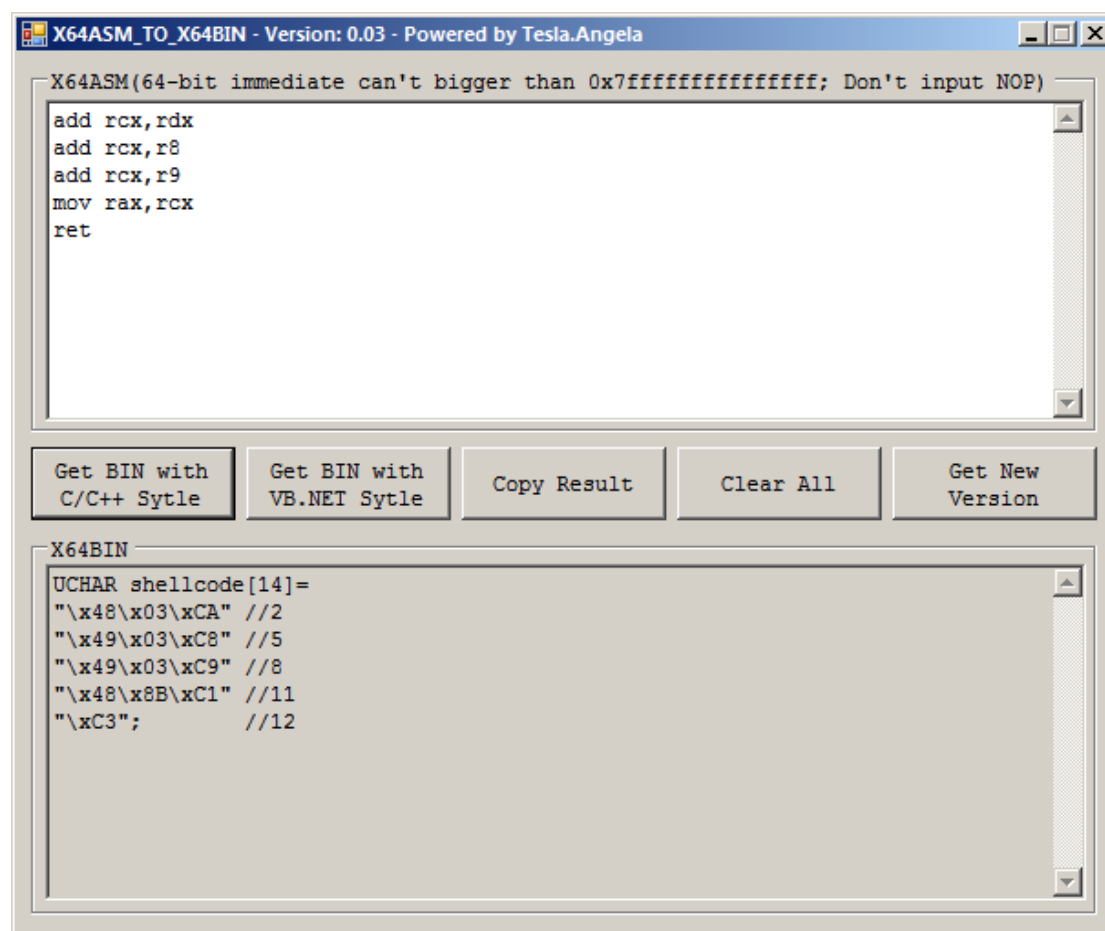


有些操作非要使用汇编语言不可，比如想获取 CPU 的信息，必须用 CPUID 指令，但是 64 位的微软编译器都不再允许内嵌汇编了，这可怎么办呢？标准方法是使用 ASM 文件和 C 文件进行混合编译，但这种方法很麻烦，特别是只要嵌入一小段汇编的情况下。所以我的解决方法是，用软件把汇编转换成机器码，然后直接执行机器码。64 位汇编转换机器码的工具可以用我的 X64ASM_TO_X64BIN（<http://www.vbasm.com/thread-5651-1-1.html>）。

先来说说如何使用 X64ASM_TO_X64BIN。要使用 X64ASM_TO_X64BIN 必须先下载 MASM64 压缩包（<http://www.m5home.com/bbs/thread-5170-1-1.html>）。下载完毕后，必须把此压缩包解压到 C 盘根目录（即解压完毕后 C 盘有个 MASM64 文件夹）；其次，必须安装好 .NET FRAMEWORK 4（<http://www.microsoft.com/zh-cn/download/details.aspx?id=17718>），否则软件无法运行。这两步都配置好后，输入汇编代码，按下『Get BIN with C/C++ Style』按钮，即可把汇编代码转换为机器码（SHELLCODE）：

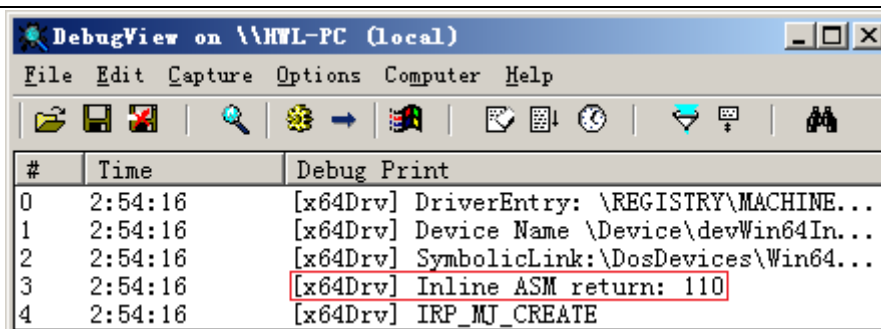


接下来说说如何让内嵌的汇编子程序有返回值以及给汇编子程序传入参数。首先是使用 typedef 定义一下你的汇编子程序的原型，然后把机器码复制到 NonPagedPool 里，最后直接把 NonPagedPool 的地址当作函数来 CALL。代码示例如下（输入四个数字相加，返回它们相加的值）：

```
typedef UINT64 (__fastcall *SCFN) (UINT64, UINT64, UINT64, UINT64);

VOID test()
{
```

```
SCFN scfn;
UINT64 ret;
UCHAR strShellCode[14]="\x48\x03\xCA\x49\x03\xC8\x49\x03\xC9\x48\x8B\xC1\xC3";
/*
add rcx,rdx
add rcx,r8
add rcx,r9
mov rax,rcx
ret
*/
scfn=ExAllocatePool(NonPagedPool,14);
memcpy(scfn,strShellCode,14);
ret=scfn(11,22,33,44);
DbgPrint("[x64Drv] Inline ASM return: %lld",ret);
ExFreePool(scfn);
}
```



#	Time	Debug Print
0	2:54:16	[x64Drv] DriverEntry: \REGISTRY\MACHINE...
1	2:54:16	[x64Drv] Device Name \Device\devWin64In...
2	2:54:16	[x64Drv] SymbolicLink:\DosDevices\Win64...
3	2:54:16	[x64Drv] Inline ASM return: 110
4	2:54:16	[x64Drv] IRP_MJ_CREATE

本文到此结束。示例代码在附件里。