Vol. 35 No. 6　　情　報　処　理　　June 1994

解　説

IPS

FPGA—その現状，将来とインパクト

1. FPGA の 現 状 と 将 来†

身　次　　茂††

1. はじめに

書換え可能なゲートアレイ (FPGA : Field Programmable Gate Array) は，ユーザの手元でディジタル回路をプログラミングすることのできる LSI であり，近年その技術的な発展が著しく，市場規模は専用目的 IC (ASIC : Application Specific Integrated Circuit) の 30% に達している[1]．FPGA は，古くから用いられてきたプログラミング可能なディジタル IC である PLD(Programmable Logic Device) に比べ，はるかに大規模の回路を実現することができる．現在，最大規模で 15000 ゲートを搭載できるチップが最大 250 MHz (内部フリップフロップの最大動作周波数) のクロックで動作し，小規模なマイクロプロセッサの実現が可能である．その柔軟性と手軽さは，ハードウェアの試作の用途を越えて，計算機のアーキテクチャの分野にも大きなインパクトを与えている．

本稿では，FPGA を従来の ASIC の主流であるゲートアレイと比較しながら紹介し，その現状と将来についてまとめる．

2. FPGA はプログラマブル

「IC をプログラムできる」とはどういうことであろうか．われわれが購入する IC は，CPU であったりメモリであったり，いずれにせよ機能は決まっている．それに対して IC 購入時は機能が白紙で，ユーザが欲する機能をユーザ自身で作り込める IC を**プログラマブル**と言っている．それを CPU などの汎用 IC と，今までの ASIC の代表であるゲートアレイとの入手方法の違いについてみてみよう．

CPU，メモリなどの汎用 IC は，出来合いのものを IC メーカから買ってくるだけである（**図-1** 参照）．ユーザが希望する機能の IC を売っているとは限らない．

ゲートアレイの場合はユーザの希望に沿った IC を作ることができる（**図-2** 参照）．まず，ユーザが必要な機能あるいは回路をデザインセンタに提示する．デザインセンタは回路図をもとに IC の配線のためのマスクパターンを設計し，マスクパターンデータを工場に渡す．工場はそれに基づき IC を製造してユーザに渡す．しかし，IC を入手した後で回路を変更しようとしても，同じ手順を踏まねばならず，時間がかかる．ユーザの手元での回路変更はできないのでプログラマブルとは言わない．

それに対して FPGA では，**図-3** に示すように

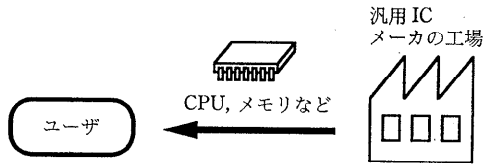


図-1　汎用 IC の購入

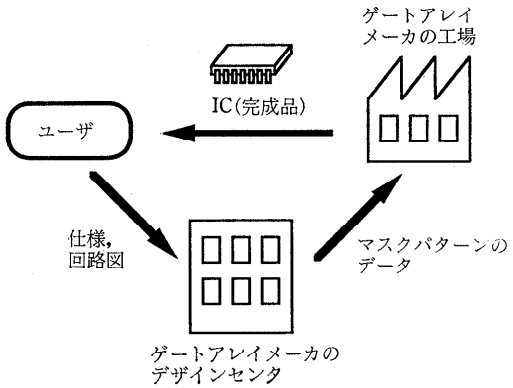


図-2　ゲートアレイを受け取るまで

† The Present and Future of FPGA by Shigeru MITSUGI (Applied Electronics Research Center, Kumamoto Technopolis Foundation).

†† 熊本テクノポリス財団　電子応用機械技術研究所

ユーザの手元でICを完成させる．まず，ユーザは半完成品の IC を FPGA メーカから購入する．それから FPGA 専用 CAD に回路図を入力し，配線パターンを CAD に計算させる．配線パターンが決まると，FPGA専用の配線パターン書込み器(FPGA 書込み器) を用いてIC内部の配線を行い，IC が完成する．回路を変更する場合は配線の消去を行い，上記の工程を繰り返す．ユーザの手元で回路を変更できるので，プログラマブルと言える．FPGA のようなプログラマブル IC はやり直しがきくという**気楽さがある**．

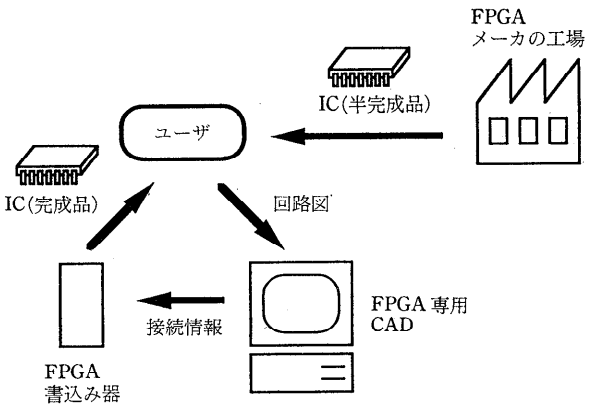


図-3　FPGA の完成まで

3. FPGA とゲートアレイの構造の違い

ゲートアレイの構造からみてみよう．**図-4** に示すように，ゲートアレイには **MOS—FET** が NAND や NOR などのゲートを作りやすい配置に数千から数万個並んでいる．この部分の構造はユーザが変更することはできない．ユーザが指定するのは MOS—FET 間の配線である．ユーザが回路図をデザインセンタに渡すと，デザインセンタでは MOS—FET 間の配線を決定するマスクパターンを設計し，工場でマスクパターンに従ってICの製造を行い，IC の完成品がユーザに届く．ユーザは回路図のかたちで MOS—FET 間の配線を間接的に指示していることになる．MOS—FET が数個が集まって，AND や OR，フリップフロップなどのゲートを実現する．ゲートがたくさん集まって，演算器やレジスタ，メモリそして CPU などができる．

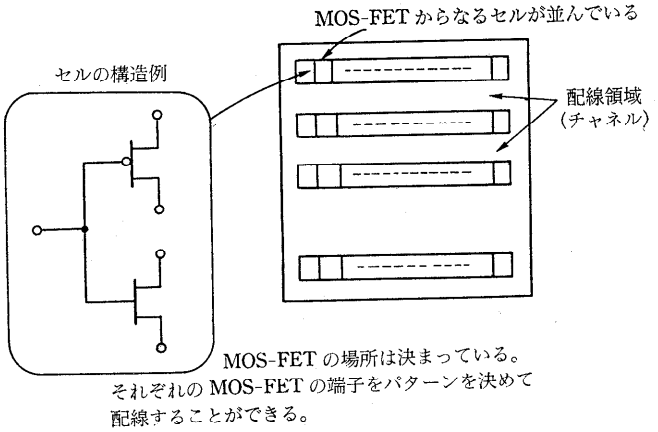


図-4　ゲートアレイの構造

FPGA では規則正しく並んでいるのは，AND や OR などのゲートからなる組合せ回路や，フリップフロップなどを実現することのできる論理ブロックである．論理ブロックの実現例を**図-5** に示す．論理ブロックはルックアップテーブルとセレクタ，フリップフロップからできている．このルックアップテーブルの内容やセレクタの接続のための情報は，論理ブロック内に配置されているヒューズや ROM あるいは SRAM の各 bit によって保持される．外部からこの内容を設定することにより，論理ブロックの機能を「プログラム」することができる．後に述べるように，FPGA の特性は，この機能情報を保持するための素子の種類によって大きな影響を受ける．

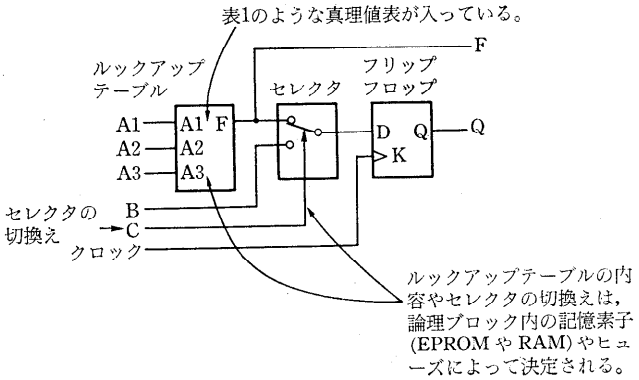


図-5　論理ブロック

ルックアップテーブルは**表-1** の真理値表で示すように，入力 A1～A3 の組合せに対して出力 F の 0/1 をそれぞれに定義でき，AND や OR など自由な論理を設定できる．ルックアップテーブルの出力は，論理ブロックの出力F およびフリップフロップのデータ入力となる．セレクタはフリップフロップのデータ端子Dに入れる信号を，入力Cにより選択する．C＝0 であればルックアップテーブルの出力Fを選択し，C＝1 であれば信号Bを選択する．

「FPGA はこのような論理ブロックが**図-6** に示すように，チップ中に配列している構造をもつ．図-4 で示すゲートアレイの構造と比較されたい．このような論理ブロックどうしをどのように結合して複雑な演算器や CPU を構成するのだろうか．ゲートアレイの場合，配線はマスクパターンによって決まるため，配線領域内に納まる限りは，自由なパターンで配線ができる．これに対して FPGA では，論理ブロック間の配線はあらかじめパターンが引いてあり，指定できるのは配線の交差する点の接続のあり／なしだけである．この交差点の接続をクロスポイントスイッチやスイッチマトリックスを介して設定する．図-6 では，論理ブロック A の端子 Q と論理ブロック E の端子 A2 を結んでいる．クロスポイントスイッチはその名のとおり，配線の交差する点にスイッチを設けたもので，**図-7** の例ではスイッチ 7 を閉じることにより端子 A3 から端子 D1 へ信号を通す．信号の方向は逆でもかまわない．スイッチのオン／オフは，記憶素子をスイッチの制御端子に直接つないで制御する．記憶状態が“1”であればスイッチを閉じ，“0”であればスイッチを開ける．スイッチマトリックスは，任意の端子とそれ自身の他のすべての端子をスイッチを介して結んだものである．**図-8** では端子 B1 からの接続のみを示し，他は省略している．

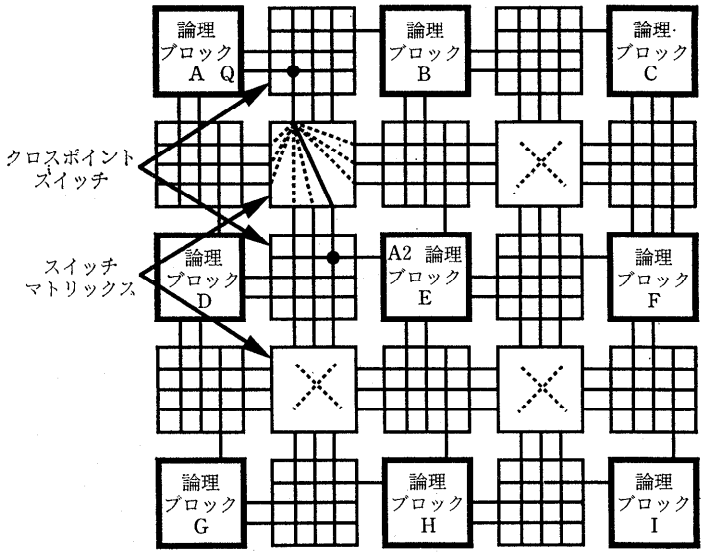


図-6 FPGA の構造例

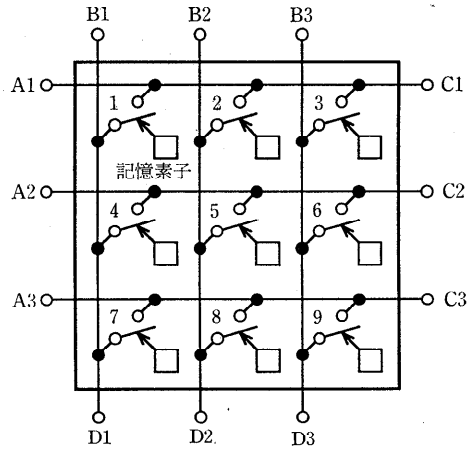


図-7 クロスポイントスイッチ

表-1 ルックアップテーブルの真理値表

A1	A2	A3	F
0	0	0	0/1
0	0	1	0/1
0	1	0	0/1
0	1	1	0/1
1	0	0	0/1
1	0	1	0/1
1	1	0	0/1
1	1	1	0/1

このクロスポイントスイッチやスイッチマトリックスの接続は，先ほど述べた機能ブロック内のルックアップテーブルやセレクタの設定情報同様にスイッチ内に配置されたヒューズや ROM あるいは SRAM の各 bit によって保持され，外部か

ら内容を設定することにより機能ブロックの設定も「プログラム」することができる．

4. FPGA の種類

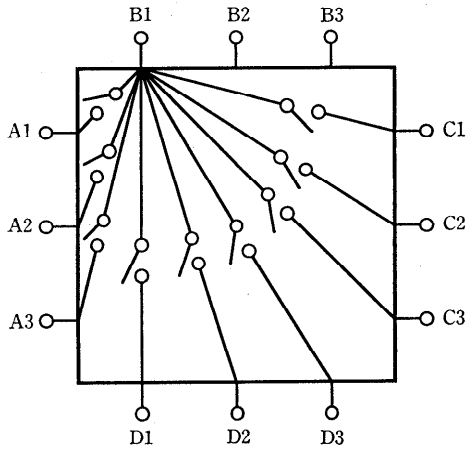
FPGA を分類するときの基準となるのは，まず第一に論理ブロック構造の違いである[2)~4)]．論理ブロックは，上述のようにルックアップテーブルとセレクタとフリップフロップでできているものもあるが，もっと単純に AND や OR だけでできているものもある．この場合，フリップフロップは，AND や OR を組み合わせて作る．逆に論理ブロックの構造がもっと複雑で，一つの論理ブロックに数十の AND や OR そして 10 前後のフリップフロップが入っていることもある．ここではこの論理ブロックの構造の違いを**粒度**という尺度で考える．粒度は論理ブロックの大きさである．この粒度の違いが，いろいろなアプリケーションでの各 FPGA の得意，不得意を生むことになる．

次に，FPGA の特性は，論理ブロックの機能や論理ブロック間の接続情報をどのような素子によって保持しているかによって，大きく影響を受ける．素子には，**表-2** に示すように，ヒューズ，EPROM（書変え可能な ROM），EEPROM（電気的に消去可能な EPROM），SRAM がある．

ヒューズを用いる方法は，一度切ったら元に戻らないため，再書込みができないがスイッチでの遅延が最も小さい．この場合，スイッチそのものがヒューズになっている．EPROM や EEPROM を用いる方法は，スイッチの ON/OFF をメモリの 1 bit がそれぞれ制御するようになっている．接続などの「プログラム」を消去し，再書込みすることが可能であり，また，電源を切っても IC の接続が消えることはない．しかし，消去や書込みには専用の書込み器を必要とする場合が多い．これに対し，SRAM で保持するタイプは電源オフのたびにチップ内部の SRAM のデータが消えてしまうため，結果として論理ブロックの機能やブロック間の接続が消えてしまうことになる．このため，チップの外部に不揮発性の ROM を設けておいて，電源を投入するたびに情報をチップ内に入力してやる必要がある．しかし，電源を投入したままで，何度もプログラムし直すことができ，プログラムの柔軟性という点では最も優れている．

表-2 書込みの詳細

項目＼記憶素子	ヒューズ	EPROM	EEPROM	SRAM
接続情報の変更（回路の変更）	×	○	○	○
書き込める回数	1	100 回程度	100 回程度	無　限
専用書込み器	必　要	必　要	必　要	不　要
消去方法	できない	紫外線を当てる	高電圧（25 V 程度）	上書き，電源オフ
接続情報を入れた外部の不揮発性メモリ	不　要	不　要	不　要	必　要



B1 以外の端子からの配線は省略

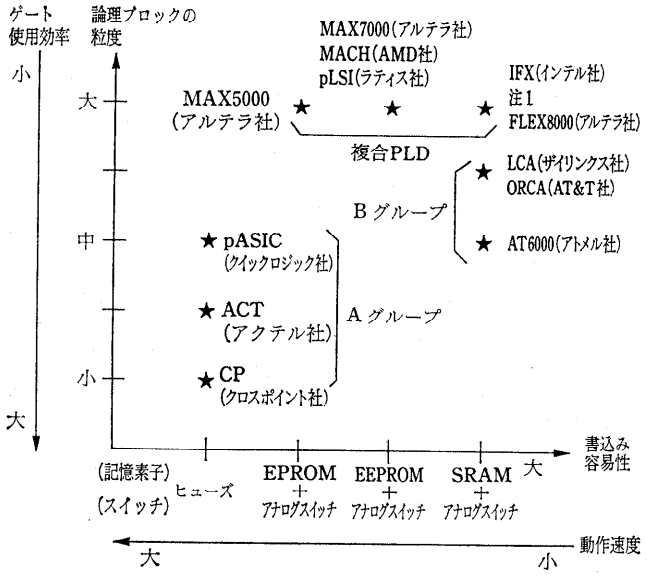
図-8 スイッチマトリックス

ユーザにとって重要な FPGA の動作速度や，使えるゲート数は，以上述べた論理ブロックの粒度と，結線情報を保持する素子により大きく影響を受ける．FPGA の動作速度は，基本的には，ゲートアレイを含む一般の IC 同様に，FET の動作速度自体で決まるが，FPGA の場合配線自体の遅延やスイッチマトリックスなどの接続のための遅延が大きく影響する．接続にヒューズを用いるタイプはこの遅延が最も小さくて済み，SRAM を用いるタイプでは最も大きくなる．

次に，使えるゲート数であるが，これは，論理ブロックの粒度によって影響される．ゲートアレイの場合もカタログスペックどおりの 100% のゲートを使えるわけではないが，FPGA の場合はこれが極端で，実際に使えるゲート数が 50% 以下もあり得る．これは，FPGA の場合，論理ブロックの機能と論理ブロック間の配線がゲートアレイに比べ制約が大きいため，どうしてもうまく使

えない部分が出てきてしまうためである．このため，論理ブロック粒度が小さいほどこの制限が緩くなり，使用効率が大きくなる．

以上述べた二つの基準により，各社FPGA を分類したのが**図-9** である．大きく分けて，速度と使用効率重視でゲートアレイに近いAグループと，柔軟性重視のBグループに分けられる．さらに，粒度が大きく，やや異なった性質をもつ複合PLD (Complex PLD) と呼ばれるグループがある．これは次章に述べる PLD の発展形態で，厳密にはFPGA に含めないが，ほぼ同じように用いられるため，区別しない場合が多い．



注 1 iFX は EPROM ももっている．

図-9 FPGA の分類

5. FPGA と PLD

プログラマブルなディジタル IC としては PLA (Programmable Logic Array) または PLD (Programmable Logic Device) と呼ばれる素子が 1970 年代から用いられてきた．これは**図-10** に示すように NOT-AND-OR の各ゲートをこの順に並べ，この間の配線を自由に変えることのできる配線マトリックスをもつ．この配線マトリックスをON/OFF することにより，入出力数の許す限りどのような多項式からなる組合せ回路でも構成することができる．この配線マトリックスの決定には主として初期にはヒューズが，最近は EEPROM が多く用いられている．さらに，OR ゲートの出力にフリップフロップを接続することにより，任意の順序回路を構成することができ，これらを一つにまとめたタイプの PLD も登場した．このような PLD は，小規模の組合せ回路，順序回路の実現に便利で，現在も Lattice 社の GAL シリーズをはじめとして広く用いられている．

しかし，これらの PLD は，基本的に実現可能な回路が単一の組合せ回路／順序回路に限られてしまい，CPU のような大規模かつ複雑な構造のディジタル回路を1チップ内に納めることはできない．そこで，これらの PLD を論理ブロックとしてチップ内に複数配置し，FPGA 同様，クロスポイントスイッチなどでこれらの接続をプログラム可能にしたデバイスが登場した．これらを複合

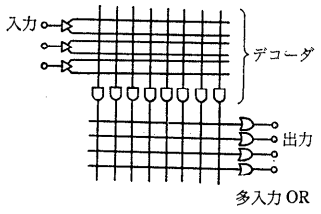


図-10 PLD の構造

PLD と呼ぶ．複合 PLD は各論理ブロックが PLD でできているために，図-9 で示したように，粒度は他の FPGA より大きくなる．しかし，論理ブロック内の構造をルックアップテーブルなどで実現するか，AND-OR 論理で実現するかは，実際上大きな相違でないため，これらはほぼ同様の特性をもち，同様の分野で用いられる．

6. FPGA の問題点

FPGA には二つの主要な問題がある．第1は**ゲート数が小さい**．FPGA の原理から分かることは，使われないゲートが出てくることと，スイッチと配線の領域を余分に必要とすることである．たとえばフリップフロップばかりを使うメモリのような回路を FPGA で実現すると，AND や OR などのゲートは使われないままになる．またゲートアレイでは，必要な配線だけパターンとして引けばよいが，FPGA では多くの配線をあらかじめ

引いておかねばならない．このような点がひびいて，ゲートアレイでは100万ゲートの製品まであるが，FPGA では2万ゲートまでである．

第2は**動作速度が遅い**．FPGA では論理ブロックから論理ブロックへ信号を伝えるとき，スイッチを通る．スイッチには“抵抗成分”と“容量成分”があり，信号に遅延を生じる．遅延は 0.数 nS 程度であるが，スイッチを何段か通るため加算され，かなり大きい遅延になる．そのため回路全体としては数十 MHz 程度までの速度でしか動作しない．ゲートアレイでは MOS—FET と MOS—FET を配線で直接結ぶため，このような問題はない．

上記の二つの問題とゲートアレイに比べて**価格が高い**ことがひびいて，現在の FPGA の使われ方は，最終製品でゲートアレイを使う前の試作用が主である[5]．

現実的な問題は**選択に困る**ことである．FPGA は複合 PLD まで含むと種類が多くメーカも多い．メーカごとに開発ツールがあり，その間の互換性に乏しい．一つの種類の FPGA が万能なら良いが，**各 FPGA ごとに得意・不得意があり，その選択によって，開発機関，開発コスト，最終的な性能に大きく影響してしまう．使い分ける必要がある．**

7. FPGA の将来？

FPGA の市場は毎年 +50% の伸びを示しており[1),2)]，速度，ゲート数とも毎年数十％の改善が見込まれている．市場の規模拡大につれて価格も下がってきており，最終製品に使われることも多くなった．開発ツールの問題も，汎用開発ツールがいくつか見受けられるようになった[3)]．

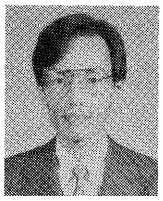
この分野では，新しい提案により新たな用途が生まれ，理論と応用の好循環が起きていることは確かである．

最後に参考文献をあげる．参考文献の 6)にゲートアレイや FPGA 全般についての分かりやすい解説があり，また，7)，8) では FPGA を用いて CPUを 設計する事例が連載で紹介されている．

参　考　文　献

1) 新市場を形成する PLD・FPGA，日経エレクトロニクス，No. 598, p. 124, 日経 BP 社 (1994. 1. 3)．
2) 選択の幅が広がってきた FPGA，参入メーカ相次ぐ，日経エレクトロニクス，No. 514, pp. 167-173, 日経 BP 社 (1990. 11. 26)．
3) 1万ゲートを超えた FPGA　マニアックな使い方から脱皮，日経エレクトロニクス，No. 595, pp. 85-104, 日経 BP 社 (1993. 11. 22)．
4) 鈴木　洋，中村　歩：CPLD/FPGA 活用研究，トランジスタ技術，別冊 pp. 4-22, CQ 出版社 (1993. 7)．
5) 中村直毅：再考する．FPGA/ComplexPLD, Computer Design, pp. 23-30, 電波新聞社 (1993. 9)．
6) 小林幹典，松村清明，前田利夫：ASIC のすすめ導入編，トランジスタ技術，別冊pp. 4-20, CQ 出版社 (1994. 1)．
7) 木村真也：FPGA を使ってマイクロコンピュータを作る(1)，トランジスタ技術，pp. 378-384, CQ 出版社 (1993. 7)．
8) 中林祥恵，江森　玲，今井正治：VHDL によるマイクロプロセッサ設計(1)，インターフェース，pp. 180-186, CQ 出版社 (1994. 2)．

(平成5年9月25日受付)



身次　茂（正会員）

1981 年熊本大学電子工学科卒業．同年パナファコム(株)入社．1986 年熊本テクノポリス財団電子応用機械技術研究所入所．3次元視覚センサ，コンピュータビジョンおよびコンピュータアーキテクチャの研究に従事．脳の情報処理に興味をもつ．電子情報通信学会，日本ロボット学会各会員．