Origin of FPGAs

ADAPTED FROM:

The Design Warrior's Guide to FPGAs
Devices, Tools, and Flows. ISBN 0750676043
Copyright © 2004 Mentor Graphics Corp.
(www.mentor.com)
Programmable Logic Devices

PROMs → PLAs → PALs → GALs → etc.

Unprogrammed PROM

Predefined AND array

Predefined link
Programmable link

Address 0
Address 1
Address 2
Address 3
Address 4
Address 5
Address 6
Address 7
Some Combinational Logic

<table>
<thead>
<tr>
<th></th>
<th>a</th>
<th>b</th>
<th>c</th>
<th>w</th>
<th>x</th>
<th>y</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>

Programmed PROM

Predefined AND array

Predefined link
Programmable link

w = (a & b)
x = !(a & b)
y = (a & b) ^ c
Unprogrammed PLA

Programmed PLA

w = (a & c) | (lb & lc)

x = (a & b & c) | (lb & lc)

y = (a & b & c)
Unprogrammed PAL

Programmable AND array

Predefined link

Programmable link

Generic CPLD Structure

Programmable Interconnect matrix

Input/output pins

SPLD-like blocks
Using Programmable Multiplexers

Programming a Physical PLD
Different Types of ASICs

- Gate Arrays
- Structured ASICs
- Standard Cell
- Full Custom

Increasing complexity

Examples of simple gate array basic cells

(a) Pure CMOS basic cell
(b) BiCMOS basic cell
Channeled gate array architectures

(a) Single-column arrays
(b) Dual-column arrays

Structured ASIC tiles

(a) Gate, mux, and flop-based
(b) LUT and flop-based
Generic structured ASIC

- Prefabricated I/O, cores, etc.
- Embedded RAM
- Sea-of-tiles

The Gap-- FPGAs

PLDs
  - SPLDs
  - CPLDs

ASICs
  - Gate Arrays
  - Structured ASICs*
  - Standard Cell
  - Full Custom

*Not available circa early 1980s
Simple Logic Block

3-input LUT

mux
flip-flop

q

y

clock
d
c
b
a

3

Configuring a LUT

Required function

y = (a & b) | !c

Truth table

<table>
<thead>
<tr>
<th>a</th>
<th>b</th>
<th>c</th>
<th>y</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

Programmed LUT

SRAM cells

y

a
b
y

111
Simple generic FPGA architecture

FPGA augmenting an ASIC
### Summary of Programming Technologies

<table>
<thead>
<tr>
<th>Feature</th>
<th>SRAM</th>
<th>Antifuse</th>
<th>E2PROM / FLASH</th>
</tr>
</thead>
<tbody>
<tr>
<td>Technology node</td>
<td>State-of-the-art</td>
<td>One or more generations behind</td>
<td>One or more generations behind</td>
</tr>
<tr>
<td>Reprogrammable</td>
<td>Yes (in system)</td>
<td>No</td>
<td>Yes (in-system or offline)</td>
</tr>
<tr>
<td>Reprogramming speed (inc. erasing)</td>
<td>Fast</td>
<td>—</td>
<td>3x slower than SRAM</td>
</tr>
<tr>
<td>Volatile (must be programmed on power-up)</td>
<td>Yes</td>
<td>No</td>
<td>No (but can be if required)</td>
</tr>
<tr>
<td>Requires external configuration file</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Good for prototyping</td>
<td>Yes (very good)</td>
<td>No</td>
<td>Yes (reasonable)</td>
</tr>
<tr>
<td>Instant-on</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>IP Security</td>
<td>Acceptable (especially when using obfuscation encryption)</td>
<td>Very Good</td>
<td>Very Good</td>
</tr>
<tr>
<td>Size of configuration file</td>
<td>Large (six transistors)</td>
<td>Very small (two transistors)</td>
<td>Medium-small (two transistors)</td>
</tr>
<tr>
<td>Power consumption</td>
<td>Medium</td>
<td>Low</td>
<td>Medium</td>
</tr>
<tr>
<td>Rad Hard</td>
<td>No</td>
<td>Yes</td>
<td>Not really</td>
</tr>
</tbody>
</table>

### Mux-based logic block

![Mux-based logic block diagram](image)
Function and Table

Required function

\[ y = (a \land b) \lor c \]

Truth table

<table>
<thead>
<tr>
<th>( a )</th>
<th>( b )</th>
<th>( c )</th>
<th>( y )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

Transmission-based LUT

![Transmission-based LUT Diagram]

Transmission gate (active low)
Transmission gate (active high)
SRAM cells
\[ y \]
Configuration cells linked in a chain

From the previous cell in the chain

SRAM cells

To the next cell in the chain

A Multifaceted LUT

16-bit SR

16 x 1 RAM

4-input LUT
Xilinx CLB

Configurable logic block (CLB)

Slice
Logic cell
Logic cell
Slice
Logic cell
Logic cell

Embedded RAM Blocks

Columns of embedded RAM blocks
Arrays of programmable logic blocks
Embedded Multipliers and RAM blocks

Multiplier-Accumulator
**Embedded Microprocessor core**

- Main FPGA fabric
- The “Stripe”
- Microprocessor core, special RAM, peripherals and I/O, etc.

**Multiple Cores**

- (a) One embedded core
- (b) Four embedded cores
Simple Clock Tree

Managing Clocks
Jitter

1 2 3 4

Ideal clock signal
Real clock signal with jitter
Cycle 1
Cycle 2
Cycle 3
Cycle 4
Superimposed cycles

Jitter remover

Clock signal from outside world with jitter

Special clock pin and pad

Clock Manager

“Clean” daughter clocks used to drive internal clock trees or output pins

CSE467 39

CSE467 40
Frequency Synthesis

- 1.0 x original clock frequency
- 2.0 x original clock frequency
- .5 x original clock frequency

Phase-shifted Clocks

- 0° Phase shifted
- 90° Phase shifted
- 180° Phase shifted
- 270° Phase shifted
Deskewing

Clock signal from outside world

Special clock pin and pad

Daughter clock (monitored downstream of the clock manager)

fed back to special input

De-skewed daughter clocks used to drive internal clock trees or output pins

Main (mother) clock

Untreated daughter clock

De-skewed daughter clock

General-purpose IO banks

General-purpose I/O banks 0 through 7