QUICK START GUIDE

AMOS-820

Linux EVK v4.1.3

1.00-03132018-151900

Revision History

Version	Date	Remarks
1.00	03/13/2018	Initial release

Table of Contents

1. Introduction

The Quick Start Guide provides an overview of how to boot the Android EVK system image on the AMOS-820 system (Bare board: VAB-820 with NXP i.MX 6Quad Cortex-A9 processor) and configure the supported hardware functions in the build.

The AMOS-820 Linux EVK v4.1.3 is developed based on the NXP fsl-yocto-L4.1.15_1.1.0-ga (Yocto 2.0 Jethro) and enables the hardware features of the AMOS-820 system.

1.1 EVK Package Content

There are two folders in the package listed as below.

Firmware folder	Description
VAB-820_Yocto2.0_BIN_v4.1.3.tar.gz	Yocto EVK system image and installation script files
Document folder	**Description**
AMOS-820_Linux_EVK_v4.1.3_Quick_Start_Guide _v1.00_20180313.pdf	Quick Start Guide
Tools folder	**Description**
BT_Config.zip	Bluetooth A2DP configuring file

AMOS-820 Linux BSP content

1.1.1 Firmware Folder Contents

VAB-820_Yocto2.0_BIN_v4.1.3.tar.gz: contains installation script files and the precompiled U-boot and image for evaluating the AMOS-820 system.

1.1.2 Document Folder Contents

AMOS-820_Linux_EVK_v4.1.3_Quick_Start_Guide _v1.00_20180313.pdf: The Quick Start Guide provides an overview of how to boot the Linux EVK system image on the AMOS-820 system (Bare board: VAB-820 with NXP i.MX 6Quad Cortex-A9 processor) and configure the supported hardware functions in the build.

1.1.3 Tools Folder Contents

BT_Config.zip: contains the Bluetooth A2DP configuring file.

1.2 Version Information and Supported Features

- U-Boot version: 2015.04
- Kernel version: 4.1.15
- Evaluation image: OpenEmbedded-core built with Yocto 2.0 Jethro
- Development based on NXP fsl-yocto-L4.1.15_1.1.0-ga (Yocto 2.0 Jethro)
- Supports SPI with eMMC or Micro SD boot (default)
- Supports HDMI display
- Supports HDMI audio output
- Supports CVBS
- Supports COM1 DTE mode, COM2 as debug port
- Supports 2 FlexCAN TX/RX
- Supports Gigabit Ethernet
- Supports Line-in, Line-out, and Mic-in
- Supports VNT9271 USB Wi-Fi dongle
- Supports EMIO-1541 miniPCIe Wi-Fi module
- Supports EMIO-2531 miniPCIe Wi-Fi & Bluetooth module
    - Supports Bluetooth A2DP and SPP profile
- Supports EMIO-2550 miniPCIe Mobile Broadband module
- Supports Watchdog Timer, GPIO and RTC

2. Image Installation

This section explains the setup requirements for booting from a Micro SD card or the SPI ROM and eMMC.

The installation script files, the precompiled U-boot, and the image are provided in the “Firmware” folder.

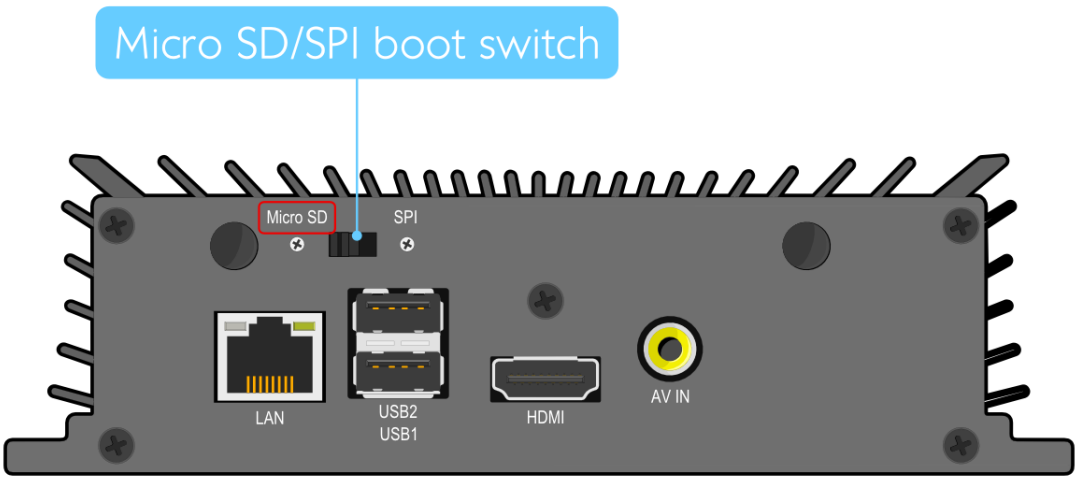
2.1 Booting from a Micro SD Card

The first step is to install this image onto a Micro SD card. Insert a Micro SD card into your Linux host machine and make sure it is not mounted. Install the SD card image onto the Micro SD card with the commands below, replacing <device name> with the correct value for the card.

***Important**: Make sure you are writing to the correct device or the host system environment could be damaged.

```
$ tar xvf VAB820_Yocto2.0_BIN_v4.1.3.tar.gz
$ cd sd_installer/
$ sudo ./mk_sd_installer.sh /dev/<device name> --yocto
```

Next, on the AMOS-820, set the boot switch to the Micro SD position as shown below.



Micro SD / SPI boot switch diagram

Insert the prepared Micro SD card into the AMOS-820, connect an HDMI display, and power on the AMOS-820 to initiate the boot process.

When the boot process is completed, you will see the OpenEmbedded desktop.

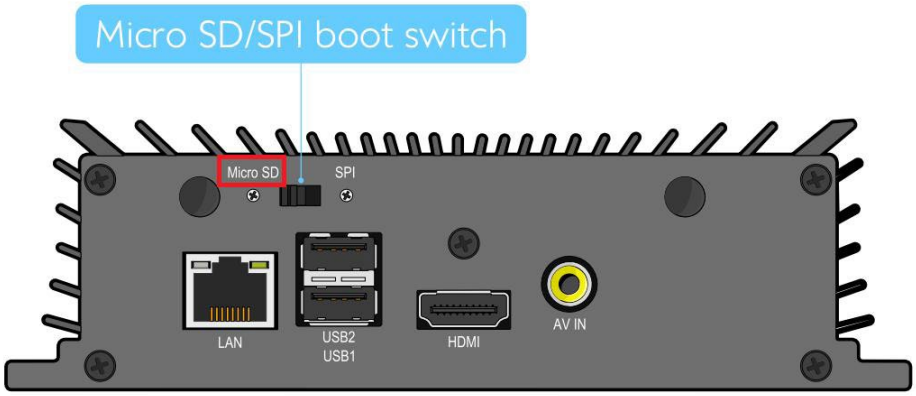
2.2 Booting from the SPI ROM with eMMC

The AMOS-820 supports booting from the SPI ROM while loading the kernel and root filesystem from the eMMC.

The first step is to prepare a Micro SD card as the system installer. Insert a Micro SD card into your Linux host machine and make sure it is not mounted. Install the system installer onto the Micro SD card with the command below, replacing <device name> with the correct value for the card.

```
$ tar xvf VAB820_Yocto2.0_BIN_v4.1.3.tar.gz
$ cd sd_installer/
$ sudo ./mk_sd_installer.sh /dev/<device name>
```

Next, on the AMOS-820, set the boot switch to the Micro SD position as shown below.

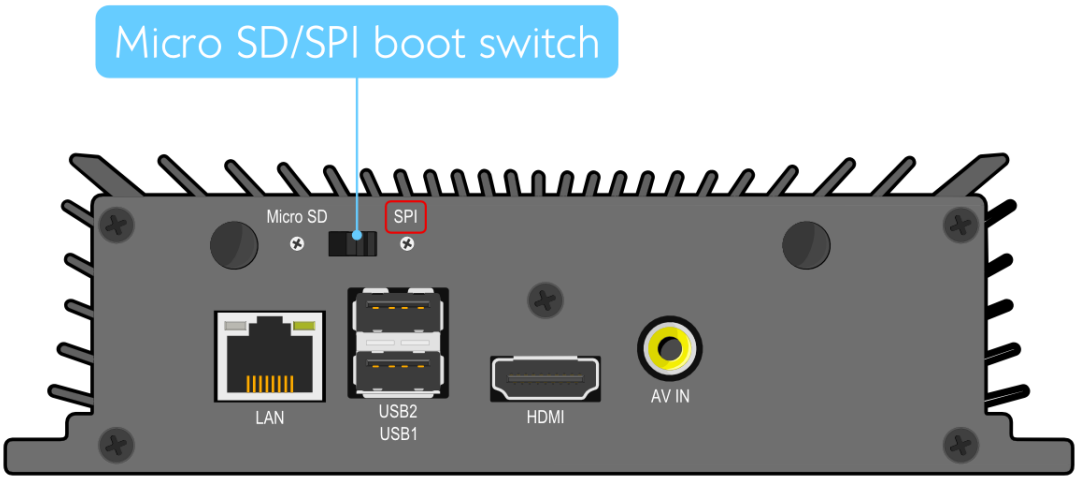


Micro SD / SPI boot switch diagram

Insert the prepared Micro SD card into the AMOS-820 system then power on. The system will automatically start to install the U-boot and image into the SPI ROM and eMMC.

When the install process is completed, unplug the power cable and remove the Micro SD card.

Next, in order to boot from the SPI ROM, make sure the boot switch is set to SPI ROM boot.



Micro SD / SPI boot switch diagram

Next, connect an HDMI display, and power on the device to initiate the boot process. When the boot process is completed, you will see the OpenEmbedded desktop.

3. Hardware Functions

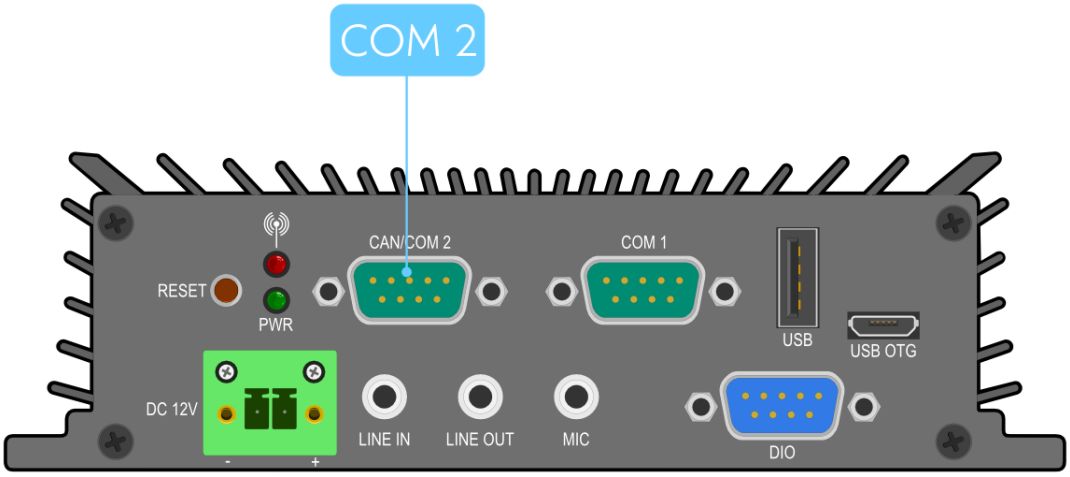
This section explains how to enable and test the hardware functions precompiled in the AMOS-820 Linux EVK including setting the U-Boot parameters, setting up the Video-in, configuring FlexCAN, configuring Watchdog Timer, setting up GPIO, and configuring RTC.

3.1 Setting Up U-Boot Parameters

When setting up the U-Boot parameters, the first step is to connect the host machine and the AMOS-820 through the COM 2 port. Use a serial port communication program such as PuTTY, GtkTerm, or Minicom to configure the serial port setting and connect to the debug console. There you will be able to see the U-Boot boot log and adjust settings in the U-Boot console..

```
+-----------------------------------------------------------+
| A   -        Serial Device              : /dev/rfcomm0    |
| B   -      Lockfile Location            : /var/lock       |
| C   -        Callin Program             :                 |
| D   -       Callout Program             :                 |
| E   -         Bps/Par/Bits              : 115200 8N1      |
| F   -      Hardware Flow Control        : No              |
| G   -      Software Flow Control        : No              |
+-----------------------------------------------------------+
```

Serial port setting of host machine



COM 2 port diagram

Next, power on the AMOS-820 to initiate the boot process. When prompted, press any key to stop the boot process and enter the U-Boot console as illustrated by the screen below.

```
U-Boot 2015.04-imx_v2015.04_4.1.15_1.0.0_ga+gd7d7c43 (Feb
20 2017 - 14:19:23)

CPU:   Freescale i.MX6Q rev1.5 at 996 MHz
CPU:   Temperature 30 C
Reset cause: POR
Board: MX6Q-VAB820
I2C:   ready
DRAM:  1 GiB
MMC:   FSL_SDHC: 0, FSL_SDHC: 1
SF: Detected W25Q32BV with page size 256 Bytes, erase size
4 KiB, total 4 MiB

No panel detected: default to Hannstar-XGA
Display: Hannstar-XGA (1024x768)
In:    serial
Out:   serial
Err:   serial
Net:   FEC [PRIME]
Warning: failed to set MAC address

Normal Boot
Hit any key to stop autoboot:  0
=>
```

Debug console view of boot process

To list the current U-Boot parameters, use the following command:

```
=> printenv
```

To have more information make sure the printout message includes “bootcmd=run bootcmd_auto”.

The default “bootcmd” parameter is set to auto detect the system location by first attempting to load the kernel from the Micro SD card. If it is not detected it continues to boot from the eMMC.

To load the kernel from the specified device, refer to the two examples below.

To load the kernel from the Micro SD card, use the following command:

```
=> setenv bootcmd ‘run bootcmd_sd’
=> saveenv
=> boot
```

To load the kernel from the eMMC, use the following command:

```
=> setenv bootcmd ‘run bootcmd_mmc’
=> saveenv
=> boot
```

3.2 Restoring Default U-Boot Parameters

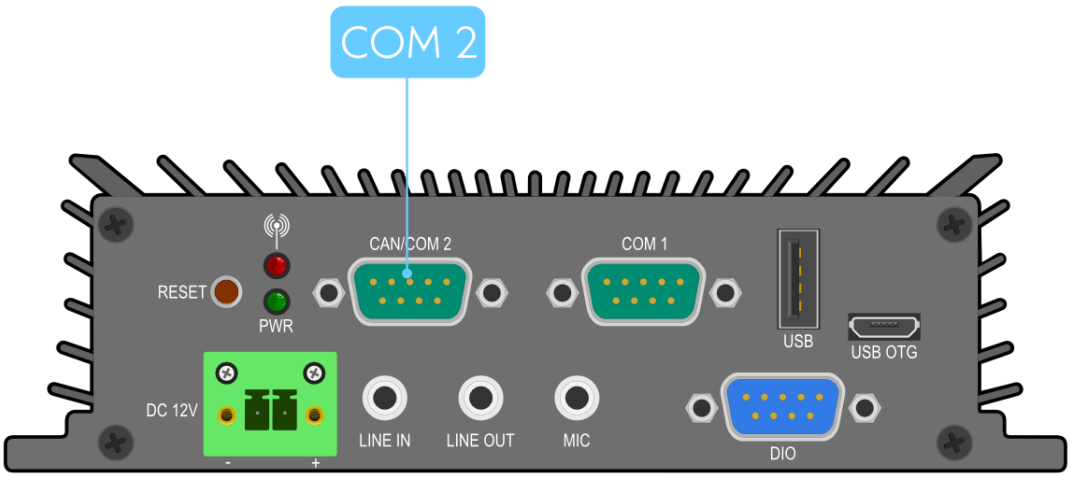
If the U-Boot parameters have been modified, the “destroyenv” command in the U-Boot console can restore the factory defaults settings.

To restart the device, use the “reset” command:

```
=> destroyenv
=> reset
```

3.3 Using the OpenEmbedded Console

The first step is to connect the host machine and the AMOS-820 through the COM 2 port. Use a serial port communication program such as PuTTY, GtkTerm, or Minicom to connect to the debug console. There you will be able to see the U-Boot boot log and adjust settings in the U-Boot console. Next, power on the AMOS-820 to initiate the boot process. When the boot process is completed you will be prompted to login.



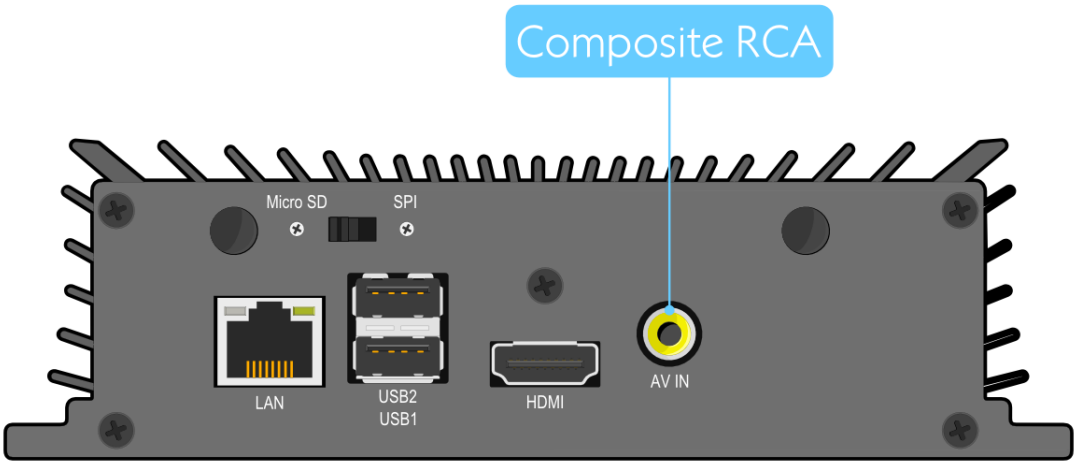
COM 2 port diagram

The default account is “root”, with no password set (just press Enter when prompted for the password).

```
...
Freescale i.MX Release Distro 4.1.15_1.1.1 imx6qvab820 /dev/ttymxc1
imx6qvab820 login: root
```

3.4 Setting Up the Video-in

The AMOS-820 supports CVBS signal input for Composite RCA jack. The following steps are for Video-in function verification.



Composite RCA jack diagram

3.4.1 Setting Up the CVBS

To enable the video-in, use the following command:

```
# gst-launch-0.10 imxv4l2src input=1 deinterlace=true! imxv4l2sink
```

If the source of the video-in is PAL, use the following command to display the video on the screen:

```
# gst-launch-0.10 imxv4l2src input=1 deinterlace=true ! imxv4l2sink crop-top-by-pixel=1
```

To display other video-in sources on the screen, use the following command:

```
# gst-launch-0.10 imxv4l2src input=1 deinterlace=true ! imxv4l2sink
```

3.4.1.1 CVBS Video-in Recording

If your application requires the ability to record the video-in feeds onto the device storage, the following commands can be used when writing your software application.

Record Feed as MP4:

To save the video-in feed as an MP4 file, use the following command:

```
# gst-launch-0.10 imxv4l2src input=1 num-buffers=<value> ! vpuenc codec=6 ! ffmux_mov !
filesink location=/<path/filename.mp4>
```

For example, to save the video-in video as a 10-second MP4 file into /home folder with a filename demo_1.mp4, use the following command:

```
# gst-launch-0.10 imxv4l2src input=1 num-buffers=300 ! vpuenc codec=6 ! ffmux_mov ! filesink
location=/home/demo_1.mp4
```

To play the saved video file, use the following command:

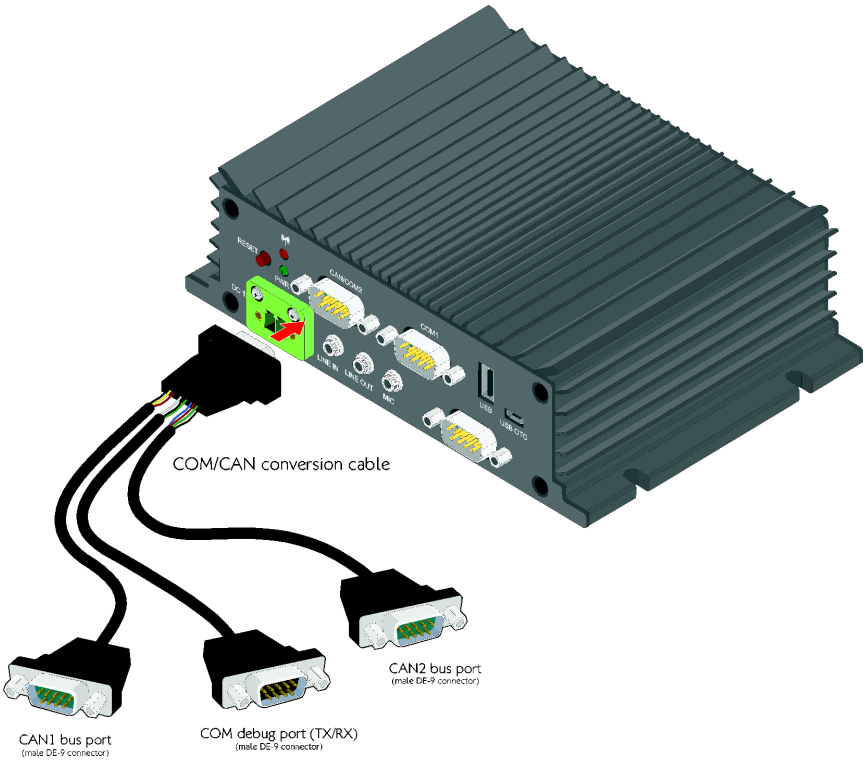
```
# gst-launch-0.10 playbin2 uri=file:/ <path/filename.mp4>
```

If you need to display the video-in source on the screen and record the feed at the same time, use the following command:

```
# gst-launch-0.10 imxv4l2src input=1 num-buffers=900 queue-size=30 ! tee name=t !
queue ! imxv4l2sink t. ! queue ! vpuenc codec=6 bitrate=3000000 ! matroskamux ! filesink
location=/<path/filename.mp4>
```

3.5 Configuring FlexCAN

The CAN/COM 2 port of the AMOS-820 supports one debug port (COM 2) and two CAN bus ports. The CAN bus supports CAN protocol specification Version 2.0 B while the COM 2 supports TX/RX for debugging purposes only. The first step is to connect the COM/CAN conversion cable.



Installing COM/CAN conversion cable diagram

Please note that you need to disable the CAN bus first, set the CAN bus bitrate. The CAN bus bitrate should be 5000 < [BRT_value] < 1000000.

CAN bus 1 is can0 and CAN bus 2 is can1.

To set the CAN bus 1 bitrate [BRT_value], use the following command:

```
# ip link set can0 type can bitrate [BRT_value]
```

To enable the CAN bus 1, use the following command:

```
# ifconfig can0 up
```

To disable the CAN bus 1, use the following command:

```
# ifconfig can0 down
```

Use the cansend command to send CAN bus 1 data:

```
# cansend can0 <ID>#<Data>
```

<ID>: Device ID of CAN bus

<Data>: Send out data

Use the candump command to receive data from CAN bus 1:

```
# candump can0 &
```

3.6 Configuring Watchdog Timer

Watchdog Timer is an electronic timer that is used to detect and recover from system malfunctions.

The **wdt_driver_test.out** provides an application to configure the Watchdog timer.

To enable Watchdog Timer with the default timeout (60 seconds), use the following command:

```
$ /unit_tests/wdt_driver_test.out enable wdt
```

To change the timeout <value> and the system reboot <value> (seconds), use the following command:

```
$ /unit_tests/wdt_driver_test.out set_timeout <value>
```

To send keep-alive requests to Watchdog Timer <value> (seconds), use the following command:

```
$ /unit_tests/ wdt_driver_test.out set_keep_alive <value>
```

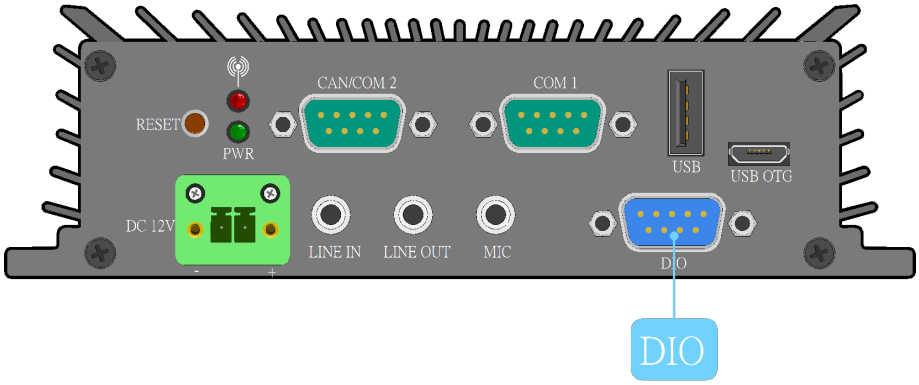
When the value of “set_keep_alive” is smaller than the value of “set_timeout”, the system will not reboot.

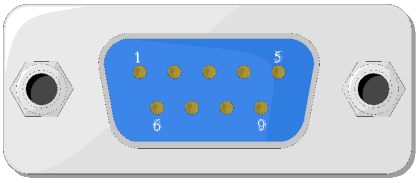
To disable Watchdog Timer, use the following command:

```
$ /unit_tests/wdt_driver_test.out disable wdt
```

3.7 Setting Up GPIO

The AMOS-820 has a DIO port which consists of 9 pins. The following section explains how to set up these pins for input/output communication





DIO port diagram

3.7.1 Setting Up DIO Port for GPIO Functionality

The echo value of GPIO pins is listed below:

Pin	Signal	GPIO echo value
1	GPIO_7	7
2	GPIO_1	1
3	GPIO_8	8
4	GPIO_2	2
5	GPIO_9	9
6	GPIO_4	4
7	GPIO_16	203
8	GPIO_5	5
9	GND	

GPIO port pinouts table

Set the multifunction pin as a GPIO pin. Here is the example for GPIO pin 1:

```
$ echo 7 > /sys/class/gpio/export
```

3.7.2 Setting Up GPI Mode

To set a pin to “input” mode (GPI), use the following command (here is the example for GPIO pin 1):

```
$ echo “in” > /sys/class/gpio/gpio7/direction
```

To read the GPI input value, use the following command:

```
$ cat /sys/class/gpio/gpio203/value
```

3.7.3 Setting Up GPO Mode

To set a pin to “output” mode (GPO), use the following command (here is the example for GPIO pin 1):

```
$ echo “out” > /sys/class/gpio/gpio7/direction
```

To output a “Low” value on the GPO pin, use the following command:

```
$ echo 0 > /sys/class/gpio/gpio7/value
```

To output a “High” value on the GPO pin, use the following command:

```
$ echo 1 > /sys/class/gpio/gpio7/value
```

3.8 Configuring RTC

The RTC (Real-Time Clock) keeps track of the current time values. The time values include the year, month, date, hours, minutes and seconds. To configure the RTC on the AMOS-820, set the system time with the Linux date command as in the example shown below.

```
$ date mmddHHMMYYYY
```

Time	Value
mm	2-digit month
dd	2-digit day
HH	2-digit hours in 24h system
MM	2-digit minutes
YYYY	4-digit year

Time value table

For example, the value 061110072013 means June 11 2013 10:07. The date command also accepts other time formats. For more information, use the following command:

```
$ date --help
```

To write the system time to the RTC, use the following command:

```
$ hwclock -w -f /dev/rtc0
```

To read the RTC time, use the following command:

```
$ hwclock -r
```

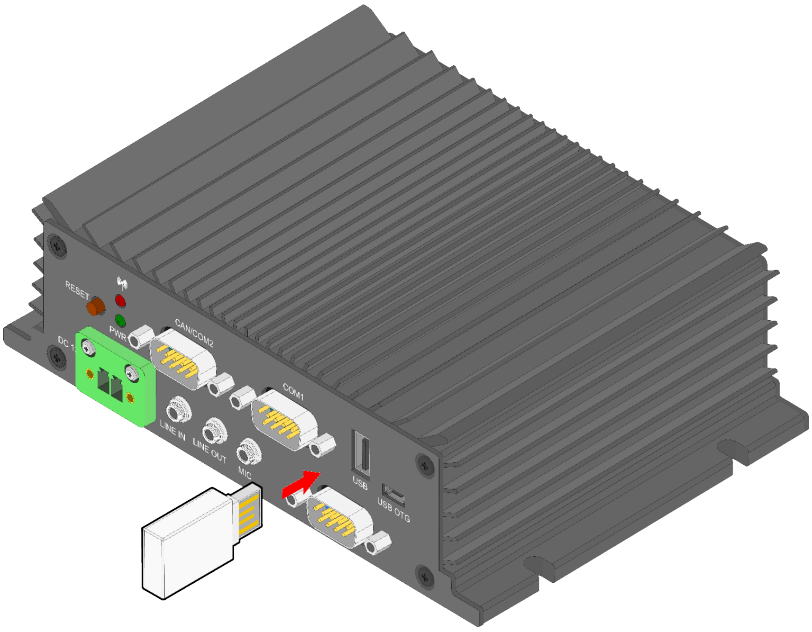
4. Accessories

This section explains how to install and configure the various EMIO modules available for the AMOS-820 system.

4.1 Configuring the VNT9271 USB Wi-Fi Dongle

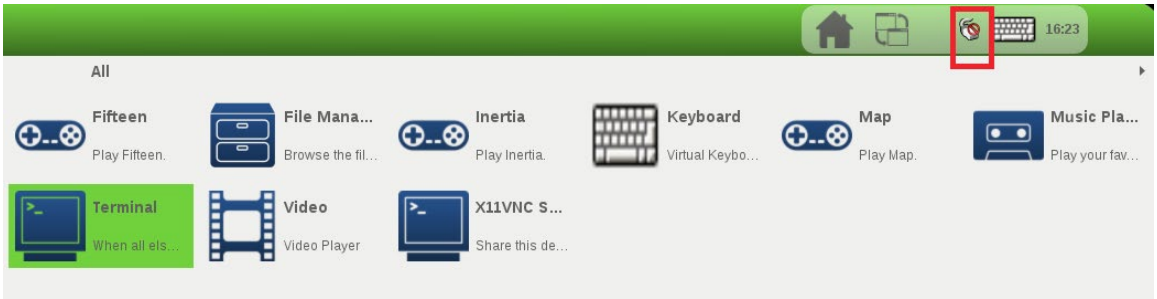
The VNT9271 USB Wi-Fi dongle supports Wi-Fi functionality through a USB port connection.

The first step is to insert the VNT9271 USB Wi-Fi dongle into a USB port. Next, make sure to unplug any LAN cables or other Wi-Fi/3G modules you have installed. Finally, power on the AMOS-820.



Inserting the VNT9271 USB Wi-Fi dongle

When the boot process is completed, click on the “Connection Manager” icon to configure WLAN.



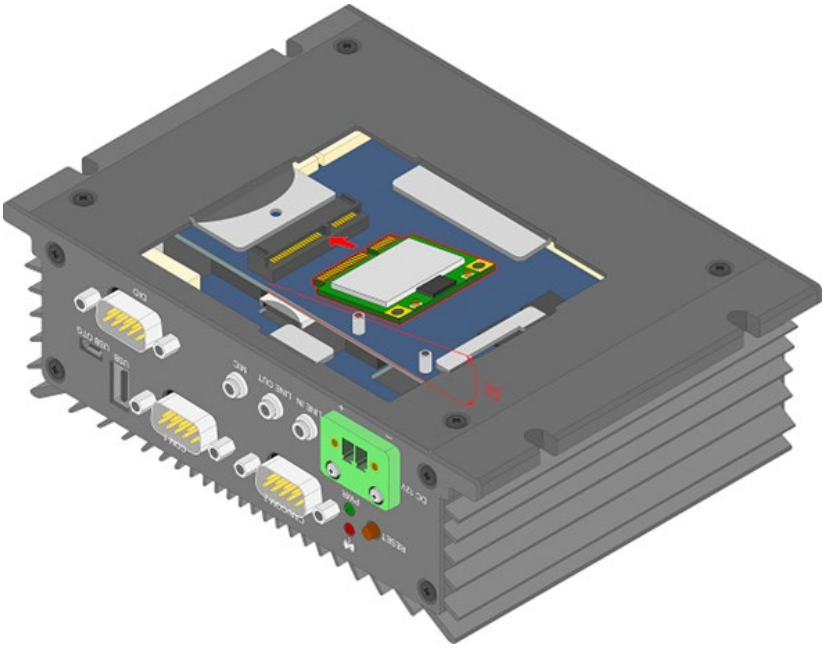
Connection Manager icon diagram

When the connection is created, connect to the internet through your web browser.

4.2 Configuring the EMIO-1541 miniPCIe Wi-Fi Module

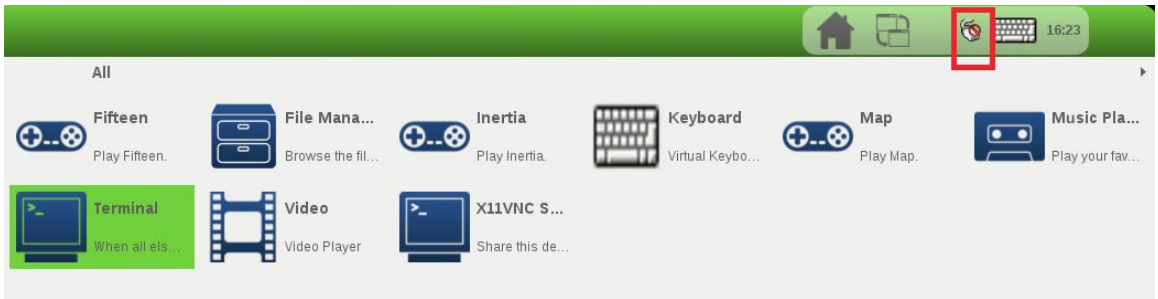
The EMIO-1541 module supports Wi-Fi through the onboard miniPCIe slot.

The first step is to insert the EMIO-1541 module into the miniPCIe slot. After installing the module, connect the provided antenna to the module. Next, make sure to unplug any LAN cables or USB Wi-Fi dongles you have installed. Finally, power on the AMOS-820.



Inserting the EMIO-1541 module

When the boot process is completed, click on the “Connection Manager” icon to configure WLAN.



Connection Manager icon diagram

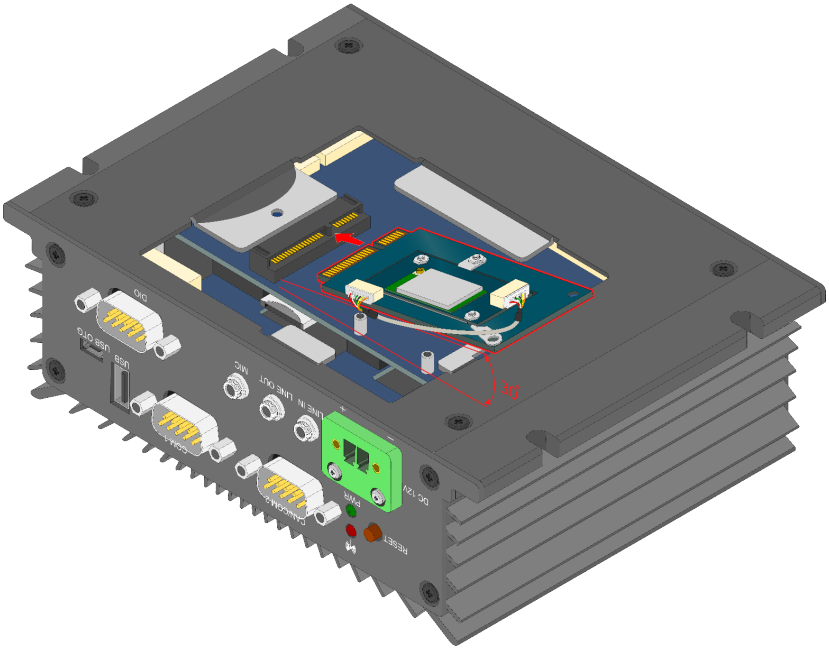
When the connection is created, connect to the internet through your web browser.

4.3 Configuring the EMIO-2531 miniPCIe USB Wi-Fi & Bluetooth Module

The EMIO-2531 module supports Wi-Fi and Bluetooth functionality through miniPCIe connections.

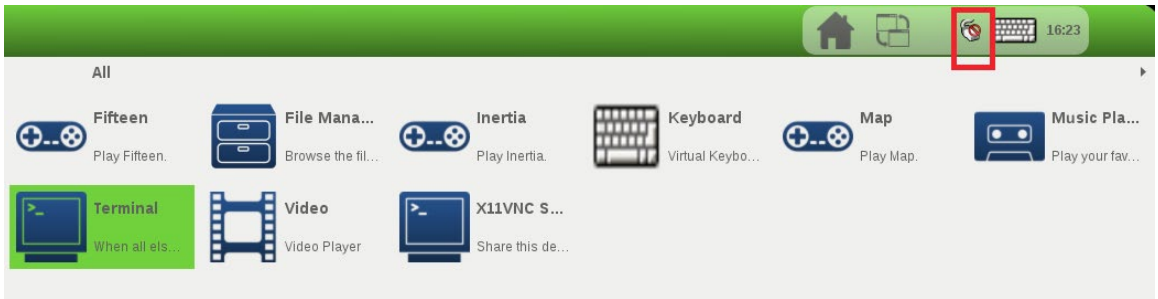
4.3.1 Connecting to the Internet

The first step is to insert the EMIO-2531 module into the miniPCIe slot. After installing either module, connect the provided antenna to the module. Next, make sure to unplug any LAN cables or USB Wi-Fi dongles you have installed. Finally, power on the AMOS-820.



Inserting the EMIO-2531 module

When the boot process is completed, click on the “Connection Manager” icon to configure WLAN.



Connection Manager icon diagram

When the connection is created, connect to the internet through your web browser.

4.3.2 Enabling Bluetooth

The following section explains setting up the Advanced Audio Distribution Profile (A2DP) and Serial Port Profile (SPP).

4.3.2.1 Setting Up Bluetooth A2DP Profile

To add A2DP support to the AMOS-820, first copy and replace the pulseaudio-system.conf, bluetooth.conf to the /etc/dbus-1/system.d folder. Next, copy and replace the system.pa files to the /etc/pulse/folder. These files are located in the EVK/BT_Config folder of the BSP.

To enable the Bluetooth service, use the following command:

```
$ rfkill unblock bluetooth
$ hciconfig hci0 up
$ hciconfig hci0 piscan
$ /usr/lib/bluez5/bluetooth/bluetoothd -C -d &
```

Use the following commands to run the “bluetoothctl” to show the EMIO-2531 Bluetooth MAC address and set up pairing mode.

```
$ bluetoothctl
[NEW] Controller 5C:F3:70:24:4B:2E
[bluetooth]#
[bluetooth]# show
Controller 5C:F3:70:24:4B:2E
      Name: BlueZ 5.25
      Alias: BlueZ 5.25
      …..
[bluetooth]# select 5C:F3:70:24:4B:2E
[bluetooth]# power on
Changing power on succeeded
[bluetooth]# agent on
Agent registered
[bluetooth]# default-agent
Default agent request successful
[bluetooth]# discoverable on
Changing discoverable on succeeded
[CHG] Controller 5C:F3:70:24:4B:2E Discoverable: yes
[bluetooth]# pairable on
Changing pairable on succeeded
[bluetooth]# scan on
Discovery started
[CHG] Controller 5C:F3:70:24:4B:2E Discovering: yes
[NEW] Device 00:1D:82:BC:C1:C4 Jabra BT-530
…
[bluetooth]# scan off
[bluetooth]# devices
Device 00:1D:82:BC:C1:C4 Jabra BT-530
Device …
[bluetooth]# pair 00:1D:82:BC:C1:C4
[bluetooth]# trust 00:1D:82:BC:C1:C4
```

To enable Bluetooth device, use the following commands:

```
[bluetooth]# connect 00:1D:82:BC:C1:C4
[bluetooth]# info 00:1D:82:BC:C1:C4
[bluetooth]# quit
```

The default audio output setting for the AMOS-820 is HDMI device.

To list all the available audio output devices, use the following command:

```
$ pactl list short sinks
0 alsa_output.platform-sound-hdmi.23.analog-stereo module-alsa-card.c s16le 2ch 48000Hz
SUSPENDED
1 alsa_output.platform-sound.22.analog-stereo  module-alsa-card.c s16le 2ch 48000Hz
SUSPENDED
2 bluez_sink.00_1D_82_BC_C1_C4    module-bluez5-device.c  s16le 2ch 48000Hz       IDLE
```

From the output, we can see the following:

- 0 = HDMI device
- 1 = Line-out
- 2 = Bluetooth device

To set up the audio output to a Bluetooth device, use the following command:

```
$ pacmd set-default-sink 2
```

***Note:** If you would like to change audio output after enabling a Bluetooth device, use the commands below:

To set up the audio output to Line-out, use the following command:

```
$ pacmd set-default-sink 1
```

To set up the audio output to HDMI device, use the following command:

```
$ pacmd set-default-sink 0
```

4.3.2.2 Setting Up Bluetooth SPP Profile

The AMOS-820 supports SPP server mode and SPP client mode.

```
$ rfkill unblock bluetooth
$ hciconfig hci0 up
$ hciconfig hci0 piscan
$ /usr/lib/bluez5/bluetooth/bluetoothd -C -d &
```

Use the following commands to run the “bluetoothctl” to show the Bluetooth MAC address and set up pairing mode for the EMIO-2531 module.

```
$ bluetoothctl
[NEW] Controller 5C:F3:70:24:4B:2E
[bluetooth]# scan on
Discovery started
[CHG] Controller 5C:F3:70:24:4B:2E Discovering: yes
[NEW] Device 5C:F3:70:25:DD:33 BlueZ 5.25
…
[bluetooth]# scan off
[bluetooth]# devices
Device 5C:F3:70:25:DD:33 BlueZ 5.25
Device …
[bluetooth]# pair 5C:F3:70:25:DD:33
[bluetooth]# trust 5C:F3:70:25:DD:33
[bluetooth]# quit
```

SPP server mode:
If you do not change the listen channel number, the service will automatically use the default channel number. Make sure the server or client mode is using the same Bluetooth channel number.

To set up the SPP server mode and allow other devices to connect, use the following commands:

```
# sdptool add SP
Serial Port service registered
# rfcomm listen hci0
Waiting for connection on channel 1
Connection from 5C:F3:70:25:DD:33 to /dev/rfcomm0
Press CTRL-C for hangup
```

SPP client mode:
If you do not change the listen channel number, the service will automatically use the default channel number. Make sure the server or client mode is using the same listen channel number.

Use the following commands to set up the SPP client mode and request a connection to SPP server.

```
# sdptool add SP
Serial Port service registered
# rfcomm connect hci0 5C:F3:70:24:4B:2E
Connected /dev/rfcomm0 to 5C:F3:70:24:4B:2E on channel 1
Press CTRL-C for hangup
```

To open the minicom serial communication program, use the following command:

```
# minicom -s
```

Connect to the /dev/rfcomm0 serial device with the following settings:

```
+-----------------------------------------------------------------+
|  A  -         Serial Device                 : /dev/ttyUSB1     |
|  B  -       Lockfile Location               : /var/lock        |
|  C  -         Callin Program                :                  |
|  D  -        Callout Program                :                  |
|  E  -          Bps/Par/Bits                 : 115200 8N1       |
|  F  -       Hardware Flow Control           : No               |
|  G  -       Software Flow Control           : No               |
+-----------------------------------------------------------------+
```

Serial port setting

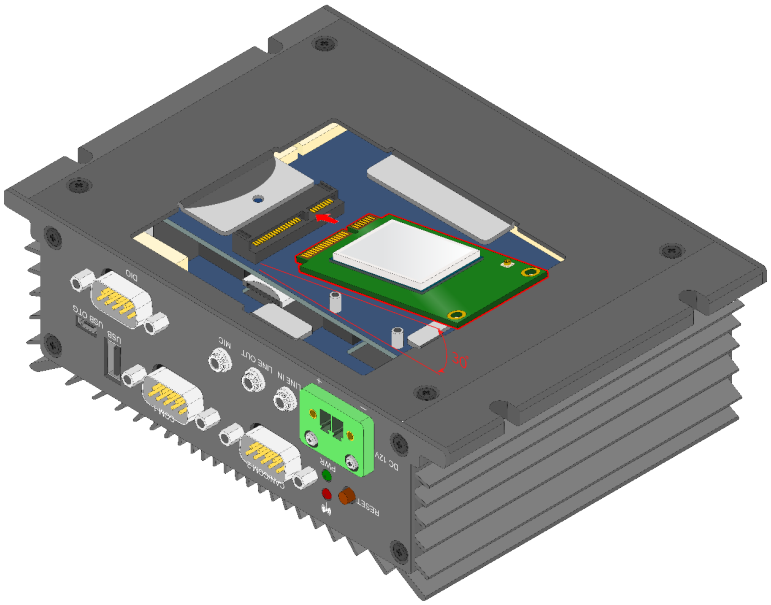
Start a serial communication program on the host machine with the same communication settings using the appropriate Bluetooth SPP device. Afterwards the Bluetooth SPP device and the AMOS-820 will be able to communicate through the programs (for example sending a keypress on one machine should be shown on the other machine).

4.4 Configuring the EMIO-2550 miniPCIe Mobile Broadband Module

The EMIO-2550 miniPCIe Mobile Broadband module supports 3G and GPS functions.

4.4.1 Connecting to the Internet

The first step is to insert an active SIM card into the EMIO-2550 module, and then insert the EMIO-2550 module into the miniPCIe slot. After installing the module, connect the provided antenna to the module. Next, make sure to unplug any LAN cables or USB Wi-Fi dongles you have installed. Finally, power on the AMOS-820.



Inserting the EMIO-2550 module

To check that the system has correctly detected the EMIO-2550 module, use the following command:

```
$ lsusb
```

Make sure the printout message includes “Bus 001 Device 004: ID 1545:1102 U-Blox AG”.

Configure the 3G function by creating a PPP connection with root privilege.

```
$ pppd call Module-ZU200
```

Once the PPP connection has been created, connect to the internet through your web browser.

4.4.2 Enabling GPS

Configuring the GPS function of the module requires the use of the Minicom serial communication program. Open the Minicom serial communication program with the following command:

```
$ minicom -s
```

Connect to the /dev/ttyUSB1 serial device with the following settings:

```
+-----------------------------------------------------------------+
|  A  -         Serial Device              : /dev/ttyUSB1        |
|  B  -       Lockfile Location            : /var/lock           |
|  C  -         Callin Program             :                     |
|  D  -        Callout Program             :                     |
|  E  -          Bps/Par/Bits              : 115200 8N1          |
|  F  -       Hardware Flow Control        : No                  |
|  G  -       Software Flow Control        : No                  |
+-----------------------------------------------------------------+
```

Serial port setting of EMIO-2550

When the setup is complete, type the following GPS AT commands into Minicom: AT+UGPRF=1, AT+UGRMC=1, AT+UGGSV=1, AT+UGZDA=1, and AT+UGPS=1,0

Wait for the “OK” message before sending the next AT command.

```
AT+UGPRF=1
OK
AT+UGRMC=1
OK
AT+UGGSV=1
OK
AT+UGZDA=1
OK
AT+UGPS=1,0
OK
```

Use the Minicom to set the serial device /dev/ttyUSB7 in another terminal and it will output the data received from the GPS module as illustrated in the screenshot below.

```
$GPRMC,142015,00,A,2501.93961,N,12133.66111,E,0.068,,030314,,,A*73
$GPGSV,4,1,13,01,34,184,29,03,49,023,47,06,26,042,41,07,44,317,26*75
$GPGSV,4,2,13,08,13,323,,11,60,192,,13,29,242,38,16,33,071,16*79
$GPGSV,4,3,13,19,65,356,44,23,18,208,12,27,35,034,27,30,37,145,29*7B
$GPGSV,4,4,13,32,01,154,*4B
$GPZDA,142015.00,03,03,2014,00,00*62
```

GPS NMEA message

Taiwan Headquarters

1F, 531 Zhong-zheng Road,
Xindian Dist., New Taipei City 231
Taiwan

Tel: 886-2-2218-5452
Fax: 886-2-2218-9860
Email: embedded@via.com.tw

USA

940 Mission Court
Fremont, CA 94539,
USA

Tel: 1-510-687-4688
Fax: 1-510-687-4654
Email: embedded@viatech.com

Japan

3-15-7 Ebisu MT Bldg. 6F,
Higashi, Shibuya-ku
Tokyo 150-0011
Japan

Tel: 81-3-5466-1637
Fax: 81-3-5466-1638
Email: embedded@viatech.co.jp

China

Tsinghua Science Park Bldg. 7
No. 1 Zongguancun East Road,
Haidian Dist., Beijing, 100084
China

Tel: 86-10-59852288
Fax: 86-10-59852299
Email: embedded@viatech.com.cn

Europe

Email: embedded@via-tech.eu