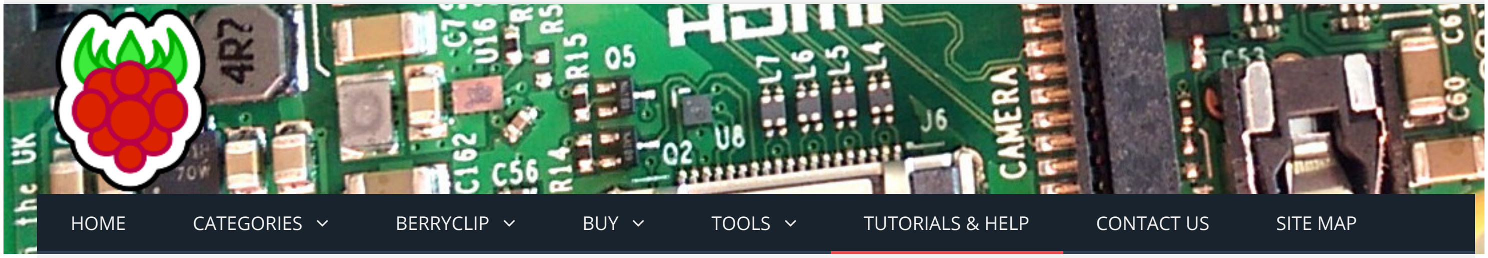
TRENDING IKEA Arcade Table Based On a Raspberry Pi Running RetroPie

Search...



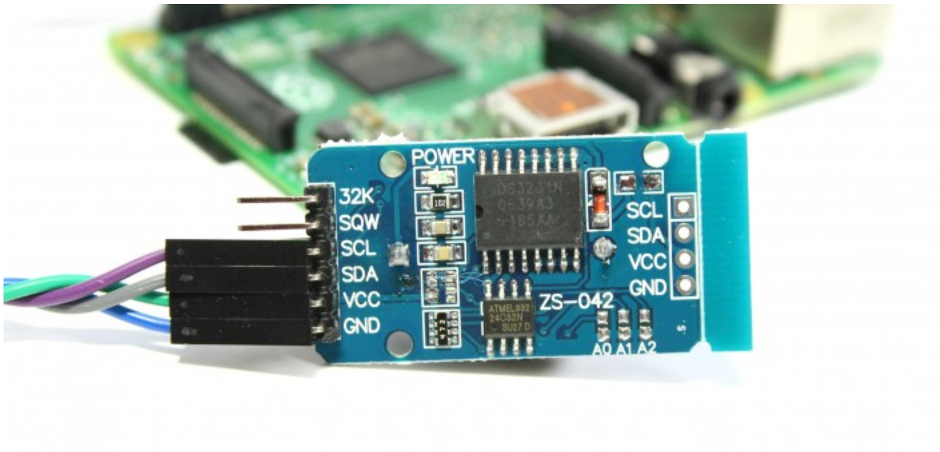
HOME CATEGORIES BERRYCLIP BUY TOOLS TUTORIALS & HELP CONTACT US SITE MAP

YOU ARE AT: Home » Hardware » Interfaces » I2C » Adding a DS3231 Real Time Clock To The Raspberry Pi

Adding a DS3231 Real Time Clock To The Raspberry Pi

22

BY MATT ON MAY 25, 2015 I2C, TUTORIALS & HELP



While your Pi is connected to a network it will be able to set it's clock correctly using NTP. Without a network connection the system time and date will almost certainly be wrong. For some projects this is a problem especially if you are logging timestamps or performing other time sensitive operations.

This can be solved using a Real Time Clock (RTC) module. This will use a small coin cell battery to keep the time for the Pi even if it is turned off. When the Pi reboots it can set it's own internal clock using the time held in by the RTC.

The easiest way of implementing an RTC is to buy a pre-made module. Luckily RTC modules are relatively cheap and easy to obtain. Some modules will attach directly the Pi's GPIO header but these may get in the way of other devices so I prefer more generic modules that keep the GPIO clear for other purposes.

DS1307 or DS3231?

Modules based on the DS1307 and DS3231 chips are popular devices and you'll see them for sale from various retailers. I purchased both types and quickly realised my DS1307 modules were useless. My advice would be to go for a DS3231 based module. They are more accurate and run happily from 3.3V. My two "Tiny RTC" DS1307 modules went straight in the bin.

Resistor Removal

Most generic I2C modules have pull-up resistors on the SDA and SCL pins but these aren't required as the Pi has it's own pull-ups. I didn't want these on-board resistors to interfere with the operation of the I2C bus so I removed them.

RECENT POPULAR NEWS

SEPTEMBER 3, 2017 0

IKEA Arcade Table Based On a Raspberry Pi Running RetroPie

AUGUST 30, 2017 2

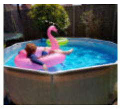
Send Pushover Notification when MotionEyeOS Boots

AUGUST 9, 2017 0

Controlling Energenie Power Sockets with the Pi-mote Addon

JULY 26, 2017 0

KKSB Metal Cases for the Raspberry Pi 2 and 3



JULY 17, 2017 2

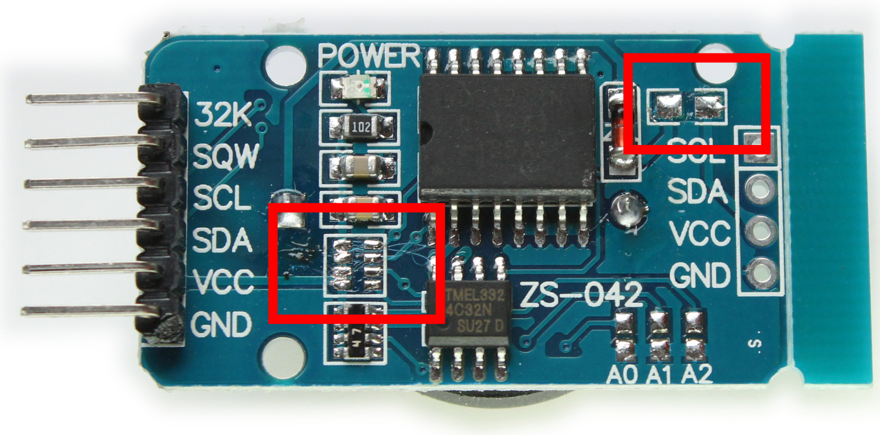
Pool Temperature Monitoring and Pump Control with the Pi Zero W

Search ... Search

CATEGORIES

3D Printing
Add-ons
BBC Micro:bit
BerryClip
Books
Camera Module
Cases
Events
General
Hardware
I2C
Infographics
Minecraft

The module also has a basic charging circuit for use with a rechargeable LIR2032 coin cell. I wanted to use a normal non-rechargeable cell so I removed another resistor to disable this charging circuit. The resistors were removed from the module using a soldering iron and some tweezers or small pliers. The location of the resistors is shown in the photo below :



The pull-up resistors are the block marked in red on the left. The charging circuit is the block marked in red on the right.

I2C Setup

As with all I2C devices you must configure the I2C interface. This is quite easy to do and explained in my Enabling The I2C Interface On The Raspberry Pi tutorial.

DS3231 Module Setup

In order to ensure you've got the latest updates you should run the following commands :

```
sudo apt-get update
sudo apt-get -y upgrade
```

Now we need to modify a system file using :

```
sudo nano /etc/modules
```

If it isn't already there add "rtc-ds1307" to the bottom so it looks something like :

```
snd-bcm2835
i2c-bcm2835
i2c-dev
rtc-ds1307
```

You can save and quit using CTRL-X, Y and ENTER.

Shutdown the Pi using "sudo halt" and remove the power when it has completed the process.

Hardware Setup

Model A+
Model B+
News
Pi 2 Model B
Pi Models
Pi Zero
Power
Programming
Python
Raspbian
RetroGaming
Robotics
Sensors
Software
SPI
Tutorials & Help

TAGS

3D Printing 44780 audio battery berryclip Birthday bluetooth cambridge camera CamJam cpu DigiMakers display games GPIO i2c Kickstarter lcd LED Linux media Minecraft Model A Model B motionEyeOS pcb photography photos Pi-Lite portable power python Raspberry Pi Bootcamp raspbian revision sd card security sensor soldering SPI temperature ultrasonic usb video WiFi

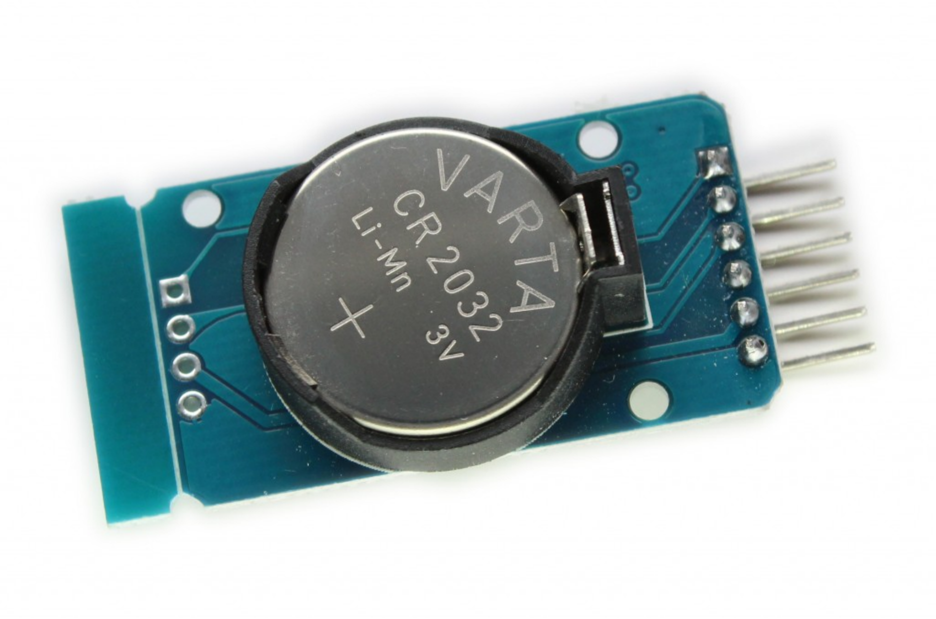
RASPBERRY PI RELATED

Averageman Vs Pi
MattsBits – Pi Resources
Official RaspBerry Pi Site
Raspberry Pi Pod
RasPi.tv
Raspihub.com

With the Raspberry Pi turned off you can now connect the module. It's a fairly simple configuration but it needs to be correct so double check it. Refer to my GPIO Diagram to ensure you get the correct pins on the Pi's header.

DS1307	Pi GPIO
GND	P1-06
Vcc	P1-01 (3.3V)
SDA	P1-03 (I2C SDA)
SCL	P1-05 (I2C SCL)

Don't forget the 3V battery!



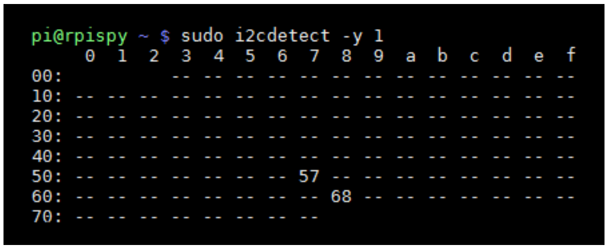
Interface Test

Power up the Pi and run the following command :

```
sudo i2cdetect -y 1
```

Note : If you are using a Rev 1 you will need to use "sudo i2cdetect -y 0".

You should see something similar to this :



```
pi@rpispy ~ $ 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: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
50: -- -- -- -- -- -- -- 57 -- -- -- -- -- -- -- --
60: -- -- -- -- -- -- -- -- 68 -- -- -- -- -- -- --
70: -- -- -- -- -- -- -- --
```

In this example "68" is the hex address of the RTC module on the I2C interface.

I2C Device Setup

To ensure the DS1307 device is setup and the time synchronised when the Pi boots we need to edit another system file :

```
sudo nano /etc/rc.local
```

RaspTut

Rastrack.co.uk

RPi School

TECH RESOURCES

How2Code

Microbit Spy

Technology Spy

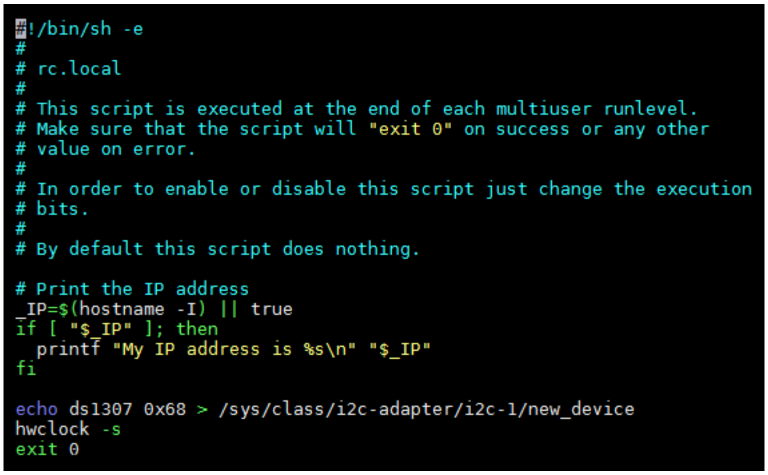
ARCHIVES

Archives Select Month ▾

Add the following two lines before the exit 0 line :

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

so it looks something like :



```
#!/bin/sh -e
#
# rc.local
#
# This script is executed at the end of each multiuser runlevel.
# Make sure that the script will "exit 0" on success or any other
# value on error.
#
# In order to enable or disable this script just change the execution
# bits.
#
# 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-1/new_device
hwclock -s
exit 0
```

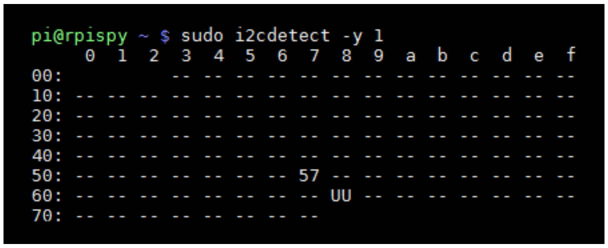
Note : If you are using a Rev 1 you will need to use "/i2c-0/" instead of "/i2c-1/".

You can save and quit using CTRL-X, Y and ENTER.

Now reboot the Pi using :

```
sudo reboot
```

Now when you repeat the i2cdetect command (see above) the 68 will turn into UU :



```
pi@rpispy ~ $ 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: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
50: -- -- -- -- -- -- -- 57 -- -- -- -- -- -- -- --
60: -- -- -- -- -- -- -- -- UU -- -- -- -- -- -- --
70: -- -- -- -- -- -- -- --
```

Time Zones and Daylight Saving Time

By default the Pi tends to show time as GMT or UTC. For me sat in Bristol this results in the time being 1 hour behind my local time. To tell your Pi what region you are in you can use :

```
sudo raspi-config
```

and then select "Internationalisation Options" followed by "Change Timezone". You can then select you location from the following screens.

I set my location to "Europe/London" and this results in "date" showing the time in BST (British Summer Time) rather than UTC. The time is then correctly adjusted by +1 hour.

Reading The Date And Time

You can read the Pi's system time using :

```
date
```

If you need to set the system time for any reason you can use the following command :

```
sudo date -s "29 AUG 1997 13:00:00"
```

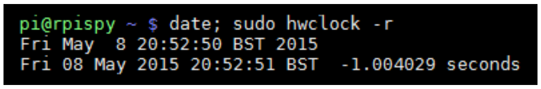
Once correct you can write the system date and time to the RTC module using :

```
sudo hwclock -w
```

You should be able to read the date and time back from the RTC using :

```
sudo hwclock -r
```

By separating the commands with a semi-colon you can read back the system time and the RTC time at the same time. Hopefully they should match and look something like this :



```
pi@rpispy ~ $ date; sudo hwclock -r
Fri May  8 20:52:50 BST 2015
Fri 08 May 2015 20:52:51 BST  -1.004029 seconds
```

The "hwclock -s" we added to "rc.local" sets the system time from the RTC module.

The Final Test

The final test is to determine if the RTC module is keeping time and that the Pi will use that time when it boots. The best way to do that is to :

- Power down the Pi
- Remove the power cable
- Remove the network connection
- Attach the Pi to a monitor and keyboard
- Leave it overnight
- Power it up and use "date" to see what time the Pi thinks it is

Hopefully your Pi is now displaying the correct date and time and will maintain it when the Pi is powered down.

SHARE.

< PREVIOUS ARTICLE

Using An I2C Enabled LCD Screen With The Raspberry Pi

NEXT ARTICLE >

Raspberry Pi Used In New Point Break Movie

RELATED POSTS

AUGUST 30, 2017 2

Send Pushover Notification when MotionEyeOS Boots

JULY 17, 2017 2

Pool Temperature Monitoring and Pump Control with the Pi Zero W

JULY 2, 2017 6

Create a Basic Python Web Server with Flask

JUNE 4, 2017 0

Installing the YouTube Video Plugin for Kodi

JUNE 3, 2017 1

Installing the Exodus Streaming Video Plugin for Kodi

MAY 31, 2017 2

How to Create a New SD Card for Raspberry Pi on Windows

MAY 30, 2017 0

Monitoring Raspberry Pi GPIO with MotionEyeOS

APRIL 24, 2017 0

Detection Area Masking in motionEyeOS on the Raspberry Pi

APRIL 18, 2017 4

Raspberry Pi CCTV Camera with motionEyeOS

22 COMMENTS

FAYE on JULY 30, 2015 8:10 AM

Hi, you say the 1307 is pretty worthless but in your scripting you keep using it. Does the 3231 use the same libraries as the 1307 ? Or should i add ds3231 as a module?

REPLY

MATT on JULY 30, 2015 11:24 AM

The 3231 can use the same libraries. You may be able to replace references to “1307” with “3231” but I didn’t get a chance to test that.

REPLY

JOHN on AUGUST 16, 2015 4:17 AM

Thanks for sharing. Could you please answer these questions: 1) why did you disable LIR recharge feed for a CR battery ? 2) why not use ntp (more accurate) to setup system time/date and then set hwclk ? 3) do you really need to remove the pull-up resistor ?

REPLY

MATT on AUGUST 16, 2015 8:46 PM

1) I couldn’t find a definitive source of information that would confirm it was safe to recharge a standard coin cell that wasn’t specifically rechargeable. So I was playing it safe.
2) You could use any technique to set the time. I just went for something basic for the sake of this tutorial. For my own experiments I tend to just connect the Pi to ethernet and let the time sync with ntp, then set the RTC module from that.
3) I’m sure it would work without removing the resistors but if you are using multiple I2C modules that all have their-own pull-ups it can cause intermittent problems.

REPLY

TYLER on MAY 11, 2016 2:58 AM

John, Matt-

Thanks for sharing – this is just what I needed!

-Tyler from Rochester NY USA

REPLY

PRAILELE on AUGUST 29, 2015 10:10 PM

Hi, which of the two red blocks (resistors removed) is the pull-up?
Thanks

REPLY ›

MATT on AUGUST 29, 2015 11:51 PM

I've updated the post text. The pull-ups are on the left, the charging circuit on the right.

REPLY ›

DOUG on OCTOBER 10, 2015 1:57 AM

Thanks, this is great. My Pi now always knows what the time is.

REPLY ›

SUPRA on DECEMBER 15, 2015 12:23 PM

Can I add this to replace ds1307 to ds3231?
sudo nano /etc/modules

snd-bcm2835
i2c-bcm2835
i2c-dev
rtc-dsds3231

REPLY ›

BUSEL7 on DECEMBER 22, 2015 4:59 PM

I've got problem with this RTC. When reboot sometimes the clock and date not set correctly. Date and clock showing "01 Jan 1970 07:00:00"

REPLY ›

N00B on JANUARY 28, 2016 5:39 PM

If I remove the pull-up resistors and connect VCC to 5V pin on the Pi, that would be fine?

REPLY ›

MATT on FEBRUARY 2, 2016 7:06 PM

It's hard to say. Powering it from 5V may result in the i2C pins outputting 5V.

REPLY ›

BS50 on FEBRUARY 23, 2016 10:04 AM

Hi! Is the RTC board in your photos a commercial one? Could you tell me the name or where I can buy it?

REPLY ›

MATT on FEBRUARY 28, 2016 12:11 AM

It was from eBay.

REPLY ›

ZOMBIEPROCESSES on APRIL 20, 2016 3:55 PM

Amazon has them as well.

REPLY ›

PRADEEO on MARCH 13, 2016 10:34 PM

I like the way you describe things especially what sensors you used and where from you bought. I couldn't find the link to eBay to buy this RTC clock in this article.

REPLY ›

MATT on MARCH 19, 2016 11:49 AM

If you search eBay you will find plenty of them. I don't usually provide links to specific items on eBay because they go out of date so quickly.

REPLY ›

PRADEEP KUMAR on APRIL 9, 2016 8:55 PM

thanks for the reply. I bought one from ebay.
Now my question is, if I don't want to remove those two resistors, will it still work?
Actually I am a bit hesitant to touch the factory made circuit.

REPLY ›

MATT on APRIL 11, 2016 8:33 PM

It would probably work with them still there. The trouble with additional pull-ups is that act in parallel with the Pi's pull-ups. This lowers the overall resistance and can increase current draw when the pins are pulled low. However for a a single device it doesn't seem to make much difference. It is often discussed in forums and there isn't a definitive answer. I'm sure I probably got the device working with them on but in my tutorials I tend to play it safe if I'm not 100% sure.

REPLY ›

SANJEEV on APRIL 3, 2016 9:47 AM

Thanks MATT, I have followed the instructions step by step and could successfully configure the external clock. The steps are beautifully written and easy to follow. Though I have not removed any register, working fine for me.

REPLY ›

PRADEEP KUMAR on MAY 14, 2016 4:31 AM

For some my RTC module (bought from ebay) is not holding time. It goes back to 1999. Pi reads it and shows 1999 too, so Pi-RTC setup is fine but RTC not holding time 🙁

REPLY ›

JORGE VELEZ on DECEMBER 28, 2016 1:19 AM

Great MAN!!! This answer helped me with my chinese Suptronics x300 expansion board using ubuntu 16.04 server preinstalled image… Just wanted to leave this comment so people knew it is the right answer…

Just one doubt, why do we use the ds1307 module instead of the ds3231?

REPLY ›

LEAVE A REPLY

Your Comment

Your Name

Your Email

Your Website

POST COMMENT

ABOUT

Unofficial site devoted to the Raspberry Pi credit card sized computer offering tutorials, guides, resources,scripts and downloads. We hope to help everyone get the most out of their Pi by providing clear, simple articles on configuring, programming and operating it.

POPULAR POSTS

JULY 27, 2012 97
16×2 LCD Module Control Using Python

OCTOBER 20, 2013 96
Analogue Sensors On The Raspberry Pi Using An MCP3008

SEPTEMBER 19, 2014 82
Top 5 Reasons The Raspberry Pi Sucks

RECENT POSTS

SEPTEMBER 3, 2017 0
IKEA Arcade Table Based On a Raspberry Pi Running RetroPie

AUGUST 30, 2017 2
Send Pushover Notification when MotionEyeOS Boots

AUGUST 9, 2017 0
Controlling Energenie Power Sockets with the Pi-mote Addon

Entries RSS | Comments RSS

This site is not associated with the official Raspberrypi.org site or the Raspberry Pi Foundation. Raspberry Pi is a trademark of the Raspberry Pi Foundation.