第一步：下载linux源码包和xenomai的源码包。放入到自己喜欢的路径下。一般来说，linux是建议放到/usr目录下的src目录下。不过都无所谓。

第二步:将xenomai的补丁代码打入到linux源码路径中。

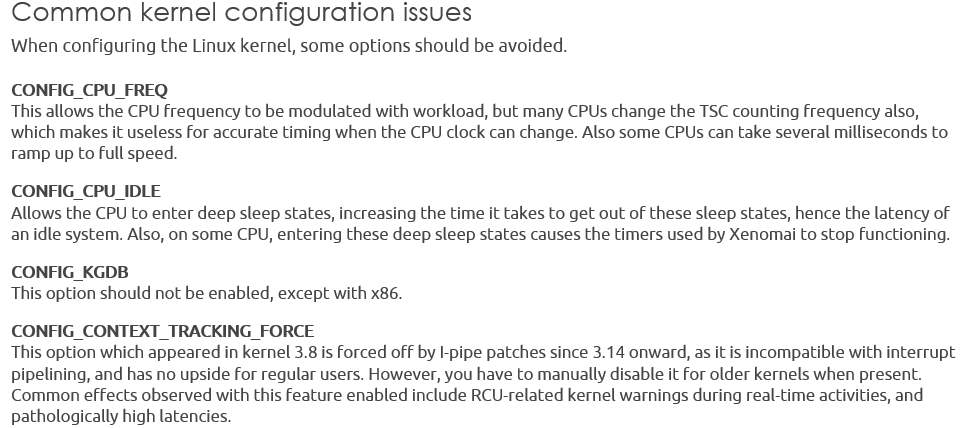
命令：scripts/prepare-kernel.sh [--linux=<linux-srctree>]

[--ipipe=<ipipe-patch>] [--arch=<target-arch>]

这里面linux-srctree就是下载并解压出来的linux源码根路径。

现在的例子中，linux解压在/home/robot/linux-3.18.20/中。这个参数就是/home/robot/linux-3.18.20。

第三步：编译Linux内核。和一般编译linux内核一样。官网上给了内核3.8版本后的需要把几个参数关掉，不然有可能会有问题。其中特别提到了**CONFIG\_CONTEXT\_TRACKING\_FORCE**这个参数。不取消掉的话，会产生预期不到的大延迟现象。



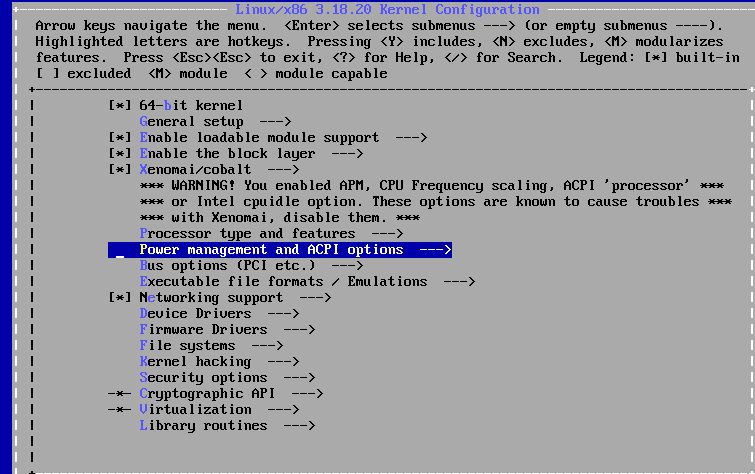
进入到linux根路径。

Make menuconfig

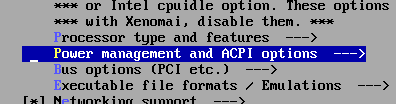
找到该参数

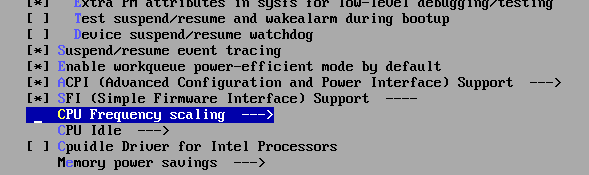
Disable it

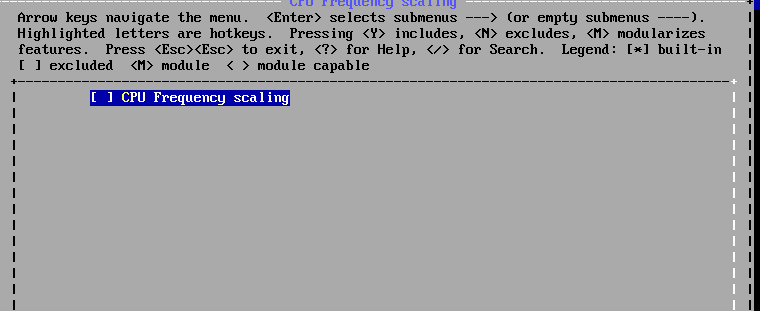
**Xenomai本身会在menuconfig里面提示一些会出现错误的选项警告。例如：**

****

**上面提示的3个参数，cpu frequency scaling:**

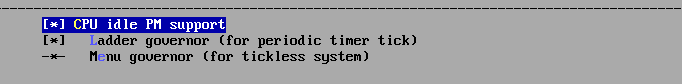
**在**

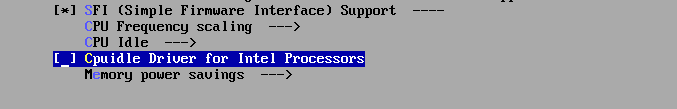
****



Diable it 没商量。

这个CPU idle我应该也是禁掉了。





这个也要禁掉。

基本上xenomai自己不报warning, 感觉应该就是没问题。

退出。

Make bzImage

Make install

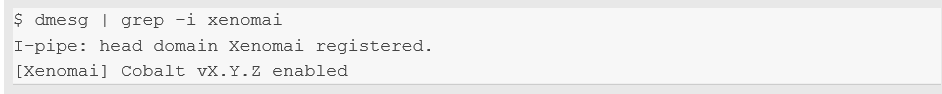
Make modules

Make modules\_install

全部通过的话，linux内核就完成了。

重新启动系统。

第四步:查看xenomai内核是否生效。Dmesg| grep xenomai 会看到内核打印的信息。

这样就说明xenomai已经在运行了。

第五步：编译xenomai工具库。在xenomai的源码根路径下执行配置命令

./configure –with-core=cobalt –enable-smp –enable-pshared

Make install

完成后会在/usr/xenomai/bin路径下生成一些可执行程序。执行latency.查看抖动效果。

可恨的是，配置好重启以后，TMD不能上网了。而且怎么也弄不好了！！！