Supplementary Information 1

The user interface for the two-temperature preference behavioral assay was written using Microsoft Visual Basic 2008 Express Edition. The Graphical User Interface (GUI) is depicted below, followed by a list of the variables named in the user interface input textboxes, check boxes, and output checkboxes. Lastly, we have included the source code used to control the Phidget devices.

User Interface

User interface variables

1) The user sets the desired temperature via the input textboxes for plates 1 and 2 (variables TempSet1 and TempSet2).

2) Check boxes indicate whether the relays controlling plates 1 and 2 are closed (variables CheckBox1 and CheckBox2).

3) The plate temperature (as recorded by the Phidget temperature sensors) is shown to the user via output textboxes (variables Temp1Txt and Temp2Txt).

4) The ambient temperature (as recorded by the Phidget temperature sensors) is shown to the user via output textboxes (variables ambientTemp1Txt and ambientTemp2Txt).

5) The current state of the Phidget relay board and the device's serial number are provided to the user via output textboxes (variables RelayBoardAttachedTxt and RelayBoardSerialTxt).

6) The current state of the Phidget temperature sensors and their serial numbers are provided to the user via output textboxes (variables TempSensor1Attachedtxt, TempSensor2attachedTxt, TempSensor1SerialTxt, TempSensor2SerialTxt).

Source code for temperature control of Phidget relays and temperature sensors

Several modifications must be made to the source code below for proper function. Phidgets devices are referenced by their serial numbers, which must be substituted appropriately. This code is written to warm each plate above ambient temperature. For cooling applications, the inequalities that compare the user-defined temperature to the current plate temperature must be reversed (there are two in the code below, one for each plate). When cooling is desired, bear in mind also that the polarity of the current supplied to the Peltier TECs must be reversed. Our source code is presented below.

```
Public Class Form1
    Dim WithEvents phidgetTemperature1 As Phidgets.TemperatureSensor
    Dim WithEvents phidgetTemperature2 As Phidgets.TemperatureSensor
    Dim WithEvents phidgetIFK As Phidgets.InterfaceKit
    ' The above statements declare the variables as types that can raise events

    Dim TempSet1 As Decimal = 23
    Dim TempSet2 As Decimal = 23
    ' These variables are the user defined temperatures that are compared to the
thermocouple temperatures


    ' Initialize temperature sensors and relay board and hook the event handlers
    Private Sub Form1_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
        'The above line was generated automatically by the Form Designer

        phidgetTemperature1 = New Phidgets.TemperatureSensor
        phidgetTemperature1.open(83240)

        phidgetTemperature2 = New Phidgets.TemperatureSensor
        phidgetTemperature2.open(82910)

        phidgetIFK = New Phidgets.InterfaceKit
        phidgetIFK.open(108305)

    End Sub
```

```vb
' TemperatureSensor1 Attach event handler, want to populate fields in the GUI
' fields will include serial number and whether the board is connected
' These must be done independently, as the sub references each board separately by
serial number (S/N)
Private Sub Phidgets_Temp1_Attach(ByVal sender As Object, ByVal e As
Phidgets.Events.AttachEventArgs) Handles phidgetTemperature1.Attach
    TempSensor1Attachedtxt.Text = sender.Attached.ToString()
    TempSensor1SerialTxt.Text = sender.SerialNumber.ToString()
End Sub

' TemperatureSensor2 Attach event handler, want to populate fields in the GUI
' fields will include serial number and whether the board is connected
' These must be done independently, as the sub references each board separately by
S/N
Private Sub Phidgets_Temp2_Attach(ByVal sender As Object, ByVal e As
Phidgets.Events.AttachEventArgs) Handles phidgetTemperature2.Attach
    TempSensor2attachedTxt.Text = sender.Attached.ToString()
    TempSensor2SerialTxt.Text = sender.SerialNumber.ToString()
End Sub

' RelayBoard Attach event handler, want to populate fields in the GUI
' fields will include serial number and whether the board is connected
' These must be done independently, as the sub references each board separately by
S/N
Private Sub Phidgets_RelayBoard_Attach(ByVal sender As Object, ByVal e As
Phidgets.Events.AttachEventArgs) Handles phidgetIFK.Attach
    RelayBoardAttachedTxt.Text = sender.Attached.ToString()
    RelayBoardSerialTxt.Text = sender.SerialNumber.ToString()
End Sub

' If the program is closed (VB calls the window the "form"), this sub will close the
phidgets that are attached
Private Sub Form1_FormClosing(ByVal sender As Object, ByVal e As
System.Windows.Forms.FormClosingEventArgs) Handles Me.FormClosing
    If phidgetIFK.Attached = True Then
        phidgetIFK.close()
    End If
    If phidgetTemperature1.Attached = True Then
        ' RemoveHandler phidgetTemperature1.Attach, AddressOf
temperatureSensor_Attach
        ' this and similar lines are included in example code - they may not be necessary

        phidgetTemperature1.close()
    End If
    If phidgetTemperature2.Attached = True Then
        phidgetTemperature2.close()
```

```vbnet
        End If
    End Sub


    ' Temperature Change event handler--display the current temperature reading
    ' from the phidgetTemperature1 sensor in GUI textboxes
    ' This code also contains a nested IF-THEN statement to open and close relays
    ' when temperatures reach user-defined set points
    Private Sub phidgettemperature1_TemperatureChange(ByVal sender As Object,
ByVal e As Phidgets.Events.TemperatureChangeEventArgs) Handles
phidgetTemperature1.TemperatureChange


        phidgetIFK.outputs(0) = False ' Closes relay 0 when the program first runs (in case
the relay is open for some reason) – this may not be necessary
        If e.Index = 0 Then
            ambientTemp1Txt.Text =
phidgetTemperature1.ambientSensor.Temperature.ToString() ' assigns the ambient
temperature to the ambientTemp1txt variable
            Temp1Txt.Text = e.Temperature.ToString() ' assigns the thermocouple
temperature to the temp1txt variable

            If e.Temperature < TempSet1 Then ' compares thermocouple 1 temperature to the
user-defined decimal variable TempSet1
                phidgetIFK.outputs(0) = True ' Sets first relay (relays are 0, 1, 2, and 3) to
boolean true, which is open
                CheckBox1.Checked = True ' Sets heating indicator checkbox so that "heating"
is checked
            Else : phidgetIFK.outputs(0) = False ' This closes the relay if the temperature falls
below the user-defined variable TempSet1
                CheckBox1.Checked = False ' Sets heating indicator checkbox so that "heating"
is checked
            End If

            ' **** need to export temperature data to a file

        End If
    End Sub
    ' Temperature Change event handler--display the current temperature reading
    ' from the phidgetTemperature2 sensor in GUI textboxes
    ' This code also contains a nested IF-THEN statement to open and close relays
    ' when temperatures reach user-defined set points
    Private Sub phidgettemperature2_TemperatureChange(ByVal sender As Object,
ByVal e As Phidgets.Events.TemperatureChangeEventArgs) Handles
phidgetTemperature2.TemperatureChange
```

```vbnet
        phidgetIFK.outputs(1) = False ' Closes relay 1 when the program first runs (in case
the relay is open for some reason) – this may not be necessary
      If e.Index = 0 Then
        ambientTemp2txt.Text =
phidgetTemperature2.ambientSensor.Temperature.ToString() ' assigns the ambient
temperature to the ambientTemp2txt variable
        Temp2Txt.Text = e.Temperature.ToString() ' assigns the thermocouple
temperature to the temp2txt variable

        If e.Temperature < TempSet2 Then ' compares thermocouple 2 temperature to the
user-defined decimal variable TempSet2
          phidgetIFK.outputs(1) = True ' Sets second relay (relays are 0, 1, 2, and 3) to
boolean true, which is open
          CheckBox2.Checked = True ' Sets heating indicator checkbox so that "heating"
is checked
        Else : phidgetIFK.outputs(1) = False ' This closes the relay if the temperature falls
below the user-defined variable TempSet2
          CheckBox2.Checked = False ' Sets heating indicator checkbox so that "heating"
is checked
        End If

      End If
    End Sub
    ' Temperature Change event handler--display the current temperature reading
    ' from the phidgetTemperature2 sensor in GUI textboxes
    Private Sub temperatureSensor2_TemperatureChange(ByVal sender As Object, ByVal
e As Phidgets.Events.TemperatureChangeEventArgs) Handles
phidgetTemperature2.TemperatureChange
      If e.Index = 0 Then
        ambientTemp2txt.Text =
phidgetTemperature2.ambientSensor.Temperature.ToString()
        Temp2Txt.Text = e.Temperature.ToString()
      End If
    End Sub

    ' This sub sets the TempSet1 variable to the user-defined temperature.
    ' TempSet1 is a public variable declared at the top, defaults to 23
    Private Sub TempSet1txt_TextChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles TempSet1txt.TextChanged
      TempSet1 = TempSet1txt.Text

    End Sub
    ' This sub sets the TempSet2 variable to the user-defined temperature.
    ' TempSet1 is a public variable declared at the top, defaults to 23
    Private Sub TempSet2_TextChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles TempSet2txt.TextChanged
```

```
        TempSet2 = TempSet2txt.Text
    End Sub
End Class
```