GY-39 Weather Sensor Guide

Background

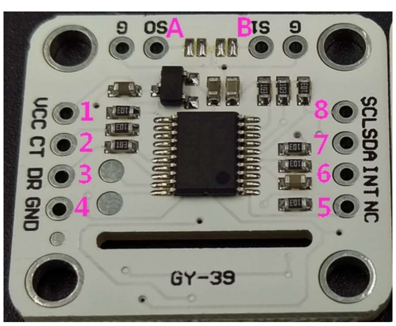
GY-39 sensor board is a low cost, pressure, humidity, light intensity sensor with onboard microcontroller (MCU). GY-39 is a low power device which required supply voltage between 3 to 5 volts. The microcontroller is to gather data from ME280, and MAX44009 sensors, calculate, and generate output. It is also used for device configuration such as setting the UART baud rate, disable/enable outputs, and setting I2C address. Two communication modes are supported, either by serial UART (TTL voltage level), or by I2C. There is also an option to work with ME280, and MAX44009 sensors directly.

The default setting is using onboard MCU and serial communication with the UART configurate for 9600bps, N, 8, 1.

Specification

Reference Data	Reference Values
Sensing Temperature	-40 ~ 85c
Sensing Humidity	0 ~ 100%
Sensing Light Intensity	0.025 ~ 188000lux
Sensing Barometric Pressure	300 ~ 1100hpa
Data Acquisition Frequency	10Hz
Supply Voltage	3 ~ 5V
Current Draw	5mA
Operating Temperature	-40 ~ 85c
Storage Temperature	-40 ~ 125c
Board Dimension	24.3mm X 26.7mm
Onboard Sensor ICs	ME280 + MAX44009

Pins Information



Pin1	VCC	Supply Voltage (3V – 5V)
Pin2	CT	MCU_UART_TX / MCU_I2C_SCL
Pin3	DR	MCU_UART_RX / MCU_I2C_SDA

Pin4	GND	Ground
Pin5	NC	Reserved, don't connect
Pin6	INT	To Disable MAX44009 (No MCU mode)
Pin7	SDA	SDA for ME280 and MAX44009 (No MCU mode)
Pin8	SCL	SCL for ME280 and MAX44009 (No MCU mode)
PinA	S0	MCU_UART by Default / MCU_I2C when S0 pull to ground
PinB	S1	MCU mode by Default / No MCU mode when S1 pull to ground

Switches Information (1=VCC, 0=GND)

S0	S1	Mode
1	1	MCU_UART (Default), Data and Control
0	1	MCU_I2C, Data
1	0	Not covered in this menu. Refer to ME280 and MAX44009 datasheets
0	0	Not covered in this menu. Unknown consequences.

Default MCU UART Mode Operation (S0=1, S1=1)

In this mode, the weather board automatically output sensors' data using the UART set to 9600bps baud rate. The output message is in bytes started with headers and ended with a checksum byte. The pattern of the message starts with two preamble bytes and ended with a lower byte for checksum.

For light intensity data (from MAX44009), assume the MCU have the following message (in Hex):

[5A,5A,15,04,00,00,00,B4,81]

Here are how we can consume the message:

[5A,5A] Two preamble bytes

[15] Light intensity data type from MAX44009

[04] Payload has 4 bytes

[00,00,00,B4] Light intensity measurement in big endian, for calculation:

(00 << 24) | (00 << 16) | (00 << 8) | B4 = B4

DEC(B6) = 182

Light intensity = 182 / 100 = 1.82 Lux

[81] Lower byte of the sum of all previous bytes in the message. To validate the data is not corrupted, compare the checksum with the running sum by this calculation:

(5A + 5A + 15 + 04 + 00 + 00 + 00 + B4) & 0xFF = 81

For weather data (from BME280), assume the MCU have the following message (in Hex):

[5A,5A,45,0A,09,88,00,95,33,47,10,39,01,2B,18]

Here are how we can consume the message:

[5A,5A] Two preamble bytes

[45] Weather data type from BME280

[0A] Payload has 10 bytes

[09,88] Temperature in big endian, for calculation:

(09 << 8) | 88 = 0908

DEC(0908) = 2440

Temperature = 2440 / 100 = 24.40 c

[00,95,33,47] Pressure in big endian, for calculation:

(00 << 24) | (95 << 16) | (33 << 8) | 47 = 953347

DEC(953347) = 9777991

Pressure = 9777991/ 100 = 97779.91 pa

[10,39] Humidity in big endian, for calculation:

(10 << 8) | 39= 0908

DEC(0908) = 1039

Humidity = 4153 / 100 = 41.53 %

[01,2B] Humidity in big endian, for calculation:

(01 << 8) | 2B = 012B

DEC(012B) = 299

Elevation = 299 m

[18] Lower byte of the sum of all previous bytes in the message. To validate the data is not corrupted, compare the checksum with the running sum by this calculation:

(5A + 5A + 45 +0A + 09 + 88 + 00 + 95 + 33 + 47 + 10 + 39 + 01 + 2B) & 0xFF = 18