Mechatronics Complete Notes

VARIOUS AUTHORS

UNIT NOTES

UNIT-I INTRODUCTION

INTRODUCTION

- 1969 -The ‘mechatronics’ word introduced by Tessturo Mori. He was a senior engineer of Japanese company Yaskawa Electric Corporation.
- 1971 – the company was granted the trademark rights on the word.
- 1971- 80- mostly the servo technology is used in mechatronics
- 1981-90-IT introduced. Microprocessors were embedded in Mechanical system

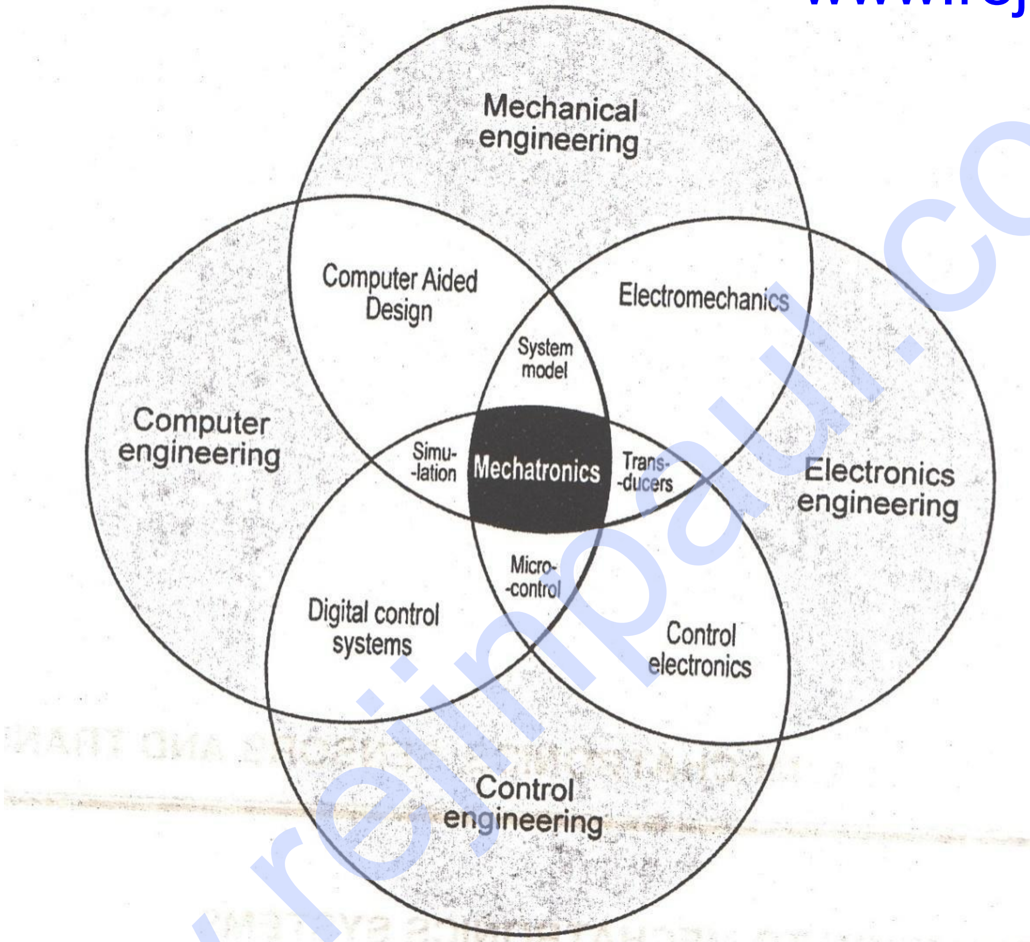
- 1991 -2000 – Communication technology was added. Remote operation and robotics were developed .

- 1996 – 1st journal IEEE on mechatronics was released.

- After 2000, finds application in aerospace, defence, bio-mechanics, automotive electronics, banking(ATM) etc.,

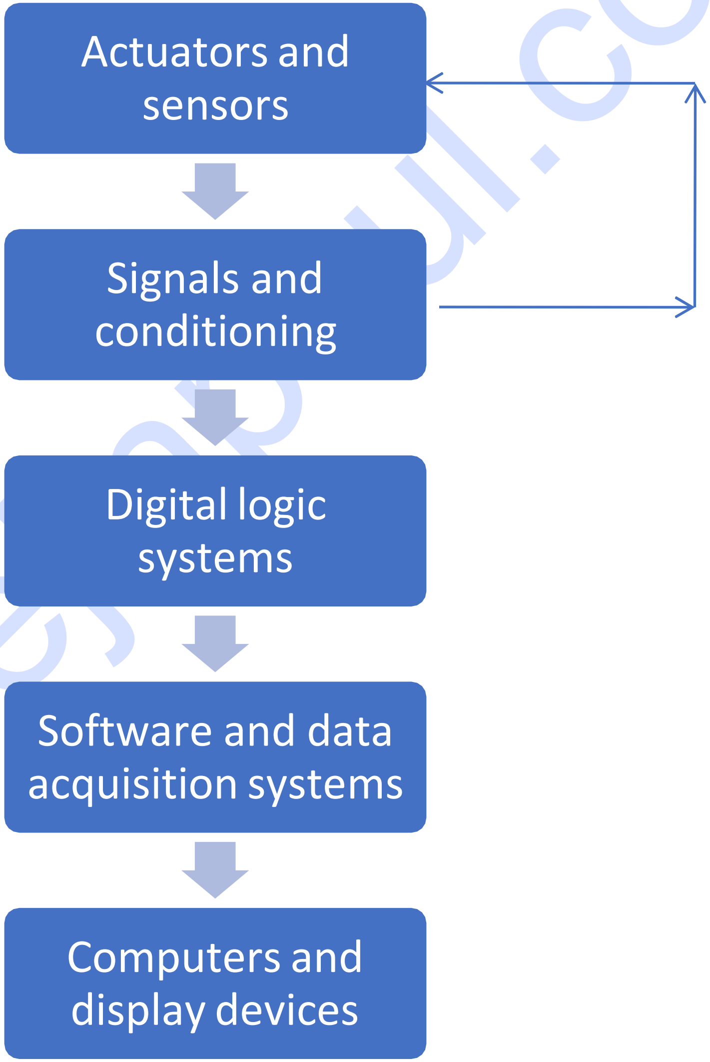
DEFINITION

- Mechatronics is " the synergistic integration of mechanics and mechanical engineering, Electronics, computer technology, and IT to produce or enhance products and system"



Graphical representation of mechatronics

Elements of mechatronics system



Elements of mechatronics system

- Actuators and sensors
  - Actuators – pneumatic & Hydraulic actuator, electromechanical actuators, electrical motor such as DC motor, AC motor, stepper motor, servo motor & piezo electric actuators
  - Sensors – linear and rotational sensor, acceleration sensor, force, torque and pressure sensor, temperature sensor, proximity sensors, light sensors
- Signals and conditioning
  - Two types: input and output
  - Input signal conditioning devices: discrete circuits, amplifiers, analog to digital(A/D) convertors, Digital to Analog (D/A) convertors.
  - Output signal conditioning devices: amplifiers, Digital to Analog (D/A) convertors, display decoders (DD) convertors, power transistors.

Elements of mechatronics system

- Digital logic systems
  - Logic circuits, micro controllers, programmable logic controllers(PLC), sequencing and timing controls, control algorithm.

- Software and data acquisition systems
  - Data logger, computer with plug in boards

- Computers and display devices
  - LED, CRT, LCD, digital displays etc.,

Examples of mechatronics systems

- NC and CNC machine tools, flexible manufacturing system, Prototyping & robots
- Photo copiers, laser printers & fax machines
- Automatic washing machines automatic ovens
- Automatic teller machine (ATM)
- Coin counter
- Automatic/digital camera, digital watch
- CT scan system, automatic blood testing equipment
- Automatic sliding door

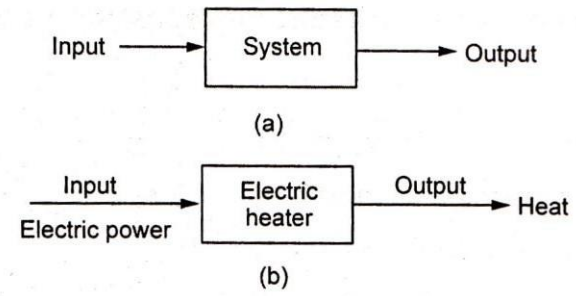
Advantages of Mechatronics systems

- Cost effective and good quality products
- High degree of flexibility to modify or redesign
- Very good performance characteristics
- Wide area of applications
- Greater productivity in case of manufacturing organization
- Posibility or remote controlling as well as centralized monitoring and control
- Greater extend of machine utilization

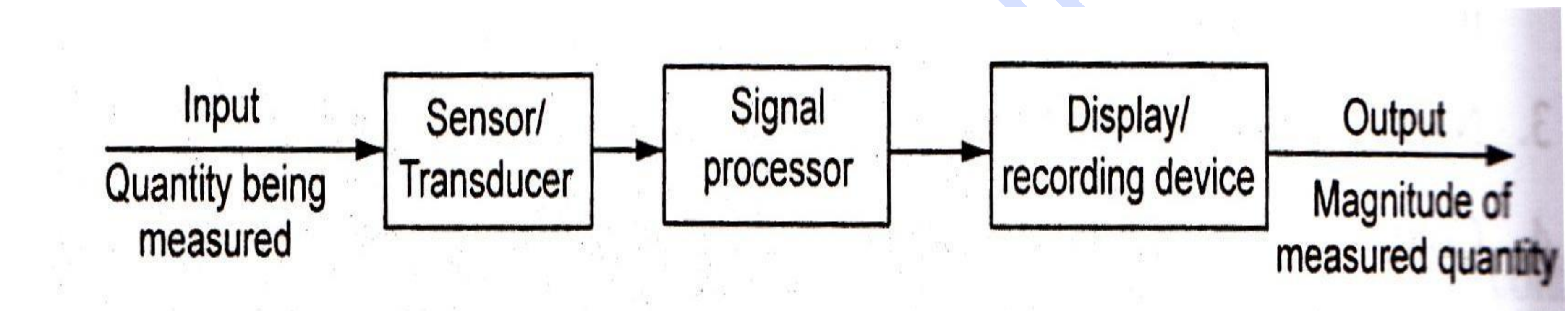
Disadvantages of Mechatronics systems

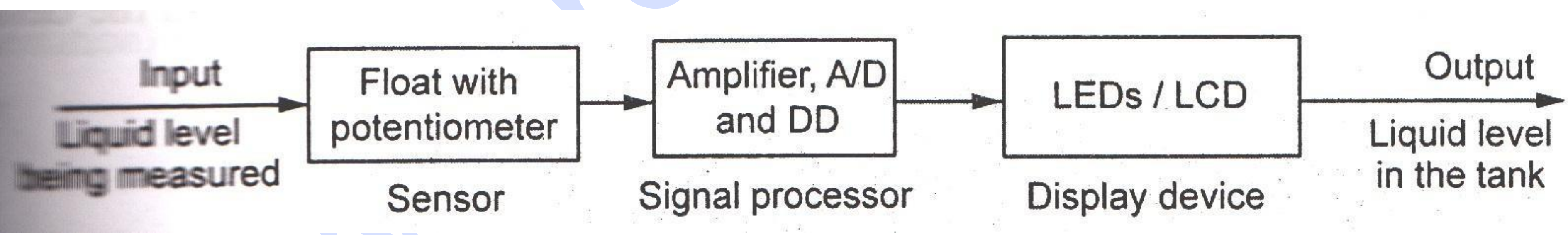
- High initial cost
- Multi-disciplinary engineering background required to design and implementation
- Need of highly trained workers
- Complexity in identification and correction of problem in the system

System


Input
System
Output
(a)
Input
Electric power
Electric heater
Output
Heat
(b)

Measurement system

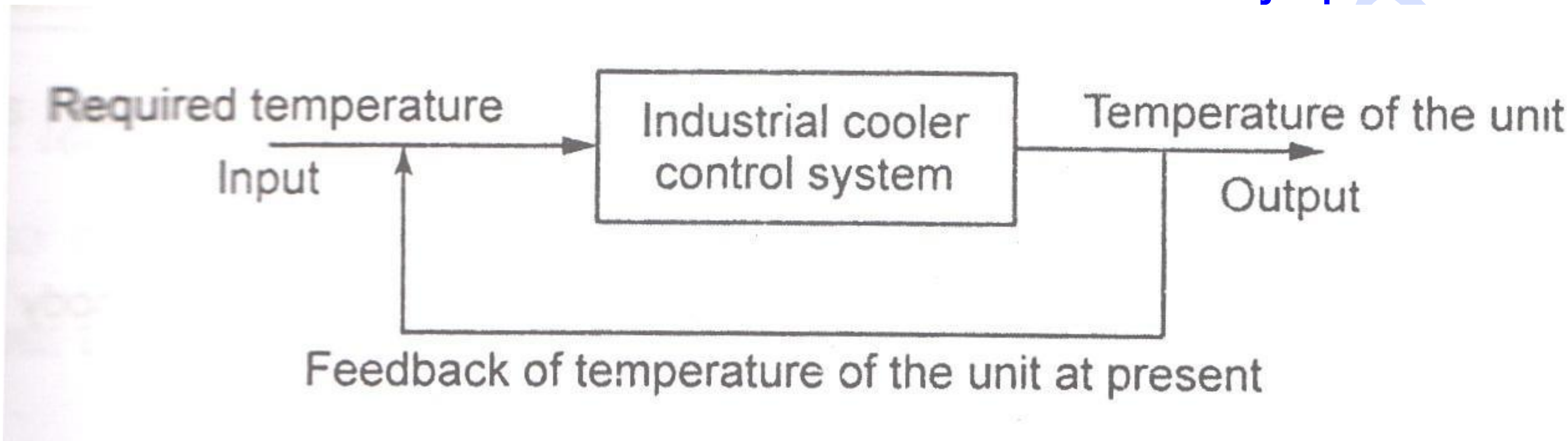




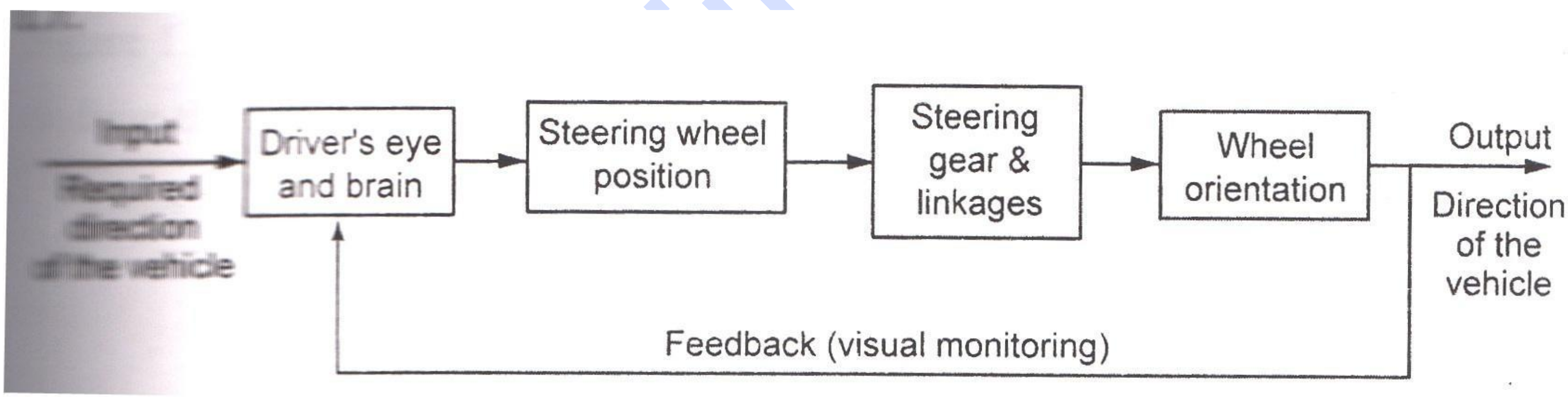
Liquid level measurement system

Control systems

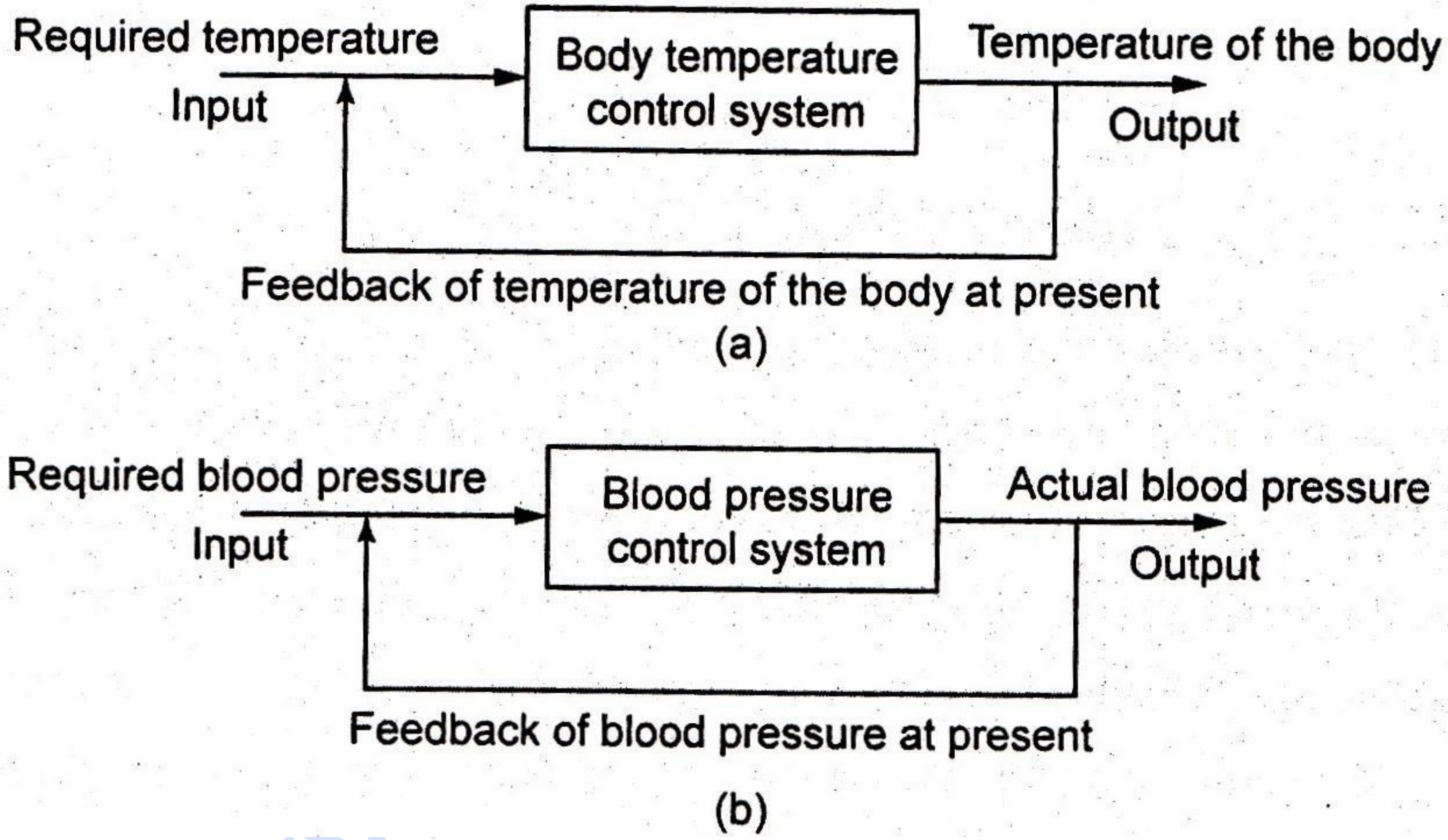
- A control system refers to a group of physical component connected or related in such a manner as to command direct or regulate itself or another system.



Cooling level control system



Steering control system of an automobile



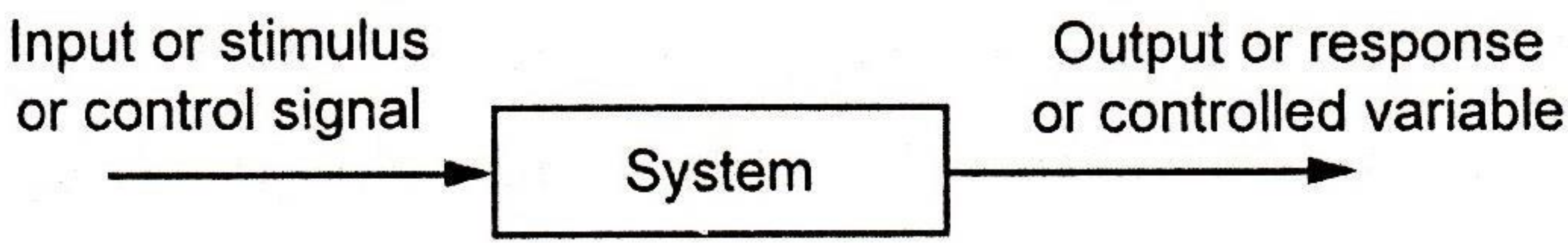
Temperature and blood pressure control system of human body

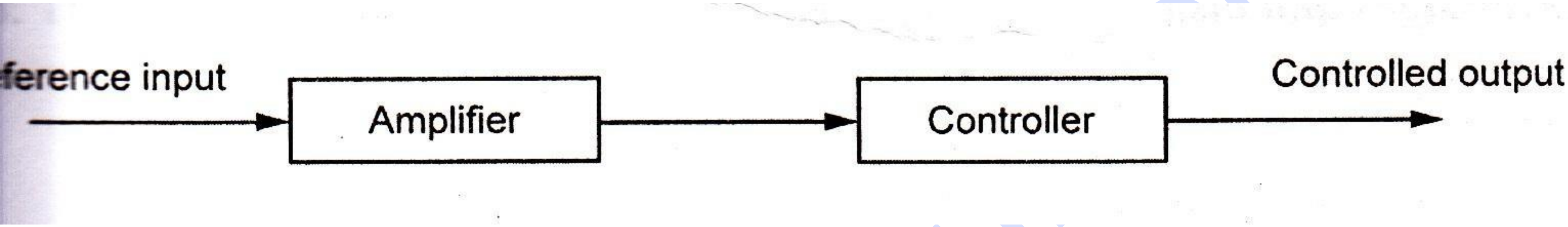
TYPES OF CONTROL SYSTEM

- Open loop control system
- Closed loop or feedback control system

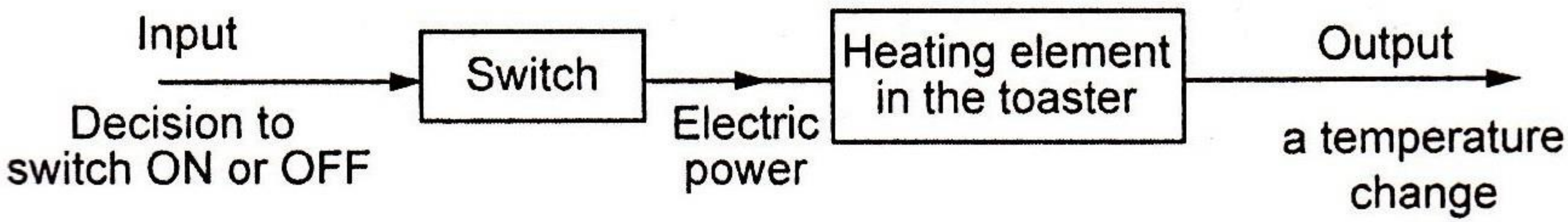
Open loop control system

- Open loop system are systems in which the output of a system is not used as a variable to control the system.





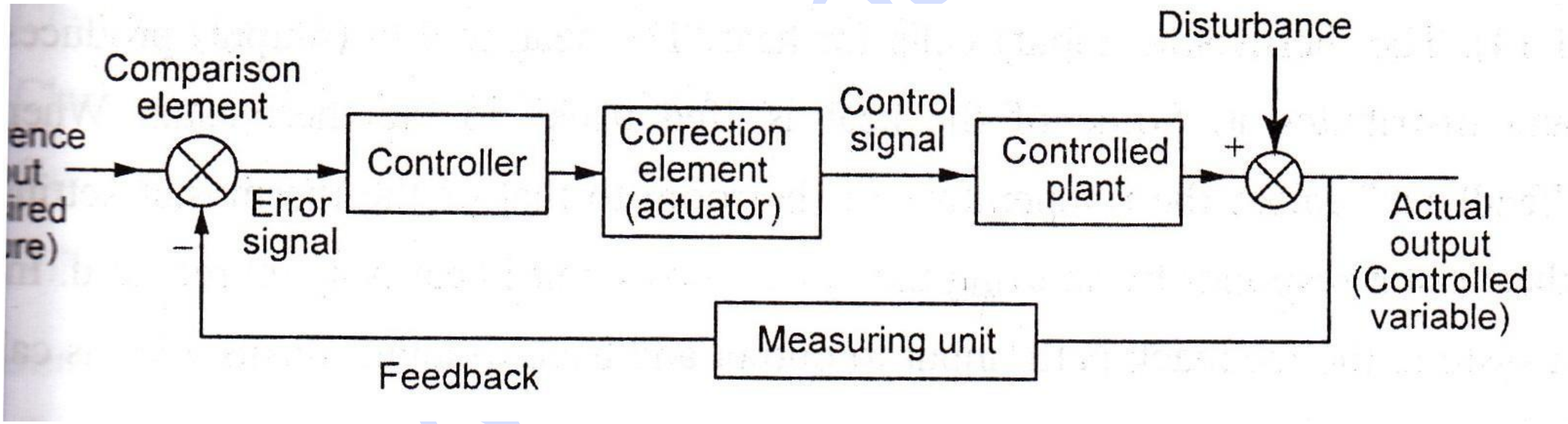
Element of open loop control systems

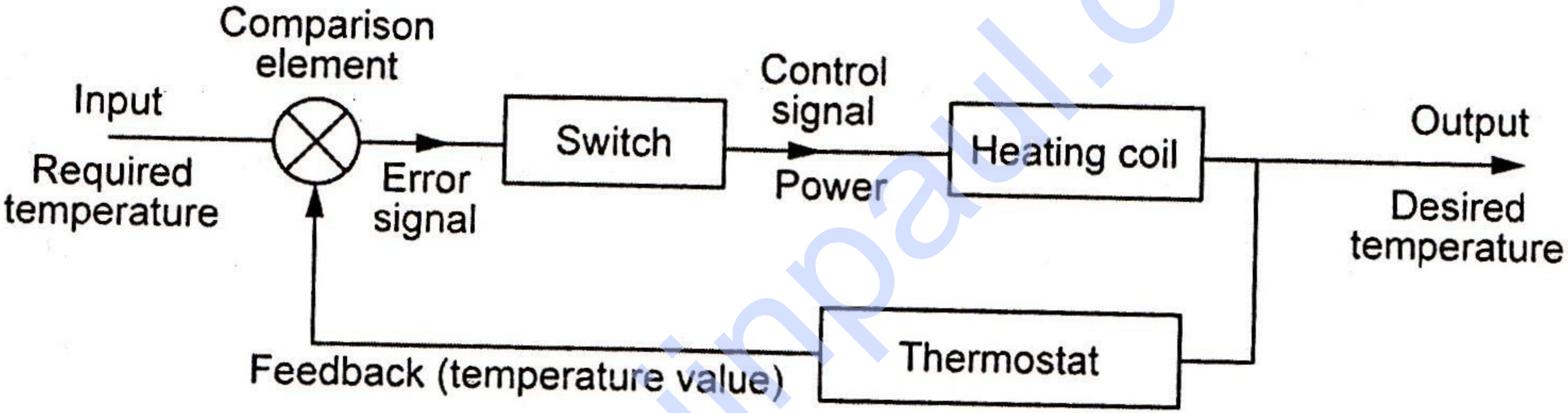


Bread toaster (open loop) control system

Closed loop control system

- Closed loop system uses on a feed back loop to control the operation of the system.





Room heating (Closed loop) control system

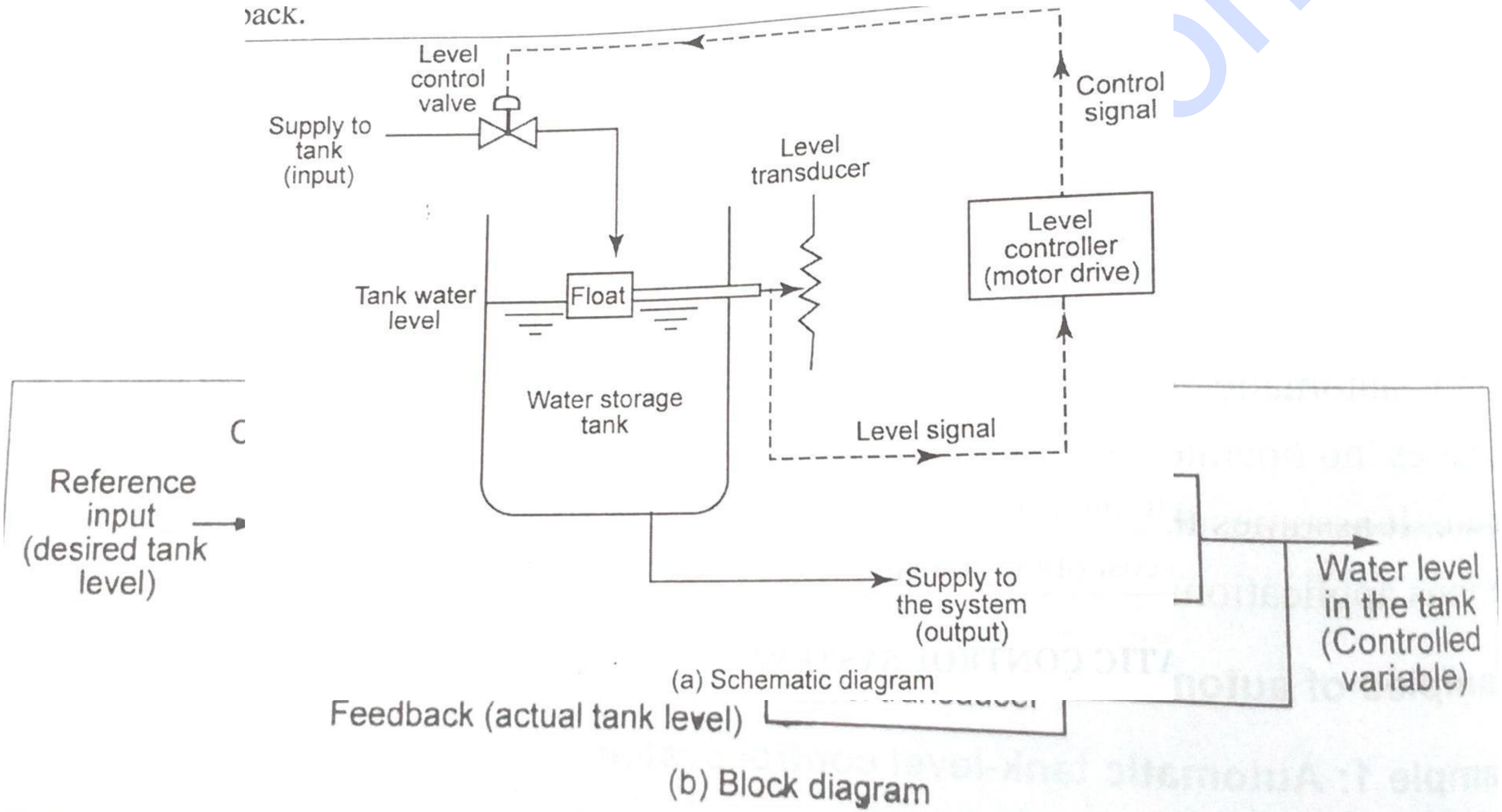
Open loop system

- It does not uses feedback
- It is less accurate
- It is simple in construction
- Presence of non-linearity causes malfunctioning
- The response is slow because manual control
- Easy maintenance because of no complex electronic circuit
- Cost is less

Closed loop system

- It uses feedback system
- It is more accurate
- It is complicated in construction
- It perform accurately even in presence of non-linearity
- It perform task faster than open loop
- It is difficult to maintain and repair
- Cost is more

Automatic tank level control system



(a) Schematic diagram

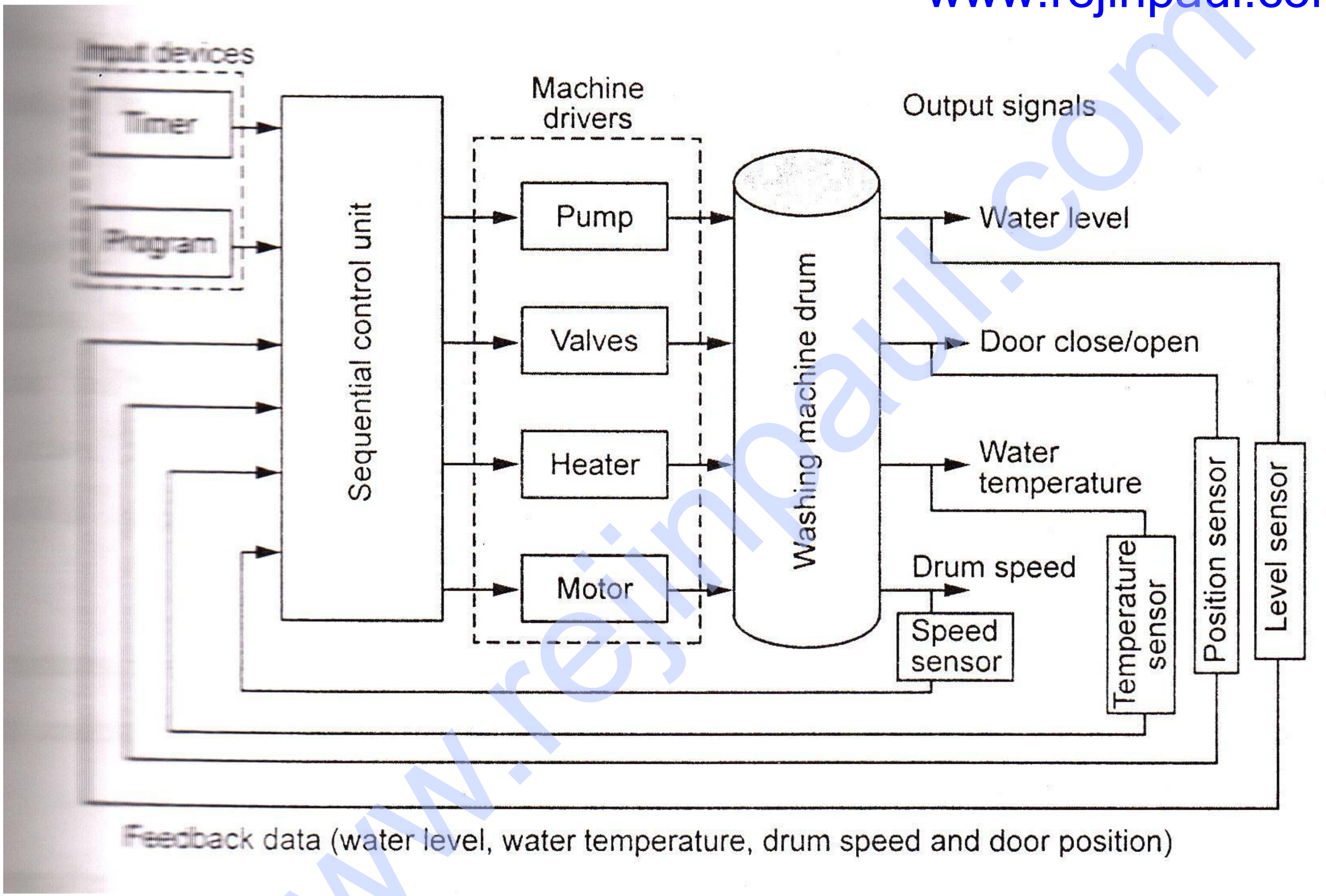
(b) Block diagram

Sequential controllers

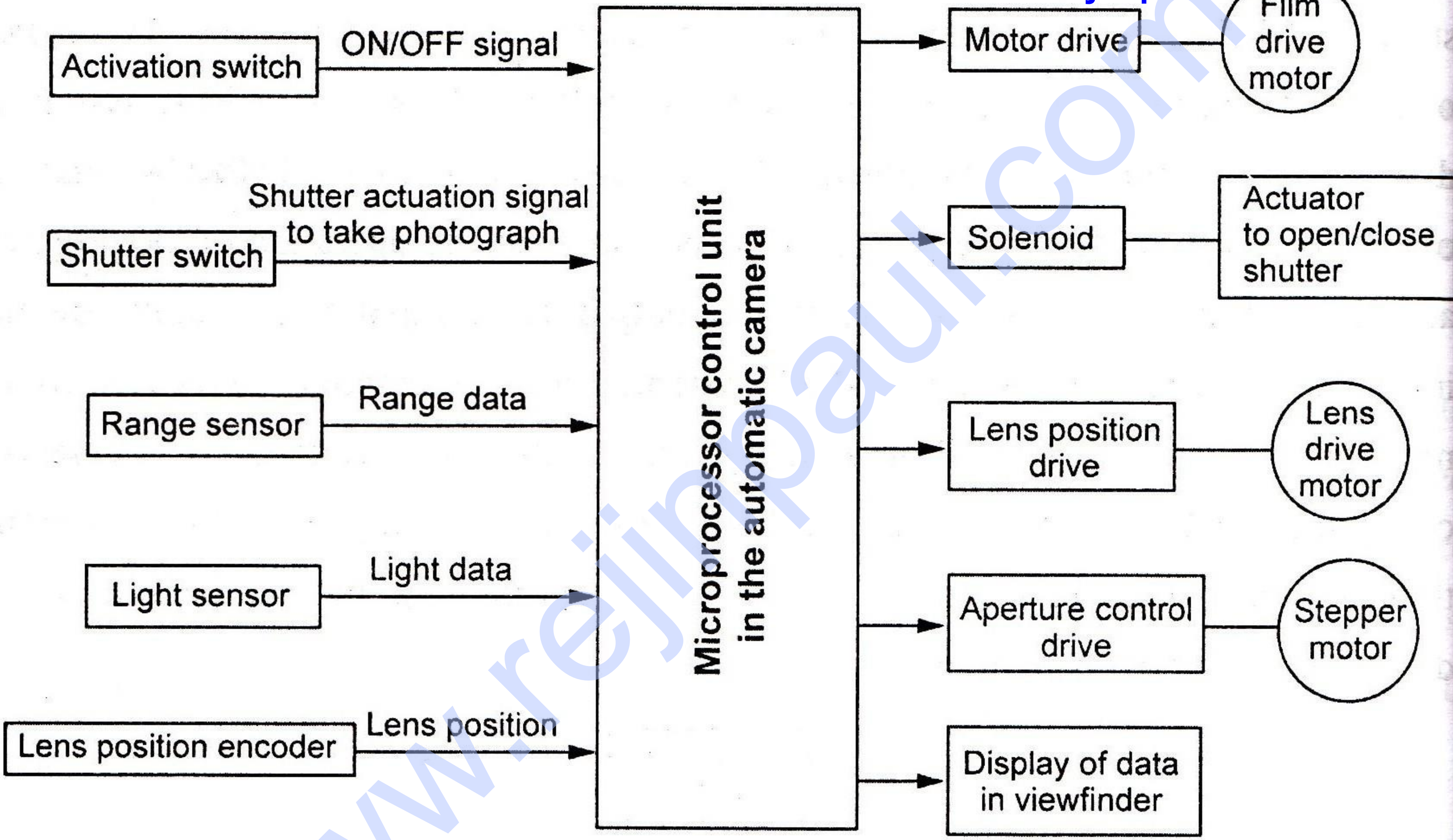
- A sequential control involve sequential execution of well defined operations.

The working of modern automatic washing machine is

- Step 1 : pre-wash cycle-(Cold water wash)
- Step 2: main wash cycle- (Hot water wash)
- Step 3: rinse cycle- (Number of preset time)
- Step 4: spin cycle- (Drain the water from clothes)

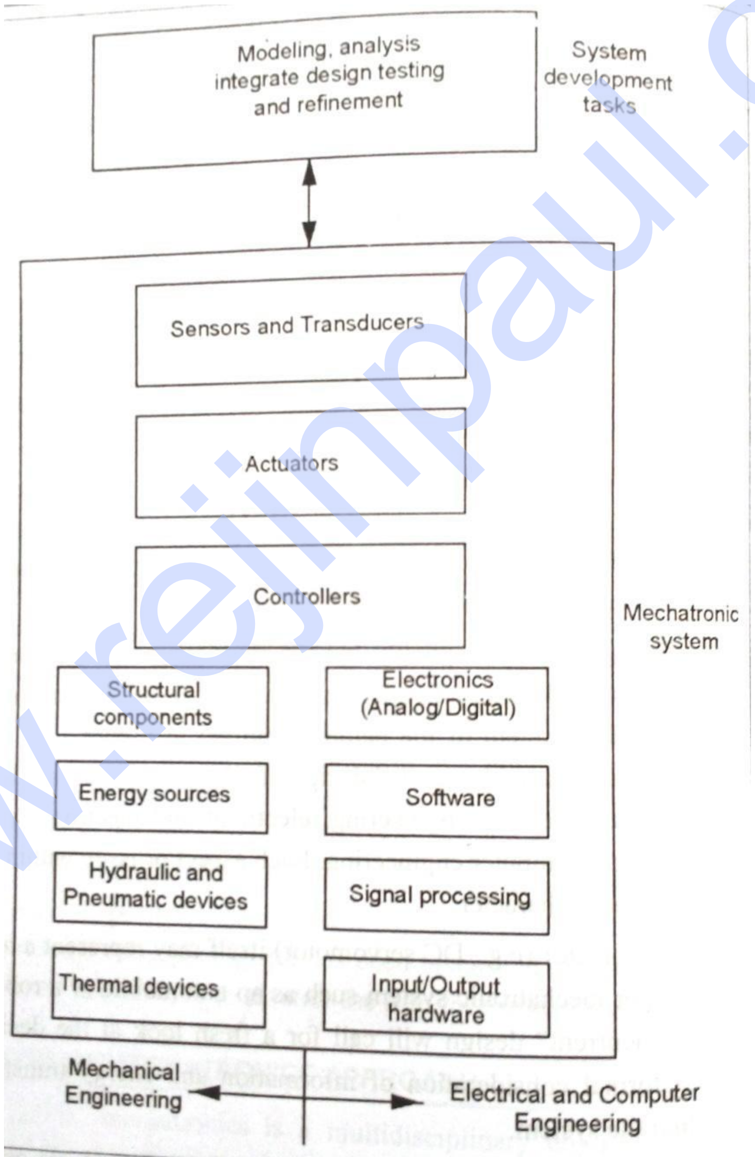


Block diagram of automatic washing machine system



Elements of control system for an automatic camera

Concept of mechatronic approach



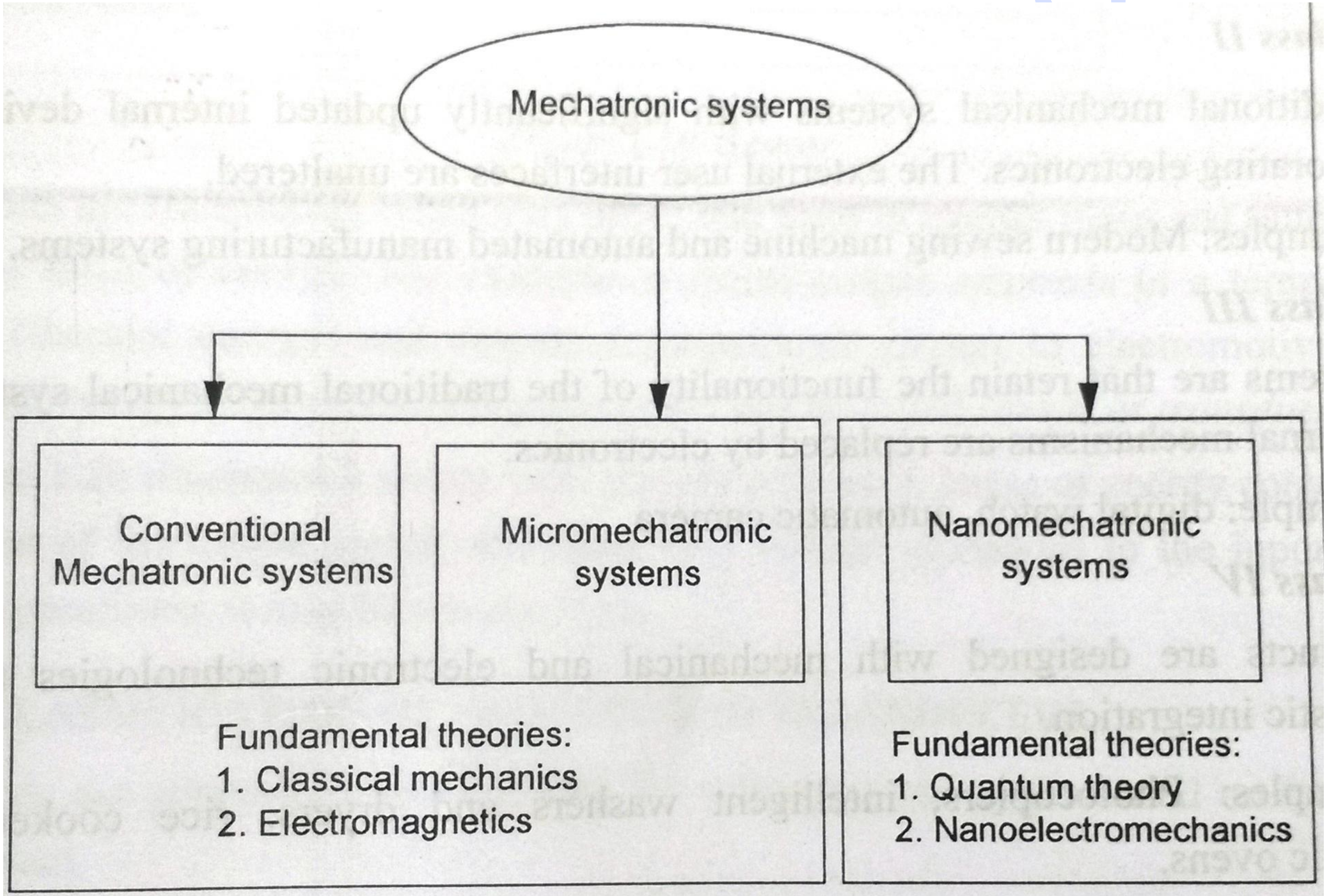
Emerging area of mechatronics

- Machine vision
- Automation and robotics
- Development of unmanned vehicles
- Design of subsystem for automotive engineering
- Sensing and control system
- Operation and maintenance of CNC machine
- Expert system and artificial intelligence
- Industrial electronics and consumer products
- Medical mechatronics and medical imaging systems
- Micro/nano mechatronics
- Computer integrated manufacturing (CIM) system

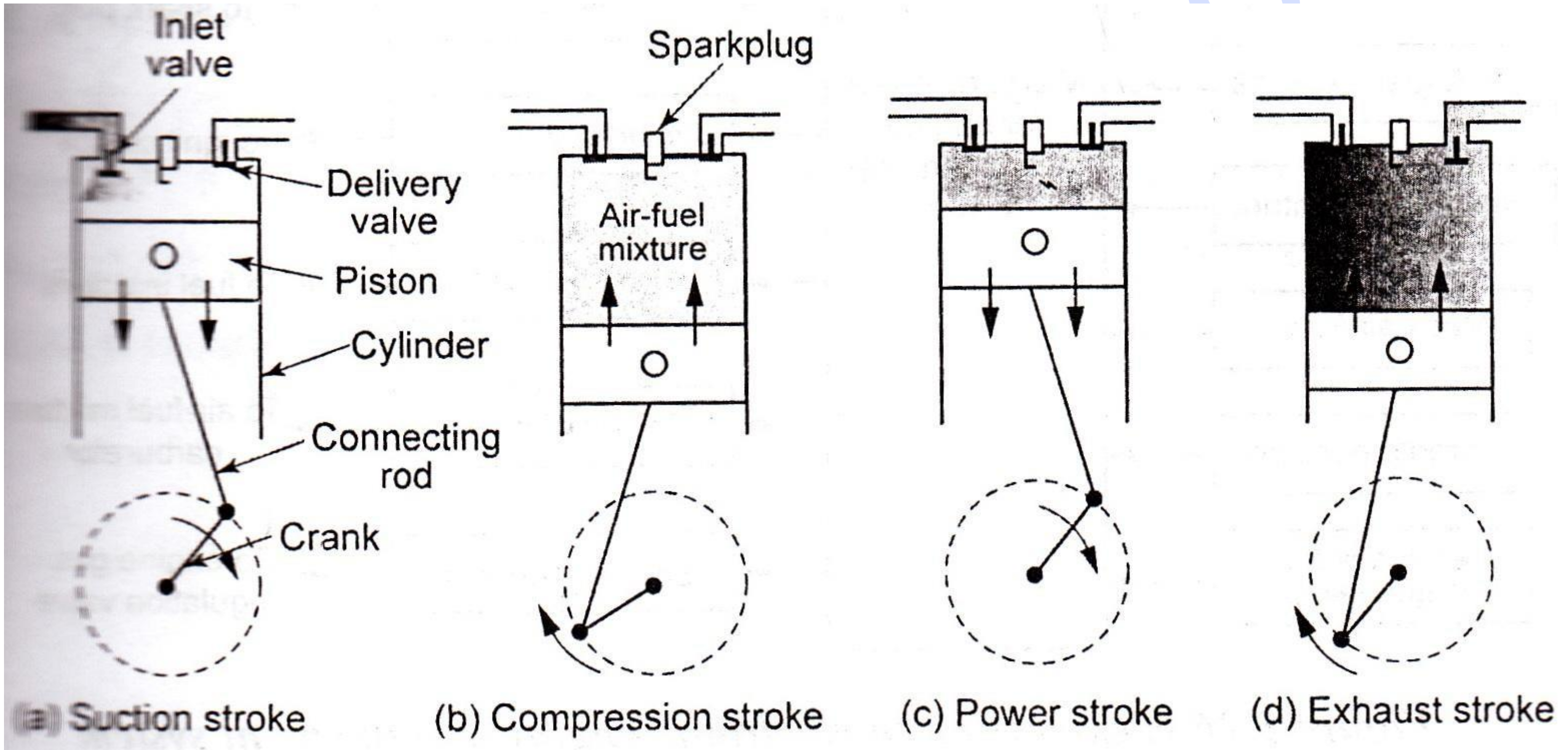
Need for mechatronics

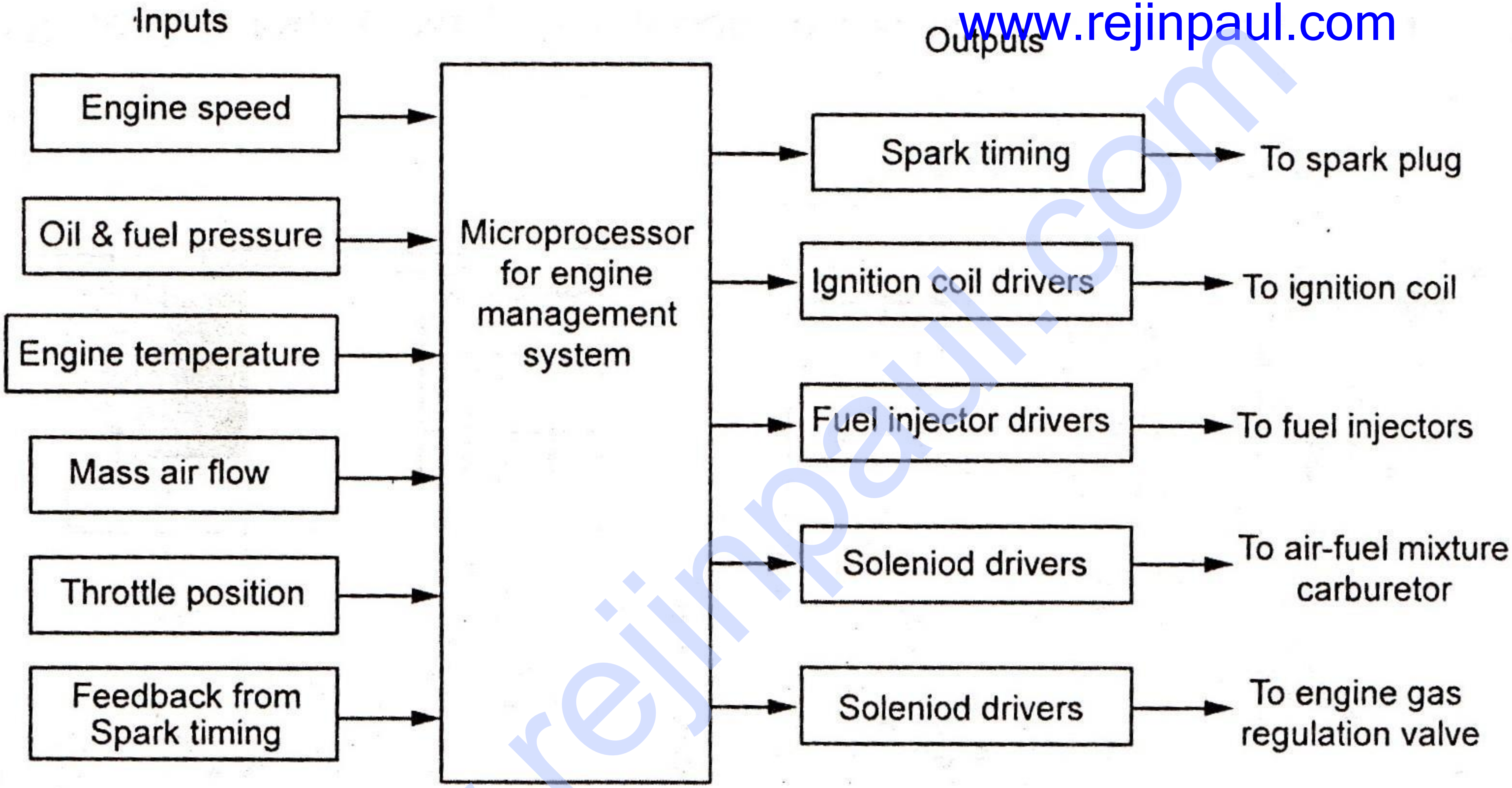
- Dynamic market conditions
- Producing next generation products
- Integration of modern technologies in products
- Variety in product ranges
- Batch production runs
- Change in design perspective
- Product quality and consistency
- Ease of reconfiguration of the process
- Demand for increased flexibility

Classification of Mechatronics



Working of a four stroke SI engine





Elements of control system for an electronic engine management system

UNIT II
8085 MICROPROCESSOR AND 8051 MICROCONTROLLER

MICROPROCESSOR

- It is a semiconductor component that incorporates the functions of a central processing unit (CPU) on a single integrated circuit (IC) . i.e., the central processing unit (CPU) built on a single IC is called microprocessor.

MICROPROCESSOR

- It is multipurpose, programmable and clock driven,
- Register based electronic device that reads binary instructions from a storage device called memory,
- Accept binary data as input, process the data according to the instruction and provides results as output.

Functional Block diagram of Microprocessor

Microprocessor

Functional Block diagram of Microprocessor

- ALU (**A**rithmetic and **L**ogic **U**nit)
  - It carries out arithmetic and logic operations on 8 bit word.
  - Arithmetic operation – addition, subtraction , multiplication , division etc.,
  - Logic operation - AND,OR,EX-OR
  - The content of accumulator and temporary register are the input to the ALU.
  - ALU output is stored in accumulator

- Register array

  - Register is a storage unit within the microprocessor used to store the data, address of instruction of any program.
  - Microprocessor contained 6 general purpose register it has 8- bit memory
  - Registers are B,C,D,E,H and L
  - To hold 16-bit data a combination of two 8-bit registers can be used.
  - The combination of two 8-bit registers is known as Register Pair (BC, DE and HL).
  - These Registers are used to store data temporarily during execution of the program.

- Control Unit
  - The timing and control unit acts as the brain of a computer.
  - It controls all operations of the CPU.
  - It controls input, output and all other devices connected to the CPU.

Evolution of Microprocessor

- First generation Microprocessor
  - 1st Microprocessor, Intel 4004, a 4 bit PMOS Microprocessor introduced in 1971 by the Intel corporation, USA.45 instructions and 2300 transistors
  - It has limited memory-simple aplications.
  - An enhanced version of Intel 4004 is Intel 4040.
  - e.g., Toshiba's 73472, Rockwell International's PPS-4 National IMP-4 etc.,

Evolution of Microprocessor

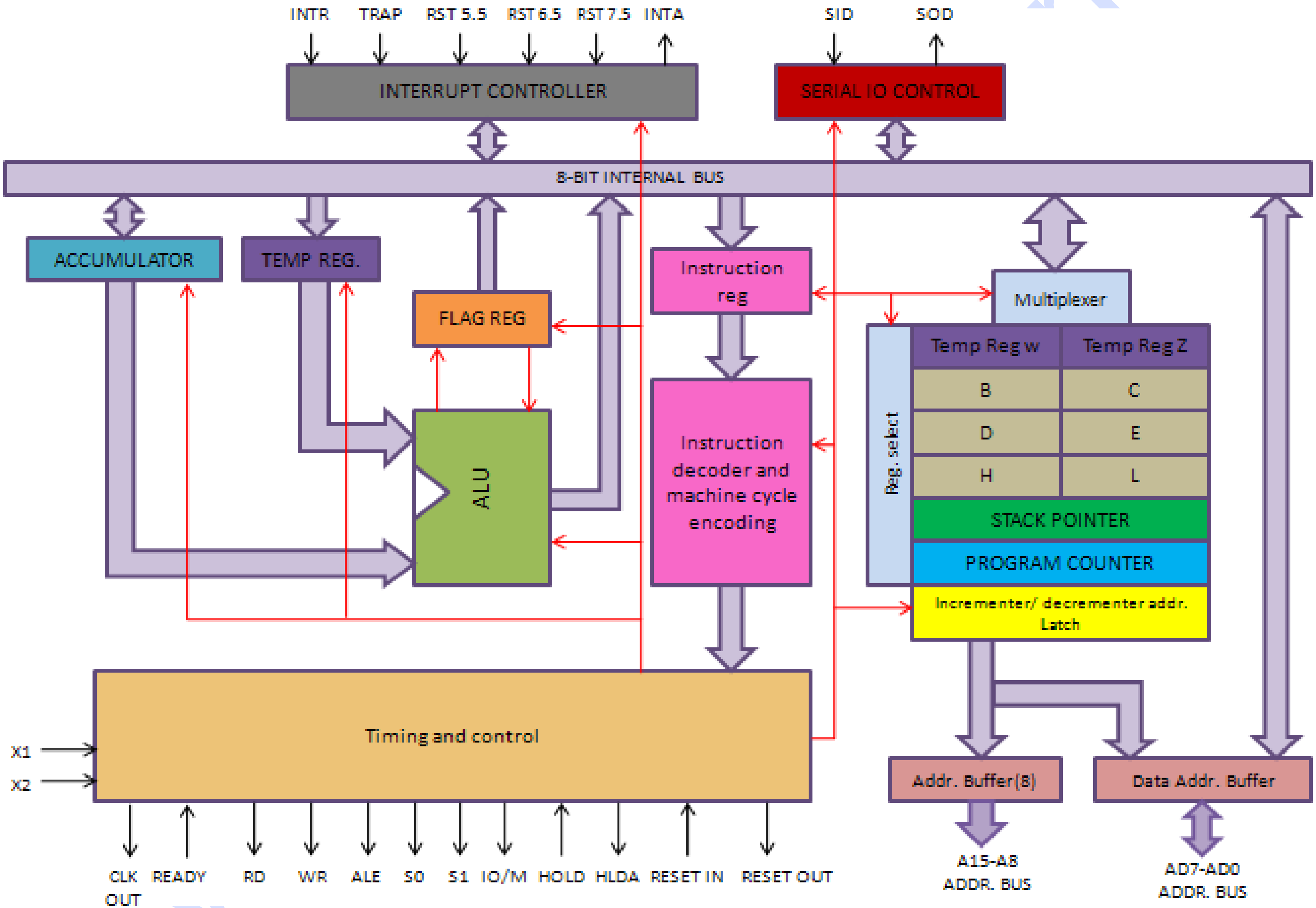
- Second generation Microprocessor
  - In 1972, Intel introduced 8- bit Microprocessor named as Intel 8008, which also uses PMOS technology.
  - But this technology was slow and not compatible with TTL logic
  - In 1973, Intel introduced more powerful and fast 8- bit NMOS Microprocessor called Intel 8080-3 power supplies.
  - 1975-Intel 8085 is the improved version of Intel 8080

- Third generation Microprocessor
  - In 1978 Intel introduced a 16- bit Microprocessor called Intel 8086.
  - Other 16- bit Microprocessor are Intel 80186, Intel 80286, zilog's z8000, Motorola's 68000, 68010 etc.,
- Forth generation Microprocessor
- In 1980-32bit-Iapx432-not popular
  - In 1985 Intel introduced a 32- bit Microprocessor called Intel 60386-desktop -386MP
- Fifth generation Microprocessor
  - Intel i860 is a 64 bit RISC microprocessor

Architecture of 8085

- Three main section
  - ALU
  - Timing and Control unit-
  - Set of register

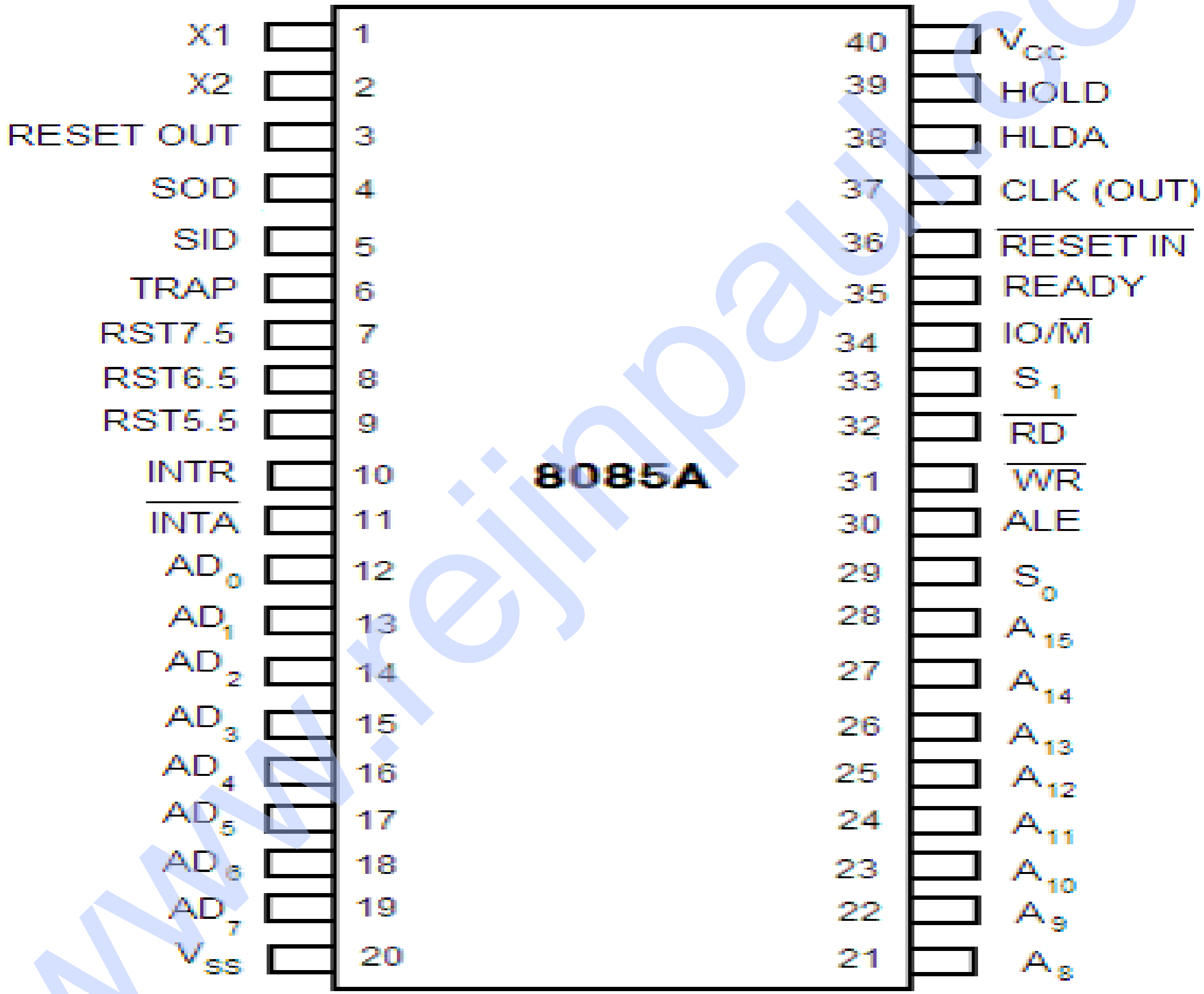
ARCHITECTURE OF 8085



- ALU
  - Addition, Subtraction, Logical AND,OR…etc
- Timing and Control Unit
  - timing and control signals –execution of instructions
  - Controls the entire operation of the microprocessor
- Register
  - 1- 8 bit Accumulator….i.e.-register A (ACC)
  - 6-8 bit general purpose register (B,C,D,E,H & L)
  - 1- 16 bit register –SP(Stack Pointer)
  - 1 -16 bit –PC (Program Counter)
  - Instruction register
  - Temporary register
  - Flag register

- Flag register
  - Carry flag (CY) – it is set, If carry or borrow occurs during the arithmetic operation.
  - Parity flag (P) – it is set, if the result has even number of it otherwise made 0.
  - Auxiliary carry flag (AC) – Binary coded decimal operations (BCD)
  - Zero flag (z) – is set if the result becomes 0
  - Sign flag (S) – is set if the result becomes –ve, if +ve, it is set to 0
  - 2 bit (don’t care)

Pin diagram


X1
X2
RESET OUT
SOD
SID
TRAP
RST7.5
RST6.5
RST5.5
INTR
INTA
AD0
AD1
AD2
AD3
AD4
AD5
AD6
AD7
VSS
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
8085A
40
39
38
37
36
35
34
33
32
31
30
29
28
27
26
25
24
23
22
21
VCC
HOLD
HLDA
CLK (OUT)
RESET IN
READY
IO/M
S1
RD
WR
ALE
S0
A15
A14
A13
A12
A11
A10
A9
A8

Signals in 8085

- **6 group of signals**
- **Address bus (A15-A8)-**
  - unidirectional
- **Data bus (AD7-AD0)**
  - bi-directional both data and address
- **Control and Status signals**
  - ALE (Address Latch Enable)
  - RD,WR,IO/M,S0,S1
- **Power supply and Clock frequency**
  - VCC +5
  - VSS-Ground
  - X1,X2
  - CLK

- **Externally initiated signals**
  - INTR
  - INTA
  - TRAP
  - RST 7.5,RST6.5,RST 5.5
  - READY
  - HOLD
  - RESET IN
  - RESET OUT
  - HLDA
- **Serial I/O Ports**
  - SID
  - SOD

ADRESSING MODES IN 8085

- Direct addressing
- Register addressing
- Register indirect addressing
- Immediate addressing
- Implicit addressing

- Direct addressing
  - LDA 240H (Load register A with the contents of memory location 240FH)
  - STA 2400H (Store the content of the accumulator in the memory location 2400H)
- Register addressing
  - MOV B, D (move the content of register D to register B)
  - INX H (increment the content of [H-L] register pair

- Register indirect addressing
  - LXI H, 2500H (Load H-L pair with 2500H)
  - MOV A, B (move the content of the memory location, whose address is in H-L pair(H-L Pair) to accumulator)
  - HLT (halt)
- Immediate addressing
  - MVI A, 05 (Move 05 in register A)
  - 3E, 05 (the code format of an instruction)

- Implicit addressing
  - There are certain instruction which operate the content of the accumulator.
  - Such instruction do not require the address of the operand
  - CMA
  - RAL
  - RAR

Instruction sets 8085

- Data transfer group
- Arithmetic group
- Logical group
- Branch group
- Stack, I/O and Machine control group

Data transfer group

- MOV r1,r2
- MOV r, M (Move the content of memory to register)
- MOV M, r
- MVI r1, data (Move Immediate DATA to register)
- MVI M, data
- LDA data (Load accumulator direct)
- STA addr (store accumulator direct)
- XCHG (exchange the content of H-L with D-E pair)

- LHLD addr (Load HL pair direct)
- SHLD addr (Store HL pair direct)
- STAX xp (store accumulator Indirect)

Arithmetic group

- ADD r
- ADD M
- ADI data
- ADC r
- ADC M
- SUB r
- SUB M
- SUI data
- SBB r
- SBB M

- INR r
- INR M
- DCR r
- DCR M

Logical group

- ANA r
- ANA M
- ANI data
- ORA r
- ORA M
- ORI data
- XRA r
- XRA m
- XRI data

- CMA (complement acc)
- CMC(complement carry)
- CMP r (compare)
- CMP M
- CPI data
- RLC (rotate)
- RRC
- RAL
- RAR

Branch group

- Two branch instruction
  - Conditional
    - The conditional branch instructions transfer the program to the specified label when certain condition is satisfied
  - Unconditional
    - The Unconditional branch instructions transfer the program to the specified label when certain condition is not satisfied

- **Conditional jumb addr (label)**
  - If the condition is true and the program jumps to the specified label, the execution of a conditional jump takes 3 machine cycles and 10 states
  - If the condition is not true, only two machine cycles and 7 states are required for the execution of the instruction.

- JZ addr (label) [jump if the result is zero]

[PC] ← address (label), jump if z=0

Machine cycle – 2/3

States – 7/10

Addressing mode – Immediate

Flags - None

- JNZ addr [jump if the result is not zero]

[PC] ← address (label), jump if z=1

- JC addr [jump if there is a carry]

[PC] address (label), jump if CS = 1

- JNC addr [jump if there is no carry]

[PC] ← address (label), jump if CS = 0

←

- JP addr [jump if the result is plus)

[PC] ← address (label), jump if S = 0

- JM addr [jump if the result is minus)

[PC] ← address (label), jump if S = 1

- JPE addr [jump if even parity)

[PC] ← address (label), jump if P = 0

- JPE addr [jump if odd parity)

[PC] ← address (label), jump if P = 1

- CALL addr (label)
  - Call the subroutine identified by the operand
  - CC addr (call subroutine if carry status CS=1)
  - CNC addr (call subroutine if carry status CS=0)
  - CZ addr (call subroutine if result is zero)
  - CNZ addr (call subroutine if result is not zero)
  - CP addr (call subroutine if result is plus)
  - CM addr (call subroutine if result is minus)
  - CPE addr (call subroutine if even parity)
  - CPOE addr (call subroutine if odd parity)

- Unconditional
  - RET(Return from Subroutine)
  - CALL addr
  - RSTn (Restart)

Stack ,I/O and Machine control Group

- PUSH rp [push the content of register pair to stack)
- PUSH PSW [push the program status to word]
- POP rp [pop the content of register pair which was saved from the stack]
- POP PSW
- IN PORT
- OUT PORT
- EI (enable interrupts)

- DI(disable interrupts)
- HLT (halt)
- NOP(notion oper)
- RIM(read interrupts mask)
- SIM (set interrupts mask)

SIM(Set Interrupts Mask)

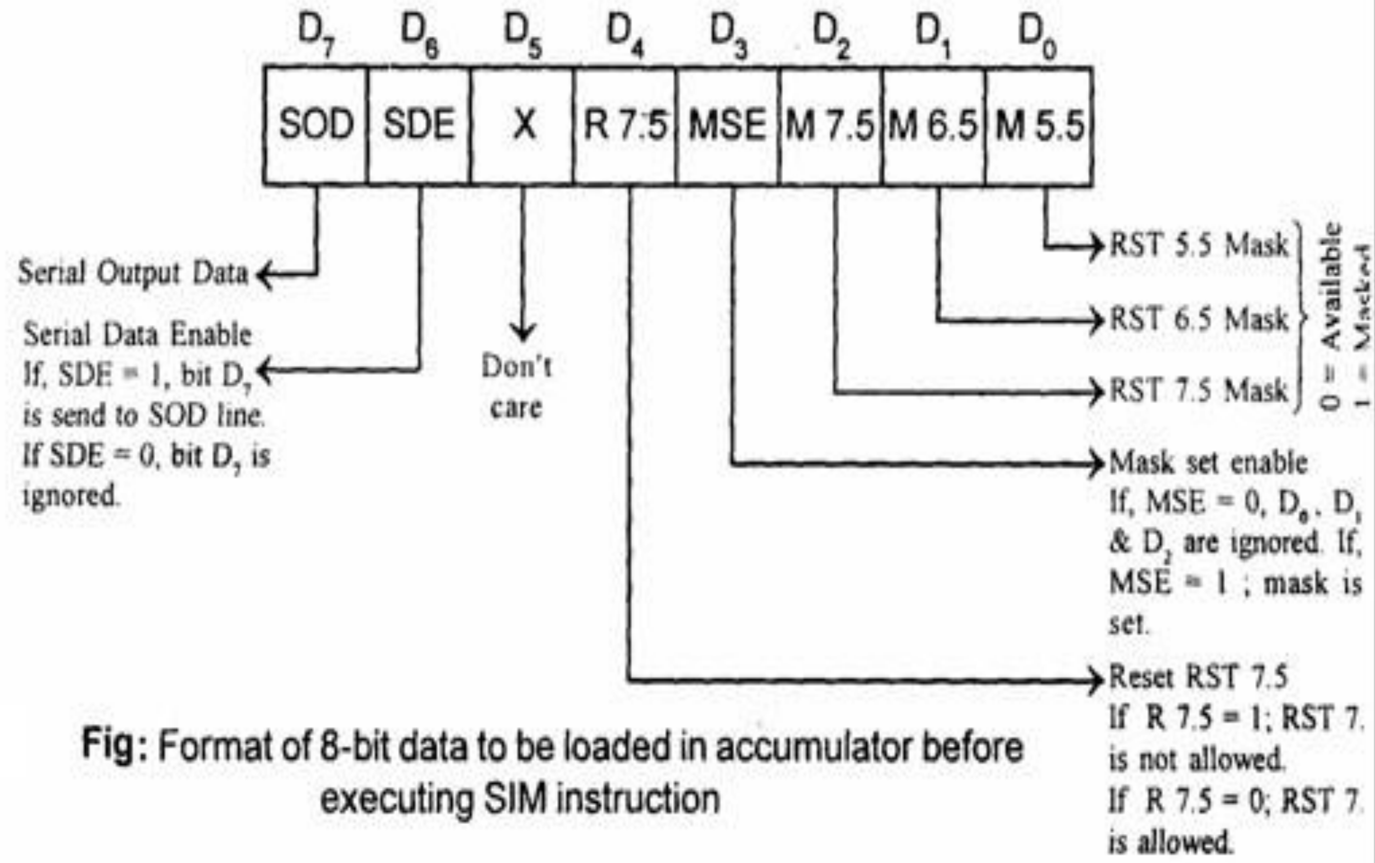
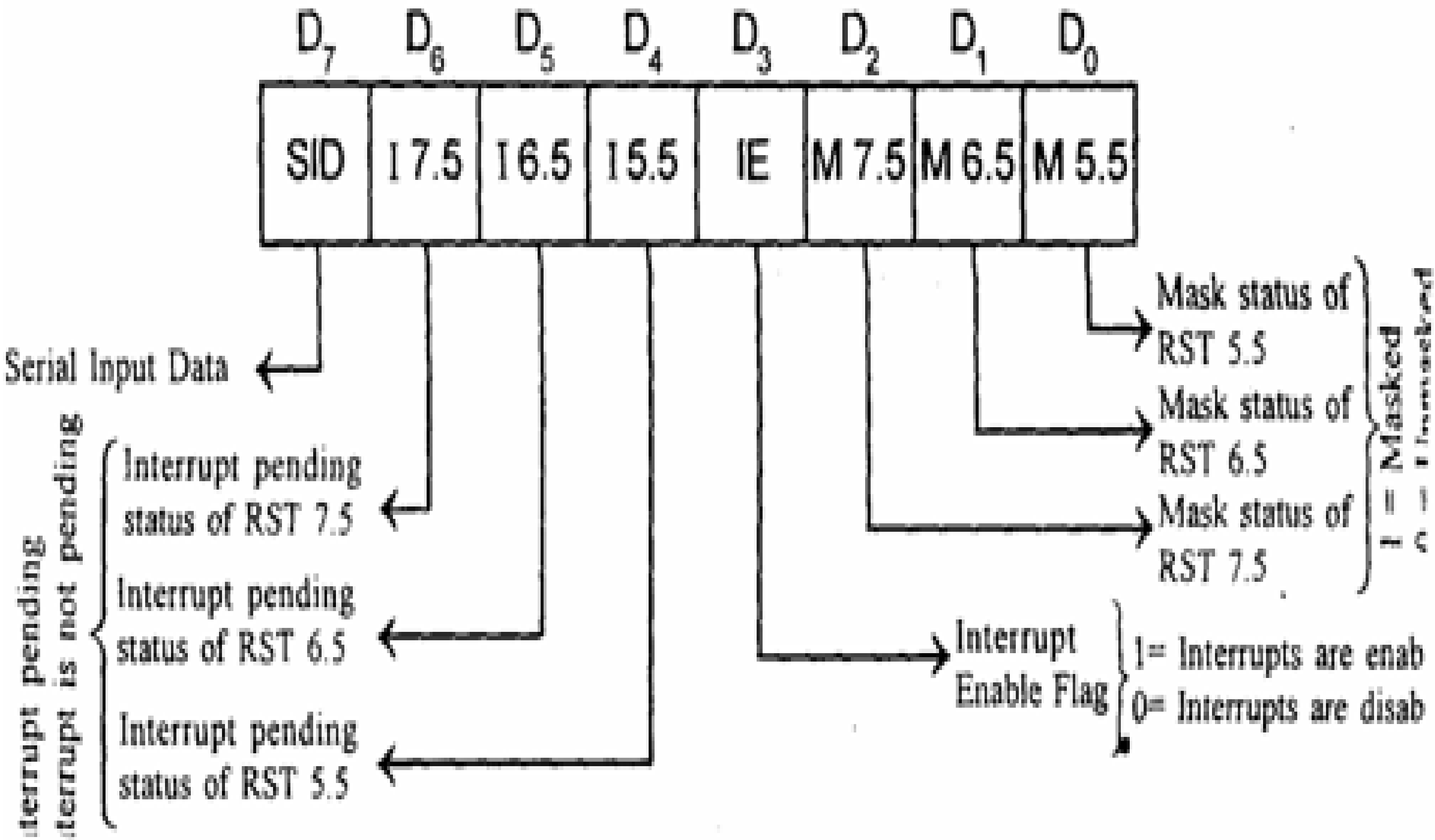


Fig: Format of 8-bit data to be loaded in accumulator before executing SIM instruction

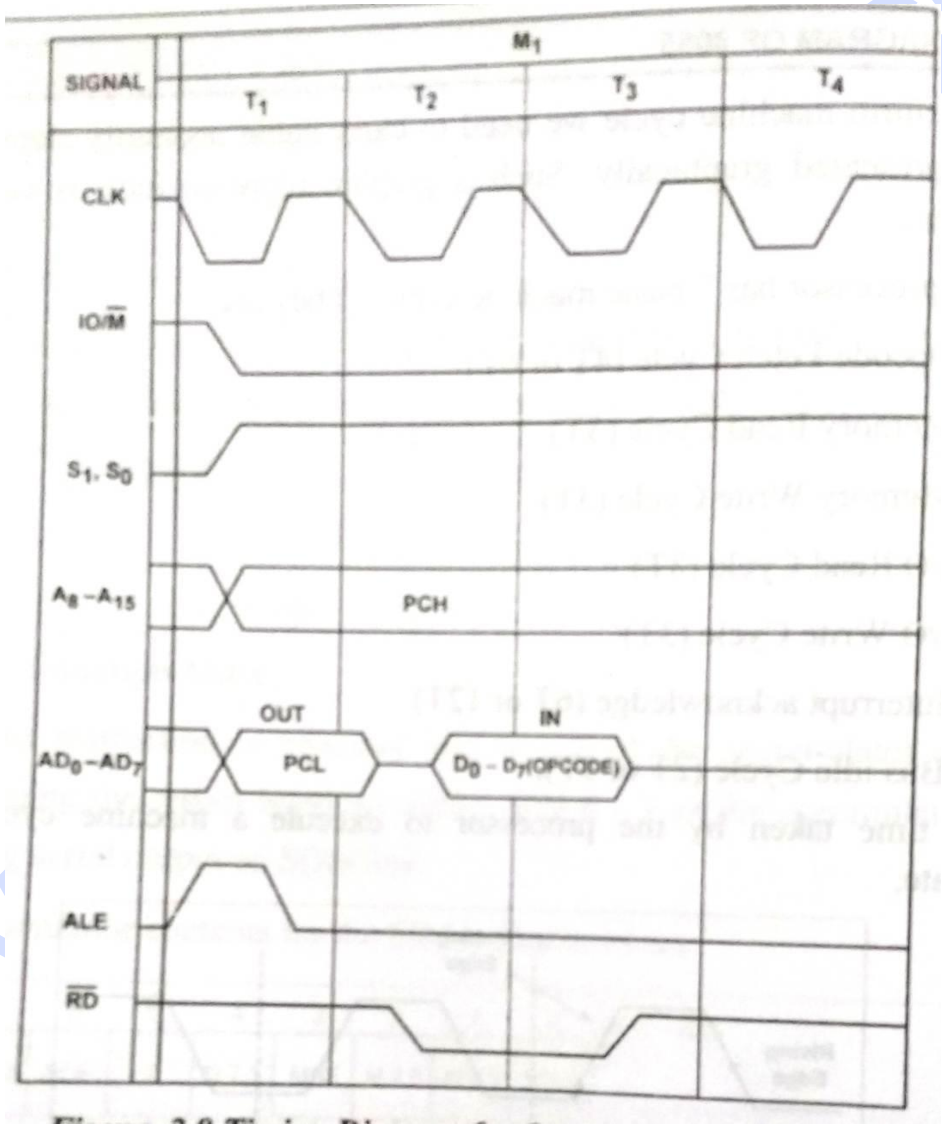
RIM (Read Interrupts Mask)



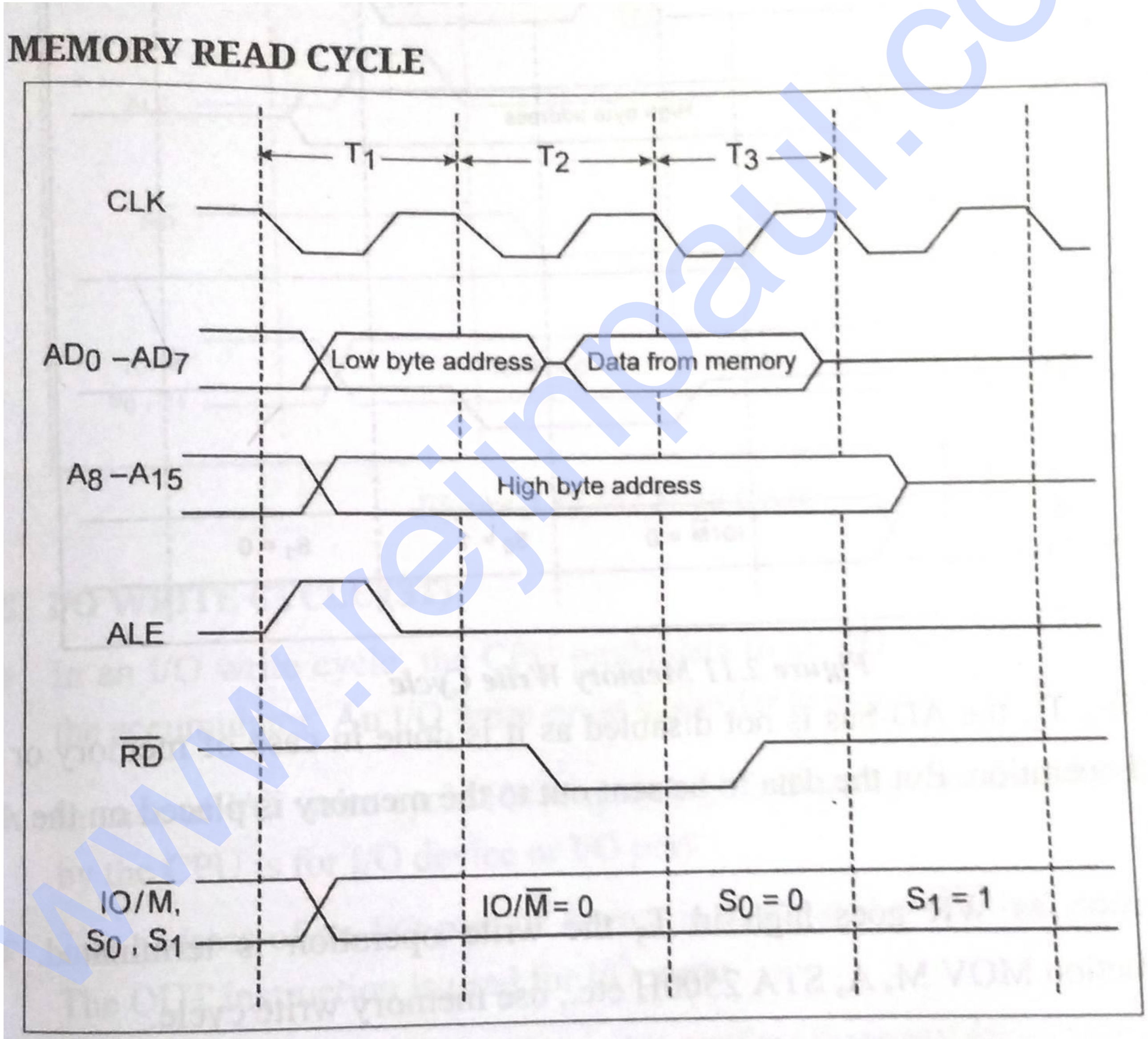
Timing diagram of 8085

- Opcode fetch cycle (4T or 6T)
- Memory Read cycle (3T)
- Memory write cycle (3T)
- I/O read cycle (3T)
- I/O write cycle (3T)
- Interrupt acknowledge (6T or 12T)
- Bus idle cycle (2T or 3T)

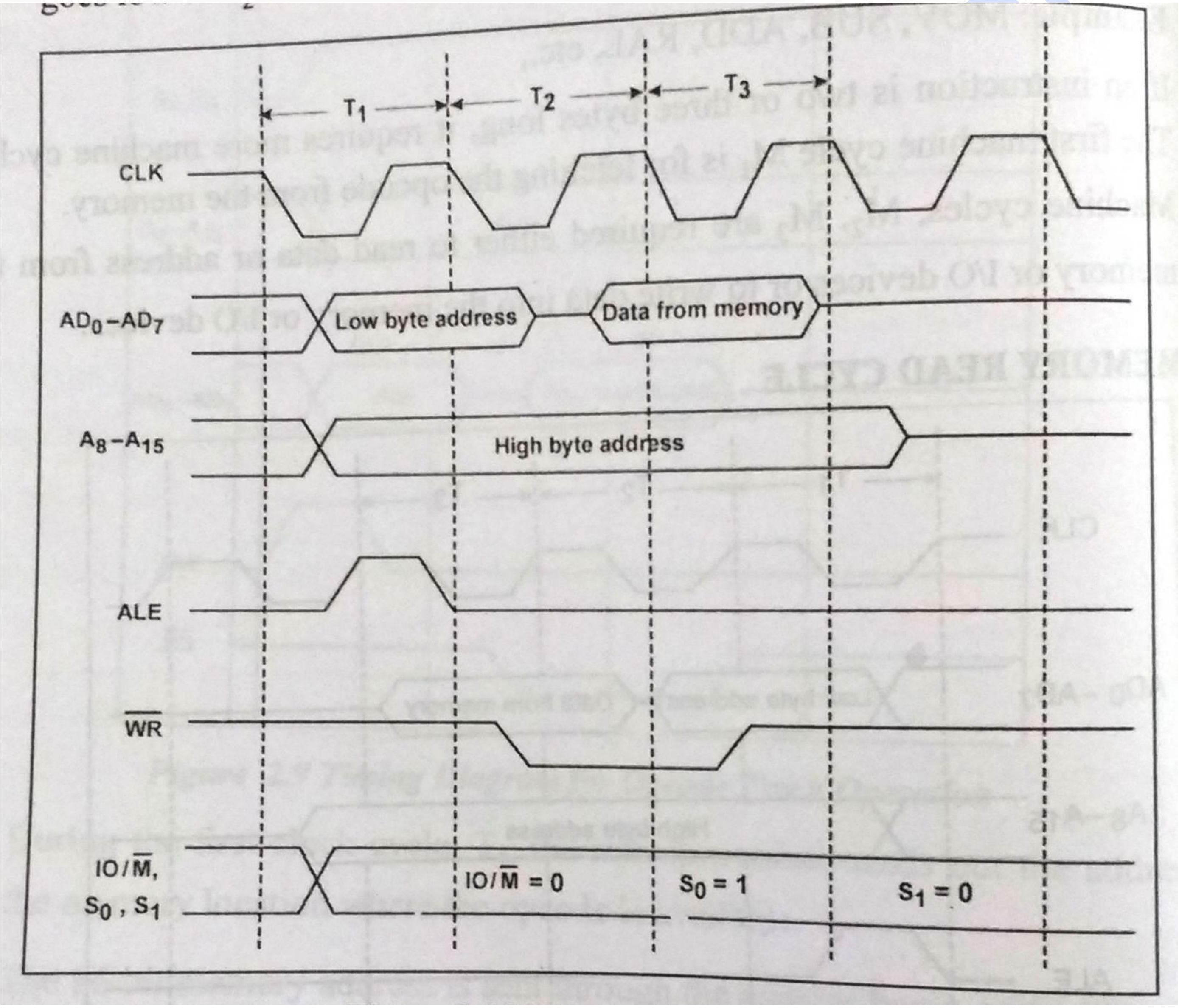
Opcode fetch cycle



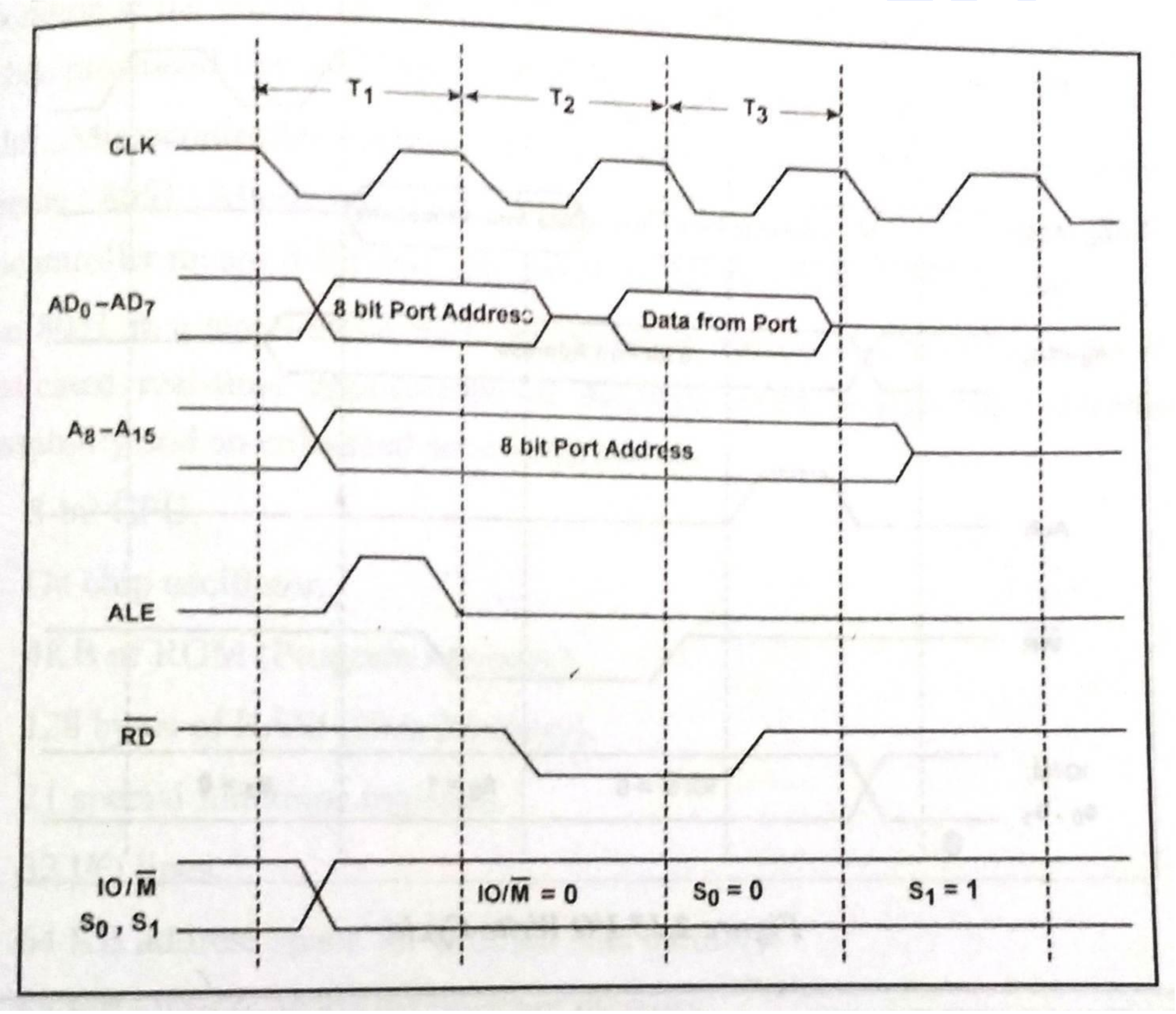
Memory Read cycle



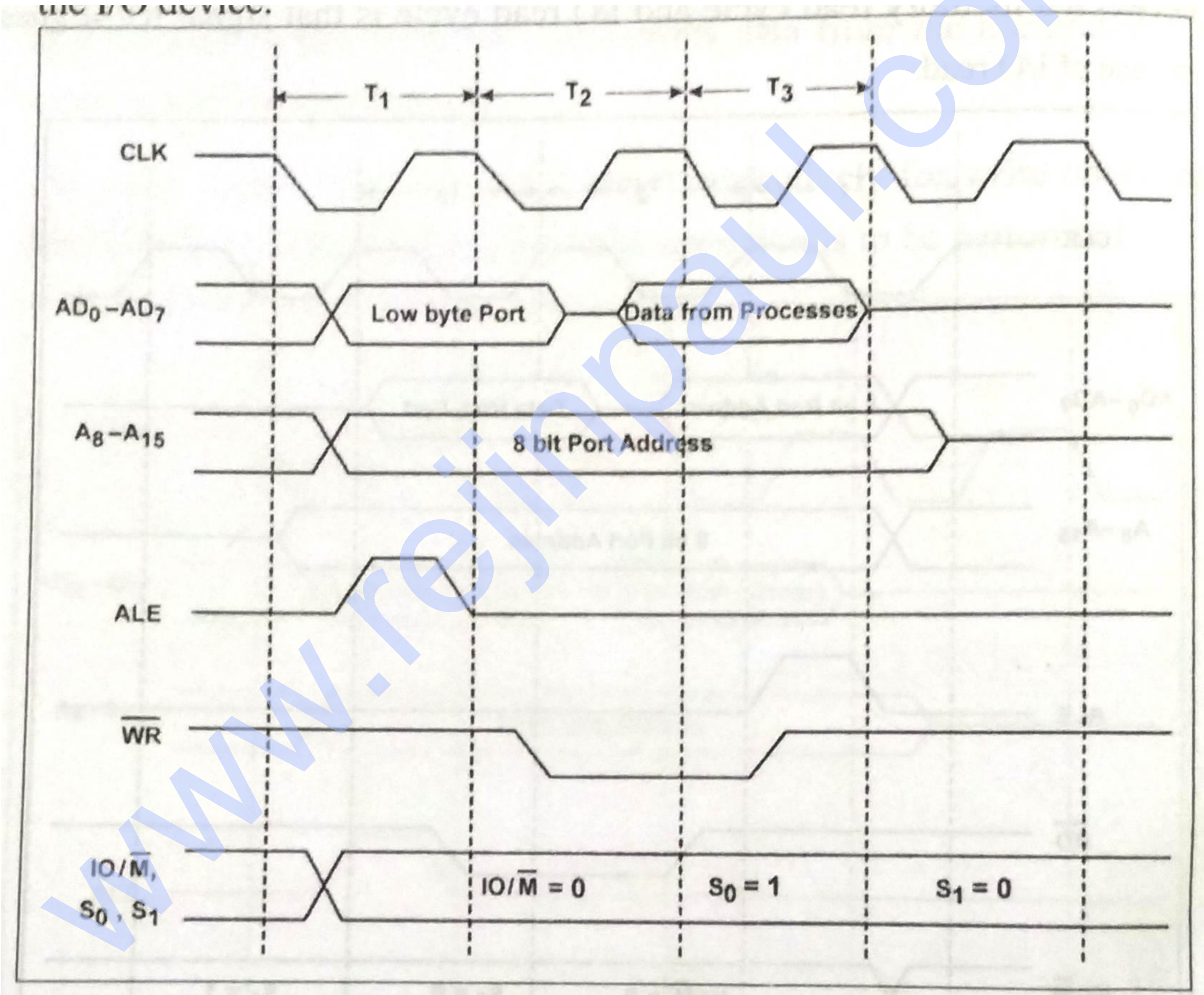
Memory write cycle



I/O read cycle



I/O write cycle



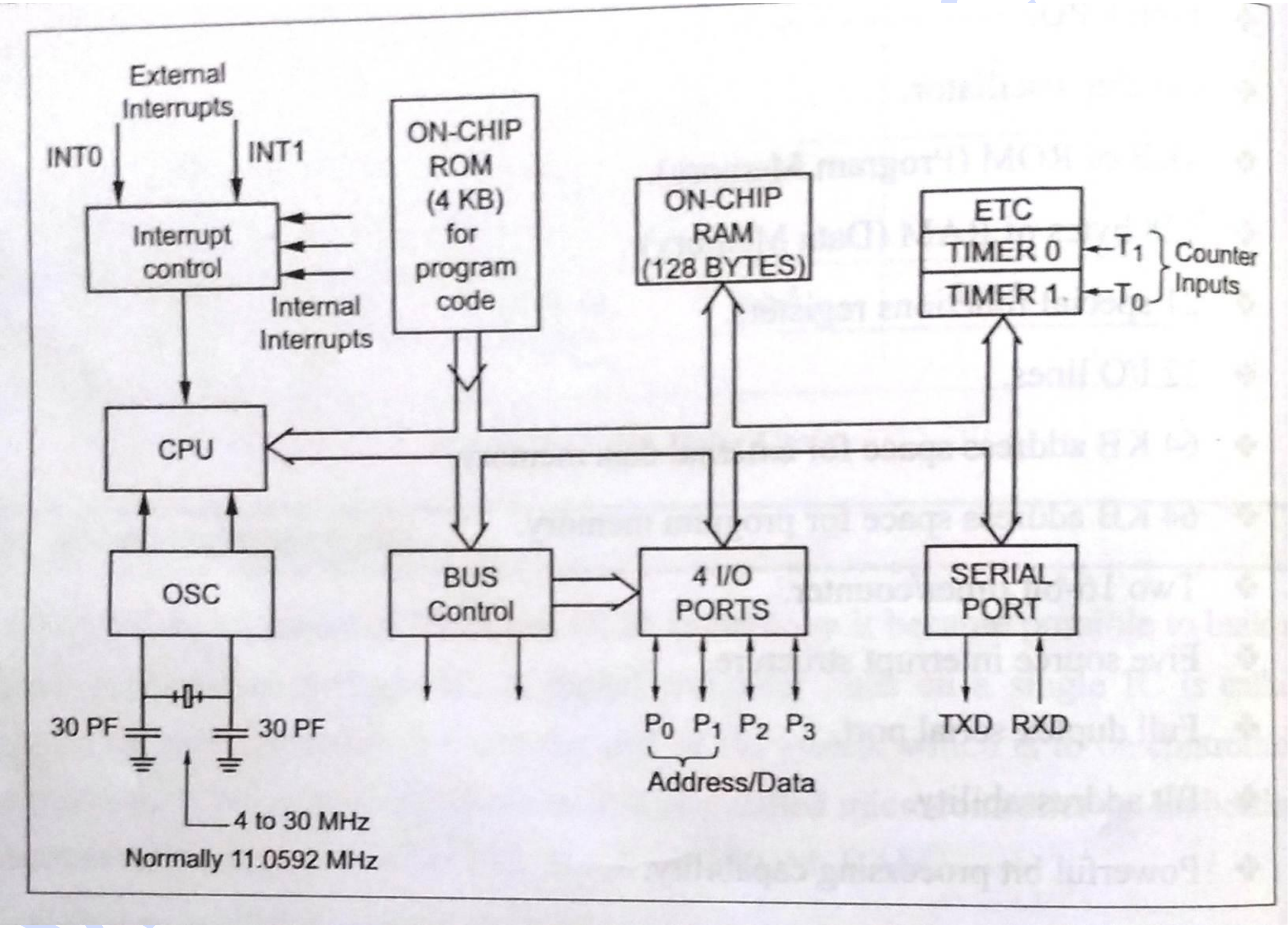
Microcontroller

- A Microcontroller is a small computer on a single integrated circuit containing a processor core, memory and programmable input/output peripherals.

Features Microcontroller

- 8 bit CPU
- On chip oscillator
- 4Kb of ROM
- 128 bytes of RAM
- 21 special functions register
- 32 I/O lines
- 64 KB address space for external data memory
- 64 KB address space for program memory
- 2 16-bit timer/counter

Block diagram of 8051



UNIT 3

PROGRAMMABLE PERIPERAL INTERFACE

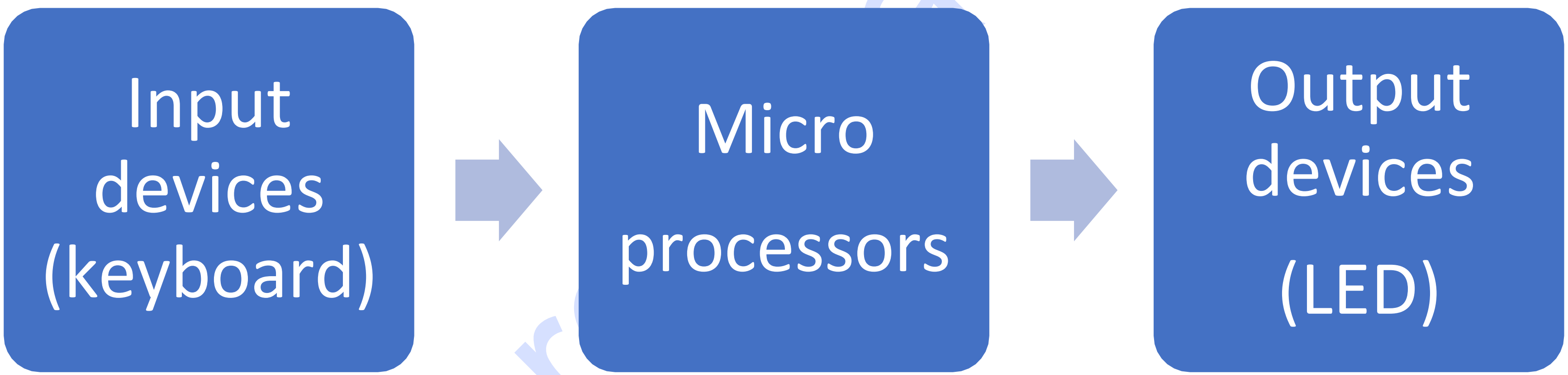
Content

- Introduction
- Architecture of 8255
- Keyboard interfacing
- LED display –interfacing
- ADC and DAC interface
- Temperature Control
- Stepper Motor Control
- Traffic Control interface

Introduction

- To communicate with the outside world, microprocessor use peripherals (I/O devices)
- Input devices – Keyboards, A/D converters etc.,
- Output devices – CRT, Printers, LEDs etc.,
- Peripherals are connected to the microprocessors through electronic circuit known as interfacing circuits.

Microprocessors unit with I/O devices



Input peripherals

Output peripherals

- Some of the general purpose interfacing devices
  - I/O ports
  - Programmable peripherals interface (PPI)
  - DMA controllers
  - Interrupt controller
- Some of the special purpose interfacing devices
  - CRT controller
  - Keyboard
  - Display
  - Floppy Disc controllers

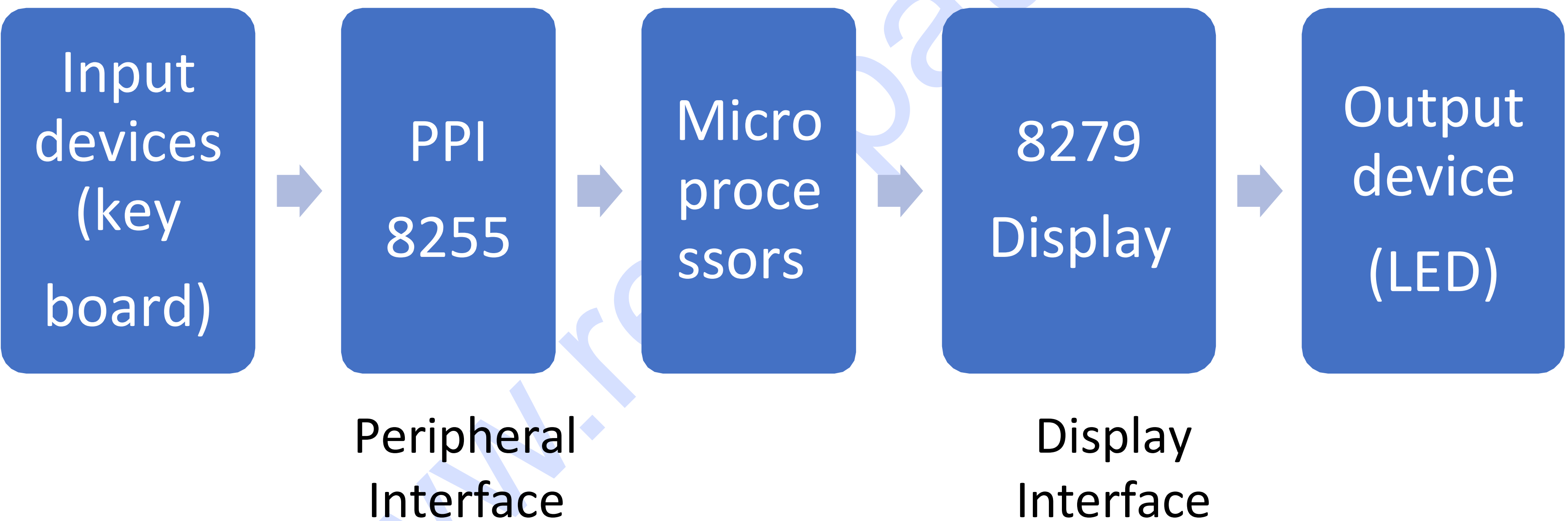
Peripheral interfacing Chips are used generation of I/O ports

- Programmable peripherals interface Inter 8255 (PPI)
- Programmable Interrupt controller (PIC) Intel 8259
- Programmable communication interface (PCI) Intel 8251
- Keyboard display Controller Intel 8279
- Programmable counter /Inverter timer Intel 8253
- A/D and D/A Converter Interfacing

Advantages

- To perform specific functions by giving a control word to the internal register.
- Control word –instructions informs the peripheral about various functions to perform.
- Format (CW) specified by the manufacturer.

Microprocessors unit with I/O devices



Address Space Partitioning

- The Microprocessors uses 16 bit wide address bus for addressing memories and I/O devices.
- Using 16 bit wide address bus, it can access 2^{16} = 64k bytes of memory and I/O devices
- Two schemes for the allocation of addresses to memories and I/O devices
  - Memory mapped I/O
  - I/O mapped I/O

Memory mapped I/O

- It has only one address space
- Address space is defined as the set of all possible addresses that a microprocessor can generate
- Some addresses assigned to memories and Some addresses to I/O devices
- Memory locations are assigned with addresses from 8000 to 84FF
- I/O devices are assigned with addresses from 8500 to 85FF

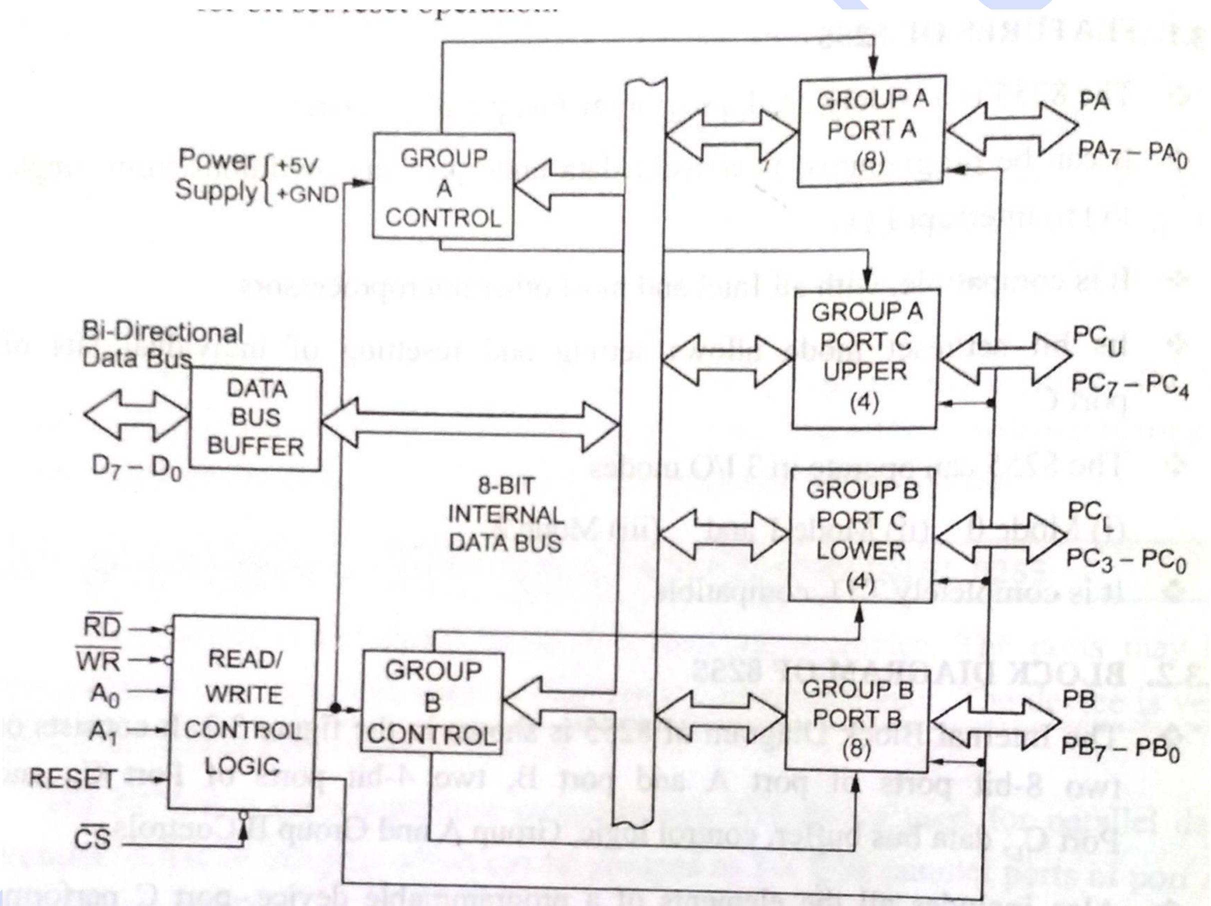
I/O mapped I/O scheme

- In this scheme, addresses assigned to memories locations can also be assigned to I/O devices
- Since the same address may be assigned to memories locations or an I/O devices
- The microprocessor has a signal to distinguish whether the address on the address bus is for memories locations or an I/O devices

I/O mapped I/O scheme

- When signal is high, then address on the address bus is for an I/O devices
- When signal is low, then address on the address bus is for memory locations
- Two extra instruction IN and OUT are used to address I/O devices.
- The IN instruction is used to read the data of an input devices.
- The OUT instruction is used to send the data of an input devices.
- This scheme is suitable for a large system.

PROGRAMMABLE PERIPHERALS INTERFACE INTER 8255 (PPI)

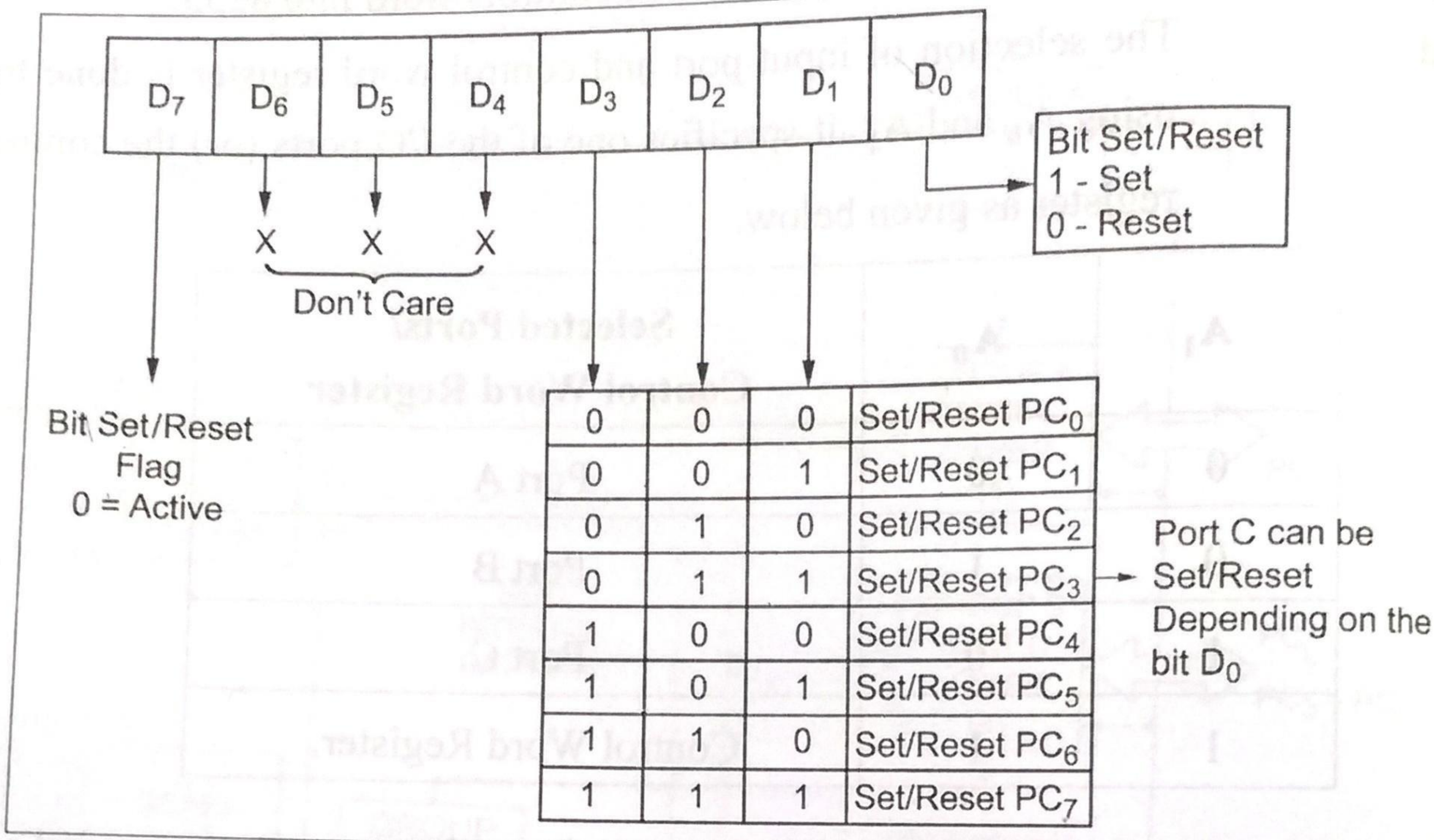


Operating mode of 8255

- Bit Set Reset (BSR) Mode
- I/O Mode

Bit Set Reset (BSR) Mode

BSR Control Word Format



BSR control word format

I/O Mode

- The 8255 has the following 3 modes of operation
  - Mode 0 – Simple Input/output
  - Mode 1 – Input / Output with the Handshake or strobed
  - Mode 2 – Bi-directional I/O

I/O Mode

Mode 0 – Simple Input/output

- Port A and port B are used as two simple 8-bit I/O port
- Port C as two 4-bit port

- Features
  - Outputs are latched
  - Inputs are buffered not latched
  - Ports do not have handshake or interrupt capability

I/O Mode

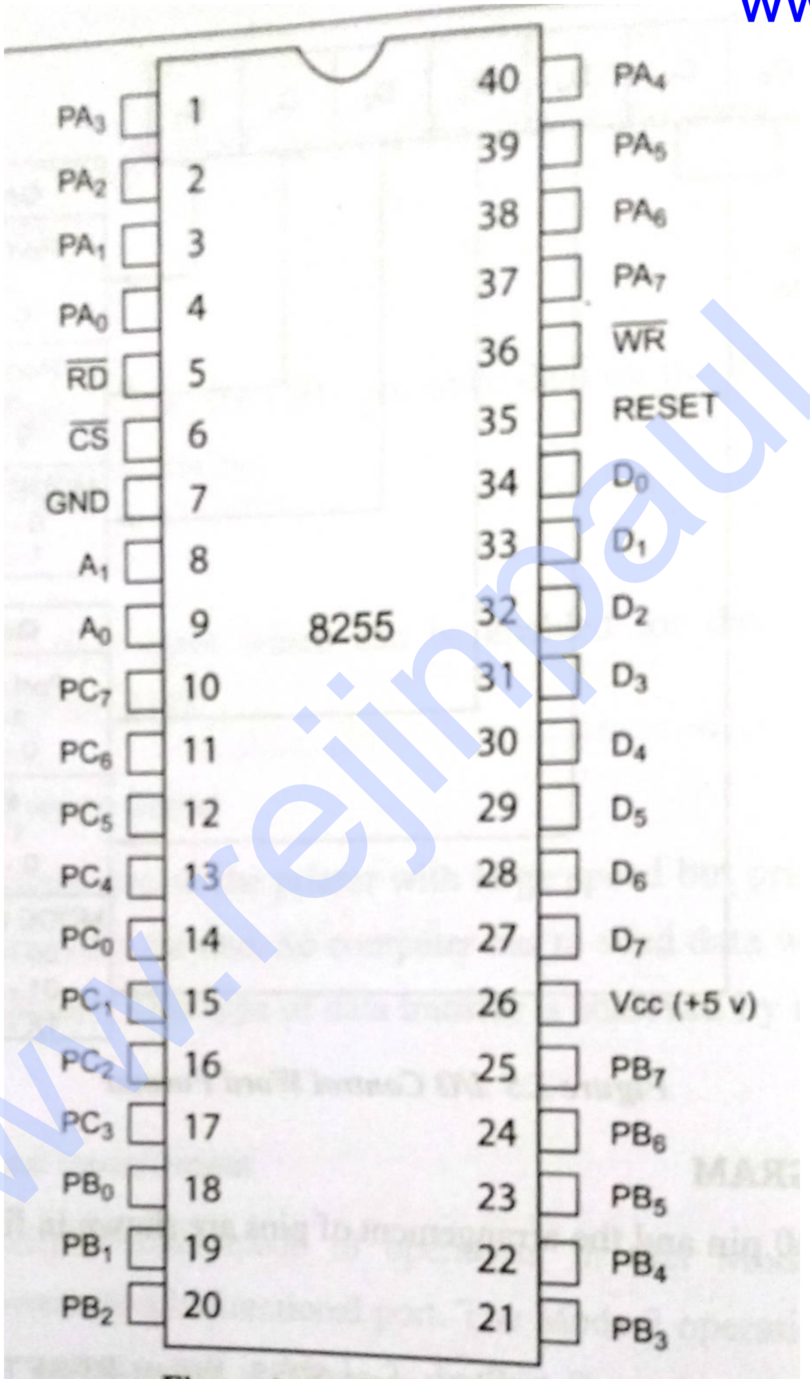
- Mode 1 – Input / Output with the Handshake
  - Input or output data transfer is controlled by handshaking signals.
  - Handshaking signals are used to transfer data between devices whose data transfer speeds are not same.
  - Port A and Port B are designed to operate with the Port C.
  - When Port A and Port B are programmed in Mode 1, 6 pins of port C is used for their control.

I/O Mode

- D0-D7 data bus
  - bi directional, tri state data bus line
  - It is used to transfer data and control word from 8085 to 8255
- RD (Read)
  - When this pin is low, the CPU can read data in the port or status word through the data buffer
- WR (write)
  - When this pin is low, the CPU can write data in the port or in the control register through the data buffer

I/O Mode

- Mode 2 – Bi-directional I/O
- Port A can be programmed to operate as a bidirectional port.
- The mode 2 operation is only for port A
- When port A is programmed in Mode 2, the Port B can be used in either Mode 1 or Mode 0.
- Mode 2 operation the port a is controlled by PC_3 to PC_7 of port C.



PIN DIAGRAM OF 8255

Pin diagram of 8255

D7 – D0	**Data Bus**
PA7 – PA0	**Port A**
PB7 – PB0	**Port B**
PC7 – PC0	**Port C**
CS	**Chip Select**
A0, A1	**Address bits**
RD	**Read Input**
WR	**Write Input**
RESET	**Reset Input**
Vcc	**+5V**
GND	**0 Volts**

PROGRAMMING and OPERATION of 8255

- Programming in MODE 0
- D7 –set to 1
- D6,D5,D2- all set to 0 –MODE 0
- D4,D3,D1 and D0- determine weather the corresponding ports are to configured as input or output

A		B		GROUP A		GROUP B	
D4	D3	D1	D0	PORT A	PORTC U	PORT B	PORT C L
0	0	0	0	OUT	OUT	OUT	OUT
0	0	0	1	OUT	OUT	OUT	IP
0	0	1	0	OUT	OUT	IP	OUT
0	0	1	1	OUT	OUT	IP	IP
0	1	0	0	OUT	IP	OUT	OUT
0	1	0	1	OUT	IP	OUT	IP
0	1	1	0	OUT	IP	IP	OUT
0	1	1	1	OUT	IP	IP	IP
1	0	0	0	IP	OUT	OUT	OUT
1	0	0	1	IP	OUT	OUT	IP
1	0	1	0	IP	OUT	IP	OUT
1	0	1	1	IP	OUT	IP	IP
1	1	0	0	IP	IP	OUT	OUT
1	1	0	1	IP	IP	OUT	IP
1	1	1	0	IP	IP	IP	OUT

Programming in MODE 1

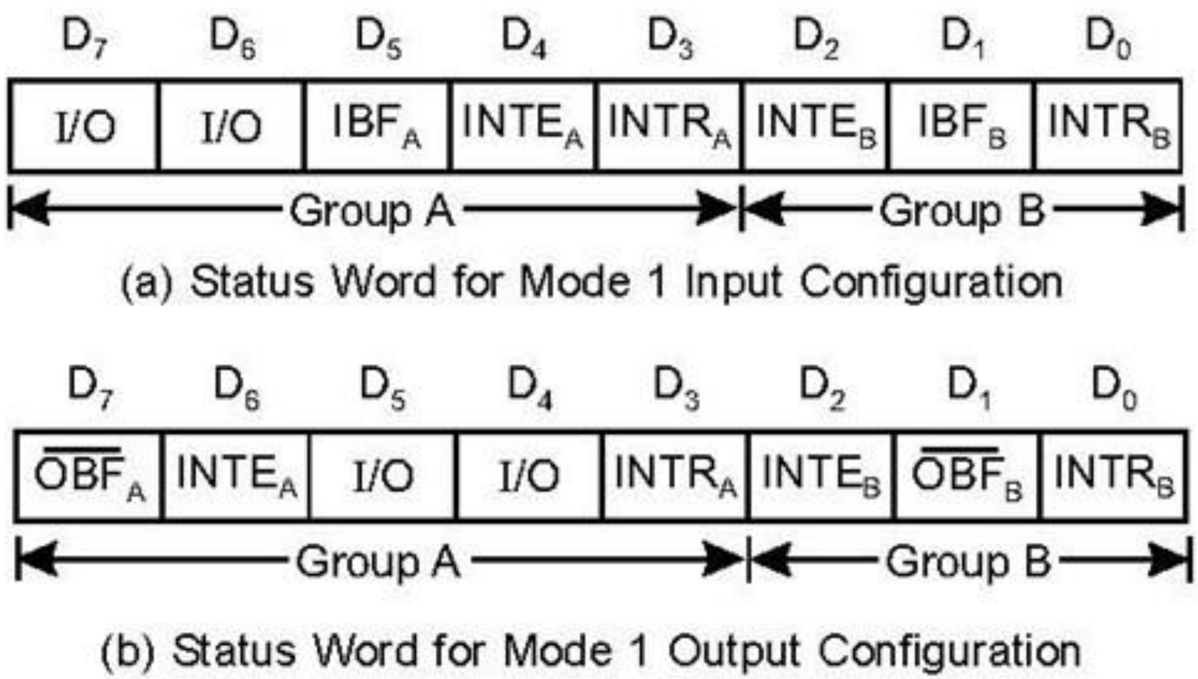


Fig. 9a.8: Status word for mode 1 (a) Input (b) Output configuration

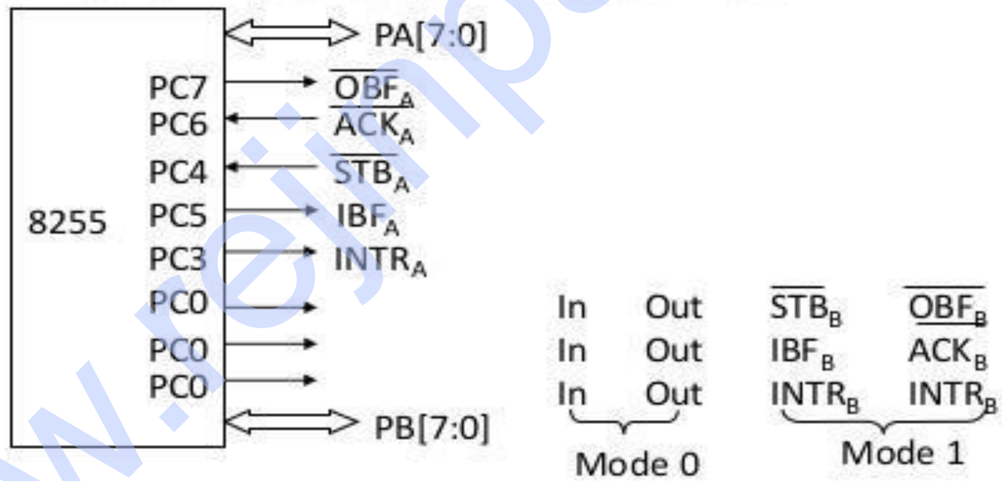
- IBF- input buffer full
- INTR- interrupt request
- INTE-interrupt enable

- OBF-output buffer full
- INTR-interrupt request
- INTE-interrupt enable

Programming in MODE 2

Programming 8255

- Mode 2:
  - Port A is programmed to be bi-directional
  - Port C is for handshaking
  - Port B can be either input or output in mode 0 or mode 1



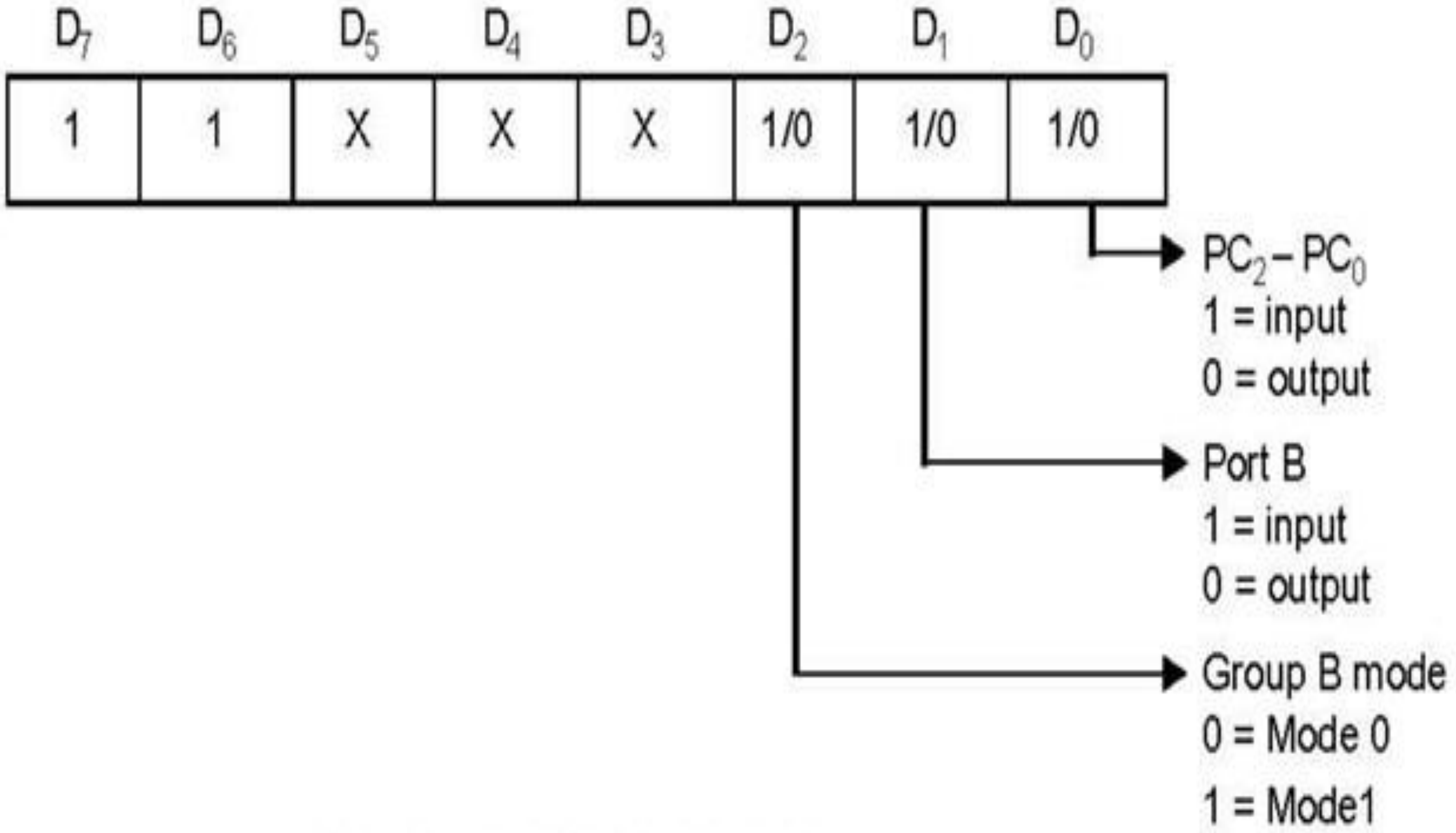
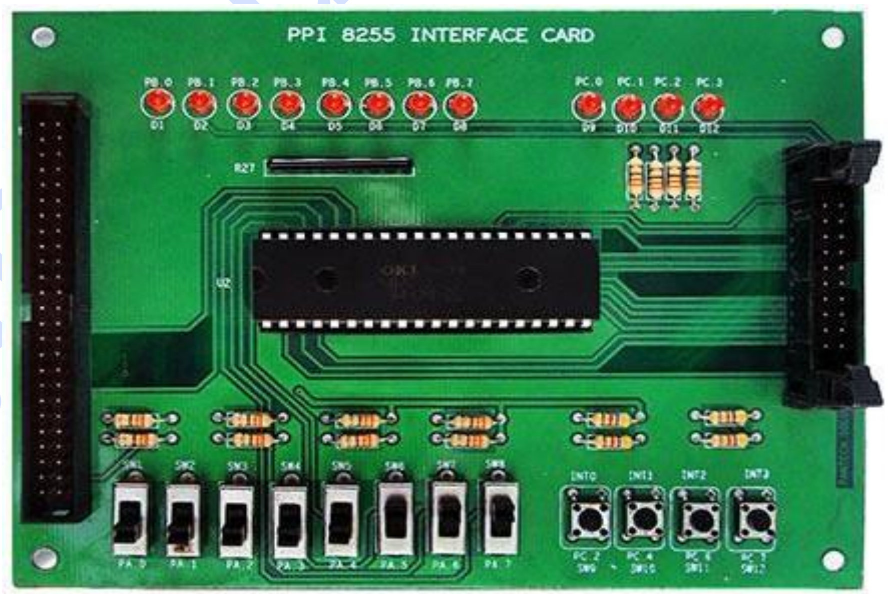
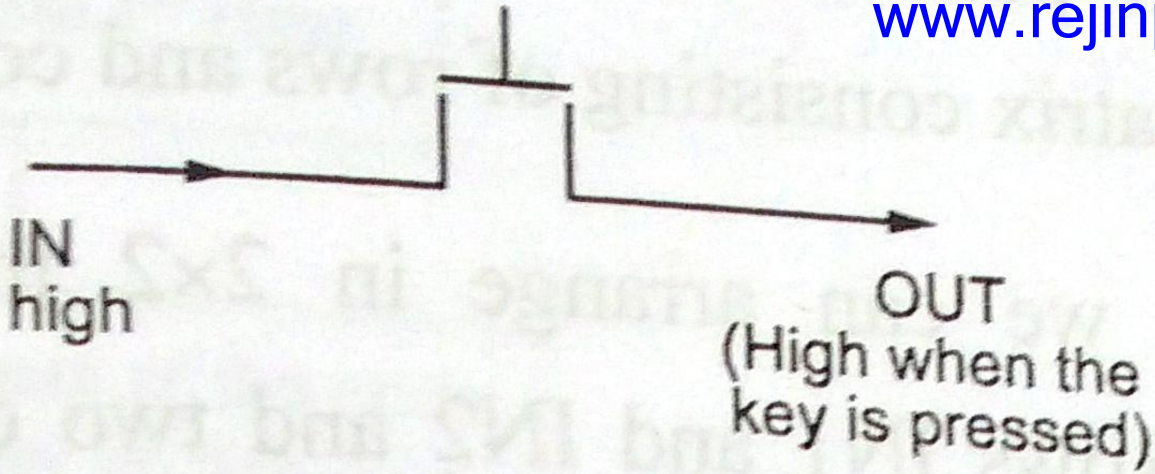


Fig. 9a.9: CWR in Mode 2

Interfacing cable





Basic Key operation

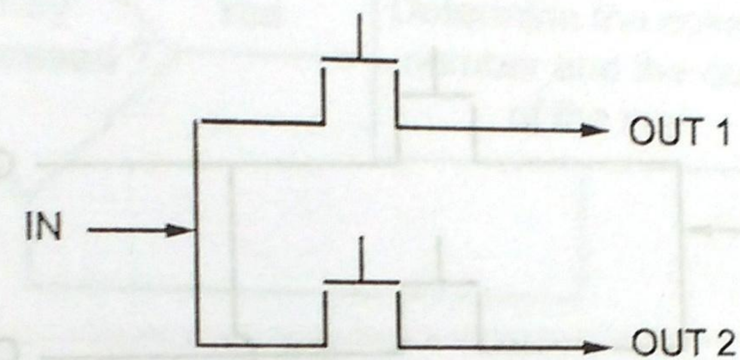


Figure 3.13 Basic Keyboard Operation – two keys

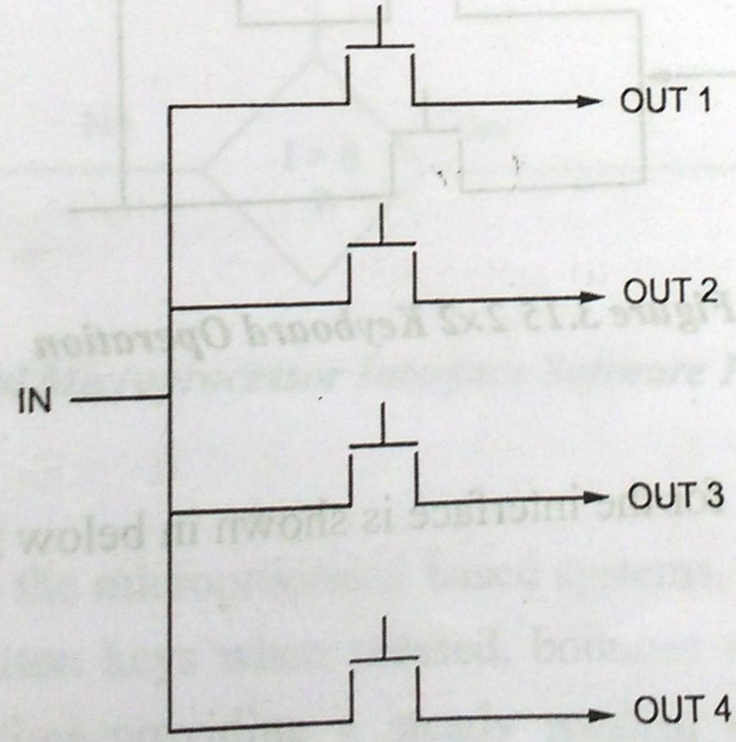
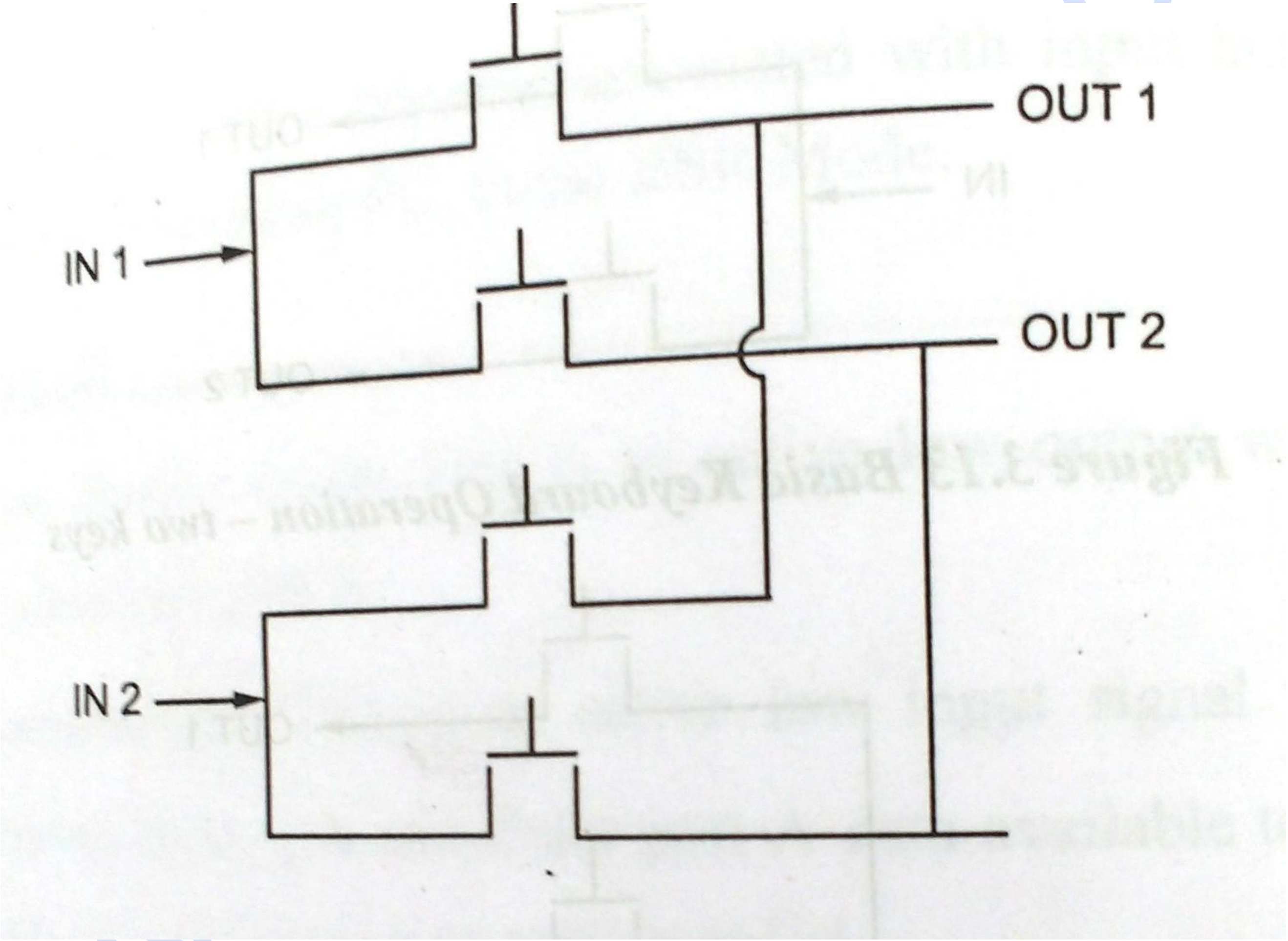
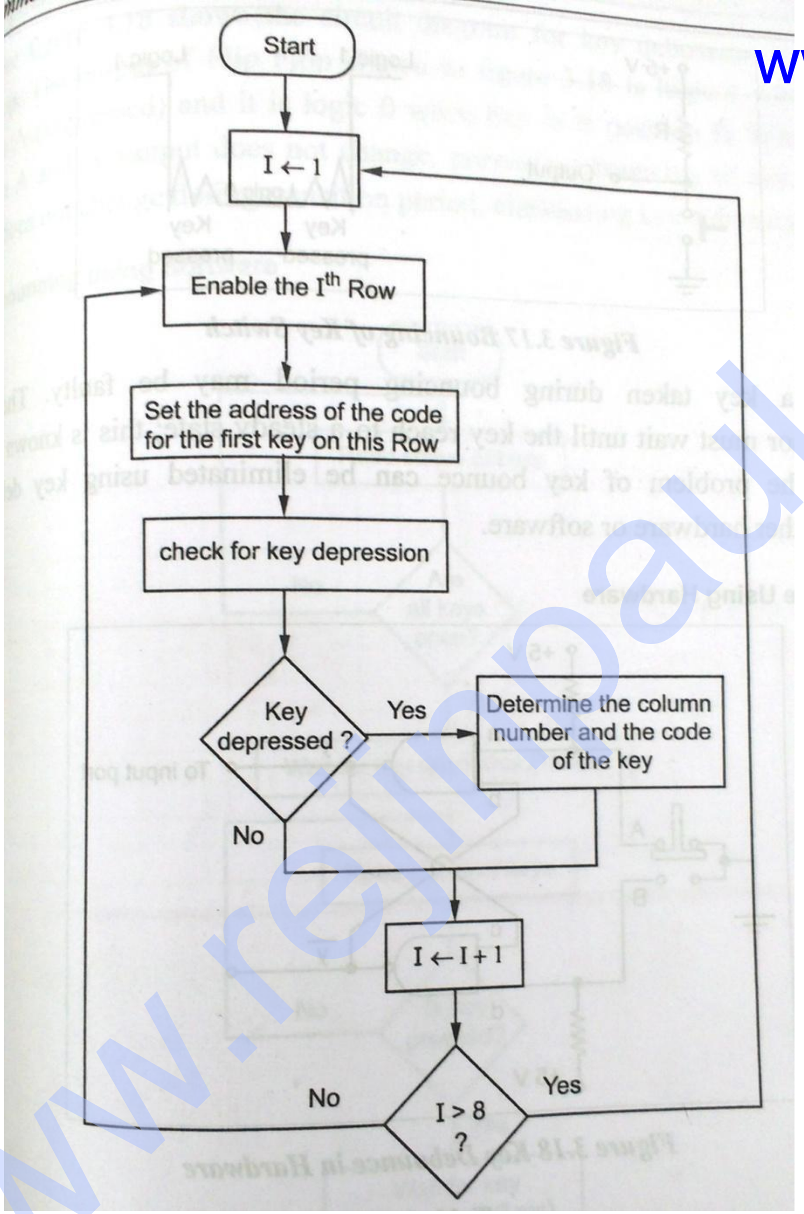


Figure 3.14 Basic Keyboard Operation – Four Keys

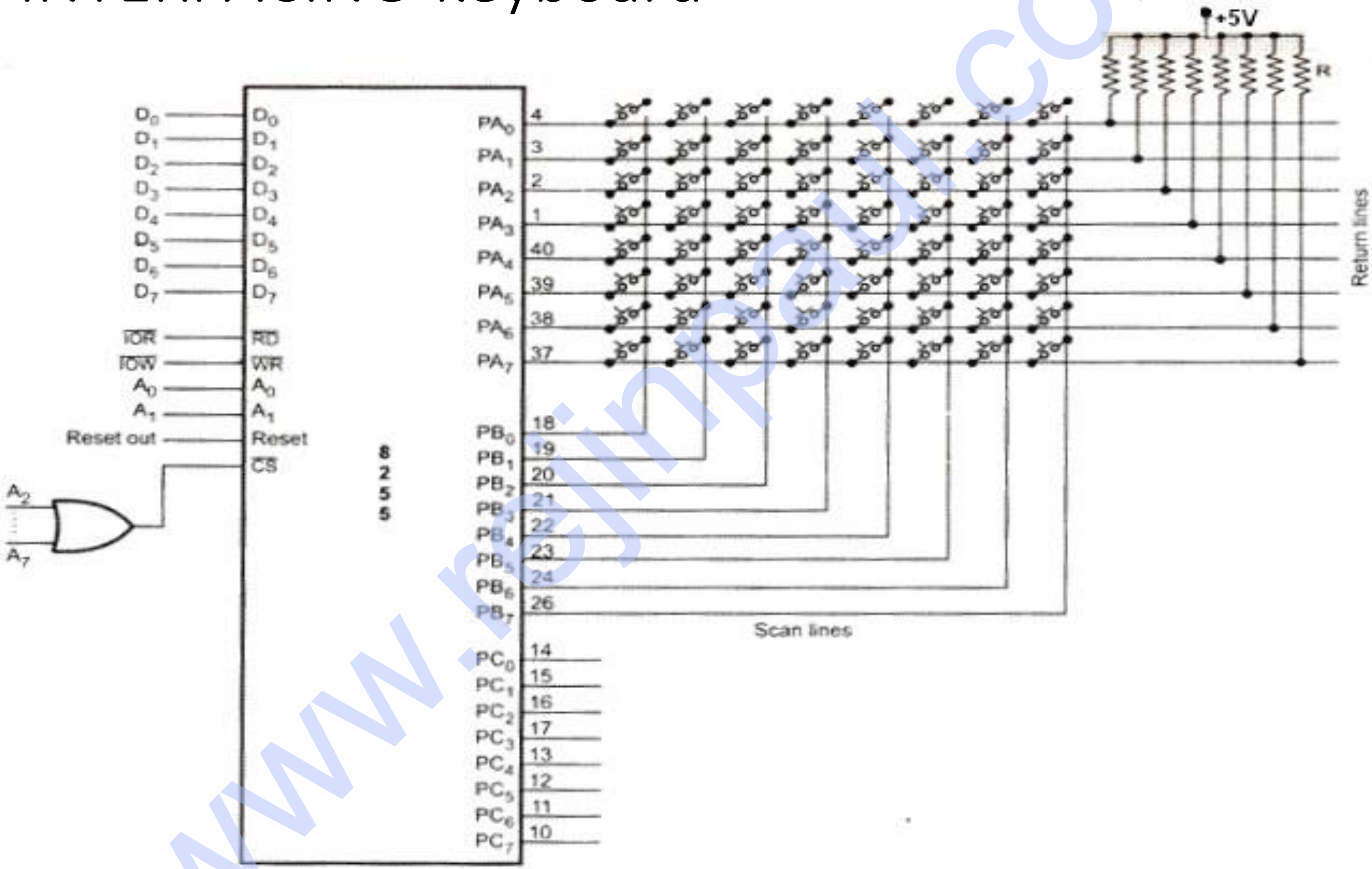


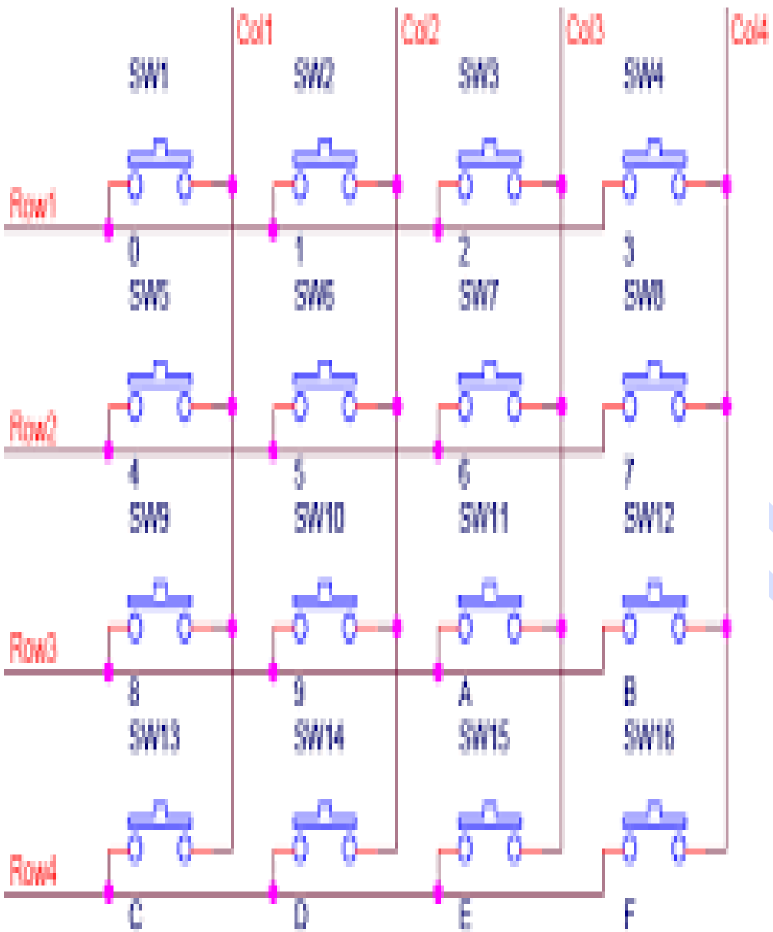
2 X 2 Key operation

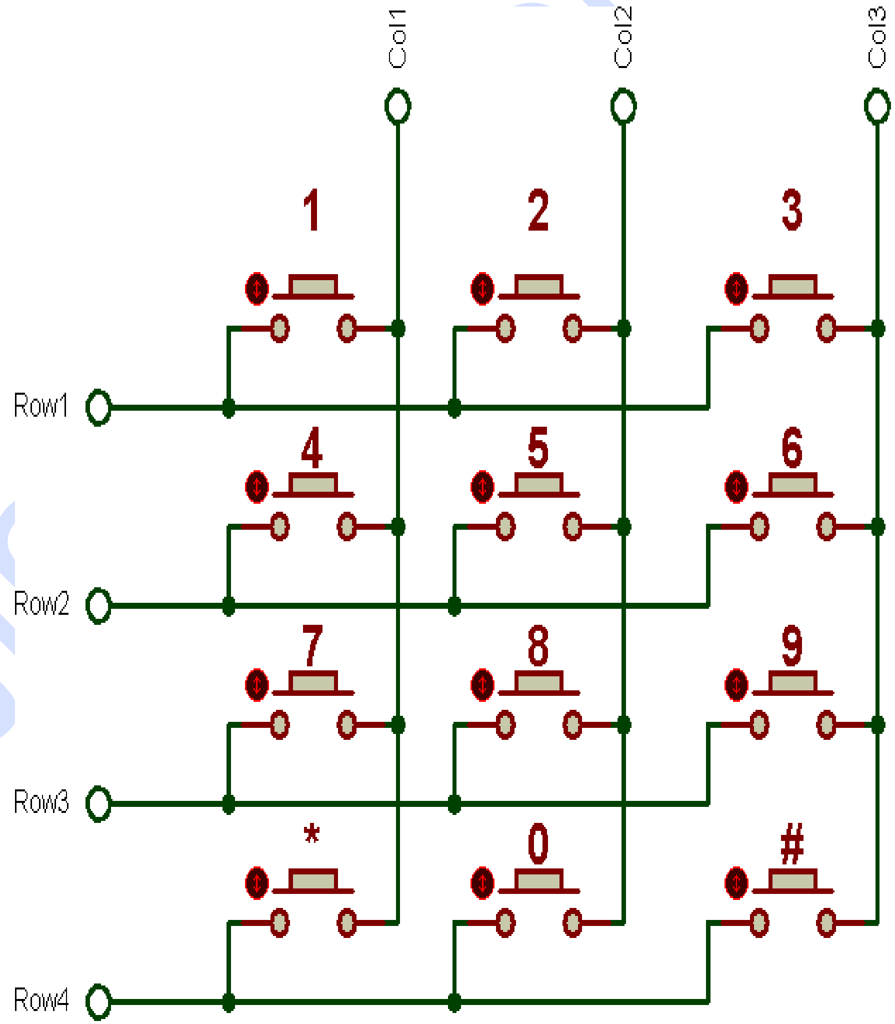


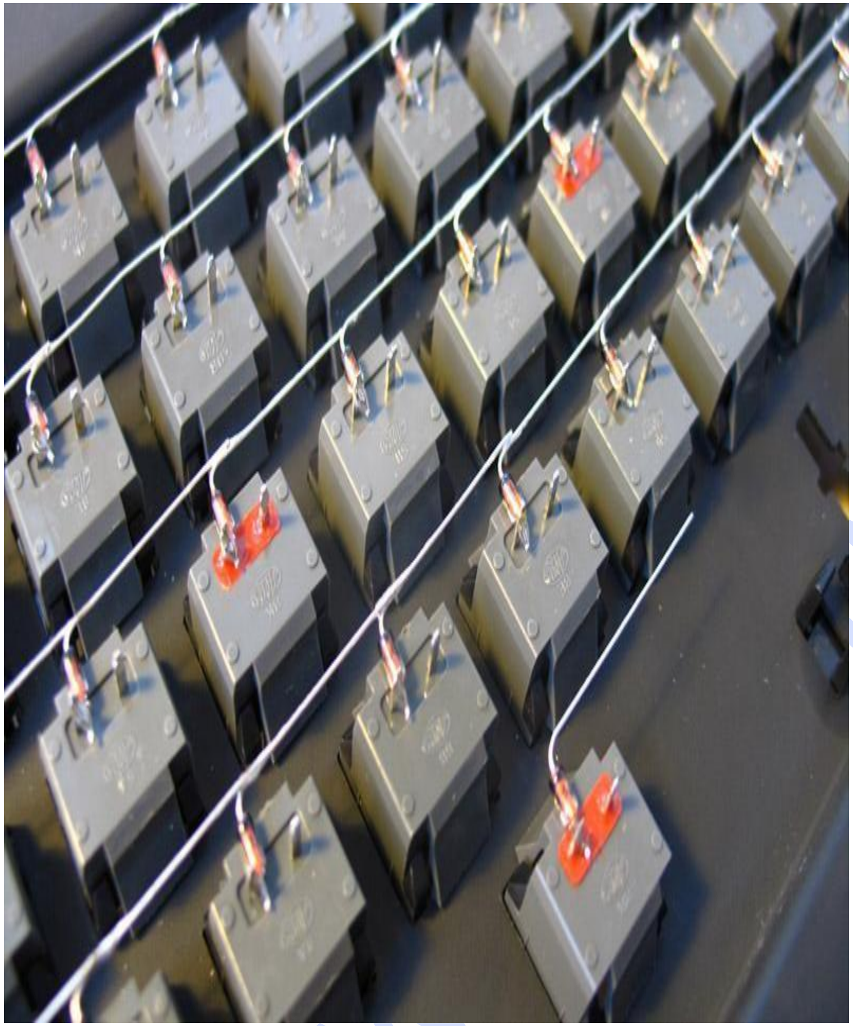
Keyboard Microprocessor Interface software Flowchart

INTERFACING-keyboard

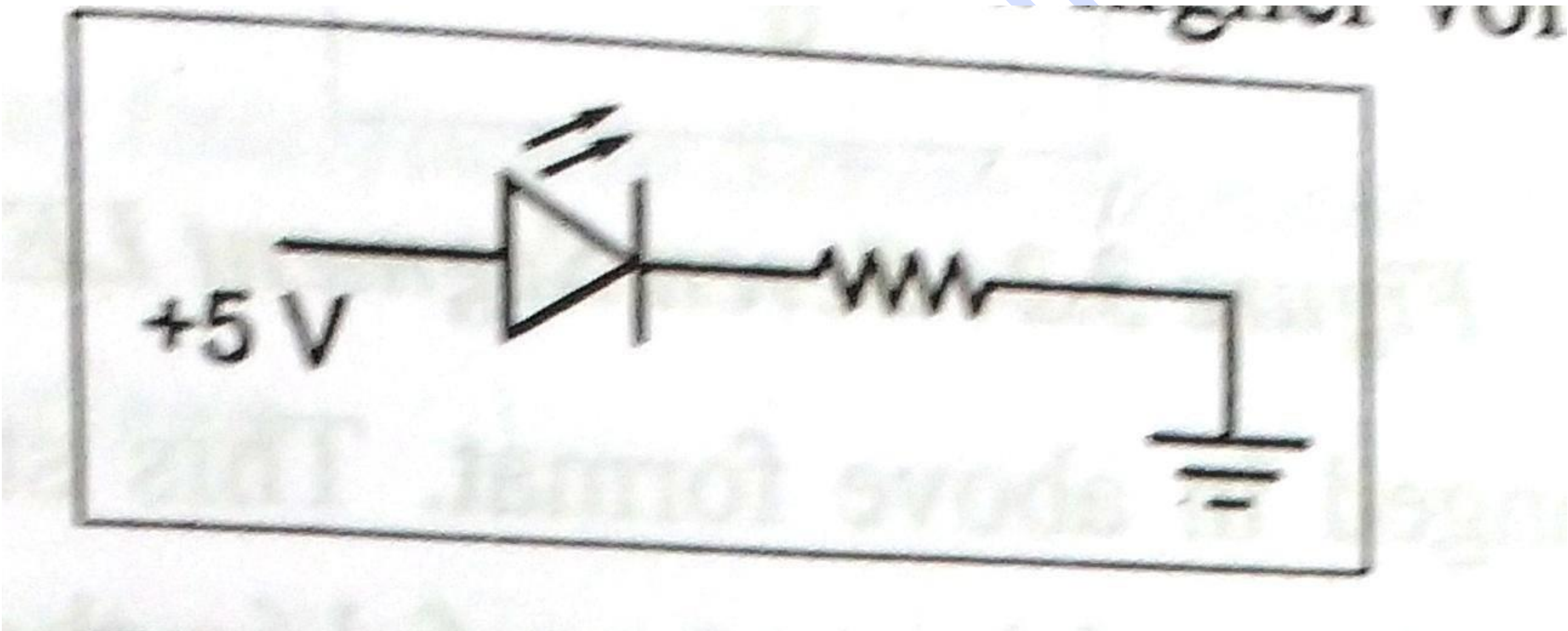



Col1
Col2
Col3
Col4
SW1
SW2
SW3
SW4
Row1
0
1
2
3
SW5
SW6
SW7
SW8
Row2
4
5
6
7
SW9
SW10
SW11
SW12
Row3
8
9
A
B
SW13
SW14
SW15
SW16
Row4
C
D
E
F

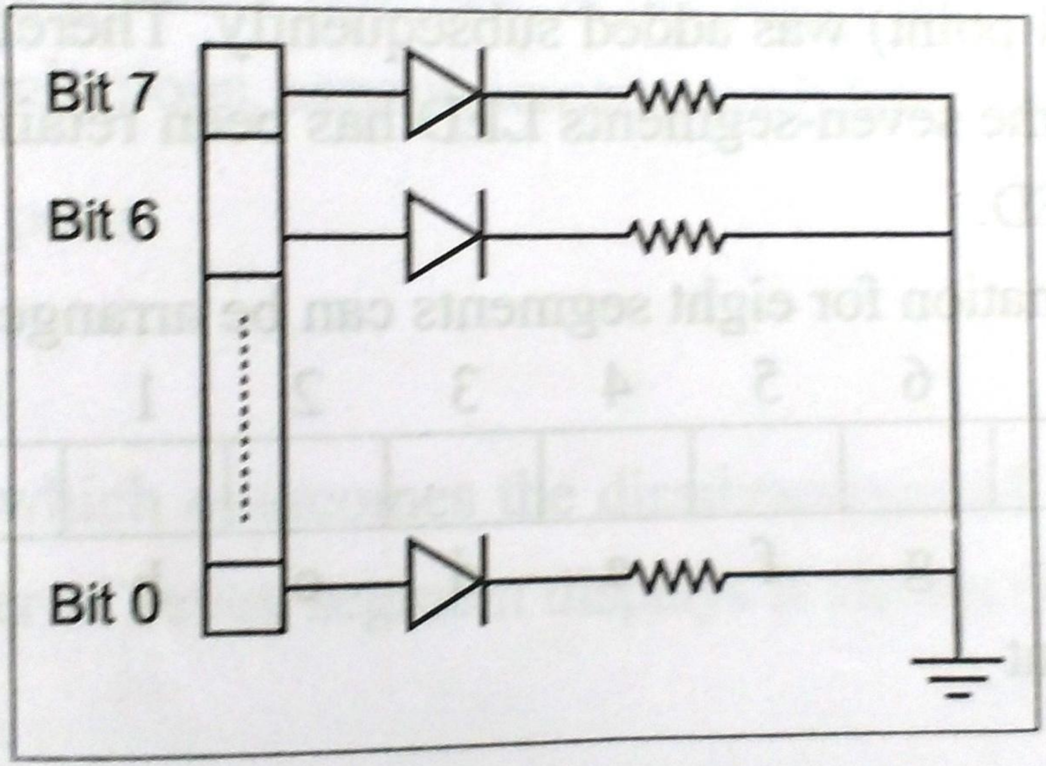

Col1
Col2
Col3
1
2
3
Row1
4
5
6
Row2
7
8
9
Row3
*
0
#
Row4



LED Operation

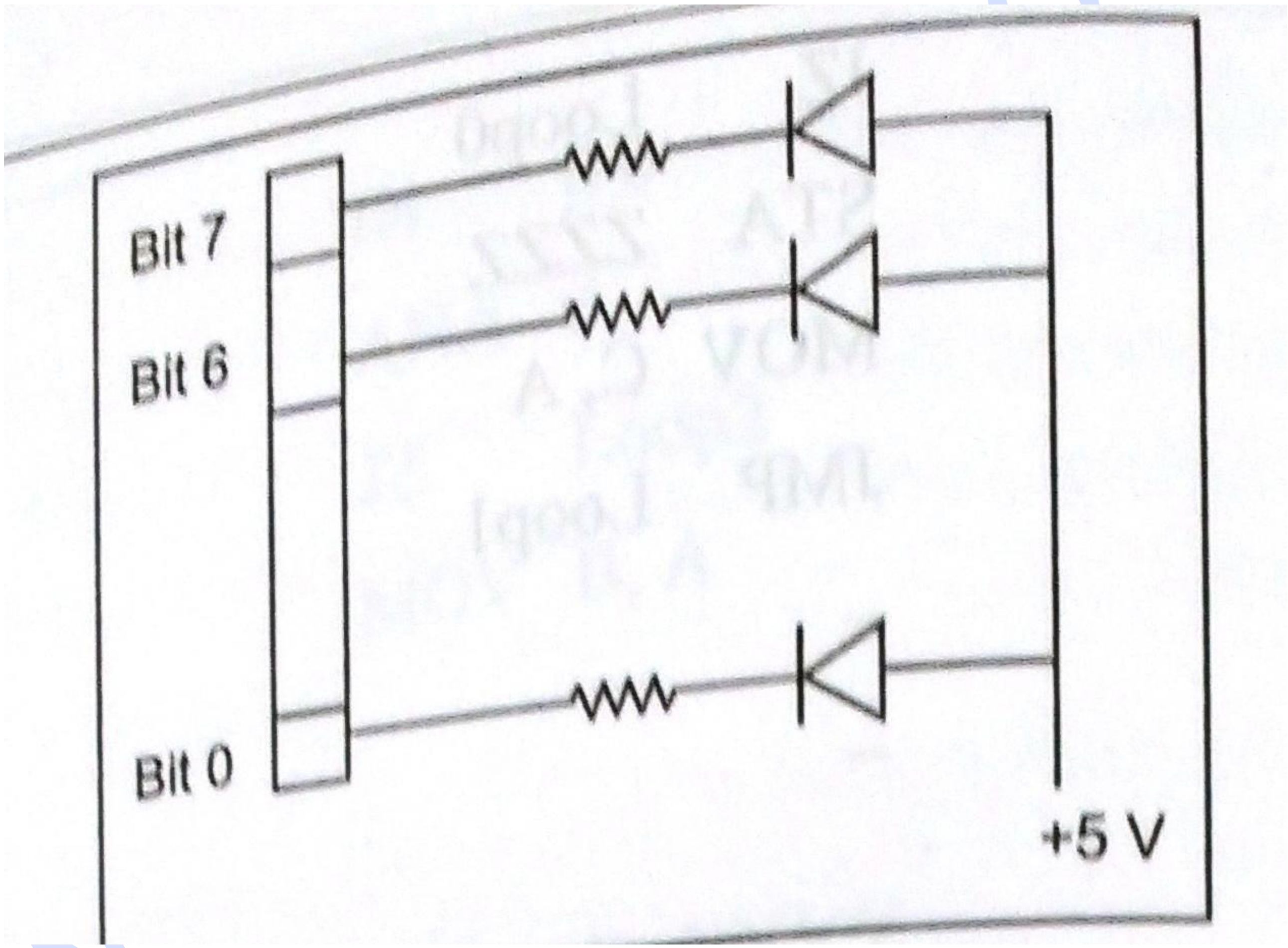


igure 3.22 :



re 3.22 Microprocessor Interface to LED (Common Cathode)

Microprocessor interface to LED (Common anode)



ited.

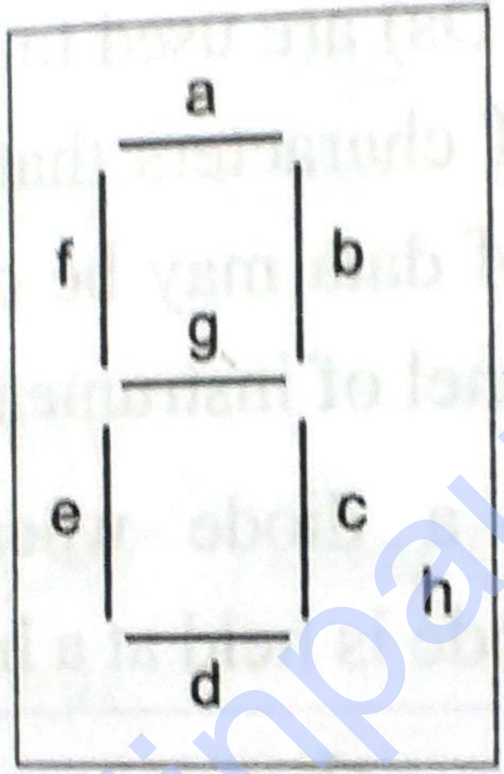
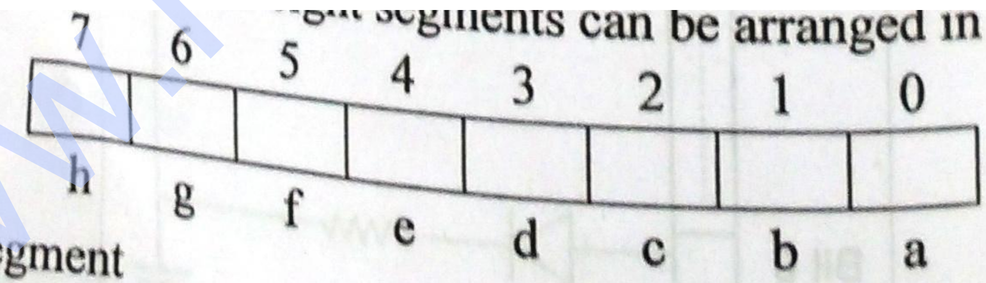


Figure 3.24 Seven Segment LED

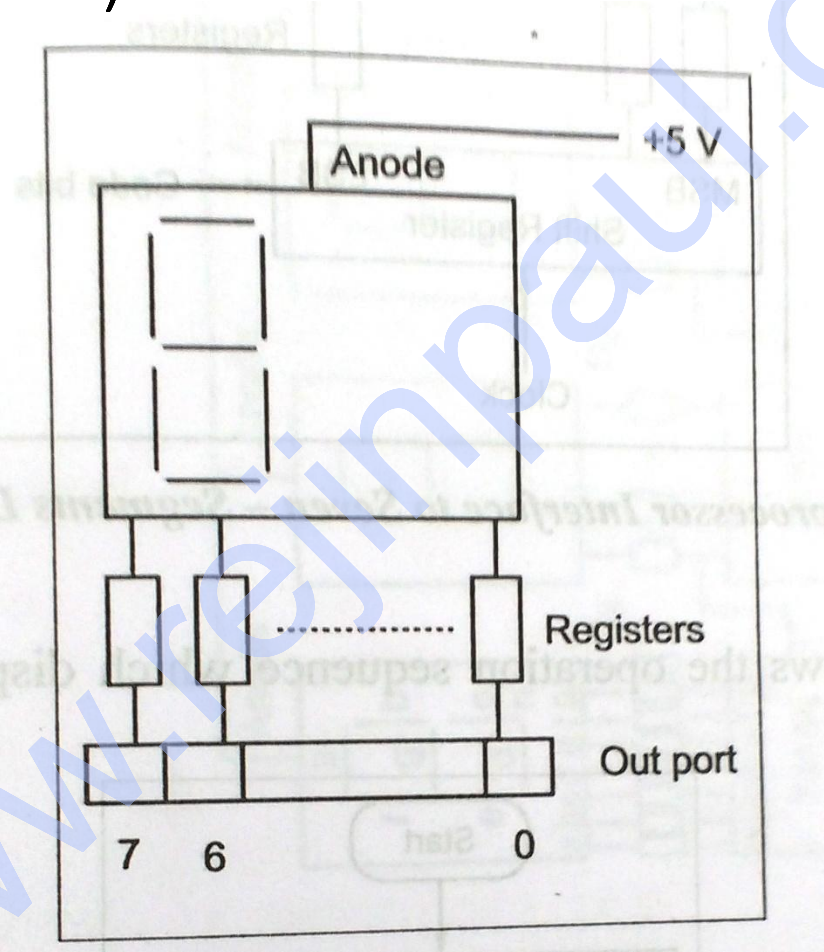
...ght segments can be arranged in on



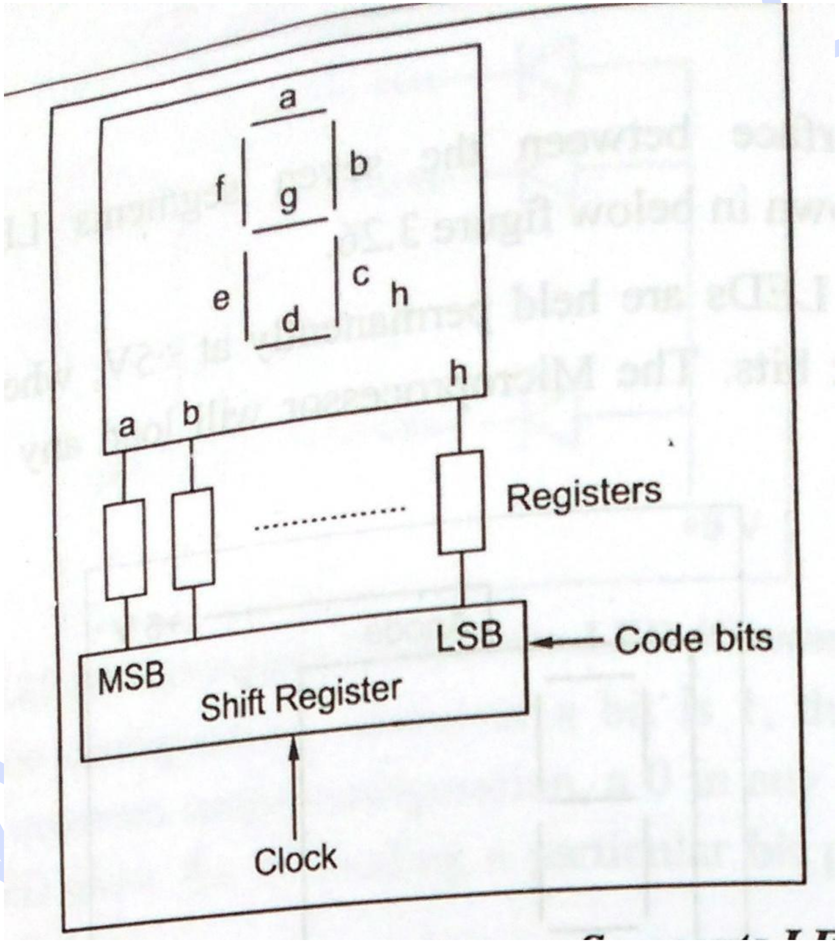
0 – Glow the segment

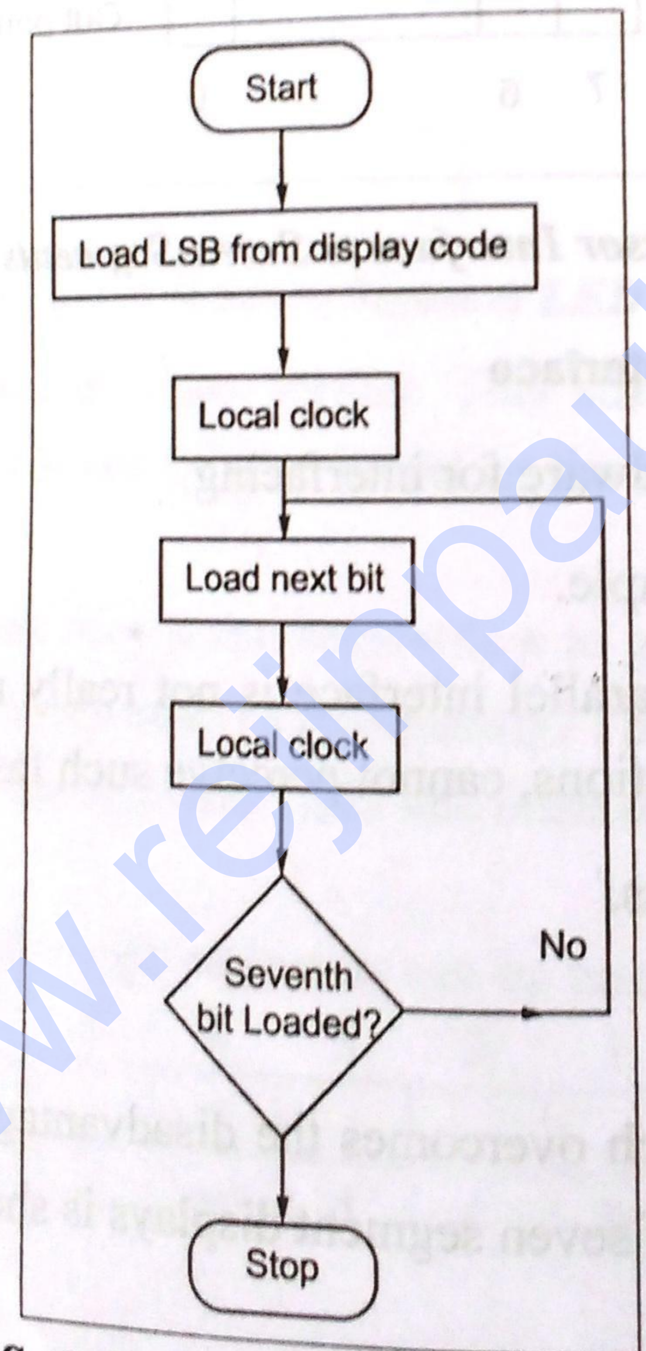
1 – Do not glow

Microprocessor interface to 7 segment LED (Parallel)



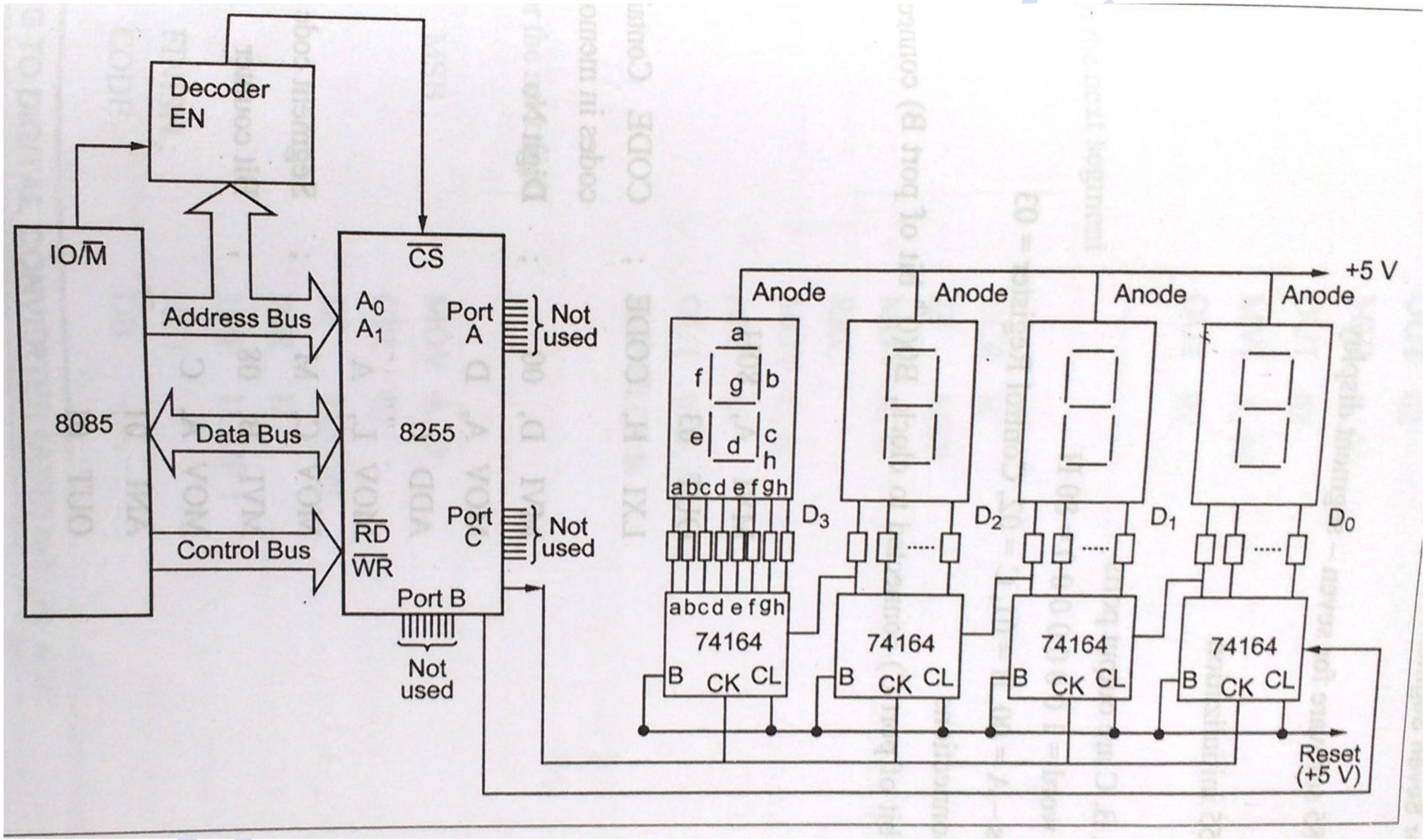
Microprocessor interface to 7 segment LED (serial)

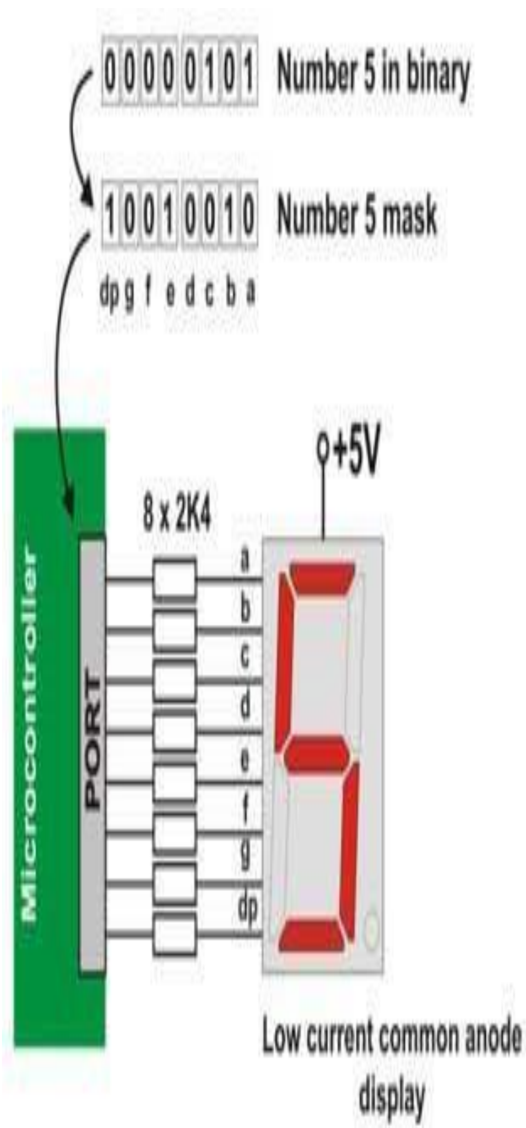




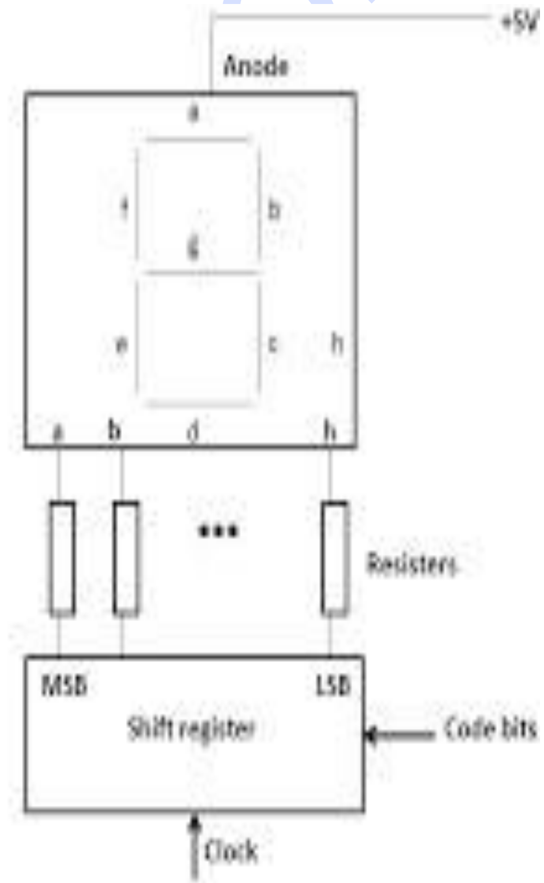
Serial interface of 7 segment LED to Microprocessor software flowchart

INTERFACE-LED display



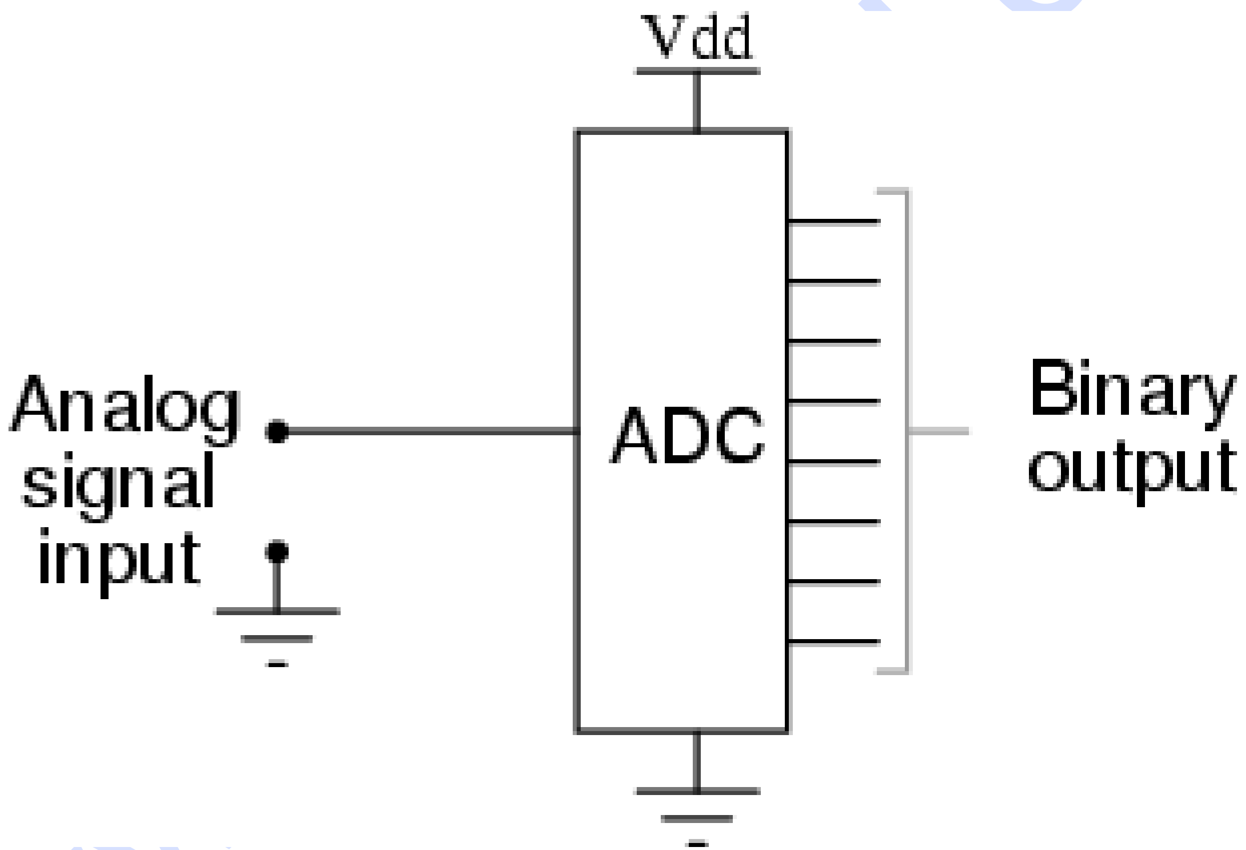


Hex value is 92H



Microprocessor interface to seven-segment LED(serial interface)

ADC INTERFACE



BLOCK diagram of ADC 0808

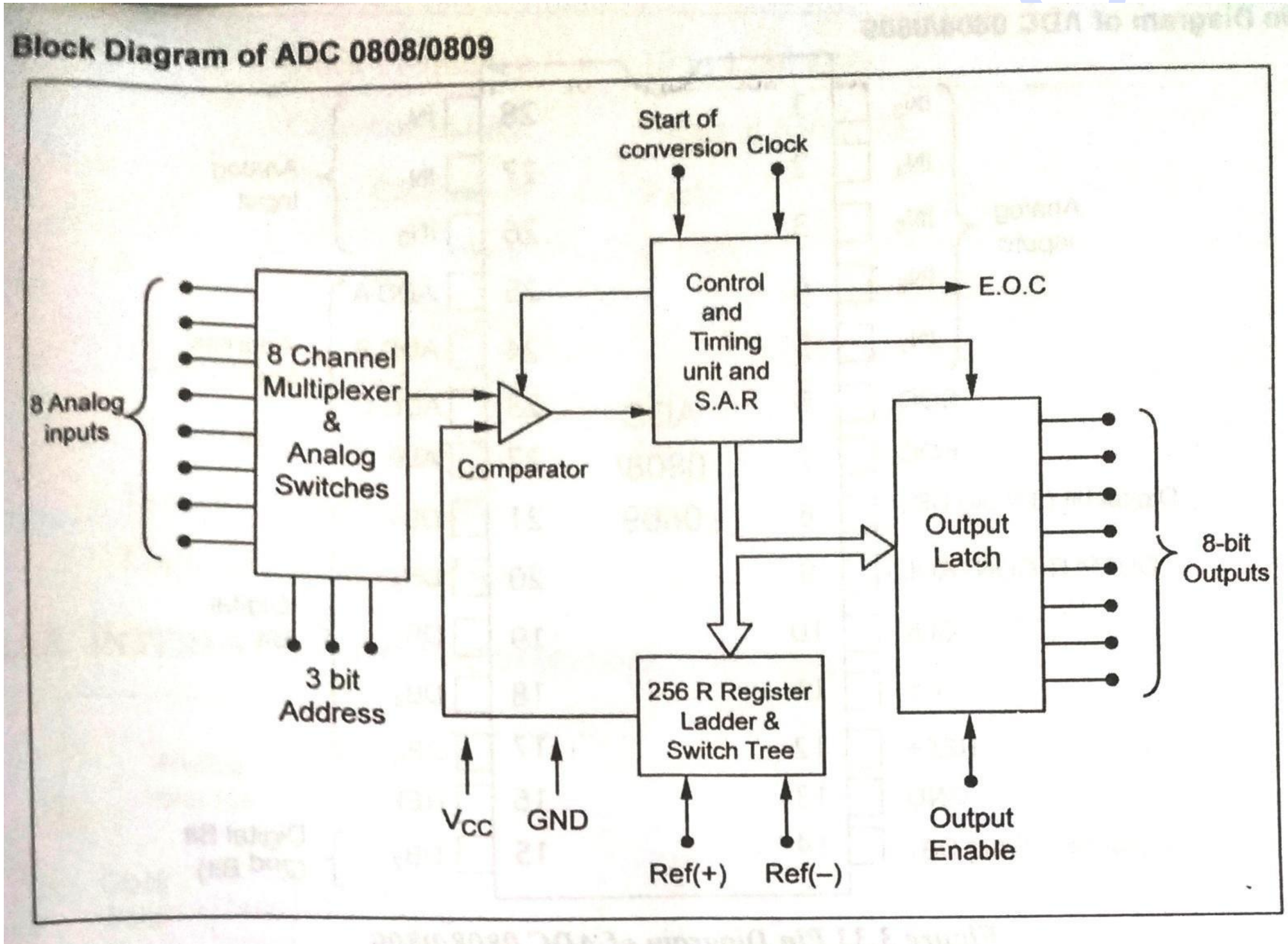


Figure 3.31 Block Diagram of ADDC 0808/ADC 0809

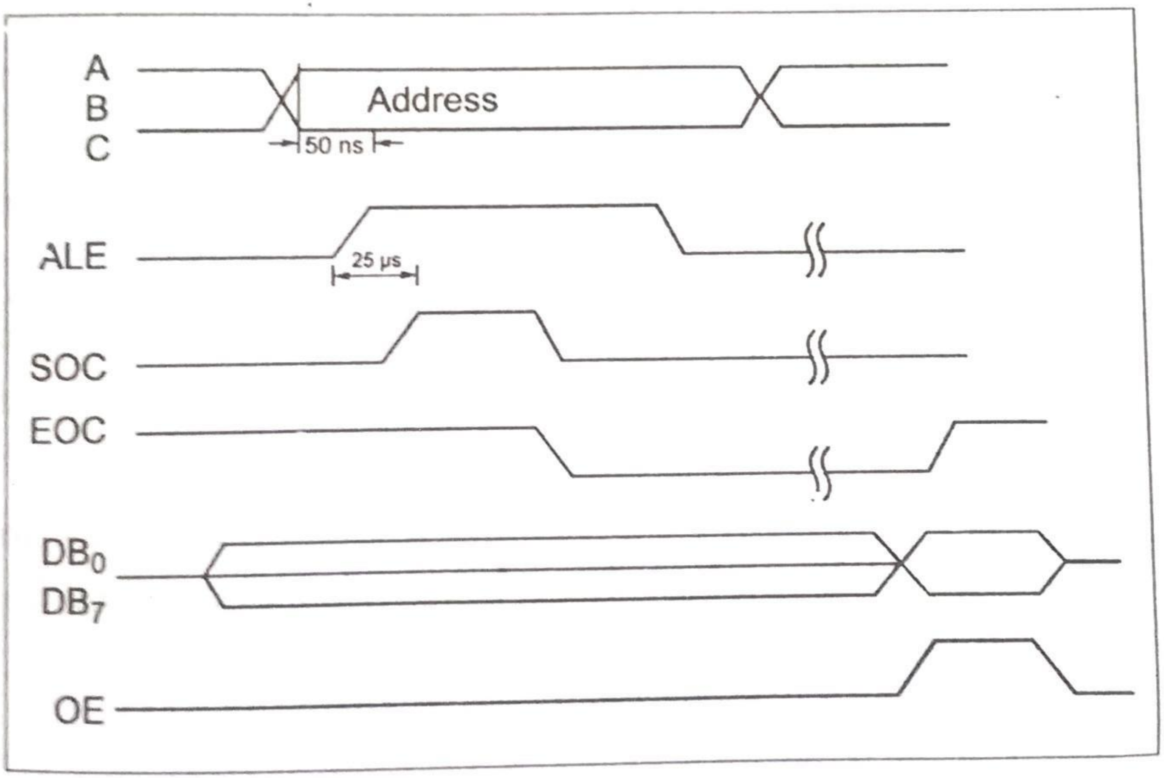
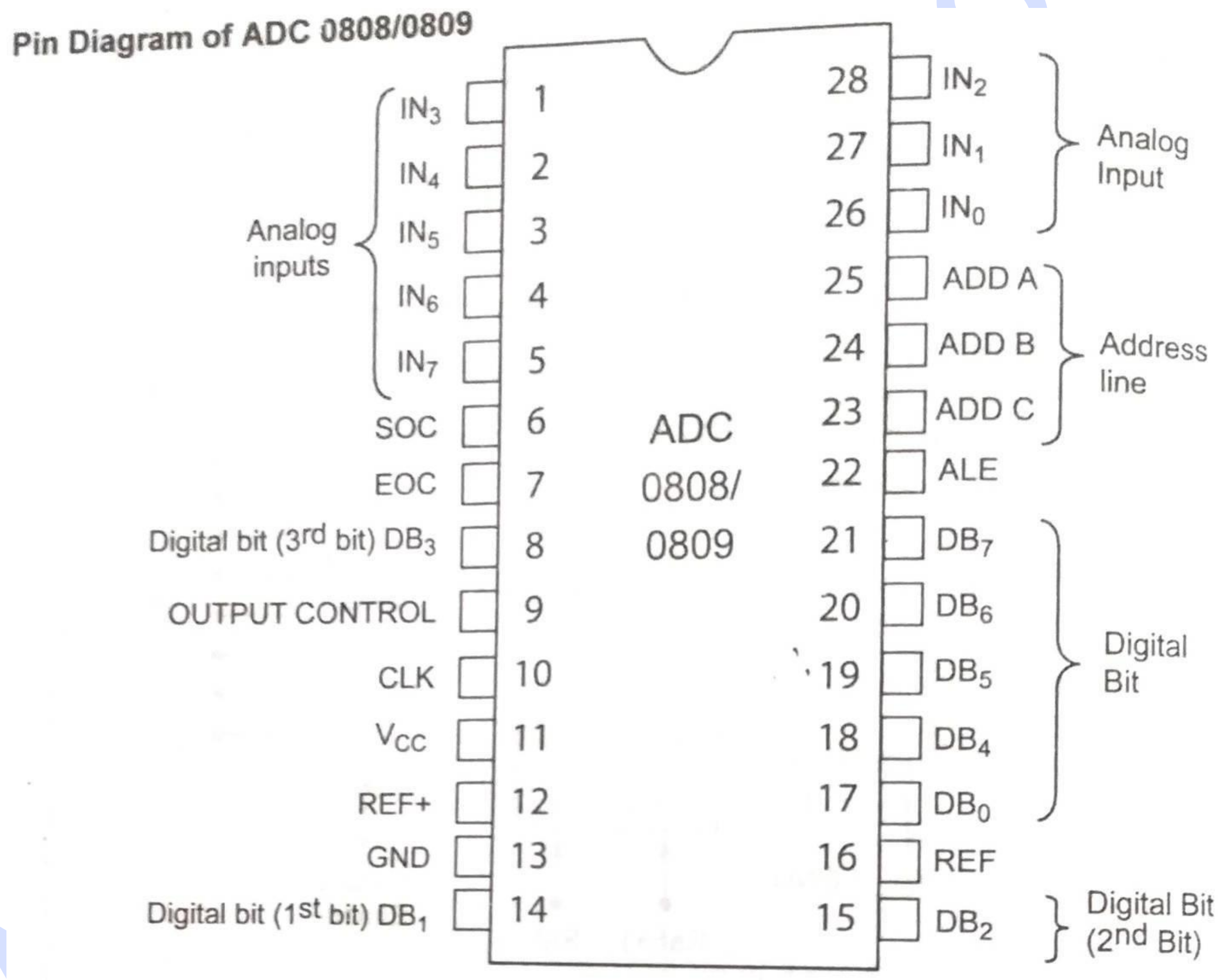
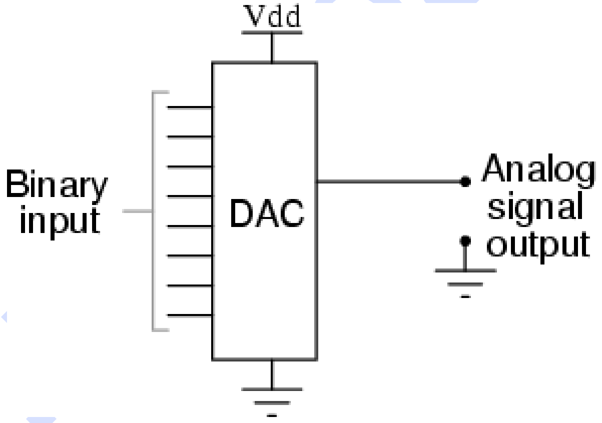


Figure 3.32 Timing Waveforms for ADC 0808

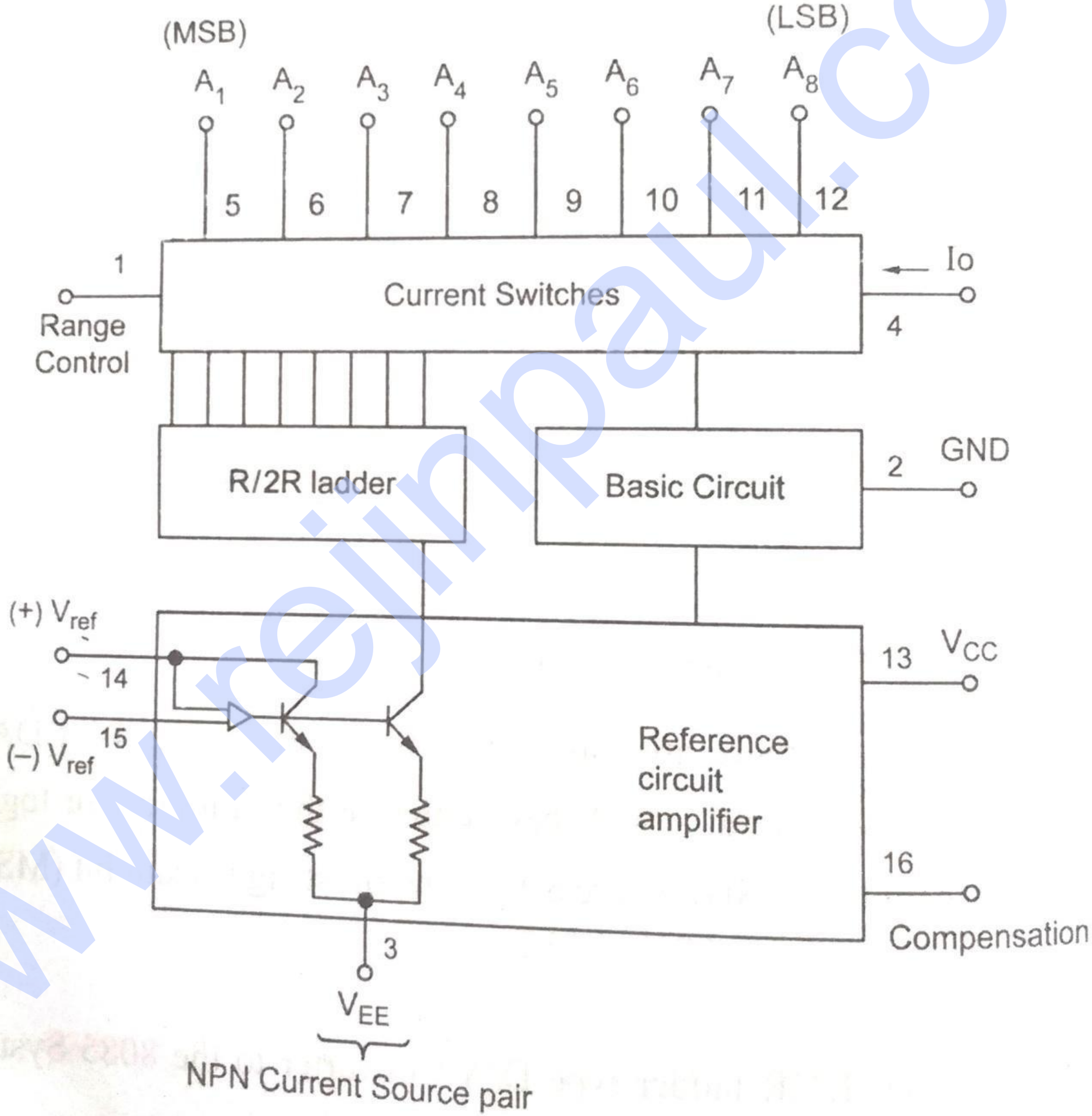
PIN diagram of ADC 0808



DAC INTERFACE



Pin diagram of DAC



Pin diagram of DAC

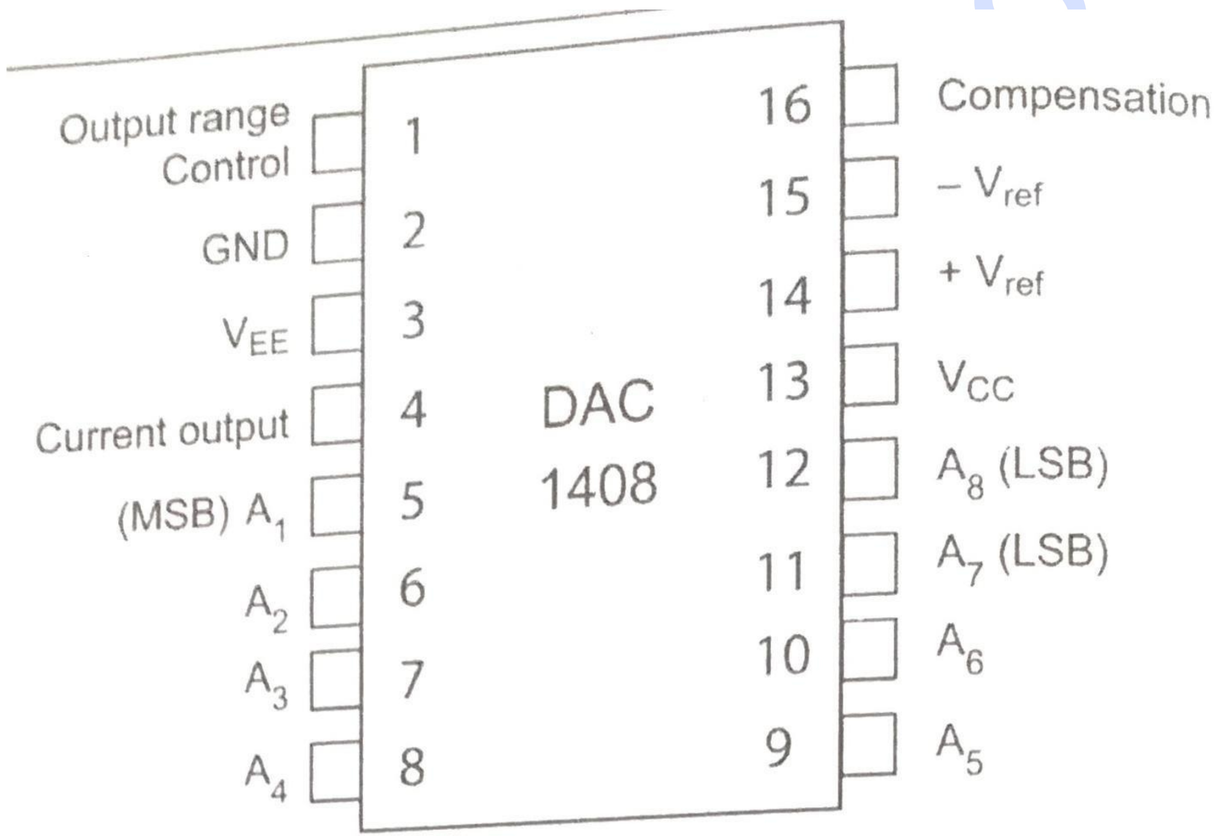
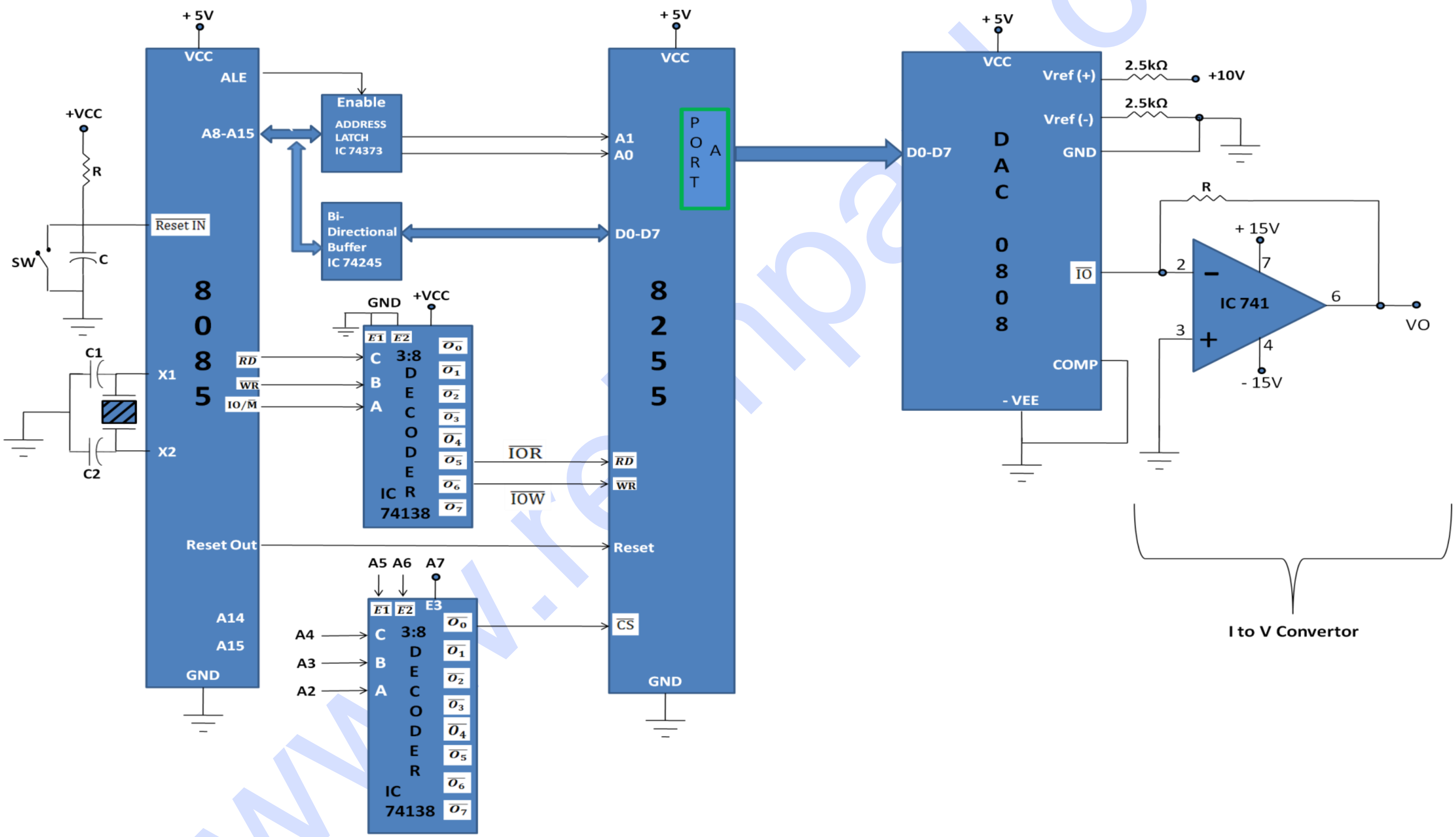


Figure 3.36 Pin Diagram of DAC 1408

INTERFACING diagram for DAC

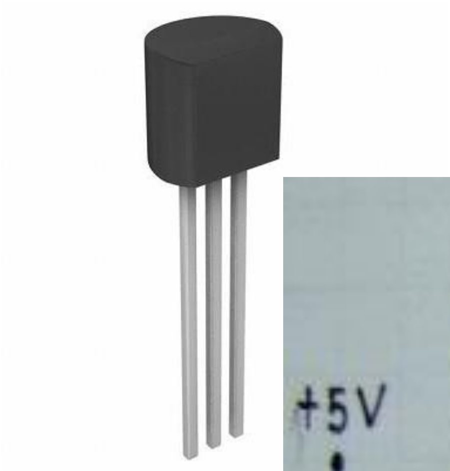


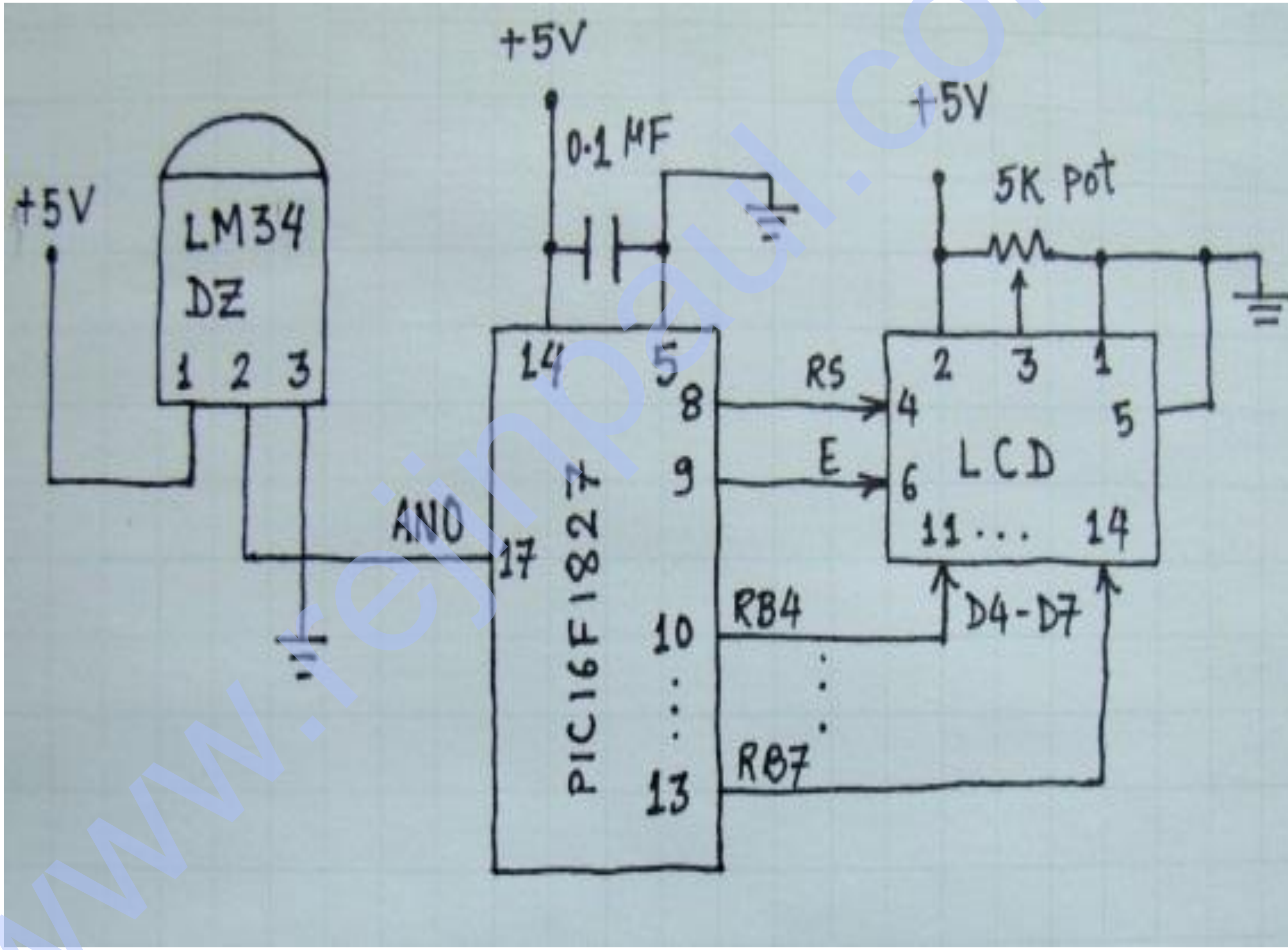
TEMPERATURE CONTROL

- **Temperature sensor** –convert temp to electrical signal by thermistor
- **Transducer** convert physical data into electrical signal
- **Physical data** –temp, light, flow, speed etc…
- **LM34 & LM35** –temperature sensor by NATIONAL SEMICONDUCTOR CO-OPERATION

- **LM34**
- Output voltage is linearly proportional to **Fahrenheit** temp
- No external **calibration**
- **10mV** for each degree of Fahrenheit temp

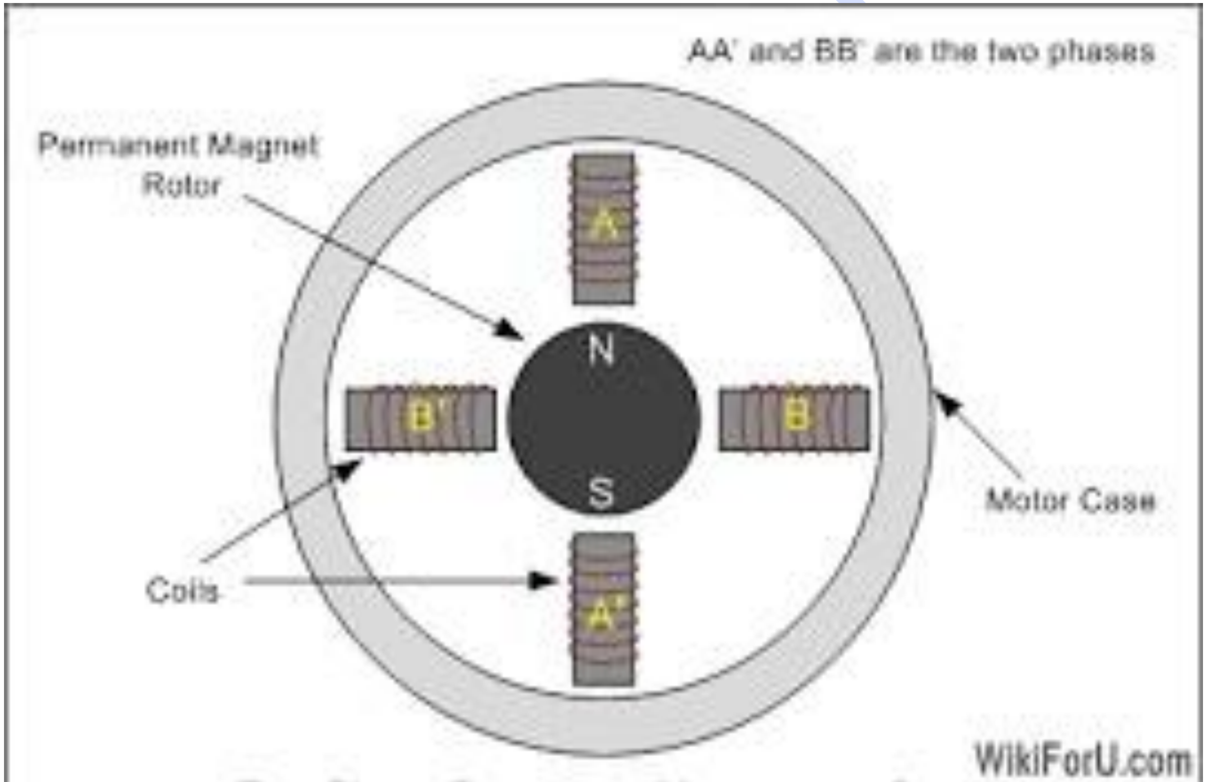
- **LM35**
- Output voltage is linearly proportional to **Celsius** temp
- No external **calibration**
- **10mV** for each degree of Centigrate temp




+5V
LM34
DZ
1 2 3
+5V
0.1 μF
14 5
8
9
17
10
13
AN0
PIC16F1827
+5V
5K Pot
RS
E
2 3 1
4
5
6
LCD
11 ... 14
RB4
RB7
D4-D7

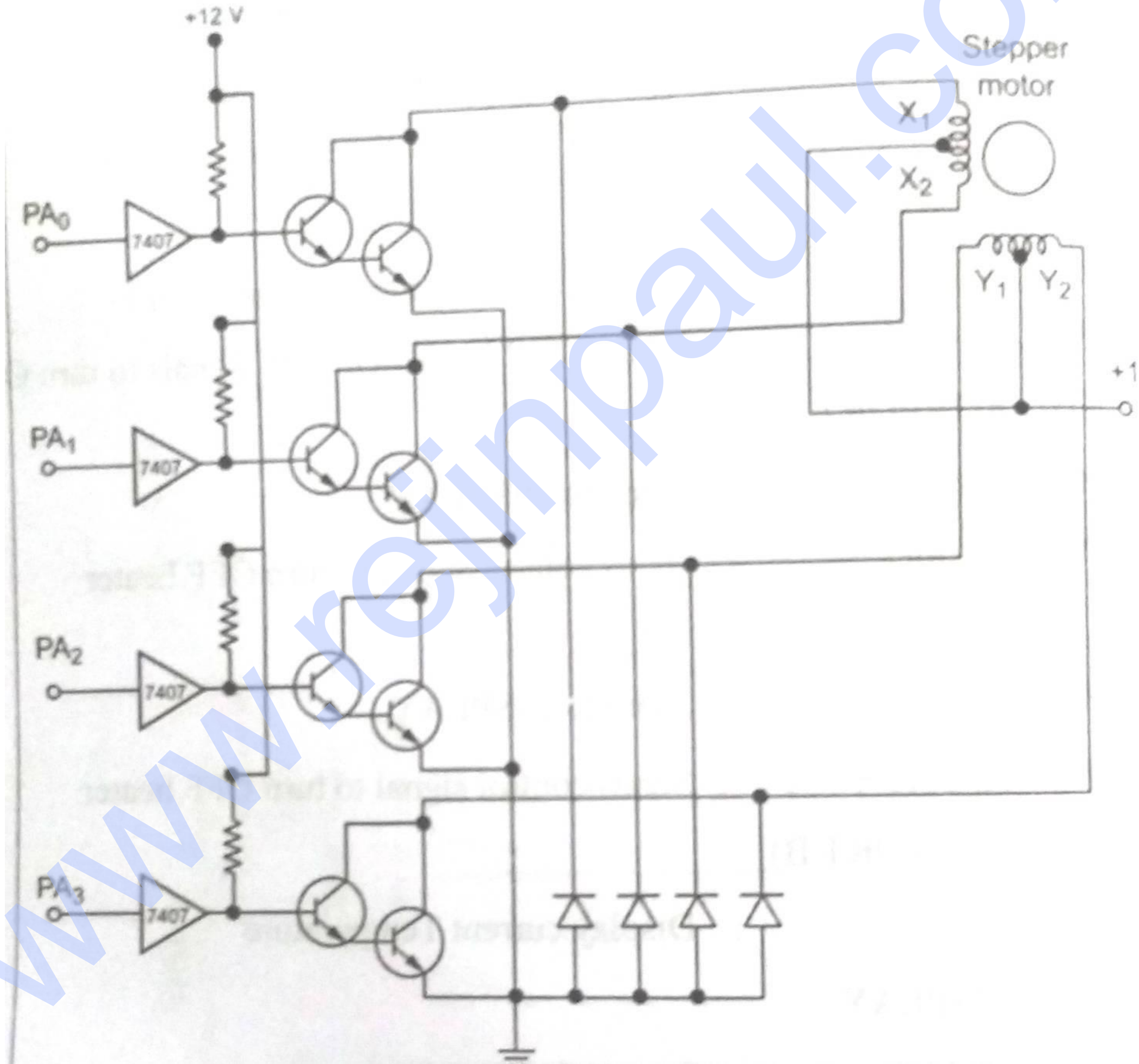
STEPPER MOTOR CONTROL interface

- Digital motor used to translate electrical pulse into mechanical movement
- Center tap winding connected to 12 V supply
- Motor can be excited by grounding four terminals of the two windings
- **ROTOR**-Stepper motor has permanent magnet rotor .It is also known as shaft
- **STEP ANGLE-**It is minimum degree of rotation associated with a single step



Two-Phase Permanent Magnet type Stepper

Stepper Motor Interface



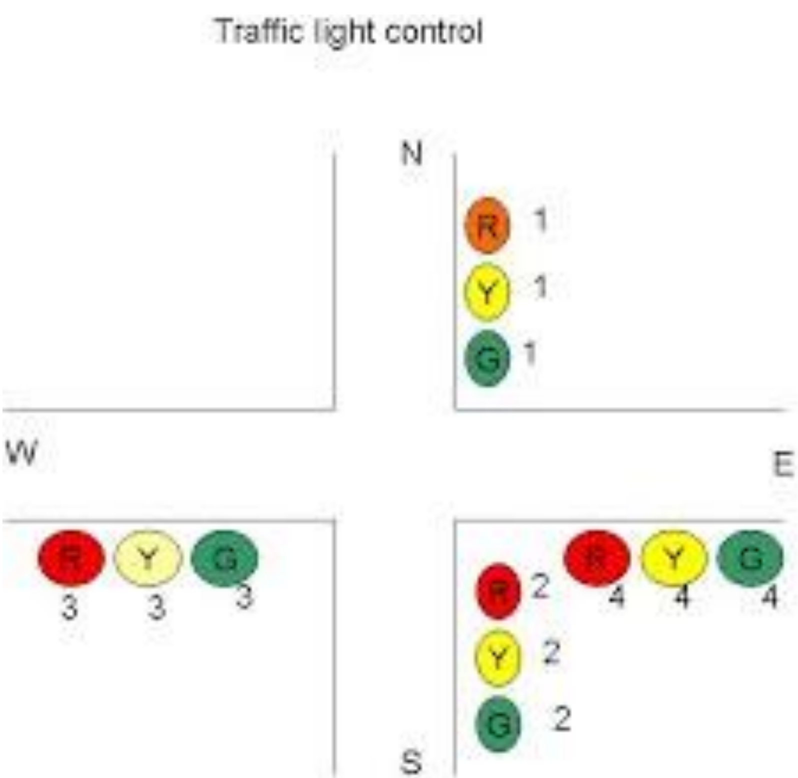
Excitation Table

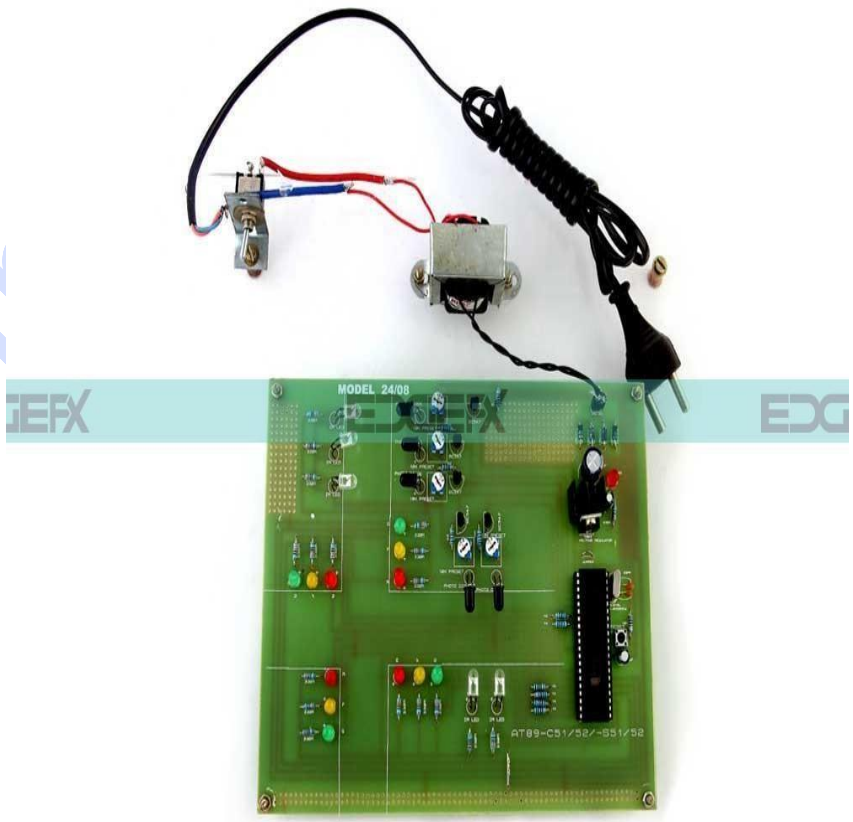
Step	X1	X2	X3	X4
1	0	1	0	1
2	1	0	0	1
3	1	0	1	0
4	0	1	1	0
1	0	1	0	1

Traffic Light Control System

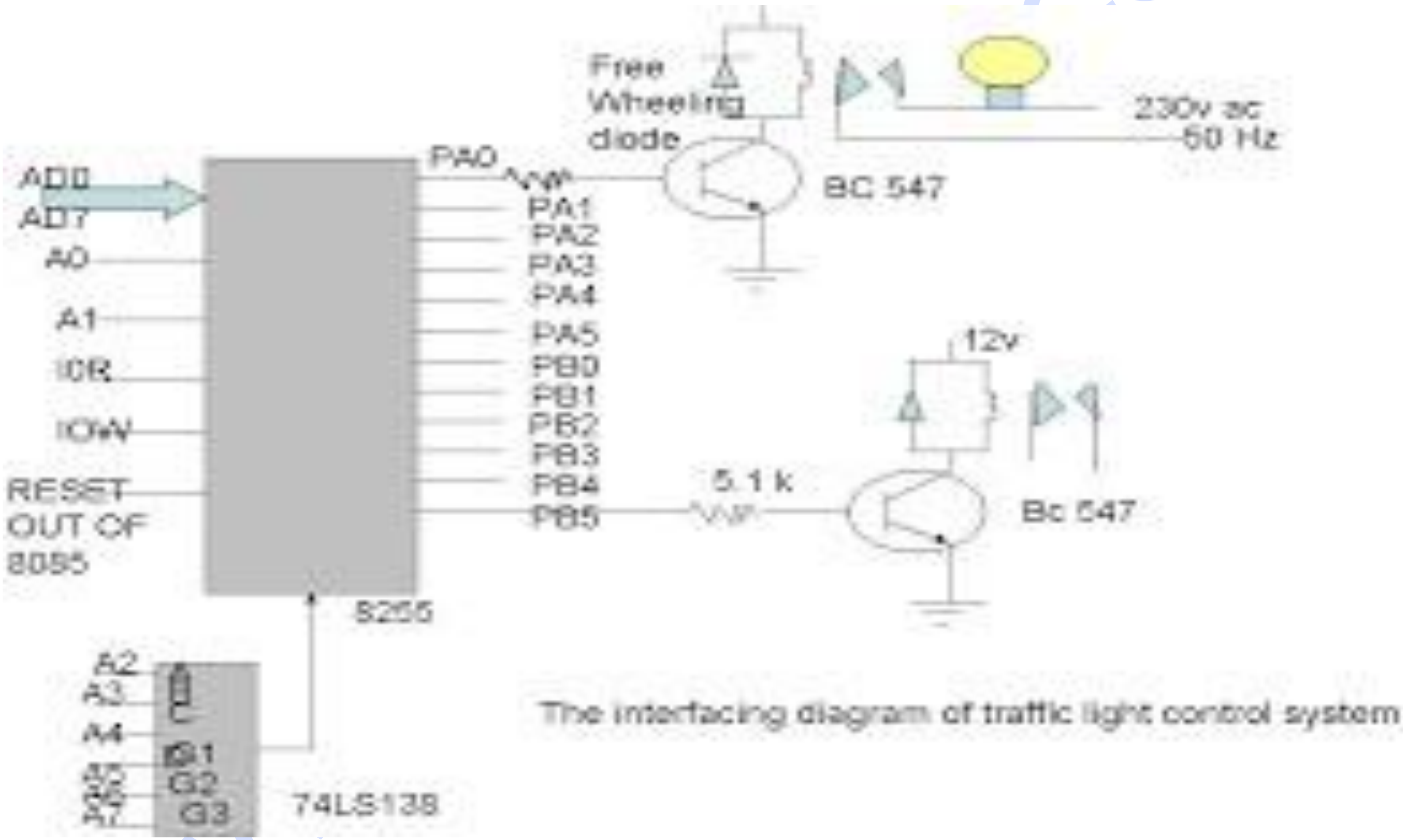
- Allow traffic from W to E and E to W transition for 20 seconds
- Give transition period of 5 seconds (yellow bulbs ON)
- Allow traffic from N to s and S to n for 20 seconds
- Give transition period of 5 seconds (yellow bulbs ON)
- Repeat the process

Traffic Light Control System





Interfacing diagram for Traffic Light Control System



The interfacing diagram of traffic light control system

Pins	Light	Pins	Light
PA_0	R_1	PB_0	R_3
PA_1	Y_1	PB_1	Y_3
PA_2	G_1	PB_2	G_3
PA_3	R_2	PB_3	R_4
PA_4	Y_2	PB_4	Y_4
PA_5	G_2	PB_5	G_4

Table 1

UNIT 4

PROGRAMMABLE LOGIC CONTROLLER

Content

- Introduction
- Basic structure
- Input and output processing
- Programming
- Mnemonics
- Timers, counters and internal relays
- Data handling
- Selection of PLC

PROGRAMMABLE LOGIC CONTROLLER

- A Programmable Logic Controller(PLC) is a digital computer used for automation of typically industrial electromechanical processes, such as control of machinery on factory assembly lines, amusement rides or light fixtures.

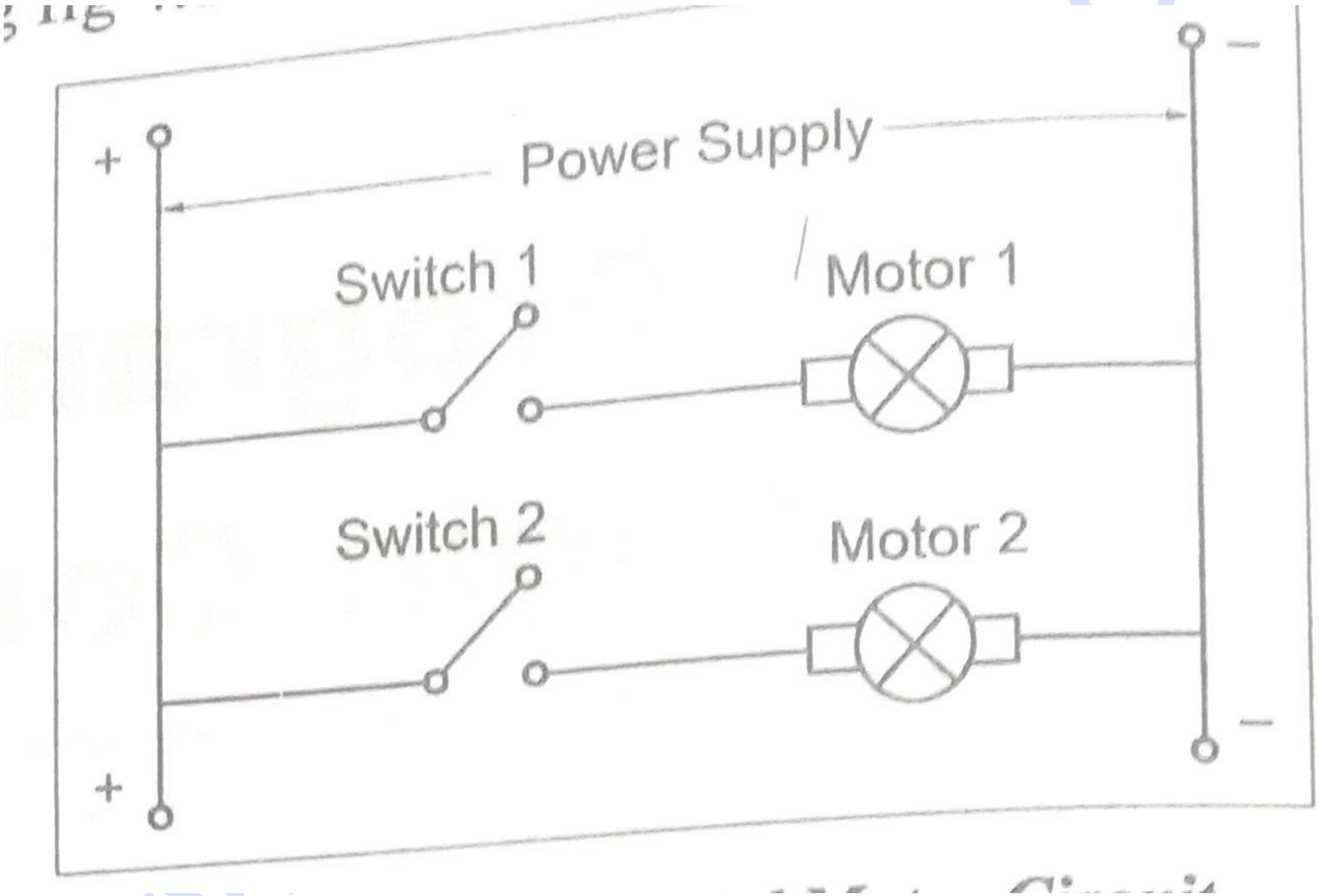
Applications

- Automated manufacturing process equipment and machinery
- Packaging and filling equipment
- Chemical mixing
- Conveyor systems and distillation etc.,

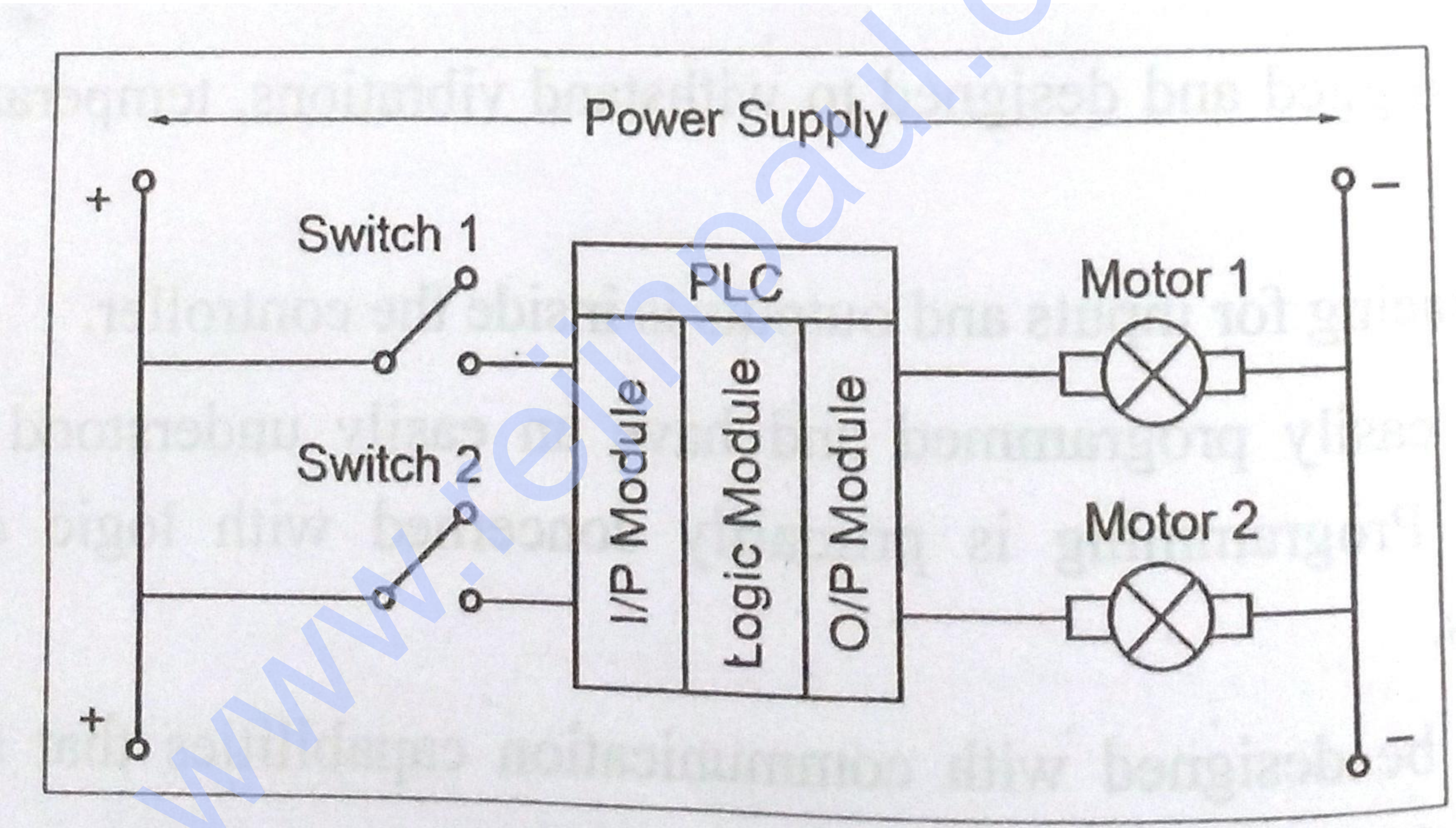
Features and specification

- They are rugged and designed to withstand vibration, temperature, humidity and noise
- The interfacing for inputs and outputs is inside the controller.
- They are easily programmed and have an easily understood programming language.
- Programming is primarily concerned with logic and switching operation.

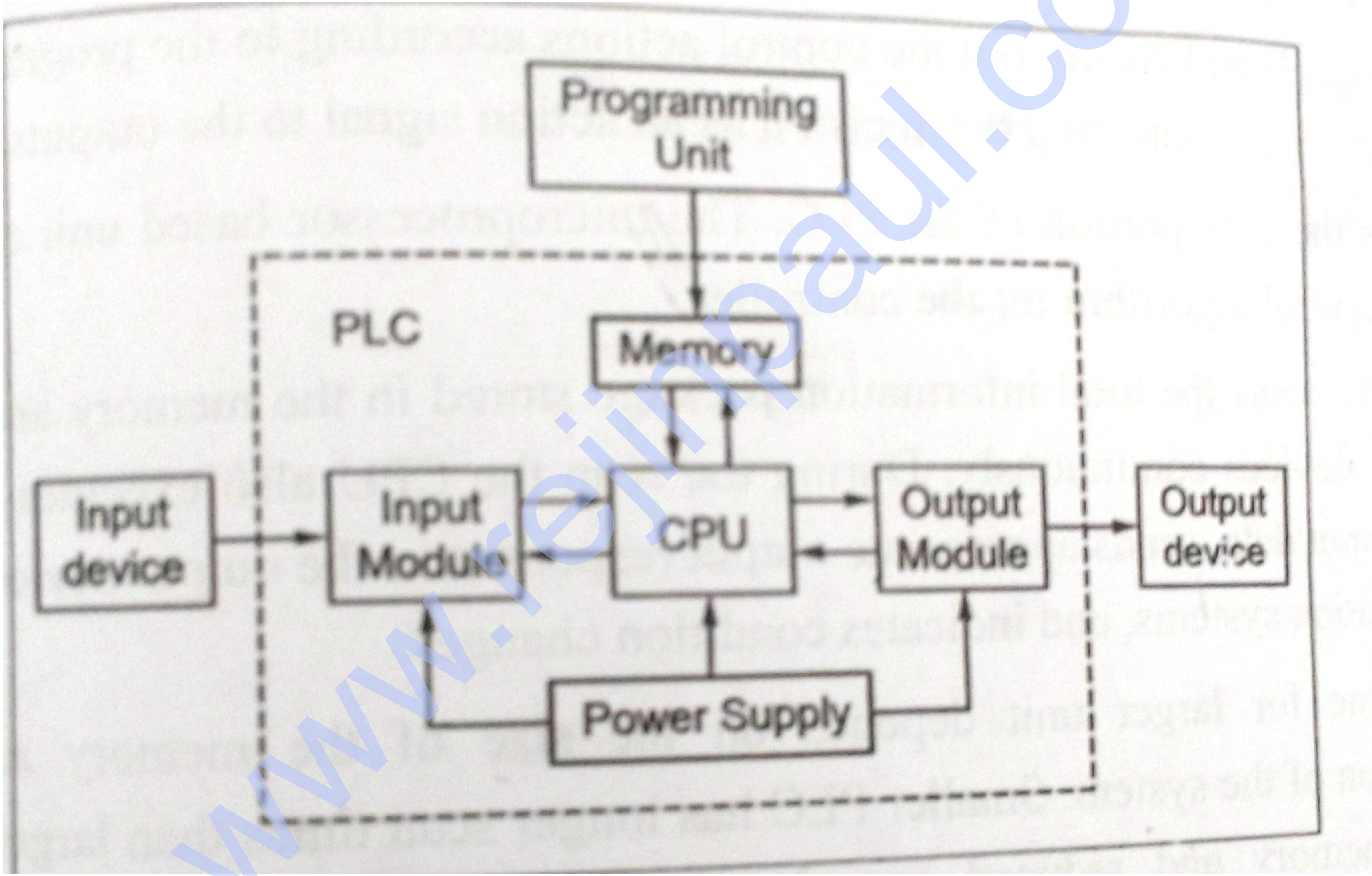
Hardwired motor circuit



Hardwired motor circuit with PLC



Basic structure



- PLC is designed as a replacement for the hardwired relay and timer logic, where PLC provides ease and flexibility of control based on programming and executive logical instruction.
- The internal functions such as timers, counter and shift registers making sophisticated control possible using even the smallest PLC.

- PLC capable of performing function such as
  - counting,
  - logistics,
  - numerical application,
  - comparing and processing of signals.
- A PLC is divided in to 4 parts. They are
  - Input/output module (I/O)
  - Central processing Unit (CPU)
  - Memory
  - Programming unit

i) Input/output module (I/O)

- It is used to transfer the data between external devices and CPU
- It is incorporated into PLC in two ways
  I. Fixed I/O – it is a small unit that comes in one piece with processor i.e., the I/O terminals cannot be changed in fixed I/O
  II. Modular I/O – it is packed together i.e., there are several compartment of I/O module are plugged together.

Central processing Unit (CPU)

- It is consisting of a microprocessor which interrupts the input signal and carries out the control actions according to the program stored in the memory, communicating the decision as an action signal to the output.
- It scan the total information package stored in the memory and input and output devices continuously.
- During the scan the CPU executes instruction based on input data, sends appropriate output responses to the output devices, updates data acquisition systems and indicate condition changes

- Scan time for larger unit depends on the size of the memory and configuration of the system
- Power supply unit is needed to convert the AC voltage to the low DC voltage necessary for the processor and to supply power to other circuit in the input and output interface module.

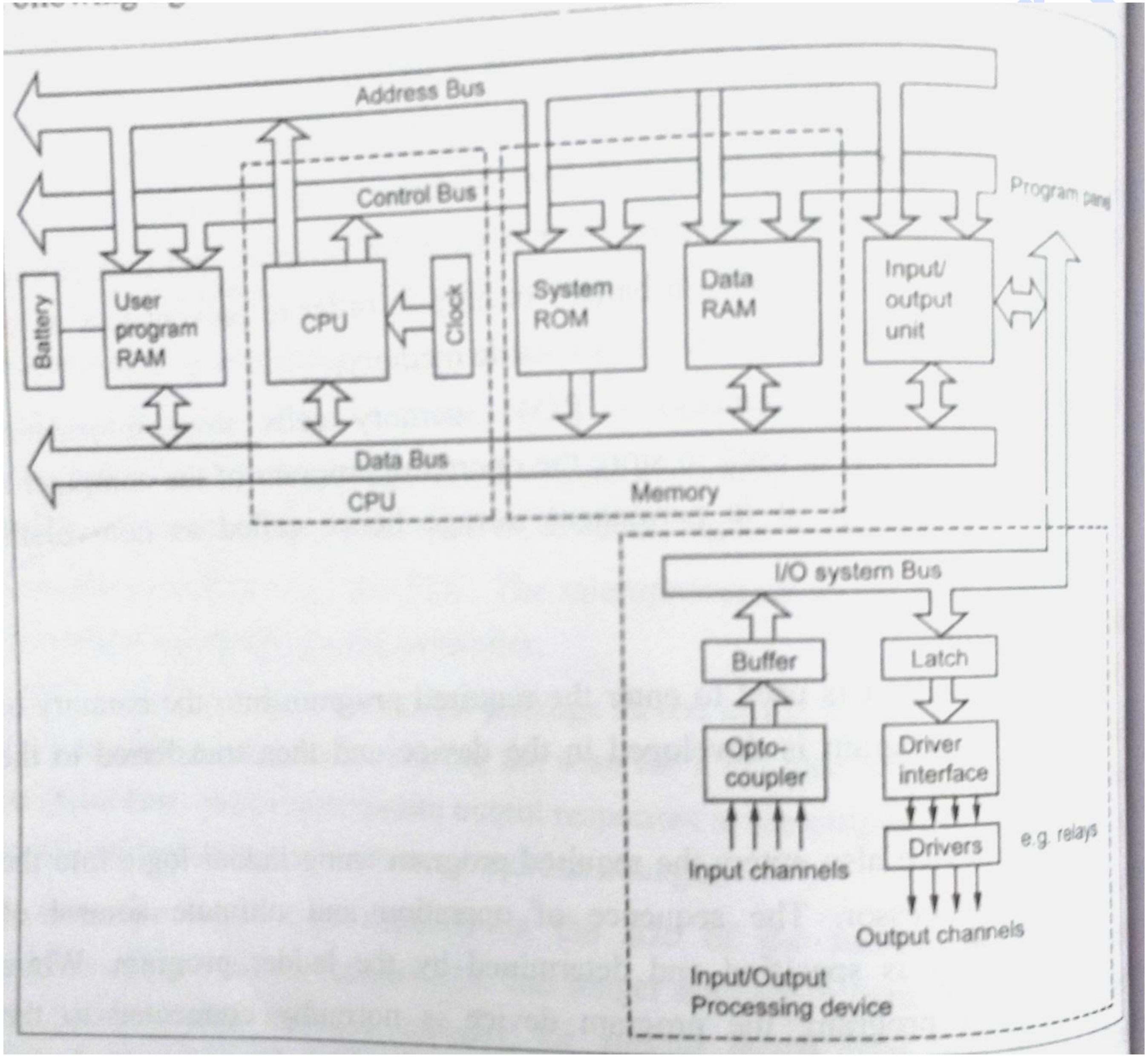
Memory Unit

- The memory in PLC stores the digital control logic, the process program and the necessary instruction to operate the system.
- The memory used in PLC are
- Non-volatile memory
- Volatile memory
- According to purpose of usage
- RAM –volatile memory
- ROM- permanent storage

Programming unit

- It is used to enter the required program into the memory of the processor
- There are normally 3 approaches followed by the program
  - Use of hand held programmer
  - Terminal with video display unit
  - PC with appropriate software

Architecture



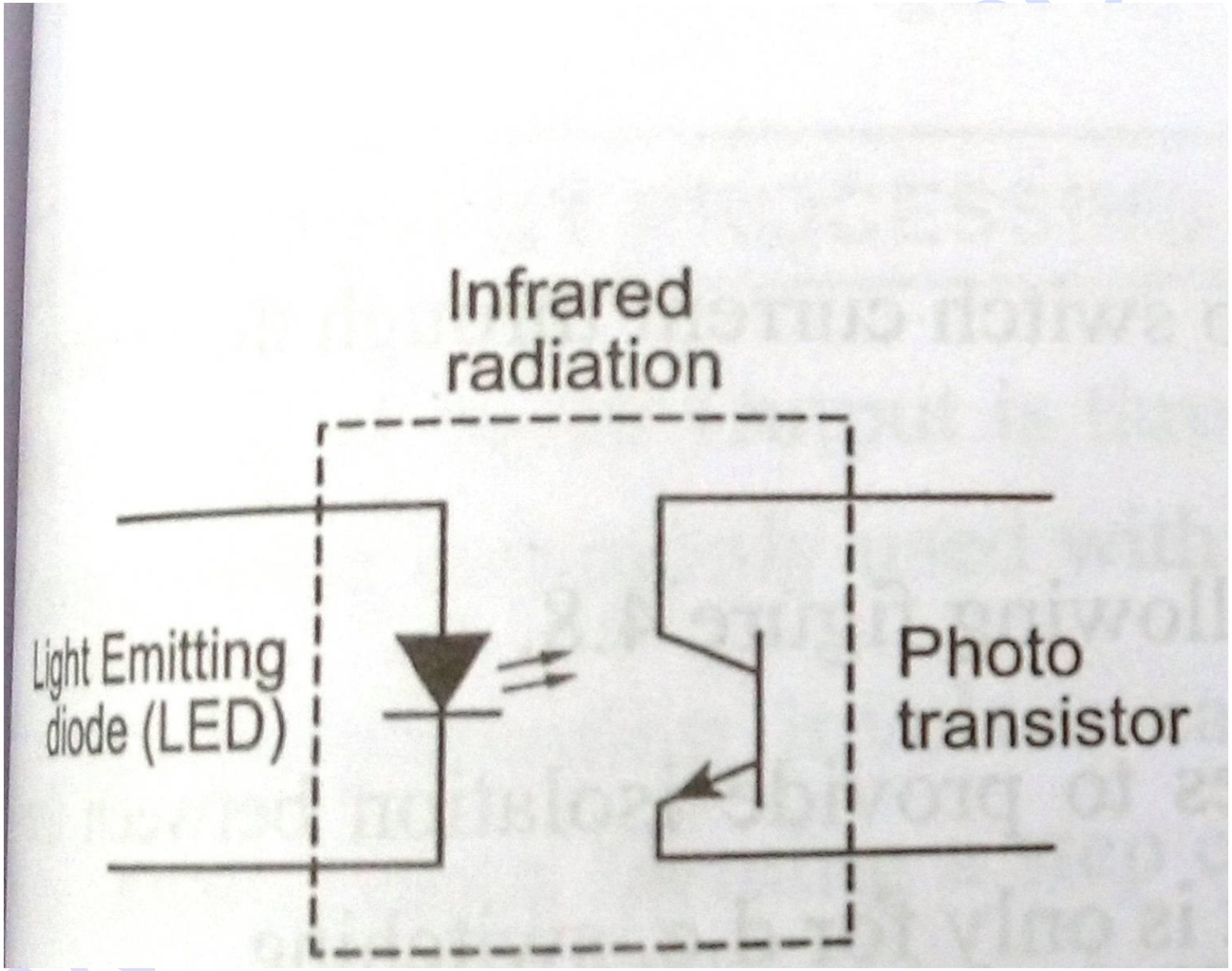
- Buses
  - Data buses – it is used for communicating data b/n elements
  - Address buses-it is used to read the address of locations for accessing stored data
  - Control buses- it is used for internal control action carried by the CPU
  - System buses- it is used for communication b/n Input/output ports and input/output units

- Memory
  - RAM
  - ROM
    - PROM
    - EPROM
    - Electrically EPROM

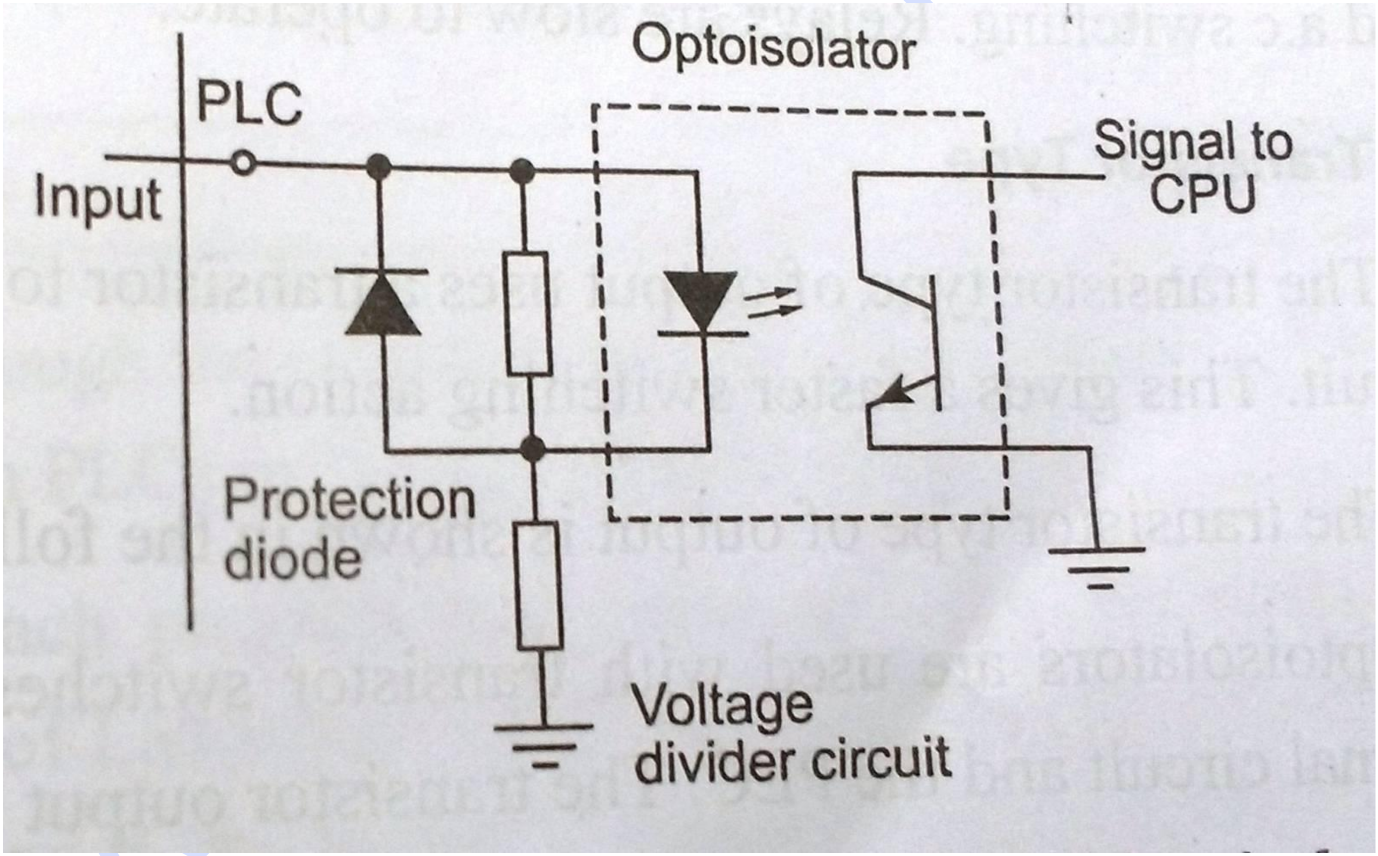
Optoisolator

- Electrical connection from the external world is usually by means of optoisolator
- When a digital pulses passes through the LED, a pulse of Infrared radiation is produced.
- This pulses is detected by the phototransistor and gives rise to a voltage in that circuit.

Optoisolator

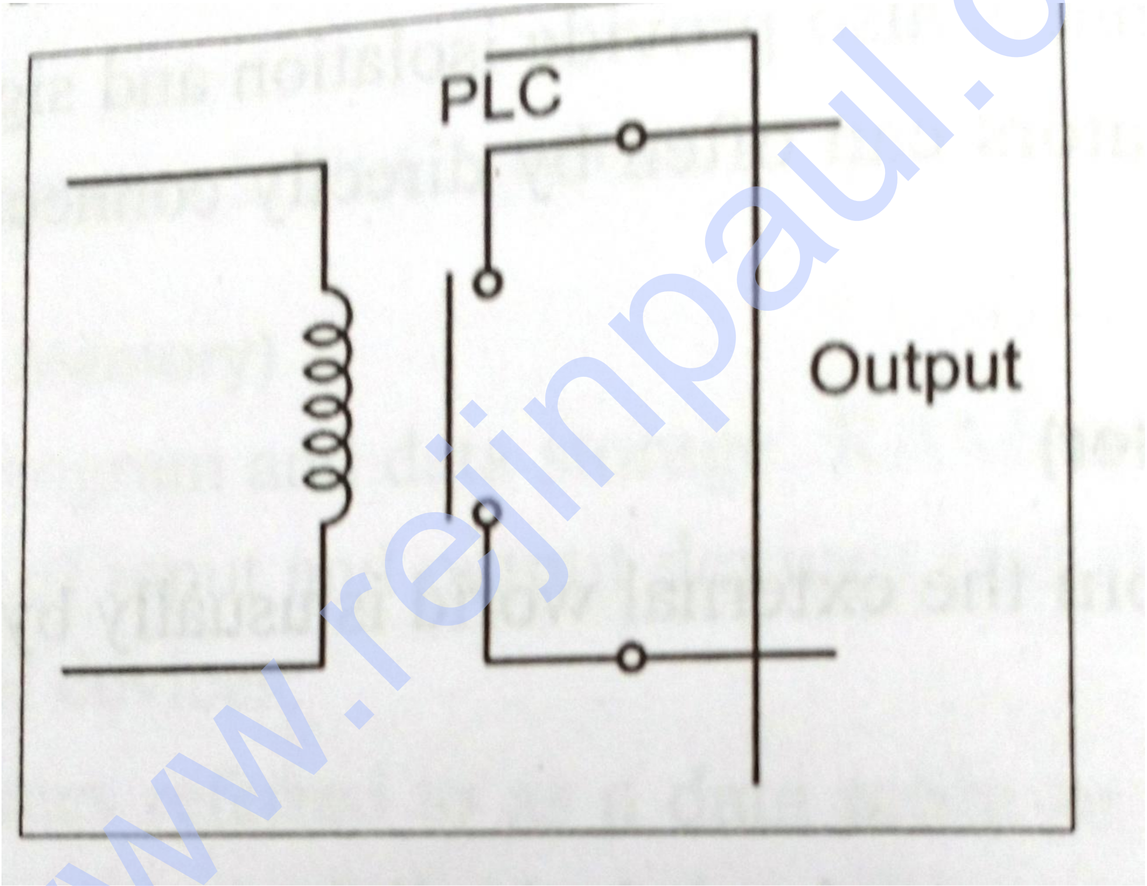


Input channel with optoisolator



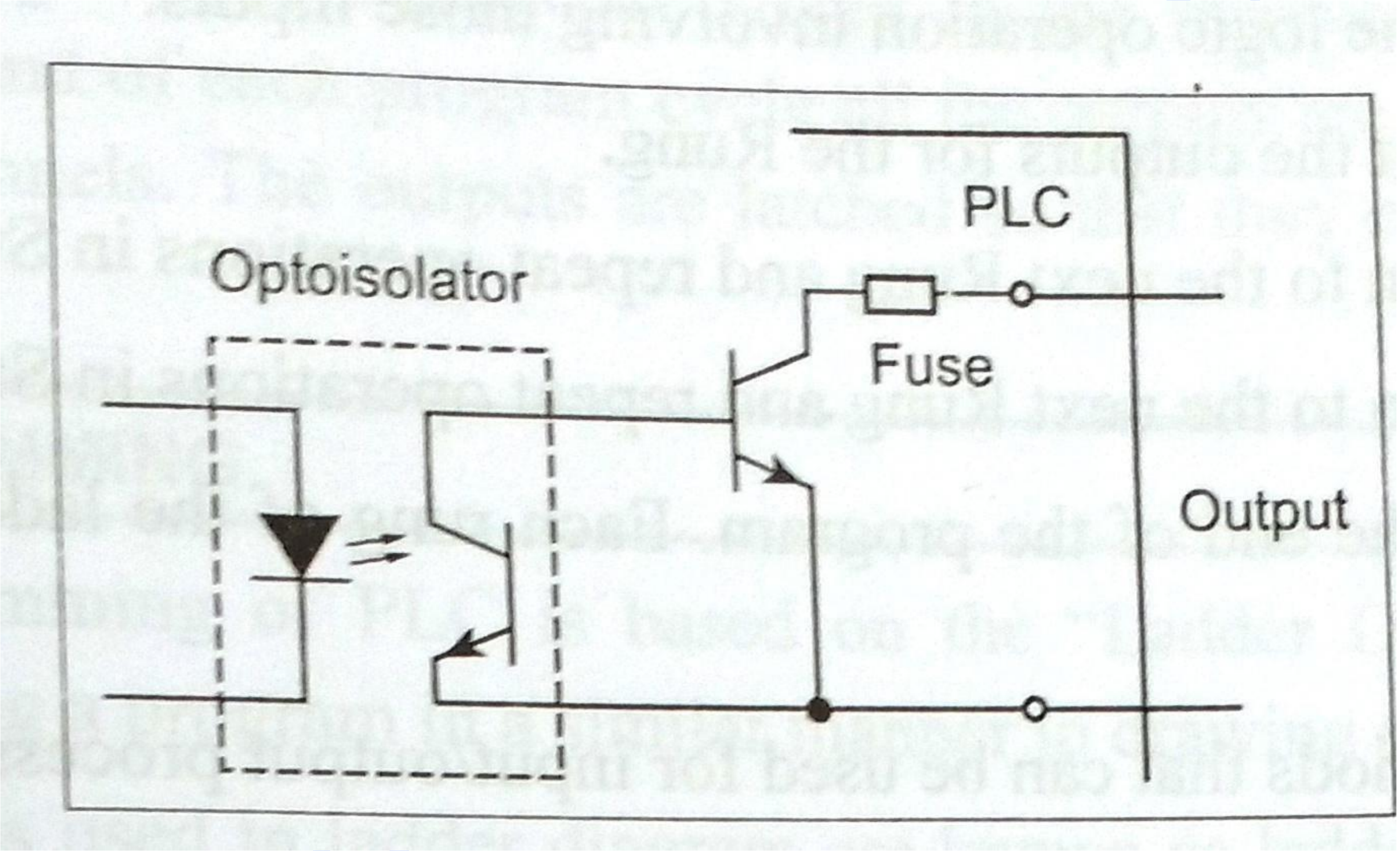
- Common input voltage is 5V and 24V
- Output voltage is 24V and 240V
- Output are often specified as being of
  - Relay type
  - Transistor type
  - Triac type

Relay type of output



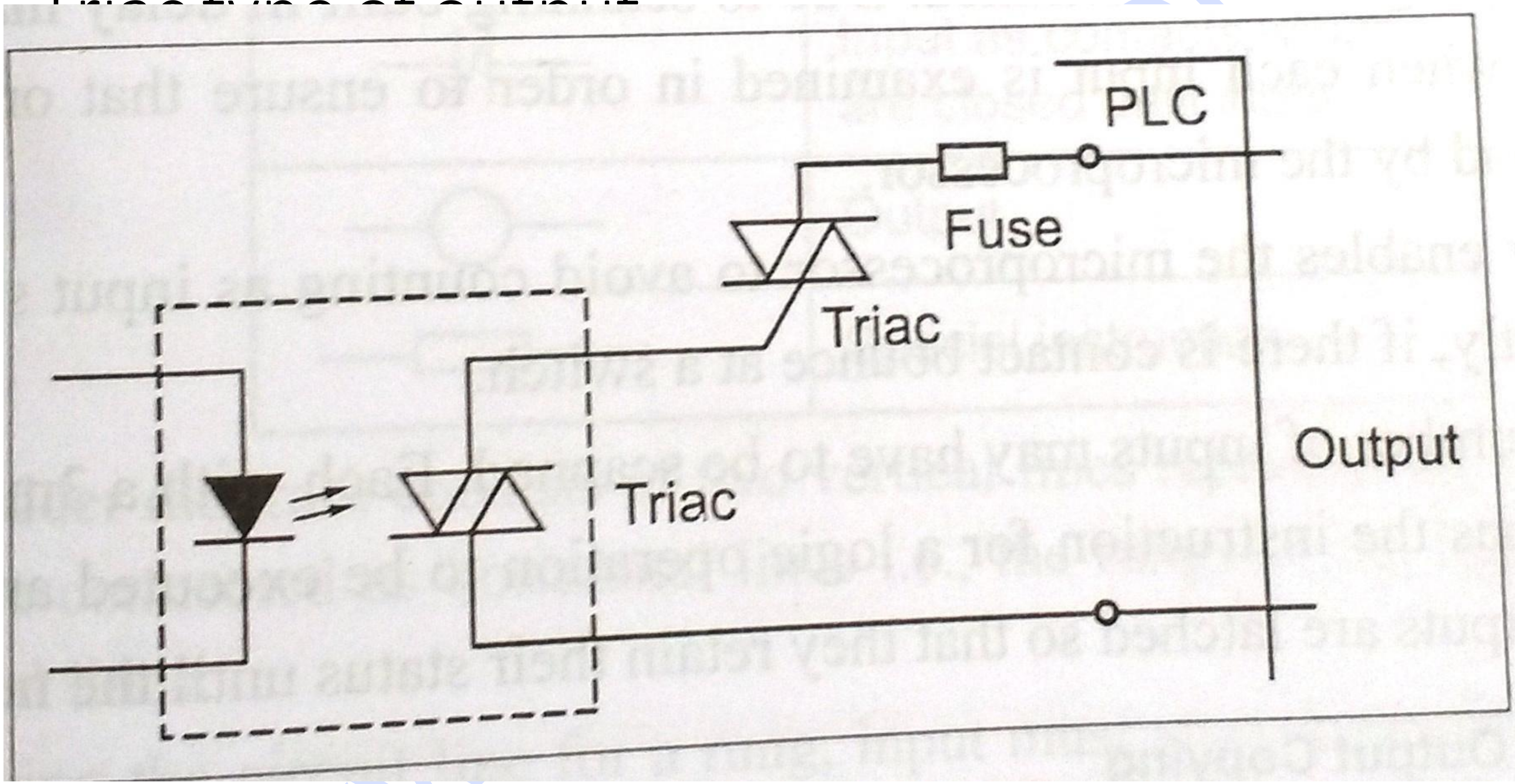
The relay type output is used for both ac and dc switching
Relay are slow to operate

Transistor type output



The transistor type output is used for dc switching
This give faster switching action

Triac type of output

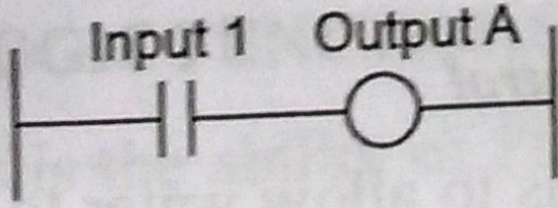


The triac type output is used for switching AC voltages

Programming

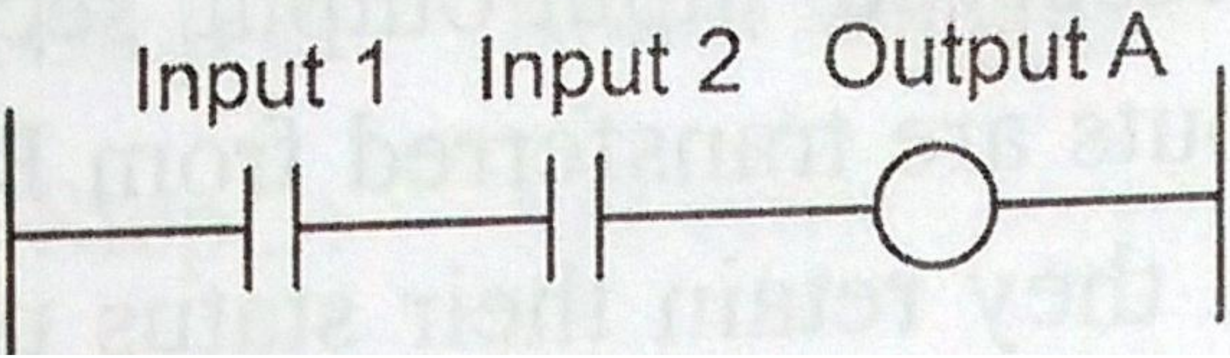
- The programming of PLC is based on the ladder diagram.
- Ladder diagram involve writing a program in a similar maner to drawing a switching circuit.

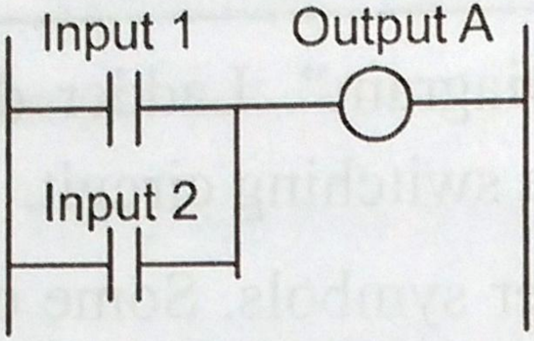
Ladder Symbol	Description
—| |—	Input as contacts but not closed until input
—|/|—	Input as contacts which are closed until input
—( )—	Output
—[ ]—	Special instruction

i) 

⇒ Output A occurs when input 1 occurs

Here the two vertical lines are called power rails.

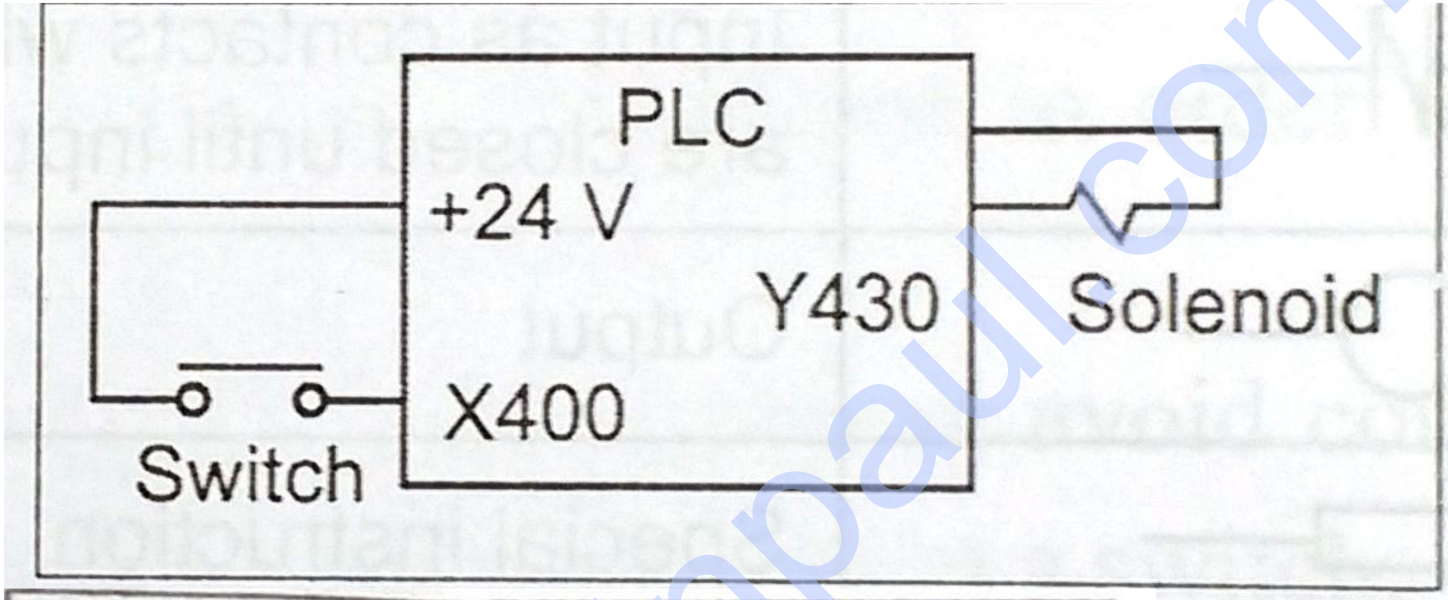
ii) 

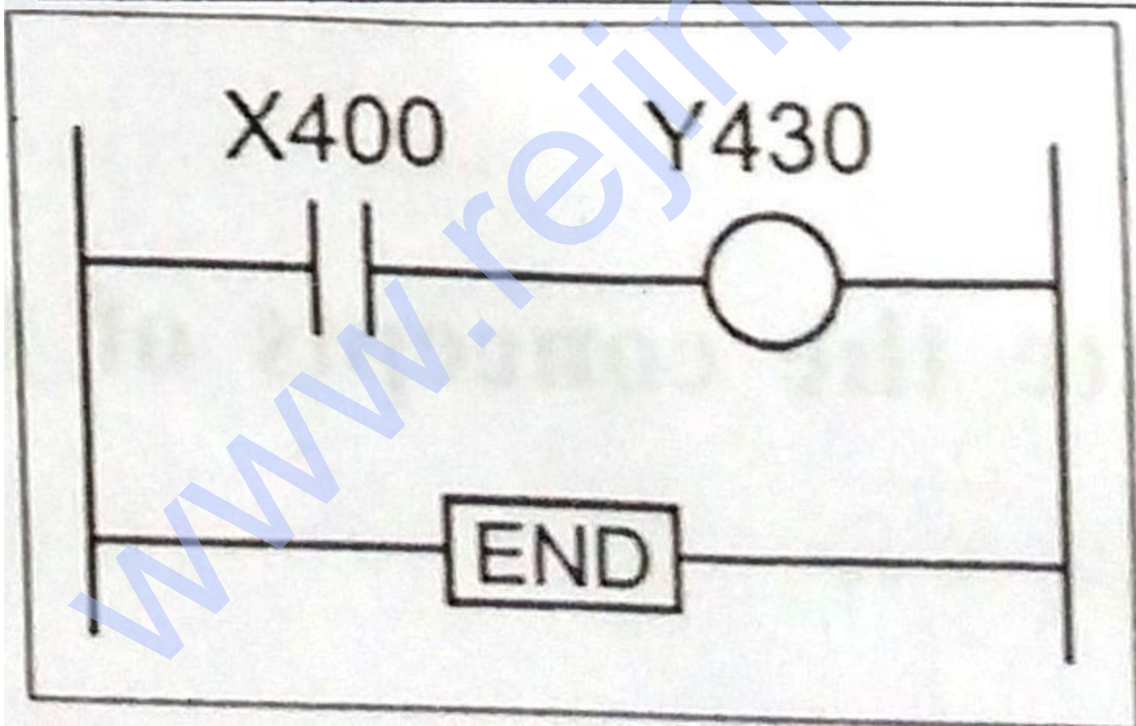
iii) 

⇒ Output A occurs when input 1 or input 2 occurs

Here the ladder diagram describes the output A when occurs.

Switch controlling Solenoid

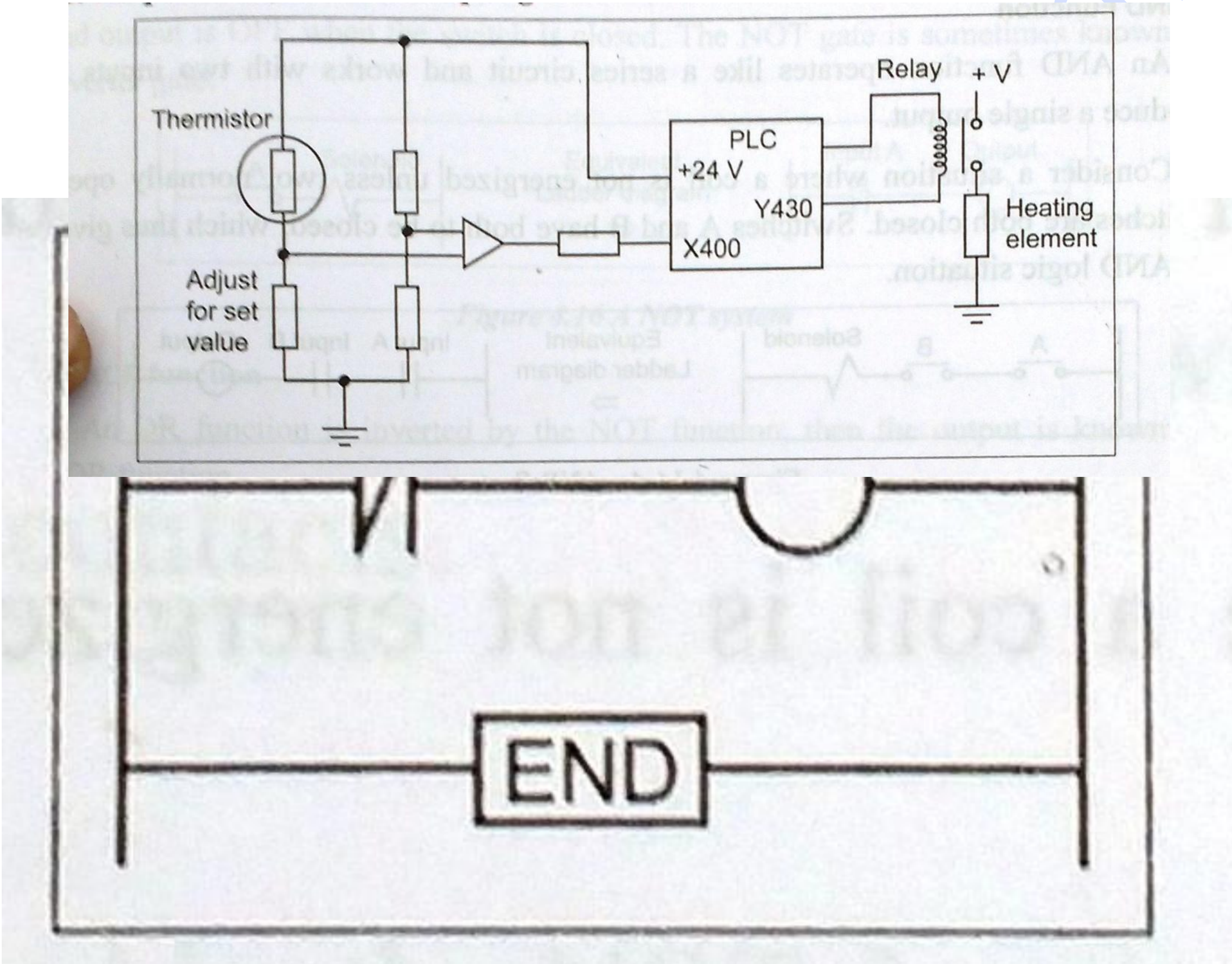




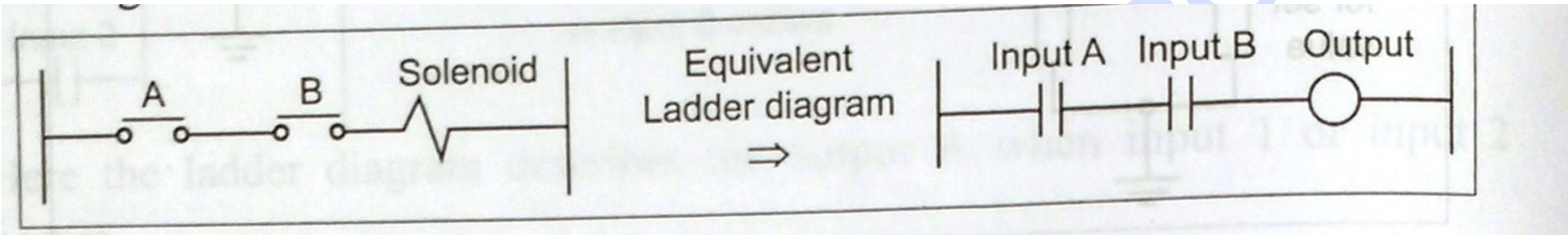
Ladder Diagram for Switch Control

e.g., solenoid valve open to allow water to enter a vessel

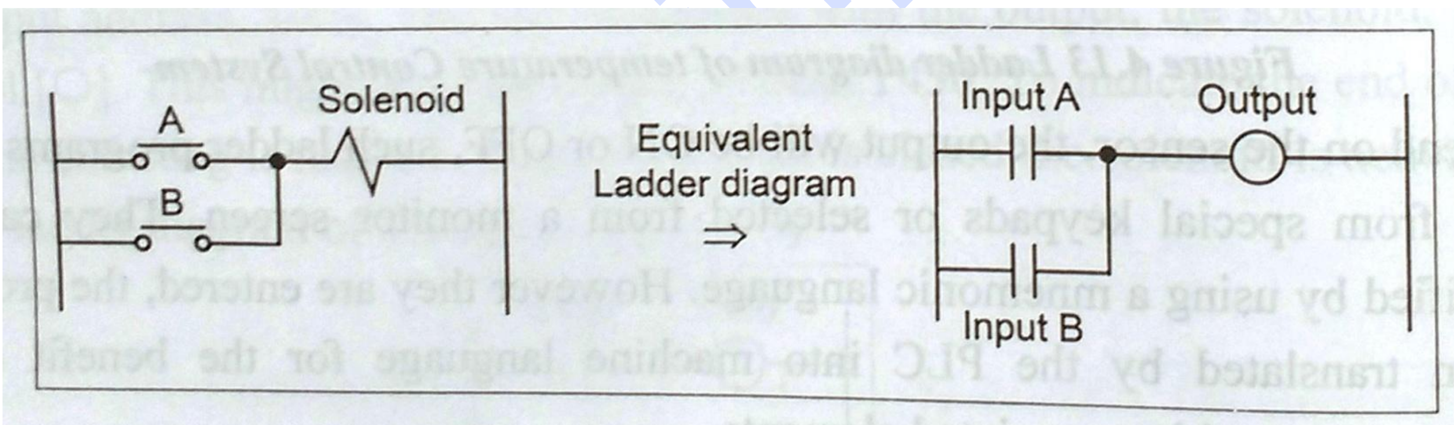
Temperature Control System



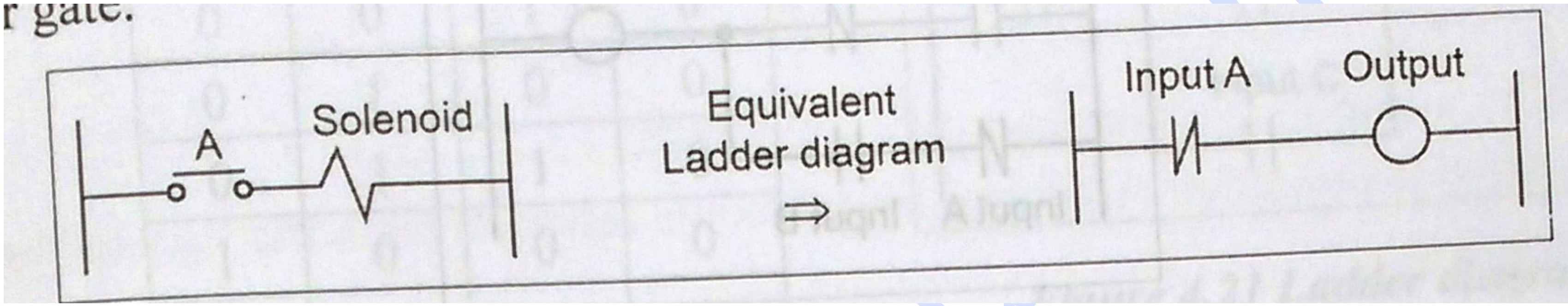
Logic functions



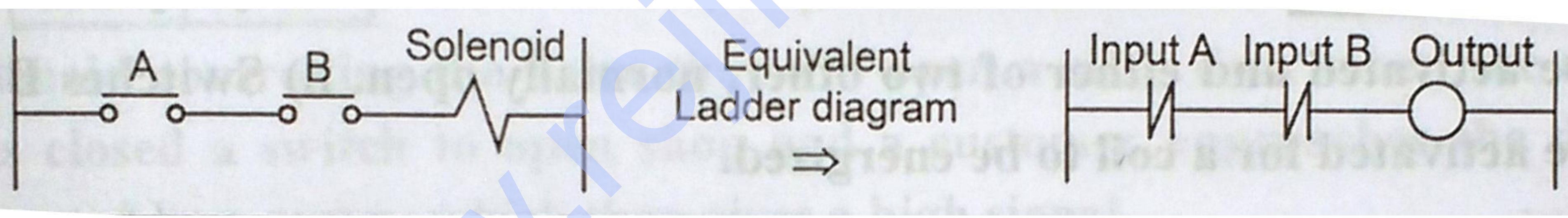
An AND System



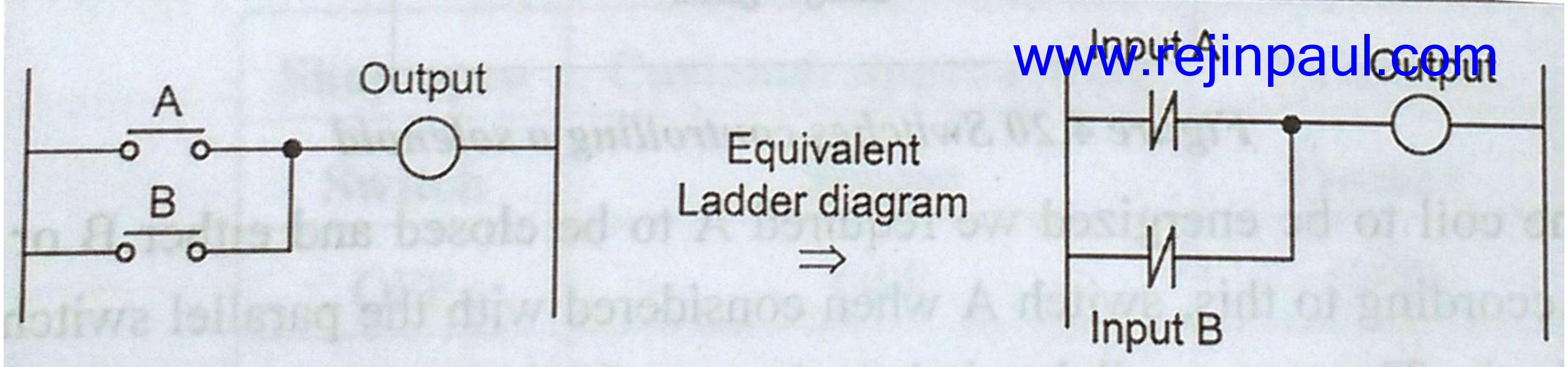
An OR System



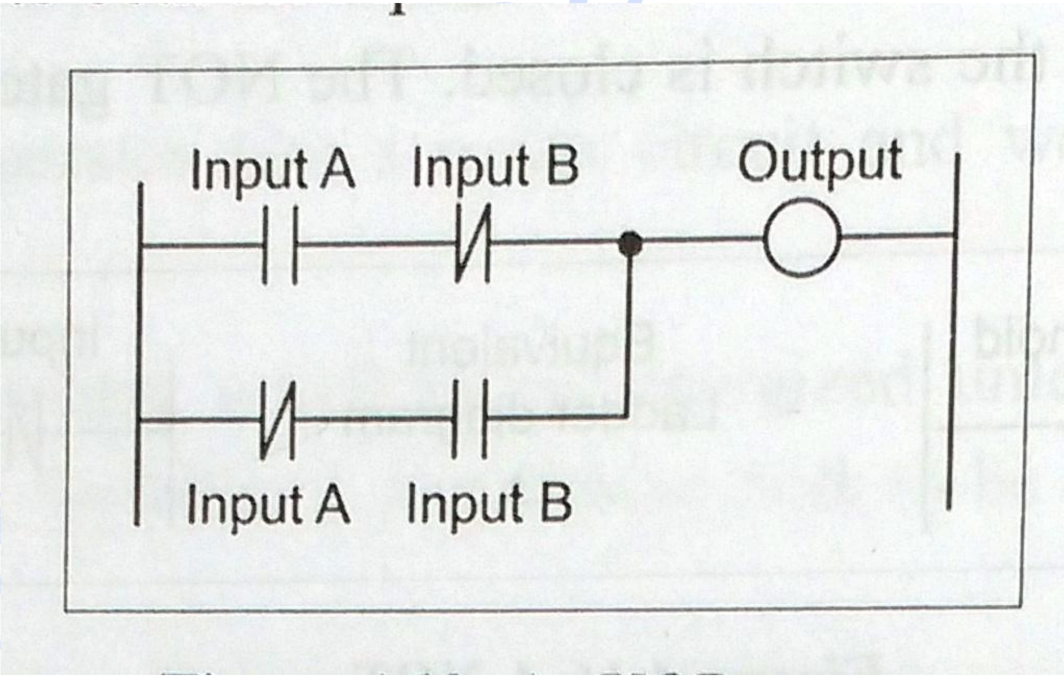
NOT System



NOR System

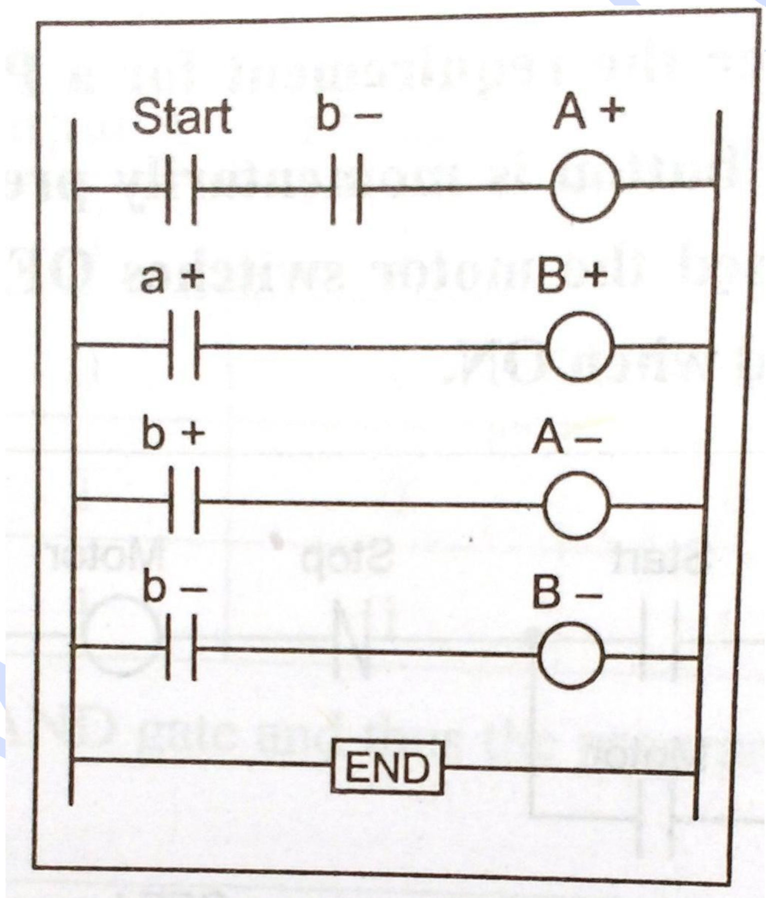


NAND System



XOR System

Cylinder Sequencing
A+, B+, A- and B-



List of Mnemonics used for the Mitsubishi f Series PLC

Mnemonics	Comment
LD	Start a rung with an open circuit
OUT	An output
AND	A series element and so an AND logic instruction
OR	Parallel elements and so an OR logic
I	A NOT logic
...I	Used in conjunction with other instruction to indicate the inverse
ORI	An OR NOT logic function
ANI	An AND NOT logic function
LDI	Start a rung with a closed contact
ANB	AND used with two sub circuits
ORB	OR used with two sub circuits
RST	Reset shift register/ counter
SHF	Shift
K	Insert a constant
END	END ladder

Mnemonics for Logic system

i) AND Logic

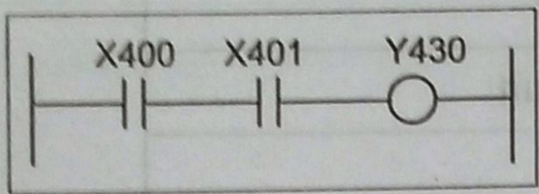


Figure 4.27 Ladder diagram

Step	Instruction	Comment
0	LD X400	Load input at address X400
1	AND X401	AND input X400 at address X401
2	OUT Y430	Output stored to address Y430

ii) OR Logic

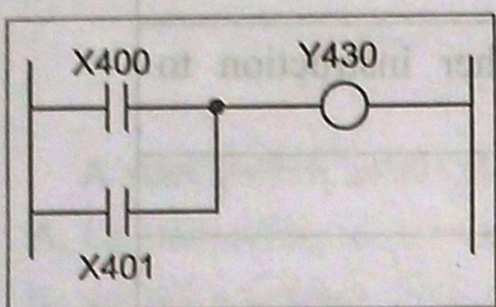


Figure 4.28 Ladder diagram for OR logic

Step	Instruction	Comment
0	LD X400	Load input at address X400
1	OR X401	OR input X400 at address X401
2	OUT Y430	Output stored to address Y430

Mnemonics for Logic system

iii) NOR Logic

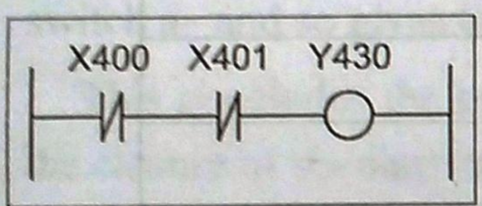


Figure 4.29 Ladder diagram for NOR logic

Step	Instruction	Comment
0	LDI X400	Load closed contact input at address X400
1	ANI X401	AND NOT logic input at address X401
2	OUT Y430	Output stored to address Y430

iv) NAND Logic

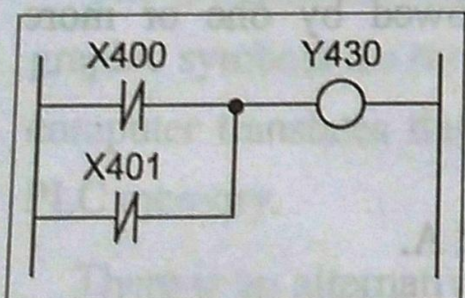
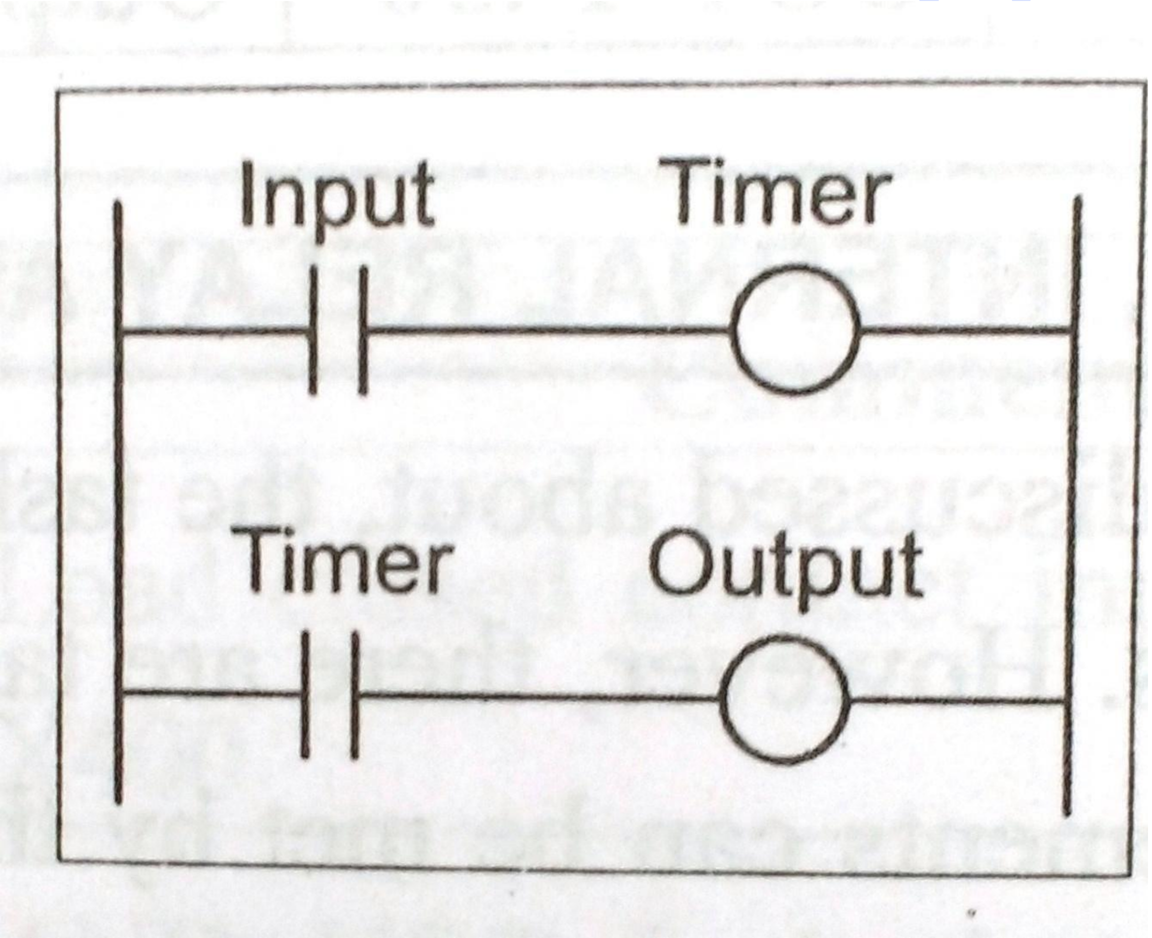


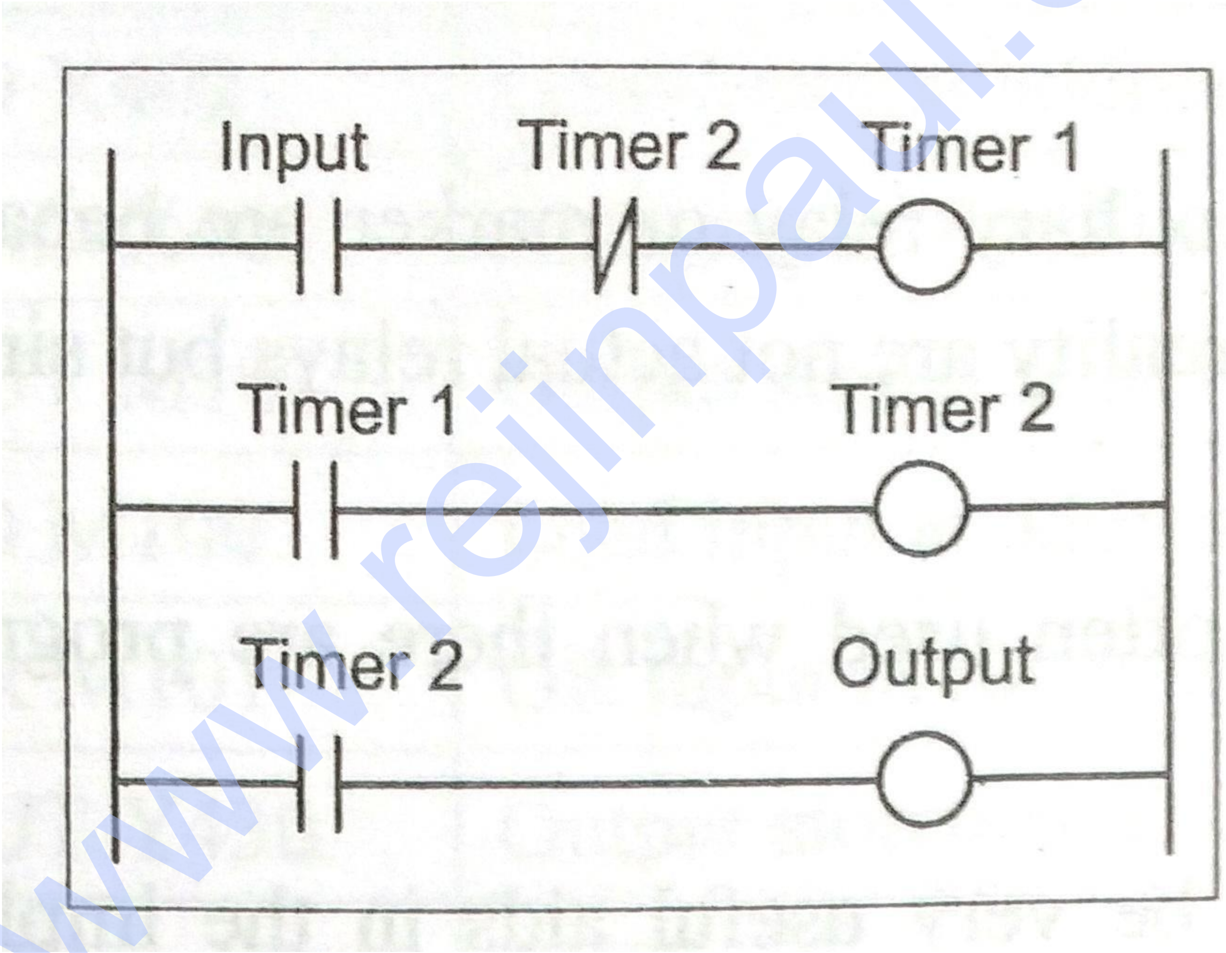
Figure 4.30 Ladder diagram for NAND logic

Step	Instruction	Comment
0	LDI X400	Load closed contact input at address X400
1	ORI X401	OR NOT logic input at address X401
2	OUT Y430	Output stored to address Y430

Timer

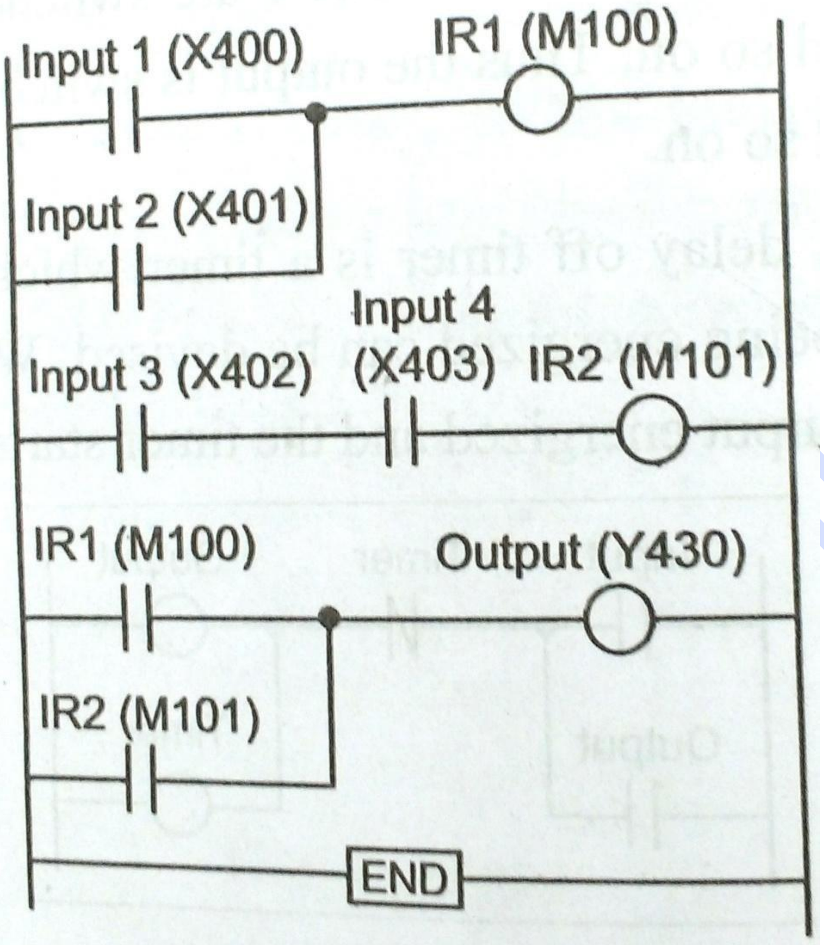


Timer circuit programmed to cause an output to go ON for 0.5s, then OFF for 0.5s, then OFF for 0.5s and so on



ON-OFF cycle timer

Internal relay



Step	Instruction	Comments
0	LD X400	Load input at address X400
1	OR X401	OR input X400 at address X401
2	OT M100	Output stored at M1000
3	LD X402	Load input at address X 402
4	AND X403	AND input X402 at address X403
5	OUT M101	Output stored at M101
6	LD M100	Load input at address M100
7	OR M101	OR input M100 at address M101
8	OUT Y430	Output stored at Y430
9	END	END

Counter

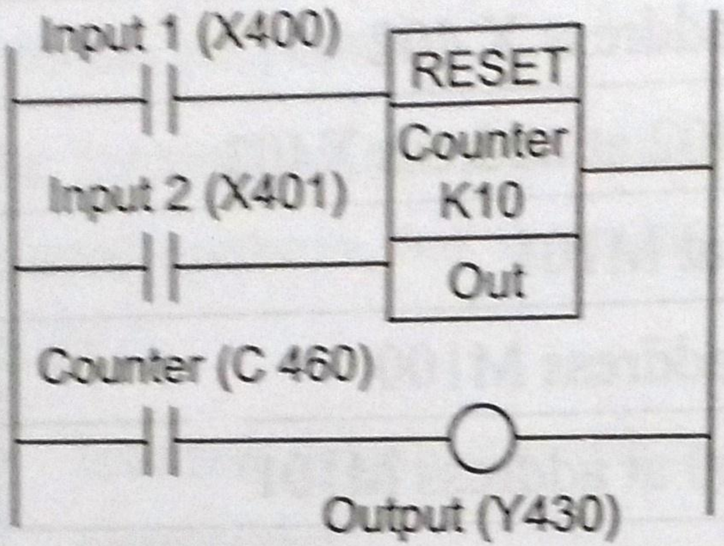
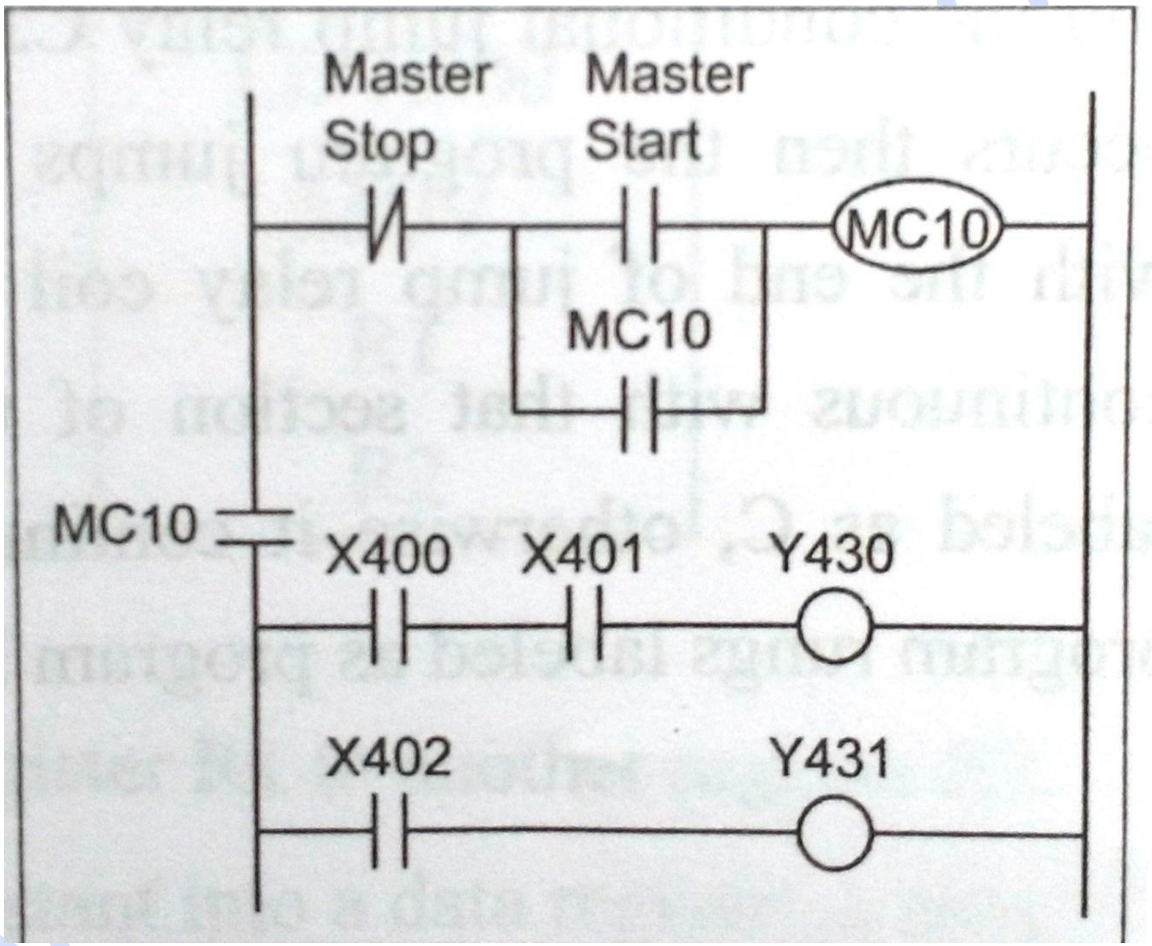


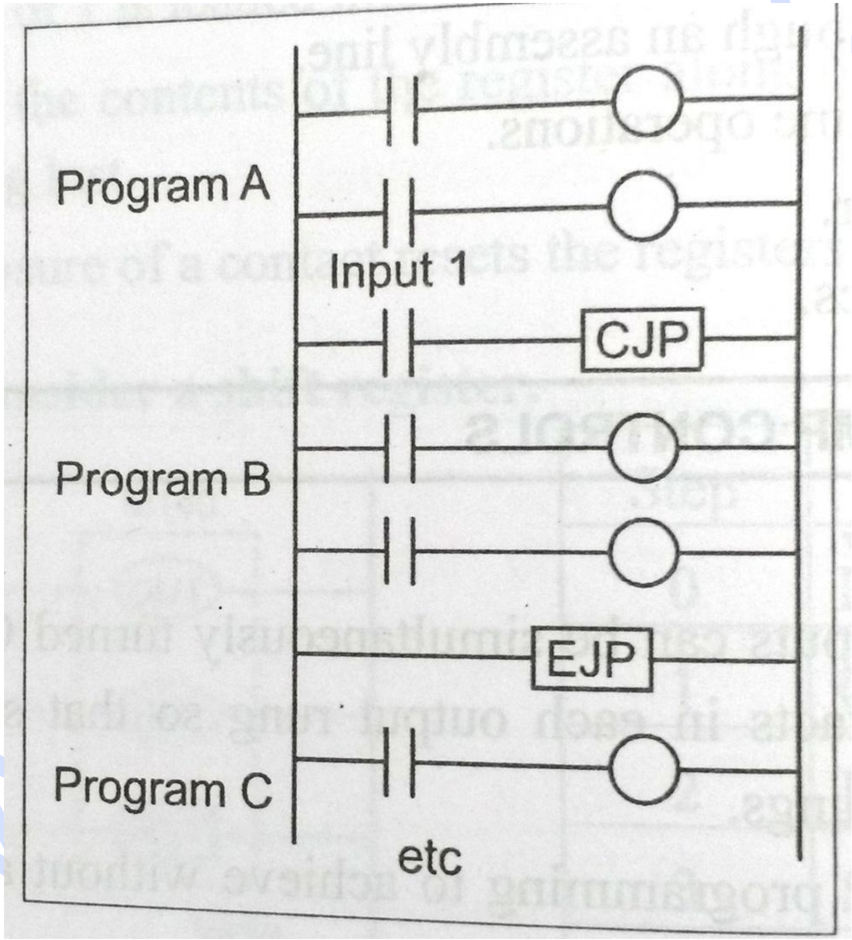
Figure 4.38 Counter

Step	Instruction
0	LD X400
1	RST C460
2	LD X401
3	OUT C460
4	K 10
5	LD C460
6	OUT Y430

Master control relay



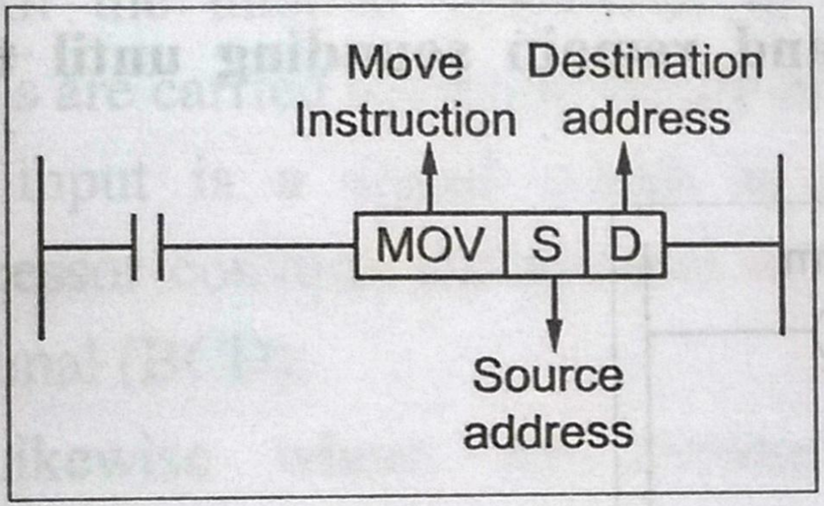
JUMP Instruction



Data handling

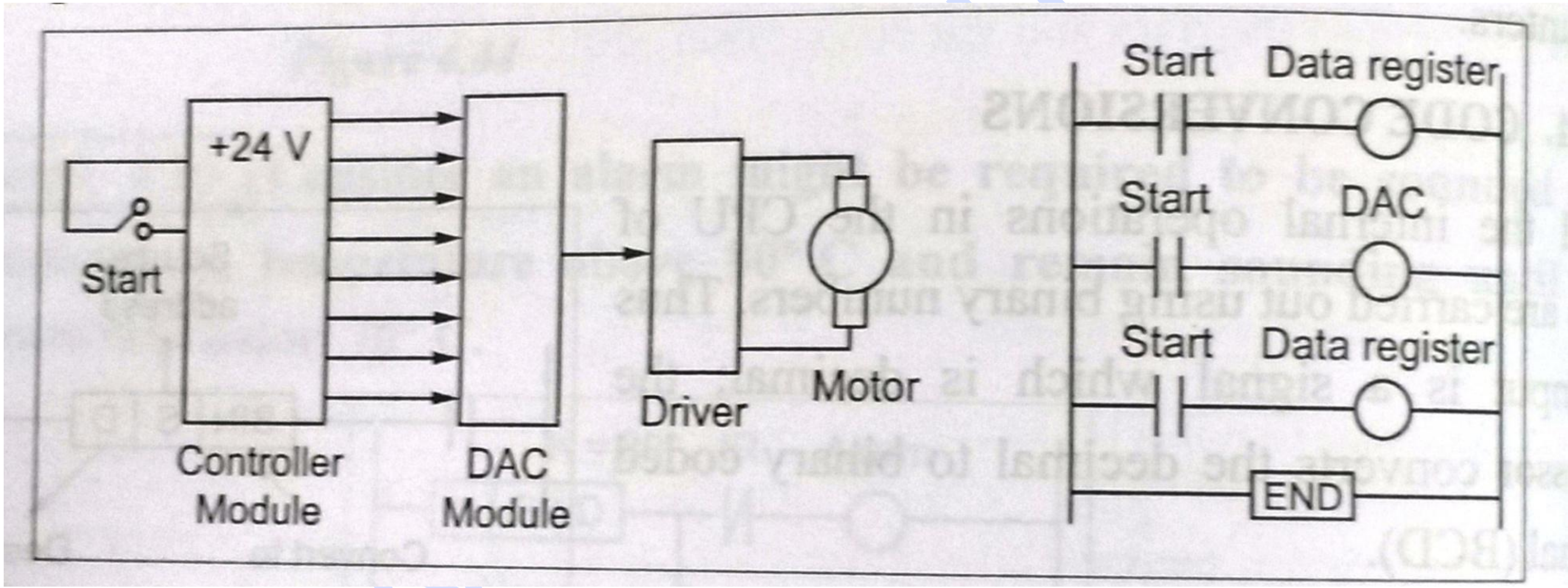
- Data movement
- Data comparison
- Arithmetic operation
- Code conversion

Data Movement



Step	Instruction
0	LD X300
1	MOV
2	R1
3	R2

Controlling the speed of motor



Selection of PLC

- System definition
- Choosing the I/O hardware
- I/O timing consideration
- Analog I/O module –resolution, voltage level
- Conversion speed
- Analog closed control
- Communication
- Counter, encoders and positioning
- Selecting suppliers

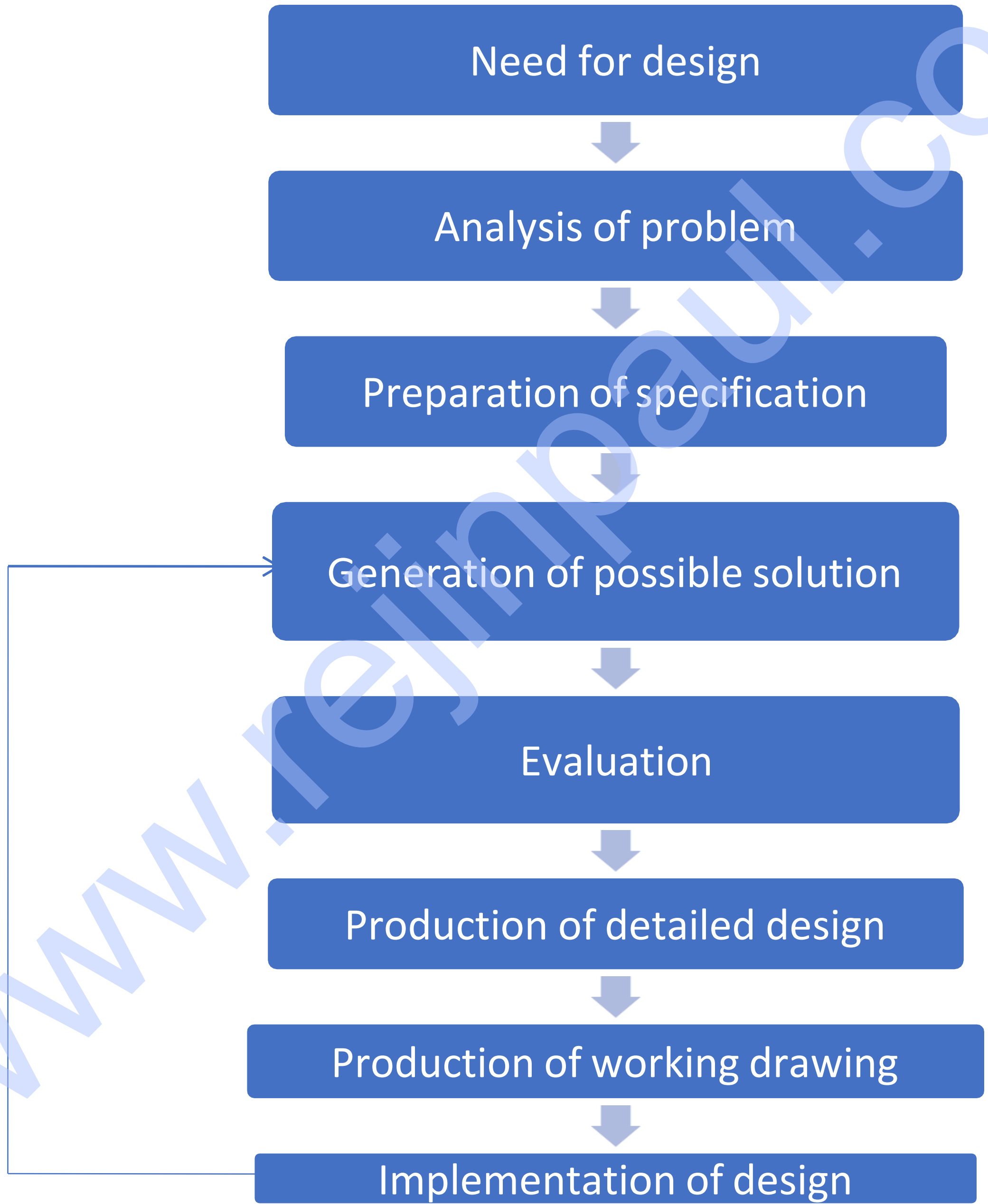
UNIT 5

ACTUATORS AND MECHATRONICS SYSTEMS DESIGN

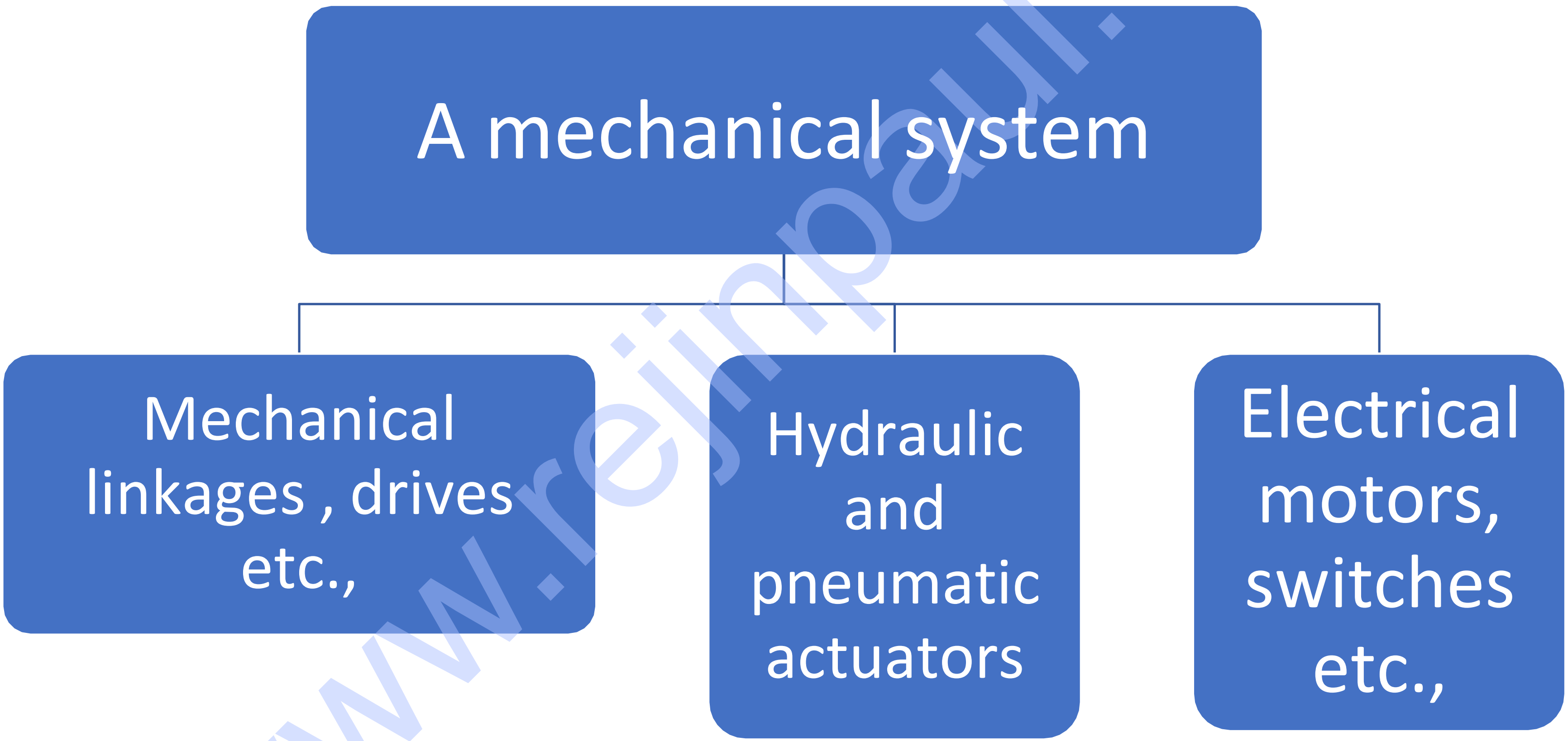
CONTENT

- ❖ Types of stepper and servo motors – construction, working principle
- ❖ Design process - Stages in designing mechatronics system
- ❖ Traditional and mechatronic design concept
- ❖ Possible design solution
- ❖ Case studies of mechatronics systems
  - ❖ Pick and place robot
  - ❖ Engine management system
  - ❖ Automatic car park barrier

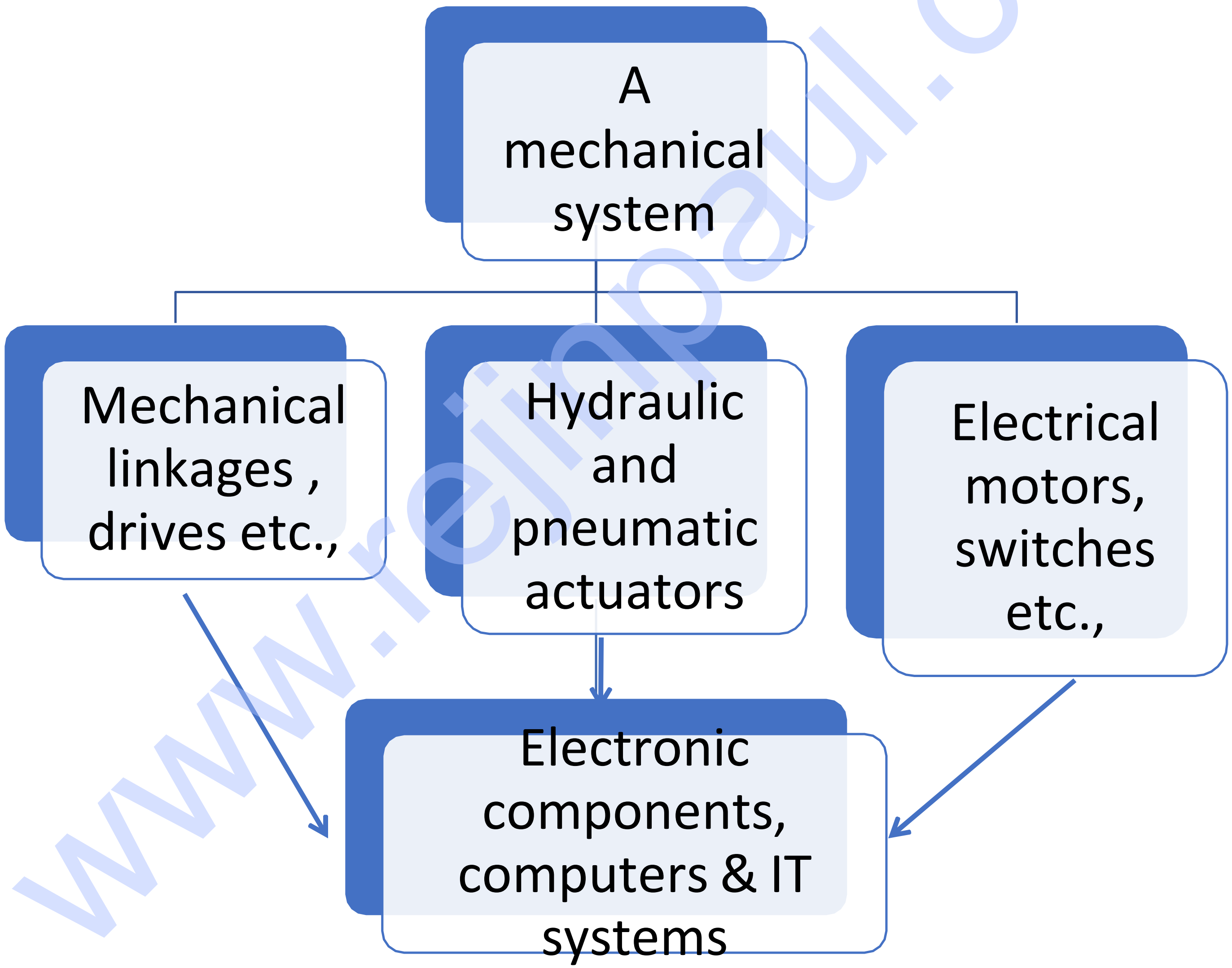
Stages in designing mechatronics system



Traditional design



Mechatronic design



Comparison of traditional and mechatronics design

Traditional design

- It is based on a traditional systems such as mechanical, hydraulic and pneumatic systems
- Less flexible
- Less accurate
- More complicate mechanism in design
- It involve more components and moving parts

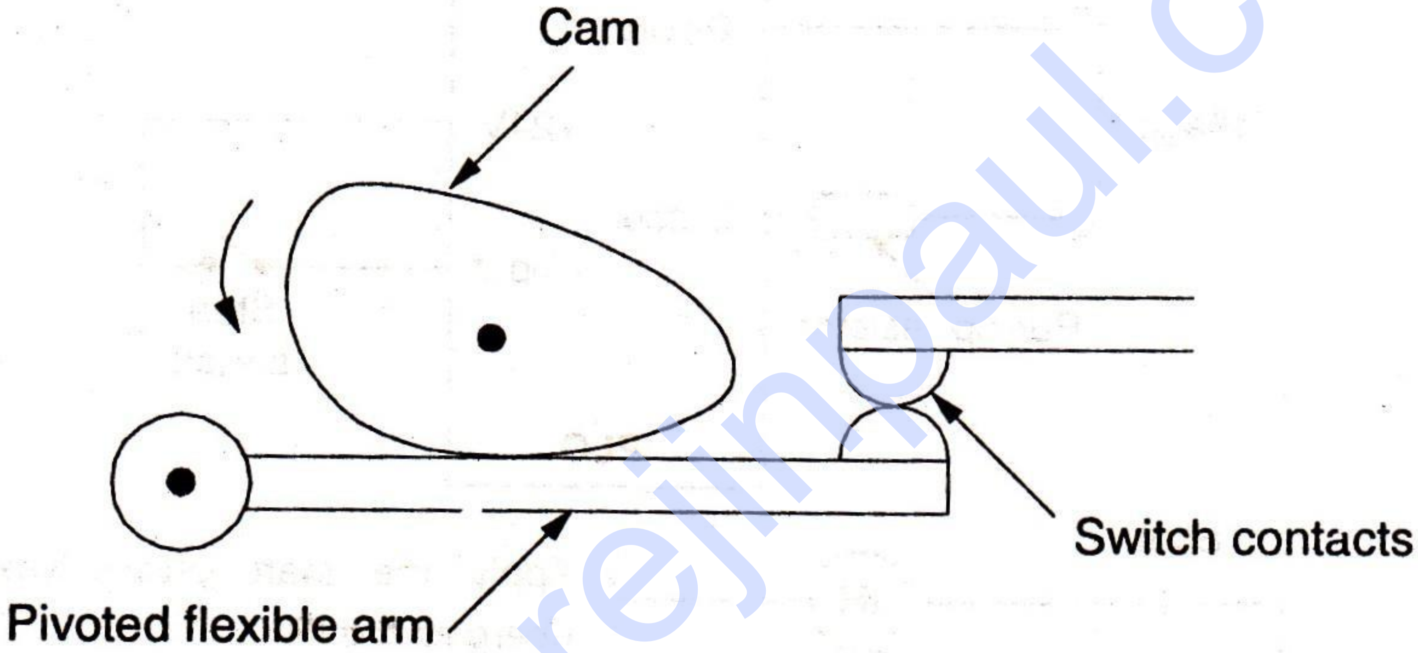
Mechatronics design

- It is based on mechanical, electronics, computer technology and control engineering.
- More flexible
- More accurate
- Less complicate mechanism in design
- It involve fewer components and moving parts

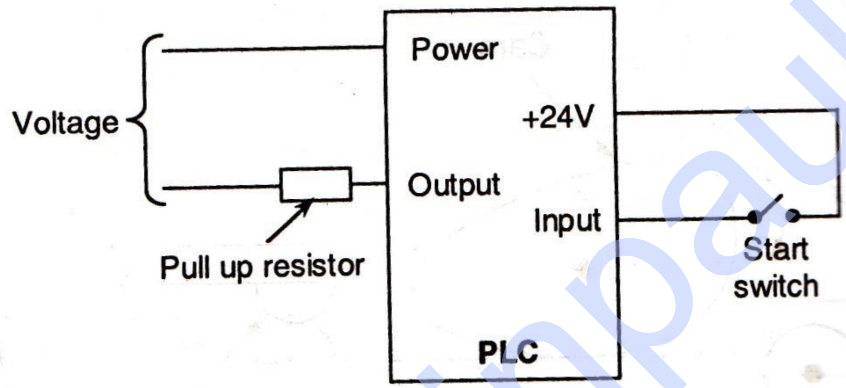
Possible design solution

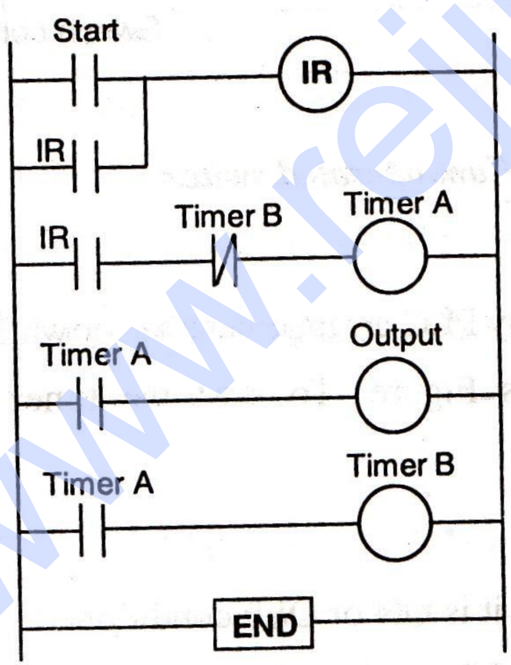
- Timer switch
- Windscreen wiper motion
- Weighing scales

Timer switch



Possible solution for timer switch





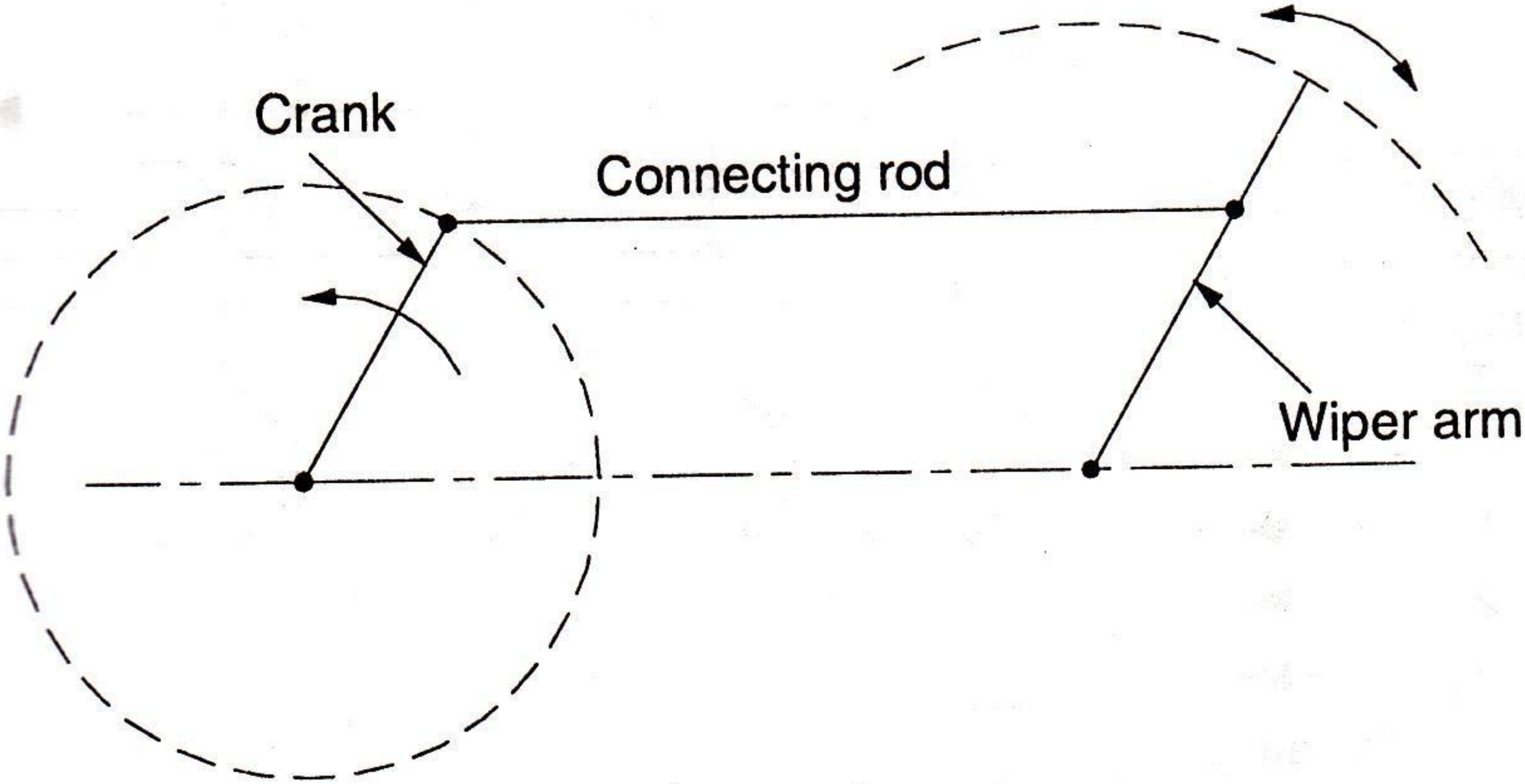
Apply the start pulse which energies the IR

Switch on timer A by IR. Set the timer for which the timer is in ON

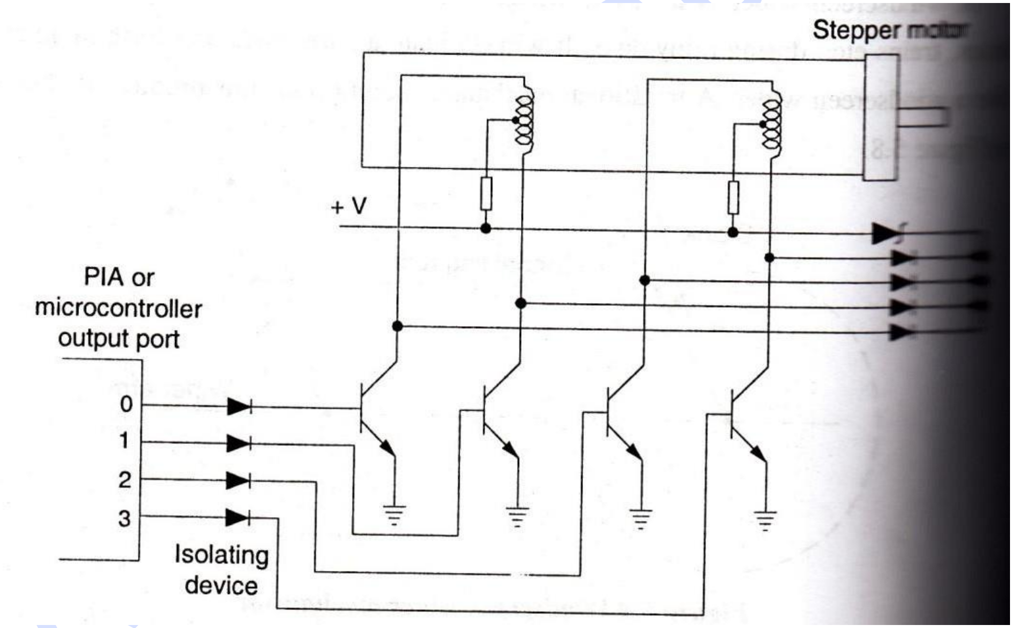
Get the output from timer A

Timer B started when timer A switched ON and determines the time at which output switched OFF

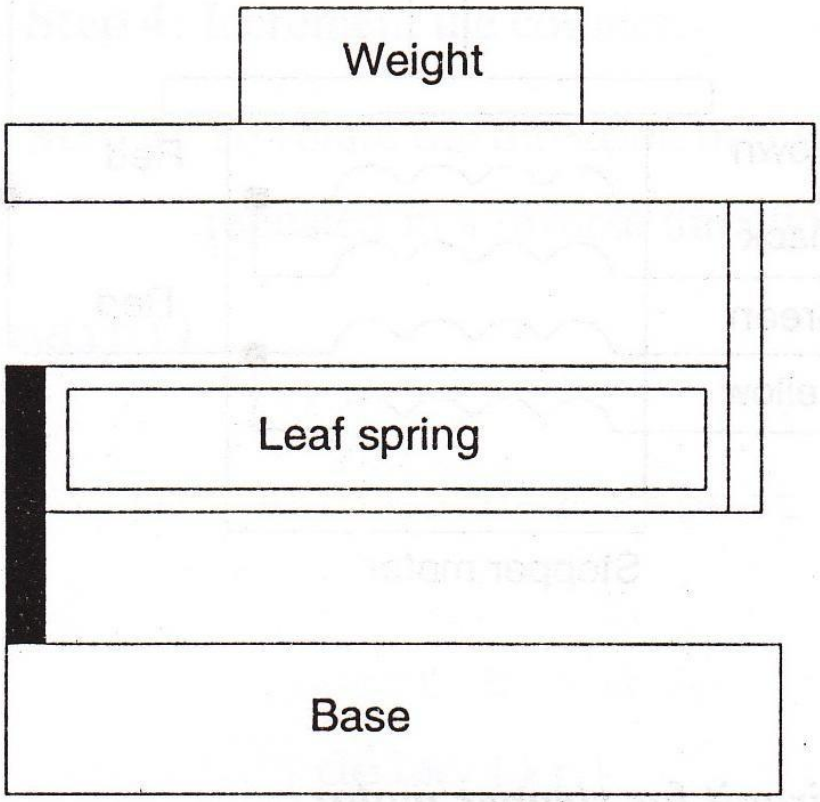
Windscreen wiper motion



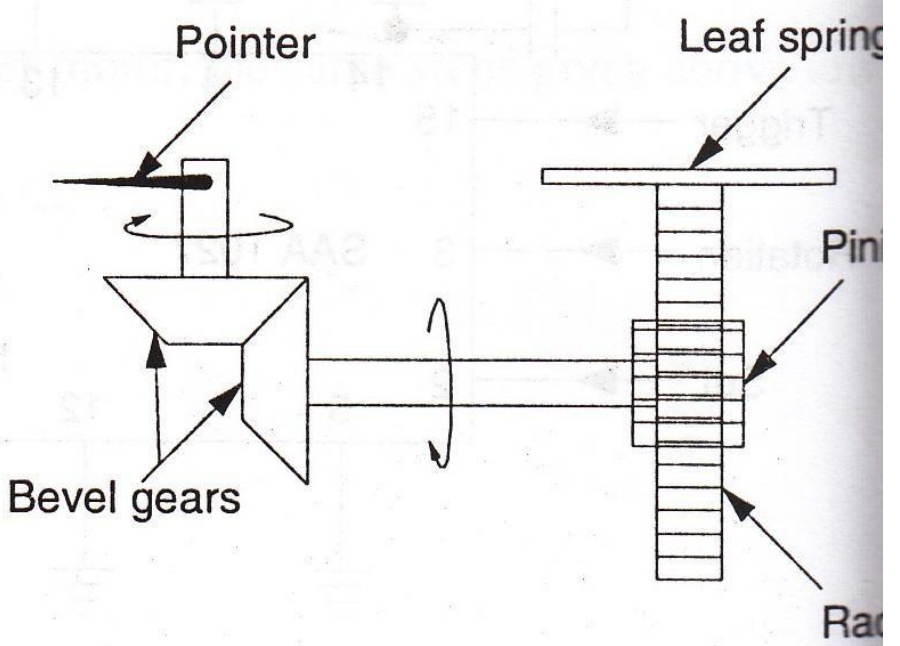
Possible solution for Windscreen wiper motion



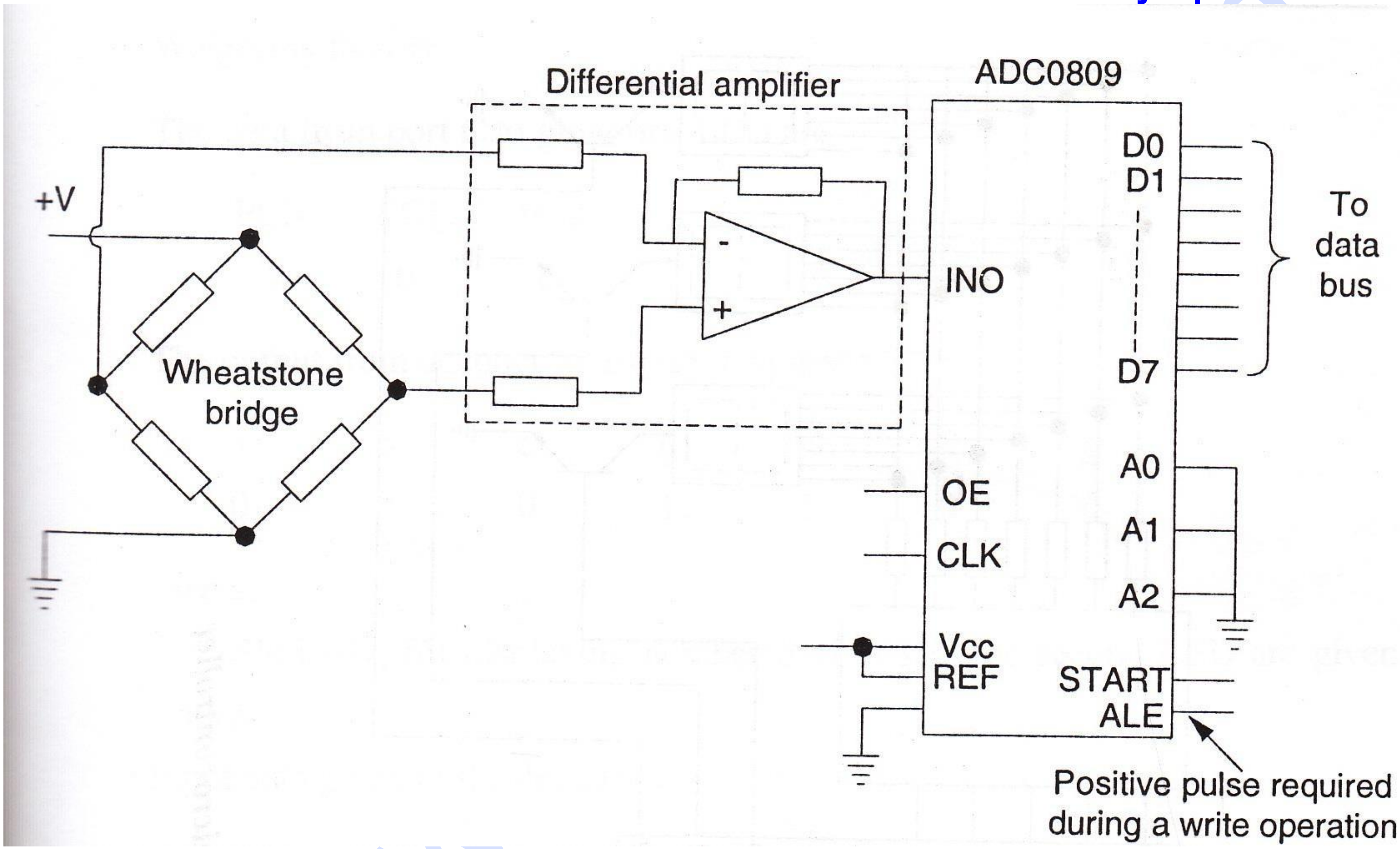
Weighing scales



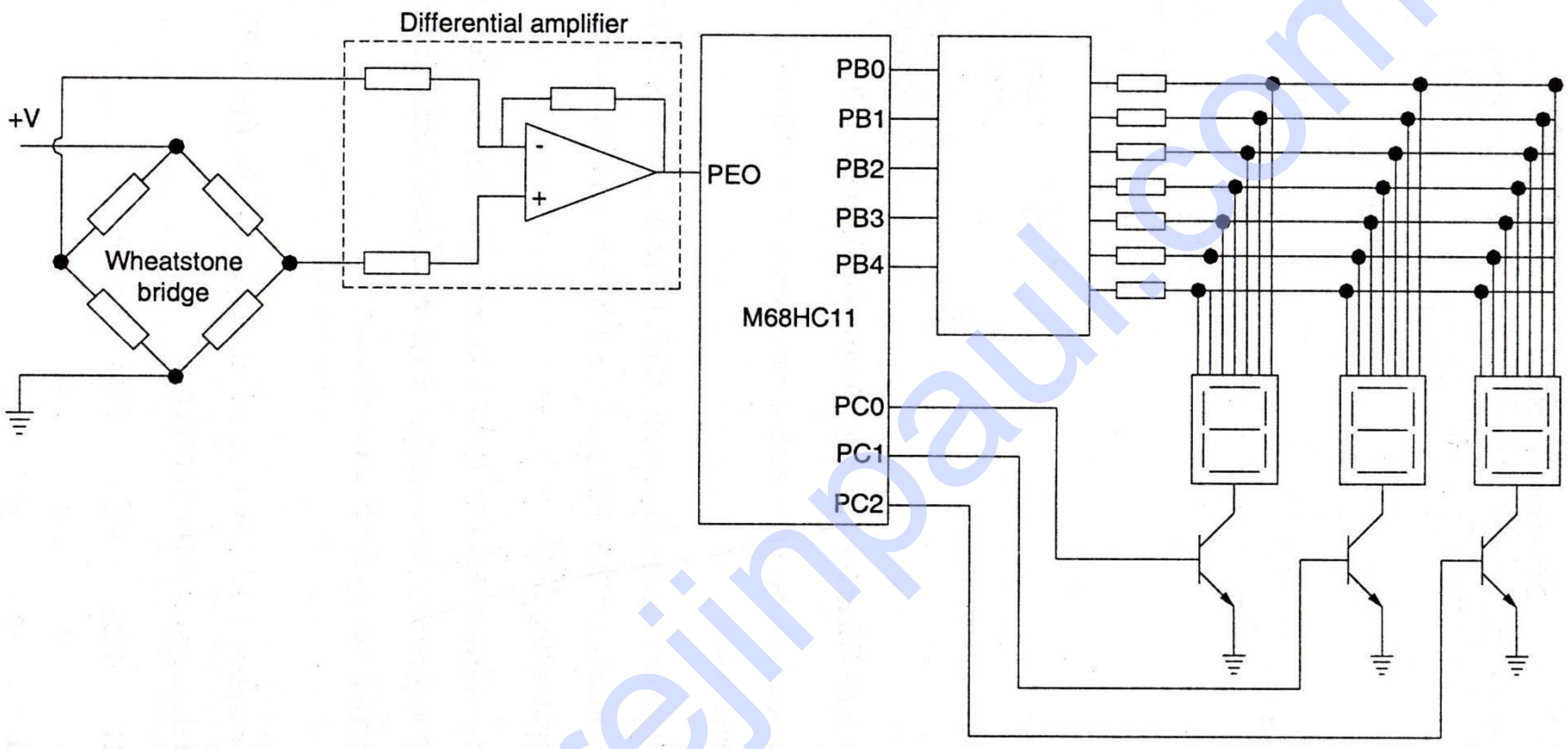
Weighing Scales



Weighing Scales mechanism



Wheatstone bridge arrangement with ADC

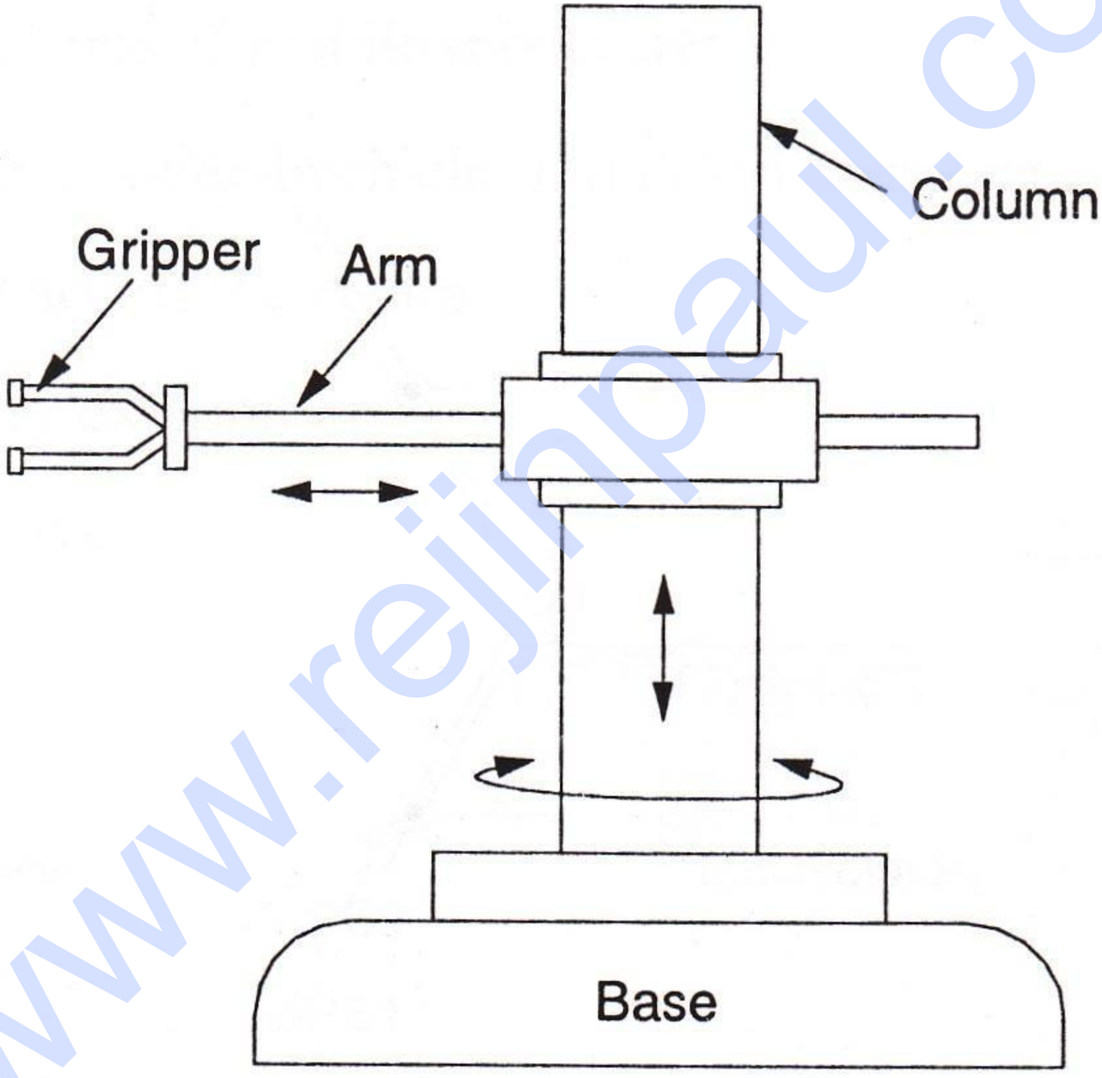


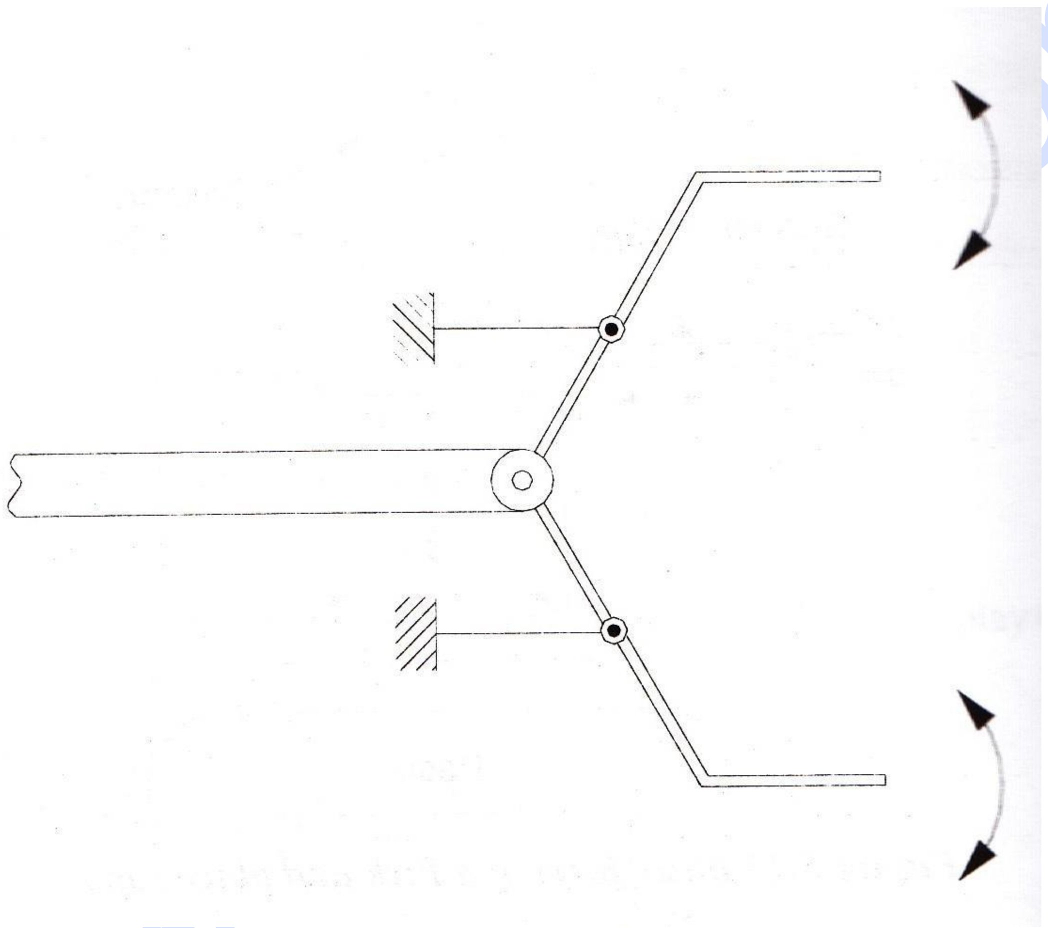
Wheatstone bridge with microcontroller

Case studies

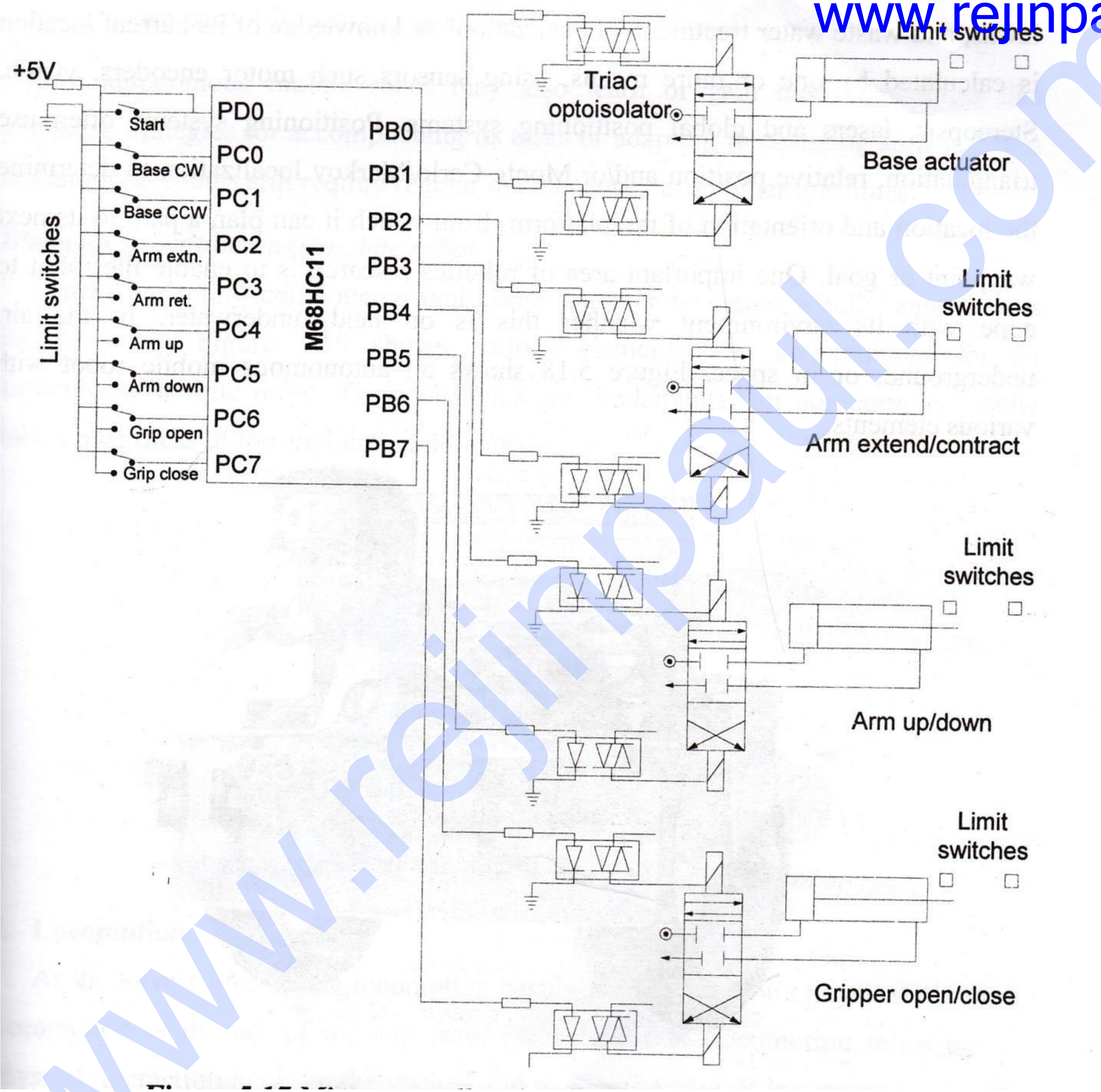
- Pick and place robot
- Autonomous mobile robot
- Wireless surveillance balloon
- Engine management system
- Automatic car park barrier

Pick and place robot



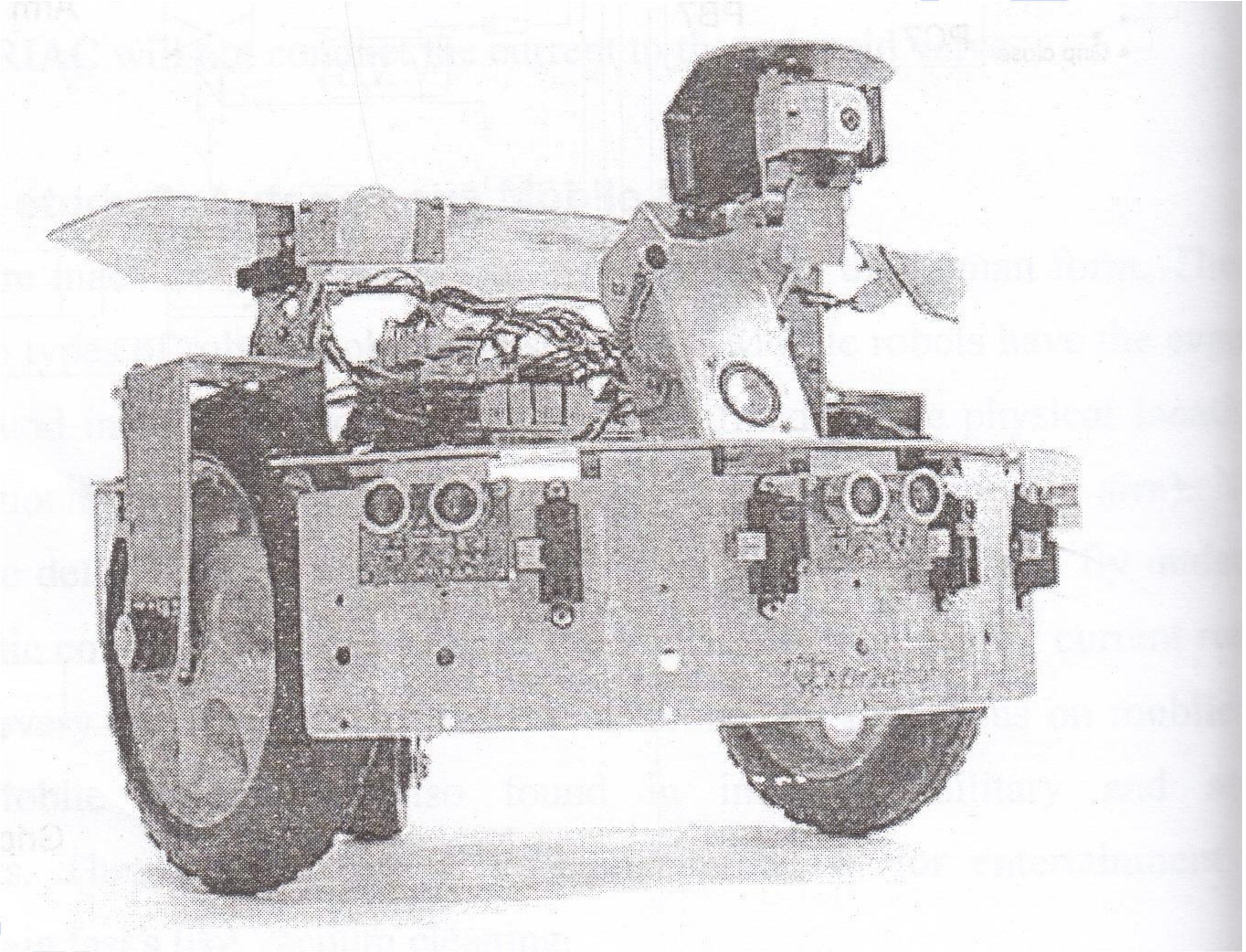


Gripper mechanism of a robot

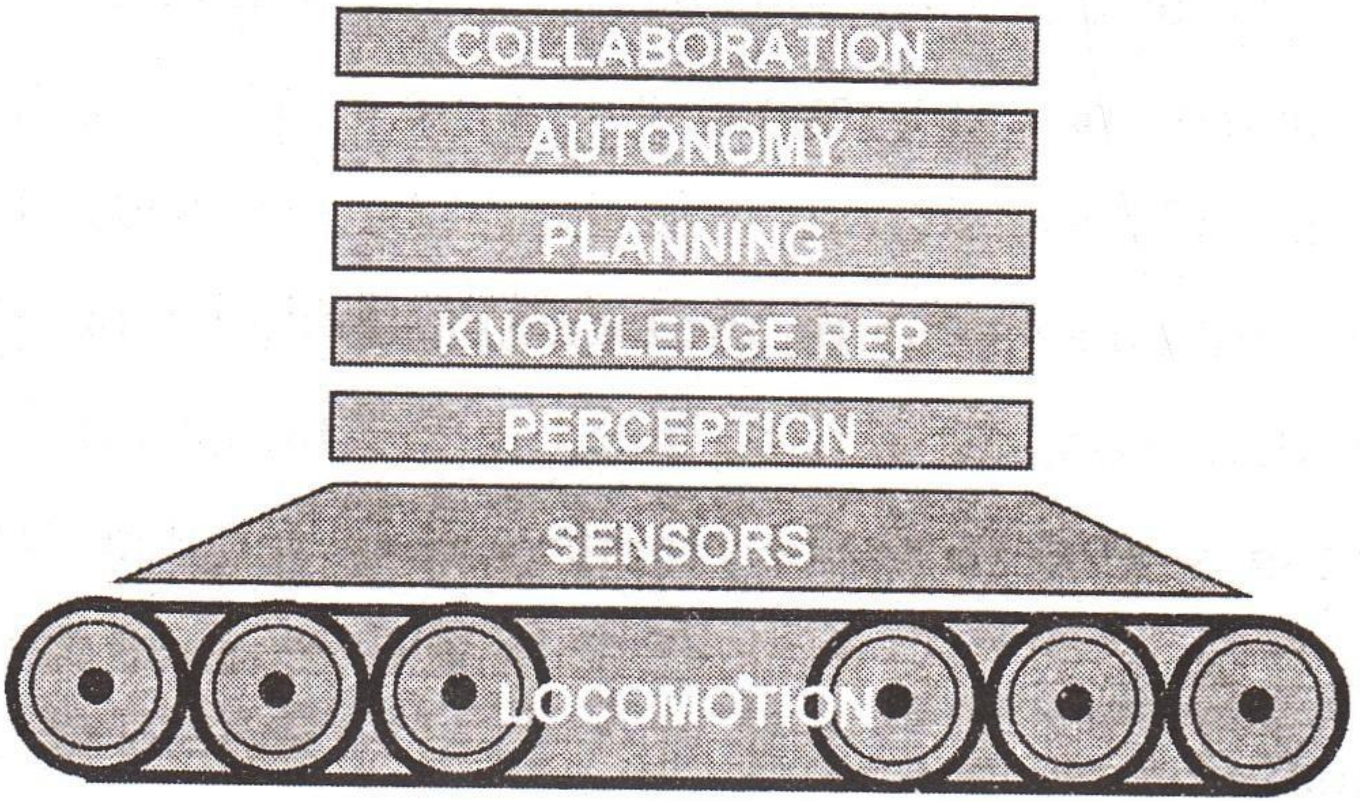


Microcontroller circuit for pick and place robot

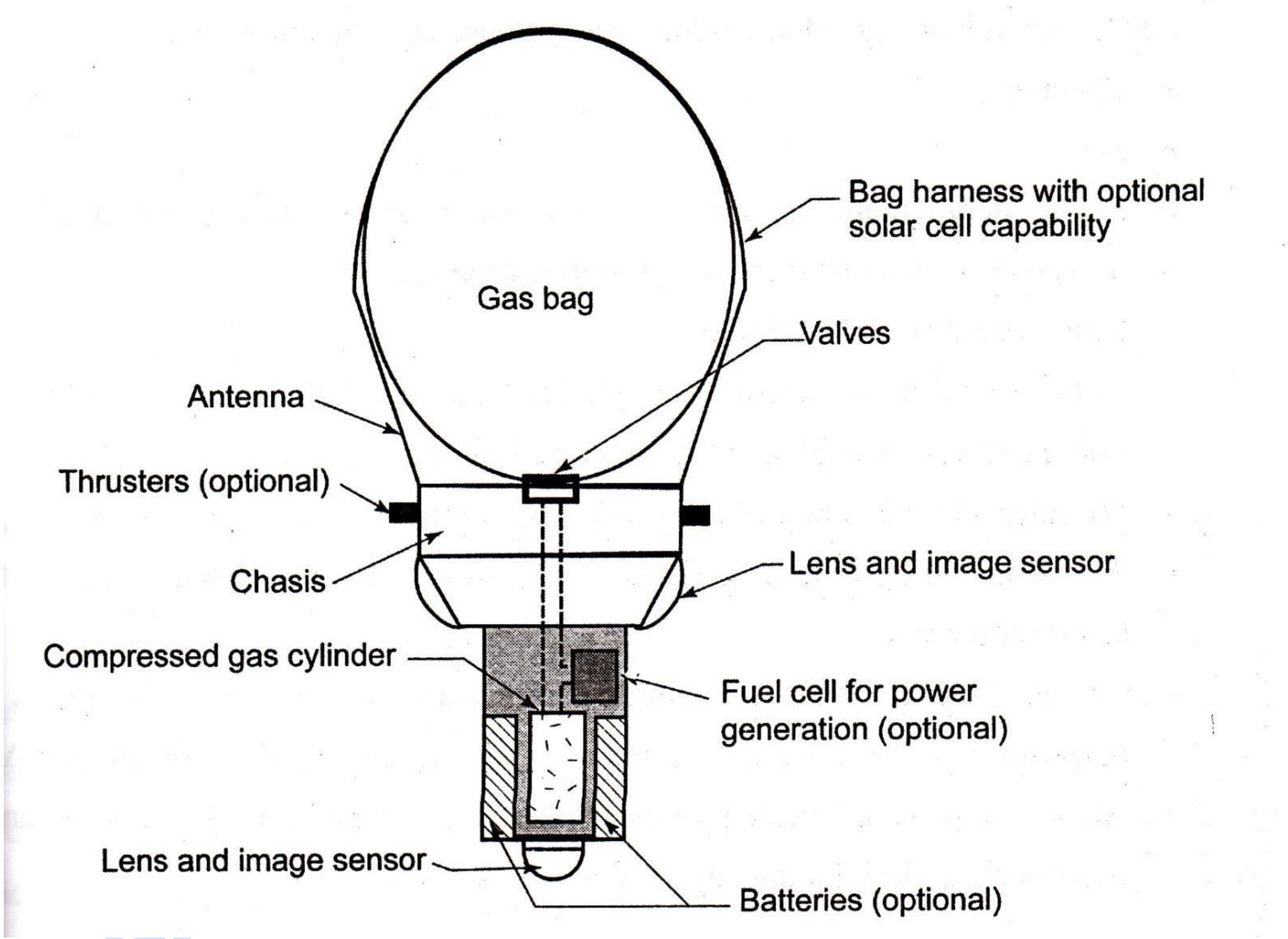
Autonomous mobile robot



Elements of autonomous mobile robot



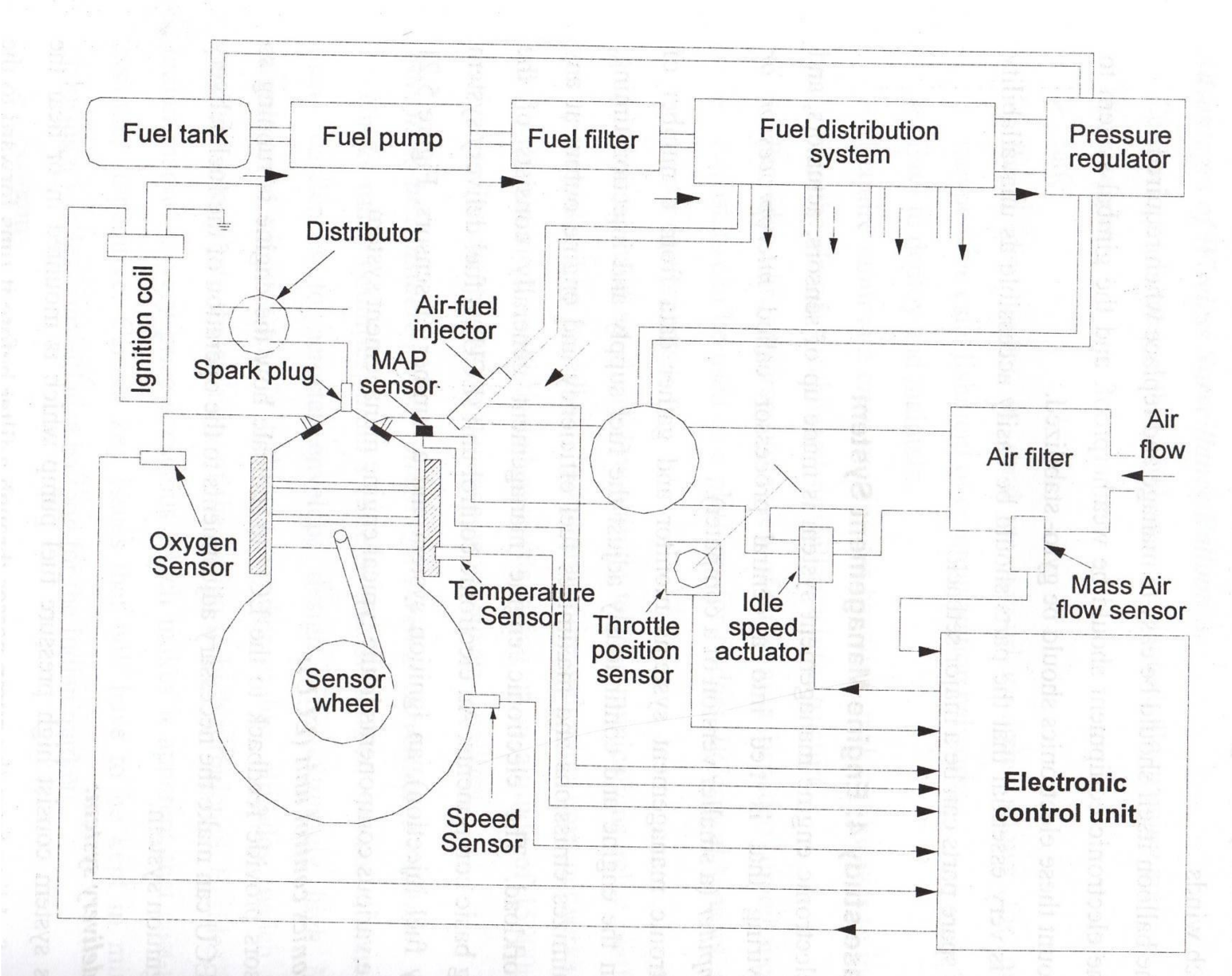
Wireless surveillance balloon



Applications of wireless surveillance Ballon

- Border security (TARS) in military
- Enhancing battle field situational awareness
- Coastal surveillance
- Platform for mounting telecommunication, television, radio transmitters etc.,
- Aerial platform for scientific instrument testing

Engine Management System



Basic components

- Electronic control unit
- Fuel delivery system
- Ignition system
- Various sensors
  - Throttle position sensors
  - Exhaust gas oxygen sensors
  - Manifold absolute pressure sensors
  - Temperature sensors
  - Engine speed/Timing sensors
  - Exhaust gas regulation sensors
  - Mass sir flow sensors

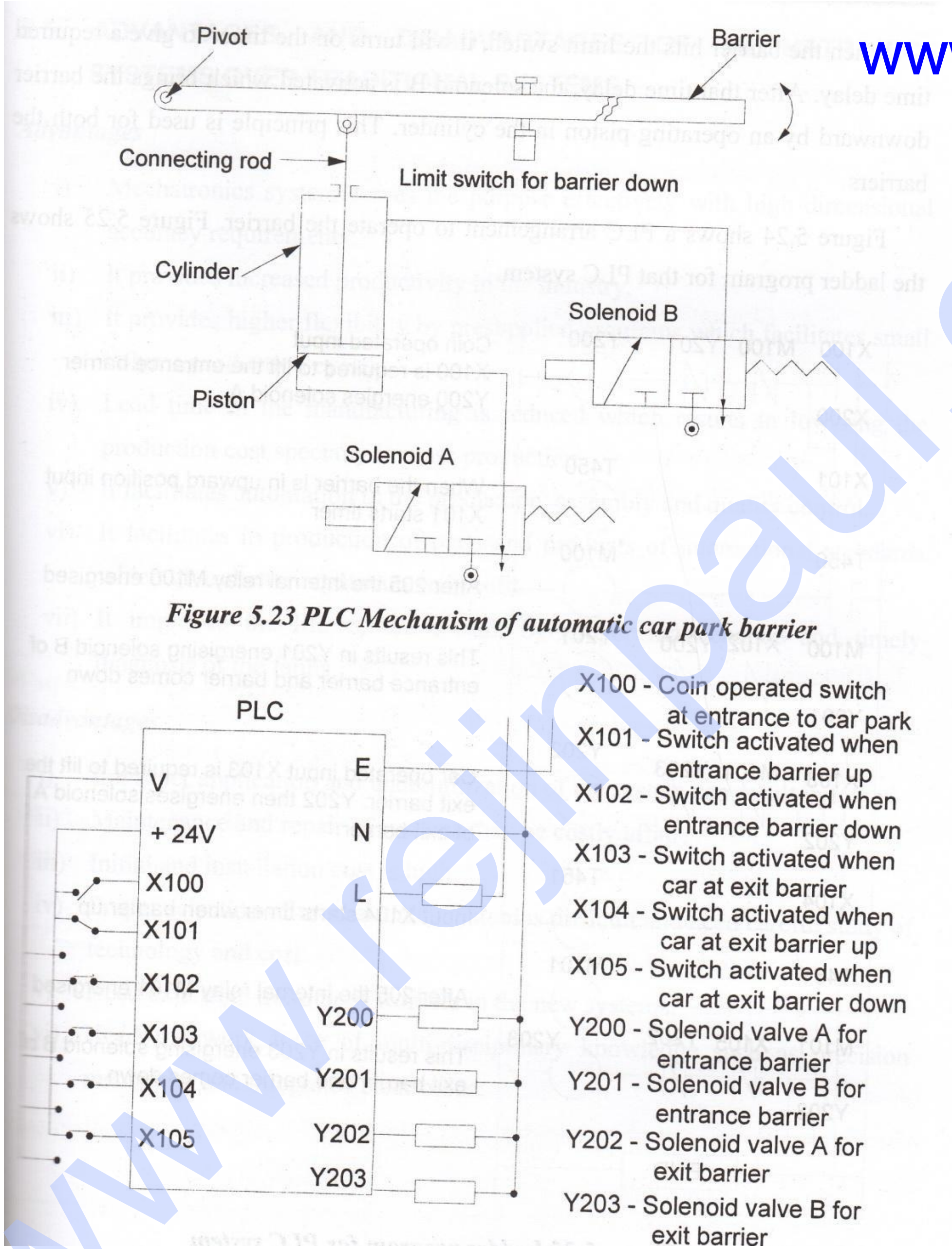


Figure 5.23 PLC Mechanism of automatic car park barrier

Interfacing of sensors with controller in Automatic car park barrier