Nadar Saraswathi College of Engineering & Technology

NSCET E-LEARNING PRESENTATION

LISTEN … LEARN… LEAD…

DEPARTMENT OF ELECTRICAL & ELECTRONICS ENGINEERING

III YEAR / V SEMESTER

EE8551 – MICROPROCESSOR AND MICROCONTROLLER

R.CHITRA M.E.,
Assistant Professor,
Nadar Saraswathi College of Engineering & Technology,
Vadapudupatti, Annanji (po), Theni – 625531.

UNIT 05

MICRO CONTROLLER PROGRAMMING & APPLICATIONS

UNIT V MICRO CONTROLLER PROGRAMMING & APPLICATIONS

Simple programming exercises- key board and display interface –Control of servo motor – Stepper motor control – Apllication to automation System

Addressing Modes:

The different ways that a microprocessor can access data are referred to as addressing modes.

The 8085 has 5 addressing modes. These are:

1. **Immediate addressing mode**:

- In an immediate addressing mode, 8 or 16 bit data can be specified as a part of instruction.
- In 8085, the instructions having 'I' letter fall under this category. "I' indicates immediate addressing mode.

Example:

MVI A,20H ; moves 8-bit immediate data(20H) into accumulator.

LXI D,10FF H ; moves 16-bit immediate data into DE register pair.

2. **Register addressing mode:**

• specifies the source operand, destination operand or both to be contained in an 8085 registers.

• This results in faster execution, since it is not necessary to access memory locations for operands.

Example: MOV A, B ; moves the contents of register B into the accumulator.

SPHL ; moves the contents of HL register pair into stack pointer

3. Direct addressing mode:

- specifies the 16- bit address of the operand within the instruction itself.
- The second and third bytes of instruction contain this 16 bit address.

Example: LDA 2000H ; loads the 8bit contents of memory location 2000H into the accumulator

4. Implied addressing mode: Opcode specifies the address of the operands.

Example: CMA; Complements contents of accumulator.

RAL; Rotates the contents of accumulator left through the carry

5. Indirect addressing mode: In indirect addressing mode, the memory address where the operand located is specified by the contents of a register pair.

Example: LDAX B ; loads the accumulator with the contents of memory location pointed by BC register pair.

MOV M, A ; Stores the contents of accumulator into the memory location pointed by HL register pair

Classification of Instruction Set

- Data Transfer Instruction
- Arithmetic Instructions
- Logical Instructions
- Branching Instructions
- Control Instructions

Data Transfer Instructions

•These instructions move data between registers, or between memory and registers.

•These instructions copy data from source to destination(without changing the original data).

•The data transfer instructions move the data between registers or between registers

•and memory. It copies the data from source location to destination location.

•No flags will be affected.

Arithmetic Instructions

These instructions perform the operations like:

◦ Addition

◦ Subtract

◦ Increment

◦ Decrement

Logical instructions:

The logical instructions includes AND, OR, XOR, Complement operations.

Branching (Control Transfer)instructions :

The branching instructions are used to change the execution order.They are divided into conditional jump/call or unconditional jump/call

Stack I/O, Machine Control Instructions

These instructions are used to manipulate the stack to perform the input /output and to alter the internal control flags. Unless specified the flags are not affected.

The internal RAM location 30H holds 40H. The value stored in RAM location 40H is 10H. What will be the value in locations 30H and 40H after the following instructions are executed?

```
MOV   R0, #30H
MOV   A, @R0
MOV   R1, A
MOV   A, # 7FH
MOV   @R1, A
XCHD  A, @R0
```

- **Solution:**

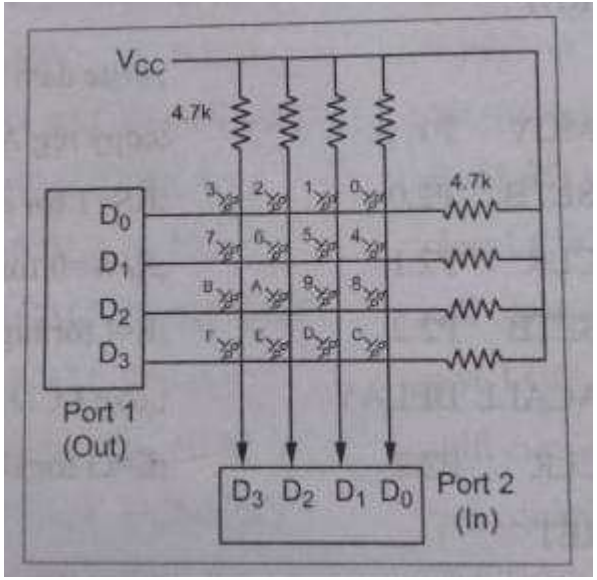
```
MOV   R0, #30H    ;   R(0) = 30H
MOV   A, @R0      ;   (A) = 40H
MOV   R1, A       ;   (R1) = 40H
MOV   A, # 7FH    ;   (A) = 7FH
MOV   @R1, A      ;   RAM (40H) = 7FH
XCHD  A, @R0      ;   RAM (30H) = 7FH, (A) = 70H
```

An array of 10 numbers is stored in the internal data RAM starting from the location 30H. Write a program to move the array starting from location 40H.

Solution:

```
COUNT:  DB      10
        MOV     R0, #30H   ; (R0) = 30H
        MOV     R1, #40H   ; (R1) = 40H
REPT    MOV     A, @ R0    ; (A) = ((R0))
        MOV     @ R1, A    ; ((R1)) = (A)
        INC     R0
        INC     R1
        ; Decrement COUNT and jump to REPT if COUNT ≠ 0
```

Keyboard Interface



Keyboard Interface

• A keyboard is a very common input device. We assume that there are only eight keys for the simple keyboard.

• Whenever a key is pressed on this keyboard by the user, the microprocessor immediately identifies which key is pressed. The action which is to be performed by the processor depends on that particular key that gets pressed.

• Example - cited as on a particular keyboard for calculator the addition operation gets performed when we press the '+' key. Moreover, a microprocessor does not get directly communicated by the input device it is done via the input port. The input port is of tri-stated gates and a tri-stated buffer is of 7244 chip.

• There are total eight buffers which we arrange into two groups of four buffers with the outputs that are non inverted and are in tri-states.

•There is a short circuit between the input and the output when the enable pin is at logic 0. As well as there is an open circuit when the enable pin is at logic 1.

• We call the outputs to be the state of high impedance significantly called as tristate.

• Hence the 74244 the content of tristate buffer gets received in the 8085 accumulator when IN 77H gets executed.

Scanning and Interfacing

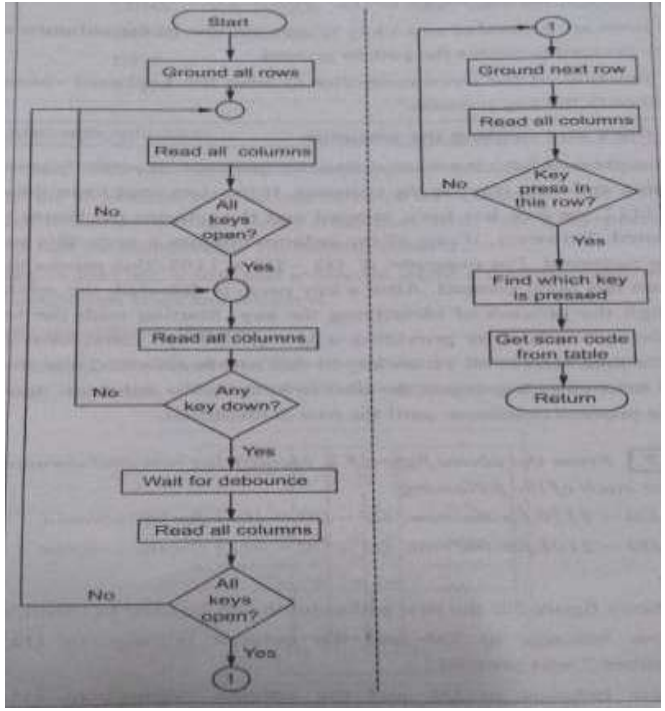
4 * 4 matrix connected to two ports

The rows are connected to an output port and the columns are connected to an input port

If no key has been pressed, reading the input port will yield 1s for all columns since they are all connected to high

It is the function of the microprocessor to scan the keyboad

```
                MOV     P2, #0FFH       ; make P2 an input port
K1:             MOV     P1, #0          ; ground all rows at once
                MCV     A, P2           ; read all col. ensure all keys open.
                ANL     A, #00001111B   ; masked unused bits
                CJNE    A, #00001111B, K1 ; check till all keys released
K2:             ACALL   DELAY           ; call 20 ms delay
                MCV     A, P2           ; see if any key is pressed
                ANL     A, #00001111B   ; mask unused bits
                CJNE    A,#00001111B,OVER ; key pressed, await closure
                SJMP    K2              ; check if key pressed
OVER:           ACALL   DELAY           ; wait 20 ms debounce time
                MCV     A, P2           ; check key closure
                ANL     A, #00001111B bits ; mask unused
                CJNE    A,#00001111B, OVER1 ; key pressed, find row
                SJMP    K2              ; if none, keep polling
OVER 1          MOV     P1, #11111110B  ; ground row 0
                MOV     A, P2           ; read all columns
                ANL     A, #00001111B   ; mask unused bits
                CJNE    A, #00001111B, ROW_0 ; key row 0, find the col.
                MOV     P1, #11111101B  ; ground row 1
                MOV     A, P2           ; read all columns
```


Start
Ground all rows
Read all columns
All keys open?
No
Yes
Read all columns
Any key down?
No
Yes
Wait for debounce
Read all columns
All keys open?
No
Yes
1
1
Ground next row
Read all columns
Key press in this row?
No
Yes
Find which key is pressed
Get scan code from table
Return

Stepper Motor

• Stepper motors are used to translate electrical pulses into mechanical movements.

• In some disk drives, dot matrix printers, and some other different places the stepper motors are used. The main advantage of using the stepper motor is the position control.

• Stepper motors generally have a permanent magnet shaft (rotor), and it is surrounded by a stator.

•The 8 Key keypad is connected with circuit through which user can give the command to control stepper motor.

• The control circuit includes micro controller 89C51, indicating LEDs, and current driver chip ULN2003A. By giving different commands the stepper motor can run clockwise, run anticlockwise, increase/decrease RPM,increase/decrease revolutions, stop motor, change the mode, etc. Stepper motor has four coils.

•One end of each coil is tied together and it gives common terminal which is always connected with positive terminal of supply

The other ends of each coil are given for interface. Specific color code may also be given.

First Coil L1-Orange

Second Coil L2 -Brown

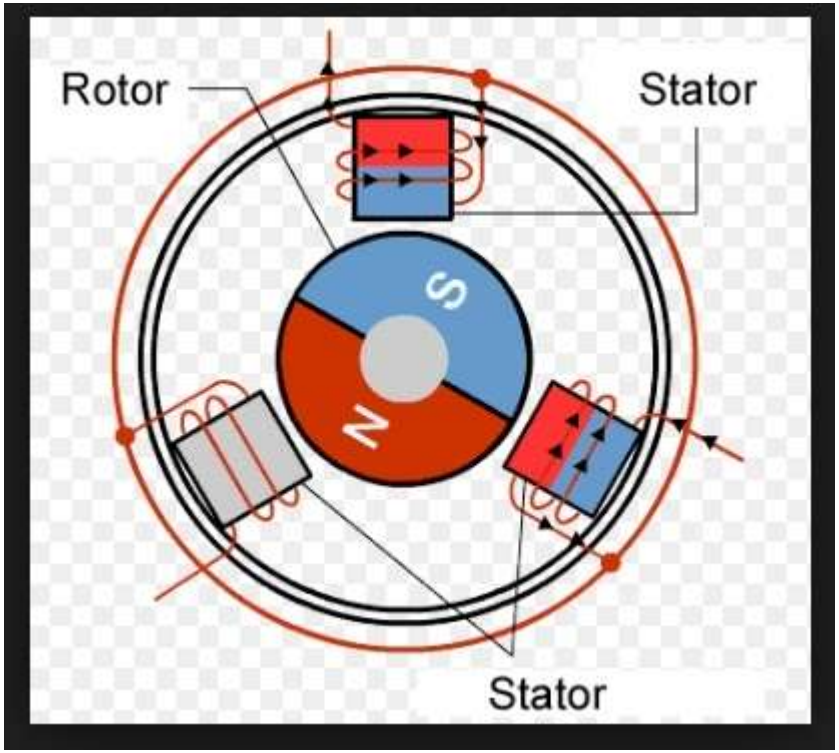
Third Coil L3 - Yellow

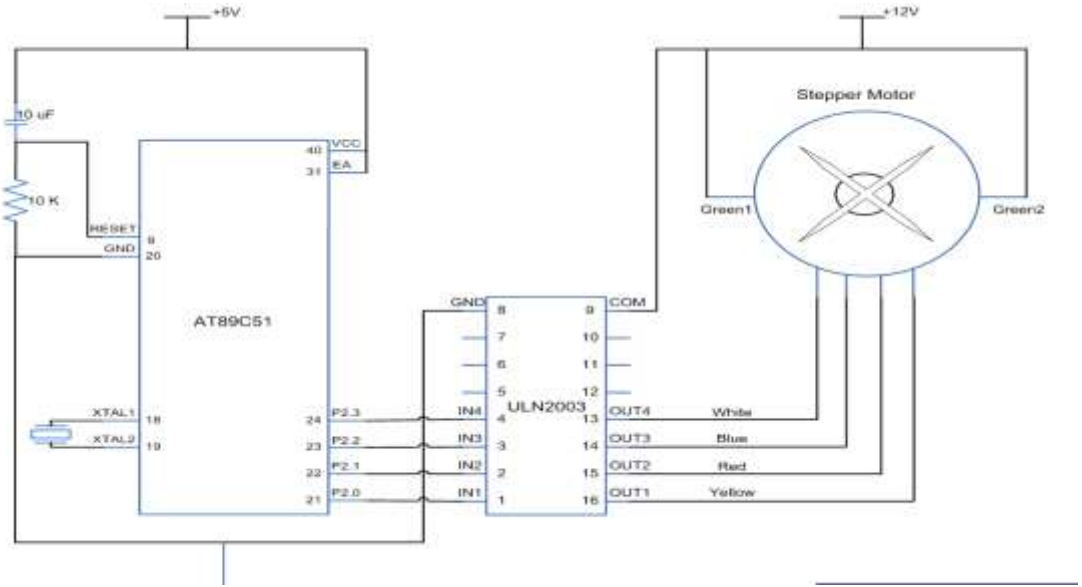
Fourth Coil L4 - Black

Common Terminal -Red

By means of controlling a stepper motor operation we can

1. Increase or decrease the RPM (speed) of it
2. Increase or decrease number of revolutions of it
3. Change its direction means rotate it clockwise or anticlockwise


Rotor
Stator
S
N
Stator


+5V
+12V
10 uF
10 K
RESET
GND
VCC
EA
AT89C51
XTAL1
XTAL2
P2.3
P2.2
P2.1
P2.0
GND
ULN2003
COM
IN4
IN3
IN2
IN1
OUT4
OUT3
OUT2
OUT1
White
Blue
Red
Yellow
Stepper Motor
Green1
Green2

Some parameters of stepper motors −

Step Angle − The step angle is the angle in which the rotor moves when one pulse is applied as an input of the stator. This parameter is used to determine the positioning of a stepper motor.

Steps per Revolution − This is the number of step angles required for a complete revolution. So the formula is 360° /Step Angle.

Steps per Second − This parameter is used to measure a number of steps covered in each second.

RPM − The RPM is the Revolution Per Minute. It measures the frequency of rotation. By this parameter, we can measure the number of rotations in one minute.

The relation between RPM, steps per revolution, and steps per second is like below:

Steps per Second = rpm x steps per revolution / 60

Interfacing Stepper Motor with 8051 Microcontroller

• Weare using Port P0 of 8051 for connecting the stepper motor. HereULN2003 is used. This is basically a high voltage, high current Darlington transistor array.

• Each ULN2003 has seven NPN Darlington pairs. It can provide high voltage output with common cathode clamp diodes for switching inductive loads.

The Unipolar stepper motor works in three modes.

Wave Drive Mode − In this mode, one coil is energized at a time. So all four coils are energized one after another. This mode produces less torque than full step drive mode.

Half Drive Mode − In this mode, one and two coils are energized alternately. At first, one coil is energized then two coils are energized. This is basically a combination of wave and full drive mode. It increases the angular rotation of the motor

Steps	Winding A	Winding B	Winding C	Winding D
1	1	0	0	0
2	0	1	0	0
3	0	0	1	0
4	0	0	0	1

Servo Motor Interfacing with 8051 Microcontroller

• A Servo motor is one of the most commonly used motor for precise angular movement.

• The advantage of using a servo motor is that the angular position of the motor can be controlled without any feedback mechanism.

• The servo motors are usually used in commercial and industrial applications They are also widely used as in drive systems such as robots, aeroplanes etc.

• The control of servo motor connected port0 of 8051 microcontroller.

• The 11.0592MHz crystal oscillator is used to provide the clock pulsed to the microcontroller and 22pf ceramic capacitors used to stabilize the operation of crystal. 10KΩ and 10uf capacitor is used to provide the power on reset to the microcontroller.

• Servo motors are self-contained mechanical devices that are used to control the machines with machines. Usually the servo motor is used to control the angular motion among from 0° to 180° and 0° to 90°. The servo motor working principle based on the PWM (pulse width modulation) pulses.

• A Servo motor is one of the most commonly used motor for precise angular movement.

• The advantage of using a servo motor is that the angular position of the motor can be controlled without any feedback mechanism.

• Pulse Width Modulated (PWM) waves are used as control signals and the angular position is definite by the width of the pulse at the control input.

• Servo motor having angle of rotation from 0-180° and angular position can be controlled by varying the duty cycles among 1ms to 2ms.

• The control of servo motor connected port0 of 8051 microcontroller.

• The 11.0592MHz crystal oscillator is used to provide the clock pulsed to the microcontroller and 22pf ceramic capacitors used to stabilize the operation of crystal.

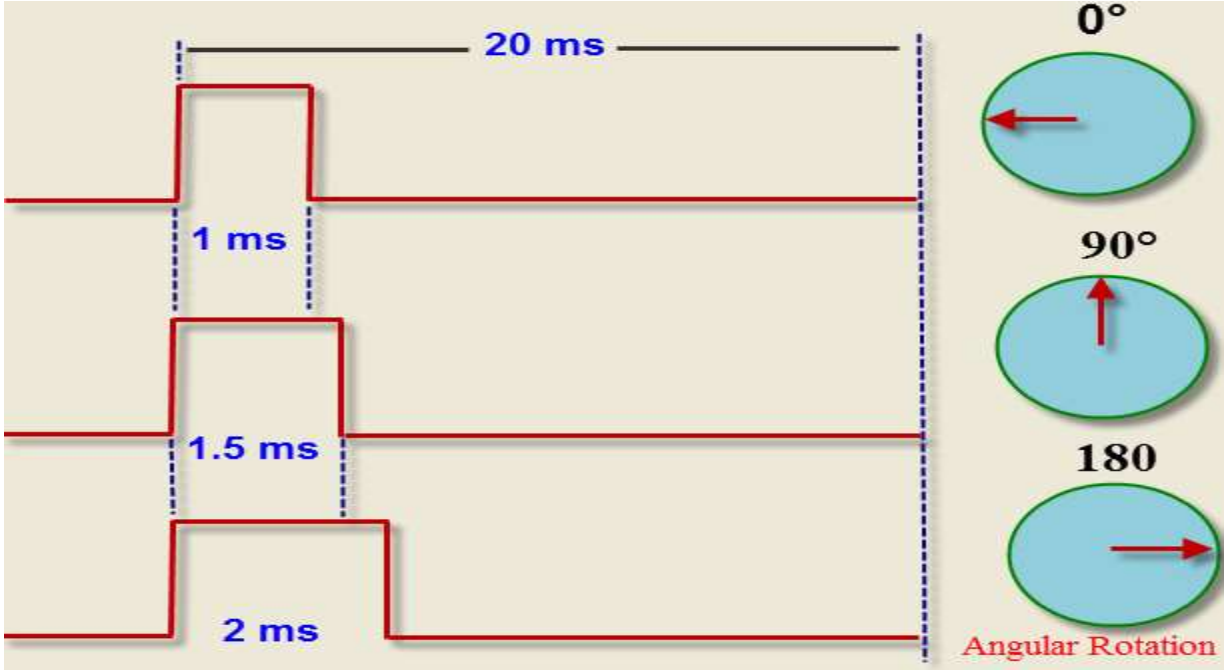
• 10KΩ and 10uf capacitor is used to provide the power on reset to the microcontroller.

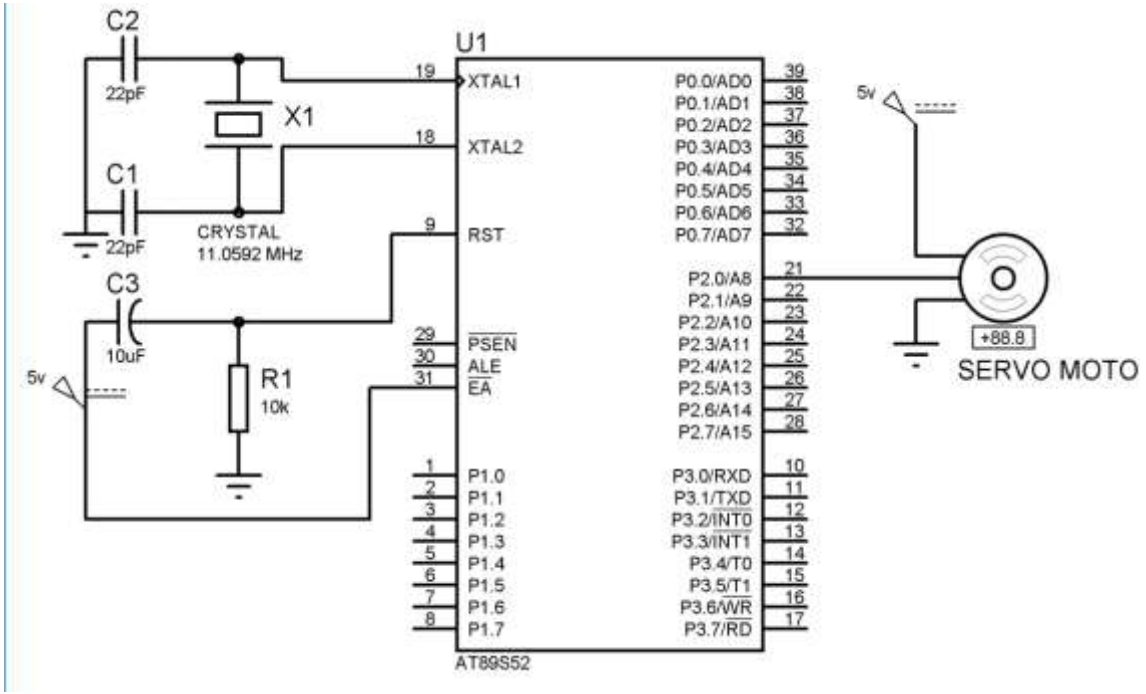
Controlling a Servo Motor with Angle rotations

Servo motor working principle mainly depends upon duty cycles. It uses Pulse Width Modulated (PWM) waves as control signals. The angle of rotation is resolute by the pulse width of the control pin. Here the servo motor used for angle of rotation from 0 to 180 degrees. We can control the precise angular position by varying the pulse among 1ms to 2ms.

Servo Motor Working Principle

The servo motor working principle mainly depends upon the „Fleming left hand rule“. Basically servo motors are adapted with DC motors, a position sensor, a Gear reduction, and an electronic circuit. The DC motors achieve powered from a battery


20 ms
1 ms
1.5 ms
2 ms
0°
90°
180
Angular Rotation


C2
22pF
X1
C1
22pF
CRYSTAL
11.0592 MHz
C3
10uF
5v
R1
10k
U1
19
XTAL1
18
XTAL2
9
RST
29
PSEN
30
ALE
31
EA
1
P1.0
2
P1.1
3
P1.2
4
P1.3
5
P1.4
6
P1.5
7
P1.6
8
P1.7
AT89S52
39
P0.0/AD0
38
P0.1/AD1
37
P0.2/AD2
36
P0.3/AD3
35
P0.4/AD4
34
P0.5/AD5
33
P0.6/AD6
32
P0.7/AD7
21
P2.0/A8
22
P2.1/A9
23
P2.2/A10
24
P2.3/A11
25
P2.4/A12
26
P2.5/A13
27
P2.6/A14
28
P2.7/A15
10
P3.0/RXD
11
P3.1/TXD
12
P3.2/INT0
13
P3.3/INT1
14
P3.4/T0
15
P3.5/T1
16
P3.6/WR
17
P3.7/RD
5v
+88.8
SERVO MOTO

• The position sensor senses the location of the shaft from its fixed position and sends the information to the control circuit.

• The control circuit decodes the signals accordingly from the position sensor and compares the actual location of the motors with the preferred position and accordingly controls the direction of rotation of the DC motor to get the necessary position.

• Generally the servo motor requires 4.8V to 6 V DC supply.

Application to automation System

• A washing machine is an electronic device that is designed to wash laundry like clothes, sheets, towels and other bedding.

• A washing machine is built with two steel tubs which are the inner tub and the outer tub whose main role is to prevent water from spilling to

other parts of the machine.

Control knobs in washing machine:

- Load select knob
- Water inlet select knob
- Mode select knob
- Program select knob

Load select knob:- load Number of clothes low medium high Load select

Water inlet select knob:- hot cold both-mixed Water inlet

Mode select knob:- Save mode Normal mode Mode

Program select knob:- Heavy Clothes very dirty Normal Normal dirty clothes LIGHT

For light dirty clothes Delicate For silk clothes

Machine Operations:-

- Fill:- water will be filled by the pump as per the load knob selected.
- Agitate:- The wash basket will rotate in a clockwise direction for 10 revolutions,

After that basket will stop for 2 seconds, then rotate 10 revolutions in anticlockwise direction. The process will be continued for specified minutes in cycle table.Drain:- After agitation, the water and detergent are drained.

Spin:- During spin, agitator will be stationary, only the basket will rotate at high speed. Then the moisture of clothes are removed through holes in the inner metallic basket.

Machine Indicators:-

Machine ON

LED ON After completion of washing cycle, buzzer sound will be generated.

Washing cycle :

Heavy

Normal

Light

Delicate

Washing Machine Drives/Connections:

The drives of the washing machine is connected to 8051 Microcontroller ports. Hot/Cold Agitator motor drive

Agitator motor drive

Spin motor drive

High level

Medium level

Low level

Drain

Washing machine ON LED

Heavy

Normal

Light

Delicate

Hot

Normal Buzzer sound Basket

Washing Machine Operation SinalsInput/Output :

The various Operation Signals are connected to microcontroller Input/output Port.

Load / water level select

Water inlet

Program select

Machine ON

Fill water

Agitation control

Output

Spin

Washing complete