ASCOM VB Drivers Development Framework Checklist

The following is a checklist and an annotated walk-through example of generating an ASCOM VB Driver for a fictitious company/product called Acme. The end result is a tested "working", "non-functional" ASCOM Focuser driver skeleton framework.

- - "working" means the driver passes the Conformance Checker tool
- - "non-functional" means it controls no actual hardware (yet)

Development Environment

- 64-bit Windows 7
- Visual Basic 2010 Express Edition
- ASCOM Platform 6.1SP1 installed
- ASCOM Platform Developer Components installed
- ASCOM Driver Conformance Checker installed

Note: All projects in a multi-project solution in Express Editions of Visual Studio must use the same programming language.

Visual Basic 2010 Express ← Run as administrator !!!

	Checklist	Image
☐	1) New Project - **ASCOM Device Driver (VB)** - Name: **Acme**	NewProject.png
☐	2) ASCOM Driver Project Wizard - Device Class - **Focuser** - Device Name/Model - **Acme** - Create	WizardFocuser.png
☐	3) File➢Save All – - Name: **Focuser** - Location: ...**VB Projects\ASCOM** - Solution Name: **Acme** ← **MAKE SURE**!!! because Visual Studio makes same - [✓] Create directory for solution as Name!!!	SaveProject.png
☐	4) Set Focuser Properties➢Application➢Assembly name: **ASCOM.Acme.Focuser** - Save project	FocuserAssemblyName.png
☐	5) Enable 32 and 64-bit code generation for the Focuser driver by modifying (outside of the Visual Basic IDE environment) the <PlatformTarget> tag in the Focuser driver's Focuser.vbproj file to be **AnyCPU** located under the following tag: (it was "x86"!!!) <PropertyGroup Condition=" '$(Configuration)\|$(Platform)' == 'Debug|AnyCPU' "> (i.e. in Focuser.vbproj: <PlatformTarget>**AnyCPU**</PlatformTarget>)	
☐	6) Enable XML documentation for the Focuser driver by modifying (outside of the Visual Basic IDE environment) the empty <DocumentationFile> tags in the Focuser driver's Focuser.vbproj file to be the project's name **Focuser.xml** located under the following tags: <PropertyGroup Condition=" '$(Configuration)\|$(Platform)' == 'Debug|AnyCPU' "> <PropertyGroup Condition=" '$(Configuration)\|$(Platform)' == 'Release|AnyCPU' "> (i.e. in Focuser.vbproj: <DocumentationFile>**Focuser.xml**</DocumentationFile>) (no "Generate XML documentation file" option in Visual Basic 2010 Express Edition's Properties➢Compile) A prompt will be given to reload the Focuser project due to the changed project file.	ReloadProject.png
☐	7) Build the project	

☐ 8) Run the ASCOM **Conform** tool. If it is running in 64 bit mode, change it to run in 32 bit mode with:
Options➢Conformance Options➢General➢Conform Settings:
[✓] Run as 32bit on a 64bit OS
(this is needed because Visual Studio's *Register for COM interop* only registers the drivers as a 32bit COM driver, but not also as a 64bit COM driver on a 64-bit machine - as would be done by the Inno Setup installer)
Conform64Bits.png
Conform32Bits.png

☐ 9) Using the ASCOM **Conform** tool, Options➢Check Focuser, Options➢Select Driver, select the *Acme Focuser*
ConformChooserFocuser.png

☐ 10) Select ASCOM Focuser Chooser➢Properties... to get the DeviceName Setup dialog for the Focuser
ConformFocuserProperties.png

☐ 11) Run the Check Conformance and verify that no errors, warnings or issues are found and the Focuser driver passes ASCOM validation!!
ConformFocuser.png

☐ 12) Add New Project for application for testing driver
- File➢Add➢New Project...
- **ASCOM Driver Test Forms Application (VB)**
- Name: **TestDriver**
AddTestDriver.png

☐ 13) ASCOM Driver Project Wizard
- Device Class - **Focuser**
- Device Name/Model - **Acme**
- Create
WizardFocuser.png

☐ 14) If Option Strict is turned on, remove the following error:
On Form1.vb, line 43, add **.ToString** to the end of the line
OptionStrictError.png

☐ 15) Right-Click Solution Explorer➢TestDriver project➢Set as Startup Project to set the **TestDriver** project as the startup project

☐ 16) Build the solution

☐ 17) Run the code (F5), click the test form's Choose button, select the *Acme Focuser*, select ASCOM Focuser Chooser➢Properties... to get the DeviceName Setup dialog for the Focuser, OK those dialogs and verify the ASCOM.Acme.Focuser is shown on the test form.
TestDriverResults.png

At this point, additional code can be added to the Focuser driver to control the Focuser hardware and additional code and controls can be added to the **TestDriver** project to exercise and debug the features of the in-proc DLL Focuser driver.

When the ASCOM driver development is complete, the ASCOM Driver Install Script Generator can be used to generate an Inno Setup script to generate a Windows setup executable that can be used to distribute the driver just developed.
Note: The driver project should be closed in the IDE before running the Inno Setup compiler.

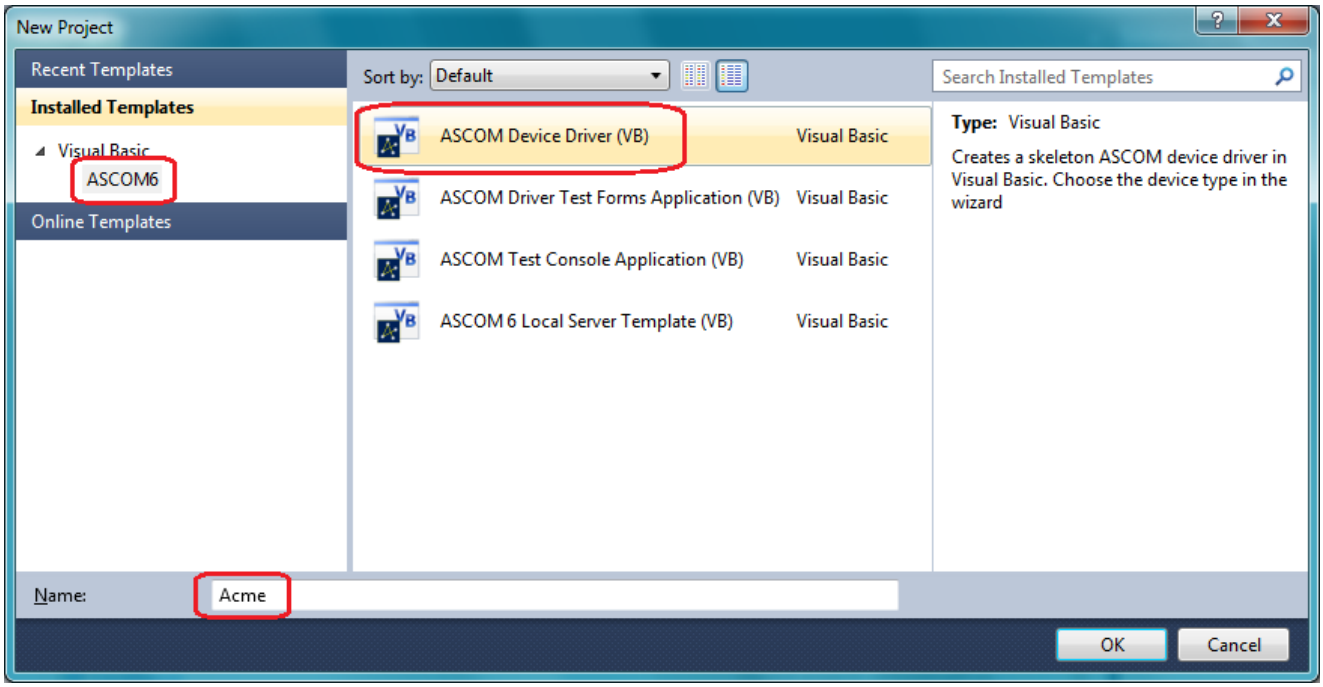
☐ 18) Clean the solution with Build➢Clean Solution so that the driver will be automatically unregistered from COM and ASCOM
(if menu Build➢Clean Solution is not shown, use Tools➢Customize➢Commands➢Menu bar: Build➢Add Command...➢Categories: Build, Commands: Clean Solution, OK, Close to add that menu item)

At this point, the *Acme Focuser* should no longer be available in **Conform**'s Select Driver.

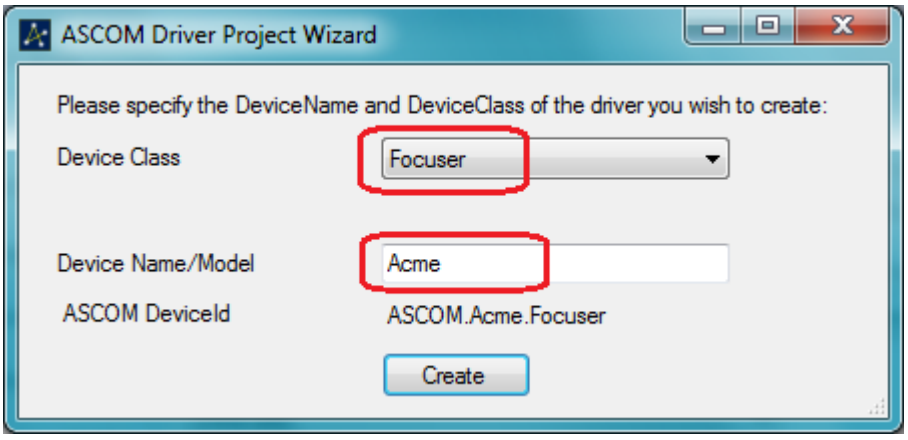
ASCOM VB Drivers Development Framework Walk-Through

Visual Basic 2010 Express ← Run as administrator !!!

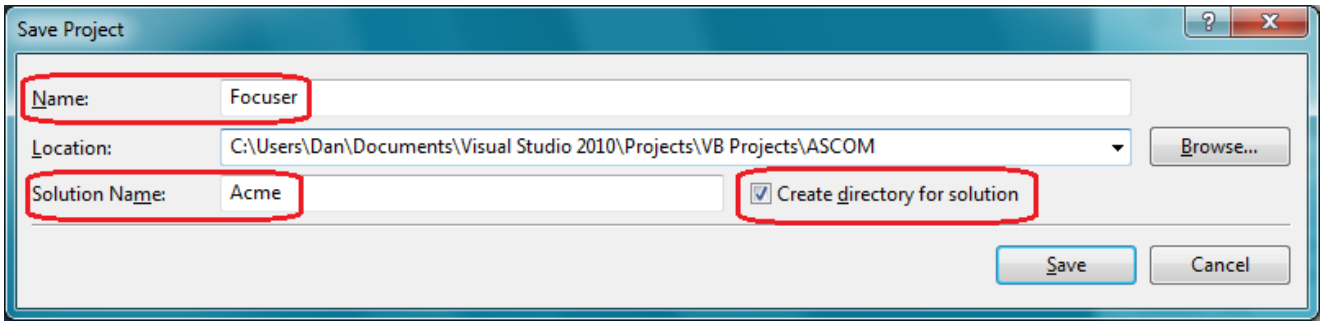
1) New Project - **ASCOM Device Driver (VB)**
   - Name: **Acme**



2) ASCOM Driver Project Wizard
   - Device Class - **Focuser**
   - Device Name/Model - **Acme**
   - Create

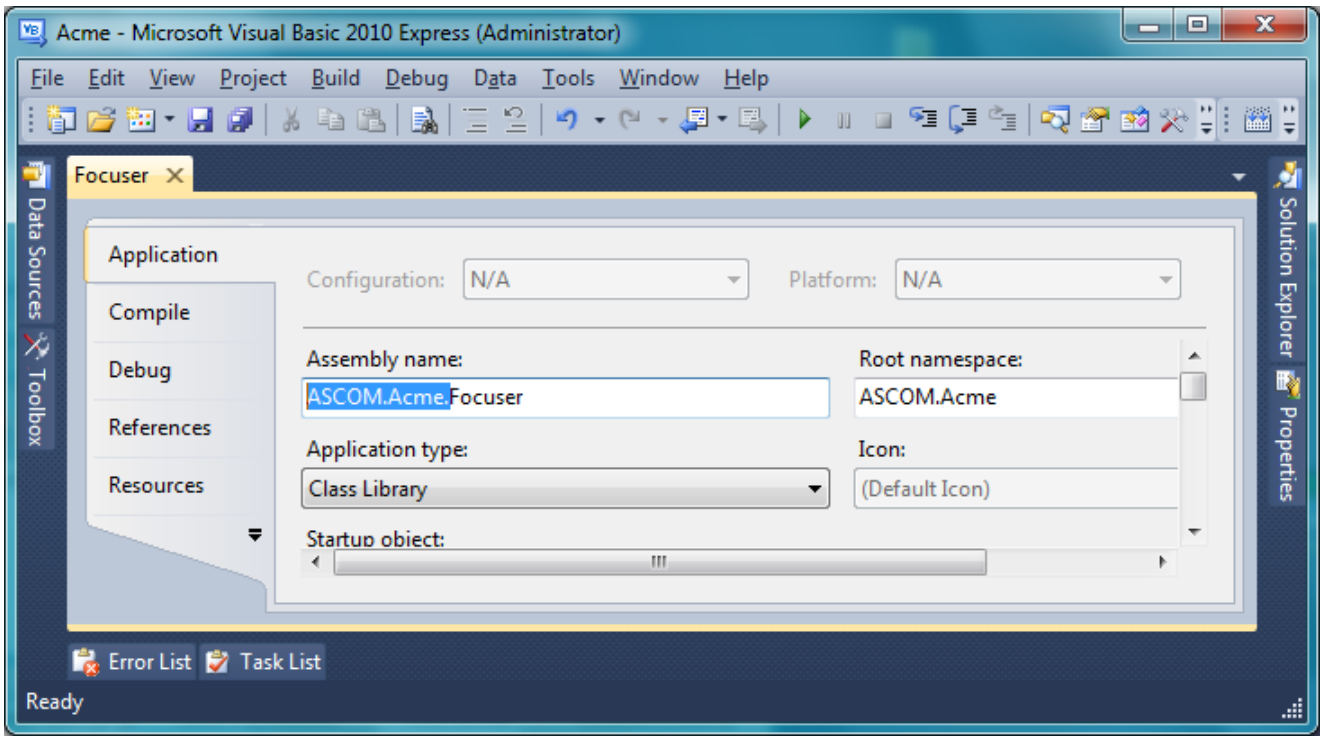


3) File➢Save All –
   - Name: **Focuser**
   - Location: ...**VB Projects\ASCOM**
   - Solution Name: **Acme** ← **MAKE SURE**!!! because Visual Studio makes same
   - [✓] Create directory for solution as Name!!!



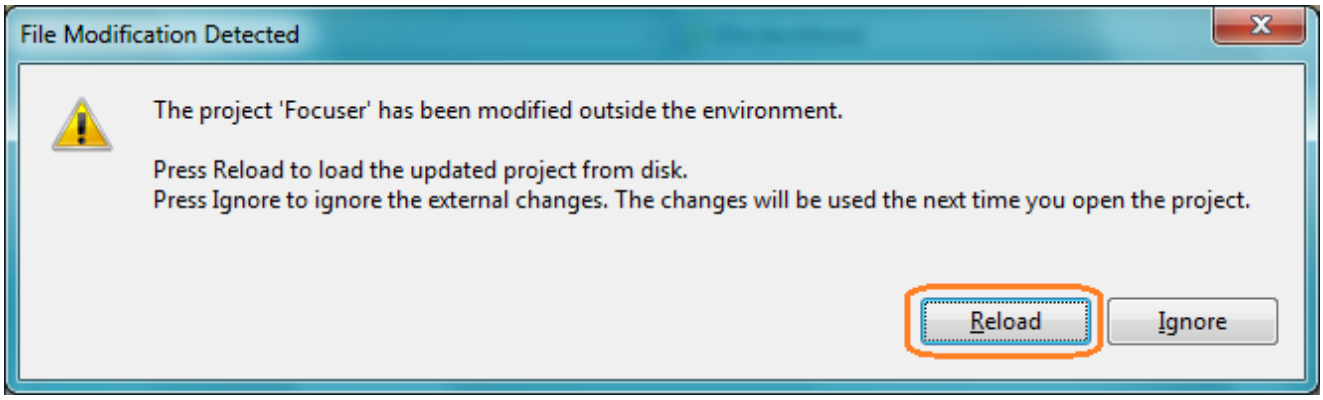
4) Set Focuser Properties➢Application➢Assembly name: **ASCOM.Acme.Focuser**
   - Save project

(The template wizard does not include full assembly name - done here for LocalServer consistency.)



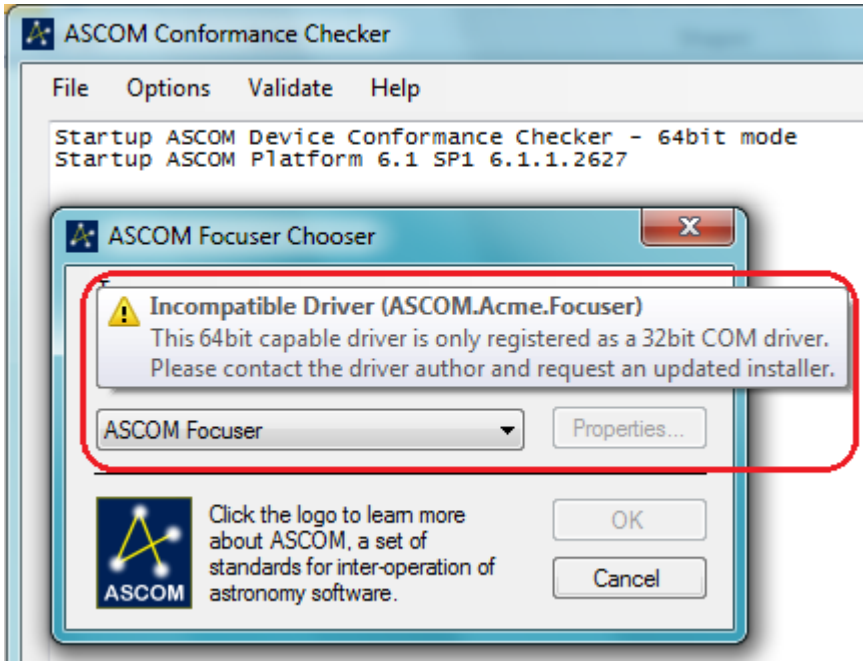
5) Enable 32 and 64-bit code generation for the Focuser driver by modifying (outside of the Visual Basic IDE environment) the <PlatformTarget> tag in the Focuser driver's Focuser.vbproj file to be **AnyCPU** located under the following tag: (it was "x86"!!!)
   <PropertyGroup Condition=" '$(Configuration)|$(Platform)' == 'Debug|AnyCPU' ">
   (i.e. in Focuser.vbproj: <PlatformTarget>**AnyCPU**</PlatformTarget>)

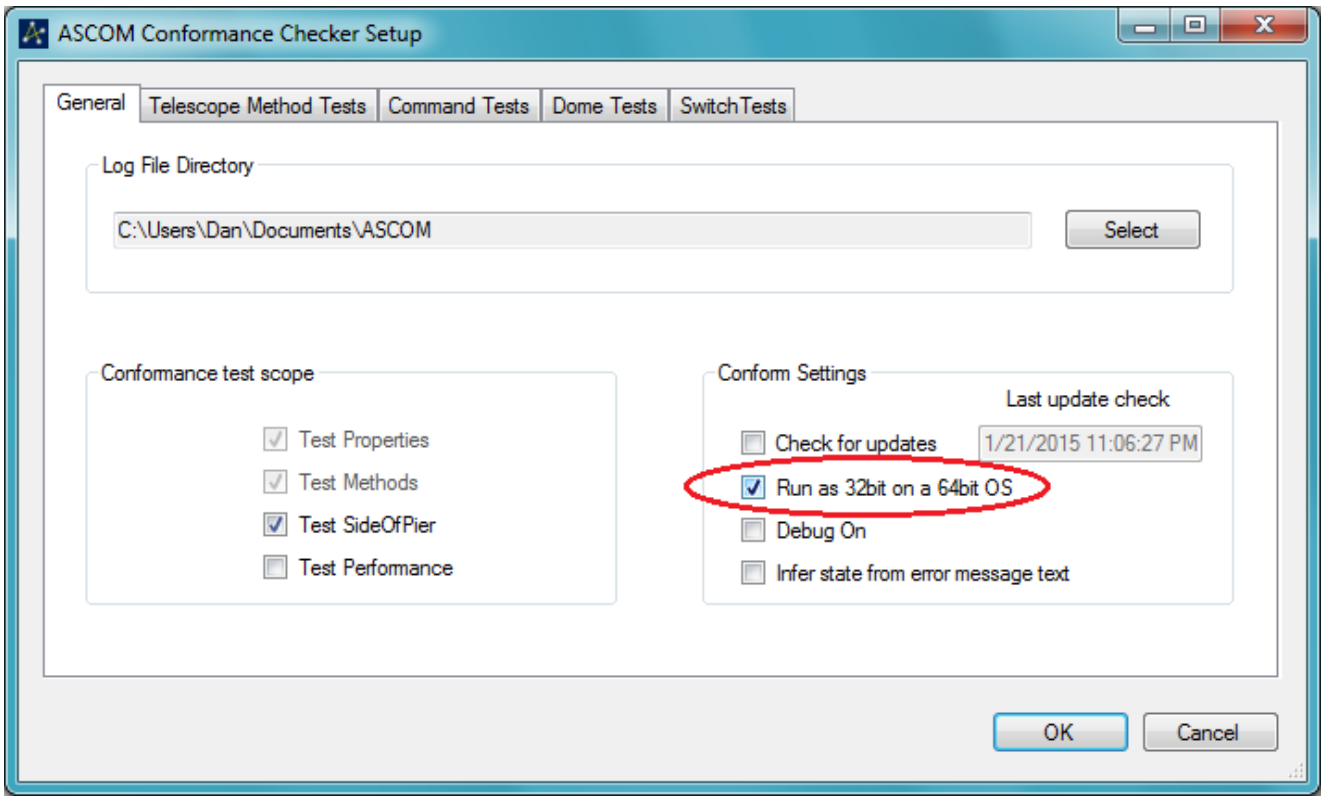
6) Enable XML documentation for the Focuser driver by modifying (outside of the Visual Basic IDE environment) the empty <DocumentationFile> tags in the Focuser driver's Focuser.vbproj file to be the project's name **Focuser.xml** located under the following tags:
<PropertyGroup Condition=" '$(Configuration)|$(Platform)' == 'Debug|AnyCPU' ">
<PropertyGroup Condition=" '$(Configuration)|$(Platform)' == 'Release|AnyCPU' ">
(i.e. in Focuser.vbproj: <DocumentationFile>**Focuser.xml**</DocumentationFile>)
(no "Generate XML documentation file" option in Visual Basic 2010 Express Edition's Properties➢Compile)
A prompt will be given to reload the Focuser project due to the changed project file.



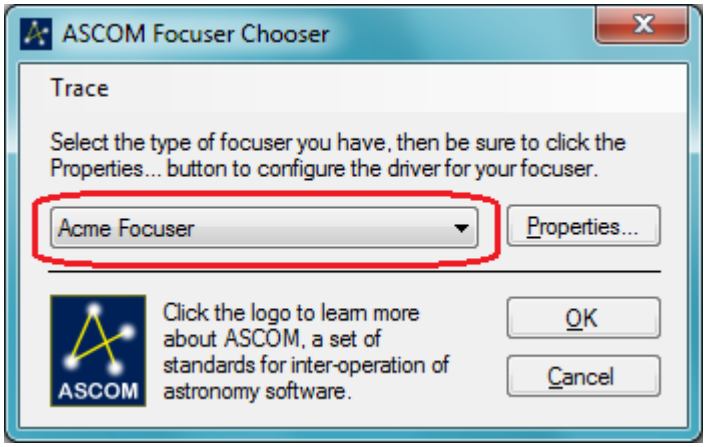
7) Build the project

8) Run the ASCOM **Conform** tool. If it is running in 64 bit mode, change it to run in 32 bit mode with:
Options➢Conformance Options➢General➢Conform Settings:
[✓] Run as 32bit on a 64bit OS
(this is needed because Visual Studio's *Register for COM interop* only registers the drivers as a 32bit COM driver, but not also as a 64bit COM driver on a 64-bit machine - as would be done by the Inno Setup installer)

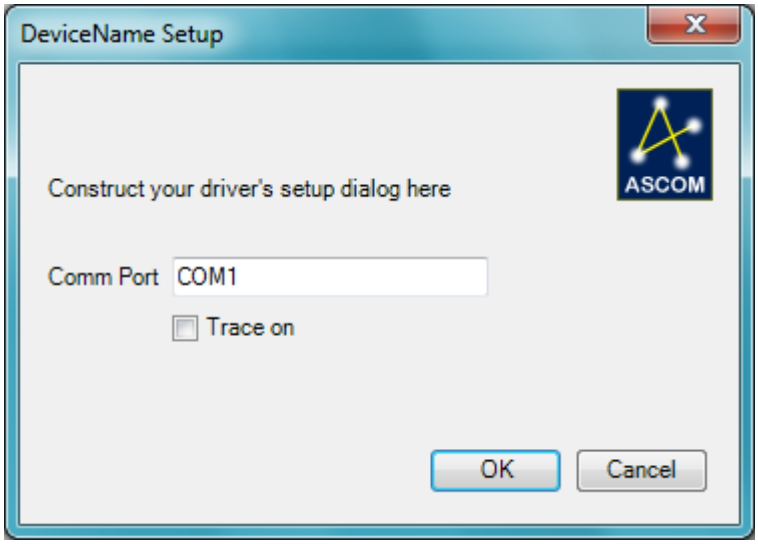




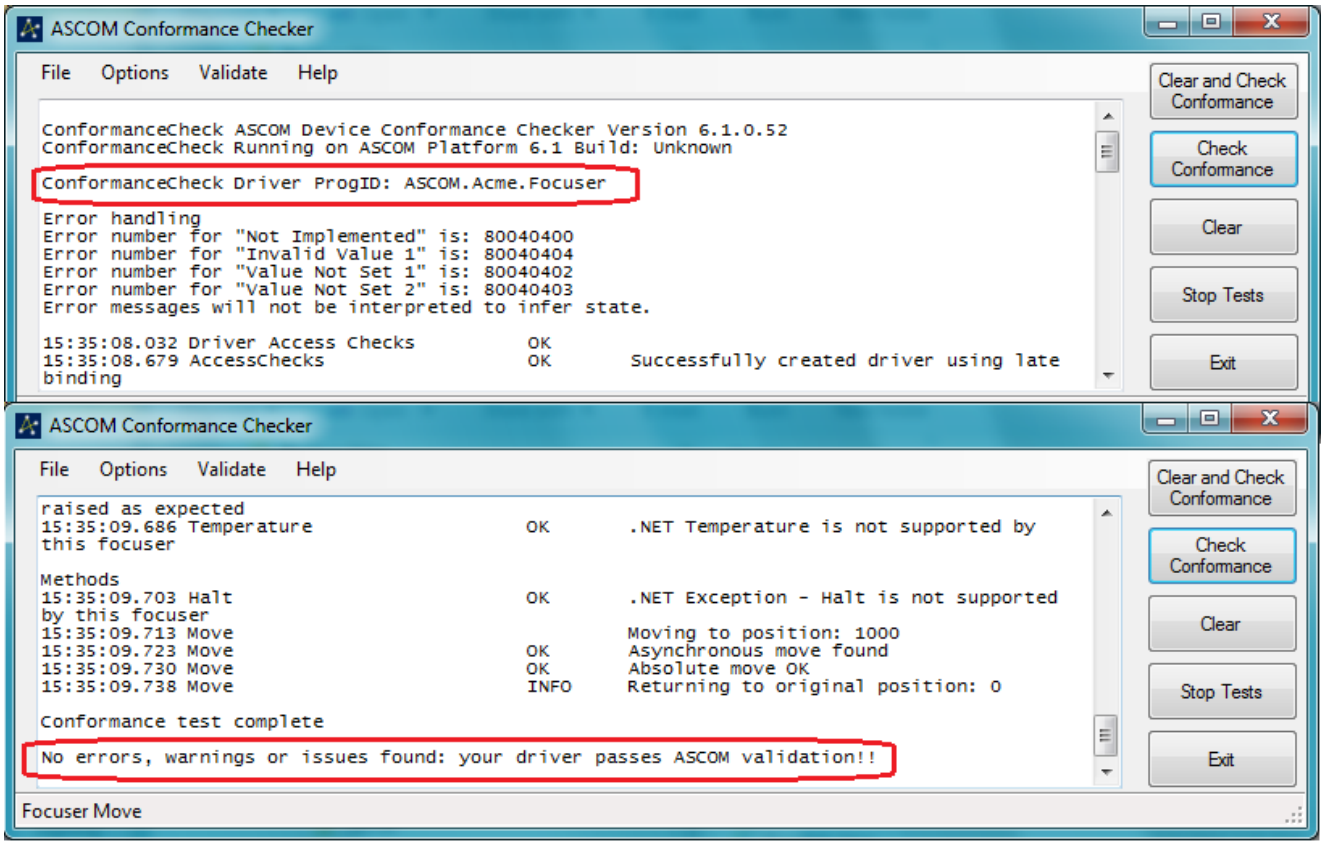
9) Using the ASCOM **Conform** tool, Options➢Check Focuser, Options➢Select Driver, select the *Acme Focuser*



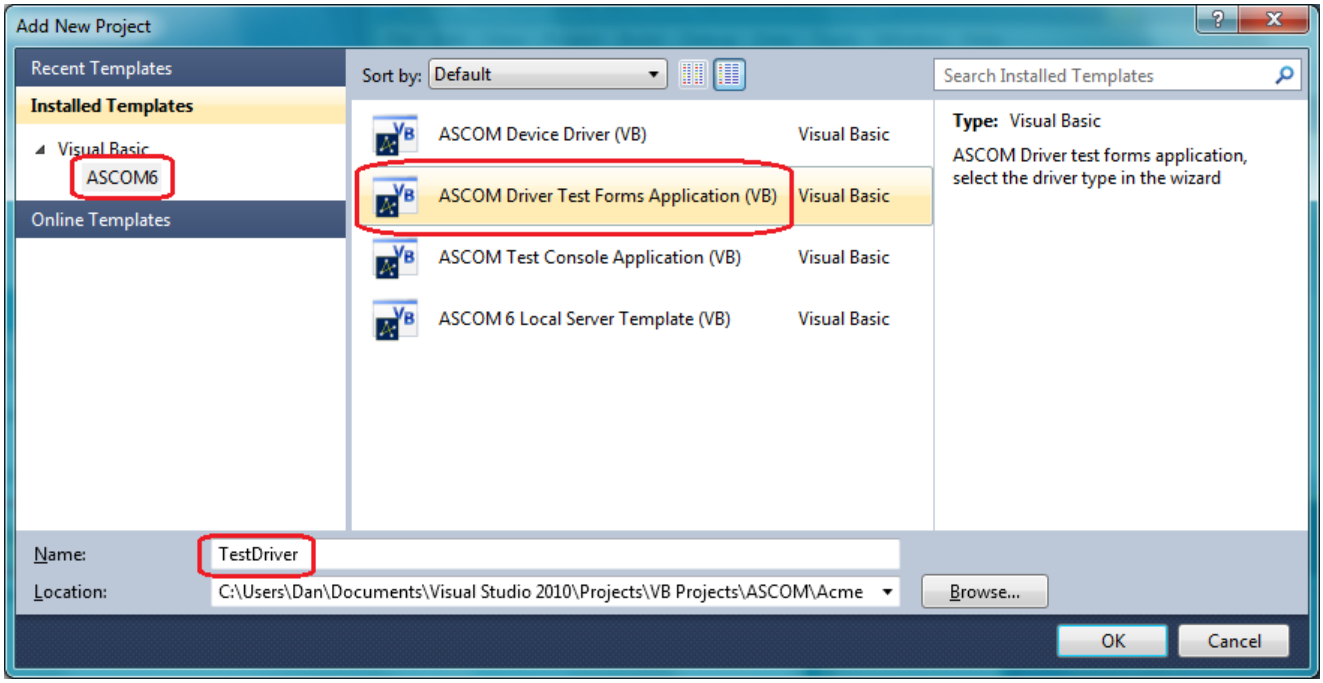
10) Select ASCOM Focuser Chooser➢Properties... to get the DeviceName Setup dialog for the Focuser



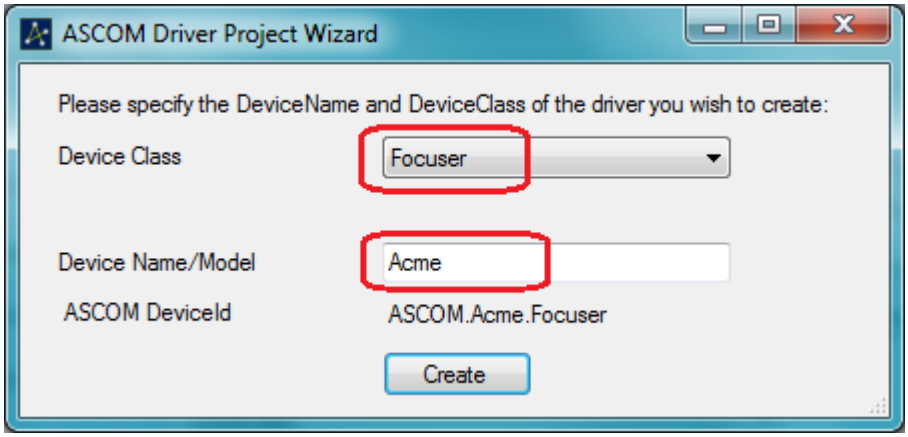
11) Run the Check Conformance and verify that no errors, warnings or issues are found and the Focuser driver passes ASCOM validation!!



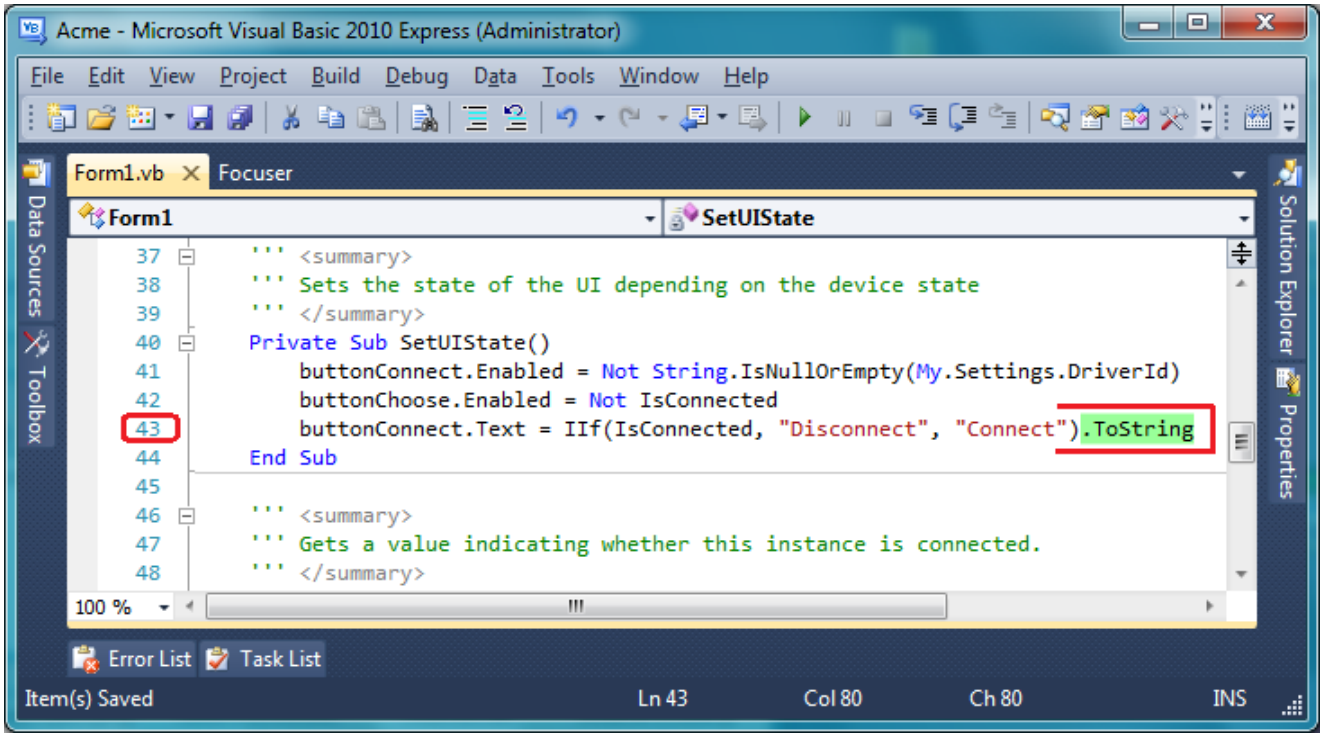
12) Add New Project for application for testing the driver
- File➢Add➢New Project...
- **ASCOM Driver Test Forms Application (VB)**
- Name: **TestDriver**



13) ASCOM Driver Project Wizard
    - Device Class - **Focuser**
    - Device Name/Model - **Acme**
    - Create



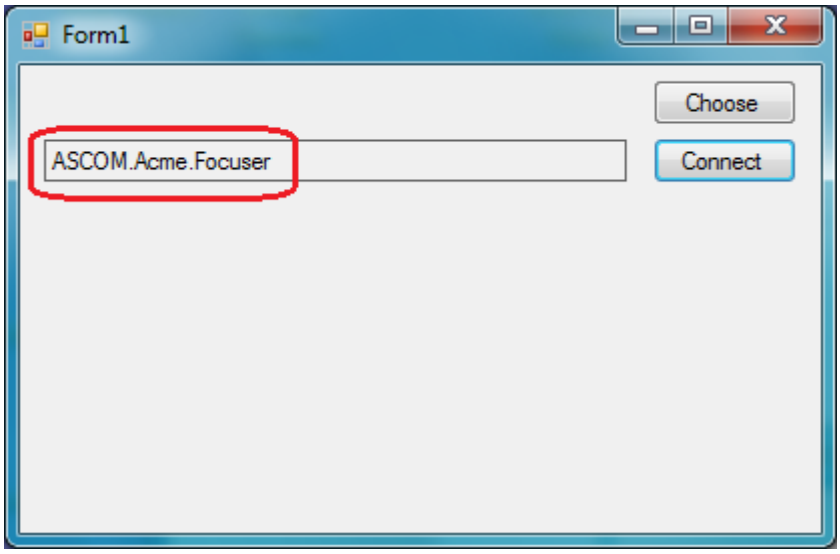
14) If Option Strict is turned on, remove the following error:
    On Form1.vb, line 43, add **.ToString** to the end of the line
(This is an issue with the VB Driver Test Forms Application template.)



15) Right-Click Solution Explorer➢TestDriver project➢Set as Startup Project to set the **TestDriver** project as the startup project

16) Build the solution

17) Run the code (F5), click the test form's Choose button, select the *Acme Focuser*, select ASCOM Focuser Chooser➢Properties... to get the DeviceName Setup dialog for the Focuser, OK those dialogs and verify the ASCOM.Acme.Focuser is shown on the test form.



At this point, additional code can be added to the Focuser driver to control the Focuser hardware and additional code and controls can be added to the **TestDriver** project to exercise and debug the features of the in-proc DLL Focuser driver.

When the ASCOM driver development is complete, the ASCOM Driver Install Script Generator can be used to generate an Inno Setup script to generate a Windows setup executable that can be used to distribute the driver just developed.
Note: The driver project should be closed in the IDE before running the Inno Setup compiler.

18) Clean the solution with Build➢Clean Solution so that the driver will be automatically unregistered from COM and ASCOM
(if menu Build➢Clean Solution is not shown, use Tools➢Customize➢Commands➢Menu bar: Build➢Add Command...➢Categories: Build, Commands: Clean Solution, OK, Close to add that menu item)

At this point, the *Acme Focuser* should no longer be available in **Conform**'s Select Driver.