Lecture #8: More FSM Design

Paul Hartke
Phartke@stanford.edu
Stanford EE121
January 31, 2002

Administrivia

- Midterm #1 is next Tuesday (February 5th) in class.
  - Will not include state machines.
- Lab 3 Design Post-Mortem
  - ¾ of you have had lab section
  - Comments/Issues?
- Lab 4 handout
  - Due next week as normal.
- HW3 handout
  - Due Next Thursday February 7th
  - Discussion later….

State Machine Design Process

1. Determination of inputs and outputs.
2. Determination of machine states.
3. Create State/Bubble Diagram—should this be a Mealy or Moore machine?
4. State Assignment—assign each state a particular value.
5. Create Transition/Output Table
6. Derive Next State Logic for each state element—using K-maps as necessary.
7. Derive Output logic.
8. Implement in Xilinx.

State Assignment

- State Assignment is a factor in designs ultimate size, speed, and power.
  - Good ones are heuristic based.
  - Use binary encoding to minimize Flip-Flops.
  - Use (modified) one-hot for minimal next state logic depth.
    - As we saw in previous example, not always a "big" win.
    - But more algorithmic for the advantages…

What about unassigned states?

- Consider "1's counter" that actually kept track of the previous number of bits is a multiple of 3.

		XY				
Meaning	**S**	**00**	**01**	**11**	**10**	**Z**
Got zero 1s (modulo 4)	S0	S0	S1	S2	S1	1
Got one 1 (modulo 4)	S1	S1	S2	S3	S2	0
Got two 1s (modulo 4)	S2	S2	S3	S0	S3	0
Got three 1s (modulo 4)	S3	S3	S0	S1	S0	0
	No Longer in Table!		S+			

States could be "Don't Cares"

- What is the advantage of Don't Cares?
- Any Downsides?

		XY				
Meaning	**S**	**00**	**01**	**11**	**10**	**Z**
Got zero 1s (modulo 4)	S0	S0	S1	S2	S1	1
Got one 1 (modulo 4)	S1	S1	S2	S3	S2	0
Got two 1s (modulo 4)	S2	S2	S3	S0	S3	0
Got three 1s (modulo 4)	S3	d	d	d	d	d
			S+			

"Safe" Method

- What if the FSM got into an invalid state.
  - Random bit error (from Gamma Radiation?)
- State machine will always get back in a known good state

		XY				
Meaning	S	00	01	11	10	Z
Got zero 1s (modulo 4)	S0	S0	S1	S2	S1	1
Got one 1 (modulo 4)	S1	S1	S2	S3	S2	0
Got two 1s (modulo 4)	S2	S2	S3	S0	S3	0
Got three 1s (modulo 4)	S3	S0	S0	S0	S0	0
		S*				

Input Invalid Values

- Same issue with some input value combinations.
  - Maybe you know something about the external circuit
    - one hot input on two signals

		XY				
Meaning	S	00	01	11	10	Z
Got zero 1s (modulo 4)	S0	S0	S1	S2	S1	1
Got one 1 (modulo 4)	S1	S1	S2	S3	S2	0
Got two 1s (modulo 4)	S2	S2	S3	S0	S3	0
Got three 1s (modulo 4)	S3	S3	S0	S1	S0	0
		S*		Invalid input		

One-Hot Encoding

- Using the "safe" method is painful for state assignment but no different for invalid input combos. Why?

Q1Q2Q3Q4	XY=00	XY=01	XY=11	XY=10	Z
0001	0001	0010	0100	0010	1
0010	0010	0100	1000	0100	0
0100	0100	1000	0001	1000	0
1000	1000	0001	0010	0001	0

Differences???

- Actually ***very*** big one.
  - Ignore bit errors for now…
- State Assignment is totally internal to the circuit.
  - Internal state
- Inputs are visible to all.
  - Interface
- What would OOP tell us about this?

Interface Rules

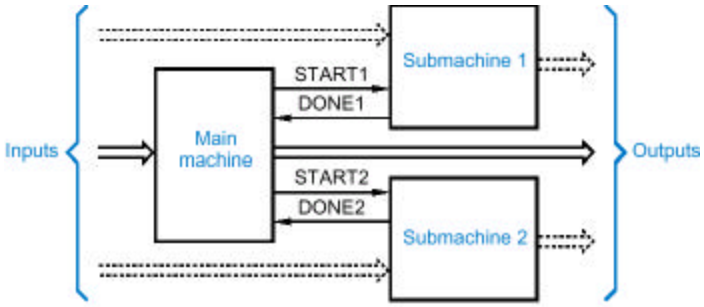
- "Be forgiving on the inputs and conservative on the outputs"
  - Some smart CS person.
- Makes your blocks easier to interface to.
  - Getting individual blocks to work is non-trivial.
  - Getting them to work together during integration is the even harder.
- Allows potential for reuse
  - Basis of "Intellectual Property" cores.

Decomposing State Machines

- Any digital system, no matter how big, can be implemented as a single state machine.
  - inputs, state vars, and outputs
  - State vars are too big because there are a lot of states that are independent of each other.
- Hierarchical design to the rescue again!!!

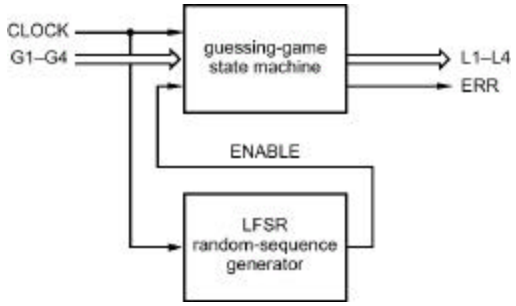
Hierarchical State Machine Design

- What is *Start*?
  - Another application for synchronous reset and enable
- *Done* is design dependent.



Counters as sub-FSMs

- Allows main state machine to wait some number of clocks between transitioning between states…
  - LFSR is a type of counter.
  - Do ***NOT*** do this on Lab 4



HW #3 Questions

- Show Timing Analyzer for other speed grade. Why is this not complete?

Midterm #1

- Boolean Algebra Axioms
- K-maps and Hazards
- Static CMOS gates
- Ground Bounce
- CV^2f Power
- Something about a combinational logic circuit idiom
- FPGA Architecture.

Midterm #1(cont)

- Open Book, Open Notes
- Not designed to trick.
- Based as much as possible on interview questions.