

# How to read firmware from YUNSIM simulator stick using Raspberry Pi

## What we need

### Hardware:

Raspberry Pi <https://www.raspberrypi.org/>

### Software:

Wiring Pi <http://wiringpi.com/download-and-install/>

CC2531 flash tool [https://github.com/jmichault/flash\\_cc2531](https://github.com/jmichault/flash_cc2531)

## Preparation

### Update Raspberry Pi

```
sudo apt update && sudo apt upgrade -y
```

### Installing WiringPi

```
sudo apt install wiringpi -y
```

Check if WiringPi was installed successful:

```
gpio -v
```

It should respond with gpio version and some information about the RaspberryPi.

```
pi@testlite:~ $ gpio -v
gpio version: 2.50
Copyright (c) 2012-2018 Gordon Henderson
This is free software with ABSOLUTELY NO WARRANTY.
For details type: gpio -warranty

Raspberry Pi Details:
  Type: Model B, Revision: 02, Memory: 512MB, Maker: Egoman
  * Device tree is enabled.
  *--> Raspberry Pi Model B Rev 2
  * This Raspberry Pi supports user-level GPIO access.
pi@testlite:~ $ █
```

### Installing git to be able to download tools

```
sudo apt install git -y
```

## Installing CC2531 flash tool for LINUX

```
git clone https://github.com/jmichault/flash_cc2531.git
```

This creates a sub directory for the tool. Go into this directory and check if all is there:

```
cd flash_cc2531
```

```
ls -al
```

```
pi@testlite:~ $ cd flash_cc2531/
pi@testlite:~/flash_cc2531 $ ls -al
insgesamt 364
drwxr-xr-x 3 pi pi 4096 Sep  6 18:08 .
drwxr-xr-x 9 pi pi 4096 Sep  6 17:20 ..
-rwxr-xr-x 1 pi pi 29992 Sep  6 17:20 cc_chipid
-rw-r--r-- 1 pi pi 2192 Sep  6 17:20 cc_chipid.c
-rw-r--r-- 1 pi pi 15385 Sep  6 17:20 CCDebugger.c
-rw-r--r-- 1 pi pi 2389 Sep  6 17:20 CCDebugger.h
-rwxr-xr-x 1 pi pi 30004 Sep  6 17:20 cc_erase
-rw-r--r-- 1 pi pi 2249 Sep  6 17:20 cc_erase.c
-rwxr-xr-x 1 pi pi 31576 Sep  6 17:20 cc_read
-rw-r--r-- 1 pi pi 4107 Sep  6 17:20 cc_read.c
-rwxr-xr-x 1 pi pi 38332 Sep  6 17:20 cc_write
-rw-r--r-- 1 pi pi 11137 Sep  6 17:20 cc_write.c
drwxr-xr-x 8 pi pi 4096 Sep  6 17:20 .git
-rw-r--r-- 1 pi pi 10 Sep  6 17:20 .gitignore
-rw-r--r-- 1 pi pi 35149 Sep  6 17:20 LICENSE
-rw-r--r-- 1 pi pi 510 Sep  6 17:20 Makefile
-rw-r--r-- 1 pi pi 3087 Sep  6 17:20 README.md
-rw-r--r-- 1 pi pi 60048 Sep  6 18:10 yunsim1.hex
-rw-r--r-- 1 pi pi 60048 Sep  6 18:05 yunsim.hex
pi@testlite:~/flash_cc2531 $
```

The green marked executable are the commands we need for check / read / erase / write firmware to CC2531 chips.

Remark: Before flashing a CC2531 we must always completely erase the flash memory (command below in the procedure).

We are ready now.

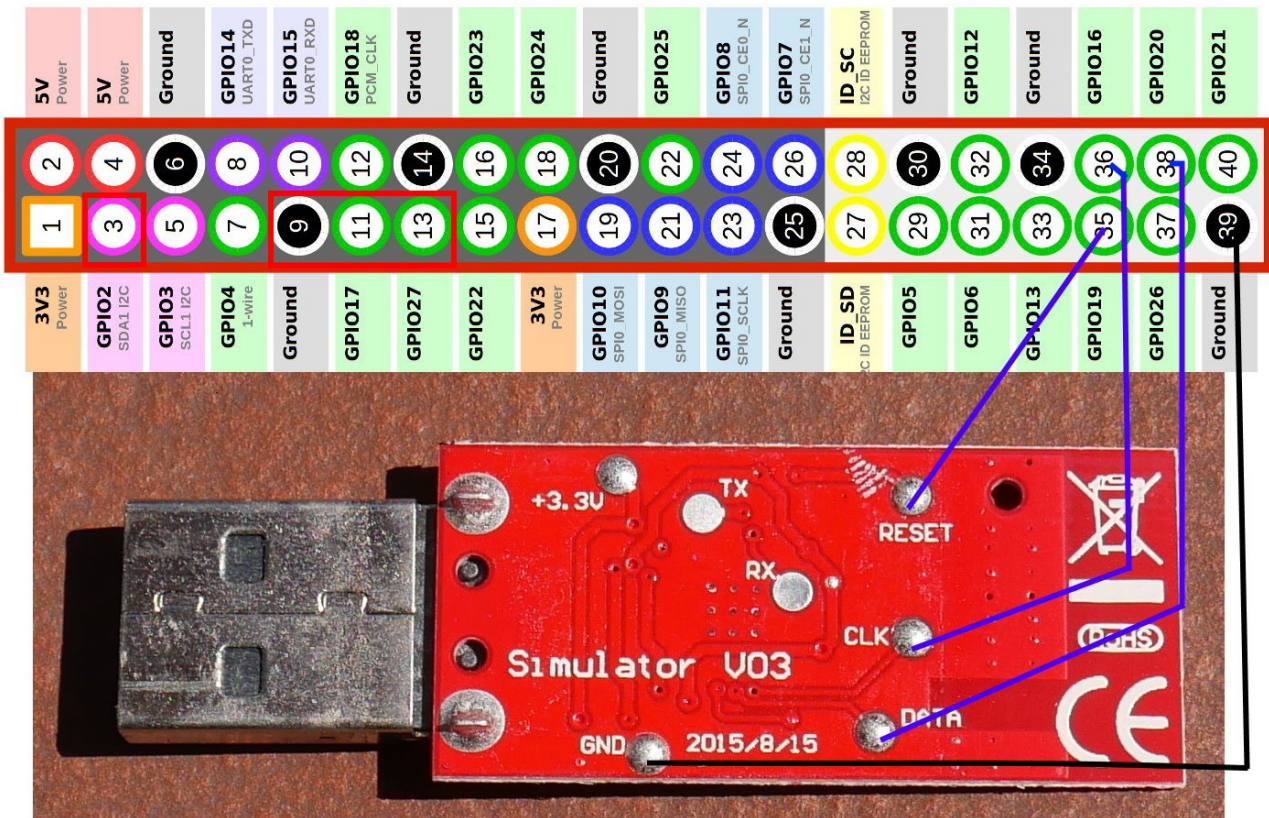
# Connect simulator stick to Raspberry Pi

You need to solder four wires to the stick: GND, RESET, DATA, CLK.

YUNSIM – flashing pads:

YUNSIM pads	Flash tool	Raspi2 GPIO pins	Raspi3 GPIO pins
GND	GND	9 Ground	39 Ground
CLK	DC	11 GPIO17	36 GPIO16
DATA	DD	13 GPIO27	38 GPIO20
RESET	RST	3 GPIO2	35 GPIO19

Connect the wires with the GPIO port of Raspberry Pi. Insert the simulator stick into the USB port of the Raspberry Pi.



Test connection:

```
./cc_chipid
```

Alternative command for Raspberry Pi 2 (with smaller GPIO port connector) to test:

```
./cc_chipid -r 8 -c 0 -d 2
```

If all is OK it returns ID = b524

## Read the firmware

```
./cc_read yunsim.hex
```

Alternative command for Raspberry Pi 2 (with smaller GPIO port connector):

```
./cc_read -r 8 -c 0 -d 2 yunsim.hex
```

It takes some time. As result you should get a file „yunsim.hex“ with 60048 bytes.

MD5 sum should be **92c2f74444819a76235d906daab2781d**.

Done. This file can be flashed now to a cheap ZigBee USB dongle.