

EPSON RX8804CE

Linux driver overview.

Ver-1.0

Category	General
Registered number	OUT-18-3819
Publisher	RTC Tech-Support DEVICE STRATEGIC PLANNING DEPARTMENT SEIKO EPSON

Contents

1.	INTRODUCTION	- 1 -
2.	RASPBERRY PI 3.....	- 2 -
3.	HARDWARE	- 4 -
3.1	PIN CONNECTION	- 5 -
4.	SOFTWARE	- 6 -
4.1	DEVICE TREE.....	- 6 -
4.2	DEVICE DRIVER.....	- 7 -

Update history

Ver	Date	Contents
1.0	Dec 18 2018	First release.

1. Introduction

This document was developed by Raspberry PI3 for the EPSON RX8804 RTC Driver.

Environment

The kernel version was developed in the environment of 4.14.XX and was written to load drivers,

Raspberry-B'd. pi3 Model B V1.2
Description : Raspbian GNU / Linux 9.6
Release : 9.6
Codename : stretch

2. Raspberry PI 3

This is related to the setup of the Raspberry PI3 board before testing the driver.
Use raspi-config for Raspberry PI I2C settings.

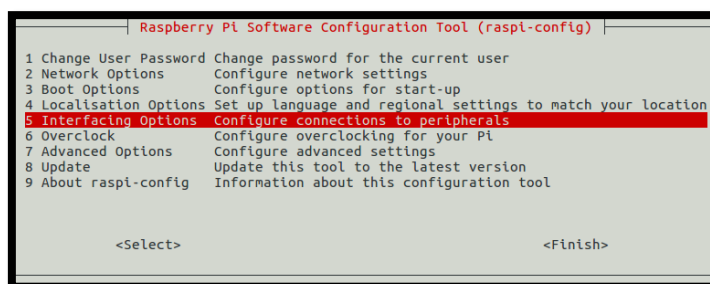


Figure 1

Select I2C.

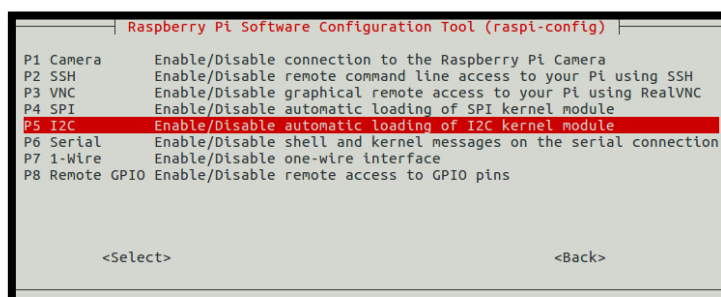


Figure 2

Select I2C to activate it.

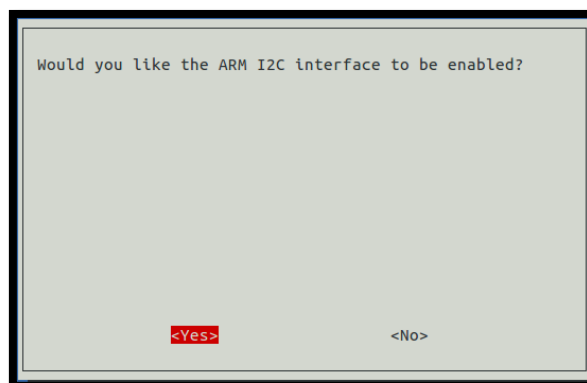


Figure 3

I2C has been activated.

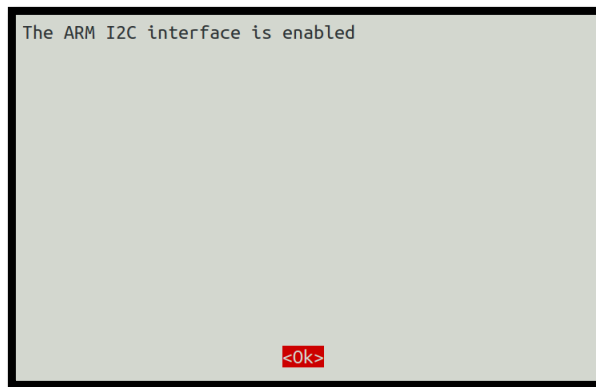


Figure 4

For I2C client registration, enter the command in Raspberry PI.

```
$ echo "rx8804 0x32">/sys/class/i2c-adapter/i2c-1/new_device
```

To use the device tree, add a dtoverlay entry to /boot/config.txt as shown below.

```
dtoverlay=rx8804  
"/boot/config.txt" 58L, 1607C
```

Figure 5

3. Hardware

The hardware is configured according to the table below in the RX8804 Data Sheet.

3.2. Pin Functions

Signal name	I/O	Function
SDA	I/O	This pin's signal is used for input and output of address, data, and ACK bits, synchronized with the serial clock used for I ² C communications. Since the SDA pin is an N-ch open drain pin during output, be sure to connect a suitable pull-up resistance relative to the signal line capacity.
SCL	Input	This is the serial clock input pin for I ² C Bus communications.
FOUT	Output	This is the C-MOS output pin with output control provided via the FOE pin. When FOE = "H" (high level), this pin outputs a 32.768 kHz signal. (depend on FSEL bit) When output is stopped, the FOUT pin = Hi-Z (high impedance).
FOE	Input	This is an input pin used to control the output mode of the FOUT pin. When this pin's level is high, the FOUT pin is in output mode. When it is low, output via the FOUT pin is stopped.
/ INT	Output	This pin is used to output alarm signals, timer signals, time update signals, and other signals. This pin is an open drain pin.
EVIN	Input	Trigger input terminal for time-stamps. Built in disconnectable Pull-up resistor.
SOUT	Output	SOUT is push-pull for the inside state output. SOUT outputs state of a specified flag bit or selected logical 1 or 0.
VDD	–	This pin is connected to a positive power supply.
GND	–	This pin is connected to a ground.
T2	–	Use only for testing in the factory. (Do not connect externally.)

Note: Be sure to connect a bypass capacitor rated at least 0.1 μ F between VDD and GND.

Table 1 RX8804 Pin Function

Since we tested using the Raspberry PI 3 board, the pin connection standards are explained in the Raspberry PI 3 standard.



Figure 6 Raspberry Pi 3 Model B V1.2

3.1 Pin Connection

This is the connection between the RX 8804 chip and the raspberry Pi 3 board.










RX8804CE	Raspberry PI3	
FOE	PIN 11, GPIO17	
VDD	PIN 01, 3.3v	
EVIN	PIN 15, GPIO22	
FOUT	PIN 07, GPIO04	
SCL	PIN 05, GPIO03	
SOUT	PIN 19, GPIO10	
SDA	PIN 03, GPIO02	
GND	PIN 09, Ground	
/INT	PIN 13, GPIO27	

Table 2























Raspberry Pi 3 GPIO Header			
Pin#	NAME		NAME Pin#
01	 3.3v DC Power		DC Power 5v 02
03	 GPIO02 (SDA1 , I ² C)		DC Power 5v 04
05	 GPIO03 (SCL1 , I ² C)		Ground 06
07	 GPIO04 (GPIO_GCLK)		(TXD0) GPIO14 08
09	 Ground		(RXD0) GPIO15 10
11	 GPIO17 (GPIO_GEN0)		(GPIO_GEN1) GPIO18 12
13	 GPIO27 (GPIO_GEN2)		Ground 14
15	 GPIO22 (GPIO_GEN3)		(GPIO_GEN4) GPIO23 16
17	3.3v DC Power		(GPIO_GEN5) GPIO24 18
19	 GPIO10 (SPI_MOSI)		Ground 20
21	GPIO09 (SPI_MISO)		(GPIO_GEN6) GPIO25 22
23	GPIO11 (SPI_CLK)		(SPI_CE0_N) GPIO08 24
25	Ground		(SPI_CE1_N) GPIO07 26

Figure 7

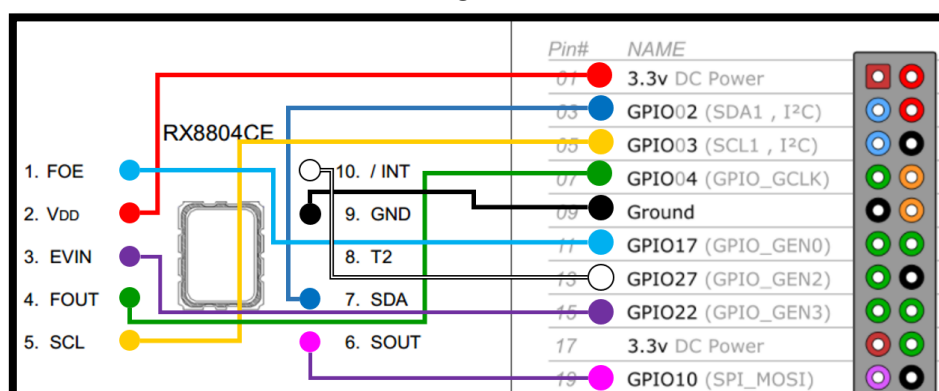


Figure 8

4. Software

Requires driver compilation, device tree modification, and compilation.

Driver is set to be compiled in the form of modules,

The device tree is include in the kernel for compile and use.

4.1 Device Tree

The device tree used an overlay of raspberry pi.

Add the rx8804.dtbo entry to the arch/arm/boot/dts/overlays/Makefile file.

```
rx8804.dtbo \
```

Copy the compiled rx8804.dtbo file to the /boot/overlay directory.

And reboot to apply the device tree.

```
// Device tree overlay for I2C RTC connected rx8804
```

```
/dts-v1/;
```

```
/plugin/;
```

```
/ {
```

```
    compatible = "brcm,bcm2708";
```

```
    fragment@0 {
```

```
        target = <&gpio>;
```

```
        __overlay__ {
```

```
            rx8804_pins: rx8804_pins {
```

```
                brcm,pins = <27>;
```

```
                brcm,function = <0>; // in
```

```
                brcm,pull = <2>; // pull-up
```

```
            };
```

```
        };
```

```
    };
```

```
    fragment@1 {
```

```
        target = <&i2c1>;
```

```
        __overlay__ {
```

```
            #address-cells = <1>;
```

```
            #size-cells = <0>;
```

```
            status = "okay";
```



```

rx8804: rx8804@32 {
    compatible = "epson,rx8804";
    reg = <0x32>;
    pinctrl-names = "default";
    pinctrl-0 = <&rx8804_pins>;
    interrupt-parent = <&gpio>;
    interrupts = <27 2>;
    irq-gpios = <&gpio 27 0>;
};

};

__overrides__ {
    interrupt = <&rx8804_pins>,"brcm,pins:0",
    <&rx8804>,"interrupts:0",
    <&rx8804>,"irq-gpios:27";
};
};

```

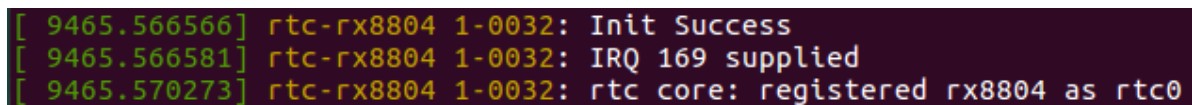
4.2 Device Driver

If the driver has been built statically in the kernel, nothing has to be done. If it is built as a module, then load it with

```
$ insmod rtc-rx8804.ko
```

On initialization, the driver will log the following message

```
$ dmesg | grep rtc-rx8804
```



```

[ 9465.566566] rtc-rx8804 1-0032: Init Success
[ 9465.566581] rtc-rx8804 1-0032: IRQ 169 supplied
[ 9465.570273] rtc-rx8804 1-0032: rtc core: registered rx8804 as rtc0

```

Figure 9

This means that rtc0 is the RX8804 rtc.