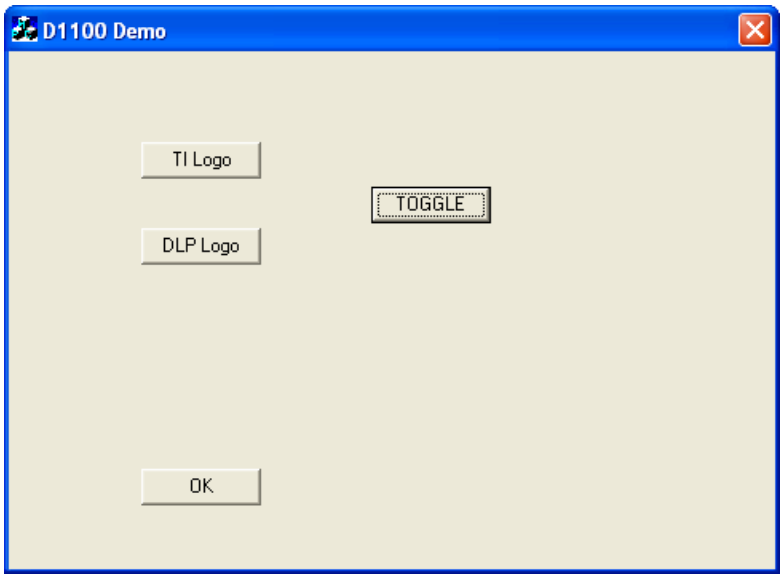
# DMD Discovery™ 1100 Visual C++ 6.0 2 DMD


## ABSTRACT

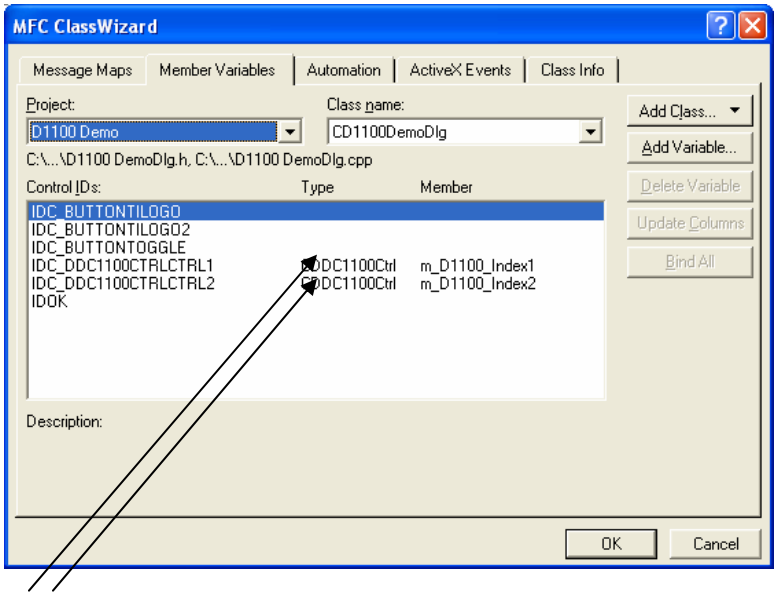
Programming sample for control of the DMD Discovery 1100 (D1100) two DMDs is provided written in Microsoft Visual C++ 6.0. The sample demonstrates the use of the Discovery ActiveX control in performing common control functions.


## 1.1 2 DMD Visual C++ ActiveX Sample Code

This sample demonstrates the use of multiple instances of the D1100 ActiveX control by controlling two D1100 boards. Source code is available in the “2 D1100 VC ActiveX Sample Code” folder in the D1100Sample.zip file. The sample opens communication to the DMD boards and displays images as selected by three control buttons.




This sample uses two instances of the CDDC1100Ctrl class to control two D1100 devices. The two instances are created using Classwizard as shown :




Two Class Instances

Additional D1100 devices could be added by adding more class instances. Source file D1100 DemoDlg.cpp contains the sample code shown below. Message handler OnInitDialog opens the D1100 devices using OpenD1100 and allocates image storage.

```
BOOL CD1100DemoDlg::OnInitDialog()
{
	CDialog::OnInitDialog();

	// 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

	OpenD1100(1); //open D1100 #1 USB
	OpenD1100(2); //open D1100 #2 USB

	if (m_pByteArray)
		{
		delete m_pByteArray;
		m_pByteArray = NULL;
		}
	m_pByteArray = new CByteArray;		// create image storage byte array
	m_pByteArray->SetSize (1024 * 768 / 8);
	m_pFrameBuffer=new COleVariant (*m_pByteArray);	//convert to variant
pointer

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

BOOL CD1100DemoDlg::OpenD1100 (int DeviceIndex)
```

```
{
   BOOL bRtn = TRUE;
        CDDC1100Ctrl *m_pDDCControl;

        if (DeviceIndex==1)
                m_pDDCControl=&m_D1100_Index1;    // use instance 1
        else if (DeviceIndex==2)
                m_pDDCControl=&m_D1100_Index2;    // use instance 2
        else
                {
                AfxMessageBox (_T("Invalid Device Index"));
                return FALSE;
                }

        // Initialize device
   if (!m_pDDCControl->GetDeviceMulti(DeviceIndex))
   {
      AfxMessageBox (_T("Error communicating with the Discovery 1100 "));
      bRtn = FALSE;
  }
   else
   {
      m_pDDCControl->EnableParallel (0);
   }


   return (bRtn);
}
```

Message handler OnButtontilogo or OnButtonDLPogo is called when a button is pressed. The ShowImage function is called to display the selected image on the devices.

```
void CD1100DemoDlg::OnButtontilogo()
{
        ShowImage(1,"TILOGO.bmp");  //instance 1
        ShowImage(2,"TILOGO.bmp");  //instance 2
}

void CD1100DemoDlg::OnButtonDLPLogo()
{
        ShowImage(1,"DLPLOGO.bmp");         //instance 1
        ShowImage(2,"DLPLOGO.bmp");         //instance 2
}

BOOL CD1100DemoDlg::ShowImage(int DeviceIndex,LPCTSTR lpFilename)
{
        CDDC1100Ctrl *m_pDDCControl;

        if (DeviceIndex==1)
                m_pDDCControl=&m_D1100_Index1;    // use instance 1
        else if (DeviceIndex==2)
                m_pDDCControl=&m_D1100_Index2;    // use instance 2
        else
                {
```

```
                AfxMessageBox (_T("Invalid Device Index"));
                return FALSE;
                }


        if (!m_pDDCControl->IsDeviceAttached ())
        {
                AfxMessageBox (_T("Error communicating with the Discovery 1100"));
                return FALSE;
        }
        else
        {
                // load image into memory buffer m_pFrameBuffer in DMD compatible
format
                m_pDDCControl-
>LoadImageFileToBuffer(lpFilename,m_pFrameBuffer,0);
                // transfer image from memory buffer to ActiveX control buffer
                m_pDDCControl->LoadFrameBuffer(m_pFrameBuffer);
                // load and reset to display the image
                m_pDDCControl->LoadResetFrame ();   // display the image
        }

        return TRUE;

}
```

When the “TI Logo/Checker Toggle” control button is clicked the image is displayed by the button event handling subroutine. The image displayed on each device is toggled.

```
void CD1100DemoDlg::OnButtontoggle()
{
        toggle++;         //inc toggle counter
        toggle%=2;

        if (toggle)
                {
                ShowImage(1,"TILOGO.bmp");
                ShowImage(2,"DLPLOGO.bmp");
                }
        else
                {
                ShowImage(1,"DLPLOGO.bmp");
                ShowImage(2,"TILOGO.bmp");
                }

}
```