Power Management HAT 用户手册

产品概述

我是一款树莓派电源管理控制板，能控制树莓派开机、关机(软件关机)、定时开机、定时关机、检测树莓派工作状态、测量工作电流，测量外接锂电池电压。

板载 Arduino(ATmega328P)和 RTC 时钟芯片(PCF8523)，可通过 Arduino 编程 DIY 树莓派的电源管理与控制。

适用于电池供电状态下，让树莓派“低功耗”运行。

特点

- 板载 MCU(ATmega328P)，支持 Arduino 编程。
- 板载 RTC 时钟芯片(PCF8523)，可提供精准时间控制。
- 板载 CP2102 芯片，可实现对 Arduino 一键下载，同时支持查看串口数据。
- 板载自定义按键，可作为电源键使用，实现树莓派的开机，软件关机，强制关机。
- 提供电源防反接，放倒灌机制，充分保证电路的安全。
- 可通过 USB 接口供电(5V),也可通过外接锂电池供电(2S~4S，7.4V~14.8V)，或外接 DC 供电(7.0~28V)。
- 配套电池及电池盒，可提供 2S，7.4V 电源，可提供容量达 2200 * 2mAh。
- 可对外接电池电压进行检测，对工作电流进行检测，可编程即时切断电源以确保锂电池的工作寿命。
- 提供完善的配套资料手册，提供丰富的电源管理与控制的 Arduino 例程。

参数

- 通信接口：UART, GPIO(自定义协议)
- 主控芯片：ATmega328P-AU
- 通信接口：UART + GPIO

版本：V1.0.0，日期：2019 年 05 月 10 日

- 通信波特率：默认 115200bps (可程序设置其他波特率)
- 供电方式：USB 接口 或 PH2.0 接口
- USB 接口电压：5V
- PH2.0 接口电压：DC 7~28V (稳压电源或锂电池)
- 内置电路：电源防反接、防倒灌电流、电流检测、电压检测
- 产品尺寸：56.5mm x 65mm
- 固定孔直径：3.0mm

工作原理

控制树莓派关机工作原理：

Arduino(ATmega328P)通过 2 个 GPIO 连接树莓派的两个 GPIO，自定义如下通信协议：

树莓派端		
树莓派 GPIO (BCM)	连接 Arduino GPIO	逻辑功能
GPIO20	PC3	输入，当检测到高电平时，执行关机命令，开机状态下应当检测到低电平。
GPIO21	PD7	输出，当输出高电平时，表示树莓派已开机，当关机后，将输出低电平。

Arduino 端		
Arduino GPIO	连接 树莓派 GPIO (BCM)	逻辑功能
PC3	GPIO20	输出，当输出高电平时，树莓派检测到后将执行关机命令，开机状态下应当输出低电平。
PD7	GPIO21	输入，当检测到高电平时，表示树莓派已开机，关机状态下应当检测到低电平。

树莓派端实现该协议：

- 开机：在开机后将 GPIO21 置为高电平。
- 关机：不断检测 GPIO20 的输入电平，当输入为高电平时，执行关机指令，关机后 GPIO21 将自动置为低电平。

Arduino(ATmega328P)端实现该协议：

- 开机：将 PC3 置为低电平，接通树莓派电源，树莓派开机，检测 PD7 输入电平，当输入为高电平时，表示树莓派已成功开机。
- 关机：将 PC3 置为高电平，树莓派检测到后将执行关机指令，检测 PD7 输入电平，当输入为低电平时，表明树莓派已成功关机，切断树莓派电源。

基础准备工作

树莓派端准备工作

关机协议配置：

打开树莓派终端窗口，分别执行以下命令，即可配置关机协议：

- 下载 shell 脚本

执行

```
sudo wget https://raw.githubusercontent.com/waveshare/Power-Management-HAT/master/Power-Management-HAT-Setup.sh
```

执行结果如下图所示，表示下载完成：

```
pi@raspberrypi:~/Power-Management-HAT/shell $ sudo wget https://raw.githubusercontent.com/waveshare/Power-Management-HAT/master/Power-Managem
ent-HAT-Setup.sh
--2019-08-02 11:42:00--  https://raw.githubusercontent.com/waveshare/Power-Management-HAT/master/Power-Management-HAT-Setup.sh
Resolving raw.githubusercontent.com (raw.githubusercontent.com)... 151.101.108.133
Connecting to raw.githubusercontent.com (raw.githubusercontent.com)|151.101.108.133|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 2737 (2.7K) [text/plain]
Saving to: 'Power-Management-HAT-Setup.sh'

Power-Management-HAT-Setup.sh       100%[===================================================================>]   2.67K  --.-KB/s    in 0s

2019-08-02 11:42:11 (5.76 MB/s) - 'Power-Management-HAT-Setup.sh' saved [2737/2737]
```

- 给 shell 脚本执行权限

执行

```
sudo chmod a+x Power-Management-HAT-Setup.sh
```

执行结果如下图所示，表示成功添加执行权限：

```
pi@raspberrypi:~/Power-Management-HAT/shell $ sudo chmod a+x Power-Management-HAT-Setup.sh
pi@raspberrypi:~/Power-Management-HAT/shell $ ls
Power-Management-HAT-Setup.sh
```

- 执行 shell 脚本

执行

```
sudo ./Power-Management-HAT-Setup.sh
```

执行结果如下图所示，当提示 Would you like to reboot now? y/n 时，输入 y，表示重启树莓派(必须重启树莓派该操作才有效，如果树莓派未重启，必须手动重启树莓派后方生效)。

```
pi@raspberrypi:~/Power-Management-HAT/shell $ sudo ./Power-Management-HAT-Setup.sh
==============================================================================
|                                                                            |
|                          Waveshare Electronic Team                         |
|             Power-Management-HAT Installation Script - Raspbian only              |
|                                                                            |
==============================================================================
Setting up the shutdown...
--2019-08-02 11:43:05--  https://raw.githubusercontent.com/waveshare/Power-Management-HAT/master/StatusDetection.py
Resolving raw.githubusercontent.com (raw.githubusercontent.com)... 151.101.108.133
Connecting to raw.githubusercontent.com (raw.githubusercontent.com)|151.101.108.133|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 412 [text/plain]
Saving to: 'StatusDetection.py'

StatusDetection.py                  100%[===================================================================>]     412  --.-KB/s    in 0s

2019-08-02 11:43:09 (2.47 MB/s) - 'StatusDetection.py' saved [412/412]

PowerManagementHAT setup complete!
Would you like to reboot now? y/n
```

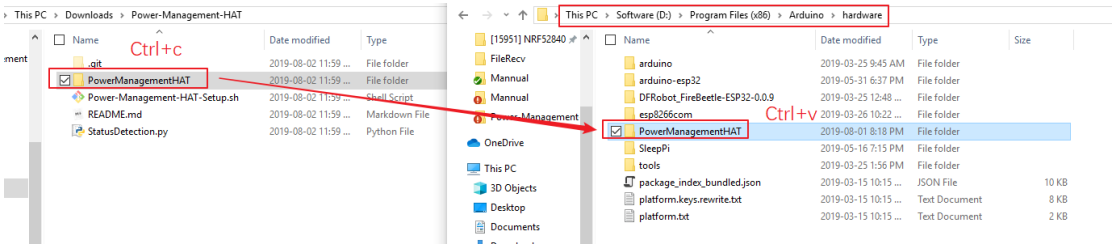
Windows PC 端准备工作

- 安装 Arduino IDE

下载完成后，直接点击下一步即可安装，记住安装路径，安装路径将在下一步中用到。

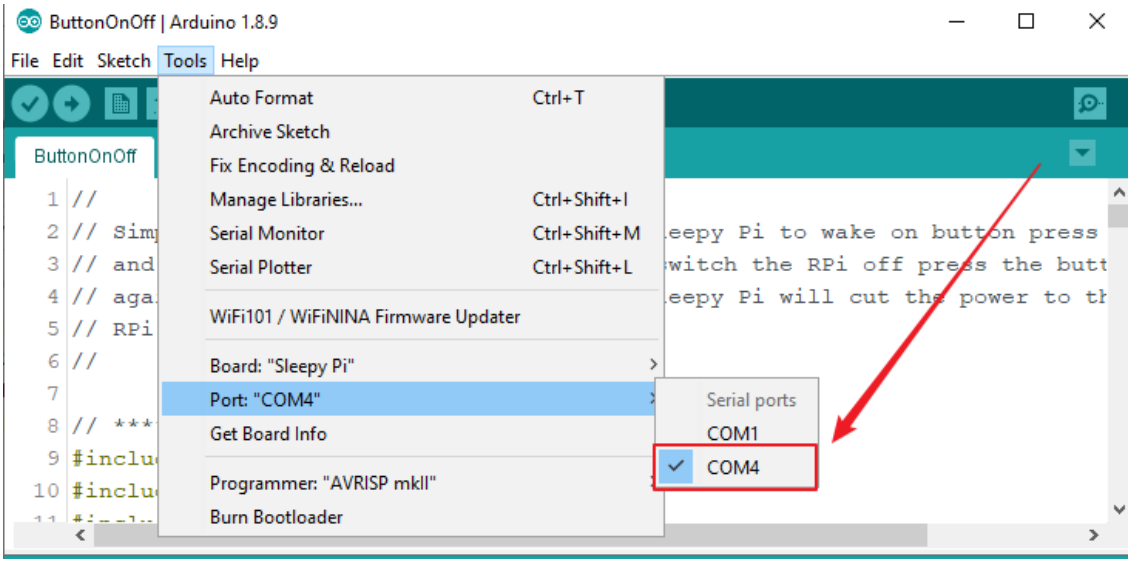
- 安装 Power Management HAT 器件包

器件包可以从 Gihub 下载地址下载，也可以从微雪官网下载，下载完成后，将下载后 Power Management HAT 文件夹（该文件夹还含有子文件夹 Power Management HAT）复制到 Arduino 安装路径下的 hardware 文件夹下，如下图所示，执行完该步骤后则可以使用 Arduino IDE 来开发 Power Management HAT 了。

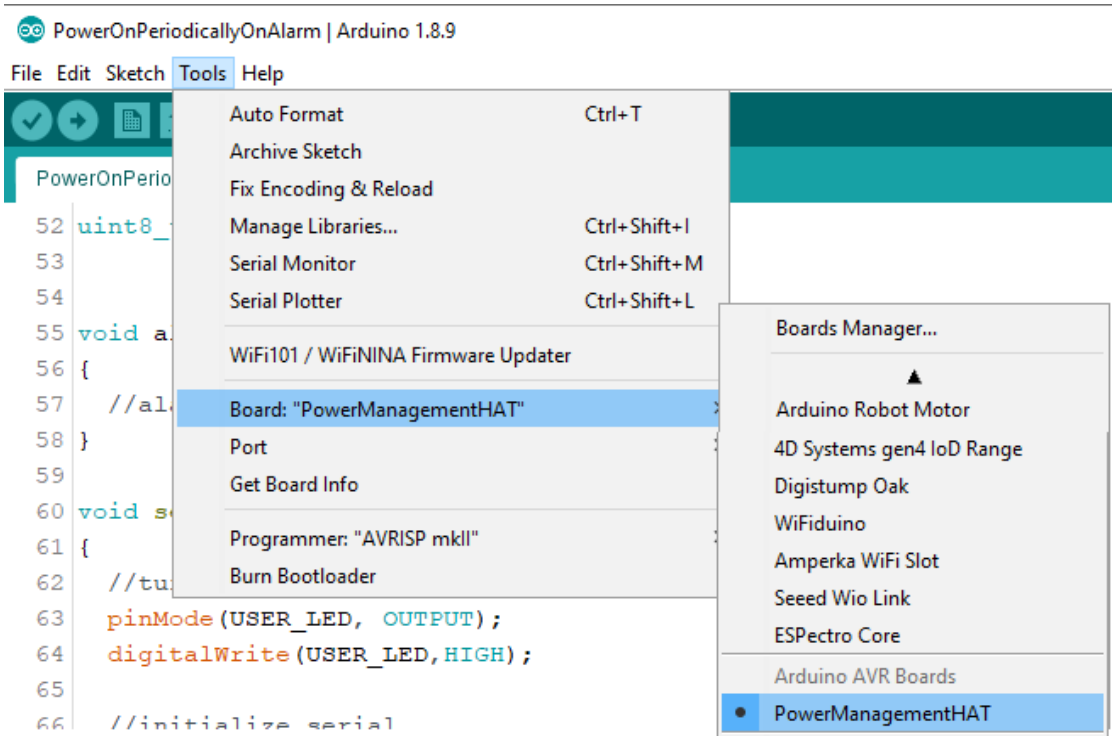


- Arduino IDE 开发环境相关配置

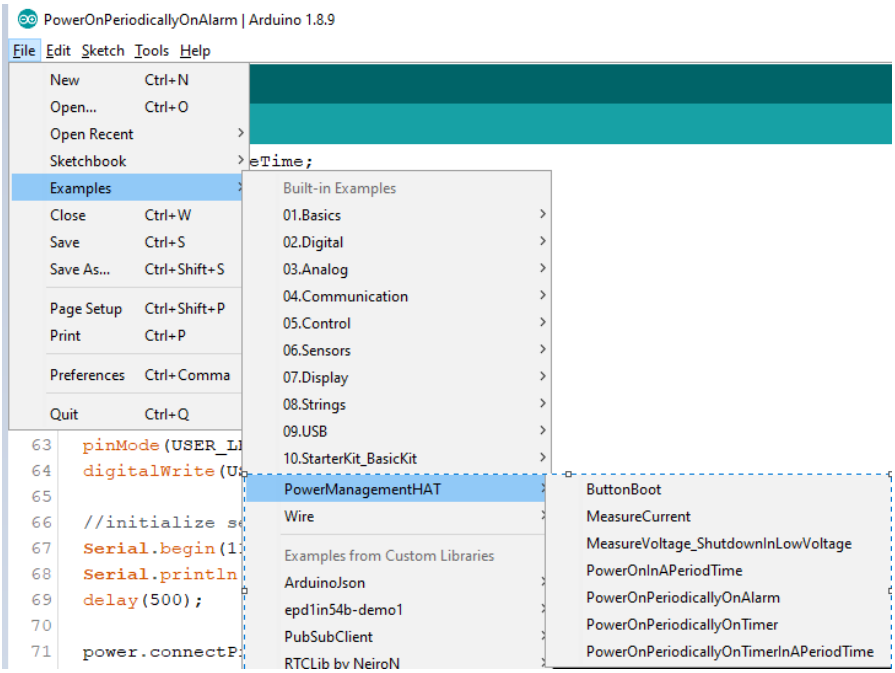
首先，重启 Arduino IDE，将 Power Management Hat 的 USB 连接至电脑，选择相应的串口：



接着，选择 Power Management HAT 器件：



最后，即可看到关于 Power Management HAT 相关的例程如下图所示，后续章节将讲解相关例程的使用：



供电准备

- Power Management HAT 插入到树莓派引脚
- Power Management HAT 将为树莓派供电，树莓派无需再供电
- Power Management HAT 采用以下一种方式供电，若同时供电，将自动选择 USB 供电
    - Power Management HAT 采用 USB 供电

- Power Management HAT 采用配套锂电池盒供电，或另选其它 DC 电源供电(可选 DC 电源电压范围为 7V-28V)

演示

一键开关机

- 完成基础准备工作
- 打开 ButtonBoot 例程，编译，下载
- 若树莓派已关机：
  - 按下电源键，树莓派将开机
- 若树莓派已开机：
  - 长按电源键 2 秒以上，但小于 8 秒，树莓派将先软件关机，软件关机后将切断树莓派电源。
  - 长按电源键 8 秒以上，将直接切断树莓派电源。
- 可调参数如下：

参数	含义	默认值
SHUTDOWN_TIME_MS	长按时间超过该阈值将软件关机。	2000
CUT_POWER_TIME_MS	长按时间超过该阈值将切断电源。	8000
USE_CURRENT_TO_CHECK_PI_STATUS	是否使用工作电流来判断树莓派的工作状态，默认采用软件来判断树莓派的工作状态。	0
CURRENT_THRESHOLD_MA	工作电流小于该阈值将判定为已关机，默认无效。	85

电流检测

- 完成基础准备工作
- 打开 MeasureCurrent 例程，编译，下载
- 若树莓派已关机：
  - 按下电源键，树莓派将开机
- 若树莓派已开机：
  - 长按电源键 2 秒以上，但小于 8 秒，树莓派将先软件关机，软件关机后将切断树莓派电源
  - 长按电源键 8 秒以上，将直接切断树莓派电源
- 打开串口终端，可查看到树莓派开机与关机状态的工作电流
- 可调参数如下：

参数	含义	默认值

SHUTDOWN_TIME_MS	长按时间超过该阈值将软件关机。	2000
CUT_POWER_TIME_MS	长按时间超过该阈值将切断电源。	8000

电压检测+低电压自动关机

- 完成基础准备工作
- 打开 MeasureVoltage_ShutdownInLowVoltage 例程，编译，下载
- 若树莓派已关机：
  - 按下电源键，树莓派将开机
- 若树莓派已开机：
  - 长按电源键 2 秒以上，但小于 8 秒，树莓派将先软件关机，软件关机后将切断树莓派电源
  - 长按电源键 8 秒以上，将直接切断树莓派电源
- 若电压低于设定的阈值，将自动关机。
- 可调参数如下：

参数	含义	默认值
SHUTDOWN_TIME_MS	长按时间超过该阈值将软件关机。	2000
CUT_POWER_TIME_MS	长按时间超过该阈值将切断电源。	8000
ON_VOLTAGE_VOLT	开机最低电压	6.6
OFF_VOLTAGE_VOLT	低电压关机阈值	6.3
FORCE_OFF_VOLTAGE_VOLT	紧急低电压关机阈值	5.8
LOW_VOLTAGE_TIME _MS	低电压关机等待时间	30000
OVERRIDE_TIME _MS	低电压续航延长时间	3600000

说明：当电压低于 OFF_VOLTAGE 时，如果在 LOW_VOLTAGE_TIME_IN_MS 时间内，按下电源按键，开机时间将延长 OVERRIDE_TIME_IN_MS 毫秒，但当电压低于 FORCE_OFF_VOLTAGE 时，将立刻让树莓派软件关机，并切断树莓派电源；当电压高于 ON_VOLTAGE 时，将重新供电使得树莓派开机。

注意：单节锂电池的电压在 3.0V~4.2V 左右，可通过锂电池电压粗略估计电池电量，4.2V 的锂电池表示已充满，3.0V 左右的锂电池表示电量很低，锂电池大多数时间工作在 3.7V；如果单节锂电池电压低于 3.0V，则可能造该锂电池永久损毁。

如果采用外接锂电池的方式给 Power Management HAT 供电，由于 Power Management HAT 不具有升压功能，则至少需要两节锂电池(包)串联，提供 5V 以上的电源，此时，请根据具体电池串联的数量设置相应的电压数据，可参考下表提供的数据(如果采用配套锂电池盒的方式给 Power Management HAT 供电，则请选择 2S 的参数)：

	1S(不支持)	2S(1S*2)	3S(1S*3)	4S(1S*4)
ON_VOLTAGE_VOLT	3.3	6.6	9.9	13.2
OFF_VOLTAGE_VOLT	3.15	6.3	9.45	12.6
FORCE_OFF_VOLTAGE_VOLT	2.9	5.8	8.7	11.6

规定时间段处于开机状态

- 完成基础准备工作
- 打开 PowerOnInAPeriodTime 例程，编译，下载
- 树莓派将在规定的时间段处于开机状态
- 可调参数如下：

参数	含义	默认值
WAKE_UP_START_HOUR	开机时间点-小时	8
WAKE_UP_START_MINUTE	开机时间点-分钟	30
SLEEP_START_HOUR	关机时间点-小时	18
SLEEP_START_MINUTE	关机时间点-小时	30

周期性地处于开机状态

- 完成基础准备工作
- 打开 PowerOnPeriodicallyOnTimer 例程或 PowerOnPeriodicallyOnAlarm 例程，编译，下载
- 树莓派将周期性地处于开机状态
- 例程区别及可调参数如下：

例程	PowerOnPeriodicallyOnTimer	PowerOnPeriodicallyOnAlarm
区别	通过定时器定时唤醒树莓派	通过 RTC 闹钟定时唤醒树莓派
特点	简洁，但是定时精度差	精度相对较高

PowerOnPeriodicallyOnTimer		
参数	含义	默认值
PERIODIC_TIMER_TIMEBASE	周期单位	eTB_MINUTE
PERIODIC_TIMER_VALUE	周期数值	10
RPI_STAY_AWAKE_TIME _MS	开机时长	60000

PowerOnPeriodicallyOnAlarm		
参数	含义	默认值
RPI_WAKE_UP_CYCLE _MINUTE	周期数值	10
RPI_STAY_AWAKE_TIME _MS	开机时长	60000
CURRENT_THRESHOLD_MA	阈值电流(工作电流低于该阈值判定为关机状态)	110

规定时间段周期性地处于开机状态

- 完成基础准备工作
- 打开 PowerOnPeriodicallyOnTimerInAPeriodTime 例程，编译，下载
- 树莓派将在规定时间段周期性地处于开机状态

- 可调参数如下：

参数	含义	默认值
PERIODIC_TIMER_TIMEBASE	周期单位	eTB_MINUTE
PERIODIC_TIMER_VALUE	周期数值	5
RPI_STAY_AWAKE_TIME _MS	开机时长	60000
WAKE_UP_START_HOUR	开机时间点-小时	8
WAKE_UP_START_MINUTE	开机时间点-分钟	30
SLEEP_START_HOUR	关机时间点-小时	18
SLEEP_START_MINUTE	关机时间点-小时	30