

# WIPI Emulator 사용자 설명서

작 성 자: 충남대학교 분산이동컴퓨팅연구실  
초     안: 2004년6월10일  
1차 개정: 2004년8월26일  
2차 개정: 2005년4월18일

## 1. 소개

### 1.1 위피 소개

위피(WIFI: Wireless Internet Platform for Interoperability)는 한국무선인터넷표준화 포럼의 무선인터넷 플랫폼 표준이며 한국정보통신기술협회(TTA)에 의해 TTA 단체 표준 TTAS.KO-06.0036으로 채택된 이동통신 단말기용 응용프로그램의 실행 환경에 대한 표준 규격이다.

### 1.2 위피 에뮬레이터

본 에뮬레이터는 ETRI의 지원으로 충남대학교 분산이동컴퓨팅연구실에서 개발한 위피 에뮬레이터로서 위피 규격 1.2를 따른다.

본 에뮬레이터는 위피 플랫폼이 탑재된 휴대 단말기에서 위피 응용프로그램이 실행되는 것과 유사한 환경을 제공하여 위피 어플리케이션을 일반 PC 환경에서 작성한 후 에뮬레이터에서 실행함으로써 응용프로그램을 실제 단말기에 적용하기 전에 동작을 확인하고 검토할 수 있도록 하는 것을 목적으로 한다.

### 1.3 위피 에뮬레이터의 지원 범위

본 위피 에뮬레이터는 위피 버전1.2 규격의 기본 API 중 C API를 지원하며, 휴대형 단말기를 윈도우 환경에서 에뮬레이션 하는 GUI(Graphical User Interface)를 제공한다. 또한 다수의 위피 프로그램을 실행시킬 수 있는 멀티 응용프로그램을 지원한다.

## 2. 에뮬레이터 설치

### 2.1 에뮬레이터 설치 구성요소

본 에뮬레이터의 구성은 다음과 같다.

- ① 위피 에뮬레이터 실행 파일
- ② 위피 에뮬레이터 스킨
- ③ 위피 응용프로그램 예제 소스

### 2.2 에뮬레이터 설치 환경

본 에뮬레이터는 Microsoft의 Windows 2000과 Windows XP 운영체제가 탑재된 컴퓨터에서 실행할 수 있다.

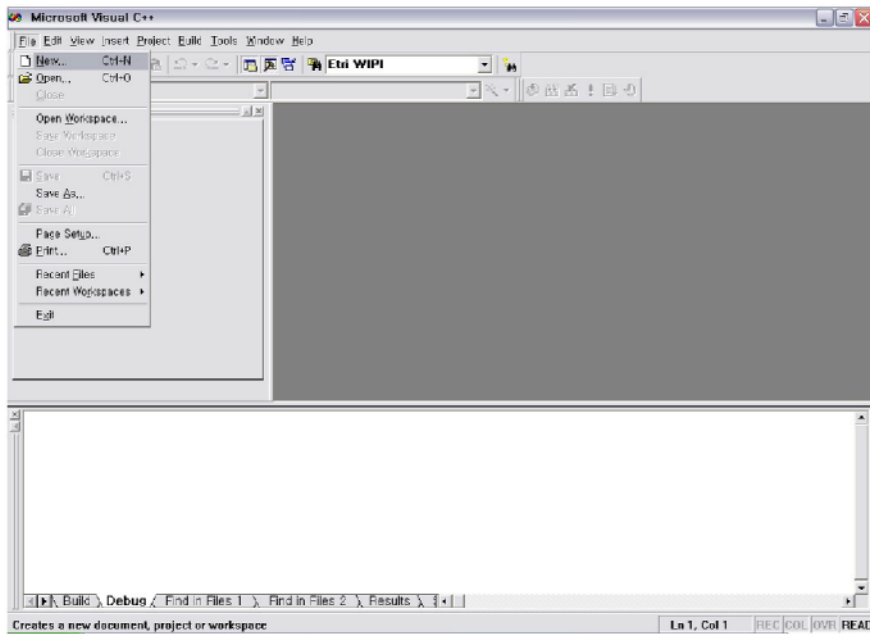
### 2.3 에뮬레이터 설치 및 삭제 방법

제공되는 설치파일인 WIPI\_c\_emulator\_1.2.2.0.exe을 실행하여 설치 과정을 진행한다. 설치가 완료되면, WIPI의 실행파일인 WIPI.exe 파일과 예제소스 폴더, WIPI 응용프로그램폴더(WIPI\_APP), 스킨파일폴더(WIPISkins), Clet생성에 필요한 파일들이 존재하는 참조폴더(include)가 생성된다. 추가적인 설정은 필요 없다. 프로그램 삭제는 간단히 설치된 파일들을 지우는 것으로 완료된다.

### 3. 에뮬레이터를 이용한 응용프로그램 개발 예

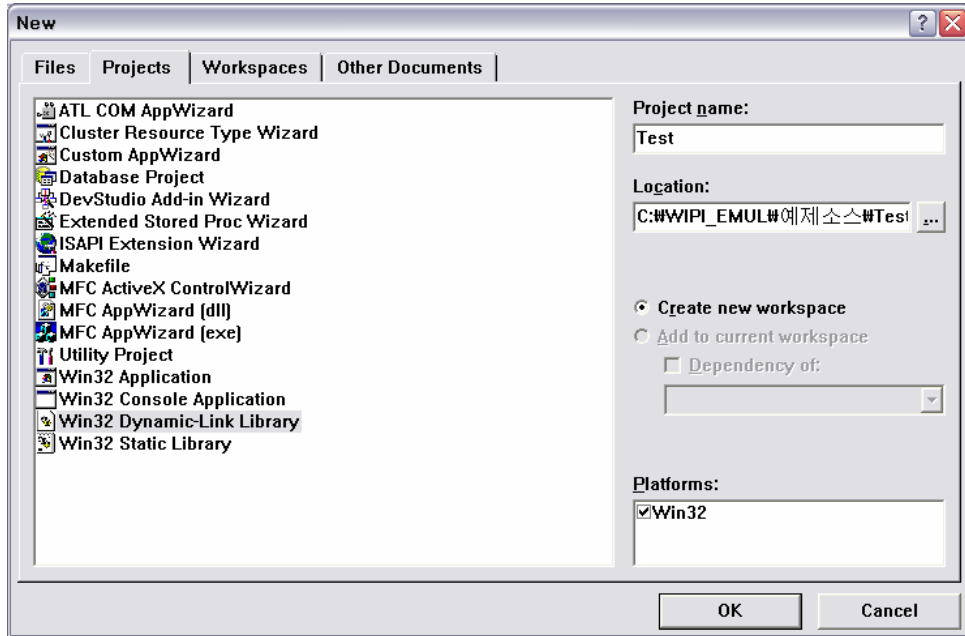
본 장에서는 본 에뮬레이터를 이용하여 위피 응용프로그램(Clet)을 개발하는 과정을 설명한다. 예제로 보인 Clet은 에뮬레이터 화면에 “Hello WIPI”를 출력하는 응용프로그램이며, 이름은 test 이다.

- Clet을 개발하기 위해 Visual Studio를 시작한다.  
File → New를 클릭하여 새로운 프로젝트를 만든다.



[그림 4.1] Visual Studio에서 새로운 프로젝트 선택

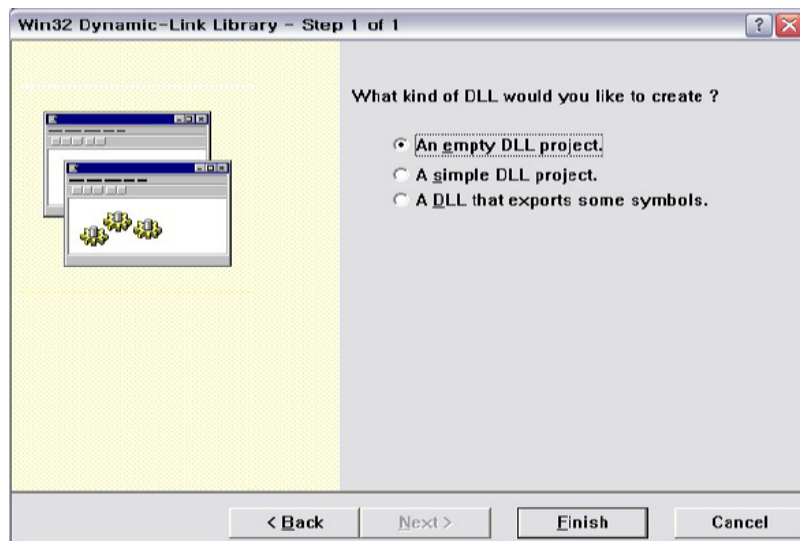
- 새로 만들 프로젝트의 종류는 Win32 Dynamic-Link Library이다. 프로젝트의 이름은 test으로 하고 생성될 위치는 프로젝트가 생성되기를 원하는 폴더를 지정한다[그림 4.2].



[그림 4.2] 프로젝트 타입 선택

OK 버튼을 누르고 다음 단계로 진행한다.

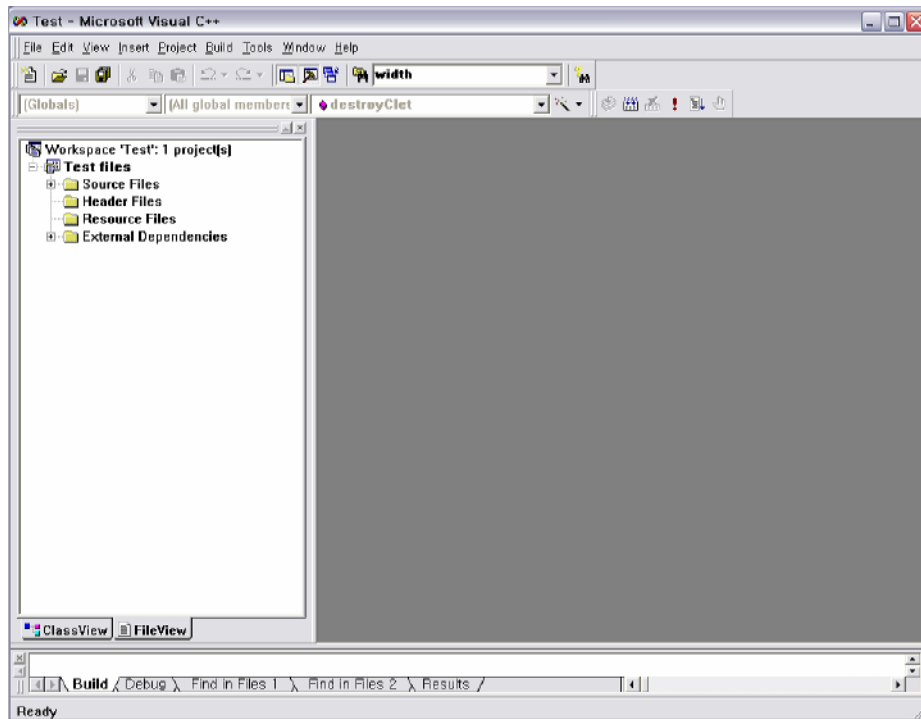
- 다음 단계에서는 어떤 종류의 DLL을 만들 것인지 물어보는데[그림 4.3], 첫 번째의 ‘An empty DLL project’를 선택하고, ‘Finish’ 버튼을 눌러서 프로젝트 생성을 종료한다. 프로젝트를 생성하고자 하는 폴더에 프로젝트가 제대로 생성 되었는지 확인한다.



[그림 4.3] DLL 종류 선택

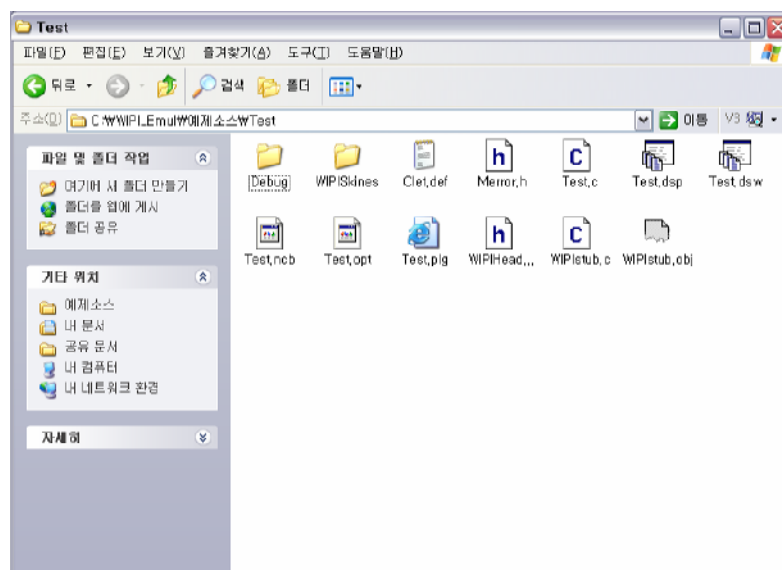
- 프로젝트가 생성되면 워크스페이스파일(test.dsw)을 연다. [그림 4.4]와 같이 아무것도

없는 빈 프로젝트가 열릴 것이다.



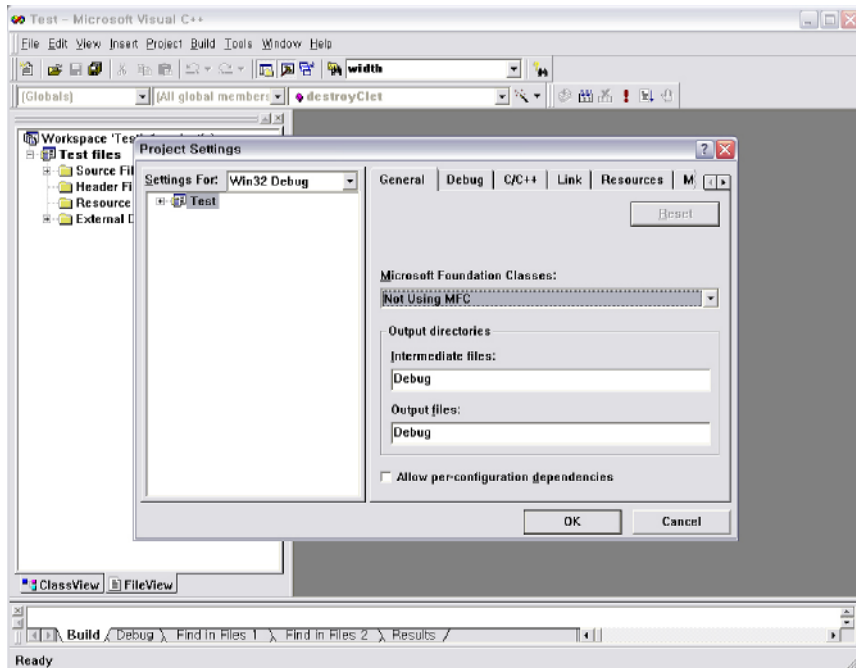
[그림 4.4] test 프로젝트

- Clet을 작성하기 위해서는 참조폴더(include)내의 파일들이 필요하다. 따라서 이 파일들을 [그림 4.5]와 같이 프로젝트 폴더(예 C:\WWIPL\_Emul\예제소스\test)에 복사하거나 임의의 폴더에 복사한 후 정확한 경로를 지정해 준다.



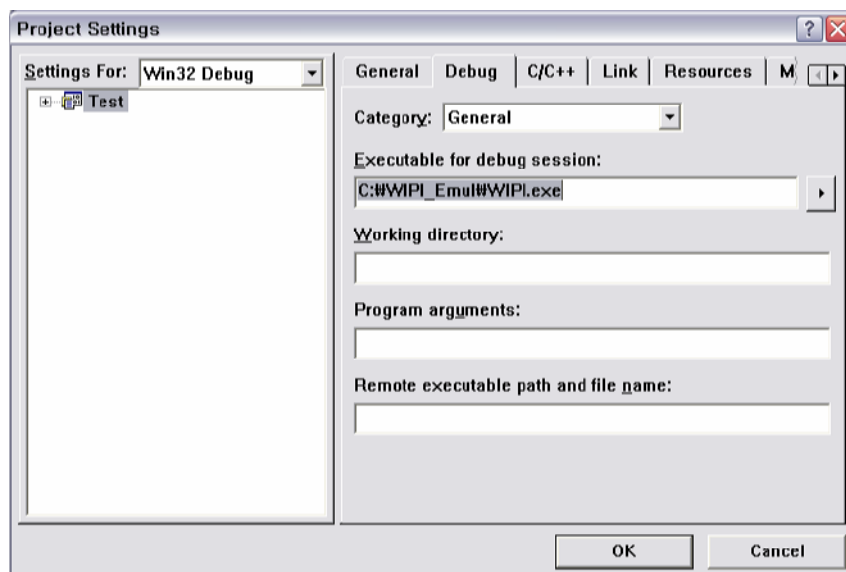
[그림 4.5] 프로젝트 폴더

- 필요한 파일까지 복사를 하면, 프로젝트의 세팅을 변경한다. [그림 4.6]와 같이 Project→Settings를 선택하여 프로젝트 세팅 창을 띄운다.



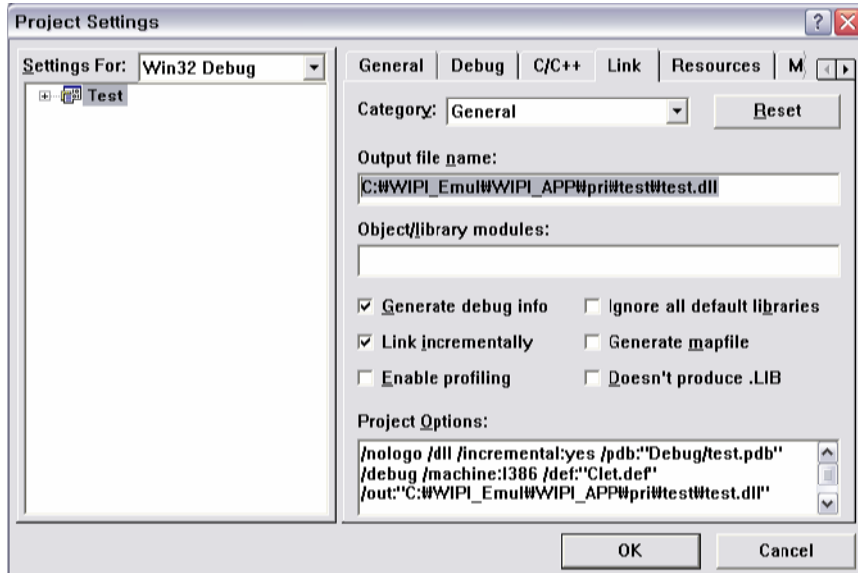
[그림 4.6] 프로젝트 세팅

- 첫 번째로 설정할 사항은 디버그 세션을 정해주는 일이다. 이것은 디버깅을 할 때 에뮬레이터가 동작하도록 설정하는 것으로, WIPi.exe의 위치(예 C:\WWIPi\_Emul\WWIPi.exe)를 정해준다[그림 4.7].



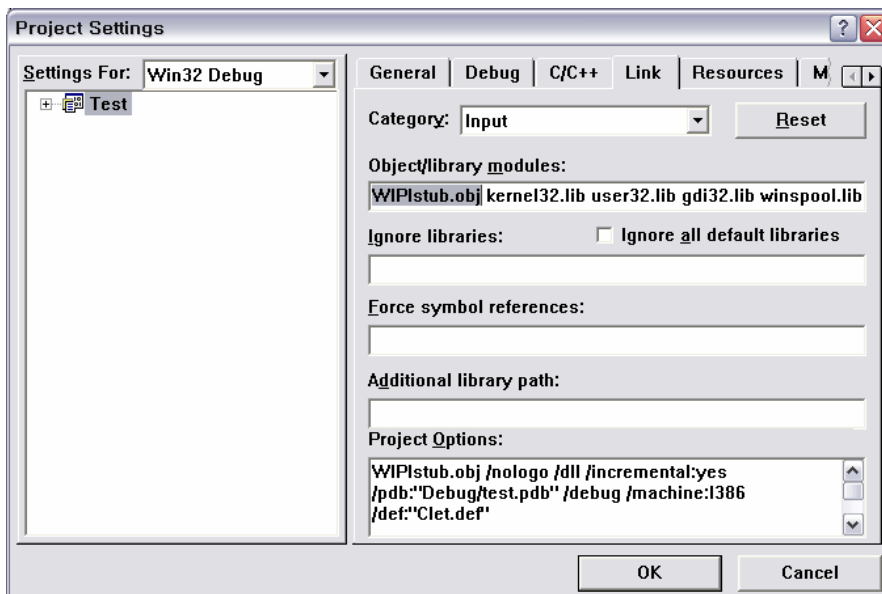
[그림 4.7] Executable for debug session 설정

- 다음으로 Link 탭의 General 카테고리에서 output file name을 위피 에뮬레이터의 프로그램폴더인 `pri`폴더의 `WtestWtest.dll`로 설정(예 `C:\WWIPI_Emul\WWIPI_APP\pri\WtestWtest.dll`)하여 준다[그림 4.8]. 이것은 `test.dll`을 생성한 후 다시 `C:\WWIPI_Emul\WWIPI_APP\pri\WtestWtest.dll` 폴더에 복사해 주는 수고를 덜어준다.



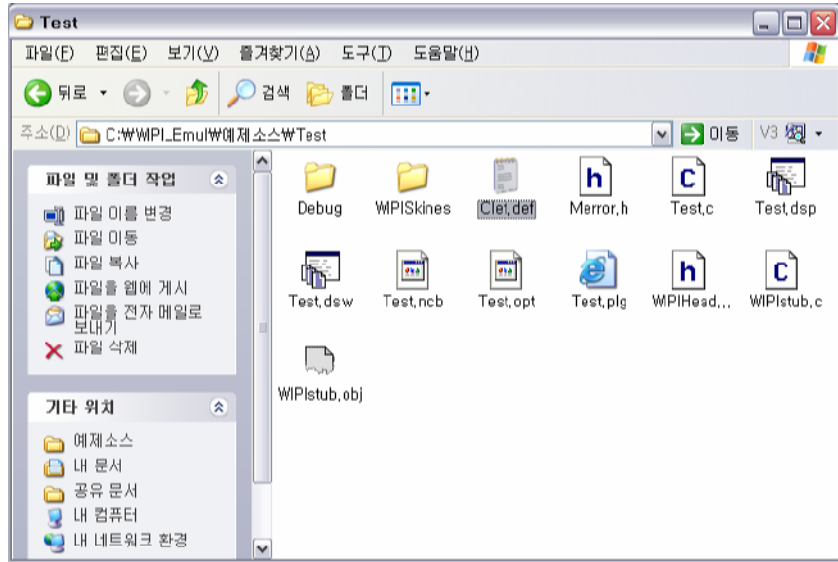
[그림 4.8] output file name 설정

- Link 탭의 Input 카테고리에서 ObjectWlibrary modules에 `WIPISub.obj`를 추가한다. 이 파일은 [그림 4.5]에서 프로젝트 폴더에 복사한 파일이다[그림 4.9].



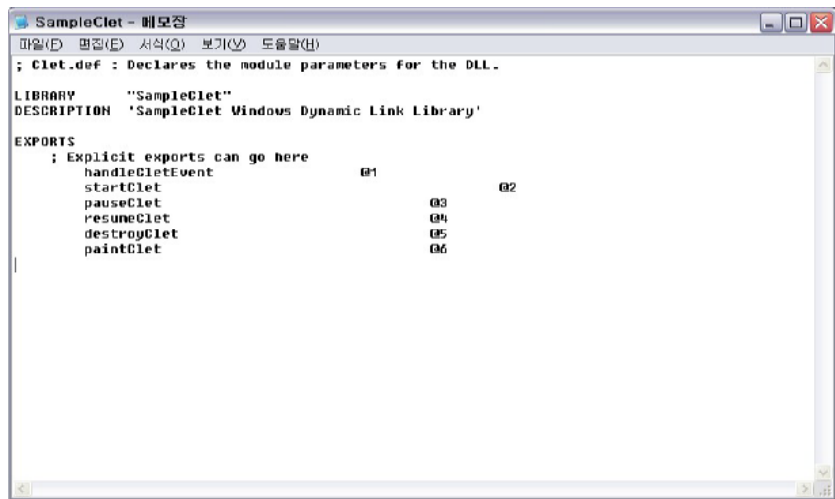
[그림 4.9] ObjectWlibrary modules 설정

- Clet dll파일은 에플레이터에서 명시적 연결 방법을 사용하여 링크하기 때문에 Definition 파일(.DEF)을 만들어 주어야 한다. Clet.def 파일은 참조폴더(include)에서 복사하여 사용해도 되고, 새로 만들어도 된다. [그림 4.10].



[그림 4.10] Clet.def 파일 생성

- 모든 def 파일의 내용은 동일하게 다음과 같다[그림 4.11].



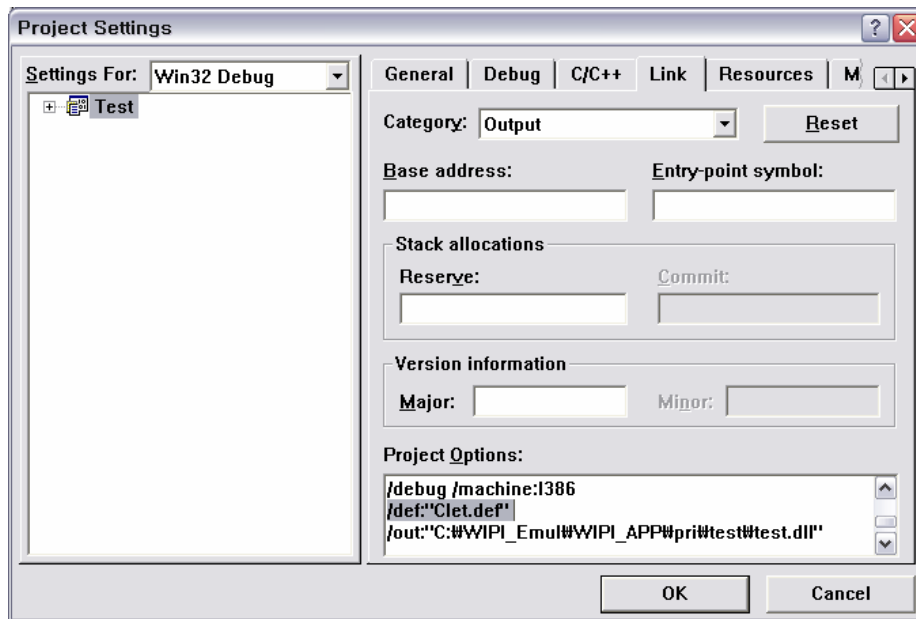
[그림 4.11] Clet.def

기존의 것을 복사했다면, LIBRARY 항목의 이름은 바꾸어 주는 것이 좋다.

- Clet.def 파일을 만들었으면 test 프로젝트에서도 Clet.def 파일을 사용할 수 있도록 옵션

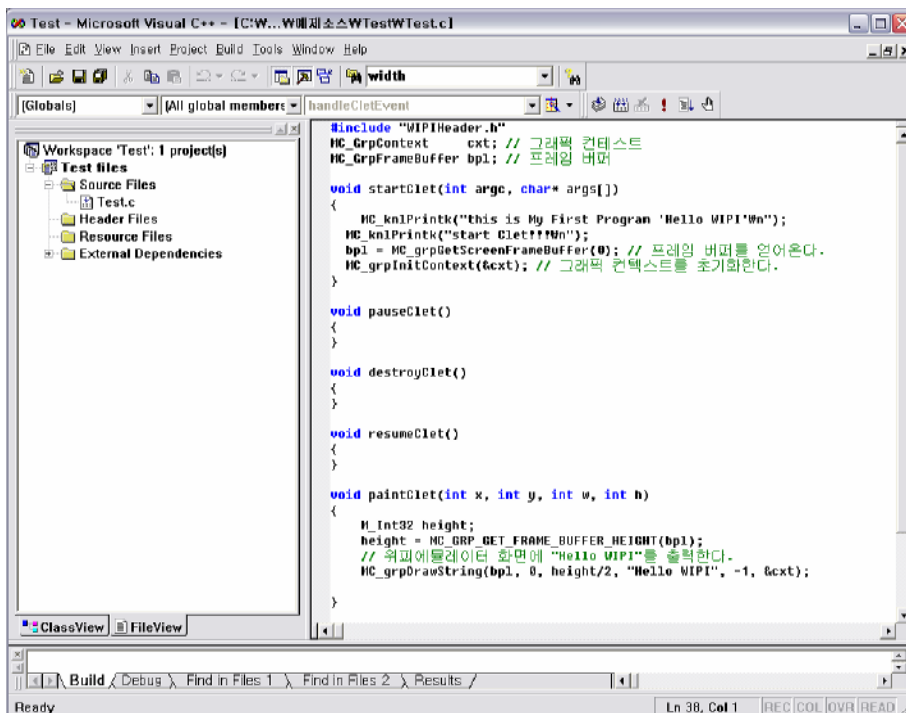


을 변경해 주어야 한다. Project→settings의 Link 탭에서 Project Options 항목에 ‘Wdef:” Clet.def”’ 를 추가해 준다[그림 4.12].



[그림 4.12] 프로젝트 옵션 변경

● 이것으로 Visual Studio의 프로젝트 세팅은 끝났다. Clet을 생성하는 코드를 작성하고, 컴파일한다[그림 4.13].



[그림 4.13] Clet 작성

```

#include "WIPIHeader.h"
MC_GrpContext    cxt; WW 그래픽 컨텍스트
MC_GrpFrameBuffer bpl; WW 프레임 버퍼

void startClet(int argc, char* args[])
{
    MC_knlPrintk("this is My First Program 'Hello WIPI'\n");
    MC_knlPrintk("start Clet!!!\n");
    bpl = MC_grpGetScreenFrameBuffer(0); WW 프레임 버퍼를 얻어온다.
    MC_grpInitContext(&cxt); WW 그래픽 컨텍스트를 초기화한다.
}

void pauseClet()
{
}

void destroyClet()
{
}

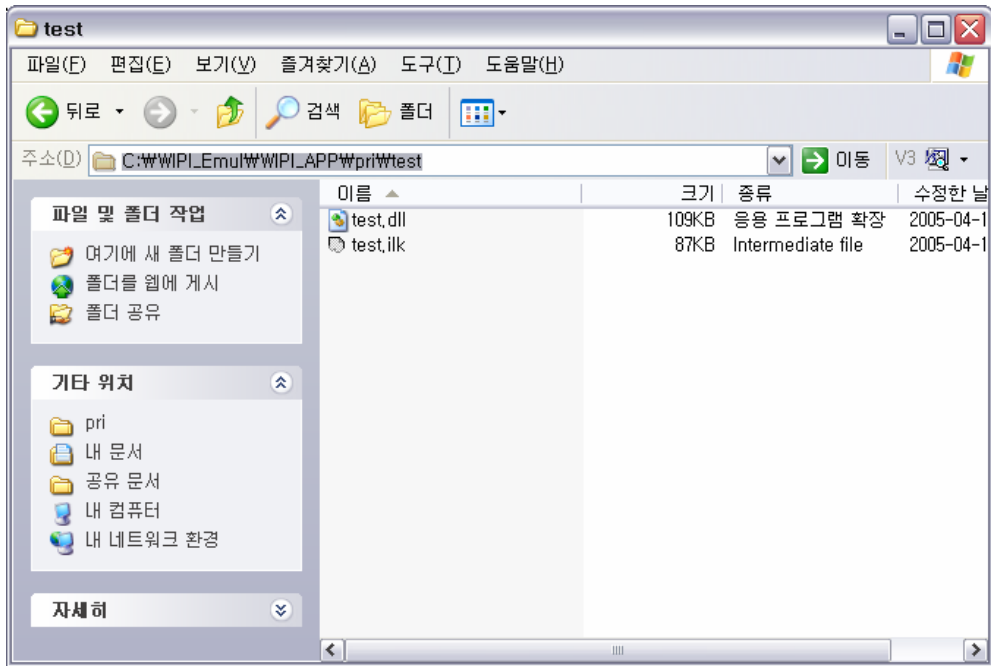
void resumeClet()
{
}

void paintClet(int x, int y, int w, int h)
{
    M_Int32 height;
    height = MC_GRP_GET_FRAME_BUFFER_HEIGHT(bpl);
    WW 위피에뮬레이터 화면에 "Hello WIPI"를 출력한다.
    MC_grpDrawString(bpl, 0, heightW2, "Hello WIPI", -1, &cxt);
}

void handleCletEvent(int type, int parm1, int parm2)
{
}

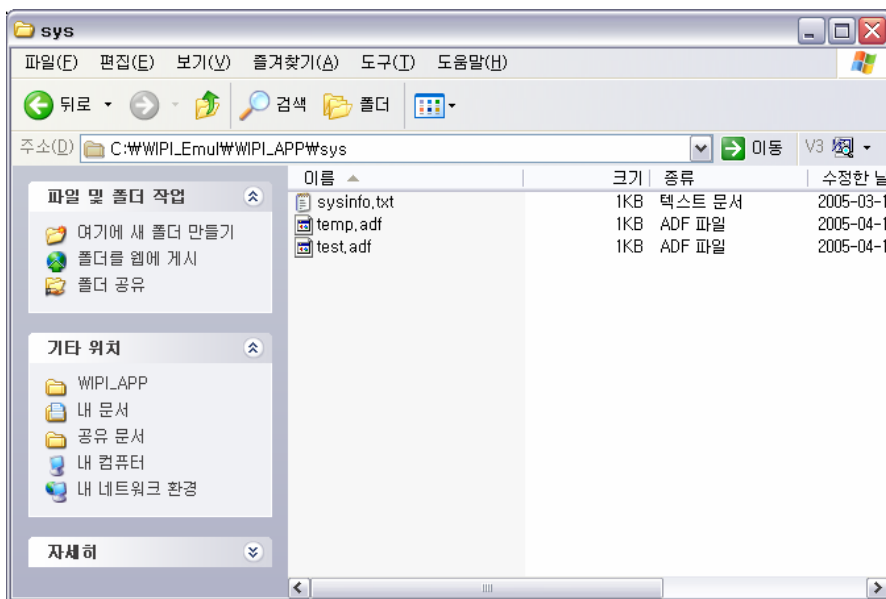
```

- 컴파일 하면 test.dll 파일이 생성된다. 생성된 dll 파일을 위피 에뮬레이터의 프로그램 폴더인 pri폴더내에 test (예 C:WWIPL\_EmulWWIPL\_APPWpriWtest) 폴더로 복사한다. [그림 4.8]와 같이 생성될 파일의 위치를 지정해 주었다면 복사하는 과정은 생략해도 무방하다. C:WWIPL\_EmulWWIPL\_APPWpriWtest폴더에서 생성된 파일을 확인한다 [그림 4.14].



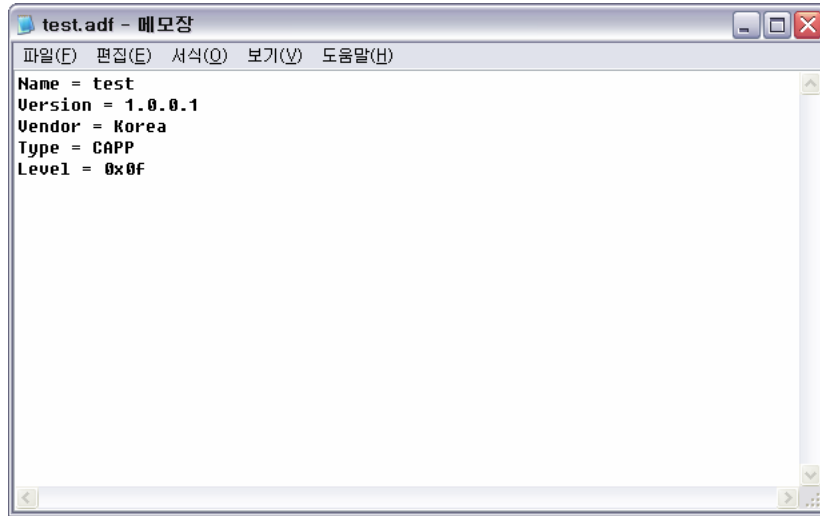
[그림 4.14] test.dll 생성

- test.dll은 생성되었으나 아직 실행은 해 볼 수 없다. 실행을 하기 위해서는 ADF(Application Description File) 파일이 필요하다. ADF 파일은 위피의 시스템폴더인 sys (예 C:\WWIPL\_Emul\WWIPL\_APP\sys) 폴더에 위치하며, 이름은 test.adf 이다[그림 4.15]. temp.adf를 수정하거나, 새로 생성한다.



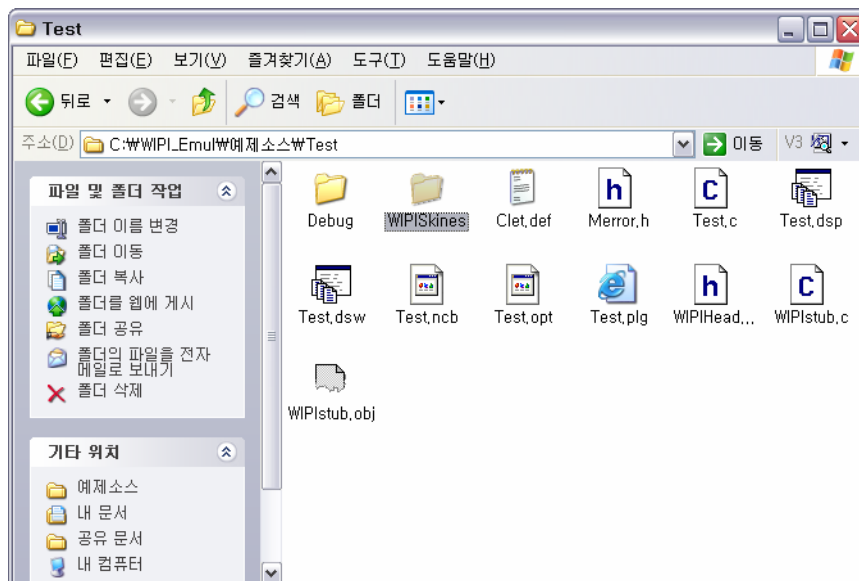
[그림 4.15] test.adf 생성

- test.adf의 내용은 [그림 4.16]과 같다.



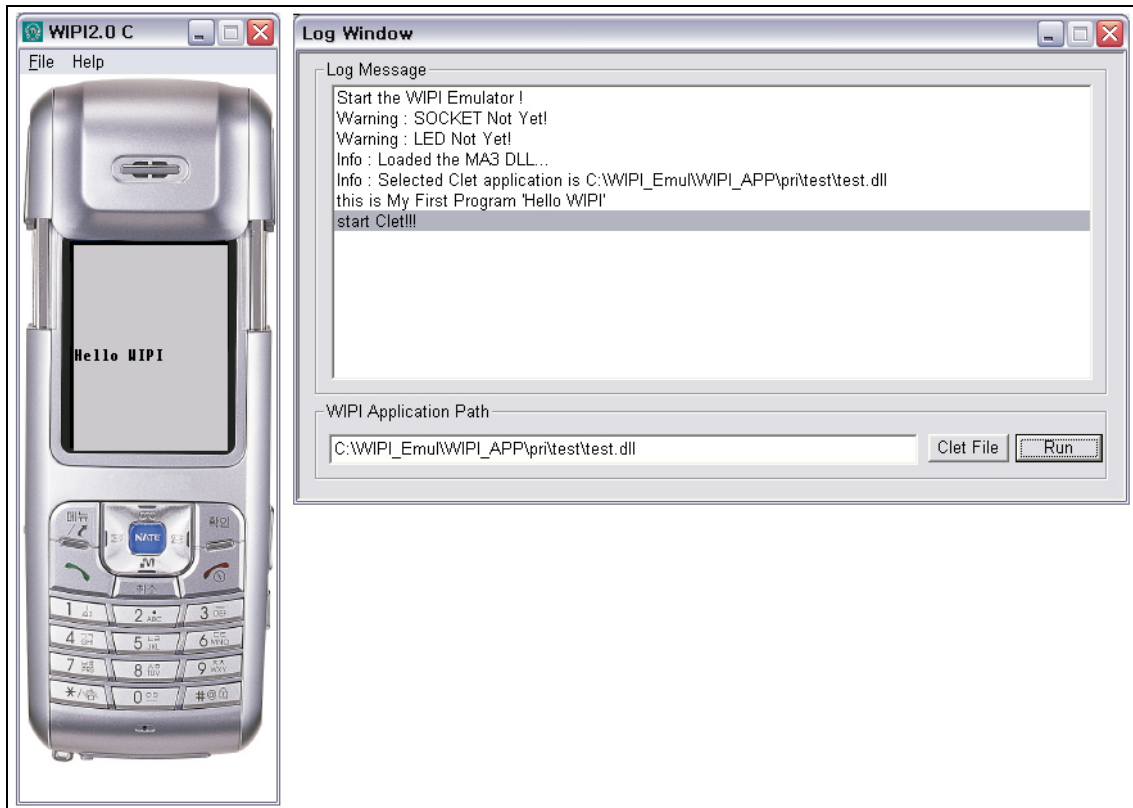
[그림 4.16] test.adf

- 이 과정까지 끝나면 test.dll을 실행해 볼 수 있다. WIPI.exe를 실행시키고, Clet file 버튼을 눌러 방금 생성한 test.dll을 선택하고 실행시킨다. [그림 4.7]에서 디버그 세션을 설정하였으면, Visual Studio에서 바로 실행해 볼 수도 있다. Visual Studio에서 F5 키를 누르면 에뮬레이터가 실행되며, 앞에서와 마찬가지로 Clet file 버튼을 눌러 방금 생성한 test.dll을 선택하고 실행시킨다. 이때 프로젝트 폴더에 스킨 파일이 없으면, 에뮬레이터가 실행되다가 바로 종료되는데, [그림 4.17]처럼 스킨 폴더를 프로젝트 폴더에 복사해 준다.



[그림 4.17] 스킨 폴더 복사

- [그림 4.18]은 test.dll을 실행한 화면이다.



[그림 4.18] test.dll 실행

- ※ 에뮬레이터 사용 중에 문의 점이 있으신 분은 아래의 연락처로 문의하시기 바랍니다.
- 충남대학교 공과대학 컴퓨터 공학과 분산 이동컴퓨팅 연구실
- Phone: 042-823-6049 / 042-821-7726
- Fax: 042-822-4997
- E-mail: here4you@ce.cnu.ac.kr