Wireless Notice Board

Engineering Design Project

Semester 2

De Silva A.L.U.P.	200105F
De Silva W.H.P.	200114G
De Zoysa T.I.R.	200115K
Deshan K.A.G.D.	200117T

Abstract

We decided to build a smart noticeboard for our design project. The main components are the HC-05 Bluetooth module, LCD display module, Atmega328p microcontroller, and the 5V voltage regulator (7805). A dedicated application on the smartphone transmits the message which is received by the Bluetooth module. It is displayed in the 16X2 display using the microcontroller. The voltage regulator is used to provide a constant power output of 5V from the 9V supply. Altium and SolidWorks were used for PCB and enclosure designing respectively.

Introduction and Method

In Sri Lanka in most places, the notice boards we use are still either paper-based where you need to change the paper every time information changes or use black or whiteboards. Our product is mainly aimed towards grocery stores. The price of grocery items is rapidly changing these days due to the negative effects of inflation. The workers have to reprint and replace the paper-based price tags every time which can be troublesome. If our wireless display is used, the workers can remotely update the displayed price an unlimited number of times using an application on their mobile phones.

The product is connected to the mobile phone via Bluetooth. We use the “Arduino Blue Control” app in the google play store to transmit messages to the display from the mobile phone. We use short-range wireless communication typically in the range of 10m using the 2.5GHz band to exchange files.

The received input is sent to the Atmega328p microcontroller which processes it and rearranges the data into a predetermined form specified in the programming of the microcontroller. Then it routes the message to the LCD display.

The LCD can display 32 characters altogether with 16 characters on each of the two lines. It displays the message starting in a predetermined position and line. However, if the message length is higher than the maximum number of characters which can be written on the line some characters will be omitted.

Components

1. Microcontroller

The microcontroller we used was the ATmega328p 28-SPDIP package. This is an 8-bit, RISC-based microcontroller with 32KB ISP Flash memory, 1024B EEPROM, 2KB SRAM and 23 general I/O pins. Also, it consists of a 6-channel 10- bit A/D converter. This device operates under 1.8-5.5 volts.

The key points we considered to select this microcontroller are its low power consumption, inbuilt ADC and ease of programming. We used an Arduino board to program the microcontroller, then remove it from the Arduino uno board and place it in our PCB. The outer appearance and the pin diagram of ATmega328p 28-SPDIP are shown in figure 1 and figure 2 respectively.

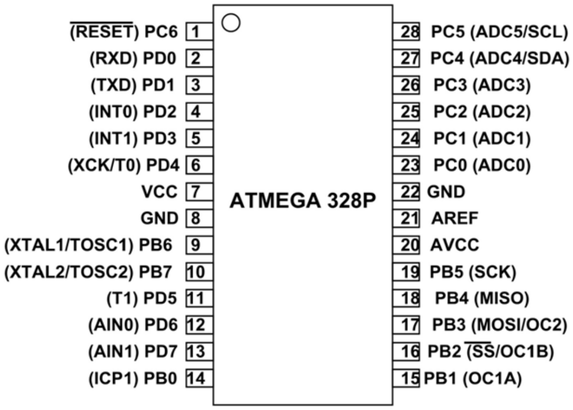


Figure 1

Figure 2

2. LCD Display

This display was selected due to its compliance with our requirements. The 16x2 LCD display was used with an I2C module. Images of the LCD and the I2C module are shown in figure 3 and figure 4 respectively.

Figure 3

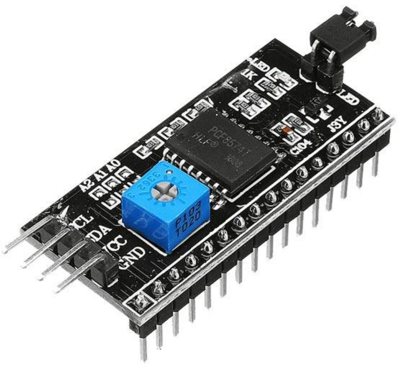


Figure 4

3. Bluetooth Module

HC-05 is used to connect small devices like mobile phones using a short-range wireless connection to exchange files. It uses the 2.45GHz frequency band. The transfer rate of the data can vary up to 1Mbps and is in the range of 10 meters. The HC-05 module can be operated within 4-6V of the power supply. It supports the baud rate of 9600, 19200, 38400, 57600, etc. Most importantly it can be operated in Master-Slave mode which means it will neither send nor receive data from external sources. When connecting password is either 1234 or 0000.

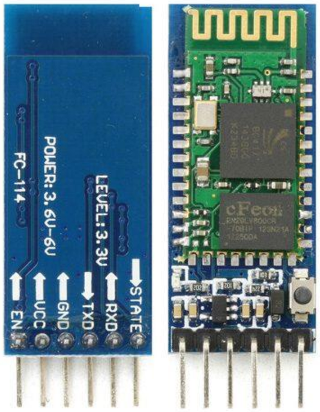


Figure 5

4. Regulator

An LM7805 voltage regulator is used to step down the 9V input to a constant 5V output voltage. The regulator is used with a heat sink to dissipate the power losses.

Figure 6

5. Other Components

Component	Value	Quantity
Resistor	470 nF	1
	100 nF	2
	22 pF	2
Oscillator	16 MHz	1
Switch		1
Battery	9V	1
Battery socket	9V	1

Cost

Item	Cost (LKR)
AT328P	1700
LCD	760
HC-05	1280
I2C	470
LM7805	60
PCB Printing	500
Enclosure Printing	2870
9V battery	345
Other	500
Total Cost	8485 LKR

Discussion

Initially, the 8051 microcontroller (AT89C51) was chosen due to the unavailability of the ATmega328p microcontroller. But later we found out that a separate programmer is needed to program 8051 microcontrollers. Also, it does not allow serial communication, so we found it difficult to program it even by connecting to the breadboard. Therefore, we had to shift to Atmega328p.

We decided to use a 16MHz external oscillator instead of an 8MHz internal oscillator to maintain a higher clock rate and a higher accuracy.

Earlier we tried connecting the LCD display using 4-bit mode which uses only 4 pins for data transfer. But we found it really hard when routing the PCB since we have to stick to a single-layer PCB. Then we went with the I2C module, which uses only 2 pins for data transfer between the microcontroller and the LCD. Therefore I2C converter is chosen for the project as it uses less number of pins of the microcontroller than the 4-bit mode.

Acknowledgement

We would like to extend our heartiest gratitude to our lecturers, Dr Ajith Pasquel and Dr Samiru Gayan, for their valuable guidance. We would like to thank each and every person who has helped us even in a very small manner in order to achieve good results in this project. We would like to especially thank Mr Saliya Dinusha who is currently a third-year undergraduate at our department, for his valuable comments and guidance, especially in PCB design. We must also thank all the lecturers, instructors, and other academic staff whose contribution indirectly helped us a lot in the completion of this project successfully. Finally, we would like to thank our fellow batch mates for sharing their knowledge and experience during the hard times.

References

1. *Kumar A, "E-Voice Based Notice Board Using ATmega328"*

Available : http://www.ijareeie.com/upload/2020/may/32_E-Voice__NC.PDF

2. *T. Elizabeth Darryl Jacob, K. Indumathi, "Real time Digi notice board using IoT"*

Available: https://www.ijert.org/research/real-time-digi-notice-board-system-using-iot-IJERTCONV5IS17021.pdf

3. *Pooja Pawar, Suvarna Langade, Mohini Bandgar, "IoT based digital notice board using Arduino"*

Available: https://www.irjet.net/archives/V6/i3/IRJET-V6I3875.pdf

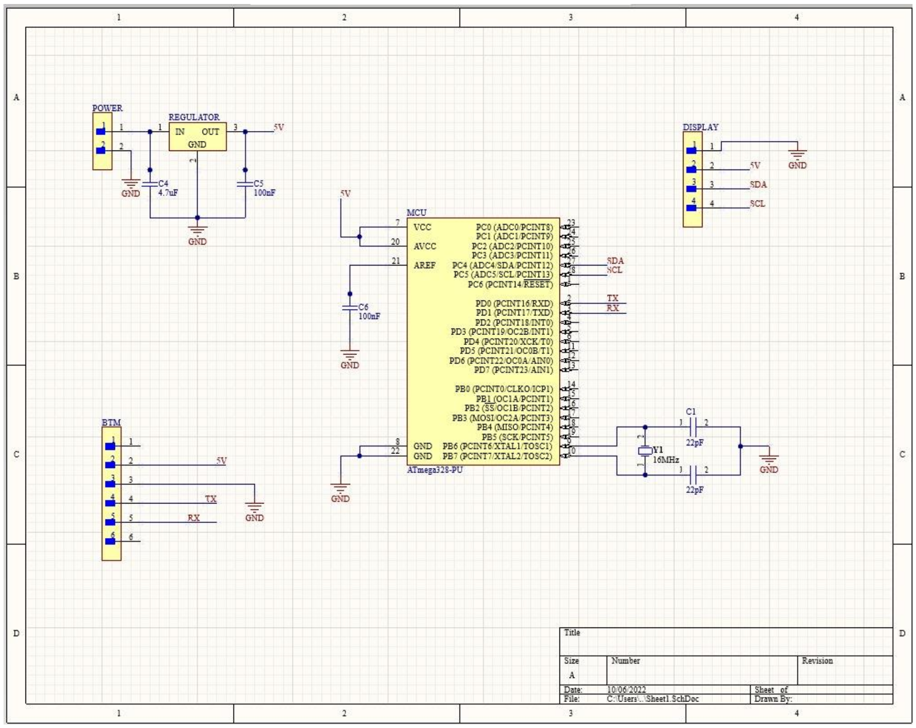
4. *E.G. Projects, "Difference between interfacing character 16×2 lcd in 4-bit and 8-bit mode with microcontroller"*

Available : Difference between interfacing character 16x2 lcd in 4-bit and 8-bit mode with microcontroller (engineersgarage.com)

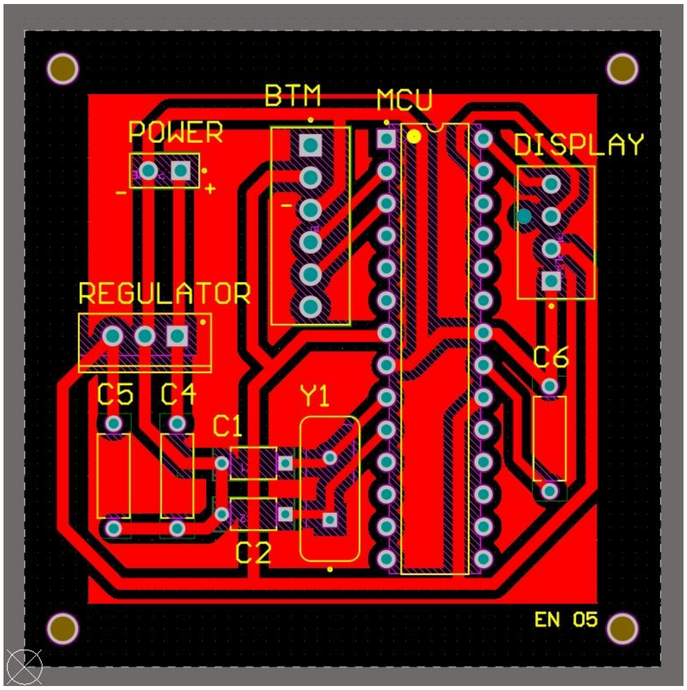
Available: https://youtu.be/PqFtSpAXB9Q

Appendices

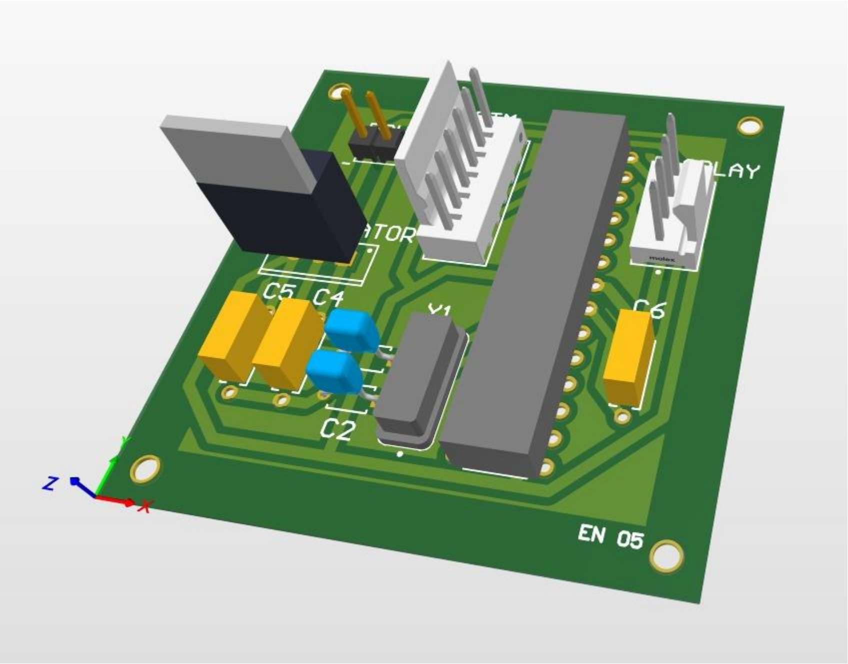
1. Appendix 1 – PCB Schematic



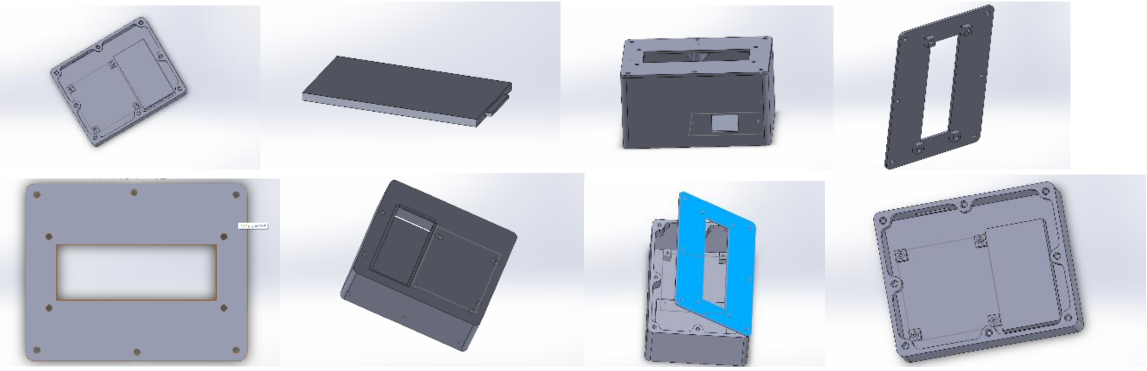
2. Appendix 2 – PCB Routing



3. Appendix 3 – PCB 3D View



4. Appendix 4 – Enclosure Design



5. Appendix 5 – Code

```
 #include <Wire.h>
#include <LiquidCrystal_I2C.h>

LiquidCrystal_I2C lcd(0x27, 16, 2);

void setup()
{

  lcd.begin();



  lcd.backlight();
  lcd.clear();

  //print welcome message
  lcd.setCursor(4,0);
  lcd.print("Welcome");
  delay(2000);
  lcd.clear();
  lcd.setCursor(5,0);
  lcd.print("Keels");
  lcd.setCursor(3,1);
  lcd.print("Katubedda");
  delay(2000);
  lcd.clear();

}

//method to print name of the item
void SetName(char name){
  lcd.setCursor(0,0);
  lcd.print(name);
}
//method to print quantity
void SetQuantity(char quantity){
  lcd.setCursor(0,1);
  lcd.print(quantity);
}
//method to print price of the item
void SetPrice(char price){
  lcd.setCursor(11,1);
  lcd.print(price);
}

int index=1;

void loop() {
  // set the cursor to column 0, line 1
  // (note: line 1 is the second row, since counting begins with 0):
  if(Serial.available()){
    char message=Serial.read();

    //code to clear the display
    if(message=="Clear" || message=="clear"){
      lcd.clear();
    }
    else{
      if(index==1){
      SetName(message);
      index=index+1;
    }
      if(index==2){
      SetQuantity(message);
      index=index+1;
    }
      if(index==3){
```

```
      SetPrice(message);
      index=1;
    }

    }

}


}
```