Introduction to Programmable Logic Devices

John Coughlan
RAL Technology Department
Detector & Electronics Division
PPD Lectures

Programmable Logic is Key Underlying Technology.

- First-Level and High-Level Triggering
- Data Transport
- Computers interacting with Hardware (VME Bus)
- Silicon Trackers (Reading out Millions of Data Channels)

Commercial Devices. Developments driven by Industry. Telecomms, Gaming, Aerospace, Automotive, Set-top boxes….
Particle Physics Electronics

CERN LHC

CMS
Custom Electronics Chips
ASICs
Rad Hard, Low Power

Electronics “Counting” Room(s)
Trigger Systems. DAQ Systems.

Purpose Built Digital Processing Boards
In VME Bus Crates

john.coughlan@stfc.ac.uk
Particle Physics Electronics

- Special Dedicated Logic Functions (not possible in CPUs)
  - Ultra Fast Trigger Systems (Trigger Algorithms) Clock Accurate Timing
  - Massively Parallel Data Processing (Silicon Trackers with Millions of Channels)

Custom Designed Printed Circuit Boards PCBs.

Commercial Programmable Logic Devices

john.coughlan@stfc.ac.uk
CMS & ATLAS DAQ/Trigger Architectures

"Telecoms Network"
Lecture Outline

- Programmable Logic Devices
  - Basics
  - Evolution

- **FPGA** Field Programmable Gate Arrays
  - Architecture

- Design Flow
  - Design Tools
  - Hardware Description Languages
Digital Logic

Connect Standard Logic Chips
Very Simple Glue Logic

Discrete Logic
Large board area

FIXED Logic

Digital Logic Function

3 Inputs

Product AND (&)
Sum OR ()

Black Box
Truth Table
(Look Up Table LUT)

Boolean Logic Minimisation

CMOS NAND gate

Transistor Switches

MOORE’S LAW
40 nm!

john.coughlan@stfc.ac.uk
Programmable Logic Devices
PLDs

- SUM of PRODUCTS
- (Re-)Programmable Links
- Reconfigurable

\[ y = (a \& b \& c) \]
\[ x = (a \& b \& c) \lor (\overline{b} \& \overline{c}) \]
\[ w = (a \& c) \lor (\overline{b} \& \overline{c}) \]
Complex PLDs

- CPLDs
- Programmable PLD Blocks
- Programmable Interconnects
- Electrically Erasable links

CPLD Architecture

Feedback Outputs

Add Flip Flops/Registers (Clocked Logic) -> State Machines
**Application Specific Integrated Circuits (ASICs)**

**Prefabricated Programmed**

- PLDs
- SPLDs
- CPLDs

**Custom Fabricated Design from Scratch**

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

*Not available circa early 1980s

**Limited Complexity**

- Thousands of Gates
- Cheap
- Easy to Design
- Reprogrammable

**Large Complex Functions. Millions of Gates**

- Customised for Extremes of Speed, Low Power, Radiation Hardness

- (Very) Expensive to Design (in small quantities)
  - > $1 Million mask set
- (Very) Hard to Design.
- Long Design cycles.
- NOT Reprogrammable. FROZEN in Silicon. High Risk

john.coughlan@stfc.ac.uk
FPGAs

PLDs

SPLDs

CPLDs

ASICs

Gate Arrays

Structured ASICs*

Standard Cell

Full Custom

*Not available circa early 1980s

Large Complex Functions

Inexpensive
Easy to Design, Rapid
Protoyping.
Reprogrammable. Changing
data standards.
Time line of Programmable devices

- Transistors
- ICs (General)
- SRAMs & DRAMs
- Microprocessors
- SPLDs
- CPLDs
- ASICs
- FPGAs

Field Programmable Gate Arrays
FPGA

Field Programmable Gate Array
- ‘Simple’ Programmable Logic Blocks
- Massive Fabric of Programmable Interconnects
- Standard CMOS Integrated Circuit fabrication process as for memory chips (Moore’s Law)

Huge Number of Logic Block ‘Islands’
1,000 … 100,000’s +
in a ‘Sea’ of Interconnects

FPGA Architecture
Logic Blocks

- Logic Functions implemented in Look Up Table LUTs. Truth Tables.
- Multiplexers (select 1 of N inputs)
Look Up Tables LUTs

- LUT contains Memory Cells to implement small logic functions
- Each cell holds ‘0’ or ‘1’.
- Programmed with outputs of Truth Table
- Inputs select content of one of the cells as output

3 Inputs LUT --> 8 Memory Cells

3 - 6 Inputs

Multiplexer MUX

Static Random Access Memory (SRAM) cells
Logic Blocks

- Larger Logic Functions built up by connecting many Logic Blocks together

![Diagram of logic block connections]

Fabric also has Larger Memory Blocks Block SRAM useful for Data storage

And other “Hard Wired” logic blocks Eg CPUs, Memory Controllers…
Programmable Routing

- Connections Routing signals between Logic Blocks
- Determined by SRAM cells

Special Routing for Clocks

SRAM cells
Clocked Logic

- Registers on outputs. CLOCKED storage elements.
- Sequential Logic Functions (cf Combinational Logic LUTs)
- Synchronous FPGA Logic Design, Pipelined Logic.
- FPGA Fabric driven by Global Clock (e.g. LHC BX frequency)

Clock from Outside world (eg LHC bunch frequency)
Pipeline

- Split the task into smaller steps with Registers in between.
- All driven by common clock

Pipeline (Car Assembly Line)
Massively Parallel

Clock
Input Output I/O Getting data in and out

Up to > 1,000 I/O “pins” (several 100 MHz)

Special I/O SERIALISERS
~ 10 GHz transfer rates

Transceiver block
Differential pairs
Optical TRx

john.coughlan@stfc.ac.uk
Designing Logic with FPGAs

- Design Capture.
- High level Description of Logic Design.
  - Graphical descriptions
  - Hardware Description Language (Textual)

When clock rises
If (s == 0)
then y = (a & b) | c;
else y = c & !(d ^ e);

Top-level block-level schematic
Graphical State Diagram
Graphical Flowchart
Textual HDL
Block-level schematic
Hardware Description Languages

- Language describing hardware (Engineers call it **FIRMWARE**)
- Doesn’t behave like “normal” programming language ‘C/C++’
- Describe Logic as collection of Processes operating in Parallel
- Language Constructs for Multiplexers, FlipFlops …etc
- Compiler (**Synthesis**) Tools recognise certain code constructs and generate appropriate logic
- Popular languages are **VHDL**, **VERILOG**

```vhd
if SEL = "00" then Y = A;
else if SEL = "01" then Y = B;
else if SEL = "10" then Y = C;
else Y = D;
end if;

2:1 MUX
D  2:1 MUX
C  2:1 MUX
B  2:1 MUX
A  Y
SEL = 10
SEL = 01
SEL = 00
```

```vhd
case SEL of;
"00": Y = A;
"01": Y = B;
"10": Y = C;
otherwise: Y = D;
end case;
```

4:1 MUX

A → 00
B → 01
C → 10
D → 11
Y
SEL
architecture Behavioral of dpmbufctrl is

begin

--bit order reverse address and data buses to match EDK scheme
bram_addr(0 to 31) <= bram_addr_i(31 downto 0);

--N.B. EDK DOCM addresses are byte orientated count in 4s for whole words
begin

if clk'event and clk = '1' then

if rst = '1' then

buf_zone:=0;

acount <= (others => '0');
dcount <= (others => '0');
bram_wen <= (others => '0');
bram_addr_i <= X"00001FFC";
--
bram_dout_i <= (others => '0');
state:=0;
elseif state = 0 then

--wait for din(0) at address 1FFC to be set to zero
--what about pipeline of BRAM - need to wait before polling?
bram_wen <= (others => '0');
acount <= (others => '0');
bram_addr_i <= X"00001FFC";
--
bram_dout_i <= (others => '0');
dcount <= dcount;
if bram_din_i = X"00000000"

state := 1;
else

state := 0;
end if;
end if;
end if;
end process;

end architecture;
Designing Logic with FPGAs

- **High level Description of Logic Design**
  - Graphical descriptions
  - Hardware Description Language (Textual)

- **Compile (Synthesis) into NETLIST.**

- **Boolean Logic Gates.**

- **Target FPGA Device**
  - Mapping
  - Routing

- **Bit File for FPGA**

- **Commercial CAE Tools**
  (Complex & Expensive)

- **Logic Simulation**
Configuring an FPGA

- Millions of SRAM cells holding LUTs and Interconnect Routing
- Volatile Memory. Lose configuration when board power is turned off.
- Keep Bit Pattern describing the SRAM cells in non-Volatile Memory e.g. PROM or Digital Camera card
- Configuration takes ~ secs

JTAG Testing

Programming Bit File

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

- Large Complex Functions
- Re-Programmability, Flexibility.

- Massively Parallel Architecture
  - Processing many channels simultaneously of MicroProcessor sequential processing

- Fast Turnaround Designs ☺
- Standard IC Manufacturing Processes ☻
- Leading Edge of Moore’s Law ☺
- Mass produced. Inexpensive. ☺

- Not Radiation Hard ☹
- Power Hungry ☹
FPGA Trends

- State of Art is 40nm on 300 mm wafers
- Top of range >100,000 Logic Blocks
- >1,000 pins (Fine Pitched BGA)
- Logic Block cost ~ 1$ in 1990

Problems
- Power. Leakage currents.
- Design Gap
  - CAE Tools
Summary

- Programmable Logic Devices
  - Basics
  - Evolution

- FPGA Field Programmable Gate Arrays
  - Architecture

- Design Flow
  - Hardware Description Languages
  - Design Tools

Importance for Particle Physics Experiments