ServoCenter3.1 DLL Control Examples – Visual C++ .NET

1. Overview

ServoCenter 3.1 comes packaged with the yeisrvo.dll runtime library, which gives programmers access to low-level predefined functions that can be used with the ServoCenter 3.1 controller board. This document covers the capabilities of the DLL, installing the DLL, and writing programs using the DLL functions.

2. ServoCenter3.1 DLL Functional Overview

The functions provided by the ServoCenter 3.1 DLL correspond with the ServoCenter 3.1 controller board commands detailed in Section 4.1.4, except as noted in the table descriptions listed below.

Function:	void InitPort(int Comm, long BaudRate)
Parameters:	Comm: Communications Port to which ServoCenter 3.1 controller is attached. BaudRate: Data rate at which Port will communicate.
Return Value:	0 – Success Other - Error

Function:	void QuickMove(int Comm, int BoardNum, int ServoNum, int ServoPos)
Parameters:	Comm: Communications Port to which ServoCenter 3.1 controller is attached. BoardNum: ID (0~15) of ServoCenter 3.1 Board to which the command will be sent. ServoNum: ID (0~15) of servo whose position is to be changed. ServoPos: Raw position (0~200) to which servo will be moved.
Return Value:	None

Function:	void ScaledQuickMove(int Comm, int BoardNum, int ServoNum, int ServoPos)
Parameters:	Comm: Communications Port to which ServoCenter 3.1 controller is attached. BoardNum: ID (0~15) of ServoCenter 3.1 Board to which the command will be sent. ServoNum: ID (0~15) of servo whose position is to be changed. ServoPos: Scaled position (0~100) to which servo will be moved.
Return Value:	None

Function:	void ServoEnable(int Comm, int BoardNum, int ServoNum)
Parameters:	Comm: Communications Port to which ServoCenter 3.1 controller is attached. BoardNum: ID (0~15) of ServoCenter 3.1 Board to which the command will be sent. ServoNum: ID (0~15) of servo to be enabled.
Return Value:	None

Function:	void ServoDisable(int Comm, int BoardNum, int ServoNum)
Parameters:	Comm: Communications Port to which ServoCenter 3.1 controller is attached. BoardNum: ID (0~15) of ServoCenter 3.1 Board to which the command will be sent. ServoNum: ID (0~15) of servo to be disabled.
Return Value:	None

Function:	void SetMin(int Comm, int BoardNum, int ServoNum, int ServoMinPos)
Parameters:	Comm: Communications Port to which ServoCenter 3.1 controller is attached. BoardNum: ID (0~15) of ServoCenter 3.1 Board to which the command will be sent. ServoNum: ID (0~15) of servo for which Min is being set. ServoPos: Minimum value to be set.
Return Value:	None

Function:	void SetMax(int Comm, int BoardNum, int ServoNum, int ServoMaxPos)
Parameters:	Comm: Communications Port to which ServoCenter 3.1 controller is attached. BoardNum: ID (0~15) of ServoCenter 3.1 Board to which the command will be sent. ServoNum: ID (0~15) of servo for which Max is being set. ServoPos: Maximum value to be set.
Return Value:	None

Function:	void SetStart(int Comm, int BoardNum, int ServoNum, int ServoStartPos)
Parameters:	Comm: Communications Port to which ServoCenter 3.1 controller is attached. BoardNum: ID (0~15) of ServoCenter 3.1 Board to which the command will be sent. ServoNum: ID (0~15) of servo for which Start is being set. ServoPos: Start value to be set.
Return Value:	None

Function:	void SetMaxSpeed(int Comm, int BoardNum, int ServoNum, int ServoSpeed)
Parameters:	Comm: Communications Port to which ServoCenter 3.1 controller is attached. BoardNum: ID (0~15) of ServoCenter 3.1 Board to which the command will be sent. ServoNum: ID (0~15) of servo for which Speed is being set. ServoSpeed: Maximum Speed value to be set.
Return Value:	None

Function:	void SetMinCurrent(int Comm, int BoardNum, int ServoNum)
Parameters:	Comm: Communications Port to which ServoCenter 3.1 controller is attached. BoardNum: ID (0~15) of ServoCenter 3.1 Board to which the command will be sent. ServoNum: ID (0~15) of servo for which Min is being set.
Return Value:	None

Function:	void SetMaxCurrent(int Comm, int BoardNum, int ServoNum)
Parameters:	Comm: Communications Port to which ServoCenter 3.1 controller is attached. BoardNum: ID (0~15) of ServoCenter 3.1 Board to which the command will be sent. ServoNum: ID (0~15) of servo for which Max is being set.
Return Value:	None

Function:	void SetStartCurrent(int Comm, int BoardNum, int ServoNum)
Parameters:	Comm: Communications Port to which ServoCenter 3.1 controller is attached. BoardNum: ID (0~15) of ServoCenter 3.1 Board to which the command will be sent. ServoNum: ID (0~15) of servo for which Start is being set.
Return Value:	None

Function:	int GetCurrentPos(int Comm, int BoardNum, int ServoNum)
Parameters:	Comm: Communications Port to which ServoCenter 3.1 controller is attached. BoardNum: ID (0~15) of ServoCenter 3.1 Board to which the command will be sent. ServoNum: ID (0~15) of servo for which position is being queried.
Return Value:	Position of servo (0~200)

Function:	int GetMin(int Comm, int BoardNum, int ServoNum)
Parameters:	Comm: Communications Port to which ServoCenter 3.1 controller is attached. BoardNum: ID (0~15) of ServoCenter 3.1 Board to which the command will be sent. ServoNum: ID (0~15) of servo for which Min is being queried.
Return Value:	Minimum position of servo (0~200)

Function:	Int GetMax(int Comm, int BoardNum, int ServoNum)
Parameters:	Comm: Communications Port to which ServoCenter 3.1 controller is attached. BoardNum: ID (0~15) of ServoCenter 3.1 Board to which the command will be sent. ServoNum: ID (0~15) of servo for which Max is being queried.
Return Value:	Maximum position of servo (0~200)

Function:	int GetStart(int Comm, int BoardNum, int ServoNum)
Parameters:	Comm: Communications Port to which ServoCenter 3.1 controller is attached. BoardNum: ID (0~15) of ServoCenter 3.1 Board to which the command will be sent. ServoNum: ID (0~15) of servo for which start position is being queried.
Return Value:	Start position of servo (0~200)

Function:	int GetMaxSpeed(int Comm, int BoardNum, int ServoNum)
Parameters:	Comm: Communications Port to which ServoCenter 3.1 controller is attached. BoardNum: ID (0~15) of ServoCenter 3.1 Board to which the command will be sent. ServoNum: ID (0~15) of servo for which max speed is being queried.
Return Value:	Maximum speed of servo(0~100)

Function:	void MoveRaw(int Comm, int BoardNum, int ServoNum, int ServoPos, int ServoSpeed)
Parameters:	Comm: Communications Port to which ServoCenter 3.1 controller is attached. BoardNum: ID (0~15) of ServoCenter 3.1 Board to which the command will be sent. ServoNum: ID (0~15) of servo to be moved. ServoPos: Position (0~200) to which servo will be moved. ServoSpeed: Speed (0~100) at which servo will move.
Return Value:	None

Function:	void MoveRawCW(int Comm, int BoardNum, int ServoNum, int ServoPos, int ServoSpeed)
Parameters:	Comm: Communications Port to which ServoCenter 3.1 controller is attached. BoardNum: ID (0~15) of ServoCenter 3.1 Board to which the command will be sent. ServoNum: ID (0~15) of servo to be moved. ServoPos: Units (0~200) servo will be moved. ServoSpeed: Speed (0~100) at which servo will move.
Return Value:	None

Function:	void MoveRawCCW(int Comm, int BoardNum, int ServoNum, int ServoPos, int ServoSpeed)
Parameters:	Comm: Communications Port to which ServoCenter 3.1 controller is attached. BoardNum: ID (0~15) of ServoCenter 3.1 Board to which the command will be sent. ServoNum: ID (0~15) of servo to be moved. ServoPos: Units (0~200) servo will be moved. ServoSpeed: Speed (0~100) at which servo will move.
Return Value:	None

Function:	void MoveScaled(int Comm, int BoardNum, int ServoNum, int ServoPos, int ServoSpeed)
Parameters:	Comm: Communications Port to which ServoCenter 3.1 controller is attached. BoardNum: ID (0~15) of ServoCenter 3.1 Board to which the command will be sent. ServoNum: ID (0~15) of servo to be moved. ServoPos: Scaled position (0~100) to which servo will be moved. ServoSpeed: Speed (0~100) at which servo will move.
Return Value:	None

Function:	void MoveScaledCW(int Comm, int BoardNum, int ServoNum, int ServoPos, int ServoSpeed)
Parameters:	Comm: Communications Port to which ServoCenter 3.1 controller is attached. BoardNum: ID (0~15) of ServoCenter 3.1 Board to which the command will be sent. ServoNum: ID (0~15) of servo to be moved. ServoPos: Scaled position (0~100) to which servo will be moved. ServoSpeed: Speed (0~100) at which servo will move.
Return Value:	None

Function:	void MoveScaledCCW(int Comm, int BoardNum, int ServoNum, int ServoPos, int ServoSpeed)
Parameters:	Comm: Communications Port to which ServoCenter 3.1 controller is attached. BoardNum: ID (0~15) of ServoCenter 3.1 Board to which the command will be sent. ServoNum: ID (0~15) of servo to be moved. ServoPos: Scaled position (0~100) to which servo will be moved. ServoSpeed: Speed (0~100) at which servo will move.
Return Value:	None

Function:	void SetPulseWidthMin(int Comm, int BoardNum, int WidthVal)
Parameters:	Comm: Communications Port to which ServoCenter 3.1 controller is attached. BoardNum: ID (0~15) of ServoCenter 3.1 Board to which the command will be sent. WidthVal: Minimum (1~239) width of servo control pulses.
Return Value:	None

Function:	void SetPulseWidthMax(int Comm, int BoardNum, int WidthVal)
Parameters:	Comm: Communications Port to which ServoCenter 3.1 controller is attached. BoardNum: ID (0~15) of ServoCenter 3.1 Board to which the command will be sent. WidthVal: Maximum (1~239) width of servo control pulses.
Return Value:	None

Function:	void InvertServo(int Comm, int BoardNum, int ServoNum)
Parameters:	Comm: Communications Port to which ServoCenter 3.1 controller is attached. BoardNum: ID (0~15) of ServoCenter 3.1 Board to which the command will be sent. ServoNum: ID (0~15) of servo to be inverted.
Return Value:	None

Function:	void NormalServo(int Comm, int BoardNum, int ServoNum)
Parameters:	Comm: Communications Port to which ServoCenter 3.1 controller is attached. BoardNum: ID (0~15) of ServoCenter 3.1 Board to which the command will be sent. ServoNum: ID (0~15) of servo to be normalized.
Return Value:	None

Function:	int GetMaxSettingsLen()
Parameters:	None
Return Value:	Maximum length of settings string obtained with GetSettings()

Function:	int GetSettings (int Comm, int BoardNum, char * SettingsInfo)
Parameters:	Comm: Communications Port to which ServoCenter 3.1 controller is attached. BoardNum: ID (0~15) of ServoCenter 3.1 Board to which the command will be sent. SettingsInfo: Buffer to hold settings information.
Return Value:	Length of SettingsInfo string

Function:	void CommitSettings(int Comm, int BoardNum)
Parameters:	Comm: Communications Port to which ServoCenter 3.1 controller is attached. BoardNum: ID (0~15) of ServoCenter 3.1 Board to which the command will be sent.
Return Value:	None

Function:	void RestoreFactorySettings(int Comm, int BoardNum)
Parameters:	Comm: Communications Port to which ServoCenter 3.1 controller is attached. BoardNum: ID (0~15) of ServoCenter 3.1 Board to which the command will be sent.
Return Value:	None

Function:	void ResetAsStartup(int Comm, int BoardNum)
Parameters:	Comm: Communications Port to which ServoCenter 3.1 controller is attached. BoardNum: ID (0~15) of ServoCenter 3.1 Board to which the command will be sent.
Return Value:	None

Function:	int GetMaxVersionLen()
Parameters:	None
Return Value:	Maximum length of version string obtained with GetSCVersion()

Function:	int GetSCVersion(int Comm, int BoardNum, char * VerInfo)
Parameters:	Comm: Communications Port to which ServoCenter 3.1 controller is attached. BoardNum: ID (0~15) of ServoCenter 3.1 Board to which the command will be sent. VerInfo: Buffer to hold version information.
Return Value:	Length of VerInfo string

Function:	void CloseCom(int Comm)
Parameters:	Comm: Communications Port to be closed.
Return Value:	None

Function:	void CloseAllComs()
Parameters:	None
Return Value:	None

3. Installing the yeisrvo.dll Runtime Library

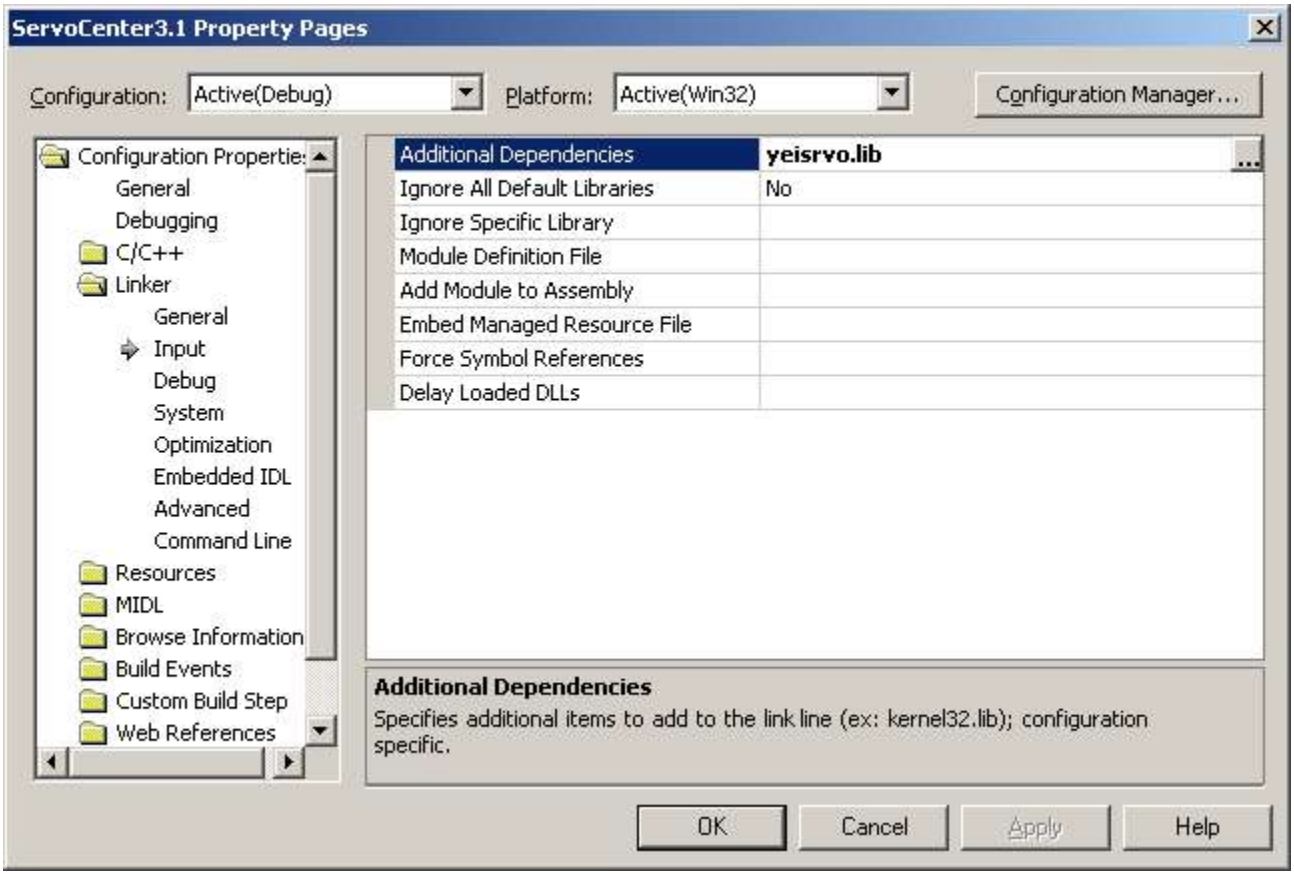
For your programs to be able to use the ServoCenter 3.1 DLL, the DLL must first be placed somewhere that your running program will be able to find it. The best location for your DLL is in the same directory as your running program. For example, if your program is found in '`C:\ServoCenter\program\`', copy the ServoCenter 3.1 DLL to that directory.

Another location where you can store the ServoCenter 3.1 DLL is the folder where Windows stores the system-wide runtime libraries. In Windows 95 and 98, this folder is '`C:\WINDOWS\system\`.' In Windows ME, 2000, and XP, the folder is '`C:\WINDOWS\system32\`.'

Once the ServoCenter 3.1 DLL has been copied to one of these directories, you are ready to begin writing programs that use it.

4. Programming with the yeisrvo.dll Runtime Library in Visual C++ .NET

Programming with the yeisrvo.dll runtime library in Visual C++ .NET is very similar to programming in Visual C++ 6.0. The only real difference is the process by which the project settings are modified to use the yeisrvo.lib import library. For Visual C++ .NET, the default library directory is '`C:\Program Files\Microsoft Visual Studio .NET\Vc7\lib.`' To add yeisrvo.lib to the list of libraries to link, right-click on the name of the project in the Solutions Explorer, then select **Properties**. Select **Linker** from the **Property Pages** dialog that appears, and then click **Input**. Next add yeisrvo.lib to the **Additional Dependencies** and click **OK** to apply the settings. The illustration below shows a properly modified Properties Dialog:



Once the yeisrvo.lib import library has been added to your project, you may follow the instructions provided in Section 4.4.4 of the ServoCenter 3.1 manual to program the ServoCenter 3.1 controller board.

6. Additional Information

Sample code is available in the ServoCenterCppNETDll.vcproj file in the Examples directory of the ServoCenter3.1 CD or online at www.YostEngineering.com/ServoCenter.