Subject: data acquisition, device control
Posted by Richard Olsen on Thu, 26 Jan 1995 19:01:56 GMT
View Forum Message <> Reply to Message

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

Subject: Re: data acquisition, device control
Posted by rivers on Sat, 28 Jan 1995 21:28:44 GMT
View Forum Message <> Reply to Message

In article <3g8rj4$smc@nps.navy.mil>, Richard Olsen <olsen@physics.nps.navy.mil> writes:
> 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?
>
We do this in our projects at the National Synchrotron Light Source and many of
the experimental groups building x-ray beamlines at the Advanced Photon Source
at Argonne National Lab plan to use IDL as an important component of the
operator interface for data acquisition and display.

It is all done with CALL_EXTERNAL. Many of these projects have two "threads"
running. A real-time thread written in C handles the fast data acquisition and
interrupt handling. The IDL thread gets data from the C thread and displays it.
The IDL thread is widget driven and allows the operator to control the
acquisition, display and data storage.

_____

Mark Rivers (312) 702-2279 (office)
CARS (312) 702-9951 (secretary)
Univ. of Chicago (312) 702-5454 (FAX)
5640 S. Ellis Ave. (708) 922-0499 (home)
Chicago, IL 60637 rivers@cars3.uchicago.edu (Internet)

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.

Subject: Re: data acquisition, device control
Posted by kaaret on Thu, 02 Feb 1995 18:18:47 GMT
View Forum Message <> Reply to Message

We do the same thing here at Columbia. We have a data input over a serial line.
The code to handle the serial line is in C and IDL gets the data via call external.
It works rather well. Writing the user interface in IDL is much easier than
writing an equivalent one in C.
