

TI DSP, MCU 및 Xilinx Zynq FPGA 프로그래밍 전문가 과정

강사 : Innova Lee(이상훈)

gcccompil3r@gmail.com

학생 : 황수정

sue100012@naver.com

64 일차 (2018. 05. 30)

FPGA 환경설정

<https://www.xilinx.com/support/download.html> 로 접속해서 Embedded Development 에서 예전 버전인

[PetaLinux 2015.4 Installer](#) (TAR/GZIP - 1.68GB)

MD5 SUM Value : 74e752d382aec28f464eb3ce0d9cbc15

를 다운 받는다. 매년 분기마다 나오는데, 2015.4 가 안정화가 잘 되어있는 것 같아 이걸로 다운 받는다. 또 매번 인터페이스가 바뀌기에 배우기 용이한 버전으로 다운 받은 것 같다.

2015.4

PetaLinux - 2015.4 Installation Files - 2015.4 Installation Files

Important Information

- [README](#)

PetaLinux 2015.4 License and copyrights info (TAR/GZIP - 3.54 MB)

MD5 SUM Value : 7e9772ce396997d2898448ef52f25597

PetaLinux 2015.4 Source code (TAR/GZIP - 778.1 MB)

MD5 SUM Value : a692ee5ce54ced935b25194edccc1036

PetaLinux 2015.4 Installer (TAR/GZIP - 1.68GB)

MD5 SUM Value : 74e752d382aec28f464eb3ce0d9cbc15

Download Type

Last Updated

Answers

Documentation

Installation Files

Dec 15, 2015

[Release Notes and Known Issues](#)

[PetaLinux Tools Documentation](#)

다운 받은 후에, 설치를 위해서

```
sudo apt-get update
```

```
sudo apt-get install tofrodos iproute tftpd-hpa gawk gcc git-core make net-tools libncurses5-dev zlib1g-dev libssl-dev flex bison lib32z1 lib32ncurses5 lib32stdc++6 libsdl1.2-dev
```

위 2 개의 명령어를 실행해준다. petalinux 툴을 생성하기 위해서이다.

```
chmod 755 petalinux-v2015.4-final-installer-dec.run
```

모드를 바꾸어준다. 권한 설정을 해 준 것이다. 그 후, 설치를 다음 명령어로 실행한다.

```
./petalinux-v2015.4-final-installer-dec.run ~/petalinux_zynq/
```

install 을 petalinux_zynq 파일에 해주는 것이다.

작업 중간 중간에 해주는 명령어이지만, 원활한 작업을 위해 다음의 미리 명령어를 실행해준다.

```

sudo dpkg --add-architecture i386
sudo apt-get update
sudo apt-get install libbz2-1.0:i386
sudo apt-get install tofrodos iproute tftpd-hpa gawk gcc git-core make net-tools libncurses5-dev zlib1g-
dev libssl-dev flex bison lib32z1 lib32ncurses5 libsdl1.2-dev
sudo apt-get install xinetd tftpd-hpa
sudo apt-get install qemu-user-static qemu-system
sudo apt-get install linaro-image-tools
sudo apt-get install gcc-arm-linux-gnueabi

```

그 후 ~/petalinux_zynq/petalinux-v2015.4-final/components/linux-kernel/xlnx-4.0/drivers/uio 위치에서 uio_pdrv_genirq.c 소스코드를 수정한다.

```

#ifdef CONFIG_OF
static struct of_device_id uio_of_genirq_match[] = {
{ .compatible = "generic-uio", },
{ /* This is filled with module_parm */ },
{ /* Sentinel */ },
};

```

위의 체크 된 부분을 추가해주면 된다.

https://www.xilinx.com/support/documentation/university/vivado/workshops/vivado-embedded-linux-zynq/materials/2015x/LiveUSB_2015.4.zip

에서 다운로드한다. 압축을 풀어주고

```
cd petalinux_zynq/petalinux-v2015.4-final
```

로 위치를 이동하고

```
cp ~/Downloads/LiveUSB_2015.4/ZYBO_petalinux_v2015_4.bsp ./
```

복사해준다.

```
chmod -R 755 petalinux-v2015.4-final
```

권한 설정을 해준다. home 에서 bashrc(vi ~/.bashrc) 파일 로 들어가서 소스를 추가해주어야 한다.

```
source /home/계정 이름/petalinux_zynq(=본인이 생성한 파일명)/petalinux-v2015.4-final/settings.sh
```

추가한 후에 source ~/.bashrc 로 실행하면 환경설정을 해준다. 환경설정이 제대로 되었는지 test 를 한 번 해본다.

mkdir fpga_test → cd fpga_test/ → petalinux-create -t project -n test --template zynq

순으로 설정하면 프로젝트가 생성되었고 위치가 어디인지에 대한 정보가 나온다. 해당 위치에서 ls 로 확인해 볼 수 있다. 테스트가 완료 되면, rm -rf test 로 test 디렉토리를 지워준다. 그리고 이 폴더(fpga_test)에 ZYBO_petalinux_v2015_4.bsp 을 가져온다. 그 후 프로젝트를 생성해준다.

```
cp ~/Downloads/LiveUSB_2015.4/ZYBO_petalinux_v2015_4.bsp ./
petalinux-create -t project -s ZYBO_petalinux_v2015_4.bsp
cd ZYBO_petalinux_v2015_4/
petalinux-build
```

앞에서 미리 실행시킨 명령어는 여기서 실행한다. 그전에 위의 작업 중에 에러가 발생하면 sudo dpkg-reconfigure dash 명령어를 실행해서 no 를 선택한다. 그 후 아래 명령어들을 실행한다.

```
sudo dpkg --add-architecture i386
sudo apt-get update
sudo apt-get install libbz2-1.0:i386
sudo apt-get install tofrodos iproute tftpd-hpa gawk gcc git-core make net-tools libncurses5-dev zlib1g-dev libssl-dev flex bison lib32z1 lib32ncurses5 libsdl1.2-dev
sudo apt-get install xinetd tftpd-hpa
```

sudo apt-get install qemu-user-static qemu-system

sudo apt-get install linaro-image-tools

sudo apt-get install gcc-arm-linux-gnueabi

그 후에 **petalinux-build** 를 해준다. 이 명령어를 실행하면 fail 2 개가 뜨는데 이는 우리가 필요한 환경 설정에 성공했다는 뜻이므로 아래 명령어를 실행하면 된다.

petalinux-boot --qemu --kernel

이 명령어는 우리가 환경설정하는 작업의 마지막이며 kernel 을 부팅하는 것이므로 로그인하라는 것이 뜬다. 이 때는 아이디와 비밀번호 둘 다 root 를 치면 된다. 그리고 이 창에서 벗어나기 위해서는 ax 를 입력하면 된다.