Lab3. HW Linux Application Build

ZynqMP PL영역에 Pmod8ld, PmodAls, PmoTmp2등을 구동하기 위해 Xilinx HW IP(axi gpio, axi quad spi, axi iic)들을 갖는 Vivado Project를 구성하고, 이들 HW IP 들을 위한 Linux Application들을 SDK와 recipe들을 통해 Build 한다. 각 HW IP(GPIO, SPI, I2C Controller)들을 Userspace에서 직접적으로 다루는 방법과 Kernel Module을 만들어 다루는 방법(Out of Tree Build)을 익힌다. Kernel Module중의 하나를 In Tree방식으로 다루어 Upstream Kernel Source을 수정하는 방법을 익힌다.

1. HW preparation

보드에 Usb-to-Uart와 Pmod Module들을 연결한다. PMod96보드의 PMOD_A, PMOD_B, PMOD_C에 각각 Pmod8ld, PmodAls, PmodTmp2을 연결한다. PmodAls는 PMOD_B의 top에 연결한다. Usb-to-Uart와 host의 usb 포트를 연결한다.

2. Export Vivado Project

Ultra96v1(hw2_v1.tcl) 또는 Ultra96v2(hw2_v2.tcl) Vivado Project를 만든다.

```
$ cd ~/work/zynqmp_linux/
$ vivado -nolog -nojournal -mode batch -source hw2_v1.tcl
$ cd hw2
$ vivado hw2.xpr
```

또는

```
$ cd ~/work/zynqmp_linux/
$ vivado -nolog -nojournal -mode batch -source hw2_v2.tcl
$ cd hw2
$ vivado hw2.xpr
```

Bitstream을 생성하고 HW export를 한다.

3. Petalinux Project Update with new HW

다음의 명령을 사용하여 hw2/ 의 xsa파일을 기초로 하여 Petalinux Project(ultra96)의 HW를 변경한다.

```
$ cd ~/work/zynqmp_linux/petalinux/ultra96
$ petalinux-config --silentconfig --get-hw-description=../../hw2/
```

4. New Device Tree Generation

다음의 명령으로 new HW에 기초한 Device Tree를 Generation 한다.

```
$ cd ~/work/zynqmp_linux/petalinux/ultra96
$ petalinux-build -c device-tree -x configure
```

Petalinux Project(ultra96) 폴더 아래의 components/plnx_workspace/device-tree/device-tree/pl.dtsi의 내용을 확인한다. pl.dtsi은 새로 추가된 PL영역의 HW IP들에 대한 Device Tree 정보를 가지고 있다.

5. Kernel Configuration for SPIDEV

SPIDEV는 SPI Controller를 Userspace에서 다루기 위해 필요하고 다음의 과정을 거쳐 Kernel에 추가한다.

```
$ cd ~/work/zynqmp_linux/petalinux/ultra96
$ petalinux-config -c kernel
```

Drivers → SPI support에서 User mode SPI device support를 활성화시킨다.

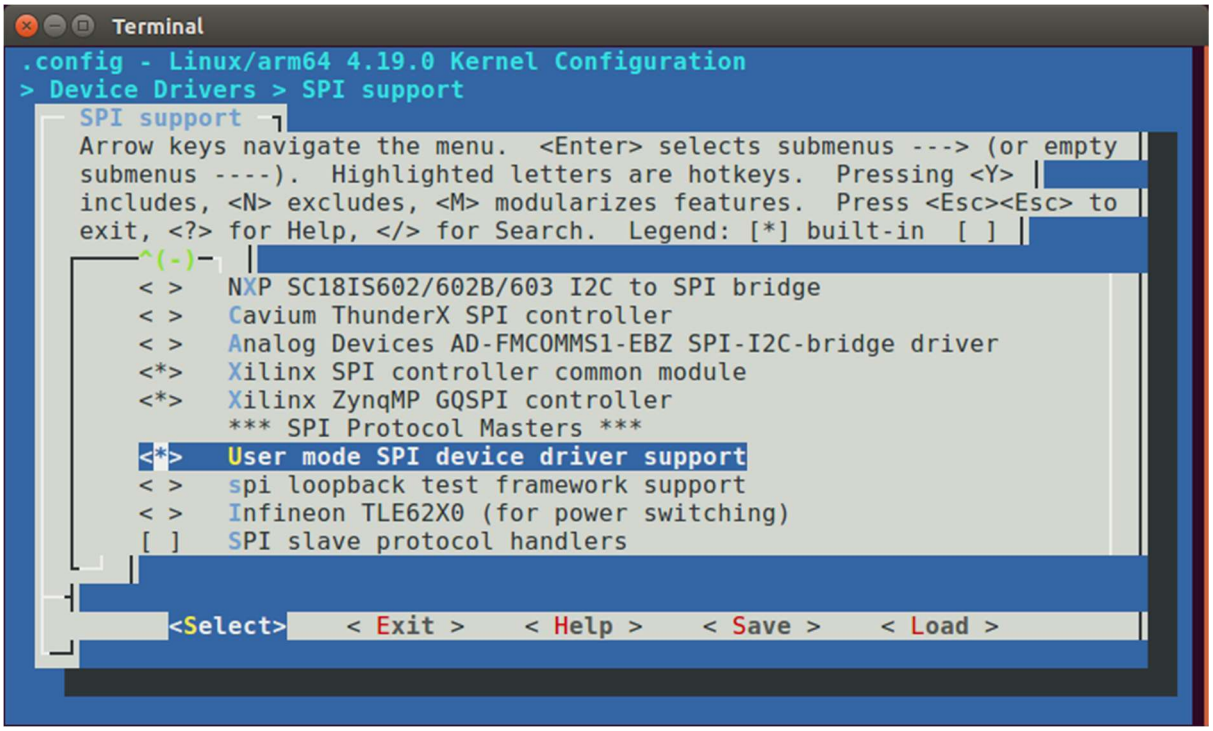

그림 1 Kernel Configuration(SPIDEV)

```
$ petalinux-config
```

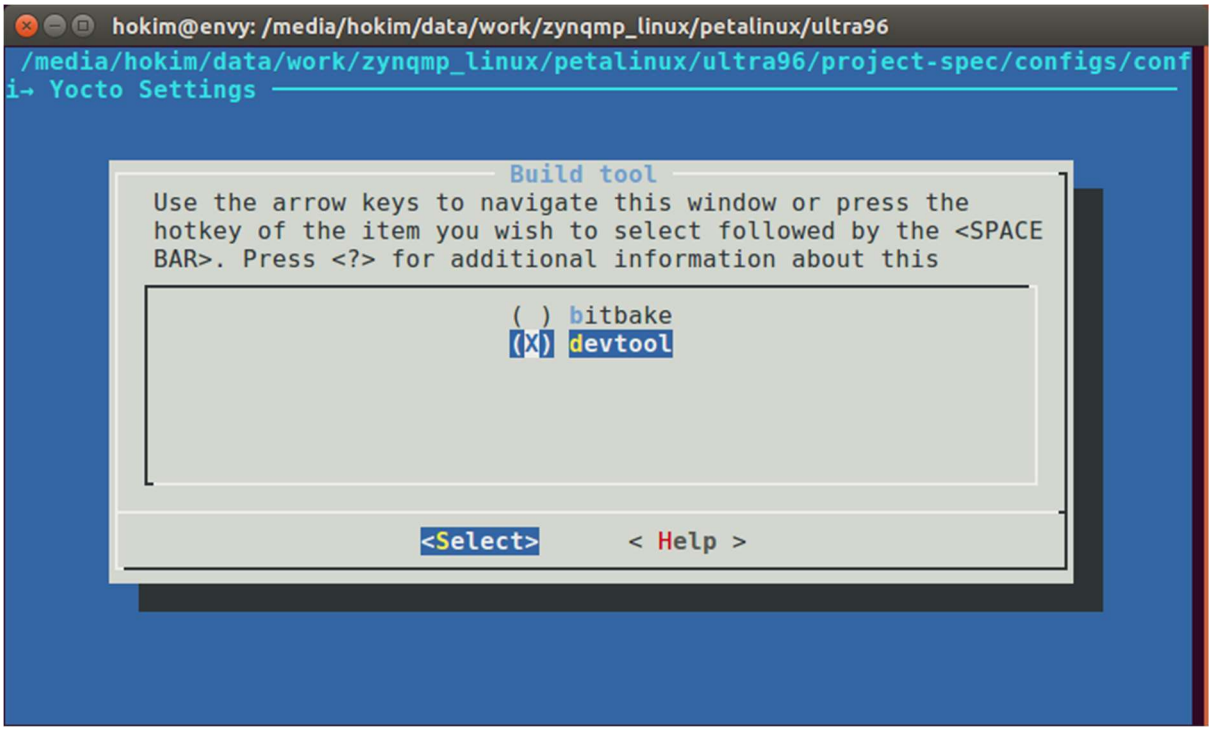

그림 2 Petalinux Configuration(devtool)

```
$ petalinux-build -c kernel -x update-recipe
$ petalinux-config
```

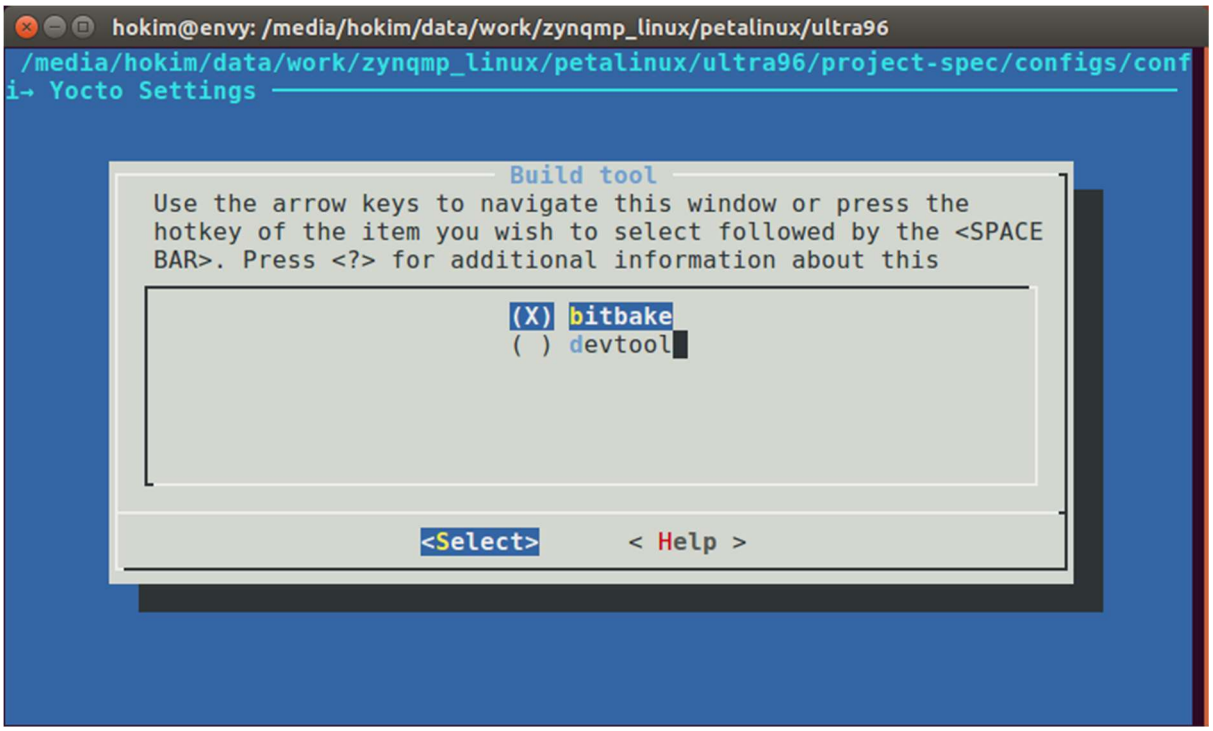

그림 3 Petalinux Configuration(bitbake)

```
$ petalinux-build -c kernel -x reset
```

6. Device Tree Modification

다음 명령으로 Device Tree를 수정한다.

```
$ cd ~/work/zynqmp_linux/petalinux/ultra96
$ vi project-spec/meta-user/recipe-bsp/device-tree/files/system-user.dtsi
```

axi_quad_spi_0 node아래에 spidev node를 다음처럼 추가한다.

Ultra96v1은 line5-11이 Ultra96v2는 line49-55가 추가된 부분이다.

그림 4 Device Tree Modification1(Ultra96v1)

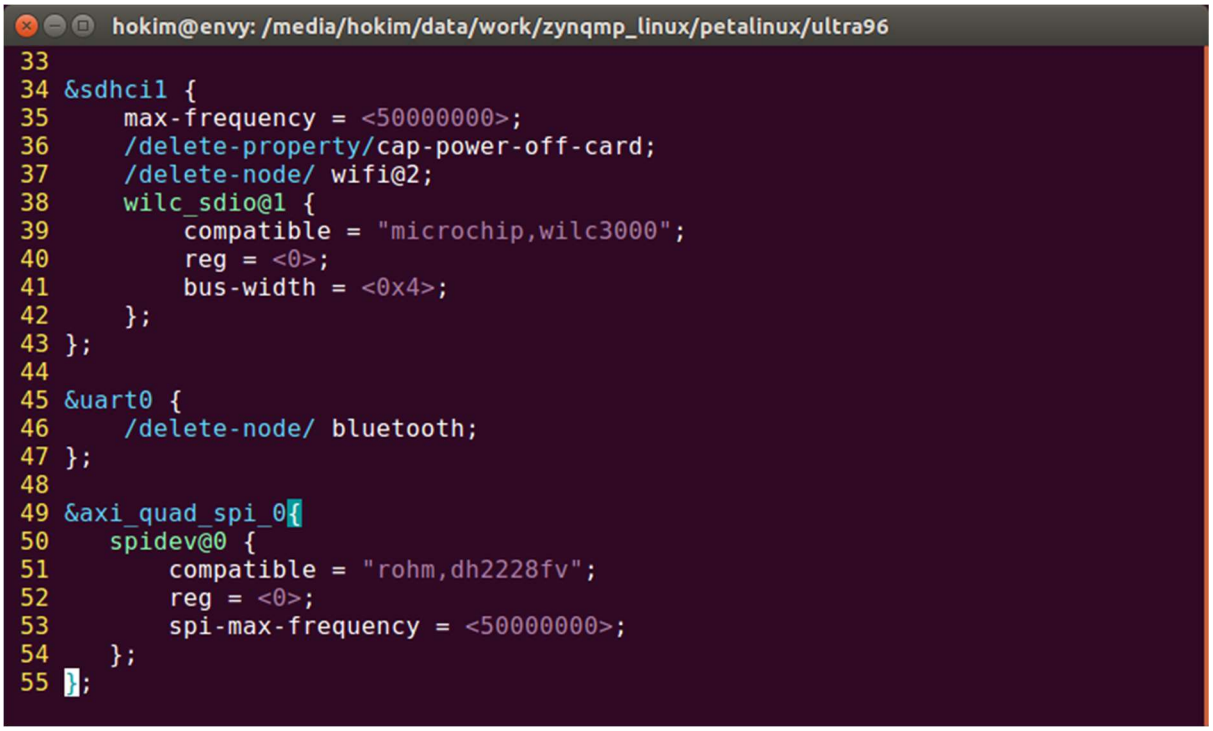

그림 5 Device Tree Modification1(Ultra96v2)

7. Update BOOT.BIN, image.ub

새로운 HW를 위한 BOOT.BIN과 image.ub를 다음과 같이 Update 한다.

```
$ cd ~/work/zynqmp_linux/petalinux/ultra96
$ petalinux-build -c virtual/boot-bin
$ petalinux-package --force –boot --fsbl images/linux/zynqmp_fsbl.elf --u-boot
images/linux/u-boot.elf --pmufw images/linux/pmufw.elf --fpga
images/linux/system.bit
$ scp images/linux/{BOOT.BIN,image.ub} root@172.30.1.39:/media/card
```

8. HW Check after Reboot

GPIO, SPI, I2C Controller Linux Driver들에 의해 Userspace로 Export되는 HW 정보를 Check하도록 한다.

```
ultra96$ ls -l /sys/class/gpio/gpio*
=========================ultra96v1=========================
lrwxrwxrwx 1 root root 0 Feb 19 19:49 /sys/class/gpio/gpio356 -
> ../../devices/platform/amba/ff0a0000.gpio/gpiochip1/gpio/gpio356
lrwxrwxrwx 1 root root 0 Feb 19 19:49 /sys/class/gpio/gpiochip326 -
> ../../devices/platform/amba/ff030000.i2c/i2c-1/i2c-7/7-005e/gpio/gpiochip326
lrwxrwxrwx 1 root root 0 Feb 19 19:49 /sys/class/gpio/gpiochip330 -
> ../../devices/platform/amba/ff0a0000.gpio/gpio/gpiochip330
lrwxrwxrwx 1 root root 0 Feb 19 19:49 /sys/class/gpio/gpiochip504 -
> ../../devices/platform/amba_pl@0/80000000.gpio/gpio/gpiochip504
=========================ultra96v2=========================
lrwxrwxrwx 1 root root 0 Feb 15 10:56 /sys/class/gpio/gpio356 -
> ../../devices/platform/amba/ff0a0000.gpio/gpiochip1/gpio/gpio356
lrwxrwxrwx 1 root root 0 Jan  1  1970 /sys/class/gpio/gpiochip330 -
> ../../devices/platform/amba/ff0a0000.gpio/gpio/gpiochip330
lrwxrwxrwx 1 root root 0 Jan  1  1970 /sys/class/gpio/gpiochip504 -
> ../../devices/platform/amba_pl@0/80000000.gpio/gpio/gpiochip504
=========================ultra96v1=========================
ultra96$ cat /sys/class/gpio/gpiochip326/label
tps65086-gpio
ultra96$ cat /sys/class/gpio/gpiochip330/label
zynqmp_gpio
ultra96$ cat /sys/class/gpio/gpiochip504/label
/amba_pl@0/gpio@80000000
=========================ultra96v2=========================
ultra96$ cat /sys/class/gpio/gpiochip330/label
zynqmp_gpio
ultra96$ cat /sys/class/gpio/gpiochip504/label
/amba_pl@0/gpio@80000000
```

```
ultra96$ ls -l /sys/class/spi_master/spi*
lrwxrwxrwx 1 root root 0 Jan   1   1970 /sys/class/spi_master/spi0 -
> ../../devices/platform/amba_pl@0/80020000.axi_quad_spi/spi_master/spi0
lrwxrwxrwx 1 root root 0 Jan   1   1970 /sys/class/spi_master/spi1 -
> ../../devices/platform/amba/ff040000.spi/spi_master/spi1
lrwxrwxrwx 1 root root 0 Jan   1   1970 /sys/class/spi_master/spi2 -
> ../../devices/platform/amba/ff050000.spi/spi_master/spi2
ultra96$ ls /sys/class/spi_master/spi0/
device   of_node   power   spi0.0   statistics   subsystem   uevent
ultra96$ i2cdetect -l
i2c-3    i2c          i2c-1-mux (chan_id 0)          I2C adapter
i2c-10   i2c          i2c-1-mux (chan_id 7)          I2C adapter
i2c-1    i2c          Cadence I2C at ff030000         I2C adapter
i2c-8    i2c          i2c-1-mux (chan_id 5)          I2C adapter
i2c-6    i2c          i2c-1-mux (chan_id 3)          I2C adapter
i2c-4    i2c          i2c-1-mux (chan_id 1)          I2C adapter
i2c-2    i2c          ZynqMP DP AUX                      I2C adapter
i2c-0    i2c          xiic-i2c                       I2C adapter
i2c-9    i2c          i2c-1-mux (chan_id 6)          I2C adapter
i2c-7    i2c          i2c-1-mux (chan_id 4)          I2C adapter
i2c-5    i2c          i2c-1-mux (chan_id 2)          I2C adapter
```

```
ultra96$ i2cdetect -y -r 0
     0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00:          -- -- -- -- -- -- -- -- -- -- -- -- --
10: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
20: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
30: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
40: -- -- -- -- -- -- -- -- -- -- -- 4b -- -- -- --
50: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
60: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
70: -- -- -- -- -- -- -- --
```

9. SDK HW Application Build & Test

SDK를 사용하여 Application을 다음과 같이 Build한다.

```
$ cd ~/work/zynqmp_linux/petalinux/workspaces
$ unset LD_LIBRARY_PATH
$ source ~/sdk/environment-setup-aarch64-xilinx-linux
$ cd gpio_test
$ mkdir build
$ cd build
$ cmake ..
$ make
$ scp gpio_test root@172.30.1.39:.
$ cd ../../spi_test
$ mkdir build
$ cd build
$ cmake ..
$ make
$ scp spi_test root@172.30.1.39:.
$ cd ../../i2c_test
$ mkdir build
$ cd build
$ cmake ..
$ make
$ scp i2c_test root@172.30.1.39:.
```

보드에서 Test는 다음과 같다. gpio_test 의 결과는 Pmod8ld의 led 점멸로 확인한다. spi_test 와 i2c_test는 1초 간격으로 각각의 adc값을 계속 출력하기 때문에 중지하려면 Control-C를 눌러야 한다.

```
ultra96$ cd ~
ultra96$ ./gpio_test 3
ultra96$ ./spi_test
light = 44
ultra96$ ./i2c_test
temp = 3176
ultra96$ rm gpio_test spi_test i2c_test
```

10. HW Application Recipe Build & Test

~/work/zynqmp_linux/petalinux/meta-custom/recipes-apps 아래에 있는 gpiotest, spitest, i2test recipes들을 다음과 같이 Build하고 보드에서 Test 한다.

```
$ cd ~/work/zynqmp_linux/petalinux/ultra96
$ petalinux-build -c gpiotest
$ petalinux-build -c spitest
$ petalinux-build -c i2ctest
$ petalinux-build -c package-index
```

```
$ cd build/tmp/deploy/rpm
$ python3 -m http.server 5678
```

```
ultra96$ dnf -y --refresh install gpiotest spitest i2ctest
ultra96$ gpio_test 5
ulta96$ spi_test
light = 44
ultra96$ i2c_test
temp = 3176
```

11. SDK HW Kernel Module Build

Kernel Module들을 사용하기 위해 Device Tree를 수정한다.

```
$ ~/work/zynqmp_linux/petalinux/ultra96
$ vi project-spec/meta-user/recipes-bsp/device-tree/files/system-user.dtsi
```

Ultra96v1은 line5-30이 Ultra96v2는 line49-74가 수정된 부분이다.

그림 6 Device Tree Modification2(Ultra96v1)

그림 7 Device Tree Modification2(Ultra96v2)

Device Tree를 다시 Build하면 image.ub가 Update되며 이를 보드로 scp한다.

```
$ petalinux-build -c device-tree
$ scp images/linux/image.ub root@172.30.1.39:/media/card
```

SDK를 사용하여 Kernel Module들을 Build한다. 먼저 SDK가 설치된 폴더의 Kernel Source에 Out of Tree Build를 할 수 있도록 Setup한다. (make modules_prepare)

```
$ cd ~/sdk
$ unset LD_LIBRARY_PATH
$ source environment-setup-aarch64-xilinx-linux
$ cd sysroots/aarch64-xilinx-linux/usr/src/kernel
$ make modules_prepare
$ cd ~/work/zynqmp_linux/petalinux/workspaces
$ cd pmod8ld
$ KERNEL_SRC=$SDKTARGETSYSROOT/usr/src/kernel make
$ scp pmod8ld.ko root@172.30.1.39:/lib/modules/4.19.0-xilinx-v2019.2/extra
$ make clean
$ cd ../pmodals
$ KERNEL_SRC=$SDKTARGETSYSROOT/usr/src/kernel make
$ scp pmodals.ko root@172.30.1.39:/lib/modules/4.19.0-xilinx-v2019.2/extra
$ make clean
$ cd ../pmodtmp2
$ KERNEL_SRC=$SDKTARGETSYSROOT/usr/src/kernel make
$ scp pmodtmp2.ko root@172.30.1.39:/lib/modules/4.19.0-xilinx-v2019.2/extra
$ make clean
```

보드에서 module을 추가하고 boot시 자동 load되도록 하기위해 다음의 과정을 수행하고 다시 boot한다.

```
ultra96$ depmod -a
ultra96$ echo pmod8ld > /etc/modules-load.d/pmod8ld.conf
ultra96$ echo pmodals > /etc/modules-load.d/pmodals.conf
ultra96$ echo pmodtmp2 > /etc/modules-load.d/pmodtmp2.conf
ultra96$ reboot
```

보드에 다시 접속하여 Kernel Module이 load되었는지 확인하고 Test를 수행한다.

다음 Test를 위해 install된 Kernel Module들을 cleanup한다.

```
ultra96$ lsmod
Module                  Size  Used by
pmod8ld                16384  0
pmodtmp2               16384  0
pmodals                16384  0
ultra96$ cd /sys/devices/platform/pmod8ld
ultra96$ cat bits
0
ultra96$ echo 10 > bits
ultra96$ cat bits
10
ultra96$ cd /sys/class/spi_master/spi0/spi0.0
ultra96$ cat adc
42
ultra96$ cd /sys/class/i2c-adapter/i2c-0/0-004b
ultra96$ cat adc
3144
ultra96$ rm   /etc/modules-load.d/pmod*.conf
ultra96$ rm /lib/modules/4.19.0-xilinx-v2019.2/extra/pmod*.ko
ultra96$ depmod -a
```

12. HW Kernel Module Recipe Build & Test

~/work/zynqmp_linux/petalinux/meta-custom/recipes-modules 아래에 있는 Pmod8ld, PmodAls, PmodTmp2 recipes들을 다음과 같이 Build하고 보드에서 Test한다. Test를 종료하고 나서 PmodTmp2 Kernel Module을 제거한다.

```
$ cd ~/work/zynqmp_linux/petalinux/ultra96
$ petalinux-build -c pmod8ld
$ petalinux-build -c pmodals
$ petalinux-build -c pmodtmp2
$ petalinux-build -c package-index
```

```
$ cd build/tmp/deploy/rpm
$ python3 -m http.server 5678
```

```
ultra96$ dnf -y --refresh install kernel-module-pmod8ld kernel-module-pmodals
kernel-module-pmodtmp2
ultra96$ rpm -ql kernel-module-pmod8ld-4.19.0-xilinx-v2019.2
/etc
/etc/modules-load.d
/etc/modules-load.d/pmod8ld.conf
/lib
/lib/modules
/lib/modules/4.19.0-xilinx-v2019.2
/lib/modules/4.19.0-xilinx-v2019.2/extra
/lib/modules/4.19.0-xilinx-v2019.2/extra/pmod8ld.ko
ulta96$ rpm -ql kernel-module-pmodals-4.19.0-xilinx-v2019.2
/etc
/etc/modules-load.d
/etc/modules-load.d/pmodals.conf
/lib
/lib/modules
/lib/modules/4.19.0-xilinx-v2019.2
/lib/modules/4.19.0-xilinx-v2019.2/extra
/lib/modules/4.19.0-xilinx-v2019.2/extra/pmodals.ko
```

```
ultra96$ rpm -ql kernel-module-pmodtmp2-4.19.0-xilinx-v2019.2
/etc
/etc/modules-load.d
/etc/modules-load.d/pmodtmp2.conf
/lib
/lib/modules
/lib/modules/4.19.0-xilinx-v2019.2
/lib/modules/4.19.0-xilinx-v2019.2/extra
/lib/modules/4.19.0-xilinx-v2019.2/extra/pmodtmp2.ko
ultra96$ cd /sys/devices/platform/pmod8ld
ultra96$ cat bits
0
ultra96$ echo 10 > bits
ultra96$ cat bits
10
ultra96$ cd /sys/class/spi_master/spi0/spi0.0
ultra96$ cat adc
42
ultra96$ cd /sys/class/i2c-adapter/i2c-0/0-004b
ultra96$ cat adc
3144
ultra96$ dnf -y --refresh remove kernel-module-pmodtmp2
```

13. Upstream Kernel Source Modification

Upstream Kernel Source를 unpack, patch, configuration하기위해 다음의 과정을 수행한다.

```
$ cd ~/work/zynqmp_linux/petalinux/ultra96
$ petalinux-build -c kernel -x configure
$ petalinux-config
```

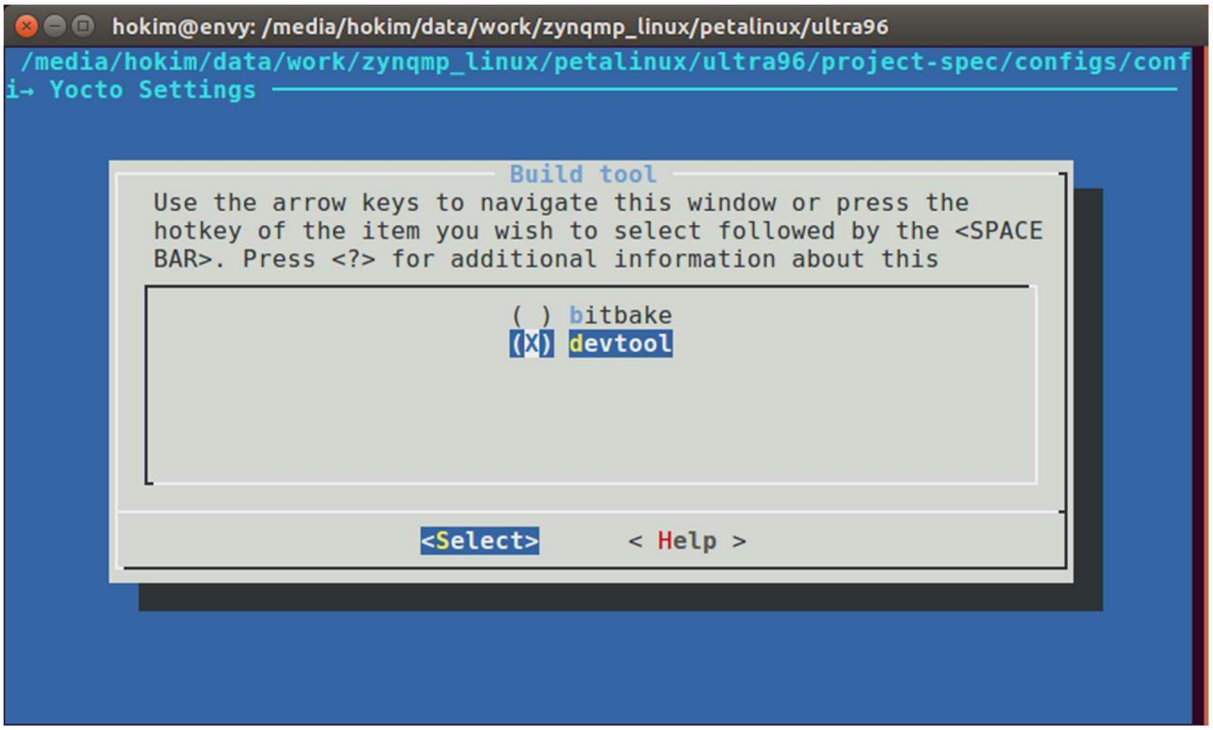

그림 8 Petalinux Configuration(devtool)

```
$ petalinux-build -c kernel -x modify
```

Upstream Kernel Source가 components/plnx_workspace/sources/linux-xlnx에 있음을 확인할 수 있다. PmodTmp2 Kernel Module을 In Tree하기 위해 Source를 다음과 같이 수정한다.

```
$ cd
~/work/zynqmp_linux/petalinux/ultra96/components/plnx_workspace/sources/linux
-xlnx
$ mkdir drivers/staging/pmods
$ vi drivers/staging/pmods/Kconfig
```

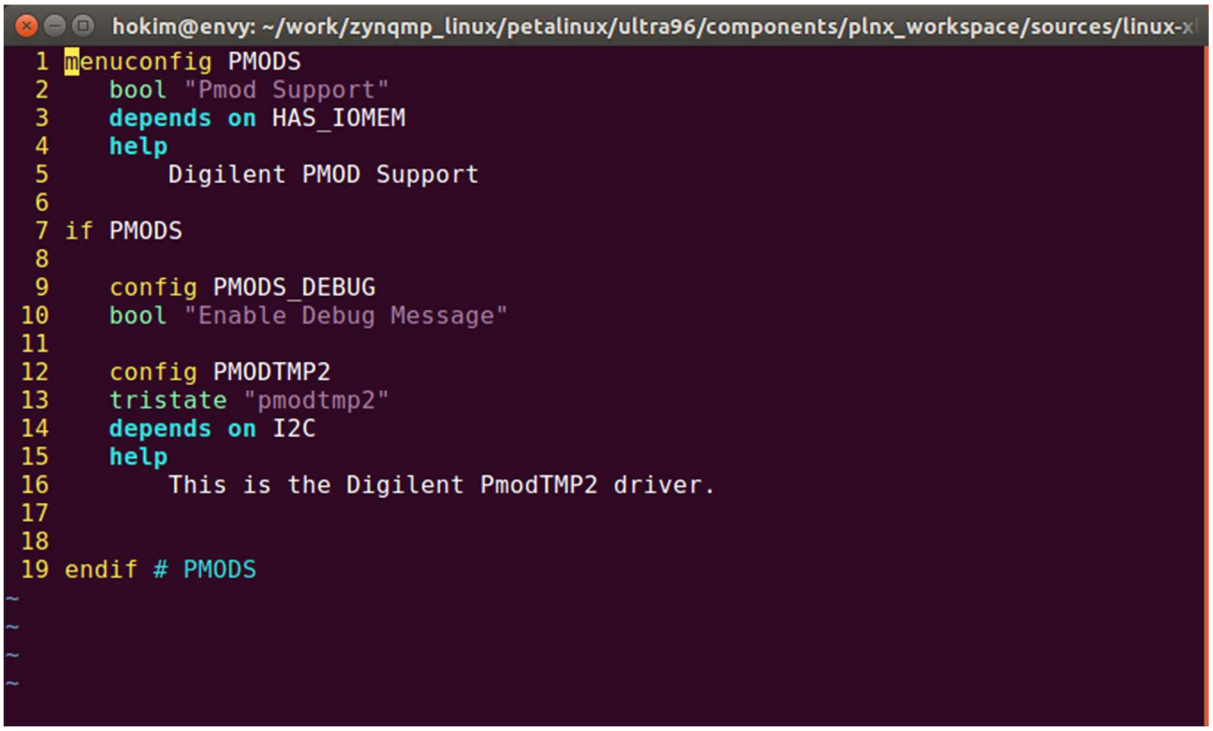

그림 9 drivers/staging/pmods/Kconfig

$ vi drivers/staging/pmods/Makefile

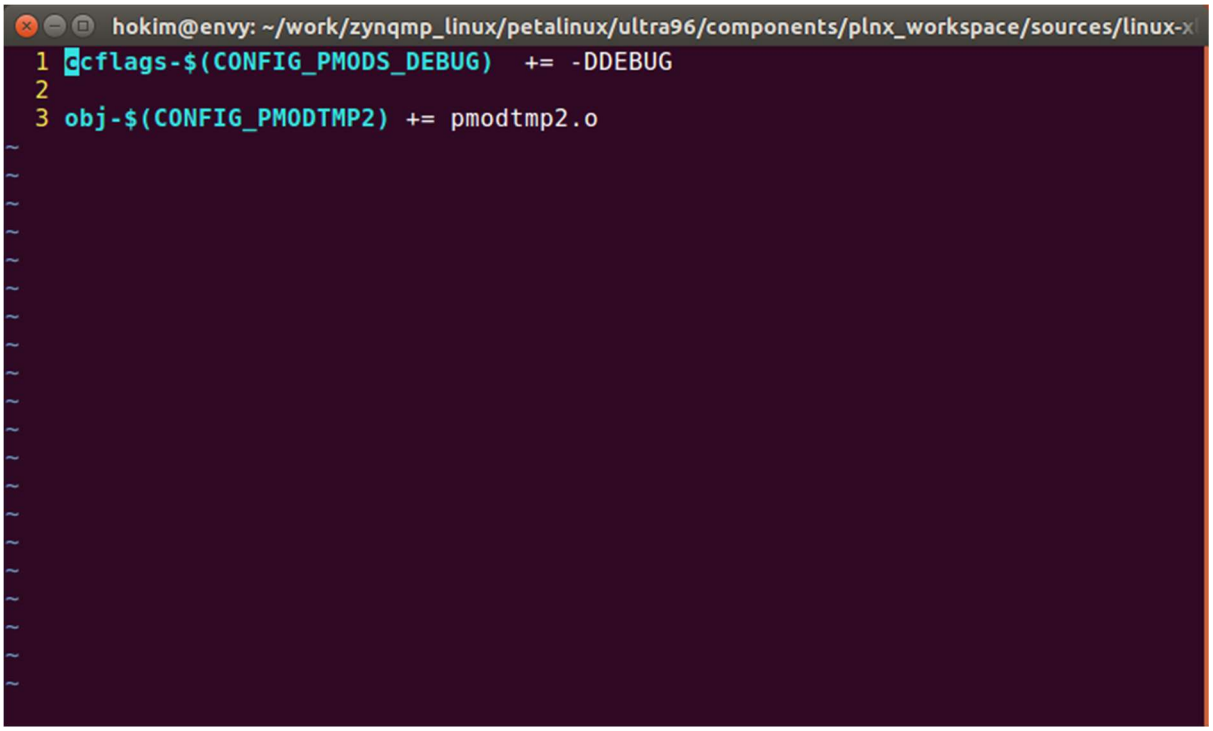

그림 10 drivers/staging/pmods/Makefile

```
$ vi drivers/staging/Kconfig
```

line145를 추가한다.

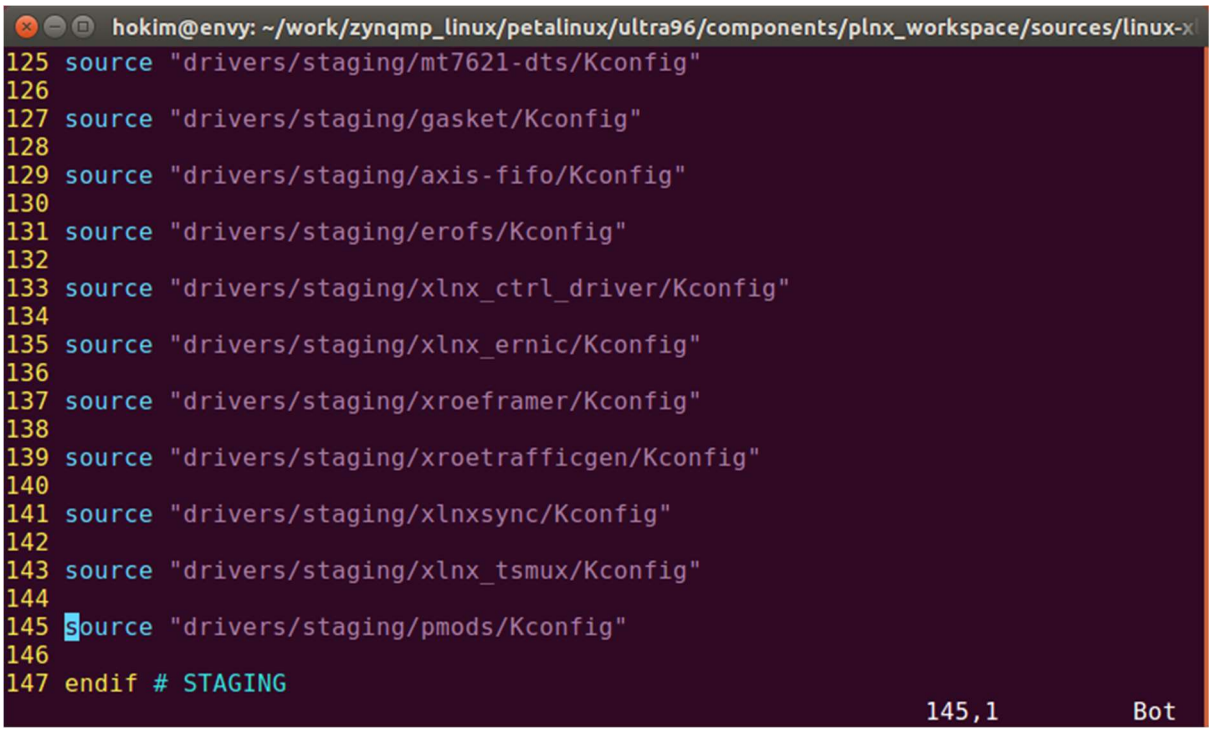

그림 11 drivers/staging/Kconfig

```
$ vi drivers/staging/Makefile
```

line63을 추가한다.

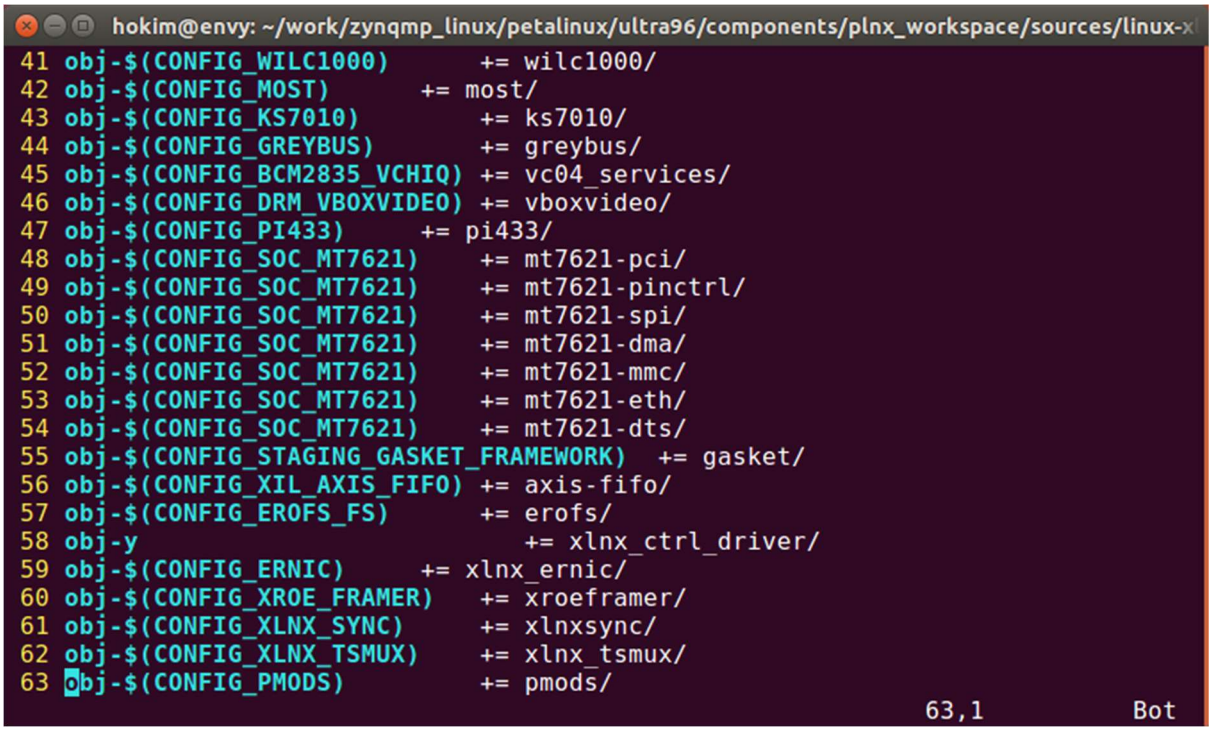

그림 12 drivers/staging/Makefile

```
$ cp ~/work/zynqmp_linux/petalinux/workspaces/pmodtmp2/pmodtmp2.c
drivers/staging/pmods/
```

PmodTmp2 Kernel Module을 Kernel Configuration에서 활성화하고 Build 한다.

```
$ cd ~/work/zynqmp_linux/petalinux/ultra96
$ petalinux-config -c kernel
```

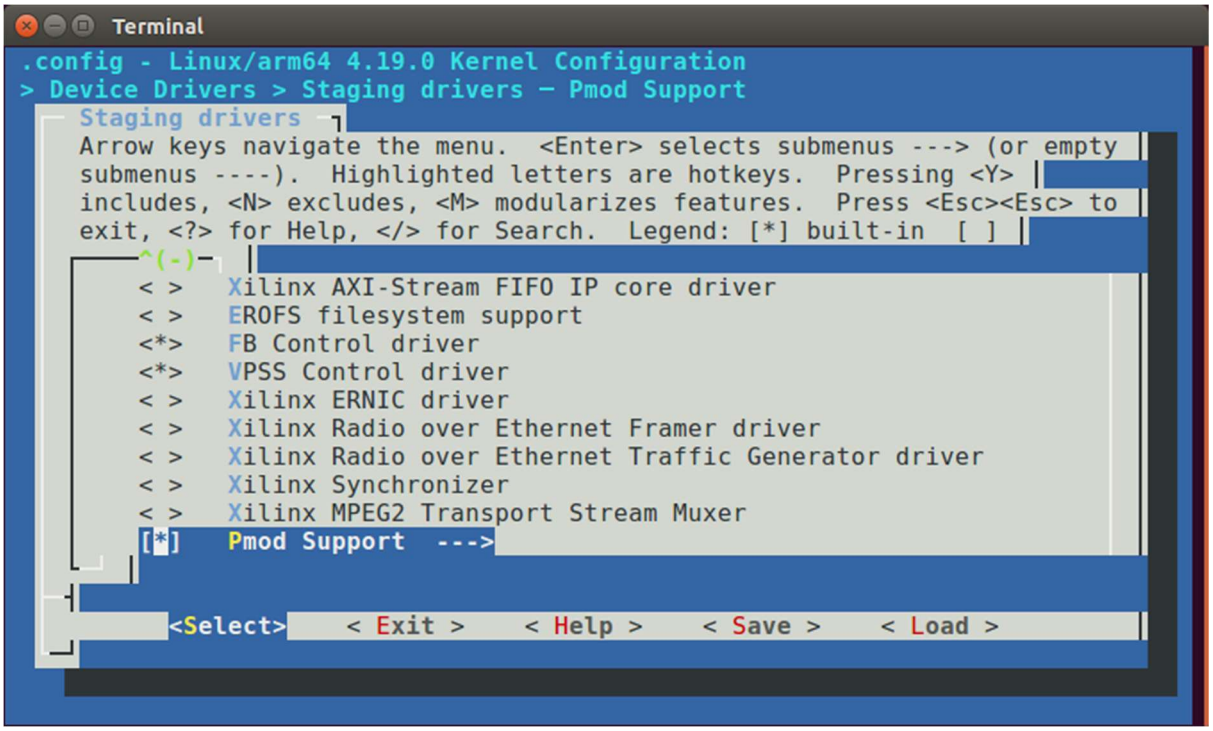

그림 13 Kernel Configuration(Pmod Support)

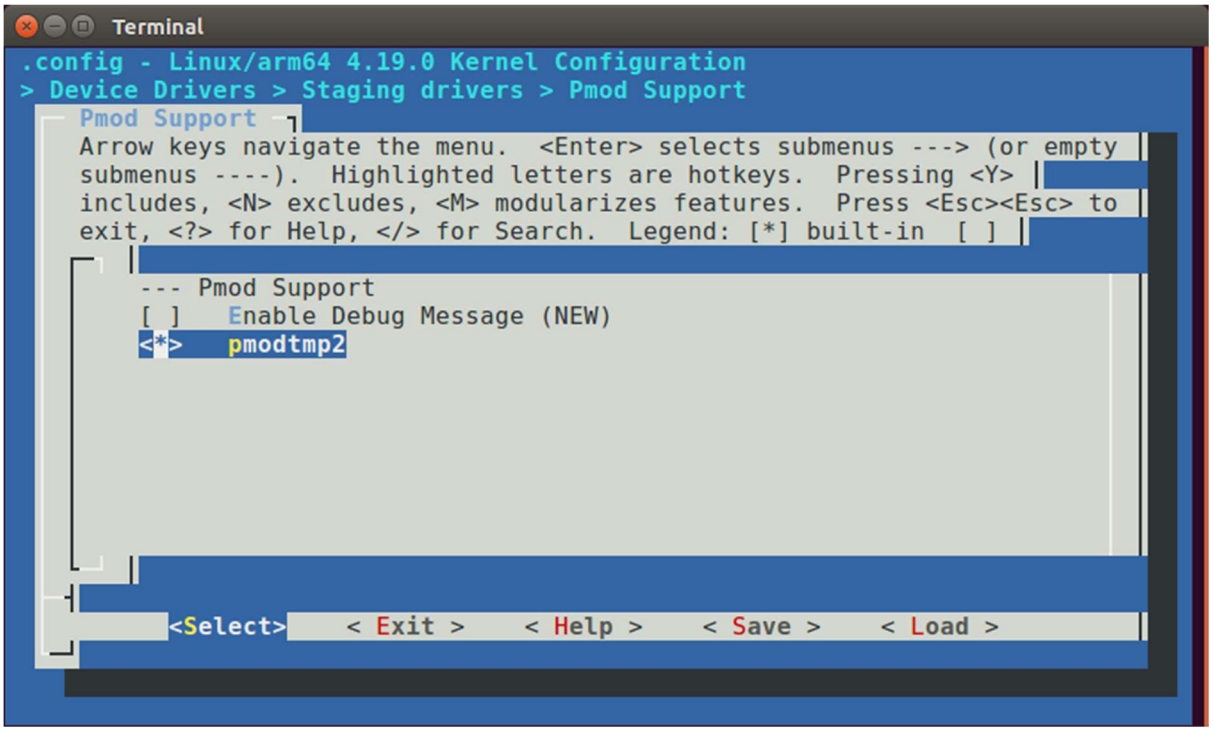

그림 14 Kernel Configuration(pmodtmp2)

```
$ petalinux-build -c kernel
```

Update된 Kernel Image를 가지고 있는 image.ub를 보드로 scp한다.

```
$ scp images/linux/image.ub root@172.30.1.39:/media/card
```

보드를 다시 boot한다.

```
ultra96$ reboot
```

Test를 수행한다.

```
ultra96$ cd /sys/class/i2c-adapter/i2c-0/0-004b
ultra96$ cat adc
3144
```

Kernel Source의 수정된 코드에 대한 patch 파일을 만든다. git configuration이 되어있지 않으면 git user.email과 user.name을 설정한다.

```
$ cd
~/work/zynqmp_linux/petalinux/ultra96/components/plnx_workspace/sources/linux
-xlnx
$ git config --global user.email "hokim@inipro.net"
$ git config --global user.name "Hyunok Kim"
$ git add .
$ git commit -s -m "Add pmodtmp2"
$ git format-patch -1
```

수정된 작업에 대해 recipe를 Update하고 cleanup한다.

```
$ cd ~/work/zynqmp_linux/petalinux/ultra96
$ petalinux-build -c kernel -x update-recipe
$ petalinux-config
```

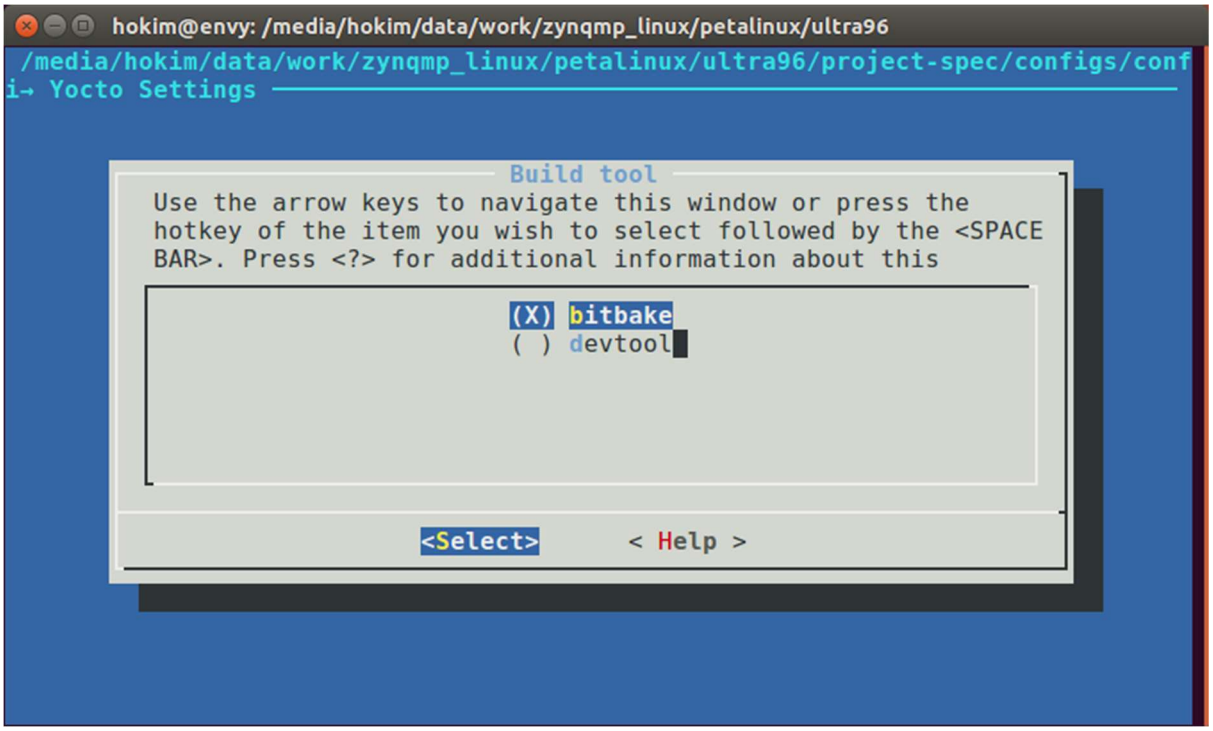

그림 15 Petalinux Configuration(bitbake)

```
$ petalinux-build -c kernel -x reset
```