COURSE MATERIAL

UNIT 4

SEC1207-DIGITAL LOGIC CIRCUITS

SYLLABUS

UNIT IV DIGITAL LOGIC FAMILIES **9 Hrs.**

Classification and characteristics of logic family - Bipolar logic family - Saturated logic family - RTL, DTL,DCTL, I^{2}L,TTL, HTL - Non saturated family - SchottkyTTL, ECL - Unipolar family - MOS, CMOS logic families. Tristate logic. Interfacing of CMOS and TTL families. Comparison of logic families

TABLE OF TOPICS

4.1 Classification and characteristics of logic family

The classification of digital ICs is given in Table

IC Classification	*Equivalent individual basic gates*	*Number of components*
Small-scale integration (SSI)	Less than 12	Up to 99
Medium-scale integration (MSI)	12–99	100–999
Large-scale integration (LSI)	100–999	1,000–9,999
Very large-scale integration (VLSI)	Above 1,000	Above 10,000

The various characteristics of digital ICs used to compare their performances are:

1. Speed of operation,
2. Power dissipation,
3. Figure of merit,
4. Fan-out,
5. Current and voltage parameters,
6. Noise immunity,
7. Operating temperature range,
8. Power supply requirements, and
9. Flexibilities available.

Speed of operation

The speed of a digital circuit is specified in terms of the propagation delay time. The input and output waveforms of a logic gate are shown in Fig. 4.1. The delay times are measured between the 50 per cent voltage levels of input and output waveforms. There are two delay times: tpHL , when the output goes from the HIGH state to the LOW state and tpLH, corresponding to theoutput making a transition from the LOW stateto the HIGH state. The propagation delay time of the logic gate is taken as the average of these two delay times.

Power dissipation

Eachgateisconnectedtoapowersupply VCC (VDDinthecaseofCMOS).It drawsacertainamountofcurrentduringitsoperation.Sinceeachgatecanbe ina High,TransitionorLowstate,thereare threedifferentcurrentsdrawnfrompower supply.

- ICCH:CurrentdrawnduringHIGH state.
- ICCT:CurrentdrawnduringHIGH toLOW,LOWtoHIGH transition.
- ICCL:CurrentdrawnduringLOWstate.

ForTTL,ICCTthetransitioncurrentisnegligible,incomparisontoICCHand ICCL.IfweassumethatICCHandICCLare equalthen,

AveragePowerDissipation=Vcc* (ICCH+ ICCL)/2

ForCMOS,ICCHandICCLcurrentisnegligible, incomparison toICCT.Sothe Averagepowerdissipationis calculatedasbelow.

AveragePowerDissipation=Vcc*ICCT.

Figure of merit

Fan-In

Fan-inisthenumberofinputsagate has,like a twoinputANDgate hasfan-inof two,athree inputNANDgateasa fan-in ofthree. SoaNOTgatealwayshasa fan-inofone.Thefigurebelowshowstheeffectoffan-inonthedelayoffered by agateforaCMOSbasedgate.Normallydelayincreasesfollowingaquadratic functionof fan-in.

Fan-out

Thenumberofgatesthateachgatecandrive,whileprovidingvoltagelevelsin theguaranteed range,iscalledthestandardloadorfan-out.Thefan-outreally dependsontheamount ofelectriccurrentagatecansourceorsinkwhiledriving other gates.Theeffectsofloadingalogic gateoutputwithmorethanitsratedfan- outhasthefollowingeffects.

Logiclevels

Logiclevelsarethevoltagelevelsforlogichighandlogiclow.

- VO_{Hmin}:TheminimumoutputvoltageinHIGHstate (logic'1').VO_{Hmin}is2.4 V forTTLand4.9 V forCMOS.
- VO_{Lmax}:ThemaximumoutputvoltageinLOWstate(logic'0').VO_{Lmax}is0.4 V forTTLand0.1 V forCMOS.
- VI_{Hmin}:Theminimuminputvoltageguaranteedtoberecognisedaslogic1. VI_{Hmin}is2 V forTTLand3.5 V forCMOS.
- VI_{Lmax}:Themaximuminputvoltageguaranteedtoberecognisedaslogic0. VI_{Lmax}is0.8 V forTTLand1.5 V forCMOS.

Currentlevels

- IO_{Hmin}: ThemaximumcurrenttheoutputcansourceinHIGHstatewhilestill maintainingtheoutputvoltageaboveVO_{Hmin}.
- IO_{Lmax}:ThemaximumcurrenttheoutputcansinkinLOWstatewhilestill maintainingtheoutputvoltagebelowVO_{Lmax}.
- I_{Imax} :Themaximumcurrentthatflowsintoaninputinanystate(1µAfor CMOS).

NoiseMargin

Gatecircuitsareconstructedtosustainvariationsininputandoutputvoltage levels.Variationsareusuallytheresultofseveraldifferentfactors.

- Batterieslosetheirfullpotential,causingthesupplyvoltagetodrop
- High operating temperatures may cause a drift in transistor voltage and current characteristics
- Spuriouspulses maybeintroducedonsignal linesby normalsurgesofcurrent inneighbouringsupplylines.

All these undesirable voltage variations that are superimposed on normal operatingvoltagelevelsarecallednoise.Allgatesaredesignedtotoleratea certainamount ofnoiseontheirinputandoutputports.Themaximum noise voltage levelthatistolerated byagateiscallednoisemargin.Itderives fromI/P- O/Pvoltagecharacteristic, measuredunderdifferentoperatingconditions.It's normallysuppliedfrommanufacturerinthegatedocumentation.

- LNM(Lownoisemargin):Thelargest noiseamplitudethatisguaranteednot tochange theoutputvoltage levelwhensuperimposedontheinputvoltageof the logicgate(whenthisvoltage isintheLOWinterval). LNM=VI_{Lmax}- VO_{Lmax}.
- HNM(Highnoisemargin):Thelargestnoiseamplitudethatisguaranteed nottochange theoutputvoltage levelifsuperimposedontheinputvoltageof thelogicgate(whenthisvoltage isintheHIGHinterval). HNM=VO_{Hmin}- VI_{Hmin}

t_r(Risetime)

ThetimerequiredfortheoutputvoltagetoincreasefromV$_{IL}$maxtoV$_{IH}$min.

t_f(Falltime)

ThetimerequiredfortheoutputvoltagetodecreasefromV$_{IH}$mintoV$_{IL}$max.

t_p (Propagationdelay)

Thetimebetweenthelogictransitiononaninputandthecorrespondinglogic transitionontheoutputofthelogicgate.Thepropagationdelay ismeasuredat midpoints.

4.1.1 Bipolar logic family

The main elements of a bipolar IC are resistors, diodes (which are also capacitors) and transis tors. Basically, there are two types of operations in bipolar ICs:

1.Saturated, and
2.Non-saturated.

In saturated logic, the transistors in the IC are driven to saturation, whereas in the case of non-saturated logic, the transistors are not driven into saturation.

The saturated bipolar logic families are:

1. Resistor–transistor logic (RTL),
2. Direct–coupled transistor logic (DCTL),
3. Integrated–injection logic (I2L),
4. Diode–transistor logic (DTL),
5. High–threshold logic (HTL), and
6. Transistor-transistor logic (TTL).

4.1.1.1 Saturated logic family

The non-saturated bipolar logic families are:

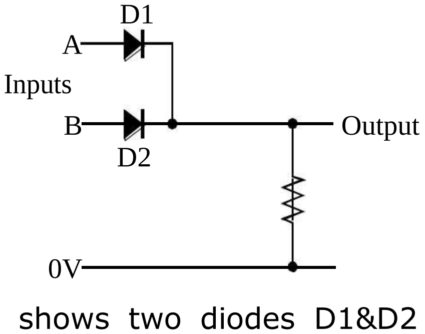
1. SchottkyTTL, and
2. Emitter-coupled logic (ECL).

Diode Logic (DL)

Diode logic gates use diodes to perform AND and OR logic functions. Diodes have the property of easily passing an electrical current in one direction, but not the other. Thus, diodes can act as a logical switch.

Diode logic gates are very simple and inexpensive, and can be used effectively in specific situations. However, they cannot be used extensively, as they tend to degrade digital signals rapidly. In addition, they cannot perform a NOT function, so their usefulness is quite limited.

Diode OR Circuit



Inputs		Output
B	A	
0V	0V	0V
0V	+5V	+4.3V
+5V	0V	+4.3V
+5V	+5V	+4.3V

The above figure shows two diodes D1&D2 with a resistor load. The table shows the voltage truth table for the circuit.

With **both inputs at 0V**, the output is at 0V.

With **either diode input at +5V**, the respective diode will be forward biased and current will flow through the diode and the load resistor. For silicon junction diodes, the output voltage will be approximately 0.7V less than the input voltage, due to the voltage drop across the forward biased diode.

With **both inputs at +5V**, the output will still be 0.7V less than the supply value (that is, 4.3V).

Note: If the input voltages for inputs are different, then the the two output voltages will depend on the inputs. Table below shows the voltage truth table for inputs of +3V& +5V.

Inputs		Output
B	A	
0V	0V	0V
0V	+3V	+2.3V
+5V	0V	+4.3V
+5V	+3V	+4.3V

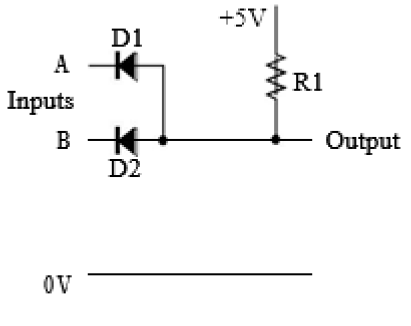
Converting the voltage levels of the above truth table to logic levels 0 & 1 and using **positive** logic gives the truth table shown below

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

From the above table it can be seen that the output is at logic 1 if input A OR B OR both inputs are at logic 1. The circuit therefore performs the **OR** function.

Circuits having inputs and an output, the output depending on the logic states of the inputs, are referred to as **gates**. A circuit with the OR characteristic is referred to as an **OR gate**.

Such a gate with two inputs would be referred to as a 2-input OR gate, one with three inputs a 3-input OR gate, etc.



Inputs		Output
B	A	
0V	0V	0.7V
0V	+5V	0.7V
+5V	0V	0.7V
+5V	+5V	+5V

The above figure shows two diodes D1&D2 and resistor R1 forming an AND circuit. The table shows the voltage truth table for the circuit.

With **both inputs at 0V**, both the diodes will be forward biased and the output will be 0.7V, this being the voltage drop across a forward biased silicon junction diode.

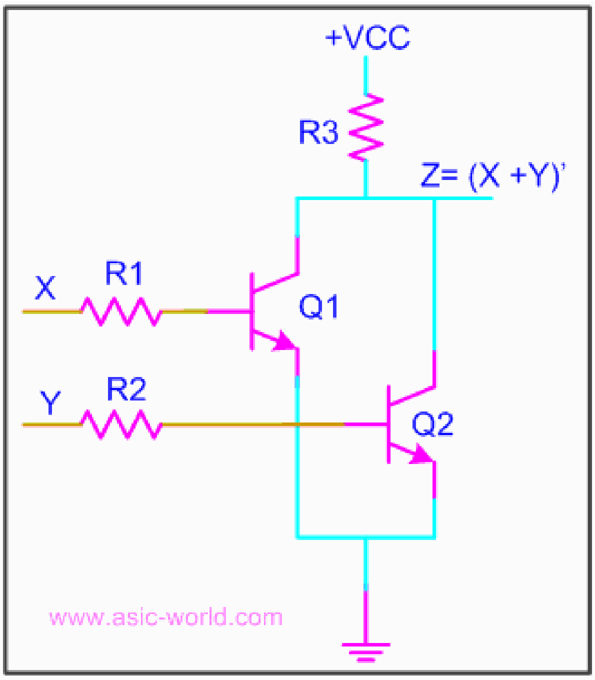
With **either diode input at 0V**, the respective diode will be forward biased and the output voltage will again be approximately 0.7V.

With **both inputs at +5V**, both diodes will be reverse biased and the output voltage will be the supply value (that is, +5V) provided there is no load resistor connected to the output circuit.

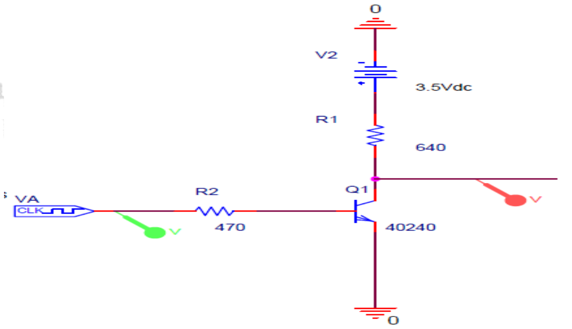
With a load resistor R_L connected to the output circuit the output voltage will be reduced to the value 5 x R_L/(R1 + R $_L$). This is illustrated in the figure below

4.1.1.1.1 ResistorTransistorLogic(*RTL)*

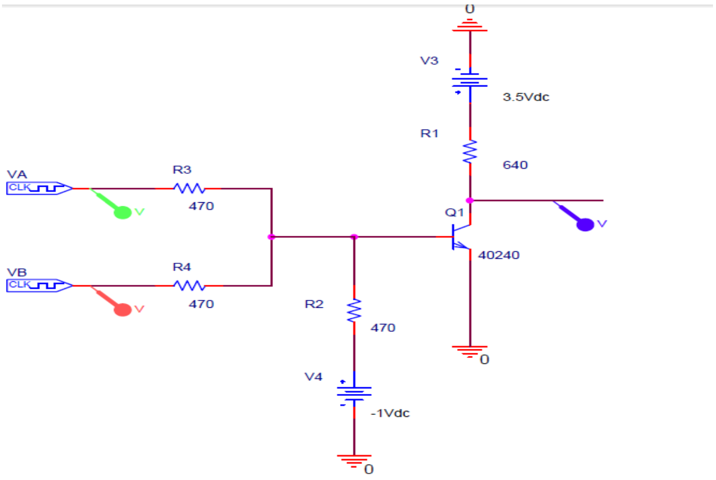
InRTL(resistor transistorlogic), all thelogic are implementedusing resistorsand transistors.Onebasicthingabout thetransistor(NPN),isthat HIGHatinput causes outputtobeLOW(i.e. likea inverter).BelowistheexampleofafewRTLlogiccircuits.



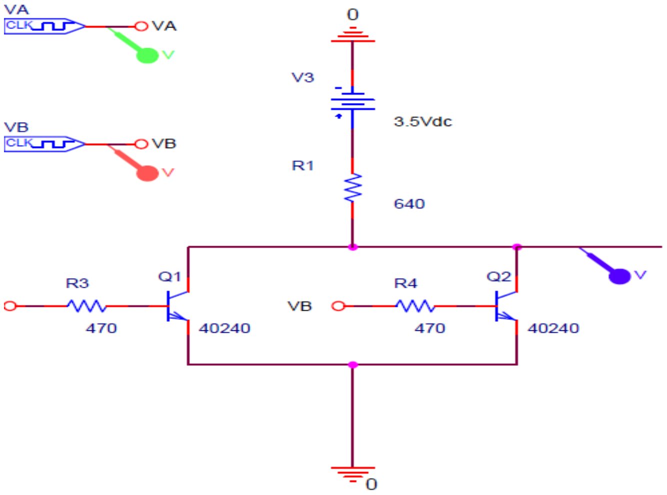
AbasiccircuitofanRTL NORgateconsistsoftwotransistorsQ1andQ2, connectedasshowninthefigureabove.WheneitherinputXorYisdriven HIGH,thecorrespondingtransistorgoestosaturationandoutputZispulledto LOW.



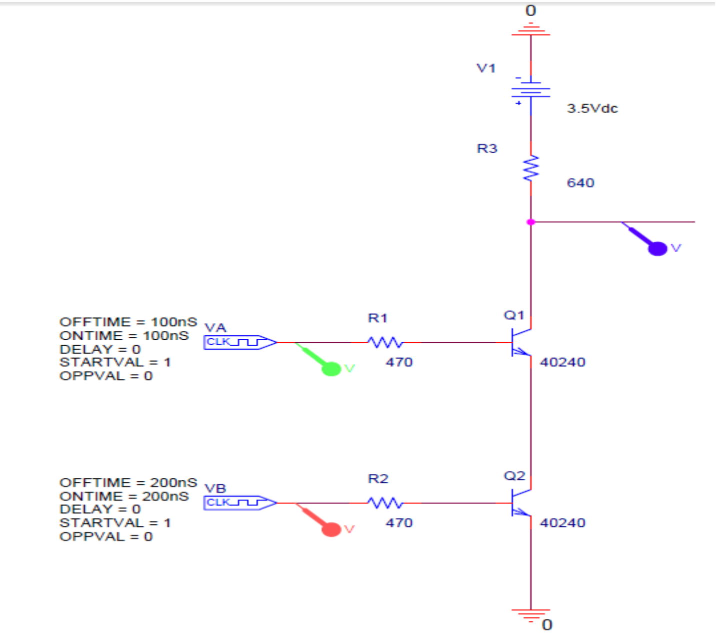
RTL implementation of the NOT function



RTL implementation of the NOR function



RTL implementation of the NOR function (AGC)

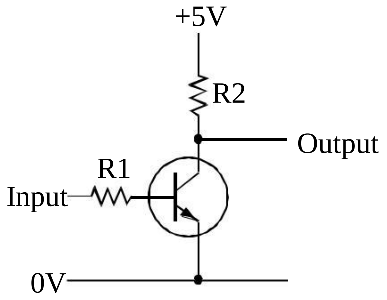


NAND function

4.1.1.1.2 DTL

By letting diodes perform the logical AND or OR function and then amplifying the result with a transistor, we can avoid some of the limitations of RTL. DTL takes diode logic gates and adds a transistor to the output, in order to provide logic inversion and to restore the signal to full logic levels.

Transistor Application as a Switch



Input	Output
0V	+5V
+5V	0V

Voltage
Truth Table

Input	Output
0	1
1	0

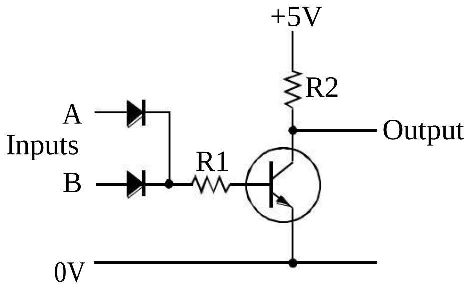
Positive Logic
Truth Table

The above figure shows an NPN transistor circuit with base feed resistor R1 and collector load resistor R2. The table shows the voltage and positive logic truth tables for the circuit.

With the input at 0V, there will be no base current and hence the transistor will be turned OFF.There will be no collector current, so the output voltage from the collector will be the supply value (+5V), provided there is no load resistance connected to the output.

With the input at +5V, there will be a current flow in the base-emitter circuit and the transistorwill be turned ON. The value of R1 is arranged to allow sufficient collector current to flow for the collector voltage to fall to approximately zero. Under this condition the transistor is said to be **saturated**.

Diode-Transistor NOR Gate



Inputs		Output
B	A	
0V	0V	+5V
0V	+5V	0V
+5V	0V	0V
+5V	+5V	0V

Voltage
Truth Table

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

Positive Logic
Truth Table

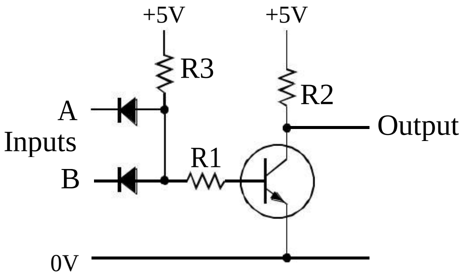
The above figure shows a 2-input diode-transistor circuit and the table shows the voltage and logic truth tables for the circuit.

With **both inputs at 0V**, the transistor will be turned OFF and the output voltage will be at +5V.

With **either or both inputs at +5V**, the transistor will be saturated and the output will be at 0V.

The output states are the inverse of those for an OR gate. That is, it is a NOT OR gate and is referred to as a **NOR** gate.

Diode-Transistor NAND Gate



Inputs		Output
B	A	
0V	0V	+5V
0V	+5V	+5V
+5V	0V	+5V
+5V	+5V	0V

Voltage
Truth Table

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

Positive Logic
Truth Table

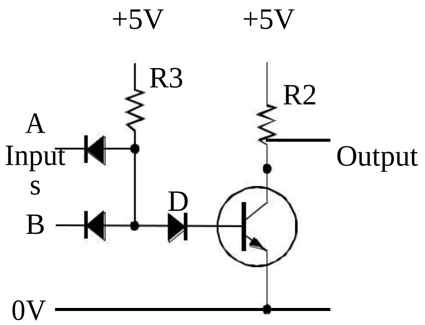
The above figure shows a 2-input diode-transistor circuit and the table shows the voltage and logic truth tables for the circuit.

With **either or both inputs at 0V**, the junction of R1&R3 will be held at 0.7V, due to the voltage drop across the forward biased diode. The transistor will be turned OFF and hence the output voltage will be at +5V.

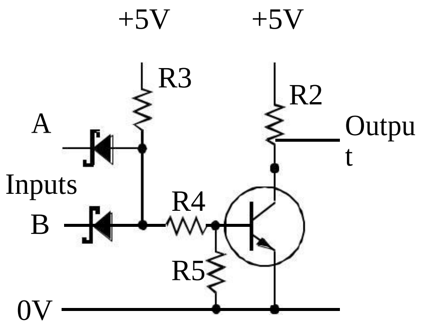
With **both inputs at +5V**, the transistor base - emitter circuit will be fed via R3&R1, the transistor will be saturated and the output will be at 0V.

The output states are the inverse of those for an AND gate. That is, it is a NOT AND gate and is referred to as a **NAND** gate.

In practice, to ensure that the transistor base voltage is less than 0.7V with either input at 0V, the resistor R1 is normally replaced with a diode as shown below



Circuit using voltage shifting diode



Circuit using Schottky diodes

Effect of Logic Convention on Gate Characteristics

Inputs		Output
B	A	
0V	0V	0V
0V	+5V	0V
+5V	0V	0V
+5V	+5V	+5V

Voltage Truth Table

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

Positive Logic

Truth Table

AND Gate

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

Negative Logic

Truth Table

OR Gate

The above tables show the voltage truth table characteristics of a gate and also the logic truth tables for positive and negative logic conventions.

For **positive** logic the gate has an output of logic 1 only with both inputs at logic 1 and therefore represents an **AND** gate.

For **negative** logic convention, the gate has an output of logic 1 when either or both inputs are at logic 1 and therefore represents an **OR** gate.

Inputs		Output
B	A	
0V	0V	0V
0V	+5V	+5V

Inputs		Output
B	A	
0	0	0
0	1	1

Inputs		Output
B	A	
1	1	1
1	0	0

+5V	0V	+5V	1	0	1	0	1	0
+5V	+5V	+5V	1	1	1	0	0	0

Voltage Truth Table

Positive Logic Truth Table OR Gate

Negative Logic Truth Table AND Gate

The above tables show the voltage truth table for a gate and its corresponding positive and negative logic truth tables. The gate represents an **OR** gate for **positive** logic and an **AND** gate for **negative** logic.

4.1.1.1.3 DCTL

In the RTL gate of Fig. 4.4, if the base resistors R_B are omitted, we obtain what is known as the direct-coupled transistor logic (DCTL) gate, in which the inputs are directly coupled to the bases. This circuit performs positive NOR logic and the voltages corresponding to logic 1 and 0 levels are $V_{BE,\text{sat}}$(~ 0.8 V) and $V_{CE,\text{ sat}}$ (~ 0.2 V) respectively. The separation between the logic 1and0levelvoltages,whichisreferredtoasthe*logicswing*,isverysmall($V_{BE,\text{sat}}$–$V_{CE,\text{sat}}$= 0.6 V). Therefore, the noise margin of this circuit is very poor.

Although the DCTL is simpler than RTL, it never became popular because of the problem of *current hogging*. The gate should be able to drive the transistors of the load gates to saturation corresponding to logic level 1.

This does not pose any problem if all the transistors have same input characteristics but, unfortunately, the input characteristics differ due to the manufacturing tolerances of different IC packages operating at different temperatures. Owing to these differences, the saturation voltages of the load transistors may be different. Let the base-emitter voltages of the transistors corresponding to saturation be 0.78, 0.79, and 0.80 V. The transistor with the base–emitter voltage of 0.78 V, when it enters saturation, will not allow other transistors to enter saturation and will take whole of the current supplied from the driver gate. This is known as *current hogging*.

4.1.1.1.4 I^2L

As discussed above, the DCTL suffers from the difficulty of current hogging which makes it unsuitable. However, based on DCTL a new logic referred to as the integrated-injection logic

(I^2L), has been developed. I^2L has the simplicity of DCTL, uses very small silicon chip area, consumes very little power, and requires only four masks and two diffusions (compared to five masks and three diffusions for BJT) and hence, is easier and cheaper to fabricate. Due to these advantages it is eminently suited for medium- and large-scale integration. It is not used for small-scale integration and is the only saturated bipolar logic employed for large-scale integra- tion. Texas Instruments SBP 9900 is a 16-bit microprocessor using I^2L technology. The genesis of I2L technology is the concept of merging the components, viz. one semiconductor region is part of two or more devices. Because of this type of merging it is also referred to as the merged-transistor logic (MTL). There is considerable saving in the silicon chip area in thisprocess.

4.1.1.1.5 TransistorTransistorLogic(*TTL)*

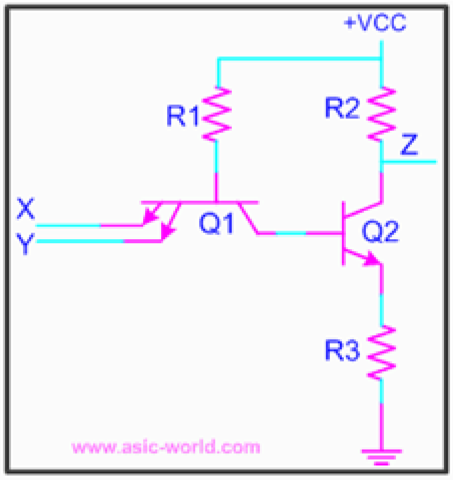
In Transistor Transistorlogic or just TTL, logic gates are built only around transistors.TTLwasdevelopedin1965.ThroughtheyearsbasicTTLhasbeen improvedtomeetperformancerequirements. Therearemanyversionsorfamiliesof TTL.

- StandardTTL.
- HighSpeed TTL
- LowPowerTTL
- SchhottkyTTL

TTLfamilieshavethree configurationsforoutputs.

- Totem-Poleoutput.
- OpenCollectorOutput.
- TristateOutput.

Theinputstage,whichisusedwithalmostallversionsofTTL,consistsofan inputtransistorandaphasesplitter transistor. Inputstageconsistsofamulti emittertransistorasshowninthefigurebelow.When anyinputisdrivenlow,the emitter basejunctionisforwardbiasedandinputtransistorconducts.Thisinturn drivesthephasesplittertransistorintocut-off.



Totem-PoleOutput

Below is the circuit of a totem-pole NANDgate, which has got three stages.

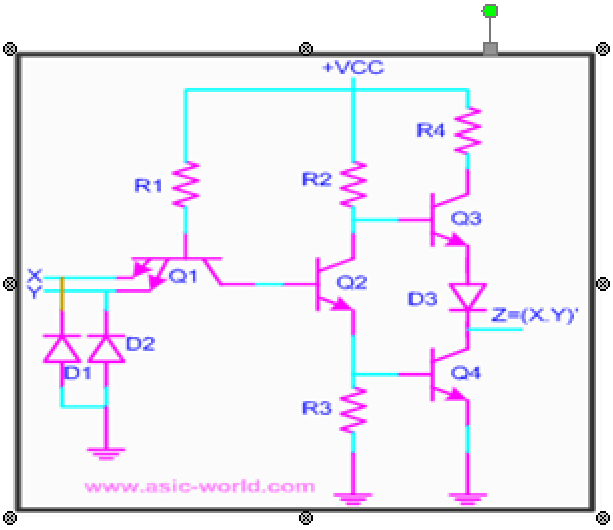
- InputStage
- PhaseSplitterStage
- OutputStage

InputstageandPhasesplitterstagehavealready beendiscussed.Outputstage is calledTotem-PolebecausetransistorQ3 sitsupon Q4.

Q2providescomplementaryvoltagesfortheoutputtransistorsQ3andQ4,which stackoneabovetheotherinsuchawaythatwhileoneoftheseconducts,the otherisin cut-off.

Q4iscalledpull-downtransistor,asitpullstheoutputvoltage down,whenit saturatesandtheother isincut-off(i.e.Q3isincut-off).Q3iscalledpull-up transistor,asitpullstheoutputvoltageup,whenitsaturatesandtheotherisin cut-off(i.e.Q4isin cut-off).

Diodes ininputareprotection diodes whichconductwhenthere islarge negative voltageatinput,shortingittotheground.

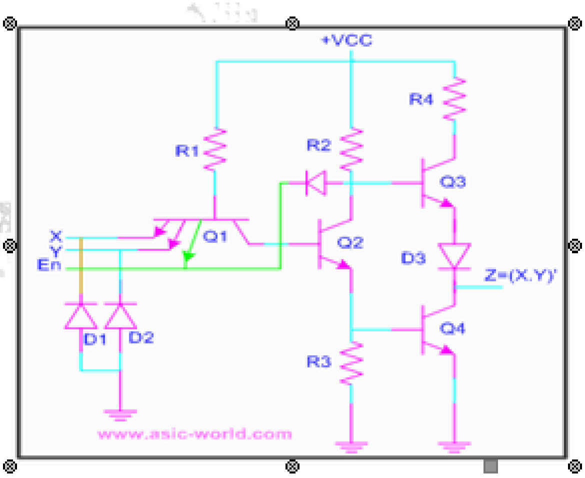


TristateOutput.

Normallywhenwehavetoimplement sharedbussystemsinsideanASICor externally tothechip,wehavetwooptions:eithertouseaMUX/DEMUXbased systemortouseatri-statebasebus system.

Inthelatter,whenlogicisnotdriving itsoutput, itdoesnotdriveLOWneither HIGH,whichmeans thatlogicoutput isfloating.Well, one mayask,whynotjust useanopencollector forsharedbussystems?The problemisthatopencollectors arenotsogood forimplementingwire-ANDs.

Thecircuitbelowisatri-stateNANDgate;whenEnableEnisHIGH,itworks likeanyotherNANDgate.ButwhenEnableEnisdrivenLOW,Q1Conducts, andthediodeconnecting Q1emitterandQ2collector,conductsdrivingQ3into cut-off.SinceQ2isnotconducting, Q4isalsoatcut-off.Whenbothpull-upand pull-downtransistorsarenotconducting,outputZisinhigh-impedancestate.



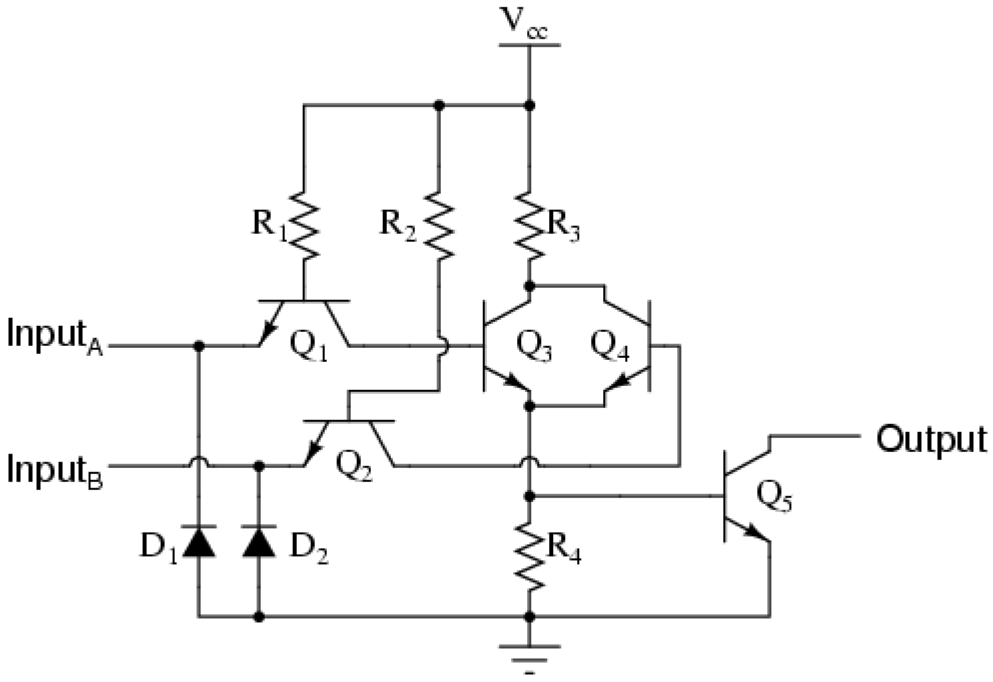
TTLNORANDORGATE

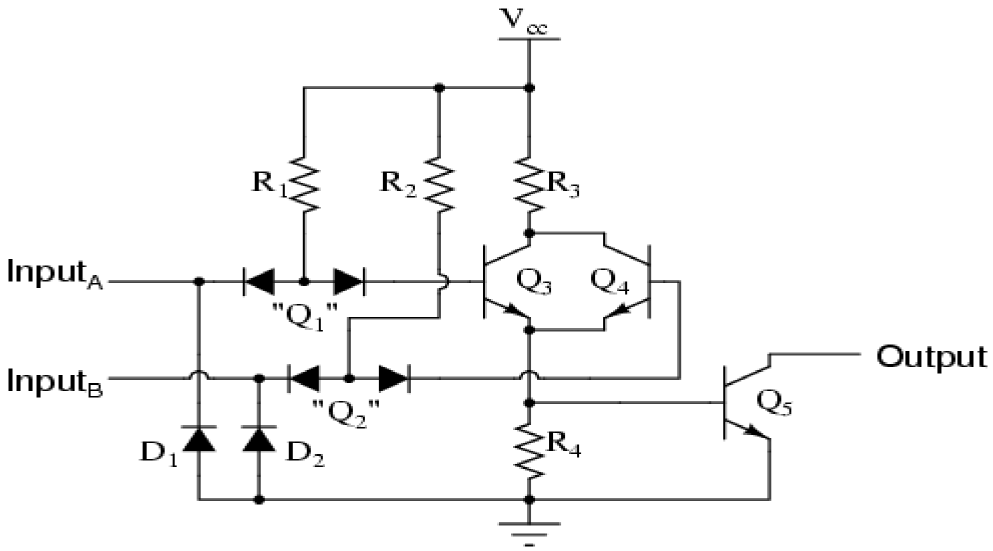
Let's examinethefollowingTTLcircuitandanalyzeitsoperation:

TransistorsQ$_1$ andQ$_2$ arebotharrangedinthesamemannerthatwe'veseenfor transistorQ$_1$inalltheotherTTLcircuits.Ratherthanfunctioningasamplifiers,Q$_1$and Q$_2$arebothbeingused astwo-diode"steering"networks.WemayreplaceQ$_1$and Q$_2$with diodesetstohelpillustrate:

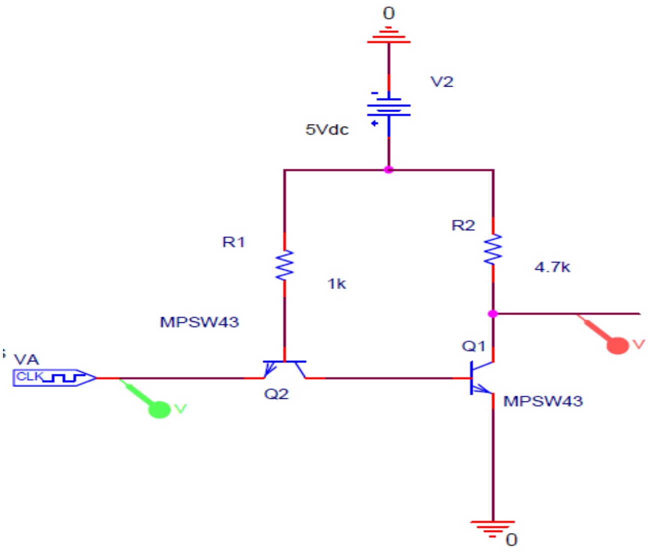
Ifinput Aisleftfloating(orconnected toV$_{cc}$),current willgothroughthebase of transistorQ$_3$,saturating it.IfinputAisgrounded, thatcurrentisdivertedawayfromQ$_3$'s basethroughtheleftsteeringdiodeof"Q$_1$,"thusforcingQ$_3$intocutoff.Thesame canbe saidforinputBandtransistorQ$_4$:thelogiclevelofinputBdetermines Q$_4$'sconduction: eithersaturatedor cutoff.

NoticehowtransistorsQ$_3$andQ$_4$are paralleled attheircollectorandemitter terminals.Inessence, thesetwotransistorsareacting asparalleled switches,allowing current throughresistorsR$_3$andR$_4$accordingtothe logiclevels ofinputsAandB.Ifany input isata"high"(1)level,then at least oneofthetwo transistors (Q$_3$and/or Q$_4$)willbe saturated,allowingcurrentthroughresistorsR$_3$ andR$_4$,andturningonthefinaloutput transistor Q$_5$fora"low" (0)logic level output.Theonlywaytheoutputofthis circuitcan everassumea"high"(1)stateisifbothQ$_3$ andQ$_4$ arecutoff,whichmeansbothinputs wouldhavetobegrounded,or"low" (0).

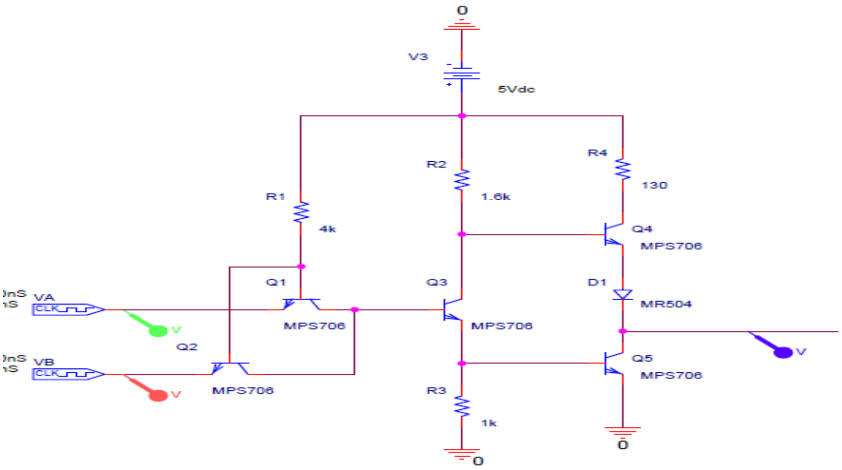

Vcc
R1
R2
R3
InputA
Q1
Q3
Q4
InputB
Q2
Output
Q5
D1
D2
R4


Vcc
R1
R2
R3
InputA
"Q1"
Q3
Q4
InputB
"Q2"
Output
Q5
D1
D2
R4

Noticehowtransistors$Q_3$and$Q_4$are paralleled attheircollectorandemitter



TTL implementation of the NOT function

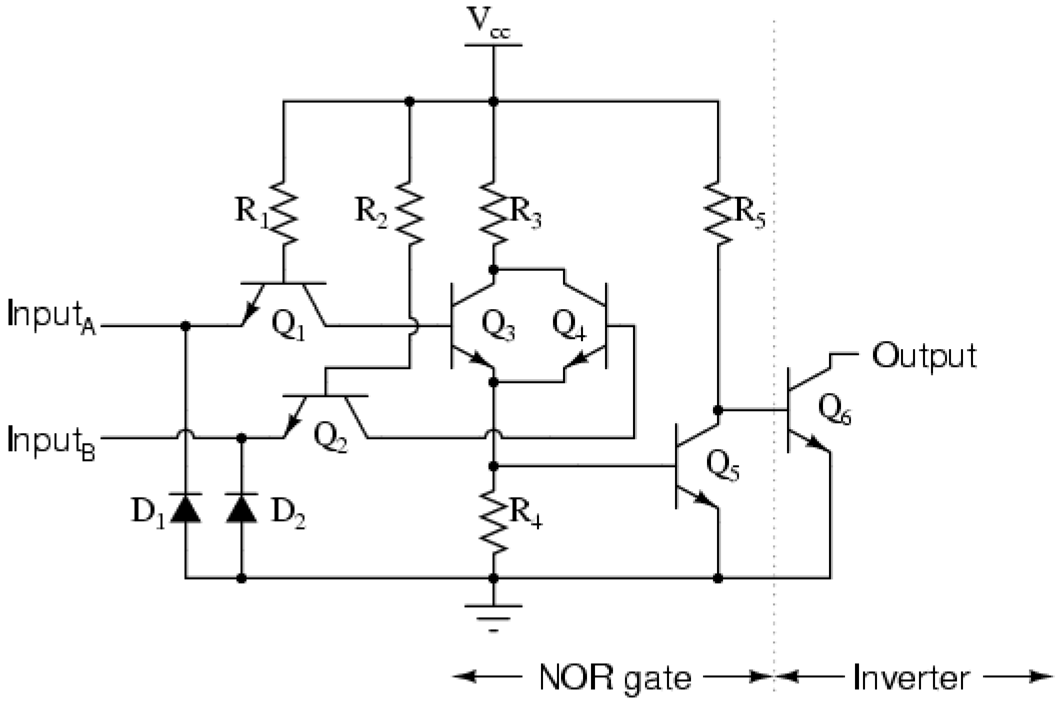


TTL implementation of the NAND function

Thiscircuit'struthtable,then,is equivalenttothatoftheNORgate:

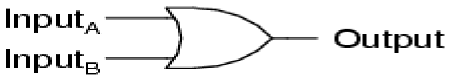
Inorder to turn thisNORgatecircuitinto anORgate,wewouldhaveto inverttheoutput logiclevelwithanother transistorstage, justlikewedidwiththe NAND-to-ANDgate example:

OR gate with open-collector output



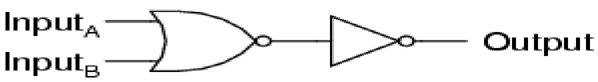
Thetruthtableandequivalentgatecircuit(aninverted-outputNORgate)areshown here:

OR gate



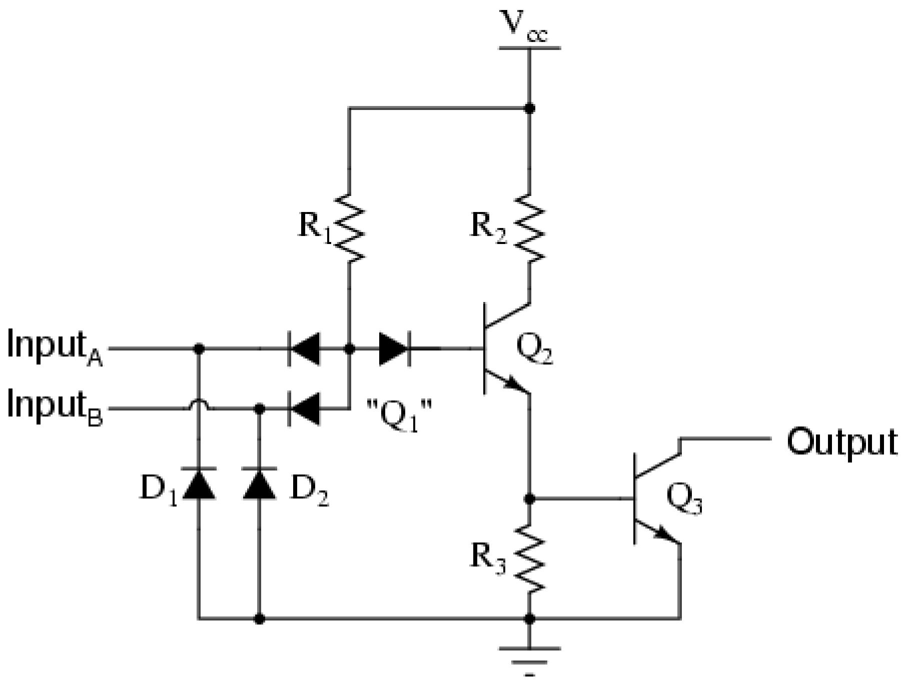
A	B	Output
0	0	0
0	1	1
1	0	1
1	1	1

Equivalent circuit



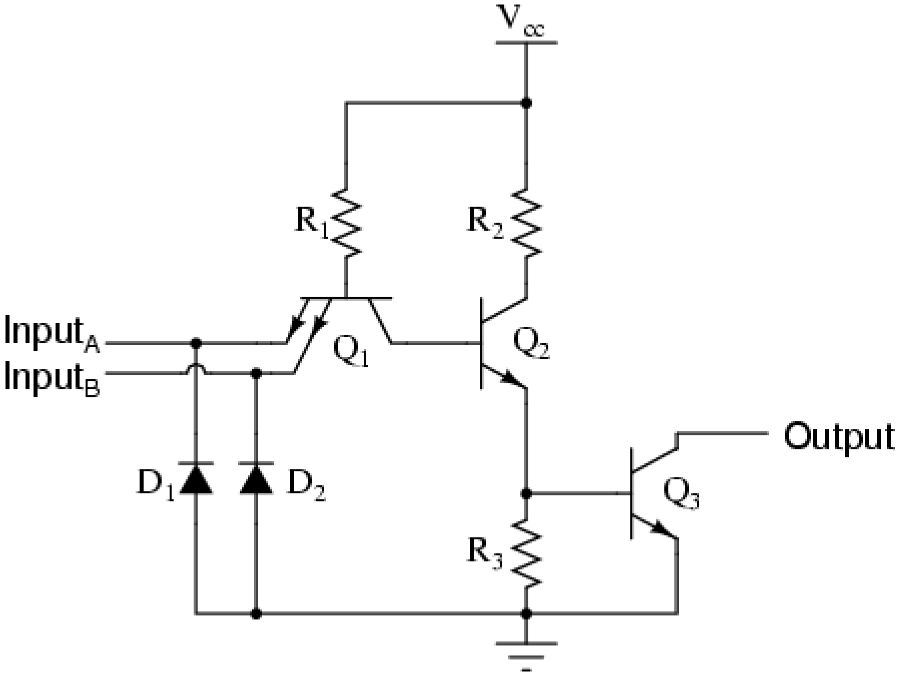
Ofcourse,totem-poleoutputstagesarealsopossibleinbothNORandORTTLlogic circuits.

A two-input inverter circuit



Thisschematic illustratesarealcircuit,butitisn'tcalleda"two-inputinverter." Through analysis we will discover what this circuit's logic function is and correspondinglywhatitshouldbedesignatedas.

Justasinthecaseoftheinverterandbuffer,the"steering" diodeclustermarked "Q_1"isactually formedlikeatransistor,eventhoughitisn'tusedinanyamplifying capacity.Unfortunately,asimpleNPNtransistorstructureisinadequate tosimulatethe threePNjunctionsnecessaryinthisdiodenetwork,soadifferenttransistor(andsymbol) isneeded.Thistransistorhasonecollector,onebase,andtwoemitters,andinthecircuit itlookslikethis:

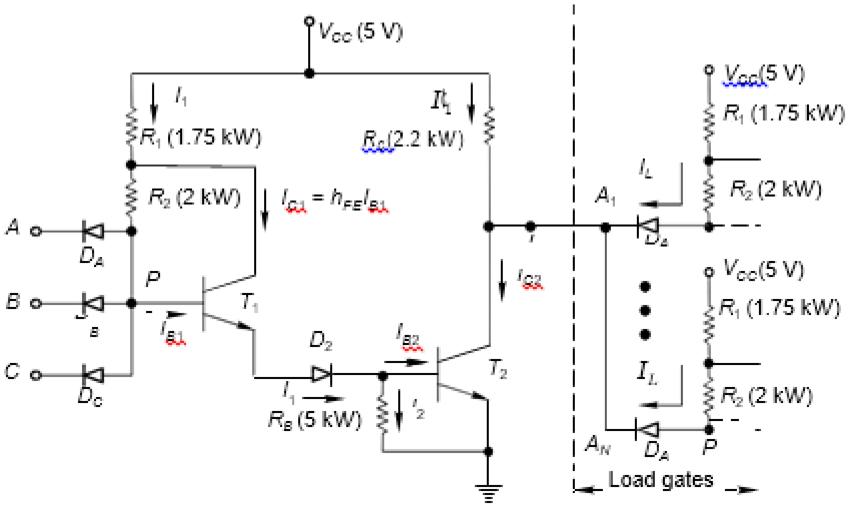


Inthesingle-input (inverter)circuit,groundingtheinputresulted inanoutput that assumedthe"high"(1)state.Inthecaseoftheopen-collector outputconfiguration,this "high"statewassimply"floating."Allowingtheinputtofloat(orbeconnected toV_{cc}) resultedintheoutputbecominggrounded,whichisthe"low"or0state.Thus,a1in resultedina0 out,andviceversa.

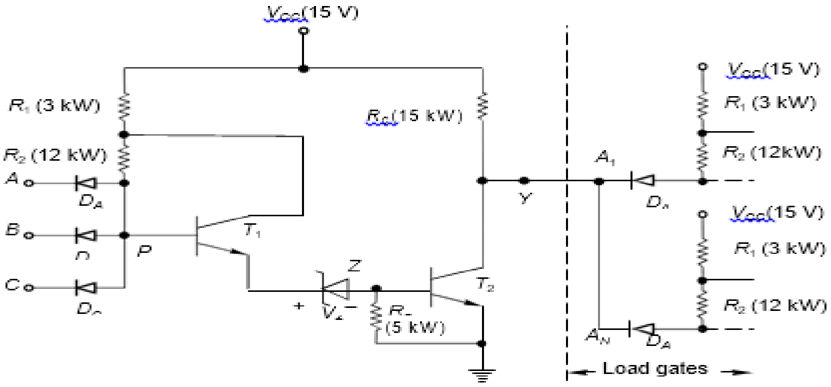
Since this circuitbearsso much resemblance to thesimple inverter circuit, the only difference beinga second input terminal connected in thesame way to thebaseof transistor Q2, we cansay that eachof the inputs willhave thesame effecton theoutput. Namely, if eitherof the inputs are grounded, transistor Q2 willbe forced intoa condition of cutoff, thus turning Q3 off and floating theoutput (output goes "high").The following seriesof illustrationsshows this for three input

4.1.1.1.6 HTL

Due to the presence of electric motors, on–off control circuits, high voltage switches, etc. in an industrial environment, the noise level is quite high and the logic families discussed so far do



not perform the intended functions. For this purpose, the DTL gate of Fig. has been redesigned with a higher supply voltage (15 V instead of 5 V). The diode D_2 has been replaced by a Zener diode with a Zener breakdown voltage of 6.9 V and the resistances have been modified so that approximately the same currents are obtained as in DTL. A 3-input HTLNANDgatewithafan-outof*N*isshowninFig.Thecircuitcanbeanalysedtodetermine the noise-margins, fan-out and power dissipation (Prob. 4.10).



4.1.1.2 *Non saturated family*

The non-saturated bipolar logic families are:

1. SchottkyTTL, and
2. Emitter-coupled logic (ECL).

4.1.1.2.1 *SchottkyTTL*

The speed limitation of TTL is mainly due to the turn-off time delays involved in transistors while making transitions from saturation to cut-off. This can be eliminated by replacing the transistors of TTL gate by Schottky transistors.

With this, the transistors are prevented from entering saturation and hence, there is saving in turn-off time. SchottkyTTL gates have propagation delay time of the order of 2 ns which is very small in comparison with the propagation delay time of standard TTL which is of the order of 10 ns. It is a nonsaturating bipolar logic.

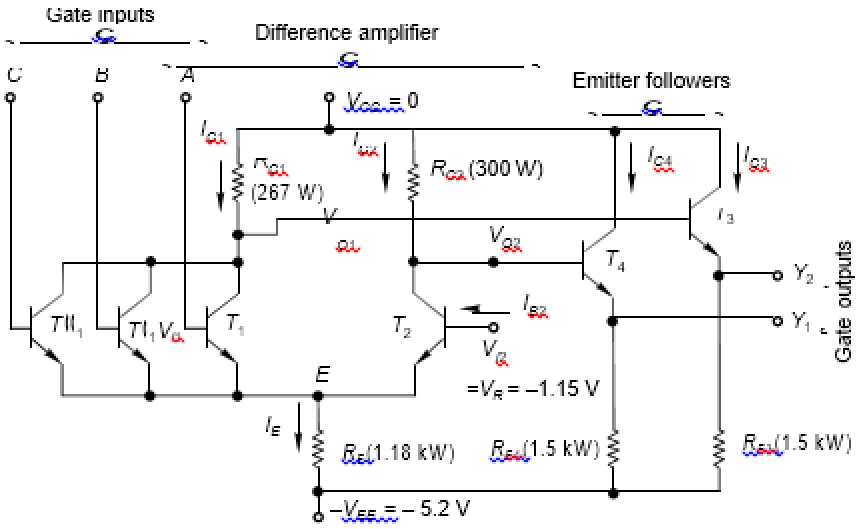
4.1.1.2.2 *ECL*

Emitter-coupled logic (ECL) is the fastest of all logic families and therefore is used in applica- tions where very high speed is essential. High speeds have become possible in ECL because the transistors are used in difference amplifier configuration, in which they are never driven into saturationandtherebythestoragetimeiseliminated.Here,ratherthanswitchingthetransistors from ON to OFF and vice-versa, they are switched between cut-off and active regions. Propagationdelaysoflessthan1nspergatehavebecomepossibleinECL.

Basically, ECL is realized using difference amplifier in which the emitters of the two transis- tors are connected and hence it is referred to as emitter-coupled logic. A 3-input ECL gate is shown in Fig. 4.19, which has three parts: The middle part is the difference amplifier which performs the logic operation.

Emitter follower are used for d.c. level shifting of the outputs, so that $V(0)$ and $V(1)$ are same for the inputs and the outputs. Note that two output Y_1 andY_2 are available in this circuit which are complementary. Y_1 corresponds to OR logic and Y_2 to NOR logic and hence it is named as an OR/NORgate.

Additional transistors are used in parallel to T_1 to get the required fan-in. There is a funda- mental difference between all other logic families (including MOS logic) and ECL as far as the



A 3-input ECL OR/NOR gate.

supply voltage is concerned. In ECL, the positive end of the supply is connected to ground in contrast to other logic families in which negative end of the supply is grounded. This is done to minimize the effect of noise induced in the power supply (Prob. 4.22), and protection of the gate from an accidental short circuit developing between the output of a gate and ground (Prob. 4.23). The voltage corresponding to $V(0)$ and $V(1)$ are both negative due to positive end of the supply being connected to ground. The symbol of an ECL OR/NOR gate is shown in Fig. 4.20.

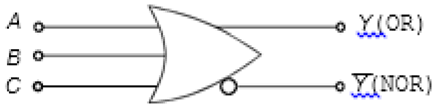


Fig. 4.20

The symbol for a 3-input OR/NOR gate.

4.1.2 Unipolar family

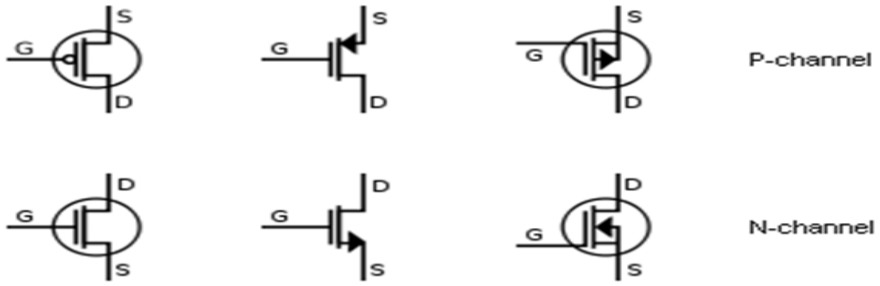
MOS devices are unipolar devices and only MOSFETs are employed in MOS logic circuits. The MOS logic families are:

1. PMOS,
2. NMOS,and
3. CMOS

While in PMOS only *p*-channel MOSFETs are used and in NMOS only *n*-channel MOSFETs are used, in complementary MOS (CMOS), both *p*- and *n*-channel MOSFETs are employed and are fabricated on the same silicon chip.

4.1.2.1 MOS

MOS types, symbols



NMOS Inverter:

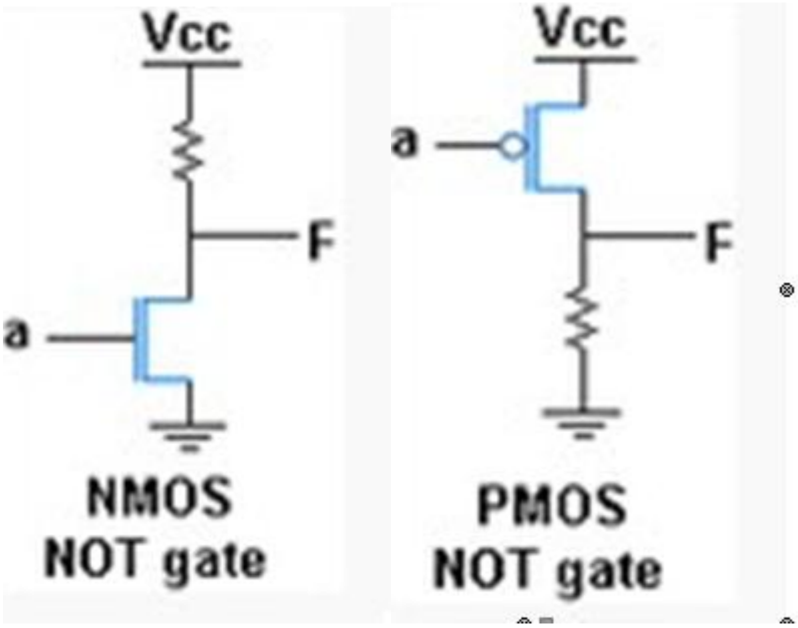
– when a=‘1’, nMOS conducts, so F=‘0’

– When a=‘0’, nMOS is cut- off, so F=Vcc=logic ‘1’

- PMOS Inverter

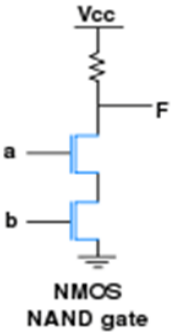
– when a=‘1’, pMOS is cut- off, so F=‘0’

– When a=‘0’, pMOS is on, so F=Vcc=logic ‘1’



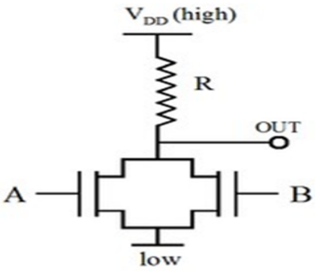
nMOS NAND

- When any input is ‘0’
  - corresponding of mos is off
  - So F=Vcc=‘1’
- When both inputs are ‘1’
  - Both mos are on
  - F = Gnd = ‘0’



nMOS -NOR

- When any input is ‘1’
  - Corresponding mos is on
  - So F=Gnd=‘0’
- When both inputs are ‘0’
  - Both mos are off
  - Out = V_{DD} = ‘1’

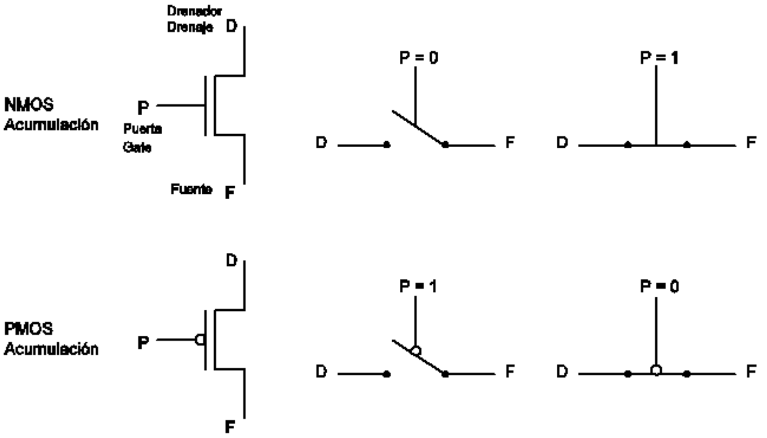


4.1.2.2 CMOS

Basic CMOS concepts

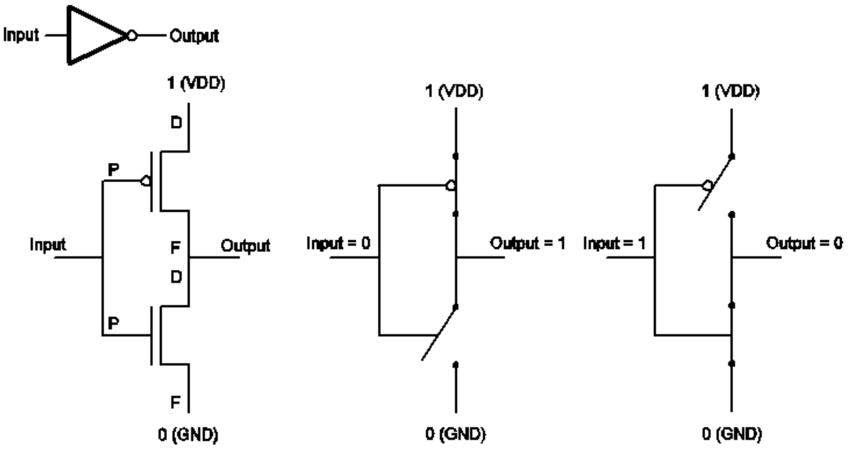
We will now see the use of transistor for designing logic gates. Further down in the course we will use the same transistors to design other blocks (such as flip-flops or memories)

Ideally, a transistor behaves like a switch. For NMOS transistors, if the input is a 1 the switch is on, otherwise it is off. On the other hand, for the PMOS, if the input is 0 the transistor is on, otherwise the transistor is off. Here is a graphical representation of these facts:



When a circuit contains both NMOS and PMOS transistors we say it is implemented in CMOS (Complementary MOS)

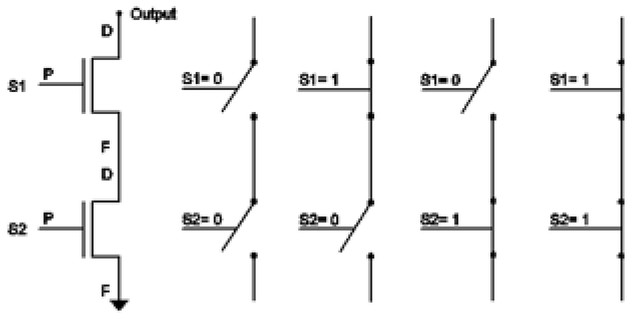
Understanding the basics of transistors, we can now design a simple NOR gate. Next figure shows the implementation in transistors of the NOR gate and how it works for different inputs (1 and 0). On the left there is the implementation, on the right the behavior. The symbol VDD is the source voltage (or the logic 1), GND is the ground (or the logical 0).



We have just seen how to implement a simple logic gate using transistors. To implement the rest of logical gates (and whatever circuit we might think off), we will analyze first the behavior of the transistors when connected in a "series" fashion or in a "parallel" way.

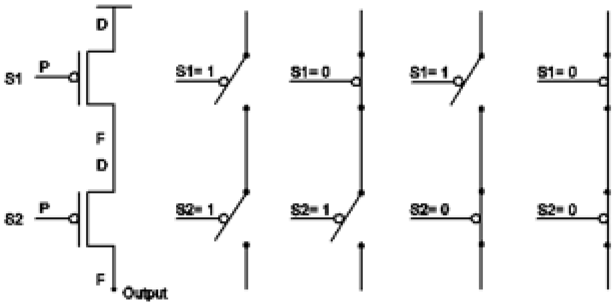
If we connect two NMOS transistors in series, we get the behaviour shown in next figure (the triangle in the bottom is a graphical representation of GND)

– Output = 0 si S1=1 i S2=1



Next figure shows the behavior of the PMOSes when connected in series. (The horizontal line on top of the first transistor is a graphical representation of VDD).

– Output = 1 si S1=0 i S2=0



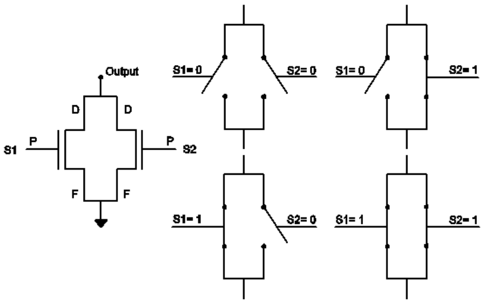
In the next figure, we can see the behavior of the NMOSes and PMOSes when connected in parallel.

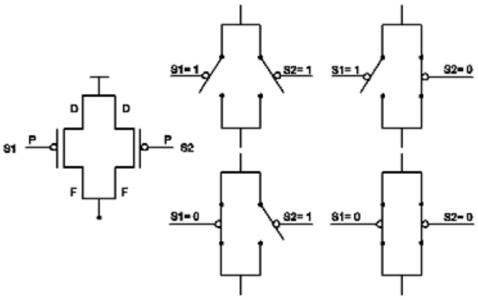
Combinació paral·lela nMOS:

– Output = 0 si S1=1 o S2=1

Combinació paral·lela pMOS:

– Output = 1 si S1=0 o S2=0



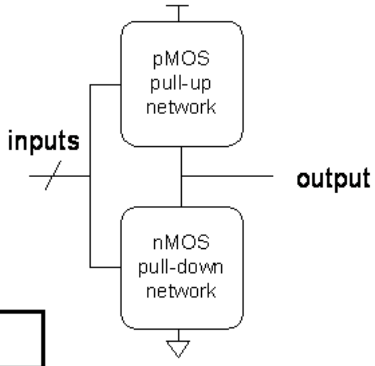


Summing up, NMOS transistors in series let the current flow when both inputs are 1; otherwise the output is undefined (Z). If we connect the NMOSes in parallel, then the current flows when any (or both) of the inputs are 1; otherwise the output is undefined (Z).

For the PMOSes, when connected in series the current flows when both inputs are 0; otherwise the output is undefined. Alternatively, when connected in parallel, if any (or both) of the inputs is 0 the current flows. Otherwise the output is undefined.

When using CMOS technology (and specifically static CMOS), we will design the circuits with two clearly defined parts. One (called *pull-up*) will be built of PMOS transistors and it has the duty of setting the output to 1 whenever the implemented function defines it. The other part (called *pull- down*) will be built of NMOS transistors and it will set the output to 0 whenever the implemented function defines it. All circuits will either set the output to 1 or 0 for any combination of the input values. Both *pull-up* and *pull-down* cannot be active at the same time (it makes no sense to set the output to 1 and 0 for the same inputs!!). Similarly, both the *pull-up* and the *pull-down* cannot be off at the same time (logic functions have always a defined output – either 0 or 1). Nevertheless, we will see further down the course that when not implementing logic functions we might be interested – sometimes- in setting the output to undetermined in certain cases.

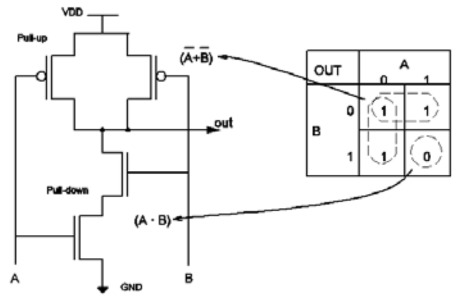
- Complementary CMOS logic gates
  - – nMOS *pull-down network*
  - – pMOS *pull-up network*
  - – a.k.a. static CMOS



	Pull-up OFF	Pull-up ON
Pull-down OFF	Z (float)	1
Pull-down ON	0	X (crowbar)

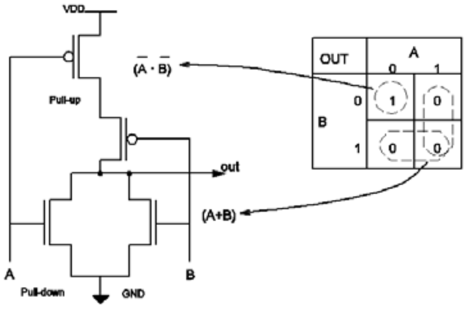
Next figures show the implementations of the NAND and NOR gates in CMOS. For each one of them, there is the truth table and clear indications of what outputs are set by the pull-up and what outputs for the pull-down.

Porta NAND



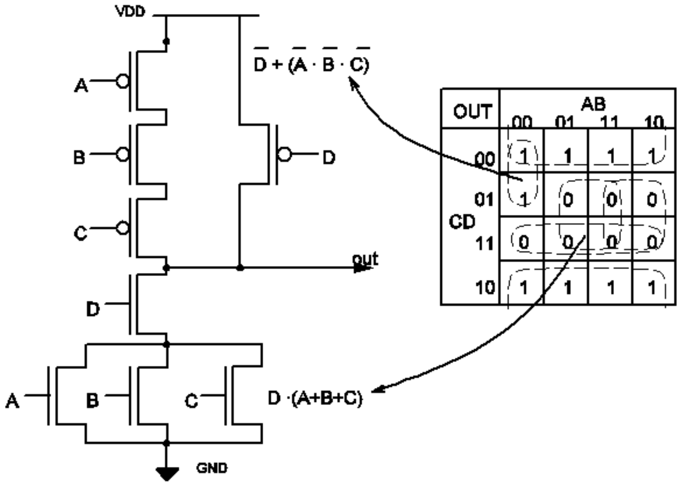
OUT		A	
		0	1
B	0	1	1
	1	1	0

Porta NOR



OUT		A	
		0	1
B	0	1	0
	1	0	0

In the same way we implement the logic gates, we can implement any logic function. Next figure shows the implementation of the logic function $f(A, B, C, D) = \overline{D\cdot(A + B + C)}$.


VDD
A
B
C
$\overline{D} + (\overline{A} \cdot \overline{B} \cdot \overline{C})$
D
out
D
A
B
C
D ·(A+B+C)
GND

OUT	AB 00	01	11	10
CD 00	1	1	1	1
01	1	0	0	0
11	0	0	0	0
10	1	1	1	1

We can actually define a design methodology for turning logical functions into CMOS circuits:

- The logic function must be complemented. (i.e., it must look likef(x,y,z)= NOT(*expression*); in a logic expression, $f(x, y, z) = \overline{expression}$
- AND operator("·"):
  - Pull-down: NMOS transistors NMOS inseries
  - Pull-up: PMOS transistors inparallel
- OR operator("+"):
  - Pull-down: NOMS transistors inparallel
  - Pull-up: PMOS transistors inseries

Problem sets to hand in:

Implement the following functions in CMOS logic.

- $Z = \overline{A \cdot B \cdot C \cdot D}$
- $Z = \overline{A + B + C + D}$
- $Z = \overline{((A \cdot B \cdot C) + D)}$
- $Z = \overline{(((A \cdot B) + C) \cdot D)}$
- $Z = \overline{(A \cdot B) + (C \cdot (A + B))}$

Implement the following functions in CMOS logic. Assume you can use a NOT gate whenever necessary.

- $Z = A$ (buffer)
- $Z = A \cdot \overline{B} + \overline{A} \cdot B$ (XOR)
- $Z = A \cdot B + \overline{A} \cdot \overline{B}$ (XNOR)
- $Z = A \cdot \overline{B} \cdot \overline{C} + \overline{A} \cdot \overline{B} \cdot C + \overline{A} \cdot B \cdot \overline{C} + A \cdot B \cdot C$ (funció suma en un sumador binari)

4.2 Tristate logic

Normally, in logic circuits, there are two states, HIGH and LOW.

The output can be either HIGH or LOW.

But in complex circuits, group of logic outputs are connected to a line called *bus*. Thus *bus* should also drive many gate inputs. Thus *fan-in* and *fan-out* of the *bus* should be properly defined to drive many gates.

Some difficulties are encountered due to the following:

1 totem-pole outputs cannot be together because of a very large current drain from the supply and consequent heating of ICs which can get damaged.

2. Open-collector outputs can be connected together with a common-collector resistor connected externally. This causes the problems of loading and speed of operation.

To overcome these difficulties, special circuits have been developed in which there is one more state of the output, referred to as the *third state* or *high-impedance state*, in addition to the LOW and HIGH states. These circuits are known as *TRI-STATE, tri-state logic* (TSL) or *three-state logic*. (TRI-STATE)

There is a basic functional difference between wired-OR and the TSL. For the wired-OR connection of two functions $Y1$ and $Y2$ is

$$Y = Y1 + Y2$$

whereas for TSL, the result is not a Boolean function but an ability to multiplex many functions economically.

TSL Inverter

A TSL inverter circuit with tri-state output is shown in Fig. 4.2.1. When the control input is LOW, the drive is removed from $T3$ and $T4$. Hence, both $T3$ and $T4$ are cut-off and the output is in the third state. When the control input is HIGH, the output Y is logic 1 or 0 depending on the data input. The logic symbol of a TSL inverter is shown in Fig. 4.2.2 and its truth table is given in Table 4.2.1. (where T1, T2, T3, T4 and T5 are Transistor shown in fig.4.2.2)

Table. 4.2.1 Truth Table of a TSL Inverter

DATA INPUT	CONTROL	DATA OUTPUT
0	0	HIGH-Z
1	0	HIGH-Z
0	1	1
1	1	0

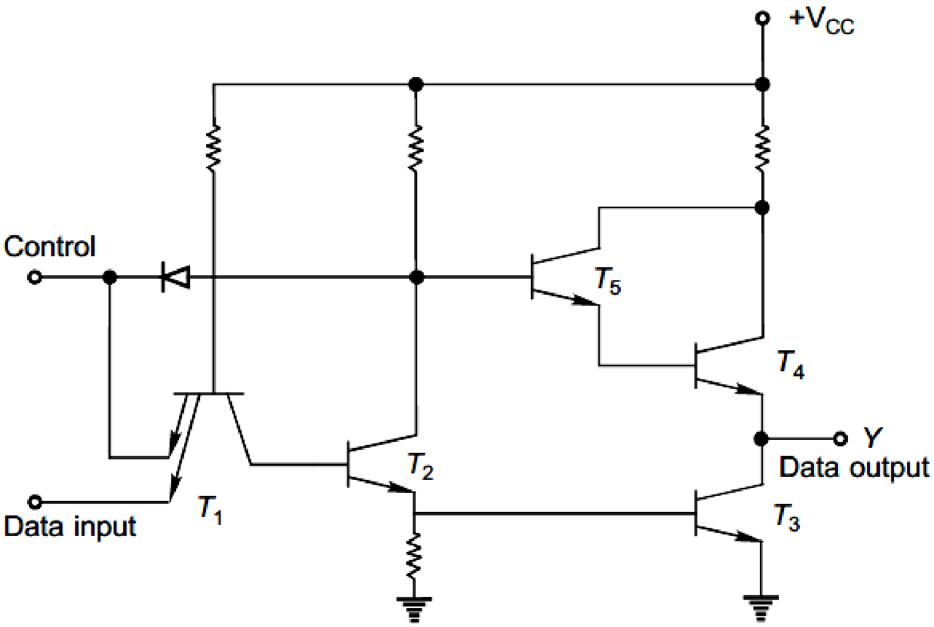


Fig. 4.2.1 TSL Inverter

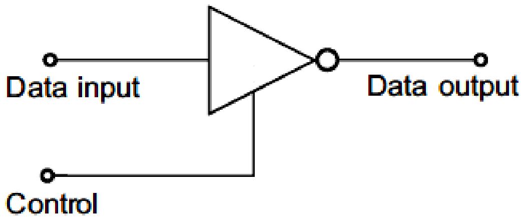


Fig. 4.2.2 Logic symbol of a TSL Inverter

4.3 Interfacing of CMOS and TTL families

Firstly we shall learn to interface LEDs (Light Emitting Diode) with TTL and CMOS circuits because LED being output indicators. For low currents and voltages LEDs are perfect since they need only 20 to 30 mA with about applied 2 V.

Here CMOS 400 series and TTL 74 series are considered.

CMOS to LED interfacing:

Six examples of CMOS-IC driving LED indicators are shown in the below figure 4.3.1 (a) to (f).

Figure 4.3.1 (a) and (b) shows *CMOS* supply voltage at +5 V. At this low voltage, no limiting resistors are needed in series with the LEDs.

In figure 4.3.1 (a), when output of CMOS inverter goes HIGH, the LED output indicator glows (LED connected in such a way that the *Anode* is connected to the Output of CMOS gate). In figure 4.3.1 (b), when output of CMOS inverter goes LOW, the LED output indicator glows (LED connected in such a way that the *Cathode* is connected to the Output of CMOS gate).

Figure 4.3.1 (c) and (d) shows *CMOS* operated in high supply voltage at +10 to +15 V. At this high voltage, *limiting resistors* (here 1 KΩ) is needed in series with the LEDs.

In figure 4.3.1 (c), when output of CMOS inverter goes HIGH, the LED output indicator glows (LED connected in such a way that the *Anode* is connected to the Output of CMOS gate). In figure 4.3.1 (d), when output of CMOS inverter goes LOW, the LED output indicator glows (LED connected in such a way that the *Cathode* is connected to the Output of CMOS gate).

Figure 4.3.1 (e) and (f) shows *CMOS Buffer* operated in high supply voltage at +5 to +15 V. At this high voltage, *limiting resistors* (here 1 KΩ) is needed in series with the LEDs.

The only difference is CMOS Buffer is used instead of CMOS gate.

TTL to LED interfacing:

Figure 4.3.1 (g) and (h) shows *a standard TTL gate* operated in high supply voltage at +5 to +15 V. At this high voltage, *limiting resistors* (here 1 KΩ) is needed in series with the LEDs.

In figure 4.3.1 (g), when output of Standard TTL gate goes HIGH, current will flow through the LED output indicator to make a glow (LED connected in such a way that the *Anode* is connected to the Output of TTL gate).

In figure 4.3.1 (h), when output of Standard TTL gate goes LOW, current will flow through the LED output indicator to make a glow (LED connected in such a way that the *Cathode* is connected to the Output of TTL gate).

The circuits in figure 4.3.1 are not proper for real-time critical use because they can exceed current ratings.

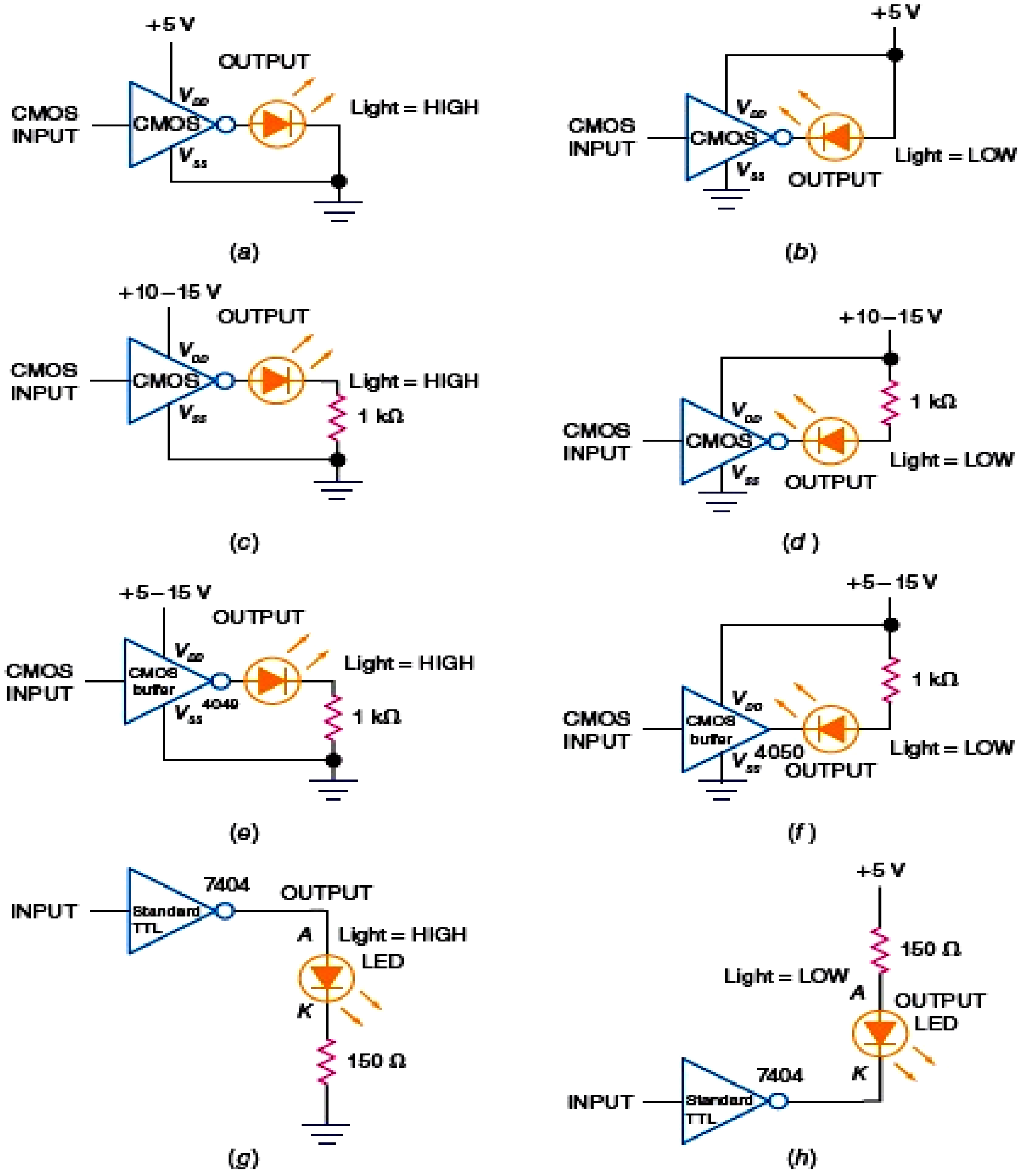


Fig. 4.3.1 A Simple LED interfacing with TTL and with CMOS.
(a.) CMOS Active HIGH (b.) CMOS Active LOW (c.) CMOS active HIGH, Supply Voltage = 10 to 15 V (d.) active LOW, Supply Voltage = 10 to 15 V (e.) CMOS inverting buffer to LED interfacing (f .) CMOS non-inverting buffer to LED interfacing (g.) TTL Active HIGH (h.) TTL Active LOW

Current Sourcing and sinking:

In figure 4.3.2 (a.) the output of the TTL AND gate is HIGH. This HIGH at the output of the AND gate makes the LED glow. The source current appears to "flow from the IC" through the external circuit (LED and limiting resistor) to ground.

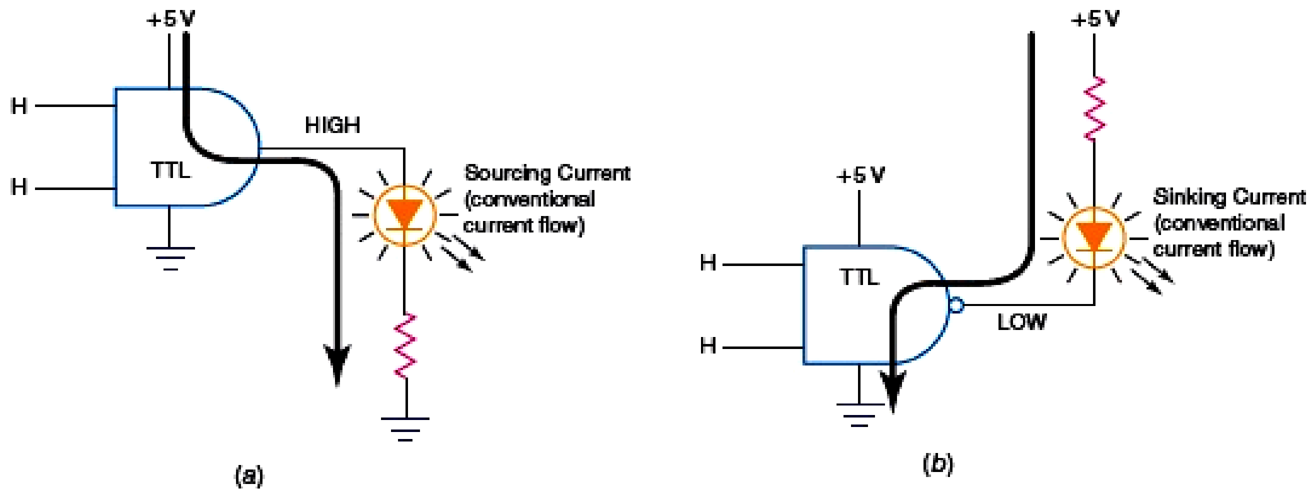

+5V
H
H
TTL
HIGH
Sourcing Current (conventional current flow)
(a)
+5V
+5V
H
H
TTL
LOW
Sinking Current (conventional current flow)
(b)

Fig. 4.3.2 (a.) TTL Current Sourcing (TTL-AND gate) (b.) TTL Current Sinking (TTL-NAND gate)

In figure 4.3.2 (b.) the output of the TTL NAND gate is LOW. This LOW at the output of the NAND gate makes the LED glow. The sink current appears to start with +5V above the external circuit (LED and limiting resistor) and "sink to ground" through the external circuit (LED and limiting resistor) and the output pin of the NAND IC.

TTL to CMOS interfacing:

CMOS and TTL logic voltage levels are differently defined. These differences are shown in figure 4.3.3 (a.). Due to these different voltage levels CMOS and TTL ICs cannot be connected together by using usual connectivity methods.

The Current and voltage levels are shown in the *Table 4.3.2 below.*

Table 4.3.2 – Current and Voltage levels of TTL and CMOS logics

	Current Levels		Voltage Levels	
Levels ➡	**LOW**	**HIGH**	**LOW**	**HIGH**
Standard TTL	16 mA	400 µA	0 V - 0.8 V	2.4 V – 5 V
CMOS	1 µA	1 µA	0 V - 1.5 V	3.5 V – 5 V

As shown in the above table, Output drive currents of standard TTL are sufficient to drive CMOS input bands because the latter is 1 μA are less than TTL current levels.

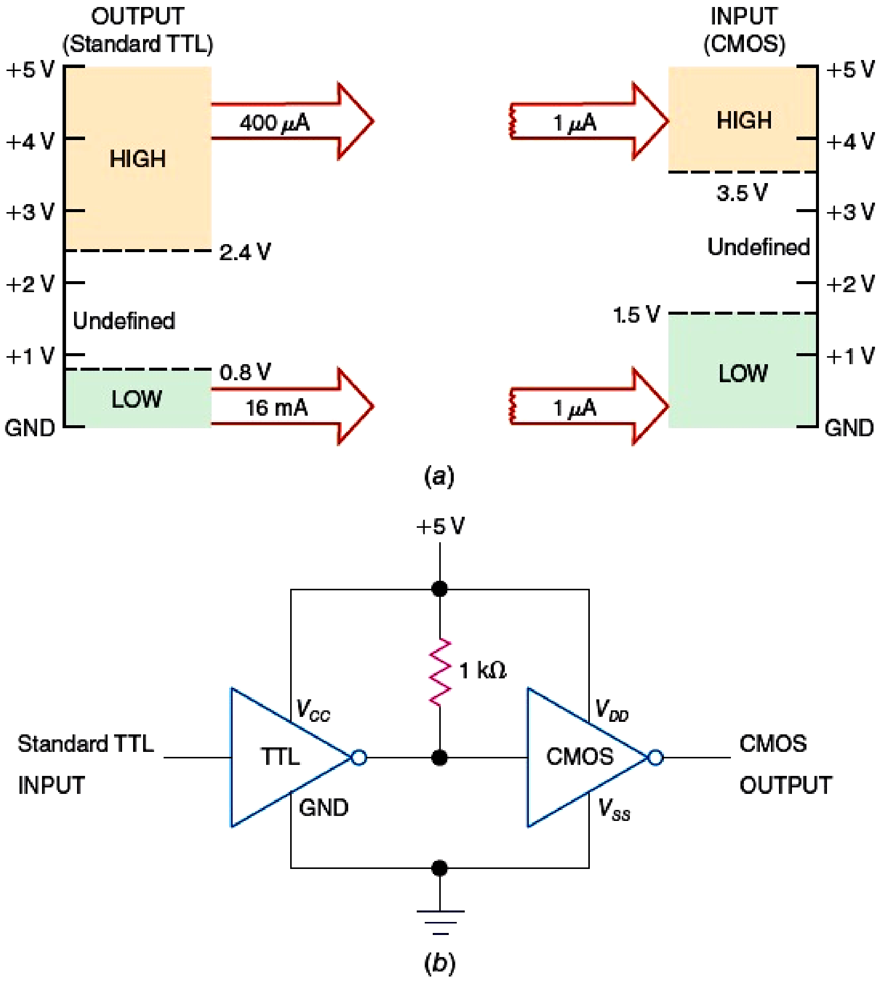


Fig. 4.3.3. TTL to CMOS interfacing (a.) TTL output and CMOS input (b.) TTL-CMOS interfacing using pull-up resistor

The HIGH voltage levels of TTL (2.4 V to 3.5 V) which are not compatible with HIGH voltage levels of CMOS (3.5 V to 5 V) and also LOW voltage levels of TTL (0 V to 0.8 V) which are not compatible with HIGH voltage levels of CMOS (0 V to 1.5 V) cause problems due to incompatibility. This problems are solved by using a pull-up resistor between gates to pull the HIGH output of standard TTL up closer to +5V as shown in the figure 4.3.3 (b.). Here a 1 KΩ is used a pull-up resistor.

Examples of converting a TTL-to-CMOS and CMOS-to-TTL interfacing sing a common 5 V power supply are shown in figure 4.3.4.

Figure 4.3.4 (a*)* shows the popular LS-TTL driving any CMOS gate. Notice the use of a 2.2-kΩ pull-up resistor. The pull-up resistor is being used to pull the TTL HIGH up near 15 V so that it will be compatible with the input voltage characteristics of CMOS ICs. In Fig. 4.3.4 (*b*), a CMOS inverter (any series) is driving an LS-TTL inverter *directly.* Complementary symmetry metal-oxide semiconductor ICs can drive LS-TTL and ALS-TTL (advanced low-power Schottky) inputs: most CMOS ICs cannot drive standard TTL inputs without special interfacing.

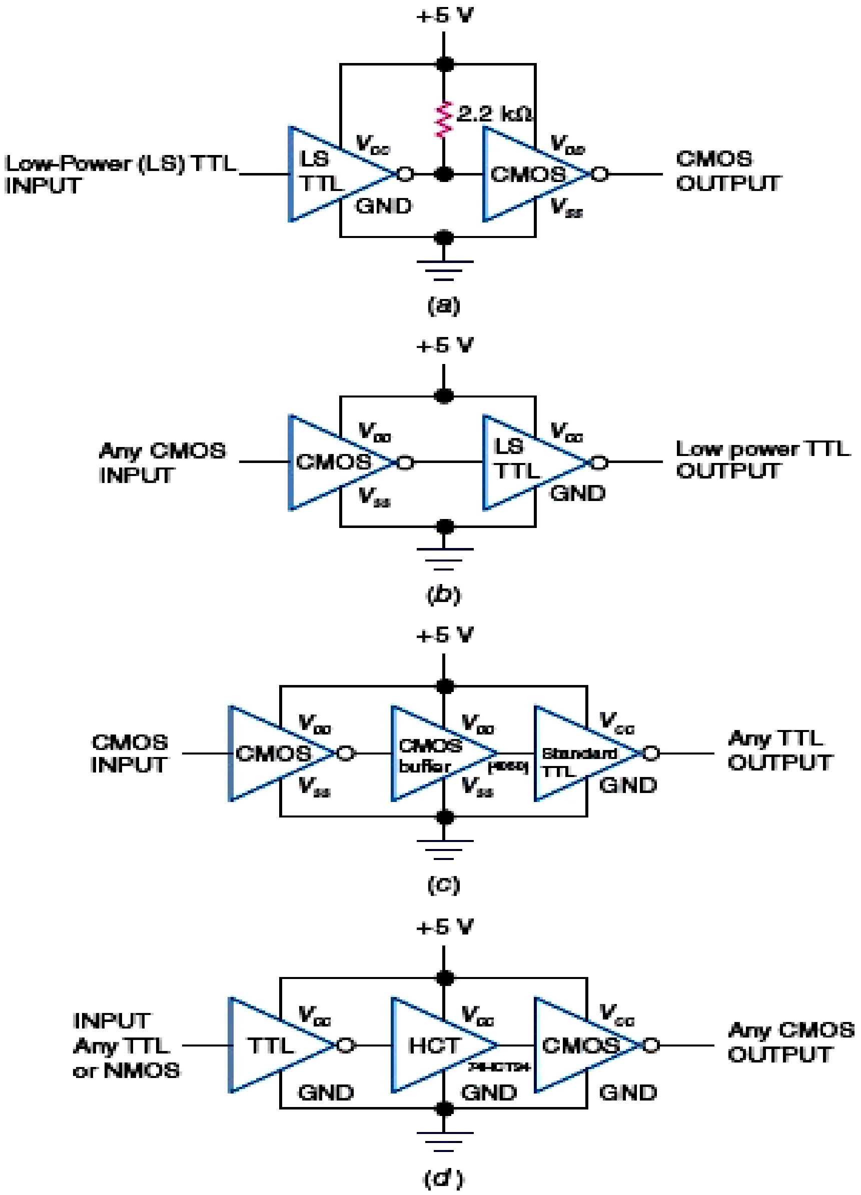


Fig. 4.3.4. Interfacing TTL and CMOS when both use common +5 V Power Supply (a.) Low-power Schottky TTL to CMOS interfacing using a pull-up resistor (b.) CMOS to low-power Schottky TTL interfacing (c.) CMOS to Standard-TTL interfacing using a CMOS buffer IC (d.) TTL to CMOS interfacing using 74HCT00 Series IC

Manufacturers have made interfacing easier by designing special buffers and other interface chips for designers. One example is the use of the 4050 non-inverting buffer in Fig. 4.3.4 (c*).* The 4050 buffer allows the CMOS inverter to have enough drive current to operate up to two standard TTL inputs. The problem of voltage incompatibility from TTL (or NMOS) to CMOS was solved in Fig. 4.3.3 using a pull-up resistor.

Another method of solving this problem is illustrated in Fig. 4.3.4 (d). The 74HCT00 series of CMOS ICs is specifically designed as a convenient interface between TTL (or NMOS) and CMOS. Such an interface is implemented in Fig. 4.3.4 (d) using the 74HCT34 non-inverting IC. The *74HCT00 series of CMOS ICs* is widely used when interfacing between NMOS devices and CMOS.

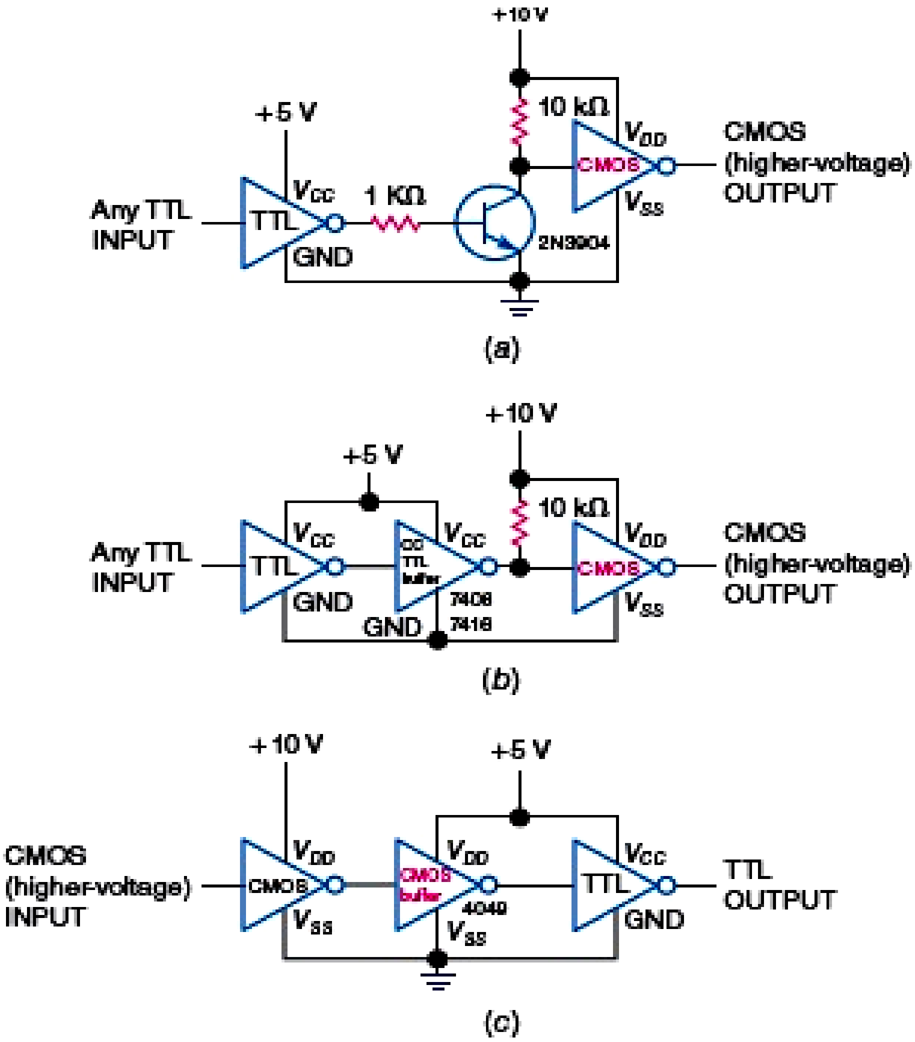


Fig. 4.3.5. Interfacing TTL and CMOS when each uses a different Power Supply voltage (a.) TTL-to-CMOS interfacing using a driver transistor (b.) TTL-to-CMOS interfacing using a an open-collector TTL buffer IC (c.) CMOS- to-TTL interfacing using a CMOS buffer IC

Interfacing CMOS devices with TTL devices takes some added components when each operates on a *different voltage power supply.* Figure 4.3.5 shows three examples of TTL-to-CMOS and CMOS-to-TTL interfacing. Figure 4.3.5 (*a*) shows the TTL inverter driving a general-purpose NPN transistor. The transistor and associated resistors translate the lower voltage TTL outputs to the higher-voltage inputs needed to operate the CMOS inverter. The CMOS output has a voltage swing from about 0 to almost 110 V. Figure 4.3.5 (*b*) shows an open-collector TTL buffer and a 10-kΩ pull-up resistor being used to translate the lower TTL to the higher CMOS voltages. The 7406 and 7416 TTL ICs are two inverting, open-collector (OC) buffers. Interfacing between a higher-voltage CMOS inverter and a lower-voltage TTL inverter is shown in Fig. 4.3.5 (c). The 4049 *CMOS buffer* is used between the higher-voltage CMOS inverter and the lower-voltage TTL IC. Note that the CMOS buffer is powered by the lower voltage (15 V) power supply in Fig. 4.3.5 (c).

4.4 Comparison of logic families

The table 4.4.1 shows the comparison of various logic families with some characteristics like Fan-out, power dissipation, Noise immunity, Propagation delay, speed-power product, Clock-Rate etc.

Table. 4.4.1 TSL Inverter

Logic family➡	**RTL**	**I^{2}L**	**DTL**	**HTL**	**TTL**	**SCHOTTKY TTL**	**ECL**	**MOS**	**CMOS**
Parameter⬇									
FAN OUT	LOW	LOW	LOW	LOW	MEDIUM	MEDIUM	MEDIUM	MEDIUM	HIGH
POWER DISSIPATION	MEDIUM	LOW	MEDIUM	HIGH	LOW	MEDIUM	HIGH	MEDIUM	MEDIUM
NOISE IMMUNITY	MEDIUM	POOR	GOOD	EXCEL	V. GOOD	V. GOOD	POOR	V. GOOD	V. GOOD
PROPAGATION DELAY	LOW	HIGH	LOW	MEDIUM	LOW	LOW	LOW	LOW	LOW
SPEED-POWER PRODUCT	MEDIUM	MEDIUM	MEDIUM	HIGH	LOW	MEDIUM	HIGH	MEDIUM	MEDIUM
CLOCK RATE	HIGH	MEDIUM	MEDIUM	MEDIUM	V. HIGH	V. HIGH	HIGH	LOW	HIGH

V. = VERY