Simblee™
Connecting Everyone and Everything

1601 Pacific Coast Hwy • Suite 290 • Hermosa Beach • CA • 90254
www.simblee.com • Tel: 949.610.0008 • contact@simblee.com

# Getting Started with SimbleeCOM

*SimbleeCOM is a high performance, low latency, wireless professional protocol that works extremely well in high noise environments. SimbleeCOM supports both non-encrypted and encrypted communication in a pre-shared static network or dynamic network.*

## Overview

You can easily dive into SimbleeCOM by checking out the examples available. In the Arduino IDE, select:

**File > Examples > SimbleeCOM**

to see and try a series of examples on SimbleeCOM. The best way to get started with SimbleeCOM is by using the examples and through experimentation.

A simple SimbleeCOM sketch contains the following functions:

```
void setup(){

}

void loop(){

}

// Function below used, if receiving data from a Simblee device

void SimbleeCOM_onReceive(unsigned int esn, const char *payload, int len, int rssi){

}
```

## Unique ESN (Electronic Serial Number)

Each Simblee device has a factory assigned ESN that is used to uniquely identify it on the network. You can access this ESN by calling the following function:

**SimbleeCOM.getESN()**

The ESN returned by the function above will be a 32-bit unique factory ESN.

You can view the example sketch “GetESN” from the Arduino IDE examples to see it in action.

## Adding the SimbleeCOM library into your sketch

Add the following line of code to the very beginning of your sketch:

```
#include “SimbleeCOM.h”
```

## SimbleeCOM Setup Functions

### SimbleeCOM.mode

There are 2 modes for SimbleeCOM to choose from:

```
SimbleeCOM.mode = LOW_LATENCY;
```

SimbleeCOM’s low latency mode enables 3ms latency along with 10us jitter, for faster communication between Simblee devices.

```
SimbleeCOM.mode = LONG_RANGE;
```

SimbleeCOM’s long range mode enables 12ms latency along with 10us jitter for up to 4x the range of low latency mode.

### SimbleeCOM.txPowerLevel

You can set the radio transmission power level of your Simblee at the following 4dBm increments:

```
SimbleeCOM.txPowerLevel = +4; //default value is +4 (-20, -16, -12, -8, -4, 0, +4)
```

### SimbleeCOM.proximityMode

Proximity mode brings the range of the Simblee module to a very close proximity. This is for use in security applications, or when connection to a Simblee device is requires the user to be very close.

```
SimbleeCOM.proximityMode(FALSE); //proximity mode is set to FALSE on default
```

### SimbleeCOM.begin

Begins the SimbleeCOM stack.

```
SimbleeCOM.begin();
```

### SimbleeCOM.end

Ends the SimbleeCOM stack.

```
SimbleeCOM.end();
```

# SimbleeCOM Communication Functions

This section will cover the send and receive functions of SimbleeCOM.

## SimbleeCOM.send

This function allows you to send data via SimbleeCOM. Here is the format:

*SimbleeCOM.send(const char *data, int len);*

Example:

```
char payload[] = { 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15 };

SimbleeCOM.send(payload, sizeof(payload));
```

## SimbleeCOM_onReceive

This function returns data from the radio.

*void SimbleeCOM_onReceive(unsigned int esn, const char *payload, int len, int rssi){}*

Example:

```
void SimbleeCOM_onReceive(unsigned int esn, const char *payload, int len, int rssi)

{

    printf("%d ", rssi); // prints RSSI to the serial port

    printf("0x%08x ", esn); // prints ESN of sender to the serial port

    for (int i = 0; i < len; i++)

        printf("%02x ", payload[i]); // prints payload data to the serial port

    printf("\n");

}
```