Combinational Logic Design Process

**Step 1: Capture the function**
- Create a truth table or equations, *whichever is most natural for the given problem*, to describe the desired behavior of the combinational logic.

**Step 2: Convert to equations**
- This step is only necessary if you captured the function using a truth table instead of equations. Create an equation for each output by ORing all the minterms for that output. Simplify the equations if desired.

**Step 3: Implement as a gate-based circuit**
- For each output, create a circuit corresponding to the output's equation. (Sharing gates among multiple outputs is OK optionally.)

---

**Digital Logic – Combinational Logic Design Process**

**Example: Three 1s Detector**

**Problem:** Detect three consecutive 1s in an 8-bit input: abcdedfgh

- 00011101 → 1
- 10110101 → 0
- 11110000 → 1

**Step 1: Capture the function**
- Truth table or equation?
  - Truth table too big: 2^8=256 rows

**Step 2: Convert to equation**
- Equation: create terms for each possible case of three consecutive 1s
- \( y = abc + bcd + cde + def + efg + fgh \)

**Step 3: Implement as a gate-based circuit**
Problem: Output in binary on two outputs yz the number of 1s on three inputs

Step 1: Capture the function
- Truth table or equation?
  - Truth table is straightforward

Step 2: Convert to equation
- y = a'bc + ab'c + abc' + abc
- z = a'b'c + a'bc' + ab'c' + abc

Step 3: Implement as a gate-based circuit

Digital Logic – Combinational Logic
Completeness of NAND
- Any Boolean function can be implemented using just NAND gates. Why?
  - Need AND, OR, and NOT
  - NOT: 1-input NAND (or 2-input NAND with inputs tied together)
  - AND: NAND followed by NOT
  - OR: NAND preceded by NOTs
- Likewise for NOR

Digital Logic – Combinational Logic
Number of Possible Boolean Functions
- How many possible functions of 2 variables?
  - 2^2 = 4 rows in truth table, 2 choices for each
  - 2^2 = 2^4 = 16 possible functions
- 2^2 = 2^4 possible functions
**Decoder**
- Popular combinational logic building block, in addition to logic gates
- Converts input binary number to one high output
- 2-input decoder: four possible input binary numbers
  - So has four outputs, one for each possible input binary number
- Internal design
  - AND gate for each output to detect input combination
- Decoder with enable e
  - Outputs all 0 if e=0
  - Regular behavior if e=1
- n-input decoder: $2^n$ outputs

**Multiplexor (Mux)**
- Another popular combinational building block
- Routes one of its N data inputs to its one output, based on binary value of select inputs
  - 4 input mux → needs 2 select inputs to indicate which input to route through
  - 8 input mux → 3 select inputs
  - N inputs → $\log_2(N)$ selects
- Like a railyard switch

**N-bit Mux**
- What output of a 3x8 decoder will be asserted if $i_2i_1i_0 = 110$?
  1. $d_0 = 1$
  2. $d_3 = 1$
  3. $d_6 = 1$
  4. $d_7 = 1$
Digital Logic – Combinational Logic

Mux Example

- City mayor (with no budget for good voting system) can set four switches up or down, representing his/her vote on each of four proposals, numbered 0, 1, 2, 3
- City manager can display any such vote on large green/red LED (light) by setting two switches to represent binary 0, 1, 2, or 3
- Use 4x1 mux

D&D 

Digital Design Copyright © 2006
Frank Vahid

Digital Logic – Combinational Logic

N-bit Mux

- Example: Two 4-bit inputs, A (a3 a2 a1 a0), and B (b3 b2 b1 b0)
  - 4-bit 2x1 mux (just four 2x1 muxes sharing a select line) can select between A or B

Digital Logic – Combinational Logic

N-bit Mux Example

- If A=5, B=2, what is the output of the 4-bit 2x1 mux if s0 = 1?
  1. 0
  2. 5
  3. 2
  4. 7

Digital Logic – Combinational Logic

N-bit Mux Example

- Four possible display items
  - Temperature (T), Average miles-per-gallon (A), Instantaneous mpg (I), and Miles remaining (M) – each is 8-bits wide
  - Choose which to display using two inputs x and y
  - Use 8-bit 4x1 mux

Digital Logic – Combinational Logic

N-bit Mux Example

- From the car’s central computer
- We’ll design this later
- To the above display

Digital Logic – Combinational Logic

N-bit Mux Example

- Four possible display items
  - Temperature (T), Average miles-per-gallon (A), Instantaneous mpg (I), and Miles remaining (M) – each is 8-bits wide
  - Choose which to display using two inputs x and y
  - Use 8-bit 4x1 mux
Real gates have some delay
- Outputs don’t change immediately after inputs change

### Encoders
- Encoder: Combinational logic building block with opposite functionality of decoder
  - Outputs binary encoding for input signal that is 1
  - 4x2 encoder would have four inputs and 2 outputs
  - What if two inputs are 1?
    - Can use a priority encoder
    - Gives priority to the highest input that is 1, and outputs binary encoding for that input
    - Example: If $d_3=1$ and $d_1=1$, will output $e_0=1$ and $e_1=1$ because $d_3$ has priority

### In Class Exercise
- Design a 4x2 encoder using AND, OR, and NOT gates.