

실습 2-1: Cross Toolchain

1. 준비

Ubuntu에서 ~/ESP2018/chap02/toolchain 디렉토리를 만든다.

2. Linaro 크로스툴 설치

다운로드

```
ohheum@ubuntu:~/ESP2018/chap02/toolchain$ wget -c https://releases.linaro.org/
components/toolchain/binaries/6.4-2017.11/arm-linux-gnueabi/gcc-linaro-6.4.1-2017.11-
x86_64_arm-linux-gnueabi.tar.xz
```

```
ohheum@ubuntu:~/ESP2018/chap02/toolchain$ tar xf gcc-linaro-6.4.1-2017.11-x86_64_arm-
linux-gnueabi.tar.xz
```

압축해제

```
ohheum@ubuntu:~/ESP2018/chap02/toolchain$ cd gcc-linaro-6.4.1-2017.11-x86_64_arm-linux-
gnueabi/bin
```

> 아웃풋..파일을 덮어쓰는거>> 파일 뒤에 추가 하는거

```
ohheum@ubuntu:~/ESP2018/chap02/toolchain/gcc-linaro-6.4.1-2017.11-x86_64_arm-linux-
gnueabi/bin$ echo PATH='$PATH':${PWD} >> ~/.profile
```

변수 값이 들어간다

주의: \$PATH를 둘러싼 따옴표는 retun키 옆에 있는 그냥
홀 따옴표이다.

```
ohheum@ubuntu:~/ESP2018/chap02/toolchain/gcc-linaro-6.4.1-2017.11-x86_64_arm-linux-
gnueabi/bin$ cat ~/.profile
```

...

```
# set PATH so it includes user's private bin directories
```

```
PATH="$HOME/bin:$HOME/.local/bin:$PATH"
```

```
PATH=$PATH:/home/ohheum/ESP2018/chap02/toolchain/gcc-linaro-6.4.1-2017.11-x86_64_arm-
linux-gnueabi/bin
```

PATH 환경변수가 제대로 설정되었음을 확인한다.

셸 스크립트를 작성하고 나면 먼저 function을 설정해줘야 한다. chmod

```
ohheum@ubuntu:~/ESP2018/chap02/toolchain/gcc-linaro-6.4.1-2017.11-x86_64_arm-linux-
gnueabi/bin$ source ~/.profile
```

이렇게하면 창을 새로 띄우면 효과가 없다.

만약 터미널을 여러 개 사용 중이라면 다른 터미널에서도 마지막 명령(source ~/.profile)을 실행해준다. 혹은
ubuntu를 한 번 logout했다가 다시 login한다.

3. 크로스툴 구성요소 살펴보기

C 라이브러리, 컴파일러 및 유틸리티들을 찾아본다.

4. 크로스툴 테스트하기

간단한 C 프로그램을 작성하여 컴파일해본다.

```
ohheum@ubuntu:~/ESP2018/chap02/toolchain$ cat test.c
#include <stdio.h>
int main()
{
```

무언가를 찾을때 명령어 :
 which : 일반적으로 명령어를 찾을 때 사용
 PATH변수를 사용해서 무언가를 찾는다
 로케이트 : 인덱스를 이용, 빠른 속도로 검색한다. 몇일에 한번 업데이트한다.
 find : 무식하게 찾을때 사용

```
printf("Hello Arm !\n");
}
```

```
ohheum@ubuntu:~/ESP2018/chap02/toolchain$ arm-linux-gnueabi-gcc test.c -o test
```

```
ohheum@ubuntu:~/ESP2018/chap02/toolchain$ ls
gcc-linaro-6.4.1-2017.11-x86_64_arm-linux-gnueabi-gcc      test      test.elf
gcc-linaro-6.4.1-2017.11-x86_64_arm-linux-gnueabi-gcc.tar.xz  test.c
```

```
ohheum@ubuntu:~/ESP2018/chap02/toolchain$ file test
test: ELF 32-bit LSB executable, ARM, EABI5 version 1 (SYSV), dynamically linked (uses
shared libs), for GNU/Linux 2.6.32,
BuildID[sha1]=4dbdae81da4059286fecb4ab1a822e4e2f57a7e2, not stripped
```

test가 arm binary 임을 알 수 있다. 즉 cross compile 이 된 것이다.

```
ohheum@ubuntu:~/ESP2018/chap02/toolchain$ ./test
bash: ./test: cannot execute binary file: Exec format error
```

PC에서 실행하면 실행이 안되는 것이 당연하다.

```
ohheum@ubuntu:~/ESP2018/chap02/toolchain$ arm-linux-gnueabi-readelf -a test >
test.elf
```

```
ohheum@ubuntu:~/ESP2018/chap02/toolchain$ cat test.elf | grep libc
0x00000001 (NEEDED)                               Shared library: [libc.so.6]
0002055c 00000216 R_ARM_JUMP_SLOT 00000000 __libc_start_main
      2: 00000000      0 FUNC      GLOBAL DEFAULT  UND __libc_start_main@GLIBC_2.4 (2)
     93: 00010431      2 FUNC      GLOBAL DEFAULT  13 __libc_csu_fini
    103: 00000000      0 FUNC      GLOBAL DEFAULT  UND __libc_start_main@@GLIBC_
    107: 000103e9     72 FUNC      GLOBAL DEFAULT  13 __libc_csu_init
000000: Version: 1 File: libc.so.6 Cnt: 1
```

/는 전체에서 찾을때
 . 은 현재 디렉토리나 하부디렉토리에서 찾을때 사용

```
ohheum@ubuntu:~/ESP2018/chap02/toolchain$ find . -name "libc.so.6" 와일드카드도 사용가능
./gcc-linaro-6.4.1-2017.11-x86_64_arm-linux-gnueabi-gcc/arm-linux-gnueabi-gcc/libc/lib/
libc.so.6
```

shared library——동적 라이브러리

실행파일을 sftp로 BBB에 전송한다. BBB가 Ubuntu machine에 연결되어 있어야 한다.

```
ohheum@ubuntu:~/ESP2018/chap02/toolchain$ sftp debian@192.168.7.2
Debian GNU/Linux 7
```

BeagleBoard.org BeagleBone Debian Image 2014-04-23

Support/FAQ: http://elinux.org/Beagleboard:BeagleBoneBlack_Debian
 debian@192.168.7.2's password:

Connected to 192.168.7.2.

```
sftp> lpwd
```

Local working directory: /home/ohheum/ESP2018/chap02

```
sftp> ll
```

```
bootloader hello.c hellostatic rootfs testsin.c
hello hello.c~ kernel testsin toolchain
```

```
sftp> pwd
```

Remote working directory: /home/debian

```
sftp> put test
```

Uploading test to /home/debian/test

```
test 100% 10KB 10.0KB/s 00:00
```

```
sftp> quit
```

상대방위치를 알때 pwd ,
나의 위치를 알때 lpwd

I 이 안붙으면 상대방.

ssh로 BBB에 접속하여 전송한 파일을 실행해본다.

```
ohheum@ubuntu:~/ESP2018/chap02/toolchain$ ssh debian@192.168.7.2
```

```
Debian GNU/Linux 7
```

```
BeagleBoard.org BeagleBone Debian Image 2014-04-23
```

```
Support/FAQ: http://elinux.org/Beagleboard:BeagleBoneBlack\_Debian
```

```
debian@192.168.7.2's password:
```

```
Last login: Wed Apr 20 00:27:34 2016
```

```
debian@beaglebone:~$ ls
```

```
bin Desktop hello test testsin
```

```
debian@beaglebone:~$ ./test
```

```
Hello Arm !
```

BBB에서는 제대로
실행된다.