
Adafruit IO Arduino Library Documentation

Release 2.7.4

ToddTreece,abachman,tdicola,jwcooper,ladyada

Jun 19, 2018

Contents

| | | |
|----------|--|----------|
| 1 | Dependencies | 3 |
| 1.1 | Adafruit Feather HUZZAH ESP8266 & Adafruit HUZZAH ESP8266 Breakout | 3 |
| 1.2 | Adafruit Feather HUZZAH32 (ESP32) | 3 |
| 1.3 | Adafruit Feather M0 WiFi with ATWINC1500 | 3 |
| 1.4 | Arduino MKR1000 | 4 |
| 1.5 | Adafruit WICED Feather WiFi | 4 |
| 1.6 | Adafruit Feather 32u4 FONA | 4 |
| 1.7 | Adafruit Ethernet FeatherWing | 4 |
| 2 | Usage | 5 |
| 2.1 | WiFi (ESP8266, M0 WINC1500, WICED) | 5 |
| 2.2 | Cellular (32u4 FONA) | 5 |
| 2.3 | Ethernet (Ethernet FeatherWing) | 6 |
| 3 | Examples | 7 |
| 4 | Compatibility | 9 |

This library provides a simple device independent interface for interacting with Adafruit IO using Arduino. It allows you to switch between WiFi (ESP8266, M0 WINC1500, & WICED), Cellular (32u4 FONA), and Ethernet (Ethernet FeatherWing) with only a two line change in your sketch.

This library requires the latest version of the [Arduino IDE](#) (tested with v1.6.10).

1.1 **Adafruit Feather HUZDAH ESP8266 & Adafruit HUZDAH ESP8266 Breakout**

- Latest version of the [ESP8266 Arduino Core](#)
- Latest version of the [Adafruit MQTT Library](#)
- Latest version of the [Arduino HTTP Client Library](#)

1.2 **Adafruit Feather HUZDAH32 (ESP32)**

- Latest version of the [ESP32 Arduino Core](#)
- Latest version of the [Adafruit MQTT Library](#)
- Latest version of the [Arduino HTTP Client Library](#)

1.3 **Adafruit Feather M0 WiFi with ATWINC1500**

- Latest version of the [Arduino SAMD Arduino Core](#)
- Latest version of the [Adafruit SAMD Arduino Core](#)
- Latest version of the [Adafruit MQTT Library](#)
- Latest version of the [WiFi101 Library](#)
- Latest version of the [Arduino HTTP Client Library](#)

You will also need to add the SSL certificate for *io.adafruit.com* using the guide on the Adafruit Learning System.

1.4 Arduino MKR1000

- Latest version of the [Arduino SAMD Arduino Core](#)
- Latest version of the [Adafruit MQTT Library](#)
- Latest version of the [WiFi101 Library](#)
- Latest version of the [Arduino HTTP Client Library](#)

You will also need to add the SSL certificate for *io.adafruit.com* using the MKR1000 SSL utility.

1.5 Adafruit WICED Feather WiFi

- Latest version of the [Adafruit WICED Arduino Core](#)
- Latest version of the [Adafruit MQTT Library](#)
- Latest version of the [Arduino HTTP Client Library](#)

1.6 Adafruit Feather 32u4 FONA

- Latest version of the [Adafruit AVR Arduino Core](#)
- Latest version of the [Adafruit MQTT Library](#)
- Latest version of the [Adafruit FONA Library](#)

1.7 Adafruit Ethernet FeatherWing

The Ethernet FeatherWing will also require a 32u4, M0, or ESP8266 based Feather. Any Arduino cores required by the host Feather board will also be required.

- Latest version of the [Adafruit Ethernet2 Library](#)
- Latest version of the [Adafruit MQTT Library](#)
- Latest version of the [Arduino HTTP Client Library](#)

The included examples sketches will walk you through all of the features of the library. They can be used on all platforms, but they default to WiFi. To change between platforms, you will need to change two lines of code in the `config.h` tab of the example. It is recommended that you start with one of the Adafruit WiFi feathers before moving on to cellular or ethernet.

For all examples, you will need to set `IO_USERNAME` and `IO_KEY` in the `config.h` tab.

The following sections demonstrate how to switch between WiFi, cellular, and ethernet.

2.1 WiFi (ESP8266, M0 WINC1500, WICED)

If you are using the included examples, you do not need to change anything for the Adafruit WiFi Feathers. All WiFi based Feathers (ESP8266, M0 WiFi, WICED) will work with the examples out of the box.

You will need to add your WiFi information to the `WIFI_SSID` and `WIFI_PASS` defines in the `config.h` tab.

2.2 Cellular (32u4 FONA)

For FONA, you will only need to change from the default WiFi constructor to the FONA specific constructor in the `config.h` tab. The rest of the sketch remains the same.

You will need to comment out these WiFi lines in `config.h`:

```
#include "AdafruitIO_WiFi.h"
AdafruitIO_WiFi io(IO_USERNAME, IO_KEY, WIFI_SSID, WIFI_PASS);
```

and uncomment the FONA lines in `config.h`:

```
#include "AdafruitIO_FONA.h"
AdafruitIO_FONA io(IO_USERNAME, IO_KEY);
```

If your carrier requires APN info, you can set it by adding a call to `io.setAPN()` after `io.connect()` in the `setup()` function of the sketch.

```
void setup() {  
  
  // start the serial connection  
  Serial.begin(115200);  
  
  // connect to io.adafruit.com  
  io.connect();  
  
  io.setAPN(F("your_apn"), F("your_apn_user"), F("your_apn_pass"));  
  
}
```

2.3 Ethernet (Ethernet FeatherWing)

For Ethernet, you will only need to change from the default WiFi constructor to the Ethernet specific constructor in the `config.h` tab. The rest of the sketch remains the same.

You will need to comment out these WiFi lines in `config.h`:

```
#include "AdafruitIO_WiFi.h"  
AdafruitIO_WiFi io(IO_USERNAME, IO_KEY, WIFI_SSID, WIFI_PASS);
```

and uncomment the Ethernet lines in `config.h`:

```
#include "AdafruitIO_Ethernet.h"  
AdafruitIO_Ethernet io(IO_USERNAME, IO_KEY);
```

CHAPTER 3

Examples

Examples for this repository are found in the */examples* folder of this repository.

They're also fully illustrated and documented on the [Adafruit IO Basics Series on the Adafruit Learning System](#)

CHAPTER 4

Compatibility

Last tested with library v2 . 7 . 4 on 06/18/2018 for ESP8266, ESP32 and M0 WiFi Targets

Previously tested on all targets using v2.0.0 on 08/10/2016.

| Example | ESP8266 | ESP82 | M0 WiFi | WICED | Ethernet* | FONA 32u4 | MKR1000 |
|--------------------|---------|-------|---------|-------|-----------|-----------|---------|
| 00_publish | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ? |
| 01_subscribe | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ? |
| 02_pubsub | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ? |
| 03_multiple_feeds | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ? |
| 04_location | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ? |
| 05_type_conversion | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ? |
| 06_digital_in | ✓ | ✓ | ✓ | ✓ | ? | ✓ | ? |
| 07_digital_out | ✓ | ✓ | ✓ | ✓ | ? | ✓ | ? |
| 08_analog_in | ✓ | ✓ | ✓ | ✓ | ? | ✓ | ? |
| 09_analog_out | ✓ | ✓ | ✓ | ✓ | ? | ✓ | ? |