

樹莓派 RTC 時鐘模組 電池可拆卸 適用樹莓派 4/3B+/ZERO 開發板

产品参数

Product parameters

本模块是专门为树莓派设计的 RTC 模块。模块采用 I2C方式和树莓派进行通讯。板载DS1307 时钟芯片以及一颗CR1220 纽扣电池保证树莓派断电后仍然能记忆时间。预留出串口，方便 TTL 方式在线调试。

可直接用 shell 脚本操作；引出树莓派串口便于调试；可编程方波输出。

典型应用：远程信息处理系统 / 智能家居。



芯片：DS1307

电池：CR1220纽扣电池

工作电压：5V

日历闹钟：2个

工作温度：-10°C至+85°C

兼容版本：兼容树莓派全系列产品

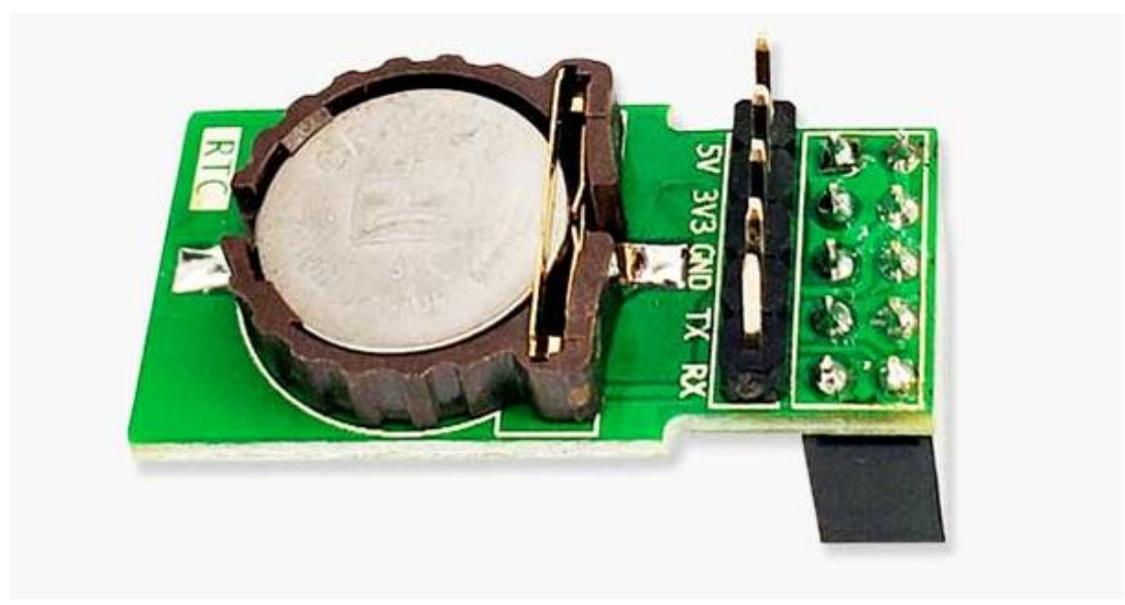
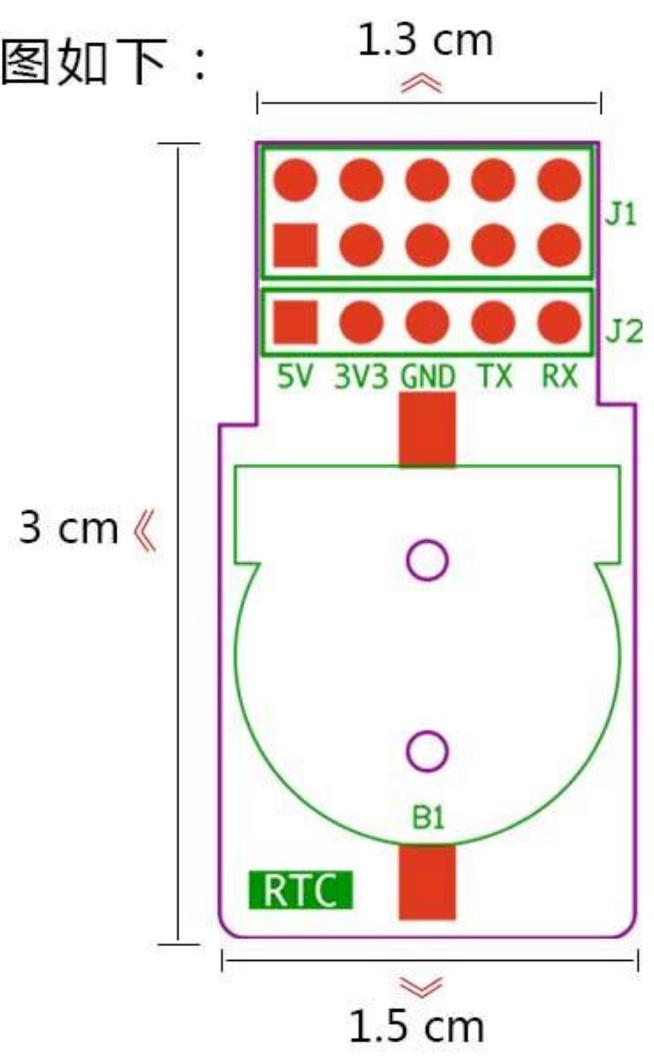
接口：2X13 和 A+/B+/2代 2X20 插座

接口间距：2 X 5p 2.54mm

0°C至+40°C范围内时钟精度为±2ppm

实时时钟提供秒、分、时、星期、日期、月、年信息，并提供有效期至2100年的闰年补偿

模块尺寸图如下：



树莓派产品搭配时钟模块实拍效果图

将对应5V 电平，GND / EX / RX 引脚和模块连接即可
(备注) 不能同时使用串口先和 USB 接口供电



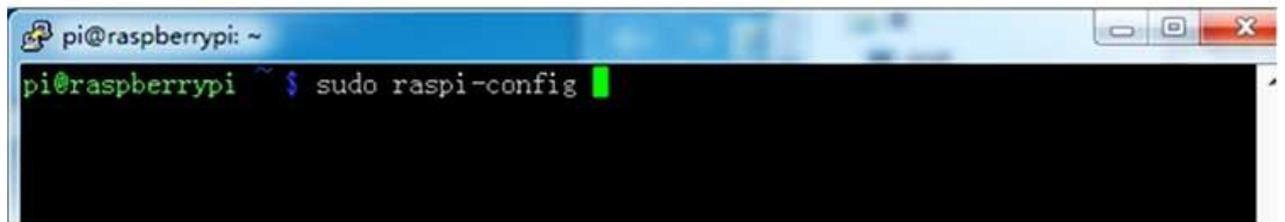
DS1307是低成本、高精度I2C实时时钟(RTC)，该器件包含电池输入端，断开主电源时仍可保持的计时Ds1307具有长期度，并减少了生产线的元件数量。DS1307提供商用级和工业级温度范围，采用8引脚64mil的SO封装,具有低功耗特性,在电池供电待机状态下,静态消耗电流低于500nA。

RTC保存秒、分、时、星期、日期、月和年信息。少于31天的月份，将自动调整月末的日期，包括闰年的修正。时钟的工作格式可以是24小时或带/AM/PM指示的12小时格式。提供两个可设置的日历闹钟和一个可设置的方波输出。地址与数据通过I2C双向总线串行传输。

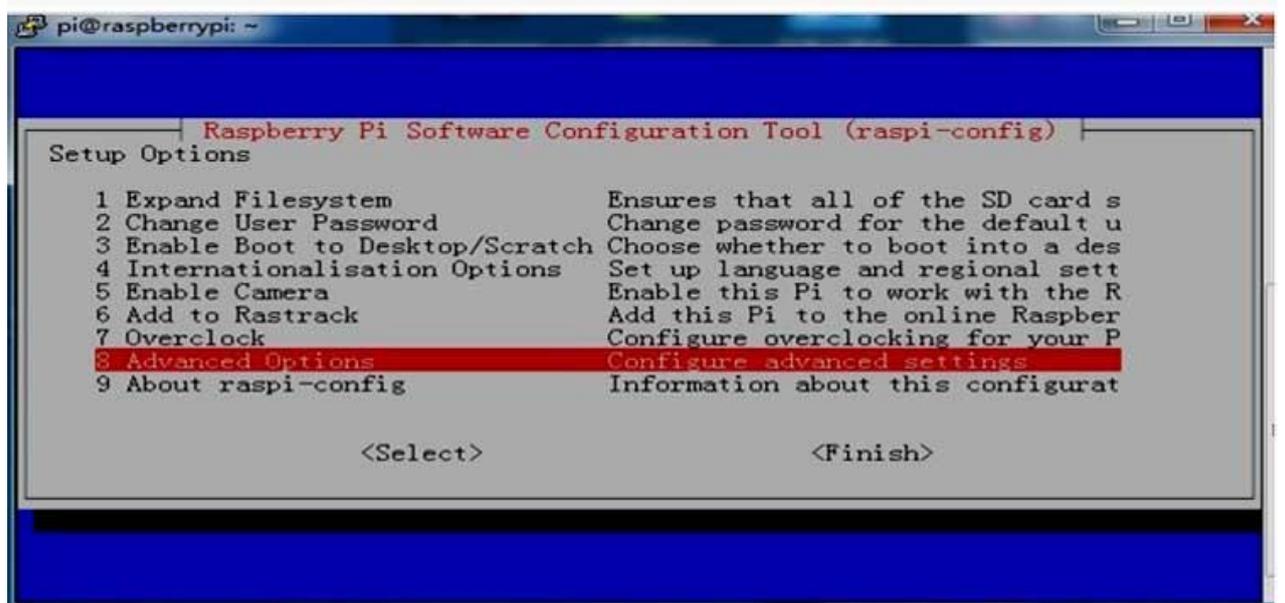
精密的、经过温度补偿的电压基准和比较器电路用来监视VCC状态，检测电源故障，提供复位输出，并在必要时自动切换到备份电源。

测试方法如下：

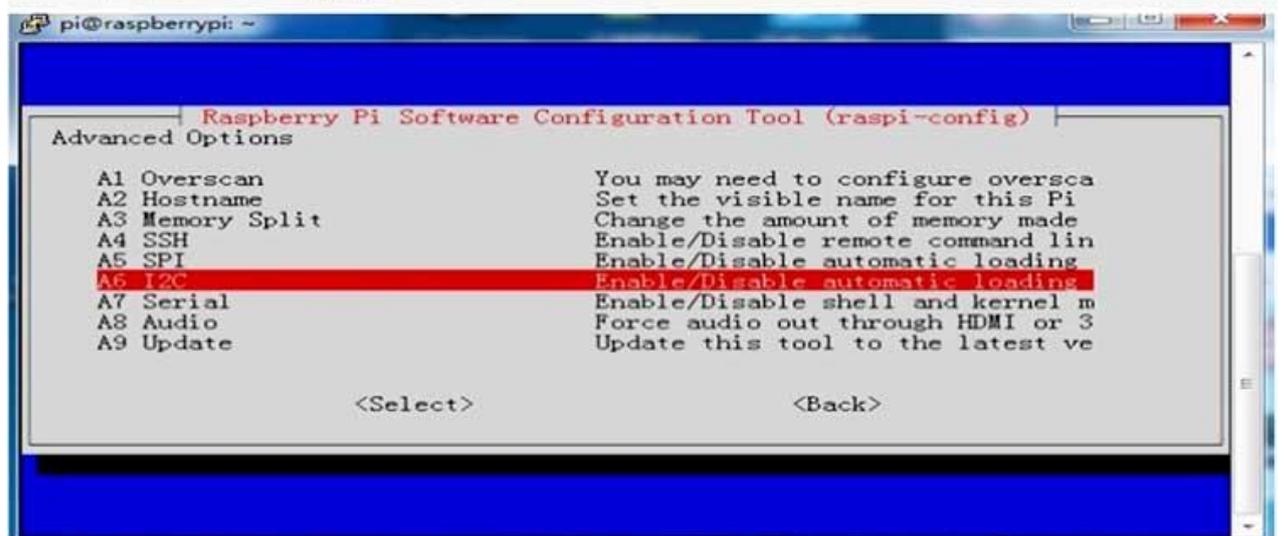
1. 输入sudo raspi-config 设置树莓派打开 i2c 接口



2. 在设置页面中选择 Advanced Options 项



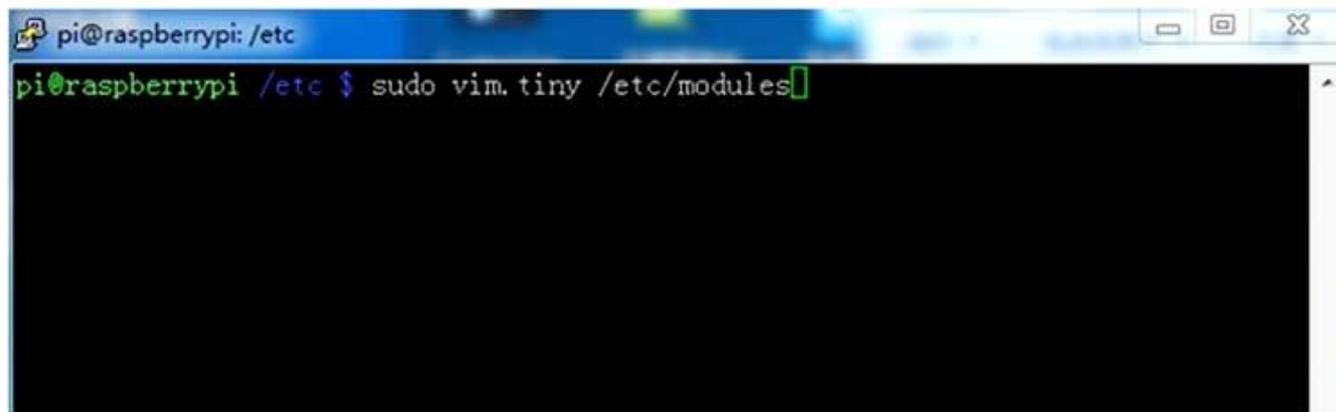
3. 选择 I2C 条目



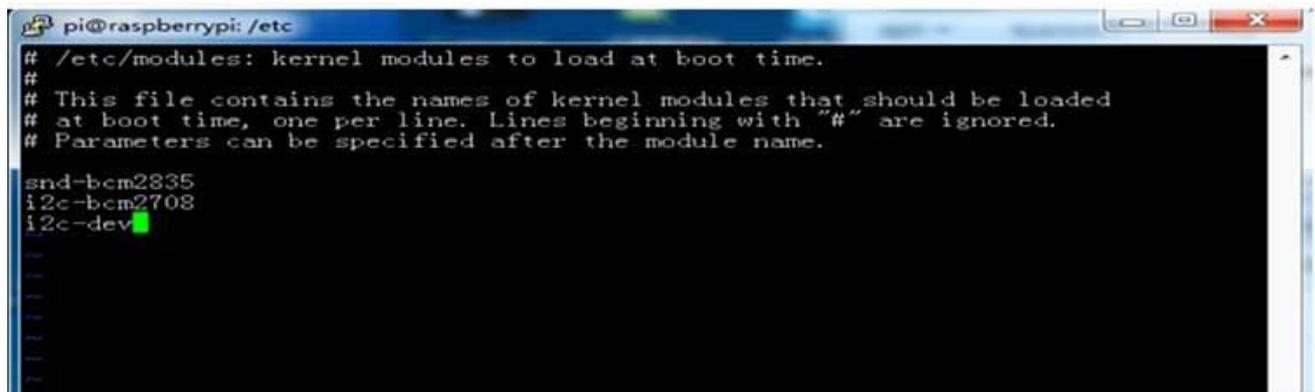
4. 选择 YES 打开 I2C 模块



5. 输入 `sudo vim.tiny /etc / modules` 添加模块

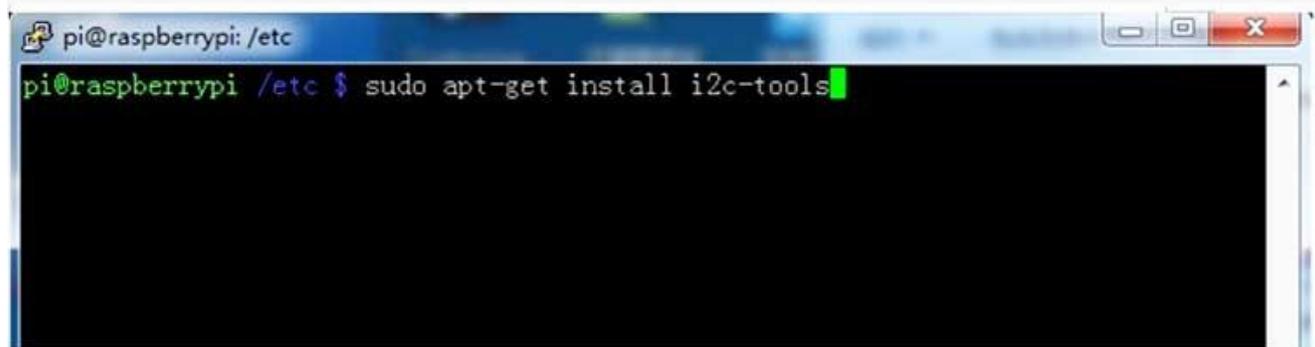


6. 如图所示添加 i2C-dev 设备

A terminal window titled 'pi@raspberrypi: /etc' showing the contents of the /etc/modules file. The text includes instructions on how to use the file and lists kernel modules: snd-bcm2835, i2c-bcm2708, and i2c-dev.

```
pi@raspberrypi: /etc
# /etc/modules: kernel modules to load at boot time.
#
# This file contains the names of kernel modules that should be loaded
# at boot time, one per line. Lines beginning with "#" are ignored.
# Parameters can be specified after the module name.
snd-bcm2835
i2c-bcm2708
i2c-dev
```

7. 安装 i2C-tools 软件，输入 `sudo apt-get install i2c-tools`

A terminal window titled 'pi@raspberrypi: /etc' showing the command 'sudo apt-get install i2c-tools' being entered at the prompt.

```
pi@raspberrypi: /etc $ sudo apt-get install i2c-tools
```

8. 树莓派，输入 `sudo reboot`。重启之后输入 `sudo i2cdetect -y 1` 如果模块正常安装就可以被识别到了。如图：

```
pi@raspberrypi: /dev
pi@raspberrypi /dev $ sudo i2cdetect -y -a 1
   0 1 2 3 4 5 6 7 8 9 a b c d e f
00: -- -- -- -- -- -- -- -- -- -- -- -- --
10: -- -- -- -- -- -- -- -- -- -- -- -- --
20: -- -- -- -- -- -- -- -- -- -- -- -- --
30: -- -- -- -- -- -- -- -- -- -- UU -- -- --
40: -- -- -- -- -- -- -- -- -- -- -- -- --
50: -- -- -- -- -- -- -- -- -- -- -- -- --
60: -- -- -- -- -- -- -- -- 68 -- -- -- -- --
70: -- -- -- -- -- -- -- -- -- -- -- -- --
pi@raspberrypi /dev $
```

如何使用模块呢输入 `sudo su-` 切换到 root 身份
输入 `modprobe i2c -dev` 加载 i2c 设备

```
pi@raspberrypi: ~
root@raspberrypi:~# modprobe i2c-dev
```

如何使用模块呢输入 `sudo su-` 切换到 root 身份
输入 `modprobe i2c -dev` 加载 i2c 设备

```
root@raspberrypi:~# echo "ds1307 0x68" > /sys/class/i2c-adapter/i2c-1/new_device ^C
root@raspberrypi:~#
```

之后就可以使用 `hwclock` 指令来使用本模块了
想要更多了解 `hwclock` 指令可以使用 `man hwclock`
来详细了解，简要说明一下 `hwclock -r` 为获取RTC模块
时间 `hwclock -w` 为设置系统时间为RTC时间

```
pi@raspberrypi: ~
pi@raspberrypi ~ $ sudo hwclock -r
Tue 22 Nov 2011 12:20:29 UTC -0.050531 seconds
pi@raspberrypi ~ $
```