

Statemachines

Exercises Digital Design

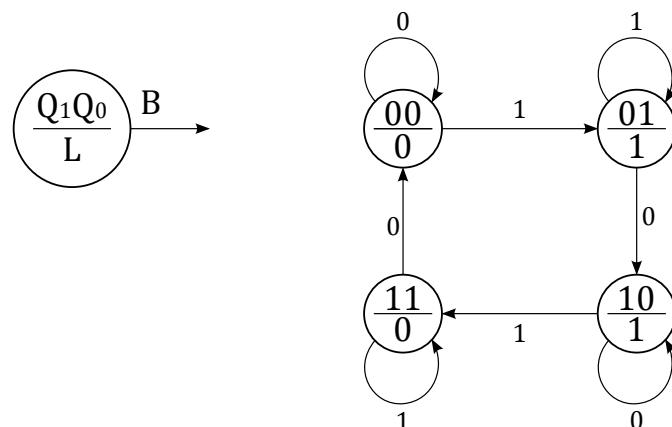


Solution vs. Hints:

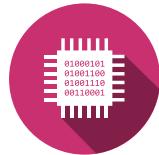
While not every response provided herein constitutes a comprehensive solution, some serve as helpful hints intended to guide you toward discovering the solution independently. In certain instances, only a portion of the solution is presented.

1 | FSM - Moore machines

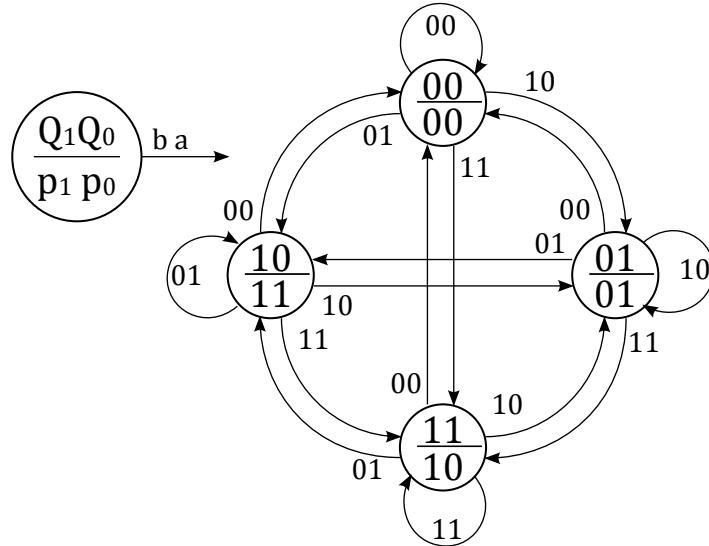
1.1 Graph of a state machine



fsm/moore-01



1.2 Graph of a state machine



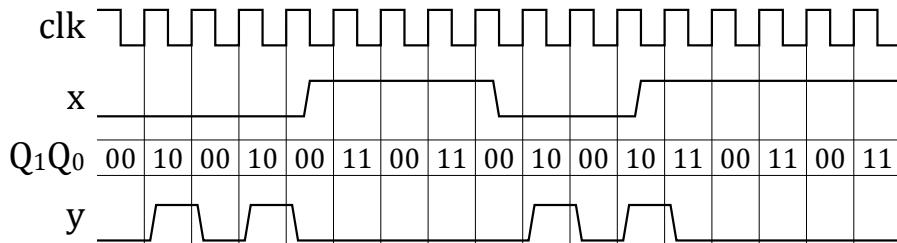
fsm/moore-02

1.3 Sequence of a counter

$\dots \Rightarrow 0 \Rightarrow 1 \Rightarrow 3 \Rightarrow 2 \Rightarrow 6 \Rightarrow 7 \Rightarrow 5 \Rightarrow 4 \Rightarrow 0 \Rightarrow \dots$ (1)

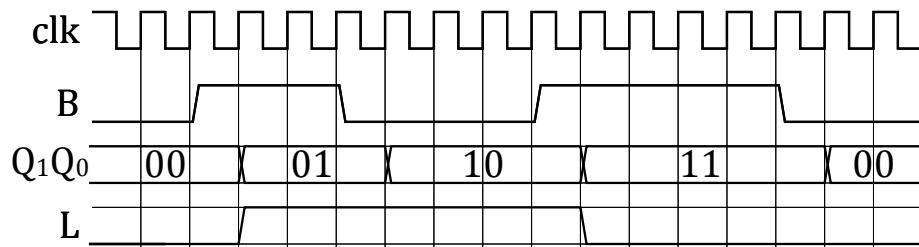
fsm/moore-03

1.4 Temporal behavior of a state machine

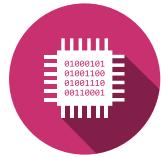


fsm/moore-04

1.5 Temporal behavior of a state machine

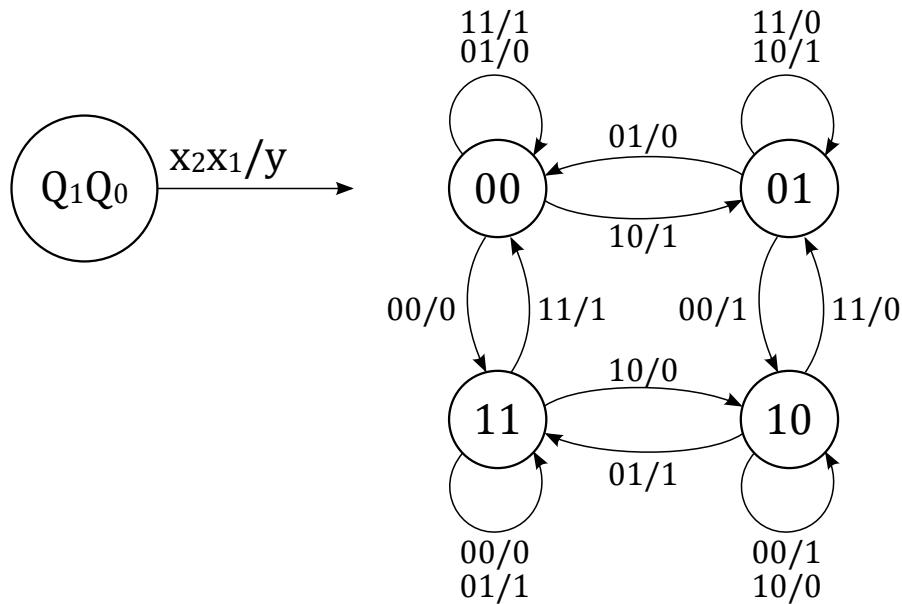


fsm/moore-05



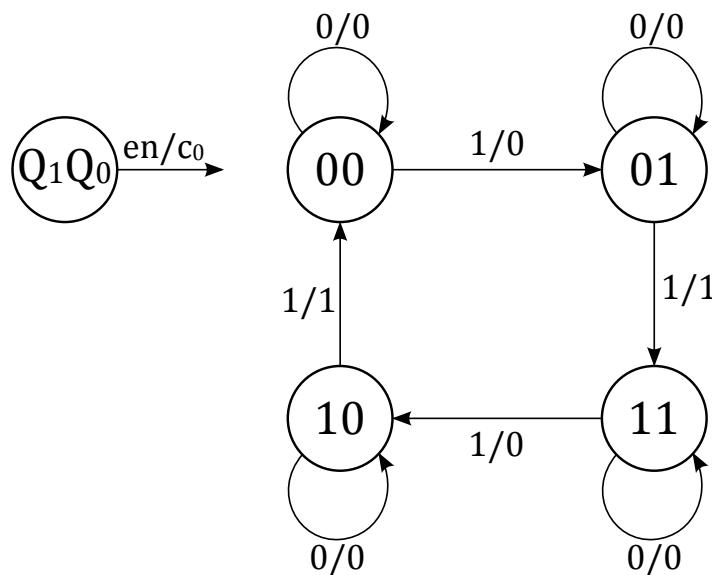
2 | FSM - Mealy machines

2.1 Graph of a state machine

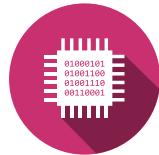


fsm/mealy-01

2.2 Graph of a state machine



fsm/mealy-02



2.3 Temporal behavior of a state machine

2.3.0.1 Initial State

$$x = 0 \Rightarrow Q = "00"$$

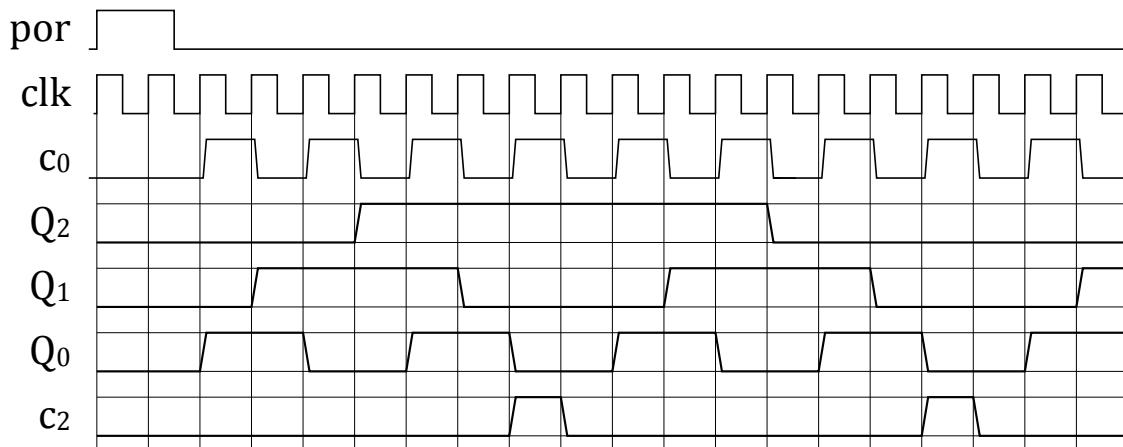
2.3.0.2 Outputs

$$\begin{aligned} y_1 = 1 &\Rightarrow \begin{cases} Q = "10" \& x = 1 \\ Q = "11" \& x = 1 \mid x = 0 \end{cases} \\ y_0 = 1 &\Rightarrow \begin{cases} Q = "01" \& x = 1 \\ Q = "11" \\ Q = "10" \& x = 0 \end{cases} \end{aligned} \quad (2)$$

fsm/mealy-03

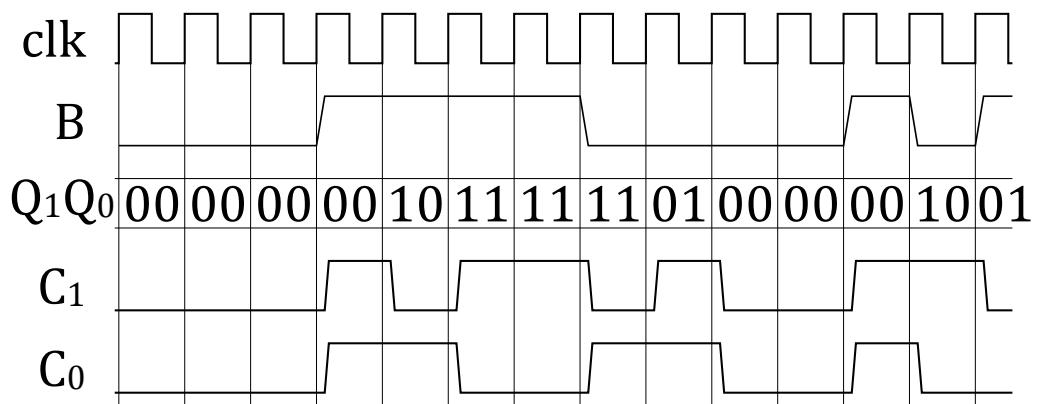
2.4 Iterative Counter

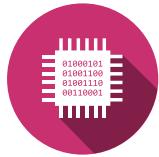
Mealy-Machine since c_2 depends on c_0 & Q_0 & Q_1 .



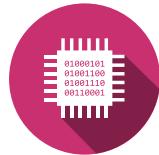
fsm/mealy-04

2.5 Temporal behavior of a state machine



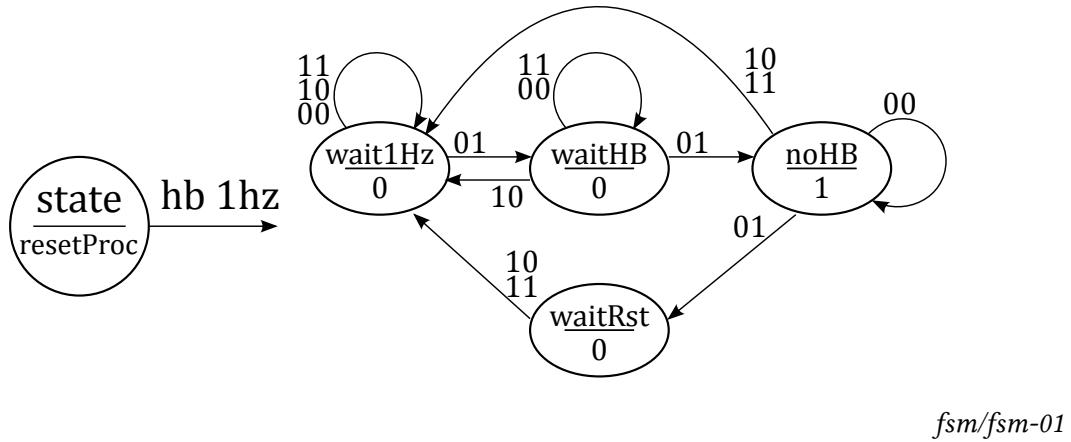


fsm/mealy-05

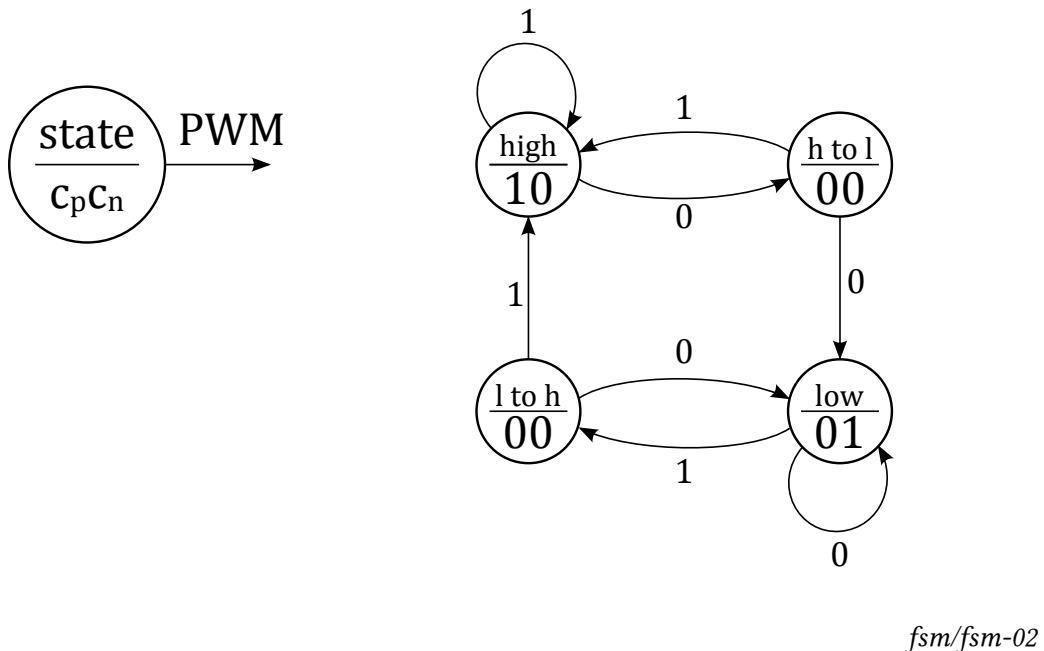


3 | FSM - State graph generation

3.1 Operation Monitoring



3.2 Generator of non-overlapping control signals

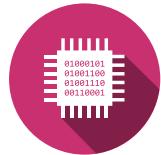


3.3 Control of a Snack Machine

FSM-Type = Moore. There is no realtime action needed

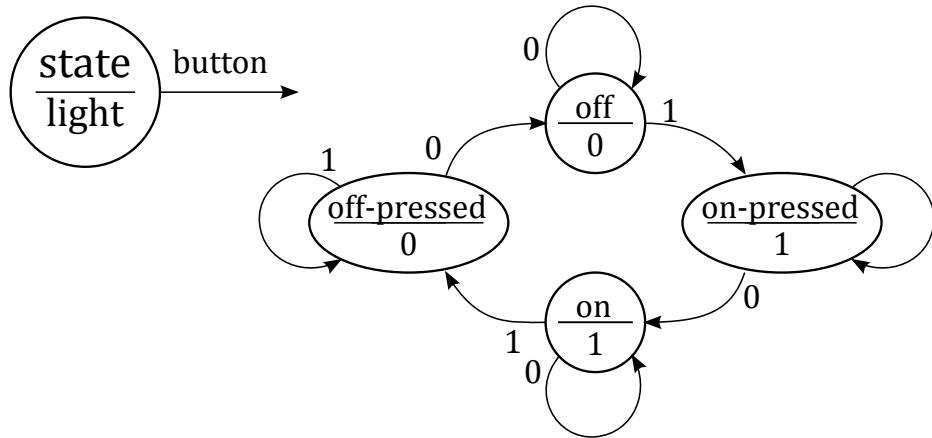
$c_1 c_2 = "11" \Rightarrow$ impossible

fsm/fsm-03



3.4 Lighting control

FSM Type = Moore. There is no realtime action needed.

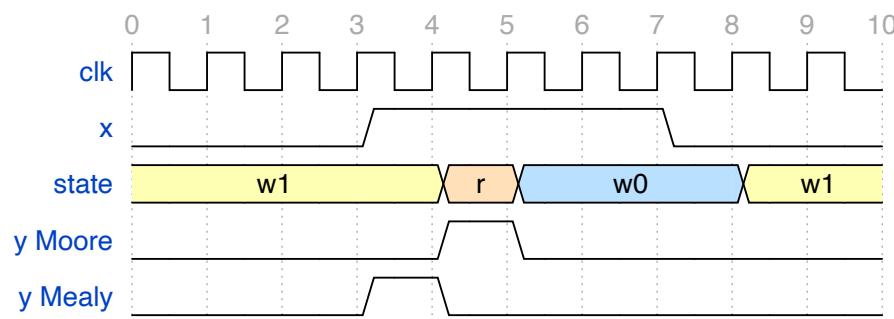


fsm/fsm-04

3.5 Detection of a rising edge

FSM Type = Moore and Mealy possible.

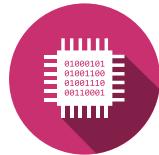
3.5.0.1 Timing Diagram



3.5.0.2 Graph

Moore FSM can be done with 3 states. Mealy FSM can be done with 2 states.

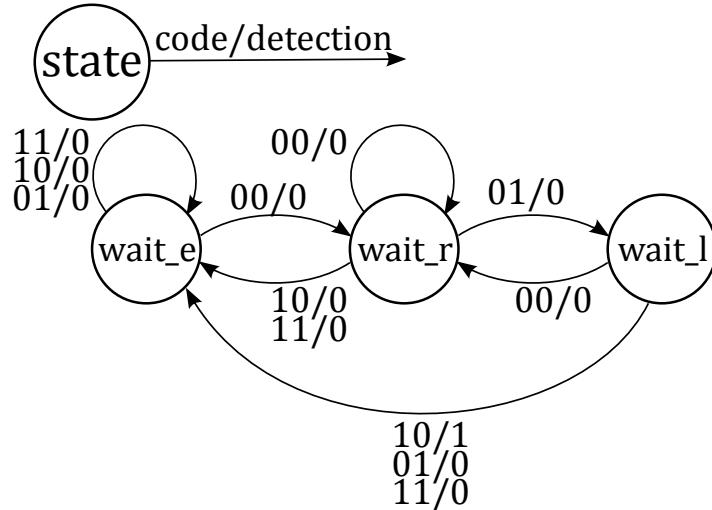
fsm/fsm-05



3.6 Recognition of character strings

FSM-Type = Mealy since an immediate response is needed.

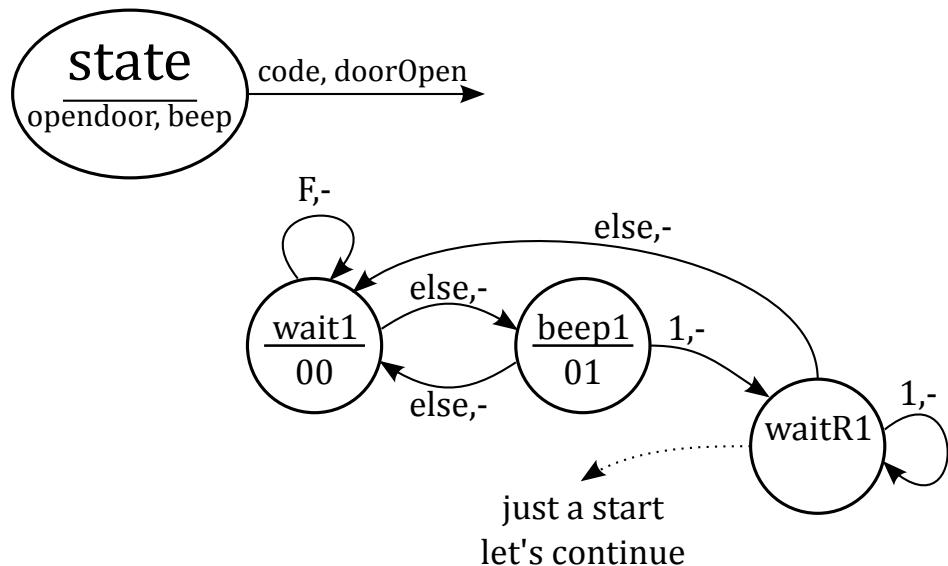
3.6.0.1 Graph



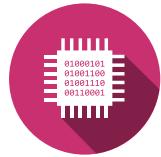
fsm/fsm-06

3.7 Electronic lock

FSM-Type = Moore. The output signal is during one clock period.



fsm/fsm-07



4 | FSM - Graph reduction

4.1 Graph Simplification

4.1.0.1 Truth Table

state \ x	0	1
st0	st0,0	st1,0
st1	st3,0	st2,0
st2	st3,0	st4,1
st3	st0,0	st1,0
st4	st5,1	st7,1
st5	st6,1	st7,1
st6	st0,0	st7,1
st7	st5,1	st4,1

The blue and green states can be combined to new states e.g. **st03** and **st47**. Draw also the new graph.

fsm/reduction-01

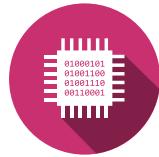
4.2 Graph Simplification

4.2.0.1 Truth Table

state \ $x_1 x_2$	00	01	10	11
st0	st0,0	st2,0	st1,0	st0,0
st1	st1,0	st2,0	st1,0	st3,0
st2	st2,0	st2,0	st1,0	st3,0
st3	st5,1	st4,1	st3,0	st3,0
st4	st4,1	st4,1	st0,0	st3,0
st5	st5,1	st5,1	st0,0	st3,0

The blue and green states can be combined to new states e.g. **st12** and **st45**. Draw also the new graph.

fsm/reduction-02



5 | FSM - State coding

5.1 Logic Circuit

$$\begin{aligned}
 Q_2^+ &= D_2 = \overline{x \oplus Q_2} \\
 Q_1^+ &= D_1 = \bar{x} \overline{Q_2} \overline{Q_1} Q_0 + \bar{x} Q_2 Q_1 Q_0 + x \overline{Q_2} Q_1 \overline{Q_0} + x Q_2 \overline{Q_1} \overline{Q_0} \\
 Q_0^+ &= D_0 = \bar{x} \overline{Q_2} \overline{Q_1} \overline{Q_0} + \bar{x} Q_2 \overline{Q_1} Q_0 + x \overline{Q_2} \overline{Q_1} Q_0 + x Q_2 Q_1 \overline{Q_0} \\
 y_1 &= Q_2 \\
 y_2 &= Q_2 \overline{Q_1} \overline{Q_0}
 \end{aligned} \tag{3}$$

fsm/coding-01

5.2 Logic Circuit

$$\begin{aligned}
 Q_1^+ &= x(Q_1 + Q_0) \\
 Q_0^+ &= xQ_1 + x\overline{Q_0} \\
 y_1 &= Q_1 Q_0 + xQ_1 \\
 y_0 &= \bar{x}Q_1 + xQ_0
 \end{aligned} \tag{4}$$

fsm/coding-02

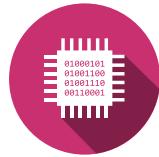
5.3 Logic Circuit

One-Hot Encoding Scheme was used.

$$\left. \begin{array}{l} \text{all arrows to a state} \\ \text{states were the output is set} \end{array} \right\} \begin{aligned}
 D_0 &= Q_0 \overline{\text{step}} + Q_7 \text{step cw} + Q_1 \text{step } \overline{\text{cw}} \\
 D_1 &= Q_1 \overline{\text{step}} + Q_0 \text{step cw} + Q_2 \text{step } \overline{\text{cw}} \\
 D_2 &= Q_2 \overline{\text{step}} + Q_1 \text{step cw} + Q_3 \text{step } \overline{\text{cw}} \\
 D_3 &= Q_3 \overline{\text{step}} + Q_2 \text{step cw} + Q_4 \text{step } \overline{\text{cw}} \\
 D_4 &= Q_4 \overline{\text{step}} + Q_3 \text{step cw} + Q_5 \text{step } \overline{\text{cw}} \\
 D_5 &= Q_5 \overline{\text{step}} + Q_4 \text{step cw} + Q_6 \text{step } \overline{\text{cw}} \\
 D_6 &= Q_6 \overline{\text{step}} + Q_5 \text{step cw} + Q_7 \text{step } \overline{\text{cw}} \\
 D_7 &= Q_7 \overline{\text{step}} + Q_6 \text{step cw} + Q_0 \text{step } \overline{\text{cw}}
 \end{aligned} \tag{5}$$

$$\left. \begin{array}{l} \text{all arrows to a state} \\ \text{states were the output is set} \end{array} \right\} \begin{cases} c_1 = Q_0 + Q_1 + Q_7 \\ c_2 = Q_1 + Q_2 + Q_3 \\ c_3 = Q_3 + Q_4 + Q_5 \\ c_4 = Q_5 + Q_6 + Q_7 \end{cases}$$

fsm/coding-03



5.4 Logic Circuit

Additional signal

The states Q_1 and Q_0 can distinguish 4 different clock periods. But the signal as 8 clockperiods repeating as a mirror.

⇒ An additional signal is needed, to differentiate.

5.4.0.1 Truth table

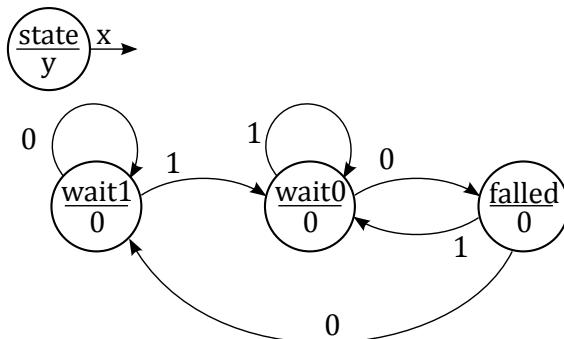
Q_2	Q_1	Q_0	Q_2^+	Q_1^+	Q_0^+	c_1
0	0	0	0	0	1	0
0	0	1	0	1	1	0
0	1	0	1	1	0	0
0	1	1	0	1	0	0
1	0	0	0	0	0	1
1	0	1	1	0	0	0
1	1	0	1	1	1	0
1	1	1	1	0	1	0

5.4.0.2 Equations

$$\begin{aligned} D_2 &= Q_0 Q_2 + \overline{Q_0} Q_1 \\ D_1 &= Q_0 \overline{Q_2} + \overline{Q_0} Q_1 \\ D_0 &= Q_1 \oplus \overline{Q_2} \\ c_1 &= Q_2 \overline{Q_1} \overline{Q_0} \end{aligned} \quad (6)$$

fsm/coding-04

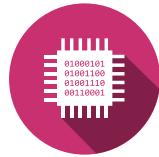
5.5 Detection of a falling edge



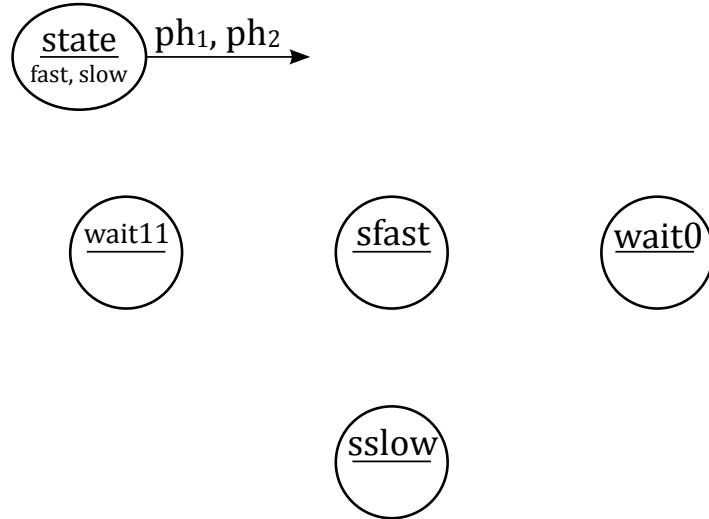
state	Q Encoding
wait1	10
wait0	00
falled	11

Next steps is to create the truth table and the Equations in order to draw the circuit.

fsm/coding-05



5.6 Phase Detector



5.6.0.1 State encoding (One-Hot)

state	Q Encoding
wait11	0001
sfast	0010
sslow	0100
wait0	1000

5.6.0.2 Equations

$$\begin{aligned}
 D_0 &= \text{ph}_1 \text{ph}_2 \\
 D_1 &= (Q_0 + Q_1) \overline{\text{ph}}_1 \text{ph}_2 \\
 D_2 &= (Q_0 + Q_2) \text{ph}_1 \overline{\text{ph}}_2 \\
 D_3 &= \overline{\text{ph}}_1 \overline{\text{ph}}_2 + (Q_1) \text{ph}_1 \overline{\text{ph}}_2 + (Q_2) \overline{\text{ph}}_1 \text{ph}_2 + Q_3 \stackrel{(7)}{=} (\text{ph}_1 \oplus \text{ph}_2) \\
 \text{fast} &= Q_1 \\
 \text{slow} &= Q_2
 \end{aligned}$$

fsm/coding-06