

# Synoptic Nodes

v1.0

Generated by Doxygen 1.9.1



<b>1 File Index</b>	<b>1</b>
1.1 File List	1
<b>2 File Documentation</b>	<b>3</b>
2.1 bridge_node.ino File Reference	3
2.1.1 Detailed Description	4
2.1.2 Macro Definition Documentation	4
2.1.2.1 HOSTNAME	4
2.1.2.2 MESH_PASSWORD	4
2.1.2.3 MESH_PORT	4
2.1.2.4 MESH_PREFIX	5
2.1.2.5 STATION_PASSWORD	5
2.1.2.6 STATION_SSID	5
2.1.3 Function Documentation	5
2.1.3.1 changedConnectionCallback()	5
2.1.3.2 getlocalIP()	5
2.1.3.3 loop()	5
2.1.3.4 myAPIP()	5
2.1.3.5 myIP()	6
2.1.3.6 newConnectionCallback()	6
2.1.3.7 server()	6
2.1.3.8 setup()	6
2.1.4 Variable Documentation	6
2.1.4.1 client_connections	6
2.1.4.2 client_password	6
2.1.4.3 client_requests	7
2.1.4.4 client_username	7
2.1.4.5 current_ticket	7
2.1.4.6 mesh	7
2.1.4.7 nodeName	7
2.2 logging_node.ino File Reference	7
2.2.1 Detailed Description	8
2.2.2 Macro Definition Documentation	9
2.2.2.1 MESH_PASSWORD	9
2.2.2.2 MESH_PORT	9
2.2.2.3 MESH_PREFIX	9
2.2.3 Function Documentation	9
2.2.3.1 appendFile()	9
2.2.3.2 logReading()	10
2.2.3.3 loop()	10
2.2.3.4 sendMessage()	10
2.2.3.5 sendReadings()	10

2.2.3.6 sendReadingsArchive()	11
2.2.3.7 setup()	11
2.2.3.8 writeFile()	11
2.2.4 Variable Documentation	12
2.2.4.1 chip_select	12
2.2.4.2 errors_archive_path	12
2.2.4.3 errors_path	12
2.2.4.4 mesh	12
2.2.4.5 minutes_interval	12
2.2.4.6 nodeName	12
2.2.4.7 prev_time	12
2.2.4.8 readings_archive_path	13
2.2.4.9 readings_path	13
2.2.4.10 settings_archive_path	13
2.2.4.11 settings_path	13
2.2.4.12 userScheduler	13
<b>Index</b>	<b>15</b>

# Chapter 1

## File Index

### 1.1 File List

Here is a list of all files with brief descriptions:

<a href="#">bridge_node.ino</a>	Sketch file for the bridge node of the mesh network using HTTP requests with a ticket system [see <a href="#">readings_node.ino</a> ] for . . . . .	3
<a href="#">logging_node.ino</a>	Sketch file for children nodes that record and send readings to the user via the bridge [see <a href="#">bridge_node.ino</a> ] . . . . .	7



## Chapter 2

# File Documentation

### 2.1 bridge\_node.ino File Reference

Sketch file for the bridge node of the mesh network using HTTP requests with a ticket system [see readings\_↔ node.ino] for.

```
#include "IPAddress.h"
#include "painlessMesh.h"
#include "namedMesh.h"
#include <map>
#include <ArduinoJson.h>
#include <TimeLib.h>
#include <AsyncTCP.h>
#include <ESPAsyncWebServer.h>
```

#### Macros

- #define MESH\_PREFIX "whateverYouLike"
- #define MESH\_PASSWORD "somethingSneaky"
- #define MESH\_PORT 5555
- #define STATION\_SSID "----"
- #define STATION\_PASSWORD "----"
- #define HOSTNAME "HTTP\_BRIDGE"

#### Functions

- IPAddress [getlocalIP](#) ()
- AsyncWebServer [server](#) (80)
- IPAddress [myIP](#) (0, 0, 0, 0)
- IPAddress [myAPIP](#) (0, 0, 0, 0)
- void [setup](#) ()
- void [loop](#) ()
- void [newConnectionCallback](#) (uint32\_t nodeId)
- void [changedConnectionCallback](#) ()

## Variables

- `std::map< unsigned char, String >` `client_requests`
- `namedMesh` `mesh`
- `String` `nodeName` = "root"
- `unsigned char` `current_ticket` = 0
- `std::map< IPAddress, String >` `client_connections`
- `const char *` `client_username` = "admin"
- `const char *` `client_password` = "admin"

### 2.1.1 Detailed Description

Sketch file for the bridge node of the mesh network using HTTP requests with a ticket system [see `readings_↔ node.ino`] for.

#### Author

T. Buckingham

The file contains ::

- Functions for querying nodes and handling user requests
  - Testing html to be used without a client program

FOR MORE DETAILS ON LIBRARIES OR EXAMPLES USED PLEASE SEE THE README

### 2.1.2 Macro Definition Documentation

#### 2.1.2.1 HOSTNAME

```
#define HOSTNAME "HTTP_BRIDGE"
```

#### 2.1.2.2 MESH\_PASSWORD

```
#define MESH_PASSWORD "somethingSneaky"
```

#### 2.1.2.3 MESH\_PORT

```
#define MESH_PORT 5555
```



#### 2.1.2.4 MESH\_PREFIX

```
#define MESH_PREFIX "whateverYouLike"
```

#### 2.1.2.5 STATION\_PASSWORD

```
#define STATION_PASSWORD "----"
```

#### 2.1.2.6 STATION\_SSID

```
#define STATION_SSID "----"
```

### 2.1.3 Function Documentation

#### 2.1.3.1 changedConnectionCallback()

```
void changedConnectionCallback ( )
```

#### 2.1.3.2 getlocalIP()

```
IPAddress getlocalIP ( )
```

#### 2.1.3.3 loop()

```
void loop ( )
```

#### 2.1.3.4 myAPIP()

```
IPAddress myAPIP (
    0 ,
    0 ,
    0 ,
    0 )
```

#### 2.1.3.5 myIP()

```
IPAddress myIP (
    0 ,
    0 ,
    0 ,
    0 )
```

#### 2.1.3.6 newConnectionCallback()

```
void newConnectionCallback (
    uint32_t nodeId )
```

#### 2.1.3.7 server()

```
AsyncWebServer server (
    80 )
```

#### 2.1.3.8 setup()

```
void setup ( )
```

### 2.1.4 Variable Documentation

#### 2.1.4.1 client\_connections

```
std::map<IPAddress, String> client_connections
```

#### 2.1.4.2 client\_password

```
const char* client_password = "admin"
```

### 2.1.4.3 client\_requests

```
std::map<unsigned char, String> client_requests
```

### 2.1.4.4 client\_username

```
const char* client_username = "admin"
```

### 2.1.4.5 current\_ticket

```
unsigned char current_ticket = 0
```

### 2.1.4.6 mesh

```
namedMesh mesh
```

### 2.1.4.7 nodeName

```
String nodeName = "root"
```

## 2.2 logging\_node.ino File Reference

Sketch file for children nodes that record and send readings to the user via the bridge [see [bridge\\_node.ino](#)].

```
#include "painlessMesh.h"  
#include "namedMesh.h"  
#include <SD.h>  
#include <SPI.h>  
#include <TimeLib.h>  
#include <FS.h>
```

### Macros

- #define [MESH\\_PREFIX](#) "whateverYouLike"
- #define [MESH\\_PASSWORD](#) "somethingSneaky"
- #define [MESH\\_PORT](#) 5555

## Functions

- void [sendReadings](#) (unsigned char ticket\_number)  
*Function used to send stored readings as a string to the bridge node when the user requests.*
- void [sendReadingsArchive](#) (unsigned char ticket\_number)  
*Function used to send stored readings as a string to the bridge node when the user requests then stores those readings in an archive file.*
- void [sendMessage](#) ()  
*Basic function used to test the connections between the root during development.*
- void [appendFile](#) (String path, String content)  
*Appends a string, in this case a log, to the specified file.*
- void [writeFile](#) (String path, String content)  
*Writes a new file onto the storage device.*
- void [logReading](#) (String path)  
*Records a reading to the readings files.*
- void [setup](#) ()
- void [loop](#) ()

## Variables

- Scheduler [userScheduler](#)
- const int [chip\\_select](#) = D8
- namedMesh [mesh](#)
- String [nodeName](#) = "lobitos"
- String [readings\\_path](#) = "/readings.txt"
- String [errors\\_path](#) = "/errors.txt"
- String [settings\\_path](#) = "/settings.txt"
- String [readings\\_archive\\_path](#) = "/readings\_archive.txt"
- String [errors\\_archive\\_path](#) = "/errors.txt"
- String [settings\\_archive\\_path](#) = "/settings.txt"
- unsigned char [minutes\\_interval](#) = 1
- time\_t [prev\\_time](#)

### 2.2.1 Detailed Description

Sketch file for children nodes that record and send readings to the user via the bridge [see [bridge\\_node.ino](#)].

#### Author

T. Buckingham

The file contains ::

- Functions for handling requests and responding to the bridge
  - Node specific details such as its string name

FOR MORE DETAILS ON LIBRARIES OR EXAMPLES USED PLEASE SEE THE README

## 2.2.2 Macro Definition Documentation

### 2.2.2.1 MESH\_PASSWORD

```
#define MESH_PASSWORD "somethingSneaky"
```

### 2.2.2.2 MESH\_PORT

```
#define MESH_PORT 5555
```

### 2.2.2.3 MESH\_PREFIX

```
#define MESH_PREFIX "whateverYouLike"
```

## 2.2.3 Function Documentation

### 2.2.3.1 appendFile()

```
void appendFile (
    String path,
    String content )
```

Appends a string, in this case a log, to the specified file.

#### Parameters

<i>fs</i>	a file system pointer used in interfacing with the file
<i>path</i>	the files path on the storage medium
<i>content</i>	the string to be appended to the file

#### Returns

Void.

### 2.2.3.2 logReading()

```
void logReading (
    String path )
```

Records a reading to the readings files.

#### Parameters

<i>path</i>	the files path on the storage medium
-------------	--------------------------------------

#### Returns

Void.

### 2.2.3.3 loop()

```
void loop ( )
```

### 2.2.3.4 sendMessage()

```
void sendMessage ( )
```

Basic function used to test the connections between the root during development.

#### Parameters

<i>Void.</i>	
--------------	--

#### Returns

Void.

### 2.2.3.5 sendReadings()

```
void sendReadings (
    unsigned char ticket_number )
```

Function used to send stored readings as a string to the bridge node when the user requests.

**Parameters**

<i>ticket_number</i>	the ticket that is currently being sent back to the bridge/root node
----------------------	--

**Returns**

Void.

**2.2.3.6 sendReadingsArchive()**

```
void sendReadingsArchive (
    unsigned char ticket_number )
```

Function used to send stored readings as a string to the bridge node when the user requests then stores those readings in an archive file.

**Parameters**

--	--

**2.2.3.7 setup()**

```
void setup ( )
```

**2.2.3.8 writeFile()**

```
void writeFile (
    String path,
    String content )
```

Writes a new file onto the storage device.

**Parameters**

<i>content</i>	the content to be saved to the file
<i>path</i>	the files path on the storage medium

**Returns**

Void.

## 2.2.4 Variable Documentation

### 2.2.4.1 chip\_select

```
const int chip_select = D8
```

### 2.2.4.2 errors\_archive\_path

```
String errors_archive_path = "/errors.txt"
```

### 2.2.4.3 errors\_path

```
String errors_path = "/errors.txt"
```

### 2.2.4.4 mesh

```
namedMesh mesh
```

### 2.2.4.5 minutes\_interval

```
unsigned char minutes_interval = 1
```

### 2.2.4.6 nodeName

```
String nodeName = "lobitos"
```

### 2.2.4.7 prev\_time

```
time_t prev_time
```



#### 2.2.4.8 readings\_archive\_path

```
String readings_archive_path = "/readings_archive.txt"
```

#### 2.2.4.9 readings\_path

```
String readings_path = "/readings.txt"
```

#### 2.2.4.10 settings\_archive\_path

```
String settings_archive_path = "/settings.txt"
```

#### 2.2.4.11 settings\_path

```
String settings_path = "/settings.txt"
```

#### 2.2.4.12 userScheduler

```
Scheduler userScheduler
```



# Index

- appendFile
  - logging\_node.ino, 9
- bridge\_node.ino, 3
  - changedConnectionCallback, 5
  - client\_connections, 6
  - client\_password, 6
  - client\_requests, 6
  - client\_username, 7
  - current\_ticket, 7
  - getlocalIP, 5
  - HOSTNAME, 4
  - loop, 5
  - mesh, 7
  - MESH\_PASSWORD, 4
  - MESH\_PORT, 4
  - MESH\_PREFIX, 4
  - myAPIP, 5
  - myIP, 5
  - newConnectionCallback, 6
  - nodeName, 7
  - server, 6
  - setup, 6
  - STATION\_PASSWORD, 5
  - STATION\_SSID, 5
- changedConnectionCallback
  - bridge\_node.ino, 5
- chip\_select
  - logging\_node.ino, 12
- client\_connections
  - bridge\_node.ino, 6
- client\_password
  - bridge\_node.ino, 6
- client\_requests
  - bridge\_node.ino, 6
- client\_username
  - bridge\_node.ino, 7
- current\_ticket
  - bridge\_node.ino, 7
- errors\_archive\_path
  - logging\_node.ino, 12
- errors\_path
  - logging\_node.ino, 12
- getlocalIP
  - bridge\_node.ino, 5
- HOSTNAME
  - bridge\_node.ino, 4
- logging\_node.ino, 7
  - appendFile, 9
  - chip\_select, 12
  - errors\_archive\_path, 12
  - errors\_path, 12
  - logReading, 9
  - loop, 10
  - mesh, 12
  - MESH\_PASSWORD, 9
  - MESH\_PORT, 9
  - MESH\_PREFIX, 9
  - minutes\_interval, 12
  - nodeName, 12
  - prev\_time, 12
  - readings\_archive\_path, 12
  - readings\_path, 13
  - sendMessage, 10
  - sendReadings, 10
  - sendReadingsArchive, 11
  - settings\_archive\_path, 13
  - settings\_path, 13
  - setup, 11
  - userScheduler, 13
  - writeFile, 11
- logReading
  - logging\_node.ino, 9
- loop
  - bridge\_node.ino, 5
  - logging\_node.ino, 10
- mesh
  - bridge\_node.ino, 7
  - logging\_node.ino, 12
- MESH\_PASSWORD
  - bridge\_node.ino, 4
  - logging\_node.ino, 9
- MESH\_PORT
  - bridge\_node.ino, 4
  - logging\_node.ino, 9
- MESH\_PREFIX
  - bridge\_node.ino, 4
  - logging\_node.ino, 9
- minutes\_interval
  - logging\_node.ino, 12
- myAPIP
  - bridge\_node.ino, 5
- myIP
  - bridge\_node.ino, 5
- newConnectionCallback

- bridge\_node.ino, [6](#)
- nodeName
  - bridge\_node.ino, [7](#)
  - logging\_node.ino, [12](#)
- prev\_time
  - logging\_node.ino, [12](#)
- readings\_archive\_path
  - logging\_node.ino, [12](#)
- readings\_path
  - logging\_node.ino, [13](#)
- sendMessage
  - logging\_node.ino, [10](#)
- sendReadings
  - logging\_node.ino, [10](#)
- sendReadingsArchive
  - logging\_node.ino, [11](#)
- server
  - bridge\_node.ino, [6](#)
- settings\_archive\_path
  - logging\_node.ino, [13](#)
- settings\_path
  - logging\_node.ino, [13](#)
- setup
  - bridge\_node.ino, [6](#)
  - logging\_node.ino, [11](#)
- STATION\_PASSWORD
  - bridge\_node.ino, [5](#)
- STATION\_SSID
  - bridge\_node.ino, [5](#)
- userScheduler
  - logging\_node.ino, [13](#)
- writeFile
  - logging\_node.ino, [11](#)