FIELD PROGRAMMABLE GATE ARRAYS (FPGAS)

Roth Text: Chapter 3 (section 3.4)
Chapter 6
Nelson Text: Chapter 11
Programmable logic taxonomy

- **Factoy Programmable Devices**
  - ROM
    - Read-Only Memory
  - MPG
    - Mask Programmable Gate Array
  - SPLD
    - Simple Programmable Logic Device
  - CPLD
    - Complex Programmable Logic Device

- **Field Programmable Devices**
  - FPGA
    - Field Programmable Gate Array
  - GAL
    - Generic Array Logic
  - PLA
    - Programmable Array Logic
  - PAL
    - Programmable Logic Array

*FPGAs*
Field Programmable Gate Arrays

Typical Complexity = 5M – 1B transistors
Basic FPGA Operation

• Writing configuration memory ⇒ defines system function
  – Input/Output Cells
  – Logic in PLBs
  – Connections between PLBs & I/O cells

• Changing configuration memory data ⇒ changes system function
  – Can change at anytime
  – Even while system function is in operation
FIGURE 3-27: **Typical Architectures for FPGAs**

(a) Matrix based (symmetrical array)

(b) Row based

(c) Hierarchical

(d) Sea of gates

**FPGAs**
<table>
<thead>
<tr>
<th>Company</th>
<th>Device Names</th>
<th>General Architecture</th>
<th>Logic Block Type</th>
<th>Programming Technology</th>
</tr>
</thead>
<tbody>
<tr>
<td>Actel</td>
<td>ProASIC/ProASIC3/ProASIC\textsuperscript{plus} SX/SXA/eX/MX Accelerator Fusion</td>
<td>Sea of Tiles Sea of Modules Sea of Tiles</td>
<td>Multiplexers &amp; Basic Gates Multiplexers &amp; Basic Gates Multiplexers &amp; Basic Gates</td>
<td>SRAM Antifuse SRAM Flash, SRAM</td>
</tr>
<tr>
<td>Xilinx</td>
<td>Virtex Spartan</td>
<td>Symmetrical Array Symmetrical Array</td>
<td>LUT LUT</td>
<td>SRAM SRAM</td>
</tr>
<tr>
<td>Atmel</td>
<td>AT40KAL</td>
<td>Cell Based</td>
<td>Multiplexers &amp; Basic Gates</td>
<td>SRAM</td>
</tr>
<tr>
<td>QuickLogic</td>
<td>Eclipse II PolarPro</td>
<td>Flexible Clock Cell Based</td>
<td>LUT LUT</td>
<td>SRAM SRAM</td>
</tr>
<tr>
<td>Altera</td>
<td>Cyclone II Stratix II APEX II</td>
<td>Two-Dimensional Row and Column Based Two-Dimensional Row and Column Based Row and Column, but Hierarchical Interconnect</td>
<td>LUT LUT LUT</td>
<td>SRAM SRAM SRAM</td>
</tr>
</tbody>
</table>

**TABLE 3-9: Architecture, Technology, and Logic Block Types of Commercial FPGAs**
# Ranges of Resources

<table>
<thead>
<tr>
<th>FPGA Resource</th>
<th>Small FPGA</th>
<th>Large FPGA</th>
</tr>
</thead>
<tbody>
<tr>
<td>Logic</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLBs per FPGA</td>
<td>256</td>
<td>25,920</td>
</tr>
<tr>
<td>LUTs and flip-flops per PLB</td>
<td>1</td>
<td>8</td>
</tr>
<tr>
<td>Routing</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Wire segments per PLB</td>
<td>45</td>
<td>406</td>
</tr>
<tr>
<td>PIPs per PLB</td>
<td>139</td>
<td>3,462</td>
</tr>
<tr>
<td>Specialized Cores</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bits per memory core</td>
<td>128</td>
<td>36,864</td>
</tr>
<tr>
<td>Memory cores per FPGA</td>
<td>16</td>
<td>576</td>
</tr>
<tr>
<td>DSP cores</td>
<td>0</td>
<td>512</td>
</tr>
<tr>
<td>Other</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Input/output cells</td>
<td>62</td>
<td>1,200</td>
</tr>
<tr>
<td>Configuration memory bits</td>
<td>42,104</td>
<td>79,704,832</td>
</tr>
</tbody>
</table>
**Programmable ASIC logic cells**

- Xilinx: “configurable logic block” (CLB) contains
  - SRAM lookup tables (LUTs) to implement combinational logic
  - D flip flops
  - Multiplexers to establish paths in the CLB
- Actel “ACT”: multiplexers implement logic
- Altera “Flex”: similar to Xilinx CLB
- Altera “MAX”: PALs implement logic

**TABLE 3-8: Characteristics of the Major FPGA Programming Technologies**

<table>
<thead>
<tr>
<th>Programming Technology</th>
<th>Volatility</th>
<th>Programmability</th>
<th>Area Overhead</th>
<th>Resistance</th>
<th>Capacitance</th>
</tr>
</thead>
<tbody>
<tr>
<td>SRAM</td>
<td>Volatile</td>
<td>In-circuit reprogrammable</td>
<td>Large</td>
<td>Medium to high</td>
<td>High</td>
</tr>
<tr>
<td>EPROM</td>
<td>Nonvolatile</td>
<td>Out-of-circuit reprogrammable</td>
<td>Small</td>
<td>High</td>
<td>High</td>
</tr>
<tr>
<td>EEPROM</td>
<td>Nonvolatile</td>
<td>In-circuit reprogrammable</td>
<td>Medium to high</td>
<td>High</td>
<td>High</td>
</tr>
<tr>
<td>Antifuse</td>
<td>Nonvolatile</td>
<td>Not reprogrammable</td>
<td>Small</td>
<td>Small</td>
<td>Small</td>
</tr>
</tbody>
</table>
Mux-based logic blocks in FPGAs

Text Figure 3.33

(a) Mux-based logic blocks in FPGAs

(b) 4-to-1 MUX

DFF

CLK
CLR
**Actel ACT architecture (Fig. 5.1)**

*mux-based logic modules*
Xilinx FPGAs

• Virtex and Spartan 2
  – Array of 96 to 6,144 PLBs
    • 4 LUTs/RAMs (4-input)
    • 4 FF/latches
  – 4 to 32 4K-bit dual-port RAMs
• Virtex II, Virtex II Pro
  – Array of 352 to 11,204 PLBs
    • 8 LUTs/RAMs (4-input)
    • 8 FF/latches
  – 12 to 444 18K-bit dual-port RAMs
  – 12 to 444 18×18-bit multipliers
  – 0 to 2 PowerPC processor cores
• Virtex 4
  – Array of 1,536 to 22,272 PLBs
    • 4 LUTs/RAMs (4-input)
    • 4 LUTs (4-input)
    • 8 FF/latches
  – 48 to 552 18K-bit dual-port RAMs
    • Also operate as FIFOs
  – 32 to 512 DSP cores include:
    – 0 to 2 PowerPC processor cores

Spartan 3

• Array of 192 to 8,320 PLBs
  • 4 LUTs/RAMs (4-input)
  • 4 LUTs (4-input)
  • 8 FF/latches
  • 4 to 104 18K-bit dual-port RAMs
  • 4 to 104 18×18-bit multipliers

FPGAs
## Xilinx 7 Series Families

<table>
<thead>
<tr>
<th>Maximum Capability</th>
<th>Artix-7 Family</th>
<th>Kintex-7 Family</th>
<th>Virtex-7 Family</th>
</tr>
</thead>
<tbody>
<tr>
<td>Logic Cells</td>
<td>215K</td>
<td>478K</td>
<td>1,955K</td>
</tr>
<tr>
<td>Block RAM&lt;sup&gt;(1)&lt;/sup&gt;</td>
<td>13 Mb</td>
<td>34 Mb</td>
<td>68 Mb</td>
</tr>
<tr>
<td>DSP Slices</td>
<td>740</td>
<td>1,920</td>
<td>3,600</td>
</tr>
<tr>
<td>Peak DSP Performance&lt;sup&gt;(2)&lt;/sup&gt;</td>
<td>929 GMAC/s</td>
<td>2,845 GMAC/s</td>
<td>5,335 GMAC/s</td>
</tr>
<tr>
<td>Transceivers</td>
<td>16</td>
<td>32</td>
<td>96</td>
</tr>
<tr>
<td>Peak Transceiver Speed</td>
<td>6.6 Gb/s</td>
<td>12.5 Gb/s</td>
<td>28.05 Gb/s</td>
</tr>
<tr>
<td>Peak Serial Bandwidth (Full Duplex)</td>
<td>211 Gb/s</td>
<td>800 Gb/s</td>
<td>2,784 Gb/s</td>
</tr>
<tr>
<td>PCIe Interface</td>
<td>x4 Gen2</td>
<td>x8 Gen2</td>
<td>x8 Gen3</td>
</tr>
<tr>
<td>Memory Interface</td>
<td>1,066 Mb/s</td>
<td>1,866 Mb/s</td>
<td>1,866 Mb/s</td>
</tr>
<tr>
<td>I/O Pins</td>
<td>500</td>
<td>500</td>
<td>1,200</td>
</tr>
<tr>
<td>I/O Voltage</td>
<td>1.2V, 1.35V, 1.5V, 1.8V, 2.5V, 3.3V</td>
<td>1.2V, 1.35V, 1.5V, 1.8V, 2.5V, 3.3V</td>
<td>1.2V, 1.35V, 1.5V, 1.8V, 2.5V, 3.3V</td>
</tr>
<tr>
<td>Package Options</td>
<td>Low-Cost, Wire-Bond, Lidless Flip-Chip</td>
<td>Low-Cost, Lidless Flip-Chip and High-Performance Flip-Chip</td>
<td>Highest Performance Flip-Chip</td>
</tr>
</tbody>
</table>

*FPGAs*
## Xilinx Artix-7 Family

<table>
<thead>
<tr>
<th>Device</th>
<th>Logic Cells</th>
<th>Configurable Logic Blocks (CLBs)</th>
<th>DSP48E1 Slices</th>
<th>Block RAM Blocks</th>
<th>CMTs</th>
<th>PCIe</th>
<th>GTPs</th>
<th>XADC Blocks</th>
<th>Total I/O Banks</th>
<th>Max User I/O</th>
</tr>
</thead>
<tbody>
<tr>
<td>XC7A15T</td>
<td>16,640</td>
<td>2,600</td>
<td>200</td>
<td>45</td>
<td>50</td>
<td>25</td>
<td>900</td>
<td>1</td>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>XC7A35T</td>
<td>33,280</td>
<td>5,200</td>
<td>400</td>
<td>90</td>
<td>100</td>
<td>50</td>
<td>1,800</td>
<td>1</td>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>XC7A50T</td>
<td>52,160</td>
<td>8,150</td>
<td>600</td>
<td>120</td>
<td>150</td>
<td>75</td>
<td>2,700</td>
<td>1</td>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>XC7A75T</td>
<td>75,520</td>
<td>11,800</td>
<td>892</td>
<td>180</td>
<td>210</td>
<td>105</td>
<td>3,780</td>
<td>1</td>
<td>8</td>
<td>1</td>
</tr>
<tr>
<td>XC7A100T</td>
<td>101,440</td>
<td>15,850</td>
<td>1,188</td>
<td>240</td>
<td>270</td>
<td>135</td>
<td>4,860</td>
<td>6</td>
<td>8</td>
<td>1</td>
</tr>
<tr>
<td>XC7A200T</td>
<td>215,360</td>
<td>33,650</td>
<td>2,888</td>
<td>740</td>
<td>730</td>
<td>365</td>
<td>13,140</td>
<td>10</td>
<td>16</td>
<td>1</td>
</tr>
</tbody>
</table>

**Notes:**
1. Each 7 series FPGA slice contains four LUTs and eight flip-flops; only some slices can use their LUTs as distributed RAM or SRLs.
2. Each DSP slice contains a pre-adder, a 25 x 18 multiplier, an adder, and an accumulator.
3. Block RAMs are fundamentally 36 Kb in size; each block can also be used as two independent 18 Kb blocks.
4. Each CMT contains one MCMC and one PLL.
5. Artix-7 FPGA Interface Blocks for PCI Express support up to x4 Gen 2.
6. Does not include configuration Bank 0.
7. This number does not include GTP transceivers.

### Package

<table>
<thead>
<tr>
<th>Package</th>
<th>Size (mm)</th>
<th>Ball Pitch (mm)</th>
<th>Device</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPG236</td>
<td>10 x 10</td>
<td>0.5</td>
<td>XC7A15T</td>
</tr>
<tr>
<td>CSG324</td>
<td>15 x 15</td>
<td>0.8</td>
<td>XC7A15T</td>
</tr>
<tr>
<td>CSG325</td>
<td>15 x 15</td>
<td>0.8</td>
<td>XC7A50T</td>
</tr>
<tr>
<td>FTG256</td>
<td>17 x 17</td>
<td>0.8</td>
<td>XC7A50T</td>
</tr>
<tr>
<td>SBG484</td>
<td>19 x 19</td>
<td>1.0</td>
<td>XC7A75T</td>
</tr>
<tr>
<td>FGG484(2)</td>
<td>23 x 23</td>
<td>1.0</td>
<td>XC7A75T</td>
</tr>
<tr>
<td>FGG676(3)</td>
<td>27 x 27</td>
<td>1.0</td>
<td>XC7A100T</td>
</tr>
<tr>
<td>FFG1156</td>
<td>35 x 35</td>
<td>1.0</td>
<td>XC7A200T</td>
</tr>
</tbody>
</table>

### I/O

<table>
<thead>
<tr>
<th>Device</th>
<th>GTP HR(4)</th>
<th>I/O GTP</th>
<th>I/O HR(4)</th>
<th>GTP HR(4)</th>
<th>I/O GTP</th>
<th>I/O HR(4)</th>
<th>GTP HR(4)</th>
<th>I/O GTP</th>
<th>I/O HR(4)</th>
<th>GTP HR(4)</th>
<th>I/O GTP</th>
<th>I/O HR(4)</th>
<th>GTP HR(4)</th>
<th>I/O GTP</th>
<th>I/O HR(4)</th>
<th>GTP HR(4)</th>
</tr>
</thead>
<tbody>
<tr>
<td>XC7A15T</td>
<td>2</td>
<td>106</td>
<td>0</td>
<td>210</td>
<td>4</td>
<td>150</td>
<td>0</td>
<td>170</td>
<td>4</td>
<td>250</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>XC7A35T</td>
<td>2</td>
<td>106</td>
<td>0</td>
<td>210</td>
<td>4</td>
<td>150</td>
<td>0</td>
<td>170</td>
<td>4</td>
<td>250</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>XC7A50T</td>
<td>2</td>
<td>106</td>
<td>0</td>
<td>210</td>
<td>4</td>
<td>150</td>
<td>0</td>
<td>170</td>
<td>4</td>
<td>250</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>XC7A75T</td>
<td>2</td>
<td>106</td>
<td>0</td>
<td>210</td>
<td>4</td>
<td>150</td>
<td>0</td>
<td>170</td>
<td>4</td>
<td>285</td>
<td>8</td>
<td>300</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>XC7A100T</td>
<td>2</td>
<td>106</td>
<td>0</td>
<td>210</td>
<td>4</td>
<td>150</td>
<td>0</td>
<td>170</td>
<td>4</td>
<td>285</td>
<td>8</td>
<td>300</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>XC7A200T</td>
<td>2</td>
<td>106</td>
<td>0</td>
<td>210</td>
<td>4</td>
<td>150</td>
<td>0</td>
<td>170</td>
<td>4</td>
<td>285</td>
<td>8</td>
<td>400</td>
<td>16</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
# Xilinx “UltraScale” Family

Kintex and Virtex UltraScale and UltraScale+

<table>
<thead>
<tr>
<th>MPSoC Processing System</th>
<th>Kintex UltraScale FPGA</th>
<th>Kintex UltraScale+ FPGA</th>
<th>Virtex UltraScale FPGA</th>
<th>Virtex UltraScale+ FPGA</th>
<th>Zynq UltraScale+ MPSoC</th>
<th>Zynq UltraScale+ RFSoC</th>
</tr>
</thead>
<tbody>
<tr>
<td>RF-ADC/DAC</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SD-FEC</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>System Logic Cells (K)</td>
<td>318-1,451</td>
<td>356-1,143</td>
<td>783-5,541</td>
<td>862-3,780</td>
<td>103-1,143</td>
<td>678-930</td>
</tr>
<tr>
<td>Block Memory (Mb)</td>
<td>12.7-75.9</td>
<td>12.7-34.6</td>
<td>44.3-132.9</td>
<td>23.6-94.5</td>
<td>4.5-34.6</td>
<td>27.8-38.0</td>
</tr>
<tr>
<td>UltraRAM (Mb)</td>
<td>0-36</td>
<td></td>
<td>90-360</td>
<td>0-36</td>
<td>13.5-22.5</td>
<td></td>
</tr>
<tr>
<td>HBM DRAM (GB)</td>
<td></td>
<td></td>
<td>0-8</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DSP (Slices)</td>
<td>768-5,520</td>
<td>1,368-3,528</td>
<td>600-2,880</td>
<td>2,280-12,288</td>
<td>240-3,528</td>
<td>3,145-4,272</td>
</tr>
<tr>
<td>DSP Performance (GMAC/s)</td>
<td>8,180</td>
<td>6,287</td>
<td>4,268</td>
<td>21,897</td>
<td>6,287</td>
<td>7,613</td>
</tr>
<tr>
<td>Transceivers</td>
<td>12-64</td>
<td>16-76</td>
<td>36-120</td>
<td>32-128</td>
<td>0-72</td>
<td>8-16</td>
</tr>
<tr>
<td>Max. Transceiver Speed (Gb/s)</td>
<td>16.3</td>
<td>32.75</td>
<td>30.5</td>
<td>32.75</td>
<td>32.75</td>
<td>32.75</td>
</tr>
<tr>
<td>Max. Serial Bandwidth (full duplex) (Gb/s)</td>
<td>2,086</td>
<td>3,268</td>
<td>5,616</td>
<td>3,268</td>
<td>1,048</td>
<td></td>
</tr>
<tr>
<td>Memory Interface Performance (Mb/s)</td>
<td>2,400</td>
<td>2,666</td>
<td>2,400</td>
<td>2,666</td>
<td>2,666</td>
<td>2,666</td>
</tr>
<tr>
<td>I/O Pins</td>
<td>312-832</td>
<td>280-668</td>
<td>338-1,456</td>
<td>208-832</td>
<td>82-668</td>
<td>280-408</td>
</tr>
</tbody>
</table>
Xilinx: Basic CLB Architecture

- Look-up Table (LUT) implements truth table
- Memory elements:
  - Flip-flop/latch
  - Some FPGAs - LUTs can also implement small RAMs
- Carry & control logic implements fast adders/subtractors
Combinational Logic Functions

- Gates are combined to create complex circuits
- Multiplexer example
  - If $S = 0$, $Z = A$
  - If $S = 1$, $Z = B$
  - Very common digital circuit
  - Heavily used in FPGAs
    - $S$ input controlled by configuration memory bit
    - We’ll see it again

<table>
<thead>
<tr>
<th>S</th>
<th>A</th>
<th>B</th>
<th>Z</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>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>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>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>
Look-up Tables

- Recall multiplexer example
- Configuration memory holds outputs for truth table
- Internal signals connect to control signals of multiplexers to select value of truth table for any given input value
Look-up Table Based RAMs

- Normal LUT mode performs read operations
- Address decoder with write enable generates clock signals to latches for write operations
- Small RAMs but can be combined for larger RAMs
A Simple CLB

- Two 3-input LUTs
  - Can implement any 4-input combinational logic function
- 1 flip-flop
  - Programmable:
    - Active levels
    - Clock edge
    - Set/reset
- 22 configuration memory bits
  - 8 per LUT
    - Co-7
    - So-7
  - 6 controls
    - CB0-7

LUT C 8x1

LUT S 8x1

D2-0

111 110 101 100 011 010 001 000

D2-0

LUT out

FPGAs
Example CLB

- Artix-7 SLICEL (1/2 shown)
- Four 6-input Look-Up Tables (LUTs)
  - Any combinational logic function of up to 6 inputs
  - SLICEM LUT can function as small RAM (16x1-bit) or shift register (up to 16-bit)
- Eight D flip-flops
  - Programmable as latches
  - Programmable clock edge, clock enable, set/reset
- Extra logic
  - Fast carry for adders
  - MUXs for Shannon expansion
  - And more
CLBs and Slices in rows/columns

Table 2-1: Logic Resources in One CLB

<table>
<thead>
<tr>
<th>Slices</th>
<th>LUTs</th>
<th>Flip-Flops</th>
<th>Arithmetic and Carry Chains</th>
<th>Distributed RAM(^{(1)})</th>
<th>Shift Registers(^{(1)})</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>8</td>
<td>16</td>
<td>2</td>
<td>256 bits</td>
<td>128 bits</td>
</tr>
</tbody>
</table>

Notes:
1. SLICEM only, SLICEL does not have distributed RAM or shift registers.
Using lookup-table (LUT) programmable logic

FIGURE 3-32: Highlighting Paths for Function $F_1$
Functions of more variables than # of LUT inputs

Shannon’s Expansion Theorem (partition into smaller functions):

\[ Z(a, b, c, d, e, f) = \overline{a} \cdot Z(0, b, c, d, e, f) + a \cdot Z(1, b, c, d, e, f) = \overline{a}Z_0 + aZ_1 \]
\[ M = \overline{S_1}S_0I_0 + \overline{S_1}S_0I_1 + S_1\overline{S_0}I_2 + S_1S_0I_3 \]

**FIGURE 6-2:** Highlighting Paths for a 4-to-1 Mux

**FIGURE 6-4:** A 4-to-1 Multiplexer in a Programmable Logic Block with Three Function Generators
FIGURE 6-5:
(a) Circular Shift Register;
(b) Implementation Using Simple FPGA Building Block
Fig. 6-13 Simplified Spartan and Virtex Slice
Input/Output Cells

- Bi-directional buffers
  - Programmable for input or output
  - Tri-state control for bi-directional operation
  - Flip-flops/latches for improved timing
    - Set-up and hold times
    - Clock-to-output delay
  - Pull-up/down resistors
- Routing resources
  - Connections to core of array
- Programmable I/O voltage & current levels
Detailed I/O Cell

**FIGURE 3-39: Programmable I/O Block for an FPGA**

The diagram illustrates the configuration of an I/O cell for an FPGA, including configuration bits for output invert, 3-state invert, latched output, slew rate, and passive pull up. The diagram also shows connections for output signal, 3-state (output enable), enable, input signal (latched), enable, flip-flop, output buffer, voltage reference, and global reset.
**Interconnect Network**

- Wire segments of varying length
  - \( xN = N \) CLBs in length
    - 1, 2, 4, and 6 are most common
  - \( xH = \) half the array in length
  - \( xL = \) length of full array

- Programmable Interconnect Points (PIPs)
  - Also known as Configurable Interconnect Points (CIPs)
  - Transmission gate connects to 2 wire segments
  - Controlled by configuration memory bit
    - 0 = wires disconnected
    - 1 = wires connected
Xilinx interconnect structures

FPGAs
**PIPs**

- **Break-point PIP**
  - Connect or isolate 2 wire segments

- **Cross-point PIP**
  - Turn corners

- **Multiplexer PIP**
  - Directional and buffered
  - Select 1-of-$N$ inputs for output
    - Decoded MUX PIP – $N$ config bits select from $2^N$ inputs
    - Non-decoded MUX PIP – 1 config bit per input

- **Compound cross-point PIP**
  - Collection of 6 break-point PIPs
    - Can route to two isolated signal nets
FIGURE 3-36: Direct Interconnects between Neighboring Logic Blocks

(a)

(b)

FIGURE 3-37: Global Lines

Logic Block

Logic Block

Logic Block

Logic Block

Tristate lines

FPGAs
Spartan 3 Routing Resources

switch matrix
over 2,400 PIPs
mostly MUX PIPs

PLB consists
of 4 slices

x6 wire segments

x2 wire segments

xH & xL wire segments

over 450 total wire segments in PLB

FPGAs
**Fully routed design**

- **Net N1**: Site S1 output pin O1 connects to input pin I1 on site S3
- **Net N2**: Site S1 output pin O2 connects to input pin I1 on site S2

Black “dots” are routing pips
Predefined connections exist between switch boxes
ELEC 4200 Lab 0 in Spartan 6
Lab 0 in Spartan 6
(routing details)
Ex: modulo 7 counter (device xc6slx25t)
FPGA clock regions

- Logic Resources
- Logic Resources
- Logic Resources
- Logic Resources
- Logic Resources
- Logic Resources
- Logic Resources
- Logic Resources

Clock Region
Clock Region
Clock Region

Center Clock Column(s)

Clock Routing
Spartan 6 clock tree example

- **BUFG**
- **CLKC tile**
- **Main vertical spine**
- **"Folded" vertical spine (one each in top and bottom half)**
- **Distribution wire in center of clock region**
- **Clock to INTs in one column**

- **BUFH**
- **INT**
- **CLEXM**

FPGAs
A clock region always contains 50 CLBs per column, ten 36K block RAMs per column (unless five 36K blocks are replaced by an integrated block for PCI Express®), 20 DSP slices per column, and 12 BUFHs. A clock region contains, if applicable, one CMT (PLL/MMC), one bank of 50 I/Os, one GT quad consisting of four serial transceivers, and half a column for PCIe® in a block RAM column.
Basic view of a clock region
Clock management tile (CMT)

Mixed-mode clock manager (MMCM)

MMCM Outputs = frequency divided, phase shifted, inverted
Up to 24 CMTs per Series 7 device
Clock management tile (CMT)  
Phase-locked loop (PLL)

PLL = frequency synthesizer using a voltage-controlled oscillator (VCO)
**Spartan 6 global clock sources**

- From top edge global clock pads
- From bottom edge global clock pads
- From left edge global clock pads
- From right edge global clock pads
- From DCM/PLL and fabric
- BUFG controls from fabric
- Switch Box
- Clock created by MIPS control logic
- From clock input pad
- From bottom edge global clock pads
- BUFGs
- Vertical spines

*FPGAs*
Specialized “hard “ cores

- **RAMs** – single-port, dual-port, FIFOs
  - 128 bits to 36K bits per RAM
  - 4 to 575 RAM cores per FPGA

- **DSPs** – 18x18-bit multiplier, 48-bit accumulator, etc.
  - up to 512 per FPGA

- **Microprocessors** and/or microcontrollers
  - Up to 2 per FPGA (hard core processor)
  - Support soft core processors
    - Synthesized from HDL into programmable resources

- **Communication** functions
  - Gigabit transceivers
  - Ethernet MAC
  - PCE Express bus
FPGA Architectures

• 4000/Spartan
  – $N \times N$ array of unit cells
    • Unit cell = CLB + routing
      – Special routing along center axes
    – I/O cells around perimeter

• Virtex/Spartan-2
  – $M \times N$ array of unit cells
  – Added block 4K RAMs at edges

• Virtex-2/Spartan-3
  – Block 18K RAMs in array
  – Added 18x18 multipliers with each RAM
  – Added PowerPCs in Virtex-2 Pro

• Virtex-4/Virtex-5
  – Added 48-bit DSP cores w/multipliers
  – I/O cells along columns for BGA
Xilinx Virtex-4 FPGAs

- Configuration memory: 4.7M to 50.8M bits of RAM
- PLBs: 1,536 to 22,272
  - 4 slices per PLB
    - 2 LUTs & 2 FFs per slice
    - 2 slices can operate as RAMs/SRs
- Block RAMs: 48 to 552 18K-bit dual-port RAMs
  - Also operate as FIFOs
- DSP cores: 32 to 512, each includes:
  - 18x18-bit multiplier
  - 48-bit adder & accumulator
- Up to 2 PowerPC processors
Block RAMs

- 36 Kbit dual-port RAM
- Each port independently configurable:
  - 1K words x 36 bits
    - 32 data bits + 4 parity bits
  - 2K words x 18 bits
    - 16 data bits + 2 parity bits
  - 4K words x 9 bits
    - 8 data bits + 1 parity bit
  - 8K words x 4 bits (no parity)
  - 16K words x 2 bits (no parity)
  - 32K words x 1 bit (no parity)
- Each port has independently programmable
  - clock edge, active levels for write enable, RAM enable, reset
FIGURE 6-19: Behavioral VHDL Code That Typically Infers Dedicated Memory

```vhdl
library IEEE;
use IEEE.numeric_bit.all;

entity Memory is
  port(Address: in unsigned(6 downto 0);
       CLK, MemWrite: in bit;
       Data_In: in unsigned(31 downto 0);
       Data_Out: out unsigned(31 downto 0));
end Memory;

architecture Behavioral of Memory is
  type RAM is array (0 to 127) of unsigned(31 downto 0);
signal DataMEM: RAM;  -- no initial values
begin
  process(CLK)
  begin
    if CLK'event and CLK = '1' then
      if MemWrite = '1' then
        DataMEM(to_integer(Address)) <= Data_In;  -- Synchronous Write
      end if;
      Data_Out <= DataMEM(to_integer(Address));  -- Synchronous Read
    end if;
  end process;
end Behavioral;
```

FPGAs
library IEEE;
use IEEE.numeric_bit.all;

entity LUTmult is
  port(Mplier, Mcand: in unsigned(3 downto 0);
       Product: out unsigned(7 downto 0));
end LUTmult;

architecture ROM1 of LUTmult is
  type PROD_ROM: =
    (x"00", x"00", x"00", x"00", x"00", x"00", x"00", x"00", x"00", x"00", x"00", x"00", x"00",
     x"00", x"01", x"02", x"03", x"04", x"05", x"06", x"07", x"08", x"09", x"0A", x"0B", x"0C", x"0D",
     x"0E", x"0F", x"00", x"02", x"04", x"06", x"08", x"0A", x"0C", x"0E", x"10", x"12", x"14", x"16",
     x"18", x"1A", x"1C", x"1E", x"00", x"03", x"05", x"07", x"09", x"0C", x"0E", x"10", x"12", x"14",
     x"16", x"18", x"1A", x"1C", x"1E", x"00", x"04", x"08", x"0C", x"10", x"14", x"18", x"1C", x"20",
     x"24", x"28", x"2C", x"30", x"34", x"38", x"3C", x"00", x"05", x"0A", x"0F", x"14", x"19", x"1E",
     x"23", x"28", x"2D", x"32", x"37", x"3C", x"41", x"46", x"4B", x"00", x"06", x"0C", x"12", x"18",
     x"1E", x"24", x"2A", x"30", x"36", x"3C", x"42", x"48", x"4E", x"54", x"5A", x"00", x"07", x"0E",
     x"15", x"1C", x"23", x"29", x"31", x"38", x"3E", x"46", x"4C", x"54", x"5B", x"62", x"69", x"00",
     x"08", x"10", x"18", x"20", x"28", x"30", x"38", x"40", x"48", x"50", x"58", x"60", x"68", x"70",
     x"78", x"00", x"09", x"12", x"18", x"24", x"2C", x"36", x"3E", x"48", x"51", x"5A", x"63", x"6C",
     x"75", x"7E", x"87", x"8F", x"96", x"00", x"0A", x"14", x"1E", x"28", x"32", x"3C", x"46", x"50",
     x"5A", x"64", x"6E", x"78", x"82", x"8C", x"96", x"00", x"0B", x"16", x"21", x"2C", x"37", x"42",
     x"4D", x"58", x"63", x"6E", x"79", x"84", x"8F", x"9A", x"A5", x"00", x"0C", x"18", x"24", x"30",
     x"3C", x"48", x"54", x"60", x"6C", x"78", x"84", x"90", x"9C", x"A8", x"B4", x"00", x"0D", x"1A",
     x"27", x"34", x"41", x"4E", x"5B", x"68", x"75", x"82", x"8F", x"9C", x"A9", x"B6", x"C3",
     x"00", x"0E", x"1C", x"2A", x"38", x"46", x"54", x"62", x"70", x"7E", x"8C", x"9A", x"A8", x"B6",
     x"C4", x"D2", x"00", x"0F", x"1E", x"2D", x"3C", x"4B", x"5A", x"69", x"78", x"87", x"96", x"A5",
     x"B4", x"C3", x"D2", x"E1"));
begin
  Product <= PROD_ROM(to_integer(Mplier&Mcand)); -- read Product LUT
end ROM1;
FIGURE 6-17: Creating Memory from LUTs

Distributed RAM

FPGAs
<table>
<thead>
<tr>
<th>FPGA Family</th>
<th>LUT-Based RAM (Kb)</th>
<th>No. of LUTs</th>
</tr>
</thead>
<tbody>
<tr>
<td>Xilinx Virtex 5</td>
<td>320–3420</td>
<td>19200–207,360</td>
</tr>
<tr>
<td>Xilinx Virtex 4</td>
<td>96–987</td>
<td>12288–126,336</td>
</tr>
<tr>
<td>Xilinx Virtex-II</td>
<td>8–1456</td>
<td>512–93,184</td>
</tr>
<tr>
<td>Xilinx Spartan 3E</td>
<td>15–231*</td>
<td>1920–29,504</td>
</tr>
<tr>
<td>Altera Stratix II</td>
<td>195–2242**</td>
<td>12480–143,520</td>
</tr>
<tr>
<td>Altera Cyclone II</td>
<td>72–1069**</td>
<td>4608–68,416</td>
</tr>
<tr>
<td>Lattice SC</td>
<td>245–1884</td>
<td>15200–115,200</td>
</tr>
<tr>
<td>Lattice ECP2</td>
<td>12–136</td>
<td>6000–68,000</td>
</tr>
</tbody>
</table>

* does not use all of the LUTs as distributed RAM  
** calculated from LUT counts
Refer to the “synthesis guide” for recommended HDL forms
FIGURE 6-21: Dedicated Multipliers

FIGURE 6-22: VHDL Code That Infers Dedicated Multipliers

```vhdl
library IEEE;
use IEEE.numeric_bit.all;

entity multiplier is
  port(A, B: in unsigned (31 downto 0);
       C: out unsigned (63 downto 0));
end multiplier;

architecture mult of multiplier is
begin
  C <= A * B;
end mult;
```
7 Series DSP48E1 DSP slice

- **25 × 18 two's-complement multiplier**: Dynamic bypass
- **48-bit accumulator**: Can be used as a synchronous up/down counter
- **Power-saving pre-adder**: Optimizes symmetrical filter applications and reduces DSP slice requirements
DSP48E1 slice details

*These signals are dedicated routing paths internal to the DSP48E1 column. They are not accessible via fabric routing resources.

FPGAs
# Embedded Processors

- **Hard core**
  - Faster
  - Fixed position
  - Few devices

- **Soft core**
  - Slower
  - Can be placed anywhere
  - Applicable to many devices

---

### Virtex-4 Processors:

<table>
<thead>
<tr>
<th>Embedded Processor</th>
<th>Core Type</th>
<th>Max Clock Frequency</th>
<th>Slices</th>
<th>PLBs</th>
<th>Block RAMs</th>
</tr>
</thead>
<tbody>
<tr>
<td>PowerPC</td>
<td>Hard</td>
<td>222 MHz</td>
<td>1000</td>
<td>250</td>
<td>9</td>
</tr>
<tr>
<td>Microblaze</td>
<td>Soft</td>
<td>180 MHz</td>
<td>940</td>
<td>235</td>
<td>9</td>
</tr>
<tr>
<td>Picoblaze</td>
<td>Soft</td>
<td>221 MHz</td>
<td>333</td>
<td>84</td>
<td>3</td>
</tr>
<tr>
<td>Picoblaze (optimized)</td>
<td>Soft</td>
<td>233 MHz</td>
<td>274</td>
<td>69</td>
<td>3</td>
</tr>
</tbody>
</table>

**ARM Processors in 7 Series**

---

![FPGAs](image-url)
**Xilinx Zynq SoC devices**

Zynq-7000 SoC: Dual-core ARM Cortex-A9 MPCore (up to 1GHz)

Zynq UltraScale+ MPSoC:
- Quad-core ARM Cortex-A53 MP (up to 1.5 GHz)
- Dual-core ARM Cortex-R5 MPCore (up to 600MHz)
- GPY ARM Mali-400 MP2 (up to 667MHz)
# Zynq-7000 SoC Features (1)

**Processing System Resources**

<table>
<thead>
<tr>
<th>Device Name</th>
<th>Z-7010</th>
<th>Z-7015</th>
<th>Z-7020</th>
<th>Z-7030</th>
<th>Z-7035</th>
<th>Z-7045</th>
<th>Z-7100</th>
</tr>
</thead>
<tbody>
<tr>
<td>Part Number</td>
<td>XC7Z010</td>
<td>XC7Z015</td>
<td>XC7Z020</td>
<td>XC7Z030</td>
<td>XC7Z035</td>
<td>XC7Z045</td>
<td>XC7Z100</td>
</tr>
</tbody>
</table>

- **Processor Core**: Dual-core ARM® Cortex™-A9 MPCore™ with CoraSight™
- **Processor Extensions**: NEON™ & Single / Double Precision Floating Point for each processor
- **Maximum Frequency**: 667 MHz (-1); 766 MHz (-2); 866 MHz (-3)
- **L1 Cache**: 32 KB Instruction, 32 KB data per processor
- **L2 Cache**: 512 KB
- **On-Chip Memory**: 256 KB
- **External Memory Support**: DDR3, DDR3L, DDR2, LPDDR2
- **External Static Memory Support**: 2x Quad-SPI, NAND, NOR
- **DMA Channels**: 8 (4 dedicated to Programmable Logic)
- **Peripherals**: 2x UART, 2x CAN 2.0B, 2x I2C, 2x SPI, 4x 32b GPIO
- **Peripherals w/ built-in DMA**: 2x USB 2.0 (OTG), 2x Tri-mode Gigabit Ethernet, 2x SD/SDIO
- **Security**: RSA Authentication, and AES and SHA 256-bit Decryption and Authentication for Secure Boot
- **Processing System to Programmable Logic Interface Ports**: 2x AXI 32b Master 2x AXI 32-bit Slave 4x AXI 64-bit/32-bit Memory AXI 64-bit ACP 16 Interrupts

*Continued next slide*
## Zynq-7000 SoC Features (2)

### Programmable Logic Resources

<table>
<thead>
<tr>
<th>Device Name</th>
<th>Z-7010</th>
<th>Z-7015</th>
<th>Z-7020</th>
<th>Z-7030</th>
<th>Z-7035</th>
<th>Z-7045</th>
<th>Z-7100</th>
</tr>
</thead>
<tbody>
<tr>
<td>Part Number</td>
<td>XC7Z010</td>
<td>XC7Z015</td>
<td>XC7Z020</td>
<td>XC7Z030</td>
<td>XC7Z035</td>
<td>XC7Z045</td>
<td>XC7Z100</td>
</tr>
<tr>
<td>Xilinx 7 Series Programmable Logic Equivalent</td>
<td>Artix-7 FPGA</td>
<td>Artix-7 FPGA</td>
<td>Artix-7 FPGA</td>
<td>Kintex-7 FPGA</td>
<td>Kintex-7 FPGA</td>
<td>Kintex-7 FPGA</td>
<td>Kintex-7 FPGA</td>
</tr>
<tr>
<td>Programmable Logic Cells (Approximate ASIC gates)</td>
<td>28K Logic Cells (-430K)</td>
<td>74K Logic Cells (-1.1M)</td>
<td>85K Logic Cells (-1.9M)</td>
<td>125K Logic Cells (-4.1M)</td>
<td>275K Logic Cells (-5.2M)</td>
<td>350K Logic Cells (-6.8M)</td>
<td>444K Logic Cells (-8.5M)</td>
</tr>
<tr>
<td>Look-Up Tables (LUTs)</td>
<td>17,600</td>
<td>46,200</td>
<td>53,200</td>
<td>78,600</td>
<td>171,900</td>
<td>218,600</td>
<td>277,400</td>
</tr>
<tr>
<td>Flip-Flops</td>
<td>35,200</td>
<td>92,400</td>
<td>106,400</td>
<td>157,200</td>
<td>343,800</td>
<td>437,200</td>
<td>554,800</td>
</tr>
<tr>
<td>Extensible Block RAM (#36 Kb Blocks)</td>
<td>240 KB (60)</td>
<td>380 KB (95)</td>
<td>560 KB (140)</td>
<td>1,060 KB (265)</td>
<td>2,000 KB (545)</td>
<td>2,180 KB (545)</td>
<td>3,020 KB (755)</td>
</tr>
<tr>
<td>Programmable DSP Slices (18x25 MACs)</td>
<td>80</td>
<td>160</td>
<td>220</td>
<td>400</td>
<td>900</td>
<td>900</td>
<td>2,020</td>
</tr>
<tr>
<td>Peak DSP Performance</td>
<td>100 GMACs</td>
<td>200 GMACs</td>
<td>276 GMACs</td>
<td>593 GMACs</td>
<td>1,334 GMACs</td>
<td>1,334 GMACs</td>
<td>2,622 GMACs</td>
</tr>
<tr>
<td>PCI Express® (Root Complex or Endpoint)</td>
<td>—</td>
<td>Gan2 x4</td>
<td>—</td>
<td>Gan2 x8</td>
<td>Gan2 x8</td>
<td>Gan2 x8</td>
<td>Gan2 x8</td>
</tr>
<tr>
<td>Analog Mixed Signal (AMS) / XADC</td>
<td>2x 12 bit, MOPS ADCs with up to 17 Differential Inputs</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Security</td>
<td>AES and SHA 256b for Boot Code and Programmable Logic Configuration, Decryption, and Authentication</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Zynq® UltraScale+™ MPSoCs

<table>
<thead>
<tr>
<th></th>
<th>CG Devices</th>
<th>EG Devices</th>
<th>EV Devices</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application Processor</td>
<td>Dual-core ARM® Cortex™-A53 MPCore™ up to <strong>1.3GHz</strong></td>
<td>Quad-core ARM Cortex-A53 MPCore up to <strong>1.5GHz</strong></td>
<td>Quad-core ARM Cortex-A53 MPCore up to <strong>1.5GHz</strong></td>
</tr>
<tr>
<td>Real-Time Processor</td>
<td>Dual-core ARM Cortex-R5 MPCore up to <strong>533MHz</strong></td>
<td>Dual-core ARM Cortex-R5 MPCore up to <strong>600MHz</strong></td>
<td>Dual-core ARM Cortex-R5 MPCore up to <strong>600MHz</strong></td>
</tr>
<tr>
<td>Graphics Processor</td>
<td></td>
<td>Mali™-400 MP2</td>
<td>Mali™-400 MP2</td>
</tr>
<tr>
<td>Video Codec</td>
<td></td>
<td></td>
<td>H.264 / H.265</td>
</tr>
<tr>
<td>Programmable Logic</td>
<td>103K–600K System Logic Cells</td>
<td>103K–1143K System Logic Cells</td>
<td>192K–504K System Logic Cells</td>
</tr>
<tr>
<td>Applications</td>
<td>Sensor Processing &amp; Fusion</td>
<td>Flight Navigation</td>
<td>Situational Awareness</td>
</tr>
<tr>
<td></td>
<td>Motor Control</td>
<td>Missile &amp; Munitions</td>
<td>Surveillance/Reconnaissance</td>
</tr>
<tr>
<td></td>
<td>Low-cost Ultrasound</td>
<td>Military Construction</td>
<td>Smart Vision</td>
</tr>
<tr>
<td></td>
<td>Traffic Engineering</td>
<td>Secure Solutions</td>
<td>Image Manipulation</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Networking</td>
<td>Graphic Overlay</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Cloud Computing Security</td>
<td>Human Machine Interface</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Data Center</td>
<td>Automotive ADAS</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Machine Vision</td>
<td>Video Processing</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Medical Endoscopy</td>
<td>Interactive Display</td>
</tr>
</tbody>
</table>
Zynq-7000 SoC Processor System

FPGAs
Zynq® UltraScale+™ MPSoCs: CG Block Diagram

Processing System
- Application Processing Unit
  - ARM® Cortex™ A53
  - NEON™ Floating Point Unit
- Memory
  - DDR4/3/L, LPDDR3/4
  - 32/64-Bit w/ECC
  - 256KB OCM with ECC
- System Functions
  - Multichannel DMA
  - Timers, WDT, Resets, Clocking & Debug
- High-Speed Connectivity
  - DisplayPort v1.2a
  - USB 3.0
  - SATA 3.1
  - PCIe® 3.0 / 2.0
- Real-Time Processing Unit
  - ARM Cortex™-R5
  - Vector Floating Point Unit
  - Memory Protection Unit
- Platform Management Unit
  - System Management
  - Power Management
  - Functional Safety
- Configuration and Security Unit
  - Config AES Encryption, Authentication, Secure Boot
  - Voltage/Temp Monitor
  - TrustZone
- Programmable Logic
  - Storage & Signal Processing
    - Block RAM
    - UltraRAM
    - DSP
  - System Monitor
  - General-Purpose I/O
    - High-Performance HP I/O
    - High-Density HD I/O
  - High-Speed Connectivity
    - GTH
    - PCIe® Gen4

FPGAs
Zynq-7000 SoC Logic Fabric
Series-7 CLBs, IOBs, etc. (as in Artix-7)
Configuration Interfaces

- Master – FPGA retrieves its own configuration from ROM after power-up
  - Serial or Parallel options

- Slave – FPGA configured by external source (i.e., a µP)
  - Serial or Parallel options
  - Used for dynamic reconfiguration
  - Can also read configuration memory contents

- Boundary Scan Interface
  - 4-wire IEEE standard serial interface for testing
  - Write and read access to configuration memory
    - Not available in all FPGAs
    - Used for dynamic partial reconfiguration
  - Interfaces to FPGA core
    - Not available in all FPGAs
    - Connections between Boundary Scan Interface and internal routing network and PLBs (Xilinx provides 2-4 of these ports)

- Other configuration interfaces in some FPGAs
Slave configuration modes

(a) Slave Serial Mode

(b) JTAG Mode

(c) Slave SelectMAP Mode
**Nexys4 DDR configuration options**

**Artix-7 100T bitstream is typically 30,606,304 bits**

1. **USB-JTAG**: PC connection via USB or JTAG
2. **Master SPI**: Program from “quad mode” flash memory (x1, x2, x4 width)
3. **USB/SD**: Program from micro SD card or USB memory stick
**FPGA Configuration Memory**

- PLB addressable
  - Good for partial reconfiguration
  - X-Y coordinates of PLB location to be written
    - Requires tag to identify which resources will be configured

- Frame addressable
  - Vertical or horizontal frame
  - Access to all PLBs in frame
    - Only portion of logic and routing resources accessible
    - Many frames to configure PLBs
      - Major address for column, minor address for frame

*Hybrid, i.e.:
- Virtex-4
- Virtex-5
- Virtex-6*
Daisy Chain Configuration

FPGAs
## Xilinx Configuration Interface Pins

<table>
<thead>
<tr>
<th>Name</th>
<th>Direction</th>
<th>Driver Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Dedicated Pins</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CCLK</td>
<td>Input/Output</td>
<td>Active</td>
<td>Configuration clock. Output in Master mode.</td>
</tr>
<tr>
<td>PROGRAM</td>
<td>Input</td>
<td>Asynchronous reset to configuration logic.</td>
<td></td>
</tr>
<tr>
<td>DONE</td>
<td>Input/Output</td>
<td>Active/Open-Drain</td>
<td>Configuration status and start-up control.</td>
</tr>
<tr>
<td>M2, M1, M0</td>
<td>Input</td>
<td>Configuration mode selection.</td>
<td></td>
</tr>
<tr>
<td>TMS</td>
<td>Input</td>
<td>Boundary-scan tap controller.</td>
<td></td>
</tr>
<tr>
<td>TCK</td>
<td>Input</td>
<td>Boundary-scan clock.</td>
<td></td>
</tr>
<tr>
<td>TDI</td>
<td>Input</td>
<td>Boundary-scan data input.</td>
<td></td>
</tr>
<tr>
<td>TDO</td>
<td>Output</td>
<td>Active</td>
<td>Boundary-scan data output.</td>
</tr>
<tr>
<td><strong>Dual Function Pins</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DIN (D0)</td>
<td>Input/Output</td>
<td>Active Bidirectional</td>
<td>Serial configuration data input.</td>
</tr>
<tr>
<td>D[0:7]</td>
<td>Input/Output</td>
<td>Active Bidirectional</td>
<td>Slave Parallel configuration data input, readback data output.</td>
</tr>
<tr>
<td>CS</td>
<td>Input</td>
<td>Directional</td>
<td>Chip Select (Slave Parallel only).</td>
</tr>
<tr>
<td>WRITE</td>
<td>Input</td>
<td>Active</td>
<td>Active Low write select, read select (Slave Parallel only).</td>
</tr>
<tr>
<td>BUSY/ DOUT</td>
<td>Output</td>
<td>Open-Drain/ Active</td>
<td>Busy/Ready status for Slave Parallel (open-drain). Serial configuration data output for serial daisy-chains (active).</td>
</tr>
<tr>
<td>INIT</td>
<td>Input/Output</td>
<td>Open-Drain</td>
<td>Delay configuration, indicate configuration clearing or error.</td>
</tr>
</tbody>
</table>
Configuration Techniques

• **Full** configuration & readback
  - Simple configuration interface
    • Internal automatic calculation of frame address
  - Long download time for large FPGAs

• **Partial** reconfiguration & readback
  - Only change portions of configuration memory with respect to reference design
    • Reduces download time for reconfiguration
  - Requires more complicated interface
    • Command Register (CMR)
    • Frame Length Register (FLR)
    • Frame Address Register (FAR)
    • Frame Data Register
      - Input (FDRI) – for download
      - Output (FDRO) – for readback *(note separate access)*

FPGAs
Full Configuration Example

- Dummy Word 0xFFFFFFFF
- Synchronize Word 0xAA995566
- CMD Write 0x30008001
  - Reset CRC 0x00000007
- FLR Write 0x30016001
  - FLR = 0x00000024
  - Frame length = 37 words
    - 1,184 bits ÷ 32 bits/word
- COR Write 0x30012001
  - COR Write 0x00003FE5
- IDCODE Write 0x3001C001
  - Device ID = 0x0140D093 (3S50)
- MASK Write 0x3000C001
  - MASK = 0x00000000
- CMD Write 0x30008001
  - Switch CCLK 0x00000009
- FAR Write 0x30002001
  - FAR = 0x00000000 (full config)
- CMD Write 0x30008001
  - Write CFG 0x00000001
- FDRI Write 0x30004000
  - # words to write 0x50003555

Xilinx ASCII Bitstream
Created by Bitstream I.32
Design name: s3mod7.ncd
Architecture: spartan3
Part: 3s50tq144
Date: Tue Sep 04 15:50:09 2007
Bits: 439264

start of actual configuration data