



# ماشین لباسشویی

## • هدف آزمایش

ساخت ماشین لباسشویی حافظه دار

## • وسایل مورد نیاز

### • برد آردوینو مگا

اعمال نوشتن و خواندن از حافظه، ارتباط با کیپد و LCD و روشن و خاموش کردن LED ها بر عهده آن است.

### • LED

به تعداد گام های هر مود یک LED خواهیم داشت که وضعیت فعلی هر مود را نشان میدهد.

### • حافظه

اطلاعات را بایت به بایت با میکرو مبادله میکند.

### • LCD

برای نمایش وضعیت فعلی در از آن استفاده میکنیم. میتوان منو، گزینه های موجود برای انتخاب مود ها، گام فعلی و زمان باقی مانده از هر گام و در پایان تعداد مود های در دسترس را روی آن نشان داد.

### • Keypad

کنترل این ماشین لباسشویی توسط کیپد انجام میشود. در بخش اول میتوان به کمک جدول زیر با دستگاه ارتباط برقرار کرد:

عمل	کلید
Add mode	+
Use default mode	-
Reset	ON/C(n)
Select mode	/
Get info	x

بعلاوه میتوان مود های جدید و تایم هم گام آن را نیز به کمک کیپد وارد کرد.

## • توابع جدید

### begin

ایجاد آبجکت جدید

• بدون آرگومان : Master

• با آرگومان آدرس: Slave

### setClock

فرکانس کلاک را برای ارتباط I2C میتوان ست کرد.

### beginTransmission

انتقال اطلاعات را با آدرس slave ای که در ورودی به آن داده ایم ر شروع میکند.

### Write

دیتا را از slave در پاسخ به درخواست master میفرستد. یا اینکه بایت ها را برای انتقال از master به slave در صف میگذارد.



# ماشین لباسشویی

- endTransmission** انتقال را که پس از دستور beginTransmission آغاز شده بود پایان میدهد و تمام byte ها ی که توسط دستور write در صف قرار گرفته بودند، ارسال میکند.
- requestFrom** master برای درخواست دیتا و بایت ها از slave این دستور را صدا میکند و بعد از آن میتواند دیتا را با read () یا available() بدست آورد. در ورودی آدرس device و سائز و تعداد بایت هارا میدهیم.
- available** Master بعد از صدا کردن دستور requestFrom با استفاده از دستور available میتواند تعداد بایت های آماده برای خواندن را بگیرد
- read** یک بایت را که بر روی باس نوشته شده میخواند و برمیگرداند.

## • محتوای حافظه

	...
98	گام ذخیره شده
99	تعداد مود ها فعلی
100	مود اول (default)
101	
102	
103	
104	مود دوم
105	
106	
107	
	...

## • سوالات تئوری

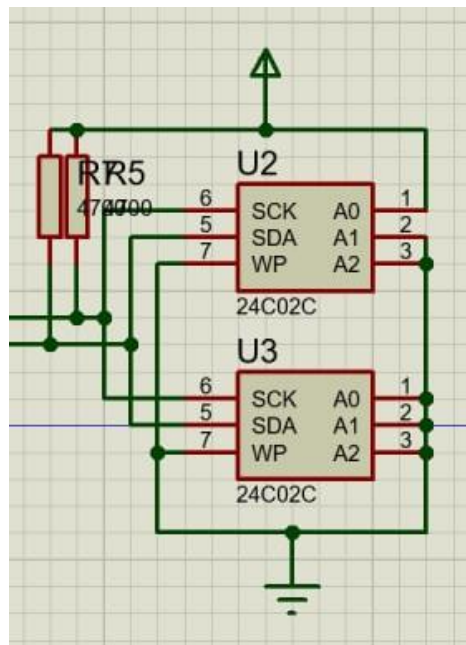
- در چه کاربرد های EEPROM به کار برده میشود؟ چرا در اینجا حافظه Flash یا RAM را به کار نمی‌بریم؟ تفاوت حافظه RAM با EEPROM در چه است ؟  
در مواقعی که به یک حافظه‌ی ثابت نیاز داریم که با قطع برق اطلاعاتش از بین نرود از EEPROM استفاده می‌کنیم حافظه‌ی RAM یک حافظه‌ی volatile است که یعنی با قطع برق اطلاعاتش از بین میرود و در اینجا استفاده نمیشود. حافظه‌ی flash فرار نیست ولی برای نوشتن در آن محدودیت داریم. از تفاوت RAM و EEPROM میتوان به فرار بودن RAM و دائمی بودن EEPROM اشاره کرد علاوه بر این، RAM از EEPROM سریع تر است.
- 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 آن بلوک را بخوانیم، آن خانه های از بلوک که می‌خواهیم تغییر را دهیم و در نهایت کل بلوک را بنویسیم.
- اگر یک حافظه ی EEPROM بیرونی ی دارای 4KB حافظه و 2 پایه آدرس باشد، در این صورت میتوان حداکثر چند KB حافظه EEPROM بیرونی بر روی یک باس مشربک داشت ؟



# ماشین لباسشویی

۲ پایه برای آدرس دهی داریم پس درکل ۴ آدرس میتوانیم داشته باشیم. به ازای هر آدرس 4 KB فضا داریم پس درکل 16kb فضا میتوانیم داشته باشیم.

- نمودار شماتیک برای اینکه دو AT24C02 را به یک باس مشترک وصل کرد و حفاظت نوشتن غیر فعال باشد را رسم کنید



- 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 10 KHz تنظیم کرده باشیم، در این صورت حداکثر با چه نرخ میتوان عملیات نوشتن را انجام داد؟  
در master پیکربندی شده و master آن را فراهم میکند. اگر از نوشتن متوالی استفاده نکنیم و بایت به بایت بنویسیم به ازای هر ۸ بیت نوشتن داده ۲۹ کلاک لازم است پس نرخ انتقال داده برابر است با:

$$8 \times 10k \div 29 = 2.76kb/s$$

- همخوانی این دنباله فریم ها را با پروتکل TWI بررسی کنید فریم های آدرس و داده را مشخص کنید، دستور خواندن یا نوشتن چگونه مشخص میشوند؟





# ماشین لباسشویی

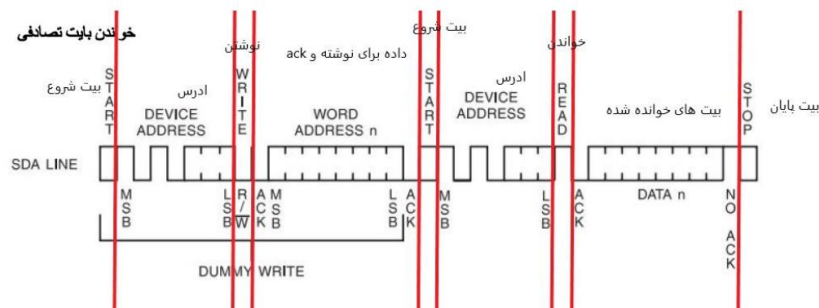
نام و نام خانوادگی: امیرحسین علی بخشی

شماره دانشجویی: ۹۷۳۱۰۹۶

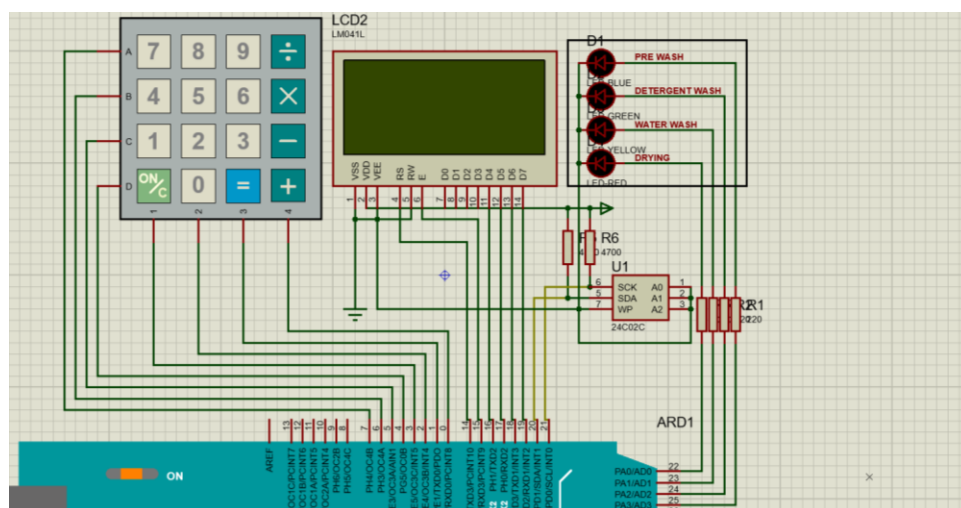
استاد: آقای معصومزاده

درس: آزمایشگاه ریزپردازنده و زبان اسمبلی

اما برای خواندن ابتدا یک نوشتن برای نوشتن آدرسی که میخواهیم بخوانیم در بافر حافظه صورت میگیرد سپس به صورت ۱۲C اعادی میخوانیم:



• مدار



• کد (کامل)

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

```
#define RS 14
#define EN 15
#define D4 16
#define D5 17
#define D6 18
#define D7 19
#define LED1 25
```



# ماشین لباسشویی

نام و نام خانوادگی: امیرحسین علی بخشی

شماره ی دانشجویی: ۹۷۳۱۰۹۶

استاد: آقای معصوم زاده

درس: آزمایشگاه ریزپردازنده و زبان اسمبلی

```
#define LED2 24
#define LED3 23
#define LED4 22
#define DELAY_PERIOD 500
#define MEMORY_ADDRESS 0b1010000

const byte ROWS = 4; // Four rows
const byte COLS = 4; //Four columns

// Define the Keymap
char keys[ROWS][COLS] = {
  {'7','8','9','/'},
  {'4','5','6','x'},
  {'1','2','3','-'},
  {'n','0','=','+'}
};

byte rowPins[ROWS] = { 7, 6, 5, 4 };// Connect keypad ROW0, ROW1, ROW2 and ROW3 to these Arduino pins.
byte colPins[COLS] = { 3, 2, 1, 0 }; // Connect keypad COL0, COL1 and COL2 to these Arduino pins.

Keypad keypad = Keypad( makeKeymap(keys), rowPins, colPins, ROWS, COLS ); // Create the Keypad

LiquidCrystal lcd(RS, EN, D4, D5, D6, D7);

String step_names[5]={"PREWASH","DETERGENT WASH","WATER WASH","DRYING","FINISH"};
int mode;
int LED[4]={LED1, LED2, LED3, LED4};

void turn_off_LEDs(){
  for (int i = 0 ; i < 4 ; i++){
    digitalWrite(LED[i],LOW);
  }
}

// write "Size" bytes on the memory from "mem_address"
void memory_write(byte mem_address, byte* data, byte Size){
  for (int i = 0 ; i < Size ; i++){
    Wire.beginTransmission(MEMORY_ADDRESS);
    Wire.write(mem_address + i);
    Wire.write(data[i]);
    Wire.endTransmission();
  }
}
```



```
        delay(200);
    }
}

// Read "Size" bytes from the memory from "mem_address"
void memory_read(byte mem_address, byte* data, byte Size){
    for (int i = 0 ; i < Size ; i++){
        Wire.beginTransmission(MEMORY_ADDRESS);
        Wire.write(mem_address + i);
        Wire.endTransmission();
        Wire.requestFrom(MEMORY_ADDRESS, Size);
        data[i] = Wire.read();
    }
}

byte read_byte(byte address){
    Wire.beginTransmission(MEMORY_ADDRESS);
    Wire.write(address);
    Wire.endTransmission();
    Wire.requestFrom(MEMORY_ADDRESS,1);
    byte num = 0;
    num = Wire.read();
    return num;
}

void write_byte(byte address,byte num){
    Wire.beginTransmission(MEMORY_ADDRESS);
    Wire.write(address);
    Wire.write(num);
    Wire.endTransmission();
    delay(200);
}

void enter_mode_settings(byte *data){
    byte index=0;
    while(true){
        lcd.clear();
        lcd.setCursor(0,0);
        lcd.print("Enter Settings:");
        lcd.setCursor(0,1);
        switch (index){
            case 0:
                lcd.print("PreWash:");
                break;
            case 1:
```



```
        lcd.print("DetergentW.");
        break;
    case 2:
        lcd.print("WaterW.");
        break;
    case 3:
        lcd.print("Drying.");
        break;
    case 4:
        lcd.print("Press +");
        break;
    }
    lcd.setCursor(0,2);
    char key;
    key = keypad.waitForKey();
    lcd.print(key);
    lcd.setCursor(0,3);
    lcd.print("...");
    delay(1000);
    data[index]= key - 48;
    index++;
    if (index != 4){
        continue;
    }else{
        lcd.clear();
        lcd.setCursor(0,0);
        lcd.print("PreWash.");
        lcd.print(data[0]);
        lcd.setCursor(0,1);
        lcd.print("DetergentW.");
        lcd.print(data[1]);
        lcd.setCursor(0,2);
        lcd.print("WaterW.");
        lcd.print(data[2]);
        lcd.setCursor(0,3);
        lcd.print("Drying.");
        lcd.print(data[3]);
        char tmp = keypad.waitForKey();
        break;
    }
}
}
}

void get_info(){
    byte num = read_byte(99);
```



# ماشین لباسشویی

نام و نام خانوادگی: امیرحسین علی بخشی

شماره دانشجویی: ۹۷۳۱۰۹۶

استاد: آقای معصوم زاده

درس: آزمایشگاه ریزپردازنده و زبان اسمبلی

```
lcd.clear();
lcd.setCursor(0,0);
lcd.print(" # of modes : ");
lcd.setCursor(0,1);
lcd.print("      ");
lcd.print(num);
lcd.setCursor(0,3);
lcd.print(" press any key. ");
keypad.waitForKey();
}

void wash_time(byte Time,byte step_num){
    Time++;
    lcd.clear();
    digitalWrite(LED[step_num],HIGH);
    unsigned long currentMillis =0;
    unsigned long previousMillis = 0;
    lcd.setCursor(0,0);
    lcd.print(step_names[step_num]);
    int i = Time;
    while(i > 0){
        currentMillis = millis();
        if(currentMillis - previousMillis >= 1000) {
            previousMillis = currentMillis;
            i=i-1;
            lcd.setCursor(0,1);
            lcd.print(i);
        }
        char skey;
        skey = keypad.getKey();
        if(skey=='+'){
            write_byte(98,step_num); // save current step at 98
        }
    }
    // digitalWrite(LED[step_num],LOW);
}

void wash_mode(byte *data){
    for (int Step = 0; Step < 4; Step++){
        lcd.setCursor(0,0);
        lcd.print(step_names[Step]);
        wash_time(data[Step], Step);
    }
    lcd.clear();
    lcd.setCursor(0,1);
```





```
lcd.print("    finished    ");
lcd.setCursor(0,2);
lcd.print(" press any key. ");
Serial.println(step_names[4]);
for(int i=0;i<4;i++){
    digitalWrite(LED[i],HIGH);
}
write_byte(98,0); // current step (98) is ZERO
keypad.waitForKey();//enter any key
turn_off_LEDS();
}
////////////////////
void setup() {
    lcd.begin(16, 4);
    lcd.setCursor(0, 1);
    lcd.print("  A.Alibakhshi  ");
    lcd.setCursor(0, 2);
    lcd.print(" WashingMachine ");
    delay(DELAY_PERIOD * 3);
    lcd.clear();
    pinMode(LED1,OUTPUT);
    pinMode(LED2,OUTPUT);
    pinMode(LED3,OUTPUT);
    pinMode(LED4,OUTPUT);

    Wire.begin();
    Serial.begin(9600);

    mode=0;//set mode to default
}

void loop() {
    byte myData[4];
    byte chosen_mode[4]; // each mode: 4 bytes
    //menu
    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print("n:RESET,x:INFO");
    lcd.setCursor(0,1);
    lcd.print("+:ADD M.");
    lcd.setCursor(0,2);
    lcd.print("/:SELECT M.");
    lcd.setCursor(0,3);
    lcd.print("-:DEFAULT WASH");
```



```
char key;
key=keypad.waitForKey();

if(key=='n'){ // reset everything to default
    lcd.clear();
    lcd.setCursor(0,1);
    lcd.print("      SET TO      ");
    lcd.setCursor(0,2);
    lcd.print("    DEFAULT!    ");
    for(int i = 0; i < 4 ; i++){
        myData[i] = i + 1;
    }
    memory_write(100,myData,4);
    write_byte(99,1); // number of modes: 1 (only the DEFAULT mode from 100 t
o 104)
    memory_read(100,myData,4);

}else if(key=='+'){ // ADD MODE
    byte num = read_byte(99); // how many modes do we have?!
    enter_mode_settings(myData);
    memory_write(100+(4*num),myData,4); // write the new mode on memory
    write_byte(99,num+1); // update the number of modes

}else if(key=='/'){ // SELECT MODE
    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print("  SELECT  MODE  ");
    lcd.setCursor(0,1);
    byte num = read_byte(99);
    lcd.print("  # modes =");
    lcd.print(num);
    key=keypad.waitForKey();
    int mode_number_modified = 10;
    if(key-48<=num){
        lcd.clear();
        lcd.setCursor(0,1);
        lcd.print("/ : WASH");
        lcd.setCursor(0,2);
        lcd.print("x : CHANGE");
        mode_number_modified = key - 49;
        memset(chosen_mode,0,4);
        memory_read(100+(4*mode_number_modified),chosen_mode,4);
        key=keypad.waitForKey();
```



```
if(key=='/'){// wash it!
    wash_mode(chosen_mode);
}else if(key == 'x'){//change
    byte data[4];
    memset(data,0,4);
    lcd.clear();
    lcd.print("change mode");
    enter_mode_settings(data);
    memory_write(100+(4 * mode_number_modified),data,4);
}else{
    lcd.clear();
    lcd.setCursor(0,1);
    lcd.print("  PLEASE TRY  ");
    lcd.setCursor(0,2);
    lcd.print("    AGAIN!    ");
    delay(1000);
}
}

}else if(key=='-'){// DEFAULT WASH
    byte default_wash[4];
    memory_read(100,default_wash,4); //read from 100 to 104
    wash_mode(default_wash);

}else if(key=='x'){// get info
    get_info();

}else{// bad input
    lcd.clear();
    lcd.setCursor(0,1);
    lcd.print("  PLEASE TRY  ");
    lcd.setCursor(0,2);
    lcd.print("    AGAIN!    ");
    delay(1000);
}
}
```