## sub1GHz\_RX\_Layer Reference Manual

Generated by Doxygen 1.8.16

Fri Nov 8 2019 13:49:18

| 1 sub1GHz_RX_Layer                           | 1  |
|--|----|
| 2 Hierarchical Index                         | 3  |
| 2.1 Class Hierarchy                          | 3  |
| 3 Data Structure Index                       | 5  |
| 3.1 Data Structures                          | 5  |
| 4 File Index                                 | 7  |
| 4.1 File List                                | 7  |
| 5 Data Structure Documentation               | 9  |
| 5.1 EasyLinkLayer Class Reference            | 9  |
| 5.1.1 Detailed Description                   | 10 |
| 5.1.2 Constructor & Destructor Documentation | 10 |
| 5.1.2.1 EasyLinkLayer()                      | 10 |
| 5.1.3 Member Function Documentation          | 11 |
| 5.1.3.1 begin()                              | 11 |
| 5.1.3.2 getAddressDestinationRX()            | 11 |
| 5.1.3.3 getAddressFilter()                   | 12 |
| 5.1.3.4 getAddressLocal()                    | 12 |
| 5.1.3.5 getRSSIRX()                          | 12 |
| 5.1.3.6 receive()                            | 13 |
| 5.1.3.7 setAddressDestinationTX()            | 13 |
| 5.1.3.8 setAddressFilter()                   | 14 |
| 5.1.3.9 transmit()                           | 14 |
| 6 File Documentation                         | 17 |
| 6.1 EasyLinkLayer.h File Reference           | 17 |
| 6.1.1 Detailed Description                   | 18 |
| 6.1.2 Typedef Documentation                  | 19 |
| 6.1.2.1 AddressFilter_t                      | 19 |
| 6.1.2.2 AddressIEEE_t                        | 19 |
| 6.2 rtosGlobals.h File Reference             | 19 |
| 6.2.1 Detailed Description                   | 20 |
| 6.3 sub1GHz_RX_Layer.ino File Reference      | 20 |
| 6.3.1 Detailed Description                   | 21 |
| 6.3.2 Macro Definition Documentation         | 22 |
| 6.3.2.1 WITH_ADDRESS_FILTERING               | 22 |
| 6.3.3 Function Documentation                 | 22 |
| 6.3.3.1 printAddress()                       | 22 |
| 6.3.3.2 printFilter()                        | 22 |
| Index  | 23 |

# sub1GHz\_RX\_Layer

```
Additional layer for EasyLink

Developed with embedXcode+

Author

Rei Vilo

https://embeddedcomputing.weebly.com

Date

05 Nov 2019 10:56

Version

103

Copyright

(c) Rei Vilo, 2019

CC = BY SA NC

See also
```

ReadMe.txt for references

# **Hierarchical Index**

## 2.1 Class Hierarchy

| This inheritance list is sorted roughly, but not completely, alphabetically: |   |
|--|---|
| EasyLink<br>EasyLinkLayer  | 9 |

**Hierarchical Index** 

# **Data Structure Index**

## 3.1 Data Structures

| Here are the data structures with brief descri | riptions |
|--|----------|
|--|----------|

| EasyLinkLayer |  |  |  |      |  |  |  |      |  |      |  |  |  |  |  |  |  |  |   |
|---------------|--|--|--|------|--|--|--|------|--|------|--|--|--|--|--|--|--|--|---|
| EasyLinkLayer |  |  |  | <br> |  |  |  | <br> |  | <br> |  |  |  |  |  |  |  |  | 9 |

6 **Data Structure Index** 

# File Index

## 4.1 File List

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

| EasyLinkLayer.h                         |    |
|---|----|
| Library header                          | 17 |
| rtosGlobals.h                           |    |
| Global variables for Energia MT project | 19 |
| sub1GHz_RX_Layer.ino                    |    |
| Main sketch                             | 20 |

8 File Index

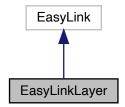
## **Data Structure Documentation**

## 5.1 EasyLinkLayer Class Reference

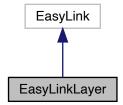
## EasyLinkLayer.

#include <EasyLinkLayer.h>

Inheritance diagram for EasyLinkLayer:



Collaboration diagram for EasyLinkLayer:



#### **Public Member Functions**

EasyLinkLayer (bool flagAddressFiltering=false)

Constructor.

• EasyLink\_Status begin ()

Initialise and start the radio.

• EasyLink\_Status transmit (const void \*payload, size\_t size)

Send a message.

• EasyLink\_Status receive (void \*payload, size\_t size, uint32\_t ms=2000)

Receive a message.

• EasyLink\_Status setAddressFilter (uint8\_t slot, AddressIEEE\_t address)

Set address to addresses filter.

EasyLink\_Status getAddressFilter (uint8\_t slot, AddressIEEE\_t \*address)

Get address to addresses filter.

• EasyLink\_Status getAddressLocal (AddressIEEE\_t \*ieeeAddress)

Get local address.

void setAddressDestinationTX (AddressIEEE\_t ieeeAddress)

Set destination for TX message.

void getAddressDestinationRX (AddressIEEE t \*ieeeAddress)

Get destination from RX message.

uint8\_t getRSSIRX ()

Get RSSI from RX message.

### 5.1.1 Detailed Description

## EasyLinkLayer.

Additional layer with address filtering for EasyLink

Note

Basic usage: begin(), transmit() and receive()

## 5.1.2 Constructor & Destructor Documentation

#### 5.1.2.1 EasyLinkLayer()

Constructor.

**Parameters** 

| in | flagAddressFiltering | Enable address filtering, default=false |
|----|----------------------|---|
|----|----------------------|---|

Note

Basic usage: no parameter, no filtering

Warning

Both RX and TX need to have the same configuration

### 5.1.3 Member Function Documentation

### 5.1.3.1 begin()

```
EasyLink_Status EasyLinkLayer::begin ( )
```

Initialise and start the radio.

Note

Basic usage: begin()

Returns

EasyLink\_Status\_Success or EasyLink\_Status\_Param\_Error

## 5.1.3.2 getAddressDestinationRX()

Get destination from RX message.

**Parameters** 

| out | ieeeAddress | destination address, uint8_t[8] |
|-----|-------------|---------------------------------|
|-----|-------------|---------------------------------|

Warning

Not part of basic usage

#### 5.1.3.3 getAddressFilter()

Get address to addresses filter.

#### **Parameters**

| in  | slot    | 0, 1 or 2                      |
|-----|---------|--------------------------------|
| out | address | IEEE address read from filters |

#### Returns

EasyLink\_Status\_Success or EasyLink\_Status\_Param\_Error

Note

The filter contains up to three addresses.

#### Warning

Not part of basic usage

### 5.1.3.4 getAddressLocal()

Get local address.

IEEE address uint8\_t[8]

#### **Parameters**

| out | ieeeAddress | pointer to uint8_t[8] |
|-----|-------------|-----------------------|

#### Returns

EasyLink\_Status

## 5.1.3.5 getRSSIRX()

```
uint8_t EasyLinkLayer::getRSSIRX ( )
```

Get RSSI from RX message.

#### Returns

**RSSI** level

### 5.1.3.6 receive()

Receive a message.

#### **Parameters**

| out | payload | pointer to the payload                               |
|-----|---------|--|
| out | size    | size of the payload                                  |
| in  | ms      | period to receive, default=2 seconds, time-out after |

#### Returns

EasyLink\_Status\_Success or EasyLink\_Status\_Param\_Error

## Note

Maximum payload size is EASYLINK\_MAX\_DATA\_LENGTH=128 Basic usage: receive(&payload, &size)

### 5.1.3.7 setAddressDestinationTX()

```
\begin{tabular}{ll} \begin{tabular}{ll} void EasyLinkLayer::setAddressDestinationTX ( \\ AddressIEEE\_t ieeeAddress ) \end{tabular}
```

Set destination for TX message.

#### **Parameters**

| in | ieeeAddress | destination address, uint8_t[8] |
|----|-------------|---------------------------------|

#### Warning

Not part of basic usage

#### 5.1.3.8 setAddressFilter()

Set address to addresses filter.

#### **Parameters**

| in | slot    | 0, 1 or 2                      |
|----|---------|--------------------------------|
| in | address | IEEE address to add to filters |

#### Returns

EasyLink\_Status\_Success or EasyLink\_Status\_Param\_Error

#### Note

Up to three addresses can be added to the filter. Recommended allocation of addresses

- Hub
  - not used
  - specific IEEE address of the hub
  - generic address for commissioning nodes to hub

#### Node

- general broadcast from hub to all nodes
- specific IEEE address of the node
- not used

#### Warning

Not part of basic usage

#### 5.1.3.9 transmit()

Send a message.

## **Parameters**

| in | payload | pointer to the payload |
|----|---------|------------------------|
| in | size    | size of the payload    |

#### Returns

EasyLink\_Status\_Success or EasyLink\_Status\_Param\_Error

#### Note

Maximum payload size is EASYLINK\_MAX\_DATA\_LENGTH=128 Basic usage: transmit(payload, size)

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

- EasyLinkLayer.h
- · EasyLinkLayer.cpp

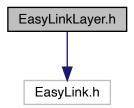
| Doto | Struc | +    | Daai | ıman | tation |
|------|-------|------|------|------|--------|
| vala | อแนน  | lure | DUCL | umen | lalion |

## **File Documentation**

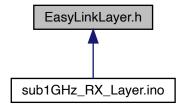
## 6.1 EasyLinkLayer.h File Reference

Library header.

#include "EasyLink.h"
Include dependency graph for EasyLinkLayer.h:



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



18 File Documentation

### **Data Structures**

class EasyLinkLayer
 EasyLinkLayer.

#### **Macros**

#define EASYLINKLAYER\_RELEASE
 Release.

## **Typedefs**

```
    typedef uint8_t AddressIEEE_t[8]
    IEEE address.
```

• typedef AddressIEEE\_t AddressFilter\_t[EASYLINK\_MAX\_ADDR\_SIZE] Addresses filter array.

## 6.1.1 Detailed Description

```
Library header.

Additional layer for EasyLink

Project sub1GHz_TX_Layer

Developed with embedXcode+: https://embedXcode.weebly.com

Author

Rei Vilo
Rei Vilo

Date

05 Nov 2019 11:00

Version

103

Copyright

(c) Rei Vilo, 2019

CC = BY SA NC
```

See also

ReadMe.txt for references

## 6.1.2 Typedef Documentation

#### 6.1.2.1 AddressFilter\_t

```
typedef AddressIEEE_t AddressFilter_t[EASYLINK_MAX_ADDR_SIZE]
```

Addresses filter array.

EASYLINK\_MAX\_ADDR\_SIZE set to 3

## 6.1.2.2 AddressIEEE\_t

```
typedef uint8_t AddressIEEE_t[8]
```

IEEE address.

uint8\_t[8]

Note

CC13xx is little endian

- (uint8\_t[8])00.12.4B.00.0A.27.CD.6A
- but (uint64\_t)6ACD270A004B1200

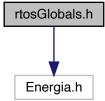
See also

http://www.ti.com/lit/ug/swcull7h/swcull7h.pdf

## 6.2 rtosGlobals.h File Reference

Global variables for Energia MT project.

```
#include "Energia.h"
Include dependency graph for rtosGlobals.h:
```



20 File Documentation

## 6.2.1 Detailed Description

Global variables for Energia MT project.

<#details#>

Developed with embedXcode+: https://embedXcode.weebly.com

Author

Rei Vilo

Rei Vilo

Date

05 Nov 2019 10:56

Version

103

Copyright

(c) Rei Vilo, 2019

CC = BY SA NC

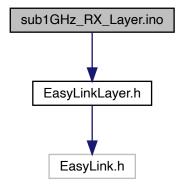
See also

ReadMe.txt for references

## 6.3 sub1GHz\_RX\_Layer.ino File Reference

Main sketch.

#include "EasyLinkLayer.h"
Include dependency graph for sub1GHz\_RX\_Layer.ino:



#### **Macros**

```
    #define WITH_ADDRESS_FILTERING 1
        Address filtering.
    #define WITHOUT_ADDRESS_FILTERING 0
        desacticated
    #define ADDRESS_FILTERING WITH_ADDRESS_FILTERING
        mode
```

#### **Functions**

```
    void printAddress (AddressIEEE_t address, bool prefix=false)
        Print IEEE address.

    void printFilter (String title="Filter")
        Print the addresses of the filter.

    void setup ()
    void loop ()
```

### **Variables**

See also

ReadMe.txt for references

```
EasyLinkLayer myLink (true)
AddressIEEE_t addressHub = { 0x00, 0x12, 0x4B, 0x00, 0x0B, 0xCA, 0x27, 0x82 }
AddressIEEE_t addressNode = { 0x00, 0x12, 0x4B, 0x00, 0x0A, 0x27, 0xCD, 0x6A }
AddressIEEE_t addressLocal
uint16_t value
```

## 6.3.1 Detailed Description

```
Main sketch.

RX example for EasyLink additional layer

Developed with embedXcode+

Author

Rei Vilo

https://embeddedcomputing.weebly.com

Date

05 Nov 2019 10:56

Version

103

Copyright

(c) Rei Vilo, 2019

CC = BY SA NC
```

22 File Documentation

### 6.3.2 Macro Definition Documentation

### 6.3.2.1 WITH\_ADDRESS\_FILTERING

```
#define WITH_ADDRESS_FILTERING 1
```

Address filtering.

activated

#### 6.3.3 Function Documentation

#### 6.3.3.1 printAddress()

Print IEEE address.

### Parameters

| address | IEEE address               |
|---------|----------------------------|
| prefix  | default=false, true=add 0x |

### 6.3.3.2 printFilter()

Print the addresses of the filter.

#### **Parameters**

| title | default="Filter" |
|-------|------------------|
|-------|------------------|

## Index

transmit

```
AddressFilter_t
     EasyLinkLayer.h, 19
AddressIEEE t
     EasyLinkLayer.h, 19
begin
     EasyLinkLayer, 11
EasyLinkLayer, 9
    begin, 11
     EasyLinkLayer, 10
    getAddressDestinationRX, 11
     getAddressFilter, 11
    getAddressLocal, 12
    getRSSIRX, 12
     receive, 13
    setAddressDestinationTX, 13
     setAddressFilter, 13
    transmit, 14
EasyLinkLayer.h, 17
     AddressFilter_t, 19
     AddressIEEE_t, 19
get Address Destination RX\\
     EasyLinkLayer, 11
getAddressFilter
     EasyLinkLayer, 11
getAddressLocal
     EasyLinkLayer, 12
getRSSIRX
     EasyLinkLayer, 12
printAddress
    sub1GHz_RX_Layer.ino, 22
printFilter
     sub1GHz_RX_Layer.ino, 22
receive
     EasyLinkLayer, 13
rtosGlobals.h, 19
setAddressDestinationTX
     EasyLinkLayer, 13
setAddressFilter
     EasyLinkLayer, 13
sub1GHz_RX_Layer.ino, 20
    printAddress, 22
     printFilter, 22
     WITH_ADDRESS_FILTERING, 22
```

EasyLinkLayer, 14 WITH ADDRESS FILTERING sub1GHz\_RX\_Layer.ino, 22