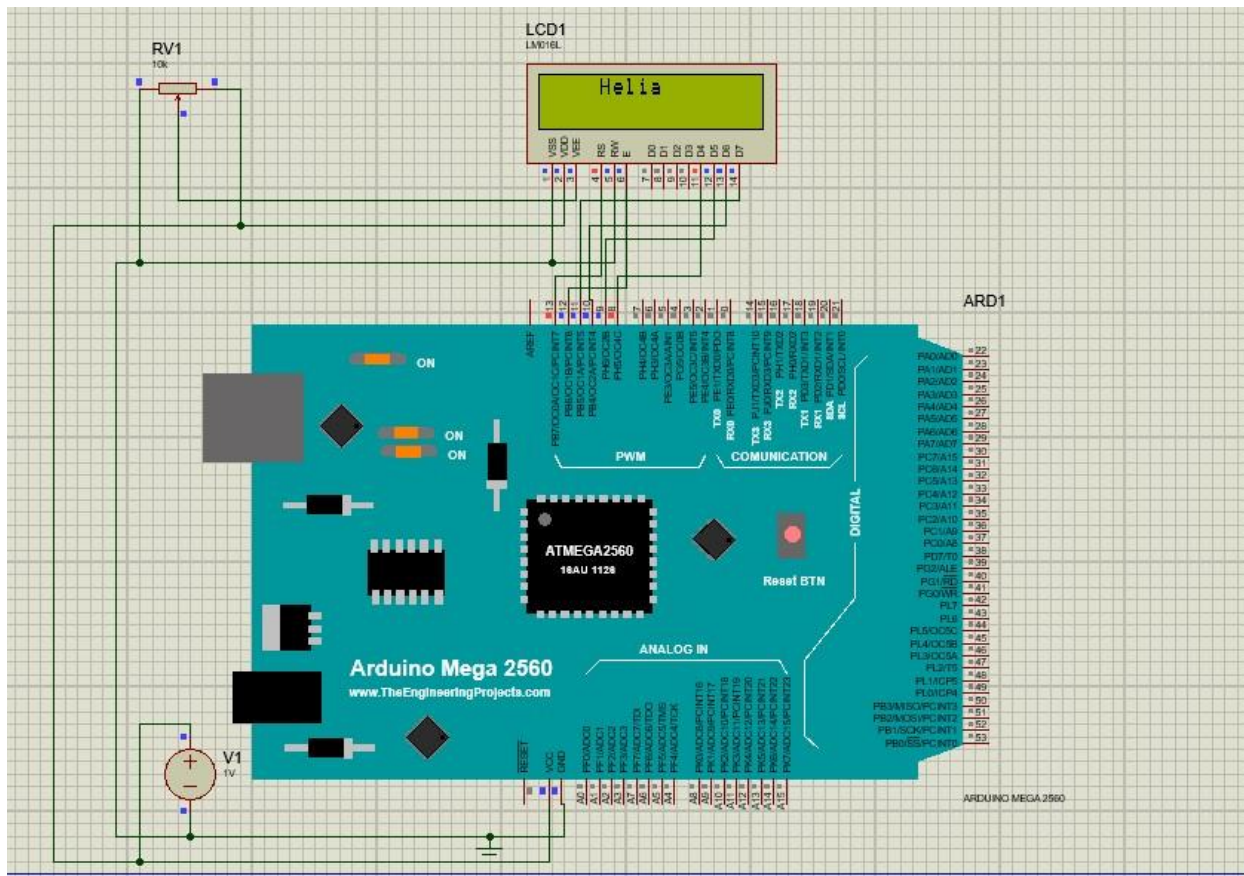


تمرین ها:

برای هر سه مدار ابتدا برد ATmega2560 را انتخاب می کنیم و LCD را به پین های مشخص شده در دستور کار وصل می کنیم و مقاومت و پنتاسیومتر و کپد را به برد خود وصل می کنیم. یک منبع ولتاژ داریم که سر منفی آن را به زمین و سر مثبت آن را به VCC وصل می کنیم.

-1

مدار:



کد:

```
#include <LiquidCrystal.h>
#include <Keypad.h>
```

کتابخانه های مورد نیاز را اضافه می کنیم.

```
#define RS_PIN 13
#define EN_PIN 12
#define D4_PIN 8
#define D5_PIN 9
#define D6_PIN 10
#define D7_PIN 11
```

مشخص کردن پین ها

```
LiquidCrystal lcd(RS_PIN, EN_PIN, D4_PIN, D5_PIN, D6_PIN, D7_PIN);
const String name="Halla";
int cursorColumn=name.length();
int cursorRow=0;
int cnt=0;
```

ایجاد یک متغیر از جنس LiquidCrystal

اسمی که می خواهیم حرکت کند.

```
void setup() {
  // put your setup code here, to run once:
  lcd.begin(16,2);
  lcd.clear();
}
```

begin برای مشخص کردن تعداد سطر و ستون استفاده کرده و یک بار هم صفحه را پاک می کنیم.

برای سریع تر دیدن عملیات اخیر 0.3 ثانیه ای در نظر گرفتیم اما اگر یک ثانیه تاخیر می خواستیم آن را به 1000 تغییر می دهیم.

```
void loop() {
  if(cnt!=12){
    lcd.setCursor(cnt,cursorRow);
    lcd.print(name);
    delay(300);
    lcd.clear();
    cnt++;
  }
```

چک کردن این که رشته از سطر گذشته یا نه یعنی به انتهای سطر رسیده است یا خیر

ابتدا کرسر را به سطر 0 و ستون cnt برده و اسم خود را درون lcd چاپ می کنم سپس بعد از 0.3 ثانیه تاخیر صفحه نمایش را پاک کرده و به ستون یکی اضافه می کنیم.

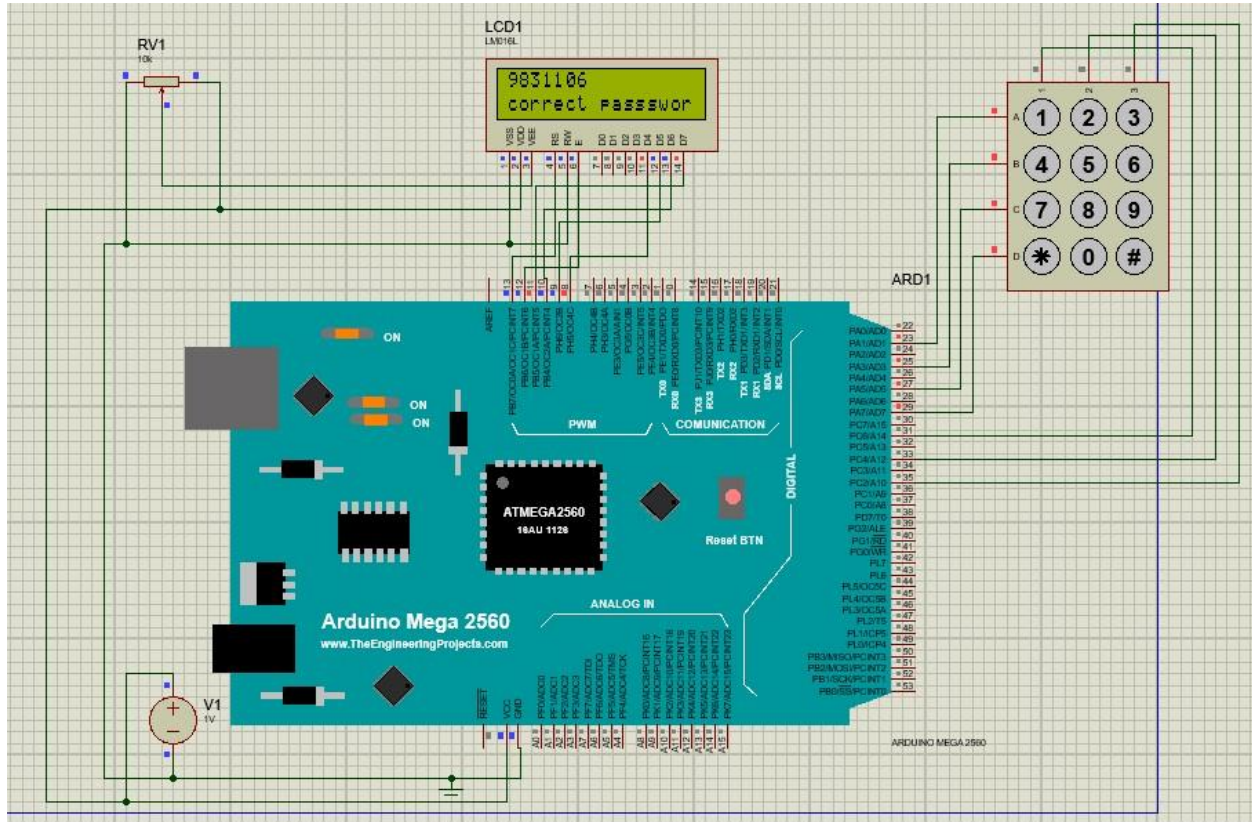
```
else if(cursorRow==0){
  cursorRow=1;
  cnt=0;
}
```

اگر به انتهای سطر برسیم و در سطر نخست باشیم سطر اول را به سطر دوم تغییر داده و مقدار ستون را ریست می کنیم.

```
else if(cursorRow==1){
  cursorRow=0;
  cnt=0;
}
```

اگر به انتهای سطر دوم برسیم سطر دوم را به سطر اول تغییر داده و مقدار ستون را ریست می کنیم.

مدار:



کد:

```
#include <LiquidCrystal.h>
#include <Keypad.h>

#define RS_PIN 13
#define EN_PIN 12
#define D4_PIN 8
#define D5_PIN 9
#define D6_PIN 10
#define D7_PIN 11

LiquidCrystal lcd(RS_PIN, EN_PIN, D4_PIN, D5_PIN, D6_PIN, D7_PIN);
const byte ROWS=4;
const byte COLS=3;
char keys[ROWS][COLS]={
  {'1','2','3'},
  {'4','5','6'},
  {'7','8','9'},
  {'*','0','#'}};
byte col=0;
byte rowPins[ROWS]={23,25,27,29};
byte colPins[COLS]={31,33,35};
Keypad keypad = Keypad(makeKeypad(keys), rowPins, colPins, ROWS, COLS);
String correctPass="9931106";
String pass;

void setup() {
  // put your setup code here, to run once:
  lcd.begin(16,2);
  lcd.clear();
}

void loop() {
  // put your main code here, to run repeatedly:
  char key=keypad.getKey();
  lcd.setCursor(col,0);
  if(key){
    if(key!='*'){
      lcd.print(key);
      col++;
      pass+=key;
    }
    if(key=='*'){
      lcd.setCursor(0,1);
      if(pass==correctPass){
        lcd.println("correct password");
      }
      else{
        lcd.print("wrong password");
        pass="";
      }
    }
  }
}
```

مشخص کردن پین ها

تعداد سطر و ستون

متغیرهای کیبورد را مشخص می کنیم.

پین های کیبورد را مشخص می کنیم.

رمز درست که همان شماره دانشجوییم است.

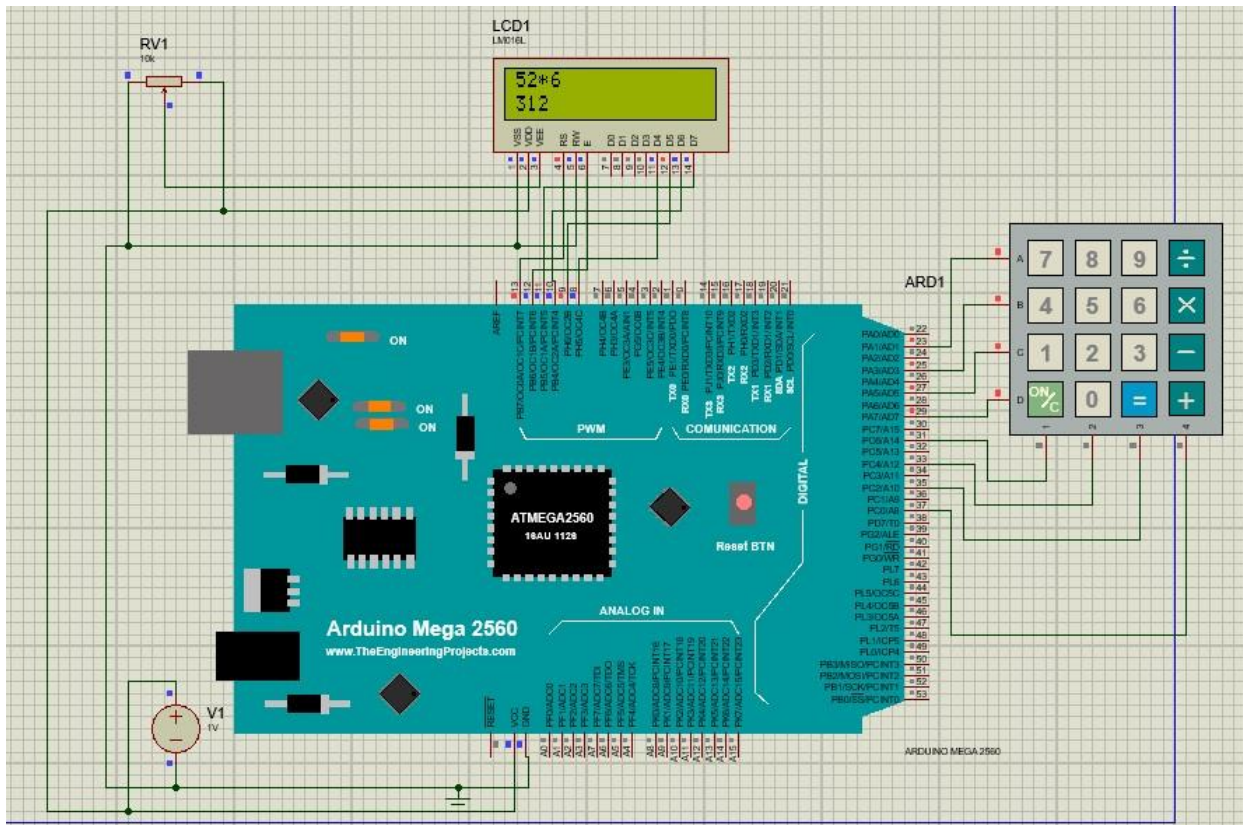
begin برای مشخص کردن تعداد سطر و ستون استفاده کرده و یک بار هم صفحه را پاک می کنیم.

یک آبجکت از کیبورد ساخته و کursor را به ستون مورد نظر و سطر صفر می بریم.

مقدار وارد شده از کیبورد بر روی lcd چاپ شده و به اندازه ی ستون یکی اضافه می شود و کاراکتر فشرده شده به رمز وارد شده اضافه می شود

اگر رمز وارد شده درست باشد با زدن * عبارت correct password چاپ می شود در غیر این صورت عبارت wrong password را در نمایشگر lcd مشاهده می کنیم.

مدار:



کد:

```
#include <LiquidCrystal.h>
#include <Keypad.h>

#define RS_PIN 13
#define EN_PIN 12
#define D4_PIN 8
#define D5_PIN 9
#define D6_PIN 10
#define D7_PIN 11

LiquidCrystal lcd(RS_PIN, EN_PIN, D4_PIN, D5_PIN, D6_PIN, D7_PIN);
const byte ROWS=4;
const byte COLS=4;
char keys[ROWS][COLS]={
  {'7','8','9','/'},
  {'4','5','6','*'},
  {'1','2','3','-'},
  {'0','0','=','+'}};
byte rowPins[ROWS]={23,25,27,29};
byte colPins[COLS]={31,33,35,37};
char str[20];
char str_1[20];
char str_2[20];
char oper;
Keypad keypad = Keypad(makeKeymap(keys), rowPins, colPins, ROWS, COLS);
int op1,op2;

void setup() {
  // put your setup code here, to run once:
  lcd.begin(16,2);
  lcd.clear();
}

void loop() {
  // put your main code here, to run repeatedly:
  char key = keypad.getKey();
  lcd.setCursor(0,0);//FIRST PLACE
  if(strcmp(str,NULL)==0){
    str[0]=key;
  }
  else if(key!='='){
    strncat(str,&key,1);
    lcd.print(str);
    if(key=='/' ||key=='*' ||key=='-' ||key=='+')
    {
      if(oper==NULL){
        oper=key;
        op1=atoi(str_1);
      }
    }
  }
  else{
    if(oper==NULL){
      if(strcmp(str_1,NULL)==0){
        str_1[0]=key;
      }
      else
        strncat(str_1,&key,1);
    }
    else{
      if(strcmp(str_2,NULL)==0){
        str_2[0]=key;
      }
      else
        strncat(str_2,&key,1);
    }
  }
}
```

مشخص کردن پین ها

متغیرهای کپی را مشخص می کنیم.

begin برای مشخص کردن تعداد سطر و ستون استفاده

کرده و یک بار هم صفحه را پاک می کنیم.

اگر کلید مساوی فشرده نشده باشد به این معنی است که هنوز ورودی نگرفته و به کارش ادامه می دهد.

اگر هرکدام از عملگرها فشرده شود آن را در op قرار داده و رشته ی قبلی را به عدد تبدیل کرده و عملگر اول خود قرار می دهیم.

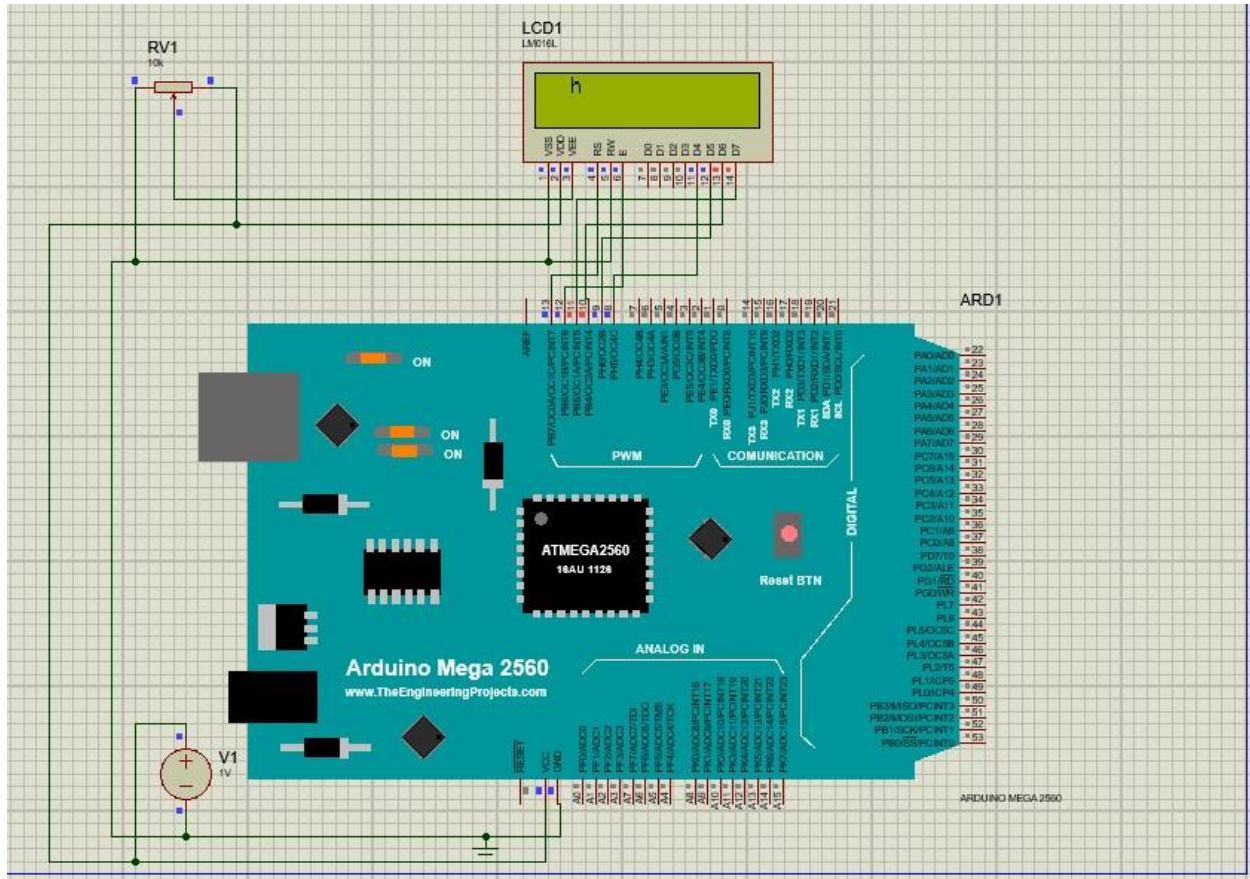
اگر عملگری نداشتیم رشته ی اول را به عدد اولی اضافه کرده و اگر وجود داشت یعنی عدد دوم وارد باید شود

```
else if(key=='='){
    op2=atoi(str_2);
    lcd.setCursor(0, 1);
    switch(oper){
        case('-'):
            lcd.print(op1-op2);
            break;
        case('+'):
            lcd.print(op1+op2);
            break;
        case('*'):
            lcd.print(op1*op2);
            break;
        case('/'):
            lcd.print((float)op1/op2);
            break;
    }
}
```

اگر کلید مساوی فشرده شود عملیات مورد نیاز را انجام داده
و در lcd نمایش داده می شود.

۴-

مدار:



کد:

```
#include <LiquidCrystal.h>
#include <Keypad.h>

#define RS_PIN 13
#define EN_PIN 12
#define D4_PIN 8
#define D5_PIN 9
#define D6_PIN 10
#define D7_PIN 11

LiquidCrystal lcd(RS_PIN, EN_PIN, D4_PIN, D5_PIN, D6_PIN, D7_PIN);
char sh='h';
int cursorCols=0;
int cursorRow=0;
void setup() {
    // put your setup code here, to run once:
    lcd.begin(16,2);
    lcd.clear();
}

void loop() {
    lcd.setCursor(0,0);
    lcd.print(sh);
    cursorCols=0;
    cursorRow=0;
    while (cursorCols!=16) {
        if (cursorCols%2==0) {
            cursorRow=0;
        }
        else{
            cursorRow=1;
        }
        lcd.clear();
        lcd.setCursor(cursorCols, cursorRow);
        lcd.print(sh);
        cursorCols++;
        delay(100);
    }
}
```

مشخص کردن پین ها

کاراکتری که می خواهیم حرکت کند.

begin برای مشخص کردن تعداد سطر و ستون استفاده کرده و یک بار هم صفحه را پاک می کنیم.

ابتدا کursor را در خانه ی 0 و 0 ست کرده و کاراکتری می خواهیم نمایش داده شود را در صفحه نمایش چاپ می کنیم .

شرط اینکه به آخر صفحه رسیده است یا نه

چون باید به صورت زیگزاگی حرکت کنیم اگر سطری که در آن کاراکتر قرار دارد زوج باشد به سطر بعدی رفته و بالعکس

ابتدا نمایشگر را پاک کرده و کursor را براساس ستون و سطر های به دست آمده ست می کنیم و ستون را یک واحد اضافه کرده تا کاراکتر حرکت کند و تاخیر را 100 در نظر گرفته برای راحتی کار.

برای سریع تر دیدن عملیات اخیر 0.1 ثانیه ای در نظر گرفتیم اما اگر یک ثانیه تاخیر می خواستیم آن را به 1000 تغییر می دهیم.

• تعریف مختصر توابع موجود در کتابخانه Liquid Crystal

• تعریف از نوع Liquid Crystal کتابخانه که به دستوراتی که آن RS, RW, enable, Liquid Crystal
• است dp, da

Liquid Crystal (RS, RW, enable, D0, D1, D2, D3, D4, D5, D6, D7)

• begin() : ابتدا به LCD را شروع کرده و به LCD را به عنوان خروجی می‌برد.

• Clear() : صفحه LCD را پاک می‌کند و اصل تصویر قبلی را پاک می‌کند و به حالت اولیه می‌گرداند.
به مختصات (0,0) می‌رود.

• SetCursor() : مکان کursor را تعیین می‌کند که در چه خط و ستونی قرار بگیرد. در اینجا مختصات را می‌توان به صورت (x, y) تعیین کرد.

• Write() : داده‌ای را به LCD می‌نویسد و به مختصات تعیین شده می‌نویسد.
return

• Print() : متن که به عنوان ورودی به آن داده می‌شود را به LCD می‌نویسد و به مختصات تعیین شده می‌نویسد.

• noDisplay() : نمایش LCD را خاموش می‌کند. اگر این دستور را می‌خواهیم خاموش کنیم باید به آن دستور را به آن دستور می‌نویسد.
نمایش داده می‌شود.

• ScrollDisplayLeft : متن را یک واحد به چپ منتقل می‌کند. به عبارت دیگر متن را به چپ منتقل می‌کند.

• auto scroll : به اصل به صورت اتوماتیک scroll می‌کند. در وقت که به دستور می‌نویسد که scroll می‌کند.
به چپ را یک واحد به چپ می‌برد.