WL Tool for Embedded 802.11 Systems CYW43xx Technical Information

Doc. No.: 002-23156 Rev. **

February 27, 2018

Cypress Semiconductor
198 Champion Court
San Jose, CA 95134-1709
www.cypress.com

Copyrights

Contents

1. Introduction

This document describes a subset of the commands available in `wl`, the Cypress WLAN client utility. It is intended for developers who are evaluating and/or testing the Cypress CYW43xx combo and embedded Wi-Fi chip solutions.

1.1 System Requirements

The hardware and software requirements are:

- The latest Cypress CYW43xx hardware.
- A Linux operating system platform.
- The latest version of the Cypress dongle driver.

1.2 Command Syntax

The syntax is as follows:

```
wl <adapter> [-h] [-d|u|x] <command> [arguments]
```

where

```
-h this message
-d signed integer
-u unsigned integer
-x hexadecimal
```

The [h,u] option is only to print help.

Other syntax specifics are as follows:

- Entries within square brackets, such as [arguments], are optional. In the above example, switches within brackets, such as –h, are typed as shown. The | symbol should not be typed, it represents the word *or*.
- Entries within angle brackets, such as <adapter>, are required and indicate that a value must be inserted in place of the item contained within the angle brackets.
- Entries shown outside of either square or angle brackets are to be typed as shown.

1.3 Options

Type `wl` at the command prompt to view the full set of available `wl` options.

Each of the remaining sections is a set of logically grouped `wl` commands. Each of the commands and their associated options are described in sufficient detail to explain command usage.

Note: The `wl` command options described in this document represent the subset of options that are most useful to those embedding CYW43xx hardware into their designs.

1.4 Document Conventions

The following conventions may be used in this document:

Convention	Description
Bold	User input and actions: type **exit**, click **OK**, press **Alt+C**
Monospace	Code: `#include <iostream>` HTML: `<td rowspan = 3>` Command line commands and parameters: `wl [-l] <command>`
< >	Placeholders for required elements: enter your <username> or `wl <command>`
[ ]	Indicates optional command-line parameters: `wl [-l]` Indicates bit and byte ranges (inclusive): [0:3] or [7:0]

1.5 Technical Support

Cypress provides customer access to a wide range of information, including technical documentation, schematic diagrams, product bill of materials, PCB layout information, and software updates through its customer support portal.

For assistance, visit Cypress Support or contact customer support at +1(800) 541-4736 Ext. 2 (in the USA) or +1 (408) 943-2600 Ext. 2 (International).

You can also use the following support resources if you need quick assistance:

- Self-help (Technical Documents)
- Local Sales Office Locations

2. Command List and Version

2.1 cmds

Generates a list of available commands.

```
wl cmds
```

Returns

All commands available to the attached 43XX chip.

2.2 mpc

Gets or sets the minimum power control mode.

```
wl mpc [<enable>]
```

Parameters

enable

Value	Description
0	Disable the minimum power control mode: Required before WLAN RF manufacturing (MFG) testing.
1	Enable the minimum power control mode. This is the default mode for nonmanufacturing firmware and most MFG firmware.

Returns

The current MPC status.

2.3 ver

Gets `wl` version information. If the -v option is used, then the version string from an NVRAM variable is also provided.

```
wl ver [-v]
```

Returns

Version number, for example, 4.10 RC47.0.

3. Initialization and Restart

3.1 down

Resets the interface and indicates that it is down (that is, disabled or nonoperational).

```
wl down
```

Some of the tasks performed include:

- Disassociation.
- Turning the radio off.
- Canceling the watchdog timer.
- Canceling the activity timer.
- Canceling active scans.
- Canceling IBSS timers.
- Canceling association timers.
- Flushing the TX control queue.
- Reclaiming SCBs (stations).
- For an AP, flushing PS-POLL response (MAC Segment Data Unit [MSDU]) packet queues and PSPOLL.
- Response (MAC Protocol Data Unit [MPDU]) packet queues.
- Restoring to a known good default state.

3.2 download

Downloads the firmware and NVRAM files over a serial interface to a device attached to a remote host. The NVRAM text file is not optional.

```
wl download <binaryfile> <nvramfile>
```

Parameters

binaryfile

Name of the firmware file to be downloaded.

nvramfile

Name of the NVRAM file to be downloaded.

3.3 isup

Gets the operational state of the driver.

```
wl isup
```

Returns

Driver state:

Value	Description
0	Down
1	Up

3.4 out

Indicates that the interface is down (that is, disabled or nonoperational) without resetting the interface.

```
wl out
```

On dual-band cards, cards must be band-locked before using this command.

3.5 up

Initializes the interface and indicates that it is up (that is, operational).

```
wl up
```

The tasks performed during initialization include:

- Reading the PHY revision.
- Setting the soft interrupt mask.
- Bringing the interface up in each band.
- Initializing the default rate, channel, and type-dependent information.
- Initializing the basic rate lookup.
- Saving, suspending, disabling interrupts, and turning the radio off.
- Starting a one-second watchdog.
- Starting the activity LED timer.

4. Debugging and Status

4.1 counters

Gets the driver counter values.

```
wl counters
```

Returns

All driver counter values:

Value	Description
txframe	Transmit frames from the driver. (TX frames from the MAC minus all control and management frames from the MAC.)
txbyte	The number of data bytes sent by the MAC.
txretrans	The number of retries by the MAC. (If an ACK is not received, the packet is sent again.).
txerror	The TX frames that were not sent, including packets that exceeded the retry limit.
rxframe	The number of data frames received by the MAC.
rxbyte	The number of data bytes received by the MAC.
rxerror	RX data errors.
txprshort	TX short preamble frames.
txdmawar	Occurrences of the PR15420 workaround.
txnobuf	TX out-of-buffers error.
txnoassoc	The number of frames that are tossed by the driver when we are not associated.
txserr	The number status returns that contain inconsistencies. For example, an indication that an ACK was received for a frame when an ACK was not expected.

Value	Description
txphyerr	TX PHY errors (indicated in TX status). TX status is returned by the MAC to the driver:
	0 corerevs <5: ACK was received for this frame. corerevs >= 5: TxStatus valid bit.
	1 RS corerevs <5: RSV. RS corerevs >= 5: ACK was received for this frame.
	4:2 SU suppress indication bits: 0 = Not suppressed. 1 = Suppressed because of the Priority Management Queue (PMQ) entry. 2 = Suppressed because of a flush request. 3 = Suppressed because of maximum retries of the previous fragment (only valid for fragmented MSDUs). 4 = Suppressed because of a channel mismatch. 5 = Suppressed because of lifetime. 6 = Suppressed because of underflow. 7 = Suppressed because of a NACK (valid in AB mode only).
	5 A-MPDU indication (AM): This TX-status package corresponds to an A-MPDU. This bit will be set for both TX-status packages returned to the host for A-MPDUs.
	6 Intermediate status (IM) indication (fragment failed after successful RTS/CTS).
	7 PM mode was indicated to the AP.
	11:8 RTS transmissions (RT).
	15:12 Fragment transmissions (FT).
txphycrs	TX carrier-sense counter for the PHY.
txfail	The MAC tried to transmit a packet. The retry limit was exceeded and this counter was updated.
d11_txfrag	Frames correctly ACKed (according to the driver¡Xshould be the same as d11_txfrmsnt if there is no fragmentation).
d11_txmulti	Multicast frames sent by the MAC.
d11_txretry	Times a frame was retried once.
d11_txretrie	Times a frame was retried more than once.
d11_txrts	RTS sent by the MAC.
d11_txnocts	CTS sent by the MAC.
d11_txnoack	ACKs sent by the MAC.
d11_txfrmsnt	Packets sent for which an ACK is received.
rxcrc	Packets received with CRC errors in the payload.
rxnobuf	RX out-of-buffers errors.
rxnondata	RX non-data frames in the data channel (errors).
rxbadds	RX bad DS errors: RX bad control or management frames.
rxdup	dot11 frame duplicates.
rxfragerr	MAC did not receive all the fragments of a fragmented packet.
rxrunt	The header indicates that the length of the packet is zero.
rxgiant	The header indicates that the length of packet is huge >4 KB
rxnoscb	RX no SCB error.
rxbadproto	RX invalid frames.

Value	Description
d11_rxfrag	Fragments received.
d11_rxmulti	Multicast packets received.
d11_rxundec	IEEE 802.11 WEP unable to decrypt.
txallfrm	Frames sent by ƒÝCODE, including data, ACK, RTS, CTS, and control management (includes retransmissions).
txackfrm	ACK frames sent.
txphyerr	TX PHY errors (indicated in TX status).
rxundec	IEEE 802.11 WEP unable to decrypt.
rxfrmtoolong	The received frame was longer than legal limit (2346 bytes).
rxfrmtooshrt	The received frame did not contain enough bytes for its frame type.
rxinvmachdr	Either the protocol version != 0 or the frame type is not data, control, or management.
rxbadfcs	Frames for which the CRC check failed in the MAC.
rxbadplcp	The parity check of the PLCP header failed.
rxcrsglitch	The PHY was able to correlate the preamble but not the header.
rxstrt	Frames received with a good PLCP (i.e., passing parity check).
rxdfrmucastmbss	Data frames received with good FCS and matching receiver address (RA).
rxmfrmucastmbss	Management frames received with good FCS and matching RA.
rxcfrmucast	Control frames received with good FCS and matching RA.
rxrtsucast	Unicast RTS addressed to the MAC (good FCS).
rxctsucast	Unicast CTS addressed to the MAC (good FCS).
rxackucast	Unicast ACKs received (good FCS).
rxdfrmocast	Data frames received with good FCS and no matching RA.
rxmfrmocast	Management frames received with good FCS and no matching RA.
rxcfrmocast	Control frames received with good FCS and no matching RA.
rxrtsocast	Multicast RTS addressed to the MAC.
rxctsocast	Multicast CTS addressed to the MAC.
rxdfrmmcast	RX data multicast frames received by the MAC.
rxmfrmmcast	The number of RX management multicast frames received by the MAC.
rxcfrmmcast	RX control multicast frames received by the MAC (unlikely to see these).
rxbeaconmbss	Beacons received from members of the BSS.
rxdfrmucastobss	Unicast frames addressed to the MAC from another BSS (WDS FRAME).
rxbeaconobss	Beacons received from another BSS.
rxrsptmout	Response timeouts for transmitted frames expecting a response.
bcntxcancl	Transmit beacons cancelled due to receipt of a beacon (IBSS).
rxf0ovfl	Receive FIFO 0 overflows.
rxf1ovfl	Receive FIFO 1 overflows (obsolete).
rxf2ovfl	Receive FIFO 2 overflows (obsolete).
txsfovfl	Transmit status FIFO overflows (obsolete).
pmqovfl	PMQ overflows.
rxcgprqfrm	Received probe requests that made it into the Probe Request Queue (PRQ) FIFO.
rxcgprsqovfl	RX probe request queue overflows in the AP.

Value	Description
txcgprsfail	TX probe response failures: AP sends a probe response but does not receive an ACK.
txcgprssuc	TX probe response success (ACK was received).
prs_timeout	Probe response timeout.
rxnack	NACK received counter.
frmscons	–
txnack tx	Side NACK counter.
txglitch_nack	–
txburst	BRCM-to-BRCM only burst packets.
txphyerror	Error counter for things such as CCK packet in A band, etc.
txchanrej	Frames dropped due to mismatched channel (usually probe requests that never went out before the channel was switched).
rxXmbps	Counters for each successful packet received at X rate.
pketngrxducast	Packets received in packet engine mode sent to the packet-engine assigned MAC address.
pktengrxdmcast	Packets received in packet engine mode sent to the multicast address.
txmpdu_sgi	TX packets with a short guard interval.
rxmpdu_sgi	RX packets with a short guard interval.
txmpdu_stbc	TX packets with space-time block coding.
rxmpdu_stbc	TX packets with space-time block coding.

Example: Return output

```
txframe 0 txbyte 0 txretrans 0 txerror 3521 rxframe 0 rxbyte 0 rxerror 0 txprshort 0 txdmawar 0
txnobuf 3521 txnoassoc 0 txchit 0 txcmiss 0 reset 2 txserr 0 txphyerr 0 txphycrs 0 txfail 0
d11_txfrag 69098 d11_txmulti 0 d11_txretry 0 d11_txretrie 0 d11_txrts 0 d11_txnocts 0 d11_txnoack
0 d11_txfrmsnt 0 rxrtsocast 986289 rxctsocast 1018345 rxdfrmmcast 37898 rxmfrmmcast 1466381¡K.
```

4.2 cur_etheraddr

Gets or sets the Medium Access Controller (MAC) address. In a set, the new address will override the current MAC address.

```
wl cur_etheraddr [<mac address>]
```

Parameters

mac address

New MAC address with format `xx:xx:xx:xx:xx:xx`.

Returns

The current MAC address from a get operation.

4.3 dump

Gets the driver state and chip registers and prints them to standard output.

```
wl dump
```

Returns

Driver state and chip register data.

Example: Return output

```
wl0: Dec 16 2005 18:52:10 version 4.10.47.3
resets 1
perm_etheraddr 00:90:4b:7a:7a:ac cur_etheraddr 00:90:4b:7a:7a:ac
board 0x1, board rev 4.5
rate_override: A 0, B 0
antdiv_override -1 (3 default) txant 3
BSS Config 0:
"\x1F\x19<O\x8E\x98\x96\x9A\x80 \x9FS.\x10\x85z\x9B\x97\x95¡P\
x00\x19\x7F(j\x96"
enable 0 up 0 wlif 0x00000000 ""
wsec 0x1 auth 0 wsec_index -1 wep_algo 0
current_bss.BSSID 00:00:00:00:00:00
current_bss.SSID ""
assoc_state 0 associated 0
```

4.4 event_msgs

Gets or sets the 128-bit hexadecimal filter bit mask for MAC events.

```
wl event_msgs [<value>]
```

Parameters

value

0x0 (default) to 0xffffffffffffffffffffffffffffffff (maximum)

Returns

The 128-bit hexadecimal filter bit mask.

4.5 msglevel

Sets the driver console debugging message bit vector.

```
wl msglevel N
```

Parameters

N

Value	Description
0x0001	error, err
0x0002	trace
0x0004	prhdrs
0x0008	prpkt

4.6 nvram_dump

Gets the NVRAM content.

```
wl nvram_dump
```

Returns

The NVRAM content.

Example: Return output

```
manfid=0x2d0prodid=0x4329manf=productname=sromrev=3boardtype=0x504boardrev=0x11boardflags=0x120
0devid=0x4330xtalfreq=37400aa2g=3aa5g=3ag0=255pa0b0=6003pa0b1=64086pa0b2=65195pa0itssit=62pa0ma
xpwr=60opo=0mcs2gpo=0x22222222pa1lob0=7436pa1lob1=63828pa1lob2=323pa1b0=7834pa1b1=63935pa1b2=60
2pa1hib0=6364pa1hib1=63469pa1hib2=65385pa1itssit=62pa1maxpwr=60opo=0mcs5gpo=0x22222222rssismf2g
=0xarssismc2g=0xbrssisav2g=0x3bxa2g=0rssismf5g=0xarssismc5g=0xarssisav5g=0x2bxa2g=0ccode=ALLcct
l=0x0rxpo2g=255boardnum=2048macaddr=00:90:4c:c5:34:23nocrc=1
```

4.7 pktcnt

Gets a summary of good and bad packets.

```
wl pktcnt
```

Returns

Standard output with the following form:

- Receive: good packet 0, bad packet 0.
- Transmit: good packet 0, bad packet 0.

4.8 reset_cnts

Resets the counter information in the driver.

```
wl reset_cnts
```

4.9 revinfo

Gets the hardware revision information.

```
wl revinfo
```

Returns

The hardware revision information.

Example Output:

Value	Description
0x14e4	vendorid
0x4320	deviceid
0x22050000	radiorev
0x4306	chipnum
0x2	chiprev
0x4	corerev
0x1	boardid
0x1028	boardvendor

Value	Description
0x45	boardrev
0x40a2f03	driverrev
0x13f0066	ucoderev
0x1	bus

4.10 status

Gets information about the current network association.

```
wl status
```

Returns

Network association information including SSID, Received Signal Strength Indicator (RSSI), noise, channel, BSSID, capability, and supported rates.

If there is no network association, this command will return the following:

Not associated, last associated with SSID: "", or the previously associated SSID may be shown, but with a BSSID of 00:00:00:00:00:00.

5. Configuration

5.1 dtim

Get or set the Delivery Traffic Indication Message (DTIM) interval.

```
wl dtim [<interval>]
```

Parameters

interval

An integer number of beacon intervals. The default is 3.

Returns

The DTIM interval.

5.2 fast_timer

Gets or sets the PHY watchdog timer for the most frequent tasks.

```
wl fast_timer [<time_val>]
```

Parameters

time_val

A 32-bit timer value in microseconds.

Returns

The PHY watchdog timer for the most frequent tasks.

5.3 frameburst

Gets or sets the Frameburst mode.

```
wl frameburst [on|off]
```

Parameters

on

Frameburst enabled

off

Frameburst disabled (default)

Returns

The Frameburst mode.

5.4 glacial_timer

Gets or sets the Local Oscillator LOFT recalibration periodicity timer.

```
wl glacial_timer [<time_val>]
```

Parameters

time_val

A 32-bit timer value in microseconds

Returns

The PHY watchdog timer for the least frequent tasks.

5.5 infra

Gets or sets the infrastructure mode.

```
wl infra [<value>]
```

Parameters

value

Infrastructure mode:

Value	Description
0	IBSS (ad hoc)
1	Infrastructure BSS
2	Either IBSS or infrastructure BSS

Returns

The infrastructure mode

Note: When frameburstsing is enabled, multiple frames are sent with a minimum interframe gap to enhance network efficiency and reduce overhead. WindowsR STA drivers contain the capability to perform framebursting.

5.6 lrl

Gets or sets the retry limit of frames longer than the RTS threshold.

```
wl lrl <limit>
```

Parameters

limit

1–255

Returns

The retry limit of frames longer than the RTS threshold

If *limit* is reduced, frames are discarded more quickly, making the buffer space requirement lower. If increased, retransmitting up to the limit takes longer and may cause TCP to throttle back on the data rate.

5.7 plcphdr

Gets or sets the Physical Layer Convergence Protocol (PLCP) preamble type.

```
wl plcphdr [<value>]
```

Parameters

value

Value	**Description**
–1	Automatic: disable the short preamble capability advertisement and never initiate a short preamble frame exchange.
0	Short: enable the short preamble capability.
1	Long: enable the long preamble capability.

Returns

The PLCP preamble type

5.8 PM

In the Cypress WLAN driver, FAST PS mode consumes slightly more power than the standard PS mode, but can operate at higher rates (close to 54gR rates). The difference between FAST PS and PS is dependent on the usage model.

- If there is no traffic, there is no difference.
- If the traffic is a steady stream (some packets every DTIM), there is no difference.
- If the traffic is a stream that happens less often than every DTIM, then the adapter consumes more power in FAST PS mode and less power but with more latency in PS mode. The power difference depends on the frequency of traffic.

```
wl PM [<value>]
```

Parameters

value

Value	**Description**
0	CAM (constantly awake)
1	PS (power-save)
2	FAST PS mode

Returns

The power management mode

5.9 promisc

Enables or disables promiscuous mode operation.

```
wl promisc <value>
```

Parameters

value

Value	Description
0	Disable
1	Enable

5.10 rifs

Enables or disables Reduced Inter-Frame Spacing (RIFS).

```
wl rifs [<value>]
```

Parameters

value

Value	Description
0	Disable
1	Enable

Returns

On/Off of the RIFS setting

5.11 rtsthresh

Gets or sets the RTS threshold value.

```
wl rtsthresh [<value>]
```

Parameters

value

Value	Description
2347 (0x92b)	Default
0 to4000 (0xfa0)	Range

Returns

The RTS threshold

5.12 slow_timer

Gets or sets the PHY watchdog timer for less frequent tasks.

```
wl slow_timer [<time_val>]
```

Parameters

time_val The timer value in microseconds

Returns

None

5.13 srl

Gets or sets the retry limit of frames shorter than the RTS threshold.

```
wl srl <limit>
```

Parameters

limit

1–255

Returns

The retry limit of frames shorter than the RTS threshold

If `limit` is reduced, frames are discarded more quickly, making the buffer space requirement lower. If increased, retransmitting up to the limit takes longer and may cause TCP to throttle back on the data rate.

6. Antenna Controls

6.1 antdiv

Gets or sets the antenna diversity protocol to use during signal reception.

```
wl antdiv <value>
```

Parameters

value

Value	Description
0	Force use of antenna 1.
1	Force use of antenna 2.
3	Automatic selection of antenna diversity.

Returns

The antenna diversity protocol during reception

Antenna diversity enhances receiver performance by overcoming multipath effects that are a function of antenna location, radiation pattern, and polarization. Providing antenna diversity increases the likelihood of successfully receiving a signal when one antenna is experiencing a deep or flat multipath-induced fade.

6.2 txant

Gets or sets the antenna to use during signal transmission.

```
wl txant <value>
```

Parameters

value

The antenna to use during signal transmission:

Value	Description
0	Force use of antenna 1.
1	Force use of antenna 2.
3	Use the receive antenna selection that was in force during the most recently received good PLCP header.

Returns

The antenna usage during transmission

7. 802.11n-Specific

7.1 ampdu

When set, sets both AMPDU TX and RX aggregation.

When get, returns the AMPDU TX state with the assumption that AMPDU RX has to be supported by default. Can only be modified when the driver is down.

```
wl ampdu [<enable>]
```

Parameters

enable

Value	Description
1	Enabled
2	Disabled

7.2 ampdu_tx

When set, sets AMPDU TX aggregation.

When get, returns the AMPDU TX state. Can only be modified when the driver is down.

```
wl ampdu_tx [<enable>]
```

Parameters

enable

Value	Description
1	Enabled
2	Disabled

7.3 ampdu_rx

When set, sets AMPDU RX aggregation.

When get, returns the AMPDU RX state. Can only be modified when the driver is down.

```
wl ampdu_rx [<enable>]
```

Parameters

enable

Value	Description
1	Enabled
2	Disabled

7.4 ampdu_mpdu

Gets or sets the maximum number of MPDU in the AMPDU mode.

```
wl ampdu_mpdu [<unit>]
```

Parameters

unit

Value	Description
15	Default
1	Enabled
2	Disabled

7.5 mimo_bw_cap

Sets or clears the supported channel bandwidth in the high throughput (HT) capability information field.

```
wl mimo_bw_cap [<capability>]
```

Parameters

capability

Value	Description
0	20 MHz in both bands
1	40 MHz in both bands
2	20 MHz in the 2.4 GHz band and 40 MHz in the 5 GHz band

7.6 mimo_txbw

Gets or sets the MIMO frame TX bandwidth.

```
wl mimo_txbw [<type>]
```

Parameters

type

Value	Description
2	20 MHz
3	20 MHz upper
4	40 MHz
5	40 MHz dup

7.7 mimo_ps

Gets or sets the MIMO power save mode.

```
wl mimo_ps [<mode>]
```

Parameters

mode

Value	Description
0	MIMO PS (MIMO disallowed)
1	Proceed MIMO with an RTS frame
2	Not applicable
3	MIMO PS (no restriction)

Returns

MIMO power save mode

7.8 nmode

Gets or sets the MIMO 802.11n mode. The driver must be down before changing nmode,that is, `wl down; wl nmode 0; wl up`.

```
wl nmode [<arg>]
```

Parameters

arg

Value	Description
0	N mode disable
1	N mode enable

Returns

The 802.11n mode from a get operation

8. 802.11ac-Specific

8.1 bw_cap

Gets or sets the bandwidth capabilities per band.

```
wl bw_cap [band][cap]
```

Parameters

band

Value	Description
2g	2.4 GHz band.
5g	5 GHz band.

cap

Value	Description
0x1	20 MHz
0x3	20 MHz and 40 MHz
0x7	20 MHz, 40 MHz, and 80 MHz
0xff	Unrestricted bandwidth

Returns

The bandwidth caps per band.

8.2 interference_override

Set or get the interference mitigation mode.

```
wl interference_override [<arg>]
```

Parameters

arg

Value	Description
Non-ACPHY	
–1	Remove override. Override disabled.
0	None.
1	Non-WLAN.
2	WLAN manual.
3	WLAN automatic.
4	WLAN automatic with noise reduction.
ACPHY	
–1	Remove override. Override disabled.
0	None.
1	Desense based on glitches.
2	Limit the packet gain based on hardware ACI.
3	Desense based on glitches and limit packet gain based on hardware ACI.

Returns

The currently selected interference mitigation mode.

8.3 mimo_preamble

Sets or gets the MIMO preamble.

```
wl mimo_preamble [<arg>]
```

Parameters

arg

Value	Description
–1	Auto
0	Mixed mode
1	Greenfield mode

Returns

The currently selected MIMO preamble mode.

8.4 rxchain

Allows the selective use of RX chains on MIMO devices.

```
wl rxchain [<arg>]
```

Parameters

arg

Value	Description
1	Chain 0 only
2	Chain 1 only
3	Chains 0 and 1

Returns

The RX chain currently in use.

8.5 tempsense_disable

Disables the periodic tempsense feature.

```
wl tempsense_disable
```

Parameters

None

Returns

None

8.6 txchain

Allows the selective use of TX chains on MIMO devices.

```
wl txchain [<arg>]
```

Parameters

arg

Value	Description
1	Chain 0 only
2	Chain 1 only
3	Chains 0 and 1

Returns

The TX chain currently in use.

8.7 txcore

Allows users to select a PHY core mapping to spatial streams.

```
wl txcore [-k <CCKcoreMask>] [-o <OFDMcoreMask>] [[-s <numStreams>] [-c <coreBitmap>]]
```

Options and Parameters

-k, CCKcoreMask

Used to set the CCK core mask (CCKcoreMask).

-o, OFDMcoreMask

Used to set the OFDM core mask (OFDMcoreMask).

-s, numStreams

Used to set the number of space-time streams (numStreams). The number of streams can range from 1 to 4.

-c, coreBitmap

Indicates the active core bitmask (coreBitmap) to use when transmitting a frame.

Returns

The PHY core mapping.

Examples:

```
wl txcore -k 0x1 (force the legacy IEEE 802.11b CCK rate to use core 0)
wl txcore -o 0x2 (force legacy IEEE 802.11a/g OFDM rates to use core 1)
wl txcore -s 1 -c 0x1 (force MCS single-stream rates to use core 0)
wl txcore -s 1 -c 0x2 (force MCS single-stream rates to use core 1)
wl txcore -s 1 -c 0x4 (force MCS single-stream rates to use core 2)
wl txcore -s 2 -c 0x3 (force MCS dual-stream rates to use core 0 and 1)
wl txcore -s 2 -c 0x5 (force MCS dual-stream rates to use core 0 and 2)
wl txcore -s 2 -c 0x6 (force MCS dual-stream rates to use core 1 and 2)
wl txcore -s 1 -c 0x7 (force MCS single-stream rates to be expanded to cores 0, 1, and 2)
wl txcore -s 2 -c 0x7 (force MCS dual-stream rates to be expanded to cores 0, 1, and 2)
```

8.8 vht_features

Enable Cypress Proprietary VHT features in a vector.

These are extensions to the IEEE 802.11ac Draft 3.0 specification. The extended VHT rate support enables all VHT MCS rates (MCS0–MCS9) for all bandwidths (160 MHz, 80 MHz, 40 MHz and 20 MHz) in use.

IEEE 802.11ac Draft 3.0 currently prohibits using VHT in the 2.4 GHz band and prohibits specific rates, such as MCS9 in 20 MHz with one spatial stream (nss1), MCS9 in 20 MHz with two spatial streams (nss2), and MCS9 in 80 MHz with three spatial streams (nss3). This is a permission bitmap, the final capability will be negotiated between the device and its communicating peer.

```
wl vht_features [<vector>]
```

Parameters

vector

Bit	Description
0	Enable VHT operation in the 2.4 GHz band.
1	Enable extended VHT rate support.
2–7	Reserved

Returns

A bit vector in the same form as the above vector parameter.

9. PHY, Radio, and PA Commands

9.1 phytype

Gets the PHY type.

```
wl phytype
```

Parameters

None

Returns

The PHY type as one of the following values:

Value	Description
0	PHY type A
1	PHY type B
2	PHY type G
4	PHY type N
5	PHY type LPPHY
6	PHY type SSLPNPHY

9.2 phy_activecal

Show whether forced calibrations have been completed or not. See phy_activecal on page 34.

```
wl phy_activecal
```

Parameters

None

Returns

Value	Description
0	No active calibrations.
1	There are active calibrations in process.

9.3 phy_forcecal

Force PHY calibration to run immediately.

```
wl phy_forcecal [<val>]
```

Parameters

val

Value	Description
0	Single-phase full calibration
1	Single-phase full calibration
2	Single-phase partial calibration
3	Multiphase full calibration if enabled for the module

Returns

None

9.4 phy_percal

Get or set the behavior for periodic calibration.

```
wl phy_percal [<mode>]
```

Parameters

mode

Value	Description
0	Disable.
1	Single phase calibration only.
2	Enable multiphase calibration if enabled for the module.
3	Manual (testing mode). All driver-initiated periodic calibrations are blocked, giving phy_forcecal (see phy_activecal on page 34) full control.

Returns

The current calibration mode.

9.5 phy_rssi_ant

Get the moving average RSSI value over all antennas or get only one reading for the SISO PHY.

```
wl phy_rssi_ant
```

Parameters

None

Returns

The moving average RSSI value over all antennas.

Example Output:

```
rssi[0] -47 rssi[1] -75 rssi[2] -85
```

9.6 phy_txpwrctrl

Enable or disable transmit power control.

```
wl phy_txpwrctrl [<val>]
```

Parameters

mode

Value	Description
0	Disable
1	Enable

Returns

The transmit power control settings.

9.7 phy_txpwrindex

Gets or sets the PHY TX power index.

```
wl phy_txpwrindex <index>
```

Parameters

index

From 0 to 127.

Returns

The current TX power index.

9.8 phy_watchdog

Gets or sets whether the PHY watchdog is enabled or disabled.

The PHY watchdog can trigger periodically to run a calibration test. This can interfere with running the packet engine (pkteng) for several minutes. Therefore, it is recommended to disable the watchdog before running the packet engine.

```
wl phy_watchdog [<val>]
```

Parameters

val

Value	Description
0	Disable
1	Enable

Returns

Whether the PHY watchdog is currently enabled or disabled.

9.9 radio

Enables or disables the wireless radio through software.

```
wl radio on|off
```

Parameters

radio

Value	Description
on	Enable the wireless radio (LSB = 0).
off	Disable the wireless radio (LSB = 1).

Returns

None

9.10 sgi_rx

Gets or sets the short guard interval (SGI) advertisement in the HT IE.

```
wl sgi_rx [<value>]
```

Parameters

value

Value	Description
0	RX SGI off
1	20 MHz SGI on
2	40 MHz SGI on
3	Both 20 MHz and 40 MHz SGI on

Returns

Current SGI_RX value.

9.11 stbc_rx

Controls the space-time block coding (STBC) receive capability advertisement in the HT IE.

```
wl stbc_rx [<value>]
```

Parameters

value

Value	Description
0	No support
1	Single stream receive support

Returns

Current STBC_RX stat

9.12 stbc_tx

Controls the STBC transmit capability.

```
wl stbc_tx [<value>]
```

Parameters

value

Value	Description
0	Never transmit STBC frames. The STBC TX bit will not be set in the HT IE.
–1	Auto: Transmit STBC if the peer indicates that it can receive it; there are two TX streams enabled, and auto rate has selected a single-stream rate. The STBC TX bit will be set in the HT IE.
1	Enable: Force to transmit STBC only when the active TX chains are greater than 1 and auto rate has selected a single-stream rate.

Returns

Current STBC_TX stat.

10. Rate Commands

10.1 a_mrate

Gets or sets of the IEEE 802.11a multicast rate.

```
wl a_mrate <rate>
```

Parameters

rate

Valid values are in Mbps with the exception of the default setting.

- For IEEE 802.11a: 6, 9, 12, 18, 24, 36, 48, and 54.
- Default: –1 (automatically determine the best rate).

Returns

The 802.11a multicast rate

10.2 a_rate

Gets or sets the IEEE 802.11a unicast rate.

```
wl a_rate <rate>
```

Parameters

rate

Valid values are in Mbps with the exception of the default setting.

- For IEEE 802.11a: 6, 9, 12, 18, 24, 36, 48, and 54.
- Default: –1 (automatically negotiate the best rate).

Returns

The current negotiated rate between a STA and an AP in infrastructure mode or the weighted average of the last 32 frames sent in ad-hoc mode between two stations.

Automatic negotiation is recommended. To override the current rate, issue this command with the new rate. It is recommended that a_rate be used in the 5 GHz band.

10.3 bg_rate

Gets or sets the IEEE 802.11b/g data rate.

```
wl bg_rate [<rate>]
```

Parameters

rate

Valid values are in Mbps with the exception of the default setting.

- For IEEE 802.11b: 1, 2, 5.5, and 11.
- For IEEE 802.11g: 1, 2, 5.5, 6, 9, 11, 12, 18, 24, 36, 48, and 54.
- Default: –1 (automatically determine the best rate).

Returns

The current IEEE 802.11b/g rate

It is recommended that bg_rate be used in the 2.4 GHz band.

10.4 mrate

Gets or sets the IEEE 802.11a/b/g multicast rate.

```
wl mrate <rate>
```

Parameters

rate

Valid values are in Mbps with the exception of the default setting.

- For IEEE 802.11a: 6, 9, 12, 18, 24, 36, 48, and 54.
- For IEEE 802.11b: 1, 2, 5.5, and 11.
- For IEEE 802.11g: 1, 2, 5.5, 6, 9, 11, 12, 18, 24, 36, 48, and 54.
- Default: –1 (automatically negotiate the best rate).

Returns

The multicast rate.

10.5 nrate

Gets or sets the legacy modulation, MCS index, or Short Training Field (STF) mode.

```
wl nrate -[[r <cck|ofdm>]|[m <mcs_index>]|[s <stf_mode>]]
```

Options and Parameters

r

Indicates that a legacy modulation follows, where cck and ofdm are the legacy modulation choices.

m

Indicates that an MCS index follows, where the mcs_index is:

- 0–7 for single-stream MIMO.
- 0–32 for multiple-stream MIMO. The actual range is limited by the number of streams, channel bandwidth, etc.

s

Indicates the STF mode, where stf_mode is one of the following:

Value	Description
0	SISO (Single Input Single Output)
1	CDD (Code Division Duplexing)
2	STBC (Space-Time Block Coding); not supported
3	SDM (Space-Division Multiplexing)

Returns

Legacy modulation, MCS index, or STF mode

Example: Return from a get:

```
legacy rate 54 Mbps stf mode 0 auto
```

10.6 2g_rate

Set or get the rate override for unicast data frames in the 2.4 GHz band.

```
wl 2g_rate [auto|<rate>]|[[-r <rate>]|[ [-h <M>]|[vht params] [-stbc] [-l] [-g]]][-x <T>]
[-b <20|40|80>]
```

Options and Parameters

-r (or --rate), rate

The -r and --rate switches indicate that a legacy rate follows. The legacy rates can be any of the following: 1, 2, 5.5, 11, 6, 9, 12, 18, 24, 36, 48, and 54.

-h, --ht

The -h and --ht switches indicate that a high throughput MCS index follows, where the index (M) is 0 to 23.
If desired, --ht=<M> can be used instead of -h <M>.

vht params: -v, --vht, M, -c, -s, and Nss

For the case of very high throughput, several different representations are available. In all representations, M represents the MCS index, which can range from 0 to 9, and Nss represents the number of spatial streams, which can range from 1 to 8. The set of possible very high throughout representations that convey the same set up are shown below along with an example when the MCS index (M) is 5 and the number of spatial streams (Nss) is 2:

```
-v <M>¡Ñ<Nss> (Example: -v 5¡Ñ2)
-v <M> -s <Nss> (Example: -v 5 -s 2)
-v <M> --s=<Nss> (Example: -v 5 --s=2)
--vht=<M>¡Ñ<Nss> (Example: --vht=5¡Ñ2)
--vht=<M> --s=<Nss> (Example: --vht=5 --s=2)
-c c<M>s<S> (Example: -c c5s2)
```

-stbc

If present, this switch indicates that Space-time Block Coding (STBC) expansion is to be used.

-x (or --exp)

The presence of either of these switches indicate a TX expansion, which is the number of TX chains beyond the minimum number of TX chains required for the space-time streams.

-l (or --ldpc)

The presence of either of these switches indicate that a Low-Density Parity Check (LDPC) is to be done.

-g (or --sgi)

The presence of either of these switches indicates that a Short Guard Interval (SGI) is to be used. A standard guard internal is otherwise used.

-b (or --bandwidth)

If either of these switches is present, the value that follows it is the transmit bandwidth to set.

Returns

The current rate override for the 2.4 GHz band or auto if no override is applied.

Examples:

```
wl 2g_rate auto
wl 2g_rate 18
wl 2g_rate -r 18 -b 40
wl 2g_rate -v 5x2 -g -l
```

10.7 2g_mrate

Set or get the rate override for multicast data frames in the 2.4 GHz band.

```
wl 2g_mrate [auto|<rate>]|[[-r <rate>]|[ [-h <M>]|[vht params] [-stbc] [-l] [-g]]][-x <T>]
[-b <20|40|80>]
```

Options and Parameters

-r (or --rate), rate

The -r and --rate switches indicate that a legacy rate follows. The legacy rates can be any of the following: 1, 2, 5.5, 11, 6, 9, 12, 18, 24, 36, 48, and 54.

-h, --ht

The -h and --ht switches indicate that a high throughput MCS index follows, where the index (M) is 0 to 23. If desired, --ht=<M> can be used instead of -h <M>.

vht params: -v, --vht, M, -c, -s, and Nss

For the case of very high throughput, several different representations are available. In all representations, M represents the MCS index, which can range from 0 to 9, and Nss represents the number of spatial streams, which can range from 1 to 8. The set of possible very high throughout representations that convey the same set up are shown below along with an example when the MCS index (M) is 5 and the number of spatial streams (Nss) is 2:

```
-v <M>¡Ñ<Nss> (Example: -v 5¡Ñ2)
-v <M> -s <Nss> (Example: -v 5 -s 2)
-v <M> --s=<Nss> (Example: -v 5 --s=2)
--vht=<M>¡Ñ<Nss> (Example: --vht=5¡Ñ2)
--vht=<M> --s=<Nss> (Example: --vht=5 --s=2)
-c c<M>s<S> (Example: -c c5s2)
```

-stbc

If present, this switch indicates that Space-time Block Coding (STBC) expansion is to be used.

-x (or --exp)

The presence of either of these switches indicate a TX expansion, which is the number of TX chains beyond the minimum number of TX chains required for the space-time streams.

-l (or --ldpc)

The presence of either of these switches indicate that a Low-Density Parity Check (LDPC) is to be done.

-g (or --sgi)

The presence of either of these switches indicates that a Short Guard Interval (SGI) is to be used. A standard guard internal is otherwise used.

-b (or --bandwidth)

If either of these switches is present, the value that follows it is the transmit bandwidth to set.

Returns

The current rate override for the 2.4 GHz band or auto if no override is applied.

Examples:

```
wl 2g_mrate auto
wl 2g_mrate 18
wl 2g_mrate -r 18 -b 40
wl 2g_mrate -v 5x2 -g -l
```

10.8 5g_rate

Set or get the rate override for unicast data frames in the 5 GHz band.

```
wl 5g_rate [auto|<rate>]|[[-r <rate>]|[ [-h <M>]|[vht params] [-stbc] [-l] [-g]]][-x <T>]
[-b <20|40|80>]
```

Options and Parameters

-r (or --rate), rate

The -r and --rate switches indicate that a legacy rate follows. The legacy rates can be any of the following: 1, 2, 5.5, 11, 6, 9, 12, 18, 24, 36, 48, and 54.

-h, --ht

The -h and --ht switches indicate that a high throughput MCS index follows, where the index (M) is 0 to 23. If desired, --ht=<M> can be used instead of -h <M>.

vht params: -v, --vht, M, -c, -s, and Nss

For the case of very high throughput, several different representations are available. In all representations, M represents the MCS index, which can range from 0 to 9, and Nss represents the number of spatial streams, which can range from 1 to 8. The set of possible very high throughout representations that convey the same set up are shown below along with an example when the MCS index (M) is 5 and the number of spatial streams (Nss) is 2:

```
-v <M>¡Ñ<Nss> (Example: -v 5¡Ñ2)
-v <M> -s <Nss> (Example: -v 5 -s 2)
-v <M> --s=<Nss> (Example: -v 5 --s=2)
--vht=<M>¡Ñ<Nss> (Example: --vht=5¡Ñ2)
--vht=<M> --s=<Nss> (Example: --vht=5 --s=2)
-c c<M>s<S> (Example: -c c5s2)
```

-stbc

If present, this switch indicates that Space-time Block Coding (STBC) expansion is to be used.

-x (or --exp)

The presence of either of these switches indicate a TX expansion, which is the number of TX chains beyond the minimum number of TX chains required for the space-time streams.

-l (or --ldpc)

The presence of either of these switches indicate that a Low-Density Parity Check (LDPC) is to be done.

-g (or --sgi)

The presence of either of these switches indicates that a Short Guard Interval (SGI) is to be used. A standard guard internal is otherwise used.

-b (or --bandwidth)

If either of these switches is present, the value that follows it is the transmit bandwidth to set.

Returns

The current rate override for the 5 GHz band or auto if no override is applied.

Examples:

```
wl 5g_rate auto
wl 5g_rate 18
wl 5g_rate -r 18 -b 40
wl 5g_rate -v 5x2 -g -l
```

10.9 5g_mrate

Set or get the rate override for multicast data frames in the 5 GHz band.

```
wl 5g_mrate [auto|<rate>]|[[-r <rate>]|[ [-h <M>]|[vht params] [-stbc] [-l] [-g]]][-x <T>]
[-b <20|40|80>]
```

Options and Parameters

-r (or --rate), rate

The -r and --rate switches indicate that a legacy rate follows. The legacy rates can be any of the following: 1, 2, 5.5, 11, 6, 9, 12, 18, 24, 36, 48, and 54.

-h, --ht

The -h and --ht switches indicate that a high throughput MCS index follows, where the index (M) is 0 to 23. If desired, --ht=<M> can be used instead of -h <M>.

vht params: -v, --vht, M, -c, -s, and Nss

For the case of very high throughput, several different representations are available. In all representations, M represents the MCS index, which can range from 0 to 9, and Nss represents the number of spatial streams, which can range from 1 to 8. The set of possible very high throughout representations that convey the same set up are shown below along with an example when the MCS index (M) is 5 and the number of spatial streams (Nss) is 2:

```
-v <M>¡Ñ<Nss> (Example: -v 5¡Ñ2)
-v <M> -s <Nss> (Example: -v 5 -s 2)
-v <M> --s=<Nss> (Example: -v 5 --s=2)
--vht=<M>¡Ñ<Nss> (Example: --vht=5¡Ñ2)
--vht=<M> --s=<Nss> (Example: --vht=5 --s=2)
-c c<M>s<S> (Example: -c c5s2)
```

-stbc

If present, this switch indicates that Space-time Block Coding (STBC) expansion is to be used.

-x (or --exp)

The presence of either of these switches indicate a TX expansion, which is the number of TX chains beyond the minimum number of TX chains required for the space-time streams.

-l (or --ldpc)

The presence of either of these switches indicate that a Low-Density Parity Check (LDPC) is to be done.

-g (or --sgi)

The presence of either of these switches indicates that a Short Guard Interval (SGI) is to be used. A standard guard internal is otherwise used.

-b (or --bandwidth)

If either of these switches is present, the value that follows it is the transmit bandwidth to set.

Returns

The current rate override for the 5 GHz band or auto if no override is applied.

Examples:

```
wl 5g_mrate auto
wl 5g_mrate 18
wl 5g_mrate -r 18 -b 40
wl 5g_mrate -v 5x2 -g -l
```

10.10 rateset

Gets or sets the supported and basic rate-set.

```
wl rateset [default|all|<rateset_arg>]
```

Parameters

default

Use driver defaults.

all

All rates are basic rates

rateset_arg

A string of rates where at least one rate is a basic rate. All rates have units of Mbps.

Returns

The rate-set if no arguments are provided.

The list of rates is in Mbps, and each rate is optionally followed by (b) or b, both of which stand for basic rate.

Example: Basic rate

1(b) 2b 5.5 11

11. Security and Encryption Controls

11.1 addwep

Sets an IEEE 802.11 Wired Equivalent Privacy (WEP) encryption key.

```
wl addwep <keyindex> <keydata> [ocb | ccm] [notx] [<address>]
```

Parameters

keyindex

Typically 1–4

keydata

5, 13, or 16-bytes (or 10, 26, 32, or 64 hexadecimal digits)

ocb

Key type is AES-OCB (i.e., the key is for use with AES-OCB encryption)

ccm

Key type is AES-CCM (i.e., the key is for use with AES-CCM encryption)

notx

Indicates that the key should not be transmitted.

address

Ethernet address associated with the key being set

Returns

None

Issuing this command results in the sending of OID_802_11_ADD_WEP OID, which requests the miniport driver to set an IEEE 802.11 WEP key to the value specified.

The encryption algorithm is automatically selected based on the key size. The key type is accepted only when the key length is 16 bytes (or 32 hexadecimal digits) and specifies whether AES-OCB or AES-CCM encryption is used. The default is AES-CCM.

11.2 auth

Gets or sets the IEEE 802.11 authentication type.

```
wl auth [<value>]
```

Parameters

value

Value	Description
0	Open system
1	Shared key

Returns

The IEEE 802.11 authentication type

11.3 authorize

Restricts traffic to IEEE 802.1x packets.

```
wl authorize
```

Parameters

None

Returns

The authorized STA MAC address (for example, 60:A8:83:BF:01:00)

11.4 deauthorize

Does not restrict traffic to IEEE 802.1x packets.

```
wl deauthorize
```

Parameters

None

Returns

The deauthorized AP’s MAC address (for example, A0:7C:87:BF:01:00)

11.5 eap

Gets or sets whether to restrict traffic to IEEE 802.1x packets until IEEE 802.1x authorization succeeds.

```
wl eap [<mode>]
```

Parameters

mode

Value	Description
0	Disable (do not restrict packets)
1	Enable (restrict packets)

Returns

Whether traffic is restricted to IEEE 802.1x packets until IEEE 802.1x authorization succeeds.

11.6 macmode

Sets the mode of the MAC source address list.

```
wl macmode <value>
```

Parameters

value

Value	Description
0	Disable MAC address matching.
1	Deny association to stations on the MAC list.
2	Allow association to stations on the MAC list.

Returns

None

11.7 pmkid_info

Gets the PMK ID table.

```
wl pmkid_info
```

Options and Parameters

Value	Description
-s <S>	The SSID (represented by S) to scan.
-t [active | passive]	The scan type.
--bss_type [bss/infra | ibss/adhoc]	The bss type to scan.
-b <MAC>	The MAC-particular BSSID address to scan, where MAC is an address of the form xx:xx:xx:xx:xx:xx.
-n <N>	The number of probes (N) per scanned channel.
-a <N>	The dwell time per channel (N) for active scanning.
-p <N>	The dwell time per channel (N) for passive scanning.
-h <N>	The dwell time for the home channel (N) between channel scans.
-c <L>	The comma- or space-separated list (L) of channels to scan.

Returns

None

11.8 primary_key

Gets or sets the primary encryption key index.

```
wl primary_key <keyindex>
```

Parameters

keyindex

Typically 1–4

Returns

The primary encryption key associated with index if a primary key was previously set.

11.9 rmwep

Removes the encryption key at the specified index.

```
wl rmwep <keyindex>
```

Parameters

keyindex

Typically 1–4

Returns

None

11.10 set_pmk

Sets the Pairwise Master Key (PMK) passphrase in the driver-resident supplicant.

```
wl set_pmk <passphrase> <length>
```

Parameters

passphrase

PMK passphrase

length

8–64 (PMK passphrase length, in bytes)

Returns

None

11.11 wepstatus

Gets or sets the Wired Equivalent Privacy (WEP) status.

```
wl wepstatus [on|off]
```

Parameters

on

Enable WEP

off

Disable WEP

Returns

WEP status from a get operation

11.12 wpa_auth

Sets the WPA. authorization mode.

```
wl wpa_auth [<value>]
```

Parameters

value

Value	Description
1	WPA-NONE
2	WPA-802.1X/WPA-Professional
4	WPA-PSK/WPA-Persona

Returns

The WPA authorization mode

11.13 wpa_cap

Gets or sets whether IEEE 802.11i Robust Security Network (RSN) is on or off.

```
wl wpa_cap [on|off]
```

Parameters

on

Turns the RSN capability on.

off

Turns the RSN capability off.

Returns

0 if the capability is off.

1 if the capability is on.

11.14 wsec

Gets or sets the wireless security bit vector.

```
wl wsec [<vector>]
```

Parameters

vector

Value	Description
Bit 0	WEP enabled
Bit 1	TKIP enabled
Bit 2	AES enabled
Bit 3	WSEC in software

Returns

A bit vector in the same form as the above vector parameter

12. Scan Engine Control

12.1 passive

Sets the scan engine to passive mode.

```
wl passive
```

Parameters

None

Returns

None

12.2 scan

Initiates a scan. The default scan is an active scan across all channels for any SSID.

```
wl scan
```

Parameters

None

Returns

None

To check the result of the scan use scanresults on page 57.

12.3 scan_channel_time

Gets or sets the scan association time.

```
wl scan_channel_time [<time>]
```

Parameters

time

0 to 65535. Default is 20.

Returns

The scan channel time

12.4 scan_home_time

Gets or sets the scan home channel dwell time.

```
wl scan_home_time [<time>]
```

Parameters

time

0 to 65535. Default is 45.

Returns

The scan home channel dwell time

12.5 scan_nprobes

Gets or sets the number of probes to use (per channel scanned).

```
wl scan_nprobes [<number>]
```

Parameters

number

Default is 2.

Returns

The number of probes to use

12.6 scan_passive_time

Gets or sets the passive scan channel dwell time.

```
wl scan_passive_time [<time>]
```

Parameters

time

Dwell time in milliseconds. The range is 10 to 1000, and the default is 3.

Returns

The passive scan channel dwell time

12.7 scan_unassoc_time

Gets or sets the unassociated scan channel dwell time.

```
wl scan_unassoc_time [<time>]
```

Parameters

time

0 to 65535. Default is 40.

Returns

The unassociated scan channel dwell time

12.8 scanresults

Gets the results of the last scan.

```
wl scanresults
```

Parameters

None

Returns

Basic network information on the APs and STAs discovered during a scan.

Example:

SSID: Cypress

Mode: Managed RSSI: -78 dBm noise: -65 dBm Channel: 40

BSSID: 00:10:18:90:2D:61 Capability: ESS

Supported Rates: [6(b) 9 12(b) 18 24(b) 36 48 54]

The `wl scan` command must be issued before this command.

12.9 scansuppress

Gets or sets whether scanning is suppressed.

```
wl scansuppress [<value>]
```

Parameters

value

Value	Description
0	Allow scans
1	Suppress scans

Returns

The scan suppression status

13. Association

13.1 assoc

Gets information about the network association.

```
wl assoc
```

Parameters

None

Returns

Network association information

Example: Command response to standard output

```
Associated with bssid: 00:10:18:90:27:7B SSID: "paul"
Last association request:
Capabilities: ESS ShortPre ShortSlot
Listen Interval: 10
SSID: "paul"
Supported Rates: [1(b) 2(b) 5.5(b) 11(b) 18 24 36 54]
ID(21): 08 0F
ID(24): 01 0E
```

13.2 assoc_info

Gets the associated request and response information (STA only).

```
wl assoc_info
```

Parameters

None

Returns

Association request and response information

13.3 assoc_pref

Gets or sets an association preference.

```
wl assoc_pref [auto|a|b|g]
```

Parameters

auto

Preference is a function of software

a

Prefer IEEE 802.11a associations

b

Prefer IEEE 802.11b associations

g

Prefer IEEE 802.11g associations

Returns

The currently configured association preference from a get operation

13.4 bi

Gets or sets the beacon interval (BI).

```
wl bi [<interval>]
```

Parameters

interval

32-bit unsigned integer in milliseconds

Returns

The beacon interval in milliseconds

13.5 bssid

Gets the BSS ID.

```
wl bssid
```

Parameters

None

Returns

The 48-bit MAC address of an AP's WLAN interface that serves the stations in a basic service set (BSS).

13.6 cap

Gets the capabilities of the driver.

```
wl cap
```

Parameters

None

Returns

The driver capabilities

Example: output
```
sta afterburner
```

13.7 closed

Hides the network from active scans.

```
wl closed <open_hide>
```

Parameters

open_hide

Value	Description
0	Open
1	Hide

Returns

None

13.8 disassoc

Disassociates from the current BSS or IBSS.

```
wl disassoc
```

Parameters

None

Returns

None

13.9 join

Joins a specified network SSID.

```
wl join <ssid> [key <xxxxx>] [imode bss|infra|ibss|adhoc]
[amode open|shared|wpa|wpapsk|wpa2|wpa2psk|wpanone]
```

Parameters

ssid

The SSID of the network to be joined

xxxxx

The WEP key value if joining an encrypted network

imode

Infrastructure mode

Value	Description
bss or infra	Indicates that the network to join is an infrastructure network.
ibss or adhoc	Indicates that the network to join is an adhoc network.

amode

Authentication mode

`open`
`shared`
`wpa`
`wpapsk`
`wpa2`
`wpa2psk`
`wpanone`

Returns

None

Example: join KILROY imode infra amode open

The authentication mode can be open or shared. Under shared-key authentication, each wireless station is assumed to have received a secret shared key over a secure channel that is independent from the IEEE 802.11 wireless network communications channel.

13.10 reassoc

Reassociates (roams) to the AP with the specified BSSID.

```
wl reassoc <bssid>
```

Parameters

bssid

The BSSID of the AP to reassociate with

Returns

None

13.11 shortslot

Gets or sets the IEEE 802.11g short-slot timing mode.

```
wl shortslot [<mode>]
```

Parameters

mode

Value	Description
0	Long
1	Short

Returns

Short-slot timing mode as:

Value	Description
0	Long
1	Short

13.12 shortslot_override

Gets or sets the IEEE 802.11g short-slot timing mode override.

```
wl Shortslot_override [<value>]
```

Parameters

value

Value	Description
–1	Auto.
0	Short slot is off.
1	Short slot is on.

Returns

The short-slot timing mode override from a get operation

13.13 ssid

Get or set a configuration's SSID.

```
wl ssid [<ssid_string>]
```

Parameters

ssid_string

The SSID string to be set.

Returns

The current SSID.

A set operation initiates an association attempt if in infrastructure mode, initiates an IBSS join/create if in IBSS mode, or creates a BSS if in AP mode.

Example: set: wl ssid testnetwork results in display of Setting SSID "testnetwork"

Example: get: wl ssid results in display of Current SSID: "testnetwork"

14. Channel and Band Control

14.1 band_range

Gets the current band range identification index number.

```
wl band_range
```

Parameters

None

Returns

The index number for band identification where:

Index Value	Description
0	2.4 GHz
1	5 GHz, low band
2	5 GHz, mid band
3	5 GHz, high band

14.2 band

Gets or sets the band.

```
wl band [<auto>]
```

Parameters

value

Value	Description
auto	Automatically switch between available bands (default)
a	Force use of IEEE 802.11a band
b	Force use of IEEE 802.11b band

Returns

The band in use.

14.3 bands

Gets the list of available IEEE 802.11 bands.

```
wl bands
```

Parameters

None

Returns

The list of available IEEE 802.11 bands

Example: band list output: b a

14.4 chan_info

Gets channel information.

```
wl chan_info
```

Parameters

None

Returns

Channel information

Example: Channel information output

```
Channel 1 B Band
Channel 2 B Band
. .
. .
Channel 10 B Band
Channel 11 B Band
Channel 36 A Band
Channel 40 A Band
Channel 44 A Band
Channel 48 A Band
Channel 52 A Band, RADAR Sensitive, Passive
Channel 56 A Band, RADAR Sensitive, Passive
Channel 60 A Band, RADAR Sensitive, Passive
Channel 64 A Band, RADAR Sensitive, Passive
Channel 149 A Band
Channel 153 A Band
Channel 157 A Band
Channel 161 A Band
Channel 165 A Band
```

14.5 channel

Gets or sets the working channel.

```
wl channel [<channel #>]
```

Parameters

channel #

Valid channels for 802.11a are: 36, 40, 44, 48, 52, 56, 60, 64, 100, 104, 108, 112, 116, 120, 124, 128, 132, 136, 140, 149, 153, 157, 161, 184, 188, 192, 196, 200, 204, 208, 212, 216.

Valid channels for 802.11b/g are 1–14.

Returns

None

14.6 channels

Gets the valid channels for the current settings.

```
wl channels
```

Parameters

None

Returns

The valid channels for the current configuration

14.7 channels_in_country

Gets the valid channels for the country specified.

```
wl channels_in_country <arg1> <arg2>
```

Parameters

arg1

Country abbreviation

arg2

Band, which can be a or b

Returns

Valid channels for the specified country and band

Two-letter country codes:

AU Australia

BN Brunei Darussalam

CA Canada

DE Germany

EU Europeanwide AP

JP Japan

KR Korea

KW Kuwait

MX Mexico

RU Russian Federation

TW Taiwan, Province Of China

UA Ukraine

AE United Arab Emirates

US United States

14.8 chanspec

This command applies to all currently shipping chip sets that have IEEE 802.11 capability. Sets or gets the channel based on the IEEE 802.11n channel allocation specifications.

```
wl chanspec [-c <channel> -b <band> -w <bandwidth> -s <sideband>] (Usage 1)
wl chanspec [<channel>[band] [bandwith][sideband]] (Usage 2)
```

Options and Parameters

channel

Usage1 and 2:

0–224

band

Usage 1:

Value	Description
2	2.4 GHz. (This is the default if the channel is 14 or less.)
5	5 GHz.

Usage 2:

Value	Description
b	2.4 GHz. (This is the default if the channel is 14 or less.)
a	5 GHz.

bandwidth

Usage 1 and Usage 2:

Value	Description
20	20 MHz
40	40 MHz

Note: Regarding Usage 2: If no bandwidth is specified, then the bandwidth is determined by the sideband entry.

Note: If the band, bandwidth, and sideband are not specified, then the default is 20 MHz with no sideband. The band is 2.4 GHz for channels 1–14 and 5 GHz for channels above channel 14.

sideband

Usage 1:

Identifies which adjacent channel to use when the bandwidth is 40 MHz.

Value	Description
–1	Low (or lower) sideband (applies to a 40 MHz channel bandwidth).
0	No sideband (for 20 MHz channel bandwidth).
1	High (or upper) sideband (applies to a 40 MHz channel bandwidth).

Usage 2:

Value	Description
u	Upper sideband (applies to a 40 MHz channel bandwidth).
l	Lower sideband (applies to a 40 MHz channel bandwidth).

Note: Regarding Usage 2: If no sideband is entered, then the channel bandwidth is 20 MHz.

Returns

The IEEE 802.11n channel specification

Example: Usage 1

```
wl chanspec -c 7 -b 2 -w 20 -s 0
wl chanspec -c 7 -b 2 -w 40 -s -1
```

Example: Usage 2

```
wl chanspec 7
wl chanspec 7u
```

15. Roam Control

15.1 prb_resp_timeout

Gets or sets the probe response timeout.

```
wl prb_resp_timeout [<time>]
```

Parameters

time

Response timeout in milliseconds. The range is 0 to 116, and the default is 0.

Returns

The probe response timeout from a get operation

15.2 roam_delta

Gets or sets the roam candidate qualification delta.

```
wl roam_delta [<integer>] a|b
```

Parameters

integer

The RSSI roam delta in dB. It must be an integer.

a/b

Optional band to which the roam delta is applied.

Returns

The roam candidate qualification delta

15.3 roam_scan_period

Get or set the roam candidate scan period.

```
wl roam_scan_period [<integer>]
```

Parameters

integer

The roam candidate scan period in milliseconds

Returns

The roam candidate scan period

15.4 roam_off

Enables or disables WLAN STA roaming.

```
wl roam_off [<stat>]
```

Parameters

stat

Value	Description
0	Roaming is on (default).
1	Roaming is off.

Returns

The roaming on/off state

15.5 roam_trigger

Gets or sets the RSSI roam trigger threshold.

```
wl roam_trigger [<integer>] a|b
```

Parameters

integer

The roam-trigger power level in dBm at the receiver input. It must be an integer.

a/b

Optional band to which the roam trigger is applied.

Returns

The roam-trigger threshold

Example:

`wl roam_trigger –75` (sets the roam trigger to –75 dBm).

16. Test and Measurements

16.1 cis_source

Gets the NVRAM medium.

```
wl cis_source
```

Parameters

None

Returns

Value	Description
0	Default
1	SROM
2	OTP (One-time programmable memory)

16.2 cisdump

Gets the binary content of the Card Information Structure (CIS).

```
wl cisdump <option>
```

Parameters

option

–b <filename>.
IOVAR directs binary output to the file specified by `filename`.

Returns

A buffer with a display of the CIS content for functions 0 through 2, regardless of the content source, that is, regardless of whether it is OTP memory or not.

Example: Return output

```
Source 2 (Internal OTP)
Byte 0: 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00
Byte 8: 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00
Byte 16: 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00
Byte 24: 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00
...
Byte 88: 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00
Byte 96: 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00
Byte 104: 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00
Byte 112: 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00
...
```

16.3 ciswrite

Writes the contents of a supplied binary file into NVRAM. (See cis_source on page 72.)

```
wl ciswrite <binary_file>
```

Parameters

None

Returns

None

16.4 Command Batching

Command batching is a mechanism whereby multiple IOCTL commands, with possible delays inserted, are sent to the driver. The driver doesn't actually run the IOCTLs until it has received the entire list.

The client machine executes a command batch sequence by:

- Sending a seq_start to mark the start of a sequence.
- Sending the appropriate IOCTLs and IOVARs with seq_delays inserted, if necessary.
- Sending a seq_stop to mark the end of a sequence.

The driver queues all IOCTLs and IOVARs until it receives seq_stop.

16.4.1 seq_delay

Indicates a time interval that the driver should delay before processing the next command.

```
wl seq_delay time_msec
```

Parameters

time_msec

Delay value in the range 1–10000 msec

IOVAR Support

Set only

Notes

None

16.4.2 seq_error_index

Retrieves the index of the command that failed within a command sequence. Index starts at 1.

```
wl seq_error_index
```

Parameters

None

IOVAR Support

Get only

Notes

A UINT32 is returned.

16.4.3 seq_start

Marks the start of command batching or queueing. This is a set IOVAR. Get IOVAR is not supported.

```
wl seq_start
```

Parameters

None

IOVAR Support

Set only

Notes

None

16.4.4 seq_stop

Marks the end of command batching or queueing.

```
wl seq_stop
```

Parameters

None

IOVAR Support

Set only

Notes

The firmware starts processing the commands sequentially, starting from the head of the queue.

Example: Sample batch command usage sequence.

```
wl seq_start
wl channel 5
wl scan
wl join ¡§helloworld¡¨
wl seq_stop
wl seq_error_index
```

16.5 country

Gets or sets the country code for use with IEEE 802.11d.

```
wl country [cc | [list] [a|b]]
```

Parameters

cc

Country code to enter during a set. This is the only parameter used during a set operation.

list

Provides a list of all country codes that the system will support. Used during get operations.

band a|b

Value	Description
a	5 GHz
b	2.4 GHz

Returns

The current country code setting or a list of possible country code settings if used with list and/or band

Notes

Use either a long name or the appropriate ISO 3166 abbreviation.

Using `wl` country with no parameters returns the currently set country code.

16.6 curpower

Gets the current TX power settings.

```
wl curpower
```

Options

–q

quiet: estimated power only

Returns

The current TX power settings

16.7 fqacurcy

Sets the frequency accuracy mode for manufacturing test purposes.

```
wl fqacurcy <channel>
```

Parameters

channel

1–14 or 0 to stop the test

Returns

None

This command is used to measure carrier center-frequency accuracy. While in this mode, a spectrum analyzer can be used to measure frequency accuracy within a specific channel.

This command is used during testing. When done testing, be sure to type wl fqacurcy 0 followed by a wl up.

16.8 interference

Gets or sets the interference mitigation mode.

```
wl interference <mode>
```

Parameters

mode

Value	Description
0	None
1	Non-WLAN. When enabled, makes the receiver less sensitive. This is used to avoid strong non-IEEE 802.11 interference, such as from 2.4 GHz phones, 2.4 GHz audio/visual wireless transmitters, and non-wireless devices in the same band.
2	WLAN manual. Enable Adjacent Channel Interference (ACI) mode. A manual setting that makes the receiver less sensitive to IEEE 802.11 traffic on adjacent channels at some cost to sensitivity.
3	WLAN automatic. Automatic ACI¡Xthe driver determines whether ACI should be enabled.

Returns

The interference mode

Notes

Automatic WLAN interference mitigation is enabled and not active.

16.9 Packet Engine Controls

The packet engine is only available with manufacturing test and internal builds. It can be used in one of the following modes:

- TX mode: In this mode, the DUT transmits data packets of configurable length with configurable Inter-frame Spacing (IFS) (in microseconds). It will transmit the number of packets specified. If this number is zero, it will transmit packets continuously. The same packet is transmitted with an incremental sequence number. The retry bit in the frame control is not set for any frame. There is no backoff procedure after a frame is transmitted. If the packet engine is put in TX mode, it must be stopped before putting it back in TX mode.
- TX mode with ACK: In this mode, the device transmits data packets the same way as it does while in TX mode, and it will also update the rxdfrmucastmbss count.
- RX Mode: In this mode, the device listens to the frames sent to the specified MAC address. The sequence number of frames into the DUT must increment. In this mode, ACKs will not be sent from the DUT. The IFS and frame count are ignored in this mode. The lost frames are counted based on the sequence number of the frames successfully received.
- RX mode with ACK: In this mode, the DUT will send an ACK to the dummy MAC address upon receipt of packets destined for the DUT. Everything else is the same as in RX mode.

The DUT does not perform its normal network functionality when operating in any of the above modes.

16.9.1 pkteng_start

Starts packet transmission or reception.

```
wl pkteng_start <des_addr> [tx|txwithack] [sync|asyn] [<ipg>] [<length>] [<framecnt>]
[<src_addr>]
wl pkteng_start des_addr [rx|rxwithack] [sync|asyn] [<framecnt>] [<time_out>]
```

Options

des_addr

In TX mode, this is the destination MAC address of the dummy MAC address. (This is different from the DUT MAC address.).

In RX mode, this is the DUT's Ethernet MAC address. (Frames destined for this address are accepted.)

In RX mode with ACK, the DUT accepts frames with this address and responds to each with an ACK.

In TX mode with ACK, the DUT acknowledges the number of packets transmitted.

rx

Enable the packet engine's RX mode.

tx

Enable the packet engine's TX mode.

rxwithack

Enable the packet engine's RX with ACK mode.

txwithack

Enable the packet engine's TX with ACK mode.

sync

Forces the packet engine call to operate in synchronize mode. Do not choose the sync option when the DUT is set to transmit continuously because this will cause the host system to freeze.

asyn

The packet engine call returns after a predefined arbitrary delay that follows the completion of transmission or reception. By default, this is set when the sync mode is not set.

ipg

Inter-packet gap in microseconds. Only used in TX mode. Value range is 20–1000 ms.

length

The number of bytes in each frame.

framecnt

In transmit mode, this is the number of frames to be transmitted. If set to 0, the DUT will transmit continuously. In receive sync mode, this is the number of expected packets.

time_out

In receive sync mode, this is the time interval over which all expected packets should be received. (The value entered is in milliseconds and it should not exceed 3 seconds.)

src_addr

In TX mode, this is the MAC address of the DUT (transmission source). In RX mode, this is the MAC address of the reference transmission source. (This option is usually omitted.)

16.9.2 pkteng_stats

Displays packet statistics.

```
wl pkteng_stats
```

Options

None

16.9.3 pkteng_stop

Stops packet transmission or reception.

```
wl pkteng_stop [tx|rx]
```

Options

tx

Stop transmitting packets.

rx

Stop receiving packets.

Usage Examples

Example

To transmit 1000 frames of 200 bytes with a 30 ƒÝs IFS:

wl pkteng_start 10:20:30:40:50:60 tx 30 200 1000

Example

To continuously transmit 1000-byte frames with a 40 ƒÝs IFS:

wl pkteng_start 10:20:30:40:50:60 tx 40 1000 0

Usage Sequence

Example

A sample packet engine usage sequence:

```
wl mpc 0
wl up
wl phy_watchdog 0
wl pkteng_start [addr] [tx|rxwithack|rx] [ipg] [length] [framecnt]
wl pkteng_stop [tx|rx]
wl pkteng_stats
```

16.10 pwr_percent

Gets or sets the output power percentage referenced to a full power percentage of 100%.

```
wl pwr_percent [<arg1>]
```

To set a transmission power percentage that is 60 percent of full power, issue the following command:
`wl pwr_percent 60`

To get the percentage of full transmission power currently being used, issue the following command:
`wl pwr_percent`

Parameters

arg1

The percentage of full transmission power to set. Range is 0–100.

Returns

The percentage of full transmit power.

16.11 rssi

Gets the latest RSSI value.

The MAC layer operates together with the physical layer by sampling transmitted energy. The PHY uses a clear channel assessment (CCA) algorithm to determine if the channel is clear. This is accomplished by measuring the received signal strength at the antenna. This measured signal is commonly known as the RSSI. If the received signal strength is below a specified threshold, then the channel is declared clear, and the MAC layer is given the clear channel status for data transmission. If the RF energy is above the threshold, data transmissions are deferred in accordance with protocol rules. The standard provides another option for CCA that can be alone or with the RSSI measurement.

```
wl rssi
```

Parameters

None

Returns

The current RSSI value

16.12 spect

Gets or sets the IEEE 802.11h spectrum management mode.

```
wl spect[<arg1>]
```

Parameters

arg1

Value	Description
0	Off
1	Loose interpretation of the IEEE 802.11h spec. May join non-IEEE 802.11h compliant APs.
2	Strict interpretation of the IEEE 802.11h spec. May not join non-IEEE 802.11h compliant APs.
3	Disable IEEE 802.11h and enable IEEE 802.11d.
4	Loose interpretation of the IEEE 802.11h and IEEE 802.11d specifications. May join non-IEEE 802.11h APs.

Returns

One of the five IEEE 802.11h spectrum management modes identified in the parameters section.

16.13 txpwr1

Gets or sets the transmit power using one of three unit definitions.

```
wl txpwr1 [-d|-q|-m] <value>] [-o]
```

Options and Parameters

Value	Description
–d	Use dBm for power units.
–q	Use quarter dBm for power units.
–m	Use milliwatts for power units.
–o	Used to override regulatory and other power-constraint limits.
value	dBm range:0–20 (typical). Quarter dBm range:0–80 (typical). mW range:1–100 (not frequently used).

Returns

None

Note: Can be combined with –o to turn on override to disable regulatory and other limitations.

17. Gmode Controls

17.1 gmode

Gets or sets the 54gR mode.

```
wl gmode [<mode>]
```

Parameters

mode

Value	Description
Auto	Rate set: 1b, 2b, 5.5b, 11b, 18, 24, 36, and 54 Extended Rate set: 6, 9, 12, and 48 Preamble: Long Shortslot: Auto
GOnly	Rate set: 1b, 2b, 5.5b, 6, 9, 11, 12, 18, 24, 36,48, and 54
LegacyB	Rate set: 1b, 2b, 5.5, and 11 Preamble: Long Shortslot: Off
LRS	Rate set: 1b, 2b, 5.5, and 11 (CCK only) Extended Rate set: 6, 9, 12, 18, 24, 36, 48, and 54 Preamble: Long Shortslot: Auto
Performance	Rate set: 1b, 2b, 5.5b, 6b, 9, 11b, 12b, 18, 24b, 36, 48, and 54 Preamble: Short required Shortslot: On and required

Returns

The 54g mode

Note: A lowercase b following a rate, such as 1b and 2b, indicates that the rate is a basic rate.

17.1.1 Auto [Default]

This mode allows for maximum compatibility and is fully 802.11g specification compliant.

All twelve 802.11g rates will be advertised: 1b, 2b, 5.5b, 6, 9, 11b, 12, 18, 24, 36, 48, and 54, but the basic rate set will only include 1, 2, 5.5, and 11, so that legacy 802.11b clients can associate (no-OFDM).

This mode defaults to high-speed 802.11g operation by using short-slot timing. The AP will switch to long-slot timing if an 802.11b or an 802.11g client that does not support short-slot timing enters the network.

In this mode, the AP will advertise short-slot timing in the beacon frames. As long as there are only 11g clients associated, short-slot timing will be used and throughput will be high.

If a client that does not support short-slot timing (e.g., a legacy 11b client) joins the BSS, the AP will cease to advertise short-slot timing. All STA devices in the BSS will then start to use the normal 11b interpacket timings. If all non-short-slot timing STAs leave the BSS, the AP and remaining STAs will then begin to use short-slot timing again.

Summary:

- Use of 802.11g short-slot timing is automatic.
- If no 802.11b clients are associated, short-slot timing is used.
- If an 802.11b client associates, then the AP uses long-slot timing.
- Since no OFDM rates are in the basic rate set, the maximum 802.11g performance is not available.

17.1.2 LegacyB

When operating in LegacyB g-mode, only Complementary Code Keying (CCK) rates are allowed in the network, and only 1 and 2 Mbps are Basic, so legacy 802.11 devices can join (older than 802.11b). In this mode, the 11g AP or IBSS will not include an ERP Information Element (IE) or an ESR IE. This mode is supposed to look like an early 802.11b network to allow interoperability with devices that have trouble with any of the newer specification changes. The AP will only advertise and use 802.11b CCK rates. 802.11g clients can still associate but will only operate at 802.11b rates.

The summary of this mode is as follows:

- Only uses 802.11 long-slot timing
- The basic rates are only 1 and 2 Mbps, so all legacy devices can join.
- For 802.11b networks only

17.1.3 LRS

When operating in LRS g-mode, all 802.11g rates are available, but only CCK rates are basic to allow 802.11 devices to join. The rate set is split, with only four rates in the Supported Rates IE to allow interoperability with 802.11 devices that have trouble with more than four rates in the IE. (Take the full 12 rates, throw none away, split the four CCK rates into the first rate element, and put all the Orthogonal Frequency Division Multiplexing (OFDM) rates in the ESR.)

A summary of this mode is as follows:

- Allows both short- and long-slot timings (11b and 11g).
- No 802.11 devices (legacy) can join.
- Supports devices which can only handle 4 CCK rates in the rate set.

17.1.4 Performance

When operating in this gmode, all twelve 802.11g rates are available and the rates are not split. OFDM basic rates are present, so CCK-only (802.11b) devices cannot join.

The rates are not split so that legacy 54g drivers can see all the rates, not just a good subset. Putting more than eight rates in the Supported Rates element does not comply with the 802.11(a, b, or g) specifications, but it works with all Cypress drivers.

Short preambles and short-slot support are required to join. Short-slot operation is always on in the network. Short-slot attention is not given to overlapping BSSs.

Summary:

- Use of 802.11g short-slot timing is mandated.
- Use of 802.11g short preamble is mandated.
- 802.11b clients cannot associate because of the above two mandates.
- Designed to use the maximum bandwidth on 11g-only network topology.

17.2 gmode_protection

Gets or sets gmode protection.

```
wl gmode_protection [on|off]
```

Features:

- Protection mechanisms are enabled automatically when an 11b STA joins the BSS.
- If no 11b STA joins, then no protection mechanisms are used, and full 11g performance is attained.
- The default protection mechanism is not CTS-to-self.
- Typing this command without any parameters displays “gmode_protection is 0 (off)” on the console.

The 802.11g standard uses OFDM to attain its high data speed. To protect 802.11b users, 802.11g is required to send a protection signal that is based on the longer CCK.

Omitting the protection signal ensures high data speeds for 11g users at the cost of locking out 802.11b users.

Without protection, an 802.11b user will be blocked by an invisible flow of 802.11g data and may assume that the wireless network has crashed.

Other considerations include:

- In a mixed environment, that is, ERP and Legacy PHYs (11, 11b and 11g coexisting), protection mechanism use is required.
- ERP-only STAs use a short-slot time to improve performance.
- The 11g spec defines a gmode protection mechanism, which involves prefixing each OFDM data frame with an RTS/CTS CCK frame sequence. The duration fields of the RTS and CTS frames should allow the 11b node to correctly set its NAV and avoid collisions with the subsequent OFDM frames.
- Per spec, the RTS/CTS frames should be sent at one of the basic rates with a CCK-only basic rate set.
- STA behavior automatically honors the bit announced in BSS beacons, so no configuration is needed on the STA side.

Parameters

On or off during a set

Returns

Value	Description
0	Protection mechanisms disabled.
1	Protection mechanisms enabled.

17.3 gmode_protection_control

Gets or sets the IEEE 802.11g protection mode control algorithm.

```
wl gmode_protection_control [<option>]
```

Parameters

option

Value	Description
0	Always off.
1	Monitor local association.
2	Monitor overlapping BSS.

Returns

The protection mode control algorithm as described in the parameters section.

17.4 gmode_protection_cts

Gets or sets the IEEE 802.11g protection type to CTS.

```
wl gmode_protection_cts [on|off]
```

Parameters

on/off

Value	Description
on	Enable CTS protection.
off	Disable CTS protection.

Returns

Whether CTS protection is enabled or disabled.

Value	Description
0	Disabled
1	Enabled

17.5 gmode_protection_override

Gets or sets the IEEE 802.11g protection mode override.

```
wl gmode_protection_override [<option>]
```

There are three modes:

- No protection (default in 54g). This is configured with wl gmode_protection_override 0.
- RTS/CTS when legacy IEEE 802.11b associated. This is configured with:
  - `wl ignore_bcns TRUE (default)`
  - `wl gmode_protection_override -1 (AUTO /default)`
- RTS/CTS when an overlapping legacy IEEE 802.11b device is detected [ignore_bcns = FALSE, may be required for Wi-Fi for IEEE 802.11g]. This is configured with:
  - `wl ignore_bcns FALSE`
  - `wl gmode_protection_override -1 (AUTO /default)`

Parameters

option

Value	Description
–1	Automatic – Protection is automatically used if either an IEEE 802.11b STA associates to the AP or if the AP detects another legacy IEEE 802.11b BSS.
0	Off – Disable protection on the 54g AP such that it will never be used.
1	On – Enable protection on the 54g AP such that it will always be used.

Returns

The IEEE 802.11g protection mode override.

17.6 legacy_erp

Gets or sets the IEEE 802.11g legacy ERP inclusion.

```
wl legacy_erp [<enable>]
```

This command is used to get or set the driver's legacy ERP inclusion flag for non-ERP element advertisement. If the flag is set, then the Legacy ERP Information Element (IE), which is ID 47, is included along with the IEEE 802.11g IE, which is ID 42. The beacon sender shall set bit 0 (NonERP_present) and bit 1 (use_protection) for the use of this element. An ERP STA that is aware of a non-ERP STA shall set bit 0 of the non-ERP IE to TRUE and transmit this information in a subsequent beacon frame.

Parameters

enable

Value	Description
0	Disabled
1	Enabled

Returns

The IEEE 802.11g legacy ERP inclusion flag setting.

Value	Description
0	Disabled
1	Enabled

18. Information Element Control

18.1 add_ie

Adds a vendor-specific IE to management packets.

```
wl add_ie <pktflag> <length> <OUI> <hexdata>
```

Parameters

pktflag

A 4-bit field that indicates packets to which the IE is to be added. The bits are defined as follows:

Value	Description
Bit 0	Beacons
Bit 1	Probe responses
Bit 2	Association and reassociation response
Bit 3	Authorization responses

length

Indicates the IEs byte length.

OUI

The vendors' 3-byte organizationally unique identifier in the format xx:yy:zz.

hexdata

The remaining IE bytes with the pktflag appended to the end.

Returns

None

Example: To add a 10 byte IE to beacons and probe responses with an OUI of 00904C and remaining data of 0101050c121a03, issue the following command to add this IE to beacons and probe responses:

```
wl add_ie 3 10 00:90:4C 0101050c121a03
```

18.2 del_ie

Deletes a vendor-specific IE from management packets.

```
wl del_ie <pktflag> <length> <OUI> <hexdata>
```

Parameters

pktflag

A 4-bit field that indicates packets to which the IE is to be added. The bits are defined as follows:

Value	Description
Bit 0	Beacons
Bit 1	Probe responses
Bit 2	Association and reassociation response
Bit 3	Authorization responses

length

Indicates the IEs byte length.

OUI

The vendor 3-byte organizationally unique identifier in the format xx:yy:zz.

hexdata

The remaining IE bytes with the pktflag appended to the end.

Returns

None

18.3 list_ie

Provides the list of vendor-specific IEs.

```
wl list_ie
```

Parameters

None

Returns

The list of vendor-specific IEs

Revision History

Document Revision History

Revision	Date	Change Description
80211-TI204-R	02/18/14	Updated: ■ By substantially revising many sections, removing several deprecated commands, and adding several new commands.
80211-TI203-R	05/17/12	Updated: ■ Changed the entire document to the new API format. Added: ■ ampdu on page 29. ■ ampdu_tx on page 29. ■ ampdu_rx on page 29. ■ ampdu_mpdu on page 30. ■ mimo_bw_cap on page 30. ■ mimo_txbw on page 30. ■ rtsthresh on page 35. ■ sgi_rx on page 35. ■ sgi_tx on page 36. ■ stbc_rx on page 36. ■ stbc_tx on page 37. ■ roam_off on page 64. ■ gpioout on page 66. ■ phy_txpwrindex on page 73. Removed: ■ Regulatory from Test and Measurements on page 65. ■ lpphy_txpwrindex (BCM4325 Only) (80211-TI202-R, page 60).

Revision	Date	Change Description
80211-TI202-R	04/20/11	Updated: ■ Introduction on page 12 ■ VER on page 13 ■ MPC on page 13 (added) ■ COUNTERS on page 17 ■ RESET_CNTS on page 20 (added) ■ EVENT_MSGS on page 21 ■ STATUS on page 23 ■ PM on page 25 ■ RIFS on page 26 (added) ■ PHY_SCRAMINIT on page 26 (added) ■ NMODE on page 29 ■ PHY_FULCAL (BCM4330 and later) on page 33 (added) ■ CHANSPEC on page 52 ■ PHY_TXPWRINDEX (BCM4330 and later) on page 65
80211-TI201-R	07/30/09	Updated ■ The entire document. Added: ■ Packet Engine Controls¡¨ on page 44. ■ Command Batching¡¨ on page 39.
80211-TI200-R	03/20/07	Initial release

Document Title: WL Tool for Embedded 802.11 Systems CYW43xx Technical Information Document Number: 002-23156				
Revision	**ECN#**	**Issue Date**	**Origin of Change**	**Description of Change**
**			KEMA	Converted to Cypress template