

# UPS HAT

---

## 说明

### 产品概述

我是块适用于 Raspberry Pi 的不间断电源(UPS)模块,支持同时充电和放电,可提供稳定 5V 电压输出,可通过 I2C 接口监测电池的电压、电流、功率等参数。

### 特点

- 基于 Raspberry Pi 40pin GPIO 接口,适用于 Raspberry Pi 系列主板
- 可通过 I2C 接口通信,测量电池电压、电流、功率和剩余电量等参数,实时检测模块工作状态
- 板载电池保护电路,防过充、防过放、防过流、防短路和防反接,均衡充电,工作稳定更安全
- 板载 5V 稳压芯片,可达 2.5A 连续输出电流
- 板载 USB 接口,可输出 5V 电源,方便给其他主控板供电
- 板载警示指示灯,方便查看电池是否接反
- 提供完善的配套资料手册

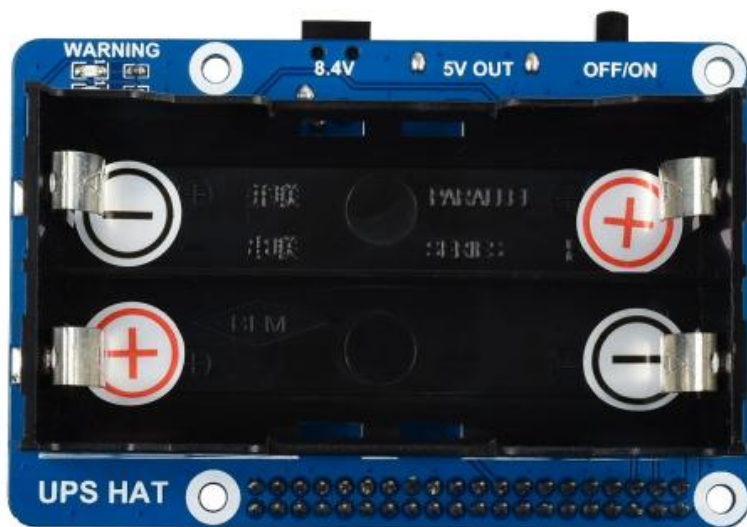
## 产品参数

- 输出电压：5V
- 充电电源：8.4V 2A
- 控制接口：I2C
- 电池规格：18650 锂电池
- 电流容量：2 节电池共 5200mAh
- 产品尺寸：56mm × 85mm
- 通孔直径：3.0mm

## 接口说明

### 硬件说明

---



8.4V 接口为充电接口，使用 8.4V 2A 电源充电。5V OUT 为 USB 母口，可输出 5V 电源。OFF/ON 丝印为电源开关丝印。

WARNING LED 灯为电池接反指示灯，LED 亮表示电池接反。

**注意 1：首次接上电池需要注意 LED 灯是否亮，如果 LED 亮则表示对应的电池接反了。电池接反情况下禁止充电。**

**注意 2：首次接上电池有可能会没有电源输出，接上 UPS 电源充一下电激活电池保护芯片才会有电源输出。**

**注意 3：要使用产品配置的电源充电，使用其他电源有可能输出电源纹波太大不稳定导致产品损坏。**

**注意 4：电池正极绝缘垫如果弄掉坏了建议不要继续使用容易导致短路起火，拆电池时切忌用螺丝刀撬电池正极容易导致短路要撬电池负极。**

## 锂电池使用注意事项

---

- 锂聚合物和锂离子电池很不稳定。如果充电或使用不当，可能会导致火灾，人身伤害和财产损失。
- 对电池组充电和放电时，切勿极性连接错误。请勿选用其他劣质的充电器或充电板对锂电池进行充电。
- 请勿混用和使用旧电池和新电池，避免使用其他品牌的电池。
- 如需另购锂电池产品，须确保电池参数与锂电池扩展板兼容，推荐选择正规的电池厂家，并需要自行老化测试确保锂电池能够稳定安全工作。
- 锂电池具有循环寿命，请在达到使用寿命或使用两年时（以先到者为准），用新电池更换旧电池。
- 请妥善放置电池类产品，远离易燃易爆类物品，远离儿童，避免因保管不慎，引发安全事故。

## 树莓派例程

### 开启 I2C 接口

- 打开树莓派终端，输入以下指令进入配置界面

```
sudo raspi-config
```

选择 Interfacing Options -> I2C ->yes 启动 i2C 内核驱动

```
Raspberry Pi Software Configuration Tool (raspi-config)

1 Change User Password Change password for the current user
2 Network Options      Configure network settings
3 Boot Options         Configure options for start-up
4 Localisation Options Set up language and regional settings to match your location
5 Interfacing Options  Configure connections to peripherals
6 Overclock           Configure overclocking for your Pi
7 Advanced Options    Configure advanced settings
8 Update              Update this tool to the latest version
9 About raspi-config  Information about this configuration tool

<Select>                                <Finish>
```

```
Raspberry Pi Software Configuration Tool (raspi-config)

P1 Camera      Enable/Disable connection to the Raspberry Pi Camera
P2 SSH         Enable/Disable remote command line access to your Pi using SSH
P3 VNC         Enable/Disable graphical remote access to your Pi using RealVNC
P4 SPI         Enable/Disable automatic loading of SPI kernel module
P5 I2C         Enable/Disable automatic loading of I2C kernel module
P6 Serial      Enable/Disable shell and kernel messages on the serial connection
P7 1-Wire      Enable/Disable one-wire interface
P8 Remote GPIO Enable/Disable remote access to GPIO pins

<Select>                                <Back>
```

```
Would you like the ARM I2C interface to be enabled?

<Yes>                                <No>
```

然后重启树莓派：

```
sudo reboot
```

在使用树莓派演示该例程时，只需要将模块插到树莓派排针中即可，或者通过杜邦线接到树莓派的 I2C 接口，VCC 接 3.3V

输入如下命令运行程序

```
sudo apt-get install p7zip
```

```
wget http://www.waveshare.net/w/upload/d/d9/UPS_HAT.7z
```

```
7z x UPS_HAT.7z -r -o./
```

```
cd UPS_HAT
```

```
python3 INA219.py
```

程序运行后会输出电池电压，电流，功率以及电池剩余电量百分比等数值。

```
Load Voltage:  7.368 V
Current:       -0.327700 A
Power:         2.416 W
Percent:       57.0%
```

**注意：如果电流为负数则表示电池输出电流(可认为树莓派负载电流)，电流为正数则表示电池充电电流。**

## 资料

文档

- [原理图](#)

程序

- [示例程序](#)

## 相关资料

- [INA219 数据手册](#)
- [HY2120 数据手册](#)
- [HY2213 数据手册](#)

## FAQ

问题： 高度多少？

- 加树莓派总高度 42.05mm。

问题： 电源适配器指示灯？

- 红灯充电，绿灯充满。

问题： 为什么开关打开后充电器由绿灯变成了红灯？

- 只要连接有负载就会是一直红灯的，不会变绿，因为充电器是一直处于供电的状态，不断有电流输出。

问题：为什么打印的电流是负数？

- 在给树莓派供电，电流是流出的。

问题：在没有电的情况下，充满电大概需要多久？

---

UPS HAT 大概是两三个小时。

问题：UPS 接到树莓派后用的是哪个地址，怎么查？

---

可以用以下指令查地址，这里用到的地址是 0x42:

```
sudo apt-get install i2c-tools
```

```
sudo i2cdetect -y 1
```

```
pi@raspberrypi:~$ sudo i2cdetect -y 1
   0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
10:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
20:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
30:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
40:  --  --  42  --  --  --  --  --  --  --  --  --  --  --  --
50:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
60:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
70:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
pi@raspberrypi:~$
```

