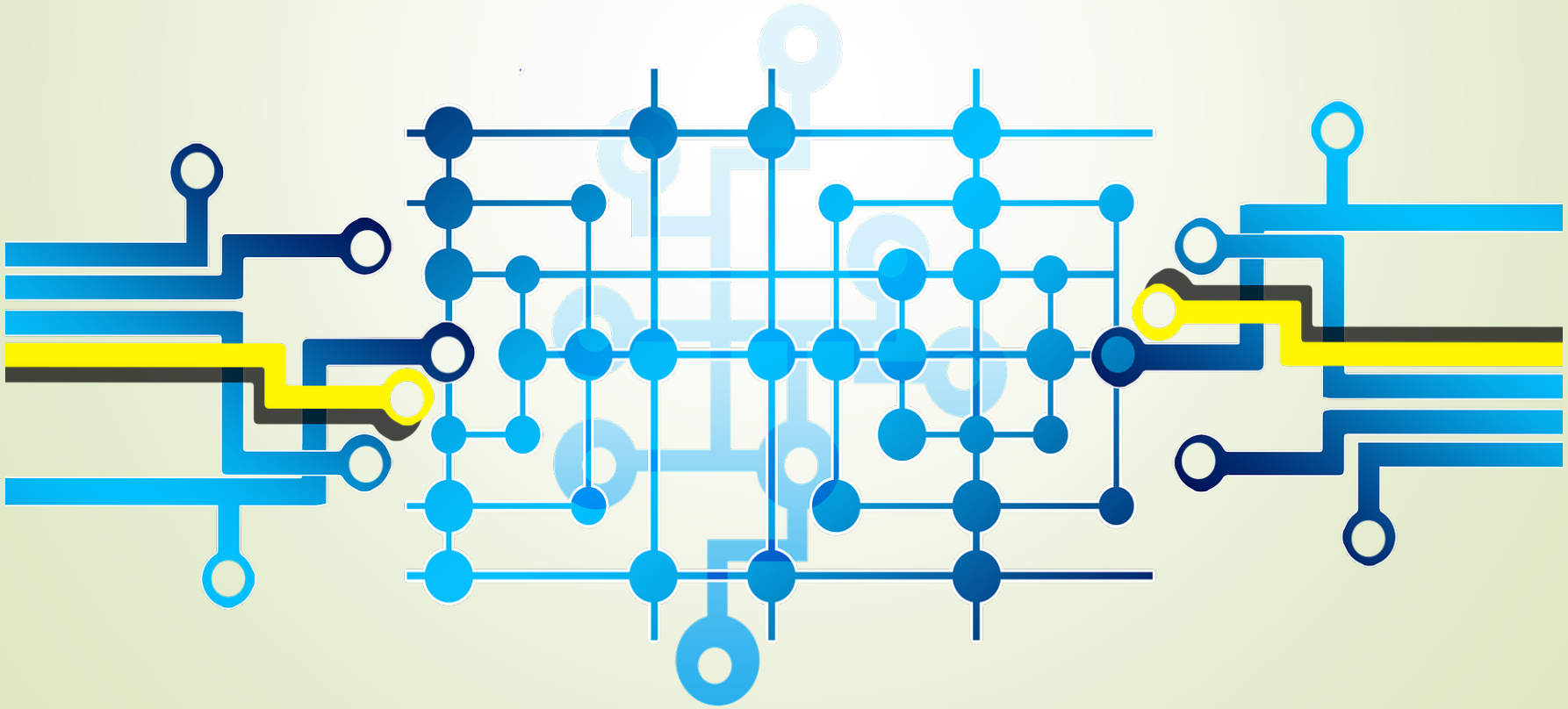
Digital Electronics (203105201)

Saurabh Srivastava, Assistant Professor
Mechatronics Engineering



CHAPTER-3

Combinational Digital Circuits

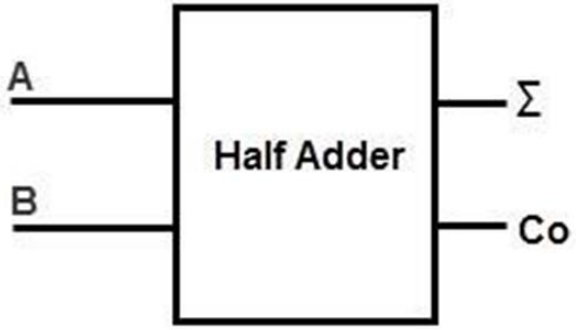
Binary Adders and Subtractors, Parallel binary adder & Subtractor, serial adder, BCD adder, carry look-ahead adder, Multiplexer, De-Multiplexer/Decoders, popular MSI chips, Magnitude comparator, parity checker/generator, code converters, priority encoders, decoders/drivers for display devices.

Combinational Digital Circuits

Binary Adder: The logic circuits designed to perform the addition of two binary numbers are called as Binary Adder Circuits

- Half Adder
- Full Adder

Half Adder: A logic circuit for adding two 1-bit numbers, or simply two bits. This circuit has two inputs and two outputs. The inputs are the two 1-bit binary numbers, and the outputs are Sum and Carry bits.

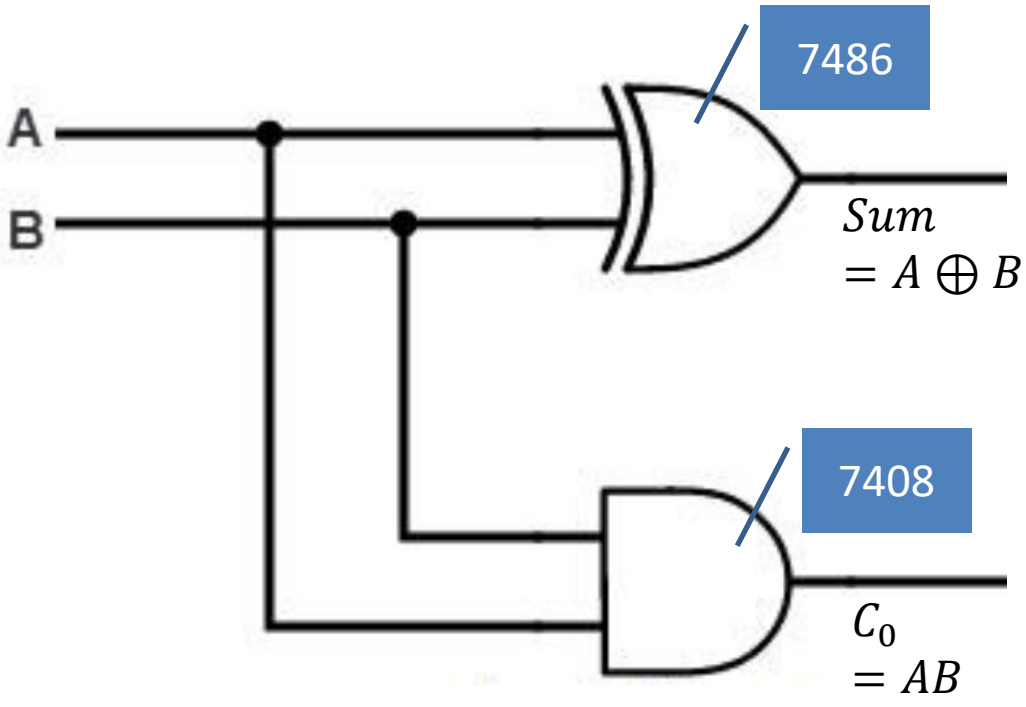


A	B	Sum (∑)	Carry Out (Co)
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

Combinational Digital Circuits

$$Sum = \bar{A}B + A\bar{B} = A \oplus B$$

$$C_0 = AB$$



A / B	0	1
0	0	**1**
1	**1**	0

A / B	0	1
0	0	0
1	0	**1**

Combinational Digital Circuits

Full Adder (CD 4008 (CMOS)) : A logic circuit for adding two 1-bit numbers, or simply two bits, with previous input carry. This circuit has three inputs and two outputs. The inputs are the three 1-bit binary numbers, and the outputs are Sum and Carry-out bits.

Inputs			Outputs	
A	*B*	C_{in}	***Sum***	C_{out}
0	0	0	0	0
0	0	1	**1**	0
0	1	0	**1**	0
0	1	1	0	**1**
1	0	0	**1**	0
1	0	1	0	**1**
1	1	0	0	**1**
1	1	1	**1**	**1**

$$Sum = \bar{A}\bar{B}C_{in} + \bar{A}B\overline{C_{in}} + A\bar{B}\,\overline{C_{in}} + ABC_{in} \qquad C_{out} = \bar{A}BC_{in} + A\bar{B}C_{in} + AB\overline{C_{in}} + ABC_{in}$$

Combinational Digital Circuits

C_{in} \ AB	00	01	11	10
0	0	1	0	1
1	1	0	1	0

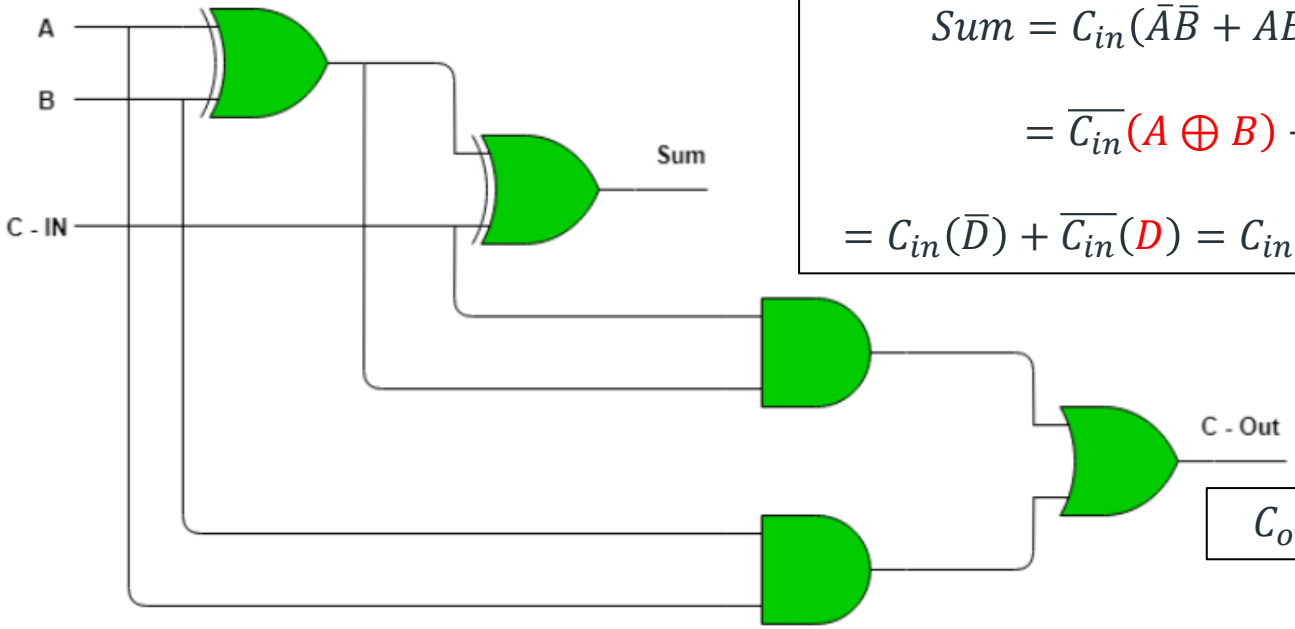
C_{in} \ AB	00	01	11	10
0	0	0	1	0
1	0	1	1	1

$$Sum = \bar{A}\bar{B}C_{in} + \bar{A}B\overline{C_{in}} + A\bar{B}\,\overline{C_{in}} + ABC_{in}$$

$$C_{out} = \bar{A}BC_{in} + A\bar{B}C_{in} + AB\overline{C_{in}} + ABC_{in} = AB + BC_{in} + AC_{in}$$

$$= (A \oplus B)C_{in} + AB$$

Combinational Digital Circuits

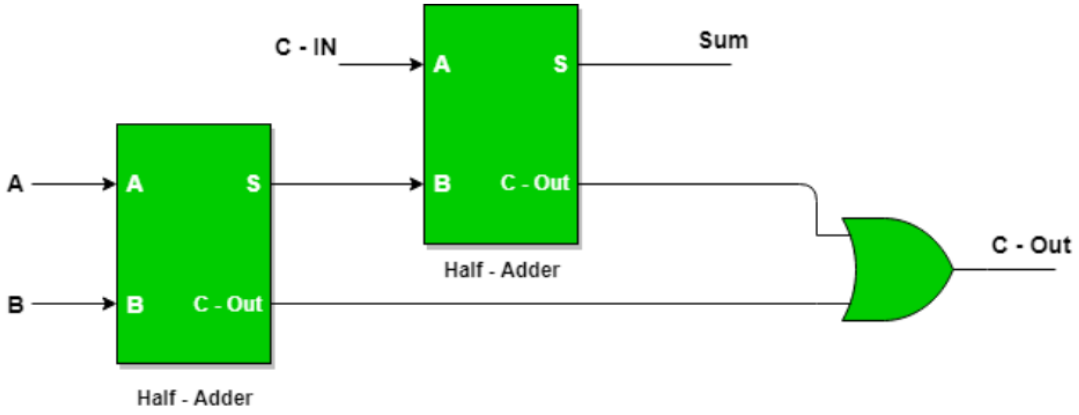


$$Sum = C_{in}(\bar{A}\bar{B} + AB) + \overline{C_{in}}(\bar{A}B + A\bar{B})$$

$$= \overline{C_{in}}(A \oplus B) + C_{in}\overline{(A \oplus B)}$$

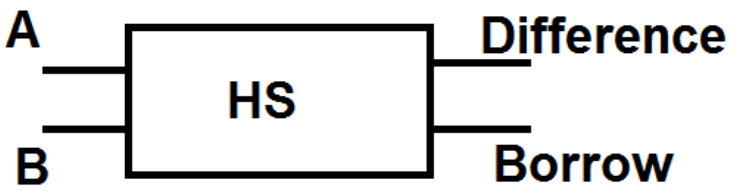
$$= C_{in}(\bar{D}) + \overline{C_{in}}(D) = C_{in} \oplus D = (A \oplus B) \oplus C_{in}$$

$$C_{out} = (A \oplus B)C_{in} + AB$$



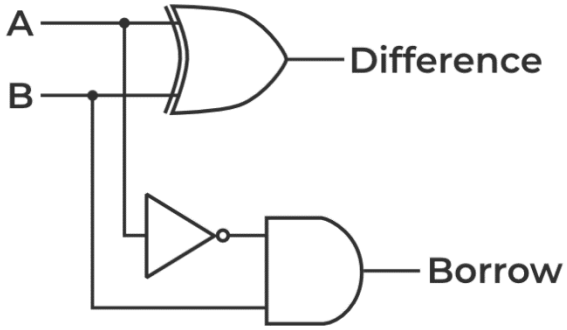
Combinational Digital Circuits

Half Subtractor: A logic circuit for subtracting two 1-bit numbers, or simply two bits. This circuit has two inputs and two outputs. The inputs are the two 1-bit binary numbers, and the outputs are Sum and Carry bits.



Half Subtractor

A	B	Diff	Borrow
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0

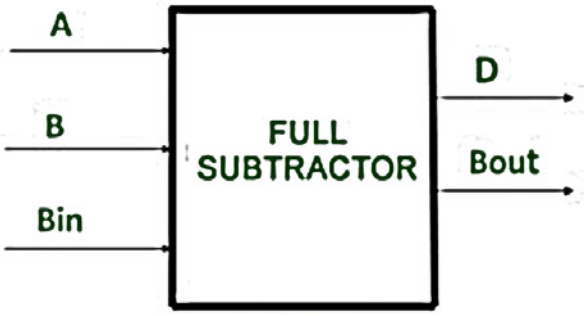


$$Difference = \bar{A}B + A\bar{B} = A \oplus B$$

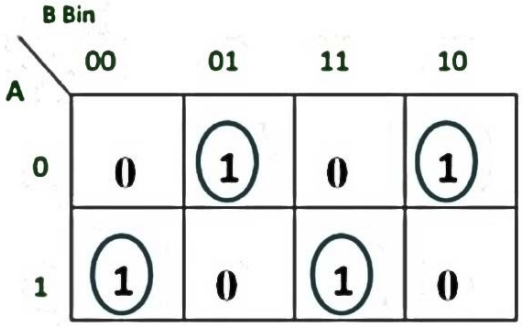
$$Borrow = \bar{A}B$$

Combinational Digital Circuits

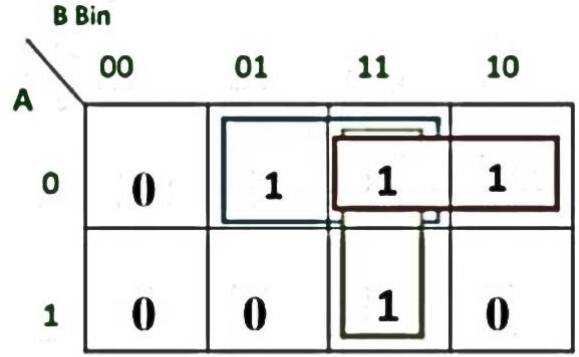
Full Subtractor: A logic circuit for subtracting two 1-bit numbers, or simply two bits, taking into account the previous borrow bit. This circuit has three inputs and two outputs. The inputs are the two 1-bit binary numbers, and the previous borrow it, and the outputs are Difference and output-borrow bits.



INPUT			OUTPUT	
A	B	Bin	D	Bout
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	0	1
1	0	0	1	0
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1



D=A'B'Bin + AB'Bin' + A'BBin' + ABBin

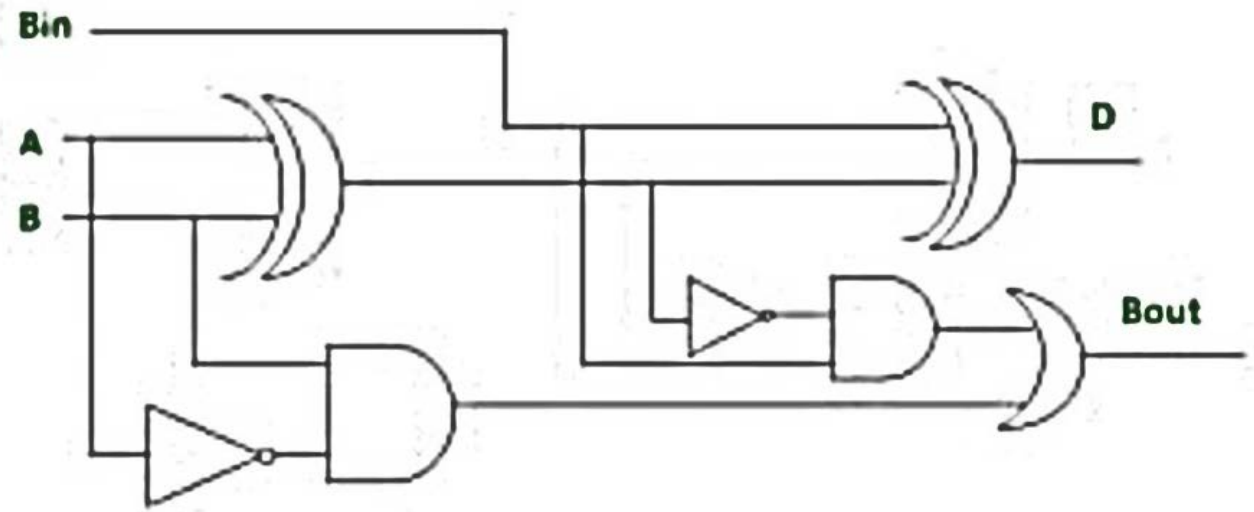


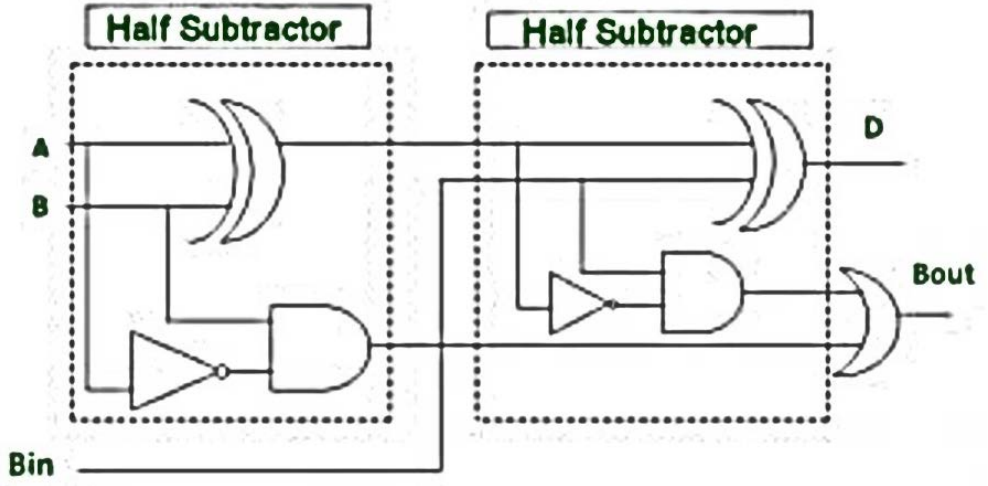
Bout=A'Bin + A'B + BBin

$$D = \bar{A}\bar{B}B_{in} + A\bar{B}\overline{B_{in}} + \bar{A}B\overline{B_{in}} + ABB_{in}$$

$$B_{out} = \bar{A}B_{in} + \bar{A}B + BB_{in}$$

Combinational Digital Circuits

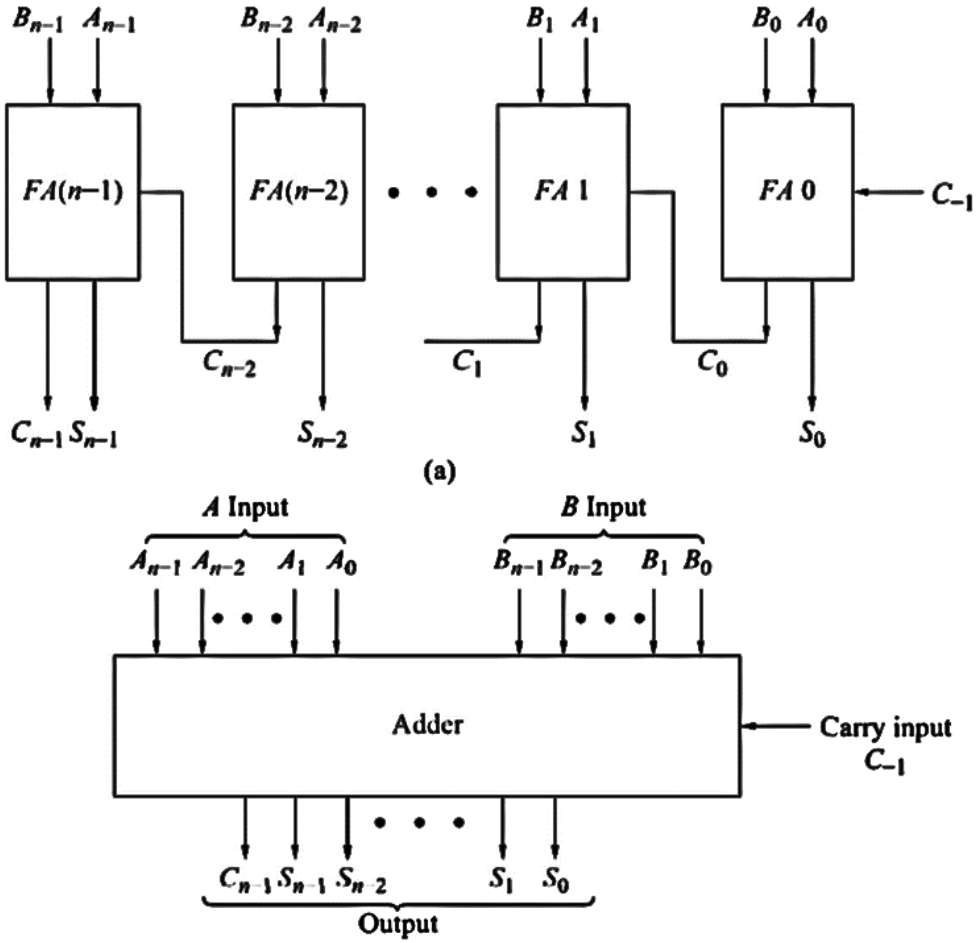




Combinational Digital Circuits

Parallel Binary Adder (74LS83):

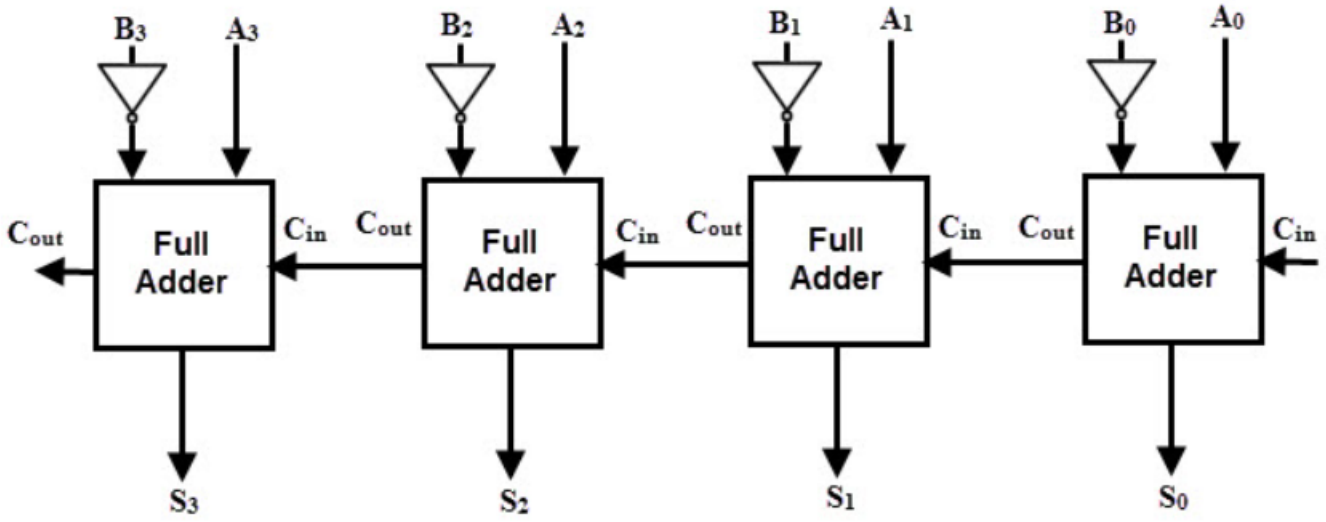
- A single full adder adds two one-bit numbers and an input carry.
- A **Parallel Adder** is a digital circuit capable of finding the arithmetic **sum** of two binary numbers that are **greater than one bit** in length by operating on corresponding pairs of bits in parallel.
- It consists of **full adders connected in a chain** where the output carry from each full adder is connected to the carry input of the next higher order full adder in the chain.
- **An n-bit parallel adder requires n full adders to perform the operation.** So for the two-bit number, two adders are needed while for four-bit numbers, four adders are needed, and so on.



Combinational Digital Circuits

Parallel Binary Subtractor:

- A Parallel Subtractor is a digital circuit capable of finding the arithmetic difference between two binary numbers that are greater than one bit in length by operating on corresponding pairs of bits in parallel.
- The parallel subtractor can be designed in several ways: including a combination of half and full subtractors, all full subtractors, or all **full adders with subtrahend complement input.**



Combinational Digital Circuits

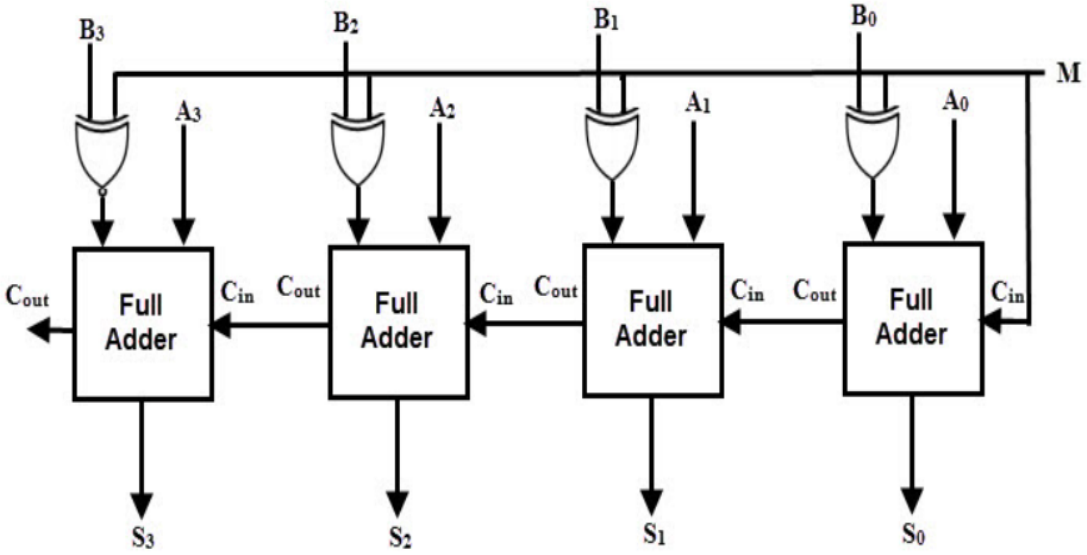
Parallel Binary Adder/ Subtractor:

•The operations of both addition and subtraction can be performed by a **one common binary adder**. Such a binary circuit can be designed by **adding an Ex-OR gate with each full adder** as shown in figure. The figure shows the 4-bit parallel binary adder/subtractor which has two 4-bit inputs as **'A_3 A_2 A_1 A_0'** and **'B_3 B_2 B_1 B_0'.**

•The **mode input control line M** is connected with the carry input of the **least significant bit** of the full adder. This **control line decides the type of operation, whether addition or subtraction.**

•When **M= 1, the circuit is a subtractor** and when **M=0, the circuit becomes adder**. The Ex-OR gate consists of two inputs to which one is connected to the B and the other to input M. When **M = 0, B Ex-OR of 0 produces B**. Then, full adders add the B with A with carry input zero and hence an addition operation is performed.

•When **M = 1, B Ex-OR of 0 produces B complement and also carry input is 1**. Hence the complemented B inputs are added to A and 1 is added through the input carry, nothing but a 2's complement operation.



Combinational Digital Circuits

Serial Adder:

•**Serial binary adder** is a combinational logic circuit that performs the addition of two binary numbers in serial form. Serial binary adder performs bit-by-bit addition.

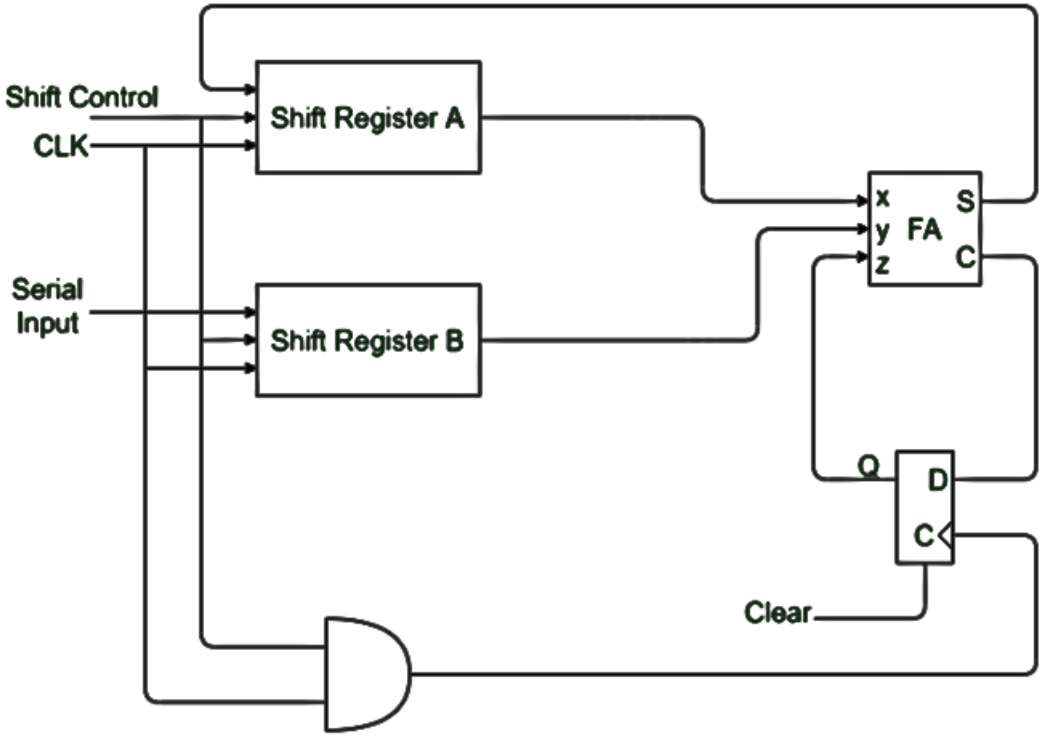
•Two shift registers are used to store the binary numbers that are to be added.

•A single full adder is used to add one pair of bits at a time along with the carry.

•The carry output from the full adder is applied to a D flip-flop.

•After that output is used as a carry for the next significant bits.

•The sum bit from the output of the full adder can be transferred into a third shift register.



Combinational Digital Circuits

Parameters	Serial Adder	Parallel Adder
Addition manner	It is used to add two binary numbers in serial form.	It is used to add two binary numbers in parallel form.
Type of Registers	A serial adder uses shift registers.	A parallel adder uses registers with parallel loads.
Requirement	It requires a single full adder.	It requires multiple full adders.
Usage of	A carry flip-flop is used in the serial adder.	Ripple carry adder is used in the parallel adder.
Circuit Type	A serial adder is a sequential circuit.	A parallel adder is a combinational circuit.
Propagation Delay	In serial adder, propagation delay is less.	In parallel adder, propagation delay is present from input carry to output carry.
Speed	The serial adder has a slow speed as compared to the parallel adder.	The parallel adder has fast speed as compared to the serial adder.
Addition process	The addition process is carried out bit by bit. Therefore, addition time relies on bit count.	The addition process is carried out simultaneously. That implies all bits sum up simultaneously. Therefore, time does not rely on bit count.
Requirement of Components	It necessitates fewer components.	It necessitates more components because of design complexity.
Number of Full Adders	The number of required full adders is fixed i.e. one.	The number of required full adders is equal to the number of bits in the binary number.

Combinational Digital Circuits

BCD adder: the 4-bit binary adder IC (7483) can be used to perform BCD addition.
If the output is not valid BCD code, or, if carry bit C3 is generated, then 6 (0110) is to be added to the sum, to get the correct output.

Q. Subtract 1 from 8

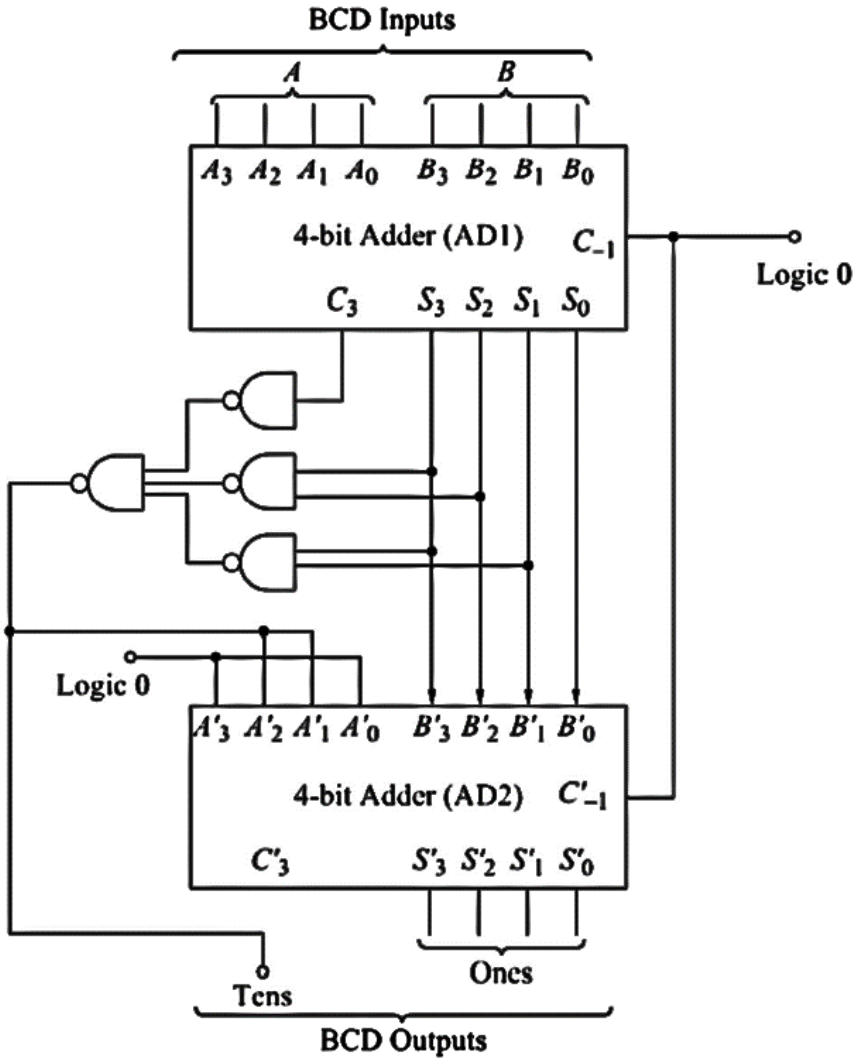
$$8 = \quad 1000$$
$$(-1) = (+)1000 \ (9's \text{ complment of } 1)$$
$$\boxed{= \quad 10000 \ (invalid)}$$

$Add\ (0110)$

$$\boxed{\begin{array}{r} 10000 \\ +0110 \\ = 10110 \end{array}}$$

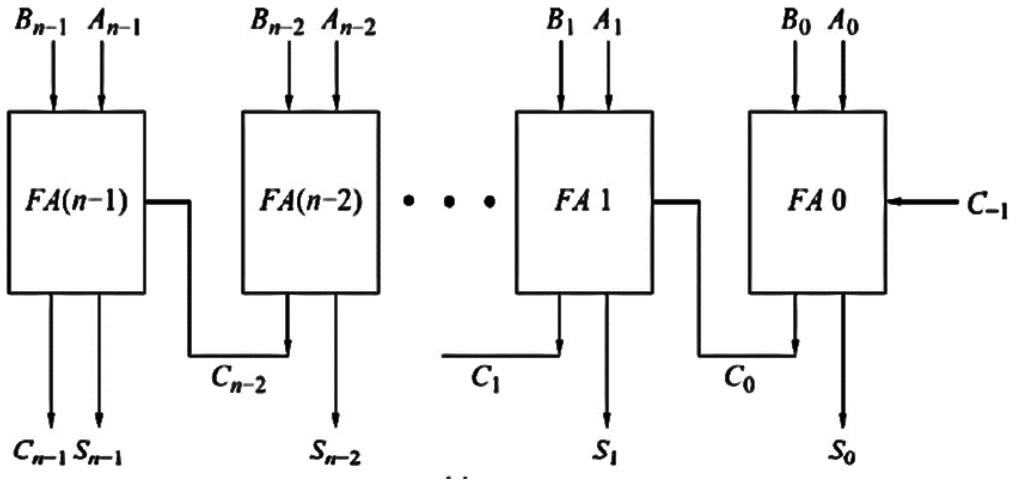
$Add\colon End - around - carry\ (1)$

$$\begin{array}{r} 10110 \\ +1 \\ = 0111 = +7 \end{array}$$



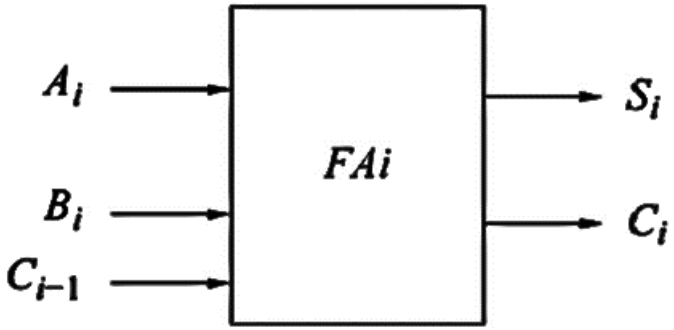
Combinational Digital Circuits

Carry Look-ahead adder (74182): Carry has to propagate along C_{-1} to C_0 to C_{n-1} (LSB to MSB position). This decreases the speed of addition.
Hence, a technique called look-ahead carry is used for addition.



C_0 and S_0 (from FA 0) are generated after some propagation delay. Only C_0 is required to propagate upto FA(n-1).
The proper output of FA1 (after its own delay) is generated only after the propagation delay of FA0, after receiving C_0.
Therefore, total delay in FA1 output = delay due to FA0 + delay due to FA1.
Hence, the propagation delay depends on the number of FA stages.

Combinational Digital Circuits

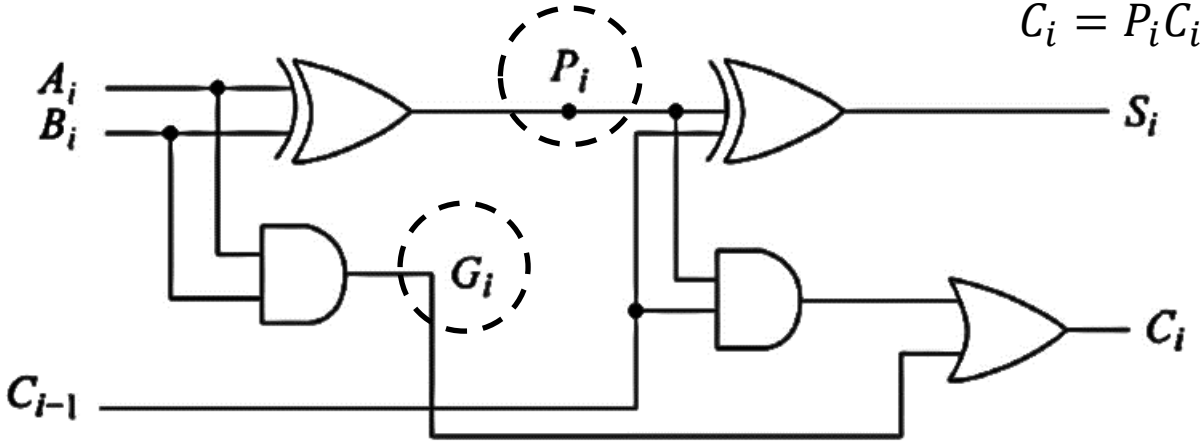


$$P_i = A_i \oplus B_i$$

$$G_i = A_i B_i$$

$$S_i = P_i \oplus C_{i-1} = A_i \oplus B_i \oplus C_{i-1}$$

$$C_i = P_i C_{i-1} + G_i$$



$$\boldsymbol{S_i = A_i \oplus B_i \oplus C_{i-1}}$$

$$\boldsymbol{C_i = P_i C_{i-1} + G_i}$$

G_i: *carry generate*
(independent of input carry)

P_i: *carry propagate*

$$C_i = G_i + P_i C_{i-1}$$

$$C_{i+1} = G_{i+1} + P_{i+1} C_i$$

$$C_{i+1} = G_{i+1} + P_{i+1}(P_i C_{i-1} + G_i)$$

Combinational Digital Circuits

Now consider a 4-bit adder:

$P_i = A_i \oplus B_i$, $G_i = A_i B_i$, $C_i = P_i C_{i-1} + G_i$

$C_0 = G_0 + P_0 C_{-1}$

$C_1 = G_1 + P_1 C_0 = G_1 + P_1(G_0 + P_0 C_{-1}) = G_1 + P_1 G_0 + P_1 P_0 C_{-1}$

$C_2 = G_2 + P_2 G_1 + P_2 P_1 G_0 + P_2 P_1 P_0 C_{-1}$

$C_3 = G_3 + P_3 G_2 + P_3 P_2 G_1 + P_3 P_2 P_1 G_0 + P_3 P_2 P_1 P_0 C_{-1}$

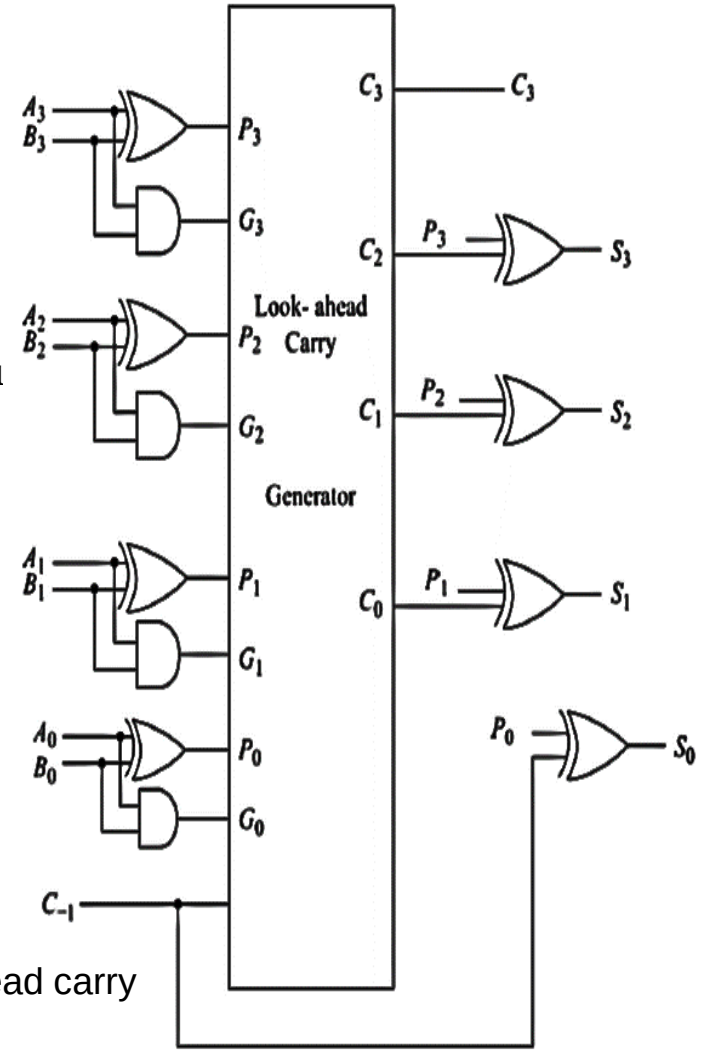
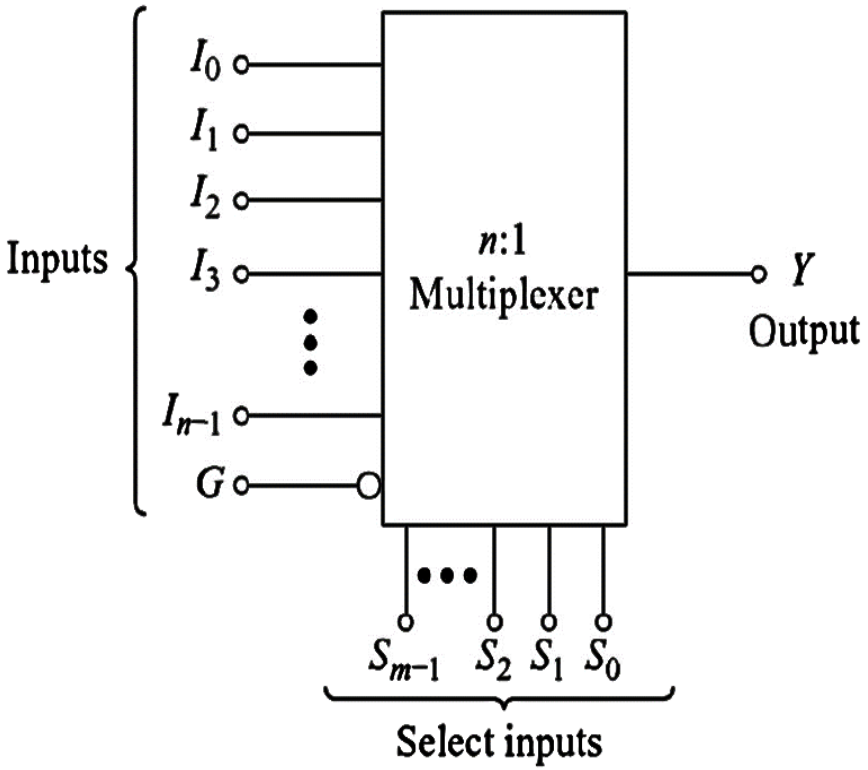


Fig. a 4-bit adder with look-ahead carry

Combinational Digital Circuits

Multiplexer: A special combinational circuit that gates one out of several inputs to a single output.

- Most widely used standard logic circuits.
- The input selected is controlled by the select lines
- For multiplexers with **n-input lines**, **m-select lines** are required for controlling (**$n=2^m$**)
- Depending upon the code at the select inputs, one out of n-inputs is selected and transferred to the output channel.
- For operating a multiplexer, an **enable (strobe) input line (G)** is utilized in the active-low state. At a high state, the multiplexer outputs zero.



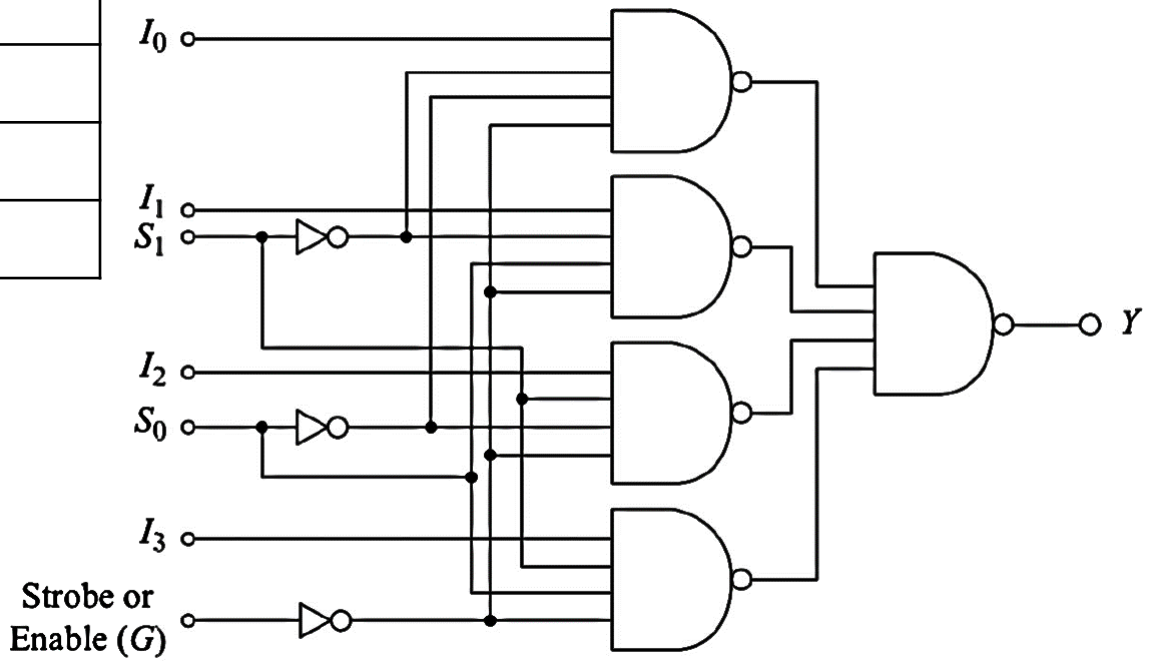
$$Y = (\overline{S_{m-1}} \dots \overline{S_1}.\overline{S_0}.I_0 + \overline{S_{m-1}} \dots \overline{S_1}.S_0.I_1 + \dots + S_{m-1} \dots S_1.S_0.I_{n-1}).\bar{G}$$

Combinational Digital Circuits

The truth table of a 4:1 multiplexer is shown below

Enable Input (G)	Select Inputs		Output Y
	S_1	S_0	
0	0	0	I_0
0	0	1	I_1
0	1	0	I_2
0	1	1	I_3
1	X	X	0

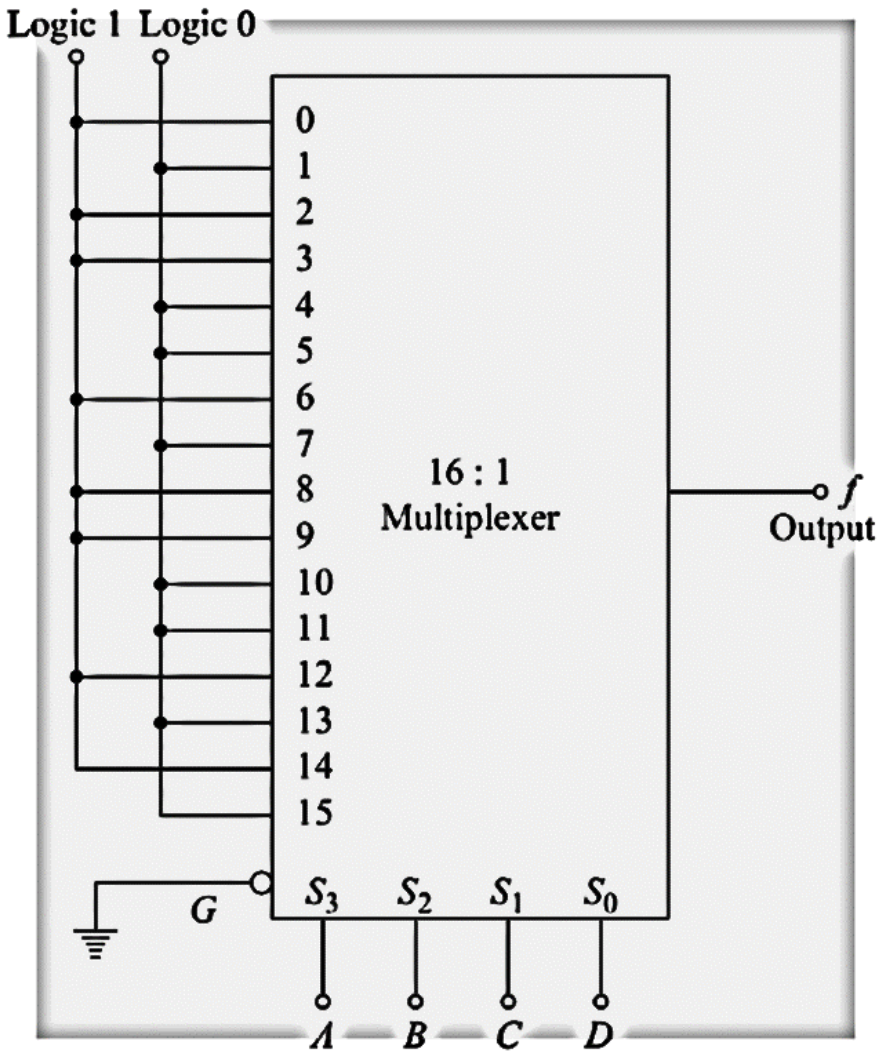
$$Y = (\overline{S_1}.\overline{S_0}.I_0 + \overline{S_1}.S_0.I_1 + S_1.\overline{S_0}.I_2 + S_1.S_0.I_3).\overline{G}$$



Combinational Digital Circuits

Q. f(A,B,C,D)= ∑m (0,2,3,6,8,9,12,14)

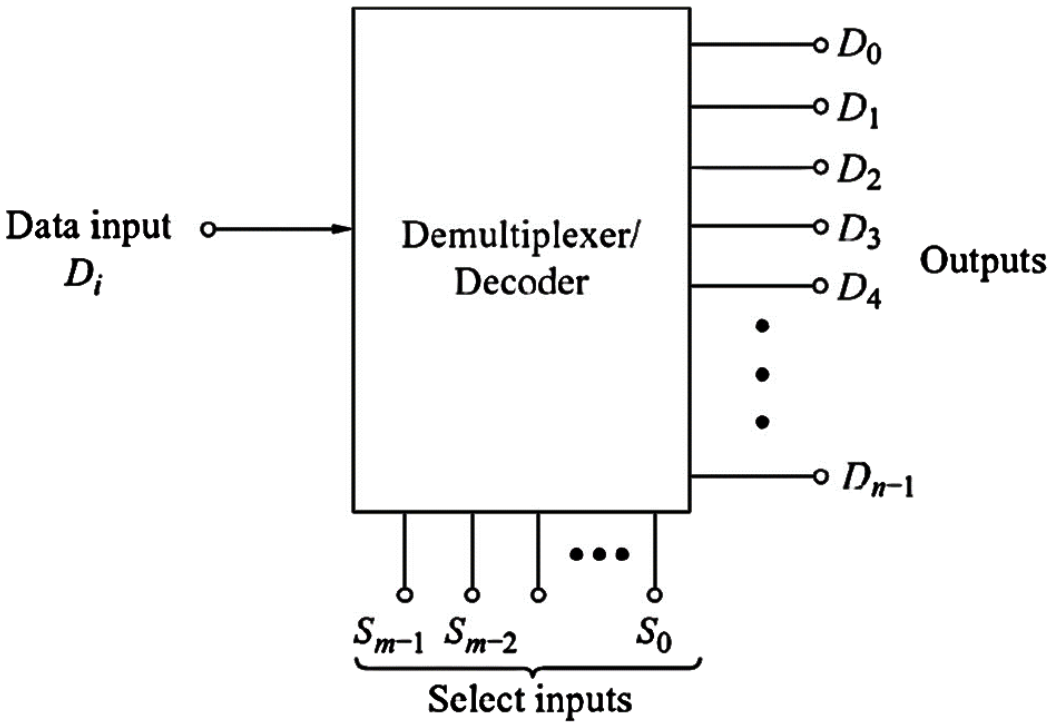
Enable Input (G)	Select Inputs				Output Y
	S_3	S_2	S_1	S_0	
0	0	0	0	0	1
0	0	0	0	1	0
0	0	0	1	0	1
0	0	0	1	1	1
0	0	1	0	0	0
0	0	1	0	1	0
0	0	1	1	0	1
0	0	1	1	1	0
0	1	0	0	0	1
0	1	0	0	1	1
0	1	0	1	0	0
0	1	0	1	1	0
0	1	1	0	0	1
0	1	1	1	0	0
0	1	1	1	1	1
1	X	X	X	X	0



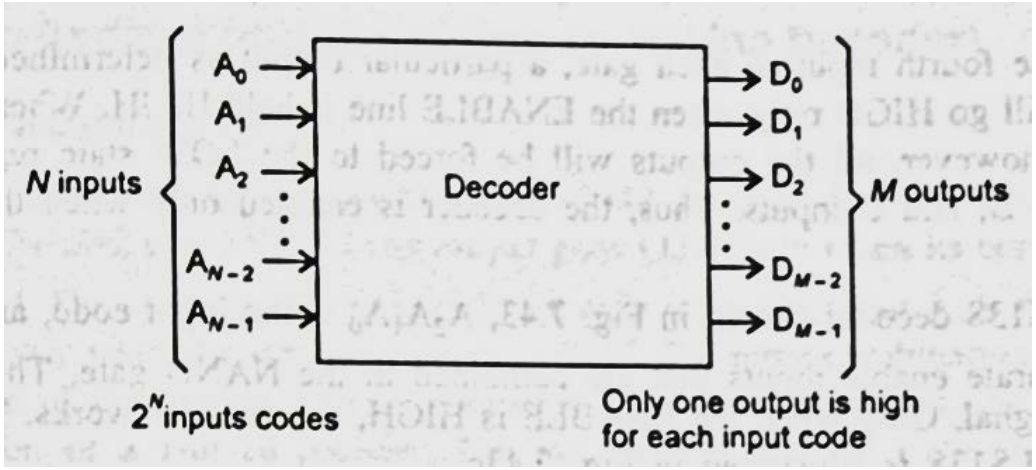
Combinational Digital Circuits

Demultiplexer: It performs the reverse of a multiplexer.

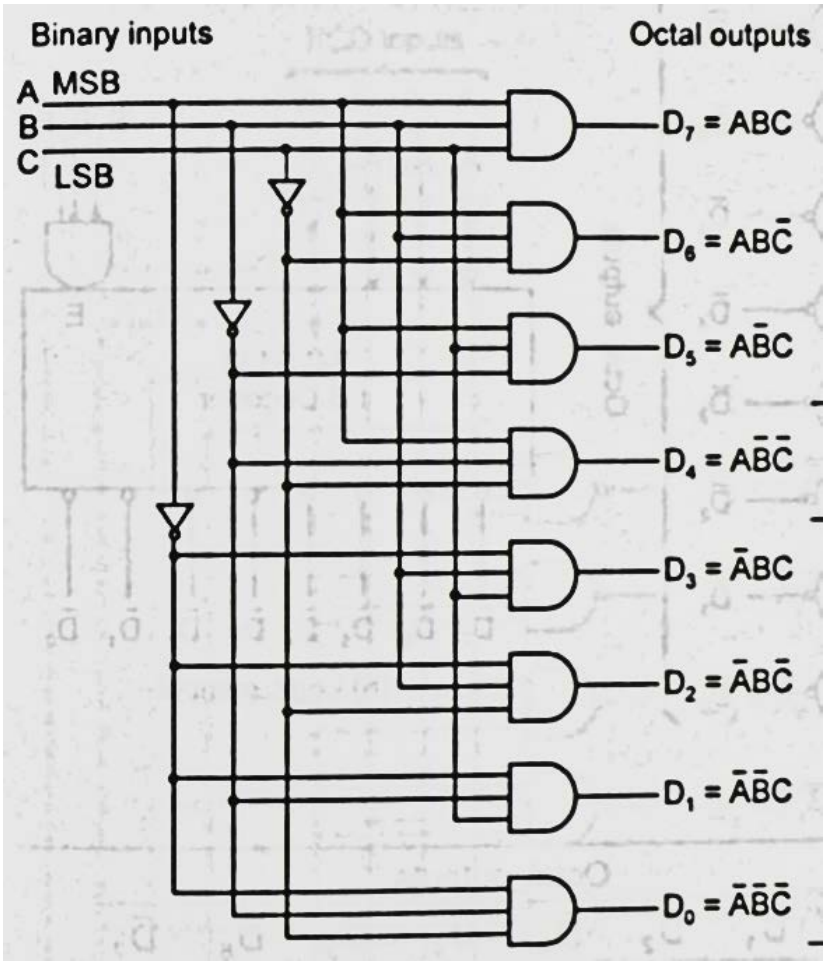
- It accepts a single input and distributes it over several outputs.
- The select input code determines to which output the data input will be transmitted.
- For demultiplexers with **n-output lines**, **m-select lines** are required for controlling (**$n=2^m$**)
- The data input D_i will appear on the output line selected by the select input.
- This can also be used as a binary to decimal decoder.



Combinational Digital Circuits



Inputs			Outputs							
A	B	C	D_0	D_1	D_2	D_3	D_4	D_5	D_6	D_7
0	0	0	1	0	0	0	0	0	0	0
0	0	1	0	1	0	0	0	0	0	0
0	1	0	0	0	1	0	0	0	0	0
0	1	1	0	0	0	1	0	0	0	0
1	0	0	0	0	0	0	1	0	0	0
1	0	1	0	0	0	0	0	1	0	0
1	1	0	0	0	0	0	0	0	1	0
1	1	1	0	0	0	0	0	0	0	1



Combinational Digital Circuits

PARAMETERS	DEMULTIPLEXER	DECODER
Definition	A demultiplexer is a circuit which takes only one input and switches it to one of the several outputs with the help of selection lines.	A decoder is the circuit which decodes the input signal fed to it with the help of control signal.
Data Input	A demultiplexer accepts the data as input.	A decoder accepts control signals as input.
Number of Inputs	Only one	One or more signals can be given as input.
Number of Outputs	The output depends on the number of selection lines. The number of outputs will be equivalent to 2 raised to the power of selection lines. If there are n selection lines then output will be 2^n.	The outputs depends on the number of control signals. If there are n control signals then the output will be 2^n.
Need of selection lines	Required	Not required, but control signals and enable signal is required.
Main Function	Used for switching	Used for decoding of the encoded input terminal.

Combinational Digital Circuits

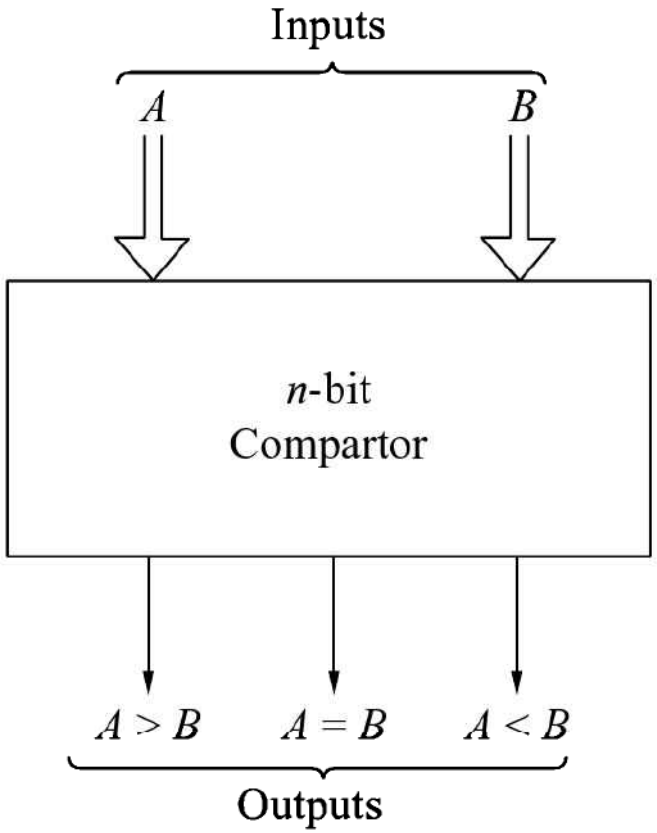
Popular MSI (Medium scale integration: 500 components (from 10 to less than 100 gates)**)**

IC No.	Description	Output
74157	Quad 2:1 Multiplexer	Same as input
74158	Quad 2:1 Multiplexer	Inverted input
74153	Dual 4:1 Multiplexer	Same as input
74352	Dual 4:1 Multiplexer	Inverted input
74151	8:1 Multiplexer	Complementary outputs
74152	8:1 Multiplexer	Inverted input
74150	16:1 Multiplexer	Inverted input
74139	Dual 1:4 Demultiplexer (2-line to 4-line decoder)	Inverted input
74155	Dual 1:4 Demultiplexer (2-line to 4-line decoder)	1 Y-inverted input 2 Y- same as input
74138	1:8 Demultiplexer (3-line to 8-line decoder)	Inverted input
74154	1:16 Demultiplexer (4-line to 16-line decoder)	Same as input

Combinational Digital Circuits

Magnitude comparator (7485 (4-bit)): a magnitude comparator compares two numbers and provides outputs depending on the values of the compared numbers.
The figure shows an n-bit comparator that produces the output as either A>B, A=B, or, A<B.
The 4-bit comparators are available as MSI IC package (7485) which can compare binary and BCD codes.

The truth table of a 2-bit comparator is given next



Combinational Digital Circuits

Inputs				Outputs		
A_1	A_0	B_1	B_0	A>B	A=B	A<B
0	0	0	0	0	1	0
0	0	0	1	0	0	1
0	0	1	0	0	0	1
0	0	1	1	0	0	1
0	1	0	0	1	0	0
0	1	0	1	0	1	0
0	1	1	0	0	0	1
0	1	1	1	0	0	1
1	0	0	0	1	0	0
1	0	0	1	1	0	0
1	0	1	0	0	1	0
1	0	1	1	0	0	1
1	1	0	0	1	0	0
1	1	0	1	1	0	0
1	1	1	0	1	0	0
1	1	1	1	0	1	0

Combinational Digital Circuits

Parity Generator (74280 (9-bit)): a combinational logic circuit that generates the parity bit in the transmitter. The other circuit that checks the parity in the receiver is called Parity Checker.

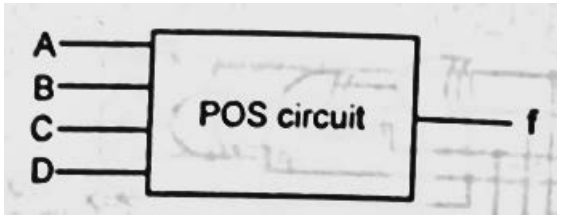
CD / AB	00	01	11	10
00	0		0	
01		0		0
11	0		0	
10		0		0

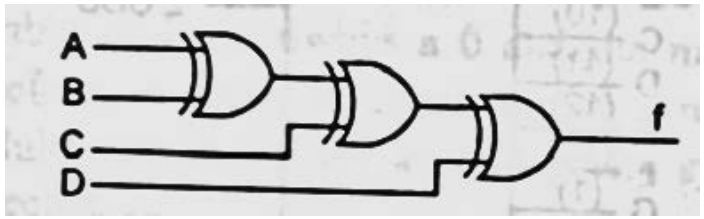
K-map for f' (POS)

A	B	C	D	f (Even-parity)
0	0	0	0	0
0	0	0	1	1
0	0	1	0	1
0	0	1	1	0
0	1	0	0	1
0	1	0	1	0
0	1	1	0	0
0	1	1	1	1
1	0	0	0	1
1	0	0	1	0
1	0	1	0	0
1	0	1	1	1
1	1	0	0	0
1	1	0	1	1
1	1	1	1	0

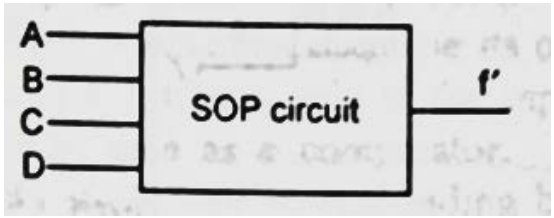
Combinational Digital Circuits

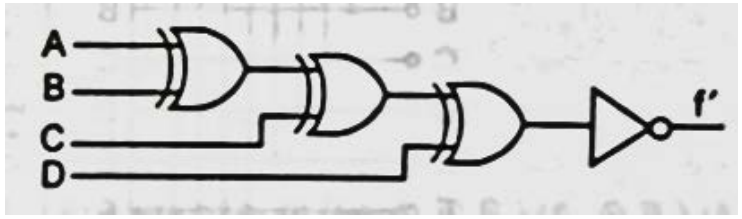
$$f(A,B,C,D)$$
$$= (A+B+C+D)(A+B+\bar{C}+\bar{D})(A+\bar{B}+C+\bar{D})(A+\bar{B}+\bar{C}+D)(\bar{A}+B+C+\bar{D})(\bar{A}+B+\bar{C}+D)(\bar{A}+\bar{B}+C+D)(\bar{A}+B+C+\bar{D})(\bar{A}+B+\bar{C}+D)$$
$$= A \oplus B \oplus C \oplus D$$





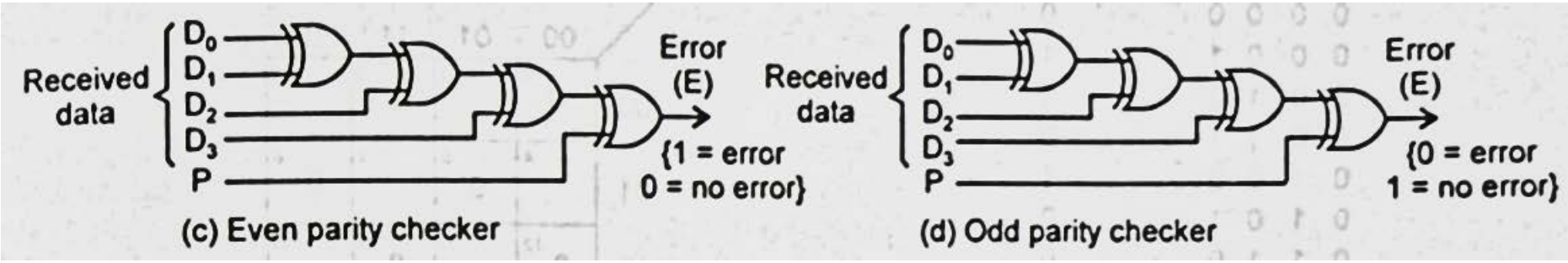
In SOP form:





Combinational Digital Circuits

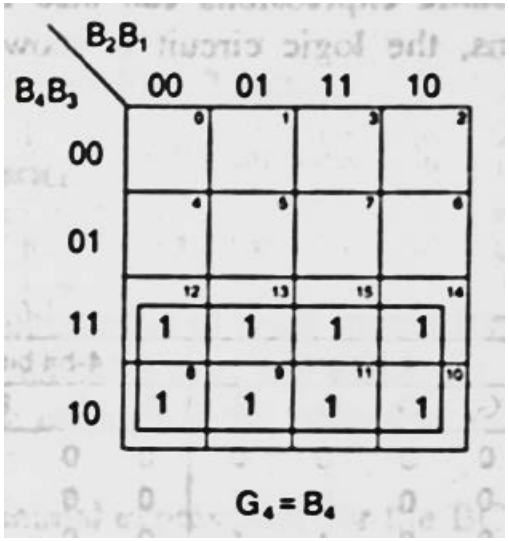
Parity Checker: a combinational logic circuit for checking the parity bit of the received codeword.

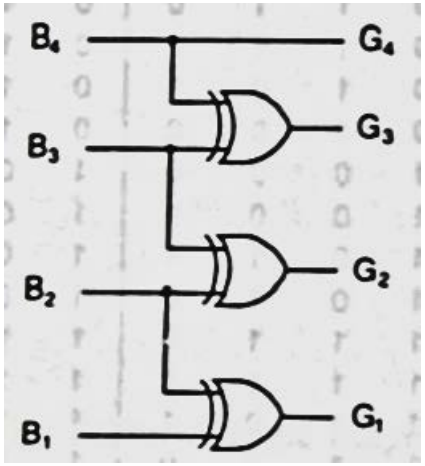


Combinational Digital Circuits

Code converter: a combinational logic circuit for converting input codeword into output codeword, eg. Binary-to-gray, gray-to-binary, binary-to-BCD, BCD-to-binary, BCD-to-excess3, excess3-to-BCD, etc.

4-bit binary				4-bit gray			
B_4	B_3	B_2	B_1	G_4	G_3	G_2	G_1
0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	1
0	0	1	0	0	0	1	1
0	0	1	1	0	0	1	0
0	1	0	0	0	1	1	0
0	1	0	1	0	1	1	1
0	1	1	0	0	1	0	1
0	1	1	1	0	1	0	0
1	0	0	0	1	1	0	0
1	0	0	1	1	1	0	1
1	0	1	0	1	1	1	1
1	0	1	1	1	1	1	0
1	1	0	0	1	0	1	0
1	1	0	1	1	0	1	1
1	1	1	0	1	0	0	1
1	1	1	1	1	0	0	0





$$G_4 = B_4$$
$$G_3 = \overline{B_4}B_3 + B_4\overline{B_3} = B_4 \oplus B_3$$
$$G_2 = \overline{B_3}B_2 + B_3\overline{B_2} = B_3 \oplus B_2$$
$$G_1 = \overline{B_2}B_1 + B_2\overline{B_1} = B_2 \oplus B_1$$

Combinational Digital Circuits

Priority encoder (74148 8-to-3 line priority encoder):

The priority encoder is a circuit that executes the priority function.

The logic of the priority encoder is such that **when two or more inputs appear simultaneously**, the **input having the largest priority** will take precedence.

Inputs				Outputs			Boolean Function
I_0	I_1	I_2	I_3	X	Y	IST (Interrupt Status)	
1	X	X	X	0	0	1	
0	1	X	X	0	1	1	$X = \overline{I_0}\overline{I_1}$
0	0	1	X	1	0	1	$X = \overline{I_0}\overline{I_1} + \overline{I_0}\overline{I_2}$
0	0	0	1	1	1	1	$(IST) = I_0 + I_1 + I_2 + I_3$
0	0	0	0	X	X	0	

The truth table of a four-input priority encoder is given in the table. The X's in the table designate don't care conditions. **Input I_0 has the largest priority**, so indifferent of the values of other inputs when this is input is 1, the output creates an output $XY = 00$.

Combinational Digital Circuits

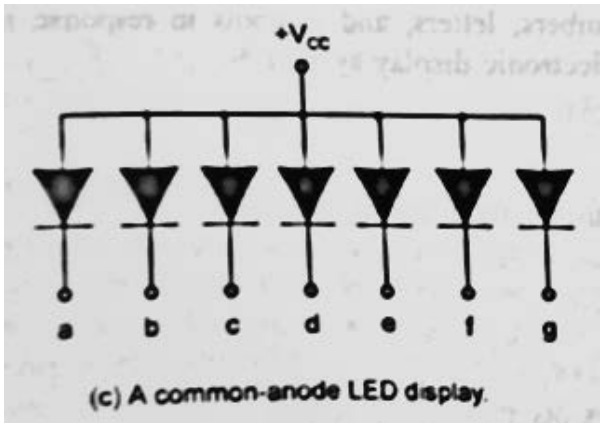
- **I_1 has the next priority level.** The output is **01 if I_1=1** supported that **I_0=0,** regardless of the values of the other two lower-priority inputs.
- The output for **I_2** is generated only if higher-priority inputs are 0, etc. down the priority level.
- **The interrupt status IST is set only when one or more inputs are equal to 1.**
- If all inputs are 0, IST is cleared to 0 and the other outputs of the encoder are not used, so they are signified with don't care condition.
- This is because the vector address is not shared with the CPU when IST=0.
- The Boolean function showed in the table determines the internal logic of the encoder.

Generally, a computer will have more than four interrupt sources. A priority encoder with eight inputs, for example, will create an output of three bits.
The output of the priority encoder can form part of the vector address for each interrupt source. The other bits of the vector address can be created any value. For instance, the vector address can be formed by joining six zeroes to the X and Y outputs of the encoder.
With this choice, the interrupt vector for the four I/O devices is created with numbers 0, 1, 2, and 3.

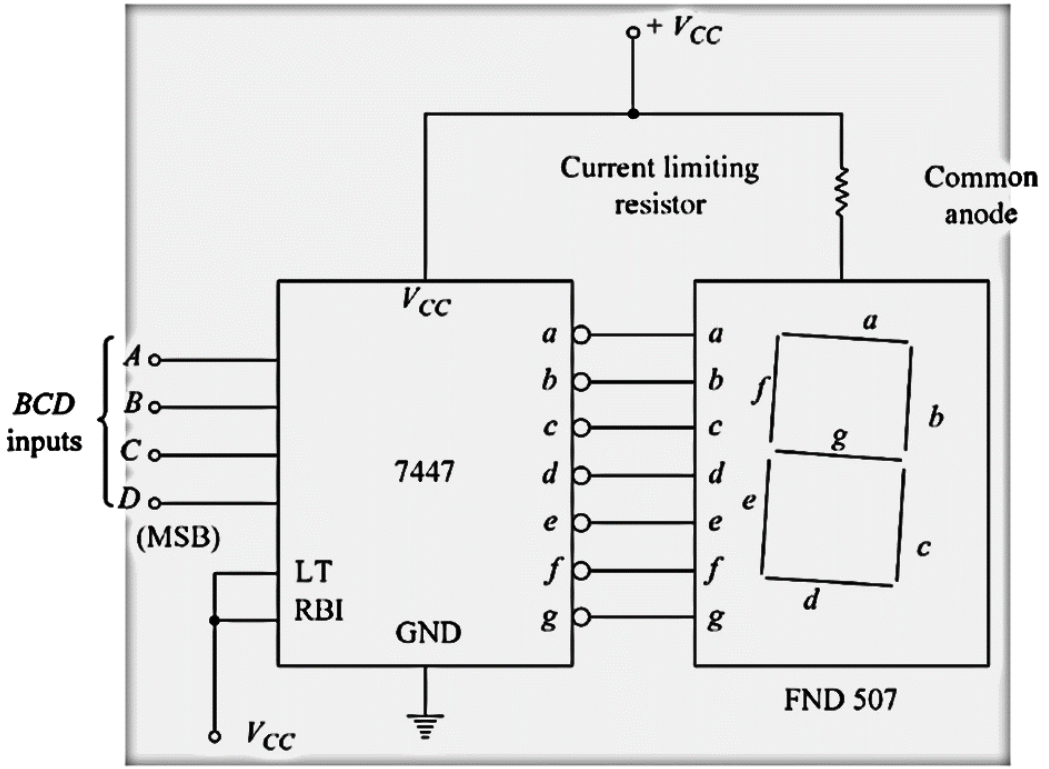
Combinational Digital Circuits

Decoders/drivers for display devices

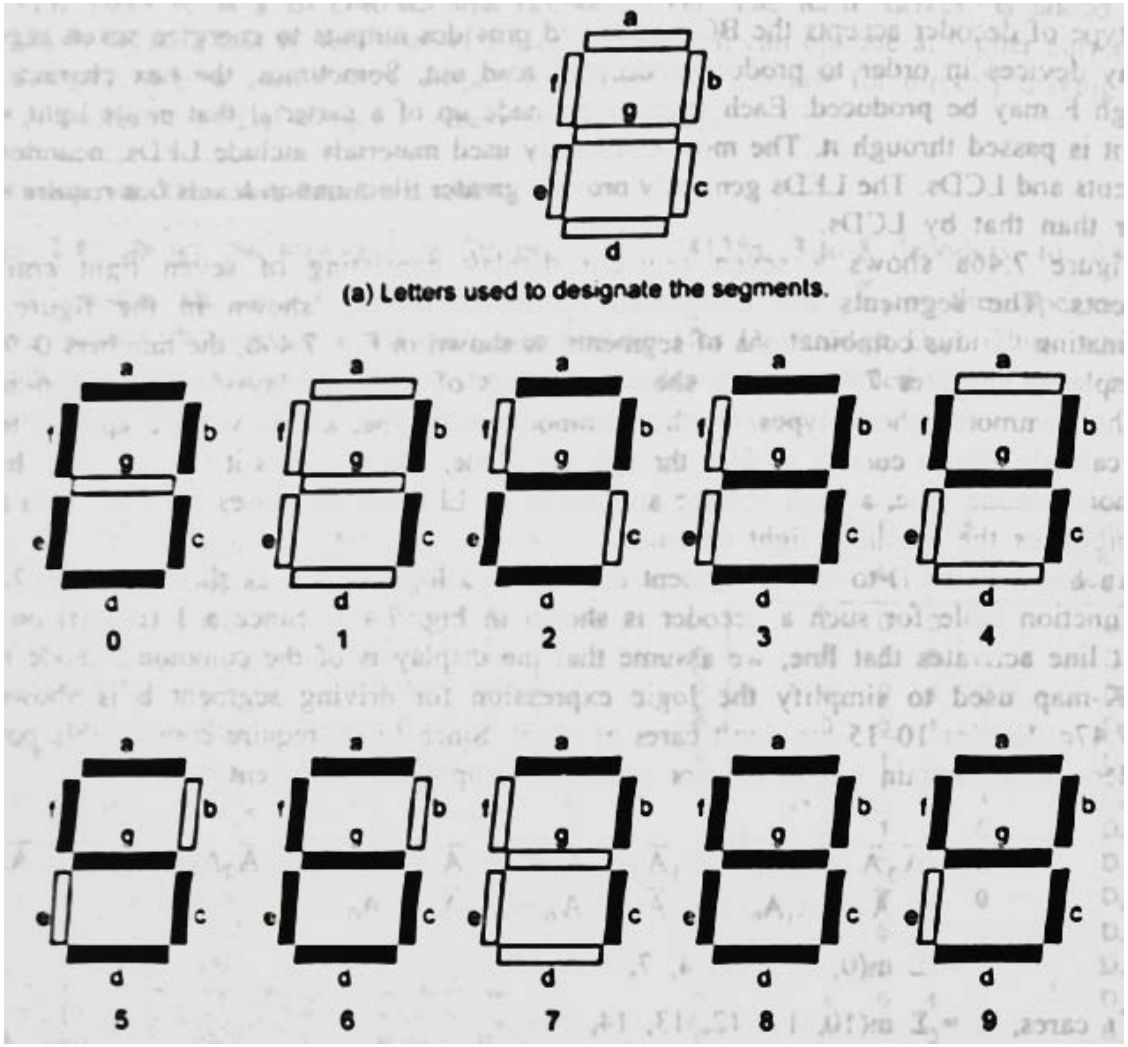
Used in display devices to view the output, e.g. indicator/relay driver, nixie tube driver, LED output, seven-segment display, etc.



(c) A common-anode LED display.



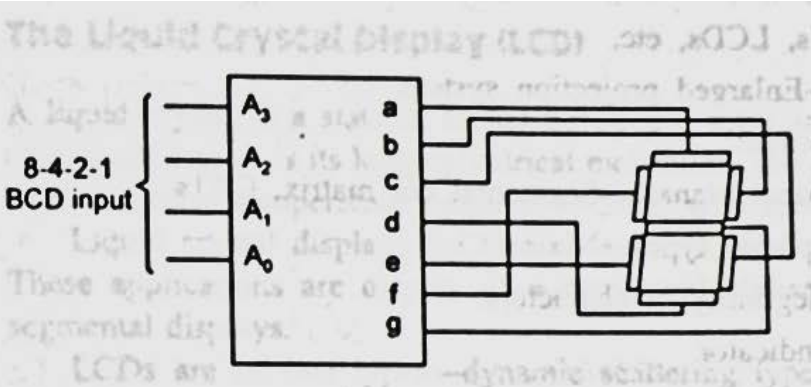
Combinational Digital Circuits



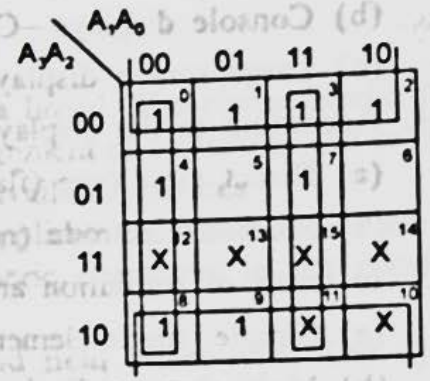
Combinational Digital Circuits

Decimal digit	8-4-2-1 BCD				Seven segment code						
	A_3	A_2	A_1	A_0	a	b	c	d	e	f	g
0	0	0	0	0	1	1	1	1	1	1	0
1	0	0	0	1	0	1	1	0	0	0	0
2	0	0	1	0	1	1	0	1	1	0	1
3	0	0	1	1	1	1	1	1	0	0	1
4	0	1	0	0	0	1	1	0	0	1	1
5	0	1	0	1	1	0	1	1	0	1	1
6	0	1	1	0	1	0	1	1	1	1	1
7	0	1	1	1	1	1	1	0	0	0	0
8	1	0	0	0	1	1	1	1	1	1	1
9	1	0	0	1	1	1	1	1	0	1	1

(b) Function table



(a) Logic circuit



$b = \bar{A}_2 + \bar{A}_1\bar{A}_0 + A_1A_0$

(c) K-map to derive simplified expression for driving segment (b)