Raspberry Pi NTP Server with RTC

status active

Raspberry Pi NTP Server with RTC

📝 Table of Contents

😎 About

This repo contains circuit, firmware and backend for Raspberry Pi NTP Server with RTC Project.

🏁 Getting Started

These instructions will get you a copy of the project up and running on your local machine for development and testing purposes. See deployment for notes on how to deploy the project on a live system.

Prerequisites

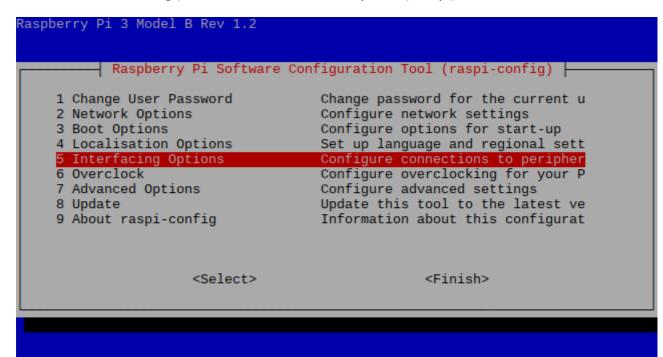What things you need to install the software and how to install them.

```
- Raspberry Pi Model 3B, 3B+, 4B or CM4
```

Installation and Configuration

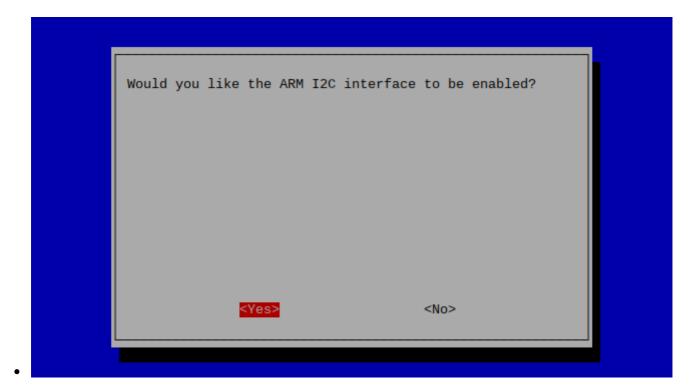A step by step series that covers how to get the Firmware running.
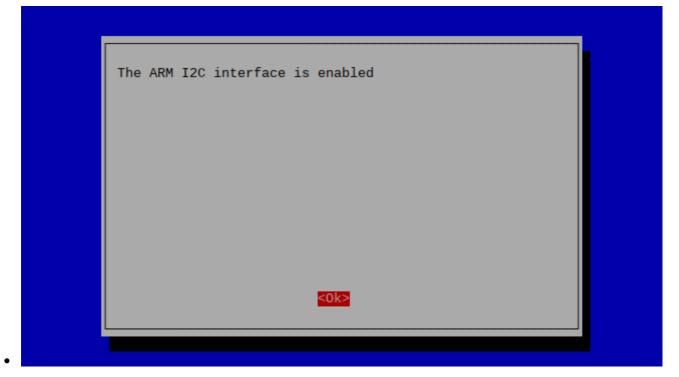
Raspberry Pi Firmware Pre-Reqs

1. Download and install the latest Raspberry Pi OS Desktop image to your SD card
2. Open the terminal and execute the following command `sudo raspi-config`
3. Then follow the following pictures to enable I2C bus on you raspberry pi
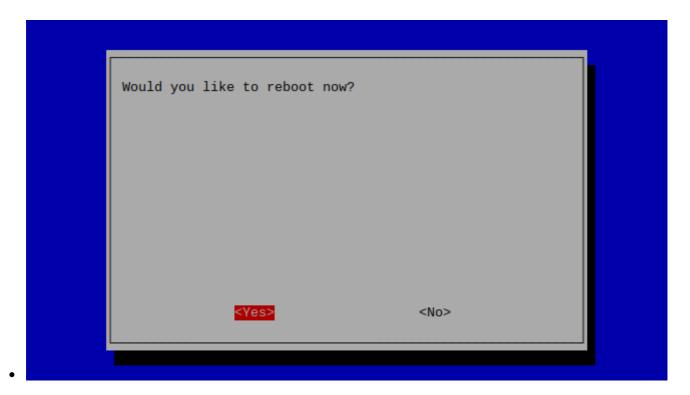
-
```
Raspberry Pi 3 Model B Rev 1.2


        ┤ Raspberry Pi Software Configuration Tool (raspi-config) ├

      1 Change User Password         Change password for the current u
      2 Network Options              Configure network settings
      3 Boot Options                 Configure options for start-up
      4 Localisation Options         Set up language and regional sett
      5 Interfacing Options          Configure connections to peripher
      6 Overclock                    Configure overclocking for your P
      7 Advanced Options             Configure advanced settings
      8 Update                       Update this tool to the latest ve
      9 About raspi-config           Information about this configurat


               <Select>                        <Finish>
```

-
```
        ┤ Raspberry Pi Software Configuration Tool (raspi-config) ├

      P1 Camera                      Enable/Disable connection to the
      P2 SSH                         Enable/Disable remote command lin
      P3 VNC                         Enable/Disable graphical remote a
      P4 SPI                         Enable/Disable automatic loading
      P5 I2C                         Enable/Disable automatic loading
      P6 Serial                      Enable/Disable shell and kernel m
      P7 1-Wire                      Enable/Disable one-wire interface
      P8 Remote GPIO                 Enable/Disable remote access to G


               <Select>                        <Back>
```
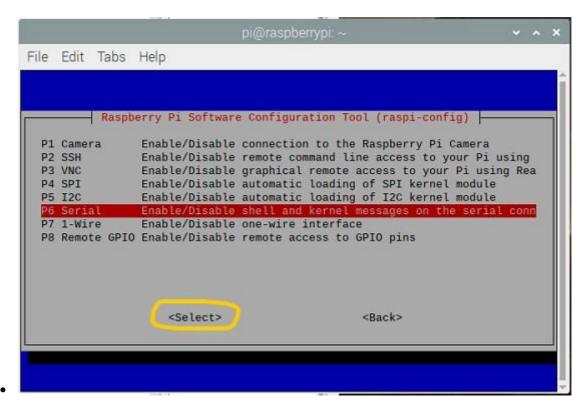
-

-

- Then do the same for Serial(UART)

Configuring Raspberry Pi

DS3231 Configurations

1. Copy Firmware folder to the desktop of your Raspberry Pi, open the terminal of your Raspberry Pi and execute the following commands

- `sudo apt-get update`
- `sudo apt-get upgrade`
- `sudo apt install python3-pip`

- `sudo apt-get -y remove fake-hwclock`
- `sudo update-rc.d -f fake-hwclock remove`
- `sudo systemctl disable fake-hwclock`
- `sudo nano /boot/config.txt`
- At the end of the file add this line `dtoverlay=i2c-rtc,ds3231`
- Then press CTRL+O and CTRL+X to save and exit
- `sudo nano /lib/udev/hwclock-set`
- and put # sign before the following lines

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

- Then press CTRL+O and CTRL+X to save and exit
- `sudo hwclock -r`
- You might see that the date/time are invalid, connect your raspberry pi to the internet and run the following commands
- `sudo hwclock -w`
- `sudo hwclock -r`
- `sudo reboot`

Now your Raspberry Pi is using external RTC module for time.

NTP Server Configurations

Execute the following commands to set NTP

- `sudo apt-get install ntp`
- `/etc/init.d/ntp stop`
- `/etc/init.d/ntp start`
- Get the NTP server links of your country from pool.ntp.org
- `sudo nano /etc/ntp.conf`
- Replace the lines starting with `server` with the server links you got in the above step.
- Then press CTRL+O and CTRL+X to save and exit
- `/etc/init.d/ntp restart`
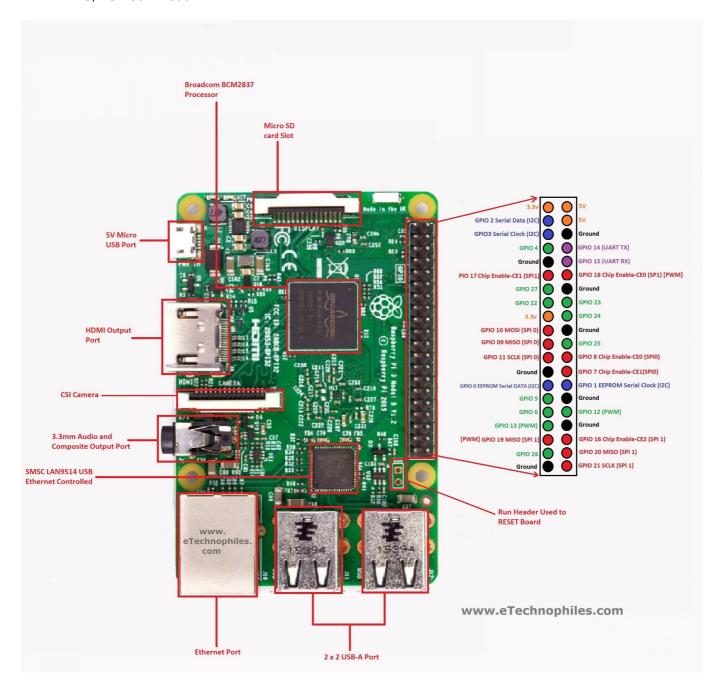- `sudo systemctl enable ntp.service`
- `sudo reboot`

Now you can use your rasberry pi IP address in your ESP32 to get the time or you can also use raspberrypi.local instead of the IP address.

🔨 Testing

1. The Firmware can be tested on Raspberry Pi 3B, 3B+ or 4B with the following modifications
2. Connect the sensor as shown in the Circuit Diagram section below.
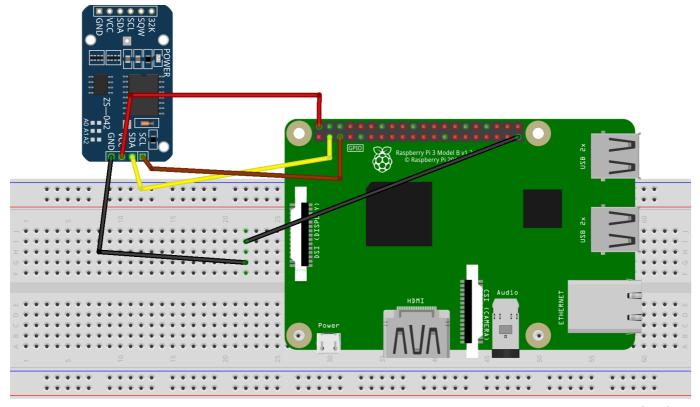
🔌 Circuit Diagram

- RPi 3,4 GPIOs Pinout

Circuit

> Pins connections

RTC DS3231	Raspberry Pi
SCL	GPIO2
SDA	GPIO3
VCC	5V
GND	GND

Components Used

1. Any Raspberry Pi (https://www.amazon.com/CanaKit-Raspberry-Micro-Supply-Listed/dp/B01C6FFNY4/ref=sr_1_1?dchild=1&keywords=raspberry+pi+3&qid=1632029848&sr=8-1)
2. RTC DS3231(https://www.amazon.de/Echtzeit-Uhr-Modul-RTC-Sensor-Pr%C3%A4zision-AT24C32-Raspberry/dp/B07V68443F/ref=sr_1_5?__mk_de_DE=%C3%85M%C3%85%C5%BD%C3%95%C3%91&keywords=ds3231&qid=1636619193&sr=8-5)
3. MCP3008
4. Logic Level Converter(https://www.amazon.com/SparkFun-Logic-Level-Converter-Bi-Directional/dp/B01N30ZCW9/ref=sr_1_6?crid=2NOGIA43AG9OS&dchild=1&keywords=logic+level+converter&qid=1632029917&sprefix=logic+level%2Caps%2C463&sr=8-6)

⚒ Built Using

- Python3 - Raspberry Pi FW

✎ Authors

- @Nauman3S - Development and Deployment