MICROCHIP AN5454 Video Streaming with SDR Using Polar Fire Instructions

Contents

MICROCHIP AN5454 Video Streaming with SDR Using Polar Fire

Introduction (Ask a Question)

This application note describes the implementation of Software Defined Radio on PolarFire® using an AD9371 radio transceiver, a dual camera sensor, and an HDMI monitor. The demo illustrates the transmission of live video wirelessly from a transmitter to a receiver setup and displays the received video on an HDMI monitor. This solution is developed on Microchip's PolarFire Video kit, which features an MPF300TS PolarFire FPGA. A Software-Defined Radio (SDR) is a versatile technology used across numerous communication systems due to its adaptive capabilities. Unlike traditional radios, which rely on fixed hardware, SDRs leverage software for signal processing and management. A well-designed SDR typically comprises various fixed components, such as an antenna, front-end RF hardware, and an Analog-to-Digital Converter (ADC) or a Digital-to-Analog Converter (DAC).

The video is captured with the camera sensor at the transmitter end. The captured data undergoes various image processing techniques such as adaptive brightness and image enhancement. Following this, the data is compressed using JPEG and then transmitted to the SDR Tx. When the RF signal is received, it is demodulated by another SDR or compatible receiver to extract the original video data. This data is encoded and then decoded with a JPEG Decoder. The receiver transfers the video data to a monitor using a High Definition Multimedia Interface (HDMI) connection. HDMI is a widely used interface that provides high-quality digital video and audio.

The demo demonstrates the following features:

- Supports wireless transmission of video between two hardware setups
- Supports configuration of AD9371 RF board using Mi-V soft processor on FPGA device
- JPEG compression of live video captured from MIPI CSI2 camera with HD resolution
- Transmission of compressed video through SDR
- Display the received video on the Rx setup over HDMI
- Provides HD resolution video transfer with 19.2 Mbps speed
- Provides carrier frequency and phase offset correction using preamble-based auto-correlation method
- Supports FEC encoding to correct data transmission errors
- Supports Custom Packetization and Depacketization method

The PolarFire Video kit enables prototyping of Video and Imaging solutions with support for the following interfaces:

- MPFS300T PolarFire Video Kit device
- MIPI CSI-2 interface
- Dual Sony Camera
- FPGA Mezzanine Card (FMC) connector
- LPDDR4, and DDR4 memories
- HDMI, camera, and other interfaces

For more information about the PolarFire Video kit, see the PolarFire FPGA Video and Imaging Kit with MPF300T Device.

Demo Requirements

The following table lists the hardware and software requirements for running the demo.

Table 1-1. Demo Requirements

Requirement	Description
Hardware and Accessories	
PolarFire® Video Kit	**MPF300-VIDEO-KIT-NS** kit Contents: • PolarFire Video and Imaging Board with MPF300T-1FCG1152E Device • Dual Camera Sensor board – VIDEO-DC-DUALCAM • HDMI cable • 12V power pack/AC adapter • USB 2.0 A male to mini-B type Two Two-for-two board demo and one-for-one board demo
Image Sensor module	LI-IMX334-MIPI-MICRO v1.0
SMA cable	Male to male Required to provide the reference clock from the clock generator to AD9528 on the AD9371 FMC card.
Clock Source	Required to provide 30.72 MHz frequency for the reference clock to AD9371
AD9371	Analog Devices Radio Frequency (RF) board for transceiver functionality and digital signal processing function Two Two-for-two board demo and one-for-one board demo
Host PC	A host PC with a USB port Two Two-for-two board demo and one-for-one board demo
HDMI Monitor	To display the processed resultant video data for an HD resolution (1280 x 720)
Omnidirectional antenna	Two antennas are required for transmitting or receiving the radio signal • One for transmitting • One for receiving
Utility Software	
FlashPro Express v2023.1	To program the .job file on the PolarFire FPGA device
Programming job file	mpf_an5454_v2024p2_df.job

Demo Prerequisites (Ask a Question)

Before you start the demo, ensure that the following components are in place:

- Download the demo design files from the following link: AN5454: Video Streaming with SDR using PolarFire and AD9371.
- The programming job file is placed at
- <$download_folder>\mpf_an5454_v2024p2_df\Programming_File.

Demo Design (Ask a Question)
The Libero® design contains both the transmitter (Tx) path and the receiver (Rx) path. The communication with the analog radio chip, AD9371, happens through JESD204B. As the design has both transmitter and receiver paths, the demo is operated in the following two modes:

- Two boards communication
- Same board communication

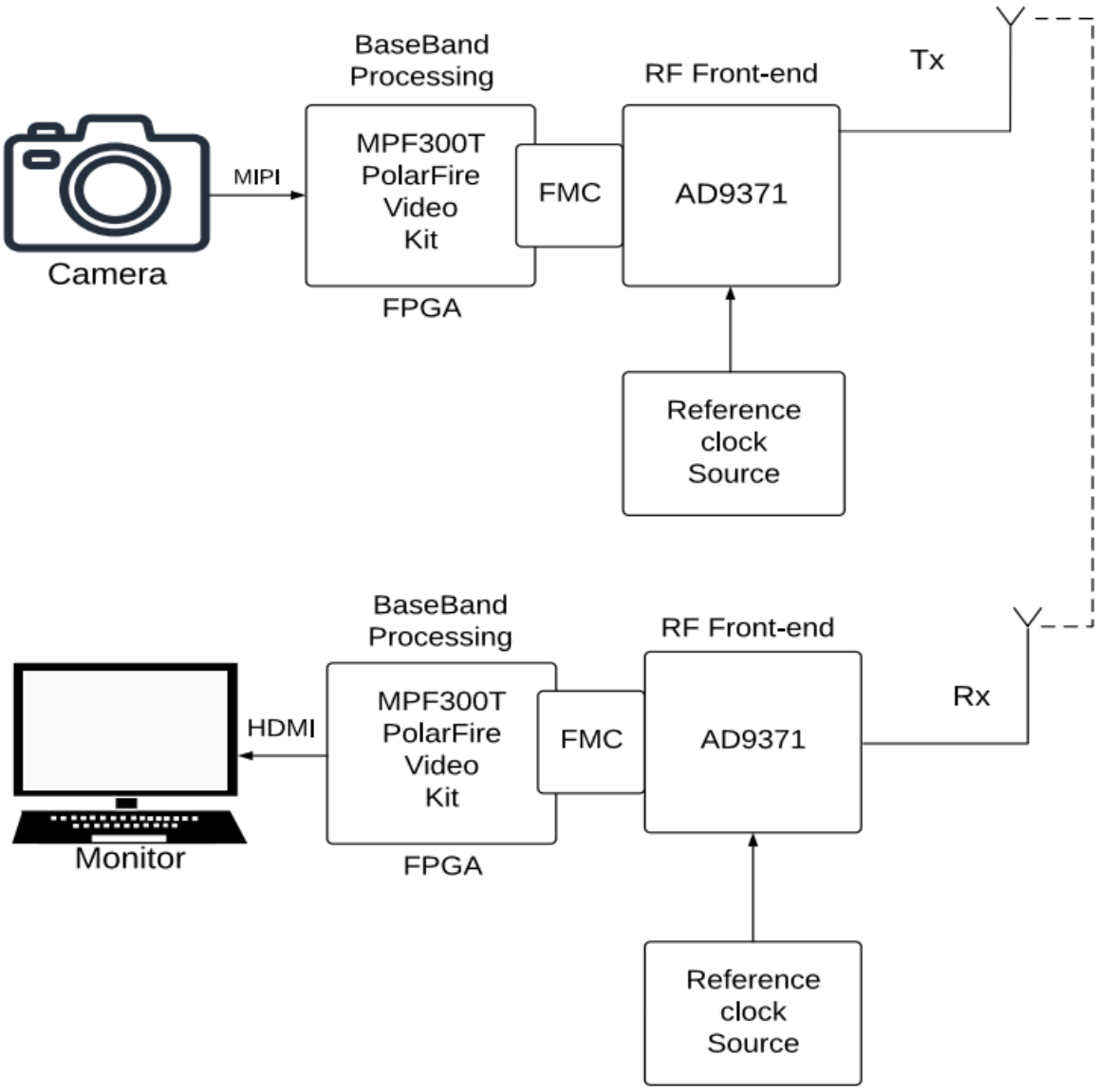
Two Boards Communication Setup (Ask a Question)

The setup for the two boards' communication, requires the following equipment:

- Two PolarFire FPGA video kits
- Two AD9371 FMC cards
- Two clock generators
- One camera sensor
- One HDMI monitor

For more instructions on setting up the demo, see Setting up the Demo section. The camera captures data, which is then transmitted through the MIPI interfaces and compressed using JPEG encoding. The encoded data undergoes processing within the Image Signal Processing (ISP) blocks. After this, the data is sent to the SDR transmitter, where it is segmented into packets of a predefined size and wirelessly transmitted. At the receiving end, the SDR receiver captures the transmitted data, decodes it with a JPEG Decoder, and finally sends the video data to the HDMI monitor. The following figure shows the two-board communication setup.

Figure 3-1. Two Boards Communication



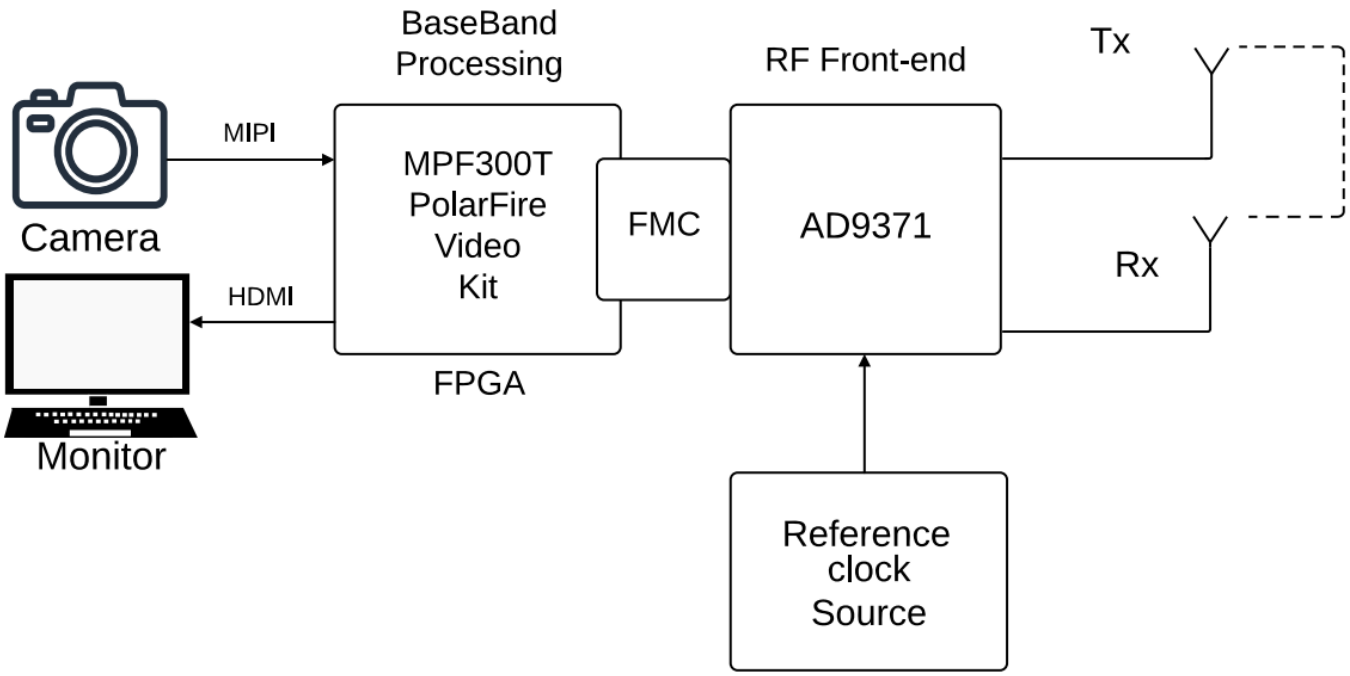
Same Board Communication Setup (Ask a Question)

The setup for the same board communication requires the following equipment:

- One PolarFire FPGA video kit
- One AD9371 FMC card
- One clock generator
- One camera sensor
- One HDMI monitor

The camera sensor captures video data which is then processed by the PolarFire FPGA. The FPGA device processes the video data and prepares it for transmission. The AD9371 FMC card modulates the processed video data and transmits it over the air through a radio frequency signal. At the receiving end, the same FMC card demodulates the incoming RF signal to retrieve the video data. Subsequently, the PolarFire FPGA processes this data and outputs it to an HDMI monitor, enabling the end-user to view the transmitted video content. The following figure shows the same board communication setup.

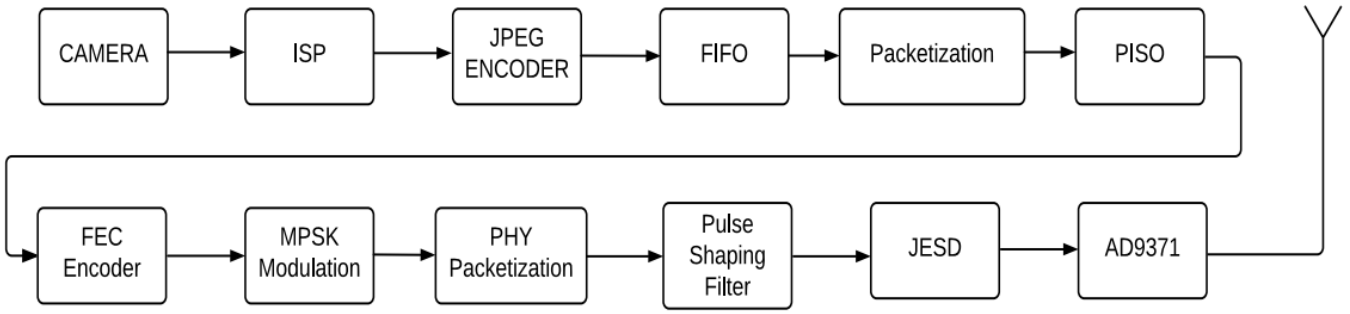
Figure 3-2. Same Board Communication



Transmitter (Ask a Question)

The following figure depicts the process flow of an SDR transmitter. A camera sensor initially captures video data, which undergoes refinement through ISP techniques such as Bayer interpolation, Gamma correction, image enhancement, and auto exposure. The data is then converted from RGB to YCbCr format. This processed data is compressed using a JPEG Encoder and then aligned through a FIFO buffer to ensure synchronization with the packetization process. The packetization involves the addition of headers, identification, access codes, and a CRC for error checking. The data is serialized into bits and further protected by Forward Error Correction (FEC) to enable error detection and correction at the receiver's end.

Figure 3-3. Process Flow of a Transmitter



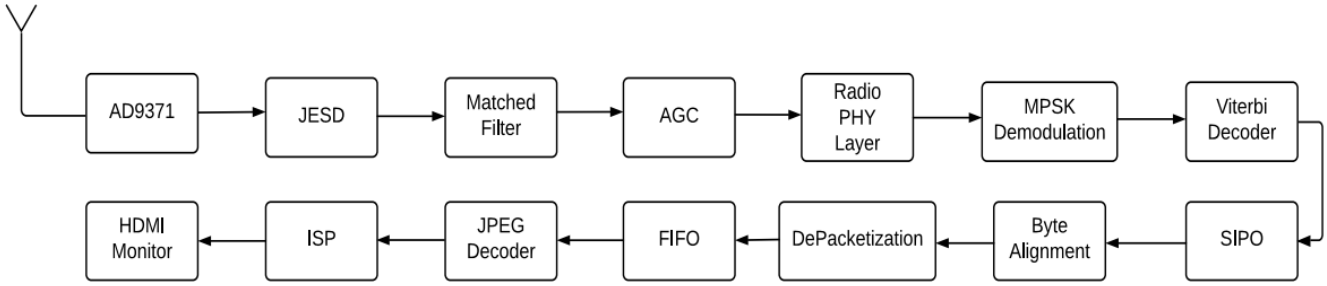
The FEC-encoded data is digitally modulated into I/Q symbols using QPSK at the physical layer, which includes a preamble and pilot symbols to adjust for frequency and phase discrepancies. To conform to emission regulations and reduce interference, thereby enhancing the signal-to-noise ratio, a Raised-Cosine filter is used for spectrum shaping. Subsequently, the refined I/Q baseband signals are transmitted through the high-speed JESD interface to the AD9371, where they are combined with the high-frequency RF signal. The resultant message signal is then broadcasted wirelessly through an antenna.

Receiver (Ask a Question)

An SDR receiver captures data through an antenna and subsequently processes it using the AD9371 RF chip. This chip demodulates the signal and then transmits it to an FPGA in I/Q format using the JESD protocol. To reduce noise, a matched filter is applied. Automatic Gain Control (AGC) is crucial for maintaining consistent signal

levels to ensure accurate decoding.
The following figure shows the process flow of an anDR receiver.

Figure 3-4. Process Flow of a Receiver



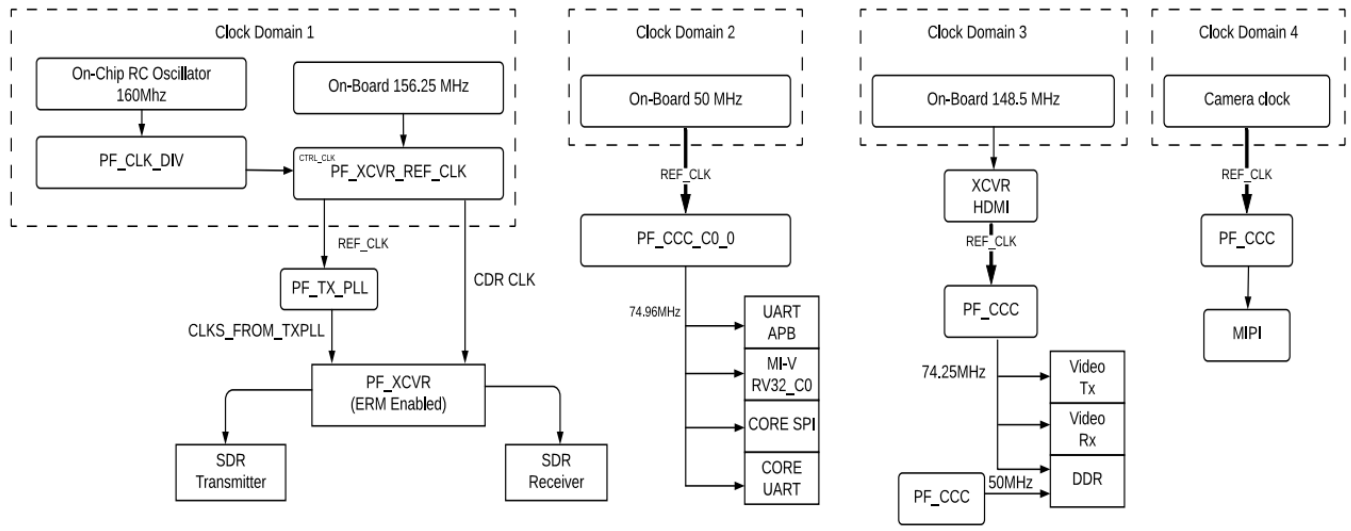
The I/Q samples are oversampled by a factor of 8 and then filtered. Auto-correlation is used on the oversampled data to find the right timing for sampling the preamble and packet symbols, with multiple auto-correlation instances staggered by one clock cycle. Any frequency and phase offsets identified by auto-correlation are corrected by adjusting the I/Q values.

The Physical Layer (PHY) encompasses the procedures leading up to the acquisition of clean In-phase/Quadrature (I/Q) values, which symbolize the transmitted data and are essential for demodulation and data bit extraction. To correct bit errors resulting from channel distortion, a Viterbi decoder is used, which yields one corrected bit for every two bits demodulated.

The data undergoes a transformation from bits to bytes through a Serial-In-Parallel-Out (SIPO) converter, with packet header access codes ensuring correct alignment. The depacketization module then authenticates the data by comparing it to the receiver's ID and conducts error checking using CRC. Upon successful ID match, the data is queued in a FIFO buffer. Subsequently, it is decoded with a JPEG decoder, converted from YCbCr to RGB format, and transmitted to an HDMI monitor using the HDMI protocol, resulting in the display of the captured video.

Clocking Structure (Ask a Question)

The PF_CCC generates the 50 MHz fabric clocks from the REF_CLK generated by PF_XCVR_REF_CLK. The onboard 156.25 MHz clock is the reference clock for the XCVR, which further provides the operating clock of 153.6 MHz for the TX_Path and RX_Path blocks. The onboard 148.5 MHz onboard oscillator provides a reference clock to generate clocks used in the video pipeline.
The following figure shows the clocking structure of the design.



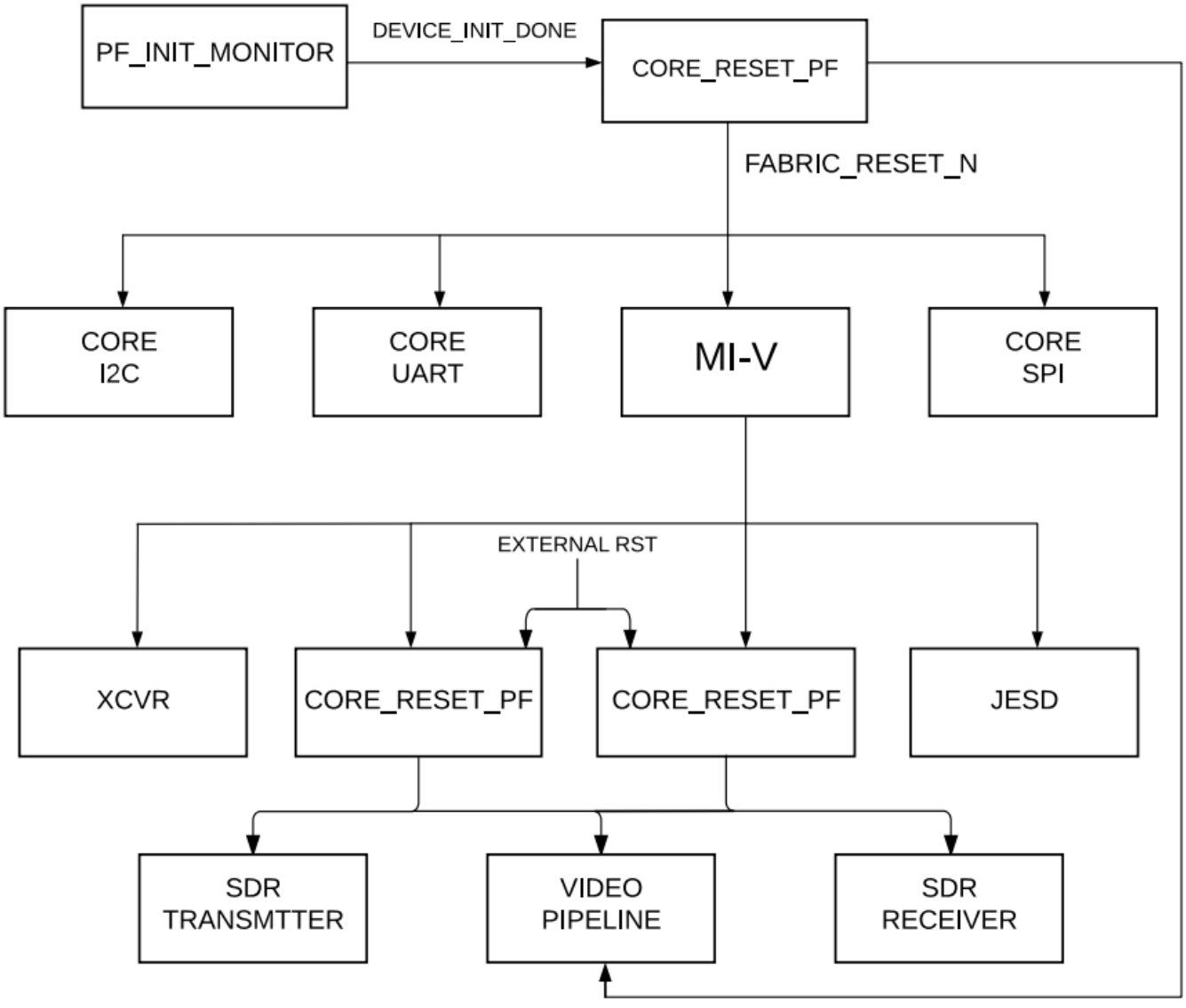
Reset Structure (Ask a Question)

The INIT_MONITOR IP asserts the following signals:

- FABRIC_RESET_N: Asserted after the initialization of the fabric, whereas FABRIC_POR_N is used to reset the MSS.
- DEVICE_INIT_DONE: Asserted after the initialization of the PolarFire SoC device.
- EXTERNAL RESET: Asserted when the user wants to reset the design, externally.

The following figure shows the reset structure of the design.

Figure 3-6. Reset Structure



Resource Utilization (Ask a Question)

The following figure shows the summary of resource utilization and performance data.

Figure 3-7. Resource Utilization and Performance

Module Name	Fabric 4LUT	Fabric DFF	Interface 4LUT	Interface DFF	Single-Ended I/O	Differential I/O Pairs	uSRAM (64x12)	LSRAM (20K)	Math (18x18)	Chip Globals	Row Global	PLL	DLL	Transceiver Lanes
Primitives	2	24	0	0	22	0	0	0	0	1	0	0	0	0
⊞ Proc_Sub_System_inst_0	9572	3900	19536	19536	2	0	20	536	0	5	0	1	0	0
⊟ SDR_inst_0	36639	25201	10440	10440	0	3	114	12	240	8	4	2	0	2
Primitives	1	0	0	0	0	1	0	0	0	0	0	0	0	0
⊞ JESD_0	2983	2811	384	384	0	2	32	0	0	3	4	0	0	2
⊟ RX_Path	32549	21395	8544	8544	0	0	82	4	206	4	0	1	0	0
Primitives	4	0	0	0	0	0	0	0	0	1	0	0	0	0
BYTE_ALLIGNMEN...	109	43	0	0	0	0	0	0	0	0	0	0	0	0
⊞ COREFIFO_C1_0	174	182	72	72	0	0	0	2	0	0	0	0	0	0
⊞ COREFIR_PF_C1_I	36	67	2328	2328	0	0	2	0	64	0	0	0	0	0
⊞ COREFIR_PF_C1_Q	36	65	2328	2328	0	0	2	0	64	0	0	0	0	0
⊞ CORERESET_PF_C4_0	1	16	0	0	0	0	0	0	0	0	0	0	0	0
MPSK_Demod_v1_0	2	5	0	0	0	0	0	0	0	0	0	0	0	0
⊞ PF_CCC_C3_0	0	0	0	0	0	0	0	0	0	2	0	1	0	0
⊞ SIGNAL_Condition...	1175	885	216	216	0	0	0	0	6	0	0	0	0	0
SIPO_0	24	34	0	0	0	0	0	0	0	0	0	0	0	0
⊞ Viterbi_Decoder_C...	13366	6317	0	0	0	0	0	0	0	1	0	0	0	0
cdc_outfifo_0	0	3	0	0	0	0	0	0	0	0	0	0	0	0
data_depacketizati...	339	96	0	0	0	0	0	0	0	0	0	0	0	0
frame_rst_0	7	26	0	0	0	0	0	0	0	0	0	0	0	0
⊞ radio_PHY_layer_0	17261	13645	3600	3600	0	0	78	2	72	0	0	0	0	0
start_gen_0	15	11	0	0	0	0	0	0	0	0	0	0	0	0
⊟ TX_Path	1017	831	1512	1512	0	0	0	8	34	1	0	1	0	0
Primitives	2	1	0	0	0	0	0	0	0	0	0	0	0	0
⊞ COREFIFO_C0_0	186	177	288	288	0	0	0	8	0	0	0	0	0	0
⊞ COREFIR_PF_C0_I	115	135	612	612	0	0	0	0	17	0	0	0	0	0
⊞ COREFIR_PF_C0_Q	116	142	612	612	0	0	0	0	17	0	0	0	0	0
⊞ CORERESET_PF_C4...	1	16	0	0	0	0	0	0	0	0	0	0	0	0
DATA_RATE_GENER...	13	9	0	0	0	0	0	0	0	0	0	0	0	0
⊞ FEC_Encoder_C0_0	5	9	0	0	0	0	0	0	0	0	0	0	0	0
MPSK_Mod_C0_0	0	5	0	0	0	0	0	0	0	0	0	0	0	0
⊞ PF_CCC_C1_0	0	0	0	0	0	0	0	0	0	1	0	1	0	0
PISO_0	61	45	0	0	0	0	0	0	0	0	0	0	0	0
UPSAMPLING_0	34	54	0	0	0	0	0	0	0	0	0	0	0	0
cdc_reg_0	0	36	0	0	0	0	0	0	0	0	0	0	0	0
data_packetization_0	177	67	0	0	0	0	0	0	0	0	0	0	0	0
phy_packetization_0	307	135	0	0	0	0	0	0	0	0	0	0	0	0
⊞ UART_interface_0	89	164	0	0	0	0	0	0	0	0	0	0	0	0
⊟ VIDEO_PIPELINE	38553	31897	12156	12156	65	10	62	232	85	14	4	4	1	4
Primitives	0	1	0	0	0	0	0	0	0	0	0	0	0	0
⊞ DDR_ARB_WR_RD_0	22120	21520	3120	3120	65	5	44	72	0	4	0	1	1	0
⊞ IMX334_IF_TOP_0	2573	3024	396	396	0	5	0	11	0	4	0	1	0	0
⊞ Video_pipeline_RST_0	5	68	0	0	0	0	0	0	0	4	0	2	0	0
⊟ Video_pipeline_RX_0	6744	3477	4428	4428	0	0	12	80	39	2	4	0	0	4
⊞ Display_Controller...	230	160	36	36	0	0	0	1	0	0	0	0	0	0
⊞ HDMI_2P0_TX_0	522	364	72	72	0	0	6	0	0	0	4	0	0	4
⊞ JPEG_Decoder_C0_0	5856	2866	4140	4140	0	0	6	79	34	2	0	0	0	0
⊞ YCbCrtoRGB_C0_0	136	87	180	180	0	0	0	0	5	0	0	0	0	0
⊟ Video_pipeline_TX_0	7111	3807	4212	4212	0	0	6	69	46	0	0	0	0	0
⊞ Bayer_Interpolatio...	729	339	108	108	0	0	0	3	0	0	0	0	0	0
⊞ Display_Controller...	79	35	0	0	0	0	0	0	0	0	0	0	0	0
⊞ Gamma_Correctio...	448	22	0	0	0	0	0	0	0	0	0	0	0	0
⊞ Image_Enhanceme...	0	26	108	108	0	0	0	0	3	0	0	0	0	0
⊞ JPEG_Encoder_C0_0	5714	3179	3672	3672	0	0	6	66	34	0	0	0	0	0
⊞ RGBtoYCbCr_C0_0	106	72	324	324	0	0	0	0	9	0	0	0	0	0
delay_module_0	1	68	0	0	0	0	0	0	0	0	0	0	0	0
intensity_average_0	34	66	0	0	0	0	0	0	0	0	0	0	0	0

Ports (Ask a Question)

The following table lists the ports of the demo design.

Table 3-1. Input and Output Ports

Port Name	Direction	Description
SDR Ports		
RESET_SW_N_I	Input	Resethe t the switch for the Design
REF_CLK_PAD_N	Input	XCVR Reference clock 156.25 MHz
REF_CLK_PAD_P		
DAC_SYNC_N	Input	Sync signal from AD9371
DAC_SYNC_P		
LANE0_RXD_N	Input	Transceiver lanes connected to AD9371
LANE0_RXD_P		
LANE1_RXD_N		
LANE1_RXD_P		
PADN	Input	SYSREF signal from AD9371 FMC card
PADP		
ADC_SYNC_N	—	Sync signal to AD9371
ADC_SYNC_P		
LANE0_TXD_N	Output	Transceiver lanes connected to AD9371
LANE0_TXD_P		
LANE1_TXD_N		
LANE1_TXD_P		
MSS Peripheral Ports		
SPI_SDI	Input	SPI data input
TCK	Input	CoreJTAG pin
TDI	Input	CoreJTAG pin
TMS	Input	CoreJTAG pin

……..continued

Port Name	Direction	Description
TRSTB	Input	CoreJTAG pin
GPIO_11_IN	Input	Connected to AD9371
GPIO_10_IN	Input	Connected to AD9371
SPI_SCLK	Output	SPI clock
SPI_SDO	Output	SPI data out

SPI_SS1	Output	SPI chip select
TDO	Output	CoreJTAG pin
CAM1_SCL	Output	Camera control pins
CAM1_SDA	Output	Camera control pins
GPIO_OUT	Output	GPIO[4-9] → Connected to AD9371
CAM1_EN	Output	Camera enable pin
Video Pipeline Ports		
CAM1_RX_CLK_N	Input	Input pads to receive the reference clock for the camera sensor module
CAM1_RX_CLK_P		
CAM1_RXD_N[3:0]	Input	Input pads to receive the live video from the camera sensor module. These pads are assigned to the Bank 7 I/Os.
CAM1_RXD_P[3:0]		
HDMI_REF_CLK_PAD_N	Input	Input pads to receive the reference clock for the HDMI XCVR
HDMI_REF_CLK_PAD_P		
REF_CLK_0	Input	Reference clock for the video pipeline CCC
CAMERA_CLK	Output	Camera clock
HDMI_LANE0_TXD_N	Output	HDMI XCVR lanes
HDMI_LANE0_TXD_P		
HDMI_LANE1_TXD_N		
HDMI_LANE1_TXD_P		
HDMI_LANE2_TXD_N		
HDMI_LANE2_TXD_P		
HDMI_LANE3_TXD_N		
HDMI_LANE3_TXD_P		

……..continued		
Port Name	Direction	Description
ACT_N	Output	DDR ports
BG		
CAS_N		
CK0_N		
CK0		
CKE		
CS_N		
ODT		
RAS_N		
RESET_N		
SHIELD0		
SHIELD1		
SHIELD2		
SHIELD3		
WE_N		
A[13:0]		
BA[1:0]		
DM_N[3:0]		
DQS_N[3:0]		
DQ[31:0]		

Setting Up the Demo (Ask a Question)

The demonstration involves the following steps:

- Setting up the Hardware
- Same Board Communication
- Two Boards Communication
- Programming the Device

Setting Up the Hardware (Ask a Question)

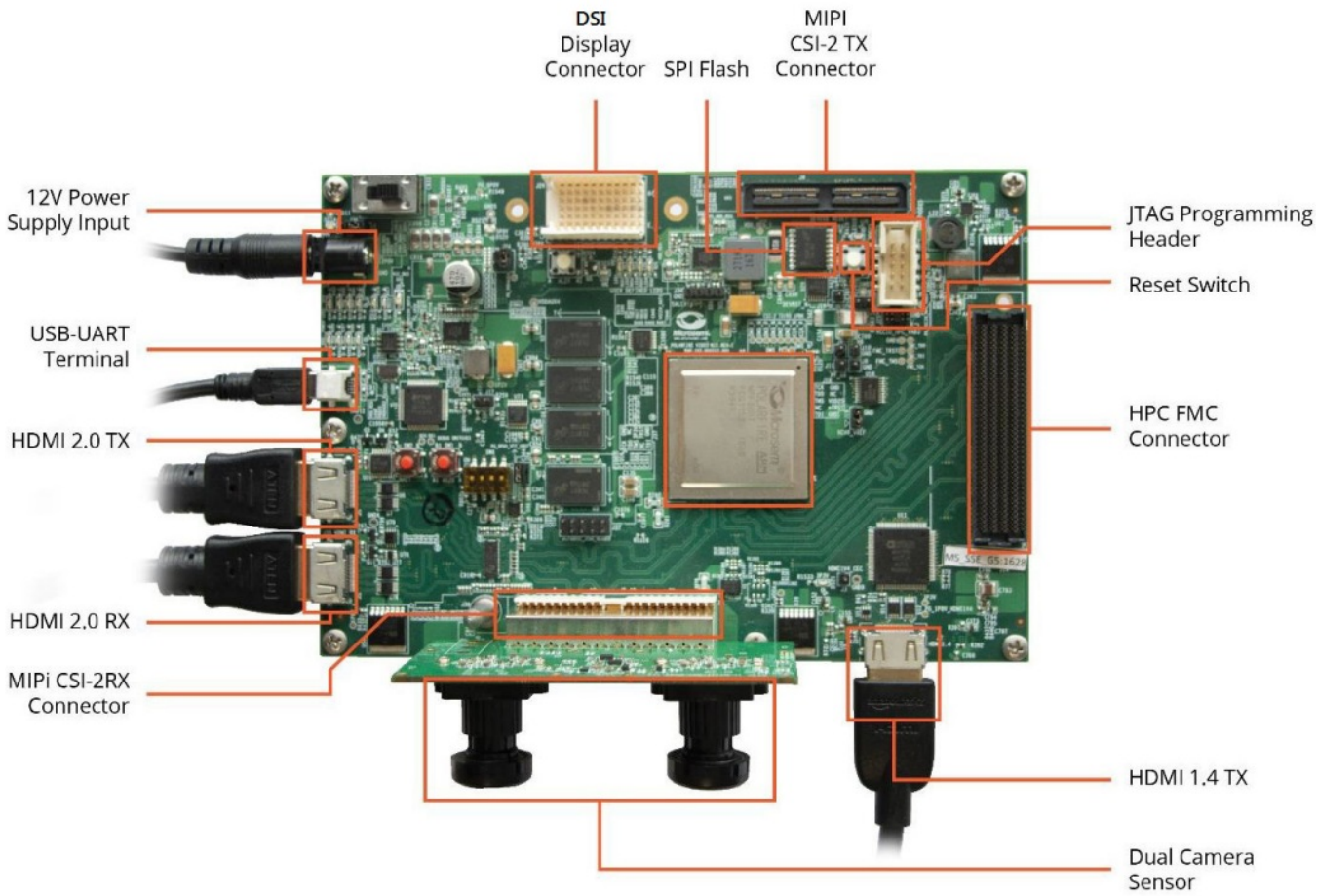
After generating the bitstream, the PolarFire device is programmed on the Video kit. Ensure that the jumper settings on the Video kit board are the same as listed in the following table.

Table 4-1. Jumper Settings for Video Kit

Jumper	Default Position	Functionality
J15	Open	SPI Slave and Master mode selection. By default, SPI master is selected.
J17	Open	100K PD for TRSTn. By default, 1K PD is connected
J19	Pin 1 and 2	Default: XCVR_VREF is connected to GND
J28	Pin 1 and 2	Default: Programming through the FTDI
J24	Pin 2 and 4	Default: VDDAUX4 voltage is set to 3V3
J25	Pin 5 and 6	Default: Bank4 voltage is set to 1V8
J36	Pin 1 and 2	Default: Board power up through the SW4
SW6	OFF	Default: 700 MHz carrier frequency is selected
SW6	ON	User DIP switch
SW4	OFF	Power ON/OFF switch
J20	12V Input	12V input to the board

The following figure shows the board setup for the mentioned jumper settings.

Figure 4-1. PolarFire Video Kit Jumper Settings



1. Connect the camera sensor to J5 on the PolarFire Video kit.

2. Connect the AD9371 to the J14 FMC connector.
3. Connect the signal generator by setting it to 30.72 MHz and 1.8 Vpp to the J401 of the AD9371.
4. Connect one omnidirectional antenna to the J305 (TX1 SMA connector) and the other antenna to the J200 (RX1 SMA connector) of AD9371.
5. Connect the HDMI cable to the J1 (HDMI 2.0 Tx port) of the PolarFire Video kit.
6. Connect the host PC to the PolarFire Video kit through J12 using a USB Mini cable.
7. Connect the power supply cable to J20 of the PolarFire Video kit.
8. Power up the HDMI monitor.
9. Power up the PolarFire Video kit using the SW4 slide switch.

Two Boards Communication (Ask a Question)

To set up two boards communication, perform the following steps:

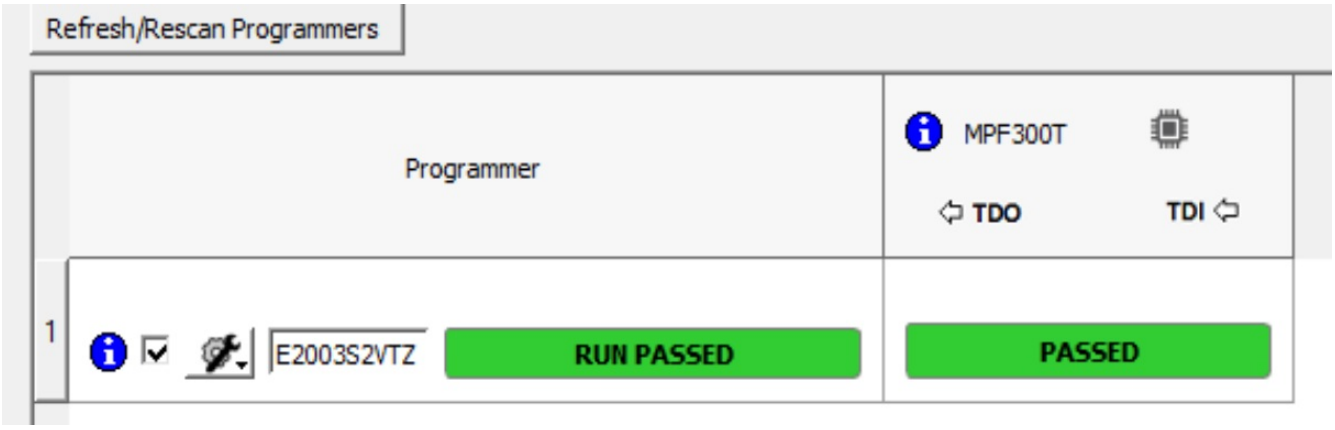
1. Connect the camera sensor to J5 on the PolarFire Video kit, which is a transmitter.
2. Connect the AD9371 to the J14 FMC connector on both PolarFire Video kits.
3. Connect the signal generator by setting it to 30.72 MHz and 1.8 Vpp to the J401 of the AD9371.
4. Connect one omnidirectional antenna to J305 (TX1 SMA connector) of the Transmitter AD9371 and the other antenna to J200 (RX1 SMA connector) of the Receiver AD9371.
5. Connect the HDMI cable to the J1 (HDMI 2.0 Tx port) of the PolarFire Video kit of the receiver.
6. Connect the host PC to the two PolarFire Video kits through J12 using a USB Mini cable.
7. Connect the power supply cable to J20 on two PolarFire Video kits.
8. Power up the HDMI monitor.
9. Power up the PolarFire Video kit using the SW4 slide switch.

Programming the Device (Ask a Question)

This section describes how to program the PolarFire FPGA Video kit with the .job file using FlashPro Express. To program the device, perform the following steps:
The mpf_an5454_v2024p2_df.job file is available at <download_folder\mpf_an5454_v2024p2_jb.

1. Connect a micro USB to J12 from the host PC and start the FlashPro Express software from its installation.
2. To create a new job project, select New or navigate to the Project menu and select New Job Project from FlashPro Express Job.
3. In the New Job Project, from the FlashPro Express Job dialog box, enter the following values:
    - Programming job file: Click Browse, navigate to the location where the .job file is located, and select the file. The default location is: <$download_folder>\mpf_an5454_v2024p2_df\Programming_Job.
    - FlashPro Express job project location: Click Browse and navigate to the location where you want to save the project.
4. Click OK. The required programming file is selected and ready to be programmed into the device.
5. The FlashPro Express window appears as shown in the following figure. Confirm that a programmer number (for example, 79DE6C53) appears in the Programmer field. If it does not, then confirm the board connections and click Refresh/Rescan Programmers. **Figure 4-2. Programming the Device**

6. Click RUN. When the device is programmed successfully, a RUN PASSED status is displayed asshown in the following figure. **Figure 4-3. FlashPro Express—RUN PASSED**



7. Close FlashPro Express or in the Project tab, click Exit.
8. Power cycle the board using SW5.

SoftConsole

The SoftConsole project, included with the Libero design, uses the APIs for AD9371 and AD9528 from Analog Devices, which are used to initialize and configure the clock generator AD9528. The RF transceiver AD9371, which has a built-in Arm® processor, runs calibration and other power-uppsequences required to initialize AD9371. The Arm binary byte file required for the built-in Arm processor is loaded by the Soft Mi-V processor in the FPGA power-up sequence, which indeed communicates with AD9371 and AD9528 through the Serial Peripheral Interface (SPI). The Mi-V processor reads the status of AD9528 and AD9371 through SPI during initialization and displays the status on the UART terminal.

\The following listed messages are displayed on the Receive Data window:

- SPI AD9371: Before initiating SPI communication between the Mi-V processor and the AD9371 (including AD9528), the process is visible on the screen following the proper initialization of the UART. If the message is not displayed or if random data is displayed on the GUI, then check the UART baudrate value in the GUI.
- Camera Initialised Successfully: The Mi-V processor utilizes the CoreI2C to configure the camera’s registers. Upon successful initialization of the camera, a confirmation message is displayed.
- PLL Locked: The Mi-V processor monitors the CLKPLLL bit and displays a message indicating whether the CLKPLL is locked. If the CLKPLL is not locked, a “PLL not locked” message is displayed.
- Multi-Chip Sync Successful: After performing Multi-Chip Sync, the Mi-V processor verifies if the Multi-Chip Sync was successful. If successful, a success message is displayed; otherwise, “Multi-Chip Sync Failed” is displayed.

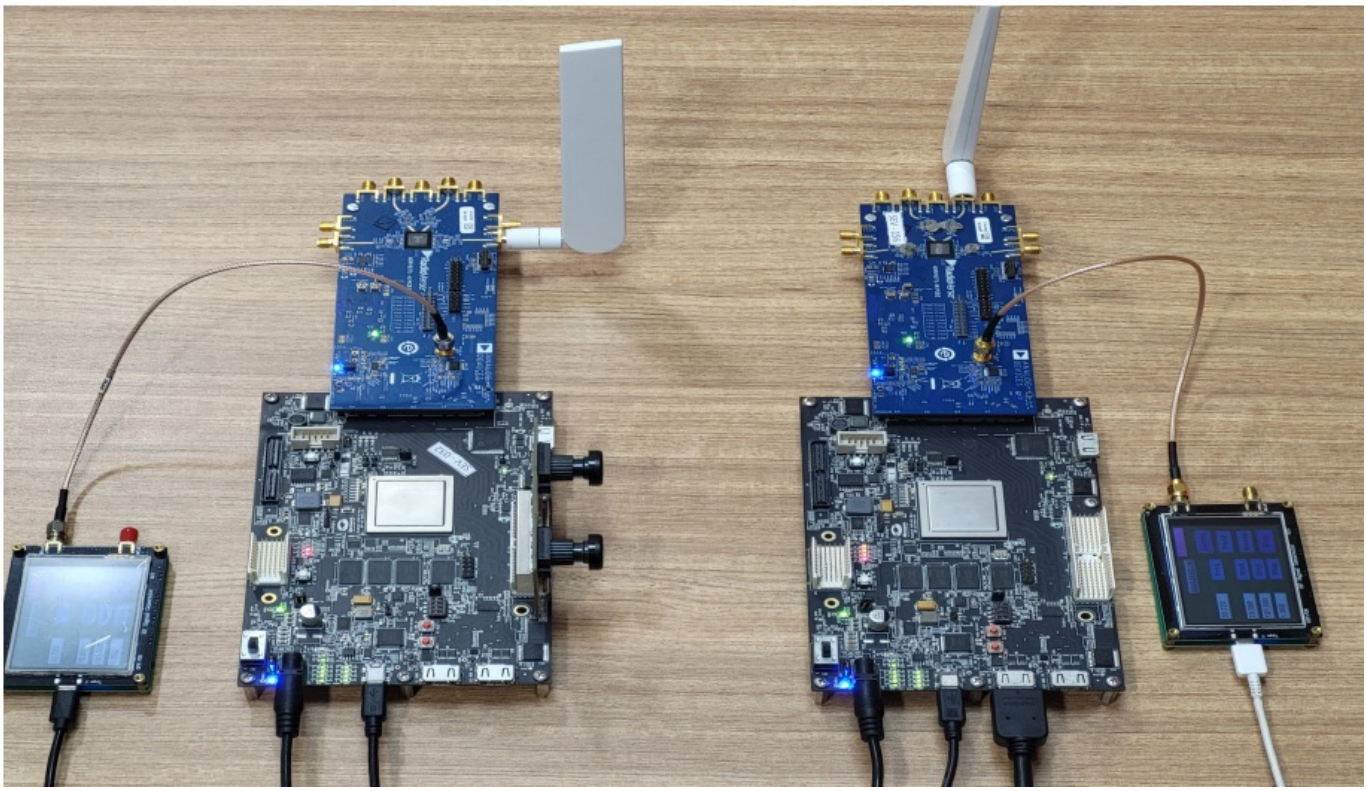
- AD9371 Arm Version 4.0.6: After loading an Arm binary byte file, the Mi-V processor reads the version of the loaded binary. This message is displayed if the binary byte array is loaded properly and the version is read back. If it is not loaded properly, an arbitrary number is displayed.
- (CLK, RX, TX, SNIFFER) PLLs Locked: The displayed message indicates that the Mi-V processor is performing a check on the PLLs for the CLK, RX, TX, and sniffer. If not locked, it displays the message “(CLK, RX, TX, SNIFFER) PLLs not locked” along with the PLL lock status. For more information about PLL status bits, see the UG992 AD9371 user guide for Analog Devices.
- Calibrations Completed Successfully: The Mi-V processor queries the calibration status of AD9371. This message is displayed if the Arm processor (in the AD9371) completes calibrations. Otherwise, this message is not displayed.
- Successfully Initialised AD9371: This final message is displayed when AD9371 is initialized and configured correctly. If AD9371 is not configured correctly or fails to initialize, the message “Initialization Failed, Status = XXXX” is displayed, along with the status bits (indicated as XXXX). The status bits are used to debug failures during initialization.

Running the Demo

To run the demo, perform the following steps:

1. After programming the PolarFire FPGA Video kit, connect AD9371 to the FMC connector.
2. Connect the SMA cable to the clock generator output and J401 on the AD9371. This acts as the reference clock for the FMC card.
3. Connect the antennas to the appropriate transmitter and receiver of the FMC card, depending on the type of demo being used.
    - For the demonstration on the two boards, connect one antenna to J305 (Tx1) of the first demo kit and connect the second antenna to J200 (Rx1) of the second demo kit, as shown in the following figure.

Figure 6-1. Video Kit Demo Setup using Two Boards



- For the demonstration on the same board, connect one antenna to the J305 (Tx1) and the second antenna to the J200 (Rx1) of the same board, as shown in the following figure.

Figure 6-2. Video Kit Demo Setup using the Same Board

Changing the Tx and Rx Frequency

The SDR demo runs with four different Local Oscillator (LO) frequencies. To modify the LO frequency, use the DIP switch SW6. Ensure to configure the same frequency on both the transmit and receive sides. In the two-board demonstration, the distance between the antennas is determined by the configured frequency. The following table lists the available LO frequencies.

Table 6-1. Frequency Settings

Switch [2:1]	LO Frequency
ON: ON	700 MHz
ON: OFF	1200 MHz
OFF: ON	2400 MHz
OFF: OFF	5800 MHz

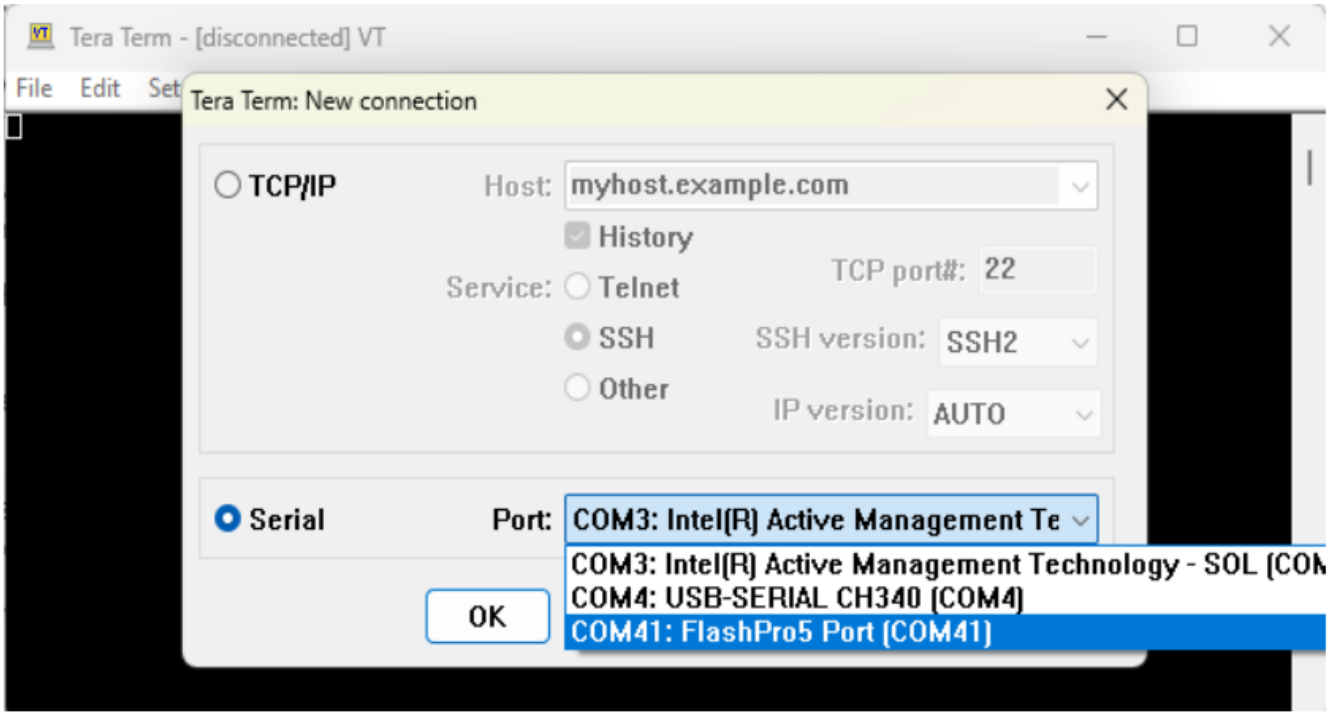
Important: EnsureA toggle SW2 after changing the LO Frequency.

Running the Demo on Two Boards (Ask a Question)

To run the demo on the two boards, perform the following steps:

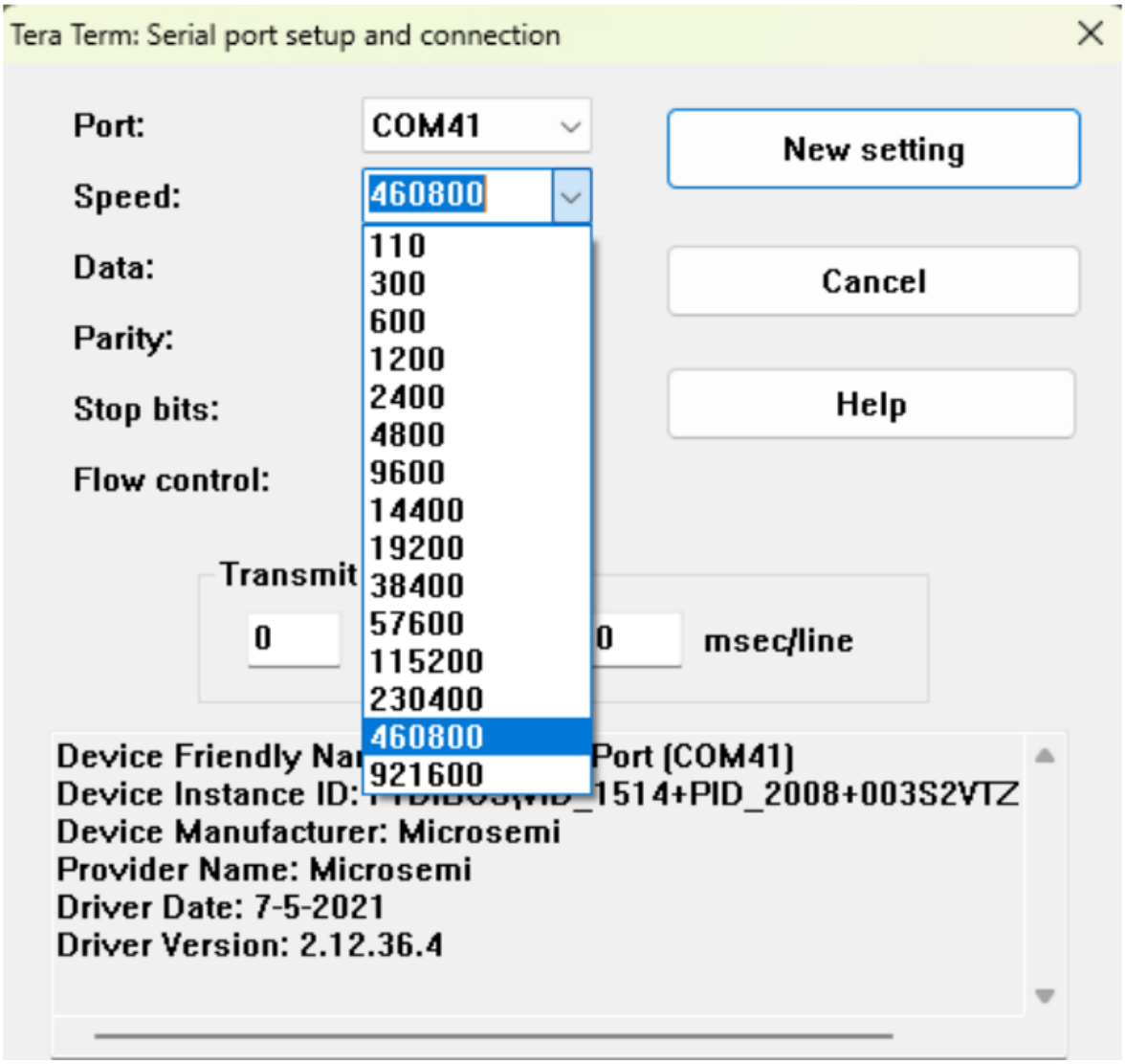
1. After programming the bit files, launch any port terminal application such as TeraTerm as shown in the following figure.

   Figure 6-3. TeraTerm Connection



2. Select the appropriate serial port (COM41 is the appropriate port for this demo setup, select accordingly for the user's setup).
3. Select the baud rate or speed as 460800 for the respective COM port, as shown in the following figure.

   Figure 6-4. Setting the Serial Port and Baud Rate



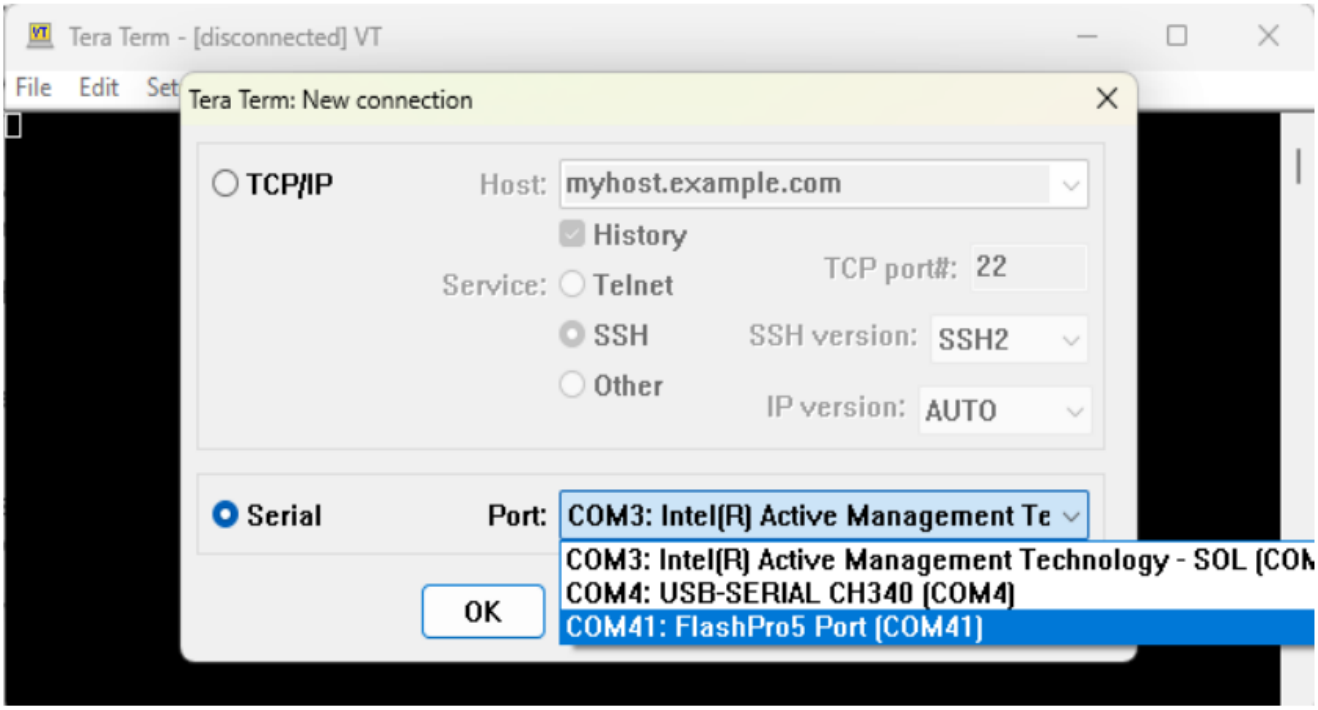
4. After completing the preceding steps, power cycle the board and wait for the following message “Successfully Initialized AD9371”.
5. Repeat the same steps for the second setup as well.
6. Once the receiver setup is ready, connect the HDMI monitor to the J1 to see the video output as shown in the following figure.

Figure 6-5. Video Output

Running the Demo on the Same Board (Ask a Question)

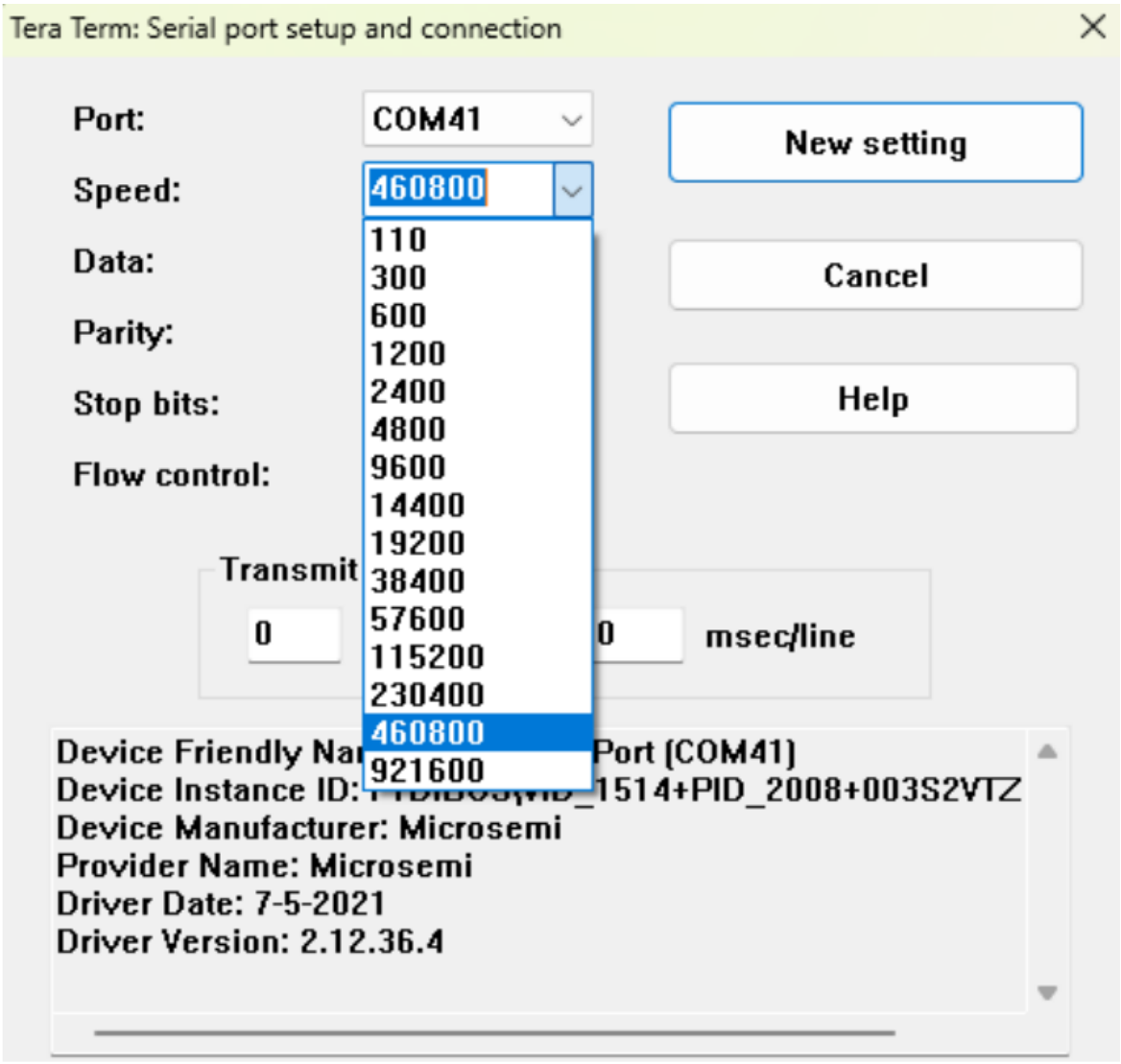
To run the demo on the same board, perform the following steps:

1. After programming the bit files, launch any port terminal application such as TeraTerm as shown in the following figure. **Figure 6-6. TeraTerm Connection**



2. Select the appropriate serial port (COM41 is the appropriate port for this demo setup, select accordingly for the user's setup).
3. Select the baud rate or speed as 460800 for the respective COM port, as shown in the following figure.

   Figure 6-7. Setting the Serial Port and Baud Rate



4. After completing the preceding steps, power cycle the board and wait for the following message “Successfully Initialised AD9371”.
5. Connect the HDMI monitor to the J1 of the PolarFire Video kit to display video on the monitor.

Revision History

The revision history describes the changes that were implemented in the document. The changes are listed by revision, starting with the most current publication.

Table 7-1. Revision History

Revision	Date	Description
B	12/2024	Updated the .job file path throughout the document.
A	07/2024	Initial release

Microchip FPGA Support

- Microchip FPGA products group backs its products with various support services, including Customer Service, Customer Technical Support Center, a website, and worldwide sales offices. Customers are suggested to visit Microchip online resources before contacting support as it is very likely that their queries have been already

answered.

- Contact the Technical Support Center through the website at **www.microchip.com/support**. Mention the FPGA Device Part number, select the appropriate case category, and upload design files while creating a technical support case.
- Contact Customer Service for non-technical product support, such as product pricing, product upgrades, update information, order status, and authorization.
- From North America, call 800.262.1060
- From the rest of the world, call 650.318.4460
- Fax, from anywhere in the world, 650.318.8044

Microchip Information

Trademarks

- **ISBN**: 979-8-3371-0304-4

Legal Notice

Microchip Devices Code Protection Feature

Note the following details of the code protection feature on Microchip products:

- Microchip products meet the specifications contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is secure when used in the intended manner, within operating specifications, and under normal conditions.
- Microchip values and aggressively protects its intellectual property rights. Attempts to breach the code protection features of Microchip products are strictly prohibited and may violate the Digital Millennium Copyright Act.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of its code. Code protection does not mean that we are guaranteeing the product is “unbreakable”. Code protection is constantly evolving. Microchip is committed to continuously improving the code protection features of our products.

Documents / Resources

	MICROCHIP AN5454 Video Streaming with SDR Using Polar Fire [pdf] Instructions AN5454 Video Streaming with SDR Using Polar Fire, AN5454, Video Streaming with SDR Using Polar Fire, SDR Using Polar Fire, Using Polar Fire, Polar Fire

References

- Design Help and Other Services | Microchip Technology
- Microchip Lightning Support
- microchipsupport.force.com/s/newcase?pub_guid=GUID-722E59E2-B7D5-4613-822C-380DA1DBB014&pub_lang=en-US&pub_ver=2&pub_type=Application%20Note&bu=fpga&tpc_guid=GUID-018B6827-45BE-4955-A01C-370DC7FF1501&cover_title=Video%20Streaming%20with%20SDR%20usi
- microchipsupport.force.com/s/newcase?pub_guid=GUID-722E59E2-B7D5-4613-822C-380DA1DBB014&pub_lang=en-US&pub_ver=2&pub_type=Application%20Note&bu=fpga&tpc_guid=GUID-0337C1E2-E951-46D9-B6FF-64F800993625&cover_title=Video%20Streaming%20with%20SDR%20usi
- microchipsupport.force.com/s/newcase?pub_guid=GUID-722E59E2-B7D5-4613-822C-380DA1DBB014&pub_lang=en-US&pub_ver=2&pub_type=Application%20Note&bu=fpga&tpc_guid=GUID-1876A696-B525-4347-BC6C-19EB759985F3&cover_title=Video%20Streaming%20with%20SDR%20usi
- microchipsupport.force.com/s/newcase?pub_guid=GUID-722E59E2-B7D5-4613-822C-380DA1DBB014&pub_lang=en-US&pub_ver=2&pub_type=Application%20Note&bu=fpga&tpc_guid=GUID-245225F8-E43F-41C3-B0C3-8A524858AFFB&cover_title=Video%20Streaming%20with%20SDR%20usi
- microchipsupport.force.com/s/newcase?pub_guid=GUID-722E59E2-B7D5-4613-822C-380DA1DBB014&pub_lang=en-US&pub_ver=2&pub_type=Application%20Note&bu=fpga&tpc_guid=GUID-3D6B2EC3-DB21-4A1D-BEBA-C0F9B4B5F73F&cover_title=Video%20Streaming%20with%20SDR%20usi
- microchipsupport.force.com/s/newcase?pub_guid=GUID-722E59E2-B7D5-4613-822C-

380DA1DBB014&pub_lang=en-US&pub_ver=2&pub_type=Application%20Note&bu=fpga&tpc_guid=GUID-483CCF91-4D52-4DE4-96DC-4EC7BDC5244C&cover_title=Video%20Streaming%20with%20SDR%20usi

- microchipsupport.force.com/s/newcase?pub_guid=GUID-722E59E2-B7D5-4613-822C-380DA1DBB014&pub_lang=en-US&pub_ver=2&pub_type=Application%20Note&bu=fpga&tpc_guid=GUID-4BB877B0-4741-4E40-B797-022E30D7F83B&cover_title=Video%20Streaming%20with%20SDR%20usi
- microchipsupport.force.com/s/newcase?pub_guid=GUID-722E59E2-B7D5-4613-822C-380DA1DBB014&pub_lang=en-US&pub_ver=2&pub_type=Application%20Note&bu=fpga&tpc_guid=GUID-647261CA-D91C-47BD-AE84-D501F4F2EB65&cover_title=Video%20Streaming%20with%20SDR%20usi
- microchipsupport.force.com/s/newcase?pub_guid=GUID-722E59E2-B7D5-4613-822C-380DA1DBB014&pub_lang=en-US&pub_ver=2&pub_type=Application%20Note&bu=fpga&tpc_guid=GUID-65F153F8-353A-4D57-AE4D-BD6928B218B5&cover_title=Video%20Streaming%20with%20SDR%20usi
- microchipsupport.force.com/s/newcase?pub_guid=GUID-722E59E2-B7D5-4613-822C-380DA1DBB014&pub_lang=en-US&pub_ver=2&pub_type=Application%20Note&bu=fpga&tpc_guid=GUID-70F1D20C-2BE7-4B86-A1D2-975A478D0CEC&cover_title=Video%20Streaming%20with%20SDR%20usi
- microchipsupport.force.com/s/newcase?pub_guid=GUID-722E59E2-B7D5-4613-822C-380DA1DBB014&pub_lang=en-US&pub_ver=2&pub_type=Application%20Note&bu=fpga&tpc_guid=GUID-727ED9A6-9CDB-45A7-973D-DA27C8A614B8&cover_title=Video%20Streaming%20with%20SDR%20usi
- microchipsupport.force.com/s/newcase?pub_guid=GUID-722E59E2-B7D5-4613-822C-380DA1DBB014&pub_lang=en-US&pub_ver=2&pub_type=Application%20Note&bu=fpga&tpc_guid=GUID-79666B68-AA7D-47D3-8A90-F1A120A4224C&cover_title=Video%20Streaming%20with%20SDR%20usi
- microchipsupport.force.com/s/newcase?pub_guid=GUID-722E59E2-B7D5-4613-822C-380DA1DBB014&pub_lang=en-US&pub_ver=2&pub_type=Application%20Note&bu=fpga&tpc_guid=GUID-867F107A-D258-461D-8F13-F68BAF93DEF3&cover_title=Video%20Streaming%20with%20SDR%20usi
- microchipsupport.force.com/s/newcase?pub_guid=GUID-722E59E2-B7D5-4613-822C-380DA1DBB014&pub_lang=en-US&pub_ver=2&pub_type=Application%20Note&bu=fpga&tpc_guid=GUID-8A3FC5B4-9D27-4F06-9251-EFAC09B47FB4&cover_title=Video%20Streaming%20with%20SDR%20usi
- microchipsupport.force.com/s/newcase?pub_guid=GUID-722E59E2-B7D5-4613-822C-380DA1DBB014&pub_lang=en-US&pub_ver=2&pub_type=Application%20Note&bu=fpga&tpc_guid=GUID-90BA9D64-B648-49B9-BFAA-42FB87A9DFCE&cover_title=Video%20Streaming%20with%20SDR%20usi
- microchipsupport.force.com/s/newcase?pub_guid=GUID-722E59E2-B7D5-4613-822C-380DA1DBB014&pub_lang=en-US&pub_ver=2&pub_type=Application%20Note&bu=fpga&tpc_guid=GUID-C4EAE990-FFD8-4045-990D-4AD15CD31698&cover_title=Video%20Streaming%20with%20SDR%20usi
- microchipsupport.force.com/s/newcase?pub_guid=GUID-722E59E2-B7D5-4613-822C-