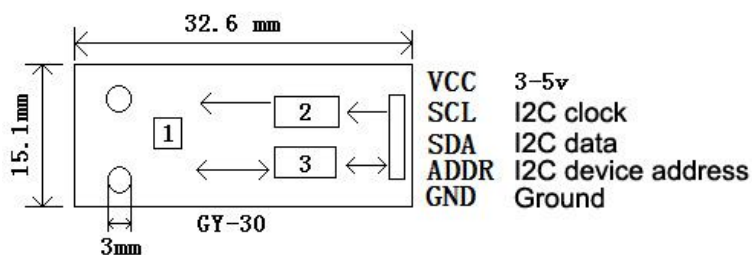


ماژول اندازه گیری شدت روشنایی BH1750FVI یا GY30

ماژول BH1750 یک برد مجهز به سنسور حساس به شدت نور است که دارای یک مبدل AD شانزده بیتی می باشد. پروتکل ارتباطی این ماژول رابط سریال I2C می باشد. داده های خروجی دیجیتال آن بر حسب لوکس می باشد. واحد بین المللی شدت روشنایی لوکس lx است. واحد دیگر شدت روشنایی بنام فوت کندل است.

لوکس: شدت روشنایی است که از یک شمع استاندارد در فاصله یک متری توسط سطح یک متر مربعی دریافت می شود (بر آن سطح تابیده میشود)

فوت کندل: شدت روشنایی است که از یک شمع استاندارد در فاصله یک فوتی توسط یک فوت مربعی دریافت شود.



- 1 BH1750FVI
- 2 Low-power 3.3V regulator (3-5V compatible)
- 3 Level conversion



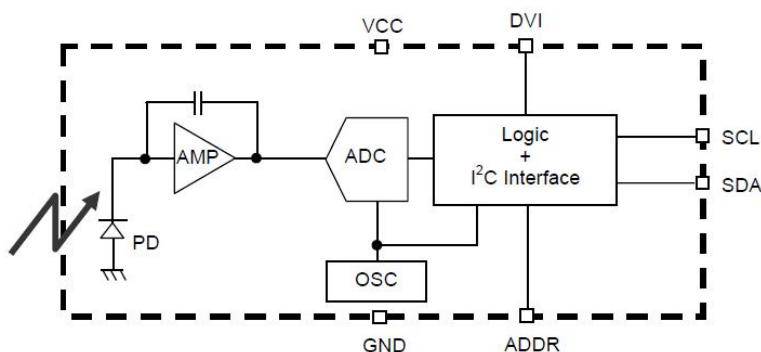
ویژگی های BH1750FVI

- ارتباط I2C
- دارای دو آدرس مختلف در ارتباط I2C
- حذف نویز 50Hz / 60Hz نور
- بدون نیاز به منبع نور برای اندازه گیری
- حساسیت بسیار کم نسبت به نور مادون قرمز
- محدوده وسیع و دقت بالا (1 ~ lx 65535)
- مبدل لومینانس به مقادیر دیجیتال

- پاسخ طیفی تقریباً مشابه چشم انسان
- محدوده تغذیه (2.4-4.5 V DC)

کاربرد

- اندازه گیری شدت روشنایی
- تنظیم خودکار نور صفحه در گوشی موبایل , دوربین , ...
- تنظیم اتوماتیک شدت روشنایی در هنگام عکاسی
- ...



راه اندازی ماژول BH1750FVI

برای اندازه گیری شدت روشنایی توسط این سنسور باید یکی از دو حالت Continuously H-resolution یا One time L-resolution mode انتخاب شود. برای انتخاب حالت اندازه گیری باید منطق پایه ADDR عوض شود که به صورت زیر است:

Continuously H-resolution mode (ADDR = 'L')

One time L-resolution mode (ADDR = 'H')

بعد از انتخاب وضعیت پایه ADDR نوبت به برنامه نویسی ماژول میرسد . این ماژول دارای دو آدرس مختلف در ارتباط I2C میباشد , که بسته به نوع انتخاب وضعیت ADDR باید عوض شود .

در ابتدا ادرس مورد نظر از طریق پروتکل I2C ارسال و سپس کد مربوط به ماژول برای اندازه گیری شدت روشنایی و در نهایت مقدار شدت روشنایی (16 بیت) خوانده میشود.

در زیر یک برنامه نمونه جهت تست این ماژول توسط کامپایلر BASCOM-AVR قرار داده شده است.

```
$regfile = "m8def.dat"
$crystal = 8000000
$hwstack = 64
$swstack = 64
$framesize = 150

#include "3310init.bas"

Dim Xl As Single
Dim High_byte As Byte
Dim Low_byte As Byte
Dim Xl_s As String * 6

Const Gy_30_address_write = &H46
Const Gy_30_address_read = &H47

'Define I/O and name them
Config Portd.0 = Output           'D/C (pin 4)
Config Portd.1 = Output           'SCE (pin 5)
Config Portd.2 = Output           'Reset (pin 8)

D3310dc Alias Portd.0
D3310ce Alias Portd.1
D3310re Alias Portd.2

'Sclk(pin 2) - Portd.3 - Sda(pin 3) - Portd.4
Config Spi = Soft , Din = Portd.5 , Dout = Portd.4 , Ss = None , Clock = Portd.3
Spiinit

'SOFT I2C CONFIG
Config Scl = Portc.5              ' needed if using soft I2C
Config Sda = Portc.4              ' needed if using soft I2C
Config I2cdelay = 20

Call D3310reset
Call D3310init
Call D3310clear

Call D3310position(0 , 0)
Call D3310print( "Digital Light")

Call D3310position(4 , 1)
Call D3310print( "Sensor")
```

```
Call D3310position(0 , 3)
Call D3310print( "R= 1-65535 LX")
```

```
Do
```

```
'ADDR='L'
I2cstart
I2cwbyte Gy_30_address_write
I2cwbyte &H10
I2cstop
```

```
Waitms 180
```

```
I2cstart
I2cwbyte Gy_30_address_read
I2crbyte High_byte , Ack
I2crbyte Low_byte , Nack
I2cstop
Waitms 10
```

```
Xl = High_byte * 256
Xl = Xl + Low_byte
Xl = Xl / 1.2
```

```
  Xl_s = Fusing(xl , "###.##")
  Call D3310position(0 , 5)
  Call D3310print(xl_s )
  Call D3310print( " LUX  ")
```

```
Waitms 100
```

```
Loop
End
$include "3310end.bas"
```