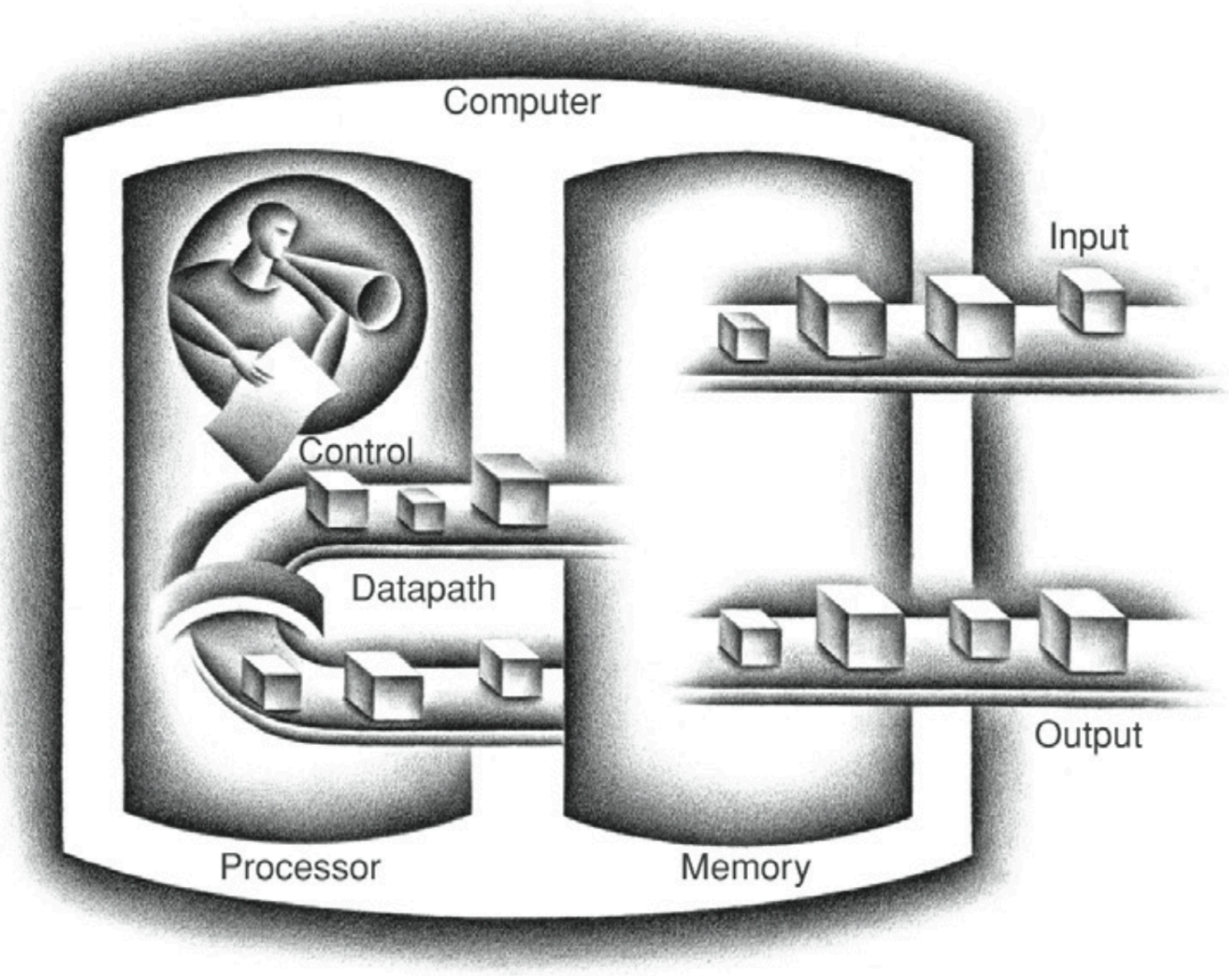
CENG 3420
Computer Organization and Design

Lecture 02: Digital Logic Review

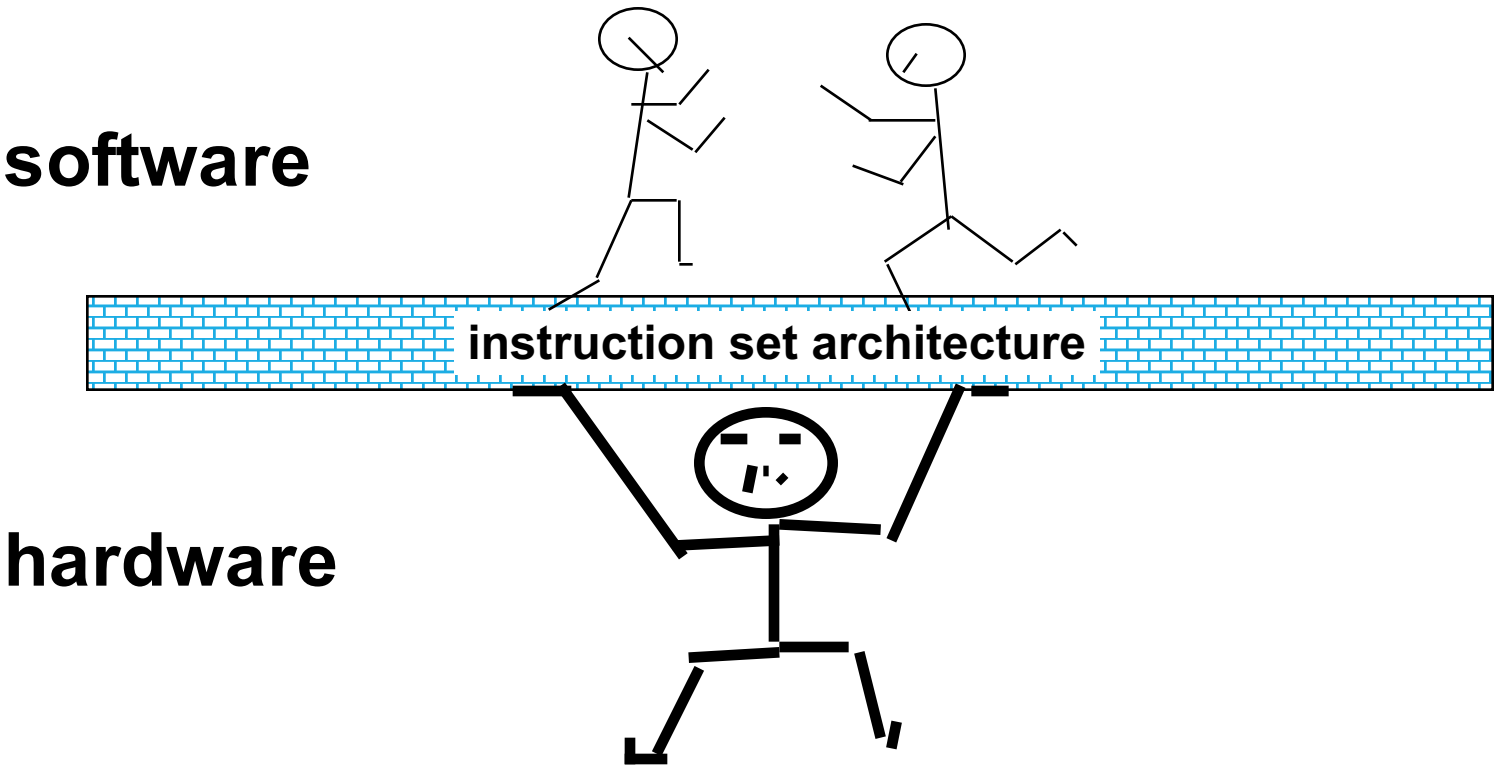
Bei Yu

香港中文大學
The Chinese University of Hong Kong

Review: Major Components of a Computer



Review: The Instruction Set Architecture (ISA)

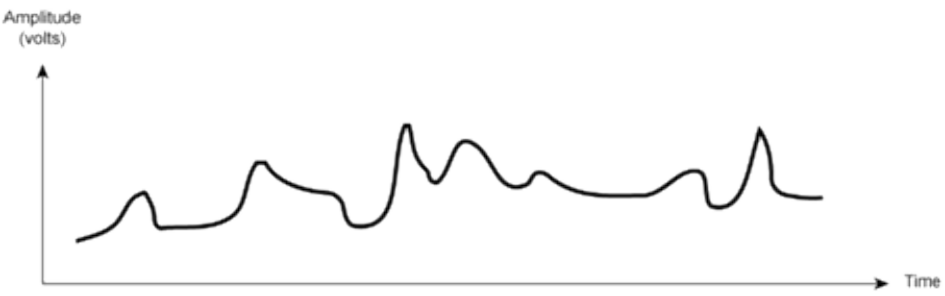


The interface description separating the software and hardware

Analog vs. Digital

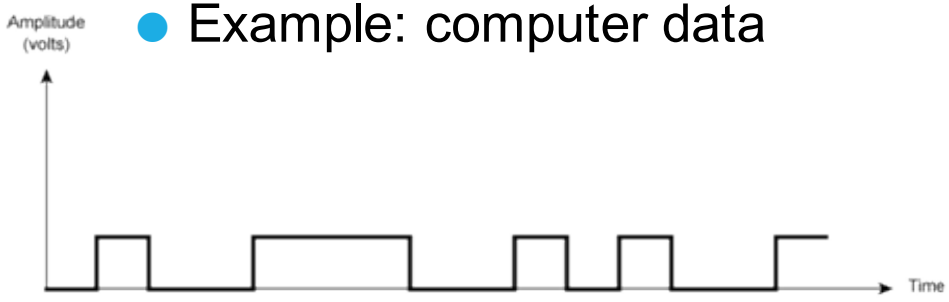
Analog Signal

- Vary in a smooth way over time
- Analog data are continuous valued
  - Example: audio, video



Digital Signal

- Maintains a constant level then changes to another constant level (generally operate in one of the two states)
- Digital data are discrete valued
  - Example: computer data



Number Systems

- An ordered set of symbols, called digits, with relations defined for addition, subtraction, multiplication, and division
- Radix or base of the number system is the total number of digits allowed in the number system
- Commonly used numeral systems

System Name	Decimal	Binary	Octal	Hexadecimal
Radix	10	2	8	16
First seventeen positive integers	0	0	0	0
	1	1	1	1
	2	10	2	2
	3	11	3	3
	4	100	4	4
	5	101	5	5
	6	110	6	6
	7	111	7	7
	8	1000	10	8
	9	1001	11	9
	10	1010	12	A
	11	1011	13	B
	12	1100	14	C
	13	1101	15	D
	14	1110	16	E
	15	1111	17	F
	16	10000	20	10

Conversion from Decimal Integer

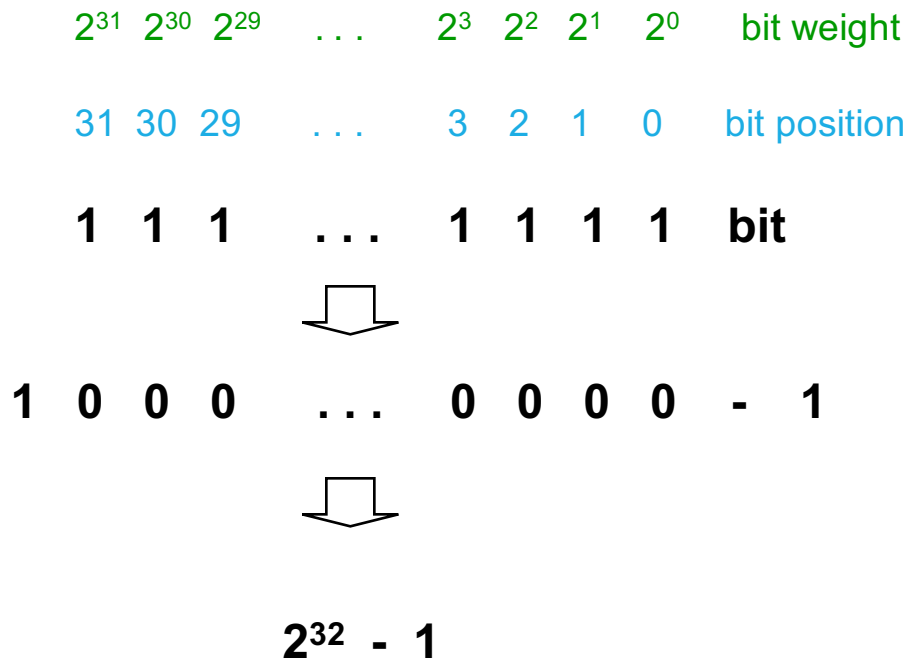
- Step 1: Divide the decimal number by the radix (number base)
- Step 2: Save the remainder (first remainder is the least significant digit)
- Repeat steps 1 and 2 until the quotient is zero
- Result is in reverse order of remainders

EX: L02-1

- EX1: Convert 36_8 to binary value
- EX2: Convert 36_{10} to binary value

Unsigned Binary Representation

Hex	Binary	Decimal
0x00000000	0…0000	0
0x00000001	0…0001	1
0x00000002	0…0010	2
0x00000003	0…0011	3
0x00000004	0…0100	4
0x00000005	0…0101	5
0x00000006	0…0110	6
0x00000007	0…0111	7
0x00000008	0…1000	8
0x00000009	0…1001	9
	…	
0xFFFFFFFC	1…1100	$2^{32} - 4$
0xFFFFFFFD	1…1101	$2^{32} - 3$
0xFFFFFFFE	1…1110	$2^{32} - 2$
0xFFFFFFFF	1…1111	$2^{32} - 1$



Signed Binary Representation

2'sc binary	decimal
1000	-8
1001	-7
1010	-6
1011	-5
1100	-4
1101	-3
1110	-2
1111	-1
0000	0
0001	1
0010	2
0011	3
0100	4
0101	5
0110	6
0111	7

-2^3 =

$-(2^3 - 1)$ =

$2^3 - 1$ =

complement all the bits

0101

and add a 1

0110

1011

and add a 1

1010

complement all the bits

EX: L02-2

- For an n-bit signed binary numeral system, what's the largest positive number and the smallest negative number?

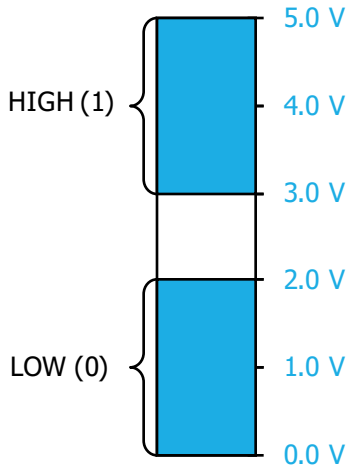
Digital Circuits

- Digital circuits generally contain two parts:
  - Combinational logic
  - Sequential logic
- Combinational circuits consist of logic gates with inputs and outputs
  - The outputs at any instance of time depend only on the combination of the input values based on logic operations such as AND, OR etc.
- Sequential circuits, in addition to inputs and outputs also have storage elements, therefore the output depends on both the current inputs as well as the stored values

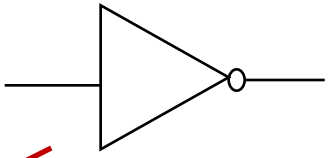
Digital Signal Representation

- Active HIGH
  - High voltage means On
- Active LOW
  - Low voltage means On

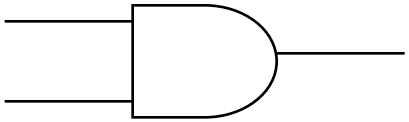
Logic 0	Logic 1
False	True
Off	On
LOW	HIGH
No	Yes
Open switch	Closed switch



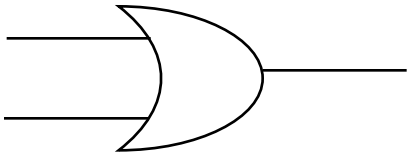
Logic Gates



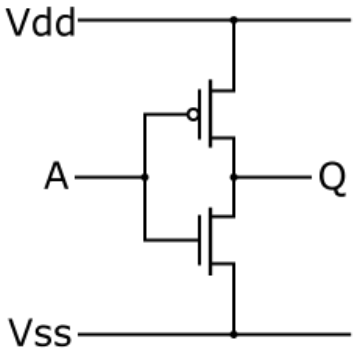
NOT (Invertor)



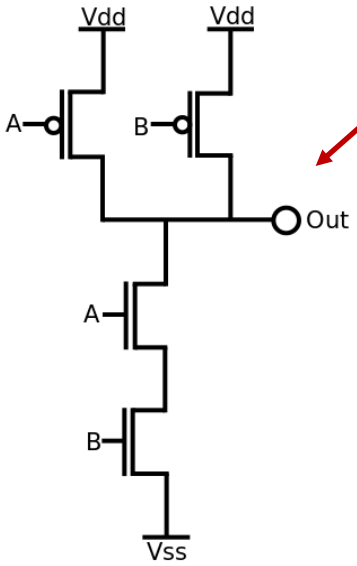
AND

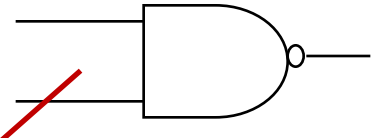


OR

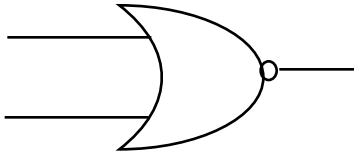


Invertor schematic view

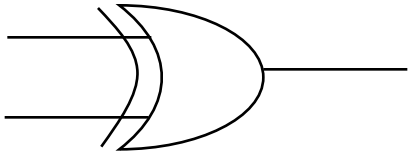




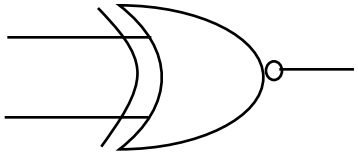
NAND



NOR



XOR



XNOR

- What is the schematic view of an AND gate?

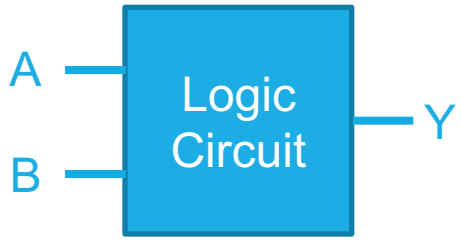
EX: L02-3

- Please draw NOR gate schematic view

Truth Table

- A means for describing how a logic circuit's output depends on the logic levels present at the circuit's inputs

- The number of input combinations will equal 2^N for an N-input truth table

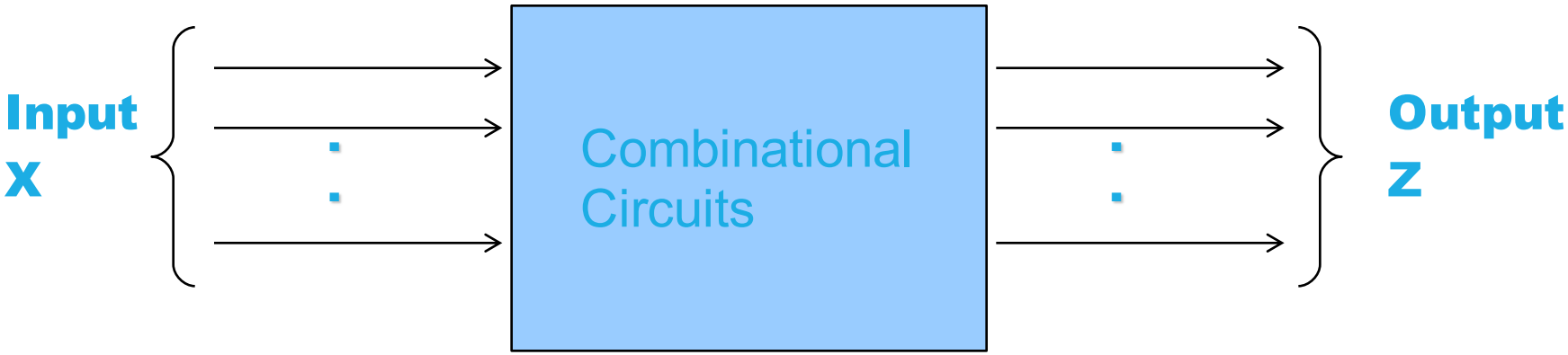


Inputs		Output
A	B	Y
0	0	0
0	1	0
1	0	0
1	1	1

EX: L02-4

Determine the true table of a three-input AND gate

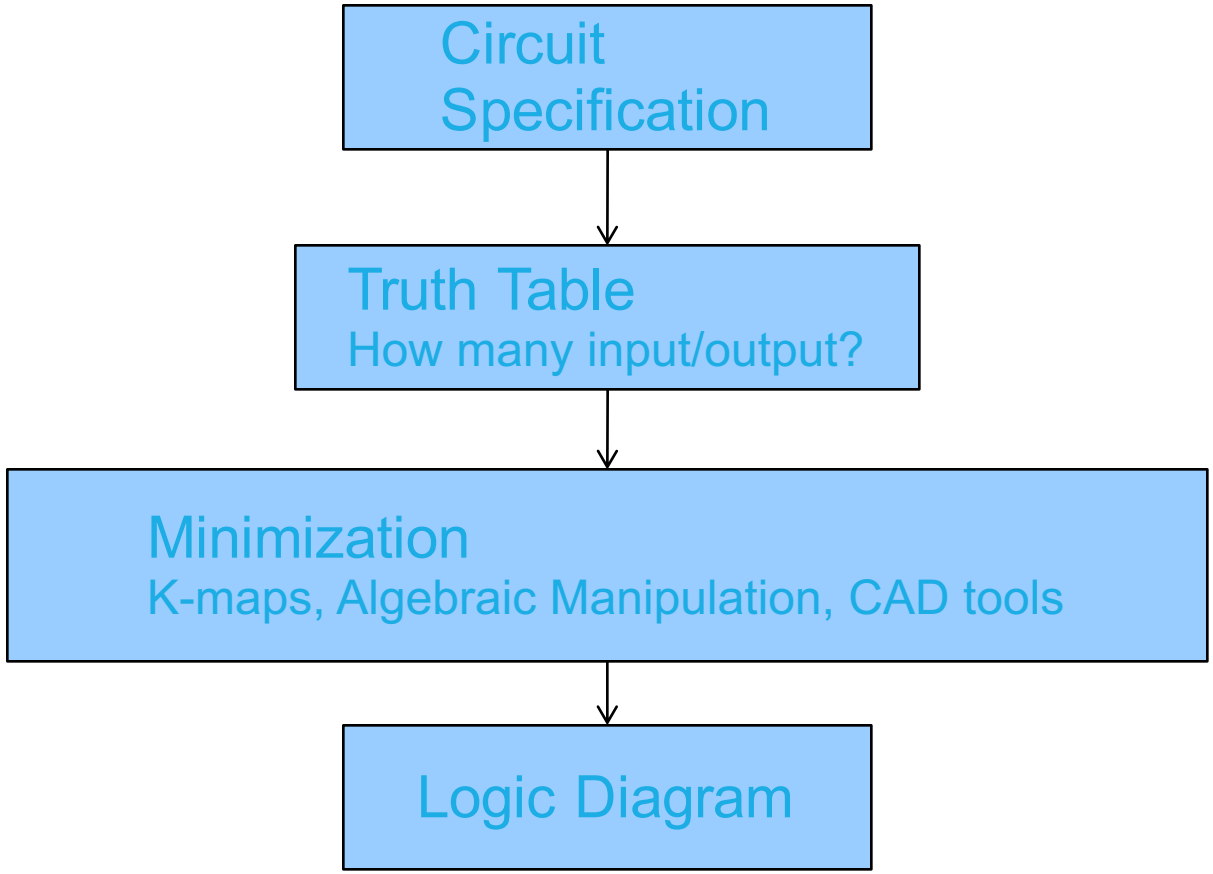
Combinational Circuits



$$Z = F(X)$$

In combinational circuits, the output at any time is a direct function of the applied external inputs

Design Procedure of Combinational Circuits

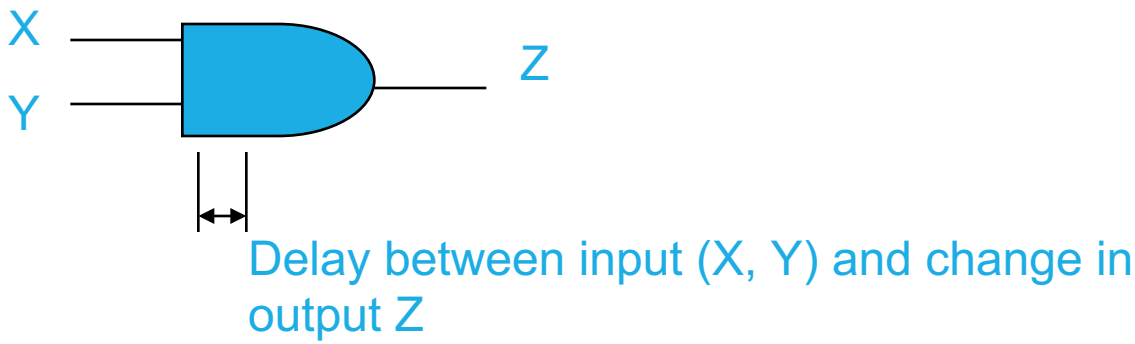


EX: L02-5

- Implement AB+CD using NAND gates only

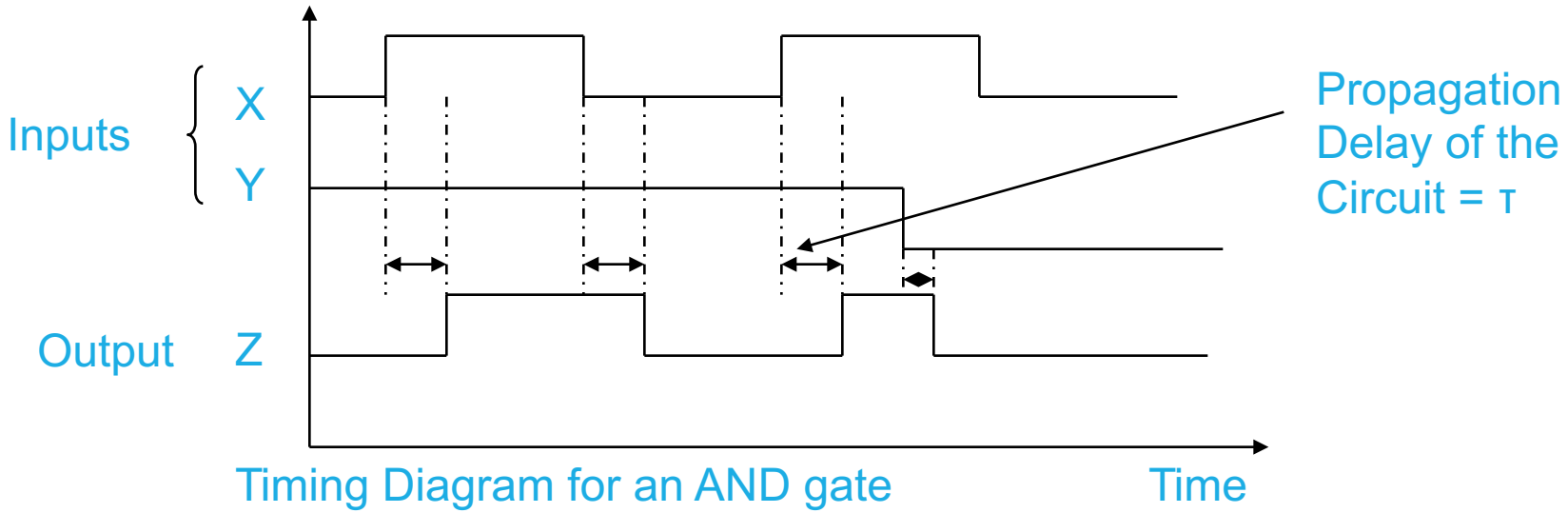
Propagation Delay

- The delay when the signal arrives at the input of a circuit, and when the output of the circuit changes, is called the propagation delay
- A circuit is considered to be fast, if its propagation delay is small (ideally as close to 0 as possible)

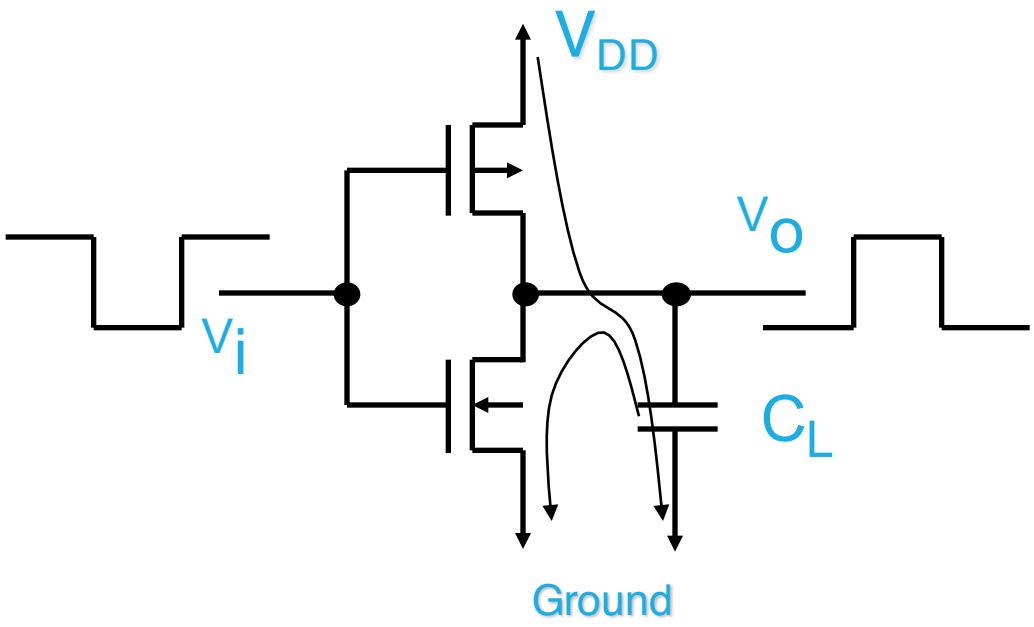


Timing Diagram

- The inputs to a circuit can be changed over time.
- The timing diagram shows the values of the input signals to a circuit with the passage of time, in the form of a waveform
- It also shows a waveform for the output



Power Consumption



Dynamic Power

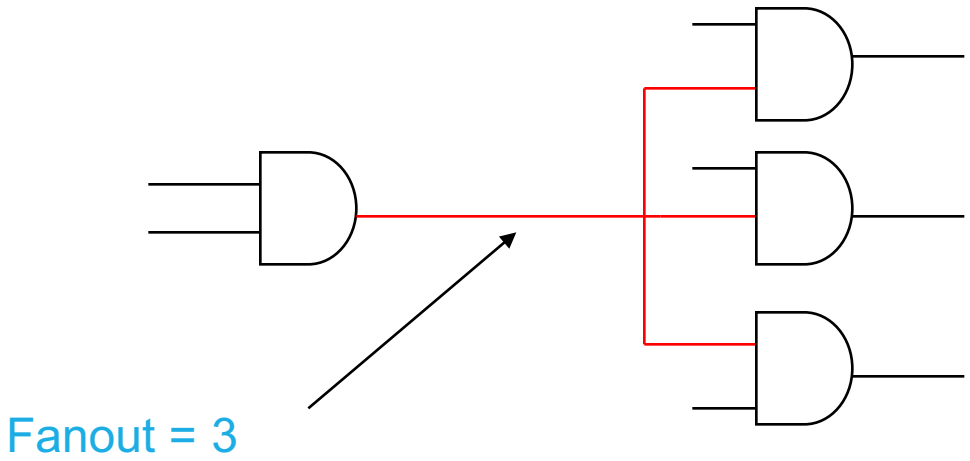
$\approx C_L V_{DD}^2/2$

Fanin

- Fanin of a gate is the number of inputs to the gate
- For a 3-input OR gate, the fanin = 3
- There is a limitation on the fanin for any gate
- In CMOS IC technology, higher fanin implies slower gates (higher propagation delays)

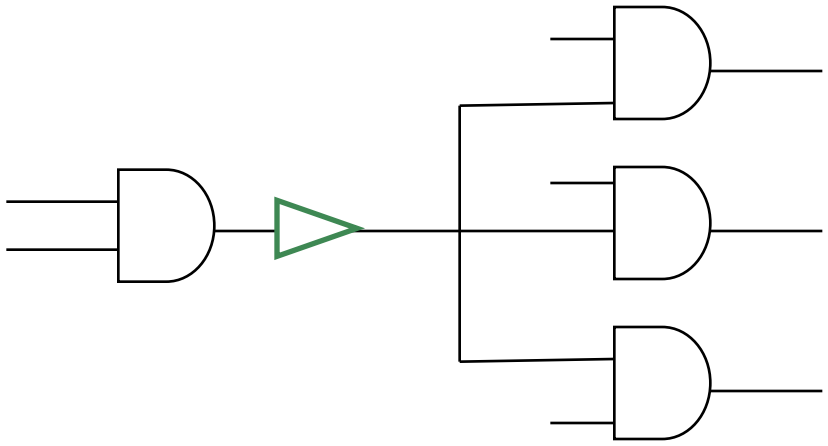
Fanout

- Fanout is the number of gates that can be driven by a driver gate
- The driven gate is called the load gate
- There is a limit to the number of load gates that can be driven by a driver gate



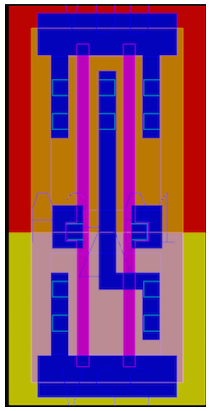
Buffers

- Buffers have a single input and a single output, where output = input
- Buffers help increase the driving capability of a circuit by increasing the fanout
- Drive strength: how much load a gate can drive
- Greater drive strength, fanout gates (dis)charged quickly

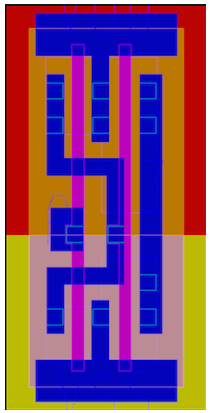


How to increase drive strength?

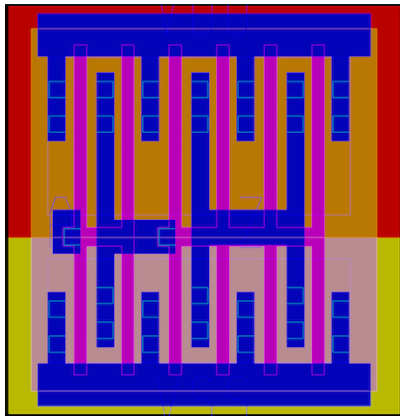
- Reduce resistance -> Increase output current
  - Increase transistor size with gate
  - Parallel a number of transistors



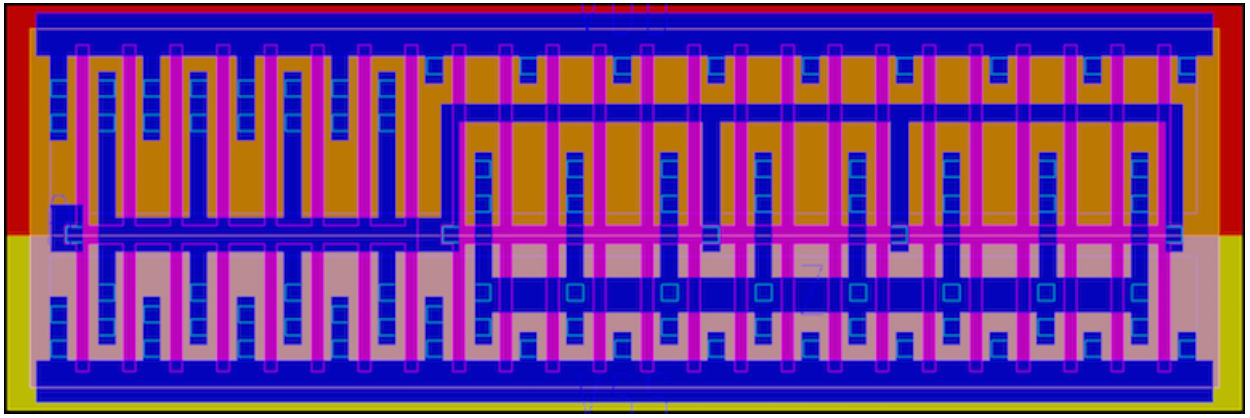
NAND



BUF_X1

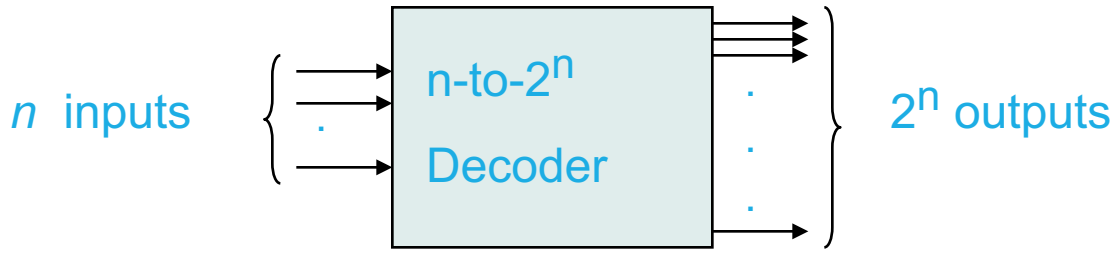


BUF_X4



BUF_X16

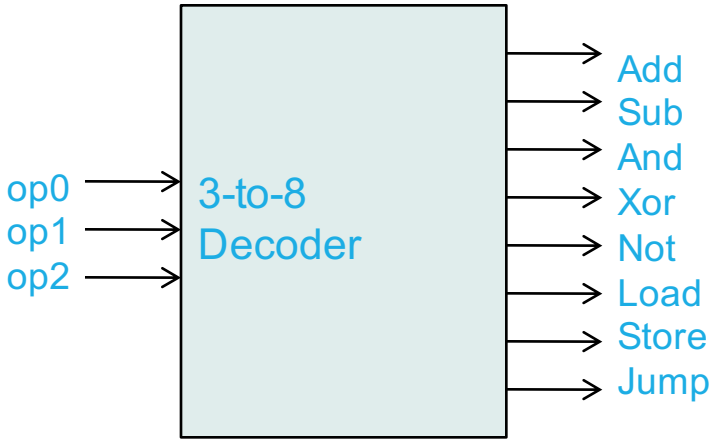
Decoder



- Information is represented by binary codes
- Decoding - the conversion of an n-bit input code to an m-bit output code with n <= m <= 2^n such that each valid code word produces a unique output code
- Circuits that perform decoding are called decoders
- A decoder is a minterm generator

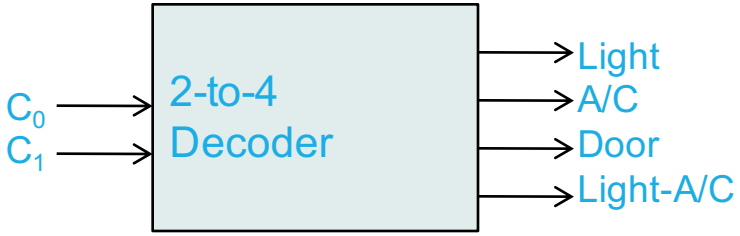
Decoder (Use Cases)

- **Decode a 3-bit op-codes:**



Load a
Add b
Store c
.
.

- **Home automation:**



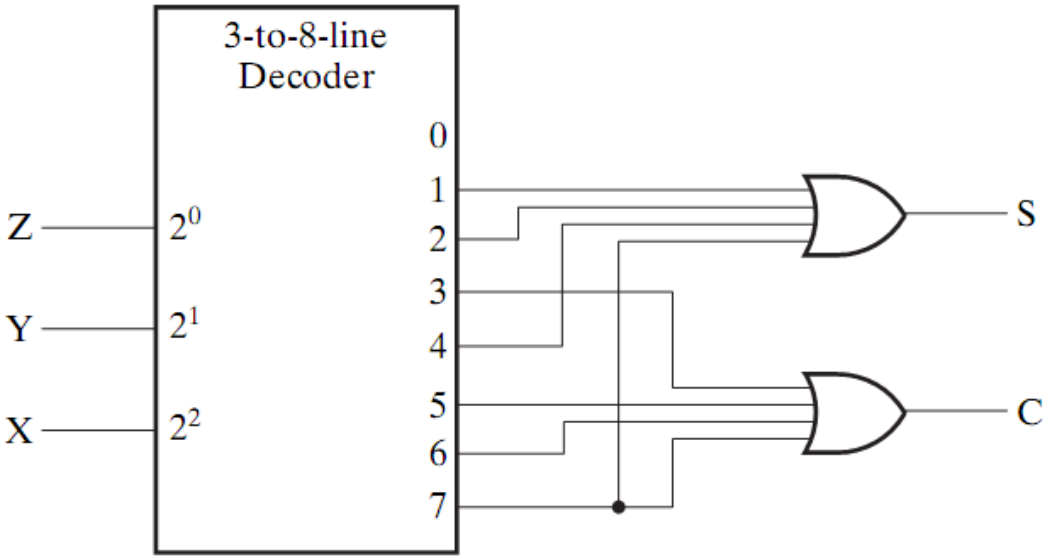
Decoder-Based Circuits

X	Y	Z	C	S
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

$S = \sum (1,2,4,7)$

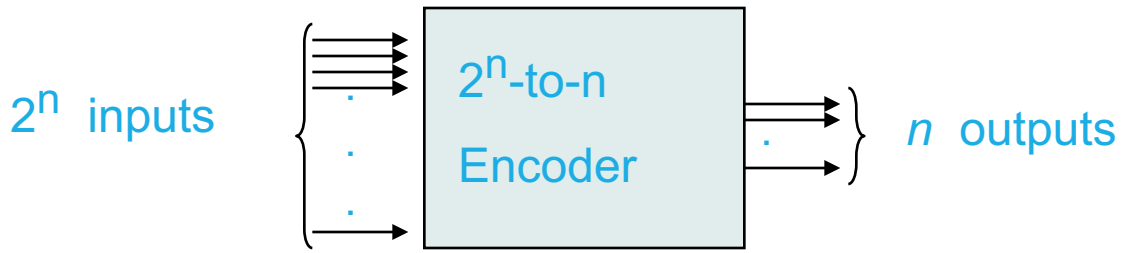
$C = \sum (3,5,6,7)$

3 inputs and 8 possible minterms
3-to-8 decoder can be used for implementing this circuit



Src: Mano's book

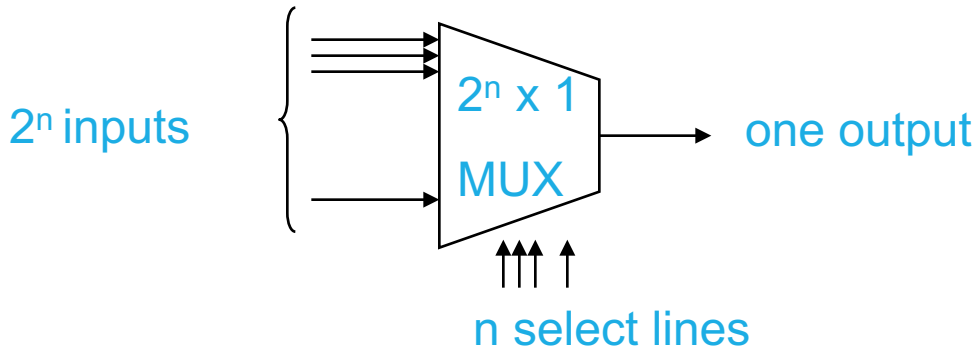
Encoder



- Encoding - the opposite of decoding - the conversion of an m-bit input code to a n-bit output code such that each valid code word produces a unique output code
- Circuits that perform encoding are called encoders
- An encoder has 2^n (or fewer) input lines and n output lines which generate the binary code corresponding to the input values
- Typically, an encoder converts a code containing exactly one bit that is 1 to a binary code corresponding to the position in which the 1 appears.

Multiplexers

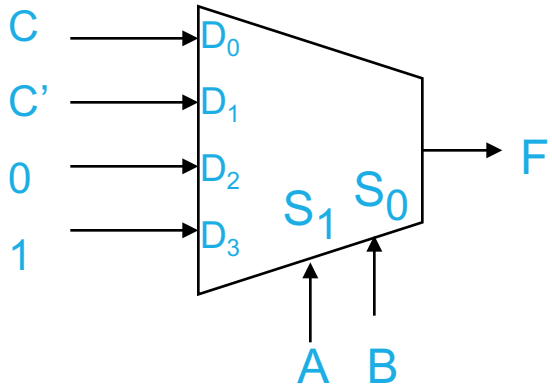
- Directs one of 2^n input to the output
- Input to output direction is done based on a set of n select bits



MUX-based Design (n-1 Select lines)

A	B	C	F	
0	0	0	0	F = C
0	0	1	1	
0	1	0	1	F = C’
0	1	1	0	
1	0	0	0	F = 0
1	0	1	0	
1	1	0	1	F = 1
1	1	1	1	

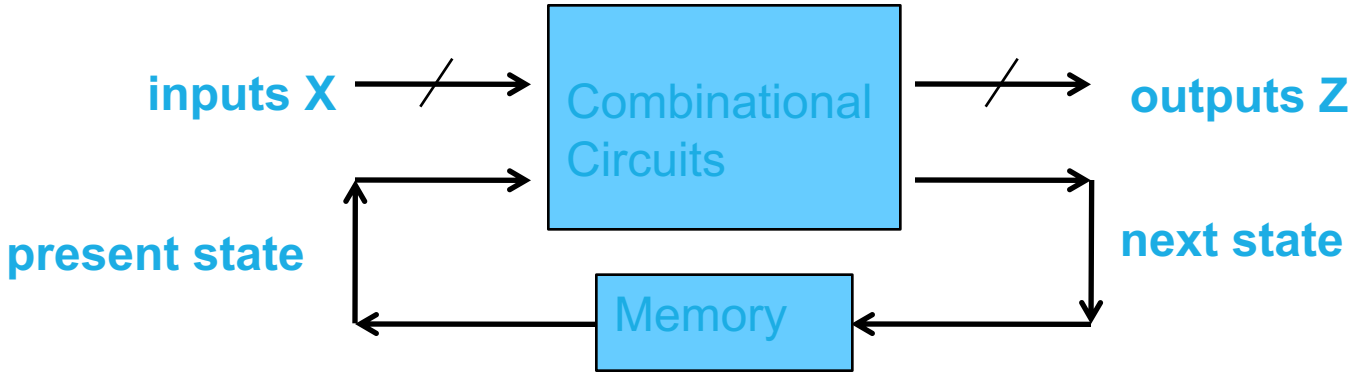
$F(A,B,C)=\sum(1,2,6,7)$



Combinational vs Sequential

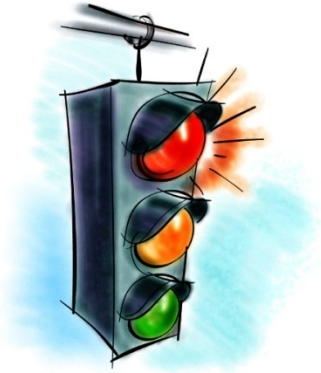
- A combinational circuit:
- At any time, outputs depends only on inputs
  - Changing inputs changes outputs
- History is ignored !

Combinational vs Sequential



- A sequential circuit:
- outputs depends on inputs and previous inputs
  - Previous inputs are stored as binary information into memory
  - The stored information at any time defines a state
- next state depends on inputs and present state

Examples of sequential systems



Traffic light

ATM

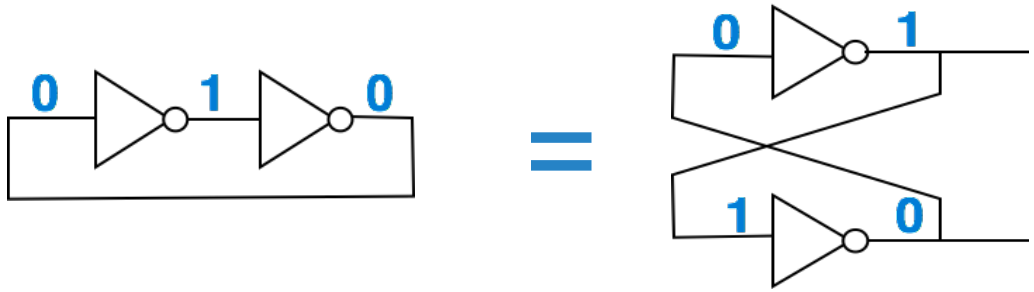
Vending machine

Types of Sequential Circuits

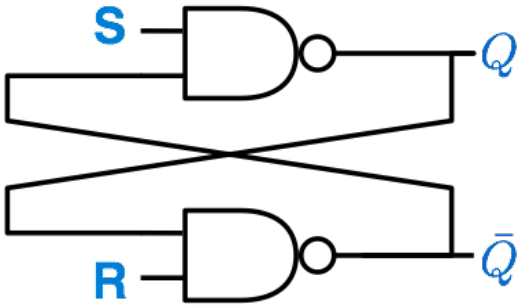
- Two types of sequential circuits:
  - Synchronous: The behavior of the circuit depends on the input signal values at discrete intervals of time (also called clocked)
  - Asynchronous: The behavior of the circuit depends on the order of change of the input signals at any instance of time (continuous)

Design A Latch

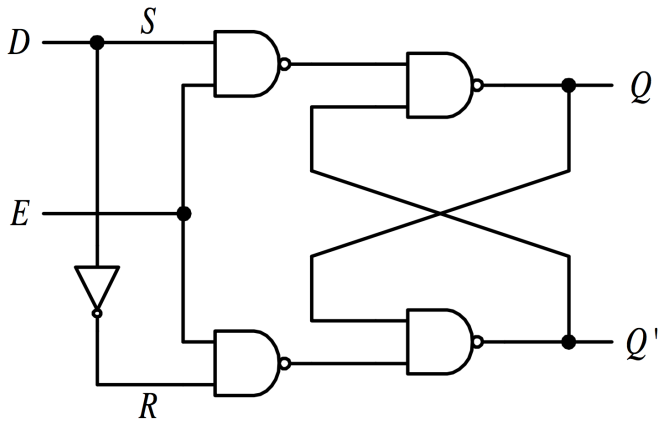
- Store one bit of information: cross-coupled invertor



- How to change the value stored?



RS-Latch

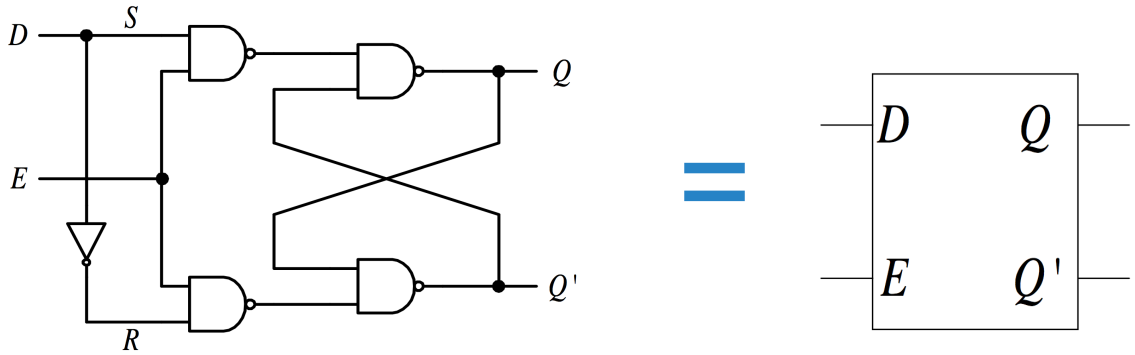


D-Latch

other Latch structures

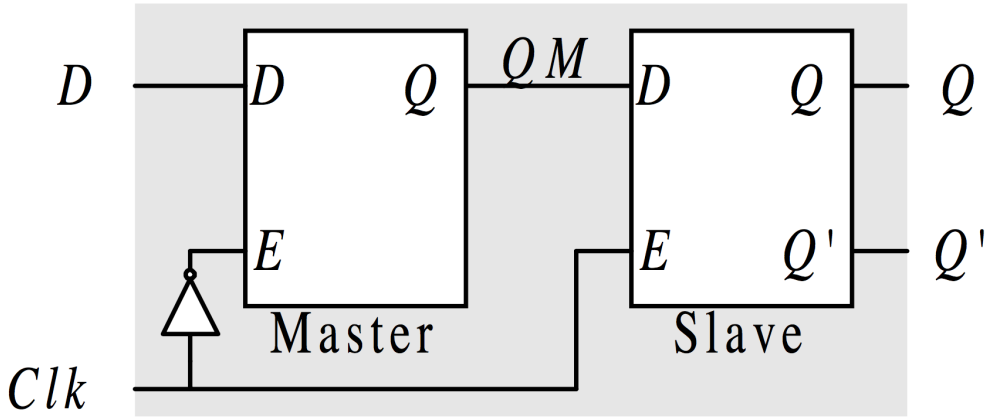
Design A Flip-Flop

- Based on D-Latch



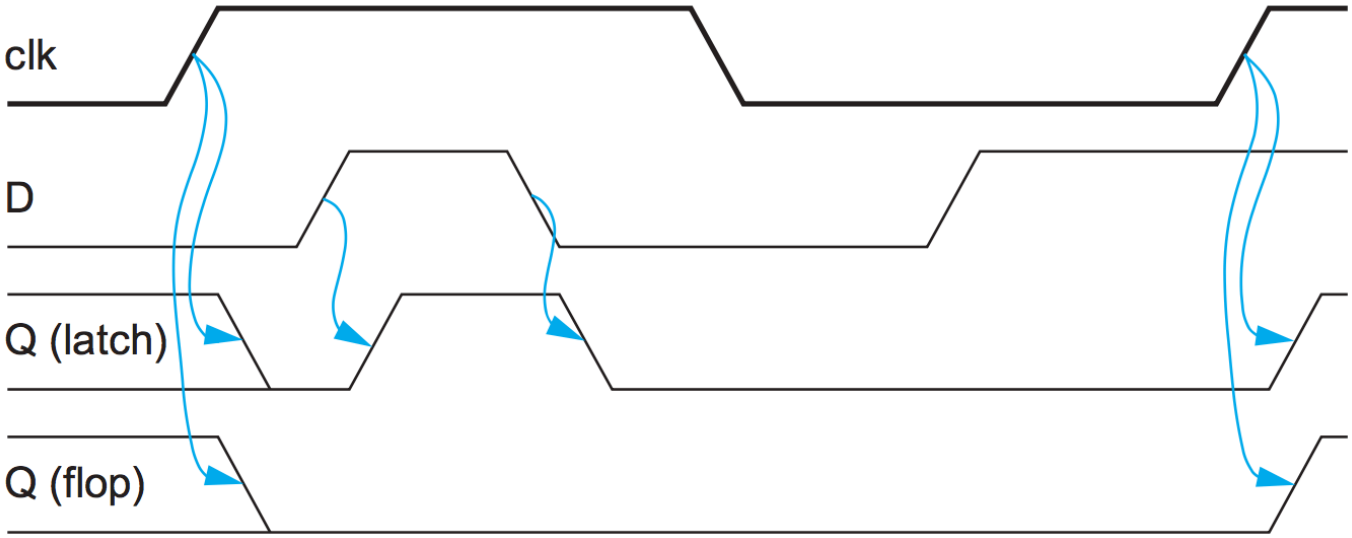
D-Latch

- Master-slave positive-edge-triggered D flip-flop



Latch and Flip-Flop

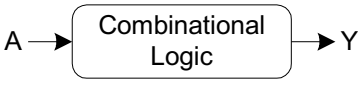
- Latch is level-sensitive
- Flip-flop is edge triggered

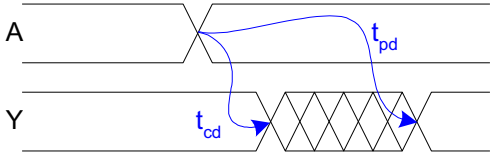


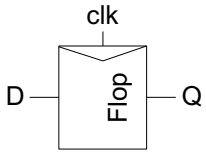
Timing Diagrams

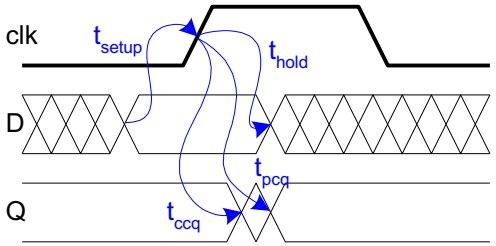
Contamination and Propagation Delays

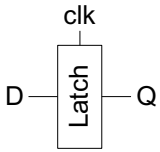
t_{pd}	Logic Prop. Delay
t_{cd}	Logic Cont. Delay
t_{pcq}	Latch/Flop Clk-Q Prop Delay
t_{ccq}	Latch/Flop Clk-Q Cont. Delay
t_{pdq}	Latch D-Q Prop Delay
t_{pcq}	Latch D-Q Cont. Delay
t_{setup}	Latch/Flop Setup Time
t_{hold}	Latch/Flop Hold Time

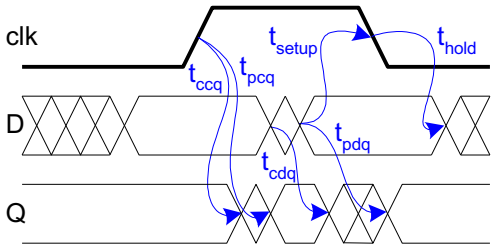










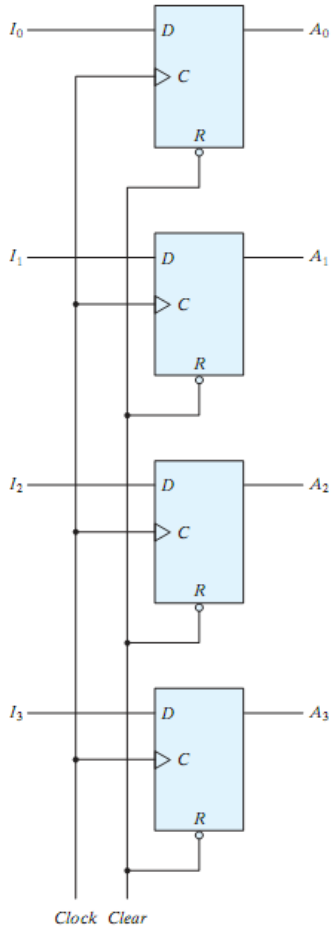


Registers

- A register is a group of flip-flops.
- An n-bit register is made of n flip-flips and can store n bits
- A register may have additional combinational gates to perform certain operations

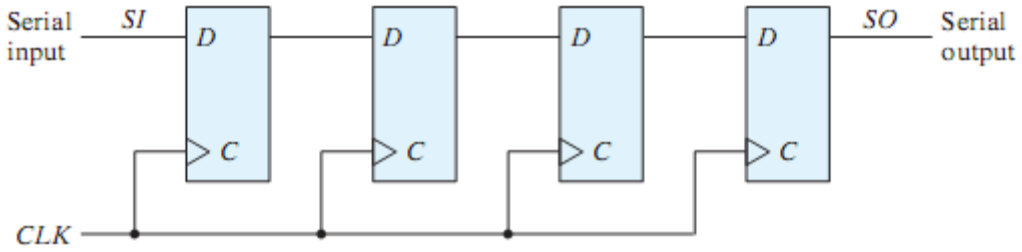
4-Bit Register

- A simple 4-bit register can be made with 4 D-FF
- Common Clock
  - At each positive-edge, 4 bits are loaded in parallel
  - Previous data is overwritten
- Common Clear
  - Asynchronous clear
  - When Clear = 0, all FFs are cleared; i.e. 0 is stored.



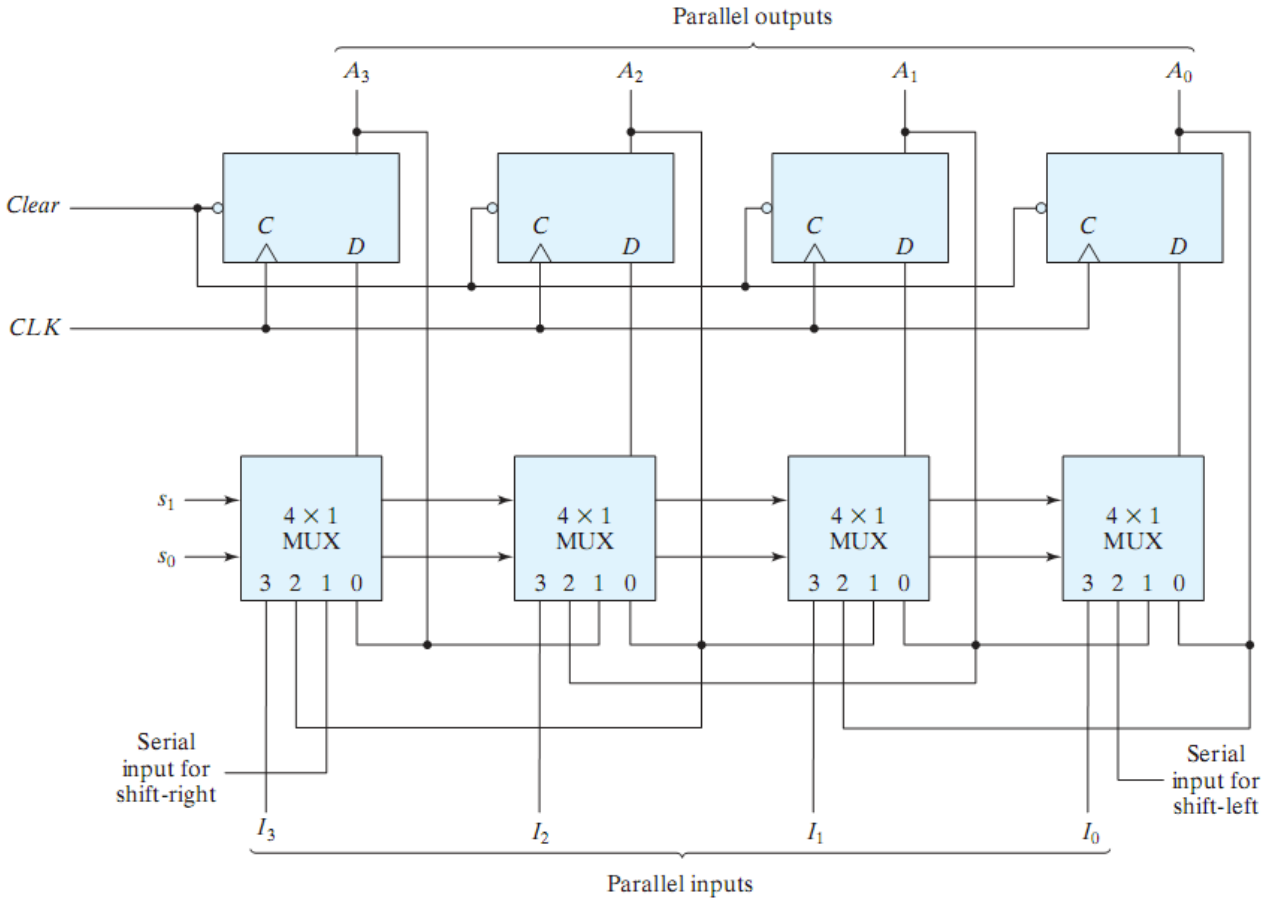
4-bit Shift Register

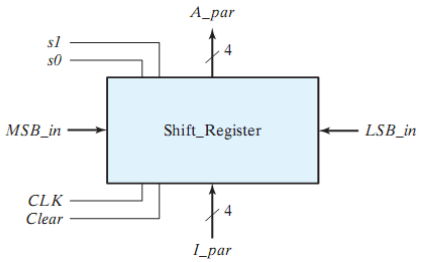
Serial-in and Serial-out (SISO)



- A simple 4-bit shift register can be made with 4 D-FF
- Common Clock
  - At each positive-edge, 1 bit is shifted in
  - Rightmost bit is discarded
- Which direction this register is shifting?

Universal Shift Register (cont.)





Mode Control		
s_1	s_0	Register Operation
0	0	No change
0	1	Shift right
1	0	Shift left
1	1	Parallel load