

Re: How to call function from driver in inline assembler

Source:

<http://www.tech-archive.net/Archive/Development/microsoft.public.development.device.drivers/2006-05/msg00928.1>

- *From:* Peter <Peter@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxx>
 - *Date:* Wed, 31 May 2006 04:50:01 -0700
-

Yes, it is not code,
but item in IAT.
:-)

Peter

"Peter" wrote:

I dont understand this.
What is good this small part of code called "import descriptor" for, when it cannot be called ?

"Skywing" wrote:

You're calling the import descriptor (a pointer) for that function instead of calling through it. Try perhaps 'call dword ptr [ExAllocatePoolWithTag]' or something along the lines of that.

"Peter" <Peter@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxx> wrote in message <news:2C76FCB2-86BB-47D1-B480-C1D6A79FEC26@xxxxxxxxxxxxxxxxxxxx>

I see difference between ExAllocatePoolwithTag caled normally like C function and called from _asm {} . In C case, there is indirect call, instruction is 6 byte and begins with FF 15 . In _asm {} there call instruction is 5 bytes and begins with E8 . When I step into call compiled in _asm {} block (described in previous mail),

Re: How to call function from driver in inline assembler

it jumps into small part of code which begins with symbol
`__imp__ExAllocatePoolWithTag` .

First instruction in this code is:

```
test [ebx],dh
```

(All I am trying in release version built with debug info)

On this instruction blue screen

`PAGE_FAULT_IN_NONPAGED_AREA` occurs.

It seems that in calling by this way must be prepared ebx register ?

Or not possible to call other functions than are part of calling driver ?

Yes I agree with you that using inline `_asm` is not very effective

I need it only very seldom, it is calculated that it will be used only in 32-bit driver build.

Peter

"Steve Dispensa" wrote:

You shouldn't use inline asm, especially for something as trivial as this, unless you're really just playing around. The 64-bit compilers don't have inline asm support at all, and in general, it creates a maintenance headache and (unwanted?) job security for the developer.

"Failed" is ambiguous; is it building? What happens when you single step it in a debugger?

-Steve

On 5/29/06 3:50 AM, in article EFE28483-3D53-42D8-9DEB-1701CB5C180E@xxxxxxxxxxxxxx, "Peter" <Peter@xxxxxxxxxxxxxxxxxxxxxxxxxxxx> wrote:

In my driver I had tried to call two functions from `_asm { }` block.

Re: How to call function from driver in inline assembler

Calling of function that is part of this driver was OK. But calling for example `ExAllocatePoolWithTag` failed on call instruction. I tried this:

```
push 0x39393939
push 0x48c
push 0
call
ExAllocatePoolWithTag
mov resultAddr, eax
```

Is possible in driver to call in `_asm{ }` block functions which are not linked in the same driver ?

Peter