GM8136

REAL TIME CLOCK

User Guide
Rev.: 1.0
Issue Date: August 2014

REVISION HISTORY

GM8136 RTC User Guide

Date	Rev.	From	To
Aug. 2014	1.0	-	Original

Printed in Taiwan 2014

Grain Media, Inc.
5F, No. 5, Li-Hsin Road III, Hsinchu Science Park, Hsinchu City, Taiwan 300, R.O.C.

Grain Media's home page can be found at:
http://www.grain-media.com

TABLE OF CONTENTS

Chapter 1 RTC Overview 1
Chapter 2 Configuration of RTC in Linux 3
Chapter 3 RTC Related Files 9
Chapter 4 External RTC Driver 11

LIST OF FIGURES

Figure 2-1. Device Driver Entry 4
Figure 2-2. Real Time Clock Entry 5
Figure 2-3. Real Time Clock Options 6
Figure 2-4. Insert RTC Module 7
Figure 2-5. Usage of hwclock 7

Chapter 1

RTC Overview

Real Time Clock (RTC) is a hardware clock. This clock can be used for time keeping even when the device is powered off. A battery should be attached to the board to keep RTC alive.

Grain Media provides a RTC IP, which is built in GM8136 and constructs a functional RTC driver based on the Linux standard. With RTC, users can easily use the common utilities, such as hwclock, to obtain the correct time and the time setting.

Chapter 2

Configuration of RTC in Linux

Before using the RTC functions of GM8136, users should make sure that the following options in Linux kernel are all enabled:

Device Drivers

- Real Time Clock
  - /sys/class/rtc/rtcN (sysfs)
  - /proc/driver/rtc (procfs for rtc0)
  - /dev/rtcN (character devices)

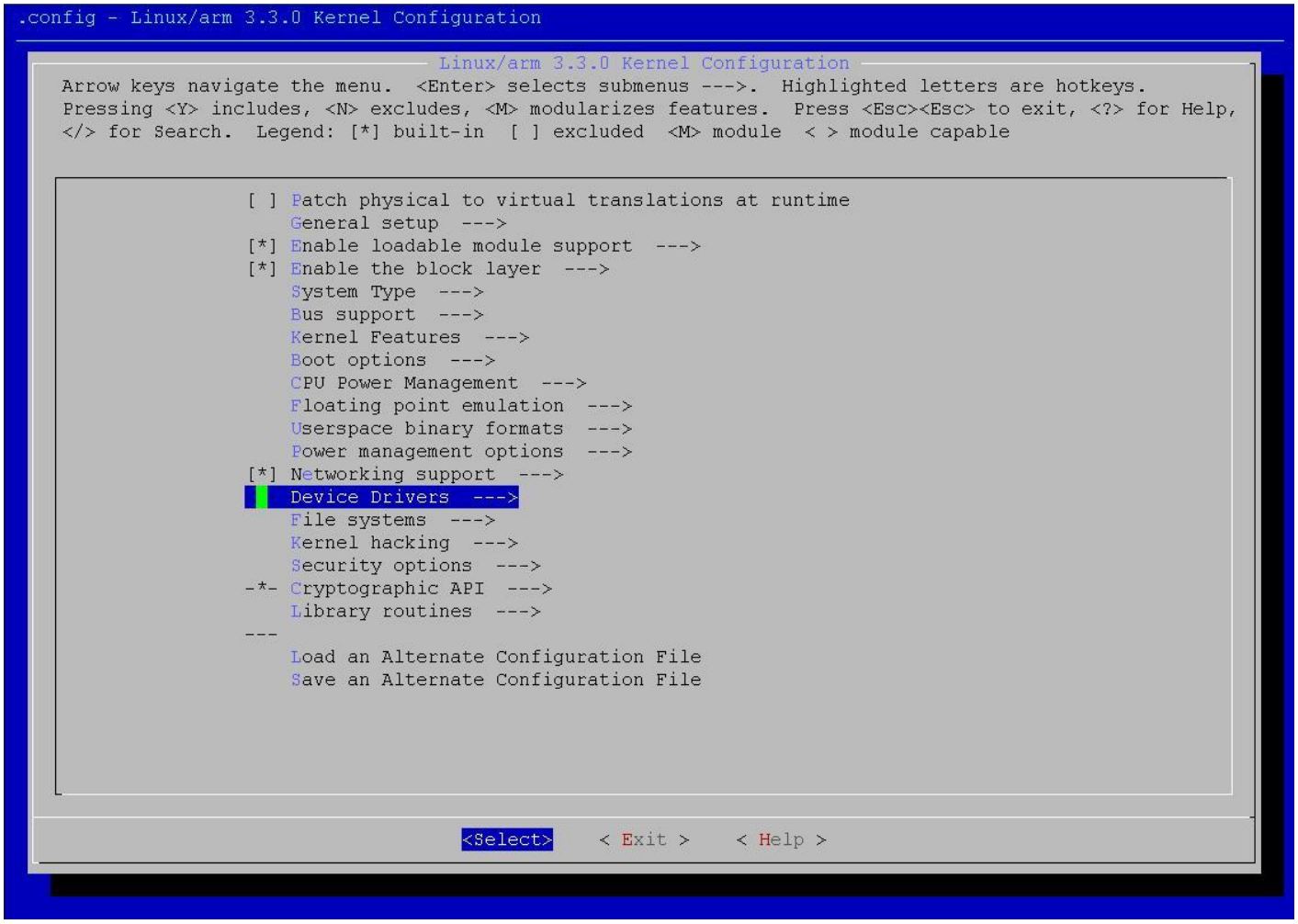


Figure 2-1. Device Driver Entry

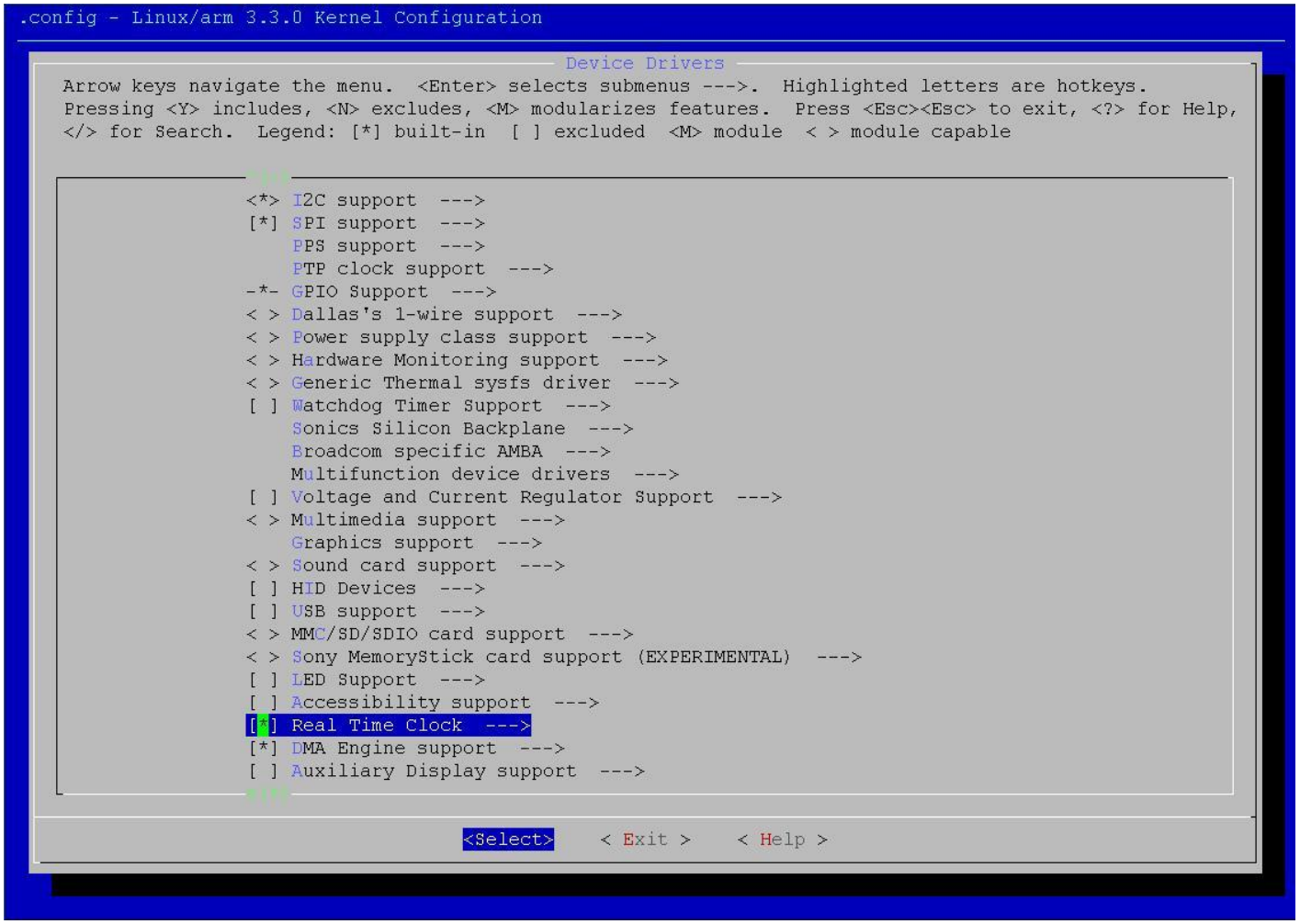


Figure 2-2. Real Time Clock Entry

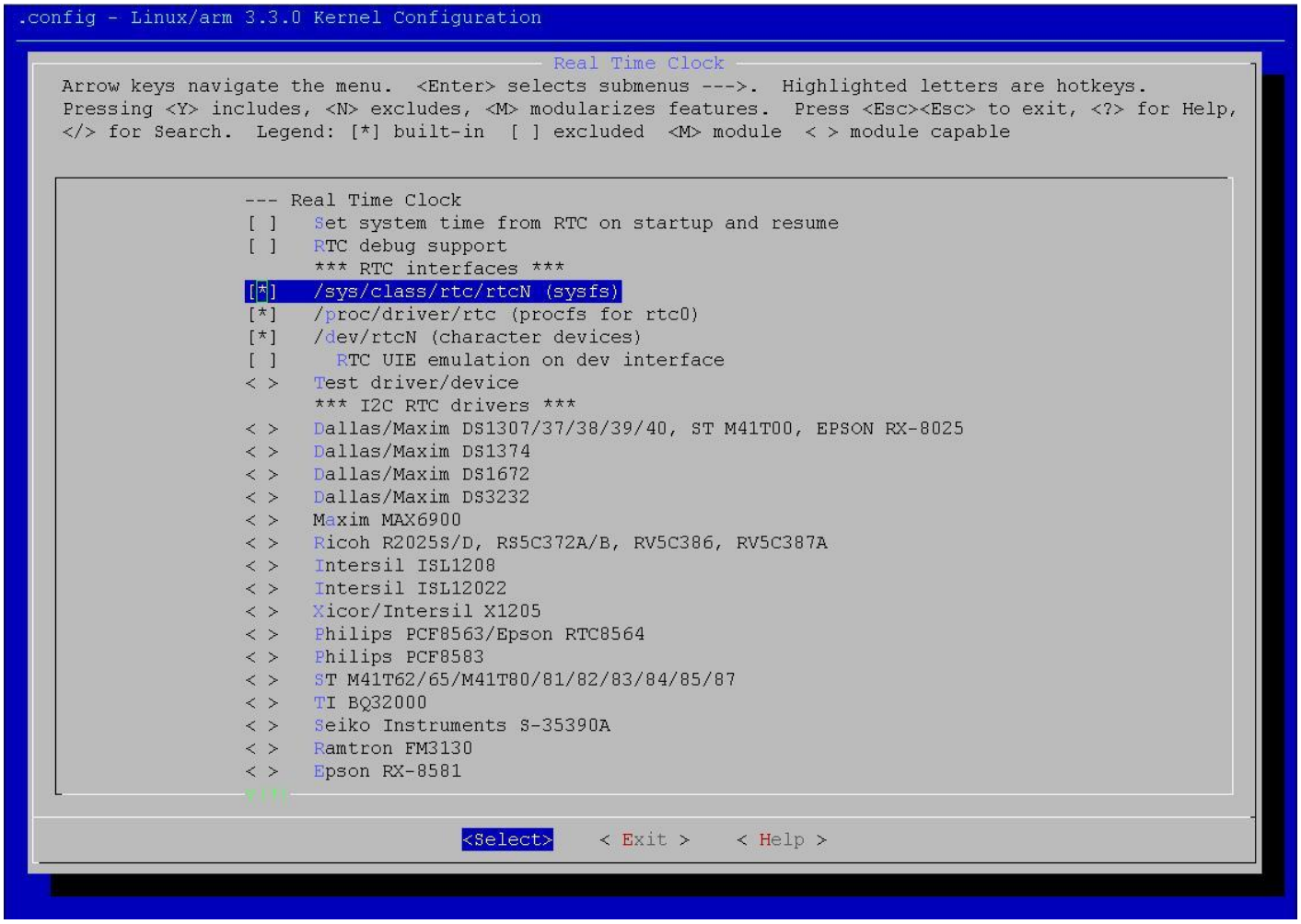


Figure 2-3. Real Time Clock Options

After completing the configuration, users should switch the current directory to arm-linux-3.3/module/RTC/FTRTC011 and make a copy of the RTC kernel module, rtc-ftrtc011.ko. Please use "insmod" to insert this module into kernel and use "mdev –s" to generate a RTC device node as shown in Figure 2-4.

The module parameter, clk_src, is used to control the RTC clock source. Users can apply different RTC clock sources when inserting the rtc-ftrtc011.ko. The clk_src definition is shown in the below table.

Name	Default Value	Description
clk_src	0	RTC Clock Source Selection 0: External OSC 32.768KHz 1: Internal PLL3 (540MHz) -> 32.768KHz

```
/lib/modules # insmod rtc-ftrtc011.ko
ftrtc011 ftrtc011: rtc core: registered ftrtc011 as rtc0
/lib/modules # mdev -s
/lib/modules # ls -lh /dev/rtc0
crw-rw----    1 root     root     254,   0 Jan  1 00:35 /dev/rtc0
/lib/modules #
```

Figure 2-4. Insert RTC Module

The GM8136 RTC can then be operating normally. Users can use "hwclock" for free.

```
/lib/modules # date -s 2010.12.06-11:40
Mon Dec  6 11:40:00 UTC 2010
/lib/modules # hwclock --help
BusyBox v1.13.4 (2010-12-06 10:32:54 CST) multi-call binary

Usage: hwclock [-r|--show] [-s|--hctosys] [-w|--systohc] [-l|--localtime]
[-u|--utc] [-f FILE]

Query and set hardware clock (RTC)

Options:
        -r      Show hardware clock time
        -s      Set system time from hardware clock
        -w      Set hardware clock to system time
        -u      Hardware clock is in UTC
        -l      Hardware clock is in local time
        -f FILE Use specified device (e.g. /dev/rtc2)

/lib/modules # hwclock -w
/lib/modules # hwclock
Thu Nov  31 11:40:43 2013  0.000000 seconds
/lib/modules #
```

Figure 2-5. Usage of hwclock

Chapter 3

RTC Related Files

By using the RTC related files, users may understand the underneath operations. All paths are related to arm-linux-3.3/.

This article goes with the Linux kernel. It describes RTC in details in Linux.

- Linux-3.3-fa/Documentation/rtc.txt

This implements the GM8136 RTC provided by Grain Media.

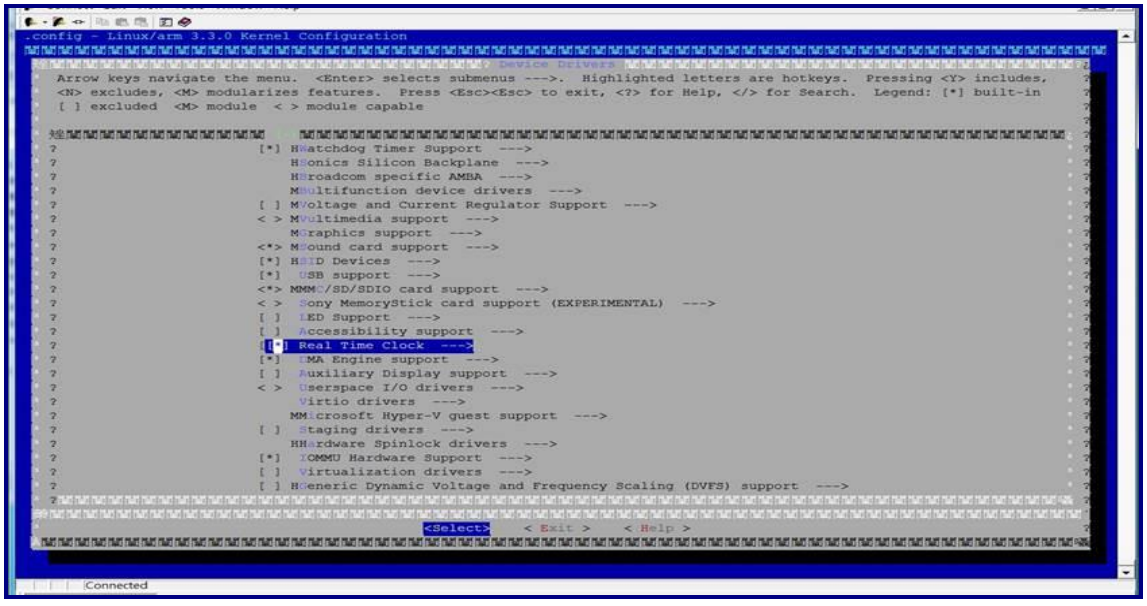
- module/RTC/FTRTC011/ftrtc011.c

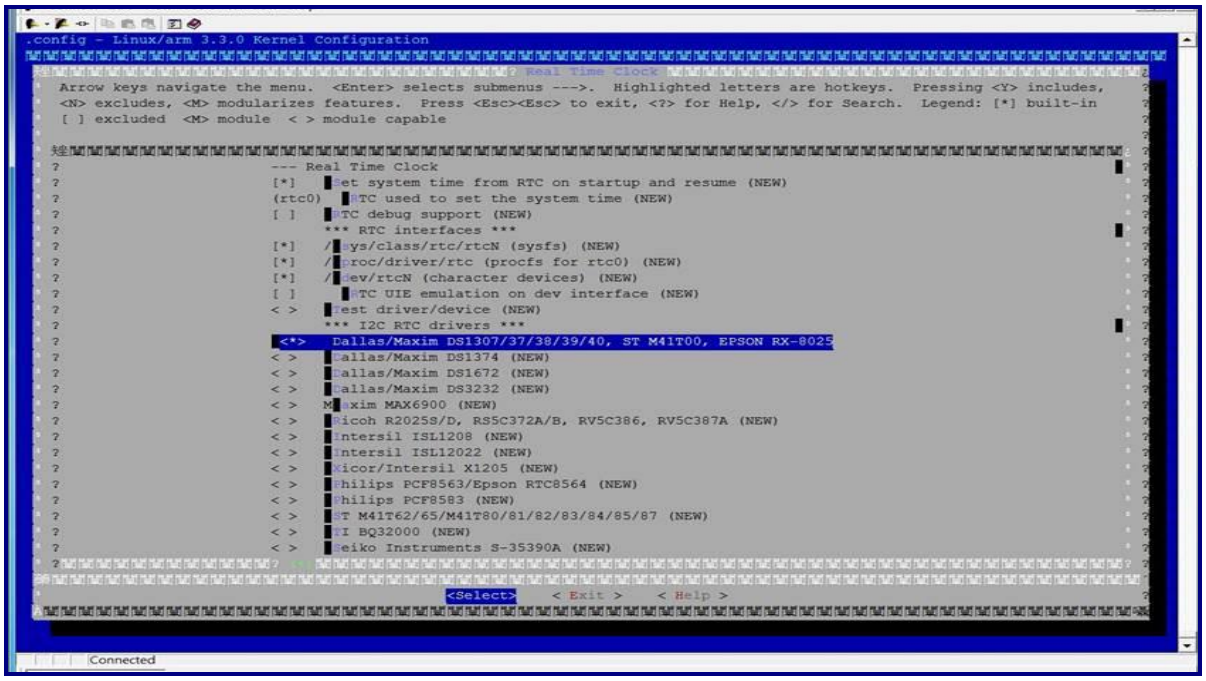
Chapter 4

External RTC Driver

Linux kernel 3.3 has built-in various RTC device drivers. Users can open the RTC driver device through the kernel configuration. Below is an example of enabling the DS1307 RTC support in kernel and user space.

Open Kernel Configuration to support DS1307 RTC and RTC subsystem





New DS1307 RTC device with I^2C address 0x68 in user space

```
#> echo ds1307 0x68 > /sys/bus/i2c/devices/i2c-0/new_device
#> mdev -s
```

Setup the system time

```
#> date "2014-08-05 10:46:20"
```

Setup the RTC time from the system time

```
#>  hwclock -w
```

Setup the system from the RTC time

```
#>  hwclock -s
```

Display the RTC time

```
#> hwclock -r
#> cat /proc/driver/rtc
rtc_time        : 10:49:15
rtc_date        : 2014-08-05
alrm_time       : 00:00:00
alrm_date       : 1970-01-01
alarm_IRQ       : no
alrm_pending    : no
update IRQ enabled      : no
periodic IRQ enabled    : no
periodic IRQ frequency  : 1
max user IRQ frequency  : 64
```