AN4529

PolarFire SoC FPGA H.264 Video Streaming Over Ethernet Application Note

Introduction (Ask a Question)

This application note describes how to run H.264 video streaming over the Ethernet demo. This solution is developed on PolarFire® SoC FPGA video kit, which features an MPFS250T PolarFire SoC device. Microchip's PolarFire SoC devices combine RISC-V based 5x core Microprocessor Subsystem (MSS) capable of running Linux® and the PolarFire Fabric in a single device. This combination enables the partitioning of user designs between the MSS (C Source code) and Fabric (RTL). Microchip's Libero® SoC enables the rapid development of RTL based designs for PolarFire SoC and many other device families. Libero SoC provides a wide range of IPs for a variety of applications such as video and imaging, signal processing, wired and wireless communications, and networking. Microchip's SoftConsole enables the rapid development of the C/C++ source code-based applications targeted for all Microchip FPGA and SoC device families.

The demo design captures live stream from a camera on the PolarFire SoC video kit and performs H.264 compression in the fabric logic. Webserver application running on MSS allows user to connect to the PolarFire SoC video kit through Ethernet using the IP address of the kit from the web browser. The web page on the browser allows user to control streaming of the live video from PolarFire SoC video kit to the connected system. On initiating the streaming from the web page, application triggers MSS that reads compressed stream of data from fabric and sends encoded H.264 RTP Ethernet packets with the IP address of the system from which the stream request was initiated. User can play the video on computer using VLC Player, Gstreamer, or FFPlay applications. Web page allows user to download and pass the SDP file to the video player application after streaming is initiated.

PolarFire SoC FPGA video kit enables prototyping of Video and Imaging solutions. The kit supports the following key features among others:

- MPFS250T PolarFire SoC FPGA device
- MIPI CSI-2 interface
- FPGA Mezzanine Card (FMC) connector
- SD and eMMC card, LPDDR4, and DDR4 memories
- HDMI, Ethernet, PCIe®, and other interfaces
- CAN, WIFI, and mikroBUS

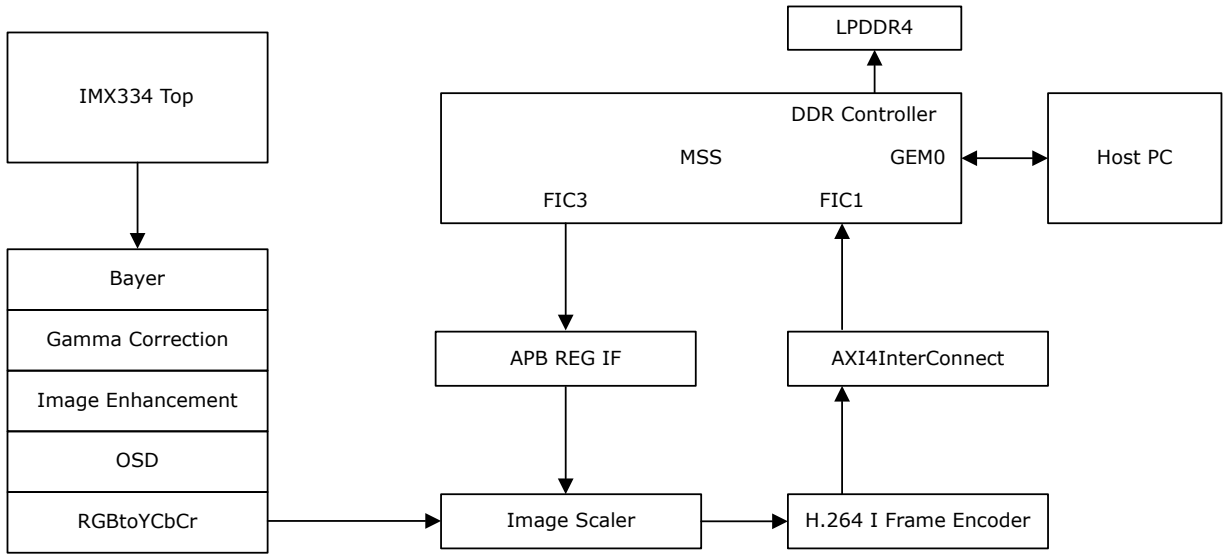
For more information about the video kit, see MPFS250-VIDEO-KIT.

Table of Contents

1. Design Overview (Ask a Question)

The following figure shows the high-level block diagram of the reference design.

Figure 1-1. High-level Block Diagram



The camera generates a live Full HD image (1920x1080 picture resolution) in the Bayer format at 30 FPS. The FPGA fabric receives these frames and performs processing as follows:

- IMX334 top block contains MIPI CSI2 Rx IP to get the camera data, which is passed to the Bayer Interpolation IP. This IP block converts each Bayer frame to RGB data format. Specifically, each single 8-bit Bayer pixel data is converted to a 24-bit RGB pixel.
- The 24-bit RGB pixel data is then passed through Image Processing IPs to add features like auto-brightness, R/G/B gains, contrast, and so on.
- The On Screen Display (OSD) logic injects the text "COMPRESSION RATIO x" into the image frame. Where the x is the value of compression ratio input from software. Each character size is 16x16 pixels. The text characters and digits in 16x16 pixel format are stored in the ROM. The OSD logic injects the text at the location specified by the coordinates input from software. The text color can also be controlled by using the color input of OSD logic. However, software is yet to support the color interface. The OSD logic takes RGB as the input frame format and gives RGB frame with the injected text in the specified coordinates as the output.
- RGB data is converted to YUV422 format using RGB to YUV convert IP.
- 1920x1080 image is scaled down to the selected resolution using Image Scaler IP and sent to H.264 Compression IP. The resolution can be changed from the user application. The APB registers are accessed from FIC3 Master Interface.
- The H.264 compressed data is written to the LPDDR memory through FIC1 AXI slave interface.
- Application running on MSS reads the compressed data from the LPDDR4 memory.
- On Linux, Fast Forward MPEG (FFMPEG) reads FIFO data and generates RTP packets to send over GEM.

RGB Gains, Compression quality factor, Compression Resolution, Contrast, Brightness, and so on are controlled by MSS through FIC3 Master interface.

1.1 IP Blocks (Ask a Question)

The following table lists the IP blocks used in the design and their functionality.

Table 1-1. IP Blocks

IP Block	Description
PF_OSC	The PolarFire® RC Oscillator block generates a 2 MHz clock.
PF_CCC	The PolarFire CCC block generates multiple clocks in the design.
Clock divider	The clock divider block divides by two configurations.
H.264	The H.264 block performs H.264 compression on YUV4:2:2 data input.
mipicsi2rxdecoder	The MIPI CSI-2 receiver decoder block for PolarFire (MIPI CSI-2 RxDecoder) decodes the data from the sensor interface. g_DATATYPE: RAW10 g_LANE_WIDTH: 4 g_NUM_OF_PIXELS: 1 g_INPUT_DATA_INVERT: 0 g_FIFO_SIZE: 12
PF_IOD_Generic_RX	The IP block operates with 500 Mbps data rate.
CoreReset	The IP block synchronizes the reset to the respective clock domain.
MSS_VIDEO_KIT	On the MSS hard IP, the Linux user space application receives the H.264 compressed frame from fabric FIFO and streams over Ethernet. For more information about the MSS configuration and LPRDDR4 memory partitioning, see 1.2. MSS Configuration.
Core AXI4 Interconnect	Single Master with 64-bit data width and Single Slave with 64-bit data width. The Slave is configured with 0xE000_0000 start and 0xE003_FFFF end addresses. Here, the AXI4 master is reading the video data from AXI4 Slave FIFO.
Bayer	This IP block converts the 8-bit raw data to 24-bit RGB data and forwards it to the Gamma Correction IP.
RGBtoYCbCr_C0	This color space conversion IP block converts OSD RGB 24-bit data format to YUV422 16-bit data format.
Gamma_Correction_C0	This IP block converts the pixel intensity to match with the perspective of human eye by using a logarithmic curve.
Image_Enhancement_C0	This IP block adjusts the brightness, contrast, and color balance through user controls.
IMX334_IF_TOP	This SmartDesign module receives the live camera feed and converts it into raw 8-bit parallel data. Each byte represents one pixel.
INIT MONITOR	The IP block triggers reset to the design.

1.2 MSS Configuration (Ask a Question)

The following table lists the configuration of MSS clock, peripherals, and memory.

Table 1-2. MSS Blocks

MSS Block	Description
MSS Clocks	MSS PLL reference clock source: 125 MHz MSS CPU clock frequency: 600 MHz
Peripherals	• eMMC: Used for storing Linux kernel images. • Gigabit Ethernet MAC: Used for copying data to/from the Linux OS to Host PC. • MMUART: Used to capture HSS boot log and Linux terminal. • I^2C: Used for initializing the camera. • Fabric Interface Controller (FIC1) AXI4 Slave interface for LPDDR access from fabric. • Fabric Interface Controller (FIC3) APB Master interface for register access.
DDR Memory	LPDDR4 with the default configuration.
Memory Partitioning	See 1.2.1.1. DDR Memory Partition.

1.2.1 DDR Memory Mappings (Ask a Question)

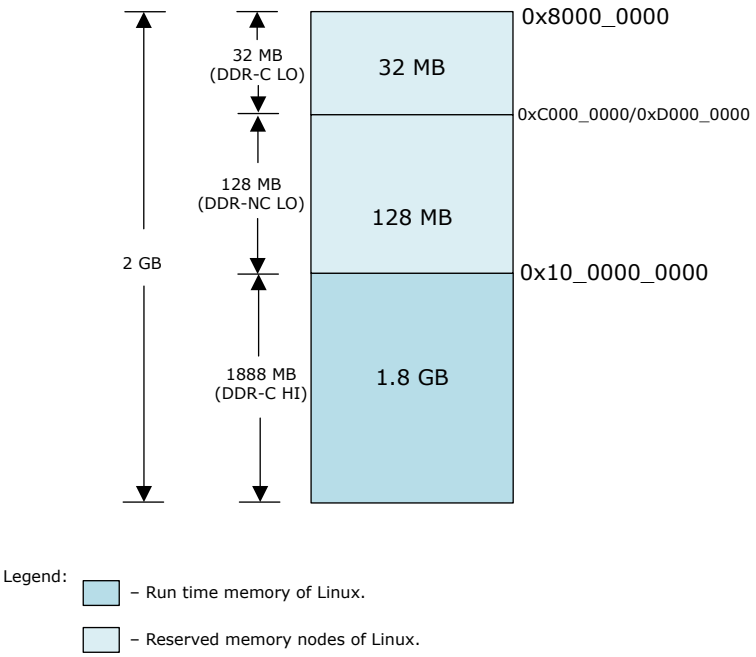
This section describes the DDR regions and their allocations. It also explains how these memory regions are defined in the Linux device tree so that the Linux user space can address these regions.

For more information about DDR memory partition regions, see the “Segmentation Blocks” section in PolarFire SoC MSS Technical Reference Manual.

1.2.1.1 DDR Memory Partition (Ask a Question)

The following figure shows the memory mapping of the 2 GB on-board DDR memory in the Linux device tree.

Figure 1-2. LPDDR4 Memory Mapping



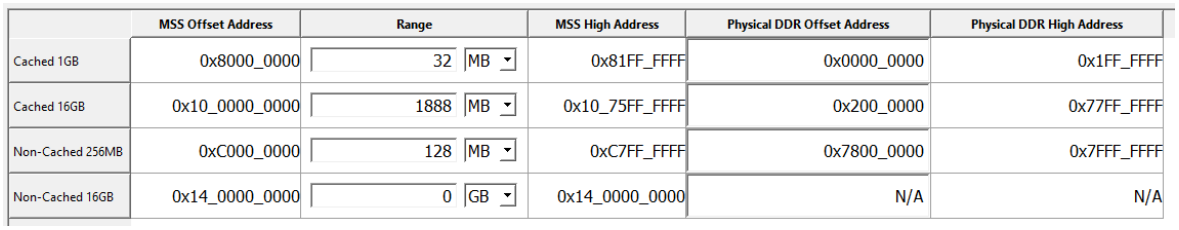
PolarFire SoC video kits come with 2 GB of DDR memory by default, partitioned as follows:

- 1.92 GB (1.88 GB + 32 MB) is allocated to higher cached memory.
  - 1.88 GB at higher address space.
  - 32 MB at lower address space.

- 128 MB is allocated to non-cached memory. This region is addressable with or without Write Combining Buffer WCB.

The following figure shows the **MSS Configurator** > **DDR Memory Partition** settings used in the design.

Figure 1-3. DDR Memory Partition



	MSS Offset Address	Range	MSS High Address	Physical DDR Offset Address	Physical DDR High Address
Cached 1GB	0x8000_0000	32 MB	0x81FF_FFFF	0x0000_0000	0x1FF_FFFF
Cached 16GB	0x10_0000_0000	1888 MB	0x10_75FF_FFFF	0x200_0000	0x77FF_FFFF
Non-Cached 256MB	0xC000_0000	128 MB	0xC7FF_FFFF	0x7800_0000	0x7FFF_FFFF
Non-Cached 16GB	0x14_0000_0000	0 GB	0x14_0000_0000	N/A	N/A

1.2.1.2 Linux Device Tree Memory Mapping (Ask a Question)

The following table lists the memory regions defined in the Linux device tree, as per the PolarFire SoC video kit. For more information about the PolarFire SoC memory mapping, see the “MSS Memory Map” section in PolarFire SoC MSS Technical Reference Manual.

Table 1-3. DDR Memory Regions and Sizes for MPFS250-VIDEO-KIT

Region Name	Region Base	Region Size	Region Properties
DDRC-LO	0x80000000	32 MB	32-bit address, cached
DDR-NC-LO	0xC0000000	128 MB	32-bit address, non-cached
DDR-NC-WCB-LO	0xD0000000	128 MB	32-bit address, non-cached, write-combine buffer
DDRC-HI	0x1000000000	1.88 GB	64-bit address, cached

Important: The 128 MB is a shared view of a single physical memory area.

The memory regions listed in the preceding table are mapped in the Linux device tree as follows.

- The 1.88 GB DDR-C is mapped at 0x10_0000_0000. The following `.dts` stanzas define these mappings.

```
ddrc_cache: memory@1000000000 {
  device_type = "memory";
  reg = <0x10 0x0 0x0 0x76000000>;
  status = "okay";
        };
```

- The remaining 160 MB of memory for other purposes is allocated into the following three buffers of reserved memory in the 32-bit address space (LO):
  - 32 MB buffer, cache-coherent
  - 64 MB buffer, non-cache-coherent
  - 64 MB buffer, non-cache-coherent, write combine buffered

These three buffers are defined in the device tree (`.dts`) file in the following reserved-memory stanzas.

```
 reserved-memory {
 ranges;
 #size-cells = <2>;
 #address-cells = <2>;
 fabricbuf0ddrc: buffer@80000000 {
   compatible = "shared-dma-pool";
```

```
    reg = <0x0 0x80000000 0x0 0x2000000>;
    no-map;
   };
    fabricbuf1ddrnc: buffer@c4000000 {
    compatible = "shared-dma-pool";
    reg = <0x0 0xc4000000 0x0 0x4000000>;
    no-map;
   };
  fabricbuf2ddrncwcb: buffer@d4000000 {
    compatible = "shared-dma-pool";
    reg = <0x0 0xd4000000 0x0 0x4000000>;
    no-map;
   };
  };
```

Important: The reserved-memory stanza must be at the top-level of the device tree file (`.dts`) for the Linux subsystem to pick.

To access this reserved memory from the Linux user space, device driver named `u-dma-buf` by Ichiro Kawazome is used. It allocates contiguous memory blocks in the kernel space as DMA buffers and makes them available from the user space.

1.3 I/O Ports (Ask a Question)

The following table lists the key I/O ports in the design.

Port Name	Direction	Description
Camera RX, Reset, and Reference Clock Ports		
CAM1_RST	Input	Input signal used to reset the camera sensor module. This signal comes from the MSS.
CAM1_RX_CLK_P CAM1_RX_CLK_N	Input	Input pads to receive the reference lock for the camera sensor.
CAM1_RXD[0] CAM1_RXD_N[0] CAM1_RXD[1] CAM1_RXD_N[1] CAM1_RXD[2] CAM1_RXD_N[2] CAM1_RXD[3] CAM1_RXD_N[3]	Input	Input pads to receive the live video from the camera sensor module. These pads are assigned to the Bank 7 I/Os.
MSS Peripheral Ports		
REFCLK REFCLK_N	Input	Input ports for receiving MSS reference clock from the on-board 125 MHz oscillator.
RESET_N	Output	This port is assigned to AB12 pin of Bank 6 MSS DDR. This port is used to reset the on-board LPDDR4.

.....continued

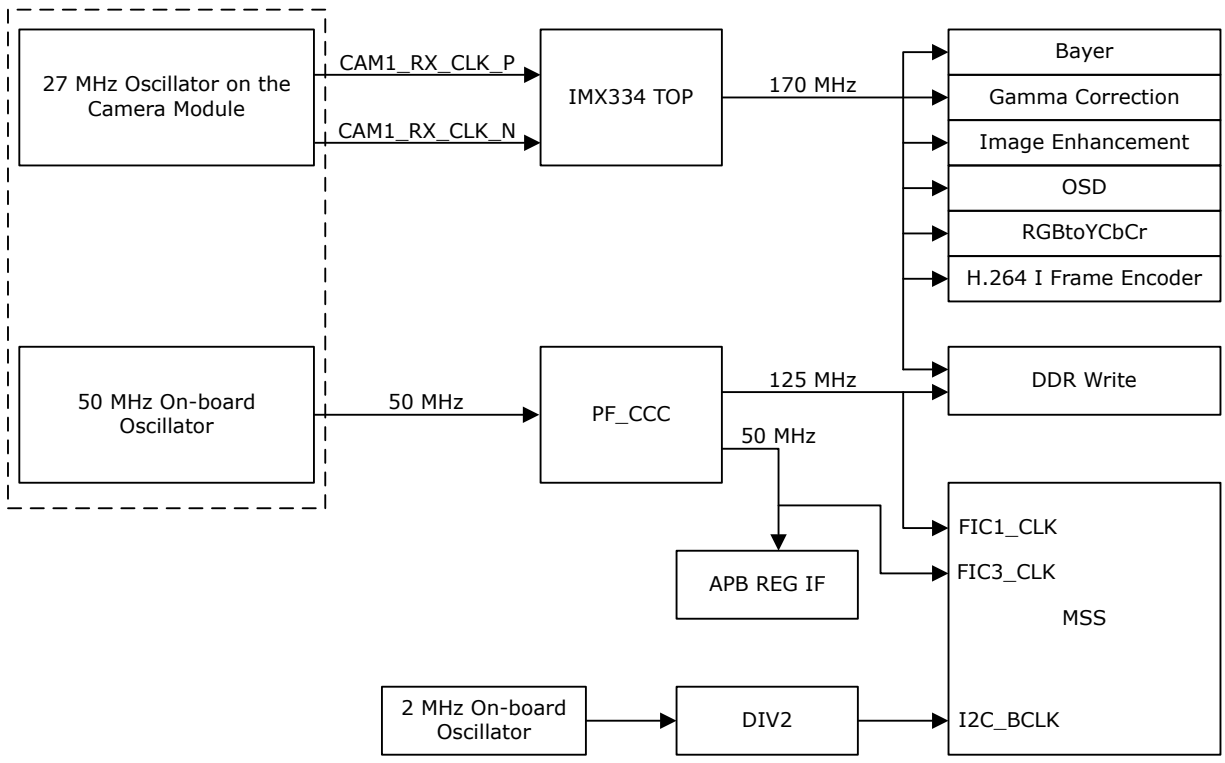
Port Name	Direction	Description
MMUART_0_RXD_F2M MMUART_0_TXD_M2F MMUART_1_RXD_F2M MMUART_1_TXD_M2F	Input and Output	Input and Output ports assigned to C7, B7, D4, and C4 pins. These pins are connected to the on-board CP2108 USB to UART chip.
USB_CLK	Input	USB clock from the on-board USB3340-EZK-TR IC.
USB_DATA0 USB_DATA1 USB_DATA2 USB_DATA3 USB_DATA4 USB_DATA5 USB_DATA6 USB_DATA7	Inout	USB data ports assigned to Bank 2 MSS I/Os.
EMMC_CLK	Output	This port is assigned to AA8 pin of the Bank 4 to provide clock to the eMMC device.
EMMC_CMD	Inout	This port is assigned to AA9 pin of Bank 4 to receive and forward eMMC commands.
EMMC_DATA0 EMMC_DATA1 EMMC_DATA2 EMMC_DATA3 EMMC_DATA4 EMMC_DATA5 EMMC_DATA6 EMMC_DATA7	Inout	eMMC data ports assigned to Bank 4 MSS I/Os.
SDIO_SW_EN_N	Output	Enable Secure Digital Input Output (SDIO) port.
SDIO_SW_SEL0 SDIO_SW_SEL1	Output	SDIO selection pin.
SGMII_RX0_N SGMII_RX0_P	Input	Input ports assigned to Bank 5 W1 and Y1 pins (MSS I/Os). These pins are connected to the on-board VSC8662 PHY device.
SGMII_TX0_N SGMII_TX0_P	Output	Output ports assigned to Bank 5 Y5 and AA5 pins (MSS I/Os). These pins are connected to the on-board VSC8662 PHY device.
Transceiver Ports		
REF_CLK_PAD_N_0 REF_CLK_PAD_P_0	Input	Inputs ports for receiving the transceiver reference clock. These ports are assigned to AF29 and AF30 pins, which are connected to the on-board 148.5 MHz oscillator.

.....continued		
Port Name	**Direction**	**Description**
LANE0_TXD_N LANE0_TXD_P LANE1_TXD_N LANE1_TXD_P LANE2_TXD_N LANE2_TXD_P LANE3_TXD_N LANE3_TXD_P	Output	Transceiver TX lanes connected to HDMI TX interface.
LED2 LED3	Output	GPIOs connected to on-board user LEDs.

1.4 Clocking Structure (Ask a Question)

The following figure shows the clocking structure of the design.

Figure 1-4. Clocking Structure

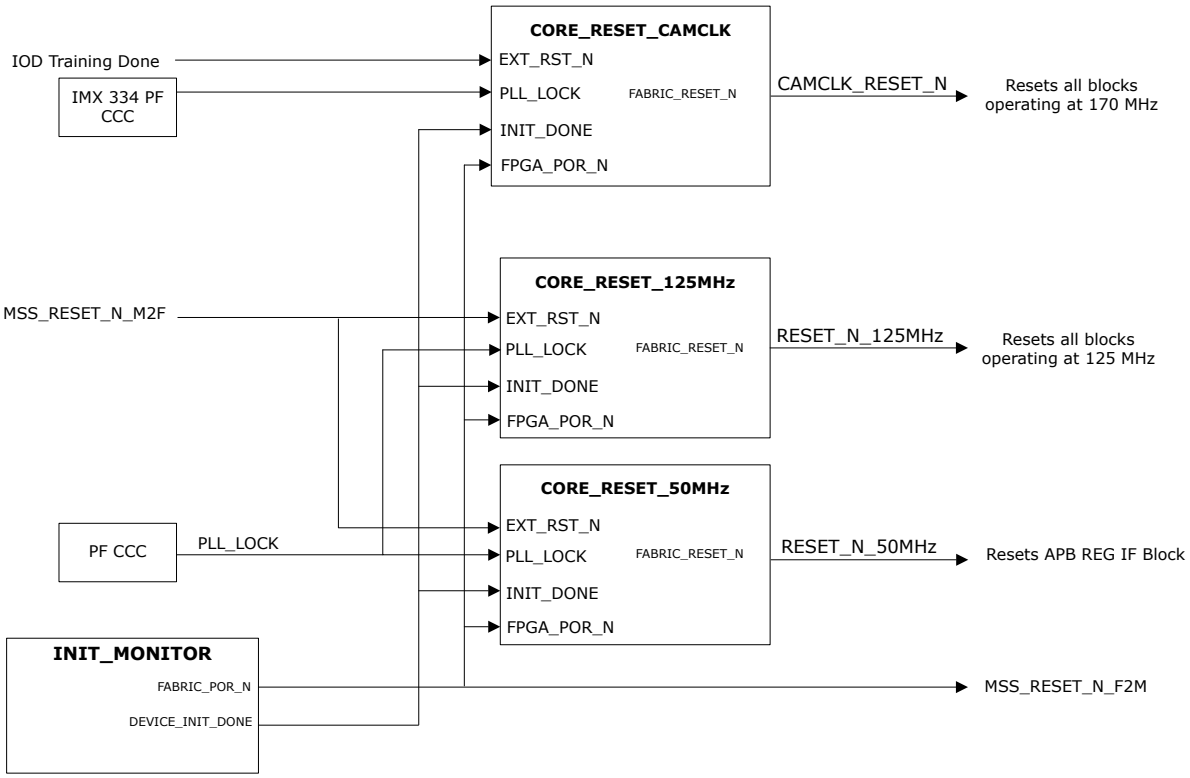


- PF_CCC generates the following fabric clocks from the 50 MHz on-board reference clock.
  - OUT0_FABCLK_0 (125 MHz): This clock is used to drive the FIC1 and AXI IF clocks.
  - OUT1_FABCLK_0 (50 MHz): This clock is used to drive the FIC3 and APB3 bus interface.
- The on-board 27 MHz Oscillator on the camera module is used as a reference clock for IOD in the IMX334 block. The CCC inside the IMX334 uses reference clock from IOD and generates 170 MHz clock.

1.5 Reset Structure (Ask a Question)

The following figure shows the reset structure of the design.

Figure 1-5. Reset Structure



The INIT_MONITOR IP asserts the following signals:

- FABRIC_POR_N: This signal is asserted after the initialization of the fabric. FABRIC_POR_N is used to reset the MSS.
- DEVICE_INIT_DONE: This signal is asserted after the initialization of the entire PolarFire SoC device.

After asserting the FABRIC_POR_N, DEVICE_INIT_DONE, and MSS_RESET_N_M2F, the CORERESET_PF IP blocks generate the Reset signals.

1.6 Resource Utilization (Ask a Question)

The following figure shows the resource utilization of the design.

Figure 1-6. Resource Utilization

Module Name	Fabric 4LUT	Fabric DFF	Interface 4LUT	Interface DFF	Single-Ended I/O	Differential I/O Pairs	uSRAM 1K	LSRAM 18K	Math (18x18)	Chip Globals	PLL
Top	27381	23621	6360	6360	98	15	23	122	47	8	2
Primitives	1	0	0	0	20	0	0	0	0	0	0
CLOCKS_AND_RESETS_inst...	2	32	0	0	0	0	0	0	0	3	1
FIC_CONVERTER_0	89	0	0	0	0	0	0	0	0	0	0
MSS	2	0	0	0	78	10	0	0	0	0	0
Video_Pipeline_0	27287	23589	6360	6360	0	5	23	122	47	5	1
IMX334_IF_TOP_0	5649	5039	396	396	0	5	0	11	0	4	1
RGBtoYCbCr_C0_0	94	69	324	324	0	0	0	0	9	0	0
apb3_if_0	31	220	0	0	0	0	0	0	0	0	0
h264_top_0	18570	17011	4524	4524	0	0	23	96	22	1	0
Primitives	1	0	0	0	0	0	0	0	0	1	0
DDR_AXI4_ARBITE...	418	445	84	84	0	0	1	2	0	0	0
H264_DDR_WRITE_64	294	346	468	468	0	0	0	13	0	0	0
H264_Iframe_Enco...	17857	16220	3972	3972	0	0	22	81	22	0	0
video_processing_0	2943	1250	1116	1116	0	0	0	15	16	0	0

2. Demo Requirements (Ask a Question)

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

Table 2-1. Demo Requirements

Requirement	Description
Hardware and Accessories	
PolarFire® SoC video kit	MPFS250-VIDEO-KIT
Image Sensor Module	LI-IMX334-MIPI-MICRO v1.0
USB A to micro-B cable	Required for: • FPGA programming • UART interface for the terminal prompt from Linux®
Power adapter	12V, 5A
Host PC	• A host PC with USB and Ethernet port • Windows® 10 and above, or Ubuntu 20.04
Ethernet cable	To communicate with the video kit while using the web GUI.
Utility Software	
FPExpress	FPGA Programming application
MobaxTerm, Putty, or TeraTerm	UART receiver-transmitter application at the host PC.
USBImager	Free utility used for flashing binary images to USB sticks or SD/CF cards.
7-zip	Free archiving utility for compressing/decompressing files and is needed for extracting `.wic` file from `.wic.gz`.
Admin privileges	Flashing Linux `.wic` images using USBImager requires admin privileges.
VLC Media Player	• 3.0.16 (Windows) and above or 3.0.9.2 (Ubuntu) and above • To play the H.264 video
Web Browser	Chrome 96.0.4664.110 (Official Build) (64-bit) and above, or Microsoft Edge Version 96.0.1054.53 (Official build) (64-bit) and above
Flashable Binaries	
Linux `wic` image	`core-image-minimal-dev-mpfs-video-kit-xxxx.rootfs.wic.gz`
Job file	`VKPFSOC_H264.job`

The name of the binaries mentioned in the preceding table may be different from the rest of the guide as these keep changing in each releases but the flashing and demo procedures remain the same.

3. Demo Prerequisites (Ask a Question)

Before you start:

1. Download the Programming Job File.
2. Download the Linux `.wic` image.

4. Setting Up the Demo (Ask a Question)

The demonstration involves the following steps:

- 4.1. Setting Up the Hardware
- 4.2. Setting Up the Serial Terminal
- 4.3. Programming the Device

4.1 Setting Up the Hardware (Ask a Question)

Setting up the hardware involves interfacing the camera sensor module and the HDMI monitor with the PolarFire SoC video kit.

Follow these steps:

1. Connect the camera sensor module at **J10** on the video kit.
2. Ensure the jumper settings on the video kit are set as per the “Jumper Settings” section in PolarFire SoC FPGA Video Kit User Guide.

4.2 Setting Up the Serial Terminal (Ask a Question)

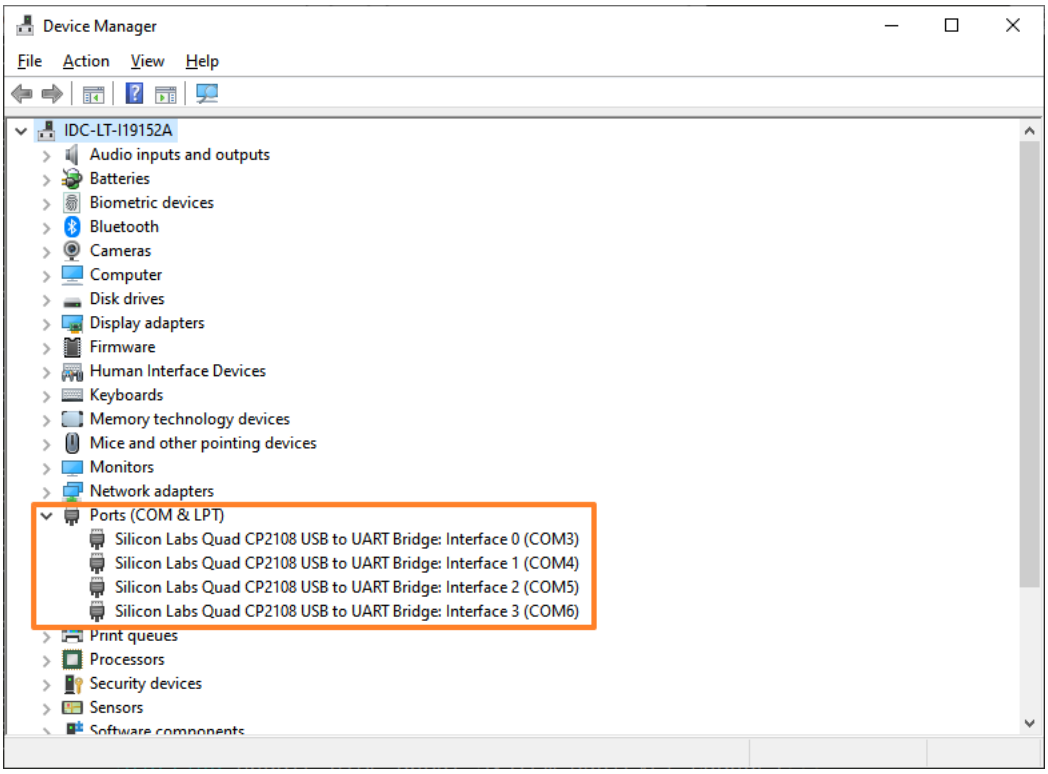
PolarFire SoC Video kit includes CP2108 USB to UART chip to interact with on-board Linux OS. Before you start:

- Download and install CP210x Universal Windows Driver.
- Unzip and see the *CP210x_Universal_Windows_Driver_ReleaseNotes*.

After installing the driver, follow these steps:

1. Connect USB cable at **J12** port on the PolarFire SoC video kit board to the host PC.
2. After connecting the power adapter to the board at **J39**, switch ON the boards power supply using the SW5 switch. This must detect the USB UART chip on the board at the host PC. You can confirm this at the host PC device manager, see the following figure.

Figure 4-1. Device Manager



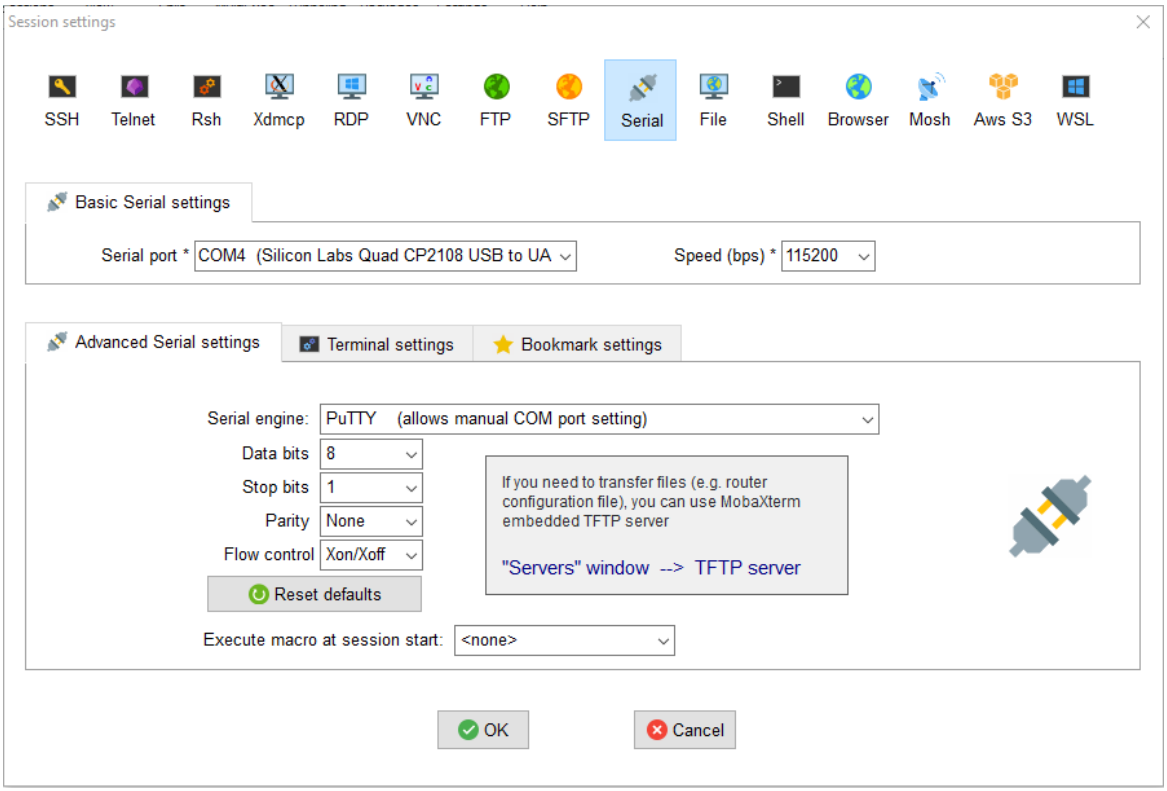
The interfaces on the host PC's device manager are active in the current demo project and display the messages when appropriately configured.

Interface 0 displays the Hart Software Service (HSS) boot messages while Interface 1 displays U-Boot messages, Linux boot messages, and provides a Linux prompt.

Important: For each user, the interfaces might be on a different COM# than what is shown in the preceding figure.

An application like MobaXterm/TeraTerm at the host PC is required to establish serial communication with the video kit board. The baud rate for such connections must be 115200 bps, see the following figure.

Figure 4-2. Serial Port Settings



3. Open Interface 0 and Interface 1.

Important: The data rate must be configured for all interfaces to establish successful communication with the PolarFire SoC video kit.

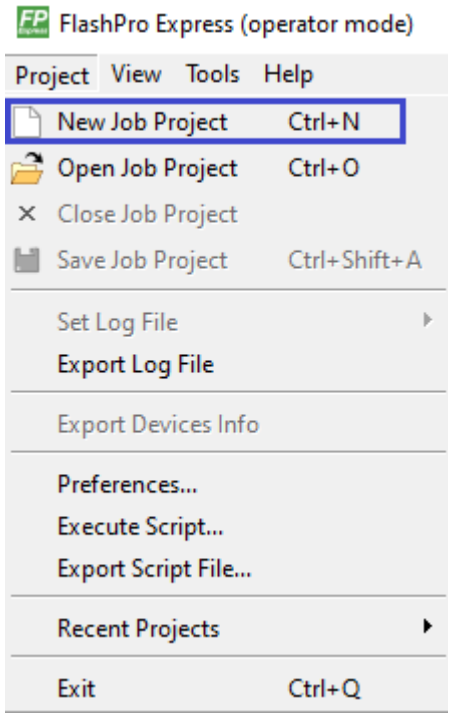
4.3 Programming the Device (Ask a Question)

This section describes the steps to program the PolarFire SoC device with the `VKPFSoC_H264.job` file using FlashPro Express.

Follow these steps:

1. Ensure to close pins 1 and 2 of J28 jumper.
2. Connect a micro-USB to **J5** from the host PC and start the FPExpress software from **Windows** > **Start**.
3. Ensure to power on the PolarFire SoC video kit.
4. Select **New** or **New Job Project** from the **Project** menu to create a new job project, see the following figure.

Figure 4-3. New FlashPro Express Project



5. In the **New Job Project** dialog box, enter the following:
   - Programming job file: Click **Browse** and navigate to the location where the job file is located and select the file.
   - FlashPro Express job project location: Select **Browse** and navigate to the location where you want to save the project.
6. Click **Open**. The required programming file is selected and ready to be programmed in the device. The FlashPro Express window appears. Confirm that a programmer number appears in the Programmer field. If it does not, check the board connections and click **Refresh/Rescan Programmers**.
7. Click **RUN** to program the device. When the device is programmed successfully, a RUN PASSED status is displayed.
8. Close FlashPro Express (**Project** > **Exit**).
9. Power-cycle the board.

5. Running Linux User Applications (Ask a Question)

The Linux `.wic` image is packaged with all the demo applications. Extract the `core-image-minimal-dev-mpfs-video-kit-xxxx.rootfs.wic.gz` with 7zip utility to any folder. The extracted `core-image-minimal-dev-mpfs-video-kit-xxxx.rootfs.wic` image can be flashed either in eMMC or SD card. Irrespective of the flashing device, running the demo user applications remain the same.

Important: If the Linux image has multi-compression extensions like `wic.gz.zip`, extract till `.wic` is obtained.

5.1 Flashing Linux .wic Image in eMMC Mode (Ask a Question)

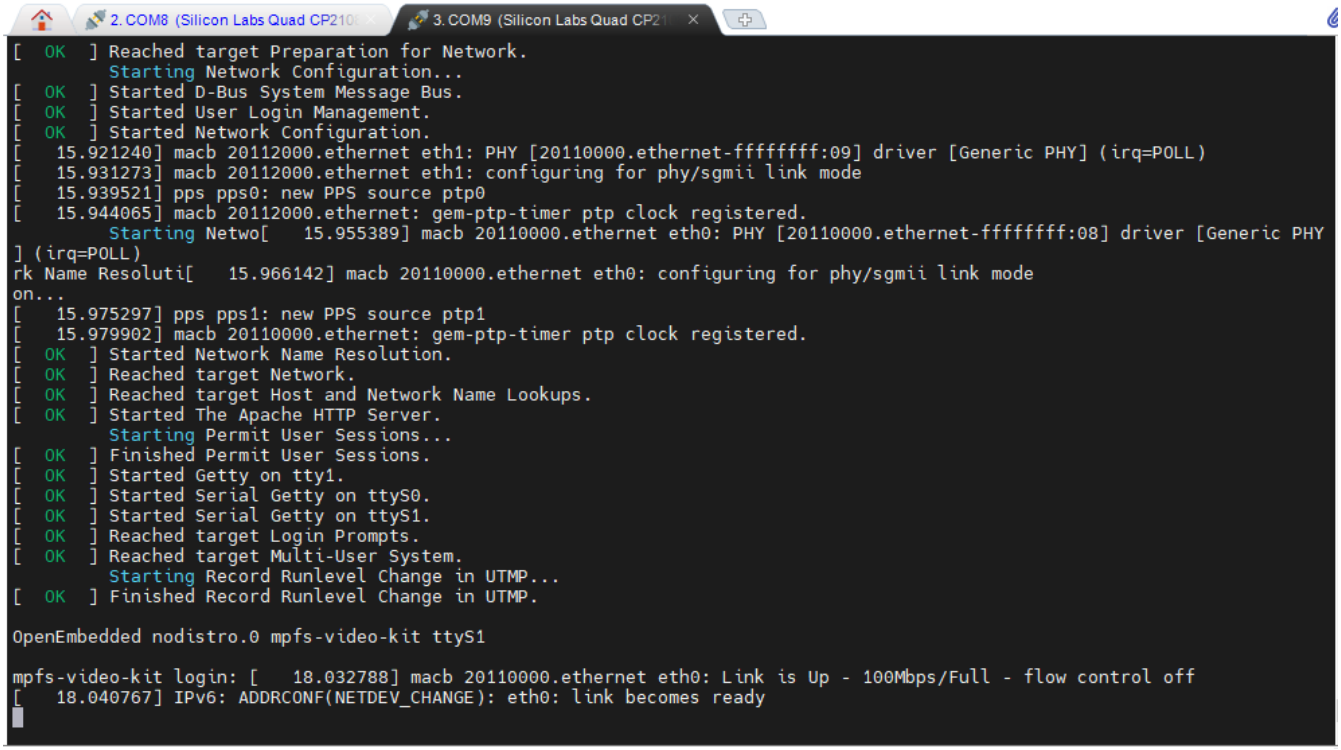
This section describes the steps to put the HSS in CLI mode, program the Linux images into eMMC using the USBImager application and initiate the Linux boot.

1. Do not insert SD card to the board. If present, SD card booting takes priority.
2. Remove the micro-USB cable connected to **J5** and connect it to **J19** (USB-OTG connector) of the board. When the USB cable is detected, the LED near **J19** glows in green.
3. Connect one more micro-USB cable to **J12** from the host PC.
4. **J18** and **J57** must be open.
5. Close pins 2 and 3 of **J36**.
6. Power ON the board and press Enter on Interface 0 while booting.

Important: The COM port number might be different for different users. To identify the COM port, see 4.2. Setting Up the Serial Terminal.

7. Follow the steps in eMMC Content Update Procedure.
8. After programming the eMMC and rebooting, check the serial terminal application connected to Interface 1. Linux will be up and running, waiting for user to login.

Figure 5-1. Linux User Login



This step demonstrated the process of running the Linux user application by programing the Linux user application image onto the on-board 8 GB eMMC NAND Flash using the `.wic` file. Upon completion the Linux OS booted up successfully from the eMMC as indicated by the Linux OS login prompt displayed to the user.

5.2 Flashing Linux .wic Image in SD Card Mode (Ask a Question)

This section describes the steps to program the Linux images into SD card using the SD card reader and USBImager application, and initiate the Linux boot.

Important: This step is required only to boot from SD card instead of on-board 8 GB eMMC NAND Flash. Skip this step to boot the Linux from eMMC Flash and run the demo.

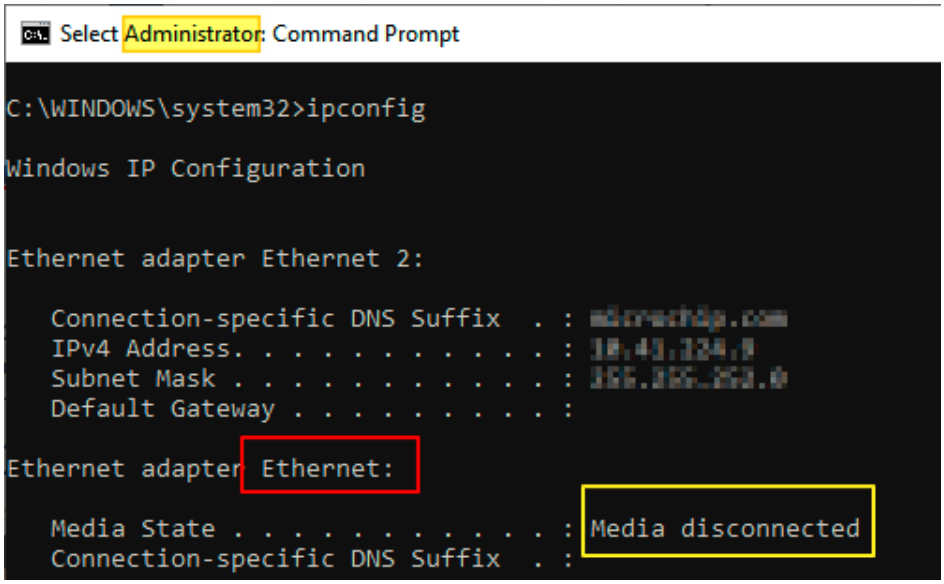
1. Power OFF the board.
2. Follow the steps in SD Card Content Updated Procedure.

5.3 Running the H.264 Demo using GUI (Ask a Question)

To run the H.264 demo using GUI, follow these steps:

1. Insert the dual-camera sensor module in **J10** on the PolarFire SoC video kit. Ensure to remove the camera lens cap.
2. Do not connect any Ethernet cable from the host to the video kit.
3. On the windows host, open command prompt in Admin mode and type `ipconfig`.

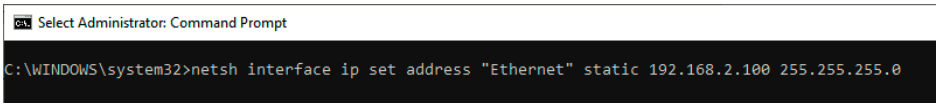
Figure 5-2. Executing the `ipconfig` Command



4. Make a note of the Ethernet adapter name displayed against the Media disconnected interface. It is **Ethernet** in this case.
5. Set the host IP address to 192.168.2.X, where X is any integer from 2 to 255 and the Subnet mask to 255.255.255.0 using the following command. This is a one-time process and need not to be done every time the video kit is connected.

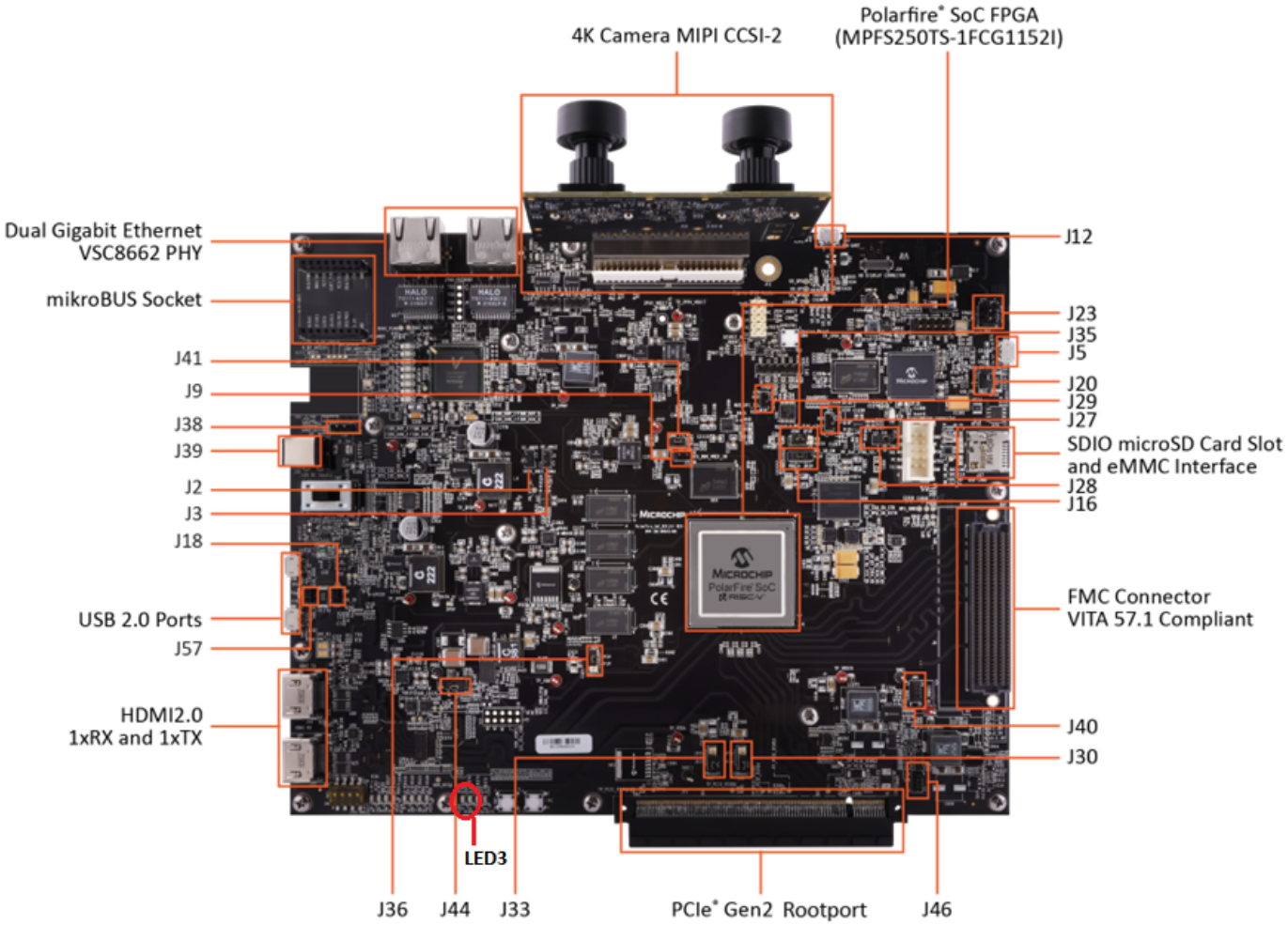
```
netsh interface ip set address "Ethernet" static 192.168.2.100 255.255.255.0
```

Figure 5-3. Setting the Host IP Address



6. Connect an Ethernet cable from video kit **eth0** port to the PC Ethernet port. There are two Ethernet ports on the video kit. The Ethernet port **eth0** is next to the camera mount.
7. Power ON the board and wait for the system boot sequence to complete. The LED highlighted (LED 3) in the following figure glows green once the Linux boot up sequence is completed.
8. Restart the board if LED 3 does not glow.

Figure 5-4. PolarFire SoC Video Kit—LED Indication



9. Log in to Linux using the user name `root` (password is not required).
10. To configure the board for static IP, follow the steps mentioned in Steps to configure static IP addresses.
11. Reboot the video kit and log in again into Linux using the user name `root`.
12. Type `ifconfig`, on the serial terminal connected to Interface 1 (Linux COM Port).
13. It must report some IP, which indicates that the Ethernet port is working, see the following figure.

Figure 5-5. Fetching the Ethernet Port Status

```
OpenEmbedded nodistro.0 mpfs-video-kit ttyS1

mpfs-video-kit login: [   26.096881] macb 20110000.ethernet eth0: Link is Up - 100Mbps/Full - flow control off
[   26.104846] IPv6: ADDRCONF(NETDEV_CHANGE): eth0: link becomes ready
root
This is version v2023.02 of the Polarfire SoC Yocto BSP.

Updated images and documentation are available at:
        https://github.com/polarfire-soc/
* FPGA reference design
        https://github.com/polarfire-soc/icicle-kit-reference-design/releases
* Yocto releases
        https://github.com/polarfire-soc/meta-polarfire-soc-yocto-bsp/releases
* Buildroot External
        https://github.com/linux4microchip/buildroot-external-microchip
root@mpfs-video-kit:~# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.2.1  netmask 255.255.255.0  broadcast 192.168.2.255
        inet6 fe80::204:a3ff:fe65:672e  prefixlen 64  scopeid 0x20<link>
        ether 00:04:a3:65:67:2e  txqueuelen 1000  (Ethernet)
        RX packets 675  bytes 197986 (193.3 KiB)
        RX errors 0  dropped 4  overruns 0  frame 0
        TX packets 87  bytes 5964 (5.8 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
        device interrupt 13
```

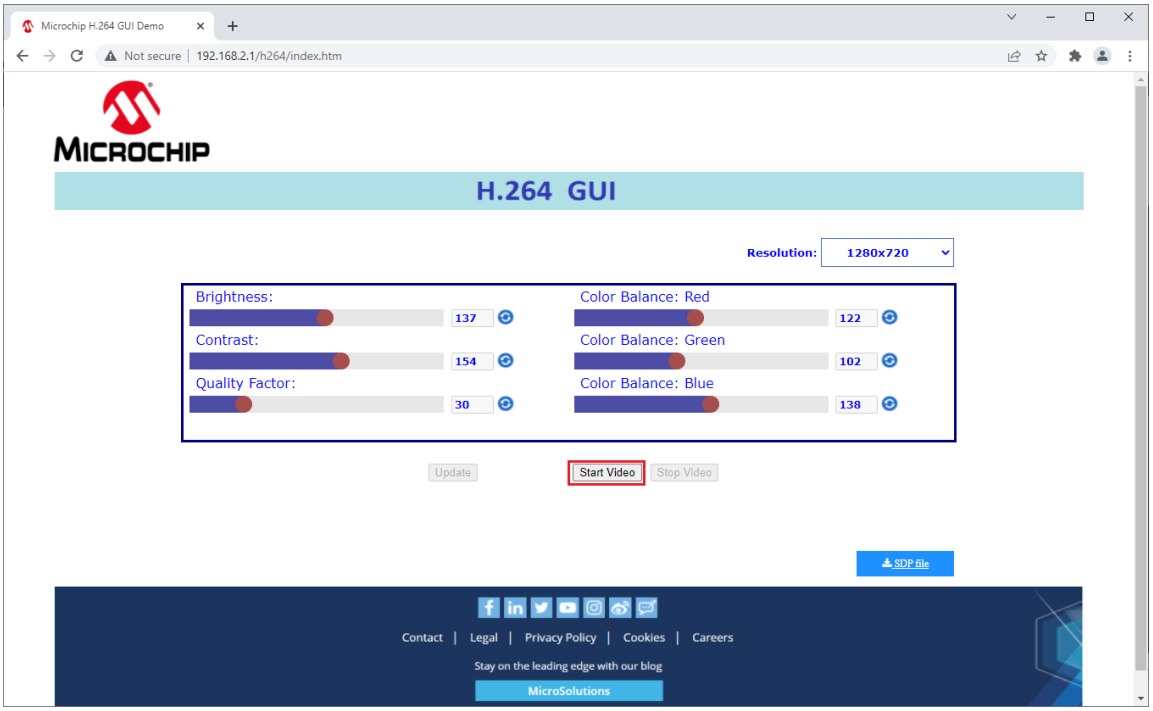
14. Check if the connection is successful by pinging the video kit, see the following figure.

Figure 5-6. Pinging the Video Kit

```
C:\WINDOWS\system32>ping 192.168.2.1

Pinging 192.168.2.1 with 32 bytes of data:
Reply from 192.168.2.1: bytes=32 time=1ms TTL=64
Reply from 192.168.2.1: bytes=32 time=1ms TTL=64
Reply from 192.168.2.1: bytes=32 time<1ms TTL=64
Reply from 192.168.2.1: bytes=32 time=3ms TTL=64

Ping statistics for 192.168.2.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 3ms, Average = 1ms

C:\WINDOWS\system32>
```

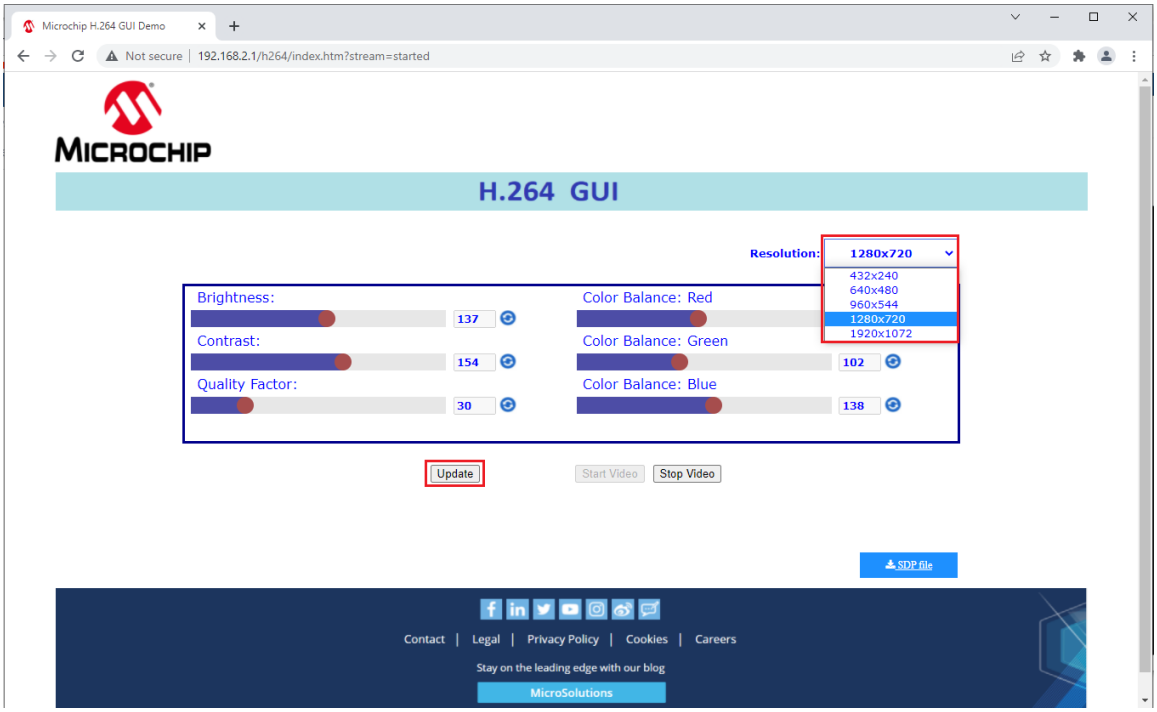
15. From the host PC, open the web browser. Enter the IP address of the PolarFire SoC video kit, depending on the Gigabit Ethernet MAC (GEM) you are connected to (J6: 192.168.2.1 or J7: 192.168.2.2), in the address bar and press **Enter**. The H.264 demonstration GUI loads in the browser, see the following figure.

Figure 5-7. H.264 GUI



16. Click the **Start Video** to initiate the video streaming. The GUI creates a Session Description Protocol (SDP) file. VLC player uses this to accept streaming packets from the Ethernet cable.
    - Click **Download SDP file** to download the SDP file.
    - Windows®: Open the SDP file with the VLC player to play the video stream.
    - Linux: Open the SDP file with the VLC player or execute the `vlc` command in the terminal, with the SDP file as argument, for example, `vlc video.sdp`.
17. Observe a live stream video captured from one of the cameras in the PolarFire SoC video kit. This is a scaled and H.264 compressed video of 1280x720 resolution. To change the resolution, select the resolution as shown in the following figure and click **Update**.

Figure 5-8. Select Resolution



18. Observe the live stream video in the VLC player, as shown in the following figure. Adjust the VLC configurations to reduce the lag in streaming. For more information about VLC configurations, see 6. Appendix A: VLC Configurations.

Figure 5-9. Live Stream Video

This concludes the demo.

To get back from static to DHCP (dynamic), use the below command in the Windows command-line prompt:

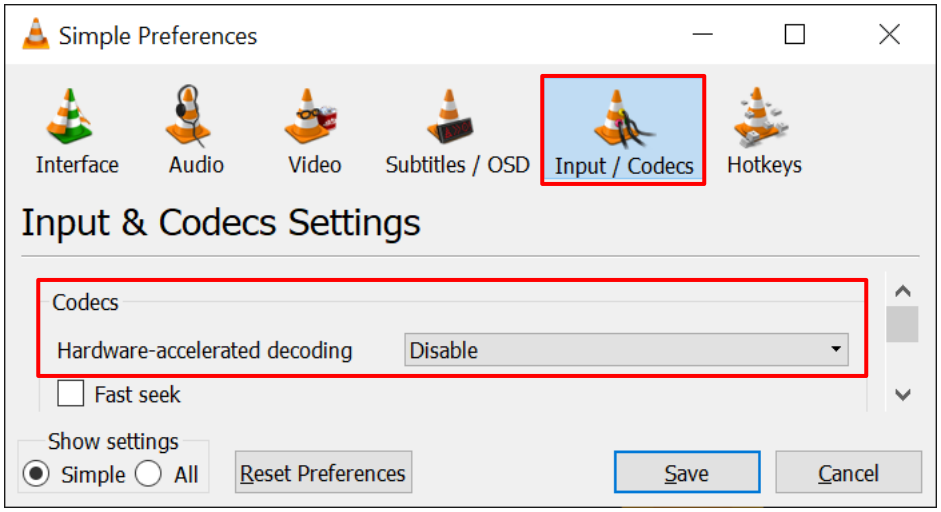
```
netsh interface ip set address name="Ethernet" dhcp
```

6. Appendix A: VLC Configurations (Ask a Question)

Launch the VLC player and configure the following for H.264 streaming.

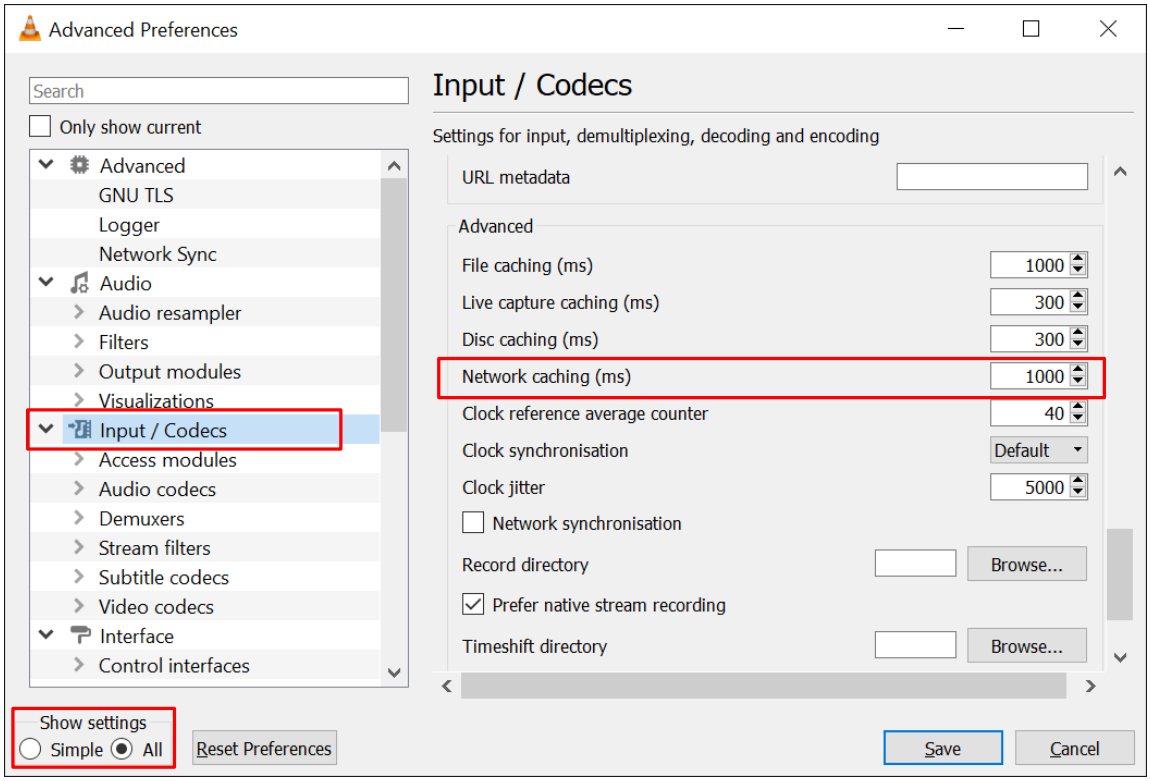
1. Disable the hardware acceleration.
   a. On the VLC media player menu bar, select **Tools** > **Preference**.
   b. In the **Simple Preferences** window, click the **Input / Codecs** tab and set the **Hardware-accelerated decoding** as **Disable**.

   Figure 6-1. Simple Preference—Input / Codecs



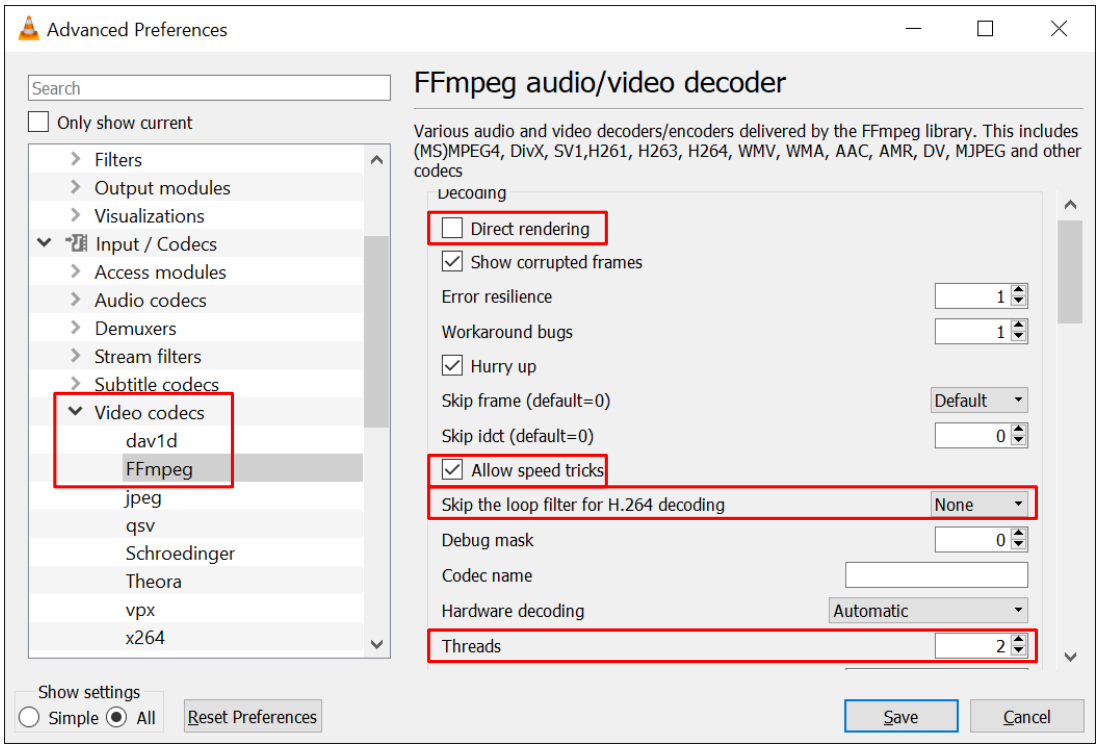
2. Alter the Network Caching.
   a. In the VLC media player menu bar, select **Tools** > **Preference**.
   b. Select **All** under **Show settings** for **Advanced Preferences**.
   c. In the **Advanced Preferences** window, select **Input / Codecs** and alter the **Network Caching (ms)** if the file you are trying to play is located on a network share. Increasing the Network Caching (ms) value increases the latency of video playing in VLC.

Figure 6-2. Advanced Preferences—Input / Codecs



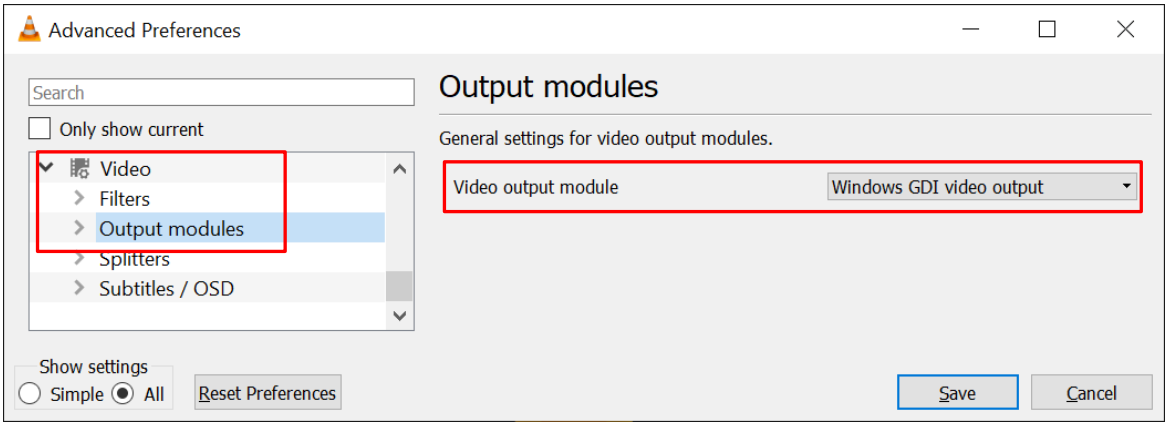
3. Configure the FFMPEG Video Decoder Parameters.
   a. Select **Input / Codecs** > **Video codecs** > **FFmjpeg**.
      i. Clear **Direct rendering**.
      ii. Select **Allow speed tricks** and set **Skip the loop filter for H.264 decoding** to **None**.
      iii. Set **Threads** to 2.

Figure 6-3. Advanced Preferences—Video Codecs



4. Configure the video output module.
   a. Select **Video** > **Output modules** and set **Video output module** as **Windows GDI video output**.

Figure 6-4. Advanced Preferences—Video Output Modules



5. Click **Save**.

7. Appendix B: Running the Tcl Script (Ask a Question)

Tcl scripts are provided in PolarFire SoC Video Kit Reference Design.

To run Tcl, follow these steps:

1. Launch the Libero software
2. Select **Project > Execute Script....**
3. Click **Browse** and select **MPFS_VIDEO_KIT_REFERENCE_DESIGN.tcl** from the downloaded TCL_Scripts directory.
4. Add any required arguments (for example, `HSS_UPDATE PROGRAM`)
5. Click **Run**.

After successful execution of the Tcl script, the Libero project is created within the **top** directory.

Important: The H.264 I-Frame Encoder Encrypted IP is Licensed. If the license is not available, the Tcl generates the design with an Evaluation license which expires after an hour's use on the hardware. So, the streaming stops after an hour.

For more information about Tcl scripts, see `readme.md`.

See Tcl Commands Reference Guide for more information about Tcl commands. Contact Technical Support for any queries about running the Tcl script.

8. Revision History (Ask a Question)

Revision	Date	Description
B	03/2023	• Replaced all instances of “SEV” with “Video” throughout the document. • Updated for Libero SoC v2022.3. • Added reference to GitHub for programming a Linux® image on the on-board eMMC and SD Card. See 5.1. Flashing Linux .wic Image in eMMC Mode and 5.2. Flashing Linux .wic Image in SD Card Mode.
A	04/2022	The first publication of this document.

Microchip FPGA Support (Ask a Question)

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 prior to contacting support as it is very likely that their queries have been already answered.

Contact Technical Support Center through the website at www.microchip.com/support. Mention the FPGA Device Part number, select 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 (Ask a Question)

The Microchip Website (Ask a Question)

Microchip provides online support via our website at www.microchip.com/. This website is used to make files and information easily available to customers. Some of the content available includes:

- **Product Support** – Data sheets and errata, application notes and sample programs, design resources, user's guides and hardware support documents, latest software releases and archived software
- **General Technical Support** – Frequently Asked Questions (FAQs), technical support requests, online discussion groups, Microchip design partner program member listing
- **Business of Microchip** – Product selector and ordering guides, latest Microchip press releases, listing of seminars and events, listings of Microchip sales offices, distributors and factory representatives

Product Change Notification Service (Ask a Question)

Microchip's product change notification service helps keep customers current on Microchip products. Subscribers will receive email notification whenever there are changes, updates, revisions or errata related to a specified product family or development tool of interest.

To register, go to www.microchip.com/pcn and follow the registration instructions.

Customer Support (Ask a Question)

Users of Microchip products can receive assistance through several channels:

- Distributor or Representative
- Local Sales Office
- Embedded Solutions Engineer (ESE)
- Technical Support

Customers should contact their distributor, representative or ESE for support. Local sales offices are also available to help customers. A listing of sales offices and locations is included in this document.

Technical support is available through the website at: www.microchip.com/support

Microchip Devices Code Protection Feature (Ask a Question)

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 product is 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.

Legal Notice (Ask a Question)

This publication and the information herein may be used only with Microchip products, including to design, test, and integrate Microchip products with your application. Use of this information in any other manner violates these terms. Information regarding device applications is provided only for your convenience and may be superseded by updates. It is your responsibility to ensure that your application meets with your specifications. Contact your local Microchip sales office for additional support or, obtain additional support at www.microchip.com/en-us/support/design-help/client-support-services.

THIS INFORMATION IS PROVIDED BY MICROCHIP "AS IS". MICROCHIP MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WHETHER EXPRESS OR IMPLIED, WRITTEN OR ORAL, STATUTORY OR OTHERWISE, RELATED TO THE INFORMATION INCLUDING BUT NOT LIMITED TO ANY IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY, AND FITNESS FOR A PARTICULAR PURPOSE, OR WARRANTIES RELATED TO ITS CONDITION, QUALITY, OR PERFORMANCE.

IN NO EVENT WILL MICROCHIP BE LIABLE FOR ANY INDIRECT, SPECIAL, PUNITIVE, INCIDENTAL, OR CONSEQUENTIAL LOSS, DAMAGE, COST, OR EXPENSE OF ANY KIND WHATSOEVER RELATED TO THE INFORMATION OR ITS USE, HOWEVER CAUSED, EVEN IF MICROCHIP HAS BEEN ADVISED OF THE POSSIBILITY OR THE DAMAGES ARE FORESEEABLE. TO THE FULLEST EXTENT ALLOWED BY LAW, MICROCHIP'S TOTAL LIABILITY ON ALL CLAIMS IN ANY WAY RELATED TO THE INFORMATION OR ITS USE WILL NOT EXCEED THE AMOUNT OF FEES, IF ANY, THAT YOU HAVE PAID DIRECTLY TO MICROCHIP FOR THE INFORMATION.

Use of Microchip devices in life support and/or safety applications is entirely at the buyer's risk, and the buyer agrees to defend, indemnify and hold harmless Microchip from any and all damages, claims, suits, or expenses resulting from such use. No licenses are conveyed, implicitly or otherwise, under any Microchip intellectual property rights unless otherwise stated.

Trademarks (Ask a Question)

The Microchip name and logo, the Microchip logo, Adaptec, AVR, AVR logo, AVR Freaks, BesTime, BitCloud, CryptoMemory, CryptoRF, dsPIC, flexPWR, HELDO, IGLOO, JukeBlox, KeeLoq, Kleer, LANCheck, LinkMD, maXStylus, maXTouch, MediaLB, megaAVR, Microsemi, Microsemi logo, MOST, MOST logo, MPLAB, OptoLyzer, PIC, picoPower, PICSTART, PIC32 logo, PolarFire, Prochip Designer, QTouch, SAM-BA, SenGenuity, SpyNIC, SST, SST Logo, SuperFlash, Symmetricom, SyncServer, Tachyon, TimeSource, tinyAVR, UNI/O, Vectron, and XMEGA are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

AgileSwitch, APT, ClockWorks, The Embedded Control Solutions Company, EtherSynch, Flashtec, Hyper Speed Control, HyperLight Load, Libero, motorBench, mTouch, Powermite 3, Precision Edge, ProASIC, ProASIC Plus, ProASIC Plus logo, Quiet- Wire, SmartFusion, SyncWorld, Temux, TimeCesium, TimeHub, TimePictra, TimeProvider, TrueTime, and ZL are registered trademarks of Microchip Technology Incorporated in the U.S.A.

Adjacent Key Suppression, AKS, Analog-for-the-Digital Age, Any Capacitor, AnyIn, AnyOut, Augmented Switching, BlueSky, BodyCom, Clockstudio, CodeGuard, CryptoAuthentication, CryptoAutomotive, CryptoCompanion, CryptoController, dsPICDEM, dsPICDEM.net, Dynamic Average Matching, DAM, ECAN, Espresso T1S, EtherGREEN, GridTime, IdealBridge, In-Circuit Serial Programming, ICSP, INICnet, Intelligent Paralleling, IntelliMOS, Inter-Chip Connectivity, JitterBlocker, Knob-on-Display, KoD, maxCrypto, maxView, memBrain, Mindi, MiWi, MPASM, MPF, MPLAB Certified logo, MPLIB, MPLINK, MultiTRAK, NetDetach, Omniscient Code Generation, PICDEM, PICDEM.net, PICkit, PICtail, PowerSmart, PureSilicon, QMatrix, REAL ICE, Ripple Blocker, RTAX, RTG4, SAM-

ISBN: 978-1-6683-2208-6

Quality Management System (Ask a Question)

For information regarding Microchip's Quality Management Systems, please visit www.microchip.com/quality.

Worldwide Sales and Service

AMERICAS	ASIA/PACIFIC	ASIA/PACIFIC	EUROPE
Corporate Office 2355 West Chandler Blvd. Chandler, AZ 85224-6199 Tel: 480-792-7200 Fax: 480-792-7277 Technical Support: www.microchip.com/support Web Address: www.microchip.com **Atlanta** Duluth, GA Tel: 678-957-9614 Fax: 678-957-1455 **Austin, TX** Tel: 512-257-3370 **Boston** Westborough, MA Tel: 774-760-0087 Fax: 774-760-0088 **Chicago** Itasca, IL Tel: 630-285-0071 Fax: 630-285-0075 **Dallas** Addison, TX Tel: 972-818-7423 Fax: 972-818-2924 **Detroit** Novi, MI Tel: 248-848-4000 **Houston, TX** Tel: 281-894-5983 **Indianapolis** Noblesville, IN Tel: 317-773-8323 Fax: 317-773-5453 Tel: 317-536-2380 **Los Angeles** Mission Viejo, CA Tel: 949-462-9523 Fax: 949-462-9608 Tel: 951-273-7800 **Raleigh, NC** Tel: 919-844-7510 **New York, NY** Tel: 631-435-6000 **San Jose, CA** Tel: 408-735-9110 Tel: 408-436-4270 **Canada - Toronto** Tel: 905-695-1980 Fax: 905-695-2078	**Australia - Sydney** Tel: 61-2-9868-6733 **China - Beijing** Tel: 86-10-8569-7000 **China - Chengdu** Tel: 86-28-8665-5511 **China - Chongqing** Tel: 86-23-8980-9588 **China - Dongguan** Tel: 86-769-8702-9880 **China - Guangzhou** Tel: 86-20-8755-8029 **China - Hangzhou** Tel: 86-571-8792-8115 **China - Hong Kong SAR** Tel: 852-2943-5100 **China - Nanjing** Tel: 86-25-8473-2460 **China - Qingdao** Tel: 86-532-8502-7355 **China - Shanghai** Tel: 86-21-3326-8000 **China - Shenyang** Tel: 86-24-2334-2829 **China - Shenzhen** Tel: 86-755-8864-2200 **China - Suzhou** Tel: 86-186-6233-1526 **China - Wuhan** Tel: 86-27-5980-5300 **China - Xian** Tel: 86-29-8833-7252 **China - Xiamen** Tel: 86-592-2388138 **China - Zhuhai** Tel: 86-756-3210040	**India - Bangalore** Tel: 91-80-3090-4444 **India - New Delhi** Tel: 91-11-4160-8631 **India - Pune** Tel: 91-20-4121-0141 **Japan - Osaka** Tel: 81-6-6152-7160 **Japan - Tokyo** Tel: 81-3-6880- 3770 **Korea - Daegu** Tel: 82-53-744-4301 **Korea - Seoul** Tel: 82-2-554-7200 **Malaysia - Kuala Lumpur** Tel: 60-3-7651-7906 **Malaysia - Penang** Tel: 60-4-227-8870 **Philippines - Manila** Tel: 63-2-634-9065 **Singapore** Tel: 65-6334-8870 **Taiwan - Hsin Chu** Tel: 886-3-577-8366 **Taiwan - Kaohsiung** Tel: 886-7-213-7830 **Taiwan - Taipei** Tel: 886-2-2508-8600 **Thailand - Bangkok** Tel: 66-2-694-1351 **Vietnam - Ho Chi Minh** Tel: 84-28-5448-2100	**Austria - Wels** Tel: 43-7242-2244-39 Fax: 43-7242-2244-393 **Denmark - Copenhagen** Tel: 45-4485-5910 Fax: 45-4485-2829 **Finland - Espoo** Tel: 358-9-4520-820 **France - Paris** Tel: 33-1-69-53-63-20 Fax: 33-1-69-30-90-79 **Germany - Garching** Tel: 49-8931-9700 **Germany - Haan** Tel: 49-2129-3766400 **Germany - Heilbronn** Tel: 49-7131-72400 **Germany - Karlsruhe** Tel: 49-721-625370 **Germany - Munich** Tel: 49-89-627-144-0 Fax: 49-89-627-144-44 **Germany - Rosenheim** Tel: 49-8031-354-560 **Israel - Ra'anana** Tel: 972-9-744-7705 **Italy - Milan** Tel: 39-0331-742611 Fax: 39-0331-466781 **Italy - Padova** Tel: 39-049-7625286 **Netherlands - Drunen** Tel: 31-416-690399 Fax: 31-416-690340 **Norway - Trondheim** Tel: 47-72884388 **Poland - Warsaw** Tel: 48-22-3325737 **Romania - Bucharest** Tel: 40-21-407-87-50 **Spain - Madrid** Tel: 34-91-708-08-90 Fax: 34-91-708-08-91 **Sweden - Gothenberg** Tel: 46-31-704-60-40 **Sweden - Stockholm** Tel: 46-8-5090-4654 **UK - Wokingham** Tel: 44-118-921-5800 Fax: 44-118-921-5820