Driver Board Dropwatcher

Generated by Doxygen 1.9.6

Chapter 1

Namespace Index

1.1 Namespace List

Here is a list of all documented namespaces with brief descriptions:

Chapter 2

Hierarchical Index

2.1 Class Hierarchy

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

Chapter 3

Class Index

3.1 Class List

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

Chapter 4

File Index

4.1 File List

Here is a list of all documented files with brief descriptions:

Chapter 5

Namespace Documentation

5.1 DriverBoardDropwatcher Namespace Reference

Classes

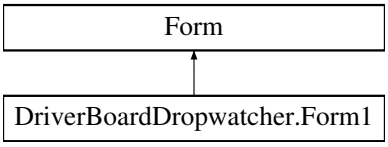
- class Form1
- class **Program**

Chapter 6

Class Documentation

6.1 DriverBoardDropwatcher.Form1 Class Reference

Inheritance diagram for DriverBoardDropwatcher.Form1:



Static Public Member Functions

- static Bitmap MakeGrayscale3 (Bitmap original)

  Grey Scale Image Function.

Protected Member Functions

- override void Dispose (bool disposing)

  Clean up any resources being used.

Private Member Functions

- void ThreadTask ()

  A Thread Task Function.
- void cbSerialPort_DropDown (object sender, EventArgs e)

  Detects Active Serial COM Ports from device.
- void cbSerialPort_SelectedIndexChanged (object sender, EventArgs e)

  Detects Changes in Serial Ports.
- void disconnect_board ()

  Disconnect Driver Board.
- void connect_board ()

  Connect Driver Board.

- void DataRecievedHandler (object sender, SerialDataReceivedEventArgs e)

  Processes data received from driver board.
- void determineStatus ()

  Determine the Status of Voltages and Temperatures.
- void **parseJsonData** (string input_string)
- void **btnConnectDisconnect_Click** (object sender, EventArgs e)
- void **powerOnOff_Click** (object sender, EventArgs e)
- void **powerOn** ()
- void **powerOff** ()
- void **voltage_ValueChanged** (object sender, EventArgs e)
- void **temperature_ValueChanged** (object sender, EventArgs e)
- void **frequencyValue_ValueChanged** (object sender, EventArgs e)
- void **cbDropWatchMode_SelectedIndexChanged** (object sender, EventArgs e)
- void **reset_Click** (object sender, EventArgs e)
- void **cbDropWatchHeadSelection_SelectedIndexChanged** (object sender, EventArgs e)
- void **ImageBoxClicked** (object sender, EventArgs e)
- void **pictureBox1_Click** (object sender, EventArgs e)
- void **pictureBox2_Click** (object sender, EventArgs e)
- void **pictureBox3_Click** (object sender, EventArgs e)
- void **pictureBox4_Click** (object sender, EventArgs e)
- void **convertImageToData** (string file_name)
- void **btnPrintImage_Click** (object sender, EventArgs e)
- void **PrintingImage** (int head)
- void **VerifyImageData** (int head, string existing_lines)
- void **NozzleValue_ValueChanged** (object sender, EventArgs e)
- void **SpanValue_ValueChanged** (object sender, EventArgs e)
- void **FillSingleNozzleButton_Click** (object sender, EventArgs e)
- void **FillSpanNozzleButton_Click** (object sender, EventArgs e)
- void **ImageModeSelection_SelectedIndexChanged** (object sender, EventArgs e)
- void **ClearHeadsButton_Click** (object sender, EventArgs e)
- void **FillCycleA_Click** (object sender, EventArgs e)
- void **FillCycleB_Click** (object sender, EventArgs e)
- void **FillCycleC_Click** (object sender, EventArgs e)
- void **FillCycle** (object sender, EventArgs e)
- void **GapValue_ValueChanged** (object sender, EventArgs e)
- void **FillGapButton_Click** (object sender, EventArgs e)
- void **fillHead_Click** (object sender, EventArgs e)
- void **PD_Polarity_SelectedIndexChanged** (object sender, EventArgs e)
- void **EncoderTrackedPositionSelection_SelectedIndexChanged** (object sender, EventArgs e)
- void **pdDirection_SelectedIndexChanged** (object sender, EventArgs e)
- void **tcDropWatchingAndImageModes_SelectedIndexChanged** (object sender, EventArgs e)
- void **DropWatchingTab_PreviewKeyDown** (object sender, PreviewKeyDownEventArgs e)
- void **DropWatchingTab_Click** (object sender, EventArgs e)
- void **btnCancel_Click** (object sender, EventArgs e)
- void **DeleteAllFilesInFolder** (object sender, EventArgs e)
- void **Form1_Load** (object sender, EventArgs e)
- void **Form1_FormClosing** (object sender, FormClosingEventArgs e)
- void **InitializeComponent** ()

  Required method for Designer support - do not modify the contents of this method with the code editor.

Private Attributes

- OpenFileDialog **ofd**
- bool **valid_port_selected** = false
- string **port_name**
- int **failCounter** = 0
- int **activeDropWatch**
- int **activeDropModeHead**
- int **activeNozzleValue**
- int **activeSpanValue**
- int **activeImageMode**
- int **activeGapValue**
- int **actFreq**
- int **timeBoardOn** = -1
- int **activePD_Polarity**
- int **activeEncoderPosition**
- int **activePDdirection**
- int **activeImageHeadIndex**
- bool **ImageHead1** = false
- bool **ImageHead2** = false
- bool **ImageHead3** = false
- bool **ImageHead4** = false
- bool **Head1ImageSend** = false
- bool **Head2ImageSend** = false
- bool **Head3ImageSend** = false
- bool **Head4ImageSend** = false
- bool **isRunning** = true
- String **CurrentFileName**
- String **datafolder** = Application.StartupPath.Replace("bin\\Debug", "Output Images\\File")
- String **newpath** = Application.StartupPath.Replace("bin\\Debug", "Output Images")
- String **dataFolderPath** = Application.StartupPath.Replace("bin\\Debug", "Output Images")
- MemoryStream **outputStream**
- int[ ] **HeadPrintCountersStoredAsInt** = new int[4]
- int[ ] **PreviousHeadPrintCounters** = new int[4]
- int[ ] **HeadStatus** = new int[4]
- byte[ ] **A_Bits** = { 0b10010010, 0b01001001, 0b00100100 }
- byte[ ] **B_Bits** = { 0b01001001, 0b00100100, 0b10010010 }
- byte[ ] **C_Bits** = { 0b00100100, 0b10010010, 0b01001001 }
- byte[ ] **BitsArray**
- System.ComponentModel.IContainer **components** = null

  Required designer variable.
- System.Windows.Forms.ComboBox **cbSerialPort**
- System.Windows.Forms.Button **btnConnectDisconnect**
- System.Windows.Forms.CheckBox **isConnected**
- System.Windows.Forms.Button **powerOnOff**
- System.Windows.Forms.CheckBox **power**
- System.Windows.Forms.Label **Status**
- System.Windows.Forms.TextBox **txtbStatusBox**
- System.Windows.Forms.Label **label4**
- System.Windows.Forms.TextBox **txtbBoardUpTime**
- System.Windows.Forms.TextBox **txtbTemperatureOutput1**
- System.Windows.Forms.NumericUpDown **nudTemperatureHead1**
- System.Windows.Forms.NumericUpDown **nudVoltageHead1**
- System.Windows.Forms.TextBox **txtbTemperatureOutput2**
- System.Windows.Forms.NumericUpDown **nudTemperatureHead2**

- System.Windows.Forms.NumericUpDown **nudVoltageHead2**
- System.Windows.Forms.TextBox **txtbTemperatureOutput3**
- System.Windows.Forms.NumericUpDown **nudTemperatureHead3**
- System.Windows.Forms.NumericUpDown **nudVoltageHead3**
- System.Windows.Forms.TextBox **txtbTemperatureOutput4**
- System.Windows.Forms.NumericUpDown **nudTemperatureHead4**
- System.Windows.Forms.NumericUpDown **nudVoltageHead4**
- System.Windows.Forms.TextBox **Head2TextStatus**
- System.Windows.Forms.TextBox **Head3TextStatus**
- System.Windows.Forms.TextBox **Head1TextStatus**
- System.Windows.Forms.TabControl **tcDropWatchingAndImageModes**
- System.Windows.Forms.Button **btnReset**
- System.Windows.Forms.TabPage **DropWatchingTab**
- System.Windows.Forms.NumericUpDown **nudFrequency**
- System.Windows.Forms.NumericUpDown **nudNozzle**
- System.Windows.Forms.ComboBox **cbDropWatchHeadSelection**
- System.Windows.Forms.Label **label24**
- System.Windows.Forms.ComboBox **cbDropWatchMode**
- System.Windows.Forms.TabPage **ImageModeTab**
- System.Windows.Forms.NumericUpDown **nudSpan**
- System.Windows.Forms.Label **label26**
- System.Windows.Forms.Label **label25**
- System.Windows.Forms.Label **label32**
- System.Windows.Forms.Label **label30**
- System.Windows.Forms.Button **btnFillCycleA**
- System.Windows.Forms.TableLayoutPanel **tableLayoutPanel1**
- System.Windows.Forms.Label **label36**
- System.Windows.Forms.Label **label35**
- System.Windows.Forms.Label **label33**
- System.Windows.Forms.Label **label34**
- System.Windows.Forms.Label **label40**
- System.Windows.Forms.Label **label37**
- System.Windows.Forms.Label **label39**
- System.Windows.Forms.Label **label38**
- System.Windows.Forms.Label **label41**
- System.Windows.Forms.TextBox **txtbPrintCounter4**
- System.Windows.Forms.Label **label2**
- System.Windows.Forms.TextBox **txtbPrintCounter1**
- System.Windows.Forms.TextBox **txtbPrintCounter2**
- System.Windows.Forms.TextBox **txtbPrintCounter3**
- System.Windows.Forms.Label **frequencyLabel**
- System.Windows.Forms.PictureBox **pictureBox1**
- System.Windows.Forms.PictureBox **pictureBox4**
- System.Windows.Forms.PictureBox **pictureBox3**
- System.Windows.Forms.PictureBox **pictureBox2**
- System.Windows.Forms.Label **label7**
- System.Windows.Forms.Label **label6**
- System.Windows.Forms.Label **label5**
- System.Windows.Forms.Label **label3**
- System.Windows.Forms.OpenFileDialog **openFileDialog1**
- System.Windows.Forms.TextBox **FileName4**
- System.Windows.Forms.TextBox **FileName3**
- System.Windows.Forms.TextBox **FileName2**
- System.Windows.Forms.TextBox **FileName1**
- System.Windows.Forms.Label **label11**

- System.Windows.Forms.Label **label10**
- System.Windows.Forms.Label **label9**
- System.Windows.Forms.Label **label8**
- System.Windows.Forms.TextBox **ImageSizeText4**
- System.Windows.Forms.Label **label15**
- System.Windows.Forms.TextBox **ImageSizeText3**
- System.Windows.Forms.Label **label14**
- System.Windows.Forms.TextBox **ImageSizeText2**
- System.Windows.Forms.Label **label13**
- System.Windows.Forms.TextBox **ImageSizeText1**
- System.Windows.Forms.Label **label12**
- System.Windows.Forms.TextBox **Head4TextStatus**
- System.Windows.Forms.Label **frequencyDuplicateLabel**
- System.Windows.Forms.TextBox **txtbFrequencyDuplicate**
- System.Windows.Forms.Button **btnClearHead**
- System.Windows.Forms.Button **btnPrintImage**
- System.Windows.Forms.Button **btnCancel**
- System.Windows.Forms.Button **btnFillNozzle**
- System.Windows.Forms.Button **btnFillSpan**
- System.Windows.Forms.Label **label1**
- System.Windows.Forms.ComboBox **ImageModeSelection**
- System.Windows.Forms.TextBox **txtbHeadStatus**
- System.Windows.Forms.Label **label16**
- System.Windows.Forms.Button **btnFillCycleC**
- System.Windows.Forms.Button **btnFillCycleB**
- System.Windows.Forms.GroupBox **FillCycleBox**
- System.Windows.Forms.Button **btnFillGap**
- System.Windows.Forms.NumericUpDown **nudGap**
- System.Windows.Forms.Label **label17**
- System.Windows.Forms.ToolTip **toolTip1**
- System.Windows.Forms.Button **btnFillHead**
- System.Windows.Forms.ComboBox **PD_Polarity**
- System.Windows.Forms.Label **polarityLabel**
- System.Windows.Forms.ComboBox **EncoderTrackedPositionSelection**
- System.Windows.Forms.Label **EncoderPositionLabel**
- System.Windows.Forms.ComboBox **pdDirection**
- System.Windows.Forms.Label **pdDirectionLabel**
- System.Windows.Forms.CheckBox **isFillSpan**
- System.Windows.Forms.CheckBox **isFillGap**
- System.Windows.Forms.CheckBox **isFillHead**
- System.Windows.Forms.CheckBox **isFillNozzle**
- System.Windows.Forms.TextBox **txtbImageHeadStatus**

Static Private Attributes

- static SerialPort **driver_board**

6.1.1 Member Function Documentation

6.1.1.1 cbSerialPort_DropDown()

```
void DriverBoardDropwatcher.Form1.cbSerialPort_DropDown (
            object sender,
            EventArgs e )  [inline], [private]
```

Detects Active Serial COM Ports from device.

This function searches and detects any available COM Ports available to open.

Parameters

sender	The object that raises the event
e	The event arguments

6.1.1.2 cbSerialPort_SelectedIndexChanged()

```
void DriverBoardDropwatcher.Form1.cbSerialPort_SelectedIndexChanged (
            object sender,
            EventArgs e )  [inline], [private]
```

Detects Changes in Serial Ports.

This function runs when the Serial COM Port is changed.

6.1.1.3 connect_board()

```
void DriverBoardDropwatcher.Form1.connect_board ( )  [inline], [private]
```

Connect Driver Board.

This function sends command to driver board to connect and checks the checkbox in the GUI. Only runs if a valid port is selected and is connected successfully to the driver board. If this fails to run, then an Error Dialog Box shows up.

6.1.1.4 DataRecievedHandler()

```
void DriverBoardDropwatcher.Form1.DataRecievedHandler (
            object sender,
            SerialDataReceivedEventArgs e )  [inline], [private]
```

Processes data received from driver board.

This function reads the data received from the driver board. If the substring matches to the expected character, then the data is sent to parse. If this fails to run multipe times, then an Error Dialog Box shows up.

6.1.1.5 determineStatus()

```
void DriverBoardDropwatcher.Form1.determineStatus ( )  [inline], [private]
```

Determine the Status of Voltages and Temperatures.

This function determines what state each Head is in and outputs the Voltage and Temperature Values to the GUI. Only runs if a valid port is selected and is connected and powered on successfully to the driver board.

6.1.1.6 disconnect_board()

```
void DriverBoardDropwatcher.Form1.disconnect_board ( )  [inline], [private]
```

Disconnect Driver Board.

This function sends command to driver board to disconnect and unchecks the checkbox in the GUI.

6.1.1.7 Dispose()

```
override void DriverBoardDropwatcher.Form1.Dispose (
            bool disposing )  [inline], [protected]
```

Clean up any resources being used.

Parameters

disposing	true if managed resources should be disposed; otherwise, false.

6.1.1.8 MakeGrayscale3()

```
static Bitmap DriverBoardDropwatcher.Form1.MakeGrayscale3 (
            Bitmap original )  [inline], [static]
```

Grey Scale Image Function.

This function grey scales any image passed to it

Parameters

original	Original Image (Coloured)

Returns

Grey-Scaled Image in Bitmap format

6.1.1.9 ThreadTask()

```
void DriverBoardDropwatcher.Form1.ThreadTask ( )  [inline], [private]
```

A Thread Task Function.

This function constantly runs in the background to ensure the board is connected at all times. It sends the command 'b' to receive relevent information such as voltage, temperature etc If this fails to run, then Error Message Box pops up signalling an error.

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

- Form1.cs
- Form1.Designer.cs

Chapter 7

File Documentation

7.1 Form1.cs File Reference

Classes

- class DriverBoardDropwatcher.Form1

Functions

- **[instance initializer]**

Variables

- $ **v**
- $ **T**
- $ **p**
- $ **n**
- $ **N**
- $ **I**

7.1.1 Detailed Description

This C# Code contains the Source Code for the Driver Board Dropwatcher Application.