# KCU Online Monitoring – almost there

I ended up using my “barebones” project to plot the baseline-corrected  waveforms.

One project for both KCUs – it figures everything out by itself

At the core of the display is the plot ( offset, x , y ,…) that plots x times y starting from offset

It got some more optional parameters to be able to make multiple displays

For example, plot (0, 6,12) make a “default” canvas with 72 histos

But it has more –

plot ( 0, 6,12, “c”, “Channels 0-71”)

makes a TCanvas named “c” with the above title.

set_canvas_size(800,600);

sets the default Canvas size so you can set a default that fits your screen

# KCU Online Monitoring – almost there

I modified the auto-generated roc1.C so it starts the online monitoring

```
$ more roc1.C
#include "roc1.h"
R__LOAD_LIBRARY(./libroc1.so)

void roc1(const char * filename)
{
  if ( filename != NULL)
    {
      pfileopen(filename);
    }

  else
    {
      rcdaqopen();
      set_canvas_size(800,600);
      TPad *c;

      c = plot (0,6,12,"c", "ASIC 0 Channels 0-71");
      pupdate(c,15);
      c = plot (72,6,12,"d", "ASIC 0 Channels 72-143");
      pupdate(c,15);
      c = plot (2*72,6,12,"e", "ASIC 1 Channels 144-215");
      pupdate(c,15);
      c = plot (3*72,6,12,"f", "ASIC 1 Channels 216-287");
      pupdate(c,15);

      pstart();
    }
}
```

You can do

**`bash roc1.sh "filename"`**

and get what you always had (process the file)

Without a parameter this starts the onl. monitoring now

How do I get the KCU1 stream?

rcdaqopen() honors the RCDAQHOST setting

**`export RCDAQHOST=localhost:1`**

**`bash roc1.sh`**

And that's it

# KCU0 or KCU1?

Here is the simple logic:

```
Packet *p = 0;

  if ( e->existPacket(12001) )
    {
      p = e->getPacket(12001);
      KCU_STRING="KCU0";
    }
  else if ( e->existPacket(12002) )
    {
      p = e->getPacket(12002);
      KCU_STRING="KCU1";
    }

  if (p)
    {
      . . .
```

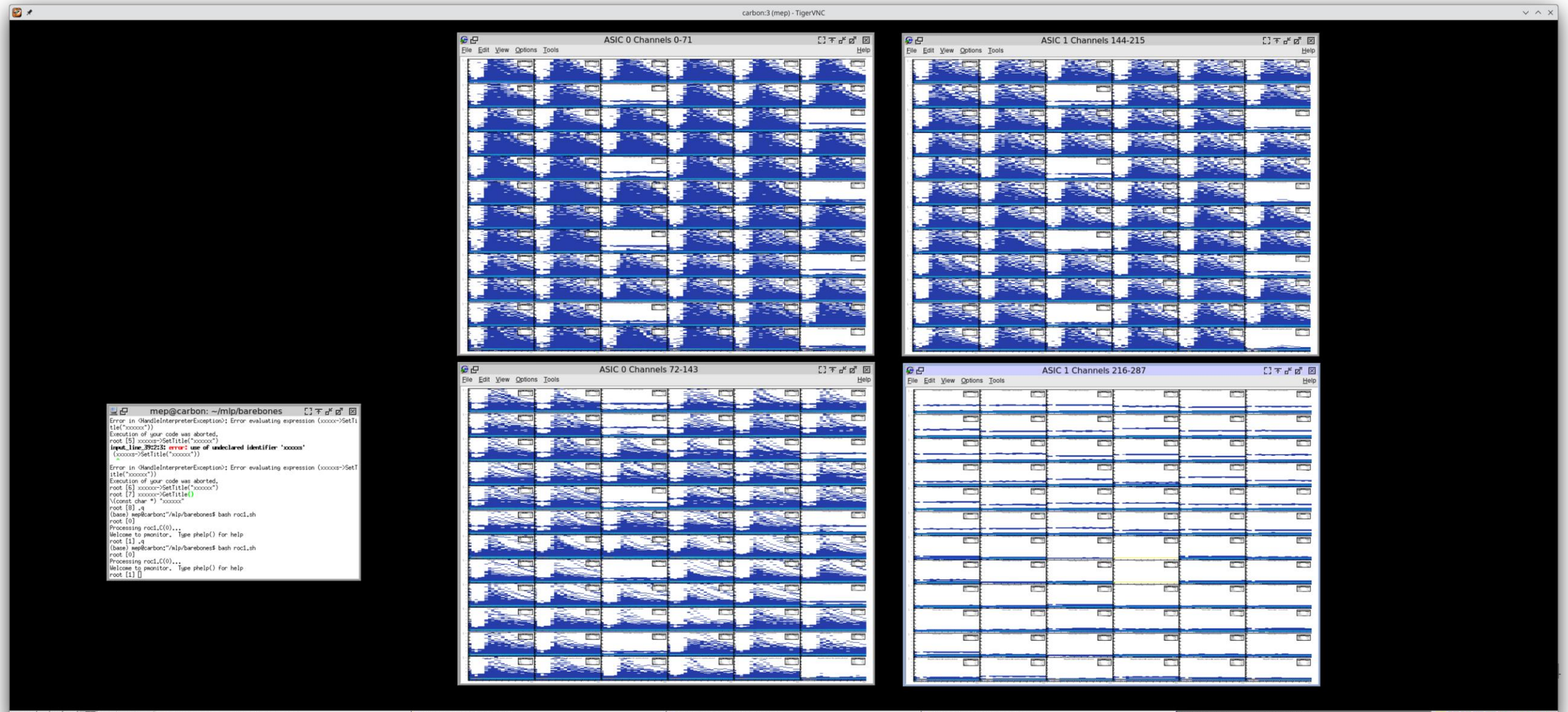
Later I'll be doing stuff with the KCU_STRING to name Canvases, save files, etc

# I’m using vnc :3 here

I’m getting results…

Currently:

desktop1 : KCU0  Desktop 2: KCU1



# Things done and left to do

- It's a *unified* monitor for both KCUs
- All histos reset on a new run begin
- Baseline correction appears to work
- Everything seems rock solid
- Startup sequence straightforward

- Saving Canvases/Histos at the end of a run
- A bit more proper naming (titles) of histos, canvases with KCU and runnumber
- Proper placement of the canvases; but that needs…
- some wrestling with the VNC as any resize will undo any fancy placement
- We don't have to stop at waveforms, can show processed values (signal, TOA, …)