AN14011

Channel State Information (CSI) on Linux OS

Rev. 9.0 — 13 May 2026 **Application note**

Document information

Information	Content
Keywords	AN14011, channel state information, CSI, API, event, mlancsi, configuration files, log files
Abstract	The application note explains how to get the channel state information from a received Orthogonal Frequency Division Multiplexing (OFDM) packet and move the CSI to the host for post-processing.

1 About this document

Channel State Information (CSI) in Wi-Fi is an indication of how the wireless channel affects the signal between two devices. Details such as the signal strength and phase of each subcarrier are captured. CSI can be used to improve the connection quality and to sense the environment.

CSI is the known channel properties of a communication link. CSI is used in Wi-Fi OFDM PHY layers (including 11a/g/n/ac/ax) to achieve reliable communication with high data rates. As such, the Wi-Fi device generates CSI every time it receives a packet seen over-the-air; the firmware and driver allow for filtering and selecting of CSI captures to send to the host.

This document explains how to get the CSI records from the Wi-Fi packets in STA mode. It covers:

- The format of CSI records
- CSI configuration
- Procedure to receive and transmit packets to generate CSI
- Procedure to capture CSI logs
- CSI-enabled sample applications
- Procedure to generate an Ambient Motion Index (AMI), which is a measure of motion

1.1 Supported products

Table 1. Supported Wi-Fi 5 products and features

Product	CSI	AMI
88W8997	Yes	No

Table 2. Supported Wi-Fi 6 products and features

Product	CSI	AMI
88Q9098/88Q9098S	Yes	Yes
88W9098	Yes	Yes
AW690	Yes	Yes
IW620	Yes	No
AW611	Yes	Yes
IW611	Yes	Yes
IW612	Yes	Yes
IW610	Yes	Yes
AW692/AW693/IW693	Yes	Yes

Note: *For optimum performance of the CSI feature, access the i.MX host console via SSH (Ethernet port).*

2 Modules and flow

The host and the wireless device exchange CSI data using:

- mlancsi: application that issues start/stop CSI commands, passes the CSI configuration file to the NXP driver, displays the calculated AMI.
- NXP host driver: applies CSI configurations and executes CSI commands, receives CSI header data through CSI/Netlink events, transfers the headers to mlancsi for processing.
- NXP Wi-Fi firmware: provides the raw CSI data to the NXP driver in the form of CSI/Netlink events.

Upon receiving a packet, the wireless device generates and sends the CSI record to the host.

Note: *From Linux 6.12.49 onward, mlancsi v1.1 is included in the BSP.*

Figure 1 shows the data and control paths used for CSI data transfer between the Wi-Fi device and the host. Refer to Section 9 "Appendix" for details on the host commands and driver event handling.

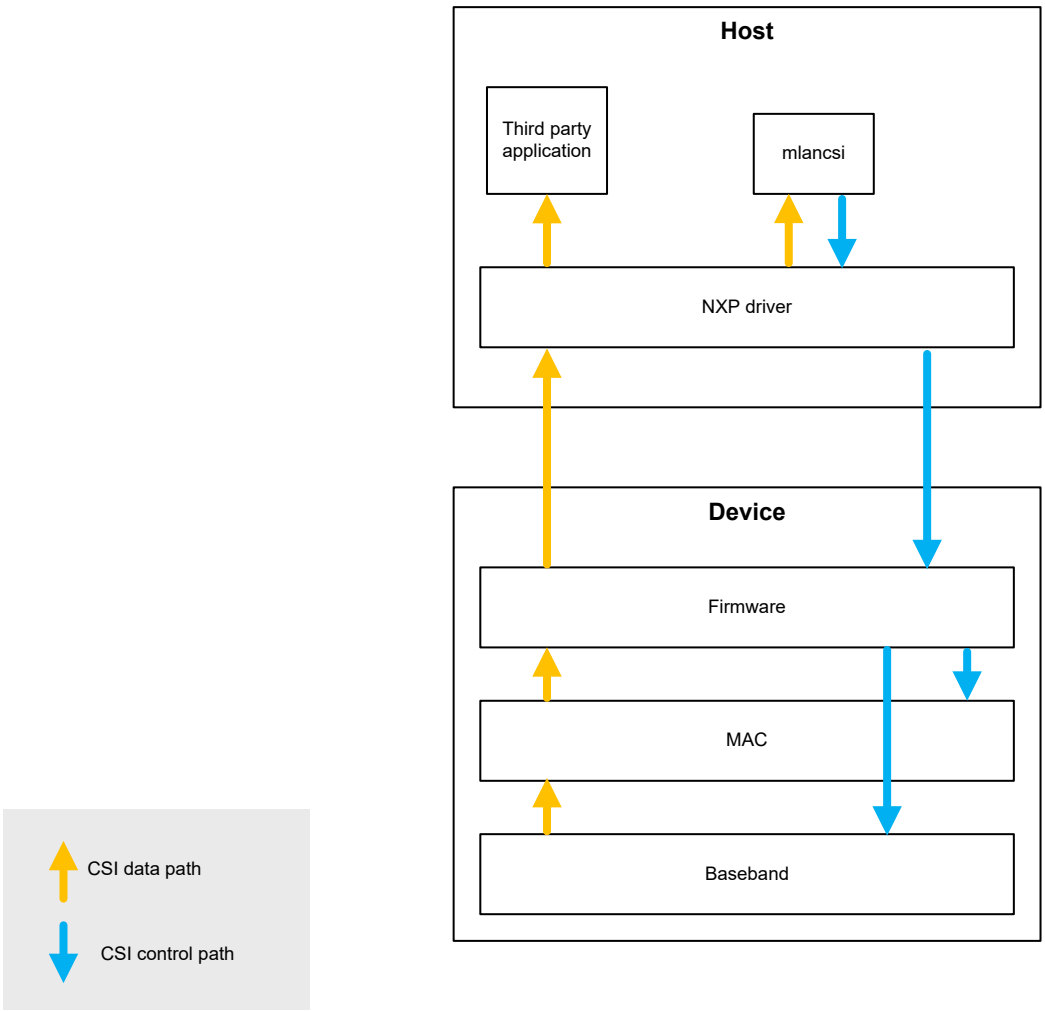

Figure 1. Data path and control path between the host driver and Wi-Fi firmware

3 CSI record

The CSI record includes the CSI header and CSI data.

3.1 CSI record format

Table 3 shows the format of CSI record.

Table 3. CSI record format

Double word	Byte			
	3	2	1	0
0	Signature[15:0]		Length[15:0]	
1	Header Signature ID			
2	PKT_info[31:0]			
3	TSF[63:32]			
4	TSF[31:0]			
5	Dst_MAC[31:0]			
6	Src_MAC[15:0]		Dst_MAC[47:32]	
7	Src_MAC[47:16]			
8	RX_NF_B	RX_NF_A	RX_RSSI_B	RX_RSSI_A
9	Chip ID	AP_TYPE	Channel	SINR
10	RSVD[7:0]	Total Gain	FCF[15:0]	
11	RSVD[15:0]		CSI Data Length[15:0]	
–	CSI Data			
–	Tail Signature ID			

3.2 Field descriptions

Table 4 described the fields of CSI record.

Table 4. Fields of CSI record

Field	Description
Length[15:0]	Actual buffer used (in double word). Indicates the CSI record length that includes CSI header and CSI data.
Signature[15:0]	16-bit signature. Always 0xABCD (not configurable). 0xABCD = indicates the beginning of a CSI record.
Header Signature ID	Header ID: 4-byte user-defined value configurable using CSI configuration file.
PKT_info[31:0]	See Table 5 and Table 6.
TSF[63:0]	Timestamp value from the Timing Synchronization Function (TSF) of the supported product. TSF is the local clock of the Wi-Fi device. The units are microseconds.
Dst_MAC[47:0]	Destination MAC address
SrcMAC[47:0]	Source MAC address
RX_RSSI_A RX_RSSI_B	Received Signal Strength Indication (RSSI): total gain for the whole receiver chain plus average power of CSI data. Signed integer (-128 to +127) in dBm steps. For 1x1 products, RSSI_B is set to 0.
RX_NF_A RX_NF_B	RX Noise Floor (NF): total gain recorded in the receiver chain before the start of a packet. For 1x1 products, RX_NF_B is set to 0.
SINR	Signal to interference noise ratio (SINR). Signed integer (-128 to +127) in dB steps.
Channel	802.11 channel number
AP_TYPE	Type of access point 0x0 = Legacy 0x2 = HT 0x3 = VHT 0x4 = HE Others = Reserved
Chip ID	Chip ID: 1-byte user-defined value configurable using `CSI.conf` configuration file
FCF[15:0]	802.11 Frame Control Field (FCF) or carried FCF in coantrol wrapper
Total Gain	Gain applied to the CSI data in dB
CSI Data Length[15:0]	CSI data length, which is the actual length (in Dwords) + 1. • The length is calculated using the formula: $L = \left\lceil \frac{nTones \times Nr \times Nc}{2} \right\rceil + 1$ Where: Nr = Number of rows in the CSI matrix, equal to the number of receiver antennas. Nc = Number of columns in the CSI matrix, equal to the number of spatial streams. nTones = Determined from the bandwidth and tone group (Ng). Refer to Table 7 for Wi-Fi 5 product category, and Table 8 for Wi-Fi 6 product category.
CSI Data	Refer to Section 3.3 "CSI data format" for CSI data format.
Tail Signature ID	Tail ID: 4-byte user-defined value configurable using `CSI.conf` configuration file

Table 5. PKT_INFO[31:0] signals for Wi-Fi 5 product category

Signal	Description
CSI format [1:0]	Adjust the CSI fixed-point format as: 00 = s8.3 01 = s8.4 10 = s8.5 11 = s8.6
Common AGC flag[0]	Applies to 2x2 products only. Indicates that common AGC is active. That is, the weaker RX path backs off gain to match the gain on the stronger path. 0 = Common AGC not active 1 = Common AGC active
RSVD[12:0]	Reserved
devBW[1:0]	The current bandwidth of the STA. 00 = 20 MHz 01 = 40 MHz 10 = 80 MHz 11 = Reserved
nRx [2:0] or Nr	Number of receiver antennas 000 = 1 001 = 2
nTx[2:0] or Nc	Number of spatial streams (SS) in a given packet. 000 = 1 001 = 2 010 = 3 011 = 4
Ng[0]	Tone grouping (Table 7) 0 = 2 1 = 4
sigBW[1:0]	The Bandwidth (BW) of a given received packet. The value of sigBW must be less than or equal to the value of devBW. 00 = 20 MHz 01 = 40 MHz 10 = 80 MHz 11 = Reserved
Primary Subband (PSB)[2:0]	If devBW[2:0] = 00 (20 MHz), not applicable If devBW[2:0] = 01 (40 MHz): Bit[0] = 0: PSB is -1 Bit[0] = 1: PSB is 1 If devBW[2:0] = 10 (80 MHz): Bit[1:0] = 00: PSB is -2 Bit[1:0] = 01: PSB is -1 Bit[1:0] = 10: PSB is 1 Bit[1:0] = 11: PSB is 2

Table 5. PKT_INFO[31:0] signals for Wi-Fi 5 product category*...continued*

Signal	Description
pktType[1:0]	Packet type 00 = Legacy Orthogonal Frequency Division Multiplexing (OFDM) 01 = High Throughout (HT) 10 = Greenfield (GF)[1] 11 = Very High Throughout (VHT)

[1] Deprecated by WFA.

Table 6. PKT_INFO[31:0] signals for Wi-Fi 6 product category

Signal	Description
CSI format [1:0]	Adjust the CSI fixed-point format as: 00 = s8.3 01 = s8.4 10 = s8.5 11 = s8.6
Common AGC flag[0]	Applies to 2x2 products only. Common Automatic Gain Control (AGC) Indicates that common AGC is active. That is, the weaker RX path backs off gain to match the gain on the stronger path. 0 = Common AGC not active 1 = Common AGC active
RSVD[8:0]	Reserved
HeLTF[1:0]	Long Training Field (LTF) duration 0 = 1x LTF 1 = 2x LTF 2 = 4x LTF
MU[0]	Number of users 0 = Single User (SU) 1 = Multiple Users (MU)
devBW[1:0]	The current bandwidth of the STA. 00 = 20 MHz 01 = 40 MHz 10 = 80 MHz 11 = reserved
nRx [2:0] or Nr	Number of receiver antennas 000 = 1 001 = 2
nTx[2:0] or Nc	Number of Spatial Streams (SS) in a received packet. For each CSI collection, the number can vary based on the SS used. 000 = 1 001 = 2 010 = 3 011 = 4
Reserved[0]	Reserved[0]

Table 6. PKT_INFO[31:0] signals for Wi-Fi 6 product category ***...continued***

Signal	Description
sigBW[1:0]	The BW of a given received packet. The value of sigBW must be less than or equal to the value of devBW. 00 = 20 MHz 01 = 40 MHz 10 = 80 MHz 11 = Reserved
Primary Subband (PSB)[2:0]	If devBW[2:0] = 00 (20 MHz), not applicable If devBW[2:0] = 01 (40 MHz): Bit[0] = 0: PSB is -1 Bit[0] = 1: PSB is 1 If devBW[2:0] = 10 (80 MHz): Bit[1:0] = 00: PSB is -2 Bit[1:0] = 01: PSB is -1 Bit[1:0] = 10: PSB is 1 Bit[1:0] = 11: PSB is 2
pktType[2:0]	Packet type 000 = Legacy Orthogonal Frequency Division Multiplexing (OFDM) 001 = High Throughout (HT) 010 = Greenfield (GF)[1] 011 = Very High Throughout (VHT) 100 = High Efficiency (HE)

[1] Deprecated by WFA.

Table 7. nTones parameter values for Wi-Fi 5 product category and the secondary radio of 88W9098, 88Q9098, and 88Q9098S (mmlan0)

nTones are based on subcarrier indexes for compressed beamforming feedback. Refer to VHT and HT subcarrier and number of matrices/carrier grouping tables in ref.[2].

Bandwidth	Tone grouping (Ng)	nTones
20 MHz	2	30
40 MHz	2	58
80 MHz[1]	4	62

[1] For Wi-Fi 5 product category, maximum CSI data length for 80 MHz 2x4 (two RX antennas receiving data from a 4-antenna device) = 249 Dwords.

Table 8. nTones parameter values for Wi-Fi 6 product category except 88W9098, 88Q9098, 88Q9098S, and IW620 (see Table 9)

nTones are based on subcarrier indexes for compressed beamforming feedback. Refer to VHT, HT, and HE subcarrier and number of matrices/carrier grouping tables in ref.[1] and ref.[2].

Bandwidth	Tone grouping (Ng)	nTones
20 MHz	1 (VHT[1] + legacy)	52
	2 (HT[2])	30
	4 (HE[3])	64
40 MHz	1 (VHT[1] + legacy)	108
	2 (HT[2])	58
	4 (HE[3])	122
80 MHz[4]	1 (VHT[1] + legacy)	234
	4 (HE[3])	250

[1] Subcarrier indexes for which a compressed beamforming feedback matrix is sent back in Table 9-70 in ref.[2].
[2] Number of matrices and carrier grouping in Table 9-54 in ref.[2].
[3] Subcarrier indexes for compressed beamforming feedback matrix in Table 9-91e in ref.[1].
[4] For Wi-Fi 6 product category, maximum CSI data length for 80 MHz 2x4 (2 RX antennas receiving data from a 4-antenna device) = 1,033 Dwords.

Table 9. nTones parameter values for 88W9098, 88Q9098, 88Q9098S primary radio and for IW620

nTones are based on subcarrier indexes for compressed beamforming feedback. Refer to VHT, HT, and HE subcarrier and number of matrices/carrier grouping tables in ref.[1] and ref.[2].

Bandwidth	Tone grouping (Ng)	nTones
20 MHz	1 (VHT[1])	52
	2 (HT[2] + legacy)	30
	4 (HE[3])	64
40 MHz	1 (VHT[1])	108
	2 (HT[2] + legacy)	58
	4 (HE[3])	122
80 MHz[4]	1 (VHT[1])	234
	4 (legacy)	62
	4 (HE[3])	250

[1] Subcarrier indexes for which a compressed beamforming feedback matrix is sent back in Table 9-70 in ref.[2].
[2] Number of matrices and carrier grouping in Table 9-54 in ref.[2].
[3] Subcarrier indexes for compressed beamforming feedback matrix in Table 9-91e in ref.[1].
[4] For Wi-Fi 6 product category, maximum CSI data length for 80 MHz 2x4 (2 RX antennas receiving data from a 4-antenna device) = 1,033 Dwords.

3.3 CSI data format

CSI data is a set of complex values with:

- Nr x Nc complex values for each subcarrier. For example:
  - Nr = 2 and Nc = 1 translates as two complex CSI values per subcarrier.
  - Nr = 1 and Nc = 1 translates as one complex CSI values per subcarrier.
- Each complex value is stored as a 2-byte value: 1 byte for the real part and 1 byte for the imaginary part. The imaginary part is in the format configured in `CSI format [1:0]` signal of PKT_info field signals (s8.3/4/5/6).

Table 10 shows the number of complex CSI values per subcarrier.

Table 10. Number of complex CSI values per subcarrier

	Nc = 1	Nc = 2	Nc = 3	Nc = 4
Nr = 1	1	2	3	4
Nr = 2	2	4	6	8

Figure 2 shows the example of CSI matrix with Nr = 2, Nc = 1, nTones = number of subcarriers = 52.

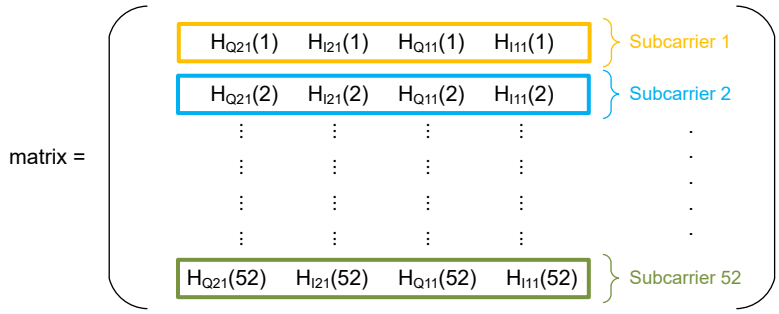

Figure 2. Example of CSI matrix

Where in CSI $matrix_{\alpha\beta y}(\delta)$:

- α: I = real part of the complex value. Q = imaginary part of the complex value
- β: Nr = Nrx = receiver antenna in the order (1, 2, ..., Nr)
- y: Nc = Ntx = spatial streams in the order (1, 2, ..., Nc)
- δ: subcarrier in the order (1, 2, 3, ..., nTones)

Table 11 shows an example of CSI matrix (Nc = 1 and Nr = 2).

Table 11. Example of CSI matrix (Nc = 1 and Nr = 2)

Double word	Byte 3	Byte 2	Byte 1	Byte 0
1	Q21(1)	I21(1)	Q11(1)	I11(1)[1]
2	Q21(2)[2]	I21(2)	Q11(2)	I11(2)
3	...	...	...	...
…	…	…	…	…
52	Q21(52)	I21(52)	Q11(52)	I11(52)

[1] I11(1): I = real part—1 = receiver antenna 1—1 = transmitter antenna 1—(1) = subcarrier 1
[2] Q21(2): Q = imaginary part—2 = receiver antenna 2—1 = transmitter antenna 1—(2) = subcarrier 2

Table 12 shows an example of CSI matrix (Nc = 2 and Nr = 2).

Table 12. Example of CSI matrix (Nc = 2 and Nr = 2)

Double word	Byte 3	Byte 2	Byte 1	Byte 0
1	Q12(1)	I12(1)	Q11(1)	I11(1)
2	Q22(1)	I22(1)	Q21(1)	H21(1)
3	Q12(2)	I12(2)	Q11(2)	I11(2)
4	Q22(2)	I22(2)	Q21(2)	H21(2)
…	…	…	…	…
51	Q12(51)	I12(51)	Q11(51)	I11(51)
52	Q22(52)	I22(52)	Q21(52)	I21(52)

Note: *The CSI data matrix is zero-padded to be Dword aligned.*

4 CSI generation

This section explains the configuration file `csi.conf` and the commands used to generate CSI records. The file `csi.conf` is stored in the `config` directory of the software package and the Linux BSP.

A CSI record is generated each time the Wi-Fi device receives a packet that matches the filters specified in the configuration file. The number of CSI records depends on the time between the start and stop commands (Section 4.1 "Start command" and Section 4.2 "Stop command").

Note: *The `mlancsi` application configuration file and README are not part of the current Linux BSP lf-6.18.2_1.0.0 release. To download them separately, refer to Wireless_Patch_Release_v0.1. Future Linux BSP releases include these components by default.*

Table 13 describes CSI parameters in `csi.conf` file.

Table 13. CSI parameters in `csi.conf` file

Parameter	Description
`HeadID`	User-defined 4-byte ID configurable (in hexadecimal)
`TailID`	User-defined 4-byte ID configurable (in hexadecimal)
`ChipID`	User-defined 1-byte ID configurable (in hexadecimal)
`bandConfig`	1-byte configuration for the bandwidth. Used when `csi_monitor_enable` = 1, to set the channel band and the device bandwidth in CSI monitor mode, when the device is not connected to an AP. Bit[0:1] = channel band 00 = 2.4 GHz 01 = 5 GHz 10 = 6 GHz (only applicable to AW692/AW693) Bit[2:3] = channel width (`chanWidth`) 00 = 20 MHz 10 = 40 MHz 11 = 80 MHz Bit[4:5] = Secondary channel offset (`chan2Offset`) 00 = None 01 = Above 11 = Below Bit[6:7] = Scan mode (`scanMode`) 00 = Manual 01 = Auto Channel Select (ACS) 10 = Adoption mode
`channel`	Channel number in hexadecimal Channel used in CSI monitor mode (`csi_monitor_enable` = 1) when the device is not connected to an AP.
`csi_monitor_enable`	Configures Normal or Monitor mode. 0 = Normal mode (MAC filter enable). CSI is generated from the packets that match the Transmitter Address (TA) configured in one of the CSI filters. 1 = CSI Monitor mode enabled. • If there is a filter with `MAC address` parameter set to 00:00:00:00:00:00, CSI is generated from any packet addressed to the device or broadcast, independent of the TA. It is also used to configure the channel to collect CSI when the device is not connected to an AP.

Table 13. CSI parameters in `csi.conf` file*...continued*

Parameter	Description
	• If there is no filter with `MAC address` parameter set to 00:00:00:00:00:00, the firmware enables software mac filter. CSI is generated from the packets that match the TA configured in one of the CSI filters.

Table 13. CSI parameters in `csi.conf` file*...continued*

Parameter	Description
`ra4us`	Receiver Address (RA) for us. The RA field is part of the MAC header. 0 = Normal mode; generates CSI only if the receiver address (RA) is set to the MAC address of either the device or the broadcast address. 1 = monitor mode; generates CSI independently of the receiver address.
`commonAGCflag`	Applies to 2x2 products only. Common Automatic Gain Control (AGC) Switch the Wi-Fi receiver to common AGC mode (the weaker RX path backs off the gain to match the gain on the stronger path). 0 = do not switch to common AGC mode 1 = switch to common AGC mode during CSI data collection
`csiformat`	CSI fixed-point format. 0 = s8.3 1 = s8.4 2 = s8.5 3 = s8.6 Refer to Section 9.2 "s8.n fixed-point format".
`CSIfilterN`[1]	The maximum number of CSI filters that can be defined is 16. Each 9-byte `CSIfilter` consists of the following fields: **Table 14. `CSIfilter` parameters** Parameter: `Packet Type` — 1-byte value that defines the Wi-Fi packet type (in hexadecimal) 0xFF = firmware does not filter packets using the packet type. Parameter: `Subtype` — 1-byte value that defines the Wi-Fi packet subtype (in hexadecimal) 0xFF = firmware does not filter packets using the packet subtypes. Parameter: `MAC address` — 6-byte value that defines the MAC ID of the device with which the firmware is communicating (in hexadecimal) Parameter: `Otherfilters` — 1 byte (in hexadecimal): • Bit[0] = 0; reserved, must be 0. • Bit[1] = 1; the Wi-Fi firmware waits for a triggering packet before sending a CSI event to the host. • Bit[2] = 1; the Wi-Fi firmware sends a CSI error event when a timeout occurs.

[1] Where N is the CSIfilter number.

Sample of `csi.conf` format:

```
[headID]
00 01 02 03
[/headID]
[tailID]
00 01 02 03
[/tailID]
[chipID]
aa
[/chipID]
[bandConfig]
01
[/bandConfig]
[channel]
24
[/channel]
[csi_monitor_enable]
0
[/csi_monitor_enable]
[ra4us]
1
[/ra4us]
[commonAGCflag]
00
[/commonAGCflag]
[csiformat]
01
[/csiformat]
[csifilter0]
#MAC address
C0 56 27 D6 2E 3C
#Packet Type
01
#Subtype
0D
#Other filter flags
#bit0 reserved, must be 0
#bit1 set to 1: wait for trigger
#bit2 set to 1: send CSI error event when timeout
00
[/csifilter0]
[csifilter1]
#MAC Address
..
#Packtet Type
..
#subtype
..
#Otherfilters
..
[/csifilters1]
[csifilter2]
..
[/csifilter2]
..
..
..
..
..
[csifilter15]
..
[/csifilter15]
```

4.1 Start command

`mlancsi` utility is used to start CSI captures with the parameters set in `csi.conf` file.

```
./mlancsi mlan0 csi config/csi.conf
```

Command output example:

```
-------------------------------------------------
------- NXP Wifi CSI Processing app v1.1 ------
-------------------------------------------------

[INFO] Initializing App
00 01 02 03
00 01 02 03
aa
01
24
01
01
00
01
Found 9 bytes in the csifilter0 section of conf file csi.conf.
c2 95 da 01 12 52 02 ff 00
Found -1 bytes in the csifilter1 section of conf file csi.conf.
Expected filter size is 9
Found 1 CSI filters
```

4.2 Stop command

`mlancsi` utility is used to stop CSI captures.

```
./mlancsi mlan0 csi 0
```

5 CSI dump details

The `csidump.txt` contains:

- All the CSI records are printed in double word text format separated by a space after each double word.
- A signature with:
  - CSI record = 0xABCD
  - Length (in Dwords)
- (CSI header + CSI data) x N, where N = number of CSI records
- File size limit = 1.2 MB (1000 records when bandwidth = 80 MHz, 2x2)

`csidump.txt` contents for a single CSI record:

```
----------------------------------------
            EVENT: MLAN_CSI Sequence: 458 Length: 28 DWORDs
            CSI record:
            abcd601c 03020100 40000001 94056699 00000000 01da95c0 5ea82a12 10f6ae45
            009d00d7 aa020b3a 00264a88 05000010 fad4ebf6 11f516da eefffa01 fcf1f1fb
            12d307e1 0fea16d7 f50601fe eefaf104 e0e4e9ee e6dfdfdd 01fff8ec e806f70a
            d9efe0fe f6d9dce6 0ffe12ef 03020100
```

Decoding of CSI record:

- `abcd` = signature
- `0041` = CSI record length in hexadecimal
- `03020100` = HeaderID same as in `csi.conf` file
- `00001000` = Packet_INFO in hexadecimal (refer to Table 5 and Table 6)
- `7d7b0414 00000000` = TSF fields
- `ffffffffffff` = Destination MAC for the CSI packets
- `4322a3d14250` = Source MAC ID of CSI packets
- `009c00c9` = `0xc9`: RX_RSSI_A, `0x00`: RX_RSSI_B, `0x9c`: RX_NF_A, `0x00`: RX_NF_B
- `aa00642d` = `0x2d`: SINR, `0x64`: channel, `0x00`: AP_type, `0xaa`: chip ID
- `00000080` = `0x0080`: FCF, `0x00`: total gain, `0x00`: reserved
- `00000035` = `0x0035`: CSI data length, `0x0000`: reserved
- `1306fd04 1306fd04 1306fd04 0610fd00 fa0eff00 f505ff02 f6fffb03 faf9faf8 fdf601f4 03f509f7 0af80e00 0f010a0b 0c0c0010 0112f40d f40fed01 eb04f0f4 eef5fbee f9ed07f0 06ee0ef9 0ef70d03 08070109 0208fe07 fe06fb05 fd04fa04 fd04f801 fa04f7fd 00000000 f9f505f5 02f10bfa 0cf50c02 1100080a 0d0c000c f50ff604 ee03f6ff f1f7f8fb fbf1faf9 06f3fef7 0bfc02f6 090508f8 01080bfe fa050c05 f9fe070d fdfafd10 04faf10c 0701eb00 fb09fce6 f4010fe9 f6f71cfb 01f11811 0cf8021d 0cf8021d 0cf8021d` = CSI data
- `03020100` = TailID

6 Processing CSI – Ambient Motion Index

Following the CSI collection, the CSI records can be processed to calculate the AMI. AMI is a measure of change in CSI that is used to detect motion near the Wi-Fi STA and/or AP. AMI is expressed in dB.

Note: *Verify that the product supports CSI in Section 1.1 "Supported products".*

The configuration file `mlancsi.conf` is used for AMI calculation. The file `csi.conf` is stored in the config directory of the software package and the Linux BSP. Refer to README_MLANCSI for more information. Table 15 shows the `mlancsi.conf` parameters.

Table 15. mlancsi.conf parameters

Parameter	Description
`MAC_ADDR`	Source MAC address of the packets to process CSI from to generate AMI. Matches one of the CSIfilterN MAC addresses defined in `csi.conf`.
`PACKET_TYPE`	Select the packet type to process: 0 = Legacy (11a/g) 1 = HT (11n) 2 = VHT (11ac) 3 = HE (11ax)
`FORMAT_BW`	Select the bandwidth to process: 0 = 20 MHz 1 = 40 MHz 2 = 80 MHz
`UPDATE_REF`	Reference update 0 = Static (uses first CSI record with no updates) 1 = Infinite Impulse Response (IIR) filter (initializes the reference with the first CSI record and updates the reference with the IIR filter using IIR filter coefficient alpha) 2 = Kalman filter (uses the first CSI record as reference and updates the record with the Kalman filter)
`IIR_ALPHA`	IIR filter coefficient Range = 0 to 1 in decimal
`KALMAN_P0`	Kalman initial uncertainty value
`KALMAN_ALPHA`	Kalman system noise per time
`KALMAN_NO`	Kalman measurement uncertainty
`NUM_CSI`	Number of CSI records to use: 0 = runs until mlancsi app is stopped Range = 1 to 255 (runs for this many CSI records)

Example of `mlancsi.conf` content:

```
CSI_EVENT_CFG={
# Source Address of packets to process CSI from to generate Ambient Motion Index
MAC_ADDR=3c:51:0e:6f:f3:6e
# Packet type
# 0: legacy - 11a/g
# 1: HT         - 11n
# 2: VHT        - 11ac
# 3: HE         - 11ax
PACKET_TYPE=0
# 0: 20MHz, 1:40MHz , 2:80MHz
FORMAT_BW=0
# Reference Update
# 0: "static"           - frst , no updates
# 1: "IIR filter"       - first CSI is reference, update with IIR fitler coefficient
 alpha
# 2: "Kalman filter"- first CSI is reference, update with Kalaman fitler
UPDATE_REF=1
# IIR update filter H_ref_n = (1-alpha)*H_ref_{n-1) + alpha*H_n
# values within [0,1]
IIR_ALPHA=0.1
# Kalman filter parameters
# Kalman filter update step:
# H_ref_n = (1-K_n)*H_ref_{n-1) + K_n*H_n
# where
# kp_0 = KALMAN_P0 - initiatlization
# p_n = kp_{n-1} + KALMAN_ALPHA*delta_time
# S_n = p_n + KALMAN_N0
# K_n = p_n / S_n
# kp_n = (1 - K_n) * p_n
# delta_time in seconds
KALMAN_P0=0.5
KALMAN_ALPHA=0.005
KALMAN_N0=0.2
# Number of CSI
# 0:    default, will run until app is killed
# 1-255:will run for this many CSI records
NUM_CSI=55
}
```

6.1 Start calculating AMI

mlancsi application is used to calculate the AMI based on `mlancsi.conf`.

```
./mlancsi mlan0 event config/mlancsi.conf
```

Table 16. Command return parameters

Parameter	Description
`Packet type`	Reflects the choice of packet type and bandwidth in `mlancsi.conf` (Table 15)
`RX/TX antennas`	Number of receiver antennas (RX or Nr) and number of spatial streams (TX or Nc)
`TSF`	Timestamp in µs
`Ambient Motion Index`	AMI value in dB Range = negative number to 0 A small AMI value means that little motion is detected. A higher AMI value means that more significant motion is detected.

Command output example:

```
-------------------------------------------------
------- NXP Wifi CSI Processing app v1.1 ------
-------------------------------------------------

[INFO] Initializing App
[INFO] Read CSI config from file mlancsi.conf
CSI_EVENT_CFG
         MAC_ADDR=c2:95:da:01:12:52
         PACKET_TYPE=3
         FORMAT_BW=2
         UPDATE_REF=0
         IIR_ALPHA=0.100000
         KALMAN_P0=0.500000
         KALMAN_ALPHA=0.005000
         KALMAN_N0=0.200000
         NUM_CSI=0
[INFO] Netlink number = 31
Compare CSI filter set MAC: c2.95.da.01.12.52, sig BW/format 2|3
CSI Processing results: HE(80), Rx/Tx 1/1, 130.83       TSF 61fe5d6d97, Ambient Motion
 Index -14.2 dB
CSI Processing results: HE(80), Rx/Tx 1/1, 127.26       TSF 61fe6d0453, Ambient Motion
 Index -16.8 dB
CSI Processing results: HE(80), Rx/Tx 1/1, 129.52       TSF 61fe7ca45c, Ambient Motion
 Index -14.9 dB
CSI Processing results: HE(80), Rx/Tx 1/1, 130.75       TSF 61fe8c4b41, Ambient Motion
 Index -14.5 dB
CSI Processing results: HE(80), Rx/Tx 1/1, 131.15       TSF 61fe9be45d, Ambient Motion
 Index -14.8 dB
CSI Processing results: HE(80), Rx/Tx 1/1, 126.83       TSF 61fea9fda6, Ambient Motion
 Index -15.3 dB
CSI Processing results: HE(80), Rx/Tx 1/1, 125.64       TSF 61feb994a8, Ambient Motion
 Index -16.5 dB
CSI Processing results: HE(80), Rx/Tx 1/1, 128.54       TSF 61ff07bb72, Ambient Motion
 Index -16.8 dB
CSI Processing results: HE(80), Rx/Tx 1/1, 130.31       TSF 61ff07c70f, Ambient Motion
 Index -18.4 dB
CSI Processing results: HE(80), Rx/Tx 1/1, 132.13       TSF 61ff07d43f, Ambient Motion
 Index -14.1 dB
CSI Processing results: HE(80), Rx/Tx 1/1, 130.74       TSF 61ff17588a, Ambient Motion
 Index -16.5 dB
CSI Processing results: HE(80), Rx/Tx 1/1, 132.06       TSF 61ff256456, Ambient Motion
 Index -15.2 dB
```

```
CSI Processing results: HE(80), Rx/Tx 1/1, 129.47       TSF 61ff3504bd, Ambient Motion
 Index -14.5 dB
CSI Processing results: HE(80), Rx/Tx 1/1, 130.74       TSF 61ff44a47d, Ambient Motion
 Index -15.6 dB
CSI Processing results: HE(80), Rx/Tx 1/1, 128.92       TSF 61ff544435, Ambient Motion
 Index -16.7 dB
```

7 Examples

This section describes several examples of how to configure the parameters in the `csi.conf` configuration file to generate CSI data. A CSI record is generated each time the Wi-Fi device receives a packet that matches the filters specified in the configuration file. The number of CSI records depends on the time between the `start` and `stop` commands (refer to Section 4.1 "Start command" and Section 4.2 "Stop command").

Note: *The `mlancsi` application configuration file and README are not part of the current Linux BSP lf-6.18.2_1.0.0 release. To download them separately, refer to Wireless_Patch_Release_v0.1. Future Linux BSP releases include these components by default.*

7.1 Beacons

This example generates CSI data for every beacon received.

Note: *CSI data is generated from OFDM packets therefore, you cannot use 802.11b beacons.*

Step 1 – Bring up Wi-Fi.

Example of driver and firmware loading:

```
modprobe moal mod_para=/nxp/wifi_mod_para.conf
```

Step 2 – Configure the device in STA mode and connect with the external AP.

Step 3 – Edit `csi.conf` to filter for beacons.

Use the following:

- `MAC address = C0 56 27 D6 2E 3C` // MAC address of external AP
- `Packet type = 00` // management packet
- `Subtype = 08` // beacon
- `commonAGCflag = 00` // do not switch to common AGC mode
- `csiformat = 00` // s8.3 fixed-point format
- `Other filter flags = 00` // no other filters are applied

```
[headID]
00 01 02 03
[/headID]
[tailID]
00 01 02 03
[/tailID]
[chipID]
aa
[/chipID]
[bandConfig]
01
[/bandConfig]
[channel]
24
[/channel]
[csi_monitor_enable]
0
[/csi_monitor_enable]
[ra4us]
1
[/ra4us]
[commonAGCflag]
00
[/commonAGCflag]
[csiformat]
00
[/csiformat]
[csifilter0]
#MAC address
C0 56 27 D6 2E 3C
#Packet Type
00
#Subtype
08
#Other filter flags
#bit0 reserved, must be 0
#bit1 set to 1: wait for trigger
#bit2 set to 1: send CSI error event when timeout
00
[/csifilter0]
```

Step 4 – Enable CSI.

```
./mlancsi mlan0 csi config/csi.conf
```

Step 5 – Execute `mlancsi` to start the CSI collection in the file.

```
./mlancsi mlan0 event dump_to_file 1
```

Step 6 – Stop CSI.

```
./mlancsi mlan0 csi 0
```

7.2 Management packets

This example uses received management packets to generate CSI data. The rate at which the CSI is generated depends on the particular packet type being filtered.

Step 1 – Bring up Wi-Fi.

Example of driver and firmware loading:

```
modprobe moal mod_para=/nxp/wifi_mod_para.conf
```

Step 2 – Configure the device in STA mode and connect with the external AP.

Step 3 – Edit `csi.conf` for management packets.

- `MAC address = C0 56 27 D6 2E 3C` // MAC address of external AP
- `Packet type = 00` // management packet
- `Subtype = FF` // no subtype filter applied
- `commonAGCflag = 00` // do not switch to common AGC mode
- `csiformat = 01` // 8.4 fixed-point format
- `Other filter flags = 00` // no other filters are applied

```
[headID]
00 01 02 03
[/headID]
[tailID]
00 01 02 03
[/tailID]
[chipID]
aa
[/chipID]
[bandConfig]
01
[/bandConfig]
[channel]
24
[/channel]
[csi_monitor_enable]
0
[/csi_monitor_enable]
[ra4us]
1
[/ra4us]
[commonAGCflag]
00
[/commonAGCflag]
[csiformat]
01
[/csiformat]
[csifilter0]
#MAC address
C0 56 27 D6 2E 3C
#Packet Type
00
#Subtype
FF
#Other filter flags
#bit0 reserved, must be 0
#bit1 set to 1: wait for trigger
#bit2 set to 1: send CSI error event when timeout
00
[/csifilter0]
```

Step 4 – Enable CSI.

```
./mlancsi mlan0 csi config/csi.conf
```

Step 5 – Execute `mlancsi` to start CSI collection in the file.

```
./mlancsi mlan0 event dump_to_file 1
```

Step 6 – Stop CSI.

```
./mlancsi mlan0 csi 0
```

7.3 Data packets

This example uses received data packets to generate CSI data. By default, pinging generates one CSI record per second, which is also configurable.

Step 1 – Bring up Wi-Fi.

Example of driver and firmware loading:

```
modprobe moal mod_para=/nxp/wifi_mod_para.conf
```

Step 2 – Configure the device in STA mode. Connect with the external AP and get the IP address.

Step 3 – Edit `csi.conf` for data packets.

- `MAC address = C0 56 27 D6 2E 3C` // MAC address of external AP
- `Packet type = 02` // data packet
- `Subtype = FF` // no subtype filter applied
- `commonAGCflag = 00` // do not switch to common AGC mode
- `csiformat = 01` // 8.4 fixed-point format
- `Other filter flags = 00` // no other filters are applied

```
[headID]
00 01 02 03
[/headID]
[tailID]
00 01 02 03
[/tailID]
[chipID]
aa
[/chipID]
[bandConfig]
01
[/bandConfig]
[channel]
24
[/channel]
[csi_monitor_enable]
0
[/csi_monitor_enable]
[ra4us]
1
[/ra4us]
[commonAGCflag]
00
[/commonAGCflag]
[csiformat]
01
[/csiformat]
[csifilter0]
#MAC address
C0 56 27 D6 2E 3C
#Packet Type
02
#Subtype
FF
#Other filter flags
#bit0 reserved, must be 0
#bit1 set to 1: wait for trigger
#bit2 set to 1: send CSI error event when timeout
00
[/csifilter0]
```

Step 4 – Enable CSI.

```
./mlancsi mlan0 csi config/csi.conf
```

Step 5 – Execute `mlancsi` to start the CSI collection in the file.

```
./mlancsi mlan0 event dump_to_file 1
```

Step 6 – Run ping from AP to STA.

Step 7 – Stop CSI.

```
./mlancsi mlan0 csi 0
```

7.4 Connectionless/monitor mode

The STA generates CSI data from the management packets received in a connectionless state. The parameter `csi_monitor_enable` must be enabled in `csi.conf`.

Step 1 – Load the wireless SoC with the drivers and firmware.

```
modprobe moal.ko mod_para=/nxp/wifi_mod_para.conf
```

Step 2 – Edit `csi.conf`.

- `csi_monitor_enable = 1` // CSI monitor mode
- `MAC address = 3C:51:0E:6F:F3:6E` // MAC address of external AP
- `Packet type` = 00 // management packet type
- `Subtype = FF` // no filter
- `commonAGCflag = 00` // do not switch to common AGC mode
- `csiformat = 01` // s8.4 fixed-point format
- `Other filter flags = 00` // no other filters are applied

```
[headID]
00 01 02 03
[/headID]
[tailID]
00 01 02 03
[/tailID]
[chipID]
aa
[/chipID]
[bandConfig]
01
[/bandConfig]
[channel]
7C
[/channel]
[csi_monitor_enable]
1
[/csi_monitor_enable]
[ra4us]
1
[/ra4us]
[commonAGCflag]
00
[/commonAGCflag]
[csiformat]
01
[/csiformat]
[csifilter0]
#MAC address
3C 51 0E 6F F3 6E
#Packet Type
00
#Subtype
FF
#Other filter flags
#bit0 reserved, must be 0
#bit1 set to 1: wait for trigger
#bit2 set to 1: send CSI error event when timeout
00
[/csifilter0]
```

Step 3 – Enable CSI.

```
./mlancsi mlan0 csi config/csi.conf
```

Step 4 – Execute `mlancsi` to start CSI collection in the file.

```
./mlancsi mlan0 event dump_to_file 1
```

Step 5 – Stop CSI.

```
./mlancsi mlan0 csi 0
```

7.5 Multi-APs

The STA generates CSI data from multiple AP beacons received in a connectionless state. Ensure that the parameter `csi_monitor_enable` is enabled in `csi.conf`.

Step 1 – Load the wireless SoC with the drivers and firmware.

```
modprobe moal.ko mod_para=/nxp/wifi_mod_para.conf
```

Step 2 – Edit `csi.conf`.

Note: *Setting the MAC address to 00:00:00:00:00:00 enables CSI data collection from beacons sent by all nearby access points. Specifying a MAC address restricts CSI data to beacons from that particular access point.*

- `csi_monitor_enable = 1` // CSI monitor mode
- `MAC address = 00:00:00:00:00:00`
- `Packet type = 00` // management packet type
- `Subtype = 08` // beacon
- `commonAGCflag = 00` // do not switch to common AGC mode
- `csiformat = 01` // s8.4 fixed-point format
- `Other filter flags = 00` // no other filters are applied

```
[headID]
00 01 02 03
[/headID]
[tailID]
00 01 02 03
[/tailID]
[chipID]
aa
[/chipID]
[bandConfig]
01
[/bandConfig]
[channel]
7C
[/channel]
[csi_monitor_enable]
1
[/csi_monitor_enable]
[ra4us]
1
[/ra4us]
[commonAGCflag]
00
[/commonAGCflag]
[csiformat]
01
[/csiformat]
[csifilter0]
#MAC address
00 00 00 00 00 00
#Packet Type
00
#Subtype
08
#Other filter flags
#bit0 reserved, must be 0
#bit1 set to 1: wait for trigger
#bit2 set to 1: send CSI error event when timeout
00
[/csifilter0]
```

Step 3 – Enable CSI.

```
./mlancsi mlan0 csi config/csi.conf
```

Step 4 – Execute `mlancsi` to start the CSI collection in the file.

```
./mlancsi mlan0 event dump_to_file 1
```

Step 6 – Stop CSI.

```
./mlancsi mlan0 csi 0
```

7.6 Multi-clients

The uAP generates CSI data from data packets received from two clients.

Step 1 – Load the wireless SoC with the drivers and firmware.

```
modprobe moal.ko mod_para=/nxp/wifi_mod_para.conf
```

Step 2 – Configure the device in uAP mode and associate with two clients.

Step 3 – Edit `csi.conf`.

Parameters of `csifilter0`:

- `MAC address = 4E:2C:ED:B9:47:27` // MAC address of client1
- `Packet type = 02` // data packet
- `Subtype = FF` // no filter
- `commonAGCflag = 00` // do not switch to common AGC mode
- `csiformat = 01` // s8.4 fixed-point format
- `Other filter flags = 00` // no other filters are applied

Parameters of `csifilter1`:

- `MAC address = D4:54:8B:61:8E:8D` // MAC address of client2
- `Packet type = 02` // data packet type
- `Subtype = FF` // no filter
- `commonAGCflag = 00` // do not switch to common AGC mode
- `csiformat = 01` // s8.4 fixed-point format
- `Other filter flags = 00` // no other filters are applied

```
[headID]
00 01 02 03
[/headID]
[tailID]
00 01 02 03
[/tailID]
[chipID]
aa
[/chipID]
[bandConfig]
01
[/bandConfig]
[channel]
24
[/channel]
[csi_monitor_enable]
1
[/csi_monitor_enable]
[ra4us]
1
[/ra4us]
[commonAGCflag]
00
[/commonAGCflag]
[csiformat]
01
[/csiformat]
[csifilter0]
#MAC address
4E 2C ED B9 47 27
#Packet Type
02
#Subtype
FF
#Other filter flags
#bit0 reserved, must be 0
#bit1 set to 1: wait for trigger
#bit2 set to 1: send CSI error event when timeout
00
[/csifilter0]
[csifilter1]
#MAC address
D4 54 8B 61 8E 8D
#Packet Type
02
#Subtype
FF
#Other filter flags
#bit0 reserved, must be 0
#bit1 set to 1: wait for trigger
#bit2 set to 1: send CSI error event when timeout
00
[/csifilter1]
```

Step 4 – The external STAs ping the uAP.

Step 5 – Enable CSI.

```
./mlancsi uap0 csi config/csi.conf
```

Step 6 – Execute `mlancsi` to start the CSI collection in the file.

```
./mlancsi mlan0 event dump_to_file 1
```

Step 6 – Stop CSI.

```
./mlancsi uap0 csi 0
```

7.7 Ambient motion detection

In this example, AMI is calculated for two cases:

- No motion
- Motion

7.7.1 No motion

The low AMI value is displayed to indicate that there is little to no motion near the STA.

Step 1 – Place the STA in an empty room and connect with external AP.

Step 2 – Configure CSI collection via beacons, management packets, data packets, and so on. Refer to Section 4 "CSI generation ".

Step 3 – Enable CSI.

```
./mlancsi mlan0 csi config/csi.conf
```

Step 4 – Configure AMI in `mlancsi.conf`.

Example of `mlancsi.conf` content:

```
CSI_EVENT_CFG={
# Source Address of packets to process CSI from to generate Ambient Motion Index
MAC_ADDR=3c:51:0e:6f:f3:6e
# Packet type
# 0: legacy - 11a/g
# 1: HT         - 11n
# 2: VHT        - 11ac
# 3: HE         - 11ax
PACKET_TYPE=0
# 0: 20MHz, 1:40MHz , 2:80MHz
FORMAT_BW=0
# Reference Update
# 0: "static"           - frst , no updates
# 1: "IIR filter"       - first CSI is reference, update with IIR fitler coefficient
 alpha
# 2: "Kalman filter"- first CSI is reference, update with Kalaman fitler
UPDATE_REF=1
# IIR update filter H_ref_n = (1-alpha)*H_ref_{n-1) + alpha*H_n
# values within [0,1]
IIR_ALPHA=0.1
# Kalman filter parameters
# Kalman filter update step:
# H_ref_n = (1-K_n)*H_ref_{n-1) + K_n*H_n
# where
# kp_0 = KALMAN_P0 - initiatlization
# p_n = kp_{n-1} + KALMAN_ALPHA*delta_time
# S_n = p_n + KALMAN_N0
# K_n = p_n / S_n
# kp_n = (1 - K_n) * p_n
# delta_time in seconds
KALMAN_P0=0.5
KALMAN_ALPHA=0.005
KALMAN_N0=0.2
# Number of CSI
# 0:    default, will run until app is killed
# 1-255:will run for this many CSI records
NUM_CSI=55
}
```

Step 5 – Start calculating the AMI to start CSI collection in `csidump.txt`.

```
./mlancsi mlan0 event config/mlancsi.conf dump_to_file
```

Example of command output:

```
--------------------------------------------------
------- NXP Wifi CSI Processing app v1.1 ------
--------------------------------------------------

[INFO] Initializing App
[INFO] Read CSI config from file mlancsi.conf
CSI_EVENT_CFG
         MAC_ADDR=c2:95:da:01:12:52
         PACKET_TYPE=3
         FORMAT_BW=2
         UPDATE_REF=0
         IIR_ALPHA=0.100000
         KALMAN_P0=0.500000
         KALMAN_ALPHA=0.005000
         KALMAN_N0=0.200000
         NUM_CSI=0
[INFO] Netlink number = 31
Compare CSI filter set MAC: c2.95.da.01.12.52, sig BW/format 2|3
CSI Processing results: HE(80), Rx/Tx 1/1, 130.83       TSF 61fe5d6d97, Ambient Motion
 Index -14.2 dB
CSI Processing results: HE(80), Rx/Tx 1/1, 127.26       TSF 61fe6d0453, Ambient Motion
 Index -16.8 dB
CSI Processing results: HE(80), Rx/Tx 1/1, 129.52       TSF 61fe7ca45c, Ambient Motion
 Index -14.9 dB
CSI Processing results: HE(80), Rx/Tx 1/1, 130.75       TSF 61fe8c4b41, Ambient Motion
 Index -14.5 dB
CSI Processing results: HE(80), Rx/Tx 1/1, 131.15       TSF 61fe9be45d, Ambient Motion
 Index -14.8 dB
CSI Processing results: HE(80), Rx/Tx 1/1, 126.83       TSF 61fea9fda6, Ambient Motion
 Index -15.3 dB
CSI Processing results: HE(80), Rx/Tx 1/1, 125.64       TSF 61feb994a8, Ambient Motion
 Index -16.5 dB
CSI Processing results: HE(80), Rx/Tx 1/1, 128.54       TSF 61ff07bb72, Ambient Motion
 Index -16.8 dB
CSI Processing results: HE(80), Rx/Tx 1/1, 130.31       TSF 61ff07c70f, Ambient Motion
 Index -18.4 dB
CSI Processing results: HE(80), Rx/Tx 1/1, 132.13       TSF 61ff07d43f, Ambient Motion
 Index -14.1 dB
CSI Processing results: HE(80), Rx/Tx 1/1, 130.74       TSF 61ff17588a, Ambient Motion
 Index -16.5 dB
CSI Processing results: HE(80), Rx/Tx 1/1, 132.06       TSF 61ff256456, Ambient Motion
 Index -15.2 dB
CSI Processing results: HE(80), Rx/Tx 1/1, 129.47       TSF 61ff3504bd, Ambient Motion
 Index -14.5 dB
CSI Processing results: HE(80), Rx/Tx 1/1, 130.74       TSF 61ff44a47d, Ambient Motion
 Index -15.6 dB
CSI Processing results: HE(80), Rx/Tx 1/1, 128.92       TSF 61ff544435, Ambient Motion
 Index -16.7 dB
```

The AMI outputs range from -14 dB to -16 dB, which is considered small AMI. No motion is detected.

7.7.2 Motion

The high AMI value is displayed to indicate some motion near the STA.

Step 1 – Configure CSI collection via beacons, management packets, data packets, and so on. Refer to Section 4 "CSI generation ".

Step 2 – Enable CSI.

```
./mlancsi mlan0 csi config/csi.conf
```

Step 3 – Configure AMI in `mlancsi.conf`.

Example of `mlancsi.conf` content:

```
CSI_EVENT_CFG={
# Source Address of packets to process CSI from to generate Ambient Motion Index
MAC_ADDR=3c:51:0e:6f:f3:6e
# Packet type
# 0: legacy - 11a/g
# 1: HT         - 11n
# 2: VHT        - 11ac
# 3: HE         - 11ax
PACKET_TYPE=0
# 0: 20MHz, 1:40MHz , 2:80MHz
FORMAT_BW=0
# Reference Update
# 0: "static"           - frst , no updates
# 1: "IIR filter"       - first CSI is reference, update with IIR fitler coefficient
 alpha
# 2: "Kalman filter"- first CSI is reference, update with Kalaman fitler
UPDATE_REF=1
# IIR update filter H_ref_n = (1-alpha)*H_ref_{n-1) + alpha*H_n
# values within [0,1]
IIR_ALPHA=0.1
# Kalman filter parameters
# Kalman filter update step:
# H_ref_n = (1-K_n)*H_ref_{n-1) + K_n*H_n
# where
# kp_0 = KALMAN_P0 - initiatlization
# p_n = kp_{n-1} + KALMAN_ALPHA*delta_time
# S_n = p_n + KALMAN_N0
# K_n = p_n / S_n
# kp_n = (1 - K_n) * p_n
# delta_time in seconds
KALMAN_P0=0.5
KALMAN_ALPHA=0.005
KALMAN_N0=0.2
# Number of CSI
# 0:    default, will run until app is killed
# 1-255:will run for this many CSI records
NUM_CSI=55
}
```

Step 4 – Walk back and forth by the STA.

Step 5 – Start calculating the AMI to start CSI collection in `csidump.txt`.

```
mlancsi mlan0 event config/mlancsi.conf dump_to_file
```

Example of command output:

```
------------------------------------------------
------- NXP Wifi CSI Processing app v1.1 ------
------------------------------------------------

[INFO] Initializing App
[INFO] Read CSI config from file mlancsi.conf
CSI_EVENT_CFG
         MAC_ADDR=c2:95:da:01:12:52
         PACKET_TYPE=3
         FORMAT_BW=2
         UPDATE_REF=0
         IIR_ALPHA=0.100000
         KALMAN_P0=0.500000
         KALMAN_ALPHA=0.005000
         KALMAN_N0=0.200000
         NUM_CSI=0
[INFO] Netlink number = 31
Compare CSI filter set MAC: c2.95.da.01.12.52, sig BW/format 2|3
CSI Processing results: HE(80), Rx/Tx 1/1, 125.20       TSF 62bcd014a2, Ambient Motion
 Index -8.8 dB
CSI Processing results: HE(80), Rx/Tx 1/1, 118.00       TSF 62bcdfc101, Ambient Motion
 Index -5.1 dB
CSI Processing results: HE(80), Rx/Tx 1/1, 132.06       TSF 62bcef5573, Ambient Motion
 Index -9.5 dB
CSI Processing results: HE(80), Rx/Tx 1/1, 136.69       TSF 62bcfef49a, Ambient Motion
 Index -9.4 dB
CSI Processing results: HE(80), Rx/Tx 1/1, 94.89        TSF 62bd0e9af7, Ambient Motion
 Index -4.3 dB
CSI Processing results: HE(80), Rx/Tx 1/1, 122.67       TSF 62bd5c3f48, Ambient Motion
 Index -6.6 dB
CSI Processing results: HE(80), Rx/Tx 1/1, 136.78       TSF 62bd5c51bc, Ambient Motion
 Index -6.7 dB
CSI Processing results: HE(80), Rx/Tx 1/1, 136.63       TSF 62bd6acd91, Ambient Motion
 Index -7.5 dB
CSI Processing results: HE(80), Rx/Tx 1/1, 135.94       TSF 62bd7a6468, Ambient Motion
 Index -6.4 dB
CSI Processing results: HE(80), Rx/Tx 1/1, 149.37       TSF 62bd8a0493, Ambient Motion
 Index -3.9 dB
CSI Processing results: HE(80), Rx/Tx 1/1, 131.97       TSF 62bd981f24, Ambient Motion
 Index -9.9 dB
CSI Processing results: HE(80), Rx/Tx 1/1, 133.53       TSF 62bda7b484, Ambient Motion
 Index -10.2 dB
CSI Processing results: HE(80), Rx/Tx 1/1, 130.42       TSF 62bdb75c98, Ambient Motion
 Index -12.3 dB
CSI Processing results: HE(80), Rx/Tx 1/1, 143.04       TSF 62bdc6f4e3, Ambient Motion
 Index -7.6 dB
CSI Processing results: HE(80), Rx/Tx 1/1, 135.45       TSF 62bdd5105a, Ambient Motion
 Index -10.2 dB
```

The AMI outputs range from -3 dB to -10 dB, which corresponds to a larger AMI. A motion is detected.

8 Sample applications

Matlab, Python, or similar tools can be used to process CSI data records. The following are the examples of applications using the CSI data.

8.1 Amplitude and phase graphs

To get the graphs:

Step 1 - Identify the nRx, nTx, and nTones values from the CSI records (Section 3.1 "CSI record format").

Step 2 - Get r, t, and k.

- r = nRx
- t = nTx
- k = nTones

Step 3 - Calculate the gain in dB.

$$G_{dB}^{(rt)}(k) = 10\,log_{10}\left((H_{lrt}(k))^2 + (H_{Qrt}(k))^2\right)$$

Step 4 - Calculate the phase in radians.

$$\varphi_{rad}^{(rt)}(k) = tan^{-1}\left(\frac{H_{Qrt}(k)}{H_{lrt}(k)}\right)$$

Figure 3 shows the relative gain at the reported subcarriers (in dB) for both receiver antennas.

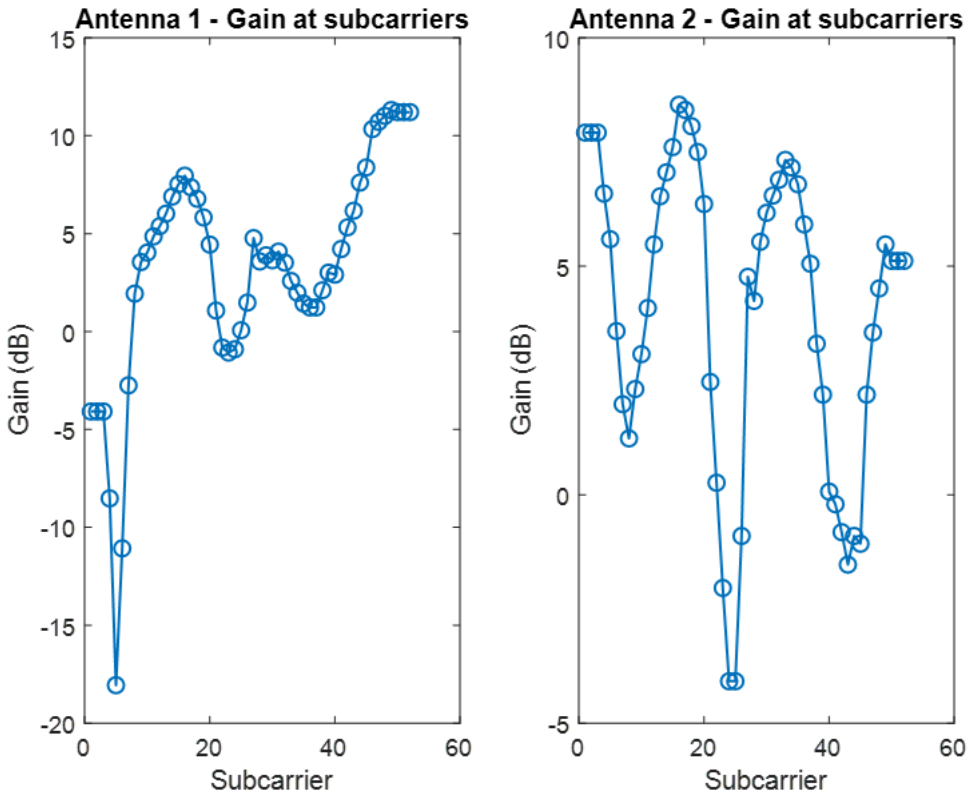

Figure 3. Relative gain at the reported subcarriers

Figure 4 shows the phase across the reported subcarriers for both receiver antennas.

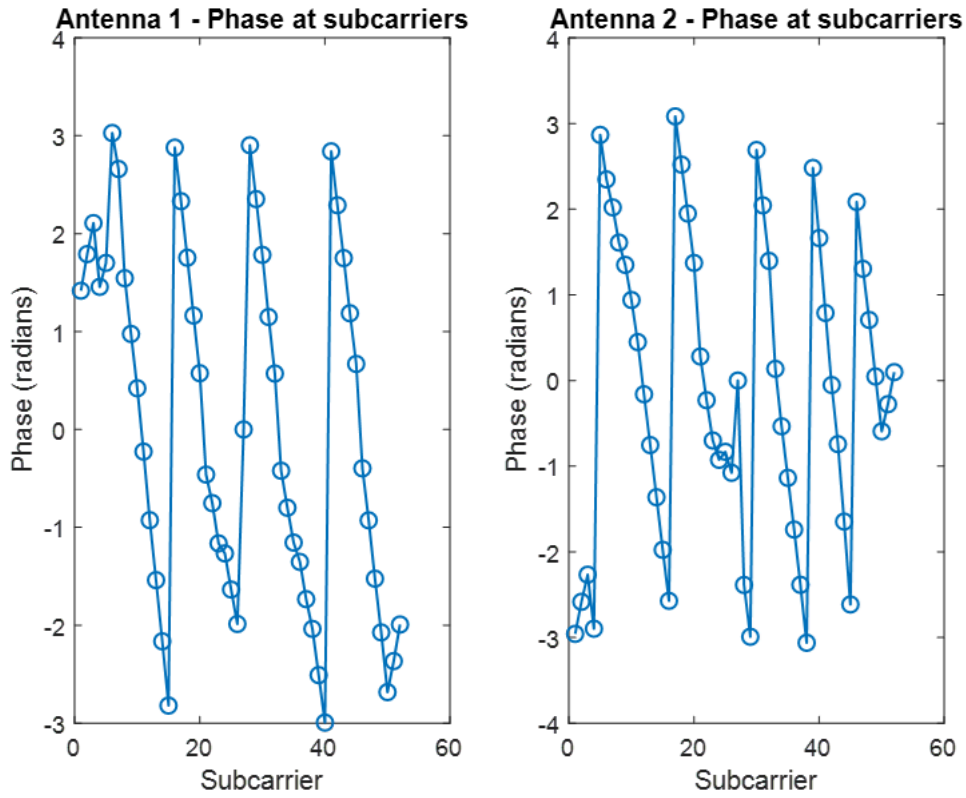

Figure 4. Phase across reported subcarriers for receive antennas

8.2 Impulse response and delay

Channel Impulse Response (CIR) = 0-padded Inverse Fast Fourier Transform (IFFT) of CSI data.

Figure 5 shows the channel impulse response for the two antennas.

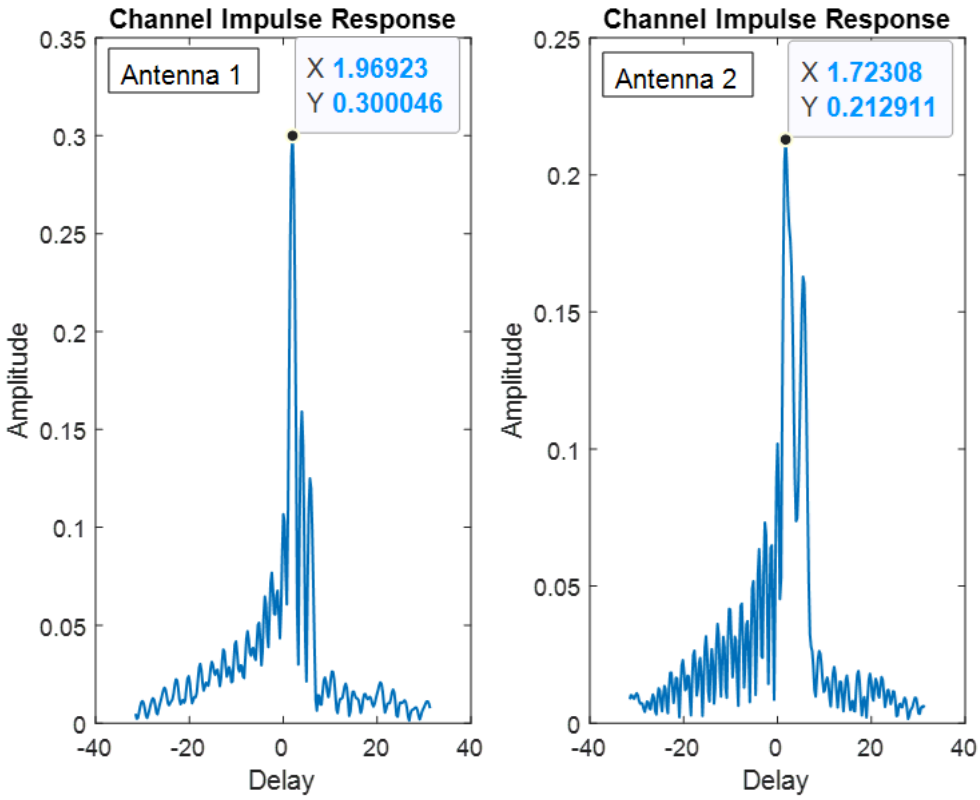

Figure 5. Channel impulse response

9 Appendix

This section documents the details of CSI functionality, which includes the following:

- API initialization sequences
- s8.n fixed-point data format specifications
- Host driver command
- Event handling procedures

This information serves as a reference for developers and system integrators working with CSI-based applications.

9.1 API sequence

Figure 6 shows the sequence of API commands and events between the host of NXP Wi-Fi device and the peer Wi-Fi device. For more information on the driver and firmware APIs, contact your NXP representative.

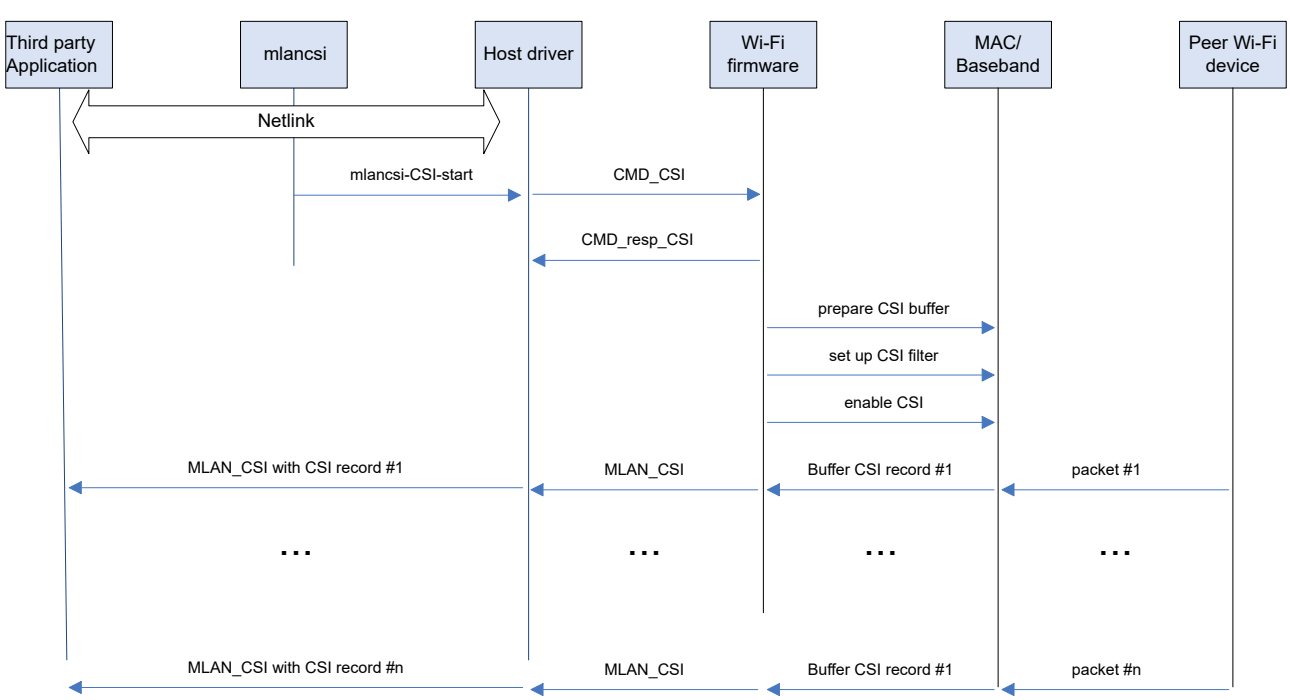

Figure 6. CSI commands and events

9.2 s8.n fixed-point format

A fixed-point format is a way to present fractional numbers using integers and fixed decimals. Based on `csi.conf`, the CSI format is in s8.3, s8.4, s8.5, or s8.6 fixed-point format.

- s8.3 fixed-point format: 8 bits total with 5 bits as integers and 3 bits as decimal
- s8.4 fixed-point format: 8 bits total with 4 bits as integers and 4 bits as decimal
- s8.5 fixed-point format: 8 bits total with 3 bits as integers and 5 bits as decimal
- s8.6 fixed-point format: 8 bits total with 2 bits as integers and 6 bits as decimal

Figure 7 shows a visual representation of s8.3 fixed-point format.

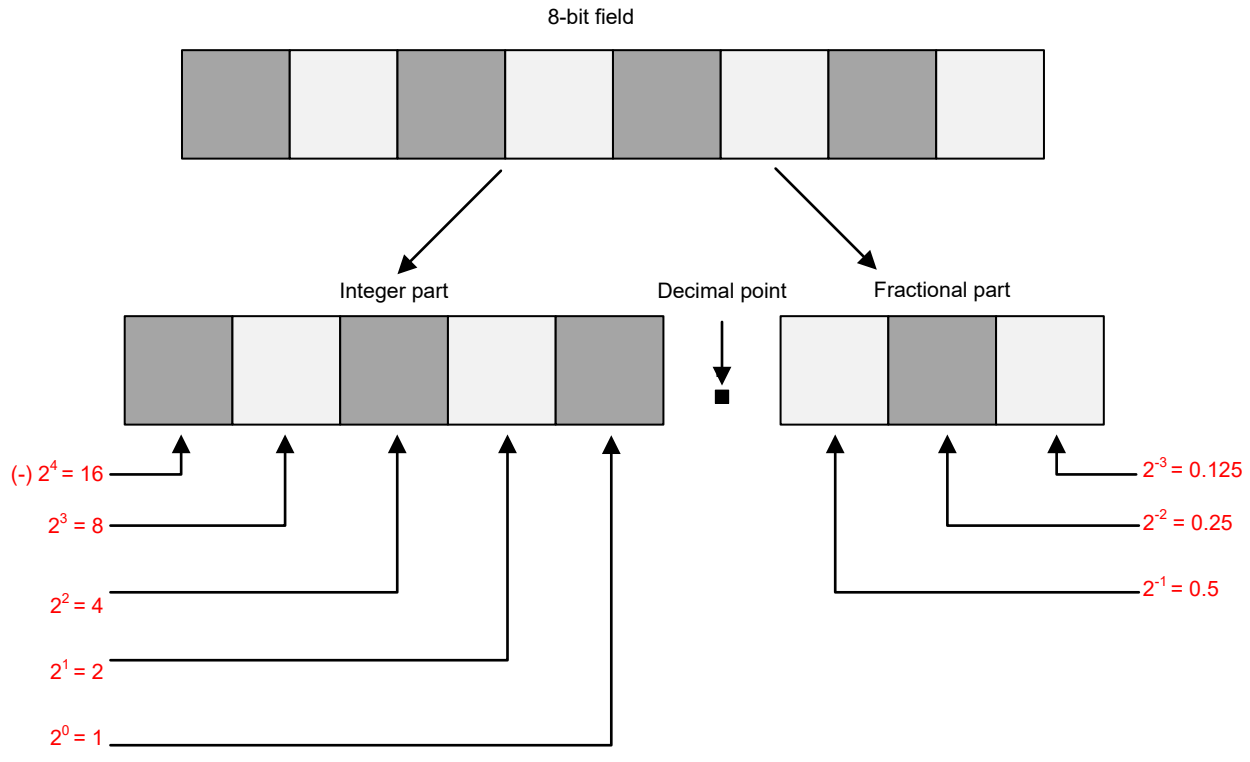

Figure 7. S8.3 fixed-point format

Example 1 – Positive number

0x16 = 0001 0110 (can be padded with leading 0s to make 8 bits)

The decimal point is placed 3 bits from the right.

00010 . 110

= 2.75

The left part of the decimal includes the positive exponents. The first bit indicates if the number is positive (0) or negative (1).

The right part of the decimal includes the negative exponents.

This example is for a positive number (0).

$2^1 + 2^{-1} + 2^{-2}$

= 2 + 0.5 + 0.25

Example 2 – Negative number

0xCA = 1100 1010

The decimal point is placed 3 bits from the right.

11001 . 010

The left part of the decimal includes the positive exponents. The first bit indicates if the number is positive (0) or negative (1).

The right part of the decimal includes the negative exponents.

Example 2 is for a negative number (1).

$-2^4 + 2^3 + 2^0 + 2^{-2}$

= -16 + 8 + 1 + 0.25 (starts as -16 because of 2's complement)

= -6.75

9.3 Host driver command

The host driver sends `CMD_CSI` host command to the Wi-Fi firmware. The Wi-Fi firmware sends `CMD_resp_CSI` response to the host driver.

`CMD_CSI` command informs the firmware to start or stop the collection of CSI data.

For MLAN CSI command request and response, contact your NXP representative.

9.4 Event handling

The driver sends a Netlink event, including the CSI event name and record to mlancsi utility for processing. The driver broadcasts a Netlink event, including the CSI event name and record.

For Netlink event format, contact your NXP representative.

10 References

[1] Published amendment to a standard – IEEE 802.11ax-2021 (link)
[2] Standard – IEEE 802.11-2016 (link)

11 Abbreviations

Table 17 lists abbreviations used in this document.

Table 17. Abbreviations

Abbreviation	Description
ACS	Auto Channel Select
AMI	Ambient Motion Index
BCC	Binary Convolutional Coding
BW	Bandwidth
CIR	Channel Impulse Response
CSI	Channel State Information
DCM	Dual Carrier Modulation
GF	Greenfield
GI	Guard Interval
HE	High Efficiency
HT	High Throughput
IFFT	Inverse Fast Fourier Transform
LDPC	Low-density Parity-check
LG	Legacy
LTF	Long Training Field
MU	Multiple Users
NF	Noise Floor
OFDM	Orthogonal Frequency Division Multiplexing
PKT	Packet
PSB	Primary Subband
RA	Receiver Address
RSSI	Received Signal Strength Indication
RSVD	Reserved
SINR	Signal to Interference Noise Ratio
SS	Spatial Stream
STA	Station
STBC	Space Time Block Code
SU	Single User
TA	Transmitter Address
TSF	Time Synchronization Function
uAP	Mobile Access Point
VHT	Very High Throughput

12 Note about the source code in the document

The example code shown in this document has the following copyright and BSD-3-Clause license:

Copyright 2025-2026 NXP Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials must be provided with the distribution.
3. Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

13 Revision history

Table 18 summarizes revisions to this document.

Table 18. Revision history

Document ID	Release date	Description
AN14011 v.9.0	13 May 2026	• Removed section "Auto transmit". • Removed all references to mlanutl. • Updated following sections for mlancsi: – Section 2 "Modules and flow " – Section 9.4 "Event handling" • Note removed from Section 5 "CSI dump details ".
AN14011 v.8.0	21 April 2026	• Tables updated: – Table 1 and Table 2 in Section 1.1 "Supported products" – Table 16 in Section 6.1 "Start calculating AMI" • Following section are updated for mlancsi: – Section 2 "Modules and flow " – Section 4.1 "Start command" – Section 4.2 "Stop command" – Get auto transmit configurations – Start auto transmit null packets – Stop auto transmit null packets – Configure auto transmit packets – Section 6.1 "Start calculating AMI" – Section 7.1 "Beacons" – Section 7.2 "Management packets" – Section 7.3 "Data packets" – Set up the uAP to send QoS null packets – Configure the STA to generate CSI from the received QoS null packets – Section 7.4 "Connectionless/monitor mode " – Section 7.5 "Multi-APs" – Section 7.6 "Multi-clients" – Section 7.7.1 "No motion" – Section 7.7.2 "Motion" • Content updated in the following sections: – Section 5 "CSI dump details " – Section 6 "Processing CSI – Ambient Motion Index" – Section 9.1 "API sequence" – Section 9.3 "Host driver command" – Section 9.4 "Event handling" – Section 10 "References" • Figure 1 in Section 2 "Modules and flow ": updated for mlancsi • Figure 6 in Section 9.1 "API sequence": : updated for mlancsi
AN14011 v.7.0	2 October 2025	• Section 1.1 "Supported products": updated. • Section 7.2 "Management packets": updated `csiformat` definition. • Section 7.3 "Data packets": updated `csiformat` definition and value. • Configure the STA to generate CSI from the received QoS null packets: updated `csiformat` definition and value. • Section 7.4 "Connectionless/monitor mode ": added. • Section 7.5 "Multi-APs": added. • Section 7.6 "Multi-clients": added.

Table 18. Revision history...*continued*

Document ID	Release date	Description
		• Section 7.7.2 "Motion": updated the first sentence. • Section 11 "Abbreviations": updated.
AN14011 v.6.0	21 August 2025	• Section 3.3 "CSI data format": updated the definition of Nc = Ntx. • Section 6 "Processing CSI – Ambient Motion Index": updated the introduction. • Section 7.7.1 "No motion": updated the command output example in **Step 6**. • Section 7.7.2 "Motion": updated the command output example in **Step 6**.
AN14011 v.5.0	29 July 2025	New version for the Beta version of the software release: • Section 2 "Modules and flow ": added mlancsi application. • Section 3.1 "CSI record format": updated Dword2 and Dword10. • Section 3.2 "Field descriptions": updated. • Section 3.3 "CSI data format": updated • Section 4 "CSI generation ": added `commonAGCflag` and `csiformat`parameters in *csi.conf* file. • Section 4.1 "Start command": updated the command output example. • Section 6 "Processing CSI – Ambient Motion Index": added. • Section 7.1 "Beacons": updated **Step 3**. • Section 7.2 "Management packets": updated **Step 3**. • Section 7.3 "Data packets": updated **Step 3**. • Configure the STA to generate CSI from the received QoS null packets: updated **Step 3**. • Section 7.7 "Ambient motion detection": added. • Section 9.2: updated the section title and introduction.
AN14011 v.4.0	7 May 2025	• Section 3.2 "Field descriptions": – Table 4: corrected TSF, moved FCF, updated the definitions of RX_RSSI and SINR. – Updated the titles of Table 7 and Table 8. – Added Table 9. • Section 3.3 "CSI data format": added Table 12.

Table 18. Revision history...*continued*

Document ID	Release date	Description
AN14011 v.3.0	20 January 2025	• Section 1 "About this document": updated. • Section 3.2 "Field descriptions": updated. • Section 3.3 "CSI data format": updated. • Section 4 "CSI generation ": updated. • Section 4.1 "Start command": updated the command output example. • Auto transmit [Linux]: added. • Section 7.1 "Beacons": updated *csi.conf* in **Step 3**. • Section 7.2 "Management packets": updated *csi.conf* in **Step 3**. • Section 7.3 "Data packets": updated *csi.conf* in **Step 3**. • QoS null packets: added. • Section 8 "Sample applications": added Python to the list of tools in the introduction. • Section 9.2 "s8.n fixed-point format": – Updated the example for **Example 1 – Positive number**. – Updated the example for **Example 2 – Negative number**. • Section 9.3 "Host driver command": updated. • Section 9.4 "Event handling": updated. • Section 11 "Abbreviations"": added.

Table 18. Revision history...*continued*

Document ID	Release date	Description
AN14011 v.2.0	27 June 2024	• Section 1.1 "Supported products": – Renamed Wi-Fi 5 as Wi-Fi 5 product category. – Renamed Wi-Fi 6 as Wi-Fi 6 product category. – Added a note. • Section 3.1 "CSI record format": updated the Header signature ID (byte 1:0] in the table. • Section 3.2 "Field descriptions": – Table 4 "Fields of CSI record": updated the fields RSVD[11:0], AP_TYPE, and CSI Data Length. – Table 5 "PKT_INFO[31:0] signals for Wi-Fi 5 product category": updated the signals devBW[1:0], nRx [2:0] or Nr, nTx[2:0] or Nc, and sigBW[1:0]. – Table 6 "PKT_INFO[31:0] signals for Wi-Fi 6 product category ": updated the signals devBW[1:0], nRx [2:0] or Nr, nTx[2:0] or Nc, and sigBW[1:0]. – Table 8 "nTones parameter values for Wi-Fi 6 product category except 88W9098, 88Q9098, 88Q9098S, and IW620 (see Table 6nTones are based on subcarrier indexes for compressed beamforming feedback. Refer to VHT, HT, and HE subcarrier and number of matrices/carrier grouping tables in *** and ***.) ": updated. – Added a note about nTones. • Section 3.3 "CSI data format": updated. • Section 8.1 "Amplitude and phase graphs": explained how to calculate the gain and phase. • Section 9.1 "API sequence": added the section heading (no changes in the content). • Section 9.2 "s8.n fixed-point format": added. • Section 10 "References": updated.
AN14011 v.1.0	8 December 2023	• Initial version

Legal information

Definitions

Draft — A draft status on a document indicates that the content is still under internal review and subject to formal approval, which may result in modifications or additions. NXP Semiconductors does not give any representations or warranties as to the accuracy or completeness of information included in a draft version of a document and shall have no liability for the consequences of use of such information.

Disclaimers

Limited warranty and liability — Information in this document is believed to be accurate and reliable. However, NXP Semiconductors does not give any representations or warranties, expressed or implied, as to the accuracy or completeness of such information and shall have no liability for the consequences of use of such information. NXP Semiconductors takes no responsibility for the content in this document if provided by an information source outside of NXP Semiconductors.

In no event shall NXP Semiconductors be liable for any indirect, incidental, punitive, special or consequential damages (including - without limitation - lost profits, lost savings, business interruption, costs related to the removal or replacement of any products or rework charges) whether or not such damages are based on tort (including negligence), warranty, breach of contract or any other legal theory.

Notwithstanding any damages that customer might incur for any reason whatsoever, NXP Semiconductors' aggregate and cumulative liability towards customer for the products described herein shall be limited in accordance with the Terms and conditions of commercial sale of NXP Semiconductors.

Right to make changes — NXP Semiconductors reserves the right to make changes to information published in this document, including without limitation specifications and product descriptions, at any time and without notice. This document supersedes and replaces all information supplied prior to the publication hereof.

Applications — Applications that are described herein for any of these products are for illustrative purposes only. NXP Semiconductors makes no representation or warranty that such applications will be suitable for the specified use without further testing or modification.

Customers are responsible for the design and operation of their applications and products using NXP Semiconductors products, and NXP Semiconductors accepts no liability for any assistance with applications or customer product design. It is customer's sole responsibility to determine whether the NXP Semiconductors product is suitable and fit for the customer's applications and products planned, as well as for the planned application and use of customer's third party customer(s). Customers should provide appropriate design and operating safeguards to minimize the risks associated with their applications and products.

NXP Semiconductors does not accept any liability related to any default, damage, costs or problem which is based on any weakness or default in the customer's applications or products, or the application or use by customer's third party customer(s). Customer is responsible for doing all necessary testing for the customer's applications and products using NXP Semiconductors products in order to avoid a default of the applications and the products or of the application or use by customer's third party customer(s). NXP does not accept any liability in this respect.

Terms and conditions of commercial sale — NXP Semiconductors products are sold subject to the general terms and conditions of commercial sale, as published at https://www.nxp.com/profile/terms, unless otherwise agreed in a valid written individual agreement. In case an individual agreement is concluded only the terms and conditions of the respective agreement shall apply. NXP Semiconductors hereby expressly objects to applying the customer's general terms and conditions with regard to the purchase of NXP Semiconductors products by customer.

Suitability for use in automotive applications — This NXP product has been qualified for use in automotive applications. If this product is used by customer in the development of, or for incorporation into, products or services (a) used in safety critical applications or (b) in which failure could lead to death, personal injury, or severe physical or environmental damage (such products and services hereinafter referred to as "Critical Applications"), then customer makes the ultimate design decisions regarding its products and is solely responsible for compliance with all legal, regulatory, safety, and security related requirements concerning its products, regardless of any information or support that may be provided by NXP. As such, customer assumes all risk related to use of any products in Critical Applications and NXP and its suppliers shall not be liable for any such use by customer. Accordingly, customer will indemnify and hold NXP harmless from any claims, liabilities, damages and associated costs and expenses (including attorneys' fees) that NXP may incur related to customer's incorporation of any product in a Critical Application.

Export control — This document as well as the item(s) described herein may be subject to export control regulations. Export might require a prior authorization from competent authorities.

HTML publications — An HTML version, if available, of this document is provided as a courtesy. Definitive information is contained in the applicable document in PDF format. If there is a discrepancy between the HTML document and the PDF document, the PDF document has priority.

Translations — A non-English (translated) version of a document, including the legal information in that document, is for reference only. The English version shall prevail in case of any discrepancy between the translated and English versions.

Security — Customer understands that all NXP products may be subject to unidentified vulnerabilities or may support established security standards or specifications with known limitations. Customer is responsible for the design and operation of its applications and products throughout their lifecycles to reduce the effect of these vulnerabilities on customer's applications and products. Customer's responsibility also extends to other open and/or proprietary technologies supported by NXP products for use in customer's applications. NXP accepts no liability for any vulnerability. Customer should regularly check security updates from NXP and follow up appropriately.

Customer shall select products with security features that best meet rules, regulations, and standards of the intended application and make the ultimate design decisions regarding its products and is solely responsible for compliance with all legal, regulatory, and security related requirements concerning its products, regardless of any information or support that may be provided by NXP.

NXP has a Product Security Incident Response Team (PSIRT) (reachable at PSIRT@nxp.com) that manages the investigation, reporting, and solution release to security vulnerabilities of NXP products.

NXP B.V. — NXP B.V. is not an operating company and it does not distribute or sell products.

Trademarks

Notice: All referenced brands, product names, service names, and trademarks are the property of their respective owners.

NXP — wordmark and logo are trademarks of NXP B.V.

Amazon Web Services, AWS, the Powered by AWS logo, and FreeRTOS — are trademarks of Amazon.com, Inc. or its affiliates.

Bluetooth — the Bluetooth wordmark and logos are registered trademarks owned by Bluetooth SIG, Inc. and any use of such marks by NXP Semiconductors is under license.

MATLAB — is a registered trademark of The MathWorks, Inc.

Tables

Figures

Contents