

به نام خدا

پروژه آزمایشگاه ریز پردازنده

ارتباط گوشی با میکرو از طریق بلوتوث

استاد آزمایشگاه : حمید اصلانی

تهیه و تنظیم : حسین غلامی

شرح پروژه :

پروژه را برای شبیه سازی به دو بخش کلی تقسیم میکنیم ،

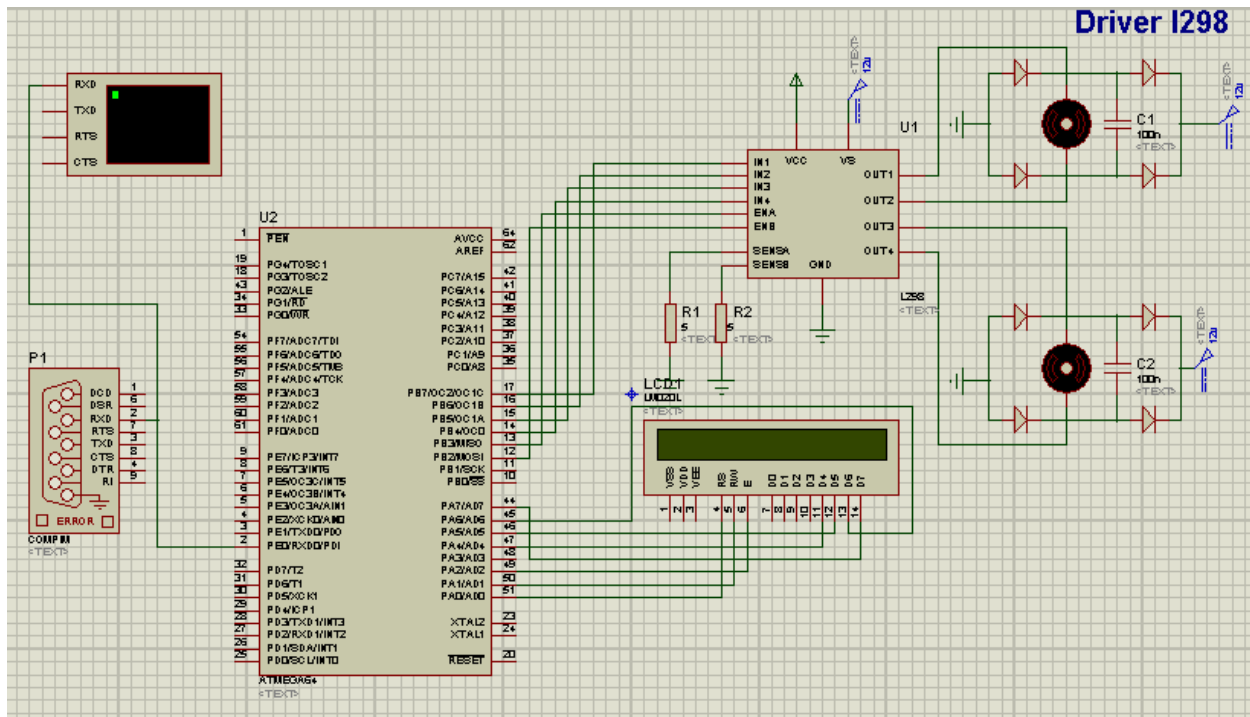
بخش اول

میکرو را از طریق COMPIM و از طریق بلوتوث لپتاپ ، به گوشی متصل نموده و موتور ها را از طریق درایور L298 به میکرو متصل میکنیم

و از طریق رجیستر UDRO و پایه PE0(RXD0) برنامه مربوطه بین Control remote و میکرو را مینویسیم به صورت که یک بافر طراحی کرده و اطلاعات وارد شده از طریق interrupt uart را وارد بافر میکنیم و با توجه به اطلاعات وارد شده برای سیگنال PWM تولید شده توسط Timer0 , Timer1 موتور را با تکنیک تانک هدایت میکنیم. همچنین 2 رجیستر نیز برای کنترل سرعت و زاویه با نام های V ,D آماده شد. که با Vol و channel کنترل میشوند.

همچنین برای دیباگ هنگام طراحی از یک LCD نیز استفاده شد تا اطلاعات منتقل شده از گوشی روی LCD نمایش داده شود

سخت افزار مربوط به بخش اول



کد C مربوط به بخش اول

```
//HOSSEIN__GHOLAMI 9321043
```

```
//LIBRARIS
```

```
#include <mega64.h>
```

```
#include <alcd.h>
```

```
#include <delay.h>
```

```
//DEFENITIONS
```

```
#define ME1 PORTB.3 // EN OF MOTOR 1 (LEFT)
```

```
#define ME2 PORTB.2 //EN OF MOTOR 2 (RIGHT)
```

```
#define SR1 OCR1CL //SPEED (PWM DUTY CYCLE) OF M1 ROTATING FORWARD
```

```
#define SL1 OCR1BL //SPEED (PWM DUTY CYCLE) OF M1 ROTATING BACKWARD
#define SR2 OCR1AL //SPEED (PWM DUTY CYCLE) OF M2 ROTATING FORWARD
#define SL2 OCR0 //SPEED (PWM DUTY CYCLE) OF M2 ROTATING BACKWARD
```

```
int i=1; // USE AS COUNTER
```

```
char data[]={ 'n','n','n','n','n','n','n','n','n'}; // USE AS BUFFER
```

```
unsigned char D = 0x10;
```

```
unsigned char V = 0x50;
```

```
// USART0 Receiver interrupt service routine
```

```
interrupt [USART0_RXC] void usart0_rx_isr(void)
```

```
{ lcd_gotoxy(0,0);
```

```
data[i]=UDR0; //daryaft data
```

```
i++;
```

```
if(data[i]==data[i-1]){ //etminan az nabood data tekrari
```

```
i--;} 
```

```
if(data[1]=='*'){ //etminan az gerftan data jadid
```

```
i=1;} 
```

```
//daryaft kilid 1-9
```

```
switch(data[1]) {
```

```
case '1':
```

```
lcd_putchar('1');
```

```
i=1;
```

```
ME1=0; ME2=1;
```

```
SR1=0x00;SL1=0x00;
```

```
SR2=V+D;SL2=0x00;
```

```
break;

case '2':

    lcd_putchar('2');

    i=1;

    ME1=1; ME2=1;

    SR1=V;SL1=0;

    SR2=V;SL2=0;

break;

case '3':

    lcd_putchar('3');

    i=1;

    ME1=1; ME2=0;

    SR1=V+D;SL1=0x00;

    SR2=0x00;SL2=0x00;

break;

case '4':

    lcd_putchar('4');

    i=1;

    ME1=1; ME2=1;

    SR1=V;SL1=0;

    SR2=0;SL2=V;

break;

case '5':

    lcd_putchar('5');

    i=1;
```

```
break;

case '6':

    lcd_putchar('6');

    i=1;

    ME1=1; ME2=1;

    SR1=0;SL1=V;

    SR2=V;SL2=0;

break;

case '7':

    lcd_putchar('7');

    i=1;

    ME1=0; ME2=1;

    SR1=0x00;SL1=0x00;

    SR2=0x00;SL2=V+D;

break;

case '8':

    lcd_putchar('8');

    i=1;

    ME1=1; ME2=1;

    SR1=0;SL1=V;

    SR2=0;SL2=V;

break;

case '9':

    lcd_putchar('9');

    i=1;
```

```
ME1=1; ME2=0;

SR1=0x00;SL1=V+D;

SR2=0x00;SL2=0x00;

break;

default:

break; }
```

```
switch(data[3]) {                                     //daryaft clid channel

case '+':

    lcd_putchar('D');

    lcd_putchar('+');

    i=1;data[3]='n' ;

    D++;

    if(D==255)D=254;

break;

case '-':

    lcd_putchar('c');

    lcd_putchar('-');

    i=1;data[3]='n';

    D--;

    if(D==0) D=1;

break;

default:

break; }
```

```
switch(data[4]) { // daryaft clid volom
```

```
    case '+':
```

```
        lcd_putchar('v');
```

```
        lcd_putchar('+');
```

```
        i=1;data[4]='\n';data[3]='\n';
```

```
        V++;
```

```
        if(V==255)V=254;
```

```
    break;
```

```
    case '-':
```

```
        lcd_putchar('v');
```

```
        lcd_putchar('-');
```

```
        i=1;data[4]='\n';data[3]='\n';
```

```
        V--;
```

```
        if(V==0)V=1;
```

```
    default:
```

```
        break; }
```

```
    if(data[1]!='*' || data[4]!='+' || data[4]!='-' || data[3]!='+' || data[3]!='-') delay_ms(100); //andaki vaghfe  
    baraye nabood data tekrari Va anjam harekat
```

```
    SR1=0;SR2=0;SL1=0;SL2=0;
```

```
    ME1=0;ME2=0;
```

```
}
```

```
void main(void)
```

```
{
```


DDRA=(0<<DDA7) | (0<<DDA6) | (0<<DDA5) | (0<<DDA4) | (0<<DDA3) | (0<<DDA2) | (0<<DDA1) |
(0<<DDA0);

PORTA=(0<<PORTA7) | (0<<PORTA6) | (0<<PORTA5) | (0<<PORTA4) | (0<<PORTA3) | (0<<PORTA2) |
(0<<PORTA1) | (0<<PORTA0);

DDRB=(1<<DDB7) | (1<<DDB6) | (1<<DDB5) | (1<<DDB4) | (1<<DDB3) | (1<<DDB2) | (1<<DDB1) |
(1<<DDB0);

PORTB=(0<<PORTB7) | (0<<PORTB6) | (0<<PORTB5) | (0<<PORTB4) | (0<<PORTB3) | (0<<PORTB2) |
(0<<PORTB1) | (0<<PORTB0);

DDRC=(0<<DDC7) | (0<<DDC6) | (0<<DDC5) | (0<<DDC4) | (0<<DDC3) | (0<<DDC2) | (0<<DDC1) |
(0<<DDC0);

PORTC=(0<<PORTC7) | (0<<PORTC6) | (0<<PORTC5) | (0<<PORTC4) | (0<<PORTC3) | (0<<PORTC2) |
(0<<PORTC1) | (0<<PORTC0);

DDRD=(0<<DDD7) | (0<<DDD6) | (0<<DDD5) | (0<<DDD4) | (0<<DDD3) | (0<<DDD2) | (0<<DDD1) |
(0<<DDD0);

PORTD=(0<<PORTD7) | (0<<PORTD6) | (0<<PORTD5) | (0<<PORTD4) | (0<<PORTD3) | (0<<PORTD2) |
(0<<PORTD1) | (0<<PORTD0);

DDRE=(0<<DDE7) | (0<<DDE6) | (0<<DDE5) | (0<<DDE4) | (0<<DDE3) | (0<<DDE2) | (0<<DDE1) |
(0<<DDE0);

PORTE=(0<<PORTE7) | (0<<PORTE6) | (0<<PORTE5) | (0<<PORTE4) | (0<<PORTE3) | (0<<PORTE2) |
(0<<PORTE1) | (0<<PORTE0);

DDRF=(0<<DDF7) | (0<<DDF6) | (0<<DDF5) | (0<<DDF4) | (0<<DDF3) | (0<<DDF2) | (0<<DDF1) |
(0<<DDF0);

PORTF=(0<<PORTF7) | (0<<PORTF6) | (0<<PORTF5) | (0<<PORTF4) | (0<<PORTF3) | (0<<PORTF2) |
(0<<PORTF1) | (0<<PORTF0);

DDRG=(0<<DDG4) | (0<<DDG3) | (0<<DDG2) | (0<<DDG1) | (0<<DDG0);

PORTG=(0<<PORTG4) | (0<<PORTG3) | (0<<PORTG2) | (0<<PORTG1) | (0<<PORTG0);

// Timer/Counter 0 initialization

ASSR=0<<AS0;

TCCR0=(1<<WGM00) | (1<<COM01) | (0<<COM00) | (1<<WGM01) | (0<<CS02) | (1<<CS01) | (0<<CS00);

```
TCNT0=0x00;
```

```
OCR0=0x00;
```

```
// Timer/Counter 1 initialization
```

```
TCCR1A=(1<<COM1A1) | (0<<COM1A0) | (1<<COM1B1) | (0<<COM1B0) | (1<<COM1C1) |  
(0<<COM1C0) | (0<<WGM11) | (1<<WGM10);
```

```
TCCR1B=(0<<ICNC1) | (0<<ICES1) | (0<<WGM13) | (1<<WGM12) | (0<<CS12) | (1<<CS11) | (0<<CS10);
```

```
TCNT1H=0x00;
```

```
TCNT1L=0x00;
```

```
ICR1H=0x00;
```

```
ICR1L=0x00;
```

```
OCR1AH=0x00;
```

```
OCR1AL=0x00;
```

```
OCR1BH=0x00;
```

```
OCR1BL=0x00;
```

```
OCR1CH=0x00;
```

```
OCR1CL=0x00;
```

```
// Timer/Counter 2 initialization
```

```
// Clock source: System Clock
```

```
// Clock value: Timer2 Stopped
```

```
// Mode: Normal top=0xFF
```

```
// OC2 output: Disconnected
```

```
TCCR2=(0<<WGM20) | (0<<COM21) | (0<<COM20) | (0<<WGM21) | (0<<CS22) | (0<<CS21) | (0<<CS20);
```

```
TCNT2=0x00;
```

```
OCR2=0x00;
```

```

// Timer/Counter 3 initialization

// Clock source: System Clock

// Clock value: Timer3 Stopped

// Mode: Normal top=0xFFFF

// OC3A output: Disconnected

// OC3B output: Disconnected

// OC3C output: Disconnected

// Noise Canceler: Off

// Input Capture on Falling Edge

// Timer3 Overflow Interrupt: Off

// Input Capture Interrupt: Off

// Compare A Match Interrupt: Off

// Compare B Match Interrupt: Off

// Compare C Match Interrupt: Off

TCCR3A=(0<<COM3A1) | (0<<COM3A0) | (0<<COM3B1) | (0<<COM3B0) | (0<<COM3C1) |
(0<<COM3C0) | (0<<WGM31) | (0<<WGM30);

TCCR3B=(0<<ICNC3) | (0<<ICES3) | (0<<WGM33) | (0<<WGM32) | (0<<CS32) | (0<<CS31) | (0<<CS30);

TCNT3H=0x00;

TCNT3L=0x00;

ICR3H=0x00;

ICR3L=0x00;

OCR3AH=0x00;

OCR3AL=0x00;

OCR3BH=0x00;

OCR3BL=0x00;

```

```
OCR3CH=0x00;
```

```
OCR3CL=0x00;
```

```
// Timer(s)/Counter(s) Interrupt(s) initialization
```

```
TIMSK=(0<<OCIE2) | (0<<TOIE2) | (0<<TICIE1) | (0<<OCIE1A) | (0<<OCIE1B) | (0<<TOIE1) | (0<<OCIE0) |  
(0<<TOIE0);
```

```
ETIMSK=(0<<TICIE3) | (0<<OCIE3A) | (0<<OCIE3B) | (0<<TOIE3) | (0<<OCIE3C) | (0<<OCIE1C);
```

```
// External Interrupt(s) initialization
```

```
// INT0: Off
```

```
// INT1: Off
```

```
// INT2: Off
```

```
// INT3: Off
```

```
// INT4: Off
```

```
// INT5: Off
```

```
// INT6: Off
```

```
// INT7: Off
```

```
EICRA=(0<<ISC31) | (0<<ISC30) | (0<<ISC21) | (0<<ISC20) | (0<<ISC11) | (0<<ISC10) | (0<<ISC01) |  
(0<<ISC00);
```

```
EICRB=(0<<ISC71) | (0<<ISC70) | (0<<ISC61) | (0<<ISC60) | (0<<ISC51) | (0<<ISC50) | (0<<ISC41) |  
(0<<ISC40);
```

```
EIMSK=(0<<INT7) | (0<<INT6) | (0<<INT5) | (0<<INT4) | (0<<INT3) | (0<<INT2) | (0<<INT1) | (0<<INT0);
```

```
// USART0 initialization
```

```
// Communication Parameters: 8 Data, 1 Stop, No Parity
```

```
// USART0 Receiver: On
```

```
// USART0 Transmitter: Off
```

```
// USART0 Mode: Asynchronous
```

```
// USART0 Baud Rate: 9600
```

```
UCSR0A=(0<<RXC0) | (0<<TXC0) | (0<<UDRE0) | (0<<FE0) | (0<<DOR0) | (0<<UPE0) | (0<<U2X0) |  
(0<<MPCM0);
```

```
UCSR0B=(1<<RXCIE0) | (0<<TXCIE0) | (0<<UDRIE0) | (1<<RXEN0) | (0<<TXEN0) | (0<<UCSZ02) |  
(0<<RXB80) | (0<<TXB80);
```

```
UCSR0C=(0<<UMSEL0) | (0<<UPM01) | (0<<UPM00) | (0<<USBS0) | (1<<UCSZ01) | (1<<UCSZ00) |  
(0<<UCPOL0);
```

```
UBRR0H=0x00;
```

```
UBRR0L=0x33;
```

```
// USART1 initialization
```

```
// USART1 disabled
```

```
UCSR1B=(0<<RXCIE1) | (0<<TXCIE1) | (0<<UDRIE1) | (0<<RXEN1) | (0<<TXEN1) | (0<<UCSZ12) |  
(0<<RXB81) | (0<<TXB81);
```

```
// Analog Comparator initialization
```

```
// Analog Comparator: Off
```

```
// The Analog Comparator's positive input is
```

```
// connected to the AIN0 pin
```

```
// The Analog Comparator's negative input is
```

```
// connected to the AIN1 pin
```

```
ACSR=(1<<ACD) | (0<<ACBG) | (0<<ACO) | (0<<ACI) | (0<<ACIE) | (0<<ACIC) | (0<<ACIS1) | (0<<ACIS0);
```

```
SFIOR=(0<<ACME);
```

```
// ADC initialization
```

```
// ADC disabled
```

```
ADCSRA=(0<<ADEN) | (0<<ADSC) | (0<<ADFR) | (0<<ADIF) | (0<<ADIE) | (0<<ADPS2) | (0<<ADPS1) |  
(0<<ADPS0);
```

```
// SPI initialization
```

```
// SPI disabled
```

```
SPCR=(0<<SPIE) | (0<<SPE) | (0<<DORD) | (0<<MSTR) | (0<<CPOL) | (0<<CPHA) | (0<<SPR1) | (0<<SPR0);
```

```
// TWI initialization
```

```
// TWI disabled
```

```
TWCR=(0<<TWEA) | (0<<TWSTA) | (0<<TWSTO) | (0<<TWEN) | (0<<TWIE);
```

```
// Alphanumeric LCD initialization
```

```
// Connections are specified in the
```

```
// Project|Configure|C Compiler|Libraries|Alphanumeric LCD menu:
```

```
// RS - PORTA Bit 0
```

```
// RD - PORTA Bit 1
```

```
// EN - PORTA Bit 2
```

```
// D4 - PORTA Bit 4
```

```
// D5 - PORTA Bit 5
```

```
// D6 - PORTA Bit 6
```

```
// D7 - PORTA Bit 7
```

```
// Characters/line: 8
```

```
lcd_init(20);
```

```
// Global enable interrupts
```

```
#asm("sei")
```

```
while (1){ delay_ms(100);}}
```

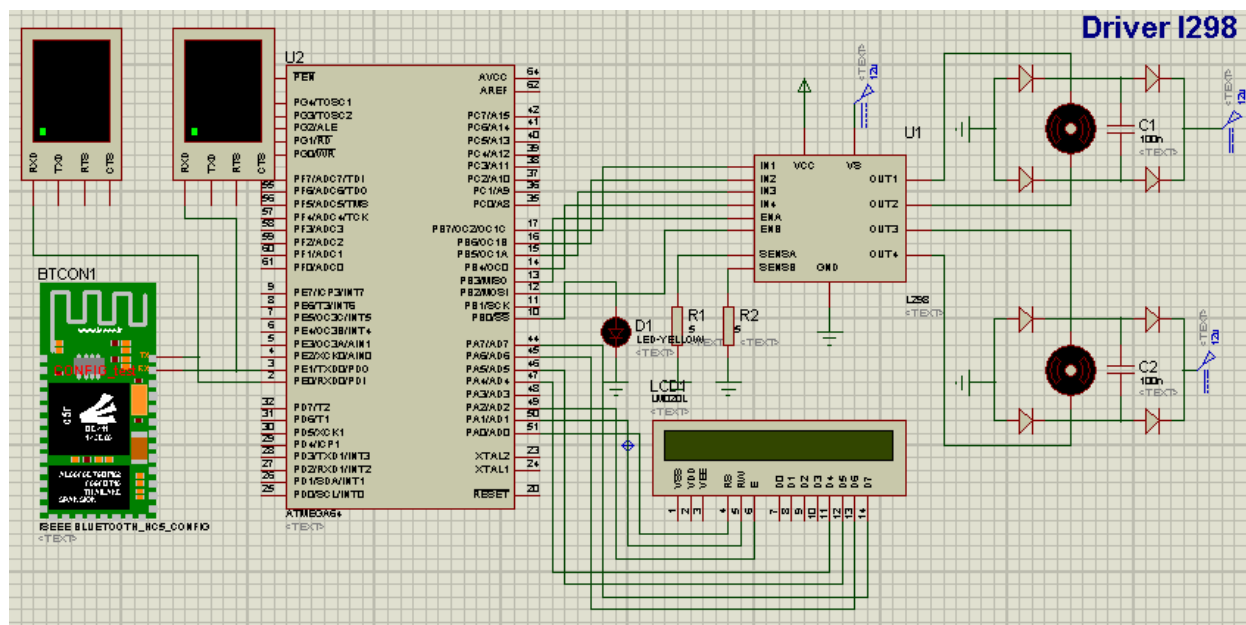
بخش دوم

حال در این بخش بجای استفاده از COMPIM از ماژول بلوتوث طراحی شده توسط شما طراحی را شروع میکنیم

و ماژول را در مد پیکر بندی قرار میدهم اجازه فعال سازی اینترایت ها را (sei) را قطع کرده و هنگامی چنین اجازه ای را میدهم که مطمئن باشیم که ماژول به میکرو متصل است و ماژول راه اندازی شده اطلاعات معرفی خود منتشر کرده است ، چرا که در غیر این صورت بافر طراحی شده توسط ما پر میشود و در بدنه (1)While ، بعد از کمی صبر تا معرفی ماژول تمام شود ، اینترایت ها را فعال کرده و AT را به ماژول میفرستیم.

در جواب ماژول به ما OK اول را میدهد که از طریق رجیستر [تعداد اوکی ها را می‌شماریم بعد از اولین اوکی که از اتصال ماژول به درستی مطمئن شدیم برای آن نام و پسورد انتخاب میکنیم. و پایه CB را صفر کرده و ادامه بخش 2 مانند بخش اول میشود.

سخت افزار بخش دوم



کد C بخش دوم

```
//HOSSEIN___GHOLAMI 9321043

//LIBRARIS

#include <mega64.h>

#include <alcd.h>

#include <delay.h>

#include <stdio.h>

//DEFENITIONS

#define ME1 PORTB.3 // EN OF MOTOR 1 (LEFT)

#define ME2 PORTB.2 //EN OF MOTOR 2 (RIGHT)

#define SR1 OCR1CL //SPEED (PWM DUTY CYCLE) OF M1 ROTATING FORWARD

#define SL1 OCR1BL //SPEED (PWM DUTY CYCLE) OF M1 ROTATING BACKWARD

#define SR2 OCR1AL //SPEED (PWM DUTY CYCLE) OF M2 ROTATING FORWARD

#define SL2 OCR0 //SPEED (PWM DUTY CYCLE) OF M2 ROTATING BACKWARD

#define CB PORTB.0 //Config of bluetooth

int i=1; // USE AS COUNTER

int j=0;

char data[]={'\n','\n','\n','\n','\n','\n','\n','\n'}; // USE AS BUFFER

unsigned char D = 0x10;

unsigned char V = 0x50;

unsigned char C = 0;
```



```
unsigned char asm = 1;
```

```
// USART0 Receiver interrupt service routine
```

```
interrupt [USART0_RXC] void usart0_rx_isr(void)
```

```
{ lcd_gotoxy(0,0);
```

```
    data[i]=UDR0;  //daryaft data
```

```
    i++;
```

```
    if(data[i]==data[i-1]){      //etminan az nabood data tekrari
```

```
        i--;} 
```

```
if (C==0){
```

```
    CB=1;
```

```
    if(data[1]=='O'&&data[2]=='K'){
```

```
        j++;
```

```
        lcd_putchar('O');
```

```
        if(j==1){
```

```
            putchar('A');
```

```
            putchar('T');
```

```
            putchar('+');
```

```
            putchar('N');
```

```
            putchar('A');
```

```
            putchar('M');
```

```
            putchar('E');
```

```
    putchar('=');  
    putchar('H');  
    putchar('G');  
    putchar('H');  
    putchar('\r');  
    putchar('\n');  
}
```

```
if(j==2){  
    putchar('A');  
    putchar('T');  
    putchar('+');  
    putchar('P');  
    putchar('S');  
    putchar('W');  
    putchar('D');  
    putchar('=');  
    putchar('1');  
    putchar('2');  
    putchar('3');  
    putchar('\r');  
    putchar('\n');
```

```
C=1;  
}
```

```
i=1;}
```

```
if(data[1]=='E' || data[1]=='F'){
```

```
    C=0;
```

```
    j=0;
```

```
    lcd_putchar('R');
```

```
    i=1;}
```

```
}
```

```
if (C==1){
```

```
    CB=0;
```

```
    if(data[1]=='*'){
```

```
        i=1;}
```

```
//daryaft kilid 1-9
```

```
switch(data[1]) {
```

```
    case '1':
```

```
        lcd_putchar('1');
```

```
        i=1;
```

```
        ME1=0; ME2=1;
```

```
        SR1=0x00;SL1=0x00;
```

```
        SR2=V+D;SL2=0x00;
```

```
break;

case '2':

    lcd_putchar('2');

    i=1;

    ME1=1; ME2=1;

    SR1=V;SL1=0;

    SR2=V;SL2=0;

break;

case '3':

    lcd_putchar('3');

    i=1;

    ME1=1; ME2=0;

    SR1=V+D;SL1=0x00;

    SR2=0x00;SL2=0x00;

break;

case '4':

    lcd_putchar('4');

    i=1;

    ME1=1; ME2=1;

    SR1=V;SL1=0;

    SR2=0;SL2=V;

break;

case '5':

    lcd_putchar('5');

    i=1;
```

```
break;

case '6':

    lcd_putchar('6');

    i=1;

    ME1=1; ME2=1;

    SR1=0;SL1=V;

    SR2=V;SL2=0;

break;

case '7':

    lcd_putchar('7');

    i=1;

    ME1=0; ME2=1;

    SR1=0x00;SL1=0x00;

    SR2=0x00;SL2=V+D;

break;

case '8':

    lcd_putchar('8');

    i=1;

    ME1=1; ME2=1;

    SR1=0;SL1=V;

    SR2=0;SL2=V;

break;

case '9':

    lcd_putchar('9');

    i=1;
```

```
ME1=1; ME2=0;

SR1=0x00;SL1=V+D;

SR2=0x00;SL2=0x00;

break;

default:

break; }
```

```
switch(data[3]) {                                //daryaft clid channel

case '+':

    lcd_putchar('D');

    lcd_putchar('+');

    i=1;data[3]='n' ;

    D++;

    if(D==255)D=254;

break;

case '-':

    lcd_putchar('c');

    lcd_putchar('-');

    i=1;data[3]='n';

    D--;

    if(D==0) D=1;

break;

default:

break; }
```

```
switch(data[4]) { // daryaft clid volom
```

```
case '+':
```

```
    lcd_putchar('v');
```

```
    lcd_putchar('+');
```

```
    i=1;data[4]='\n';data[3]='\n';
```

```
    V++;
```

```
    if(V==255)V=254;
```

```
break;
```

```
case '-':
```

```
    lcd_putchar('v');
```

```
    lcd_putchar('-');
```

```
    i=1;data[4]='\n';data[3]='\n';
```

```
    V--;
```

```
    if(V==0)V=1;
```

```
default:
```

```
break; }
```

```
if(data[1]!='*' || data[4]!='+' || data[4]!='-' || data[3]!='+' || data[3]!='-') delay_ms(100); //andaki vaghfe  
baraye nabood data tekrari Va anjam harekat
```

```
SR1=0;SR2=0;SL1=0;SL2=0;
```

```
ME1=0;ME2=0;
```

```
}
```

```
}
```

```

void main(void)

{

C=0;

// Port A initialization

DDRA=(0<<DDA7) | (0<<DDA6) | (0<<DDA5) | (0<<DDA4) | (0<<DDA3) | (0<<DDA2) | (0<<DDA1) |
(0<<DDA0);

PORTA=(0<<PORTA7) | (0<<PORTA6) | (0<<PORTA5) | (0<<PORTA4) | (0<<PORTA3) | (0<<PORTA2) |
(0<<PORTA1) | (0<<PORTA0);

// Port B initialization

DDRB=(1<<DDB7) | (1<<DDB6) | (1<<DDB5) | (1<<DDB4) | (0<<DDB3) | (0<<DDB2) | (0<<DDB1) |
(0<<DDB0);

PORTB=(0<<PORTB7) | (0<<PORTB6) | (0<<PORTB5) | (0<<PORTB4) | (0<<PORTB3) | (0<<PORTB2) |
(0<<PORTB1) | (0<<PORTB0);

// Port C initialization

DDRC=(0<<DDC7) | (0<<DDC6) | (0<<DDC5) | (0<<DDC4) | (0<<DDC3) | (0<<DDC2) | (0<<DDC1) |
(0<<DDC0);

PORTC=(0<<PORTC7) | (0<<PORTC6) | (0<<PORTC5) | (0<<PORTC4) | (0<<PORTC3) | (0<<PORTC2) |
(0<<PORTC1) | (0<<PORTC0);

// Port D initialization

DDRD=(0<<DDD7) | (0<<DDD6) | (0<<DDD5) | (0<<DDD4) | (0<<DDD3) | (0<<DDD2) | (0<<DDD1) |
(0<<DDD0);

PORTD=(0<<PORTD7) | (0<<PORTD6) | (0<<PORTD5) | (0<<PORTD4) | (0<<PORTD3) | (0<<PORTD2) |
(0<<PORTD1) | (0<<PORTD0);

// Port E initialization

DDRE=(0<<DDE7) | (0<<DDE6) | (0<<DDE5) | (0<<DDE4) | (0<<DDE3) | (0<<DDE2) | (0<<DDE1) |
(0<<DDE0);

PORTE=(0<<PORTE7) | (0<<PORTE6) | (0<<PORTE5) | (0<<PORTE4) | (0<<PORTE3) | (0<<PORTE2) |
(0<<PORTE1) | (0<<PORTE0);

// Port F initialization

```



```
DDRF=(0<<DDF7) | (0<<DDF6) | (0<<DDF5) | (0<<DDF4) | (0<<DDF3) | (0<<DDF2) | (0<<DDF1) |  
(0<<DDF0);
```

```
PORTF=(0<<PORTF7) | (0<<PORTF6) | (0<<PORTF5) | (0<<PORTF4) | (0<<PORTF3) | (0<<PORTF2) |  
(0<<PORTF1) | (0<<PORTF0);
```

```
// Port G initialization
```

```
DDRG=(0<<DDG4) | (0<<DDG3) | (0<<DDG2) | (0<<DDG1) | (0<<DDG0);
```

```
PORTG=(0<<PORTG4) | (0<<PORTG3) | (0<<PORTG2) | (0<<PORTG1) | (0<<PORTG0);
```

```
// Timer/Counter 0 initialization
```

```
// Clock source: System Clock
```

```
// Clock value: 1000.000 kHz
```

```
// Mode: Fast PWM top=0xFF
```

```
// OC0 output: Non-Inverted PWM
```

```
// Timer Period: 0.256 ms
```

```
// Output Pulse(s):
```

```
// OC0 Period: 0.256 ms Width: 0.080314 ms
```

```
ASSR=0<<AS0;
```

```
TCCR0=(1<<WGM00) | (1<<COM01) | (0<<COM00) | (1<<WGM01) | (0<<CS02) | (1<<CS01) | (0<<CS00);
```

```
TCNT0=0x00;
```

```
OCR0=0x00;
```

```
// Timer/Counter 1 initialization
```

```
// Clock source: System Clock
```

```
// Clock value: 1000.000 kHz
```

```
// Mode: Fast PWM top=0x00FF
```

```
// OC1A output: Non-Inverted PWM
```

```

// OC1B output: Non-Inverted PWM

// OC1C output: Non-Inverted PWM

// Noise Canceler: Off

// Input Capture on Falling Edge

// Timer Period: 0.256 ms

// Output Pulse(s):

// OC1A Period: 0.256 ms Width: 0.080314 ms

// OC1B Period: 0.256 ms Width: 0.080314 ms

// OC1C Period: 0.256 ms Width: 0.080314 ms

// Timer1 Overflow Interrupt: Off

// Input Capture Interrupt: Off

// Compare A Match Interrupt: Off

// Compare B Match Interrupt: Off

// Compare C Match Interrupt: Off

TCCR1A=(1<<COM1A1) | (0<<COM1A0) | (1<<COM1B1) | (0<<COM1B0) | (1<<COM1C1) |
(0<<COM1C0) | (0<<WGM11) | (1<<WGM10);

TCCR1B=(0<<ICNC1) | (0<<ICES1) | (0<<WGM13) | (1<<WGM12) | (0<<CS12) | (1<<CS11) | (0<<CS10);

TCNT1H=0x00;

TCNT1L=0x00;

ICR1H=0x00;

ICR1L=0x00;

OCR1AH=0x00;

OCR1AL=0x00;

OCR1BH=0x00;

OCR1BL=0x00;

OCR1CH=0x00;

```

```
OCR1CL=0x00;
```

```
// Timer/Counter 2 initialization
```

```
// Clock source: System Clock
```

```
// Clock value: Timer2 Stopped
```

```
// Mode: Normal top=0xFF
```

```
// OC2 output: Disconnected
```

```
TCCR2=(0<<WGM20) | (0<<COM21) | (0<<COM20) | (0<<WGM21) | (0<<CS22) | (0<<CS21) | (0<<CS20);
```

```
TCNT2=0x00;
```

```
OCR2=0x00;
```

```
// Timer/Counter 3 initialization
```

```
// Clock source: System Clock
```

```
// Clock value: Timer3 Stopped
```

```
// Mode: Normal top=0xFFFF
```

```
// OC3A output: Disconnected
```

```
// OC3B output: Disconnected
```

```
// OC3C output: Disconnected
```

```
// Noise Canceler: Off
```

```
// Input Capture on Falling Edge
```

```
// Timer3 Overflow Interrupt: Off
```

```
// Input Capture Interrupt: Off
```

```
// Compare A Match Interrupt: Off
```

```
// Compare B Match Interrupt: Off
```

```
// Compare C Match Interrupt: Off
```

```
TCCR3A=(0<<COM3A1) | (0<<COM3A0) | (0<<COM3B1) | (0<<COM3B0) | (0<<COM3C1) |  
(0<<COM3C0) | (0<<WGM31) | (0<<WGM30);
```

```
TCCR3B=(0<<ICNC3) | (0<<ICES3) | (0<<WGM33) | (0<<WGM32) | (0<<CS32) | (0<<CS31) | (0<<CS30);
```

```
TCNT3H=0x00;
```

```
TCNT3L=0x00;
```

```
ICR3H=0x00;
```

```
ICR3L=0x00;
```

```
OCR3AH=0x00;
```

```
OCR3AL=0x00;
```

```
OCR3BH=0x00;
```

```
OCR3BL=0x00;
```

```
OCR3CH=0x00;
```

```
OCR3CL=0x00;
```

```
// Timer(s)/Counter(s) Interrupt(s) initialization
```

```
TIMSK=(0<<OCIE2) | (0<<TOIE2) | (0<<TICIE1) | (0<<OCIE1A) | (0<<OCIE1B) | (0<<TOIE1) | (0<<OCIE0) |  
(0<<TOIE0);
```

```
ETIMSK=(0<<TICIE3) | (0<<OCIE3A) | (0<<OCIE3B) | (0<<TOIE3) | (0<<OCIE3C) | (0<<OCIE1C);
```

```
// External Interrupt(s) initialization
```

```
// INT0: Off
```

```
// INT1: Off
```

```
// INT2: Off
```

```
// INT3: Off
```

```
// INT4: Off
```

```
// INT5: Off
```

```
// INT6: Off
```

```
// INT7: Off
```

```
EICRA=(0<<ISC31) | (0<<ISC30) | (0<<ISC21) | (0<<ISC20) | (0<<ISC11) | (0<<ISC10) | (0<<ISC01) |  
(0<<ISC00);
```

```
EICRB=(0<<ISC71) | (0<<ISC70) | (0<<ISC61) | (0<<ISC60) | (0<<ISC51) | (0<<ISC50) | (0<<ISC41) |  
(0<<ISC40);
```

```
EIMSK=(0<<INT7) | (0<<INT6) | (0<<INT5) | (0<<INT4) | (0<<INT3) | (0<<INT2) | (0<<INT1) | (0<<INT0);
```

```
// USART0 initialization
```

```
// Communication Parameters: 8 Data, 1 Stop, No Parity
```

```
// USART0 Receiver: On
```

```
// USART0 Transmitter: On
```

```
// USART0 Mode: Asynchronous
```

```
// USART0 Baud Rate: 9600
```

```
UCSR0A=(0<<RXC0) | (0<<TXC0) | (0<<UDRE0) | (0<<FE0) | (0<<DOR0) | (0<<UPE0) | (0<<U2X0) |  
(0<<MPCM0);
```

```
UCSR0B=(1<<RXCIE0) | (0<<TXCIE0) | (0<<UDRIE0) | (1<<RXEN0) | (1<<TXEN0) | (0<<UCSZ02) |  
(0<<RXB80) | (0<<TXB80);
```

```
UCSR0C=(0<<UMSEL0) | (0<<UPM01) | (0<<UPM00) | (0<<USBS0) | (1<<UCSZ01) | (1<<UCSZ00) |  
(0<<UCPOL0);
```

```
UBRR0H=0x00;
```

```
UBRR0L=0x33;
```

```
// USART1 initialization
```

```
// USART1 disabled
```

```
UCSR1B=(0<<RXCIE1) | (0<<TXCIE1) | (0<<UDRIE1) | (0<<RXEN1) | (0<<TXEN1) | (0<<UCSZ12) |  
(0<<RXB81) | (0<<TXB81);
```

```

// Analog Comparator initialization

// Analog Comparator: Off

// The Analog Comparator's positive input is
// connected to the AIN0 pin

// The Analog Comparator's negative input is
// connected to the AIN1 pin

ACSR=(1<<ACD) | (0<<ACBG) | (0<<ACO) | (0<<ACI) | (0<<ACIE) | (0<<ACIC) | (0<<ACIS1) | (0<<ACIS0);

SFIOR=(0<<ACME);


// ADC initialization

// ADC disabled

ADCSRA=(0<<ADEN) | (0<<ADSC) | (0<<ADFR) | (0<<ADIF) | (0<<ADIE) | (0<<ADPS2) | (0<<ADPS1) |
(0<<ADPS0);


// SPI initialization

// SPI disabled

SPCR=(0<<SPIE) | (0<<SPE) | (0<<DORD) | (0<<MSTR) | (0<<CPOL) | (0<<CPHA) | (0<<SPR1) | (0<<SPR0);


// TWI initialization

// TWI disabled

TWCR=(0<<TWEA) | (0<<TWSTA) | (0<<TWSTO) | (0<<TWEN) | (0<<TWIE);


// Alphanumeric LCD initialization

// Connections are specified in the

// Project|Configure|C Compiler|Libraries|Alphanumeric LCD menu:

// RS - PORTA Bit 0

```

```
// RD - PORTA Bit 1

// EN - PORTA Bit 2

// D4 - PORTA Bit 4

// D5 - PORTA Bit 5

// D6 - PORTA Bit 6

// D7 - PORTA Bit 7

// Characters/line: 20

lcd_init(20);


// Global enable interrupts


while (1)

{

    delay_ms(100);


    if(C==0){

        if(asm==1){

            delay_ms(500);

            #asm("sei")

            delay_ms(400);

            asm--;}

        putchar('A');
```

```
    putchar('T');
```

```
    putchar('\r');
```

```
    putchar('\n');
```

```
}
```

```
}
```

```
}
```