Digital Logic

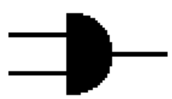
COMP375 Computer Architecture
and Organization

Types of Digital Logic

- **Combinational Logic** – Digital circuits that have no memory. The same inputs always produce the same output.
- **Sequential Logic** – Logic elements with memory whose output depends on the input and the current contents of the memory.

Logic Gates

AND *

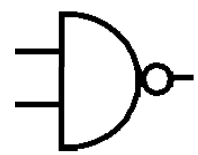


OR +

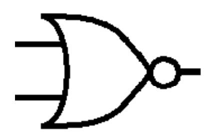
NOT $\overline{X}$

Negative Logic Gates

NAND



NOR



Boolean Laws

Identity	$X + 0 = X$ and $X * 1 = X$
One	$X + 1 = 1$
Zero	$X * 0 = 0$
Inverse	$\overline{X} + X = 1$ and $\overline{X} * X = 0$
Reflexive	$X + X = X$ and $X * X = X$
Commutative	$X + Y = Y + X$ and $X*Y = Y*X$
Associative	$X + (Y + Z) = (X + Y) + Z$
	$X * (Y * Z) = (X * Y) * Z$

Boolean Laws

Distributive	$X * (Y + Z) = (X * Y) + (X * Z)$
	$X + (Y * Z) = (X + Y) * (X + Z)$
DeMorgan's	$X + Y = \overline{(\overline{X} * \overline{Y})}$
	$X * Y = \overline{(\overline{X} + \overline{Y})}$

Sum of Product Form

Sum of Product	$(A*B) + (C*D)$
Product of Sums	$(A+B) * (C + D)$

Truth Table to Function

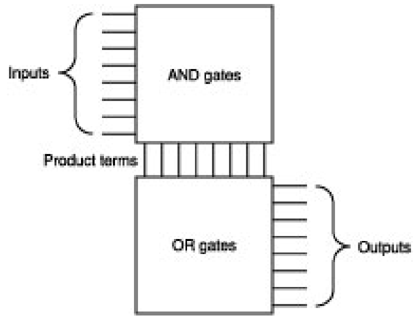
- A sum of products solution can be written by ORing the lines of the truth table that are true.

$F = \overline{A}\overline{B}\overline{C} + A\overline{B}C + AB\overline{C}$

A	B	C	**F**
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	0

Programmable Logic Array

- PLAs implement the sum of products

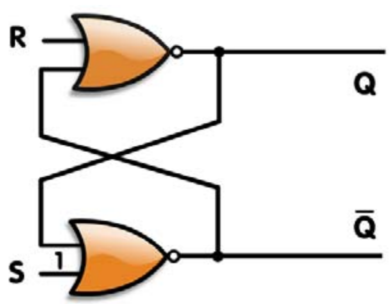


Sequential Logic

- Some logic circuits have memory that determines the future outcome of the circuit.
- Flip-flops are a simple sequential logic elements.

SR Flip-Flops

- An SR flip-flop can be constructed from two NOR gates

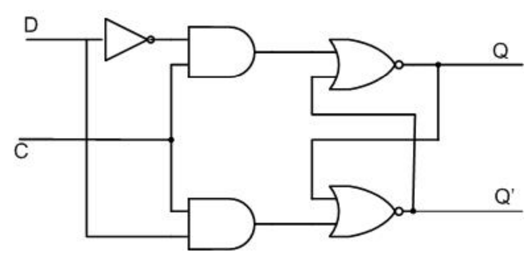


S	R	Action
0	0	Keep state
0	1	Q = 0
1	0	Q = 1
1	1	Restricted combination

D Flip-Flop

- A D flip-flop has only one data input plus enable, C

C	D	Q_{next}	Comment
0	X	Q_{prev}	No change
1	0	0	Reset
1	1	1	Set



Logic Simplification

- It is frequently possible to simplify a logical expression. This makes it easier to understand and requires fewer gates to implement.
- There are several simplification techniques including Boolean algebra and Karnough maps.

Karnough maps

	$\bar{C}\bar{D}$	$\bar{C}D$	CD	$C\bar{D}$
$\bar{A}\bar{B}$	0000	0001	0011	0010
$\bar{A}B$	0100	0101	0111	0110
AB	1100	1101	1111	1110
$A\bar{B}$	1000	1001	1011	1010

Single Variable Group of 8

	$\bar{C}\bar{D}$	$\bar{C}D$	CD	$C\bar{D}$
$\bar{A}\bar{B}$	0000	0001	0011	0010
$\bar{A}B$	0100	0101	0111	0110
AB	1100	1101	1111	1110
$A\bar{B}$	1000	1001	1011	1010

Single Variable Group of 8

	$\bar{C}\bar{D}$	$\bar{C}D$	CD	$C\bar{D}$
$\bar{A}\bar{B}$	0000	0001	0011	0010
$\bar{A}B$	0100	0101	0111	0110
AB	1100	1101	1111	1110
$A\bar{B}$	1000	1001	1011	1010

Single Variable Group of 8

	$\bar{C}\bar{D}$	$\bar{C}D$	CD	$C\bar{D}$
$\bar{A}\bar{B}$	0000	0001	0011	0010
$\bar{A}B$	0100	0101	0111	0110
AB	1100	1101	1111	1110
$A\bar{B}$	1000	1001	1011	1010

Single Variable Group of 8

	$\bar{C}\bar{D}$	$\bar{C}D$	CD	$C\bar{D}$
$\bar{A}\bar{B}$	0000	0001	0011	0010
$\bar{A}B$	0100	0101	0111	0110
AB	1100	1101	1111	1110
$A\bar{B}$	1000	1001	1011	1010

Two Variable Group of 4

	$\bar{C}\bar{D}$	$\bar{C}D$	CD	$C\bar{D}$
$\bar{A}\bar{B}$	0000	0001	0011	0010
$\bar{A}B$	0100	0101	0111	0110
AB	1100	1101	1111	1110
$A\bar{B}$	1000	1001	1011	1010

Two Variable Group of 4

	$\bar{C}\bar{D}$	$\bar{C}D$	CD	$C\bar{D}$
$\bar{A}\bar{B}$	0000	0001	0011	0010
$\bar{A}B$	0100	0101	0111	0110
AB	1100	1101	1111	1110
$A\bar{B}$	1000	1001	1011	1010

Two Variable Group of 4

	$\bar{C}\bar{D}$	$\bar{C}D$	CD	$C\bar{D}$
$\bar{A}\bar{B}$	0000	0001	0011	0010
$\bar{A}B$	0100	0101	0111	0110
AB	1100	1101	1111	1110
$A\bar{B}$	1000	1001	1011	1010

Two Variable Group of 4

	$\bar{C}\bar{D}$	$\bar{C}D$	CD	$C\bar{D}$
$\bar{A}\bar{B}$	0000	0001	0011	0010
$\bar{A}B$	0100	0101	0111	0110
AB	1100	1101	1111	1110
$A\bar{B}$	1000	1001	1011	1010

Two Variable Group of 4

	$\bar{C}\bar{D}$	$\bar{C}D$	CD	$C\bar{D}$
$\bar{A}\bar{B}$	0000	0001	0011	0010
$\bar{A}B$	0100	0101	0111	0110
AB	1100	1101	1111	1110
$A\bar{B}$	1000	1001	1011	1010

Two Variable Group of 4

	$\bar{C}\bar{D}$	$\bar{C}D$	CD	$C\bar{D}$
$\bar{A}\bar{B}$	0000	0001	0011	0010
$\bar{A}B$	0100	0101	0111	0110
AB	1100	1101	1111	1110
$A\bar{B}$	1000	1001	1011	1010

Three Variable Pairs

	$\bar{C}\bar{D}$	$\bar{C}D$	CD	$C\bar{D}$
$\bar{A}\bar{B}$	0000	0001	0011	0010
$\bar{A}B$	0100	0101	0111	0110
AB	1100	1101	1111	1110
$A\bar{B}$	1000	1001	1011	1010

Three Variable Pairs

	$\bar{C}\bar{D}$	$\bar{C}D$	CD	$C\bar{D}$
$\bar{A}\bar{B}$	0000	0001	0011	0010
$\bar{A}B$	0100	0101	0111	0110
AB	1100	1101	1111	1110
$A\bar{B}$	1000	1001	1011	1010

Three Variable Pairs

	$\bar{C}\bar{D}$	$\bar{C}D$	CD	$C\bar{D}$
$\bar{A}\bar{B}$	0000	0001	0011	0010
$\bar{A}B$	0100	0101	0111	0110
AB	1100	1101	1111	1110
$A\bar{B}$	1000	1001	1011	1010

Three Variable Pairs

	$\bar{C}\bar{D}$	$\bar{C}D$	CD	$C\bar{D}$
$\bar{A}\bar{B}$	0000	0001	0011	0010
$\bar{A}B$	0100	0101	0111	0110
AB	1100	1101	1111	1110
$A\bar{B}$	1000	1001	1011	1010

Example 1 bit Subtraction

Diff = X - Y

Bin	X	Y	Dif	Bout
0	0	0	0	0
0	0	1	1	1
0	1	0	1	0
0	1	1	0	0
1	0	0	1	1
1	0	1	0	1
1	1	0	0	0
1	1	1	1	1

Subtraction Karnough Maps

	$\bar{X}\bar{Y}$	$X\bar{Y}$	XY	$\bar{X}Y$
$\overline{Bin}$	0	1	0	1
Bin	1	0	1	0

$$Dif = \bar{X}\bar{Y}Bin + X\bar{Y}\overline{Bin} + XYBin + \bar{X}Y\overline{Bin}$$

Subtraction Karnough Maps

	$\bar{X}\bar{Y}$	$\bar{X}Y$	XY	$X\bar{Y}$
$\overline{Bin}$	0	1	0	0
Bin	1	1	1	0

$$Bout = \bar{X}Bin + \bar{X}Y + YBin$$