Solar Panel Tracker

1.0

Generated by Doxygen 1.8.2

Wed Aug 22 2012 16:40:41

Contents

1 Namespace Index **1**

1.1 Packages . . . 1

2 Hierarchical Index **3**

2.1 Class Hierarchy . . . 3

3 Class Index **5**

3.1 Class List . . . 5

4 Namespace Documentation **7**

4.1 gui Namespace Reference . . . 7

4.1.1 Detailed Description . . . 7

4.2 launcher Namespace Reference . . . 7

4.2.1 Detailed Description . . . 7

4.3 tiny26 Namespace Reference . . . 8

4.3.1 Detailed Description . . . 8

4.3.2 Variable Documentation . . . 9

4.3.2.1 vid . . . 9

5 Class Documentation **11**

5.1 gui.GUI_Launcher Class Reference . . . 11

5.1.1 Detailed Description . . . 11

5.2 gui.MotorGUI Class Reference . . . 11

5.2.1 Detailed Description . . . 12

5.2.2 Member Function Documentation . . . 12

5.2.2.1 OnAboutRequested . . . 12

5.2.2.2 OnButtonPressed . . . 13

5.2.2.3 OnMoveServoRequest . . . 13

5.2.2.4 OnSolarMaxRequest . . . 13

5.2.2.5 OnSolarMeasureUpdate . . . 13

5.2.2.6 OnWindowClosed . . . 13

5.2.2.7 StartUpdateThread . . . 14

5.2.3 Member Data Documentation . . . 14

5.2.3.1 display . . . 14
5.2.3.2 gauge . . . 14
5.2.3.3 move . . . 14
5.2.3.4 position . . . 14
5.3 tiny26.SolarEvent Class Reference . . . 14
5.3.1 Detailed Description . . . 15
5.3.2 Member Function Documentation . . . 15
5.3.2.1 GetValue . . . 15
5.4 tiny26.SolarTrackingThread Class Reference . . . 15
5.4.1 Detailed Description . . . 15
5.4.2 Member Function Documentation . . . 16
5.4.2.1 QuitThread . . . 16
5.4.2.2 run . . . 16
5.5 tiny26.USB_Interface Class Reference . . . 16
5.5.1 Detailed Description . . . 17
5.5.2 Member Function Documentation . . . 17
5.5.2.1 CloseConnection . . . 17
5.5.2.2 FindMostSun . . . 17
5.5.2.3 IsConnected . . . 17
5.5.2.4 MeasureSolarPanel . . . 17
5.5.2.5 SetSolarPosition . . . 18
5.5.2.6 ShowErrorMsg . . . 18

Index **18**

Chapter 1

Namespace Index

1.1 Packages

Here are the packages with brief descriptions (if available):

gui 7
launcher 7
tiny26 8

Chapter 2

Hierarchical Index

2.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

App
    gui.GUI_Launcher 11
Frame
    gui.MotorGUI 11
PyCommandEvent
    tiny26.SolarEvent 14
tiny26.USB_Interface 16
Thread
    tiny26.SolarTrackingThread 15

Chapter 3

Class Index

3.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

gui.GUI_Launcher
: Abstracts away the creation of the MotorGUI and launches it in the default constructor 11

gui.MotorGUI
: Defines the main frame for the GUI 11

tiny26.SolarEvent
: Event class that allows a background thread to send messages to GUI to tell it to update itself . 14

tiny26.SolarTrackingThread
: Contains the definition for the background thread that polls the ADC input from the solar panel and sends that value to the GUI to be displayed to the user 15

tiny26.USB_Interface
: The usb interface for the OSUisp2 and the Tiny261 is contained in this class 16

Chapter 4

Namespace Documentation

4.1 gui Namespace Reference

Classes

- class MotorGUI

  Defines the main frame for the GUI.
- class GUI_Launcher

  Abstracts away the creation of the MotorGUI and launches it in the default constructor.

4.1.1 Detailed Description

```
Defines the GUI that abstracts away the connection with the OSUisp2 and Tiny261
Created on Aug 13, 2012

@author: Robert L. Phillips III
@version: 1.0

@section DEPENDENCIES
    -  wxPython
          The GUI is generated using the third party library wxPython.  All documentation
          for the wxPython library can be found at wxpython.org.

    -  tiny26
          Contains the libraries necessary for the GUI to establish a connection to the
          Tiny26 and send commands to it.

    -  time
          Contains the sleep function, which is needed to block certain threads for a certain
          amount of time
```

4.2 launcher Namespace Reference

Variables

- tuple app GUI_Launcher(0)

  New instance of the application class.

4.2.1 Detailed Description

```
Launches the MotorGUI
```

```
Created on Aug 17, 2012

@author: Robert L. Phillips III
@version: 1.0

@section DEPENDENCIES
    -  python target
  Python 2.7 (x86) (http://www.python.org/getit/)

    -  wxPython version
  wxPython2.8-win32-unicode-py27 (http://wxpython.org/download.php)

    -  build environment
  This project was originally coded and built using Eclipse Indigo and the pydev
  extension.  The third party library wxPython was used in order to generate the
  GUI and its components
```

4.3 tiny26 Namespace Reference

Classes

- class USB_Interface

  The usb interface for the OSUisp2 and the Tiny261 is contained in this class.
- class SolarTrackingThread

  Contains the definition for the background thread that polls the ADC input from the solar panel and sends that value to the GUI to be displayed to the user.
- class SolarEvent

  Event class that allows a background thread to send messages to GUI to tell it to update itself.

Variables

- int vid 0x16c0

  Vendor ID of the OSUisp2 programmer.
- int pid 0x05df

  Product ID of the OSUisp2 programmer.

4.3.1 Detailed Description

```
The classes necessary to interface with the Tiny261 and OSUisp2

Created on Aug 17, 2012

@author: Robert L. Phillips III
@version: 1.0

@section DEPENDENCIES
    -  tiny26usb.dll
         Contains the following methods that communicate commands to the Tiny26 microcontroller via
         the OSUisp2 programmer (See usb.c, usb.h and the OSUisp2 firmware in order to understand more
         about how the programmer acts as a middleman between the computer and Tiny26):

                int UsbOpenDevice(usbDevice_t **device, int vendor, char *vendorName, int product, char *produ
                int UsbEstablishConnection(usbDevice_t *dev)
                int UsbUpdateServoPosition(usbDevice_t *dev, int speed)
                void UsbCloseDevice(usbDevice_t * dev)

    -  wxPython
         In order to show error messages, this module relies on wxPython to generate and show message
         boxes to the user

    -  ctypes
         Because the usb interface is written in C, ctypes is used to open up the tiny26usb DLL and
```

```
use the methods that have been exported to it in a Python module.  Therefore, the previously
written Tiny26 interface did not need to be ported to Python, but the C code needed to be
modified to be exported as a DLL and called from Python code.
```

4.3.2 Variable Documentation

4.3.2.1 int tiny26.vid 0x16c0

Vendor ID of the OSUisp2 programmer.

Chapter 5

Class Documentation

5.1 gui.GUI_Launcher Class Reference

Abstracts away the creation of the MotorGUI and launches it in the default constructor.

Inheritance diagram for gui.GUI_Launcher:

Public Member Functions

- def OnInit

  Constuctor for a wxPython app.

5.1.1 Detailed Description

Abstracts away the creation of the MotorGUI and launches it in the default constructor.

The documentation for this class was generated from the following file:

- gui.py

5.2 gui.MotorGUI Class Reference

Defines the main frame for the GUI.

Inheritance diagram for gui.MotorGUI:

Public Member Functions

- def __init__

  Constructor for a wxPython frame.
- def StartUpdateThread

  Initiates the thread that polls the Tiny for the value of the solar panel ADC and then updates the GUI accordingly.
- def OnMoveServoRequest

  Handles the event that is fired when the "Move Servo" button is pressed.

- def OnSolarMaxRequest

  Handles the event that is fired when the "Maximize Solar Efficiency" button is pressed.
- def OnButtonPressed

  Handles the event that is fired when the "Connect" button is pressed.
- def OnWindowClosed

  Handles the event that is called when the program is closed.
- def OnSolarMeasureUpdate

  Handles the event that is called when a new ADC measurment from the solar panel connected to the tiny is received.
- def OnAboutRequested

  Handles the event that is called when the about menu button is pressed.

Public Attributes

- solar_event

  Holds the event type definition for the event that is called when the GUI needs to be updated with the latest solar panel measurement.
- usb

  Holds the usb connection variable that is used to interface with the Tiny261.
- connect

  The "Connect" button.
- find_sun

  The "Maximize Solar Efficiency" button.
- position

  The "Position" text control (allows the user to specifiy a position to move the servo).
- display

  The "Display" text control (shows the solar panel measurements).
- move

  The "Move Servo" button.
- gauge

  The solar measurements Gauge (displays the measurements in gauge form).
- updater

  Holds the object that corresponds to the solar panel measurements update thread.

5.2.1 Detailed Description

Defines the main frame for the GUI.

It provides a simple interface for the user to send commands to a Tiny261 microcontroller (that has the approriate firmware) via the OSUisp2 Universal Programmer. The commands move a solar panel to certain positions and take measurements that correspond to the amount of light that is hitting the solar panel. Those measurements are echoed back to the computer, which are displayed on the GUI.

5.2.2 Member Function Documentation

5.2.2.1 def gui.MotorGUI.OnAboutRequested (*self, event*)

Handles the event that is called when the about menu button is pressed.

Shows the user a message box containing information about the program

Parameters

self,:	The object pointer
event,:	The object that is associated with the event request

5.2.2.2 def gui.MotorGUI.OnButtonPressed (*self, event*)

Handles the event that is fired when the "Connect" button is pressed.

If a usb connection is not currently active, the GUI sends a request to the OSUisp2 programmer to connect to the Tiny and verify that the correct firmware is installed.

Parameters

self,:	The object pointer
event,:	The object that is associated with the event request

5.2.2.3 def gui.MotorGUI.OnMoveServoRequest (*self, event*)

Handles the event that is fired when the "Move Servo" button is pressed.

If a usb connection is currently active, a request is sent to the Tiny to move the motor to the position that corresponds to value given by the user in the position text control. The solar measurement update thread is paused until the Tiny26 is finished repositioning the motor

Parameters

self,:	The object pointer
event,:	The object that is associated with the event request

5.2.2.4 def gui.MotorGUI.OnSolarMaxRequest (*self, event*)

Handles the event that is fired when the "Maximize Solar Efficiency" button is pressed.

If a usb connection is currently active, a request to find the position that produces the highest voltage output from the solar panel is sent. The solar panel measurement update thread is paused until the Tiny had finished positioning the solar panel in the position that receives the most light.

Parameters

self,:	The object pointer
event,:	The object that is associated with the event request

5.2.2.5 def gui.MotorGUI.OnSolarMeasureUpdate (*self, event*)

Handles the event that is called when a new ADC measurment from the solar panel connected to the tiny is received.

Sets the value of the text control equal to the measurement that is contained in the event request.

Parameters

self,:	The object pointer
event,:	The object that is associated with the event request

5.2.2.6 def gui.MotorGUI.OnWindowClosed (*self, event*)

Handles the event that is called when the program is closed.

If a usb connection is currently active, the GUI sends a request to the OSUisp2 programmer to close the connection to the Tiny and to end the thread that updates the GUI with the solar panel measurments

Parameters

self,:	The object pointer
event,:	The object that is associated with the event request

5.2.2.7 def gui.MotorGUI.StartUpdateThread (*self*)

Initiates the thread that polls the Tiny for the value of the solar panel ADC and then updates the GUI accordingly.

Parameters

self,:	The object pointer

5.2.3 Member Data Documentation

5.2.3.1 gui.MotorGUI.display

The "Display" text control (shows the solar panel measurements).

5.2.3.2 gui.MotorGUI.gauge

The solar measurements Gauge (displays the measurements in gauge form).

5.2.3.3 gui.MotorGUI.move

The "Move Servo" button.

5.2.3.4 gui.MotorGUI.position

The "Position" text control (allows the user to specifiy a position to move the servo).

The documentation for this class was generated from the following file:

- gui.py

5.3 tiny26.SolarEvent Class Reference

Event class that allows a background thread to send messages to GUI to tell it to update itself.

Inheritance diagram for tiny26.SolarEvent:

Public Member Functions

- def __init__

  Constructor for the solar event.
- def GetValue

  Returns the value of the event that is set in the constructor.

5.3.1 Detailed Description

Event class that allows a background thread to send messages to GUI to tell it to update itself.

5.3.2 Member Function Documentation

5.3.2.1 def tiny26.SolarEvent.GetValue (*self*)

Returns the value of the event that is set in the constructor.

Parameters

self,:	The object pointer

The documentation for this class was generated from the following file:

- tiny26.py

5.4 tiny26.SolarTrackingThread Class Reference

Contains the definition for the background thread that polls the ADC input from the solar panel and sends that value to the GUI to be displayed to the user.

Inheritance diagram for tiny26.SolarTrackingThread:

Public Member Functions

- def __init__

  Constructor for the solar tracking thread.
- def run

  Called when the thread is started.
- def QuitThread

  Sets the stop flag to stop the thread.

Public Attributes

- usb

  The usb interface associated with the thread.
- event

  The event that needs to be fired when the GUI needs to be updated.
- parent

  The parent frame that should be updated when a new solar measurement is received.
- stop

  The stop flag for the thread.

5.4.1 Detailed Description

Contains the definition for the background thread that polls the ADC input from the solar panel and sends that value to the GUI to be displayed to the user.

5.4.2 Member Function Documentation

5.4.2.1 def tiny26.SolarTrackingThread.QuitThread (*self*)

Sets the stop flag to stop the thread.

Sets the stop flag, which causes the loop in the thread to stop if it was running

Parameters

self,:	The object pointer

5.4.2.2 def tiny26.SolarTrackingThread.run (*self*)

Called when the thread is started.

As long as the stop flag is not set, the thread sends a request to the Tiny for solar measurements. When the computer receives a response from the Tiny, it fires a SolarEvent, which tells the GUI to update itself with the latest solar measurements.

Parameters

self,:	The object pointer

The documentation for this class was generated from the following file:

- tiny26.py

5.5 tiny26.USB_Interface Class Reference

The usb interface for the OSUisp2 and the Tiny261 is contained in this class.

Public Member Functions

- def __init__

  Constructor for the usb interface.
- def IsConnected

  Returns the connection status.
- def MeasureSolarPanel

  Reads the output of the solar panel.
- def FindMostSun

  Moves the solar panel to its optimal position.
- def SetSolarPosition

  Moves the solar panel to the specified position.
- def ShowErrorMsg

  Displays an error message box with the supplied message and title.
- def CloseConnection

  Cleans up the connection to the Tiny26.

Public Attributes

- window

  The frame that is associated with the USB interface.

- lib

  tiny26usb.dll library
- device

  *usbDevice_t * needed to establish a connection with OSUisp2 and the Tiny*
- connected

  Holds the connection status.

5.5.1 Detailed Description

The usb interface for the OSUisp2 and the Tiny261 is contained in this class.

It contains all the methods for establishing connections, closing connections, and sending request to the Tiny261 to move the servo and to get the ADC measurements from the solar panel.

5.5.2 Member Function Documentation

5.5.2.1 def tiny26.USB_Interface.CloseConnection (*self*)

Cleans up the connection to the Tiny26.

Sends a request to the OSUisp2 Universal Programmer to disconnect from the Tiny

Parameters

self,:	The object pointer

5.5.2.2 def tiny26.USB_Interface.FindMostSun (*self*)

Moves the solar panel to its optimal position.

Sends a request to the Tiny to search for the position that gives the solar panel the most light. The Tiny does not respond to any movement requests while it is searching for the solar panel's optimal position

Parameters

self,:	The object pointer

5.5.2.3 def tiny26.USB_Interface.IsConnected (*self*)

Returns the connection status.

Parameters

self,:	The object pointer

5.5.2.4 def tiny26.USB_Interface.MeasureSolarPanel (*self*)

Reads the output of the solar panel.

Sends a request to the Tiny to keep the motor in the current position respond with the light measurements

Parameters

self,:	The object pointer

5.5.2.5 def tiny26.USB_Interface.SetSolarPosition (*self, position*)

Moves the solar panel to the specified position.

Sends a request to the Tiny to move the servo to the specified position. The Tiny will not respond to any requests while it is moving the servo. If the given position is out of the valid range, the servo moves to 254 or 1 depending on which extrema was violated

Parameters

self,:	The object pointer
position,:	The desired position (0 $<$ position $<$ 255)

5.5.2.6 def tiny26.USB_Interface.ShowErrorMsg (*self, message, title*)

Displays an error message box with the supplied message and title.

Parameters

self,:	The object pointer
message,:	The desired message for the Message Box
title,:	The desired title for the Message Box

The documentation for this class was generated from the following file:

- tiny26.py

Index

CloseConnection
    tiny26::USB_Interface, 17

display
    gui::MotorGUI, 14

FindMostSun
    tiny26::USB_Interface, 17

gauge
    gui::MotorGUI, 14
GetValue
    tiny26::SolarEvent, 15
gui, 7
gui.GUI_Launcher, 11
gui.MotorGUI, 11
gui::MotorGUI
    display, 14
    gauge, 14
    move, 14
    OnAboutRequested, 12
    OnButtonPressed, 13
    OnMoveServoRequest, 13
    OnSolarMaxRequest, 13
    OnSolarMeasureUpdate, 13
    OnWindowClosed, 13
    position, 14
    StartUpdateThread, 14

IsConnected
    tiny26::USB_Interface, 17

launcher, 7

MeasureSolarPanel
    tiny26::USB_Interface, 17
move
    gui::MotorGUI, 14

OnAboutRequested
    gui::MotorGUI, 12
OnButtonPressed
    gui::MotorGUI, 13
OnMoveServoRequest
    gui::MotorGUI, 13
OnSolarMaxRequest
    gui::MotorGUI, 13
OnSolarMeasureUpdate
    gui::MotorGUI, 13
OnWindowClosed
    gui::MotorGUI, 13

position
    gui::MotorGUI, 14

QuitThread
    tiny26::SolarTrackingThread, 16

run
    tiny26::SolarTrackingThread, 16

SetSolarPosition
    tiny26::USB_Interface, 17
ShowErrorMsg
    tiny26::USB_Interface, 18
StartUpdateThread
    gui::MotorGUI, 14

tiny26, 8
    vid, 9
tiny26.SolarEvent, 14
tiny26.SolarTrackingThread, 15
tiny26.USB_Interface, 16
tiny26::SolarEvent
    GetValue, 15
tiny26::SolarTrackingThread
    QuitThread, 16
    run, 16
tiny26::USB_Interface
    CloseConnection, 17
    FindMostSun, 17
    IsConnected, 17
    MeasureSolarPanel, 17
    SetSolarPosition, 17
    ShowErrorMsg, 18

vid
    tiny26, 9