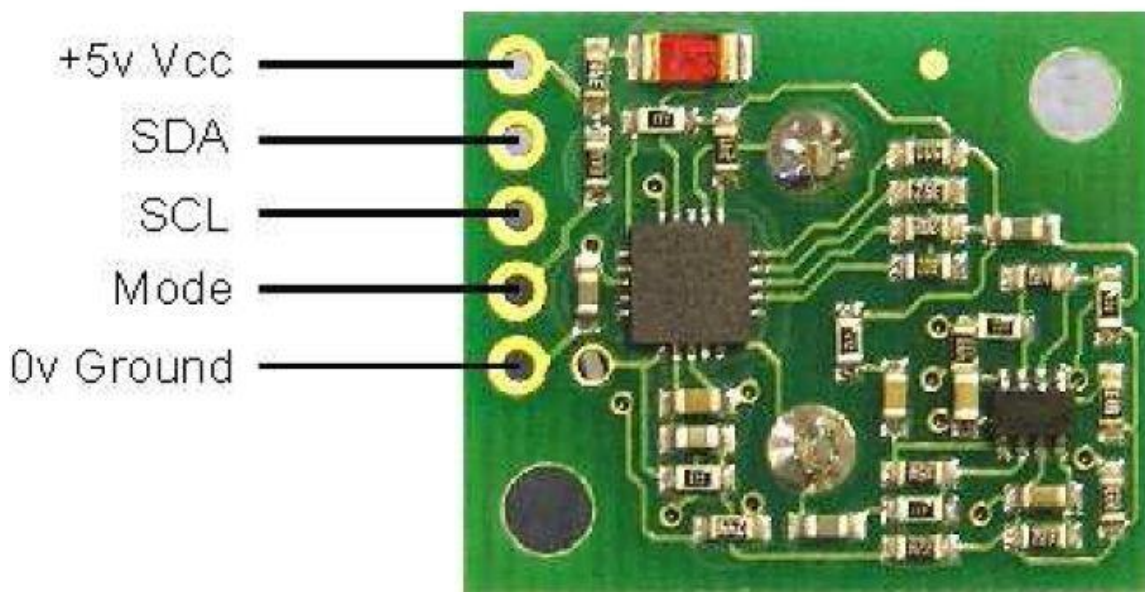


## آشنایی و راه اندازی ماژول الیتراسونیک SRF02

این ماژول دارای دو اینترفیس I2C Mode و یا Serial mode برای راه اندازی و خواندن فاصله می باشد. که در این جا به بررسی کامل ماژول به همراه برنامه عملی میکرو AVR جهت خواندن فاصله و نمایش آن بر روی LCD پرداخته می شود.

زمانی که از پشت ماژول را به سمت خود نگه میدارید اولین پایه سوکت VCC (5ولت رگوله شده) پایه دوم SDA (خط دیتا باس I2C) پایه سوم SCL (خط کلاک باس I2C) پایه چهارم مد ارتباطی (اگر صفر باشد یا رها باشد مد I2C اگر به 5 ولت وصل شود مد سریال) پایه پنجم صفر مدار .



باس I2C به این نحو کار می کند که همراه خط دیتا خط کلاک وجود دارد برای مشخص کردن یا همزمانی در خواندن یا ارسال دیتای معتبر بر روی خط دیتا.

در این پروژه میکرو نقش Master و ماژول نقش Slave را دارا می باشند.

ممکن است چندین دستگاه به باس I2C متصل باشند که برای تمایز هرکدام یک آدرس منحصر به فرد در این باس دارند. ماژول SRF02 نیز دارای همچنین آدرسی می باشد و برای تعیین آن باید ماژول را به 5 ولت وصل کنیم و طبق جدول زیر و تعداد روشن و خاموش شدن LED بر روی ماژول آدرس ماژول بدست می آید. همچنین می توان این آدرس را تغییر داد. (آدرس ماژول مورد تست ما 0XE0 بود).

Address		Long Flash	Short flashes
Decimal	Hex		
224	E0	1	0
226	E2	1	1
228	E4	1	2
230	E6	1	3
232	E8	1	4
234	EA	1	5
236	EC	1	6
238	EE	1	7
240	F0	1	8
242	F2	1	9
244	F4	1	10
246	F6	1	11
248	F8	1	12
250	FA	1	13
252	FC	1	14
254	FE	1	15

این ماژول برای تنظیم، راه اندازی و بدس آوردن مقدار فاصله دارای 6 رجیستر زیر می باشد.

Location	Read	Write
0	Software Revision	Command Register
1	Unused (reads 0x80)	N/A
2	Range High Byte	N/A
3	Range Low Byte	N/A
4	Autotune Minimum - High Byte	N/A
5	Autotune Minimum - Low Byte	N/A

مقدار فاصله ای که ماژول محاسبه می کند می تواند بر حسب سانتی متر، اینچ و یا زمان رفت و برگشت موج باشد. که به واسطه ی ریختن یکی از مقادیر زیر در Command Register واحد فاصله مشخص می شود

Command		Action
Decimal	Hex	
80	0x50	Real Ranging Mode - Result in inches
81	0x51	Real Ranging Mode - Result in centimeters
82	0x52	Real Ranging Mode - Result in micro-seconds

توضیحات برنامه:

```
SRF02_write(0x00,0x51);
```

در ابتدا باید در Command Reg مقدار 0x51 را بریزیم تا مقدار فاصله بر حسب سانتی متر بدست آید. تابع بالا این کار را انجام میدهد.

```
dish=256* SRF02_read(2);
```

فاصله ی محاسبه شده توسط ماژول در دو رجیستر ذخیره می شود که به بایت بالا و پایین تقسیم می شود و آدرس این دو رجیستر 2 و 3 می باشد که ابتدا بایت بالا خوانده می شود و برای تبدیل به دسیمال در 256 ضرب میشود.

```
#include <mega32.h>
#include <delay.h>
#include <stdio.h>
#include <stdlib.h>
#include <lcd.h>
#include <i2c.h>
//I2C Bus functions
#asm
. equ __i2c_port=0x1B ;PORTA
. equ __sda_bit=1 //SDA Port A.1
. equ __scl_bit=2 // SCL Port B.2
#endasm
//Alphanumeric LCD Module function
#asm
```

```

.equ __lcd_port=0x18 ;PORTB

#endasm

*/write a byte to the SRF02/*

void SRF02_write(unsigned char address, unsigned char data) (

i2c_start();

i2c_write(0xe0); //درس ماژول

i2c_write(address);

i2c_write(data);

i2c_stop();

*/ms delay to complete the write operation/*

delay_ms(10);

};

*/read a byte from the SRF02/*

unsigned char SRF02_read(unsigned char address) (

unsigned char data;

i2c_start();

i2c_write(0xe0);

i2c_write(address);

i2c_start();

i2c_write(0xe0|1);

data=i2c_read(0);

i2c_stop();

return data;

}

void main(void)

}

PORTA=0x00;

```

```

DDRA=0x00;
PORTB=0x00;
DDRB=0x00;
PORTC=0x00;
DDRC=0x00;
PORTD=0x00;
DDRD=0x00;
// I2C Bus initialization
i2c_init();
// LCD module initialization
lcd_init(16);
int dish,disl,dis;
char str1[],str2[];
while (1){
    SRF02_write(0x00,0x51);
    dish=256* SRF02_read(2);
    disl= SRF02_read(3);
    dis=dish+disl;
    dis=((dis)/(100)); //metr
    lcd_clear();
    lcd_gotoxy(0,0);
    ftoa(dis,2,str1);
    sprintf(str2,"dis:%5s",str1);
    lcd_puts(str2);
    delay_ms(200);
}
};

```

این برنامه تست شده است و به درستی جواب داده است.

تهیه و تنظیم : داود رئیسی

تدریس دروس برق

(تدریس تخصصی ریاضی مهندسی، کنترل و سیگنال)

انجام پروژه های کارشناسی ، ارشد و صنعتی برق و مهندسی پزشکی

Adr\_bme@yahoo.com