

Lab 3. Finite State Machine

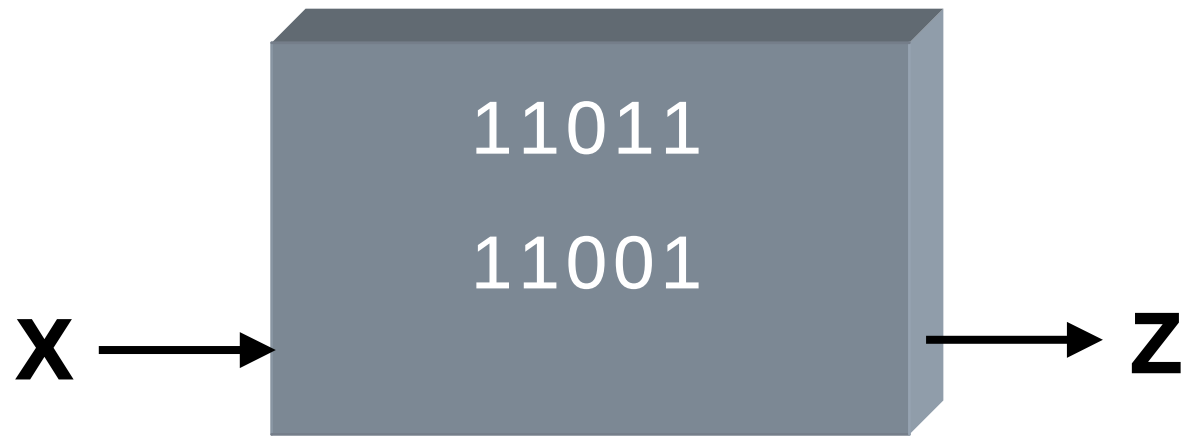
CSE 140L

May 4, 2005

Agenda

- ☐ Lab assignment
- ☐ Design flow
- ☐ Tutorial on a simple design
- ☐ Advanced topics

Pattern Recognizer



X 11101110011011

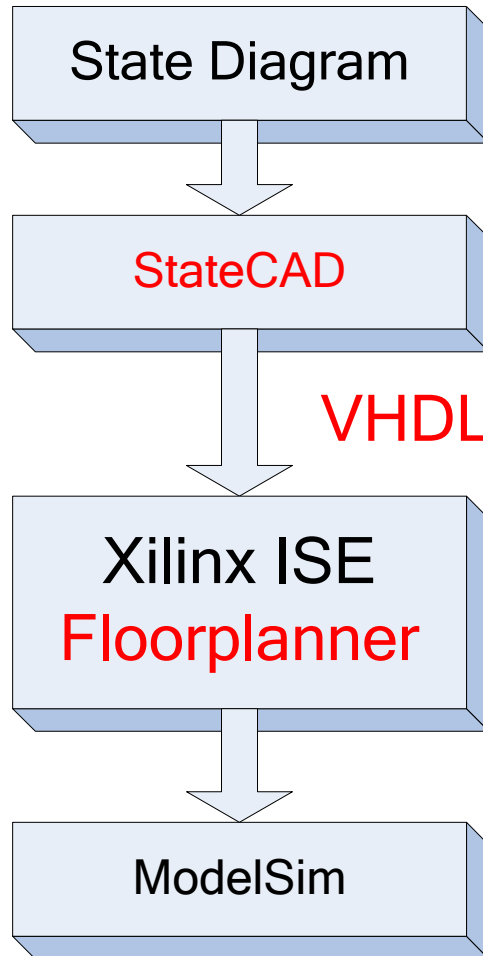
Z 00000100010001

Overlaps allowed

Design Space

- Finite state machine
- Mealy and Moore machines
 - Different state diagrams
 - Mealy outputs at transitions; Moore outputs at states
- State encodings
 - 3 different encodings for each machine
- Comparison on #states, #FFs, #logic blocks, #line of codes.

Design Flow



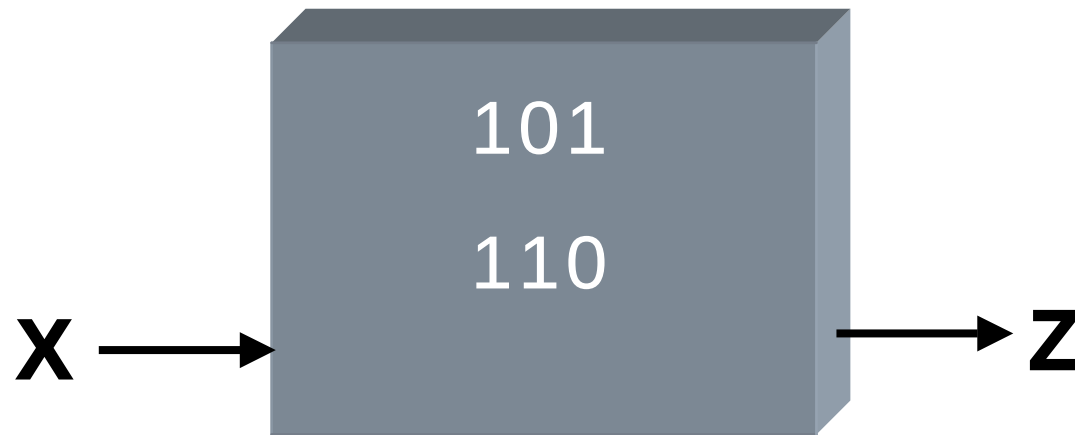
State diagram input

Interfacing

Layout of the design

Tutorial

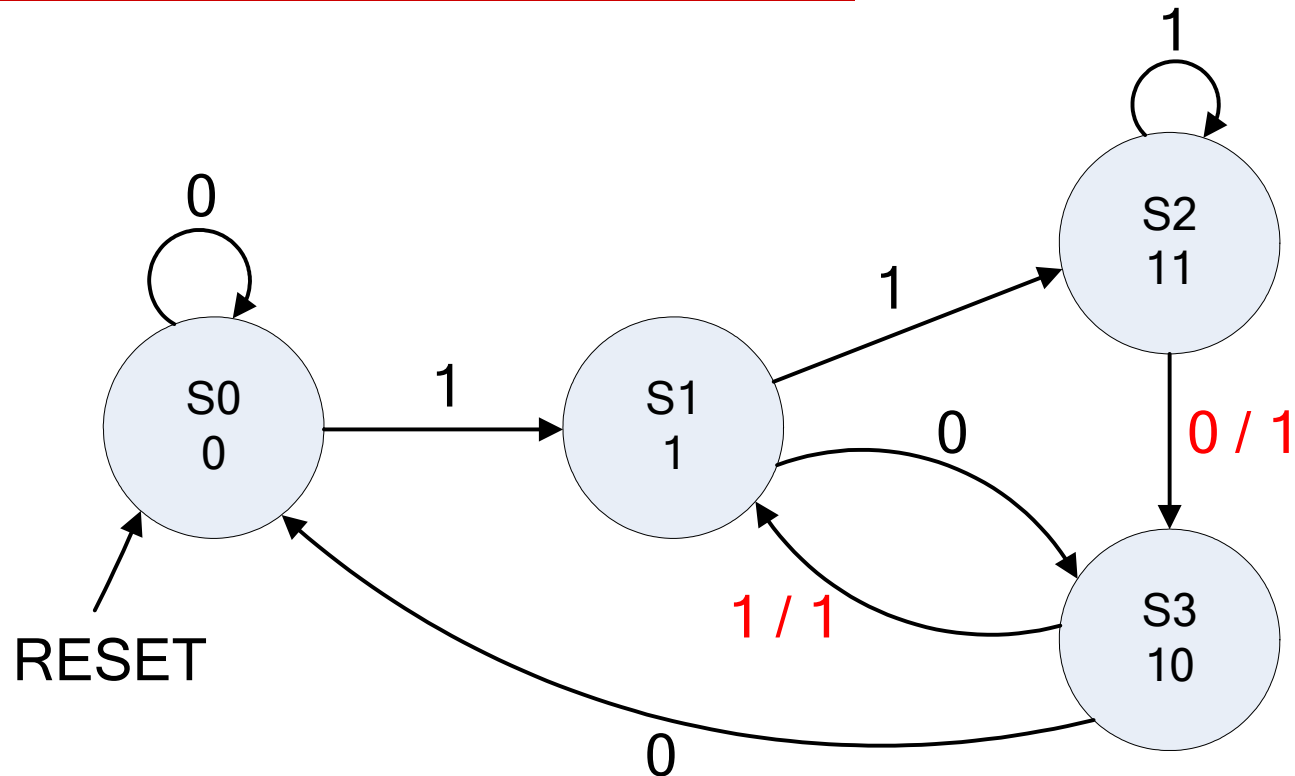
- The pattern recognizer for tutorial
 - Recognize 110 and 101



X 01101001010

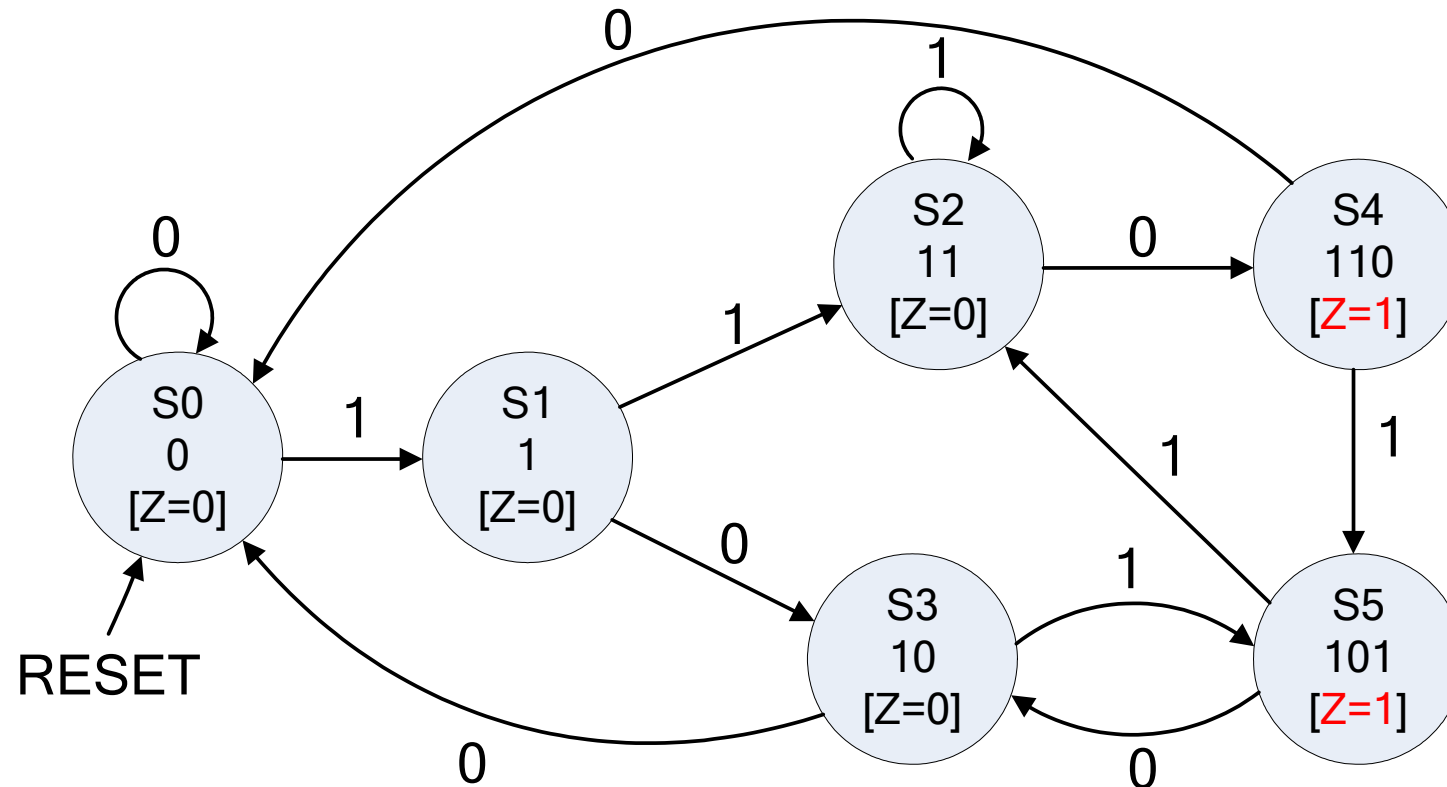
Z 00011000010

State Diagram (Mealy)



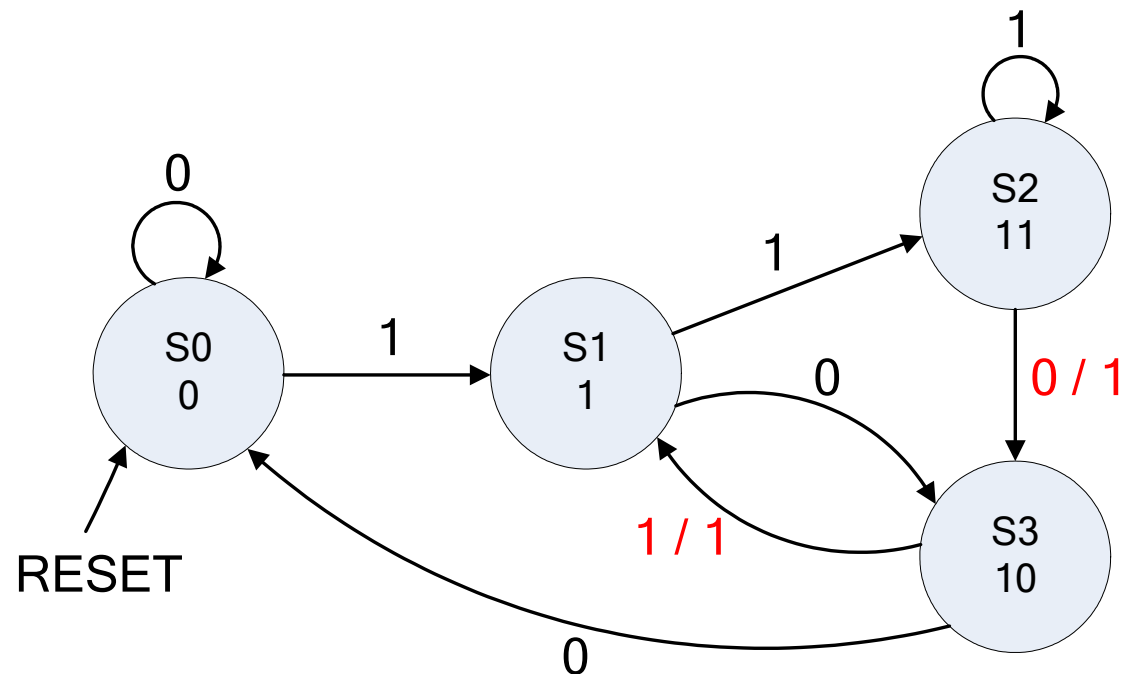
- ❑ 4 states, 9 transitions, 2 outputs at transitions
- ❑ Each node has two outgoing edges

State Diagram (Moore)



□ 6 states, 13 transitions, outputs at states

State Encodings



	S0	S1	S2	S3
Code1 (Q_1Q_0)	00	01	10	11
Code2 (Q_1Q_0)	11	10	01	00

Tutorial Design

- ☐ Input Mealy state diagram in stateCAD
- ☐ Generate VHDL codes
- ☐ Implement VHDL in Xilinx
 - Synthesis
 - Mapping
 - Place and route
 - Layout checking
- ☐ Simulate in ModelSim

Manual State Assignment

- ❑ Modify the VHDL program with state assignments.

ARCHITECTURE BEHAVIOR OF MEALY IS

TYPE type_sreg IS (s0,s1,s2,s3);

ATTRIBUTE enum_encoding : string;

ATTRIBUTE enum_encoding of type_sreg :
type is "00 01 10 11";

SIGNAL sreg, next_sreg : type_sreg;

BEGIN

PROCESS (CLK, next_sreg)

BEGIN