yocto系统为我们提供了很好的制作嵌入式linux基础镜像的途径，yocto默认采用分层结构来组织所有的软件包。下面介绍一下如何在yocto上创建一个层以及如何使用该层。我们的目标是向linux内核源代码打patch，我们不希望去修改yocto目前已有的层，我们自己创建一个层来实现对linux内核打patch的工作，这样即使yocto的linux内核层在以后的版本中出现变更也不会影响到我们自己创建的层。

1、生成linux patch文件。作为例子我们向linux内核的init/calibrate.c文件中添加开机启动打印信息，具体patch文件如下：

diff --git a/init/calibrate.c b/init/calibrate.c  
index fda0a7b..01e3a5f 100644  
--- a/init/calibrate.c  
+++ b/init/calibrate.c  
@@ -265,6 +265,12 @@ void \_\_cpuinit calibrate\_delay(void)  
        static bool printed;  
        int this\_cpu = smp\_processor\_id();  
   
+       printk("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");  
+       printk("\*                                   \*\n");  
+       printk("\*        HELLO YOCTO KERNEL         \*\n");  
+       printk("\*                                   \*\n");  
+       printk("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");  
+  
        if (per\_cpu(cpu\_loops\_per\_jiffy, this\_cpu)) {  
                lpj = per\_cpu(cpu\_loops\_per\_jiffy, this\_cpu);  
                if (!printed)  
该patch名称为0001-calibrate-Add-printk-example.patch。

2、在poky同级目录下创建一个新目录meta-mylayer，并且在该目录下生成conf, recipes-kernel/linux/linux-yocto目录结构；

3、在meta-mylayer/conf目录下创建新层的配置文件layer.conf，layer.conf的具体内容如下：

# We have a conf and classes directory, add to BBPATH  
BBPATH .= ":${LAYERDIR}"  
  
# We have recipes-\* directories, add to BBFILES  
BBFILES += "${LAYERDIR}/recipes-\*/\*/\*.bb \  
            ${LAYERDIR}/recipes-\*/\*/\*.bbappend"  
  
BBFILE\_COLLECTIONS += "mylayer"  
BBFILE\_PATTERN\_mylayer = "^${LAYERDIR}/"  
BBFILE\_PRIORITY\_mylayer = "5"

4、在meta-mylayer/recipes-kernel/linux/目录下生成linux-yocto\_3.10.bbappend文件，该文件用于通知bitbake有新的内容要加载到linux-yocto编译过程中，具体linux-yocto\_3.10.bbappend文件内容如下：

FILESEXTRAPATHS\_prepend := "${THISDIR}/${PN}:"  
  
SRC\_URI += "file://0001-calibrate-Add-printk-example.patch"

5、将生成的.patch文件放在meta-mylayer/recipes-kernel/linux/linux-yocto/目录下；

到此，新建的层已经完毕。该层的具体目录结构如下：

meta-mylayer/  
├── conf  
│   └── layer.conf  
└── recipes-kernel  
    └── linux  
        ├── linux-yocto  
        │   └── 0001-calibrate-Add-printk-example.patch  
        └── linux-yocto\_3.10.bbappend  
6、使能新加入的层。修改build/conf/bblayer.conf文件，将新加入的层添加到bblayer.conf文件中，即：

BBLAYERS += " ${BSPDIR}/sources/meta-mylayer"

7、重新编译打patch的源代码包。

    # bitbake -c cleansstate linux-yocto

    # bitbake -k linux-yocto

8、验证打patch后的内核。

    # runqemu qemux86

    # dmesg | less

到此，在yocto上如何添加一层介绍完毕。

来源： <http://my.oschina.net/linuxhunter/blog/341756>