Application Note

Detailed Explanation of AWR2188 CSI2 Usage and Data Output

Chris Meng *Central mmWave FAE*

ABSTRACT

This document was translated from a simplified Chinese source. (ZHCAGA0)

The AWR2188, TI's next-generation high-performance mmWave RF front end, is widely applicable in automotive ADAS applications, such as forward radar. CSI2 is an important interface as an output port for AWR2188 raw radar ADC data. The accuracy of any subsequent radar signal processing results first depends on the correctness of data output. Starting with the hardware interface, this article provides an in-depth and detailed description of AWR2188 CSI2 hardware connections and software configuration, and provides configuration examples, aiming to provide AWR2188 users with comprehensive CSI2 reference materials and help them achieve rapid development.

Table of Contents

List of Figures

List of Tables

Trademarks

All trademarks are the property of their respective owners.

1 Introduction

The AWR2188 is TI's next-generation high-performance mmWave RF front end. Built on TI's low-power 45nm RFCMOS process, it integrates an 8-transmitter, 8-receiver system with built-in PLL and ADC converters in a monolithic form factor. By allowing users to modify the programming model, it supports multiple sensor implementations, such as short-, medium-, and long-range, and supports dynamic reconfiguration to realize a multi-mode sensor. Raw radar ADC data from the AWR2188 is output via CSI2 to a processor for edge processing, or input via a serializer to a domain controller for remote data analysis.

2 AWR2188 CSI2 Hardware Interface

The CSI2 interface of AWR2188 ES2.0 consists of five pairs of differential signals, supporting up to four data lanes and one clock lane.

Table 2-1. AWR2243 CSI2 Pin[1]

Signal Name	Pin Type	Description	Pin Number
CSI2A_TXM0	Output	Lane 0 differential output negative	V1
CSI2A_TXM1	Output	Lane 1 differential output negative	T1
CSI2A_CLKM	Output	Differential clock output negative	Y1
CSI2A_TXM2	Output	Lane 2 differential output negative	AE1
CSI2A_TXM3	Output	Lane 3 differential output negative	AC2
CSI2A_TXP0	Output	Lane 0 differential output positive	U1
CSI2A_TXP1	Output	Lane 1 differential output positive	R1
CSI2A_CLKP	Output	Differential clock output positive	W1
CSI2A_TXP2	Output	Lane 2 differential output positive	AE2
CSI2A_TXP3	Output	Lane 3 differential output positive	AC1

The function of these five pairs of differential signals is software configurable. Two pairs of differential signal hardware pins, V1/U1 and Y1/W1, are fixed to Lane 0 and the clock lane, respectively. Other pins for differential pairs can be set to Lane 1, Lane 2, or Lane 3, but the same lane cannot be configured repeatedly.

Table 2-2. Correspondence Between CSI2 Hardware Pin and Lane Position

Lane Position	AWR2188 Associated Pin	CSI2 Functionality
M_ML_MSS_CSI2_LANE_POSITION_1	V1/U1	Fixed to Lane 0
M_ML_MSS_CSI2_LANE_POSITION_2	T1/R1	Can be configured to Lane 1, Lane 2, or Lane 3
M_ML_MSS_CSI2_LANE_POSITION_3	Y1/W1	Fixed to the clock lane
M_ML_MSS_CSI2_LANE_POSITION_4	AE1/AE2	Can be configured to Lane 1, Lane 2, or Lane 3
M_ML_MSS_CSI2_LANE_POSITION_5	AC2/AC1	Can be configured to Lane 1, Lane 2, or Lane 3

In addition to providing the AWR2188 chip, TI also provides the AWR2188 satellite radar reference design for user reference [2] . The hardware connection diagram of AWR2188 CSI2 is shown in Figure 2-1.

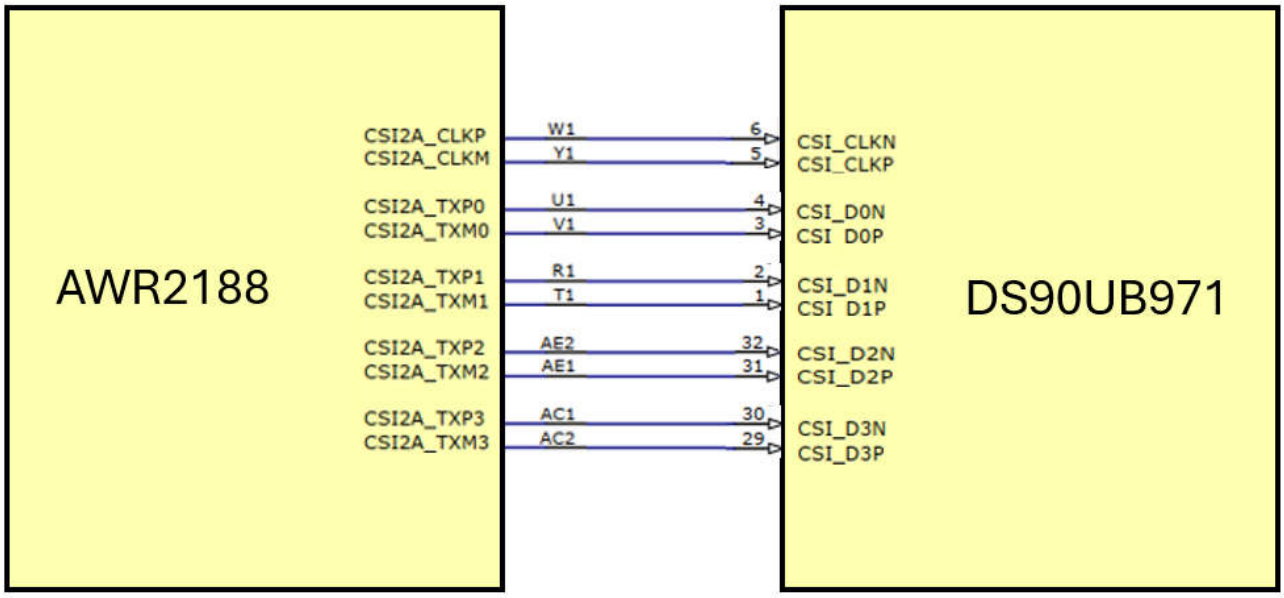

Figure 2-1. AWR2188 Satellite Radar Reference Design - CSI2 Connection Diagram

3 AWR2188 CSI2 Output Configuration

The AWR2188 allows users to more flexibly configure CSI2 output content, compared to TI's previous generation RF front-end chip, AWR2243. Users need to call the MSS RX Data Path Configuration Set API (ml_mssDataPathConfig). This API is used to set the CSI2 pin functionality of the AWR2188 and the format of radar raw ADC data output via CSI2 frames.

3.1 Hardware-Related Configuration of CSI2

The c_HsIfALaneConfig and w_LaneCfg parameters in T_ML_MSS_DATAPATH_CFG are used to configure the CSI2 pin functionality of the AWR2188.

The c_HsIfALaneConfig parameter enables single or multiple data lanes for CSI2. To maximize the CSI2 data throughput, users typically enable all data lanes. However, if the back-end chip supports a limited number of lanes, the AWR2188 can be used together by reducing the number of enabled lanes.

The w_LaneCfg parameter matches CSI2 lanes to specific functionalities and sets the polarity of each lane. The user's hardware pin connection design must be aligned with software configuration, while the correct lane polarity is set depending on the requirements of the back-end chip to get correct CSI2 data.

The CSI2 connection of the AWR2188 satellite radar reference design mentioned in the previous section is based on the CSI2 pin functionality configuration shown in Table 3-1 and Table 3-2.

Table 3-1. AWR2188 Satellite Radar Reference Design - c_HsIfALaneConfig Configuration Example

Bit	Lane	Configuration	Meaning
BIT[0]	Lane 0	1	Enable Lane 0
BIT[1]	Lane 1	1	Enable Lane 1
BIT[2]	Lane 2	1	Enable Lane 2
BIT[3]	Lane 3	1	Enable Lane 3
BIT[7:4]	Reserved	/	/

Table 3-2. AWR2188 Satellite Radar Reference Design - w_LaneCfg Configuration Example

Parameter	Bit	Value	Meaning
Lane 0 position	BIT[2:0]	1	Position 1
Lane 0 polarity	BIT[3]	0	Lane 0 polarity order: Positive -> negative
Lane 1 position	BIT[6:4]	2	Position 2
Lane 1 polarity	BIT[7]	0	Lane 1 polarity order: Positive -> negative
Lane 2 position	BIT[10:8]	4	Position 4
Lane 2 polarity	BIT[11]	0	Lane 2 polarity order: Positive -> negative
Lane 3 position	BIT[14:12]	5	Position 5
Lane 3 polarity	BIT[15]	0	Lane 3 polarity order: Positive -> negative
Clock lane position	BIT[18:16]	3	Position 3
Clock lane polarity	BIT[19]	0	Clock lane polarity order: Positive -> negative
Reserved	BIT[31:19]	/	Reserved

3.2 CSI2 Output Data Content Configuration

The c_RxInterleaveCtrl, c_HsiMode, and z_HsiAChirpDataCfgLst parameters in T_ML_MSS_DATAPATH_CFG are related to CSI2 output data content/format.

The c_RxInterleaveCtrl parameter sets the interleaved mode control for receive channels. The AWR2188 only supports the non-interleaved mode, so this parameter is set to 1 accordingly. Below is an example of data output in non-interleaved mode, where Rx0, Rx1, Rx2, Rx3, Rx4, Rx5, Rx6, and Rx7 represent different receive channels. The number following the receive channel indicates the number of sampling points for that channel. The number of sampling points per chirp is N+1, for example:

Rx00 Rx01 Rx02 Rx03.....Rx0N Rx10 Rx11 Rx12 Rx13.....Rx3N

The c_HsiMode parameter specifies the contents of the CSI2 data output, which is configured to 0xAA (Sensor Data Mode) for normal wave transmission.

Table 3-3. About c_HsiMode

Value	Description	Macro
0xA0U	Test sequence data mode	M_ML_MSS_DATAPATH_MODE_TEST_PATTERN
0xA5U	Continuous wave (CW) data mode	M_ML_MSS_DATAPATH_MODE_CW
0xAAU	Sensor data mode	M_ML_MSS_DATAPATH_MODE_SENSOR

The z_HsiAChirpDataCfgLst parameter is the configuration table for the AWR2188 in sensor mode. This configuration table is used to specify the form of each chirp data output over the CSI2 interface. Each long packet over CSI2 has a corresponding linked list entry description, supporting a maximum of 16 linked list entries. The contents of each entry can be specified by the user, mainly including:

- Whether to include Hsync start or Hsync end in the CSI2 packet (i.e., line start/end short packets over CSI2);
- Whether to include a long packet header;
- The virtual channel (VC channel) number used;
- The data format over CSI2, such as RAW8 (16-bit ADC), RAW12 (12-bit ADC), or RAW14 (14-bit ADC);
- The CBUFF data format;
- The contents in long packets, such as receive channels (RX0–RX7), CP (chirp profile), and CQ (chirp quality).

It is generally recommended that the user configure at least eight z_HsiAChirpDataCfgLst instances to specify the data output format over eight receive channels (RXs), with extra instances to be added when CP or CQ data is included. Table 3-4 and Table 3-5 provide partial linked list reference configuration examples for 16-bit ADC (with CP) and 12-bit ADC.

Table 3-4. Partial Linked List Reference Configuration Example for 16-Bit ADC (With CP) Data

Chirp Linked List	Parameter	Bit	Value	Meaning
.z_HsiAChirpDataCfgLst[0]	c_LLConfig	0	1	Enable linked list
		2~1	1	Enable line sync start
		5~3	0	Use virtual channel 0
		6	1	Enable long packet header
		7	1	The data load for outputting a long packet is a multiple of 96 bits
	c_HsDataFormat	5~0	0x2A	RAW8
		7~6	0	CBUFF data is in 16-bit format (16-bit ADC data)
	c_ContentType		0	The long packet output contains ADC data for the signal received by R0
	h_NumDummySamples		0	The padding data length is 0

Table 3-4. Partial Linked List Reference Configuration Example for 16-Bit ADC (With CP) Data (continued)

Chirp Linked List	Parameter	Bit	Value	Meaning
.z_HsiAChirpDataCfgLst[7]	c_LLConfig	0	1	Enable linked list
		2~1	0	Disable line sync short packet
		5~3	0	Use virtual channel 0
		6	0	Disable long packet header
		7	1	The data load for outputting a long packet is a multiple of 96 bits
	c_HsDataFormat	5~0	0x2A	RAW8
		7~6	0	CBUFF data is in 16-bit format (16-bit ADC data)
	c_ContentType	Unit8	7	The long packet output contains ADC data for the signal received by R7
	h_NumDummySamples	Unit16	0	The padding data length is 0
.z_HsiAChirpDataCfgLst[8]	c_LLConfig	0	1	Enable linked list
		2~1	3	Enable line sync end short packet
		5~3	0	Use virtual channel 0
		6	0	Disable long packet header
		7	1	The data load for outputting a long packet is a multiple of 96 bits
	c_HsDataFormat	5~0	0x2A	RAW8
		7~6	0	CBUFF data is in 16-bit format (16-bit ADC data)
	c_ContentType	Unit8	8	The long packet output contains CP
	h_NumDummySamples	Unit16	0	The padding data length is 0
c_AdcDataLen=M_ML_RSS_ADC_DATA_LEN_16_BITS				

Table 3-5. Partial Linked List Reference Configuration Example for 12-Bit ADC Data

Chirp Linked List	Parameter	Bit	Value	Meaning
.z_HsiAChirpDataCfgLst[0]	c_LLConfig	0	1	Enable linked list
		2~1	1	Enable line sync start
		5~3	0	Use virtual channel 0
		6	1	Enable long packet header
		7	1	The data load for outputting a long packet is a multiple of 96 bits
	c_HsDataFormat	5~0	0x2C	RAW12
		7~6	2	CBUFF data is in 12-bit format (12-bit ADC data)
	c_ContentType		0	The long packet output contains ADC data for the signal received by R0
	h_NumDummySamples		0	The padding data length is 0

Table 3-5. Partial Linked List Reference Configuration Example for 12-Bit ADC Data (continued)

Chirp Linked List	Parameter	Bit	Value	Meaning
.z_HsiAChirpDataCfgLst[7]	c_LLConfig	0	1	Enable linked list
		2~1	3	Enable line sync end short packet
		5~3	0	Use virtual channel 0
		6	0	Disable long packet header
		7	1	The data load for outputting a long packet is a multiple of 96 bits
	c_HsDataFormat	5~0	0x2C	RAW12
		7~6	2	CBUFF data is in 12-bit format (12-bit ADC data)
	c_ContentType	Unit8	7	The long packet output contains ADC data for the signal received by R7
	h_NumDummySamples	Unit16	0	The padding data length is 0
c_AdcDataLen=M_ML_RSS_ADC_DATA_LEN_12_BITS				

The CSI2 output configuration of the AWR2188 is flexible, allowing better adaptation to different back-end chips, such as serializers, processors, and so forth. If the back-end chip has restrictions on the data size per line of CSI2 frames, the user can divide the data of one chirp into two or more lines for output, thereby reducing the amount of data per line and ensuring that the back-end chip receives it properly. Each long packet incurs about 3us of CSI protocol overhead. This overhead must be considered by extending the chirp idle time appropriately when the user divides chirp data into multiple lines for output. Table 3-6 shows a configuration scheme where the data of one chirp over eight receive channels is output in two lines. In this scheme, the data over Rx0 to Rx3 forms one line, and the data over Rx4 to Rx7 forms the other line.

Table 3-6. Partial Reference Configuration Example for Data of One Chirp (16-Bit ADC) Output in Two Lines

Chirp Linked List	Parameter	Bit	Value	Meaning
.z_HsiAChirpDataCfgLst[0]	c_LLConfig	0	1	Enable linked list
		2~1	1	Enable line sync start
		5~3	0	Use virtual channel 0
		6	1	Enable long packet header
		7	1	The data load for outputting a long packet is a multiple of 96 bits
	c_HsDataFormat	5~0	0x2A	RAW8
		7~6	0	CBUFF data is in 16-bit format (16-bit ADC data)
	c_ContentType		0	The long packet output contains ADC data for the signal received by R0
	h_NumDummySamples		0	The padding data length is 0
.z_HsiAChirpDataCfgLst[3]	c_LLConfig	0	1	Enable linked list
		2~1	3	Enable line sync end short packet
		5~3	0	Use virtual channel 0
		6	0	Disable long packet header
		7	1	The data load for outputting a long packet is a multiple of 96 bits
	c_HsDataFormat	5~0	0x2A	RAW8
		7~6	0	CBUFF data is in 16-bit format (16-bit ADC data)
	c_ContentType	Unit8	3	The long packet output contains ADC data for the signal received by R3
	h_NumDummySamples	Unit16	0	The padding data length is 0

Table 3-6. Partial Reference Configuration Example for Data of One Chirp (16-Bit ADC) Output in Two Lines (continued)

Chirp Linked List	Parameter	Bit	Value	Meaning
.z_HsiAChirpDataCfgLst[4]	c_LLConfig	0	1	Enable linked list
		2~1	1	Enable line sync start
		5~3	0	Use virtual channel 0
		6	1	Enable long packet header
		7	1	The data load for outputting a long packet is a multiple of 96 bits
	c_HsDataFormat	5~0	0x2A	RAW8
		7~6	0	CBUFF data is in 16-bit format (16-bit ADC data)
	c_ContentType		4	The long packet output contains ADC data for the signal received by R4
	h_NumDummySamples		0	The padding data length is 0
.z_HsiAChirpDataCfgLst[7]	c_LLConfig	0	1	Enable linked list
		2~1	3	Enable line sync end short packet
		5~3	0	Use virtual channel 0
		6	0	Disable long packet header
		7	1	The data load for outputting a long packet is a multiple of 96 bits
	c_HsDataFormat	5~0	0x2A	RAW8
		7~6	0	CBUFF data is in 16-bit format (16-bit ADC data)
	c_ContentType	Unit8	7	The long packet output contains ADC data for the signal received by R7
	h_NumDummySamples	Unit16	0	The padding data length is 0
c_AdcDataLen=M_ML_RSS_ADC_DATA_LEN_16_BITS				

4 CSI2 Output Data Format Examples

4.1 CP Data Format

On the CSI2 interface of the AWR2188, in addition to the ADC data output, the chirp profile (CP), which contains the chirp sequence number, subframe sequence number, number of frames, and so forth, can be simultaneously output. Table 4-1 shows the CP data format when the ADC output is set to 16-bit.

Table 4-1. CP Data Format (16-Bit ADC)

	16 bits																16 bits								
Bit ->	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	15~8	7	6	5	4	3	2	1	0
CP Register 0	Padding				Chirp sequence number [16:27]												Padding	Profile [2:7]						Padding	
CP Register 1	Padding								Subframe sequence number								Padding	CSI Identifier							
CP Register 2	Padding								Burst sequence number byte 1								Padding	Burst sequence number byte 0							
CP Register 3	Padding								Frame sequence number byte 1								Padding	Frame sequence number byte 0							
CP Register 4	Padding								Frame sequence number byte 3								Padding	Frame sequence number byte 2							
CP Register 5	Padding								Reserved								Padding	Receive channel enable mask							
CP Register 6	Padding								Synthesizer linearity error byte 1								Padding	Synthesizer linearity error byte 0							
CP Register 7	Padding								Synthesizer linearity error byte 3								Padding	Synthesizer linearity error byte 2							
CP Register 8	Padding								Reserved								Padding	Reserved							
CP Register 9	Padding								Reserved								Padding	Reserved							

When the user-set length of ADC data for the AWR2188 is 12 bits, the CP data is slightly different from the CP data for 16-bit ADC data, as shown in Table 4-2.

Table 4-2. CP Data Format (12-Bit ADC)

	12 bits												12 bits											
Bit ->	11	10	9	8	7	6	5	4	3	2	1	0	11	10	9	8	7	6	5	4	3	2	1	0
CP Register 0	Chirp sequence number [16:27]												Padding				Profile [2:7]						Padding	
CP Register 1	Padding				Subframe sequence number								Padding				CSI Identifier							
CP Register 2	Padding				Burst sequence number byte 1								Padding				Burst sequence number byte 0							
CP Register 3	Padding				Frame sequence number byte 1								Padding				Frame sequence number byte 0							

Table 4-2. CP Data Format (12-Bit ADC) (continued)

	12 bits		12 bits	
CP Register 4	Padding	Frame sequence number byte 3	Padding	Frame sequence number byte 2
CP Register 5	Padding	Reserved	Padding	Receive channel enable mask
CP Register 6	Padding	Synthesizer linearity error byte 1	Padding	Synthesizer linearity error byte 0
CP Register 7	Padding	Synthesizer linearity error byte 3	Padding	Synthesizer linearity error byte 2
CP Register 8	Padding	Reserved	Padding	Reserved
CP Register 9	Padding	Reserved	Padding	Reserved

4.2 CQ Data Format

In addition to the CP data, the AWR2188, along with the ADC, can output the digital front end (DFE) statistics of chirp quality and ADC data saturation statistics. Table 4-3 and Table 4-4 show the CQ data format when the ADC is set to 16-bit and 12-bit, respectively.

Table 4-3. CQ Data Format (16-Bit ADC)

Address -->	16 bits	16 bits	16 bits	16 bits	16 bits	16 bits	16 bits	16 bits	
Digital front end ADC saturation count cumulative data	Receive channel 1	Receive channel 2	Receive channel 3	Receive channel 4	Receive channel 5	Receive channel 6	Receive channel 7	Receive channel 8	7
Sum of sampled data	Receive channel 1		Receive channel 2		Receive channel 3		Receive channel 4		
	Receive channel 5		Receive channel 6		Receive channel 7		Receive channel 8		
Sum of the magnitude of sampled data	Receive channel 1		Receive channel 2		Receive channel 3		Receive channel 4		
	Receive channel 5		Receive channel 6		Receive channel 7		Receive channel 8		
Sum of the difference in the amplitude of sampled data	Receive channel 1		Receive channel 2		Receive channel 3		Receive channel 4		
	Receive channel 5		Receive channel 6		Receive channel 7		Receive channel 8		
	128 bits								

Table 4-4. CQ Data Format (12-Bit ADC)

Address -->	12 bits	12 bits	12 bits	12 bits	12 bits	12 bits	12 bits	12 bits	
Digital front end ADC saturation count cumulative data	Receive channel 1	Receive channel 2	Receive channel 3	Receive channel 4	Receive channel 5	Receive channel 6	Receive channel 7	Receive channel 8	7
Sum of sampled data	Receive channel 1		Receive channel 2		Receive channel 3		Receive channel 4		
	Receive channel 5		Receive channel 6		Receive channel 7		Receive channel 8		
Sum of the magnitude of sampled data	Receive channel 1		Receive channel 2		Receive channel 3		Receive channel 4		
	Receive channel 5		Receive channel 6		Receive channel 7		Receive channel 8		
Sum of the difference in the amplitude of sampled data	Receive channel 1		Receive channel 2		Receive channel 3		Receive channel 4		
	Receive channel 5		Receive channel 6		Receive channel 7		Receive channel 8		
	96 bits								

4.3 Example of CSI2 Output Data Format in Sensor Mode

The AWR2188 supports 12/14/16-bit ADC data, with the ADC data of different bit widths varying in output over CSI2. Table 13 shows an example of a 12-bit ADC outputting one frame of ADC data over CSI2, where the frame contains 1024 sampling points, 1024 chirps, and data from 8 receive channels, along with CP information. Users should note that the raw ADC data on CSI2 and the data stored in the back-end chip may not be completely consistent. Some processors store the 12-bit data they receive in 12-bit format. Other processors, however, store the 12-bit data in 2-byte (or 16-bit) format.

Table 4-5. Example of Output Data Format in Sensor Mode (12-Bit ADC)

Size (bits)		CSI2 Data Content
12288	12	Chirp 0, receive channel 0, sampling point 1
	12	Chirp 0, receive channel 0, sampling point 2
	12	Chirp 0, receive channel 0, sampling point 3
	12	Chirp 0, receive channel 0, sampling point 4
	…	…
	12	Chirp 0, receive channel 0, sampling point 1024
12288		Chirp 0, receive channel 1, sampling points 1~1024
12288		Chirp 0, receive channel 2, sampling points 1~1024
12288		Chirp 0, receive channel 3, sampling points 1~1024
12288		Chirp 0, receive channel 4, sampling points 1~1024
12288		Chirp 0, receive channel 5, sampling points 1~1024
12288		Chirp 0, receive channel 6, sampling points 1~1024
12288		Chirp 0, receive channel 7, sampling points 1~1024
240		Chirp 0, CP data
…		
12288	12	Chirp 1023, receive channel 0, sampling point 1
	12	Chirp 1023, receive channel 0, sampling point 2
	12	Chirp 1023, receive channel 0, sampling point 3
	12	Chirp 1023, receive channel 0, sampling point 4
	…	…
	12	Chirp 1023, receive channel 0, sampling point 1024
12288		Chirp 1023, receive channel 1, sampling points 1~1024

Table 4-5. Example of Output Data Format in Sensor Mode (12-Bit ADC) (continued)

Size (bits)	CSI2 Data Content
12288	Chirp 1023, receive channel 2, sampling points 1~1024
12288	Chirp 1023, receive channel 3, sampling points 1~1024
12288	Chirp 1023, receive channel 4, sampling points 1~1024
12288	Chirp 1023, receive channel 5, sampling points 1~1024
12288	Chirp 1023, receive channel 6, sampling points 1~1024
12288	Chirp 1023, receive channel 7, sampling points 1~1024
240	Chirp 1023, CP data

4.4 Example of CSI2 Output Data Format in Continuous Wave (CW) Mode

The CSI2 ADC data output described earlier is based on normal wave transmission mode. When the user enables continuous wave (CW) transmission, CW ADC data can still be output, but only single-frame CSI2 data is generated. Each line of data will have a line flag with a length of 8 * ADC bit width. Line flag length for the corresponding bit width:

- 16-bit ADC: A line flag is 128 bits long
- 12-bit ADC: A line flag is 96 bits long

The output data format in CW mode depends on the configured parameters. The following example shows the parameter configuration scheme when ADC data is output in CW mode. The corresponding CSI2 output data can be found in Table 4-6.

```
c_AdcMode=0,//ADC 全速模式
.c_SamplingRateSel = 30, //(53.33 msps 采样率)
.c_AdcDataLen = M_ML_RSS_ADC_DATA_LEN_12_BITS,
.c_HsiMode=M_ML_MSS_DATAPATH_MODE_CW,
.z_CwTestModeCfg.h_NumTestSamplesPerLine = 2048,
.z_CwTestModeCfg.w_NumTestLines = 1024,
.z_CwTestModeCfg.c_LineVCConfig = M_ML_MSS_DATAPATH_VC_0,
.z_CwTestModeCfg.c_HsDataFormat = ((M_ML_MSS_LNG_PKT_DATA_FMT_12_BIT) |\
 (M_ML_MSS_CBUFF_DATA_FMT_12_BIT <<M_ML_MSS_DATAPATH_CBUFF_DATA_FMT_POS) ),
.z_CwTestModeCfg.c_HsiARxEnable = 0xFFU,
.c_HsIfALaneConfig = 0xFU,
.c_DataRate = M_ML_MSS_DATAPATH_RATE_1500_MBPS,
```

Table 4-6. Example of Output Data Format in CW Mode

Size (bits)	CSI2 Data Content
8*12	Line flag
24576	Chirp 0, receive channel 0, sampling points 1~2048
24576	Chirp 0, receive channel 1, sampling points 1~2048
24576	Chirp 0, receive channel 2, sampling points 1~2048
24576	Chirp 0, receive channel 3, sampling points 1~2048
24576	Chirp 0, receive channel 4, sampling points 1~2048
24576	Chirp 0, receive channel 5, sampling points 1~2048
24576	Chirp 0, receive channel 6, sampling points 1~2048
24576	Chirp 0, receive channel 7, sampling points 1~2048
…	
96	Line flag
24576	Chirp 1023, receive channel 0, sampling points 1~2048
24576	Chirp 1023, receive channel 1, sampling points 1~2048
24576	Chirp 1023, receive channel 2, sampling points 1~2048
24576	Chirp 1023, receive channel 3, sampling points 1~2048
24576	Chirp 1023, receive channel 4, sampling points 1~2048
24576	Chirp 1023, receive channel 5, sampling points 1~2048
24576	Chirp 1023, receive channel 6, sampling points 1~2048

Table 4-6. Example of Output Data Format in CW Mode (continued)

Size (bits)	CSI2 Data Content
24576	Chirp 1023, receive channel 7, sampling points 1~2048

5 Conclusion

This article systematically introduces the associated pins, hardware wiring, and software configuration in different modes for AWR2188 CSI2. Through the above sections, AWR2188 users are provided with comprehensive CSI2 reference documentation, which helps them gain deep insight and quickly achieve the required data output to accelerate product development.

6 References

1. AWR2188 Datasheet
2. AWR2188 Satellite Radar Reference Design
3. AWR2188 Interface control API documentation

IMPORTANT NOTICE AND DISCLAIMER

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

These resources are intended for skilled developers designing with TI products. You are solely responsible for (1) selecting the appropriate TI products for your application, (2) designing, validating and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, regulatory or other requirements.

These resources are subject to change without notice. TI grants you permission to use these resources only for development of an application that uses the TI products described in the resource. Other reproduction and display of these resources is prohibited. No license is granted to any other TI intellectual property right or to any third party intellectual property right. TI disclaims responsibility for, and you fully indemnify TI and its representatives against any claims, damages, costs, losses, and liabilities arising out of your use of these resources.

TI’s products are provided subject to TI’s Terms of Sale, TI’s General Quality Guidelines, or other applicable terms available either on ti.com or provided in conjunction with such TI products. TI’s provision of these resources does not expand or otherwise alter TI’s applicable warranties or warranty disclaimers for TI products. Unless TI explicitly designates a product as custom or customer-specified, TI products are standard, catalog, general purpose devices.

TI objects to and rejects any additional or different terms you may propose.

Last updated 10/2025