

Working With the Basler A406K Camera and the BitFlow Karbon

Revision 1.1, 2009-02-11

Introduction

The Basler A406K camera outputs is an extremely fast camera, capable of acquiring 2320 x 1726 images at 200 FPS. That results in a data rate of over 800 MB/S. This requires using the maximum bandwidth the Camera Link interconnect is capable of, that is, 10 taps with a clock rate of 85 MHz. The current generation of Karbon can acquire at this rate, but there is not enough head room to DMA data at this rate continuously over the PCI Express bus for long term reliable operation. For this reason, BitFlow has designed the "2x" mode, which shares the DMA responsibility between two DMA engines. The 2x mode effectively doubles the bandwidth of the Karbon, and provides plenty of headroom to DMA images from the camera continuously, regardless of system load.

In 2x mode the DMA operations are split up on a line by line basis. One DMA engine is responsible for all of the odd lines and the other DMA engine is responsible for all the even lines. Both DMA engines can DMA to the same host buffer simultaneously. Both DMA engines are synchronized because both Virtual Frame Grabber's (VFG's) acquisition engines are synchronized in hardware. The end result is a seamless end-to-end acquisition system.

Requirements

The following items are required for interfacing to the Basler A406K:

- BitFlow Karbon Frame Grabber - Model KBN-PCE-CL4F-IP4
- BitFlow SDK 5.20 (or later) - Download from www.bitflow.com, serial number required
- Camera Link Cable - Two required
- PC with at least on x8 PCI Express slot
- Microsoft Windows - 32-bit or 64-bit

Software Installation

Instal the SDK according the Getting started documentation. It is generally recommended that you install the SDK before installing the Karbon into the system.

After installing the Karbon, make sure that the Karbon works correctly with the synthetic camera file. The Karbon 4F will appear to Windows as to separate devices, known as Virtual Frame Grabbers (VFGs). Make sure to test both VFGs with the synthetic camera file before proceeding.

Mechanical Setup

Connect the A406K to the karbon as follows:

- Camera top connector (the on nearest the LED) → Karbon top connector (CL1)
- Camera bottom connector → Karbon bottom connector (CL2)

Note: that the Karbon CL4F comes with an auxiliary board, this board is not need when using the A406K. You do not need to install the auxiliary board for this camera.

Note: Because this camera uses the maximum Camera Link clock of 85 MHz, BitFlow recommends using the shortest CL capables possible that will work with your application.

Example Applications

BitFlow SDK 5.2 comes with two example applications which have been designed to work specifically with the A406K. For live display of the A406K, CiView can be used. For capturing sequences of images, BiFlow2x can be used.

CiView

Normally CiView displays the output of only one board. However, by running CiView with the “-2” command line parameter, it will let the user open two boards and acquire synchronized acquisition from both boards (VFGs). This dual board mode is what is needed to acquired from the A406K.

CiView with the “-2” command line parameter can be from a console window using the following command:

```
>CiView -2
```

Alternatively, a shortcut can be created in Windows for CiView. Once the short cut is created, the properties of the shortcut can be modified. In the shortcut properties dialog, command line parameters can be added, so that every time the short cut is double click, the application is run with the given command line parameters.

When running CiView with the “-2” parameter, two boards need to be opened. Therefore, the board open dialog will appear twice. Open the Master board first (connections CL1/CL2) and the slave board second (connections CL3/CL4).

CiView will display the live full-bandwidth output fomr the A406K.

BiFlow2x

BiFlow is BitFlow’s main sequence capture example application. Sequences can be capture to host memory, the displayed individually or played back at a selectable frame rate. Sequences can also be save to disk as video or sequentially number images files.

BiFlow2x works exactly like the regular version of BiFlow, except that two boards can be opened and then they are setup for synchronized acquisition from the A406K. Once both boards are opened, all operations in BiFlow2x work the same the regular version of BiFlow.

Note: Because the data rate of the Basler A406K is so high, it can quickly fill memory capturing even short sequences. BitFlow recommends using a 64-bit operating system with this camera which supports very large amounts of memory that can be used for image capture. All of the instructions in this document work for both 32-bit operating systems and 64-bit operating systems.

API Support for the A406K

Very few changes are needed in order to support writing applications for the A406K. The functions listed below have been added to SDK 5.20 to provide support for this camera. With the exception of these changes, all other functions work the same way as they would for single board acquisition.

`CiAqSetup2Brds()` - Function sets up acquisition from two boards (i.e two DMA engines) into one buffer. Once this function is called, all other Ci acquisition control commands can be used as in single board applications. This function is illustrated in the CiView source code.

`BiCircAqSetup()` and `BiSeqAqSetup()` - These functions are not new, however, the Options parameter now takes two new flags (which can be ORed with other options if needed). The flags are `OnlyOddLines` and `OnlyEvenLines`. The idea is the two boards are opened and initialized normally, then acquisition is setup for both boards using the same set of buffers (either for sequence acquisition or for circular acquisition). However, for the first board, the `OnlyEvenLines` flag is used, and for the second board the `OnlyOddLines` flag is used. This technique programs both DMA engines to DMA to the same set of buffers, but each DMA engine only handles half of the lines. These functions are illustrated in the BiFlow2xsource code.