개발환경 구축:
Cross compiler, bootp, tftp, NFS 등

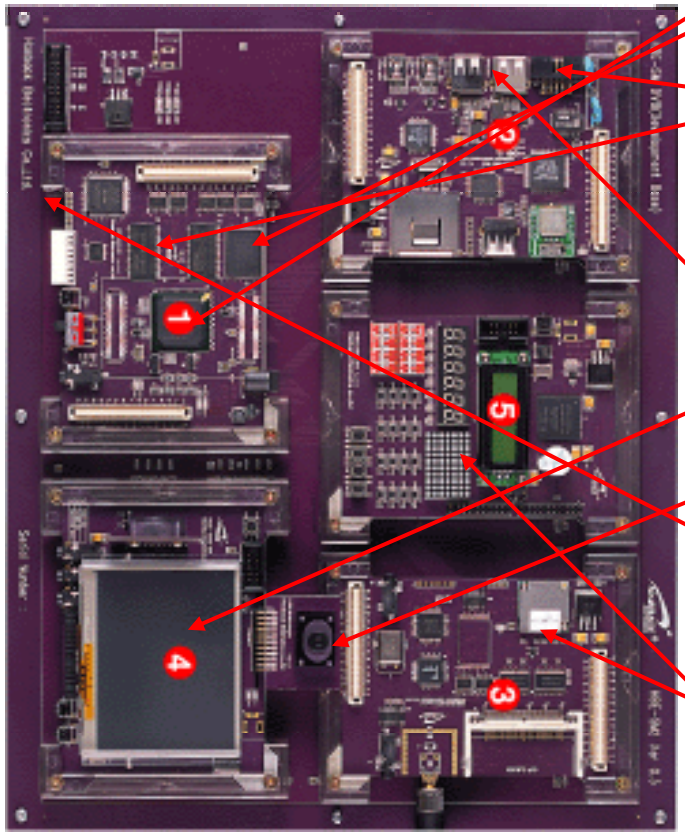
단국대학교
컴퓨터학과
2009
백승재
baeksj@dankook.ac.kr
http://embedded.dankook.ac.kr/~baeksj

강의 목표

- Target board에 대한 기본적인 이해
- 개발 과정 이해 및 환경 구축
- 개발 사례 파악

Target board

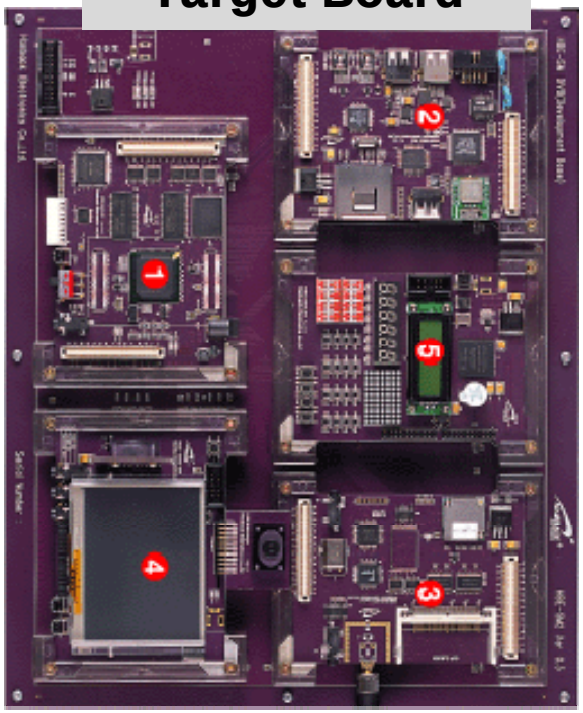
- HBE-SM II-P320



Items	Hardware Specifications
CPU	Marvell PXA320B1C806 (806MHz)
Memory	NAND Flash Memory : Samsung K9F1G08U0B (128M x 8bit) Mobile-DDR RAM : 128MByte (512Mbit x 2ea)
FPGA	Altera Cyclone-2 (EP2C8F256C8ES) x 2ea
CPLD	Altera MAXII (EPM1270GF256C5) x 1ea
SRAM	Samsung K6R4016V1C (256Kx16 Bit) x 3ea
Ethernet	1 Port 10/100Base-T Ethernet
Audio	AC'97 Audio Codec (Built-in Touch Screen I /F) 1 Port Stereo Out (Stereo Jack), 1 Port Line IN (Stereo Jack), 1 Port MIC IN (Stereo Jack)
LCD	3.5" TFT LCD 1ea
Touch Screen	3.5 Touch Screen x 1ea
Camera	1.3M Pixel CMOS Camera x 2ea
USB	USB 1.1 Host : A type Connector x 1ea USB 1.1 Client : Mini type Connector x 1ea USB 2.0 Host : A type Connector x 1ea USB 2.0 OTG : Mini type Connector x 1ea
UART	RS232 Level : Stereo type Connector x 3ea 3.3V TTL Level : 2 Port for GPS & Bluetooth : 1 Port
I^2C	1 Port
IrDA	1 Port (Compliant to IrDA 1.1)
Storage	1ea CF Connector 1ea SD / MMC Connector
Wireless LAN	1 Port Wireless LAN (IEEE802.11b/g)
I / O Application	Text LCD, 7-Segment, Dot Matrix (7 x 5 x 2ea), Key Pad (4 x 4), DIP Switch (8point x 2ea), LED (8ea), Buzzer, Tact Switch (4ea)
I / F Module	1ea Bluetooth Module (On-Board)

Step1: Serial port & minicom settup

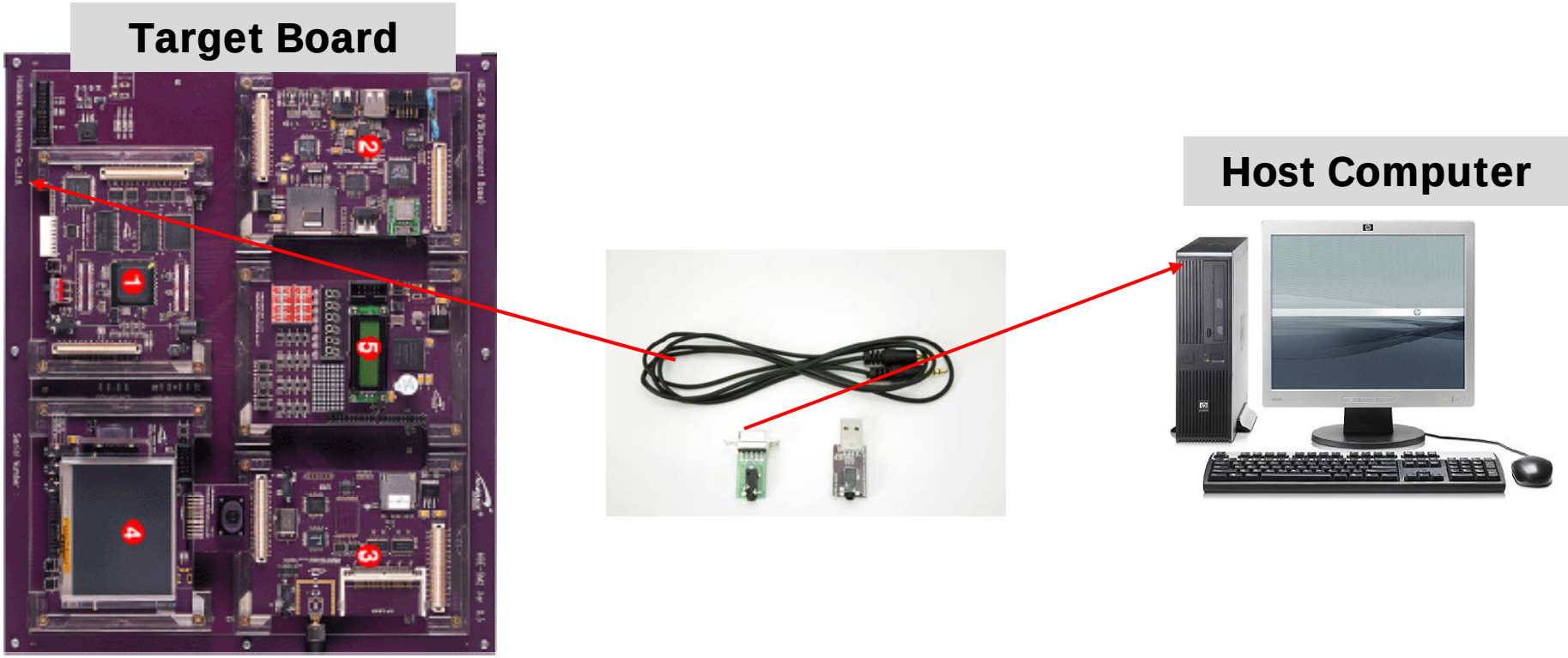
Target Board



Host Computer

- Target board have no ‘monitor’
- Serial port와 minicom 프로그램을 이용해 보드에 가상적인 모니터를 만들어 주자

Step1: Serial port & minicom settup



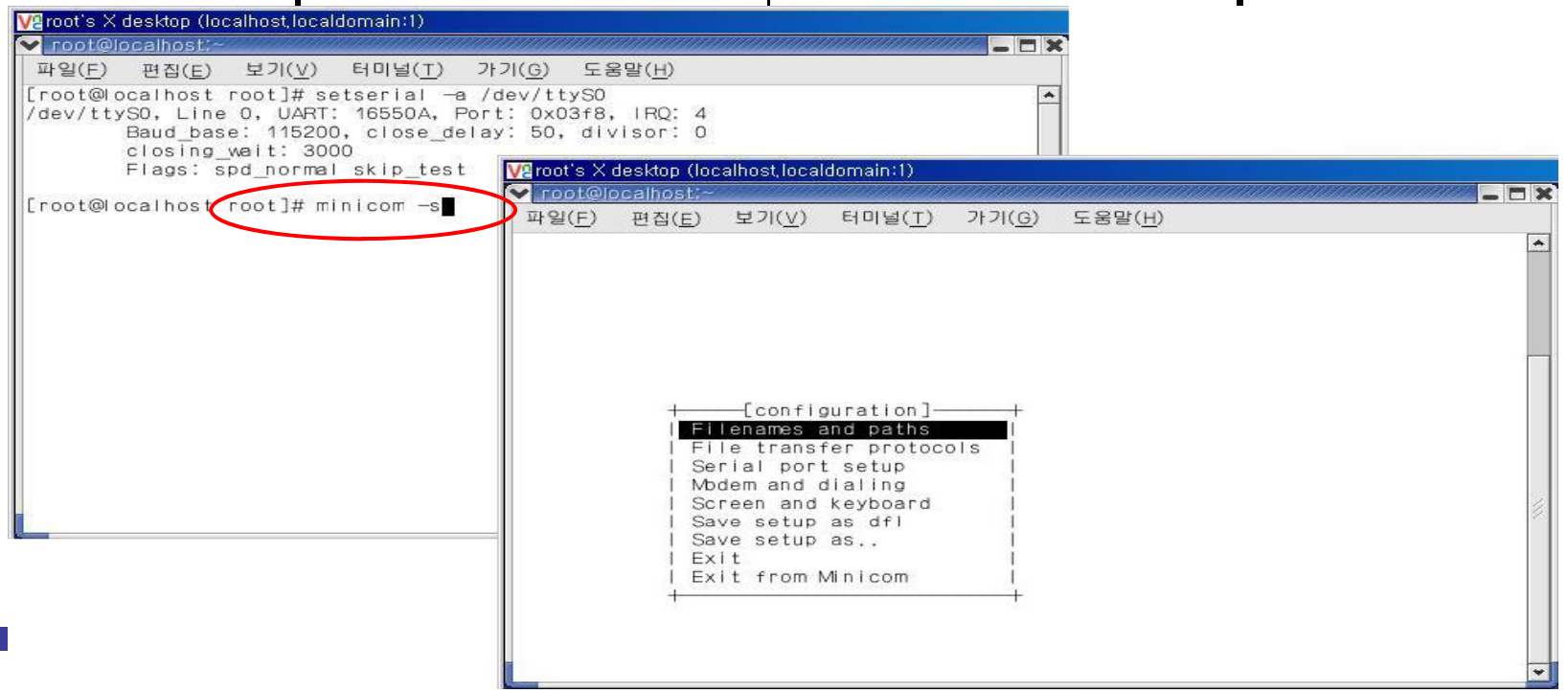
- Serial이나 USB 둘 중 하나만 사용하여 target board와 host computer를 연결

Step1: Serial port & minicom settup

- minicom : 리눅스용 터미널 에뮬레이터

  2. Minicom 환경 설정

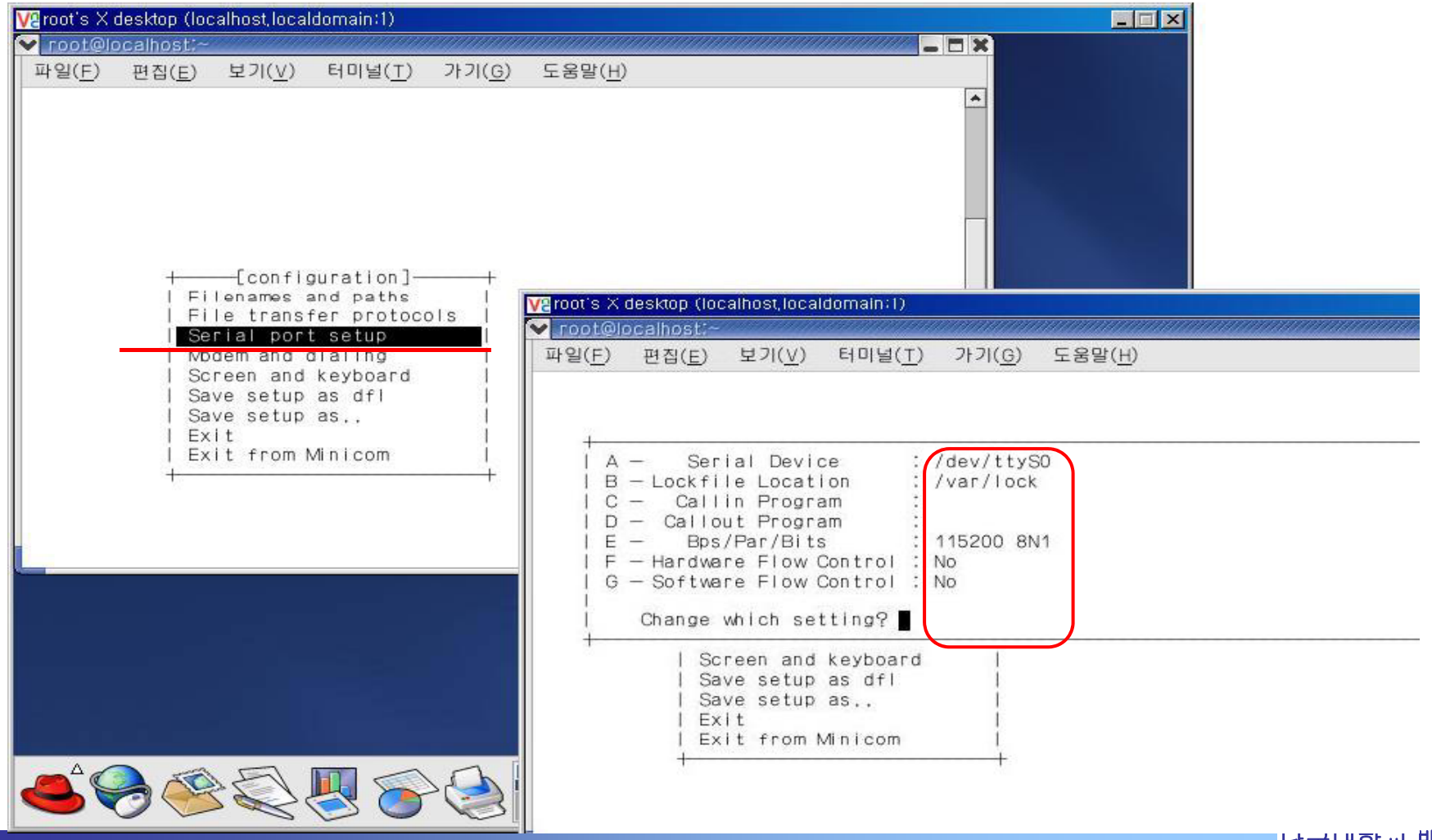
Boudrate	115200bps
Date Size	8
Parity	None
Stop bits	1
H/W flow control	NO
S/W flow control	NO



Step1: Serial port & minicom settup

- minicom : 리눅스용 터미널 에뮬레이터

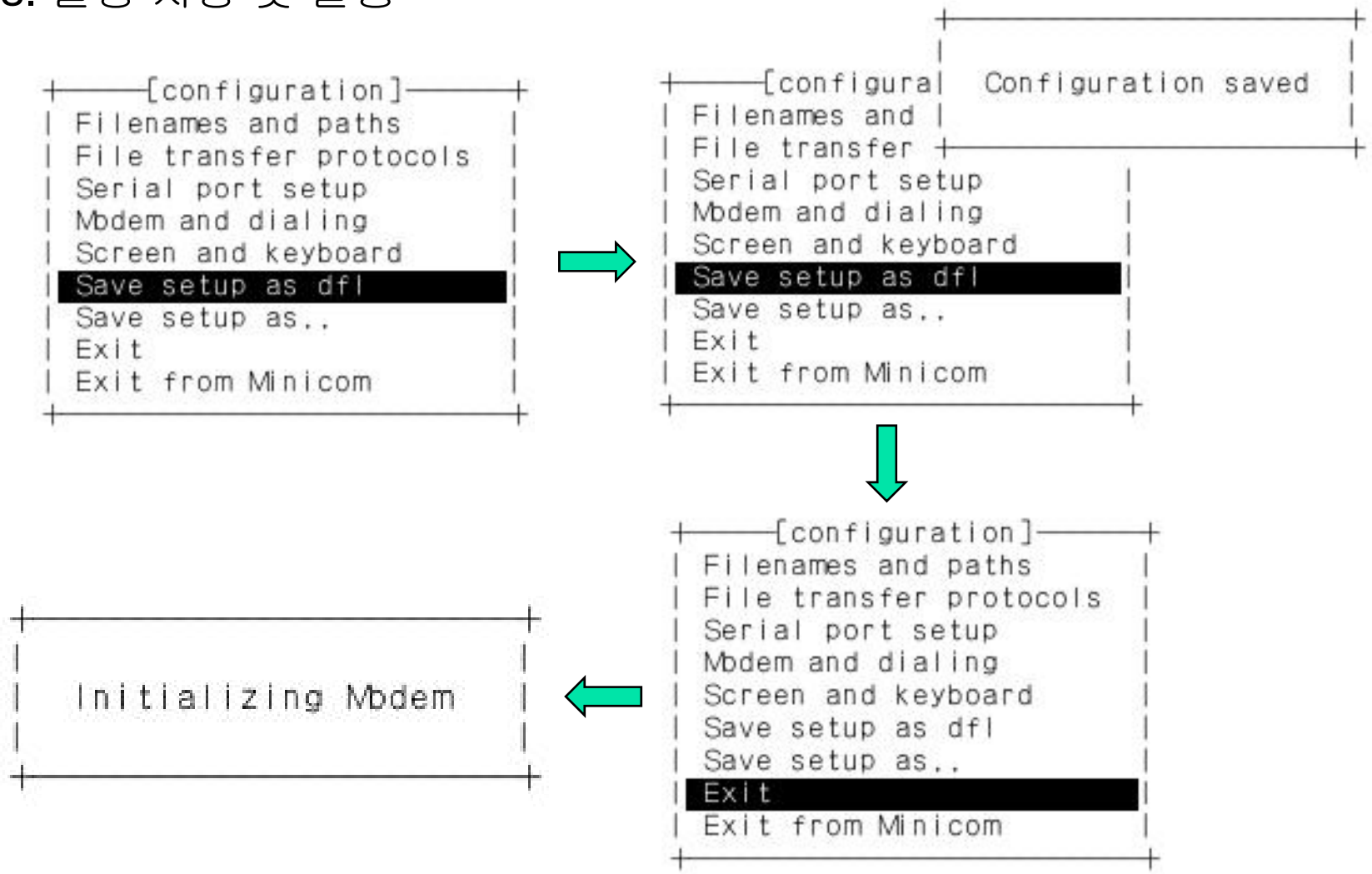
  2. Minicom 환경 설정



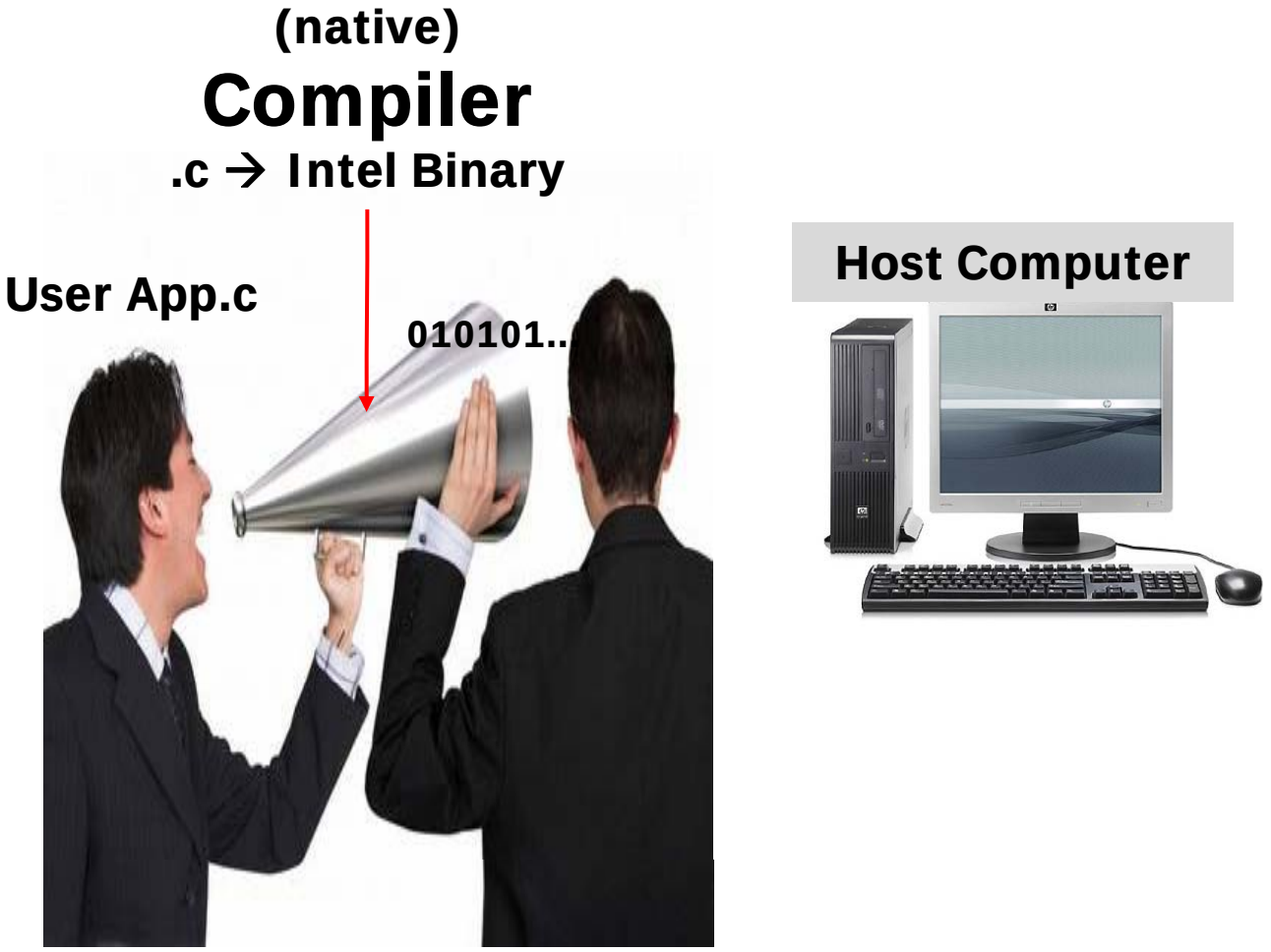
Step1: Serial port & minicom settup

- minicom : 리눅스용 터미널 에뮬레이터

  3. 설정 저장 및 실행

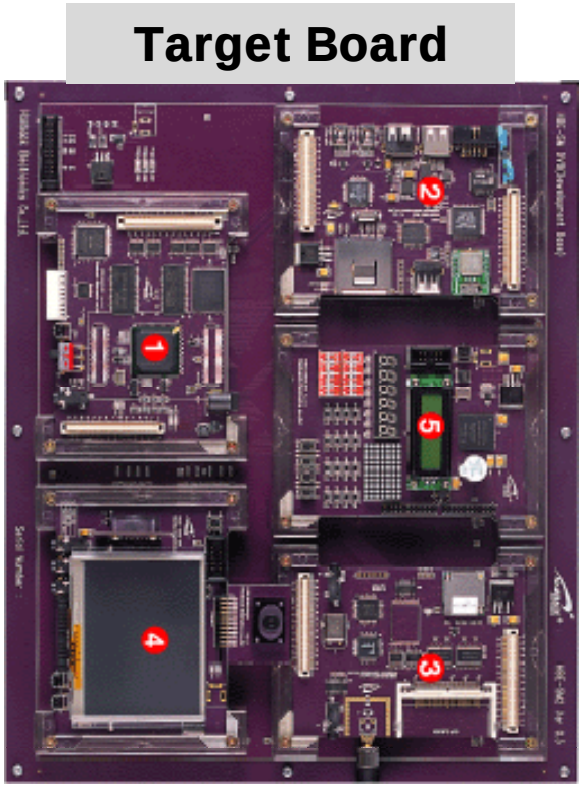


Step2: Cross compiler



- Native compiler
  - ✓ generates code for its own execution environment

Step2: Cross compiler

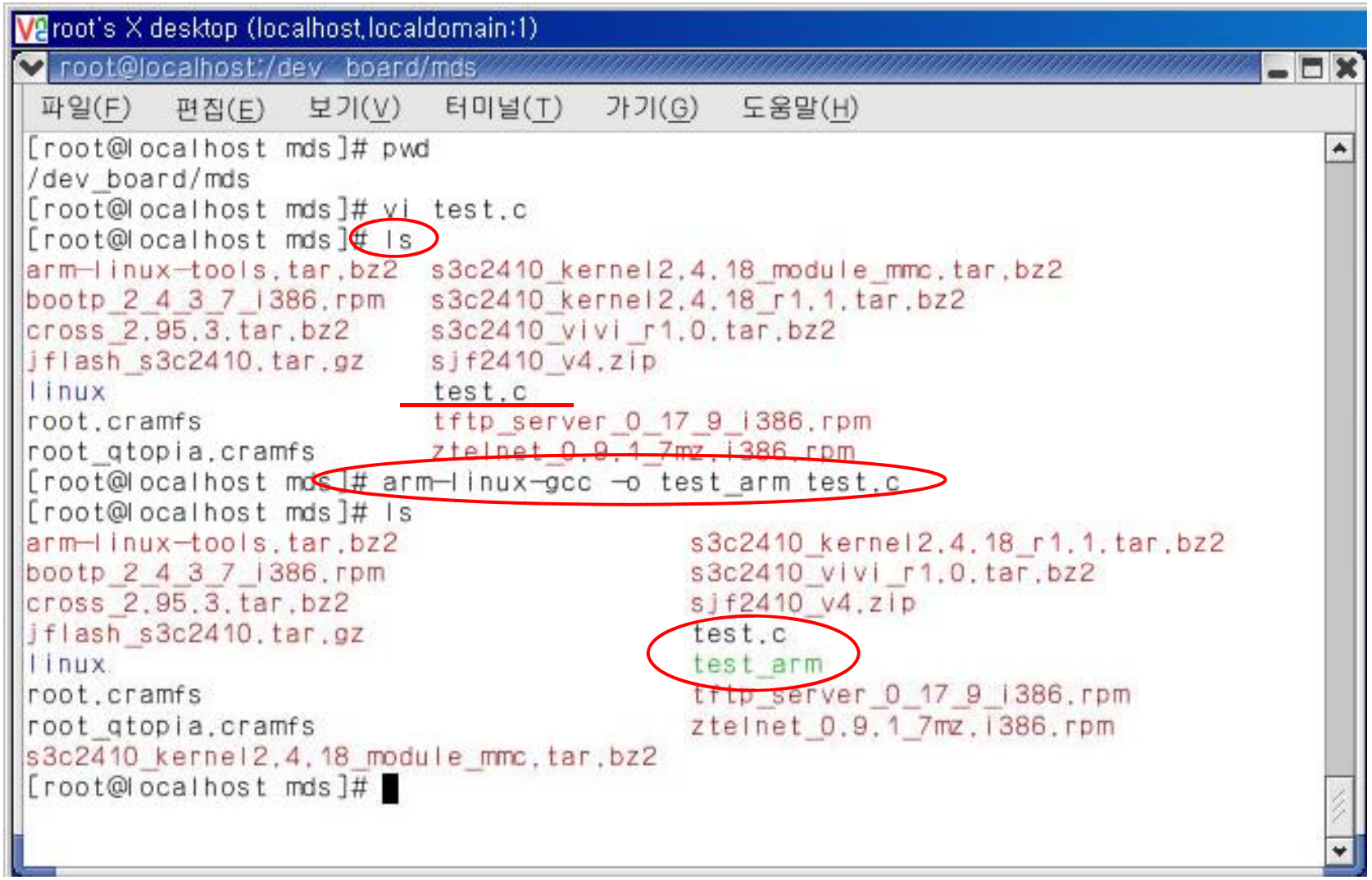


- Cross compiler
  - ✓ executes in one environment and generates code for another

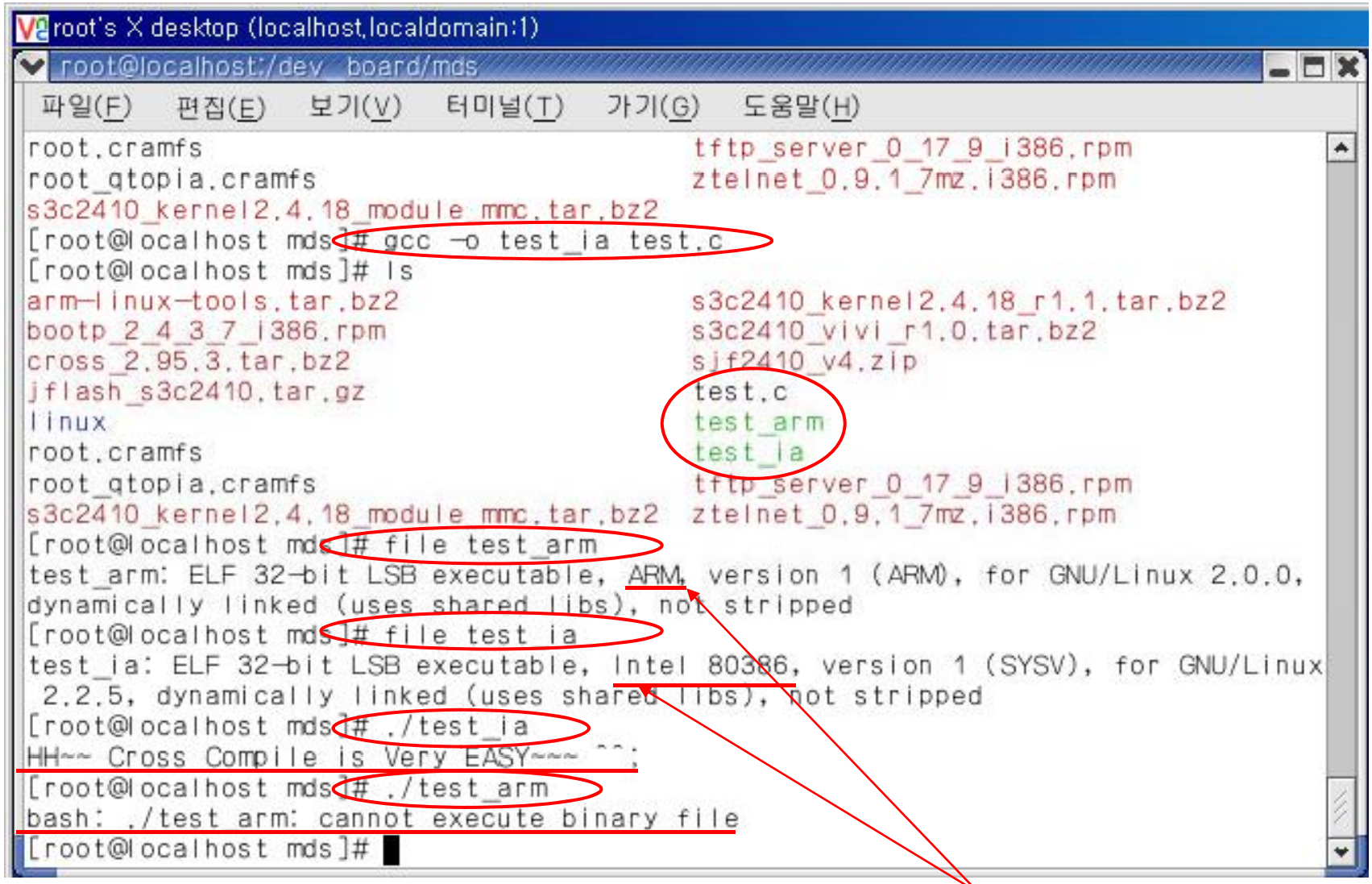
Step2: Cross compiler

- Cross compiler
  - ✓ Modify the compiler, compile the compiler and use it
  - ✓ Download the compiler(already modified) for your architecture
    - tar.gz
    - rpm, ...
- HBE-SMII-P320 보드의 경우 CD에 cross compiler를 제공한다
  - ✓ /usr/local 디렉토리에 압축해제 한 후
  - ✓ PATH 설정 하여 사용하면 된다

Step2: Cross compiler



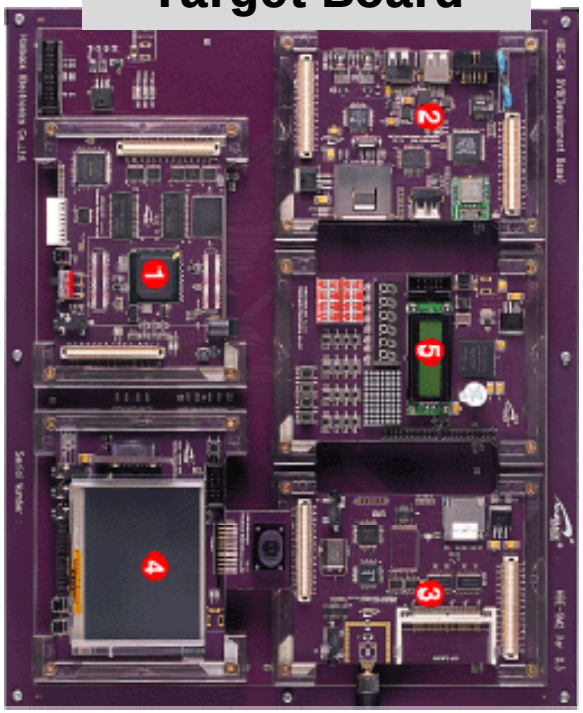
Step2: Cross compiler



각 파일이 어느 아키텍쳐에서 실행가능한지 확인할 수 있다.

Step3: JTAG

Target Board

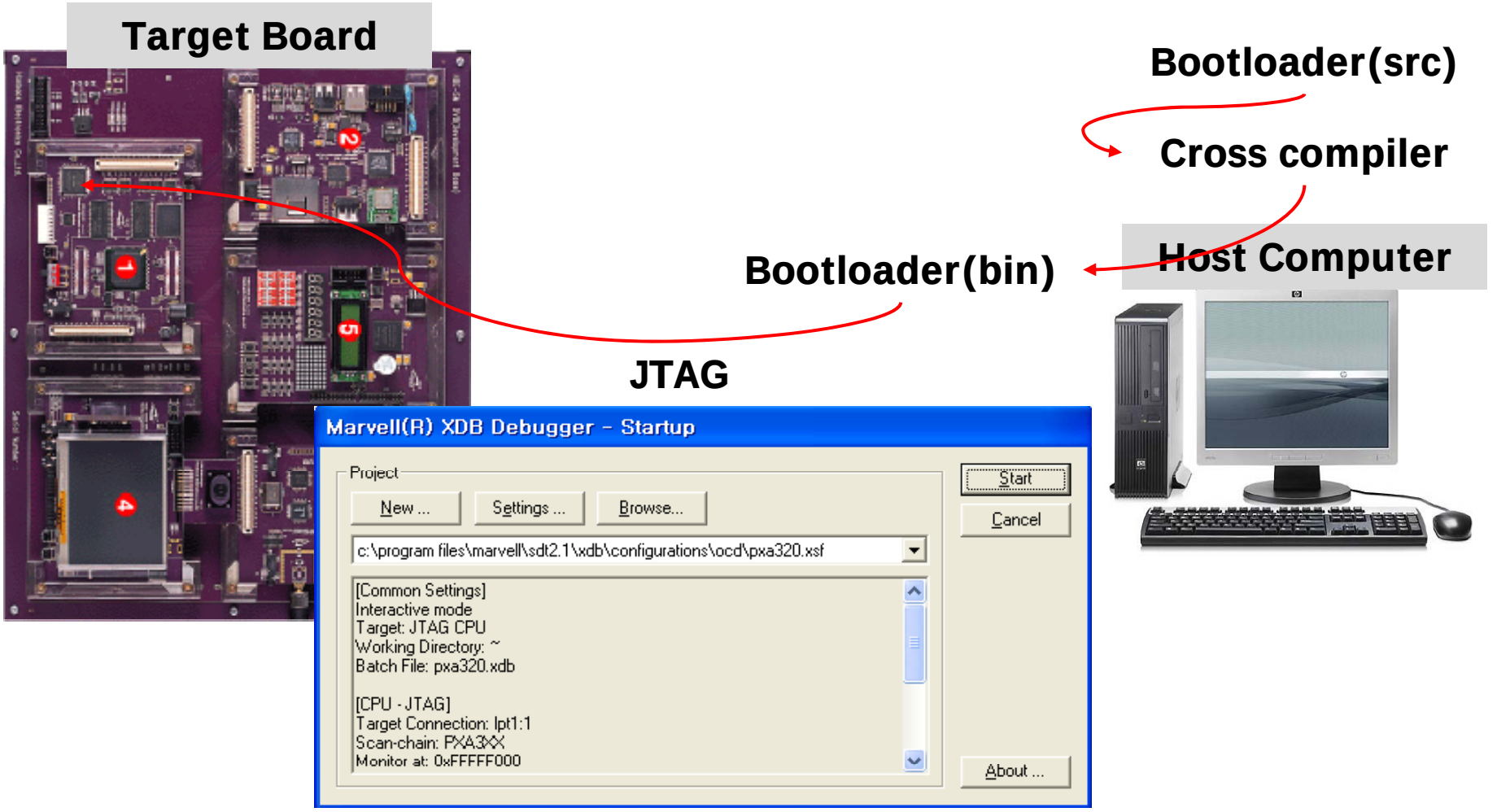


Host Computer

- PC of ARM cpu have the value 0x00000000 after power on
- Bootloader must be written in 0x00000000
- NOR flash memory located in 0x00000000

- How can you writing bootloader into NOR flash memory?

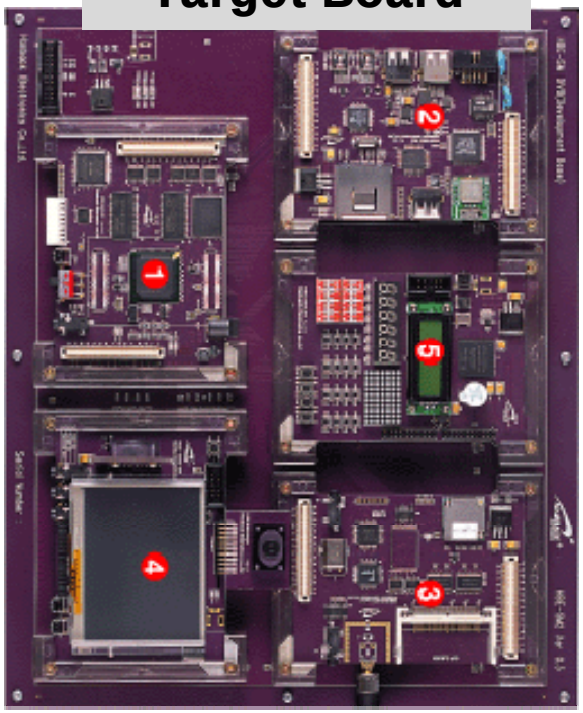
Step3: JTAG



- In the Linux?

Step4: bootp

Target Board

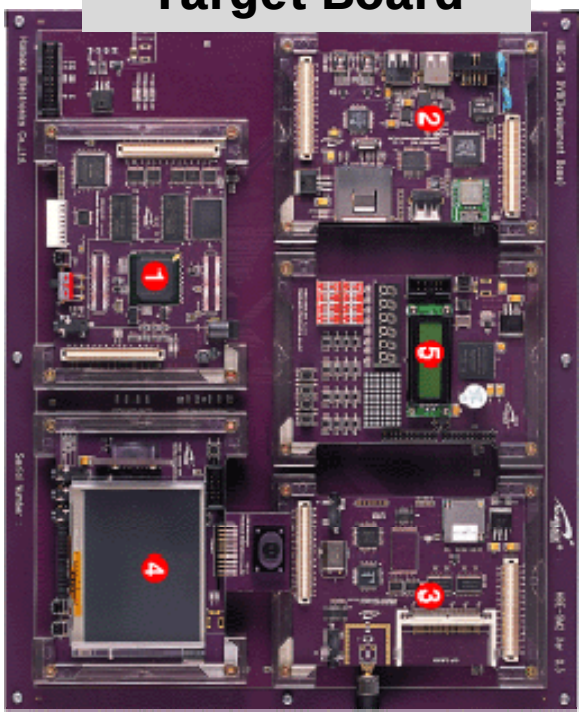


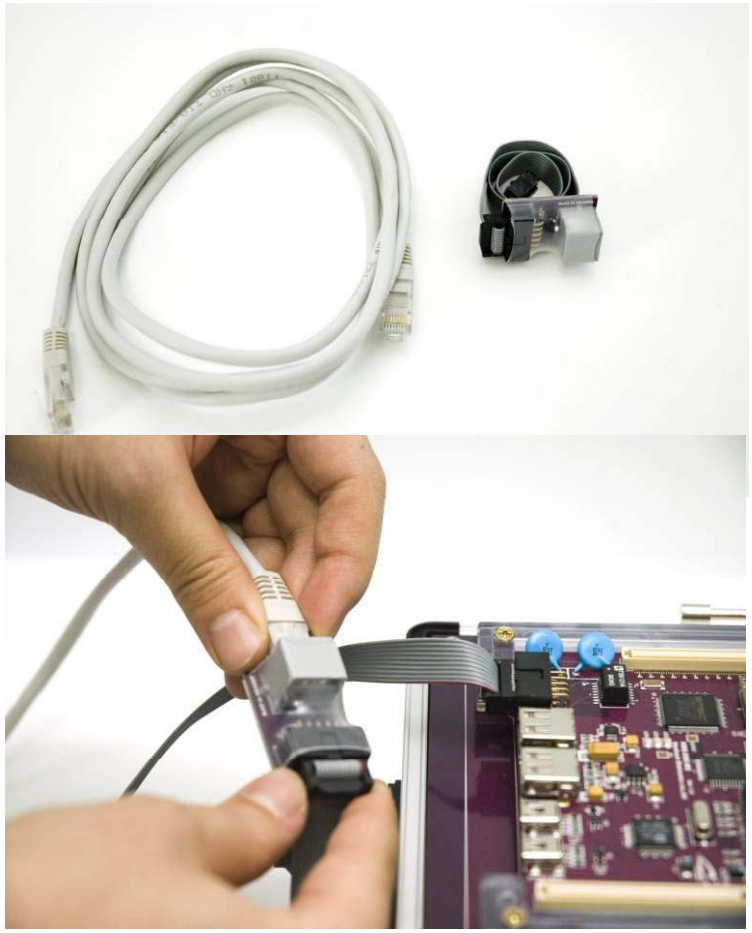
Host Computer

- Bootloader will running on the target board
  - ✓ You can use peripherals
- Kernel image and rootfilesystem also should be written to Flash memory on the board
- Use faster interface than JTAG

Step4: bootp

Target Board

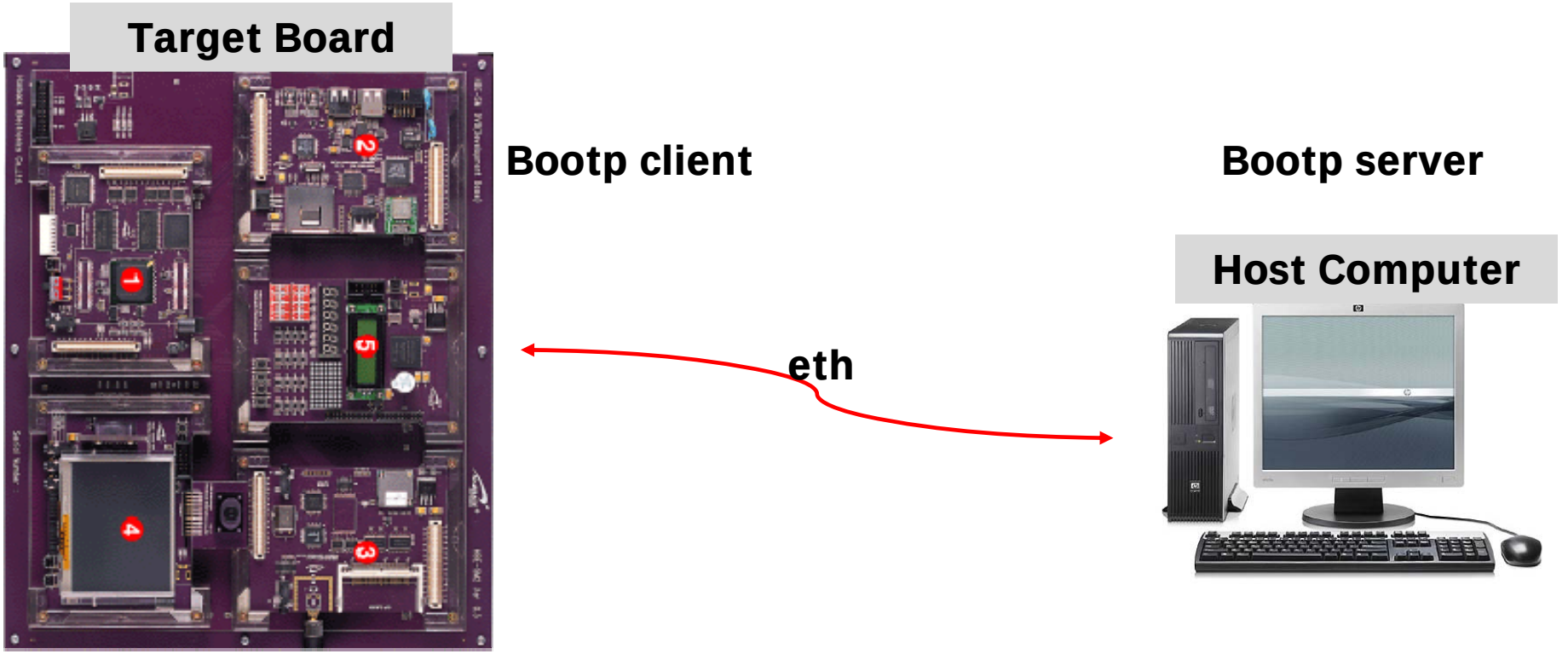




Host Computer

- (Cross)Ethernet cable 연결

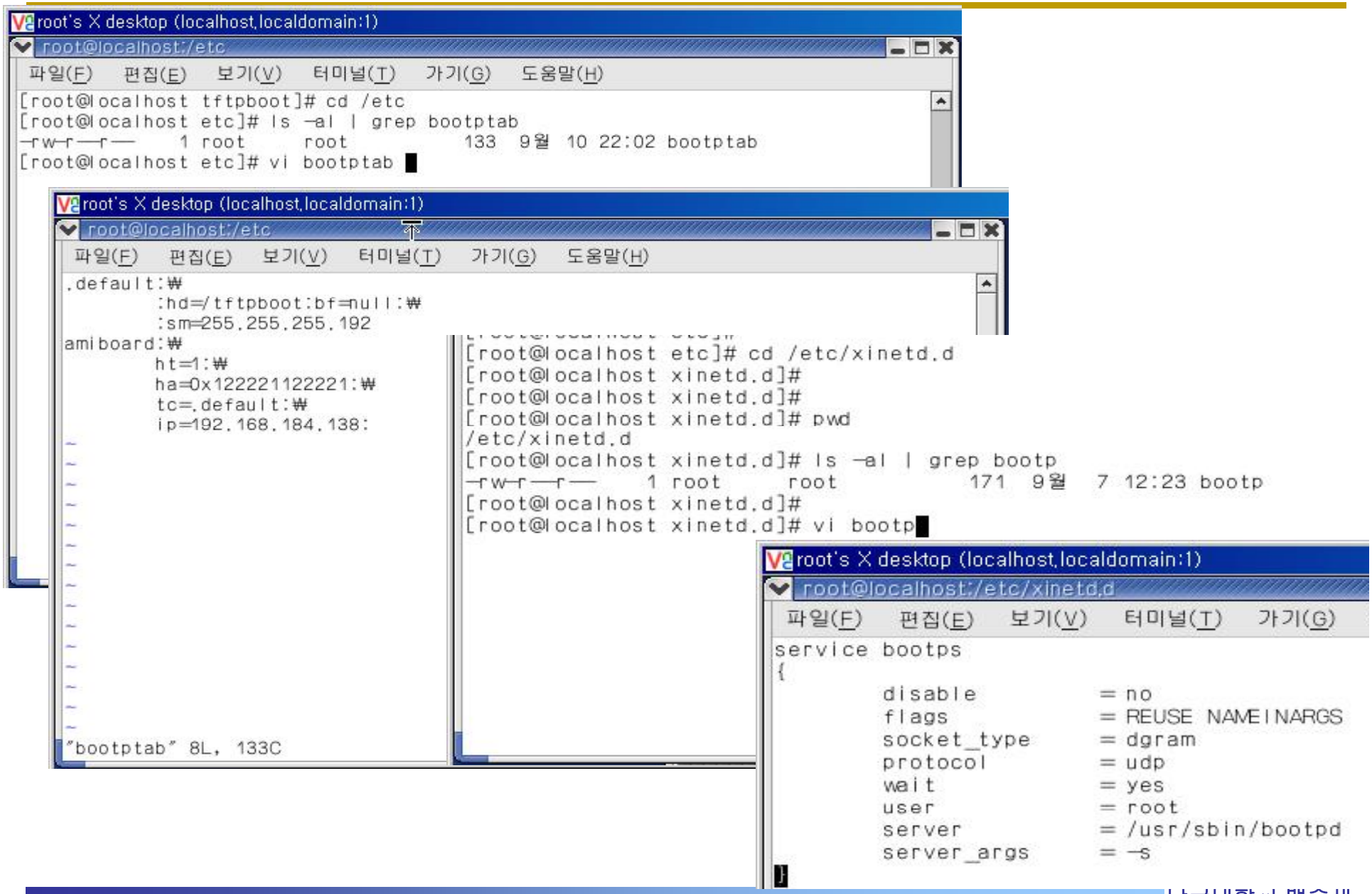
Step4: bootp



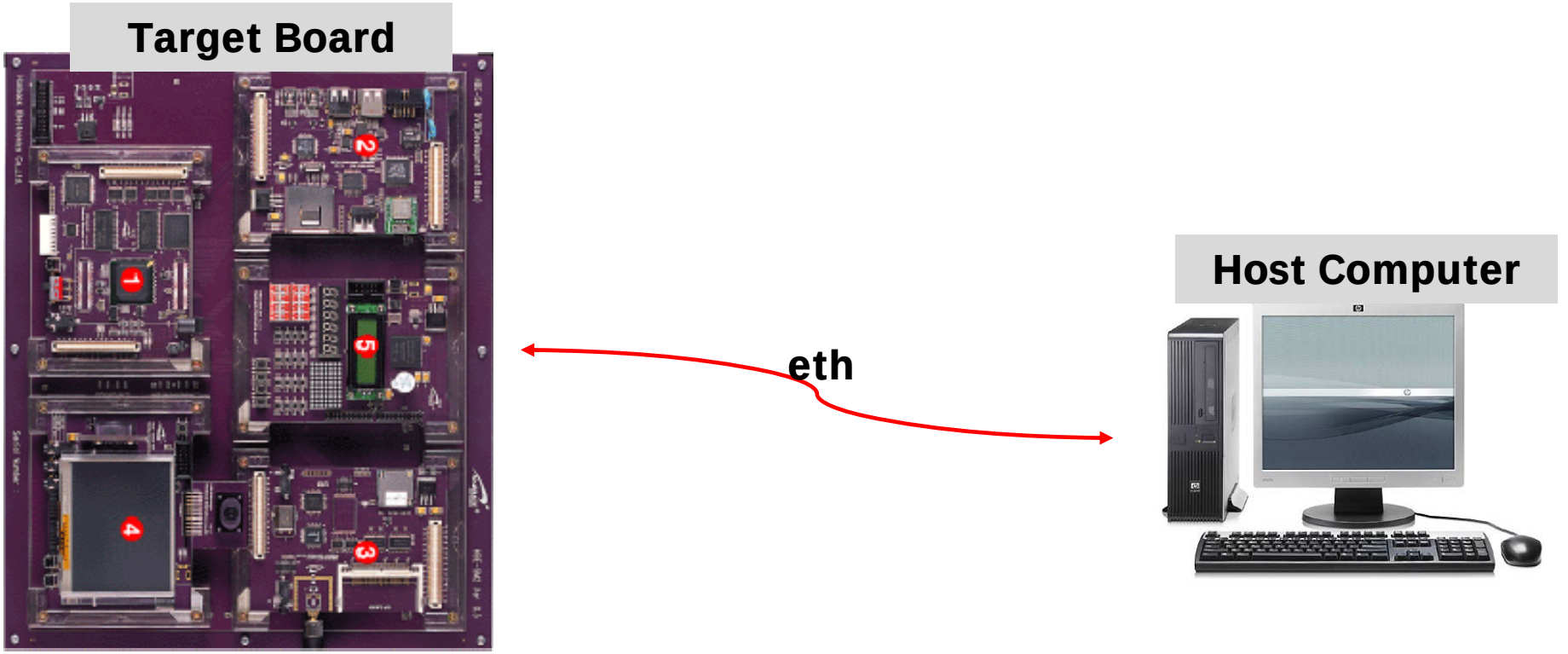
- IP address required(for target board)
- You can get IP address automatically using DHCP if there is DHCP server

- Bootp (your host computer shoud be bootp server)

Step4: bootp


root's X desktop (localhost.localdomain:1)
root@localhost:/etc
파일(F) 편집(E) 보기(V) 터미널(T) 가기(G) 도움말(H)
[root@localhost tftpboot]# cd /etc
[root@localhost etc]# ls -al | grep bootptab
-rw-r--r-- 1 root root 133 9월 10 22:02 bootptab
[root@localhost etc]# vi bootptab
root's X desktop (localhost.localdomain:1)
root@localhost:/etc
파일(F) 편집(E) 보기(V) 터미널(T) 가기(G) 도움말(H)
.default:₩
:hd=/tftpboot:bf=null:₩
:sm=255.255.255.192
amiboard:₩
ht=1:₩
ha=0x122221122221:₩
tc=.default:₩
ip=192.168.184.138:
"bootptab" 8L, 133C
[root@localhost etc]# cd /etc/xinetd.d
[root@localhost xinetd.d]#
[root@localhost xinetd.d]#
[root@localhost xinetd.d]# pwd
/etc/xinetd.d
[root@localhost xinetd.d]# ls -al | grep bootp
-rw-r--r-- 1 root root 171 9월 7 12:23 bootp
[root@localhost xinetd.d]#
[root@localhost xinetd.d]# vi bootp
root's X desktop (localhost.localdomain:1)
root@localhost:/etc/xinetd.d
파일(F) 편집(E) 보기(V) 터미널(T) 가기(G)
service bootps
{
disable = no
flags = REUSE NAMEINARGS
socket_type = dgram
protocol = udp
wait = yes
user = root
server = /usr/sbin/bootpd
server_args = -s
}

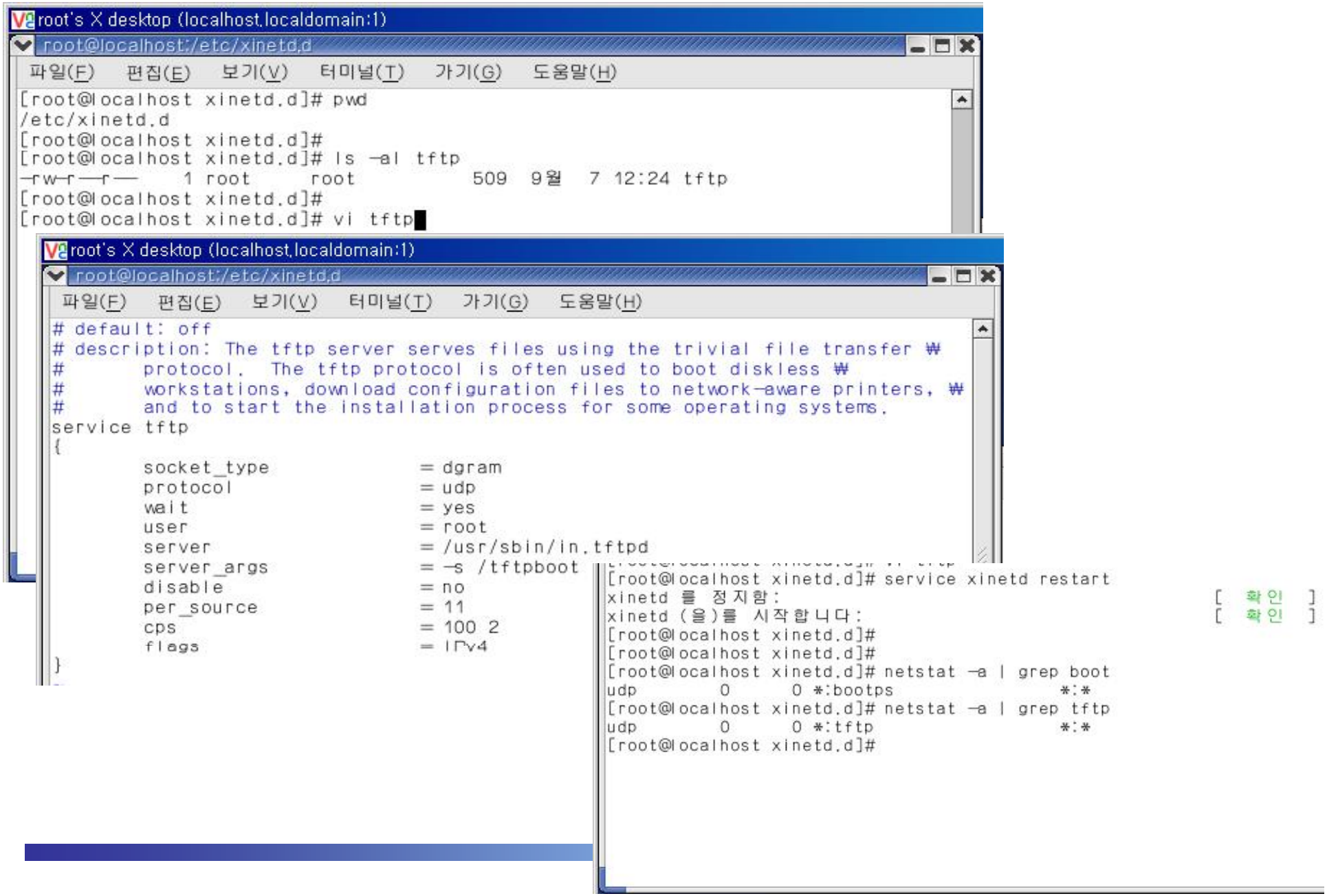
Step4: tftp



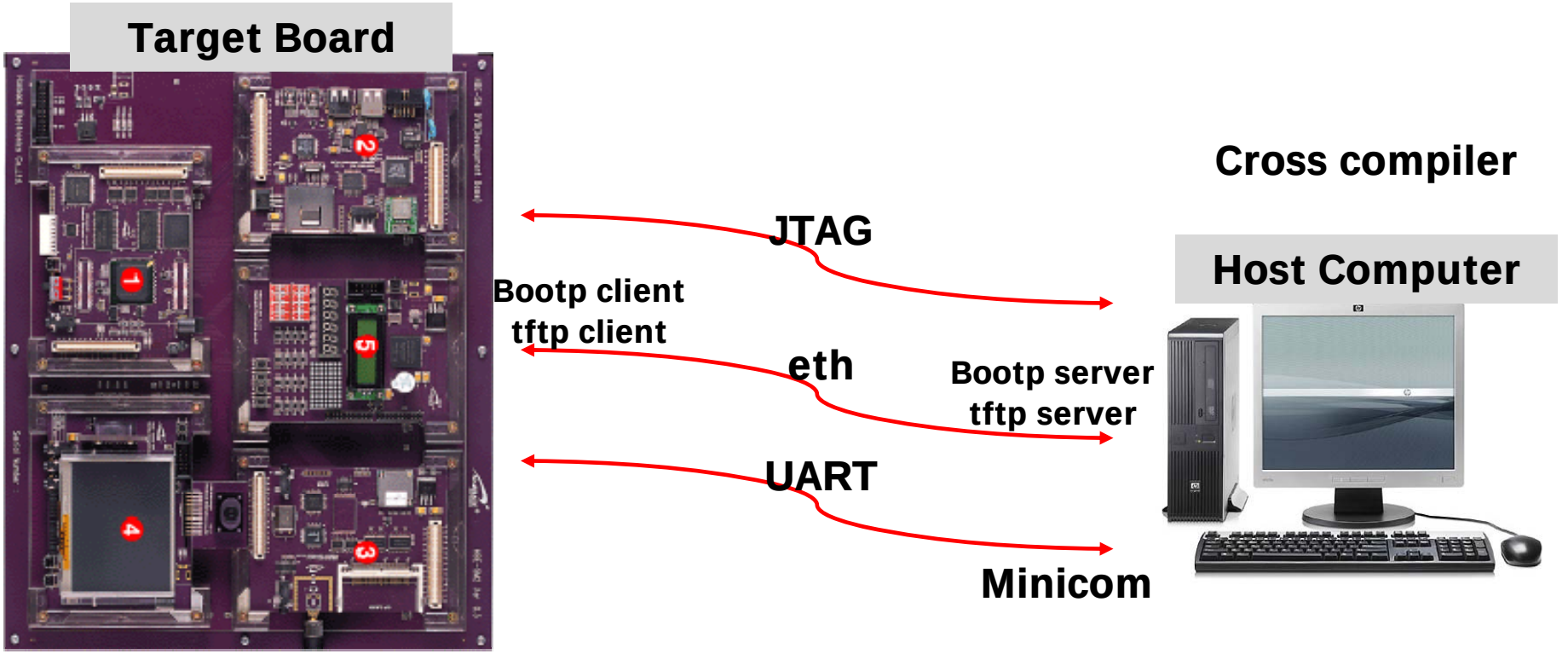
- What program do you use when you download some files from server?
- You can download file from FTP server using FTP client program

- TFTP(your host computer should be tftp server)

Step4: tftp

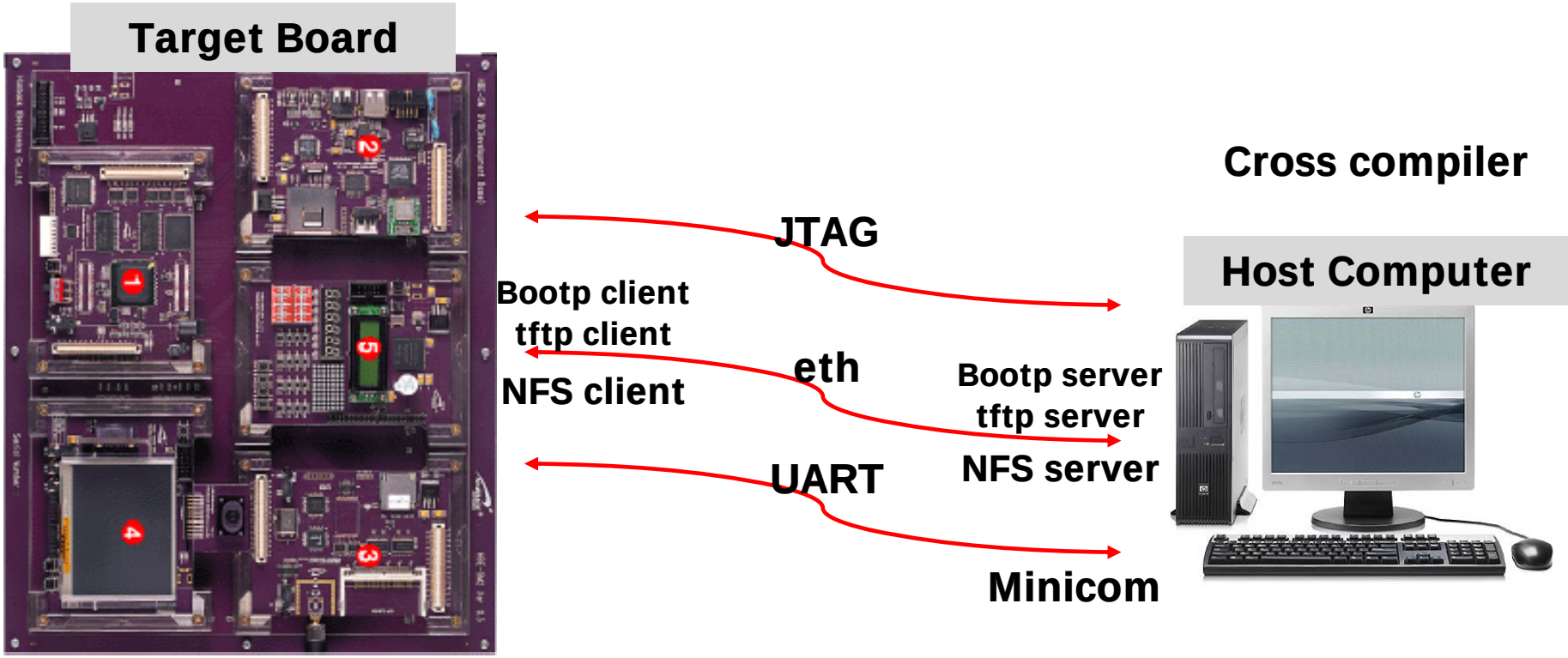

root's X desktop (localhost.localdomain:1)
root@localhost:/etc/xinetd.d
파일(F) 편집(E) 보기(V) 터미널(T) 가기(G) 도움말(H)
[root@localhost xinetd.d]# pwd
/etc/xinetd.d
[root@localhost xinetd.d]#
[root@localhost xinetd.d]# ls -al tftp
-rw-r--r-- 1 root root 509 9월 7 12:24 tftp
[root@localhost xinetd.d]#
[root@localhost xinetd.d]# vi tftp
root's X desktop (localhost.localdomain:1)
root@localhost:/etc/xinetd.d
파일(F) 편집(E) 보기(V) 터미널(T) 가기(G) 도움말(H)
default: off
description: The tftp server serves files using the trivial file transfer ₩
protocol. The tftp protocol is often used to boot diskless ₩
workstations, download configuration files to network-aware printers, ₩
and to start the installation process for some operating systems.
service tftp
{
socket_type = dgram
protocol = udp
wait = yes
user = root
server = /usr/sbin/in.tftpd
server_args = -s /tftpboot
disable = no
per_source = 11
cps = 100 2
flags = IPv4
}
[root@localhost xinetd.d]# service xinetd restart
xinetd 를 정지함: [확인]
xinetd (을)를 시작합니다: [확인]
[root@localhost xinetd.d]#
[root@localhost xinetd.d]#
[root@localhost xinetd.d]# netstat -a | grep boot
udp 0 0 *:bootps *:*
[root@localhost xinetd.d]# netstat -a | grep tftp
udp 0 0 *:tftp *:*
[root@localhost xinetd.d]#

전체 구성도


Target Board
Cross compiler
JTAG
Host Computer
Bootp client
tftp client
eth
Bootp server
tftp server
UART
Minicom

Linux booting 확인

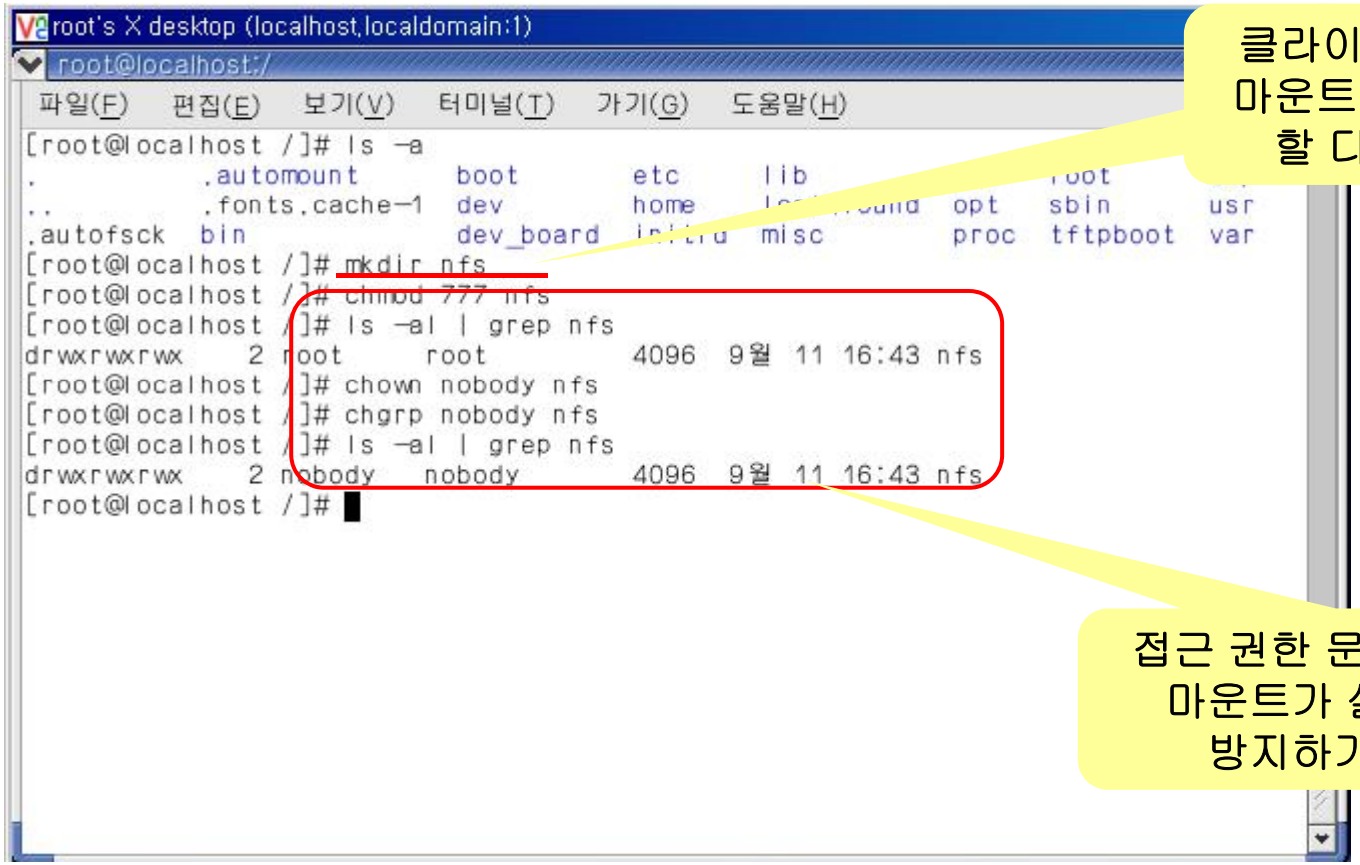
Step5: NFS



- Target board에서 Linux booting후
  - ✓ Device driver 및 application 개발

- 매번 tftp로 download?

Step5: NFS

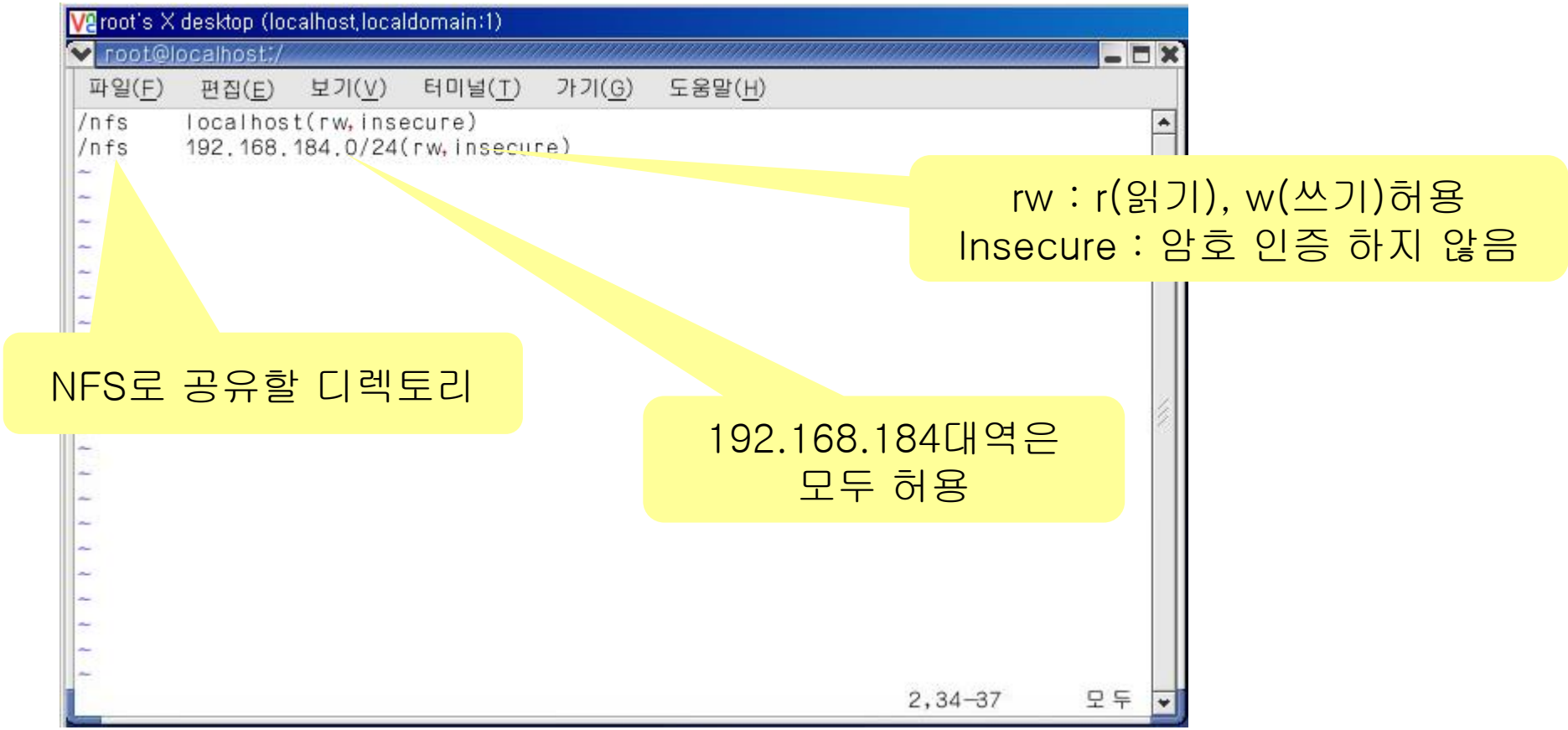


클라이언트로 부터의 마운트 요청시 서비스 할 디렉토리 생성

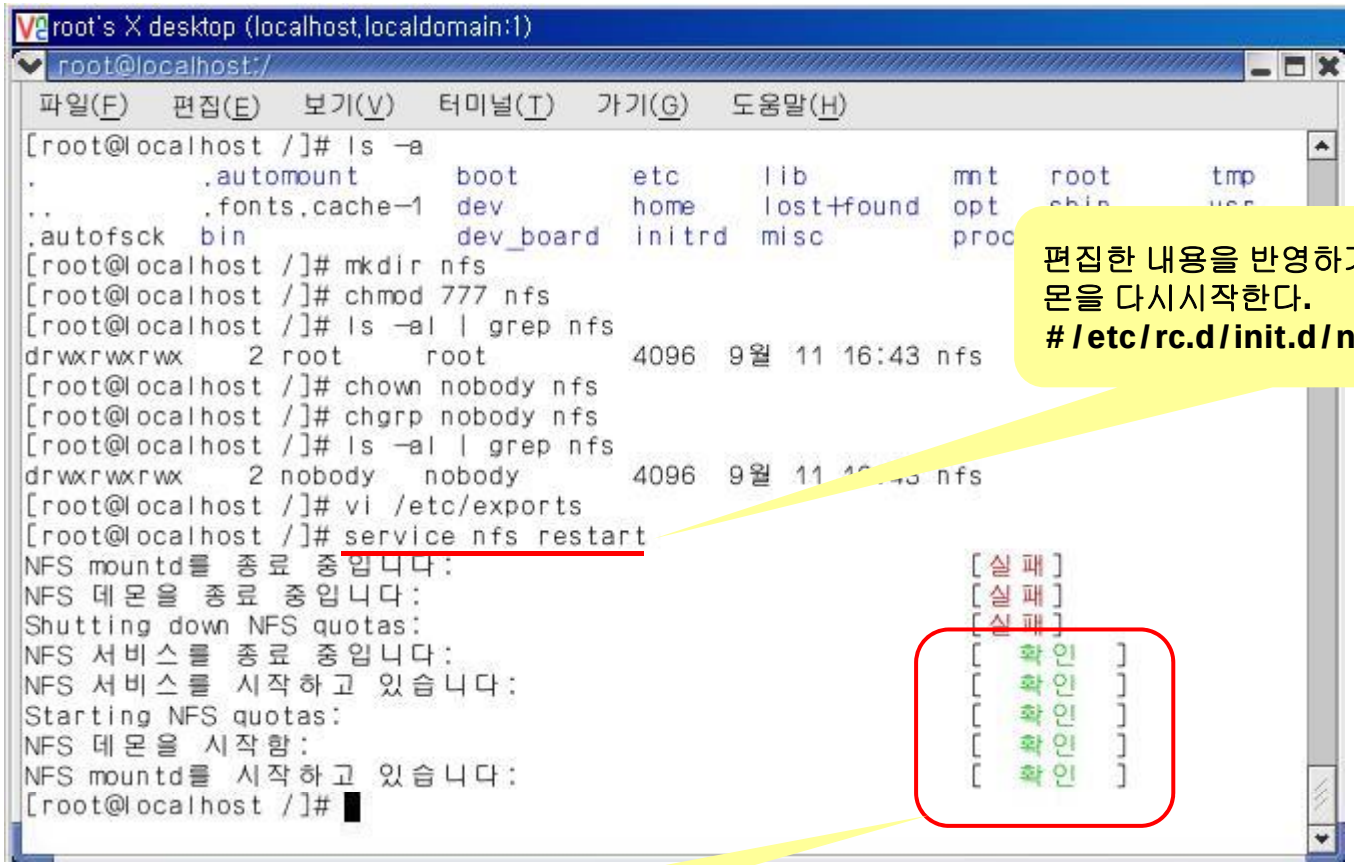
접근 권한 문제 등으로 인해 마운트가 실패하는 것을 방지하기 위한 설정

Step5: NFS

- NFS환경 파일 수정
  - ✓ /etc/exports 의 내용을 읽어 어떤 호스트에게 마운트를 허가하고, 어떤 제한을 둬야 하는지 판단



Step5: NFS

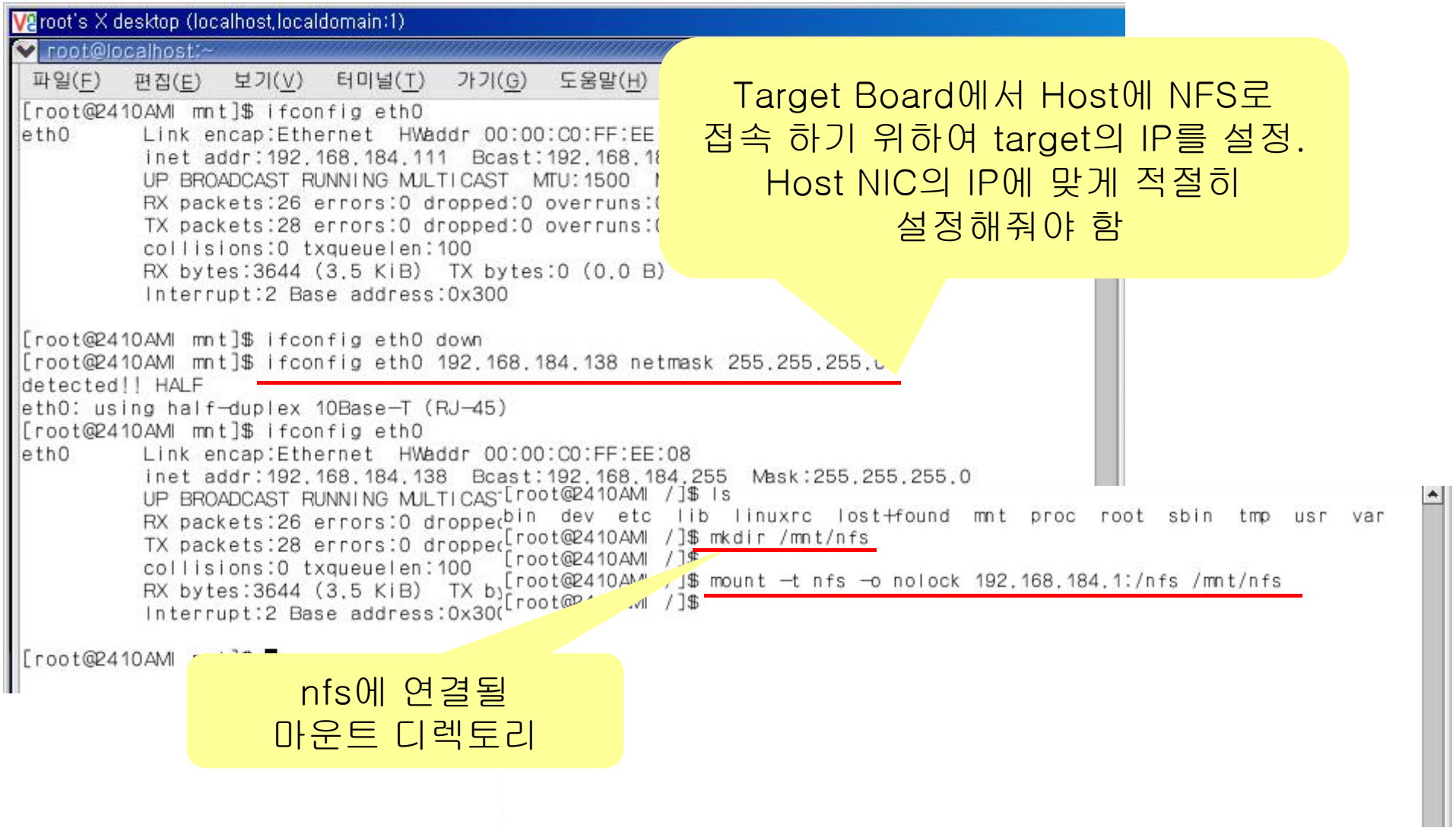


편집한 내용을 반영하기 위해서 **nfs** 데몬을 다시시작한다.
 # /etc/rc.d/init.d/nfs restart

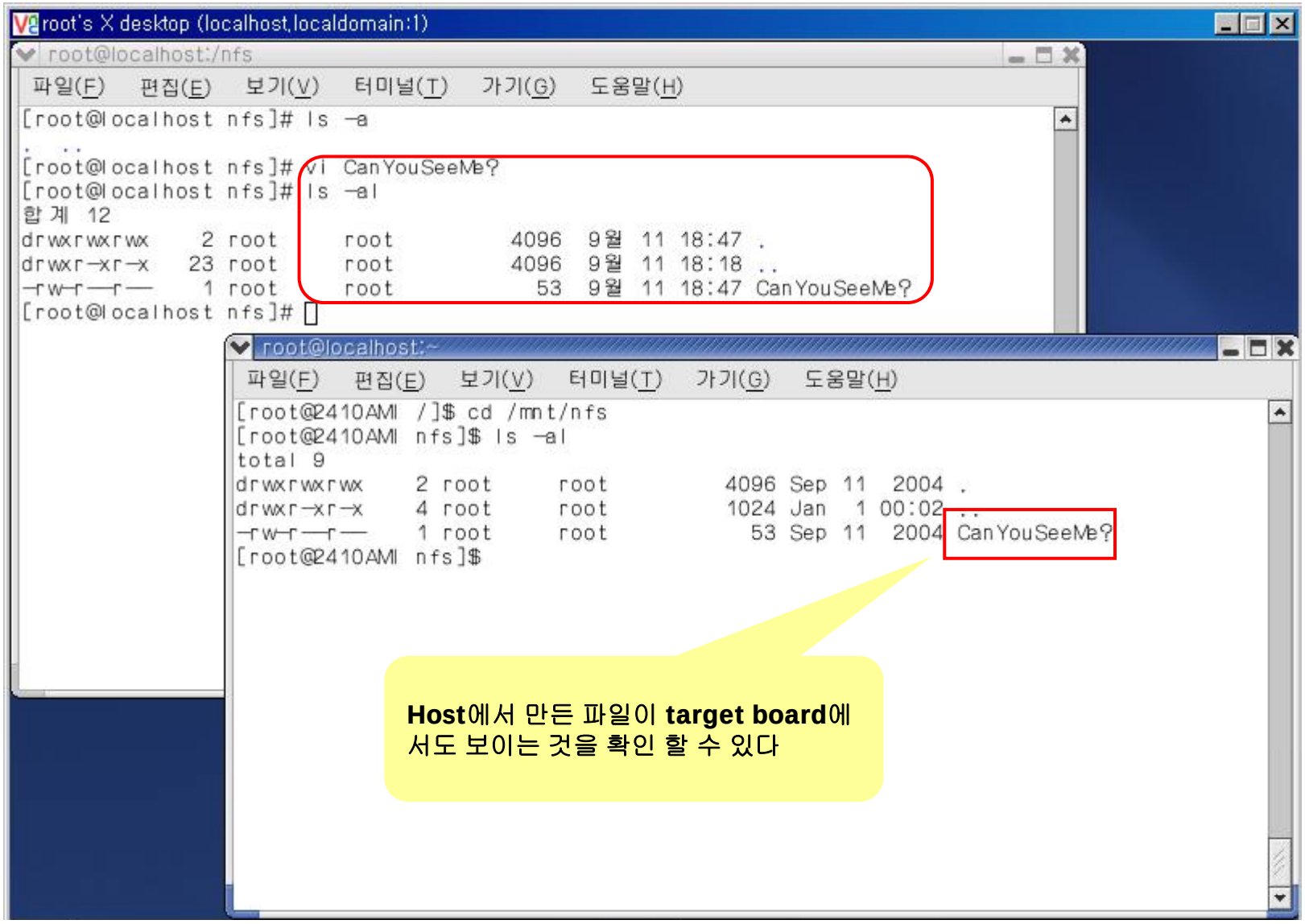
데몬이 잘 작동 되는지 확인

Step5: NFS

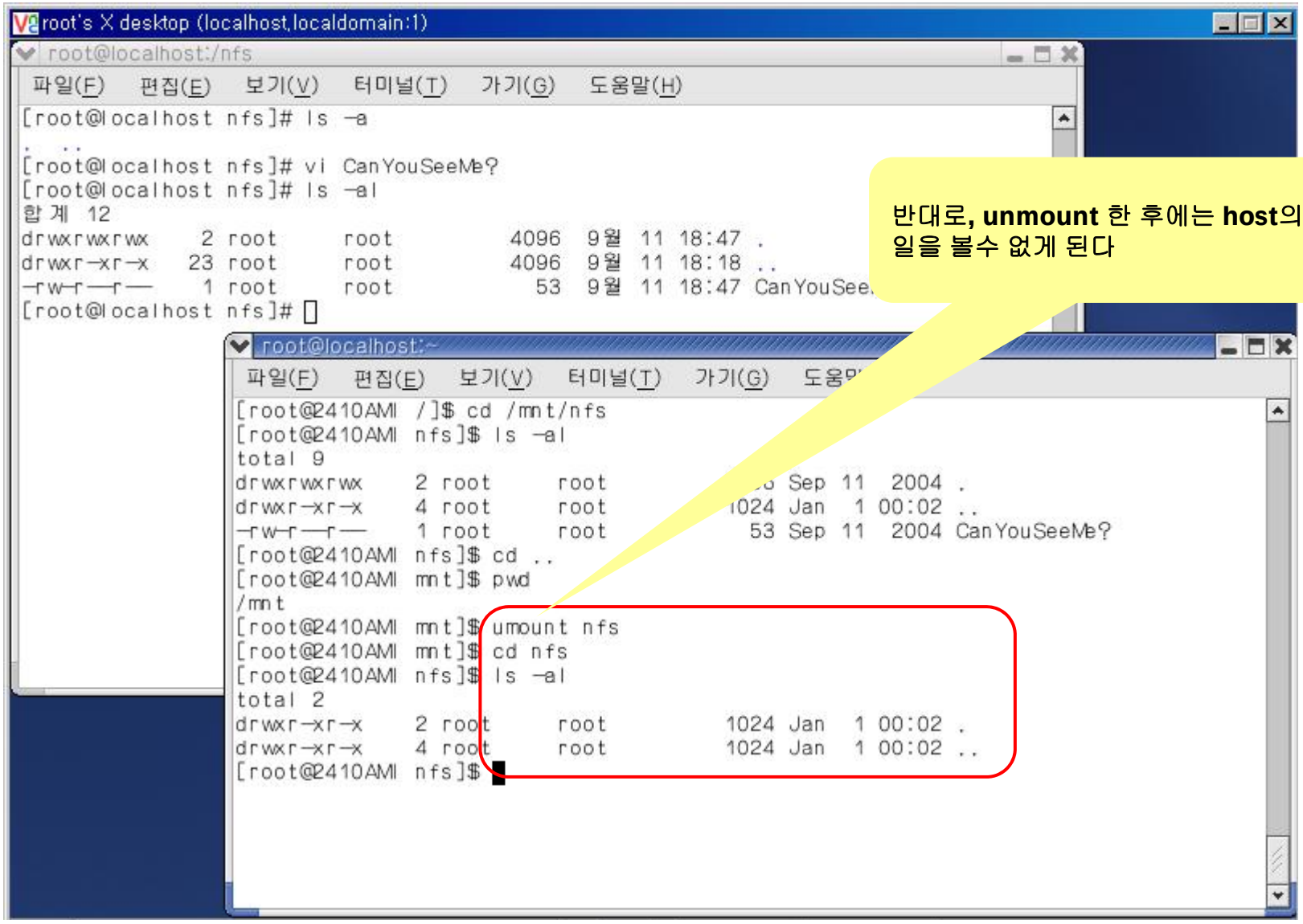
- Board에서 nfs서버 접속



Step5: NFS


root's X desktop (localhost.localdomain:1)
root@localhost:/nfs
파일(F) 편집(E) 보기(V) 터미널(T) 가기(G) 도움말(H)
[root@localhost nfs]# ls -a
. ..
[root@localhost nfs]# vi CanYouSeeMe?
[root@localhost nfs]# ls -al
합계 12
drwxrwxrwx 2 root root 4096 9월 11 18:47 .
drwxr-xr-x 23 root root 4096 9월 11 18:18 ..
-rw-r--r-- 1 root root 53 9월 11 18:47 CanYouSeeMe?
[root@localhost nfs]#
root@localhost:~
파일(F) 편집(E) 보기(V) 터미널(T) 가기(G) 도움말(H)
[root@2410AMI /]$ cd /mnt/nfs
[root@2410AMI nfs]$ ls -al
total 9
drwxrwxrwx 2 root root 4096 Sep 11 2004 .
drwxr-xr-x 4 root root 1024 Jan 1 00:02 ..
-rw-r--r-- 1 root root 53 Sep 11 2004 CanYouSeeMe?
[root@2410AMI nfs]$
Host에서 만든 파일이 target board에 서도 보이는 것을 확인 할 수 있다

Step5: NFS


root's X desktop (localhost.localdomain:1)
root@localhost:/nfs
파일(F) 편집(E) 보기(V) 터미널(T) 가기(G) 도움말(H)
[root@localhost nfs]# ls -a
. ..
[root@localhost nfs]# vi CanYouSeeMe?
[root@localhost nfs]# ls -al
합계 12
drwxrwxrwx 2 root root 4096 9월 11 18:47 .
drwxr-xr-x 23 root root 4096 9월 11 18:18 ..
-rw-r--r-- 1 root root 53 9월 11 18:47 CanYouSee.
[root@localhost nfs]#
root@localhost:~
파일(F) 편집(E) 보기(V) 터미널(T) 가기(G)
[root@2410AMI /]$ cd /mnt/nfs
[root@2410AMI nfs]$ ls -al
total 9
drwxrwxrwx 2 root root Sep 11 2004 .
drwxr-xr-x 4 root root 1024 Jan 1 00:02 ..
-rw-r--r-- 1 root root 53 Sep 11 2004 CanYouSeeMe?
[root@2410AMI nfs]$ cd ..
[root@2410AMI mnt]$ pwd
/mnt
[root@2410AMI mnt]$ umount nfs
[root@2410AMI mnt]$ cd nfs
[root@2410AMI nfs]$ ls -al
total 2
drwxr-xr-x 2 root root 1024 Jan 1 00:02 .
drwxr-xr-x 4 root root 1024 Jan 1 00:02 ..
[root@2410AMI nfs]$
반대로, unmount 한 후에는 host의 파일을 볼수 없게 된다

생각해 볼 사항

- Cross compiler가 없다면?
- 보드의 CD가 없다면?