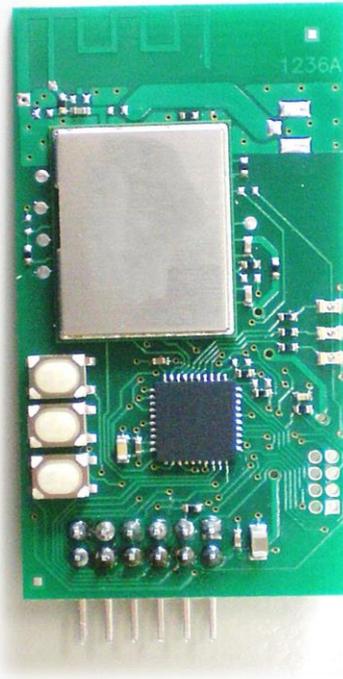


Aurel XTR-WiFi - User Guide



Aurel XTR-WiFi interface manages a Wi-Fi TCP/IP connection, with no need for complex stack operations by the user after an initial configuration. It automatically connects to a WiFi infrastructure (Access Point) with its own IP address, providing the native serial protocol (encapsulated in a TCP/IP packet) for remote application usage.

XTR-WiFi incorporates in a single interface all the functions normally required for a CPU host: TCP/IP stack, Wi-Fi client/server configuration, DHCP or static IP address assignment, network MAC address assignment.

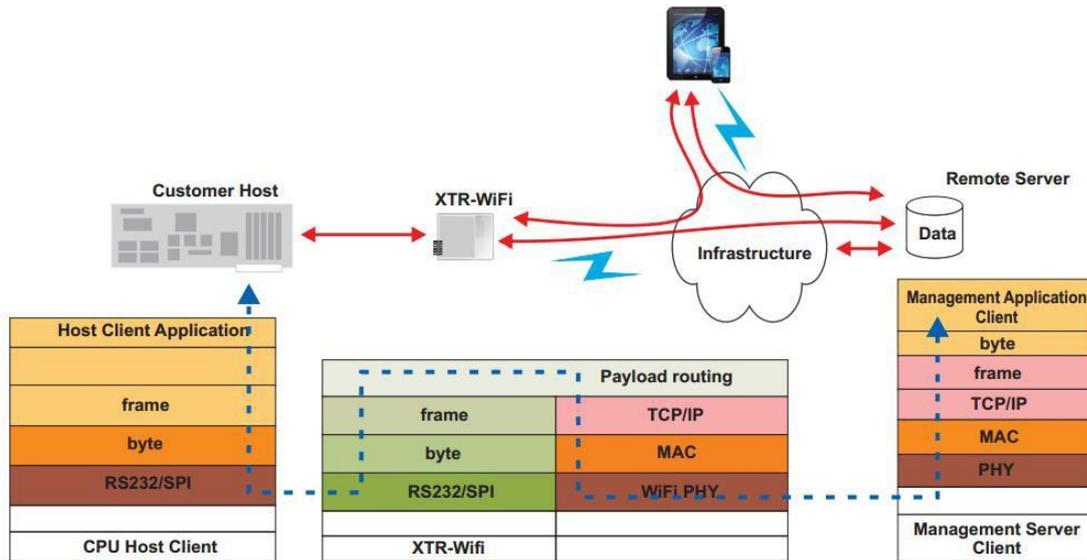
Features

- **WiFi Security** - WEP, WPA, WPA2 supported;
- **DHCP client** - module's IP set by DHCP server or static configuration;
- **DNS client** - IP address retrieving from symbolic name by DNS server;
- **PING** - Connectivity test towards other network nodes (ICMP protocol);
- **SCAN** - Specific command provided to scan for available Access Points;
- **Smart Config™** - Texas Instruments tool to configure modules via smart devices avoiding user commands;
- **TCP Server Mode** - Default configuration to listen for incoming connection, UART local notifications, transparent data transfer modality;
- **TCP Client Mode** - Remote server connection, transparent data transfer modality;
- **mDNS Advertiser** - Periodic multicast DNS packet for specific browsers.

Technical features are subject to change without notice. AUREL S.p.A. does not assume responsibilities for any damages caused by the device's misuse.

Overview

XTR-WiFi is an integrated module, compliant to IEEE 802.11b/g, supporting WLAN connection for infrastructured networks based on IPv4 TCP/IP protocol. A sample PC application



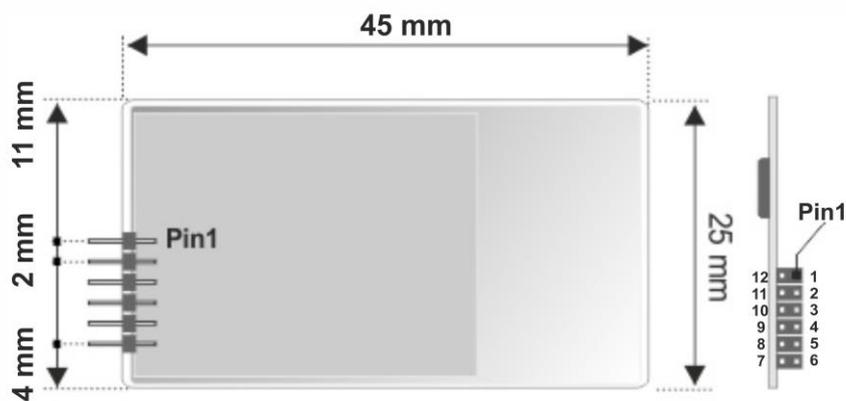
It can be configured as a client, in order to connect to a remote server, or as server, listening for incoming connection. At startup module is in command mode and through a simple AT command set the customer host can configure it and start the data exchange entering in data mode. Once connection is established, serial data (UART or SPI) are encapsulated in TCP/IP packets and are available for the remote host that can use them at application level, extracting them from TCP packets in a transparent way. An escape sequence is provided to close data connection and return to command mode.

A sample PC application has been developed in Visual Basic 2010 Express (.NET Framework 4.0) in order to provide a reference example for remote applications (client or server) to exchange data with XTR-WiFi. Whole project and source code are available for users and developers.

Suggested Applications

- ✓ WiFi connectivity for home appliances
- ✓ M2M communications for industrial monitoring and control
- ✓ Patient monitoring and medical instrumentation
- ✓ Home and building automation
- ✓ Energy Control

Mechanical Characteristics



XTR-WiFi is accessible by a 2mm-pitch strip. Strip can be provided straight or in 90° version, in order to meet mechanical requirements of the final application.

By default an integrated PIFA antenna is embedded in the module. If necessary, a UFL connector equipped module can be provided (it results in a different size), in order to use an external antenna. This solution should be used when the module is enclosed in a metallic case or whenever radio conditions do not allow an acceptable budget link.

Pinout

Pin	Name	Function	
1	GND	IN	Ground
2	SPI-IRQ	OUT	Used in SPI mode (active high) to indicate data ready from XTR-WiFi. If SPI is used add a pull-down to this net in order to avoid IRQ spurious activation.
3	SPI-MOSI/UART-TX	I/O	Tx output from XTR-WiFi if used by UART, SPI-MOSI input for XTR-WiFi if used by SPI.
4	SPI-CLK	IN	SPI clock.
5	GND	IN	Ground
6	Button 1	IN	Active low, internal pull-up. SmartConfig™ procedure start signal.
7	GPIO		Unused
8	UART/SPI*	IN	Selection between UART (logic level high) and SPI (logic level low) user serial interface. Value is read at startup, to switch interface real time XTR-WiFi should be reset. Pin is equipped with an internal pull-up to use module in UART mode by default.
9	SPI-MISO/UART-RX	I/O	Rx input for XTR-WiFi if used by UART, SPI-MOSI output for XTR-WiFi if used by SPI.
10	SPI-CS	IN	SPI Chip Select.
11	RST	IN	XTR-WiFi reset, active low, with internal pull-up. Connected to RST Button.
12	VCC	IN	Power supply (3.3V typical).

Technical features are subject to change without notice. AUREL S.p.A. does not assume responsibilities for any damages caused by the device's misuse.

Electrical Characteristics

	Min	Typical	Max	Unit	Note
Power Supply	2.7	3.3	3.6	V _{DC}	
Current Consumption	280		295	mA	802.11b - Tx
	112		123	mA	802.11b - Rx
	210		227	mA	802.11g - Tx
	112		123	mA	802.11g - Rx
Frequency	2.401		2.483	GHz	IEEE STD 802.11b ⁽¹⁾
RMS Output Power		17		dBm	
Temperature range	-20		70	°C	

NOTE 1: Standard 20MHz channel bandwidth

Device usage

In order to obtain performances described in the technical specifications and to comply with operating conditions which characterize the Certification, the device must be mounted on a printed circuit taking into account the following:

Voltage Supply

1. Module must be powered by a low voltage safety source, protected against short circuits.
Maximum allowed voltage variations is 2.7 ÷ 3.6V.
2. Provide voltage decoupling near the transmitter with a ceramic capacitor of at least 100 nF.

Ground

Ground must surround in the best possible way the transceiver welding area. The ground plane should be on the lower side and there must be no integrated circuits near the antenna in order to avoid coupling.

Other components

1. Keep the device away at least 5mm from all other components on the circuit
2. Keep microprocessors and their clock circuits particularly far away and shielded.

Reference Standards

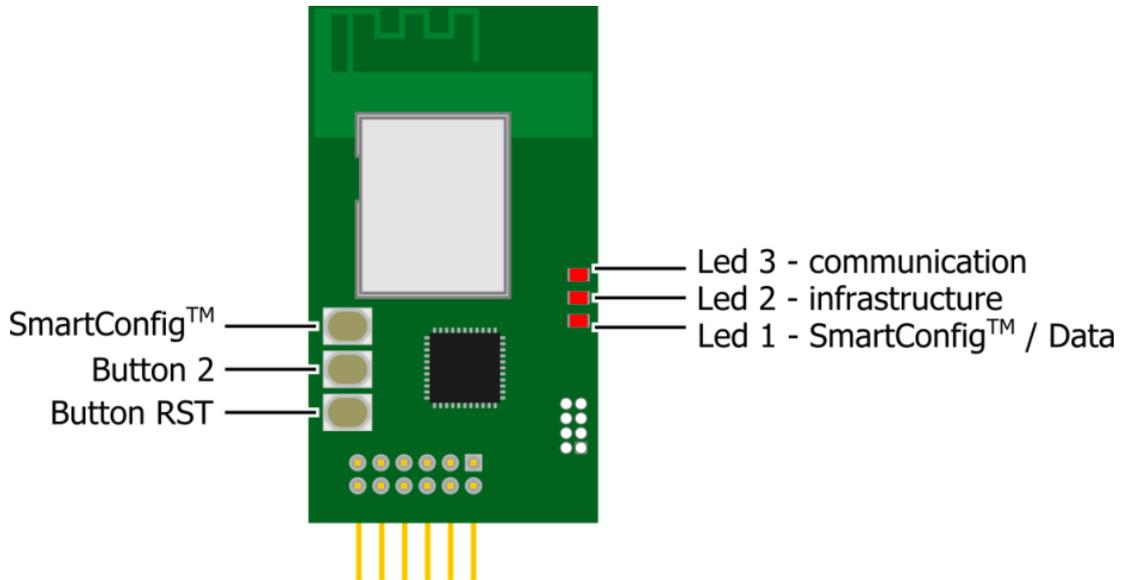
The XTR-WiFi modules comply with the European standards **EN 300-228** and **EN 301-489** with maximum power supply of 3.6V. The product has been tested according to EN 60950 and can be used inside a special housing that ensures compliance with the above mentioned regulations. The device must be powered by a low voltage safety source protected against short circuits.

The use of the module is foreseen inside housings that assure the overcoming of standards EN 61000 not directly applicable to the module itself. In particular, it is the user's care isolation antenna as the RF output of the transmitter is not able to support directly the electrostatic charges foreseen by the standard EN 61000-4-2.

Technical features are subject to change without notice. AUREL S.p.A. does not assume responsibilities for any damages caused by the device's misuse.

User interfaces

XTR-WiFi is equipped with buttons and LEDs on board that allow users to directly interact with the module.



Button	Function
SmartConfig™	SmartConfig™ procedure manual start (connected to pin 6 'Button 1').
Button 2	unused
Button RST	XTR-WiFi reset (connected to pin 11 'RST').

LED	Function
SmartConfig™ / Data	Blink during SmartConfig™ procedure and switch on when it's complete. Turn off when XTR-WiFi enters in data mode (module ready to exchange data with a remote node) and blink when data are effectively exchanged.
Infrastructure	Infrastructure ready: switched on at XTR-WiFi connected to an Access Point.
Communication	Communication established: switched on when XTR-WiFi has acquired an IP address.

At startup all LEDs make a check blink to indicate that the module is ready.

Serial interfaces

For user data transmit XTR-WiFi is equipped with both UART and SPI serial interfaces. The choice between the two is done by a dedicated pin (UART/SPI*). If logical level is high (or, due to the internal pull-up, pin is unconnected) UART is used, if logical level is low SPI is activated. UART/SPI* value is read at startup, so it is necessary to reset the module after a switch from using a serial interface to the other one.

UART settings

Baud rate	19200 bps
Data bits	8
Parity	None
Stop bits	1
Flow control	None

TTL levels for UART lines are required.

SPI features

In SPI mode XTR-WiFi act as a slave. Both cases with data available from master or form slave are explained.

➤ **Data available from XTR-WiFi (Slave).**

XTR-WiFi raises SPI-IRQ line (pin 2).

Master should activate SPI interface (signals SPI-CLK, pin 4, and SPI-CS, pin 10) and read all data from SPI-MISO line (pin 9, in this case output for XTR-WiFi) until SPI-IRQ is lowered. SPI-IRQ is lowered at the beginning of the second-last byte transmission, so pay attention to read the whole available packet. In this phase SPI-MOSI line is ignored.

In order to avoid spurious activation of SPI-IRQ signal, a pull-down resistor is recommended on this pin. Note that XTR-WiFi gets stuck until all available data from master are properly read.

➤ **Data available from Master.**

Master can write directly to the XTR-WiFi using SPI interface: driving properly SPI-CLK (pin 4), SPI-CS (pin 10) and SPI-MOSI (pin 3, in this case input for the XTR-WiFi) lines.

In this case SPI-MISO line is ignored.

In both cases data are supposed to be stable and should be sampled on the rising edge of SPI-CLK. The most significant bit of any byte is transmitted first.

TCP/IP basic concepts

Two logical steps are necessary to have a working device on a WiFi network:

1. **Connection** to the physical infrastructure: device should be recognized by the infrastructure manager, an Access Point in the case of WiFi, that sort data packets from each node of the network to the others through the physical transmission medium available (a 2.4GHz radio channel).
2. Acquisition of **communication** parameters: once the device is able to physically exchange data with other nodes in the network, it's necessary that it be identifiable by an address, in order to be reached from the other nodes of the same network. Identification parameters depends by the network protocol, in this case Internet Protocol is used, so each node is identified by a unique IP address.

Communication between two nodes of a WiFi network is done at the same way as computer networks, and is based on the **client - server model**. This is a structure in computing that partitions tasks between the provider of a resource, called server, and service requesters, called clients.

A TCP server serve one or more clients establishing a TCP connection between itself and the clients. For each side there is an endpoint of the communication flow that is called **network socket**. A server creates sockets that are in *listening state*, waiting for initiatives from client programs. Each socket is characterized by a unique combination of an IP address and a TCP port number. For each connection an unique *dedicated socket* is created but a server could also create several concurrently TCP sockets, it depends by available computing resources.

TCP protocol is connection oriented and Internet socket usage is usually based on the Berkeley sockets standard.

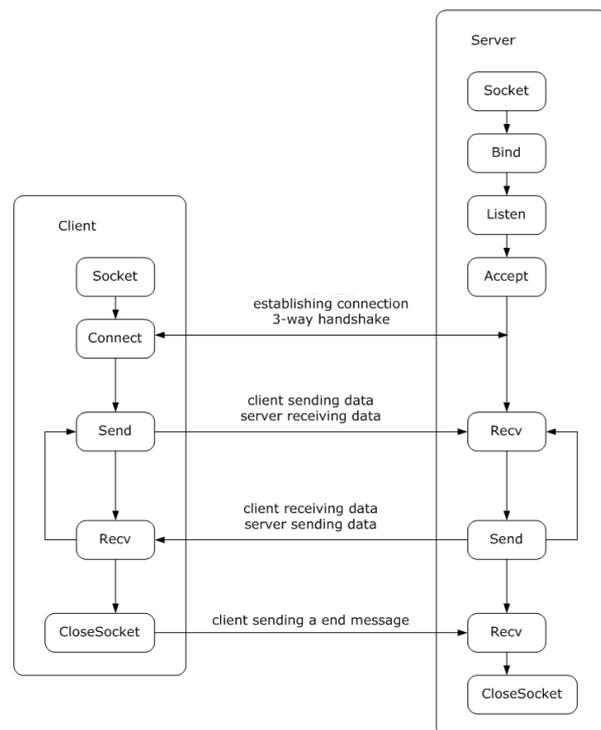
It assumes that servers are in listen state, ready to accept incoming connections.

A client make a request to the server to establish a connection, specifying address and port number of the server itself.

Server accepts the connection and authorizes client to send data; at this time client and server are connected and can transfer data each other.

If there are no more data to transmit the client call to release the connection, the server accept, and both client and server can free resources for reuse in further connections.

It's much important to release resources in a connection-oriented protocol so, in order to prevent failures on receiving release requests (i.e. if the transmission channel fall), usually a timeout (for each side) unilaterally close the communication after a given time from the last data exchanged.



Technical features are subject to change without notice. AUREL S.p.A. does not assume responsibilities for any damages caused by the device's misuse.

AT Commands

At start-up a welcome string is reported to customer host¹:

```
Aurel WiFi: Version 0.3-1.24CR
```

It indicates the firmware version currently loaded on the module (0.3-1.24 in this case) and that XTR-WiFi is on and ready to accept commands.

In fact, at startup XTR-WiFi is in command mode, so can accept commands through serial interface (the command set is the same using either UART or SPI interface).

Any command is parsed and interpreted by XTR-WiFi when a carriage return (0x13) is detected at the end of the string. When a command is sent to module without carriage return at the end, string is processed anyway after about 3 milliseconds from last received character.

If a command is unrecognized an appropriate message is returned.

```
AT+WRONGCOMMANDCR  
Invalid CommandCR
```

Basics

+++ (Escape Sequence)

To return to command mode after a data transfer it is necessary to send a "+++" sequence at the beginning of a new data packet.

```
+++CR  
Passing in command modeCR  
Connection waiting...CR
```

XTR-WiFi indicate the switch of modality and implicitly that previous active connection has been closed. Pay attention to avoid the "+++" sequence while exchanging application data because it involves at any time a connection closing.

AT

Sending a simple AT command (with no arguments) XTR-WiFi answers just to check UART communication.

```
ATCR  
OKCR
```

¹ For each example RealTerm serial terminal has been used [<http://realterm.sourceforge.net/>]

Communication

Communication parameters are hierarchically more important than connection ones, though they regards only the second stage of the network join. For this reason user must set communication parameters first than connection parameters, because when the Access Point accept the module to join the network, affiliation modality need to be already configured (the concept is valid also if DHCP is used).

```
AT+CONF=[IP_Address];[NetMask];[Gateway];<DNS_Server>
```

This command set statically IP parameters for Wi-Fi network identification and location addressing: Module IP Address, NetMask, Gateway and DNS server (each ';' is needed, <DNS_Server> parameter is optional).

```
AT+CONF=192.168.0.3;255.255.255.0;192.168.0.1;␣  
OK␣
```

IP_Address: Internet Protocol (local) network address of the device.

NetMask: bitmask that separate network identifier ('192.168.0' in the example) from host identifier ('3' in the example), used to discriminate if the message destination is a node of the local network or it must be externally reached through the gateway.

Gateway: IP address of the device to which forward messages that are destined out from the local network, as the result of the net mask process.

DNS_Server (optional): IP address (usually external to the local network) of the server that translates domain names to numerical IP addresses, needed for the purpose to locating services and devices worldwide.

AT+CONF command force the disconnection from current network in order to perform requested operation.

```
AT+CONF=192.168.0.3;255.255.255.0;192.168.0.1;192.168.0.1␣  
Disconnected from AP␣  
OK␣  
Connected to AP␣  
IP Acquired␣  
Connection waiting...␣
```

If the module was connected to an Access Point before receiving the command, it tries to reconnect to the same, but using new settings.

Be careful not to create conflicts allocating static IP Addresses (i.e. the same Address to multiple nodes), it can compromise network functionalities.

IP settings are also stored in non-volatile memory to be preserved after a reboot.

AT+DHCP

Instruct node to retrieve IP configuration from a DHCP server. DHCP is the default IP assignment mode for XTR-WiFi.

```
AT+DHCP\r\nOK\r\n
```

If the module was connected to an Access Point AT+DHCP command force disconnection from current network in order to perform the requested operation.

```
AT+DHCP\r\nDisconnected from AP\r\nOK\r\nConnected to AP\r\nIP Acquired\r\nConnection waiting...\r\n
```

If DHCP mode is chosen but no DHCP server is provided, XTR-WiFi can get stuck waiting for IP address assignment, pay attention to avoid this condition.

Infrastructure connection**AT+SCAN**

With this command is possible to force XTR-WiFi to perform a scan for available WiFi networks in the area.

```
AT+SCAN\r\nSSID: AUREL-WIFI - RSSI: -45dBm - Security: WPA2\r\nSSID: iPhone - RSSI: -64dBm - Security: WPA2\r\nSSID: AndroidAP - RSSI: -45dBm - Security: UNSEC\r\n
```

For each network SSID, RSSI value and security encryption type are reported. In case of no network is detected an appropriate message is produced.

```
AT+SCAN\r\nNo network in the area\r\n
```

Pay attention that all desired SSID name are visible, some Access Points can hide it in their settings.

Furthermore some N network mode Access Points sets by default channel bandwidth at 40MHz (wide mode), it's recommended to choose standard 20MHz channel bandwidth.

AT+CONN=[SecurityLevel];[Ssid];<SecurityKey>

With this command is possible to instruct XTR-WiFi to connect to an Access Point specifying its infrastructure parameters (each ';' is needed, <SecurityKey> is optional).

SecurityLevel:

- UNSEC - no security (open access point): no encryption
- WEP - Wired Equivalent Privacy: 64bit key encryption
- WPA - Wi-Fi Protected Access: dynamic 128bit key encryption
- WPA2 - Wi-Fi Protected Access II: 256bit key encryption

SSID: Service Set Identifier, commonly called the "network name".

SecurityKey: omitted for unsecured connections, for WEP connections key should be composed only of ASCII characters.

When the connection is established an apposite message is given from XTR-WiFi. Than the module start network join negotiation, obtaining an IP address from the Access Point according to the previous defined modality (static or DHCP assigned).

```
AT+CONN=WPA2;AUREL-WIFI;aurel-key␣  
Ok␣  
Connected to AP␣  
IP Acquired␣  
Connection waiting...␣
```

When XTR-WiFi is connected to an Access Point it automatically issue a TCP server listening on last used port (default port 4444 if not previously chosen a different one). With further commands is possible to make it work as a client or a server listening on a different user-defined port.

At the following startup XTR-WiFi will try to connect automatically to the same Access Point, remembering last connection parameters.

If XTR-WiFi is connected to an infrastructure that goes down a message is reported.

```
Disconnected from AP␣
```

AT+SMRT

XTR-WiFi is compliant with SmartConfig™ tool² by Texas Instruments, that is a one-time process to connect an embedded Wi-Fi device to a wireless network. It allows to configure a module using a smartphone, a tablet or a PC, with no need for most complex operations from customer host.

In details the application allows the user to connect XTR-WiFi to an existing Access Point passing to it the infrastructure parameters directly from a smart device, by-passing the AT+CONN serial command. The smart device must be connected to the same Access Point via WiFi, so SmartConfig™ application³ running on it can grab infrastructure parameters and transfer them to the module, previously initialized to accept data. SmartConfig™ App than show on the screen a confirm message in case of success.

Procedure:

1. Connect a smart device to the desired WiFi network.
2. Send a AT+SMRT command to XTR-WiFi (or push SmartConfig™ Button, or drive low BUTTON 1 pin).

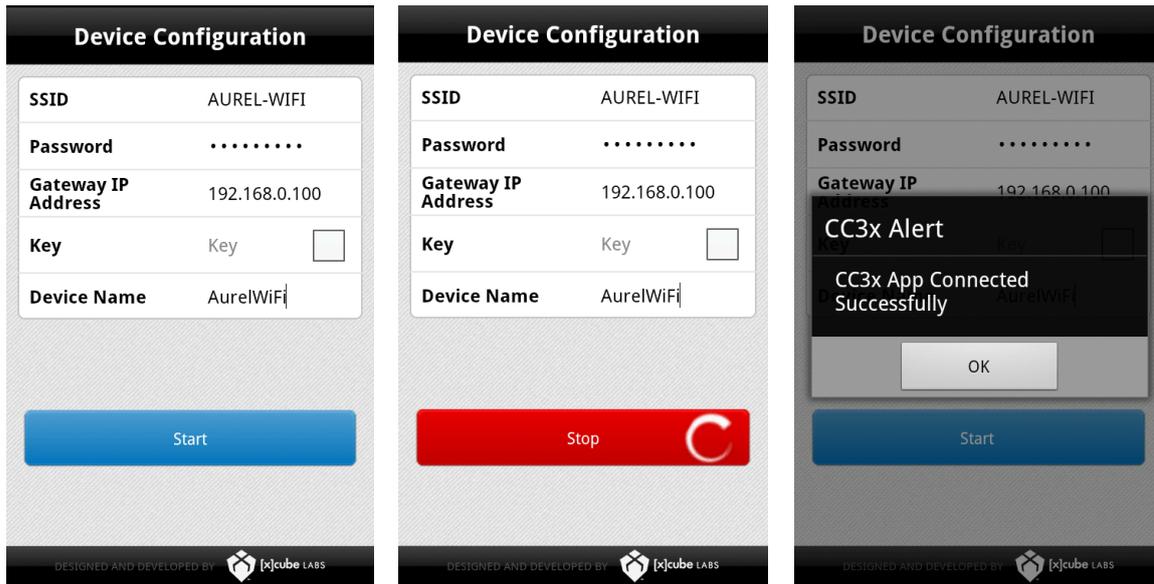
```
AT+SMRT Cr
Smart Config in progress... Cr
```

A message is produced to indicate that command has been received and procedure has begun.

3. Launch SmartConfig™ application in the smart device and configure it:
 - SSID** is automatically grabbed from the active Wi-Fi connection;
 - if encryption is used, type the network key in **Password** field;
 - Gateway IP Address** is automatically grabbed from the active Wi-Fi connection;
 - Key** checkbox must be unchecked;
 - Device Name** has to match the default mDNS name given by the module at the end of SmartConfig™ procedure: type "AurelWiFi".
4. press **Start** button.

² http://processors.wiki.ti.com/index.php/CC3000_Smart_Config

³ SmartConfig™ for iOS, Android and PC can be downloaded from <http://www.ti.com/tool/smartconfig>



After few seconds, if procedure complete successfully, the module is connected to the Access Point and automatically issue a TCP server listening on last used port (default port 4444 if not previously chosen a different one).

Subsequently with further commands is possible to make it work as a client or as a server listening on a different user-defined port.

```

AT+SMRT␣
Smart Config in progress...␣
Smart Config Done␣
Connected to AP␣
IP Acquired␣
Connection waiting...␣
    
```

During SmartConfig™ procedure no command can be received, it does not have an expiration time-out, so to break it it's necessary to reset the module (pin 11 'RST', or Button RST) or turn it off.

If the module was previously connected to an Access Point, during the SmartConfig™ procedure some error messages can occur (due to the fact that in background it try to join with the known AP) but normally it does not affect the successfully procedure ending, signaled reported messages.

Note: pay attention using DHCP or static IP functionality according to network capabilities, also if SmartConfig™ is used. During SmartConfig™ procedure the security key is passed unencrypted from smart device to XTR-WiFi module.

At the following startup XTR-WiFi will try to connect automatically to the same Access Point, remembering last connection parameters.

AT+DSCN

XTR-WiFi disconnects from the current Access Point.

```
AT+DSCN\r\nOK\r\nDisconnected from AP\r\n
```

In case it was already disconnected an apposite message is produced.

```
AT+DSCN\r\nAlready Disconnected\r\n
```

Diagnostic**AT+IPC**

Interrogate XTR-WiFi to read all current Wi-Fi connection parameters (module IP Address, Gateway, current SSID and MAC address).

```
AT+IPC\r\nIP: 192.168.0.106\r\nGateway: 192.168.0.100\r\nSSID: AUREL-WIFI - MAC: 08:00:28:01:A2:2F\r\n
```

Informations are correctly retrieved if the module is connected to an Access Point, otherwise various '0' fields are reported.

```
AT+IPC\r\nIP: 0.0.0.0\r\nGateway: 0.0.0.0\r\nSSID: AUREL-WIFI - MAC: 00:00:00:00:00:00\r\n
```

In this case SSID reported is the last used by the module.

AT+PING=[IP Address]

Ping (ICMP protocol) a remote host to test network connectivity. Four 56-byte packets are sent to the remote host and afterwards a report is produced.

```
AT+PING=192.168.0.9␣  
Pinging...␣  
Packets - Sent: 4, Received: 2␣  
Delay - Min: 10 ms, Max: 11 ms, Avg: 10 ms␣
```

In case a DNS server is available, it's possible to specify a symbolic name instead of the explicit IP address. DNS server's IP is the last parameter given in a manual specified configuration (command AT+CONF), otherwise is automatically returned by the Access Point at the connection if DHCP modality is chosen.

```
AT+PING=Lightnetc␣  
Pinging 192.168.43.229␣  
Pinging...␣  
Packets - Sent: 4, Received: 4␣  
Delay - Min: 5 ms, Max: 230 ms, Avg: 60 ms␣
```

A local DNS server resolve symbolic name replacing it with the correspondent device IP address. If XTR-WiFi has access to the Internet, a public DNS server (like OpenDNS, 208.67.222.222) resolve also IP addresses of domain name resources.

```
AT+PING=google.com␣  
Pinging 173.194.35.4␣  
Pinging...␣  
Packets - Sent: 4, Received: 4␣  
Delay - Min: 367 ms, Max: 628 ms, Avg: 437 ms␣
```

If DNS server is not properly configured, or is not available, a specific error code is returned.

```
AT+PING=Lightnetc␣  
Error - Code 11␣
```

AT+MDNS=MyDevice

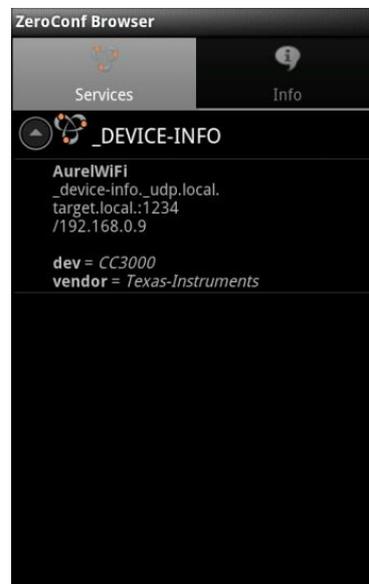
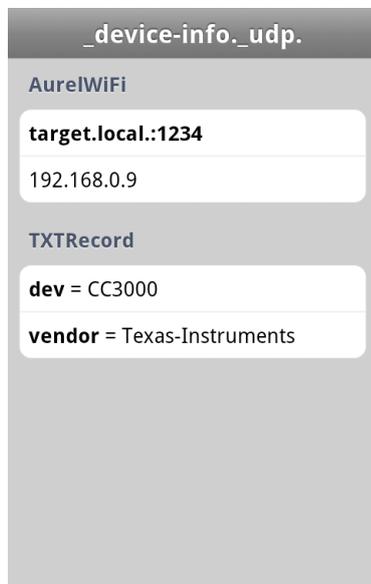
mDNS (multicast DNS⁴) is a host name resolution service that uses same semantics as DNS within networks that do not include a local name server.

AT+MDNS command set a custom DNS name for XTR-WiFi and broadcast a relative mDNS packet on the network so, as happens for DNS, is possible to associate a specified symbolic name to the IP address.

```
AT+MDNS=XTR-WiFi␣
OK␣
```

mDNS protocol is implemented by Apple Bonjour and Linux nss-mdns services and informations can be retrieved using apposite mDNS browsers, for example: 'Discovery' or 'mDNS Watch' for iOS and 'Bonjour Browser' or 'ZeroConf Browser' (both exemplad) for Android.

Anyway, if the module is connected to an Access Point, about every 10 seconds an 'AurelWiFi' default mDNS message is automatically broadcasted through the network, to inform remote nodes about the presence of an XTR-WiFi module.



If the module is not connected to an Access Point is not possible to send mDNS informations and an apposite message is returned.

```
AT+DSCN␣
OK␣
Disconnected from AP␣
AT+MDNS=XTR-WiFi␣
Error - Code 10␣
```

At the end of the SmartConfigTM procedure is precisely a mDNS packet with the default name 'AurelWiFi' that notify to the App that procedure has been successfully completed.

⁴ http://en.wikipedia.org/wiki/Multicast_DNS

Data transmission

```
AT+CLNT=[IP Address];[Port Number]
```

Connect XTR-WiFi as a client to a remote socket indicated by server IP address and port number.

```
AT+CLNT=192.168.0.9;1910␣  
Connected to server - Passing in data mode␣
```

If connection is successful XTR-WiFi passes to data mode, otherwise an error message is reported.

```
AT+CLNT=192.168.0.9;1910␣  
Error - Code 9␣
```

Errors are reported within 60 seconds from the connection attempt. Pay attention that remote server is not protected by a firewall to avoid systematic connection failures.

Similarly to AT+PING command, if a DNS server is correctly configured it is possible to specify a symbolic name instead of the explicit IP address.

```
AT+CLNT=Lightnetc;1989␣  
Connecting to 192.168.43.229␣  
Connected to server - Passing in data mode␣
```

DNS server resolve symbolic name replacing it with the correspondent device IP address and show it.

If XTR-WiFi is in data mode up to 61 byte at a time are sent to remote node, no special characters are used apart the "+++" sequence that switch to command mode, pay attention to avoid this sequence while exchange application data.

After about 60 seconds from the last transmitted data, connection automatically shut down to free resources and the module return to command mode. If remote server is a XTR-WiFi it also automatically shut down the connection after 50 seconds from last transmitted data, the connected client node can detect this trying to send data, and then returning a message error.

```
[user data]␣  
[user data]␣  
[user data]␣  
Error - Code 6␣  
Passing in command mode␣  
Connection waiting...␣
```

When XTR-WiFi returns to command mode, by default automatically issue a TCP server listening on last used port as server (default port 4444 if not previously chosen a different one).

AT+SRVR=[Port Number]

Issue a TCP server on a specified port, than XTR-WiFi starts listening for a client connection request.

```
AT+SRVR=1910␣  
Connection waiting...␣  
Starting data mode - Accepting connection from: 192.168.0.106␣
```

When an incoming connection is received the module returns the IP address of the requesting client node and then switch to data mode.

Data received from remote nodes are redirected to serial interface in transparent mode and become available for customer host.

To return to command mode (thus closing active connection) it is necessary to send a "+++" sequence at the beginning of a new data packet: pay attention to avoid this sequence while exchange application data.

```
+++␣  
Passing in command mode␣  
Connection waiting...␣
```

Returning in command mode, or also connecting the module to another Access Point, XTR-WiFi issue by default a server listening on last used port (default port 4444 if not previously chosen a different one).

Therefore, after about 50 seconds from last transmitted data connection automatically shut down (to free resources) and the module returns to command mode.

```
+++␣  
Passing in command mode␣  
Connection waiting...␣  
Starting data mode - Accepting connection from: 192.168.43.67␣  
[user data]␣  
[user data]␣  
[user data]␣  
[user data]␣  
[user data]␣  
Passing in command mode␣  
Connection waiting...␣
```

To start a server on a different port is first necessary to stop the current issued server (AT+STOP command) ad then start a new one. At startup port number returns the default 4444.

AT+STOP

Stop server listening. If a connection from a client was established the module is in data mode, so it's necessary to send a "+++" sequence before.

```
+++  
Passing in command mode  
Connection waiting...  
AT+STOP  
OK
```

If no server is currently running an apposite message is returned.

```
AT+STOP  
No Server Listening
```

Quick start

A simple sequence of commands is suggested for a first approach to the module.

Apart from basic and diagnostic commands, for the core functionality there are 2 ways to do the same things: a 'professional' one, in which user must know WiFi parameters to configure the connection, and a 'smart' one, in which some kind of automatic configuration does the job instead of the user.

AT		check UART communication
AT+SCAN		check if desired WiFi network is available in the area
PRO approach		SMART approach
AT+CONF	set manually IP parameters	AT+DHCP (default setting) retrieve IP configuration automatically from the AP
AT+CONN	connect module to a specified Access Point	AT+SMRT connect module through SmartConfig™ procedure
AT+SRVR	use module as a server	AT+CLNT connect module as a client

Error Table

If a command return an error it is followed by an exit value with its explicit meaning:

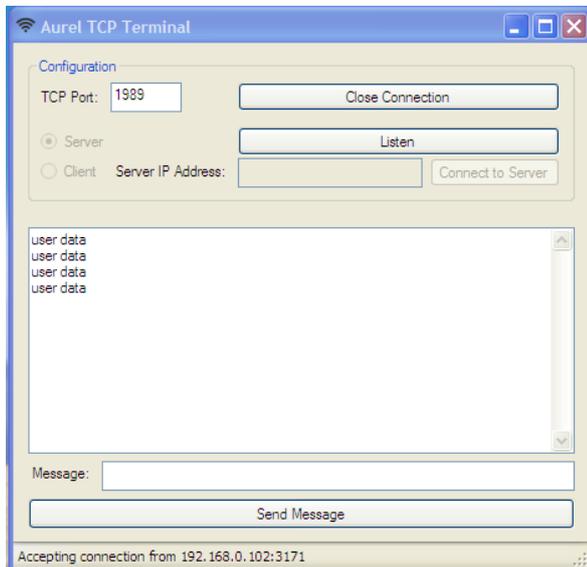
Error Code	Description
Error 1	Syntax Error
Error 2	Error in connection to the Access Point
Error 3	Error opening TCP socket
Error 4	Error binding TCP socket
Error 5	Error while listening for incoming connections
Error 6	Error sending data
Error 7	Error closing socket
Error 8	Invalid number of arguments
Error 9	Error connecting to remote server
Error 10	Error during mDNS
Error 11	Error interrogating DNS server, while resolving a symbolic name
Error 12	Error while executing ping request

Technical features are subject to change without notice. AUREL S.p.A. does not assume responsibilities for any damages caused by the device's misuse.

Aurel TCP Terminal

In order to support developers, a sample application is provided⁵. It has been developed in Visual Basic 2010 Express⁶ (.NET Framework 4.0) and whole project and source code are available for users.

It implements a simple TCP socket which can be used as a client or as a server, and allow to set connection parameters.



Server mode

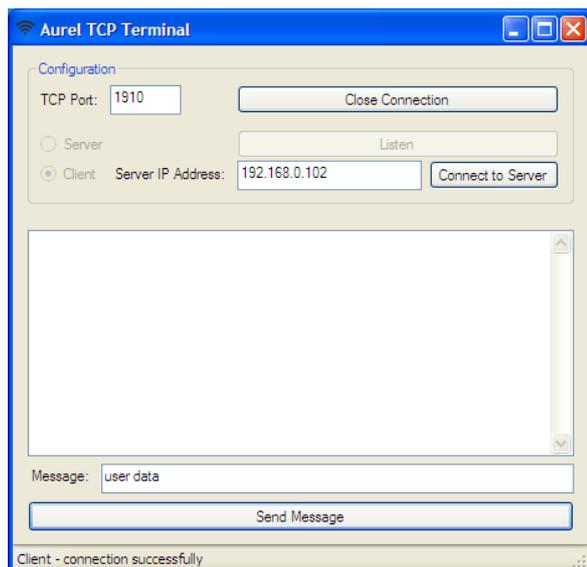
- IP Address: given by PC
- port number: should be specified.

Listen start server listening.

Remote client request acceptance is notified in the status bar.

Send Message send data to remote node

Close Connection close the socket



Client mode

- IP Address: should be specified
- Port number: should be specified

Connect to Server let application try to connect to specified remote server.

A successful connection is notified in the status bar.

Send Message send data to remote node

Close Connection close the socket

When connection is established it is possible to send ascii characters to remote node (e.g. XTR-WiFi) and view received data in the large textbox.

PC firewalls can block TCP Terminal's socket communication, be sure to unblock it.

⁵ Project and source code files are downloadable from <http://www.aurelwireless.com/en/wireless-news/wi-fi-interface-rf-module/>, the executable file is "AUREL_TCP_Terminal/0.2/bin/Debug/AUREL_TCP_Terminal.exe"

⁶ <http://www.visualstudio.com/downloads/download-visual-studio-vs>