

Xilinx FPGA, TI DSP·MCU

기반의 회로 설계 및 임베디드 전문가 과정

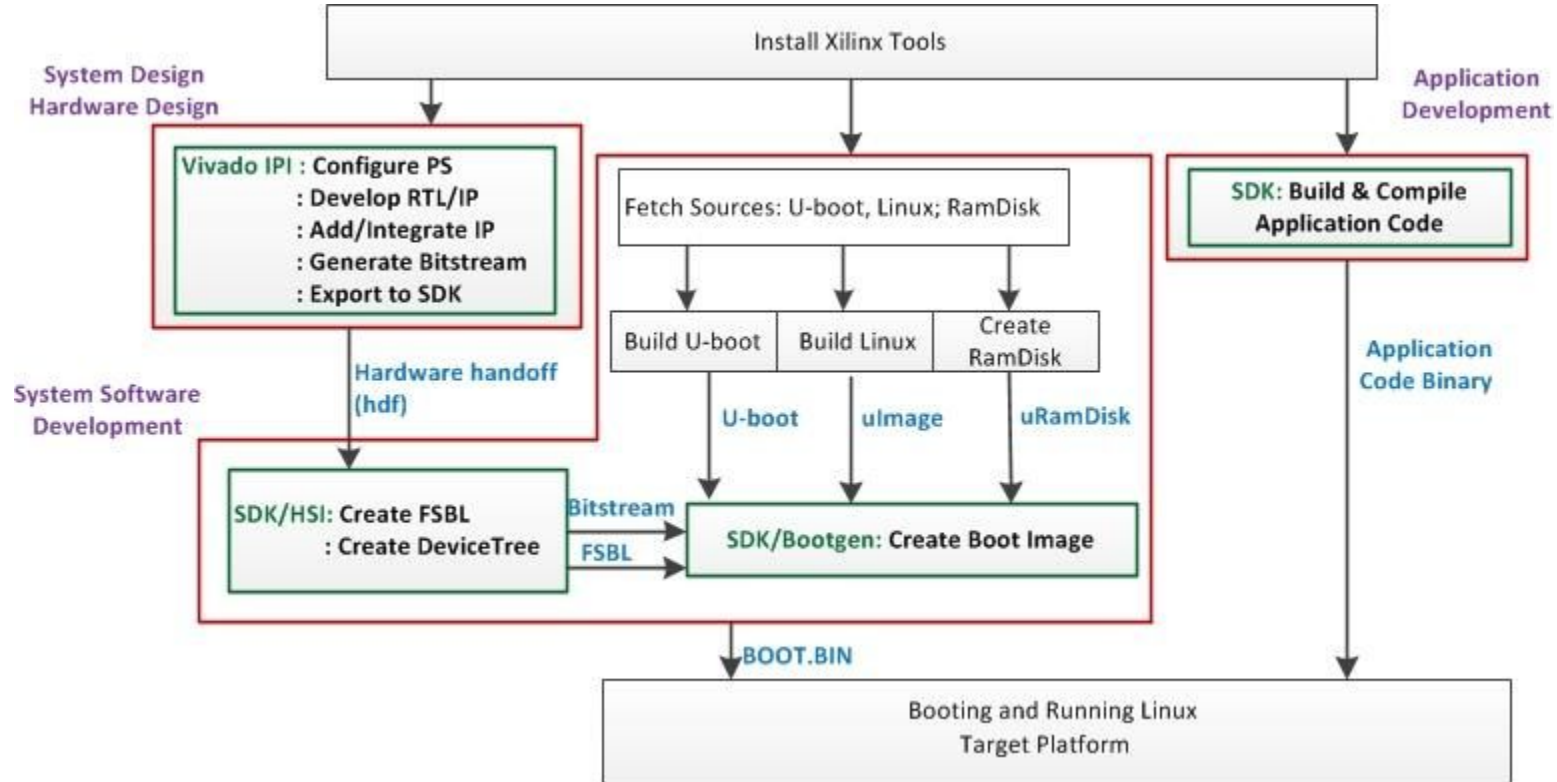
최준호
계획/성과
7주차

목차

- 내 역할
- 전체 일정
- FPGA 일정
 - FPGA 다음 주 목표
- 지난 주 성과

내 역할

- FPGA Linux Porting 및 Linux Device Driver 개발 및 장치 개발 및 PL 구현



하

2017.01

2015.04



```
peluza@peluza-B85H3-M7: ~
After this operation, 16.2 MB disk space will be freed.
Do you want to continue? [Y/n] Y
Preconfiguring packages ...
(Reading database ... 295959 files and directories currently installed.)
Preparing to unpack .../libpcre3_2%3a8.38-3.1_i386.deb ...
Unpacking libpcre3:i386 (2:8.38-3.1) over (1:8.31-2ubuntu2) ...
Processing triggers for libc-bin (2.23-0ubuntu7) ...
Processing triggers for man-db (2.7.5-1) ...
Setting up libpcre3:amd64 (2:8.38-3.1) ...
Setting up libpcre3:i386 (2:8.38-3.1) ...
Processing triggers for libc-bin (2.23-0ubuntu7) ...
(Reading database ... 295959 files and directories currently installed.)
Preparing to unpack .../libncursesw5-dbg_6.0+20160213-1ubuntu1_amd64.deb ...
Unpacking libncursesw5-dbg (6.0+20160213-1ubuntu1) over (5.9+20140118-1ubuntu1) ...
Preparing to unpack .../libncurses5-dbg_6.0+20160213-1ubuntu1_amd64.deb ...
Unpacking libncurses5-dbg (6.0+20160213-1ubuntu1) over (5.9+20140118-1ubuntu1) ...
Preparing to unpack .../libtinfo5-dbg_6.0+20160213-1ubuntu1_amd64.deb ...
Unpacking libtinfo5-dbg (6.0+20160213-1ubuntu1) over (5.9+20140118-1ubuntu1) ...
Preparing to unpack .../libncurses5-dev_6.0+20160213-1ubuntu1_amd64.deb ...
Unpacking libncurses5-dev:amd64 (6.0+20160213-1ubuntu1) over (5.9+20140118-1ubuntu1) ...
Preparing to unpack .../libncursesw5-dev_6.0+20160213-1ubuntu1_amd64.deb ...
Unpacking libncursesw5-dev:amd64 (6.0+20160213-1ubuntu1) over (5.9+20140118-1ubuntu1) ...
Preparing to unpack .../libtinfo-dev_6.0+20160213-1ubuntu1_amd64.deb ...
Unpacking libtinfo-dev:amd64 (6.0+20160213-1ubuntu1) over (5.9+20140118-1ubuntu1) ...
Preparing to unpack .../libncursesw5_6.0+20160213-1ubuntu1_amd64.deb ...
Unpacking libncursesw5:amd64 (6.0+20160213-1ubuntu1) over (5.9+20140118-1ubuntu1) ...
Processing triggers for man-db (2.7.5-1) ...
Processing triggers for libc-bin (2.23-0ubuntu7) ...
Setting up libncursesw5:amd64 (6.0+20160213-1ubuntu1) ...
Processing triggers for libc-bin (2.23-0ubuntu7) ...
(Reading database ... 295865 files and directories currently installed.)
Preparing to unpack .../libselinux1_2.4-3build2_i386.deb ...
Unpacking libselinux1:i386 (2.4-3build2) over (2.2.2-1ubuntu0.1) ...
dpkg: error processing archive /var/cache/apt/archives/libselinux1_2.4-3build2_i386.deb (--unpack):
 trying to overwrite shared '/usr/share/doc/libselinux1/changelog.Debian.gz', which is different from other instances of package libselinux1:i386
Processing triggers for libc-bin (2.23-0ubuntu7) ...
Errors were encountered while processing:
 /var/cache/apt/archives/libselinux1_2.4-3build2_i386.deb
E: Sub-process /usr/bin/dpkg returned an error code (1)
peluza@peluza-B85H3-M7:~$ sudo mv /usr/share/doc/libselinux1 /usr/share/doc/libselinux1.backup
peluza@peluza-B85H3-M7:~$ sudo apt-get -f install
```



```
root@peluza-B85H3-M7: /opt/pkg
peluza@peluza-B85H3-M7:~$ sudo su -
root@peluza-B85H3-M7:~# cd /opt/
google/      pkg/      sublime_text/ Xilinx/
root@peluza-B85H3-M7:~# cd /opt/
google/      pkg/      sublime_text/ Xilinx/
root@peluza-B85H3-M7:~# cd /opt/pkg/
root@peluza-B85H3-M7:/opt/pkg# ls
components  etc  packages  petalinux  petalinux_installation_log  petalinux-v2017.1-final-installer.run  settings.csh  settings.sh  tools
root@peluza-B85H3-M7:/opt/pkg# ./petalinux-v2017.1-final-installer.run /opt/pkg
INFO: Checking installer checksum...
INFO: Extracting PetaLinux installer...
INFO: Installing PetaLinux...
INFO: Checking PetaLinux installer integrity...
*****
WARNING: PetaLinux installation directory: /opt/pkg/. is not empty!
*****
Please input "y" to continue to install PetaLinux in that directory?[n]y
INFO: Extracting Installation files...

LICENSE AGREEMENTS

PetaLinux SDK contains software from a number of sources. Please review
the following licenses and indicate your acceptance of each to continue.

You do not have to accept the licenses, however if you do not then you may
not use PetaLinux SDK.

Use PgUp/PgDn to navigate the license viewer, and press 'q' to close

Press Enter to display the license agreements
Do you accept Xilinx End User License Agreement? [y/N] > y
Do you accept Webtalk Terms and Conditions? [y/N] > y
Do you accept Third Party End User License Agreement? [y/N] > y
INFO: Checking installation environment requirements...
INFO: Checking free disk space
INFO: Checking installed tools
INFO: Checking installed development libraries
INFO: Checking network and other services
WARNING: No tftp server found - please refer to "PetaLinux SDK Installation Guide" for its impact and solution
INFO: Installing PetaLinux SDK to "/opt/pkg/."
```



```
peluza@peluza-B85H3-M7:~/petalinux/led_test2$ petalinux-  
petalinux-boot      petalinux-config  petalinux-package  
petalinux-build     petalinux-create  petalinux-util
```

```
peluza@peluza-B85H3-M7:~/petalinux/led_test2$ petalinux-boot --help  
petalinux-boot (c) 2005-2013 Xilinx, Inc.  
  
This command boots the MicroBlaze/Zynq systems with Petalinux images  
through JTAG/QEMU.  
Usage:  
  petalinux-boot --qemu|--jtag -c|--component <COMPONENT> [options]  
Required:  
  --jtag|--qemu          JTAG/QEMU boot mode  
  
Options:  
  --prebuilt <BOOT_LEVEL>    Boot prebuilt images (override all settings).  
                             supported boot level 1 to 3  
                             1 - download FPGA bitstream (and FSBL for Zynq)  
                             2 - Boot U-Boot only  
                             3 - Boot Linux Kernel only  
  --boot-addr <BOOT_ADDR>    boot address  
  -i, --image <IMAGE>        image to boot  
  --u-boot                   boot images/linux/u-boot.elf image  
                             if --kernel is specified, --u-boot will not take  
                             effect.  
  --kernel                   boot images/linux/zImage for Zynq  
                             boot images/linux/image.elf for MicroBlaze  
                             if --kernel is specified, --u-boot will not take  
                             effect.  
  -v, --verbose               output debug messages  
  -h|--help                   Display help messages  
Please specify a boot mode for the detailed options:  
Show jtag boot options:  
  $ petalinux-boot --jtag --help  
  
Show qemu boot options:  
  $ petalinux-boot --qemu --help
```




```
peluza@peluza-B85H3-M7:~/petalinux/led_test2$ petalinux-config --help
petalinux-config (c) 2005-2013 Xilinx, Inc.
```

```
INFO: Checking component...
Configures the project or the specified component with menuconfig.
```

```
Usage:
  petalinux-config [options] [--component <COMPONENT>] [--get-hw-description[=SRC]] [--searchpath <--ACTION>]
  > [VALUE]]
```

```
Options:
  -h, --help                show function usage
  -p, --project <PROJECT>  path to Petalinux SDK project.
                           default is the working project
  --oldconfig               takes the working configuration
  -c, --component <COMPONENT> Specify the component
                           If no component is specified, it will do
                           top level subsystem configuration only
                           all: to configure the whole project
                           If you specify other component, it will
                           configure that component
                           E.g. -c rootfs
  --get-hw-description [SRC] get hardware description.
                           if [SRC] is specified, look in that
                           location for an Vivado export to SDK directory.
                           Otherwise, this MUST be run from
                           WITHIN the vivado export to SDK directory.
  --defconfig [DEFCONFIG_TARGET] defconfig the specified component.
                           It only applies to kernel for now.
  --searchpath              edit project search path
  -v, --verbose             verbose mode
```

```
Available project user searchpath actions:
  --prepend <SEARCHPATH>  prepend <SEARCHPATH> to project external searchpath
  --append <SEARCHPATH>   append <SEARCHPATH> to project external searchpath
  --replace <SEARCHPATH>  replace project user searchpath with <SEARCHPATH>
  --print                  print full project searchpath
  --delete                delete project external searchpath
```

```
Available Components of linux for this command:
  * kernel      # is of linux-kernel type
  * rootfs      # is of rootfs type
  * u-boot      # is of u-boot type
```

EXAMPLES:

```
Edit searchpath:
  Default Petalinux tools will look into <PROJECT>/components/ first and then
  ${PETALINUX}/components/ for components
```

```
Prepend external searchpath:
  $ petalinux-config --searchpath --prepend <EXTERN_SEARCHPATH0>
```

```
Prepend external searchpath:
  $ petalinux-config --searchpath --prepend <EXTERN_SEARCHPATH0>
  the components searchpath will become:
  <PROJECT>/components:<EXTERN_SEARCHPATH0>:${PETALINUX}/components/
```

```
Append external searchpath:
  $ petalinux-config --searchpath --append <EXTERN_SEARCHPATH1>
  the components searchpath will become:
  <PROJECT>/components:<EXTERN_SEARCHPATH0>:<EXTERN_SEARCHPATH1>:${PETALINUX}/components/
```

```
Delete external searchpath:
  $ petalinux-config --searchpath --delete
  the components searchpath will become:
  <PROJECT>/components:${PETALINUX}/components/
```

Sync hardware description:

```
Sync hardware description from Vivado export to Petalinux BSP project:
  $ cd <Vivado_Export_to_SDK_Directory>
  $ petalinux-config --get-hw-description
  It will sync up the XML file and ps7_init.c/.h for Zynq from
  <Vivado_Export_to_SDK_Directory> to subsystems/linux/hw-description/ directory.
```

```
Sync hardware description inside Petalinux project but outside Vivado export to SDK directory:
  $ petalinux-config --get-hw-description=<Vivado_Export_to_SDK_Directory>
```

Configure Petalinux project:

```
Configure subsystem level configuration:
  $ petalinux-config
```

```
Configure kernel:
  $ petalinux-config -c kernel
```

```
Configure rootfs:
  $ petalinux-config -c rootfs
```

```
Defconfig kconfig kernel with xilinx_zynq_base_trd_defconfig:
  $ petalinux-config -c kernel --defconfig xilinx_zynq_base_trd_defconfig
```



```
peluza@peluza-B85H3-M7:~/petalinux/led_test$ petalinux-build --help
petalinux-build (c) 2005-2013 Xilinx, Inc.
```

```
INFO: Checking component...
Builds the project or the specified components.
```

```
Usage:
  petalinux-build [options]
```

Required:

```
Options:
  -h, --help                show function usage
  -p, --project <PROJECT>  path to Petalinux SDK project.
                           Default is working project.
  -c, --component <COMPONENT> Specify the component
                           all: to build the whole project
                           If you specify other component, it will
                           build that component
                           E.g. -c rootfs
                           E.g. -c rootfs/myapp
                           If you use -c with --help option, it will
                           show you subcomponents.
                           E.g. -c rootfs --help shows subcomponents
                           of rootfs.
  -x, --execute <GNU_MAKE_TARGET> Specify a GNU make command of the component
  --makeenv <MAKE_ENV>          Pass GNU make environment variables
  -v, --verbose               Show compile messages verbose mode
```

Available Components for linux:

* bootloader	# is of bootloader type
* device-tree	# is of device-tree type
* kernel	# is of linux-kernel type
* rootfs	# is of rootfs type
* u-boot	# is of u-boot type

Available make target for linux:

Quick reference for various supported build targets for linux.

clean	clean out build objects
distclean	clean out build
all	build subsystem and generate final image
build	build subsystem
install	install built objects to target subsystem host copy
package	combine target file system and kernel into final image

EXAMPLES:

```
Build the project:
$ petalinux-build
It is the same as "petalinux-build -c all"
the bootable images are in <PROJECT>/images/linux/.
```

```
Build kernel only:
$ petalinux-build -c kernel
```

```
Build kernel and update the bootable images:
```

Build kernel and update the bootable images:

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

Build rootfs only:

```
$ petalinux-build -c rootfs
```

Build myapp of rootfs only:

```
$ petalinux-build -c rootfs/myapp
```

Clean up u-boot and build again:

```
$ petalinux-build -c u-boot -x distclean
## above command will remove the <PROJECT>/build/linux/u-boot/ directory.
$ petalinux-build -c u-boot
```

Clean up the project build and build again:

```
$ petalinux-build -x distclean
## above command will remove the <PROJECT>/build/ directory.
$ petalinux-build
```

Clean up the project build and the generated bootable images:

```
$ petalinux-build -x mrproper
## above command will remove <PROJECT>/images/ and <PROJECT>/build/ directories
```



```
peluza@peluza-B85H3-M7:~/petalinux/led_test2$ petalinux-package --help
petalinux-package
(c) 2005-2013 Xilinx, Inc.

This command packages various image format, firmware, prebuilt
and bsp
Usage:
  petalinux-package --boot|--bsp|--firmware|--image|--prebuilt [options]

Required:
  --boot|--bsp|--firmware|--image|--prebuilt
    Various package mode.
    boot: packages a boot.bin for Zynq
    bsp: packages a bsp
    firmware: creates a firmware package used
    by Petalinux firmware upgrade demo app to
    upgrade firmwares.
    image: package various image type
    prebuilt: package images to prebuilt

Options:
  -h|--help          Display help messages
Please specify a package mode option for the detailed options
Show package boot options:
  $ petalinux-package --boot --help

Show package bsp options:
  $ petalinux-package --bsp --help

Show package firmware options:
  $ petalinux-package --firmware --help

Show package image options:
  $ petalinux-package --image --help

Show package prebuilt options:
  $ petalinux-package --prebuilt --help

ERROR: No package mode has been specified.
```

```
peluza@peluza-B85H3-M7:~/petalinux/led_test2/images/linux$ \
> petalinux-package --boot --fsbl zynq fsbl.elf \
> --fpga ./system_wrapper.bit --u-boot
```



```
peluza@peluza-B85H3-M7:~/petalinux/led_test2$ cd build/ && ls
bootgen.bif  build.log  build.log.old  linux  qemu_image.elf
```

```
1 [ALL] ] make: Entering directory '/home/peluza/petalinux/led_test2/build/linux/kernel'
2 [ALL] ] make: Nothing to be done for 'pre-build'.
3 [ALL] ] make: Leaving directory '/home/peluza/petalinux/led_test2/build/linux/kernel'
4 [ALL] ] make: Entering directory '/home/peluza/petalinux/led_test2/build/linux/kernel'
5 [ALL] ] mkdir -p "/home/peluza/petalinux/led_test2/build/linux/kernel/xlnx-4.0"
6 [ALL] ] mkdir -p /home/peluza/petalinux/led_test2/build/linux/kernel/xlnx-4.0/modules
7 [ALL] ] rm -rf link-to-kernel-build
8 [ALL] ] ln -s /home/peluza/petalinux/led_test2/build/linux/kernel/xlnx-4.0 link-to-kernel-build
9 [ALL] ] rsync /home/peluza/petalinux/led_test2/subsystems/linux/configs/kernel/config /home/peluza/petalinux/led_test2/build/linux/kernel/xlnx-4.0/.config
10 [ALL] ] rm -rf /home/peluza/petalinux/led_test2/build/linux/kernel/boot /home/peluza/petalinux/led_test2/build/linux/kernel/usr
11 [ALL] ] ln -s "/home/peluza/petalinux/led_test2/build/linux/kernel/xlnx-4.0/arch/arm/boot" "/home/peluza/petalinux/led_test2/build/linux/kernel/boot"
12 [ALL] ] ln -s "/home/peluza/petalinux/led_test2/build/linux/kernel/xlnx-4.0/usr" "/home/peluza/petalinux/led_test2/build/linux/kernel/usr"
13 [INFO] ] build linux/kernel
14 [ALL] ] make ARCH=arm CROSS_COMPILE=arm-xilinx-linux-gnueabi-0=/home/peluza/petalinux/led_test2/build/linux/kernel/xlnx-4.0 KCONFIG_CONFIG="/home/peluza/petalinux/
led_test2/build/linux/kernel/xlnx-4.0/.config" j3 -C /opt/peta/petalinux-v2015.4-final/components/linux-kernel/xlnx-4.0 all
15 [ALL] ] make[1]: Entering directory '/opt/peta/petalinux-v2015.4-final/components/linux-kernel/xlnx-4.0'
16 [ALL] ] make[2]: Entering directory '/home/peluza/petalinux/led_test2/build/linux/kernel/xlnx-4.0'
17 [ALL] ] GEN      ./Makefile
18 [ALL] ] HOSTCC    scripts/basic/fixdep
19 [ALL] ] HOSTCC    scripts/kconfig/conf.o
20 [ALL] ] SHIPPED    scripts/kconfig/zconf.tab.c
21 [ALL] ] SHIPPED    scripts/kconfig/zconf.lex.c
22 [ALL] ] SHIPPED    scripts/kconfig/zconf.hash.c
23 [ALL] ] HOSTCC    scripts/kconfig/zconf.tab.o
24 [ALL] ] HOSTLD    scripts/kconfig/conf
25 [ALL] ] scripts/kconfig/conf --silentoldconfig Kconfig
26 [ALL] ] CHK      include/config/kernel.release
27 [ALL] ] GEN      ./Makefile
28 [ALL] ] CHK      include/generated/uapi/linux/version.h
29 [ALL] ] UPO      include/generated/uapi/linux/version.h
30 [ALL] ] HOSTCC    scripts/basic/bin2c
31 [ALL] ] WRAP      arch/arm/include/generated/asm/auxvec.h
32 [ALL] ] WRAP      arch/arm/include/generated/asm/bitsperlong.h
33 [ALL] ] WRAP      arch/arm/include/generated/asm/cputime.h
34 [ALL] ] WRAP      arch/arm/include/generated/asm/current.h
35 [ALL] ] WRAP      arch/arm/include/generated/asm/emergency-restart.h
36 [ALL] ] WRAP      arch/arm/include/generated/asm/errno.h
37 [ALL] ] WRAP      arch/arm/include/generated/asm/exec.h
38 [ALL] ] WRAP      arch/arm/include/generated/asm/atomic.h
39 [ALL] ] WRAP      arch/arm/include/generated/asm/ipcbuf.h
40 [ALL] ] WRAP      arch/arm/include/generated/asm/lrq_regs.h
41 [ALL] ] WRAP      arch/arm/include/generated/asm/kdebug.h
42 [ALL] ] WRAP      arch/arm/include/generated/asm/local.h
43 [ALL] ] WRAP      arch/arm/include/generated/asm/local64.h
44 [ALL] ] WRAP      arch/arm/include/generated/asm/mcs_spinlock.h
45 [ALL] ] WRAP      arch/arm/include/generated/asm/msgbuf.h
46 [ALL] ] WRAP      arch/arm/include/generated/asm/param.h
47 [ALL] ] WRAP      arch/arm/include/generated/asm/parport.h
48 [ALL] ] WRAP      arch/arm/include/generated/asm/poll.h
49 [ALL] ] WRAP      arch/arm/include/generated/asm/preempt.h
50 [ALL] ] WRAP      arch/arm/include/generated/asm/resource.h
51 [ALL] ] WRAP      arch/arm/include/generated/asm/rsem.h
52 [ALL] ] WRAP      arch/arm/include/generated/asm/scatterlist.h
53 [ALL] ] WRAP      arch/arm/include/generated/asm/sections.h
54 [ALL] ] WRAP      arch/arm/include/generated/asm/segment.h
55 [AI] ] WRAP      arch/arm/include/generated/asm/sembuf.h
2246 [ALL] ] LD [M] net/ipv6/xfrm6_mode_transport.ko
2247 [ALL] ] LD [M] net/ipv6/xfrm6_mode_tunnel.ko
2248 [ALL] ] make[2]: Leaving directory '/home/peluza/petalinux/led_test2/build/linux/kernel/xlnx-4.0'
2249 [ALL] ] make[1]: Leaving directory '/opt/peta/petalinux-v2015.4-final/components/linux-kernel/xlnx-4.0'
2250 [ALL] ] make: Leaving directory '/home/peluza/petalinux/led_test2/build/linux/kernel'
2251 [ALL] ] make: Entering directory '/home/peluza/petalinux/led_test2/build/linux/kernel'
2252 [ALL] ] make: Nothing to be done for 'post-build'.
2253 [ALL] ] make: Leaving directory '/home/peluza/petalinux/led_test2/build/linux/kernel'
2254 [ALL] ] make: Entering directory '/home/peluza/petalinux/led_test2/build/linux/kernel'
2255 [ALL] ] make: Nothing to be done for 'pre-install'.
2256 [ALL] ] make: Leaving directory '/home/peluza/petalinux/led_test2/build/linux/kernel'
2257 [ALL] ] make: Entering directory '/home/peluza/petalinux/led_test2/build/linux/kernel'
2258 [INFO] ] install linux/kernel
2259 [ALL] ] if [ ! -f /home/peluza/petalinux/led_test2/build/linux/kernel/xlnx-4.0/vmlinux ]; then
vmlinux is not found. Please build linux first!"; exit 255; fi
2260 [ALL] ] cp "/home/peluza/petalinux/led_test2/build/linux/kernel/xlnx-4.0"/vmlinux "/home/peluza/petalinux/led_test2/build/linux/kernel/xlnx-4.0"/System.map
2261 [ALL] ] cp "/home/peluza/petalinux/led_test2/build/linux/kernel/xlnx-4.0"/System.map "/home/peluza/petalinux/led_test2/build/linux/kernel/xlnx-4.0"/System.map
2262 [ALL] ] make: Leaving directory '/home/peluza/petalinux/led_test2/build/linux/kernel'
2263
```

하

```
peluza@peluza-B85H3-M7:~/petalinux/led_test2$ cd build/ && ls
bootgen.bif  build.log  build.log.old  linux  qemu_image.elf
```

```
2027 [ALL ] /opt/peta/petalinux-v2015.4-final/components/linux-kernel/xlnx-4.0/drivers/uis/uis_pdrv_genirq.c:257:4: error: expected expression before ',' token
2028 [ALL ] { ,compatible = "generic-uis" },
2029 [ALL ] ^
2030 [ALL ] /opt/peta/petalinux-v2015.4-final/components/linux-kernel/xlnx-4.0/scripts/Makefile.build:258: recipe for target 'drivers/uis/uis_pdrv_genirq.o' failed
2031 [ERROR] make[5]: *** [drivers/uis/uis_pdrv_genirq.o] Error 1
2032 [ALL ] /opt/peta/petalinux-v2015.4-final/components/linux-kernel/xlnx-4.0/scripts/Makefile.build:403: recipe for target 'drivers/uis' failed
2033 [ERROR] make[4]: *** [drivers/uis] Error 2
2034 [ALL ] make[4]: *** Waiting for unfinished jobs....
```

```
256 // { /* This is filled with module_parm */ },
257 { .compatible = "generic-uis" },
258 { /* Sentinel */ },
```



```
peluza@peluza-B85H3-M7:~/petalinux/led_test2/build$ petalinux-build
INFO: Checking component...
INFO: Generating make files and build linux
INFO: Generating make files for the subcomponents of linux
INFO: Building linux
[INFO ] pre-build linux/rootfs/fwupgrade
[INFO ] pre-build linux/rootfs/peekpoke
[INFO ] build linux/kernel
[INFO ] build zynq_fsbl
[INFO ] Setting up stage config
[INFO ] Setting up rootfs config
[INFO ] Updating for cortex9-vfp-neon
[INFO ] Updating package manager
[INFO ] Expanding stagefs
[INFO ] build linux/rootfs/fwupgrade
[INFO ] build linux/rootfs/peekpoke
[INFO ] build kernel in-tree modules
[INFO ] modules linux/kernel
[INFO ] post-build linux/rootfs/fwupgrade
[INFO ] post-build linux/rootfs/peekpoke
[INFO ] pre-install linux/rootfs/fwupgrade
[INFO ] pre-install linux/rootfs/peekpoke
[INFO ] install system.dtb
[INFO ] install linux/kernel
[INFO ] Expanding rootfs
[INFO ] install sys_init
[INFO ] install linux/rootfs/fwupgrade
[INFO ] install linux/rootfs/peekpoke
[INFO ] install kernel in-tree modules
[INFO ] modules install linux/kernel
[INFO ] post-install linux/rootfs/fwupgrade
[INFO ] post-install linux/rootfs/peekpoke
[INFO ] package rootfs.cpio to /home/peluza/petalinux/led_test2/images/linux
[INFO ] Update and install vmlinux image
[INFO ] vmlinux linux/kernel
[INFO ] install linux/kernel
[INFO ] package zImage
[INFO ] zImage linux/kernel
[INFO ] install linux/kernel
[INFO ] Package HDF bitstream
```

```
peluza@peluza-B85H3-M7:~/petalinux/led_test2$ petalinux-build
INFO: Checking component...
INFO: Generating make files and build linux
INFO: Generating make files for the subcomponents of linux
INFO: Building linux
[INFO ] pre-build linux/rootfs/led_test_module
[INFO ] pre-build linux/rootfs/fwupgrade
[INFO ] pre-build linux/rootfs/led_test_app
[INFO ] pre-build linux/rootfs/peekpoke
[INFO ] build system.dtb
[INFO ] build linux/kernel
[ERROR] make[3]: *** [include/generated/utsrelease.h] Error 1
[ERROR] make[3]: *** wait: No child processes. Stop.
[ERROR] make[2]: *** [sub-make] Error 2
[ERROR] make[1]: *** [/home/peluza/petalinux/led_test2/build/linux/kernel/linux-4.4.32/vmlinux] Error 2
ERROR: Failed to build linux
```



```
U-Boot-Petalinux> printenv
autoload=no
baudrate=115200
boot_img=BOOT.BIN
bootcmd=run default_bootcmd
bootdelay=4
bootenvsize=0x20000
bootenvstart=0x500000
clobstart=0x01000000
console=console=ttyPS0,115200
cp_kernel2ram=mmcinfo && fatload mmc 0 ${netstart} ${kernel_img}
default_bootcmd=run cp_kernel2ram && bootm ${netstart}
dtb_img=system.dtb
dtbnetstart=0x02800000
eraseenv=sf probe 0 && sf erase ${bootenvstart} ${bootenvsize}
ethact=Gem.e000b000
ethaddr=00:0a:35:00:1e:53
fault=echo ${img} image size is greater than allocated place - partition ${img} is NOT UPDATED
hostname=led_test2
install_boot=mmcinfo && fatwrite mmc 0 ${clobstart} ${boot_img} ${filesize}
install_jffs2=sf probe 0 && sf erase ${jffs2start} ${jffs2size} && sf write ${clobstart} ${jffs2start} ${jffs2_img}
install_kernel=mmcinfo && fatwrite mmc 0 ${clobstart} ${kernel_img} ${filesize}
jffs2_img=rootfs.jffs2
kernel_img=image.ub
load_boot=tftpboot ${clobstart} ${boot_img}
load_dtb=tftpboot ${clobstart} ${dtb_img}
load_jffs2=tftpboot ${clobstart} ${jffs2_img}
load_kernel=tftpboot ${clobstart} ${kernel_img}
loadaddr=0x01000000
nc=setenv stdout nc;setenv stdin nc;
netboot=tftpboot ${netstart} ${kernel_img} && bootm
netstart=0x01000000
psserial0=setenv stdout ttyPS0;setenv stdin ttyPS0
sd_update_dtb=echo Updating dtb from SD; mmcinfo && fatload mmc 0:1 ${clobstart} ${dtb_img} && run instab
sd_update_jffs2=echo Updating jffs2 from SD; mmcinfo && fatload mmc 0:1 ${clobstart} ${jffs2_img} && run2
sdboot=echo boot Petalinux; mmcinfo && fatload mmc 0 ${netstart} ${kernel_img} && bootm
serial=setenv stdout serial;setenv stdin serial
serverip=192.168.219.103
test_crc=if !m ${clobstart}; then run test_img; else echo ${img} Bad CRC - ${img} is NOT UPDATED; fi
test_img=setenv var "if test ${filesize} -gt ${psize}; then run fault; else run ${installcmd}; fi"; run r
update_boot=setenv img boot; setenv psize ${bootsize}; setenv installcmd "install_boot"; run load_boot $d
update_dtb=setenv img dtb; setenv psize ${dtbsize}; setenv installcmd "install_dtb"; run load_dtb_test_id
update_jffs2=setenv img jffs2; setenv psize ${jffs2size}; setenv installcmd "install_jffs2"; run load_jfd
update_kernel=setenv img kernel; setenv psize ${kernelsize}; setenv installcmd "install_kernel"; run load

Environment size: 2587/131068 bytes
U-Boot-Petalinux>
```


하

```
U-Boot-PetaLinux> dhcp
Gem.e000b000:1 is connected to Gem.e000b000. Reconnecting to Gem.e000b000
Gem.e000b000 Waiting for PHY auto negotiation to complete.... done
BOOTP broadcast 1
*** Unhandled DHCP Option in OFFER/ACK: 26
*** Unhandled DHCP Option in OFFER/ACK: 26
DHCP client bound to address 192.168.123.100 (34 ms)
Using Gem.e000b000 device
TFTP from server 192.168.0.120; our IP address is 192.168.123.100; sending through gateway 192.168.123.
Filename 'image.ub'.
Load address: 0x1000000
Loading: *
TFTP error: 'File not found' (1)
Not retrying...
```

```
peluza@peluza-B85H3-M7:~/petalinux/led_test2$ ls components/
apps  bootloader  modules
```


하

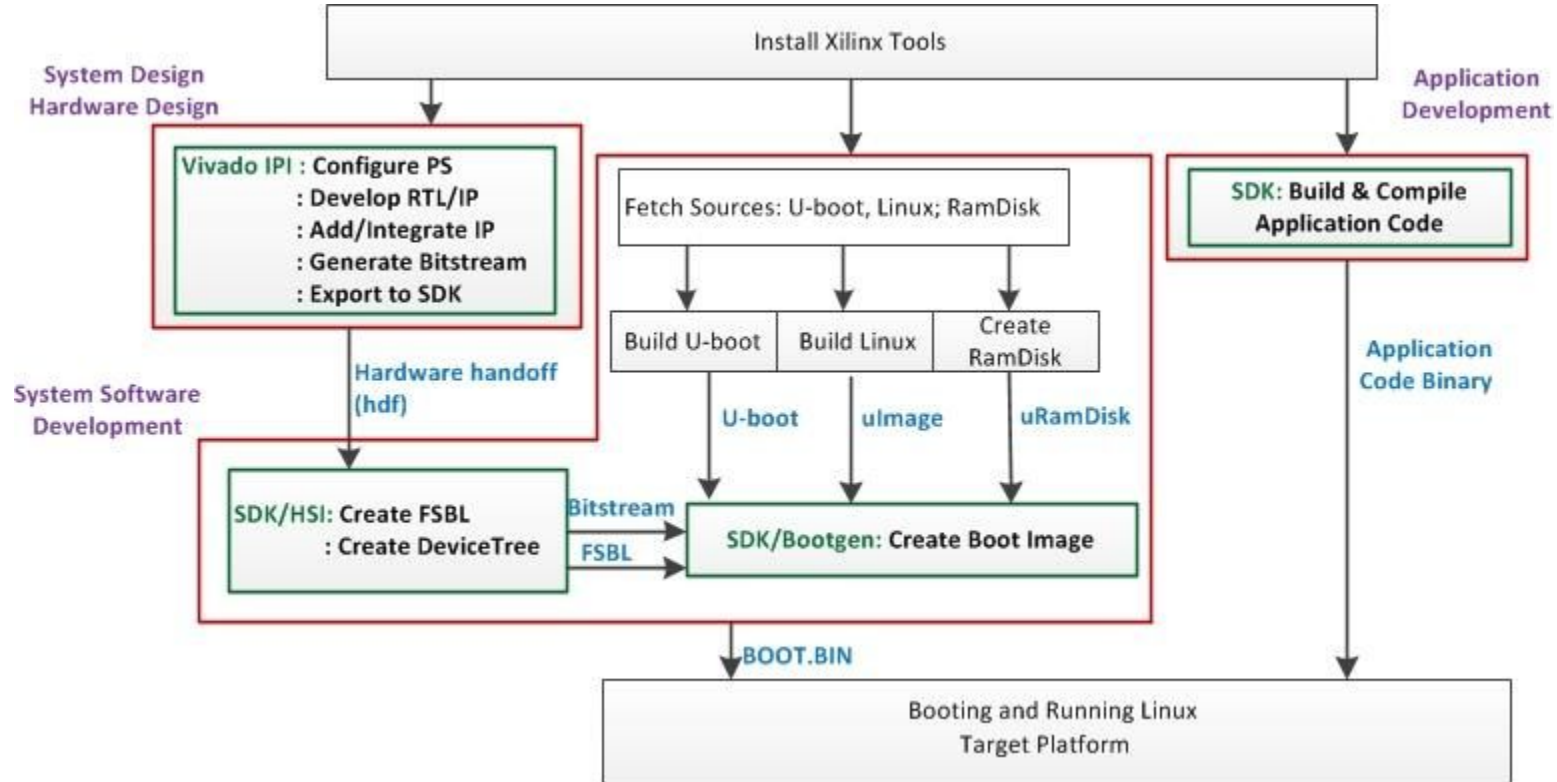
```
1 /* led_test_module.c - The simplest kernel module */
2 */
3 #include <linux/kernel.h>
4 #include <linux/init.h>
5 #include <linux/module.h>
6 #include <linux/slab.h>
7 #include <linux/io.h>
8 #include <linux/interrupt.h>
9
10 #include <linux/of_address.h>
11 #include <linux/of_device.h>
12 #include <linux/of_platform.h>
13
14 /* Standard module information, edit as appropriate */
15 MODULE_LICENSE("GPL");
16 MODULE_AUTHOR
17     ("Xilinx Inc.");
18 MODULE_DESCRIPTION
19     ("led_test_module - loadable module template generated by petalinux-create -t modules");
20
21 #define DRIVER_NAME "led_test_module"
22
23 /* Simple example of how to receive command line parameters to your module.
24  * Delete if you don't need them */
25 unsigned myint = 0xdeadbeef;
26 char *mystr = "default";
27
28 module_param(myint, int, S_IRUGO);
29 module_param(mystr, charp, S_IRUGO);
30
31 struct led_test_module_local {
32     int irq;
33     unsigned long mem_start;
34     unsigned long mem_end;
35     void __iomem *base_addr;
36 };
37
38 static irqreturn_t led_test_module_irq(int irq, void *lp)
39 {
40     printk("led_test_module interrupt\n");
41     return IRQ_HANDLED;
42 }
43
44 static int led_test_module_probe(struct platform_device *pdev)
45 {
46     struct resource *r_irq; /* Interrupt resources */
47     struct resource *r_mem; /* IO mem resources */
48     struct device *dev = &pdev->dev;
49     struct led_test_module_local *lp = NULL;
50
51     int rc = 0;
52
53     dev_info(dev, "Device Tree Probing\n");
54
```

```
55 /* Get iospce for the device */
56 r_mem = platform_get_resource(pdev, IORESOURCE_MEM, 0);
57 if (!r_mem) {
58     dev_err(dev, "invalid address\n");
59     return -ENODEV;
60 }
61
62 lp = (struct led_test_module_local *) kmalloc(sizeof(struct led_test_module_local), GFP_KERNEL);
63 if (!lp) {
64     dev_err(dev, "Could not allocate led_test_module device\n");
65     return -ENOMEM;
66 }
67
68 dev_set_drvdata(dev, lp);
69
70 lp->mem_start = r_mem->start;
71 lp->mem_end = r_mem->end;
72
73 if (!request_mem_region(lp->mem_start,
74     lp->mem_end - lp->mem_start + 1,
75     DRIVER_NAME)) {
76     dev_err(dev, "Couldn't lock memory region at %p\n",
77         (void *)lp->mem_start);
78     rc = -EBUSY;
79     goto error1;
80 }
81
82 lp->base_addr = ioremap(lp->mem_start, lp->mem_end - lp->mem_start + 1);
83 if (!lp->base_addr) {
84     dev_err(dev, "led_test_module: Could not allocate iomem\n");
85     rc = -EIO;
86     goto error2;
87 }
88
89 /* Get IRQ for the device */
90 r_irq = platform_get_resource(pdev, IORESOURCE_IRQ, 0);
91 if (!r_irq) {
92     dev_info(dev, "no IRQ found\n");
93     dev_info(dev, "led_test_module at 0x%08x mapped to 0x%08x\n",
94         (unsigned int __force)lp->mem_start,
95         (unsigned int __force)lp->base_addr);
96     return 0;
97 }
98 lp->irq = r_irq->start;
99
100 rc = request_irq(lp->irq, &led_test_module_irq, 0, DRIVER_NAME, lp);
101 if (rc) {
102     dev_err(dev, "testmodule: Could not allocate interrupt %d.\n",
103         lp->irq);
104     goto error3;
105 }
106
```

하

```
122 static int led_test_module_remove(struct platform_device *pdev)
123 {
124     struct device *dev = &pdev->dev;
125     struct led_test_module_local *lp = dev_get_drvdata(dev);
126     free_irq(lp->irq, lp);
127     release_mem_region(lp->mem_start, lp->mem_end - lp->mem_start + 1);
128     kfree(lp);
129     dev_set_drvdata(dev, NULL);
130     return 0;
131 }
132
133 #ifdef CONFIG_OF
134 static struct of_device_id led_test_module_of_match[] = {
135     { .compatible = "vendor,led_test_module", },
136     { /* end of list */ },
137 };
138 MODULE_DEVICE_TABLE(of, led_test_module_of_match);
139 #else
140 # define led_test_module_of_match
141 #endif
142
143
144 static struct platform_driver led_test_module_driver = {
145     .driver = {
146         .name = DRIVER_NAME,
147         .owner = THIS_MODULE,
148         .of_match_table = led_test_module_of_match,
149     },
150     .probe = led_test_module_probe,
151     .remove = led_test_module_remove,
152 };
153
154 static int __init led_test_module_init(void)
155 {
156     printk("<1>Hello module world.\n");
157     printk("<1>Module parameters were (0x%08x) and \"%s\"\n", myint,
158         mystr);
159
160     return platform_driver_register(&led_test_module_driver);
161 }
162
163
164 static void __exit led_test_module_exit(void)
165 {
166     platform_driver_unregister(&led_test_module_driver);
167     printk(KERN_ALERT "Goodbye module world.\n");
168 }
169
170 module_init(led_test_module_init);
171 module_exit(led_test_module_exit);
172
```

```
1  /*
2   * Placeholder PetaLinux user application.
3   *
4   * Replace this with your application code
5   */
6  #include <stdio.h>
7
8  int main(int argc, char *argv[])
9  {
10     printf("Hello, PetaLinux World!\n");
11     printf("cmdline args:\n");
12     while(argc--)
13         printf("%s\n", *argv++);
14
15     return 0;
16 }
17
18
19
```





- Wiki Home
- Recent Changes
- Pages and Files
- Members

Getting Started

Linux
U-Boot
Zynq AP SoC
Zynq UltraScale+ MPSoC
MicroBlaze

Linux Release Notes

Technical Articles
Release Images
Submit a Patch to Xilinx

Disclaimer
Privacy Policy
Wiki Terms of Use

Style Guide
Contact Us

Index of Tags
Xilinx.com

Cadence I2C Driver

Cadence I2C Linux driver for Zynq and Zynq Ultrascale+ MPSoC

Introduction

This page provides information about the Cadence I2C driver which can be found on Xilinx Git and mainline as i2c-cadence.c. Zynq has two I2C hard IP. I2C can be used as a master with this linux driver. I2C slave support is yet to be added in the driver. There is support for repeated start with some limitations.

HW IP Features

- Master mode
- Support 16 bytes FIFO
- Programmable normal and fast bus data rates
- Interrupt support
- Repeated start support using HOLD bit
- FIFO control using HOLD bit
- Slave monitoring support in Master mode.

Known issues and limitations

Repeated start after a read transfer is not supported by this controller; a warning is given when this condition is detected by the driver.

The following are the controller errata:

- Missing glitch filter (<http://www.xilinx.com/support/answers/61861.html>)
- I2C Master Generates Invalid Read Transactions (<http://www.xilinx.com/support/answers/61664.html>)
- Missing I2C Master Completion Interrupt (<http://www.xilinx.com/support/answers/61665.html>)
- Timing requirement violations
 - <http://www.xilinx.com/support/answers/59366.html>
 - <http://www.xilinx.com/support/answers/60693.html>
 - <http://www.xilinx.com/support/answers/60694.html>
- I2C Missing Arbitration On Repeated Start (<http://www.xilinx.com/support/answers/60695.html>)

Kernel Configuration Options for Driver

The following config options need to be enabled:

CONFIG_I2C_CADENCE



Kernel Configuration Options for Driver

The following config options need to be enabled:

`CONFIG_I2C_CADENCE`

It depends on I2C and ARCH_ZYNQ

Cadence I2C Controller

`CONFIG_I2C_CADENCE:`

Say yes here to select Cadence I2C Host Controller. This controller is e.g. used by Xilinx Zynq.

Symbol: `I2C_CADENCE` [=y]

Type : tristate

Prompt: Cadence I2C Controller

Location:

-> Device Drivers

-> I2C support

-> I2C support (I2C [=y])

-> I2C Hardware Bus support

Defined at drivers/i2c/busses/Kconfig:413

Depends on: `I2C` [=y] && `HAS_IOMEM` [=y] && `(ARCH_ZYNQ [=y])`



Devicetree

Refer to Documentation/devicetree/bindings/i2c/i2c-cadence.txt for complete description.

Example

The following example shows adding an I2C node to the devicetree with the various interfaces connected to i2c on zc702:

```
i2c0 {
    status = "okay";
    clock-frequency = <400000>;
    pinctrl-names = "default";
    pinctrl-0 = <pinctrl_i2c0_defaults>;
};
```

Linux Drivers

Edit

0

This page is intended to give more details on the Xilinx drivers for Linux, such as testing, how to use the drivers, known issues, etc. The drivers included in the kernel tree are intended to run on ARM (Zynq, Zynq Ultrascale+ MPSoC) and MicroBlaze Linux.

Driver Information

There are a number of drivers in the kernel tree due to history and they may work, but the following list of drivers are currently what's tested and users are encouraged to use these rather than others. Any other drivers, not in the mainline and only in the Xilinx tree, may be old and obsolete such that they could be removed at any time.

```
reg = <0>;
si570: clock-generator@5d {
    #clock-cells = <0>;
    compatible = "silabs,si570";
    temperature-stability = <50>;
    reg = <0x5d>;
    factory-fout = <156250000>;
    clock-frequency = <148500000>;
};

i2c@2 {
    #address-cells = <1>;
    #size-cells = <0>;
    reg = <2>;
    eeprom@54 {
        compatible = "at,24c08";
        reg = <0x54>;
    };
};
```

감사합니다