README for Bassler camera controller

by Conrad Foo, 20 Nov 2020

I have written this code for acquiring video (1280x1024) at high frame rates (>200Hz) for a prolonged period of time (20minutes - 1hr continuous, depending on machine specifications).

There are 3 files, divided into 2 parts:

1. baslerController and baslerProperties: These are basic objects that you can use to set parameters for acquisition. Most of the parameters that can be changed are in the baslerController params struct; for advanced users, the rest of the settings will be under baslerController.src. Configuring the GPIO ports and trigger sources is best done in pylon Viewer before opening MATLAB; the configuration will be preserved.

2. baslerGUI: This is a gui window that will provide a graphical interface to interact with the baslerController. The GUI should be self-explanatory but please let me know if you have any questions.

Usage:
```
controller = baslerController(vid);
gui = baslerGUI(controller);
```
where controller is the baslerController object and vid is the videoinput associated with your camera.

For example, if you have one basler USB camera: `vid = videoinput('gentl',1);` You may need to install the appropriate add-on from MATLAB.

To acquire data without the GUI:
```
start(controller.vid);
stop(controller.vid);
```

By default, the controller will write acquired frames to memory. To write to disk, you must instantiate the logger property of the controller as a VideoWriter object. Example:
```
myfile = VideoWriter('data.avi','Grayscale AVI');
controller.logger = myfile;
```
For high speed acquisition, an uncompressed format is best (grayscale AVI works well).

I can currently acquire at ~350Hz at 720x720 continuously for an indefinite amount of time; this requires an SSD and a relatively recent CPU. I am currently using an NVME M.2 SSD with an i7-7700 (3.5 GHz). The GUI will estimate the bandwidth necessary to acquire indefinitely based on acquisition frame rate and frame size. I find that keeping this number below ~185-200 MB/s is good enough for my system, but you should check to see what your bandwidth limit is. Above this, you will rapidly fill your system memory with buffered frames (you can calculate the memory necessary for acquisition parameters above your bandwidth limit based on the estimated bandwidth needed).

by Conrad Foo, 21 Nov 2020

Slight bug fix/QOL change for baslerController

The behavior of the hardware trigger is a bit unintuitive; FramesPerTrigger will run continuously after 1 software trigger, but in hardware triggering mode, you will need to trigger the acquisition of each frame. This might be dependent on your camera model, but the acA1300-200um I used does not have an option to hardware trigger multiple frames.

by Arash Fassihi Zakeri, 21 Nov 2020

You can trigger the initiation of the bursting. Using that method of acquisition, you can trigger n frames from the start of a single trigger pulse. Then you can read out the actual time of each frame by setting up your camera's digital output. I guess all Basler ace series should have similar features, but I might be wrong.
Arash