
Subject: Troubles compiling code with -fPIC
Posted by [dolik.rce](#) on Sun, 30 Jun 2013 19:38:38 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi everyone,

I've hit a problem when trying to compile a shared library using U++ with -fPIC
switch:/home/h/upp-production/uppsrc/Core/Cpu.cpp: In function 'void Upp::sCheckCPU()':
/home/h/upp-production/uppsrc/Core/Cpu.cpp:40:84: error: PIC register clobbered by '%ebx' in
'asm'

__asm__("movl \$1, %%eax\n\tcpuid" : "=d" (info1), "=c" (info2) : : "%eax", "%ebx");
As the error message says, the problem is caused by changing a value of ebx register. After a little googling, I found that the widely used fix in this situation is to store the value of ebx before calling cpuid instruction and restore it afterwards. When I looked into the code of SCheckCpu() to do that, to my great surprise I found the code was already there, just commented out

So my question is, does anyone remember if there was some valid reason to comment this line out? If not, I propose to start using it: Index: /home/h/upp-production/uppsrc/Core/Cpu.cpp

```
=====
--- /home/h/upp-production/uppsrc/Core/Cpu.cpp (revision 6151)
+++ /home/h/upp-production/uppsrc/Core/Cpu.cpp (working copy)
@@ -15,7 +15,6 @@
     if(done) return;
     done = true;
#ifdef PLATFORM_OSX11
-// __asm__("pushl %%ebx\n\tmovl $1, %%eax\n\tcpuid\n\tpopl %%ebx" : "=d" (info1), "=c"
(info2) : : "%eax");
    sHasMMX = true;
    sHasSSE = true;
    sHasSSE2 = true;
@@ -37,7 +36,7 @@
#else
    dword info1;
    dword info2;
-    __asm__("movl $1, %%eax\n\tcpuid" : "=d" (info1), "=c" (info2) : : "%eax", "%ebx");
+    __asm__("pushl %%ebx\n\tmovl $1, %%eax\n\tcpuid\n\tpopl %%ebx" : "=d" (info1), "=c"
(info2) : : "%eax");
#endif
    sHasMMX = ((info1 >> 23) & 0x1);
    sHasSSE = ((info1 >> 25) & 0x1);
```

With this patch my code compiles and I noticed no difference for other programs.

Best regards,
Honza
