http://www.candelatech.com
sales@candelatech.com
+1 360 380 1618 [PST, GMT -8]

Ath10k Candela Technologies (CT) Firmware User Guide

This documents some of the features of the Ath10k-CT Firmware and driver.

Setting specific encoding rates.

Users may wish to set specific rates for a variety of reasons, including regulatory testing. CT firmware has some additional features to give you more control. Stock firmware will support some of this as well. These commands were tested on a recent LEDE image running latest ath10k-CT driver and firmware as of October 10, 2017. An additional patch was added to the LEDE mac80211 backports package to enable setting a single rate without generating an error return value.

Enable/disable specific bandwidths for TX frames.

This is a CT specific feature using the 'ct-special' API. This only works with 10.1 (wave-1) CT firmware currently. If you want to use this option, configure the bandwidth first, and then set a rate. **The bandwidth constraint will not take effect until you set the rate using the 'iw' commands below**:

```
# Configure for 20Mhz only (disable 80, 40):
echo 0xE00000006 > /sys/kernel/debug/ieee80211/phy0/ath10k/ct_special
```

```
# Configure for 40Mhz only (disable 80, 20).  Please note than legacy
# rates cannot ever use 40Mhz.
echo 0xE00000005 > /sys/kernel/debug/ieee80211/phy0/ath10k/ct_special
```

```
# Configure for 80Mhz only (disable 40, 20).  Please note that HT
# and legacy rates cannot ever use 80Mhz.
echo 0xE00000003 > /sys/kernel/debug/ieee80211/phy0/ath10k/ct_special
```

```
# Configure for any available Mhz (default)
echo 0xE00000000 > /sys/kernel/debug/ieee80211/phy0/ath10k/ct_special
```

Configure a specific TX rate.

These commands probably work fine on stock firmware, and on 10.4 (wave-2) CT firmware as well.

```
# NOTE:  VHT rates are not normally available on the 2.4Ghz band without additional
```

```
#        kernel modifications.
# Set for vht-rateset, MCS 0, NSS 1:
iw dev wlan0 set bitrates legacy-5 ht-mcs-5 vht-mcs-5 1:0
# Set for vht-rateset, MCS 0, NSS 3:
iw dev wlan0 set bitrates legacy-5 ht-mcs-5 vht-mcs-5 3:0
# Set for vht-rateset, MCS 9, NSS 3:
iw dev wlan0 set bitrates legacy-5 ht-mcs-5 vht-mcs-5 3:9
# The ath10k 9880 3x3 NIC supports up to MCS 9, NSS 3.
```

```
# Set for ht-rateset, MCS 0, nss 1:
iw dev wlan0 set bitrates legacy-5 ht-mcs-5 0 vht-mcs-5
# For HT MCS 8, nss2:
iw dev wlan0 set bitrates legacy-5 ht-mcs-5 8 vht-mcs-5
#  The ath10k 9880 3x3 NIC supports ht-mcs 0-23 settings.
```

```
# Set for legacy (a/g) 6Mbps
iw dev wlan0 set bitrates legacy-5 6 ht-mcs-5 vht-mcs-5
# For legacy 54Mbps
iw dev wlan0 set bitrates legacy-5 54 ht-mcs-5 vht-mcs-5
#  Available legacy rates for the 2.4Ghz band are: 1, 2, 5.5, 11, 6, 9, 12, 18, 24, 36, 48, 54
#  Available legacy rates for the 5Ghz band are: 6, 9, 12, 18, 24, 36, 48, 54
```

```
# Set back to default rates.
iw dev wlan0 set bitrates
```

Verify tx/rx rates.

In order to verify that the rate configuration is working as expected, you can use an RF sniffer, but you may also be able to just dump the station info to see the current rateset. The TX rate should match the expected values.

```
iw dev wlan0 station dump
```

WMI firmware keepalive (10.1 wave-1 firmware only at this time)

The ath10k-CT driver and firmware support a WMI keepalive message. Once the firmware receives a keepalive message, then it will assert in the future if it does not receive a keepalive within about 8 seconds. Often this crash is better than an indefinite hang. But, some systems may not be able to reliably send keepalives often enough, so you may wish to disable the keepalive, or set it to a larger value. You can do this with the 'ct_special' debugfs file. This setting will be saved through firmware restarts, but a reboot or driver reload will initialize the setting back to defaults.

```
# Disable WMI timeout assert in the firmware:
echo 0xCFFFFFFFF > /sys/kernel/debug/ieee80211/phy0/ath10k/ct_special

# Set to 8 second timeout
echo 0xC00001F40 > /sys/kernel/debug/ieee80211/phy0/ath10k/ct_special
```

Enable getting CFR data (at least for probe-response ACKs) (9984, 9888 only it seems)

For recent wave-2 firmware for 9984 and (maybe) 9888, you may configure it to return CFR data for probe-response frames. This does not work with 4019 or 9980 chipsets it seems. The data is delivered in multiple chunks over WMI. See ath10k_wmi_event_csi_mesg() in the ath10k-ct driver.

```
# Enable CFR data reporting.
echo 0xD00000001 > /sys/kernel/debug/ieee80211/phy1/ath10k/ct_special

# Disable CFR data reporting.
echo 0xD00000000 > /sys/kernel/debug/ieee80211/phy1/ath10k/ct_special
```

Enable receiving TXBF frames in the driver (wave-2 only)

For wave-2 firmware built after Nov 7, 2017, you can enable the receipt of txbf frames when you put the radio into monitor mode and also when you just use the ct_special command to enable the txbf reporting logic. Frames that are consumed by the firmware (ones directed at local peers), will not be delivered in this way. They can only be received using the txbf_cv WMI message, which is also enabled/disabled with this same command. See ath10k_wmi_event_txbf_cv_mesg in the ath10k-ct driver for details.

```
# Enable txbf frames and txbf_cv WMI messages to be sent to the driver.
echo 0xF00000001 > /sys/kernel/debug/ieee80211/phy1/ath10k/ct_special

# Disable txbf frames and txbf_cv WMI messages from being sent to the driver (default).
echo 0xF00000000 > /sys/kernel/debug/ieee80211/phy1/ath10k/ct_special
```

Enable receiving all management frames on the host (wave-2 only)

With ath10k-ct wave-2 firmware from Nov 8, 2017 or later, you can set a flag to cause all received management frames to be sent to the host driver. The one big exception is txbf messages: They are consumed by the txbf_cv logic and so the only way you can get notification of them is by using the txbf_cv WMI message (see above). RX filters are not modifed as part of this setting.

```
# Enable receiving all mgt frames
echo 0x1000000001 > /sys/kernel/debug/ieee80211/phy1/ath10k/ct_special

# Disable receiving all mgt frames
echo 0x1000000000 > /sys/kernel/debug/ieee80211/phy1/ath10k/ct_special
```

Set the SU or MU Sounding timer in ms (wave-2 only)

With ath10k-ct driver code from Nov 7, 2017 or later, you can now set the SU and MU sounding frame timer. Minimum is 0, maximum is 500. This should work on stock firmware as well as CT firmware, but only tested on CT firmware. The default SU timer is 100ms, and the default MU timer is 40ms.

```
# Set MU sounding timer to 16ms
echo 0x100300000010 > /sys/kernel/debug/ieee80211/phy1/ath10k/ct_special

# Set SU sounding timer to 32ms
echo 0x100200000020 > /sys/kernel/debug/ieee80211/phy1/ath10k/ct_special
```

```
# Set values to 0 and restart the firmware (or just reload the entire driver),
# and values will be automatically set back to firmware defaults.
```

Force sending sounding frames (wave-2 only)

With ath10k-ct driver and firmware code from Nov 16, 2017 or later, you can now use the ct_special API to make generic 'fwtest' configuration commands. And, one of them can be used to force sending a sounding frame to a specific peer. The API is a bit complicated: First, the 'ID' will have '0xFF0000' set as a mask to identify this as being a 'fwtest' command. The low bits of the ID will be the fwtest ID. The low 32 bits of the ct_special command are the value to set. To send a sounding frame, you need ID 74, and the value is the 'aid' of the peer. The driver should print the aid when a station associates, and there are other ways to find it as well. If you have a single station, then aid == 1. In addition, the ID=74 command in stock firmware will not actually send an sounding frame. In order to be somewhat backwards compatible, you have to set bit 0x80000000 in the value to get the new behaviour. So, to force a sounding frame to the first station, you can use this command:

```
# Force sounding frame to station 1.
echo 0xFF004A80000001 > /sys/kernel/debug/ieee80211/hy1/ath10k/ct_special
```

While testing this, I noticed that the TX logic in the NIC/Firmware would hang if I ran this command in a fast loop. So, the firmwre now has protection where it will ignore any of these commands that are closer than 25ms apart. Possibly it will still be unstable in long runs even with 25ms spacing...time and testing will tell.