
Subject: Re: call_external with PASCAL calling convention

Posted by [rivers](#) on Tue, 06 Oct 1998 07:00:00 GMT

[View Forum Message](#) <> [Reply to Message](#)

In article <3614D696.8B3EDE4C@mssl.ucl.ac.uk>, "E. Breeveld" <erb@mssl.ucl.ac.uk> writes:

> Any call_external gurus out there?

>

> We have a windows DLL that interfaces with a PC card and

> we'd like it to be incorporated (via call_external) into IDL, but

> can't make it work. The equivalent C program does work however!

> We are using IDL Version 5.1 Win32 (x86) with windows 98.

...

> The test idl program is:

>

> card = 0b ; Byte

> basadd2 = 0 ; Int

> irq1 = 0b ; Byte

> ; test without cdecl

> result = call_external('7200.dll', 'W_7200_Initial', \$

> card, basadd2, irq1, val=[1, 0, 0])

> help, result, basadd2, irq1

> end

>

> The C program that works looks like:

>

> #include <windows.h>

> #include <stdio.h>

> #include "c:\7200\include\7200.h"

>

> unsigned char irq1;

> unsigned short base_address2;

> long status4;

>

> void main(void)

> {

> status4 = W_7200_Initial(0, &base_address2, &irq1);

> printf("Status= %ld\n"

> "Base Address= 0x%X\n"

> "IRQ= %d\n", status4, base_address2, irq1);

> }

...

> We could write a C wrapper for 7200.dll, but this is messy. Is there

> any other solution, or is there a bug in IDL with the /cdecl switch?

The problem here is clear: you cannot call this function W_7200_Initial directly from IDL because it does not use the (argc, argv) argument passing mechanism which IDL uses with CALL_EXTERNAL. You must write a wrapper routine to convert IDL's (argc, argv) calling convention to the parameter list your

function needs.

Mark Rivers	(773) 702-2279 (office)
CARS	(773) 702-9951 (secretary)
Univ. of Chicago	(773) 702-5454 (FAX)
5640 S. Ellis Ave.	(708) 922-0499 (home)
Chicago, IL 60637	rivers@cars.uchicago.edu (e-mail)

or:

Argonne National Laboratory	(630) 252-0422 (office)
Building 434A	(630) 252-0405 (lab)
9700 South Cass Avenue	(630) 252-1713 (beamline)
Argonne, IL 60439	(630) 252-0443 (FAX)
