Overview of FPGA Interconnect

CSET 4650
Field Programmable Logic Devices

Dan Solarek
Programmable Interconnect

- In addition to programmable logic cells, FPGAs must have *programmable interconnect*.
- Structure and complexity of the interconnect is determined by the programming technology and architecture of the logic cell.
- Interconnect is typically aluminum-based metal layers:
  - Resistance of approximately 50 mΩ/square
  - Line capacitance of approximately 0.2 pF/cm
- Early FPGAs had two metal interconnect layers, but current, high density parts may have three or more metal layers.
Field-Programmable Gate Arrays

Requires some form of programmable interconnect at crossovers …

over simplified
Tradeoffs in FPGA Interconnect

- How are logic blocks arranged?
- How “rich” is interconnect between channels?
- How many wires will be needed between them?
- Are wires evenly distributed across chip?
- How should wires be segmented (short, long)?
- How long is the average wire?
- How much buffering do we add to wires?
Tradeoffs in FPGA Interconnect

- Programmability slows signals down …
  - are some wires specialized to long distances?
- How many inputs/outputs must be routed to/from each configurable logic block?
- What utilization are we willing to accept?
  - 20%? 50%? 90%?
Interconnect Comes With a Cost

SpartanXL XCS20XL
0.35μm 160 I/O
20K System Gates

Gate Array
0.5μm 160 I/O
50K System Gates
Routing: Choosing a Path

Routing is done by a software tool.

LEs hold previously placed functions

LE

wiring channel

switch

wire

LE
Routing Considerations

- **Global routing:**
  - Which combination of channels?

- **Local routing:**
  - Which wire in each channel?

- **Routing metrics:**
  - Net length
  - Delay
Programmable vs. Fixed Interconnect

- Switch adds delay
- Transistor off-state is worse in advanced technologies
- FPGA interconnect has extra length = added capacitance
Interconnect Strategies

- Some wires will not be utilized
- Congestion will not be same throughout chip
- Types of wires:
  - Short wires: local LE connections
  - Global wires: long-distance, buffered communication
  - Special wires: clocks, etc.
Paths in Interconnect

- Connections may be long and complex
- Long wires can help simplify

![Diagram of paths and wiring channels]
Interconnect architecture

- Connections from wiring channels to LEs.
- Connections between wires in the wiring channels.
Interconnect Richness

Within a channel:
- How many wires
- Length of segments
- Connections from LE to interconnect channel

Between channels:
- Number of connections between channels
- Channel structure
Segmented Wiring

Length 1

Length 2
Offset Segments
Switchbox

- Multiple switch points
- Increased flexibility
Actel FPGAs

Rows of programmable logic building blocks

+ rows of interconnect

Anti-fuse Technology: Program Once

Use Anti-fuses to build up long wiring runs from short segments

8 input, single output combinational logic blocks
FFs constructed from discrete cross coupled gates
Actel Programmable Interconnect

- Actel interconnect is similar to a channeled gate array
  - Horizontal routing channels between rows of logic modules
  - Vertical routing channels on top of cells
- Each channel has a fixed number of tracks each of which holds one wire
- Wires are divided into segments of various lengths
  - *segmented channel routing*
- Long vertical tracks (LVT) extend the entire height of the chip
Actel Programmable Interconnect

- Each logic module has connections to its inputs and outputs called stubs
  - Input stubs extend vertically into routing channels above and below logic module
  - Output stub extends vertically 2 channels up and 2 channels down
- Wires are connected by antifuses
Actel Interconnect

Interconnection Fabric
Actel Routing Example

jogs cross an anti-fuse

minimize the number of jogs for speed critical circuits

2 - 3 jogs for most interconnections
Metal to Metal Antifuse

Metal to metal antifuse moved the antifuse out of silicon making the part denser and faster.
Metal to Metal Antifuse

TWO DIMENSIONAL

MODULES

TRACKS

SEA OF MODULES
Actel Programmable Interconnect

Each LM has 8 inputs: 4 input stubs on top and 4 on bottom. 

Output stubs are connected to long vertical track (LVT).

Routing channels: 7 or 13 (A1010/20) full-size and 2 half-size (top and bottom).

Logic Modules (LM): 8 or 14 (A1010/20) rows of 44 modules.

ActelACT

Each LM output drives an output stub that spans 2 channels up and 2 channels down.

two-antifuse connection

four-antifuse connection

input stub

antifuse
Detail of ACT1 Channel Architecture

ACT 1 horizontal and vertical channel architecture

5 vertical tracks: 4 tracks for output stubs, 1 track for long vertical track (LVT)

8 vertical tracks for input stubs

Logic Module (LM)

channel height

track number

module height

column width

expanded view of part of the channel

25 horizontal tracks per channel, varying between 1 column and 44 columns long: 22 signal tracks, global clock, VDD, and GND

programmed antifuse
dedicated connection to module output—no antifuse needed
Routing Resources

ACT 1 interconnection architecture

- 22 horizontal tracks per channel for signal routing with 3 dedicated for VDD, GND, GCLK
- 8 vertical tracks per LM are available for inputs (4 from the LM above the channel, 4 from the LM below)
  - input stub
- 4 vertical tracks per LM for outputs – output stub
  - a vertical track extends across the two channels above the module and the two channels below
- 1 long vertical track (spans the entire height of the chip)
Elmore’s Constant

- Approximation of waveform at node $i$:
  
  $V_i(t) = e^{-t/\tau_{Di}}$; \quad $\tau_{Di} = \sum_{k=1}^{n} R_{ki} C_k$

  where $R_{ki}$ is the resistance of the path to $V_0$ shared by node $k$ and node $i$

- Examples: $R_{24} = R_1$, $R_{22} = R_1 + R_2$, and $R_{31} = R_1$

- If the switching points are assumed to be at the 0.35 and 0.65 points, the delay at node $i$ can be approximated by $\tau_{DI}$

Measuring the delay of a net. (a) An RC tree. (b) The waveforms as a result of closing the switch at $t=0$. 
Elmore’s Constant

- $\tau_{\text{DI}}$ is the Elmore time constant
- It serves as a reminder that, if we approximate $V_i$ by an exponential waveform, the delay of the RC tree using 0.35/0.65 trip points is approximately $\tau_{\text{DI}}$ seconds.
RC Delay in Antifuse Connections

Actel routing model. (a) A four-antifuse connection. L0 is an output stub, L1 and L3 are horizontal tracks, L2 is a long vertical track (LVT), and L4 is an output stub. (b) An RC-tree model. Each antifuse is modeled by a resistance and each interconnect segment is modeled by a capacitance.
RC Delay in Antifuse Connections

- $R_n$ - resistance of antifuse, $C_n$ - capacitance of wire segment

\[
\tau_{D4} = R_{14}C_1 + R_{24}C_2 + R_{34}C_3 + R_{44}C_4
= (R_1 + R_2 + R_3 + R_4)C_4 + (R_1 + R_2 + R_3)C_3 + (R_1 + R_2)C_2 + R_1C_1
\]

- If all antifuse resistances are approximately equal and much larger than the resistance of the wire segment, then: $R_1 = R_2 = R_3 = R_4$, and:

\[
\tau_{D4} = 4RC_4 + 3RC_3 + 2RC_2 + RC_1
\]

- A connection with two antifuses will generate a $3RC$ time constant, a connection with three antifuses will generate a $6RC$ time constant, and a connection with 4 antifuses will generate a $10RC$ time constant

- Interconnect delay grows quadratically ($\propto n^2$) as the number of antifuses $n$ increases
## Actel Routing Resources

<table>
<thead>
<tr>
<th>Actel FPGA routing resources</th>
<th>Horizontal tracks per channel, H</th>
<th>Vertical tracks per column, V</th>
<th>Rows, R</th>
<th>Columns, C</th>
<th>Total antifuses on each chip</th>
<th>H×V×R×C</th>
</tr>
</thead>
<tbody>
<tr>
<td>A1010</td>
<td>22</td>
<td>13</td>
<td>8</td>
<td>44</td>
<td>112,000</td>
<td>100,672</td>
</tr>
<tr>
<td>A1020</td>
<td>22</td>
<td>13</td>
<td>14</td>
<td>44</td>
<td>186,000</td>
<td>176,176</td>
</tr>
<tr>
<td>A1225A</td>
<td>36</td>
<td>15</td>
<td>13</td>
<td>46</td>
<td>250,000</td>
<td>322,920</td>
</tr>
<tr>
<td>A1240A</td>
<td>36</td>
<td>15</td>
<td>14</td>
<td>62</td>
<td>400,000</td>
<td>468,720</td>
</tr>
<tr>
<td>A1280A</td>
<td>36</td>
<td>15</td>
<td>18</td>
<td>82</td>
<td>750,000</td>
<td>797,040</td>
</tr>
</tbody>
</table>
Xilinx LCA Interconnect

- Xilinx LCA interconnect has a hierarchical architecture:
  - *Vertical lines* and *horizontal lines* run between CLBs
  - *General-purpose interconnect* joins *switch boxes* (also known as *magic boxes* or *switching matrices*)
  - *Long lines* run across the entire chip - can be used to form internal buses using the three-state buffers that are next to each CLB
  - *Direct connections* bypass the switch matrices and directly connect adjacent CLBs
  - *Programmable Interconnect Points* (PIPs) are programmable pass transistors the connect CLB inputs and outputs to the routing network
  - *Bi-directional interconnect buffers* (BIDI) restore the logic level and logic strength on long interconnect paths
Xilinx FPGA Internals

- Portion of a Xilinx 4000 FPGA
- Shows relative sizes of major elements
- Need more detail about interconnect architecture
Xilinx 4000 Interconnect

- A closer look
- Programmable switch matrices
- Single length lines between adjacent PSMs
- Double length lines skip a PSM
Switch Detail and Scale

- CLBs in a sea of interconnect
- Programmable Switch Matrix (PSM)
- Connections are controlled by SRAM bits
- Long lines
- Global lines
Programmable Switch Matrix

Double
Singles
Double

Six Pass Transistors Per Switch Matrix Interconnect Point

Programmable Switch Matrix (PSM)
Programmable Switch Matrix

(a) Switch Matrix Transistors

(b) Examples of Connections
Pass Transistor Control
Programmable Switch Matrix

programmable switch element

turning the corner, etc.
Xilinx LCA Interconnect (cont.)

(a) The LCA architecture (notice the matrix element size is larger than a CLB). (b) A simplified representation of the interconnect resources. Each of the lines is a bus.
Components of interconnect delay in a Xilinx LCA array. (a) A portion of the interconnect around the CLBs. (b) A switching matrix. (c) A detailed view inside the switching matrix showing the pass-transistor arrangement. (d) The equivalent circuit for the connection between nets 6 and 20 using the matrix. (e) A view of the interconnect at a Programmable Interconnection Point (PIP. (f) and (g) The equivalent schematic of a PIP connection (h) The complete RC delay path.
Routing Connections

A connection is realized in an FPGA interconnect fabric by enabling routing switches in the connection and switch boxes.
Routing Connections

The parasitic contribution from the switches (realized as pass transistors) and the metal trace constitute the total resistive and capacitive components of the interconnect.
Routing Connections

Based on the switch and wire parasitic, interconnect routes can be modeled as $RC$ networks.

For typical parasitic values, $R_{wire}$ is so negligible when compared to $R_{on}$, and thus can be dropped.
Routing Connections

The capacitance of a route segment is given by:

\[ C_{seg} = 10C_{diff} + C_{wire} \]

This can be used to model the energy of the route as

\[ Energy (E) \propto 50C_{diff} + 4C_{wire} \]

The delay of the route can be compute as follows:

\[ Delay (D) \propto 10R_{on}C_{wire} + 125R_{on}C_{diff} \]

This modeling of the interconnect can be used to compute the cost of the architectural modifications.
Xilinx EPLD Interconnect

- Xilinx EPLD family uses an interconnect bus called a Universal Interconnection Module (UIM)
- UIM is a programmable AND array with constant delay from any input to any output

The Xilinx EPLD UIM (Universal Interconnection Module). (a) A simplified block diagram of the UIM. The UIM bus width, n, varies from 68 (XC7236) to 198 (XC73108). (b) The UIM is actually a large programmable AND array. (c) The parasitic capacitance of the EPROM cell.
Altera MAX 5K & 7K Interconnect

- Altera MAX 5000 and 7000 devices use a Programmable Interconnect Array (PIA)
- PIA is also a programmable AND array with constant delay from any input to any output

A simplified block diagram of the Altera MAX interconnect scheme. (a) The PIA (Programmable Interconnect Array) is deterministic - delay is independent of the path length. (b) Each LAB (Logic Array Block) contains a programmable AND array. (c) Interconnect timing within a LAB is also fixed.
Altera MAX 9K Interconnect Architecture

Altera MAX 9000 devices use long row and column wires (FastTracks) connected by switches.

The Altera MAX 9000 interconnect scheme. (a) A 4 X 5 array of Logic Array Blocks (LABs), the same size as the EMP9400 chip. (b) A simplified block diagram of the interconnect architecture showing the connection of the FastTrack buses to a LAB.
Altera Flex devices also use FastTracks connected by switches, but the wiring is more dense (as are the logic modules).

The Altera FLEX interconnect scheme. (a) The row and column FastTrack interconnect. (b) A simplified diagram of the interconnect architecture showing the connections between the FastTrack buses and a LAB.
Summary

- Antifuse FPGA architectures are dense and regular
- SRAM architectures contain nested structures of interconnect resources
- Complex PLD architectures use long interconnect lines but achieve deterministic routing