ATME COLLEGE OF ENGINEERING

13th KM Stone, Bannur Road, Mysore - 570 028

DEPARTMENT OF ELECTRICAL & ELECTRONICS ENGINEERING

NOTES

Course Title : Digital Logic Circuits

Course CODE: BEE306A

SEMESTER: III

Academic Year - 2024-25

INSTITUTIONAL VISION AND MISSION

VISION:

- Development of academically excellent, culturally vibrant, socially responsible and globally competent human resources.

MISSION:

- To keep pace with advancements in knowledge and make the students competitive and capable at the global level.
- To create an environment for the students to acquire the right physical, intellectual, emotional and moral foundations and shine as torchbearers of tomorrow's society.
- To strive to attain ever-higher benchmarks of educational excellence.

Department Vision and Mission

Vision:

To create Electrical & Electronics Engineers who excel to be technically competent and fulfill the cultural and social aspirations of the society.

Mission:

- To provide knowledge to students that builds a strong foundation in the basic principles of electrical engineering, problem solving abilities, analytical skills, soft skills and communication skills for their overall development.
- To offer outcome based technical education.
- To encourage faculty in training & development and to offer consultancy through research & industry interaction.

Program Educational Objectives (PEOs)

PEO1:

To produce competent and ethical Electrical and Electronics Engineers who will exhibit the necessary technical and managerial skills to perform their duties in society

PEO2:

To make students continuously acquire and enhance their technical and socio-economic skills

PEO3:

To allow students to embark on R&D activities leading to offering solutions and excel in various career paths.

PEO4:

To produce quality engineers who have the capability to work in teams and contribute to real time projects

Program Outcomes (POs)

Engineering Graduates will be able to:

PO1: Engineering Knowledge: Apply the knowledge of mathematics, science, engineering fundamentals and an engineering specialization to the solution of complex engineering problems.

PO2: Problem Analysis: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

PO3: Design / Development of solutions: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

PO4: Conduct investigations of complex problems: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

PO5: Modern tool usage: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.

PO6: The engineer and society: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

PO7: Environment and sustainability: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

PO8: Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

PO9: Individual and team work: Function effectively as an individual and as a member or leader in diverse teams, and in multidisciplinary settings.

PO10: Communication: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

PO11: Project management and finance: Demonstrate knowledge and understanding of the engineering management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

PO12: Life-long learning: Recognize the need for and have the preparation and ability to engage in independent and lifelong learning in the broadest context of technological change.

Program Specific Outcomes (PSOs)

The students will develop an ability to produce the following engineering traits:

PSO1:Apply the concepts of Electrical & Electronics Engineering to evaluate the performance of power systems and also to control industrial drives using power electronics

PSO2: Demonstrate the concepts of process control for Industrial Automation, design models for environmental and social concerns and also exhibit continuous self- learning

Digital Logic Circuit

	Academic Year: 2024-2025
Department: Electrical and Electronics Engineering	

Course Code	Course Title	Core/Elective	Prerequisite	Contact Hours			Total Hrs/ Sessions
				L	T	P	
BEE306A	**Digital Logic Circuit**	**Elective**	**Basic Electronics**	**3**	**-**	**2**	**40 theory**

Topics Covered as per Syllabus

Module-1

Definition of combinational logic, canonical forms, Generation of switching equations from truth tables, Karnaugh maps-3,4,5 variables, Incompletely specified functions (Don‘t care terms) Simplifying Max term equations, Quine-McCluskey minimization technique, Quine-McCluskey using don‘t care terms, Reduced prime implicants Tables.

Module-2

General approach to combinational logic design, Decoders, BCD decoders, Encoders, digital multiplexers, Using multiplexers as Boolean function generators, Adders and subtractors, Cascading full adders, Look ahead carry, Binary comparators

Module-3

Basic Bistable elements, Latches, Timing considerations, The master-slave flip-flops (pulsetriggered flip-flops): SR flip-flops, JK flip-flops, Edge triggered flip- flops, Characteristic equations

Module-4

Registers, binary ripple counters, synchronous binary counters, Counters based on shift registers, Design of a synchronous counter, Design of a synchronous mod-n counter using clocked T, JK, D and SR flip-flops.

Module-5

Mealy and Moore models, State machine notation, Synchronous Sequential circuit analysis, Construction of state diagrams, counter design.

Memories: Read only and Read/Write Memories, Programmable ROM, EPROM, Flash memory.

List of Text Books

“Digital Logic Applications and Design” by John M Yarbrough, 2011 edition.
“HDL Programming (VHDL and Verilog)” by Nazeih M. Botros, 1 st Edition
“Digital Principles and Design “, Donald D Givone, Tata McGraw Hill Edition,2002.

List of Reference Books

“Logic Design” by RD Sudhaker Samuel

List of URLs, Text Books, Notes, Multimedia Content, etc: https://www.youtube.com/watch?v=VnZLRrJYa2I

MODULE2

Analysis&DesignofCombinationalLogic

Structure

- Objective
- Introduction
- Generalapproach
- Decoders-BCDdecoders,Encoders.
- Digitalmultiplexers-usingmultiplexersasBooleanfunctiongenerators& Design methods ofbuilding blocks of combinational logics
- AddersandSubtractors-Cascadingfulladders
- Lookaheadcarry
- Binarycomparators..
- Outcome
- FutureReadings

Objective

- **Abilitytounderstand,analyzeanddesignvariouscombinationalcircuit.**
-

Introduction

The complex combinational circuits can be designed usingfundamentalcircuits,thisfundamental circuits mean the we have consideredhalf adder, full adder, the decoder. Now, we will read how the combinational circuits can bedesigned using anotherfundamental circuitscalled multiplexer

Generalapproach

Combinational CircuitsA combinational circuit consistsof logic gateswhose outputs, at anytime, are determined by combining the values of the inputs. A combinational circuit consists of logicgates whoseoutputs, at any time, are determined bycombiningthevaluesoftheinputs. For n inputvariables, there are 2 n possiblebinary input combinations. Forn inputvariables, thereare 2 n possible binary input combinations. For each binary combination of the input variables, there is one possible binary value on each output. For each binary combination of the input variables, there is one possible binary value oneach output.

1. **Designacombinationalcircuitthatwillmultiplytwotwo-bitbinaryvalues**

   Solution:

   1. inputvariables(A_1,A_0,B_1,B_0)
      outputvariables(P_3,P_2,P_1,P_0)

   Constructatruthtable

Inputs				Outputs			
A_1	A_0	B_1	B_0	P_3	P_2	P_1	P_0
0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0
0	0	1	0	0	0	0	0
0	0	1	1	0	0	0	0
0	1	0	0	0	0	0	0
0	1	0	1	0	0	0	1
0	1	1	0	0	0	1	0
0	1	1	1	0	0	1	1
1	0	0	0	0	0	0	0
1	0	0	1	0	0	1	0
1	0	1	0	0	1	0	0
1	0	1	1	0	1	1	0
1	1	0	0	0	0	0	0
1	1	0	1	0	0	1	1
1	1	1	0	0	1	1	0
1	1	1	1	1	0	0	1

TheoutputSOPequationsare:

$P_3=f(A_1,A_0,B_1,B_0)=\sum(15)$

$P_2=f(A_1,A_0,B_1,B_0)=\sum(10,11,14)$

$P_1=f(A_1,A_0,B_1,B_0)=\sum(6,7,9,11,13,14)$

$P_0=f(A_1,A_0,B_1,B_0)=\sum(5,7,13,15)$

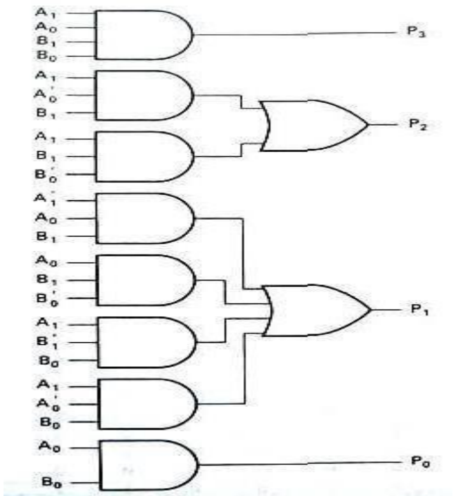
Theindividuallysimplifiedequationsare

$P_3=A_1A_0B_1B_0$

$P_2=A_1A_0'B_1+A_1B_1B_0'$

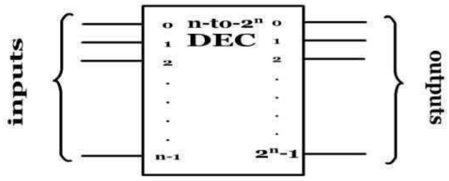
$P_1=A_1'A_0B_1+A_0B_1B_0'+A_1B_1'B_0+A_1A_0'B_0$

$P_0=A_0B_0$



Decoders-BCD decoders,Encoders.

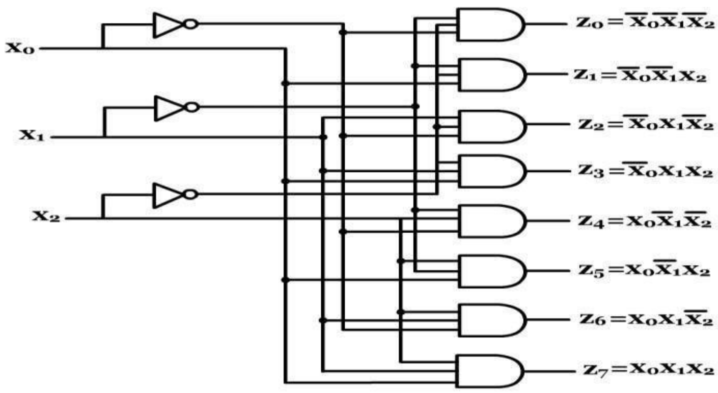
ADecoderisamultipleinput,multipleoutputlogic circuit.Theblockdiagramofadecoder isas shownbelow.



An n-to-2^nline decoder symbol.

Themost commonlyuseddecoderisan–to2^ndecoderwhichhaninputsand2^nOutputlines.

3-to-8decoderlogicdiagram



A 3-to-8 decoder Logic diagram

Inputs			Outputs							
X_2	X_1	X_0	Z_0	Z_1	Z_2	Z_3	Z_4	Z_5	Z_6	Z_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

Truth table.

In this realization shown abovethe threeinputs are assigned x_0,x_1,and x_2, and the eightoutputsare Z_0 to Z_7.

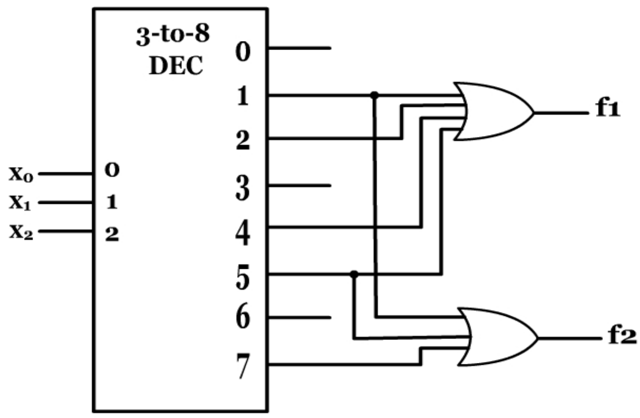
Function specifc decoders also exist which have less than 2^n outputs . examples are 8421 code decoder also called BCD to decimal decoder. Decoders that drive seven segment displays also exist

RealizationofbooleanexpressionusingDecoderandORgate

We see from the above truth table that the output expressions corrwespond to a single minterm. Hence an–to2^n decoder is aminterm generator. Thus byusing OR gates inconjunctionwitha a n–to 2^n decoder boolean function realization is possible.

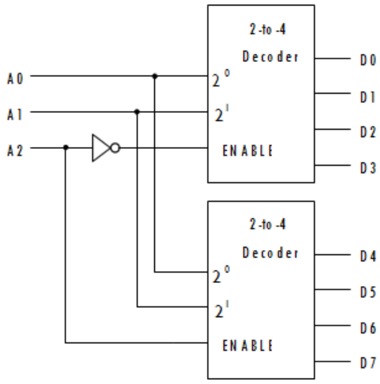
P1:torealizetheBooleanfunctionsgivenbelowusingdecoders…

- F1=Σm(1,2,4,5)
- F2=Σm(1,5,7)



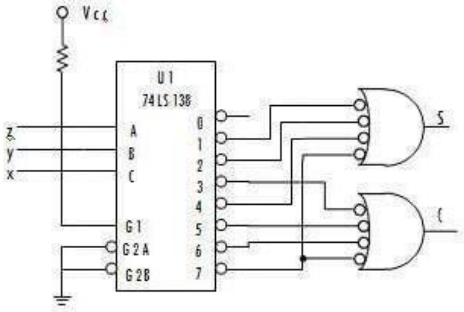
Realisation of boolean expressions

P2:A3-to-8Decoder constructed



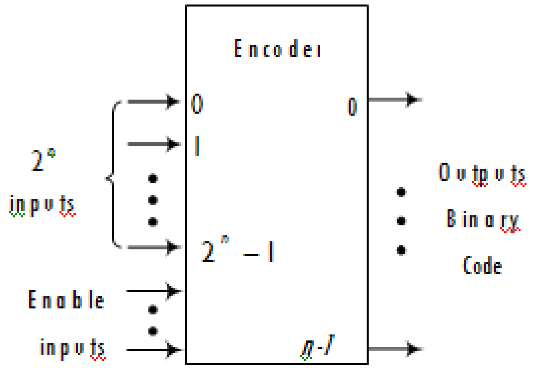
P3:Designabinary3-bitadderwitha74xxx138and NAND gates.

$$S = f(x, y, z) = \sum m(1, 2, 4, 7),\ C f(X, Y, Z)\ \sum m(3, 5, 6, 7)$$



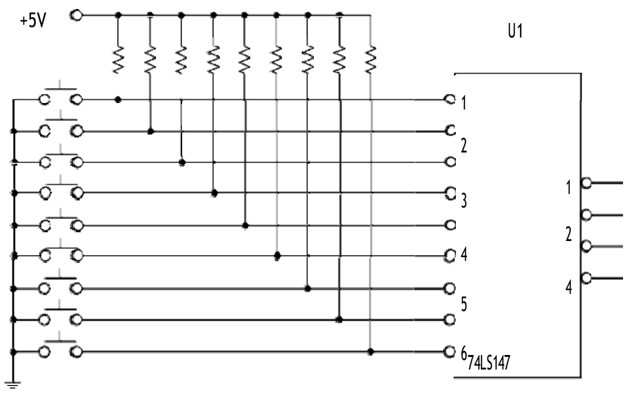
Encoder

Itisainverseofdecoderhaving2^ninputandnoutput.



P4:**Decimal-to-BCDEncoder(74xxx147)**

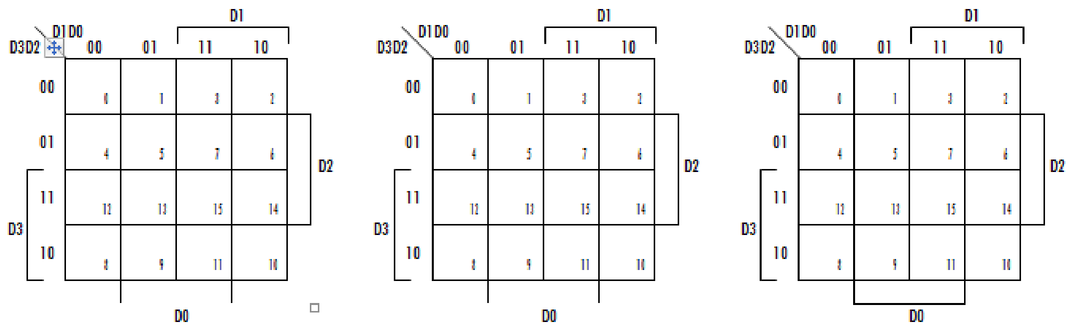
Inputs									Outputs			
1	2	3	4	5	6	7	8	9	D	C	B	A
1	1	1	1	1	1	1	1	1				
0	1	1	1	1	1	1	1	1				
x	0	1	1	1	1	1	1	1				
x	x	0	1	1	1	1	1	1				
x	x	x	0	1	1	1	1	1				
x	x	x	x	0	1	1	1	1				
x	x	x	x	x	0	1	1	1				
x	x	x	x	x	x	0	1	1				
x	x	x	x	x	x	x	0	1				
x	x	x	x	x	x	x	x	0				



priorityencoder

Severalpossibleeventsmayoccurinanindustrialsystem, andyouwanttoidentifyanevent and assignand transmit a code tothe controlunit based on some priority.

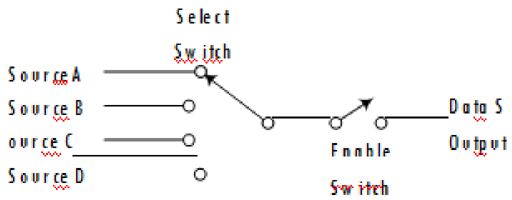
Inputs				Outputs		
D3	D2	D1	D0	A1	A0	/V
		0	0			
0	0	0	1			
0	0	1	X			
0	1	X	X			
1	X	X	X			

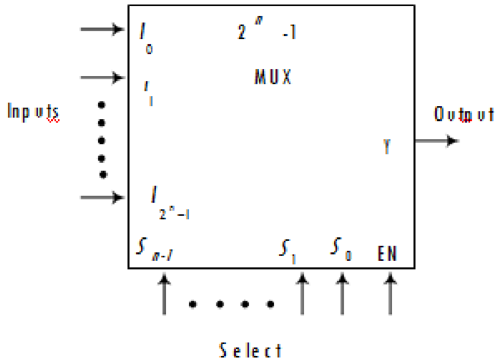


A1 = A2 = IV=

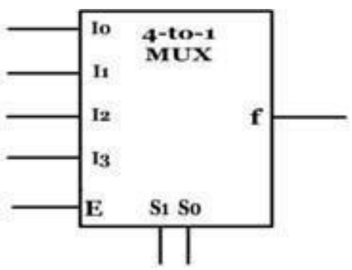
Digital multiplexers-using multiplexers as Boolean function generators. & Design methods of building blocks of combinationallogics.

Multiplexers also called dataselectors are another MSI devices with a wide range of applications in microprocessor and their peripherals design. The followind diagramsshowthe symbol and truth table for the 4-to –1 mux.





P1:4-to-1LineMultiplexer



A 4-to-1 line multiplexer symbol.

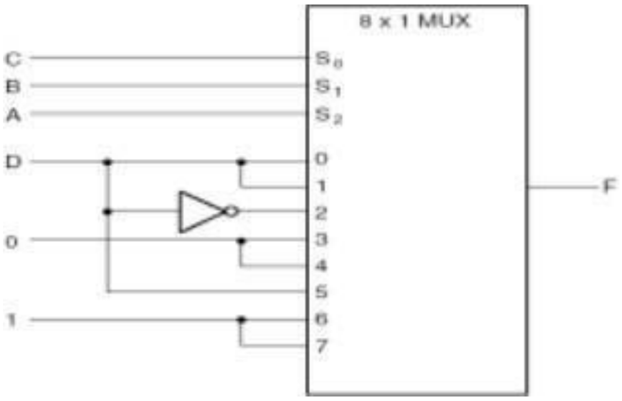
E	S1	So	Io	I1	I2	I3	f
0	x	x	x	x	x	x	0
1	0	0	0	x	x	x	0
1	0	0	1	x	x	x	1
1	0	1	x	0	x	x	0
1	0	1	x	1	x	x	1
1	1	0	x	x	0	x	0
1	1	0	x	x	1	x	1
1	1	1	x	x	x	0	0
1	1	1	x	x	x	1	1

Compressed Truth table

P2:**ConsiderthefunctionF(A,B,C,D)=∑(1,3,4,11,12,13,14,15)**

This functioncanbe implementedwithan8-to-1 line MUX(seeFigure7)A,B, andCare applied to the select inputs as follows: A⇒ S2 , B ⇒ S1, C ⇒ S0

A	B	C	D	F	
0	0	0	0	0	F = D
0	0	0	1	1	
0	0	1	0	0	F = D
0	0	1	1	1	
0	1	0	0	1	F = $\overline{D}$
0	1	0	1	0	
0	1	1	0	0	F = 0
0	1	1	1	0	
1	0	0	0	0	F = 0
1	0	0	1	0	
1	0	1	0	0	F = D
1	0	1	1	1	
1	1	0	0	1	F = 1
1	1	0	1	1	
1	1	1	0	1	F = 1
1	1	1	1	1	



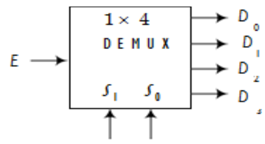
Demultiplexers

• Performtheoppositefunctionofmultiplexers

• Placingthevalueofa singledatainput ontooneofthemultipledataoutputs

• Sameimplementationasdecoderwithenable

• Enableinputofdecoderservesasthedatainput forthedemultiplexer



P1: A1-to-4lineDemux

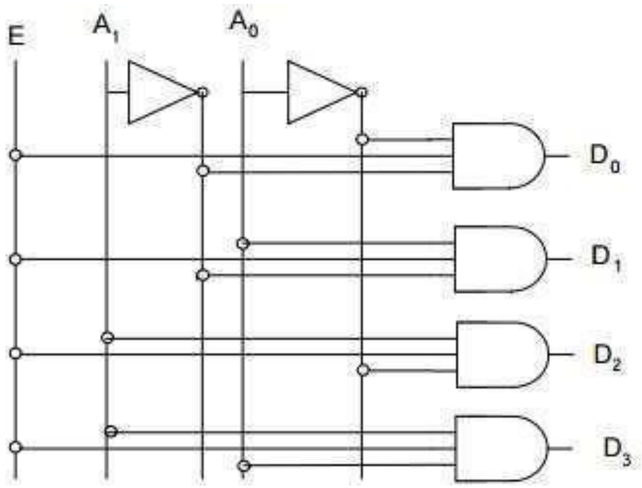
Theinput Eis directed to oneoftheoutputs,as specifiedbythe twoselectlines S1 and S0. D0 = E ifS1S0 = 00 ⇒ D0 = S1' S0' E

D1 = E if S1S0 = 01 ⇒ D1 = S1' S0 E

D2 = E if S1S0 = 10 ⇒ D2 = S1 S0' E

D3 = E ifS1S0 = 11 ⇒ D3 = S1 S0 E

A careful inspection of the Demux circuit shows that it isidentical to a2 to 4 decoder with enable input.

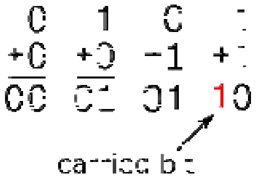


Decimal value	Enable	Inputs		Outputs			
	E	A_1	A_0	D_0	D_1	D_2	D_3
	0	X	X	0	0	0	0
0	1	0	0	1	0	0	0
1	1	0	1	0	1	0	0
2	1	1	0	0	0	1	0
3	1	1	1	0	0	0	1

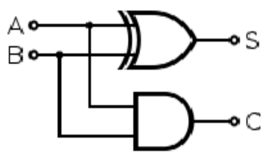
Tablefor1-to-4linedemultiplexer

AddersandSubtractors-Cascadingfulladders

Consideraddingtwobinarynumberstogether:



We see that the bit inthe "two's" column is generated whenthe additioncarried over. A half-adderisacircuitwhichaddstwobitstogetherand outputsthesumofthosetwobits.The half- adder has two outputs: **sum** and **carry**. Sum represents the remainder of the integer division A+B/2, while carry is the result. This canbe expressed as follows:



A	B	A+B	S	C
0	0	0	0	0
0	1	**1**	1	0
1	0	**1**	1	0
1	1	2	0	1

S=AxorB

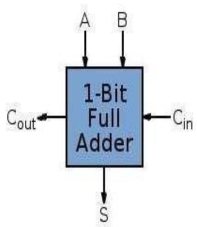
C=AB

FullAdder:

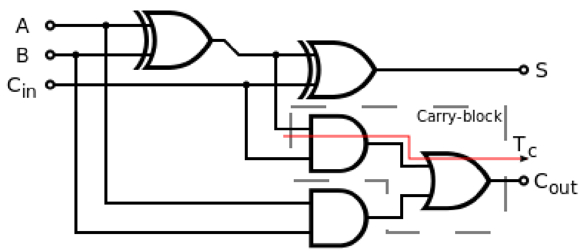
Half-adders have a major limitation inthat they cannot accept a carrybit from a previous stage, meaningthattheycannotbechainedtogethertoadd multi-bitnumbers. However,thetwooutput bitsofahalf-addercanalsor e presenttheresultA+B=3assumandcarrybothbeinghigh.

Assuch,thefull-addercana c ceptthreebitsasaninput.Commonly,onebit isreferredtoasthe carry-in bit. Full adders can be cascaded to produce adders of any number of bits by daisy-chaining the carry of one output tothe input of the next

The full-adder is usually shown as a single unit. The sum output is usually on the bottom on the block, and the carry-out output is on the left, so the devices can be chained together, most significant bit leftmost:



LogicSymbolofFulladder

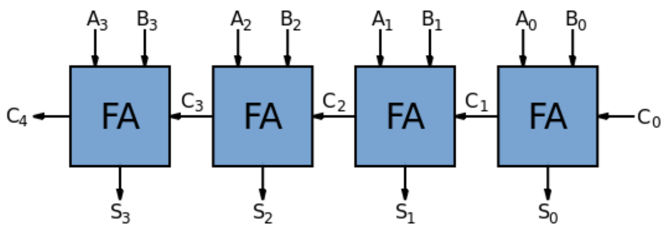


A	B	C_{in}	$A+B+C_{in}$	S	C_{out}
0	0	0	0	0	0
0	0	1	1	1	0
0	1	0	1	1	0
0	1	1	2	0	1
1	0	0	1	1	0
1	0	1	2	0	1
1	1	0	2	0	1
1	1	1	3	1	1

RippleCarryAdder:

A ripple carry adder is simply several full adders connected in a series so that the carry must propagate through every full adder before the addition iscomplete. Ripplecarry adders require the least amount of hardware ofall adders, but they are theslowest.

Thefollowing diagram shows a four-bit adder, which adds the numbersA[3:0]and B[3:0], as well as a carry input, together to produce S[3:0] and the carry output



PropagationDelayinFullAdders

Real logic gates do not react instantaneously to the inputs, and therefore digital circuits have a maximum speed. Usually, the delay through a digital circuit is measured in gate-delays, as this allows the delay of a design to be calculated for different devices. AND and OR gates have a nominal delay of 1 gate-delay, and XOR gates have a delay of 2, because they are really made up ofa combination of ANDs and ORs.

Afulladderblockhasthefollowingworstcasepropagationdelays:

- From$A_i$or$B_i$toC_{i+1}:4gate-delays(XOR→AND →OR)
- From$A_i$orB_i toS_i:4gate-delays(XOR→XOR)
- From$C_i$toC_{i+1}:2gate-delays(AND→OR)
- From$C_i$toS_i:2gate-delays(XOR)

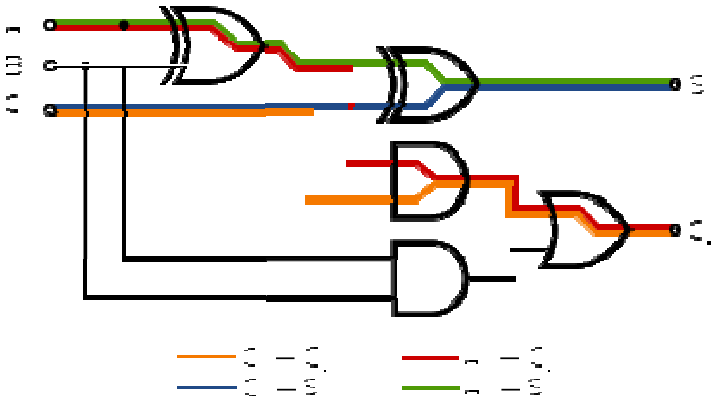
Becausethecarry-outofone stageisthenext'sinput, theworstcasepropagationdelayisthen:

- 4gate-delaysfromgeneratingthefirstcarrysignal(A_0/B_0→C_1).
- 2gate-delaysperintermediate stage(C_i→C_{i+1}).

- 2gate-delaysatthelaststagetoproduceboththesumand (

  Carry-outoutputs(C_{n-1}→$C_n$andS_{n-1}).

Soforan$n$-bitadder,wehaveatotalpropagationdelay,$t_p$of:



$$t_p = 4 + 2(n-2) + 2 = 2n + 2$$

This islinearin n, andfor a 32-bit number, would take 66 cycles to complete the calculation.This is rather slow, and restricts the word length in our device somewhat. We would like to find ways to speed it up.

Lookaheadcarry

A fast method of adding numbersiscalled carry-lookahead. Thismethod doesn'trequire thecarry signal to propagate stage by stage, causing a bottleneck. Instead it uses additional logic to expedite the propagation and generation of carry information, allowing fast additionattheexpense of more hardware.

In a ripple adder, each stage compares the carry-in signal, C_i, with the inputs A_i and B_i and generates a carry-out signal C_{i+1} accordingly. In a carry-lookahead adder, we define two new functions.

The generate function, G_i, indicates whether that stage causes a carry-out signal$C_i$tobegenerated if no carry-insignal exists. This occurs ifboththe addends contain a 1 inthat bit:

$G_i = A_i.B_i$

The propagate function, P_i, indicates whether a carry-in to the stage is passed to the carry-out for the stage. This occurs ifeither the addends have a 1 inthat bit

$P_i = A_i + B_i$

Note that both these values can be calculated from the inputs in a consta ı
delay. Now, the carry-out from a stage occurs if that stage generates a carry ($G_i = 1$) or there is a carry-in and the stage propag ŧes the carry($P_i \cdot C_i = 1$)

$C_{i+1} = A_iB_i + A_iC_i + B_iC_i$ $C_{i+1} = A$

$_i B_i + C_i (A_i + B_i)$ $C_{i+1} = G_i$

$+ P_i C_i$

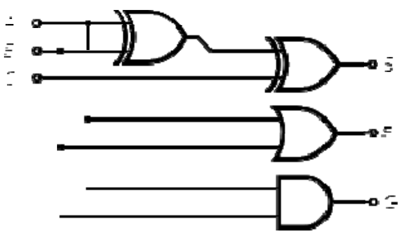
Truthtable

A_i	B_i	C_i		G_i	P_i	C_{i+1}
0	0	0		0	0	0
0	0	1		0	0	0
0	1	0		0	1	0
0	1	1		0	1	1
1	0	0		0	1	0
1	0	1		0	1	1
1	1	0		1	1	1
1	1	1		1	1	1

$$c_{i+1} = G_i + P_i c_i$$
$$c_{i+1} = G_i + P_i (G_{i-1} + P_{i-1} c_{i-1})$$
$$c_{i+1} = G_i + P_i G_{i-1} + P_i P_{i-1} (G_{i-2} + P_{i-2} c_{i-2})$$
$$c_{i+1} = \vdots$$
$$c_{i+1} = G_i + P_i G_{i-1} + P_i P_{i-1} G_{i-2} + P_i P_{i-1} P_{i-2} G_{i-3} + \ldots + P_i P_{i-1} \cdots P_1 P_0 c_0$$

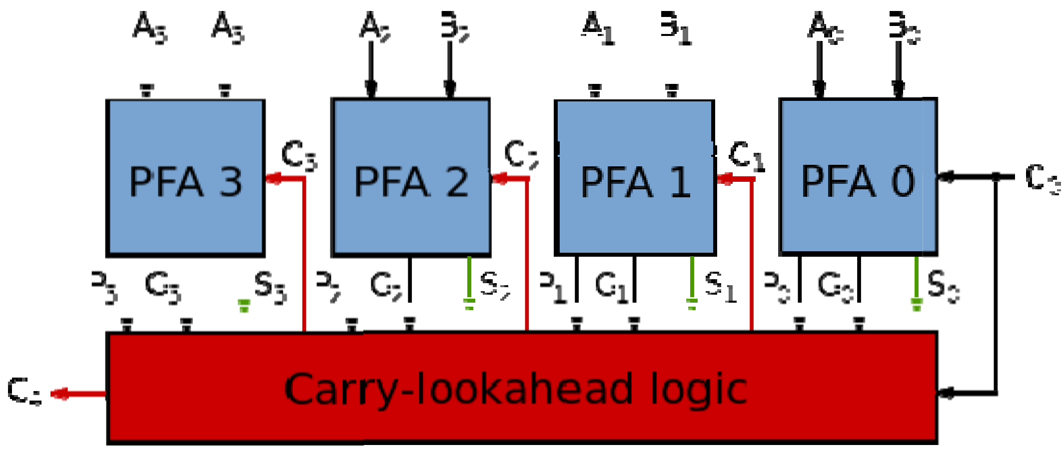
Note that this does not require the carry-out signals from the previous stages, so we don't have to waitfor changes to ripple through the circuit.In fact, a given stage'scarry signalcanbe computed once the propagate and generate signals are ready with only two more gate delays (one AND and one OR). Thus the carry-out for a given stage can be calculated in constant time, and therefore so can the sum.

The *S*, *P*, and *G* signals are all generated by a circuit called a "partial full adder" (PFA), which is similar to a full adder.



For a slightly smaller circuit, the propagate signal can be taken as the outputof the firstXOR gate instead of using a dedicated OR gate, because if both A and B are asserted, thegeneratesignal will force a carry. However, this simplifiaction meansthat the propagate signal will take two gate delays to produce, rather than justone.

Acarrylookaheadadderthencontains *n*PFAsand the logictoproducecarriesfrom thestage propagate and generate signals:

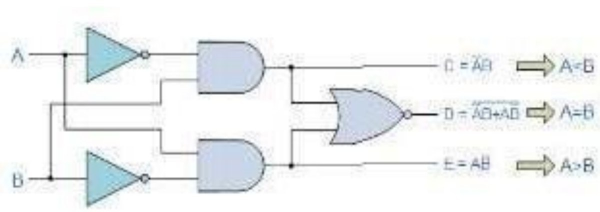


Two numbers can therefore be added in constant time, O(1), of just6gatedelays, regardless ofthe length, *n* of the numbers. However, this requires AND and OR gates with up to*n* inputs. If logic gates are available with a limited number of inputs, trees will need to be constructed to compute these, and the overall computation time islogarithmic, O(ln(*n*), which isstill muchbetter thanthe linear time for ripple adders.

Binarycomparators

Anothercommonand veryusefulcombinationallogiccircuit isthat ofthe **Digital Comparator** circuit. Digital or BinaryComparators are made up from standardAND,NORandNOTgatesthatcomparethedigitalsignalspresentattheirinput terminals and produce an output depending uponthe condition of those inputs.

Anothercommonand veryusefulcombinationallogiccircuit isthat ofthe **Digital Comparator** circuit. Digital or BinaryComparators are made up from standardAND,NORandNOTgatesthatcomparethedigitalsignalspresentattheirinput terminals and produce an output depending uponthe condition ofthose inputs.



For example, along with being able to add and subtract binary numbers we need to be able to compare them and determine whether the value of input A isgreater than, smallerthan orequal to the value at input B etc. The digital comparatoraccomplishesthisusingseverallogic gates that operate on the principles of *Boolean Algebra*.

ThereAretwomaintypesof**Digital Comparator**availableandthese are.

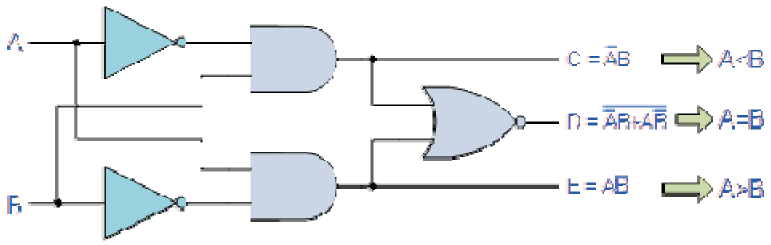
- 1. Identity Comparator – an *Identity Comparator* is a digital comparator that has only one output terminal for when A= B either “HIGH” A= B = 1 or “LOW”A= B = 0
- 2. Magnitude Comparator – a *Magnitude Comparator* is a digitalcomparator which has threeoutputterminals,oneeachforequality,A=Bgreaterthan,A > Bandless than A < B

The purpose of a **Digital Comparator** is to compare a set of variables or unknown numbers, for example A (A1, A2, A3, …. An, etc) against that of a constant or unknown value such as B (B1, B2, B3, …. Bn, etc) and produce an output condition or flag depending upon the result of the comparison. For example, a magnitude comparator of two 1-bits, (A andB)inputswouldproduce the following three output conditions when compared to each other.

Whichmeans:AisgreaterthanB,AisequaltoB,andAislessthan B

Thisis useful if we want to compare twovariables andwant to producean output when any of the above three conditions are achieved. For example, produce an output from a counter when a certain count number is reached. Consider the simple 1-bit comparatorbelow.

1-bitDigitalComparatorCircuit



Thentheoperationofa1-bitdigitalcomparatorisgiveninthefollowingTruthTable.

DigitalComparatorTruthTable

Inputs		Outputs		
B	A	A>B	A= B	A<B
0	0	0	1	0
0	1	1	0	0

1	0	0	0	1
1	1	0	1	0

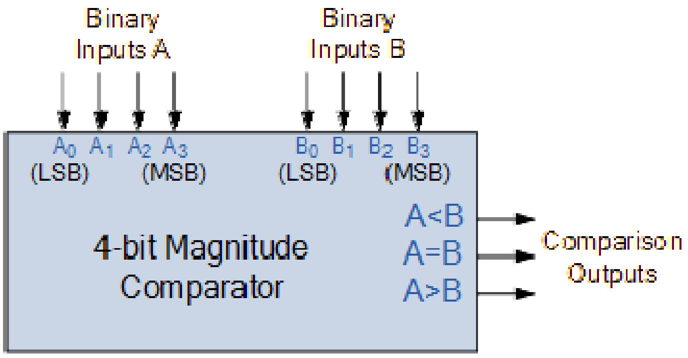
You may notice two distinct features about the comparator from the above truth table. Firstly, the circuit does not distinguish between either two "0" or two "1"'s as an output A = B is produced when they are bothequal, eitherA = B = "0"or A = B = "1".Secondly,the output condition for A = B resembles thatofacommonlyavailablelogicgate,the Exclusive-NOR or Ex- NOR function(equivalence) on each ofthe n-bits giving: $Q = A \oplus B$

Digital comparators actually use Exclusive-NOR gates within their design for comparing their respective pairs of bits. When we are comparing two binary or BCD values or variables against each other, we are comparing the "magnitude" of these values, a logic "0" against a logic "1" which is where the term **Magnitude Comparator** comes from.

As well as comparingindividual bits, wecan design largerbitcomparatorsbycascading together n of these and produce a n-bit comparator just as we didfor the n-bit adderin the previous tutorial. Multi-bit comparators can beconstructed to compare whole binary or BCD wordstoproduce anoutput if one word is larger, equalto or lessthantheother.

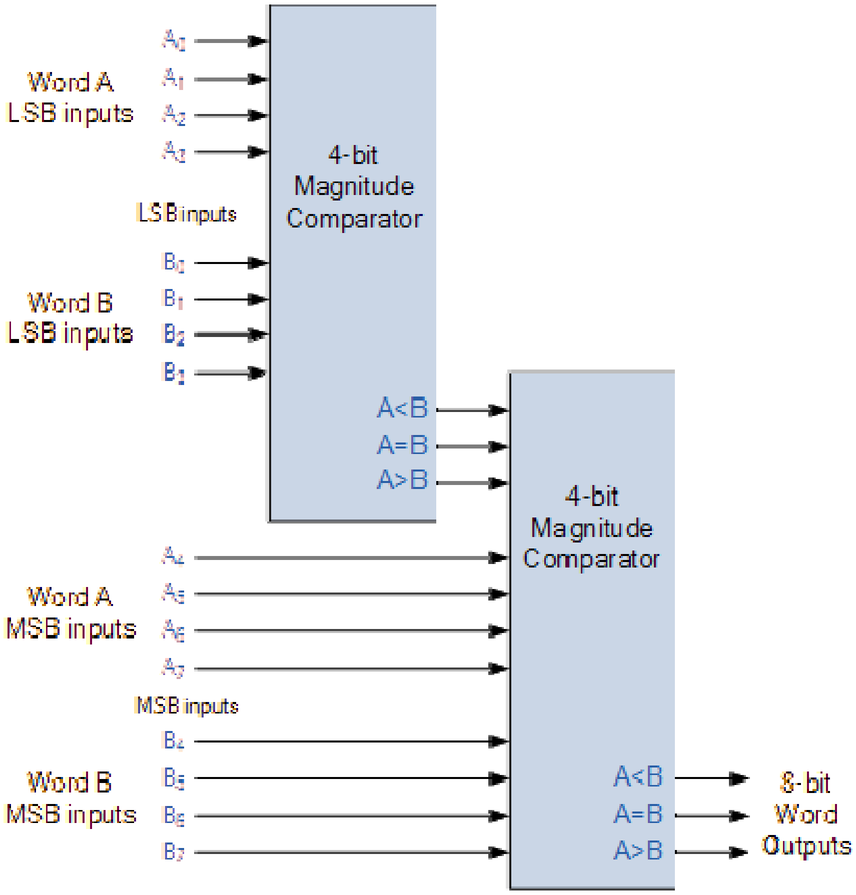
A very good example of this is the 4-bit **Magnitude Comparator**. Here, two 4-bit words ("nibbles")are compared to each otherto produce the relevantoutputwith one word connected to inputs Aand the other tobe compared against connected to input B as shownbelow.

4-bitMagnitudeComparator



Some commercially available digital comparators such as the TTL 74LS85 or CMOS 4063 4-bit magnitude comparator have additional input terminals that allowmore individual comparators to be "cascaded" together to compare words larger than 4-bits with magnitude comparators of "n"- bits being produced. These cascading inputsare connected directly to the correspondingoutputs ofthe previous comparator as shown to compare 8, 16 or even32-bitwords.

8-bitWordComparator



When comparing large binary or BCD numbers like the example above, to save time the comparator starts by comparing the highest-order bit (MSB) first. If equality exists, A = B then it compares the next lowest bit and so on until it reaches the lowest-order bit,(LSB).Ifequality still exists then the two numbers are defined as beingequal.

If inequality is found, either A > B or A < B the relationship between the two numbers is determined and the comparison between any additional lower order bits stops.**Digital Comparator** are used widely in Analogue-to-Digital converters, (ADC) and Arithmetic Logic Units, (ALU) to perform a variety of arithmetic operations.

Outcome

- **Cancreateaappropriatetruthtablefrom thedescriptionof combinationallogic function.**
- **Abletodesignanylogiccircuit usingMUX,DEMUX,encodersanddecoders based onthe application such that the gates used in a circuits arereduced.**

Future Readings

http://nptel.ac.in/courses/117105080/

https://www.youtube.com/watch?v=VnZLRrJYa2I“Lo

gic Design” byRD SudhakerSamuel

“DigitalLogicApplicationsandDesign”byJohnMYarbrough,2011edition