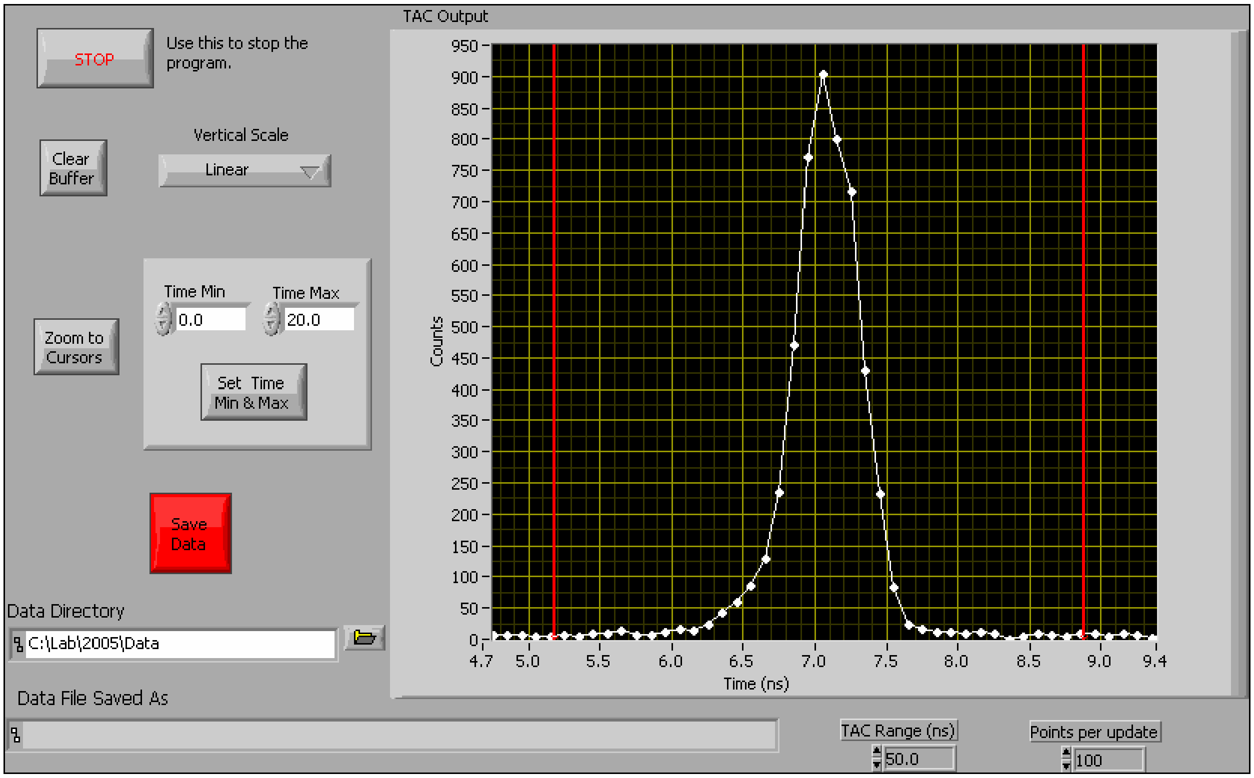
Appendix C - MCA.vi

This appendix describes the LabView vi used to mimic the behavior of a multichannel analyzer (MCA). This is useful for measuring the arrival time of photons.

MCA.vi

Front Panel



Description

This VI emulates a multi-channel-analyzer (MCA). An MCA essentially digitizes a series of voltage pulses, and creates a histogram of the pulse heights. The purpose of this VI is to analyze the voltage pulses from a Time-to-Amplitude Converter (TAC) [This VI was tested using the TAC in an ORTEC 567 TAC/SCA module].

The start and stop pulses going into the TAC come from photon counters, and the TAC output voltage is proportional to the time between the start and stop. Thus, the MCA is creating a histogram of the time interval between the arrivals of the two photons.

The main difference between this VI and a real MCA is that an MCA takes only one input-the voltage pulses. This VI also needs timing information about when the pulses are arriving. Fortunately, this can be obtained from the Valid Conversion output of the TAC. The Valid Conversion output is an approximately 3 microsecond long TTL pulse coincident with the TAC output pulse.

We have tested this VI with 2 different National Instruments multifunction cards: 6036E and 6052E. With these cards there are two clock signals that are needed to perform an A/D conversion: the trigger and the sample clock.

The trigger (input through PFI0) comes directly from the Valid Conversion output of the TAC. The program is configured to trigger on a low-to-high transition of the trigger -triggering arms the A/D, but does not actually cause an A/D conversion.

The program is configured to perform an A/D conversion on a high-to-low transition of the Sample Clock. The Sample Clock is obtained by generating a delayed pulse from the trigger. The delayed pulse is generated by one of the counter/timers on the multifunction board: counter 0. The output of counter 0 is sent directly to the sample clock, so the user need only input the trigger pulse to PFI0, and the sample clock will take care of itself (the user can monitor the counter 0 output to observe the timing of the sample clock). The timing can be changed by adjusting the duration of the pulse (using the “High Time”) parameter in the block diagram).

The TAC output to be digitized is input though Analog Input Channel 0.

Unlike many VI's you'll encounter, the screen updates after a specified number of data points are acquired (determined by the parameter "Points per update"), not after a specific time interval. Because of this, if the detectors are off, nothing will happen.

NOTE: The A/D boards we use are just barely fast enough to catch the 3 microsecond long TAC output (the width of the TAC output could also be increased for better results)--indeed, they're probably technically too slow. Because of this, the accuracy of the measurement of the TAC output is not very good (can be off by as much as 10 or 20%), which means that the time scale is poorly calibrated. However, the measurement is repeatable, so the time scale could be calibrated much more accurately. Despite this problem, the VI works quite well enough to accomplish its main goals-to display a bump in the histogram of arrival times between the photons from two detectors, and to allow one to set the SCA window to select coincidences (only needed if you’re using the TAC/SCA for coincidence counting).

ALSO NOTE: This vi has only been seriously tested for START-STOP times ranging from a few up to about 30 ns. With much longer intervals than this the output pulses may be delayed, and the timing may change.

Rough Timing diagram:

```
            ____________
 ________|             |___________   Trigger (PFI0)
                  __
 _____________|   |_______________   Sample clock (counter 0)
              ________
  __________/          \____________   A/D Ch 0
```

Help for each of the controls and indicators can be obtained from the Contextual Help window <cntrl - H> by mousing over each control or indicator. Full documentation for each control and indicator can be obtained by printing using: File>Print>Custom, and then checking "All controls" and "Descriptions"

FINAL NOTE: For whatever reason, the vi “aquire_daqmx_subvi.vi” does NOT get saved if you save the entire vi hierarchy. So, you explicitly need to copy this vi to the new library from the old one. After you copy it, LabView will STILL look for it in the old location, so you again need to explicitly find this vi for LabView.

<u>Controls and Indicators</u>

Stop

Use this to stop.

Clear Buffer

Clears the histogram memory, and the accumulation starts over again.

Vertical Scale

Change the scaling of the vertical scale. Options are Linear and Logarithmic.

Zoom to cursors

Horizontal scale will zoom in to the region specified by the cursor positions.

Time Min

Set minimum of horizontal axis to this value after pressing "Set Time Min & Max".

Time Max

Set maximum of horizontal axis to this value after pressing "Set Time Min & Max".

Set min & max

Set the minimum and maximum values of the horizontal axis to those specified.

Save Data

Save the current data to a file. Program exits after this is done.

Data Directory

Path to directory where the data will be saved.

Data File Saved As

Path to the data file. The data file is automatically named using the date and time.

TAC Range (ns)

The full scale range of the TAC output. Used to scale the horizontal axis.

Points per update

Number of data points acquired before updating the screen. Remember, the screen updates after a specified number of data points are acquired, not after a specific time interval.

If you have very low count rates, you might want to decrease this from it's default value of 100.

TAC Output

Histogram of the time interval distribution.