Synoptic Nodes

v1.0

Generated by Doxygen 1.9.1

Chapter 1

File Index

1.1 File List

Here is a list of all files with brief descriptions:

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

fs	a file system pointer used in interfacing with the file
path	the files path on the storage medium
content	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

path	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

Void.	

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

ticket_number	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

content	the content to be saved to the file
path	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