

APPLICATION NOTE

IEEE 1588 Precision Time Protocol Prosilica GT, GC, Manta, and Mako G cameras

 V1.3.0
2021-03-18

Supported models

This application note is applicable to the following models:

Camera family	Supported model series	Firmware version
Prosilica GT	GT1290, GT1380, GT1600, GT1660, GT1910, GT1920, GT1930, GT1930L, GT2000, GT2050, GT2300, GT2450, GT2750, GT3300, GT3400, GT4090, GT4400, GT4096, GT4905, GT4907, GT5120, GT5400, GT6400, GT6600	≥ 01.54.11026
Prosilica GC ¹	GC660, GC1290, GC1380H, GC1600H, GC2450	≥ 01.54.16528
Manta	G-040, G-158, G-223, G-235, G-282, G-283, G-319, G-419, G-505, G-507, G-895, G-917, G-1236, G-1620, G-2040, G-2460	≥ 01.54.11026
Mako G	G-040, G-158, G-234, G-319, G-507, G-508	≥ 01.54.21000
Mako G	G-030, G-223, G-419	≥ 01.54.26005

¹ Hardware revision D models only.

table 1: Supported models



For PTP-capable Allied Vision cameras with firmware version < 01.54.11026, please refer to the following application note:

www.alliedvision.com/fileadmin/content/documents/products/cameras/Prosilica_GT/appnote/PTP-IEEE-1588_Prosilica-GT_Below-01.54.11026.pdf



Compatibility between two PTP implementations

PTP-capable cameras with the older PTP implementation have limitations with the newer PTP implementation.

When using PIP in a network with cameras running a mix of older and newer PIP firmware, the **Auto** option in **PtpMode** cannot be used, and the Master camera (where **PtpMode = Master**) must be a camera running the newer firmware.

Description

IEEE 1588 Precision Time Protocol (PTP) manages clock synchronization of multiple devices across an Ethernet network. Once the clocks of devices such as cameras, PCs, and sensors are synchronized, future software based triggers can be synchronized within two microseconds.

Background

PTP was designed to improve on existing clock synchronization methods such as Network Time Protocol (NTP) and Global Positioning System (GPS). NTP suffers from poor accuracy, often quoted to be several milliseconds using a fast Ethernet network. GPS provides nanosecond precision using atomic clock and satellite triangulation; however, it is an expensive component to incorporate into a camera.

PTP provides microsecond precision without increasing component cost, providing better accuracy than NTP at a lower cost than GPS.

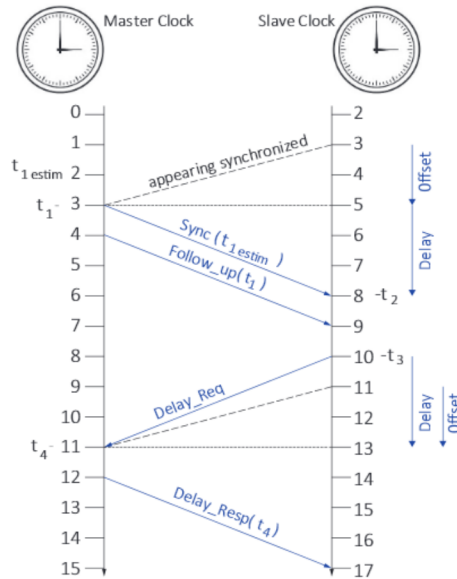


Figure 1: Multicast sync packets

How is PTP synchronization achieved?

Synchronization begins when the device configured as the Master PTP clock transmits a sync telegram using multicast messaging. Devices configured as Slave PTP clocks calculate the time difference between their clock and the Master PTP clock, and adjust accordingly.

Slave clock frequencies are constantly adjusted, through follow up and delay messages, to keep their clock value as close as possible to the master clock. While all Slave clocks are within two microseconds of the master, PTP sync is achieved.

Network Topology

Achieving PTP synchronization between multiple devices requires that all devices be on the same network or subnet. This restriction is due to the current inability of any network card to forward PTP sync multicast packets between ports within the two microseconds requirement.



Figure 2: Multiple Allied Vision cameras on a switch

Therefore, all devices must be connected to a single port through a switch. Note that the total amount of bandwidth from all cameras must not exceed 125 megabytes on a single GigE connection, or packets are dropped. Use **StreamBytesPerSecond** to reduce the bandwidth for each camera. Users of Allied Vision cameras requiring full GigE bandwidth from multiple cameras are encouraged to use a switch with a 10 GigE uplink and 10 GigE network card on the host.

Enabling PTP synchronization with Allied Vision GigE cameras

PTP is enabled with the camera **PtpMode** attribute. Once a device's **PtpMode** is changed to **Master**, **Slave**, or **Auto**, synchronization of devices on the network begins to occur provided there is at minimum one master.

PtpMode = Off

This device's clock is not synchronized with any other device. This is the factory preset PTP mode on Allied Vision GigE cameras.

PtpMode = Slave

This device's clock is altered to align with a master device's clock.

PtpMode = Master

This device's clock is the master clock. All other PTP enabled slave devices synchronize their clock to this device.

PtpMode = Auto

This device uses the IEEE 1588 best master clock algorithm to determine which device is master, and which devices are slaves. It may be assigned as either. There may be several state transitions prior to synchronization.

Determining status of PTP synchronization

All Allied Vision cameras' device clocks are represented by the **GevTimestampValue** feature. Synchronization is complete on all devices when all **GevTimestampValues** are equal to the Master **GevTimestampValue**, and no device's **PtpStatus** = **Initializing**, **Listening**, or **Uncalibrated**. **PtpStatus** is a read only feature.

PtpStatus = Disabled

The camera **PtpMode** is set to **Off**.

PtpStatus = Initializing

PTP is being initialized. If camera or PTP device is being initialized, all device's statuses are set to **Initializing**. This state appears very briefly.

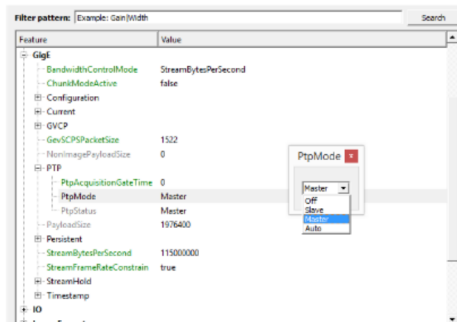


Figure 3: PtpMode in Vimba Viewer

PtpStatus = Listening

Device is listening for other PTP enabled devices. The purpose of this state is to determine which device acts as master.

PtpStatus = Master

Device acting as master clock. If a better master clock is determined, device goes to *Listening*, *Uncalibrated*, and finally *Slave*.

PtpStatus = Passive

If there are two or more devices with **PtpMode = Master**, this device has an inferior clock and is not synchronized to the master.

PtpStatus = Uncalibrated

PTP synchronization not yet achieved. Slaves are syncing with master.

PtpStatus = Slave

PTP synchronization between this device and master is achieved. Device is acting as a slave to another device's master clock.



Previous PTP implementation firmware

PTP capable cameras with firmware version < 01.54.11026 have **PtpStatus = [Off, Master, Syncing, Slave, Error]**.

PTP Sync Failure and Event Callbacks

In the event that a device loses synchronization, it does not disconnect or produce an error status; but it keeps trying to regain sync. Vimba users can use event callbacks to determine when sync is achieved, and if sync is ever lost. This is superior to polling **PtpStatus**.



Event callbacks for PTP

If using the camera event channel, a **EventPtpSyncLost** is sent if **PtpMode** is changed. **EventPtpSyncLocked** is sent once PTP synchronization is re-established.

The cause of a PTP sync failure may be related to a network hardware component's inability to forward multicast sync packets successfully.

Executing a synchronized scheduled software trigger

Once all device clocks are synchronized, it is possible to issue a future simultaneous software trigger event to all Allied Vision cameras using the `PtpAcquisitionGateTime` feature. Follow these steps to enable a synchronized scheduled trigger on multiple cameras:

1. Achieve clock synchronization. Set `PtpMode = Master, Slave` or `Auto`.
2. Set camera `TriggerSelector = FrameStart` and `TriggerSource = Software`.
3. Call `AcquisitionStart()` to start an acquisition stream.
4. Determine the current camera `GevTimestampValue` using `GevTimestampControlLatch`.
5. Set `PtpAcquisitionGateTime` to a value that sufficiently exceeds the current camera `GevTimestampValue`.
6. Execute `TriggerSoftware` command.

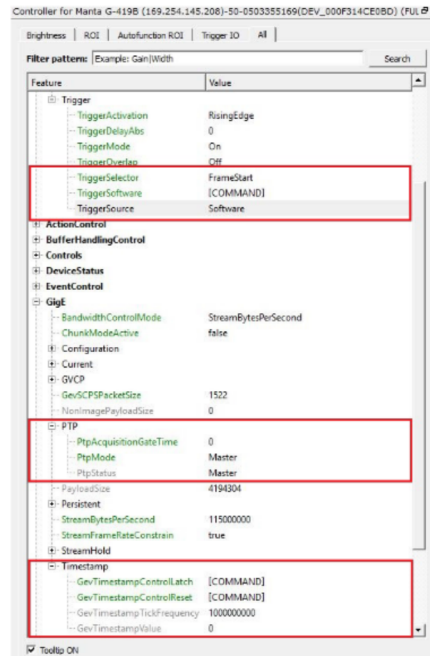


Figure 4: PTP features in the Vimba Viewer

$$\text{PtpAcquisitionGateTime} = \text{Current GevTimestampValue} + \Delta$$

Where, Δ is the combined time required to set `PtpAcquisitionGateTime` on each camera (at minimum). This time varies depending on operating system, host, network hardware, and system load.

Setting `PtpAcquisitionGateTime` suspends all frame triggers in a camera's current `AcquisitionStart()` stream until `GevTimestampValue ≥ PtpAcquisitionGateTime`, at which point frame triggers resumes.

Typical use cases

Case 1 - Fixed frame rate synchronization

1. Synchronize camera clocks.
2. Set `TriggerSelector = FrameStart` and `TriggerSource = FixedRate` on all cameras.
3. Set `AcquisitionFrameRateAbs` to the same value on all cameras.
4. Set `PtpAcquisitionGateTime` the same for all cameras and greater than existing `GevTimestampValue`.
5. Start streaming each camera.

When the camera `GevTimestampValue` increments to the `PtpAcquisitionGateTime`, all cameras start imaging synchronously at the configured frame rate. Reading the `GevTimestampValue` contained in the image header verifies this synchronization.

Case 2 - Precision timing by using GPS

Allied Vision cameras can be synchronized to a GPS timer, allowing real world time synchronization. Configure `PtpMode` on all of the cameras to `Slave` or `Auto`. In `Auto`, the IEEE 1588 best master clock algorithm elects the GPS clock as the master. Each camera synchronizes to the GPS master clock.



Tested GPS clock

We have tested Tekron TTM 01-E GPS clock with sub-microsecond timing. Tekron GPS clock is connected to:

- GPS antenna via a coaxial cable
- Ethernet switch via a Category 6 or higher Ethernet cable

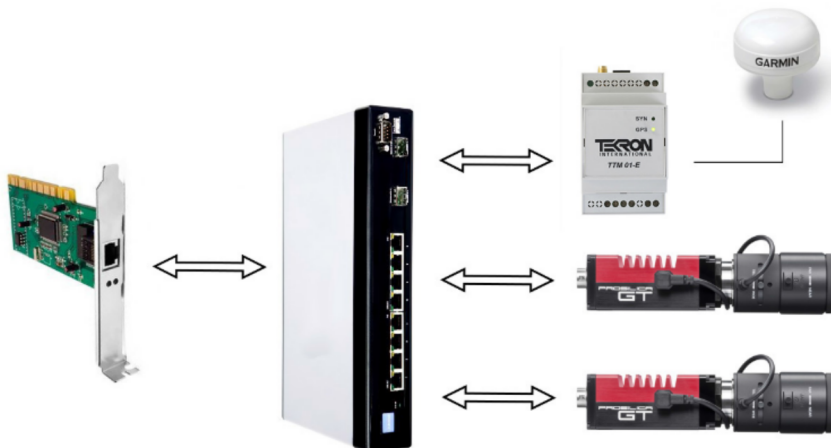


Figure 5: Synchronization using GPS clock

Validating synchronization accuracy of PTP

Method 1

Vimba users can view the time stamp associated with any given frame through the `GevTimestampValue`. A total of four Allied Vision cameras were connected and set up in the following way:

- All cameras were configured to `TriggerSelector = FrameStart` and `TriggerSource = FixedRate`, `AcquisitionFrameRateAbs = 1` fps.
- `PtpMode` on one camera was set to `Master`, `PtpMode` on other three cameras set to `Slave`.
- Same `PtpAcquisitionGateTime` was uploaded to all the cameras using approach shown in Case 1- Fixed frame rate synchronization.

The following graph shows a histogram of the calculated offset between the recorded Master camera frame `GevTimestampValue` and the `GevTimestampValue` of the Slave cameras' `EventLineXRisingEdge`. The data presented is compiled from 69 hours of continuous operation. The maximum offset recorded was less than $\pm 1 \mu\text{s}$. This means that in this test, the time stamp of our three Slave cameras were synchronized to within $1 \mu\text{s}$ of the Master camera.

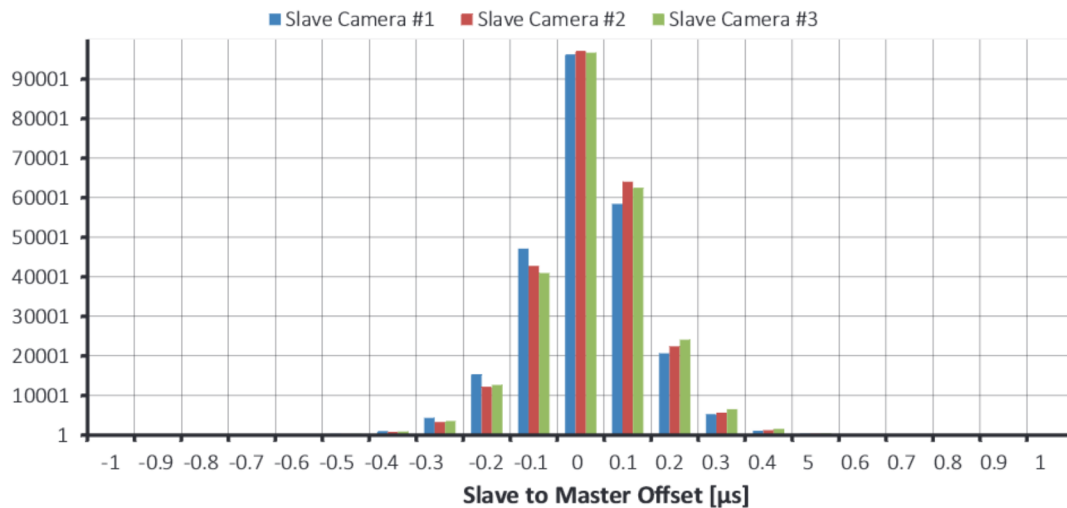


Figure 6: Histogram of slave to master offset

Method 2

To measure PTP accuracy in a different way we can use an oscilloscope and probe output triggers from multiple cameras. Two cameras were connected and setup in the following way:

- All cameras were configured to **TriggerSelector = FrameStart** and **TriggerSource = FixedRate**, **AcquisitionFrameRateAbs = 15 fps**.
- **PtpMode** on one camera was set to **Master**, **PtpMode** on other camera set to **Slave**.
- Same **PtpAcquisitionGateTime** was uploaded to all cameras using approach shown in Case 1- Fixed frame rate synchronization.
- The **SyncOut** signals on both cameras were set to **Strobe1**, **StrobeDurationMode = Controlled**, **StrobeDuration = 10 μs**, **StrobeSource = FrameTrigger**.
- The **SyncOut** signals were monitored on a oscilloscope and viewed over a period of time.

The following capture shows the readout from the oscilloscope when monitoring the **SyncOut** signals from both cameras. In the image, channel 1 is approximately 125 nanoseconds offset from channel 2. This shows that the **GevTimestampValue** synchronization between the two cameras is approximately 125 nanoseconds apart.

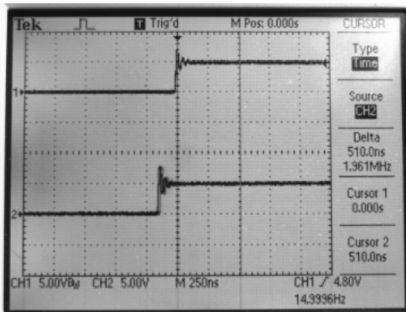


Figure 7: Scope image of oscilloscope showing trigger waveforms

Frequently asked questions:

1. Can I use a host driven, software based PTP clock?

Allied Vision has tested the possibility of using a PC based, PTP software clock to synchronize cameras across multiple adapters. In the process, we discovered that using a host driven master clock on a non real time operating system such as Windows or Linux generated very poor synchronization plagued by significant jitter between the time stamp of cameras. This approach may be sufficient for some applications but it does not generate the precise time synchronization we hope to achieve using PTP.

2. How do I translate **GevTimestampValue** to real world time?

Cameras offering clock synchronization via PTP has a **GevTimestampTickFrequency** of 1 GHz. You can use this to determine the value in seconds that the **GevTimestampValue** represents.

$$\text{RealWorldTime in seconds} = \frac{\text{GevTimestampValue}}{\text{GevTimestampTickFrequency}}$$

To associate this with real world time a GPS master clock needs to be used in the PTP network.

Alternatively, you can reference the latest **GevTimestampValue** to the PC clock; however, this can suffer from significant jitter.

3. What are the differences between the current and new implementation of PTP?

There have been various enhancements included with the latest implementation of PTP including the following changes:

- **PtpStatus** enumerations updated to IEEE 1588-2008 Precision Time Protocol standard:
Initializing(0), Faulty(1), Disabled(2), Listening(3), PreMaster(4), Master(5), Passive(6), Uncalibrated(7), Slave(8), Invalid(9)
- **GevTimestampValue** is no longer reset by turning off PTP
- Acquisition is no longer stopped when changing **PtpMode**
- **PtpAcquisitionGateTime** now reset to zero when **PtpMode** is set to *Off*
- **EventPtpSyncLost** now sent when **PtpMode** changed
- **PtpMode** is no longer set to *Off* when PTP synchronization is lost
- **PtpMode Auto** now correctly finds best master clock
- PTP synchronization drift lessened

For technical support, please contact support@alliedvision.com.

For comments or suggestions regarding this document, please contact info@alliedvision.com.

Copyright and trademarks

All text, pictures, and graphics are protected by copyright and other laws protecting intellectual property. All content is subject to change without notice.

All trademarks, logos, and brands cited in this document are property and/or copyright material of their respective owners. Use of these trademarks, logos, and brands does not imply endorsement.

Copyright © 2021 Allied Vision GmbH. All rights reserved.