EDA321: Digital Design
Solutions of Re-Exam - August 2013

Question 1: (5 points)

Apply deMorgan Theorems to the expressions:

$$\overline{(A+B+C)D}$$

$$\overline{ABC+DEF}$$

$$\overline{A\bar{B}+\bar{C}D+EF}$$

$$\overline{\overline{A+B\bar{C}}+D(\overline{E+\bar{F}})}$$

Solution:

$$\overline{(A+B+C)\cdot D}=\overline{A+B+C}+\overline{D}=\bar{A}\cdot\bar{B}\cdot\bar{C}+\overline{D}$$

$$\overline{A\cdot B\cdot C+D\cdot E\cdot F}=\overline{A\cdot B\cdot C}\cdot\overline{D\cdot E\cdot F}=(\bar{A}+\bar{B}+\bar{C})\cdot(\bar{D}+\bar{E}+\bar{F})$$

$$\overline{A\cdot\bar{B}+\bar{C}\cdot D+E\cdot F}=\overline{A\cdot\bar{B}}\cdot\overline{\bar{C}\cdot D}\cdot\overline{E\cdot F}=(\bar{A}+B)(C+\bar{D})(\bar{E}+\bar{F})$$

$$\overline{\overline{A+B\bar{C}}+D\cdot(\overline{E+\bar{F}})}=(A+B\bar{C})(\overline{D\cdot(\overline{E+\bar{F}})})=(A\cdot B\bar{C})(\bar{D}+(E+\bar{F}))=(A+B\cdot\bar{C})(\bar{D}+E+\bar{F}))$$

Question 2: (15 points)

*Minimize the cost of the following function F(x0,x1,x2,x3,x4) = Σ(6,7,23,24,26,28) + D(14,22,30). Measure the cost of the minimized function by counting the total number of 2-input gates of the circuit (e.g. a*b + c*d, has cost of 3).*

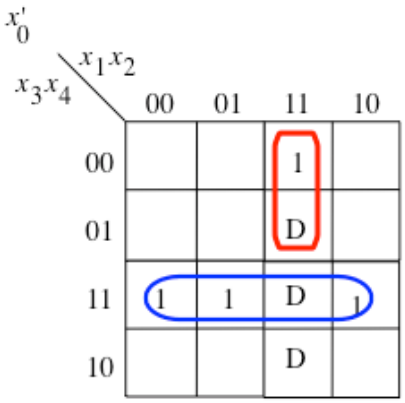
a) Using Quine-McCluskey? (9 points)

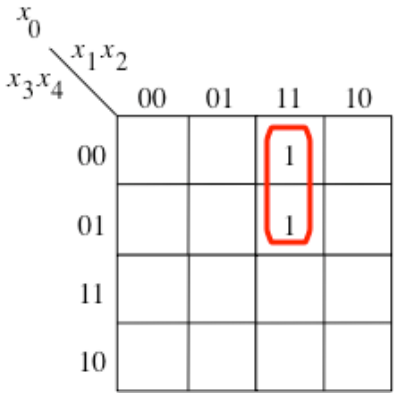
b) using Karnaugh (6 points)

Solution:

$$F(x_0, x_1, x_2, x_3, x_4)=\sum m(6,7,23,24,26,28)+D(14,22,30)$$

a)





$$f(x_0, x_1, x_2, x_3, x_4) = \overline{x_0} \cdot x_3 \cdot x_4 + \overline{x_3} \cdot x_2 \cdot x_1$$

This form has a cost of 5.

b)

using Quine-McCluskey:

6	00110	√
24	11000	√
7	00111	√
14	01110	√
22	10110	√
26	11010	√
28	11100	√
23	10111	√
30	11110	√

6,7 (1)	0011x	√
6,14 (8)	0x110	√
6,22 (16)	x0110	√
24,26 (2)	110x0	√
24,28 (4)	11x00	√
7,23 (16)	x0111	√
14,30 (16)	x1110	√
22,23 (1)	1011x	√
22,30 (8)	1x110	√
26,30 (4)	11x10	√
28,30 (2)	111x0	√

6,7,22,23 (1,16)	x011x	p_1
6,14,22,30 (8,16)	xx110	p_2
24,26,28,30 (2,4)	11xx0	P_3

Prime implicant	*Minterm* 6	7	23	24	26	28
p_1=x011x	√	√	√			
p_2=xx110	√					
p_3=11xx0				√	√	√

P1, P3 are essential prime implicants and cover the function F resulting in the following form:

$$f(x_0, x_1, x_2, x_3, x_4) = \overline{x_0} \cdot x_3 \cdot x_4 + \overline{x_3} \cdot x_2 \cdot x_1$$

This form has a cost of 5.

Question 3: (20 points)

a) *Describe an unsigned binary multiplication between a 4-bit multiplier x4x3x2x1 and a 4-bit multiplicand y4y3y2y1. (3 points)*
b) *Draw the circuit for a 4-bit Array Multiplier and describe how it works. (10 points)*
c) *Show the critical path of the 4-bit Array Multiplier? (3 points)*
d) *Why is it faster than a serial multiplier? (4 points)*

Solution:

a) Lecture 11, slide 15
b) Lecture 11, slide 18
c) Lecture 11, slide 18

(Above mentioned lecture slides can be found at the end of this document)

d)

A serial multiplier checks the LSbit of he multiplier and accordingly adds the multiplicand, subsequently it shifts the multiplier. In total a multiplication of N-bits numbers needs N additions and N shifts of N-bit numbers.

Therefore, the serial multiplier would have a delay of:

N*(delay of N-bit addition + N-bit shift) >= N*(delay of N-bit addition) which is
>= N*logN* (FA delay) , assuming a CLA or
= N*N*(FA delay), assuming a ripple carry adder

On the other hand the delay of the array multiplier is about 2*N*(FA delay) and consequently it is faster.

Question 4: (15 points)

Consider a sequence detector circuit, which detects the sequence "01". It has one input W and one output Z (each of 1-bit). The output Z is equal to 1 if in two consecutive (immediately preceding) clock cycles the input was equal to 0 and then 1; otherwise the value of Z is equal to 0. All changes in the circuit occur on the positive edge of a clock signal.

e.g.

Clock cycle:	*T0*	*T1*	*T2*	*T3*	*T4*	*T5*	*T6*	*T7*	*T8*	*T9*
W:	*0*	*1*	*0*	*1*	*1*	*0*	*0*	*0*	*1*	*1*
Z:	*0*	*0*	*1*	*0*	*1*	*0*	*0*	*0*	*0*	*1*

a) *Draw the state diagram of the circuit. (2 point)*
b) *Fill in the state table of the circuit and make a state assignment. (5 points)*
c) *Draw the actual implementation of the circuit after extracting the Boolean functions of each D-flip-flop. (5 points)*
d) *Did you create a Mealy or a Moore type of circuit? What is the difference between the two types? (2 points)*
e) *What was your choice for the state-assignment? Why? (1 point)*

Solution:

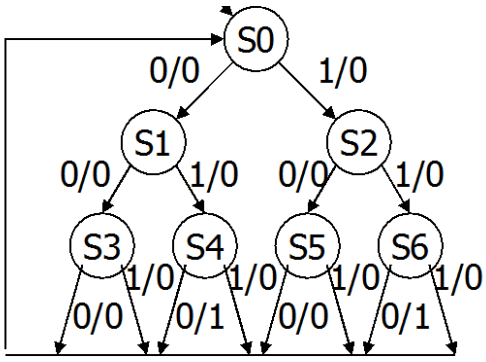
a) similar to Lecture 4, slides 38-40
b) similar to Lecture 4, slides 41-42
c) similar to Lecture 4, slides 44-45

(Above mentioned lecture slides can be found at the end of this document)

d) first part of the answer depends on your implementation. The output of a Moore type FSM depends only on the current state, while the output of a Mealy depends in addition to the input of the FSM.
e) Depends on your state assignment. In general a binary or gray encoding would require less bits than a one-hot encoding, while the later would be faster. In some cases the binary would be slower than the gray encoding, but this depends on the state-diagram, how the transitions between states happen, and how can one "match" the output encoding with the state-assignment.

Question 5: (8 points)

Use partitioning to minimize the number of states in the following Finite State Machine (FSM).



a) *First create the state table of the FSM. (1 points)*
b) *Minimize the number of state using partitioning. (5 points)*
c) *Create the state table of the minimized version of the FSM. (1point)*

d) Draw the state diagram of the minimized version of the FSM. (1point)

Solution:

a)

Input Sequence	Present State	Next State X=0	Next State X=1	Output X=0	Output X=1
Reset	S0	S1	S2	0	0
0	S1	S3	S4	0	0
1	S2	S5	S6	0	0
00	S3	S0	S0	0	0
01	S4	S0	S0	1	0
10	S5	S0	S0	0	0
11	S6	S0	S0	1	0

b)

P1= (S0 S1 S2 S3 S4 S5 S6)

P2= (S0 S1 S2 S3 S5) (S4 S6)

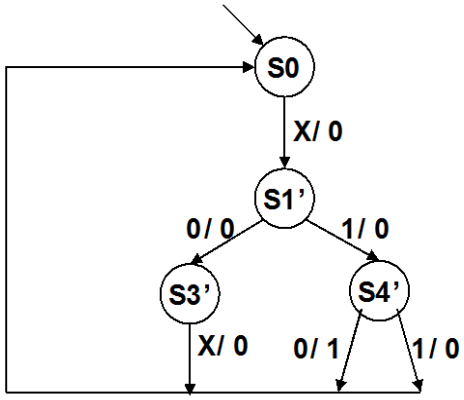
P3= (S0 S3 S5) (S1 S2) (S4 S6)

P4= (S0) (S3 S5) (S1 S2) (S4 S6)

c)

Input Sequence	Present State	Next State X= 0	Next State X= 1	Output X= 0	Output X= 1
Reset	S0	S1'	S1'	0	0
0 + 1	S1'	S3'	S4'	0	0
X0	S3'	S0	S0	0	0
X1	S4'	S0	S0	1	0

d)



Question 6: (9 points)

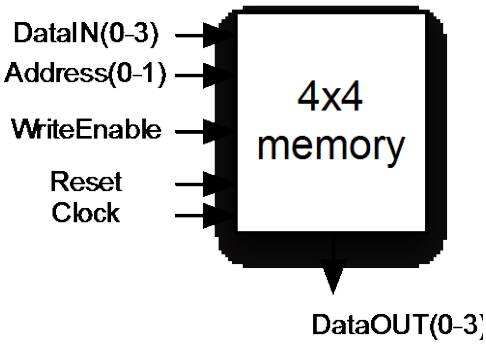
(a) (6 points)

Draw the block diagram of a 4x4 memory, using as "building blocks" D flip-flops and logic gates. The memory has the following inputs:

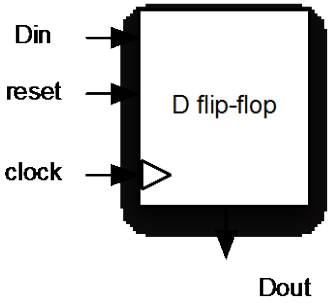
- *DataIN(0), DataIN(1), DataIN(2), DataIN(3),*
- *Write-enable,*
- *Address(0), Address(1)*
- *Reset, Clock*

And the following outputs:

- *DataOUT(0), DataOUT(1), DataOUT(2), DataOUT(3),*



The D flip-flop has the following interface:



(b) (3 points)

Taking the above 4x4 memory as a black box (not having the ability to change its internal design) what can we add to use it as a 16x1 memory?

Solution:

a) *lecture 11, slide 34 (but for 4x4 instead of 4x3)*
b) *a column multiplexer as shown in lecture 11, slide 39*

(Above mentioned lecture slides can be found at the end of this document)

Question 7: (5 points)

Explain the advantages and disadvantages of the following computing alternatives:

1. *ASICs (Application Specific Integrated Circuits)*
2. *Field Programmable gate Arrays (FPGAs)*
3. *General Purpose processors (running software)*

Solution:

Lecture 12 Slide3

(Above mentioned lecture slides can be found at the end of this document)

Question 8: (12 points)

Explain the timing constraints of a D flip-flop.

a) *What is the propagation time, setup time and hold time? (6 points)*
b) *Make a timing diagram to show the above time constraints in a D flip-flop. (3 points)*
c) *Explain when a flip-flop may enter a metastable state. (3 points)*

Solution:

a) *In lecture 16 slide 18*
b) *something similar to lecture 16 slide 19*
c) *In lecture 16 slide 31*

(Above mentioned lecture slides can be found at the end of this document)

Question 9: (6 points)

a) *Explain the difference between permanent, transient and intermittent faults. (3 points)*
b) *Name at least 2 causes for each type of faults. (3 points)*

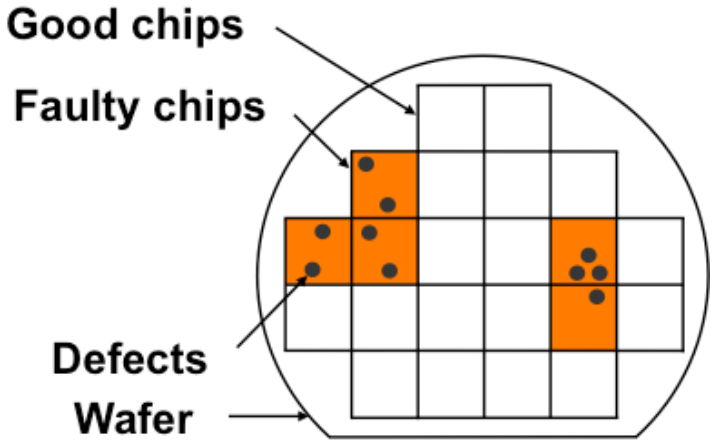
Solution:

In lecture 13 slide 5

(Above mentioned lecture slides can be found at the end of this document)

Question 10: (5 points)

What is the yield for the following wafer:



Solution:

Lecture 13 slide 10

(Above mentioned lecture slides can be found at the end of this document)

Lecture Slides

Question 3:

Adding Partial Products

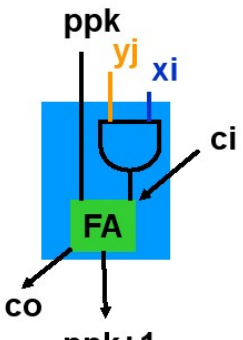
multiplicand $1\,0\,0\,0_{two} = 8_{ten}$
multiplier $1\,0\,0\,1_{two} = 9_{ten}$

```
        1 0 0 0
      0 0 0 0
    0 0 0 0
  1 0 0 0
-----------------
1 0 0 1 0 0 0two = 72ten
```

y3	y2	y1	y0	multiplicand
x3	x2	x1	x0	multiplier
x0y3	x0y2	x0y1	x0y0	four
carry←x1y3	x1y2	x1y1	x1y0	partial
carry←x2y3	x2y2	x2y1	x2y0	products
carry← x3y3	x3y2	x3y1	x3y0	to be summed
p7	p6	p5	p4	p3 p2 p1 p0

Requires three 4-bit additions. Slow.

Array Multiplier



Question 4:

Manual design steps (1): specifications

Specifications:

1. the circuit has one input, w, and one output, z.
2. all changes in the circuit occur on the positive edge of a clock signal.
3. the output z is equal to 1 if during two immediately preceding clock cycles the input w was equal to 1. otherwise, the value of z is equal to 0.

Clockcycle:	t_0	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8	t_9	t_{10}
w:	0	1	0	1	1	0	1	1	1	0	1
z:	0	0	0	0	0	1	0	0	1	1	0

Manual design steps (2): state diagram

- *starting* state A: when power is on or *reset* signal is applied.
- As long as *w* is 0, it remains in A.
- After *w* → *1*, it moves to state B.
- Then,
  - If *w* →*0*, it moves back to state A.
  - If *w* →*1*, it moves to state C, and z=1.
- When in state C,
  - If *w* → *0*, back to state A, and z=0;
  - If *w* →*1*, remain in state C.

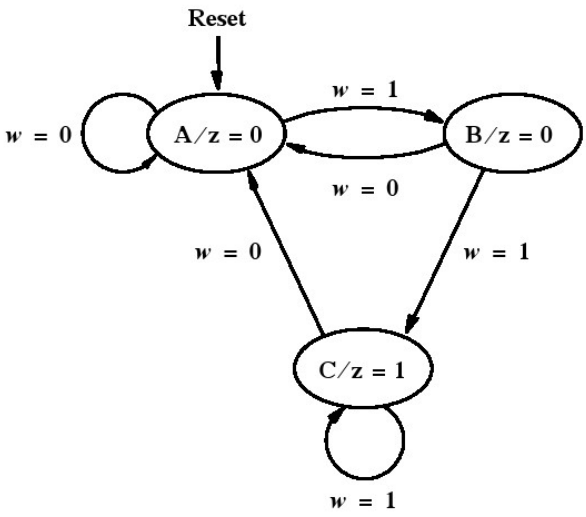
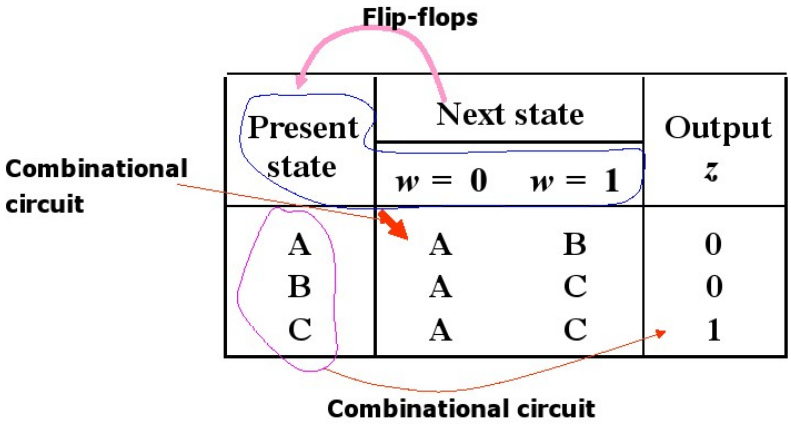


Figure 8.3. State diagram of a simple sequential circuit.

Manual design steps (3): state table

From the state diagram, we have the state table



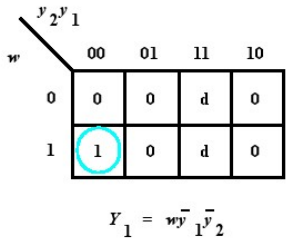
Manual design steps (4): state assignment

A: 00
B: 01
C: 10

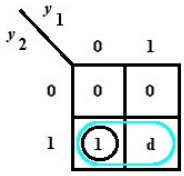
	Present state y_2y_1	Next state $w = 0$ Y_2Y_1	Next state $w = 1$ Y_2Y_1	Output z
A	00	00	01	0
B	01	00	10	0
C	10	00	10	1
	11	*dd*	*dd*	*d*

Manual design steps (5): implementation

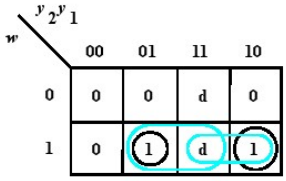
From state-assigned table: we have the following Karnaugh maps



$Y_1 = w\bar{y}_1\bar{y}_2$

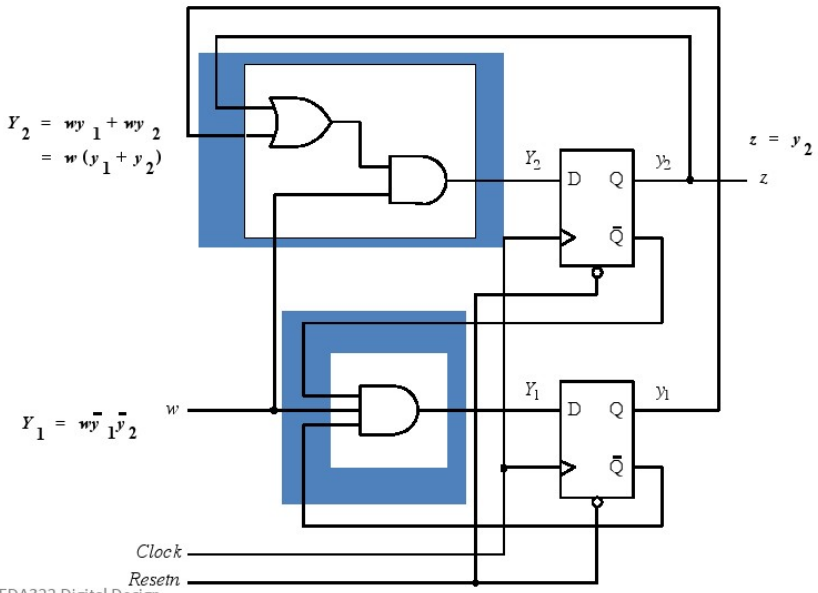


$z = y_2$



$Y_2 = wy_1 + wy_2$
$= w(y_1 + y_2)$

Final implementation



Question 6:

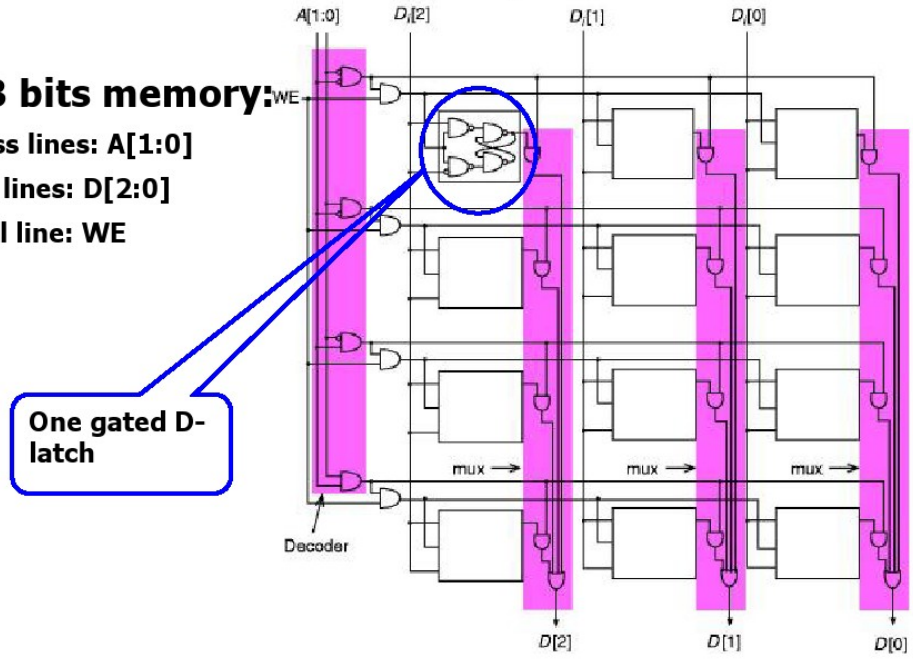
Memory Example

A 2^2 by 3 bits memory:

two address lines: A[1:0]

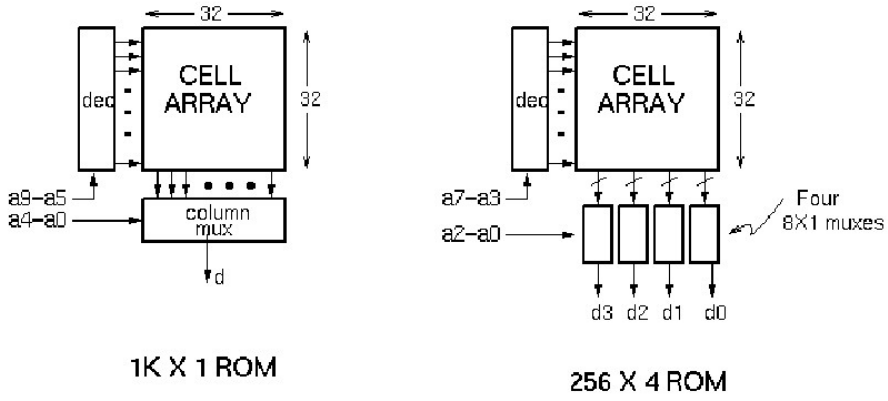
three data lines: D[2:0]

one control line: WE



Column MUX in ROMs and RAMs:

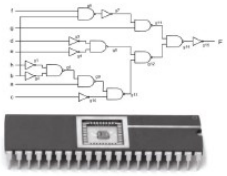
- Controls physical aspect ratio
- In DRAM, allows reuse of chip address pins



Question 7:

Computing alternatives

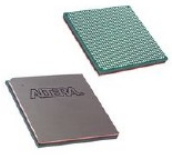
Hardware (Application Specific Integrated Circuits)



Advantages:
- very high performance and efficient

Disadvantages:
- not flexible (can't be altered after fabrication)
- High NRE Cost

Reconfigurable computing



Advantages:
- much higher performance than software / lower performance than ASIC
- higher level of flexibility than hardware / more difficult to program than SW

- **fills the gap between hardware and software**

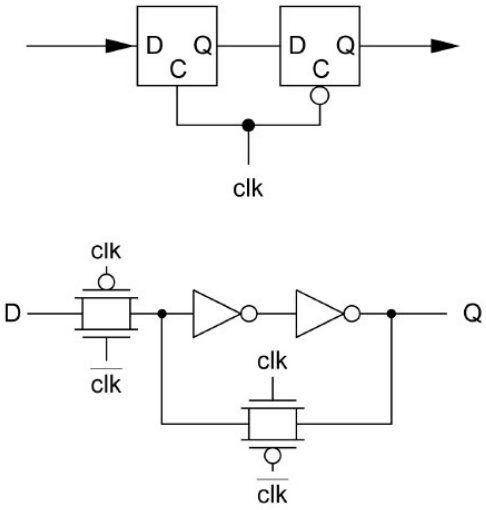
Software-programmed processors

Advantages:
- software is very flexible to change

Disadvantages:
- performance can suffer if clock is not fast
- fixed instruction set by hardware

Question 8:

Delay in Flip-flops



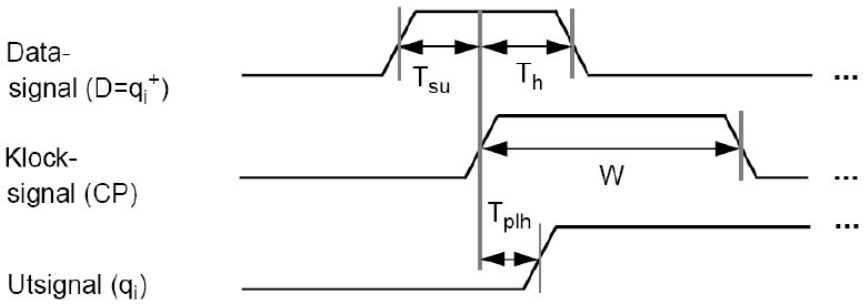
T_{su}= setup time
(the time the input of a flip flop needs to be stable before the clock edge to be stored correctly)

T_h= hold time
(the time the input of a flip flop needs to be held after the clock edge to be stored correctly)

T_p=propagation time (clock to Q)
(the time a flip-flop needs to propagate a value stored to the output)

EDA322 Digital design, 2015-2016, Lecture 16 I. Sourdis, CSE, Chalmers Page 18

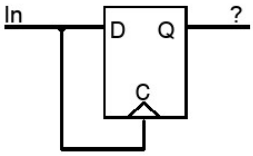
Time parameters for clocked memory elements



EDA322 Digital design, 2015-2016, Lecture 16 I. Sourdis, CSE, Chalmers 19

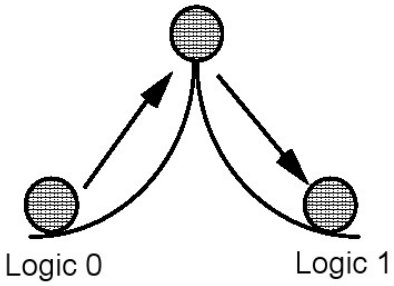
Metastability and Asynchronous Inputs

Synchronizer Failure

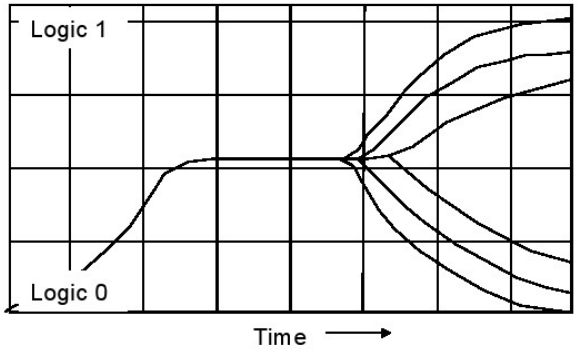


When FF input changes close to clock edge, the FF may enter the *metastable* state: neither a logic 0 nor a logic 1

It may stay in this state an indefinate amount of time, although this is not likely in real circuits



Small, but non-zero probability that the FF output will get stuck in an in-between state



Oscilloscope Traces Demonstrating Synchronizer Failure and Eventual Decay to Steady State

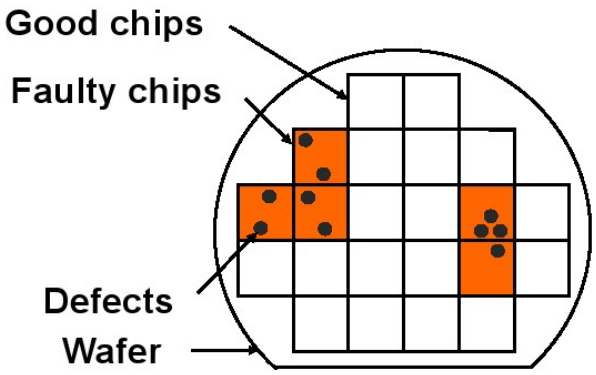
Question 9:

Types of Faults

- Transient faults
  - Faults that happen only once
    - and it's VERY unlikely to happen again
  - Causes:
    - Electromagnetic Interference
      - Neighbors mobile phone
      - Static electricity
    - Various particles hitting the silicon surface
      - Heavy ions such as iron, α-particles, neutrons.
    - Internal effects
      - Crosstalk, metastability, power supply disturbances
- Permanent faults
  - Faults that are always there
  - Causes:
    - Design defects
    - Manufacturing defects
    - Transistor aging
- Intermittent faults
  - Faults that come and go (probably periodically)
  - Causes: Variations
    - **Static**: transistors on a chip may not be exactly the same although they were supposed to be
    - **Dynamic**: temperature changes

Question 10:

Circuit Fabrication and Defects



Defects are permanent faults caused during the Manufacturing process

$$\text{yield} = \frac{\text{number of working chips produced}}{\text{Total number of chips produced}}$$

Wafer yield = 17/22 = 0.77

EDA322 Digital Design, 2015-2016, Lecture 13 I. Sourdis, CSE, Chalmers 10