



数字式气压传感器

BM62S2201-1 Arduino Library 说明

版本: V1.00 日期: 2023-04-10

www.bestmodulescorp.com

目录

简介	3
Arduino Lib 函数	3
Arduino Lib 下载及安装	5
Arduino 范例	6
范例 1: readTemperatureAndPressureWithIIC	6
范例 2: readTemperatureAndPressureWithUART	8

简介

BM62S2201-1 是倍创推出的数字式气压传感器，使用 UART 或 I²C 通信方式。本文档对 BM62S2201-1 的 Arduino Lib 函数、Arduino Lib 安装方式进行说明；范例使用 BME21M621 模块，演示了读取气压值等功能。

适用型号：

型号	说明
BM62S2201-1	数字式气压传感器
BME21M621	板载 BM62S2201-1 传感器

Arduino Lib 函数

Arduino Lib 名称：BM62S2201-1		Lib 版本：V1.0.1
构造函数 & 初始化		
1	BM62S2201_1(TwoWire *theWire=&Wire)	
	描述	构造函数，选择 I ² C 通信方式
	参数	*theWire: I ² C 通信接口选择
	返回值	—
	备注	—
2	BM62S2201_1(HardwareSerial *theSerial)	
	描述	构造函数，选择硬件 UART 通信方式
	参数	*theSerial: 硬件 UART 通信接口选择
	返回值	—
	备注	—
3	BM62S2201_1(uint8_t rxPin, uint8_t txPin)	
	描述	构造函数，选择软件 UART 通信方式
	参数	rxPin: RX 引脚，连接 BM62S2201-1 的 TX 引脚或 BME21M621 的 TX 引脚 txPin: TX 引脚，连接 BM62S2201-1 的 RX 引脚或 BME21M621 的 RX 引脚
	返回值	—
	备注	—
4	void begin(uint8_t addr=BM62S2201_1_ADDR)	
	描述	模块初始化
	参数	addr: 从机地址，默认 0x50
	返回值	void
	备注	—
功能函数		
5	float readTemperature()	
	描述	获取温度值
	参数	—
	返回值	温度，单位 °C
	备注	—

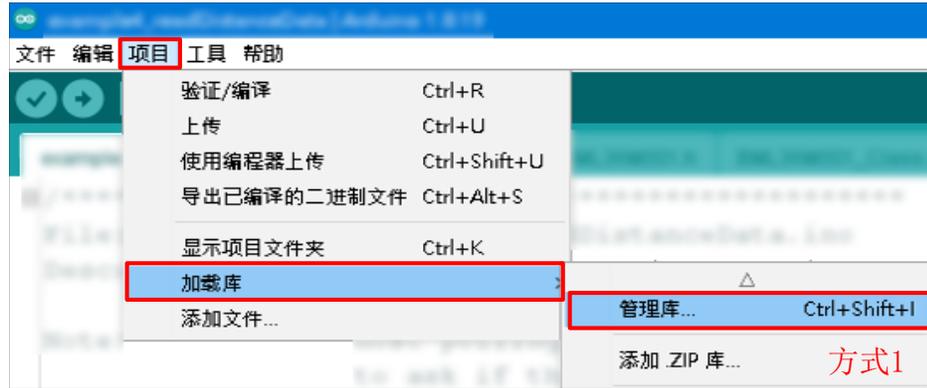
6	uint16_t readPressure()	
	描述	获取压力值
	参数	—
	返回值	压力值, 单位 mpsi
	备注	—
7	uint8_t getDevID(uint8_t devIDbuff[])	
	描述	获取传感器 ID
	参数	devIDbuff[]: 用于存储 ID (8bytes)
	返回值	执行情况 0: 成功 1: 错误 2: 无 ACK
	备注	—
8	uint8_t getI2CAddress()	
	描述	获取传感器 I ² C 地址
	参数	—
	返回值	I ² C 从机地址
	备注	—
9	uint8_t getDeviceStatus()	
	描述	获取传感器状态寄存器值
	参数	—
	返回值	状态寄存器值
	备注	—
10	void enterSleepMode()	
	描述	进入休眠模式
	参数	—
	返回值	void
	备注	—
11	void reset()	
	描述	复位
	参数	—
	返回值	void
	备注	—
12	void restoreDefault()	
	描述	恢复出厂设置
	参数	—
	返回值	void
	备注	CONFIG 和 I2CADR 寄存器恢复为出厂配置
参数配置		
13	void setI2CAddress(uint8_t addr)	
	描述	设置 I ² C 从机地址
	参数	addr: 要设置的 I ² C 从机地址
	返回值	void
	备注	—

Arduino Lib 下载及安装

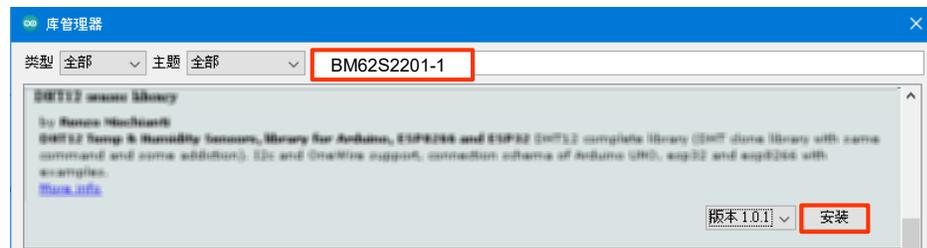
BM62S2201-1 Library: 可参考下面两种方法安装 BM62S2201-1 的 Arduino Library

方式 1: 搜索安装

搜索安装: Arduino IDE → 项目 → 加载库 → 管理库 → 搜索 BM62S2201-1 → 安装



搜索安装流程 1

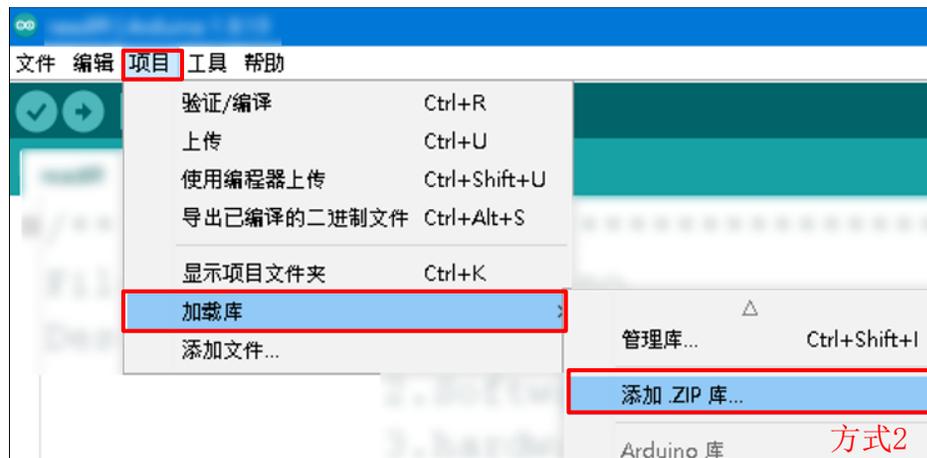


搜索安装流程 2

方式 2: 添加 .ZIP 库, 需提前下载 .ZIP 库

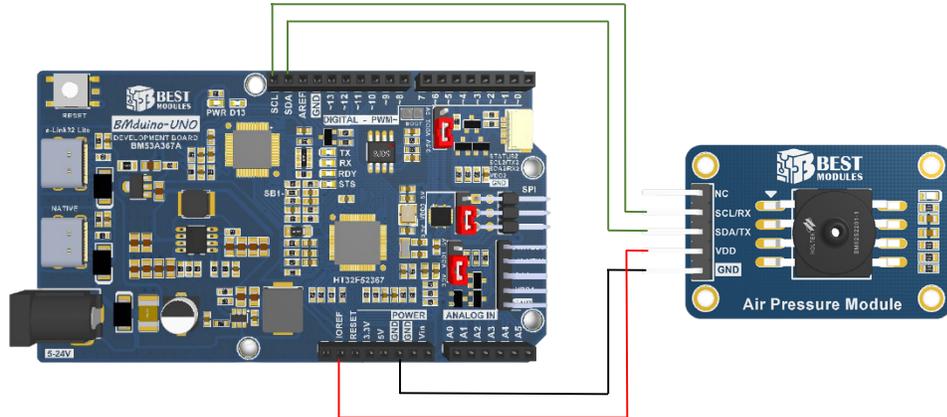
下载方法: 打开倍创官方网站 (<https://www.bestmodulescorp.com/bm62s2201-1.html#tab-product2>) 文件目录下的 Arduino 范例程序 (BM62S2201-1 Library)。

添加 .ZIP 库: Arduino IDE → 项目 → 加载库 → 添加 .ZIP 库 ...



Arduino 范例

范例 1: readTemperatureAndPressureWithIIC



实物连接示意图

范例实现功能：以 I²C 方式与模块通信，获取温度值以及气压值，并在串口监视器上显示。

1. 范例打开：文件 → 示例 → Lib 选择 (BM62S2201-1) → 选择范例 (readTemperatureAndPressureWithIIC)
2. 示例说明：
 - a. 构建 & 初始化对象

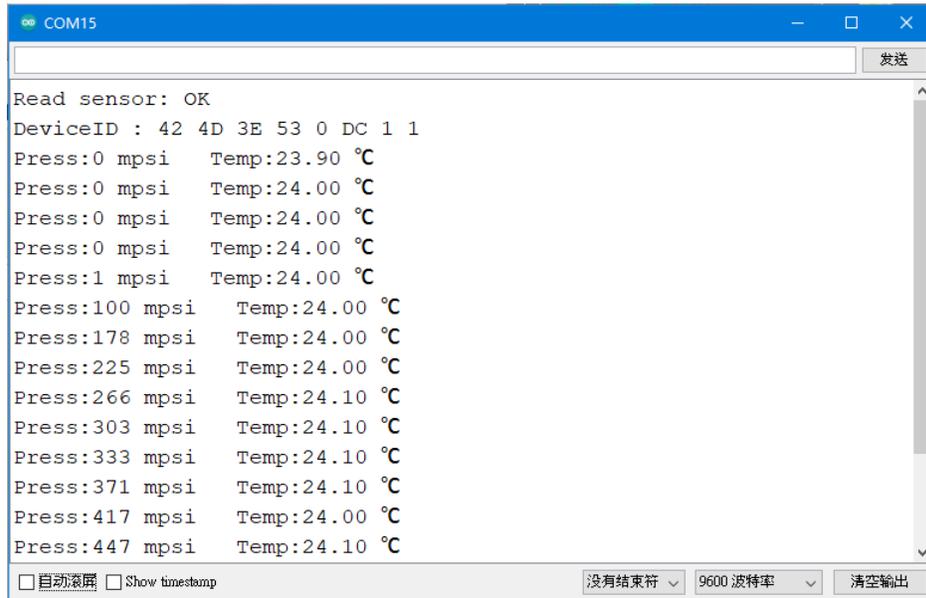
```
#include "BM62S2201-1.h"
BM62S2201_1 Press(&Wire); // 选择 I2C 方式通信
uint8_t devIDbuff[8]; // 创建数组用于存储产品 ID 信息
void setup()
{
  Serial.begin(9600); // 串口监视器初始化，波特率为 9600
  Press.begin(); // 模块初始化
  uint8_t chk = Press.getDevID(devIDbuff); // 获取产品 ID 信息
  if(chk != READ_OK)
  {
    switch(chk)
    {
      case READ_ERROR:Serial.println("Read sensor: CheckSum error"); break;
      case NO_ACK:Serial.println("Read sensor: NO_ACK"); break;
      default:Serial.println("Read sensor: Unknown error"); break;
    }
    Serial.println("Please check connection...");
  }
  else
  {
    Serial.println("Read sensor: OK");
    Serial.print("DeviceID : ");
    for(int i = 0; i<8 ; i++)
    {
      Serial.print(devIDbuff[i],HEX);
    }
  }
}
```

```
        Serial.print(" ");  
    }  
    Serial.println();  
}  
Serial.println();  
delay(1000);  
}
```

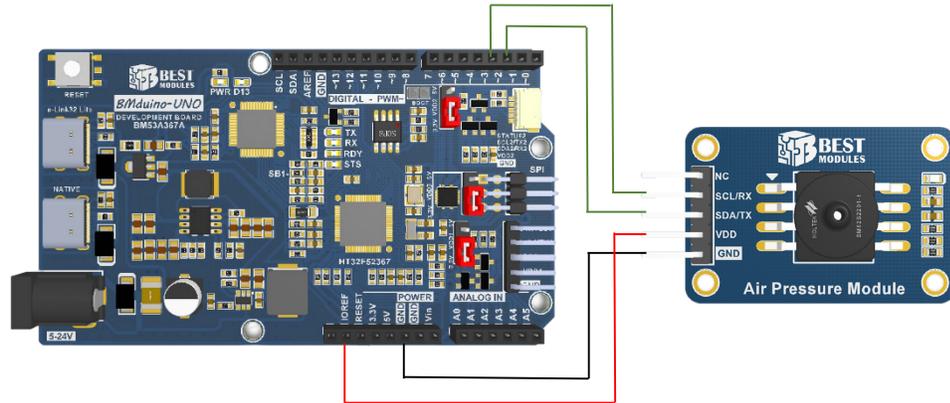
b. 获取温度值、气压值并在串口监视器中显示

```
void loop()  
{  
    Serial.print("Press:");  
    Serial.print(Press.readPressure()); // 获取气压值  
    Serial.print("mpsi");  
    Serial.print("Temp:");  
    Serial.print(Press.readTemperature()); // 获取温度值  
    Serial.println(" °C");  
    delay(1000);  
}
```

3. 打开串口监视器，波特率选择 9600；串口监视器显示数据如下：



范例 2: readTemperatureAndPressureWithUART



实物连接示意图

范例实现功能: 以 UART 方式与模块通讯, 获取温度值以及气压值, 并在串口监视器上显示。

1. 范例打开: 文件 → 示例 → Lib 选择 (BM62S2201-1) → 选择范例 (readTemperatureAndPressureWithUART)
2. 示例说明:
 - a. 构建 & 初始化对象

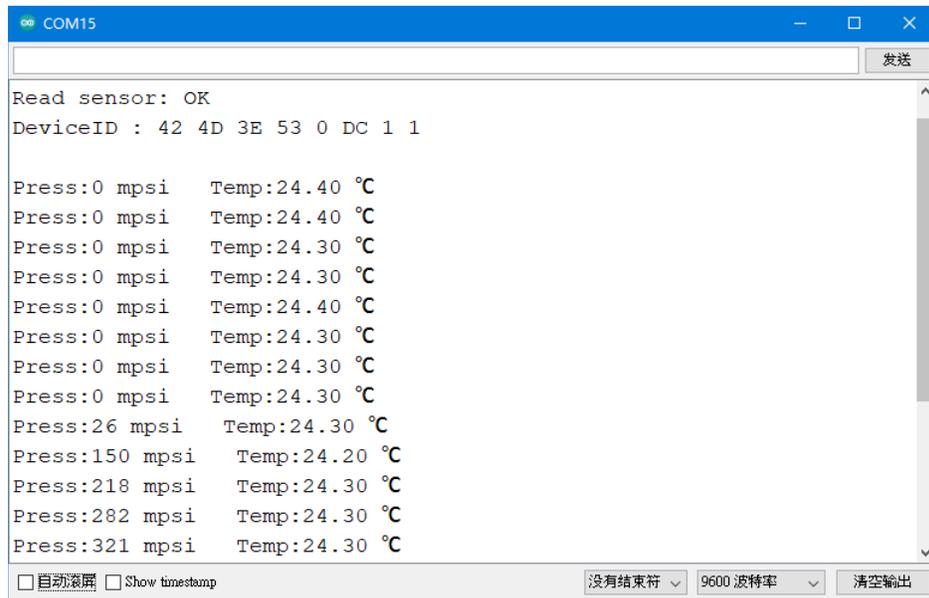
```
#include "BM62S2201-1.h"
#define RX_PIN 2
#define TX_PIN 3
BM62S2201_1 Press(RX_PIN,TX_PIN); // 选择软件 UART 方式通信
uint8_t devIDbuff[8]; // 创建数组用于存储产品 ID 信息
void setup()
{
  Serial.begin(9600); // 串口监视器初始化, 波特率为 9600
  Press.begin(); // 模块初始化
  uint8_t chk = Press.getDevID(devIDbuff); // 获取产品 ID 信息
  if(chk != READ_OK)
  {
    switch(chk)
    {
      case READ_ERROR:Serial.println("Read sensor: CheckSum error");
      break;
      case NO_ACK:Serial.println("Read sensor: NO_ACK"); break;
      default:Serial.println("Read sensor: Unknown error"); break;
    }
    Serial.println("Please check connection...");
  }
  else
  {
    Serial.println("Read sensor: OK");
    Serial.print("DeviceID : ");
    for(int i = 0; i<8 ; i++)
    {
      Serial.print(devIDbuff[i],HEX);
      Serial.print(" ");
    }
  }
}
```

```
Serial.println();  
}  
Serial.println();  
delay(1000);  
}
```

b. 获取温度值、气压值并在串口监视器中显示

```
void loop()  
{  
  Serial.print("Press:");  
  Serial.print(Press.readPressure()); // 获取气压值  
  Serial.print(" mpsi ");  
  Serial.print("Temp:");  
  Serial.print(Press.readTemperature()); // 获取温度值  
  Serial.println(" °C");  
  delay(1000);  
}
```

3. 打开串口监视器，波特率选择 9600；串口监视器显示数据如下：



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

本文件出版时倍创已针对所载信息为合理注意，但不保证信息准确无误。文中提到的信息仅是提供作为参考，且可能被更新取代。倍创不承担任何明示、默示或法定的，包括但不限于适合商品化、令人满意的质量、规格、特性、功能与特定用途、不侵害第三方权利等保证责任。倍创就文中提到的信息及该信息之应用，不承担任何法律责任。此外，倍创并不推荐将倍创的产品使用在会由于故障或其他原因而可能会对人身安全造成危害的地方。倍创特此声明，不授权将产品使用于救生、维生或安全关键零部件。在救生 / 维生或安全应用中使用倍创产品的风险完全由买方承担，如因该等使用导致倍创遭受损害、索赔、诉讼或产生费用，买方同意出面进行辩护、赔偿并使倍创免受损害。倍创 (及其授权方，如适用) 拥有本文件所提供信息 (包括但不限于内容、数据、示例、材料、图形、商标) 的知识产权，且该信息受著作权法和其他知识产权法的保护。倍创在此并未明示或暗示授予任何知识产权。倍创拥有不事先通知而修改本文件所载信息的权利。如欲取得最新的信息，请与我们联系。