VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

School of Electronics Engineering (SENSE)

PROJECT REPORT

COURSE CODE / TITLE	BECE204L– Microprocessors and Microcontrollers	
PROGRAM / YEAR/ SEM	B.Tech (BEC)/II Year/ WINTER 2023-2024	
DATE OF SUBMISSION	30-04-2024	
TEAM MEMBERS DETAILS	**REGISTER NO.**	**NAME**
	22BEC****	Syed Nabiel Hasaan M
	21BCE****	Shaun George Rajesh
	22BEC****	P Aashi Apuurvaa
	22BEC****	Guruvishwanath S
	22BAI****	Satish Kumar. V
	22BEC****	Pushkar Nidagundi
	22BEC****	Harish Krishnan P
PROJECT TITLE	**SMART IRRIGATION SYSTEM USING 8051**	

COURSE HANDLING FACULTY	**Dr. S. REVATHI** **Associate Professor, SENSE**	**REMARKS**	
FACULTY SIGNATURE			

OBJECTIVE

The objective of the "**Smart Irrigation System using 8051**" project is to develop an automated irrigation solution for agriculture. The system integrates an 8051 microcontroller, a moisture sensor, an analog-to-digital converter (ADC), a submersible pump, and an LCD display. The primary goal is to accurately measure soil moisture levels using the sensor, convert the analog data into digital signals through the ADC, and make informed irrigation decisions based on this data. By implementing intelligent control algorithms, the system activates the submersible pump only when necessary to water the crops, thereby optimizing water usage. Real-time feedback is provided to the user through the LCD display, showing moisture levels, system status, and any relevant alerts. Overall, the project aims to enhance agricultural efficiency, conserve water resources, and simplify irrigation management for farmers.

INTRODUCTION

The "**Smart Irrigation System using 8051**" project represents a significant advancement in agricultural technology, aiming to address the challenges of water management in farming. Traditional irrigation methods often result in water wastage and inefficient use of resources, leading to increased costs and reduced crop yields. To tackle these issues, our project introduces an automated irrigation system that utilizes modern technology to optimize water usage and improve crop health.

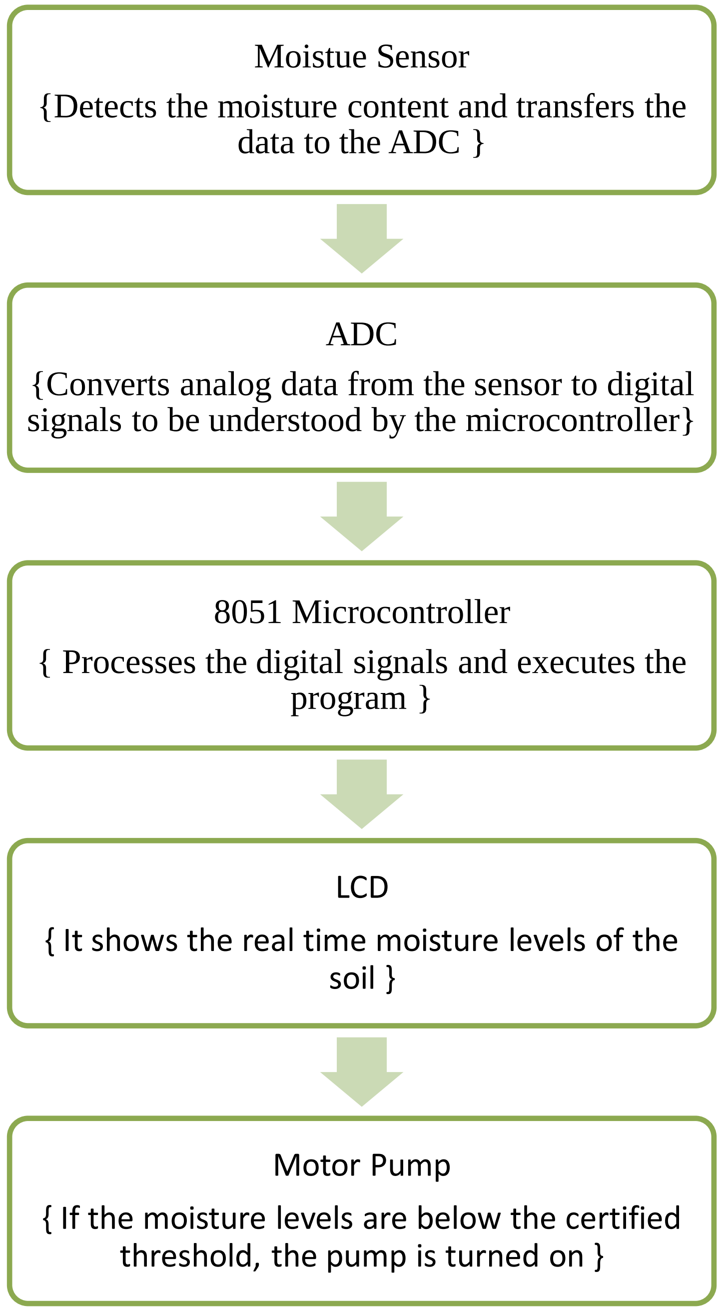
Existing irrigation systems typically rely on manual monitoring and control, which can be time-consuming and prone to human error. Our motivation for this project stems from the desire to create a more efficient and sustainable solution for farmers. By integrating an 8051 microcontroller, a moisture sensor, an ADC, a submersible pump, and an LCD display, we're developing a system that can accurately measure soil moisture levels, make intelligent irrigation decisions, and provide real-time feedback to users.

Unlike traditional irrigation methods, which may apply water indiscriminately, our system ensures that water is only applied when and where it is needed. This not only conserves water but also promotes healthier plant growth and higher crop yields.

Moreover, the user-friendly interface of our system makes it accessible to farmers of all levels of expertise. With the LCD display providing easy-to-understand information about moisture levels and system status, farmers can effectively manage their irrigation processes without the need for extensive training or technical knowledge.

Our project aims to revolutionize agriculture by introducing a more efficient, precise, and user-friendly irrigation solution. By harnessing the power of modern technology, we're enabling farmers to optimize their water usage, increase crop yields, and contribute to a more sustainable future for agriculture.

BLOCK DIAGRAM



1. Moisture Sensor

The moisture sensor detects the moisture content in the soil by measuring its electrical conductivity or resistance. It provides analog output signals proportional to the soil moisture levels. These analog signals are then fed into the Analog-to-Digital Converter (ADC) for further processing.

2. ADC (Analog-to-Digital Converter)

The ADC converts the analog signals from the moisture sensor into digital data that can be understood by the microcontroller. It samples the analog signal at regular intervals and quantizes it into discrete digital values. The digital data is then passed to the 8051 microcontroller for analysis and decision-making.

3. 8051 Microcontroller

The 8051 microcontroller receives the digital signals from the ADC and processes them using a programmed algorithm. It executes the intelligent irrigation control program, which includes logic to determine when and how much to irrigate based on the soil moisture readings. The microcontroller interfaces with the LCD display to provide real-time feedback on the soil moisture levels and system status.

4. LCD Display

The LCD display provides a user-friendly interface for farmers to monitor the soil moisture levels and system operation. It shows real-time data such as the current moisture level in the soil and any alerts or messages indicating the system status. This display enables farmers to make informed decisions about irrigation management without the need for specialized technical knowledge.

5. Motor Pump

The motor pump is controlled by the microcontroller based on the moisture readings from the sensor. If the moisture levels detected by the sensor fall below a predefined threshold indicating dry soil, the microcontroller activates the pump to irrigate the soil. This ensures that water is only applied when necessary, optimizing water usage and promoting healthy plant growth.

COMPONENTS/SOFTWARE REQUIRED

HARDWARE :

1. AT89S52 8051 Microcontroller
2. 16x2 LCD
3. L293D Motor Driver
4. ADC0804
5. 8051 Developer Board
6. Soil Moisture Sensor
7. 6V Submersible Pump
8. 5V Regulated Power Supply
9. Breadboard , Resistors , Capacitors and wires

SOFTWARE :

1. Keil Uvision
2. Proteus 8 Professional

PROJECT DESCRIPTION

The "Smart Irrigation System using 8051" project is designed to automate the irrigation process for agricultural applications, operating on a closed-loop principle. The system continuously monitors the soil moisture level, and when it falls below the specified threshold level, the pumps are instructed to turn on and water the plants. This ensures that the plants receive the required amount of water precisely when needed, optimizing water usage and promoting healthy crop growth.

1. The moisture sensor is an integral component of the system, continuously monitoring the soil's moisture level by establishing consistent contact with it. It accurately measures the soil's moisture content at regular intervals. The sensor then translates this data into analog output signals, providing insights into the soil's hydration levels. This information serves as the foundation for the system's operation, allowing it to make informed decisions regarding irrigation.

2. The analog output signals generated by the moisture sensor undergo processing through the ADC0804 analog-to-digital converter. This component of the system plays an role in transforming the analog signals into digital format, making them compatible with the 8051 microcontroller. Through precise conversion, the ADC0804 ensures that the moisture data is accurately represented in a digital form, facilitating efficient processing by the microcontroller.

3. Upon receiving the digitized moisture data from the ADC0804, the 8051 microcontroller takes charge of the system's decision-making process. Equipped with a programmed algorithm for irrigation control, the microcontroller meticulously analyzes the moisture readings.

4. Real-time feedback on the soil's moisture levels is essential for effective irrigation management. This feedback is provided through the 16x2 LCD display connected to the microcontroller. With the LCD display, users can easily monitor the moisture levels and system status, enabling them to make timely adjustments and interventions as needed.

5. The microcontroller takes proactive measures to ensure that the soil remains adequately hydrated. When the moisture level falls below a predefined threshold, indicating dry soil conditions, the microcontroller activates the L293D motor driver. This component serves as the interface between the microcontroller and the pump, providing the necessary control signal to activate the irrigation system.

6. With the pump now in operation, a predetermined amount of water is delivered to the plants, replenishing the soil's moisture content. The pump's activation and water delivery process are carefully

orchestrated by the microcontroller, ensuring that the plants receive the optimal amount of water required for healthy growth. Once the soil's moisture level reaches the desired threshold, the system returns to its monitoring state, ready to initiate irrigation again as needed.

SIMULATION

1. Open Proteus software.

2. Create a new project and add the required components to the workspace:

   - AT89S52 Microcontroller
   - 16x2 LCD Display
   - ADC0804
   - Soil Moisture Sensor
   - L293D Motor Driver
   - Submersible Pump
   - Power Supply (5V and 6V)
   - Wires and Ground Connections

3. Connect the components in the schematic:

   - Connect the output of the Soil Moisture Sensor to the ADC0804.
   - Connect the output of the ADC0804 to the Microcontroller.
   - Connect the Microcontroller to the LCD Display for output.
   - Connect the Microcontroller to the L293D Motor Driver for controlling the pump.
   - Connect the pump to the Motor Driver.

4. Configure the components:

   - Set the properties of the Microcontroller, ADC0804, LCD, and Motor Driver according to your circuit requirements.

5. Write code for the Microcontroller in Keil uVision and compile it to generate the hex file.

6. Load the hex file into the Microcontroller component in Proteus.

7. Run the simulation and observe the results:

- Monitor the LCD display for real-time feedback on moisture levels and system status.

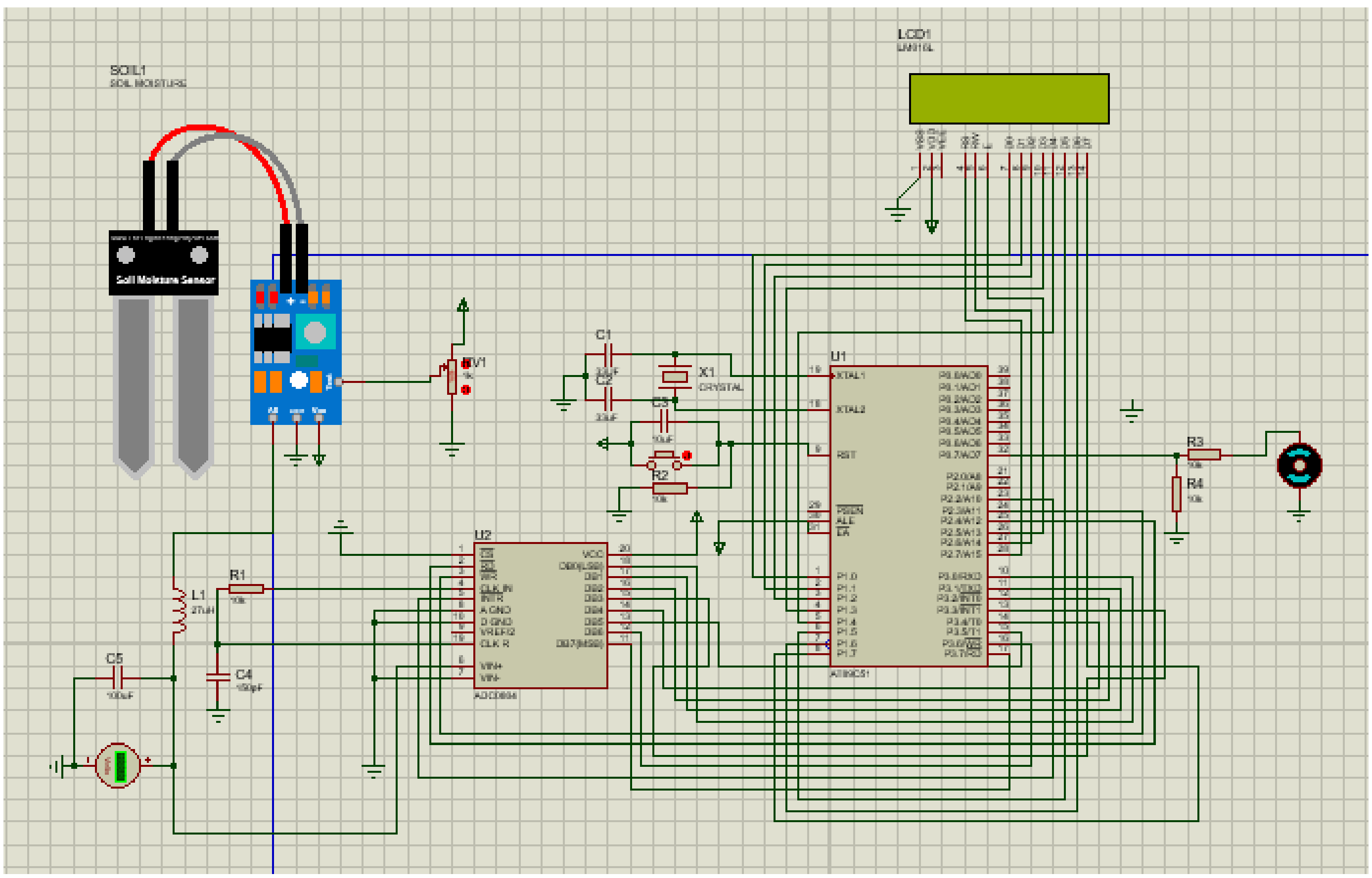
- Interact with the system by adjusting the soil moisture level and observing how the system responds.

8. Troubleshoot any issues that arise during simulation:

- Check the connections, component properties, and code for errors.

- Make necessary adjustments and iterate through the simulation process until the desired functionality is achieved.

9. Save your Proteus project and document your simulation setup for future reference.



ONCEPTS LEARNED

1. Closed-loop Control:

Understanding the concept of closed-loop control systems where the system continuously monitors and adjusts its operation based on feedback from sensors, such as soil moisture levels.

2. Microcontroller Programming:

Learning to program and interface an 8051 microcontroller to perform tasks such as data processing, decision-making, and controlling external devices.

3. Analog-to-Digital Conversion:

Understanding the process of converting analog signals from sensors into digital data for processing by microcontrollers, using components like the ADC0804.

4. Sensor Interfacing:

Understanding how to interface sensors such as the soil moisture sensor with microcontrollers to measure physical quantities and provide input data.

5. Motor Driver Control:

Learning to control external devices such as pumps using motor drivers like the L293D, enabling precise control over irrigation systems.

6. Real-time Feedback:

Implementing real-time feedback mechanisms using components like the 16x2 LCD display to provide users with information about the system's operation and status.

7. System Integration:

Integrating hardware components and software programs to create a cohesive system that performs complex tasks such as automated irrigation.

8. Simulation Techniques:

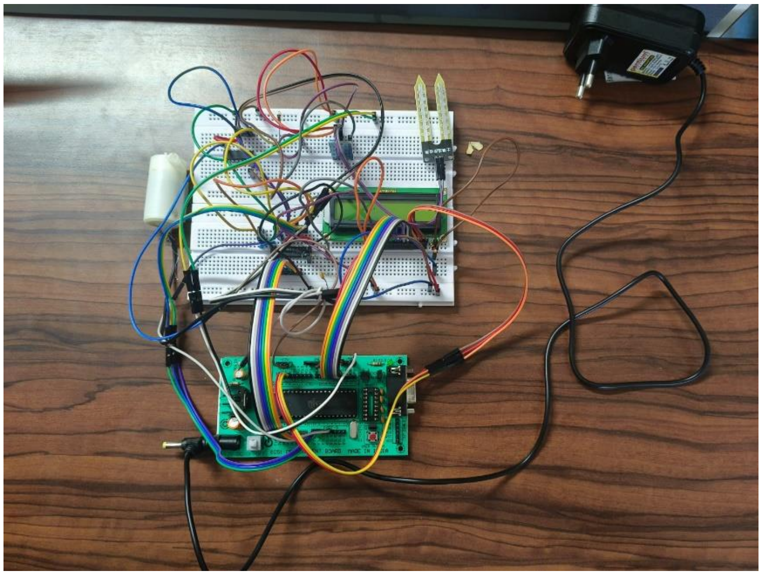
Learning to simulate electronic circuits and microcontroller programs using software tools like Proteus, enabling testing and validation of designs before implementation.

9. Efficient Resource Management:

Understanding the importance of efficient resource management, such as optimizing water usage in agriculture, and implementing strategies to achieve it.

IMPLEMENTATION

1. Prototype Setup:



This photograph shows the complete setup of the Smart Irrigation System prototype. The components include the 8051 microcontroller, moisture sensor, ADC, LCD display, motor driver, and submersible pump. The system is ready for testing and demonstration.

2. Normal Operation:

In this scenario, the system is operating under normal conditions. The moisture sensor continuously monitors the soil moisture levels, and the LCD display shows real-time feedback to the user. The pump remains inactive as the soil moisture is above the predefined threshold.

3. Low Moisture Level Detected:

Here, the moisture sensor has detected a low moisture level in the soil. The microcontroller analyzes the data and activates the pump through the motor driver to irrigate the plants. The LCD display indicates the irrigation process in progress.

4. Pump Activation:

This photograph captures the moment when the pump is activated by the microcontroller in response to low soil moisture levels. Water is being delivered to the plants through the submersible pump to replenish the soil's moisture content.

5. System Shutdown:

After the soil moisture levels are restored to the desired threshold, the pump is deactivated, and the system returns to standby mode. The LCD display indicates that the irrigation process is complete, and the system is ready for further monitoring.

6. Manual Override:

In this scenario, the user manually adjusts the irrigation settings using the control interface. The LCD display provides options for adjusting watering frequency or duration. The system responds accordingly to the user's input.

CHALLENGES FACED

1. Component Availability: Procuring all the required components was challenging due to limited availability in local markets and variations in stock levels.

2. Hardware Integration: Integrating various hardware components, such as the microcontroller, ADC, motor driver, and pump, required careful wiring and configuration to ensure compatibility and functionality.

3. Sensor Calibration: Calibrating the soil moisture sensor to accurately measure soil moisture levels posed a challenge, as it required fine-tuning to account for environmental factors and soil variations.

4. Programming Complexity: Writing and debugging the code for the microcontroller to control the irrigation system and interface with the sensor and display involved dealing with complex algorithms and debugging issues.

5. Simulation Setup: Setting up the simulation in Proteus to accurately model the behavior of the hardware components and test the functionality of the system required thorough understanding of the software and circuit design.

6. Power Supply Issues: Ensuring stable and sufficient power supply to all components, especially the pump, was a challenge to prevent voltage fluctuations and ensure reliable operation of the system.

7. Testing and Validation: Testing the system under various conditions to ensure its reliability and effectiveness in real-world scenarios required extensive trial and error, as well as iterative adjustments to the hardware and software.

8. Documentation: Documenting the project, including circuit diagrams, code, and simulation setup, posed a challenge in organizing and presenting the information in a clear and concise manner for future reference and replication.

APPLICATIONS

1. Agriculture: The Smart Irrigation System can be used in various agricultural applications to efficiently manage water usage and optimize crop yield. It ensures that plants receive the right amount of water at the right time, leading to healthier plants and increased productivity.

2. Greenhouses: In greenhouse farming, the system can help maintain optimal moisture levels for different types of plants, creating an ideal environment for growth. It allows growers to automate the irrigation process and monitor soil moisture levels remotely.

3. Landscaping: Landscapers and gardeners can utilize the system to maintain lawns, gardens, and other outdoor spaces by automatically watering plants based on their moisture needs. This ensures lush and healthy landscapes while conserving water resources.

4. Urban Gardening: In urban environments, where space is limited, the system can be used to cultivate rooftop gardens, vertical gardens, and community gardens. It enables urban dwellers to grow their own food efficiently and sustainably.

5. Research and Education: The project can serve as a valuable tool for research and educational purposes in agriculture, engineering, and environmental science. Students and researchers can study irrigation techniques, sensor technology, and microcontroller programming in practical applications.

6. Environmental Monitoring: Beyond agriculture, the system can be adapted for environmental monitoring in natural habitats, parks, and conservation areas. It can help monitor soil moisture levels in forests, wetlands, and other ecosystems to support conservation efforts and biodiversity.

7. Water Conservation: By optimizing water usage and reducing wastage, the system contributes to water conservation efforts, particularly in regions facing water scarcity or drought conditions. It promotes sustainable water management practices in both rural and urban areas.

8. Smart Cities: As part of smart city initiatives, the system can be integrated into urban infrastructure for automated irrigation of public parks, green spaces, and streetscapes. It enhances the aesthetic appeal of the city while promoting sustainability and environmental stewardship.

CONCLUSION

In conclusion, the "Smart Irrigation System using 8051" project presents a practical and effective solution for optimizing irrigation processes in agriculture and other related fields. By leveraging microcontroller technology, sensor data, and automation, the system ensures efficient water usage, promotes healthier plant growth, and contributes to sustainability efforts.

Through the integration of components such as the 8051 microcontroller, moisture sensor, ADC, LCD display, and motor driver, the system can accurately monitor soil moisture levels, make intelligent irrigation decisions, and provide real-time feedback to users. This not only simplifies irrigation management but also enhances crop yield and conserves water resources.

The project has a wide range of applications, from agriculture and greenhouse farming to landscaping, urban gardening, and environmental monitoring. It can be adapted for use in various settings, including rural farms, urban gardens, research facilities, and smart city initiatives, making it a versatile and valuable tool for sustainable water management.

In conclusion, the Smart Irrigation System represents a significant advancement in agricultural technology, offering an efficient, reliable, and user-friendly solution for irrigation management. It has the potential to revolutionize the way we approach water usage in agriculture and contribute to a more sustainable future for farming and environmental conservation.

APPENDIX

```
ORG 0000H
        SJMP 0030H
ORG 0030H
        ACALL LCD_INITIALIZE                ; LCD INITIALIZATION SUBROUTION

        MOV DPTR,#MESSAGE1                  ; SOIL MOISTURE
        ACALL DISPCH2                       ; display on LCD
        MOV A,#0C0H                         ; command for lcd 2nd line
        ACALL CMD                           ; command routine for LCD
        MOV DPTR,#MESSAGE2
        ACALL DISPCH2
        ACALL DELAY
        MOV A,#01H ;command for clear lcd
        ACALL CMD
        MOV A,#80H ;command for first line
        ACALL CMD
        MOV DPTR ,#MESSAGE3
        ACALL DISPCH2
        ACALL DELAY1
        MESG_VAL: MOV A,#01H                ; CLEAR THE LCD
                ACALL CMD
                MOV A,#80H                  ; START FROM THE FIRST LINE
                ACALL CMD
                MOV DPTR ,#MESSAGE4         ; moisture value
```

```
LCALL DISPCH2
REPEAT:                           ;SETB P2.0 ;
                                  CS for adc
                                  ;SETB P2.4
                                  ; RD for adc
                                  ;SETB P2.3
                                  ;WR for adc
                                  ;SETB P2.2
                                  ;INTR for adc
ACALL ADC_READ                    ; return the ADC value in A reg
MOV R4,A
LCALL BCD                         ; hex to bcd conversion FF > 255
LCALL UNPACK                      ; BCD to ASCII
                          ;returns value in 62h, 61h and 60h location ;
LCALL OUTPUT                      ;display the result
MOV A,R4                  ;check for RELAY on if 5%%<percent<20%
CJNE A,#00D,CHK2          ;5%
CHK2:JNC OK               ;check carry flag
SJMP LED_CHK
OK:CJNE A,#80D,CHK3       ;20%
CHK3:JNC LED_CHK
LCALL TILL
SJMP MESG_VAL
LED_CHK:CJNE A,#60D,LCHK          ;LED ON FOR PERCENT>60%
LCHK:JNC L_ON                     ;CHECK carry flag
SJMP BACK
```

```
    L_ON:LCALL LED_R
    BACK: SJMP REPEAT
ADC_READ:
    ;CLR P2.0 ;CS=0 ;CS = 2.0 , Rd = 2.4 , WR = 2.3 , INTR = 2.2
    ACALL DELAY
    CLR P2.3                        ;WR
    SETB P2.3                       ; WR
    ;SETB P2.0                      ;CS=1
    HERE1: JB P2.2, HERE1           ; INTR PIN POLLING
    CLR P2.0                        ;CS=0
    CLR P2.4                        ; RD
    MOV P3,A
    ;SETB P2.0
    SETB P2.4
    SETB P2.2
    MOV R5,A
    ACALL PERCENT
    RET

    PERCENT:MOV B,#51D          ; LET A BE x
    DIV AB                      ; x/51
    MOV 75H,B                   ; 75H = x%B
    MOV B,#20D                  ; B = 20D
    MUL AB                      ; A = (20x)/51
    MOV 50H,A                   ; 50H = (20x)/51
    MOV A,#100D                 ; A = 100
```

```
	SUBB A,50H				; 100-((20X)/51)
	MOV R0,75H				; R0 = x%B
	CJNE R0,#25,CK ;
	CK:JNC check_3_4
	SJMP Check_1_4
	check_3_4:CJNE R0,#38,CK2
	CK2:JNC AD
	SUBB A,#9
	sjmp T
	AD:SUBB A,#15
	SJMP T
	Check_1_4: CJNE R0,#12,CK3
	CK3:JNC SD
	SJMP T
	SD:SUBB A,#5
	T:RET

;PERCENT:
;MOV A, ADC_VALUE
BCD:MOV A,R4				;hex to bcd conversion
	MOV R3,A
	CJNE A,#00H, SKIP2
	MOV R6,A
	MOV R7,A
	RET
	SKIP2: CLR A
```

```
MOV R7,A
UP1: ADD A,#01H
DA A
JNC SKP
INC R7
SKP: DJNZ R3,UP1
MOV R6,A
RET


UNPACK:MOV A,R7                    ;unpack bcd values
ADD A,#30H
MOV 62H,A
MOV A,R6
ANL A,#0FH
ADD A,#30H
MOV 60H,A
MOV A,R6
SWAP A
ANL A,#0FH
ADD A,#30H
MOV 61H,A
RET

OUTPUT:MOV A,#0C0H                 ; command for lcd 2 nd line
LCALL CMD                          ; command routine for LCD
```

```
    MOV A,62H
    LCALL DISP                          ; display a byte on LCD
    MOV A,61H
    LCALL DISP
    LCALL DELAY
    MOV A,60H
    LCALL DISP
    LCALL DELAY
    MOV A,#25H
    LCALL DISP
    LCALL DELAY1
        ;MOV A,R5
        ;ACALL DISP
    RET


TILL:MOV A,#01H                         ;function for motor
        LCALL CMD
        MOV A,#80H                      ;command for first line
        LCALL CMD
        MOV DPTR,#MOTOR                 ; message pointer
        LCALL DISPCH2
        SETB P0.7 ;MOTOR on
        LCALL DELAY1
        LCALL DELAY1
        CLR P0.7
```

```
LCALL DELAY1
LCALL DELAY1
LCALL DELAY1
LCALL ADC_READ                     ; return the ADC value in A reg
MOV R4,A
LCALL BCD
LCALL UNPACK
MOV A,#01H
LCALL CMD
MOV A,#80H
LCALL CMD
MOV DPTR ,#MESSAGE4
LCALL DISPCH2
LCALL OUTPUT
MOV A,R4
CJNE A,#28H,CHK                    ;40%
CHK:JC TILL
LED_R: SETB P0.2 ;LED
        SETB P0.3
        SETB P0.4
        SETB P0.5
        LCALL DELAY1
        CLR P0.2
        CLR P0.3
        CLR P0.4
        CLR P0.5
```

```
        RET

LCD_INITIALIZE:
    MOV A,#38H
    ACALL CMD
    MOV A,#0EH
    ACALL CMD
    MOV A,#01H
    ACALL CMD
    MOV A,#06H
    ACALL CMD
    MOV A,#80H
    ACALL CMD
    RET

CMD:
    LCALL READY
    MOV P1,A
    CLR p2.7                ; low on RS
    CLR p2.6
    SETB p2.5               ; high to low on En line
    ACALL DELAY
    CLR p2.5
    RET
```

```
DISP:LCALL READY
        MOV P1, A
        SETB p2.7                    ; high RS
        CLR p2.6                     ;; low RW
        SETB p2.5                    ; high to low En
        CLR p2.5
        RET

DISPCH2: CLR A
        MOVC A,@A+DPTR
        JZ EXIT
        ACALL DISP
        INC DPTR
        SJMP DISPCH2
        EXIT: RET

READY:CLR 0A7H                       ;read busy flag
        MOV P1,#0FFH
        CLR 0A5H
        SETB 0A6H

WAIT:CLR 0A7H
        LCALL DELAY
        SETB 0A7H
        JB p0.7,WAIT
        RET
```

```
DELAY: MOV R6,#0ffh                          ;delay routine for firing
        HERE3: MOV R5,#0ffH
        REPEAT1: DJNZ R5,REPEAT1
        DJNZ R6,HERE3
        RET

DELAY1: MOV R5,#0AH                          ; two loop delay
        BK: MOV R1,#0FFH
        BACK1: MOV R0,#0FFH
        BACK2: DJNZ R0,BACK2
        DJNZ R1,BACK1
        DJNZ R5,Bk
        RET

 ORG 0300H                                   ; the messages to be displayed on the LCD

MESSAGE1: DB "SOIL MOISTURE",0, '$'
MESSAGE2: DB "DETECTION",0, '$'
MESSAGE3: DB "PLACE THE PROBE",0, '$'
MESSAGE4: DB "MOISTURE VALUE",0, '$'
MOTOR:DB "MOTOR ON",0, '$'

END
```