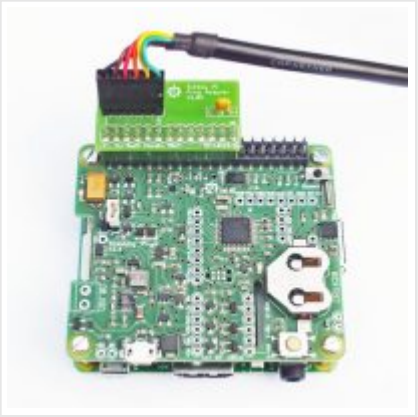
Configuring The GPIO Serial Port On Raspbian Jessie Including Pi 3



Jon Watkins on May 29, 2016 in **Raspberry Pi**

Contents [hide]

This used to be relatively straightforward, but with move from Raspbian Wheezy to Raspbian Jessie, things changed. Add to this, the new Raspberry Pi 3 with new hardware and the whole thing became a bit of a Dog's Breakfast and issues with the Bluetooth to boot.

It suddenly got very confusing.

Following lots of wasted time, I've noted down what I think I know so far in this post. With luck you can have your cake and eat it: use the serial port on a Raspberry Pi 3 and use the

NOTE: You will need the latest firmware May 2016 (2016-05-10) or later for this to work. Should this change again, I'll update this post.

History

Before I dive into the configuration, it's worth taking a moment for a little history and orientation about the serial port on the Raspberry Pi.

If you're a bit old school like me, you'd be expecting to find something called COM1 or similar on a header. In Raspberry Pi / Linux land this COM1 equivalent is found on pins 14 and 15 of the GPIO header and is called **/dev/ttyAMA0** (obvious, right?).

So in Raspberry Pi land, you can use the serial port as a terminal to log in, which is useful if you don't have a network connection to hand. You can connect to another computer via their serial ports and run a terminal emulator on the other computer and you'll get a login prompt.

By default the Raspberry Pi uses the serial port for this "console" login and via a software service called "getty".

Using the serial port with other hardware

So that's the 'normal" configuration of the serial port, but serial ports are very useful things. What if we want to use the serial port to get data from a GPS card or program an arduino? In this case we need to disable the console login so that we alone get control of the port. Easy right? Yes and no. There is a big elephant in the room and he's called Raspberry Pi 3.

Before we can describe using the serial port, we have to talk about Raspberry Pi 3, which throws a great big spanner in the works as far as serial ports are concerned.

Raspberry Pi 3

Raspberry Pi 3's are great little beasts, and add Bluetooth, yay! However, in order to use the Bluetooth correctly the **/dev/ttyAMA0** has been "stolen" from the GPIO header and an inferior second one has been substituted in it's place. No-one will ever know! Unfortunately **/dev/ttyAMA0** was a hardware serial port (uart) and high performance (hence it was nabbed for the Bluetooth) and the second port is partly software and a bit

The second serial port you will see referred to as the “mini uart” and lives at **/dev/ttyS0**. It also calculates it’s bit timing’s from the CPU cores frequency and if the CPU is under heavy load it can corrupt the serial communications. Not good.

In order to work around this, many people “fix” the CPU core frequency so that the serial port is stable. This comes at a slight loss in performance (though normally not noticeable). I’ll describe how you do this in the next section.

the way, it’s not all bad for the change of serial port on the Raspberry Pi 3. The Arduino expects the serial communications to be on **/dev/ttyS0** so you have no work to do to up the serial ports across. Yay!

summarise the ports on a Raspberry Pi 3 and be crystal clear:

ev/ttyAMA0 -> Bluetooth
ev/ttyS0 -> GPIO serial port.

ou stick with these as is, your Bluetooth will work as nature intended AND you can use erial port over the GPIO (there is a way of swapping the serial ports around if you don’t nt to use the Bluetooth and I’ll cover that at the end of this post).

Enabling

There is yet another wrinkle in that in the latest Jessie releases (as of May 2016) the GPIO serial port is disabled by default. In order to enable it, edit **config.txt**:

```
$ sudo nano /boot/config.txt
```

and add the line (at the bottom):

```
enable_uart=1
```

As of May 2016 this will also lock the cpu core frequency for you so there’s nothing else you need to do (If you aren’t convinced and you really like to belt and braces it the command is: **core_freq=250** which you add to the same file aswell).

Reboot for the changes to take effect.

Serial Aliases

On the Raspberry Pi 3 the second serial port is called **/dev/ttyS0** and is by default mapped to the GPIO pins 14 and 15. So immediately, if you have code that references **/dev/ttyAMA0** you're going to have problems and things aren't going to work.

You could go through your code and replace **ttyAMA0** with **ttyS0** and that should work. However, if you find yourself use the same SD card on a Raspberry Pi other than a rpi3 ur code won't work again.

order to try and get around this the Foundation have introduced a *serial port alias* (as of ıy 2016 – 2016-05-10). Thus you have serial ports: serial0 and serial1 (rpi3). The spberry Pi kernel sorts out where these point to depending on which Raspberry Pi you e on. Thus on a Raspberry Pi 3 serial0 will point to GPIO pins 14 and 15 and use the **ini-uart**" aka **/dev/ttyS0**. On other Raspberry Pi's it will point to the hardware UART and **ev/ttyAMA0**.

find out where it is pointing you can use the command:

```
ls -l /dev
```

```
crw-rw-r-- 1 root root     10,  58 May 28 12:14 rfkill
lrwxrwxrwx 1 root root           5 May 28 12:14 serial0 -> ttyS0
lrwxrwxrwx 1 root root           7 May 28 12:14 serial1 -> ttyAMA0
drwxrwxrwt 2 root root          40 May 28 12:15 shm
```

Default Raspberry PI 3 serial port aliases

```
crw-rw-r-- 1 root root     10,  58 May 28 13:59 rfkill
lrwxrwxrwx 1 root root           7 May 28 13:59 serial0 -> ttyAMA0
drwxrwxrwt 2 root root          40 May 28 14:05 shm
```

Default Raspberry PI 2 serial port aliases

So where possible refer to the serial port via it's alias of "serial0" and your code should work on both Raspberry Pi 3 and other Raspberry Pi's.

Disabling the Console

If you are using the serial port for anything other than the console you need to disable it. This will be slightly different depending on whether you are running a Raspberry Pi 3 or

For non Raspberry Pi 3 machines, remember it's **/dev/ttyAMA0** that is linked to the getty (console) service. So you need to perform this command from a terminal window:

```
$ sudo systemctl stop serial-getty@ttyAMA0.service
$ sudo systemctl disable serial-getty@ttyAMA0.service
```

The "disable" will stop it loading in the future.

For Raspberry Pi 3's the command is similar but referencing **/dev/ttyS0**:


```
sudo systemctl stop serial-getty@ttyS0.service
sudo systemctl disable serial-getty@ttyS0.service
```


u also need to remove the console from the cmdline.txt. If you edit this with:

```
sudo nano /boot/cmdline.txt
```

u will see something like:

```
c_otg.lpm_enable=0 console=serial0,115200 console=tty1 root=/dev/mmcblk0p2 rootfstyp
ext4 elevator=deadline fsck.repair=yes root wait
```

nove the line: console=serial0,115200 and save and reboot for changes to take effect.

Swapping the Serial Ports on Raspberry Pi 3

What if you don't want to use the Bluetooth and you want that high performance **/dev/ttyAMA0** back on the GPIO? Well you can do this and the way you do this is via a device overlay called "**pi3-miniuart-bt**" i.e. use the mini-uart (**/dev/ttyS0**) for Bluetooth (you may get some loss of performance on your Bluetooth though).

You can also just disable the Bluetooth all together by using another overlay "**pi3-disable-bt**". In both cases if you can find out more of what they do here: **/boot/overlays/README**

To use add the following line to the /boot/config.txt

```
$ sudo nano /boot/config.txt
```

and add:

```
dtoverlay=pi3-miniuart-bt
```

You can check that it has worked by:

```
$ ls -l /dev
```

and you'll see something like this:

```
crw-rw-r-- 1 root root     10,  58 May 27 22:04 rfkill
lrwxrwxrwx 1 root root           7 May 27 22:04 serial0 -> ttyAMA0
lrwxrwxrwx 1 root root           5 May 27 22:04 serial1 -> ttyS0
drwxrwxrwt 2 root root          40 May 27 22:04 shm
```

Swapped Raspberry PI 3 serial port aliases

Related Posts:

- Introducing Sleepy Pi 2
- Project: ARUPi – A Low-Cost Automated Recording Unit for Soundscape Ecologists
- Project: Remote Water Monitoring With A Solar Powered Raspberry Pi
- Using Sleepy Pi With A Raspberry Pi 2 and A+
- Project: Bird Feeder Tweeter by Manifold

Raspberry Pi

‹ Introducing Sleepy Pi 2

67 Responses to Configuring The GPIO Serial Port On Raspbian Jessie Including Pi 3

mva August 23, 2017 at 2:15 am #

REPLY

Miles August 11, 2017 at 7:46 am #

REPLY

Like so many others I spent hours trying to get my Zero and 3 to talk to a serial device, you saved my sanity and even more hours of head banging. Thank you, thank you.

Jon Watkins August 11, 2017 at 4:24 pm #

REPLY

Thanks you!

Stephen Mann August 8, 2017 at 5:04 pm #

REPLY

Thankyou Thankyou thankyou.

After spending days trying to get my Pi to talk to an ESP8266 over the GPIO serial port, your post made using the GPIO serial port so amazingly clear.

Jon Watkins August 8, 2017 at 10:14 pm #

REPLY

Thanks!

Jhonny92 April 22, 2017 at 3:37 pm #

REPLY

Does this procedure work also with a Rapberry Pi 3 with Kali Linux? I am going to realize a project with it and I need the GPS signal to track its position.

Thanks 🙂

Bill April 13, 2017 at 4:25 pm #

REPLY

Great post, you should know this also applies to the zero W!

Thanks for letting us know!

Madhur March 30, 2017 at 1:50 pm #

REPLY

Hello!
I'm using raspi 3 and we are trying to send data from an rfid reader(EM18) over the ttl to USB converter.

We are using the following code:

```
import serial #import serial module

def read_rfid ():
ser = serial.Serial ("/dev/ttyAMA0") #Open named port
ser.baudrate = 9600 #Set baud rate to 9600
data = ser.read(12) #Read 12 characters from serial port to data
ser.close () #Close port
return data #Return data

id = read_rfid () #Function call
print id
```

The code does not throw any errors but we dont get any output when the code runs. We have tried using ttyS0 and ttyUSB0 as well.

Please please help.

Madhur March 30, 2017 at 1:52 pm #

REPLY

And we are using Raspberry pi 3.

sravanthi March 10, 2017 at 5:30 am #

REPLY

i want to connect with the raspberry pi bluetooth n send data from the android to be read in a python script.how to access the bluetooth module?

chogyi March 3, 2017 at 8:01 am #

REPLY

Jon Watkins March 3, 2017 at 1:42 pm #

REPLY

First thing to check is the wiring, have you got the RX pin connected / to the right pin?

chogyi March 6, 2017 at 2:53 am #

REPLY

i have already check. It is all right. thz for answer. pls say me next advise.

Daniel March 12, 2017 at 3:04 pm #

REPLY

My serial port ttyS0 on pi 3 (/dev/serial0) was not working due to permission error when I try to control with python. Investigation show of the ttyS0 was not on the group dialout.

Adding this command has fixed the problem:

sudo systemctl mask serial-getty@ttyS0.service

Rayn March 2, 2017 at 2:13 am #

REPLY

If on a Windows box, be sure the Prolific USB-to-Serial device is set to 115200, along with your PUTTY session. I had the garbled text issue on both a Rasp Pi 3 and a Rasp Pi 2. Once I changed my device speed in device manager, I was getting clear text over my serial session.

Thank you for the additional details! 🙂

Jon Watkins March 3, 2017 at 1:41 pm #

REPLY

Thanks for that extra little nugget!

It works. Thank you so much! I've been stuck with this for days.

Luc Eeckelaert February 14, 2017 at 11:42 am #

REPLY

Great post, thank you!

Ilker January 25, 2017 at 6:10 am #

REPLY

Thanks, this was a very helpful explanation. Since it was quite a bit of hassle to do this manually each time, I scriptized it. On Raspberry Pi3, the script simply disables Bluetooth. It is available on GitHub: https://github.com/itemir/raspberry_pi_utils

Comments welcome.

Jon Watkins January 25, 2017 at 8:31 am #

REPLY

Wow, thanks for sharing that!

Yann January 16, 2017 at 3:41 pm #

REPLY

For some reason, the serial ports swapping doesn't work on my RPI3.
I must have done Something wrong but I can't figure out what.
When I type ls -l /dev, ttyS0 and ttyAMA remain unchanged.
RPi gurus please advise.
Regards,
Yann

Espen E January 16, 2017 at 12:39 pm #

REPLY

amazing

i was searching the hold wide web yseterday, and found "jolly" explenations to get this working.

Why did i not find this yesterday.

everything explained, I love you!!!!

Jon Watkins January 16, 2017 at 9:24 pm #

REPLY

…And I love you too!

Patrick Zorba January 13, 2017 at 5:13 am #

REPLY

Thank you so much Jon. I wish I read your article long time ago! Also thanks for explaining the PI3 issues clearly.

pierre January 12, 2017 at 8:32 am #

REPLY

thank you soooooo mucj

Ron January 10, 2017 at 8:57 am #

REPLY

this topic is discussed here.

https://www.raspberrypi.org/forums/viewtopic.php?f=28&t=165897&p=1069386&hilit=remove+splash#p1069386

Cheers Ron

Vicente F. December 31, 2016 at 6:27 pm #

REPLY

Guy, you deserve a big statue somewhere in the galaxy.

Gracias, gracias, gracias.

Jon Watkins January 3, 2017 at 9:53 pm #

REPLY

DJ_exx January 9, 2017 at 10:05 am #

REPLY

Hi,
I have a problem with serial port /dev/ttyAMA0 on Jessie (Raspberry Pi 2B)
During boot time it sends one byte (0xFF) to serial port which confuses the connected device (DS2480B 1-wire bus-master chip).

Console is disabled in raspi-config, uart is enabled in /boot/config.txt, gpio pins 14,15 in ALT0 mode.

On Wheezie everything worked OK.

Few years ago there was similar issue in Linux kernel https://github.com/raspberrypi/linux/issues/12 and also similar problem with "Uncompressing Linux kernel" message sent to all serial ports.

Gonzalo February 26, 2017 at 3:50 pm #

REPLY

do you find solution for 0xFF issue? please share your solution =)

Alex December 14, 2016 at 9:26 am #

REPLY

Thank you for posting this. I was going round in circles before reading this. This is the only article on the web that I could find that explains the changes in the PI3 is a simple and concise manner.

Bryan December 1, 2016 at 3:19 am #

REPLY

Has anyone attempted this yet on the Pi Zero? I'm finding that neither /dev/ttyAMA0 nor /dev/ttyS0 exist. I was able to follow the instructions successfully on both the Pi 2 and 3.

Many thanks!
Very good, simple and clear article!!

Ti November 9, 2016 at 1:31 am #

REPLY

Beautiful. Thank You.

Sean November 6, 2016 at 12:13 pm #

REPLY

Great Post, thank you for this,

Seth November 1, 2016 at 8:50 am #

REPLY

Thanks for this. I had some other problems as well. I also had to issue:
sudo systemctl disable hciuart
to prevent (I am told) the bt modem from trying to keep the port. Also beware that if you use raspi-config to turn off the console AFTER you have edited /boot/config.txt that it seems to also reset enable_uart=0. Too me a while to find that problem 😉

Thanks again

A follow on question. Have moved ttyAMA0 back to gpio14/15, and moved bt to miniuart, and have disabled console on serial as above. It all seems to wrk – except for one very peculiar thing. I am using this as a coprocessor for vision tasks an a flying quad. There is a serial link to the quad's flight controller that talks to ttyAMA0. The communications works perfectly on pi 0 – just need more computation. When moved to pi 3 – it works occasionally – BUT attaching flight controller TX to gpio15 frequently causes the pi to close all ssh sessions! there is plenty of power (tried separate power supplies, etc), the cable has been replaced and tested on pi 0.

So the question is what the heck could be happening on the uart rx pin that would cause the pi 3 to kick out active ssh connections? I am starting to guess that I just have a bad pi 3 – but thought I would ask first.

Thanks for any and all input

Wayne November 1, 2016 at 1:32 am #

REPLY

IS there a way to ID one of the native (Serial0 / Serial1) as "COMn"? I am wanting to run an application (AdvancedHMI) using mono on an R Pi 3, and the application's ModbusRTU comm driver is configured to address COM1, 2 or 3……..

harold October 22, 2016 at 5:21 pm #

REPLY

Amazing clear ! Struggle a full day and you post solved the tty issue in minutes.
Thank you. Thank you. Thank you…

Josiah Hamilton October 15, 2016 at 9:31 pm #

REPLY

thank you thank you thank you!

Maikel October 11, 2016 at 9:01 am #

REPLY

Thank you, helped me out a lot !

excellent – this was exactly what i needed and it worked a treat, thanks for putting this on the web!

Jon Watkins September 29, 2016 at 9:57 pm #

REPLY

Thanks!

jp herrenknecht September 24, 2016 at 9:52 am #

REPLY

Thanks a lot,

After 3 days of strugling with my rpi3, i found your article and it saved me with a microstack gps.
Your article is very clear and I finally understand the source of my problems.

Jon Watkins September 24, 2016 at 10:07 am #

REPLY

Thanks! Very much appreciated. 🙂

gavan September 24, 2016 at 2:17 am #

REPLY

This is a fantastic post, thanks so much for this, totally solved my problem

Jon Watkins September 24, 2016 at 10:07 am #

REPLY

Thanks for that 🙂

Erdogan Kurtur September 11, 2016 at 1:41 pm #

REPLY

Hi,

Great article, saved me a lot of trouble. Just noting on RPi3B v1.2 (jessie 8.0, 4.4.13-v7+), I had to disable serial getty using

Pi to MicroView arduino using direct rx-tx cable connection.

Thank you very much for this piece of gold

Jon Watkins September 11, 2016 at 7:24 pm #

REPLY

Erdogan,

Thanks for the tip!

Gj September 8, 2016 at 5:02 pm #

REPLY

which pin shall I use for the second serial port? Thanks

Jon Watkins September 11, 2016 at 7:28 pm #

REPLY

The serial port appears on the GPIO on GPIO 14 and GPIO 15. These are physically pins 8 (TXD) and 10 (RXD). These are the same pins on all RPi's. On the RPi 3 it's referred to as a second serial port.

Bret January 18, 2017 at 5:48 am #

REPLY

Ok i was wondering that too Jon. So if I'm using pin 10 (RX) and have disabled the BT then it should be using the fast UART?

ls -l /dev
…Shows me
serial0/TTYAMA0

I am getting output from my accelerometer over the RX pin which is good i just wanted to make sure this is the fastest RX available on the PI3

Do you have a write up for the pi2? Im trying to write at commands to an hm10 module connected to the gpio pins.

Jayson J Thomas August 29, 2016 at 8:46 am #

REPLY

Thanks so much for this write-up 🙂

Brian July 17, 2016 at 10:18 pm #

REPLY

I use the getty command '/sbin/getty -L ttyUSBupper 9600 vt100' , and success prompt to upper USB , but why I put in

/etc/inittab :

T1:23:respawn:/sbin/getty -L ttyUSBupper 9600 vt100

It just can not work while the Raspberry pi3 boot up ?

Brian

JHBR July 14, 2016 at 11:09 am #

REPLY

Thank you for collecting all this updated information in an informative post!

A lot if misleading information out there regarding how to use the UART that depends on which raspberry pi you're sporting and if you're using Jessie or an older version of rasbian.

JHBR July 22, 2016 at 10:32 am #

REPLY

Running my code on a RPi 3, I had troubles with receiving data on the UART unless I added the line to lock the CPU core frequency (core_freq=250): Adding the line to enable the UART (enable_uart=1) was not enough.

Kass July 7, 2016 at 6:43 pm #

REPLY

Peter June 30, 2016 at 9:06 pm #

REPLY

Thank you for collecting all this information here and making it straightforward and easy to follow.

Rock Lu June 28, 2016 at 10:53 am #

REPLY

Excellent post! make me clear ever than befor!

x92127 June 19, 2016 at 2:20 pm #

REPLY

Excellent write-up, thank you! This is exactly what I was looking for after 10 hours of banging my head against the wall.

ackbacks/Pingbacks

spberry Pi & OpenCV – UART Communication (Part 2) – Richard Arthurs – August 22, 2017
[…] First, follow the instructions here: https://spellfoundry.com/2016/05/29/configuring-gpio-serial-port-raspbian-jessie-including-pi-3/ […]

Motion Sensor PoC: BNO055 and Raspberry Pi subtleties – Hauke's Projects – March 21, 2017
[…] For a very good and complete description of the serial interface and its configuration on Raspberry Pi 3 read this article. […]

OpenStack part 9: Using a Raspberry Pi as out-of-band server management – Panic1 – February 19, 2017
[…] Information has been found here: Link. […]

Enable serial port on raspbian jessie | Ryan Martin, PhD – November 2, 2016
[…] More help: here here […]

Téléinfo EDF avec un Raspberry Pi – MagdiBlog – August 3, 2016
[…] Pour plus d'information sur ces changements : https://spellfoundry.com/2016/05/29/configuring-gpio-serial-port-raspbian-jessie-including-pi-3/ […]

Leave a Reply

Name (required)

Email (will not be published) (required)

Website

SUBMIT COMMENT

nformation

ookie Policy
rivacy Policy
Terms of Sale
Returns

Quick Links

Blog
Shop
Resources
Shipping

Currency conversions are estimated and should be used for informational purposes only. Note prices include UK VAT at 20%. This is not applicable outside the EU

GBP USD EUR