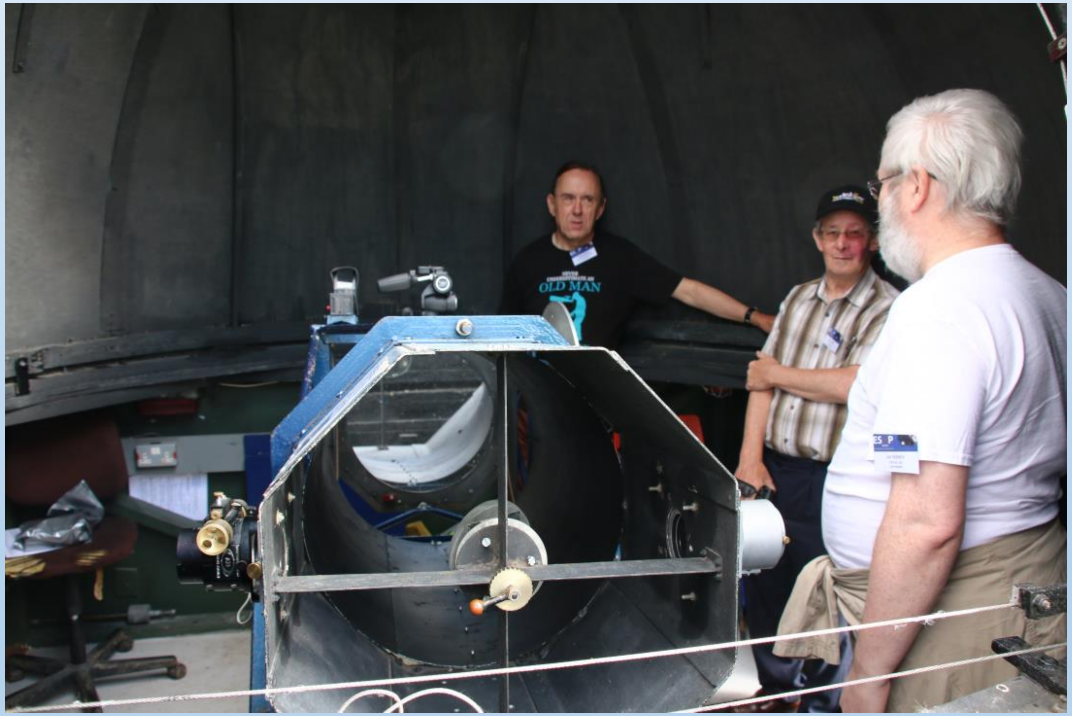
Construction and configuration of a Raspberry Pi GPS time server.
David Briggs – Hampshire Astronomy Group at Clanfield, UK

Demonstrated at ESOP XXXV, Guildford, UK.
2016 Aug 19-21

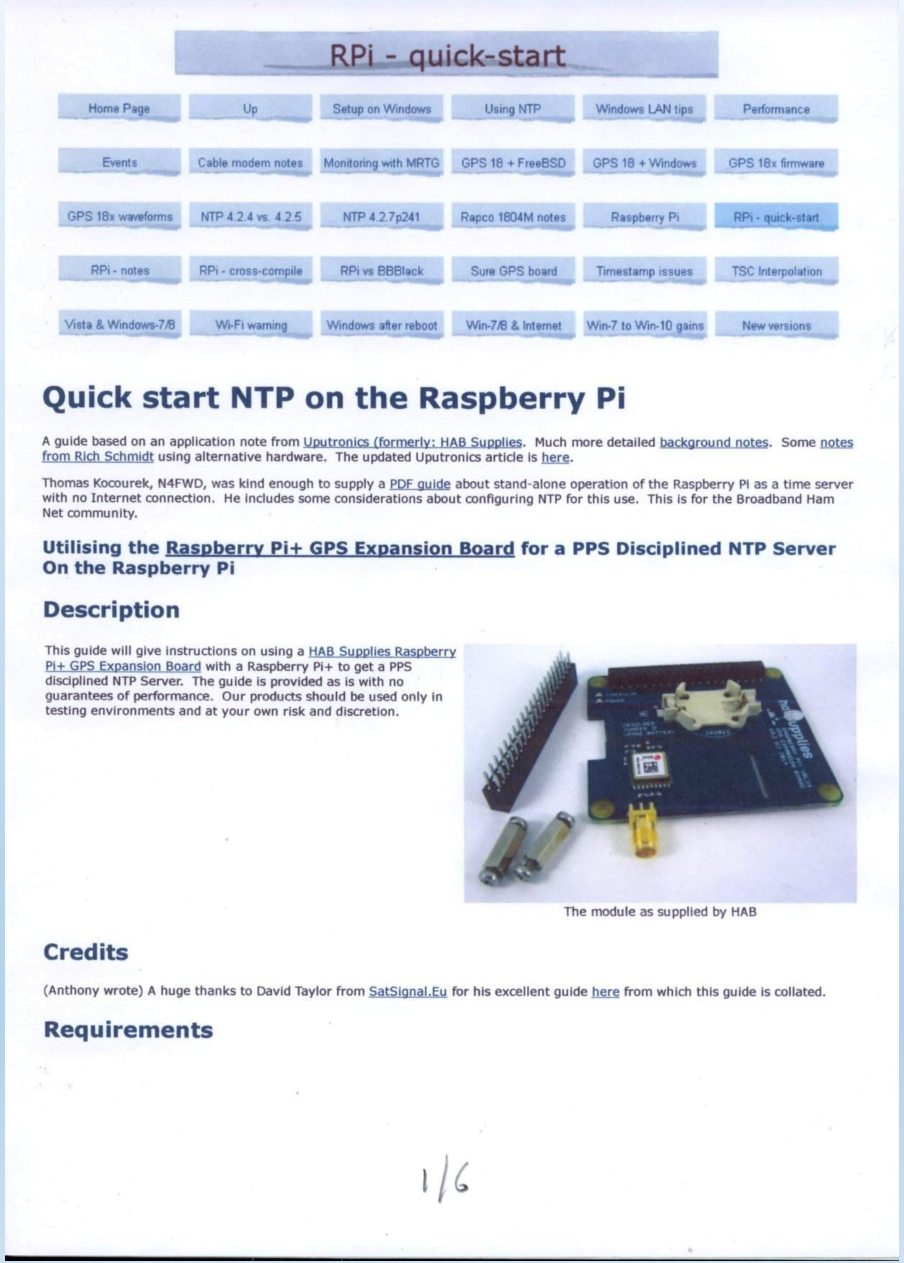
http://www.hantsastro.org.uk/

Delegates see the time server in action outside the lecture theatre during the tea break (and between showers !)



Dave Briggs, Alex Pratt and Jan Manek with the 16” used for astrometry and occultations at Clanfield.

Notes on construction and configuration – David Briggs



RPi - quick-start

Home Page	Up	Setup on Windows	Using NTP	Windows LAN tips	Performance
Events	Cable modem notes	Monitoring with MRTG	GPS 18 + FreeBSD	GPS 18 + Windows	GPS 18x firmware
GPS 18x waveforms	NTP 4.2.4 vs. 4.2.5	NTP 4.2.7p241	Rapco 1804M notes	Raspberry Pi	RPi - quick-start
RPi - notes	RPi - cross-compile	RPi vs BBBlack	Sure GPS board	Timestamp issues	TSC Interpolation
Vista & Windows-7/8	Wi-Fi warning	Windows after reboot	Win-7/8 & Internet	Win-7 to Win-10 gains	New versions

Quick start NTP on the Raspberry Pi

A guide based on an application note from Uputronics (formerly: HAB Supplies. Much more detailed background notes. Some notes from Rich Schmidt using alternative hardware. The updated Uputronics article is here.

Thomas Kocourek, N4FWD, was kind enough to supply a PDF guide about stand-alone operation of the Raspberry Pi as a time server with no Internet connection. He includes some considerations about configuring NTP for this use. This is for the Broadband Ham Net community.

Utilising the Raspberry Pi+ GPS Expansion Board for a PPS Disciplined NTP Server On the Raspberry Pi

Description

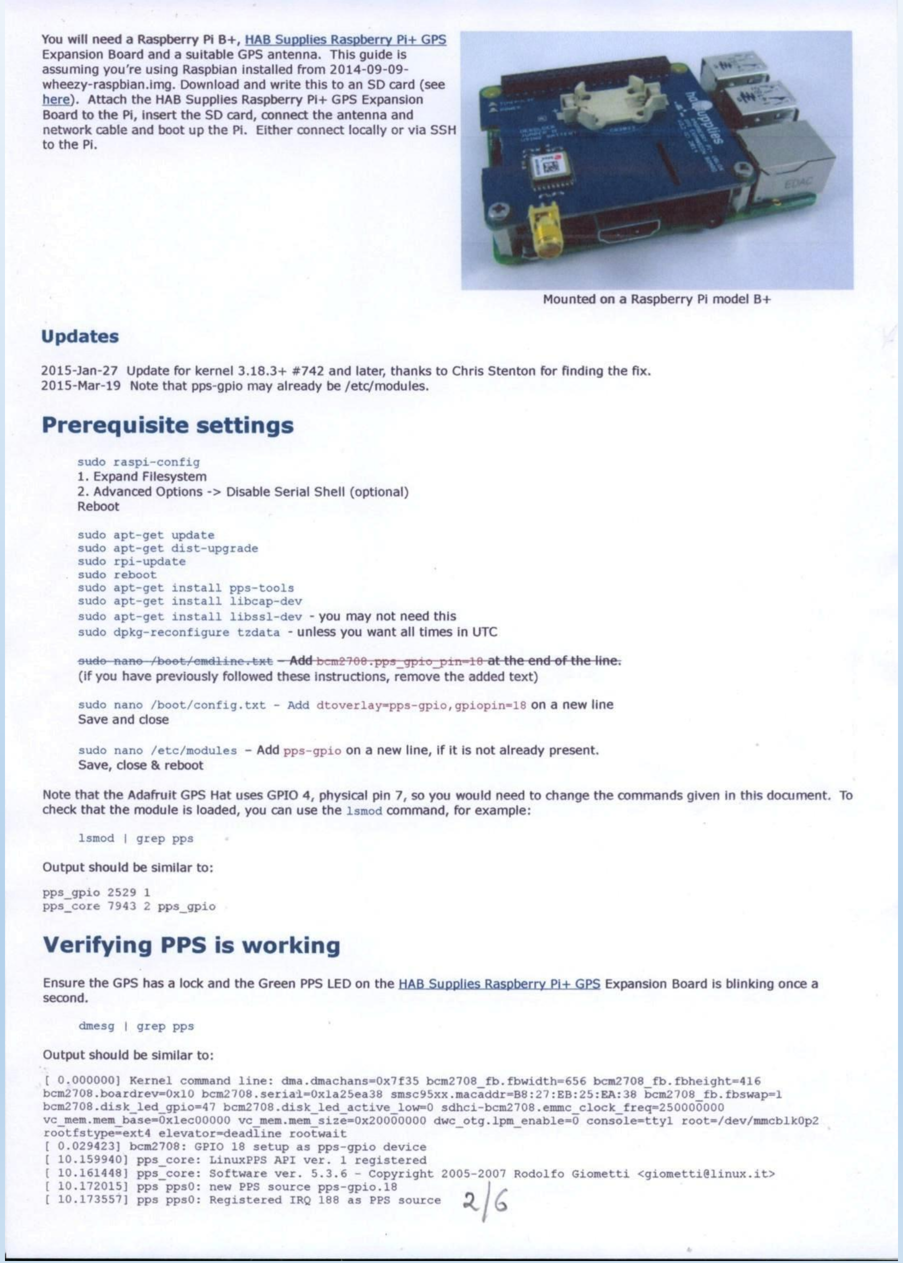
This guide will give instructions on using a HAB Supplies Raspberry Pi+ GPS Expansion Board with a Raspberry Pi+ to get a PPS disciplined NTP Server. The guide is provided as is with no guarantees of performance. Our products should be used only in testing environments and at your own risk and discretion.

The module as supplied by HAB

Credits

(Anthony wrote) A huge thanks to David Taylor from SatSignal.Eu for his excellent guide here from which this guide is collated.

Requirements



You will need a Raspberry Pi B+, HAB Supplies Raspberry Pi+ GPS Expansion Board and a suitable GPS antenna. This guide is assuming you're using Raspbian installed from 2014-09-09-wheezy-raspbian.img. Download and write this to an SD card (see here). Attach the HAB Supplies Raspberry Pi+ GPS Expansion Board to the Pi, insert the SD card, connect the antenna and network cable and boot up the Pi. Either connect locally or via SSH to the Pi.

Mounted on a Raspberry Pi model B+

Updates

2015-Jan-27 Update for kernel 3.18.3+ #742 and later, thanks to Chris Stenton for finding the fix.
2015-Mar-19 Note that pps-gpio may already be /etc/modules.

Prerequisite settings

```
sudo raspi-config
1. Expand Filesystem
2. Advanced Options -> Disable Serial Shell (optional)
Reboot

sudo apt-get update
sudo apt-get dist-upgrade
sudo rpi-update
sudo reboot
sudo apt-get install pps-tools
sudo apt-get install libcap-dev
sudo apt-get install libssl-dev - you may not need this
sudo dpkg-reconfigure tzdata - unless you want all times in UTC
```

~~sudo nano /boot/cmdline.txt – Add bcm2708.pps_gpio_pin=18 at the end of the line.~~
(if you have previously followed these instructions, remove the added text)

```
sudo nano /boot/config.txt - Add dtoverlay=pps-gpio,gpiopin=18 on a new line
Save and close

sudo nano /etc/modules – Add pps-gpio on a new line, if it is not already present.
Save, close & reboot
```

Note that the Adafruit GPS Hat uses GPIO 4, physical pin 7, so you would need to change the commands given in this document. To check that the module is loaded, you can use the `lsmod` command, for example:

```
lsmod | grep pps
```

Output should be similar to:

```
pps_gpio 2529 1
pps_core 7943 2 pps_gpio
```

Verifying PPS is working

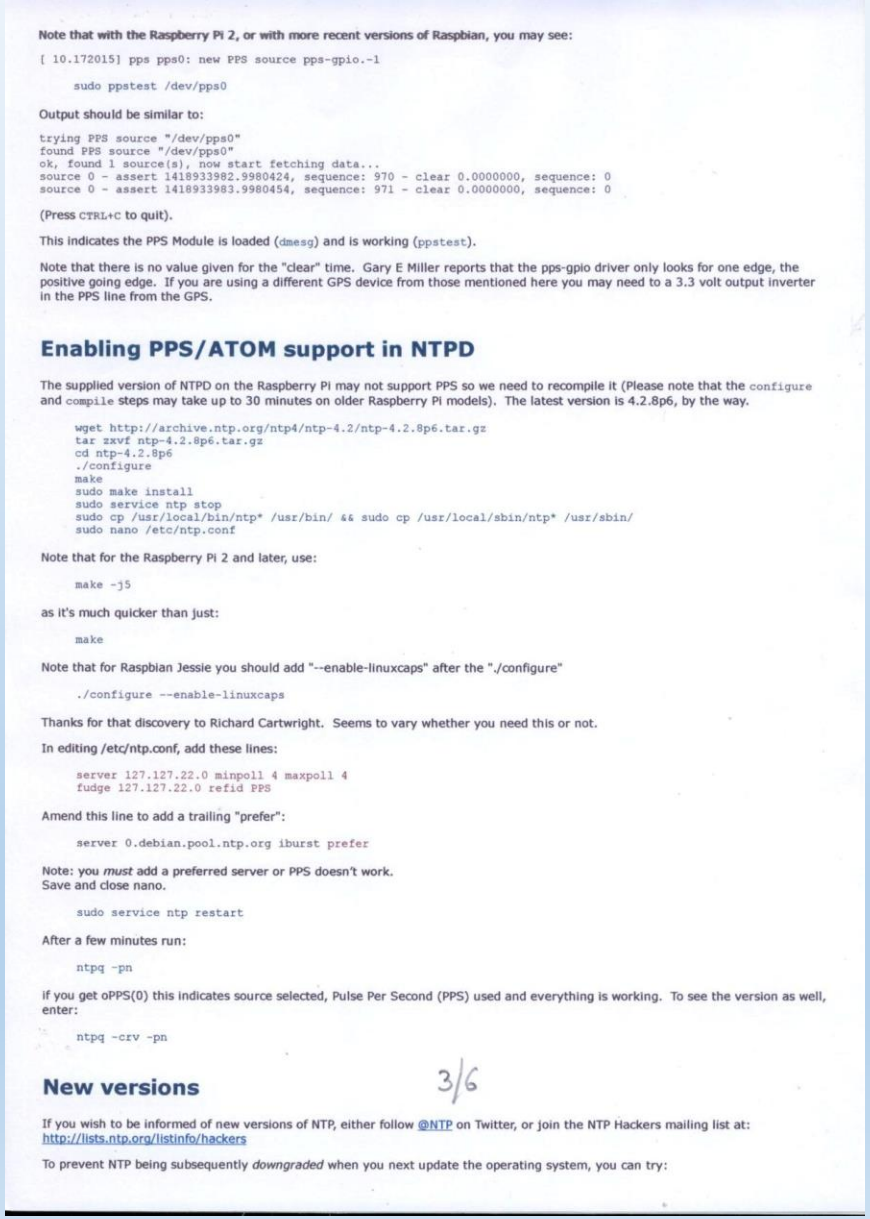
Ensure the GPS has a lock and the Green PPS LED on the HAB Supplies Raspberry Pi+ GPS Expansion Board is blinking once a second.

```
dmesg | grep pps
```

Output should be similar to:

```
[ 0.000000] Kernel command line: dma.dmachans=0x7f35 bcm2708_fb.fbwidth=656 bcm2708_fb.fbheight=416
bcm2708.boardrev=0x10 bcm2708.serial=0x1a25ea38 smsc95xx.macaddr=B8:27:EB:25:EA:38 bcm2708_fb.fbswap=1
bcm2708.disk_led_gpio=47 bcm2708.disk_led_active_low=0 sdhci-bcm2708.emmc_clock_freq=250000000
vc_mem.mem_base=0x1ec00000 vc_mem.mem_size=0x20000000 dwc_otg.lpm_enable=0 console=tty1 root=/dev/mmcblk0p2
rootfstype=ext4 elevator=deadline rootwait
[ 0.029423] bcm2708: GPIO 18 setup as pps-gpio device
[ 10.159940] pps_core: LinuxPPS API ver. 1 registered
[ 10.161448] pps_core: Software ver. 5.3.6 - Copyright 2005-2007 Rodolfo Giometti <giometti@linux.it>
[ 10.172015] pps pps0: new PPS source pps-gpio.18
[ 10.173557] pps pps0: Registered IRQ 188 as PPS source
```

Notes on construction and configuration cont. – David Briggs



Note that with the Raspberry Pi 2, or with more recent versions of Raspbian, you may see:

```
[ 10.172015] pps pps0: new PPS source pps-gpio.-1
```

```
     sudo ppstest /dev/pps0
```

Output should be similar to:

```
trying PPS source "/dev/pps0"
found PPS source "/dev/pps0"
ok, found 1 source(s), now start fetching data...
source 0 - assert 1418933982.9980424, sequence: 970 - clear 0.0000000, sequence: 0
source 0 - assert 1418933983.9980454, sequence: 971 - clear 0.0000000, sequence: 0
```

(Press `CTRL+C` to quit).

This indicates the PPS Module is loaded (`dmesg`) and is working (`ppstest`).

Note that there is no value given for the "clear" time. Gary E Miller reports that the pps-gpio driver only looks for one edge, the positive going edge. If you are using a different GPS device from those mentioned here you may need to a 3.3 volt output inverter in the PPS line from the GPS.

Enabling PPS/ATOM support in NTPD

The supplied version of NTPD on the Raspberry Pi may not support PPS so we need to recompile it (Please note that the `configure` and `compile` steps may take up to 30 minutes on older Raspberry Pi models). The latest version is 4.2.8p6, by the way.

```
     wget http://archive.ntp.org/ntp4/ntp-4.2/ntp-4.2.8p6.tar.gz
     tar zxvf ntp-4.2.8p6.tar.gz
     cd ntp-4.2.8p6
     ./configure
     make
     sudo make install
     sudo service ntp stop
     sudo cp /usr/local/bin/ntp* /usr/bin/ && sudo cp /usr/local/sbin/ntp* /usr/sbin/
     sudo nano /etc/ntp.conf
```

Note that for the Raspberry Pi 2 and later, use:

```
     make -j5
```

as it's much quicker than just:

```
     make
```

Note that for Raspbian Jessie you should add "--enable-linuxcaps" after the "./configure"

```
     ./configure --enable-linuxcaps
```

Thanks for that discovery to Richard Cartwright. Seems to vary whether you need this or not.

In editing /etc/ntp.conf, add these lines:

```
     server 127.127.22.0 minpoll 4 maxpoll 4
     fudge 127.127.22.0 refid PPS
```

Amend this line to add a trailing "prefer":

```
     server 0.debian.pool.ntp.org iburst prefer
```

Note: you *must* add a preferred server or PPS doesn't work.
Save and close nano.

```
     sudo service ntp restart
```

After a few minutes run:

```
     ntpq -pn
```

if you get oPPS(0) this indicates source selected, Pulse Per Second (PPS) used and everything is working. To see the version as well, enter:

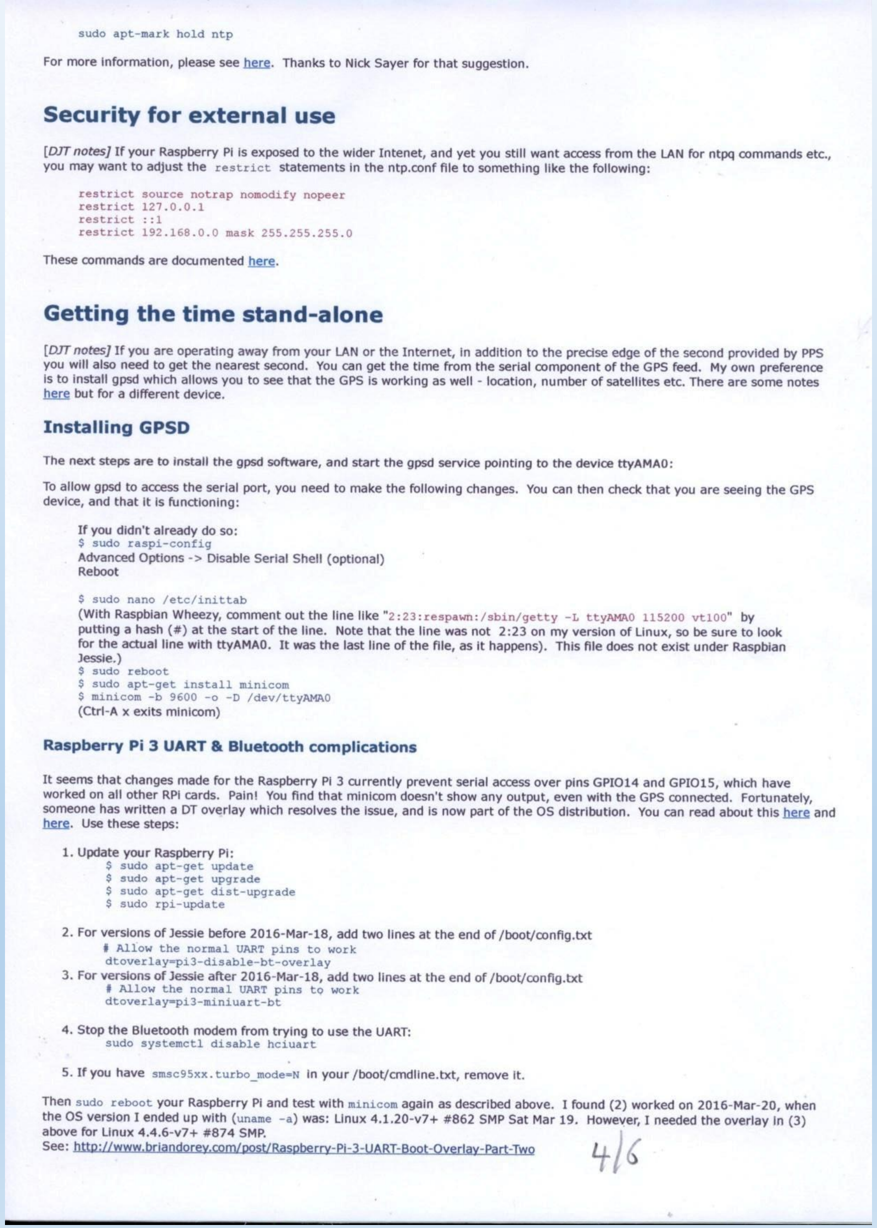
```
     ntpq -crv -pn
```

New versions

3/6

If you wish to be informed of new versions of NTP, either follow @NTP on Twitter, or join the NTP Hackers mailing list at: http://lists.ntp.org/listinfo/hackers

To prevent NTP being subsequently *downgraded* when you next update the operating system, you can try:



```
     sudo apt-mark hold ntp
```

For more information, please see here. Thanks to Nick Sayer for that suggestion.

Security for external use

[*DJT notes]* If your Raspberry Pi is exposed to the wider Intenet, and yet you still want access from the LAN for ntpq commands etc., you may want to adjust the `restrict` statements in the ntp.conf file to something like the following:

```
     restrict source notrap nomodify nopeer
     restrict 127.0.0.1
     restrict ::1
     restrict 192.168.0.0 mask 255.255.255.0
```

These commands are documented here.

Getting the time stand-alone

[*DJT notes]* If you are operating away from your LAN or the Internet, in addition to the precise edge of the second provided by PPS you will also need to get the nearest second. You can get the time from the serial component of the GPS feed. My own preference is to install gpsd which allows you to see that the GPS is working as well - location, number of satellites etc. There are some notes here but for a different device.

Installing GPSD

The next steps are to install the gpsd software, and start the gpsd service pointing to the device ttyAMA0:

To allow gpsd to access the serial port, you need to make the following changes. You can then check that you are seeing the GPS device, and that it is functioning:

> If you didn't already do so:
> `$ sudo raspi-config`
> Advanced Options -> Disable Serial Shell (optional)
> Reboot
>
> `$ sudo nano /etc/inittab`
> (With Raspbian Wheezy, comment out the line like "`2:23:respawn:/sbin/getty -L ttyAMA0 115200 vt100`" by putting a hash (#) at the start of the line. Note that the line was not 2:23 on my version of Linux, so be sure to look for the actual line with ttyAMA0. It was the last line of the file, as it happens). This file does not exist under Raspbian Jessie.)
> `$ sudo reboot`
> `$ sudo apt-get install minicom`
> `$ minicom -b 9600 -o -D /dev/ttyAMA0`
> (Ctrl-A x exits minicom)

Raspberry Pi 3 UART & Bluetooth complications

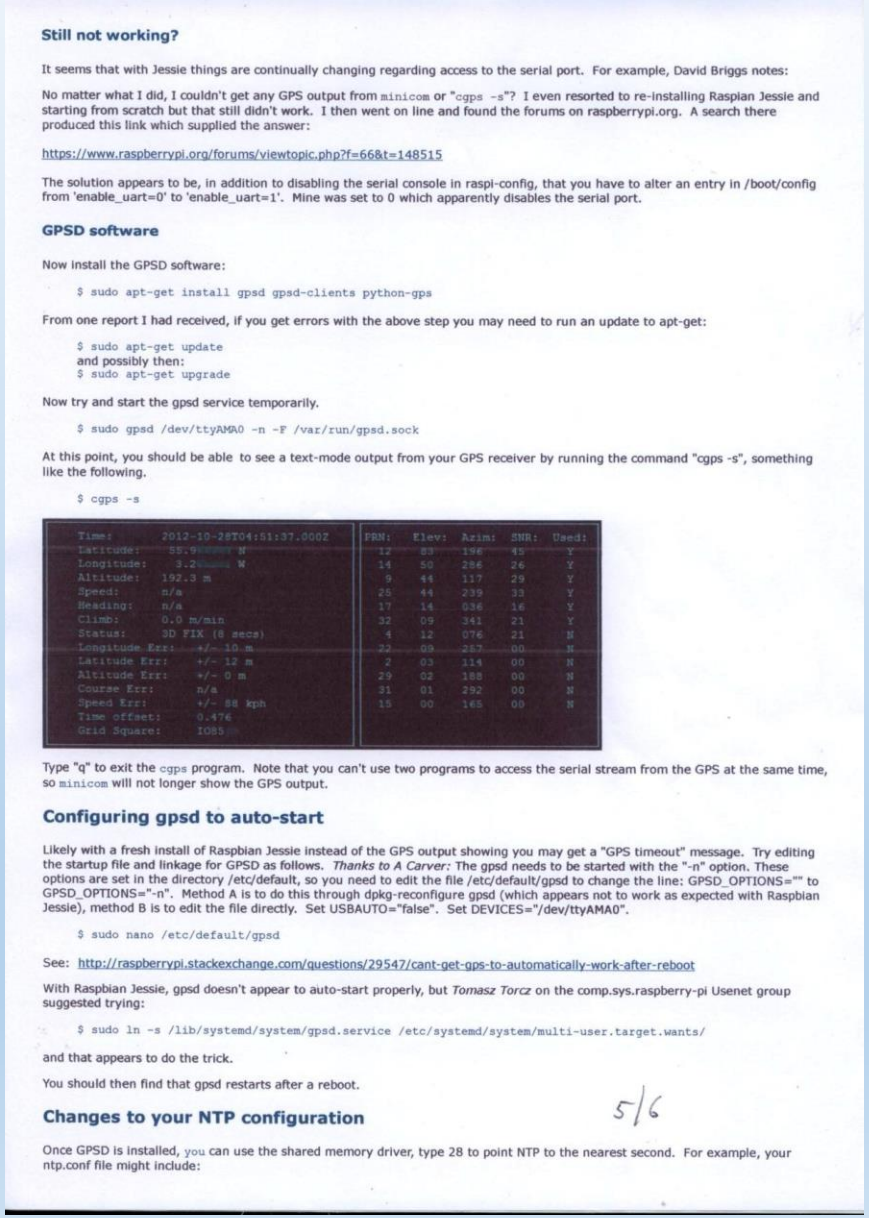
It seems that changes made for the Raspberry Pi 3 currently prevent serial access over pins GPIO14 and GPIO15, which have worked on all other RPi cards. Pain! You find that minicom doesn't show any output, even with the GPS connected. Fortunately, someone has written a DT overlay which resolves the issue, and is now part of the OS distribution. You can read about this here and here. Use these steps:

1. Update your Raspberry Pi:
   ```
   $ sudo apt-get update
   $ sudo apt-get upgrade
   $ sudo apt-get dist-upgrade
   $ sudo rpi-update
   ```
2. For versions of Jessie before 2016-Mar-18, add two lines at the end of /boot/config.txt
   ```
   # Allow the normal UART pins to work
   dtoverlay=pi3-disable-bt-overlay
   ```
3. For versions of Jessie after 2016-Mar-18, add two lines at the end of /boot/config.txt
   ```
   # Allow the normal UART pins to work
   dtoverlay=pi3-miniuart-bt
   ```
4. Stop the Bluetooth modem from trying to use the UART:
   ```
   sudo systemctl disable hciuart
   ```
5. If you have `smsc95xx.turbo_mode=N` in your /boot/cmdline.txt, remove it.

Then `sudo reboot` your Raspberry Pi and test with `minicom` again as described above. I found (2) worked on 2016-Mar-20, when the OS version I ended up with (`uname -a`) was: Linux 4.1.20-v7+ #862 SMP Sat Mar 19. However, I needed the overlay in (3) above for Linux 4.4.6-v7+ #874 SMP.
See: http://www.briandorey.com/post/Raspberry-Pi-3-UART-Boot-Overlay-Part-Two

4/6

Notes on construction and configuration cont. – David Briggs



Still not working?

It seems that with Jessie things are continually changing regarding access to the serial port. For example, David Briggs notes:

No matter what I did, I couldn't get any GPS output from `minicom` or "`cgps -s`"? I even resorted to re-installing Raspian Jessie and starting from scratch but that still didn't work. I then went on line and found the forums on raspberrypi.org. A search there produced this link which supplied the answer:

https://www.raspberrypi.org/forums/viewtopic.php?f=66&t=148515

The solution appears to be, in addition to disabling the serial console in raspi-config, that you have to alter an entry in /boot/config from 'enable_uart=0' to 'enable_uart=1'. Mine was set to 0 which apparently disables the serial port.

GPSD software

Now install the GPSD software:

```
$ sudo apt-get install gpsd gpsd-clients python-gps
```

From one report I had received, if you get errors with the above step you may need to run an update to apt-get:

```
$ sudo apt-get update
and possibly then:
$ sudo apt-get upgrade
```

Now try and start the gpsd service temporarily.

```
$ sudo gpsd /dev/ttyAMA0 -n -F /var/run/gpsd.sock
```

At this point, you should be able to see a text-mode output from your GPS receiver by running the command "cgps -s", something like the following.

```
$ cgps -s
```

```
Time:          2012-10-28T04:51:37.000Z     PRN:   Elev:  Azim:  SNR:  Used:
Latitude:       55.9[redacted] N             12     83     196    45     Y
Longitude:       3.2[redacted] W             14     50     286    26     Y
Altitude:      192.3 m                        9     44     117    29     Y
Speed:         n/a                           25     44     239    33     Y
Heading:       n/a                           17     14     036    16     Y
Climb:         0.0 m/min                     32     09     341    21     Y
Status:        3D FIX (8 secs)                4     12     076    21     N
Longitude Err:   +/- 10 m                    22     09     257    00     N
Latitude Err:    +/- 12 m                     2     03     114    00     N
Altitude Err:    +/- 0 m                     29     02     188    00     N
Course Err:      n/a                         31     01     292    00     N
Speed Err:       +/- 88 kph                  15     00     165    00     N
Time offset:     0.476
Grid Square:     IO85[redacted]
```

Type "q" to exit the `cgps` program. Note that you can't use two programs to access the serial stream from the GPS at the same time, so `minicom` will not longer show the GPS output.

Configuring gpsd to auto-start

Likely with a fresh install of Raspbian Jessie instead of the GPS output showing you may get a "GPS timeout" message. Try editing the startup file and linkage for GPSD as follows. *Thanks to A Carver:* The gpsd needs to be started with the "-n" option. These options are set in the directory /etc/default, so you need to edit the file /etc/default/gpsd to change the line: GPSD_OPTIONS="" to GPSD_OPTIONS="-n". Method A is to do this through dpkg-reconfigure gpsd (which appears not to work as expected with Raspbian Jessie), method B is to edit the file directly. Set USBAUTO="false". Set DEVICES="/dev/ttyAMA0".

```
$ sudo nano /etc/default/gpsd
```

See: http://raspberrypi.stackexchange.com/questions/29547/cant-get-gps-to-automatically-work-after-reboot

With Raspbian Jessie, gpsd doesn't appear to auto-start properly, but *Tomasz Torcz* on the comp.sys.raspberry-pi Usenet group suggested trying:

```
$ sudo ln -s /lib/systemd/system/gpsd.service /etc/systemd/system/multi-user.target.wants/
```

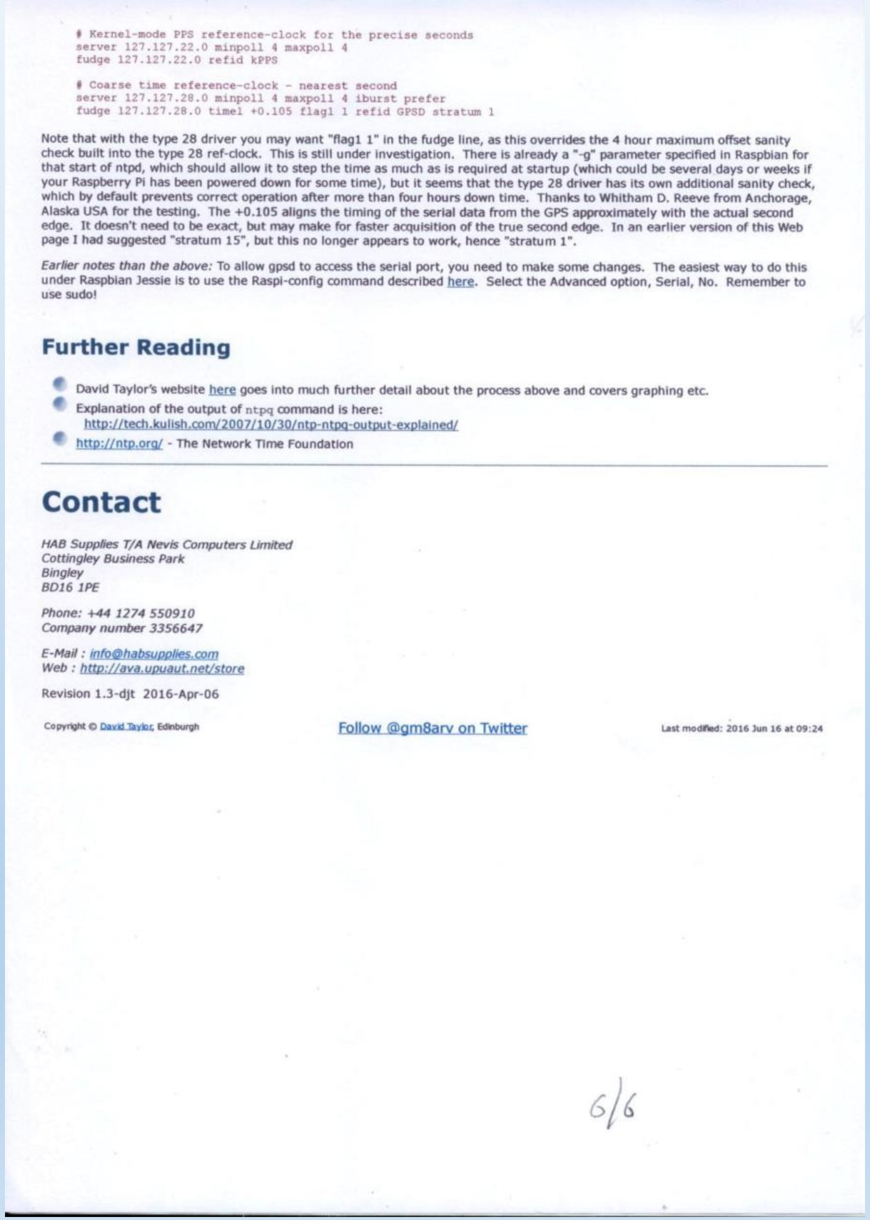
and that appears to do the trick.

You should then find that gpsd restarts after a reboot.

5/6

Changes to your NTP configuration

Once GPSD is installed, you can use the shared memory driver, type 28 to point NTP to the nearest second. For example, your ntp.conf file might include:



```
# Kernel-mode PPS reference-clock for the precise seconds
server 127.127.22.0 minpoll 4 maxpoll 4
fudge 127.127.22.0 refid kPPS

# Coarse time reference-clock - nearest second
server 127.127.28.0 minpoll 4 maxpoll 4 iburst prefer
fudge 127.127.28.0 time1 +0.105 flag1 1 refid GPSD stratum 1
```

Note that with the type 28 driver you may want "flag1 1" in the fudge line, as this overrides the 4 hour maximum offset sanity check built into the type 28 ref-clock. This is still under investigation. There is already a "-g" parameter specified in Raspbian for that start of ntpd, which should allow it to step the time as much as is required at startup (which could be several days or weeks if your Raspberry Pi has been powered down for some time), but it seems that the type 28 driver has its own additional sanity check, which by default prevents correct operation after more than four hours down time. Thanks to Whitham D. Reeve from Anchorage, Alaska USA for the testing. The +0.105 aligns the timing of the serial data from the GPS approximately with the actual second edge. It doesn't need to be exact, but may make for faster acquisition of the true second edge. In an earlier version of this Web page I had suggested "stratum 15", but this no longer appears to work, hence "stratum 1".

Earlier notes than the above: To allow gpsd to access the serial port, you need to make some changes. The easiest way to do this under Raspbian Jessie is to use the Raspi-config command described here. Select the Advanced option, Serial, No. Remember to use sudo!

Further Reading

- David Taylor's website here goes into much further detail about the process above and covers graphing etc.
- Explanation of the output of `ntpq` command is here:
  http://tech.kulish.com/2007/10/30/ntp-ntpq-output-explained/
- http://ntp.org/ - The Network Time Foundation

Contact

HAB Supplies T/A Nevis Computers Limited
Cottingley Business Park
Bingley
BD16 1PE

Phone: +44 1274 550910
Company number 3356647

E-Mail : info@habsupplies.com
Web : http://ava.upuaut.net/store

Revision 1.3-djt 2016-Apr-06

Follow @gm8arv on Twitter

Last modified: 2016 Jun 16 at 09:24

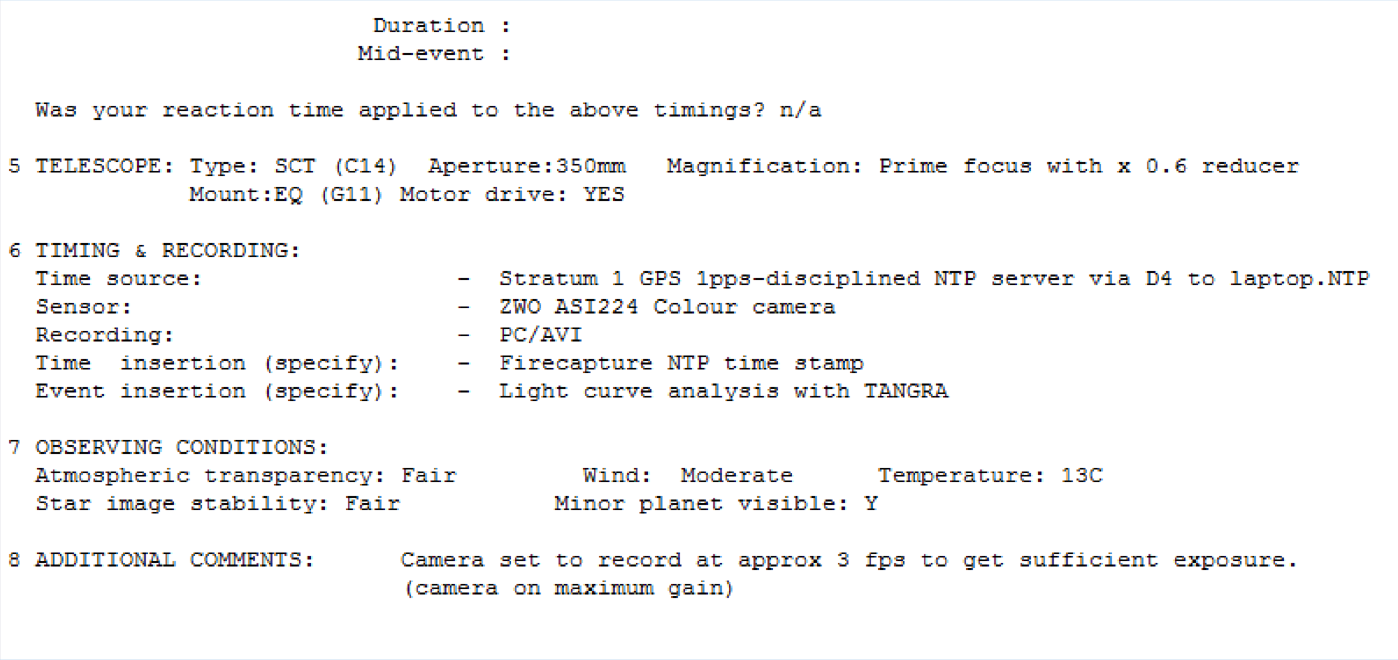
6/6

Negative Observation of (696)Leonora on 2016-08-07.
USB camera and Stratum 1pps-disciplined NTP server via D4:

This page and data is courtesy of Simon Kidd (UK) – (Private Communication)

Server Described and demonstrate by David Briggs.

Hampshire Astronomy Group (HAG)



```
                            Duration :
                           Mid-event :

  Was your reaction time applied to the above timings? n/a

5 TELESCOPE: Type: SCT (C14)  Aperture:350mm   Magnification: Prime focus with x 0.6 reducer
             Mount:EQ (G11) Motor drive: YES

6 TIMING & RECORDING:
  Time source:                   -  Stratum 1 GPS 1pps-disciplined NTP server via D4 to laptop.NTP
  Sensor:                        -  ZWO ASI224 Colour camera
  Recording:                     -  PC/AVI
  Time  insertion (specify):     -  Firecapture NTP time stamp
  Event insertion (specify):     -  Light curve analysis with TANGRA

7 OBSERVING CONDITIONS:
  Atmospheric transparency: Fair         Wind:  Moderate      Temperature: 13C
  Star image stability: Fair           Minor planet visible: Y

8 ADDITIONAL COMMENTS:      Camera set to record at approx 3 fps to get sufficient exposure.
                            (camera on maximum gain)
```

Dimension-4 corrections at 1 min intervals Courtesy of Simon Kidd

```
• 2016-08-07    01:28:31.181         -3.957748e-005  192.168.0.11   SNTP
• 2016-08-07    01:29:31.195          1.434565e-003  192.168.0.11   SNTP
• 2016-08-07    01:30:31.224          1.926899e-003  192.168.0.11   SNTP
• 2016-08-07    01:31:31.251          3.108501e-003  192.168.0.11   SNTP
• 2016-08-07    01:32:31.263          -2.396107e-004 192.168.0.11   SNTP
• 2016-08-07    01:33:31.299          1.608372e-003  192.168.0.11   SNTP
• 2016-08-07    01:34:31.328          1.972437e-003  192.168.0.11   SNTP
• 2016-08-07    01:35:31.353          1.992226e-003  192.168.0.11   SNTP
• 2016-08-07
```