Network Algorithms, Lecture 2: Enough Hardware Knowledge to be Dangerous

Tom Edsall, Cisco
Outline

• Gates
• Combinatorial Logic
• Sequential Logic
• Memories
• Other Important Considerations
• Rules of Thumb
Gates
N-Channel MOSFET

Input voltage

Output voltage

Current

Vdd

Vss

N-Channel MOSFET
Current

None

Lots

Voltage

Vss

"0"

"Off"

"On"

Vdd

"1"

N-Channel MOSFET
In | Out
---|---
0 | 1
1 | 0
Combinatorial Logic
Common Logic Blocks

- **Multiplexor** – “mux”
  - connect input I of N inputs to output as selected by S (2:1, 4:1, 32:1 x 16, etc.). S is usually binary encoded.

- **De-multiplexor**
  - Opposite of mux – usually demuxes time interleaved data

- **Comparator** (=, <, >)

- **Arithmetic Unit** (+, -, *, /)
  - watch out for multiply and divide

- **Encoder/Decoder**
  - convert from N bits unary to log(N) bits binary and vice versa
    - 00100000 => 5 (encode)
    - 4 => 00010000 (decode)

- **Priority Encoder** – binary or unary result

- **Shifter, Barrel Shifter** – divide by $2^N$

- **Arbiter** - generic
Priority Encoder - Unary

<table>
<thead>
<tr>
<th>Input 543210</th>
<th>Output 543210</th>
</tr>
</thead>
<tbody>
<tr>
<td>XXXX1</td>
<td>000001</td>
</tr>
<tr>
<td>XXXX10</td>
<td>000010</td>
</tr>
<tr>
<td>XXX100</td>
<td>000100</td>
</tr>
<tr>
<td>XX1000</td>
<td>001000</td>
</tr>
<tr>
<td>X10000</td>
<td>010000</td>
</tr>
<tr>
<td>100000</td>
<td>100000</td>
</tr>
<tr>
<td>000000</td>
<td>000000</td>
</tr>
</tbody>
</table>

Out_0 = i_0
Out_1 = !i_0 & i_1
Out_2 = !i_0 & !i_1 & i_2
Out_3 = !i_0 & !i_1 & !i_2 & i_3
Out_4 = !i_0 & !i_1 & !i_2 & !i_3 & i_4
Out_5 = !i_0 & !i_1 & !i_2 & !i_3 & !i_4 & i_5

Note: “A & B” is the same as “A * B” is the same as “AB”
Out_5 = !i_0 \& !i_1 \& !i_2 \& !i_3 \& !i_4 \& i_5
Out_4 = !i_0 \& !i_1 \& !i_2 \& !i_3 \& i_4
Out_3 = !i_0 \& !i_1 \& !i_2 \& i_3
Out_2 = !i_0 \& !i_1 \& i_2
Out_1 = !i_0 \& i_1
Out_0 = i_0
PE – Small
if (i0) begin
    out <= 5'b000001
end else if (i1) begin
    out <= 5'b000010
end else if (i2) begin
    out <= 5'b000100
end else if (i3) begin
    out <= 5'b001000
end else if (i4) begin
    out <= 5'b010000
end else if (i5) begin
    out <= 5'b100000
else begin
    out <= 5'b000000
end
Sequential Logic

What about state?
Flip-Flop

- When clock goes high, data at D gets stored in FF and is available at Q
- Most versatile Storage Element
- Fastest storage element
- Area about 8-10 gates
Flip-Flop
Flip-Flop

- Common usage
- Data loaded when LOAD active
- Value does not change when LOAD inactive
Combining Logic and FF - FSM

- Finite State Machine
- Large Design may have 1,000’s
- Often design policy to use this form
- Lots of related tools
- Divide and conquer – large SMs broken into many small SMs
Timing

Priority Encoder

Clock:

REQ0
REQ1
REQ5

GNT0
GNT1
GNT2

Clock:

REQ: 000 101 10 010 111 100 000
GNT: 000 001 010 010 001
PEin: 000 101 110 010 111 100
PEout: 000 001 010 010 001 100
Startpoint: fi0_fp_free_drdy
  (input port clocked by clk)
Endpoint: fro_fp_logic_inst/fi0_free_addr_reg_1_
  (rising edge-triggered flip-flop clocked by clk)
Path Group: clk
Path Type: max

<table>
<thead>
<tr>
<th>Point</th>
<th>Incr</th>
<th>Path</th>
</tr>
</thead>
<tbody>
<tr>
<td>clock clk (rise edge)</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>clock network delay (ideal)</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>input external delay</td>
<td>5.00</td>
<td>5.00 r</td>
</tr>
<tr>
<td>fi0_fp_free_drdy (in)</td>
<td>0.04</td>
<td>5.04 r</td>
</tr>
<tr>
<td>fro_fp_logic_inst/fi0_fp_free_drdy</td>
<td>0.00</td>
<td>5.04 r</td>
</tr>
<tr>
<td>fro_fp_logic_inst/U6346/Z (N1BF)</td>
<td>0.06</td>
<td>5.10 f</td>
</tr>
<tr>
<td>fro_fp_logic_inst/U8343/Z (AO21CNDFP)</td>
<td>0.32</td>
<td>5.41 f</td>
</tr>
<tr>
<td>fro_fp_logic_inst/U8345/Z (N1DFP)</td>
<td>0.12</td>
<td>5.54 r</td>
</tr>
<tr>
<td>fro_fp_logic_inst/U5691/Z (MUX21HAFP)</td>
<td>0.24</td>
<td>5.78 f</td>
</tr>
<tr>
<td>fro_fp_logic_inst/U4654/Z (AND2AFP)</td>
<td>0.11</td>
<td>5.88 f</td>
</tr>
<tr>
<td>fro_fp_logic_inst/U3855/Z (NR2BNAFP)</td>
<td>0.13</td>
<td>6.01 f</td>
</tr>
<tr>
<td>fro_fp_logic_inst/fp_fi0_free_addr_reg_1_/D</td>
<td>0.00</td>
<td>6.01 f</td>
</tr>
<tr>
<td>data arrival time</td>
<td>6.01</td>
<td></td>
</tr>
<tr>
<td>clock clk (rise edge)</td>
<td>6.50</td>
<td>6.50</td>
</tr>
<tr>
<td>clock network delay (ideal)</td>
<td>0.00</td>
<td>6.50</td>
</tr>
<tr>
<td>clock uncertainty</td>
<td>-0.20</td>
<td>6.30</td>
</tr>
<tr>
<td>fro_fp_logic_inst/fp_fi0_free_addr_reg_1_/CP</td>
<td>0.00</td>
<td>6.30 r</td>
</tr>
<tr>
<td>library setup time</td>
<td>-0.26</td>
<td>6.04</td>
</tr>
<tr>
<td>data required time</td>
<td>6.04</td>
<td></td>
</tr>
<tr>
<td>data required time</td>
<td>6.04</td>
<td></td>
</tr>
<tr>
<td>data arrival time</td>
<td>6.04</td>
<td></td>
</tr>
</tbody>
</table>

slack (MET) | 0.03 |
Startpoint: ddr_ctl_out_i[0]  
(input port clocked by clk)  

Endpoint: F0_checkers_data_o_reg_132_  
(rising edge-triggered flip-flop clocked by clk)  

Path Group: clk  
Path Type: max  

<table>
<thead>
<tr>
<th>Point</th>
<th>Incr</th>
<th>Path</th>
</tr>
</thead>
<tbody>
<tr>
<td>clock clk (rise edge)</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>clock network delay (ideal)</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>input external delay</td>
<td>5.00</td>
<td>5.00 r</td>
</tr>
<tr>
<td>ddr_ctl_out_i[0] (in)</td>
<td>0.09</td>
<td>5.09 r</td>
</tr>
<tr>
<td>U10390/Z (N1DFP)</td>
<td>0.07</td>
<td>5.16 f</td>
</tr>
<tr>
<td>U8472/Z (ND3DFP)</td>
<td>0.05</td>
<td>5.21 r</td>
</tr>
<tr>
<td>U8453/Z (ND3DFP)</td>
<td>0.11</td>
<td>5.32 f</td>
</tr>
<tr>
<td>U8340/Z (OA22BFP)</td>
<td>0.24</td>
<td>5.56 f</td>
</tr>
<tr>
<td>U8332/Z (N1DFP)</td>
<td>0.06</td>
<td>5.62 r</td>
</tr>
<tr>
<td>U863/Z (AOI21CFP)</td>
<td>0.12</td>
<td>5.74 f</td>
</tr>
<tr>
<td>U8459/Z (OA111DFP)</td>
<td>0.14</td>
<td>5.88 r</td>
</tr>
<tr>
<td>U4/Z (AOI22AFP)</td>
<td>0.21</td>
<td>6.09 f</td>
</tr>
<tr>
<td>U855/Z (N1AFP)</td>
<td>0.13</td>
<td>6.22 r</td>
</tr>
<tr>
<td>U462/Z (ND2AFP)</td>
<td>0.11</td>
<td>6.33 f</td>
</tr>
<tr>
<td>U2962/Z (NR2BNBFP)</td>
<td>0.06</td>
<td>6.39 r</td>
</tr>
<tr>
<td>F0_checkers_data_o_reg_132_/D (FD1QAFP)</td>
<td>0.00</td>
<td>6.39 r</td>
</tr>
<tr>
<td>data arrival time</td>
<td>6.39</td>
<td></td>
</tr>
</tbody>
</table>

clock clk (rise edge)                           | 6.50 | 6.50|
| clock network delay (ideal)                     | 0.00 | 6.50|
| clock uncertainty                               | -0.20 | 6.30|
| F0_checkers_data_o_reg_132_/CP (FD1QAFP)        | 0.00 | 6.30 r|
| library setup time                              | -0.25 | 6.05|
| data required time                              | 6.05 |
| data arrival time                               | -6.39 |

slack (VIOLATED)  
-0.34
Common Building Blocks

- Register
- Shift register
- FIFO
- Counter
- Skid Buffer
Memories

What if I need a lot of Flip-flops?
SRAM

Word Line

BL

!BL

Q

!Q

D
SRAM

• Good density
• Pretty fast
• Mostly single port
• Size from 32 to 8K words by 8 to 256 bits
• Use multiple together to get larger configs
• We use 10’s of Mbytes
• On-chip
• Think “tables”
DRAM

- 8192 X 4096 Array
- Sense Amplifiers - 4096
- Transceiver
- Data
- Controller
  - Column Address
- Column Decoder - 4096
- Row Decoder - 8192
  - Row Address
- Latch
  - Address
DRAM

Controller

Row Decoder - 8192

Sense Amplifiers - 4096

Column Decoder - 4096

Latch

Address

Data

8192 x 4096 Array
Figure 5: 8 Meg x 36 Functional Block Diagram
Packet Buffer Example

• We want to build a router for 10Gbps Ethernet
• Need to have 25msec of buffering
• How can we implement the buffer?
• How many ports can we support with 1 ASIC and external memory?
CIO RLDRAM® II
MT49H32M9 – 32 Meg x 9 x 8 Banks
MT49H16M18 – 16 Meg x 18 x 8 Banks
MT49H8M36 – 8 Meg x 36 x 8 Banks

Features
- 533 MHz DDR operation (1.067 Gbps/s pin data rate)
- 38.4 Gbps peak bandwidth (x36 at 533 MHz clock frequency)
- Organization
  - 32 Meg x 9, 16 Meg x 18, and 8 Meg x 36
- 8 internal banks for concurrent operation and maximum bandwidth
- Reduced cycle time (15ns at 533 MHz)
- Nonmultiplexed addresses (address multiplexing option available)
- SRAM-type interface
- Programmable READ latency (RL), row cycle time, and burst sequence length
- Balanced READ and WRITE latencies in order to optimize data bus utilization
- Data mask for WRITE commands
- Differential input clocks (CK, CK#)
- Differential input data clocks (D1x8, D#x8)
- On-die DLL generates CK edge-aligned data and output data clock signals
- Data valid signal (QVLD)
- 32ms refresh (8k refresh for each bank; 64k refresh command must be issued in total each 32ms)
- 144-ball μPGA package
- HSTL I/O (1.5V or 1.8V nominal)
- 25–60Ω matched impedance outputs
- 2.5V VEXT, 1.8V VDD, 1.5V or 1.8V VDDQ I/O
- On-die termination (ODT) RTT

Options
- Clock cycle timing
  - 1.875ns @ 8C = 15ns
  - 2.5ns @ 8C = 15ns
  - 2.5ns @ 8C = 20ns
  - 3.3ns @ 8C = 20ns
  - 5.0ns @ 8C = 20ns
- Configuration
  - 32 Meg x 9
  - 16 Meg x 18
  - 8 Meg x 36
  - 32M9
  - 16M18
  - 8M36
- Operating temperature
  - Commercial (0C to +95C)
  - Industrial (Tc = -40C to +85C)
- Package
  - 144-ball μPGA
  - 144-ball μPGA (Pb-free)
- Revision
  - B

Notes: 1 Not all options listed can be combined to define an offered product. Use the part catalog search on www.micron.com for available offerings.

10Gbps link
25mSec of buffering
10Gbps * 25ms = 250Mbits

Need 250Mbits of buffering

10Gbps write + 10G Read = 20Gbps
75% overhead (may be too low)

Need 35Gbps

What about pins?
Table 4: Ball Descriptions

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>A0-A20</td>
<td>Input</td>
<td>Address inputs: A0–A20 define the row and column addresses for READ and WRITE operations. During a MODE REGISTER SET, the address inputs define the register settings. They are sampled at the rising edge of CK.</td>
</tr>
<tr>
<td>BA0-BA2</td>
<td>Input</td>
<td>Bank address inputs: Select to which internal bank a command is being applied.</td>
</tr>
<tr>
<td>CK, CK#</td>
<td>Input</td>
<td>Input clock: CK and CK# are differential input clocks. Addresses and commands are latched on the rising edge of CK. CK# is ideally 180 degrees out of phase with CK.</td>
</tr>
<tr>
<td>CS#</td>
<td>Input</td>
<td>Chip select: CS# enables the command decoder when LOW and disables it when HIGH. When the command decoder is disabled, new commands are ignored, but internal operations continue.</td>
</tr>
<tr>
<td>DK, DK#</td>
<td>Input</td>
<td>Input data clock: DK and DK# are the differential input data clocks. All input data is referenced to both edges of DK. DK# is ideally 180 degrees out of phase with DK. For the x36 device, D00–D15 are referenced to DK0 and DK0# and D16–D31, D32–D35 are referenced to DK1 and DK1#. For the x9 and x18 devices, all D0s are referenced to DK and DK#. All DK and DK# pins must always be supplied to the device.</td>
</tr>
<tr>
<td>DM</td>
<td>Input</td>
<td>Input data mask: The DM signal is the input mask signal for WRITE data. Input data is masked when DM is sampled HIGH. DM is sampled on both edges of DK (DK1 for the x36 configuration). Tie signal to ground if not used.</td>
</tr>
<tr>
<td>TCK</td>
<td>Input</td>
<td>IEEE 1149.1 clock input: This signal must be tied to VSS if the JTAG function is not used.</td>
</tr>
<tr>
<td>TMS, TDI</td>
<td>Input</td>
<td>IEEE 1149.1 test inputs: These pins may be left as no connections if the JTAG function is not used.</td>
</tr>
<tr>
<td>WEP, WE#</td>
<td>Input</td>
<td>Command inputs: Sensed at the positive edge of CK. WE# and WE# define (together with CS#) the command to be executed.</td>
</tr>
<tr>
<td>DQ0-DQ35</td>
<td>I/O</td>
<td>Data input: The DQ signals form the 36-bit data bus. During READ commands, the data is referenced to both edges of Qk. During WRITE commands, the data is sampled at both edges of DK.</td>
</tr>
<tr>
<td>ZQ</td>
<td>Reference</td>
<td>External impedance (25–60Ω): This signal is used to tune the device outputs to the system data bus impedance. DQ output impedance is set to 0.2 × Qk, where Qk is the resistor from this signal to ground. Connecting ZQ to GND invokes the minimum impedance mode. Connecting ZQ to VDD invokes the maximum impedance mode. Refer to Figure 10 on page 34 to activate this function.</td>
</tr>
<tr>
<td>Qk, Qk#</td>
<td>Output</td>
<td>Output data clocks: Qk and Qk# are opposite polarity, output data clocks. They are free-running, and during READs, are edge-aligned with data output from the RLDARAM. Qk# is ideally 180 degrees out of phase with Qk. For the x36 device, Qk0 and Qk0# are aligned with D00–D15, and Qk1 and Qk1# are aligned with D16–D31, D32–D35. For the x9 device, Qk0 and Qk0# are aligned with D00–D15, while Qk1 and Qk1# are aligned with Q9–Q17. For the x9 device, all Qs are aligned with Qk0 and Qk0#.</td>
</tr>
<tr>
<td>QVLQ</td>
<td>Output</td>
<td>Output data valid: The QVLQ pin indicates valid output data. QVLQ is edge-aligned with Qk and Qk#.</td>
</tr>
<tr>
<td>TDO</td>
<td>Output</td>
<td>Output data valid: The TDO pin indicates valid output data. QVLQ is edge-aligned with Qk and Qk#.</td>
</tr>
<tr>
<td>VDD</td>
<td>Supply</td>
<td>Power supply: Normally, 1.8V. See Table 8 on page 20 for range.</td>
</tr>
<tr>
<td>VDDQ</td>
<td>Supply</td>
<td>GG power supply: Normally, 1.5V or 1.8V. Isolated on the device for improved noise immunity. See Table 8 on page 20 for range.</td>
</tr>
<tr>
<td>VREF</td>
<td>Supply</td>
<td>Power supply: Normally, 2.5V. See Table 8 on page 20 for range.</td>
</tr>
<tr>
<td>VREF#</td>
<td>Supply</td>
<td>Input reference voltage: Normally VDDQ/2. Provides a reference voltage for the input buffers.</td>
</tr>
<tr>
<td>VSS</td>
<td>Ground</td>
<td>Supply: Ground.</td>
</tr>
<tr>
<td>VSSG</td>
<td>Supply</td>
<td>VDD ground: Isolated on the device for improved noise immunity.</td>
</tr>
<tr>
<td>VTR</td>
<td>Supply</td>
<td>Power supply: Isolated termination supply. Normally, VDDQ/2. See Table 8 on page 20 for range.</td>
</tr>
<tr>
<td>A21</td>
<td>–</td>
<td>Reserved for future use: This signal is internally connected and can be treated as an address input.</td>
</tr>
<tr>
<td>A22</td>
<td>–</td>
<td>Reserved for future use: This signal is not connected and can be connected to ground.</td>
</tr>
</tbody>
</table>

Target max 1200 pins
⇒8 ports, 8 RLDARAM2
Total pins: 1088

![RLDRAM Diagram](image-url)
Example Reality Check

• CPU interface
• Test pins
• Connection to rest of system
• Minimum data unit effect
  – must burst to get perf
    • On chip cache
    – buffer must be spread across all devices
• Probably real number is 4-6 ports
Address Lookup – Bank Interleave

Clock

Bank

Address

Data
DRAM

• Page Mode
• Banking for performance
• 10’s of Mbytes to Gbytes
• Embedded (eDRAM) and external
• Think “packet buffer”
• Denser than SRAM
• Slower than SRAM
• Higher leakage
• Higher soft error rate (SER)
• Usually very limited configurations for eDRAM
TCAM

• Ternary Content Addressable Memory
  – put in data, get address
• Ternary because can match ‘0’, ’1’, or ‘X’
• Great for partial match
  – Longest prefix
  – Access lists
• 6X more power than SRAM
• 7X more area than SRAM
• 4X higher latency than SRAM
• 2X slower
TCAM

• Ternary Content Addressable Memory
  – put in data, get address
• Ternary because can match ‘0’, ‘1’, or ‘X’
• Great for partial match
  – Longest prefix
  – Access lists
• 6X more power than SRAM
• 7X more area than SRAM
• 4X higher latency than SRAM
• 2X slower
TCAM

- Ternary Content Addressable Memory
  - put in data, get address
- Ternary because can match ‘0’, ‘1’, or ‘X’
- Great for partial match
  - Longest prefix
  - Access lists
- 6X more power than SRAM
- 7X more area than SRAM
- 4X higher latency than SRAM
- 2X slower
TCAM

- Ternary Content Addressable Memory
  - put in data, get address
- Ternary because can match ‘0’, ’1’, or ‘X’
- Great for partial match
  - Longest prefix
  - Access lists
- 6X more power than SRAM
- 7X more area than SRAM
- 4X higher latency than SRAM
- 2X slower
TCAM

- Ternary Content Addressable Memory
  - put in data, get address
- Ternary because can match ‘0’, ’1’, or ‘X’
- Great for partial match
  - Longest prefix
  - Access lists
- 6X more power than SRAM
- 7X more area than SRAM
- 4X higher latency than SRAM
- 2X slower
Typical Forwarding Diagram

IP Address → TCAM → Address → SRAM → Fwding Info
Relative Size – 40nm

- DRAM Bit: 0.31um^2, 3,319.335/mm^2
- SRAM Bit: 0.67um^2, 1,500,00/mm^2
- NAND Gate: 1.14um^2, 875,000/mm^2
- TCAM bit: 4.44um^2, 225,000/mm^2
- Flip-Flop: 9.14um^2, 110,000/mm^2
- Human Hair: 7854um^2, 127/mm^2
Other Important Considerations

• Power = $CV^2f$ (Capacitance, Voltage, frequency)
  – often ultimate limiter of what we can do
  – 80 watts per ASIC is very high
  – 40 watts is comfortable
  – What router can handle depends greatly on cooling system
  – 15kW to 30kW for a rack

• Area/Routing
  – Can run out of space
  – think about physical implications
    • ex: cross-bars, 1000-bit busses to lots of places, resources used by lots other blocks
    • 60% utilization
Rules of Thumb

- DRAM latency – 20ns – 60ns external, 3-5ns internal
- SRAM latency – 5-6ns external, 1-2ns internal
- DRAM on chip – 10MB – 100MB
- SRAM on chip – 4KB is small, 1MB is large
  - we have devices that are ~25MB total
- Logic speed – at 1GHz, 40nm – 12-20 levels of logic
- Next generation (28nm) is smaller, but not much faster
- Maximum pins (incl pwr and gnd) – 1400
  - sweet spot around 700
- For slow speed signals (<1GHz) 1 power or ground per 3-4 signal pins
- For high speed signals (>1GHz) 1 power or ground per signal
- For very high speed signals (>4GHz) 2 power or ground per signal
- Flip-Flip is 10 gates
- State machine – small = 500 gates, large = 1000 gates, very large = 5000 gates
- All your final size estimates is 10% too low
- CPU updates take zero bandwidth
- Stay below 17X17mm die. Max is 21X21
- It takes 2 years to design and build an ASIC (start-up can do it in 15 months)
  - 6 months is just manufacturing time
What you should know

- Basic logic building blocks and their function
- Basic memory building blocks and their trade-offs
  - FF, SRAM, DRAM
- TCAM
- Timing, Power, Area, Pin Count
- Rules of Thumb
End