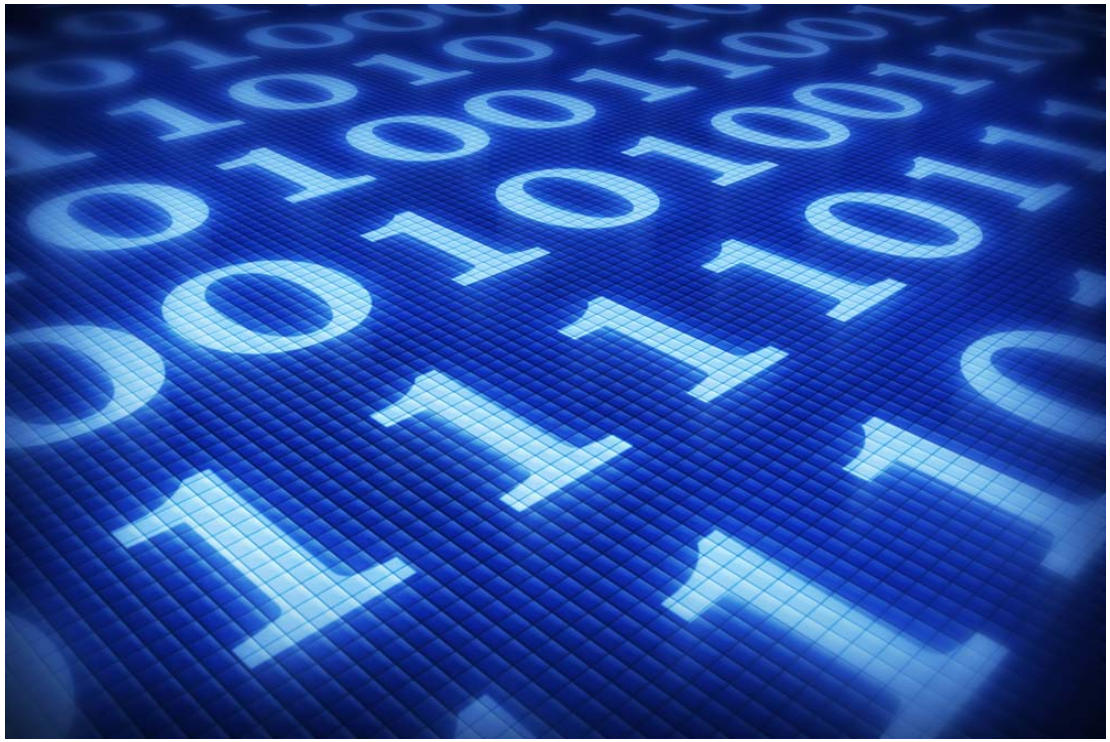
Chapter 25

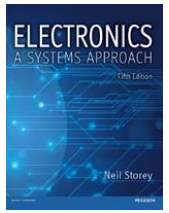
Last time: Sequential logic

- Introduction
- Bistables
- Monostables or one-shots
- Astables
- Timers
- Memory registers
- Shift registers
- Counters



Key points

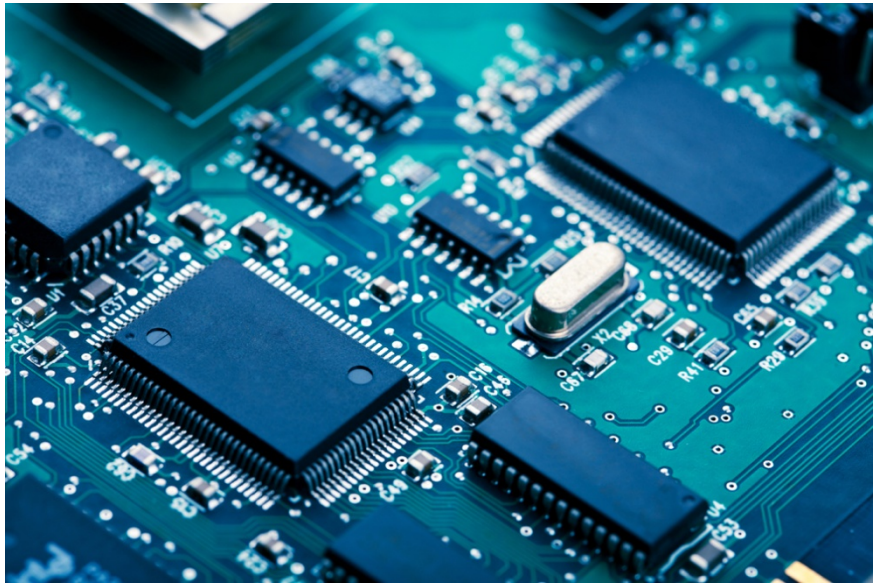
- Sequential logic circuits have the characteristic of memory
- Among the most important groups of sequential components are the various forms of multivibrator
  - bistables
  - monostables
  - astables
- The most widely used form is the bistable which includes
  - latches, edge-triggered flip-flops and master/slave devices
- Registers form the basis of various memories
- Counters are widely used in a range of applications
- Monostables and astables perform a range of functions

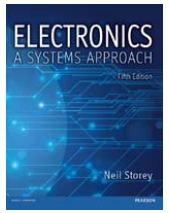


Chapter 26

Today: Digital devices

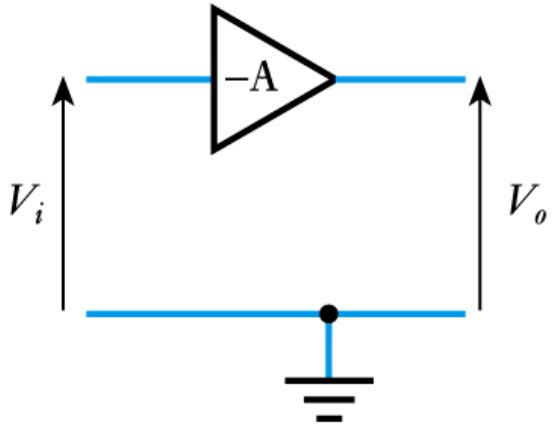
- Introduction
- Gate characteristics
- Logic families
- TTL
- CMOS
- Interfacing
- Noise and EMC in digital systems



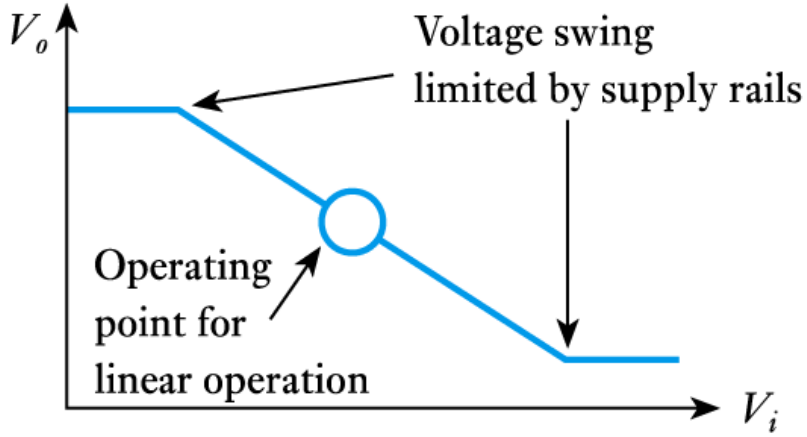


Gate characteristics

- **The inverter or NOT gate**
  - consider the characteristics of a simple inverting ***amplifier*** as shown below
  - we normally use only the **linear region**

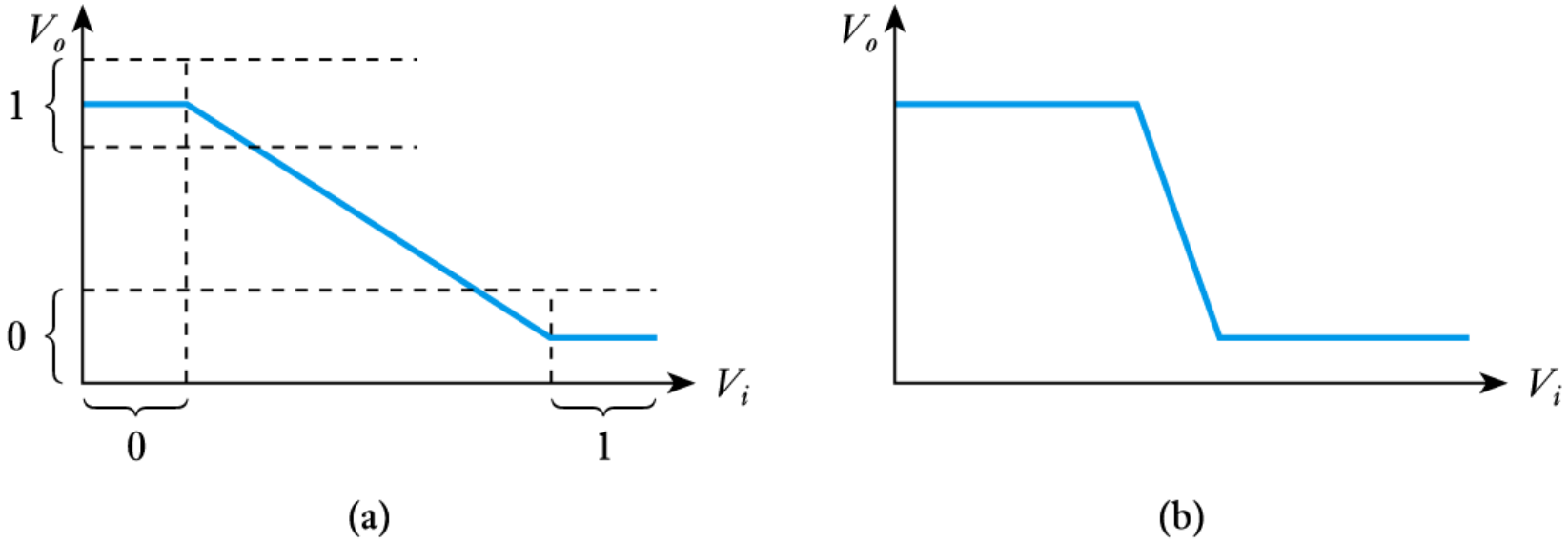


(a) Circuit symbol



(b) Characteristic

- We can use an inverting amplifier as a logical inverter but using only the **non-linear** region (a)

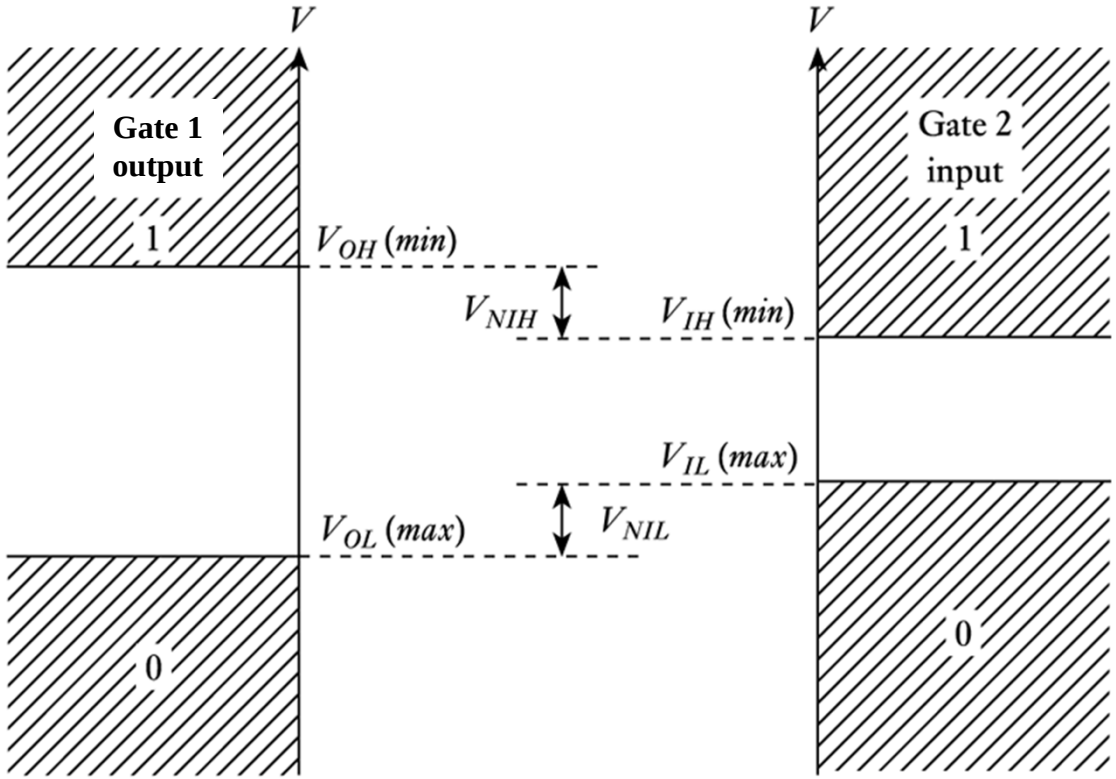


- And we use circuits with a rapid transition between the non-linear regions – as in (b)

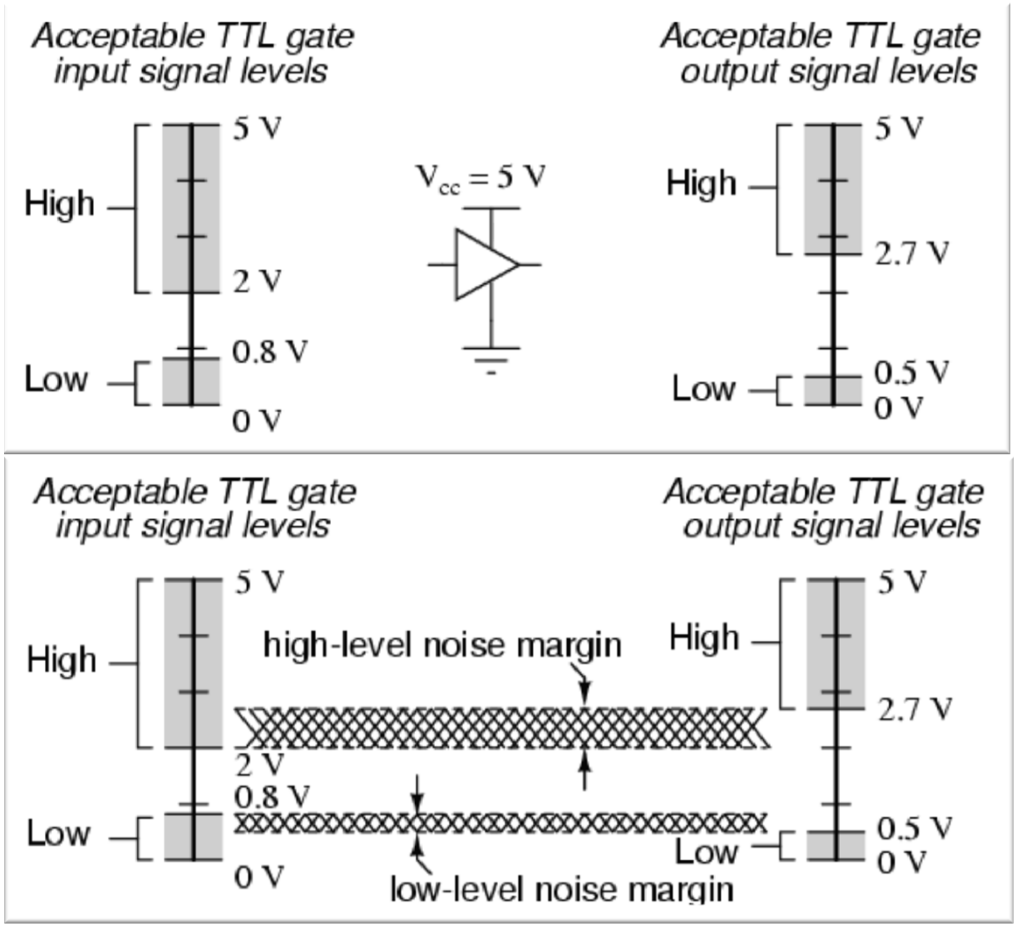
- **Logic levels**
  - The voltage ranges representing ‘0’ and ‘1’ represent the **logic levels** of the circuit
  - Often **logic 0** is represented by a voltage close to 0 V but the allowable voltage range varies considerably
  - The voltage used to represent **logic 1** also varies greatly. In some circuits it might be 2-4 V, while in others it might be 12-15 V
  - In order for one gate to work with another the logic levels must be compatible

- **Noise immunity**
  - Noise is present in all real systems
  - This adds random fluctuations to voltages representing logic levels
  - To cope with noise, the voltage ranges defining the logic levels are more tightly constrained at the output of a gate than at the input
  - Thus small amounts of noise will not affect the circuit
  - The maximum noise voltage that can be tolerated by a circuit is termed its **noise immunity**, V_{NI}

- **A graphical representation of noise immunity**



A graphical representation of noise immunity



Tolerance ranges for output are narrower than for input signal levels

Ensures that output signal levels will be acceptable to input of next gate even if it picks up some noise

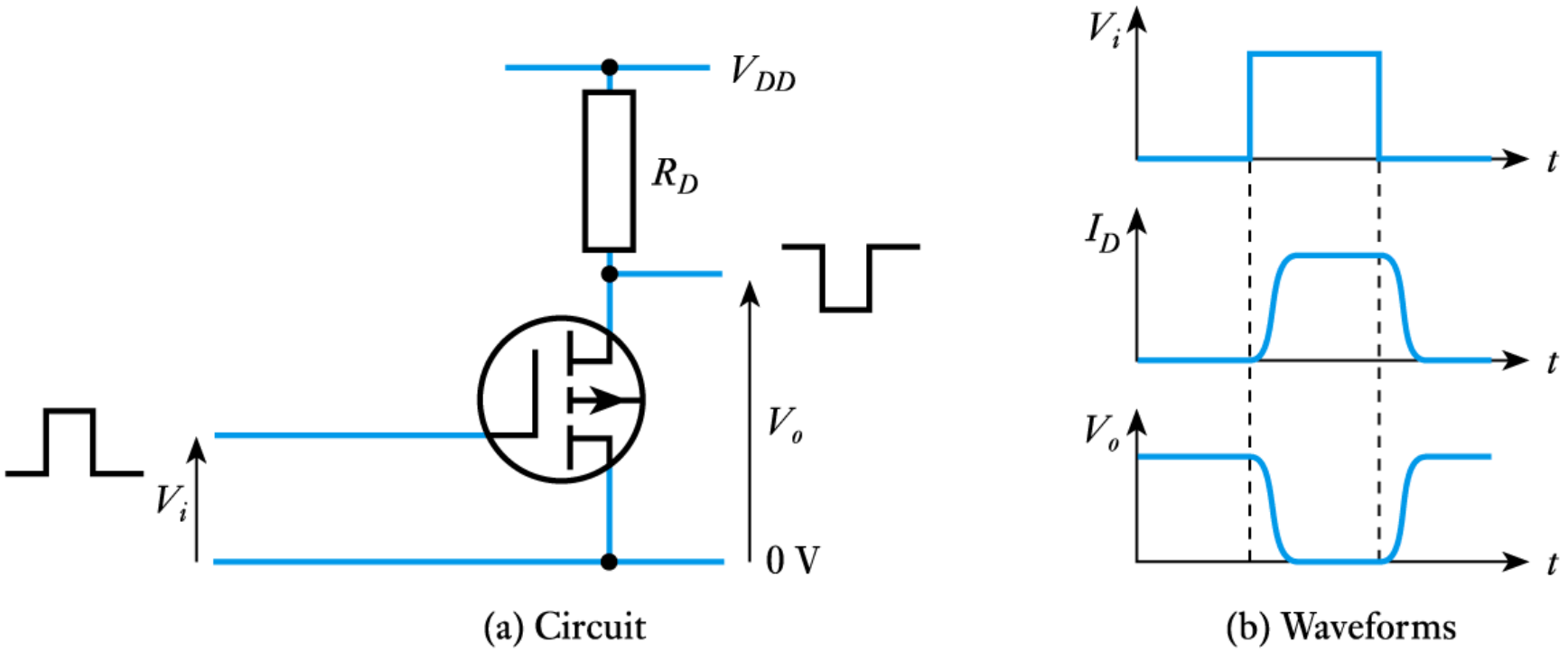
How much Noise?
The difference between the output and input tolerance ranges is called the *noise margin* of the gate.

$$V_{NIH} = V_{OH(min)} - V_{IH(min)}$$
$$V_{NIL} = V_{IL(max)} - V_{OL(max)}$$

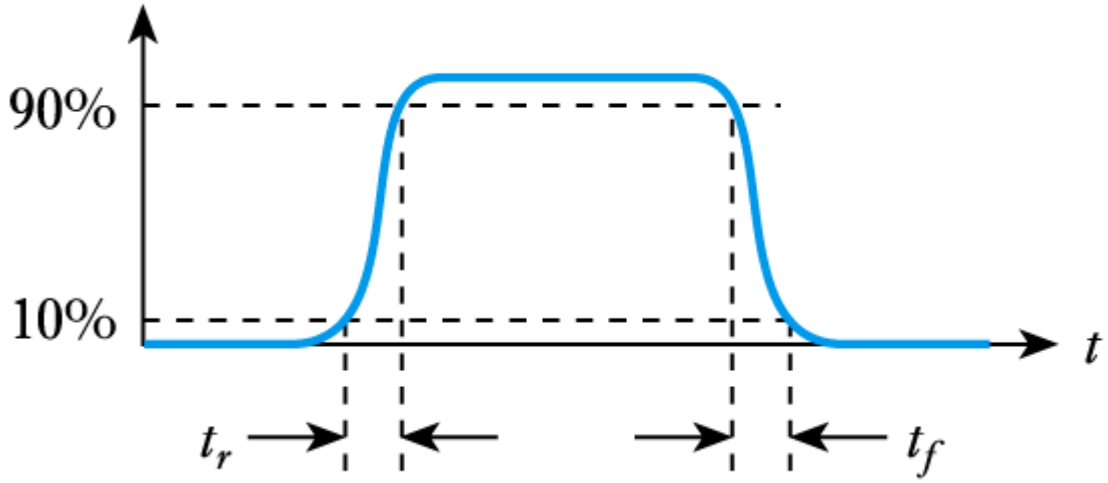
Video 26A

- **Transistors as switches**
  - Both FETs and bipolar transistors make good switches
  - Neither form produce *ideal* switches and their characteristics are slightly different
  - Both forms of device take a finite time to switch and this produces a slight delay in the operation of the gate
  - This is termed the **propagation delay** of the circuit

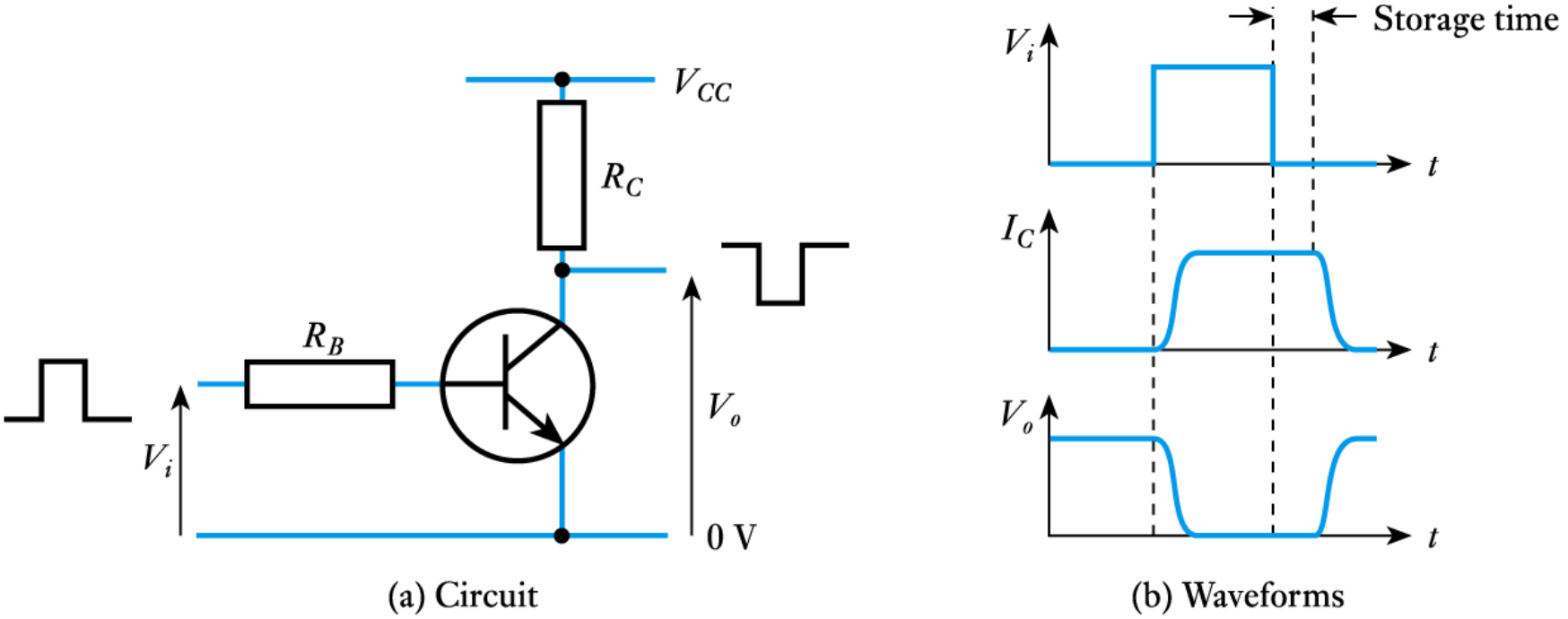
The FET as a logical switch



- **Rise and fall times**
  - because the waveforms are not perfectly square we need a way of measuring switching times
  - we measure the **rise time**, t_r and **fall time**, t_f as shown below

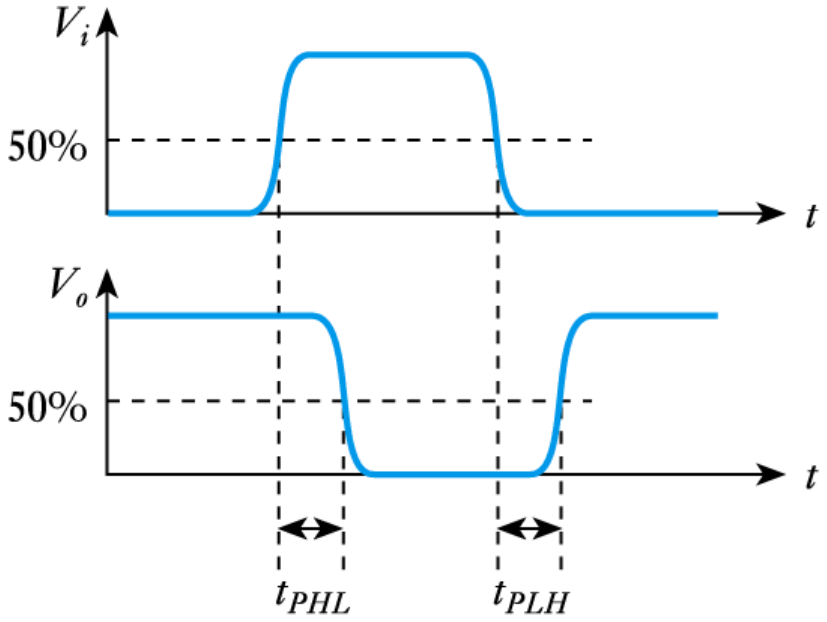


- **The bipolar transistor as a logical switch**



- When the input voltage to a bipolar transistor is high the transistor turns ON and the output voltage is driven down to its **saturation voltage,** which is about 0.1 V
- However, saturation of the transistor results in the storage of excess charge in the base region
- This *increases the time taken to turn OFF the device* – an effect known as **storage time**
- This makes the device faster to turn ON than OFF
- Some switching circuits increase speed by preventing the transistors from entering saturation

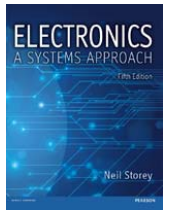
- **Timing considerations**
  - all gates have a certain **propagation delay time**, t_{PD}
  - this is the average of the two switching times



$$t_{PD} = \frac{1}{2}(t_{PHL} + t_{PLH})$$

Logic families

Video 26B

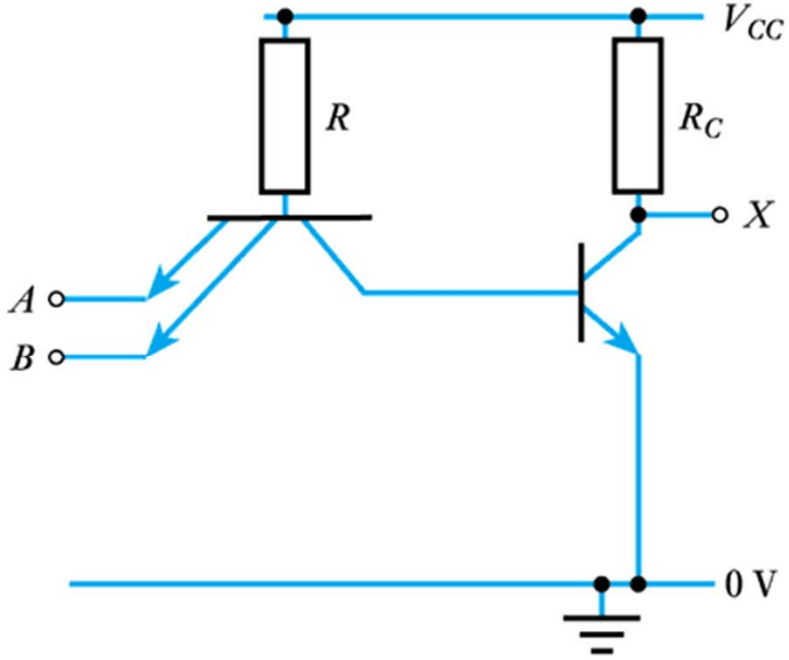


26.3

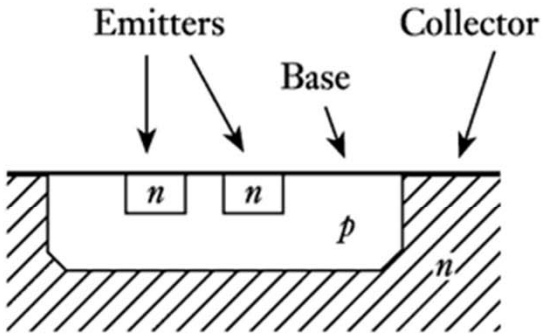
- We have seen that different devices use different voltages ranges for their logic levels
- They also differ in other characteristics
- In order to assure correct operation when gates are interconnected they are normally produced in families
- We will look briefly at the most important ones, namely **TTL and CMOS**

Inputs can source current
Outputs can sink current

- **Transistor-transistor logic (TTL)**
  - a simple TTL NAND gate using BJT transistors

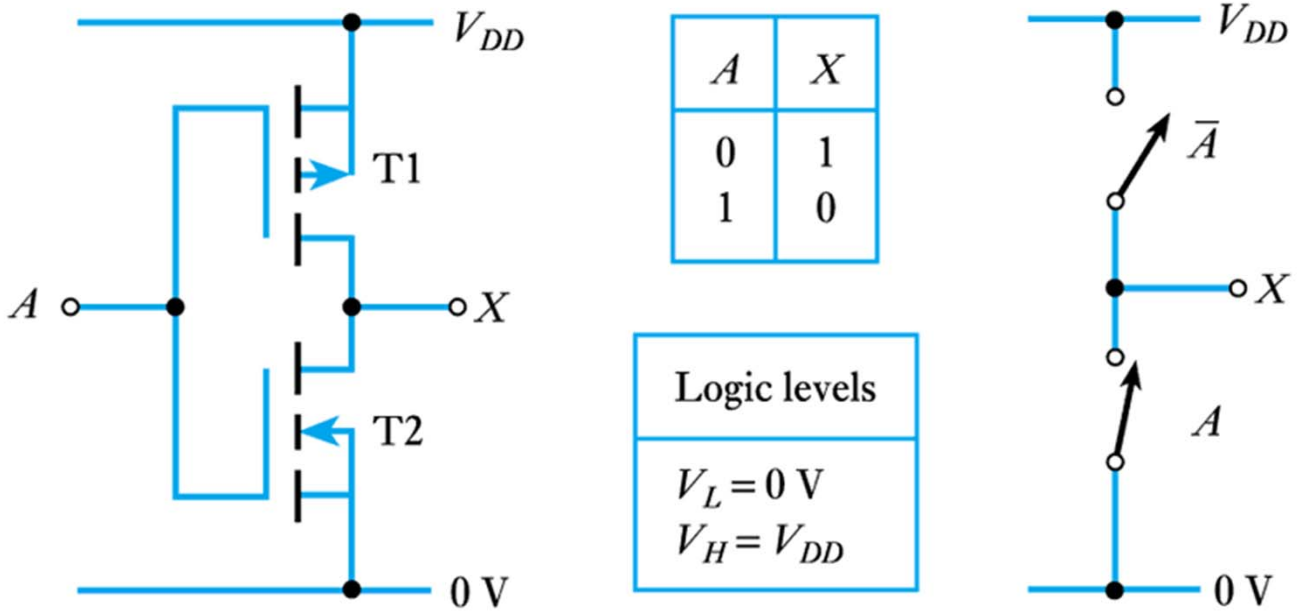


(a) Circuit

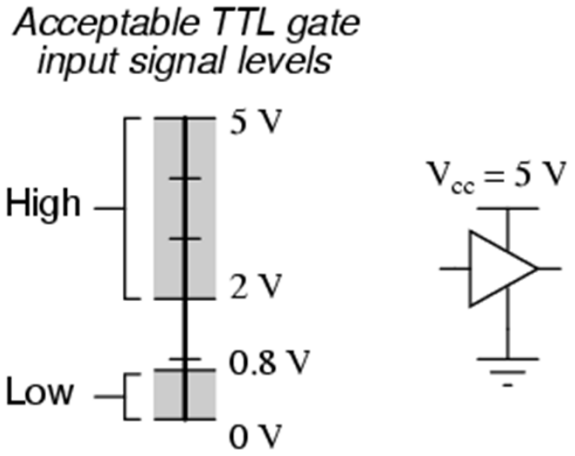


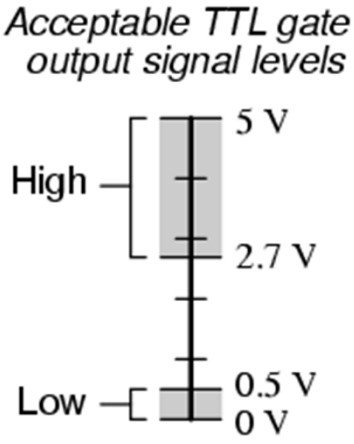
(b) Multi-emitter transistor

- **Complementary metal oxide semiconductor (CMOS) logic**
  - a CMOS inverter using FET transistors



Noise immunity TTL (top) vs. CMOS (bottom)





noise immunity in logic 1 (high)

$V_{NIH} = V_{OH(min)} - V_{IH(min)}$

$= 2.4 - 2.0$

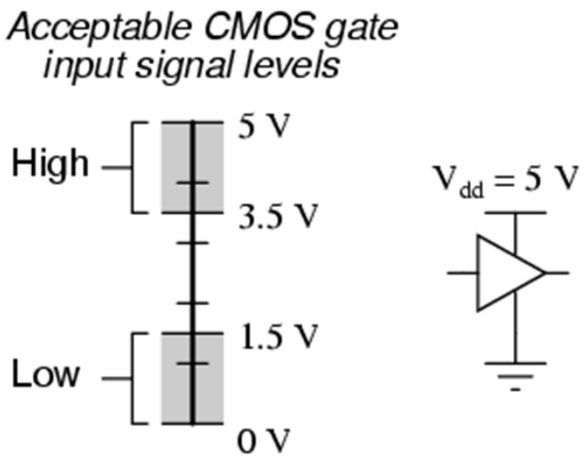
$= 0.4$ V

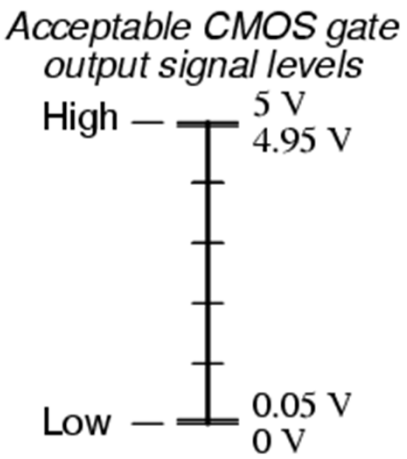
noise immunity in logic 0 (low)

$V_{NIL} = V_{IL(max)} - V_{OL(max)}$

$= 0.8 - 0.4$

$= 0.4$ V





noise immunity in logic 1 (high)

$V_{NIH} = V_{OH(min)} - V_{IH(min)}$

$= V_{DD} - 0.7 \times V_{DD}$

$= 0.3 \times V_{DD}$

noise immunity in logic 0 (low)

$V_{NIL} = V_{IL(max)} - V_{OL(max)}$

$= 0.3 \times V_{DD} - 0$

$= 0.3 \times V_{DD}$

A comparison of logic families

Transistor-Transistor (TTL) Logic (bipolar)
Metal oxide semiconductor (MOS) logic (FET)

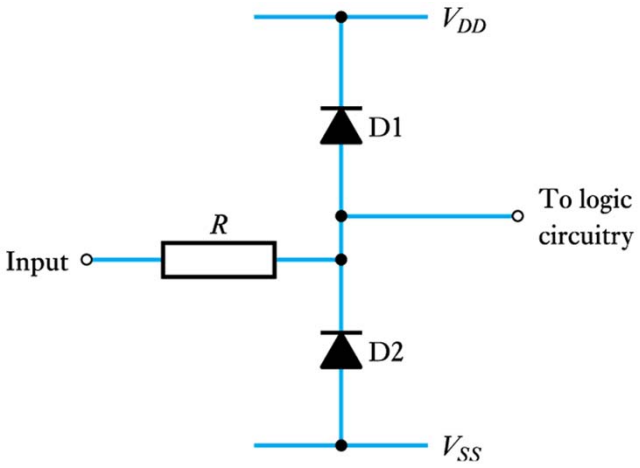
Parameter	TTL	CMOS
Basic gate	NAND	NAND-NOR
Fan-out	10	>50
Power per gate (mW)	1 – 22	1@1 MHz
Noise immunity	Very good	Excellent
T_{PD} (ns)	.5 – 33	1.5 – 200

Inputs can source current
Outputs can sink current

- **TTL inputs**
  - unused inputs, if left unconnected, will float to logical 1
  - it is not advisable to allow them to float since they are then very susceptible to noise
  - unused inputs should be tied to ground (logic 0) or through a resistor to the positive supply rail (logic 1)
    - unused inputs to an AND or NAND gate should be tied *high*
    - unused inputs to an OR or NOR gate should be tied *low*

Can be easily “blown” with too high a voltage
Sensitive to static

- **CMOS inputs**
  - CMOS gate protection circuitry

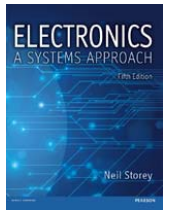


  - CMOS inputs must *not* be left unconnected
  - unused inputs should be tied to ground (logic 0) or to the positive supply rail (logic 1)
    - unused inputs to an AND or NAND gate should be tied *high*
    - unused inputs to an OR or NOR gate should be tied *low*

- **CMOS outputs**
  - typical output resistance of about 250Ω (5V operation)
  - high input resistance produces a high fan-out
  - propagation delay increases with the number of gates being driven
  - if high-speed operation is not required, at least 50 gates can be driven from a single output
  - no CMOS equivalent of open-collector output
  - some CMOS gates have three-state facility

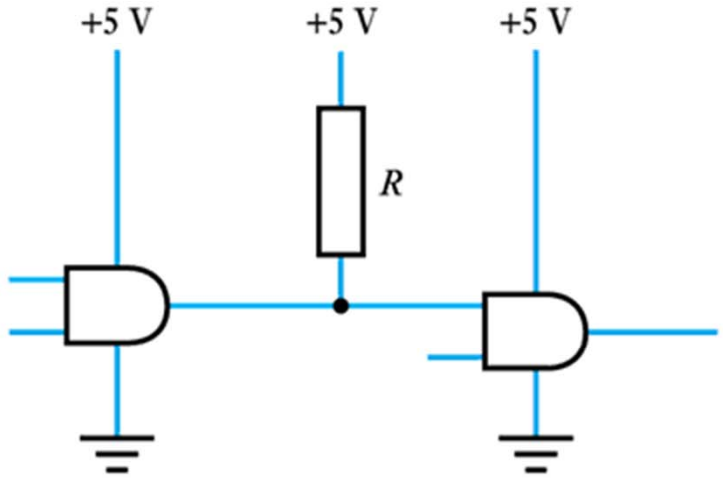
Interfacing TTL and CMOS

Video 26F



26.6

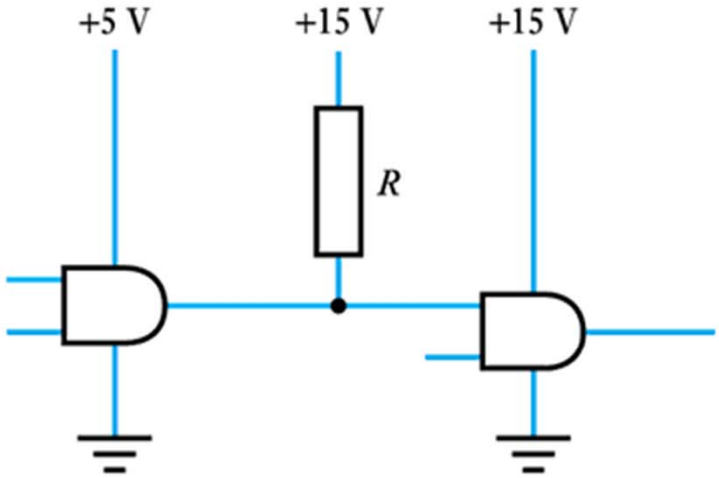
- **Driving CMOS from TTL**



TTL gate – open collector or totem-pole

CMOS gate

(a) Driving 5 V CMOS logic

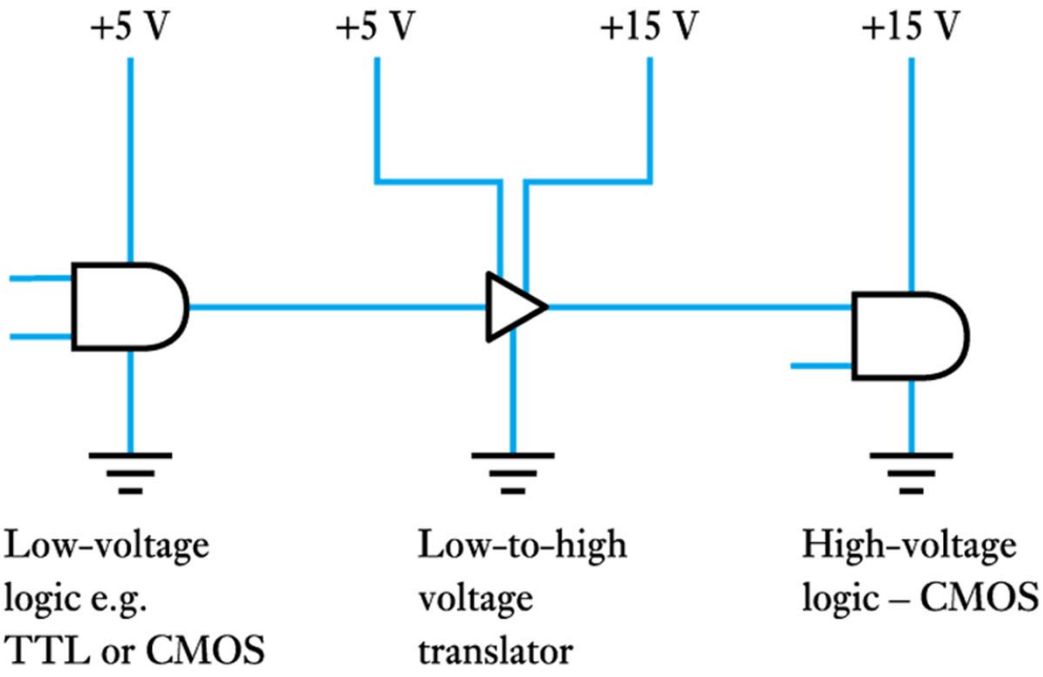


TTL gate – high-voltage open collector

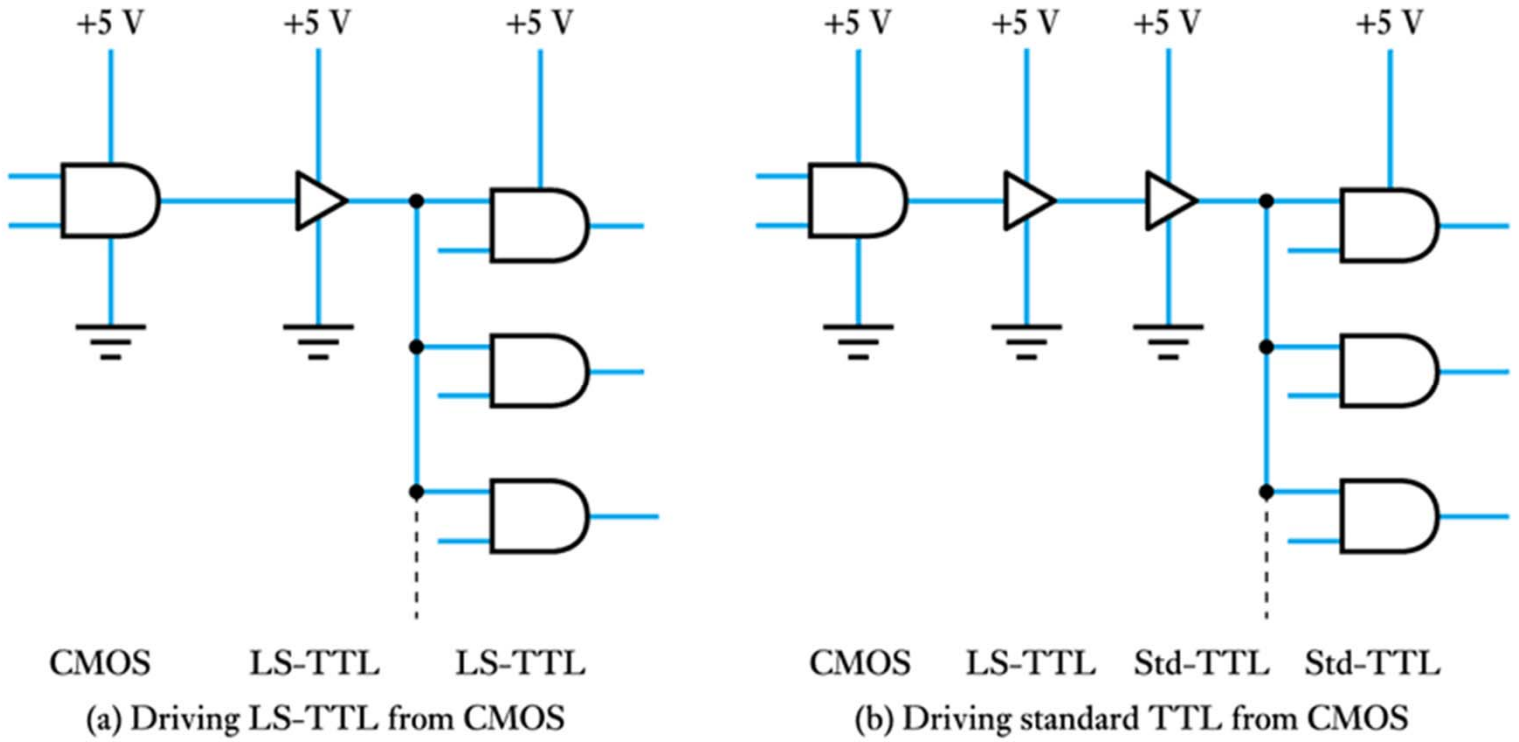
CMOS gate

(b) Driving 15 V CMOS logic

– use of a low-to-high voltage translator



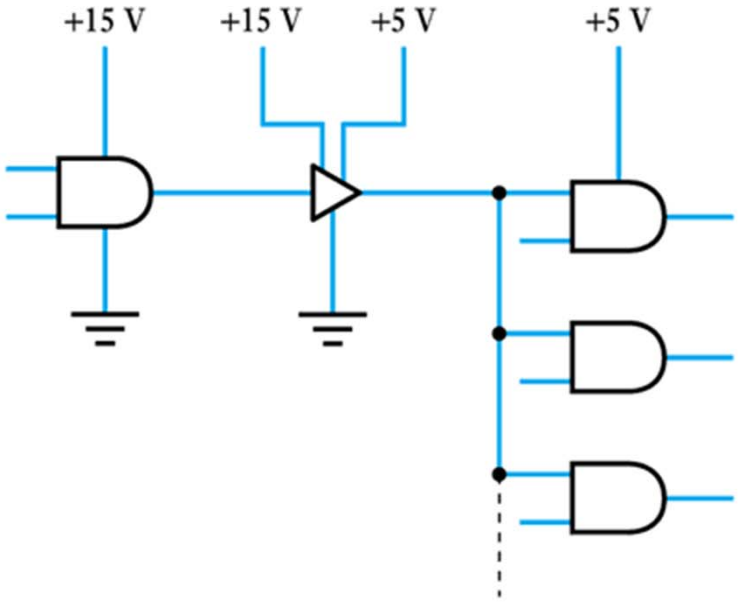
Driving TTL from CMOS



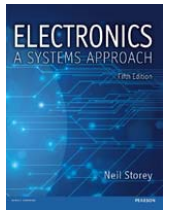
(a) Driving LS-TTL from CMOS

(b) Driving standard TTL from CMOS

– using a high-to-low voltage translator



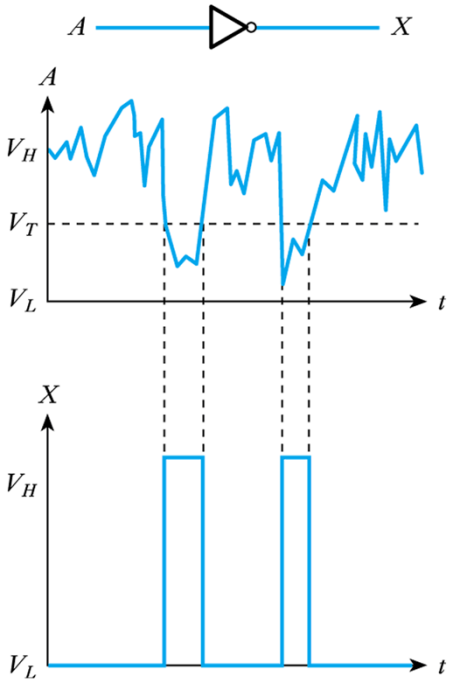
(c) Driving TTL from high-voltage CMOS



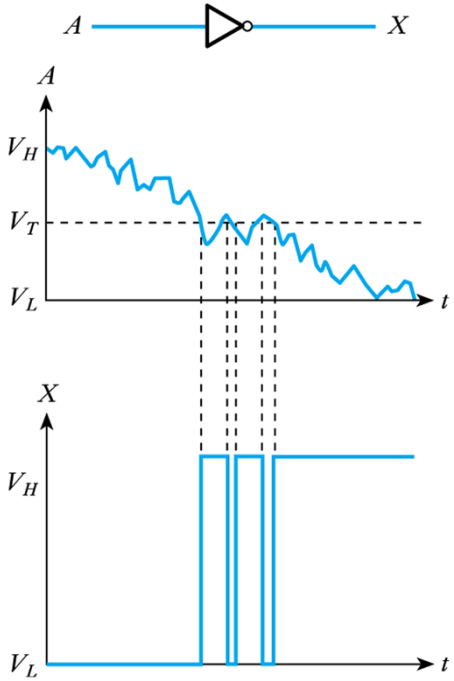
Noise and EMC in digital systems

- **Digital noise sources**
  - Electronic noise
  - Interference
  - Internal noise
  - Power supply noise
  - CMOS switching transients

The effects of noise in digital systems



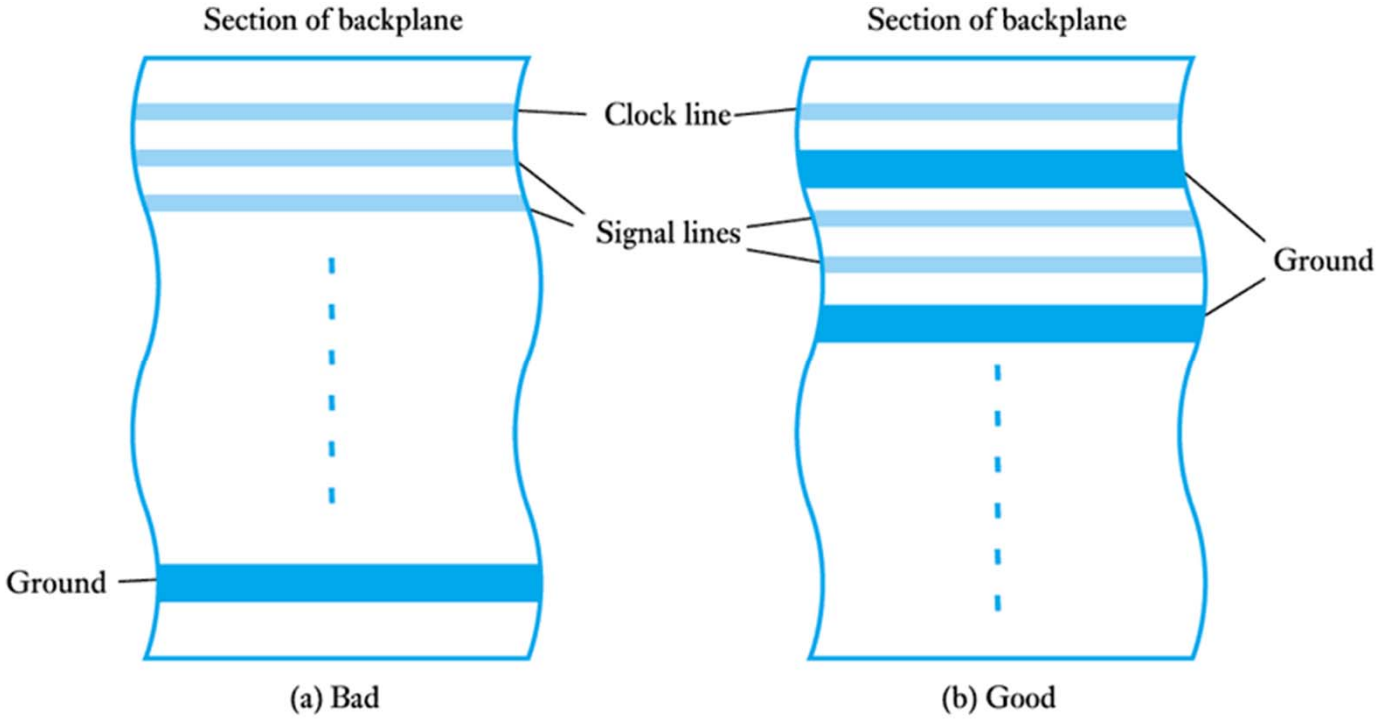
Steady logic voltages



Slowly varying signals

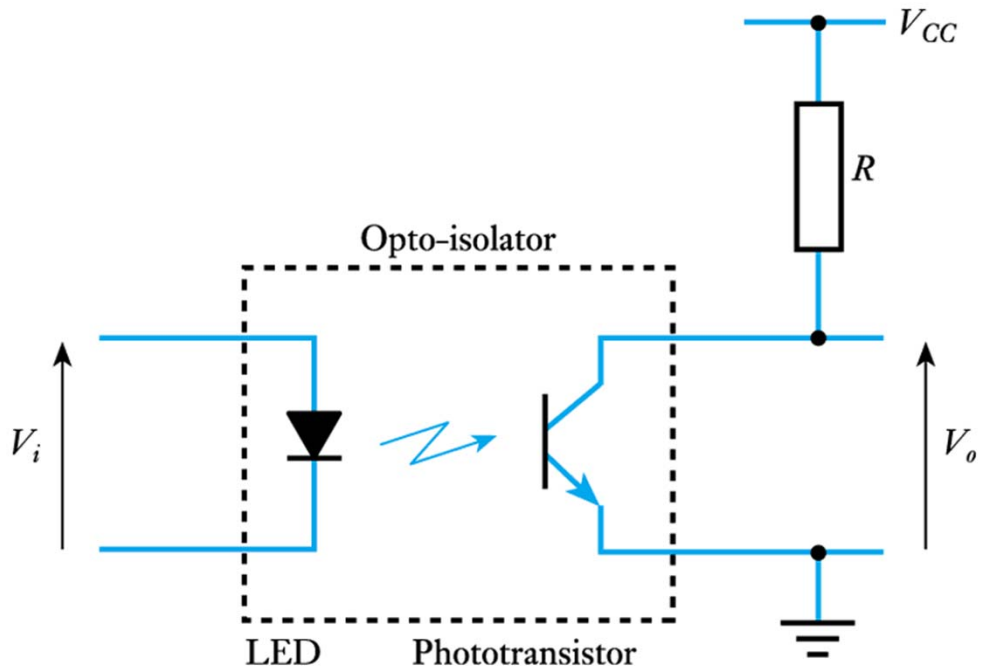
Noise and EMC in digital circuits

- **Designing digital systems for EMC**
  - backplane arrangements



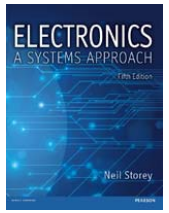
Good for long input wires with digital data

– opto-isolation

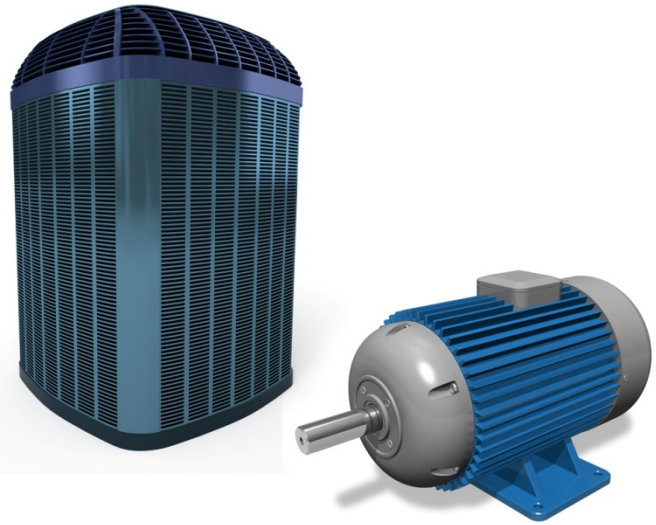


Further Study

Video 26H Further Study



- The Further Study section at the end of Chapter 26 is concerned with noise in digital systems.
- It looks at a unit that takes inputs from two mechanical switches and provides outputs to control a heater and a motor.



- Consider how the system could be designed to minimise the effects of noise on its operation, and then look at the treatment given in the video.

Key points

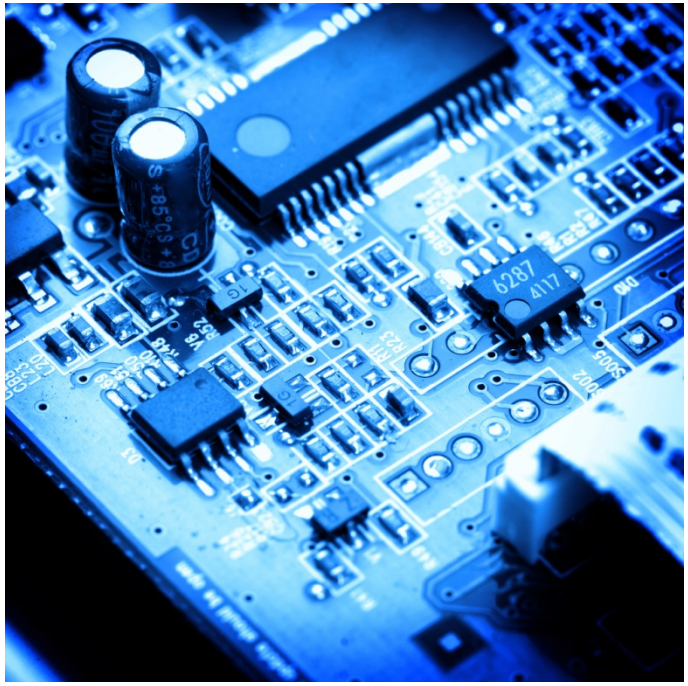
- Physical gates are not ideal components
- Logic gates are manufactured in a range of logic families
- The ability of a gate to ignore noise is its ‘noise immunity’
- Both MOSFETs and bipolar transistors are used in gates
- All logic gates exhibit a propagation delay
- The most widely used logic families are TTL and CMOS.
- Both TTL and CMOS gates are produced in a range of versions, each optimised for a particular characteristic
- Interface circuitry may be needed to link devices of different families
- Noise and EMC issues must be considered during design

- Questions?

Chapter 27

Implementing digital systems

- Introduction
- Array logic
- Microprocessors
- System-on-a-chip devices
- Selecting an implementation method



Introduction

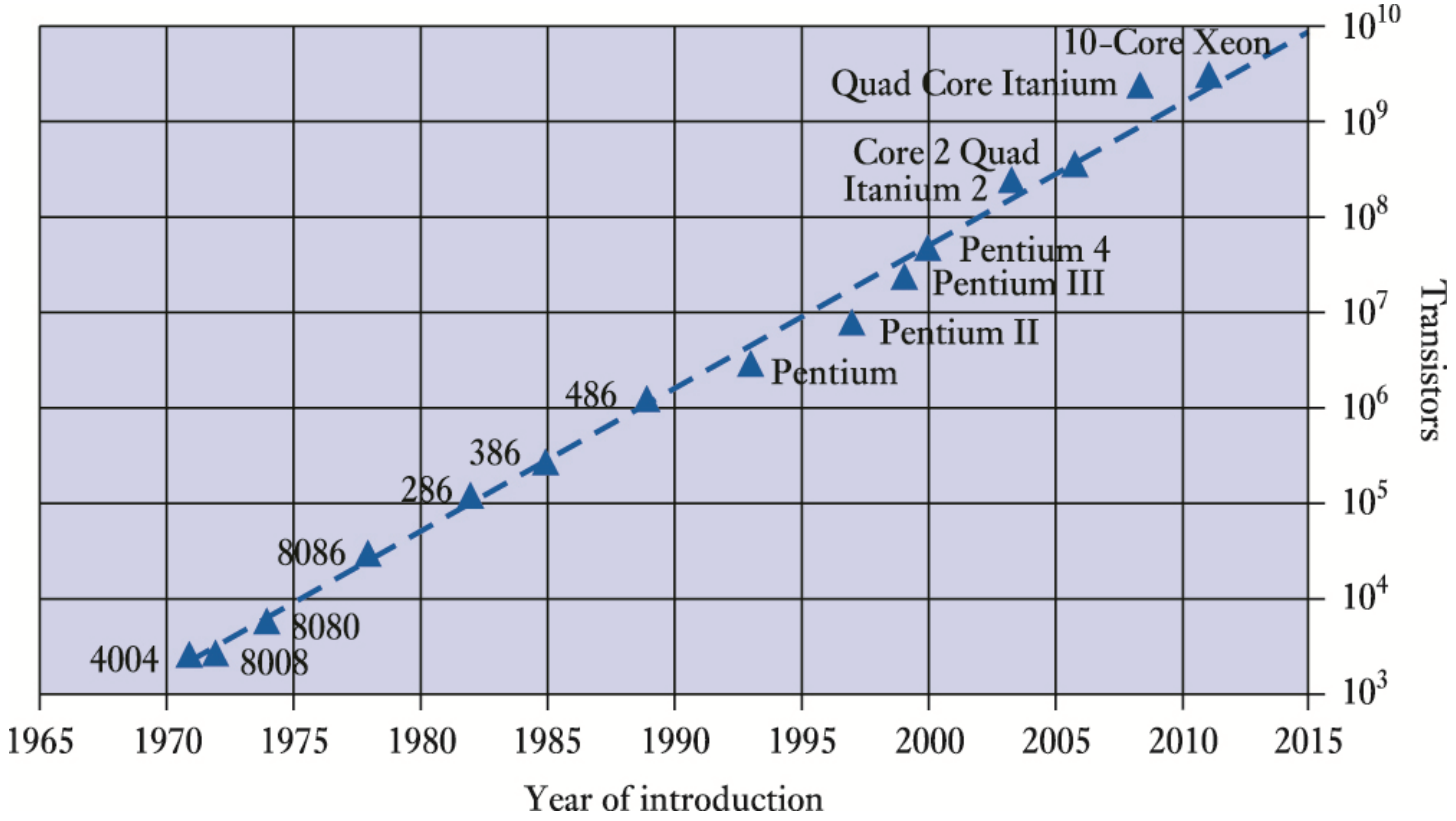
Video 27A

27.1

- In this lecture we will look at the techniques used to implement complex digital systems
- We will begin by looking at the nature of various complex integrated circuits
- The complexity of the integrated circuits used within electronic circuits is increasing rapidly
- Available integration level increases exponentially with time (**Moore's Law**)

Moore's Law

- **Integration densities of Intel microprocessors**



Array logic (for information)

- **Programmable logic devices (PLDs)**
  - these are examples of **uncommitted logic**
  - forms include:
    - programmable logic array (PLA)
    - programmable array logic (PAL) 3,
    - programmable read only memory (PROM)
    - generic array logic (GAL)
    - erasable programmable logic device (EPLD)
    - programmable electrically erasable logic (PEEL)
    - complex programmable logic device (CPLD)
    - field programmable gate array (FPGA)

Re-programmable

Arrays of arrays

Video 27B

- **Programmable logic array (PLA)**
  - has an array of inverters, AND gates and OR gates
  - can implement any logic function (given limits on numbers of inputs and outputs)

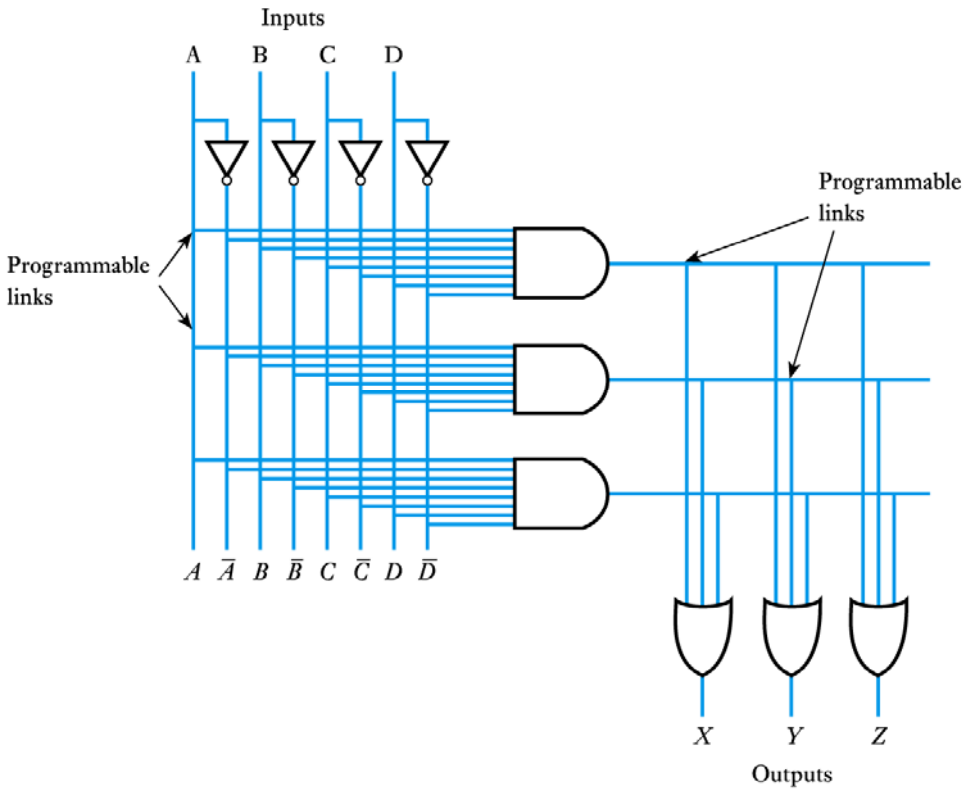
Example: consider a system with four inputs *A*, *B*, *C* and *D* and three output *X*, *Y* and *Z*, where

$$X = \overline{A}\,\overline{B}\,\overline{C}D + \overline{A}\,\overline{B}CD$$

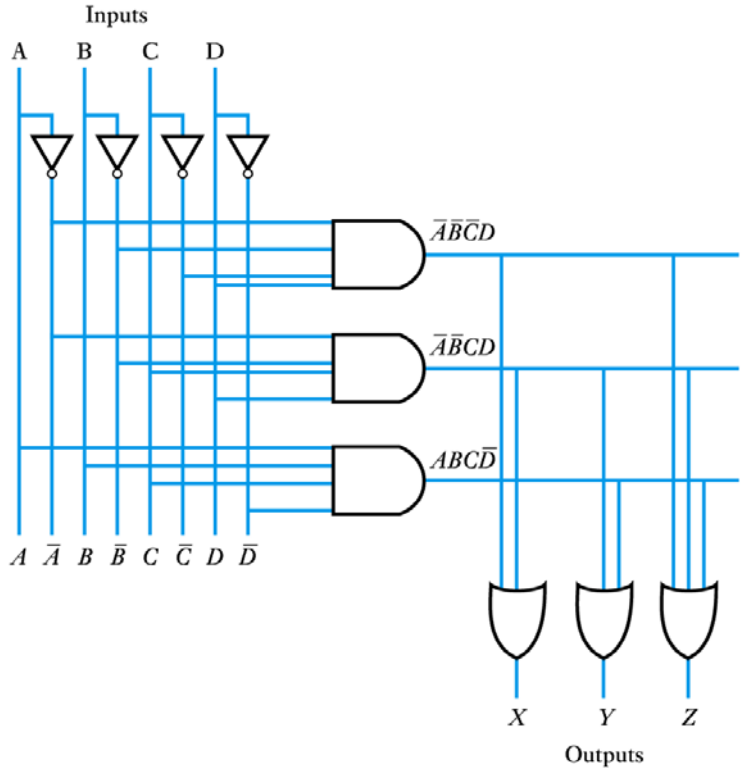
$$Y = \overline{A}\,\overline{B}CD + ABC\overline{D}$$

$$Z = \overline{A}\,\overline{B}\,\overline{C}D + \overline{A}\,\overline{B}CD + ABC\overline{D}$$

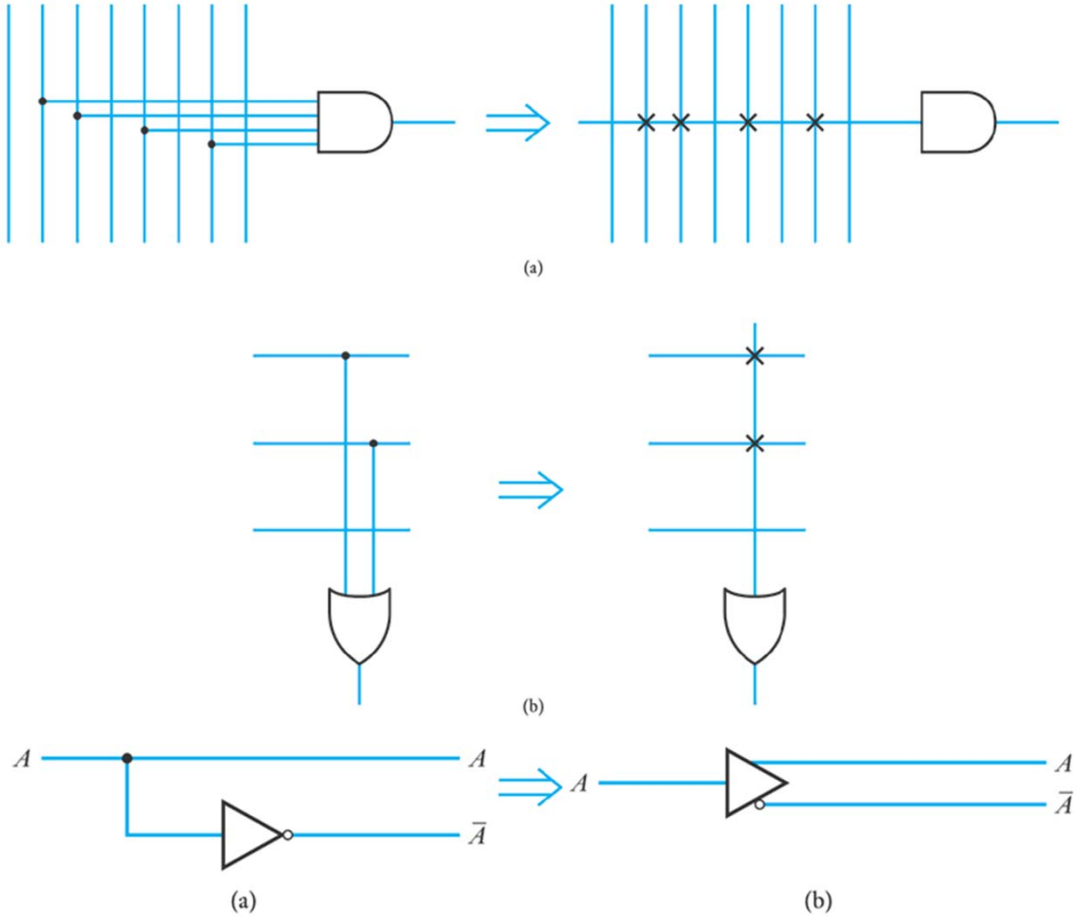
– the structure
of a simple PLA



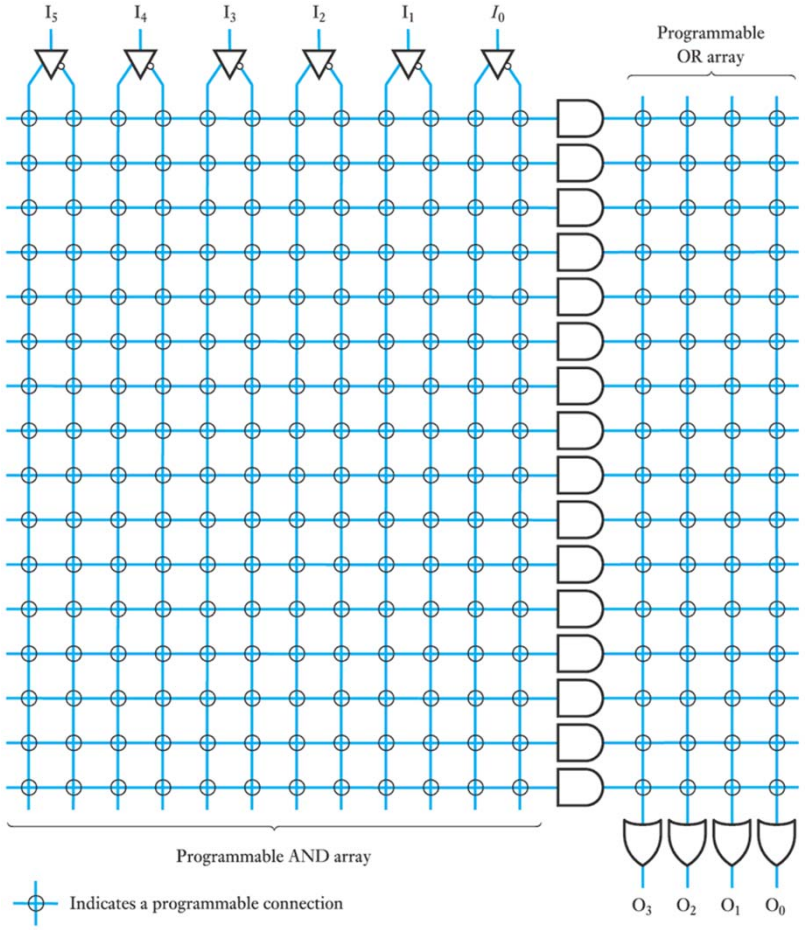
– the PLA programmed to give the required output functions
– the device is programmed by blowing fusible links at the various interconnection points



– logic array symbolic notation



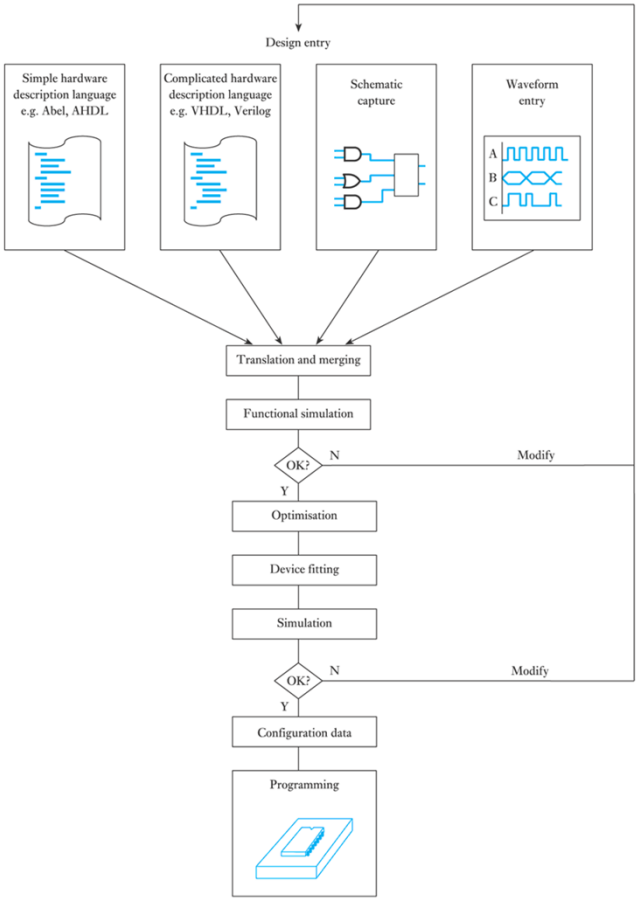
– a PLA with 6 inputs,
4 outputs and
16 product terms



Programmable read-only memory (PROM)

- similar structure to a PLA but has only 1 programmable array (NANDS as in the PAL) but fixed OR output array
- the pattern written into the OR array determines the outputs produced for each combination of inputs
- can be used to implement logic functions or to store **data** – when storing data the inputs are the **address**

- **Programming tools for array logic**
  - automated tools are used in almost all cases
  - often make use of **hardware description languages**
  - fuse maps are passed to a **programmer** to configure the device



- **Custom and semi-custom ICs**
  - Some equipment may contain a **custom IC** designed by the manufacturer
  - Others may have **semi-custom** or **application specific integrated circuits** (**ASIC**)
    - produced by combining a number of standard cells (such as registers, counters, input/output circuitry and memory)
    - much less costly than a complete design from scratch
- Very difficult to find out what they do!