XFUSE.AI

XFUSE PHOENIX ISP & IMAGE SENSOR MODULE— DATA SHEET AMD/XILINX KRIA KV260 STARTER KIT

Overview

Xfuse Phoenix is an HDR Image Signal Processor (ISP) optimized for the Kria™ KV260 Vision AI Starter Kit. When the Xfuse (ISP) is combined with the Xfuse image sensor module (AR0239), users can take advantage of this full featured ISP to custom develop vision and deep learning applications with real-time performance in high contrast environments with up to 120dB of dynamic range. Designed for plug-and-play performance with Kria SOM, the (AR0239) imaging module and system are ready for quick deployment in challenging environments encountered in AI, security, robotics, or consumer applications.

Sensor Module Specifications

KV260 Vision AI Starter Kit + Xfuse Image Sensor Module

Xfuse Image Sensor Module (AR0239)

Item		Parameter	
Module P/N		SMK-239-E345-1	
Sensor Type		On-Semi AR0239	
Active Array Size		1936(H) x 1188(V) (16:9)	
Power Supply	Digital Core	1.2 V	
	Analog	2.8 V	
	Digital I/O	1.8 V	
Lens Size		1/2.7”	
F. No		2.0 ±5%	
FOV(D)		128.7°	
Focal Length		2.9±5% mm	
Optical Distortion		<-48.5%	
Pixel Size		3um x 3um	
IR Filter Spec		420-630nm	Tavg>92%
		650±10nm	T=50%
		690-700nm	Tmax<3%
		700-1000nm	Tmax<1%
		1000-1100nm	Tmax<5%
Sensor Temperature Range	Operating	-30°C~+85°C (Junction)	
	Stable Image	0°C~+50°C	
Maximum Image Transfer Rate		1080P (90FPS)	
S/N Ratio (Max)		43dB	
Scan Mode		Progressive	
Power Consumption (Typical)	Linear HiSPi	218mW, 1080P (30FPS)	
	Linear MIPI	190mW, 1080P (30FPS)	
	Line-Interleaved HiSPi	264mW, 1080P (30FPS)	
	Line-Interleaved MIPI	252mW, 1080P (30FPS)	
Output Interface		4-lane HiSPi / MIPI	
Output Format		10-12bit RAW	
Color		RGB	
Connector P/N		HRS BM20B (0.8)-34DP-0.4V (51)	

Table of Contents

1. Introduction

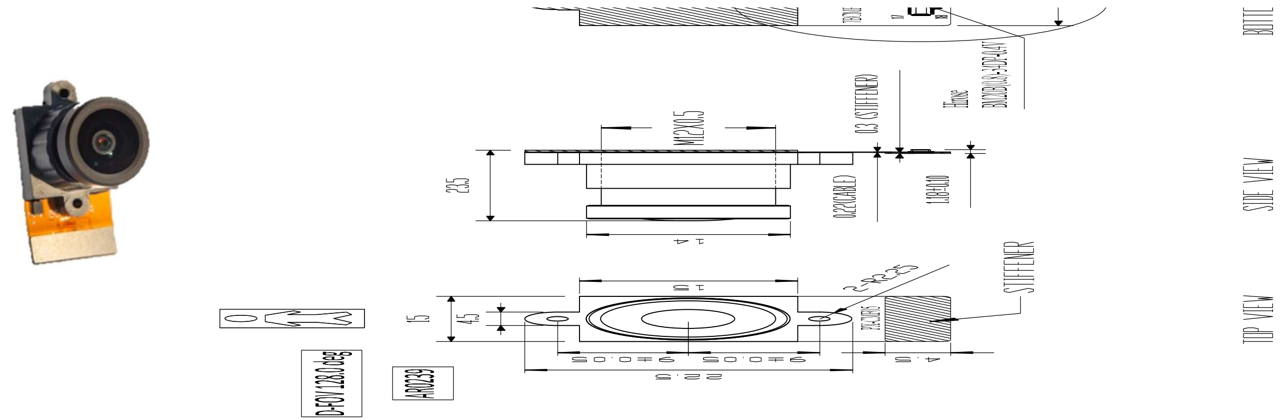
Thank-you for purchasing the Xfuse Phoenix ISP Accelerated Application. This document provides end user with step-by-step instructions to run Xfuse Phoenix ISP Accelerated Application on Kria KV260 Vision AI starter kit, enable one instance of Phoenix GStreamer application pipeline, and control of the Phoenix ISP using the Xfuse PC application.

Should issues arise or require support for this application, please contact us via the Xilinx Kria partner application forums. [Forums -> Kria SOMs]

2. Hardware Requirements

1. Kria KV260 Vision AI starter Kit board (target)
2. Windows machine (host)
3. 16GB microSD card
4. USB-A to micro-B cable
5. RJ45 Ethernet cable
6. Display Port or HDMI Cable
7. 1080p capable DisplayPort or HDMI monitor
8. Xfuse Sensor Module (AR0239) – see specification below

Xfuse Sensor Module (AR0239)



Definition	Symbols	Min	Max	Unit
Analog Voltage	VAA_MAX	-0.3	4	V
Pixel Supply Voltage	VAA_PIX	-0.3	4	V
I/O Digital Voltage	VDD_IO_MAX	-0.3	4	V
Power to MIPI	VDD_IO_PHY	-0.3	4	V
Core Digital Voltage	VDD_MAX	-0.3	2.4	V
HiSPi I/O Digital Voltage	VDD_SLVS_MAX	-0.3	2.4	V
Storage Temperture	tST	-40	125	°C

3. Starter Kit microSD Card Preparation

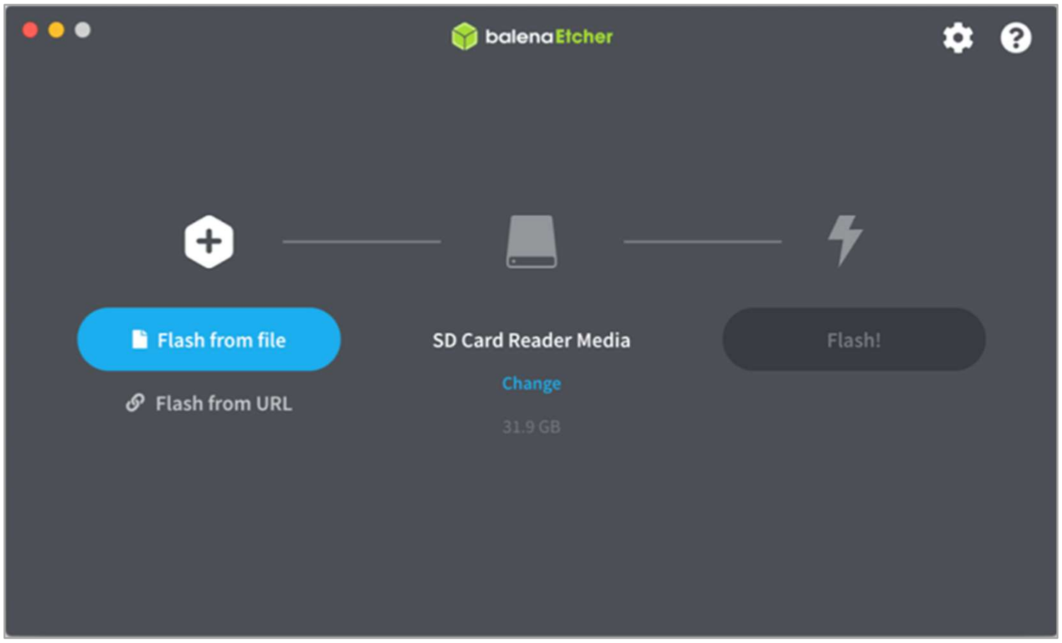
The first step is to prepare a microSD for use with the Starter Kit. The Starter Kit has a primary and secondary boot device, isolating the boot firmware from the run-time OS and application. This is to focus on developing and updating your application code within the application image on the secondary boot device, without having to touch the boot firmware. The primary boot device is a QSPI memory located on the SOM, which is pre-programmed (pre-loaded QSPI image) at the factory. The secondary boot device is a microSD card interface on the carrier card.

For setting up the microSD card, download the latest SD card image and then write it using an Image Flashing tool.

Follow the steps below to image a microSD card for use with the KV260 Starter Kit:

1. Download the SD Card Image, https://www.xilinx.com/member/forms/download/xef.html?filename=petalinux-sdimage.wic.gz (petalinux-sdimage.wic.gz)

   This image and app support petalinux 2020.2 only
2. Unzip the file you downloaded and save it on your computer (petalinux-sdimage.wic)
3. Insert the microSD card into an available slot on your PC
4. Use the Balena Etcher tool, which supports Windows, Linux, and macOS, to flash a suitable microSD card



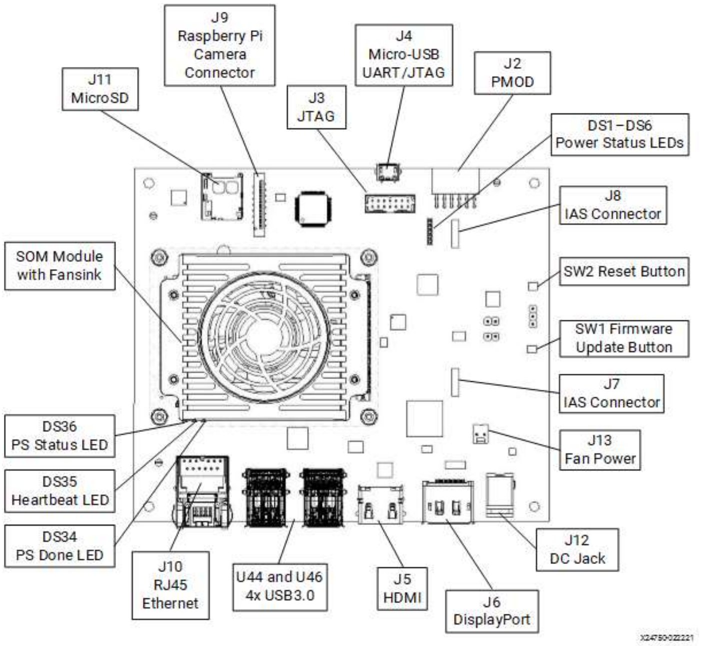
5. Follow the Etcher prompts to select the *.wic image file and flash it to the microSD card as illustrated in the image below. When this process is complete, eject the card from your computer

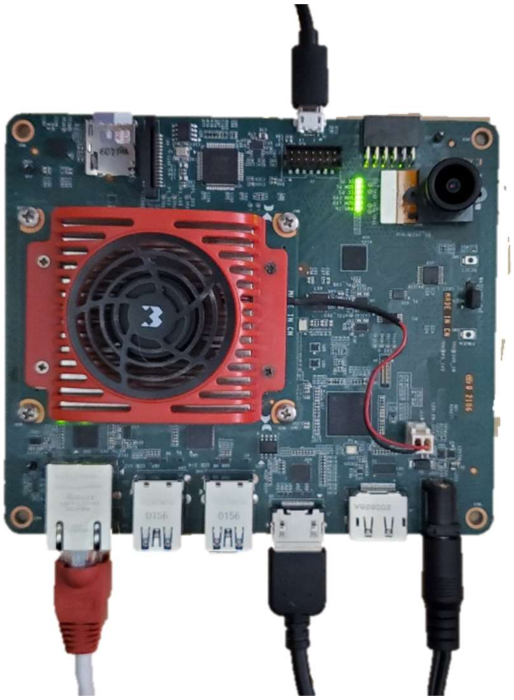
4. Starter Kit Hardware Setup

Next, prepare the KV260 Starter Kit to run the application.

The basic steps are encapsulated below:

1. Insert the microSD card into the slot on the KV260 carrier card (J11)
2. Connect a micro-USB cable to J4 on the KV260 carrier card. This USB serial interface will be leveraged to interact with the application and view boot time messages
3. Connect an Ethernet cable from J10 on the KV260 carrier card to your local network
4. Connect a DisplayPort cable and DisplayPort monitor to J6 (or HDMI to J5) on the KV260 carrier card
5. Connect the Xfuse AR0239 IAS sensor module to IAS1 (J8) on the KV260 carrier card
6. Power on the board





Warning!

Sensor Module Connection

Misorientation of the connector can lead to bad performance or even permanent damage!

Never connect an FSM directly (without FSA) to an FPA, or a carrier that was not designed explicitly for this purpose.

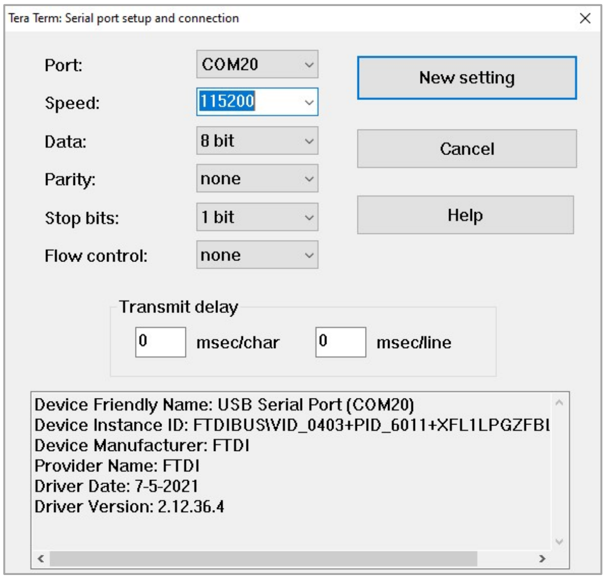
ATTENTION
STATIC SENSITIVE DEVICES
HANDLE ONLY AT
STATIC SAFE WORK STATIONS

Handling ESD Sensitive Components

The electronic components like Printed Circuit Boards (PCB) described in this document are sensitive to Electrostatic Discharge (ESD) and need to be handled with high care in static controlled environments. It is strongly recommended to follow the general handling practices for ESD sensitive parts

5. Connecting the KV260

Connect to the KV260 Starter Kit and bring up the Xfuse Phoenix application.



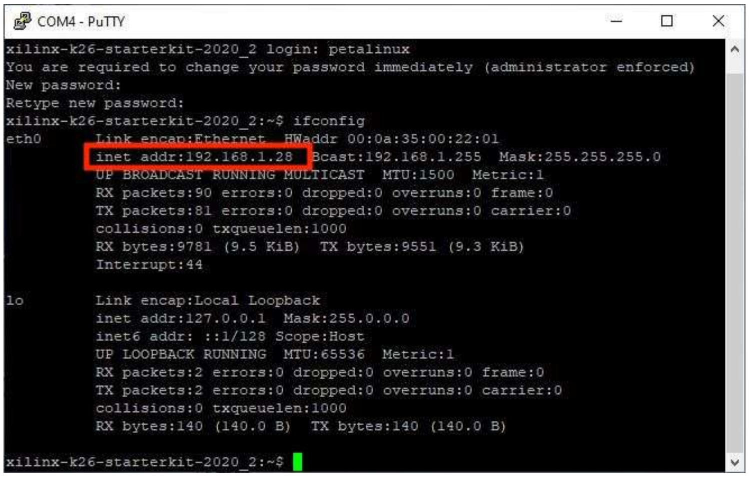
Port: COM20
Speed: 115200
Data: 8 bit
Parity: none
Stop bits: 1 bit
Flow control: none

Transmit delay
0 msec/char 0 msec/line

```
Device Friendly Name: USB Serial Port (COM20)
Device Instance ID: FTDIBUS\VID_0403+PID_6011+XFL1LPGZFBI
Device Manufacturer: FTDI
Provider Name: FTDI
Driver Date: 7-5-2021
Driver Version: 2.12.36.4
```

1. The KV260 Starter Kit uses an FTDI USB to COM port device that requires the FTDI virtual COM (VCOM) port driver be installed on your machine. Set-up your USB-based UART connection via the J4 carrier card interface with the configuration 115200, N, 8, 1, No Flow Control using a terminal program (e.g., TeraTerm, PuTTY, Minicom). The Linux UART is enumerated as the lower of two VCOM ports.
2. After powering on the SOM, it should automatically boot to Linux. The green LED's active and observe the Linux UART response on the terminal program interface. If the LED's active but no UART response, verify that the machine has the FTDI driver installed and that the terminal program is connected to the correct COM port.
3. At initial log-in, the platform requires a password. The default username is petalinux. Make note of the password for future use. The petalinux user does not have root privileges. Most commands used in subsequent tutorials must be run using sudo and may be prompted to enter a password.

   If the Kria SOM is connected to a network that supports DHCP, it can determine the local IP address by issuing the command “ifconfig” via the serial terminal application:



```
xilinx-k26-starterkit-2020_2 login: petalinux
You are required to change your password immediately (administrator enforced)
New password:
Retype new password:
xilinx-k26-starterkit-2020_2:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 00:0a:35:00:22:01
          inet addr:192.168.1.28  Bcast:192.168.1.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:90 errors:0 dropped:0 overruns:0 frame:0
          TX packets:81 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:9781 (9.5 KiB)  TX bytes:9551 (9.3 KiB)
          Interrupt:44

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:2 errors:0 dropped:0 overruns:0 frame:0
          TX packets:2 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:140 (140.0 B)  TX bytes:140 (140.0 B)

xilinx-k26-starterkit-2020_2:~$
```

   It is also possible to manually assign an IP address via the command “sudo ifconfig eth0 xxx.xxx.xxx.xxx”, replacing xxx.xxx.xxx.xxx with the target IP address. This will be required if the network does not support DHCP, or if using a direct connection from the Kria SOM to the designated PC.

   Kria SOM IP address will be needed later, please make a note.

6. Install the Xfuse Phoenix ISP application package

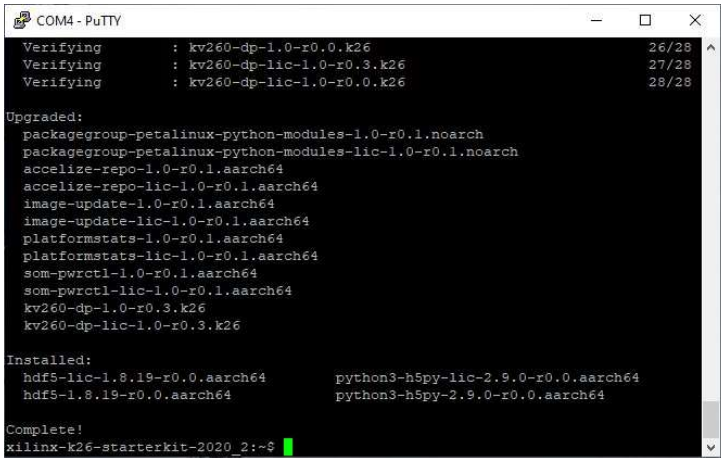
AMD/Xilinx provides the Phoenix package feeds in runtime package management (RPM) format for the users to dynamically load it on top of running Xilinx SOM Starter Linux image using dnf package manager utility command “dnf install”.

First, execute the below commands to perform a system update. ‘dnf clean all’ command cleans the cached RPMs and metadata that dnf stores, and dnf update command pulls the latest Xilinx package feed updates.

6.1 sudo dnf clean all --refresh

6.2 sudo dnf update --refresh

The below screenshot is a snippet of dnf update command for reference:



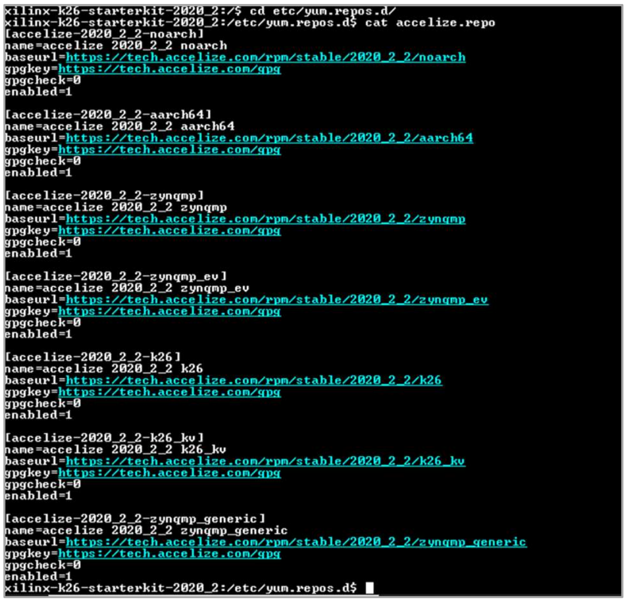
```
  Verifying        : kv260-dp-1.0-r0.0.k26                            26/28
  Verifying        : kv260-dp-lic-1.0-r0.3.k26                        27/28
  Verifying        : kv260-dp-lic-1.0-r0.0.k26                        28/28

Upgraded:
  packagegroup-petalinux-python-modules-1.0-r0.1.noarch
  packagegroup-petalinux-python-modules-lic-1.0-r0.1.noarch
  accelize-repo-1.0-r0.1.aarch64
  accelize-repo-lic-1.0-r0.1.aarch64
  image-update-1.0-r0.1.aarch64
  image-update-lic-1.0-r0.1.aarch64
  platformstats-1.0-r0.1.aarch64
  platformstats-lic-1.0-r0.1.aarch64
  som-pwrctl-1.0-r0.1.aarch64
  som-pwrctl-lic-1.0-r0.1.aarch64
  kv260-dp-1.0-r0.3.k26
  kv260-dp-lic-1.0-r0.3.k26

Installed:
  hdf5-lic-1.8.19-r0.0.aarch64        python3-h5py-lic-2.9.0-r0.0.aarch64
  hdf5-1.8.19-r0.0.aarch64            python3-h5py-2.9.0-r0.0.aarch64

Complete!
xilinx-k26-starterkit-2020_2:~$
```

Expected output: User can observe “Complete!” print message at the end of dnf update command console logs.

Next, we need to enable the accelize.repo by setting the enabled bit to 1. Execute the following command:

6.3 sudo sed -i 's/enabled=0/enabled=1/g' /etc/yum.repos.d/accelize.repo

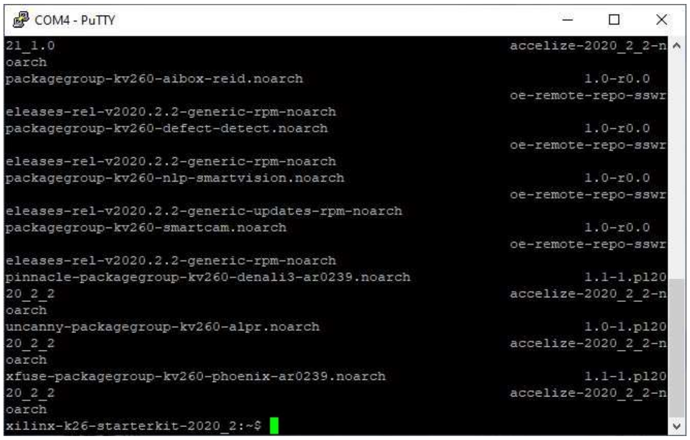
After this command has executed, the accelize repo should look like this:



```
xilinx-k26-starterkit-2020_2:/$ cd etc/yum.repos.d/
xilinx-k26-starterkit-2020_2:/etc/yum.repos.d$ cat accelize.repo
[accelize-2020_2_2-noarch]
name=accelize 2020_2_2 noarch
baseurl=https://tech.accelize.com/rpm/stable/2020_2_2/noarch
gpgkey=https://tech.accelize.com/gpg
gpgcheck=0
enabled=1

[accelize-2020_2_2-aarch64]
name=accelize 2020_2_2 aarch64
baseurl=https://tech.accelize.com/rpm/stable/2020_2_2/aarch64
gpgkey=https://tech.accelize.com/gpg
gpgcheck=0
enabled=1

[accelize-2020_2_2-zynqmp]
name=accelize 2020_2_2 zynqmp
baseurl=https://tech.accelize.com/rpm/stable/2020_2_2/zynqmp
gpgkey=https://tech.accelize.com/gpg
gpgcheck=0
enabled=1

[accelize-2020_2_2-zynqmp_ev]
name=accelize 2020_2_2 zynqmp_ev
baseurl=https://tech.accelize.com/rpm/stable/2020_2_2/zynqmp_ev
gpgkey=https://tech.accelize.com/gpg
gpgcheck=0
enabled=1

[accelize-2020_2_2-k26]
name=accelize 2020_2_2 k26
baseurl=https://tech.accelize.com/rpm/stable/2020_2_2/k26
gpgkey=https://tech.accelize.com/gpg
gpgcheck=0
enabled=1

[accelize-2020_2_2-k26_kv]
name=accelize 2020_2_2 k26_kv
baseurl=https://tech.accelize.com/rpm/stable/2020_2_2/k26_kv
gpgkey=https://tech.accelize.com/gpg
gpgcheck=0
enabled=1

[accelize-2020_2_2-zynqmp_generic]
name=accelize 2020_2_2 zynqmp_generic
baseurl=https://tech.accelize.com/rpm/stable/2020_2_2/zynqmp_generic
gpgkey=https://tech.accelize.com/gpg
gpgcheck=0
enabled=1
xilinx-k26-starterkit-2020_2:/etc/yum.repos.d$
```

Now, run the command below to get the list of available Xfuse accelerated application package groups:

6.4 sudo xmutil getpkgs



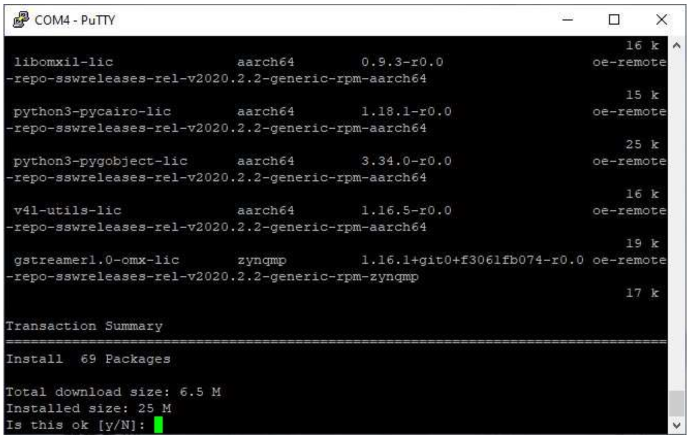
```
21_1.0                                                      accelize-2020_2_2-n
oarch
packagegroup-kv260-aibox-reid.noarch                                  1.0-r0.0
                                                            oe-remote-repo-sswr
eleases-rel-v2020.2.2-generic-rpm-noarch
packagegroup-kv260-defect-detect.noarch                               1.0-r0.0
                                                            oe-remote-repo-sswr
eleases-rel-v2020.2.2-generic-rpm-noarch
packagegroup-kv260-nlp-smartvision.noarch                             1.0-r0.0
                                                            oe-remote-repo-sswr
eleases-rel-v2020.2.2-generic-updates-rpm-noarch
packagegroup-kv260-smartcam.noarch                                    1.0-r0.0
                                                            oe-remote-repo-sswr
eleases-rel-v2020.2.2-generic-rpm-noarch
pinnacle-packagegroup-kv260-denali3-ar0239.noarch                     1.1-1.pl20
20_2_2                                                      accelize-2020_2_2-n
oarch
uncanny-packagegroup-kv260-alpr.noarch                                1.0-1.pl20
20_2_2                                                      accelize-2020_2_2-n
oarch
xfuse-packagegroup-kv260-phoenix-ar0239.noarch                        1.1-1.pl20
20_2_2                                                      accelize-2020_2_2-n
oarch
xilinx-k26-starterkit-2020_2:~$
```

Expected output: Check to be sure the xfuse-packagegroup-kv260-phoenix-ar0239.noarch package in the listing.

If complete, run the dnf install in order to install the Phoenix application on Kria:

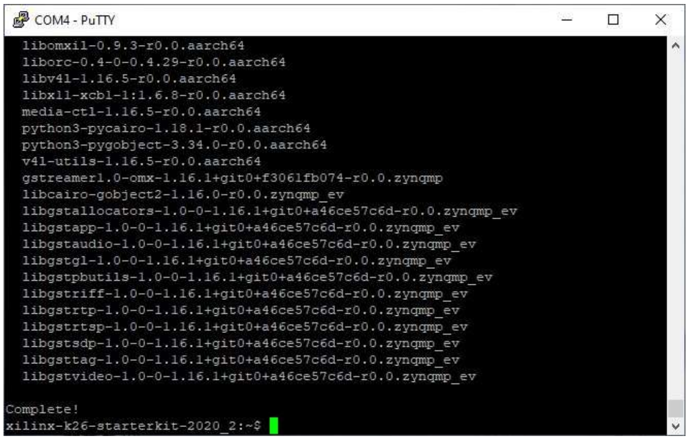
6.5 sudo dnf install xfuse-packagegroup-kv260-phoenix-ar0239.noarch

When prompted, press "y" to install the packages:



```
                                                                          16 k
 libomxil-lic              aarch64        0.9.3-r0.0                 oe-remote
-repo-sswreleases-rel-v2020.2.2-generic-rpm-aarch64
                                                                          15 k
 python3-pycairo-lic       aarch64        1.18.1-r0.0                oe-remote
-repo-sswreleases-rel-v2020.2.2-generic-rpm-aarch64
                                                                          25 k
 python3-pygobject-lic     aarch64        3.34.0-r0.0                oe-remote
-repo-sswreleases-rel-v2020.2.2-generic-rpm-aarch64
                                                                          16 k
 v4l-utils-lic             aarch64        1.16.5-r0.0                oe-remote
-repo-sswreleases-rel-v2020.2.2-generic-rpm-aarch64
                                                                          19 k
 gstreamer1.0-omx-lic      zynqmp         1.16.1+git0+f3061fb074-r0.0 oe-remote
-repo-sswreleases-rel-v2020.2.2-generic-rpm-zynqmp
                                                                          17 k

Transaction Summary
================================================================================
Install  69 Packages

Total download size: 6.5 M
Installed size: 25 M
Is this ok [y/N]:
```

The output log of this command should appear as follows. Indicator should be a "Complete!" message at the end of dnf install command console log.



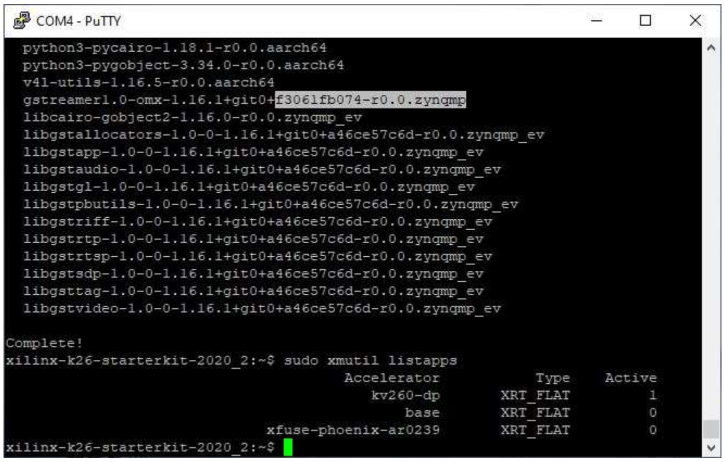
```
  libomxil-0.9.3-r0.0.aarch64
  liborc-0.4-0-0.4.29-r0.0.aarch64
  libv4l-1.16.5-r0.0.aarch64
  libx11-xcb1-1:1.6.8-r0.0.aarch64
  media-ctl-1.16.5-r0.0.aarch64
  python3-pycairo-1.18.1-r0.0.aarch64
  python3-pygobject-3.34.0-r0.0.aarch64
  v4l-utils-1.16.5-r0.0.aarch64
  gstreamer1.0-omx-1.16.1+git0+f3061fb074-r0.0.zynqmp
  libcairo-gobject2-1.16.0-r0.0.zynqmp_ev
  libgstallocators-1.0-0-1.16.1+git0+a46ce57c6d-r0.0.zynqmp_ev
  libgstapp-1.0-0-1.16.1+git0+a46ce57c6d-r0.0.zynqmp_ev
  libgstaudio-1.0-0-1.16.1+git0+a46ce57c6d-r0.0.zynqmp_ev
  libgstgl-1.0-0-1.16.1+git0+a46ce57c6d-r0.0.zynqmp_ev
  libgstpbutils-1.0-0-1.16.1+git0+a46ce57c6d-r0.0.zynqmp_ev
  libgstriff-1.0-0-1.16.1+git0+a46ce57c6d-r0.0.zynqmp_ev
  libgstrtp-1.0-0-1.16.1+git0+a46ce57c6d-r0.0.zynqmp_ev
  libgstrtsp-1.0-0-1.16.1+git0+a46ce57c6d-r0.0.zynqmp_ev
  libgstsdp-1.0-0-1.16.1+git0+a46ce57c6d-r0.0.zynqmp_ev
  libgsttag-1.0-0-1.16.1+git0+a46ce57c6d-r0.0.zynqmp_ev
  libgstvideo-1.0-0-1.16.1+git0+a46ce57c6d-r0.0.zynqmp_ev

Complete!
xilinx-k26-starterkit-2020_2:~$
```

7. Load the ISP application firmware

Prior to executing the installed Phoenix application, its firmware (PL bitstream + Device tree overlay + PL drivers) needs to be loaded using xmutil utility commands. Follow the below steps to load Phoenix application firmware on the KV260 Vision AI starter kit.

Run the below command to list the existing application installed on Vision AI starter kit.

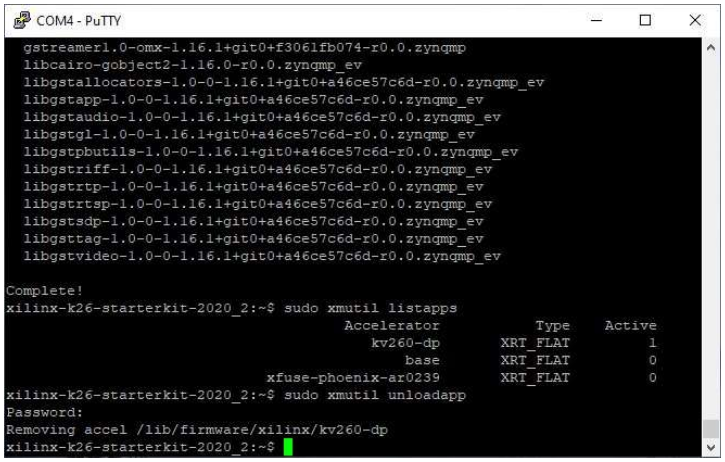
7.1 sudo xmutil listapps



```
  python3-pycairo-1.18.1-r0.0.aarch64
  python3-pygobject-3.34.0-r0.0.aarch64
  v4l-utils-1.16.5-r0.0.aarch64
  gstreamer1.0-omx-1.16.1+git0+f3061fb074-r0.0.zynqmp
  libcairo-gobject2-1.16.0-r0.0.zynqmp_ev
  libgstallocators-1.0-0-1.16.1+git0+a46ce57c6d-r0.0.zynqmp_ev
  libgstapp-1.0-0-1.16.1+git0+a46ce57c6d-r0.0.zynqmp_ev
  libgstaudio-1.0-0-1.16.1+git0+a46ce57c6d-r0.0.zynqmp_ev
  libgstgl-1.0-0-1.16.1+git0+a46ce57c6d-r0.0.zynqmp_ev
  libgstpbutils-1.0-0-1.16.1+git0+a46ce57c6d-r0.0.zynqmp_ev
  libgstriff-1.0-0-1.16.1+git0+a46ce57c6d-r0.0.zynqmp_ev
  libgstrtp-1.0-0-1.16.1+git0+a46ce57c6d-r0.0.zynqmp_ev
  libgstrtsp-1.0-0-1.16.1+git0+a46ce57c6d-r0.0.zynqmp_ev
  libgstsdp-1.0-0-1.16.1+git0+a46ce57c6d-r0.0.zynqmp_ev
  libgsttag-1.0-0-1.16.1+git0+a46ce57c6d-r0.0.zynqmp_ev
  libgstvideo-1.0-0-1.16.1+git0+a46ce57c6d-r0.0.zynqmp_ev

Complete!
xilinx-k26-starterkit-2020_2:~$ sudo xmutil listapps
                                       Accelerator          Type    Active
                                          kv260-dp      XRT_FLAT         1
                                              base      XRT_FLAT         0
                               xfuse-phoenix-ar0239      XRT_FLAT         0
xilinx-k26-starterkit-2020_2:~$ 
```

If any apps are active, you must unload them before loading the Phoenix application. Unload any active apps with the following command:

7.2 sudo xmutil unloadapp

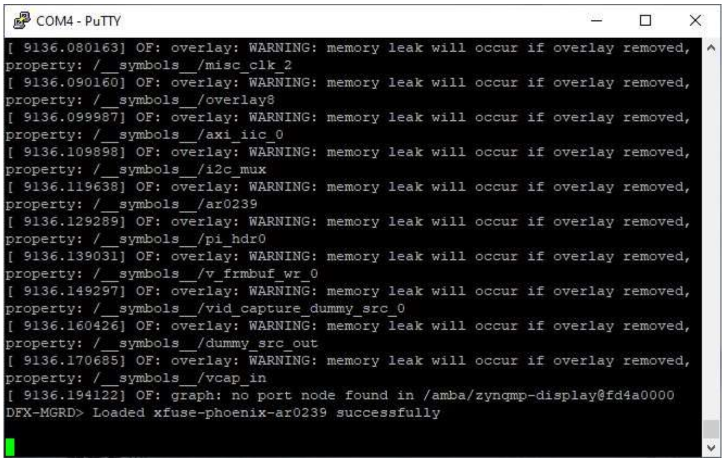


```
  gstreamer1.0-omx-1.16.1+git0+f3061fb074-r0.0.zynqmp
  libcairo-gobject2-1.16.0-r0.0.zynqmp_ev
  libgstallocators-1.0-0-1.16.1+git0+a46ce57c6d-r0.0.zynqmp_ev
  libgstapp-1.0-0-1.16.1+git0+a46ce57c6d-r0.0.zynqmp_ev
  libgstaudio-1.0-0-1.16.1+git0+a46ce57c6d-r0.0.zynqmp_ev
  libgstgl-1.0-0-1.16.1+git0+a46ce57c6d-r0.0.zynqmp_ev
  libgstpbutils-1.0-0-1.16.1+git0+a46ce57c6d-r0.0.zynqmp_ev
  libgstriff-1.0-0-1.16.1+git0+a46ce57c6d-r0.0.zynqmp_ev
  libgstrtp-1.0-0-1.16.1+git0+a46ce57c6d-r0.0.zynqmp_ev
  libgstrtsp-1.0-0-1.16.1+git0+a46ce57c6d-r0.0.zynqmp_ev
  libgstsdp-1.0-0-1.16.1+git0+a46ce57c6d-r0.0.zynqmp_ev
  libgsttag-1.0-0-1.16.1+git0+a46ce57c6d-r0.0.zynqmp_ev
  libgstvideo-1.0-0-1.16.1+git0+a46ce57c6d-r0.0.zynqmp_ev

Complete!
xilinx-k26-starterkit-2020_2:~$ sudo xmutil listapps
                                       Accelerator          Type    Active
                                          kv260-dp      XRT_FLAT         1
                                              base      XRT_FLAT         0
                               xfuse-phoenix-ar0239      XRT_FLAT         0
xilinx-k26-starterkit-2020_2:~$ sudo xmutil unloadapp
Password:
Removing accel /lib/firmware/xilinx/kv260-dp
xilinx-k26-starterkit-2020_2:~$ 
```

Now that no apps are active, we can load the Phoenix application using the following command:

7.4 sudo xmutil loadapp xfuse-phoenix-ar0239

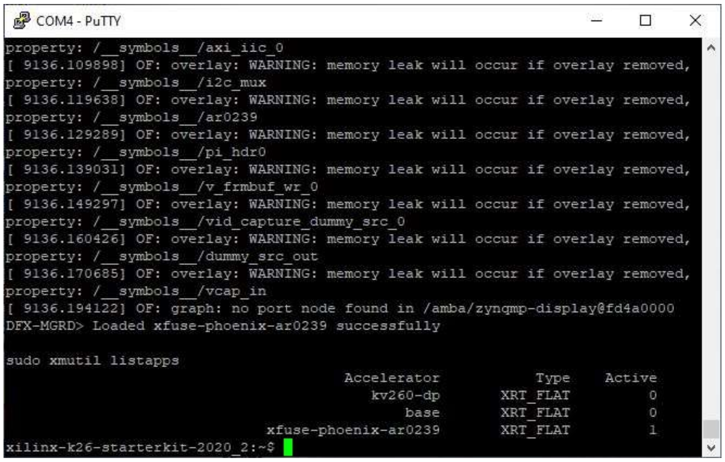
Ignore the overlay warnings. At the bottom, see that xfuse-phoenix-ar0239 has been loaded successfully.



```
[ 9136.080163] OF: overlay: WARNING: memory leak will occur if overlay removed,
property: /__symbols__/misc_clk_2
[ 9136.090160] OF: overlay: WARNING: memory leak will occur if overlay removed,
property: /__symbols__/overlay8
[ 9136.099987] OF: overlay: WARNING: memory leak will occur if overlay removed,
property: /__symbols__/axi_iic_0
[ 9136.109898] OF: overlay: WARNING: memory leak will occur if overlay removed,
property: /__symbols__/i2c_mux
[ 9136.119638] OF: overlay: WARNING: memory leak will occur if overlay removed,
property: /__symbols__/ar0239
[ 9136.129289] OF: overlay: WARNING: memory leak will occur if overlay removed,
property: /__symbols__/pi_hdr0
[ 9136.139031] OF: overlay: WARNING: memory leak will occur if overlay removed,
property: /__symbols__/v_frmbuf_wr_0
[ 9136.149297] OF: overlay: WARNING: memory leak will occur if overlay removed,
property: /__symbols__/vid_capture_dummy_src_0
[ 9136.160426] OF: overlay: WARNING: memory leak will occur if overlay removed,
property: /__symbols__/dummy_src_out
[ 9136.170685] OF: overlay: WARNING: memory leak will occur if overlay removed,
property: /__symbols__/vcap_in
[ 9136.194122] OF: graph: no port node found in /amba/zynqmp-display@fd4a0000
DFX-MGRD> Loaded xfuse-phoenix-ar0239 successfully
```

Check to ensure that the Phoenix app is active by issuing the following command:

7.5 sudo xmutil listapps



```
property: /__symbols__/axi_iic_0
[ 9136.109898] OF: overlay: WARNING: memory leak will occur if overlay removed,
property: /__symbols__/i2c_mux
[ 9136.119638] OF: overlay: WARNING: memory leak will occur if overlay removed,
property: /__symbols__/ar0239
[ 9136.129289] OF: overlay: WARNING: memory leak will occur if overlay removed,
property: /__symbols__/pi_hdr0
[ 9136.139031] OF: overlay: WARNING: memory leak will occur if overlay removed,
property: /__symbols__/v_frmbuf_wr_0
[ 9136.149297] OF: overlay: WARNING: memory leak will occur if overlay removed,
property: /__symbols__/vid_capture_dummy_src_0
[ 9136.160426] OF: overlay: WARNING: memory leak will occur if overlay removed,
property: /__symbols__/dummy_src_out
[ 9136.170685] OF: overlay: WARNING: memory leak will occur if overlay removed,
property: /__symbols__/vcap_in
[ 9136.194122] OF: graph: no port node found in /amba/zynqmp-display@fd4a0000
DFX-MGRD> Loaded xfuse-phoenix-ar0239 successfully

sudo xmutil listapps
                                       Accelerator           Type    Active
                                          kv260-dp       XRT_FLAT         0
                                              base       XRT_FLAT         0
                              xfuse-phoenix-ar0239       XRT_FLAT         1
xilinx-k26-starterkit-2020_2:~$
```

The application was successfully loaded and is now the active application on the Kria KV260 Starter Kit.

8. Starting the Phoenix Application

Now it is time to start the Xfuse Phoenix ISP Application. Ensure that the DisplayPort (or HDMI) connector is plugged into the designated monitor. Ensure that the IP address of the Kria SOM board, this is required to control the camera.

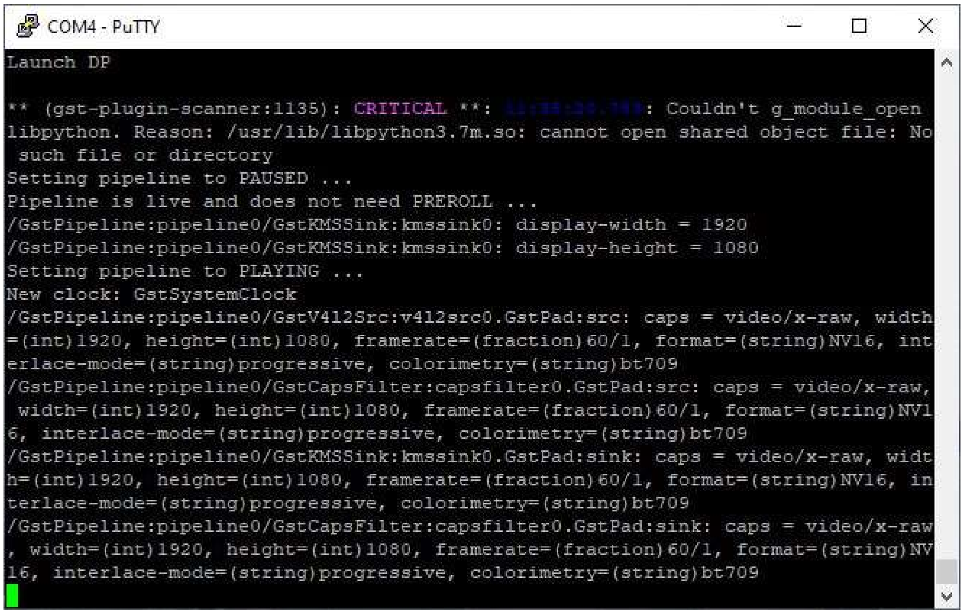
From the KV260 command prompt, execute the following commands. The camera must enumerate to /dev/media0, if there are other cameras connected, please check this.

8.1 cd /opt/xfuse/bin/

8.2 sudo ./phoenixcamstart.sh

It is possible to specify the frame rate as shown below. The resolution is set at 1920x1080, the frame rate can be 30 or 60. Default is 60.

8.3 sudo ./phoenixcamstart.sh 1920 1080 60 or sudo ./phoenixcamstart.sh 1920 1080 30

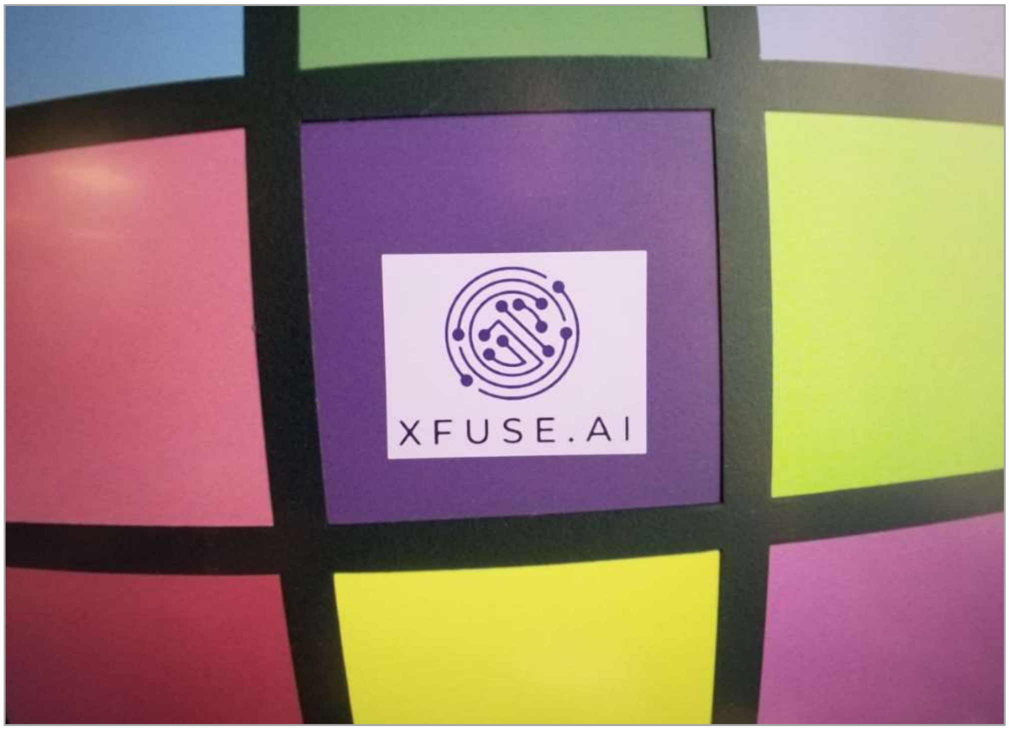


```
Launch DP

** (gst-plugin-scanner:1135): CRITICAL **: [illegible]: Couldn't g_module_open
libpython. Reason: /usr/lib/libpython3.7m.so: cannot open shared object file: No
 such file or directory
Setting pipeline to PAUSED ...
Pipeline is live and does not need PREROLL ...
/GstPipeline:pipeline0/GstKMSSink:kmssink0: display-width = 1920
/GstPipeline:pipeline0/GstKMSSink:kmssink0: display-height = 1080
Setting pipeline to PLAYING ...
New clock: GstSystemClock
/GstPipeline:pipeline0/GstV4l2Src:v4l2src0.GstPad:src: caps = video/x-raw, width
=(int)1920, height=(int)1080, framerate=(fraction)60/1, format=(string)NV16, int
erlace-mode=(string)progressive, colorimetry=(string)bt709
/GstPipeline:pipeline0/GstCapsFilter:capsfilter0.GstPad:src: caps = video/x-raw,
 width=(int)1920, height=(int)1080, framerate=(fraction)60/1, format=(string)NV1
6, interlace-mode=(string)progressive, colorimetry=(string)bt709
/GstPipeline:pipeline0/GstKMSSink:kmssink0.GstPad:sink: caps = video/x-raw, widt
h=(int)1920, height=(int)1080, framerate=(fraction)60/1, format=(string)NV16, in
terlace-mode=(string)progressive, colorimetry=(string)bt709
/GstPipeline:pipeline0/GstCapsFilter:capsfilter0.GstPad:sink: caps = video/x-raw
, width=(int)1920, height=(int)1080, framerate=(fraction)60/1, format=(string)NV
16, interlace-mode=(string)progressive, colorimetry=(string)bt709
```

The above messages should appear, and video will appear on the designated monitor. The Kria IP address in the terminal window should be visible.

Note this IP address for the next section.

The video on your display will appear with a Xfuse watermark in the center as shown below.



This version of the ISP is provided as a free fully functional unlimited trial with the embedded watermark. The watermark will be removed when a paid/licensed version of the ISP is installed.

The AR0239 IAS Sensor Module uses an M12 lens mount. The focus point of the lens can be adjusted by rotating the lens clockwise or counterclockwise until the subject is fully in focus. If the video on the screen seems out of focus when initially displayed, please adjust the lens until it comes in to focus.

Please be careful to keep the lens clean of fingerprints, dust, or dirt.

Wipe the lens with a clean, soft cloth if necessary.

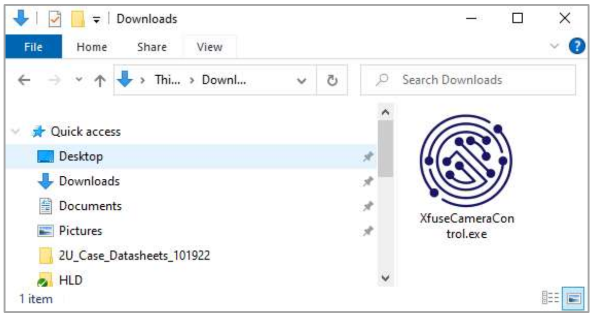
9. Connecting Kria SOM to the Xfuse Camera Control Software

Evaluation of the Kria SOM Phoenix ISP Application is supported by Xfuse's Camera Control Software package which is available for Windows platforms. This document does not provide an in-depth perspective on the Camera Control Software. Refer to the latest Camera Control Software User's Guide, which is included with the download, for additional information.

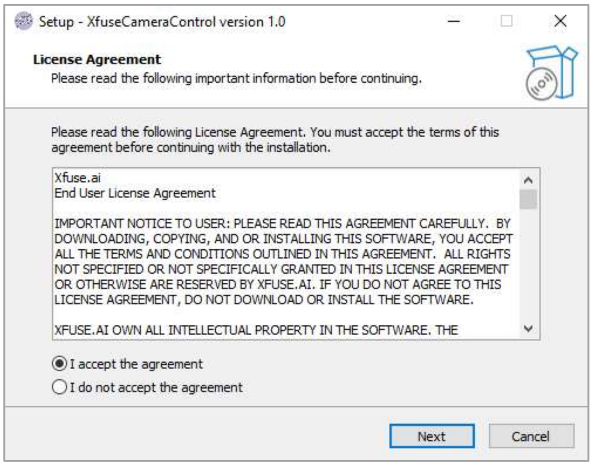
Important: At this point, it may be necessary to disable the firewall and/or VPN to ensure that the Xfuse Camera Control Software can communicate with the Kria SOM. Please check this prior to continuing.

Follow these steps to get started with the Xfuse Camera Control software:

1. Request the software from https://xfuse.ai/isptools/ downloads page. After entering a username and email address, look to receive an email containing a download link for the software
   - Please check appropriate junk email for the link if it does not send within 5 minutes
2. Download the software installer and launch the XfuseCameracontrol.exe



3. Follow the instructions in the installer and accept the End User License Agreement (EULA)


Setup - XfuseCameraControl version 1.0

License Agreement
Please read the following important information before continuing.

Please read the following License Agreement. You must accept the terms of this agreement before continuing with the installation.

Xfuse.ai
End User License Agreement

IMPORTANT NOTICE TO USER: PLEASE READ THIS AGREEMENT CAREFULLY. BY DOWNLOADING, COPYING, AND OR INSTALLING THIS SOFTWARE, YOU ACCEPT ALL THE TERMS AND CONDITIONS OUTLINED IN THIS AGREEMENT. ALL RIGHTS NOT SPECIFIED OR NOT SPECIFICALLY GRANTED IN THIS LICENSE AGREEMENT OR OTHERWISE ARE RESERVED BY XFUSE.AI. IF YOU DO NOT AGREE TO THIS LICENSE AGREEMENT, DO NOT DOWNLOAD OR INSTALL THE SOFTWARE.

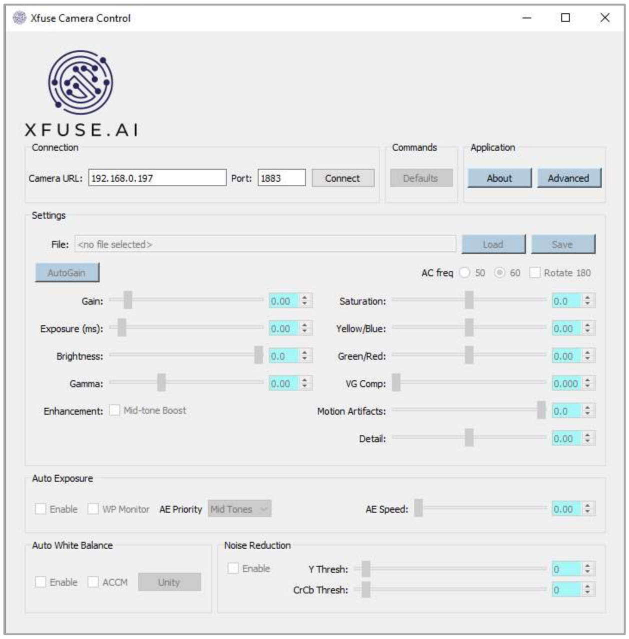
XFUSE.AI OWN ALL INTELLECTUAL PROPERTY IN THE SOFTWARE. THE

● I accept the agreement
○ I do not accept the agreement

4. After the installation is complete, Shortcuts to the application and user guide will be placed on the desktop
5. Open the Xfuse Quick Start Guide for future reference. It was extracted to Windows (C) > Program Files > Xfuse.AI Xfuse > Xfuse Camera Control > doc

6. To launch the software, double-click the installed desktop shortcut, select it from the Windows Programs Menu or go to Windows (C) > Program Files > Xfuse.AI > Xfuse Camera Control > XfuseCameraControl.exe

   Most items are grayed out until the application is connected. Enter the Kria SOM IP address in the Camera URL:Port window. Leave the port value as default. Then press connect.



Xfuse Camera Control

XFUSE.AI

Connection
Camera URL: 192.168.0.197 Port: 1883 Connect

Commands
Defaults

Application
About Advanced

Settings
File: <no file selected> Load Save
AutoGain
AC freq ○ 50 ● 60 □ Rotate 180

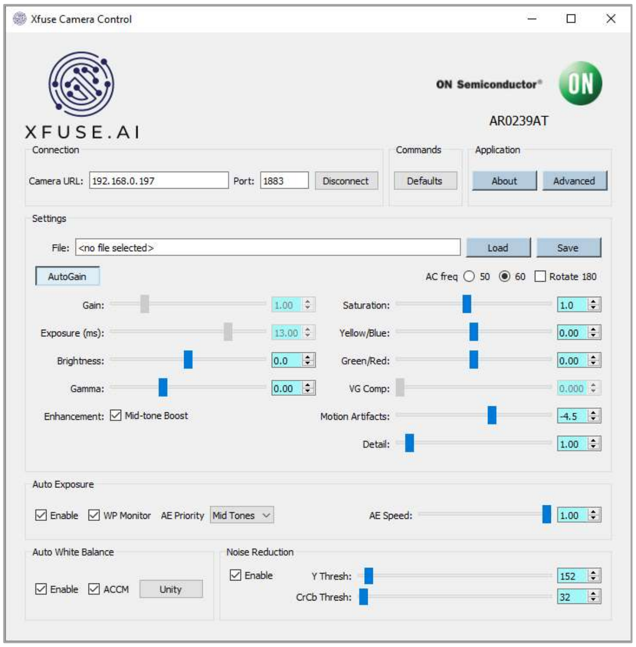
Gain: 0.00
Exposure (ms): 0.00
Brightness: 0.0
Gamma: 0.00
Enhancement: □ Mid-tone Boost
Saturation: 0.0
Yellow/Blue: 0.00
Green/Red: 0.00
VG Comp: 0.000
Motion Artifacts: 0.0
Detail: 0.00

Auto Exposure
□ Enable □ WP Monitor AE Priority Mid Tones
AE Speed: 0.00

Auto White Balance
□ Enable □ ACCM Unity

Noise Reduction
□ Enable
Y Thresh: 0
CrCb Thresh: 0

Once connected, the dialog boxes and sliders are now active. It is not possible to control some of camera features such as exposure, gain, and color correction.



Xfuse Camera Control

XFUSE.AI

ON Semiconductor®
AR0239AT

Connection
Camera URL: 192.168.0.197 Port: 1883 Disconnect

Commands
Defaults

Application
About Advanced

Settings
File: <no file selected> Load Save
AutoGain
AC freq ○ 50 ● 60 □ Rotate 180

Gain: 1.00
Exposure (ms): 13.00
Brightness: 0.0
Gamma: 0.00
Enhancement: ☑ Mid-tone Boost
Saturation: 1.0
Yellow/Blue: 0.00
Green/Red: 0.00
VG Comp: 0.000
Motion Artifacts: -4.5
Detail: 1.00

Auto Exposure
☑ Enable ☑ WP Monitor AE Priority Mid Tones
AE Speed: 1.00

Auto White Balance
☑ Enable ☑ ACCM Unity

Noise Reduction
☑ Enable
Y Thresh: 152
CrCb Thresh: 32

Please refer to the "Users Guide" for additional details on the use of the Xfuse Camera Control application.

10. Questions / Feedback

We value feedback and appreciate any input to make the system features and functions better. Please take a moment to share experiences, questions, and feedback with the Xfuse team via the Xilinx Kria Partner forum. Enjoy!

11. Next Steps

For pricing and licensing information, please contact Xfuse directly at info@xfuse.ai or inquire on our website at https://xfuse.ai/contact-us/.