

Standing on the shoulders of giants: Let the compiler tell you what the ABI is

 devblogs.microsoft.com/oldnewthing/20220112-00

January 12, 2022



Raymond Chen

A necessary (but not sufficient) condition for verifying that your handwritten assembly language conforms to the ABI is to compare it to something that definitely conforms to the ABI, namely, a compiler for that ABI.

In my various historical CPU architecture series, I had to figure out the calling convention. Sometimes I was able to find documentation. Sometimes I had to infer documentation from comments. And sometimes I just had to treat the compiler as an oracle: To see what the calling convention is for a particular scenario, code up that scenario, send it through the compiler, and see what comes out.

The compiler is what created the operating system, so even if the compiler doesn't follow the *de jure* ABI, what it produces is the *de facto* ABI.

Mind you, matching the compiler output is a necessary but not sufficient condition. There may be other things that are part of the ABI (like exception unwind codes and stack alignment) which are not immediately visible in the compiler output. But at least you'll know that you at least got the parameters in the right place.

[Raymond Chen](#)

Follow

