
Subject: Re: data acquisition, device control
Posted by [Gary Kushner](#) on Mon, 30 Jan 1995 17:01:26 GMT
[View Forum Message](#) <> [Reply to Message](#)

Richard Olsen <olsen@physics.nps.navy.mil> wrote:

>
> We'd like to use IDL as a front-end for data acquisition on a PC,
> which basically means being able to talk to an A-D converter, and
> presenting the results in ~real time with IDL
>
> Has anyone done this, or something similar?
>
> rc olsen
> olsen@physics.nps.navy.mil

We have also been using IDL to control several devices in our lab. We use the CALL_External function and have also used the OUTPW and INPW(?) calls. The OUTPW and INPW calls allow you to write and read ports and boards on a PC. You need to know the base addresses of the boards you want to communicate to.

We use CALL_EXTERNAL to control our GPIB devices and to collect data from a microchannel plate detector system. Our data acquisition system consists of a National Instruments GPIB card and a National Instruments digital I/O card. We wrote a windows dll that acts as a "wrapper" for the GPIB functions provided by National Instruments. For the digital I/O we required a fast data acquisition rate. We wrote a windows dll to directly control the I/O card and to collect the data. The dll then passes the data to IDL.

Dll's are not that difficult to write. Getting CALL_EXTERNAL to talk to your dll can be a little tricky though.

The main problem we encountered was that a windows 3.1x dll can only pass 64k of data at a time. Thus, if you wish to acquire a large amount of data, you have to write your IDL code and your dll to pass the data in small chunks or you could use interrupt driven code. I am thinking of switching to Windows NT so that I can have the process run under a different thread and so I can ignore the 64k limit.

If anyone is interested, I can make available our IDL code and our dll source code.
