FNPL Digital Camera Readout System

Jason Wennerberg
Fermi National Accelerator Laboratory

August 11, 2004

Abstract

This note describes the software created to facilitate the readout of the FNPL digital camera system

1 Introduction

A new system is currently in development at the writting of this note to replace the current analog cameras of the FNPL beamline with digital cameras. The current status is that the cameras have been chosen to be SONY XCD-X710, and an apparatus has been designed and built to hold the cameras in place at the crosses. Software has also been written and will be the main focus of this paper.

2 Hardware Information

As stated before the cameras selected are the SONY XCD-X710. These cameras are black and white high resolution CCD cameras. They operate on the firewire IEEE1394 bus. They were selected because of their high resolution and previous dificulties with other manufacturers in the sytem used at DESY.

The cameras will be placed in the A0 cave and stretch firewire optical repeaters will be used to connect them to the Linux box in the control room that contains the firewire PCI card. There are now plans for 8 digital cameras.

3 Software Information

The software used to monitor the camera system is diveded into 4 parts. A comercial package called Coriander is used to adjust cameras specific settings such as shutter speed, gain and trigger. The control system for normal use is then divided into 3 parts: a server which reads the cameras, a cgi program that serves the images over the net, and a display program to view the images and save them.

3.1 Server

The server is based on a sample program that is included in the libdc1394 library called grab_grey_image. This program sets up and reads an image from a single camera over the firewire bus. The program was modified to become digital_server. It was modified to function in a continuous loop and to read from multiple cameras. The server reads out the cameras and places the .pgm (raw) image into the /tmp directory as /tmp/cam#.pgm where number is the index of the array starting at zero and continuing to number_of_cameras - 1.

3.2 CGI

The next component of the system is the cgi program. This program is a very small python script which is called pushthis.cgi. When invoked the script retrieves the .pgm file of interest and converts it to the file type you want and then sends it over the net. With this method wget or a similar program can be used in a script to retrieve the images.

3.3 Beamview Viewer Program

The final component of the system is called beamview.py. This program calls the cgi program to get the images and then displays them using standard python image libraries. The program also allows one to save the images and even do background subtraction and false color viewing.

3.3.1 Adding Cameras

Adding a camera to the list of available cameras can be done by modifying beamview.py. At about line 89 in the code you will find a section that looks like this:

```
cammenu = Menu(menubar, tearoff=0)
cammenu.add_command(label="Camera 0", command=self.set_cam_0)
cammenu.add_command(label="Camera 1", command=self.set_cam_1)
```

The cammenu.add_command command adds items to the menu. For example to add a third camera (Camera2) you would add the line:

```
cammenu.add_command(label="Camera 2", command=self.set_cam_2)
```

And then add the function set_cam_2 as discribed below.

Further down in the code at about line 187 you will find:

```
##Copy these and paste to add more cameras
def set_cam_0(self):
self.stop_update()
self.stop_update()
time.sleep(.5)
self.camera = 0
self.update_image()

def set_cam_1(self):
self.stop_update()
time.sleep(.5)
self.camera = 1
self.update_image()
```

To add camera 2 add the following code:

```
def set_cam_2(self):
self.stop_update()
```

```
self.stop_update()
time.sleep(.5)
self.camera = 2
self.update_image()
```

4 Further information

Further detail is available at

`http://home.fnal.gov/~jasonw/cameras/camhowto.html`

about how to install the system. The author is also available by email at jason_wennerberg@yahoo.com. Also the code is very well commented and should be read.

5 User Manual

To run this sytem one simply goes to the computer with the server software and starts it running with digital_server -q. The -q switch makes it run with no standard output. To stop the server from running one does a kill -ALRM *pid*. The server and the cgi program are located on a0pi-video1.fnal.gov. The server program is in /home/a0pi/argpush. Beamview is available on a0pi-spitefire6 and a0pi-spitefire2.