



Raspberry Pi GPS/RTC Expansion Board Datasheet

Description

The Uputronics Raspberry Pi GPS/RTC Expansion Board provides a modern multi GNSS GPS receiver and hardware RTC with quick no soldering required connection to all Raspberry Pi boards with the 2x20 header (the original Pi Model A/B are not supported). Featuring PPS (Pulse per second) output to permit the use of the board for PPS disciplined NTP servers, RAW output available from the UBlox™ GPS module and twin super caps for GPS hot start/setting retention & maintaining RTC.

This board retains backwards compatibility with previous generation Uputronics Pi GPS boards.

Pin Configurations

Pin	Raspberry Pi Name	HAB-GPSPI+-ASSY
01	3.3V DC Power	3.3V DC Power
02/04	5V DC Power	5V DC Power used for super caps
03	GPIO02(SDA1,I2C)	GPS/RTC SDA
05	GPIO03(SCL1,I2C)	GPS/RTC SCL
08	GPIO14(TXD0)	GPS SERIAL RXD
10	GPIO15(RXD0)	GPS SERIAL TXD
12	GPIO18	TIME PULSE

Board Physical Specifications

Weight	15g excluding fixings and antenna.
Batteries	2 x 0.2F 3.3V Super Capacitors
Connector Pitch	2.54mm pitch Raspberry Pi 2x20 Header.
Power Usage (from 3.3V)	Acquire 25mA / Tracking 21mA / Cyclic PSM Mode 9mA
GPS Antenna Connector	SMA Female

Board Environmental

Operating Temperature -40°C to +85°C

GPS Specifications

The GPS has the following specifications:

- 72-channel u-blox™ M8 engine
- Up to 3 concurrent GNSS (BeiDou, Galileo, GLONASS, GPS/QZSS)
- Default GNSS: GPS/GLONASS.
- -167dBm Navigation Sensitivity
- 10Hz navigation rate in multi GNSS mode, 18Hz in GPS only
- RAW Output available via UBX-RXM-RAWX Message
- 3V supplied to antenna port to power active antenna
- Antenna requirements: Active 3V Ceramic Patch
- No EEPROM settings must be sent at power up or saved to battery backed RAM.
- Default baud rate: 115200bps
- I²C Bus (Address 0x42)
- Thermal assistance vents

For full documentation please consult the u-blox 8/u-blox M8 receiver description including protocol specification document (UBX-13003221) linked here:

<https://store.uputronics.com/files/UBX-13003221.pdf>

RTC Specifications

The RTC has the following specifications:

- Micro Crystal RV-3028-C7 Real Time Clock Module
- Factory calibrated ± 1 ppm @ 25°C
- I²C Bus (Address 0x52)
- Built in kernel compatibility
- Provides year, month, date, weekday, hours, minutes and seconds

In The Box

1 x Raspberry Pi GPS/RTC Expansion Board
4 x 11mm standoffs w/8 M2.5 Screws
1 x Header with 4.93mm tails

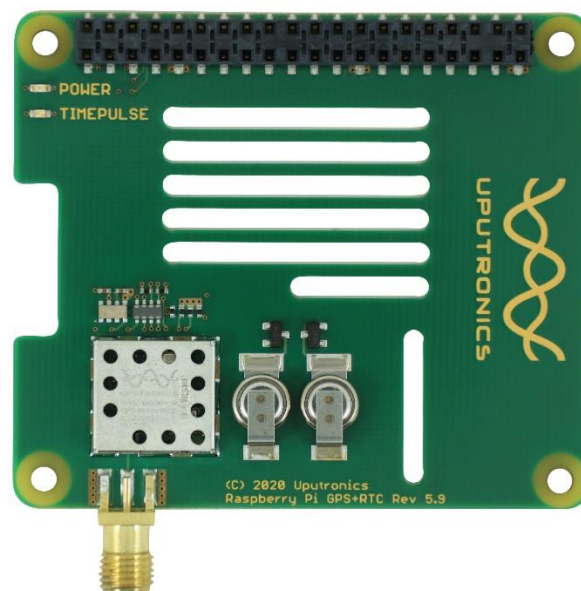
Disclaimer

All Uputronics products are sold as test equipment with no guarantees of performance or operation, they are intended for engineering, research or lab use only not for use in production or commercial systems.

This is not to say there is any quality issue with our boards, people are using these in such varying and dynamic environments its not possible for us to test every possible scenario. Therefore we advise you do your own testing to ensure operation in your environment.

Installation

Attach the header to the Raspberry Pi 2x20 header. Attach the four supplied standoffs to the Raspberry Pi using the M2.5mm screws. Clip the GPS/RTC Expansion Board on top of the header, the header pins push through the bottom with a click. The supercaps should face upwards when correctly installed. Secure the board using the remaining four screws.



RTC Operation

This guide assumes you are using Raspbian.

```
sudo raspi-config
```

Select Interfacing Options and Enable I2C

```
sudo apt-get install python-smbus i2c-tools
sudo i2cdetect -y 1
```

```
      0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00:      -- -- -- -- -- -- -- -- -- -- -- -- --
10: -- -- -- -- -- -- -- -- -- -- -- -- -- --
20: -- -- -- -- -- -- -- -- -- -- -- -- -- --
30: -- -- -- -- -- -- -- -- -- -- -- -- -- --
40: -- -- 42 -- -- -- -- -- -- -- -- -- -- --
50: -- -- 52 -- -- -- -- -- -- -- -- -- -- --
60: -- -- -- -- -- -- -- -- -- -- -- -- -- --
70: -- -- -- -- -- -- -- -- --
```

Here you can see the RTC (0x52) and GPS (0x42) on the I2C bus.

```
sudo vi /boot/config.txt
```

Add a line with:

```
dtoverlay=i2c-rtc,rv3028
```

```
sudo reboot
```

On reboot rerun the i2cdetect line and you should see 52 has been replaced with UU to indicate the kernel driver is loaded.

```
sudo apt-get -y remove fake-hwclock
sudo update-rc.d -f fake-hwclock remove
sudo systemctl disable fake-hwclock
sudo vi /lib/udev/hwclock-set
```

Comment out:

```
#if [ -e /run/systemd/system ] ; then
# exit 0
#fi
# /sbin/hwclock --rtc=$dev --systz --badyear
```

To read time:

```
sudo hwclock -v -r
```

To write time:

```
sudo hwclock -w
```

GPS/PPS Operation

To verify operation of PPS:

```
sudo apt-get install pps-tools  
sudo nano /boot/config.txt
```

add the following line:

```
dtoverlay=pps-gpio  
  
sudo nano /etc/modules
```

add the following line:

```
pps-gpio  
  
sudo reboot  
sudo ppstest /dev/pps0
```

To verify the operation of the GPS via serial

```
sudo apt-get install minicom  
minicom -b 115200 -o -D /dev/ttyS0
```

It is beyond the scope of this datasheet to explain how to use this board as a GPS disciplined NTP server.

Note On PPS

Since V3.00 of the Ublox firmware the time pulse is not released until all time parameters are known including leap seconds. There it could be up to 12.5 minutes before time pulse is available however positional lock is achieved from cold in the expected sub 30 seconds.

Note On RTC

If you are losing time over reboot or seeing:

```
kernel: [ 156.227408] rtc-rv3028 1-0052: Voltage low, data is invalid.
```

This relates to an issue with the current kernel + the RV3028 module needing certain registers to be set correctly. The following script fixes this:

```
http://store.uputronics.com/files/configure-rv3028.sh
```

For more information see:

<https://www.satsignal.eu/raspberry-pi/Uputronics-RTC.html>

http://www.philrandal.co.uk/blog/archives/2019/04/entry_213.html