## Rs-Pi DS2482-800 2-Wire 17 Bus Board V1.5 User Manual





The 1-Wire port is based around a DS2482-800 &DS2482-100 I2C to 1-Wire bridge device. The DS2482-800 provides bi-directional protocol conversion between the I2C port on the Raspberry Pi and any attached 1-Wire® slave devices. An ESD Protection Diode is used to protect the Board and Raspberry Pi from electrostatic spikes on the 1-Wire port. Connections to the 1-Wire port can be made through pin on the PCB.

1-Wire 2482-100 1 channel WITH 1 ESD protection diode

1 Wire 2482-800 8 channel with 8 ESD protection diode

with DS18B20 Temperature Sensor

 \*. U11 DS2482-100 I2C to 1-Wire bridge device J11 1-wire Port pin 1 - pin4 (5V,GND, OW (1-Wire Data, ESD Protected). RT (1-Wire Return/Ground ,ESC protected)

\* U1,U2,U4,U5,U6,U7,U8,U9, U12, DS9503P ESD protection diode

 \*. U3 DS2482-800 I2C to 1-Wire bridge device J2,J4,J7,J8,J1,J3,J6,J9 2 1-wire Port pin 1 - pin4 (5V,GND, OW (1-Wire Data, ESD Protected). RT (1-Wire Return/Ground ,ESC protected)

### \* J10 RTC SQW OUTPUT

\*. J36 2P Terminal block 5V input to 5V 1-Wire output to pin 1 or choose from P1 pin2 select from JP2

 $^{\ast}$  J5 I2c input port Pin1 to pin4 ( GND,5V,SDA,SCL) through Logic Level converter

\* U22 DS1307 RTC need CR1220 3V backup battery



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

RTC DS1307 - 68 in i2cdetect -y 0 or i2cdetect -y

 $1 \quad \mbox{for Rs-Pi V2 you will see 68} \ \mbox{in the screen} \\$ 

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 –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 & temp tmp102 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



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





i2c bus device detect 18 -- > DS2482-100 1c --> DS2482-800 1f -->

DS2482-800 68 --> DS1307 RTC

₽ 192.168.0.17 - PuTTY	
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.	
Last login: Tue Sep 23 16:17:39 2014	
root@raspberrypi:~# /opt/owfs/bin/owserveri2c=/dev/i2c-1:all	
root@raspberrypi:~# /opt/owfs/bin/owdir	
/28.0993E0050000	
/bus.16	
/bus.15	
/bus.14	
/bus.13	
/bus.12	
/bus.11	
/bus.10	
/bus.9	
/bus.8	
/bus.7	
/bus.6	
/bus.5	
/bus.4	
/bus.3	
/ bus.2	
/bus.i	
/ bus.o	
/uncached	
/sectings	
/system	
/structure	
rootBresherruni.~#	
receptaspertypt///	<u> </u>

It appears that OWServer has found 17 1-wire busses,

DS2482-100 is bus.0 DS2482-800 is bus.1 .. bus 8 DS2482-800 is bus.9 .. bus 16

/28.xxxxxx - Connect & Detect DALLAS 18B20P TEMP Sensor

# Installation of the OWFS (One Wire File System)

First you need to install the following packages:

#### sudo apt-get update

#### sudo apt-get install automake autoconf autotools-dev gcc-4.7 libtool libusb-dev libfuse-dev swig python2.6-dev tcl8.4-dev php5-dev i2c-tools

If promoted answer Yes on any questions during the install.

Download the latest version of OWFS to your usr/src directory

#### cd /usr/src

#### sudo wget -O owfs-

### latest.tgz http://sourceforge.net/projects/owfs/files/latest/download

Unpack with the following command:

#### sudo tar xzvf owfs-latest.tgz

Next you must configure OWFS: (replace X.XXXX with the version number you downloaded)

#### cd owfs-X.XXXX

#### sudo ./configure

If everything is correct, you should get a result like this:

Current configuration:

Deployment location: /opt/owfs

Compile-time options:

Caching is enabled

USB is DISABLED

#### etc.

Next you need to compile OWFS which will take approx. 30 minutes with the following command:

#### sudo make

Next install OWFS which will take a few minutes

#### sudo make install

Once the installation has completed you need to create a mountpoint for the 1wire folder:

#### sudo mkdir /mnt/1wire

In order to use the 1wire devices without root privileges you have to change the FUSE settings, edit the fuse configuration file with:

#### sudo nano /etc/fuse.conf

Update this line: #user\_allow\_other and remove the # from the start, then save your changes

You can now start using OWFS to access your i2c devices and any connected sensors:

### sudo /opt/owfs/bin/owfs --i2c=ALL:ALL --allow\_other /mnt/1wire/

Using a terminal window navigate to the /mnt/1wire/ directory and use the Is command to list all connected devices.

If you have a temperature sensor connected you should have a folder starting with 28.xxxxxx

cd into this folder and then enter cat temperature to read the temperature of the sensor.

cat temperature -- 22.312 & 32.125

## **Download User manual**

### http://www.pridopia.co.uk/pi-2482-800-2.html

## **Package Content**

1x Rs-Pi DS2482-800 17 Bus 1-wire for Pi B+ board 1x Manual



Example for connect DS18B20