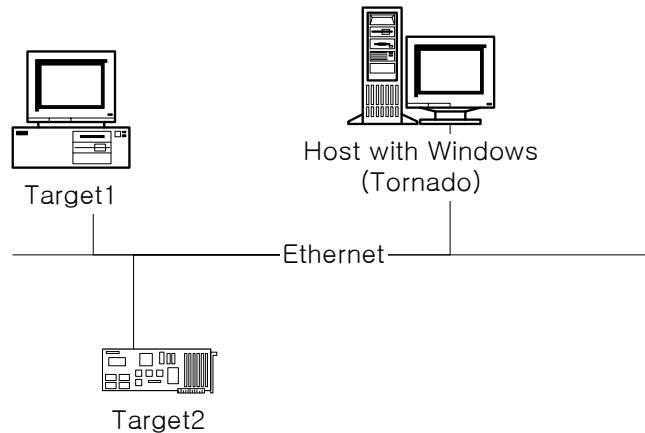


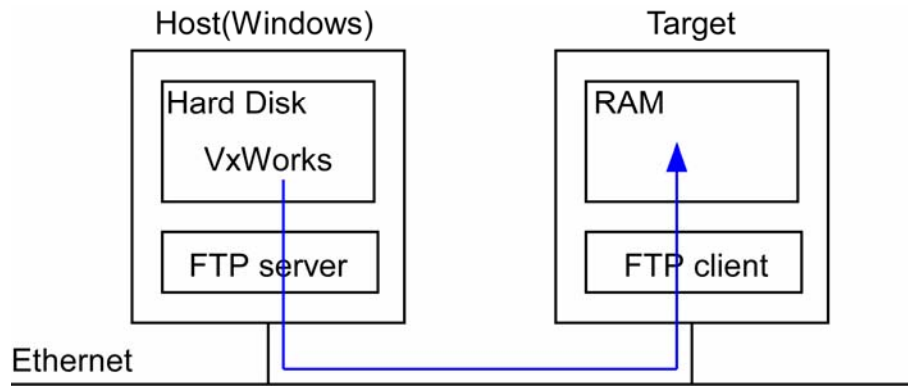
LAB0: Embedded real-time operating system VxWorks의 개발 환경

최근에 개발되고 있는 embedded system의 소프트웨어들은 네트워크 기능의 포함 등 그 기능이 날로 복잡해지고 있다. 따라서, 과거와 같이 프로그래머가 처음부터 끝까지 모든 기능을 다 구현하는 방법으로는 필요한 소프트웨어의 개발이 점차로 불가능해지고 있다. 이와 같은 문제를 해결하기 위하여, embedded operating system을 도입하여 소프트웨어를 개발하는 방법이 점차로 확산되고 있다. 이때, embedded system software를 개발 하려는 사람은 embedded operating system을 자신의 하드웨어에 porting 시킨 후, 그 위에 자신이 필요로 하는 응용 프로그램을 개발하는 방법이 채택되고 있다. 특히 이러한 embedded operating system의 경우는 일반 사무용 컴퓨터의 operating system과는 달리 여러 가지의 입력 신호에 대해서 실시간(real-time)으로 응답을 할 수 있어야 하므로 이러한 시스템 소프트웨어를 real-time embedded operating system이라고 한다. WindRiver사에서 개발된 VxWorks는 이러한 real-time embedded system software의 한가지 종류로서 현재 세계적으로 가장 큰 시장 점유율을 확보하고 있는 소프트웨어이다.(www.windriver.com 혹은 www.windriver.co.kr 참조) 본 실험은 VxWorks의 개발 환경인 Tornado의 사용법을 익히고, 또한 이를 이용하여 실시간 처리가 필요한 응용 소프트웨어의 개발 방법을 익힌다.

VxWorks의 개발 환경은 다음 그림과 같다.



위의 그림에서 개발용 소프트웨어인 Tornado가 내장된 컴퓨터를 host, 개발 대상인 하드웨어를 target이라고 한다. 일반적으로, embedded system 개발 단계에서는 개발자가 개발한 고유의 하드웨어가 target 하드웨어가 되나, 여기에서는 편의상 Pentium PC를 target으로 사용한다. 최종적으로 프로그램 개발이 완료 되면, 개발된 응용 프로그램은 VxWorks kernel 과 함께 ROM에 내장되어서 embedded hardware board에 내장이 되나, 응용 프로그램을 개발하는 과정에는 수시로 프로그램의 변경과 테스트가 반복이 되어야 하므로, VxWorks kernel 과 응용 프로그램은 LAN을 통하여 host로부터 target으로 down load 되어 시험을 실시한다. Host PC의 하드디스크에 있는 VxWorks kernel과 응용 프로그램은 FTP를 이용하여 target의 메모리로 down load 하며, 이를 위해서 host에는 FTP server가 동작하고 있어야 한다. Windows의 시작을 booting이라고 하듯이 VxWorks kernel을 시작하는 것도 booting 이라고 부른다. 아래의 그림은 VxWorks의 booting 개념을 그림으로 나타낸 것이다.



VxWorks의 booting을 위해서 필요한 작업은 아래와 같다.

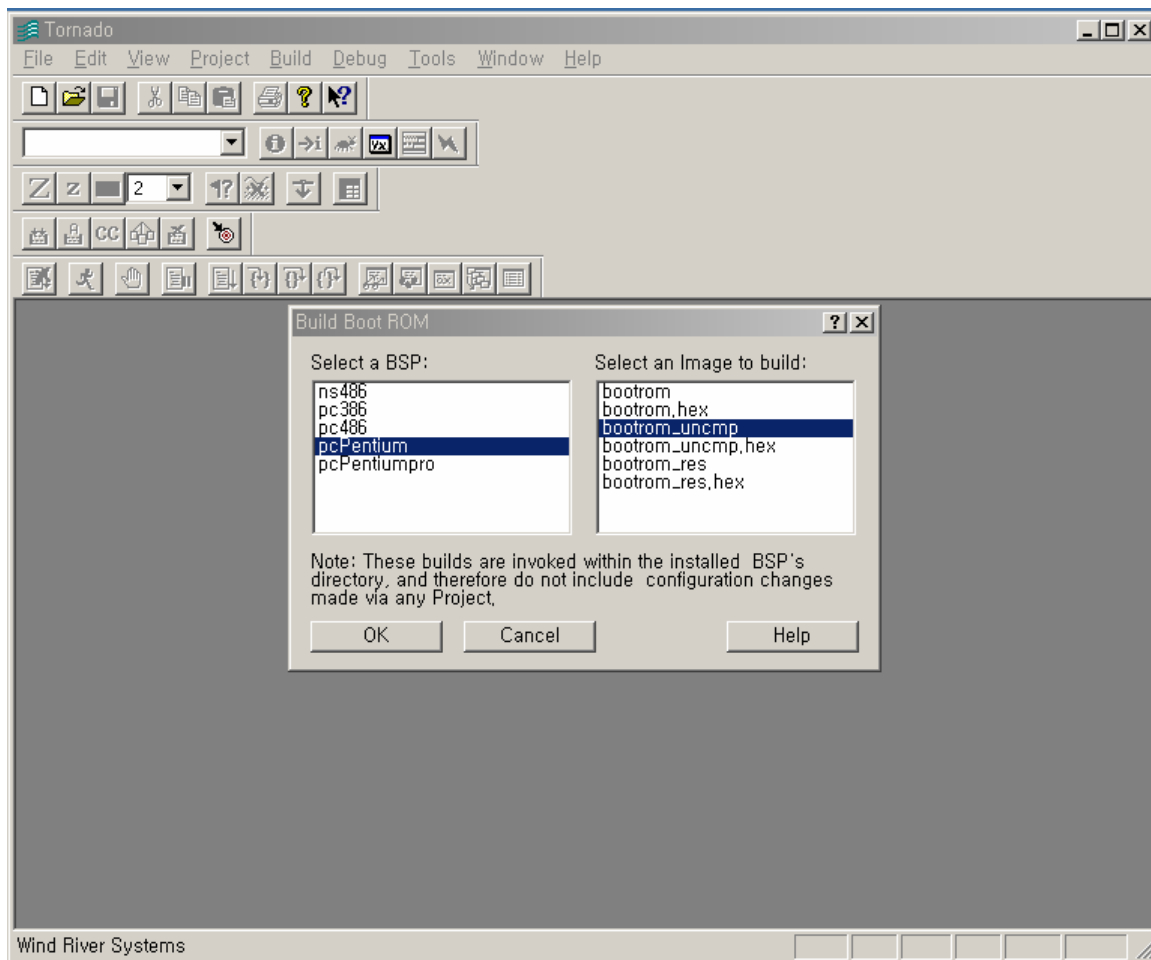
- 1) Boot diskette 만들기
- 2) VxWorks kernel의 build 과정
- 3) Host computer에서 FTP server 실행하기
- 4) VxWorks의 booting

Booting을 위한 위의 4가지 작업과 응용프로그램의 실행 방법은 아래에 상세히 설명한다.

1) Boot diskette 만들기

Target hardware가 host로부터 VxWorks file과 응용 프로그램을 down load 할 수 있기 위해서 target hardware에는 기본적인 boot program 이 내장 되어 있어야 한다. 일반 target hardware에서는 이러한 boot program이 ROM에 내장이 되나, Pentium PC가 target으로 사용 될 때는 floppy disk가 boot ROM의 기능을 수행하게 된다.

PC용 boot diskette를 만드는 과정은 다음과 같다. Tornado를 실행 시킨 후 Build menu에서 Build Boot ROM을 선택하면, 다음 그림과 같은 화면이 나오며 이때 OK를 클릭하면 boot rom을 만드는 과정이 실행이 되고 C:\Tornado\target\config\pcPentium에 bootrom_uncmp라는 파일이 새로 생성이 된다.



다음으로 a: diskette drive에 diskette 을 넣고 DOS창에서 다음과 같이 실행한다.

```
C:\W>cd C:\Tornado\target\config\pcPentium
```

```
C:\Tornado\target\config\pcPentium>mkboot a: bootrom_uncmp
```

위의 과정이 끝나면 boot diskette이 만들어진 것이다.

Booting을 하기 위해서는 여러가지의 boot parameter들을 설정해야 하는데, 이 설정 값들은

C:\Tornado\target\config\pcPentium 의 config.h 에서 수정하도록 되어있다. 가장 중요한 설정값으로 host와 target의 IP 주소인데, 이는 config.h의 다음 부분에서 설정하도록 되어 있다.

```
#if (CPU == I80386)
#define DEFAULT_BOOT_LINE W
    "fd=0,0(0,0)host:/fd0/vxWorks.st h=90.0.0.3 e=90.0.0.50 u=target"
#elif (CPU == I80486)
#define DEFAULT_BOOT_LINE W
    "fd=0,0(0,0)host:/fd0/vxWorks.st h=90.0.0.3 e=90.0.0.50 u=target"
#elif (CPU_VARIANT == PENTIUM)
#define DEFAULT_BOOT_LINE W
    "elPci(0,0)host:vxWorks h=166.104.225.252 e=166.104.225.251 u=target
pw=target"
#elif (CPU_VARIANT == PENTIUMPRO)
#define DEFAULT_BOOT_LINE W
    "fd=0,0(0,0)host:/fd0/vxWorks.st h=90.0.0.3 e=90.0.0.50 u=target"
#endif /* (CPU == I80386) */
```

위에서, 현재 target인 PENTIUM 의 경우를 보면, down load하기 위한 kernel file 명은 vxWorks 이며, host IP 주소는 166.104.225.252 이고, target IP 주소는 166.104.225.251 이며, host의 ftp 에 login 하기 위한 id와 password는 모두 target으로 설정이 되어 있다. Config.h 파일의 수정시 반드시 확인해야 할 부분은 사용된 Ethernet board에 대한 드라이버 설정 부분이다. 현재 사용된 Ethernet board는 3COM의 3C900 model 이므로 다음과 같이 되어 있는지 확인하고, 그렇지 않다면 수정한다.

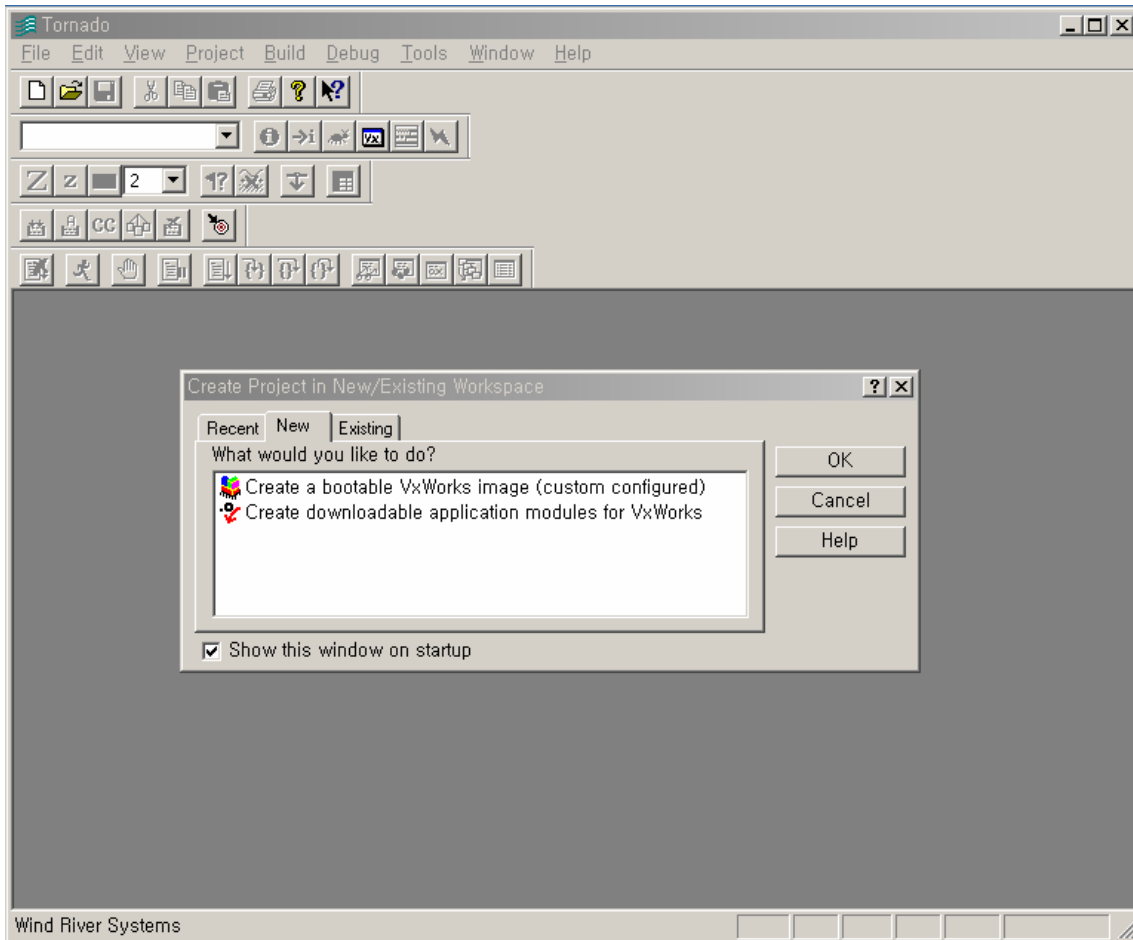
```
/* Network driver options */
#define INCLUDE_EL_3C90X_END /* 3com fast etherLink XL PCI */
```

만약 config.h의 파일을 수정했다면, 위에서 설명한 boot diskette 만드는 과정을 처음부터 다시 수행해야만 수정한 것이 반영될 수 있다.

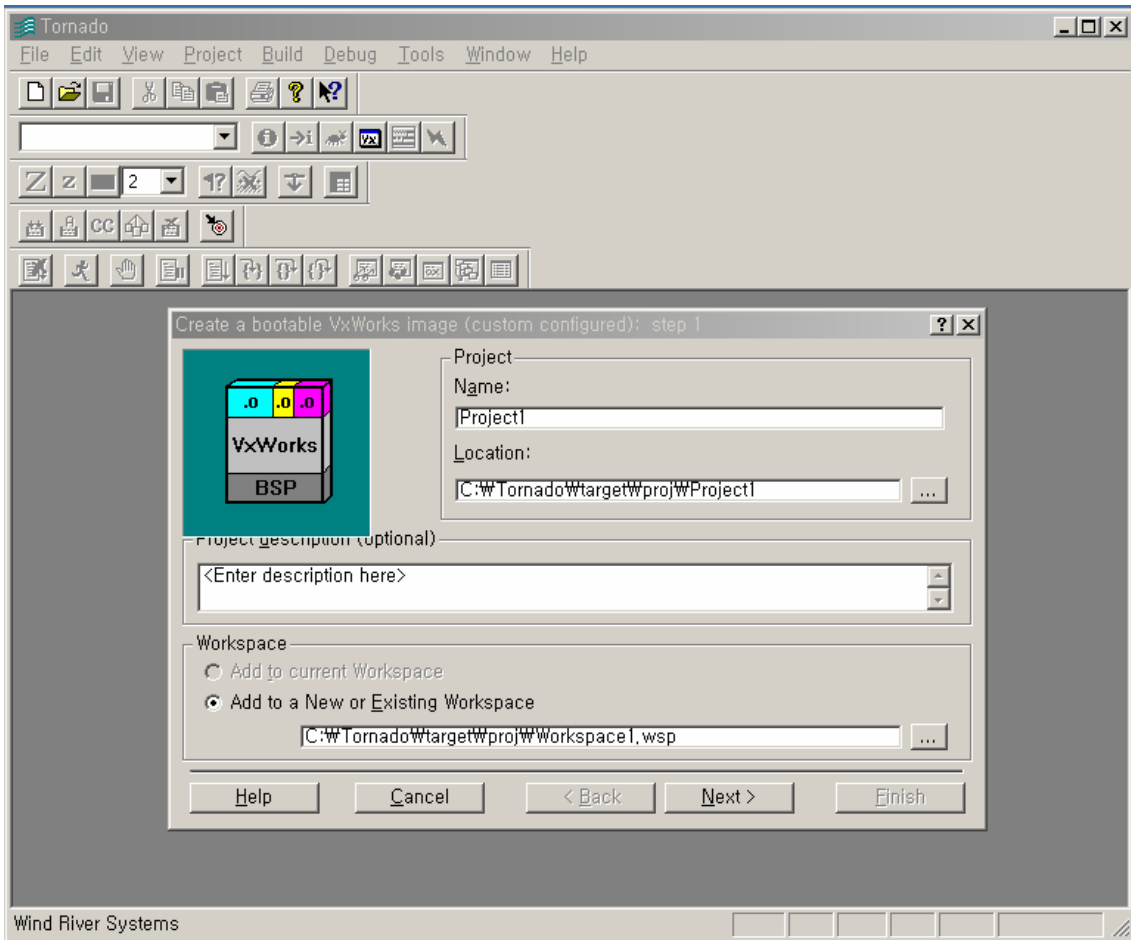
Boot parameter의 변경은 config.h 를 수정하지 않고도, booting 시 키보드에서 입력하여 변경 할 수도 있다. 그러나, 키보드에서 변경하는 것은 매번 booting 할때마다 입력 해야 하므로, 이러한 번거로움을 피하기 위해서는 config.h 에서 변경하는 것이 편리하다.

2) VxWorks kernel의 build 과정

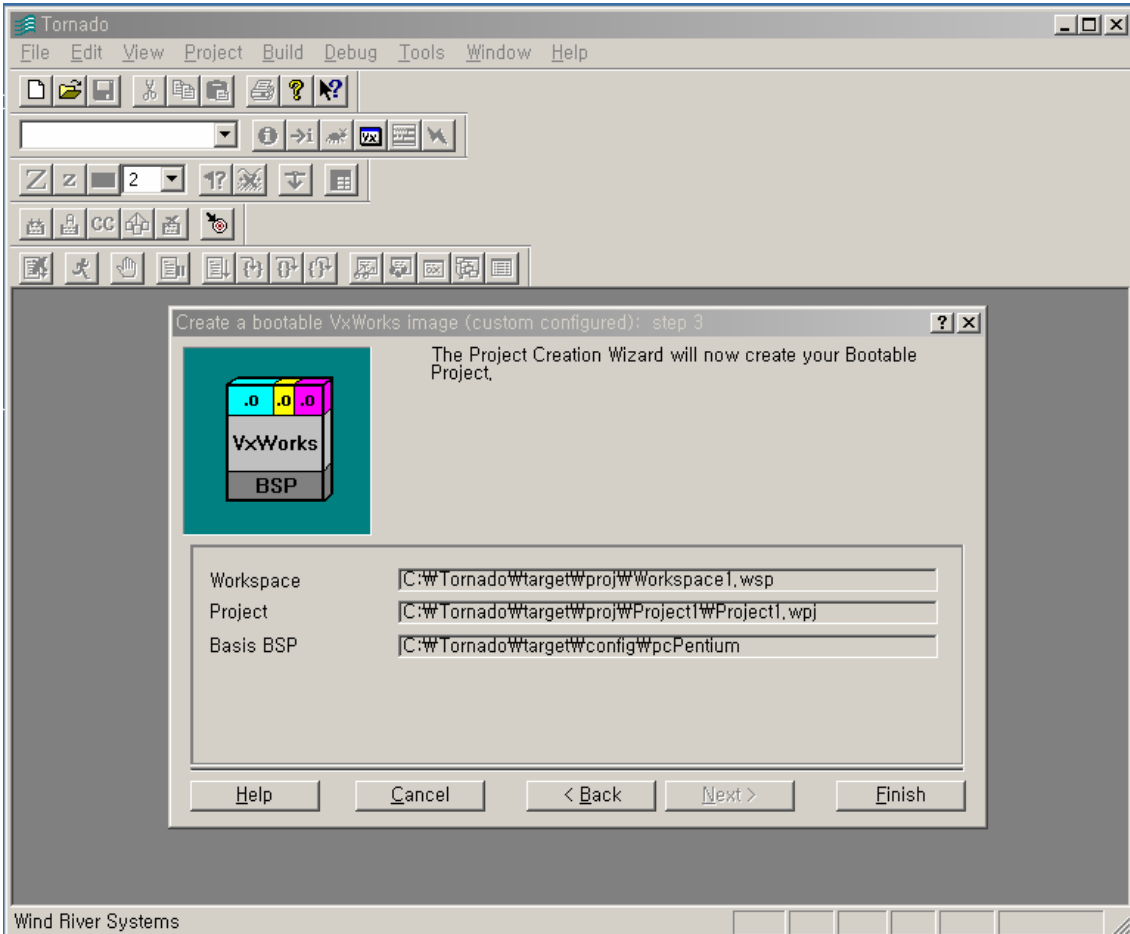
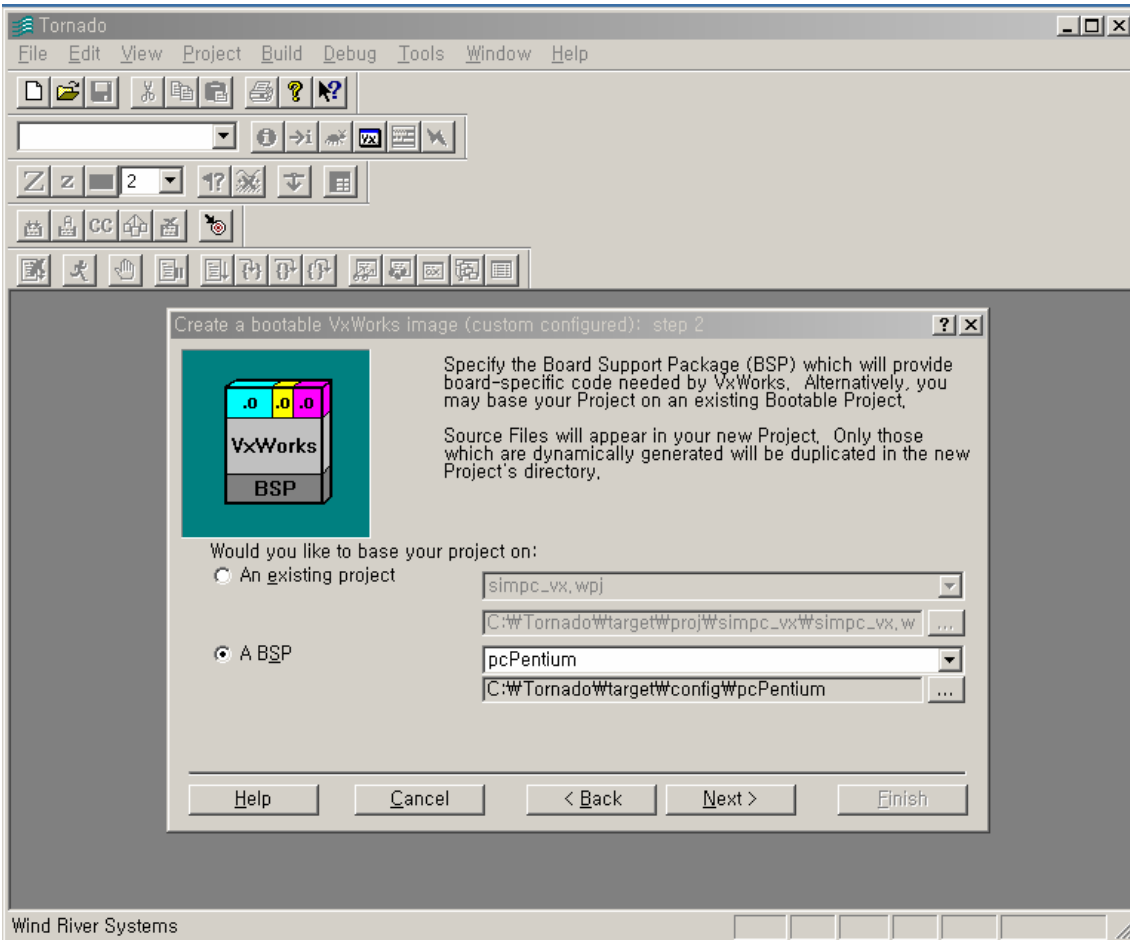
Tornado에서 VxWorks를 build 하기 위해서 다음과 같은 과정을 수행한다. Tornado의 file menu에서 New Project를 선택하면 다음과 같은 화면이 나타난다.



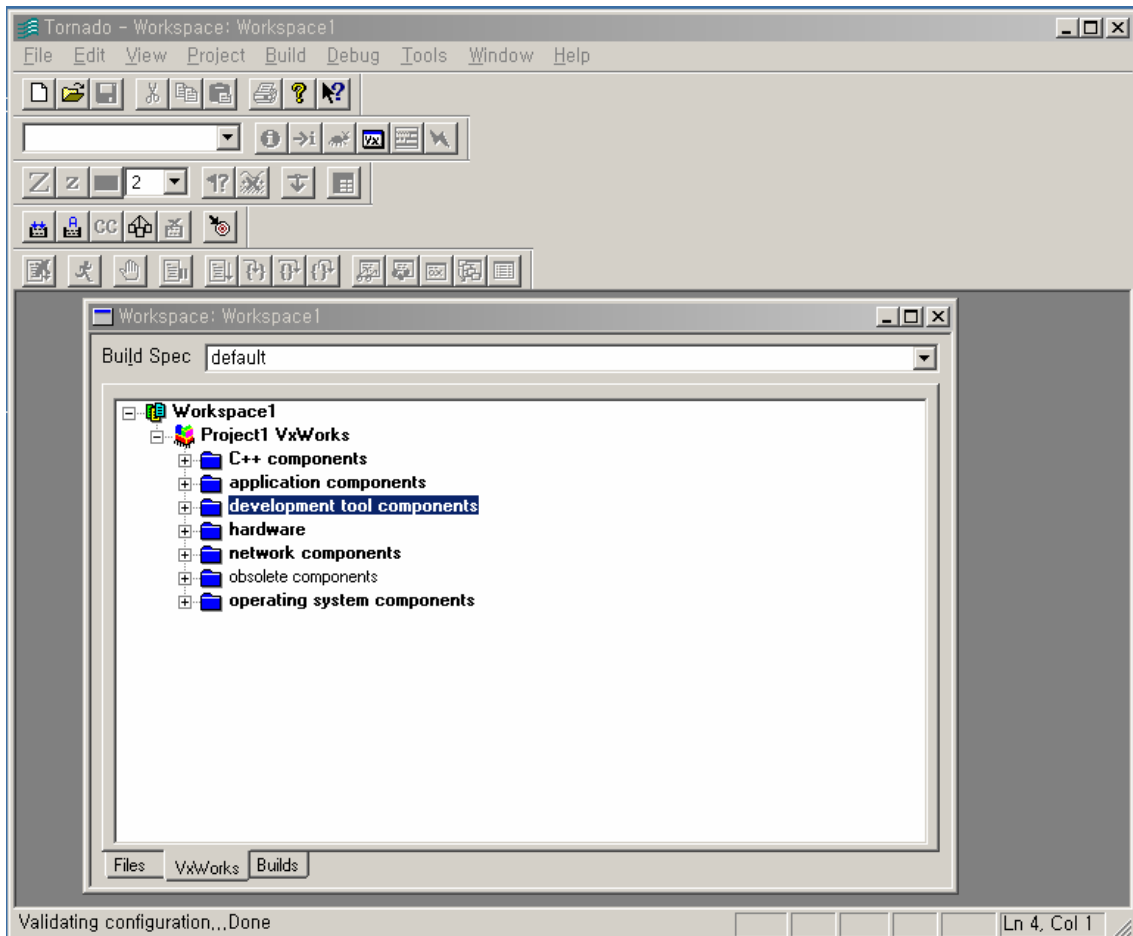
위의 화면에서 Create a bootable VxWorks image를 선택한후 OK를 클릭하면 다음 화면이 나타난다.



위 화면의 경우에는 project 이름이 Project1으로 되어 있다. Project의 이름은 Tornado가 자동으로 정해주거나 사용자가 임의로 정할 수 있으므로, 위의 화면과 똑같이 Project1이 아니라고 해도 굳이 변경할 필요는 없다. 이 위의 화면에서 Next를 누르면 연달아 다음의 화면들이 나타나며, 다음 그림들에 나와 있는 대로 선택을 하며 진행을 해 나간다. 아래의 화면에서 BSP는 board support package의 약자이며, 현재 사용하려는 하드웨어를 나타낸다. 이 실험에서 사용하는 하드웨어는 Pentium이므로 pcPentium이 선택 되도록한다.



위의 화면이 나왔을 때 Finish를 누르면 필요한 파일들이 자동으로 생성되는 과정이 진행되고, 그 과정이 다 끝나면 Workspace 화면이 나온다. 이 화면에서 VxWorks 탭을 선택하면 아래의 그림과 같은 화면을 볼 수 있다.



위의 화면에서 Build를 수행하면, C:\Tornado\target\proj\Project1\default의 directory에는 vxWorks라는 파일이 만들어 지며, 이 파일이 target으로 다운 로드하기 위한 kernel file이다. 위의 directory 이름에서 Project1은 현재 수행중인 project의 이름이므로, project마다 달라질 수 있다. 위의 상태에서 build를 수행하면 기본적인 소프트웨어 컴포넌트만을 포함하여 build가 수행된다. Build를 수행하기 전에 소프트웨어 컴포넌트들을 빼거나 넣는 것이 가능한데, 본 실험에서 추가 해야 할 컴포넌트는 다음과 같다.

built-in symbol table

shell banner

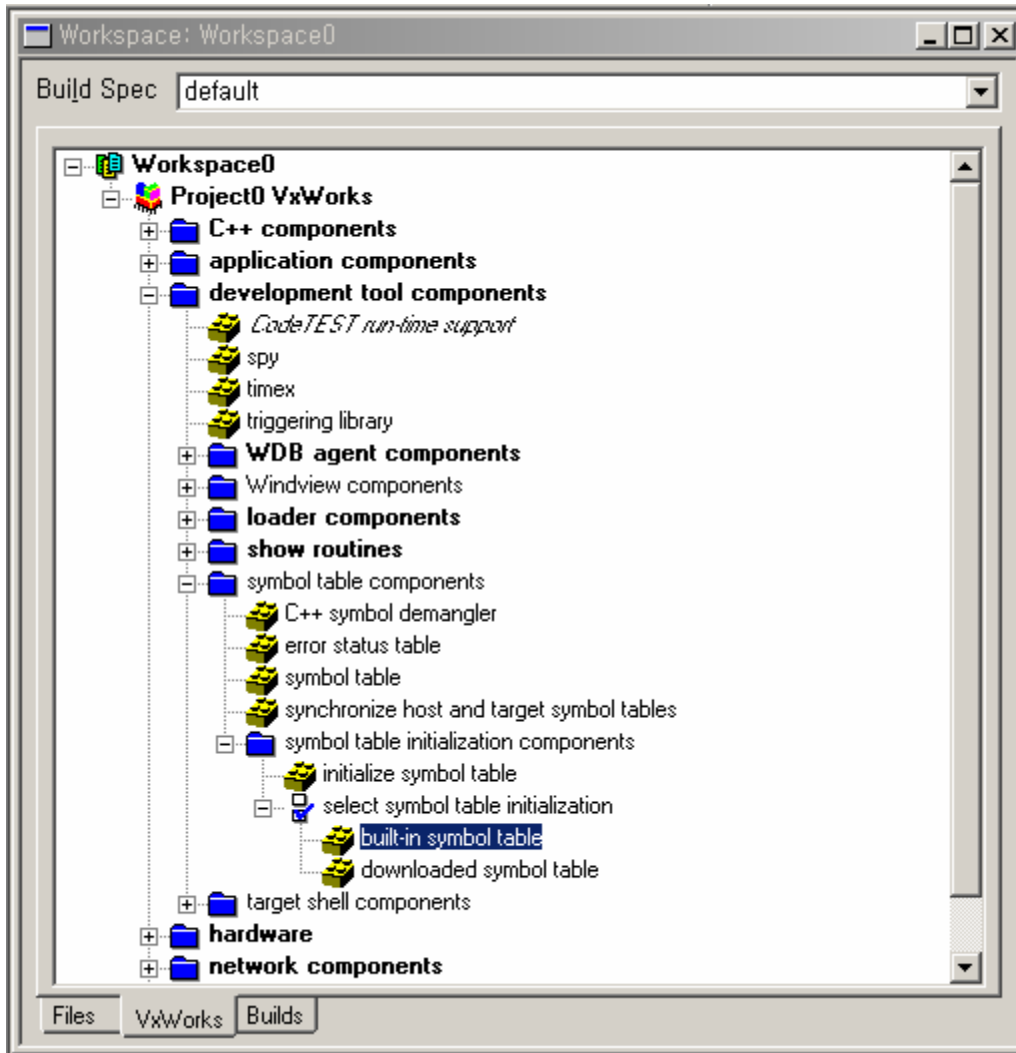
target shell

TELNET server

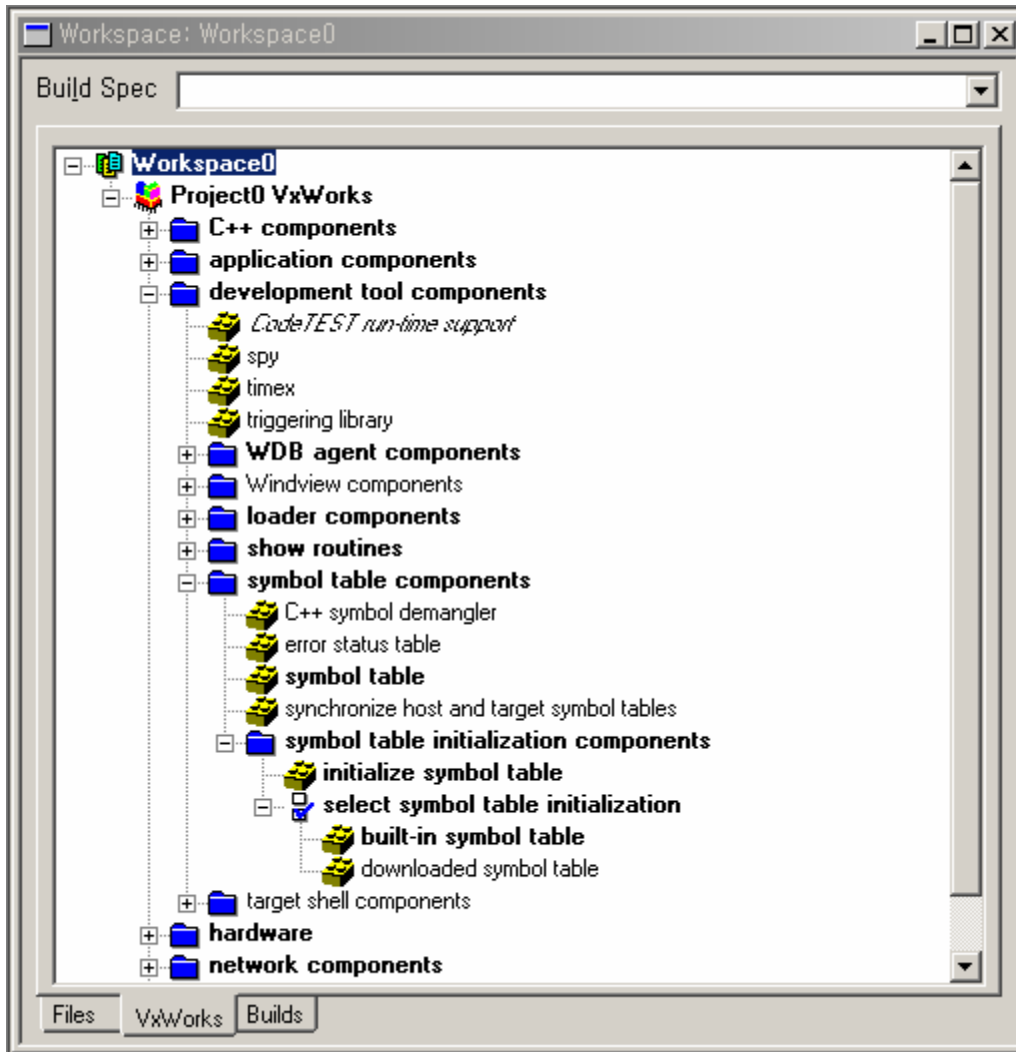
이 소프트웨어 컴포넌트들의 추가 방법은 아래에 상세히 설명하기로 한다.

먼저, built-in symbol table의 추가는 다음과 같이 한다. 아래의 그림과 같이 built-in symbol table을 찾아서 선택한후, **오른쪽 마우스 버튼**을 누르면 몇가지 menu가 나오는데, 이 menu

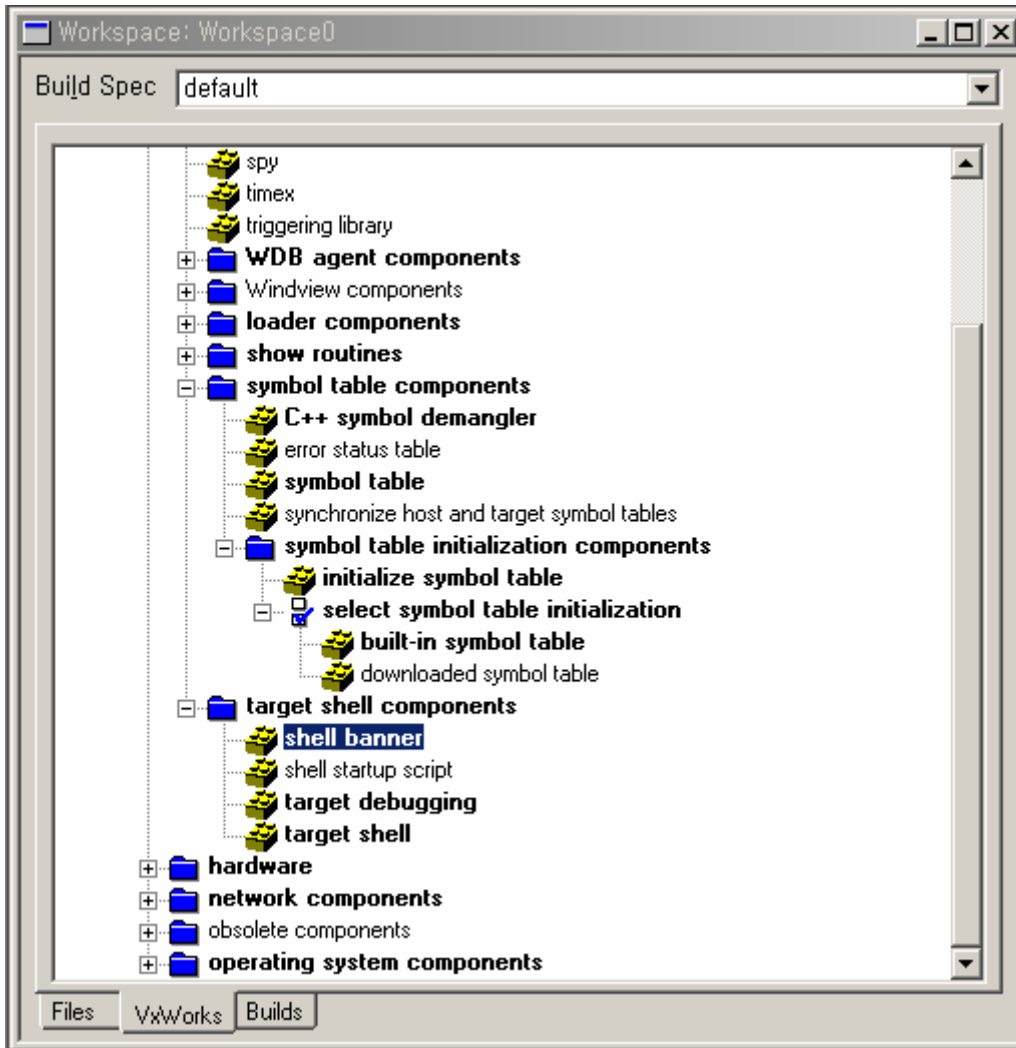
중 include 'built-in symbol table'을 선택하면, 확인창이 나오고 OK를 누르면 built-in symbol table component의 선택이 끝난다.



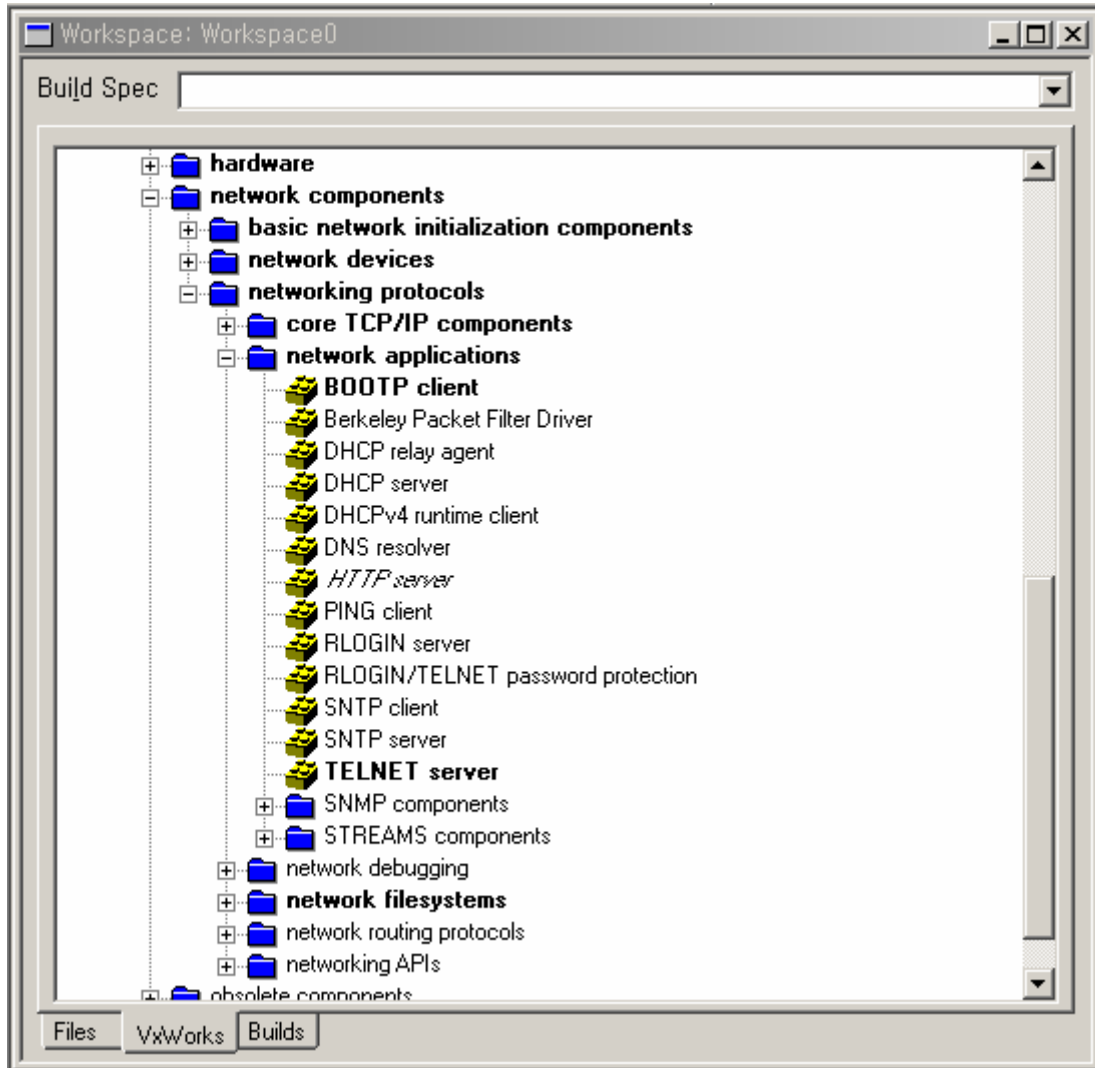
정상적인 선택이 이루어 졌다면, 아래의 그림과 같이 선택된 요소들이 bold 체로 바뀐 것을 확인할 수 있다. 아래와 같이 built-in symbol table이 bold체로 바뀌지 않았다면 정상적인 선택이 이루어지지 않았음을 의미한다.



target shell component에 대해서도 아래 그림과 같이 shell banner와 target shell이 선택되도록 한다. 이때, shell banner를 먼저 선택하면 target shell의 선택은 자동으로 이루어 짐을 알 수 있다. Shell banner는 target shell이 먼저 선택 되어야 선택 될 수 있는 컴포넌트이므로, shell banner의 선택시 target shell이 선택되어 있지 않다면 Tornado는 이의 선택을 자동으로 해준다.



TELNET server 컴포넌트에 대해서도 아래의 그림과 같이 선택을 한다.



각 소프트웨어 컴포넌트의 선택이 끝난후에는 메뉴 툴바로부터 Build를 선택하여 수행한다.

3) Host computer에서 FTP server 실행하기

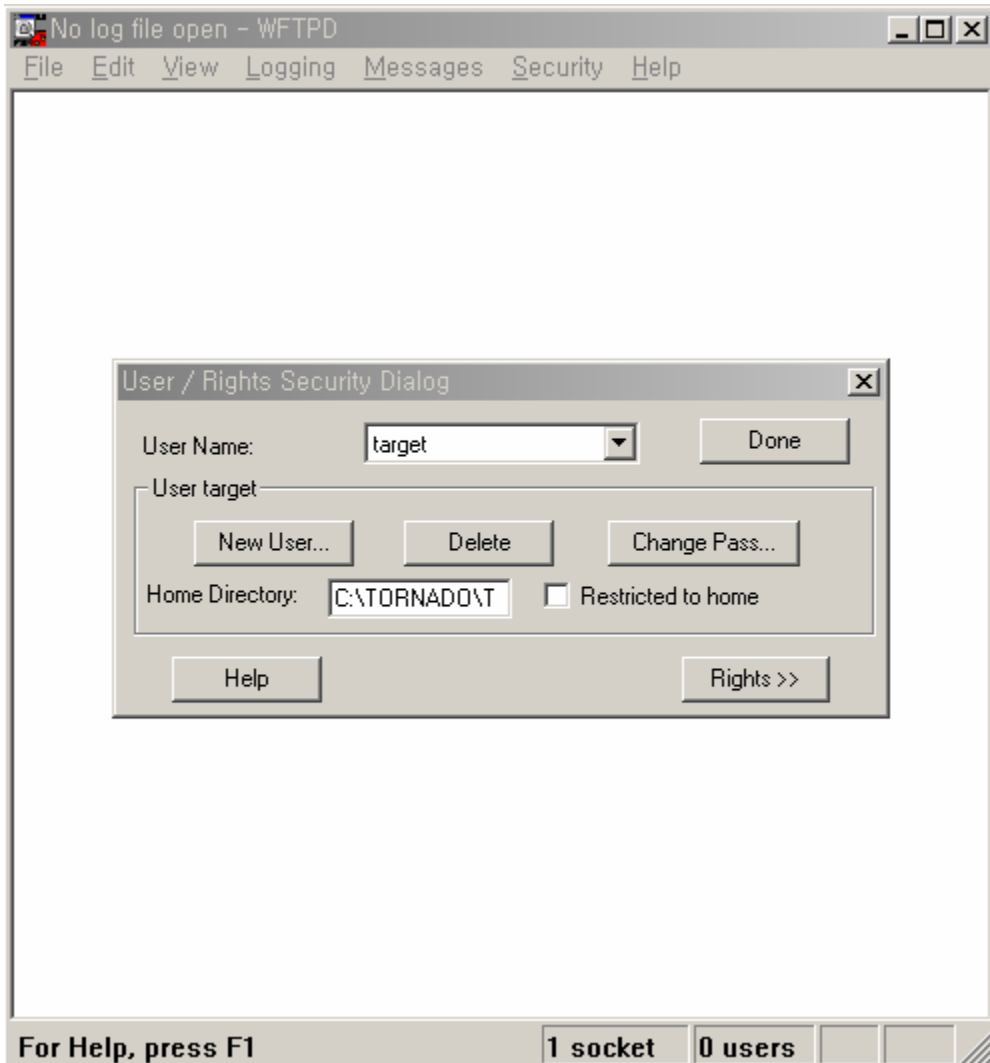
FTP server program은 프로그램 그룹의 Tornado2 Menu 안에 포함이 되어 있다. 앞의 VxWorks build과정을 통해서 만들어진 VxWorks 파일이 FTP 를 통해서 다운로드 되기 위해서는 FTP server program에

user id: target

password:target

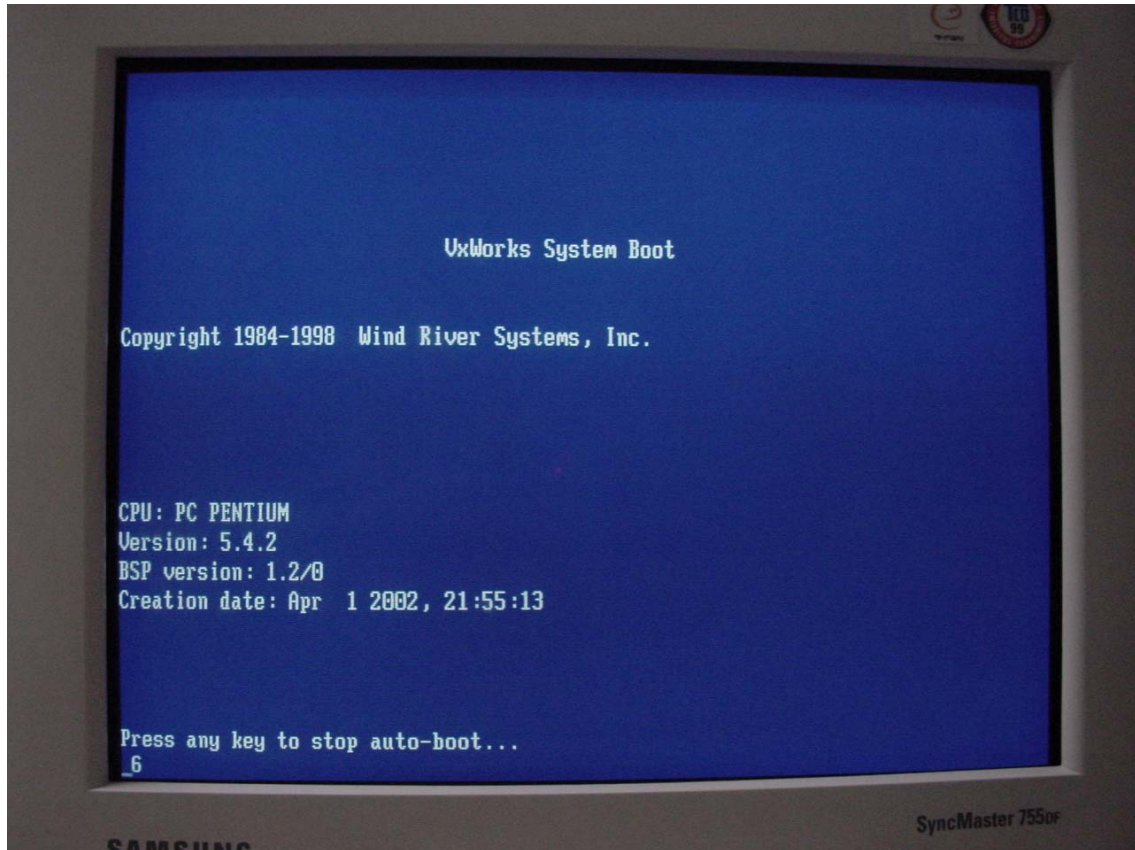
home directory: C:\Tornado\target\proj\Project1\default

와 같은 계정이 만들어져 있어야 한다. 위의 target user의 home directory는 VxWorks file이 있는 directory임에 유의해야 한다. 즉, Tornado에서 VxWorks를 build할때에 사용되었던 project 이름에 따라서 이 home directory는 변경되어야 함을 특별히 주의한다. FTP 계정의 home directory는 다음 그림과 같이 확인 할 수 있다. 만약 home directory의 이름이 VxWorks가 있는 directory와 다르다면 변경하도록 한다.



4) VxWorks의 booting

Host computer에서 FTP server를 실행시킨 상태에서, boot diskette을 target PC에 넣고 PC를 boot 시키면 다음 그림과 같은 화면이 나오며, booting이 실행이 된다. (Booting이 원만히 진행되지 않을 경우, diskette을 format 하고 boot diskette를 다시 만들어 위의 과정을 수행한다.)



Auto boot 를 실행하기 위한 count down을 중지하려면 아무 키나 누르면 중지가 되고, Boot ROM prompt상에서 다음과 같은 몇가지의 명령어를 사용할 수 있다.

H: Help 명령어이다. 사용가능한 명령어들에 대한 설명을 보여준다.

P : Boot parameter를 화면에 보여준다. IP 주소등이 정확히 설정되었는지를 확인하기 위해서는 p 명령어를 치면된다.

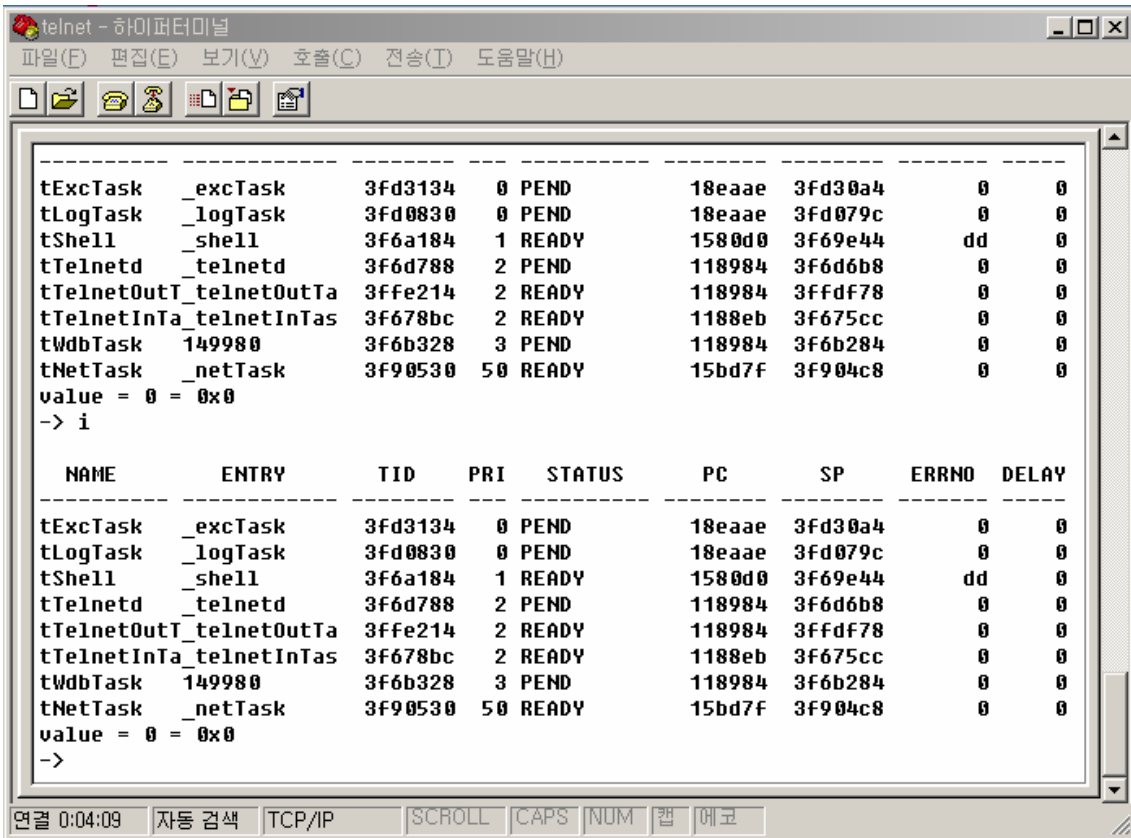
C : Boot parameter를 변경할 수 있는 명령어이다.

@ : Boot 명령어이다. 즉, auto boot가 정지 된 상태에서 이 명령어는 booting을 다시 시작하게 해준다.

이상의 작업들을 다 수행한 후 booting이 원만히 수행이 되었다면 target의 모니터 상에 다음과 같은 화면이 나와야 한다.



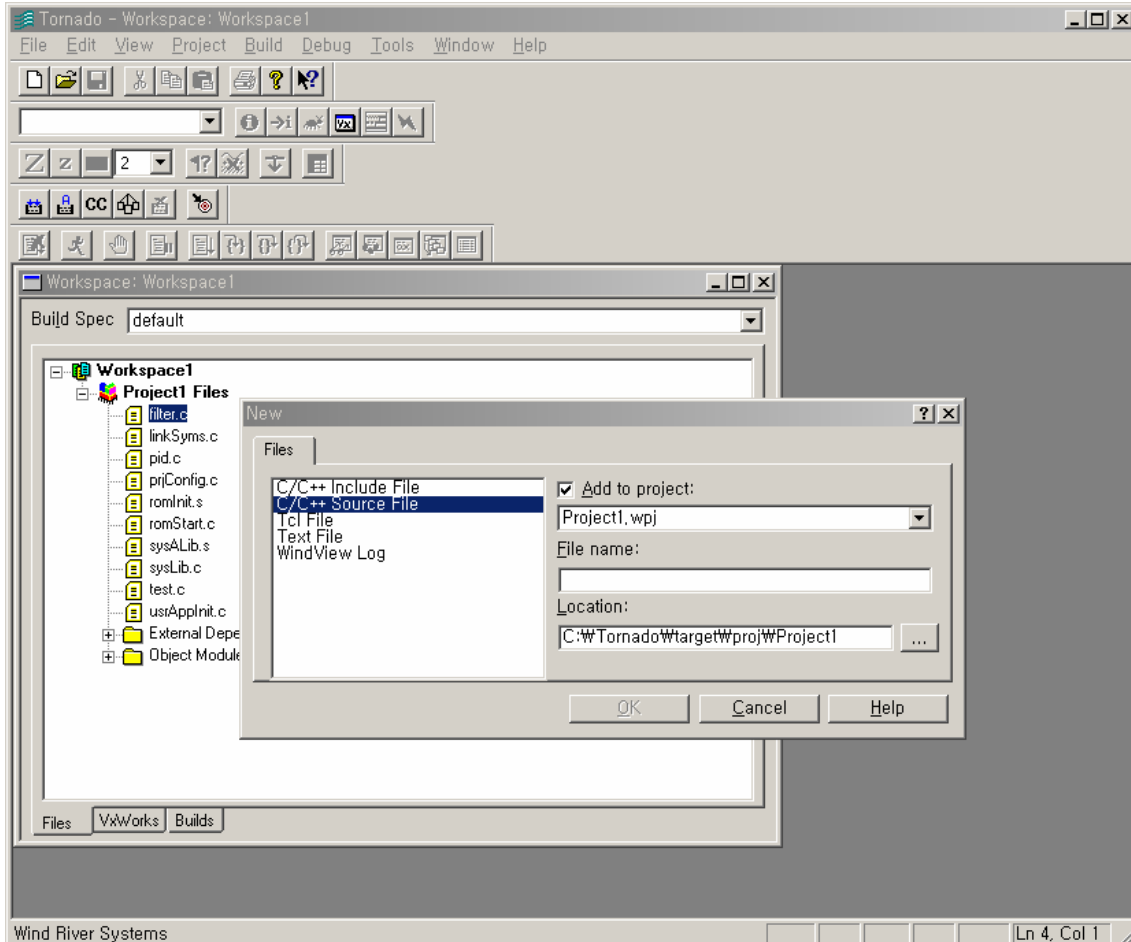
이와 같이 booting 된 target의 vxWorks에는 TELNET server가 있으므로, host에서 윈도우 상에서 수행 가능한 TELNET client program(하이퍼터미널, 새롬데이터맨 등)을 수행 한 후 target의 IP address로 접속을 하면 다음 그림과 같이 target 의 shell 로 들어갈 수 있다.



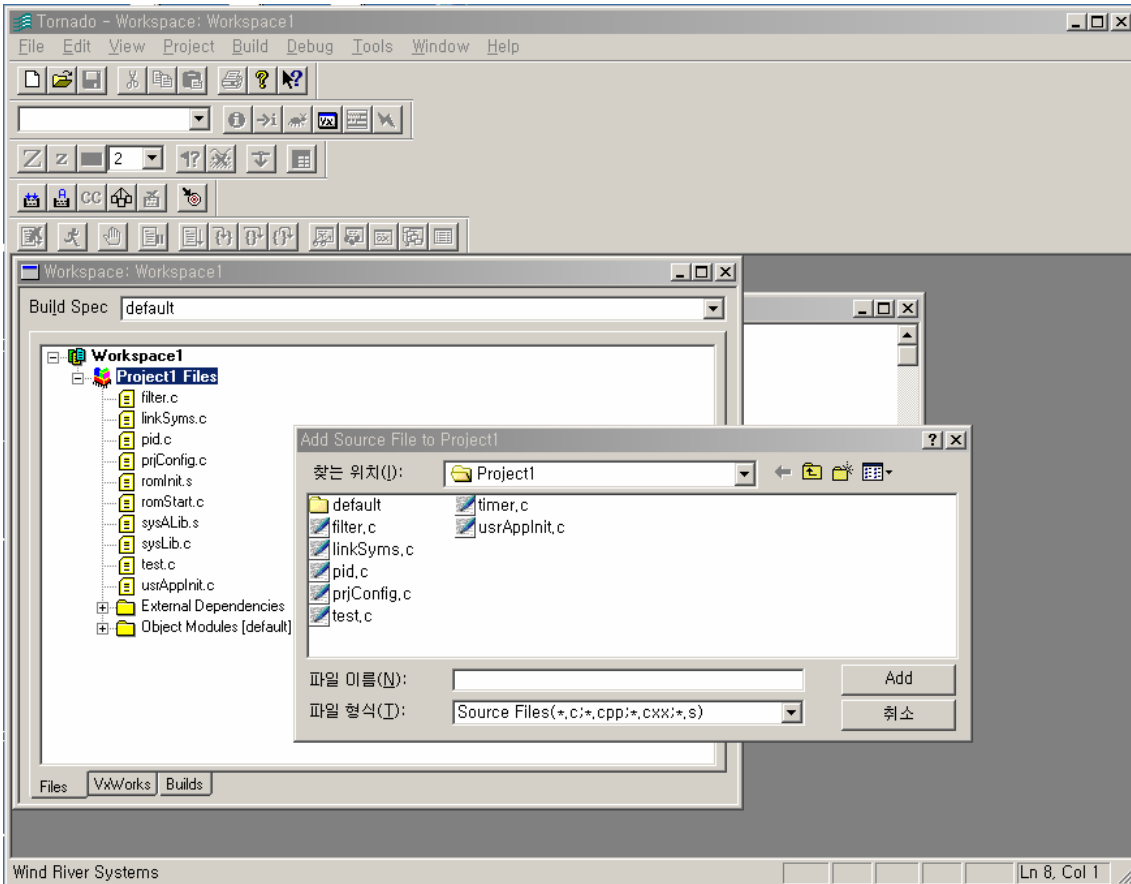
위의 그림에서 i 명령은 현재 수행중인 task 들의 상태를 보여주는 shell 명령어 이다.

5) User program 작성 및 수행 방법

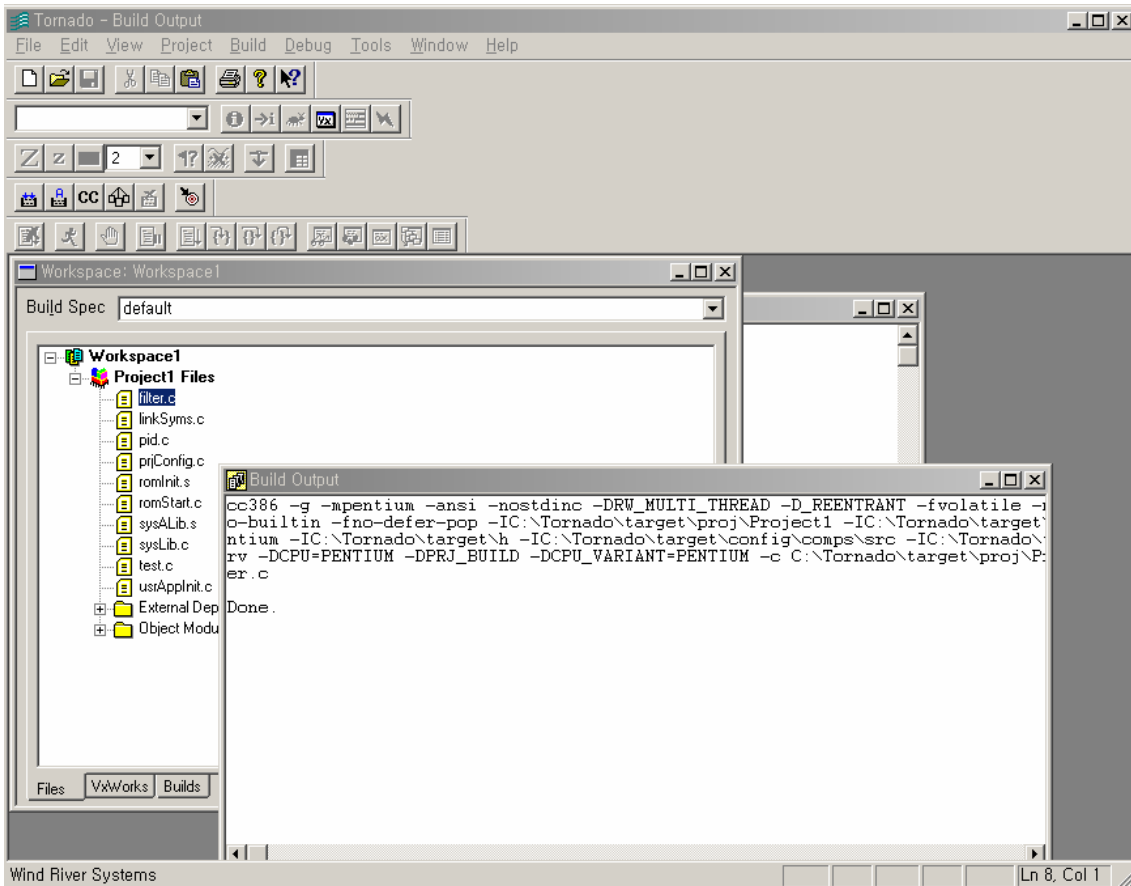
User application program을 작성하기 위해서는 우선 C-source file을 열고 작성하는 과정이 필요하다. File menu에서 New 항목을 선택하면, 다음 그림과 같은 화면이 나오고, 여기에서 작성할 파일의 이름을 입력하면, 새로운 source file을 만들 수 있다.



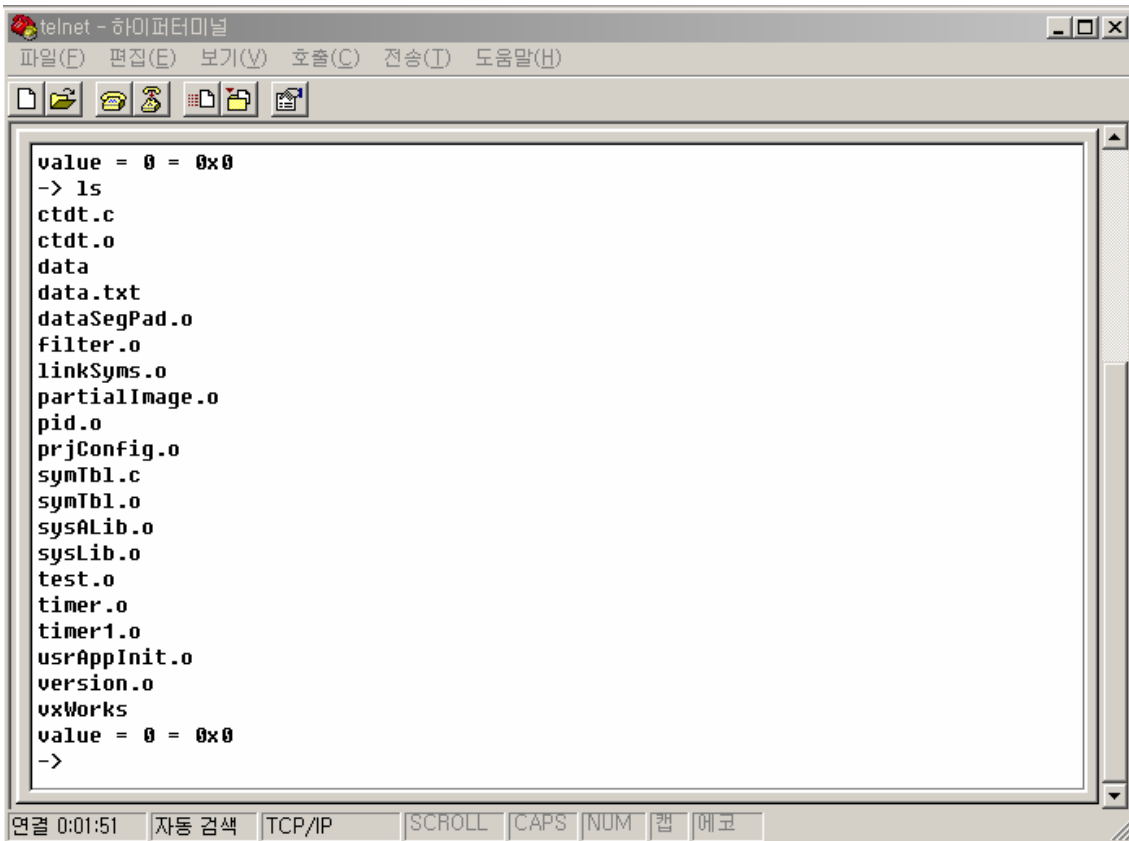
응용프로그램의 source를 다 작성한 후, compile을 하기 위해서는 작성된 파일을 현재 열려 있는 프로젝트에 포함 시키는 작업이 필요하다. 이를 위해서는 Project menu의 Add/Include를 선택하면 다음과 같은 화면이 나오며, 여기에서 작성된 파일을 선택하여 추가 시킬 수 있다. 응용 프로그램의 작성시 주의 할 점은 프로그램의 수행을 위한 entry function은 main()이 아니다. 응용 프로그램에서 main()의 이름을 갖는 함수는 존재할 수 없다. Entry function의 이름은 본인이 임의로 선택하여, 이후 vxWorks에서 수행 시 사용할 수 있다.



다음으로 compile 하고자 하는 파일 명을 클릭하여 선택한 후, Build menu에서 compile을 선택하면 다음 그림과 같이 compile이 수행된다. 만약 compile시 오류가 발생된다면, 오류 메시지의 내용을 파악하여 모든 오류가 수정될 때까지 수정을 계속한다.



Compile이 성공적으로 수행이 되면, *.o 형태의 object file이 생성 되었으며, 이 object file을 target에 down load 시켜서 수행을 할 수 있다. Object file은 vxWorks의 shell 화면에서 down load 할 수 있다. Shell command에서 ls 명령을 입력하면, 현재 연결 되어 있는 host의 현재 directory list를 다음과 같이 보여준다.



The image shows a telnet terminal window titled "telnet - 하이퍼터미널". The window has a menu bar with options: 파일(F), 편집(E), 보기(V), 호출(C), 전송(T), 도움말(H). Below the menu bar is a toolbar with icons for file operations. The main text area displays the following output:

```
value = 0 = 0x0
-> ls
ctdt.c
ctdt.o
data
data.txt
dataSegPad.o
filter.o
linkSyms.o
partialImage.o
pid.o
prjConfig.o
symTbl.c
symTbl.o
sysALib.o
sysLib.o
test.o
timer.o
timer1.o
usrAppInit.o
version.o
vxWorks
value = 0 = 0x0
->
```

At the bottom of the window, there is a status bar with the following information: 연결 0:01:51, 자동 검색, TCP/IP, SCROLL, CAPS, NUM, 캡, 예코.

여기에서 자신이 방금 compile한 파일의 object file이 있는지를 확인한다. 현재 down load 하려는 object file의 이름이 filter.o 라면 shell 에서

-> ld < filter.o

의 명령을 수행하여 down load 한다.

```

telnet - 하이퍼터미널
파일(F) 편집(E) 보기(V) 호출(C) 전송(T) 도움말(H)
tNetTask _netTask      3F90530  50 READY      15be1f  3f904c8      0      0
value = 0 = 0x0
-> ld < filter.o
value = 66462656 = 0x3f623c0 = filter.o_bss + 0x8
-> sp timer
task spawned: id = 0x3f5d6d8, name = t1
value = 66442968 = 0x3f5d6d8
-> i

  NAME      ENTRY      TID  PRI  STATUS  PC      SP      ERRNO  DELAY
  -----
tExcTask   _excTask   3fd3134  0  PEND    18eb4e  3fd30a4      0      0
tLogTask   _logTask   3fd0830  0  PEND    18eb4e  3fd079c      0      0
tShell     _shell     3f6a184  1  READY   158170  3f69e44      0      0
tTelnetd   _telnetd   3f6d788  2  PEND    118a24  3f6d6b8      0      0
tTelnetOutT_telnetOutTa  3ffe214  2  READY   118a24  3ffdf78      0      0
tTelnetInTa_telnetInTas  3f678bc  2  READY   11898b  3f675cc      0      0
tWdbTask   _wdbTask   149a20   3f6b328  3  PEND    118a24  3f6b284      0      0
tNetTask   _netTask   3f90530  50  READY   15be1f  3f904c8      0      0
t1         _timer     3f5d6d8 100  READY   3f62386  3f5d6a0      dd     0
value = 0 = 0x0
-> taskDelete t1
value = 0 = 0x0
->

```

연결 0:00:55 | 자동 검색 | TCP/IP | SCROLL | CAPS | NUM | 캡 | 예코

다음 과정으로, 작성된 응용 프로그램의 entry function의 이름이 timer() 라면,

-> sp timer

와 같이 입력하여 수행할 수 있다. (sp 는 spawn의 의미를 갖는 명령어임. Spawn은 새로운 task process를 발생시킨다는 의미임) 만약 현재 수행 시킨 프로그램이 즉시 끝나지 않고 계속해서 수행되는 프로그램이라면, 위의 그림과 같이 i 명령에 의해서 수행중인 task 목록 중에서 방금 수행시킨 task를 확인할 수 있다. 위의 예에서는 task name이 t1인 task 가 방금 수행 시킨 task 이다. 이 task를 제거 하기 위해서는

-> taskDelete t1

의 명령에 의해서 제거할 수 있다. 응용 프로그램의 개발 과정에서는 같은 파일을 반복적으로 수정하고 수행하는 과정이 필요한데, vxWorks를 재 부팅할 필요 없이, 위에서 설명한 과정을 반복적으로 수행할 수 있다.

VxWorks의 응용 프로그램에서 file을 읽거나 쓰는 작업이 가능한데, 일반적인 C-program의 방법과 동일하다. 이때 사용되는 default file system은 현재 network file system에 의해서 접속된 host의 file system 이다. 즉, 위에서 설명한 ls 명령에 의해서 볼 수 있는 directory가 현재 접속된 directory이며, 이는 host에서 수행중인 FTP server의 user name중 현재 접속중인 user(현재 사용중인 user name: target)의 home directory이다.

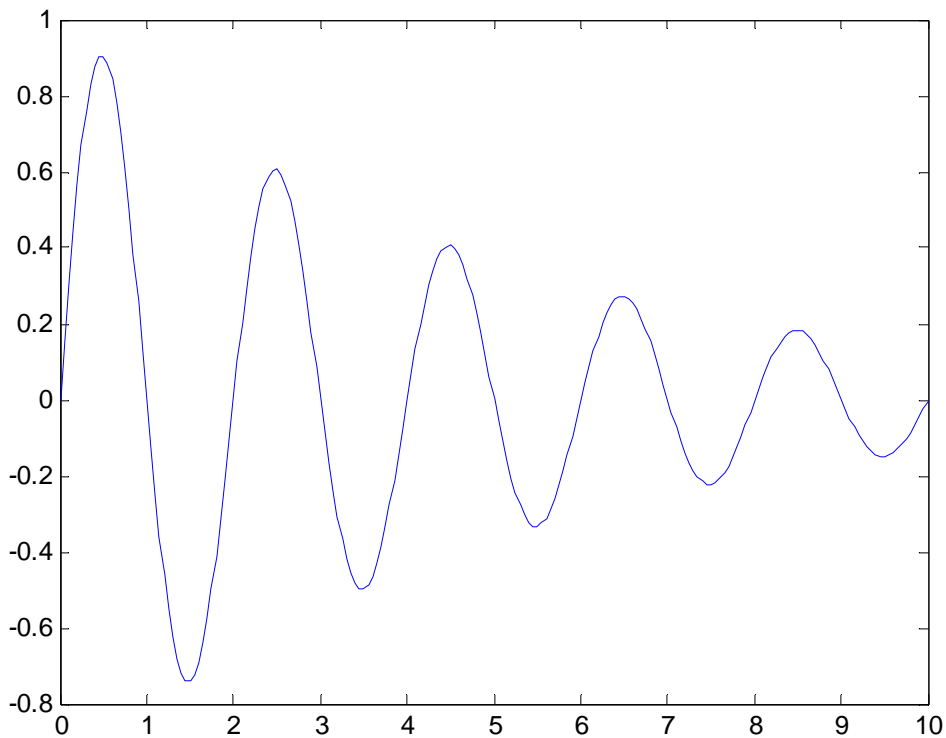
실습:

VxWorks에서 $t = 0$ 에서 $t = 2.0$ 사이의 구간에서 아래의 함수 값을 계산하여 변수에 저장한 후, host 컴퓨터의 파일에 저장한다. 그리고, MATLAB을 사용하여 host 컴퓨터에 저장된 데이터로 아래와 같은 그래프를 그려본다.

$$f(t) = e^{-0.2t} \sin(\pi t)$$

VxWorks의 exponential 함수는 `exp()`이며, sine 함수는 `sin()`이다. 또한 이와 같은 수학 함수를 사용하기 위해서는 다음과 같은 include file이 포함되어야 한다.

```
#include "math.h"
```



이 실습은 VxWorks의 응용 프로그램을 작성하고 데이터 파일을 저장하는 연습을 하기 위한 실습이다. 이 실습에 대해서는 보고서를 제출할 필요가 없다.