

# Integrating a WiLink8™ Module with the AM65x EVM

---

---

---

## ABSTRACT

This application report describes how to integrate the WiLink8 module with the AM65x EVM.

---

### Contents

1	Required Hardware .....	2
2	Prepare the Board for Working With Images in eMMC .....	2
3	Setup the AM654x EVM k the WL18xx-based Module .....	4
4	Integration With the Processor SDK SDK Image for the AM654x EVM .....	5

### List of Figures

1	WiLink™ WL18xx SDIO Board .....	2
2	WL1835MOD COM8 Kit .....	2
3	WL1837MOD COM8 Kit .....	2
4	Boot Mode Switches .....	4
5	Connect the EVM .....	5

### List of Tables

1	Connecting Wires .....	4
---	------------------------	---

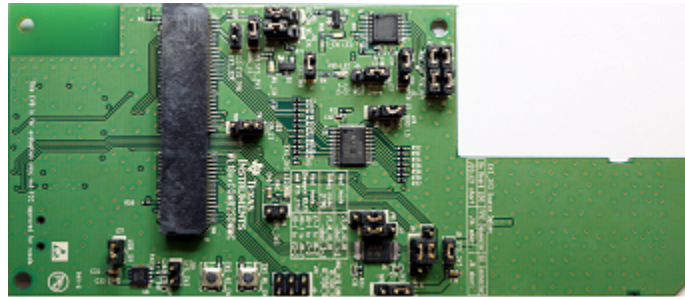
## Trademarks

WiLink8 is a trademark of Texas Instruments.  
Bluetooth is a registered trademark of Bluetooth SIG.  
Linux is a registered trademark of Linus Torvalds.  
Wi-Fi is a registered trademark of Wi-Fi Alliance.  
All other trademarks are the property of their respective owners.

## 1 Required Hardware

In addition to the am65x-evm, the following components are used for this demo setup:

- [wl18xxcom82sdmmc Adapter Kit](#)



**Figure 1. WiLink™ WL18xx SDIO Board**

- [WL1835MOD COM8 Kit](#) or [WL1837MOD COM8 Kit](#)



**Figure 2. WL1835MOD COM8 Kit**



**Figure 3. WL1837MOD COM8 Kit**

## 2 Prepare the Board for Working With Images in eMMC

When using the WiLink8 module with the adapter board, the SD card slot cannot be used for images or the filesystem. As a result, before working with WiLink8, move everything to eMMC (boot loaders and file system).

### 2.1 Getting SDK Images for the AM65x EVM

Remove the WiLink8 adapter board from the sd-card slot.

Download a working SDK image from the following link and create a bootable working SD card that can fully boot the board to the Linux® prompt: [http://software-dl.ti.com/processor-sdk-linux/esd/AM65X/latest/index\\_FDS.html](http://software-dl.ti.com/processor-sdk-linux/esd/AM65X/latest/index_FDS.html)

You can use the following SDK prebuilt image for this: <http://software-dl.ti.com/processor-sdk-linux/esd/AM65X/latest/exports/am65xx-evm-linux-05.01.00.11.img.zip>



5. Untar the image in the new filesystem on the eMMC partition that is mounted by default now in `/var/run/media/mmcbk0p1`.

## 2.6 Switch the Board to Booting From Internal eMMC (No SD Card in Slot)

When the board is programmed to eMMC, set the boot mode switches as shown in [Figure 4](#) to switch to booting from internal eMMC:

- SW2[3:0]=0010
- SW3[3:0]=1101
- SW4[1:0]=11

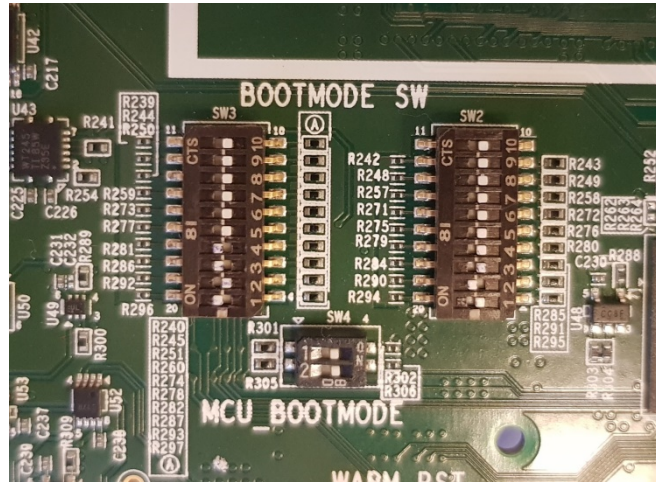


Figure 4. Boot Mode Switches

Next, power off the board, remove the SD card, and power up the board. If this process was completed correctly, the board should fully boot into a Linux prompt.

## 3 Setup the AM654x EVM k the WL18xx-based Module

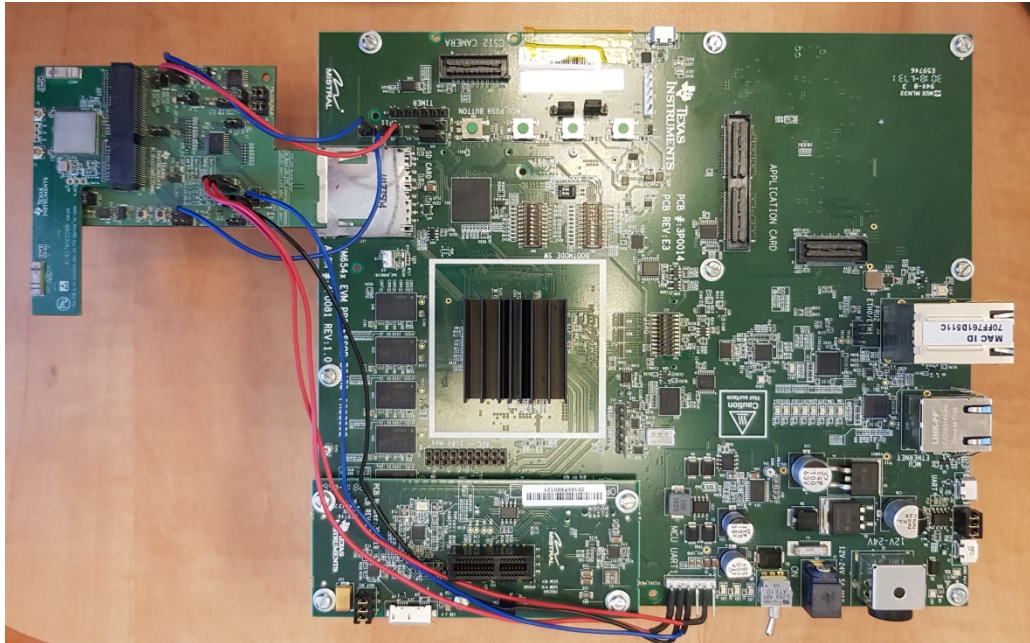
This section describes setting up the am65x-evm platform for use with the WL18xx-based module, which is connected using the SD connection/SDIO slot port of the EVM.

1. Plug the WL18xx adapter board (WL18xxCOM82SDMMC) into the SD card slot port of the AM654x EVM. The adapter board comes with the [\[default\]](#) jumper assembly to work with a host.
2. Plug the WL18xxMOD COM8 kit into the WL18xx SDIO board. Connect the following wires between the AM654x EVM and the WL18xx SDIO board. See [Table 1](#) as a reference. Note the difference in CTS/RTS between the Rev:E2 EVM and Rev:1.0.

Table 1. Connecting Wires

WL18xx Pin Name	AM654x EVM (Rev:E2) Connector-Pin	AM654x EVM (Rev:1.0) Connector-Pin	WL18xxCOM82SDMMC Connector-Pin
WLAN_Enable	J20-1	J20-1	J12-2
WLAN_IRQ	J20-2	J20-2	J19-IRQ-3V3
BT_Enable	J20-4	J20-4	J9-2
BT-UART Tx	J30-5	J30-5	J8-2
BT-UART Rx	J30-4	J30-4	J7-2
BT_UART Rts	J30-6	J30-2	J10-2
BT_UART Cts	J30-2	J30-6	J11-2

3. Connect the AM65x EVM with the WL18xx module mounted into the SD card slot, as shown in [Figure 5](#).



**Figure 5. Connect the EVM**

## 4 Integration With the Processor SDK SDK Image for the AM654x EVM

### 4.1 Getting the TI-Linux Kernel

To build and integrate the WiLink8-related software components into the file system, start by getting the TI Linux kernel being used with the SDK.

Clone the following git: [git://git.ti.com/processor-sdk/processor-sdk-linux.git](https://git.ti.com/processor-sdk/processor-sdk-linux.git)

Checkout the following commit (kernel commit for PROCESSOR-SDK-LINUX-AM65X 05\_01\_00\_11): `git checkout d315a9bb00`

Alternatively, get the 0.5.01.00.11 Linux BSP source code tar: <http://software-dl.ti.com/processor-sdk-linux/esd/AM65X/latest/exports/am65xx-evm-linux-sdk-src-05.01.00.11.tar.xz>

### 4.2 Patch the Kernel for Enabling WiLink8 Support

Apply the following patches to the SDK kernel:

[https://git.ti.com/wilink8-wlan/build-utilites/blobs/internal/patches/kernel\\_patches/am65xx-kernel/0001-mmc-sdhci-of-arasas-Add-sdhci\\_arasas\\_set\\_power.patch](https://git.ti.com/wilink8-wlan/build-utilites/blobs/internal/patches/kernel_patches/am65xx-kernel/0001-mmc-sdhci-of-arasas-Add-sdhci_arasas_set_power.patch)

[https://git.ti.com/wilink8-wlan/build-utilites/blobs/internal/patches/kernel\\_patches/am65xx-kernel/0002-mmc-sdhci-of-arasas-am654-add-a-set\\_power-callback.patch](https://git.ti.com/wilink8-wlan/build-utilites/blobs/internal/patches/kernel_patches/am65xx-kernel/0002-mmc-sdhci-of-arasas-am654-add-a-set_power-callback.patch)

[https://git.ti.com/wilink8-wlan/build-utilites/blobs/internal/patches/kernel\\_patches/am65xx-kernel/0003-arm64-dts-am654-base-board-add-wl18xx-support.patch](https://git.ti.com/wilink8-wlan/build-utilites/blobs/internal/patches/kernel_patches/am65xx-kernel/0003-arm64-dts-am654-base-board-add-wl18xx-support.patch)

### 4.3 Update the Kernel Configuration File (defconfig)

Update the kernel configuration file for enabling “Fixed voltage regulator” used for power control of the WL18xx device from the SDIO driver.

Add the following switches to the defconfig file:

```
CONFIG_REGULATOR=y
CONFIG_REGULATOR_FIXED_VOLTAGE=y
```

### 4.4 Build the Updated Kernel, Modules, and Device Tree Files

Build the updated .dts file kernel (image) and modules using the SDK build instructions:

[http://software-dl.ti.com/processor-sdk-linux/esd/docs/05\\_01\\_00\\_11/linux/Foundational\\_Components\\_Kernel\\_Users\\_Guide.html](http://software-dl.ti.com/processor-sdk-linux/esd/docs/05_01_00_11/linux/Foundational_Components_Kernel_Users_Guide.html)

### 4.5 Update the Kernel, Modules, and Device Tree Files on the Target

After running the kernel build, update the following in the target root file system:

- /boot/Image
- /boot/lk3-am654-base-board.dtb
- Kernel modules. Because TI has changed the kernel configuration and rebuilt the modules, that modules that are under /lib/modules/4.14.67-54512-gd315a9b-dirty/ must be updated.

The following method can be used to transfer all the updated modules over tftp to the target and extract them.

On the build server:

```
$ make modules_install INSTALL_MOD_PATH=/tftpboot
INSTALL arch/arm64/crypto/aes-neon-blk.ko
INSTALL arch/arm64/crypto/aes-neon-bs.ko
INSTALL sound/soc/generic/snd-soc-simple-card-utils.ko
-
-
-
INSTALL sound/soc/generic/snd-soc-simple-card.ko
INSTALL sound/usb/snd-usb-audio.ko
INSTALL sound/usb/snd-usbmidi-lib.ko
DEPMOD 4.14.67-54512-gd315a9b-dirty
$ cd /tftpboot/
/tftpboot$ tar czf modules-4.14.67.tar.gz lib/modules/4.14.67-54512-gd315a9b-dirty/
```

On the AM65xx EVM (Assuming that the tftp server address is 192.168.100.3):

```
root@am65xx-evm:~# tftp -g -r modules-4.14.67.tar.gz 192.168.100.3
root@am65xx-evm:~# tar xf modules-4.14.67.tar.gz -C /
```

Now reboot the board and verify that the modules have been loaded using the lsmod command.

## 4.6 Add the WiLink8 Related Files into the SDK File System

Currently, the root file system provided for the AM65x EVM does not include the connectivity-related components (wpa\_supplicant, hostapd, firmware files, bluez, scripts, and so forth), so those must be added manually to the file system.

Download the following packages and save them to the tftp server folder (/tftpboot):

[https://git.ti.com/wilink8-wlan/build-utilites/blobs/internal/patches/kernel\\_patches/am65xx-kernel/am65x-evm-wl18xx-utils.tbz2](https://git.ti.com/wilink8-wlan/build-utilites/blobs/internal/patches/kernel_patches/am65xx-kernel/am65x-evm-wl18xx-utils.tbz2)

[https://git.ti.com/wilink8-wlan/build-utilites/blobs/internal/patches/kernel\\_patches/am65xx-kernel/bluez5\\_5.46-r0\\_aarch64.ipk](https://git.ti.com/wilink8-wlan/build-utilites/blobs/internal/patches/kernel_patches/am65xx-kernel/bluez5_5.46-r0_aarch64.ipk)

[https://git.ti.com/wilink8-wlan/build-utilites/blobs/internal/patches/kernel\\_patches/am65xx-kernel/bt-fw\\_8.7.1+git0+0ee619b598-r0\\_aarch64.ipk](https://git.ti.com/wilink8-wlan/build-utilites/blobs/internal/patches/kernel_patches/am65xx-kernel/bt-fw_8.7.1+git0+0ee619b598-r0_aarch64.ipk)

Use the following command sequence for using tftp on the target, to download the above packages to the target and install them into the file system.

Assuming that the tftp server address is 192.168.100.3:

```
root@am65xx-evm:~#
root@am65xx-evm:~# tftp -g -r am65x-evm-wl18xx-utils.tbz2 192.168.100.3
root@am65xx-evm:~# tar xf am65x-evm-wl18xx-utils.tbz2 -C /
root@am65xx-evm:~# tftp -g -r bluez5_5.46-r0_aarch64.ipk 192.168.100.3
root@am65xx-evm:~# opkg install bluez5_5.46-r0_aarch64.ipk
root@am65xx-evm:~# tftp -g -r bt-fw_8.7.1+git0+0ee619b598-r0_aarch64.ipk 192.168.100.3
root@am65xx-evm:~# opkg install bt-fw_8.7.1+git0+0ee619b598-r0_aarch64.ipk
```

At this point everything should be set up.

## 4.7 Boot the Updated Platform and Test

Power off the EVM, plug the WiLink8 adapter board back into the SD card slot, and boot the board.

If everything was set up correctly, you should see WL18xx modules detected on boot. You can try scanning for both Wi-Fi® and Bluetooth® as below:

```
root@am65xx-evm:~# iw wlan0 scan | grep SSID
    SSID: net4guest
    SSID: halekoa75
    SSID: externalhotspot84
    SSID: cpn84
root@am65xx-evm:~#
root@am65xx-evm:~# hciconfig hci0 up
root@am65xx-evm:~# hcitool scan
Scanning ...
    00:0A:9B:30:1D:71          n/a
    90:3A:E6:51:71:D5          FC6000P-PF815115AD8I001869
```

## IMPORTANT NOTICE AND DISCLAIMER

TI PROVIDES TECHNICAL AND RELIABILITY DATA (INCLUDING DATASHEETS), DESIGN RESOURCES (INCLUDING REFERENCE DESIGNS), APPLICATION OR OTHER DESIGN ADVICE, WEB TOOLS, SAFETY INFORMATION, AND OTHER RESOURCES "AS IS" AND WITH ALL FAULTS, AND DISCLAIMS ALL WARRANTIES, EXPRESS AND IMPLIED, INCLUDING WITHOUT LIMITATION ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT OF THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

These resources are intended for skilled developers designing with TI products. You are solely responsible for (1) selecting the appropriate TI products for your application, (2) designing, validating and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, or other requirements. These resources are subject to change without notice. TI grants you permission to use these resources only for development of an application that uses the TI products described in the resource. Other reproduction and display of these resources is prohibited. No license is granted to any other TI intellectual property right or to any third party intellectual property right. TI disclaims responsibility for, and you will fully indemnify TI and its representatives against, any claims, damages, costs, losses, and liabilities arising out of your use of these resources.

TI's products are provided subject to TI's Terms of Sale ([www.ti.com/legal/termsofsale.html](http://www.ti.com/legal/termsofsale.html)) or other applicable terms available either on [ti.com](http://ti.com) or provided in conjunction with such TI products. TI's provision of these resources does not expand or otherwise alter TI's applicable warranties or warranty disclaimers for TI products.

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265  
Copyright © 2019, Texas Instruments Incorporated