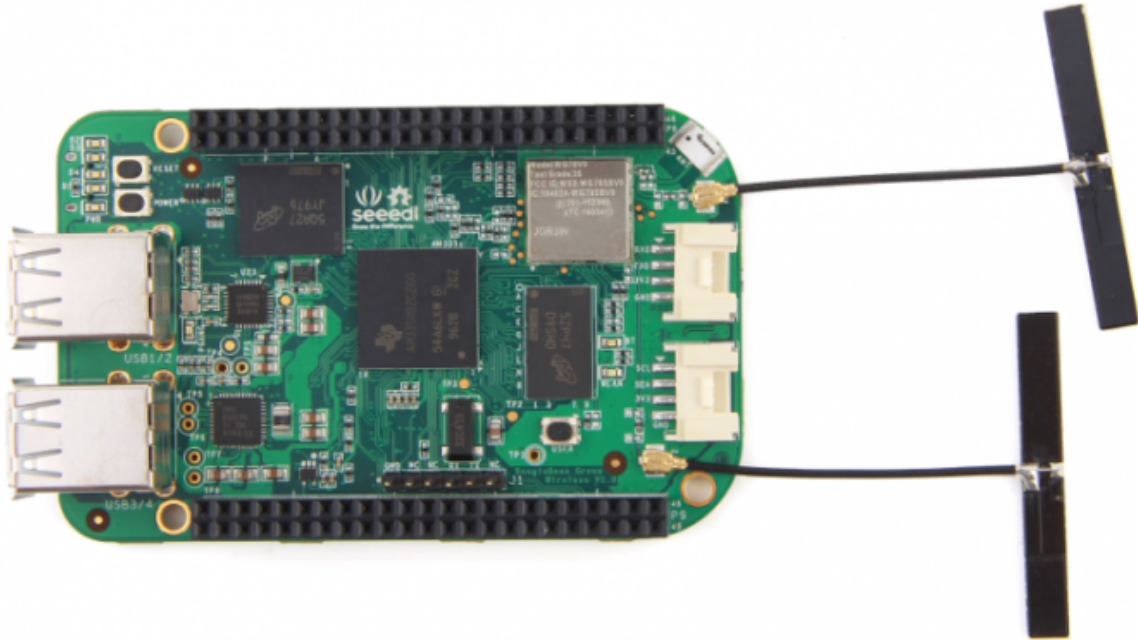


BeagleBone Green Wireless SKU:102010048



SeeedStudio BeagleBoneGreen Wireless (BBGW) 是由 BeagleBoard.org 和 Seeed Studio 共同努力生产的成果。它是基于 BeagleBone Black 的开源硬件设计，并开发成这种辨识度很高版本。SeeedStudio BeagleBone Green Wireless 包括高性能灵活的 WiFi /蓝牙接口和两个 Grove 连接器，使其更容易连接到大型 Grove 传感器系列。移除了板载 HDMI 和以太网，为这些无线功能和 Grove 连接模块提供空间。

产品特性

- 完全兼容 **BeagleBone Black**
- 处理器: **AM335x 1GHzARM®Cortex-A8**
- 4GB 8位eMMC 板上闪存
- 3D 图形加速器
- NEON 浮点加速器
- 2x PRU 32位微控制器
- 连接特性
- 用于电源和通信的 USB 客户端
- 带 4 端口集线器的 USB 主机

- WiFi 802.11 b / g / n 2.4GHz
- 蓝牙 4.1 与 BLE
- 2x 46 针头
- 2x Grove 连接器（ I2C 和 UART ）
- 软件兼容性
- Debian
- Android
- Ubuntu
- Cloud9 IDE on Node.js w/ BoneScript 库
- 更多

规格参数

项目	参数值
处理器	AM335x 1GHz ARM Cortex-A8
内存	512MB DDR3
板载闪存	4GB eMMC
CPU支持	NEON 浮点运算和 3D 图形加速器
USB支持	供电与通信
USB	USB2.0 主机 x4
Grove 接口	2 (一个 I2C 和一个 UART)
GPIO	2 x 46 引脚
以太网	Wi-Fi 802.11b/g/n 2.4GHz 和 蓝牙 4.1 LE
工作温度	0 ~ 75 °C

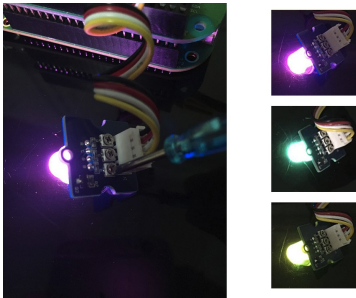
创意应用

- 物联网
- 智慧之家
- 工业
- 自动化与过程控制
- 人机接口
- 传感器中心
- 机器人

BBGW入门教程

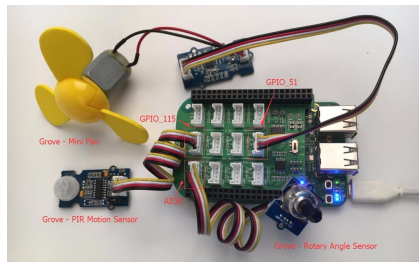
以下是初学者使用一些 Grove 模块来实现使用 BeagleBone Green Wireless (BBGW) 完成有趣的想法的教程。本教程基于 Python 和 mraa / upm 库。

#1 The Breath LED



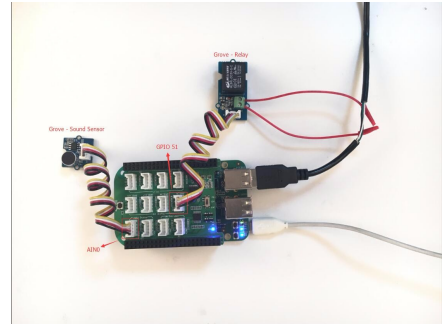
[观看视频](#)

#2 Storm on your table



[观看视频](#)

#3 Speak Louuuuudly



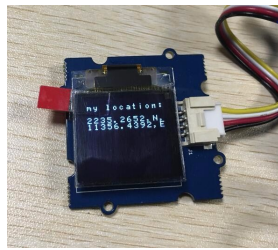
[观看视频](#)

#4 How hot is it today?



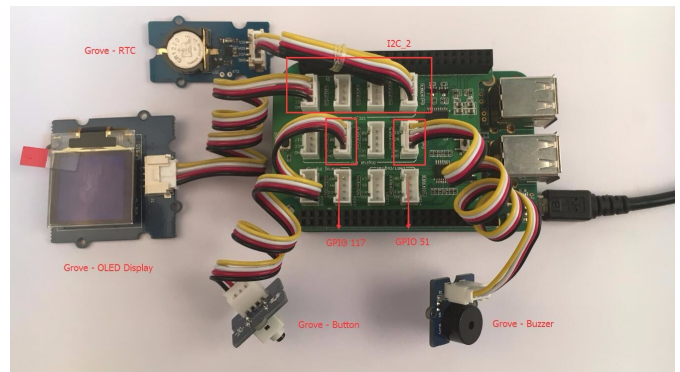
[观看视频](#)

#5 Where are you?



[观看视频](#)

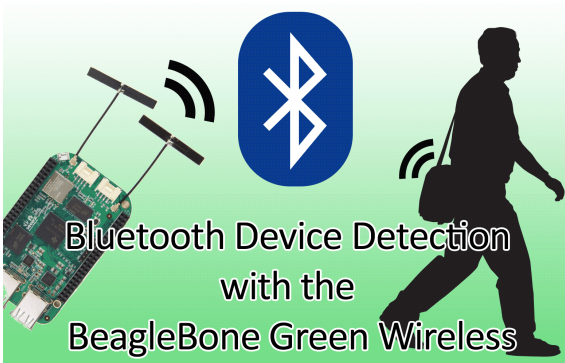
#6 My Little alarm clock



[观看视频](#)

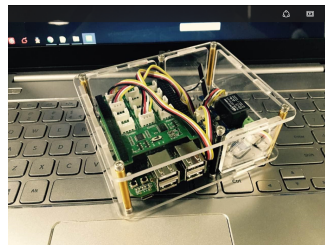
趣味项目

Bluetooth Device Detection



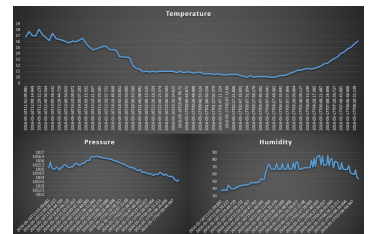
[立即参考](#)

Home Control Center



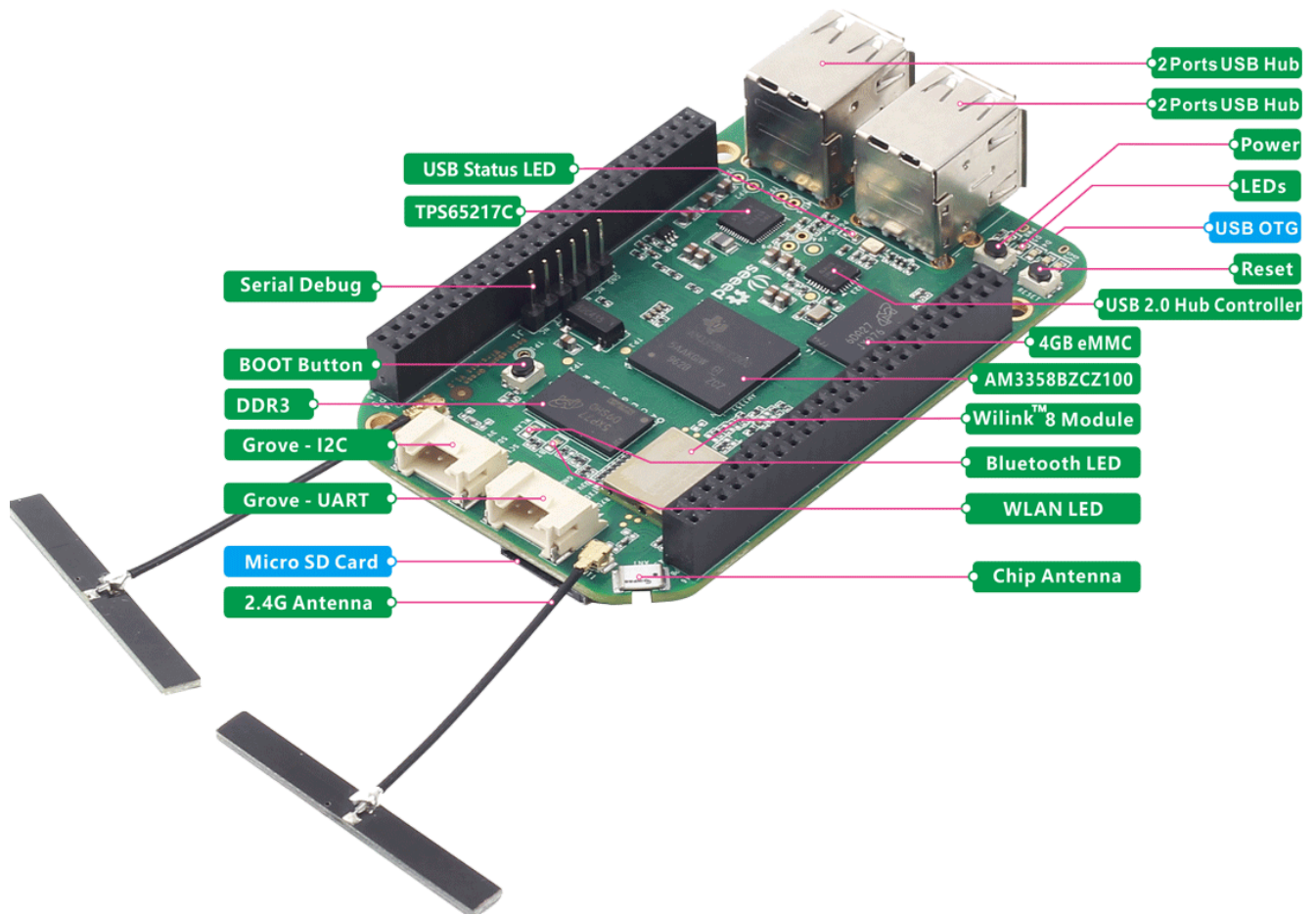
[立即参考](#)

SAP HCP IoT Service



[立即参考](#)

硬件概述



引脚功能

每个数字 I / O 引脚有8种可以选择的不同模式，包括 GPIO 引脚

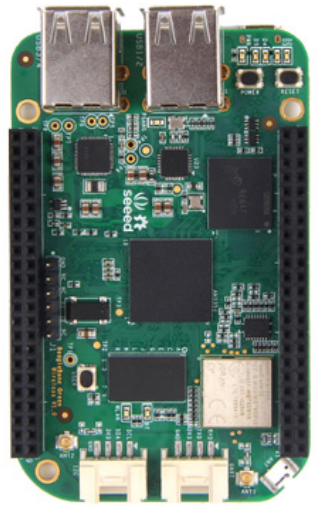
65种可能的数字 I / O

!!!Note - 在GPIO模式下，每个数字I / O都可产生中断

Cape Expansion Headers

P9

DGND	1	2	DGND
VDD_3V3	3	4	VDD_3V3
VDD_5V	5	6	VDD_5V
SYS_5V	7	8	SYS_5V
PWR_BUT	9	10	SYS_RESETn
UART4_RXD	11	12	GPIO_60
UART4_TXD	13	14	EHRPWM1A
GPIO_48	15	16	EHRPWM1B
SPI0_CS0	17	18	SPI0_D1
I2C2_SCL	19	20	I2C2_SDA
UART2_TXD	21	22	UART2_RXD
GPIO_49	23	24	UART1_TXD
GPIO_117	25	26	UART1_RXD
GPIO_115	27	28	GPIO_113
GPIO_111	29	30	GPIO_112
GPIO_110	31	32	VDD_ADC
AIN4	33	34	GND_ADC
AIN6	35	36	AIN5
AIN2	37	38	AIN3
AIN0	39	40	AIN1
GPIO_20	41	42	ECAFPWM0
DGND	43	44	DGND
DGND	45	46	DGND



P8

DGND	1	2	DGND
MMC1_DAT6	3	4	MMC1_DAT7
MMC1_DAT2	5	6	MMC1_DAT3
GPIO_66	7	8	GPIO_67
GPIO_69	9	10	GPIO_68
GPIO_45	11	12	GPIO_44
EHRPWM2B	13	14	GPIO_26
GPIO_47	15	16	GPIO_46
GPIO_27	17	18	GPIO_65
EHRPWM2A	19	20	MMC1_CMD
MMC1_CLK	21	22	MMC1_DAT5
MMC1_DAT4	23	24	MMC1_DAT1
MMC1_DAT0	25	26	GPIO_61
GPIO_86	27	28	GPIO_88
GPIO_87	29	30	GPIO_89
GPIO_10	31	32	GPIO_11
GPIO_9	33	34	GPIO_81
GPIO_8	35	36	GPIO_80
GPIO_78	37	38	GPIO_79
GPIO_76	39	40	GPIO_77
GPIO_74	41	42	GPIO_75
GPIO_72	43	44	GPIO_73
GPIO_70	45	46	GPIO_71

LEGEND

- Power/Ground/Reset
- Available Digital
- Available PWM
- Shared I2C Bus
- Reconfigurable Digital
- Analog inputs (1.8V)

PWM 和定时器

!!!Note - 多达8个数字 I/O 引脚可配置脉冲宽度调制器 (PWM)，以产生信号控制电机或创建伪模拟电压电平，而不占用任何额外的 CPU 周期。

8 PWMs and 4 timers

P9				P8			
DGND	1	2	DGND	DGND	1	2	DGND
VDD_3V3	3	4	VDD_3V3	GPIO_38	3	4	GPIO_39
VDD_5V	5	6	VDD_5V	GPIO_34	5	6	GPIO_35
SYS_5V	7	8	SYS_5V	TIMER4	7	8	TIMER7
PWR_BUT	9	10	SYS_RESETN	TIMER5	9	10	TIMER6
GPIO_30	11	12	GPIO_60	GPIO_45	11	12	GPIO_44
GPIO_31	13	14	EHRPWM1A	EHRPWM2B	13	14	GPIO_26
GPIO_48	15	16	EHRPWM1B	GPIO_47	15	16	GPIO_46
GPIO_5	17	18	GPIO_4	GPIO_27	17	18	GPIO_65
I2C2_SCL	19	20	I2C2_SDA	EHRPWM2A	19	20	GPIO_63
EHRPWMOB	21	22	EHRPWM0A	GPIO_62	21	22	GPIO_37
GPIO_49	23	24	GPIO_15	GPIO_36	23	24	GPIO_33
GPIO_117	25	26	GPIO_14	GPIO_32	25	26	GPIO_61
GPIO_115	27	28	ECAPPWM2	GPIO_86	27	28	GPIO_88
EHRPWMOB	29	30	GPIO_112	GPIO_87	29	30	GPIO_89
EHRPWM0A	31	32	VDD_ADC	GPIO_10	31	32	GPIO_11
AIN4	33	34	GNDA_ADC	GPIO_9	33	34	EHRPWM1B
AIN6	35	36	AIN5	GPIO_8	35	36	EHRPWM1A
AIN2	37	38	AIN3	GPIO_78	37	38	GPIO_79
AIN0	39	40	AIN1	GPIO_76	39	40	GPIO_77
GPIO_20	41	42	ECAPPWMO	GPIO_74	41	42	GPIO_75
DGND	43	44	DGND	GPIO_72	43	44	GPIO_73
DGND	45	46	DGND	EHRPWM2A	45	46	EHRPWM2B

模拟输入

!!!Note - 确保模拟输入引脚输入不多于 1.8V。这是一个具有 8 个通道的 12 位模数转换器，其中 7 个在连接头上可用。

7 analog inputs (1.8V)

P9				P8			
DGND	1	2	DGND	DGND	1	2	DGND
VDD_3V3	3	4	VDD_3V3	GPIO_38	3	4	GPIO_39
VDD_5V	5	6	VDD_5V	GPIO_34	5	6	GPIO_35
SYS_5V	7	8	SYS_5V	GPIO_66	7	8	GPIO_67
PWR_BUT	9	10	SYS_RESETN	GPIO_69	9	10	GPIO_68
GPIO_30	11	12	GPIO_60	GPIO_45	11	12	GPIO_44
GPIO_31	13	14	GPIO_50	GPIO_23	13	14	GPIO_26
GPIO_48	15	16	GPIO_51	GPIO_47	15	16	GPIO_46
GPIO_5	17	18	GPIO_4	GPIO_27	17	18	GPIO_65
I2C2_SCL	19	20	I2C2_SDA	GPIO_22	19	20	GPIO_63
GPIO_3	21	22	GPIO_2	GPIO_62	21	22	GPIO_37
GPIO_49	23	24	GPIO_15	GPIO_36	23	24	GPIO_33
GPIO_117	25	26	GPIO_14	GPIO_32	25	26	GPIO_61
GPIO_115	27	28	GPIO_113	GPIO_86	27	28	GPIO_88
GPIO_111	29	30	GPIO_112	GPIO_87	29	30	GPIO_89
GPIO_110	31	32	VDD_ADC	GPIO_10	31	32	GPIO_11
AIN4	33	34	GNDA_ADC	GPIO_9	33	34	GPIO_81
AIN6	35	36	AIN5	GPIO_8	35	36	GPIO_80
AIN2	37	38	AIN3	GPIO_78	37	38	GPIO_79
AIN0	39	40	AIN1	GPIO_76	39	40	GPIO_77
GPIO_20	41	42	GPIO_7	GPIO_74	41	42	GPIO_75
DGND	43	44	DGND	GPIO_72	43	44	GPIO_73
DGND	45	46	DGND	GPIO_70	45	46	GPIO_71

UART

!!!Note - 有一个专用连接头用于连接 UART0 引脚并用于连接调试。另外还有五个串行扩展连接端口，但其中一个只有一个是具有方向的。

4 UARTs and 1 TX only

P9				P8			
DGND	1	2	DGND	DGND	1	2	DGND
VDD_3V3	3	4	VDD_3V3	GPIO_38	3	4	GPIO_39
VDD_5V	5	6	VDD_5V	GPIO_34	5	6	GPIO_35
SYS_5V	7	8	SYS_5V	GPIO_66	7	8	GPIO_67
PWR_BUT	9	10	SYS_RESETN	GPIO_69	9	10	GPIO_68
UART4_RXD	11	12	GPIO_60	GPIO_45	11	12	GPIO_44
UART4_TXD	13	14	GPIO_50	GPIO_23	13	14	GPIO_26
GPIO_48	15	16	GPIO_51	GPIO_47	15	16	GPIO_46
GPIO_5	17	18	GPIO_4	GPIO_27	17	18	GPIO_65
UART1_RTSN	19	20	UART1_CTSN	GPIO_22	19	20	GPIO_63
UART2_TXD	21	22	UART2_RXD	GPIO_62	21	22	GPIO_37
GPIO_49	23	24	UART1_TXD	GPIO_36	23	24	GPIO_33
GPIO_117	25	26	UART1_RXD	GPIO_32	25	26	GPIO_61
GPIO_115	27	28	GPIO_113	GPIO_86	27	28	GPIO_88
GPIO_111	29	30	GPIO_112	GPIO_87	29	30	GPIO_89
GPIO_110	31	32	VDD_ADC	UART5_CTSN+	31	32	UART5_RTSN
AIN4	33	34	GNDA_ADC	UART4_RTSN	33	34	UART3_RTSN
AIN6	35	36	AIN5	UART4_CTSN	35	36	UART3_CTSN
AIN2	37	38	AIN3	UART5_TXD+	37	38	UART5_RXD+
AIN0	39	40	AIN1	GPIO_76	39	40	GPIO_77
GPIO_20	41	42	UART3_TXD	GPIO_74	41	42	GPIO_75
DGND	43	44	DGND	GPIO_72	43	44	GPIO_73
DGND	45	46	DGND	GPIO_70	45	46	GPIO_71

I2C 端口

!!!Note 第一个 I2C 总线用于读取 Cape 附加板上的 EEPROMS ，不能用于其他数字 I/O 端口操作，而不会影响该功能，但您仍然可使用它在可用地址中添加其他 I2C 设备。第二个 I2C 总线可供您配置和使用。

2 I2C ports

P9				P8			
DGND	1	2	DGND	DGND	1	2	DGND
VDD_3V3	3	4	VDD_3V3	GPIO_38	3	4	GPIO_39
VDD_5V	5	6	VDD_5V	GPIO_34	5	6	GPIO_35
SYS_5V	7	8	SYS_5V	GPIO_66	7	8	GPIO_67
PWR_BUT	9	10	SYS_RESETN	GPIO_69	9	10	GPIO_68
GPIO_30	11	12	GPIO_60	GPIO_45	11	12	GPIO_44
GPIO_31	13	14	GPIO_50	GPIO_23	13	14	GPIO_26
GPIO_48	15	16	GPIO_51	GPIO_47	15	16	GPIO_46
I2C1_SCL	17	18	I2C1_SDA	GPIO_27	17	18	GPIO_65
I2C2_SCL	19	20	I2C2_SDA	GPIO_22	19	20	GPIO_63
I2C2_SCL	21	22	I2C2_SDA	GPIO_62	21	22	GPIO_37
GPIO_49	23	24	I2C1_SCL	GPIO_36	23	24	GPIO_33
GPIO_117	25	26	I2C1_SDA	GPIO_32	25	26	GPIO_61
GPIO_115	27	28	GPIO_113	GPIO_86	27	28	GPIO_88
GPIO_111	29	30	GPIO_112	GPIO_87	29	30	GPIO_89
GPIO_110	31	32	VDD_ADC	GPIO_10	31	32	GPIO_11
AIN4	33	34	GNDA_ADC	GPIO_9	33	34	GPIO_81
AIN6	35	36	AIN5	GPIO_8	35	36	GPIO_80
AIN2	37	38	AIN3	GPIO_78	37	38	GPIO_79
AIN0	39	40	AIN1	GPIO_76	39	40	GPIO_77
GPIO_20	41	42	GPIO_7	GPIO_74	41	42	GPIO_75
DGND	43	44	DGND	GPIO_72	43	44	GPIO_73
DGND	45	46	DGND	GPIO_70	45	46	GPIO_71

SPI

!!!Note 为了快速移出数据，您可以考虑使用其中一个 SPI 端口。

2 SPI ports

P9				P8			
DGND	1	2	DGND	DGND	1	2	DGND
VDD_3V3	3	4	VDD_3V3	GPIO_38	3	4	GPIO_39
VDD_5V	5	6	VDD_5V	GPIO_34	5	6	GPIO_35
SYS_5V	7	8	SYS_5V	GPIO_66	7	8	GPIO_67
PWR_BUT	9	10	SYS_RESETN	GPIO_69	9	10	GPIO_68
GPIO_30	11	12	GPIO_60	GPIO_45	11	12	GPIO_44
GPIO_31	13	14	GPIO_50	GPIO_23	13	14	GPIO_26
GPIO_48	15	16	GPIO_51	GPIO_47	15	16	GPIO_46
SPIO_CS0	17	18	SPIO_D1	GPIO_27	17	18	GPIO_65
SPI1_CS1	19	20	SPI1_CS0	GPIO_22	19	20	GPIO_63
SPIO_DO	21	22	SPIO_SCLK	GPIO_62	21	22	GPIO_37
GPIO_49	23	24	GPIO_15	GPIO_36	23	24	GPIO_33
GPIO_117	25	26	GPIO_14	GPIO_32	25	26	GPIO_61
GPIO_115	27	28	SPI1_CS0	GPIO_86	27	28	GPIO_88
SPI1_DO	29	30	SPI1_D1	GPIO_87	29	30	GPIO_89
SPI1_SCLK	31	32	VDD_ADC	GPIO_10	31	32	GPIO_11
AIN4	33	34	GNDA_ADC	GPIO_9	33	34	GPIO_81
AIN6	35	36	AIN5	GPIO_8	35	36	GPIO_80
AIN2	37	38	AIN3	GPIO_78	37	38	GPIO_79
AIN0	39	40	AIN1	GPIO_76	39	40	GPIO_77
GPIO_20	41	42	SPI1_CS1	GPIO_74	41	42	GPIO_75
DGND	43	44	DGND	GPIO_72	43	44	GPIO_73
DGND	45	46	DGND	GPIO_70	45	46	GPIO_71

入门指导

!!!Note 本章正在 Win10 下编写。 这些步骤对于其他操作系统也是适用的。

步骤1： 通过 USB 连接您的 BBGW

使用提供的 USB 数据线将您的 BBGW 连接到计算机。 这将同时为电路板供电并提供开发接口。 BBGW 将从 板载 2GB 或 4GB 的 eMMC 启动 Linux。

BBGW 将作为闪存驱动器运行，为您提供文档和驱动程序的本地副本。 请注意，此接口可能不能用于重新配置具有新映像的 microSD 卡，但可用于使用 uEnv.txt 文件更新引导参数。

您将看到 PWR LED 稳定点亮。 在10秒钟内，您应该看到其他LED在其默认配置中闪烁。

- 在引导时设置 **D2** 端口，以呼吸节奏闪烁
- 在 microSD 卡访问期间，将 **D3** 端口设置为启动时亮起
- 在启动时，将 **D4** 端口设置为在 CPU 活动期间点亮
- 将 **D5** 在启动时设置为在 eMMC 访问期间点亮

步骤2： 安装驱动程序

为您的操作系统安装驱动程序，使您可以通过 USB 访问 Beagle。 其他驱动程序可让您连续访问您的主板。

操作系统	USB驱动程序	备注
Windows (64-bit)	64-bit installer	
Windows (32-bit)	32-bit installer	
Mac OS X	Network Serial	安装两套驱动程序.
Linux	mkudevrule.sh	驱动程序安装不是必需的, 但您可能会发现几个 udev 规则有用.

!!!Note 对于窗口系统, 请注意:

- * **Windows Driver** 认证警告可能弹出两三次。 点击“忽略”, “安装”或“运行”
- * 检查您是否正在运行32位或64位Windows
- * 在没有最新服务版本的系统上, 您可能会收到错误 (0xc000007b)。 在这种情况下请 [安装] (<https://www.microsoft.com/en-us/download/confirmation.aspx?id=13523>) 并重试:
- * 您可能需要重新启动 Windows。
- * 这些驱动程序已经过测试, 适用于 Windows 10

步骤3: 浏览你的Beagle

使用 Chrome 或 Firefox (如果Internet Explorer不起作用), 浏览您的电路板上运行的 Web 服务器。 它将加载一个演示文稿, 向您展示电路板的功能。 使用键盘上的箭头键导航到演示文稿。

单击<http://192.168.7.2> 启动您的 BBGW。 较旧的软件会要求您使用 BEAGLE_BONE 驱动器启动网络。 使用最新的软件, 就不再需要该步骤。

The screenshot shows the BeagleBoard.org website interface. At the top, there's a navigation bar with the BeagleBoard logo and social media links. Below that, a 'New Tutorial' section is visible. The main content area features a green notification box stating 'Your board is connected! BeagleBone Green S/N BBG115070062 running BoneScript 0.2.4 at beaglebone.local'. Below this, the page title is 'BeagleBone: open-hardware expandable computer' with the tagline 'Artist-tested, engineer approved'. The page includes a grid of images showing various BeagleBone projects and a list of 'Latest ARM open source focused on easy hardware experimentation'.

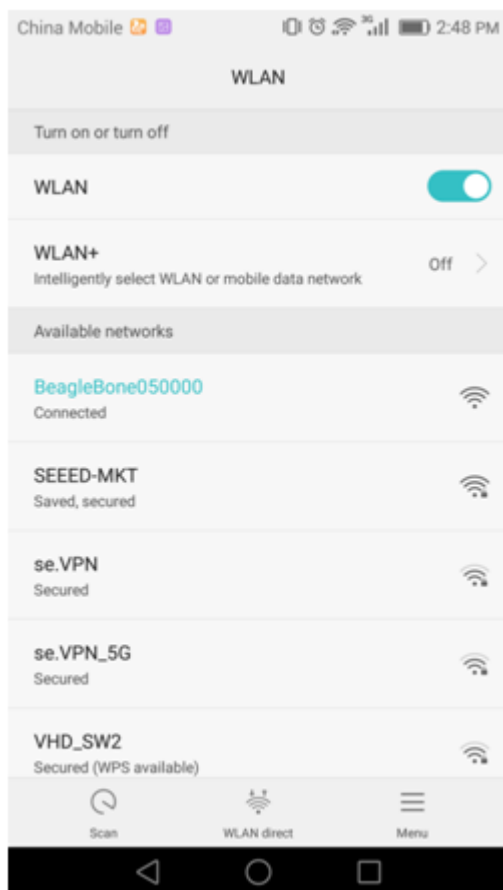
步骤4: Cloud9 IDE

要开始编辑您的主板上的程序，可以单击使用Cloud9 IDE

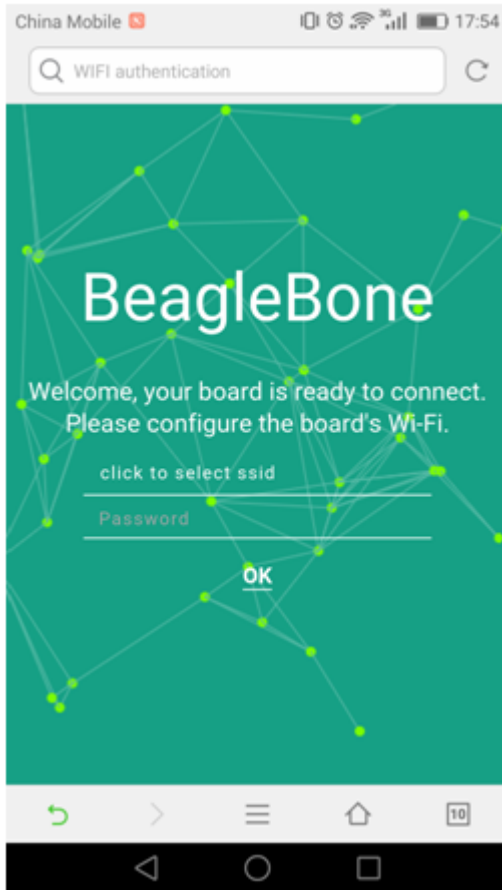
Open Cloud9 IDE of BBG

步骤5：将您的BBGW连接到Wi-Fi

使用智能手机或电脑扫描本地 Wi-Fi 网络并连接到名为 “BeagleBone XXX” 的 AP



连接成功后，将自动进入登录页面。选择您的 Wi-Fi 的 SSID 并输入密码，然后单击确定。



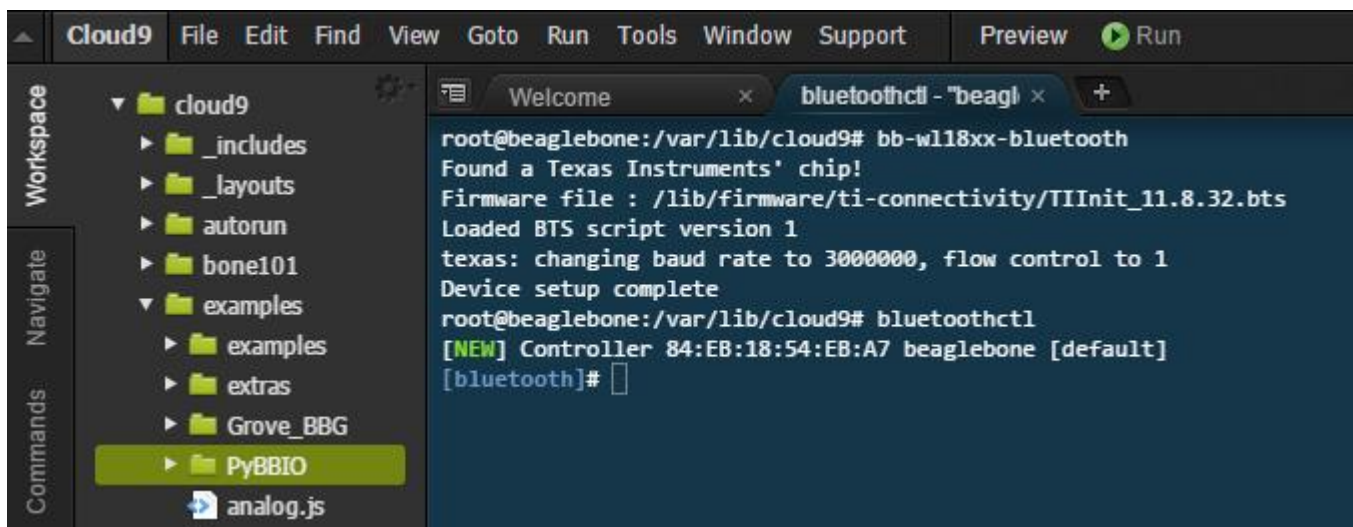
现在您的 BBGW 已连接到 Wifi 。



步骤6: 将您的 BBGW 连接到蓝牙设备

连接到 Cloud9 IDE 并启动一个新的终端。使用以下命令启动蓝牙配置:


```
bb-wl18x-bluetooth
bluetoothctl
```



```
Cloud9 File Edit Find View Goto Run Tools Window Support Preview Run
Workspace
  cloud9
    _includes
    _layouts
    autorun
    bone101
    examples
      examples
      extras
      Grove_BBG
      PyBBIO
    analog.js
  Navigate
  Commands

Welcome x bluetoothctl - "beagl x +
root@beaglebone:/var/lib/cloud9# bb-wl18x-bluetooth
Found a Texas Instruments' chip!
Firmware file : /lib/firmware/ti-connectivity/TIInit_11.8.32.bts
Loaded BTS script version 1
texas: changing baud rate to 3000000, flow control to 1
Device setup complete
root@beaglebone:/var/lib/cloud9# bluetoothctl
[NEW] Controller 84:EB:18:54:EB:A7 beaglebone [default]
[bluetooth]#
```

键入 `scan on` 来扫描本地蓝牙设备。我的设备名为“jy”。



```
[bluetooth]# scan on
Discovery started
[CHG] Controller 84:EB:18:54:EB:A7 Discovering: yes
[NEW] Device 00:11:11:11:11:07 LAPTOP-Q2IJ7NUQ
[NEW] Device 10:10:10:10:10:0A 红米手机
[NEW] Device 0C:xx:xx:xx:xx:0B 0C-xx-xx-xx-xx-0B
[NEW] Device 49:49:49:49:49:47 49-F...-47
[CHG] Device 10:10:10:10:10:0A RSSI: -79
[CHG] Device 0C:xx:xx:xx:xx:0B RSSI: -59
[CHG] Device 0C:xx:xx:xx:xx:0B Name: jy
[CHG] Device 0C:xx:xx:xx:xx:0B Alias: jy
[CHG] Device 0C:xx:xx:xx:xx:0B UUIDs:
00001200-0000-1000-8000-000000000000
0000111f-0000-1000-8000-000000000000
0000112f-0000-1000-8000-000000000000
0000110a-0000-1000-8000-000000000000
0000110c-0000-1000-8000-000000000000
00001132-0000-1000-8000-000000000000
00000000-0000-1000-8000-000000000000
2d8d2466-0000-1000-8000-000000000000
[NEW] Device 0C:xx:xx:xx:xx:0B C6 E3
[bluetooth]#
```

复制设备mac地址，然后使用以下命令连接到设备：

```
pair 0C:xx:xx:xx:xx:0B
trust 0C:xx:xx:xx:xx:0B
connect 0C:xx:xx:xx:xx:0B
```

```
[bluetooth]# pair 0C: :0B
Attempting to pair with 0C: :0B
[CHG] Device 0C: :0B Connected: yes
[CHG] Device 0C: :0B Modalias: bluetooth:v004Cp6F02d0930
[CHG] Device 0C: :0B UUIDs:
00000000-
00001000-
0000110a-
0000110c-
0000110e-
00001116-
0000111f-
0000112f-
00001132-
00001200-
[CHG] Device 0C: :0B Paired: yes
Pairing successful
[CHG] Device 0C: :0B Connected: no

[bluetooth]# trust 0C: :0B
[CHG] Device 0C: :0B Trusted: yes
Changing 0C: :0B trust succeeded
[bluetooth]# connect 0C: :0B
Attempting to connect to 0C: :0B
[CHG] Device 0C: :0B Connected: yes
Connection successful
```

现在您的 BBGW 已连接到蓝牙设备。键入`quit`回到终端。在 BBGW 播放音乐，然后您会听到蓝牙音箱上的音乐。

更新到最新的软件

您需要将主板更新到最新的软件以保持更好的性能，这里我们将向您展示如何逐步实现。

步骤1：下载最新的软件

首先，您必须在这里下载合适的图像。

[Download the latest image of BBGW](#)

!!!Note 由于文件较大，此下载可能需要约30分钟或更长时间。

您下载的文件将有一个 `.img.xz` 扩展名。这是SD卡的压缩扇区图像。

步骤2：安装压缩实用程序并解压缩

下载并安装 [7-zip](#)。

!!!Note 选择适合您系统的版本。

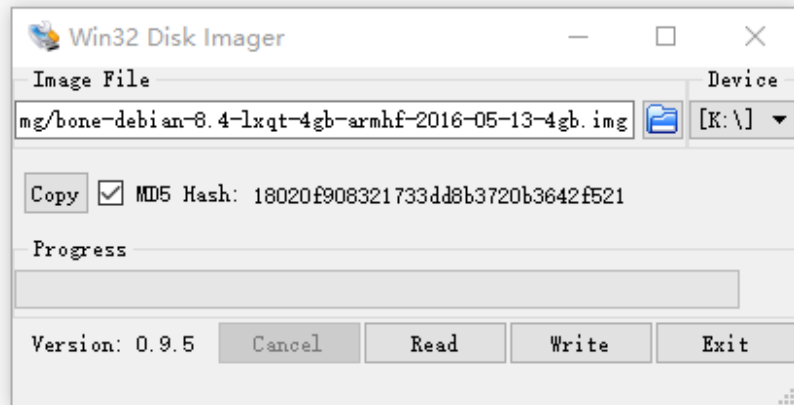
使用7-zip解压缩SD卡 `.img file`

步骤3：安装SD卡编程工具

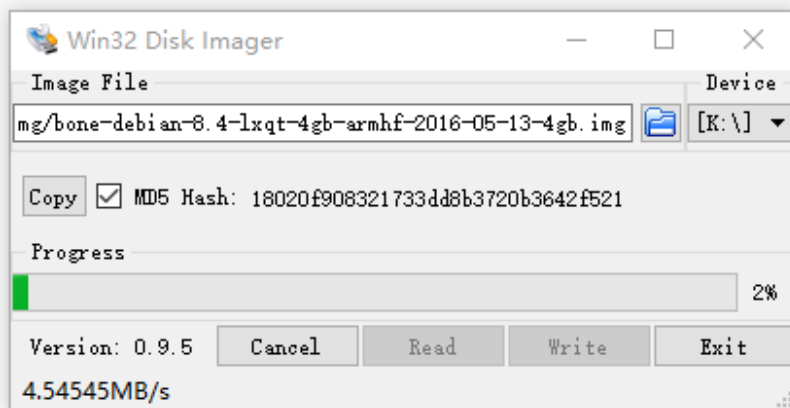
下载并安装 [Windows的图像编写器](#)。确保下载的版本正确。

步骤4： 将图像写入SD卡

首先需要有一个 SD 适配器将 microSD 卡连接到电脑。 然后使用软件 Image Write for Windows 将解压缩的图像写入 SD 卡。



点击 写 按钮，然后开始。



!!!Note * 您可能会看到有关损坏设备的警告。 只要你指定了你的 SD 卡，这将是问题。 * 此时您不应该将 BeagleBone 连接到您的计算机。 * 此过程最多可能需要 10 分钟。

步骤5： 从 SD 卡启动您的电路板

将 SD卡插入（第一电源）板。 然后板子将从 SD卡启动。

!!!Note 如果您不需要将图像写入您的机载 eMMC ， 则无需阅读本章最后一章。 否则请看看下面。

如果您希望将图像写入您的板载 eMMC ， 则需要启动板并修改文件。

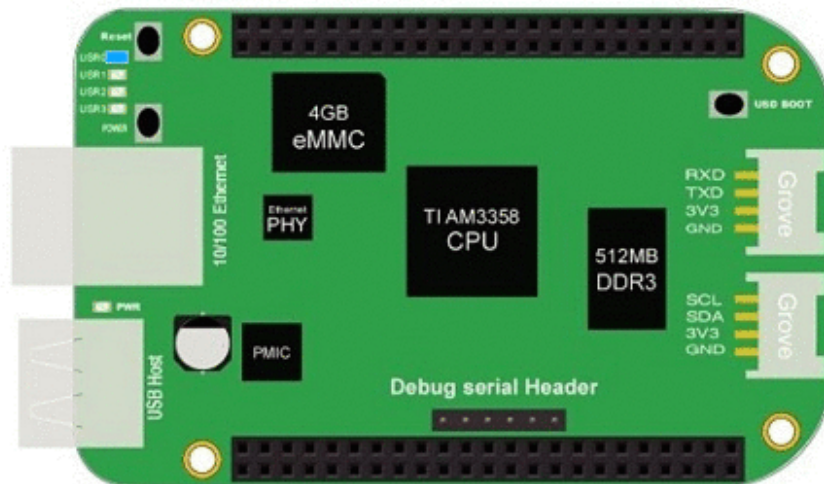
在 **/boot/uEnv.txt**:

```
##enable Generic eMMC Flasher:
##make sure, these tools are installed: dosfstools rsync
#cmdline=init=/opt/scripts/tools/eMMC/init-eMMC-flasher-v3.sh
```

改变为:

```
##enable Generic eMMC Flasher:
##make sure, these tools are installed: dosfstools rsync
cmdline=init=/opt/scripts/tools/eMMC/init-eMMC-flasher-v3.sh
```

那么你会发现 4 个用户的 led 灯如下:



!!!Note 如果没有找到上面的跟踪灯，请先关闭电源再重新打开电源。

当闪烁完成时，所有 4 个 USRx LED 将熄灭。最新的 Debian 闪光灯在完成闪烁后将自动关闭电路板。最多可能需要 **10 分钟**。关闭电源，取出 SD 卡，再次接通电源即可完成。

程序 Grove 模块与 Mraa 和 UPM

我们提供了 Mraa 库和 UPM 库，使开发人员和传感器制造商能够轻松地将其传感器和执行器安装到支持的硬件之上，并允许通过高级语言和结构控制实现通信。

什么是 Mraa 和 UPM ?

Mraa 是一个能够使用 C / C ++ 库，与 Python, Javascript 和 Java 的平台，使 BBG, BBGW 和其他平台上的 I / O 接口能够通过 API 兼容。使用 Mraa 则不需要您指定到特定的硬件，在运行时就已经完成了板子的检测，您马上就可以创建在支持平台上运行的代码。UPM 是使用 MRAA 的传感器的高级存储库。尽管有些传感器组可能不是互连的，但是每个传感器能够链接到 MRAA。每个传感器都包含一个允许与其进行连接的端口。

安装和更新

Mraa 和 UPM 已经安装在 BBGW 的系统中，所以 你不需要安装 但是，如果要更新库或要升级库，请使用 `apt-get update` 和 `apt-get upgrade` 请参考 <https://github.com/intel-iot-devkit/mraa> 和 <https://github.com/intel-iot-devkit/upm>。

Mraa 例子

- 点亮一个led

```
import mraa
import time
#mraa.gpio60 = P9_14 = GPIO_50
led = mraa.Gpio(60)
led.dir(mraa.DIR_OUT)

while True:
    led.write(1)
    time.sleep(1)
    led.write(0)
    time.sleep(1)
```

- Grove - PIR 传感器

```
import mraa
import time
#mraa.gpio73 = P9_27 = GPIO_115
pir = mraa.Gpio(73)
pir.dir(mraa.DIR_IN)

while True:
    print (pir.read())
    time.sleep(1)
```

- Grove - Rotary Angle 传感器

```
import mraa
import time
#mraa.aio1 = AIN0
rotary = mraa.Aio(1)

while True:
    print(rotary.read())
    time.sleep(1)
```

- 更多教程

[Grove - 3-Axis Digital Accelerometer\(±16g\)](#) [Grove - Variable Color LED](#)

[Grove - Mini Fan](#) [Grove - PIR Motion Sensor](#) [Grove - Rotary Angle Sensor](#)

[Grove - Relay](#) [Grove - Sound Sensor](#)

[Grove - OLED Display 0.96"](#) [Grove - Light Sensor](#) [Grove - Temperature Sensor](#)

Grove - GPS Grove - Button(P) Grove - Buzzer Grove - RTC v2.0

Mraa Map for BBGW

BBGW Mraa Gpio					
Mraa	phy	GPIO	Mraa	phy	GPIO
7	P8_07	GPIO_66	42	P8_42	GPIO_75
8	P8_08	GPIO_67	43	P8_43	GPIO_72
9	P8_09	GPIO_69	44	P8_44	GPIO_73
10	P8_10	GPIO_68	45	P8_45	GPIO_70
13	P8_13	GPIO_23	46	P8_46	GPIO_71
19	P8_19	GPIO_22	57	P9_11	GPIO_30
27	P8_27	GPIO_86	59	P9_13	GPIO_31
28	P8_28	GPIO_88	60	P9_14	GPIO_50
29	P8_29	GPIO_87	61	P9_15	GPIO_48
30	P8_30	GPIO_89	62	P9_16	GPIO_51
31	P8_31	GPIO_10	63	P9_17	GPIO_5
32	P8_32	GPIO_11	64	P9_18	GPIO_4
33	P8_33	GPIO_9	67	P9_21	GPIO_3
34	P8_34	GPIO_81	68	P9_22	GPIO_2
35	P8_35	GPIO_8	69	P9_23	GPIO_49
36	P8_36	GPIO_80	70	P9_24	GPIO_15
37	P8_37	GPIO_78	71	P9_25	GPIO_117
38	P8_38	GPIO_79	72	P9_26	GPIO_14
39	P8_39	GPIO_76	73	P9_27	GPIO_115
40	P8_40	GPIO_77	87	P9_41	GPIO_20
41	P8_41	GPIO_74	88	P9_42	GPIO_7

BBGW Mraa I2C			
Mraa	I2C	PIN	FUN
0	I2C1	P9_17	I2C1_SCL
		P9_18	I2C1_SDA
1	I2C2	P9_19	I2C2_SCL
		P9_20	I2C2_SDA

BBGW Mraa PWM		
Mraa	PWM	PIN
68	EHRPWM0A	P9_22
67	EHRPWM0B	P9_21
36 60	EHRPWM1A	P8_36 P9_14
62 34	EHRPWM1B	P9_16 P8_34
19 45	EHRPWM2A	P8_19 P8_45
13 46	EHRPWM2B	P8_13 P8_46
88	ECAPPWM0	P9_42

BBGW Mraa ADC	
Mraa	ADC
1	AIN0
2	AIN1
3	AIN2
4	AIN3
5	AIN4
6	AIN5
7	AIN6

BBGW Mraa UART				
MRAA	PIN	FUN	UART	DEV
0	P9_24	UART1_TXD	UART1	/dev/ttyO1
	P9_26	UART1_RXD		
1	P9_21	UART2_TXD	UART2	/dev/ttyO2
	P9_22	UART2_RXD		
2			UART3	/dev/ttyO3
3	P9_13	UART4_TXD	UART4	/dev/ttyO4
	P9_11	UART4_RXD		
4	P8_37	UART5_TXD+	UART5	/dev/ttyO5
	P8_38	UART5_RXD+		

参考资料和资源

参考

参考资料可以帮助您获得有关板子的更多信息。

- [BeagleBoard Main Page](#)
- [BeagleBone Green Wireless info at BeagleBoard page](#)
- [BeagleBoard Getting Started](#)
- [Troubleshooting](#)
- [Hardware documentation](#)
- [Projects of BeagleBoard](#)

资源下载

- [BeagleBone_Green_Wireless Schematic(pdf)]
(https://github.com/SeeedDocument/BeagleBone_Green_Wireless/tree/master/resources/BeagleBone_Green_Wireless_V1.0_SCH_20160314.pdf)