Digital Design and Synthesis

COEN 6501
Lecture_1

In this lecture we will review:
The Digital Design process
Introduce and review Adders
a) The Carry Ripple Through Adder
b) The Carry Look Ahead Adder
System Design Description

Systems are described in terms of three domains:

Behavioural domain
Structural domain
Physical domain
## Optimization Levels

<table>
<thead>
<tr>
<th>Level</th>
<th>Transformation</th>
<th>Expected Power Saving</th>
</tr>
</thead>
<tbody>
<tr>
<td>Algorithmic</td>
<td>Algorithm selection</td>
<td>Orders of magnitude</td>
</tr>
<tr>
<td>Behavioural</td>
<td>Concurrency</td>
<td>Several times</td>
</tr>
<tr>
<td>Register Transfer Level</td>
<td>Structural transformations</td>
<td>~10 - 15%</td>
</tr>
<tr>
<td></td>
<td>Clock control</td>
<td>~10 - 90%</td>
</tr>
<tr>
<td>Data/signal encoding</td>
<td></td>
<td>~20%</td>
</tr>
<tr>
<td>Technology independent</td>
<td>Extraction/decomposition</td>
<td>~15%</td>
</tr>
<tr>
<td>Technology dependant</td>
<td>Technology mapping</td>
<td>~20%</td>
</tr>
<tr>
<td></td>
<td>Gate sizing</td>
<td>~20%</td>
</tr>
<tr>
<td>Layout</td>
<td>Placement</td>
<td>20%</td>
</tr>
</tbody>
</table>
Design Process:

It starts with behavioural description, decomposing the high level of constructs into more precise functional units, then mapping these units into physical elements.
Design Strategies

Hierarchy

- A repeated process of dividing large modules into smaller sub-modules until the complexity of sub-modules are at an appropriately comprehensible level of detail.
- Parallel hierarchy is implemented in all domains.
A Structured Design

- **Regularity**
  - Divide the hierarchy into similar building blocks whenever possible.
  - Some programmability could be added to achieve regularity.

- **Modularity**
  - Well defined behavioural, structural and physical interface.
  - Helps: divide tasks into well defined modules, design integration, aids in team design.

- **Locality**
  - Internals of the modules are unimportant to any exterior interface.
System Design Methodology

Market Analysis
- Market windows
- System features & requirements
- Standards

System Specifications
- Functional
- Electrical
- Mechanical
- Environmental

System Architecture
- Strategies
- Modelling
- Verification

System Partitioning
- Dictated by complexity, I/O pins, off-the-shelf components, special requirements
- Partitioning guidelines
- Partitioning approaches: vertical, horizontal, functional, performance
Testability

- Strategies, chip testing, board testing
- Testability features
- Penalties

Technology Selection

- Dictated by: speed, power dissipation, driving capability, cost, lead time

Detailed Design

- Logic design/synthesis
- Optimization
- Verification

Implementation

- Off-the-shelf ICs
- Application Specific ICs
Assembly

- Decide on packaging technical components
- Design/manufacture
- Components
- Electrical/mechanical assembly
- Mechanical assembly & components sales

Testing

- Functional
- DC test
- AC test
- Burn-in

Documentation

- Technical documents
- H/W & S/W & mechanical
- User manual
- Test document

Production
Verify at every step
IC Design Methodology

Requirement specification

- most important function which impacts the ultimate success of an IC relates to how firm and clear the device specifications are.

- Device specification may be updated throughout the design cycle.

- Main items in the specifications are:
  - functional intent: brief description of the device, the technology and the task it performs.
  - Packaging specification
    - device port number
    - package type, dimension, material
Functional Description

- Functional description

  - high-level block diagram: all major blocks including intra block connections and connections to pin-outs indicating direction and signal flow.

  - Intra block signal function: description of how blocks interact with each other supported with timing diagram where necessary.

  - Internal block description of internal operation of each block. Where necessary, the following to be included: timing diagram, state diagram, truth table.
Specifications

- **I/O specifications**
  - pin-out diagram
  - I/O functional description
  - loading
  - ESD requirements
  - latch-up protection

- **D.C. specifications**
  - absolute maximum ratings for: supply voltage, pin voltages
  - main parameters: VIL and VIH for each input, VOL and VOH for each output, input loading, output drive, leakage current for tri-state operation, quiescent current, power-down current (if applicable)
Specification, continued

- **AC specifications**
  - inputs: set-up and hold times, rise and fall times
  - outputs: propagation delays, rise and fall times, relative timing
  - critical thinking

- **Environmental requirements**
  - operating temperature, storage temperature, humidity condition (if applicable)

- **Testing**
Device Specification

- Functional intent: briefly describe the device, the technology, and the circuits it will replace as well as the task it will perform.

- Design concept
  - Pin-out diagram: describe the device using a block diagram of the external view of the chip - basically, a box with all the I/O pins labelled and numbered
  - I/O description: use a chart to define the I/O signals shown in the pin-out diagram
Example:

<table>
<thead>
<tr>
<th>Pin #</th>
<th>Pin Name</th>
<th>I/O Type</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>P1</td>
<td>V&lt;sub&gt;DD&lt;/sub&gt;</td>
<td>Power Supply</td>
<td>Power Supply, +5V dc with respect to V&lt;sub&gt;ss&lt;/sub&gt;</td>
</tr>
<tr>
<td>P2</td>
<td>TXCLK</td>
<td>Input</td>
<td>Transmit Clock, 5.12 MHz rate</td>
</tr>
<tr>
<td>P3</td>
<td>TXP1</td>
<td>Output</td>
<td>Transmit output – channel 1, +ve polarity</td>
</tr>
</tbody>
</table>
Functional Specification

- Internal block diagram: draw blocks for major functions, show all connections including: connection to all pin-outs, connections between blocks, and direction of signal flow
- Inter-block signal function: describe how the blocks interact with each other and support this with timing diagrams where necessary
- Internal block description: describe the internal operation of each block. When necessary, include: timing diagrams, state diagrams, and truth table
- Logic description: circuit schematic or logic diagram using standard cell library components
- Package description: device port number, package type, dimensions, materials
Operating characteristics
Absolute maximum stress ratings.

Example:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Symbol</th>
<th>Min.</th>
<th>Max.</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Storage T</td>
<td>$T_s$</td>
<td>-65</td>
<td>+150</td>
<td>°C</td>
</tr>
<tr>
<td>Operating T</td>
<td>$T_A$</td>
<td>-40</td>
<td>+85</td>
<td>°C</td>
</tr>
<tr>
<td>Supply V</td>
<td>$V_{DD}$</td>
<td>-0.5</td>
<td>7</td>
<td>V</td>
</tr>
<tr>
<td>Input V</td>
<td>$V_I$</td>
<td>-0.3</td>
<td>$V_{DD} + 3$</td>
<td>V</td>
</tr>
<tr>
<td>Supply I</td>
<td>$I_{DD}$</td>
<td></td>
<td>5</td>
<td>mA</td>
</tr>
</tbody>
</table>
Requirements

1. Operating power and environmental requirement:

   - power supply voltage
   - operating supply current (specify conditions, e.g., power up, power down, frequency, output conditions)
   - storage temperature
   - operating temperature
   - humidity conditions (if applicable)
Input characteristics. Example chart:
(V reference is VSS = 0, temperature range is 0°C to 70°C)

<table>
<thead>
<tr>
<th>Pins</th>
<th>Symbol</th>
<th>Parameter</th>
<th>Min</th>
<th>nom</th>
<th>Max</th>
<th>Units</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>TXDAT2</td>
<td>VIL</td>
<td>Input low V</td>
<td>-0.3</td>
<td>0.4</td>
<td>0.8</td>
<td>V</td>
<td></td>
</tr>
<tr>
<td>TXDAT2</td>
<td>VIH</td>
<td>Input high V</td>
<td>2.0</td>
<td>2.4</td>
<td>VCC +0.3</td>
<td>V</td>
<td>Inputs protected against static damage</td>
</tr>
<tr>
<td>TXCK</td>
<td>CI</td>
<td>Input C to VSS</td>
<td>10</td>
<td>pF</td>
<td></td>
<td>pF</td>
<td>V</td>
</tr>
<tr>
<td>TXFRM</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ENB1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Inputs protected against static damage</td>
</tr>
<tr>
<td>ENB2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>V</td>
</tr>
<tr>
<td>ICK</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Vin = 0V</td>
</tr>
<tr>
<td>LFPM</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CSBL</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RX1N1</td>
<td>IIL</td>
<td>Input low I</td>
<td>+/- 10</td>
<td>A</td>
<td></td>
<td></td>
<td>Vin = 5.25V</td>
</tr>
<tr>
<td>RX1N2</td>
<td>IIH</td>
<td>Input high I</td>
<td>+/- 10</td>
<td>A</td>
<td></td>
<td></td>
<td>AC coupled input</td>
</tr>
<tr>
<td>RX1N1</td>
<td>VIP</td>
<td>Input peak V</td>
<td>VDD +0.3</td>
<td>V</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Output Interface Characteristics

Example chart: (VSS = 0, T range 0oC to 70oC)

<table>
<thead>
<tr>
<th>Pin names</th>
<th>Parameter</th>
<th>Symbol</th>
<th>min</th>
<th>max</th>
<th>units</th>
<th>Test condns</th>
</tr>
</thead>
<tbody>
<tr>
<td>LABUS &lt;0..15&gt;,</td>
<td>High level Vout</td>
<td>VOH</td>
<td>$V_{DD} - 0.1$</td>
<td></td>
<td>volt</td>
<td>IO&lt;=1(\mu)A</td>
</tr>
<tr>
<td>RABUS &lt;0..15&gt;</td>
<td>Low level Vout</td>
<td>VOL</td>
<td>0.1</td>
<td></td>
<td>volt</td>
<td>IO&lt;=1(\mu)A</td>
</tr>
<tr>
<td></td>
<td>High level Iout</td>
<td>IOH</td>
<td>0.25</td>
<td>1.6</td>
<td>mA</td>
<td>$V_O = 4.6V$</td>
</tr>
<tr>
<td></td>
<td>Low level Iout</td>
<td>IOL</td>
<td>1.6</td>
<td>3.4</td>
<td>mA</td>
<td>$V_O = 2.5V$</td>
</tr>
<tr>
<td></td>
<td>High level tristate Iout leakage</td>
<td>IOIH</td>
<td>10</td>
<td></td>
<td>microA</td>
<td>$V_O = V_{DD}$</td>
</tr>
<tr>
<td></td>
<td>Low level tristate Iout leakage</td>
<td>IOIL</td>
<td>10</td>
<td></td>
<td>microA</td>
<td>$V_O = 0V$</td>
</tr>
<tr>
<td>Cout</td>
<td>CO</td>
<td>10</td>
<td></td>
<td></td>
<td>pF</td>
<td></td>
</tr>
</tbody>
</table>
AC description
Timing diagram: include well-labelled signal drawings of all significant input and output relationships, rise and fall times, data set-up and hold times. Indicate the voltage range over which timing must be guaranteed.

Definitions:

- Cout
- input
- output

- VIH
- VIL
- Set-up
- hold
Example: timing diagram and chart
<table>
<thead>
<tr>
<th>Pins</th>
<th>Symbol</th>
<th>Parameter</th>
<th>Min</th>
<th>Nom</th>
<th>Max</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>RXCK</td>
<td>t19</td>
<td>Clock high</td>
<td>68</td>
<td>110</td>
<td>110</td>
<td>ns</td>
</tr>
<tr>
<td></td>
<td>t20</td>
<td>Clock low</td>
<td>68</td>
<td>110</td>
<td>110</td>
<td>ns</td>
</tr>
<tr>
<td></td>
<td>t16</td>
<td>Period</td>
<td></td>
<td></td>
<td>195.3125</td>
<td>ns</td>
</tr>
<tr>
<td></td>
<td>t16</td>
<td>Period</td>
<td>194</td>
<td>197</td>
<td>197</td>
<td>ns</td>
</tr>
<tr>
<td></td>
<td>t22</td>
<td>RXIN to RXCK delay</td>
<td>90</td>
<td></td>
<td></td>
<td>ns</td>
</tr>
<tr>
<td>RXFRM</td>
<td>t17</td>
<td>Frame delay</td>
<td></td>
<td></td>
<td></td>
<td>ns</td>
</tr>
<tr>
<td></td>
<td>t18</td>
<td>Frame hold</td>
<td></td>
<td></td>
<td></td>
<td>ns</td>
</tr>
</tbody>
</table>
Critical Path

- Signal paths with ‘tight’ timings (if applicable)
- Potential ‘race’ conditions (if applicable)
- Any set of paths with the same source and destination such as a clock signal and its complement (if applicable)
Test Description

- Test strategy: written description of functions to be tested. This section is a guide for determining and explaining simulation patterns
- Simulation input/output patterns: timing diagrams which include stimulus to be applied to input pins and the expected response on the output pins
Example:

Multiplicand = $10001001_2 = 89_{16}$
Multiplier = $10101011_2 = AB_{16}$
Expected Result = $1011011100000111_2 = 5B83_{16}$
System Level Design

- Top down approach
- Using behavioural constructs, top level architecture is defined
- Design validation is technology independent
- Use HDL to model the design (e.g., VHDL and Verilog)
- RTL is efficient for describing data flow
System Level design (Continued)

- Timing verification is difficult unless structure logic is defined

- VHDL representation can be changed into structural logic through - manual design, design synthesis: automated process which involves the conversion of VHDL/RTL into a set of registers and combinational circuits
Synthesis report

**Report:** fpga
**Design:** SQUARER_8
**Version:** V-2004.06-SP1
**Date:** Tue Mar 7 21:00:01 2006

---

**Xilinx FPGA Design Statistics**

---

<table>
<thead>
<tr>
<th>Description</th>
<th>Quantity</th>
</tr>
</thead>
<tbody>
<tr>
<td>FG Function Generators</td>
<td>203</td>
</tr>
<tr>
<td>H Function Generators</td>
<td>0</td>
</tr>
<tr>
<td>Number of CLB cells</td>
<td>232</td>
</tr>
<tr>
<td>Number of Hard Macros and Other Cells</td>
<td>0</td>
</tr>
<tr>
<td>Number of CLBs in Other Cells</td>
<td>0</td>
</tr>
<tr>
<td>Total Number of CLBs</td>
<td>232</td>
</tr>
<tr>
<td>Number of Ports</td>
<td>28</td>
</tr>
<tr>
<td>Number of Clock Pads</td>
<td>1</td>
</tr>
<tr>
<td>Number of IOBs</td>
<td>27</td>
</tr>
<tr>
<td>Number of Flip Flops</td>
<td>29</td>
</tr>
<tr>
<td>Number of 3-State Buffers</td>
<td>0</td>
</tr>
<tr>
<td>Total Number of Cells</td>
<td>260</td>
</tr>
</tbody>
</table>
Area report after Synthesis

Report : area
Design : SQUARER_8
Version: V-2004.06-SP1
Date : Tue Mar  7 21:00:01 2006

Library(s) Used:

  xfpga_4000e-3 {File: /CMC/tools/xilinx.vM3.li/synopsys/libraries/syn/xfpga_4000e-3.db}
  xprim_4010e-3 {File: /CMC/tools/xilinx.vM3.li/synopsys/libraries/syn/xprim_4010e-3.db}

Number of ports: 28
Number of nets: 117
Number of cells: 46
Number of references: 20

Combinational area: 0.000000
Noncombinational area: 259.000000
Net Interconnect area: undefined (Wire load has zero net area)

Total cell area: 259.000000
Total area: undefined
Power report after Synthesis

Report: power
   -analysis_effort low
Design: SQUARER_8
Version: V-2004.06-SP1
Date: Tue Mar 7 21:00:03 2006

Library(s) Used:

xfpga_4000e-3 (File:/CMC/tools/xilinx.v3.1i/synopsys/libraries/syn/xfpga_4000e-3.db)
xprim_4010e-3 (File:/CMC/tools/xilinx.v3.1i/synopsys/libraries/syn/xprim_4010e-3.db)

Warning: The library cells used by your design are not characterized for internal power. (FWR-2

Operating Conditions: WCCOM Library: xprim_4010e-3
Wire Load Model Mode: top

Design Wire Load Model Library
-----------------------------
SQUARER_8 4010e-3_avg xprim_4010e-3

Global Operating Voltage = 4.75
Power-specific unit information :
   Voltage Units = 1V
   Capacitance Units = 1.000000pf
   Time Units = 1ns
   Dynamic Power Units = 1mW   (derived from V,C,T units)
   Leakage Power Units = Unilless

Cell Internal Power = 0.0000 mW  (0%)
Net Switching Power = 22.6160 mW (100%)
-------------
Total Dynamic Power = 22.6160 mW (100%)
Cell Leakage Power = 0.0000
### Timing Report

**After Synthesis**

**Operating Conditions:**
- **Library:** xprir4_gen3e-3
- **Version:** 2008.06-EPJ

**Startpoint:** SPREG1/SPREG2/DPF4/Q_INSIDE_reg
- Rising edge-triggered flip-flop clocked by CLK

**Endpoint:** DPF1/INSIDE_reg
- Rising edge-triggered flip-flop clocked by CLK

**Path:** CL1

---

<table>
<thead>
<tr>
<th>Point</th>
<th>Inor</th>
<th>Path</th>
</tr>
</thead>
<tbody>
<tr>
<td>clock CL1 (rise edge)</td>
<td>0.00</td>
<td>6.00</td>
</tr>
<tr>
<td>clock network delay (ideal)</td>
<td>0.00</td>
<td>6.00</td>
</tr>
<tr>
<td>SPREG1/SPREG2/DPF4/Q_INSIDE_reg/K (cib_4000)</td>
<td>0.00</td>
<td>6.00 r</td>
</tr>
<tr>
<td>SPREG1/SPREG2/DPF4/Q_INSIDE_reg/Q (cib_4000)</td>
<td>4.50</td>
<td>4.50 r</td>
</tr>
<tr>
<td>SPREG2/SPREG2/DPF4/Q (DPF_8)</td>
<td>0.00</td>
<td>4.50 r</td>
</tr>
<tr>
<td>SPREG2/SPREG2/INSIDE_reg (DFF_8)</td>
<td>0.00</td>
<td>4.50 r</td>
</tr>
<tr>
<td>Us/PAD (cib_4000)</td>
<td>8.50</td>
<td>13.00 r</td>
</tr>
<tr>
<td>MUX/S (MUX_2to1_3)</td>
<td>0.00</td>
<td>13.00 r</td>
</tr>
<tr>
<td>MUX/NOC1/A (MUX_2to1_3)</td>
<td>0.00</td>
<td>13.00 r</td>
</tr>
<tr>
<td>MUX/NOC1/UT/X (cib_4000)</td>
<td>3.15</td>
<td>16.15 r</td>
</tr>
<tr>
<td>MUX/NOC1/D (MUX_2to1_3)</td>
<td>0.00</td>
<td>16.15 r</td>
</tr>
<tr>
<td>MUX/NCE2/B (MUX_2to1_3)</td>
<td>0.00</td>
<td>16.15 r</td>
</tr>
</tbody>
</table>

---

**Additional Timing Parameters:**
- **Data arrival time:** 92.80
- **Clock CL1 (rise edge):** 96.00
- **Clock network delay (ideal):** 96.00
- **Clock uncertainty:** -0.05, 95.95
- **library setup time:** -2.42, 93.83
- **data required time:** 93.93
AIMs

What the CUSTOMER wants
- High Quality
- Low Cost
- Small Size/Weight

What the EMPLOYER wants
- Design the:
  - Best
  - Cheapest
  - In shortest time
  - Follow the Spec or better.

What you CHIP DESIGNER should do:
- Design a chip with:
  - High speed
  - Small area
  - Low power
  - Testable and reliable
  - Delivered in a short time
Logic Design

- Evaluation of library constructs (basic & macro) function, timing, area
- Logic minimization
- NAND/NOR transformation
- Buffering
- Fan-out reduction
- Fan-in reduction
Logic Level design (Continued)

- Critical timing
- Priority routing
- I/O compatibility
- Logic optimization
- Cost function: area, speed, power, or a combination
Logic Simulation

- Simulation is the process of exercising a theoretical model of the design as a function of time for some applied input sequence.

- Logic simulation is to aid in verification of a digital system.
Logic Simulation (Continued)

- Components
  - models: functional, timing
  - connectivity: a description of how the basic components are connected together
  - stimulus: 1’s and 0’s that are applied at specific times to the primary inputs of the design
  - simulation control

- States: basic (0, 1, X), strength could be combined with basic; strong (S), resistive (R), high impedance (Z), indeterminate (I)
Simulation model

- logical

*******************************************************************************
** Library:   ACME
** Technology:  2u CMOS
** Part:       fdrc
**
** Description:   D flip-flop with rising edge, async. Clear
*******************************************************************************

model fdrc: table
input d, rn;
edge_sense input cp;
output q, qn;
State table

<table>
<thead>
<tr>
<th>rn, cp, d, q</th>
<th>::</th>
<th>q, qn;</th>
</tr>
</thead>
<tbody>
<tr>
<td>0, (??), ?, ?</td>
<td>::</td>
<td>0, 1,</td>
</tr>
<tr>
<td>1, (01), ?, ?</td>
<td>::</td>
<td>(d), !(d);</td>
</tr>
<tr>
<td>1, (?0), ?, ?</td>
<td>::</td>
<td>N, !(q);</td>
</tr>
<tr>
<td>1, (1?), ?, ?</td>
<td>::</td>
<td>N, !(q);</td>
</tr>
</tbody>
</table>
end (fdrc: table);
Timing Verification

- Process of making accurate delay prediction and to detect timing violation in the design. These violations include set-up time, hold time, races and spikes.

- Delay through the circuit is a function of:
  - intrinsic delay
  - number of loads connected to each net
  - temperature
  - voltage
  - process variation, layout

- Typically, best and worst case scenarios should be considered.
Simulator uses a set of equations to calculate exact delays

ô Fan-out

\[ td = t(int) + K \times L \]
\[ t(int) = \text{intrinsic delay} \]
\[ K = \text{drive factor} \]
\[ L = \text{sum of equivalent loads} \]
Timing Verification (Continued)

- **temperature**
  \[ td = \frac{td}{FT} \]
  \[ FT = \frac{T2}{T1} \]

- **voltage**
  \[ t'd = \frac{td}{[VDDr(1 + 0.0f)]} \]

- **process**
  \[ t'd = td(1 + 0.01Fp) \]
  \[ Fp = \text{processing variation factor} \]

- **layout information** is normally supplied in two forms:
  - **pre-layout estimation**
  - **post-layout**: back annotation
Timing

hazards
- spikes: inertial and transport delays

\[ t_{PLH} = 2 \]
\[ t_{PHL} = 1 \]

set-up time/hold time/minimum pulse width
Timing

Critical path analysis
- detection of timing violation for data path structure
- the process is simply adding up path delays and compute the result with the period of the clock at the destination (F/F)
- path analysis is not simulation and does not utilize information about the functionality of the device
- look for two parameters
  - hold slack = clock period - hold path time
  - set up slack = clock period - set up path time
  - slack >= 0
  - paths are chosen to provide the least amount of available set up or hold times
Structural layout synthesis

- **Floor planning**
  - it is the exercise of arranging blocks of layout within a chip to minimize area or to maximize speed
  - floor plan editors provide graphical feedback about the size and placement of modules (without showing details), also the connectivity information between the modules in the form rat’s-not
  - floor planning could be done manually, or automatically with manual intervention
  - factors influencing floor planning (core & I/Os)
Placement and routing

- Placement: is the task of placing modules adjacent to each other to minimize area or cycle time
- Two algorithms: min-cut, simulated annealing
- Routing: a router takes a module placement and a list of connections, connects the modules with wires
- Types of routers: channel, switch box, maze
Channel route

Channel route
Layout

- Other layout tools
  - synthesis
  - compaction

- Layout verification
  - design rule checking
  - layout extraction
  - layout vs. schematic

- Back annotation of post layout simulation
Testing

- to verify the correct operation of the device by exercising it by a set of test patterns, and then to check the output patterns to see whether they are identical to the ones predicted by the simulator.

- tester also verifies DC and AC parameters on the pins of the device.
Timing Analysis

- Tester operates in a periodic fashion
- Input signals charge states at the beginning of the test period
- Output signals are strobed at the end of the period to determine whether the measured values matches the simulated values.
Types of Testing

- **Functional** (mostly at lower speeds)
  - static
  - dynamic (refresh if required)

- **DC test**
  - continuity
  - leakage, power consumption
  - high/low voltage levels, drive capability

- **AC test**
  - rise/fall times, propagation delays
  - set-up and hold times, access times
COEN 7741  
Advance Comp. Arch

COEN 7501  
Formal Verification

ENCS 6521  
Design for Testability

COEN 6531  
ASIC Synthesis

ENCS 6511

ELEC 6501

ELEC 6231

ELEC 6241

LOGIC  
CIRCUIT  
LAYOUT  
FABRICATION
## Binary Arithmetic

<table>
<thead>
<tr>
<th>Operation</th>
<th>Unsigned</th>
<th>Signed Magnitude</th>
<th>One’s Complement</th>
<th>Two’s Complement</th>
</tr>
</thead>
<tbody>
<tr>
<td>No change</td>
<td></td>
<td>If +ve then MSB = 0 else MSB = 1</td>
<td>If –ve then flip bits</td>
<td>If –ve then flip bits, add 1</td>
</tr>
<tr>
<td>3 =</td>
<td>0011</td>
<td>0011</td>
<td>0011</td>
<td>0011</td>
</tr>
<tr>
<td>-3 =</td>
<td>NA</td>
<td>1011</td>
<td>1100</td>
<td>11101</td>
</tr>
<tr>
<td>Zero =</td>
<td>0000</td>
<td>0000 or 1111 or 0000</td>
<td>0000</td>
<td>0000</td>
</tr>
<tr>
<td>Max. +ve =</td>
<td>1111 = 15</td>
<td>0111 = 7</td>
<td>0111 = 7</td>
<td>0111 = 7</td>
</tr>
</tbody>
</table>
Max. – ve = 0000 = 0

Addition = S = A + B

A + B = addend +
 augend SG(A)
 = sign of A

Addition result : OR = COUT(MSB)
 OV = overflow,
 OR = out of
 range

SG(S) = sign of S, S
 = A + B

1111 = -7
1000 = -7
1000 = -8

If SG(A) = SG(B) then S = A + B else {if B < A then S = A – B else S = B – A}

S = A + B + COUT(MSB)
COUT IS CARRY OUT

OV = XOR{COUT(MSB ), COUT(MSB – 1)}

OV = XOR{COUT(MSB ), COUT(MSB – 1)}

NA

NA

NA

NA
Subtraction = $D = A - B$

- minuend - subtrahend

$D = A - B$

SG(B) = NOT(SG(B)); $D = A + B$

$Z = -B$ (negate); D = A + Z

Subtraction result: OV = BOUT(MSB)

- As in addition

OR =

- As in addition

OR = out of range

Overflow, BOUT is borrow out

Negation: $Z = -A$ (negate)

NA $Z = A;$

$SG(Z) = NOT(SG(A))$

$Z = NOT(A) + 1$
Example: design an addition overflow circuit, in accordance with the following specification:

- When the operation is addition and both addend and augend are +ve, overflow is indicated by a carry from the most significant digit (MSD).
- When the operation is addition and both addend and augend are -ve, overflow is indicated by the absence of carry from the MSD.
- When the operation is subtraction and the minuend is +ve and the subtrahend -ve, overflow is indicated by a carry from the MSD.
- When the operation is subtraction and the minuend is -ve and subtrahend is +ve, overflow is indicated by absence of a carry from the MSD.
THE END