Digitalteknik EITF65

Lecture 11: Asynchronous Sequential Circuits

October 1, 2018

Mealy

Definition

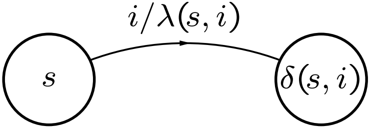
A Mealy graph is defined by the five-tuple $\mathcal{M} = \{\mathcal{I}, \mathcal{S}, \mathcal{Z}, \delta, \lambda\}$ where

- δ is the state transition function,

$$\delta : \mathcal{S} \times \mathcal{I} \to \mathcal{S}$$

- λ is the output function,

$$\lambda : \mathcal{S} \times \mathcal{I} \to \mathcal{Z}$$



Moore

Definition (6.6)

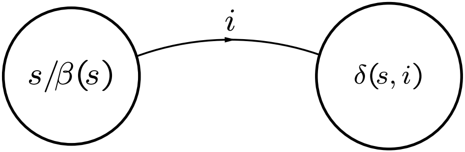
A Moore graph is defined by the five-tuple $\mathcal{M} = \{\mathcal{I}, \mathcal{S}, \mathcal{Z}, \delta, \beta\}$ where

- δ is the state transition function,

$$\delta : \mathcal{S} \times \mathcal{I} \rightarrow \mathcal{S}$$

- β is the output function,

$$\beta : \mathcal{S} \rightarrow \mathcal{Z}$$

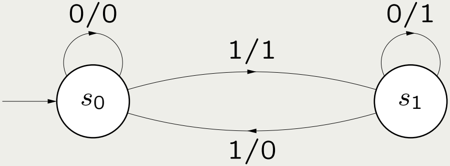


Parity check

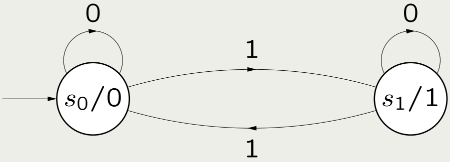
Example 6.7

The (even/odd) parity check graph can be written as a

- Mealy graph:



- Moore graph:



Conversion

Mealy $\rightarrow$ Moore

All Mealy graphs can be rewritten as Moore graphs if we allow that

- the output is delayed one step.
- we might need more states.

Conversion:

- Split the states such that all entering edges into a state have the same output.
- Move the output into the state pointed out by the edge (next state).

Conversion

Moore $\rightarrow$ Mealy

All Moore graphs can be rewritten as Mealy graphs if we allow that the output is affected directly (asynchronously) by the input.

Conversion:

- Move the output to the entering edges.
- Use the RF-algorithm.

Asynchronous sequential circuits

Asynchronously realizable

Definition (6.7)

A state s is stable for the input i if

$$\delta(s, i) = s$$

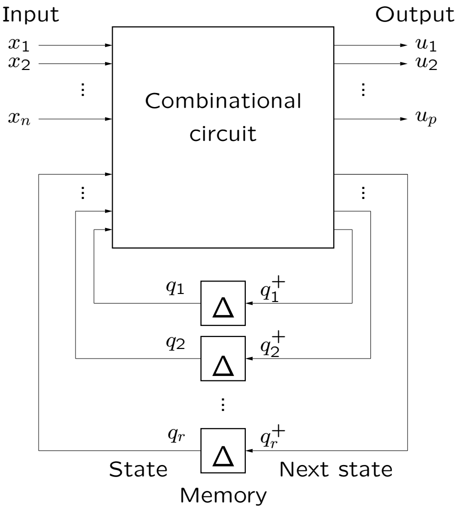
i.e., if i gives an edge back to s.

If there is a path for input i from state s_0 to the stable state s then s is a successor state of s_0 for input i.

Definition (6.8)

A graph is asynchronously realizable if all states have successor states for all inputs.

Sequential circuit (canonical form)



Race free state assignment

Definition

In a race free state assignment only one state variable changes when a state changes.

It is always possible to rewrite an asynchronously realizable graph such that it can be encoded race free. One way is to think of the states as corners in an (n-dimensional) cube. The allowed transitions are the edges of the cube.

Hazard free realization (C5.3)

Due to different delays in the components, there might appear transients in the output. This is called hazard.

To avoid hazard the realization must be hazard free.
That is, *all* prime implicants must be in the function.

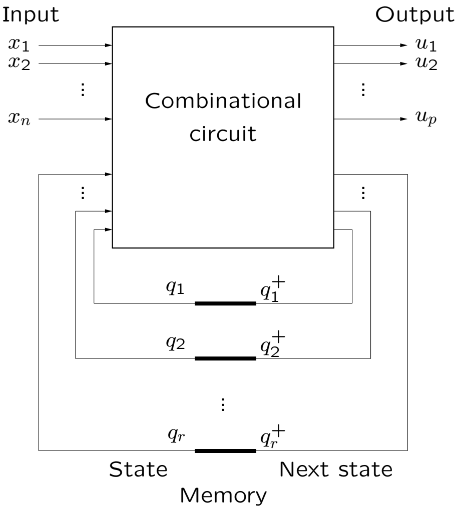
Asynchronous sequential circuit

An asynchronous sequential circuit is not clock controlled, and the states are updated continuously. That gives an event controlled circuit.

Therefore, we need that

- the graph is asynchronously realizable.
- the state assignment is race free.
- the functions are hazard free.

Sequential circuit (canonical form)



Latch

Example 6.13

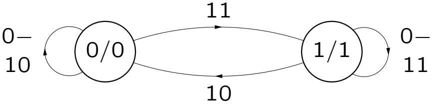
A latch is a simple memory element. Use the signal ϕ to control the output such that

$$z = \begin{cases} x, & \text{if } \phi = 1 \\ x_0, & \text{if } \phi = 0 \end{cases}$$

where x_0 is the input the latest occasion $\phi = 1$.

Latch (realisation)

Graph:



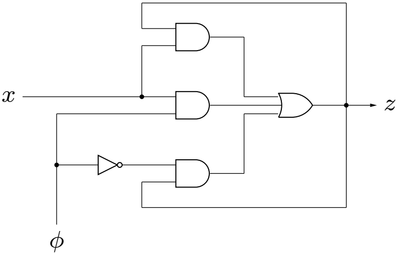
Functions:

$$q^+ = q\phi' \vee qx \vee \phi x$$
$$z = q$$

Karnaugh map:

q^+ \ ϕx	00	01	11	10
q = 0	0	0	1	0
q = 1	1	1	1	0

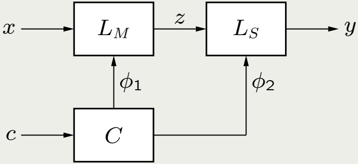
Realisation:



D-element

Example 6.14

To avoid direct connection between the input and the output we cascade two latches.



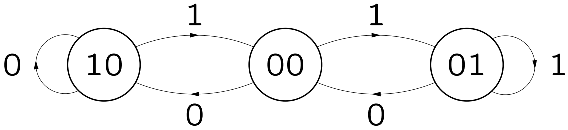
The control circuit C:

$$(\phi_1, \phi_2) = \begin{cases} (0,1), & c = 1 \\ (1,0), & c = 0 \end{cases}$$

At the transition $(\phi_1, \phi_2) = (0, 0)$.

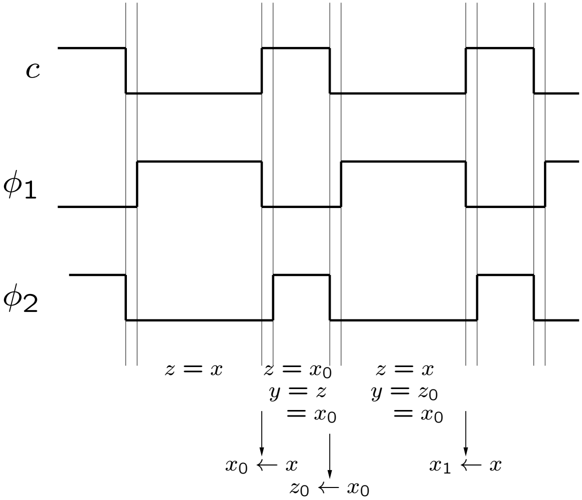
C circuit (graph)

Moore graph for the C circuit:



D-element (time schedule)

Time schedule for c, ϕ_1, and ϕ_2:



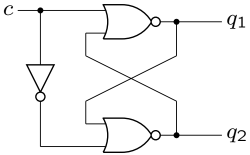
C circuit (realisation)

Karnaugh maps:

q_1^+ (q_1q_2 \ c)	0	1
00	1	0
01	0	0
11	-	-
10	1	0

q_2^+ (q_1q_2 \ c)	0	1
00	0	1
01	0	1
11	-	-
10	0	0

$$q_1^+ = q_2'c' = (q_2 \vee c)'$$
$$q_2^+ = q_1'c = (q_1 \vee c')'$$



D-element (circuit)