```
#include <reg51f.h>
#include <stdio.h>
#include <intrins.h>

#define  CH0  0x41
#define  CH1  0x42
#define  CH2  0x43
#define  CH3  0x44

#define        high    1
#define        low     0

//BUTTON     PIN  AND SIGNAL PULSE
sbit     SW1          = P0^0;
sbit     SW2          = P0^1;
sbit     SW3          = P0^2;
sbit     PULSE1       = P0^6;
sbit     PULSE2       = P0^7;
// LCD PIN
sbit     RS           = P2^0;
sbit     En           = P2^1;
sbit     D1           = P2^2;
sbit     D2           = P2^3;
sbit     D3           = P2^4;
sbit     D4           = P2^5;
// KEYPAD PIN
sfr    PORTKEY = 0x90;  // P1

sbit     col1    = P1^0;
sbit     col2    = P1^1;
sbit     col3    = P1^2;
sbit     row1    = P1^3;
sbit     row2    = P1^4;
sbit     row3    = P1^5;
sbit     row4    = P1^6;
```

```
// eeprom PIN
sbit      scl          = P3^2;
sbit      sda          = P3^3;
// LED PIN
sbit      LED       = P2^6;
// DS18S20 PIN
sbit  TMDAT     = P0^5;
// BUZZER1 ALARM PIN
sbit   BUZ1     = P3^4;
// BANDPASS1 YELLOW
sbit   YELL      = P3^7;
// BANDPASS2 GREEN
sbit   GREEN       = P3^5;
//LED ALARM RED
sbit   ALARM   = P3^6;



bit                 TXOK;
bit                 WORK_GSM;
bit                 st_temp;

bit                 s1,s2,s3,s7,s8;
char     CHECK_GSM,INDEX_GSM;

unsigned char st_input;


float temperature;


unsigned char BUFTEMP [10];

unsigned char BUFNUM1 [11];

void delay_ms(unsigned int count) ;

#include          "gsm.c"
#include    "ds18s20.c"
#include    "lcd4.c"
```

```
#include    "key.c"
#include    "i2c.c"
#include    "24lc32.c"

/*****************************************************/
void delay_ms(unsigned int count)  // mSec Delay   x2(22.1184MHz)
{
   unsigned int i;
   while (count) {
      i = 230; while (i>0) i--;
      count--;
   }
}
/*****************************************************/


void read_ds18s20 (void)
{
         bit f;
         unsigned int x;
   unsigned char t,i;

         tmstart ();
   x = tmrtemp ();
         t = x;
         f = x & 0x0100;


         for(i=0;i<10;i++) BUFTEMP[i] = 0;                    // Clear Buf

         if (f)
         {
                  temperature = t ;
                  temperature = temperature + 0.5;
         }
         else
         {
                  temperature = t ;
                  temperature = temperature + 0.0;
         }

         sprintf(BUFTEMP,"%4.1fC  ",temperature);
```

```
        printl_xy("TEMP:",1,3);
        printl_xy(BUFTEMP,1,9);
}
/*****************************************************/

/*****************************************************/
void send_sms (void)
{
        LED = 1;
        printf("AT\r");
        read_ds18s20();



   lcd_clear ();
        printl_xy("Send SMS",1,5);


        printf("AT+CMGS=\"%s\"\r",BUFNUM1);
        delay_ms(2000);
        printf("Vehicle Warning\r");
        printf("The door opening\r");
        printf("TEMP = %s ",BUFTEMP);


        delay_ms(2000);
        putchar(0x1A);
        delay_ms(5000);
        printl_xy("Send SMS",1,5);
        printl_xy("Complete!",2,5);
   CHECK_GSM = 3; WORK_GSM = 0;   delay_ms(1000);  // Mode Check RING
        lcd_clear ();
        read_ds18s20();

}
/*****************************************************/
void send_smsA (void)
{
        LED = 1;
        printf("AT\r");
```

```
        lcd_clear ();
        printl_xy("Send SMS",1,5);


        printf("AT+CMGS=\"%s\"\r",BUFNUM1);
        delay_ms(2000);
        printf("Vehicle vibration\r");
        printf("Case1 frequency 3.4-4.4kHz");

        delay_ms(2000);
        putchar(0x1A);
        delay_ms(5000);
        printl_xy("Send SMS",1,5);
        printl_xy("Complete!",2,5);
    CHECK_GSM = 3; WORK_GSM = 0;   delay_ms(1000);  // Mode Check RING
        lcd_clear ();
        read_ds18s20();

}
/*****************************************************/
void send_smsB(void)
{
        LED = 1;
        printf("AT\r");

        lcd_clear ();
        printl_xy("Send SMS",1,5);


        printf("AT+CMGS=\"%s\"\r",BUFNUM1);
        delay_ms(2000);
        printf("Vehicle vibration\r");
        printf("Case1 frequency 12-16kHz");

        delay_ms(2000);
        putchar(0x1A);
        delay_ms(5000);
        printl_xy("Send SMS",1,5);
        printl_xy("Complete!",2,5);
```

```
   CHECK_GSM = 3; WORK_GSM = 0;   delay_ms(1000);  // Mode Check RING
         lcd_clear ();
         read_ds18s20();

}
/*****************************************************/
void send_smsC (void)
{
         LED =1;
         printf("AT\r");

         lcd_clear ();
         printl_xy("Send SMS",1,5);


         printf("AT+CMGS=\"%s\"\r",BUFNUM1);
         delay_ms(2000);
         printf("High temperature alarm");
         printf(" TEMP = %s ",BUFTEMP);

         delay_ms(2000);
         putchar(0x1A);
         delay_ms(5000);
         printl_xy("Send SMS",1,5);
         printl_xy("Complete!",2,5);
   CHECK_GSM = 3; WORK_GSM = 0;   delay_ms(1000);  // Mode Check RING
         lcd_clear ();
         read_ds18s20();

}
/*****************************************************/
void call_user (void)
{
         unsigned char i,c;
         LED = 1;
         c = 0;

   lcd_clear ();
         printl_xy("Call User",1,5);
   printl_xy("ALARM WARNING",2,3);
```

```
loop1:

        c++;
        if(c>1) goto end;

        printf("ATH\r"); delay_ms(1000);
        printf("AT\r");  delay_ms(500);
        printf("AT\r");  delay_ms(500);

        i = 0;
        printf("ATD%s;\r",BUFNUM1);
        delay_ms(1000); CHECK_GSM = 2; WORK_GSM = 0;

        while(WORK_GSM==0)
        {
                if(i>=65) goto end;
                i++;
                delay_ms(500);
        }
        if(i<65) goto loop1;
end:
        printf("ATH\r"); delay_ms(1000);
        printf("ATH\r"); delay_ms(3000);
        lcd_clear ();
}
/*****************************************************/
void check_number_call (void)
{
        char i;
CHECK_GSM = 4; WORK_GSM = 0;  delay_ms(1000);  // Mode Check Number Call
        i = 0;
        LED = 0;
        while(WORK_GSM==0)
        {
                if(i > 15)
                {
                        printf("ATH\r"); delay_ms(1000);
                        printf("ATH\r"); delay_ms(2000);
```

```
                        CHECK_GSM = 3; WORK_GSM = 0;   delay_ms(1000);      //
Mode Check RING
                        return;
                }
                i++;
                delay_ms(1000);
        }
        printf("ATH\r"); delay_ms(1000);
        printf("ATH\r"); delay_ms(2000);

        send_sms ();                                                        //
send sms Function

}
/*****************************************************/
void check_ring (void)
{
        if(WORK_GSM)
        // Check "RING"  bit WORK_GSM = 1
        {
        lcd_clear ();
                printl_xy("GSM RING!!!",1,5);
        check_number_call();                    // Check Number Function
    }
}
/*****************************************************/

void check_temp (void)
{
    BUZ1 = 0;
        read_ds18s20();
    ALARM=0;

        if(temperature>85.0)
        {
                if(st_temp==0)
                {
                        BUZ1 = 1;
                        ALARM=1;
                        send_smsC();
                }
                st_temp = 1;
```

```
        }
        else if(temperature<55.0)
        {
                st_temp = 0;
        }
}
/*****************************************************/
void check_input (void)
{
        BUZ1 = 0;
        ALARM=0;

        if(SW1==0)
        {
                BUZ1 = 1;ALARM=1;
                if(s1==0){ call_user(); send_sms();     s1 = 1; }
        }
        else s1 = 0;
        if(SW2==0)
        {
                BUZ1 = 1;ALARM=1;
                if(s2==0){ call_user(); send_sms();     s2 = 1; }
        }
        else s2 = 0;
        if(SW3==0)
        {
                BUZ1 = 1; ALARM=1;
                if(s3==0){ call_user(); send_sms();     s3 = 1; }
        }
        else s3 = 0;

}
/*****************************************************/
void check_PULSE (void)
{
        BUZ1 = 0;
        YELL = 0;
        GREEN  = 0;
```

```
        if(PULSE1==0)
        {

                if(s7==1)
                {
                        BUZ1=1;
                        YELL  = 1;
                        send_smsA();
                        s7 = 0;

                }
        }
        else s7 =1;
        if(PULSE2==0)
        {

                if(s8==1)
                {
                        BUZ1=1;
            GREEN =1;
                        send_smsB();
                        s8 = 0;
                }

        }
        else s8 = 1;
 }
/*****************************************************/
void read_num_eeprom (void)
{
        unsigned char i;
        for(i=0;i<10;i++) BUFNUM1[i] = eeprom_read(i);
}
/*****************************************************/
void menuset (void)
{
        unsigned char id=0,k,i;
        unsigned int  t=0;
        lcd_clear ();
        printl_xy("SET NUMBER PHONE",1,1);
```

```
printl_xy(">>",2,1);
printl_xy(BUFNUM1,2,3);
while(1)
{
        k = key();

        if(k==0);
        else if(k=='*')
        {
                lcd_clear ();
                read_ds18s20();

                read_num_eeprom();
                return;
        }
        else if(k=='#')
        {
                for(i=0;i<10;i++) eeprom_write (BUFNUM1[i],i);
                read_num_eeprom();
                printl_xy("Save Ok!!      ",2,1);
                delay_ms(1500);
                printl_xy(">>",2,1);
                printl_xy(BUFNUM1,2,3);
        }
        else
        {
                BUFNUM1[id] = k;
                lcd_xy(2,id+3);
                putl(BUFNUM1[id]);
                if(id==9)id=0;
                else id++;
                t = 1;
        }


        if(t==1)
        {
                lcd_xy(2,id+3);
                putl(BUFNUM1[id]);
        }
```

```
                else if(t==40)
                {
                        lcd_xy(2,id+3);
                        putl(' ');
                }
                else if(t==80) t = 0;
                t++;
        }
}
/*****************************************************/
void callready_gsm (void)
{
        char i;
        LED = 0;
        CHECK_GSM = 1; WORK_GSM = 0;   delay_ms(1000);      // Mode Check
Ready
        lcd_clear ();
        printl_xy("CHECK READY",1,4);
        printl_xy("Wait..connection",2,1);
loop1:

        printf("AT+CPIN?\r");
        // AT COMMAND     Check Ready

        for(i=0;i<10;i++)

        {
                LED = 0; delay_ms(500);
                LED = 1; delay_ms(500);
        }
        if(WORK_GSM==1)
        {
                CHECK_GSM = 3; WORK_GSM = 0;   delay_ms(1000);      // Mode
Check RING
                return;
        }
        else goto loop1;
        lcd_clear ();

}
/*****************************************************/
```

```
void serial_init (void)
{
   SCON = 0x50;                    // set RS232
   TMOD = 0x21;                    // t1=serial
   TH1  = 0xfa;                    // set baud-rate 9600   x2(22.1184MHz)

         TR1  = 1;
         TI   = 1;
         TXOK = 1;                 // set TXOK = 1  Variable

         ES = 1;                   // serial interrupt enable
   EA = 1;                         // enable all interrupt



         st_temp = 0;

         s1 = 0;s2 = 0;s3 = 0;s7 = 0;s8 = 0;
         delay_ms(1000);
}

/*****************************************************/
```

```
void main (void)
{
        unsigned char k;

        lcd_init();
        serial_init();                                  // Set serial
Function
        callready_gsm();                                // Check Ready
Function
        lcd_init();
        printl_xy("NETWORK COMPLETE",2,1);
        printl_xy("Connection",1,4);
        delay_ms(3000);
   lcd_init();

        read_ds18s20();


        read_num_eeprom ();

        while(1)
        {
           check_PULSE ();
                check_ring ();                          // check ring
Function
                check_temp ();                          // check temp
Function
                check_input();                          // check_input
Function

                k = key();
                if(k=='#') menuset ();


                LED = ~LED;                             // LED Status

        }

}
```