

# BananaPi uses DVK-511

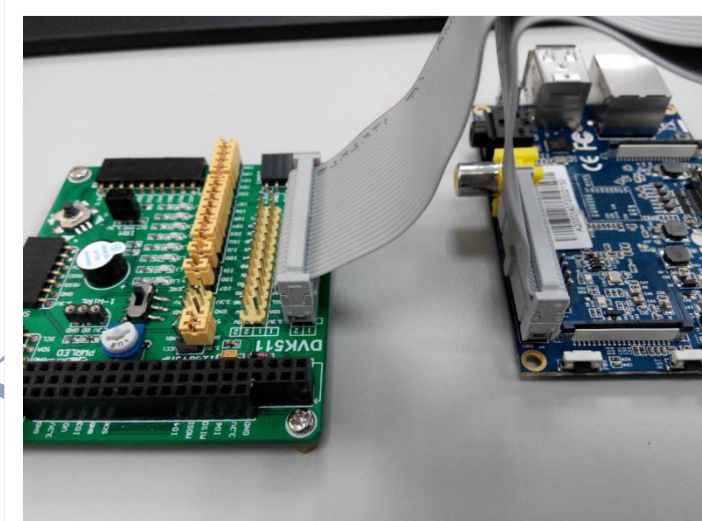
## Buzzer

By Justin Chen

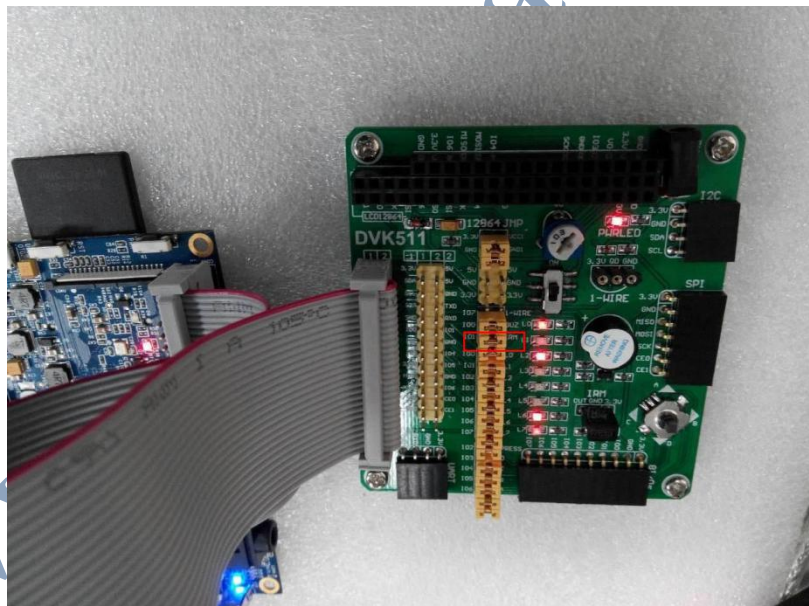
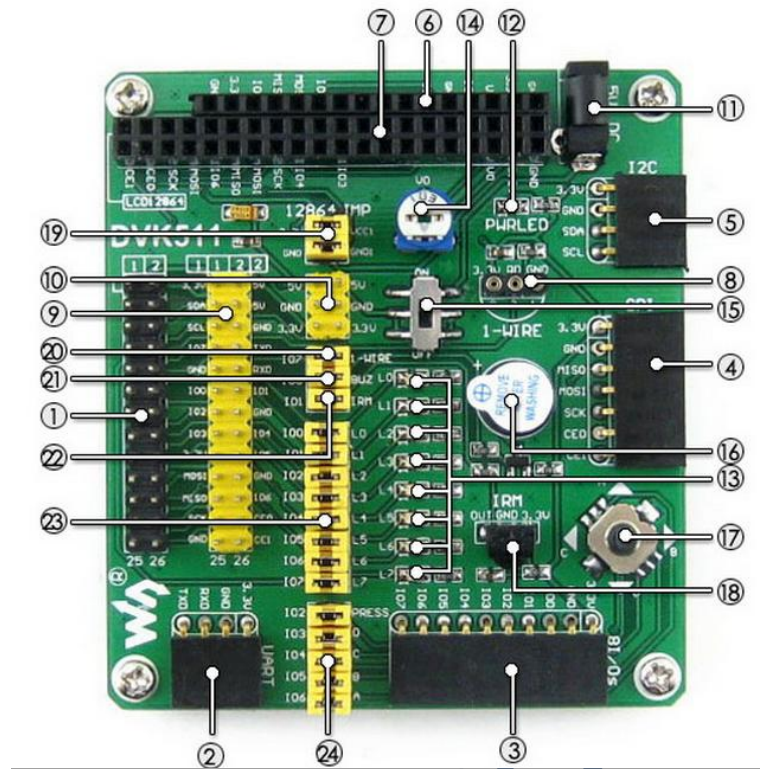
1. Please download the customized Raspbian Image for BananaPi from our website <http://www.bananapi.com> For how to burn the images to the SD card, please go to <http://www.bananapi.com/index.php/download?layout=edit&id=42>
2. The Image burn in SD card has preload the customized WiringPi Lib before, if download WiringPi Lib by yourself, you will need to modify it, otherwise it can't use; WiringPi Lib can find in /opt/gpio-lib.

```
pi@bananapi: /opt/gpio-lib
File Edit Tabs Help
pi@bananapi /opt/gpio-lib $ ls
RPi.GPIO-0.5.5 ScratchGPIO5 WiringBPi_Beta_V2.0
```

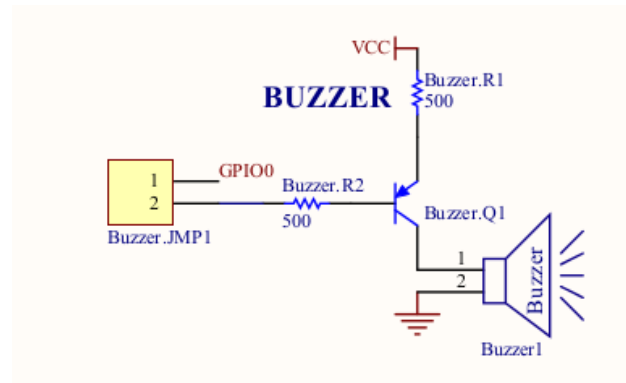
3. Please connect the BananaPi to the interface of DVK511.



4. The Jack No.16 of DVK511 is the Buzzer, and the jack NO.21 is the jump of it. Please do not pull it out while the testing or the Buzzer would be invalid.

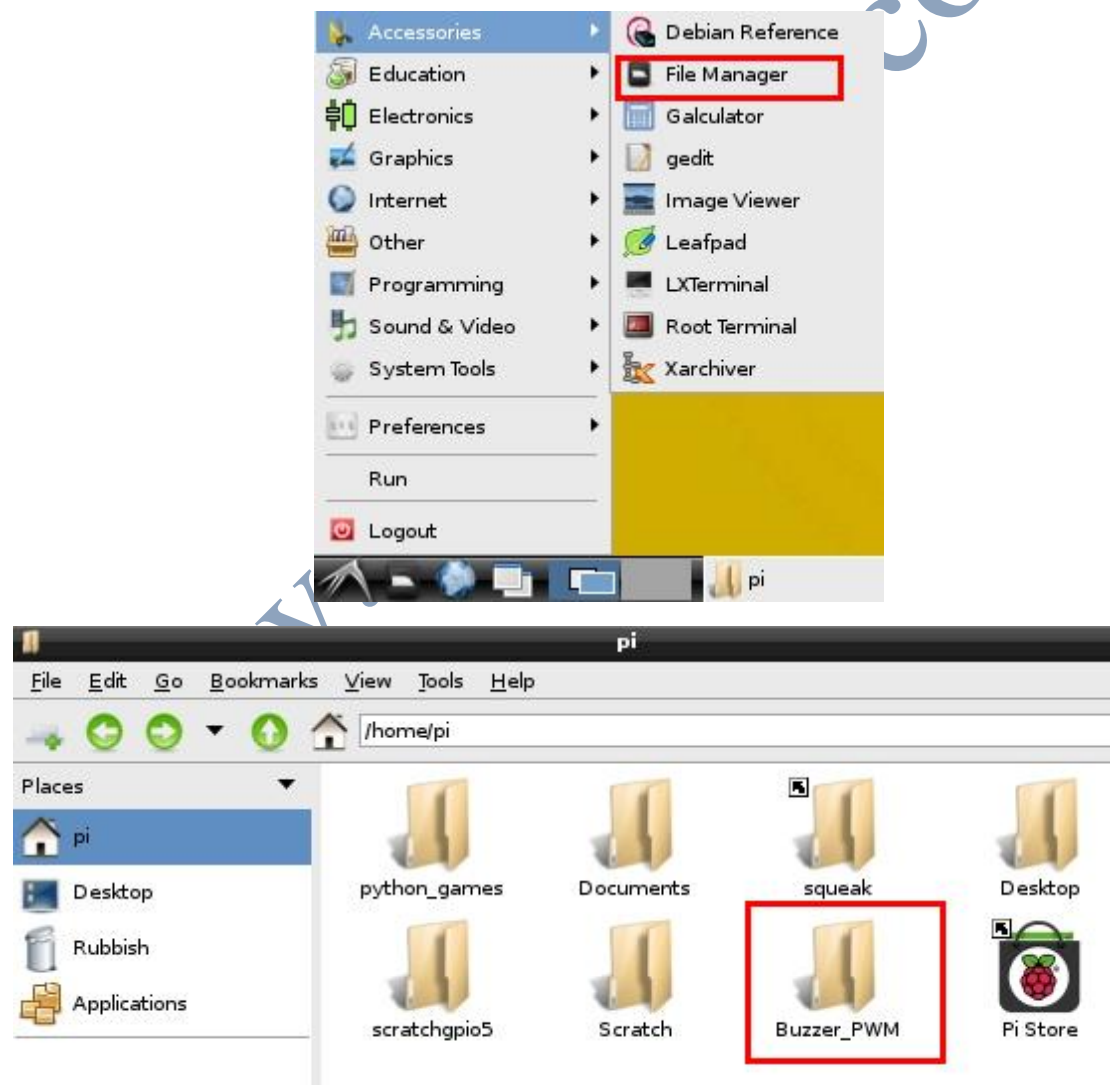


5. Check the Buzzer Map to find the corresponding point for each PIN.



Above picture showed Buzzer PIN's corresponding table

6. Use the Buzzer sample code to check the functionality, outset File Manager and copy the Buzzer\_PWM to the home dictionary.



Then outset LXTerminal,switch it to the Buzzer-PWM Folder.

```
pi@bananapi ~ $ cd /home/pi/Buzzer_PWM
```

CompileBuzzer\_PWM

```
pi@bananapi ~/Buzzer_PWM $ make
```

Run the code and outset the Buzzer.

```
pi@bananapi ~/Buzzer_PWM $ sudo ./buz_pwm
```

7. At the end, find out whether there are sounds heard from the Buzzer.
8. The GPIO PIN can also be used to control the Buzzer, See the No.5 Buzzer images you can know that the name of Buzzer GPIO is GPIO0,Corresponding Banana PI GPIO is GPIO17(run gpio readall order to check corresponding GPIO PIN)

```
pi@bananapi ~ $ gpio readall
```

wiringPi	GPIO	Phys	Name	Mode	Value
0	17	11	GPIO 0	IN	High
1	18	12	GPIO 1	IN	High
2	27	13	GPIO 2	IN	High
3	22	15	GPIO 3	IN	High
4	23	16	GPIO 4	IN	Low
5	24	18	GPIO 5	IN	Low
6	25	22	GPIO 6	IN	High
7	4	7	GPIO 7	IN	High
8	2	3	SDA	ALT5	Low
9	3	5	SCL	ALT5	Low
10	8	24	CE0	IN	Low
11	7	26	CE1	IN	Low
12	10	19	MOSI	IN	Low
13	9	21	MISO	IN	Low
14	11	23	SCLK	IN	Low
15	14	8	TxD	ALT0	High
16	15	10	RxD	ALT0	Low
17	28	3	GPIO 8	IN	High
18	29	4	GPIO 9	ALT4	Low
19	30	5	GPIO10	OUT	High
20	31	6	GPIO11	ALT4	Low

9. Outset LXTerminal,run order to relative paths.

```
pi@bananapi ~ $ cd /sys/class/gpio/
```

EnableCorrespondingGPIO PIN

```
pi@bananapi: /sys/class/ $ echo 17 > export
```

At this time can add a gpio17 contents under the gpio contents.

```
pi@bananapi /sys/class/gpio $ ls
export gpio17 gpiochip1 unexport
```

Switch to gpio17

```
pi@bananapi: /sys/class/gpio $ cd gpio17
```

You can see a direction contents down below,need to change the setpoint of this contents.

```
pi@bananapi: /sys/class/gpio/gpio17 $ ls
active_low device direction edge power pull subsystem uevent value
```

Switch root Competence to control GPIO

```
pi@bananapi: /sys/class/gpio/gpio17 $ sudo su
```

Use cat order to check the setpoint of direction contents

```
root@bananapi: /sys/devices/platform/gpio-sunxi/gpio/gpio17# cat direction
in
```

You need to change direction setpoint,and then can use GPIO PIN to control the Buzzer.



```
pi@bananapi: /sys/class/gpio/gpio17
File Edit Tabs Help
root@bananapi:/sys/devices/platform/gpio-sunxi/gpio/gpio17# echo out > direction
```

After change the direction setpoint, Use the cat order to check the direction contents whether change succeeded.

```
pi@bananapi: /sys/class/gpio/gpio17
File Edit Tabs Help
root@bananapi:/sys/devices/platform/gpio-sunxi/gpio/gpio17# cat direction
out
```

Change the Number of value data to control the status of Buzzer.

➔echo 1 > value //Close the Buzzer

➔echo 0 > value //Outset the Buzzer

www.banana-pi.com