Why does the x86 have so few registers?

devblogs.microsoft.com/oldnewthing/20040105-00

January 5, 2004

Raymond Chen

One of the comments to my discussion of 16-bit calling conventions wondered why the 8086 had so few registers.

The 8086 was a 16-bit version of the even older 8080 processor, which had six 8-bit registers, named A, B, C, D, E, H, and L. The registers could be used in pairs to products three 16-bit pseudo-registers, BC, DE, and HL. What's more, you could put a 16-bit address into the HL register and use the pseudo-register "M" to deference it. So, for example, you could write "MOV B, M" and this meant to load the 8-bit value pointed to by the HL register pair into the B register.

The 8086 took these 8080 registers and mapped them sort of like this:

- A -> AL
- H -> BH, L -> BL; HL -> BX; M -> [BX]
- B -> CH, C -> CL; BC -> CX
- D -> DH, E -> DL; DE -> DX

This is why the 8086 instruction set can only dereference through the [BX] register and not the [CX] or [DX] registers: On the original 8080, you could not dereference through [BC] or [DE], only thorugh M=[HL].

This much so far is pretty official. The instruction set for the 8086 was chosen to be upwardly-compatible with the 8080, so as to facilitate machine translation of existing 8-bit code to this new 16-bit processor. Even the MS-DOS function calls were designed so as to faciliate machine translation.

What about the SI and DI registers? I suspect they were inspired by the IX and IY registers available on the Z-80, a competitor to the 8080 which took the 8080 instruction set and extended it with more registers. The Z-80 allowed you to dereference through [IX] and [IY], so the 8086 lets you dereference through [SI] and [DI].

And what about the BP register? I suspect that was invented on the fly in order to facilitate stack-based parameter passing. Notice that the BP register is the only 8086 register that defaults to the SS segment register and which can be used to access memory directly.

Why not add even more registers, like today's processors with their palette of 16 or even 128 registers? Why limit the 8086 to only eight registers (AX, BX, CX, DX, SI, DI, BP, SP)? Well, that was then and this is now. At that time, processors did not have lots of registers. The 68000 had a whopping sixteen registers, but if you look more closely, only half of them were general purpose arithmetic registers; the other half were used only for accessing memory.

Raymond Chen

Follow