

Steps for Raw ADC Data Streaming in IWRL6432:

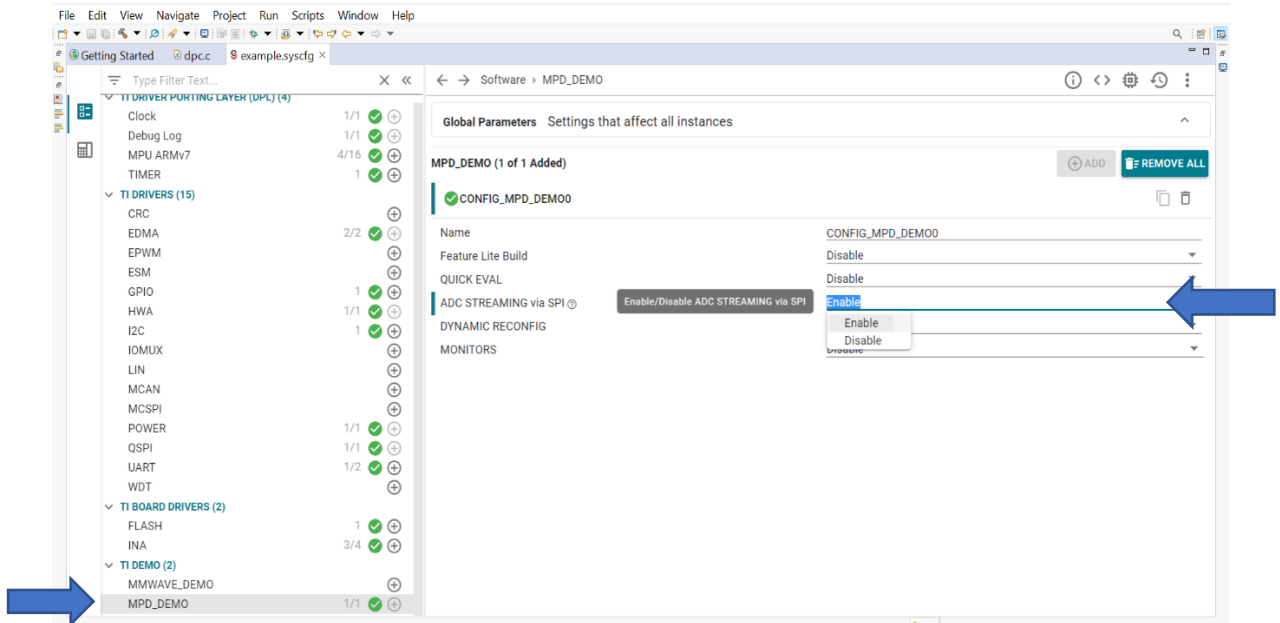
Note : If the User is flashing the attached appimage directly, skip to step 7.

1. In dpc.c file, add these two lines of code.

```
#define MAXSPISIZEFTDI (65536U)
extern uint32_t gpioBaseAddrLed, pinNumLed;
```

```
File Edit View Navigate Project Run Scripts Window Help
Getting Started example.syscfg dpc.c
Project Explorer x
motion_and_presence_detection_demo_xwrl64xx-evm_m4fs
  Generated Source
  Binaries
  Includes
  Debug
  targetConfigs
  ADC_testbuf.c
  dpc.c
  factory_calc
  interrupts.c
  linker.cmd
  main.c
  mmw_cli.c
  mmw_demo_utils.c
  mmw_flash_calc
  mmw_wave_control_config.c
  monitors
  motion_detect.c
  power_management.c
  range_phase_bias_measurement.c
  tracker_utils.c
  example.syscfg
  makefile_ccs_bootimage_gen
  README.html
  syscfg_c.rov.xs
85 #define MAX_NUM_DETECTIONS (MMWDEMO_OUTPUT_POINT_CLOUD_LIST_MAX_SIZE)
86
87 #define FRAME_REF_TIMER_CLOCK_MHZ 40
88
89 #define MMWDEMO_RFPARSER_SPEED_OF_LIGHT_IN_METERS_PER_SEC (3e8)
90
91 #define DPC_DPU_DPFI_DETMATRIX_FORMAT_2 2
92 #define MMW_DEMO_MAJOR_MODE 0
93 #define MMW_DEMO_MINOR_MODE 1
94
95 #define DPC_OBJDET_HWA_WINDOW_RAM_OFFSET 0
96 #define DPC_DPU_RANGEPROC_FFT_WINDOW_TYPE MATHUTILS_WIN_BLACKMAN
97 #define DPC_OBJDET_QFORMAT_RANGE_FFT 17
98 #define MMW_DEMO_TEST_ADC_BUFF_SIZE 1024 //maximum 128 real samples (int16_t), 3 Rx channels
99
100 #define MMWDEMO_RFPARSER_SPEED_OF_LIGHT_IN_METERS_PER_SEC (3e8)
101
102
103 #define MAXSPISIZEFTDI (65536U)
104 extern uint32_t gpioBaseAddrLed, pinNumLed;
105
106 extern MmwDemo_HSS_HCB gMmwHssHCB;
107 extern MmwWave_TemperatureStats tempStats;
108 extern uint8_t pgVersion;
109 extern float gTestMinMpdCentroid;
110
111 extern void mmwDemo_ProfilesSwitchStateMachine();
112
113 /*! L3 RAM buffer for object detection DPC */
114 #define L3_MEM_SIZE (0x40000 + 160*1024)
115 extern uint8_t gMmwL3[L3_MEM_SIZE] __attribute__((section(".L3")));
116 /*! Local RAM buffer for object detection DPC */
117 #define MMWDEMO_OBJDET_CORE_LOCAL_MEM_SIZE ((8U+6U+4U+2U+8U) * 1024U)
118 extern uint8_t gMmwCoreLocMem[MMWDEMO_OBJDET_CORE_LOCAL_MEM_SIZE];
119 /*! Local RAM buffer for tracker */
120 #define MMWDEMO_OBJDET_CORE_LOCAL_MEM2_SIZE ((8U+6U+4U+2U+8U) * 1024U)
```

2. Open example.syscfg and click MPD_DEMO. In that enable the field ADC Streaming via SPI.

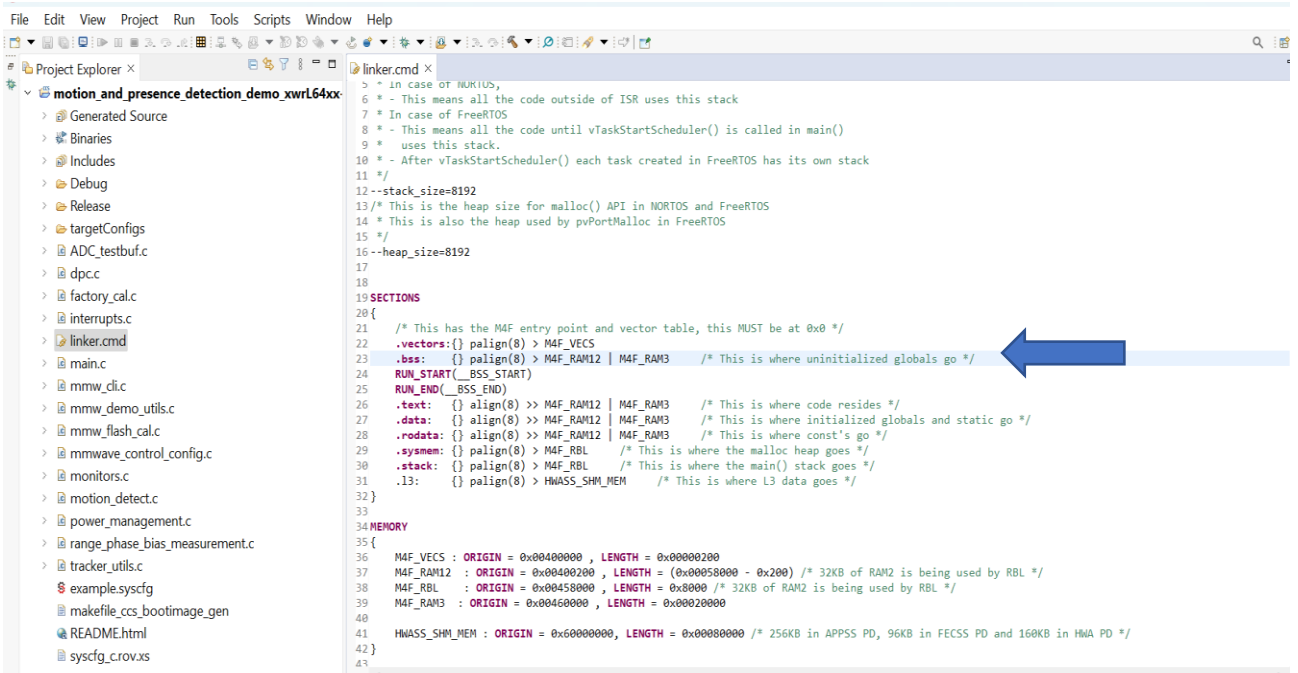


3. Disable the default configuration sent out via mmw_cli.c



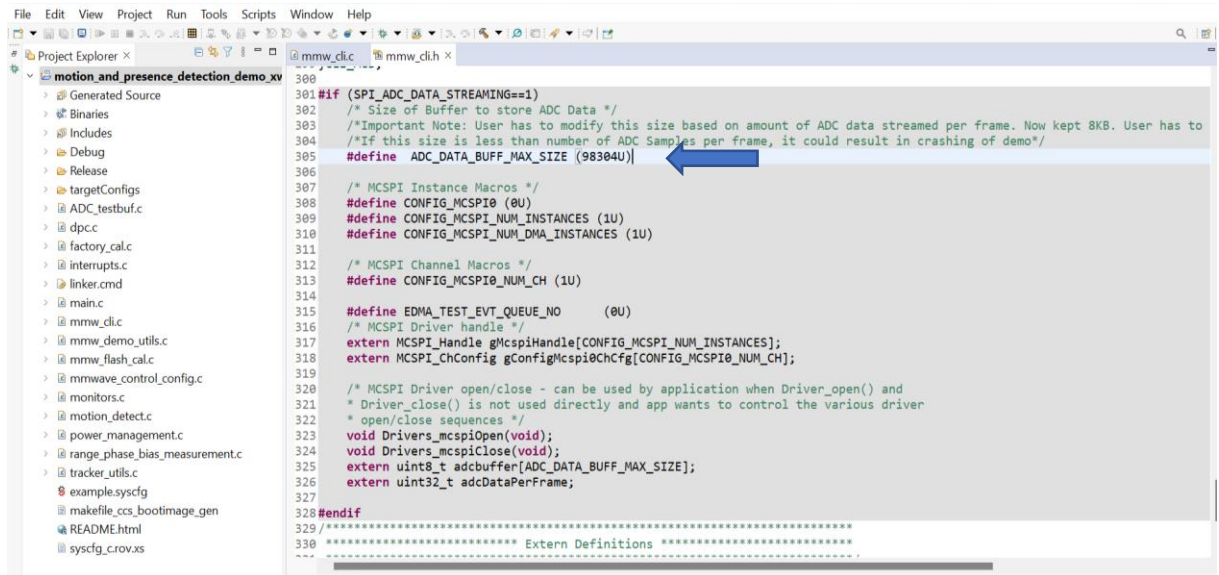
```
430 #define MAX_PRESENCE_CMD 16
431 #define MAX_RADAR_CMD 20
432 #define MAX_RADAR_CMD_AOP 21
433 char* radarCmdStrIng[MAX_RADAR_CMD] =
434 {
435 #if 0
436 "channelCfg 7 3 0 \r\n",
437 "chirpCommCfg 18 0 0 128 4 30 0 \r\n",
438 #if defined(SOC_XWR164XX)
439 "chirpTimingCfg 6.00 28 0.00 90 59.75 \r\n",
440 #else
441 "chirpTimingCfg 6.00 28 0.00 90 77 \r\n",
442 #endif
443 "frameCfg 8 0 403 1 250.0 0 \r\n",
444 "guiMonitor 0 0 0 0 0 1 0 0 0 1 \r\n",
445 "sigProcChainCfg 64 8 2 0 4 0 .5 \r\n",
446 "cFarCfg 2 4 3 2 0 12.00 0 0.5 0 1 0 1 \r\n",
447 "aoaFovCfg -60.00 60.00 -40.00 40.00 \r\n",
448 "rangeSelCfg 0.07 0.6 \r\n",
449 "clutterRemoval 1 \r\n",
450 "compRangeBiasAndRxChanPhase 0.00 1.00 0.00 -1.00 0.00 1.00 0.00 -1.00 0.00 1.00 0.00 -1.00 0.00 \r\n",
451 "adcDataSource 0 adcData_1_000.bin \r\n",
452 "adcLogging 0 \r\n",
453 "lowPowerCfg 0 \r\n",
454 "factoryCalibCfg 1 0 40 0 0x1ff000 \r\n",
455 "mpdBoundaryBox 1 -0.10 0.10 0.07 0.6 -0.10 0.10 \r\n",
456 "sensorPosition 0.00 0.00 0.00 0.00 0.00 \r\n",
457 "minorStateCfg 5 4 40 8 4 30.00 4 4 \r\n",
458 "clusterCfg 1 0.50 2 \r\n",
459 "sensorStart 0 0 0 \r\n"
460 #endif
461};
```

4. Update the linker.cmd file as described below.



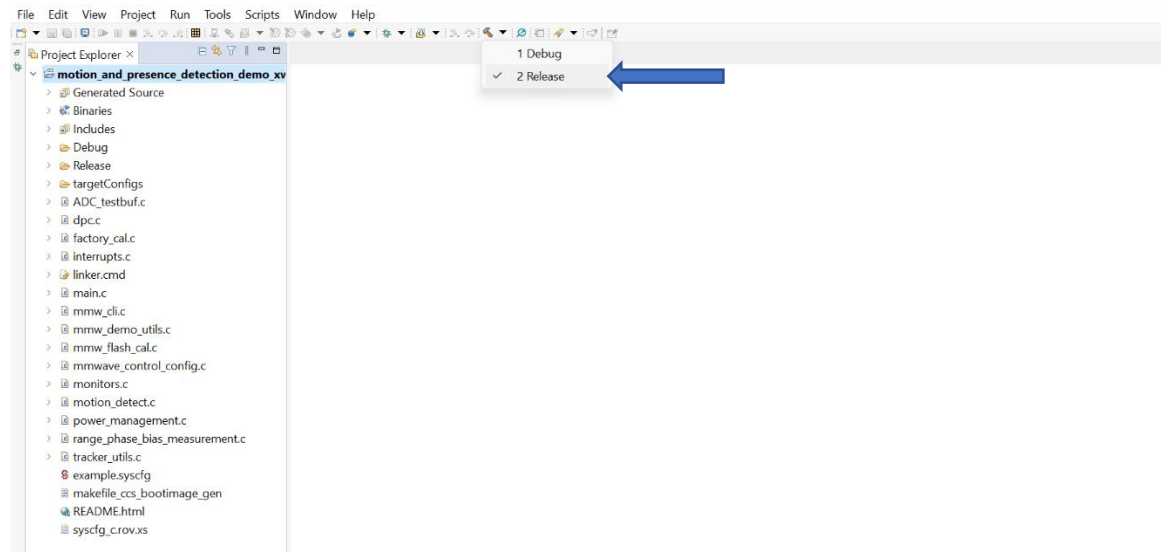
```
21 /* This has the M4F entry point and vector table, this MUST be at 0x0 */
22 .vectors: {} palign(8) > M4F_VECS
23 .bss: {} palign(8) > M4F_RAM12 | M4F_RAM3 /* This is where uninitialized globals go */
24 RUN_START(_BSS_START)
25 RUN_END(_BSS_END)
26 .text: {} align(8) >> M4F_RAM12 | M4F_RAM3 /* This is where code resides */
27 .data: {} align(8) >> M4F_RAM12 | M4F_RAM3 /* This is where initialized globals and static go */
28 .rodata: {} align(8) >> M4F_RAM12 | M4F_RAM3 /* This is where const's go */
29 .sysmem: {} palign(8) > M4F_RBL /* This is where the malloc heap goes */
30 .stack: {} palign(8) > M4F_RBL /* This is where the main() stack goes */
31 .l3: {} palign(8) > HMASS_SHM_MEM /* This is where L3 data goes */
32}
33
34 MEMORY
35{
36 M4F_VECS : ORIGIN = 0x00400000 , LENGTH = 0x0000200
37 M4F_RAM12 : ORIGIN = 0x00400200 , LENGTH = (0x00058000 - 0x200) /* 32KB of RAM2 is being used by RBL */
38 M4F_RBL : ORIGIN = 0x00458000 , LENGTH = 0x8000 /* 32KB of RAM2 is being used by RBL */
39 M4F_RAM3 : ORIGIN = 0x00460000 , LENGTH = 0x00020000
40
41 HMASS_SHM_MEM : ORIGIN = 0x60000000 , LENGTH = 0x00080000 /* 256KB in APPSS PD, 96KB in FECS5 PD and 160KB in HMA PD */
42}
43
44}
45
```

5. Update the ADC buffer size as per requirement in mmw_cli.h

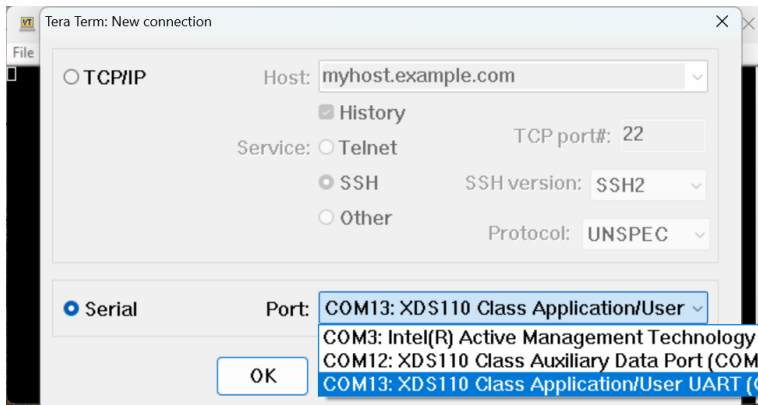


```
301 #if (SPI_ADC_DATA_STREAMING==1)
302 /* Size of Buffer to store ADC Data */
303 /*Important Note: User has to modify this size based on amount of ADC data streamed per frame. Now kept 8KB. User has to
304 /*If this size is less than number of ADC Samples per frame, it could result in crashing of demo*/
305 #define ADC_DATA_BUFFER_MAX_SIZE (98304U)
306
307 /* MCSPi Instance Macros */
308 #define CONFIG_MCSPi0 (0U)
309 #define CONFIG_MCSPi_NUM_INSTANCES (1U)
310 #define CONFIG_MCSPi_NUM_DMA_INSTANCES (1U)
311
312 /* MCSPi Channel Macros */
313 #define CONFIG_MCSPi0_NUM_CH (1U)
314
315 #define EDMA_TEST_EVT_QUEUE_NO (0U)
316 /* MCSPi Driver handle */
317 extern MCSPi_Handle gMcspiHandle[CONFIG_MCSPi_NUM_INSTANCES];
318 extern MCSPi_ChConfig gConfigMcspi0ChCfg[CONFIG_MCSPi0_NUM_CH];
319
320 /* MCSPi Driver open/close - can be used by application when Driver_open() and
321 * Driver_close() is not used directly and app wants to control the various driver
322 * open/close sequences */
323 void Drivers_mcspiOpen(void);
324 void Drivers_mcspiClose(void);
325 extern uint8_t adcbuffer[ADC_DATA_BUFFER_MAX_SIZE];
326 extern uint32_t adcDataPerFrame;
327
328 #endif
329
330 ***** Extern Definitions *****
---
```

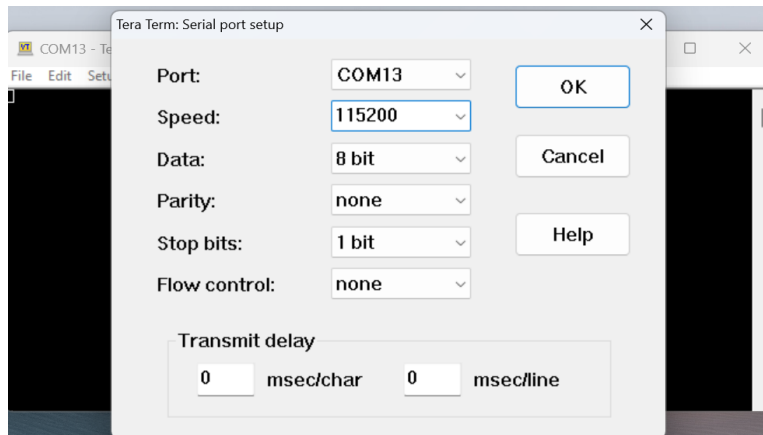
6. Ensure that the build is Release version and rebuild the project.



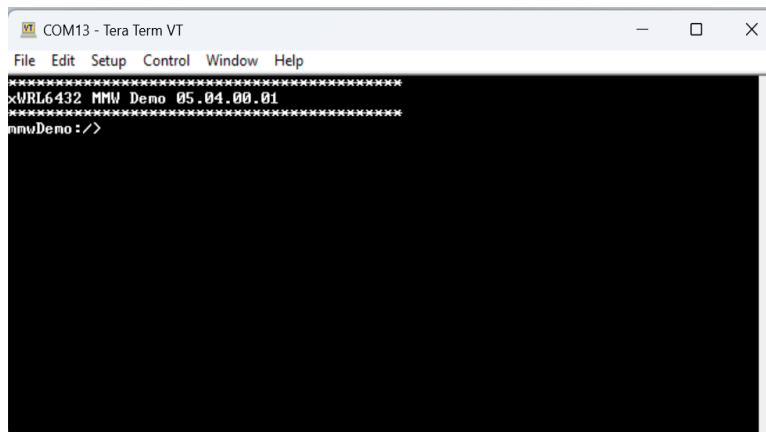
- Flash the release appimage built in the previous step.
- Setup the device for SPI transfer by ensuring switch S1.1 and S1.6 are ON.
- Open tera term and choose the corresponding COM Port.



- Setup the serial port with baud rate 115200.



- Press reset switch from FCCSP. Once pressed, tera term should look like this.



12. Paste the configuration(attached in an earlier email) to tera term. **Do not press enter after sensor start command.**

```

COM13 - Iota Term V1
File Edit Setup Control Window Help
*****
U18L6432 MPU Demo 05_04_00_01
*****
userDemo:/sensorStart 0
Done

userDemo:/channeIDfg 7 3 0
Done

userDemo:/chirpGomCfg 8 0 0 256 4 28 0
Done

userDemo:/chirpInjCfg 6 63 0 75 60
Done

userDemo:/framCfg 2 0 200 32 258 1
Done

userDemo:/spiMonitor 2 1 0 0 1 0 0 0 0
Done

userDemo:/sigProcChainCfg 32 2 1 0 4 4 0 15
Done

userDemo:/cfarCfg 2 9 4 3 0 12.0 0 0.5 0 1 1 1
Done

userDemo:/anaFavCfg -60 60 -40 40
Done

userDemo:/rangeIDfg 0.1 12.0
Done

userDemo:/clusterRemoval 1
Done

userDemo:/compRangeBiasAndBChanPhase 0.0 1.00000 0.00000 -1.00000 0.00000 1.00000 0.00000 -1.00000 0.00000 1.00000 0.00000 -1.00000 0.00000
Done

userDemo:/adcDataSource 0 C:/I/omove_ip_sdk/examples/datapath/common/testBench/major_motion/adc_data_0001_CtestAdc6dat.bin
Done

userDemo:/adcLogging 2
Done

userDemo:/lowPowerCfg 0
Done

userDemo:/factoryCalibCfg 1 0 40 0 0xffff000
Done

userDemo:/spiBoundaryBox 1 0 1.48 0 1.95 0 3
Done

userDemo:/spiBoundaryBox 2 0 1.48 0 1.95 3.9 0 3
Done

userDemo:/spiBoundaryBox 3 -1.48 0 0 1.95 0 3
Done

userDemo:/spiBoundaryBox 4 -1.48 0 1.95 3.9 0 3
Done

userDemo:/sensorPosition 0 0 1.44 0 0
Done

userDemo:/minorStateCfg 5 4 40 8 4 30 8 8
Done

userDemo:/majorStateCfg 4 2 30 10 8 80 4 4
Done

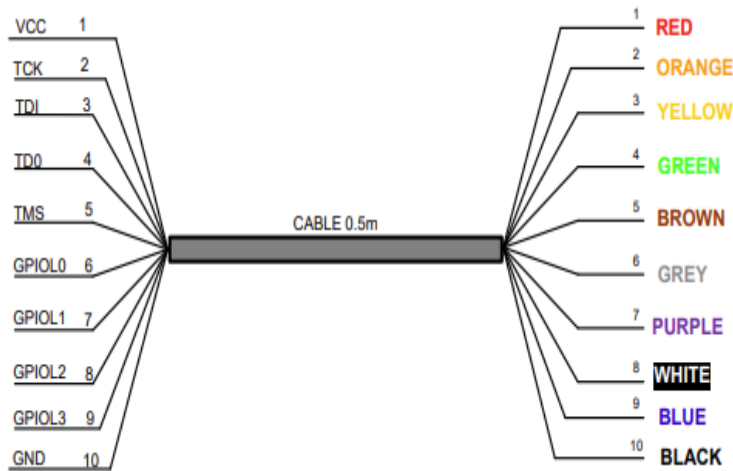
userDemo:/clusterCfg 1 0.5 2
Done

userDemo:/profileSwitchCfg 0 25 25
Done

userDemo:/sensorStart 0 0 0 0

```

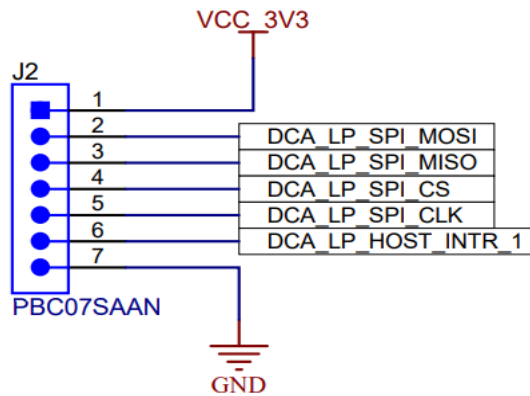
13. FCCSP EVM does not have on board SPI FTDI chip. User has to use external converter cable. We show the usage with C232HM-DDHSL-0 cable here (FT232H device) (<https://ftdichip.com/products/c232hm-ddhsl-0-2/>).



13.1 Connection Table of SPI Interface of xWRL6432 with C232HM-DDHSL-0 Cable

XWRLx4XX FCCSP Device	C232HM-DDHSL-0 Cable
MOSI	YELLOW WIRE
MISO	GREEN WIRE
CHIP SELECT	BROWN WIRE
SPI CLOCK	ORANGE WIRE
SPI BUSY	GREY WIRE
GROUND	BLACK WIRE

14. SPI Header for FTDI Interface (DCA_LP_HOST_INTR_1 is SPI BUSY SIGNAL)

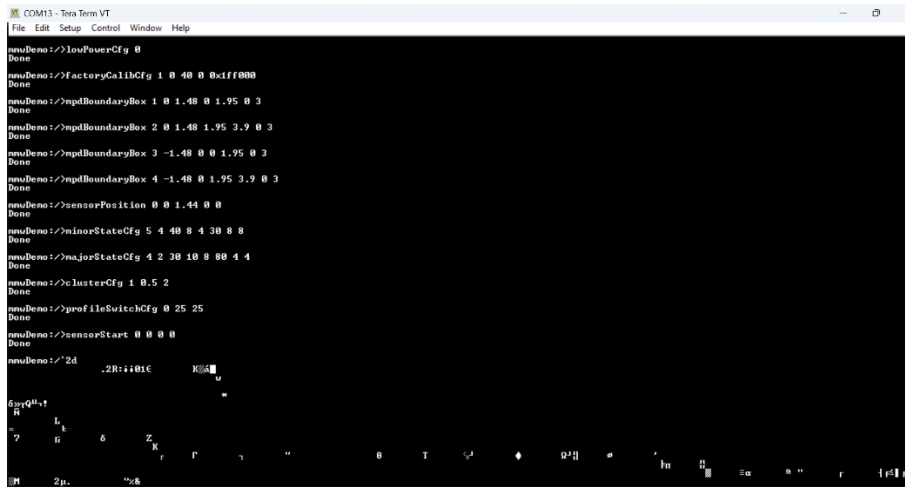


15. Open the adcDataSPIFTDI application and provide all the required parameters.

```

C:\MMWAVE_L_SDK_05_04_ x + v
"1" for AOP
"2" for FCCSP
Enter Device:
2
Enter no of adc samples:
256
Enter no of chirps per burst:
2
Enter no of bursts per frame:
32
Enter no of frames:
1
Enter no of rx antennas:
3
98304
FTDI device opened:
{'type': 8, 'id': 67330068, 'description': b'C232HM-DDHSL-0', 'serial': b'FTU7BWDG'}
Starting write loop...
    
```

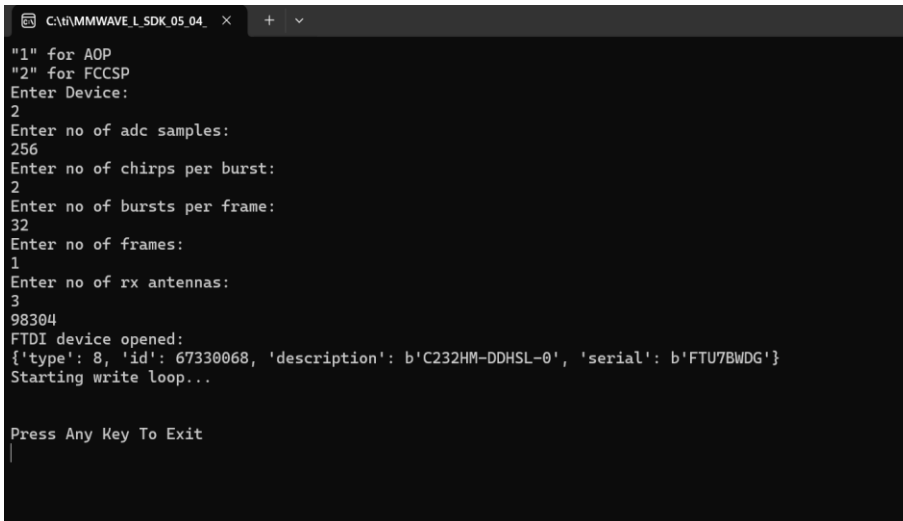
16. Now, Issue sensor start command from tera term.



```
COM13 - Tera Term VT
File Edit Setup Control Window Help
msuDemo:~>lowPowerCfg 0
Done
msuDemo:~>factoryCalibCfg 1 0 40 0 0xff000
Done
msuDemo:~>mpdBoundaryBox 1 0 1.48 0 1.95 0 3
Done
msuDemo:~>mpdBoundaryBox 2 0 1.48 1.95 3.9 0 3
Done
msuDemo:~>mpdBoundaryBox 3 -1.48 0 0 1.95 0 3
Done
msuDemo:~>mpdBoundaryBox 4 -1.48 0 0 1.95 3.9 0 3
Done
msuDemo:~>sensorPosition 0 0 1.44 0 0
Done
msuDemo:~>minorStateCfg 5 4 40 0 4 30 8 8
Done
msuDemo:~>majorStateCfg 4 2 30 10 0 80 4 4
Done
msuDemo:~>clusterCfg 1 0.5 2
Done
msuDemo:~>profileSwitchCfg 0 25 25
Done
msuDemo:~>sensorStart 0 0 0 0
Done
msuDemo:~>2d
.2R+401c K104

```

17. Open adcDataSPIFTDI application and press any key to exit.



```
C:\B\MMWAVE_1_SDK_05_04
"1" for AOP
"2" for FCCSP
Enter Device:
2
Enter no of adc samples:
256
Enter no of chirps per burst:
2
Enter no of bursts per frame:
32
Enter no of frames:
1
Enter no of rx antennas:
3
98304
FTDI device opened:
{'type': 8, 'id': 67330068, 'description': b'C232HM-DDHSL-0', 'serial': b'FTU7BWDG'}
Starting write loop...

Press Any Key To Exit

```

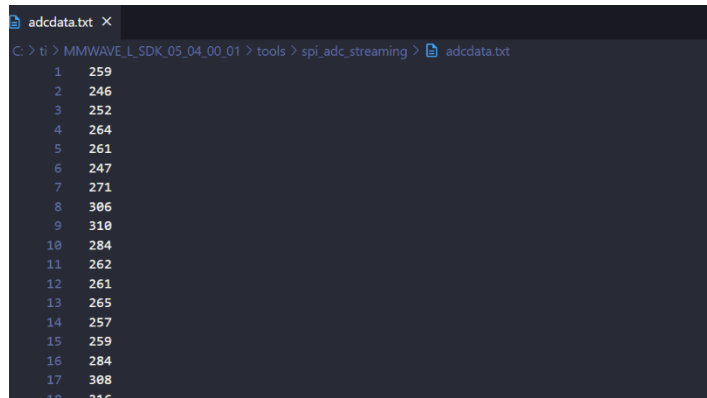
18. In the same directory, adc data would be stored in a text file.

Name	Date modified	Type	Size
source	4/23/2024 4:24 PM	File folder	
<input checked="" type="checkbox"/> adcddata	5/13/2024 11:13 AM	Text Source File	232 KB
adcDataSPIFTDI	3/4/2024 7:42 PM	Application	21,323 KB

Post Processing of Captured Data:

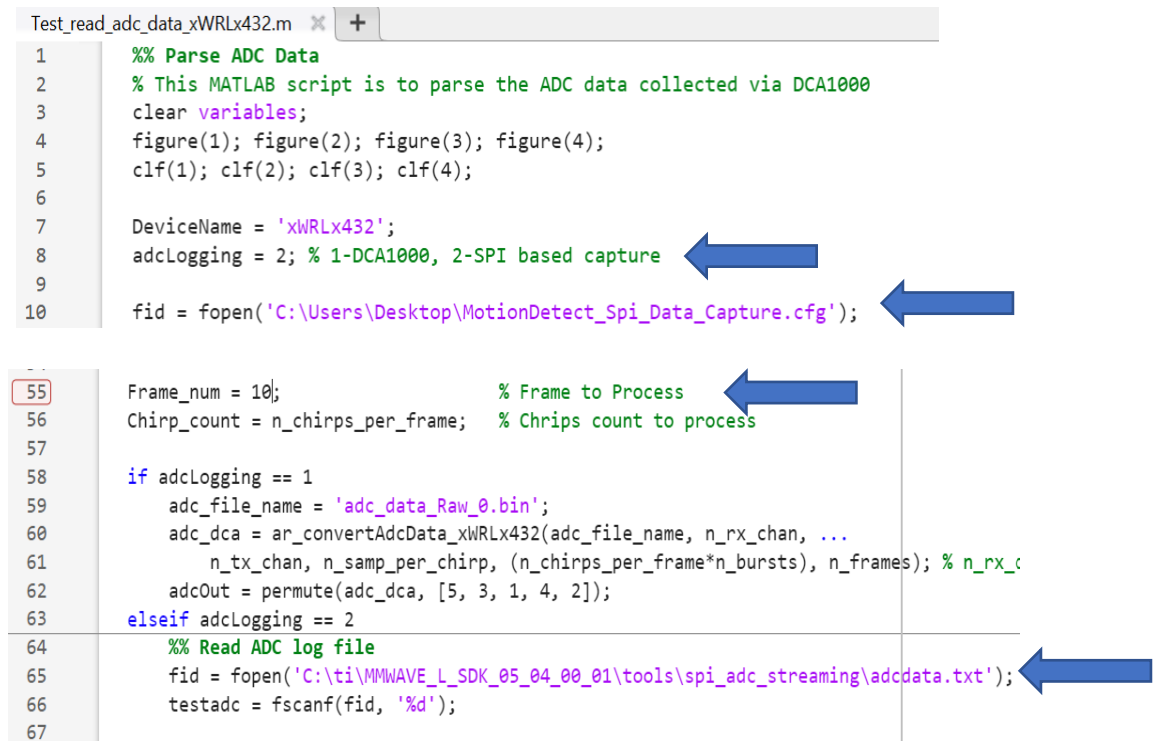
1. Data Format of the captured values in adcddata.txt file : Frame1 Chirp1 Antenna1, Frame1 Chirp1 Antenna2 and so on.

Example : 259 – Frame 1 Chirp1 Antenna1; 246 – Frame 1 Chirp1 Antenna2;



```
adcddata.txt
C:\ti> MMWAVE_L_SDK_05_04_00_01 > tools > spi_adc_streaming > adcddata.txt
1 259
2 246
3 252
4 264
5 261
6 247
7 271
8 306
9 310
10 284
11 262
12 261
13 265
14 257
15 259
16 284
17 308
```

2. The captured data can be used to perform post processing with the Matlab script provided in {SDK_INSTALL_DIR}/tools/ADC_parser/Test_read_adc_data_xWRLx432.m
3. Once the above mentioned script is opened in Matlab, ensure that adclogging is 2 in the script. Also, provide the path for adcddata.txt file and configuration file used. Only one frame could be processed with the script. The specific frame number to be processed should be indicated in the script.



```
Test_read_adc_data_xWRLx432.m
1 %% Parse ADC Data
2 % This MATLAB script is to parse the ADC data collected via DCA1000
3 clear variables;
4 figure(1); figure(2); figure(3); figure(4);
5 clf(1); clf(2); clf(3); clf(4);
6
7 DeviceName = 'xWRLx432';
8 adcLogging = 2; % 1-DCA1000, 2-SPI based capture
9
10 fid = fopen('C:\Users\Desktop\MotionDetect_Spi_Data_Capture.cfg');

...

55 Frame_num = 10; % Frame to Process
56 Chirp_count = n_chirps_per_frame; % Chrips count to process
57
58 if adcLogging == 1
59     adc_file_name = 'adc_data_Raw_0.bin';
60     adc_dca = ar_convertAdcData_xWRLx432(adc_file_name, n_rx_chan, ...
61         n_tx_chan, n_samp_per_chirp, (n_chirps_per_frame*n_bursts), n_frames); % n_rx_c
62     adcOut = permute(adc_dca, [5, 3, 1, 4, 2]);
63 elseif adcLogging == 2
64     %% Read ADC log file
65     fid = fopen('C:\ti\MMWAVE_L_SDK_05_04_00_01\tools\spi_adc_streaming\adcddata.txt');
66     testadc = fscanf(fid, '%d');
67
```

4. Run the script to view the output figures.

Sample Output Figures:

