

Adafruit ATWINC1500 WiFi Breakout

Created by lady ada



Last updated on 2018-11-11 10:18:14 PM UTC

Guide Contents

Guide Contents	2
Overview	3
Pinouts	6
Power Pins	6
SPI Pins	6
Other SPI Interface Pins	6
Assembly	8
Prepare the header strip:	8
Add the breakout board:	9
And Solder!	9
Wiring & Test	12
Install the Library	12
Check Connections & Version	13
Scanning WiFi	15
Connect & Read Webpage	16
Updating Firmware	18
Updating SSL Certificates	23
What SSL/TLS support is available with the WINC1500?	26
Downloads	27
Datasheets & Files	27
Schematic	27
Fabrication Print	27

Overview



Connect your Arduino to the Internet with this fine new FCC-certified WiFi module from Atmel. This 802.11bgn-capable WiFi module is the best new thing for networking your devices, with SSL support and rock solid performance - running our adafruit.io MQTT demo for a full weekend straight with no hiccups (it would have run longer but we had to go to work, so we unplugged it). We like these so much, they've completely replaced the CC3000 module on all our projects.



The ATWINC1500 uses SPI to communicate, so with about 5 or 6 wires, you can get your wired up and ready to go. Right now the Atmel-supplied library works great with Arduino M0 & M4, but won't work with 328P, or 32u4 and may not work on other Arduinos. You can clock it as fast as 12MHz for speedy, reliable packet streaming. Communication is done through the standard Client & Server interface so all your Ethernet & older WiFi code is easy to adapt. Scanning/connecting to networks is very fast, a few seconds.



This module works with 802.11b, g, or n networks & supports WEP, WPA and WPA2 encryption. The datasheet says it can do Soft-AP mode but we don't have any code to actually use that.

Since this is our new favoritest SPI-protocol WiFi module we've decided to make a little breakout for it. The breakout comes with level shifting on all the input pins so you can use it with 3V or 5V logic. A 3.3V voltage regulator that can handle the 300mA spikes lets you power from 3-5.5VDC. There's also 3 LEDs that you can control over the SPI interface (part of the library code) or you can have controlled by the Arduino library. They'll light up when connected to an SSID, or transmitting data.



Comes with a stick of header you can solder on, to plug into a breadboard and a set of tutorials & code so you can follow along!

Pinouts



The ATWINC1500 module does have a bunch of pins, but they're pretty easy to understand. Lets check it out!

Power Pins

- Vin this is the power-in pin. Connect to 3.3 5.5VDC The wifi module can draw up to 300mA during transmit (for small blips of time) So make sure that your power supply can supply it.
- GND ground for signal and power

SPI Pins

This is how you send and receive data from the module

- SCK SPI clock input, 3V or 5V compliant
- MISO SPI data out from module, 3.3V line level
- MOSI SPI data into module, 3V or 5V compliant
- CS SPI chip select, pull down when transmitting to/from the ATWINC. By default this is pulled to Vin with a 100K resistor

Other SPI Interface Pins

- EN Enables the entire module, by default tied low with a 100K resistor. Tie to 3-5V to keep the module on all the time, connect to a ground signal to disable the module
- IRQ Interrupts from the module, connect to your microcontroller's INT input line. 3.3V logic level
- RST Module reset, by default tied low with a 100K resistor. Pull high to bring out of reset.
- Wake wake input signal, used to wake up the module (not used in existing code, but available if you can figure it out!) 3-5V logic in
- CFG allows you to select between SPI (default) or UART data transport. Since we don't have any UART code,

keep disconnected

• RXD/TXD - UART data transport pins. Since we don't have any UART code, keep disconnected

Assembly





Prepare the header strip: Cut the strip to length if necessary. It will be easier to solder if you insert it into a breadboard - long pins down



Add the breakout board:

Place the breakout board over the pins so that the short pins poke through the breakout pads



And Solder! Be sure to solder all pins for reliable electrical contact.

(For tips on soldering, be sure to check out our Guide to Excellent Soldering (https://adafru.it/aTk)).





You're done! Check your solder joints visually and continue onto the next steps

Wiring & Test



For this initial demo we'll be using an Arduino UNO to connect. You can also use an Arduino Zero, but you'll have to use the ICSP 6-pin header not pins 11,12,13

- Vin connect this to 3.3V or 5V, whichever is the logic voltage of the microcontroller you're using. For UNO this will be 5V, for Zero its 3.3V
- GND connect to common ground
- SCK Connect to SPI clock. On UNO this is pin #13 on Zero its on the 6-pin ISP header (https://adafru.it/iCE)
- MISO Connect to SPI MISO. On UNO this is pin #12 on Zero its on the 6-pin ISP header (https://adafru.it/iCE)
- MOSI Connect to SPI MOSI. On UNO this is pin #11 on Zero its on the 6-pin ISP header (https://adafru.it/iCE)

For the remaining pins, you can be a little flexible:

- CS Connect to any digital I/O pin, we use #8 by default
- EN connect this to 3.3V or 5V, whichever is the logic voltage of the microcontroller you're using. For UNO this will be 5V, for Zero its 3.3V; later on if you want to enable/disable the module, connect it to a digial I/O pin
- IRQ Connect to any digital I/O pin, preferrably one with an interrupt capability. We use #7 by default
- RST Connect to any digital I/O pin. We use #4 by default

You can change these pins later but for now use them, so you can verify your setup!

Install the Library

We will start by installing the official Arduino WiFi101 library (https://adafru.it/kUF).

We want the latest version so visit the Library Manager



Type in wifi101 and when the library comes up, click Install or Update to make sure its the most recent one!

If you're not familiar with installing Arduino libraries, please visit our tutorial: All About Arduino Libraries (https://adafru.it/aYM)!

Restart the Arduino IDE.

Use the very latest version of the Arduino IDE!

Check Connections & Version

Before we start, its important to verify you have the right setup & firmware version.

Load up the WiFi101->CheckWifi101Firmware sketch

💿 в	link Arduino 1	1.6.9		RTClib	•	
File	Edit Sketch 1	Tools Help		RTCZero	•	
	New	Ctrl+N		SAMD_AnalogCorrection	×	
	Open	Ctrl+O		SPI	×	
	Open Recent			Temboo	×	
	Sketchbook			TinyWireM	•	
	Examples		•	USBHost	•	. 17
	Close	Ctrl+W		WiFi101	I	AP_SimpleWebServer
	Save	Ctrl+S		Wire	1	CheckWifi101FirmwareVersion
	Save As	Ctrl+Shift+S		RETIRED	1	ConnectNoEncryption
	David Catura	Chill Childh D		Formation from Contained Diseasion		ConnectWithWEP
	Page Setup Print	Ctrl+Shift+P		Examples from Custom Libraries		ConnectWithWPA
	Print	Ctrl+P		AccelStepper]	FirmwareUpdater
	Preferences	Ctrl+Comma		Adafruit BME680 Library	1	MDNS WiFiWebSeprer

Note that to use the official Arduino WiFi101 Library, we must configure it to use the pins specific to the

```
ATWINC1500 Breakout. With each example sketch, you'll need to add WiFi.setPins(8,7,4); to the top of the setup function, before WiFi.status() is called.
```

//Configure pins for Adafruit ATWINC1500 Breakout
WiFi.setPins(8,7,4);

Like so:

💿 CheckWifi101FirmwareVersion Arduino 1.6.9	x
<u>File E</u> dit <u>S</u> ketch <u>T</u> ools <u>H</u> elp	
🛛 🔶 🕒 🔛 🔛 🔛 🔛 🔛).
CheckWifi101FirmwareVersion §	
*/	*
<pre>#include <spi.h></spi.h></pre>	
<pre>#include <wifi101.h></wifi101.h></pre>	
<pre>void setup() {</pre>	
WiFi.setPins(8,7,4);	
	=
// Initialize serial	_
Serial.begin(9600);	
while (!Serial) {	
; // wait for serial port to connect. Needed for native USB port only }	
}	
// Print a welcome message	
<pre>Serial.println("WiFi101 firmware check.");</pre>	
<pre>Serial.println();</pre>	
// Check for the presence of the shield	
<pre>Serial.print("WiFi101 shield: ");</pre>	
if (WiFi.status() == WL_NO_SHIELD) {	
Serial.println("NOT PRESENT");	

Upload to your Arduino and open up the Serial Console at 9600 baud:

💿 COM220 (Arduino/Genuino Uno)		. 🗆 🗙
		Send
WINC1500 firmware check.		<u>^</u>
WINC1500: DETECTED		
Firmware version installed: 19.4.4		
Firmware version required : 19.4.4		
Check result: PASSED		
V Autoscroll	Both NL & CR 👻 9	9600 baud 👻

You should see the firmware version. If your version has not **PASSED** follow this page to update your firmware! (https://adafru.it/vff)

If you have version 19.3 or less, the firmware is too old

If you get not response, the firmware is either waaay to old, or something is amiss with your wiring!

If you get a warning that says

WiFi101 shield: DETECTED Firmware version installed: 19.5.2 Latest firmware version available : 19.5.4

Specifically with 19.5.4 and 19.5.2, you can continue without updating

💿 COM220 (Arduino/Genuino Uno)	
	Send
WINC1500 firmware check.	
WINC1500: DETECTED	
Firmware version installed: 19.3.0	
Firmware version required : 19.4.4	
Check result: NOT PASSED - The firmware version on the WINC1500 do not match the version required by the library, you may experience issues or failures.	E
	-
✓ Autoscroll Both NL & CR	600 baud 🛛 👻

Scanning WiFi

Now that you have the right firmware version, lets scan for network!

Run the WiFi101->ScanNetworks example to see a list of available visible networks

Don't forget to add WiFi.setPins(8,7,4) at the top of setup()



Connect & Read Webpage

OK finally you get to connect and read some data!

Open up the WiFi101->WiFi101WebClient example, then edit the ssid and pass variables to contain your network and password

```
34
35
36 char ssid[] = "adafruit"; // your network SSID (name)
37 char pass[] = "supersekret"; // your network password (use for WPA, or use as key for WEP)
8 int keyIndex = 0; // your network key Index number (needed only for WEP)
9
40 int status = WL_IDLE_STATUS;
41 // if you don't want to use DNS (and reduce your sketch size)
42 // use the numeric IP instead of the name for the server:
43 // use the numeric IP instead of the name for the server:
44 // use the numeric IP instead of the name for the server:
45 // use the numeric IP instead of the name for the server:
46 // use the numeric IP instead of the name for the server:
47 // use the numeric IP instead of the name for the server:
48 // use the numeric IP instead of the name for the server:
49 // use the numeric IP instead of the name for the server:
40 // use the numeric IP instead of the name for the server:
41 // if you have the numeric IP instead of the name for the server:
42 // use the numeric IP instead of the name for the server:
43 // use the numeric IP instead of the name for the server:
44 // use the numeric IP instead of the name for the server:
45 // use the numeric IP instead of the name for the server:
46 // use the numeric IP instead of the name for the server:
47 // use the numeric IP instead of the name for the server:
48 // use the numeric IP instead of the name for the server:
49 // use the numeric IP instead of the name for the server:
40 // use the numeric IP instead of the name for the server:
41 // use the numeric IP instead of the name for the server:
41 // use the numeric IP instead of the name for the server:
41 // use the numeric IP instead of the name for the server:
42 // use the numeric IP instead of the name for the server:
43 // use the numeric IP instead of the name for the server:
44 // use the numeric IP instead of the name for the server:
44 // use the numeric IP instead of the name for the server:
44 // use the numeric IP instead of the name for the server:
44 // use the numeric IP instead of the name for the se
```

Add the following lines at the top of setup()

//Configure pins for Adafruit ATWINC1500 Feather WiFi.setPins(8,7,4);

It will connect to the website in server and read the webpage manually:

Send Attempting to connect to SSID: 10th Floor Connected to wifi SSID: 10th Floor IP Address: 10.0.0.92 signal strength (RSSI):-46 dBm
Connected to wifi SSID: 10th Floor IP Address: 10.0.0.92
SSID: 10th Floor IP Address: 10.0.0.92
IP Address: 10.0.0.92
signal strength (RSSI):-46 dBm
Starting connection to server
connected to server
HTTP/1.1 200 OK
Date: Wed, 21 Sep 2016 22:33:32 GMT
Expires: -1
Cache-Control: private, max-age=0
Content-Type: text/html; charset=ISO-8859-1
P3P: CP="This is not a P3P policy! See https://www.google.com/support/accour
Server: gws
X-XSS-Protection: 1; mode=block
X-Frame-Options: SAMEORIGIN
Set-Cookie: NID=87=kmIfFqMh8M_7gv9BDt982kjfllc-imsU4d-X3fZthsS_ye4Sfa541V33C
Accept-Ranges: none
Vary: Accept-Encoding
Connection: close
html

That's it! pretty easy, huh? There's other examples you can try such as server mode, UDP data transmission & SSL

Updating Firmware

If you're running 19.5.2 there is no need to update to 19.5.4, despite what the WiFi101 library says it should work just fine and you could brick your device by updating, it is not perfectly foolproof! Thanks :)

As new versions of the WiFi101 library come out, you may end up getting a complaint that the library and WINC1500 firmware are out of sync:

💿 COM21 (Adafruit Feather M0)	
	Send
WiFi101 firmware check.	
WiFi101 shield: DETECTED	
Firmware version installed: 19.4.4	
Latest firmware version available : 19.5.2	
Check result: NOT PASSED	
- The firmware version on the shield do not match the	
version required by the library, you may experience	
issues or failures.	
	115200 baud 👻

No problem - you can update the firmware through your Arduino/compatible! Start by loading up the FirmwareUpdater sketch



If you are using a Feather M0 or WINC1500 breakout, don't forget to update the pins as necessary with setPins()!

If you are using a WiFi101 or WINC1500 shield, you do not have to add setPins() code



Upload it to your board. Make sure the Serial console is not open before or after uploading.

Then select the Updater tool built into the IDE

💿 FirmwareUpdater	Arduino 1.8.1	
File Edit Sketch Too	ols Help	
FirmwareUpda	Auto Format Archive Sketch Fix Encoding & Reload	Ctrl+T
<pre>// payloadL } UartPacket;</pre>	Serial Monitor Serial Plotter	Ctrl+Shift+M Ctrl+Shift+L
static const	WiFi101 Firmware Updater	
<pre>#define CMD_R #define CMD_W #define CMD_E #define CMD_M #define CMD_H</pre>	Board: "Adafruit Feather M Port: "COM21 (Adafruit Fea Get Board Info Programmer: "USBtinyISP"	ather M0)"
#define CMD_H	Burn Bootloader	

Select the right COM port, and click Test Connection

6	WiFi101 Firmware/Certificates Updater	
n	1. Select port of the WiFi module	
A	If the port is not listed click "Refresh list" button to regenerate the list	
I	COM1	
A	COM21	Refresh list
K.		Test connection

If all is good you'll get a confirmation



Next, select the firmware - we of course recommend the latest version!

If you don't see the right/matching version you may need to update the IDE

💿 WiFi101 Firmware/Certificate	es Updater	
1. Select port of the WiFi module	2	
If the port is not listed click "Re	fresh list" button to regenerate the list	
COM1 COM21		Refresh list
		Test connection
2. Update firmware		
Select the firmware from the dro	opdown box below	
WINC1501 Model B (19.5.2)		•
WINC1501 Model B (19.5.2)		
WINC1501 Model B (19.4.4) WINC1501 Model A (19.4.4)		
3. Update SSL root certificates		
Add domains in the list below us	ing "Add domain" button	
arduino.cc:443		Add domain
		Remove domain
	Upload Certificates to WiFi module	
		,

Once you feel ready - make sure the USB cable is connected solidly! Click Update Firmware

💿 WiFi101 Firmware/Certificates Updater	
1. Select port of the WiFi module	
If the port is not listed click "Refresh list" button to regenerate the list	
COM1 COM21	Refresh list
	Test connection
2. Update firmware	
Select the firmware from the dropdown box below	
WINC1501 Model B (19.5.2)	-
Update Firmware	
3. Update SSL root certificates	
Add domains in the list below using "Add domain" button	
arduino.cc:443	Add domain
	Remove domain
Upload Certificates to WiFi module	
Programmitig 413604 bytes	

And a minute or two later...

The firmware has been updated!
OK

Now you're ready to rock! Reload the Firmware Check sketch from before, this time you will see:



Updating SSL Certificates

If you're trying to connect to a computer or service via SSL and the connection is failing, you may need to update the certificates built into the WINC1500. By default it comes with many of the most popular SSL certificates but you may bump into a site that requires one that isnt included.

Its quite easy to update the certificates, you'll need to upload some code and run the uploaders but it only has to happen once

Start out by uploading the FirmwareUpdater sketch from WiFi101

💿 CheckWifi101Fi	irmwareVersion A	Arduino 1.8.1	G
File Edit Sketch	Tools Help		
New Open Open Recent	Ctrl+N Ctrl+O		
Sketchbook Examples	•		}
Close Save Save As	Ctrl+W Ctrl+S Ctrl+Shift+S	WiFi101 Examples for Adafruit Feather M0 I2S	AP_SimpleWebServer CheckWifi101FirmwareVersion ConnectNoEncryption ConnectWithWEP
Page Setup Print	Ctrl+Shift+P Ctrl+P	SAMD_AnalogCorrection SPI USBHost	ConnectWithWPA FirmwareUpdater
Preferences	Ctrl+Comma	Wire	MDNS_WiFiWebServer Provisioning_WiFiWebServer

If you are using a Feather M0 or WINC1500 breakout, don't forget to update the pins as necessary with setPins()!

If you are using a WiFi101 or WINC1500 shield, skip this step

💿 FirmwareUpdater Arduino 1.8.1
Eile Edit Sketch Tools Help
FirmwareUpdater § Endianess
<pre>void setup() {</pre>
WiFi.setPins(8,7,4,2);
Serial.begin(115200);

and upload it!

After uploading be sure to note what is the name of the COM or Serial port for the Arduino Zero or Feather...You'll need this for the next step

💿 Firmw	vareUpdater A	Arduino 1.6.7	-	
File Edit	t Sketch Tool	s Help		
		Auto Format	Ctrl+T	
		Archive Sketch		
Firmv	wareUpda	Fix Encoding & Reload		
	include	Serial Monitor	Ctrl+Shift+M	
23		Serial Plotter	Ctrl+Shift+L	
24 25 //	Define	Board: "Adafruit Feather M0 (Native USB Por	+)"	
26 //	/ If you		() ·	
27 //	/ guide	de Port		Serial ports
28 #d	lefine W			COM1
29 #d	lefine W	Programmer: "USBtinyISP"	1	COM3 (Adafruit Feather M0 (Native USB Port))
30 #d	lefine W	Burn Bootloader		
31 #d	define WINC EN 2 // or, tie EN to VCC		COM15	
32 11	The SPT ni	ns of the WINC1500 (SCK, MOSI, MIS	0) should be	

Upload it to your Feather. Make sure the Serial console is not open before or after uploading.

Then select the Updater tool built into the IDE

	💿 FirmwareUpdat	er A	arduino 1.8.1	
	File Edit Sketch	Tool	s] Help	
	FirmwareUpda		Auto Format Archive Sketch Fix Encoding & Reload	Ctrl+T
	<pre>// payloadL } UartPacket;</pre>		Serial Monitor Serial Plotter	Ctrl+Shift+M Ctrl+Shift+L
	static const		WiFi101 Firmware Updater	
	<pre>#define CMD_R #define CMD_W #define CMD_E #define CMD_M #define CMD_H</pre>		Board: "Adafruit Feather M0" Port: "COM21 (Adafruit Feather M0)" Get Board Info	Þ
			Programmer: "USBtinyISP" Burn Bootloader	×

Select the right COM port, and click Test Connection

	So WiFi101 Firmware/Certificates Updater	
n	1. Select port of the WiFi module	
A	If the port is not listed click "Refresh list" button to regenerate the list	
I	COM1	
A	COM21	Refresh list
K. L		Test connection

If all is good you'll get a confirmation



Now at the bottom of the page, click Add Domain and type in the URL of the site you want to access:



Then click Upload Certificates

💿 WiFi101 Firmware/Certificates Updater	
1. Select port of the WiFi module	
If the port is not listed click "Refresh list" button to regenerate the list	
COM1 COM21	Refresh list
	Test connection
2. Update firmware	
Select the firmware from the dropdown box below	
WINC 1501 Model B (19.5.2)	-
Update Firmware	
3. Update SSL root certificates	
Add domains in the list below using "Add domain" button	
arduino.cc:443 io.adafruit.com:443	Add domain
Upload Certificates to WiFi module	Remove domain

A few moments later...success!

So WiFi101 Firmware/Certificates Updater	
1. Select port of the WiFi module	
If the port is not listed click "Refresh list" button to regenerate the list	
COM1	
COM21	Refresh list
	Test connection
2. Update firmware	
Select the firmware from Success	
WINC1501 Model B (19 The certificates have been uploaded	
ОК	
3. Update SSL root certificates	
Add domains in the list below using "Add domain" button	
arduino.cc:443 io.adafruit.com:443	Add domain
	Remove domain
Upload Certificates to WiFi module	
Done!	
ounc.	

What SSL/TLS support is available with the WINC1500?

Officially Atmel lists TLS 1.0 & 1.1, however we have noticed that the firmwares shipping on boards today seem to also support TLS 1.2 (verified by checking the results of www.howsmyssl.com).

The supported ciphers are:

.

TLS_DHE_RSA_WITH_AES_128_GCM_SHA256	
TLS_RSA_WITH_AES_128_GCM_SHA256	
TLS_DHE_RSA_WITH_AES_128_CBC_SHA256	
TLS_DHE_RSA_WITH_AES_128_CBC_SHA	
TLS_RSA_WITH_AES_128_CBC_SHA256	
TLS_RSA_WITH_AES_128_CBC_SHA	

Downloads

Datasheets & Files

- All the details you could ever want are over at the Atmel website (https://adafru.it/kUD)
- EagleCAD PCB files on GitHub (https://adafru.it/oeL)
- Fritzing objects in the Adafrut Fritzing library (https://adafru.it/aP3)

Schematic

Click to embiggen. Please note we ship with the 'PB rev module, not 'PA!



Fabrication Print

Dimensions in inches

