

# Optimus Buy 3000 Arduino

## 1.0

Generated by Doxygen 1.8.6

Tue Apr 22 2014 14:55:17



# Contents

<b>1</b>	<b>Hierarchical Index</b>	<b>1</b>
1.1	Class Hierarchy	1
<b>2</b>	<b>Data Structure Index</b>	<b>3</b>
2.1	Data Structures	3
<b>3</b>	<b>File Index</b>	<b>5</b>
3.1	File List	5
<b>4</b>	<b>Data Structure Documentation</b>	<b>7</b>
4.1	HTTPClient Class Reference	7
4.1.1	Detailed Description	7
4.2	item Struct Reference	7
4.2.1	Detailed Description	8
4.3	WiFi Class Reference	8
4.3.1	Detailed Description	9
<b>5</b>	<b>File Documentation</b>	<b>11</b>
5.1	Arduino/BTLE_SendIDToAndroid_OneReader/BTLE_SendIDToAndroid_OneReader.ino File Reference	11
5.1.1	Detailed Description	11
5.1.2	Function Documentation	11
5.1.2.1	getRFIDReaderOne	11
5.1.2.2	sendIDViaBTLE	12
5.2	Arduino/RFID_IdSpeichern_ohneKommentare_Ausgabe/RFID_IdSpeichern_ohneKommentare_Ausgabe.ino File Reference	12
5.2.1	Detailed Description	13
5.2.2	Function Documentation	13
5.2.2.1	blinkLed	13
5.2.2.2	getRFIDReaderOne	13
5.2.2.3	getRFIDReaderTwo	13
5.2.2.4	insertNewItem	13
5.2.2.5	printList	14

5.2.2.6	<a href="#">removeItem</a>	14
5.2.2.7	<a href="#">search_for</a>	14
5.2.2.8	<a href="#">sendIDToServer</a>	14
5.2.3	<a href="#">Variable Documentation</a>	14
5.2.3.1	<a href="#">list</a>	14
5.2.3.2	<a href="#">switchPin</a>	14
<b>Index</b>		<b>15</b>

# Chapter 1

## Hierarchical Index

### 1.1 Class Hierarchy

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

HTTPClient . . . . .	7
item . . . . .	7
Stream	
WiFi . . . . .	8



## Chapter 2

# Data Structure Index

### 2.1 Data Structures

Here are the data structures with brief descriptions:

<a href="#">HTTPClient</a>	7
<a href="#">item</a>	7
<a href="#">WiFiFly</a>	8





## Chapter 3

# File Index

### 3.1 File List

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

Arduino/BTLE_SendIDToAndroid_OneReader/ <a href="#">BTLE_SendIDToAndroid_OneReader.ino</a>	
Get id of current item and send it over BLE to Smartphone . . . . .	<a href="#">11</a>
Arduino/Rfid_IdSpeichern_ohneKommentare_Ausgabe/ <b>Debug.h</b> . . . . .	??
Arduino/Rfid_IdSpeichern_ohneKommentare_Ausgabe/ <b>HttpClient.cpp</b> . . . . .	??
Arduino/Rfid_IdSpeichern_ohneKommentare_Ausgabe/ <b>HttpClient.h</b> . . . . .	??
Arduino/Rfid_IdSpeichern_ohneKommentare_Ausgabe/ <a href="#">Rfid_IdSpeichern_ohneKommentare_Ausgabe.ino</a>	
Save id, send it over WiFi, access LEDs . . . . .	<a href="#">12</a>
Arduino/Rfid_IdSpeichern_ohneKommentare_Ausgabe/ <b>WiFiFly.cpp</b> . . . . .	??
Arduino/Rfid_IdSpeichern_ohneKommentare_Ausgabe/ <b>WiFiFly.h</b> . . . . .	??



## Chapter 4

# Data Structure Documentation

### 4.1 HTTPClient Class Reference

#### Public Member Functions

- int **get** (const char \*url, int timeout=HTTP\_CLIENT\_DEFAULT\_TIMEOUT)
- int **get** (const char \*url, const char \*header, int timeout=HTTP\_CLIENT\_DEFAULT\_TIMEOUT)
- int **post** (const char \*url, const char \*data, int timeout=HTTP\_CLIENT\_DEFAULT\_TIMEOUT)
- int **post** (const char \*url, const char \*headers, const char \*data, int timeout=HTTP\_CLIENT\_DEFAULT\_TIMEOUT)

#### Private Member Functions

- int **parseURL** (const char \*url, char \*host, int max\_host\_len, uint16\_t \*port, char \*path, int max\_path\_len)
- int **connect** (const char \*url, const char \*method, const char \*data, int timeout=HTTP\_CLIENT\_DEFAULT\_TIMEOUT)
- int **connect** (const char \*url, const char \*method, const char \*header, const char \*data, int timeout=HTTP\_CLIENT\_DEFAULT\_TIMEOUT)

#### Private Attributes

- [WiFly](#) \* **wifly**

#### 4.1.1 Detailed Description

Definition at line 17 of file HTTPClient.h.

### 4.2 item Struct Reference

#### Data Fields

- char **id** [11]
- unsigned long **registeredTime**
- struct [item](#) \* **next**

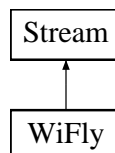
### 4.2.1 Detailed Description

Struct for a list of items that store an id (RFID-Tag) and the time of registering.

Definition at line 37 of file RFID\_IdSpeichern\_ohneKommentare\_Ausgabe.ino.

## 4.3 WiFly Class Reference

Inheritance diagram for WiFly:



### Public Member Functions

- **WiFly** (Stream \*)
- **WiFly** (Stream &)
- size\_t **write** (uint8\_t)
- size\_t **write** (const uint8\_t \*, size\_t)
- int **available** ()
- int **read** ()
- int **peek** ()
- void **flush** ()
- boolean **init** ()
- boolean **reset** ()
- boolean **save** ()
- boolean **reboot** ()
- boolean **join** (const char \*ssid)
- boolean **join** (const char \*ssid, const char \*phrase=NULL, int auth=WIFLY\_AUTH\_OPEN)
- boolean **isAssociated** ()
- boolean **isAssociated** (const char \*ssid)
- boolean **leave** ()
- boolean **connect** (const char \*host, uint16\_t port, int timeout=DEFAULT\_WAIT\_RESPONSE\_TIME)
- boolean **staticIP** (const char \*ip, const char \*mask, const char \*gateway)
- int **send** (const char \*data, int timeout=DEFAULT\_WAIT\_RESPONSE\_TIME)
- int **send** (const uint8\_t \*data, int len, int timeout=DEFAULT\_WAIT\_RESPONSE\_TIME)
- int **receive** (uint8\_t \*buf, int len, int timeout=DEFAULT\_WAIT\_RESPONSE\_TIME)
- boolean **ask** (const char \*q, const char \*a, int timeout=DEFAULT\_WAIT\_RESPONSE\_TIME)
- boolean **sendCommand** (const char \*cmd, const char \*ack=NULL, int timeout=DEFAULT\_WAIT\_RESPONSE\_TIME)
- boolean **commandMode** ()
- boolean **dataMode** ()
- void **clear** ()
- float **version** ()

### Static Public Member Functions

- static **WiFly** \* **getInstance** ()

### Private Attributes

- Stream \* **serial**
- boolean **command\_mode**
- boolean **associated**
- uint8\_t **dhcp**

### Static Private Attributes

- static [WiFly](#) \* **instance**

#### 4.3.1 Detailed Description

Definition at line 22 of file WiFly.h.



## Chapter 5

# File Documentation

### 5.1 [Arduino/BTLE\\_SendIDToAndroid\\_OneReader/BTLE\\_SendIDToAndroid\\_OneReader.ino](#) File Reference

Get id of current item and send it over BLE to Smartphone.

```
#include <SPI.h>
#include <boards.h>
#include <services.h>
#include <ble_shield.h>
```

#### Functions

- void [getRFIDReaderOne](#) ()
- void [sendIDViaBTLE](#) (char[])
- void **setup** ()
- void **loop** ()

#### Variables

- int **valReaderOne** = 0
- char **codeReaderOne** [11]
- boolean **readerOneGotID** = false
- unsigned long **timeReaderOne** = 0
- int **bytesread** = 0

#### 5.1.1 Detailed Description

Get id of current item and send it over BLE to Smartphone.

Definition in file [BTLE\\_SendIDToAndroid\\_OneReader.ino](#).

#### 5.1.2 Function Documentation

##### 5.1.2.1 void [getRFIDReaderOne](#) ( )

RFID-reader connected to Serial1 reads data and stores them in codeReaderOne[].

RFID-reader connected to Serial2 reads data and stores them in the code[]. If the data is a whole RFID-Tag, the code will be inserted in the list of items.

Definition at line 45 of file BTLE\_SendIDToAndroid\_OneReader.ino.

#### 5.1.2.2 void sendIDViaBTLE ( char id[] )

Sends an id via BT to a connected device

##### Parameters

<i>id</i>	
-----------	--

Definition at line 76 of file BTLE\_SendIDToAndroid\_OneReader.ino.

## 5.2 Arduino/RFID\_IdSpeichern\_ohneKommentare\_Ausgabe/RFID\_IdSpeichern\_ohneKommentare\_Ausgabe.ino File Reference

Save id, send it over WiFi, access LEDs.

```
#include <Arduino.h>
#include <SoftwareSerial.h>
#include "WiFly.h"
#include "HTTPClient.h"
#include <stdbool.h>
```

### Data Structures

- struct [item](#)

### Macros

- #define **RFID\_LEGACY** 0x0F
- #define **SSID** "ArduinoTest"
- #define **KEY** "password"
- #define **AUTH** WIFLY\_AUTH\_WPA2\_PSK
- #define **HTTP\_POST\_URL\_server** "http://mobile1.informatik.uni-ulm.de:80/websrv/index.php"

### Functions

- void [printList](#) (struct [item](#) \*)
- int [search\\_for](#) (char[], struct [item](#) \*)
- void [insertNewItem](#) (char[], struct [item](#) \*)
- void [getRFIDReaderOne](#) ()
- void [getRFIDReaderTwo](#) ()
- void [blinkLed](#) (int)
- void [sendIDtoServer](#) (struct [item](#) \*)
- void [setup](#) ()
- void [loop](#) ()
- void [insertNewItem](#) (char registeredCode[11], struct [item](#) \*item)
- void [removeItem](#) (char registeredCode[11], struct [item](#) \*item)
- void [sendIDtoServer](#) (struct [item](#) \*item)



- int [switchPin](#) = 11
- int **switchState**
- int **greenLed** = 8
- int **yellowLed** = 9
- int **redLed** = 10
- [WiFi](#) **wifly** (Serial1)
- [HTTPClient](#) **http**
- char **get**
- char **data** [] = "tag=fridge&action=sendallrfids&userid=4"
- char **data\_SEND\_RFID\_Markers** [200]
- boolean **markersSend** = true
- struct [item](#) \* [list](#)

### 5.2.1 Detailed Description

Save id, send it over WiFi, access LEDs.

Definition in file [RFID\\_IdSpeichern\\_ohneKommentare\\_Ausgabe.ino](#).

### 5.2.2 Function Documentation

#### 5.2.2.1 void blinkLed ( int *color* )

A LED given by color blinks for a couple of times

Parameters

<i>color</i>	
--------------	--

Definition at line 263 of file [RFID\\_IdSpeichern\\_ohneKommentare\\_Ausgabe.ino](#).

#### 5.2.2.2 void getRFIDReaderOne ( )

RFID-reader connected to Serial2 reads data and stores them in the code[]. If the data is a whole RFID-Tag, the code will be inserted in the list of items.

Definition at line 205 of file [RFID\\_IdSpeichern\\_ohneKommentare\\_Ausgabe.ino](#).

#### 5.2.2.3 void getRFIDReaderTwo ( )

Same as in [getRFIDReaderTwo\(\)](#). The RFID-reader is connected to Serial3

Definition at line 233 of file [RFID\\_IdSpeichern\\_ohneKommentare\\_Ausgabe.ino](#).

#### 5.2.2.4 void insertNewItem ( char *registeredCode*[11], struct item \* *item* )

inserts the given charsequence (registeredCode[]) in the struct item \*item at the last position

Parameters

<i>registeredCode</i> []	RFID-Tag
--------------------------	----------

<i>*item</i>	Head of the list
--------------	------------------

Definition at line 141 of file RFID\_IdSpeichern\_ohneKommentare\_Ausgabe.ino.

#### 5.2.2.5 void printList ( struct item \* item )

Prints all items in the struct item at \*item

##### Parameters

<i>*item</i>	Head of the list
--------------	------------------

Definition at line 111 of file RFID\_IdSpeichern\_ohneKommentare\_Ausgabe.ino.

#### 5.2.2.6 void removeItem ( char registeredCode[11], struct item \* item )

removes the given charsequence (registredCode[]) in the struct item \*item

##### Parameters

<i>registeredCode[]</i>	RFID-Tag
<i>*item</i>	Head of the list

Definition at line 172 of file RFID\_IdSpeichern\_ohneKommentare\_Ausgabe.ino.

#### 5.2.2.7 int search\_for ( char registeredCode[], struct item \* item )

searches for the given charsequence (registredCode[]) in the struct item \*item

##### Parameters

<i>registeredCode[]</i>	RFID-Tag
<i>*item</i>	Head of the list

Definition at line 126 of file RFID\_IdSpeichern\_ohneKommentare\_Ausgabe.ino.

#### 5.2.2.8 void sendIDToServer ( struct item \* item )

Send all the RFID-Tags to the server by extracting the id of any item found in \*item

##### Parameters

<i>*item</i>	Head of the list
--------------	------------------

Definition at line 279 of file RFID\_IdSpeichern\_ohneKommentare\_Ausgabe.ino.

### 5.2.3 Variable Documentation

#### 5.2.3.1 struct item\* list

list variable which represents the head of the list of items

Definition at line 46 of file RFID\_IdSpeichern\_ohneKommentare\_Ausgabe.ino.

#### 5.2.3.2 int switchPin = 11

If switchpin

Definition at line 21 of file RFID\_IdSpeichern\_ohneKommentare\_Ausgabe.ino.

# Index

blinkLed  
    RFID\_IdSpeichern\_ohneKommentare\_Ausgabe.-  
        ino, [13](#)

getRFIDReaderOne  
    BTLE\_SendIDToAndroid\_OneReader.ino, [11](#)  
    RFID\_IdSpeichern\_ohneKommentare\_Ausgabe.-  
        ino, [13](#)

getRFIDReaderTwo  
    RFID\_IdSpeichern\_ohneKommentare\_Ausgabe.-  
        ino, [13](#)

HTTPClient, [7](#)

insertNewItem  
    RFID\_IdSpeichern\_ohneKommentare\_Ausgabe.-  
        ino, [13](#)

item, [7](#)

list  
    RFID\_IdSpeichern\_ohneKommentare\_Ausgabe.-  
        ino, [14](#)

printList  
    RFID\_IdSpeichern\_ohneKommentare\_Ausgabe.-  
        ino, [14](#)

RFID\_IdSpeichern\_ohneKommentare\_Ausgabe.ino  
    blinkLed, [13](#)  
    getRFIDReaderOne, [13](#)  
    getRFIDReaderTwo, [13](#)  
    insertNewItem, [13](#)  
    list, [14](#)  
    printList, [14](#)  
    removeItem, [14](#)  
    search\_for, [14](#)  
    sendIDToServer, [14](#)  
    switchPin, [14](#)

removeItem  
    RFID\_IdSpeichern\_ohneKommentare\_Ausgabe.-  
        ino, [14](#)

search\_for  
    RFID\_IdSpeichern\_ohneKommentare\_Ausgabe.-  
        ino, [14](#)

sendIDToServer  
    RFID\_IdSpeichern\_ohneKommentare\_Ausgabe.-  
        ino, [14](#)

sendIDViaBTLE  
    BTLE\_SendIDToAndroid\_OneReader.ino, [12](#)

switchPin

RFID\_IdSpeichern\_ohneKommentare\_Ausgabe.-  
    ino, [14](#)

WiFly, [8](#)