AIW FAQ

Porting AIW-169 WiFi driver for QCS6490 Yocto Linux OS			
Category	AIW	**Date**	2025/11/10
Keyword	AIW-169BR-GX1, WiFi, Yocto, Qualcomm QCS6490, EPC-R2860		
HW model	EPC-R2860 (Qualcomm QCS6490)		
OS version	Yocto 4.0.22 (LE 1.3)(Linux kernel version: 6.6.52) BSP version: aom2721a1_yl01301_k0606052_q6490_08g		

The following steps show how to build AIW-169 (Realtek RTL8852CE) WiFi driver for Yocto Linux OS. It creates a BitBake recipe, copies and extracts the driver source code to compile. After the build, the generated "8852ce.ko" file can be found in the specific path. You can have more detail about compiling environment setup from the MEMO_QCS6490 Yocto Build Process manual.

Command and Response	Description
$ cd /home/adv/BSP/layers/meta-advantech/recipes-kernel/linux $ touch aiw-169.bb	Create a Yocto bitbake recipes file (.bb file)
Add the following content to the file "aiw-169.bb": *SUMMARY = "Example of how to build an external Linux kernel module"* *DESCRIPTION = "${SUMMARY}"* *LICENSE = "GPL-2.0-only"* *LIC_FILES_CHKSUM = "file://${COREBASE}/meta/files/common-licenses/GPL-2.0-only;md5=801f80980d171dd6425610833a22dbe6"* *inherit module* *SRC_URI = "file://rtl8852CE"* *S = "${WORKDIR}/rtl8852CE"* *# The inherit of module.bbclass will automatically name module packages with* *# "kernel-module-" prefix as required by the oe-core build environment.* *EXTRA_OEMAKE += "KSRC=${STAGING_KERNEL_DIR}"*	
You can download the driver from : rtl8852CE_WiFi_linux_v1.19.4.4-0_for_EPC-R2860_QCS649.tar.gz	Download the driver

$ cd /home/adv/BSP/layers/meta-advantech/recipes-kernel/linux/files
$ cp /mnt/rtl8852CE_WiFi_linux_v1.19.4.4-0_for_EPC-R2860_QCS649.tar.gz ./
$ tar -xzvf rtl8852CE_WiFi_linux_v1.19.4.4-0_for_EPC-R2860_QCS649.tar.gz
$ mv rtl8852CE_WiFi_linux_v1.19.4.4-0_for_EPC-R2860_QCS649 rtl8852CE
	Copy the driver source code to the path and decompress it.
	Change the directory name to "rtl8852CE" which is same as aiw-169.bb defined.

$ cd /home/adv/BSP
$ source scripts/env.sh
$ unset DEBUG_BUILD
$ MACHINE=qcs6490aom2721a1 DISTRO=qcom-robotics-ros2-humble QCOM_SELECTED_BSP=custom source setup-robotics-environment
	source the environment for using "bitbake" to compile

$ bitbake aiw-169	Compile the driver
$ ls /home/adv/BSP/build-qcom-robotics-ros2-humble/tmp-glibc/sysroots-components/qcs6490_rb3gen2_vision_kit/aiw-169/usr/lib/modules/6.6.52-debug-ga97804e47e26/updates/8852ce.ko	After compiling the driver, you can check whether 8852ce.ko exists under the path.

After building the driver, the following steps guide you how to copy the compiled driver file to the specific directory and modify "qcom-robotics-full-image.bbappend" file to install the driver into the root file system.

Command and Response	Description
$ cd /home/adv/BSP $ cp /home/adv/BSP/build-qcom-robotics-ros2-humble/tmp-glibc/sysroots-components/qcs6490_rb3gen2_vision_kit/aiw-169/usr/lib/modules/6.6.52-debug-ga97804e47e26/updates/8852ce.ko /home/adv/BSP/layers/meta-advantech/recipes-products/images/files/rootfs/etc	Copy the 8852ce.ko driver to the path so that it will be packaged into the root file system.
$ vi /home/adv/BSP/layers/meta-advantech/recipes-products/images/qcom-robotics-full-image.bbappend Add the following content into the function “do_adv_post_install()”: *do_adv_post_install() {* *install -m 0755 ${ADDON_FILES_DIR}/rootfs/etc/8852ce.ko ${IMAGE_ROOTFS}/etc* *}*	Modify the “qcom-robotics-full-image.bbappend” file to include the driver in the root file system of the image.
$ scripts/build_release.sh -yocto -qcs6490aom2721a1 -user	Build the Yocto image

After flashing the Yocto image to the EPC-R2860, it needs to disable the NetworkManager service and reboot then load the driver manually. Here is an example to show you how to connect a WiFi AP to establish

Command and Response	Description
sh-5.1# systemctl disable NetworkManager sh-5.1# reboot	Disable the NetworkManager service to avoid conflicts, otherwise, it will compete with wpa_supplicant. A reboot is required for the change to take effect.
sh-5.1# cd /etc sh-5.1# insmod 8852ce.ko rtw_country_code=US sh-5.1# ifconfig -a wlan0 Link encape: Ethernet HWaddr 00:0E:8E:C5:41:19 UP BROADCAST MULTICAST MTU:1500 Metric:1 RX packets:0 errors:0 dropped:0 overruns:0 frame:0 TX packets:0 errors:0 dropped:0 overruns:0 carrier:0 collisions:0 txqueuelen:1000 RX bytes:0 (0.0 B) TX bytes:0 (0.0 B) sh-5.1# iw reg set US	Load the 8852ce.ko driver with the “rtw_country_code” set to US. After loading the driver, use “ifconfig” to check whether wlan0 interface is available. Using “iw” command to set the wireless regulatory domain to “US”.
sh-5.1# vi /etc/wpa_supplicant.conf *ctrl_interface=/var/run/wpa_supplicant* *update_config=1* *ap_scan=1* *fast_reauth=1* *network={* *ssid="XXXX"* *psk="XXXXX"* *key_mgmt=WPA-PSK* *ieee80211w=1* *}*	Create and edit the wpa_supplicant.conf file to configure WiFi connection parameters such like WiFi AP SSID and password.
sh-5.1# wpa_supplicant -D nl80211 -i wlan0 -c /etc/wpa_supplicant.conf	Establisht a WiFi connection
sh-5.1# ifconfig -a wlan0 Link encape: Ethernet HWaddr 00:0E:8E:C5:41:19 inet addr:192.168.1.170 Bcast:192.168.1.255 Mask:255.255.255.0 inet6 addr: fe80:af88:76b5:ca1b:148c/64 Scope:Link UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1 RX packets:52 errors:0 dropped:0 overruns:0 frame:0	Then you should be able to see that “wlan0” interface has been assigned an IP.

```
TX packets:37 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:15176 (14.8 KiB) TX bytes:4264 (4.1 KiB)
```