

Arduino (AT)tention Device

Generated by Doxygen 1.8.11

Contents

1	Hierarchical Index	2
1.1	Class Hierarchy	2
2	Class Index	2
2.1	Class List	2
3	File Index	2
3.1	File List	2
4	Class Documentation	3
4.1	AttentionDevice Class Reference	3
4.1.1	Detailed Description	3
4.1.2	Member Function Documentation	3
4.2	SoftwareSerialAttentionDevice Class Reference	5
4.2.1	Detailed Description	6
4.2.2	Constructor & Destructor Documentation	7
4.2.3	Member Function Documentation	7
4.2.4	Member Data Documentation	12
5	File Documentation	12
5.1	AttentionDevice.cpp File Reference	12
5.2	AttentionDevice.cpp	13
5.3	AttentionDevice.h File Reference	13
5.4	AttentionDevice.h	13
5.5	SoftwareSerialAttentionDevice.cpp File Reference	14
5.6	SoftwareSerialAttentionDevice.cpp	14
5.7	SoftwareSerialAttentionDevice.h File Reference	15
5.7.1	Macro Definition Documentation	16
5.8	SoftwareSerialAttentionDevice.h	17
	Index	19

1 Hierarchical Index

1.1 Class Hierarchy

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

AttentionDevice	3
SoftwareSerialAttentionDevice	5
SoftwareSerial	
SoftwareSerialAttentionDevice	5

2 Class Index

2.1 Class List

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

AttentionDevice	
Arduino - AT device	3
SoftwareSerialAttentionDevice	5

3 File Index

3.1 File List

Here is a list of all files with brief descriptions:

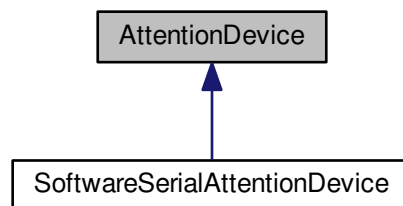
AttentionDevice.cpp	12
AttentionDevice.h	13
SoftwareSerialAttentionDevice.cpp	14
SoftwareSerialAttentionDevice.h	15

4 Class Documentation

4.1 AttentionDevice Class Reference

```
#include <AttentionDevice.h>
```

Inheritance diagram for AttentionDevice:



Public Member Functions

- virtual bool [sendCommandExpecting](#) (const char *command, const char *expectation, bool append, unsigned long timeout)=0
- virtual bool [doesResponseContains](#) (const char *expectation)=0
- virtual unsigned int [sendCommand](#) (const char *command, bool append, unsigned long timeout)=0
- virtual unsigned int [readResponse](#) (unsigned long timeout, bool append)=0
- virtual int [waitUntilReceive](#) (const char *str, unsigned int timeout)=0

4.1.1 Detailed Description

Arduino - AT device.

Author

Dalmir da Silva dalmirdasilva@gmail.com

Definition at line 10 of file [AttentionDevice.h](#).

4.1.2 Member Function Documentation

4.1.2.1 virtual bool AttentionDevice::doesResponseContains (const char * *expectation*) [pure virtual]

Checks if the last response contains the given sub-string.

Parameters

<i>expectation</i>	The expectation string.
--------------------	-------------------------

Returns

Implemented in [SoftwareSerialAttentionDevice](#).

4.1.2.2 `virtual unsigned int AttentionDevice::readResponse (unsigned long timeout, bool append)` `[pure virtual]`

Reads the response from the device.

Parameters

<i>timeout</i>	The maximum time to perform the op.
<i>append</i>	Append the response in the internal buffer.

Returns

How many bytes was received. 0 if timeout.

Implemented in [SoftwareSerialAttentionDevice](#).

4.1.2.3 `virtual unsigned int AttentionDevice::sendCommand (const char * command, bool append, unsigned long timeout)` `[pure virtual]`

Sends a command to the device.

Parameters

<i>command</i>	The command string, should be \0 ended.
<i>timeout</i>	The maximum time to perform the op.

Returns

Implemented in [SoftwareSerialAttentionDevice](#).

4.1.2.4 `virtual bool AttentionDevice::sendCommandExpecting (const char * command, const char * expectation, bool append, unsigned long timeout)` `[pure virtual]`

Sends a command expecting some result.

Parameters

<i>command</i>	The command string, should be \0 ended.
<i>expectation</i>	The expectation string.
<i>timeout</i>	The maximum time to perform the op.

Returns

Implemented in [SoftwareSerialAttentionDevice](#).

4.1.2.5 `virtual int AttentionDevice::waitUntilReceive (const char * str, unsigned int timeout)` [pure virtual]

Keeps reading the response until finds the *str* or timeout.

Parameters

<i>str</i>	String it tries to find
<i>timeout</i>	Timeout in millis

Returns

The position to the first char it finds in the internal buffer, -1 otherwise

Implemented in [SoftwareSerialAttentionDevice](#).

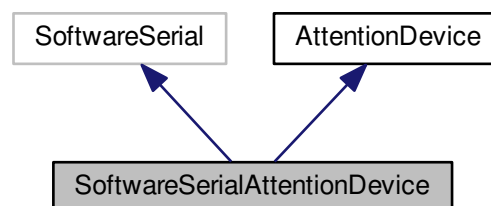
The documentation for this class was generated from the following file:

- [AttentionDevice.h](#)

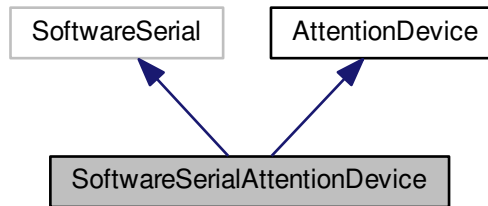
4.2 SoftwareSerialAttentionDevice Class Reference

```
#include <SoftwareSerialAttentionDevice.h>
```

Inheritance diagram for SoftwareSerialAttentionDevice:



Collaboration diagram for SoftwareSerialAttentionDevice:



Public Member Functions

- [SoftwareSerialAttentionDevice](#) (unsigned char receivePin, unsigned char transmitPin)
- unsigned char * [getLastResponse](#) ()
- bool [sendCommandExpecting](#) (const char *command, const char *expectation, bool append, unsigned long timeout)
- bool [sendCommandExpecting](#) (const char *command, const char *expectation, bool append)
- bool [sendCommandExpecting](#) (const char *command, const char *expectation, unsigned long timeout)
- bool [sendCommandExpecting](#) (const char *command, const char *expectation)
- bool [doesResponseContains](#) (const char *expectation)
- unsigned int [sendCommand](#) (const char *command, bool append, unsigned long timeout)
- unsigned int [sendCommand](#) (const char *command, bool append)
- unsigned int [sendCommand](#) (const char *command, unsigned long timeout)
- unsigned int [sendCommand](#) (const char *command)
- unsigned int [sendCommand](#) ()
- unsigned int [readResponse](#) (unsigned long timeout)
- unsigned int [readResponse](#) (unsigned long timeout, bool append)
- bool [wasResponseFullyRead](#) ()
- const char * [findInResponse](#) (const char *str)
- int [waitUntilReceive](#) (const char *str, unsigned int timeout)
- void [discardBuffer](#) ()

Private Attributes

- unsigned char [rxBuffer](#) [[AT_DEVICE_RX_BUFFER_SIZE](#)]
- unsigned int [rxBufferPos](#)
- bool [responseFullyRead](#)

4.2.1 Detailed Description

Definition at line 17 of file [SoftwareSerialAttentionDevice.h](#).

4.2.2 Constructor & Destructor Documentation

4.2.2.1 SoftwareSerialAttentionDevice::SoftwareSerialAttentionDevice (unsigned char *receivePin*, unsigned char *transmitPin*)

Definition at line 4 of file [SoftwareSerialAttentionDevice.cpp](#).

4.2.3 Member Function Documentation

4.2.3.1 void SoftwareSerialAttentionDevice::discardBuffer ()

Discards the internal buffer.

Definition at line 82 of file [SoftwareSerialAttentionDevice.cpp](#).

4.2.3.2 bool SoftwareSerialAttentionDevice::doesResponseContains (const char * *expectation*) [virtual]

Checks if the last response contains the given sub-string.

Parameters

<i>expectation</i>	The expectation string.
--------------------	-------------------------

Returns

Implements [AttentionDevice](#).

Definition at line 16 of file [SoftwareSerialAttentionDevice.cpp](#).

4.2.3.3 const char * SoftwareSerialAttentionDevice::findInResponse (const char * *str*)

Searches the response for a given string.

Parameters

<i>str</i>	The string to be searched inside response
------------	---

Returns

The pointer to the first found string, NULL if not found

Definition at line 68 of file [SoftwareSerialAttentionDevice.cpp](#).

4.2.3.4 unsigned char* SoftwareSerialAttentionDevice::getLastResponse () [inline]

Get a pointer to the last response.

Returns

Definition at line 44 of file [SoftwareSerialAttentionDevice.h](#).

4.2.3.5 `unsigned int SoftwareSerialAttentionDevice::readResponse (unsigned long timeout)` `[inline]`

Reads the response from the device.

Parameters

<i>timeout</i>	The maximum time to perform the op.
----------------	-------------------------------------

Returns

How many bytes was received. 0 if timeout.

Definition at line 155 of file [SoftwareSerialAttentionDevice.h](#).

4.2.3.6 `unsigned int SoftwareSerialAttentionDevice::readResponse (unsigned long timeout, bool append)` `[virtual]`

Reads the response from the device.

Parameters

<i>timeout</i>	The maximum time to perform the op.
<i>append</i>	Append the response in the internal buffer.

Returns

How many bytes was received. 0 if timeout.

Implements [AttentionDevice](#).

Definition at line 31 of file [SoftwareSerialAttentionDevice.cpp](#).

4.2.3.7 `unsigned int SoftwareSerialAttentionDevice::sendCommand (const char * command, bool append, unsigned long timeout)` `[virtual]`

Sends a command to the device.

Parameters

<i>command</i>	The command string, should be \0 ended.
<i>timeout</i>	The maximum time to perform the op.

Returns

Implements [AttentionDevice](#).

Definition at line 20 of file [SoftwareSerialAttentionDevice.cpp](#).

4.2.3.8 `unsigned int SoftwareSerialAttentionDevice::sendCommand (const char * command, bool append)` `[inline]`

Sends a command to the device.

Parameters

<i>command</i>	The command string, should be \0 ended.
<i>append</i>	Boolean saying if the AT must be appended.

Returns

Definition at line 117 of file [SoftwareSerialAttentionDevice.h](#).

4.2.3.9 `unsigned int SoftwareSerialAttentionDevice::sendCommand (const char * command, unsigned long timeout)`
`[inline]`

Sends a command to the device.

Parameters

<i>command</i>	The command string, should be \0 ended.
<i>append</i>	Boolean saying if the AT must be appended.

Returns

Definition at line 128 of file [SoftwareSerialAttentionDevice.h](#).

4.2.3.10 `unsigned int SoftwareSerialAttentionDevice::sendCommand (const char * command)` `[inline]`

Sends a command to the device.

Parameters

<i>command</i>	The command string, should be \0 ended.
----------------	---

Returns

Definition at line 138 of file [SoftwareSerialAttentionDevice.h](#).

4.2.3.11 `unsigned int SoftwareSerialAttentionDevice::sendCommand () [inline]`

Sends a command to the device.

Definition at line 145 of file [SoftwareSerialAttentionDevice.h](#).

4.2.3.12 `bool SoftwareSerialAttentionDevice::sendCommandExpecting (const char * command, const char * expectation, bool append, unsigned long timeout) [virtual]`

Sends a command expecting some result.

Parameters

<i>command</i>	The command string, should be \0 ended.
<i>expectation</i>	The expectation string.
<i>timeout</i>	The maximum time to perform the op.

Returns

Implements [AttentionDevice](#).

Definition at line 9 of file [SoftwareSerialAttentionDevice.cpp](#).

4.2.3.13 `bool SoftwareSerialAttentionDevice::sendCommandExpecting (const char * command, const char * expectation, bool append) [inline]`

Sends a command expecting some result.

Parameters

<i>command</i>	The command string, should be \0 ended.
<i>expectation</i>	The expectation string.
<i>append</i>	If should append AT+ in the command.

Returns

Definition at line 66 of file [SoftwareSerialAttentionDevice.h](#).

4.2.3.14 `bool SoftwareSerialAttentionDevice::sendCommandExpecting (const char * command, const char * expectation, unsigned long timeout) [inline]`

Sends a command expecting some result.

Parameters

<i>command</i>	The command string, should be \0 ended.
<i>expectation</i>	The expectation string.
<i>timeout</i>	Timeout.

Returns

Definition at line 78 of file [SoftwareSerialAttentionDevice.h](#).

4.2.3.15 `bool SoftwareSerialAttentionDevice::sendCommandExpecting (const char * command, const char * expectation)`
[inline]

Sends a command expecting some result.

Parameters

<i>command</i>	The command string, should be \0 ended.
<i>expectation</i>	The expectation string.

Returns

Definition at line 89 of file [SoftwareSerialAttentionDevice.h](#).

4.2.3.16 `int SoftwareSerialAttentionDevice::waitUntilReceive (const char * str, unsigned int timeout)` [virtual]

Keeps reading the response until finds the str or timeout.

Parameters

<i>str</i>	String it tries to find
<i>timeout</i>	Timeout in millis

Returns

The position to the first char it finds in the internal buffer, -1 otherwise

Implements [AttentionDevice](#).

Definition at line 72 of file [SoftwareSerialAttentionDevice.cpp](#).

4.2.3.17 `bool SoftwareSerialAttentionDevice::wasResponseFullyRead ()`

Checks if the rx is fully read.

If false, a new call to `readResponse` will read more data from tx placing the new data over the current data in it. Consecutive calls to `readResponse` will write data into the same buffer.

use `getLastResponse` to consume the current data and then call `readResponse` again in case tx was not fully read.

Definition at line 64 of file [SoftwareSerialAttentionDevice.cpp](#).

4.2.4 Member Data Documentation

4.2.4.1 `bool SoftwareSerialAttentionDevice::responseFullyRead` [private]

Bool indicating the last command's response was fully read by `readResponse` method.

Definition at line 33 of file [SoftwareSerialAttentionDevice.h](#).

4.2.4.2 `unsigned char SoftwareSerialAttentionDevice::rxBuffer[AT_DEVICE_RX_BUFFER_SIZE]` [private]

RX buffer.

Definition at line 22 of file [SoftwareSerialAttentionDevice.h](#).

4.2.4.3 `unsigned int SoftwareSerialAttentionDevice::rxBufferPos` [private]

RX buffer position.

Definition at line 27 of file [SoftwareSerialAttentionDevice.h](#).

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

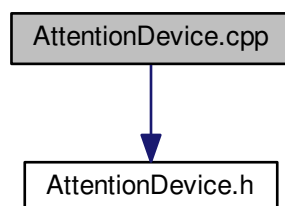
- [SoftwareSerialAttentionDevice.h](#)
- [SoftwareSerialAttentionDevice.cpp](#)

5 File Documentation

5.1 AttentionDevice.cpp File Reference

```
#include "AttentionDevice.h"
```

Include dependency graph for `AttentionDevice.cpp`:

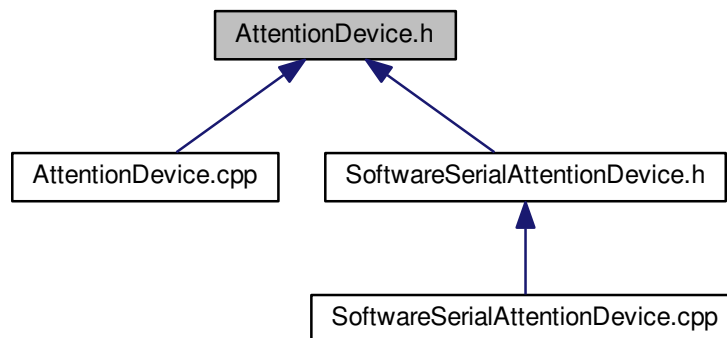


5.2 AttentionDevice.cpp

```
00001 #include "AttentionDevice.h"
```

5.3 AttentionDevice.h File Reference

This graph shows which files directly or indirectly include this file:



Classes

- class [AttentionDevice](#)

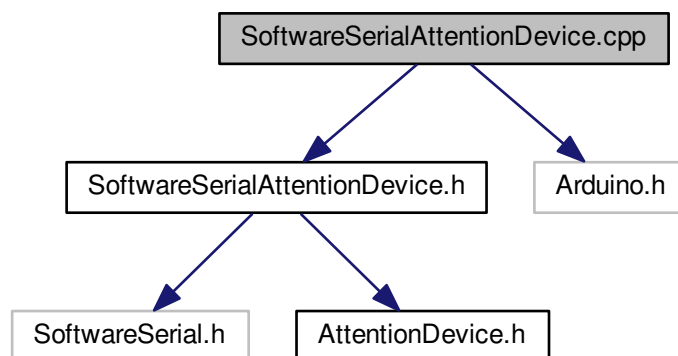
5.4 AttentionDevice.h

```

00001
00007 #ifndef __ARDUINO_ATTENTION_DEVICE_H__
00008 #define __ARDUINO_ATTENTION_DEVICE_H__ 1
00009
00010 class AttentionDevice {
00011
00012 public:
00013
00022     virtual bool sendCommandExpecting(const char *command, const char *expectation,
    bool append, unsigned long timeout) = 0;
00023
00030     virtual bool doesResponseContains(const char *expectation) = 0;
00031
00039     virtual unsigned int sendCommand(const char *command, bool append, unsigned long timeout) =
    0;
00040
00048     virtual unsigned int readResponse(unsigned long timeout, bool append) = 0;
00049
00057     virtual int waitUntilReceive(const char *str, unsigned int timeout) = 0;
00058 };
00059
00060 #endif /* __ARDUINO_ATTENTION_DEVICE_H__ */
  
```

5.5 SoftwareSerialAttentionDevice.cpp File Reference

```
#include "SoftwareSerialAttentionDevice.h"
#include <Arduino.h>
Include dependency graph for SoftwareSerialAttentionDevice.cpp:
```



5.6 SoftwareSerialAttentionDevice.cpp

```

00001 #include "SoftwareSerialAttentionDevice.h"
00002 #include <Arduino.h>
00003
00004 SoftwareSerialAttentionDevice::SoftwareSerialAttentionDevice
00005     (unsigned char receivePin, unsigned char transmitPin)
00006     : SoftwareSerial(receivePin, transmitPin), rxBufferPos(0), responseFullyRead(true) {
00007     rxBuffer[0] = '\0';
00008 }
00009 bool SoftwareSerialAttentionDevice::sendCommandExpecting
00010     (const char *command, const char *expectation, bool append, unsigned long timeout) {
00011     if (sendCommand(command, append, timeout) == 0) {
00012         return false;
00013     }
00014     return doesResponseContains(expectation);
00015 }
00016 bool SoftwareSerialAttentionDevice::doesResponseContains
00017     (const char *expectation) {
00018     return findInResponse(expectation) != NULL;
00019 }
00020 unsigned int SoftwareSerialAttentionDevice::sendCommand(const
00021     char *command, bool append, unsigned long timeout) {
00022     rxBufferPos = 0;
00023     rxBuffer[0] = '\0';
00024     flush();
00025     if (append) {
00026         print("AT");
00027     }
00028     println(command);
00029     return readResponse(timeout);
00030 }
00031 unsigned int SoftwareSerialAttentionDevice::readResponse(
00032     unsigned long timeout, bool append) {
00033     int availableBytes;
00034     unsigned long pos, last, now, start;
00035     unsigned int read;
00036     last = start = millis();
00037     if (!append) {
00038         rxBufferPos = 0;

```

```

00038     }
00039     pos = rxBufferPos;
00040     responseFullyRead = true;
00041     while (available() <= 0 && (millis() - start) < timeout)
00042     ;
00043     start = millis();
00044     do {
00045         availableBytes = available();
00046         if (availableBytes > 0) {
00047             if (rxBufferPos + availableBytes >=
00048                 AT_DEVICE_RX_BUFFER_SIZE) {
00049                 availableBytes = AT_DEVICE_RX_BUFFER_SIZE - (
00049                     rxBufferPos + 1);
00050                 responseFullyRead = false;
00051             }
00052             if (availableBytes > 0) {
00053                 last = millis();
00053                 // we have the guaranty that is not going to be too big because it is constrained by the
00054                 buffer size.
00054                 read = readBytes((char *) &rxBuffer[rxBufferPos], availableBytes);
00055                 rxBufferPos += read;
00056                 rxBuffer[rxBufferPos] = '\0';
00057             }
00058         }
00059         now = millis();
00060         } while ((availableBytes > 0 || (now - last) <
00060             AT_DEVICE_SERIAL_INTERBYTE_TIMEOUT) && (now - start) < timeout &&
00061             responseFullyRead);
00061         return rxBufferPos - pos;
00062     }
00063
00064     bool SoftwareSerialAttentionDevice::wasResponseFullyRead
00065     () {
00065         return responseFullyRead;
00066     }
00067
00068     const char *SoftwareSerialAttentionDevice::findInResponse(
00068         const char *str) {
00069         return strstr((const char *) &rxBuffer[0], str);
00070     }
00071
00072     int SoftwareSerialAttentionDevice::waitUntilReceive(const
00072         char *str, unsigned int timeout) {
00073         const char *pos;
00074         while ((pos = findInResponse(str)) == NULL && readResponse(timeout,
00074             responseFullyRead) > 0)
00075             ;
00076         if (pos != NULL) {
00077             return (int) (pos - (const char *) &rxBuffer[0]);
00078         }
00079         return -1;
00080     }
00081
00082     void SoftwareSerialAttentionDevice::discardBuffer() {
00083         rxBuffer[0] = '\0';
00084     }

```

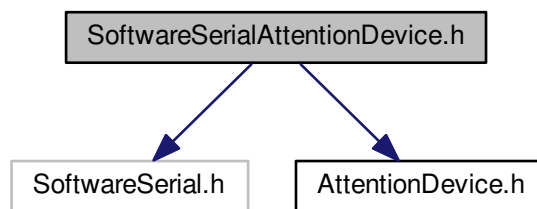
5.7 SoftwareSerialAttentionDevice.h File Reference

```

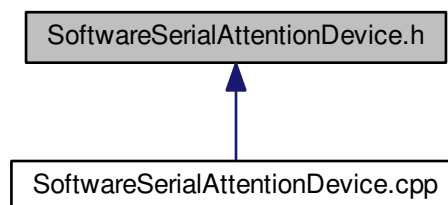
#include <SoftwareSerial.h>
#include <AttentionDevice.h>

```


Include dependency graph for SoftwareSerialAttentionDevice.h:



This graph shows which files directly or indirectly include this file:



Classes

- class [SoftwareSerialAttentionDevice](#)

Macros

- `#define AT_DEVICE_RX_BUFFER_SIZE 256`
- `#define AT_DEVICE_DEFAULT_COMMAND_RESPONSE_TIMEOUT 1000UL`
- `#define AT_DEVICE_SERIAL_INTERBYTE_TIMEOUT 50UL`

5.7.1 Macro Definition Documentation

5.7.1.1 `#define AT_DEVICE_DEFAULT_COMMAND_RESPONSE_TIMEOUT 1000UL`

Definition at line 14 of file [SoftwareSerialAttentionDevice.h](#).

5.7.1.2 #define AT_DEVICE_RX_BUFFER_SIZE 256

Arduino - AT device.

Author

Dalmir da Silva dalmirdasilva@gmail.com

Definition at line 13 of file [SoftwareSerialAttentionDevice.h](#).

5.7.1.3 #define AT_DEVICE_SERIAL_INTERBYTE_TIMEOUT 50UL

Definition at line 15 of file [SoftwareSerialAttentionDevice.h](#).

5.8 SoftwareSerialAttentionDevice.h

```

00001
00007 #ifndef __ARDUINO_SOFTWARE_SERIAL_ATTENTION_DEVICE_H__
00008 #define __ARDUINO_SOFTWARE_SERIAL_ATTENTION_DEVICE_H__ 1
00009
00010 #include <SoftwareSerial.h>
00011 #include <AttentionDevice.h>
00012
00013 #define AT_DEVICE_RX_BUFFER_SIZE 256
00014 #define AT_DEVICE_DEFAULT_COMMAND_RESPONSE_TIMEOUT 1000UL
00015 #define AT_DEVICE_SERIAL_INTERBYTE_TIMEOUT 50UL
00016
00017 class SoftwareSerialAttentionDevice : public SoftwareSerial, public
AttentionDevice {
00018
00022     unsigned char rxBuffer[AT_DEVICE_RX_BUFFER_SIZE];
00023
00027     unsigned int rxBufferPos;
00028
00033     bool responseFullyRead;
00034
00035 public:
00036
00037     SoftwareSerialAttentionDevice(unsigned char receivePin, unsigned char
transmitPin);
00038
00044     unsigned char *getLastResponse() {
00045         return &rxBuffer[0];
00046     }
00047
00056     bool sendCommandExpecting(const char *command, const char *expectation, bool append
, unsigned long timeout);
00057
00066     inline bool sendCommandExpecting(const char *command, const char *expectation, bool
append) {
00067         return sendCommandExpecting(command, expectation, append,
AT_DEVICE_DEFAULT_COMMAND_RESPONSE_TIMEOUT);
00068     }
00069
00078     inline bool sendCommandExpecting(const char *command, const char *expectation,
unsigned long timeout) {
00079         return sendCommandExpecting(command, expectation, false, timeout);
00080     }
00081
00089     inline bool sendCommandExpecting(const char *command, const char *expectation) {
00090         return sendCommandExpecting(command, expectation, (bool) false);
00091     }
00092
00099     bool doesResponseContains(const char *expectation);
00100
00108     unsigned int sendCommand(const char *command, bool append, unsigned long timeout);
00109
00117     inline unsigned int sendCommand(const char *command, bool append) {
00118         return sendCommand(command, append, 1000);
00119     }
00120
00128     inline unsigned int sendCommand(const char *command, unsigned long timeout) {
00129         return sendCommand(command, (bool) false, timeout);
00130     }

```

```
00131
00138 inline unsigned int sendCommand(const char *command) {
00139     return sendCommand(command, (bool) false);
00140 }
00141
00145 inline unsigned int sendCommand() {
00146     return sendCommand("");
00147 }
00148
00155 inline unsigned int readResponse(unsigned long timeout) {
00156     return readResponse(timeout, false);
00157 }
00158
00166 unsigned int readResponse(unsigned long timeout, bool append);
00167
00178 bool wasResponseFullyRead();
00179
00186 const char *findInResponse(const char *str);
00187
00195 int waitUntilReceive(const char *str, unsigned int timeout);
00196
00200 void discardBuffer();
00201 };
00202
00203 #endif /* __ARDUINO_SOFTWARE_SERIAL_ATTENTION_DEVICE_H__ */
```

Index

AT_DEVICE_DEFAULT_COMMAND_RESPONSE_TIMEOUT
 SoftwareSerialAttentionDevice.h, 16

AT_DEVICE_RX_BUFFER_SIZE
 SoftwareSerialAttentionDevice.h, 16

AT_DEVICE_SERIAL_INTERBYTE_TIMEOUT
 SoftwareSerialAttentionDevice.h, 17

AttentionDevice, 3
 doesResponseContains, 3
 readResponse, 4
 sendCommand, 4
 sendCommandExpecting, 4
 waitUntilReceive, 5

AttentionDevice.cpp, 12, 13

AttentionDevice.h, 13

discardBuffer
 SoftwareSerialAttentionDevice, 7

doesResponseContains
 AttentionDevice, 3
 SoftwareSerialAttentionDevice, 7

findInResponse
 SoftwareSerialAttentionDevice, 7

getLastResponse
 SoftwareSerialAttentionDevice, 7

readResponse
 AttentionDevice, 4
 SoftwareSerialAttentionDevice, 8

responseFullyRead
 SoftwareSerialAttentionDevice, 12

rxBuffer
 SoftwareSerialAttentionDevice, 12

rxBufferPos
 SoftwareSerialAttentionDevice, 12

sendCommand
 AttentionDevice, 4
 SoftwareSerialAttentionDevice, 8–10

sendCommandExpecting
 AttentionDevice, 4
 SoftwareSerialAttentionDevice, 10, 11

SoftwareSerialAttentionDevice, 5
 discardBuffer, 7
 doesResponseContains, 7
 findInResponse, 7
 getLastResponse, 7
 readResponse, 8
 responseFullyRead, 12
 rxBuffer, 12
 rxBufferPos, 12
 sendCommand, 8–10
 sendCommandExpecting, 10, 11
 SoftwareSerialAttentionDevice, 7
 waitUntilReceive, 11
 wasResponseFullyRead, 11

SoftwareSerialAttentionDevice.cpp, 14

SoftwareSerialAttentionDevice.h, 15, 17
 AT_DEVICE_DEFAULT_COMMAND_RESPONSE_TIMEOUT, 16
 AT_DEVICE_RX_BUFFER_SIZE, 16
 AT_DEVICE_SERIAL_INTERBYTE_TIMEOUT, 17

waitUntilReceive
 AttentionDevice, 5
 SoftwareSerialAttentionDevice, 11

wasResponseFullyRead
 SoftwareSerialAttentionDevice, 11