CS221: Digital Design

ASM/ FSMD/ RTL Design

A. Sahu

Dept of Comp. Sc. & Engg.

Indian Institute of Technology Guwahati

Outline

- Drawbacks of state diagrams for real systems
- FSMD/ASM
- ASM Specification
- Comparison of FSM Vs ASM
  - Conversion of FSM to ASM, vice versa
- RTL Design

Reference Material for Lec 33, 34, 35

- Chapter 8 of Mano Book
  - Design at Register Transfer Level
  - Classic Example: Booth Multiplication
- Chapter 15 of Kumar Book
  - Algorithmic State Machine

Algorithmic State Machine

Algorithmic State Machine –

- Another representation of a Finite State Machine
- Suitable for FSMs with
  - a larger number of inputs and outputs
- As compared to FSMs expressed using
  - state diagrams and
  - state tables.

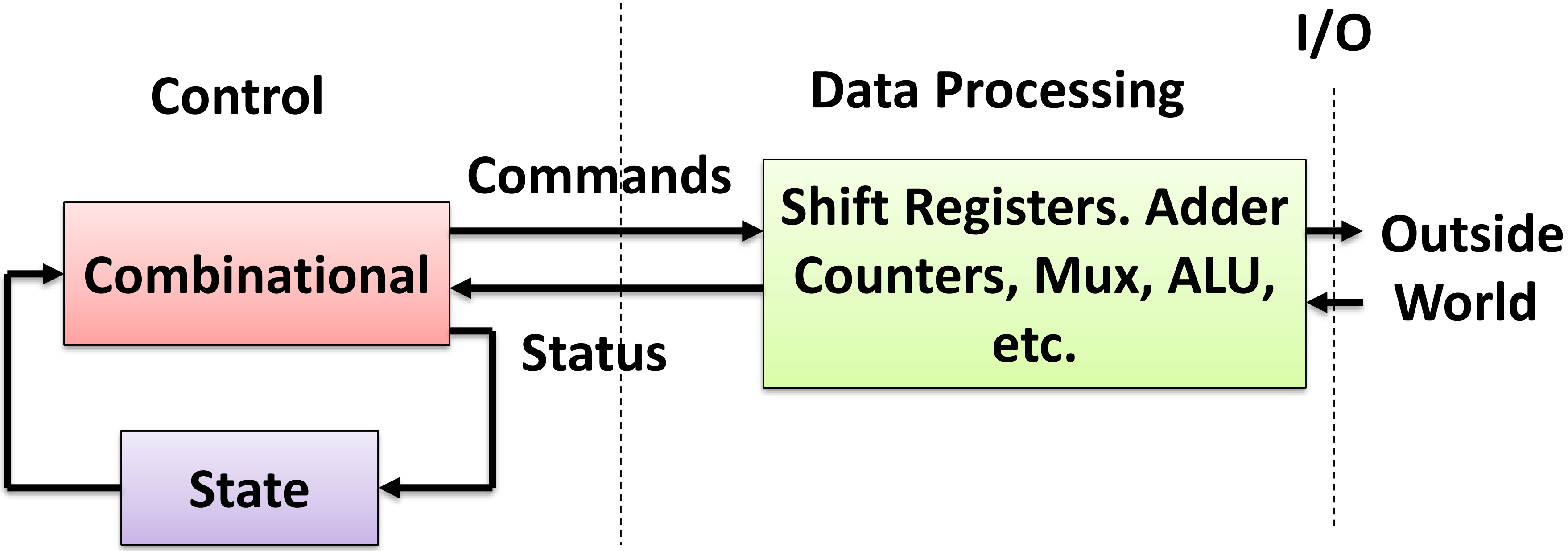
ASM Overview

- We need to separate controller & data processor
  - Controller – What actions need to be taken? What is fundamental operating mode?
  - Processor – Undertake the action. Manipulate the data

The ultimate Goal of this course : Design using Control Path + Data Approach : RTL Design

ASM Overview

- Control and data path interaction

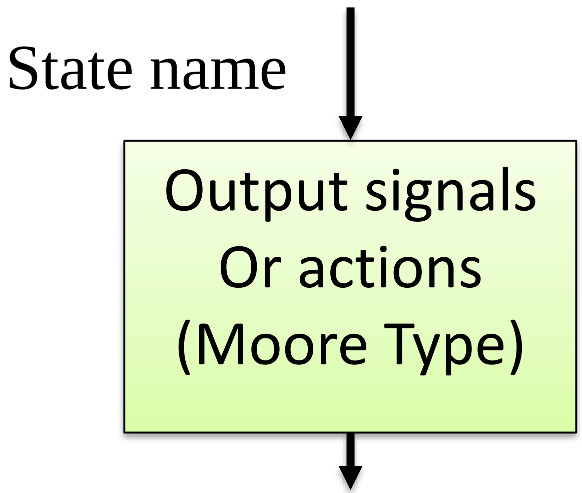


- Our circuit is now explicitly separated

ASM Design : ASM Chart

- ASM charts are like flowcharts, with a few crucial differences.
- Be careful, especially with timing.
- Three type components/Box
  - **State Box**
  - **Decision Box**
  - Combinational Box/Transition Box/**Conditional Box**

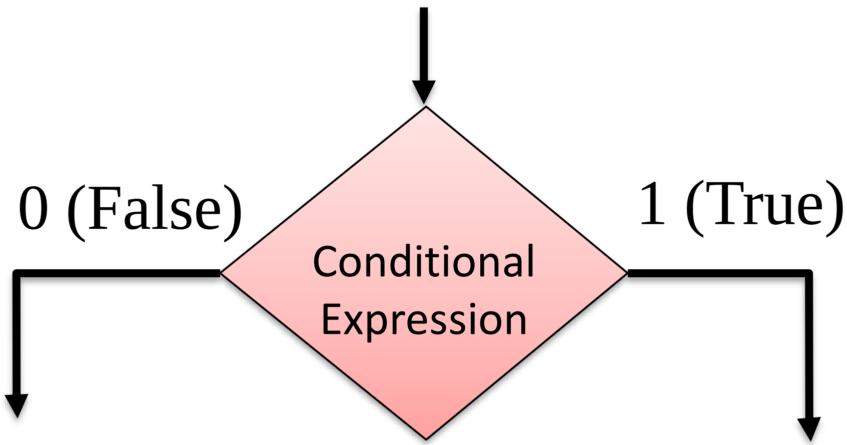
ASM charts : 3 Elements used



State name

Output signals
Or actions
(Moore Type)

(a) State box

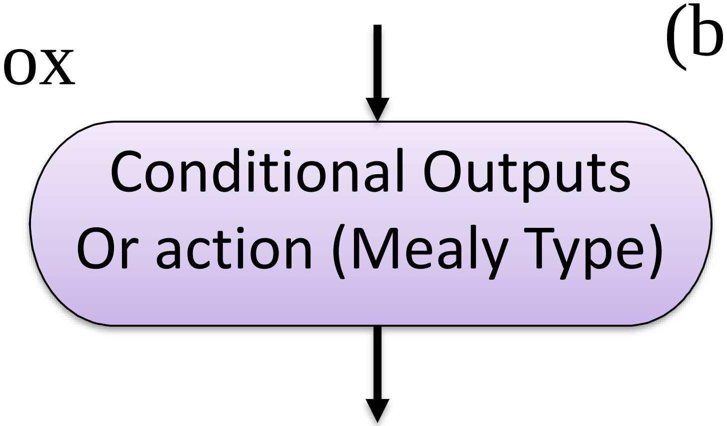


0 (False)

Conditional
Expression

1 (True)

(b) Decision box

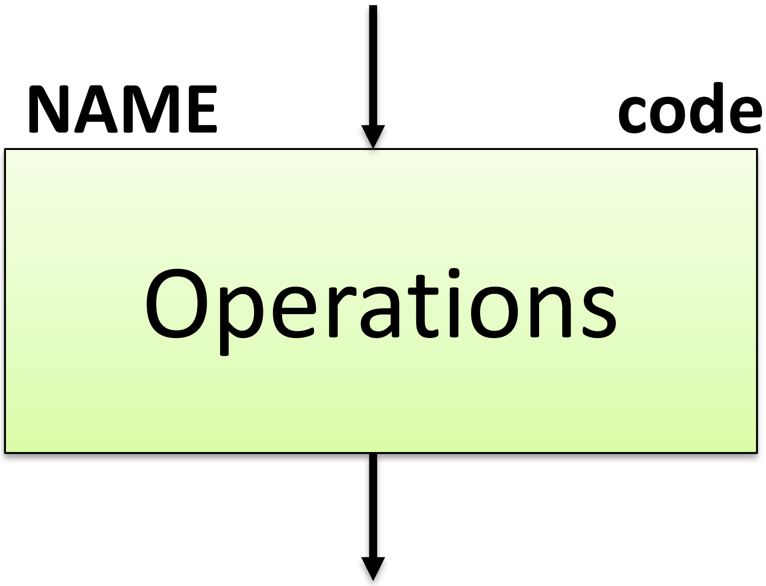


Conditional Outputs
Or action (Mealy Type)

(c) Conditional output box

ASM Design : State Box

- State Box – one box per system state



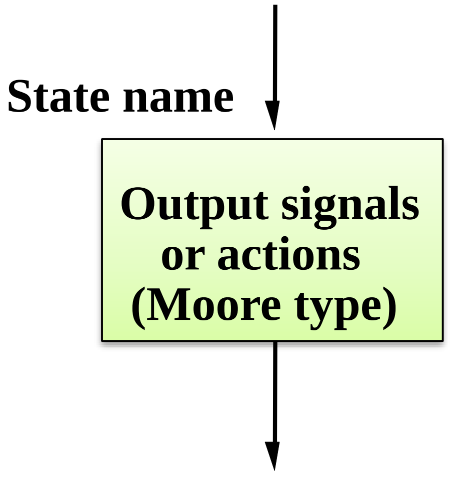
i.e. optional binary state code

ASM Design : State Box

- Operation notation:
  - Sum <- 0 or Carry <- 0 or LOAD A
  - Combinational variable: S=0, T=S+V
- Idea: keep operations abstract & high level.
  - Don't work in detailed language of processing logic (i.e. write Sum <- 0, not $CLR_{Sum\ Reg}=1$)
- Operations will take place at the end of the clock period

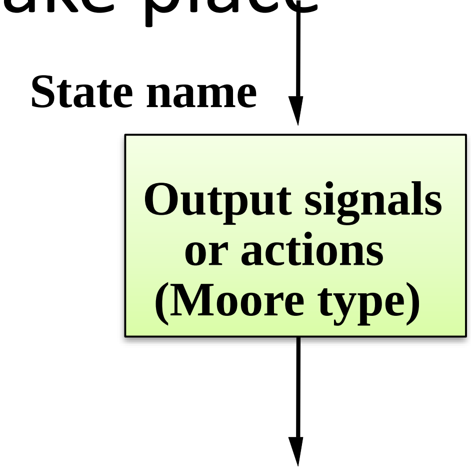
ASM: State Box

- **State box** – represents a state.
- Equivalent to a node in a state diagram or a row in a state table.
- Contains **register transfer** actions or output signals
- **Moore-type outputs are listed inside of the box.**



ASM: State Box

- It is customary to write only the name of the signal that has to be asserted in the given state,
  - e.g., z instead of z<=1.
- Also, it might be useful to write an action to be taken,
  - e.g., count <= count + 1,
- And only later translate it to asserting a control signal that causes a given action to take place
  - (e.g., enable signal of a counter).

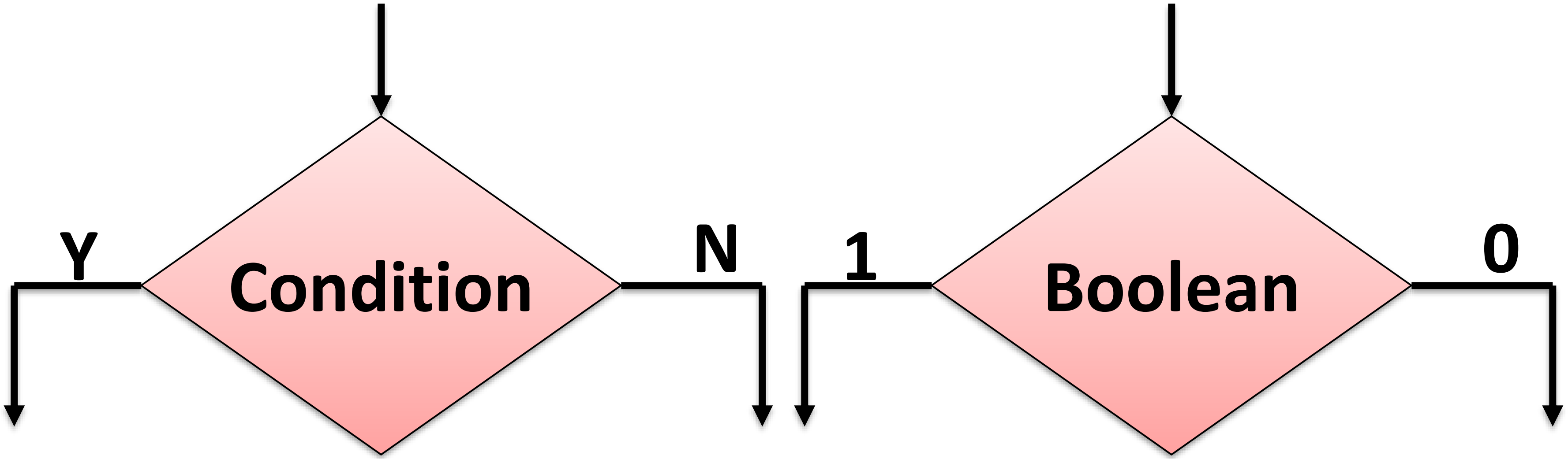


State name

Output signals or actions (Moore type)

ASM Design : Decision Box

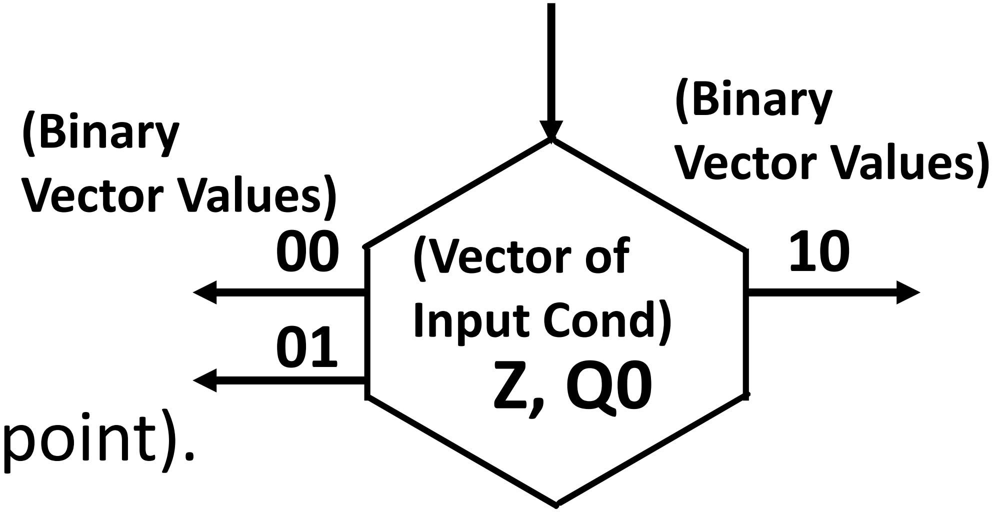
- Decision Box - Basic condition, i.e. logic flow control.
- Only the decision boxes depend on inputs.



ASM Design : Decision Box

- **Decision box** – indicates that
  - a given condition is to be tested and
  - the exit path is to be chosen accordingly
- The condition expression may include
  - One or more inputs to the FSM.

Vector Decision Box



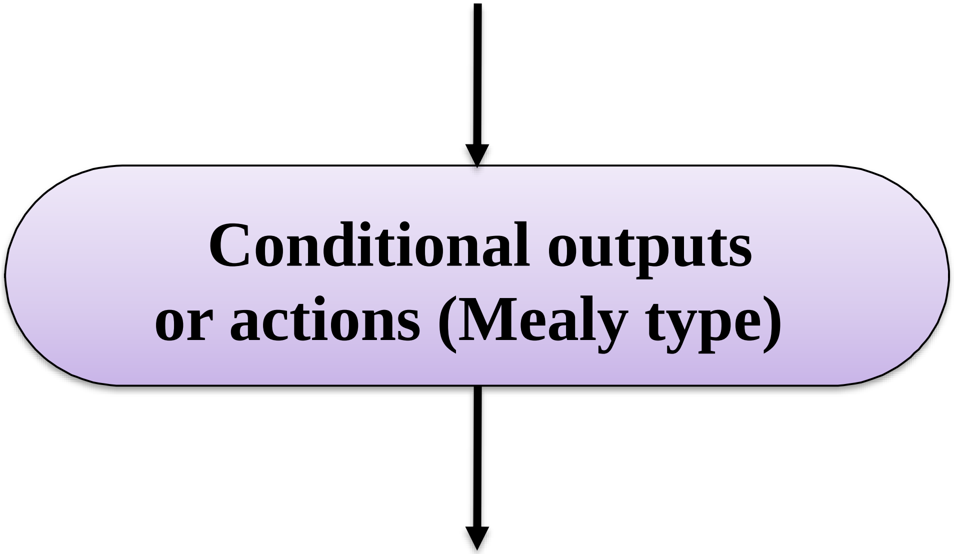
- A hexagon with:
  - One Input Path (entry point).
  - A vector of input conditions, placed in the center of the box, that is tested.
  - Up to 2^n output paths. The path taken has a binary vector value that matches the vector input condition

ASM Design

- Keep conditions as general as possible.
- Prefer: Carry high? Over $Q_{FF\#5}$=1?

ASM Design: Conditional Box

- Conditional Box - An action/operation
  - – to be undertaken conditioned on some earlier decision box.

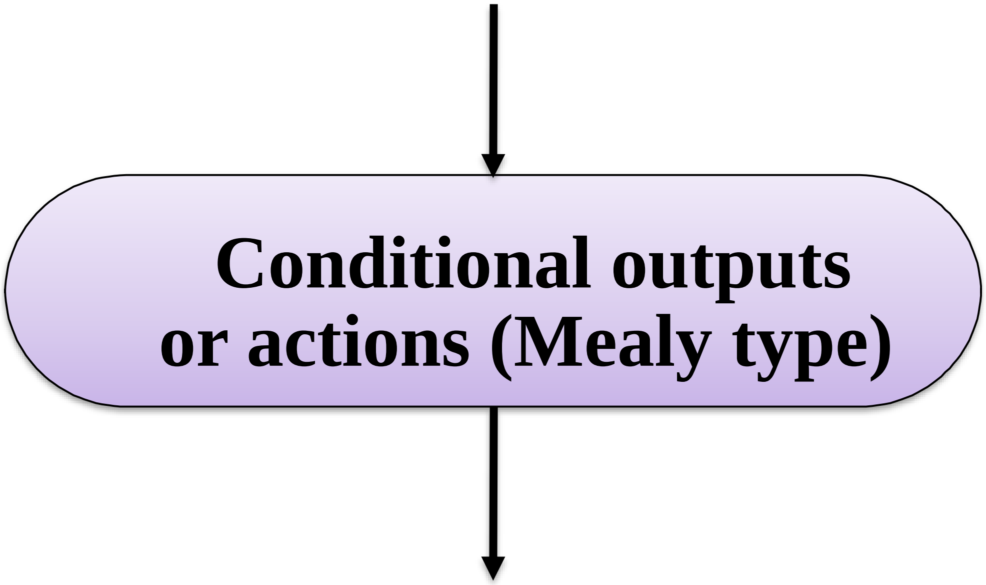


ASM Design Vs Flowchart

- Conditional boxes do not appear in normal flowcharts.
- The essential difference is timing:
  - Flowcharts are sequential
  - ASM charts are not. All of the operations associated with a given state take place simultaneously.

ASM Design: Conditional Output Box

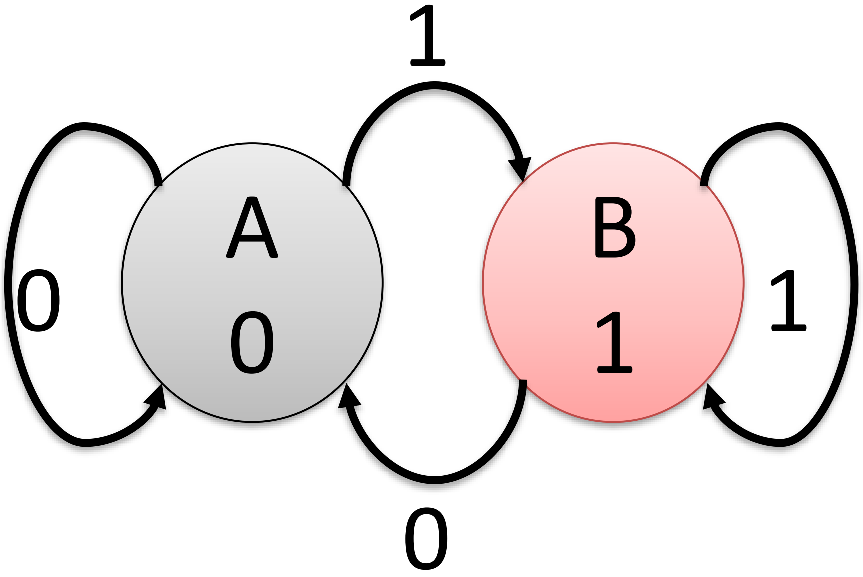
- **Conditional output box**
- Denotes output signals that are of the Mealy type.
- The condition that determines whether such outputs are generated is specified in the decision box.



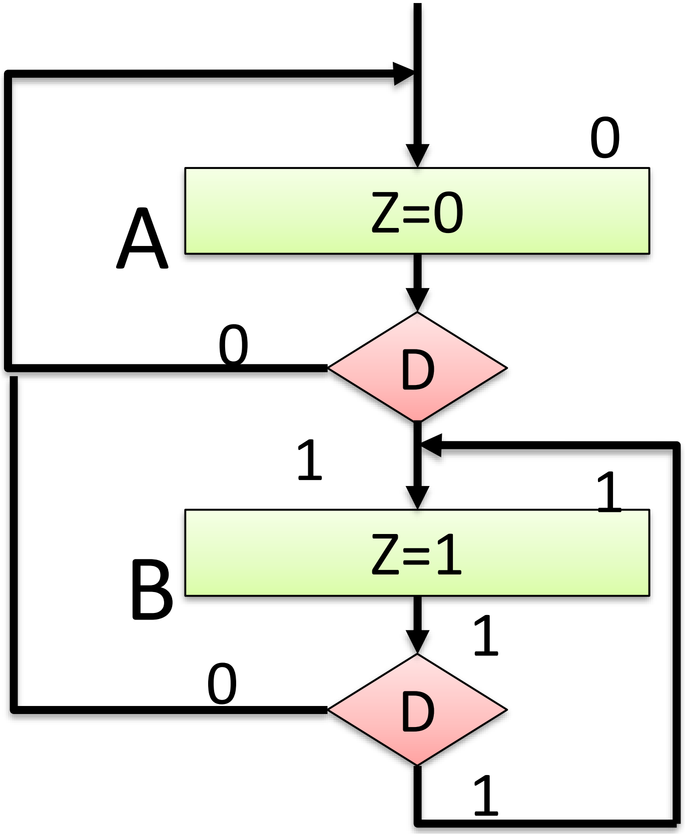
ASMs representing simple FSMs

- Algorithmic state machines can model both
  - Mealy FSM
  - Moore Finite State Machines
- **They can also model machines that are of the mixed type**

Example 1: Draw ASM of D-FF

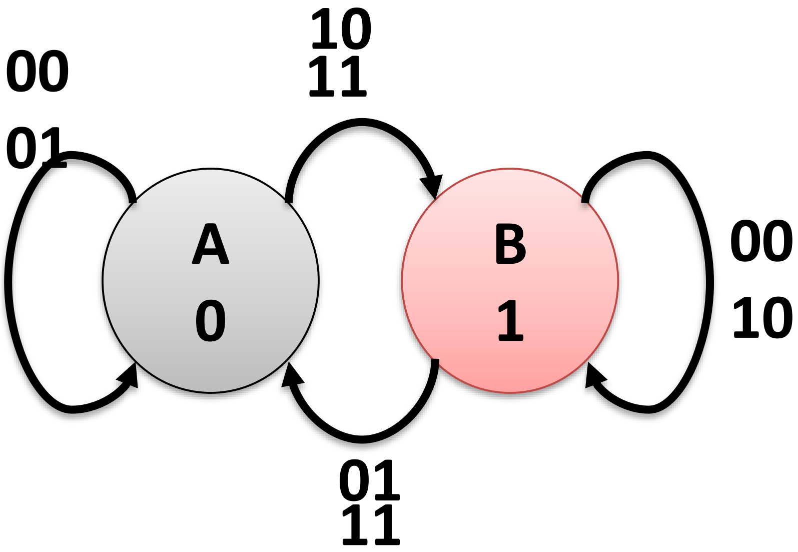


FSM

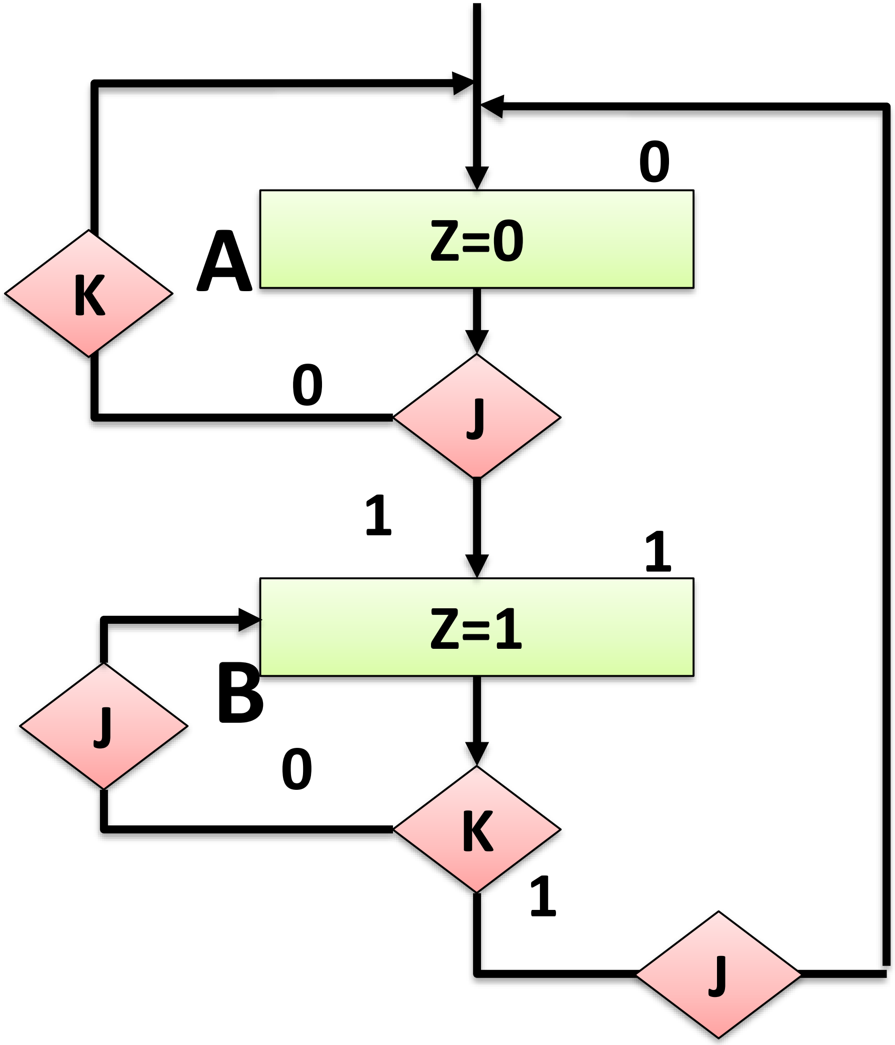


ASM

Example 2: Draw ASM of JK-FF

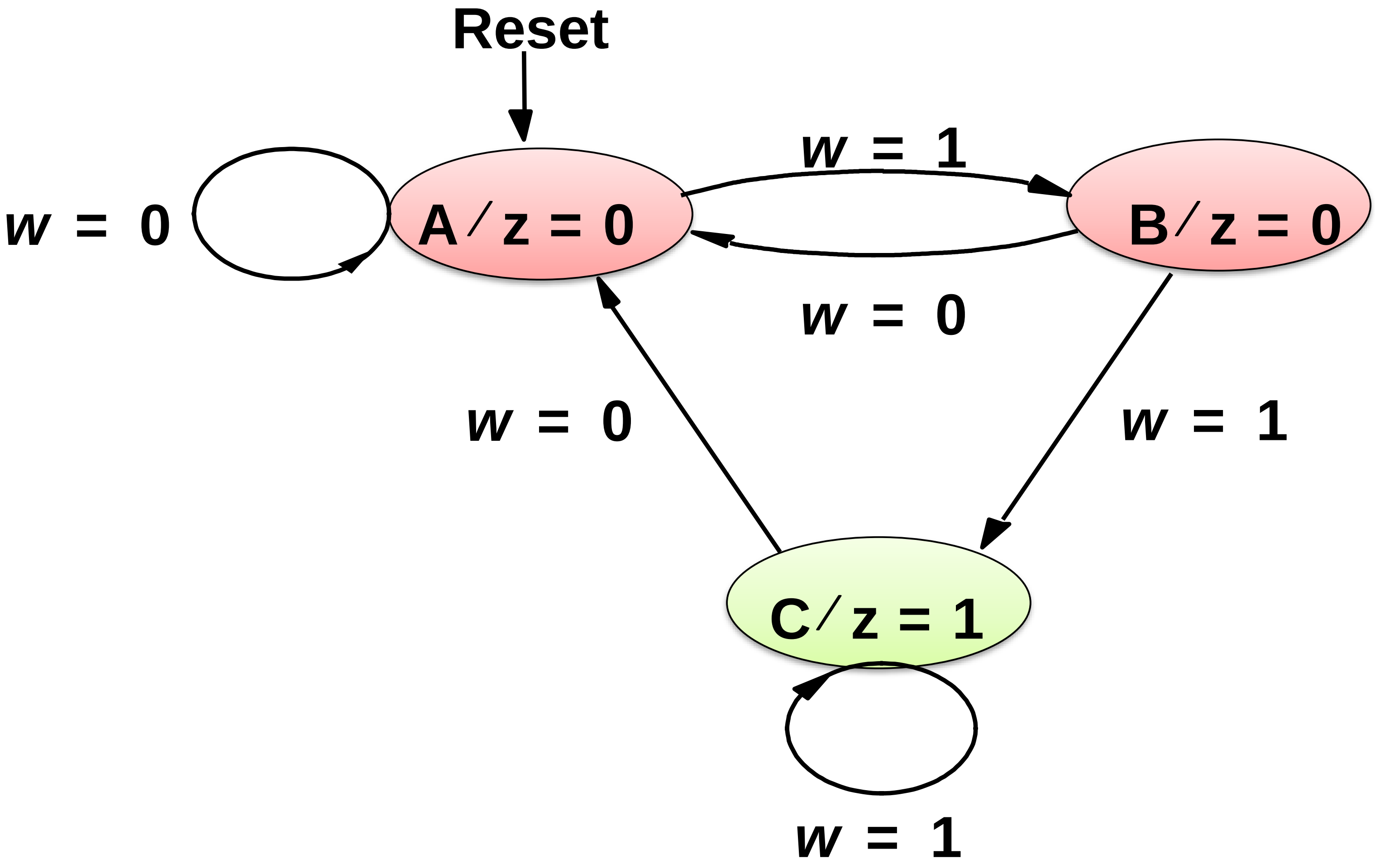


FSM



ASM

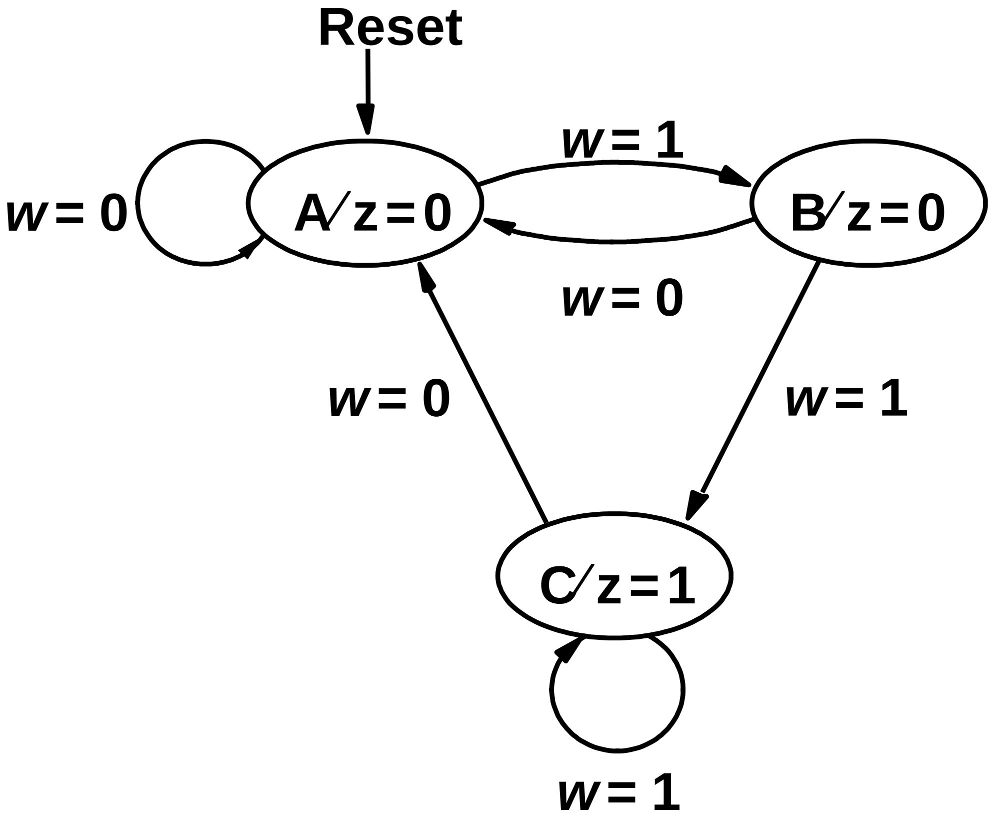
Moore FSM – Example 3: Sequence of two 1’s



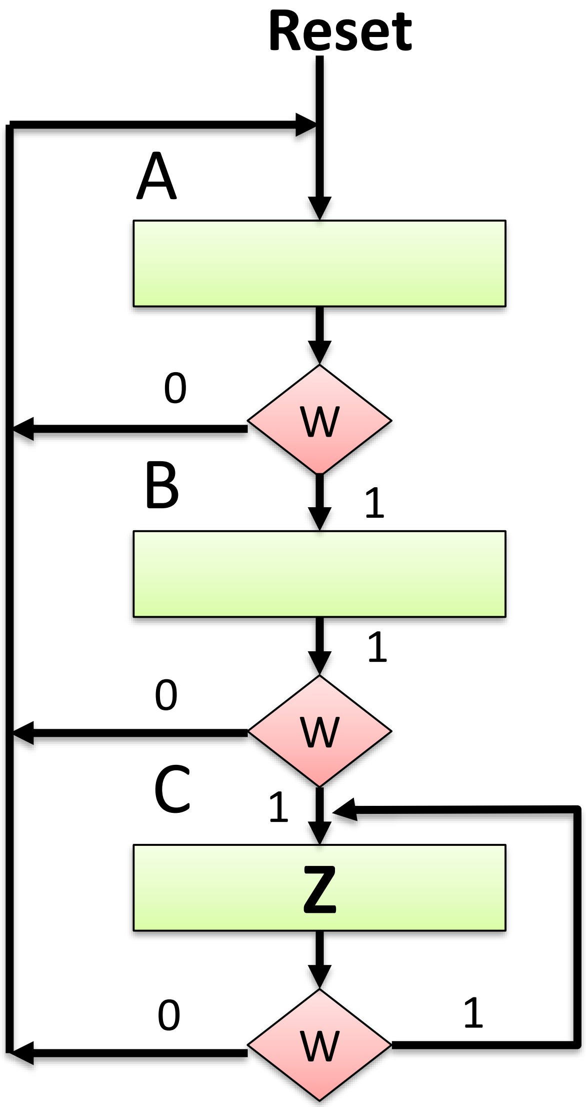
Moore FSM – Example 3: Sequence of two 1’s

Present state	Next state		Output
	$w = 0$	$w = 1$	z
A	A	B	0
B	A	C	0
C	A	C	1

Example 3: ASM Chart for Moore FSM

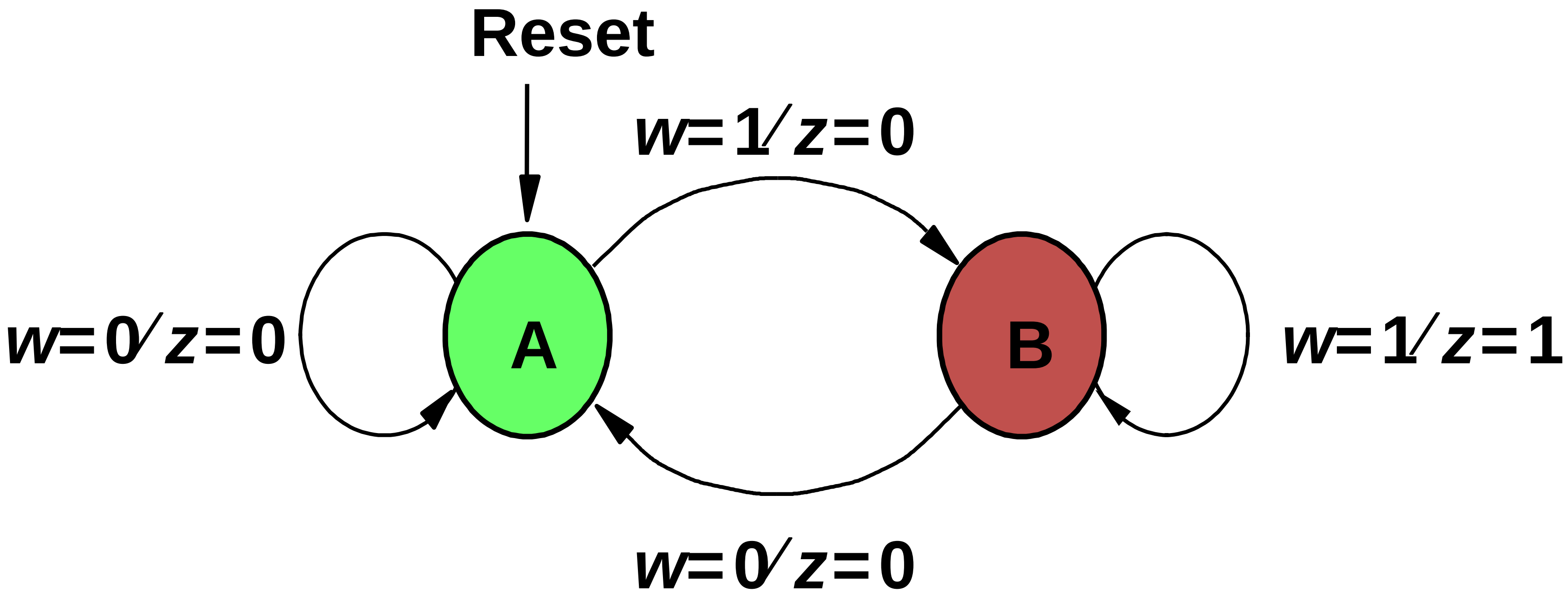


FSM

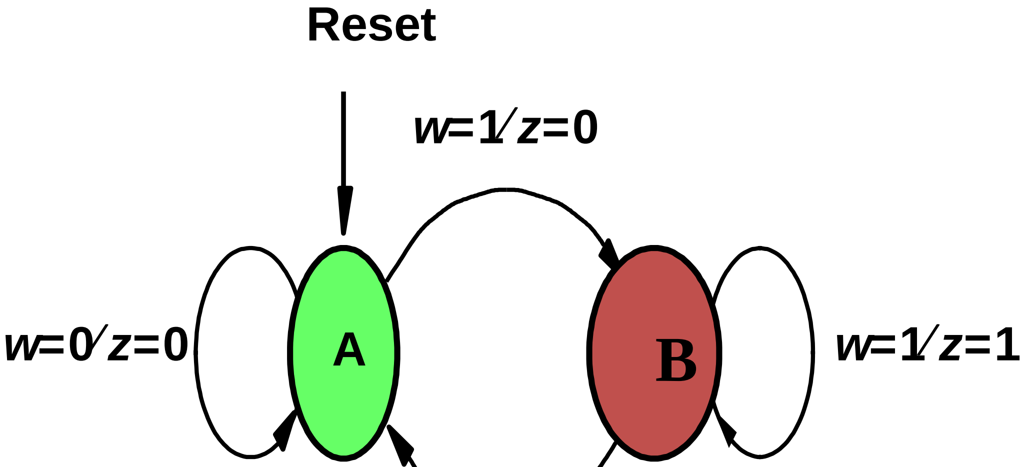


ASM

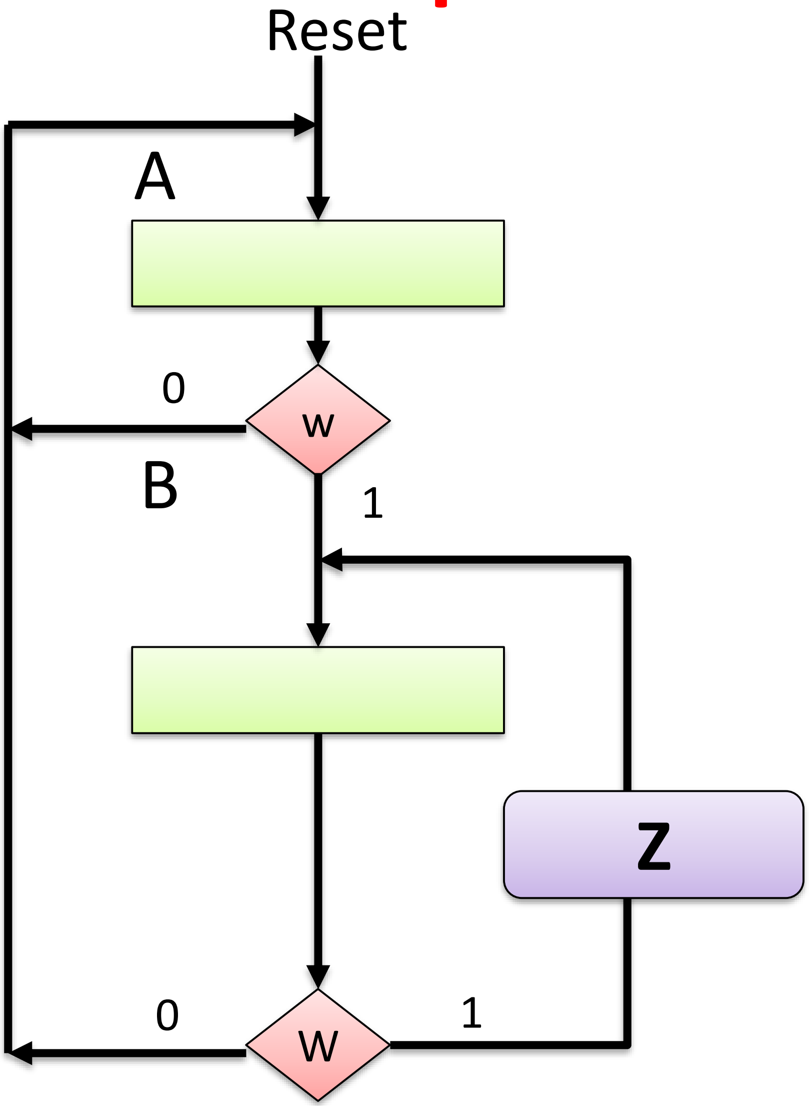
Mealy FSM –Example 4: Sequence of two 1’s



ASM Chart for Mealy FSM – Example 4

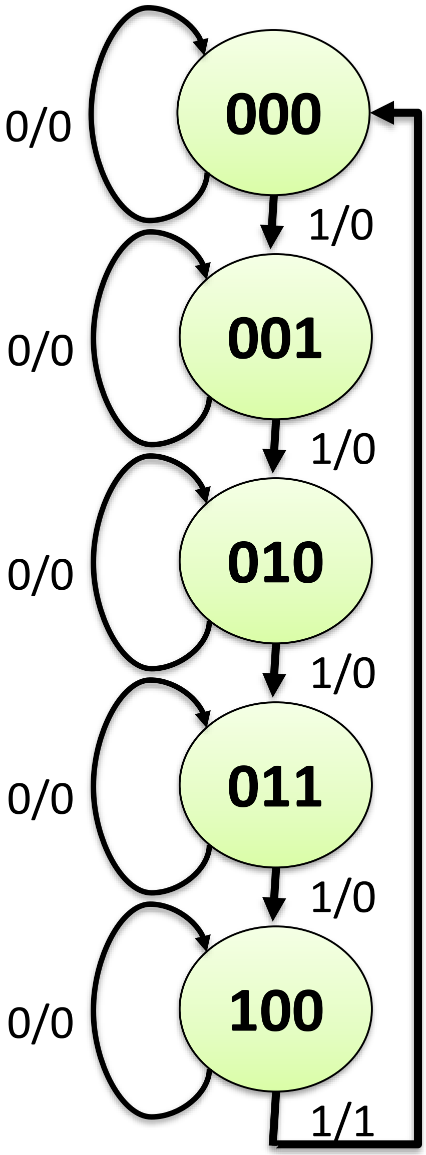


FSM

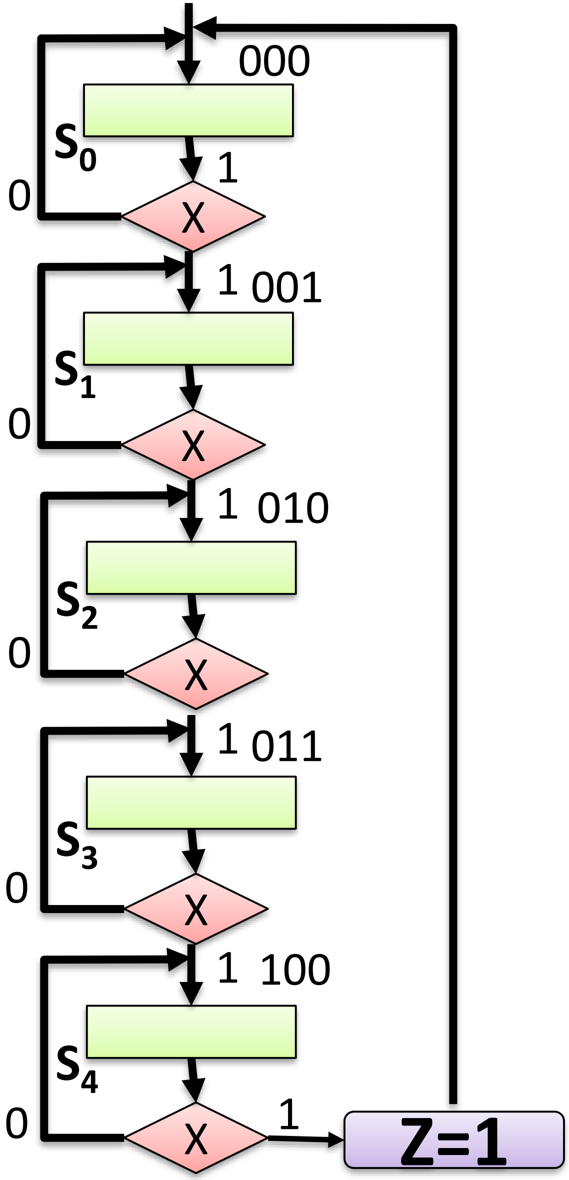


ASM

ASM Chart : Example 5, mod 5 counter



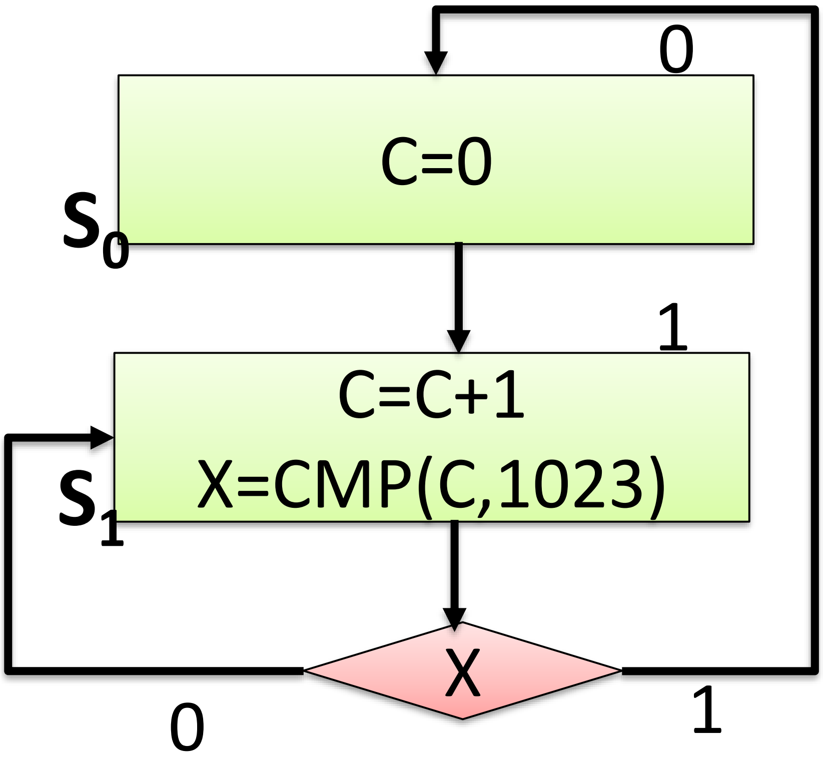
FSM



ASM

ASM Chart : Example 6: 10 bit counter

FSM not possible to Draw/Tabulate

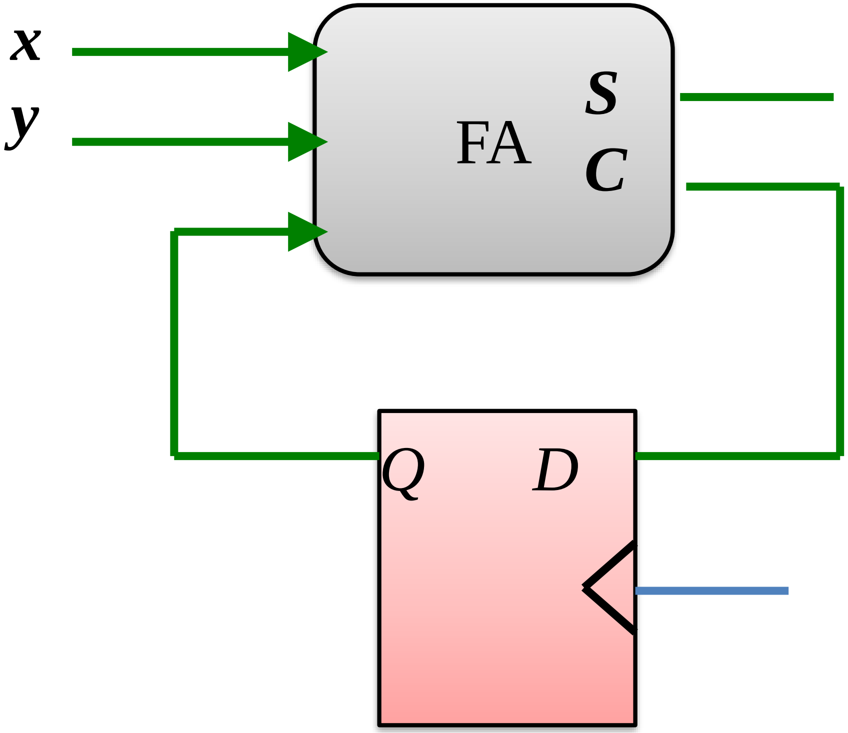


Require : addition, comparison data path

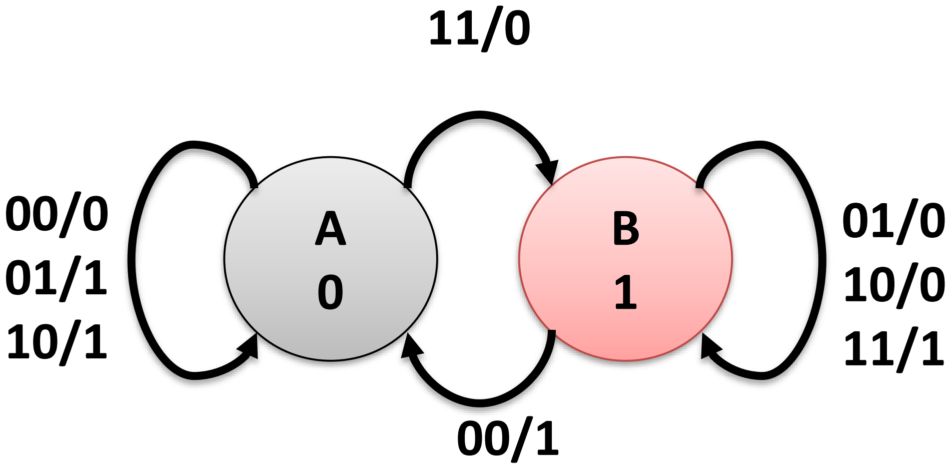
In state : We can put RTL like statement
C=C+1, X=CMP(C,1023)

Remember : Serial Addition

- Model S in terms of X, Y and Q (State)

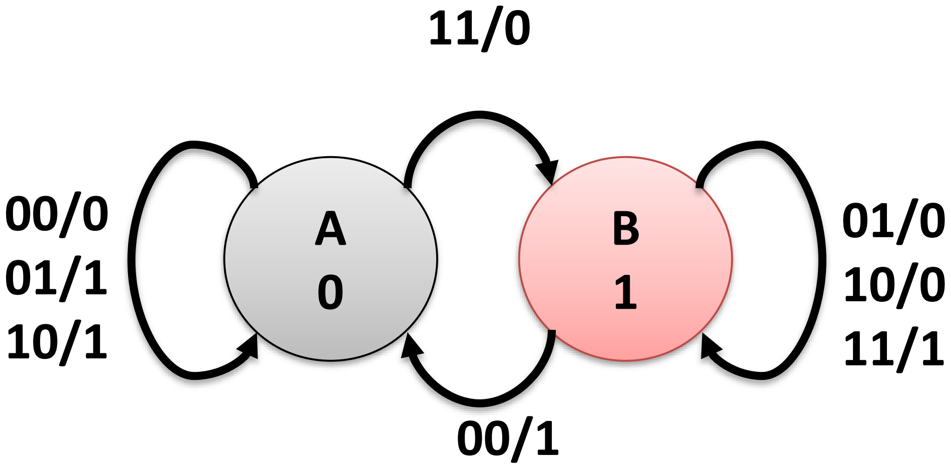


Mealy FSM for Binary Adder

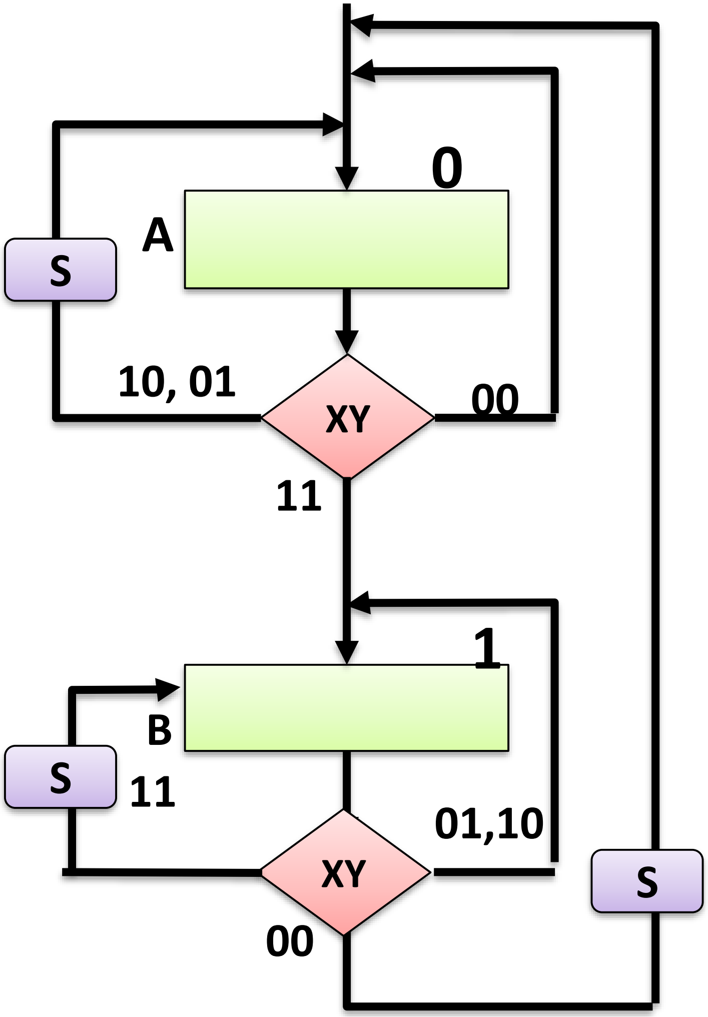


FSM

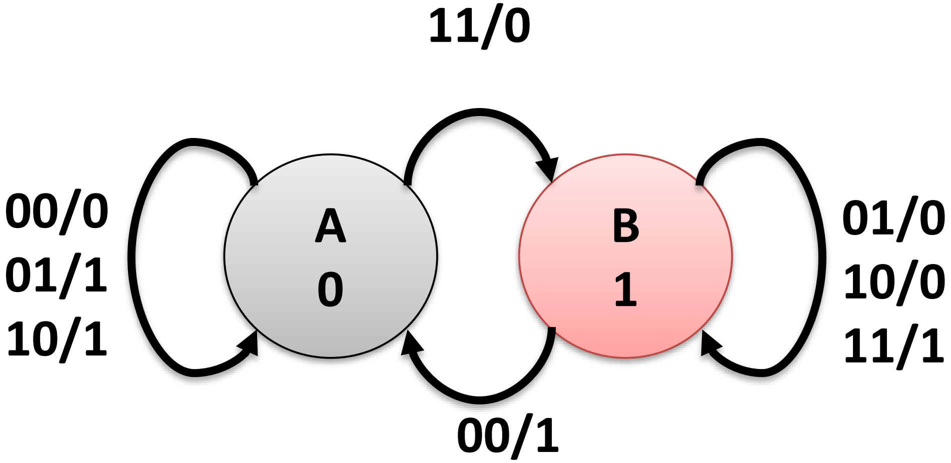
Mealy FSM for Binary Adder



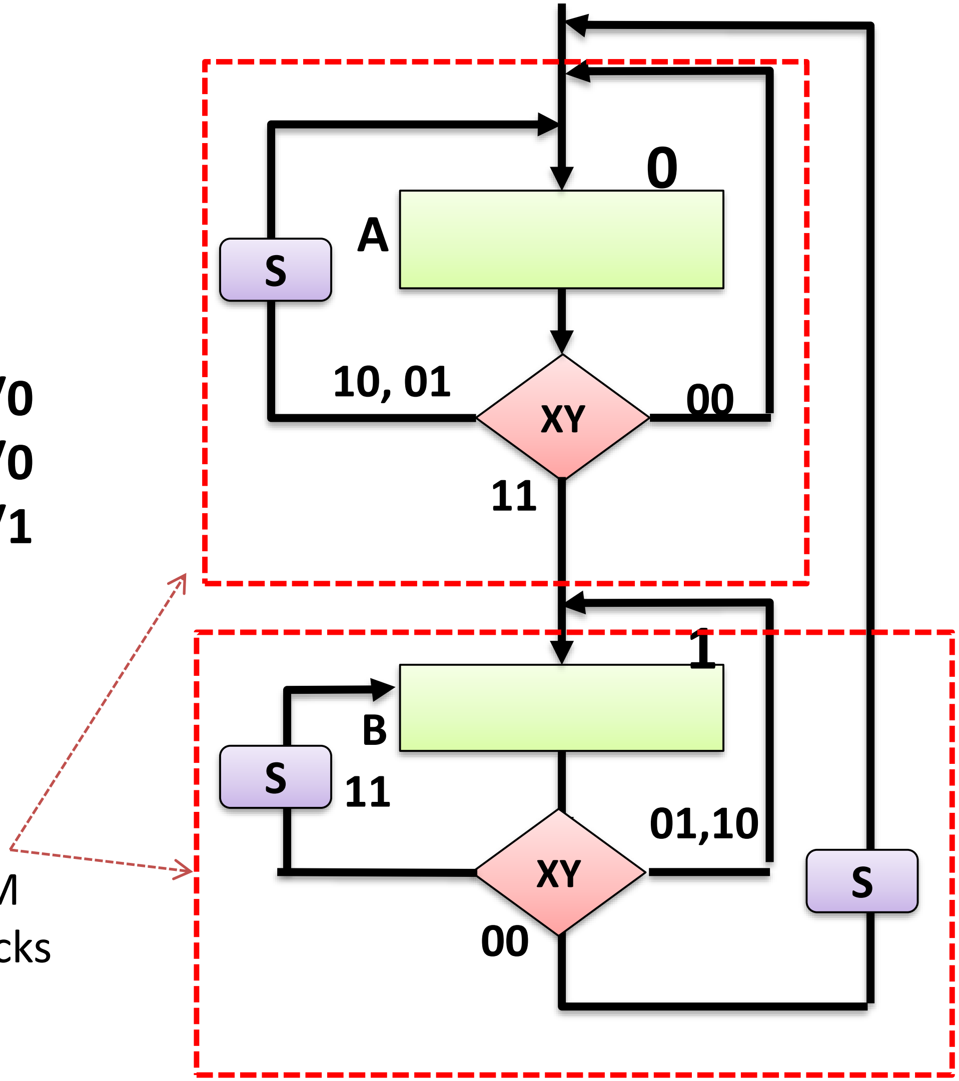
FSM



Mealy FSM for Binary Adder



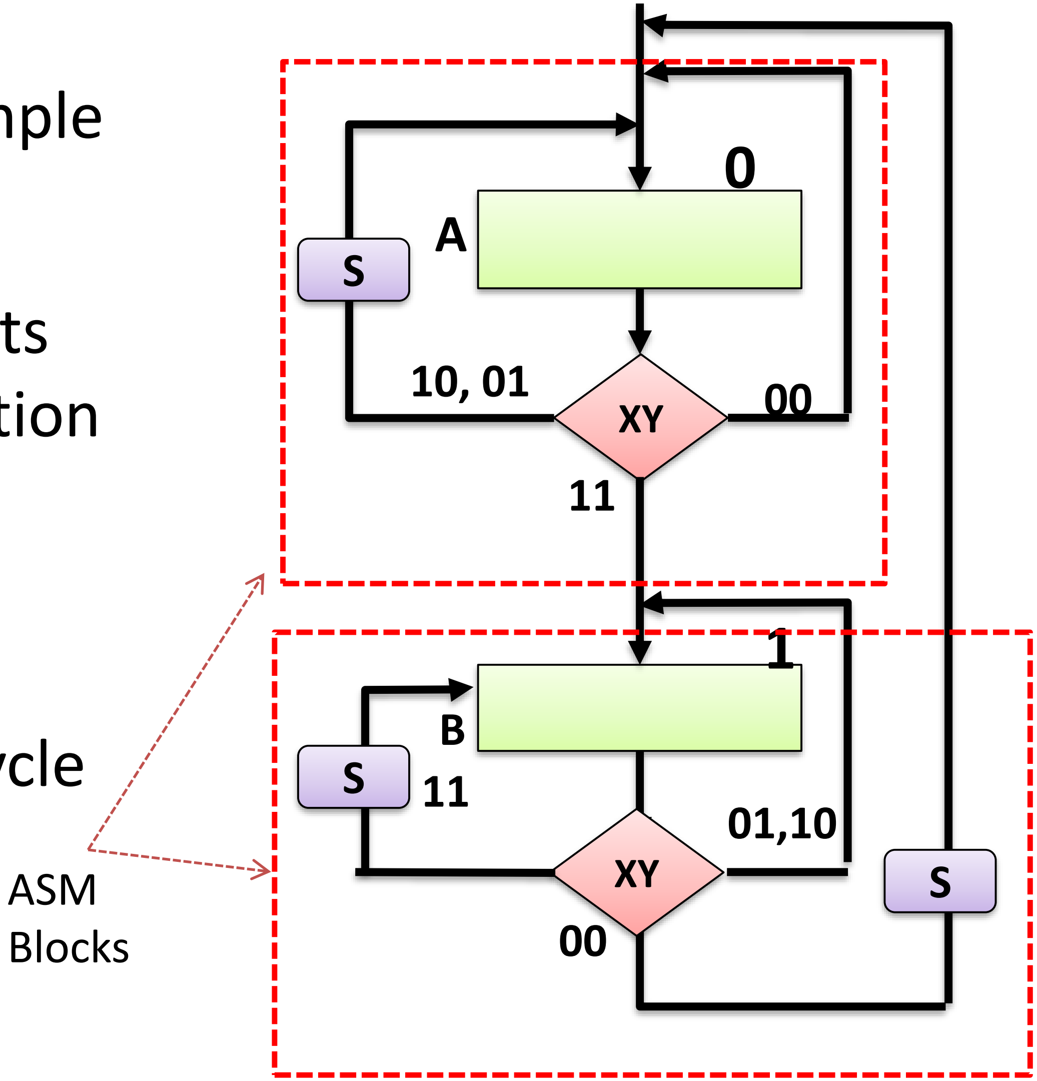
FSM



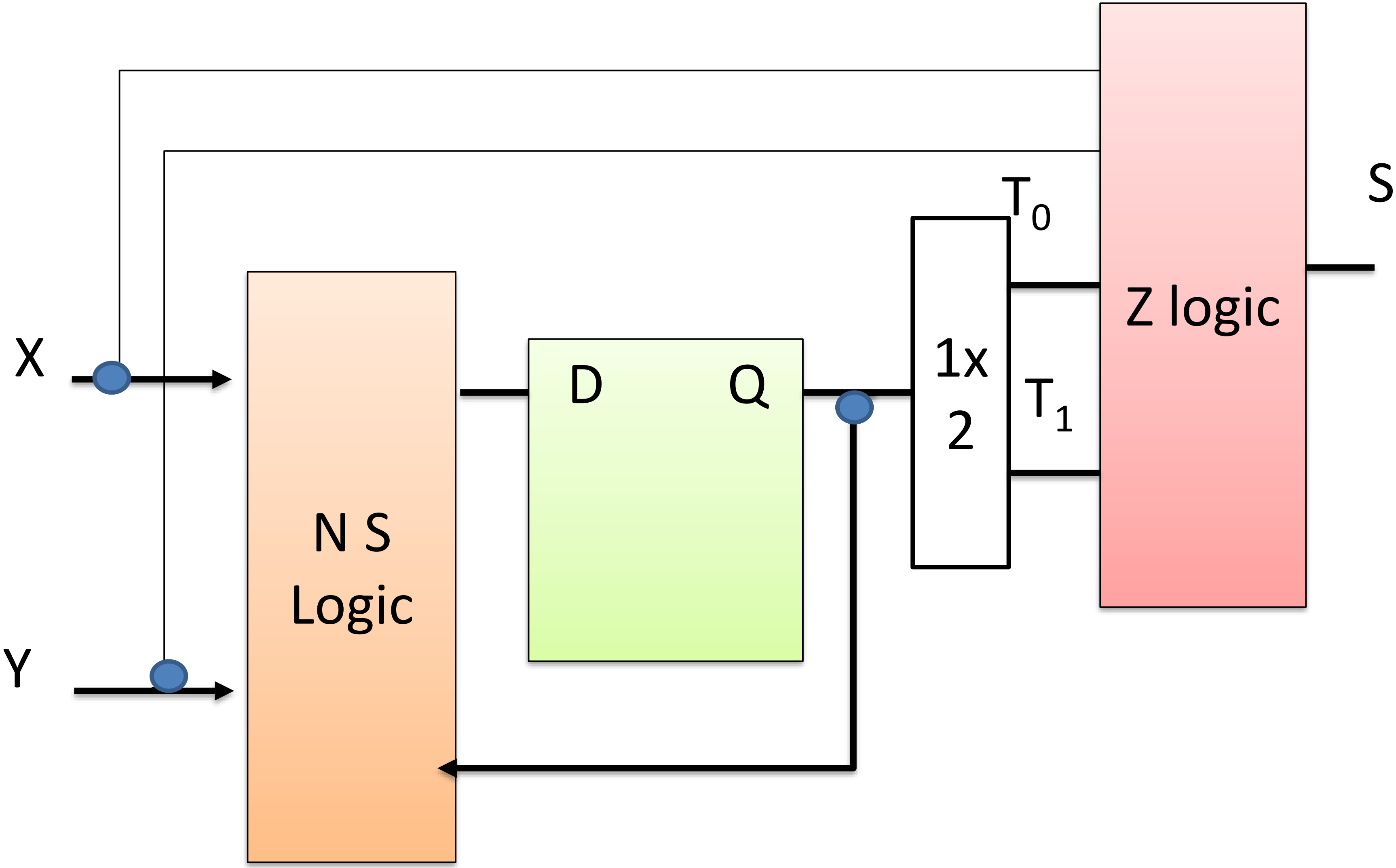
ASM Blocks

Mealy FSM for Binary Adder

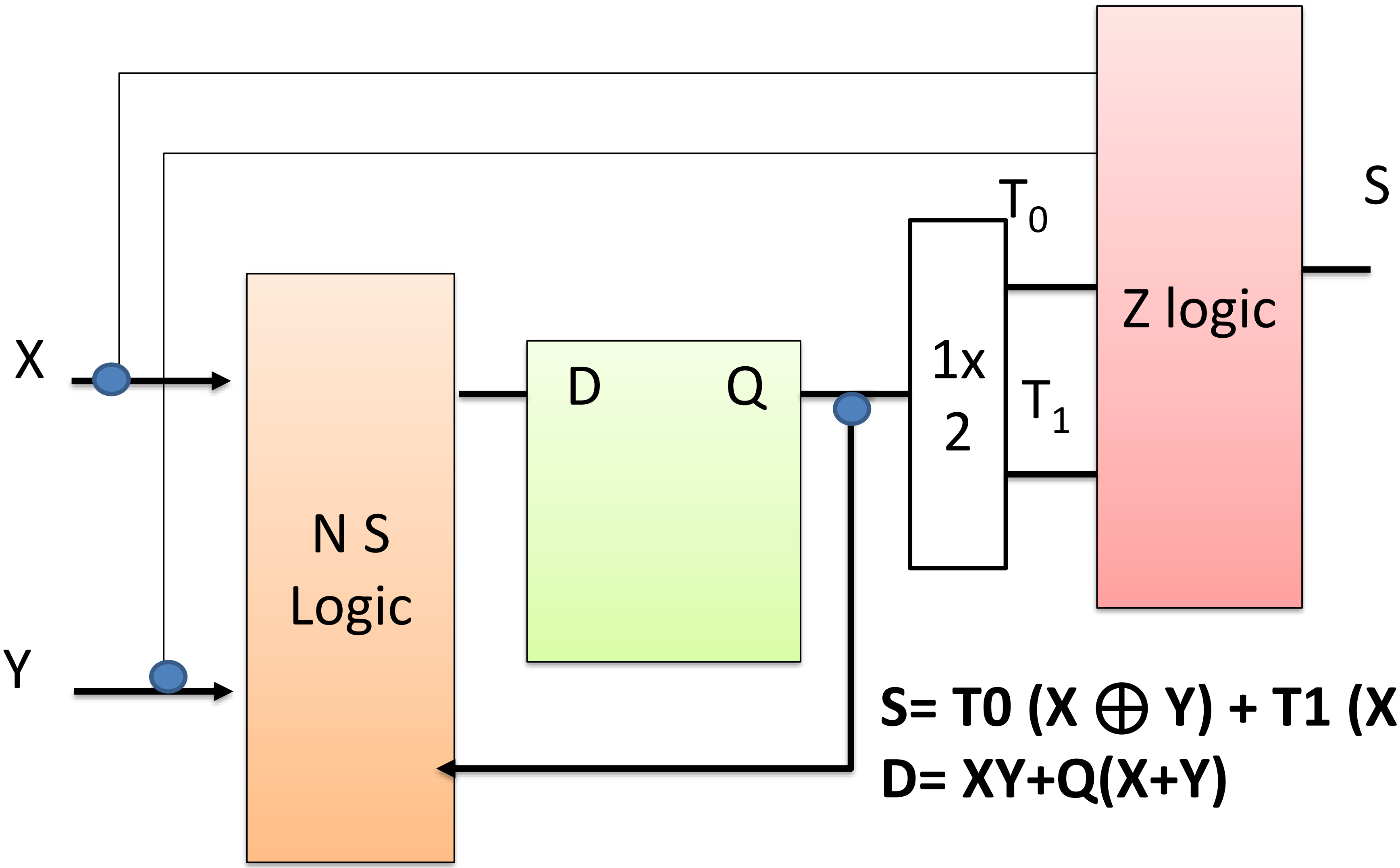
- ASM Blocks
  - Two blocks in this example
- An ASM Block
  - Include a state and all its out going edges, condition boxes and conditional state boxes
  - All the parts of an ASM block execute in one cycle



Mealy FSM for Binary Adder



Mealy FSM for Binary Adder



$S = T0\ (X \oplus Y) + T1\ (X \odot Y)$

$D = XY + Q(X+Y)$