**Description of Functions Used**

**AT commands**

These commands are used to control GSM/GPRS modem or mobile phone and is used to connect to a web-server using TCP over GPRS. Here is the sequence of AT commands that have been used in our program for establishing connection.

/\*first test if everything is okay\*/

=>AT

/\*this should come back, SIM900 default is to echo back commands you enter\*/

<=AT

/\*to deactivate GPRS pdp(packet data protocol) content

=>AT+CIPSHUT

/\*configure device for single or multi IP connection\*/

=>AT+CIPMUX=value

If value=0 //single IP

Else

If value=1 //multiple IP

=>AT+CGATT=1 /\*perform a GPRS attach

=0 /\*perform a GPRS detach

/\* to set up the APN for PDP context\*/

=>AT+CSTT=\”TATA.DOCOMO.INTERNET\”user”pwd”

/\*bring up wireless connection with GPRS\*/

=>AT+CIICR

/\*check if module being allocated IP address by the network\*/

=>AT+CIFSR

=> AT+SAPBR=1,1 /\*to open a GPRS context\*/

=0,1 /\*close\*/

=2,1 /\*to query a GPRS context\*/

/\*to initialize HTTP service\*/

=>AT+HTTPINIT

=>AT+HTTPACTION=0 /\*get session stopped\*/

=1 /\*post session start \*/

/\*to read the data of HTTPSERVER\*/

=>AT+HTTPREAD

**Header file**

**reg51.h**

It is one of the important header file used while writing c program for interfacing among modules (RFID Reader, GSM/GPRS ) with microcontroller.

It contains definition of sfr and sbit

sfr (Special Function Register)-> It is used to select entire ports(0,1,2,3)

sbit-> It is used to define hardware set at particular port at specified pin no. of microcontroller

Here is the basic code for blinking LED in our c program

#include<reg51.h>

// It defines a hardware(LED) set at port 1 at a pin 0 of microcontroller

sbit led = P1^0;

void main(){

unsigned int i, j;

while(1) {

led = 1; //5v to LED

for(i=0; i<=40000; i++); //this for loop is used to provide delay

led=0;

for(i=0; i<=40000; i++); // ,, ,,

}

}

**Basic commands used by LCD controller**

Register Select (RS): Determines whether a command (RS = 0) is sent to set up the display or actual data (RS = 1) is sent.

Read/Write: RW=0; writes to the LCD. RW = 1; reads from the LCD

We have used two receive function , **receive1()** for storing bytes of RFID Tag ID(12 bytes) and **receive2()** is for displaying the message on LCD after getting response from server.

The RFID Tag ID used in our system consists of 12 characters and the microcontroller has to pick it when ever the reader transmits it. This is done by checking the status of RI flag in the SCON register. The data transmitted to the microcontroller is received by the sbuf register and this happens automatically. Whenever the SBUF receives a complete block of data, RI is made high. So whenever RI goes high , byte inside SBUF is moved into A register and then it is moved into a memory location (160D) is pre-loaded into the R1. Then R1 is incremented, R1 flag is cleared and the entire step is repeated 12 times for receiving all the 12 characters.

void recieve1()

{

unsigned char k;

for(k=0;k<12;k++)

{

while(RI==0);

rdata1[k]=SBUF;

RI=0;

}

}

void recieve2()

{

unsigned char k;

for(k=0;k<30;k++)

{

while(RI==0);

rdata2[k]=SBUF;

RI=0;

}

}

For database connectivity we have used a PHP page to upload entire war file on cloud.

This file contains code to calculate distance between two stops and their corresponding fare. It sends one status code(like 1, 2, 3, 4) at a time according to the success of card validity or invalidity, sufficient balance or not sufficient balance.

**//.............PHP file description.............//**

**JSP Pages**

1. **AddUser.jsp**=> This page was designed for the purpose to provide GUI to new user to get added to avail the services.
2. **AdminLogin.jsp**=>This is used to provide interface for Admin login to make changes in user database or making system changes.
3. **UserLogin.jsp**=> It aims to provide user to manage his/her account or check balance information or other.
4. **Card\_valid.jsp=>** This page was used to check validity of card in both ways for cashless and cash purposes.
5. **Total\_user.jsp=>** This page was designed for computing total count of user added to AICTSL database.