[CAMERA USER'S MANUAL]

EMERGENT VISION TECHNOLOGIES INC
1.0.4
July 16, 2014

SUITE #239 - 552A CLARKE ROAD, COQUITLAM, V3J 0A3, B.C. CANADA
WWW.EMERGENTVISIONTEC.COM

Table of Contents

Contact

Emergent Vision Technologies Canada (Headquarters)

Suite #239 - 552A Clarke Road,

Coquitlam,B.C.

V3J 0A3

CANADA

info@emergentvisiontec.com

www.emergentvisiontec.com

Technical Support

info@emergentvisiontec.com

Legal

Life Support Applications

These products are not designed for use in life support systems.

Trademarks

All trademarks appearing in this document are protected by law.

Warranty

The information provided is supplied without any guarantees or warranty.

Copyright

All texts, pictures, files, and graphics are protected by copyright and other laws protecting intellectual property. It is not permitted to copy or modify them for and use.

Overview

This document is the User's Manual for Emergent Vision Technologies cameras.
It serves as a ground-up reference for getting started and working with our cameras.

The Hardware Installation chapter includes all required to install the Myricom PCIe NIC, cables, camera and camera power supply. It also covers the interfaces associated with the camera such as LEDs, and the SFP+ 10G interface. GPIO is postponed until a later chapter. Installation of the NIC driver is also covered.

The Software Installation chapter includes all required to install the Emergent Vision Technologies software which includes eCapture and eSDK. Configuration of the NIC is also covered. All are covered on both Windows and Linux.

The eCapture chapter allows the user to get imaging quickly with the camera powered up and configured as described in previous chapters. eCapture is Emergent's GenICam based viewer software which also provides built-in functionality such as IP configuration and remote firmware upgrades which are described in later chapters.

The IP Configurator chapter describes the IP configuration functionality within eCapture which allows the user to set volatile and non-volatile IP address and associated subnet mask and default gateway for the selected camera.

The Firmware Upgrades chapter describes the firmware upgrade functionality within eCapture which allows the user to perform remote upgrades on the camera with Emergent provided firmware to take advantage of feature adds and other updates.

The eSDK chapter includes all required for the programmer to get up and running using the eSDK and this involves compiling and running the provided examples.

The camera interfaces chapter describes all interfaces of the camera and goes into detail for the GPIO interface for those connecting custom hardware to the GPIO connector.

Hardware Installation

Installing Myricom 10G NIC

As with the installation of any PCIe card, the PC must be powered down and the necessary ESD precautions must be observed. Both the dual port and single port 10G NICs take a x8 PCIe slot (they will fit into a x16 slot also) but the chosen slot MUST be wired for at least x8. Some PCIe slots may be x16 or x8 in mechanical size but wired as x4. Please check your motherboard specifications.

Under Windows, the installation of the Myricom MVA driver is quite straightforward. Simply run the provided .msi installation file, accept the license agreement, choose the installation directory, and the installation will complete and install MVA to all Myricom NICs in the PC. This is covered as part of the install script on the provided software CD.

Under Linux, using 2.x.x Emergent software the Myricom software is installed as part of the eCapture/eSDK installation. Please see the Software Installation chapter and skip the steps just below. Note, all models are supported and will be continued to be supported under 2.x.x software.

Under Linux, using earlier than 2.x.x software, in order to support multiple distributions and kernels things are handled a bit differently. On the CD, there will be a directory named "Linux" and this directory will contain only two files: one of which is the tarball of the Myricom driver and will be named "myri_mva.tgz". To install the Myricom driver we type the following in a terminal (Examples are all for Ubuntu Linux distribution 12.04 LTS):

```
(After copying the myri_mva.tgz file to the /opt directory we unzip there)
#cd /opt
#sudo tar xzf myri_mva.tgz

(This rebuilds the Myricom driver for your kernel version)
#cd /opt/myri_mva
#sudo ./sbin/rebuild.sh
```

NOTE: if the Kernel version is ever updated then the above step will need to be re-run.

Under Linux we now must also tell the system to load the driver at system start-up. We do this by adding the following line to rc.local file in the /etc directory:

```
/opt/myri_mva/sbin/myri_start_stop start
```

We can now run a utility to show our properly configured Myricom NICs

```
# sudo /opt/myri_mva/bin/myri_nic_info

# Serial MAC               ProductCode      Driver   Version      License
0 432237 00:60:dd:45:00:8c 10G-PCIE2-8C2-2S myri_mva 0.3.0.51214 Valid
1 432237 00:60:dd:45:00:8d 10G-PCIE2-8C2-2S myri_mva 0.3.0.51214 Valid
```

Installing SFP+ Direct Attach Cables

SFP+ is a 10G connector standard and is keyed to prevent incorrect insertion. Direct Attach cables are one piece cables that plug into both the cameras and NIC to complete the link.

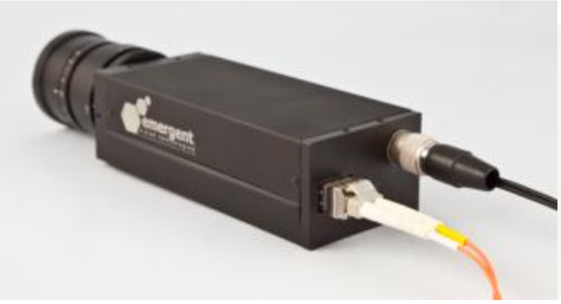
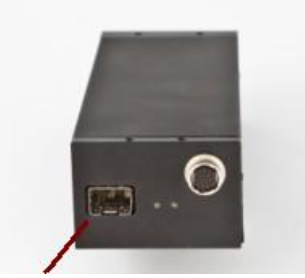
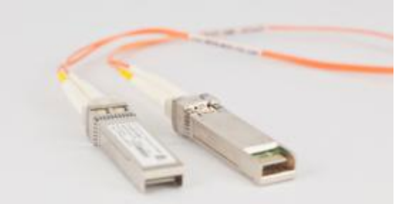
Simply insert each end of the SFP+ cable into both the NIC SFP+ port and the camera SFP+ port. With the cable completely inserted there should be a clicking sound to indicate it is latched.

Removing the cables is accomplished by pulling on the provided tab which releases the latching mechanism on the underside of the connector.

Installing SFP+ Fiber Transceivers and Cables

Insert the LC fiber into the optical transceiver and then install the completed SFP+ assembly into the NIC and camera as shown in the previous section. Fiber cables thus consist of three parts: 2 x Fiber Transceivers and 1x LC-LC fiber cable.

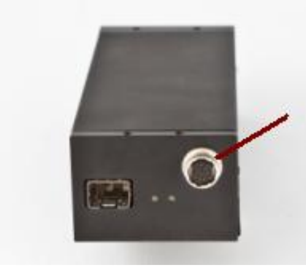
Removing the assembly from either end is accomplished by first removing the LC-LC fiber by pushing down on the release latch on the top side of the assembly. Then the optical transceiver can be removed by pulling the latch at the top of the transceiver outward and down - the whole SFP+ transceiver module then simply slides out of the SFP+ cage.

Connecting Power

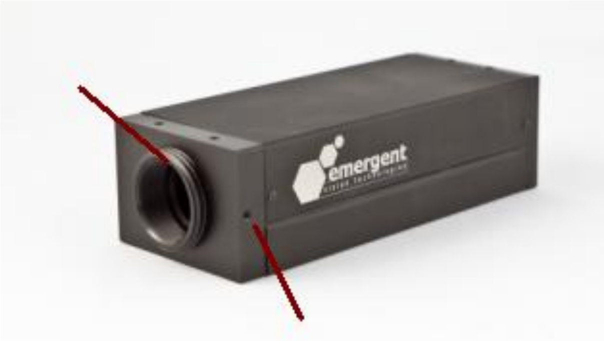
Only use Emergent approved power supplies with Emergent cameras. Note that while other cameras use the same connector they do not necessarily use the same pinout. Also, FCC/CE testing was performed using the Emergent provided supplies and some power supplies are better at reducing power line transients than other poorly designed supplies. A poorly designed non-approved supply can, under the right influence, damage the camera and will not be covered by the warranty.

Simply insert the HR12 connector into the camera HR12 port. This is a latching connector and requires pulling back on the outer shell of the connector and rotating around until seated. Emergent GPIO wiring harnesses are installed in the same manner.



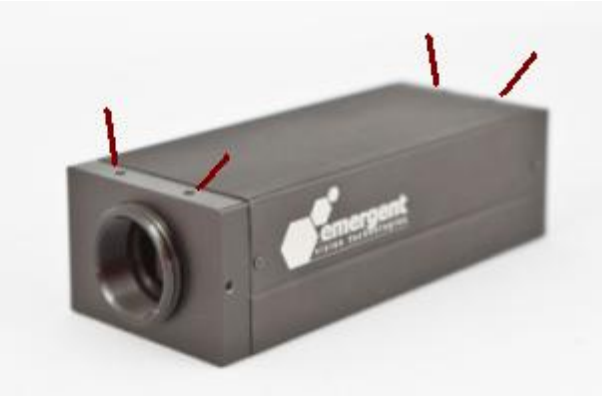
Adjusting Backflange Distance

Standard practices apply to adjusting back flange distance but Emergent cameras are equipped with user adjustable distance. As shown, there are entry points on the side of the front plate to a hex key screw which when loosened allows the camera lens barrel to screw in and out. With the barrel in the accepted position, simply gently tighten the hex screws to complete the operation. Standard C mount lenses then screw into the end of the barrel. The process is the same for the M42 and M52 F-Mount adapters for the larger format models.



Tripod Mount

A tripod mount and screws is provided with every camera and can easily be mounted on top or bottom of the camera using the screw locations shown. A cutout is made in the mount for easy viewing of the camera label.



GPIO

Please see the Camera Interfaces chapter in this manual.

Software Installation

Installing Software(Windows)

The software installation is quite straightforward. Place the provided software CD in the computer and the setup will launch automatically and install the Myricom software and drivers followed by the Emergent eSDK and eCapture software.

For 2.x.x software versions, the Microsoft 2012 VC++ runtime is required but .NET is not required as with earlier versions of the software. See below regarding the runtime. For 2.x.x eSDK examples are built with Microsoft Visual Studio Express 2012. Note, all models are supported and will be continued to be supported under 2.x.x software. Note reboot may be required to have environment variables take effect.

For earlier than 2.x.x the software requires Microsoft .NET 4.0 and if this is not already installed then this is also available on the relevant software CD for convenience.

For earlier than 2.x.x the software also requires the Microsoft VC++ runtime environment which is also available on the software CD for convenience. The install file is vcredist_x64.exe.

For earlier than 2.x.x, for building and running the eSDK examples, install Microsoft Visual Studio Express 2010 followed by the Microsoft SDK 7.1(allows 64 bit support). Web installation links are also available on the applicable software CD for convenience. Note: MSVS 2010 includes .NET 4.0.

Installing Software(Linux)

For 2.x.x software versions, the process has been simplified. You just need to copy the install .tgz file to a directory on the Linux machine an then run from that directory:

```
# tar xzf emergent_camera_2.0.1.20003.x86_64.tgz  (or provided .tgz file name)
# ./install_eSDK
```

This installs Emergent software and Myricom's MVA driver. Reboot may be required. Internet connectivity is required for the installer to fetch all the dependencies.

To uninstall:

```
# ./uninstall_eSDK
```

Note, all models are supported and will be continued to be supported under 2.x.x software. Software is installed to /opt/EVT directory.

For earlier than 2.x.x, under Linux, in order to support multiple distributions and kernels things are handled a bit differently. On the CD, there will be a directory named "Linux" and this directory will contain only two files: one of which is the driver and the other is the Emergent eSDK tarball. We install the eSDK as follows:

```
# cd /opt
# sudo tar xzf EVT_SDK.tgz
```

The eSDK is now installed to /opt/EVT

We now will setup the required environment.
We do this by adding the following lines to "environment" text file in the /etc directory:

```
EMERGENT_DIR=/opt/EVT
GENICAM_ROOT_V2_1=/opt/EVT/eSDK/genicam
```

...and further adding the following line at the end of a text file called .bashrc file in the Home directory(get there by typing cd):

```
export LD_LIBRARY_PATH=/opt/EVT/eSDK/lib/i86_x64:/opt/EVT/eSDK/genicam/bin/Linux64_x64
```

...and lastly we need to install some libraries required by Opencv which is used by our software(need an internet connection for this):

```
# sudo apt-get install libavcodec-dev libavformat-dev libswscale-dev libv4l-dev
```

Ubuntu Notes

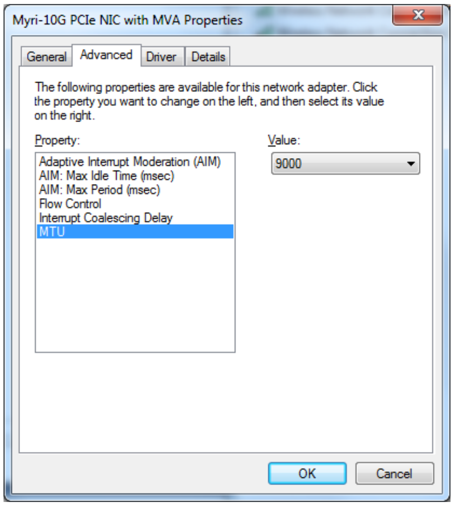
Emergent's Linux eSDK was primarily tested under Ubuntu 12.04 LTS and users may download this here:
http://www.ubuntu.com

For step by step Ubuntu installation instructions see the Ubuntu documentation at:
https://help.ubuntu.com/12.04/installation-guide/index.html

For 2.x.x software, Ubuntu 12.04 LTS is only supported and has been tested with 12.04.02 and earlier. Kernel conflicts arise with the Myricom driver for versions later than 12.04.02.

Configuring NIC

The NIC should be correctly configured if the instructions in the section on Installing the NIC are followed. MTU 9000 should be set by default as below.



Myri-10G PCIe NIC with MVA Properties

The following properties are available for this network adapter. Click the property you want to change on the left, and then select its value on the right.

Property:
Adaptive Interrupt Moderation (AIM)
AIM: Max Idle Time (msec)
AIM: Max Period (msec)
Flow Control
Interrupt Coalescing Delay
MTU

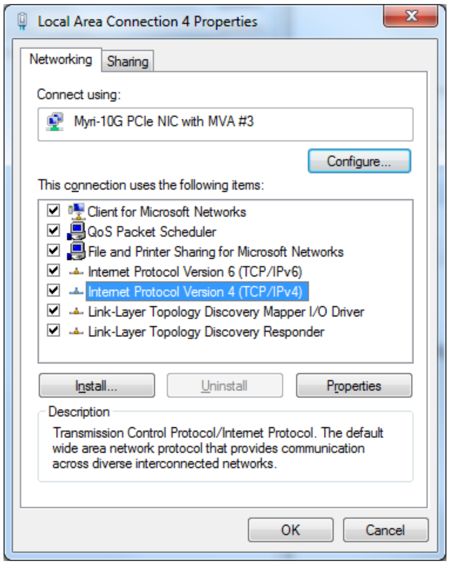
Value: 9000

We naturally must configure the IP address of the NIC and coordinate this with the camera and other devices with respect to subnets, IP address conflicts, etc. For simplicity, our example shows static IP address configuration. We thus might setup our camera with an IP address of 192.168.2.21 which would put the camera on the same subnet as the NIC and also avoid an IP address conflict with the host IP of 192.168.2.20. This is handled very much the same through the Network Connections dialogs when working under Ubuntu Linux. A common mistake is not setting up the IP address and subnet mask correctly and in some cases this results in 0 cameras being found by the software. Other times 0 cameras are found as a result of a firewall which does not pass certain IP addresses.

An important point is that the default camera IP address is a static 192.168.1.69 so, provided no IP address conflicts with this or the host IP address, then a suitable host IP address and subnet mask would be:

IP:192.168.1.20
Subnet Mask: 255.255.255.0

with all being on the same 192.168.1.x subnet.



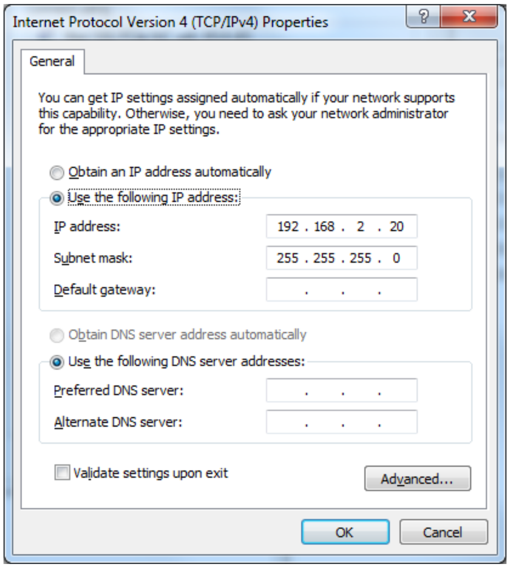
Local Area Connection 4 Properties

Connect using:
Myri-10G PCIe NIC with MVA #3

This connection uses the following items:

- ☑ Client for Microsoft Networks
- ☑ QoS Packet Scheduler
- ☑ File and Printer Sharing for Microsoft Networks
- ☑ Internet Protocol Version 6 (TCP/IPv6)
- ☑ Internet Protocol Version 4 (TCP/IPv4)
- ☑ Link-Layer Topology Discovery Mapper I/O Driver
- ☑ Link-Layer Topology Discovery Responder

Description
Transmission Control Protocol/Internet Protocol. The default wide area network protocol that provides communication across diverse interconnected networks.



Internet Protocol Version 4 (TCP/IPv4) Properties

You can get IP settings assigned automatically if your network supports this capability. Otherwise, you need to ask your network administrator for the appropriate IP settings.

○ Obtain an IP address automatically
● Use the following IP address:

IP address: 192 . 168 . 2 . 20
Subnet mask: 255 . 255 . 255 . 0
Default gateway: . . .

○ Obtain DNS server address automatically
● Use the following DNS server addresses:

Preferred DNS server: . . .
Alternate DNS server: . . .

□ Validate settings upon exit

eCapture

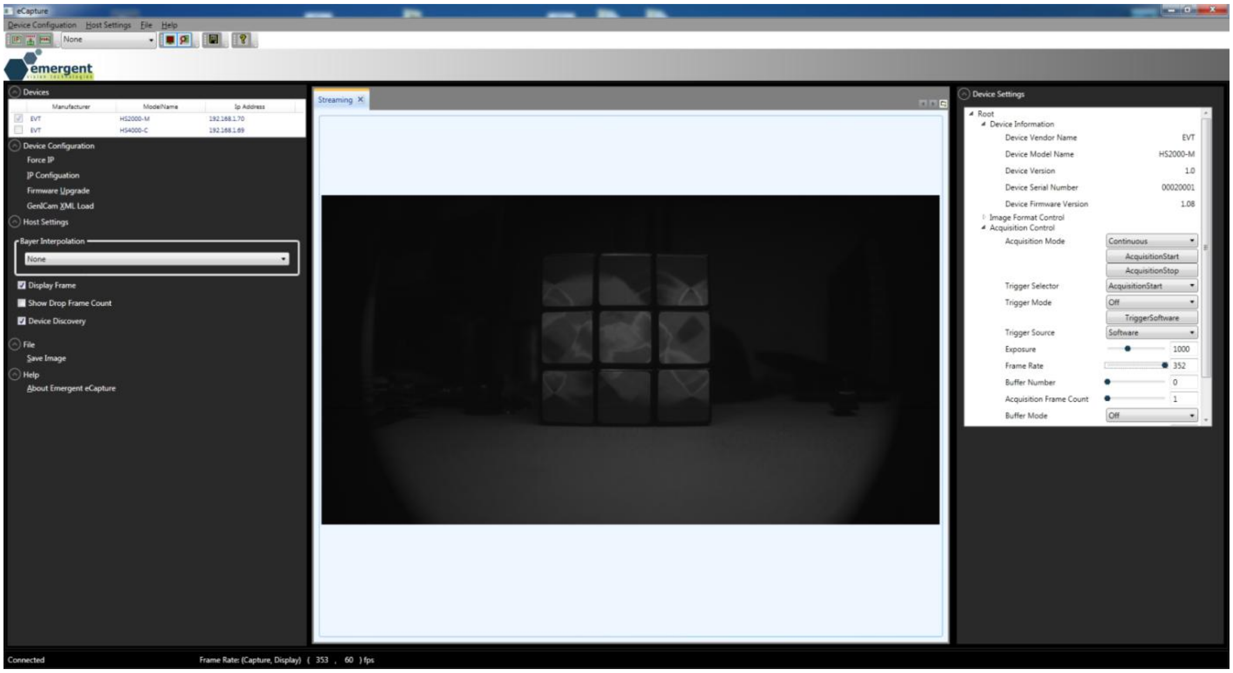
Start eCapture

Plug in the camera and power up and configure as described in the previous chapters.

Start the eCapture application. On the left hand side of the main window the "Devices" view will show all Emergent cameras found during the discovery process. Clicking on a camera in this list gains control of this camera which also involves retrieving the camera GenICam XML feature file and filling in the GenICam feature tree "Device Settings" view on the right side of the main window. Clicking on the same camera disconnects that camera. Clicking on another camera disconnects any other camera already connected and connects the new camera. eCapture only connects to one camera at a time. The eSDK can be used to connect and stream from multiple cameras.

Under Linux we would simply launch eCapture by typing in the following in a terminal:

```
# /opt/EVT/eCapture/bin/eCapture
```



	Manufacturer	ModelName	Ip Address
☑	EVT	HS2000-M	192.168.1.70
☐	EVT	HS4000-C	192.168.1.69

☑ Display Frame
☐ Show Drop Frame Count
☑ Device Discovery

- Root
  - Device Information
    - Device Vendor Name: EVT
    - Device Model Name: HS2000-M
    - Device Version: 1.0
    - Device Serial Number: 00020001
    - Device Firmware Version: 1.08
  - Image Format Control
  - Acquisition Control
    - Acquisition Mode: Continuous
    - Trigger Selector: AcquisitionStart
    - Trigger Mode: Off
    - Trigger Source: Software
    - Exposure: 1000
    - Frame Rate: 352
    - Buffer Number: 0
    - Acquisition Frame Count: 1
    - Buffer Mode: Off

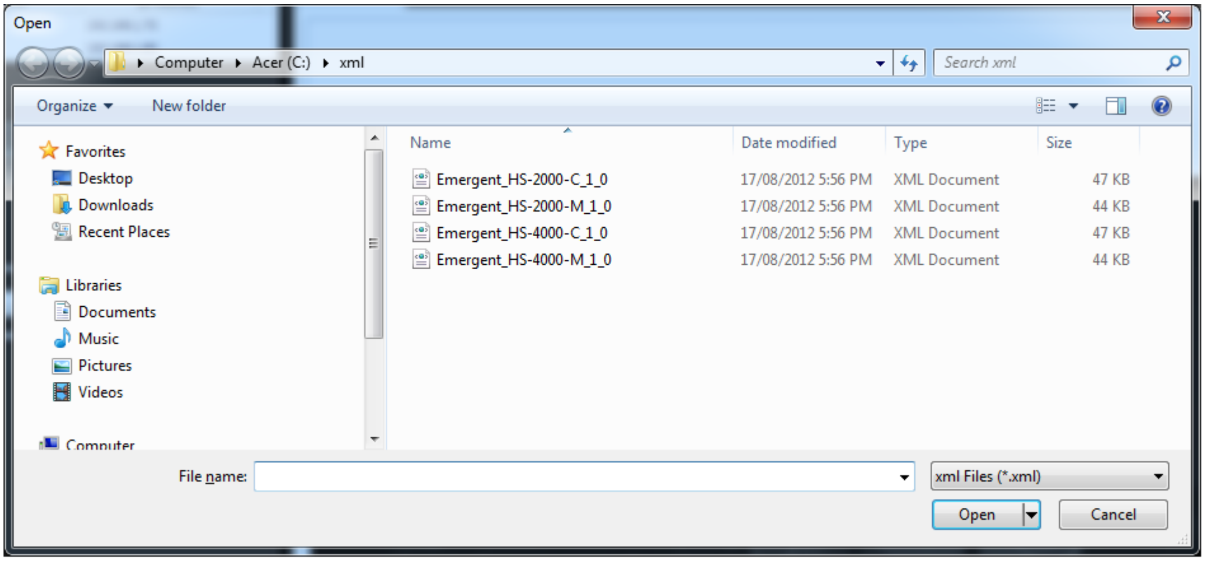
Connected   Frame Rate: (Capture, Display) (353 , 60) fps

XML Updates

This step can usually be skipped.

If there is an updated XML file for your camera then use the “Device Configuration” view “GenICam XML Load” feature to locate and use this updated file. Note that this is a volatile operation and the updated XML file is not programmed to the camera non-volatile memory so after power cycling this procedure will need to be repeated. Writing the XML file to the camera is supported but please contact support for procedure and approved XML files. Depending on the camera and new or custom features, this step should be completed before continuing. The ability to update the XML is also exposed through the eSDK.

However, Emergent cameras always ship with the latest well tested firmware and XML file in camera flash memory so updates are not usually required unless new features are added or custom firmware provided.



Computer ▸ Acer (C:) ▸ xml

Name	Date modified	Type	Size
Emergent_HS-2000-C_1_0	17/08/2012 5:56 PM	XML Document	47 KB
Emergent_HS-2000-M_1_0	17/08/2012 5:56 PM	XML Document	44 KB
Emergent_HS-4000-C_1_0	17/08/2012 5:56 PM	XML Document	47 KB
Emergent_HS-4000-M_1_0	17/08/2012 5:56 PM	XML Document	44 KB

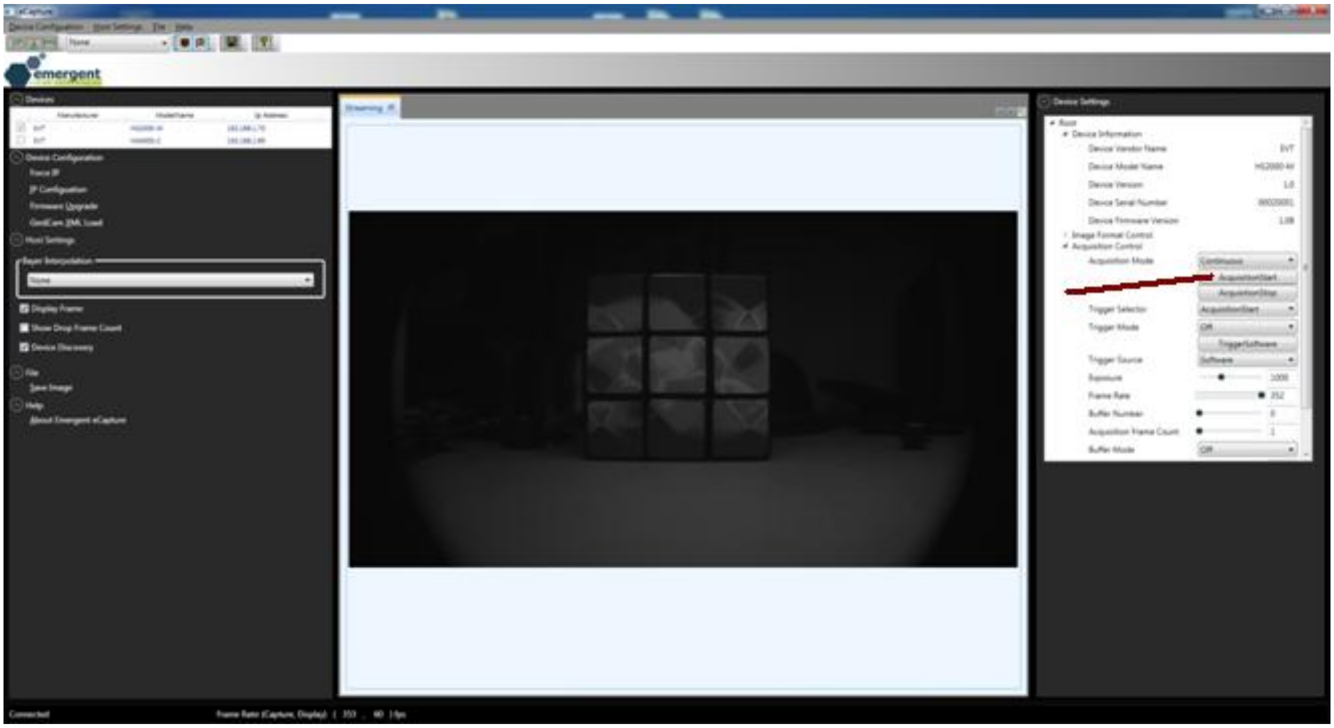
File name:

xml Files (*.xml)

Start Imaging

With above two steps completed, expand the GenICam feature tree in the ”Device Settings” view and expand the “AcquisitionControl” node. Then click the “Acquisition Start” button to start imaging.

In this same view, one can exercise the full functionality of the camera and make changes to camera parameters such as area of interest(AOI), exposure, gain, offset, pixel format and also to utilize special camera triggering modes as described in the EVT Camera Attributes documentation. See also GPIO in the Camera Interfaces chapter of this manual for external triggering and other IO activities using GPIO.



Manufacturer	ModelName	Ip Address
EVT	HS2000-M	192.168.1.70
EVT	HA4000-C	192.168.1.49

Bayer Interpolation: None

☑ Display Frame
□ Show Drop Frame Count
☑ Device Discovery

Device Settings

- Root
  - Device Information
    - Device Vendor Name: EVT
    - Device Model Name: HS2000-M
    - Device Version: 1.0
    - Device Serial Number: 00020001
    - Device Firmware Version: 1.08
  - Image Format Control
  - Acquisition Control
    - Acquisition Mode: Continuous
    - AcquisitionStart
    - AcquisitionStop
    - Trigger Selector: AcquisitionStart
    - Trigger Mode: Off
    - TriggerSoftware
    - Trigger Source: Software
    - Exposure: 1000
    - Frame Rate: 252
    - Buffer Number: 0
    - Acquisition Frame Count: 1
    - Buffer Mode: Off

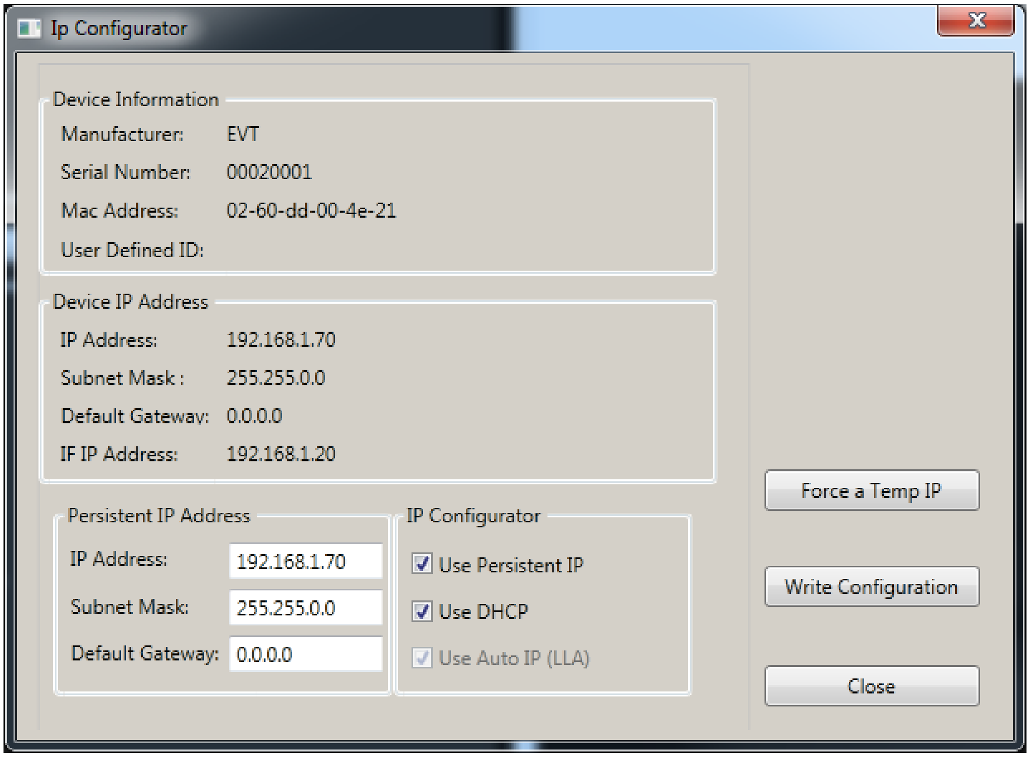
Connected

Frame Rate (Capture, Display): (353 , 60) fps

IP Configurator

Starting IP Configurator

Start the eCapture application as previously described and select a camera to control through the "Devices" view. Then navigate to "Device Configuration" and "IP Configurator."



Ip Configurator

Device Information

Manufacturer: EVT

Serial Number: 00020001

Mac Address: 02-60-dd-00-4e-21

User Defined ID:

Device IP Address

IP Address: 192.168.1.70

Subnet Mask : 255.255.0.0

Default Gateway: 0.0.0.0

IF IP Address: 192.168.1.20

Persistent IP Address

IP Address: 192.168.1.70

Subnet Mask: 255.255.0.0

Default Gateway: 0.0.0.0

IP Configurator

☑ Use Persistent IP

☑ Use DHCP

☑ Use Auto IP (LLA)

Using IP Configurator

Selecting “IP Configurator” as detailed above presents a dialog with the following:

Current IP Settings:

Current IP Address
Current Subnet Mask
Current Default Gateway

Current Persistent IP Settings:

Persistent IP Address
Persistent Subnet Mask
Persistent Default Gateway
Persistent Configuration Method(Static/Fixed, DHCP, LLA/AutoIP)

Options available to the user are as follows:

Write Configuration: to write new persistent IP settings to the camera.

Force IP: to write and force new IP configuration data into the camera.

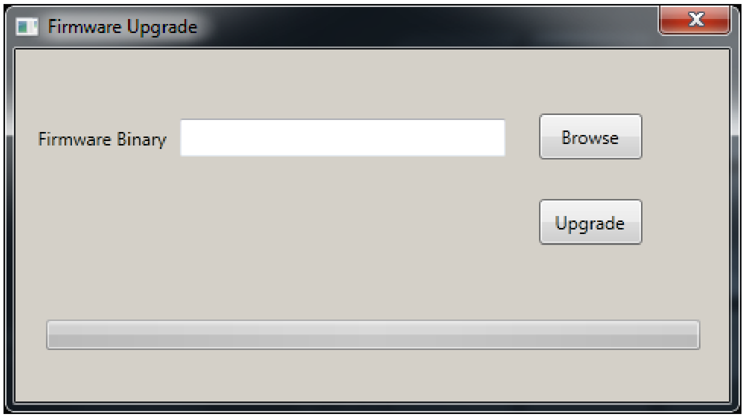
For both the above, the new settings change will cause the camera to be disconnected, re-configured and then re-discovered. It will then show up again in the “Devices” view under the new IP configuration.

The default factory configuration is with DHCP enabled with LLA as a fallback should no DHCP server be available. The user can enable and configure static IP as described above. Some firewalls prevent LLA so may need to turn off the firewall temporarily to perhaps set a static IP address that the firewall can be programmed to accept.

Firmware Upgrades

Starting Firmware Upgrade

Start the eCapture application as previously described and select a camera to control through the "Devices" view. Then navigate to "Device Configuration" and "Firmware Upgrade."



Using Firmware Upgrade

For updated firmware please contact technical support.

Assuming a compatible firmware upgrade file is available, simply select the file for upgrading through the "Firmware Upgrade" dialog and then click the Upgrade button to begin the short upgrade process. The new firmware will take effect upon the next power cycle.

Intelligence is built in to revert back to the factory firmware in the event of a power outage or other failure. Nonetheless, precautions should be taken to not power down the camera, overload the network, or otherwise while the short upgrade is in progress.

Depending on the nature of the firmware upgrade it may also be necessary to obtain an updated XML feature file to be used as previously described.

Emergent cameras always ship with the latest well tested firmware and XML file in camera flash so updates are not usually required unless new features are added or custom work performed.

eSDK

Introduction

Included as part of the eSDK installation are a number of examples for exercising the camera functionality. Under Windows, these examples have been created, built and run with Microsoft Visual Studio Express 2010 and the examples are pre-configured projects for this build environment. Building the examples also requires installing a Microsoft SDK designed for 64-bit designs. The examples were tested with Microsoft SDK v7.1.

Building Examples

Within VS Express, use Open Project and navigate to the Examples directory within the install directory of the eSDK installation and then into the desired example directory(ie. EVT_DeviceInformation) and select the VS solution file(ie. EVT_DeviceInformation.sln).

With the project now open select Build to build the project. The short process should output 0 Errors and 0 Warnings to the build console.

Under Linux a fresh install of Ubuntu 12.04 LTS would require the g++ tools which would be installed as part of 2.x.x software installations.
We install these with one terminal command (internet connection required):

```
#sudo apt-get install build-essential
```

Under Linux we would build an example in a terminal as follows:

```
# cd /opt/EVT/eSDK/Examples/EVT_DeviceInformation/
# make
```

Running Examples

Open a DOS window and cd to the project Debug directory and then run the newly built program by simply typing: EVT_DeviceInformation

The output will look similar to the following:

```
C:\.....\Debug>EVT_DeviceInformation
      -------------------------------------
      DeviceInformation : Example program
      -------------------------------------
      Open Camera:            Camera Opened

      DeviceVendorName:       EVT
      DeviceModelName:        HS2000-M
      DeviceVersion:          1.0
      DeviceSerialNumber:     00010000
      DeviceFirmwareVersion:  1.20

      Close Camera:           Camera Closed
```

Or under Linux, the output will look similar to the following:

```
# cd /opt/EVT/eSDK/Examples/EVT_DeviceInformation/
# ./out/EVT_DeviceInformation
      -------------------------------------
      DeviceInformation : Example program
      -------------------------------------
      Open Camera:            Camera Opened

      DeviceVendorName:       EVT
      DeviceModelName:        HS2000-M
      DeviceVersion:          1.0
      DeviceSerialNumber:     00010000
      DeviceFirmwareVersion:  1.20

      Close Camera:           Camera Closed
```

Camera Interfaces

SFP+

The SFP+ port is a standard port for 10Gb Ethernet applications.

SFP+ stands for “Enhanced small form-factor pluggable.”

While SFP+ supports hot-plug, Emergent at this point do not support this as this feature is more applicable to telecom and less to machine vision applications.

SFP+ has some significant advantages over other Ethernet cabling options including a variety of cabling options including “Direct Attach” which is a copper base solution for reliable transmission up to 10m and Fiber Based Transceiver solutions for significantly longer transmission distances only limited by the available fiber transceiver technologies.

Power is not provided over these cables and instead enters through the GPIO connector.

LEDs

Two LEDs are present on the back of the camera and their functionality is as follows:

Green LED: Simply indicates that the camera is powered and active.
This LED will always be steady on while the camera is powered.

Yellow LED: Indicates Ethernet Rx and Tx activity.

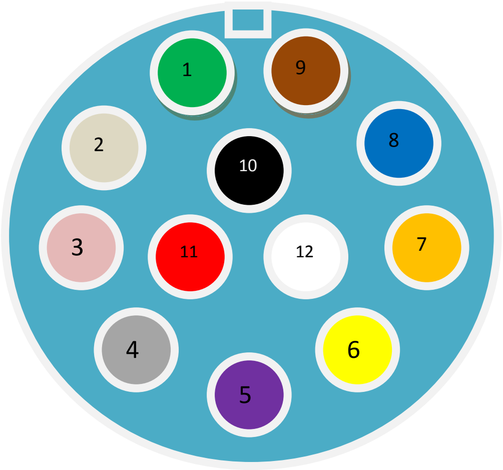
GPIO

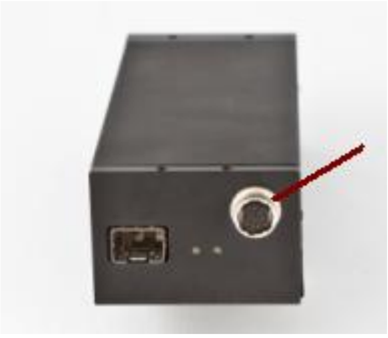
This port is used for carrying the GPIO signals as well as 12V power and ground signals from the Emergent supplied power supply. Only use Emergent supplied or approved power supply for connecting to the camera otherwise permanent damage could occur.

The pin-out of this port is as follows:

1	GND	Ground
2	GND	Ground
3	GPO_0	General Purpose Output
4	GPO_1	General Purpose Output
5	ISO_GND	Isolated Ground
6	GPO_2	General Purpose Output
7	GPO_3	General Purpose Output
8	VEXT	External 12V Power
9	VEXT	External 12V Power
10	NC	No Connect – not connected internally.
11	GPI_4	General Purpose Input
12	GPI_5	General Purpose Input

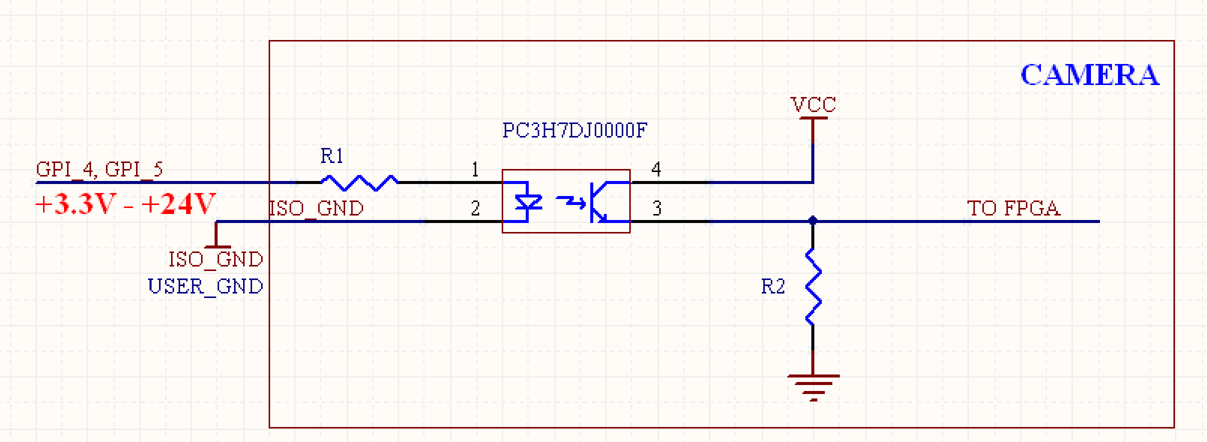
Looking into the back of the camera has the pins numbered and colored as follows:





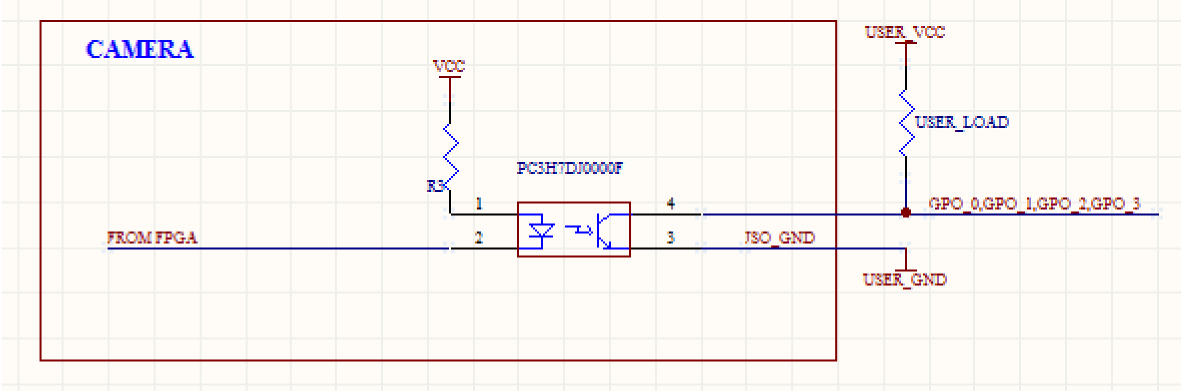
The schematic of the Camera IO ports is as follows:

The input ports GPI_4 and GPI_5 are as follows:



R1 = 3K, R2 =2K:

The output ports GPO_0, GPO_1, GPO_2, and GPO_3 are as follows:

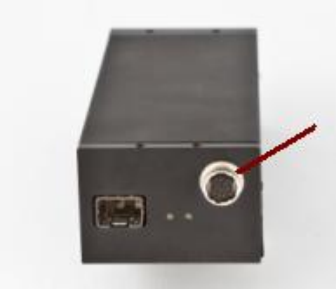
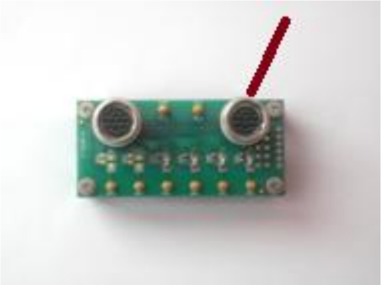


R3 = 200R.

R(USER_LOAD) is chosen depending on USER_VCC with the 10mA limit of the interface. So, for a 24V USER_VCC, R(USER_LOAD) = 24V/10mA = 2.4K (240mW package).

Power Adapter

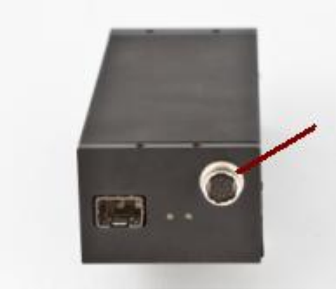
This accessory is used when no GPIO is needed and thus simply powers the camera.
It can also be used with the GPIO Breakout PCB.
Plug the Hirose connector into the camera Hirose port or into the PWR port of the GPIO Breakout PCB

Part number: PWR

Power Adapter w/GPIO Breakout

The breakout wires are color coded and colors and signal descriptions are documented at the beginning of this section. Pins 2 and 8 are not provided but 12V VEXT and GND are provided through pins 1 and 9 for external use.
Plug the Hirose connector into the camera Hirose port.



Part number: PWR_BREAK

GPIO Breakout Cable

The breakout wires are color coded and colors are documented at the beginning of this section. Note that the power supply must still be Emergent approved.
Plug the Hirose connector into the camera Hirose port.

Part number: GPIO_BREAK

GPIO Triggering Options

External hardware triggering can be implemented in a few different ways using Emergent camera products.

The system designer can provide an external signal (called EXT TRIG in the timing diagrams below) and this signal can be connected for example to the GPIO signal GPI_4 to fully control and synchronize the camera exposure to some external process.

For the timing diagram below and using GPI_4 for this EXT TRIG signal we would set the camera registers as follows:

GPI_Start_Exp_Mode = GPI_4 (enum)
GPI_Start_Exp_Event = RisingEdge (enum)

GPI_End_Exp_Mode = GPI_4 (enum)
GPI_End_Exp_Event = FallingEdge (enum)

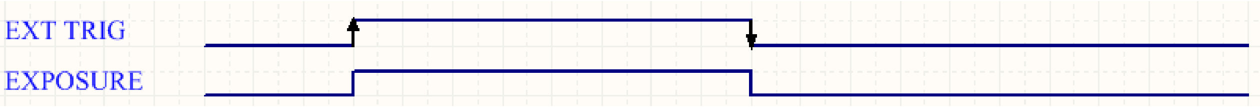


Figure 1

For the next timing diagram, a user may wish to have an exposure started some fixed time period after the EXT TRIG signal. In this case, we would additionally set the Trigger_Delay register as follows:

Trigger_Delay = 100 (measured in us)

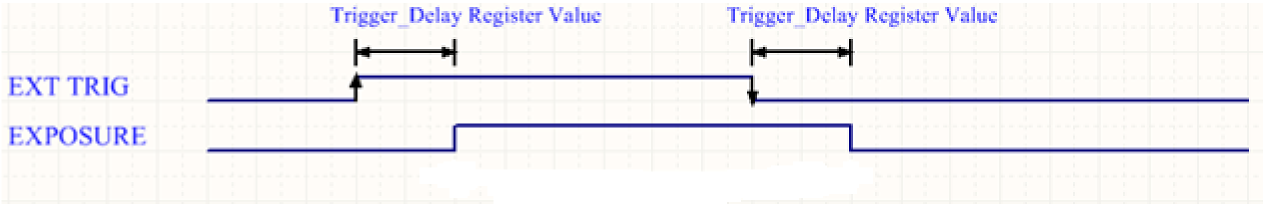


Figure 2

In the next timing diagram, the user may wish to use the internally generated exposure counter but have the beginning of the exposure triggered by an external signal. For this, we would set up the registers as follows:

GPI_Start_Exp_Mode = GPI_4 (enum)
GPI_Start_Exp_Event = RisingEdge (enum)

GPI_End_Exp_Mode = Internal (enum)

And, again, this mode can utilize the Trigger_Delay feature if needed to accomplish the timing also shown in this figure:

Trigger_Delay = 100 (measured in us)

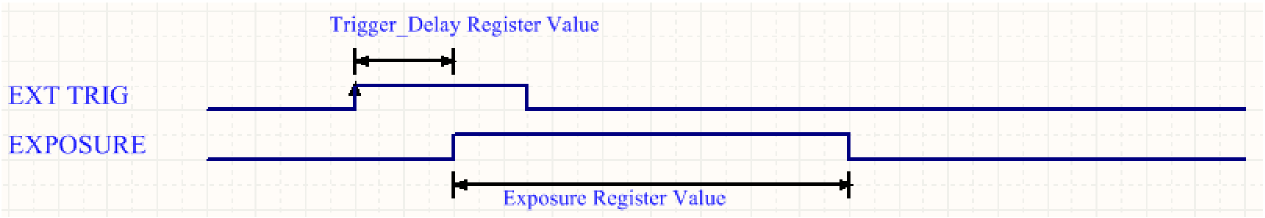


Figure 3

Some camera systems may wish to synchronize multiple cameras and yet have no external triggering signal to work with. The Test Generator feature can be used to output from a camera GPIO output port a programmable signal in both duty cycle(set high time) and period. For this, the registers are set as follows(for example to use GPO_0):

GPO_0_Mode = Test_Generator(enum)
GPO_0_Polarity = TRUE (Bool) (Positive or negative)
TG_Frame_Time = 16000 (us) (ie. or 62.5 fps)
TG_High_Time = 1000 (us) (ie. to set exposure of 1ms)

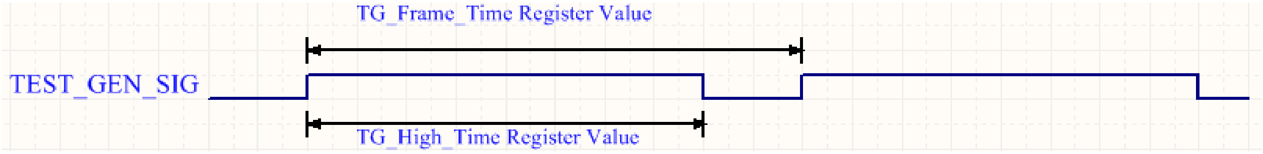


Figure 4

This signal can feed other GPI_n input triggering signals including its own GPI_n signal.

Other options of GPO usage are as follows and can be used to synchronize external system components to a camera exposing in Continuous (non-external trigger mode).

Each GPO_n signal can be programmed with any of the following options:

Exposure: To alert the system that the camera is exposing.

Readout: To alert the system that the camera is reading out data from the ccd.

GPO: With this option, the GPO_n signal will be set by the GPO_n_Polarity signal.

Test_Generator: As described above.

Document History

Version	Date	Description
1.01	29 May 2012	Initial Version
1.02	3 Mar 2013	Adding Linux Details
1.03	10 Dec 2013	Clarified IP configuration, added PCIe slot requirements.
1.04	16 July 2014	Added 2.x.x software notes