Problem sets

Contents

Getting Started

- Getting Started ✓ (/wiki/step_one)
- Output Zero ✓ (/wiki/zero)

Verilog Language

Basics

- Simple wire ✓ (/wiki/wire)
- Four wires ✓ (/wiki/wire4)
- Inverter ✓ (/wiki/notgate)
- AND gate ✓ (/wiki/andgate)
- NOR gate ✓ (/wiki/norgate)
- XNOR gate ✓ (/wiki/xnorgate)
- Declaring wires ✓ (/wiki/wire_decl)
- 7458 chip ✓ (/wiki/7458)

Vectors

- Vectors✓ (/wiki/vector0)
- Vectors in more detail✓ (/wiki/vector1)
- Vector part select✓ (/wiki/vector2)
- Bitwise operators✓ (/wiki/vectorgates)
- Four-input gates✓ (/wiki/gates4)
- Vector concatenation operator✓ (/wiki/vector3)
- Vector reversal 1✓ (/wiki/vectorr)
- Replication operator✓ (/wiki/vector4)
- More replication✓ (/wiki/vector5)

Modules: Hierarchy

- Modules✓ (/wiki/module)
- Connecting ports by position✓ (/wiki/module_pos)
- Connecting ports by name✓ (/wiki/module_name)
- Three modules✓ (/wiki/module_shift)
- Modules and vectors✓ (/wiki/module_shift8)
- Adder 1✓ (/wiki/module_add)
- Adder 2⊖ (/wiki/module_fadd)
- Carry-select adder○ (/wiki/module_cseladd)
- Adder-subtractor○ (/wiki/module_addsub)

Procedures

Procedures include **always**, initial, task, and function blocks. Procedures allow sequential statements (which cannot be used outside of a procedure) to be used to describe the behaviour of a circuit.

- Always blocks (combinational)○ (/wiki/alwaysblock1)
- Always blocks (clocked)○ (/wiki/alwaysblock2)
- If statement○ (/wiki/always_if)
- If statement latches○ (/wiki/always_if2)
- Case statement○ (/wiki/always_case)
- Priority encoder○ (/wiki/always_case2)
- Priority encoder with casez○ (/wiki/always_casez)
- Avoiding latches○ (/wiki/always_nolatches)

More Verilog Features

- Conditional ternary operator○ (/wiki/conditional)
- Reduction operators○ (/wiki/reduction)
- Reduction: Even wider gates✓ (/wiki/gates100)
- Combinational for-loop: Vector reversal 2○ (/wiki/vector100r)
- Combinational for-loop: 255-bit population count○ (/wiki/popcount255)
- Generate for-loop: 100-bit binary adder 2○ (/wiki/adder100i)
- Generate for-loop: 100-digit BCD adder○ (/wiki/bcdadd100)

Circuits

Combinational Logic

Basic Gates

- Wire✔ (/wiki/exams/m2014_q4h)
- GND○ (/wiki/exams/m2014_q4i)
- NOR○ (/wiki/exams/m2014_q4e)
- Another gate○ (/wiki/exams/m2014_q4f)
- Two gates○ (/wiki/exams/m2014_q4g)
- More logic gates○ (/wiki/gates)
- 7420 chip○ (/wiki/7420)
- Truth tables○ (/wiki/truthtable1)
- Two-bit equality○ (/wiki/mt2015_eq2)
- Simple circuit A○ (/wiki/mt2015_q4a)
- Simple circuit B○ (/wiki/mt2015_q4b)
- Combine circuits A and B○ (/wiki/mt2015_q4)
- Ring or vibrate?○ (/wiki/ringer)
- Thermostat○ (/wiki/thermostat)
- 3-bit population count○ (/wiki/popcount3)
- Gates and vectors○ (/wiki/gatesv)
- Even longer vectors○ (/wiki/gatesv100)

Multiplexers

- 2-to-1 multiplexer○ (/wiki/mux2to1)
- 2-to-1 bus multiplexer○ (/wiki/mux2to1v)
- 9-to-1 multiplexer○ (/wiki/mux9to1v)
- 256-to-1 multiplexer○ (/wiki/mux256to1)
- 256-to-1 4-bit multiplexer○ (/wiki/mux256to1v)

Arithmetic Circuits

- Half adder○ (/wiki/hadd)
- Full adder○ (/wiki/fadd)
- 3-bit binary adder○ (/wiki/adder3)
- Adder○ (/wiki/exams/m2014_q4j)
- Signed addition overflow○ (/wiki/exams/ece241_2014_q1c)
- 100-bit binary adder○ (/wiki/adder100)
- 4-digit BCD adder○ (/wiki/bcdadd4)

Karnaugh Map to Circuit

- 3-variable○ (/wiki/kmap1)
- 4-variable○ (/wiki/kmap2)
- 4-variable○ (/wiki/kmap3)
- 4-variable○ (/wiki/kmap4)
- Minimum SOP and POS○ (/wiki/exams/ece241_2013_q2)

- Karnaugh map○ (/wiki/exams/m2014_q3)
- Karnaugh map○ (/wiki/exams/2012_q1g)
- K-map implemented with a multiplexer○ (/wiki/exams/ece241_2014_q3)

Sequential Logic

Latches and Flip-Flops

- D flip-flop○ (/wiki/dff)
- D flip-flops○ (/wiki/dff8)
- DFF with reset○ (/wiki/dff8r)
- DFF with reset value○ (/wiki/dff8p)
- DFF with asynchronous reset○ (/wiki/dff8ar)
- DFF with byte enable○ (/wiki/dff16e)
- D Latch○ (/wiki/exams/m2014_q4a)
- DFF○ (/wiki/exams/m2014_q4b)
- DFF○ (/wiki/exams/m2014_q4c)
- DFF+gate○ (/wiki/exams/m2014_q4d)
- Mux and DFF○ (/wiki/mt2015_muxdff)
- Mux and DFF○ (/wiki/exams/2014_q4a)
- DFFs and gates○ (/wiki/exams/ece241_2014_q4)
- Create circuit from truth table○ (/wiki/exams/ece241_2013_q7)
- Detect an edge⛔ (/wiki/edgedetect)
- Detect both edges○ (/wiki/edgedetect2)
- Edge capture register○ (/wiki/edgecapture)
- Dual-edge triggered flip-flop○ (/wiki/dualedge)

Counters

- Four-bit binary counter○ (/wiki/count15)
- Decade counter○ (/wiki/count10)
- Decade counter again○ (/wiki/count1to10)
- Slow decade counter○ (/wiki/countslow)
- Counter 1-12○ (/wiki/exams/ece241_2014_q7a)
- Counter 1000○ (/wiki/exams/ece241_2014_q7b)
- 4-digit decimal counter○ (/wiki/countbcd)
- 12-hour clock○ (/wiki/count_clock)

Shift Registers

- 4-bit shift register○ (/wiki/shift4)
- Left/right rotator○ (/wiki/rotate100)
- Left/right arithmetic shift by 1 or 8○ (/wiki/shift18)
- 5-bit LFSR○ (/wiki/lfsr5)
- 3-bit LFSR○ (/wiki/mt2015_lfsr)
- 32-bit LFSR○ (/wiki/lfsr32)
- Shift register○ (/wiki/exams/m2014_q4k)
- Shift register○ (/wiki/exams/2014_q4b)

- 3-input LUT○ (/wiki/exams/ece241_2013_q12)

More Circuits

- Cellular automata
  - Rule 90○ (/wiki/rule90)
  - Rule 110○ (/wiki/rule110)
  - Conway's Game of Life 16x16○ (/wiki/conwaylife)

Finite State Machines

- Simple FSM 1 (asynchronous reset)○ (/wiki/fsm1)
- Simple FSM 1 (synchronous reset)○ (/wiki/fsm1s)
- Simple FSM 2 (asynchronous reset)○ (/wiki/fsm2)
- Simple FSM 2 (synchronous reset)○ (/wiki/fsm2s)
- Simple state transitions 3○ (/wiki/fsm3comb)
- Simple one-hot state transitions 3○ (/wiki/fsm3onehot)
- Simple FSM 3 (asynchronous reset)○ (/wiki/fsm3)
- Simple FSM 3 (synchronous reset)○ (/wiki/fsm3s)
- Design a Moore FSM○ (/wiki/exams/ece241_2013_q4)
- Lemmings 1○ (/wiki/lemmings1)
- Lemmings 2○ (/wiki/lemmings2)
- Lemmings 3○ (/wiki/lemmings3)
- Lemmings 4○ (/wiki/lemmings4)
- One-hot FSM○ (/wiki/fsm_onehot)
- PS/2 packet parser○ (/wiki/fsm_ps2)
- PS/2 packet parser and datapath○ (/wiki/fsm_ps2data)
- Serial receiver○ (/wiki/fsm_serial)
- Serial receiver and datapath○ (/wiki/fsm_serialdata)
- Serial receiver with parity checking○ (/wiki/fsm_serialdp)
- Sequence recognition○ (/wiki/fsm_hdlc)

- Q8: Design a Mealy FSM○ (/wiki/exams/ece241_2013_q8)
- Q5a: Serial two's complementer (Moore FSM)○ (/wiki/exams/ece241_2014_q5a)
- Q5b: Serial two's complementer (Mealy FSM)○ (/wiki/exams/ece241_2014_q5b)
- Q3a: FSM○ (/wiki/exams/2014_q3fsm)
- Q3b: FSM○ (/wiki/exams/2014_q3bfsm)
- Q3c: FSM logic○ (/wiki/exams/2014_q3c)
- Q6b: FSM next-state logic○ (/wiki/exams/m2014_q6b)
- Q6c: FSM one-hot next-state logic○ (/wiki/exams/m2014_q6c)
- Q6: FSM○ (/wiki/exams/m2014_q6)
- Q2a: FSM○ (/wiki/exams/2012_q2fsm)
- Q2b: One-hot FSM equations○ (/wiki/exams/2012_q2b)
- Q2a: FSM○ (/wiki/exams/2013_q2afsm)
- Q2b: Another FSM○ (/wiki/exams/2013_q2bfsm)

Building Larger Circuits

- Counter with period 1000○ (/wiki/exams/review2015_count1k)
- 4-bit shift register and down counter○ (/wiki/exams/review2015_shiftcount)
- FSM: Sequence 1101 recognizer○ (/wiki/exams/review2015_fsmseq)
- FSM: Enable shift register○ (/wiki/exams/review2015_fsmshift)
- FSM: The complete FSM○ (/wiki/exams/review2015_fsm)
- The complete timer○ (/wiki/exams/review2015_fancytimer)
- FSM: One-hot logic equations○ (/wiki/exams/review2015_fsmonehot)

Verification: Reading Simulations

Finding bugs in code

- Mux○ (/wiki/bugs_mux2)
- NAND○ (/wiki/bugs_nand3)
- Mux○ (/wiki/bugs_mux4)
- Add/sub○ (/wiki/bugs_addsubz)
- Case statement○ (/wiki/bugs_case)

Build a circuit from a simulation waveform

- Combinational circuit 1○ (/wiki/sim/circuit1)
- Combinational circuit 2○ (/wiki/sim/circuit2)
- Combinational circuit 3○ (/wiki/sim/circuit3)
- Combinational circuit 4○ (/wiki/sim/circuit4)
- Combinational circuit 5○ (/wiki/sim/circuit5)
- Combinational circuit 6○ (/wiki/sim/circuit6)
- Sequential circuit 7○ (/wiki/sim/circuit7)
- Sequential circuit 8○ (/wiki/sim/circuit8)
- Sequential circuit 9○ (/wiki/sim/circuit9)
- Sequential circuit 10○ (/wiki/sim/circuit10)

Verification: Writing Testbenches

- Clock○ (/wiki/tb/clock)
- Testbench1○ (/wiki/tb/tb1)
- AND gate○ (/wiki/tb/and)
- Testbench2○ (/wiki/tb/tb2)
- T flip-flop○ (/wiki/tb/tff)

Retrieved from "http://hdlbits.01xz.net/mw/index.php?title=Problem_sets&oldid=1795 (http://hdlbits.01xz.net/mw/index.php?title=Problem_sets&oldid=1795)"