GE863-PRO^3 GPS Reference Design User Guide

With Atheros AR1511 GPS Module

1vv0300812 Rev.0 - 21/04/09

Making machines talk.

Disclaimer

The information contained in this document is the proprietary information of Telit Communications S.p.A. and its affiliates (“TELIT”).

The contents are confidential and any disclosure to persons other than the officers, employees, agents or subcontractors of the owner or licensee of this document, without the prior written consent of Telit, is strictly prohibited.

Telit makes every effort to ensure the quality of the information it makes available. Notwithstanding the foregoing, Telit does not make any warranty as to the information contained herein, and does not accept any liability for any injury, loss or damage of any kind incurred by use of or reliance upon the information.

Telit disclaims any and all responsibility for the application of the devices characterized in this document, and notes that the application of the device must comply with the safety standards of the applicable country, and where applicable, with the relevant wiring rules.

Telit reserves the right to make modifications, additions and deletions to this document due to typographical errors, inaccurate information, or improvements to programs and/or equipment at any time and without notice.

Such changes will, nevertheless be incorporated into new editions of this document.

Applicable Products

GE863-PRO3	GE863PR3***-***

The suffix "***-***" depends on the module HW/SW configuration. Please contact your Telit representative for details

Contents

1 Overview

1.1 Scope

The purpose of this document is to suggest general hardware and software guidelines and instructions to integrate Atheros AR1511 GPS module with Telit GE863-PRO^3.

The topics covered in this document are the following:

- Hardware integration
- Software integration
- Antenna selection and positioning guidelines

1.2 Audience

This User Guide is intended for hardware and software developers who develop custom applications based on Telit GE863-PRO³ module.

1.3 Contact Information, Support

Our aim is to make this guide as helpful as possible. Keep us informed of your comments and suggestions for improvements.

For general contact, technical support, report documentation errors and to order manuals, contact Telit's Technical Support Center at:

TS-EMEA@telit.com or http://www.telit.com/en/products/technical-support-center/contact.php

Telit appreciates feedback from the users of our information.

1.4 Open Source Licenses

Linux system is made up of many Open Source device drivers licensed as follows:

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
675 Mass Ave, Cambridge, MA 02139, USA

Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Please refer to the following web page for the full text of the license:

http://www.gnu.org/licenses/gpl-2.0.html

1.5 Related Documents

The following documents are related to this user guide:

[1] Telit GE863-PRO³ Hardware User Guide 1vv0300773a
[2] Telit GE863-PRO^3 EVK User Guide 1VV0300776
[3] Telit GE863-PRO^3 Linux SW User Guide 1vv0300781
[4] Telit GE863-PRO^3 GPS Reference Design – EVK User Guide 1vv0300809
[5] Telit GE863-PRO^3 Development Environment Linux User Guide 1vv0300780

All documentation can be downloaded from Telit's official web site www.telit.com if not otherwise indicated.

1.6 Document History

Revision	Date	Changes
ISSUE #0	21/04/09	First Release

2 Hardware integration

The Atheros AR1511 module is a GPS single chip receiver suitable for both host based and stand alone solutions (with the insertion of an additional flash memory). In this reference design a host based solution has been chosen in order to exploit the computational capacity of the GE863-PRO3 on board ARM9 and obtaining a cost effective GSM-GPRS + GPS solution.

The AR1511 hosted solution requires two supply voltages (2.7 ÷ 3.3V, typ. 3.0V, and 1.7 ÷ 1.9V, typ. 1.8V), a TCXO, a 32.768 kHz crystal for RTC, a SAW filter, a 2-wire UART connection to GE863-PRO3 and an active low reset signal driven by one GE863-PRO3 GPIO.

Moreover, this solution has been designed to work with an appropriate active antenna for better sensitivity (please refer to chapter §2.3.1 for antenna specification). Voltage of 3.0V is supplied to the active antenna through the antenna connector and ESD protection on the antenna line is guaranteed by an ESD suppression diode.

The required SAW filter is a 50:100 ohm unbalanced-balanced filter, while the oscillator is a typical TCXO for GPS applications with temperature stability ±0.5ppm, frequency tolerance ±2.0ppm and frequency 16.369MHz (different frequencies can be used simply changing configuration pins value, but this is the value chosen for this reference design).

In order to obtain the same power supply input range of the GE863-PRO3 (3.4 ÷ 4.2V, typ. 3.8V) two voltage regulators are required. Considering the low power consumption, if there aren't power consumption and power dissipation constraints, LDO voltage regulators can be used placing the 1.8V LDO regulator in series with the 3.0V LDO regulator (in order to reduce the power dissipated in the 1.8V regulator) as done in this reference design. Alternatively it is possible to use one dual output voltage regulator (in this case the 1.8V will be generated from the 3.4 ÷ 4.2V input).

In both cases, particular attention must be paid to the power dissipation issues of the design. Please note that the Atheros VDD_REG power input of the internal regulator can be connected to an external back-up battery for improved performances and power saving, if no backup battery is present, it must be equal or greater than the higher power supply of the AR1511 chip (please refer to Atheros datasheet for more detailed information).

The figure shows the AR1511 module with red circles on the main connection signals to the GE863-PRO^3 (UART, Reset and WAKE1 signal short circuited to the RX_UART signal).

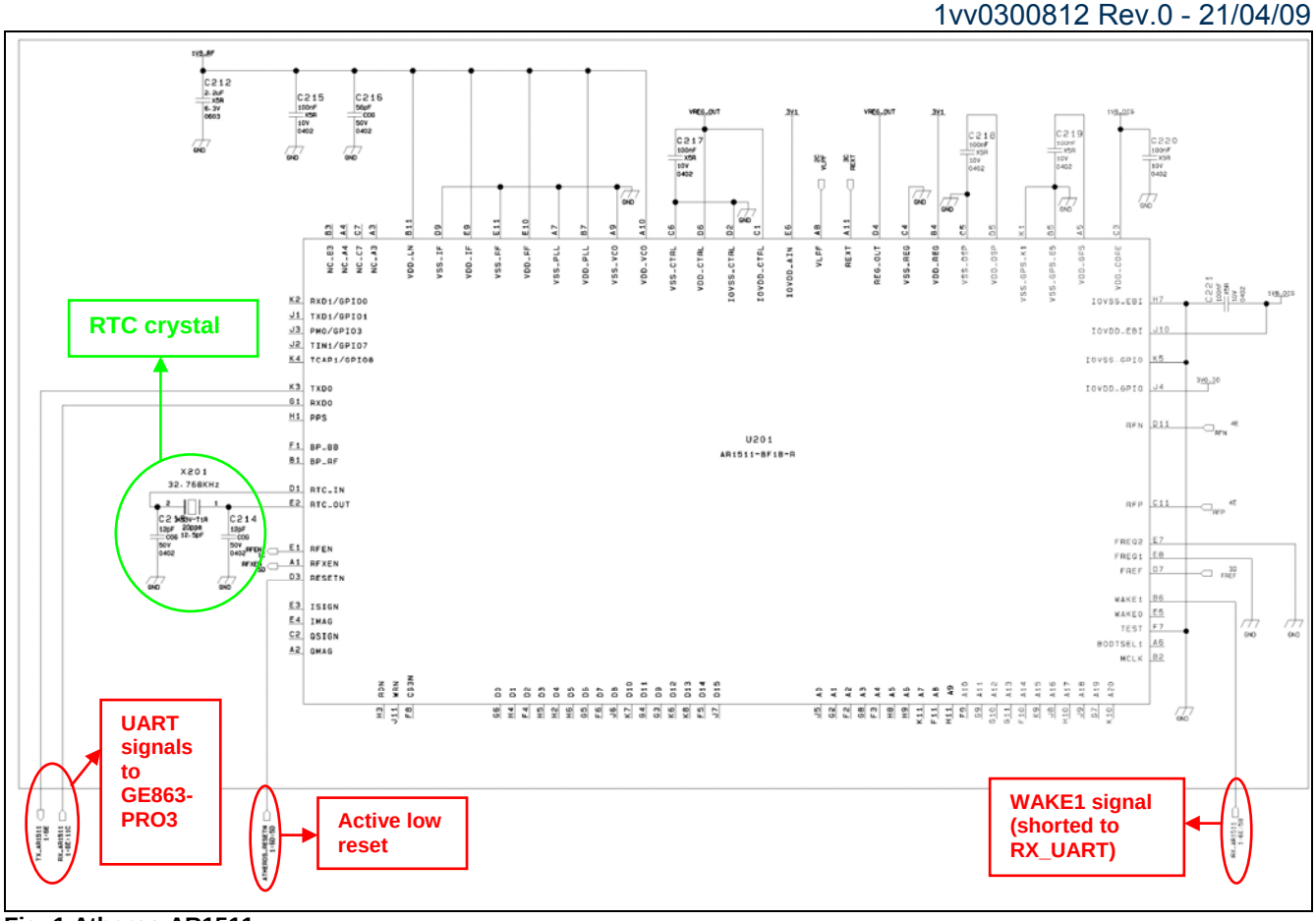


Fig. 1 Atheros AR1511

2.1 Power consumption improvements

For additional consumption reduction in some power saving modes an additional, optional, control circuit (see figure below) can be implemented. This circuit, controlled by the RFEN signal generated internally by the AR1511, allows avoiding feeding the TCXO and the active antenna when the module doesn't require it, by simply interrupting the 3V0_RF supply.

The following figure shows the remaining circuitry for the Atheros AR1511 host based solution (except for the power supply).

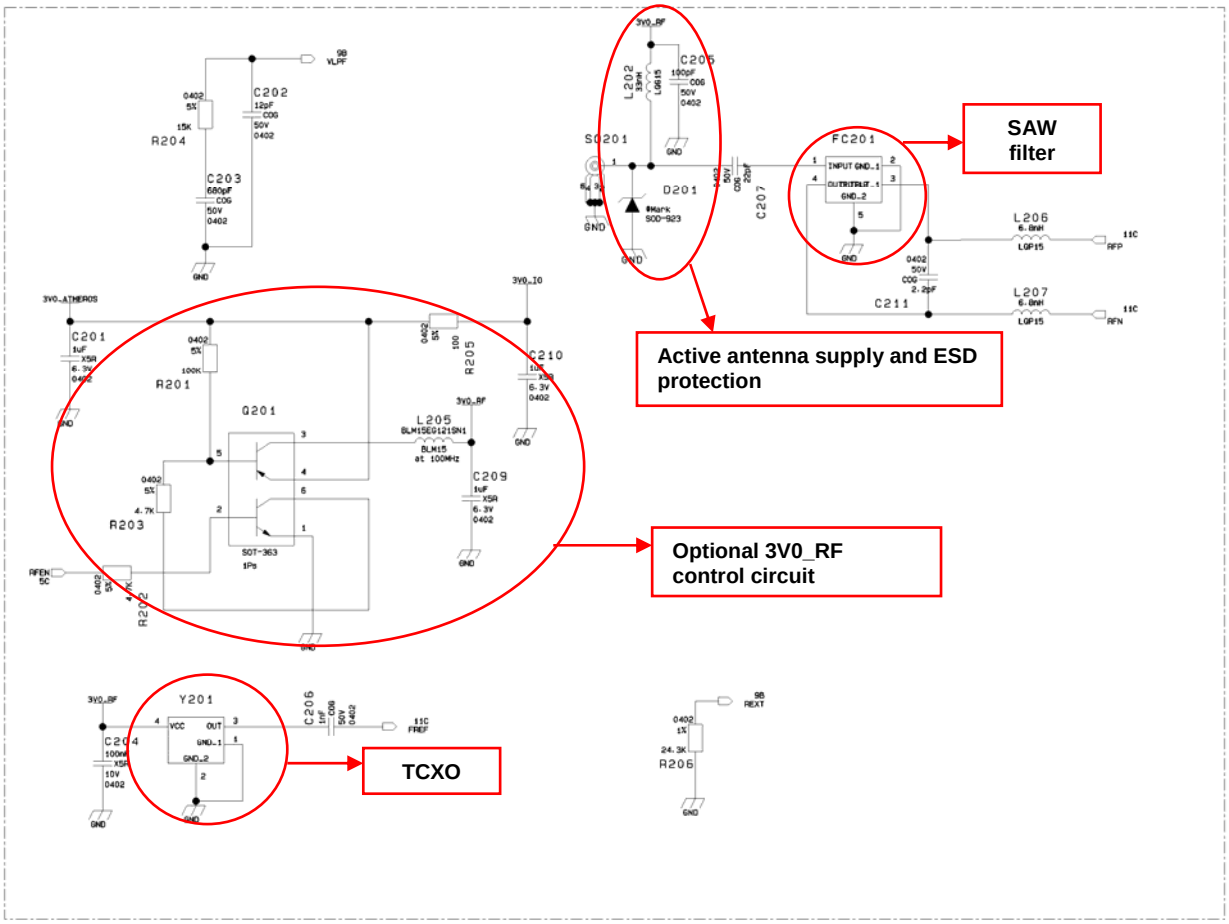


Fig. 2 AR1511 additional circuitry

In the previous schemes both the 3.0V and 1.8V power supplies are divided by passive filters (ferrite bead plus capacitor) into a voltage dedicated to the RF part (3V0_RF and 1V8_RF) and another one dedicated to the baseband or to the I/O (3V0_IO and 1V8_dig) in order to filter noise sharing between RF and baseband.

2.2 RF performance improvements

In order to further improve the RF performances of the module it is possible to reduce the noise in the sensitive 1V8_RF supply rail generating 1V8_RF (supplying the RF AFE of the AR1511) and 1V8_dig (supplying the baseband of the AR1511) from two different regulators (e.g. from the separate outputs of a dual LDO).
In order to easily implement this solution without the necessity to add a regulator to the previously described configuration, it is recommended to use a TCXO with supply voltage 1.8V fed by 1V8_RF, then to bias the antenna with the 3.4 ÷ 4.2V (3.8V typ.) input voltage of the system and to obtain the 3V1 reported in the schematics from the 3.1V output of the GE863-PRO^3 (ball P11).

In this configuration the TCXO can be turned off simply disabling the 1V8_RF output of the LDO using either the RFXEN (if the enable is active low) or RFEN signal (if the enable is active high) of the AR1511 (in order to reduce leakage currents during the switch down the configuration with an active low enable is recommended).
Atheros AR1511 is designed to work in conjunction with a suitable active GPS antenna. If the application relies on a passive antenna, a LNA is required to meet GPS spec sensitivity level.

2.3 RF power consumption optimization

In order to avoid undesired power consumption when not using the RF, the antenna supply can be controlled as well. The following figure reports an implementation of this solution with all the 3V1 coming from the GE863-PRO^3 (only external circuitry changes).

PLEASE NOTE: unlike the previous configuration, this one has not been tested.

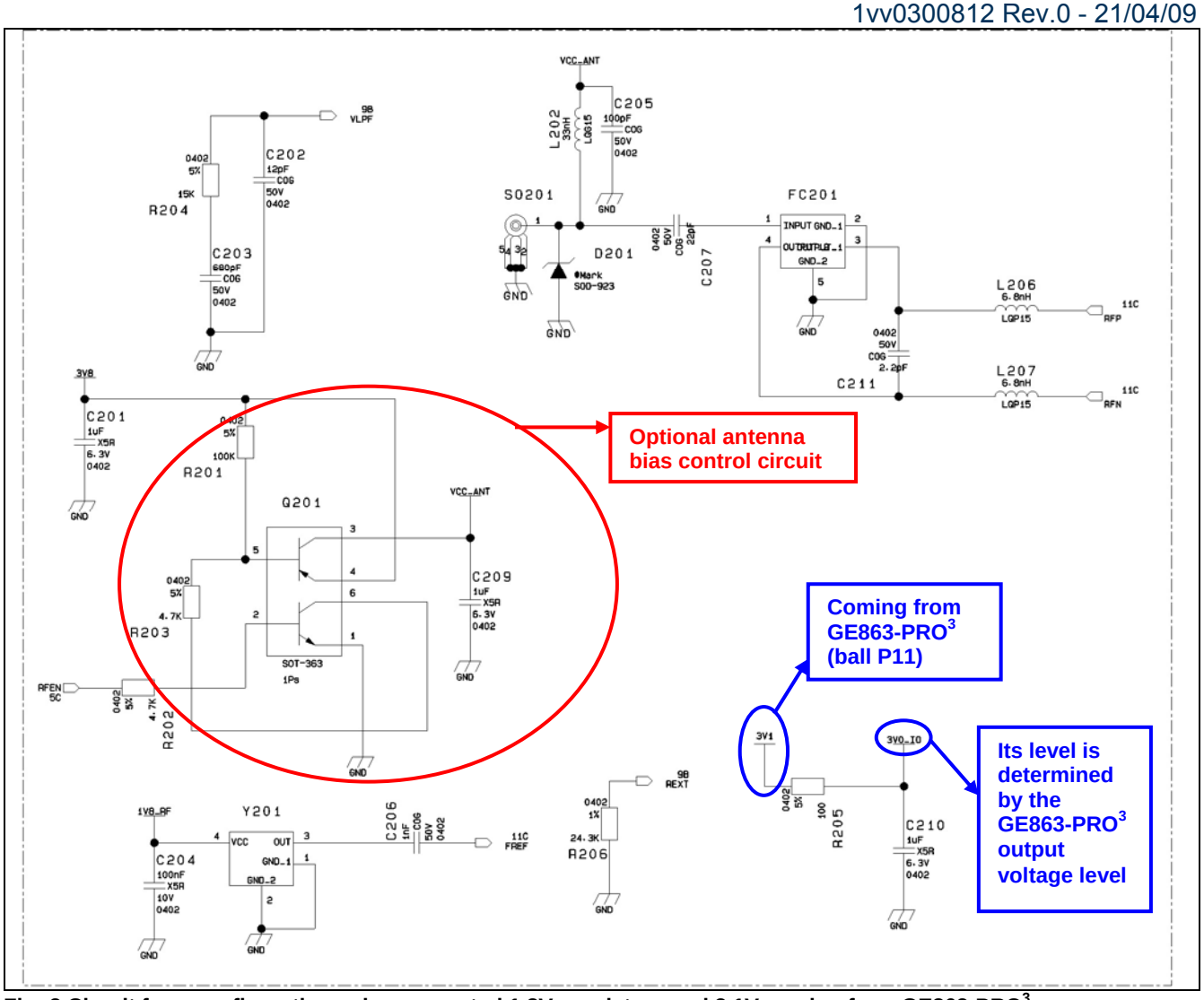


Fig. 3 Circuit for a configuration using separated 1.8V regulators and 3.1V coming from GE863-PRO^3

2.3.1 Antenna requirements

The reference design is supposed to work with an appropriate active antenna in order to obtain higher sensitivity and performances.

The recommended specifications for an active antenna working with the Atheros module are (excluding allowed tolerances):

- **Center frequency:** 1575.42MHz
- **Bandwidth:** 10MHz
- **Gain:** ≤ 28dB
- **NF:** ≤ 1.5dB
- **Power supply:** 3.0V

3 Software integration

The Atheros AR1511 is a "hosted" module: therefore a control application is needed to initialize/control the hosted module and to generate the NMEA protocol data stream.
Hosted Orion is the Atheros control application that manages the AR1511 module.
Since the Atheros Hosted Orion requires about 5MB of Ram and 7.5MB of Flash to run, one of the following configurations of GE863-PRO³ with external Flash is needed:

A. GE863-PRO³ 64MB Ram/4MB Flash + GE863-PRO^3 Flash Memory Extension
B. USB pen drive with at least 8 MB of free space
C. SD Card with at least 8 MB of free space

Before you begin, please ensure to have an up-to-date system.
For further information on how to install the GE863-PRO^3 Flash Memory Extension board for **EVK-PRO^3** please refer to [4].
Latest software releases are available on the Telit's official web site Download Zone http://www.telit.com, in *Software Tools_GSM/GPRS* » *GE863-PRO³_with_Linux.*

3.1 Package contents

The package contains the Atheros HostedOrion control application and GCC library files along with the Atheros firmware images, which are contained in the "upload/" directory.
Moreover, Telit provides a simple startup script named "startOrion", located in the package root directory, which already contains all the default parameters needed to start managing the GPS module.

3.2 System setup

Follow this procedure the first time you set up the **GE863-PRO^3** to work with Atheros AR1511 GPS module.

3.2.1 Startup process on GE863-PRO^3

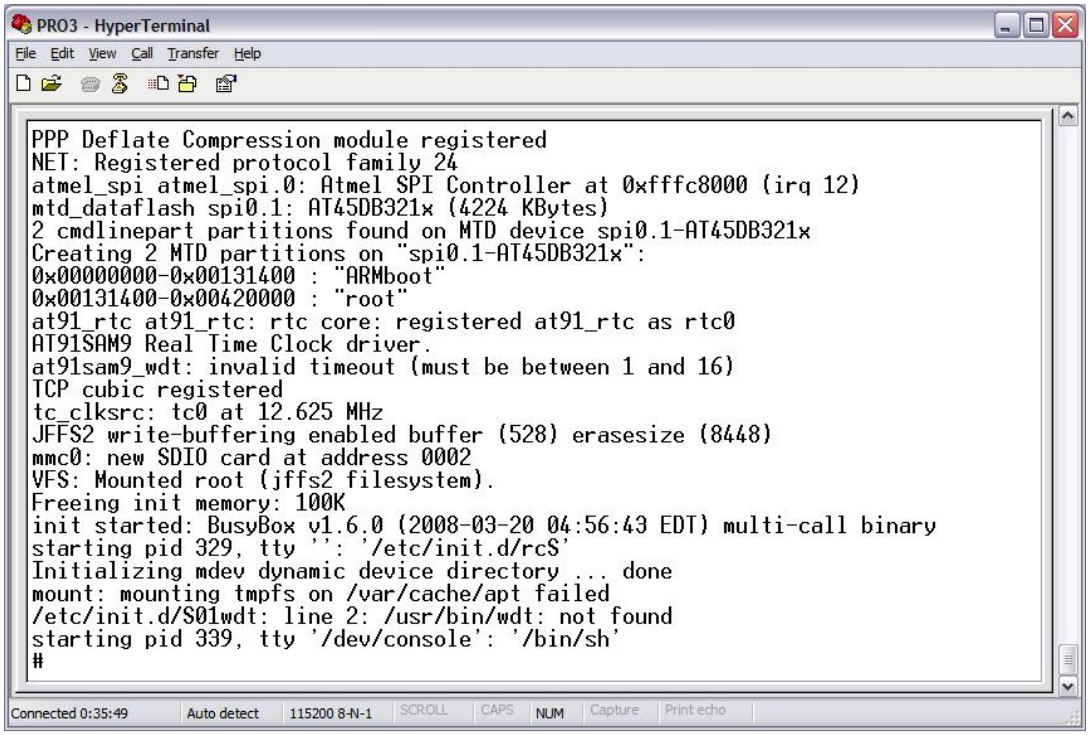
Connect the GE863-PRO³ to your host system via serial cable (if using the **EVK-PRO^3**, please use Debug port, for further details refer to [2]).

Open a terminal program (such as Hyperterminal) on your host system and use the following parameters for the connection:

Bits per second: 115200

Data bits: 8
Parity: None
Stop bits: 1
Flow Control: None

Turn the GE863-PRO³ on. Once the system startup has finished, the terminal will display the shell prompt as shown below.



Since two GPIOs are needed to turn AR1511 ON/OFF and/or to reset it, the startup values for these GPIOs must be set. Please see [1] for a complete list of available GPIOs.
Let us assume PA28 (at91sam9260_gpio.28) and PC28 (at91sam9260_gpio.92) GPIOs are used for Enable and Reset respectively. Since these two signals are active low, they must be high by default: create a start script which sets the correct GPIO levels, as shown below.

```
# vi /etc/init.d/S03
```

Press the “i” key and write the following lines:

```
echo 01 > /dev/at91sam9260_gpio.28
echo 01 > /dev/at91sam9260_gpio.92
```

now press the “Esc” and then “:” keys, and type:

`wq` followed by <return>.

Reboot the system. Now the Atheros AR1511 GPS module can be turned ON.

3.2.2 Hosted Orion setup

Before using the Atheros AR1511, the Atheros Hosted Orion must be downloaded onto the GE863-PRO³ filesystem.

Please remind that the external Flash shall be mounted before turning on the system. Ensure the external Flash is correctly formatted. Let us suppose the external Flash is mounted on `/mnt/ExternalFlash`.
Copy the Atheros Hosted Orion on the 8MB PRO³ Flash memory, in `/mnt/ExternalFlash`. There are several ways to download a file onto the target: please refer to [5] for further details.

If a SD Card or USB pen drive is used, you may first copy Atheros Hosted Orion into the Flash device, then insert it into the proper connector available on your system: see [2] for information on how to connect an external flash device on the **EVK-PRO³**.
Please refer to [3] for information on how to mount a SD Card or a USB pen drive respectively.

Move into the device folder containing the GPS SW Package, e.g:

```
# cd /mnt/ExternalFlash/HostedOrion
```

Setup the libraries needed to run Atheros Hosted Orion:

```
# ln -s /mnt/ExternalFlash/HostedOrion/libgcc_s.so.1 /lib/libgcc_s.so.1

# ln -s /mnt/ExternalFlash/HostedOrion/libstdc\+\+.so.6 /lib/libstdc++.so.6
```

The system is now ready to run the Atheros Hosted Orion for the Atheros AR1511.

3.3 AR1511 ON/OFF and Reset

To turn the Atheros AR1511 ON/OFF and/or to reset it, it is necessary to perform simple GPIO writes on the pins used to control AR1511. Please refer to [3] for further information on how to use GPIO under Linux.
According to the current design, we are assuming that PA28 (at91sam9260_gpio.28) and PC28 (at91sam9260_gpio.92) GPIOs are used for Enable and Reset respectively.
In this case, to:

- **turn ON** AR1511, set at91sam9260_gpio.28 to “0”
- **turn OFF** AR1511, set at91sam9260_gpio.28 to “1”
- **reset** AR1511, set at91sam9260_gpio.92 to “0” for at least one second, then to “1” again

Below is an example C code fragment to control ON/OFF and Reset.

```
/* Open Enable GPIO */
if((fdA1 = open(“at91sam9260_gpio.28”, O_RDWR)) < 0)
{
```


```
        printf("\nError while opening gpio device.\n");
        return -1;
}

/* Open Reset GPIO */
if((fdA2 = open(“at91sam9260_gpio.92”, O_RDWR)) < 0)
{
        printf("\nError while opening gpio device.\n");
        return -1;
}

/* Turn the module ON */
if(write(fdA1, “O0”, 2)) < 0)
{
        /* error management routine... */
}


/* Reset the module*/
if(write(fdA2, “O0”, 2)) < 0)
{
        /* error management routine... */
}

/* wait at least 1 second of pull down */
usleep(1000000);

if(write(fdA2, “O1”, 2)) < 0)
{
        /* error management routine... */
}


/* Turn the module OFF */
if(write(fdA1, “O1”, 2)) < 0)
{
        /* error management routine... */
}
```

3.4 Atheros Hosted Orion functionalities

Atheros Hosted Orion is used to getting Position/ Velocity/Time (PVT), maintaining Ephemeris and Almanac, processing aiding data, and estimating code phases and Doppler. It also encodes and decodes NMEA messages and handles the transfer of NMEA and aiding data over physical and virtual serial ports.

At startup, Hosted Orion waits for a reset on AR1511, and then it uploads the firmware (usually located in the "./upload" directory) onto the AR1511 module through the physical serial port used to connect it with GE863-PRO^3.
A pseudo terminal device is also created to receive and send NMEA protocol sentences.

3.4.1 Running Hosted Orion

First move to the folder containing the Atheros Hosted Orion e.g.:

```
# cd /mnt/ExternalFlash/HostedOrion
```

The Atheros Hosted Orion can be run by typing:

```
# ./Orion_elf -limage OrionUNAV1.img -lport /dev/ttyS4 -u0 /dev/ptyp0 -u1
/dev/ttyS4 -fsh ./fsh
```

and resetting AR1511 through PC28 GPIO (at91sam9260_gpio.92 device),

OR simply running the `startOrion` script (which contains the command line above and performs AR1511 reset):

```
# ./startOrion
```

Once Hosted Orion has been successfully launched, the NMEA protocol data stream can be retrieved from /dev/ttyp0, which is the slave pseudo terminal device corresponding to the one specified with the `-u0` command line option above.

All Orion_elf command line options are shown below; the default values are within parenthesis.

Synopsis:

Orion_elf [-u0 <Port name>] [-u1 <Port name>] [-fsh <dir/filename>] [-ldir <dir>]
[-lport <Port name>] [-lbtlod <filename>] [-limage <filename>] [-setsystime]

Parameters:

-u0 <Port name> — UART0 port (NMEA port) (using /dev/ttyS0)
It specifies the serial device to be created to send/receive NMEA protocol data stream. Please note that when a pseudo terminal device is used, e.g. –u0 /dev/ptyp0, the port to be accessed to send/receive NMEA protocol data stream is /dev/ttyp0.

-u1 <Port name> — UART1 port (UBP port) (using /dev/ttyS1)
It specifies the serial device used to physically connect the Atheros AR1511 module.

-fsh <dir/filename> — FSH location (using ./fsh)
It specifies the directory used by Hosted Orion to save its configuration and collected information.

-ldir <dir>	Upload directory (using ./upload) It specifies the directory containing all the files to be uploaded onto AR1511.
-lport <Port name>	Upload port name (using /dev/ttyS0) It specifies the serial device to be used for uploading.
-lbtlod <filename>	Boot loader lod filename (using AR1511_bootloader.lod) It specifies the filename for the AR1511 Boot Loader to be used.
-limage <filename>	Orion image filename (using Orion.img) It specifies the filename for the AR1511 Firmware to be used.
-setsystime	Allow system time update using GPS

3.5 NMEA Sending/Receiving Example

Sending and receiving the NMEA protocol data stream is as simple as writing to and reading from a serial device. Please refer to [3] for a complete description on how to perform writing/reading on a serial device.
The C source code below shows an example on how to open a serial device, perform a read and write on a buffer containing a NMEA sentence to be sent.

....

```
int fd; // file descriptor for /dev/ttyp0 device
int len; // number of bytes read from /dev/ttyp0
unsigned char inbuf[1024];
unsigned char outbuf[ ] = ”$PUNV,START,HOT*0E\r\n“; // Atheros proprietary NMEA
                                                    // sentence to be sent

memset(inbuf, 0, sizeof(inbuf));  // initialize inbuf to 0


/* Open the serial device to be read/written */

if((fd = open(“/dev/ttyp0”, O_RDWR | O_NOCTTY | O_NDELAY)) < 0)
{
          /* ERROR MANAGEMENT ROUTINE */
} else {
          /* SERIAL PORT OPENED */
}

/* Read from /dev/ttyp0 */
len = read(fd, inbuf, sizeof(inbuf)); // read (receive) up to 1024 bytes from
                                      // /dev/ttyp0
/* Write to /dev/ttyp0 */
/*
Note for Atheros Hosted Orion users: current version of Orion may not operate
properly if the user sends more than one custom NMEA message within a second.
```


```
*/

if(write(fd, outbuf, strlen(outbuf) < 0);  // write (send) the NMEA sentence to
{                                          // /dev/ttyp0
       /* error management routine... */
}

close(fd);

....
```