



PIR 探测数字模块

BM22S4221-1 Arduino Library V1.0.2 说明

版本: V1.10 日期: 2025-09-11

www.bestmodulescorp.com

目录

简介	3
Arduino Lib 函数	3
Arduino Lib 下载及安装	7
Arduino 范例	8
Example1: readStatusPin	8

简介

BM22S4221-1 是倍创推出的 PIR 探测数字模块。使用 UART 通信方式。本文档对 BM22S4221-1 的 Arduino Lib 函数、Arduino Lib 安装方式进行说明；范例使用 BMA46M422/BMA46M422A 模块，演示了报警检测等功能。

适用型号：

型号	说明
BM22S4221-1	PIR 探测数字模块
BMA46M422	板载 BM22S4221-1 模块
BMA46M422A	板载 BM22S4221-1 模块

Arduino Lib 函数

Arduino Lib 名称: BM22S4221-1		Lib 版本: V1.0.2
构造函数 & 初始化		
1	BM22S4221_1(uint8_t statusPin, HardwareSerial* theSerial)	
	描述	构造函数硬件 UART
	参数	statusPin: STATUS 引脚, 连接 BM22S4221-1 的 STATUS 引脚或 BMA46M422/BMA46M422A 的 STA 引脚 * theSerial: 硬件 UART 通信接口选择
	返回值	—
	备注	—
2	BM22S4221_1(uint8_t statusPin, uint8_t rxPin, uint8_t txPin)	
	描述	构造函数软件 UART
	参数	statusPin: STATUS 引脚, 连接 BM22S4221-1 的 STATUS 引脚或 BMA46M422/BMA46M422A 的 STA 引脚 rxPin: RX 引脚, 连接模块的 TX 引脚 txPin: TX 引脚, 连接模块的 RX 引脚
	返回值	—
	备注	—
3	void begin()	
	描述	模块初始化
	参数	—
	返回值	—
	备注	波特率固定为 9600

功能函数		
4	uint8_t getSTATUS()	
	描述	获取 STATUS 引脚电平
	参数	—
	返回值	STATUS 引脚电平： 0: 低电平 1: 高电平
	备注	—
5	uint8_t requestInfoPackage(uint8_t buff[])	
	描述	获取设备当前所有数据
	参数	buff[]: 存放读取到的数据 (25-byte)
	返回值	执行情况： 0: 成功 1: 失败
	备注	数据内容具体参考 BM22S4221-1 规格书
6	uint8_t getFWVer()	
	描述	获取模块软件版本号
	参数	—
	返回值	模块软件版本 FWVer
	备注	—
7	uint8_t getProDate(uint8_t buff[])	
	描述	获取模块生产日期
	参数	buff[]: 用于存储生产日期, buff[0]/buff[1]/buff[2]: 年/月/日
	返回值	执行情况： 0: 成功 1: 失败
	备注	如: 0x21, 0x06, 0x2 表示 2021/6/2
8	bool isAotuTx()	
	描述	获取 TX 自动输出是否使能
	参数	—
	返回值	判断结果： true: 是 false: 不是
	备注	—
9	uint8_t getStatusPinActiveMode()	
	描述	获取报警输出电平
	参数	—
	返回值	报警电平： 0x80: 高电平 0x00: 低电平
	备注	—

10	uint8_t getVBG()	
	描述	获取当前内部 V _{BG} 电压 A/D 值
	参数	—
	返回值	V _{BG} 电压 A/D 值
	备注	—
11	bool isInfoAvailable()	
	描述	判断是不是模块自动输出的 25 字节数据包
	参数	—
	返回值	判断结果： true: 是 false: 不是
	备注	—
12	uint8_t readInfoPackage(uint8_t array[])	
	描述	读取 TX 引脚自动输出的 25 字节数据
	参数	array[]: 存放读取到的数据
	返回值	执行情况： 0: 成功 1: 失败
	备注	被动读取数据，数据内容具体参考 BM22S4221-1 规格书
13	uint8_t resetModule()	
	描述	复位模块
	参数	—
	返回值	执行情况： 0: 成功 1: 失败
	备注	—
14	uint8_t restoreDefault()	
	描述	设置模块恢复出厂设置
	参数	—
	返回值	执行情况： 0: 成功 1: 失败
	备注	—
参数配置		
15	uint8_t setAutoTx(uint8_t state)	
	描述	设置自动输出功能是否使能
	参数	state: 自动输出功能是否使能 0x08 (AUTO): 使能 0x00 (PASSIVE): 除能
	返回值	执行情况： 0: 成功 1: 失败
	备注	—

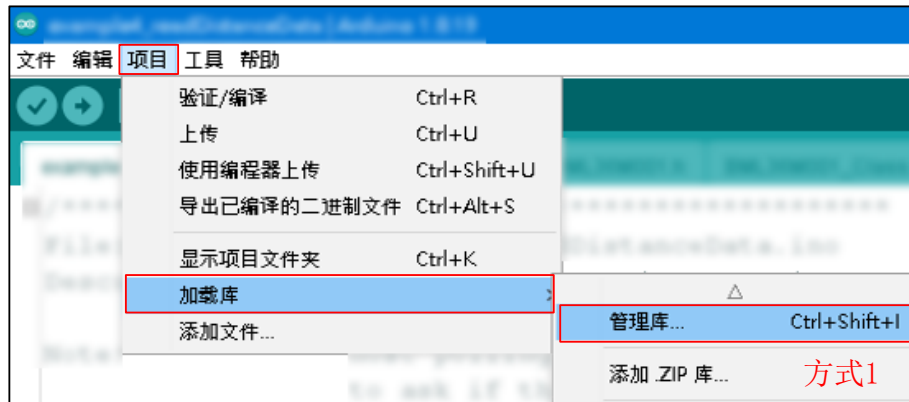
16	uint8_t setStatusPinActiveMode(uint8_t state)	
	描述	设置报警输出电平
	参数	state: 报警输出电平 0x08 (HIGH_LEVEL): 高电平 0x00 (LOW_LEVEL): 低电平
	返回值	执行情况: 0: 成功 1: 失败
	备注	—
17	uint8_t setOpaGain(uint8_t value)	
	描述	设置内部 OPA 增益
	参数	value: OPA 增益参数, 范围 0~31
	返回值	执行情况: 0: 成功 1: 失败
	备注	OPA gain=128+value×8
18	uint8_t setAlarmThreshold(uint8_t Threshold)	
	描述	设置 PIR 报警阈值
	参数	Threshold: PIR 报警阈值, 范围 15~120
	返回值	执行情况: 0: 成功 1: 失败
	备注	Threshold = 报警触发值 - PIR 中心电压值的绝对值
19	uint8_t setAlarmDetectDelay(uint8_t time)	
	描述	设置触发报警后再次检测的延时时间
	参数	time: 延时时间, 单位 s
	返回值	执行情况: 0: 成功 1: 失败
	备注	模块出厂默认 3s
20	uint8_t setAlarmOutputTime(uint8_t time)	
	描述	设置报警信号 STATUS 引脚持续输出时间
	参数	time: 持续输出时间, 单位 s
	返回值	执行情况: 0: 成功 1: 失败
	备注	模块出厂默认 3s
21	uint8_t setPreheaTime(uint8_t time)	
	描述	设置预热时间
	参数	time: 预热时间, 范围 30~127s
	返回值	执行情况: 0: 成功 1: 失败
	备注	模块出厂默认 30s

Arduino Lib 下载及安装

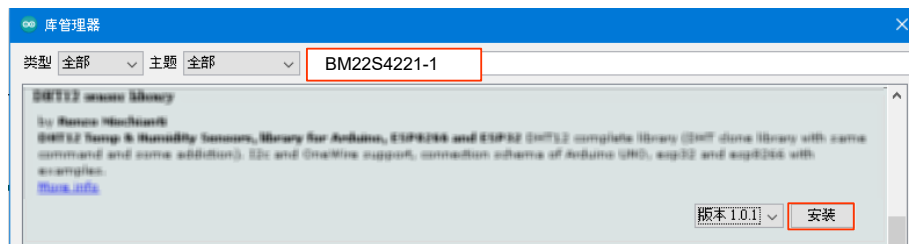
BM22S4221-1 Library: 可参考下面两种方法安装 BM22S4221-1 的 Arduino Library

方式 1: 搜索安装

搜索安装: Arduino IDE → 项目 → 加载库 → 管理库 → 搜索 BM22S4221-1 → 安装



搜索安装流程 1

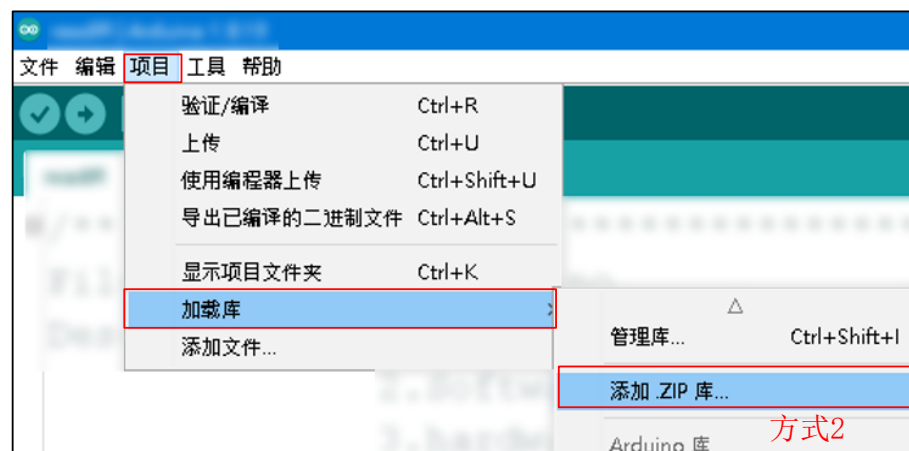


搜索安装流程 2

方式 2: 添加 .ZIP 库, 需提前下载 .ZIP 库

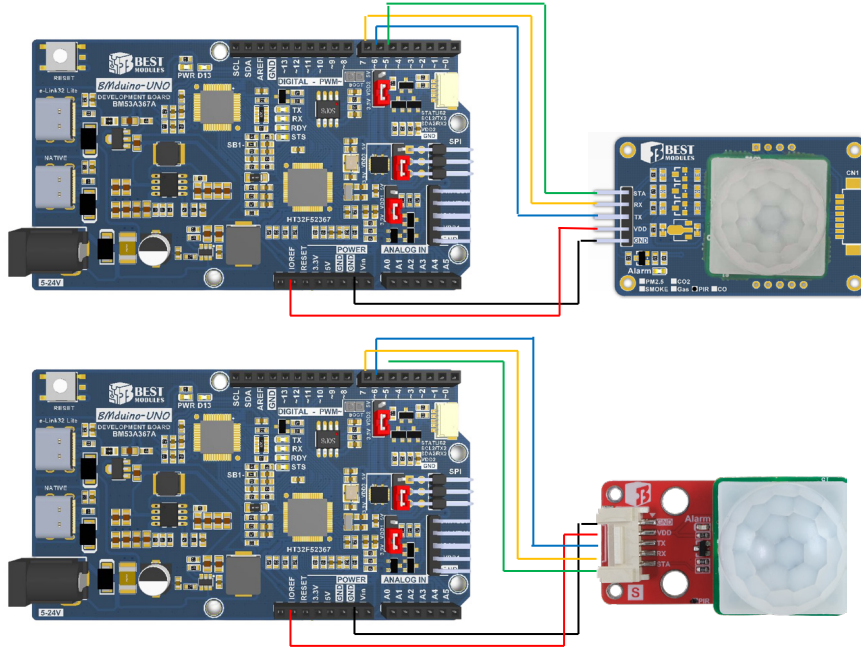
下载方法: 打开倍创官方网站 (<https://www.bestmodulescorp.com/bm22s4221-1.html>) 文件目录下的 Arduino 范例程序 (BM22S4221-1 Library)。

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



Arduino 范例

Example1: readStatusPin



实物连接示意图

范例实现功能：模块正常状态下 LED13 缓慢闪烁，检测到模块报警则 LED13 常亮。模块状态变化会打印对应信息在串口监视器。

1. 范例打开：Arduino IDE → 文件 → 示例 → Lib 选择 (BM22S4221-1) → 选择范例 (readStatusPin)

2. 示例说明：

a. 构建对象

```
#include "BM22S4221-1.h"
uint8_t STATUS=5;           // 更改 STATUS PIN
uint8_t flag=3;
BM22S4221_1 PIR(STATUS,6,7); // Software Serial
```

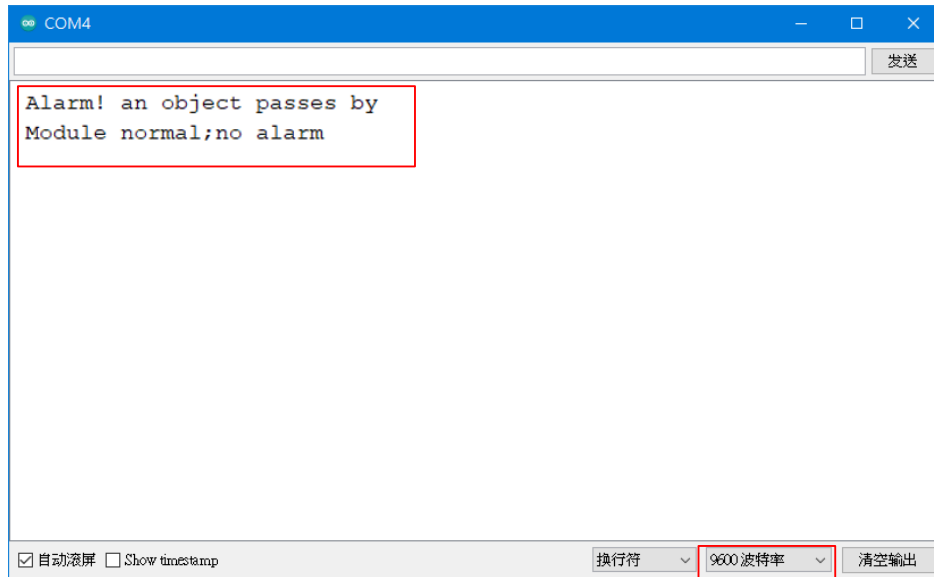
b. 模块初始化配置 I/O

```
void setup()
{
  Serial.begin(9600);
  PIR.begin();
  pinMode(STATUS, INPUT);
  pinMode(13, OUTPUT);
}
```

c. 循环检测模块状态，正常 / 报警 LED 闪亮状态不同

```
if (PIR.getSTATUS() == HIGH && flag != 1)
{
  Serial.println("Alarm! an object passes by"); // 触发报警状态
  flag = 1;
}
if (flag != 0 && PIR.getSTATUS() == LOW)
{
  flag = 0;
  Serial.println("Module normal;no alarm"); // 模块当前正常
}
switch (flag)
{
  case 0: // 正常
    digitalWrite(13, 1);
    delay(100);
    digitalWrite(13, LOW);
    delay(900);
    break;
  case 1: // 报警
    digitalWrite(13, 1);
    delay(1000);
    break;
}
```

3. 打开串口监视器，波特率选择 9600；串口监视器显示如下：



Copyright® 2025 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。