

به نام خدا

دانشگاه صنعتی امیرکبیر (پلی تکنیک تهران)

دانشکده مهندسی کامپیوتر



## آزمایشگاه ریزپردازنده

### گزارش کار آزمایش ۷

استاد درس: مهندس معصوم زاده

مهدی رحمانی / ۹۷۳۱۷۰۱

محمد امین رضائی / ۹۷۳۱۰۲۴

در این آزمایش قصد داریم با کمک حافظه EEPROM و برد آردوینو یک ماشین لباس شویی

درست کنیم که دارای ۴ حالت می باشد و با ال ای دی هر حالت مشخص می شود.

ابتدا به پرسش ها پاسخ می دهیم:

**پرسش:** در چه کاربردهایی EEPROM به کار برده می شود؟ چرا در اینجا حافظه Flash یا RAM را به کار نمی بریم؟ تفاوت حافظه RAM با EEPROM چیست؟

در مواقعی که حافظه با حجم کم برای ما کفایت می کند و نیز به ماندگاری بالای اطلاعات نیاز داریم، به جای حافظه ی گران قیمت تری مثل Flash یا Ram از EEPROM استفاده می کنیم.

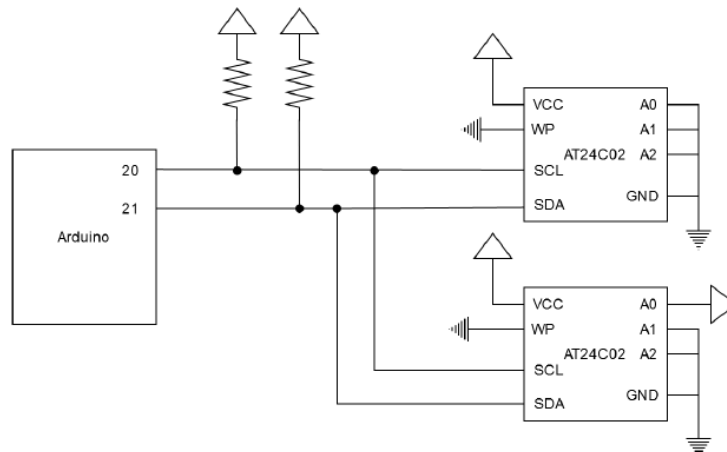
**پرسش:** اگر بخواهیم برای نگهداری مدهای کاری حافظه Flash را به کار ببریم، فرآیند نوشتن باید چگونه انجام شود که داده های دیگری که بر روی همان بلاک هستند از دست نروند؟

باید ابتدا بلاک مورد نظر را بخوانیم و سپس مقادیر مورد نیاز را عوض کنیم و دوباره روی همان بلاک بنویسیم.

**پرسش:** اگر یک حافظه ی EEPROM بیرونی دارای 4KB حافظه و 2 پایه آدرس باشد، در این صورت می توان حداکثر چند KB حافظه EEPROM بیرونی بر روی یک باس مشترک داشت؟

به کمک دو پایه ی A1 و A2 میتوان ۴ دستگاه را آدرس دهی کرد، پس یعنی ۴ ضربدر ۴، یعنی ۱۶ کیلوبایت را روی باس داشت.

**پرسش:** نمودار شماتیک برای این که دو AT24C02 را به یک باس مشترک وصل کنیم و حفاظت نوشتن غیر فعال باشد را رسم کنید (آدرس دهی سخت افزاری دلخواه - باس را هم به پایه های میکروکنترلر متصل کنید)



**پرسش:** فرکانس کلاک در کدام دستگاه پیکربندی می‌شود؟ کلاک را کدام دستگاه فراهم می‌کند؟ با توجه به زمان مورد نیاز برای انجام عملیات نوشتن، با فرض اینکه کلاک را 10KHz تنظیم کرده باشیم، در این صورت حداکثر با چه نرخ می‌توان عملیات نوشتن را انجام داد؟

فرکانس کلاک در بودر آردوینو پیکر بندی میشود و همچنین همین بودر کلاک را تولید میکند.

برای نوشتن هر بایت ما به ۲۹ بیت داده نیاز داریم پس سرعت نرخ نوشتن به صورت زیر میباشد:

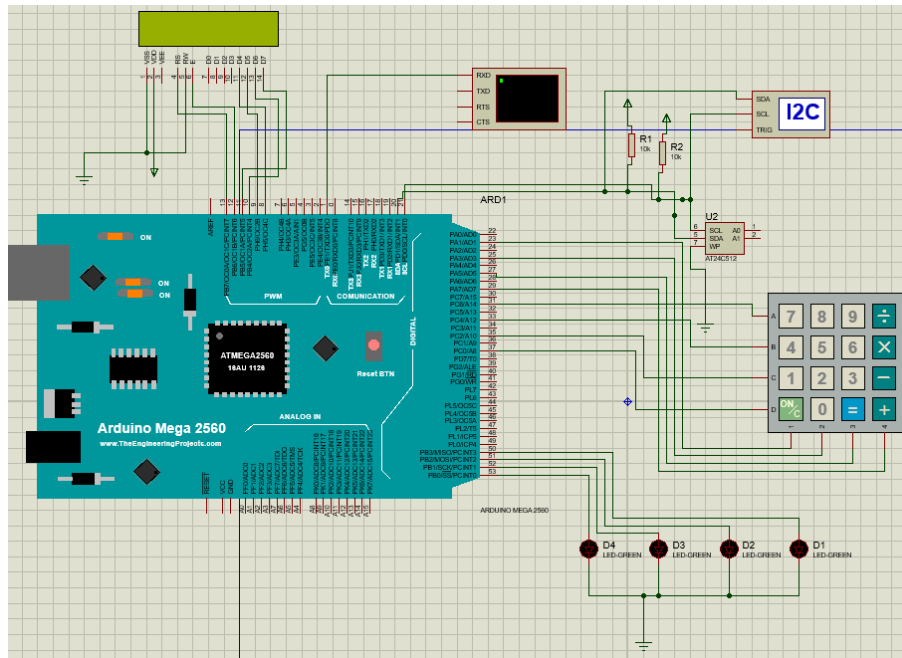
$$10 \times 1000 / 29 = 344 \text{ Byte/s}$$

**پرسش:** هر یک از تابع‌های نوشته‌شده را از راه لینک کتابخانه Wire، در مستندات آردوینو بررسی کنید و کد لازم را برای تولید دنباله‌ی فریم‌ها برای عملیات نوشتن و خواندن گفته‌شده (با این تابع‌ها) بنویسید.

دستور begin برای شروع ارتباط است و دستور set clock برای تنظیم

- `begin()`
- `setClock()` کلاک در صورت نیاز. دستور write برای نوشتن روی حافظه است و
- `beginTransmission()` دستور endTransmission پایان ارتباط را مشخص می‌کند.
- `write()`
- `endTransmission()` requestFrom یک مقدار بولین را دریافت میکند و رفتار I2C
- `requestFrom()`
- `available()` را تعیین میکند. Available یک مقدار بولین برمیگرداند که نشان میدهد
- `read()` Wire مشغول است یا خیر. دستور read نیز با توجه به پروتکل مقدار را می‌خواند.

حال ابتدا مدار را مشاهده میکنیم که دارای قطعاتی مثل برد آردوینو و قطعه AT24C512 و ال سی دی و صفحه کلید و ۴ ال ای دی میباشد:



حال به کد میپردازیم:

```
#include <Wire.h>
#include <LiquidCrystal.h>
#include <Keypad.h>
#define DEVICE_ADDRESS 0b1010000

const int rs = 13, en = 12, d4 = 8, d5 = 9, d6 = 10, d7 = 11;
LiquidCrystal lcd(rs, en, d4, d5, d6, d7);
const byte numRows= 4; //number of rows on the keypad
const byte numCols= 4; //number of columns on the keypad
char keymap[numRows][numCols]=
{
  {'0', '0', '=', '+'},
  {'1', '2', '3', '-'},
  {'4', '5', '6', '*'},
  {'7', '8', '9', '/'},
};

//Code that shows the the keypad connections to the arduino terminals
byte rowPins[numRows] = {37,35,33,31}; //Rows 0 to 3
byte colPins[numCols]= {23,25,27,29}; //Columns 0 to 3
char key;
long int Number = 0 ;
int current_state = 0;
Keypad myKeypad= Keypad(makeKeymap(keymap), rowPins, colPins, numRows, numCols);
```

در این بخش ابتدا یک دیوایس آدرس برای حافظه مان تعیین میکنیم و پورت های ال سی دی را تعیین میکنیم، سپس صفحهکلید را تعریف کرده و بعد پورت های آن را مشخص میکنیم و در آخر

صفحه کلید را میسازیم.

```
void setup() {  
    Wire.begin();  
    Serial.begin(9600);  
    Wire.beginTransmission(DEVICE_ADDRESS);  
    Wire.write(0x00);  
    Wire.write(0xAA);  
    Wire.write(current_state);  
    Wire.endTransmission();  
    pinMode(50,OUTPUT);  
    pinMode(51,OUTPUT);  
    pinMode(52,OUTPUT);  
    pinMode(53,OUTPUT);  
}  
int times[] = {0,0,0,0};  
int n_time_setted = 0;  
int stopped = 1;
```

سپس در بخش ست آپ، پین های ال ای دی ها را مشخص کرده و در حافظه مان وضعیت فعلی یعنی صفر را ذخیره میکنیم. آرایه times شامل زمان های تعیین شده برای ۴ حالت میباشد.

```
void loop() {  
    key = myKeypad.getKey();  
    if (key!=NO_KEY){  
        DetectButtons();  
        lcd.clear();  
        lcd.print(Number);  
    }  
    if (stopped == 0){  
        for(int i=0;i < current_state;i++){  
            digitalWrite(50+i,HIGH);  
        }  
        if (n_time_setted > 4 & current_state <= 3){  
            for (int i=current_state ;i<4;i++){  
                {  
                    Serial.println(current_state);  
                    Serial.println(i);  
  
                    digitalWrite(50+i,HIGH);  
                    for(int j =0;j<times[i];j++){  
                        lcd.print(times[i]-j);  
                        delay(1000);  
                        lcd.clear();  
                    }  
                    Wire.beginTransmission(DEVICE_ADDRESS);  
                    Wire.write(0x00);  
                    Wire.write(0xAA);  
                    Wire.write(i);  
                    Wire.endTransmission();  
                }  
                current_state++;  
            }  
        }  
    }  
}
```

در بخش loop ، کاراکتر ورودی را گرفته و به تابع DetectButtons میدهیم. سپس در شرط بعدی کار روشن کردن همه ال ای دی ها انجام میشود و در شرط بعدی روشن کردن ال ای دی ها و نوشتن حالت فعلی انجام میشود. سپس در حافظه مقدار مورد نظر ذخیره میشود و استیت یکی به جلو میرود.

```
void DetectButtons()
{
    if (key=='0')
    {Serial.println ("Start"); Number = 0;}

    if (key == '1') //If Button 1 is pressed
    {Serial.println ("Button 1");
    if (Number==0)
    Number=1;
    else
    Number = (Number*10) + 1; //Pressed twice
    }

    if (key == '4') //If Button 4 is pressed
    {Serial.println ("Button 4");
    if (Number==0)
    Number=4;
    else
    Number = (Number*10) + 4; //Pressed twice
    }
```

حال به بررسی تابع DetectButtons میپردازیم. ابتدا این تابع کلید را چک میکند و مقدار زمان ورودی کاربر را برای هر مرحله ذخیره میکند.

```

if (key == '=') {
  if(n_time_setted < 3){
    Serial.println ("Number is:");
    Serial.println(Number);
    times[n_time_setted++] = Number;
    Number = 0;
  }
  else if (n_time_setted ==3){
    Number = 0;
    Serial.println ("Times Setted");
    Serial.println ("Enter #OP");
    lcd.print("Enter #OP");
    n_time_setted++;
  }
  else if(n_time_setted ==4) {
    current_state = Number-1;
    Wire.beginTransaction(DEVICE_ADDRESS);
    Wire.write(0x00);
    Wire.write(0xAA);
    Wire.write(current_state);
    Wire.endTransmission();
    Serial.println ("Current state Setted");
    Serial.println (current_state+1);
    n_time_setted++;
    Number = 0;
  }
  else{
    Serial.println(current_state+1);
  }
}

```

سپس در کد بالا ورودی مساوی چک میشود، اگر هنوز کاربر زمان میداد، رمان حالت متاظرش تعیین میشود، اگر زکان دادن تمام شده بود، میگوییم که کاربر حالت شروع را انتخاب کند. سپس استیت اولیه را از کاربر گرفته و در حافظه ذخیره میکنیم.

```

    if (key == '*')
    {
        Serial.println ("Restoring Last State");
        Wire.beginTransaction (DEVICE_ADDRESS);
        Wire.write (0x00);
        Wire.write (0xAA);
        Wire.endTransmission ();
        Wire.requestFrom (DEVICE_ADDRESS,1);
        current_state = Wire.read()+1;
        Serial.println ("Last State is");
        Serial.println (current_state);
    }

    if (key == '-')
    {
        current_state = 0;
        stopped =1;
        for (int i=0;i < 4;i++){
            digitalWrite (50+i,LOW);
        }
        Wire.beginTransaction (DEVICE_ADDRESS);
        Wire.write (0x00);
        Wire.write (0xAA);
        Wire.write (current_state);
        Wire.endTransmission ();
        lcd.print ("Reseted");
        n_time_setted =3;
        Serial.println ("Current State IS:");
        Serial.println (current_state+1);
        delay (1000);
        lcd.clear ();
    }
}

```

در کد بالا، ابتدا ورودی ضرب چک میشود که آخرین حالت را از حافظه گرفته و نمایش میدهد. سپس ورودی منها چیک شده که ال ای دی ها را خاموش کرده و حالت اولیه را یک میکند و برنامه را ریست کرده و پیغام Reseted را چاپ میکند.

```

        if (key == '/')
        {
            stopped = 1;
            lcd.clear ();
            lcd.print ("STOPPED");
            delay (1000);
            lcd.clear ();
        }

        if (key == 'O')
        {
            stopped = 0;
            lcd.clear ();
            lcd.print ("Started");
            delay (1000);
            lcd.clear ();
        }
}

```

در بخش بالا نیز دو ورودی تقسیم برای هولد کردن برنامه و کلید استارت برای شروع برنامه در نظر گرفته شده است.

**یاعلی**