RFID Based Attendance System

Raj Lavingia

Final Project Report

ECEN 5613 Embedded Systems Design

May 05,2017

1 INTRODUCTION

The applications using Radio Frequency Identification (RFID) increasing nowadays are seen being applied in various fields such as transportation, agriculture, industries etc. Organizing large professional gatherings such as an International technical exhibitions, conferences, competitions, training events, etc. is a challenging task where a number of participants/delegates can be at the range of hundreds or even thousands. The event organizing committee has to correctly estimate an interest to sessions among the delegates, and to allocate premises, facilities and equipment to different topic sessions, presentations, demonstrations, etc.

Moreover, for daily attendance which is a critical issue in many companies. It plays an important role in managing salaries of employees in companies, to keep a track of certain goods, to guide the parents about their students activities in schools, colleges. The process of taking attendance through pen and paper has become an outdated approach and sometimes it also involves a lot of fake entries which leads to improper evaluation. To overcome all these drawbacks, I intend to develop a system which not only removes the tedious task of tracking attendance manually but also helps in maintaining a system which will help an organisation in appropriate evaluation, regularity, punctuality in attendance.

1.1 System Overview

The attendance of an employee will be taken with the help of unique RFID tag given to each employee and the time will be recorded and stored into the database with the help of RTC(Real Time Clock) .A welcome message will be spoken after every clock in time with the help of Text To Speech Module. According to the number of hours worked, total salary will be calculated on its own and paid at the end of the month. The system records details such as arrival, departure of employees and also their personal records. The employee's designation, department, shift hours, leave status, compensations can also be linked to the system and the system can be made more automated.

2 Technical Description

2.1 Introduction to RFID

2.1.1 History

In 1973, Mario Cardullo's device, patented on January 23, 1973, was the first true ancestor of modern RFID. It was a passive radio transponder with memory. In 1983, Charles Walton first patented the abbreviation RFID. He earned $3 million from royalties coming from his patents. But unfortunately, his patent about RFID expired in mid 1990s. After that he invented proximity card with incorporated PIN code protection which he got patented in 2004.

2.1.2 What is RFID Technology?

It has 3 Components.

1) **RFID Tag**

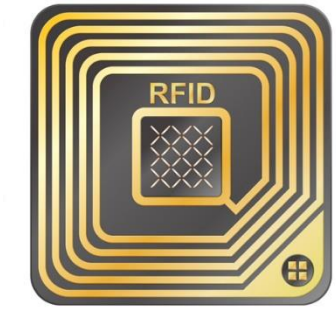


Figure 2.1 : Inside of RFID Tag

There are 3 types of RFID Cards :-

1) Active Tags: These types of tags have an on board Battery. They periodically transmit its ID signal. It does not require the Reader signals to get activated. It gets its power from the battery located on it.

2) Passive Tags: These types of tags do not have an on board battery. The tag uses the radio energy transmitted by the reader when it comes in contact of the reader to get activated and after it gets activated it is able to transmit data.

3) Battery Assisted Passive Tags: They have a small battery on board. These type of cards gets activated in the presence of RFID reader.

Data capacity can range from 16 bits on low end to as much as thousand bits on the higher end. Higher the storage capacity, higher is the price.

Frequencies are different for each type of cards in a way it communicates.
There are 3 types of frequencies :-

1) **Low Frequency** :- 125-134kHz
2) **High Frequency** :- Till 13.5MHz
3) **Ultra High Frequency** :- 300MHz-3GHz

Obviously , the prices of UHF card are higher than the low frequency cards.

2) **RFID Reader (Transceiver):-**

There are 3 types of RFID tags :-

1)Passive Reader Active Tag(PRAT) :- It is only capable of receiving radio signals. It can not transmit any signal on its own.

2)Active Reader Passive Tag(ARPT) :- It is capable of transmitting signals and receiving data from passive tags.

3)Active Reader Active tag(ARAT) :- Active tags awoken by an Active Reader only

3) **Antenna** :- It emits radio signals to activate the tag and read and write data to it They are available in different shapes and sizes. It is mostly packed with the transceiver and a decoder to become a reader. It emits radio waves in the range of 1 inch to 100 feet or more depending upon its capacity.

2.1.3 Working of RFID Module :-

The reader then converts the radio waves to a more usable form of data. Information collected from the tags is then transferred through a communications interface to a host computer system, where the data can be stored in a database and analysed at a later time.

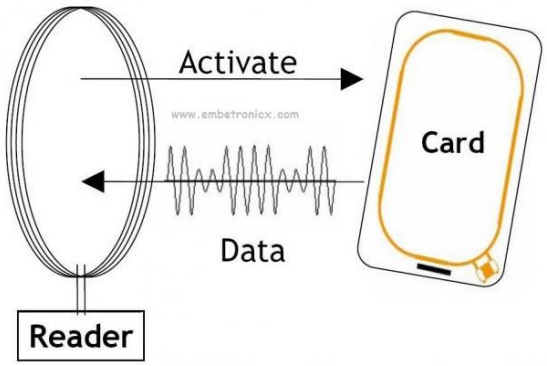


Figure 2.2 : Card -Reader Mechanism

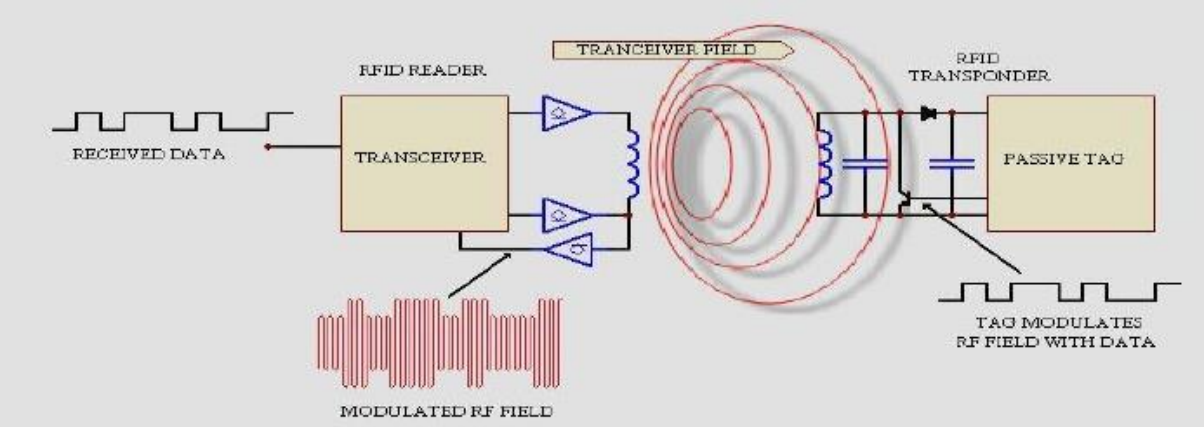


Figure 2.3 Reader and Passive Tag Mechanism

As shown in the figure, there is a passive tag which draws its power when it comes in contact of the transceiver. These passive tags do not have their own power but are able to generate enough power to transmit data when they are near the receiver. Once activated they send the data which is inside them.

2.1.4 Applications of RFID :-

1) Attendance system in offices, schools, colleges. Companies, Industries
2) Inventory Management
3) Patient Monitoring
4) Toll roads
5) Vehicle Identification
6) Access Control
7) IT Asset Tracking
8) Logistics and Supply chain Visibility and many more…

2.1.5 Advantages of RFID over Bar Coding :-

1) RFID avoids the limitations of barcode scanning, which requires line-of-sight access to each barcode and can only be used to scan one item at a time
2) They can be read from a variety of distances based on the type of tag and the use of a handheld reader or a fixed RFID reader combined with an antenna
3) A barcode can only be read, and the data can never be changed once it has been printed onto a label. RFID tags can be read and the data on the tag can be rewritten or modifies as needed.
4) RFID tags are durable and reusable
5) Barcodes are easily counterfeited, and the data itself is always readable. With RFID tags, your data is much more secure as the information has the ability to be encrypted. Also, it's much more difficult to replicate RFID tags recognized in your system.
6) RFID tags are capable of storing more data.

2.1.6 Disadvantages of RFID tags :-

1) Tags are application specific. No one tag fits all.
2) More than one tag can respond at the same time

RFID system can have 4 types of users :-

1)Admin :- He will have access to everything in the system. He can change things according to his wish.

2)Operator :- He will also have access level next to admin. Operator will operate the system according to the admin requirements

3)Employee :- He will having access to his personal information only like profiles, attendance, salary etc.

4)Guest :- He will have lowest level of access rights to the system.

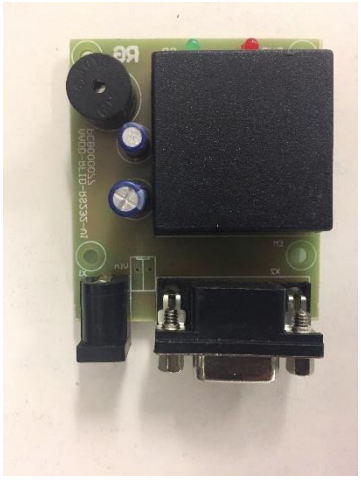


Figure 2.4(a): RFID Module Front Side

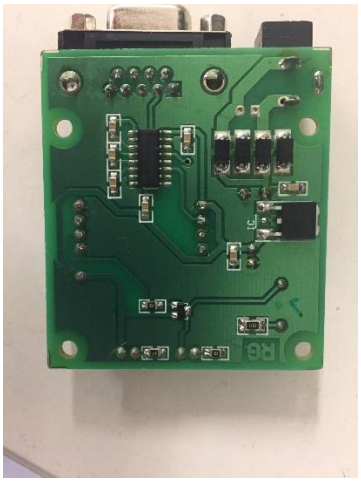


Figure 2.4(b): RFID Module Back Side

2.2 Text to Speech Module :-

The main purpose of this module is to convert the text to speech. This can be used as a welcome message for the employees. This module has contains an integrated chip called " PIC24FJ64GA002". It is a 16 bit flash microcontroller.This built in algorithm allows for the automatic real-time translation of English ASCII characters into allophone addresses . It's a 28 pin package with a supply voltage requirement of 5v.

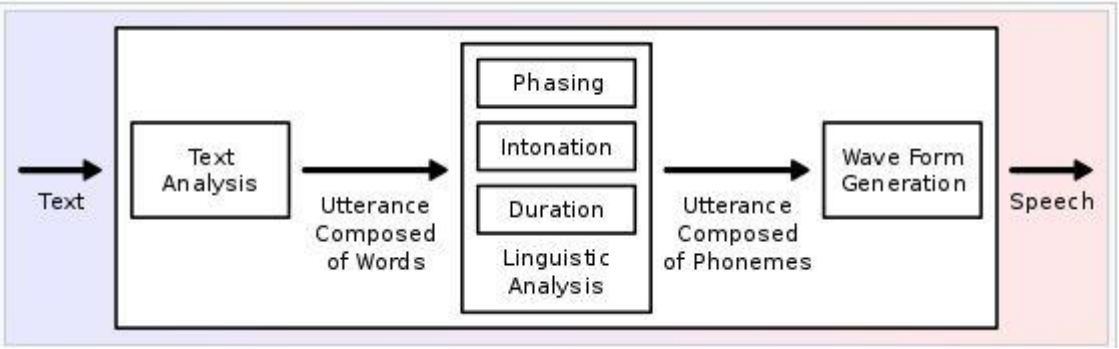


Figure 2.5 Text to speech analysis

The module also contains Digital to analog converter(DAC) and some power amplifiers for the sound amplification. The module has 2 LEDs. One is orange colour and another one green colour. The green led indicates that the module is getting power. Yellow led if on it means the module is speaking and if it is off than the module is not speaking.

The module has 4 lines at the end. 1st one is **ground**. 2nd one is **Vcc**. 3rd one is **TX-OUT** which is Typically connected to RXD pin of microcontroller or MAX232 or USB-TTL. 4th one is **RX-OUT**, Incoming serial data at 5V or 3V level, typically connected to TXD pin. The serial communication should be at 9600 bps, No parity, 1 stop bit. The module also has two 2 audio output pins. Out of which 1 pin is ground and other pin is AUD (For external audio amplifier or Headphone jack connection).The text which is written on the terminal is converted first to the hex data which is considered as a digital data which is given to the IC which converts it into the analog signal and outputs back. To be able to get the sound from the module whatever text written on a printf command must be ended with the **" \r " symbol** at the end of the text.

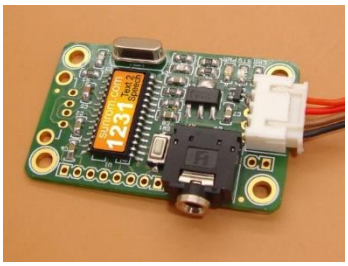


Figure 2.5(a) Front Side of TTS Module

Figure 2.5(b) Back Side of TTS Module

2.3 Real Time Clock (RTC)

RTC Module is used to monitor the real time in any applications. It is able to show the seconds, minutes, hours, date, month, year.

RTC devices are divided into 3 parts :-

1)Two – Wire Serial Interface (**I2C Bus**)

2)Three -Wire Serial Interface (**USB Bus**)

3)Four -Wire Serial Interface (**SPI Bus**)

In this project I have used the 2 wire serial interface (I2C Bus).

In this module, the IC used is **DB1307.**

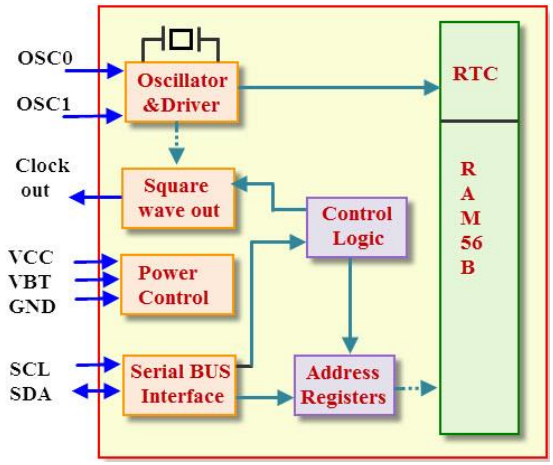


Figure 2.6 : Internal Architecture of RTC Module

A0,A1,A2 are the address pins of the RTC DB1307 chip, which can be used to communicate with the master device. Since there are 3 address pins, **2^3 = 8** . We can control eight devices with RTC interfacing with 8051 microcontroller using I2C protocol.

VCC and Ground :- There are 2 pins of VCC and Ground on that chip. The device is operated with **1.8v to 5.5v** range.

VBT :- This pin is a battery power supply pin which is connected to the cell of **3.3v** so that if the device turns off and when turned back on , we can still get the real time and the RTC does not stop when the microcontroller stops giving it the required power.

SCL :- SCL is a serial clock pin and it is used to synchronize data on serial interface.

SDL: It is a serial input and output pin. It is used to transmit and receive the data on serial interface.

Clock Out: It is an optional square wave output pin.

OSC0 and OSC1: These are crystal oscillator pins which are used to provide the clock signals to the RTC device. The standard quartz crystal frequency is **22.768KHzs.**

2.3.1 Device Addressing :-

I2C bus protocol allows many slave devices at a time.

RTC Device address is D i.e. 1101 which is given by the manufacturer. It has total of 8 bits out of which first four bits starting from MSB are 1101 which is fixed , after that 5th, 6th, 7th bit are A0, A1,A2 respectively which can be varied from "000" to "111" depending on the user and the last 8th bit is R/W which when set to "0" is used for write command and which when set to "1" is used for read command.

For example :-

Address of Read operation of RTC= "0xD3"

Address of Write operation of RTC= "0xD2"

2.3.2 Memory Register and Addresses :-

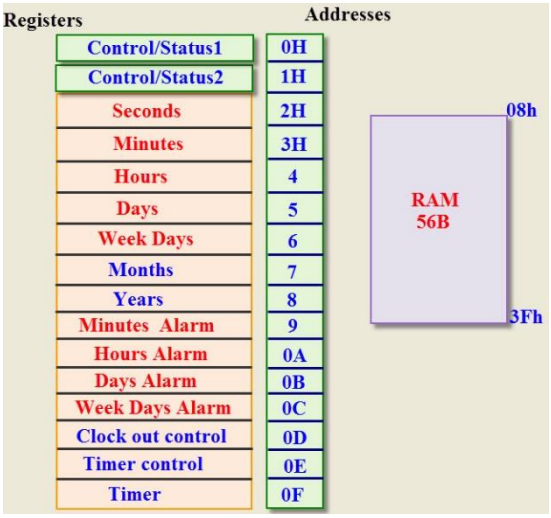


Figure 2.7 : Memory Register

RTC registers are located in address locations from 00h to 0Fh and RAM memory registers are located in address locations from 08h to3Fh. RTC registers are used to provide calendar functionality and drive time of day and to display the weekends.

Control/Status Registers:

DB1307 consists of two additional registers such as control/status1 and control/status2 which are used to control real time clock and interrupts.

Control/Status Register 1 :-

Figure 2.8 :Control/Status Register 1 8 bits Frame

TEST1=0 normal mode

=1 EXT-clock test mode

STOP=0 RTC starts

=1 RTC stop

TESTC=0 power on reset disabled
= 1 power on reset enabled

Control/Status Register2:

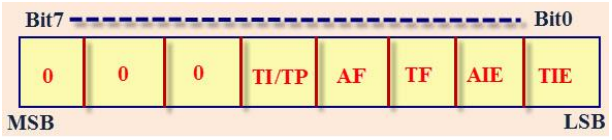


Figure 2.9 : Control/Status Register 2 8 Bits Frame

TI/TP= 0 INT active all the time
=1 INT active required time

AF=0 Alarm does not match
= 1 Alarm match

TF=0 Timer overflow does not occur

=1 Timer overflow occurs

ALE=0 Alarm interrupts disable
=1 Alarm interrupts enabled

TIE=0 Timer interrupts disable
=1 Timer interrupts enable

Transmitting Data Frame :-

Start	slave address	R/W	Control register Address	Control Status register1	Control Status register2	8-bit Data	ACK	Stop

Figure 2.10 : Transmitting Data Frame 8 Bits

Start: Primarily, the data transfer sequence initiated by the master generating the start condition.

7-bit Address: After that the master sends the slave address in two 8-bit formats instead of a single 16-bit address.

Control/Status Register Address: The control/status register address is to allow the control status registers.

Control/Status Register1: The control status register1 used to enable the RTC device

Control/Status Register2: It is used to enable and disable interrupts.

R/W: If read and write bit is low, then the write operation is performed.

ACK: If write operation is performed in the slave device, then the receiver sends 1-bit ACK to microcontroller.

Stop: After completion of write operation in the slave device, microcontroller sends stop condition to the slave device.

Receiving Data Frame:

Start	slave address	R/W =1	Control register Address	Control Status register1	Control Status register2	8-bit Data	ACK	Stop

Figure 2.11 : Receiving Data Frame 8 Bits

Start: Primarily, the data transfer sequence initiated by the master generating the start condition.

7-bit Address: After that the master sends slave address in two 8-bit formats instead of a single 16-bit address.

Control/Status Register Address: The control/status register address is to allow control status registers.

Control/Status Register1: The control status register1 used to enable the RTC device.

Control/Status Register2: It is used to enable and disable interrupts.

R/W: If read and write bit is high, then the read operation is performed.

ACK: If write operation is performed in the slave device, then the receiver sends 1-bit ACK to microcontroller.

Stop: After completion of write operation in the slave device, microcontroller sends stop condition to the slave device.

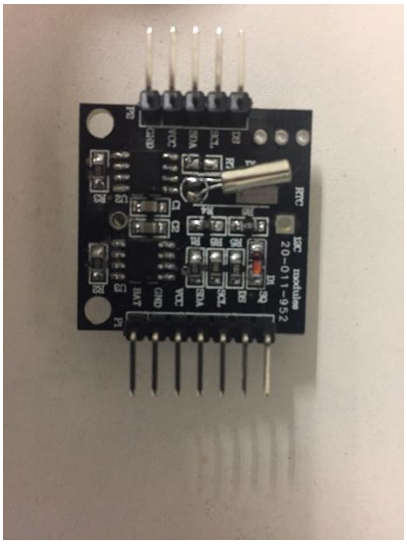


Figure 2.12(a) I2C Module Front Side

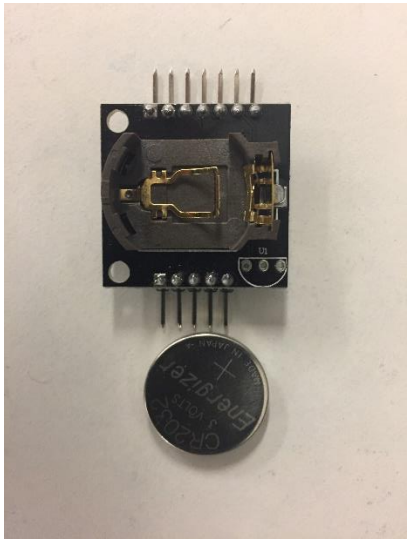


Figure 2.12(b) I2C Module Back Side

2.4 8051 Micro Controller Board :-

The **Atmel – At89c51RC2** (Referred directly as 8051 thorough out this document), an 8 bit MCU with 32KB flash memory and 256 bytes of standard RAM is used in this project as the main controlling unit. Even though it is capable to run in x2 mode at maximum clock frequency up to 40MHz, in this project I have used it in x1 mode from a clock source of 11.0592MHz crystal. In addition to bare minimum hardware requirements, 8051 development board kit developed through out the semester has many other hardware peripherals interfaced to it and enhanced its capability in many ways.

First of all I have designed one power supply circuit on this board which consists of power jack, bridge rectifier, LED in the initial phase. The output of bridge rectifier is given to IC **LM7805** which converts the 9v from the power adapter to 5v. This 5v is later given to the 8051 MCU. The port 0 of the 8051 is pulled up and given to **74LS373** latch which is later connected the NVSRAM.

The PORT 0 of 8051 is already multiplexed, which means it can send data and address both. To use the the external memory with it we need to latch the address. Therefore, the latch 74LS373 is used to interface external memory to 8051. For this project I have used TI's 32KB NVSRAM. Port 0 lines of MCU go to NVSRAM through the latch.

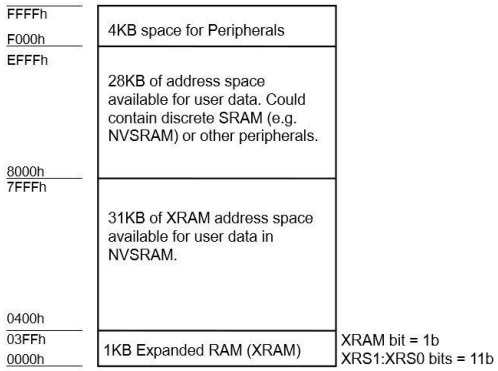


Figure: 2.13 Memory Mapping

8051 comes with 1KB XRAM which is programmable, in order to access the external memory , SPLD needs to be connected in a memory mapped configuration in order to get better performance. I have interfaced SPLD with 8051 and hence accessed the external memory and finally used that memory to run the LCD (**DMC16433 LCD Module by Optrex**).

The LCD has 14 pins out of which 1st is ground,2nd is VCC, 3rd is VEE for controlling the potentiometer for brightness adjustment of the LCD. After that 4th, 5th and 6th pins are EN,RS,RW. The enable signal allows LCD to accept commands from 8051. RS(Register Select) is used to select instruction memory. R/W is used for read and write to the LCD.

Basically, this LCD has 64 pixels in it, each row contains 16 pixels. There are around 10-15 in total hidden pixels which the company has provided if the user does not want to display it to the screen but wants it hidden in the LCD data.

The A15 is always set high while accessing the higher 32KB, is used to enable the LCD. A13 is RS and A12 is RW. The table is shown Below.

A15(EN)	A14	A13	A12	Address	Data/Instr	Read/Write
1	0	0	0	8000	Instruction	Write
1	0	0	1	9000	Data	Read
1	0	1	0	A000	Data	Write
1	0	1	1	B000	Instruction	Read

Table 2.1 Address allocation (Memory map)

```
/* *************** INPUT PINS *********************/
PIN 5    = A12; /*address pins*/
PIN 4    = A13; /*address pins*/
PIN 3    = A14;  /*address pins*/
PIN 2    = A15; /*address pins*/
PIN 7    = RD;  /*read pin */
PIN 8    = WR;  /*write pin */

/* *************** OUTPUT PINS *********************/
Pin 15   = Rs; /* Register select Pin */
Pin 14   = RW; /* read write pin */
Pin 13   = E;  /*Enable pin */

RW = (A15 & !A14) & ((A13  & A12)#(!A13 & A12));
Rs = (A15 & !A14 & A13 $ A12);

/* Enable Pin Logic */

E = (!RD # !WR)&(A15 & !A14 );
```

Figure: 2.14 Logic of SPLD

The remaining 8 pins of LCD are data pins ranging from A0 to A7. These pins are connected to the PORT 0 of 8051. After these connections are made, for initialisation of LCD there is a whole process which is shown below.

After the connections are made, to load the program into 8051 I have used FLIP software for the programming of 8051 through serial communication at baud rate of 9600bps. For this MAX232 IC is used. We can not directly connect the 8051 and the PC because they both understand different logics.The MAX232 IC enables logic level conversion between RS-232 standards and UART RX/TX standards. RS-232 output is connected to the DB9 serial port. This allowed me to make use of In system programmer to flash the 8051's code space through the serial port with the help of ATMEL FLIP utility on a host computer.

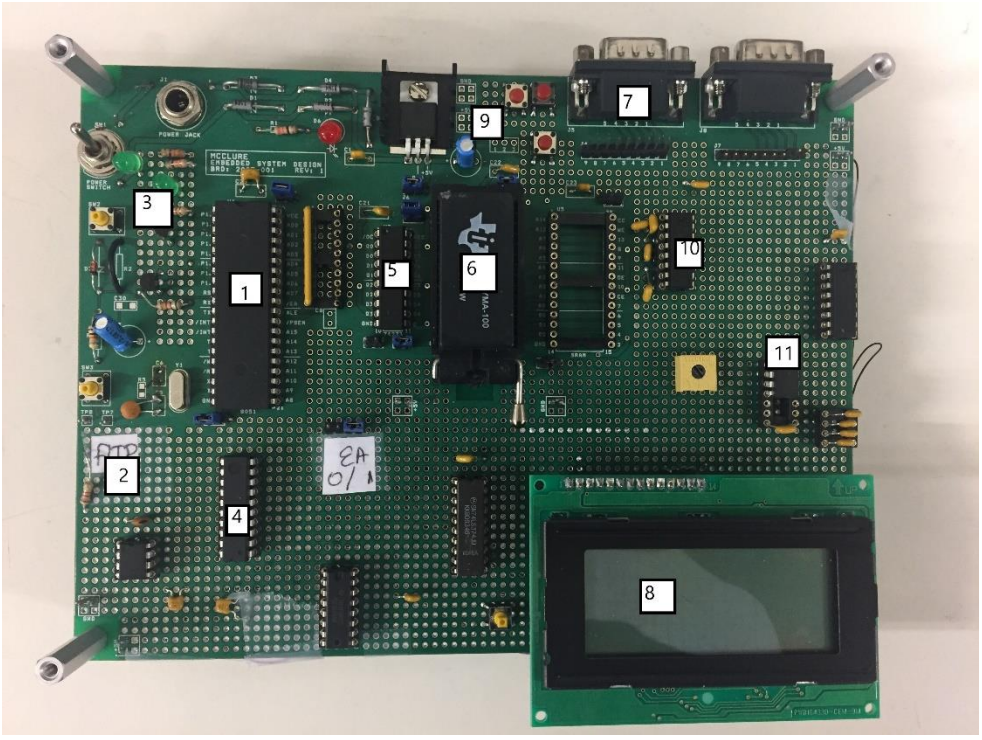


Figure 2.15 8051 Development Board

1)ATMEL AT89C51RC2 MCU

2) Bootloader Circuit

3)Power Circuit

4)SPLD – ATMEL ATF16V8C-7PU

5)74LS373 Latch

6)Texas Instruments BQ4011YMA-100 :32 KB NVSRAM

7) DB9 Serial Port

8)OPTREX DMC16433 LCD MODULE

9) SET TIMING BUTTONS

10) MAX-232 IC

11) I2C 24LC16B

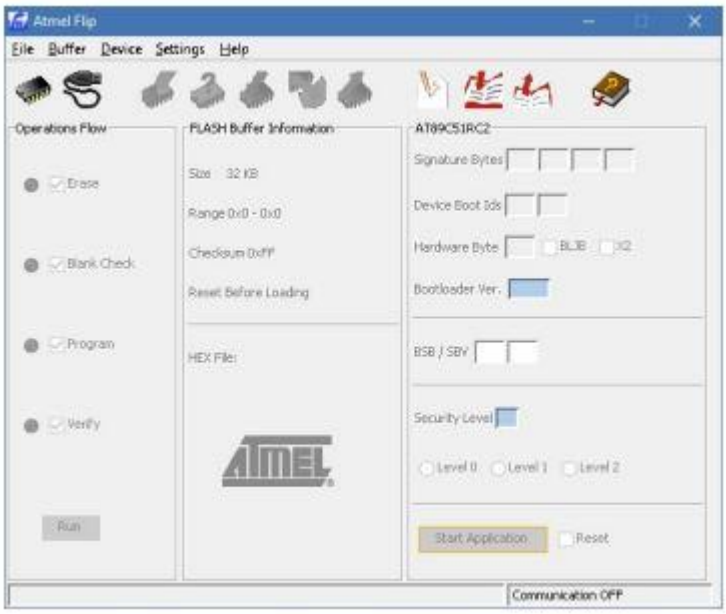


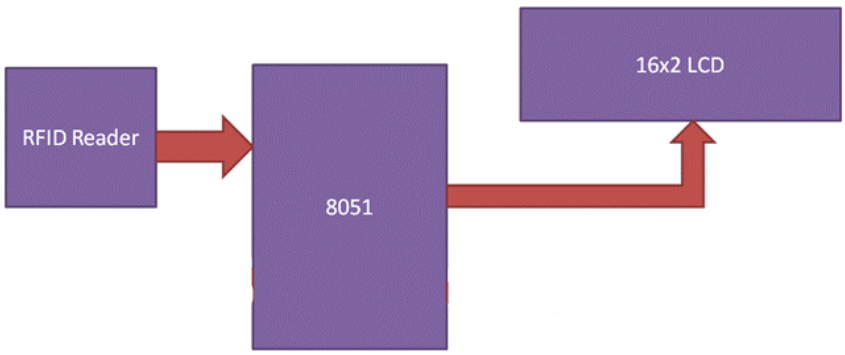
Figure 2.16 Atmel Flip utility

To program the 8051 , I have to first put it in the bootloader mode. For this, PSEN signal should be low when the 8051 comes out of the reset to execute the bootloader mode. For this, a push button has been connected to the PSEN pin to pull it low, so that if the push button is pressed and released, the 8051 would enter into bootloader mode and will be ready to be used by the ISP.

2.5 Interfacing RFID Module with 8051 :-

RFID module has 4 pins which contains TX , RX , VCC and Ground. The output of RFID module was given to the MAX 232 IC which is located on the module itself which was later given to the MAX232 located on the 8051 development board which was eventually connected with the 8051 chip's TX RX and finally a communication is made between the RFID module and the 8051.

Whenever , the RFID tag is put near the RFID module the tag gets activated through the signal received from the RFID reader and eventually send the data to the MCU. To know the data inside the Tag , there is a 10 digit code which is written on the top of the tag.



2.17 Interfacing of RFID and LCD with 8051

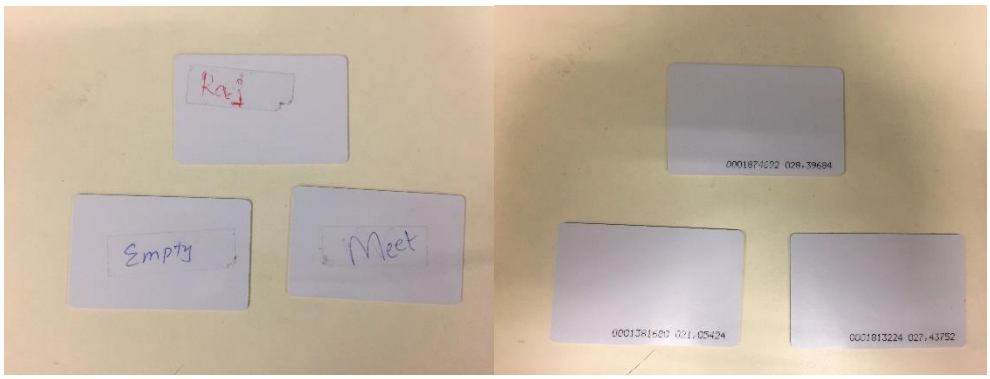
Lets suppose the card number 0200107D0D62 is read from the reader , then the 10 digit number on the card will be according to the process below :-

First 2 numbers “02” is the preamble

Next 8 Numbers are in hex, so we have to convert them into decimal. “00107D0D” = “1080589”

Final 2 numbers “62” is XOR value for (02 XOR 00 XOR 10 XOR 7D XOR 0D)

Hence the 10 digit card number on the top of the card will be 0001080589.



2.18 RFID cards with 10 digit number

2.6 RTC Interfacing with 8051 Board :-

For interfacing RTC with 8051 , I connected it through I2C interface. For continuous power, I have connected a 3v battery at the back side of the RTC module. I have connected the SDA and SCL pins to 8051 at the same pins where the SDA and SCL of I2C (24LC16B) is connected.

The IC 24LC16B is a 8 pin dip package with Pin 5,6 as SDA and SCL respectively. It is 16K serial EEPROM. Pins 1,2,3 are A0,A1,A2 which can vary from “000” to “111” depending on the user. There is also a write protection feature in this IC. When this pin is tied to VCC then it protects all the array from 000-7FF. There is a also noise protection system in this IC.

SCL and SDA lines have Schmitt Trigger and filter circuits which supress noise spikes to assure proper device operation, even on a noisy bus.

I2c is a half duplex meaning it can communicate with master and slave once at a time. The master can be multiple and so does the slave. This particular IC can have 128 slaves at a same time. We also use pull up resistors for this IC because the rate at which this IC can communicate is much slower than the rate at which 8051 communicates. So to maintain the rate of data transfer and to avoid the data loss we use Pull up resistors. We also use decoupling capacitor to avoid noise and to encourage less loss of data. We can transfer data

to I2c and read data from I2c. For that first of all there is Start bit which is followed by a default address which is “1010” which is then followed by the bits A0, A1,A2 and then at the end there is R/W bit. If “0” then it will write and if “1” it will read.The formation of this 8 bit is called collectively “ Control Byte”.

First of all, Master send the slave this address consisting of 8 bits, after receiving the address the slave sends an ACK bit to the master telling that I have received the address. After which the master sends the “Word Address” to the slave. Again the slave sends a ACK telling the master that it has received the “Word Address”. After receiving the Acknowledgement from the slave , master finally sends the Data to the slave and after receiving the data slave will send an ACK to the master. After ACK is received from the slave, master send a Stop Bit to the Slave indicating that the data transfer procedure has been completed.

Now, if the master wants to send a lot of data without the stop bit , he can do that also. This is called Page write. In this case, data is sent from the master on a continuous basis. Similarly, there is also a sequential read procedure where the data is read one after the other on a continuous basis. Other function is called random read where the slave sends one ACK bit to the master after every read instruction happens. So, we can write and read to a particular location starting from 000-7FF.

FIGURE 4-1: BYTE WRITE

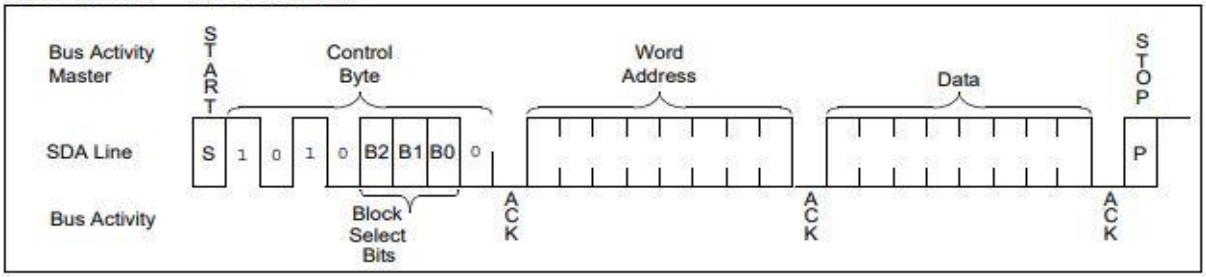


FIGURE 4-2: PAGE WRITE

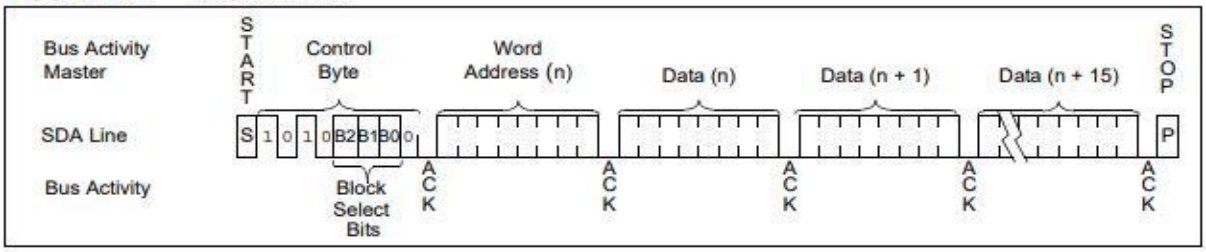


FIGURE 7-2: RANDOM READ

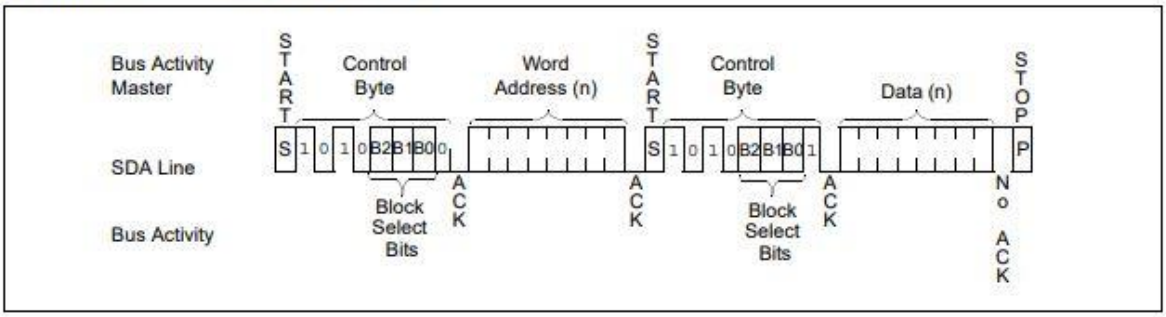


FIGURE 7-3: SEQUENTIAL READ

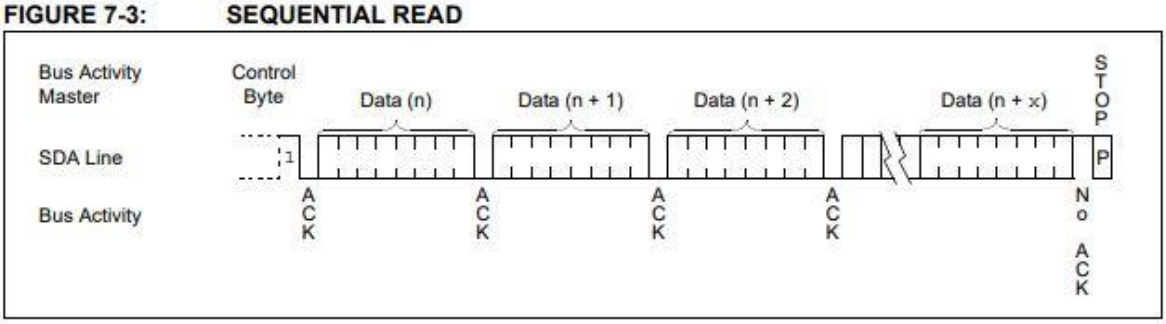


Figure 2.19 I2C communication of Read and Write

```
void I2CStart();
void I2CStop();
unsigned char I2CSend(unsigned char Data);
unsigned char I2CRead(char ack);
int BCDToDecimal(char bcdByte);
char DecimalToBCD (int decimalByte);
```

Figure 2.20 I2C Communication

Moreover, in the initial phase we have to set the clock or else it will run according to its default value only. So in order to set the clock , I have configured 3 buttons on the 8051 development board which are connected to 3 pins of 8051. Three buttons are to set, increment digit and increment value. To set the time, first of all we have to put the system into the clock set mode which can be done by pressing the set button at the time of initialisation of LCD. The cursor on the LCD will navigate you to change the time and date.

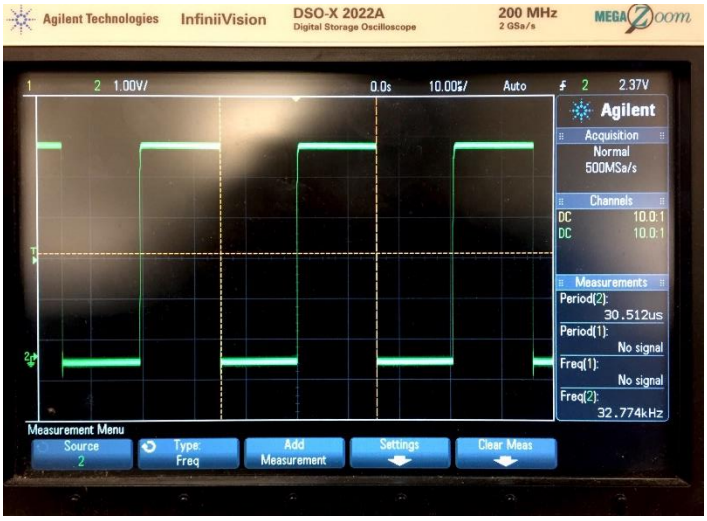


Figure 2.21 RTC Module Frequency

Every RTC has its own frequency. As shown in the figure, the RTC module which I have used for this project is of 32KHz approximately.

Setting Up the Time :-

To set the time, while the LCD is initialised the user has to press the “SET” button continuously so that it is short with the ground and let the system go into time setting mode.

```
void show_time()
{
 char var[5];char var1[5];char var2[5];

 *lcd_instruction_write = 0xD8;
   sprintf(var,"%d",hour);
   stringprint(var);
   sprintf(var1,":%d",min);
   stringprint(var1);
   sprintf(var2,":%d",sec);
   stringprint(var2);


}
```

Figure 2.22 To display current time function

```
void set_time()                                          //time set funct
{
 *lcd_instruction_write = 0x0C;
 while(k<7)
 {
  while(k==3)                                          //set date
  {
   x=year%4;
   if(inc==0)
   {
       date++;while(inc==0);
   if(x==1 && mon==2 && date==28){date=1;}              //check for 28 da
   if(x==0 && mon==2 && date==29){date=1;}              //check for 29 da
   if((date==31) && (mon==4) || (mon==6) || (mon==9) || (mon==17)){date=1;}
   if(date==32){date=1;}
   show_time();
   }
   if(next==0)
   {
     k=5;

     while(next==0);
     }                                                 //check for next digi
 *lcd_instruction_write =(0x85);
  }
```

Figure 2.23(a) Setting time with the help of buttons

```
337
338     while(k==2)
339     {
340      if(inc==0)
341      {mon++;while(inc==0);
342      if(mon==13){mon=1;}
343      show_time(); }
344      if(next==0){k=3;
345      while(next==0);
346
347      }
348      *lcd_instruction_write =(0x88);
349     }
350
351     while(k==1)
352     {
353      if(inc==0)
354      {year++;while(inc==0);
355      if(year==30){year=0;}
356      show_time();      }
357      if(next==0){k=2;
358
359      while(next==0);}
360    *lcd_instruction_write = (0x8d);
361     }
362
363     while(k==5)
364     {
365      if(inc==0)
366      {hour++;while(inc==0);
367      if(hour==24){hour=0;}
368      show_time();}
369      if(next==0){k=6;
370      while(next==0);}
```

Figure 2.23(b) Setting time with the help of buttons

```
370      while(next==0);}
371      *lcd_instruction_write = (0xc5);
372     }
373
374     while(k==6)
375     {
376      if(inc==0)
377      {min++;while(inc==0);
378      if(min==60){min=0;}
379      show_time();}
380      if(next==0){k=10;
381      while(next==0);}
382      *lcd_instruction_write = (0xc8);
383     }
384    }
385   }
```

Figure 2.23(c) Setting time with the help of buttons

3 USER INTERFACE

The final part of the project after interfacing all the modules was to create a user interface.

I have included the following functions for this project:-

1) Register an attendance
2) Clock in time
3) Clock out time
4) Register a new card
5) Delete attendance
6) View Logbook

When the system starts a message will be displayed on the LCD saying " RFID Based Attendance System ". At this time if one wants to set the time , then he can press the "set" button on the 8051 development board and can set the time and date. If he does not want to set the time then he can ignore pressing that button and the by default or the previously set time will appear on the screen. If you have set your time once then you don't need to set the time again and again after every restart. The 3v cell provided at the back of the RTC module will take care of the time and will never stop running even after the power shut off from the 8051 board.

At the initialisation of the LCD, the text to speech module comes into picture and speaks "How are you today?" , " Welcome to class" .This can be heard with external speakers connected at the output of the audio.

After that , on the PUTTY , there will be user interface displayed giving options in the following manner :-

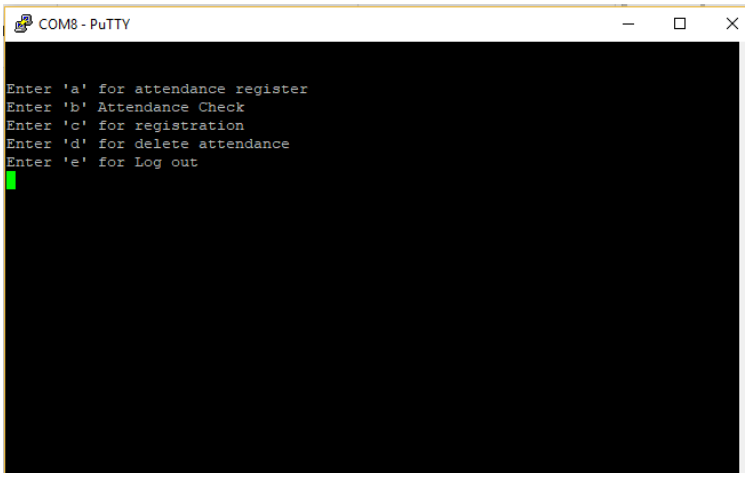


Figure 3.1 Initialisation of system

For this particular project I have created a database of 2 people and the third card I have kept for new registration.

Whenever the user presses "a" , First of all it will ask the user to place the card near the module. After the user places the card near the module there will be 3 possibilities for

my project. Either it will be one of the 2 registered cards or the unregistered one. If the card is not registered then it will display " Your card is not registered . please register it".

```
Enter 'a' for attendance register
Enter 'b' Attendance Check
Enter 'c' for registration
Enter 'd' for delete attendance
Enter 'e' for Log out
You Entered
Place Your Card


Your Card is not registered. Please register it.
```

Figure 3.2 Card if not registered

If the card is already registered , which here I have registered under names "Raj" and "Meet" then after placing the card near the RFID module , the LCD will show the particular person is present and it will show the time at which the card was placed and it will get recorded in the database. It will also show the unique if which is 12 digit number and is different for every user.

Figure 3.3 LCD Display with name and time (Clock in Time)

Here the unique id of raj is "04001C9B0487". The time which is shown at the end of the screen is the check in time. Now, after every check in time , there has to be a check out time. If the user does not have the check out time than he can not register for a new attendance.

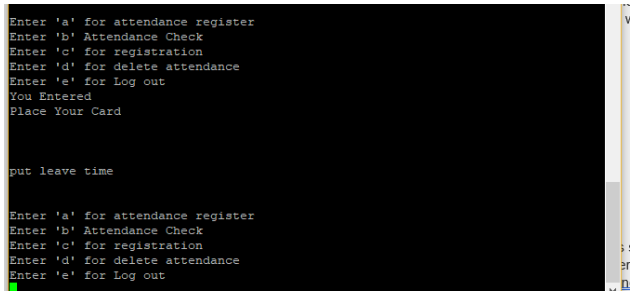


Figure 3.4 If clock in time entered two times in a row without putting leave time

After clock in, while doing clock out it will show a message that "raj is leaving" and will display the time when he has left at the bottom of the screen.

Figure 3.5 LCD display of person leaving (Clock Out Time)

So as you have seen the clock in time is 18:15:54 and the clock out time is 18:16:5 . Whenever the person wants to see the log, like the numbers of hours, seconds he worked on that day he can opt for option “b” which has a attendance check and a log book. It will display the total number of attendance and the time he worked on that day.

```
Presence Book
Attendance done at 1st time 18:15:54

Leave Book
leaving done at 1st time 18:16:5


Seconds worked in 1st day 11
Hours worked on 1st day 0
```

Figure 3.6 Log Book

Now if a person is not registered and taps the card reader than a system will prompt it to register the card.In order to register the person he will be asked for a password for registration as a security purpose because any person cannot randomly come into office and register himself. So if the correct password “8438” is entered from the user than only the system will let it register him. After entering the correct password , the system will ask for his / her name and than store that name into the database .It will also show the unique 12 digit id of the user and store it into the system. So every time when the newly registered user places the card than the system will show his/her name on the screen which he wrote at the time of registration.

```
Do you want to register yourself ? Y or N
You Entered :
Y
We have checked your card. It is not registered.
Enter the pass code for registration

Password is correct

This is your unique id :
040015153034
Write your name
user entered name is
David
You are now registered.
```

Figure 3.7 Registration Process

As shown in the figure , we have registered a new card in the name of “David”. Now whenever David will tap his card near the module it will show “David is present” with the time when he checked in. Similarly, we can also find the check out time for David. Thus , now David’s card will function similarly as the pre registered cards only.

Figure 3.8(a) : New user clock in **Figure 3.8(b) New User Clock out**

There is another feature of deletion of attendance which can be handled by the admin only. For this project I have made “Raj” as the admin. If any other person tries to remove the attendance of some person than he wont be allowed to delete it.

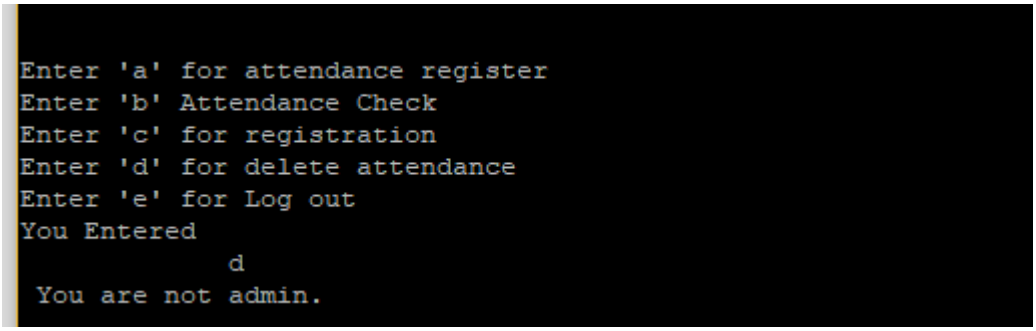


Figure 3.9 : Admin Protection for Deletion of Attendance

If admin enters the card, then also he will be asked for a password for double security purpose. The password which is set as “8538”. If this is entered then he will be asked to delete the attendance. The attendance will be deleted just one time. Here , I am displaying the 1 attendance of David turning into 0 after the admin delete his attendance.

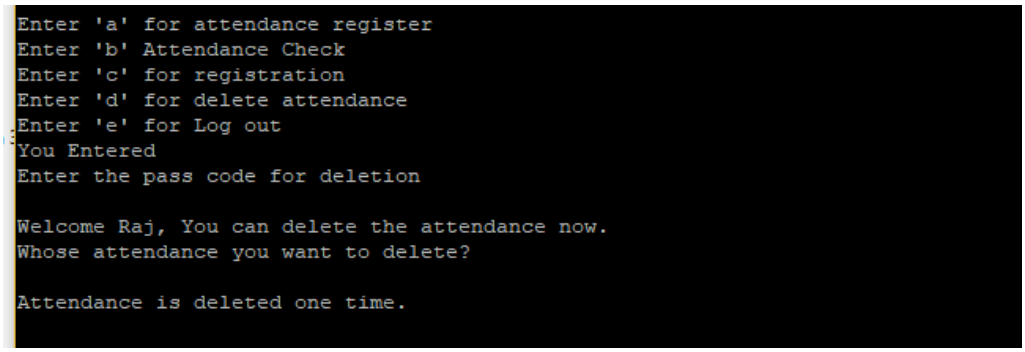


Figure 3.10 Admin Deleting the attendance procedure

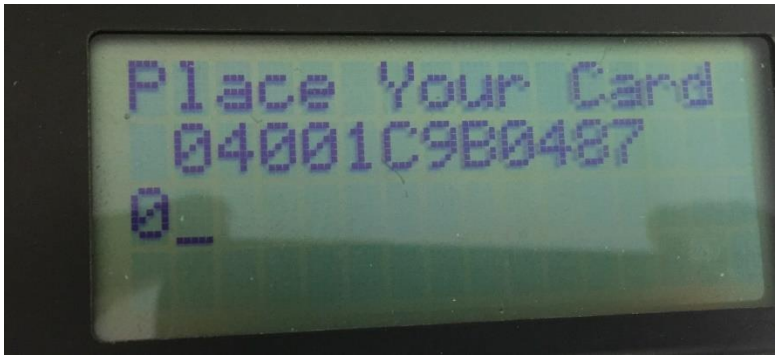


Figure 3.11 Attendance Record of the New user

4 FLOWCHARTS :-

4.1Hardware Flowchart :-

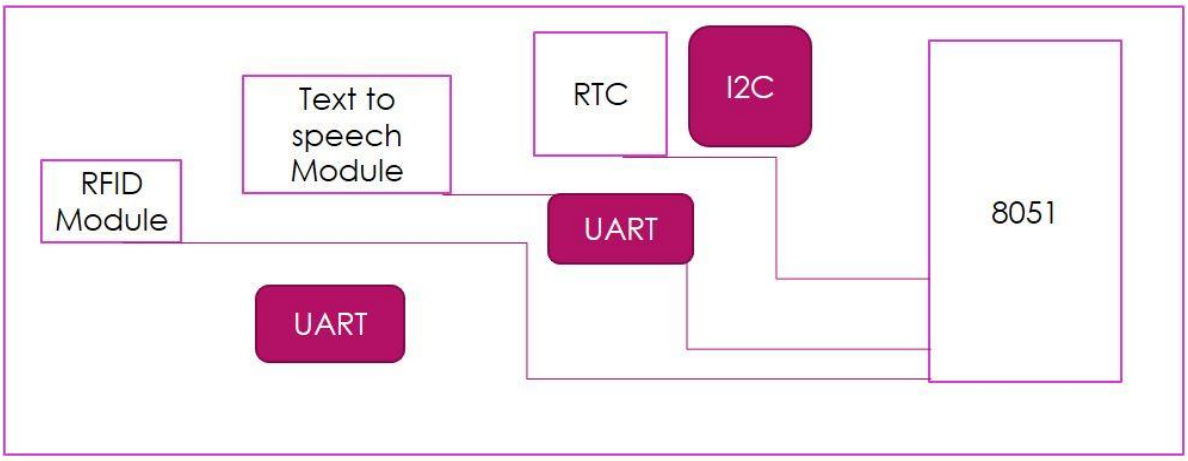


Figure 4.1 Hardware Flowchart

4.2 Software Flowchart :-

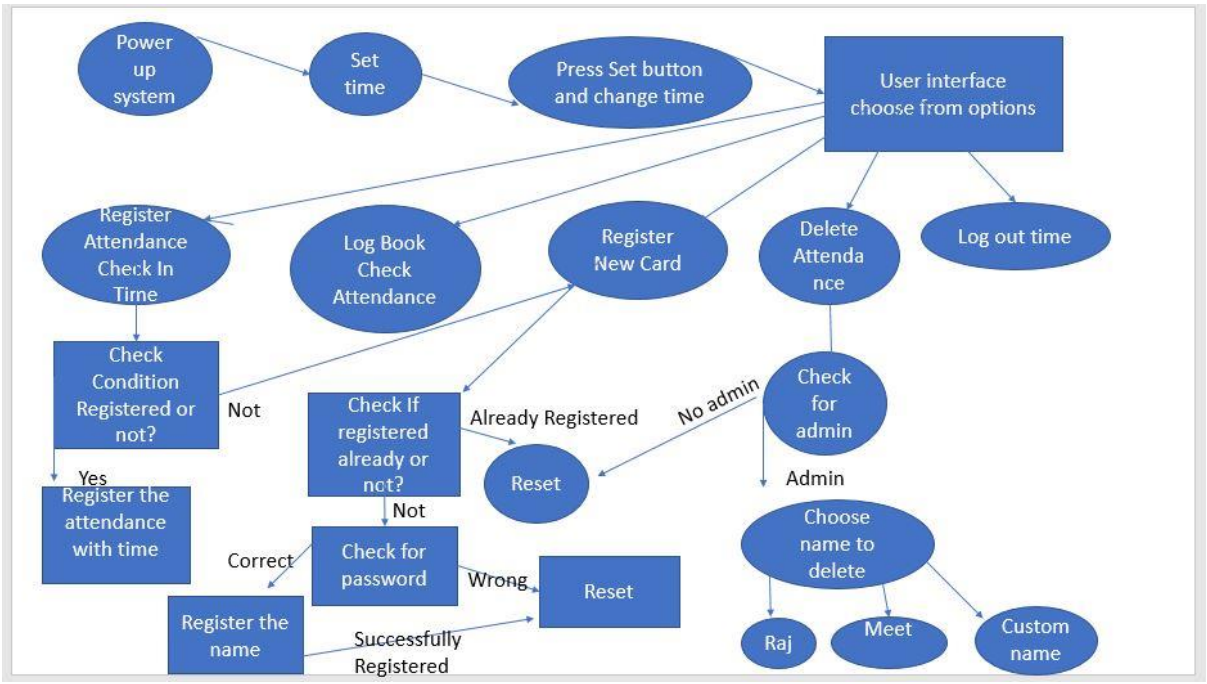


Figure 4.2 Software Flowchart

5 CHALLENGES FACED :-

1) Configuring RFID module and Programming 8051 Development Board at the same time. Initially, I only had one RS-232 port so I was not able to program the 8051 board for my debugging issues and also get the data from the RFID module at the same time. Whenever I tried attaching the RFID module to the MAX 232 of the board while programming the 8051, it either stopped the programming procedure or after programming while getting the values from the RFID module of the RFID card I used to get the wrong 12 digit numbers on the screen.

   To overcome this challenge , I had to attach another MAX 232 with another RS-232 port , so now I am able to attach the RFID output at 1 MAX232 and at the second MAX232 I am able to program the 8051 and also open the UART communication on PUTTY. I am able to get the correct 12 digit numbers of the card emitted by the RFID module now.

2) While configuring the RTC module with the 8051 board, I was not able to run the clock and provide interrupt of the user interface at the same time. Since the RTC was I2C interfaced, I had kept it in a while loop thus blocking my user interface through serial communication to do any procedure.

   To overcome this I have to run the clock in the back and show the time only when I wanted. Thus I cannot show the time continuously running on the front but the real time values come whenever I want it for example tapping the card etc.

6 CONCLUSION :-

Thus, we get a fully functional system which eliminates all the tedious procedures of handling attendance by hand and making the system error free. If more time was there I would have included more modules and more features in the system and tried to make the system more reliable. But in the given time, I learnt interfacing many modules in a system which will help me a lot in future.

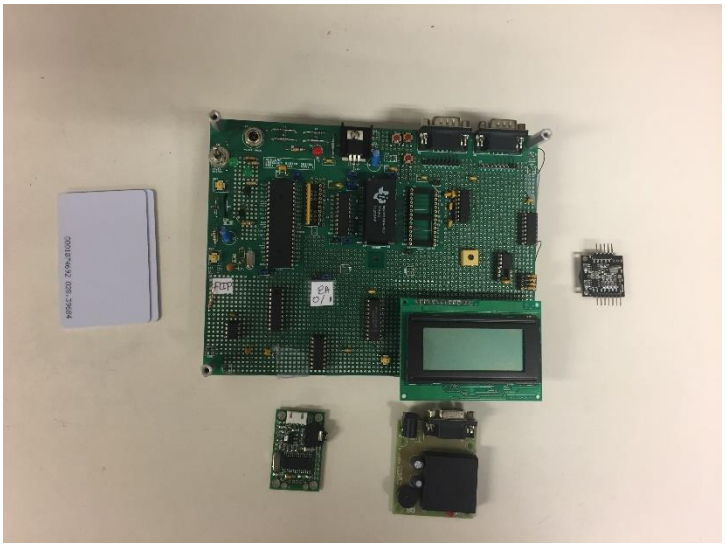


Figure :- 6.1 All modules used in project

7 ACKNOWLEDGEMENTS

1) I would like to thank Professor Linden McClure for such an amazingly designed course structure which guided me through many difficult situations

2) I would like to thank my TAs' Sandeep Raj Kumbargeri and Sanjana Kalyanappagol for helping me throughout the semester

8 REFERENCES :-

1)https://www.elprocus.com/ds1307-rtc-interfacing-with-8051
2)https://www.scribd.com/doc/125666230/RFID-Based-Employee-Attendance-System-Complete-Report
3)https://en.wikipedia.org/wiki/Radio-frequency_identification
4)http://ecee.colorado.edu/~mcclurel
5)https://en.wikipedia.org/wiki/RTC
6)https://en.wikipedia.org/wiki/Speech_synthesis
8) https://www.sunrom.com/p/text-to-speech-tts-module
9) https://ieeexplore.ieee.org/document/5356360/
10) https://ieeexplore.ieee.org/document/7887874/
11) https://ieeexplore.ieee.org/document/5326382/

9 APPENDICES :-

9.1 Appendix - Bill of materials:-

Part Description	Source	Cost
8051 Development Board	Embedded Systems Part Kit	0.00$
RFID Module	Local Store in India	11.1$
RFID tags (10 nos)	Amazon www.amazon.com	5.49$
Male female jumpers	Amazon www.amazon.com	3.4
Female male jumpers	Amazon www.amazon.com	3.45$
Male Male jumpers	Amazon www.amazon.com	3.45$
RFID Module	Amazon www.amazon.com	12.45$
Text to speech Module	Sunrom Electronics www.sunrom.com	18.75$
Max232 IC	Electronics Store	1.00$
Capacitors (5 nos)	Electronics Store	0.50$
Switch (3 nos)	Electronics Store	1.5$
Total		**61.14$**

9.2 Appendix - Schematics

Submitted Separately

9.3 Appendix – Firmware Source Code

Submitted Separately

9.4 Appendix -Data Sheets and Application Notes

Submitted Separately