



**24-Bit ADC Module**

**BMH23M001**

# **Arduino Library Description**

Revision: V1.00    Date: April 19, 2023

[www.bestmodulescorp.com](http://www.bestmodulescorp.com)

## Contents

<b>Introduction</b> .....	<b>3</b>
<b>Arduino Lib Functions</b> .....	<b>3</b>
<b>Arduino Lib Download and Installation</b> .....	<b>6</b>
<b>Arduino Example</b> .....	<b>7</b>
Example: readADCData .....	7

## Introduction

The Best Modules BMH23M001 is a high precision 24-bit ADC module, which uses the I<sup>2</sup>C communication method. This document provides the description of the BMH23M001 Arduino Lib functions and how to install the Arduino Lib. The example demonstrates the function of obtaining the sensor AD value.

## Arduino Lib Functions

Arduino Lib Name: BMH23M001		Lib Version: V1.0.1
<b>Constructors &amp; Initialisation</b>		
1	BMH23M001(TwoWire *theWire=&Wire)	
	Description	Constructor, configure the wire parameters
	Parameter	*theWire: I <sup>2</sup> C communication interface selection
	Return Value	—
	Note	—
2	void begin(uint8_t i2c_addr=0x68)	
	Description	Module initialisation, registers and parameters are configured as default values
	Parameter	i2c_addr: I <sup>2</sup> C communication address, 0x68
	Return Value	void
	Note	Parameter configuration initialisation: No clock debounce Data rate=10Hz External reference voltage pair Offset=+0V Differential input signal gain=32 Differential reference voltage gain=1/2
<b>Performance Functions</b>		
3	void beginADC()	
	Description	Start the ADC conversion
	Parameter	—
	Return Value	void
	Note	Cyclically sampling data at a specific data rate
4	long readADCData()	
	Description	Obtain the ADC conversion data
	Parameter	—
	Return Value	ADC value N: Converted data
	Note	—
5	void endADC()	
	Description	Make the module to enter the sleep mode
	Parameter	—
	Return Value	void
	Note	—
6	void powerDown()	
	Description	Make the module to enter the power down mode
	Parameter	—
	Return Value	void
	Note	—

7	void wakeup()	
	Description	Wake up module
	Parameter	—
	Return Value	void
8	void writeReg(uint8_t regAddr, uint8_t regData)	
	Description	Register write operation
	Parameter	regAddr: Register address regData: Data to be written
	Return Value	void
9	void readReg(uint8_t rbuf[], uint8_t regAdd)	
	Description	Register read operation
	Parameter	rbuf[]: Used to store read data regAdd: Register address
	Return Value	void
<b>Parameter Configuration</b>		
10	bool setADConverter(uint8_t sampleRate, uint8_t reference)	
	Description	Configure ADC
	Parameter	sampleRate: Data rate 0x00(RATE_5Hz): Data rate=5Hz 0x01(RATE_10Hz): Data rate=10Hz 0x02(RATE_20Hz): Data rate=20Hz 0x03(RATE_40Hz): Data rate=40Hz 0x04(RATE_80Hz): Data rate=80Hz 0x05(RATE_160Hz): Data rate=160Hz 0x06(RATE_320Hz): Data rate=320Hz 0x07(RATE_640Hz): Data rate=640Hz Reference: Reference voltage pair 0x00(REFERENCE_VCM_AVSS): Internal reference voltage pair 0x01(REFERENCE_VREFP_VREFN): External reference voltage pair
	Return Value	Execution result: false: Setting failed true: Setting succeeded
11	bool setDecset(uint8_t decset)	
	Description	Configure differential input signal offset
	Parameter	decset: Signal offset 0x00(DECSET_0): Offset +0V 0x01(DECSET_P5): Offset +0.5V×ΔVR_I 0x02(DECSET_P25): Offset +0.25V×ΔVR_I 0x03(DECSET_P75): Offset +0.75V×ΔVR_I 0x05(DECSET_N5): Offset -0.5V×ΔVR_I 0x06(DECSET_N25): Offset -0.25V×ΔVR_I 0x07(DECSET_N75): Offset -0.75V×ΔVR_I
	Return Value	Execution result: false: Setting failed true: Setting succeeded
	Note	—

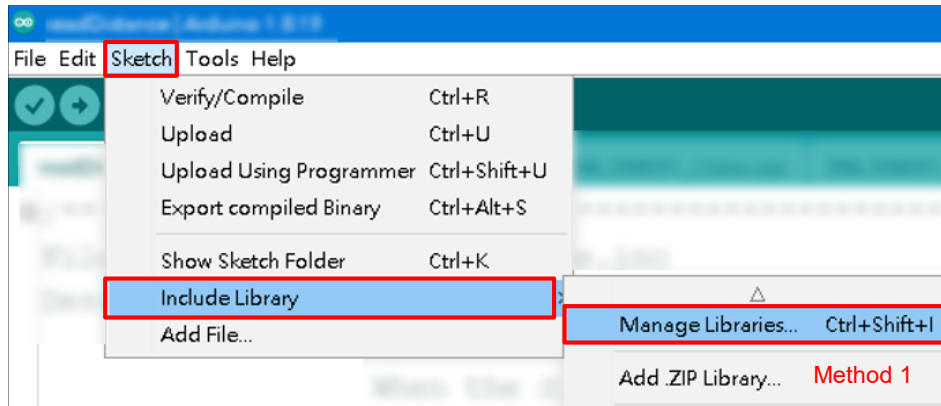
12	<b>bool setGain(uint8_t gain, uint8_t vrefGain)</b>	
	Description	Configure gain
	Parameter	gain: PGA gain 0x00(GAIN_1): Differential input signal gain=1 0x01(GAIN_2): Differential input signal gain=2 0x02(GAIN_4): Differential input signal gain=4 0x03(GAIN_8): Differential input signal gain=8 0x04(GAIN_16): Differential input signal gain=16 0x05(GAIN_32): Differential input signal gain=32 0x0D(GAIN_64): Differential input signal gain=64 0x0E(GAIN_128): Differential input signal gain=128 vrefGain: Reference voltage gain 0x00(VREF_GAIN_1): Differential reference voltage gain=1 0x01(VREF_GAIN_0_5): Differential reference voltage gain=1/2 0x02(VREF_GAIN_0_25): Differential reference voltage gain=1/4
	Return Value	Execution result: false: Setting failed true: Setting succeeded
	Note	—
13	<b>bool setChannel(uint8_t channel)</b>	
	Description	Configure signal input channels IN1 and IN2
	Parameter	channel: Signal channel 0x00(CHSP_AN0_N_AN1): Wheatstone bridge sensor interface 0x01(CHSP_AN2_N_AN3): Analog signal input interface
	Return Value	Execution result: false: Setting failed true: Setting succeeded
	Note	—
14	<b>bool setDebounce(uint8_t I2Cdebounce)</b>	
	Description	Configure I <sup>2</sup> C debounce time
	Parameter	I2Cdebounce: I <sup>2</sup> C debounce time (0x00)DEBOUNCE_NO: No clock debounce (0x01)DEBOUNCE_2_CLOCK: 2 clock debounce (0x11)DEBOUNCE_4_CLOCK: 4 clock debounce
	Return Value	Execution result: false: Setting failed true: Setting succeeded
	Note	—

# Arduino Lib Download and Installation

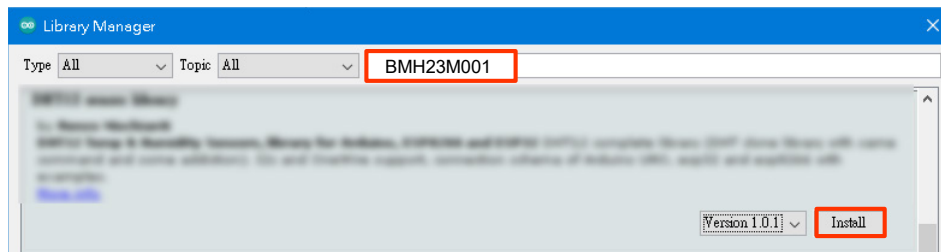
BMH23M001 Library: Refer to the following two methods to install the BMH23M001 Arduino Library.

## Method 1: Search for installation

Arduino IDE→Sketch→Include Library→Manage Libraries...→Search BMH23M001→Install



Search for Installation Step 1

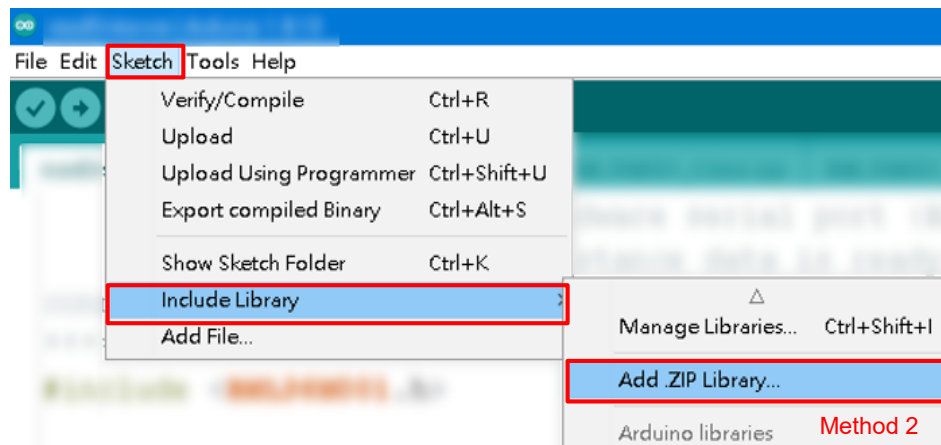


Search for Installation Step 2

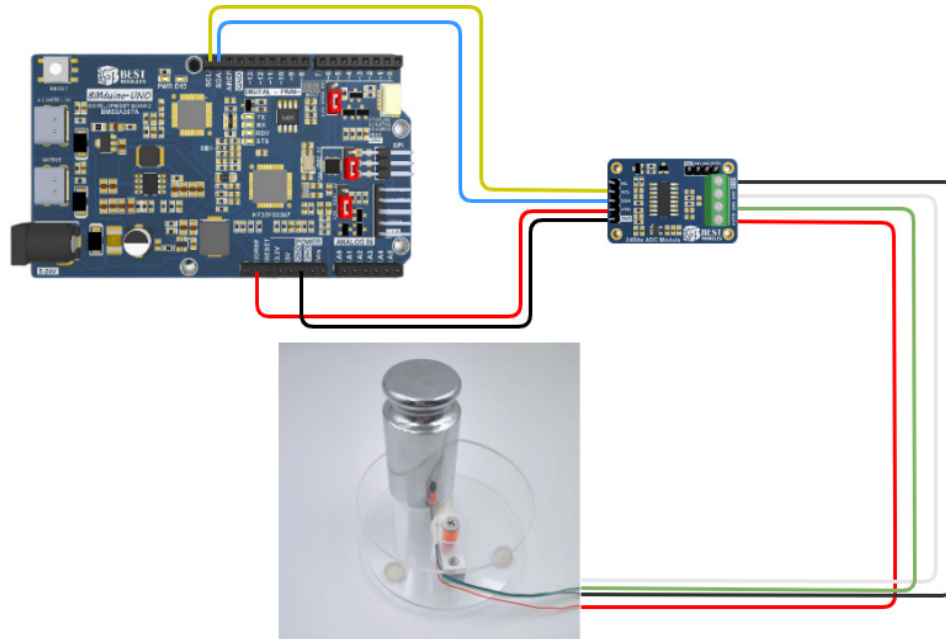
## Method 2: Download the .ZIP library before adding it

Download the Arduino example (BMH23M001 Library) under the DOCUMENTS menu from the Best Modules website (<https://www.bestmodulescorp.com/bmh23m001.html#tab-product2>).

Add .ZIP library: Arduino IDE→Sketch→Include Library→Add .ZIP Library...



## Arduino Example



**Physical Connection Diagram**

### Example: readADCData

Function: Obtain the sensor AD value and display it in the serial monitor.

1. Open the example: File→Examples→Select Lib (BMH23M001)→Select example (readADCData)
2. Example Description:

- a. Create object & initialise object

```
#include "BMH23M001.h"
BMH23M001 myADC(&Wire); // Create object
void setup()
{
  Serial.begin(9600); // Configure the serial monitor
  myADC.begin(); // Module initialisation, registers are configured
                // as default values
  myADC.setChannel(CHSP_AN0_N_AN1); // Configure signal input interface
  myADC.beginADC (); // Start ADC conversion
}
```

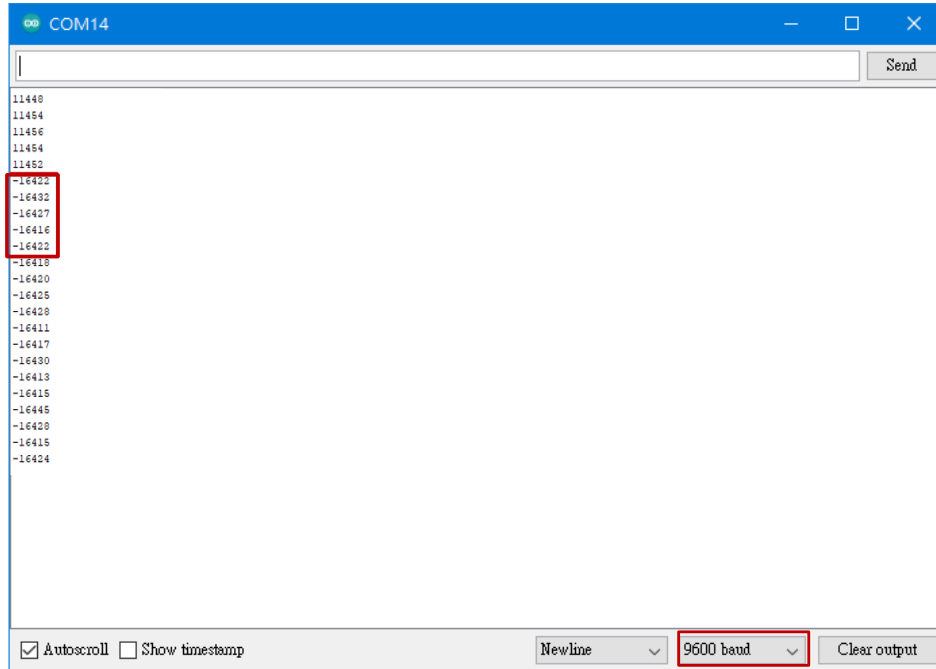
- b. Obtain the sensor value and display it in the serial monitor

```
void loop()
{
  Serial.println(myADC.readADCData ()); // Obtain the ADC conversion
                                        // data and display it in the
                                        // serial monitor
}
```

Load	ADC Value
No load	-16422
500g weight	11455
1,000g weight	39348
1,500g weight	67236

ADC value is linear with weight.

- Open the serial monitor and set the baud rate to be 9600. The serial monitor will display sensor data as follows.



Copyright© 2023 by BEST MODULES CORP. All Rights Reserved.

The information provided in this document has been produced with reasonable care and attention before publication, however, BEST MODULES does not guarantee that the information is completely accurate. The information contained in this publication is provided for reference only and may be superseded by updates. BEST MODULES disclaims any expressed, implied or statutory warranties, including but not limited to suitability for commercialization, satisfactory quality, specifications, characteristics, functions, fitness for a particular purpose, and non-infringement of any third-party's rights. BEST MODULES disclaims all liability arising from the information and its application. In addition, BEST MODULES does not recommend the use of BEST MODULES' products where there is a risk of personal hazard due to malfunction or other reasons. BEST MODULES hereby declares that it does not authorize the use of these products in life-saving, life-sustaining or safety critical components. Any use of BEST MODULES' products in life-saving/sustaining or safety applications is entirely at the buyer's risk, and the buyer agrees to defend, indemnify and hold BEST MODULES harmless from any damages, claims, suits, or expenses resulting from such use. The information provided in this document, including but not limited to the content, data, examples, materials, graphs, and trademarks, is the intellectual property of BEST MODULES (and its licensors, where applicable) and is protected by copyright law and other intellectual property laws. No license, express or implied, to any intellectual property right, is granted by BEST MODULES herein. BEST MODULES reserves the right to revise the information described in the document at any time without prior notice. For the latest information, please contact us.