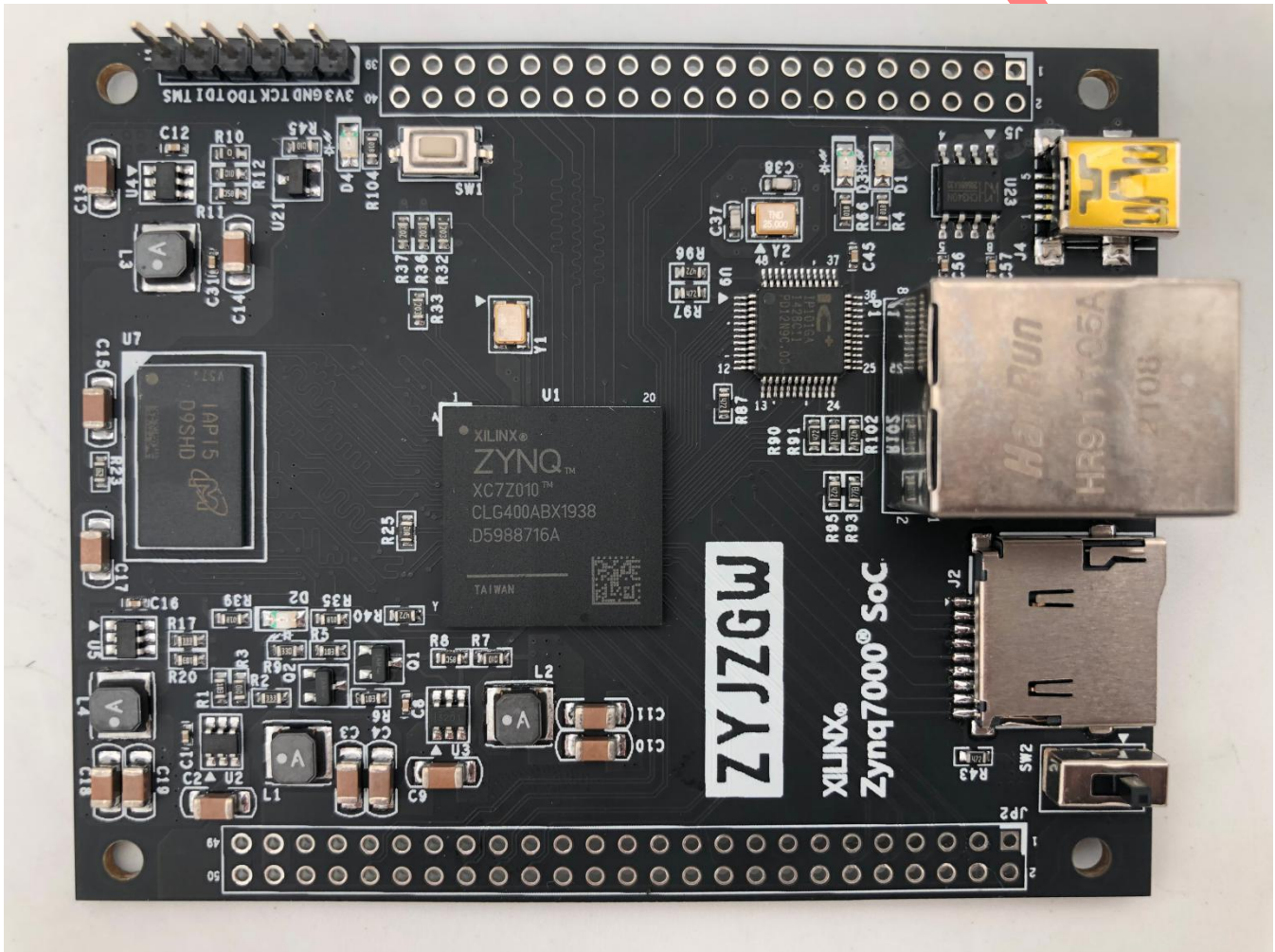


ZYJZGW ZYNQ XC7Z010 STARTER KIT

USER MANUAL



Preface

The ZYJZGW® ZYNQ7000 Starter Kit uses Xilinx Zynq®-7000 device which integrates the software programmability of an ARM®-based processor with the hardware programmability of an FPGA, enabling key analytics and hardware acceleration while integrating CPU, DSP, ASSP, and mixed signal functionality on a single device. Consisting of single-core Zynq-7000S and dual-core Zynq-7000 devices, the Zynq-7000 family is the best price to performance-per-watt, fully scalable SoC platform for your unique application requirements.

Table of Contents

1. INTRODUCTION.....	3
1.1 DOCUMENT SCOPE.....	3
2. GETTING STARTED.....	3
2.1 STEPS TO CUSTOMIZE THE PETALINUX.....	4
2.2 TEST THE ETHERNET UNDER LINUX ENVIRONMENT.....	16
3. REFERENCE.....	17
4. REVISION.....	18

1. Introduction

1.1 Document Scope

This user manual introduces the procedure to make the PetaLinux environment running on the ZYJZGW ZYNQ7000 Starter Kit. The PetaLinux environment mainly covers three parts: u-boot, Linux OS and file system. All of those parts are developed with PetaLinux 2019.2 under Ubuntu 18.04.1 (64bit) environment. The prerequisites before working with the PetaLinux are shown as below:

1. Preferred and verified Ubuntu version is Ubuntu 18.04.1 (64bit).

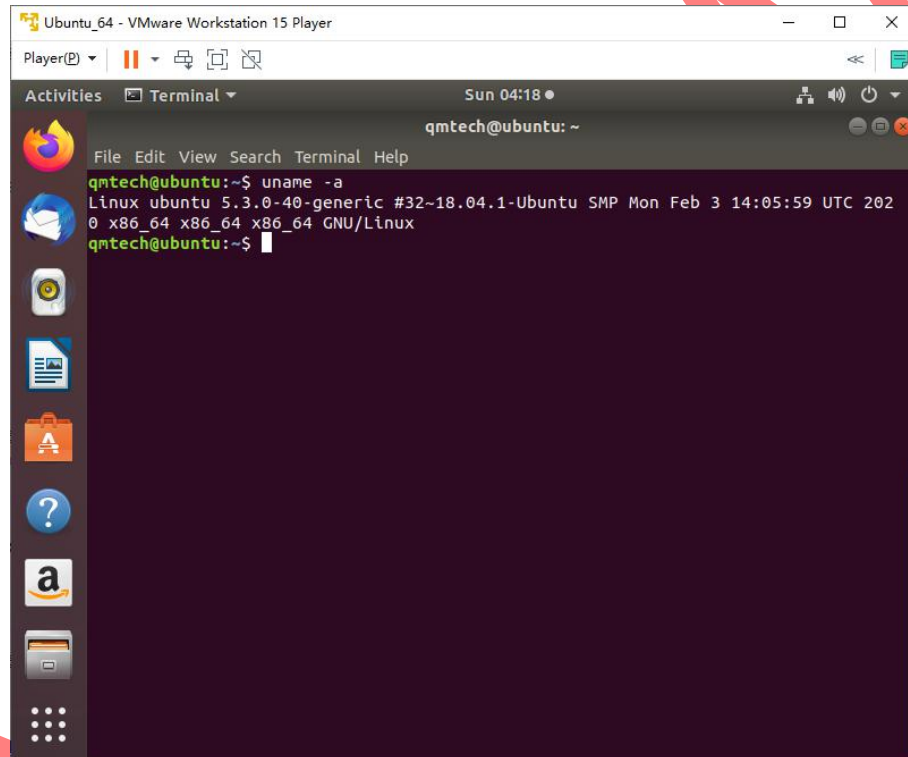


Figure 1-1. Ubuntu Version

2. Users shall install the required packages before install the PetaLinux 2019.2 in Ubuntu. The detailed required packages are mentioned in **UG1144: Table 2: Packages and Linux Workstation Environments**.
Notification: Users need to have root access to install the required packages mentioned in that table. The PetaLinux tools need to be installed as a non-root user.
3. Users shall have the basic knowledge about the usage of the Linux environment. Know how to use the cross-compile toolchain including arm-gcc-linux-, makefile, etc.

2. Getting Started

This chapter describes the detailed steps to create a customized PetaLinux. Comparing to the existing ZYNQ development board e.g. Xilinx ZC702, there are many differences in the ZYJZGW ZYNQ7000 Starter Kit. For example, there's only one 16bit width DDR3 memory chip connected to PS ARM core. And MII Ethernet interface is implemented at the PL side. Hence, the u-boot/ Linux device tree for the Bajie Board needs to be updated here.

2.1 Steps to Customize the PetaLinux

The first step is to customize the hardware info required by PetaLinux. The hardware info could be retrieved from below Vivado project:

\$ZYJZGW_XC7Z010_STARTER_KIT_V01\PetaLinux\Project04_Uboot_20191101.zip

Below image shows the customized ZYNQ system in Vivado 2018.3. Make sure the project could successfully pass the three steps: Synthesis, implementation and Generate Bitstream.

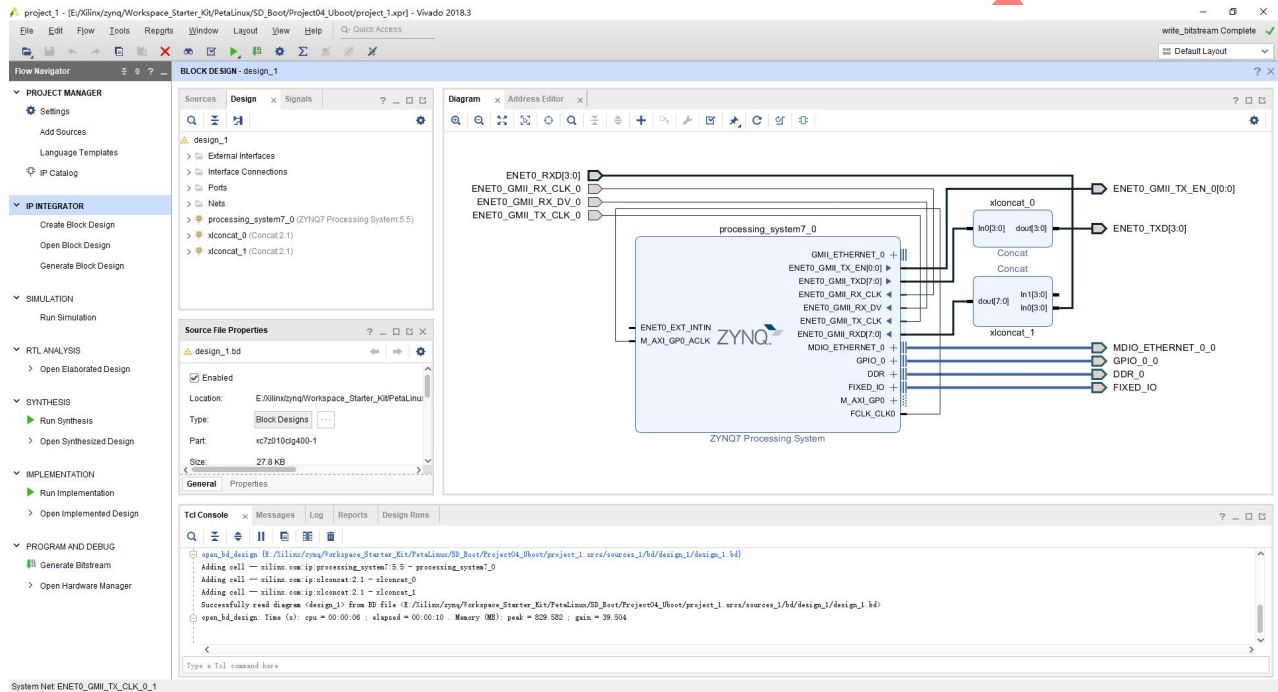


Figure 2-1. Modifications

Then users could retrieve the customized hardware info by clicking the [File] -> [Export] -> [Export Hardware]. And remember check the [Include Bitstream] before click the [OK] button.

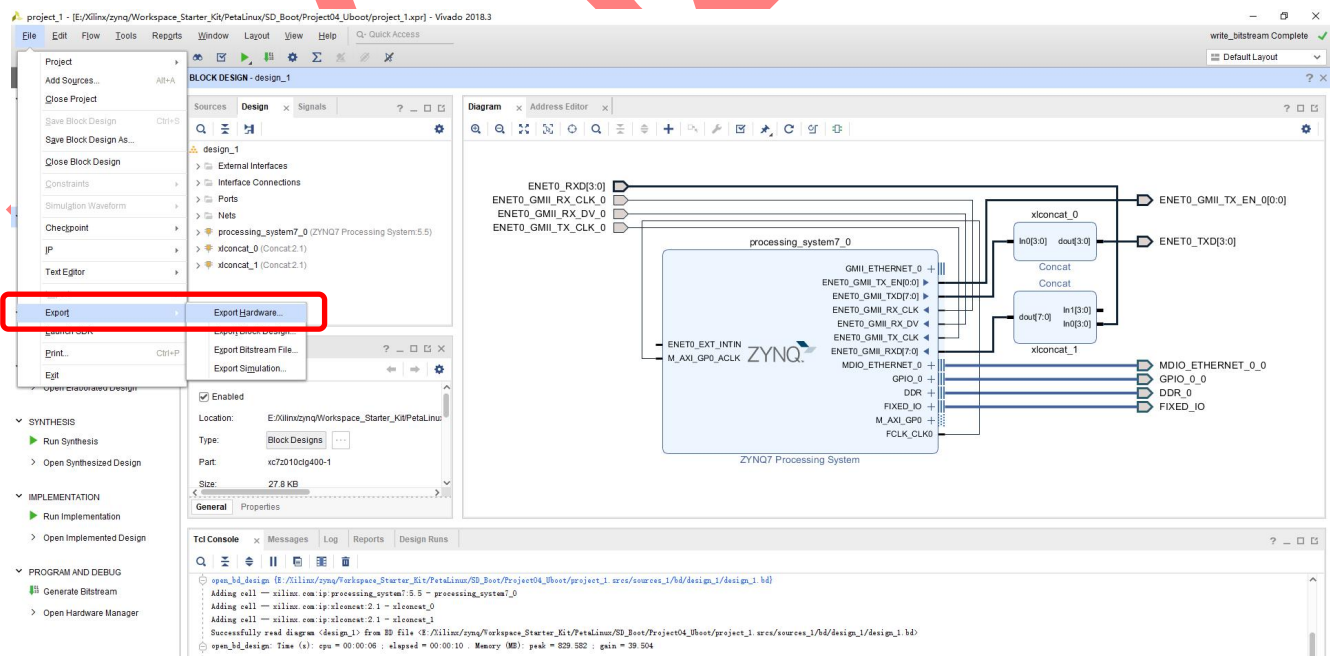


Figure 2-2. Export Hardware Info

The generated hardware info file could be found here:

\$ZYJZGW_XC7Z010_STARTER_KIT_V01\PetaLinux\Project04_Uboot_20191101\project_1.sdk\design_1_wrapper.hdf

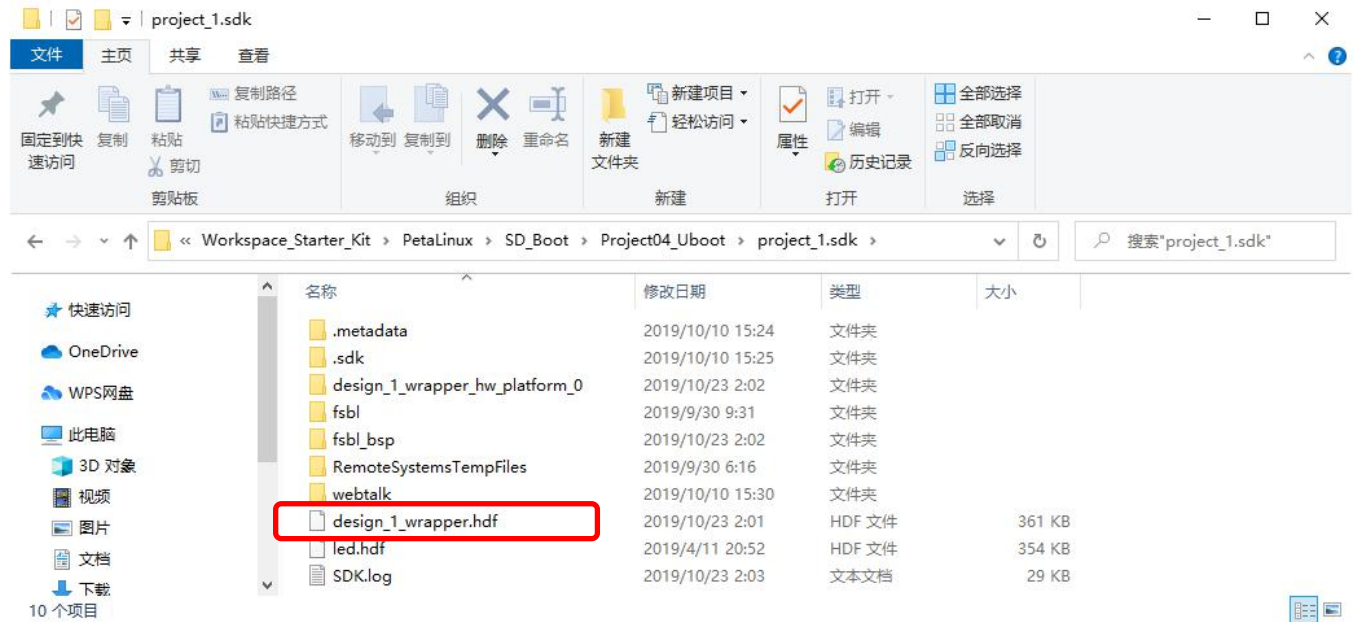


Figure 2-3. Target Hardware Info File

Copy the design_1_wrapper.hdf file into Ubuntu environment. E.g. put the file in folder \$workspace/ZYJZGW_Workspace/Linux_base_Starter_Kit_V02.sdk.

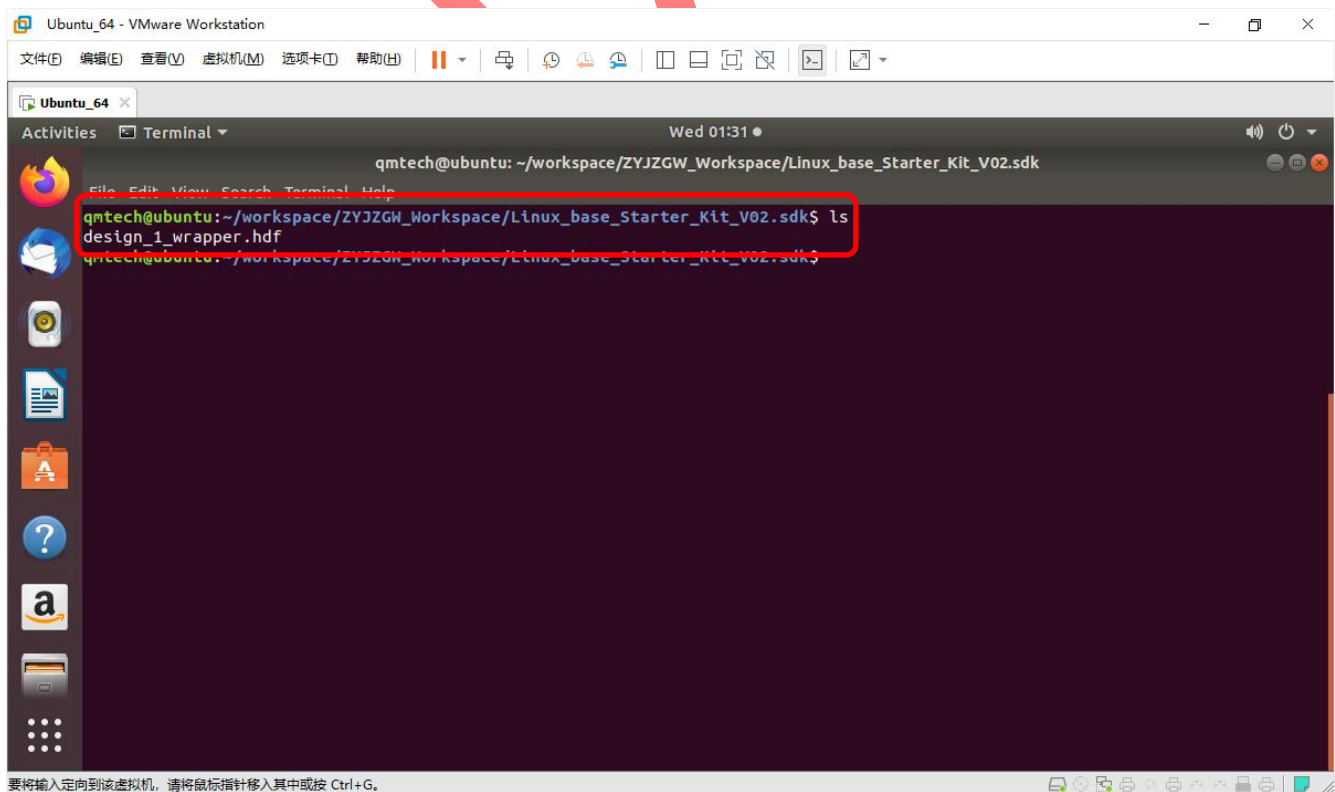


Figure 2-4. Import Hardware Info File

In our example, the PetaLinux 2019.2 package is installed in folder /opt/pkg/petalinux. Users need change the below commands according to the detailed directory that contains the PetaLinux. Type command in the terminal to source the PetaLinux: **/opt/pkg/petalinux/settings.sh**. Create a new folder named as **ZYJZGW_Starter_Kit** by typing command: **petalinux-create --type project --template zynq --name ZYJZGW_Starter_Kit**. This new folder is in the same directory as the Linux_base_Starter_Kit_V02.sdk and used as the PetaLinux workspace.

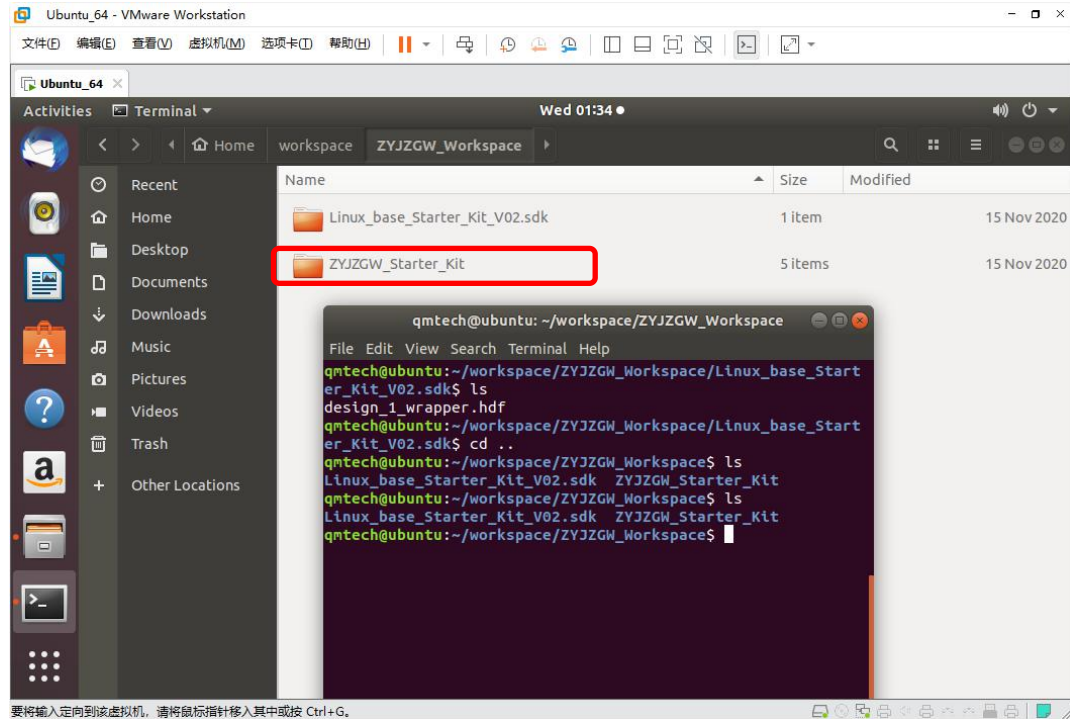


Figure 2-5. Create PetaLinux Working Folder

Then enter into the folder **ZYJZGW_Starter_Kit**.

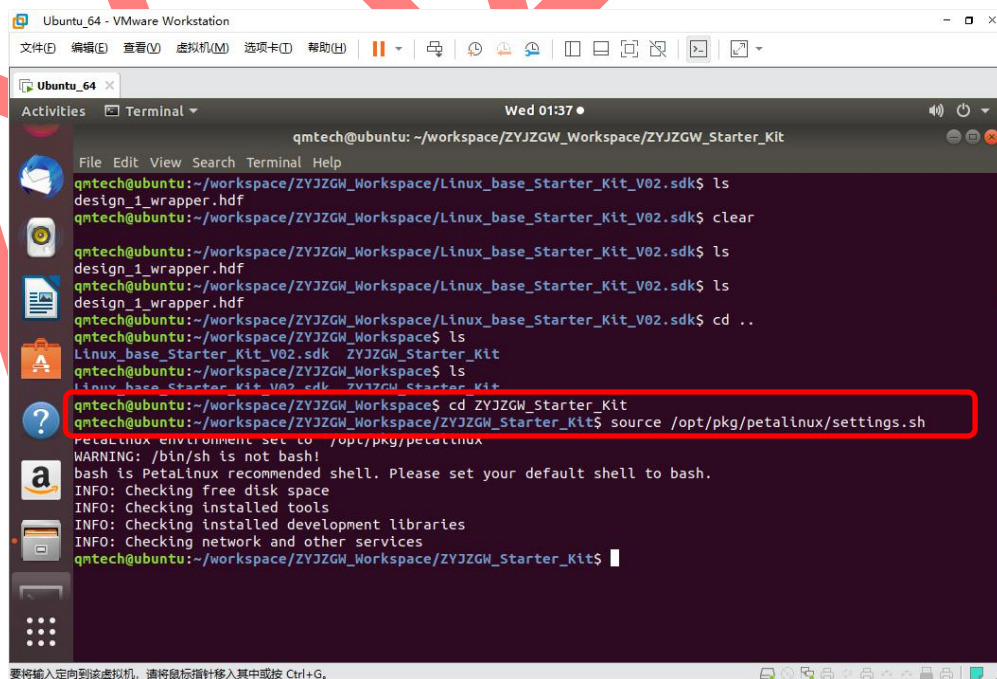


Figure 2-6. Enter Workspace

Import the hardware info into PetaLinux by command: **petalinux-config --get-hw-description ../linux_base.sdk**. Below configuration image will display and nothing needs to be changed here.

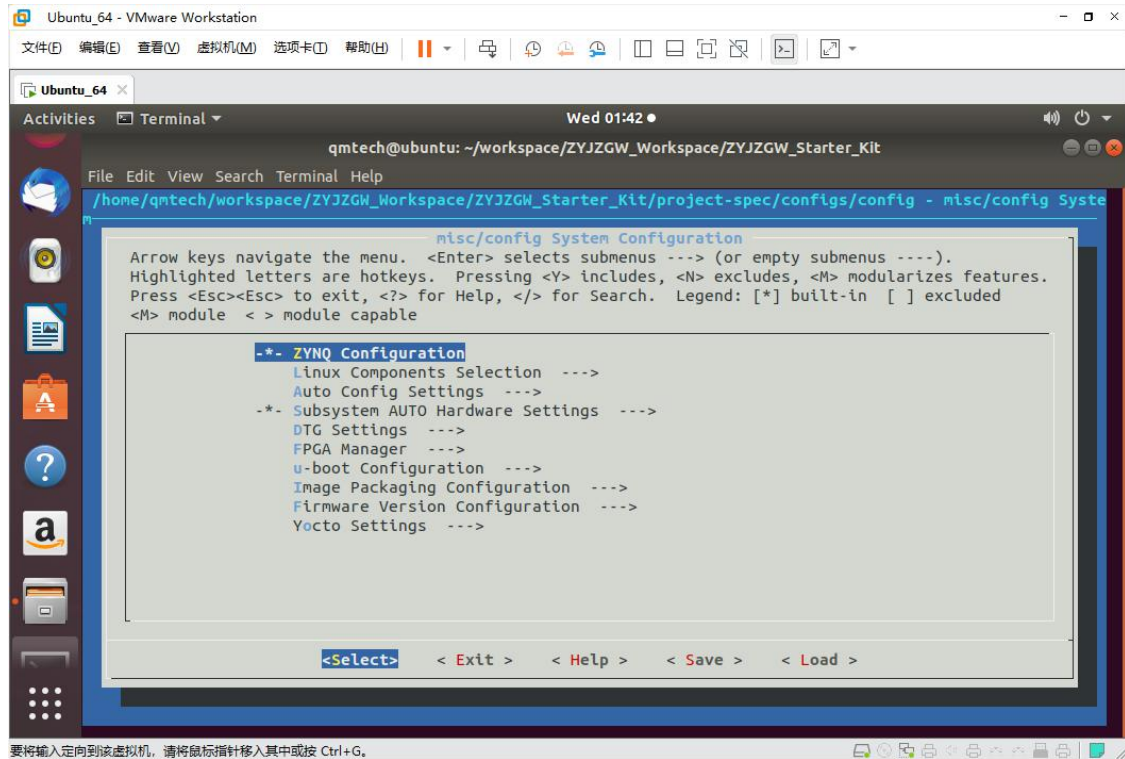


Figure 2-7. ZYNQ Configuration

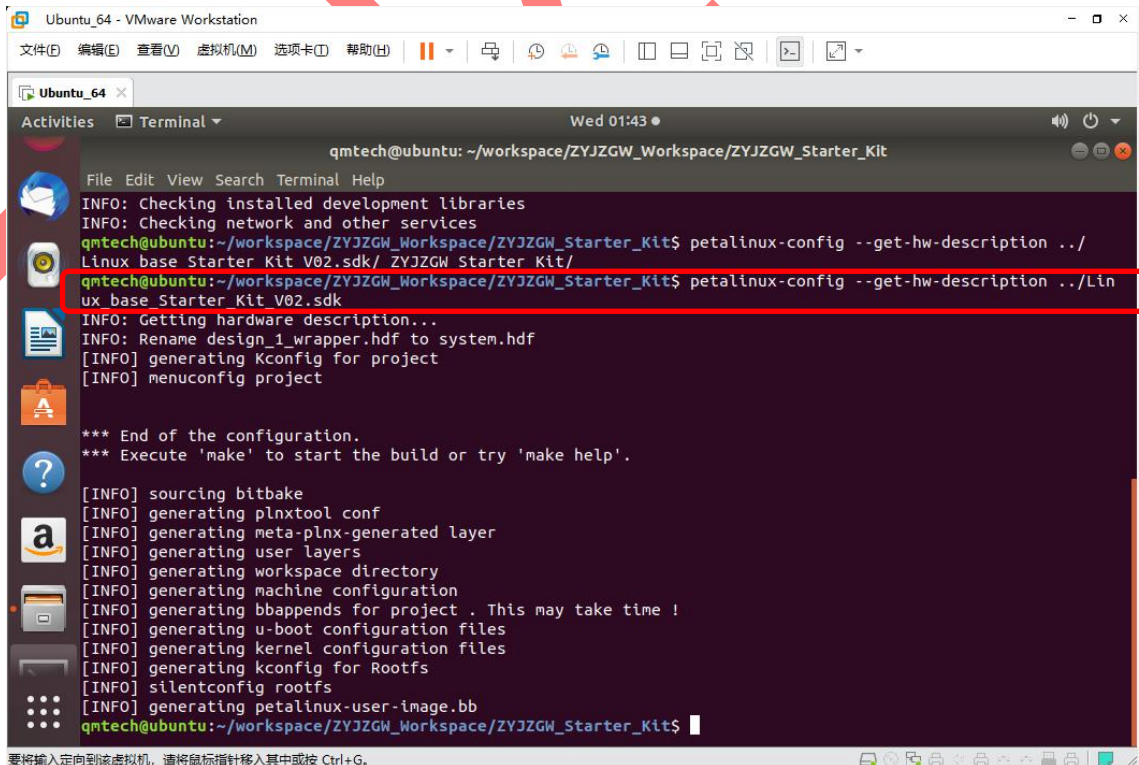
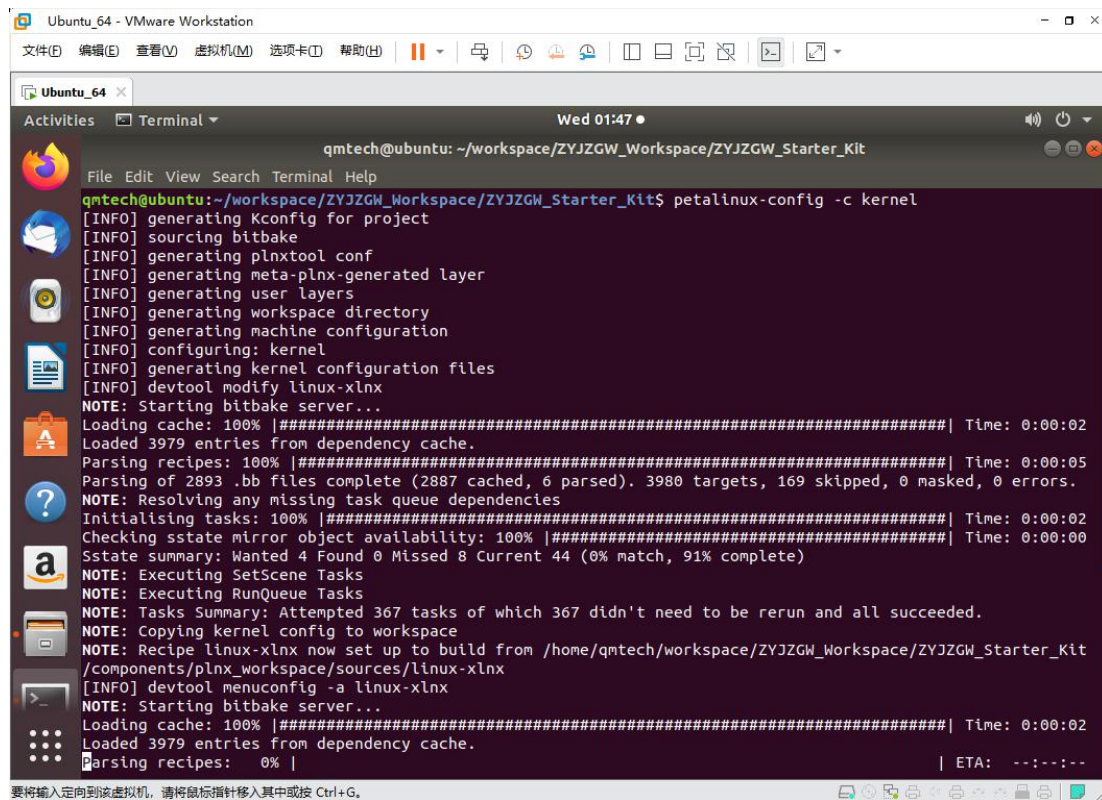


Figure 2-8. Import Hardware Info

Once the hardware info is successfully imported, users may start to configure the PetaLinux kernel by command: **petalinux-config -c kernel**
This step may take a really long time. Be patient here and make sure the VM's network status is perfect.

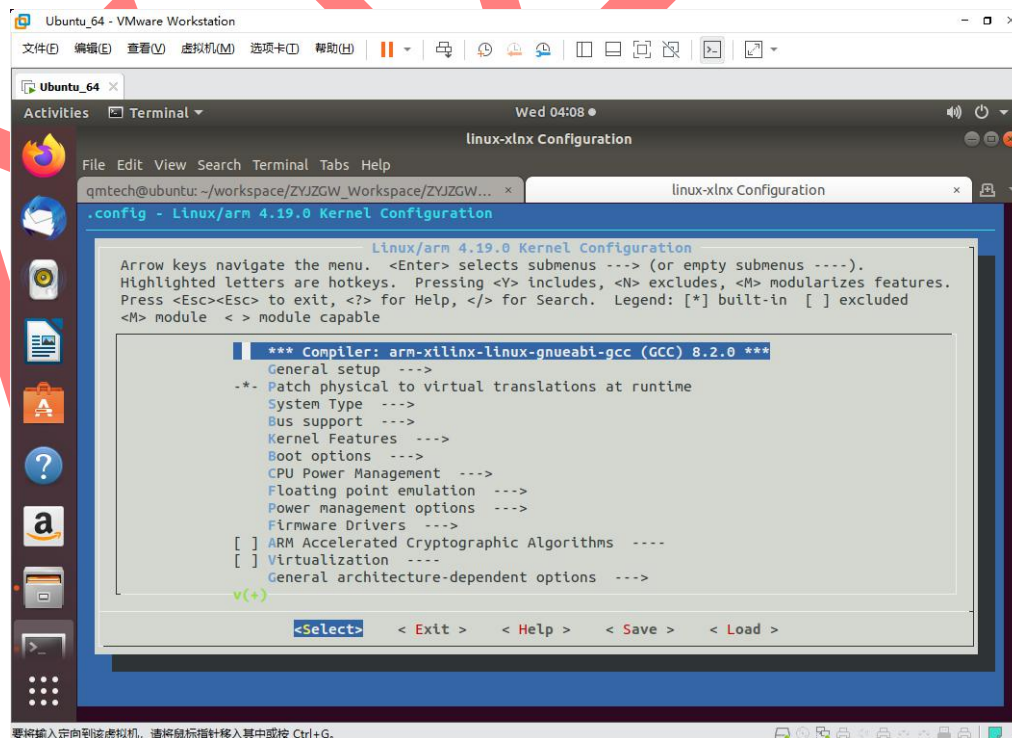


```

qmttech@ubuntu: ~/workspace/ZYJZGW_Workspace/ZYJZGW_Starter_Kit
qmttech@ubuntu:~/workspace/ZYJZGW_Workspace/ZYJZGW_Starter_Kit$ petalinux-config -c kernel
[INFO] generating Kconfig for project
[INFO] sourcing bitbake
[INFO] generating plnxtool conf
[INFO] generating meta-plnx-generated layer
[INFO] generating user layers
[INFO] generating workspace directory
[INFO] generating machine configuration
[INFO] configuring: kernel
[INFO] generating kernel configuration files
[INFO] devtool modify linux-xlnx
NOTE: Starting bitbake server...
Loading cache: 100% |#####| Time: 0:00:02
Loaded 3979 entries from dependency cache.
Parsing recipes: 100% |#####| Time: 0:00:05
Parsing of 2893 .bb files complete (2887 cached, 6 parsed). 3980 targets, 169 skipped, 0 masked, 0 errors.
NOTE: Resolving any missing task queue dependencies
Initialising tasks: 100% |#####| Time: 0:00:02
Checking sstate mirror object availability: 100% |#####| Time: 0:00:00
Sstate summary: Wanted 4 Found 0 Missed 8 Current 44 (0% match, 91% complete)
NOTE: Executing SetScene Tasks
NOTE: Executing RunQueue Tasks
NOTE: Tasks Summary: Attempted 367 tasks of which 367 didn't need to be rerun and all succeeded.
NOTE: Copying kernel config to workspace
NOTE: Recipe linux-xlnx now set up to build from /home/qmttech/workspace/ZYJZGW_Workspace/ZYJZGW_Starter_Kit
/components/plnx_workspace/sources/linux-xlnx
[INFO] devtool menuconfig -a linux-xlnx
NOTE: Starting bitbake server...
Loading cache: 100% |#####| Time: 0:00:02
Loaded 3979 entries from dependency cache.
Parsing recipes: 0% |

```

Figure 2-9. Configure PetaLinux Kernel



```

linux-xlnx Configuration
.config - Linux/arm 4.19.0 Kernel Configuration
Linux/arm 4.19.0 Kernel Configuration
Arrow keys navigate the menu. <Enter> selects submenus --- (or empty submenus ---).
Highlighted letters are hotkeys. Pressing <Y> includes, <N> excludes, <M> modularizes features.
Press <Esc><Esc> to exit, <?> for Help, </> for Search. Legend: [*] built-in [ ] excluded
<M> module < > module capable

*** Compiler: arm-xilinx-linux-gnueabi-gcc (GCC) 8.2.0 ***
General setup ---
*- Patch physical to virtual translations at runtime
System Type ---
Bus support ---
Kernel Features ---
Boot options ---
CPU Power Management ---
Floating point emulation ---
Power management options ---
Firmware Drivers ---
[ ] ARM Accelerated Cryptographic Algorithms ---
[ ] Virtualization ---
General architecture-dependent options ---

v(+)
<Select> < Exit > < Help > < Save > < Load >

```

Figure 2-10. No Need to change the Kernel Configurations

Once the PetaLinux Kernel info is successfully configured, users may start to configure the PetaLinux file system by command: **petalinux-config -c rootfs**
Below image will display and nothing needs to be changed here.

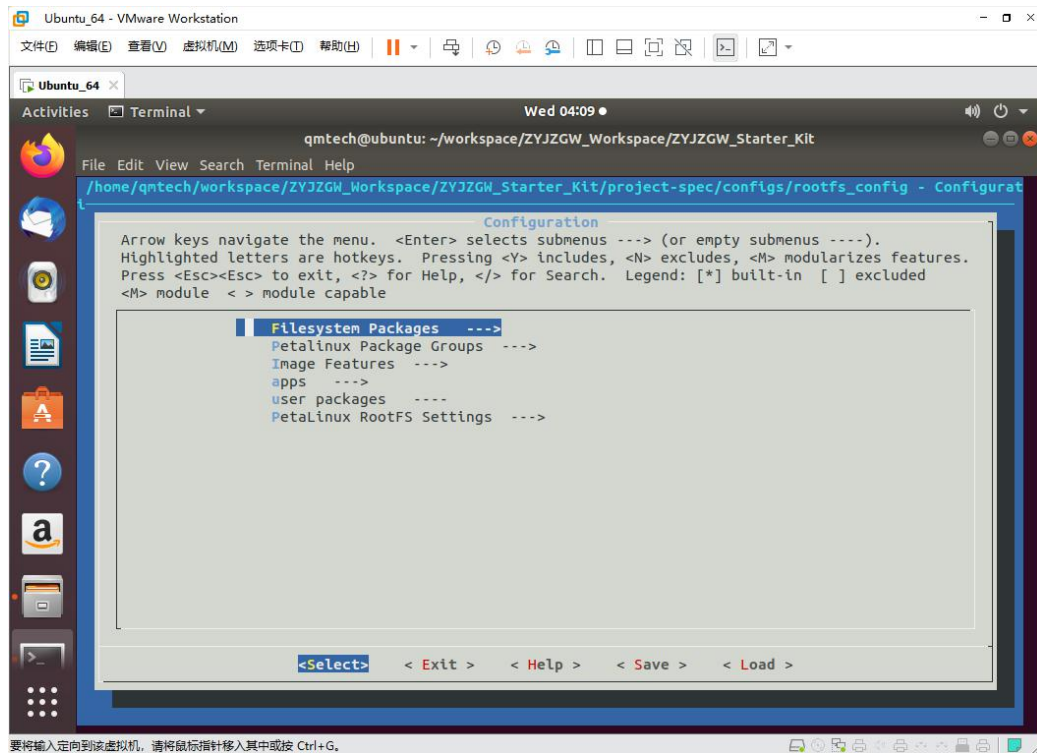


Figure 2-11. No Need to change the rootfs

Start to build the PetaLinux by command: **petalinux-build**

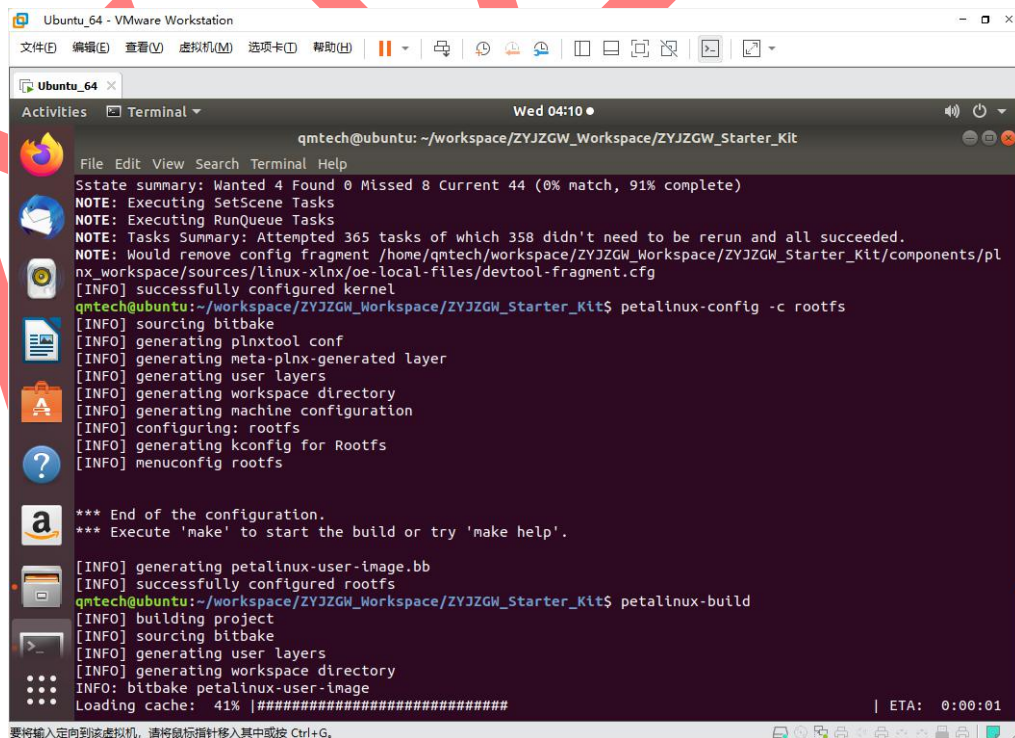
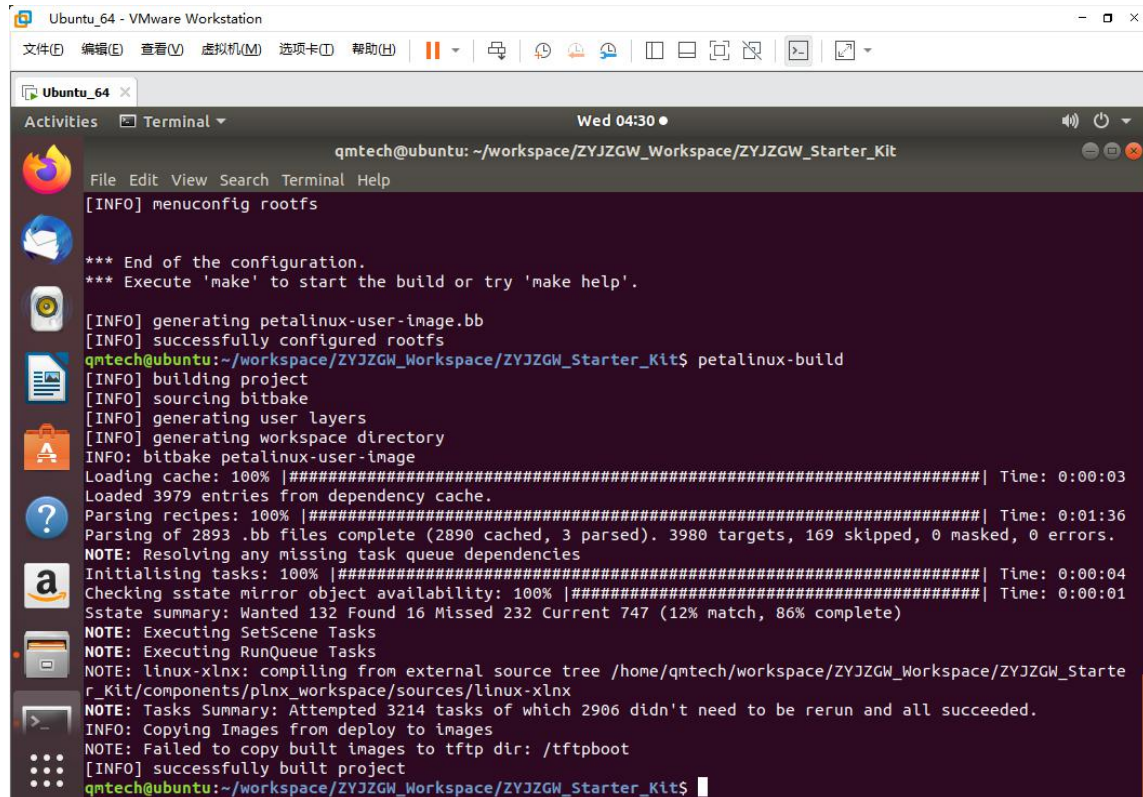


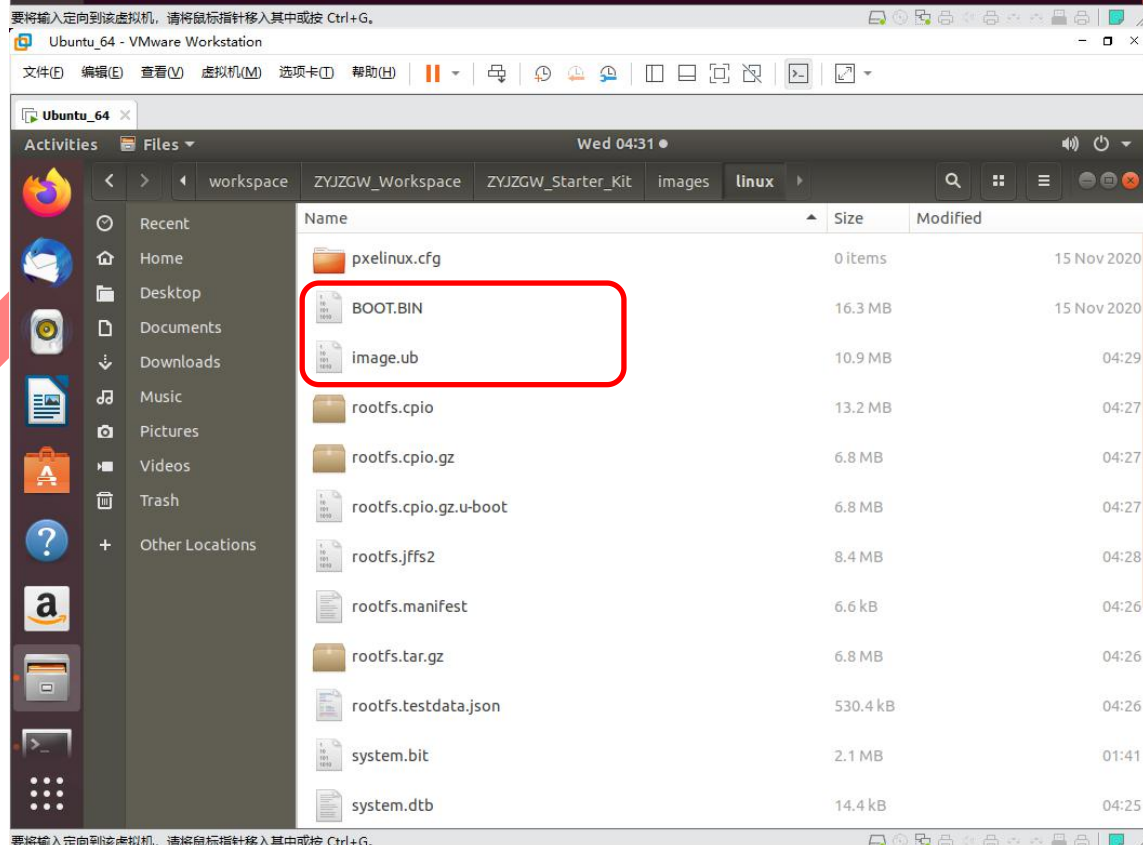
Figure 2-12. Build PetaLinux

After the PetaLinux is successfully built, users could generate the target BOOT.bin and image.ub by command: **petalinux-package --boot --fsbl ./images/linux/zynq_fsbl.elf --fpga --u-boot --force**

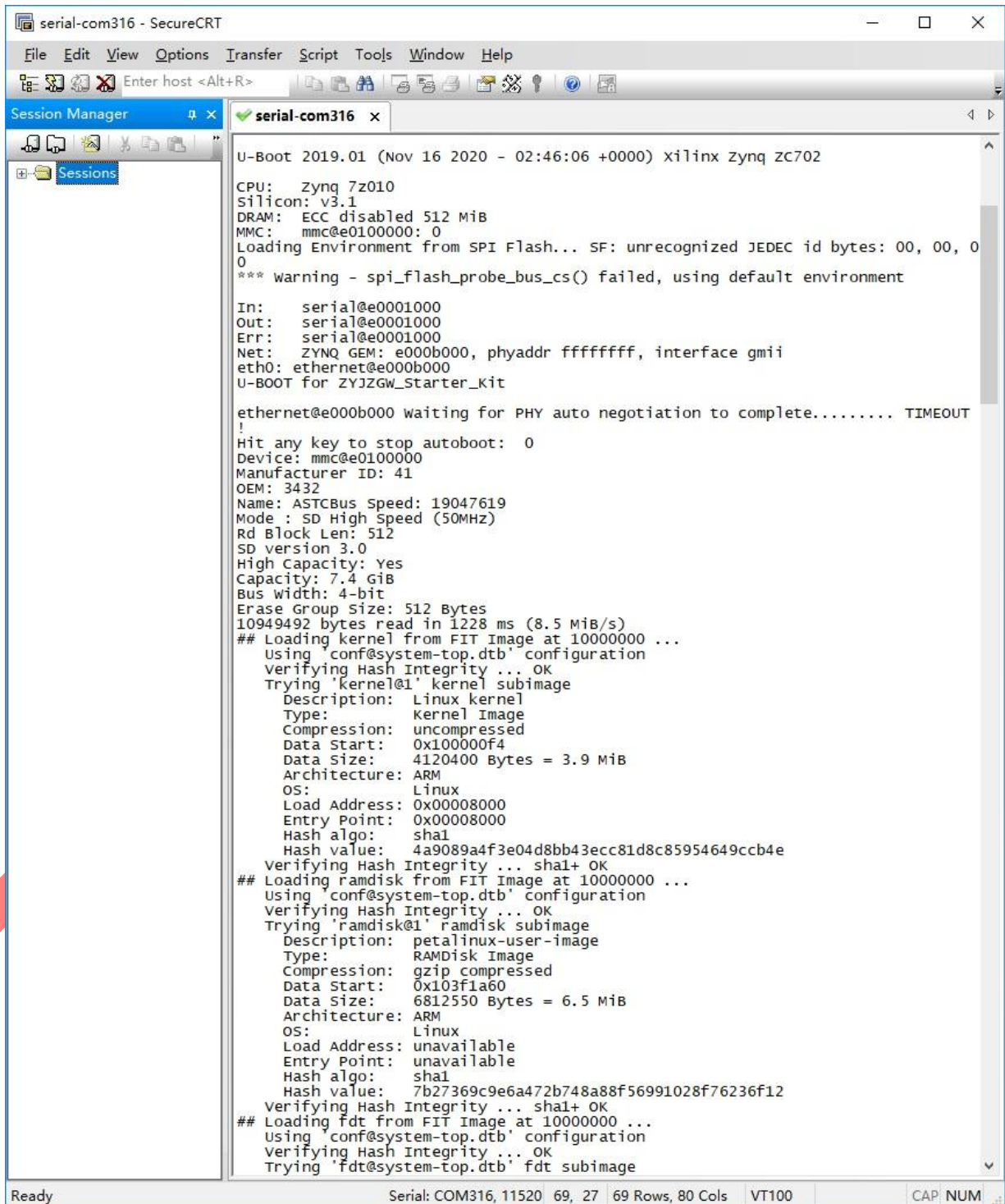


```
qmtch@ubuntu: ~/workspace/ZYJZGW_Workspace/ZYJZGW_Starter_Kit
[INFO] menuconfig rootfs
*** End of the configuration.
*** Execute 'make' to start the build or try 'make help'.

[INFO] generating petalinux-user-image.bb
[INFO] successfully configured rootfs
qmtch@ubuntu:~/workspace/ZYJZGW_Workspace/ZYJZGW_Starter_Kit$ petalinux-build
[INFO] building project
[INFO] sourcing bitbake
[INFO] generating user layers
[INFO] generating workspace directory
INFO: bitbake petalinux-user-image
Loading cache: 100% |#####| Time: 0:00:03
Loaded 3979 entries from dependency cache.
Parsing recipes: 100% |#####| Time: 0:01:36
Parsing of 2893 .bb files complete (2890 cached, 3 parsed). 3980 targets, 169 skipped, 0 masked, 0 errors.
NOTE: Resolving any missing task queue dependencies
Initialising tasks: 100% |#####| Time: 0:00:04
Checking sstate mirror object availability: 100% |#####| Time: 0:00:01
Sstate summary: Wanted 132 Found 16 Missed 232 Current 747 (12% match, 86% complete)
NOTE: Executing SetScene Tasks
NOTE: Executing RunQueue Tasks
NOTE: linux-xlnx: compiling from external source tree /home/qmtch/workspace/ZYJZGW_Workspace/ZYJZGW_Starter_Kit/components/plnx_workspace/sources/linux-xlnx
NOTE: Tasks Summary: Attempted 3214 tasks of which 2906 didn't need to be rerun and all succeeded.
INFO: Copying Images from deploy to images
NOTE: Failed to copy built images to tftp dir: /tftpboot
[INFO] successfully built project
qmtch@ubuntu:~/workspace/ZYJZGW_Workspace/ZYJZGW_Starter_Kit$
```



Copy the BOOT.bin and image.ub into MicroSD card and then insert the MicroSD card into ZYNQ7000 Starter Kit. Plug the MiniUSB cable into the board and then power on the power source switch. Below image shows the serial output from the board.



```
serial-com316 - SecureCRT
File Edit View Options Transfer Script Tools Window Help
Enter host <Alt+R>
Session Manager
Sessions
serial-com316 x
U-Boot 2019.01 (Nov 16 2020 - 02:46:06 +0000) xilinx Zynq zc702
CPU: Zynq 7z010
Silicon: v3.1
DRAM: ECC disabled 512 MiB
MMC: mmc@e0100000: 0
Loading Environment from SPI Flash... SF: unrecognized JEDEC id bytes: 00, 00, 0
0
*** warning - spi_flash_probe_bus_cs() failed, using default environment

In: serial@e0001000
Out: serial@e0001000
Err: serial@e0001000
Net: ZYNQ GEM: e000b000, phyaddr ffffffff, interface gmii
eth0: ethernet@e000b000
U-BOOT for ZYJZGW_Starter_Kit

ethernet@e000b000 waiting for PHY auto negotiation to complete..... TIMEOUT
!
Hit any key to stop autoboot: 0
Device: mmc@e0100000
Manufacturer ID: 41
OEM: 3432
Name: ASTCbus Speed: 19047619
Mode : SD High Speed (50MHz)
Rd Block Len: 512
SD version 3.0
High Capacity: Yes
Capacity: 7.4 GiB
Bus width: 4-bit
Erase Group Size: 512 Bytes
10949492 bytes read in 1228 ms (8.5 MiB/s)
## Loading kernel from FIT Image at 10000000 ...
Using 'conf@system-top.dtb' configuration
Verifying Hash Integrity ... OK
Trying 'kernel@1' kernel subimage
Description: Linux kernel
Type: Kernel Image
Compression: uncompressed
Data Start: 0x100000f4
Data Size: 4120400 Bytes = 3.9 MiB
Architecture: ARM
OS: Linux
Load Address: 0x00008000
Entry Point: 0x00008000
Hash algo: sha1
Hash value: 4a9089a4f3e04d8bb43ecc81d8c85954649ccb4e
Verifying Hash Integrity ... sha1+ OK
## Loading ramdisk from FIT Image at 10000000 ...
Using 'conf@system-top.dtb' configuration
Verifying Hash Integrity ... OK
Trying 'ramdisk@1' ramdisk subimage
Description: petalinux-user-image
Type: RAMDisk Image
Compression: gzip compressed
Data Start: 0x103f1a60
Data Size: 6812550 Bytes = 6.5 MiB
Architecture: ARM
OS: Linux
Load Address: unavailable
Entry Point: unavailable
Hash algo: sha1
Hash value: 7b27369c9e6a472b748a88f56991028f76236f12
Verifying Hash Integrity ... sha1+ OK
## Loading fdt from FIT Image at 10000000 ...
Using 'conf@system-top.dtb' configuration
Verifying Hash Integrity ... OK
Trying 'fdt@system-top.dtb' fdt subimage
```

Ready Serial: COM316, 11520 69, 27 69 Rows, 80 Cols VT100 CAP NUM


```
serial-com316 - SecureCRT
File Edit View Options Transfer Script Tools Window Help
Enter host <Alt+R>
Session Manager
serial-com316 x
Trying 'fdt@system-top.dtb' fdt subimage
Description: Flattened Device Tree Blob
Type: Flat Device Tree
Compression: uncompressed
Data Start: 0x103ee144
Data Size: 14427 Bytes = 14.1 KiB
Architecture: ARM
Hash algo: sha1
Hash value: 415706275fb54305a546e6411bdb8bda27fd8615
Verifying Hash Integrity ... sha1+ OK
Booting using the fdt blob at 0x103ee144
Loading Kernel Image ... OK
Loading Ramdisk to 07980000, end 07fff386 ... OK
Loading Device Tree to 07979000, end 0797f85a ... OK

Starting kernel ...

Booting Linux on physical CPU 0x0
Linux version 4.19.0-xilinx-v2019.2 (oe-user@oe-host) (gcc version 8.2.0 (GCC))
#1 SMP PREEMPT Wed Jun 23 11:12:30 UTC 2021
CPU: ARMv7 Processor [413fc090] revision 0 (ARMv7), cr=18c5387d
CPU: PIPT / VIPT nonaliasing data cache, VIPT aliasing instruction cache
OF: fdt: Machine model: xlnx,zynq-7000
earlycon: cdns0 at MMIO 0xe0001000 (options '115200n8')
bootconsole [cdns0] enabled
Memory policy: Data cache writealloc
cma: Reserved 16 MiB at 0x1f000000
random: get_random_bytes called from start_kernel+0x80/0x3c4 with crng_init=0
percpu: Embedded 16 pages/cpu @(ptrval) s35916 r8192 d21428 u65536
Built 1 zonelists, mobility grouping on. Total pages: 130048
Kernel command line: console=ttyPS0,115200 earlycon
Dentry cache hash table entries: 65536 (order: 6, 262144 bytes)
Inode-cache hash table entries: 32768 (order: 5, 131072 bytes)
Memory: 486460K/524288K available (6144K kernel code, 204K rwddata, 1604K rodata,
1024K init, 132K bss, 21444K reserved, 16384K cma-reserved, 0K highmem)
Virtual kernel memory layout:
vector : 0xffff0000 - 0xffff1000 ( 4 kB)
fixmap : 0xffc00000 - 0xffff0000 (3072 kB)
vmalloc : 0xe0800000 - 0xff800000 ( 496 MB)
lowmem : 0xc0000000 - 0xe0000000 ( 512 MB)
pkmap : 0xbfe00000 - 0xc0000000 ( 2 MB)
modules : 0xbf000000 - 0xbfe00000 ( 14 MB)
.text : 0x(ptrval) - 0x(ptrval) (7136 kB)
.init : 0x(ptrval) - 0x(ptrval) (1024 kB)
.data : 0x(ptrval) - 0x(ptrval) ( 205 kB)
.bss : 0x(ptrval) - 0x(ptrval) ( 133 kB)
rcu: Preemptible hierarchical RCU implementation.
rcu: RCU restricting CPUs from NR_CPUS=4 to nr_cpu_ids=2.
Tasks RCU enabled.
rcu: Adjusting geometry for rcu_fanout_leaf=16, nr_cpu_ids=2
NR_IRQS: 16, nr_irqs: 16, preallocated irq: 16
efuse mapped to (ptrval)
slcr mapped to (ptrval)
L2C: platform modifies aux control register: 0x72360000 -> 0x72760000
L2C: DT/platform modifies aux control register: 0x72360000 -> 0x72760000
L2C-310 erratum 769419 enabled
L2C-310 enabling early BRESP for Cortex-A9
L2C-310 full line of zeros enabled for Cortex-A9
L2C-310 ID prefetch enabled, offset 1 lines
L2C-310 dynamic clock gating enabled, standby mode enabled
L2C-310 cache controller enabled, 8 ways, 512 kB
L2C-310: CACHE_ID 0x410000c8, AUX_CTRL 0x76760001
zynq_clock_init: clk starts at (ptrval)
zynq clock init
sched_clock: 64 bits at 333MHz, resolution 3ns, wraps every 4398046511103ns
clocksource: arm_global_timer: mask: 0xffffffffffffffff max_cycles: 0x4ce07af025
, max_idle_ns: 440795209040 ns
Switching to timer-based delay loop, resolution 3ns
clocksource: ttc_clocksource: mask: 0xffff max_cycles: 0xffff, max_idle_ns: 5375
Ready
Serial: COM316, 11520 69, 27 69 Rows, 80 Cols VT100 CAP NUM
```



```
serial-com316 - SecureCRT
File Edit View Options Transfer Script Tools Window Help
Enter host <Alt+R>
Session Manager
Sessions
serial-com316 x
Switching to timer-based delay loop, resolution 3ns
clocksource: ttc_clocksource: mask: 0xffff max_cycles: 0xffff, max_idle_ns: 5375
38477 ns
timer #0 at (ptrval), irq=17
Console: colour dummy device 80x30
Calibrating delay loop (skipped), value calculated using timer frequency.. 666.6
6 BogomIPS (lpj=3333333)
pid_max: default: 32768 minimum: 301
Mount-cache hash table entries: 1024 (order: 0, 4096 bytes)
Mountpoint-cache hash table entries: 1024 (order: 0, 4096 bytes)
CPU: Testing write buffer coherency: ok
CPU0: Spectre v2: using BPIALL workaround
CPU0: thread -1, cpu 0, socket 0, mpidr 80000000
Setting up static identity map for 0x100000 - 0x100060
rcu: Hierarchical SRCU implementation.
smp: Bringing up secondary CPUs ...
CPU1: thread -1, cpu 1, socket 0, mpidr 80000001
CPU1: Spectre v2: using BPIALL workaround
smp: Brought up 1 node, 2 CPUs
SMP: Total of 2 processors activated (1333.33 BogomIPS).
CPU: All CPU(s) started in SVC mode.
devtmpfs: initialized
VFP support v0.3: implementor 41 architecture 3 part 30 variant 9 rev 4
clocksource: jiffies: mask: 0xffffffff max_cycles: 0xffffffff, max_idle_ns: 1911
2604462750000 ns
futex hash table entries: 512 (order: 3, 32768 bytes)
pinctrl core: initialized pinctrl subsystem
NET: Registered protocol family 16
DMA: preallocated 256 KiB pool for atomic coherent allocations
cpuidle: using governor menu
hw-breakpoint: found 5 (+1 reserved) breakpoint and 1 watchpoint registers.
hw-breakpoint: maximum watchpoint size is 4 bytes.
zynq-ocm f800c000.ocmc: ZYNQ OCM pool: 256 KiB @ 0x(ptrval)
zynq-pinctrl 700.pinctrl: zynq pinctrl initialized
e0001000.serial: ttyPS0 at MMIO 0xe0001000 (irq = 25, base_baud = 6249999) is a
xuartps
console [ttyPS0] enabled
console [ttyPS0] enabled
bootconsole [cdns0] disabled
bootconsole [cdns0] disabled
vgaarb: loaded
SCSI subsystem initialized
usbcore: registered new interface driver usbfs
usbcore: registered new interface driver hub
usbcore: registered new device driver usb
media: Linux media interface: v0.10
videodev: Linux video capture interface: v2.00
pps_core: LinuxPPS API ver. 1 registered
pps_core: Software ver. 5.3.6 - Copyright 2005-2007 Rodolfo Giometti <giometti@1
inux.it>
PTP clock support registered
EDAC MC: Ver: 3.0.0
FPGA manager framework
Advanced Linux Sound Architecture Driver Initialized.
clocksource: Switched to clocksource arm_global_timer
NET: Registered protocol family 2
tcp_listen_portaddr_hash hash table entries: 512 (order: 0, 6144 bytes)
TCP established hash table entries: 4096 (order: 2, 16384 bytes)
TCP bind hash table entries: 4096 (order: 3, 32768 bytes)
TCP: Hash tables configured (established 4096 bind 4096)
UDP hash table entries: 256 (order: 1, 8192 bytes)
UDP-Lite hash table entries: 256 (order: 1, 8192 bytes)
NET: Registered protocol family 1
RPC: Registered named UNIX socket transport module.
RPC: Registered udp transport module.
RPC: Registered tcp transport module.
RPC: Registered tcp NFSv4.1 backchannel transport module.
Trying to unpack rootfs image as initramfs...
Freeing initrd memory: 6656K
Ready Serial: COM316, 115200, 69, 27 69 Rows, 80 Cols VT100 CAP NUM
```

```
serial-com316 - SecureCRT
File Edit View Options Transfer Script Tools Window Help
Enter host <Alt+R>
Session Manager
Sessions
serial-com316 x
Freeing initrd memory: 6656K
hw perfevents: no interrupt-affinity property for /pmu@f8891000, guessing.
hw perfevents: enabled with armv7_cortex_a9 PMU driver, 7 counters available
workingset: timestamp_bits=30 max_order=17 bucket_order=0
jffs2: version 2.2. (NAND) (SUMMARY)  漏 2001-2006 Red Hat, Inc.
io scheduler noop registered
io scheduler deadline registered
io scheduler cfq registered (default)
io scheduler mq-deadline registered
io scheduler kyber registered
dma-pl330 f8003000.dmac: Loaded driver for PL330 DMAC-241330
dma-pl330 f8003000.dmac:          DBUFF-128x8bytes Num_Chans-8 Num_Peri-4 Num_Events-16
brd: module loaded
loop: module loaded
m25p80 spi0.0: unrecognized JEDEC id bytes: 00, 00, 00
libphy: Fixed MDIO Bus: probed
CAN device driver interface
libphy: MACB_mii_bus: probed
Generic PHY e000b000.ethernet-ffffffff:00: attached PHY driver [Generic PHY] (mi
i_bus:phy_addr=e000b000.ethernet-ffffffff:00, irq=POLL)
macb e000b000.ethernet eth0: Cadence GEM rev 0x00020118 at 0xe000b000 irq 27 (00
:0a:35:00:1e:53)
e1000e: Intel(R) PRO/1000 Network Driver - 3.2.6-k
e1000e: Copyright(c) 1999 - 2015 Intel Corporation.
ehci_hcd: USB 2.0 'Enhanced' Host Controller (EHCI) Driver
ehci-pci: EHCI PCI platform driver
usbcore: registered new interface driver usb-storage
i2c /dev entries driver
cdns-wdt f8005000.watchdog: Xilinx Watchdog Timer with timeout 10s
EDAC MC: ECC not enabled
Xilinx Zynq CpuIdle Driver started
sdhci: Secure Digital Host Controller Interface driver
sdhci: Copyright(c) Pierre Ossman
sdhci-pltfm: SDHCI platform and OF driver helper
mmc0: SDHCI controller on e0100000.mmc [e0100000.mmc] using ADMA
ledtrig-cpu: registered to indicate activity on CPUs
usbcore: registered new interface driver usbhid
usbhid: USB HID core driver
fpga_manager fpga0: Xilinx Zynq FPGA Manager registered
NET: Registered protocol family 10
Segment Routing with IPv6
sit: IPv6, IPv4 and MPLS over IPv4 tunneling driver
NET: Registered protocol family 17
can: controller area network core (rev 20170425 abi 9)
NET: Registered protocol family 29
can: raw protocol (rev 20170425)
can: broadcast manager protocol (rev 20170425 t)
can: netlink gateway (rev 20170425) max_hops=1
Registering SWP/SWPB emulation handler
of-fpga-region fpga-full: FPGA Region probed
hctosys: unable to open rtc device (rtc0)
of_cfs_init
of_cfs_init: OK
ALSA device list:
  No soundcards found.
Freeing unused kernel memory: 1024K
Run /init as init process
mmc0: Problem switching card into high-speed mode!
mmc0: new SDHC card at address 0001
INIT: mmcblk0: mmc0:0001 ASTC 7.37 GiB
      mmcblk0: p1
version 2.88 booting
Starting udev
udevd[726]: starting version 3.2.5
random: udevd: uninitialized urandom read (16 bytes read)
random: udevd: uninitialized urandom read (16 bytes read)
random: udevd: uninitialized urandom read (16 bytes read)
udevd[727]: starting udevd-3.2.5

Ready
Serial: COM316, 11520 69, 27 69 Rows, 80 Cols VT100 CAP NUM
```



```
serial-com316 - SecureCRT
File Edit View Options Transfer Script Tools Window Help
Enter host <Alt+R>
Session Manager
Sessions
serial-com316 x
mmcblk0: p1
version 2.88 booting
Starting udevd
udev[726]: starting version 3.2.5
random: udevd: uninitialized urandom read (16 bytes read)
random: udevd: uninitialized urandom read (16 bytes read)
random: udevd: uninitialized urandom read (16 bytes read)
udev[727]: starting eudev-3.2.5
FAT-fs (mmcblk0p1): volume was not properly unmounted. Some data may be corrupt.
Please run fsck.
hwclock: can't open '/dev/misc/rtc': No such file or directory
wed Jun 23 11:26:49 UTC 2021
hwclock: can't open '/dev/misc/rtc': No such file or directory
urandom_read: 2 callbacks suppressed
random: dd: uninitialized urandom read (512 bytes read)
Configuring packages on first boot....
(This may take several minutes. Please do not power off the machine.)
Running postinst /etc/rpm-postinsts/100-sysvinit-inittab...
update-rc.d: /etc/init.d/run-postinsts exists during rc.d purge (continuing)
Removing any system startup links for run-postinsts ...
/etc/rcs.d/s99run-postinsts
INIT: Entering runlevel: 5
Configuring network interfaces... IPV6: ADDRCONF(NETDEV_UP): eth0: link is not ready
udhcpd: started, v1.29.2
udhcpd: sending discover
udhcpd: sending discover
udhcpd: sending discover
udhcpd: no lease, forking to background
done.
Starting haveged: haveged: listening socket at 3
haveged: haveged starting up

Starting Dropbear SSH server: random: dropbearkey: uninitialized urandom read (32
2 bytes read)
random: dropbearkey: uninitialized urandom read (32 bytes read)
Generating 2048 bit rsa key, this may take a while...
haveged: haveged: ver: 1.9.4; arch: generic; vend: ; build: (gcc 8.2.0 CTV); collect: 128k

haveged: haveged: cpu: (VC); data: 16K (D); inst: 16K (D); idx: 12/40; sz: 15012/57848

haveged: haveged: tot tests(BA8): A:1/1 B:1/1 continuous tests(B): last entropy estimate 7.99967

haveged: haveged: fills: 0, generated: 0

random: crng init done
Public key portion is:
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQDQn6HHqWzhkoibo9CbrRomxB0fHFINUsoLGGgdZ9bqACwJ6qXozYitQPvNkRtediyOr+ePM1urjQ42aGdmwtkYHZ+wlrQzxc8HZky06Er23I46c6iGjYfDzx60tHeezmkDN6h5VxzHwDuzDWFa4aM16OU5hQryIw6K6SHAc8aJMYJnx1M4aqe025i4YAm9U6TJ9Lp9gu8NBixb1PEI4D4CJtaKL+gToFyxZsM0T0xtD6hLzdgGLKZV1L4zeaJoMxbm2kgxA/czUMDCzFRGQ2a9sflw+Fho9RyzsOBZzf76CGcfjP8dmpfHzpDyt3mXqsGB83Hi5HQYZUrU3aUFR5Y9 root@ZYJZGW_Starter_Kit
Fingerprint: sha1!! a0:11:be:ec:f6:d2:44:5b:6e:69:ff:a1:8c:fa:2d:6b:a6:74:18:57
dropbear.
hwclock: can't open '/dev/misc/rtc': No such file or directory
Starting internet superserver: inetd.
Starting syslogd/klogd: done
Starting tcf-agent: OK

PetaLinux 2019.2 ZYJZGW_Starter_Kit /dev/ttyPS0
ZYJZGW_Starter_Kit login: root
Password:
root@ZYJZGW_Starter_Kit:~#

Ready Serial: COM316, 11520 69, 28 69 Rows, 80 Cols VT100 CAP NUM
```

Figure 2-13. PetaLinux Boot Log

2.2 Test the Ethernet Under Linux Environment

Plug the ethernet cable into the Starter Kit and power on it. Below log will be displayed on the terminal tool when the ethernet link is ready. Users may type below ethernet test related commands to check the status of ethernet interface.

Figure 2-14. Log Info

3. Reference

- [1] ug585-Zynq-7000-TRM.pdf
- [2] ds187-XC7Z010-XC7Z020-Data-Sheet.pdf
- [3] ug865-Zynq-7000-Pkg-Pinout.pdf
- [4] MT41K256M16TW-107:P.pdf
- [5] tps563201.pdf

ZYJZGW

4. Revision

Doc. Rev.	Date	Comments
0.1	03/06/2021	Initial Version.
1.0	23/06/2021	V1.0 Formal Release.