XC4614 AT Commands Guide

Factory Settings:

Factory settings are 115200 baud, 8 bits, one stop bit, no parity. All commands must end in CRLF, and the commands themselves must be upper case.

The Baud can be changed via commands to the firmware, but not other settings.

Sending Commands:

The unit must be communicated through a serial port with the above settings. To do this, your Arduino must use its own serial to communicate (on pins 0 and 1).

This means, however, that you will not be able to use the Arduino Serial Monitor to communicate with the Arduino. So if you want to see the output of information, it's best to use another method, such as an LCD screen.

We have used a LEONARDO device, which has 2 Serials; one going to the computer, and one going to the module on pins (0,1) – this is perfect for testing via the computer:

```
void setup() {
  Serial1.begin(115200); //serial to device
  Serial.begin(115200);  //serial to computer
}
byte s = 0;
byte s1 = 0;
void loop() {
  if(Serial1.available())
    s1 = Serial1.read();
  if(Serial.available())
    s = Serial.read();

  if (s)
    Serial1.write(s);
  if (s1)
    Serial.write(s1);

  s = s1 = 0;
  delay(1);
}
```

This program simply acts as an echo between the module and the Computer, so that you can use the Serial Monitor to connect directly to the device.

Note that the Serial Monitor must be set up to send both newline and carriage return. These as ASCII code 13 followed by ASCII code 10. You can set this by the dropdown on the bottom of the serial monitor.

Most commands should return a blank line first, and then the payload, which could be either OK or ERROR, or something else.

Test and Information Commands:

AT

Command:

```
AT<CR><LF>
```

Response:

```
OK<CR><LF>
```

For testing that communication is occurring and the module is on. If there is no response, check that the module is turned on and the serial data switches are on. You might also have the wrong baud rate set if you get no response to the AT command.

AT+GMR

Command:

```
AT+GMR<CR><LF>
```

Response:

```
AT version:0.40.0.0(Aug 8 2015 14:45:58)<CR><LF>
SDK version:1.3.0<CR><LF>
Ai-Thinker Technology Co.,Ltd.<CR><LF>
Build:1.3.0.2 Sep 11 2015 11:48:04<CR><LF>
OK<CR><LF>
```

This command can be used to check the firmware version- different versions may have different commands built in. The version we are using is the 1.3.0.2 as seen above.

AT+RST

Command:

```
AT+RST<CR><LF>
```

Response:

```
(various boot data…)
Ai-Thinker Technology Co.,Ltd.<CR><LF>
<CR><LF>
ready<CR><LF>
```

This command is used to so a soft reset of the shield. Some mode changes (eg stopping a server) require a soft reset.

AT+IPR

Command:

```
AT+IPR=115200<CR><LF>
```

Response:

```
<CR><LF>
OK<CR><LF>
```

Use this command to change the serial port baud rate. The default is 115200, and valid values are 9600, 19200, 38400, 74880, 115200 and 230400.

Wifi and Network Connectivity:

AT+CWMODE

Query:

```
AT+CWMODE?<CR><LF>
```

Response:

```
+CWMODE:1<CR><LF>
<CR><LF>
OK<CR><LF>
```

Command:

```
AT+CWMODE=1<CR><LF>
Response:
<CR><LF>
OK<CR><LF>
```

This command sets whether the shield is an access point, station or both.

1: Station only (ie a device that connects to an access point).

2: Access Point only

3: Both Station and Access Point.

The query command can be used to find out the current state of the shield. By default, the shield is set to mode 3- both Station and Access Point. This setting is saved through power off and reset.

AT+CWLAP

Command:

```
AT+CWLAP<CR><LF>
```

Response:

```
+CWLAP:(3,"Arduino",-52,"74:da:38:3c:71:99",1,-46)<CR><LF>
+CWLAP:(3,"ESP",-54,"1a:fe:34:d4:96:19",11,-367)<CR><LF>
<CR><LF>
OK<CR><LF>
```

This command lists available access points. Needs CWMODE=1 command to be executed otherwise it will error. The fields in the brackets above are:

Encryption: 0=open, 1=WEP, 2=WPA_PSK, 3=WPA2_PSK, 4=WPA_WPA2_PSK

SSID: AKA Wifi network name

RSSI: or relative signal strength indicator, in dB. Higher (less negative) is stronger.

MAC: hardware address of router creating the access point.

Channel: Wifi channel number.

Frequency Offset: in kHz.

AT+CWJAP

Query:

```
AT+CWJAP?<CR><LF>
```

Response:

```
+CWJAP:"ESP","1a:fe:34:d4:96:19",11,-61<CR><LF>
<CR><LF>
OK<CR><LF>
```

Or:

```
No AP<CR><LF>
<CR><LF>
OK<CR><LF>
```

Command:

```
AT+CWJAP="ESP","Arduino!"<CR><LF>
```

Response:

```
WIFI CONNECTED<CR><LF>
WIFI GOT IP<CR><LF>
<CR><LF>
OK<CR><LF>
```

This command connects to a Wifi network. The two items in the quote marks above are the SSID name (as retrieved from the CWLAP command) and the password. This command takes a few seconds to execute, and you will see the above lines of the response occur slightly apart. It can also be used with the query syntax to test if there is a network connected. If the shield is reset or powered down with a Wifi network connected, it will attempt to reconnect when it restarts.

AT+CWQAP

Command:

```
AT+CWQAP<CR><LF>
```

Response:

```
WIFI DISCONNECT<CR><LF>
<CR><LF>
OK<CR><LF>
```

This command disconnects from a Wifi network. If you are not currently connected to a network, you won't get a 'WIFI DISCONNECT' response.

AT+CWSAP

Query:

```
AT+CWSAP?<CR><LF>
```

Response:

```
+CWSAP:"AI-THINKER_D492CE","",11,0,4<CR><LF>
<CR><LF>
OK<CR><LF>
```

Command:

```
AT+CWSAP="ESP","Arduino!",11,3,4<CR><LF>
```

Response:

```
<CR><LF>
OK<CR><LF>
```

Use this command to set the name and password of the access point. By default, the access point has the IP address 192.168.4.1. Settings made with this command will not change with power off or reset. The parameters given in the command and response are:

SSID: AKA Wifi network name

Password: must be empty for open network, and at least 8 characters for WPA.

Channel: 1-14 are valid, but generally 1, 6 and 11 are used to avoid interference.

Encryption: 0=open, 1=WEP, 2=WPA_PSK, 3=WPA2_PSK, 4=WPA_WPA2_PSK

Maximum number of connections: 1-4 are valid. This shield can handle at most 4.

AT+CWLIF

Command:

```
AT+CWLIF<CR><LF>
```

Response:

```
192.168.4.2,34:23:ba:e1:de:23<CR><LF>
<CR><LF>
OK<CR><LF>
```

This command lists the IP addresses and MAC addresses of all devices connected to the access point.

AT+CIFSR

Command:

```
AT+CIFSR<CR><LF>
```

Response in Access Point mode:

```
+CIFSR:APIP,"192.168.4.1"<CR><LF>
+CIFSR:APMAC,"1a:fe:34:d4:92:ce"<CR><LF>
<CR><LF>
OK<CR><LF>
```

Response in Station mode:

```
+CIFSR:STAIP,"192.168.4.2"<CR><LF>
+CIFSR:STAMAC,"18:fe:34:d4:92:ce"<CR><LF>
<CR><LF>
OK<CR><LF>
```

The CIFSR command can be used to give IP address info- probably most useful if you need to find out the IP address the shield has been given when connecting to a network. If the shield is in CWMODE=3 (AP and station mode), you will get both AP and station responses before the OK.

Server Commands:

AT+CIPMUX

Command:

```
AT+CIPMUX=1<CR><LF>
```

Response:

```
<CR><LF>
OK<CR><LF>
```

Although not a server command, this command needs to be used before a server can be set up. It allows multiple connections to be made to the shield at the same time. Even if the maximum number of connections is set to 1, it must be activated before starting a server.

AT+CIPSERVER

Command:

```
AT+CIPSERVER=1,80<CR><LF>
```

Response:

```
<CR><LF>
OK<CR><LF>
```

This sets up the shield to behave as a server and listen for incoming connections (in the above case, port 80). The server can be turned off by using 0 instead of 1, but the

shield should be restarted (eg by the AT+RST command) to make sure the all the settings are correct. Once the server is activated, you should expect incoming data from clients connecting, and this may occur even without you issuing a command.

AT+CIPSTO

Query:

```
AT+CIPSTO?<CR><LF>
```

Response:

```
+CIPSTO:180<CR><LF>
<CR><LF>
OK<CR><LF>
```

Command:

```
AT+CIPSTO=120<CR><LF>
```

Response:

```
<CR><LF>
OK<CR><LF>
```

This command sets the server timeout in seconds, which is the time before the server automatically disconnects from a connected client. This can be handy for cases where the sketch loses track of what clients have connected. If a timeout occurs, then a CLOSED response will be triggered.

Connection and Communication:

AT+CIPSTART

Command:

```
AT+CIPSTART="TCP","192.168.4.1",80<CR><LF>
AT+CIPSTART=0,"TCP","192.168.4.1",80<CR><LF>
```

Response:

```
CONNECT<CR><LF>
<CR><LF>
OK<CR><LF>
```

This is used by a client trying to connect to a server. There are different command structures depending on whether the AT+CIPMUX command has been set to 0 or 1. In the second example command above, we have specified we are using connection 0, connecting with the TCP protocol to address 192.168.4.1 on port 80 (this would typically be used by a web browser trying to access a webpage via http). Note that only clients can start a connection.

AT+CIPSEND

Command:

```
AT+CIPSEND=5<CR><LF>
AT+CIPSEND=0,5<CR><LF>
```

Response:

```
OK<CR><LF>
>
```

The AT+CIPSEND command is used to send data over a connection previously established by the AT+CIPSTART command. The first example above is used when CIPMUX=0 and the second is used when CIPMUX=1, and is using connection number 0. The shield will respond with the OK, followed by a '>' prompt, after which it expects (in this case) a further 5 characters, which will be sent when the required number of characters are received. This number can be anything up to 2048. At the receiving end, the data will appear as a +IPD response (note there is no CR/LF):

```
+IPD,0,5:12345
```

AT+CIPCLOSE

Command:

```
AT+CIPCLOSE=0<CR><LF>
```

Response:

```
0,CLOSED<CR><LF>
<CR><LF>
OK<CR><LF>
```

This is used to end a connection, and can be used at either the server or client end. In a typical web server request, the server would close the connection after sending the requested data. Alternatively, a client downloading emails from a POP server would close the connection after retrieving all the emails it wanted.

AT+CIPSTATUS

Command:

```
AT+CIPSTATUS<CR><LF>
```

Response:

```
STATUS:5<CR><LF>
+CIPSTATUS:0,"TCP","192.168.4.2",41466,1<CR><LF>
<CR><LF>
OK<CR><LF>
```

If you want to find out what is happening with the connections, then the AT+CIPSTATUS command returns useful information. The STATUS line can have the following values:

2: Has IP address

3: Connected

4: Not Connected

5: Connection Fail

The information from the +CIPSTATUS line is connection number, connection type, connected IP address, connected port (note that this is the port number on the client, which may be different to the port number on the server), and 1 at the end indicates the shield is server (this would be 0 if the shield is a client).

Responses without Commands:

The following are responses that may occur with a command being issued, for example, a client trying to connect or send data.

CONNECT

Data:

```
0,CONNECT<CR><LF>
```

A new client has connected on connection number 0.

CLOSED

Data:

```
0,CLOSED<CR><LF>
```

The client on connection 0 has closed the connection. Note that this is the same response as sending the AT+CIPCLOSE command as mentioned above.

IPD

Data:

```
+IPD,0,492:GET / HTTP/1.1<CR><LF>
Host: 192.168.4.1<CR><LF>
```

(and more data following…)

This data indicates that data has been received from a client. The first number (0 above) indicates the connection number, and the second number (492 above) is the number of bytes received, which will be the number of bytes after the ":" in the data.

The following data might also be received if the access point is turned on or off:

```
WIFI DISCONNECT<CR><LF>
WIFI CONNECTED<CR><LF>
WIFI GOT IP<CR><LF>
```