QUICK START GUIDE

AMOS-825

Android EVK v5.0.3

1.00-051702018-120700

Copyright

Copyright © 2018 VIA Technologies Incorporated. All rights reserved.

No part of this document may be reproduced, transmitted, transcribed, stored in a retrieval system, or translated into any language, in any form or by any means, electronic, mechanical, magnetic, optical, chemical, manual or otherwise without the prior written permission of VIA Technologies, Incorporated.

Trademarks

All brands, product names, company names, trademarks and service marks are the property of their respective holders.

Disclaimer

VIA Technologies makes no warranties, implied or otherwise, in regard to this document and to the products described in this document. The information provided in this document is believed to be accurate and reliable as of the publication date of this document. However, VIA Technologies assumes no responsibility for the use or misuse of the information (including use or connection of extra device/ equipment/add-on card) in this document and for any patent infringements that may arise from the use of this document. The information and product specifications within this document are subject to change at any time, without notice and without obligation to notify any person of such change.

VIA Technologies, Inc. reserves the right the make changes to the products described in this manual at any time without prior notice.

Revision History

Version	Date	Remarks
1.00	05/17/2018	Initial release

Table of Contents

1. Introduction

This Quick Start Guide provides an overview on how to boot the Android EVK system image for the AMOS-825 system and configure the supported hardware functions in the build.

The AMOS-825 Android EVK v5.0.3 is developed based on the NXP android_m6.0.1_1.0.0-ga (Android 6.0 Marshmallow) and enables the hardware features of the AMOS-825 system.

1.1 EVK Package Content

There are three folders in the package listed as below.

Firmware folder	Description
Images_autoinstall_sd.zip	Android EVK system image and installation script files
Document folder	**Description**
AMOS-825_Android_EVK_v5.0.3_Quick_Start_Guide _v1.00_20180517.pdf	Quick Start Guide
Tool folder	**Description**
AMOS-825_Smart_ETK_Demo_v1.0.apk	Smart ETK demo program
BluetoothSPPTest.apk	Bluetooth SPP testing program

AMOS-825 Android EVK contents

1.1.1 Firmware Folder Contents

Images_autoinstall_sd.zip: contains installation script files and the precompiled U-boot and Android image for evaluating the AMOS-825 system.

1.1.2 Document Folder Contents

AMOS-825_Android_EVK_v5.0.3_Quick_Start_Guide _v1.00_20180517.pdf: This Quick Start Guide provides an overview on how to boot the Android EVK system image for the AMOS-825 system and configure the supported hardware functions in the build.

1.1.3 Tools Folder Contents

AMOS-825_Smart_ETK_Demo_v1.0.apk: is the demo program of the Smart ETK.

BluetoothSPPTest.apk: is the testing program of the Bluetooth SPP profile.

1.2 Version Information and Supported Features

- U-Boot version: 2015.04
- Kernel version: 3.14.52
- Evaluation image: Android Marshmallow 6.0
- Development based on NXP android_M6.0.1_1.0.0-ga (Android 6.0 Marshmallow)
- Supports SPI with eMMC boot
- Supports 7” Projective capacitive touch monitor (800x480) through I2C interface
- Supports COM as debug port
- Supports two FlexCAN TX/RX
- Supports Gigabit Ethernet
- Supports Mic-in
- Supports IEEE 802.11b/g/n Wi-Fi
- Supports Bluetooth 4.0
  - A2DP and SPP profile
- Supports U-blox MAX-7 GPS/GNSS module
- Supports EMIO-2550 miniPCIe Mobile Broadband module
- Supports Smart ETK v1.00: Watchdog Timer, UART, and FlexCAN
- Supports OTA (Over-The-Air technology)
- Supports shutdown option in Quick Settings
- Supports Ethernet configuration in Settings

2. Image Installation

The AMOS-825 Android system only provides booting from the SPI ROM with eMMC. This section explains the setup requirements for booting from the SPI ROM and eMMC.

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

2.1 Booting from a Micro SD Card

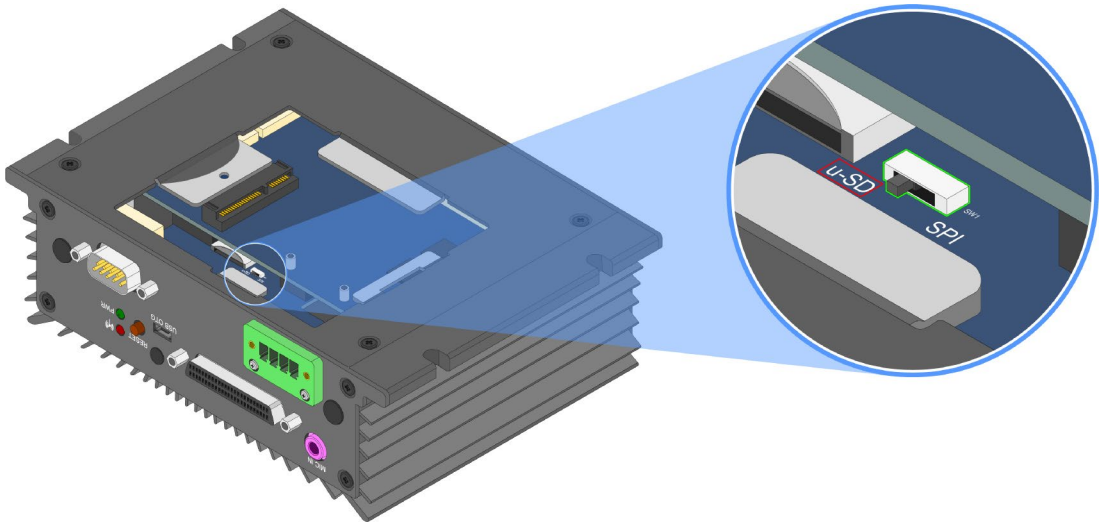
The **Images_autoinstall_sd.zip** includes the Android EVK system image and the installation scripts files.

The first step is to extract **Images_autoinstall_sd.zip** file to make the bootable Micro SD card. Insert a Micro SD card into your Linux host machine and make sure it is not mounted. Open the terminal on your host machine. Run the **mk_android6_install_sd.sh** as shown below, replacing <device name> with the correct value for the card, for example “sdb”.

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

```
$ sudo ./mk_android6_install_sd.sh /dev/<device name>
```

Next, on the AMOS-825, set the Micro SD/SPI boot switch to the Micro SD position as shown below.



Micro SD/SPI boot switch diagram

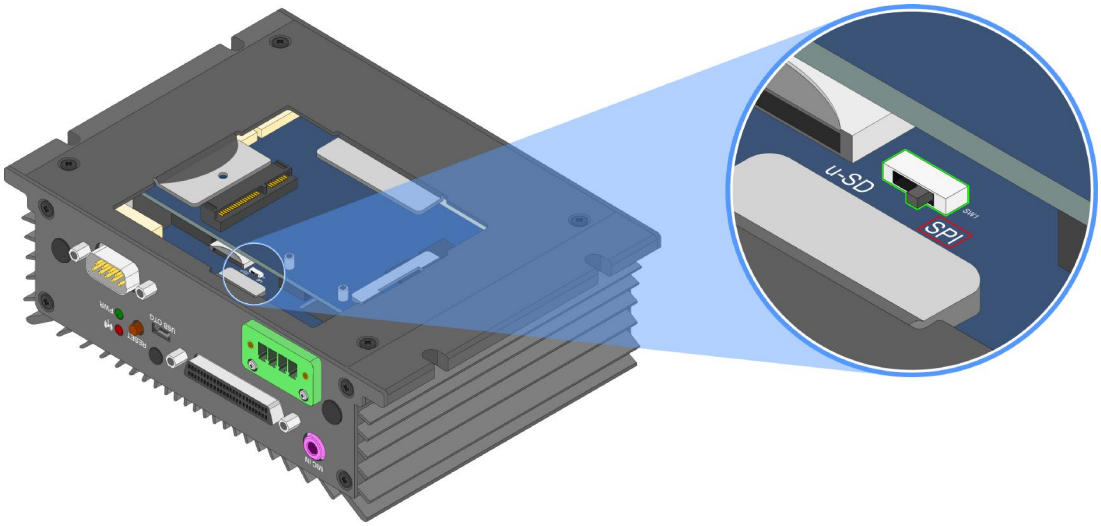
Insert the prepared Micro SD card into the AMOS-825, connect an HDMI display, and power on the device to initiate the update process automatically.

```
VIA Technologies, Inc.
iMX6_Android6.0.1_v5.0.3_beta1

-----------------------------------

   U-Boot Version : v2015.04-0.1.0
   Kernel Version : v3.14.52-0.1.0

   Base File System Version : 0.1.0
   Reference File System Version : 0.1.0
   OtherInfo :

   [=========                    ] 30 %
   erasing spi flash (mtd0)

   Warnings! Please don't power off! Please wait...
```

Update process screen

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

In order to boot from the SPI ROM make sure the Micro SD/SPI boot switch is set to SPI ROM boot.



Micro SD/SPI boot switch diagram

Next, power on the device to initiate the boot process. When the boot process is completed, you will see the Android desktop.

3. Hardware Function

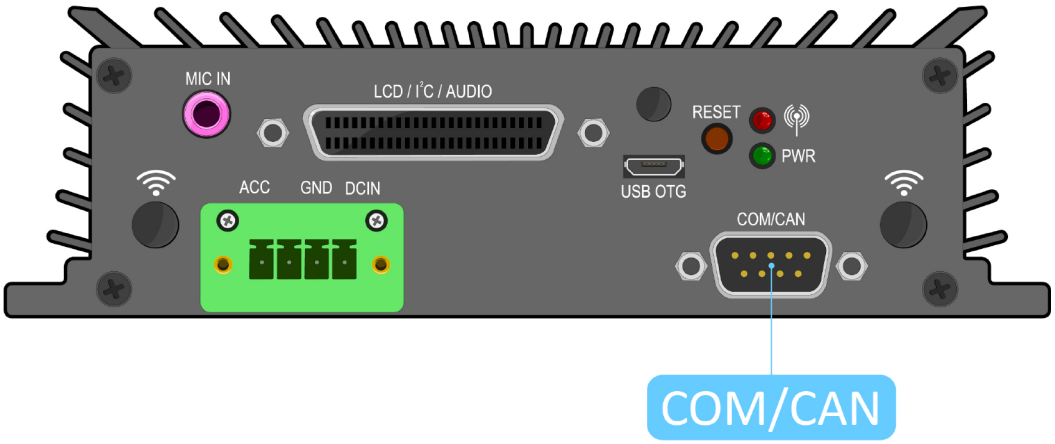
This section explains how to enable and test the hardware functions precompiled in the AMOS-825 Android EVK including setting U-Boot parameters, restoring default U-Boot parameters, using the Android Console, enabling GPS, enabling Wi-Fi, enabling Bluetooth, as well as installing and applying the different functions included in the VIA Smart ETK sample program which are testing Watchdog timer function and testing the FlexCAN function.

3.1 Setting Up U-Boot Parameters

The first step is to connect the host machine and the AMOS-825 through the COM 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 port diagram

Next, power on the AMOS-825 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 screenshot below.

```
U-Boot 2015.04 (Mar 7 2017 - 18:49:13)AMOS-825 ver:5.0.3

CPU:   Freescale i.MX6Q rev1.2 at 996 MHz
CPU:   Temperature 30 C, calibration data: 0x5894f169
Reset cause: POR
Board: iMX6Q-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
*** Warning - bad CRC, using default environment

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
```

Make sure the printout message is as follows:

```
=> printenv

baudrate=115200
boot_emmc_root=mmc dev ${root_media_uboot};ext2load mmc ${root_media_uboot}:${root_partition}
${loadaddr} uImage;ext2load  mmc ${root_media_uboot}:${root_partition} ${dtbaddr} imx6q-
vab820.dtb;bootm ${loadaddr} - ${dtbaddr};
boot_media=booti mmc1
```

3.2 Restoring Default U-Boot Parameters

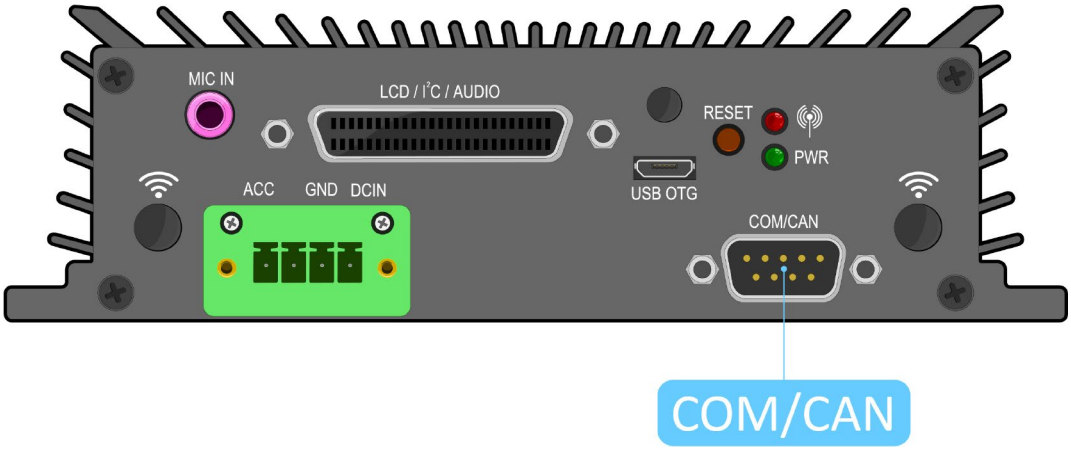
If the U-Boot parameters have been modified, the “destroyenv” command in the U-Boot console can restore the factory default settings.

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

```
=> destroyenv
=> saveenv
=> Reset
```

3.3 Using the Android Console

The first step is to connect the host machine and the AMOS-825 through the COM 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-825 to initiate the boot process. When the boot process is completed you will be prompted to login.



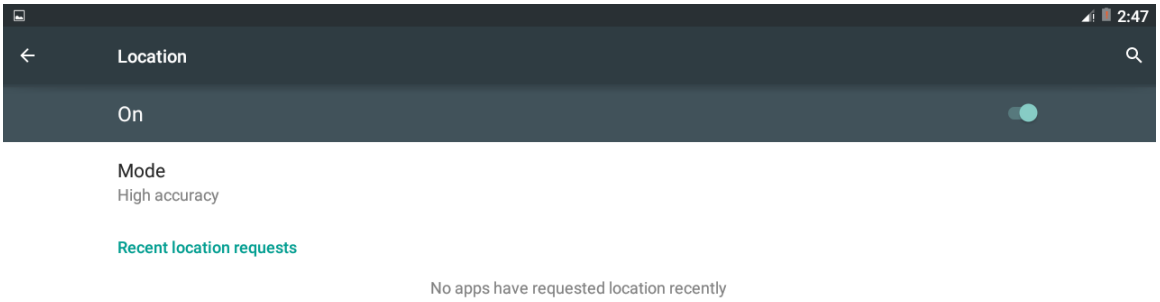
COM port diagram

When the boot process is completed, you will automatically log in to an Android console.

```
root@vab820_6dq:/ #
```

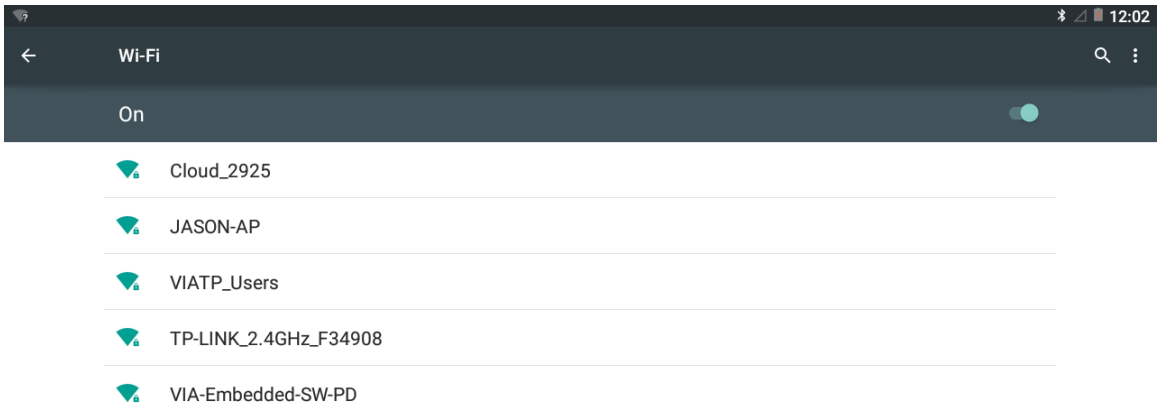
3.4 Enabling GPS

Click Settings->Location to enable the option.



3.5 Enabling Wi-Fi

Next, from the Settings screen, click Wi-Fi -> On to enable Wi-Fi function.



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

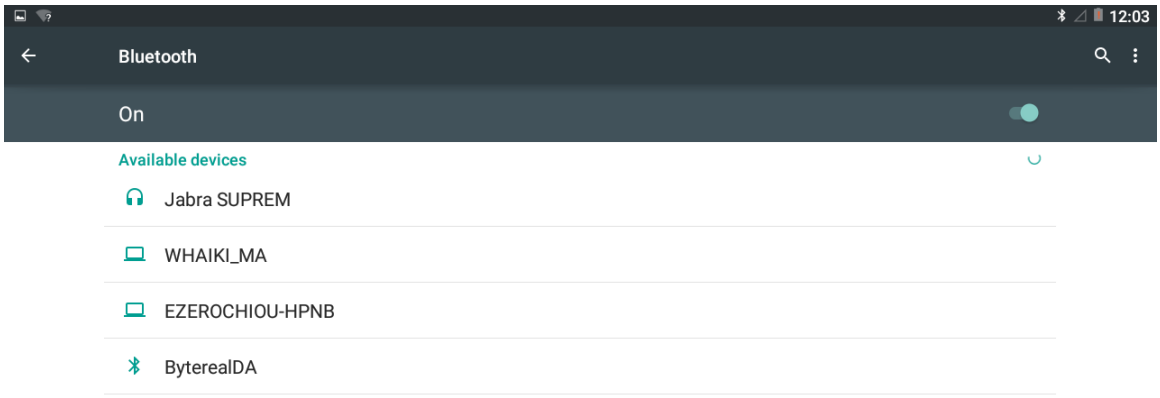
3.6 Enabling Bluetooth

The following sections show how to enable the Bluetooth Advanced Audio Distribution Profile (A2DP) to allow audio playback through a connected Bluetooth device as well as how to configure the Bluetooth Serial Port Profile (SPP).

3.6.1 Setting Up Bluetooth A2DP Profile

First, put the accessory you want to use into discovery mode. The exact way to do this depends on the accessory. If you have a headset, you may need to hold a button down on the headset for several seconds until a light starts flashing. It will only stay discoverable for a few minutes.

If you are not sure how to put your accessory into discovery mode, please refer to its manual, check the manufacturer's website, or perform a web search for instructions. To enable the Bluetooth A2DP function, go to Settings -> Bluetooth and set the switch to On.



A list of local devices will appear on the screen, select the appropriate device to complete the Bluetooth pairing.

3.6.2 Setting Up Bluetooth SPP Profile

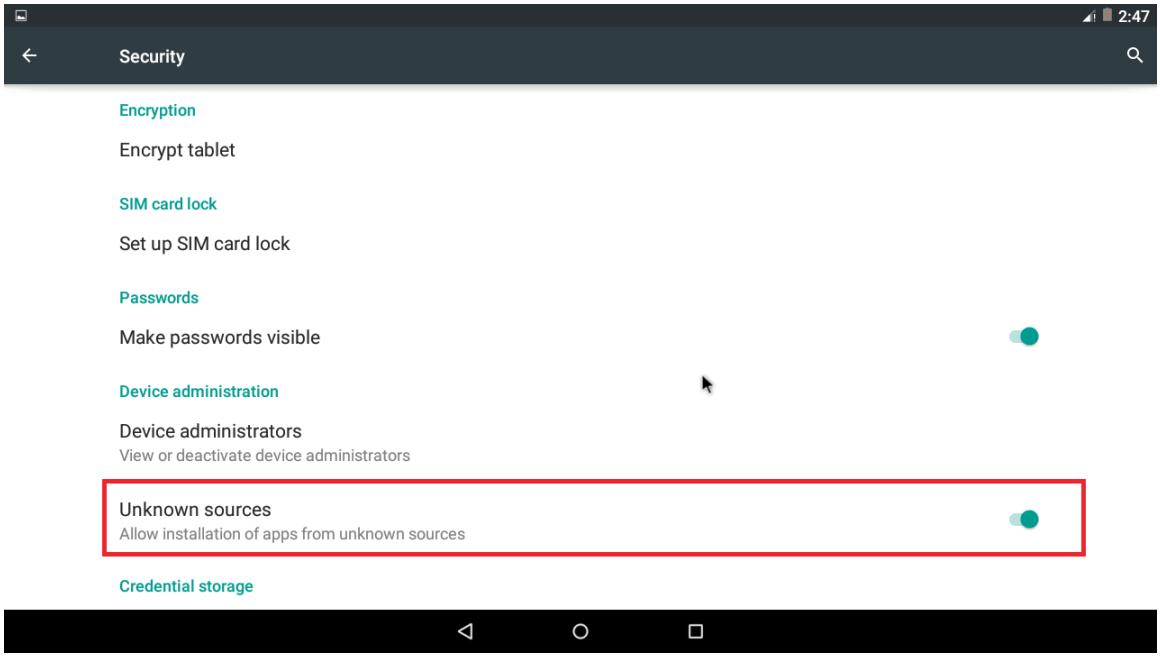
The first step is to copy the **BluetoothSPPTest.apk** onto a mass storage device such as USB thumb drive. Next, from the Settings screen, click Security -> Unknown sources to allow installation of non-Market apps.

The AMOS-825 implements the Bluetooth Serial Port Profile allowing serial port communication between two Android devices.

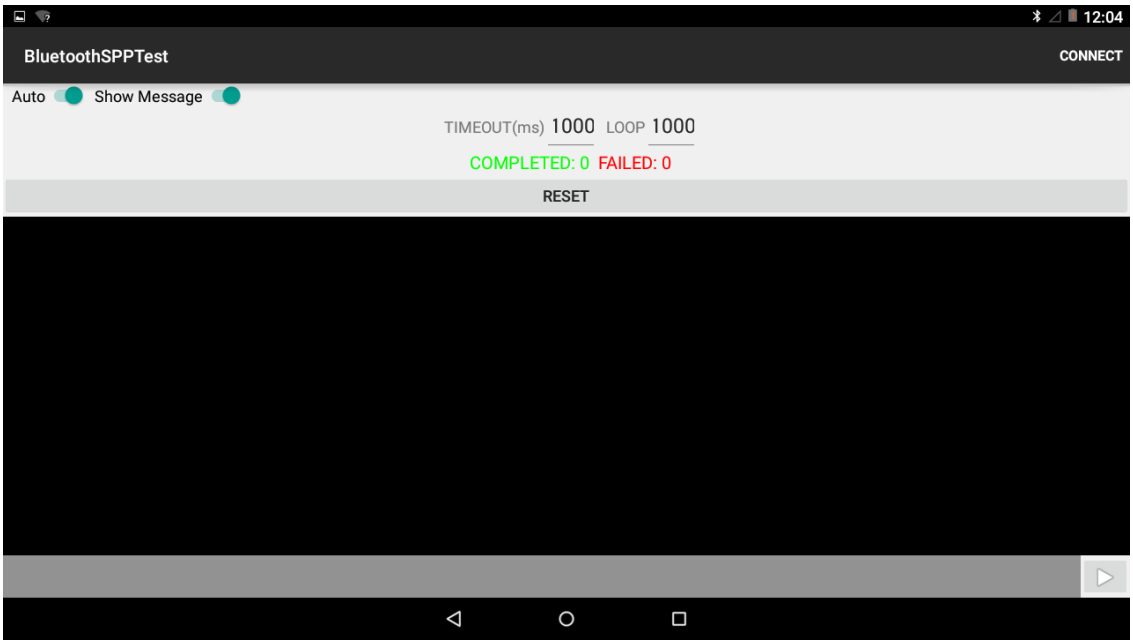
Included in the AMOS-825 Tools folder is the **BluetoothSPPTest.apk** which is a simple communication application which utilizes the Bluetooth SPP Profile to transmit and receive data between two paired Android devices.

The following example will demonstrate how to use the **BluetoothSPPTest.apk** to communicate over the Bluetooth SPP Profile between two AMOS-825 systems.

First, the **BluetoothSPPTest.apk** must be installed onto each AMOS-825 system. From the Settings screen, click Security -> Unknown sources and allow the installation of non- Market apps. Next, copy the **BluetoothSPPTest.apk** onto a mass storage device, such as a USB thumb drive, and install the **BluetoothSPPTest.apk** onto both AMOS-825 systems.



After the installation process has completed, go to Settings -> Bluetooth -> On to enable the Bluetooth function on both AMOS-825 systems. A list of local devices will then appear on each screen. From either screen, select the AMOS-825 system from the list to complete the pairing process as seen in the figure below.



BluetoothSPPTest diagram

After the devices have paired, open the **BluetoothSPPTest.apk** on both AMOS-825 systems and configure the settings as follows in both apps.

“Auto” – enabled

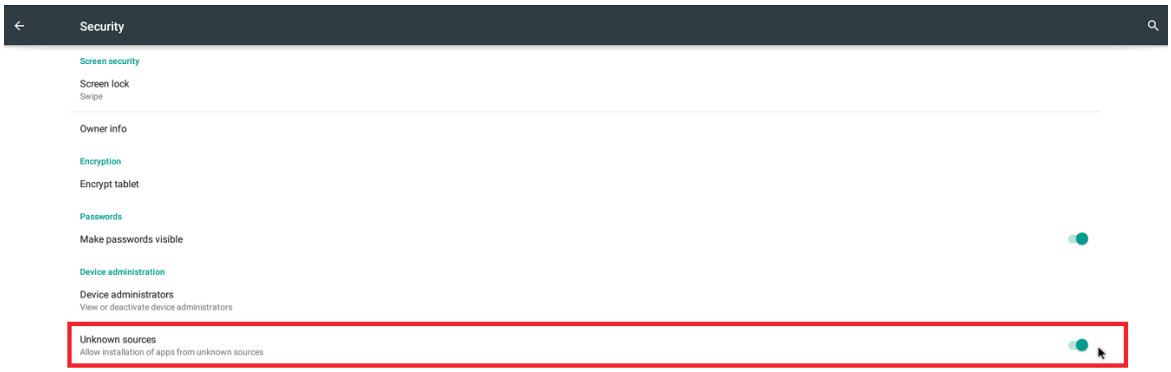
“Show Message” – enabled

Select CONNECT from either AMOS-825 to create the connection between the two. Both AMOS-825 can communicate over the Bluetooth SPP Protocol.

3.7 Smart ETK

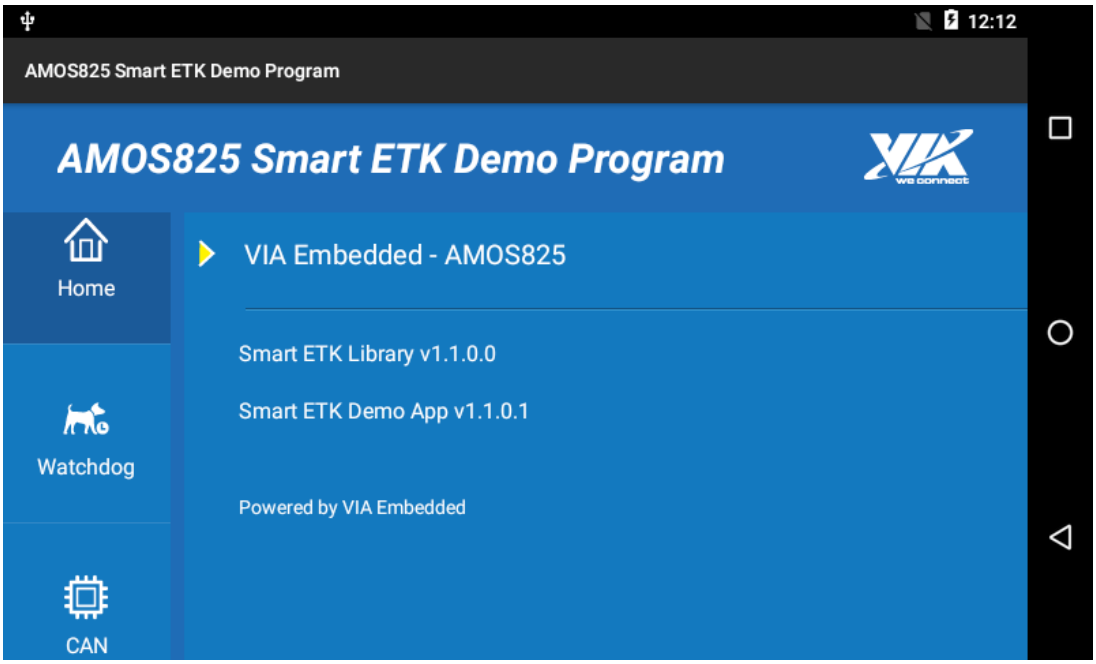
The AMOS-825 Smart ETK supports Watchdog, UART and CAN bus functions. Please follow the procedures below to experiment with the Smart ETK functions on the AMOS-825 system.

The first step is to copy the **AMOS-825_Smart_ETK_Demo_v1.0.apk** onto a mass storage device such as a USB thumb drive. Next, from the Settings screen, click Security, and then switch on the "Unknown sources" as shown in the diagram below.



Finally, insert the USB thumb drive into the AMOS-825 and double click on the AMOS-825_Smart_ETK_Demo_v1.0.apk file to install.

When the installation process has completed, run the AMOS-825_Smart_ETK_Demo_v1.0.apk and start to test the different functions with it.

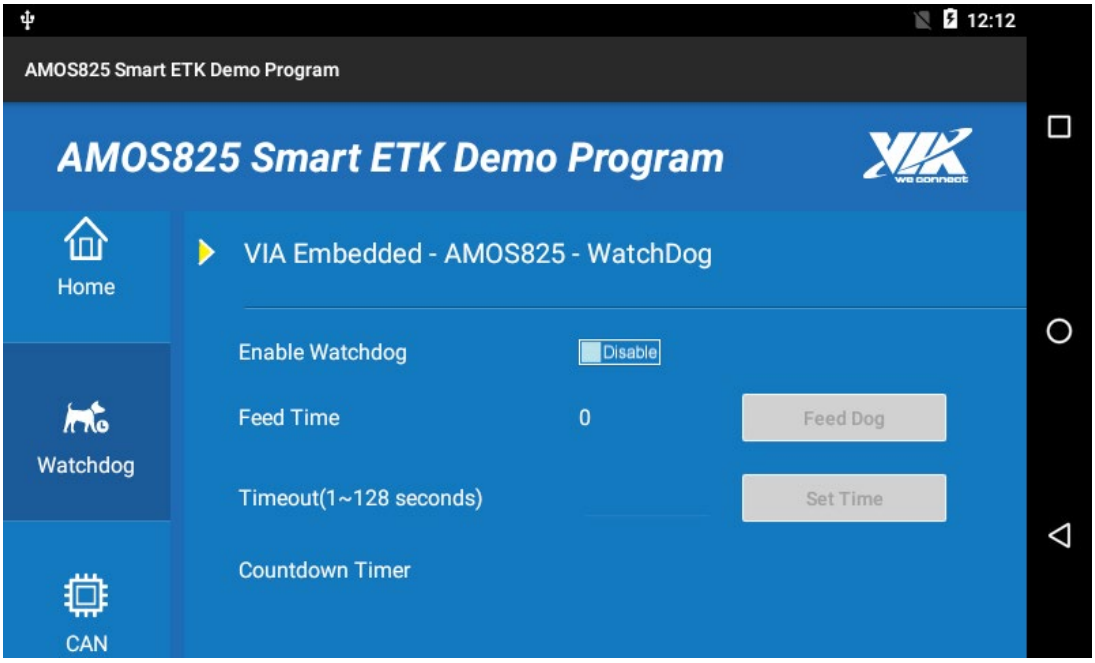


Smart ETK SDK sample screen

3.7.1 Testing Watchdog Timer Function

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

The Watchdog Timer includes Enable/Disable, Set Timeout, Keep Alive and Countdown Timer functions.



Smart ETK Watchdog timer diagram

A list of options will appear on the Smart ETK Demo Program page, select the appropriate option to start testing.

To test the Watchdog function, the first step is to enter the time value (1~128 seconds) in the Timeout setting.

Click on the Enable/Disable button to start the watchdog timer function.

Click on the REFRESH button to refresh the countdown time value back to the beginning.

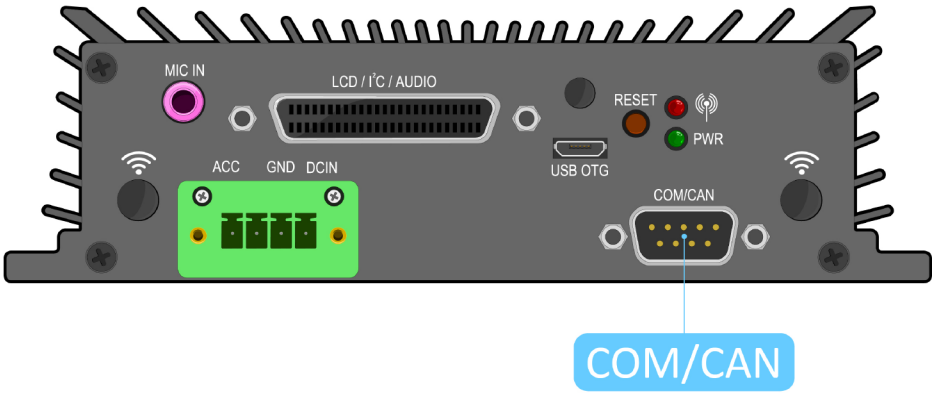
Click on the STOP button to stop the watchdog timer function.

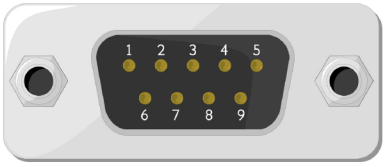
3.7.2 Testing FlexCAN Function

The COM/CAN port of the AMOS-825 supports debug(COM) and two CAN bus. The CAN bus supports CAN protocol specification Version 2.0 B while the COM 2 supports TX/RX for debugging purposes only.

In order to connect the CAN device for testing, you need to add additional transceiver circuit. Please visit Texas Instruments website (https://www.ti.com/) to search information on “TI SN65HVD1050 CAN Bus Transceiver”.

The first step is to make sure there is connection with CAN1 and CAN2.

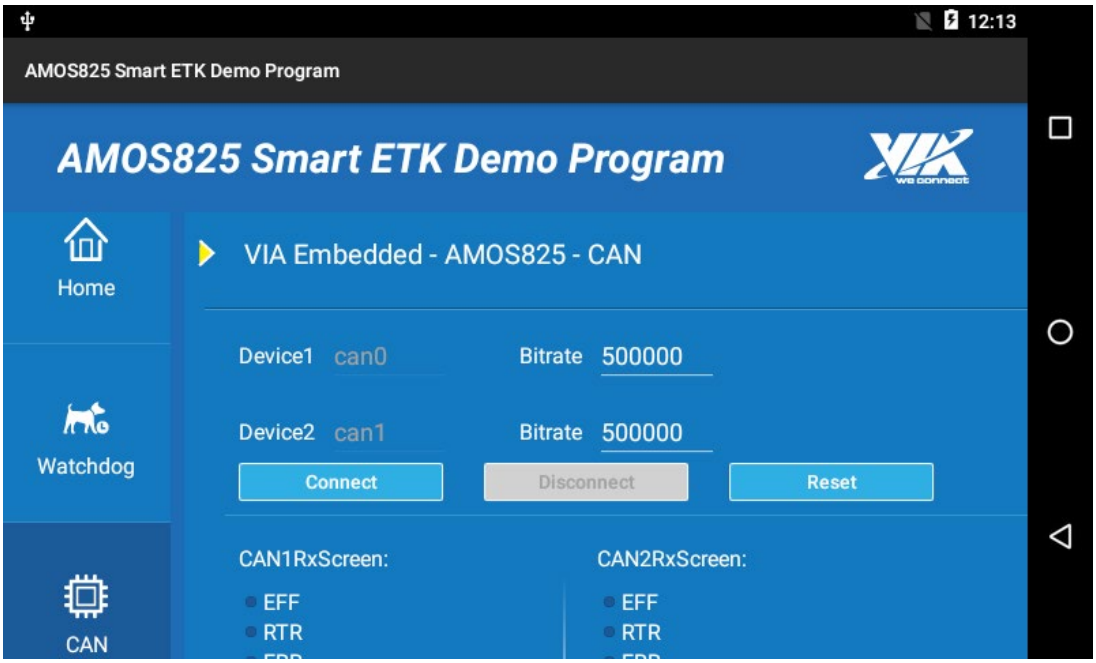




COM/CAN port diagram

Pin	Signal
1	CANH1
2	RX
3	TX
4	CANL2
5	GND
6	CANL1
7	GND
8	CANH2
9	VCC5
10	CAN RX1

COM/CAN port pin definition table



Smart ETK FlexCAN diagram

The CAN ID frame structure are 32 bits, one frame includes 1-bit EFF flag, 1-bit RTR flag, 1-bit ERR flag, an 18-bit extension ("identifier extension") and 11-bit identifier ("base identifier").

In order to use the identifier extension, you need to click "EFF".

If you select "RTR, the RTR-bit is transmitted as a dominant bit in the Data Frame and the Data Field will be empty in the Remote Field.

First, please open the Smart ETK sample program, then from the left side of the SmartETK sample page; select CAN to start testing.

For base testing, set the BitRate value and click on the Connect button.

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

Next, type the ID and data in the CAN1 (white block in the diagram) and click the Send button.

The CAN1 transfers data to CAN2, the data shows in the CAN2 (red block in the diagram).

Click on the Reset button to reset this CAN bus function.

Click on the Disconnect button to disable this function.

For advanced testing, you'll use Filter_ID and FilterMask functions.

The filter mask is used to determine which bits in the identifier of the received frame are compared with the filter. Filter_ID and FilterMask must be typed in hexadecimal values.

Note:
The filter is used to get specific information. Each receiver must set the filter to get the information that is needed.

If a mask bit is set to zero, the corresponding ID bit will automatically be accepted, regardless of the value of the filter bit.

If a mask bit is set to one, the corresponding ID bit will be compare with the value of the filter bit.

If they match, it is accepted otherwise the frame is rejected.

There are four examples and these value are hexadecimal values below.

Note:
The four examples have four different statuses.

Example 1.

You wish to accept only frames with ID of 00001567 (hexadecimal values)

Set the filter to 00001567 and mask to 1FFFFFFF.

When a frame arrives, its ID is compared with the filter and all bits must match; any frame that does not match ID 00001567 is rejected

Example 2.

You wish to accept only frames with IDs of 00001560 thru to 0000156F (hexadecimal values)

Set the filter to 00001560 and mask to 1FFFFFF0 .

When a frame arrives, its ID is compared with the filter and all bits except bits 0 to 3 must match; any other frame is rejected

Example 3.

You wish to accept only frames with IDs of 00001560 thru to 00001567 (hexadecimal values)

Set the filter to 00001560 and mask to 1FFFFFF8 .

When a frame arrives its ID is compared with the filter and all bits except bits 0 to 2 must match; any other frame is rejected

Example 4.

You wish to accept any frame

Set the filter to 0 and set mask to 0.

All frames are accepted

4. Accessories

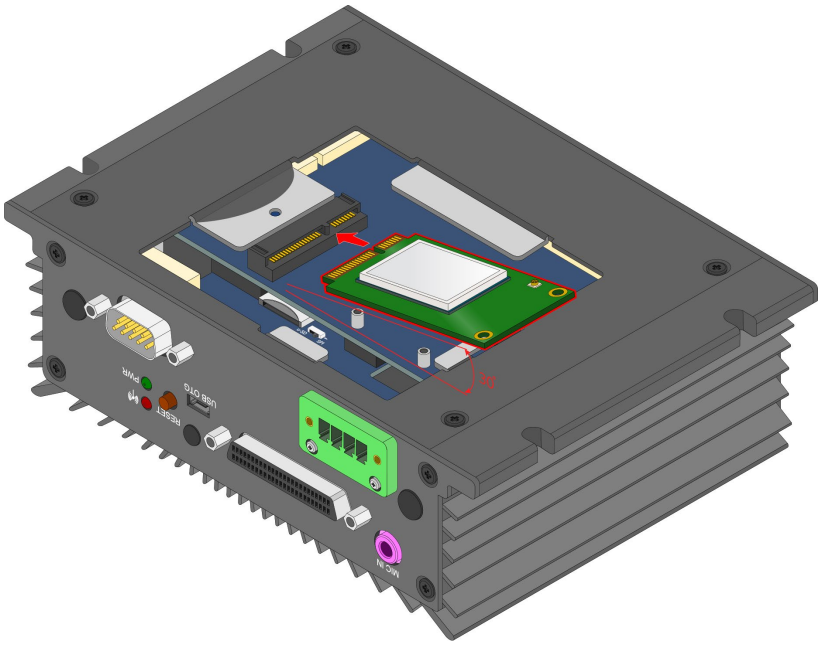
This section explains how to install and configure the EMIO-2550 miniPCIe Mobile Broadband Module available for the AMOS-825 system.

4.1 Configuring the EMIO-2550 miniPCIe Mobile Broadband Module

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

4.1.1 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-825.



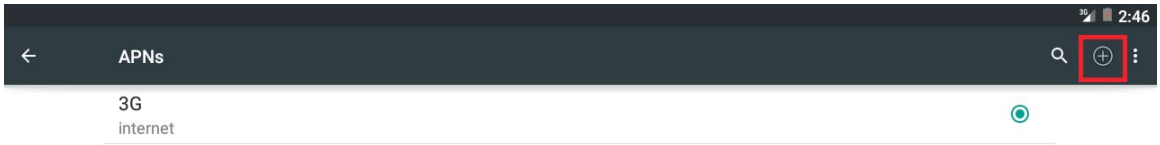
Inserting the EMIO-2550 module

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

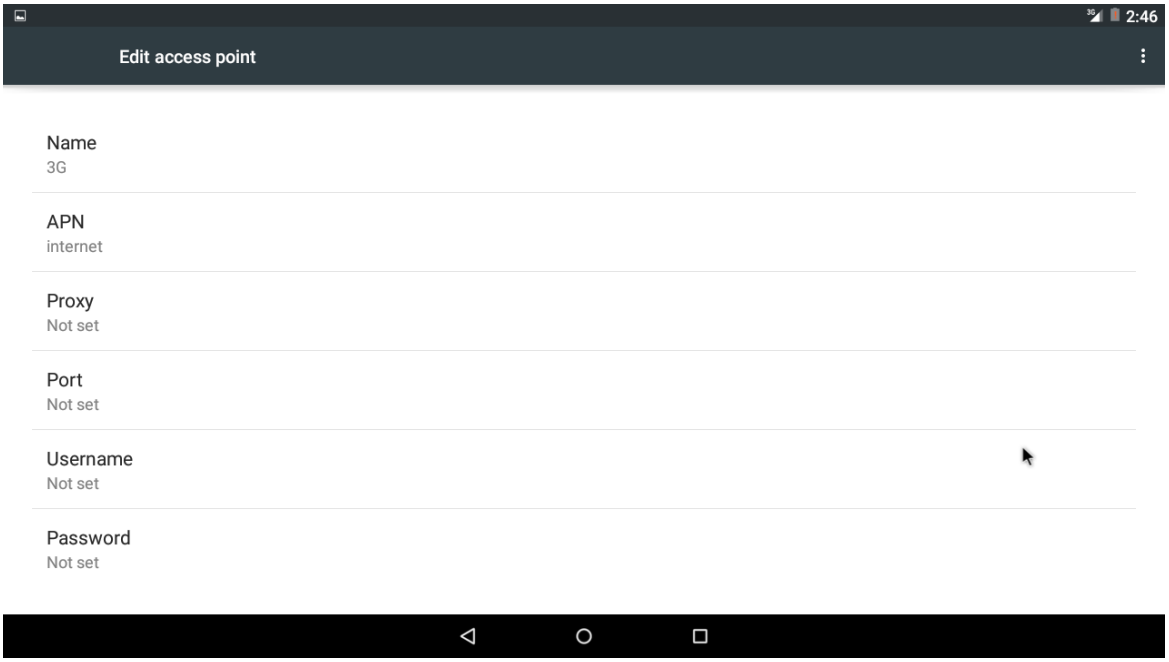
```
root@vab820_6dq:/ # busybox ifconfig
```

Make sure the printout message includes “ppp0 Link encap: Point-to-Point Protocol”.

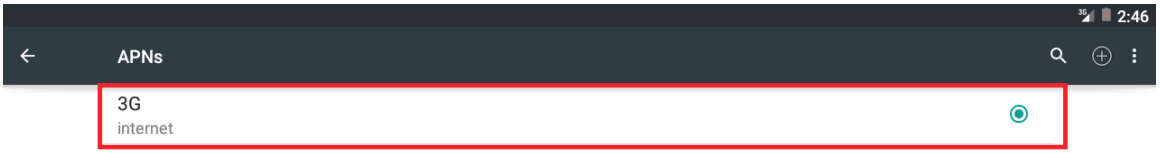
Click Settings->Wireless & networks-> More -> Cellular network ->Access Point Names. Next, click the plus button to add your APNs setting.



Fill in the required fields for APNs setting. If you are unsure of what the required fields and value are, check with your Mobile Broadband provider.



When the APNs setting is completed, click your APNs to enable the 3G network. Next, open the browser to connect to the Internet.



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