

RTC

Timestamping the data and information is a key feature for any IoT and M2M hardware and SW. Standard high precision battery backed RTC (Real Time Clock) is a low cost solution to have a critical timestamping. One can have much more synchronized timestamping and HW synchronization using GPS timing, VIBE also supports GPS base HW time synchronization (See application note). RTC act as a heartbeat for both HW and SW timestamping. The RTC maintains minutes, hours, day, date and year information and automatically account changes for the month date depending on the number of days in the month. Also it includes two programmable time-of-day alarm which can be programmed and a programmable square wave output.

- VIBE uses DS3231 as RTC which also comes with compensated crystal oscillator (TCXO) and crystal.
- The RTC uses I2C2 and all the transfers are made through it.
- Vibe comes with a 3 volt non-rechargeable coin cell(CR1216) for RTC backup.
- RTC can be time synchronized with GPS (Optional with Vibe) that can give a time synchronized to GPS for days with minor correction sent from GPS timing.

SETTING UP RTC on VIBE

Shipped Beaglebone black does not include a RTC. Vibe added RTC to have a key feature for M2M and IoT application where the remote data timestamping is a key feature.

The RTC uses I2C, so we can check it using the command

i2cdetect -y -r 1

```
root@Yantrr:~# i2cdetect -y -r 1
   0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
10:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
20:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
30:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
40:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
50:  --  --  --  --  UU UU UU UU  --  --  --  --  --  --  --
60:  --  --  --  --  --  --  --  --  68  --  --  --  --  --  --
70:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
```

An ID 68 can be seen on the address 0x68 which shows the presence of RTC. This means that the module is wired up and is verified. After this execute the following for our DS3231 module

echo ds1307 0x68 > /sys/class/i2c-adapter/i2c-1/new_device

The RTC is now ready. Check the present time on the RTC by :

hwclock -r -f /dev/rtc1

```

root@Vayu-Cape:~# i2cdetect -y -r 1
    0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
10:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
20:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
30:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
40:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
50:  --  --  --  --  UU UU UU UU  --  --  --  --  --  --  --
60:  --  --  --  --  --  --  --  --  68  --  --  --  --  --  --
70:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
root@Vayu-Cape:~# echo ds3231 0x68 > /sys/class/i2c-adapter/i2c-1/new_device
root@Vayu-Cape:~# hwclock -r -f /dev/rtc1
Sat 01 Jan 2000 12:02:30 AM UTC -0.490685 seconds

```

If it has been accessed the first time then you will see the date as 1 January 2000 as can be seen from the figure. To update to a proper time use ntp:

sudo apt-get install ntp

To update the time, use

ntpdate -b -s -u pool.ntp.org

after this check logcat system date and time using

date

After this write this by using

hwclock -w -f /dev/rtc1

```

root@Vayu-Cape:~# ntpdate -b -s -u pool.ntp.org
root@Vayu-Cape:~# date
Fri Jul 3 07:23:17 UTC 2015
root@Vayu-Cape:~# hwclock -w -f /dev/rtc1
root@Vayu-Cape:~# hwclock -r -f /dev/rtc1
Fri 03 Jul 2015 07:23:46 AM UTC -0.412788 seconds
root@Vayu-Cape:~# █

```

Note: The updated time would be in UTC.

This will update the time. Check it again using the hwclock command:

hwclock -r -f /dev/rtc1

To start the script at bootup

Create a new script in any folder

```
cd /usr/share/
```

```
mkdir rtc
```

```
nano init.sh
```

Contents of the script

```
#!/bin/bash
```

```
echo ds3231 0x68 > /sys/class/i2c-adapter/i2c-1/new_device
```

```
hwclock -s -f /dev/rtc1
```

```
hwclock -w
```

```
root@Vayu-Cape:~# cat /usr/share/rtc/init.sh
#!/bin/bash

echo ds3231 0x68 > /sys/class/i2c-adapter/i2c-1/new_device
hwclock -s -f /dev/rtc1
hwclock -w
```

Save the script and then use crontab to start the script at the bootup

```
crontab -e
```

```
@reboot bash /usr/share/rtc/init.sh
```

Reboot the system and then check using

```
hwclock -r -f /dev/rtc1
```

It should give the updated time.