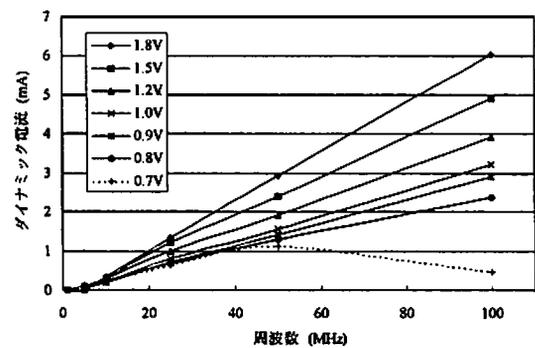


図9 ダイナミック消費電流特性

### 6. おわりに

研究を通して、VLSIの機能設計から試作品の評価まで全工程を体験出来た。実際に自分の考案したものを作ることができ、実践力の養成になった。しかし、やってみると、さらに学ぶべき分野あり、掘り下げるべき技術が多いのも判った。これ等については、今後も研究を続けて行くつもりである。

謝辞：本チップ試作は東京大学大規模集積システム設計教育研究センターを通し、株式会社日立製作所および大日本印刷株式会社の協力で行われたものである。ここに感謝の意を表します。

#### 参考文献

- 1) 藤田実, 佐藤和善: 大学における VLSI の研究と産学協同について, 法政大学工学部研究集報, No.39, pp.17-21, 2003
- 2) 深山正幸, 北川章夫, 秋田純一, 鈴木正國: HDL による VLSI 設計, 共立出版, 2002
- 3) 浅田: デジタル集積回路の設計と試作, 培風館, 2000
- 4) J.O.Hamblen, M.D.Furman: Rapid Prototyping of Digital Systems, Kluwer Academic Publishers, 2001
- 5) VDEC ホームページ: [www.vdec.u-tokyo.ac.jp/](http://www.vdec.u-tokyo.ac.jp/)