Visual Basic.NET and PicBasic Pro Tutorial
Visual Basic 2008 Controls PIC Microcontroller
Cornerstone Electronics Technology and Robotics II

- **Administration:**
    - Prayer
- **Sending Byte Data to a Microcontroller:**
    - In this lesson we are going use Visual Basic.NET to send data in the form of a byte array to a PIC microcontroller and have the microcontroller control output devices. The PIC microcontroller is programmed in PicBasic Pro.
- **PicBasic Pro and Visual Basic Programs Used in This Lesson:**
    - General PicBasic Pro Program Listing:
      http://www.cornerstonerobotics.org/picbasic.php
    - Lab 1 PicBasic Pro program, pbp_vb_led1 as .pdf file:
      http://cornerstonerobotics.org/code/pbp_vb_led1.pdf
    - Lab 1 Visual Basic.NET program, vb_led1 as .pdf file:
      http://cornerstonerobotics.org/code/vb_led1.pdf
    - Lab 2 PicBasic Pro program, pbp_vb_servo1 as .pdf file:
      http://cornerstonerobotics.org/code/pbp_vb_servo1.pdf
    - Lab 2 Visual Basic.NET program, vb_servo1 as .pdf file:
      http://cornerstonerobotics.org/code/vb_servo1.pdf
- **Software:**
    - Visual Basic 2008 Express Edition:
        - Visual Basic 2008 Express Edition is available free online at: http://www.microsoft.com/express/download/ (Requires XP or greater.)
        - Free Visual Basic.NET tutorials:
            - http://www.homeandlearn.co.uk/NET/vbNET.html
            - http://www.functionx.com/vbnet/index.htm
            - http://msdn.microsoft.com/en-us/vbasic/bb466159.aspx
    - PicBasic Pro:
        - PicBasic Pro is available from micro-Engineering Labs, Inc., melabs.com. We prefer the melabs U2 Programmer over the EPIC Programmer. Both programmers are displayed at: http://melabs.com/products/bundles.htm. The cost for both the PicBasic Pro Compiler and the U2 Programmer with Accessories is $359.90 USD
- **Hardware:**
    - For a lesson on serial connections between the PC and the PIC microcontroller, see: http://cornerstonerobotics.org/curriculum/lessons_year2/erii_rs232_1.pdf

Cornerstone Electronics Technology and Robotics II
VB.NET and PicBasic Pro Lab 1 – LED On/Off

- **Purpose:** The purpose of this lab is to establish basic serial communications between Visual Basic.NET and a PIC microcontroller.

- **Apparatus and Materials:**

    o 1 – Breadboard with +5V Power Supply
    o 1 – MAX232 RS-232 Driver/Receiver
    o 1 – PIC16F88
    o 5 – 1 uF Capacitors
    o 1 – 150 Ohm Resistor
    o 1 – 4.7K Resistor
    o 1 – LED
    o 1 – Modified DB9 Connector - Connector,D-SUB,.318"RT,9P-F, Short Metal Housing, Jameco Part # 104952

- **Procedure:**
    o Wire the vb_led1 circuit below. Read the schematic note.
    o To make connection between the DB9 connector and the breadboard, remove the mounting posts and 4 front pins on the connector. See the photos below.
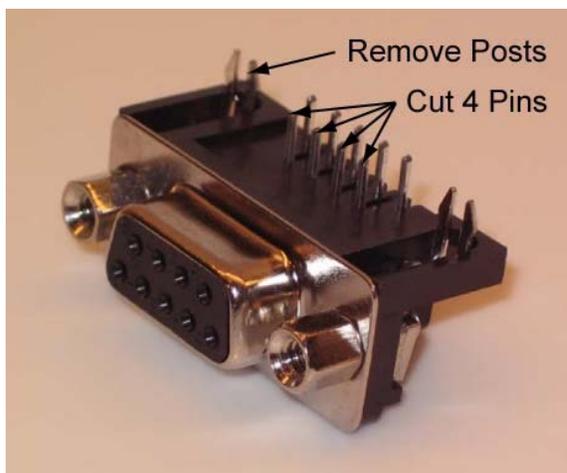
New DB9 Connector   **Modified DB9 Connector**
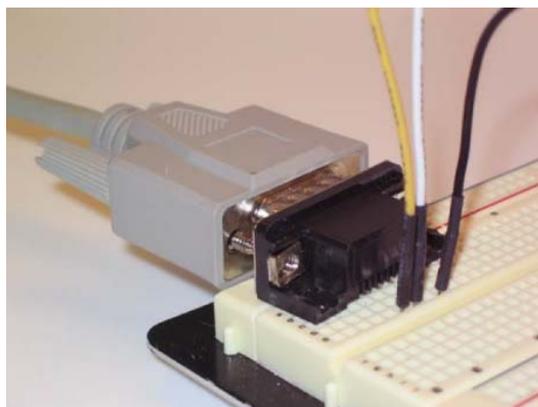
Modified DB9 Connector Mounted on a Breadboard

vb_led1

NOTE: Make certain that the polarity of each capacitor
is wired correctly.

- o Open the Visual Basic 2008 Express Edition
- o Create a new project named vb_led.
- o From the Toolbox window on the left side of the IDE, add 1 label, 2 radiobuttons, and 1 button by dragging them over to the form.
- o Set the properties to the following:

Properties for Objects on VB Form

Object	Property	Setting
Label1	Text	"LED On/Off"
Radiobutton1	Text	"Off"
Radiobutton2	Text	"On"
Button1	Name	"btnSend"
Button1	Text	"Send"
Form1	Text	"LED Control"
Form1	Name	"LED_Control"

- o Click on the form and resize to the general shape below. The form should look like this:

3

- o Click on the View Code button:

- o Copy code from http://cornerstonerobotics.org/code/vb_led1.pdf onto the Code Editor.
- o Make sure that "Handles btnSend.Click" is on the same line and follows "Private Sub btnSend_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)"
- o Program the PIC16F88 with http://cornerstonerobotics.org/code/pbp_vb_led1.pbp
- o Click the Start Debugging button on the Standard toolbar.
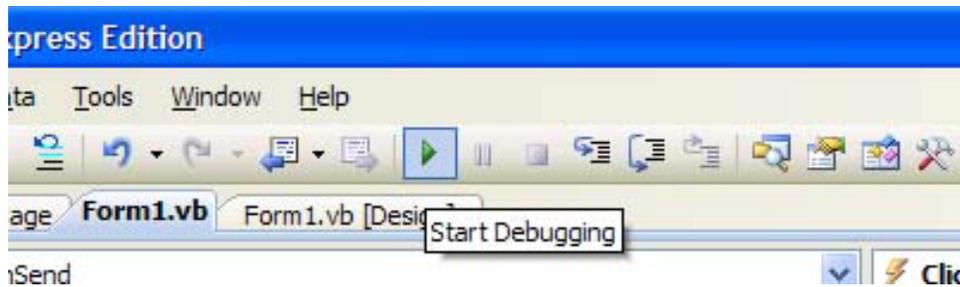
- o Visual Basic displays the LED Control form in the Visual Studio IDE.
- o Choose the HIGH RadioButton and press the Send button. The LED should light up. Choose the LOW RadioButton and press the Send button. The LED should turn off.

Cornerstone Electronics Technology and Robotics II
VB.NET and PicBasic Pro Lab 2 – Servo Position Control

- **Purpose:** The purpose of this lab is to send data in the form of a byte from Visual Basic.NET to a PIC microcontroller.

- **Apparatus and Materials:**

    - 1 – Breadboard with +5V Power Supply
    - 1 – MAX232 RS-232 Driver/Receiver
    - 1 – PIC16F88
    - 5 – 1 uF Capacitors
    - 1 – 150 Ohm Resistor
    - 1 – 4.7K Resistor
    - 1 – Hobby Servo
    - 1 – Modified DB9 Connector - Connector,D-SUB,.318"RT,9P-F, Short Metal Housing, Jameco Part # 104952

- **Procedure:**
    - Wire the vb_servo1 circuit below. Read the schematic note.
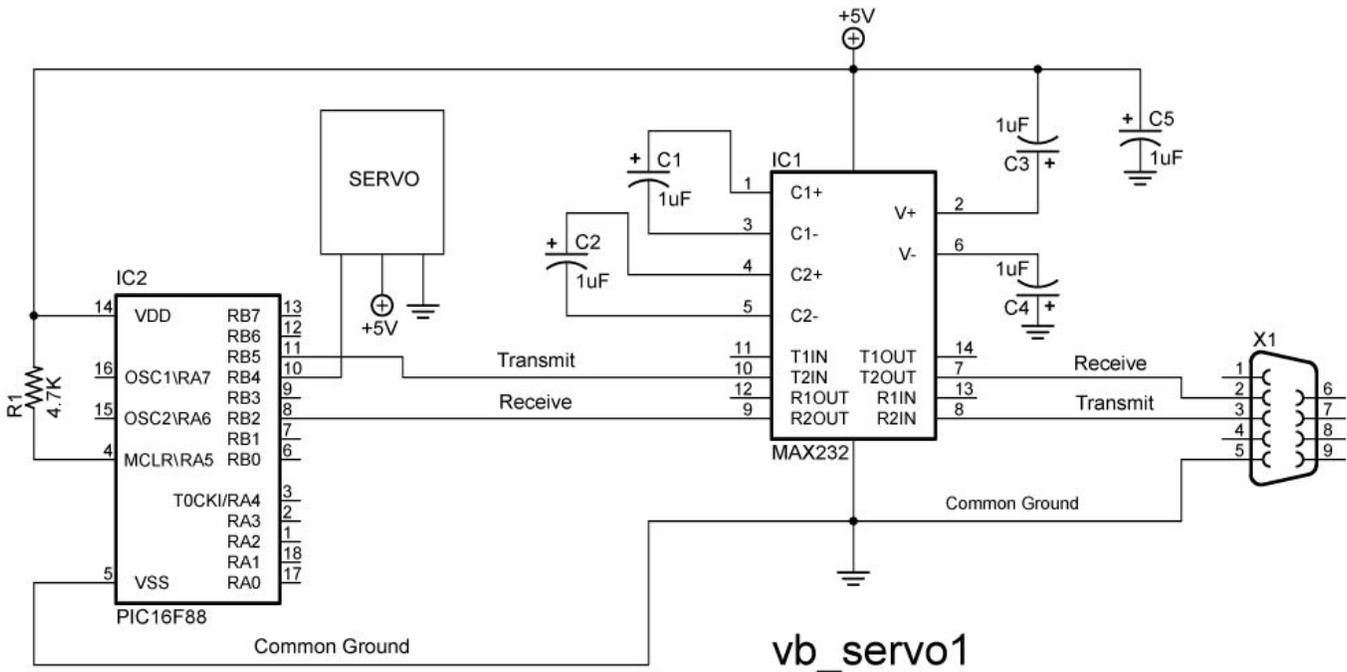
NOTE: Make certain that the polarity of each capacitor is wired correctly.

- WITH THE PIC16F88, MAKE SURE TO HAVE SEPARATE POWER SOURCES FOR THE PIC AND THE SERVO. MAKE SURE TO HAVE A COMMON GROUND BETWEEN THE PIC AND SERVO. We use one 9V battery and two 78L05 voltage regulators.
    - Create a new project named vb_servo.
    - From the Toolbox, add 2 buttons to the form.

o Set the properties to the following:

Properties for Objects on VB Form

Object	Property	Setting
Button1	Name	"btnEnter"
Button1	Text	"Enter Position"
Button2	Name	"btnSend"
Button2	Text	"Send"
Form1	Text	"Servo Control"

o Click on the form and resize to the general shape below. Now click on each button and expand so that "Enter Position" displays. The form should look like this:

o Click on the View Code button:
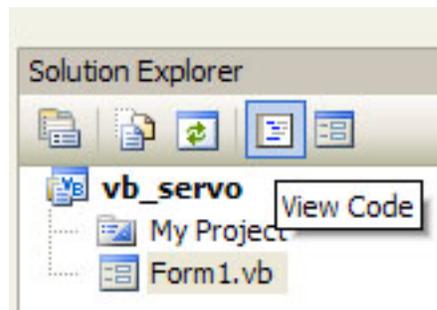
o Copy code from http://www.cornerstonerobotics.org/code/vb_servo1.pdf onto the Code Editor.
o As in Lab 1, make sure that both "Handles btnSend.Click"s are moved to the end of the line above.
o Program the PIC16F88 with http://www.cornerstonerobotics.org/code/pbp_vb_servo1.pbp

o   Click the Start Debugging button on the Standard toolbar.
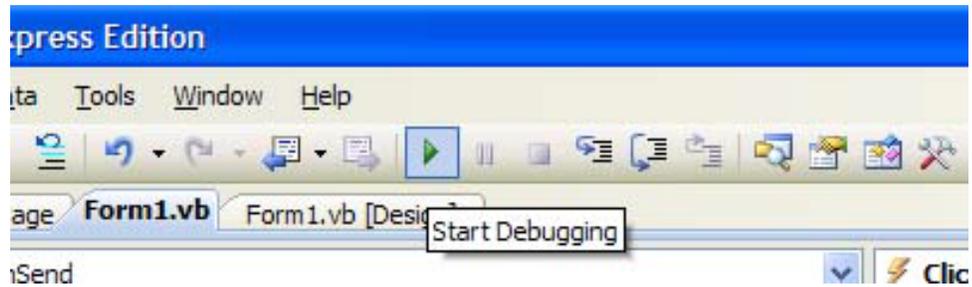
o   Visual Basic displays the Servo Control form in the Visual Studio IDE.
o   Click on the Enter Position button; the Servo Position Input window appears. Enter a number between 80 and 220 and press OK. Now press the Send button in the Servo Control window. If the servo does not respond, it may already be in that position so enter 220 and the servo should rotate.