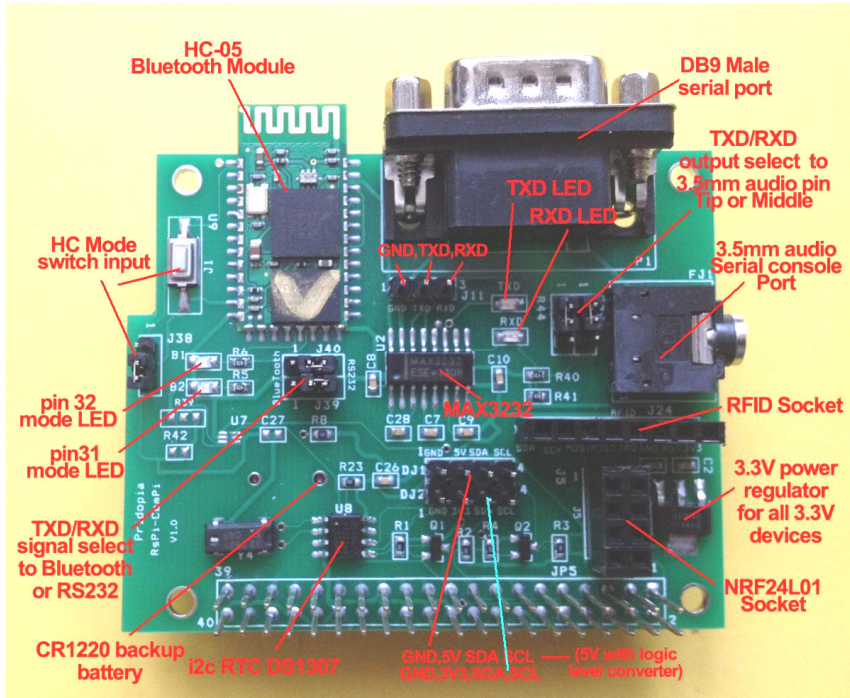


# Rs-Pi RTC & COM Port board User Manual



Raspberry Pi - RS232 COM & Serial console Board

1. provide DB9 male COM port
2. provide i2c DS1307 RTC with CR1220 backup battery
3. provide RFID socket ( SPI signal) / NRF24L01 socket (SPI signal)  
you can choose use RFID or NRF24L01 ( both use SPI signal)
4. Provide 2 extra i2c device input port, one for 3.3V, one for 5V (with Logic Level converter)
5. provide 3.3V power Regulator for 3.3V device ( RFID/ NRF24L01) i2c device
6. provide 3.5mm serial console cable socket,
7. provide jumper setting can change 3.5mm output TX, RX position.



First Install battery for RTC , "+" mark on top

RTC DS1307 - 68 in i2cdetect -y 0 or  
i2cdetect -y 1 for Rs-Pi V2 you will see 68 in the screen  
68 -> RTC DS1307

This requires a Raspberry Pi running a kernel with the RTC module and

DS1307 module included.

then, load up the RTC module by running

**sudo modprobe rtc-ds1307**

Then, as root (type in **sudo bash**) run

**echo ds1307 0x68 > /sys/class/i2c-adapter/i2c-0/new\_device** (if you have a rev 1 Pi)

**echo ds1307 0x68 > /sys/class/i2c-adapter/i2c-1/new\_device** (if you have a rev 2 Pi)

**hwclock -r** read time

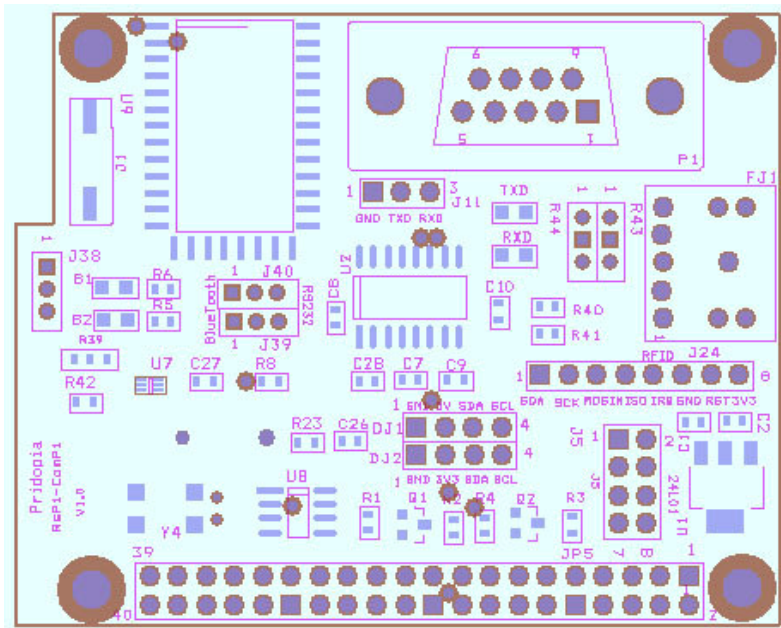
**hwclock -w** write time in RTC

**hwclock -s** write time in System

**hwclock --set --date="2013-08-21 08:00:12" --utc**

write in custom Time in RTC

you'll want to add the RTC kernel module to the /etc/modules list, so its loaded when the machine boots. Run **sudo nano /etc/modules** and add **rtc-ds1307** at the end of the file



```
COM37 - PuTTY
# /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
spi-bcm2708
i2c-bcm2708
i2c-dev
tmp102
rtc-ds1307
```

Then you'll want to create the DS1307 device creation at boot, edit  
/etc/rc.local by running  
**sudo nano /etc/rc.local**

and add **echo ds1307 0x68 > /sys/class/i2c-adapter/i2c-0/new\_device** before **exit 0**

```
COM37 - PuTTY
GNU nano 2.2.6 File: /etc/rc.local

#
# By default this script does nothing.

# Print the IP address
_IP=$(hostname -I) || true
if [ "$_IP" ]; then
    printf "My IP address is %s\n" "$_IP"
fi
echo ds1307 0x68 > /sys/class/i2c-adapter/i2c-0/new_device
echo tmp102 0x48 > /sys/class/i2c-adapter/i2c-0/new_device
exit 0
```

The necessary settings are:

- **Speed: 115200 baud**    **Data bits: 8**    **Stop bits: 1**
- **Parity: None**    **Flow control: None**

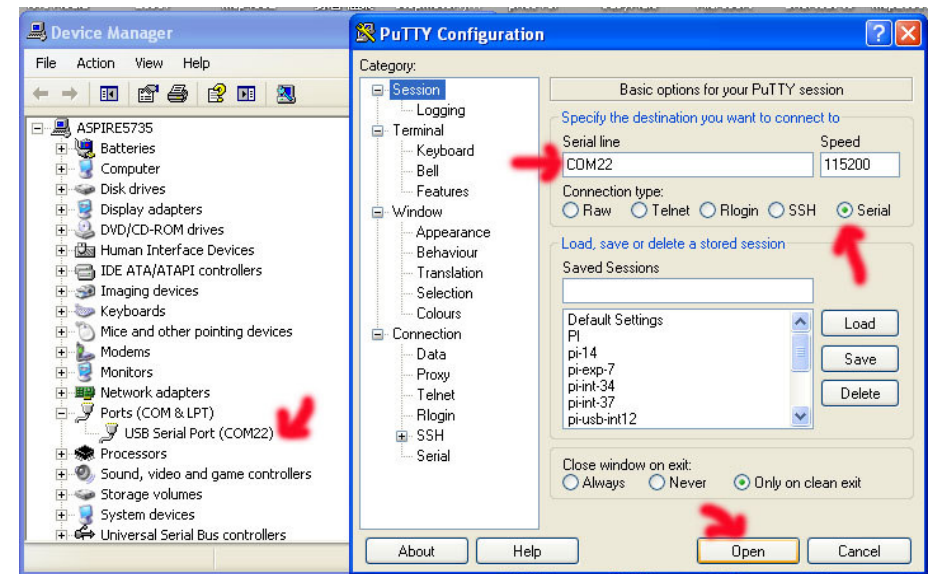
```
COM13 - PuTTY
[ 12.098988] Adding controls ..
[ 12.104479] Registering card ....
[ 12.118146] bcm2835 ALSA CARD CREATED!
[ 12.130070] ### BCM2835 ALSA driver init OK ###
[ 12.825626] Adding 131068k swap on /var/swapfile. Priority:-1 extents:4 across:147456k SS
[ 14.499551] fuse init (API version 7.17)
Debian GNU/Linux 6.0 raspberrypi ttyAMA0

raspberrypi login: pi
Password:
Last login: Thu Jan 1 01:00:41 BST 1970 on ttyAMA0
Linux raspberrypi 3.1.9+ #138 PREEMPT Tue Jun 26 16:27:52 BST 2012 armv6l

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Date and Time are unknown - using 2012-07-16 14:32 as an approximation
Correct the date and time using e.g: sudo date --set="2012-07-16 14:32"
pi@raspberrypi:~$
```

In PuTTY



Setting in Putty

## Raspberry Pi Serial Port Usage

The serial port on the Raspberry Pi is configured as default for console input/output. This allows you to login and interact with the Raspberry Pi via the serial port but you cannot use the serial port with your programs.

To use the serial port with other programs and hardware such as modems, arduino boards etc you need to disable the console login.

### To Disable Serial Port Login

You need to edit two files in order to use the serial port with your own programs.

When the Raspberry Pi boots, the bootup information is sent to the serial port. You can disable this by editing the **/boot/cmdline.txt** file

The contents of the file look like this

```
dwc_otg.lpm_enable=0 console=ttyAMA0,115200  
kgdboc=ttyAMA0,115200 console=tty1 root=/dev/mmcblk0p2  
rootfstype=ext4 elevator=deadline rootwait  
Remove all references to ttyAMA0 so the file looks like this:  
dwc_otg.lpm_enable=0 console=tty1 root=/dev/mmcblk0p2  
rootfstype=ext4 elevator=deadline rootwait
```

Save the file to save your changes.

The second file to edit is **/etc/inittab**

Edit using: **sudo nano /etc/inittab**

The **/etc/inittab** file has the command which enables the login prompt which needs to be disabled.

Near the end of the file will be a line similar to this:

```
respawn:/sbin/getty -L ttyAMA0 115200 vt100  
Disable this line by adding a # character to the beginning.
```

```
#respawn:/sbin/getty -L ttyAMA0 115200 vt100
```

Save the file.

You should then reboot your raspberry pi with the following command

**sudo shutdown -r now**

You can now use your serial port with other applications on your Raspberry Pi.

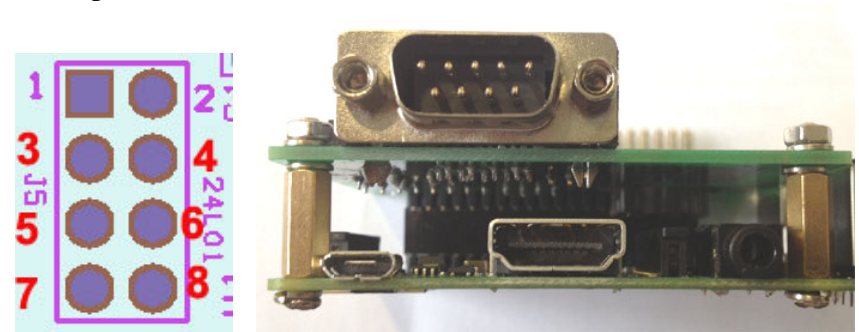
RFID socket

From pin1 (SDA,SCK,MOSI,MISO,IRQ,GND,RST,3V3)

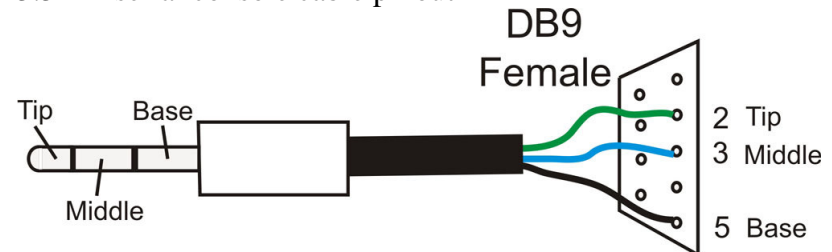


NRF24L01 socket

From pin1 (GND,3V3,GPIO25,CE0,SCLK,MOSI,MISO,X)



3.5mm serial console cable pinout

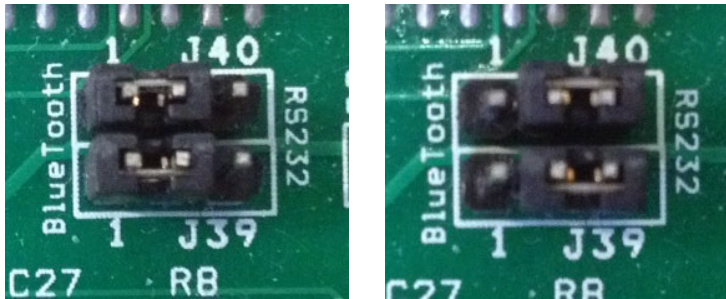


Detail information

<http://www.pridopia.co.uk/pi-com-bt-bp.html>



\*. Pi Setting serial console port for bluetooth



If you use RS232 or COM port change setting to RS232

If you need use Bluetooth change the RXD-TXD to Bluetooth

### Step 1. Modify the inittab File:

At a terminal command prompt issue this command:

**sudo nano /etc/inittab**

Look near the end of the file and find the lines that look like this:

**#Spawn a getty on Raspberry Pi serial line**

**T0:23:respawn:/sbin/getty -L ttyAMA0 115200 vt100**

change to

**#Spawn a getty on Raspberry Pi serial line**

**T0:23:respawn:/sbin/getty -L ttyAMA0 9600 vt100**

```
192.168.0.13 - PuTTY
GNU nano 2.2.6 File: /etc/inittab Modified

# Example how to put a getty on a serial line (for a terminal)
#
#T0:23:respawn:/sbin/getty -L ttyS0 9600 vt100
#T1:23:respawn:/sbin/getty -L ttyS1 9600 vt100

# Example how to put a getty on a modem line.
#
#T3:23:respawn:/sbin/mgetty -x0 -s 57600 ttyS3

#Spawn a getty on Raspberry Pi serial line
T0:23:respawn:/sbin/getty -L ttyAMA0 9600 vt100

^G Get Help ^O WriteOut ^R Read File ^Y Prev Page ^K Cut Text ^C Cur Pos
^X Exit ^J Justify ^W Where Is ^V Next Page ^U UnCut Text ^T To Spell
```

### Step 2.

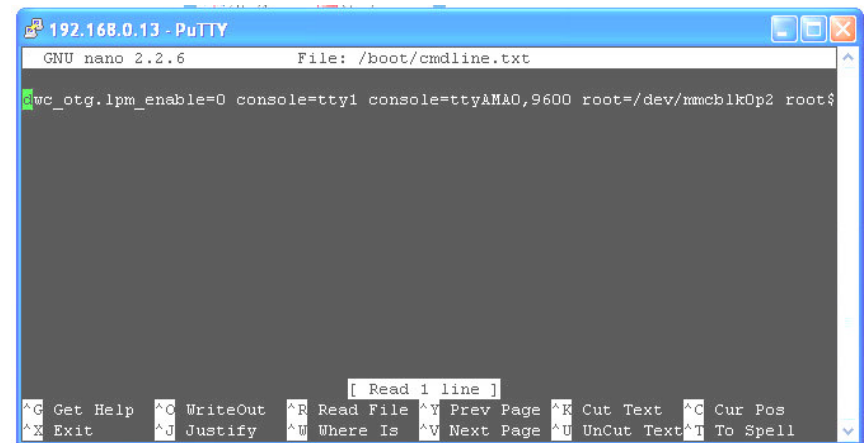
**sudo nano /boot/cmdline.txt**

The entire contents of the file look like this:

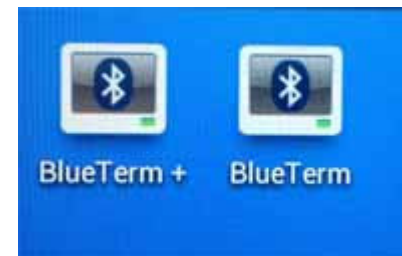
**dwc\_otg.lpm\_enable=0 console=ttyAMA0,115200 kgdboc=ttyAMA0,115200  
console=tty1 root=/dev/mmcbk0p2 rootfstype=ext4 elevator=deadline rootwait**

change to

**dwc\_otg.lpm\_enable=0 console=ttyAMA0,9600 kgdboc=ttyAMA0,9600  
console=tty1 root=/dev/mmcbk0p2 rootfstype=ext4 elevator=deadline  
rootwait**

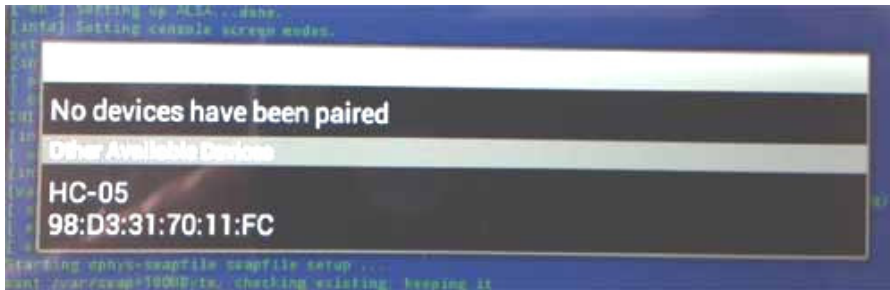


### Step 3 .Bluetooth setting for console port



BlueTerm

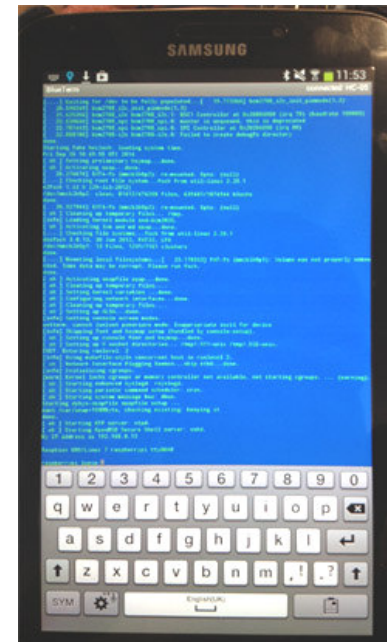
use Android Tablet BlueTerm+ or



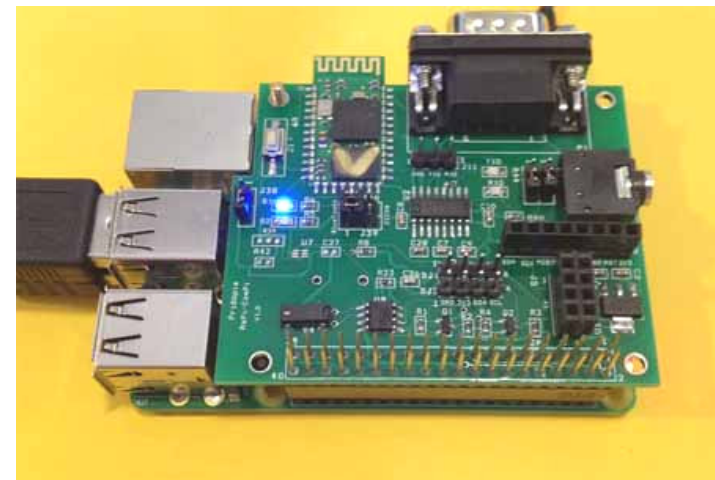
find "HC-05" bluetooth device



input pin "1234" for pair



you will see raspberry pi boot from Android Tablet screen



B2 LED flashing, B1 always On