

This document contains relevant notes for the libflipro API release. In addition, set up procedures and trouble-shooting techniques for specific Windows or Linux issues can also be found here.

API Documentation

The API Documentation is included with the API library. It is located in the `html` folder within the release package. To begin reading the API documentation, launch a web browser and locate and open the file `html/index.html`.

Windows Notes

Getting the Camera Recognized Properly on Windows

This section was originally written for Windows 7 issues. However, some later version of Windows and their updates do not always handle WinUSB driver setup appropriately, particularly with respect to Container ID descriptor support. If you are having trouble, getting your camera to show up in Device Manager correctly, it is a good idea to run through this section and make sure the registry entries contain the values outlined below.

- The libflipro API uses the standard WinUSB driver to communicate to the camera. WinUSB driver support is found in the OS for versions 8.1 and beyond. In order to get support on Windows 7, you must make sure that ALL the updates have been installed on your machine.
- Make sure your Windows 7 settings are configured to automatically search and get driver updates.
- In addition, you may find that if you plug in the camera prior to installing the updates, that the camera is not recognized properly even after the updates are installed. This is due to the WinUSB driver behavior as documented here <https://docs.microsoft.com/en-us/windows-hardware/drivers/usbcon/microsoft-defined-usb-descriptors> at the time of this writing.

HKLM\SYSTEM\CurrentControlSet\Control\UsbFlags\vvvvpppprrrr

The operating system creates a registry entry, named **osvc**, under this registry key that indicates whether the device supports Microsoft OS Descriptors. If the device does not provide a valid response the first time that the operating system queries it for a Microsoft OS String Descriptor, the operating system will make no further requests for that descriptor.

In this situation, the camera will appear in Device Manager->Other devices->Kepler SCMOS Camera, most likely with a yellow caution icon. In order to fix this, follow the steps below:

1. 'Uninstall' this device from the system.
 - a. Right click on the Device Manager->Other devices->Kepler SCMOS Camera entry and select 'Uninstall'. This device will be removed from Device Manager.
2. Turn off the power to the camera.
3. Edit the registry entry shown above so windows will correctly install the WinUSB device driver for the camera device.
 - a. Run the 'regedit' program as administrator. You can do this by clicking the Start Menu and then just start typing 'regedit'. When you

see the 'regedit.exe' program in the Programs window, right click on it and select 'Run as administrator'. If it asks do you want to allow the program to make changes, click Yes.

- b. Open the Registry key
`HKLM\SYSTEM\CurrentControlSet\Control\UsbFlags\vvvvpppprrrr` where
`vvvvpppprrrr` is `0F18000E0000` – this is the Camera Device. The
`0F18` and `000E` is the Vendor ID (VID) and Product ID (PID)
respectively for the Kepler cameras. If you have a different model you
are trying to get recognized, look for the VID/PID of your camera model.
The VID should always be `0F18`.
 - c. Select the `osvc` entry in the right window pane, right click and select
'Modify...'.
d. In the edit window, change the value to `01 01`. Make sure you do not add
values; `01 01` must be the only numbers in the entry. Click OK.
 - e. Similarly, edit the `SkipContainerIdQuery` entry and make the value
`01 00`. If this field does not exist create it as follows:
 - i. Right click in the right-hand pane and select New->Binary
Value
 - ii. For the name, enter `SkipContainerIdQuery`. Note that case
is important.
 - iii. Right click on the entry and select Modify... as above and change
the value to `01 00` again making sure those are the only values.
 - f. Exit the `regedit` program.
4. Now, power up the camera. If Device Manager is still up, you may see the camera
appear momentarily in the Other devices tree as before. This only occurs
while the WinUSB driver is being installed. Once WinUSB is installed for the
camera, the device shows up as Device Manager->Universal Serial
Bus devices->WinUsb Device. Hopefully this will be your last step.
 5. If the camera still shows up as an Other Device, try and update the driver
manually.
 - a. Right click on the Device Manager->Other devices->Kepler
SCMOS Camera entry and select Update Driver... .
 - b. Choose the links that automatically search for driver updates.
 - c. Windows should find the WinUSB driver and install it for the camera.
 6. Now you should be ready to use the camera on Windows 7!

Visual C++ Redistributable for Visual Studio 2015

It may be necessary to download and install the Visual C++ Redistributable. This can be found at <https://www.microsoft.com/en-us/download/details.aspx?id=48145> at the time of this writing. Download and install the appropriate version (x86 or x64) based on your hardware and operating system.

Linux Notes

Linux Device Permissions

In order for the camera to be accessible to normal users, you need to apply some rules to the udev subsystem. Along with your release, you will find a file '10-flirusb.rules'. Copy this file to the /etc/udev/rules.d directory. You may of course modify this file to suit your particular installation needs. It is simply provided as a working sample to get your development started. There is a brief description of its function inside the file so you may read that for more information.

Libusb Support

The Linux libflirpro API relies on the Linux libusb library. You must install this on your machine for the API to communicate with the camera. On Ubuntu, you can install this with the command 'sudo apt-get install libusb-1.0'. If you are running a different Linux variant, run the appropriate command to install the library.

Libusb Memory Allocation

The USB subsystem has a chunk of memory allocated for it by linux and its default is (usually) 16MB. To try and get the most out of your USB connection, you can modify this memory size as follows:

You can change the amount of memory that is allocated to the USB subsystem through the /sys/module/usbcore/parameters/usbfs_memory_mb system file. The command

```
cat /sys/module/usbcore/parameters/usbfs_memory_mb
```

will show you how many MB's are allocated. Reboot your system and before you connect your camera you can change that with the command like the following that increases it to 64 MB

```
echo 64 > /sys/module/usbcore/parameters/usbfs_memory_mb
```

This will only last until the next time you reboot your system however. But it is best to test it this way first before trying to make the change permanent.

If you do want to make that a permanent change, you need to do that within the `initd` or `systemctl` subsystem in linux. The following should work:

- If there is already a file /etc/rc.d/rc.local, then all you need to do is add that `echo 64` line above to the file. Add it at the end, but before the `exit 0` line if it exists.

- If the file does not exist, then you can create one with your favorite editor and make it look like the following

```
#!/bin/bash
echo 64 > /sys/module/usbcore/parameters/usbfs_memory_mb
exit 0
```

- Make the rc.local file executable

```
sudo chmod 755 /etc/rc.d/rc.local
```

Now you can reboot your system and there should be 64MB allocated for the USB subsystem. You should not need any more than that as there are no frames larger than that. If it is still 16MB, you may need to enable the `rc.local` execution at boot with the following command:

```
sudo systemctl enable rc-local
```

Other Linux Support

For version 2.x.y or greater of the API, you need to install the Simple DirectMedia Layer. On Ubuntu, this can be done with

```
> sudo apt-get install libsdl2-2.0
```

Use the appropriate command for your linux distro.

Release Notes

API Version 2.1.5

Date 12/18/2023

General: Bugfix.

Version 2.x.y of the API is a major update to the FLI Camera API. See Notes for version 2.0.1 if you are updating from version 1.x.y.

API UPDATES:

1. None.

Issues Addressed:

1. Fixed Windows PCIE Fibre connection issue not detecting connected camera..

Features Added:

1. None

Features Removed:

1. None

API Version 2.1.4

Date 11/02/2023

General: Bugfix.

Version 2.x.y of the API is a major update to the FLI Camera API. See Notes for version 2.0.1 if you are updating from version 1.x.y.

API UPDATES:

1. Updated shutter control documentation. See `FPROCtrl_GetShutterOpen()`, `FPROCtrl_SetShutterOpen()`, `FPROCtrl_GetShutterOverride()`, `FPROCtrl_SetShutterOverride()`.

Issues Addressed:

1. Fixed power up settings issue with 6060 that caused incorrect images unless image dimensions explicitly set by application.
2. Addressed some USB command/response timeout that were too small allowing commands and responses to get out of sync with camera under heavy load. Should update camera code as well to 1.12.19 or newer.
3. Fixed power up sensor training issue with Kepler400 cameras causing invalid images in sample applications.

Features Added:

1. None

Features Removed:

1. None

API Version 2.1.3

Date 09/08/2023

General: Bugfix.

Version 2.x.y of the API is a major update to the FLI Camera API. See Notes for version 2.0.1 if you are updating from version 1.x.y.

API UPDATES:

1. `FPROFrame_IsAvailable()` has been removed. `FPROFrame_GetVideoFrameExt()` now waits indefinitely for image data (until aborted) and no longer uses the function.

Issues Addressed:

1. Fixed logging bug on Linux which would always cause TRACE level messages to be logged rather than only up to the set level.
2. Updated linux pcie driver to use dma_ functions rather than pci_ functions.

Features Added:

1. None

Features Removed:

1. None

API Version 2.1.2

Date 02/22/2023

General: Bugfix.

Version 2.x.y of the API is a major update to the FLI Camera API. See Notes for version 2.0.1 if you are updating from version 1.x.y.

API UPDATES:

1. Added API `FPROFrame_GetVideoFrameUnpackedExt()`. This allows the caller to get unpacked images when using the external triggering support.

Issues Addressed:

1. Fixed mutex problem with `FPROSensor_GetGainTable()` that could cause a deadlock.

Features Added:

1. None

Features Removed:

1. None

API Version 2.1.1

Date 02/07/2023

General: Version 2.x.y of the API is a major update to the FLI Camera API. See Notes for version 2.0.1 if you are updating from version 1.x.y.

API UPDATES:

1. Modified the `FPROUNPACKEDIMAGES` structure. The generic `pMetaData` field was removed and `pLowMetaData`, `pHighMetaData`, and `pMergedMetaData` were added to reflect the image data for their respective image frames.
2. Updated API interfaces to meta data. Added the `FPRO_META_KEYS` enumeration and the following related calls:
`FPROFrame_MetaValueInit(wchar_t* pFileName);`

```
FPROFrame_MetaValueInitBin(uint8_t* pMetaData, uint32_t uiLength);
FPROFrame_MetaValueGet(FPRO_META_KEYS eMetaKey, FPROMETAVALUE* pMetaValue);
FPROFrame_MetaValueGetNext(FPROMETAVALUE* pMetaValue);
```

Issues Addressed:

1. None

Features Added:

1. None

Features Removed:

1. None

API Version 2.0.9

Date 11/23/2022

General: Version 2.x.y of the API is a major update to the FLI Camera API. See Notes for version 2.0.1 if you are updating from version 1.x.y.

API UPDATES:

1. None.

Issues Addressed:

1. Fixed GetMode() hang in Linux API.

Features Added:

1. None

Features Removed:

1. None.

API Version 2.0.6

Date 11/07/2022

General: Version 2.x.y of the API is a major update to the FLI Camera API. See Notes for version 2.0.1 if you are updating from version 1.x.y.

API UPDATES:

1. Changed the name of the Linux libraries. The extra 'lib' prefix was removed to better align with standard Linux practices.

Issues Addressed:

1. None.

Features Added:

1. Added FPROCAP_ROI_SUPPORT capability and associated API support on the 4040. The API does not support any other camera models with this feature as of this release. Also see the FPROCAPROI enumeration.

Features Removed:

1. None.

API Version 2.0.5

Date 10/21/2022

General: Version 2.x.y of the API is a major update to the FLI Camera API. See Notes for version 2.0.1 if you are updating from version 1.x.y.

API UPDATES:

1. Added API call `FPROFrame_MetaDataToStringBin()` to allow conversion of meta data straight from image data.

Issues Addressed:

1. Fixed missing dependencies in linux builds.

Features Added:

1. None.

Features Removed:

1. None.

API Version 2.0.1

Date 09/14/2022

General: Version 2.x.y of the API is a major update to the FLI Camera API. It now includes support for the LS line of cameras as well as the Kepler and Cobalt lines. Included here is a summary of the modifications made to the previous versions (1.x.y) of the API. Many function signatures and data structures have changed. Please review and make sure you have updated your code accordingly.

API UPDATES:

1. `FPRODEVICEINFO` structure changed. This was done in order to better support multiple USB connection types and future network connections.
 - a. A new `FPROCONINFO` structure was introduced.

- b. The attr union has been introduced to include specific physical connection information.
2. FPROCAP structure and its associated calls have been changed to a per capability access mode. See the new enum FPROCAPS (yes with an 'S'). In addition, the FPRO_GET_MAX_PIXEL_DEPTH macro has been removed.
 - a. FPROSensor_GetCapabilities() was removed and replaced with FPROSensor_GetCapabilityList(). The list is indexed with the FPROCAPS enumeration.
3. FPROFrame_GetReferenceRowPostFrameCount() and FPROFrame_GetReferenceRowPreFrameCount() have been consolidated into a single function FPROFrame_GetFrameReferenceRows() that returns both pre and post row counts.
 - a. Same for FPROFrame_SetReferenceRowPreFrameCount() and FPROFrame_SetReferenceRowPostFrameCount().
4. FPROSensor_GetBlackLevelAdjust() has changed signature. It adds the FPROBLACKADJUSTCHAN as a parameter. Related to that, the FPROSensor_GetBlackLevelAdjustEx() function has been removed as it was redundant. If you were using FPROSensor_GetBlackLevelAdjust() previously, just pass FPRO_BLACK_ADJUST_CHAN_LDR as the eChan argument to get the same results. We could have just removed the FPROSensor_GetBlackLevelAdjust() call instead, but having an 'ex' function didn't seem to make sense without a non-'extended' function.
 - a. Same for FPROSensor_SetBlackLevelAdjust().
5. Similarly, FPROSensor_GetBlackSunAdjustEx() has been removed and FPROSensor_GetBlackSunAdjust() has an eChan parameter.
 - a. Same for FPROSensor_SetBlackSunAdjust().
6. The FPRODBGLEVEL enumeration was changed to #defines so multiple levels could be defined. A typedef was created that typedefs FPRODBGLEVEL to an unsigned int in order to keep changes to existing code to a minimum. Existing app code should work with no changes with respect to the debug functionality.
7. Added the FPROHDR enumeration. Related to this, the FPROSensor_GetHDREnable API signature changed and a FPROSensor_SetHDREnable API was added.
8. FPROFrame_GetPixelConfig() and FPROFrame_SetPixelConfig have been renamed to FPROFrame_GetPixelFormat() and FPROFrame_SetPixelFormat() respectively. In addition, the signature of those functions has been changed.
 - a. Supporting this change, the FPRO_PIXEL_FORMAT enum has been added and a new function, FPROFrame_GetSupportedPixelFormats() has been added to allow an application to discover what pixel formats are supported by the connected camera.
9. The FPRO_MERGEALGO enumeration has been removed as the FPROMERGE_ALGO setting is no longer supported. All merging is done with the same reference frame algorithm.
 - a. The FPROUNPACKEDIMAGES structure has changed. The eMergAlgo, cropRect, and bRequestCrop members have been removed.
 - b. A new member eMergeFormat was added to the structure. On calling the FPROFrame_GetVideoFrameUnpacked() API it is the requested image/file format for the merged data. On return, it is the actual format. Not input types can be converted to all the other types. In the event the API cannot convert to the requested type, it will default to TIFF and set this field accordingly for the caller.
 - c. FPRO_CROP structure has been removed. It was never implemented.

10. The FPROFrame_SetUnpackingBiasFrames() and FPROFrame_SetUnpackingFlatFieldFrames() API's have been removed. The corrections are handled via the reference frames.
11. Added FPROCAP_MERGE_REFERENCE_FRAMES_SUPPORTED capability. 0= not supported. Use this to enable/disable sending of reference frames in your application.
12. Changed FPROCtrl_GetTemperatures() signature. Specifically, the name of the first parameter changed since there is no ambient temperature on Keplers. There may be on other cameras however (or another temperature) so the name was changed to pOther. Made each parameter optional as well.
13. FPROAuxIO_GetExposureActivePolarity() and FPROAuxIO_SetExposureActivePolarity() have been removed. The polarity parameter is now part of the FPROAuxIO_GetExposureActiveType() and FPROAuxIO_SetExposureActiveType() API's.
 - a. The signatures of FPROAuxIO_GetExposureActiveType and FPROAuxIO_SetExposureActiveType() have changed. Each add a ePin parameter and the polarity parameter as described above.
 - b. The FPROAUXIO_EXPACTIVETYPE enumeration has changed. A new enumeration has been added with a value of FPRO_AUXIO_EXPTYPE_FLASH.
14. Added FPROSensor_GetCameraFlatFieldCorrection() and FPROSensor_SetCameraFlatFieldCorrection() API Functions. See the docs for info on how to disable/enable and if it supported on your camera.
15. Added enumeration FPROGPSOPT. Changed the signature of the FPROCtrl_GetGPSState() API. It now contains the pOptions parameter.
 - a. Added the FPROCtrl_SetGPSOptions() API call.
16. Removed FPROCAM_GetHostSerialNumbers() and added FPROCAM_GetHostInterfaceInfo(). To support the new call, added the structure FPROHOSTINFO. This allows PCIE card information (e.g. version info) to be extracted without a camera being connected.

Issues Addressed:

1. None.

Features Added:

1. None.

Features Removed:

1. None.

API Version 1.12.63

Date 08/30/2022

General: Bug Fix Feature Enhancements.

Issues Addressed:

1. Fixed GPS time error.
2. Added BSI/FSI differences for 2020 black level.

Features Added:

1. Internal: Added EOL configuration capability to the engineering library.

Features Removed:

1. None.

API Version 1.12.59

Date 03/22/2022

General: Bug Fix Feature Enhancements.

Issues Addressed:

1. Fixed intermittent Fibre connection issues.

Features Added:

1. None.

Features Removed:

1. None.

API Version 1.12.58

Date 01/06/2022

General: Bug Fix Feature Enhancements.

Issues Addressed:

1. None.

Features Added:

1. Added FPROAuxIO_GetExposureActivePolarity() and FPROAuxIO_SetExposureActivePolarity() API's.

Features Removed:

1. None.

API Version 1.12.58

Date 01/06/2022

General: Bug Fix Feature Enhancements.

Issues Addressed:

1. Fixed Fiber connection issues. Requires update of PCIE Fiber card FPGA Firmware: fiber_fpga_v1.2.0_0xDE2B0E1F, and pcie_fpga_v1.3.3_0x192B0190 or later versions.

Features Added:

1. None.

Features Removed:

1. None.

API Version 1.12.56

Date TBD

General: Bug Fix Feature Enhancements.

Issues Addressed:

1. SIGNATURE CHANGE: The FPROAlgo_StackFinish() signature changed. Parameters were added so the user can get the meta data for the stacked frames if desired.
2. Fixed API stacking function. The last 384 bytes were not being stacked properly (the size of the meta data).
3. Fixed Multi-Camera connection issues around correct capture stopping/abort procedures.

Features Added:

1. None.

Features Removed:

1. None.

API Version 1.12.54

Date 10/15/2021

General: Bug Fix Feature Enhancements.

Issues Addressed:

1. Fixed 2020 Image sizing issues.

Features Added:

1. None.

Features Removed:

1. None.

API Version 1.12.52

Date 09/01/2021

General: Linux only release to fix image pipe timeout.

Issues Addressed:

1. Fixed Meta Data date string parsing.
2. Fixed image pipe timeout on Linux side. It was being set to no timeout (0) which is an infinite timeout. This fix put it back to the normal camera default (e.g. 10 seconds).

Features Added:

1. Added new debug level `FPRO_DEBUG_REGRW` to help aid in hardware debugging.

Features Removed:

1. None.

API Version 1.12.51

Date 08/12/2021

General: Bug Fix Feature Enhancements.

Issues Addressed:

1. **STRUCTURE CHANGE:** The `FPROUNPACKEDIMAGES` structure did not correctly support FITS and TIFF formats for the Merged Image buffer size. New fields have been added to the structure: `uiLowBufferSize`, `uiHighBufferSize`, and `uiMergedBufferSize`. In each case they represent the size of their corresponding buffer in bytes. The existing `uiLowImageSize`, `uiHighImageSize`, and `uiMergedImageSize` fields are the size of the image data in pixels and do not include whatever header bytes are included for FITS and TIFF formats.
2. **STRUCTURE CHANGE:** In the `FPRO_HWMERGEENABLE` structure, the `bGenerateTIFF` field has been replaced by the `FPRO_IMAGE_FORMAT` enumeration field `eMergeFormat` in order to support the new FITS merging functionality.
3. For the `FPROAlgo_SetHardwareMergeReferenceFiles` API, at least one of the parameters must now point to a valid file. Without a valid file, setting

a valid reference image size was not reliable. If you need to create identity frames, use the `FPROAlgo_SetHardwareMergeReferenceFrames` instead.

4. API merging performance and memory footprint issues addressed.
5. Fixed linux libusb issue with multiple cameras connected to the same computer.
6. Fixed linux signal issues with long exposure times. The cancel signal was not being cleared properly causing an erroneous user 'cancel'.
7. Fixed LED Duration settings for various newer camera firmware versions.
8. Fixed Cobalt data read out times causing long exposures to fail.

Features Added:

1. FITS format support for API merging. See the documentation for `FPROUNPACKEDIMAGES`, and `FPRO_HWMERGEENABLE`. The `FPRO_HWMERGEENABLE` structure has changed as described above.

Features Removed:

1. The dependency on the external `cfitsio` has been removed. **NOTE: Fits support remains, the dependency on the third party library has just been removed.**

API Version 1.12.46

Date 06/16/2021

General: Bug Fix Feature Enhancements.

Issues Addressed:

1. Fixed problems with API merging when not requesting the merged image plane.
2. Fixed TIFF generation issues related to API merging.
3. Fixed image plane statistics computation when not requesting the merged plane statistics.
4. Fixed FTM FPGA Programming.
5. Fixed flushing of Windows side image pipes on aborted image captures.

Features Added:

1. When using the API merging capability and the `FPROMERGE_ALGO_REF_FRAME` algorithm, if no reference frames are sent, the API attempts to compute a set of reference frames from the Meta Data of the image. Previously, straight identity frames were used.
2. Added FITS support through the `FPROFrame_ConvertFile()` API.

Features Removed:

1. None.

API Version 1.12.36

Date 04/30/2021

General: Bug Fix Feature Enhancements.

Issues Addressed:

1. Fixed NV Storage issue with upgrading from very old releases.
2. Fixed HDR mode exposure time readback issue on 400. Exposure time was being reported as twice the actual value.
3. `FPROFrame_CaptureAbort()` and `FPROFrame_CaptureStop()` now disables external triggers.

Features Added:

1. Changed the signatures of `FPROCtrl_GetExternalTriggerEnable()` and `FPROCtrl_SetExternalTriggerEnable()` The new signatures are now
 - `FPROCtrl_GetExternalTriggerEnable(int32_t iHandle, FPROEXTTRIGINFO* pTrigInfo);`
 - `FPROCtrl_SetExternalTriggerEnable(int32_t iHandle, uint32_t uiFrameCount, FPROEXTTRIGINFO *pTrigInfo);`
2. Added `FPROEXTTRIGINFO *pTrigInfo` structure for above calls and new `SingleFramePerTrigger` feature.

Features Removed:

1. `FPROFrame_CaptureEnd()` has been removed (it was never included functionality on any camera.). Use `FPROFrame_CaptureAbort()` instead.

API Version 1.12.30

Date 03/19/21

General: Bug Fix Feature Enhancements.

Issues Addressed:

1. Cleaned up Fibre card warnings when trying to open a fibre card connection when no fibre card is present.
2. Modified the `FPRO_HWMERGEENABLE` structure to take an enum for the planes to merge rather than separate Booleans which could not all be set at the same time.

Features Added:

1. Added FPROAlgo_SetHardwareMergeReferenceFiles API function
2. Added FTM support.

Features Removed:

1. None

API Version 1.12.28

Date 02/10/21

General: Bug Fix Feature Enhancements.

Issues Addressed:

1. Fixed erroneous success values being returned from Hardware Merge functions.
2. API Now initializes Hardware Merging Reference Frames on connection.
3. Improved streaming preview performance.
4. Fixed false errors from HWMerge* related functions when using them via the API over USB connection.

Features Added:

1. Added support for Linux PCIE V2 fibre driver.

Features Removed:

1. None

API Version 1.12.22

Date 11/03/2020

General: Added PCIE V2 Fibre support.

Issues Addressed:

1. None.

Features Added:

1. Added Hardware Merging support. Implemented on PCIE Fibre connections with V2 hardware or better. See the FPROAlgo_*HardwareMerge*() functions for details.
2. Added HW Merge emulation support for merging through the API on USB connections or older V1 fibre connections. See the same FPROAlgo_*HardwareMerge*() functions for details.

3. Added `FPROCtrl_SetBurstModeEnable()` and `FPROCtrl_GetBurstModeEnable()` API's.
4. Modified `FPROCtrl_GetCameraBufferBypass()` and `FPROCtrl_GetCameraBufferBypass()` API's to take both camera and host PCIE card bypass enables.
5. Added `FPROCtrl_GetElectricallyBlackPixelEnable()` and `FPROCtrl_SetElectricallyBlackPixelEnable()` API calls.
6. Added host PCIE card Reference Frame support. See `FPROAlgo_SetReferenceFrames()`.
7. Added `FPROCtrl_GetPCIETemperatures()` API call.
8. Added `FPROCtrl_GetFrameDelayMinimum()` API call.

Features Removed:

1. None

API Version 1.12.9

Date 06/04/2020

General: Bug Fix Feature Enhancements.

Issues Addressed:

1. Fixed issue in multithreaded Disk Writer implementation causing Streamer thread failing to start.
2. Added a minimum open to open shutter interval of 2 seconds to protect mechanical limitations.

Features Added:

1. Added `FPROFrame_UnpackFile()` and `FPROFrame_MetaDataToString()` API functions.
2. Added overall throughput measurements to the Streamer statistics. See `FPROSTREAMSTATS`.
3. Added `FPROFrame_StreamGetPreviewImageEx()` function.
4. Added `FPROCtrl_GetCameraBufferBypass()` and `FPROCtrl_SetCameraBufferBypass()` functions.

Features Removed:

1. None

API Version 1.12.3

Date 02/01/2020

General: Bug Fix Feature Enhancements.

Issues Addressed:

1. Fixed intermittent reconnection problems in the event applications do not close down the connection properly- command and image pipes needed to be reset.

Features Added:

1. Added camera command to support active feedback during programming operations. Only used by the Programming Application.

Features Removed:

1. None

API Version 1.11.28

Date 01/15/2020

General: Bug Fix Feature Enhancements.

Issues Addressed:

1. Fixed issue with some Cobalts with `FPROSensor_SetMode()` taking too long to complete causing communications failures. Increased the timeout for the command to allow it to complete on the camera.
2. Fixed 6060 image data unpacking issue. Fixed 6060 Binning Registers.
3. Changed the signature of the unpacking API's `FPROFrame_FreeUnpackedBuffers()` and `FPROFrame_FreeUnpackedStatistics()` to take the camera handle. This was done to support an internal refactor to allow these API's to be used on multiple camera connections.
4. Fixed Unpacking issues with Cobalts when single channel readout modes are used.
5. Updated unpacking API path to always generate latest version of meta data for processed files.
6. Added support for additional hardware (FPGA) version information for 2020 and 6060 cameras.
7. Fixed unpacking issue with Cobalt cameras. Later models changed the Model Name format in the meta data.
8. Changed the signature of the `FPROFrame_ComputeFrameSize()` API. Removed the size parameters because they were not used. The signature is now
`LIBFLIPRO_API FPROFrame_ComputeFrameSize(int32_t iHandle);`

Features Added:

1. Added Bias and Flat Field correction support when using the unpacking API's. See `FPROFrame_SetUnpackingBiasFrames()` and `FPROFrame_SetUnpackingBiasFrames()`.
2. Added support for Sony imx183 sensor based cameras.

Features Removed:

1. None

API Version 1.11.23

Date 10/08/2019

General: Bug Fix Feature Enhancements.

Issues Addressed:

1. Minimum allowed exposure time counts were being violated. Updated so that the minimum setting on the cameras do not go below about 20000 nsecs.
2. Fixed 6060 Black Level adjustment settings.

Features Added:

1. Added Cobalt 230-84 sub-framing, binning, and Sensor readout Configuration support.
2. Added Cobalt 4320 Support.
3. Support added for 6060 Sensor board FPGA programming (internal).
4. Changed the definition of Binning Table entries. See `FPROSensor_GetBinningTable()` in the API documentation for details.

Features Removed:

1. None

API Version 1.11.20

Date 09/09/2019

General: Bug Fix Feature Enhancements.

Issues Addressed:

1. Fixed issue with long exposure times causing a timeout.

Features Added:

1. Added API's allowing application to get unpacked images directly from the cameras. See documentation for `FPROFrame_GetVideoFrameUnpacked()` for more information.

Features Removed:

1. None

API Version 1.11.15

Date 07/17/2019

General: Bug Fix and internal feature enhancements.

Issues Addressed:

1. Changed the FPGA Version number check for the Image Block Size (internal).
2. Fixed bug in FPROFrame_ComputeFrameSize() for Cobalt Cameras. Reference rows were not being counted correctly.

Features Added:

1. Support for 6060 Sensor Introduced.

Features Removed:

2. None

API Version 1.11.9

Date 06/21/2019

General: Bug Fix and internal feature enhancements.

Issues Addressed:

1. Decreased camera detection time on Fibre side.
2. Fixed bugs in ComputeFrameSize() API.
3. Force Frame Count to 1 for External Trigger.
4. Fixed Fibre command transaction timeouts at the end of a streaming session.
5. Streamed file name number now begins with '1' rather than '0'.

Features Added:

1. Added support for generic camera configuration (internal). Updated camera configuration file format accordingly.

Features Removed:

1. None

API Version 1.11.3

Date 05/16/2019

General: Bug Fix and internal feature enhancements.

Issues Addressed:

1. Fixed non streaming side of fibre interface during preview support.

Features Added:

1. Added FPROFrame_ComputeFrameSize() function to API to help users determine how many bytes will be received for a frame.

2. Added Generic Camera Configuration support (internal).

Features Removed:

1. None

API Version 1.10.33

Date 03/27/2019

General: Bug Fix.

Issues Addressed:

1. Fixed Fibre PCIe soft reset issue that was causing connection problems.
2. Fixed Streaming Abort transfer cleanup.

Features Added:

1. None.

Features Removed:

1. None

API Version 1.10.30

Date 03/07/2019

General: Feature Enhancements.

Issues Addressed:

1. None.

Features Added:

1. Added Windows PCIE Fibre Card support.

Features Removed:

1. None

API Version 1.10.24

Date 12/13/2018

General: Feature Enhancements.

Issues Addressed:

1. None.

Features Added:

1. Added Generic camera support (internal only).
2. Added following API calls:
 - FPROFrame_GetFrameType
 - FPROFrame_SetFrameType
 - FPROCtrl_GetLED
 - FPROCtrl_SetLED (not new but included for completeness)
 - FPROCtrl_GetLEDDuration
 - FPROCtrl_SetLEDDuration
 - FPROSensor_GetTrainingEnable
 - FPROSensor_SetTrainingEnable

Features Removed:

1. None

API Version 1.10.21

Date 11/16/2018

General: Bug fix.

Issues Addressed:

1. Fixed issue with retrieving Exposure setting on 4040 cameras - FPROCtrl_GetExposure() was returning incorrect value.

Features Added:

1. None.

Features Removed:

1. None

API Version 1.10.20

Date 11/09/2018

General: Development update.

Issues Addressed:

1. None.

Features Added:

1. Added support for Cobalt DC230 cameras.

Features Removed:

1. None

API Version 1.10.19

Date 10/26/2018

General: Development update.

Issues Addressed:

1. Fixed logging issue in x64 builds of the API.

Features Added:

1. (Internal) Added support for Simple (Generic) Camera type with minimal configuration and settings.

Features Removed:

1. None

API Version 1.10.17

Date 10/12/2018

General: Development update.

Issues Addressed:

1. Fixed issue with updating DAC values when gain index values are changed in HDR modes.

Features Added:

1. None.

Features Removed:

1. None

API Version 1.10.15

Date 09/20/2018

General: Development update primarily for support of 4040 Sensor. Requires camera firmware update to version 1.10.1 or newer. The following API items have changed:

- FPROGAINTABLE enumeration. The names within the enumeration have changed to reflect the correct use of the gain tables. See the documentation for a complete description
- The signature for the following function has changed in order to support the new 4040 Gain Tables. This includes the addition of the new FPROGAINVALUE enumeration.

- FPROSensor_GetGainTable(int32_t, FPROGAINTABLE, FPROGAINVALUE *, uint32_t *);
- Added additional Black Level access functions to support 4040 functionality:
 - FPROSensor_GetBlackLevelAdjustEx(iHandle, FPROBLACKADJUSTCHAN, uint32_t *);
 - FPROSensor_SetBlackLevelAdjustEx(int32_t, FPROBLACKADJUSTCHAN, uint32_t);

Issues Addressed:

1. None.

Features Added:

1. Added Support for Gains, Black Level, and Black Sun on 4040 Camera
1. Refactored and moved Gain tables up into API (deprecated in camera firmware). No operational change to application software. Fixed 4040 gain index conversion issue.

Features Removed:

1. None

API Version 1.10.1

Date 04/10/2018

General: Development Update. Requires camera firmware update to version 1.10.1 or newer.

Issues Addressed:

1. Updated Windows 7 Setup notes in this document to include Windows settings for automatically searching and installing new device drivers.

Features Added:

1. Updated capabilities to version 2.
 - a. Added Meta Data Size to camera capabilities.
 - b. Updated sample programs to reflect new Meta Data use.
2. DAC Value translation was added internal to the API in order to present an ‘increasing’ list to the user of DAC values.
3. Added GPS Detection on camera and support in API to retrieve the GPS state
 FPROCtrl_GetGPSState(int32_t iHandle, FPROGPSSTATE *pState);.

Features Removed:

1. None

API Version 1.7.5

Date 2/20/2018

General: Development Update. Requires camera firmware update to version 1.7.3 or newer.

The following API functions have changed names in this release:

- FPROFrame_GetDummyRowPostFrameEnable() changed to FPROFrame_GetReferenceRowPostFrameCount()
- FPROFrame_GetDummyRowPreFrameEnable() changed to FPROFrame_GetReferenceRowPreFrameCount()
- FPROFrame_SetDummyRowPostFrameEnable() changed to FPROFrame_SetReferenceRowPostFrameCount()
- FPROFrame_SetDummyRowPreFrameEnable() changed to FPROFrame_SetReferenceRowPreFrameCount()

Issues Addressed:

1. When in an LDR imaging mode, setting the LDR gain index now also sets the HDR gain index to the same index value. Image artifacts result if they are different. The API returns an error when attempting to set an HDR gain index when in an LDR imaging mode. Different LDR and HDR gain settings are still possible when in an HDR imaging mode.
2. Fixed internal issue of serial number string descriptor request failing. The Setup packet was being built incorrectly with the Interface as the target. The target must be Device for a descriptor request. Does not affect execution of any other commands. This only happens on device enumeration and a work around was already in place.
3. Fixed internal issue of failure to transmit image frames properly if the frame size was not a multiple of 32 bits (4 bytes).
4. Added pointer to API documentation in this file.

Features Added:

1. Added Non Volatile Storage available, Pre/Post Reference Row fields to the capabilities structure. Bumped the version for the structure to 2 for this change.
2. Added Non Volatile Storage API's to allow users to save camera specific information directly on the camera.
3. Added API use of Image Size, Binning values, and Reference Rows through Capabilities Structure.
4. Added initial 4040 sensor support to API through Camera Capabilities structure.

Features Removed:

1. None

API Version 1.6.1

Date 11/29/2017

General: Development Update. Requires Camera firmware update to version 1.6.1 or better.

Issues Addressed:

1. Abort processing was not reliable and could cause USB communications failure to the camera. Internal API processing was modified with custom 'reset pipes' command to the camera.

Features Added:

1. Added the release notes (this file) to the Linux Release script.

Features Removed:

1. None

API Version 1.5.6

Date 10/30/2017

General: Development Update

Issues Addressed:

1. Shutter close delay changed to signed magnitude on camera.
2. Removed low level pop up on USB Read error. Error is now just logged at ERROR level and FPROFrame_GetVideoFrame() API call returns error code.

Features Added:

1. Added linux release generation script.

Features Removed:

1. None

API Version 1.5.5

Date 10/12/2017

General: Initial Release

Issues Addressed:

1. None

Features Added:

1. None

Features Removed:

1. None

.....