编译kernel出现no machine record defined 错误，网上有一些解法，其实都是错误的，以讹传讹。不打算自己写，找到一篇还算靠谱的，转摘一下。

其根本原因是没有在 \_\_proc\_info\_end 与 \_\_proc\_info\_begin 之间找到有效的machine ID。

<http://hi.baidu.com/rwen2012/item/9f5191c64036d87ecfd4f849>

在新的kernel 2.6.29 上编译不能通过，终端提示：   
  
arm-none-linux-gnueabi-ld: no machine record defined   
  
解决方法   
放狗搜后，按照如下方法可以解决。将arch/arm/kernel/vmlinux.lds的最后两行(如下)，给注释起来，但都没说是为了什么   
  
ASSERT((\_\_proc\_info\_end - \_\_proc\_info\_begin), "missing CPU support")   
ASSERT((\_\_arch\_info\_end - \_\_arch\_info\_begin), "no machine record defined")   
  
  
自己到arch/arm/kernel/vmlinux.lds里看的时候，才发现那两行的上头写着   
  
/\* These must never be empty   
If you have to comment these two assert statements out, your binutils is too old (for other reasons as well) \*/  
ASSERT((\_\_proc\_info\_end - \_\_proc\_info\_begin), "missing CPU support")   
ASSERT((\_\_arch\_info\_end - \_\_arch\_info\_begin), "no machine record defined")   
  
  
我想可能不是我交叉编译器太旧的缘故，而是太新了！   
我用的是：Sourcery G++ Lite 2009q1-126 for ARM GNU/Linux   
而官方是：Sourcery G++ Lite 2008q1-126 for ARM GNU/Linux   
  
机器ID错误：   
现在注释后可以正常编译内核了！替换新的内核文件，在boot后提示：   
uncompressing linux .................................................   
然后再没有动静了，我想可能解压后给kernel传递参数时出现问题了吧。但是是什么参数可能出的问题就不晓得啦。   
  
在这里困住了好久！因为这句话啥都没提示！好歹有句话啊。。。。。。。。   
  
经过多方搜索，发现了一个调试技巧，嘿嘿嘿，一般人我不告诉他！那就是下面解决方法中提到的设置Kernel debugging，设置后，再重启启动内核就会出现详细的问题说明。  
  
  
  
因为内核启动时首先进入SVC模式，关闭中断，并machine ID，如果uboot传输过来的 machine ID跟编译的内核machine ID不一致，那就就咯屁了。具体表现为：啥反应都没有！  
  
解决方法：   
1）。执行make xconfig   
  
在配置中进入Kernel hacking,打开Kernel debugging和Kernel low-level debugging functions.   
  
选中这两项的目的是打印出调试信息，重新make uImage   
  
Starting kernel ...   
  
Uncompressing Linux........................................................................................... done, booting the kernel.  
  
Error: unrecognized/unsupported machine ID (r1 = 0x000007d1).   
  
Available machine support:   
  
ID (hex) NAME   
  
罓?0009   
  
Please check your kernel config and/or bootloader.   
  
  
终于看到阻在这里的原因啦，是机器ID错误，别人的机器ID错误至少有个NAME值啊，我的咋为空呢？   
  
2） 是不是没有注册机器ID？猜测。这一步保证系统支持的ID里面含有devkit8000！   
  
make xconfig 发现system type菜单下面没有devkit的选项，肯定是哪里的配置文件还没有添加，经过分析，找到了/kernel/arch/arm/mach-omap2/Kconfig文件，在末尾配置添加了devkit8000支持。  
  
config MACH\_OMAP3\_BEAGLE   
  
bool "OMAP3 BEAGLE board"   
  
depends on ARCH\_OMAP3 && ARCH\_OMAP34XX   
  
config MACH\_OMAP3\_DEVKIT8000   
  
bool "OMAP3 DEVKIT8000 board"   
  
depends on ARCH\_OMAP3 && ARCH\_OMAP34XX   
  
  
然后make xconfig 发现system type菜单，选择devkit选项，选devkit8000，保存。   
  
3）上一步仅仅能保证可以支持，要想能正确支持，还要修改下面的文件。   
  
1. 我们进到arch/arm/mach-omap2/board\_devkit8000.c，在最后一段有这句   
  
MACHINE\_START(OMAP3\_DEVKIT8000 , ”OMAP3 beggle board”)   
  
  
这里OMAP3\_DEVKIT8000就是machine ID的代号~ 呢具体值是多少呢?~   
  
2. 在文件arch/arm/tools/mach-types中最后一行添加，设置我们的machine ID是2001。   
  
omap3\_devkit8000 ARCH\_OMAP3\_DEVKIT8000 OMAP3\_DEVKIT8000 2001   
  
  
通过修改Kconfig和mach-types两个文件，我们可以使系统正确的识别我们的machine ID，并且从上面我们也可以看到r1= 0x000007d1，0x7d1就是2001。与我们所设置的machine ID是相一致的。