# 一、创建VSCode工程

 1、将NXP官方的linux内核拷贝到Ubuntu。

 2、解压缩tar -vxjf linux-imx-rel\_imx\_4.1.15\_2.1.0\_ga.tar.bz2

# 二、NXP官方开发板Linux内核编译

 编译NXP官方EVK开发板对应的Linux系统，默认配置文件存放路径arch/arm/configs。最终编译出：zImage，和 imx6ull-14x14-evk-emmc.dtb，imx6ull-14x14-evk.dtb。

 将zImage，和 imx6ull-14x14-evk-emmc.dtb拷贝到tftpboot目录下，然后在uboot中通过tftp服务启动。

 经过测试，发现NXP官方开发板对应的zImage和.dtb可以在正点原子开发板上启动。



Linux启动遇到上述错误，那是因为没有跟文件系统。

# 三、使能8线EMMC

 修改设备树，imx6ull-alientek-emmc.dts。节点usdhc2。

# 四、网络驱动修改

 为何要一开始就修改网络驱动？？在做Linux驱动开发的时候一般都是通过网络调试的。

 修改网络复位IO和PHY ID以后，Linux内核内部通用PHY驱动已经正常工作。

 LAN8720的生产厂家是SMSC。使能SMSC驱动，然后重新编译Linux内核，并启动，tftp启动。

# 三、在Linux中添加自己的开发板

 2、一个就是imx6\_alientek\_emmc\_defconfig默认配置文件。

 3、imx6ull-alientek-emmc.dts编译出来就是.dtb文件。

 复制arch/arm/configs/imx\_v7\_mfg\_defconfig文件为imx\_alientek\_emmc\_defconfig。

 复制arch/arm/boot/dts/imx6ull-14x14-evk.dts文件为imx6ull-alientek-emmc.dts

 修改arch/arm/boot/dts/Makefile，

# 四、CPU主频和网络驱动修改

 1、修改驱动之前，要先保证板子能够正常启动。

 2、根文件系统处理好，使用现成根文件系统。保证EMMC烧写了系统，设置bootcmd和bootargs。

 Bootcmd设置默认从网络启动，通过TFTP。

Setenv bootcmd ‘tftp 80800000 zImage;tftp 83000000 imx6ull-alientek-emmc.dtb;bootz 80800000 - 83000000;’

 Bootargs设置，根文件系统存放在EMMC的分区2里面，命令如下：

Setenv bootargs ‘console=ttymxc0,115200 root=/dev/mmcblk1p2 rootwait rw’

 现在的问题就是EMMC驱动有问题，在imx6ull-alientek-emmc.dts中找到usdhc2节点。将usdhc2改为：

&usdhc2 {

 pinctrl-names = "default", "state\_100mhz", "state\_200mhz";

 pinctrl-0 = <&pinctrl\_usdhc2\_8bit>;

 pinctrl-1 = <&pinctrl\_usdhc2\_8bit\_100mhz>;

 pinctrl-2 = <&pinctrl\_usdhc2\_8bit\_200mhz>;

 bus-width = <8>;

 non-removable;

 status = "okay";

};

完了以后编译设备树。使用如下命令：

Make dtbs。

 1、配置主频

 超频到696MHz，NXP官方宣传为700MHz。打开imx6ull.dtsi文件，