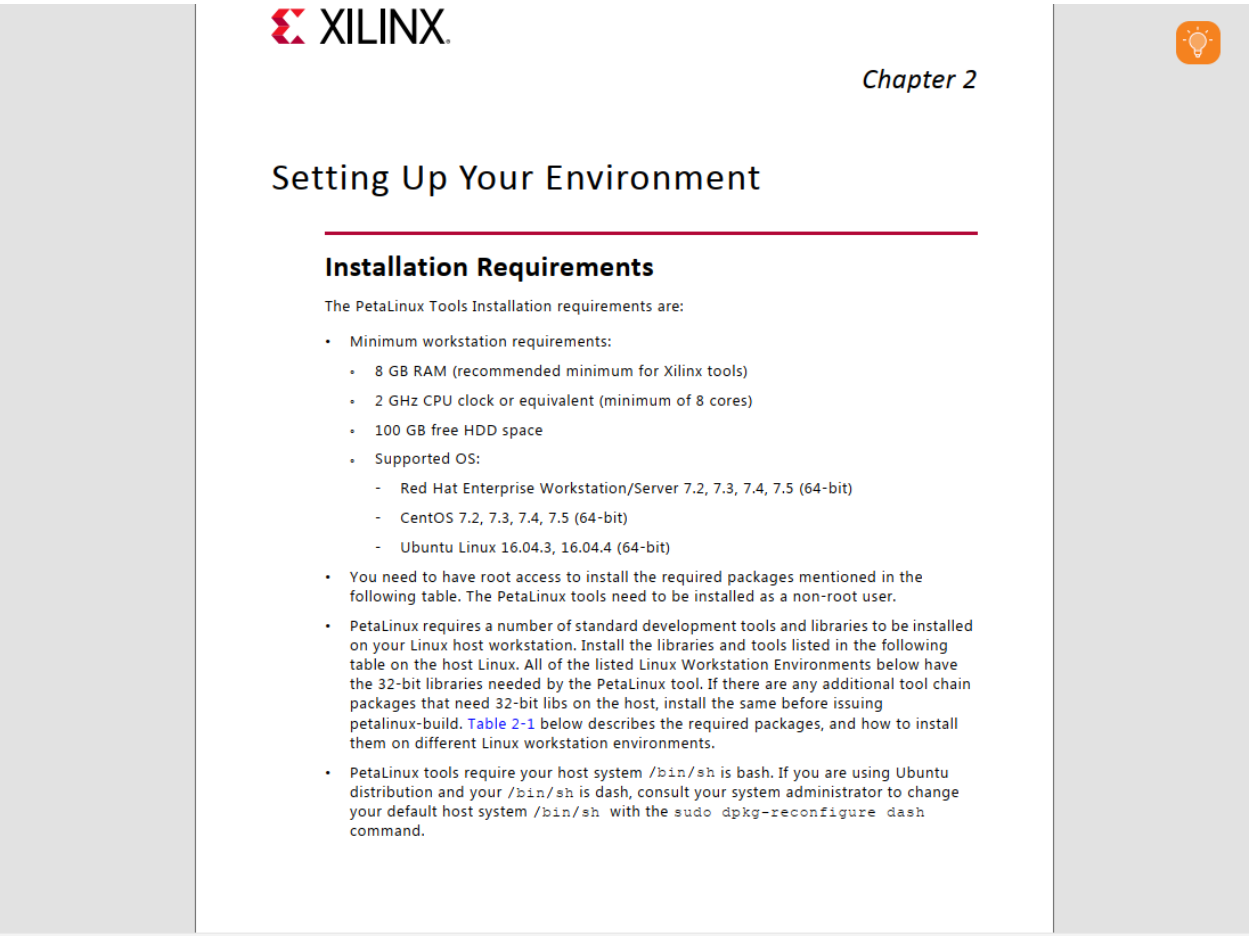
[L14]最小PetaLinux启动

PetaLinux比较挑系统,因此只能按着他游戏规则来,因为里面涉及的Yocto比较挑,推荐使用和Vivado匹配的PetaLinux版本,小版本号比较难以匹配也没办法,比如我只能用Ubuntu 16.04的最后一个LTS,另外2018.3是最后一个依赖hdf的,之后的依赖xsa,所以暂时选2018.3.

Xilinx文档编号:UG1144 - 必须先阅读文档再看这个笔记.



XILINX

Chapter 2

Setting Up Your Environment

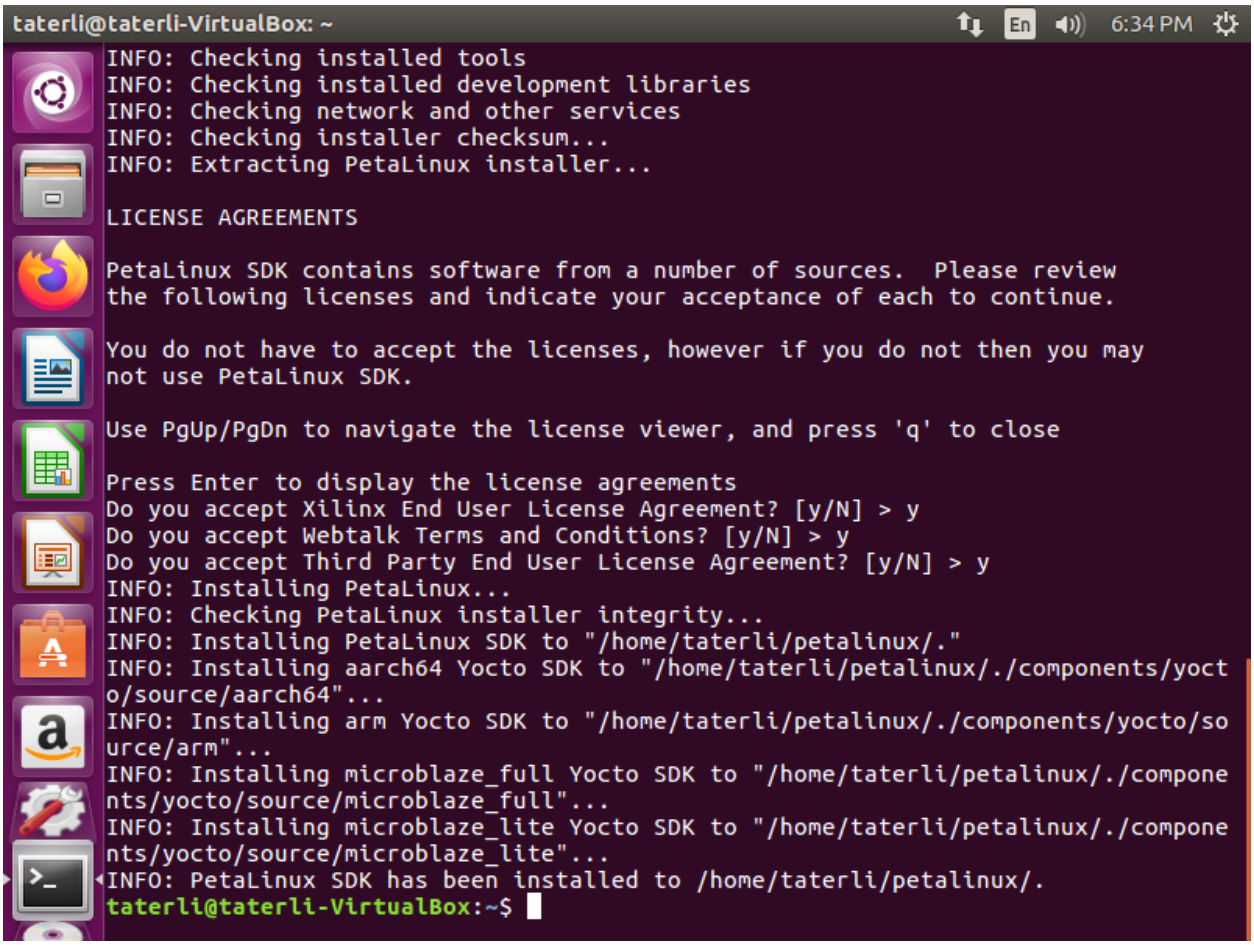
Installation Requirements

The PetaLinux Tools Installation requirements are:

- Minimum workstation requirements:
  - 8 GB RAM (recommended minimum for Xilinx tools)
  - 2 GHz CPU clock or equivalent (minimum of 8 cores)
  - 100 GB free HDD space
  - Supported OS:
    - Red Hat Enterprise Workstation/Server 7.2, 7.3, 7.4, 7.5 (64-bit)
    - CentOS 7.2, 7.3, 7.4, 7.5 (64-bit)
    - Ubuntu Linux 16.04.3, 16.04.4 (64-bit)
- You need to have root access to install the required packages mentioned in the following table. The PetaLinux tools need to be installed as a non-root user.
- PetaLinux requires a number of standard development tools and libraries to be installed on your Linux host workstation. Install the libraries and tools listed in the following table on the host Linux. All of the listed Linux Workstation Environments below have the 32-bit libraries needed by the PetaLinux tool. If there are any additional tool chain packages that need 32-bit libs on the host, install the same before issuing petalinux-build. Table 2-1 below describes the required packages, and how to install them on different Linux workstation environments.
- PetaLinux tools require your host system `/bin/sh` is bash. If you are using Ubuntu distribution and your `/bin/sh` is dash, consult your system administrator to change your default host system `/bin/sh` with the `sudo dpkg-reconfigure dash` command.

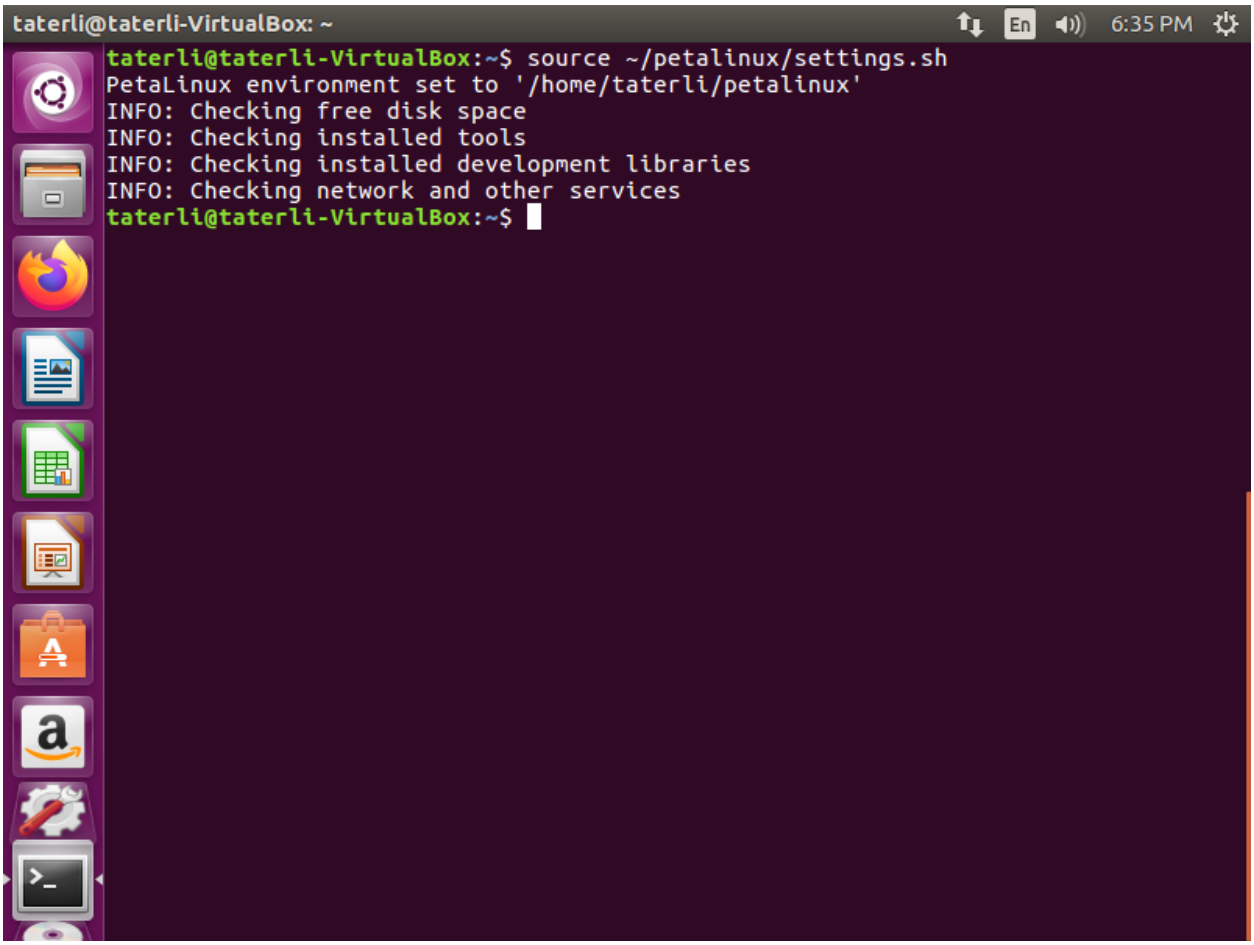
安装Ubuntu应该人人都没难度,然后直接运行上面的.run文件,推荐在虚拟机中配置,这样在外面一个Proxifer就可以把虚拟机全局代理起来,因为后续很多工具还需要科学上网手段,而且对版本也比较挑,然后缺什么补什么,把工具包装上为止.

```
./petalinux-v2018.3-final-installer.run ~/petalinux
```



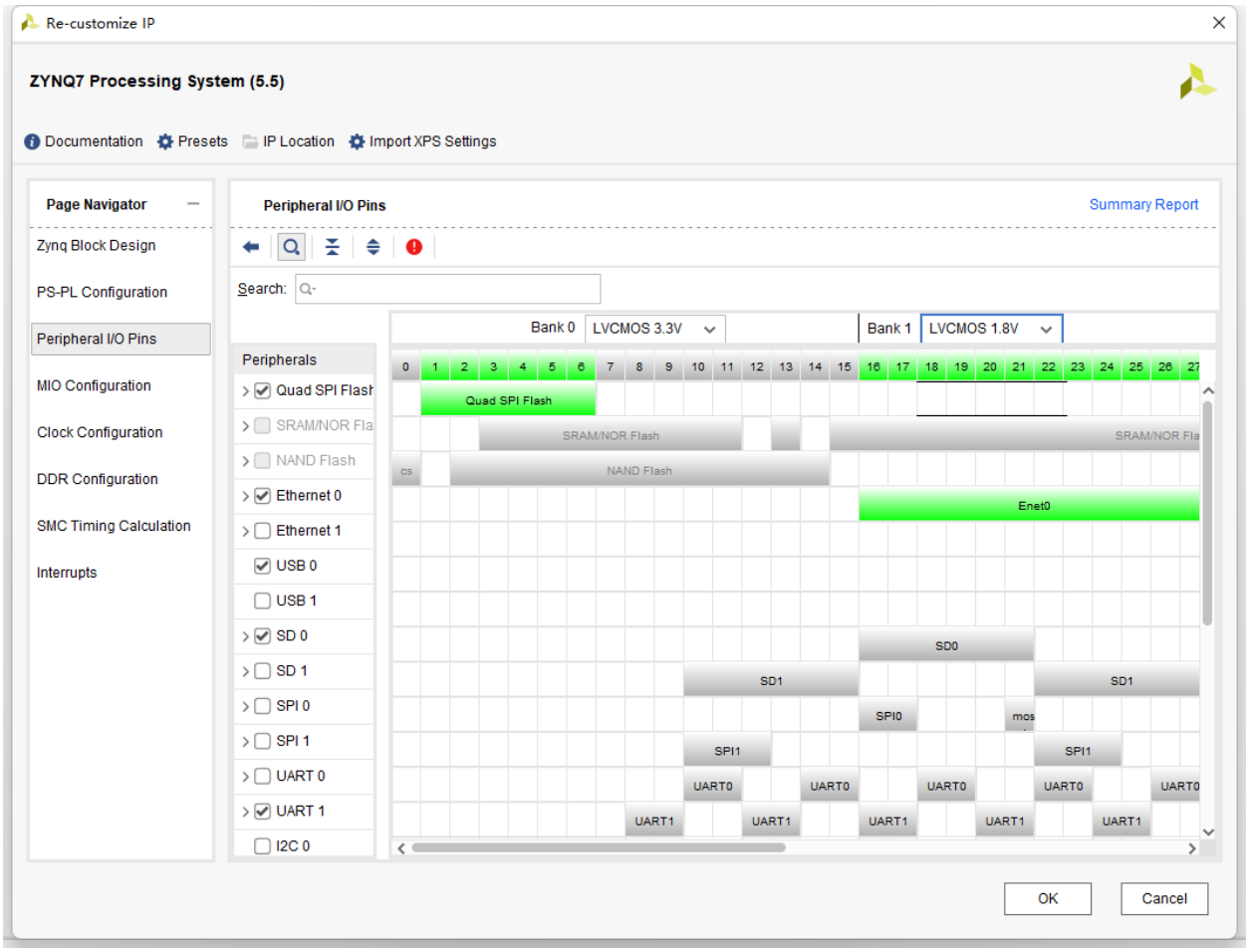
然后检查一下安装结果,有错误就解决错误,通常需要/bin/sh指向/bin/bash才能正常解释.

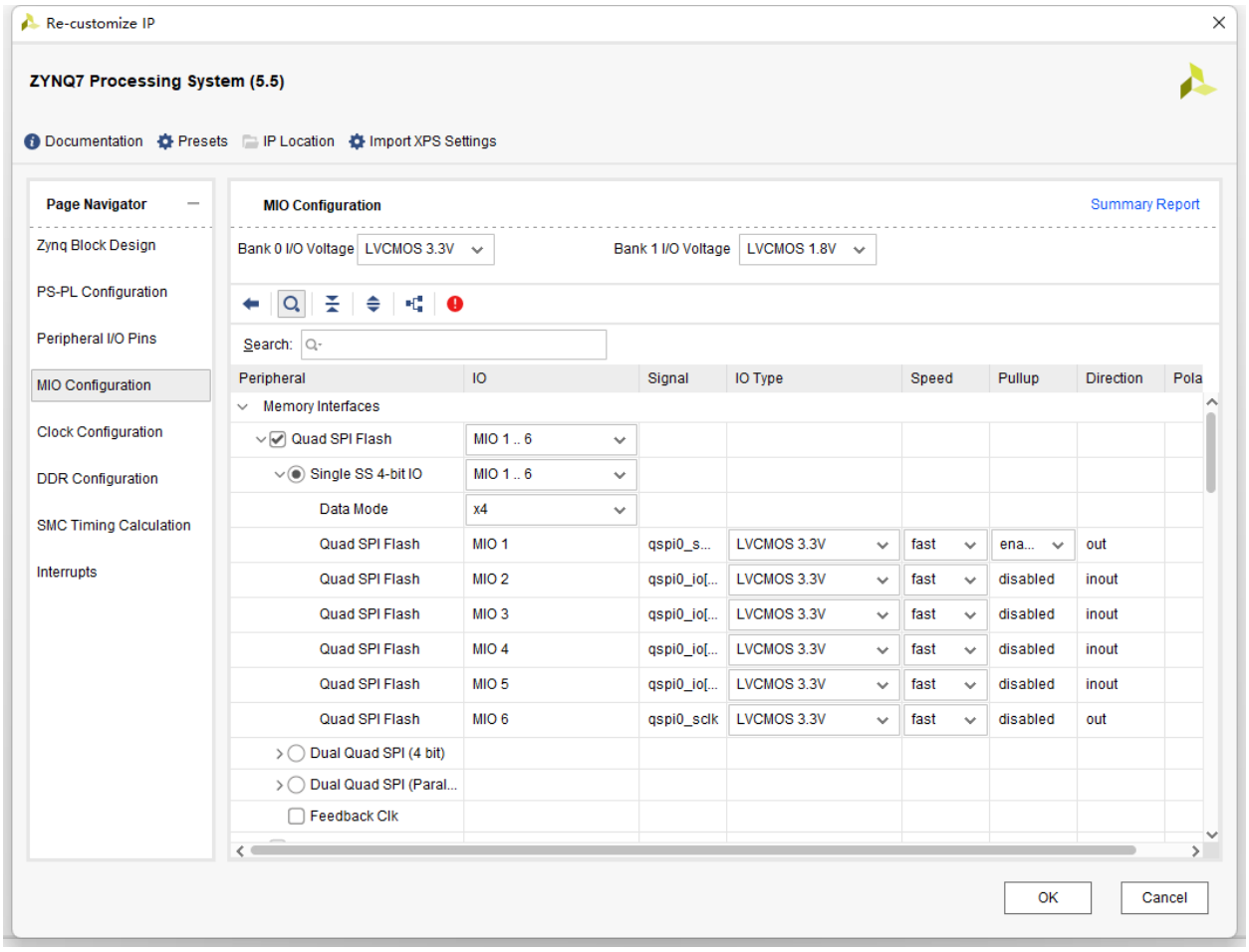
```
source ~/petalinux/settings.sh
```

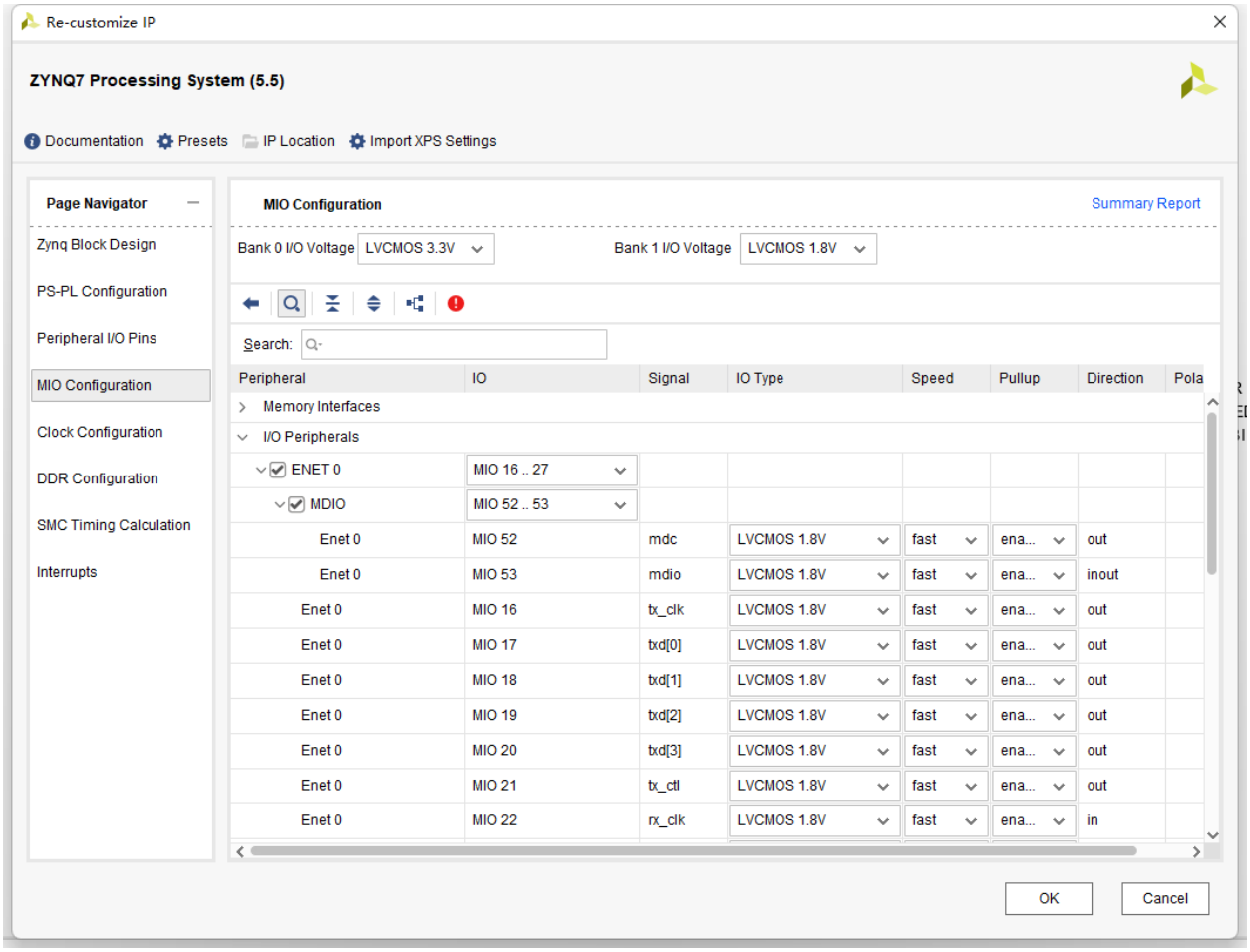


为了能编译PetaLinux需要一个hdf文件,hdf文件由Vivado导出,于是新建一个bd,配置IO.

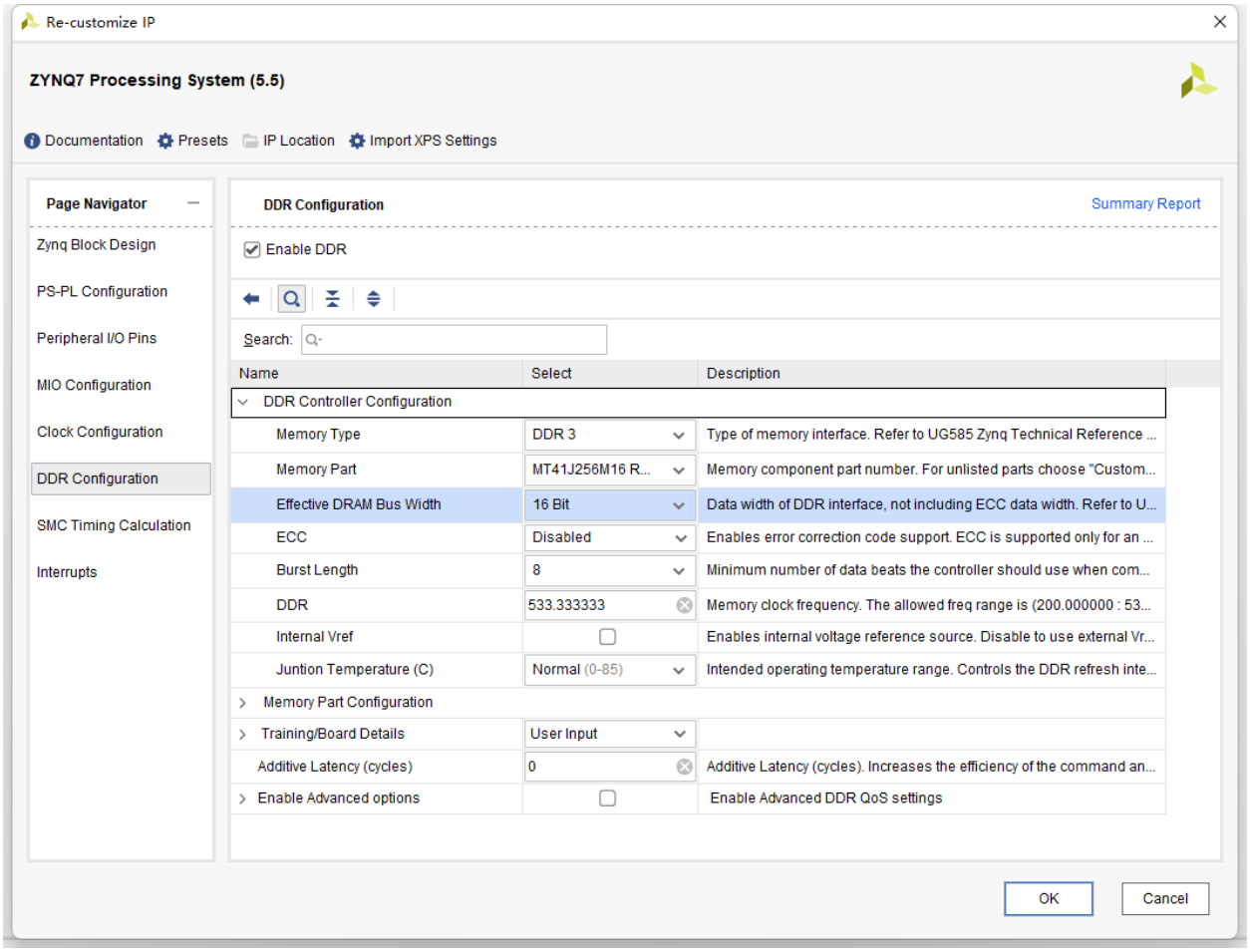
QSPI位于MIO1-6,以太网位MIO16-27,SD0位于MIO40-45,USB位于MIO28-39,串口MIO48-49,BANK 0是3.3V,BANK1是1.8V,QSPI,以太网,以太网MDIO,USB都设置成fast.


Re-customize IP
ZYNQ7 Processing System (5.5)
Documentation
Presets
IP Location
Import XPS Settings
Page Navigator
Zynq Block Design
PS-PL Configuration
Peripheral I/O Pins
MIO Configuration
Clock Configuration
DDR Configuration
SMC Timing Calculation
Interrupts
Peripheral I/O Pins
Summary Report
Search:
Bank 0
LVCMOS 3.3V
Bank 1
LVCMOS 1.8V
Peripherals
Quad SPI Flash
SRAM/NOR Fla
NAND Flash
Ethernet 0
Ethernet 1
USB 0
USB 1
SD 0
SD 1
SPI 0
SPI 1
UART 0
UART 1
I2C 0
Quad SPI Flash
SRAM/NOR Flash
SRAM/NOR Fla
cs
NAND Flash
Enet0
SD0
SD1
SD1
SPI0
mos
SPI1
SPI1
UART0
UART0
UART0
UART0
UART0
UART1
UART1
UART1
UART1
UART1
OK
Cancel

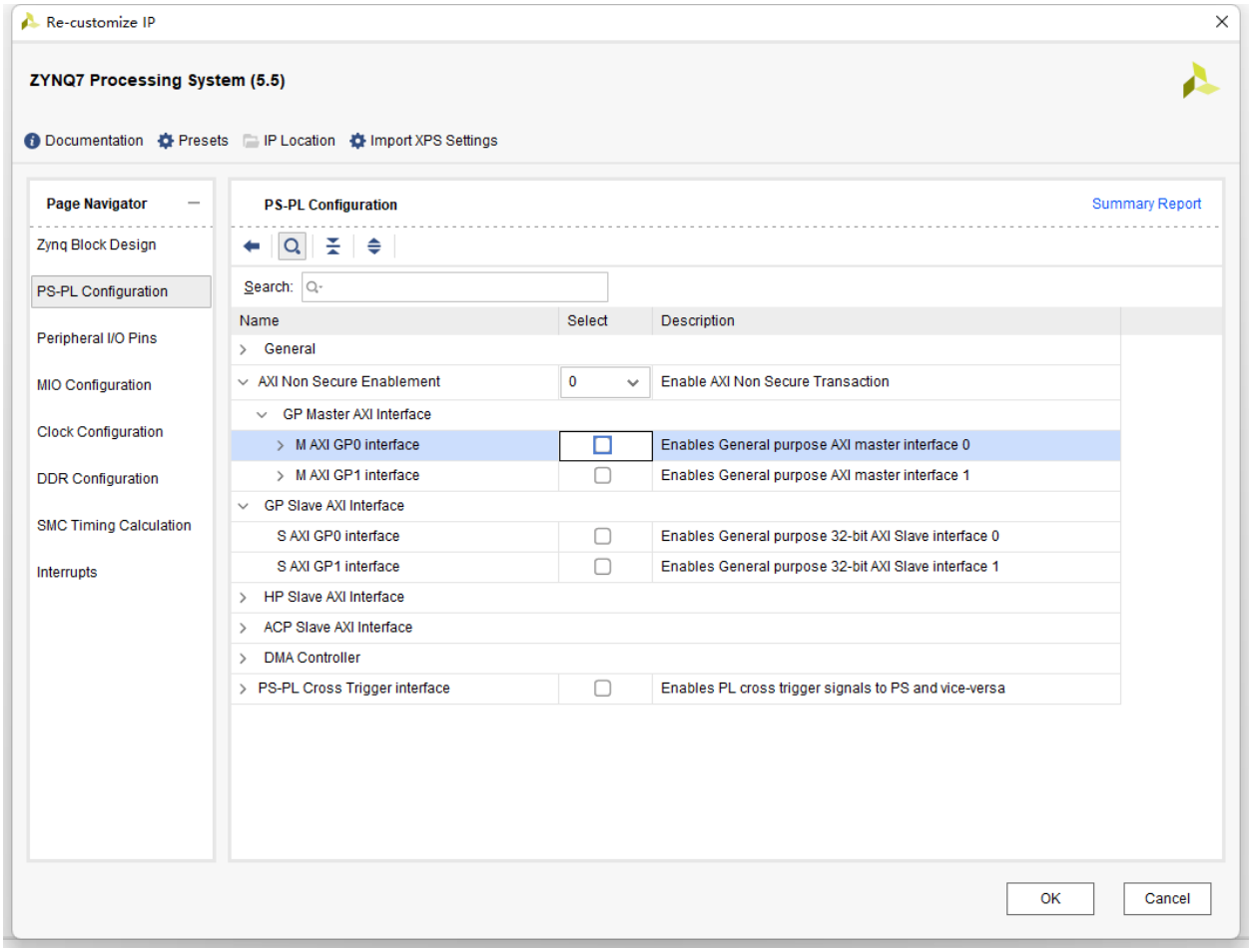

Re-customize IP
ZYNQ7 Processing System (5.5)
Documentation
Presets
IP Location
Import XPS Settings
Page Navigator
Zynq Block Design
PS-PL Configuration
Peripheral I/O Pins
MIO Configuration
Clock Configuration
DDR Configuration
SMC Timing Calculation
Interrupts
MIO Configuration
Summary Report
Bank 0 I/O Voltage LVCMOS 3.3V
Bank 1 I/O Voltage LVCMOS 1.8V
Search:
Peripheral
IO
Signal
IO Type
Speed
Pullup
Direction
Pola
Memory Interfaces
Quad SPI Flash
MIO 1 .. 6
Single SS 4-bit IO
MIO 1 .. 6
Data Mode
x4
Quad SPI Flash
MIO 1
qspi0_s...
LVCMOS 3.3V
fast
ena...
out
Quad SPI Flash
MIO 2
qspi0_io[...
LVCMOS 3.3V
fast
disabled
inout
Quad SPI Flash
MIO 3
qspi0_io[...
LVCMOS 3.3V
fast
disabled
inout
Quad SPI Flash
MIO 4
qspi0_io[...
LVCMOS 3.3V
fast
disabled
inout
Quad SPI Flash
MIO 5
qspi0_io[...
LVCMOS 3.3V
fast
disabled
inout
Quad SPI Flash
MIO 6
qspi0_sclk
LVCMOS 3.3V
fast
disabled
out
Dual Quad SPI (4 bit)
Dual Quad SPI (Paral...
Feedback Clk
OK
Cancel



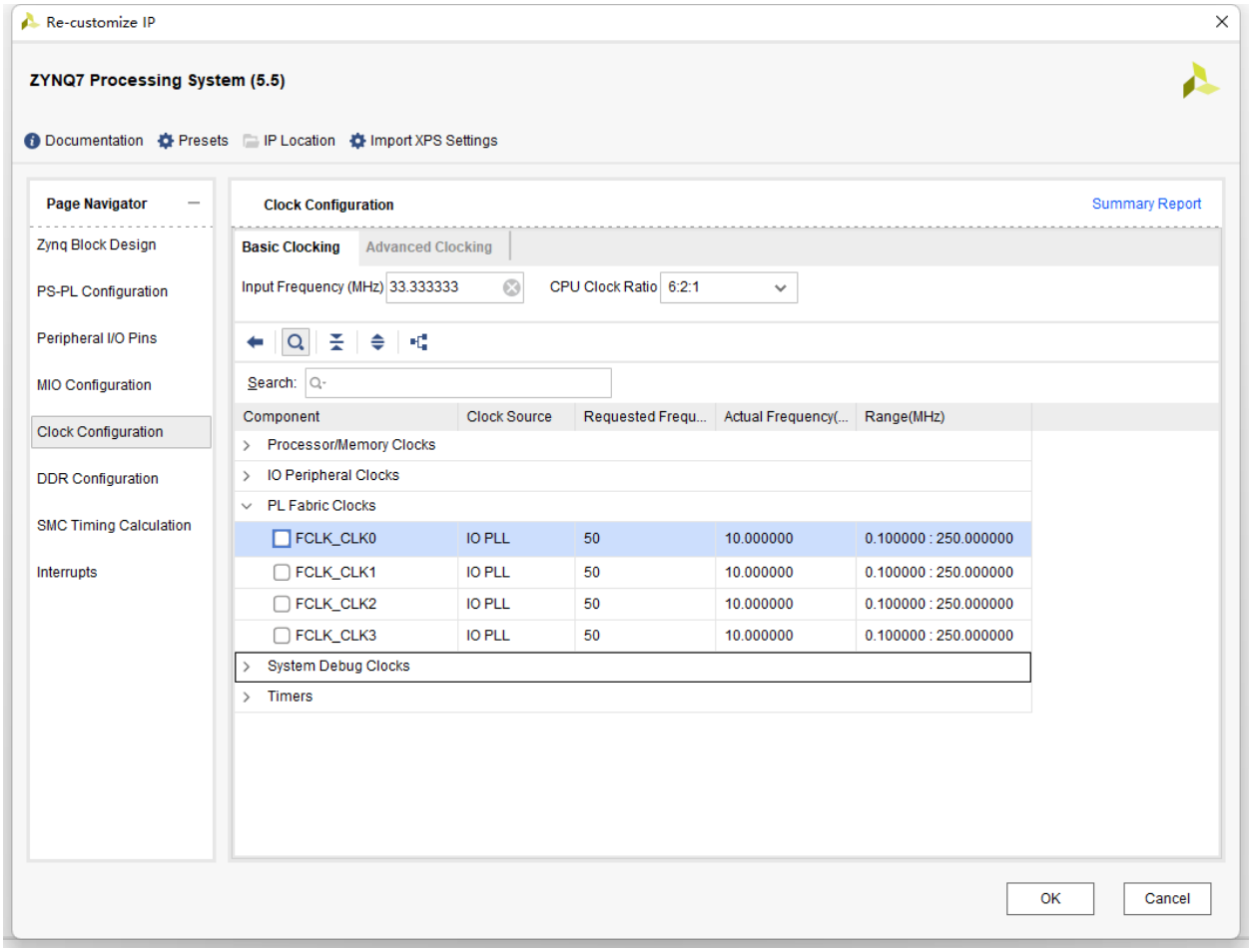
内存配置,已经是轻车熟路了.



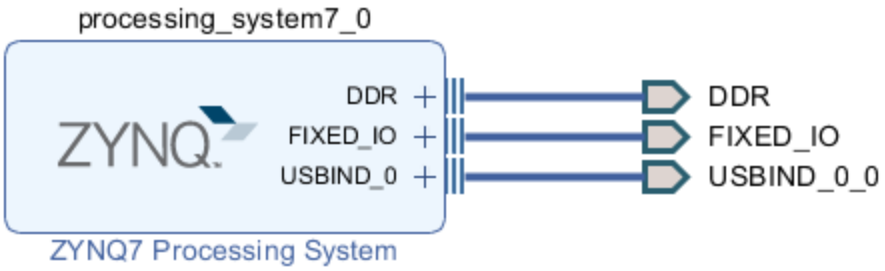
暂时不和PL通信,取消PL通信接口.



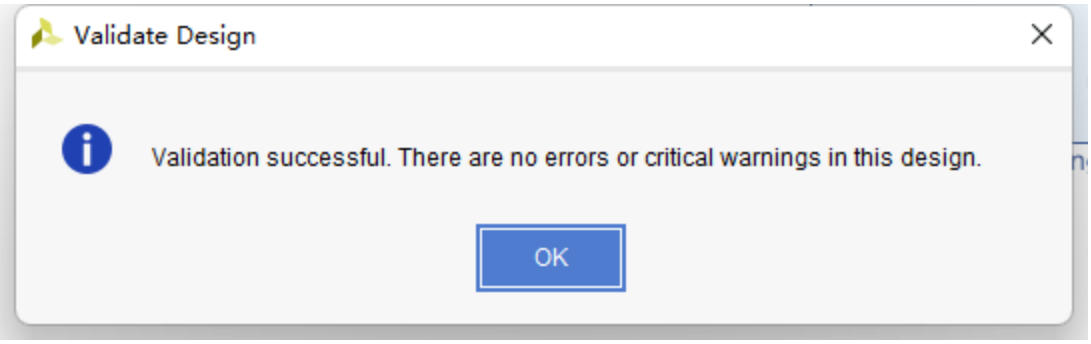
暂时也不对外输出时钟和复位.(复位没有截图,在另外页面.)



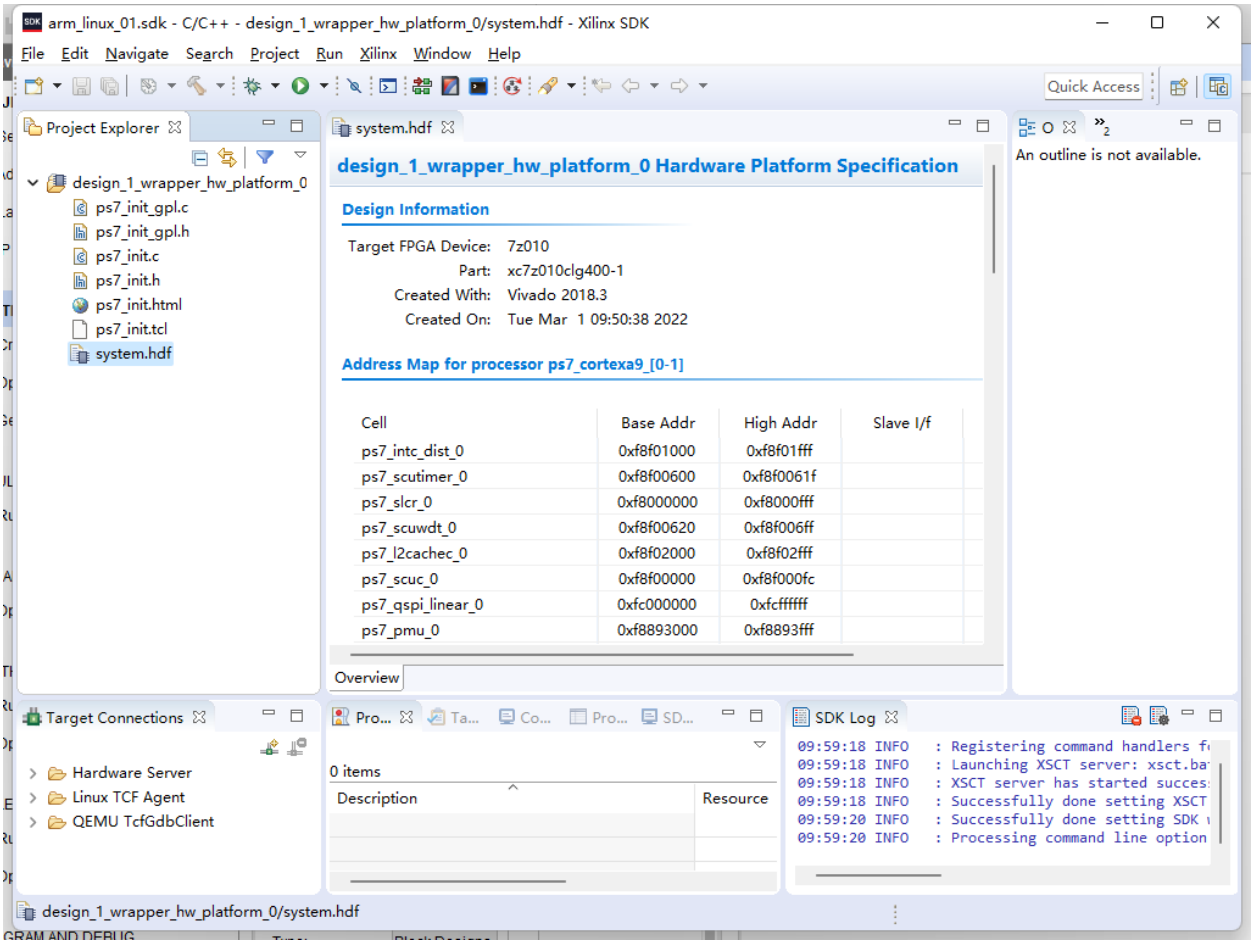
然后设法让他全部导出.



最后养成好习惯检查一下图纸.



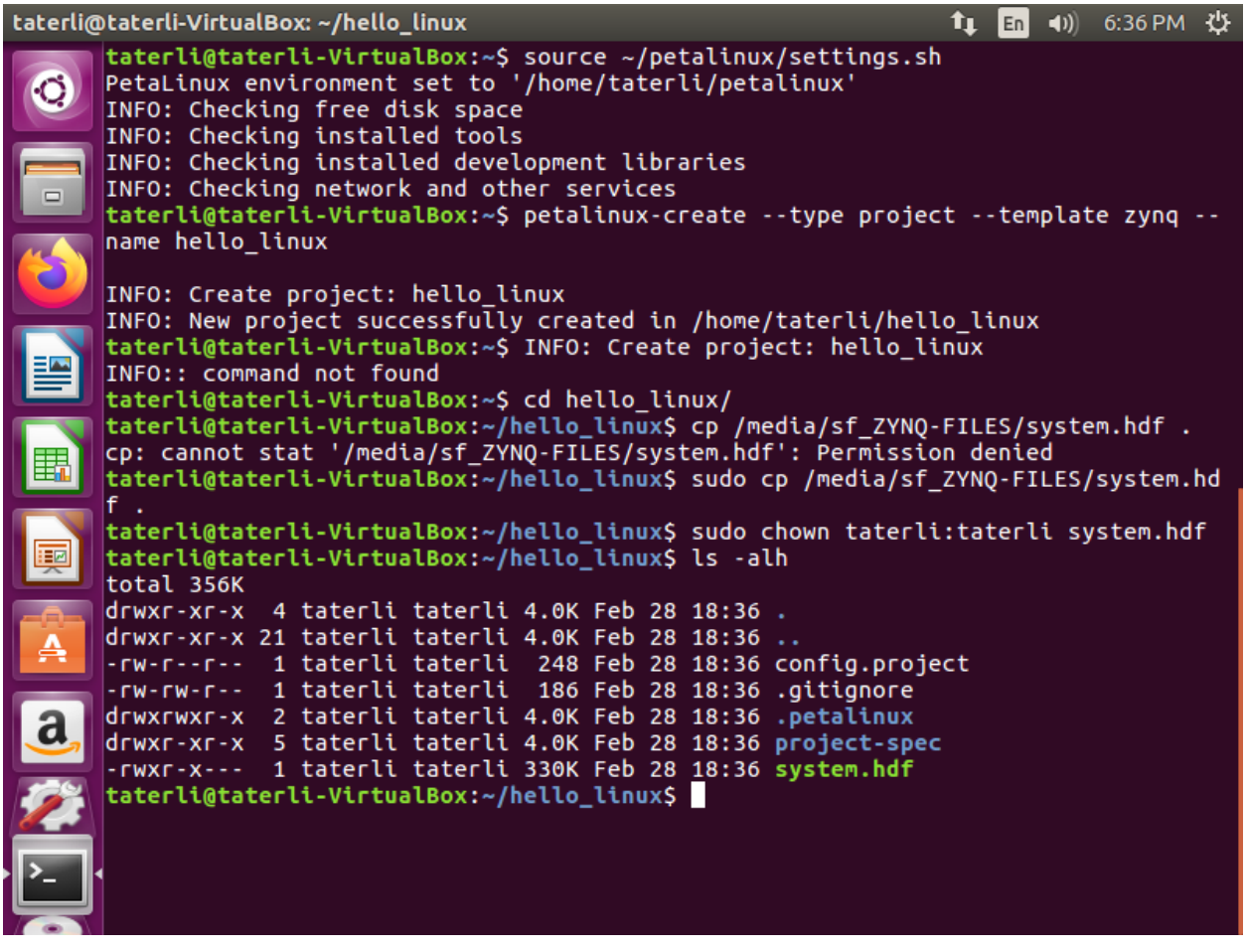
保存后一步一步生成直到导出HDF.



创建工程并复制HDF.(大致流程,并非一成不变!)

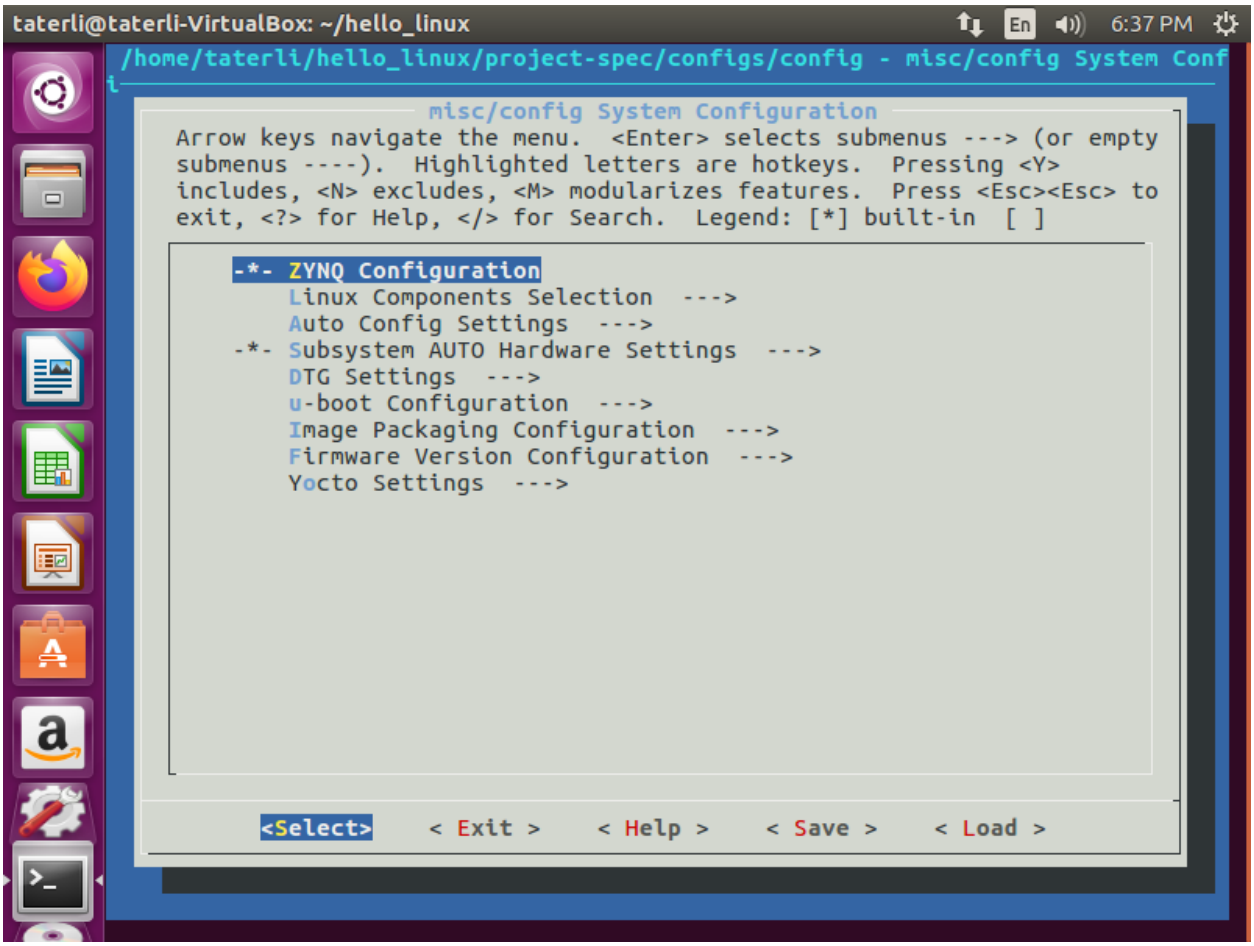
```
~$ petalinux-create --type project --template zynq --name hello_linux
INFO: Create project: hello_linux
INFO: New project successfully created in /home/tater/hello_linux
~$ cd hello_linux/
```

```
~/hello_linux$ cp [system.hdf 路径] .
~/hello_linux$ ls -alh
total 356K
drwxr-xr-x  4 tater tater 4.0K Feb 28 18:01 .
drwxr-xr-x 20 tater tater 4.0K Feb 28 18:01 ..
-rw-r--r--  1 tater tater  248 Feb 28 18:01 config.project
-rw-rw-r--  1 tater tater  170 Feb 28 18:01 .gitignore
drwxrwxr-x  2 tater tater 4.0K Feb 28 18:01 .petalinux
drwxr-xr-x  5 tater tater 4.0K Feb 28 18:01 project-spec
-rw-rw-r--  1 tater tater 330K Feb 28 18:01 system.hdf
~/hello_linux$
```

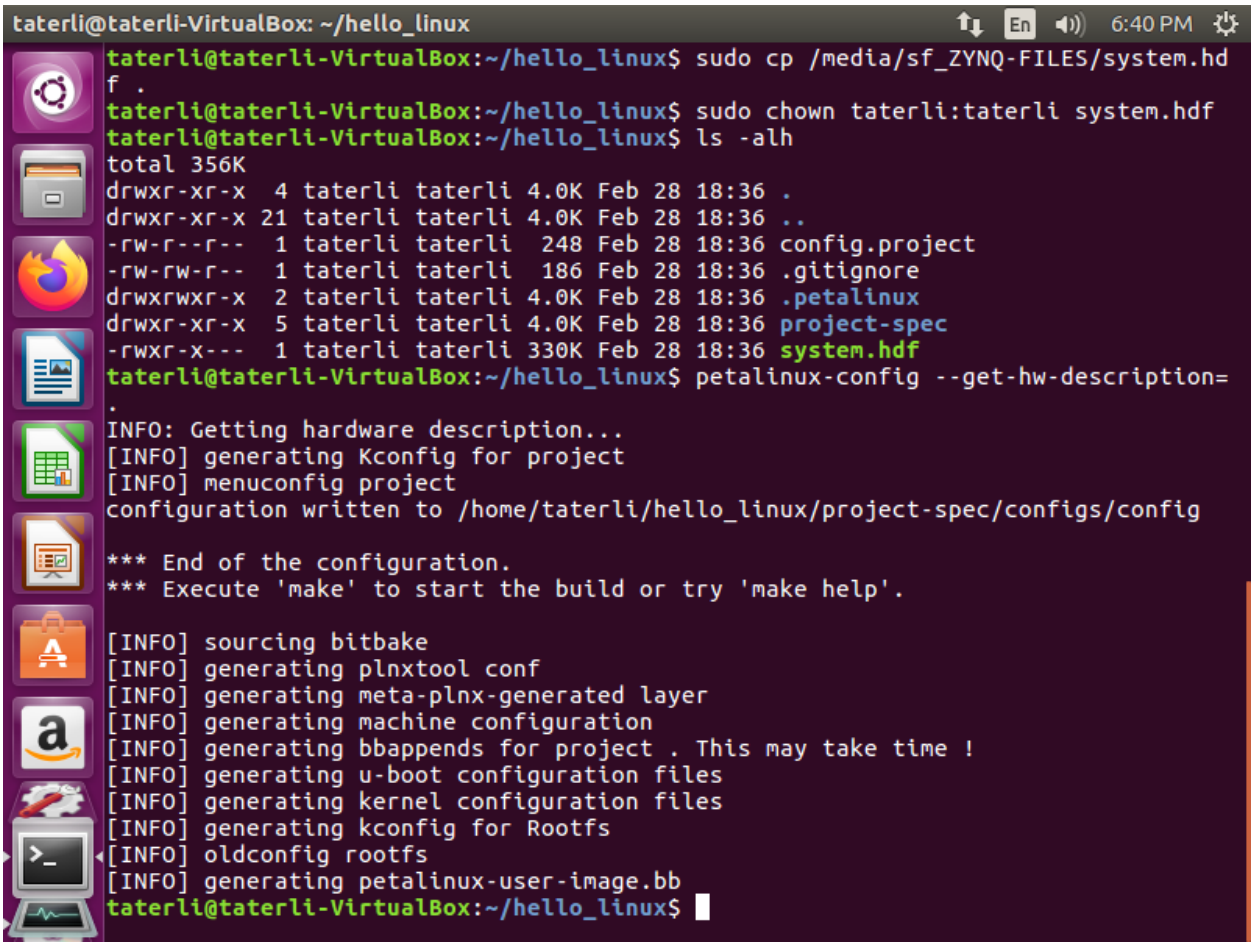


然后开始配置.

```
petalinux-config --get-hw-description=.
```



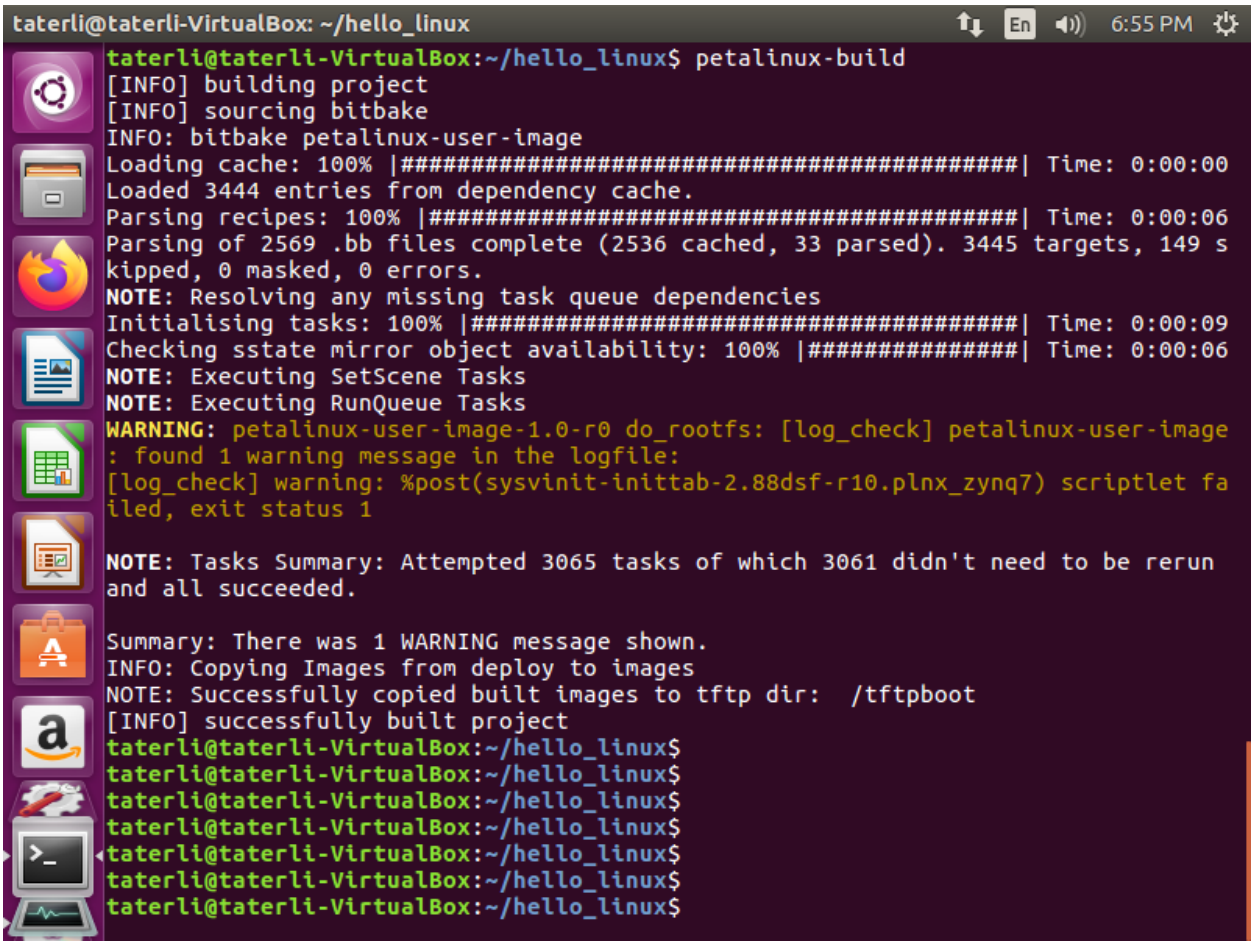
我们做最小启动可以完全不配置这里,不过我们依然可以大致浏览下有什么东西,退出配置界面就会自动开始配置,期间CPU占用比较高.



然后开始编译.

```
petalinux-build
```

耐心等待编译完成,有错误就解决错误.



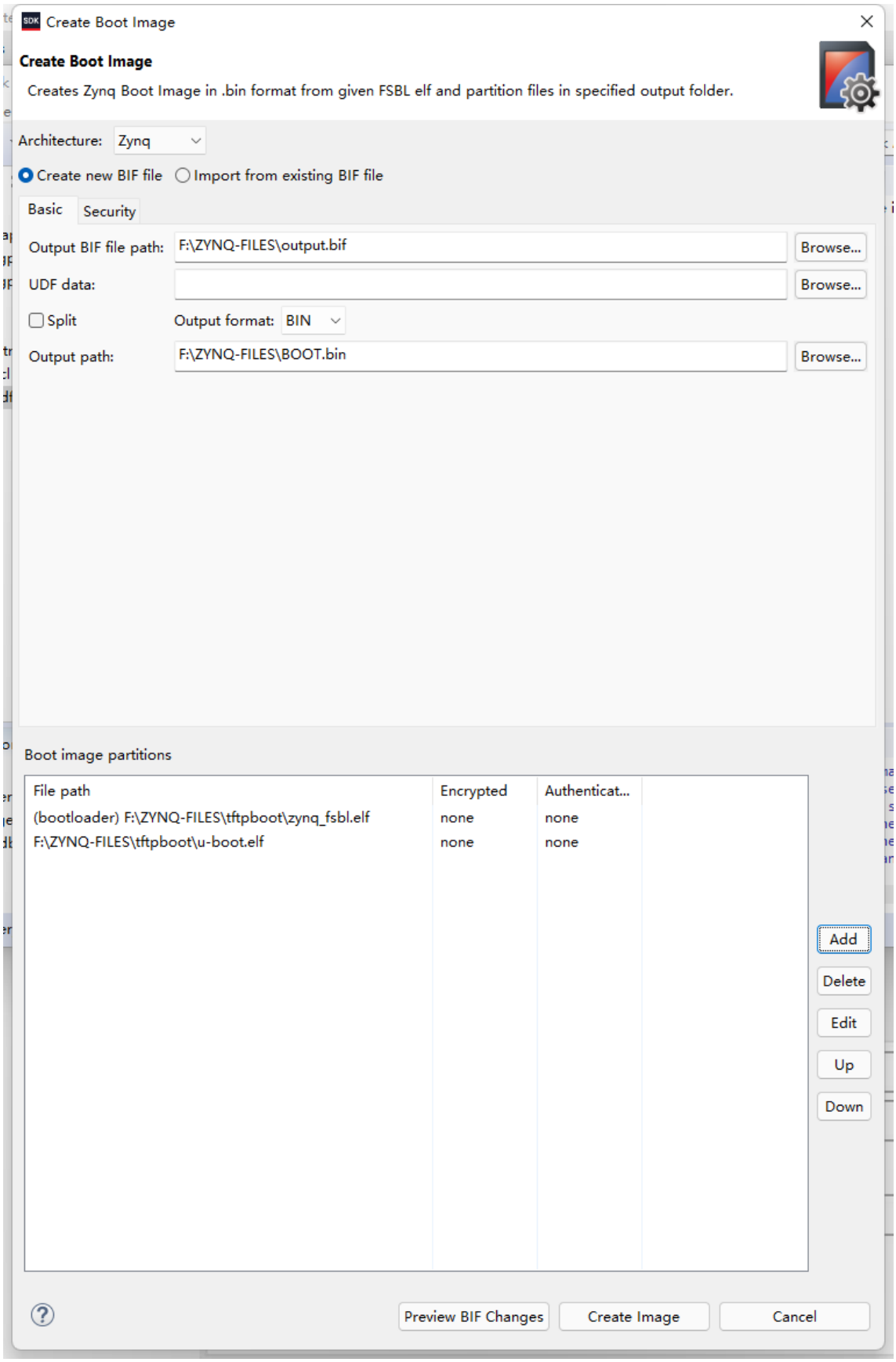
需要拷贝的编译结果有以下几个.

1. zynq_fsbl.elf - 完成对PL的配置
2. u-boot.elf - 负责引导ARM完成Linux启动
3. image.ub - 真实Linux镜像

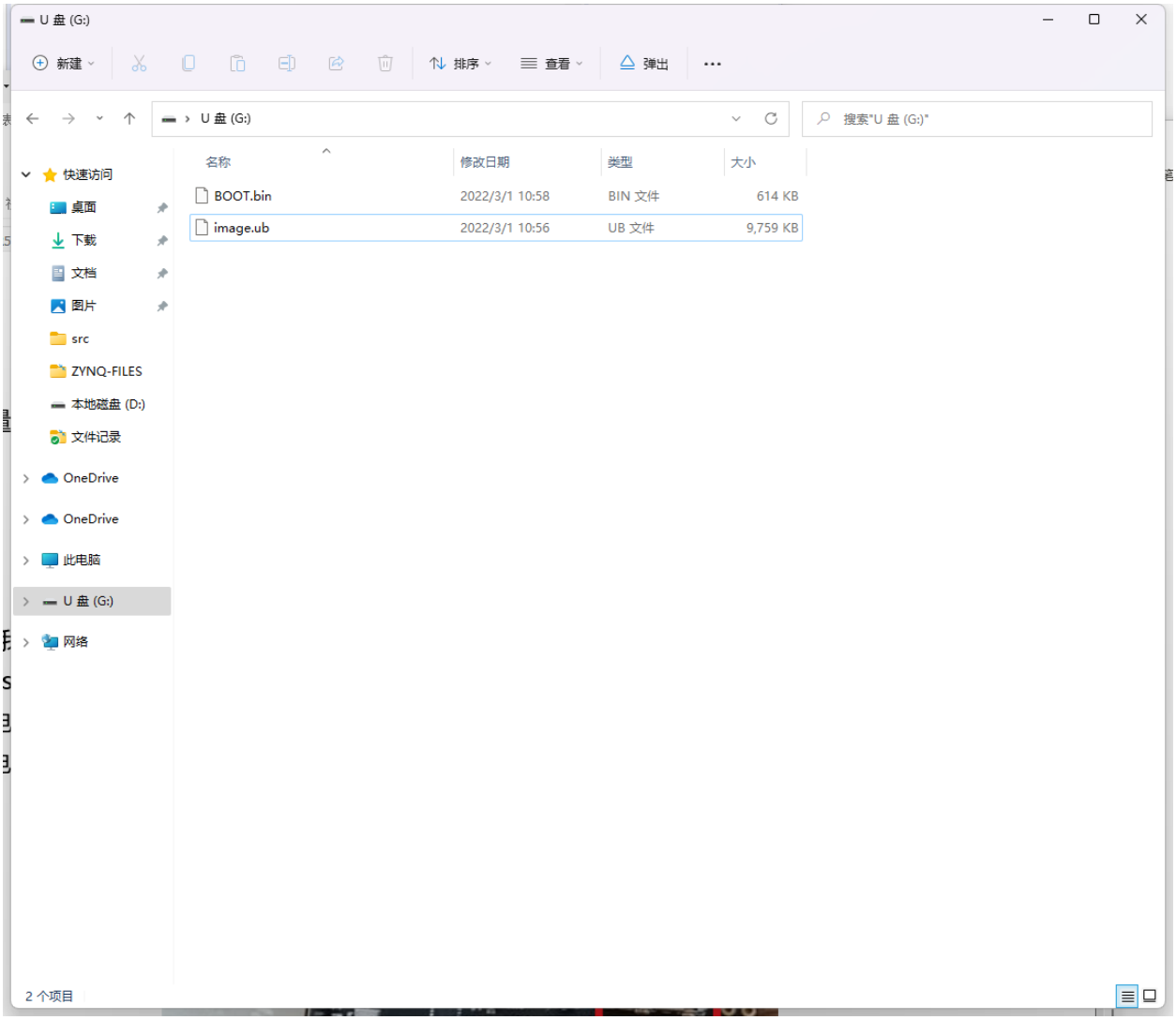
```
sudo cp /tftpboot /media/sf_ZYNQ-FILES -r
```

名称	修改日期	类型	大小
image.ub	2022/3/1 10:56	UB 文件	9,759 KB
rootfs.bin	2022/3/1 10:56	BIN 文件	9,759 KB
rootfs.cpio	2022/3/1 10:56	CPIO 文件	13,225 KB
rootfs.cpio.bz2	2022/3/1 10:56	BZ2 压缩文件	5,444 KB
rootfs.cpio.gz	2022/3/1 10:56	GZ 压缩文件	5,895 KB
rootfs.cpio.gz.u-boot	2022/3/1 10:56	U-BOOT 文件	5,895 KB
rootfs.ext3	2022/3/1 10:56	EXT3 文件	65,536 KB
rootfs.ext3.bz2	2022/3/1 10:56	BZ2 压缩文件	5,490 KB
rootfs.ext4	2022/3/1 10:56	EXT4 文件	65,536 KB
rootfs.ext4.gz	2022/3/1 10:56	GZ 压缩文件	5,956 KB
rootfs.its	2022/3/1 10:56	ITS 文件	3 KB
rootfs.jffs2	2022/3/1 10:56	JFFS2 文件	7,424 KB
rootfs.manifest	2022/3/1 10:56	MANIFEST 文件	6 KB
rootfs.tar.bz2	2022/3/1 10:56	BZ2 压缩文件	5,438 KB
rootfs.tar.gz	2022/3/1 10:56	GZ 压缩文件	5,912 KB
rootfs.testdata.json	2022/3/1 10:56	JSON 源文件	283 KB
system.dtb	2022/3/1 10:56	DTB 文件	14 KB
System.map.linux	2022/3/1 10:56	LINUX 文件	1,922 KB
u-boot.bin	2022/3/1 10:56	BIN 文件	512 KB
u-boot.elf	2022/3/1 10:56	ELF 文件	4,336 KB
vmlinux	2022/3/1 10:56	文件	24,622 KB
zImage	2022/3/1 10:56	文件	9,761 KB
zynq_fsbl.elf	2022/3/1 10:56	ELF 文件	434 KB

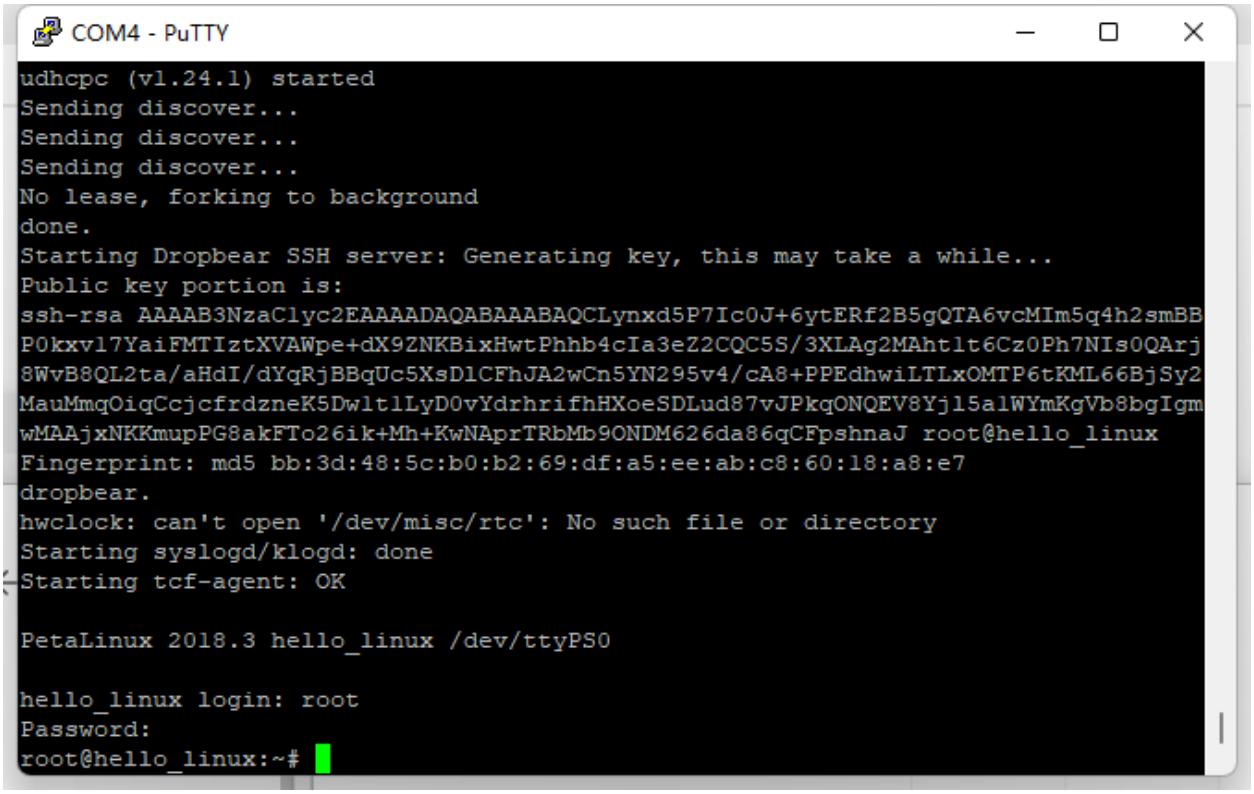
回到Windows的SDK,选Xilinx→Create Boot Image菜单,制作BOOT.BIN.


Create Boot Image
Create Boot Image
Creates Zynq Boot Image in .bin format from given FSBL elf and partition files in specified output folder.
Architecture: Zynq
Create new BIF file
Import from existing BIF file
Basic
Security
Output BIF file path: F:\ZYNQ-FILES\output.bif
Browse...
UDF data:
Browse...
Split
Output format: BIN
Output path: F:\ZYNQ-FILES\BOOT.bin
Browse...
Boot image partitions
File path
Encrypted
Authenticat...
(bootloader) F:\ZYNQ-FILES\tftpboot\zynq_fsbl.elf
none
none
F:\ZYNQ-FILES\tftpboot\u-boot.elf
none
none
Add
Delete
Edit
Up
Down
Preview BIF Changes
Create Image
Cancel

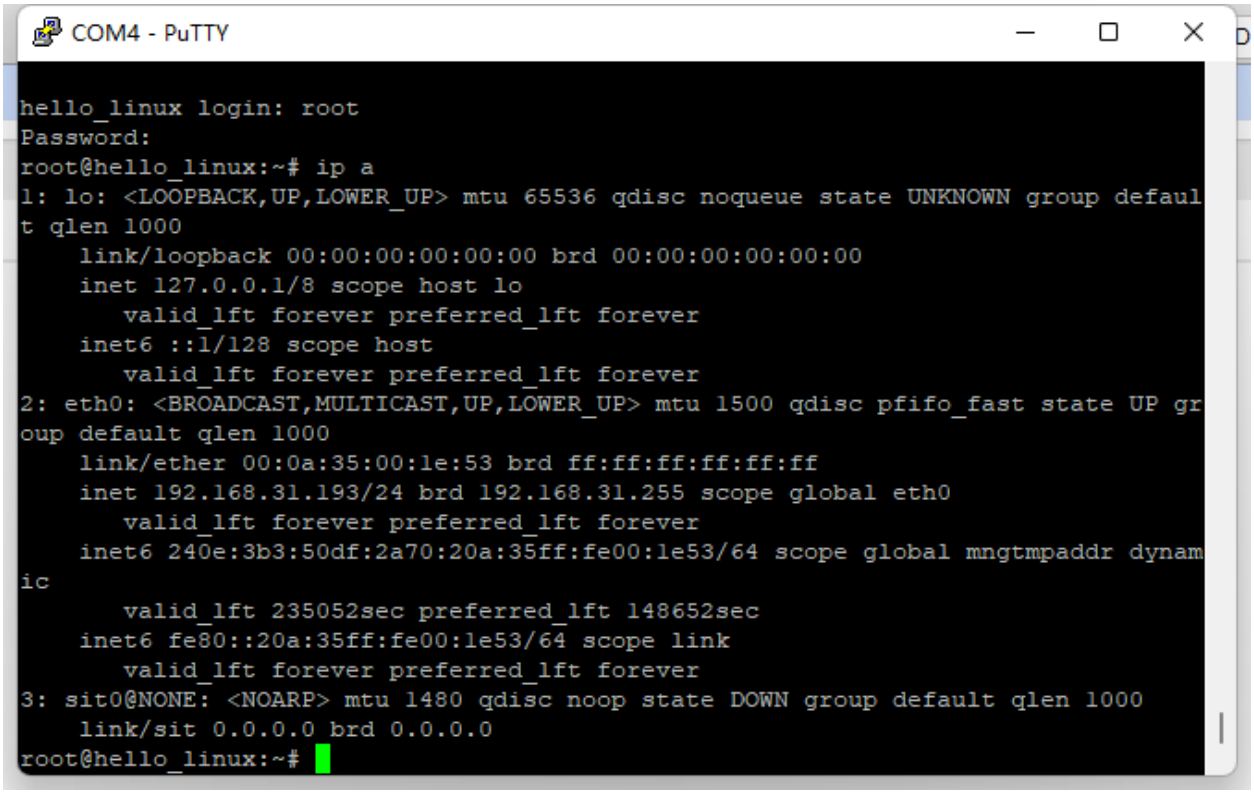
创建后就有BOOT.BIN,然后把BOOT.bin和image.ub复制到FAT32的TF卡中,并塞到TF启动卡槽,调整BOOT,比如ZYNQ ECO板子就是把BOOT两个引脚都拨到SPI Flash那一侧.

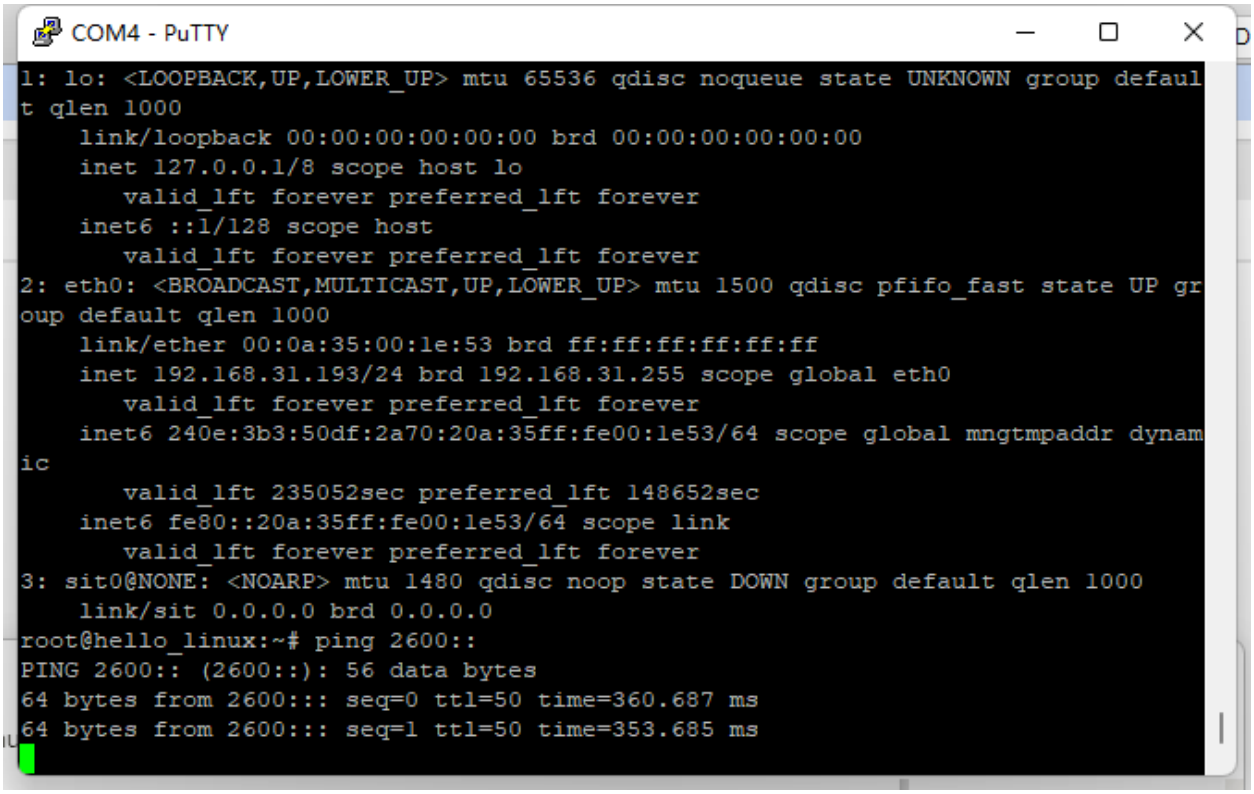


顺利启动,用户名密码都是root.


COM4 - PuTTY
udhcpc (v1.24.1) started
Sending discover...
Sending discover...
Sending discover...
No lease, forking to background
done.
Starting Dropbear SSH server: Generating key, this may take a while...
Public key portion is:
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQCLynxd5P7Ic0J+6ytERf2B5gQTA6vcMIm5q4h2smBB
P0kxvl7YaiFMTIztXVAWpe+dX9ZNKBixHwtPhhb4cIa3eZ2CQC5S/3XLAg2MAht1t6Cz0Ph7NIs0QArj
8WvB8QL2ta/aHdI/dYqRjBBqUc5XsDlCFhJA2wCn5YN295v4/cA8+PPEdhwiLTLxOMTP6tKML66BjSy2
MauMmqOiqCcjcfrdzneK5Dwl1LyD0vYdrhrifhHXoeSDLud87vJPkqONQEV8Yj15alWYmKgVb8bgIgm
wMAAjxNKKmupPG8akFTo26ik+Mh+KwNAprTRbMb9ONDM626da86qCFpshnaJ root@hello_linux
Fingerprint: md5 bb:3d:48:5c:b0:b2:69:df:a5:ee:ab:c8:60:18:a8:e7
dropbear.
hwclock: can't open '/dev/misc/rtc': No such file or directory
Starting syslogd/klogd: done
Starting tcf-agent: OK
PetaLinux 2018.3 hello_linux /dev/ttyPS0
hello_linux login: root
Password:
root@hello_linux:~#

网络OK.


COM4 - PuTTY
hello_linux login: root
Password:
root@hello_linux:~# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group defaul
t qlen 1000
link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
inet 127.0.0.1/8 scope host lo
valid_lft forever preferred_lft forever
inet6 ::1/128 scope host
valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP gr
oup default qlen 1000
link/ether 00:0a:35:00:1e:53 brd ff:ff:ff:ff:ff:ff
inet 192.168.31.193/24 brd 192.168.31.255 scope global eth0
valid_lft forever preferred_lft forever
inet6 240e:3b3:50df:2a70:20a:35ff:fe00:1e53/64 scope global mngtmpaddr dynam
ic
valid_lft 235052sec preferred_lft 148652sec
inet6 fe80::20a:35ff:fe00:1e53/64 scope link
valid_lft forever preferred_lft forever
3: sit0@NONE: <NOARP> mtu 1480 qdisc noop state DOWN group default qlen 1000
link/sit 0.0.0.0 brd 0.0.0.0
root@hello_linux:~#


COM4 - PuTTY
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group defaul
t qlen 1000
link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
inet 127.0.0.1/8 scope host lo
valid_lft forever preferred_lft forever
inet6 ::1/128 scope host
valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP gr
oup default qlen 1000
link/ether 00:0a:35:00:1e:53 brd ff:ff:ff:ff:ff:ff
inet 192.168.31.193/24 brd 192.168.31.255 scope global eth0
valid_lft forever preferred_lft forever
inet6 240e:3b3:50df:2a70:20a:35ff:fe00:1e53/64 scope global mngtmpaddr dynam
ic
valid_lft 235052sec preferred_lft 148652sec
inet6 fe80::20a:35ff:fe00:1e53/64 scope link
valid_lft forever preferred_lft forever
3: sit0@NONE: <NOARP> mtu 1480 qdisc noop state DOWN group default qlen 1000
link/sit 0.0.0.0 brd 0.0.0.0
root@hello_linux:~# ping 2600::
PING 2600:: (2600::): 56 data bytes
64 bytes from 2600::: seq=0 ttl=50 time=360.687 ms
64 bytes from 2600::: seq=1 ttl=50 time=353.685 ms