

CS M51A

Logic Design of Digital Systems

Winter 2021

Some slides borrowed and modified from:

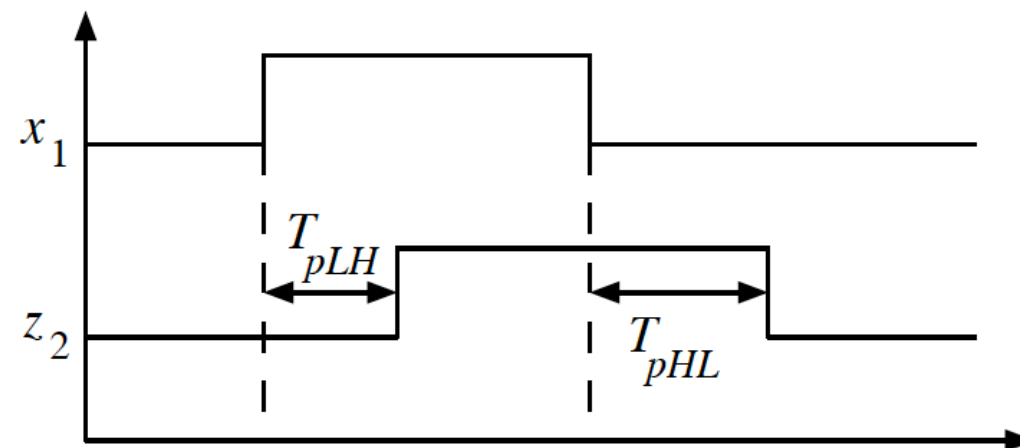
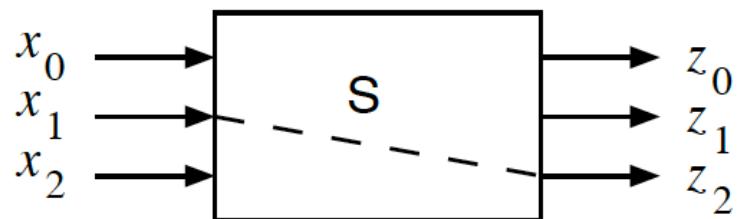
M.D. Ercegovac, T. Lang and J. Moreno, Introduction to Digital Systems.

D. Patterson and J. Hennessy, Computer Organization and Design

Review

- Switching Expression and Boolean Algebra
- Truth table
- Sum of Minterms and Product of Maxterms
- Symbols and Gate level design
- Transistors and gate implementation

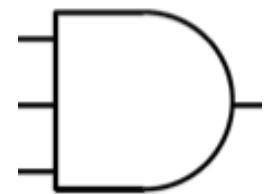
Timing Analysis



Gate type	Fan-in	Propagation delays	
		t_{pLH} [ns]	t_{pHL} [ns]
AND	2	$0.15 + 0.037L$	$0.16 + 0.017L$
AND	3	$0.20 + 0.038L$	$0.18 + 0.018L$
OR	2	$0.12 + 0.037L$	$0.20 + 0.019L$
OR	3	$0.12 + 0.038L$	$0.34 + 0.022L$
NOT	1	$0.02 + 0.038L$	$0.05 + 0.017L$

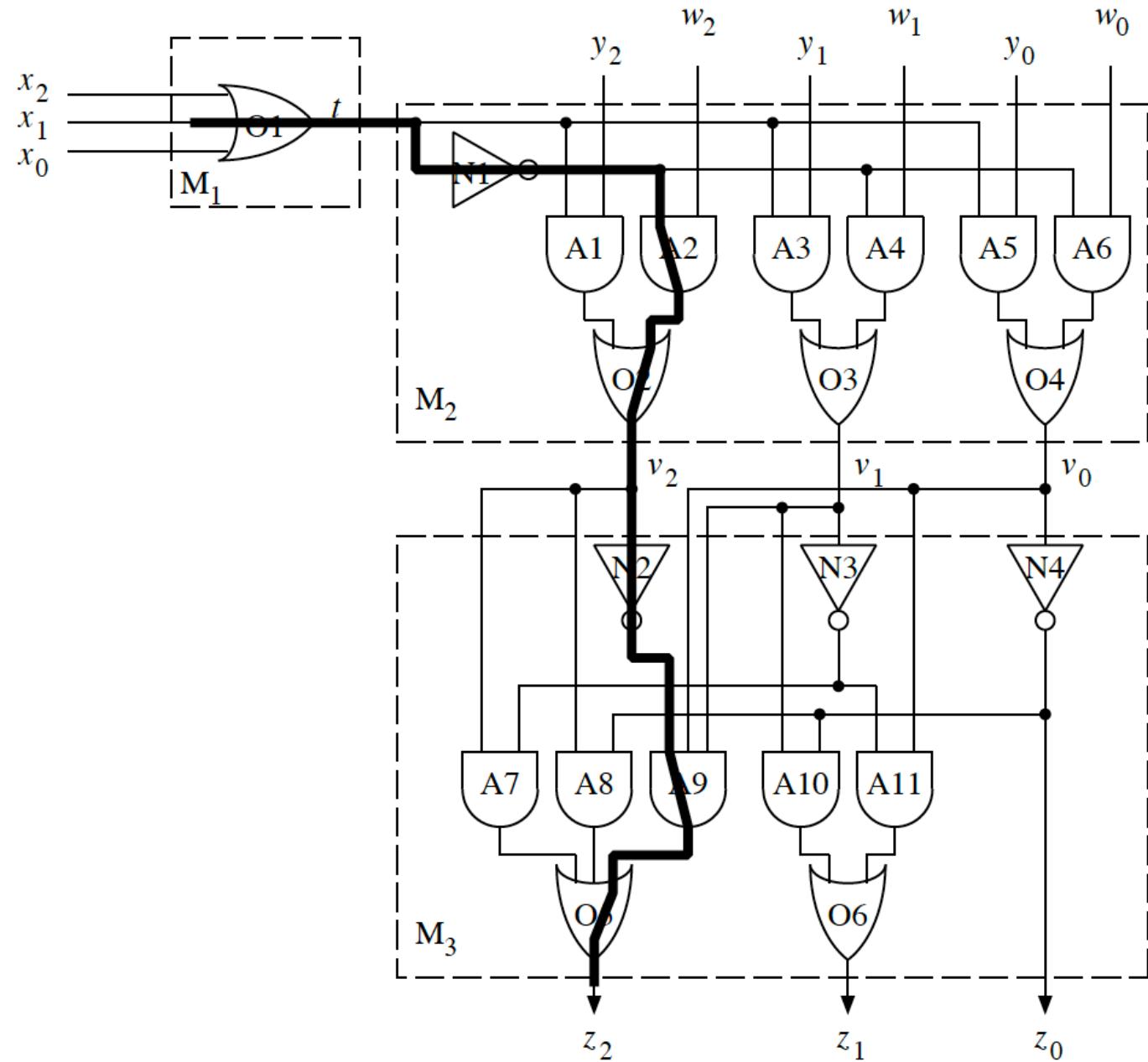
Clicker Question

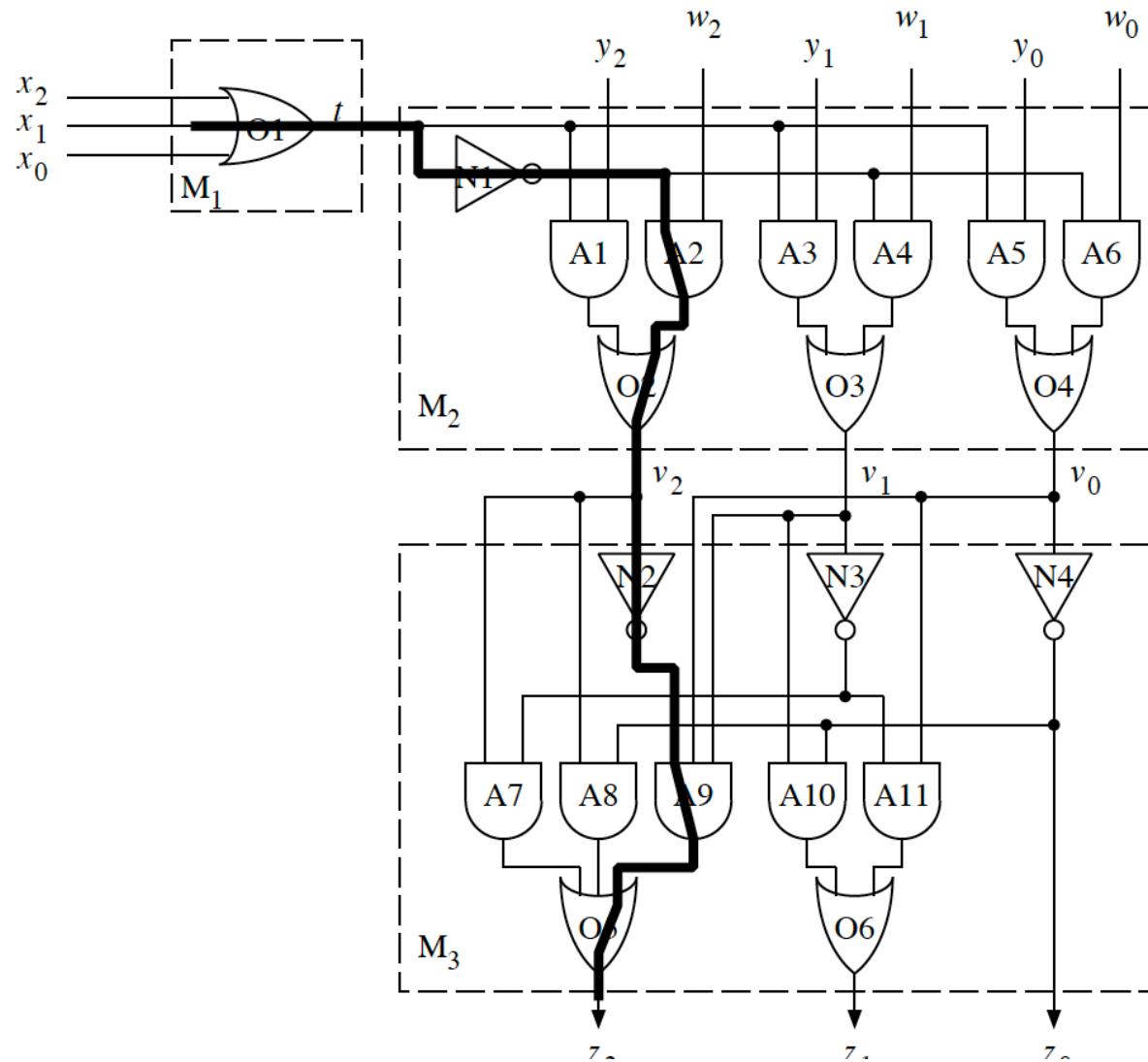
Using the gate characteristics given below,
determine equation for the propagation delay
 t_{pHL} for this circuit



- a) 0.20+0.038
- b) 0.18+0.018
- c) 0.12+0.038
- d) 0.34+0.022
- e) Not clear

Gate type	Fan-in	Propagation delays	
		t_{pLH} [ns]	t_{pHL} [ns]
AND	2	$0.15 + 0.037L$	$0.16 + 0.017L$
AND	3	$0.20 + 0.038L$	$0.18 + 0.018L$
OR	2	$0.12 + 0.037L$	$0.20 + 0.019L$
OR	3	$0.12 + 0.038L$	$0.34 + 0.022L$
NOT	1	$0.02 + 0.038L$	$0.05 + 0.017L$





NETWORK DELAY Example of path delay calculation:

$$O_1 \rightarrow N_1 \rightarrow A_2 \rightarrow O_2 \rightarrow N_2 \rightarrow A_9 \rightarrow O_5$$

NETWORK DELAY Example of path delay calculation:

$$O_1 \rightarrow N_1 \rightarrow A_2 \rightarrow O_2 \rightarrow N_2 \rightarrow A_9 \rightarrow O_5$$

$$\begin{aligned} T_{pLH}(x_1, z_2) &= t_{pLH}(O_1) + t_{pHL}(N_1) + t_{pHL}(A_2) + t_{pHL}(O_2) \\ &\quad + t_{pLH}(N_2) + t_{pLH}(A_9) + t_{pLH}(O_5) \end{aligned}$$

$$\begin{aligned} T_{pHL}(x_1, z_2) &= t_{pHL}(O_1) + t_{pLH}(N_1) + t_{pLH}(A_2) + t_{pLH}(O_2) \\ &\quad + t_{pHL}(N_2) + t_{pHL}(A_9) + t_{pHL}(O_5) \end{aligned}$$

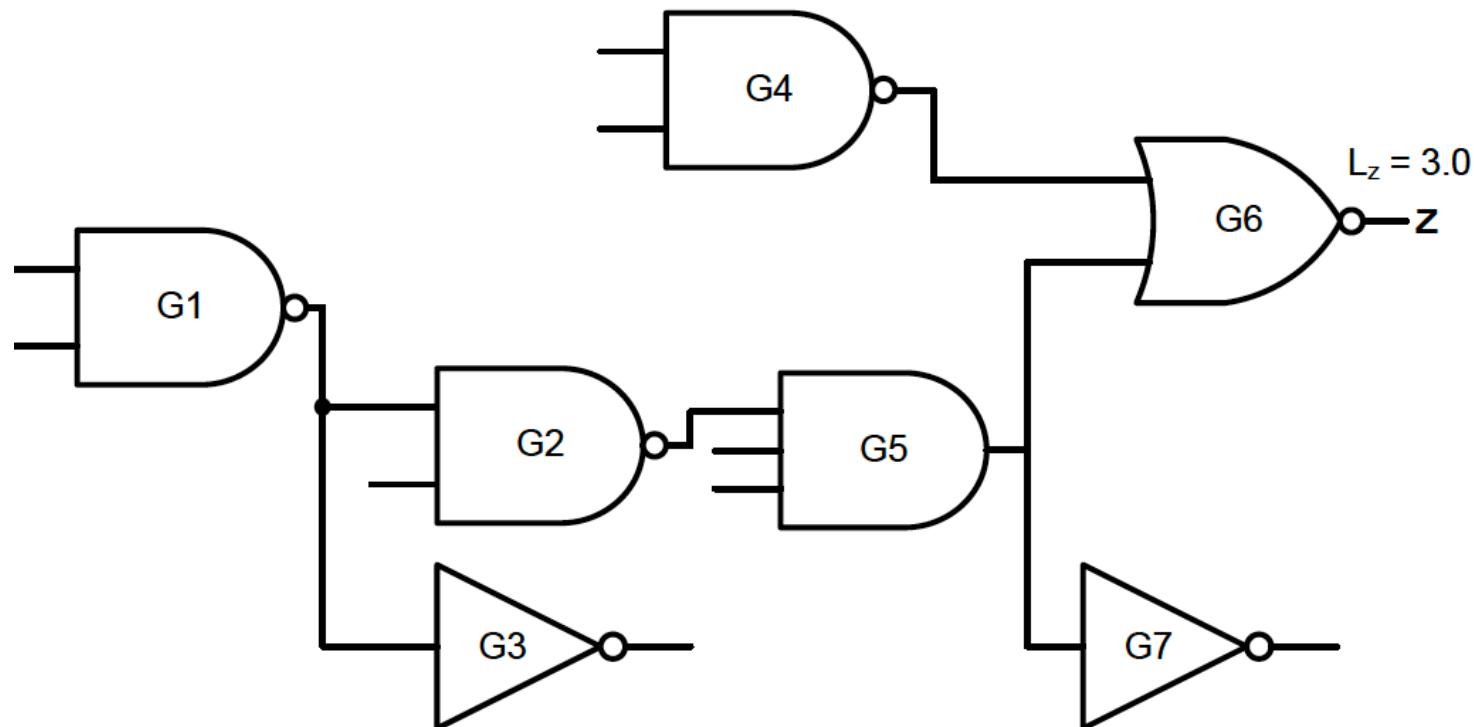
Gate	Identifier	Output load	t_{pLH} [ns]	t_{pHL} [ns]
OR3	O_1	4	0.27	0.43
NOT	N_1	3	0.13	0.10
AND2	A_2	1	0.19	0.18
OR2	O_2	3	0.23	0.26
NOT	N_2	1	0.06	0.07
AND3	A_9	1	0.24	0.20
OR3	O_5	L	$0.12 + 0.038L$	$0.34 + 0.022L$

$$\begin{aligned} T_{pLH}(x_1, z_2) &= 0.27 + 0.10 + 0.18 + 0.26 + 0.06 \\ &\quad + 0.24 + 0.12 + 0.038L = 1.23 + 0.038L \text{ [ns]} \end{aligned}$$

$$\begin{aligned} T_{pHL}(x_1, z_2) &= 0.43 + 0.13 + 0.19 + 0.23 + 0.07 \\ &\quad + 0.20 + 0.34 + 0.022L = 1.59 + 0.022L \text{ [ns]} \end{aligned}$$

Clicker Question

What is the path with worst case delay?



- a) $G1 \rightarrow G3$
- b) $G1 \rightarrow G2 \rightarrow G5 \rightarrow G7$
- c) $G4 \rightarrow G6$
- d) $G1 \rightarrow G2 \rightarrow G5 \rightarrow G6$
- e) $G1 \rightarrow G2 \rightarrow G5 \rightarrow G4 \rightarrow G6$

Gate Type	Fan-in	Propagation Delays (ns)	
		t_{pLH}	t_{pHL}
AND	3	$0.20 + 0.038L$	$0.18 + 0.018L$
NOT	1	$0.02 + 0.038L$	$0.05 + 0.017L$
NAND	2	$0.05 + 0.038L$	$0.08 + 0.027L$
NOR	2	$0.06 + 0.075L$	$0.07 + 0.016L$

Process of Designing a Hardware

Design Implementation

VHDL

```
library IEEE;
use IEEE.STD_Logic_1164.all;

entity LATCH_IF_ELSIF is
    port (En1, En2, En3, A1, A2, A3: in std_logic;
          Y: out std_logic);
end entity LATCH_IF_ELSIF;

architecture RTL of LATCH_IF_ELSIF is
begin
    process(En1, En2, En3, A1, A2, A3)
    begin
        if(En1 = '1') then
            Y <= A1;
        elsif(En2 = '1') then
            Y <= A2;
        elsif(En3 = '1') then
            Y <= A3;
        end if;
    end process;

end architecture RTL;
```

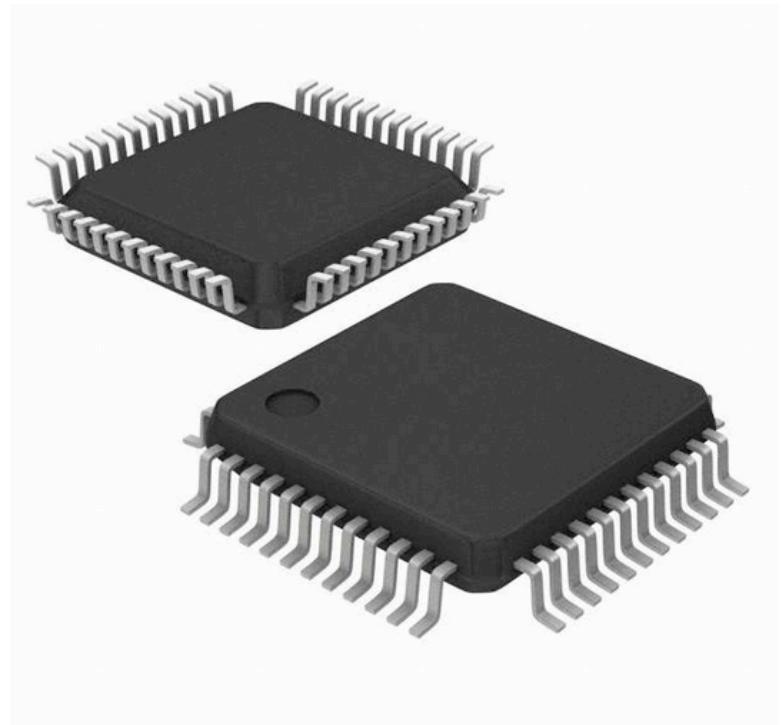
Verilog

```
module LATCH_IF_ELSIF (En1, En2, En3, A1, A2, A3,Y);
    input En1, En2, En3, A1, A2, A3;
    output Y;
    reg Y;

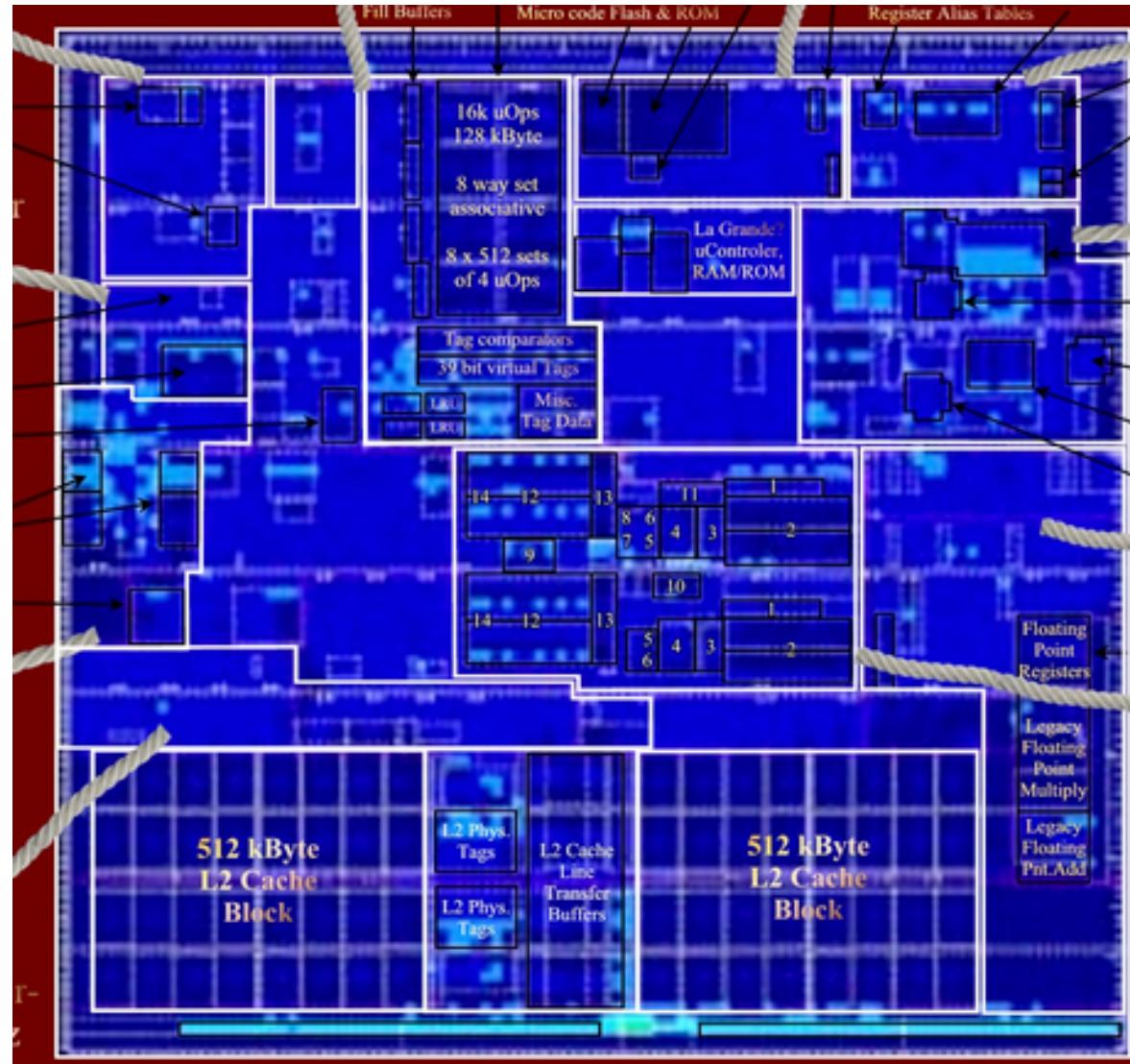
    always @(En1 or En2 or En3 or A1 or A2 or A3)
        if(En1 == 1)
            Y =A1;
        else if (En2 == 1)
            Y = A2;
        else if (En3 == 1)
            Y = A3;

end module
```

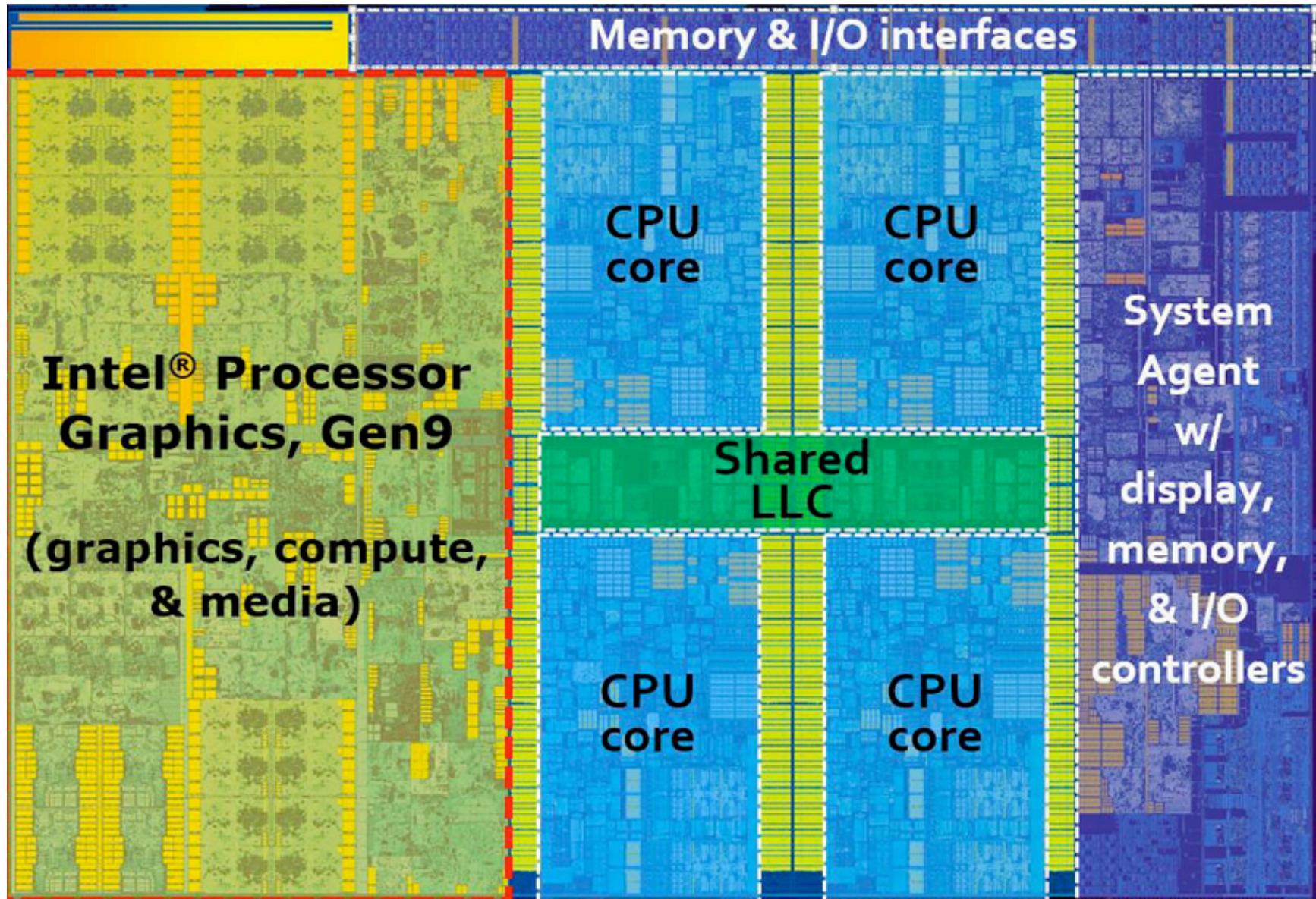
FPGA versus ASIC



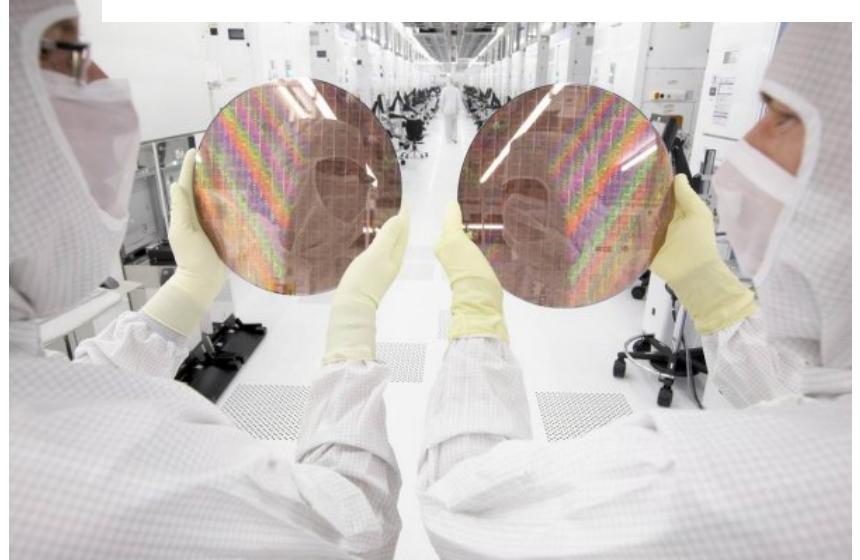
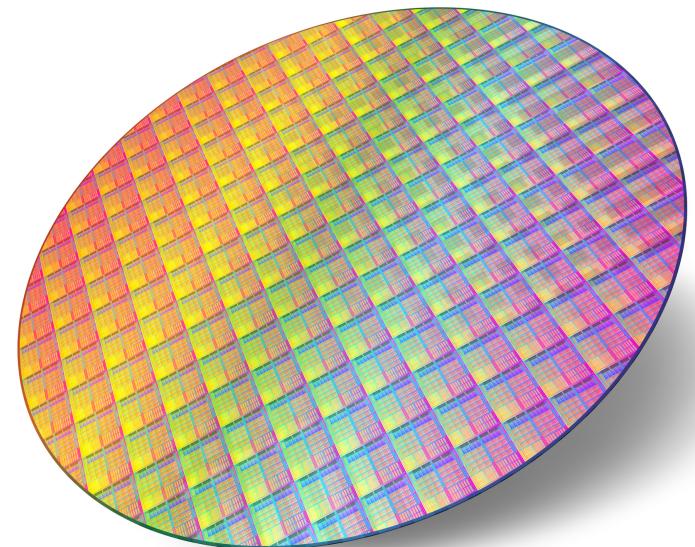
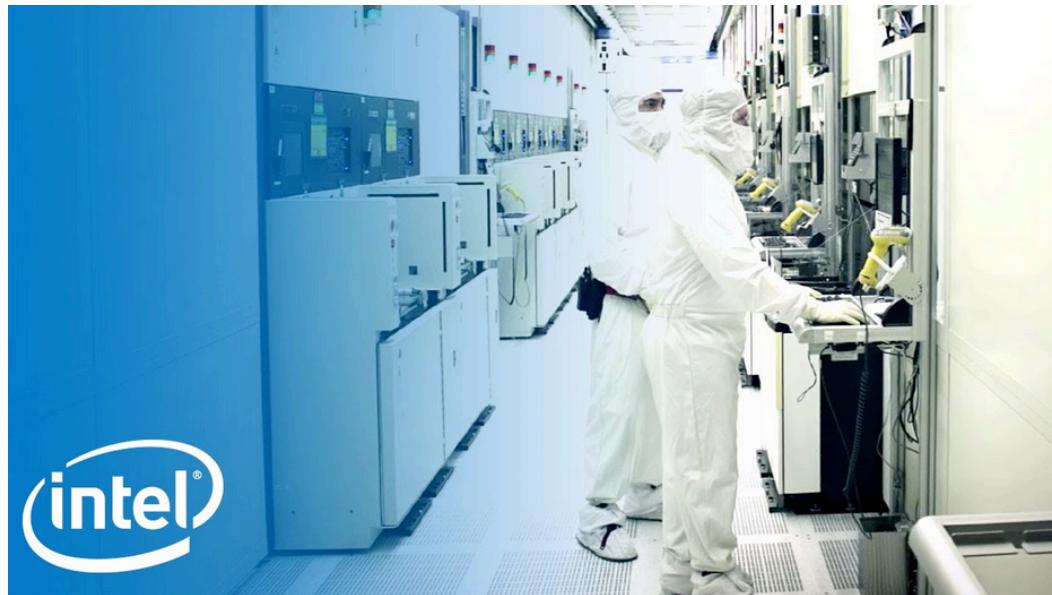
Intel Pentium Chip



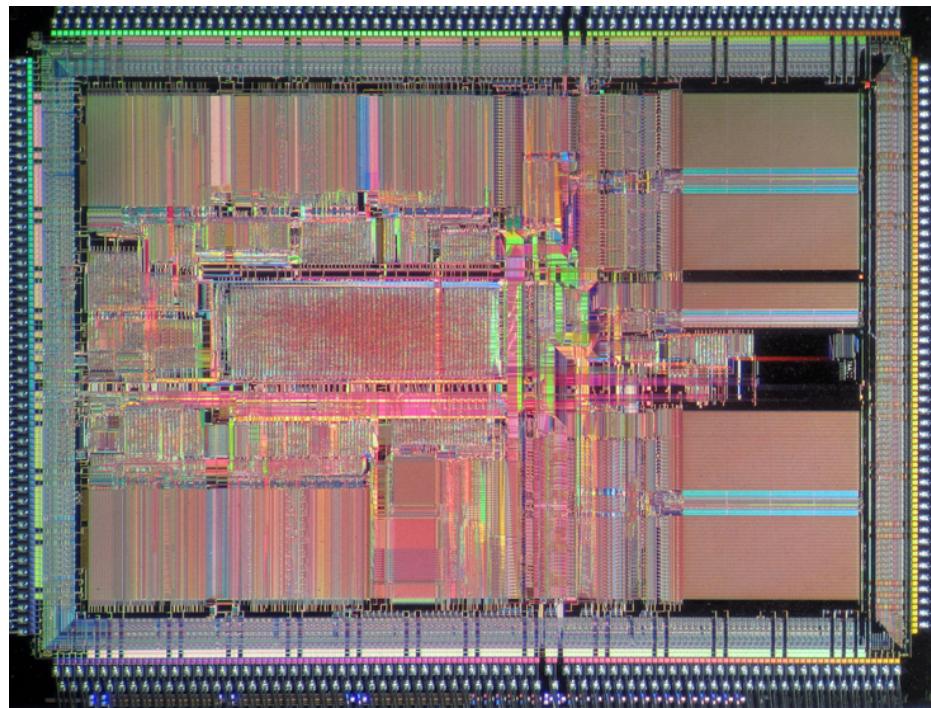
Another CPU



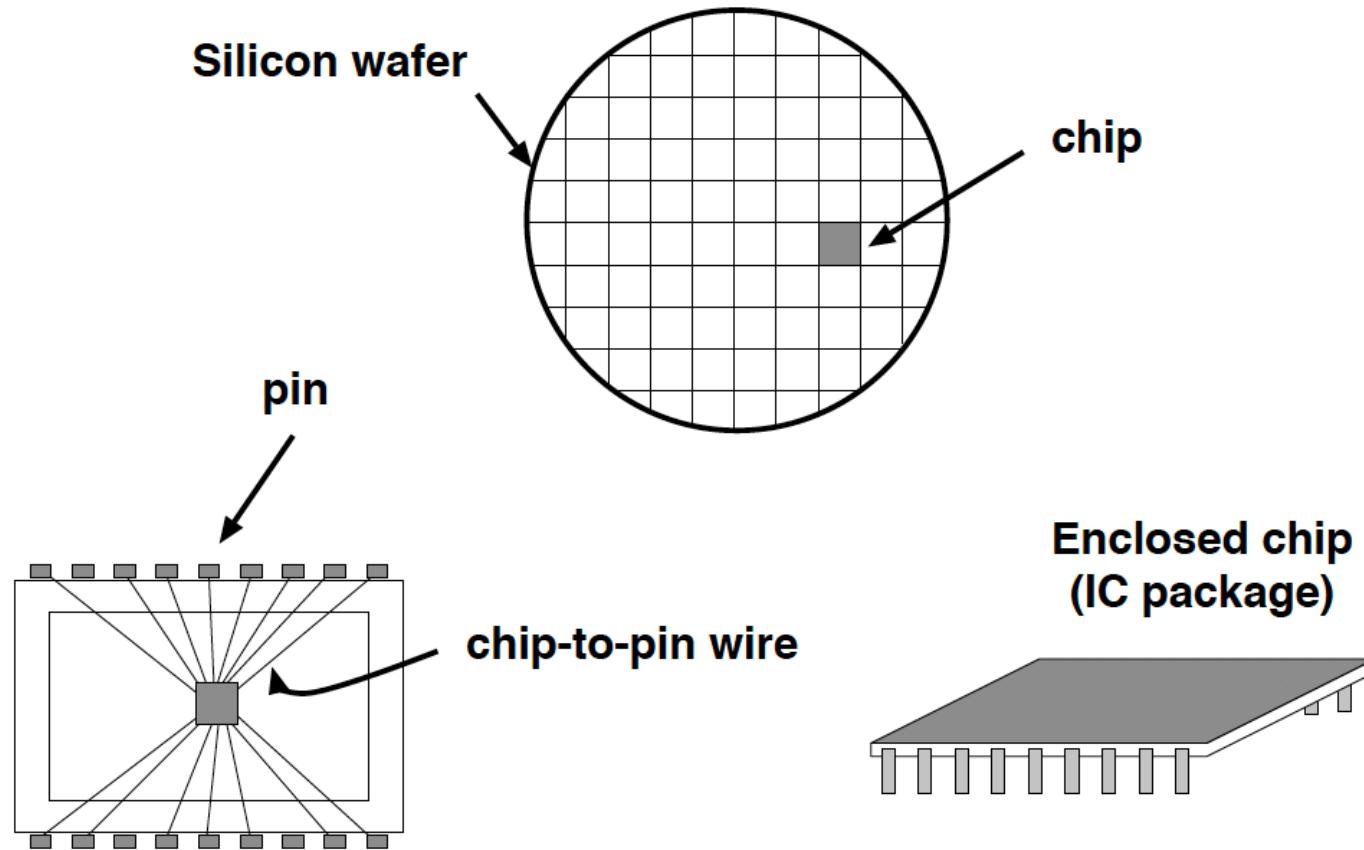
Fabrication



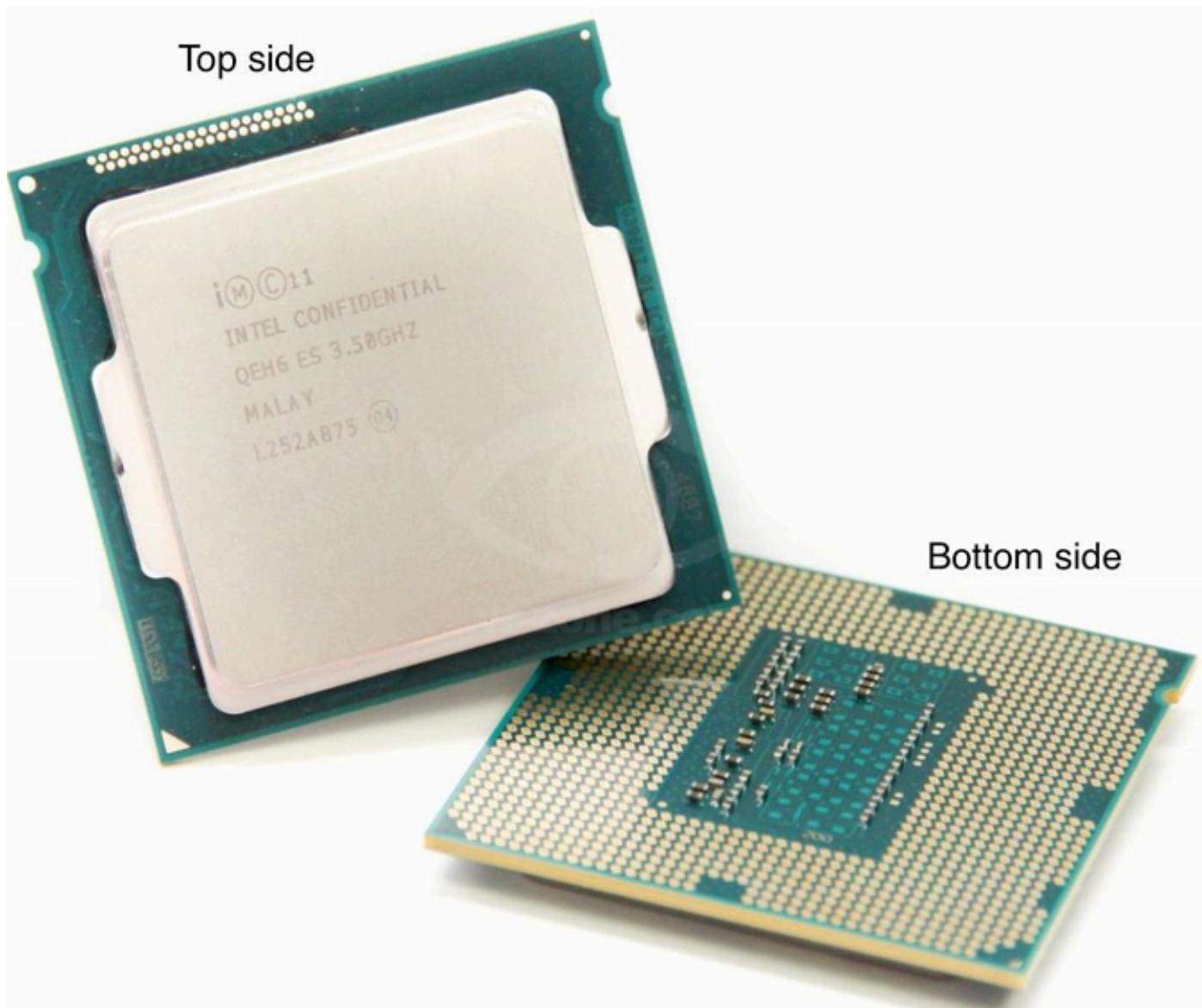
Die



Packaging



Packaging



Board design



Two-Level Systems

TWO-LEVEL NETWORKS

- TWO TYPES OF TWO-LEVEL NETWORKS:

AND-OR NETWORK \Leftrightarrow SUM OF PRODUCTS (NAND-NAND NETWORK)

OR-AND NETWORK \Leftrightarrow PRODUCT OF SUMS (NOR-NOR NETWORK)

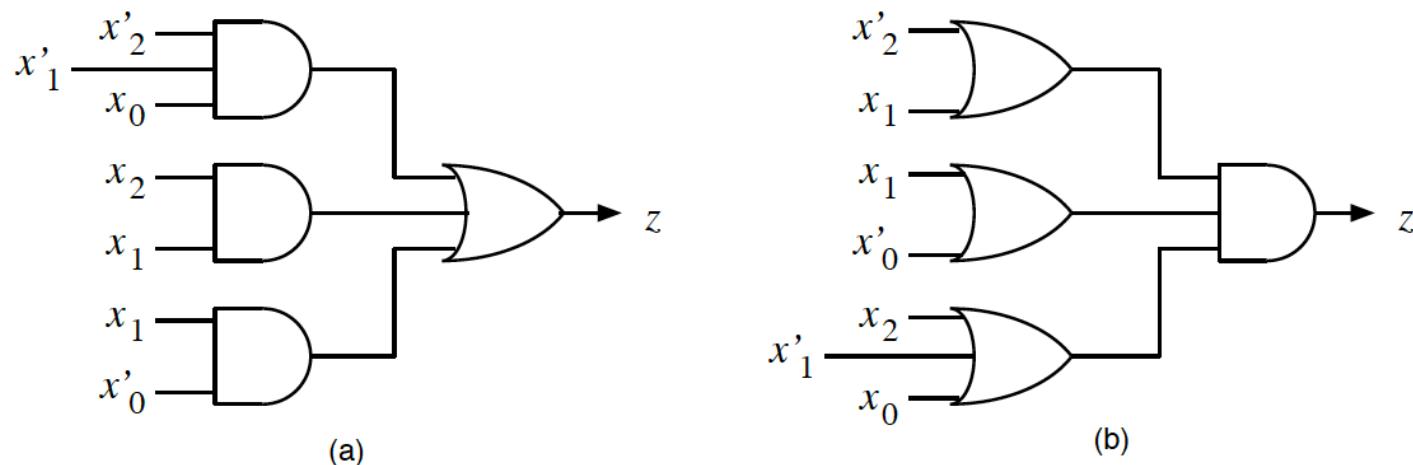


Figure 5.2: AND-OR and OR-AND NETWORKS.

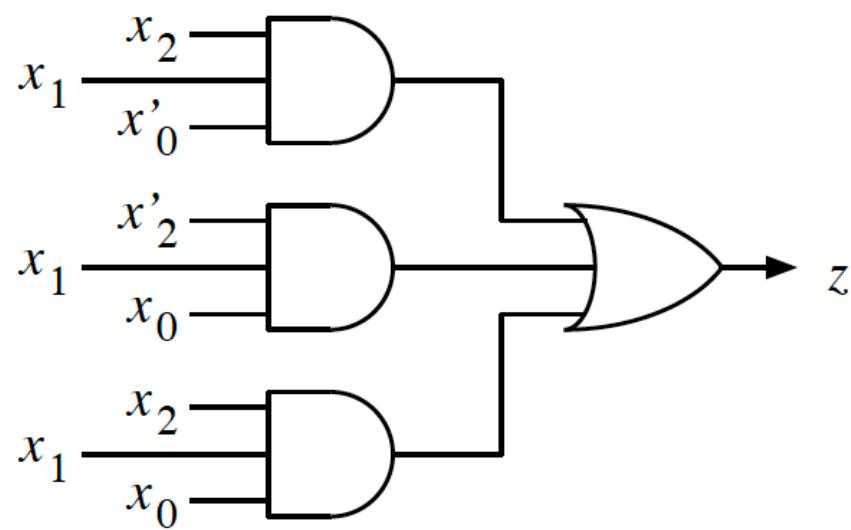
$$E(x_2, x_1, x_0) = x'_2 x'_1 x_0 + x_2 x_1 + x_1 x'_0$$

$$E(x_2, x_1, x_0) = (x'_2 + x_1)(x_1 + x'_0)(x_2 + x'_1 + x_0)$$

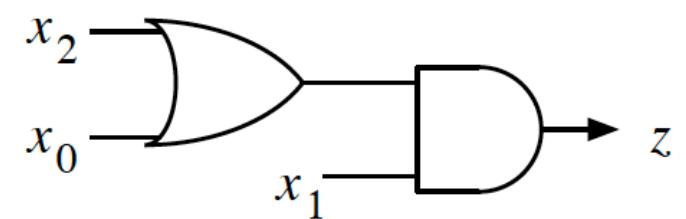
MINIMAL TWO-LEVEL NETWORKS

1. INPUTS: UNCOMPLEMENTED AND COMPLEMENTED
2. FANIN UNLIMITED
3. SINGLE-OUTPUT NETWORKS
4. MINIMAL NETWORK:
MINIMUM NUMBER OF GATES WITH MINIMUM NUMBER OF INPUTS

NETWORKS WITH DIFFERENT COST

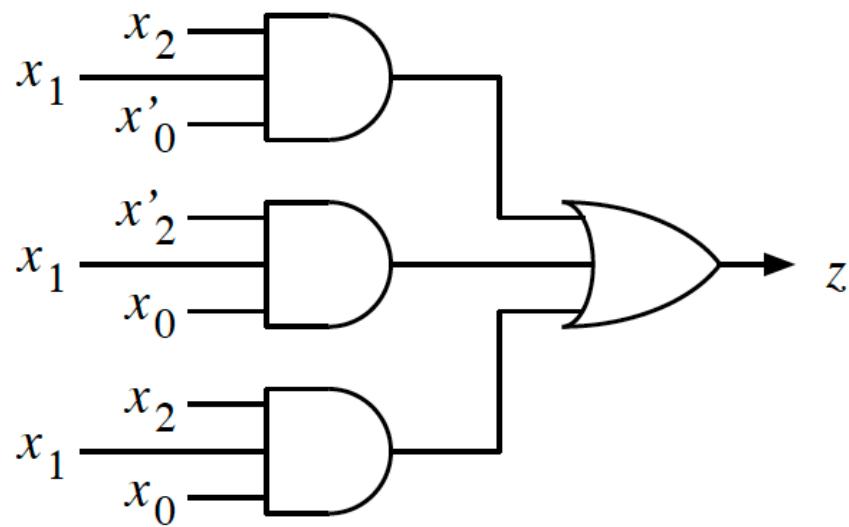


Network A

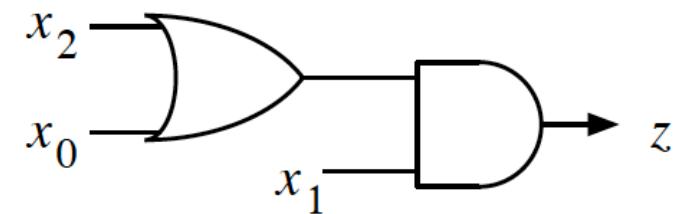


Network B

NETWORKS WITH DIFFERENT COST



Network A



Network B

EQUIVALENT BUT DIFFERENT COST

BOTH MINIMAL SP AND PS MUST BE OBTAINED AND COMPARED