Would it be possible to use the DS1302 made for Arduino?

I got some spare DS1302 chips that I'd like to connect to my Raspberry Pi to use as a RTC, but I'm not so sure h
I go about it?

gpio hardware rtc

edited Dec 17 '13 at 1:33

syb0rg
3,781 1 14 39

asked Dec 16 '13 at 7:48

Ivan
46 5

2 So, you have answered your own question right? If so, you should put this as an answer, not inside of a question. Right now it's unclear if you have any unanswered question. – Krzysztof Adamski Dec 16 '13 at 10:21

hey there @KrzysztofAdamski, i can't because i just made this account. something about having to wait 8 hours – Ivan Dec 16 '13 at 10:31

OK, just be sure to move it to the answer when you can. You can even self accept it. This will help others having the same issue. Also note that DS1302 is supported by Linux kernel driver and this is another option to use it. I may add another answer describing how to do this when I have some time or you may explore this yourself. – Krzysztof Adamski Dec 16 '13 at 11:20

@KrzysztofAdamski, sure thing. – Ivan Dec 16 '13 at 13:49

You can also use the DS1307. It's easy to interface because you can just use I2C – John La Rooy Dec 17 '13 at 23:10

1 Answer

Using the guide from hobbytronics as a reference, with this module i got from Cytron

1. Connect it to the RPi's GPIO

```
RST  (CE)  -> GPIO#17
I/O  (DAT) -> GPIO#18
SCLK (CLK) -> GPIO#21
GND        -> GND
VCC        -> 3v
```

2. compile source

http://pastebin.com/gF09XSFn

```
cc rtc-pi.c
```

this is for the rev 1 board, mine was the rev 2 so i had to update the definitions in the source:

http://pastebin.com/YJ22Sayh

```
#define SCLK_OUTPUT *(gpio+GPIO_SEL2) &= 0xFF1FFFFFL; *(gpio+GPIO_SEL2) |=0x00200000L
#define SCLK_HIGH *(gpio+GPIO_SET) = 0x08000000L
#define SCLK_LOW *(gpio+GPIO_CLR) = 0x08000000L
```

3. execute the compilation

output from the compilation would be a a.out file, rename it to rtc-pi and set the date with something like this

```
sudo ./rtc-pi 20131216175500
```

or to set it from the current time

```
sudo ./rtc-pi `date +"%Y%m%d%H%M%S"`
```

this will set the time. now all you need to do is to invoke it during every startup to keep the date

```
sudo ./rtc-pi
```

Misc

to quote the guide from hobbytronics

> I also found that a pullup resistor is needed between DAT and VCC (= 3.3V). A 10k..30k resistor seems to work fine.

Now i didn't do this at the moment because i don't have any resistors with me, not so sure about the impact of it.

edited Apr 25 at 19:29

answered Dec 16 '13 at 18:31

Ivan

46 5

Now you should accept this answer! – syb0rg Dec 16 '13 at 23:07

a resistor between DAT and VCC makes sure your I2C setup conforms to the standard -- DAT should be HIGH when there's no data transfer going on. – lenik Dec 16 '13 at 23:46