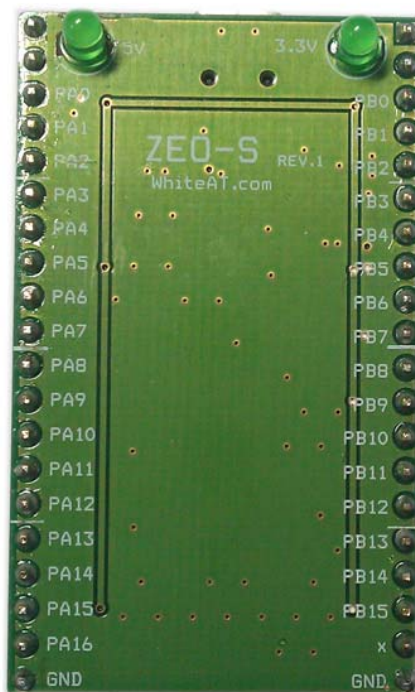


ZEO

Programming Guide



HOME PAGE: <http://WhiteAT.com>

E-MAIL: whiteat@whiteat.com

1. 준비	1
1.1 OS	1
1.2 사용 툴	1
1.3 드라이버 설치	1
1.4 라이브러리	1
 2. C#으로 시작	2
2.1 필요한 라이브러리	2
2.2 프로젝트 생성	3
2.3 코딩	7
2.4 소스 코드	13
2.5 결과	14
 3. C++ 로 시작	15
3.1 필요한 파일	15
3.2 프로젝트 생성	16
3.3 소스 코드	21
3.4 결과	25

1. 준비

1.1. OS

- Windows XP, Windows 7 (32bit, 64bit 모두 가능)

1.2. 사용 툴

- C# - Visual Studio 2008 사용 (Visual Studio 2005 도 가능)
- C++ - Visual C++ 6.0 사용 (MFC 사용)

1.3. 드라이버 설치

ZEO 모듈을 사용하려면 ZEO드라이버를 설치해야 하는데 자세한 드라이버 설치 방법은 [ZEO 매뉴얼](#)¹⁾에 설명되어 있습니다.

1.4. 라이브러리

ZEO 모듈을 제어하는데 몇 가지 라이브러리가 필요합니다. 이 라이브러리는 PC와 ZEO 제어 및 감시용 함수로 구성되어 있으며 사용자는 이 함수의 호출만으로 ZEO 모듈 제어가 가능합니다.

라이브러리는 사용 툴에 따라 달라지므로 각각의 프로그램 작성방법에서 알려 드리겠습니다.

1) 매뉴얼, 드라이버 제공: <http://whiteat.com/57501>

2. C# 으로 시작

이 단원에서는 Visual Studio 2008를 사용하였으며 2005 이상 버전이면 상관 없습니다.

C# 기본학습을 원하시는 분을 위해 [C# 완전 기초 편](#)²⁾이 준비되어 있습니다.

2.1. 필요한 라이브러리

C# 으로 프로그래밍하려면 2개의 라이브러리가 필요합니다. 새로운 프로젝트를 생성할 때 다운 받은 라이브러리를 해당 폴더로 복사해서 사용해야 하니 바탕화면이나 원하는 위치에 [다운로드](#)³⁾합니다.

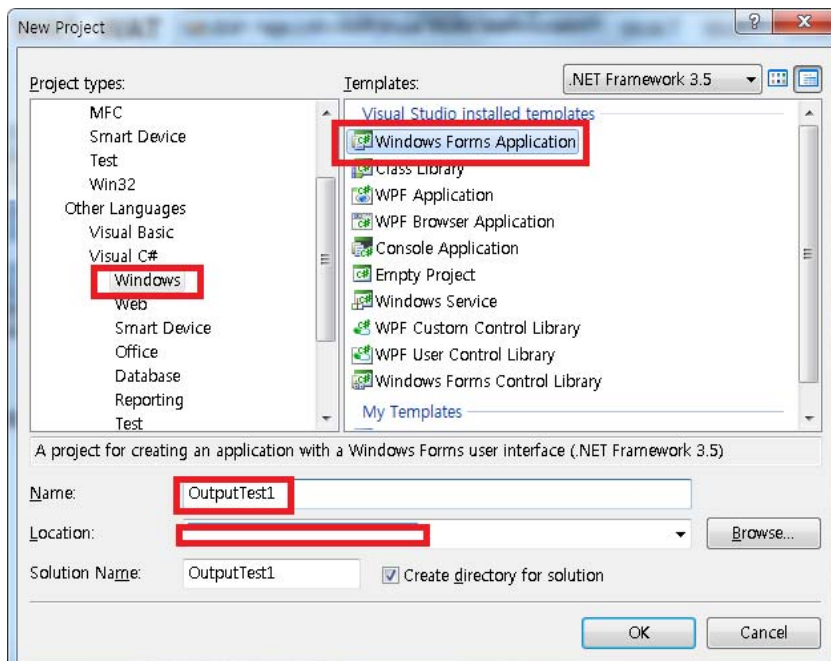
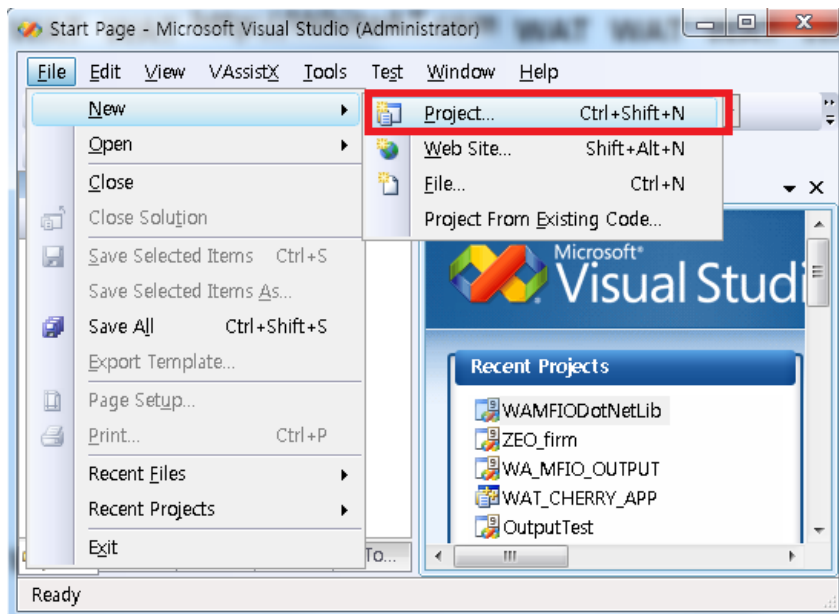
파일명	용도
LibUsbDotNet.dll	기본적인 USB 통신을 위한 라이브러리
ZeoDotNetLib.dll	ZEO 모듈 통신용 라이브러리

2) C# 완전 기초편: <http://whiteat.com/31559>

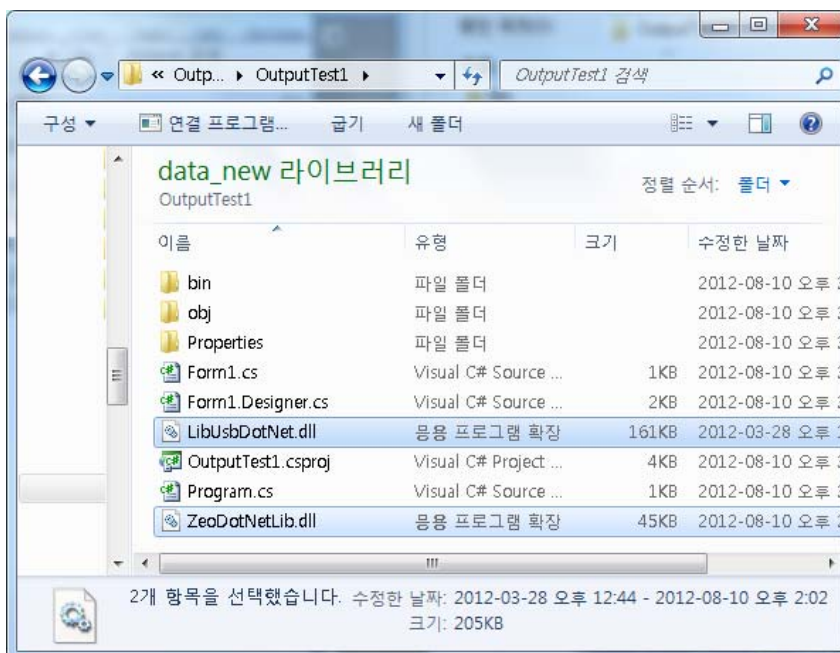
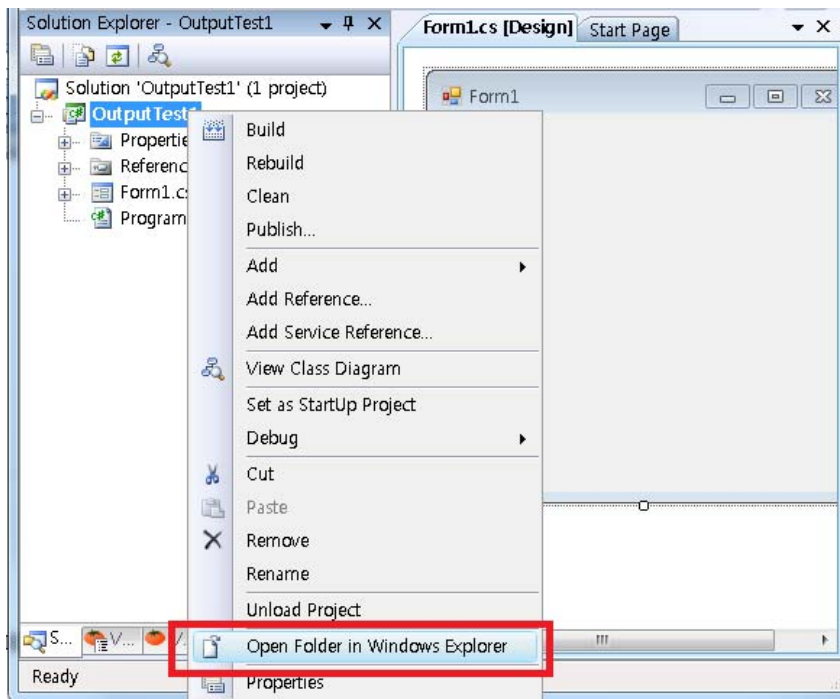
3) 라이브러리 다운로드 경로: <http://whiteat.com/185841>

2.2. 프로젝트 생성

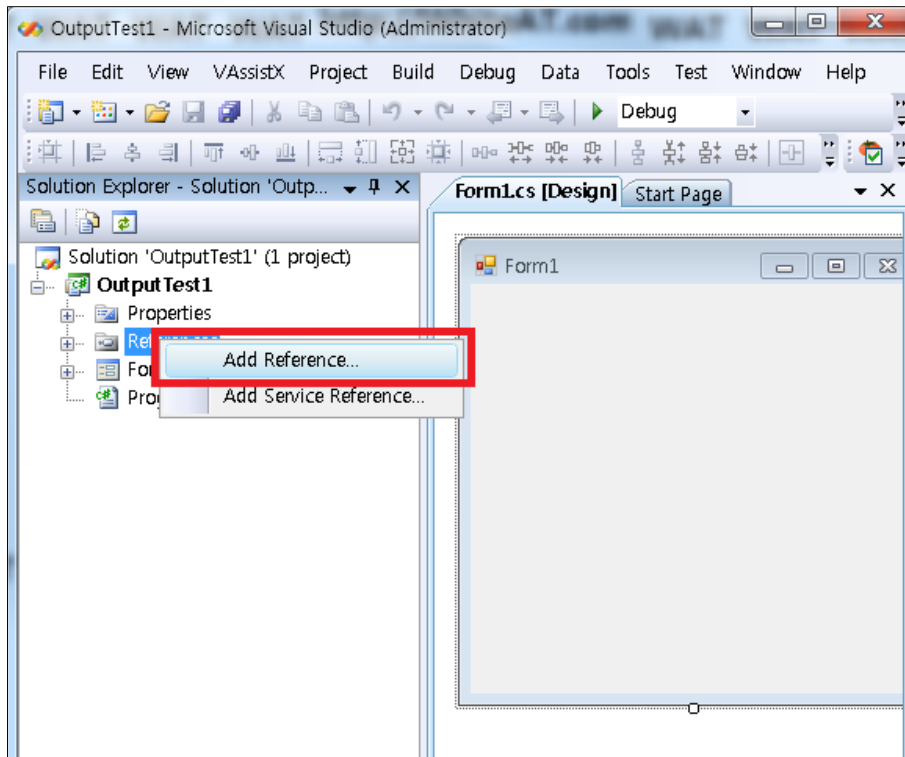
새로운 프로젝트를 생성합니다.



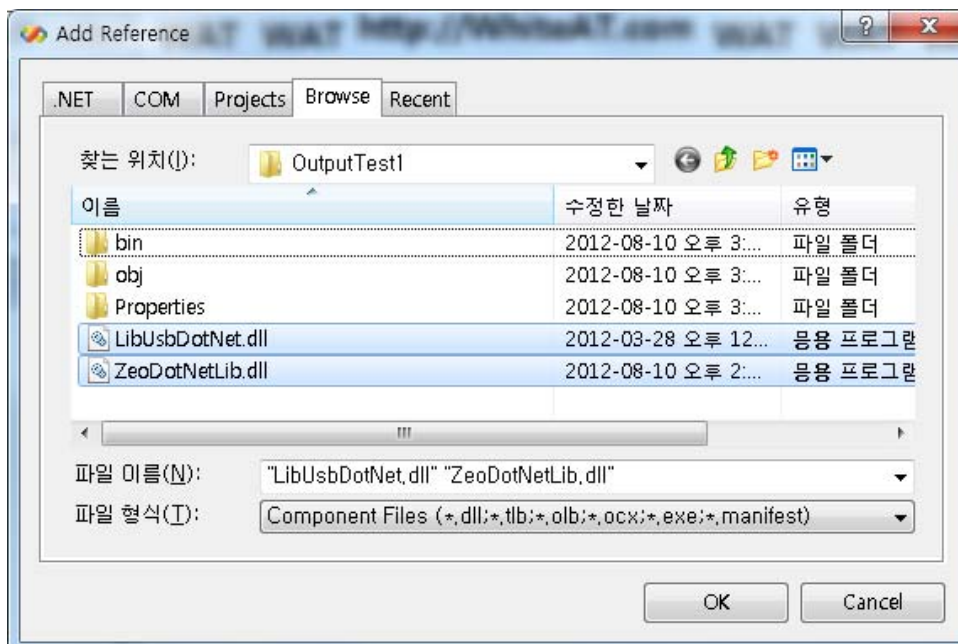
앞에서 다운로드한 2개의 라이브러리를 프로젝트에 추가해야 합니다. 프로젝트 폴더를 열어서 다운로드 받은 2개의 라이브러리를 복사합니다. 복사되는 위치는 어디든지 상관없지만 [Open Folder in Windows Explorer]로 열린 폴더에서 Form1.cs 이 있는 하위 폴더에 복사하는 것이 좋습니다.



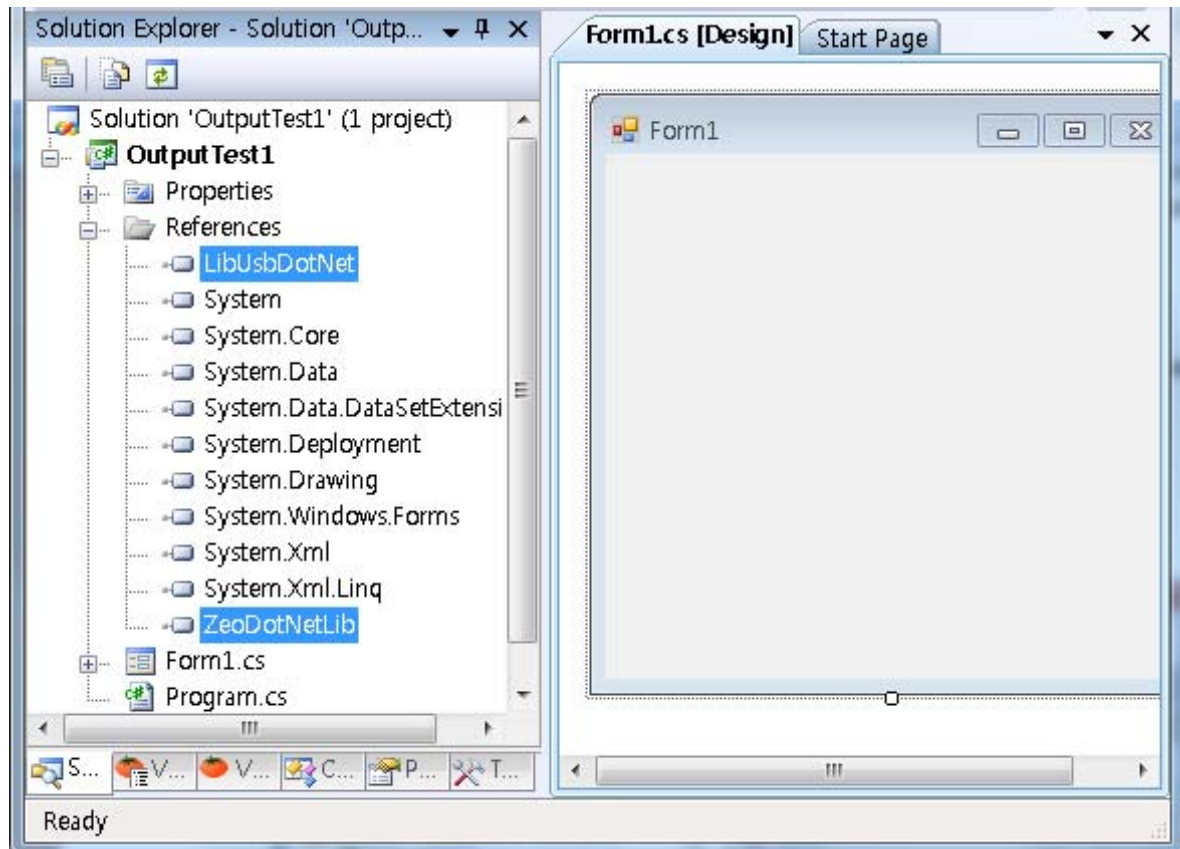
라이브러리를 가져 왔으니 프로젝트에 포함해야 합니다.



[Browse]탭에서 아래처럼 2개의 라이브러리를 선택합니다.



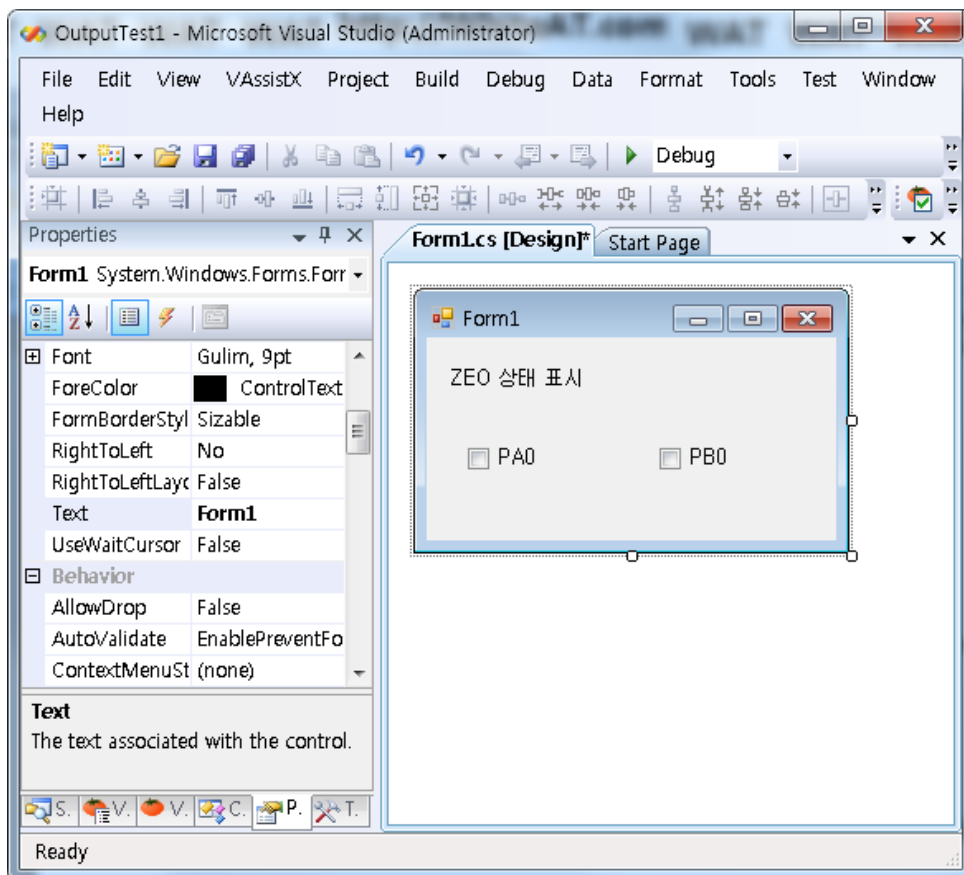
라이브러리를 추가한 것으로 준비를 프로젝트 생성을 완료하였습니다.



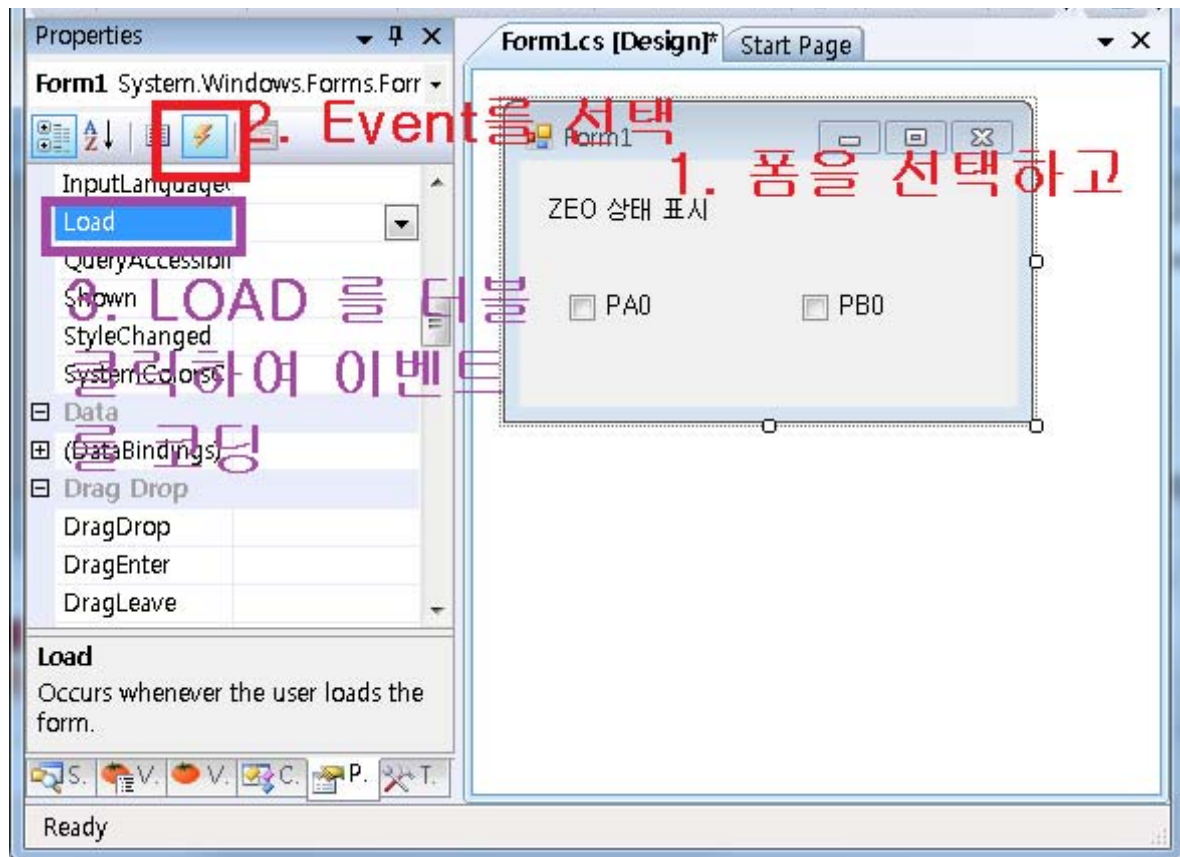
2.3. 코딩

체크 박스의 상태에 따라 PA0 과 PB0 의 출력 상태를 변경해 보겠습니다.

CheckBox 2개와 Label 을 1개 추가합니다. 각각의 이름을 chbPA0, chbPB0, label1으로 하고, 텍스트는 PA0, PB0 로 하였습니다.



프로그램이 시작할 때 ZEO 모듈간의 통신을 초기화해야 하는데 아래의 그림 순서대로 Form_Loading() 함수를 생성합니다.



Form_Load 함수 편집이 가능해 지면 아래의 그림을 참조하여 6 줄을 추가하고 실행합니다.

```
using ZeoDotNetLib;
```

```
ZeoLib ZEO = new ZeoLib();
```

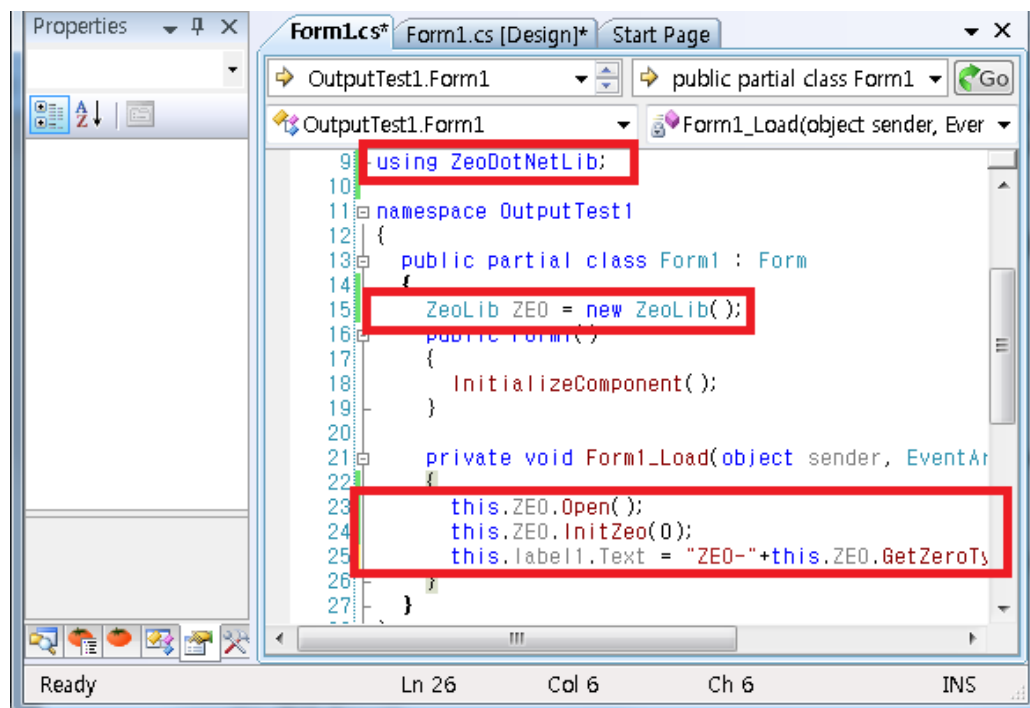
```
this.ZEO.Open();
```

```
this.ZEO.InitZeo(0);
```

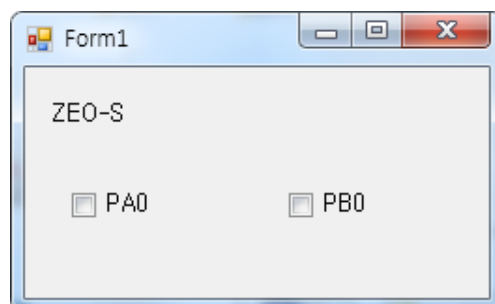
```
this.label1.Text = "ZEO-"+this.ZEO.GetZeroType().ToString();
```

```
this.ZEO.Pin_Set( this.ZEO.PIN_ALL); // 모든 LED OFF
```

```
this.ZEO.PORT_DirOutputALL(); // 모든 핀을 출력으로 설정
```

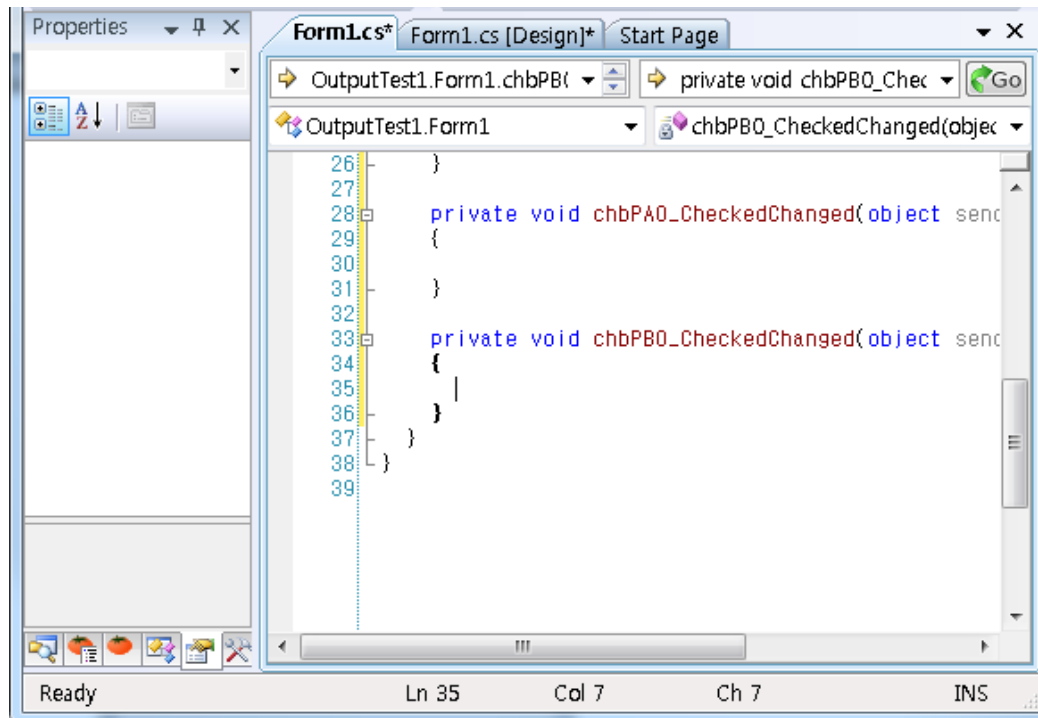


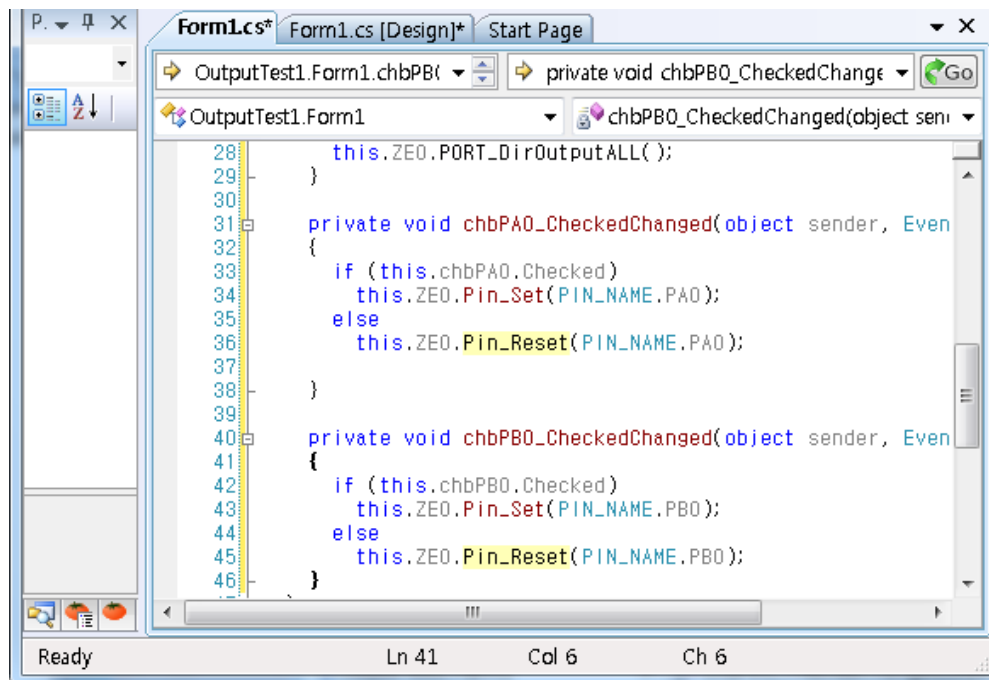
실행했을 때 위쪽에 ZEO-S 모델명이 나타나면 ZEO-S와의 연결이 정상적으로 된 것입니다.



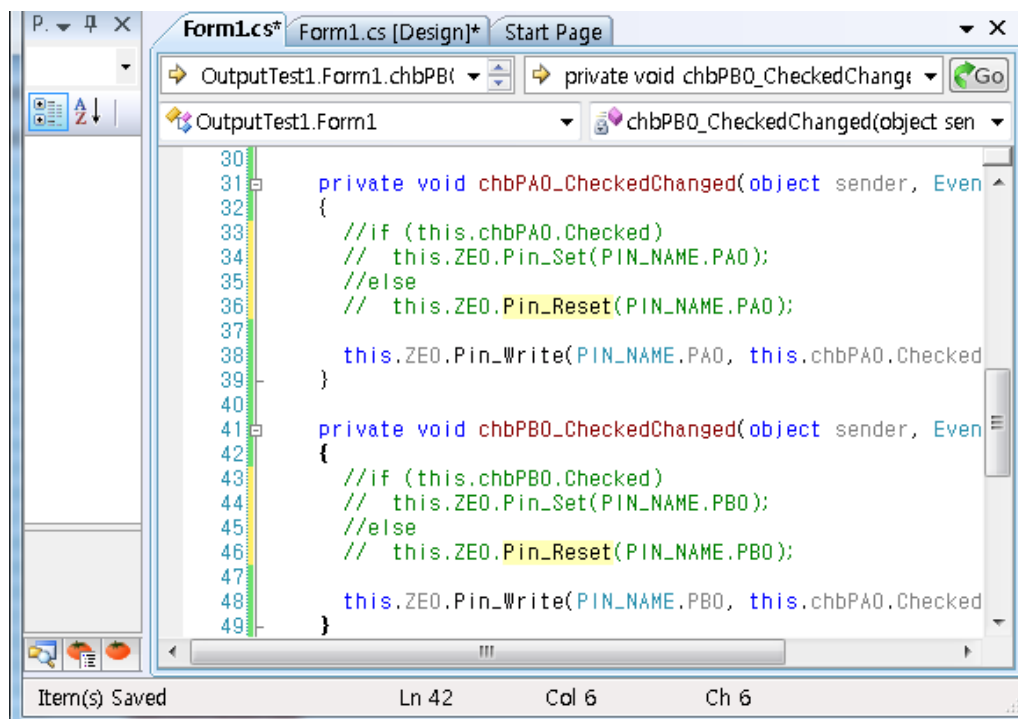
체크박스를 클릭했을 때 포트의 출력을 변하게 해보겠습니다.

각각의 체크박스를 더블클릭하여 클릭했을 때의 함수가 나타나게 하여 코딩합니다.



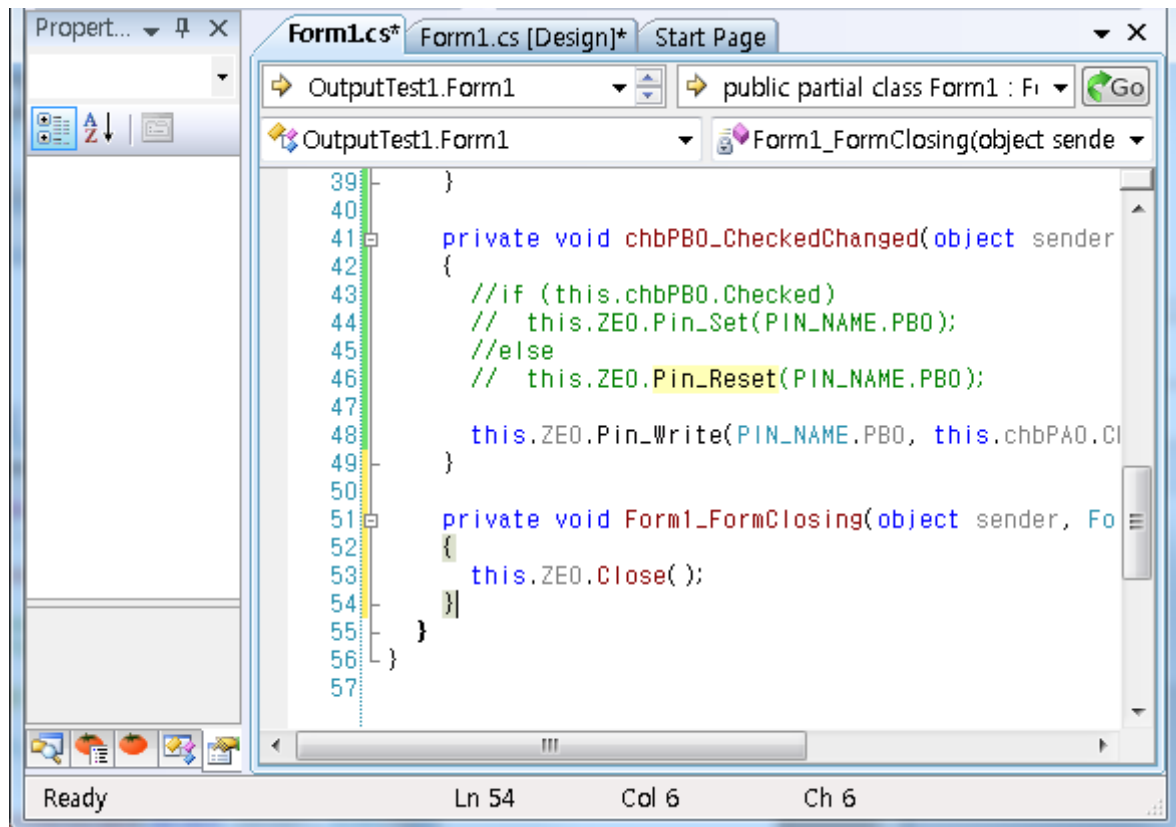


< 코딩 방법 1 >



< 코딩 방법 2 >

프로그램이 종료될 때 ZEO와의 접속을 끊어줘야 하는데 Close() 메소드를 사용하면 됩니다.



2.4. 소스코드

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;
using ZeoDotNetLib;

namespace OutputTest1
{
    public partial class Form1 : Form
    {
        ZeoLib ZEO = new ZeoLib();
        public Form1()
        {
            InitializeComponent();

            private void Form1_Load(object sender, EventArgs e)
            {
                this.ZEO.Open();
                this.ZEO.InitZeo(0);
                this.label1.Text = "ZEO-" + this.ZEO.GetZeroType().ToString();

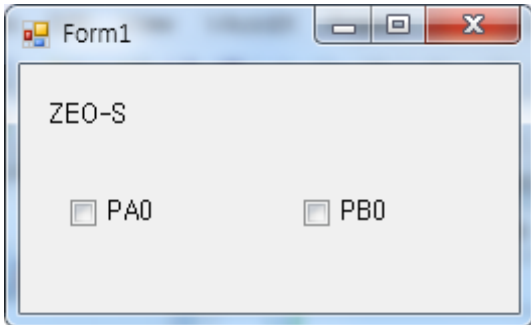
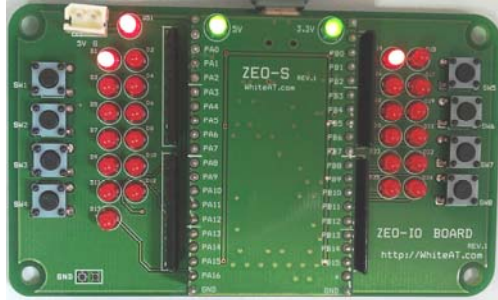
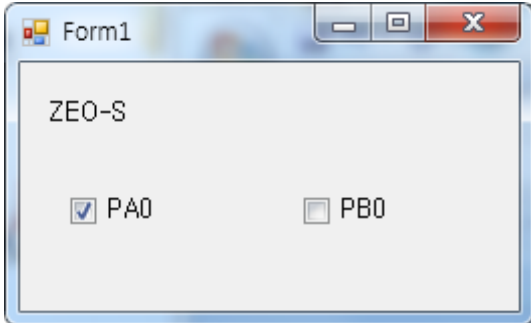
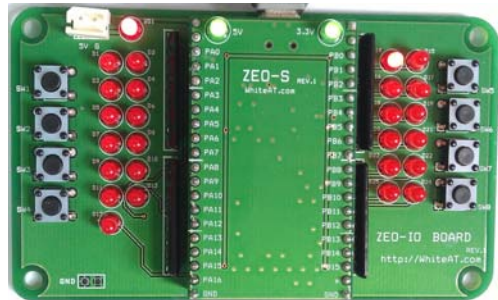
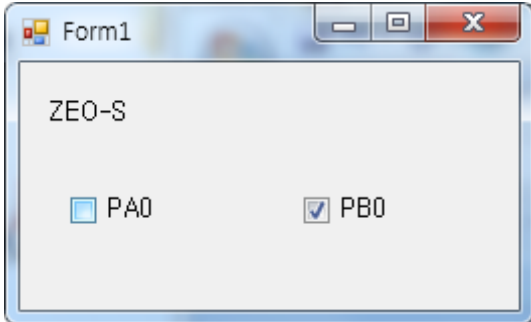
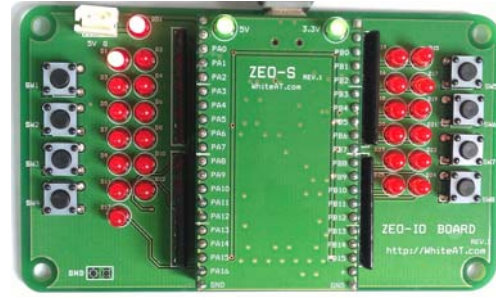
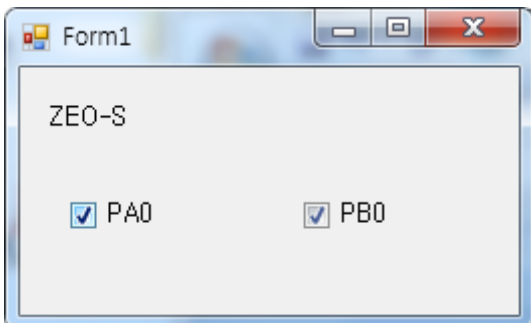
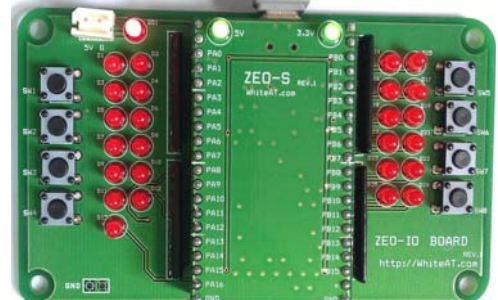
                this.ZEO.Pin_Set( this.ZEO.PIN_ALL );
                // 모든 핀을 출력으로 설정
                this.ZEO.PORT_DirOutputALL();
            }

            private void chbPA0_CheckedChanged(object sender, EventArgs e)
            {
                this.ZEO.Pin_Write(PIN_NAME.PA0, this.chbPA0.Checked);
            }

            private void chbPB0_CheckedChanged(object sender, EventArgs e)
            {
                this.ZEO.Pin_Write(PIN_NAME.PB0, this.chbPB0.Checked);
            }

            private void Form1_FormClosing(object sender, FormClosingEventArgs e)
            {
                this.ZEO.Close();
            }
        }
    }
}
```

2.5. 결과

프로그램	결과
	 < 모든 LED ON >
	 < 좌측 LED OFF, 우측 LED ON >
	 < 좌측 LED ON, 우측 LED OFF >
	 < 모든 LED OFF >

3. C++ 으로 시작

이 단원에서는 Visual C++ 6.0 사용하였으며 그 이상 버전이면 상관 없습니다.

3.1. 필요한 파일

C++ 으로 프로그래밍하려면 3개의 라이브러리와 3개의 헤더 파일이 필요합니다. 새로운 프로젝트를 생성할 때 다운 받은 라이브러리를 해당 폴더로 복사해서 사용해야 하니 바탕화면이나 원하는 위치에 [다운로드](#)⁴⁾합니다.

파일명	용도
LibUsb.dll	기본적인 USB 통신을 위한 라이브러리
ZEO_lib_vc60.DLL	ZEO 모듈 통신용 라이브러리
ZEO_lib_vc60.LIB	ZEO 모듈 통신용 라이브러리

< 필요한 라이브러리 >

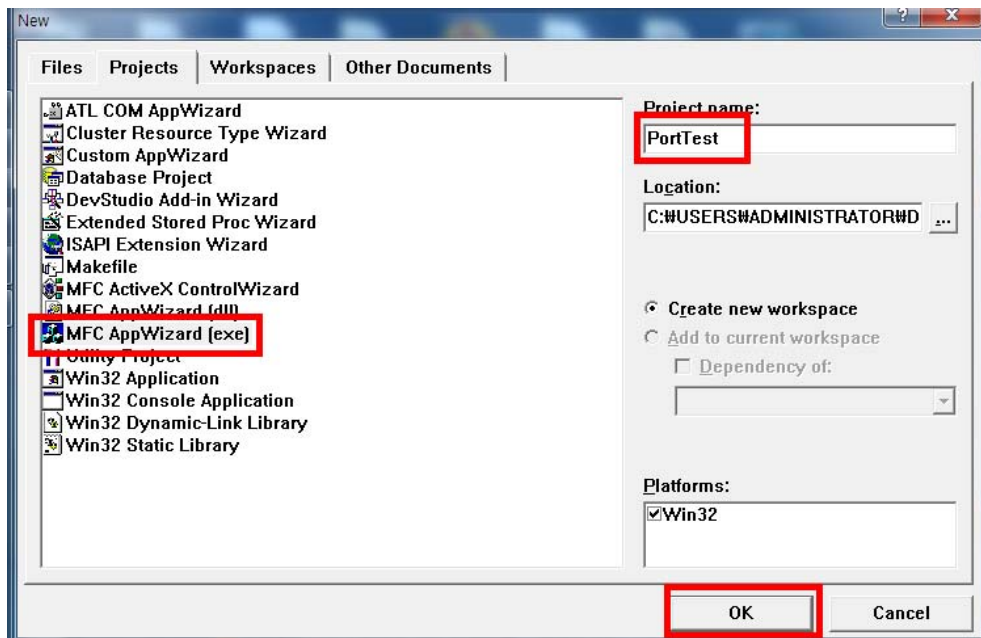
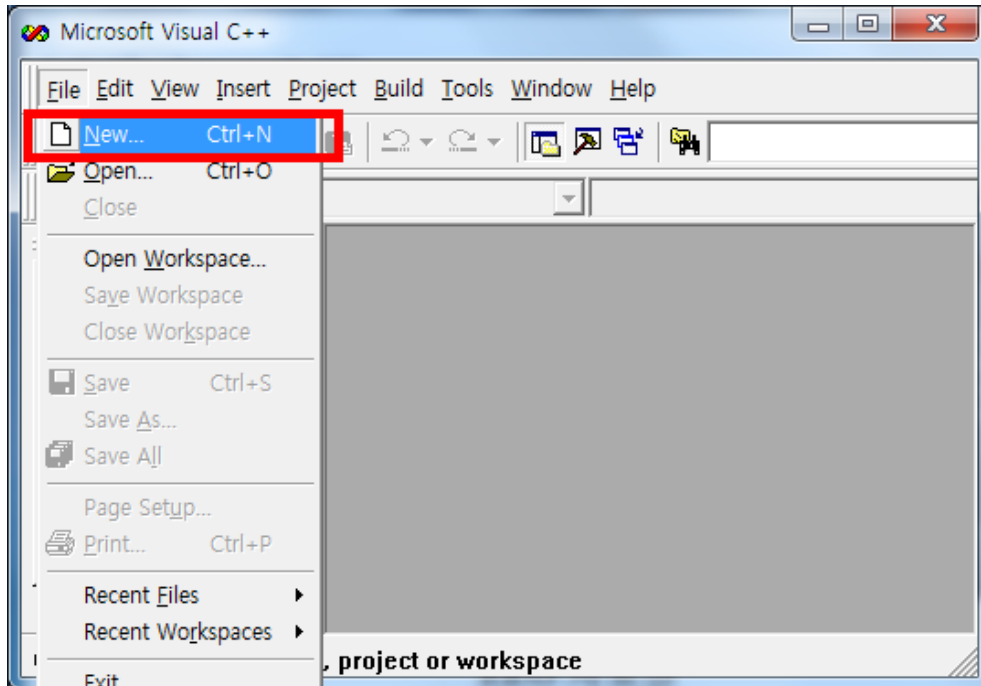
파일명	용도
usb.h	기본적인 USB 통신을 위한 헤더 파일
ZEOLib.h	ZEO 모듈 통신용 헤더 파일
define.h	ZEO 모듈의 기능 정의

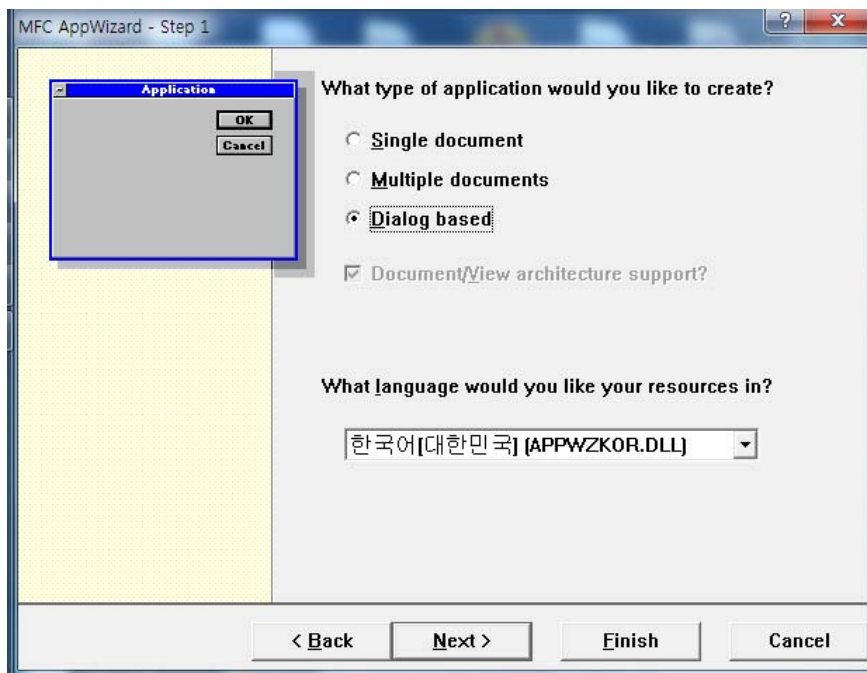
< 필요한 헤더 파일 >

4) 라이브러리 다운로드 경로: <http://whiteat.com/57501>

3.2. 프로젝트 생성

새로운 프로젝트를 생성합니다. 프로젝트명을 PortTest로 진행하겠습니다.





앞에서 다운로드한 6개의 파일을 추가해야 합니다. 프로젝트 폴더를 열어서 다운 받은 6개의 파일을 프로젝트 파일이 있는 폴더에 복사해야 합니다.

이름	유형	수정한 날짜	크기
Debug	파일 폴더	2014-06-01 오후...	
res	파일 폴더	2014-06-01 오후...	
libusb.lib	C compiler library...	2007-03-20 오전...	11KB
usb.h	C compiler heade...	2007-03-20 오전...	11KB
define.h	C compiler heade...	2014-05-28 오후...	3KB
ZEODll_VC.lib	C compiler library...	2014-05-29 오전...	13KB
ZEODll_VC.dll	응용 프로그램 확장	2014-05-30 오후...	59KB
ZEOLib.h	C compiler heade...	2014-05-30 오후...	4KB
ZEOSPWM11Dlg.cpp	C++ Source	2014-05-30 오후...	8KB

그리고 다음과 같이 헤더 파일과 라이브러리를 추가하고 ZEOLib를 객체와 ZEO 핸들러를 선언합니다.

```
#include "stdafx.h"
#include "ZEOSPWM11.h"
#include "ZEOSPWM11Dlg.h"

#ifdef _DEBUG
#define new DEBUG_NEW
#undef THIS_FILE
static char THIS_FILE[] = __FILE__;
#endif

#include "ZEOLib.h"           // header file
#pragma comment(lib, "ZEODll_VC.lib") // library

ZEODll::ZEOLib g_ZEO;        // ZEO object
usb_dev_handle *m_hDEV;      // ZEO handle
```

다이얼로그 상태에서 프로그램 시작시 실행되는 메소드에 ZEO-S 초기화와 모든 포트를 출력으로 설정합니다.

```
// TODO: Add extra initialization here
g_ZEO.Inital();
m_hDEV = q_ZEO.Open();

if(!(m_hDEV )){
    MessageBox("ZEO-S 연결을 확인해 주세요.\n","연결 실패",
    m_strVersion = "ZEO-S와 연결되지 않아 진행할 없습니다";
    UpdateData(FALSE);
    return 0;
}

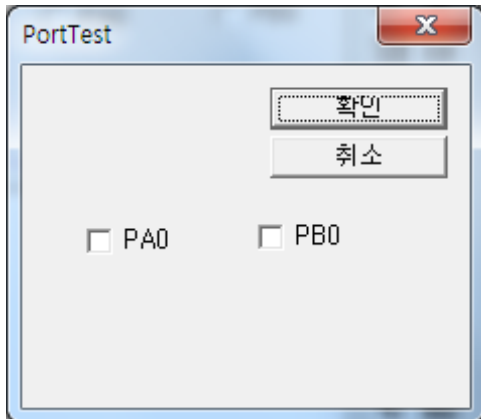
if(ZEO_OK != g_ZEO.RequestBoardInfo())
{
    MessageBox("연결 실패");
    return TRUE;
}
```

```
g_ZEO.Pin_DirOutput(PORT_PA_ALL);
g_ZEO.Pin_DirOutput(PORT_PB_ALL);
```

프로그램 종료 메소드에 Close()를 호출해 줍니다.

```
void CPortTestDlg::OnSysCommand(UINT nID, LPARAM lParam)
{
    if ((nID & 0xFFF0) == IDM_ABOUTBOX)
    {
        CAboutDlg dlgAbout;
        dlgAbout.DoModal();
    }
    else
    {
        if( nID==SC_CLOSE)
        {
            g_ZEO.Close();
        }
        CDialog::OnSysCommand(nID, lParam);
    }
}
```

핀 ON/OFF 에 사용할 체크박스를 만들고 각각의 멤버함수를 만듭니다.



```
void CPortTestDlg::OnCheck1()
{
    // TODO: Add your control notification handler code here
}

void CPortTestDlg::OnCheck2()
{
    // TODO: Add your control notification handler code here
}
```

< 멤버 함수 생성 >

```
void CPortTestDlg::OnCheck1()
{
    UpdateData(TRUE);
    if(m_bPA0) g_ZEO.Pin_Set(PORT_PA0);
    else g_ZEO.Pin_Reset(PORT_PA0);
}

void CPortTestDlg::OnCheck2()
{
    UpdateData(TRUE);
    if(m_bPB0) g_ZEO.Pin_Set(PORT_PB0);
    else g_ZEO.Pin_Reset(PORT_PB0);
}
```

< 상세 코드 추가 >

3.3. 소스코드

```
// PortTestDlg.cpp : implementation file
//

#include "stdafx.h"
#include "PortTest.h"
#include "PortTestDlg.h"

#ifdef _DEBUG
#define new DEBUG_NEW
#undef THIS_FILE
static char THIS_FILE[] = __FILE__;
#endif

#include "ZEOLib.h" // header file
#pragma comment(lib, "ZEODll_VC.lib") // library

ZEODll::ZEOLib g_ZEO; // ZEO object
usb_dev_handle *m_hDEV; // ZEO handle

////////////////////////////////////
// CAboutDlg dialog used for App About

class CAboutDlg : public CDialog
{
public:
    CAboutDlg();

    // Dialog Data
   //{{AFX_DATA(CAboutDlg)
    enum { IDD = IDD_ABOUTBOX };
    //}}AFX_DATA

    // ClassWizard generated virtual function overrides
   //{{AFX_VIRTUAL(CAboutDlg)
protected:
    virtual void DoDataExchange(CDataExchange* pDX); // DDX/DDV support
    //}}AFX_VIRTUAL

    // Implementation
protected:
   //{{AFX_MSG(CAboutDlg)
    //}}AFX_MSG
    DECLARE_MESSAGE_MAP()
};

CAboutDlg::CAboutDlg() : CDialog(CAboutDlg::IDD)
{
    //{{AFX_DATA_INIT(CAboutDlg)
    //}}AFX_DATA_INIT
}

void CAboutDlg::DoDataExchange(CDataExchange* pDX)
{
    CDialog::DoDataExchange(pDX);
    //{{AFX_DATA_MAP(CAboutDlg)
    //}}AFX_DATA_MAP
}
```

```
BEGIN_MESSAGE_MAP(CAboutDlg, CDialog)
//{{AFX_MSG_MAP(CAboutDlg)
// No message handlers
//}}AFX_MSG_MAP
END_MESSAGE_MAP()

////////////////////////////////////
// CPortTestDlg dialog

CPortTestDlg::CPortTestDlg(CWnd* pParent /*=NULL*/)
: CDialog(CPortTestDlg::IDD, pParent)
{
    //{{AFX_DATA_INIT(CPortTestDlg)
    m_bPA0 = FALSE;
    m_bPB0 = FALSE;
    //}}AFX_DATA_INIT
    // Note that LoadIcon does not require a subsequent DestroyIcon in Win32
    m_hIcon = AfxGetApp()->LoadIcon(IDR_MAINFRAME);
}

void CPortTestDlg::DoDataExchange(CDataExchange* pDX)
{
    CDialog::DoDataExchange(pDX);
    //{{AFX_DATA_MAP(CPortTestDlg)
    DDX_Check(pDX, IDC_CHECK1, m_bPA0);
    DDX_Check(pDX, IDC_CHECK2, m_bPB0);
    //}}AFX_DATA_MAP
}

BEGIN_MESSAGE_MAP(CPortTestDlg, CDialog)
//{{AFX_MSG_MAP(CPortTestDlg)
ON_WM_SYSCOMMAND()
ON_WM_PAINT()
ON_WM_QUERYDRAGICON()
ON_BN_CLICKED(IDC_CHECK1, OnCheck1)
ON_BN_CLICKED(IDC_CHECK2, OnCheck2)
//}}AFX_MSG_MAP
END_MESSAGE_MAP()
```



```
BOOL CPortTestDlg::OnInitDialog()
{
    CDialog::OnInitDialog();

    // IDM_ABOUTBOX must be in the system command range.
    ASSERT((IDM_ABOUTBOX & 0xFFF0) == IDM_ABOUTBOX);
    ASSERT(IDM_ABOUTBOX < 0xF000);

    CMenu* pSysMenu = GetSystemMenu(FALSE);
    if (pSysMenu != NULL)
    {
        CString strAboutMenu;
        strAboutMenu.LoadString(IDS_ABOUTBOX);
        if (!strAboutMenu.IsEmpty())
        {
            pSysMenu->AppendMenu(MF_SEPARATOR);
            pSysMenu->AppendMenu(MF_STRING, IDM_ABOUTBOX, strAboutMenu);
        }
    }

    // Set the icon for this dialog. The framework does this automatically
    // when the application's main window is not a dialog
    SetIcon(m_hIcon, TRUE); // Set big icon
    SetIcon(m_hIcon, FALSE); // Set small icon

    // TODO: Add extra initialization here
    g_ZEO.Initial();
    g_ZEO.Open();
    if(ZEO_OK != g_ZEO.RequestBoardInfo())
    {
        MessageBox("연결 실패");
        return TRUE;
    }

    if(ZEO_OK != g_ZEO.InitZeo(0))
    {
        MessageBox("Init 실패");
        return TRUE;
    }

    g_ZEO.Pin_DirOutput(PORT_PA_ALL);
    g_ZEO.Pin_DirOutput(PORT_PB_ALL);

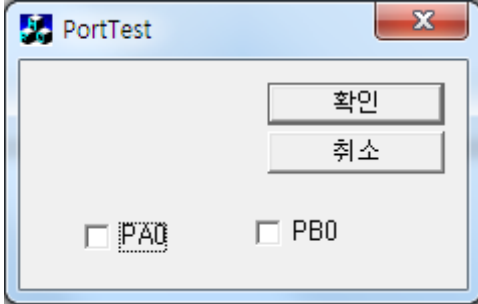
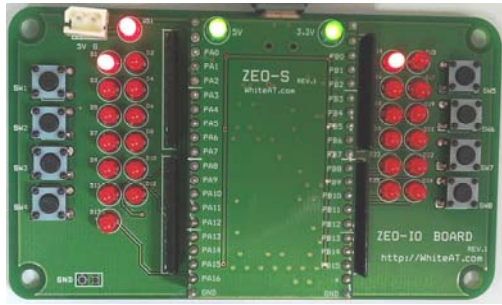
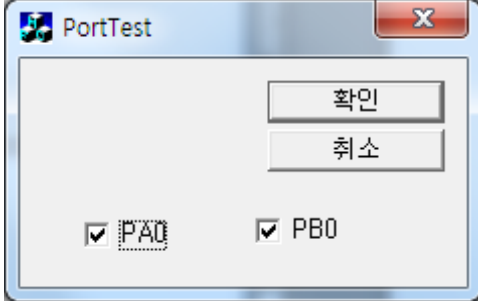
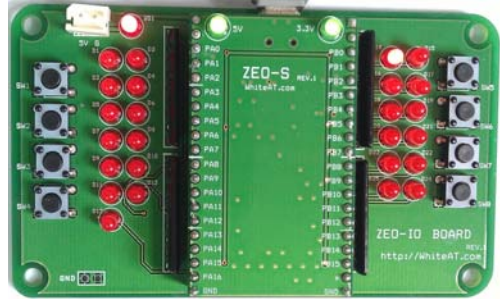
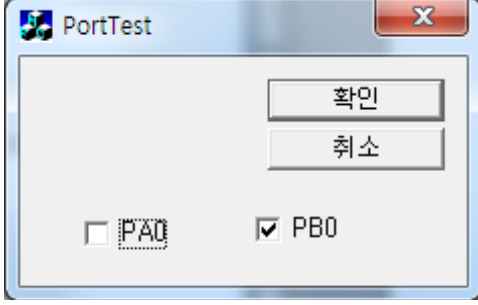
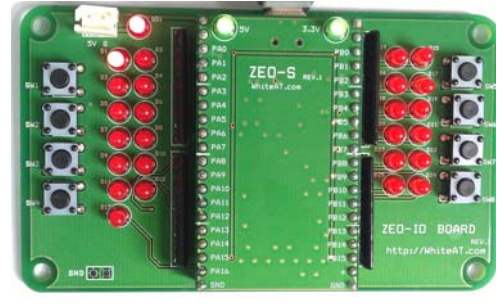
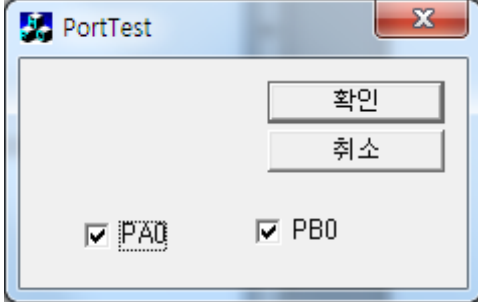
    return TRUE; // return TRUE unless you set the focus to a control
}
```

```
void CPortTestDlg::OnSysCommand(UINT nID, LPARAM lParam)
{
    if ((nID & 0xFFFF) == IDM_ABOUTBOX)
    {
        CAboutDlg dlgAbout;
        dlgAbout.DoModal();
    }
    else
    {
        if( nID==SC_CLOSE)
        {
            g_ZEO.Close();
        }
        CDialog::OnSysCommand(nID, lParam);
    }
}

void CPortTestDlg::OnCheck1()
{
    UpdateData(TRUE);
    if(m_bPA0) g_ZEO.Pin_Set(PORT_PA0);
    else g_ZEO.Pin_Reset(PORT_PA0);
}

void CPortTestDlg::OnCheck2()
{
    UpdateData(TRUE);
    if(m_bPB0) g_ZEO.Pin_Set(PORT_PB0);
    else g_ZEO.Pin_Reset(PORT_PB0);
}
```

3.4. 결과

프로그램	결과
	 < 모든 LED ON >
	 < 좌측 LED OFF, 우측 LED ON >
	 < 좌측 LED ON, 우측 LED OFF >
	 < 모든 LED OFF >

더 많은 자료와 업데이트된 소스는 홈페이지(<http://WhiteAT.com>)에서 확인 및 다운로드 가능합니다.

화이트앳 (WhiteAT)

Homepage: <http://WhiteAT.com>

E-MAIL: whiteat@whiteat.com

ADDRESS: 서울시 용산구 청파동1가 183번지 대산빌딩 201호

ROOM 201, DAESAN Building, Cheongpa-dong 1-ga Yongsan-gu Seoul