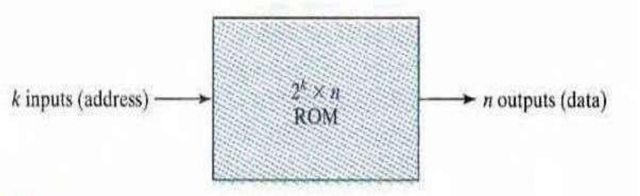
EC 8392 – DIGITAL ELECTRONICS

UNIT – V : MEMORY DEVICES AND DIGITAL INTEGRATED CIRCUITS

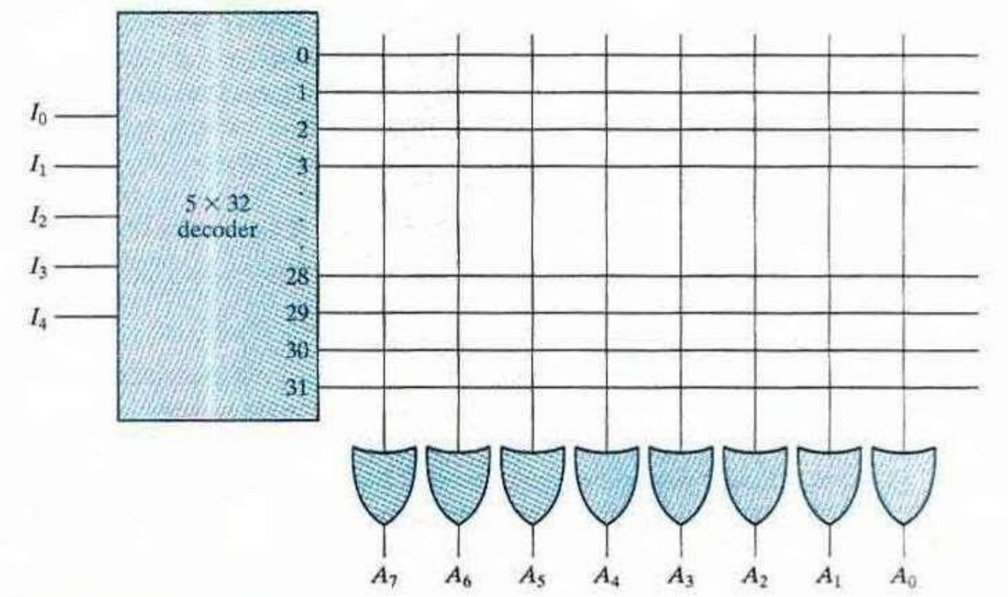
ROM

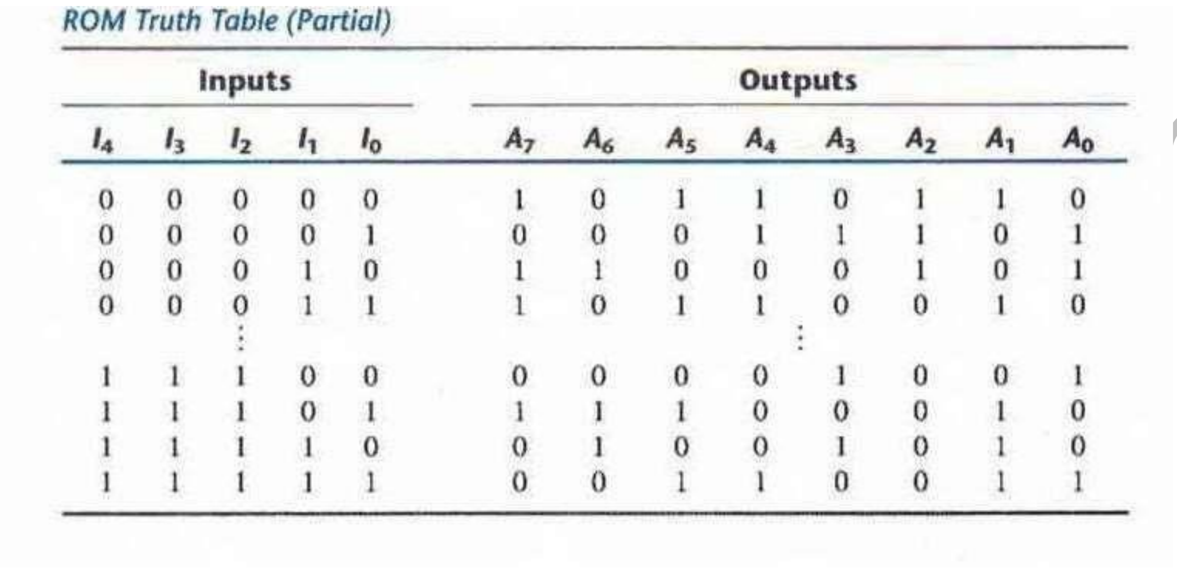
A ROM is essentially a memory device in which permanent binary information is stored. The binary information must be speified by the designer and is then embedded in the unit to form the required interconnection pattern. Once the pattern is established, it stays within the unit even when power is turned off and on again.



The inputs provide the address for memory and the outputs give the data bits of the stored word that is selected by the address. The number of words in a ROM is determined from the fact that k address input lines are needed to specify 2^k words. Note that ROM does not have data inputs, because it does not have a write operation. Integrated circuit ROM chips have one or more enable inputs and sometimes come with three -state outputs to facilitate the construction of large arrays of ROM.

Consider, for example, a 32 X 8 ROM. The unit consists of 32 word s of 8 bits each. There are five input lines that form the binary number s from 0 through 31 for the address. The below figure shows the internal logic construction of this ROM. The five inputs are decoded into 32 distinct outputs by means of a 5 X 32 decoder. Each output of the decoder represents a memory address.

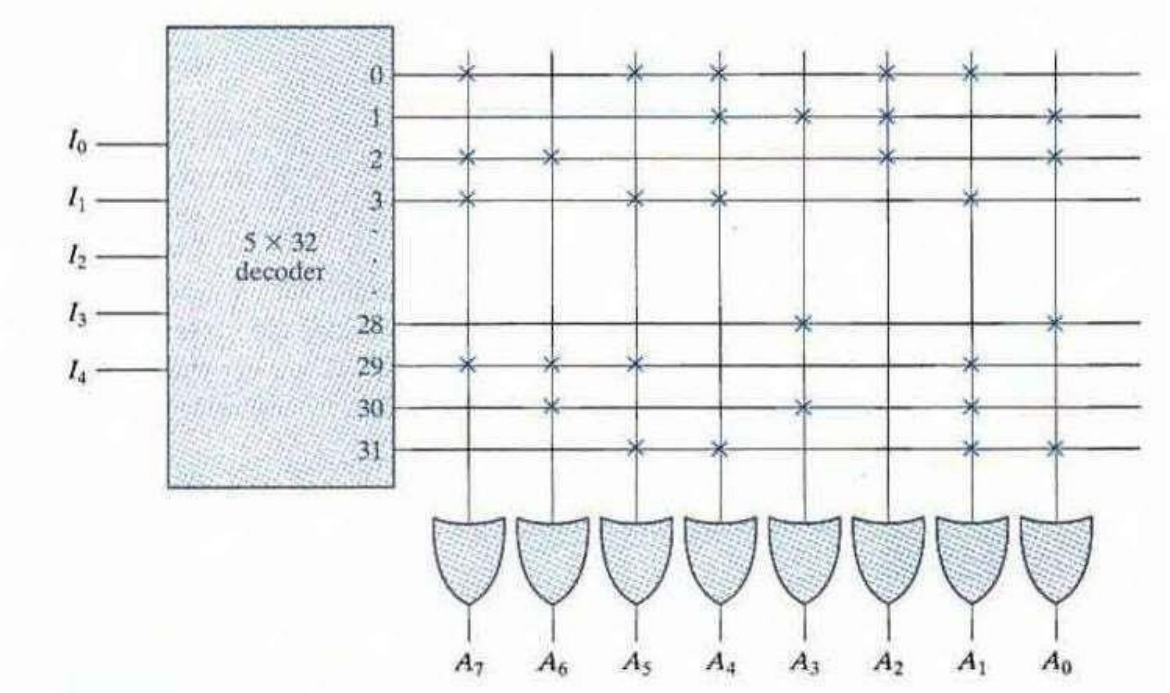




ROM Truth Table (Partial)

Inputs					Outputs							
I_4	I_3	I_2	I_1	I_0	A_7	A_6	A_5	A_4	A_3	A_2	A_1	A_0
0	0	0	0	0	1	0	1	1	0	1	1	0
0	0	0	0	1	0	0	0	1	1	1	0	1
0	0	0	1	0	1	1	0	0	0	1	0	1
0	0	0	1	1	1	0	1	1	0	0	1	0
		⋮							⋮			
1	1	1	0	0	0	0	0	0	1	0	0	1
1	1	1	0	1	1	1	1	0	0	0	1	0
1	1	1	1	0	0	1	0	0	1	0	1	0
1	1	1	1	1	0	0	1	1	0	0	1	1

The 32 outputs of the decoder are connected to each of the eight OR gates. The diagram shows the array logic convention used in complex circuits. Each OR gate must be considered as having 32 inputs. Each output of the decoder is connected to one of the inputs of each OR gate. Since each OR gate has 32 input connections and there are 8 OR gates, the ROM contains 32 x 8 = 256 internal connections.



Types of ROM

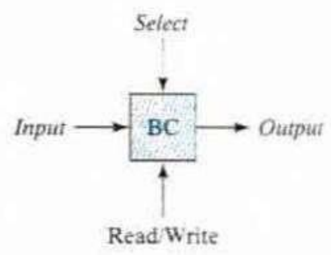
The required paths in a ROM may be programmed in four different ways. The first is called mask programming and is done by the semiconductor company during the last fabrication process of the unit. The procedure for fabricating a ROM requires that the customer fill out the truth table he or she wishes the ROM to satisfy. Th e truth table may be submitted in a special form provided by the manufacturer or in a specified form at on a computer output medium. The manufacturer makes the corresponding mask for the path s to produce the 1's and D's according to the customer's truth table. This procedure is costly because the vendor charges the customer a special fee for custom masking the particular ROM. For this reason, mask programming is economical only if a large quantity of the same ROM configuration is to be ordered.

For small quantities, it is more economical to use a second type of ROM called programmable read-only memory, or PROM. When ordered, PROM units contain all the fuses intact, giving all ls in the bits of the stored words. The fuses in the PROM are blown by the application of a high-voltage pulse to the device through a special pin. A blown fuse defines a binary 0 state and an intact fuse gives a binary 1 state. This procedure allows the user to program The PROM in the laboratory to achieve the desired relationship between input addresses and stored words. Special instruments called PROM programmers are available commercially to facilitate the procedure. In any case, all procedures for programming ROMs are hardware procedures, even though the word programming is used. The hardware procedure for programming ROMs or PROMs is irreversible and once programed, the fixed pattern is permanent and cannot be altered. Once a bit pattern has been established, the unit must be discarded if the bit pattern is to be changed. A third type of ROM is the erasable PROM or EPROM, which can be restructured to the initial state even though it has been programmed previously. When the EPROM is placed under a special ultraviolet light for a given length of time, the shortwave radiation discharges the internal floating gates that serve as the programmed connections. After erasure, the EPROM returns to its initial state and can be reprogrammed to a new set of values.

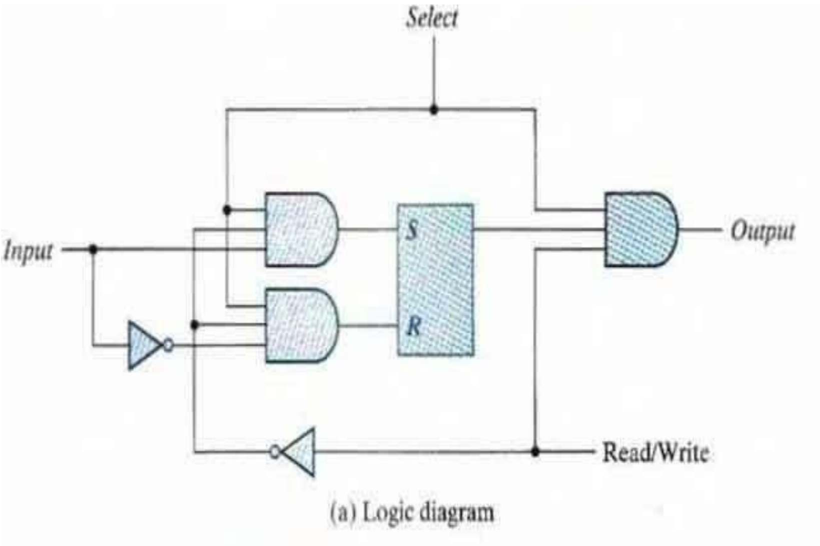
The fourth type of ROM is the electrically erasable PROM (EEPROM). This device is like the EPROM, except that the previously programmed connections can be erased with an electrical signal instead of ultraviolet light. The advantage is that the device can be erased without removing it from its socket.

MEMORY DECODING

The internal construction of a RAM of m words and n bits per word consists of m X n binary storage cells and associated decoding circuits for selecting individual words. The binary storage cell is the basic building block of a memory unit. The equivalent logic of a binary cell that stores one bit of information is shown in below figure. The storage part of the cell is modeled by an SR latch with associated gates to form a D latch. Actually, the cell is an electronic circuit with four to six transistors. Nevertheless, it is possible and convenient to model it in terms of logic symbols.

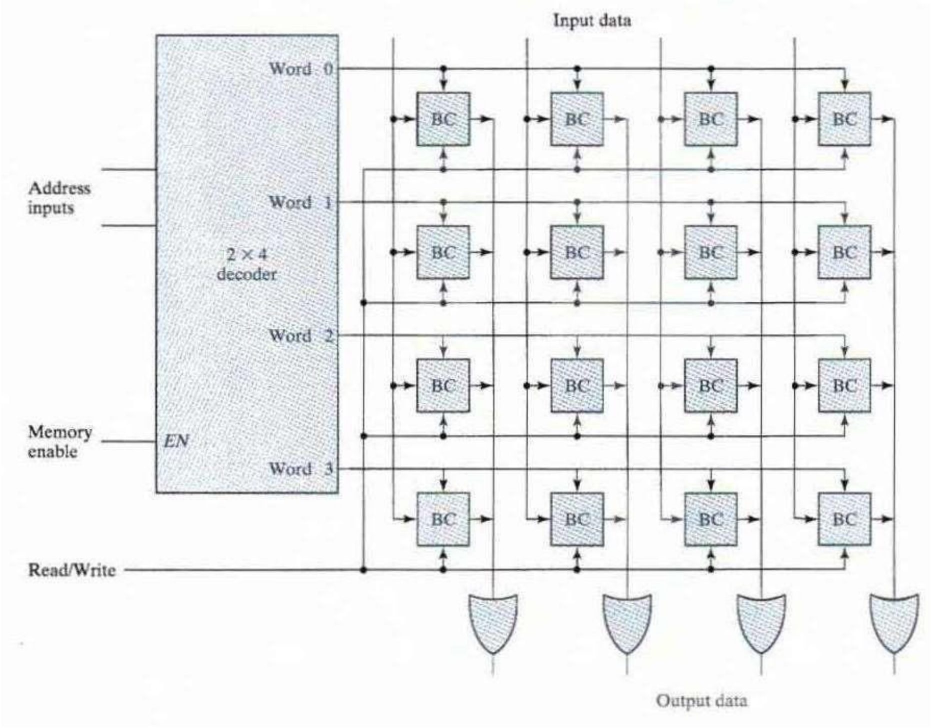


(b) Block diagram



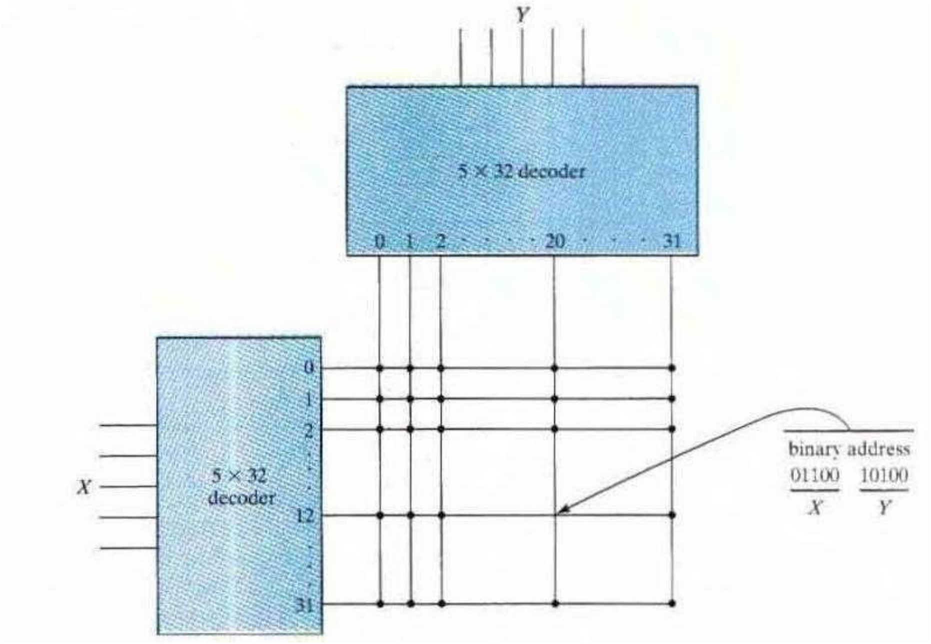
(a) Logic diagram

A binary storage cell must be very small in order to be able to pack as many cells as possible in the small area available in the integrated circuit chip. The binary cell stores one bit in its internal latch. The select input enables the cell for reading or writing and the read/write input determines the operation of the cell when it is selected. A 1 in the read/write input provides the read operation by forming a path from the latch to the output terminal. A 0 in the read/write input provides the write operation by forming a path from the input terminal to the latch. A memory with 2^k words of n bits per word requires k address lines that go into a k X 2^kdecoder. Each one of the decoder outputs selects one word of n bits for read ing or writing.



Coincident Decoding

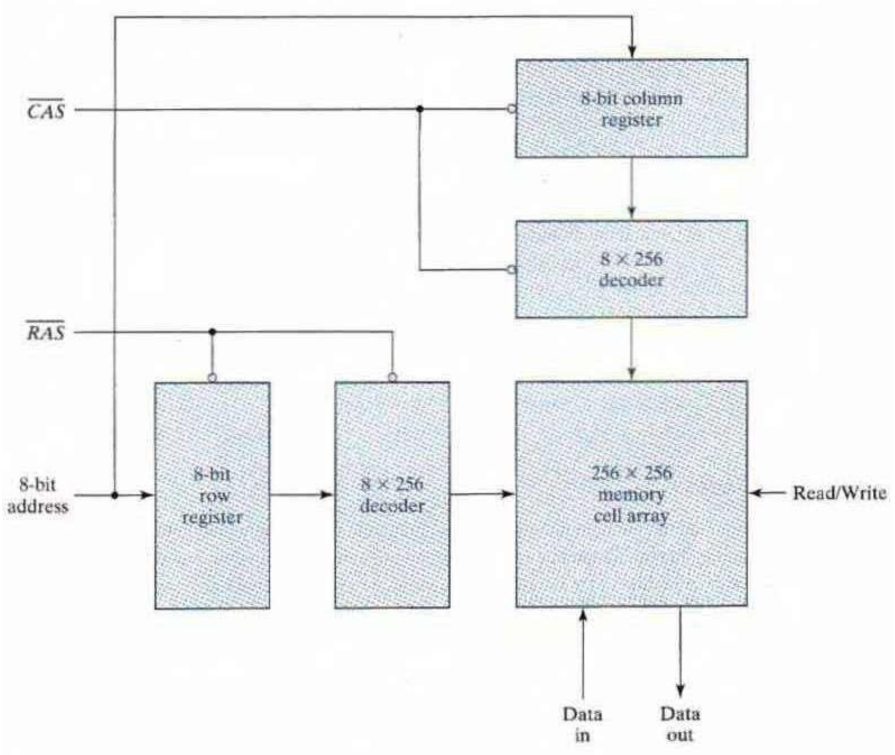
A decoder with k inputs and 2^k outputs requires 2^k AND gates with k inputs per gate. The total number of gates and the number of inputs per gate can hereduced by employing two decoders in a two-dimensional selection scheme. The basic idea in two-dimensional decoding is to arrange the memory cells in an array that is close as possible to square. In this configuration, two k/2-input decoders are used instead of one k-input decoder. One decoder performs the row selection and the other the column selection in a two-dimensional matrix configuration



Address Multiplexing

The SRAM memory cell typically contains six transistors. In order to build memories with higher density, it is necessary to reduce the number of transistors in a cell. The DRAM cell contains a single MOS transistor and a capacitor. The charge stored on the capacitor discharges with line, and the memory cells must be periodically recharged by refreshing the memory. Because of their simple cell structure, DRAMs typically have four times the density of SRAMs. This allows four times as much memory capacity to be placed on a given size of chip.

The memory consists of a two-dimensional array of cells arranged into 256 rows by 256 columns for a total of 2^8 x 2^8 = 216 = 64K words. There is a single data input line, a single data output line, and a read/write control as well as an eight-bit address input and two address, strobes, the latter included for enabling the row and column address into their respective registers. The row address strobe (RAS) enables the eight-bit row register and the column address strobe (CAS) enables the eight-bit column register. The bar on top of the name of the strobe symbol indicates that the registers are enabled on the zero level of the signal.


$\overline{CAS}$
$\overline{RAS}$
8-bit address
8-bit column register
8 × 256 decoder
8-bit row register
8 × 256 decoder
256 × 256 memory cell array
Read/Write
Data in
Data out

EC 8392 – DIGITAL ELECTRONICS

UNIT – V : MEMORY DEVICES AND DIGITAL INTEGRATED CIRCUITS

Digital integrated circuits

TTL Family

Introduction

Transistor-Transistor Logic (TTL) and Emitter Coupled Logic (ECL) are the most commonly used bipolar logic families. Bipolar logic families use semiconductor diodes and bipolar junction transistors as the basic building blocks of logic circuits. Simplest bipolar logic elements use diodes and resistors to perform logic operation; this is called diode logic. Many TTL logic gates use diode logic internally, and boost their output drive capability using transistor circuits. Other TTL gates use parallel configurations of transistors to perform logic functions. It turned out at the time of introducing TTL circuits that they were adaptable to virtually all forms of IC logic and produced the highest performance-to-cost ratio of all logic types. In view of its versatility a variety of subfamilies (Low Power, High Frequency, Schottky) representing a wide range of speed-power product have also been introduced. The Schottky family has been selected by the industry to further enhance the speed-power product. In Schottky family circuits, a Schottky diode is used as a clamp across the base-collector junction of a transistor to prevent it from going into saturation, thereby reducing the storage time. Several sub-families have evolved in the Schottky TTL family to offer several speed-power products to meet a wide variety of design requirements. These sub-families are:

- Low-power Schottky TTL (LSTTL)
- Fairchild Advanced Schottky TTL (FAST)
- Advanced Low Power Schottky TTL (ALSTTL)
- Advanced Schottky TTL (ASTTL)

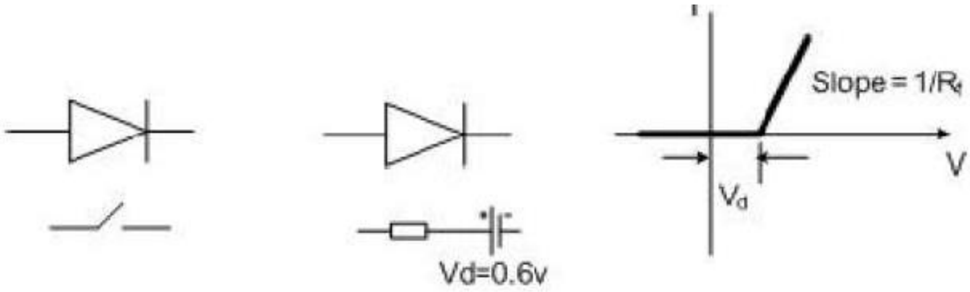
We will explore the characteristics of the TTL family in this Learning Unit.

Diodes

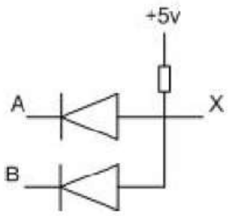
A semiconductor diode is fabricated from two types, p-type and n-type, of semiconductor material that are brought into contact with each other. The point of contact between the p and n materials is called p-n junction. Actually, a diode is fabricated from a single monolithic crystal of semiconductor material in which the two halves are doped with different impurities to give them p-type and n-type properties. A real diode can be modelled as shown in the figure 1.

- It is an open circuit when it is reverse biased (we ignore its leakage current)
- It acts like a small resistance, Rf, called the forward resistance, in series with Vd, called a diode drop, a small voltage source.

- The forward diode drop would be about 0.6 V and Rf is about 25 Ω.

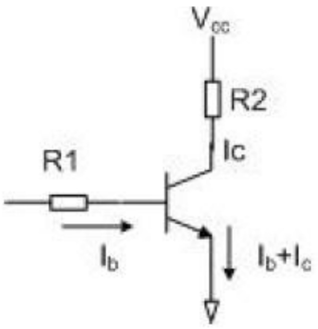


Diode action is exploited to perform logical operations. The circuit shown in the figure 2 performs AND function if 0-2 V (Low) input is considered logic 0 and 3-5 V (High) input is considered as logic 1. When both A and B inputs are High, the output X will be High. If any one of the inputs is at Low level, the output will also be at Low level.



Bipolar Junction Transistor

A bipolar junction transistor is a three terminal device and acts like a current-controlled switch. If a small current is injected into the base, the switch is “on”, that is, the current will flow between the other two terminals, namely, collector and emitter. If no current is put into the base, then the switch is “off” and no current flows between the emitter and the collector. A transistor will have two p-n junctions, and consequently it could be pnp transistor or npn transistor. An npn transistor, found more commonly in IC logic circuits, is shown in the figure 3 in its common-emitter configuration.



The relations between different quantities are given as in the following: Ib =

(VIN - 0.6)/R1

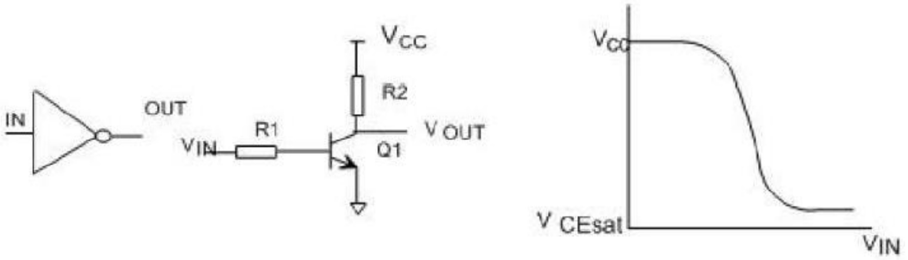
IC = β . Ib

VCE = VCC - IC . R2

= VCC - β . Ib . R2

= VCC - β(VIN - 0.6).R2/R1

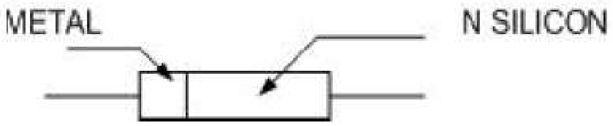
where β is called the gain of the transistor and is in the range of 10 to 100 for typical transistors. Figure 4 shows a logic inverter from an npn transistor in the common-emitter configuration. When the input voltage VIN Low, the output voltage is High, and vice versa.



When the input of a saturated transistor is changed, the output does not change immediately; it takes extra time, called storage time, to come out of saturation. In fact, storage time accounts for a significant portion of the propagation delay in the earlier TTL families. Present day TTL logic families reduce this storage time by placing a Schottky diode between the base and collector of each transistor that might saturate.

Schottky Barrier Diode

A Schottky Barrier Diode (SBD) is illustrated in figure 5. It is a rectifying metalsemiconductor contact formed between a metal and highly doped N semiconductor.

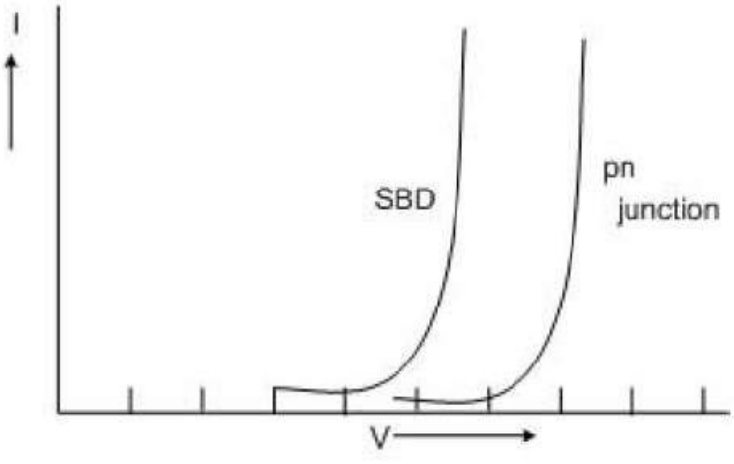


The valence and conduction bands in a metal overlap making available a large number of free-energy states. The free-energy states can be filled by any electrons which are injected into the conduction band. A finite number of electrons exist in the conduction band of a semiconductor.

The number of electrons depends mainly upon the thermal energy and the level of impurity atoms in the material. When a metal-semiconductor junction is formed, free electrons flow across the junction from the semiconductor, via the conduction band, and fill the free-energy states in the metal. This flow of electrons builds a depletion potential across the barrier. This depletion potential opposes the electron flow and, eventually, is sufficient to sustain a balance where there is no net electron flow across the barrier. Under the forward bias (metal positive), there are many electrons with enough thermal energy to cross the barrier potential into the metal. This forward bias is called "hot injection." Because the barrier width is decreased as forward bias VF increases, forward current will increase rapidly with an increase in VF.

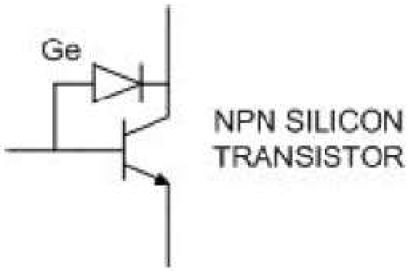
When the SBD is reverse biased, electrons in the semiconductor require greater energy to cross the barrier. However, electrons in the metal see a barrier potential from the side essentially independent of the bias voltage and small net reverse current will flow. Since this current flow is relatively independent of the applied reverse bias, the reverse current flow will not increase significantly until avalanche breakdown occurs. A simple metal/nsemiconductor collector contact is an ohmic contact while the SBD contact is a rectifying contact. The difference is controlled by the level of doping in the semiconductor material. Current in SBD is carried by majority carriers. Current in a p-n junction is carried by minority carriers and the resultant minority carrier storage causes the switching time of a pn junction to be limited when switched from forward bias to reverse bias.

A p-n junction is inherently slower than an SBD even when doped with gold. Another major difference between the SBD and p-n junction is forward voltage drop. For diodes of the same surface area, the SBD will have a larger forward current at the same forward bias regardless of the type of metal used. The SBD forward voltage drop is lower at a given current than a p-n junction. Figure 6 illustrates the forward current-voltage characteristic differences between the SBD and p-n junction.

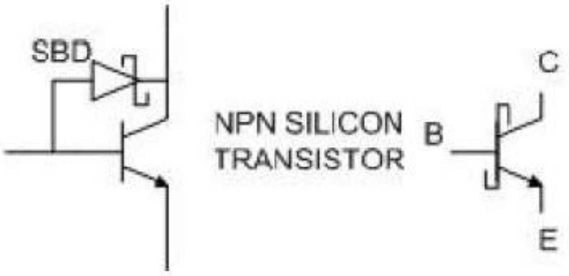


Schottky Transistor

The Schottky transistor makes use of two earlier concepts: Baker clamp and the SchottkyBarrier-Diode (SBD). The Schottky clamped transistor is responsible for increasing the switching speed. The use of Baker Clamp, shown in the figure 7, is a method of avoiding saturation of a discrete transistor.



The germanium diode forward voltage is 0.3 V to 0.4 V as compared to 0.7 V for the baseemitter junction silicon diode. When the transistor is turned on, base current drives the transistor toward saturation. The collector voltage drops, the germanium diode begins to conduct forward current, and excess base drive is diverted from the base-collector junction of the transistor. This causes the transistor to be held out of deep saturation, the excess base charge not stored, and the turn-off time to be dramatically reduced. However, a germanium diode cannot be incorporated into a monolithic silicon integrated circuit. Therefore, the germanium diode must be replaced with a silicon diode which has a lower forward voltage drop than the base-collector junction of the transistor. A normal p-n diode will not meet this requirement. An SBD can be used to meet the requirement as shown in the figure 8.



The SBD meets the requirements of a silicon diode which will clamp a silicon npn transistor out of saturation.

BASIC NAND GATE

The familiarization with a logic family is acquired, in general, through understanding the circuit features of a NAND gate. The circuit diagram of a two-input LSTTL NAND gate, 74LS00, is shown in the figure 9. D1 and D2 along with 18 KΩ resistor perform the AND function. Diodes D3 and D4 do nothing in normal operation, but limit undesirable negative excursions on the inputs to a signal diode drop. Such negative excursions may occur on High- to-Low input transitions as a result of transmission-line effects. Transistor Q1 serves as an inverter, so the output at its collector represents the NAND function. It also, along with its resistors, forms a phase splitter that controls the output stage. The output state has two transistors, Q3 and Q4, only one of which is on at any time. The TTL output state is sometimes called a totem-pole output. Q2 and Q5 provide active pull-up and pull-down to the High and Low states, respectively. Transistor Q5 regulates current flow into the base of Q4 and aids in turning Q4 off rapidly. Transistors Q3 and Q2 constitute a Darlington driver, with Q3 not being permitted to saturate. The network consisting of Schottky diodes D3 and D4 and a 5 KΩ resistor is connected to the output and aids in charging and discharging load capacitance when Q3 and Q4 are changing states. Transistor Q4 conducts when the output is in Low state.

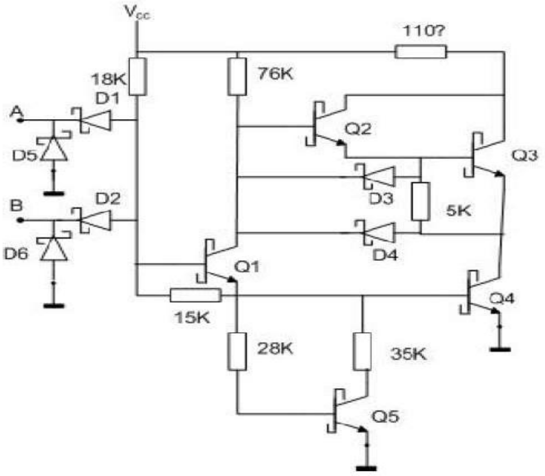
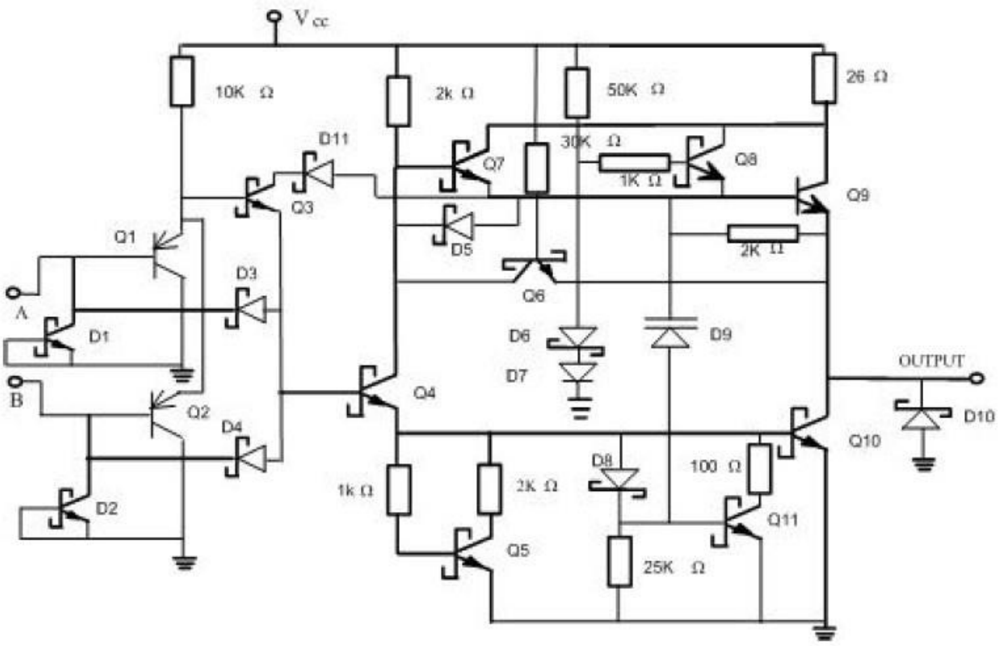


Figure 11 shows one gate in 74ALS00A quad 2-input NAND gate parallel-connected pnp transistors Q1 and Q2 are used at the input. These transistors reduce the current flow, IR, when the inputs are low and thus increase fan out. If inputs A, B, or both are low, then the respective pnp transistors turn on because their emitters are then more positive than their bases. If at least one of the inputs is low, the corresponding pnp transistor conducts, making the base of Q3 low and keeping Q3 off. If both the inputs A and B are high, both switches are open and Q3 turns on. Q3 drives Q4 (by emitter follower action), and Q4 drives the output totem pole. Schottky diodes D3, D4 and D5 are used to speed the switching and do not affect the logic. Note that the output and the inputs have Schottky protective diodes. Figure 12 shows one gate in 74AS00 gate.

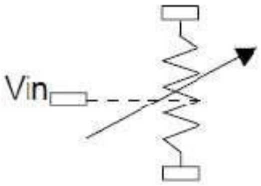


CMOS FAMILY

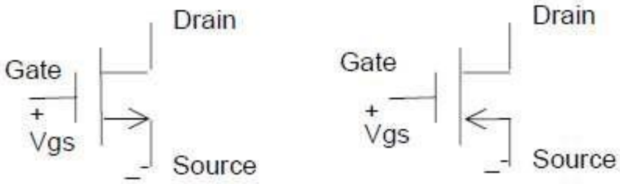
CMOS has often been called the ideal technology. It has low power dissipation, high noise immunity to power supply noise, symmetric switching characteristics and large supply voltage tolerance. Reducing power requirements leads to reduction in the cost of power supplies, simplifies power distribution, possible elimination of cooling fans and a denser PCB, ultimately leading to lower cost of the system. Though the operation of a MOS transmission was understood long before bipolar transistor was invented, its fabrication could not be monitored. Consequently development of MOS circuits lagged bipolar circuits considerably, and initially they were attractive only in selected applications. In recent years, advances in the design of MOS circuits have vastly increased their performance and popularity. By far majority of the large scale integrated circuits such as microprocessors and memories use CMOS. The usage of CMOS logic is increasing in applications that use small and medium scale integrated circuits as CMOS circuits, while offering functionality and speed similar to bipolar logic circuits, consume very much less power.

CMOS LOGIC CIRCUITS

The basic building blocks in CMOS logic circuits are MOS transistors. A MOS transistor can be received as a 3-terminal device that acts like a voltage-controlled resistance, as shown in the figure 1.



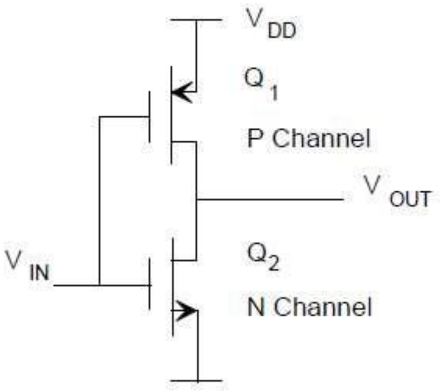
An input voltage applied to one terminal controls the resistance between the remaining two terminals. In digital applications, a MOS transistor is operated so its resistance is always either very high (and the transistor "off") or very low (and the transistor is always "on"). There are two types of MOS transistors n-channel and pchannel. The circuit symbols for NMOS and PMOS transistors are shown in the figure 2.



The terminals are called gate, source and drain. The voltage from gate to source (VGS) in NMOS device is normally zero or positive. If VGS = 0 then the resistance from drain to source (RDS) is very high, of the order of mega ohm or more. When VGS is made positive RDS can decrease to a very low value, of the order of 10 ohms. In the PMOS transistor VGS is normally zero or negative. If VGS is zero, then the resistance from source to drain (RDS) is very large, and when VGS is negative RDS can decrease to a very low value. The gate of a MOS transistor has very high impedance, as it is separated from the source and drain by an insulating material with a very high resistance. However, the gate voltage creates an electric field that enhances or retards the flow of current between source and drain. This is the "field effect" in a MOSFET. The high resistance between the gate and the other terminals keeps the gate current to values lower than a microampere irrespective of the gate voltage. This current is called "leakage current". The gate of a MOS transistor is capacitively coupled to the source and drain. In high speed circuits, the power needed to charge and discharge these capacitances on each input signal transition accounts for a non trivial portion of a circuit's power consumption.

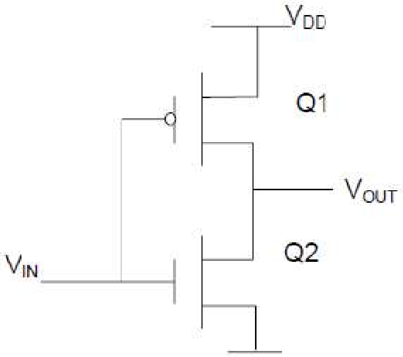
Basic CMOS Inverter circuit:

NMOS and PMOS transistors are used together in a complementary way to form CMOS logic, as shown in the figure 3. The power supply voltage VDD, typically is in the range of 2- 6 V, and is most often set at 5.0 V for compatibility with TTL circuits.



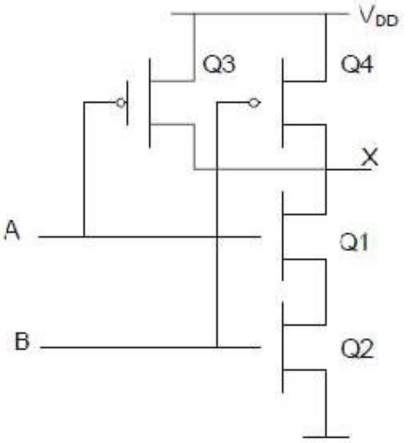
V_{IN}	Q_1	Q_2	V_{OUT}
0.0	On	Off	5V
5.0	Off	On	0V

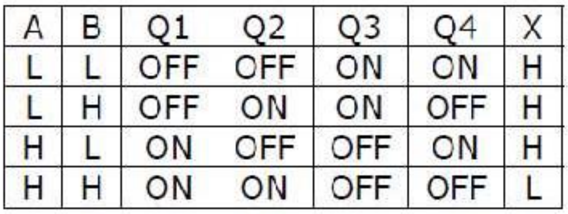
When VIN is at 0.0 V, the lower n-channel MOSFET Q1 is OFF since its VGS is 0, but the upper p-channel MOSFET Q2 is ON since its VGS would be -5.0 V. Consequently Q2 presents a small resistance while Q1 presents a large resistance. VOUT at the output terminal would be +5.0 V. Similarly when VIN is at 5.0 Q1 will be ON presenting a small resistance to ground while Q2 will be OFF presenting a large resistance. The output terminal voltage (VOUT) would be 0 V. Obviously this circuit behaves as an inverter. As we associated a logic state 0 or 1 with a voltage, we can say when the input signal is asserted Q1 is ON and Q2 is OFF, and when the input signal is not asserted Q1 is OFF and Q2 is ON. We make use of this interpretation to further simplify the circuit representation of MOSFETs, as shown in the figure 4. The bubble convention goes along with the convention followed in drawing logic diagrams.



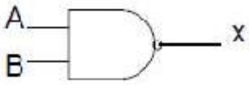
CMOS NAND and NOR gates:

Logic gates can be realised using CMOS circuits. A k-input gate uses k p-channel MOSFETs and k n-channel MOSFETs. Figure 5 shows a 2-input NAND gate. If either input is Low, the output X is High with low impedance connection to VDD through the corresponding p-channel transistor, and the path to the ground is blocked by the corresponding OFF n-channel MOSFET. If both inputs are High, the two n-channel MOSFETs are ON and the two p- channel MOSFETs are OFF. This is the operation required for the circuit to function as a NAND gate.

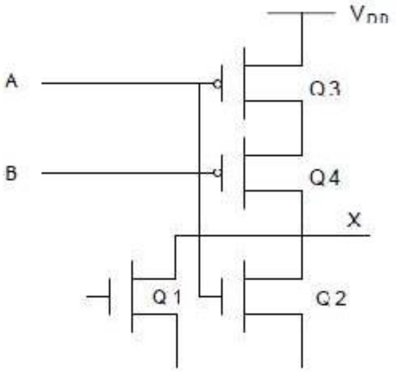


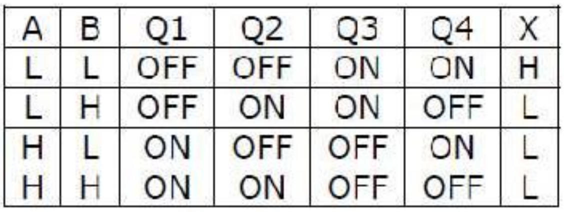


A	B	Q1	Q2	Q3	Q4	X
L	L	OFF	OFF	ON	ON	H
L	H	OFF	ON	ON	OFF	H
H	L	ON	OFF	OFF	ON	H
H	H	ON	ON	OFF	OFF	L

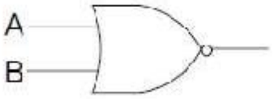


A 2-input NOR gate is shown in figure 6. Only when A and B are Low the output X is High and for all other combination of input levels the output is Low.





A	B	Q1	Q2	Q3	Q4	X
L	L	OFF	OFF	ON	ON	H
L	H	OFF	ON	ON	OFF	L
H	L	ON	OFF	OFF	ON	L
H	H	ON	ON	OFF	OFF	L

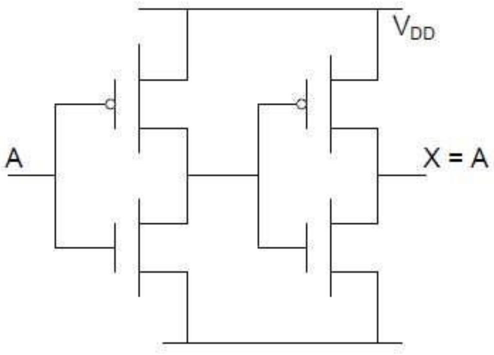


Non Inverting Gates:

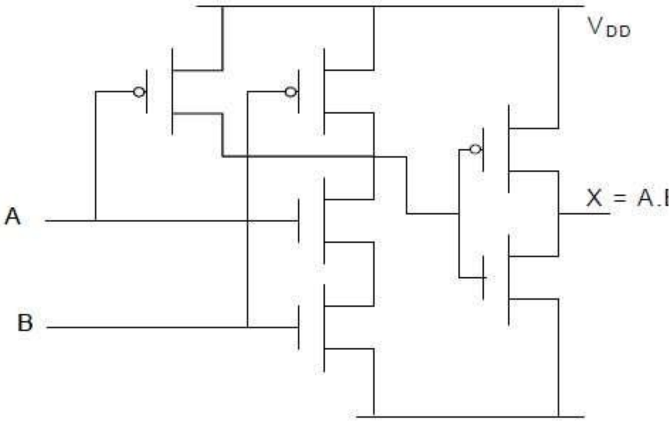
In all logic families, the simplest gates are inverters, and the next simplest are NAND and NOR gates. It is typically not possible to design a non-inverting gate with a smaller number of transistors than an inverting one. CMOS non-inverting buffers and AND and OR gates are obtained by connecting an inverter to the output of the corresponding inverting gate. Figure 7 shows a non inverting buffer and an AND gate

Buffering:

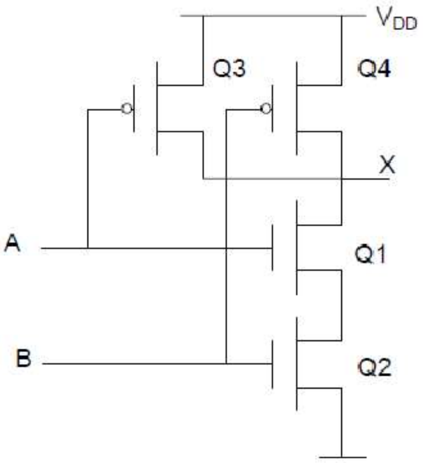
Most of the CMOS families are buffered. Buffering CMOS logic merely denotes designing the IC so that the output is taken from an inverting buffer stage. An unbuffered and buffered NAND gates are illustrated in the figure 8.

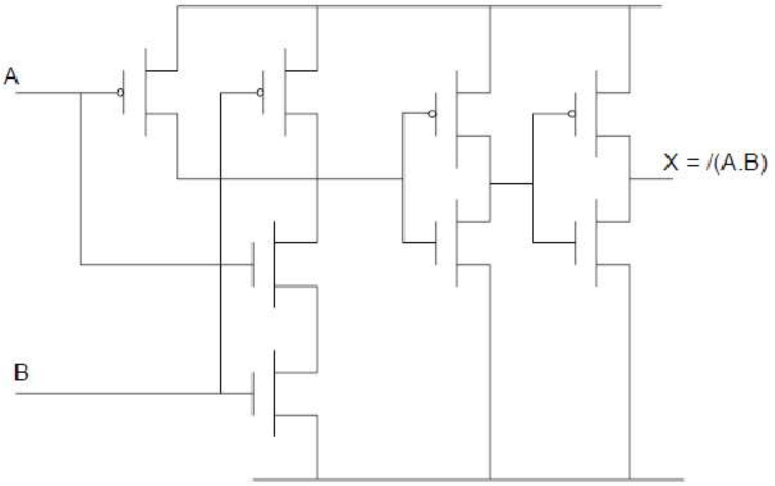


Buffer



AND gate

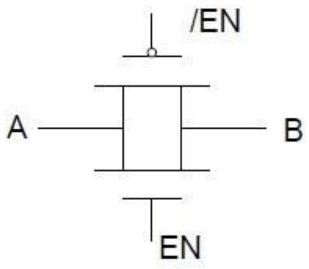




There are several advantages to buffering. By using the standardised buffer, the output characteristics of all devices are more easily made identical. Multistage gates will have better noise immunity due to their higher gain caused by having several stages from input to output. Also, the output impedance of an unbuffered gate may change with input logic level voltage and input logic combination, whereas buffered output are unaffected by input conditions. Single stage gates implemented would require large transistors due to the large output drive requirements. These large devices would have a large input capacitance associated A X = A VDD A B V DD X = A.B X VDD Q2 Q1 Q3 Q4 A B A B X = /(A.B) with them. This would affect the speed of circuits driving into an unbuffered gate, especially when driving large fan outs. Buffered gates have small input transistors and correspondingly small input capacitances. One may think that a major disadvantage of buffered circuits would be speed loss. It would seem that a two or three stage gate would be two to three times slower than a buffered one. However, internal stages are much faster than the output stage and speed lost by buffering is relatively small.

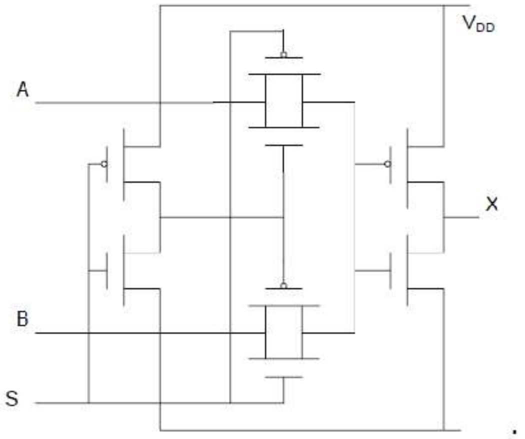
Transmission Gates:

A p-channel and n-channel transistor pair can be used as a logic-controlled switch. This circuit, shown in the figure 9, is called a CMOS transmission gate.



A transmission gate is operated so that its input signals EN and /EN are always at opposite levels. When EN is High and /EN is Low, there is a low impedance connection (as low as 5 Ω) between points A and B. When EN is Low and /EN is High, points A and B are disconnected. Once transmission gate is enabled, the propagation delay from A to B (or vice versa) is very short. Because of their short delays and conceptual simplicity, transmission gates are often used internally in larger-scale CMOS devices such as multiplexers and flip-flops. For example, figure 10 shows how transmission gates can be used to create a 2-input multiplexer

When S is Low, the B is connected to X, and when S is High, A is connected to X. While it may take some nanoseconds for the transmission gate to change its state, the propagation delay from input to output of the gate would be very small.



CMOS LOGIC FAMILIES

The first commercially successful CMOS family was 4000-series CMOS. Although 4000-series circuits offered the benefit of low power dissipation, they were fairly slow and were not easy to interface with the most popular logic family of the time, bipolar TTL. Thus, the 4000 series was supplanted in most of applications by CMOS families that had better performance characteristics. The first two 74-series CMOS families are HC (High- speed CMOS) and HCT (High-speed CMOS, TTL compatible). HC and HCT both have higher speed and better current sinking and sourcing capability. The HCT family uses a power supply voltage VDD of 5 V and can be intermixed with TTL device, which also use a 5-V supply. The HC is mainly optimised for use in systems that use CMOS logic exclusively, and can use any power supply voltage between 2 and 6 V.

A higher voltage is used for higher speed, and lower voltage for lower power dissipation. Lowering the supply voltage is especially effective, since most CMOS power dissipation is proportional to the square of the voltage (CV2 f). Even when used with a 5 V power supply, HC devices are not quite compatible with TTL. In particular, HC circuits are designed to recognise CMOS input levels. The output levels produced by TTL devices do not quite match this range, so HCT devices use the different input levels. These levels are established in the fabrication process by making transistors with different switching threshold, producing the different transfer characteristics. Two more CMOS families, known as AC (Advanced CMOS) and ACT (Advanced CMOS, TTL compatible) were introduced in mid-1980s. These families are fast, comparable to ALSTTL, and they can source or sink more current than most of the TTL circuits can. Like HC and HCT, the AC and ACT families differ only in the input levels that they recognise; their output characteristics are the same. Also like HC/HCT, AC/ACT outputs have symmetric output drive.

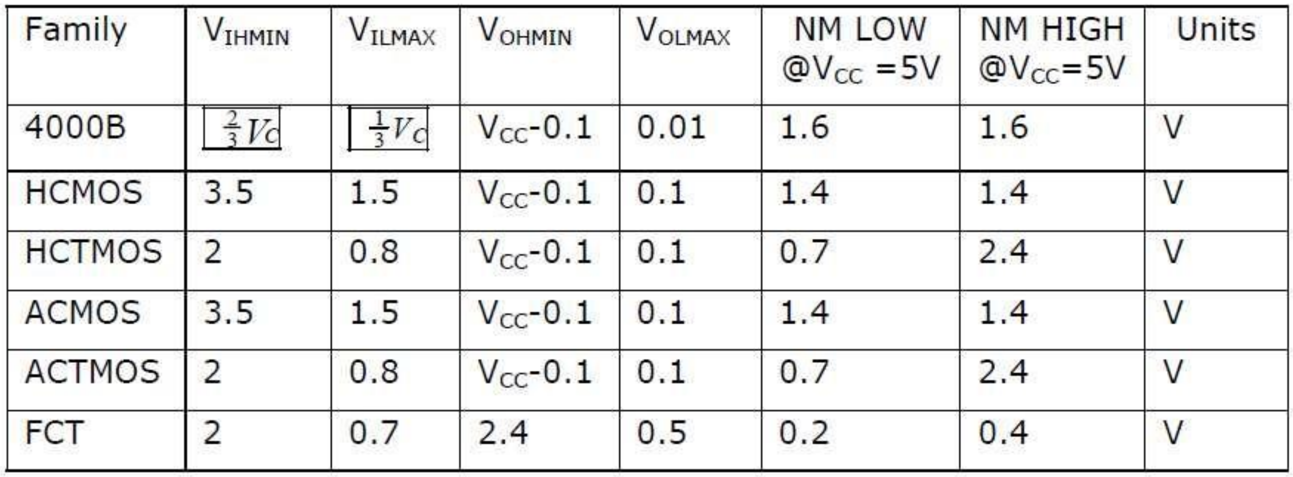
In the early 1990s, yet another CMOS family was launched. The FCT (Fast CMOS, TTL compatible) family combines circuit innovations with smaller transistor geometries to produce devices that are even faster than AC and ACT while reducing power consumption and maintaining full compatibility with TTL. There are two subfamilies, FCT-T and FCT2-T. These families represent a “technology crossover point” that occurred when the performance achieved using CMOS technology matched that of bipolar technology, and typically one third the power. Both the logic families are TTL compatible, which means that they conform to the industry-standard TTL voltage levels and threshold point (1.5 V), and operate from a 5 Volt VCC power source. All inputs are designed to have a hysterisis of 200 mV (low-to-high threshold of 1.6 V and high-to-low threshold of 1.4V). This hysteresis increases both the static and dynamic noise immunity, as well as reducing the sensitivity to noise superimposed on slowly rising or falling inputs. Individual logic gates are not manufactured in the FCT families. Just about the simplest FCT logic element is a 74FCT138/74FCT138T decoder, which has six inputs, eight outputs and contains the equivalent of about twelve 4-input gates internally

ELECTRICAL BEHAVIOUR OF CMOS CIRCUITS

This section presents the electrical characteristics of CMOS families. The electrical characteristics refer to DC noise margins, fan out, speed, power consumption, noise, electrical discharge, open drain outputs and three state outputs. Logical Levels and Noise Margins: The generated voltage levels given by the manufacturing data sheet for HCMOS circuits operating at VDD = 5 V, are given in the Table 1. The input parameters are mainly determined by the switching threshold of the two transistors, while the output parameters are determined by the

ON resistance of the transistors. These parameters apply when the device inputs and outputs are connected only to other CMOS devices. The dc voltage levels and noise margins of CMOS families are given in the Table 1.

TABLE 1: DC Characteristics of CMOS Families



Family	V_{IHMIN}	V_{ILMAX}	V_{OHMIN}	V_{OLMAX}	NM LOW @V_{CC} =5V	NM HIGH @V_{CC}=5V	Units
4000B	$\boxed{\frac{2}{3}V_C}$	$\boxed{\frac{1}{3}V_C}$	V_{CC}-0.1	0.01	1.6	1.6	V
HCMOS	3.5	1.5	V_{CC}-0.1	0.1	1.4	1.4	V
HCTMOS	2	0.8	V_{CC}-0.1	0.1	0.7	2.4	V
ACMOS	3.5	1.5	V_{CC}-0.1	0.1	1.4	1.4	V
ACTMOS	2	0.8	V_{CC}-0.1	0.1	0.7	2.4	V
FCT	2	0.7	2.4	0.5	0.2	0.4	V

These dc noise margins are significantly better than those associated with TTL families. As CMOS circuits can be operated with VDD = 2 V to VDD = 6 V the voltage levels associated with CMOS gates may be expressed as

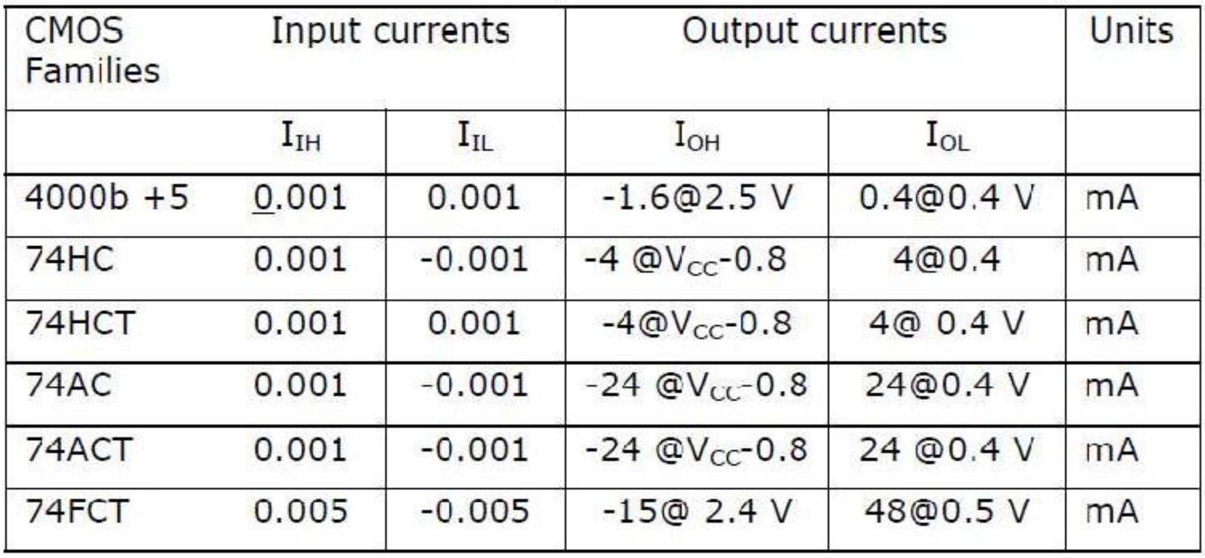
VIL(max) = 30% VDD

VOH(min) = VDD - 0.1 V

VIH(min) = 70% VDD

Regardless of the voltage applied to the input of a CMOS inverter, the input currents are very small. The maximum leakage current that can flow, designated as II max, is + 1µA for HCMOS with 5 V power supply. As the load on a CMOS gate could vary, the output voltage would also vary. Instead of specifying the output impedance under all conditions of loading the manufacturers specify a maximum load for the output in each state, and guarantee a worst-case output voltage for that load. The load is specified in terms of currents. The input and output currents are given in the Table 2.

TABLE 2: Input and Output Current Levels of CMOS Families



CMOS Families	Input currents		Output currents		Units
	I_{IH}	I_{IL}	I_{OH}	I_{OL}	
4000b +5	0.001	0.001	-1.6@2.5 V	0.4@0.4 V	mA
74HC	0.001	-0.001	-4 @V_{CC}-0.8	4@0.4	mA
74HCT	0.001	0.001	-4@V_{CC}-0.8	4@ 0.4 V	mA
74AC	0.001	-0.001	-24 @V_{CC}-0.8	24@0.4 V	mA
74ACT	0.001	-0.001	-24 @V_{CC}-0.8	24 @0.4 V	mA
74FCT	0.005	-0.005	-15@ 2.4 V	48@0.5 V	mA

These specifications are given at voltages which are normally associated with TTL gates. If the current drawn by the load is smaller, the voltage levels would improve significantly. This happens when CMOS gates are connected to CMOS loads. It is important to note that in a CMOS circuit the output structure by itself consumes very little current in either state, High or Low. In either state, one of the transistors is in the high impedance OFF state. When no load is connected the only current that flows through the transistors is their leakage current. With a load, however, current flows through both the load and the ON transistor, and power is consumed in both.

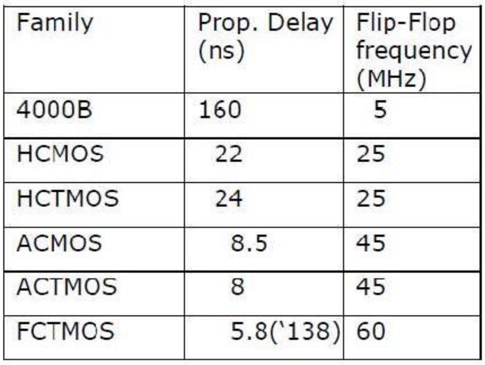
Fan out:

The fan out of a logic gate is the number of inputs that the gate can drive without exceeding its worst- case loading specifications. The fan out depends not only on the characteristics of the output, but also on the inputs that it is driving. When a HCMOS gate is driving HCMOS gates, we note that IILmax is +1 μA in any state, and IOHmax = -20 μA and IOLmax = 20 μA. Therefore, the Low-state fan out is 20 and High-state fan out is 20 for HCMOS gates. However, if we are willing to work with slightly degraded output voltages, which would reduce the available noise margins, we can go for IOHmax and IOLmax of 4.0 mA. This would mean that an HCMOS gate can drive as many as 4000 HCMOS gates. But in actuality this would not be true, as the currents we are considering are only the steady state currents and not the transition currents. The actual fan out under degraded load conditions would be far less than 4000. During the transitions, the CMOS output must charge or discharge the capacitance associated with the inputs that it derives. If this capacitance is too large, the transition from Low to High (or vice versa) may be too slow causing improper system operation.

CMOS DYNAMIC ELECTRICAL BEHAVIOUR

Both the speed and the power consumption of CMOS devices depend on to a large extent on AC or dynamic characteristics of the device and its load, that is, what happens when the output changes between states. The speed depends on two factors, transition times and propagation delay. The rise and fall times of an output of CMOS IC depend mainly on two factors, the ON transistor resistance and the load capacitance. The load capacitance comes from three different sources: output circuits including a gate's output transistors, internal wiring and packaging, have capacitances associated with them (of the order of 2-10 pF); wiring that connects an output to other inputs (about 1pF per inch or more depending on the wiring technology); and input circuits including transistors, internal wiring and packaging (2-15 pF per input). The OFF transistor resistance would be about 1 MΩ, the ON resistance of p-channel transistor would be of the order of 200 Ω, and the ON resistance of n-channel resistance would be about 100 Ω. We can compute the rise and fall times from the equivalent circuits. Several factors lead to nonzero propagation delays. In a CMOS device, the rate at which transistors change state is influenced both by the semiconductor physics of the device and by the circuit environment including input-signal transition rate, input capacitance, and output loading. The speed characteristics of CMOS families are given in the Table 3.

TABLE 3: Speed Characteristics of CMOS families



Family	Prop. Delay (ns)	Flip-Flop frequency (MHz)
4000B	160	5
HCMOS	22	25
HCTMOS	24	25
ACMOS	8.5	45
ACTMOS	8	45
FCTMOS	5.8('138)	60

Device outputs in AC and ACT families have very fast rise and fall times. Input signals should have rise and fall times of 3.0 ns (400 ns for HC and HCT devices) and signal swing of 0V to 3.0V for ACT devices or 0V to VDD for AC devices. Obviously such signal transition times are a major source of analog problems, including switching noise and "ground bounce".

Power Consumption:

A CMOS circuit consumes significant power only during transition, that is dynamic power dissipation is more. One source of dynamic power dissipation is the partial short-circuiting of the CMOS output structure. When the input voltage is changing from one state to the other, both the p-channel and nchannel output transistors may be partially ON, creating a series resistance of 600 Ω or less. During this transition period, current flows through the transistors from VDD to ground. The amount of power consumed in this way depends on the value of VDD, the frequency of output transitions, and an equivalent dissipation capacitance CPD as given by the manufacturer. PT = CPD . V2 DD. f PT is the internal power dissipation given in watts, VDD is the supply voltage in volts, f is frequency of output transitions in Hz, and CPD is the power dissipation capacitance in farads. CPD for a gate of HCMOS is about 24 pF. This relationship is valid only if the rise and fall times of the input signal are within the recommended maximum values.

ECL Family

The key to propagation delay in bipolar logic family is to prevent the transistors in a gate from saturating. Schottky families prevent the saturating using Schottky diodes across the base-collector junctions of transistors. It is also possible to prevent saturating by using a structure called Current Mode Logic (CML). Unlike other logic families considered so far, CML does not produce a large voltage swing between low and high levels. Instead, it has a small voltage swing, less than a volt, and it internal switches current between two possible paths depending on the output state.

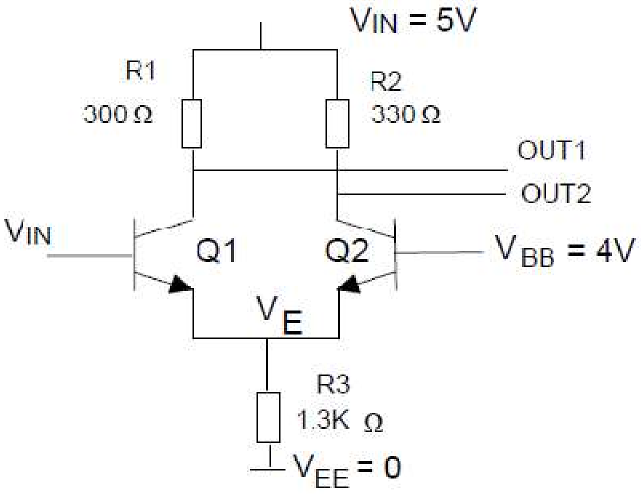
The first CML logic family was introduced by General Electric in 1961. The concept was refined by Motorola and others to produce today's 10K, 100K Emitter Coupled Logic (ECL) families. These ECL families are fast and offer propagation delays as short as 1 ns. In fact, through out the evolution of digital circuit technology, some type of CML has always been the fastest commercial logic family. However commercial ECL families are not nearly as popular as TTL and CMOS mainly because they consume too much power. In fact, high power consumption has made the design of ECL super computers, such as CRAY as much of a challenge in cooling technology as in digital design. In addition, ECL has poor power-speed product, does not provide a high level of integration, has fast edge rates requiring design for special transmission line effect, and is not directly compatible with TTL and CMOS. But ECL family continues to survive and in applications which require maximum speed regardless of cost.

ECL Circuits

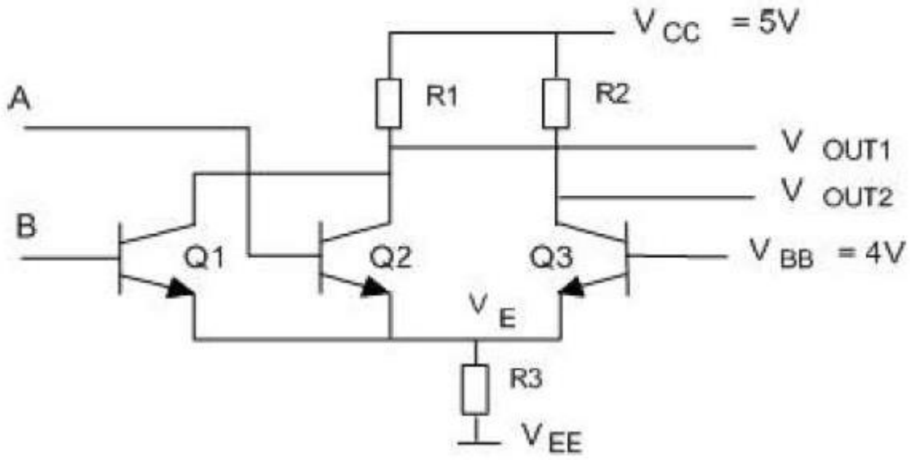
Basic CML Circuit:

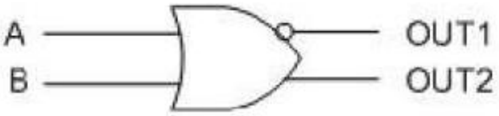
The basic idea of current mode logic is illustrated by the inverter/buffer circuit in the figure 1. This circuit has both inverting (OUT1) and non-inverting output (OUT2). Two transistors are connected as a differential amplifier with a common emitter resistor R3. Let the supply VCC = 5 V, VBB = 4 V and VEE = 0 V. Input Low and High levels are defined to be 3.6 and 4.4 V. This circuit produces output Low and High levels 0.6 V higher (4.2 and

5.0 V). When VIN is high transistor Q1 is ON, but not saturated, and transistor Q2 is OFF. When Q1 is ON VE is one diode drop lower than VIN, or 3.8 V. Therefore, current through R3 is (3.8/1.3 KΩ) 2.92 mA. If Q1 has a β of 10, then 2.65 mA of this current comes through the collector and R1, so VOUT1 is 4.2V (Low) since the voltage across Q1 (= 4.2 - 3.8= 0.4 V) is greater than VCEsat,, Q1 is not saturated Q2 is off because of its base to emitter voltage (4.0 - 3.8 = 0.2 V) is less than 0.6 V. Thus VOUT2 is at 5.0 V (High) as no current passes through R2.



When VIN is Low, transistor Q1 is OFF, and Q2 is ON but not saturated. VE will be one diode drop below VBB (4.0 - 0.6 = 3.4 V). The current trough R3 is (3.4/1.3 KΩ =) 2.6 mA. The collector current of Q2 is 2.38 mA for a β of 10. The voltage drop across R2 is (2.38 x 0.33 =) 0.5 V, and VOUT2 is about 4.2 V. Since the collector emitter voltage of Q2 is (4.2 - 3.4 =) 0.8V, it is not saturated. Q1 is off because its base-emitter voltage is (3.6 - 3.4 =) 0.2 and is less than 0.6 V. Thus VOUT1 is pulled up to 5.0 V through R1. To perform logic with the basic unit of figure 1, we simply place additional transistors in parallel with Q1. Figure 2 shows a 2-input OR/NOR gate. If any input is High, the corresponding input transistor is active, and VOUT1 is Low (NOR output). At the same time, Q3 is off, and VOUT2 is High (OR output). However, the circuit shown in figure 2 cannot meet the input/output loading requirements effectively.





ECL SUBFAMILIES

Motorola has offered MECL circuits in five logic families: MECL I, MECL II, MECL III, MECL 10000 (MECL 10K), and MECL 10H000 (MECL 10KH). The MECL I family was introduced in 1962, offering 8 ns gate propagation delay and 30 MHz toggle rates. This was the highest performance from any logic family at that time. However, this family required a separate bias driver package to be connected to each logic function. The ten pin packages used by this family limited the number of gates per package and the number of gate inputs. MECL II was introduced in 1966. This family offered 4 ns propagation delay for the basic gate, and 70 MHz toggle rates. MECL II circuits have a temperature compensated bias driver internal to the circuits, which simplifies circuit interconnections. MECL III was introduced in 1968. They offered 1 ns gate propagation delays and flip-flop toggle rates higher than 500 MHz. The 1 ns rise and fall times required a transmission line environment for all but the smallest systems. For this reason, all circuit outputs are designed to drive transmission lines and all output logic levels are specified when driving 50-ohm loads. For the first time with MECL, internal input pull down resistors are included with the circuits to eliminate the need to tie unused inputs to VEE..

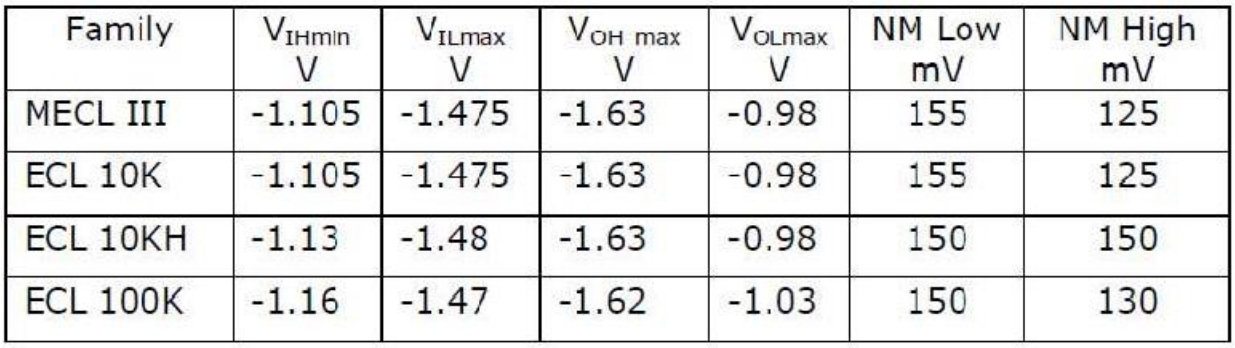
Motorola introduced MECL 10K series in 1971 with 2 ns propagation delays. In order to make the circuits comparatively easy to use, edge speed was slowed down to 3.5 ns. Subsequently, the basic MECL 10K series has been expanded by a subset of devices with even greater speed. These subfamilies are 10100 and 10500 series (propagation delay of 2 ns, edge speed of 3.5 ns and flip-flop toggle rate of 160 MHz), 10200 and 10600 series (propagation delay of 1.5 ns, edge 5 speed of 2.5 ns and flip-flop toggle rate of 250 MHz), and 10800 LSI family (propagation delay of 1 - 2.5 ns and edge speed of 3.5 ns) MECL 10KH family was introduced in 1981. This family provides a propagation delay of 1 ns with edge speed at 1.8 ns. These speeds, which were attained with no increase in power over MECL 10K, are due to both advanced circuit design techniques and new oxide isolated process called MOSAIC.

To enhance the existing systems, many of the MECL 10KH devices are pin-out/functional duplications of the MECL 10K family. Also, MECL 10K/10KH are provided with logic levels that are completely compatible with MECL III. Another important feature of MECL 10K/10KH is the significant power reduction over both MECL III and the older MECL II. Because of the power reductions and advanced circuit design techniques, the MECL 10KH family has many new functions not available with the other families. The latest entrant to the ECL family is ECL 100K, having 6-digit part numbers. This family offers functions, in general, different from those offered by 10K series. This family operates with a reduced power supply voltage -4.5 V, has shorter propagation delay of 0.75 ns, and transition time of 0.7 ns. However, the power consumption per gate is about 40 mW.

ELECTRICAL CHARACTERISTICS OF ECL FAMILY

The input and output levels, and noise margins of ECL gates are given in the Table 1. These values are specified at TA = 25o C and the nominal power supply voltage of VEE = -5.2 V

TABLE 1: Voltage levels and noise margins of ECL family ICs

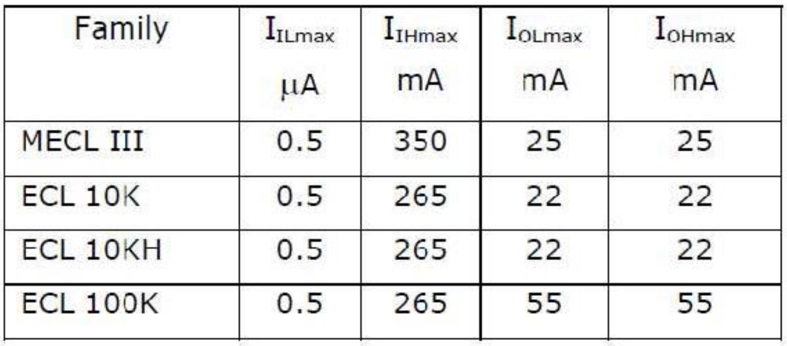


Family	V_{IHmin} V	V_{ILmax} V	$V_{OH\ max}$ V	V_{OLmax} V	NM Low mV	NM High mV
MECL III	-1.105	-1.475	-1.63	-0.98	155	125
ECL 10K	-1.105	-1.475	-1.63	-0.98	155	125
ECL 10KH	-1.13	-1.48	-1.63	-0.98	150	150
ECL 100K	-1.16	-1.47	-1.62	-1.03	150	130

The noise margin levels are slightly different in High and Low states. This specification by itself does not give complete picture regarding the noise immunity of a system built with a particular set of circuits. In general, noise immunity involves line impedances, circuit output impedances, and propagation delay in addition to noise- margin specifications.

Loading Characteristics:

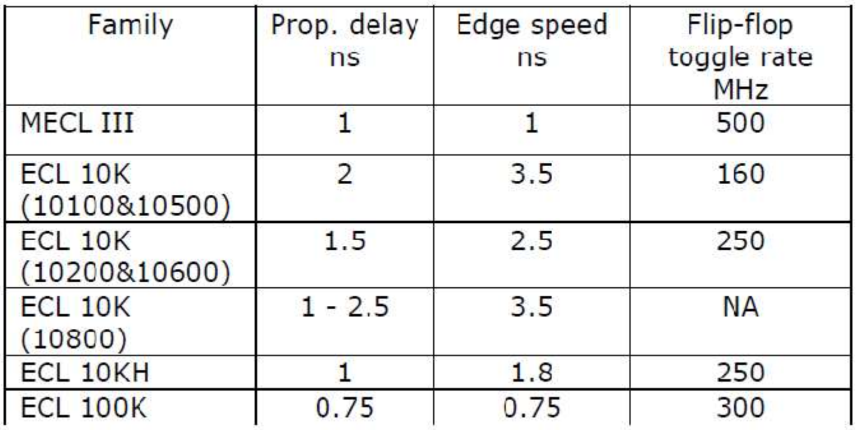
The differential input to ECL circuits offers several advantages. Its common-mode-rejection feature offers immunity against powersupply noise injection, and its relatively high input impedance makes it possible for any circuit to drive a relatively large number of inputs without deterioration of the guaranteed noise margin. Hence, DC fan out with ECL circuits does not normally present a design problem. Graphs given by the vendor showing the output voltage levels as a function load current can be used to determine the actual output voltages for loads exceeding normal operation.



Family	I_{ILmax} µA	I_{IHmax} mA	I_{OLmax} mA	I_{OHmax} mA
MECL III	0.5	350	25	25
ECL 10K	0.5	265	22	22
ECL 10KH	0.5	265	22	22
ECL 100K	0.5	265	55	55

Transition Times and Propagation Delays:

The transition times and delays associated with different ECL families are given in the following.

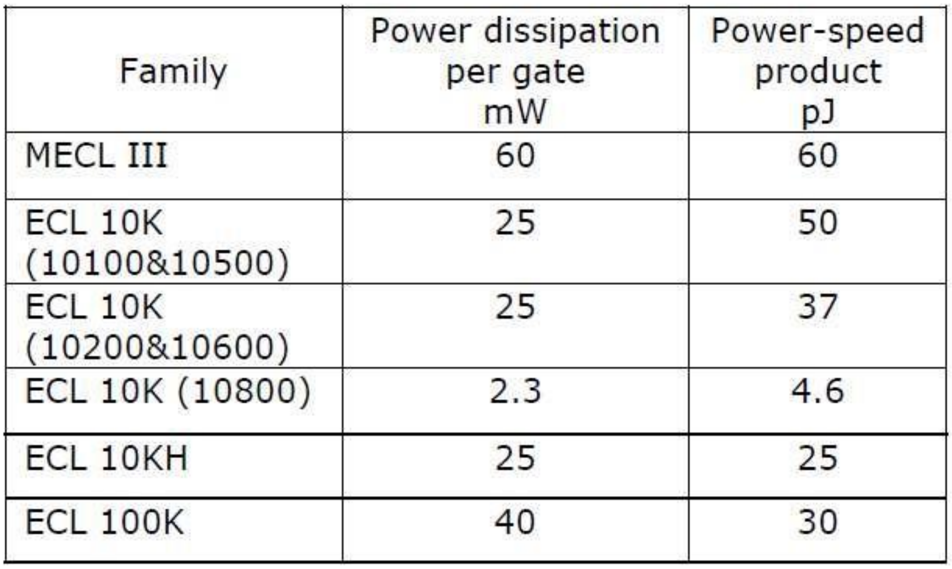


Family	Prop. delay ns	Edge speed ns	Flip-flop toggle rate MHz
MECL III	1	1	500
ECL 10K (10100&10500)	2	3.5	160
ECL 10K (10200&10600)	1.5	2.5	250
ECL 10K (10800)	1 - 2.5	3.5	NA
ECL 10KH	1	1.8	250
ECL 100K	0.75	0.75	300

The rise and fall times of an ECL output depend mainly on two factors, the termination resistor and the load capacitance. Most of the ECL circuits typically have a 7 ohm output impedance and are relatively unaffected by capacitive loading on positive going output signal. However, the negative-going edge is dependent on the output pull down or termination resistor. Loading close to a ECL output pin will cause an additional propagation delay of 0.1 ns per fan-out load 7 with 50 ohm resistor to -2.0 Vdc or 270 ohms to -5.2 Vdc. The input loading capacitance of an ECL 10K gate is about 2.9 pF. To allow for the IC connector or solder connection and a short stub length 5 to 7 pF is commonly used in loading calculations.

Power Consumption:

The power dissipation of ECL functional blocks as specified by the manufacturer does not include power dissipated in the output devices due to output termination. The omission of internal output pull down resistors permits the use of external terminations designed to yield best system performance. To obtain total operating power dissipation of a particular functional block in a system, the dissipation of the output transistor, under load, must be added to the circuit power dissipation. The power dissipation and power-speed products of various ECL families are given in the Table 4



Family	Power dissipation per gate mW	Power-speed product pJ
MECL III	60	60
ECL 10K (10100&10500)	25	50
ECL 10K (10200&10600)	25	37
ECL 10K (10800)	2.3	4.6
ECL 10KH	25	25
ECL 100K	40	30

EC 8392 – DIGITAL ELECTRONICS

UNIT – V : MEMORY DEVICES AND DIGITAL INTEGRATED CIRCUITS

ERROR DETECTION AND CORRECTION

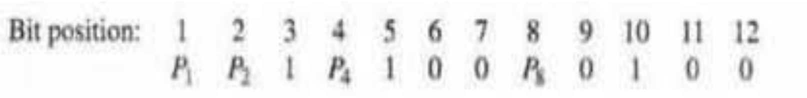
The dynamic physical interaction of the electrical signals affecting the data pam of a memory unit may cause occasional errors in storing and retrieving the binary information. The reliability of a memory unit may be improved by employing error-detecting and error- correcting codes. The most common error detection scheme is the parity bit. A parity bit is generated and stored along with the data word in memory. The parity of the word is checked after reading it from memory. The data word is accepted if the parity of the bits read out is correct. If the parity checked results in an inversion, an error is detected, but it cannot be corrected.

An error-correcting code generates multiple parity check bits that are stored with the data word in memory. Each check bit is parity over a group of bits in the data word. When the word is read back from memory; the associated parity bits are also read from memory and compared with a new set of check bits generated from the data that have been read. If the check bits are correct, no error has occurred. If the check bits do not match the stored parity they generate a unique pattern called a syndrome that can be used to identify the bit that is in error. A single error occurs when a bit changes in value from 1 to 0 or from 0 to 1 during the write or read operation. If the specific bit in error is identified, then the error can be corrected by complementing the erroneous bit.

Hamming Code

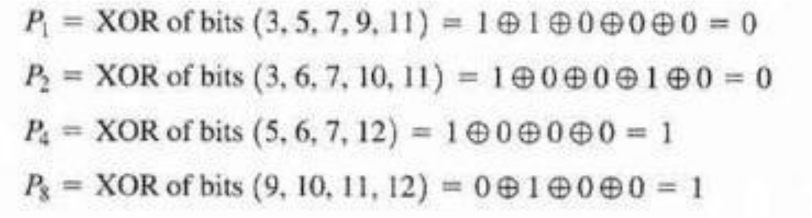
One of the most common error-correcting codes used in RAMs was devised by R. W. Hamming. In the Hamming code, k parity bits are added to an n-bit data word, forming a new word of n + k bits. The hit positions are numbered in sequence from 1 to n + k. These positions numbered as a power of 2 are reserved for the parity bits. The remaining bits are the data bits. The code can be used with words of any length.

Consider, for example, the 8-bit data word 11000100. We include 4 parity bits with the 8-bit word and arrange the 12 bits as follows:



Bit position:	1	2	3	4	5	6	7	8	9	10	11	12
	P_1	P_2	1	P_4	1	0	0	P_8	0	1	0	0

The 4 parity bits, P1, P2,P3 and P4 are in positions 1, 2, 4 and 8 respectively. The 8 bits of the data word are in the remaining positions. Each parity bit is calculated as follows:



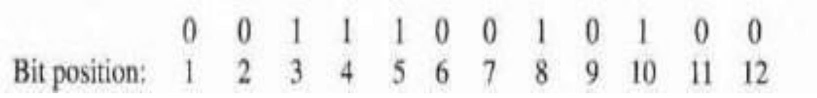
$$P_1 = \text{XOR of bits } (3, 5, 7, 9, 11) = 1 \oplus 1 \oplus 0 \oplus 0 \oplus 0 = 0$$

$$P_2 = \text{XOR of bits } (3, 6, 7, 10, 11) = 1 \oplus 0 \oplus 0 \oplus 1 \oplus 0 = 0$$

$$P_4 = \text{XOR of bits } (5, 6, 7, 12) = 1 \oplus 0 \oplus 0 \oplus 0 = 1$$

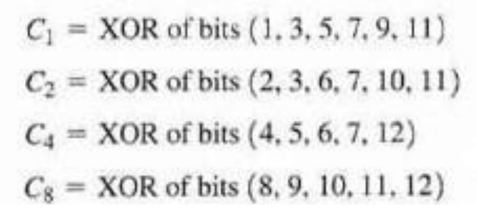
$$P_8 = \text{XOR of bits } (9, 10, 11, 12) = 0 \oplus 1 \oplus 0 \oplus 0 = 1$$

The 8-bit data word is stored in memory together with the 4 parity bits as a 12-bit compos ite word. Substituting the 4 P bits in their proper positions, we obtain the 12-bit composite word stored in memory:



	0	0	1	1	1	0	0	1	0	1	0	0
Bit position:	1	2	3	4	5	6	7	8	9	10	11	12

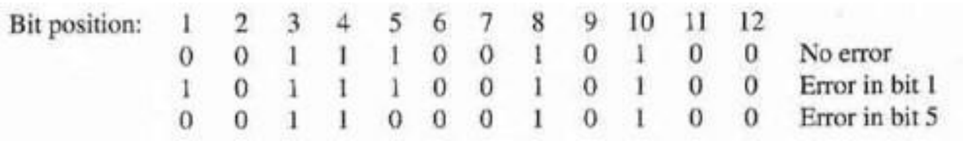
When the 12 bits arc read from memory, they are checked again for errors. The parity is checked over the same combination of bits, including the parity bit. The 4 check bits are evaluated as follows:



$$C_1 = \text{XOR of bits } (1, 3, 5, 7, 9, 11)$$
$$C_2 = \text{XOR of bits } (2, 3, 6, 7, 10, 11)$$
$$C_4 = \text{XOR of bits } (4, 5, 6, 7, 12)$$
$$C_8 = \text{XOR of bits } (8, 9, 10, 11, 12)$$

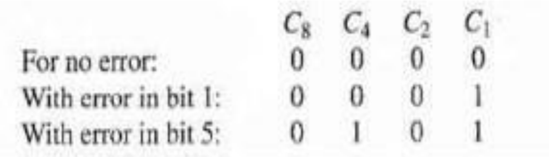
A 0 check bit de signates e ven parity over the checked bits and a 1 designates odd parity. Since the bits were stored with even parity, the result, C= $C_8C_4C_2C_1$ = 0000, indicates that no error has occurred. However, if C

≠0, then the 4 -bit binary number formed by the check bits gives the position of the error bit. For example, consider the following three cases:



Bit position:	1	2	3	4	5	6	7	8	9	10	11	12	
	0	0	1	1	1	0	0	1	0	1	0	0	No error
	1	0	1	1	1	0	0	1	0	1	0	0	Error in bit 1
	0	0	1	1	0	0	0	1	0	1	0	0	Error in bit 5

In the first case, there is no error in the l2·bit word. In the second case, there is an error in bit position number I because it changed from 0 to I. The third case shows an error in bit position 5, with a change from 1 to 0. Evaluating the XQR of the corresponding bits, we determine the 4 check bits to be as follows:



	C_8	C_4	C_2	C_1
For no error:	0	0	0	0
With error in bit 1:	0	0	0	1
With error in bit 5:	0	1	0	1

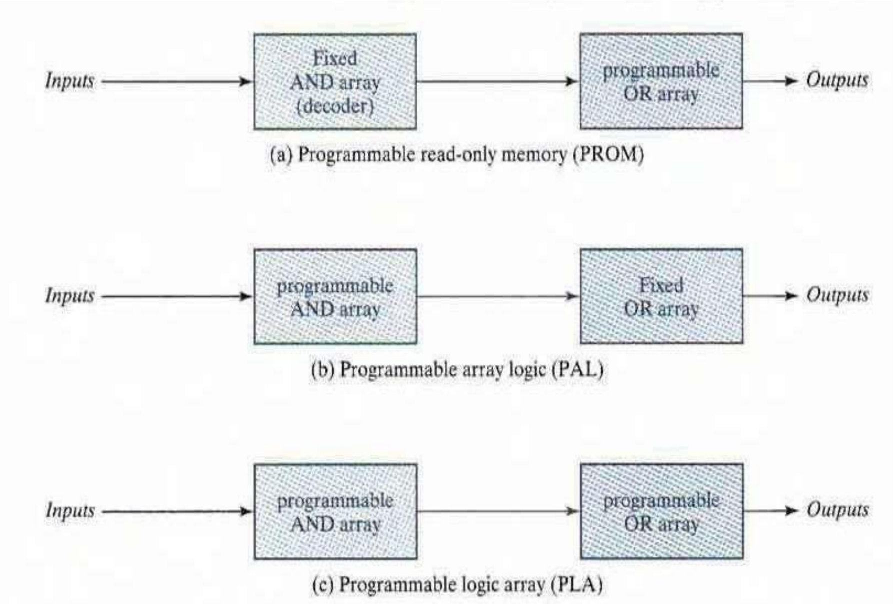
Thus, for no error, we have C = 0000; with an error in bit 1, we obtain C = 0001; and with an error in bit 5, we get C = 0101. When the binary number C is not equal to 0000, it gives the position of the bit in error. The error can be corrected by complementing the corresponding bit. Note that an error can occur in the data word or in one of the parity bits.

EC 8392 – DIGITAL ELECTRONICS

UNIT – V : MEMORY DEVICES AND DIGITAL INTEGRATED CIRCUITS

COMBINATIONAL PLDs

The PROM is a combinational programmable logic device (PLD) - an integrated circuit with programmable gates divided into an AND array and an OR array to provide an AND-OR sum of- product implementation. There are three major types of combinational PLDs differing in the placement of the programmable connections in the AND-OR array. The PROM has a fixed AND array constructed as a decoder and a programmable OR array. The programmable OR gates implement the Boolean functions in sum-of-minterms form. The PAL has a programmable ASD array and a fixed OR array. The AND gates are programmed to provide the product terms for the Boolean functions, which are logically summed in each OR gate. The most flexible PLD is the PLA, in which both the AND and OR arrays can be programmed. The product terms in the AND array may be shared by any OR gate to provide the required sum-of-products implementation.



(a) Programmable read-only memory (PROM)

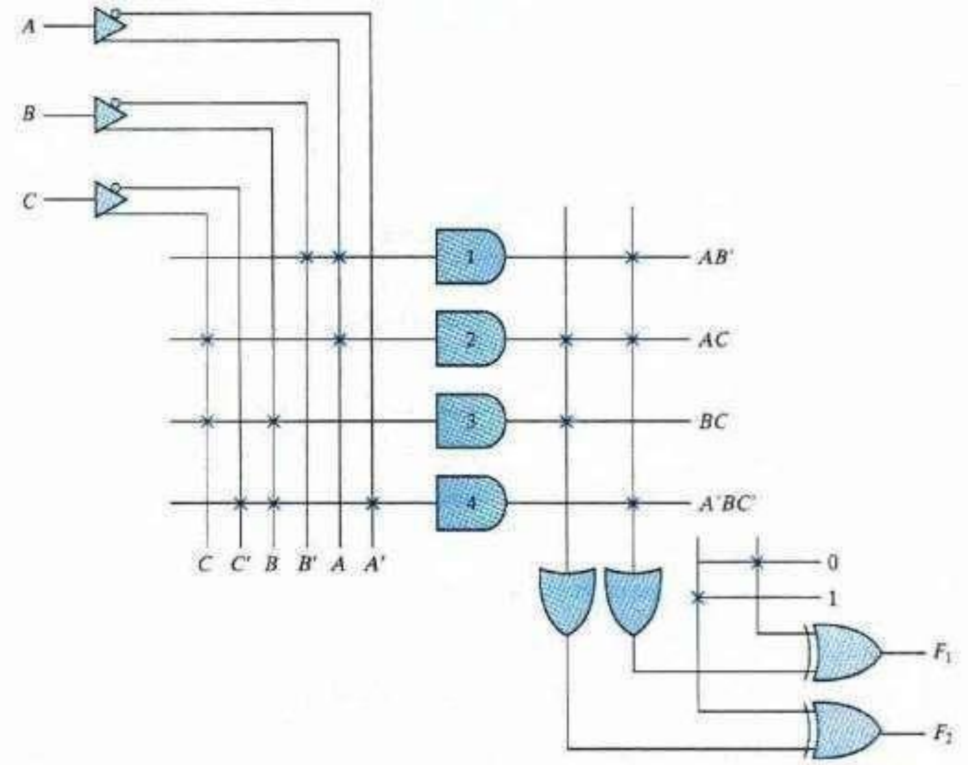
(b) Programmable array logic (PAL)

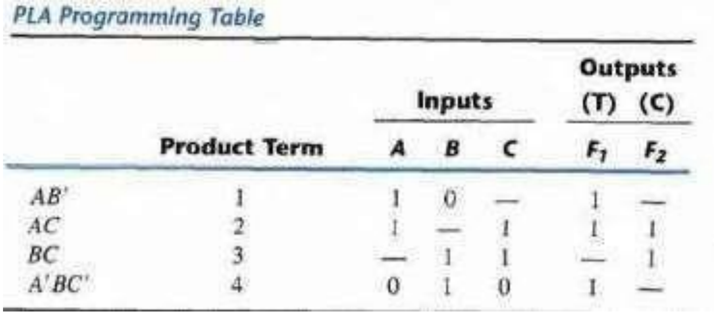
(c) Programmable logic array (PLA)

PROGRAMMABLE LOGIC ARRAY

The PLA is similar in concept to the PROM, except that the PLA does not provide full decoding of the variables and does not generate all the minterms. The decoder is replaced by an array of AND gates that can be programmed to generate any product term of the input variables. The product terms are then connected to OR gates to provide the sum of products for the required Boolean functions. The diagram uses the array logic graphic symbols for complex circuits. Each input goes through a buffer-inverter combination, shown in the diagram with a composite graphic symbol, that has both the true and complement outputs. Each input and its complement are connected to the inputs of each ANDgate, as indicated by the intersections between the vertical and horizontal lines. The outputs of the AND gates are connected to the inputs of each OR gate. The output of the OR gate goes to an XOR gate, where the other input can be programmed to receive a signal equal to either logic 1 or logic 0. The output is inverted when the XOR input is connected to 1. The output does not change when the XOR input is connected to 0. The particular Boolean functions implemented in the
PLA of below figure are,

$$F_1 = AB' + AC + A'BC'$$
$$F_2 = (AC + BC)'$$





PLA Programming Table

		Inputs			Outputs	
					(T)	(C)
	Product Term	A	B	C	F_1	F_2
AB'	1	1	0	—	1	—
AC	2	1	—	1	1	1
BC	3	—	1	1	—	1
A'BC'	4	0	1	0	1	—

PROGRAMMABLE ARRAY LOGIC

The PAL is a programmable logic device with a fixed OR array and a programmable AND array. Because only the AND gates are programmable, the PAL is easier to program than, but is not as flexible as the PLA. The below figure shows, the logic configuration of a typical PAL with four inputs and four outputs. Each input has a buffer- inverter gate, and each output is generated by a fixed OR gate. There are four sections in the unit. each composed of an AND-OR array that is three wide, the term used to indicate that there are three programmable AND gates in each section and one fixed OR gate. Each AND gate has 10 programmable input connections, shown in the diagram by 10 vertical lines intersecting each horizontal line. The horizontal line symbolizes the multiple-input configuration of the AND gate. One of the outputs is connected to a buffer-inverter gate and then fed back into two inputs of the AND gates.

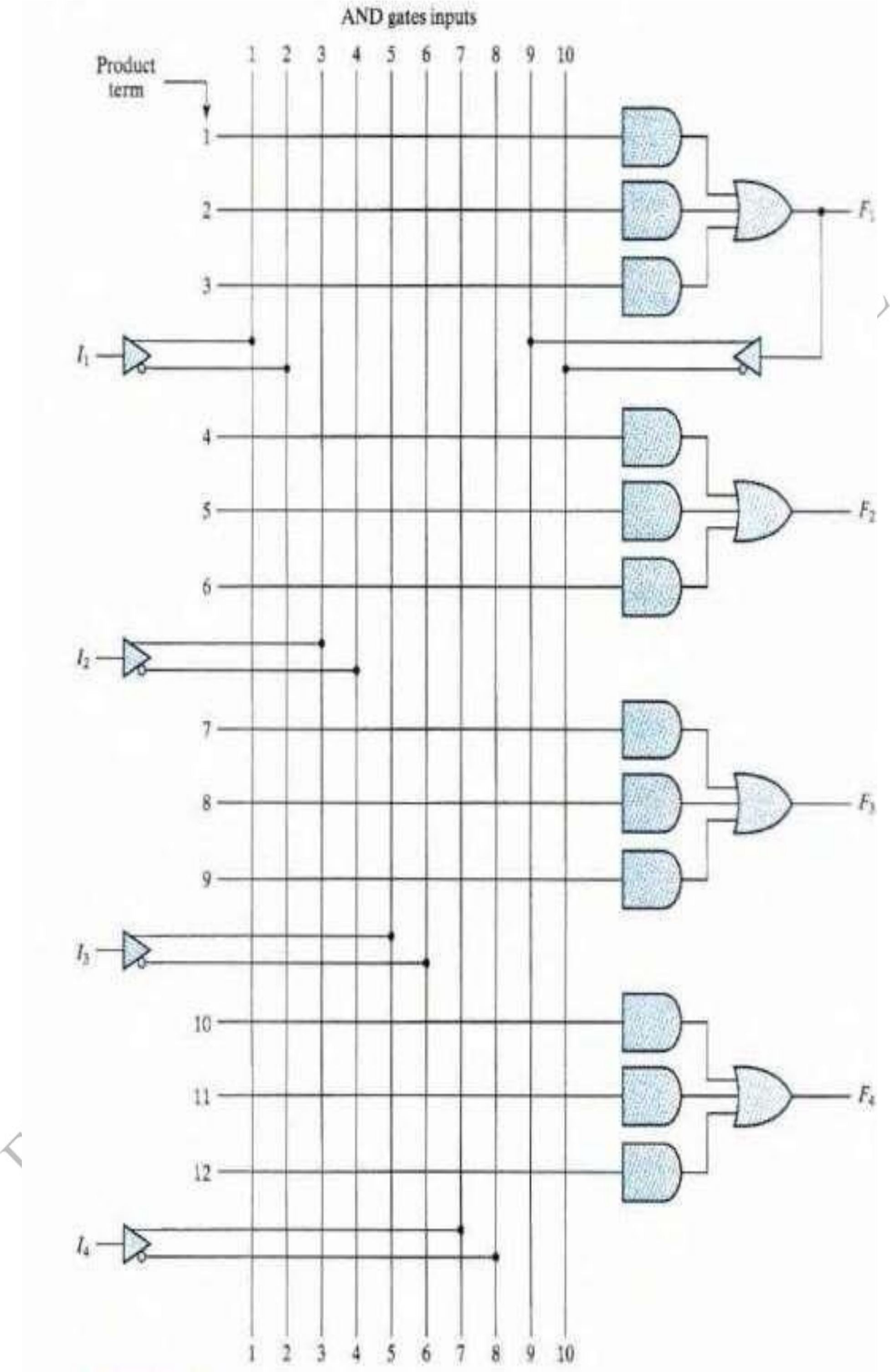
In designing with a PAL, the Boolean functions must be simplified to fit into each section. Unlike the situation with a PLA, a product term cannot be shared among two or more OR gates. Therefore, each function can be simplified by itself, without regard to common product terms. The number of product terms in each section is fixed and if the number of terms in the function is too large, it may be necessary to use two sections to implement one Boolean function.

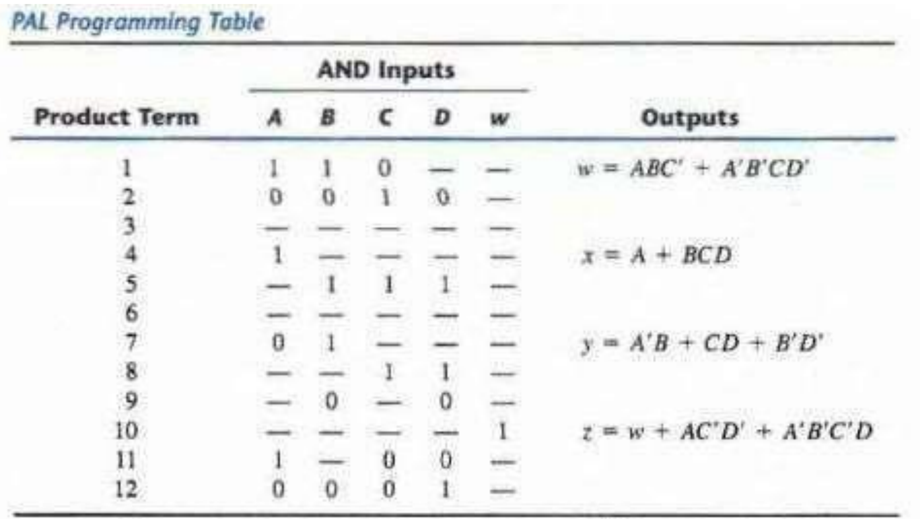
As an example of using a PAL in the design of a combinational circuit, consider the following Boolean functions, given in sum-of-minterms form:

$$
\begin{aligned}
w(A, B, C, D) &= \Sigma(2, 12, 13)\\
x(A, B, C, D) &= \Sigma(7, 8, 9, 10, 11, 12, 13, 14, 15)\\
y(A, B, C, D) &= \Sigma(0, 2, 3, 4, 5, 6, 7, 8, 10, 11, 15)\\
z(A, B, C, D) &= \Sigma(1, 2, 8, 12, 13)
\end{aligned}
$$

Simplifying the four functions to a minimum number of terms results in the following Boolean functions:

$$
\begin{aligned}
w &= ABC' + A'B'CD'\\
x &= A + BCD\\
y &= A'B + CD + B'D'\\
z &= ABC' + A'B'CD' + AC'D' + A'B'C'D\\
&= w + AC'D' + A'B'C'D
\end{aligned}
$$


AND gates inputs
1 2 3 4 5 6 7 8 9 10
Product term
1
2
3
I_1
F_1
4
5
6
F_2
I_2
7
8
9
F_3
I_3
10
11
12
F_4
I_4
1 2 3 4 5 6 7 8 9 10



PAL Programming Table

	AND Inputs					
Product Term	***A***	***B***	***C***	***D***	***w***	**Outputs**
1	1	1	0	—	—	$w = ABC' + A'B'CD'$
2	0	0	1	0	—	
3	—	—	—	—	—	
4	1	—	—	—	—	$x = A + BCD$
5	—	1	1	1	—	
6	—	—	—	—	—	
7	0	1	—	—	—	$y = A'B + CD + B'D'$
8	—	—	1	1	—	
9	—	0	—	0	—	
10	—	—	—	—	1	$z = w + AC'D' + A'B'C'D$
11	1	—	0	0	—	
12	0	0	0	1	—	

EC 8392 – DIGITAL ELECTRONICS

UNIT – V : MEMORY DEVICES AND DIGITAL INTEGRATED CIRCUITS

INTRODUCTION

A memory unit is a device to which binary information is transferred for storage and from which information is retrieved when needed for processing. When data processing takes place, information from memory is transferred to selected registers in the processing unit. Intermediate and final results obtained in the processing unit are transferred back to be stored in memory. Binary information received from an input device is stored in memory and information transferred to an output device is taken from memory. A memory unit is a collection of cells capable of storing a large quantity of binary information.

There are two types of memories that are used in digital systems: random-access memory (RAM) and read-anly memory (ROM). RAM stores new information for later use. The Process of storing new information into memory is referred to as a memory write operation. The process of transferring the stored infomation out of memory is referred to as a memory read operation. RAM can perform both write and read operations. ROM can perfonn only the read operation. This means that suitable binary information is already stored inside memory and can be retrieved or read at any time. However, that information cannot be altered by writing.

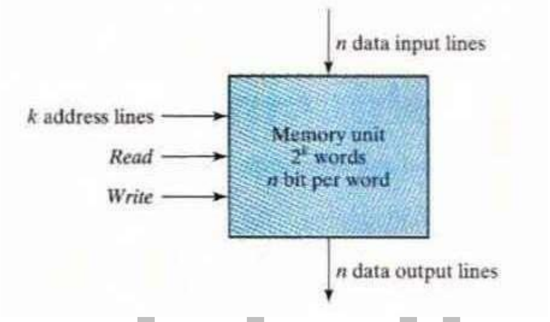
RAM

A memory unit is a collection of storage cells, together with associated circuits needed to transfer information into and out of a device. The architecture of memory is such that information can be selectively retrieved from any of its internal allocations. The time it takes to transfer information to or from any desired random location is always the same- hence the name random access memory, abbreviated RAM. In contrast, the time required to retrieve information that is stored on magnetic tape depends on the location of the data.

A memory unit stores binary information in groups of bits called words. A word in a memory is an entity of bits that move in and out of storage as a unit. A memory word is a group of 1 's and 0's and may represent a number, an instruction , one or more alphanumeric characters or any other binary-coded information. A group of 8 bits is called a byte. Most computer memories use words that are multiples of 8 bits in length. Thus, a

16·bit word contains two bytes, and a 32-bit word is made up of four bytes. The capacity of a memory unit is usually stated as the total number of bytes that the unit can store.

The n data input lines provide the information to be stored in memory, and the n data output lines supply the information coming out of memory. The k address lines specify the particular word chosen among the many available. The two control inputs specify the direction of transfer desired: The Write input causes binary data to be transferred into the memory and the Read input causes binary data to be transferred out of memory.



The memory unit is specified by the number of words it contains and the number of bits in each word. The address lines select one particular word. Each word in memory is assigned an identification number, called an address, starting fro 0 to $2^k - 1$, where k is the number of address lines. The selection of a specific word inside memory is done by applying the k· bit address to the address lines.

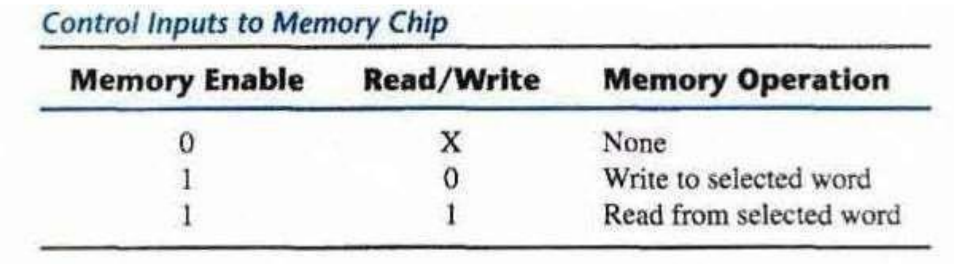
Write and Read Operations

The two operations that RAM can perform are the write and read operations. The write signal specifies a transfer-in operation and the read signal specifies a transfer out operation. On accepting one of these control signals, the internal circuits inside the memory provide the desired operation. The steps that must be taken for the purpose of transferring a new word to be stored into memory are as follows:

1. Apply the binary address of the desired word to the address lines.
2. Apply the data bits that must be stored in memory to the data input lines.
3. Activate the write input.

The memory unit will then take the bits from the input data lines and store them in the word specified by the address lines. The steps that must be taken for the purpose of transferring a stored word out of memory are as follows:

1. Apply the binary address of the desired word to the address lines.
2. Activate the read input.



Control Inputs to Memory Chip

Memory Enable	Read/Write	Memory Operation
0	X	None
1	0	Write to selected word
1	1	Read from selected word

Types of Memories

The mode of access of a memory system is determined by the type of components used. In a random-access memory, the word locations may be thought of as being separated in space, each word occupying one particular location. In a sequential-access memory, the information stored in some medium is not immediately accessible, but is available only at certain intervals of time. A magnetic disk or tape unit is of this type. Each memory location passes me read and write heads in turn, but information is read out only when the requested word has been reached. In a random-access memory, the access time is always the same regardless of the particular location of the word. In a sequential - access memory, the time it takes to access a word depends on the position of the word with respect to the position of the read head: therefore, the access time is variable.

Integrated circuits RAM units are available in two cpereting modes: static and dynamic. Static RAM (SRAM) consists essentially of internal latches that store the binary information. The stored information remains valid as long as power is applied to the unit. Dynamic RAM (DRAM) stores the binary information in the form of electric charges on capacitors provided inside the chip by MOS transistors. The stored charge on the capacitors tends to discharge with time, and the capacitors must be periodically recharged by refreshing the dynamic memory. Refreshing is done by cycling through the words few milliseconds to restore the decaying charge. DRAM offers reduced power consumption and larger storage capacity in a singte memory chip. SRAM is easier to use and has shorter read and write cycles.

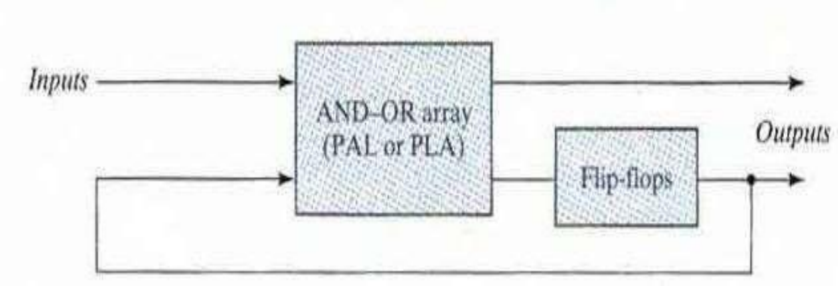
EC 8392 – DIGITAL ELECTRONICS

UNIT – V : MEMORY DEVICES AND DIGITAL INTEGRATED CIRCUITS

SEQUENTIAL PROGRAMMABLE DEVICES

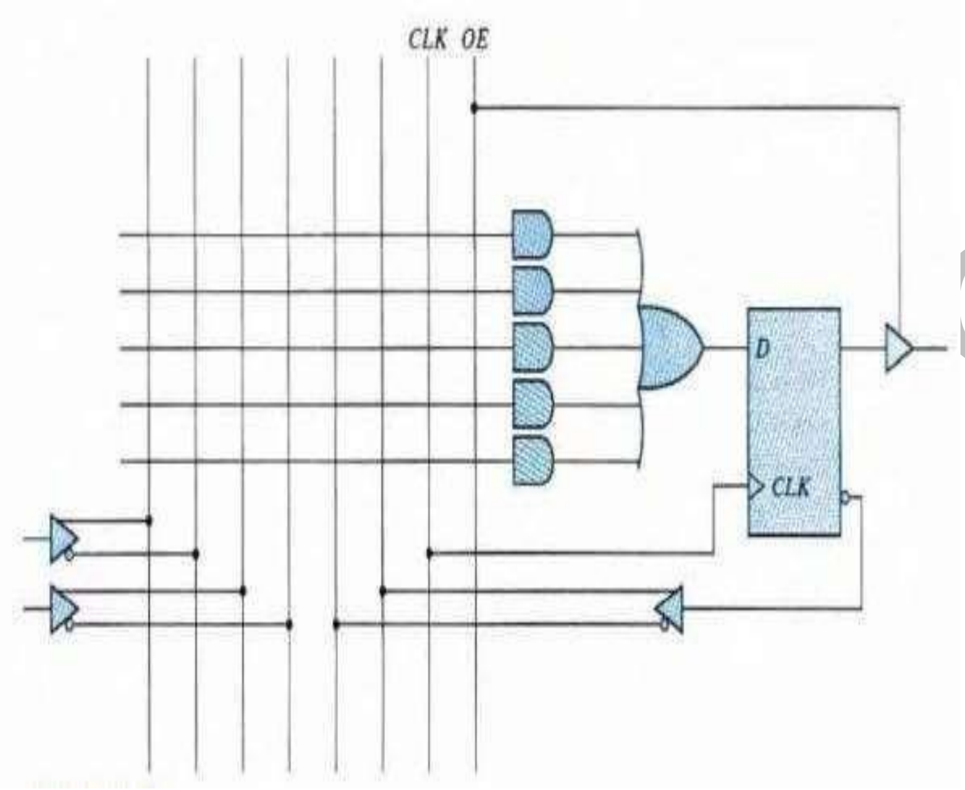
Digital systems are designed with flip-flops and gates. Since the combinational PLD consists of only gates, it is necessary to include external flip-flops when they are used in the design. Sequential programmable devices include both gates and flip-flops. In this way, the dev ice can be programmed to perform a variety of sequential-circuit functions.

1. Sequential (or simple) programmable logic device (SPLD)
2. Complex programmable logic device (CPLD)
3. Field-programmable gate array (FPGA)

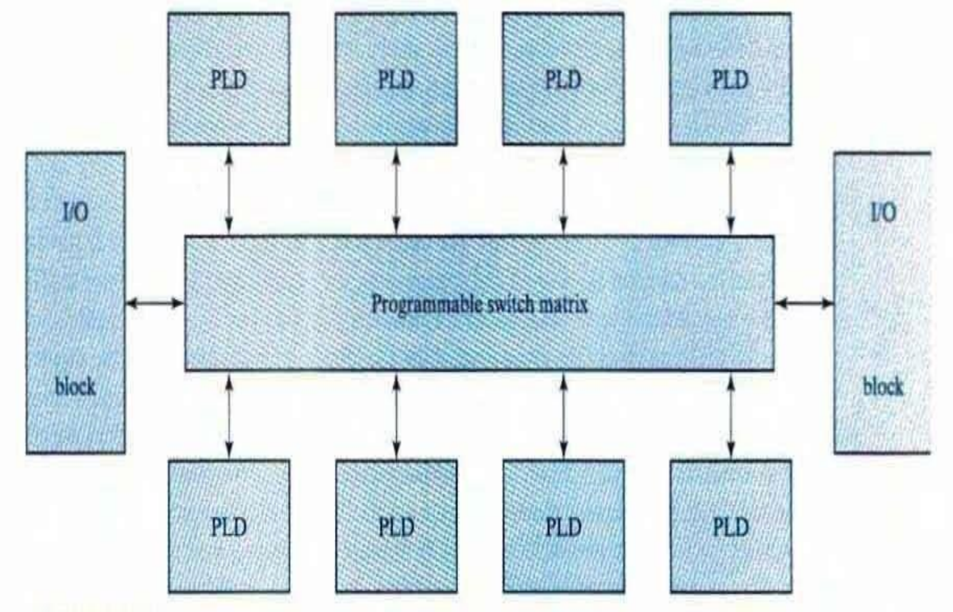


The sequential PLD is sometimes referred 10as a simple PLD to differentiate it from the complex PLD. The SPLD includes flip-flops, in addition to the AND-OR array, within the integrated circuit chip. The result is a sequential circuit as shown in Fig. 7.18. A PAL or PLA is modified by including a number of flip-flops connected to for a register, The circuit outputs can be taken from the OR gates or from the outputs of the flip-flops. Additional programmable connections are available to include the flip-flop outputs in the product terms formed with the AND array. The flip-flops may be of the D or the JK type. The first programmable device developed to support sequential circuit implementation is the fieldprogrammable logic sequencer (FPLS). A typical FPLS is organized around a PLA with several outputs driving flip- flops. The flip-flops are flexible in that they can be programmed to operate as either the JK o r the D type. The FPLS did not succeed commercially, because it has too many programmable connections. The configuration mostly used in an SPLD is the combinational PAL together with D flip-flops. A PAL that includes flip -flops is referred to as a registered PAL, to signify that the device contains flip-flops in addition to the AND-DR array.

Each section of an SPLD is called a macrocell, which is a circuit that contains a sum- of-products combinational logic function and an optional flip-flop. The output is driven by an edge-triggered D flip-flop connected to a common clock input and changes state on a clock edge. The output of the flipflop is connected to a three- state buffer (or inverter) controlled by an output-enable signal marked in the diagram as OE. The output of the flip -flop is fed back into o ne of the inputs of the programmable AND gates to provide the present-state condition for the sequential circuit. A typical SPLD has from 8 to 10 macrocells within one IC package. All the flip -flop s are connected to the common CLK input, and all three-Male buffers are controlled by the OE input.



The design of a digital system using PLDs often requires the connection of several devices to produce the complete specification. For this type of application, it is more economical to use a complex programmable logic device (CPLD), which is a collection of individual PLDs on a single integrated circuit. A programmable interconnection structure allows the PLDs to be connected to each other in the same way that can be done with individual PLDs.



The device consists of multiple PLDs interconnected through a programmable switch matrix. The input- output blocks provide the connections to the IC pins. Each I/O pin is driven by a three-state buffer and can be programmed to act as input or output, the switch matrix receives inputs from the I/O block and directs them to the individual microcells. Similarly, selected outputs from macrocells are sent to the outputs as needed. Each PLD typically contains from 8 to 16 macrocells, usually fully connected. If a microcell has unused product terms, they can be used by other nearby macrocells. In some cases the macrocell flip-flop is programmed to act as a D, JK or T flip-flop.

A field-programmable gate array (FPGA) is a VLSI circuit that can be programmed at the user's location. A typical FPGA consists of an array of hundreds or thousands of logic blocks, surrounded by programmable input and output blocks and connected together via programmable interconnections. There is a wide variety of internal configurations within this group of devices. The performance of each type of device depends on the circuit contained in its logic blocks and the efficiency of its programmed interconnections.

APPLICATION SPECIFIC INTEGRATED CIRCUITS

Application Specific Integrated Circuit (ASIC) is an integrated circuit(IC) customized for a particular use, rather than intended for general-purpose use. ASICs are used in a wide- range of applications, including auto emission control, environmental monitoring, and personal digital assistants (PDAs). Field-programmable gate arrays (FPGA) are the modern- day technology for building a breadboard or prototype from standard parts; programmable logic blocks and programmable interconnects allow the same FPGA to be used in many different applications. For smaller designs or lower production volumes, FPGAs may be more cost effective than an ASIC design even in production.

Gate - Array Design

Gate-array design is a manufacturing method in which the diffused layers, i.e. transistors and other active devices, are predefined and wafers containing such devices are held in stock prior to metallization - in other words, unconnected. The physical design process then defines the interconnections of the final device. For most ASIC manufacturers, this consists of from two to as many as nine metal layers, each metal layer running perpendicular to the one below it. Non-recurring engineering costs are much lower, as photolithographic masks are required only for the metal layers, and production cycles are much shorter, as metallization is a comparatively quick process. Gate-array ASICs are always a compromise as mapping a given design onto what a manufacturer held as a stock wafer never gives 100% utilization. Often difficulties in routing the interconnect require migration onto a larger array device with consequent increase in the piece part price. These difficulties are often a result of the layout software used to develop the interconnect.

Full – Custom Design

By contrast, full-custom ASIC design defines all the photolithographic layers of the device. Full-custom design is used for both ASIC design and for standard product design. The benefits of full-custom design usually include reduced area (and therefore recurring component cost), performance improvements, and also the ability to integrate analog components and other pre-designed - and thus fully verified - components, such as microprocessor cores that form a system-on-chip. The disadvantages of full-custom design can include increased manufacturing and design time, increased non-recurring engineering costs, more complexity in the computer- aided design (CAD) system, and a much higher skill requirement on the part of the design team.