Designing the VEML3235P Into an Application

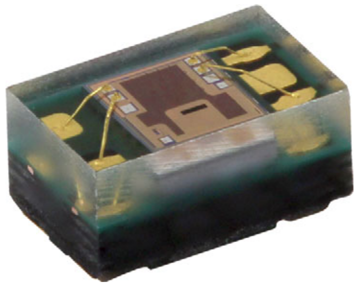
By Dio Yang

DESCRIPTION

Designed using CMOS process technology, the VEML3235P is an advanced ambient light sensor featuring an I^2C protocol interface. It is easy to operate via simple I^2C commands.

Incorporating a photodiode, amplifiers, and analog circuits within a single chip, it achieves spectral sensitivity optimized to closely match the response of the human eye.

Offering excellent temperature compensation, the VEML3235P also features a robust refresh rate setting that eliminates the need for an external RC low pass filter. A software shutdown mode reduces power consumption to less than 1 μA.



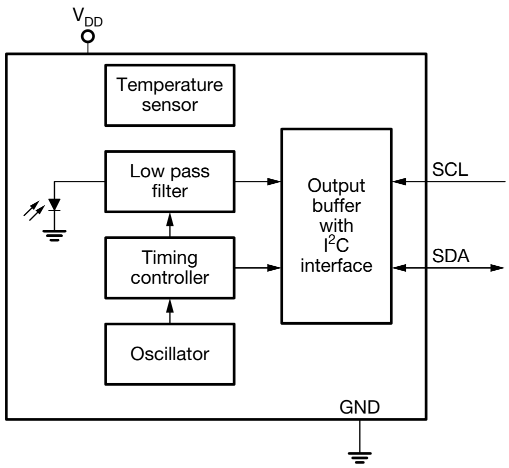


Fig. 1 - VEML3235P Block Diagram

Designing the VEML3235P Into an Application

APPLICATION CIRCUITRY FOR THE VEML3235P

Known for being a cost-effective solution, the VEML3235P is an ambient light sensor featuring an I^2C interface. You can easily access "light intensity" readings through its standard serial digital interface without complex calculations or external controller programming.

Locating a 0.1 µF capacitor near the V_{DD} pin in the circuit is recommended for power supply noise rejection. Ensuring proper I^2C communication, the design suggests using 2.2 kΩ pull-up resistors on the bus lines.

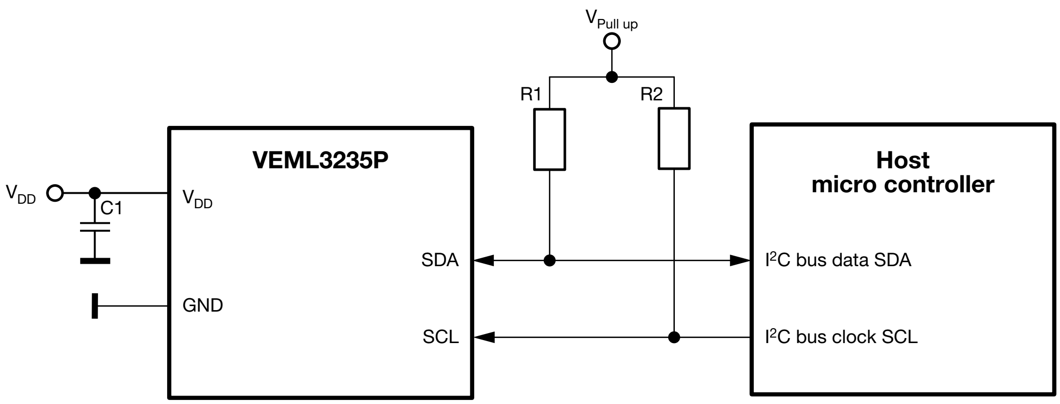


Fig. 2 - VEML3235P Application Circuit

Designing the VEML3235P Into an Application

REGISTERS OF THE VEML3235P

The VEML3235P has a 16-bit resolution per the G and W channels, and communicates via a 7-bit I²C protocol assigned to the 0x10 slave address.

COMMAND CODE DESCRIPTION

COMMAND CODE	REGISTER NAME	DATE BYTE LOW / HIGH	BIT	FUNCTION DESCRIPTION	R / W
0x00	Reserved	L	7	Set 0	R / W
	ALS/W_IT	L	6 : 4	ALS / W integration time setting (0 : 0 : 0) = 50 ms (0 : 0 : 1) = 100 ms (0 : 1 : 0) = 200 ms (0 : 1 : 1) = 400 ms (1 : 0 : 0) = 800 ms	
	Reserved	L	3 : 1	Set 0	
	SD	L	0	00 = power on 01 = initialized 11 = shut down	
		H	7		
	Reserved	H	6	Set 0	
	Gain1	H	5	0 = × 1, 1 = ×2	
	Gain2	H	4 : 3	(0 : 0) = ×1 (0 : 1) = ×2 (1 : 0) = reserved (1 : 1) = ×4	
	Reserved	H	2 : 1	Set 0	
	INIT	H	0	0 = INIT disable 1 = INIT enable	
0x04	W_LSB	L	7 : 0	W LSB data	R
	W_MSB	H	7 : 0	W MSB data	
0x05	ALS_LSB	L	7 : 0	ALS LSB data	
	ALS_MSB	H	7 : 0	ALS MSB data	
0x09	ID_L	L	7 : 0	ID part number: 3235 = 0011 0101	
	Reserved	H	7 : 0	Reserved	

Designing the VEML3235P Into an Application

MECHANICAL CONSIDERATIONS AND WINDOW CALCULATION FOR THE VEML3235P

The ambient light sensor will be placed behind a window or cover. The window material should be completely transmissive to visible light (400 nm to 700 nm). For optimal performance the window size should be large enough to maximize the light irradiating the sensor. In calculating the window size, the only dimensions that the design engineer needs to consider are the distance from the top surface of the sensor to the outside surface of the window and the size of the window. These dimensions will determine the size of the detection zone.

First, the center of the sensor and center of the window should be aligned. The VEML3235P has an angle of half sensitivity of about ± 55°, as shown in the figure below.

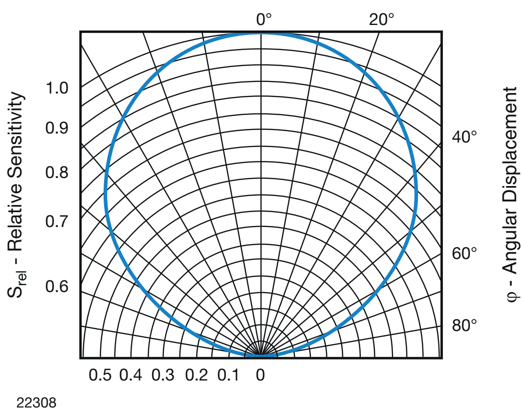


Fig. 3 - Relative Sensitivity vs. Angular Displacement

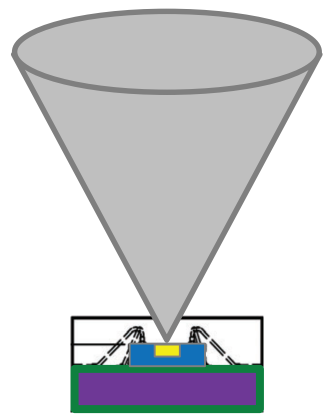


Fig. 4 - Angle of Half Sensitivity: Cone

Remark:

This wide angle and the placement of the sensor as close as possible to the cover is needed if it should show comparable results to an optometer, which also detects light reflections from the complete surroundings.

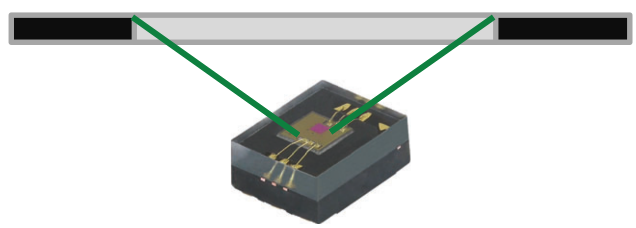
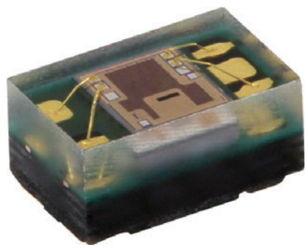


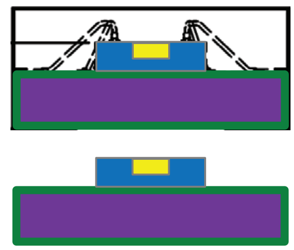
Fig. 5 - Windows Above Sensitive Area

The size of the window is simply calculated according to triangular rules. The dimensions of the device are shown within the datasheet, and with the known distance below the window's upper surface and the specified angle below the given window diameter (w), the best results are achieved.

Designing the VEML3235P Into an Application

VEML3235P





Dimensions (L × W × H in mm): 2.0 × 1.25 × 1

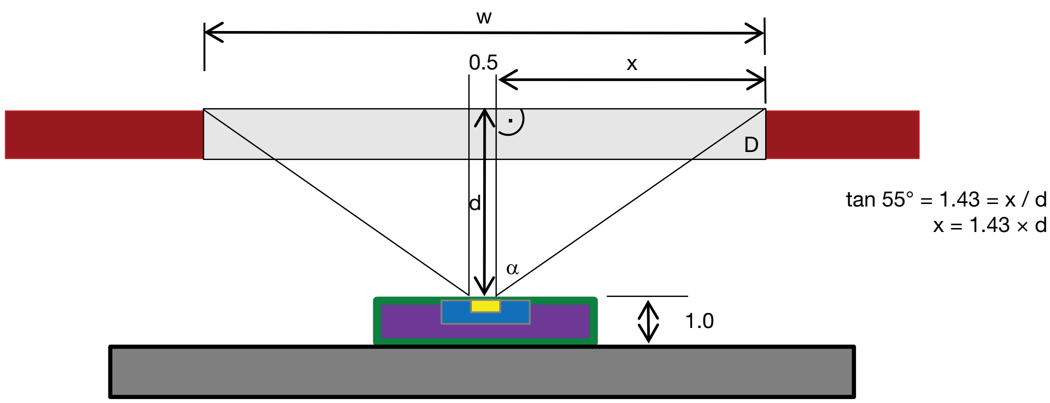


Fig. 6 - Window Area for an Opening Angle of ± 55°

The calculation is then: $\tan \alpha = x / d \rightarrow$ with $\alpha = 55°$ and $\tan 55°$ $1.43 = x / d \rightarrow x = 1.43 \times d$
Then the total width is $w = 0.5\text{ mm} + 2 \times x$.

d = 0.5 mm	→	x = 0.72 mm	→	w = 0.5 mm + 1.44 mm	=	1.94 mm
d = 1.0 mm	→	x = 1.43 mm	→	w = 0.5 mm + 2.86 mm	=	3.36 mm
d = 1.5 mm	→	x = 2.15 mm	→	w = 0.5 mm + 4.30 mm	=	4.80 mm
d = 2.0 mm	→	x = 2.86 mm	→	w = 0.5 mm + 5.72 mm	=	6.22 mm
d = 2.5 mm	→	x = 3.58 mm	→	w = 0.5 mm + 7.16 mm	=	7.66 mm
d = 3.0 mm	→	x = 4.29 mm	→	w = 0.5 mm + 8.58 mm	=	9.08 mm

A smaller window is also sufficient if reference measurements can be done and / or if the output result does not need to be as exact as an optometer.

Designing the VEML3235P Into an Application

PROGRAMMING REFERENCE CODE

```
//--------------------------------------------------------------------------
// Global definition for VEML3235P registers
//--------------------------------------------------------------------------

// 7bits SLAVE_ADD=0x10
#define VEML3235P_SLAVE_ADD 0x10
#define CONF      0x00
#define W_DATA    0x04
#define G_DATA    0x05

//--------------------------------------------------------------------------
// C main function
//--------------------------------------------------------------------------
void main ()
{
  WORD  VEML3235P_DATA_W;
  WORD  VEML3235P_DATA_G;

  //-------------------------------------------------------------------------------
  // Write Shut Down Command to VEML3235P
  // Command Code: 0x00
  // Low Byte           0x11 (0001:0001)
  // ALS_IT = 100 ms
  // SD = 1: shut down

  // High Byte    0x80 (1000:0000)
  // SD = 1: shut down
  // Gain1 = *1, Gain2 = *1
  // INIT = 0: INIT disable
  //-------------------------------------------------------------------------------
  // Write Initial Command to VEML3235P
  // Command Code: 0x00
  // Low Byte           0x10 (0001:0000)
  // ALS_IT = 100 ms
  // SD = 0: Initialized

  // High Byte    0x81 (1000:0001)
  // SD = 1: Initialized
  // Gain1 = *1, Gain2 = *1
  // INIT = 1: INIT enable

  //-------------------------------------------------------------------------------
  // Write Enable Command to VEML3235P
  // Command Code: 0x00
  // Low Byte           0x10 (0001:0000)
  // ALS_IT = 100 ms
  // SD = 0: power on

  // High Byte    0x00 (0000:0000)
  // SD = 0: power on
  // Gain1 = *1, Gain2 = *1
  // INIT = 0: INIT disable
  // -----------------------------------------------------------------------------


  // Shut Down Color Sensor
  WriteBytes(VEML3235P_SLAVE_ADD, CONF, 0x11, 0x80, 2);

  // Initial Color Sensor
   WriteBytes(VEML3235P_SLAVE_ADD, CONF, 0x10, 0x81, 2);

  // Enable Color Sensor
   WriteBytes(VEML3235P_SLAVE_ADD, CONF, 0x10, 0x00, 2);


  // Read VEML3235P Data Loop
  while (1)
  {
    Delay(150);

    // Read VEML3235P W Channel Data
    VEML3235P_DATA_W = read_VEML3235P_data(W_DATA);
    // Read VEML3235P G Channel Data
    VEML3235P_DATA_G = read_VEML3235P_data(G_DATA);
  }

//--------------------------------------------------------------------------
// FUNCTION NAME: read_VEML3235P_data
//
// DESCRIPTION:
//--------------------------------------------------------------------------
WORD read_VEML3235P_data(WORD channel)
{
  BYTE buff[2];
  BYTE lsb, msb;
  WORD channel_data;

  ReadBytes(VEML3235P_SLAVE_ADD, channel, buff, 2);
  lsb = buff[0];
  msb = buff[0];
  channel_data = ((WORD)msb << 8) | (WORD)lsb;
  return channel_data;
}
```