

Alcohol Digital Sensor

BM22S3421-1
Arduino Library Description

Revision: V1.10 Date: February 02, 2024

www.bestmodulescorp.com

Contents

Introduction	3
Arduino Lib Functions	3
Arduino Lib Download and Installation	6
Arduino Example	7
Example: readAlcLevelAndADValue	7

Introduction

The Best Modules BM22S3421-1 is an alcohol digital sensor, which uses the UART communication mode. This document provides the description of the BM22S3421-1 Arduino Lib functions and how to install the Arduino Lib. The example uses the BMA34M421 module to demonstrate the function of reading alcohol gas concentration information.

Applicable types:

Part No.	Description
BM22S3421-1	Alcohol digital sensor
BMA34M421	On-board BM22S3421-1 sensor

Arduino Lib Functions

Arduino Lib name: BM22S3421-1		Lib version: V1.0.1
Constructors & Initialisation		
1	BM22S3421_1(uint8_t statusPin, HardwareSerial *theSerial=&Serial)	
	Description	Constructor, uses hardware UART
	Parameter	statusPin: STATUS pin, which is connected to the BM22S3421-1 STATUS pin or the BMA34M421 STA pin *theSerial: select hardware UART communication interface (default to Serial interface)
	Return Value	—
	Note	—
2	BM22S3421_1(uint8_t statusPin, uint8_t rxPin, uint8_t txPin)	
	Description	Constructor, uses software UART
	Parameter	statusPin: STATUS pin, which is connected to the BM22S3421-1 STATUS pin or the BMA34M421 STA pin rxPin: RX pin, which is connected to the BM22S3421-1 or the BMA34M421 TX pin txPin: TX pin, which is connected to the BM22S3421-1 or the BMA34M421 RX pin
	Return Value	—
	Note	—
3	void begin()	
	Description	Module initialisation
	Parameter	—
	Return Value	void
	Note	—
Parameter Check		
4	uint8_t getSTATUS()	
	Description	Get the STATUS pin level
	Parameter	—
	Return Value	STATUS pin level: 0: Low level 1: High level
	Note	The high or low level indicates whether the module alarms, the level can be set when alarming (default to high level)
5	uint16_t readADValue()	
	Description	Read the real-time alcohol gas A/D value
	Parameter	—
	Return Value	12-bit A/D value
	Note	The greater the A/D value, the higher the alcohol gas concentration It is generally used to obtain data actively and recommended to use after the serial port automatic output function is disabled ⁽²⁾

6	uint8_t readAlcLevel()	
	Description	Read the alcohol gas concentration level
	Parameter	—
	Return Value	Alcohol gas concentration level: 1: Level 1 (low concentration) 2: Level 2 3: Level 3 4: Level 4 (high concentration)
	Note	It is generally used to obtain data actively and recommended to use after the serial port automatic output function is disabled ⁽²⁾
7	uint8_t readParam(uint8_t cmd, uint8_t addr)	
	Description	Read the module parameter
	Parameter	cmd: Instruction code addr: Address code
	Return Value	Module parameter
	Note	Refer to the BM22S3421-1 datasheet instruction table (special query instruction)
8	bool isInfoAvailable()	
	Description	Check whether the module automatic output information is received (14-byte)
	Parameter	—
	Return Value	Receive status: true: has received false: not received
	Note	It is required to use after the serial port automatic output function is enabled. The module is enabled by factory default.
9	void readInfoPackage(uint8_t array[])	
	Description	Read the module automatic output information
	Parameter	array[0]~array[4]: Fixed data array[5]: Alcohol gas real-time A/D value high byte array[6]: Alcohol gas real-time A/D value low byte array[7]: Alcohol gas concentration level, the valid values are 1~4, which indicates the level 1~level 4 array[8]: Sensor status bit 0=1: Calibration in progress bit 1~bit 4: Reserved bit 5=1: Calibration completed bit 6=1: Module fault bit 7=1: Module alarm array[9]: Calibration countdown. The default calibration time is 60s. A value of 0 indicates that the calibration is completed array[10]: Power-on warm-up countdown. The warm-up time is 180s. A value of 0 indicates that the warm-up operation is completed array[11]: Software version number high byte. A value of 0x11 indicates that the software version number is 1.1.x array[12]: Software version number low byte. A value of 0x02 indicates that the software version number is x.x.2 array[13]: Check code. Take the lower 8 bits of the sum of the first 13 bytes, complement and increment by one
	Return Value	void
	Note	This function should be used after the "if (isInfoAvailable() == true)" instruction is executed
Parameter Setting Function		
10	uint8_t writeCommand(uint8_t cmd, uint8_t addr, uint8_t param)	
	Description	Write parameters to the module
	Parameter	cmd: Instruction code addr: Address code param: Parameters to be written
	Return Value	Execution status ⁽¹⁾
	Note	Refer to the BM22S3421-1 datasheet instruction table (special modification instruction)

General Function		
11	uint8_t reset()	
	Description	Reset the module
	Parameter	—
	Return Value	Execution status ⁽¹⁾
	Note	—
12	uint8_t requestInfoPackage(uint8_t array[])	
	Description	Actively obtain all the information of the module
	Parameter	array[0]~array[4]: Fixed data array[5]: Alcohol gas real-time A/D value high byte array[6]: Alcohol gas real-time A/D value low byte array[7]: Alcohol gas concentration level, the valid values are 1~4, which indicates the level 1~level 4 array[8]: Sensor status bit 0=1: Calibration in progress bit 1~bit 4: Reserved bit 5=1: Calibration completed bit 6=1: Module fault bit 7=1: Module alarm array[9]: Calibration countdown. The default calibration time is 60s. A value of 0 indicates that the calibration is completed array[10]: Power-on warm-up countdown. The warm-up time is 180s. A value of 0 indicates that the warm-up operation is completed array[11]: Software version number high byte. A value of 0x11 indicates that the software version number is 1.1.x array[12]: Software version number low byte. A value of 0x02 indicates that the software version number is x.x.2 array[13]: Check code. Take the lower 8 bits of the sum of the first 13 bytes, complement and increment by one
	Return Value	Execution status ⁽¹⁾
	Note	It is generally used to obtain data actively and recommended to use it after the serial port automatic output function is disabled ⁽²⁾
13	uint8_t restoreDefault()	
	Description	Reset the module parameters to their factory settings
	Parameter	—
	Return Value	Execution status ⁽¹⁾
	Note	—
14	uint8_t calibrateModule(uint8_t calibrateMode)	
	Description	Alcohol concentration level calibration
	Parameter	calibrateMode: calibration function selection 0x01: Calibration trigger point 1 (Level 1/2 judgment threshold) 0x02: Calibration trigger point 2 (Level 2/3 judgment threshold) 0x03: Calibration trigger point 3 (Level 3/4 judgment threshold) 0x06: Trigger the calibration check and update the calibration point 1~3 check value
	Return Value	Execution status ⁽¹⁾
	Note	1. Users can customize the concentration level using this function. 2. There is no fixed order for calibration points 1/2/3, it can be calibrated in any order and select any one, two or three of these points. 3. The calibration point must be set according to the value, and calibration point 1 < calibration point 2 < calibration point 3. 4. After the point calibration is completed, the calibration check instruction needs to be used, the device will enter the calibration completion state.

Note: 1. 0 – Instruction executed succeed. 1 – Module response data error. 2 – Communication time-out

2. Serial port automatic output function:

Enable: use writeCommand(0xe0,0x1e, AUTO_MODE)

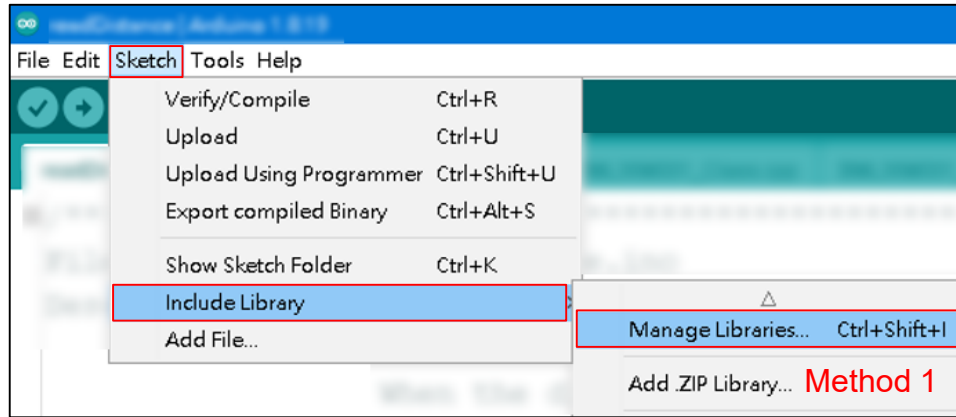
Disable: use writeCommand(0xe0,0x1e, CMD_MODE)

Arduino Lib Download and Installation

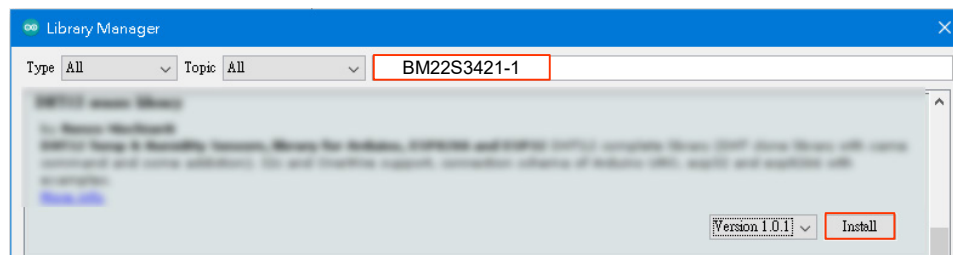
BM22S3421-1 Library: Refer to the following two methods to install the BM22S3421-1 Arduino Library

Method 1: Search for installation

Arduino IDE → Sketch → Include Library → Manage Libraries → Search BM22S3421-1 → Install



Search for Installation Step 1

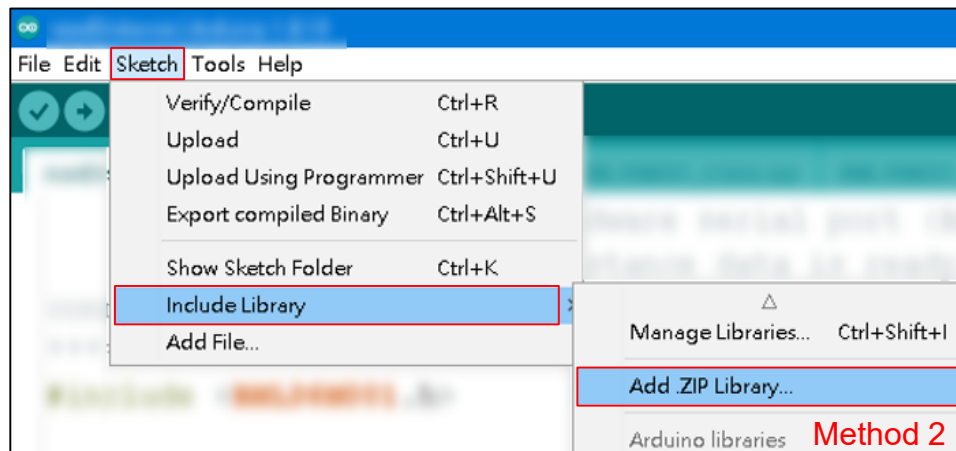


Search for Installation Step 2

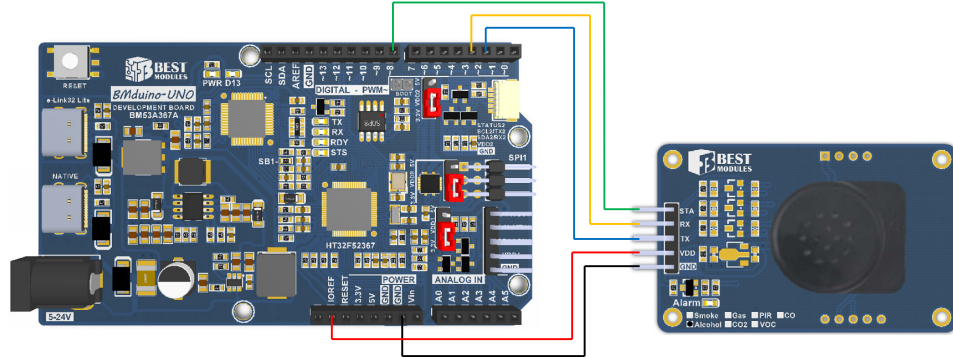
Method 2: Download the .ZIP library before adding it

Download the Arduino example (BM22S3421-1 Library) under the DOCUMENTS menu from the Best Modules website (<https://www.bestmodulescorp.com/bm22s3421-1.html>).

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



Arduino Example



Physical Connection Diagram

Example: readAlcLevelAndADValue

Example function: Receive the module automatic output information every second and print the concentration level and A/D value of the alcohol gas to the serial monitor

1. Open an example program: Arduino IDE → File → Examples → Select Lib (BM22S3421-1) → Select the program (readAlcLevelAndADValue)
2. Example program description:
 - a. Create object & Initialise the module

```
#include < BM22S3421-1.h> // Call the BM22S3421-1 library
/* Create arrays and variables for storing data */
uint8_t moduleInfo[14] = {0};
uint16_t ADValue, AlcLevel;
/* Create object & Set Software serial pin */
BM22S3421_1 Alc(8, 2, 3); // Software serial: D8:STATUS, D2:RX, D3:TX
void setup()
{
  Alc.begin(); // Module initialisation
  Serial.begin(9600); // Serial monitor initialisation,
                    // baud rate is 9600bps
  /* Wait for the end of the module warm-up */
  Serial.println("Module preheating...(about 3 mins)");
  preheatCountdown(); // Wait for the end of the module warm-up
  Serial.println("End of module preheating.");
  Serial.println();
}
```

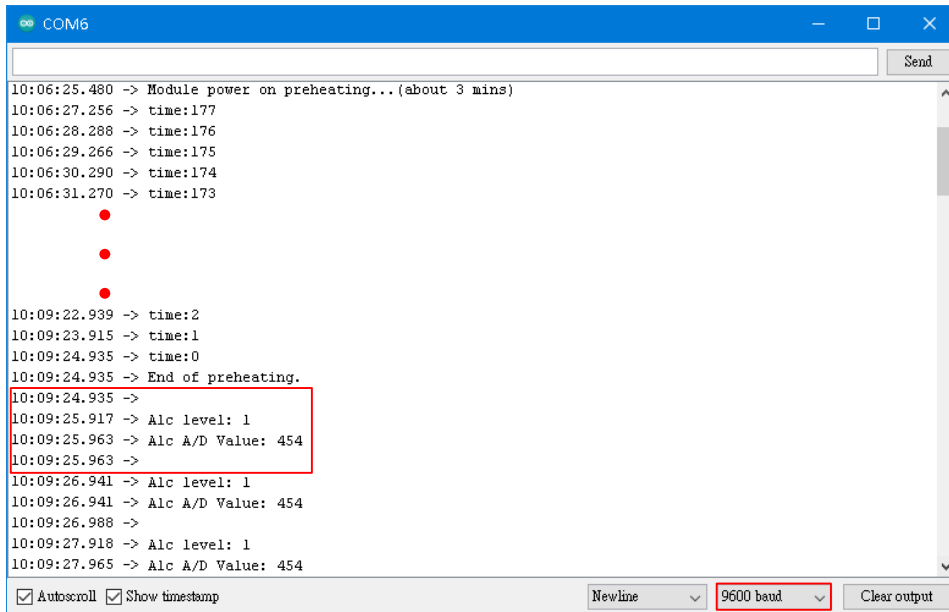
- b. Receive the module auto-transmit data

```
void loop()
{
  if (Alc.isInfoAvailable() == true) // Poll whether the module
  // transmitting data is received
  {
    Alc.readInfoPackage(moduleInfo); // Read the module transmitting
  // data to moduleInfo[]
    printInfo(); // Print partial information of the module
  }
}
```

c. According to the received data to print partial information.

```
void printInfo()
{
    /* Print alcohol concentration level */
    Serial.print("Alcohol level: ");
    AlcLevel = moduleInfo[7];
    Serial.println(AlcLevel);
    /* Print current alcohol gas concentration A/D value (12-bit) */
    Serial.print("Alcohol A/D Value: ");
    ADValue = ((uint16_t)moduleInfo[5] << 8) + moduleInfo[6];
    Serial.println(ADValue);
    Serial.println();
}
```

3. Open the serial monitor and select the baud rate to be 9600. The serial monitor will display as follows.



```
COM6
10:06:25.480 -> Module power on preheating...(about 3 mins)
10:06:27.256 -> time:177
10:06:28.288 -> time:176
10:06:29.266 -> time:175
10:06:30.290 -> time:174
10:06:31.270 -> time:173
.
.
.
10:09:22.939 -> time:2
10:09:23.915 -> time:1
10:09:24.935 -> time:0
10:09:24.935 -> End of preheating.
10:09:24.935 ->
10:09:25.917 -> Alc level: 1
10:09:25.963 -> Alc A/D Value: 454
10:09:25.963 ->
10:09:26.941 -> Alc level: 1
10:09:26.941 -> Alc A/D Value: 454
10:09:26.988 ->
10:09:27.918 -> Alc level: 1
10:09:27.965 -> Alc A/D Value: 454
```

Autoscroll Show timestamp Newline 9600 baud Clear output

Copyright© 2024 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.