

USB

Version 2. Copyright (c) 2023 Terrortronics / Bradley Elenbaas. All rights reserved.

Generated by Doxygen 1.9.7

1 Hierarchical Index	1
1.1 Class Hierarchy	1
2 Class Index	3
2.1 Class List	3
3 File Index	5
3.1 File List	5
4 Class Documentation	7
4.1 USB Class Reference	7
4.1.1 Detailed Description	9
4.1.2 Member Enumeration Documentation	9
4.1.2.1 IncomingOption	9
4.1.2.2 OutgoingMessages	10
4.1.3 Member Function Documentation	10
4.1.3.1 CatenateSerialPrefixAndPostfix()	10
4.1.3.2 CloseSerialPorts()	10
4.1.3.3 ConnectSerialPort()	11
4.1.3.4 ConnectSerialPorts()	11
4.1.3.5 InvokedFunctionStart()	11
4.1.3.6 InvokedFunctionStop()	11
4.1.3.7 OnApplicationQuit()	12
4.1.3.8 OnGUI()	12
4.1.3.9 OutgoingMessageToString()	12
4.1.3.10 ReceiveData()	12
4.1.3.11 ReceiveDatas()	12
4.1.3.12 ReceiveOptionToString()	13
4.1.3.13 ResetArduinos()	13
4.1.3.14 Start()	13
4.1.3.15 Teardown()	13
4.1.3.16 ThreadedFunction()	13
4.1.3.17 Update()	13
4.1.4 Member Data Documentation	14
4.1.4.1 debug	14
4.1.4.2 INCOMING_IN_INVOKED_FUNTION	14
4.1.4.3 INCOMING_IN_THREADED_FUNCTION	14
4.1.4.4 INCOMING_IN_UPDATE_FUNCTION	14
4.1.4.5 INCOMING_IN_USER_INTERFACE_FUNCTION	14
4.1.4.6 incomingOption	14
4.1.4.7 INVOKED_FUNCTION	14
4.1.4.8 invokedFunctionDelay	15
4.1.4.9 mutex	15

4.1.4.10 OUTGOING_RESET	15
4.1.4.11 OUTGOING_START	15
4.1.4.12 SERIAL_POSTFIX_MAX_LINUX	15
4.1.4.13 SERIAL_POSTFIX_MAX_MAC	15
4.1.4.14 SERIAL_POSTFIX_MAX_WINDOWS	15
4.1.4.15 SERIAL_POSTFIX_MIN_LINUX	15
4.1.4.16 SERIAL_POSTFIX_MIN_MAC	16
4.1.4.17 SERIAL_POSTFIX_MIN_WINDOWS	16
4.1.4.18 SERIAL_PREFIX_LINUX	16
4.1.4.19 SERIAL_PREFIX_MAC	16
4.1.4.20 SERIAL_PREFIX_WINDOWS	16
4.1.4.21 SERIAL_TIMEOUT	16
4.1.4.22 serialBaudRates	16
4.1.4.23 serialPorts	17
4.1.4.24 threadedFunction	17
4.1.5 Property Documentation	17
4.1.5.1 Configs	17
4.1.5.2 IncomingData	17
5 File Documentation	19
5.1 USB.cs File Reference	19
Index	21

Chapter 1

Hierarchical Index

1.1 Class Hierarchy

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

MonoBehaviour	
USB	7

Chapter 2

Class Index

2.1 Class List

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

USB	The USB library	7
---------------------	-------------------------------------------	-------------------

Chapter 3

File Index

3.1 File List

Here is a list of all files with brief descriptions:

USB.cs	19
----------------------------------	----

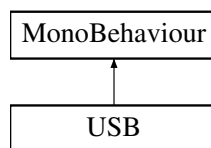
Chapter 4

Class Documentation

4.1 USB Class Reference

The [USB](#) library.

Inheritance diagram for USB:



Public Types

- enum [IncomingOption](#) { [UpdateFunction](#) = INCOMING_IN_UPDATE_FUNCTION , [InvokedFunction](#) = INCOMING_IN_INVOKED_FUNTION , [ThreadedFunction](#) = INCOMING_IN_THREADED_FUNCTION , [UserInterfaceFunction](#) = INCOMING_IN_USER_INTERFACE_FUNCTION }
- Enum of options for how to receive incoming data.*
- enum [OutgoingMessages](#) { [Start](#) = OUTGOING_START , [Reset](#) = OUTGOING_RESET }
- An enum of messages to be sent to any Arduino.*

Static Public Attributes

- static Thread [threadedFunction](#)
- A thread for [ThreadedFunction\(\)](#).*

Properties

- static string[] [Configs](#) [get, private set]
- A set of configs received from connected Arduinos.*
- static string [IncomingData](#) = "" [get, private set]
- Incoming data from thread.*

Private Member Functions

- void [Start](#) ()
Pseudo-Constructor. Calls the [ConnectSerialPorts\(\)](#) function.
- void [OnApplicationQuit](#) ()
Pseudo-Destructor. Kills the [threadedFunction](#) thread.
- void [OnGUI](#) ()
The user interface function. Receive incoming data.
- void [Update](#) ()
The main function. Receive incoming data.
- string [CatenateSerialPrefixAndPostfix](#) (int postfix)
Catenates the platform specific prefix and the iterated postfix.
- void [CloseSerialPorts](#) ()
Close all serial ports.
- SerialPort [ConnectSerialPort](#) (int index, int baudRate)
Open and return a single instance of SerialPort.
- void [ConnectSerialPorts](#) ()
Connect to all possible SerialPort connections.
- void [InvokedFunctionStart](#) ()
Start invoking the [InvokedFunction\(\)](#) function.
- void [InvokedFunctionStop](#) ()
Stop invoking the [InvokedFunction\(\)](#) function.
- string [OutgoingMessageToString](#) ([OutgoingMessages](#) message)
Transform [OutgoingMessages](#) enum into string.
- void [ReceiveData](#) (SerialPort serialPort, string catenatedSerialPort, int configIndex)
The Invoked function. Receive incoming data.
- void [ReceiveDatas](#) ()
Starts one instance of [InvokedFunction\(SerialPort, string\)](#) per [serialPorts](#) member.
- string [ReceiveOptionToString](#) ([IncomingOption](#) option)
Transform [ReceiveOption](#) enum into string.
- void [ResetArduinos](#) ()
Call the reset() function on each connected Arduino.
- void [Teardown](#) ()
Pseudo-Destructor. Stop repeating functions, and shutdown Arduinos.
- void [ThreadedFunction](#) ()
The threaded function. Receive incoming data.

Private Attributes

- bool [debug](#) = false
Check to print debug tracers to console.
- readonly int[] [serialBaudRates](#)
Baud rates (pair with Arduino).
- SerialPort[] [serialPorts](#)
Successful serial port connections.
- [IncomingOption](#) [incomingOption](#)
Choose when to receive incoming data.
- readonly Mutex [mutex](#) = new Mutex()
A mutex.
- float [invokedFunctionDelay](#) = 0.1f
Delay between [InvokedFunction\(\)](#) calls.

Static Private Attributes

- const int `SERIAL_TIMEOUT` = 1000
Timeout for incoming data (milliseconds).
- const int `SERIAL_POSTFIX_MIN_WINDOWS` = 0
Minimum COM port (Windows).
- const int `SERIAL_POSTFIX_MIN_LINUX` = 0
Minimum COM port (Linux).
- const int `SERIAL_POSTFIX_MIN_MAC` = 0
Minimum COM port (Mac).
- const int `SERIAL_POSTFIX_MAX_WINDOWS` = 256
Maximum COM port (Windows).
- const int `SERIAL_POSTFIX_MAX_LINUX` = 256
Maximum COM port (Linux).
- const int `SERIAL_POSTFIX_MAX_MAC` = 256
Maximum COM port (Mac).
- const string `SERIAL_PREFIX_WINDOWS` = "COM"
The device prefix (Windows).
- const string `SERIAL_PREFIX_LINUX` = "/dev/ttyACM"
The device prefix (Linux).
- const string `SERIAL_PREFIX_MAC` = ""
The device prefix (Mac).
- const int `INCOMING_IN_UPDATE_FUNCTION` = 0
Wait for incoming data using the `Update()` event function.
- const int `INCOMING_IN_INVOKED_FUNTION` = 1
Wait for incoming data using `InvokeRepeating(string)`.
- const int `INCOMING_IN_THREADED_FUNCTION` = 2
Wait for incoming data on a new thread.
- const int `INCOMING_IN_USER_INTERFACE_FUNCTION` = 3
Wait for incoming data on the UI thread.
- const int `OUTGOING_START` = 0
Calls the Arduino's `start()` function.
- const int `OUTGOING_RESET` = 1
Calls the Arduino's `reset()` function.
- const string `INVOKED_FUNCTION` = "ReceiveDatas"
For use with `InvokeRepeating(string, float, float)` and `CancelInvoke(string)` functions.

4.1.1 Detailed Description

The `USB` library.

4.1.2 Member Enumeration Documentation

4.1.2.1 IncomingOption

```
enum USB.IncomingOption
```

Enum of options for how to receive incoming data.

Enumerator

UpdateFunction	
InvokedFunction	
ThreadedFunction	
UserInterfaceFunction	

4.1.2.2 OutgoingMessages

enum `USB.OutgoingMessages`

An enum of messages to be sent to any Arduino.

Max range of 256. Must be paired with a similar enum on Arduino.

Enumerator

Start	
Reset	

4.1.3 Member Function Documentation**4.1.3.1 CatenateSerialPrefixAndPostfix()**

```
string USB.CatenateSerialPrefixAndPostfix (
    int postfix ) [inline], [private]
```

Catenates the platform specific prefix and the iterated postfix.

Warning: In a UNITY_EDITOR_WIN build, the UNITY_STANDALONE_WIN section is being reached erroneously.

Parameters

<i>postfix</i>	The iterated postfix.
----------------	-----------------------

Returns

Returns the catenated sum of the platform specific prefix and the iterated postfix.

4.1.3.2 CloseSerialPorts()

```
void USB.CloseSerialPorts ( ) [inline], [private]
```

Close all serial ports.

4.1.3.3 ConnectSerialPort()

```
SerialPort USB.ConnectSerialPort (
    int index,
    int baudRate ) [inline], [private]
```

Open and return a single instance of SerialPort.

Parameters

<i>index</i>	The serial port index.
<i>baudRate</i>	The serial baud rate.

Returns

An instance of SerialPort.

4.1.3.4 ConnectSerialPorts()

```
void USB.ConnectSerialPorts ( ) [inline], [private]
```

Connect to all possible SerialPort connections.

Cross-Platform Steps:

- Count the connected devices.
- Use count to define the size of [serialPorts](#) and [configs](#).
- Populate [serialPorts](#) and [configs](#).
- Start a function to receive incoming data.

Warnings: In C, the namespace Connect is already taken.

4.1.3.5 InvokedFunctionStart()

```
void USB.InvokedFunctionStart ( ) [inline], [private]
```

Start invoking the [InvokedFunction\(\)](#) function.

4.1.3.6 InvokedFunctionStop()

```
void USB.InvokedFunctionStop ( ) [inline], [private]
```

Stop invoking the [InvokedFunction\(\)](#) function.

4.1.3.7 OnApplicationQuit()

```
void USB.OnApplicationQuit ( ) [inline], [private]
```

Pseudo-Destructor. Kills the [threadedFunction](#) thread.

4.1.3.8 OnGUI()

```
void USB.OnGUI ( ) [inline], [private]
```

The user interface function. Receive incoming data.

4.1.3.9 OutgoingMessageToString()

```
string USB.OutgoingMessageToString (
    OutgoingMessages message ) [inline], [private]
```

Transform [OutgoingMessages](#) enum into string.

Parameters

<i>message</i>	OutgoingMessages member to transform.
----------------	-------------------------------------------------------

Returns

A string transformed from a [OutgoingMessages](#) member.

4.1.3.10 ReceiveData()

```
void USB.ReceiveData (
    SerialPort serialPort,
    string catenatedSerialPort,
    int configIndex ) [inline], [private]
```

The Invoked function. Receive incoming data.

Parameters

<i>serialPort</i>	A member of serialPorts .
<i>catenatedSerialPort</i>	A member of config paired with a member of serialPorts .

4.1.3.11 ReceiveDatas()

```
void USB.ReceiveDatas ( ) [inline], [private]
```

Starts one instance of [InvokedFunction\(SerialPort, string\)](#) per [serialPorts](#) member.

4.1.3.12 ReceiveOptionToString()

```
string USB.ReceiveOptionToString (
    IncomingOption option ) [inline], [private]
```

Transform [ReceiveOption](#) enum into string.

Parameters

<i>option</i>	ReceiveOption member to transform.
---------------	----------------------------------------------------

Returns

A string transformed from a [ReceiveOption](#) member.

4.1.3.13 ResetArduinos()

```
void USB.ResetArduinos ( ) [inline], [private]
```

Call the reset() function on each connected Arduino.

Cannot be named Reset.

4.1.3.14 Start()

```
void USB.Start ( ) [inline], [private]
```

Pseudo-Constructor. Calls the [ConnectSerialPorts\(\)](#) function.

4.1.3.15 Teardown()

```
void USB.Teardown ( ) [inline], [private]
```

Pseudo-Destructor. Stop repeating functions, and shutdown Arduinos.

4.1.3.16 ThreadedFunction()

```
void USB.ThreadedFunction ( ) [inline], [private]
```

The threaded function. Receive incoming data.

4.1.3.17 Update()

```
void USB.Update ( ) [inline], [private]
```

The main function. Receive incoming data.

4.1.4 Member Data Documentation

4.1.4.1 debug

```
bool USB.debug = false [private]
```

Check to print debug tracers to console.

4.1.4.2 INCOMING_IN_INVOKED_FUNTION

```
const int USB.INCOMING_IN_INVOKED_FUNTION = 1 [static], [private]
```

Wait for incoming data using `InvokeRepeating(string)`.

4.1.4.3 INCOMING_IN_THREADED_FUNCTION

```
const int USB.INCOMING_IN_THREADED_FUNCTION = 2 [static], [private]
```

Wait for incoming data on a new thread.

4.1.4.4 INCOMING_IN_UPDATE_FUNCTION

```
const int USB.INCOMING_IN_UPDATE_FUNCTION = 0 [static], [private]
```

Wait for incoming data using the `Update()` event function.

4.1.4.5 INCOMING_IN_USER_INTERFACE_FUNCTION

```
const int USB.INCOMING_IN_USER_INTERFACE_FUNCTION = 3 [static], [private]
```

Wait for incoming data on the UI thread.

4.1.4.6 incomingOption

```
IncomingOption USB.incomingOption [private]
```

Choose when to receive incoming data.

4.1.4.7 INVOKED_FUNCTION

```
const string USB.INVOKED_FUNCTION = "ReceiveDatas" [static], [private]
```

For use with `InvokeRepeating(string, float, float)` and `CancellInvoke(string)` functions.

4.1.4.8 invokedFunctionDelay

```
float USB.invokedFunctionDelay = 0.1f [private]
```

Delay between [InvokedFunction\(\)](#) calls.

4.1.4.9 mutex

```
readonly Mutex USB.mutex = new Mutex() [private]
```

A mutex.

4.1.4.10 OUTGOING_RESET

```
const int USB.OUTGOING_RESET = 1 [static], [private]
```

Calls the Arduino's `reset()` function.

4.1.4.11 OUTGOING_START

```
const int USB.OUTGOING_START = 0 [static], [private]
```

Calls the Arduino's `start()` function.

4.1.4.12 SERIAL_POSTFIX_MAX_LINUX

```
const int USB.SERIAL_POSTFIX_MAX_LINUX = 256 [static], [private]
```

Maximum COM port (Linux).

4.1.4.13 SERIAL_POSTFIX_MAX_MAC

```
const int USB.SERIAL_POSTFIX_MAX_MAC = 256 [static], [private]
```

Maximum COM port (Mac).

4.1.4.14 SERIAL_POSTFIX_MAX_WINDOWS

```
const int USB.SERIAL_POSTFIX_MAX_WINDOWS = 256 [static], [private]
```

Maximum COM port (Windows).

4.1.4.15 SERIAL_POSTFIX_MIN_LINUX

```
const int USB.SERIAL_POSTFIX_MIN_LINUX = 0 [static], [private]
```

Minimum COM port (Linux).

4.1.4.16 SERIAL_POSTFIX_MIN_MAC

```
const int USB.SERIAL_POSTFIX_MIN_MAC = 0 [static], [private]
```

Minimum COM port (Mac).

4.1.4.17 SERIAL_POSTFIX_MIN_WINDOWS

```
const int USB.SERIAL_POSTFIX_MIN_WINDOWS = 0 [static], [private]
```

Minimum COM port (Windows).

4.1.4.18 SERIAL_PREFIX_LINUX

```
const string USB.SERIAL_PREFIX_LINUX = "/dev/ttyACM" [static], [private]
```

The device prefix (Linux).

4.1.4.19 SERIAL_PREFIX_MAC

```
const string USB.SERIAL_PREFIX_MAC = "" [static], [private]
```

The device prefix (Mac).

4.1.4.20 SERIAL_PREFIX_WINDOWS

```
const string USB.SERIAL_PREFIX_WINDOWS = "COM" [static], [private]
```

The device prefix (Windows).

4.1.4.21 SERIAL_TIMEOUT

```
const int USB.SERIAL_TIMEOUT = 1000 [static], [private]
```

Timeout for incoming data (milliseconds).

4.1.4.22 serialBaudRates

```
readonly int [] USB.serialBaudRates [private]
```

Initial value:

```
= {  
    9600,  
    14400,  
    19200,  
    28800,  
    31250,  
    38400,  
    57600,  
    115200,  
    300,  
    600,  
    1200,  
    2400,  
    4800,  
}
```

Baud rates (pair with Arduino).

4.1.4.23 serialPorts

```
SerialPort [] USB.serialPorts [private]
```

Successful serial port connections.

4.1.4.24 threadedFunction

```
Thread USB.threadedFunction [static]
```

A thread for [ThreadedFunction\(\)](#).

4.1.5 Property Documentation

4.1.5.1 Configs

```
string [] USB.Configs [static], [get], [private set]
```

A set of configs received from connected Arduinos.

4.1.5.2 IncomingData

```
string USB.IncomingData = "" [static], [get], [private set]
```

Incoming data from thread.

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

- [USB.cs](#)

Chapter 5

File Documentation

5.1 USB.cs File Reference

Classes

- class [USB](#)
The [USB](#) library.

Index

CatenateSerialPrefixAndPostfix
USB, [10](#)

CloseSerialPorts
USB, [10](#)

Configs
USB, [17](#)

ConnectSerialPort
USB, [10](#)

ConnectSerialPorts
USB, [11](#)

debug
USB, [14](#)

INCOMING_IN_INVOKED_FUNTION
USB, [14](#)

INCOMING_IN_THREADED_FUNCTION
USB, [14](#)

INCOMING_IN_UPDATE_FUNCTION
USB, [14](#)

INCOMING_IN_USER_INTERFACE_FUNCTION
USB, [14](#)

IncomingData
USB, [17](#)

IncomingOption
USB, [9](#)

incomingOption
USB, [14](#)

INVOKED_FUNCTION
USB, [14](#)

InvokedFunction
USB, [10](#)

invokedFunctionDelay
USB, [14](#)

InvokedFunctionStart
USB, [11](#)

InvokedFunctionStop
USB, [11](#)

mutex
USB, [15](#)

OnApplicationQuit
USB, [11](#)

OnGUI
USB, [12](#)

OUTGOING_RESET
USB, [15](#)

OUTGOING_START
USB, [15](#)

OutgoingMessages
USB, [10](#)

OutgoingMessageToString
USB, [12](#)

ReceiveData
USB, [12](#)

ReceiveDatas
USB, [12](#)

ReceiveOptionToString
USB, [12](#)

Reset
USB, [10](#)

ResetArduinos
USB, [13](#)

SERIAL_POSTFIX_MAX_LINUX
USB, [15](#)

SERIAL_POSTFIX_MAX_MAC
USB, [15](#)

SERIAL_POSTFIX_MAX_WINDOWS
USB, [15](#)

SERIAL_POSTFIX_MIN_LINUX
USB, [15](#)

SERIAL_POSTFIX_MIN_MAC
USB, [15](#)

SERIAL_POSTFIX_MIN_WINDOWS
USB, [16](#)

SERIAL_PREFIX_LINUX
USB, [16](#)

SERIAL_PREFIX_MAC
USB, [16](#)

SERIAL_PREFIX_WINDOWS
USB, [16](#)

SERIAL_TIMEOUT
USB, [16](#)

serialBaudRates
USB, [16](#)

serialPorts
USB, [16](#)

Start
USB, [10](#), [13](#)

Teardown
USB, [13](#)

ThreadedFunction
USB, [10](#), [13](#)

threadedFunction
USB, [17](#)

Update

- USB, [13](#)
- UpdateFunction
 - USB, [10](#)
- USB, [7](#)
 - CatenateSerialPrefixAndPostfix, [10](#)
 - CloseSerialPorts, [10](#)
 - Configs, [17](#)
 - ConnectSerialPort, [10](#)
 - ConnectSerialPorts, [11](#)
 - debug, [14](#)
 - INCOMING_IN_INVOKED_FUNTION, [14](#)
 - INCOMING_IN_THREADED_FUNCTION, [14](#)
 - INCOMING_IN_UPDATE_FUNCTION, [14](#)
 - INCOMING_IN_USER_INTERFACE_FUNCTION,
[14](#)
 - IncomingData, [17](#)
 - IncomingOption, [9](#)
 - incomingOption, [14](#)
 - INVOKED_FUNCTION, [14](#)
 - InvokedFunction, [10](#)
 - invokedFunctionDelay, [14](#)
 - InvokedFunctionStart, [11](#)
 - InvokedFunctionStop, [11](#)
 - mutex, [15](#)
 - OnApplicationQuit, [11](#)
 - OnGUI, [12](#)
 - OUTGOING_RESET, [15](#)
 - OUTGOING_START, [15](#)
 - OutgoingMessages, [10](#)
 - OutgoingMessageToString, [12](#)
 - ReceiveData, [12](#)
 - ReceiveDatas, [12](#)
 - ReceiveOptionToString, [12](#)
 - Reset, [10](#)
 - ResetArduinos, [13](#)
 - SERIAL_POSTFIX_MAX_LINUX, [15](#)
 - SERIAL_POSTFIX_MAX_MAC, [15](#)
 - SERIAL_POSTFIX_MAX_WINDOWS, [15](#)
 - SERIAL_POSTFIX_MIN_LINUX, [15](#)
 - SERIAL_POSTFIX_MIN_MAC, [15](#)
 - SERIAL_POSTFIX_MIN_WINDOWS, [16](#)
 - SERIAL_PREFIX_LINUX, [16](#)
 - SERIAL_PREFIX_MAC, [16](#)
 - SERIAL_PREFIX_WINDOWS, [16](#)
 - SERIAL_TIMEOUT, [16](#)
 - serialBaudRates, [16](#)
 - serialPorts, [16](#)
 - Start, [10](#), [13](#)
 - Teardown, [13](#)
 - ThreadedFunction, [10](#), [13](#)
 - threadedFunction, [17](#)
 - Update, [13](#)
 - UpdateFunction, [10](#)
 - UserInterfaceFunction, [10](#)
- USB.cs, [19](#)
- UserInterfaceFunction
 - USB, [10](#)