Imaginant, Inc

Frequently Asked Questions

HR Camera SDK

14 February 2005

1.) How do I 'connect' to the HR_Lib dll ?

Expand the HR_SDK zip file into a work folder. Compile, link, and then run SampleCode01. That sample code will dynamically load the library, initialize the library, open and initialize an HR-1100c camera, trigger the camera to make an image exposure, wait for the image, open the image, save it as a JPEG file on your host computer, then close the image, delete the image, close the camera, unload the library,

2.) How do I connect to a camera and take an Image ?

After loading the DLL you can simply open the library, scan for a camera, and open the camera. To see how this is done, see the project SampleCode01.

3.) Do images have to be opened and closed like camera objects?

Yes. Images must be opened before attempting any operations with them. An image must be closed before it can be deleted.

4.) The camera is polled to detect an image transfer having been completed. Does this polling event flag self-reset upon reading it ?

Yes, this is the way all event flags work. The reading of the event group will reset all the events in that group. So when a polling group is read, you must deal with all the events returned in the bitfields, because if you call again, all those events will be gone.

5.) Is there a way to discover than an image is incoming before it has fully arrived?

Yes. Images will show up in the image list retrieved by a call to HR_GetImageList() even before all the data has arrived.

In addition, the camera object can be polled to discover a new image by using the event HR_EVENT_CAM_TRANSFER_NEW_IMAGE_DATA_AVAILABLE.

6.) Can I start to make use of image data before it is fully transferred?

Yes, you can do a GetParam call for HR_PID_IMG_IMAGE_BUFFER_STRUCT. This structure (HR_ImageBufferStruct) will contain a pointer to the image receiving buffer, and a length of the allocated buffer size. To get the current amount of image data in the buffer, it will be necessary to make another called to GetParam with

HR_PID_IMG_BYTES_COLLECTED. This param will be the current amount of data in the HR_ImageBufferStruct at the time the call is made.

7.) Is the param HR_PID_IMG_WHOLE_IMAGE_BYTE_COUNT valid as soon as the image is created, or does transfer have to be complete?

This parameter will be returned as zero until the image transfer is complete. There is a good reason for this. This parameter is the compressed file size of a JPEG stream if the image is a JPEG. The camera firmware can start sending the stream of compressed data before the entire image compression is complete. There is no way for the firmware to know the length of the stream until the compression is complete. Therefore the host cannot know the final size either until the whole image is compressed.

8.) Does the HR_SDK 'own' the HR_ImageBufferStruct, or does my application have to delete or free that when I'm done with it?

The HR_SDK creates the buffer, and it is managed by the image object. The application cannot free or delete this buffer or else a Windows exception will likely follow. The buffer will be freed when the image object is deleted.

9.) Will HR_SDK re-use image handles after I delete them?

There are over 60,000 unique image handles. After cycling through all of the image handles, the library restarts at the beginning of the list. If you are expecting to use over 60,000 images in one instance, you might run up against repeated image handles

10.) Is there a way I can tell via some status check to see if an image is already open?

There is no harm in trying to open an image which is already open. In this case, the call will fail with an error code of HR_ERR_IMG_OPEN.

11.) Should camera objects and images be closed before closing the library?

Yes. When the library is closed, it frees all of its objects, and deallocates the memory it has allocated. If the user does not close the camera objects and image objects themselves, the library guesses at what the best way to close is, and does it. This means that some memory being used by the end user may end up being deallocated by the library if the library believes it is still tied to its objects.

12.) This question concerns preview mode. Do I need to set the current file type to a particular value, or will the images come in as raw without errors even though the current type is JPEG?

When preview mode is engaged via SetParam with HR_PID_CAM_RUN_PREVIEW, preview images will start becoming available. They will be in the raw preview format, thus the current file type will be ignored during preview mode.

13.) Are there any differences between how regular images and preview images are detected, opened, and deleted?

No, they will appear the image list just like regular captured images. However, they come automatically at a much faster rate without being triggered. When the application turns preview mode on, it should be ready to receive, display and delete them quickly.

14.) An image object can be opened and closed again repeatedly without a problem. Do camera objects behave the same way?
No. Opening a camera object begins a complex negotiation with the camera hardware. Upon closing, the conversation is halted and the camera may even reboot the firmware. In this case, the camera would then show up in the list of available cameras for a new connection, even though the original camera object still exists. To best avoid difficulties, always delete a camera object after closing it. If communication with the camera is to be reestablished, simply open the camera when it appears in the cameras available list.

15.) Should camera objects and images be closed before closing the library?
Yes. Although the library will try to delete any objects left, it is better from a reliability standpoint to close and delete all image and camera objects before closing the library.

16.) When a call to SetParam changes a camera setting such as ISO, will HR_SDK wait until the camera has made the change before the call to SetParam returns?
Yes. Be aware that this means the call to SetParam may take some time before returning. Some parameters such as SetFocus may take several seconds to execute. For applications which need to be responsive, the application programmer may want to create a separate thread for all communication with the HR_SDK.

17.) How is bounds checking for parameters handled? For example, if I try to set an ISO of 10000, will I get an error because it is out of range?
First of all, the application can retrieve the min and max values for a parameter with GetParam calls.
When the SetParam call is made, HR_SDK first checks the parameter against the min and max values stored in library. If it fails this test, the call to SetParam returns immediately with an error code and the attempt is not communicated to the camera. No settings change results.
If the HR_SDK approved the value, it will then be sent to the camera. If the camera rejects the value, an error code will be returned, but the camera may have attempted to choose the closest legal value. This is unlikely, but possible.
The best way to ensure the camera is using the desired value is to issue SetParam again with a value which does not generate an error.

18.) What happens if you exit your app without closing the library? What happens to the camera? Does it reboot?
The Library destructor checks if the library is closed, and calls the library close function at destruct time if it is not. Any open objects will be closed and deleted, but it is best not to rely on this. This can make the camera reboot, but there may be cameras in the future which do not need to actually reboot when the connection terminates.

19.) When closing the camera, will HR_SDK not return until the camera is actually closed and the communication shutdown?
Yes.

20.) My application would like to read the values of various camera parameters before it is opened. Will GetParam calls work with a camera object before it is opened?

Yes, but the status return value from the GetParam call will be HR_ERR_CAM_DISCONNECTED_PARAM_NOT_VALID. This return value indicates that GetParam did retrieve a default value for you, but it is not guaranteed to be correct after the camera has opened. For example, the factory-specified white balance parameter can be unique in every camera unit. The HR_SDK cannot know this value until the camera is opened. GetParam will return a reasonable value for white balance if called before opening the camera, but the camera may change it to the factory setting upon startup. Issuing another GetParam after the camera is open will result in the actual value being used by the camera.

21.) Can multiple applications running on the same computer open the HR_SDK library?

No, only one application should have the HR_SDK library open.

22.) I modified your sample code to make TIFF images instead of JPEG images. The resulting images are very dark and have a weird checkerboard pattern. What gives?

The TIFF formatted image supported at this time is the raw color filter array mosaic image, with only one 12-bit color component at each pixel location. The image data needs to be interpolated to make an RGB pixel at each point, then processed through a gamma curve and color correction matrix.