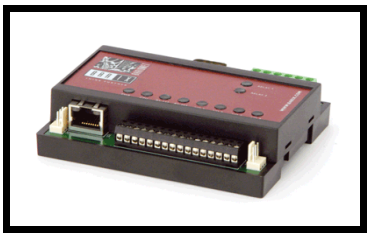
BARIONET

# BARIONET

**Universal network-enabled automation interface in commercial control, monitoring and home automation applications**



## User Manual

**Firmware** V2.13
**Released** 21. Nov. 2008
**Supports:**

- BARIONET (LX XPort)
- BARIONET (EX XPort)

BARIX

# Table of Contents

# 1 Introduction

## 1.1 About the Barionet

Barionet is a fully programmable network controller for interfacing devices to open, IP-based networks. With the Barionet, any device can be network-enabled and easily controlled via web browser and standards-based automation systems.

Barionet is a modular component that operates either stand-alone or with other units, web servers and control systems, thus capable of building large, intelligent, distributed systems for a wide variety of commercial control, monitoring, and home automation applications. SNMP V1.1 is supported, a private MIB is included.

Numerous inputs and outputs make the device ideal for any kind of interfacing application. Barionet features twelve I/O pins, which offer digital and analog input and digital output capabilities; two relays, which can be used to activate bells, door strikes, motors, blinds and lights; and two expansion ports, which offer connectivity to Dallas 1-Wire fi devices, such as temperature sensors, and Barix extension modules. Two integrated bus connectors and the serial interfaces allow for the connection of a wide range of additional I/O modules and smart sensors from various manufacturers. The serial ports can also be used independently for tunnelling applications.

The unit is programmable for stand-alone or connected operation in a powerful control language called BCL, enhanced with advanced functions for network communications. The language syntax is very similar to the well known Basic language, so most programmers can instantly craft interface solutions. The built-in web server can serve user specific pages, which can contain data generated by the control program or the various inputs.

## 1.2 Features

- Monitoring, controlling, and interfacing devices to IP-based networks or central management systems in commercial, industrial, and home applications
- Access control, machine date collection, and environmental monitoring
- Measuring voltages
- Monitoring contact closures
- Measuring Temperatures
- 10/100 Mbit Ethernet connection supports automatic network configuration (BOOTP, DHCP, AutoIP and IPzator) as well as manual static IP configuration
- Control and configuration using a standard web browser
- Remote monitoring using SNMP
- Remote controllable using HTTP, TCP and UDP
- Supports Serial Port relaying (Serial gateway over Network)

## 1.3 Installing the device

For the installation of the Barix Barionet please refer to the corresponding Quick Install Guide .
A printed version is included in the box and can also be downloaded from our site www.barix.com.

## 1.4 Additional documents

Technical specifications can be found in the corresponding product sheet which can be downloaded from our site www.barix.com.

For detailed technical information about the programming language please download the Barix Control Language (BCL) Programmers Manual from our website.

## 1.5 Preloaded Firmware

Barix preloads all Barionet devices with the Barionet firmware including a BCL sample application named Digital IO Tunnel .

Before continuing with this manual make sure that the firmware version is up to date and corresponds with this manual. Please proceed to chapter 8 Updating the device in order to do so.

## 1.6 About this manual

### Links to chapters

References to chapters (e.g. X Chapter name) are red and underlined and serve as direct links when viewed in Adobe Acrobat Viewer. Click on the link to jump to the referenced chapter, click on the left arrow icon to jump back to where you came from.

### Links to the Dictionary

Some technical terms (e.g. DHCP) are underlined and red. Click on them to jump to the dictionary at the end of this manual, click on the left arrow icon to jump back.

### Bookmarks pane in Adobe Acrobat

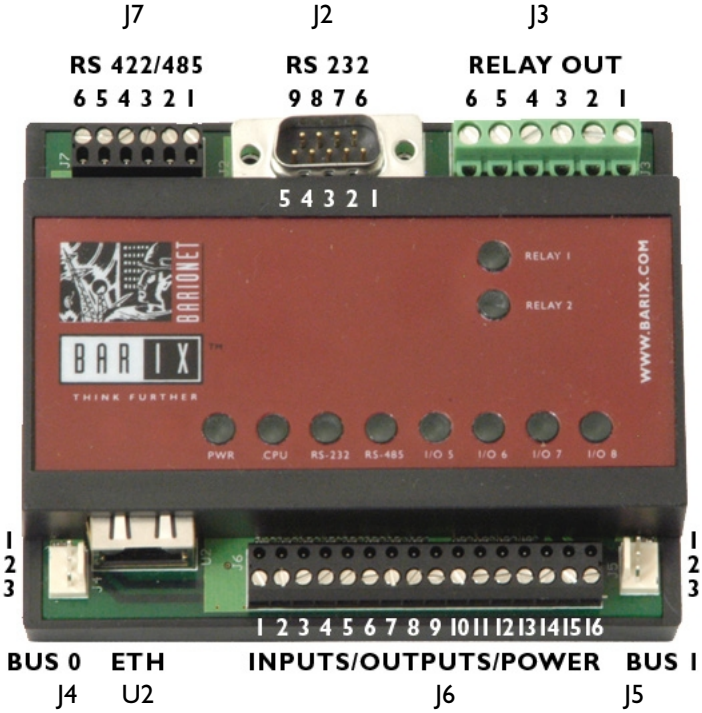
The complete Table of Contents is available in Adobe Acrobat Viewer. Click on the Bookmarks pane tab on the left side of Adobe Acrobat Viewer to open it. Click on any bookmark to directly jump to the corresponding part of the manual.

## Chapter overview

This manual is divided into the following chapters:

- Barionet Hardware explaining all interfaces, LED's and how to attach external devices
- 1st Time Browser Access explaining how to access the device without knowing the IP address
- Device Status and Control explaining status display and control using a browser
- Device Configuration explaining all configuration parameters
- Reverting to factory defaults explaining how to revert configuration parameters using a browser
- Rebooting the device explaining how to reboot the device using a browser
- Updating the device explaining how to update the device using a browser as well as how to rescue a unresponsive device using a serial cable
- BCL Application Digital I/O Tunnel explaining a sample BCL application
- HTTP interface (CGI API / DHTML) explaining how to control the Barionet via internal & external web pages
- ASCII interface (TCP/UDP API) explaining how to control the Barionet via the API Interfaces
- I/O Address Mapping listing all Input/Output addresses
- Configuration Setup Record explaining and listing all used and reserved Setup addresses
- Dictionary explaining technical names and expressions used in this manual
- IP Address, Netmask etc. providing fundamental network knowledge
- BIN / DEC / HEX conversion showing a look up table for quick conversion

# 2 Barionet Hardware




## 2.1 Case and Lids

The Barix Barionet is a rail mountable device, which snaps onto a 55mm metal rail (DIN-rail) widely used in industrial automation as well as electrical installations. In addition, two screw holes in the case allow for direct wall mounting the case if a rail mount is not possible.

The Barix Barionet comes with two lids, which can be exchanged with a simple snap mechanism. The smaller, low profile lid is intended for installations where a certain height is not needed. The larger high profile lid fits well for installations in electrical cabinets, where the front of the Barionet is then elevated to the same height

as a typical European circuit breaker and thus viewable at the front plate.
An optional metal DIN-rail (4 1/8 , 105mm) with two screw holes for wall mounting can be ordered separately. This gives the user the advantage of an easy and quick snap on but still durable wall mounting.

## 2.2 J7 RS-422 / RS-485 Serial Interface

This connector is a removable screw block, and it carries the RS485 (2 wire) and RS-422 (4 wire) signals, as well as a connection for reference ground. All I/O pins are ESD protected. For RS-485, the TX pair (pins 5 and 6) are used for both transmit and receive function.

The Baud rate is software selectable from 300-19200 baud, 1 or 2 stop bits, data format 8 bit, no, odd or even parity. A blinking RS-485 LED shows activity. If the LED stays lit then most probably the A/B wires are swapped.

| Pin | RS-422 | RS-485 |
|---|---|---|
| 1 | Shield * | Shield * |
| 2 | RXA | Not used |
| 3 | RXB | Not used |
| 4 | TXA | A |
| 5 | TXB | B |
| 6 | Shield * | Shield * |

*** Attention:**
Pins 1 and 6 are connected to the Ground of the Barionet via a 100 Ohm resistor ! Make sure that attached devices are powered by a DC power supply.

The Barionet acts on the RS-422/485 interface as a Master and accepts only Slave devices like the Barix IO12, the Barix R6 and the Barix X8 as well as 3rd party Slave devices (e.g. Modbus-Slaves). Therefore do not connect another Master device (e.g. two Barionets) !

## 2.3 J2 RS-232 Serial Interface

Configuration of this connector (9 pin sub D, male, DTE) is similar to a PC, however, only 4 modem lines are supported, so the interface is a 5-wire version only.

The Baud rate is software selectable from 300-19200 baud, 1 or 2 stop bits, data format 8 bit, no parity and with or without hardware flow control (RTS/CTS).
A blinking RS-232 LED shows activity.

| Pin | Direction | Function |
|---|---|---|
| 1 | | NC |
| 2 | to Barionet | RxD Receive data |
| 3 | from Barionet | TxD Transmit data |
| 4 | from Barionet | NC or unregulated Power (VCC) when L2 closed (see warning !) |
| 5 | from Barionet | Ground |
| 6 | | NC |
| 7 | from Barionet | RTS Ready to send |
| 8 | to Barionet | CTS Clear to send |
| 9 | | NC |

On pin 4 of the interface, the raw, unregulated supply voltage can be supplied when a bridge over L2 contacts is soldered by the user allowing for devices like scanners, terminals etc. to be supplied through this connector.

The L2 contacts are located next to C31 , left of pin 16 of the SIPEX serial driver chip U8 .

**WARNING:**
Operating the Barix Barionet with a voltage higher than 15 VDC when the L2 bridge is soldered and a device is attached to the RS-232 port is potentially dangerous as the supply pin is shared with a modem signal and the RS-232 specification allows a maximum voltage of 15V on inputs and outputs!

## 2.4 J3 Relay Connector

Each of the two relays provide separate Normally Open (NO) and Normally Closed (NC) capability, and can switch 5 Amps at 250 VAC max.

The contacts are provided on the top right fixed screw terminal block. The indicator LEDs (labelled Relay 1 and Relay 2) are activated together with the relay.

When the relay is inactive (LED is off), it connects between the NC and the REL screw terminals, when it is active, it connects the NO and REL terminals.

| Pin | Name | Function |
|---|---|---|
| 1 | REL 2 | Relay 2 common contact |
| 2 | NO 2 | Relay 2 Normally Open contact |
| 3 | NC 2 | Relay 2 Normally Closed contact |
| 4 | REL 1 | Relay 1 common contact |
| 5 | NO 1 | Relay 1 Normally Open contact |
| 6 | NC 1 | Relay 1 Normally Closed contact |

## 2.5 J4 Expansion Port

The J4 expansion port connector located at the left side is used for the serial rescue procedure (update over serial cable) which needs a jumper to be set across pin 2 and 3. The connector carries a 5V bus data signal, the unregulated supply voltage and ground.

| Pin | Name | Function |
|---|---|---|
| 1 | Vcc (+9..30 VDC) | Power Supply |
| 2 | Vss (Ground) | Supply / Rescue Jumper |
| 3 | BUS 0 (5VDC) | Rescue Jumper |

BUS 0

## 2.6 J5 Expansion Port

The J5 expansion port connector located at the right side is intended to connect up to 50 DALLAS 1-wire temperature sensors. The connectors carries a 5V bus data signal, the unregulated supply voltage and ground.

| Pin | Direction | Function |
|---|---|---|
| 1 | Vcc (+9..30 VDC) | Power Supply |
| 2 | Vss (Ground) | Supply / 1-wire ground |
| 3 | BUS 1 (5VDC) | 1-wire supply and data |

BUS 1

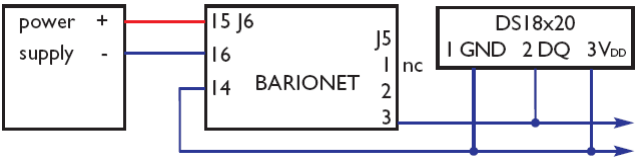
## Connecting 1-wire Temperature Sensors

To connect Dallas 1-Wire fi temperature sensors (DS1820, 18B20 and18S20) two different ways are possible (in both the power is being received from the data line voltage on pin 2):

1.) Use the included extension cable (2 x 3 pin connectors, red, black and white cable) and connect 2 wires that will serve as a bus cable and bridge the power ground with the digital output ground but loosing the common mode coil power filter.


2.) Use the included extension cable (2 x 3 pin connectors, red, black and white cable) and connect 1 wire to pin 3 (white cable) and the other cable to the digital output ground.

This way is recommended by Barix for being more reliable and more resistant to electromagnetic disturbances.




To distinguish which sensors is represented by which ID on the UI or by which I/O address we recommend the proven method of using a cooling spray on each sensor until you see the according value dropping.

### Barix TS Temperature Sensor

Barix offers a rubber encapsulated digital 1-wire temperature sensor (DS18B20) under the product name BARIX TS TEMPERATURE SENSOR.

Features:

- digital temperature sensor with 12 bit resolution
- Wide range: 55 C to +125 C ( 67 F to +257 F)
- 0.5C accuracy from 10 C to +85 C
- Rubber encapsulated cable (1 ft. / 33 cm) and housing with mounting hole

This sensors cable has a green wire (1 GND) and green wire with a white strip (2 DQ) which can be attached to the Barionet as explained in the previous section.

## 2.7 U2 Ethernet Network Interface

The Barix Barionet is equipped with a standard Ethernet 10/100MBit, full / half duplex, auto negotiation, interface.

Power over Ethernet (PoE) is not implemented but can be achieved with an external adapter that splits the Power from the Ethernet signal.

The two LEDs on the RJ-45 connector display the network status as explained in the following table.

| Left | Right | Function |
|---|---|---|
| Orange | | Link OK 10 Mbit LAN speed |
| Green | | Link OK 100 Mbit LAN speed |
| | Orange | Blinking = Traffic 10 Mbit |
| | Green | Blinking = Traffic 100 Mbit |

The older Barionets (produced 2005 and earlier) came with a Lantronix LX-XPort while newer Barionets (2006 and later) have the Lantronix EX-XPort!

If you don·t know which one is built in please remove the Barionet·s top plastic cover and check the product number on the XPort/Network connector.

For the new XPort product number : XP1001001-03R see the above LED status table.

For the older XPort product number : XP1001000-01 see the table below.

| Left | Right | Function |
|---|---|---|
| Orange | | Link OK 10 Mbit LAN speed, blinking = Traffic |
| | Green | Link OK 100 Mbit LAN speed, blinking = Traffic |

## Hardware MAC Address, Revision and Production Date

The MAC address of the Barionet unit is printed on a label placed on the bottom of the device. In addition to this the hardware revision can be found on the bottom left of the sticker (e.g. HW:2.4) and the production date (e.g. PW:09.07 for week 9 in 2007) on the right side.

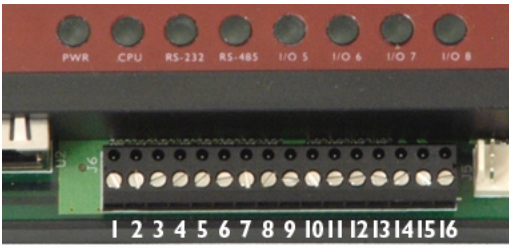
There are two additional MAC address stickers included in the product box which also has a sticker on the bottom. One sticker can be attached on the front of the device and one can be used for documentation purposes.

## 2.8 J6 Inputs / Outputs / Power

The connector row at the bottom of the Barix Barionet provides RIA removable screw block for the analog (A) and digital (D) inputs and outputs. DC power to the Barix Barionet is supplied using pin 15 and pin 16. The maximum voltage is 30VDC or 24VAC. Higher voltages will damage the unit.

**Attention:**
An AC power supply can be used if the Barionet is used stand alone. Use a DC power supply instead if you connect other Barix devices (IO12, R6, X8) over the RS-422/485 interface to the Barionet.

| Pin | Function | Pin | Function |
|---|---|---|---|
| 1 | Input 1 (A/D) | 9 | Ground for Inputs |
| 2 | Input 2 (A/D) | 10 | Output 1 (D) |
| 3 | Input 3 (A/D) | 11 | Output 2 (D) |
| 4 | Input 4 (A/D) | 12 | Output 3 (D) |
| 5 | Input 5 (D) | 13 | Output 4 (D) |
| 6 | Input 6 (D) | 14 | Ground for Outputs |
| 7 | Input 7 (D) | 15 | Vcc (+9..30VDC) |
| 8 | Input 8 (D) | 16 | Vss Ground |



### J6 1..8 Digital Inputs

Eight digital inputs are available on the Barix Barionet. The leftmost four inputs (1..4) can be used as either analog, supervised or standard digital inputs. The right four inputs (5..8) can only be used as digital inputs and the state is shown by LEDs I/O 5 through I/O 8 .

All eight inputs feature a programmable pull-up (default configuration: ON) which sources current so that the inputs can be directly connected to a contact.

If changed to OFF the input can be connected to a digital signal with max. 24 VDC high -level.

Please note that some input current (up to 2mA) might be drawn from the signal if the high level voltage is >5 V.

### J6 1..4 Analog Inputs

The leftmost four inputs can be used to measure voltages from 0 to 5 VDC . Higher voltages can be measured using an external resistor network (see next section).

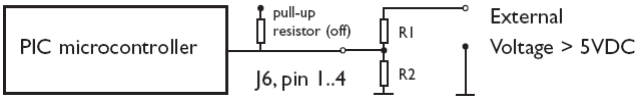
The resolution of the internal A/D converter is 10 bit, however, with the input protection circuits and series resistor, an accuracy of better than 9 bit is only achievable under ideal conditions (good ground connection, low impedance source).

**Attention**:
Make sure that none of the attached input signals (1..8) carry a voltage greater then 5 Volts if you want to measure analog values (Inputs 1..4) as the above mentioned input current (see Digital Inputs) will lead to wrong analog values.

The simplest way to achieve this when applying higher voltages is to use a resistor before the input and a so called Zener diode across the input (to ground) that limits the voltage to 4.7 Volts (commonly used value).

## Analog Input Voltage Divider

Using a voltage divider the measuring range can be increased. The resistor network should be small enough (R1+R2<10KOhm).




Example:

Ext. Voltage=24VDC max, V Ratio=24V / 5V = 4.8

Using a 3K9 for R1 and 1K for R2 brings us close:
4K9 / 1K = 4.9

This way we measure 4.899V (24V / 4.9 ) for the 24VDC.

The built in correction in the analog display tag in DHTML allows to still show the value correctly by applying a correction factor to the result.

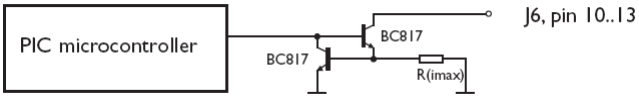
## Supervised Inputs

In conjunction with the internal, programmable pull-up resistors, the analog inputs can be used to form supervised digital inputs, which can detect both the break of a line as well as the opening of a contact.

Depending on the resistor values used the according logic can be programmed in BCL by the user.

## J6 10..13 Digital Outputs

The four digital outputs are implemented as open collector outputs. They can switch up to 24V, and can sink up to 100mA. They are suitable for driving LEDs, smaller signaling lamps, and relays up to 24VDC / 100mA types.

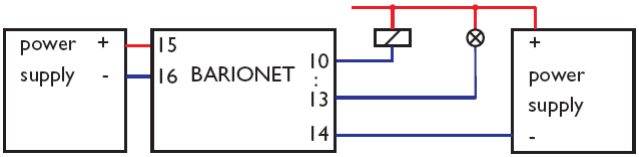


### Digital Outputs on Single Power Supply

Use the following schematic as an example when wired to a single power supply.


### Digital Outputs on Dual Power Supply

Use the following schematic as an example when wired to two power supplies.




### J6 15..16 Power Supply Inputs, Requirements and Display

The Barionet does not come with a power supply.
It can be operated on any DC power supply with an output voltage between 9 and 30 Volts DC or on a AC transformer (Door bell transformer) with an output voltage between 12 and 24 Volts AC due to a built-in power switching regulator.

The overall consumption is 3.5 Watt max. (all relays active). Therefore we suggest to calculate a power requirement of 4 Watt for each attached Barix Barionet.

The PWR LED is lit when the Barionet is powered.
A blinking CPU LED shows CPU activity.

**Attention:**
An AC power supply can be used if the Barionet is used stand alone. Use a DC power supply instead if you connect other Barix devices (IO12, R6, X8) over the RS-422/485 interface to the Barionet.

# 3 1st Time Browser Access

If the IP Address of the Barix Barionet is not known or undefined, it can be found out using two different procedures or set temporary via the third procedures as described in the following three sections.

## 3.1 DHCP Device List from Router

This procedure is applicable only if you have administrative access (user name and password) to a router with DHCP service (usually your Internet modem/gateway). Otherwise skip to the next section.

**Step 1**

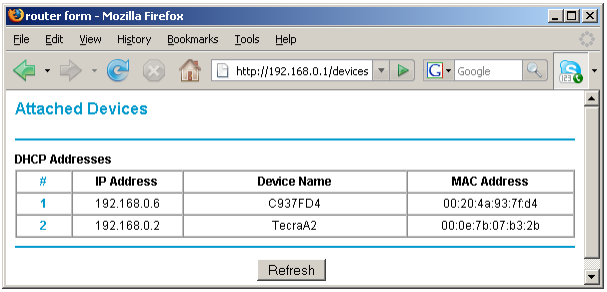
Use two network cables to connect the Barionet and the PC to the local network (LAN) over a network switch or directly to the router which is running the DHCP service.

**Step 2**

Power the Barionet and note the MAC address to find the IP address in the next step.

**Step 3**

Open a browser window on your PC and log into the router which is running the DHCP service. Look for Diagnostics or Maintenance and find the link to show the issued DHCP addresses (on some routers called attached devices ). Find the MAC address in the the list to obtain the issued IP address. Note the IP address and continue with the next chapter.



## 3.2 Getting IP from Syslog Message

This procedure is applicable only if you have a router with DHCP service running (usually your Internet modem/gateway) and can run a SYSLOG application like the free Kiwi Syslog Daemon (www.kiwisyslog.com). Otherwise skip to the next section.

**Step 1**

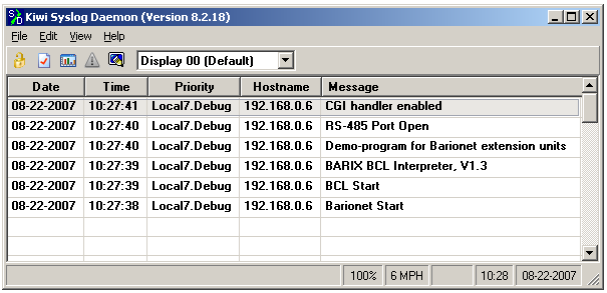
Use two network cables to connect the Barionet and the PC to the local network (LAN) over a network switch or directly to the router which is running the DHCP service.

**Step 2**

Run the SYSLOG application on your PC and power the Barionet.

**Step 3**

Find the issued IP address in the SYSLOG application in the Hostname column. Note the IP address and continue with the next chapter.



## 3.3 Setting temporary IP address using ARP command

This procedure is to temporarily set the IP address for the first time browser access.

**Attention:**
After this procedure the temporary IP will be active only as long as the Barionet stays powered. After a restart the procedure has to be repeated unless you have configured the IP address as described in the next chapter.

**Step 1**

Use either a crossover network cable between the Barionet and the PC or use two network cables to connect the Barionet and the PC to a network switch.

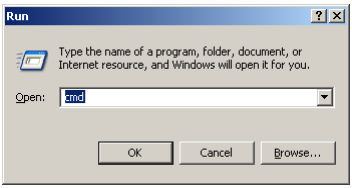
**Step 2**

Make sure that you have a valid IP address configured on your PC (e.g. 192.168.0.2) and power the Barionet.

**Step 3**

Open a command window with a click on Start followed by a click on Run.. .

In the Open field type **cmd** and click on OK



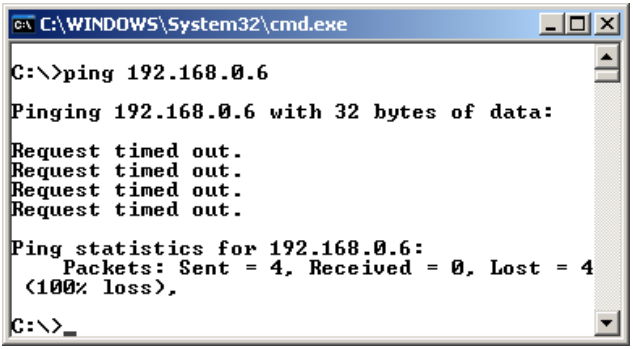

**Step 4**

Please skip this step if you used a crossover network cable in step 1 and proceed to step 5.

To ensure that we use a free IP address (not already used by another device in the network) we have to use the Ping command.

In this example we assume the PC to have the IP address 192.168.0.2 and want to check if 192.168.0.6 is free. To do so type into the command window

`ping 192.168.0.6` and hit the Enter key.




If you get a reply (IP already used) then try to ping another IP until you find one that is not used.

If the request times out (as shown in the above window) then the pinged IP is free and we can continue with the next step.

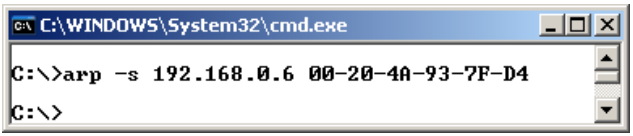
**Step 5**

Now we are going to make your PC talk to the Barionet's MAC address using the IP address 192.168.0.6. (The MAC address of the Barionet unit is printed on a label placed on the bottom of the device) using the arp command with parameter -s for set followed by the chosen IP address and the MAC address (12 hex digits, separated by a hyphen every 2 digits)

To do so type into the command window

`arp -s 192.168.0.6 00-20-4A-93-7F-D4`

and hit the Enter key.



### Step 6

Now we have to make the Barionet listening to the IP address 192.168.0.6 using the Telnet command.

To do so type into the command window

```
telnet 192.168.0.6 1
```

and hit the Enter key (the number one must be there for this command to work correctly !!!)

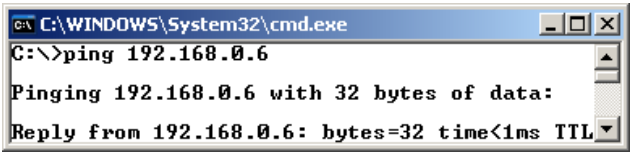

The Barionet will refuse the connection on port 1 (one) immediately but will be available for web access on the IP address used as long as the Barionet stays powered.

### Step 7

To check if the Barionet is responding you can use the ping command again. To do so type into the command window

```
ping 192.168.0.6
```

and hit the Enter key.




If you do get a reply (as shown in the above window) the IP address 192.168.0.6 can be used to access the Barionet using a web browser. Please proceed to the next chapter to set the IP address permanently.

If you do get Request timed out instead of a reply then please repeat step 6 carefully (most likely mistyped the telnet command).

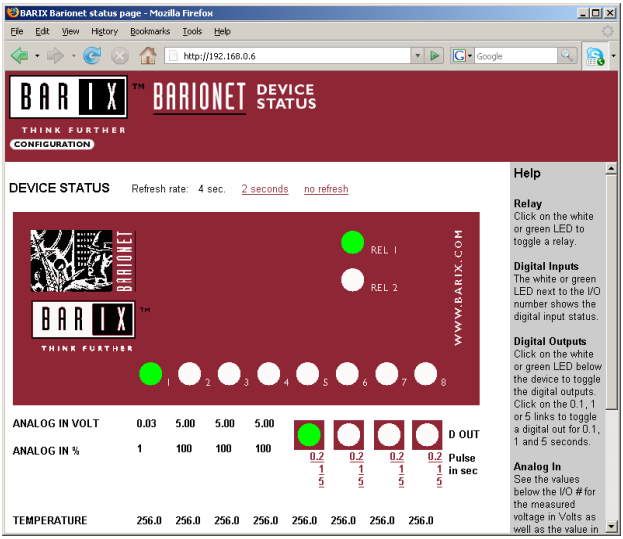
# 4 Device Status and Control

To view the current status and control it using the Web interface you will need:

- the IP address (configured or given by a DHCP server, see previous chapter)
- a standard web browser

Open your web browser and type in the IP address of the Barix device in the URL field and hit the Enter key.

You will see the following window content:



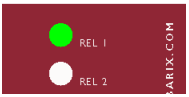

## Help

The right frame (grey background) contains help regarding the status and control functionality on this status page. Use the scrollbar to see further topics.

## Status page refresh rate


Next to the title DEVICE STATUS the refresh rate can be adjusted. The default is a refresh every 4 seconds which can be increased to a refresh every 2 seconds or switched off to reduce bandwidth usage as all graphics are loaded with every refresh.

## Status and Control of the Relays

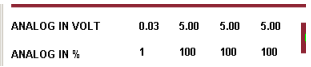



The 2 LED s on the right side of the red box show the state of the 2 relays and can be used to switch the relays on (green) and off (white) by clicking on them.

## Status of the Digital Inputs

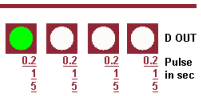

The eight LEDs on the bottom of the red box show the state of the 8 digital inputs (white for inactivated, green for activated). For simplicity reasons the first four LEDs do not have the same function as the LEDs on the device where LED 1 to 4 show PWR, CPU, RS-232 and RS-485.

## Status of the Analog Inputs



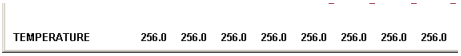

Below the red box the measured voltage of inputs on to four is displayed in Volts (0..5V) as well as in percent.

## Status and Control of the Outputs




The 4 LED s below the red box show the state of the 4 digital outputs and can be used to switch the outputs on (green) and off (white) by clicking on them. Below each output LED are 3 pulse links which toggle the output for 0.2, 1 or 5 seconds (e.g. on for 5 seconds and then off again when it was off before and vice versa).

## Status of Temperature Sensors




The temperature of the first eight attached 1-wire sensors is displayed in degrees Celsius. If the sensor is not attached a value of 256 degrees is displayed.

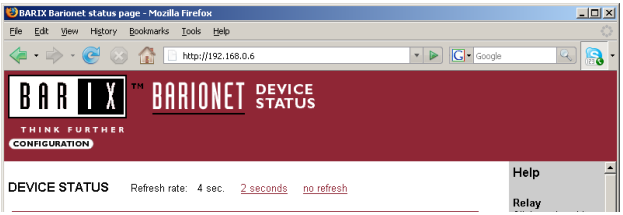
The status web page can by altered by the user to show up to 50 sensors and to display the value in Fahrenheit.

# 5 Device Configuration

To adjust the device settings (network, serial etc.) using the Web interface you will need:

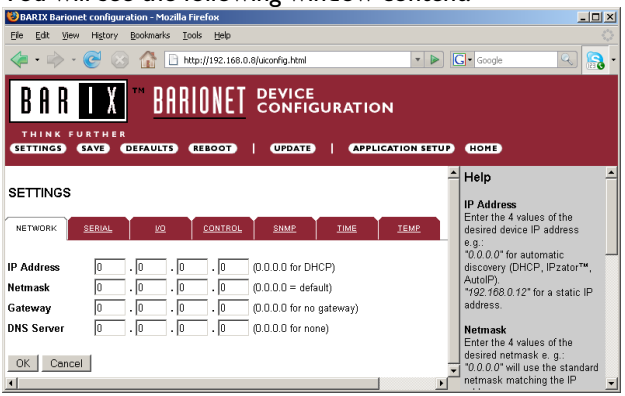
- the IP address (configured or given by a DHCP server, see previous chapter)
- a standard web browser

Open your web browser and type in the IP address of the Barix device in the URL field and hit the Enter key. You will see the following window content:




Click on the button labelled CONFIGURATION .

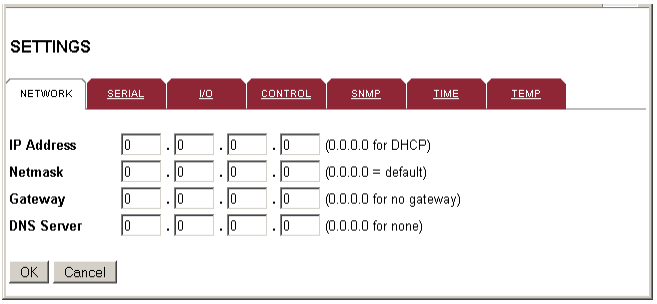

You will see the following window content:




The right frame (grey background) contains help on the configuration of each selected Settings tab.

## 5.1 Network settings

This chapter explains how to adjust the network settings. The changes will be stored by a click on OK . To apply the changes the Barionet needs a reboot.




### IP Address

We recommend that you set a Static IP address. With a static (permanent) IP address the device does not have to get a new IP address at power on or reboot.
See also chapter 3 1st Time Browser Access.

Enter the 4 values of the desired static IP address e.g.:

- `192.168.0.6` for an internal LAN

**Attention**: Make sure that you enter a free IP address. The device checks this at startup and will not be available until the device using the same IP is disconnected or powered off. The command Ping can be used to ensure that a specific IP address is unused (i.e. No reply).

The automatic discovery functions (see next section are not executed if a Static IP address is set.

### Automatic IP Discovery Functions

Enter 4 zeroes in the IP Address field to enable it:

- `0.0.0.0` for automatic discovery
  (Discovery order: BOOTP, DHCP, IPzator, AutoIP)

To enable automatic discovery but disable certain IP discovery functions set all but the third value to zero. For each function you want to disable add the value from the table below:

- `0.0.1.0` to disable AutoIP
- `0.0.2.0` to disable DHCP
- `0.0.4.0` to disable BOOTP
- `0.0.8.0` to disable IPzator

Examples:

- `0.0.3.0` disables AutoIP and DHCP
- `0.0.11.0` disables all but BOOTP

**Attention**: `0.0.15.0` disables all discovery functions which locks you out unless you reset the device to factory defaults as described in chapter X.

### Netmask

Enter the 4 values of the desired Netmask e.g.:

- `0.0.0.0` for a default Netmask depending on the IP Address used
- `255.255.255.0` for a C class network

**Note**: Try first the Netmask your PC is set to or ask your Network Administrator.

### Gateway IP Address

A correct setting of the Gateway IP address is needed when communicating over the Internet or to destinations outside your local network (LAN).

Enter the 4 values of the desired Gateway IP address e.g.:

- `0.0.0.0` for no Gateway or automatic discovery
- `192.168.0.1` for a Gateway in a LAN

**Note**: If you have set the devices IP address to automatic discovery and your server or router has issued an IP

address then most probably it also has supplied a valid Gateway address, 0.0.0.0 will then work fine.

If you have set a Static IP address then you will need to configure a valid Gateway address manually. Try first the same Gateway IP address your PC is set to. If it doesn t work then ask your Network Administrator for a correct Gateway IP address.

### DNS Server

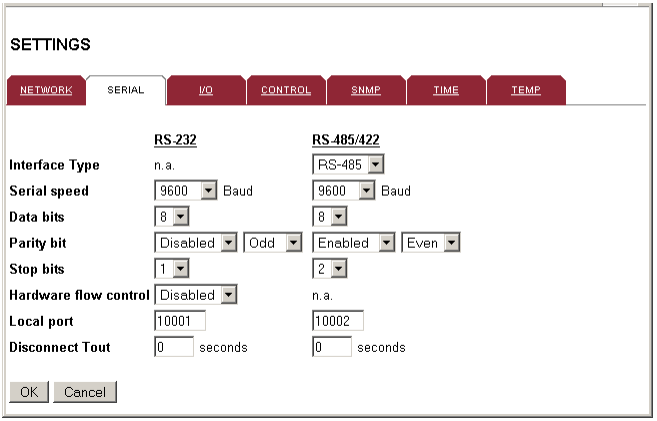
When using a URL (e.g. **`www.myserver.com`**) instead of an IP address in a BCL application, the IP address of an available DNS Server is needed to translate the URL into an IP address (resolving the URL).

Enter the 4 values of the desired DNS server e.g.:

- `0.0.0.0` for no DNS server
- `192.168.0.1` for a DNS server in a LAN

## 5.2 Serial settings

This chapter explains how to adjust the settings for both serial interfaces. The changes will be stored by a click on OK . To apply the changes the Barionet needs a reboot.



**Interface type**

For the RS-485/422 serial port choose the desired wiring type between 2 wire (RS-485) or 4 wire (RS-422).

**Serial Speed**

The baud rate for both the RS-232 and the RS-485 serial interface can be set to the common values of 300, 600, 1200, 2400, 4800, 9600 and 19200 Baud. The factory default is 9600 Baud for both interfaces.

**Data bits**

Select the number of serial data bits (8 or 7).
Default setting is 8 data bits.

**Parity bit**

Select the parity bit settings.
Default setting is disabled.
If enabled select Even or Odd parity.

**Stop bits**

Select the number of serial stop bits (1 or 2).
Default setting is 1 stop bit.

**Hardware flow control**

For the RS-232 interface only select the Hardware flow control (RTS/CTS) to be enabled or disabled.

**Local port**

Defines the TCP port on which the serial port can be accessed for a remote application.
Enter 0 to disable the remote access.

**Disconnection Tout**

Enter the amount of seconds after which the TCP connection (see Local port above) will be closed due to lack of traffic (no RX or TX data).

### Discontinued: Gateway Destination IP and Port

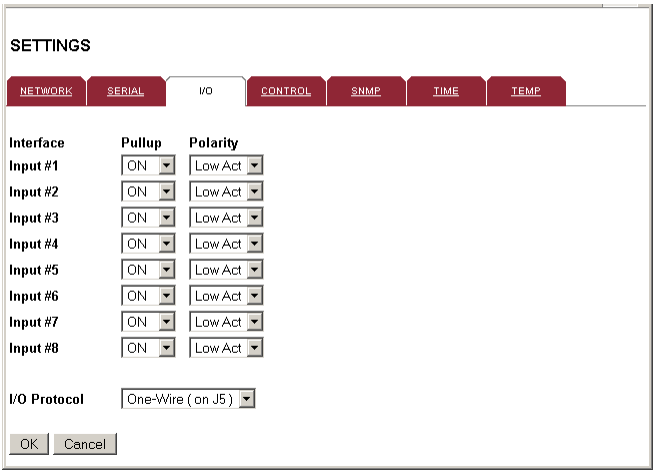
In a early version the Barix Barionet allowed to set up a serial gateway. This is no longer supported by the firmware.

Instead one can load a BCL application called Serial Tunnel which supports the same functionality and has the advantage that it can be customized by the user.

The BCL application Serial Tunnel is included in the Update & Rescue Kit V2.08.

## 5.3 I/O Settings

This chapter explains how to adjust the interface characteristics of the digital inputs and the selection of the used I/O protocol. The changes will be stored by a click on OK . To apply the changes the Barionet needs a reboot.




### Pull-up Resistors

The Barix Barionet digital/analog inputs have individual, configurable pull up resistors with a value of 10 K Ohms each.

If an input is used with a contact (or button), the pull up

must be enabled to get reliable data. If a digital signal which is actively driven high is applied to an input, the pull up should be disabled.

If an input is used as an analog input, the pull up **must** be disabled to achieve accurate readings. In connection with an external loop resistor, the pull up may be used on an analog input to achieve line supervision.
Default setting on all inputs is on .

### Polarity

The digital inputs of the Barix Barionet can be used to interface directly to switches, push buttons and the like, or, alternatively, to digital signals providing a voltage signal.

Depending on the application, the polarity of the input should be set correctly so that the application can read the correct state of the input.

With pull-ups enabled and switches or buttons connected, the polarity should be set to Low Act . With an actively high voltage (> 1.2 VDC) the polarity should be set to High Act .

Default setting is Low Act .

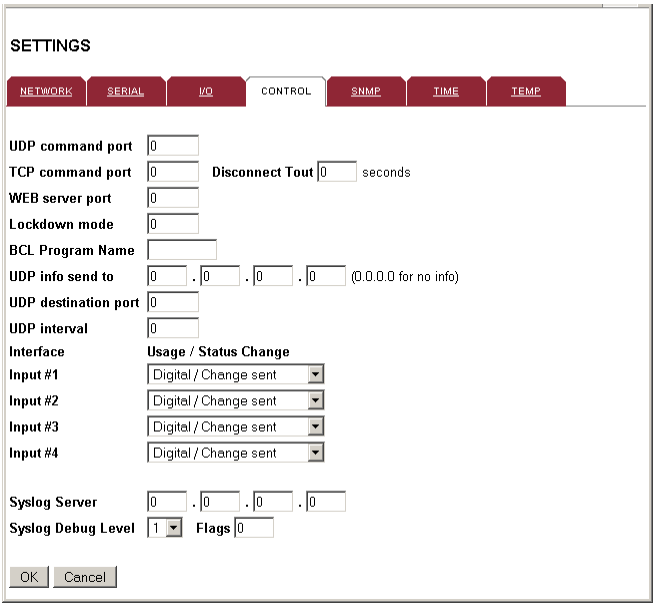
### I/O Protocol

The I/O protocol and interface can be chosen. Select between Dallas 1-Wirefi on J5 (default setting) and Wiegandfi protocol interface on J6 connector.

Selecting Wiegand disables the Dallas 1-Wirefi temperature sensors and displays 256.0 C for all 8 sensors on the status page (HOME).

## 5.4 Control Settings

This chapter explains how to adjust control settings. The changes will be stored by a click on OK . To apply the changes the Barionet needs a reboot.




### UDP command port

Configure the listening port (1 . . 65535) for the UDP command interface (API) to which commands can be sent. The same port number is used for outgoing UDP messages as the source port.
Default setting is 0 (disabled).

### TCP command port

Configure the connection port (1 . . 65535) for the TCP command interface. An application can connect to this port and will receive changes of any I/O signal and can issue commands to change output states. Only one application can talk to the TCP port of the Barix Barionet at a given time. Default setting is 0 (disabled).

### Web server port

Defines the port on which the web server of the Barionet can be reached. If set to "0" the default HTTP port 80 is used. Default setting is `0` .

### Lockdown mode

This parameter can be used to lock access to specific services at start up. To unlock them during runtime the BCL command **`lock`** can be used.
Please refer to the BCL Programming Manual .

Default setting is `0` which means all unlocked.

| Bit index | Service |
|---|---|
| 0 | snmp write |
| 1 | snmp read |
| 2 | modbus/tcp write |
| 3 | modbus/tcp read |
| 4..7 | reserved |
| 8 | rc.cgi |
| 9 | i/o dynamic tags |
| 10 | setup.cgi |
| 11 | setup dynamic tags |
| 12 | BAS.cgi |
| 13 | basic variable dynamic tags |
| 14 | Basic.cgi |
| 15 | tftp |

### BCL program name

Defines the name of the BCL application (without the **`.tok`** extension) to be run at startup of the Barionet. This allows for multiple BCL application (using different names) to be loaded into the Barionet. If no name is given then **`barionet`** will be used instead.
Default setting is empty (equal to **`barionet`** ).

### UDP Info Send To

A specific destination IP address (e.g. `192.168.0.9`) or a subnet broadcast address (e.g. `192.168.0.255`) can be set to define where the I/O status change information should be sent to using UDP data packets.

### UDP Destination Port

Defines the UDP destination port to be used when sending I/O status change information.

### UDP Interval

Defines the time interval in seconds for repeatedly sending current I/O status information.

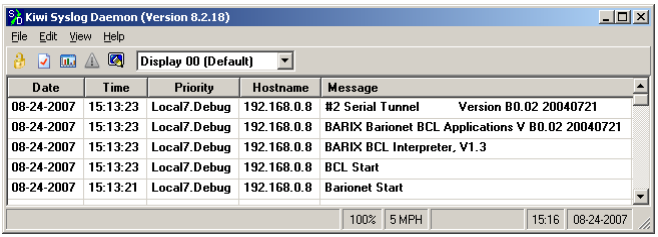
### Usage / Status Change

For input #1 to #4 select the input to be either used as an analog input and to suppress status change information or to be used as a digital input and to include status change information in UDP Info Send (see above).

### Syslog Server IP Address

Syslog is a well known reporting application using UDP port 514.
Check the Internet for a free Syslog daemon like the free Kiwi Syslog Daemon (www.kiwisyslog.com).

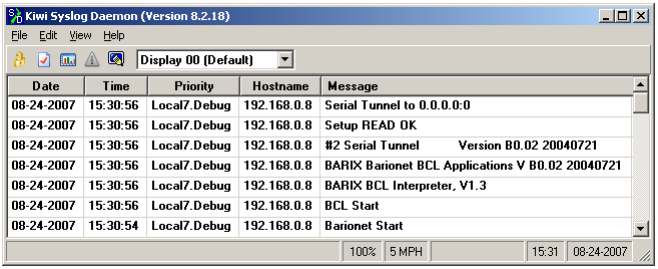
Enter either "0.0.0.0" for subnet broadcasting of Syslog messages or the IP address of a dedicated Syslog server.



## Syslog Debug Level

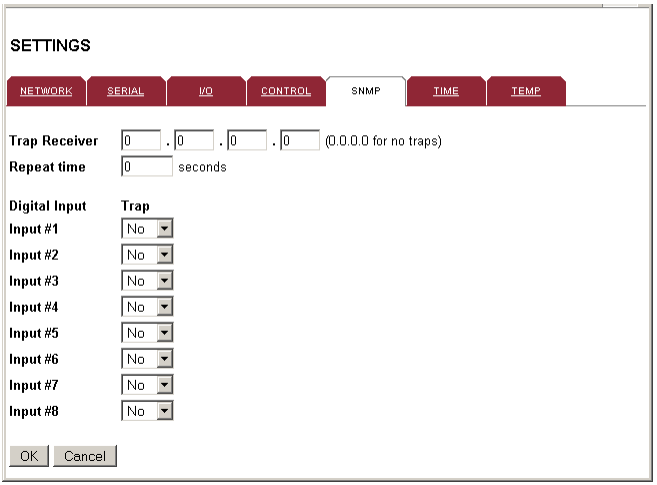
Enter the level of Syslog debug information e. g.:

- "0" for no debug information
- "1" for system debug information (default setting see also screen shot below)
- "2" to "9" for user defined BCL debug information.



## 5.5 SNMP Settings

This chapter explains how to adjust SNMP settings. The changes will be stored by a click on OK . To apply the changes the Barionet needs a reboot.




### Trap Receiver

This parameter specifies to which IP address SNMP traps, which can be generated under certain conditions in the Barix Barionet, will be sent. If set to `0.0.0.0` (factory default), no traps will be generated.

### Repeat Time

When an input is active the trap will be issued repeatedly with this time in seconds until the input becomes inactive.

Enter `0` to deactivate repetition (results in two traps being sent, one for an input going active and one for it going inactive but no further traps will be sent).

### Digital Input Trap

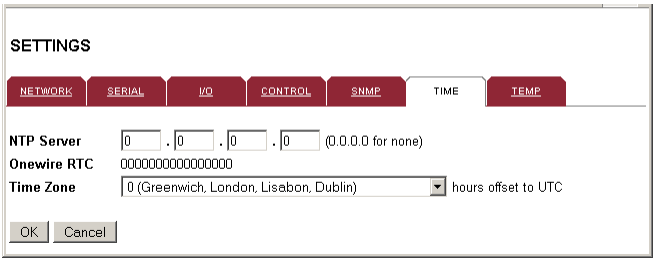
For each digital input select if a trap should be issued when the state changes. Default setting is **`No`** .

### SNMP definition file (Private MIB)

In order to use the SNMP functions a private MIB definition file called Barionet.MIB is needed. See the Help frame on the right to download the MIB file as a text or a ZIP file.

## 5.6 Time Settings

This chapter explains how to adjust time settings. The changes will be stored by a click on OK . To apply the changes the Barionet needs a reboot.




### NTP Server

When a valid NTP Server is given, the Barix Barionet will set the internal clock accordingly at start up and will synchronize again approximately every 12 hours.

Enter "0.0.0.0" for no NTP Server

or the IP address for a specific NTP Server.

### Onewire RTC

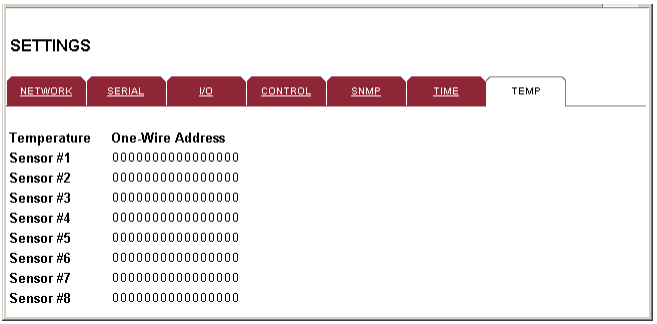
Displays the 1-wire Real Time Clock unique hardware ID if attached or 0000000000000000 if the RTC is absent.

### Time Zone

Enter the time zone information for your geographic location in order to display the correct local time. The reference is called UTC and is based in Greenwich UK.

## 5.7 Temperature Settings

This chapter explains the function of the TEMP tab. No changes can be made on this page.



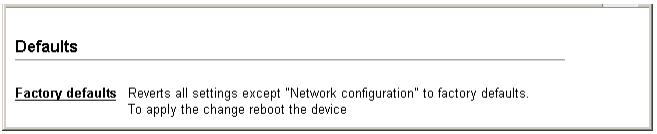

### Temperature Sensor Information

This page shows the detected unique ID's of the Dallas 1-wirefi bus temperature sensors connected to the Barix Barionet.

They will be entered every time in exactly that sequence in the table by the discovery function, which is executed after a reset and on start up of the Barix Barionet.

# 6 Reverting to factory defaults

To revert settings to factory defaults click the link DEFAULTS in the Device Configuration page.




The Barix Barionet can be reset to the factory defaults. However, the network settings will NOT be changed to prevent one from loosing the network connection.

When clicking on the link Factory defaults the following information appears for about 3 seconds:

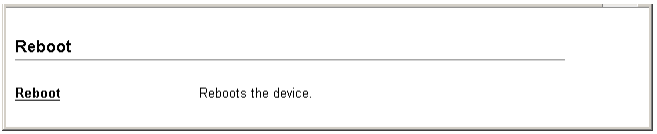

To apply the changes the Barionet needs a reboot. The user is guided automatically to the Reboot page.


Click on the link Reboot .


After approx. 5 seconds the following will be displayed.

# 7 Rebooting the device

To reboot the Barionet click the link REBOOT in the Device Configuration page.




Click on the link Reboot .


After approx. 5 seconds the following will be displayed.

# 8 Updating the device

Barix preloads all Barionet devices with firmware which is current at the day of production.

Barix constantly enhances the capabilities and functions and recommends to keep the firmware on the Barix Barionet up-to-date.

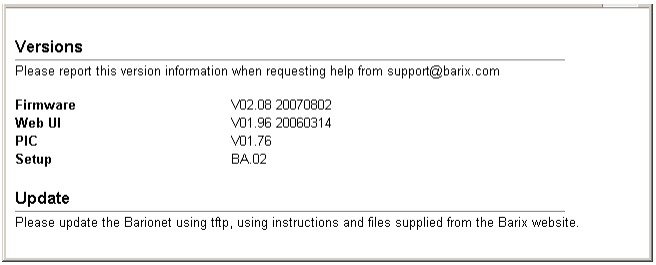
Barix recommends the use of the Serial Rescue method to update the firmware.

For the alternative method Update over network using TFTP we strongly recommend use the supplied batch files which are included in the Barionet Update Kit .

If the network update is interrupted during the process (power or network loss) the device might become unreachable. In that case the Serial Rescue procedure is the only remedy. Please keep that in mind when planning a remote Update over network .

## 8.1 Checking the Current Version

To see which version is currently installed in the Barionet click the link UPDATE in the Device Configuration page. user is guided automatically to the Reboot page.




Verify if a newer version is available as described in the next chapter.

**Version information**

- Firmware shows the version and the release date of the Barionet firmware.
- Web UI shows the version and the release date of the Web User Interface.
- PIC Version shows the versions of the embedded PIC micro controller.
- Setup shows the version of the setup table in the configuration EEPROM.

## 8.2 Downloading the latest firmware

To download the latest Barionet firmware version please visit www.barix.com.

- In the menu on the left side click on **`downloads`**
- In the center column click on **`Barionet`**
- In the section **`Firmware`** verify if the version of the **`Barionet Update Kit`** is higher then the currently loaded (see version information above) If there is a newer version click in the **`Barionet Update Kit`** link to download it.
- Save the ZIP file and unpack it to a local drive.

Read the "**`Readme1st.txt`**" file for detailed instructions.

## 8.3 Serial Rescue

This procedure using a serial cross over cable is needed in case a Barionet is not reachable over the network.

### Pin out null modem cable, 2 x D-Sub 9 pin, female

| Function | Pin | Cable | Pin | Function |
|---|---|---|---|---|
| RXD | 2 | | 2 | RXD |
| TXD | 3 | | 3 | TXD |
| GND | 5 | | 5 | GND |

The Serial Rescue procedure loads the firmware, the Web UI (user interface) but no BCL application and resets the device to factory default settings. The current configuration is overwritten except for the network settings.

In order to load the BCL application complete the serial rescue first and run then the batch file **`bcl.bat`** described in the next chapter.

### Different XPort Types (LX / EX)

There are two different start commands for the Barionet serial rescue depending on the built-in XPort.
The older Barionets (produced 2005 and earlier) came with a LX-XPort while newer Barionets (2006 and later) have the EX-XPort!

If you don·t know which one is built in please remove the Barionet·s top plastic cover and check the product number on the XPort Network connector.

For the new Barionet EX-XPort
(XPort product number : XP1001001-03R) use:

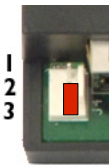
- rescue1.bat (when using COM port 1)
- rescue2.bat (when using COM port 2)
- rescue3.bat (when using COM port 3)
- rescue4.bat (when using COM port 4)

For the older Barionet LX-XPort
(XPort product number : XP1001000-01) use:

- lxrescue1.bat (when using COM port 1)
- lxrescue2.bat (when using COM port 2)
- lxrescue3.bat (when using COM port 3)
- lxrescue4.bat (when using COM port 4)

## Serial Rescue Procedure

- disconnect the Barionet from the power supply
- set the update jumper on pin 2-3 of connector J4




- connect the serial null-modem cable (for pin out see previous section) to the Barionet·s RS232 port and to the PC·s COM port
- start the batch command depending on the XPort type and PC COM port (see section above)
- wait until you see the prompt:
"Unplug the device's power and plug it back. Transmitting requests to the device."
- power the Barionet
- wait until you see the message: "Rebooting the device... The device should have been rebooted now."
- disconnect the Barionet from the power supply, remove the jumper and power the Barionet

The Serial Rescue procedure loads the firmware, the Web UI (user interface) but no BCL application and resets the device to factory default settings. The current

configuration is overwritten except for the network settings. If you do not know the configured IP address please refer to chapter 3_1st Time Browser Access.

In order to load the BCL application complete the serial rescue first and run then the batch file **`bcl.bat`** described in the next chapter.

Calculate approximately 2 minutes to complete the Serial Rescue procedure. After a successful rescue the device is ready for configuration according to your needs (see chapter 5_Device Configuration).

## 8.4 Update over Network

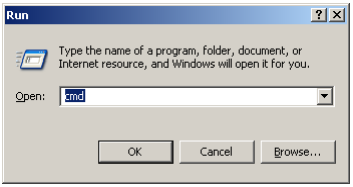
For the alternative method Update over network using TFTP we strongly recommend use the supplied batch file **`barionet`** which is included in the Barionet Update Kit . To update over the network you will need:

- the IP address (configured or given by a DHCP server, see previous chapter)

**Attention:**
If the network update is interrupted during the process (power or network loss) the device might become unreachable. In that case the Serial Rescue procedure is the only remedy. Please keep that in mind when planning a remote Update over network .

### Update over network procedure

- Open a command window with a click on Start followed by a click on Run.. .
- In the Open field type **`cmd`** and click on OK




- Change the working directory to the barionet_update_kit folder and type the command:

  **`barionet <IP address>`**

  where <IP address> is the IP address of your Barionet.

  Example: **`barionet 192.168.0.6`**

This will create two COB files: **`barionetweb.cob`** and **`barionetbcl.cob`** and transfers both to your Barionet using TFTP.
The firmware **`bn208.rom`** is transferred at last and will cause the Barionet to restart.

The complete procedure takes about 25 seconds.

The device is now ready for configuration according to your needs (see chapter 5 Device Configuration).

**Note:** The Update over network procedure does not change the current configuration.

Barix recommends nevertheless checking for correct parameters in the configuration as well as to set parameters for newly added features.

If you observe strange behavior after an update Barix recommends resetting the configuration to factory defaults (see chapter 6 Reverting to factory defaults) and to start over with the configuration of the device.

## 8.5 Additional Batch Files for Programmers

Programmers can use the following batch files to update or erase portions of the WEB pages stored in the internal flash memory.

### Updating WEB application

To COB and upload the WEB application type:

**`web <IP address>`** e.g. **`web 192.168.0.6`**

The WEB application (including web UI, web help, graphics, MIB, MIB.zip, PIC firmware and version file) is 116 KB (119'149 bytes) in size.

The upload to **`WEB1`** (64KB) also uses **`WEB2`** (64KB) to store the file **`barionetweb.cob`** .

As you might add more graphics and HTML pages when customizing it we have reserved **`WEB3`** (64KB) for you. This allows for a WEB application COB file up to 192 KB in size without the need of changing the **`bcl`** batch command pointing at **`WEB4`** (64KB) as described in the next section.

### Updating BCL application

To tokenize the BCL source code, to COB all files and to upload the complete BCL application type:

**`bcl <IP address>`** e.g. **`bcl 192.168.0.6`**

The included sample BCL application "Digital I/O tunnel", it's web setup HTML pages and the BCL source **`barionet.bas`** is 30.9 KB (31,682 bytes) in size.

To store the file **`barionetweb.cob`** the upload is done to **`WEB4`** (64KB). Please refer to the WEB usage table further below for information regarding space requirements.

### WEB usage table:

Rules:

- Each COB file will occupy at least 64 KB
- If it is bigger than 64 KB it will use two or more WEB pages
- A file contained in a COB file should not exceed 64KB

| Name | Size KB | Usage |
|---|---|---|
| X1 | 64 | Firmware |
| WEB1 | 64 | Web UI application , Help, PIC Firmware |
| WEB2 | 64 | Web UI application , Help continued |
| WEB3 | 64 | Reserved for Web UI application extension |
| WEB4 | 64 | Sample BCL application Digital I/O Tunnel (including application setup) |
| WEB5 | 64 | Free |
| WEB6 | 64 | Free |
| WEB7 | 64 | Free |

### Erasing the BCL application

To erase the BCL application (content of WEB4) type:

**`erasebcl <IP address>`**
e.g. **`erasebcl 192.168.0.6`**

### Erasing all WEB pages

To erase all data (content of WEB1 .. WEB7) type

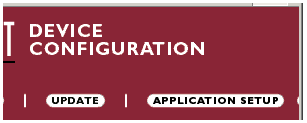
**`erase <IP address>`** e.g. **`erase 192.168.0.6`**

# 9 BCL Application Digital I/O Tunnel

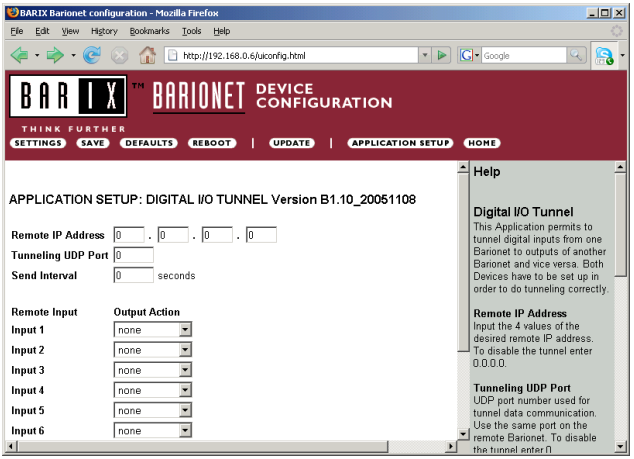
The sample BCL application Digital I/O Tunnel is included in the Barionet Update Kit and allows for transmitting of digital input states of one Barionet to outputs of another Barionet over the network and vice versa. Using a Barionet on each side up to 6 signals can be transmitted in both directions. Using a Barionet and a Barix R6 on each side up to 8 signals can be transmitted in both directions. In addition one output can be used as an alarm in case the network connection is lost.

Both devices have to loaded with the Digital I/O Tunnel and both sides have to be set up to work correctly.

To configure the application "Digital I/O Tunnel" click the menu button "APPLICATION SETUP" in the CONFIGURATION menu :

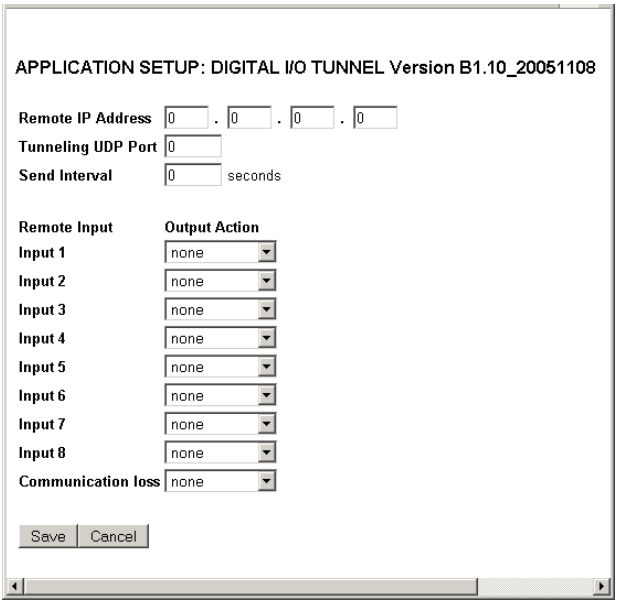



You will see the following configuration window:




The grey frame on the right contains help information.

## 9.1 Application Setup




### Remote IP Address

Input the 4 values of the desired remote IP address. To disable the tunnel enter `0.0.0.0`.

### Tunnelling UDP Port

Enter the UDP port number (`1..65535`) used for tunnel data communication. Use the same port on the remote Barionet. To disable the tunnel enter `0`.

### Send Interval

Input the time in seconds (`2..65535`) between two status messages. In case no inputs have changed (changes are sent immediately) a status message is sent to permit the other side to detect a communication loss.

The same value should be set on the remote Barionet. To disable Loss checking enter 0.

The communication loss will be triggered 1 second after the double time of the remote stations send interval to allow 1 lost UDP packet eg.:

- when interval is 5 seconds the "comuncication loss action" will be triggered 11 seconds after the last received I/O status

### Output Action

Choose an output for each remote input and one for the communication loss.

Choose Relays 1 or 2, Output 1 to 4 or Extension Relays 1 to 8 (requires Relay module Barix R6).

# 10 HTTP interface (CGI API / DHTML)

Custom web pages can be generated and uploaded by the user, and are then served by the web server built into the Barix Barionet.

Outputs can be controlled via CGI commands, and the status of all I/O functions can be used to dynamically create content in HTML/XML pages.

With this functionality, one can build an XML or HTML representation of whatever is connected to the Barix Barionet without changing the firmware of the device.

## 10.1 CGI commands

### CGI command o= to set an output

The relays and digital outputs can be controlled via the following CGI command:

- **`http://ip/rc.cgi?o=I/O,value`**

where

`I/O` is the I/O address in the Barix Barionet (see chapter 12 I/O Address Mapping)

and **`value`** is the action (what to do):

| I/O | Output |
|---|---|
| 1 | Relay 1 |
| 2 | Relay 2 |
| 101 | Digital out 1 |
| 102 | Digital out 2 |

| value | Function |
|---|---|
| 0 | Set output to inactive |
| 1 | Set output to active |
| 999 | Revert (toggle) output |
| n | Pulse output for n*100ms, values 2-998 |

Example: switch on Relay 1:

**`http://192.168.0.6/rc.cgi?o=1,1`**

### CGI command L= to load a HTML page

To specify a custom response page in XML or HTML use the command L=filename (the & can be used to concatenate several commands in a single CGI request).

Example: switch off Relay 2 and load uistatus.html:

```
http://192.168.0.6/rc.cgi?o=2,0&L=uistatus.html
```

## 10.2 Dynamic HTML tags (DHTML)

Certain HTML tags will be interpreted by the web server in the Barix Barionet and replaced by a string formatted to the users needs using the values of the inputs or outputs.

These tags can then be used to simply display values, add the values into XML documents, but also to generate links, file names for graphic symbols etc.

### DHTML Initialization Tag

To be interpreted, a HTML file must have the following tag in the first 512 bytes (typically, the header):

- `&L(0,"*");`

The tag may be embedded in a comment, it just needs to be present in the file.

Example:

- `<!-- &L(0,"*"); -->`

## 10.3 DHTML I/O Tag Type 1

The following tag addresses the I/O function.

- `&LIO(1,"format",I/O);`

See chapter 12 I/O Address Mapping for the I/O addresses for this and other functions.

The format string is a typical C style string which is used to generate the output, and must contain an unsigned data type with optional digit number for the standard inputs and outputs.

The following sections explain how to use these tags.

### LIO Tag for Digital Output State

This function reads back the status of the first digital output and will yield either 0 or 1.

- `&LIO(1,"%u",101);`

### LIO Tag for Digital Input State

This function reads back the status of the second digital input and will yield either 0 or 1.

- `&LIO(1,"%u",202);`

### LIO Tag for Analog Input Binary Value

This function reads the value of the analog input #2, and will yield a four-digit number in the range 0000..1023 with leading zeros.

- `&LIO(1,"%04u",502);`

## 10.4 DHTML I/O Tag Type 2

The second function type can be used to display analog values in a certain range, by first multiplying them with a constant, then adding an offset, then dividing by another constant, and then displaying with a potential decimal point.

```
&LIO(2,"format",I/O,mul,ofs,div,exp);
```

Please note that this function only makes sense for the analog inputs and for the temperature sensors.

The following sections explain how to use these tags.

### LIO Tag for Analog Input Voltage Value (0 to 5 VDC)

This function displays the voltage of the analog input #1 (I/O=501) in the range of 0.00 to 5.00 VDC.

```
&LIO(2,"%.2F",501,500,0,1023,100);
```

The function reads analog input #1,
multiplies it by 500 (mul=500),
does not add an offset (ofs=0),
then divides by 1023 (div=1023 , result is 0..500),
dividing by 100 (exp=100, range 0.00 to 5.00)
and displays the value with two decimal digits after the decimal point ( %.2F ).

### LIO Tag for Analog Input Voltage in Percent (100%=5 VDC)

This function displays the voltage of the analog input #1 (I/O=501) in the range of 0 to 100 percent.

```
&LIO(2,"%u",501,100,0,1023);
```

The function reads analog input #1,
multiplies it by 100 (mul=100),
does not add an offset (ofs=0),
then divides by 1023 (div=1023 , result is 0..500)
and displays the value without decimal point ( %u ).

### LIO Tag for Temperature Sensor Value in Celsius

This function displays the temperature of sensor #2 (I/O=602) in the range of -55.0 to 125.0 degree Celsius.

If no sensor is attached than the value 256.0 is shown.

- `&LIO(2,"%.1F",602,5,0,8);`

The function reads temperature sensor #1, multiplies it by 5, does not add an offset, then divides by 8 (explained further below) and then displays the value as degrees Celsius with one decimal digit after the decimal point.

This tag values work for the suggested 1-wire sensors DS18B20 and DS1820.

Explanation:
The resolution of the sensors is 0.0625 Celsius,
5/8=0.625 and the %.1F divides at the end by 10.

### LIO Tag for Temperature Sensor Value in Fahrenheit

This tag displays the temperature of sensor #3 (I/O=603) in the range of -67.0 to 257.0 degree Fahrenheit.

- `&LIO(2,"%.1F",603,9,2560,8);`

The function reads temperature sensor #1 in Celsius but displays the temperature in degree Fahrenheit.

The resolution of the sensors is 0.0625 Celsius,
5/8=0.625 and the %.1F divides at the end by 10.
Fahrenheit is 1.8 times Celsius plus 32.
1.8 * 5 (from above) = 9 (mul)
As we divide by 8 (div) we have to multiply 32 by 8:
32*8=256
now multiply by 10 as %.1F divides at the end by 10:
10*256=2560 (ofs)

## 10.5 DHTML Setup Tags

The HTML pages for the device configuration use DHTML tags and the CGI interface setup.cgi .

All of the configuration parameters are placed in HTML forms and are transferred by the method GET. Some of the values are checked by the Java script util.js to prevent wrong values.

Not all of the configuration parameters have to be present in a form. It is possible to have only a part of the configuration on a web page.

### HTML form tags

The form has to start with the following two tags:

- `<form action=setup.cgi method=get>`
- `<input type="hidden" type="text" name=L value=uisaved.html>`

The answer after transmitting the form will be the HTML page uisaved.html . There you are asked to click on the Reboot button to restart the Barionet with the new settings. Here the link used:

- `<a href="setup.cgi?L=uireboot2.html&R" >Reboot</a>`

The command `&R` ill cause the device to reboot. To skip this additional step you could insert the command in the second tag above after the uisaved.html and before the `>` .

The following tags are needed at the end of the form:

- `<input type="submit" value=" OK ">`
- `<input type="reset" value=" Cancel ">`
- `</form>`

## DHTML Names and &Lsetup Tags

The input element name is a defined string:

- If the value is an integer the first character is a **B**.
- If the value is a Netmask the first character is an **N**.
- If the value is a string the first character is an **S**.
- If the value is a long (4 byte) the first character is a **D**.
- If the value is a signed byte the character is a **C**.
- if the value is a word the character is a **W**.

The following example shows how to implement a form field for the configuration value of the highest byte in the 'own IP address'.

- **<input name=B0 size=3 maxlength=3 value=&LSetup(1,"%u",0); onChange=IPCheck(this)>**

The type character B stands for an unsigned value (see table above). The 0 next to it is the address of the expected configuration parameter (see the 2nd column in the Setup Record Table) .
Together they form the so called DHTML Name (see the 3rd column in the Setup Record Table) .

The input element value uses the &Lsetup tag which has the following syntax:

- **&Lsetup(id,"format",address,type);**

Where id **1** is used for an unsigned Byte or Word, **2** for a Netmask and **3** for Bit values.

Here some examples that are used in the Barionet:

- **<input name=S100 size=8 maxlength=8 value=&LSetup(1,"%s",100,S);>**
- **<input name=B0 size=3 maxlength=3 value=&LSetup(1,"%u", 0);onChange=IPCheck(this)>**
- **<input name=W8 size=5 maxlength=5 value=&LSetup(1,"%u",8,W); onChange=PortCheck(this)>;**

- `<input name=N0 size=3 maxlength=3 value=&LSetup(2,"%u",6,0);`
- `<option value=0&LSetup(3,"%s",64,b0,0," selected");>ON</option>`

The part `onChange=IPCheck(this)` is making use of the mentioned Java script for validity checking.

For more examples (like drop down selection fields) please refer to the HTML files included in the stuff folder of the Barionet Update Kit.

# 11 ASCII interface (TCP/UDP API)

The Barix Barionet ASCII based protocol allows easy control of the Barix Barionet via either a TCP connection or UDP.

Only one application can talk to the TCP port, whereas multiple control servers can talk quasi simultaneously to the Barix Barionet using the UDP version of the protocol.

In addition, the Barix Barionet sends out status and status change information via UDP and the TCP connection unsolicited, if configured in the CONTROL configuration tab.

## 11.1 General rules

Every command of the protocol consists of the command word, eventual parameters, which, if existing, have to be added to the command, delimited via comma, plus the carriage return character (hex 0x0d).

Via the UDP interface, only one command can be sent at a time via a UDP block.

The Barix Barionet might bundle multiple responses, which are built up the same way, in one UDP data block to save space and eliminate unnecessary network traffic.

All commands are case sensitive.

If commands address I/O functions, they must carry an I/O address as the second parameter. Valid I/O addresses are defined in chapter 12 I/O Address Mapping.

## 11.2 ASCII Commands

Commands can be sent via TCP or UDP using the port as configured in the CONTROL configuration tab.

In the UDP interface, only one command may be sent at a time in UDP data blocks.

Unknown or invalid commands yield the response **`cmderr`** .

The following sections explain all available commands.

### iolist

The **`iolist`** command retrieves a list of the I/O capabilities in the Barix Barionet.

The return string contains the counts of the following I/O types (in this sequence and comma separated):

- analog inputs
- digital inputs
- analog outputs *
- digital outputs
- IR outputs *
- relays
- temperature sensors

* Please note that these I/O functions are not available in the standard Barix Barionet (HW 2.4 and earlier).

The answer of a standard Barix Barionet with no connected temperature sensors is as following:

- **`io,4,8,0,4,0,2,0<CR>`**

Which translates to:

4 analog inputs, 8 digital inputs, 4 digital outputs, 2 relays

### getversion

This command retrieves the hardware type and software version of the device in the following syntax.

- **`version,type <firmware version>`**

The standard Barix Barionet returns

- **`version,BARIONET 2.08`**

### interval,x

This command can be sent to configure the Barix Barionet to send status information for all inputs and outputs in regular intervals, indicated by the parameter **`x`** , in seconds. This is useful if analog values should be read and continuously updated without polling. Setting the interval to `0` switches off the regular update.

**Attention:**
The command only works for the TCP interface.
For the UDP interface the interval is set in the configuration tab Control only and cannot be changed.

### setio,I/O,val

This command is used to set an output of the Barix Barionet to either active, inactive, toggle it or pulse it (reverse it) for a certain time. The `I/O` address is given as defined in Appendix C. The value **`val`** can either be `1` for on , `0` for off , `999` to toggle the current state or a number between `2` and `500` for the pulse option. The length of the pulse is given in units of 100 ms (1/10$^{th}$ of a second). Here some examples:

- **`setio,1,1`** activate relay 1
- **`setio,101,999`** toggle digital output 1
- **`setio,2,50`** pulse relay 2 for 5 seconds

The **`setio`** command can only address existing outputs, it will yield a command error (**`cmderr`**) if applied to inputs or invalid I/O addresses.

### getio,I/O

This command is used to enquire the state of a specific input or output of the Barix Barionet. In response to this enquiry, the Barix Barionet sends a state message indicating the actual state (or value for analog inputs) of the addressed function.

Example 1:

- **`getio,501`** get value for analog input #1

Barionet responds with:

- **`state,501,555`** (Analog input #1 value = 555)

Example 2:

- **`getio,1`**

Barionet responds with:

- **`state,1,0`** (Relay #1 is inactive)

### State change messages

Barix Barionet sends change information for bit variables automatically via UDP and TCP command interfaces. This feature cannot be disabled. In addition, regular updates can be sent, however, this has to be configured for UDP or set with the **`interval,x`** command via TCP.

The status change messages are sent any time the status of a Barix Barionet digital input or output changes, and are transmitted in the following format:

- **`statechange,I/O,value`**

I/O is the address of the I/O function which changed (see table above), value is 0 (inactive) or 1 (active). Note that also the change of outputs is indicated !

In addition, if enabled, Barix Barionet sends the state of

ALL I/O variables which are used (including analog inputs and virtual variables) in the set interval. To distinguish from the above change notifications, Barix Barionet sends the regular status messages as

- **`state,I/O,val`**

just like the answer to **`getio`** inquiries.

### State information upon TCP connect

When connecting to the TCP command port the Barionet will issue the following states (without sending any **`getio`** command).

- **`state,201,0<cr>`**
  **`state,202,0<cr>`**
  **`state,203,0<cr>`**
  **`state,204,0<cr>`**
  **`state,205,0<cr>`**
  **`state,206,0<cr>`**
  **`state,207,0<cr>`**
  **`state,208,0<cr>`**
  **`state,1,0<cr>`**
  **`state,2,0<cr>`**
  **`state,101,0<cr>`**
  **`state,102,0<cr>`**
  **`state,103,0<cr>`**
  **`state,104,0<cr>`**
  **`state,9,0<cr>`**

This speeds up the state polling process as a TCP connect is all you need to obtain the above status.
To obtain status of analog inputs or other I/O use the **`getio`** command as described above.
The mapping of I/O addresses is described in the next chapter 12 I/O Address Mapping.

# 12 I/O Address Mapping

The Barix Barionet can be accessed with standard Modbus/TCP commands, via TCP & UDP protocols as described in the previous chapter, or via CGI commands for setting and HTML tags for reading.

Access to Barix Barionet variables is done via I/O Bits (can be zero or one) and via I/O Words, which are 16 Bit values. All variables share a common data space so that the coils can be accessed by both bit (coil read) as well as word (holding register) commands. Reading a bit variable through a Word access command yields `0` or `1`. Reading a WORD variable through a bit access command yields zero or one for non-zero values.

## 12.1 I/O Address Map

### 1...100 Relay outputs (Read, Write)

| I/O | Output |
|---|---|
| 1 | Relay 1 |
| 2 | Relay 2 |
| 3...8 | reserved |
| 9 | RTS out on RS-232 |
| 10...100 | external |

| value | Function |
|---|---|
| 0 | Set output to inactive |
| 1 | Set output to active |
| 9999 | Revert (toggle) output |
| n | Pulse output for n*100ms, valid values 2-9998 |

If set with the register write command to `9999`, the output toggles. If set with the register write command to a value >1, sets pulse of duration in 100ms counts.

The addresses 3 to 8 are reserved for future internal use. Address 9 is mapped to the RS-232 RTS output signal and should only be addressed if HW flow control is disabled. The addresses 10 to 100 can be used to map external relays (e.g. Modbus module Barix R6) in a BCL application to enable the control via the API.

### 101...200 Digital outputs (Read, Write)

| I/O | Output |
|---|---|
| 101 | Output 1 |
| 102 | Output 2 |
| 103 | Output 3 |
| 104 | Output 4 |
| 105...109 | reserved |
| 110...200 | external |

| value | Function |
|---|---|
| 0 | Set output to inactive |
| 1 | Set output to active |
| 9999 | Revert (toggle) output |
| n | Pulse output for n*100ms, valid values 2-9998 |

If set with the register write command to `9999` , the output toggles. If set with the register write command to a value `>1`, sets pulse of duration in 100ms counts.

The addresses 105 to 109 are reserved for future internal use. The addresses 110 to 200 can be used to map external outputs (e.g. Modbus module Barix IO12) in a BCL application to enable the control via the API.

### 201...300 Digital inputs (Read only)

| I/O | Input |
|---|---|
| 201 | Input 1 |
| 202 | Input 2 |
| 203 | Input 3 |
| 204 | Input 4 |
| 205 | Input 5 |
| 206 | Input 6 |
| 207 | Input 7 |
| 208 | Input 8 |
| 209 | reserved |
| 210...300 | external |

| value | Function |
|---|---|
| 0 | Input is inactive |
| 1 | Input is active |

**Attention:**
The polarity (e.g. Low-active or high-active) can be changed per input in the I/O settings configuration.

The address 209 are reserved for future internal use. The addresses 210 to 300 can be used to map external inputs (e.g. Modbus module Barix IO12) in a BCL application to enable the control via the API.

### 301...400 Virtual I/O bits (Read, Write)

| I/O | Function |
|---|---|
| 301...400 | Virtual bit |

| value | Function |
|---|---|
| 0 | Bit is inactive |
| 1 | Bit is active |

The addresses 301 to 400 can be used to map virtual I/O bits (implemented in memory only) in a BCL application to talk to the BCL application via the API.

### 401...500 Digital input 32bit Counters (Read, Write)

| I/O | Function |
|---|---|
| 401 | Low 16bit Counter 1 |
| 402 | High 16bit Counter 1 |
| " | " |
| 415 | Low 16bit Counter 8 |
| 416 | High 16bit Counter 8 |
| 417... 500 | reserved |

| value | Function |
|---|---|
| 0...65535 | Count value |

Each of the digital inputs is fed into a 32bit counter (counts from 0 to 4'294'967'295) which counts every active input edge up to a frequency of 30 Hz (i.e. 15 pulses per second). When the Low 16bit Counter is filled (e.g. `65535` ) the next pulse will clear the counter and will increment the High 16bit Counter by one. To calculate the complete count use the formula:

- `Count=65536*High16bit+Low16bit`

The Low 16bit Counter register can be reset with a value `0` or preset with a value between `1` and `65535` by writing the value to the register which will alos clear the High 16bit Counter register (e.g. `0` ).

**Attention:**
After a reboot or power cycling of the Barionet the Low 16bit Counters can show a value of `1` or `2` depending on the I/O settings Pullup and Polarity . Before using the counters make sure to set them to a correct value (e.g. `0` ).

### 501...600 Analog input 9bit Values (Read only)

| I/O | Function |
|---|---|
| 501 | Analog input 1 Value |
| 502 | Analog input 2 Value |
| 503 | Analog input 3 Value |
| 504 | Analog input 4 Value |
| 505... 600 | Reserved |

| value | Function |
|---|---|
| 0...1023 | Analog value |

Inputs 1 to 4 are measured using a 9bit A/D conversion. The value registers can only be read (no write).

A value of `0` corresponds to a voltage of 0 VDC whereas a value of `1023` is corresponding to 5 VDC.

### 601...650 1-wire Temperature Sensor 12bit Values (Read only)

| I/O | Function |
|---|---|
| 601 | Temp. sensor 1 Value |
| " | " |
| 650 | Temp. sensor 50 Value |

| value | Function |
|---|---|
| 0...4095 | Temp. value |
| 4096 | No sensor |

1-wire sensors deliver a 12bit value (see Dallas data sheet for conversion information).

If a sensor is not present bit `13` is set ( `4096` ).

The value registers can only be read (no write).

### 651...700 1-wire Temperature Sensor unique HW-ID (Read only)

| I/O | Function |
|---|---|
| 651 | Address Sensor 1 |
| " | " |
| 700 | Temp. sensor 50 Value |

| value | Function |
|---|---|
| 0...4095 | Temp. value |
| 4096 | No sensor |

Each 1-wire sensors has a unique hardware ID (see Dallas data sheet for more information).

If a sensor is not present the register reads `0` .

The register holds a 8bit family code and a 24bit address and can only be read (no write).

The register is to be read byte wise from right to left. Here an example:

- **`getio,651`**

has returned:

- **`state,651,1970071592`**

The value **`1970071592`** is **`75 6C E8 28`** Hex and reads as Family Code = **`28`** and the ID = **`E8 6C 75`**.

### 701...1000 Virtual 16bit registers (Read, Write)

| I/O | Function |
|---|---|
| 701...1000 | Virtual 16bit registers |

| value | Function |
|---|---|
| 0...65535 | Register value |

The addresses 301 to 400 can be used to map virtual 16bit registers (Words, implemented in memory only) in a BCL application to talk to the BCL program or other protocol handlers via the API.

All unimplemented addresses read as zero, all write access attempts to unimplemented addresses yield an error.

# 13 Configuration Setup Record

## 13.1 EEPROM Organization

The lower 512 bytes of the EEProm (1024 bytes) are reserved for storing production, OEM and standard configuration parameters.

The upper 512 bytes of the EEProm can be used for storing custom information such as configuration parameters of your own BCL applications.

Accessing the EEProm (Read and Write) is possible via Setup CGI commands (complete range) as well as via BCL by using the Setup file handler STP (in chunks of 256 bytes).

### General Terms

The following terms apply for the setup record:

- IP addresses are always stored with the highest byte at the lowest address.
- Strings are coded in ASCII and terminated with 0x00. The Length includes the termination.
- Values are stored in little endian format (Intel, low byte first)
- All Values are integer.
- Signed values are stored in 2-complement.
- Unused bytes must be set to 0x00.

## 13.2 Setup Record Table

In the following table the column Pos shows the Position (starting with 0 ) in the 512 bytes of the EEProm , reserved for configuration.

The column DHTML Name shows the Name when used with Dynamic HTML within a setup CGI command.

The column L. shows the length in bytes.

The column Info/Link shows information or contains a link to a chapter with further information.

| Parameter | Pos | DHTML Name | L | Info/ link |
|---|---|---|---|---|
| IP Address | 0 | B0 - B3 | 4 | 5.1 |
| Product ID | 4 | | 1 | 0xBA for Barionet |
| Setup Version | 5 | | 1 | 0x02 for version 02 |
| Netmask | 8 | N0, N1, N2, N3B6 | 1 | The value is the count of the zero bits counted from the lowest byte. (e.g. 8 for 255.255.255.0) 5.1 |
| Reserved | 7 | | 1 | |
| Web server port | 8 | W8 | 2 | Default 0 = 80 5.4 |
| Lockdown mode | 10 | W10 | 2 | 5.4 |
| Gateway IP | 12 | B12-B15 | 4 | 5.1 |
| RS-232 Baud rate | 16 | B16 | 1 | 4=19200, 8=9600, 16=4800, 32=2400, 64=1200, 128=600 |
| RS-232 control | 17 | B17 | 1 | See next 6 rows |
| Stop bits | 17 | B17b7 | b | 0=1, 1=2 stop bits |
| Data bits | 17 | B17b6 | b | 0=8, 1=7 data bits |
| Parity control | 17 | B17b5 | b | 0=Off, 1=On |
| Parity mode | 17 | B17b4 | b | 0=Even, 1=Odd |

| Parameter | Pos | DHTML Name | L | Info/ link |
|---|---|---|---|---|
| Flow Control | 17 | B17b2 | b | 0=None, 1=RTS/CTS |
| Buffer flushing | 17 | B17b0 | b | 0=No, 1=Flush buffer |
| Local port TCP | 18 | W18 | 2 | 5.2 |
| Reserved SGW | 20 | W20 | 2 | Destination port |
| Reserved SGW | 22 | B22-B25 | 4 | Dest. IP Gateway |
| Disconnect Time out | 26 | B26 | 1 | 0=Disabled, 1...255 seconds |
| Reserved | 27 | | 5 | |
| RS-422/485 Baud rate | 32 | B32 | 1 | 4=19200, 8=9600, 16=4800, 32=2400, 64=1200, 128=600 |
| RS-422/485 control | 33 | B33 | 1 | See next 6 rows |
| Stop bits | 33 | B33b7 | b | 0=1, 1=2 stop bits |
| Data bits | 33 | B33b6 | b | 0=8, 1=7 data bits |
| Parity control | 33 | B33b5 | b | 0=Off, 1=On |
| Parity mode | 33 | B33b4 | b | 0=Even, 1=Odd |
| Type selection | 33 | B33b3 | b | 0=RS-485, 1=RS-422 |
| Buffer flushing | 33 | B33b0 | b | 0=No, 1=Flush buffer |
| Local port TCP | 34 | W34 | 2 | 5.2 |
| Reserved SGW | 36 | W36 | 2 | Destination port |
| Reserved SGW | 38 | B38-B41 | 4 | Dest. IP Gateway |
| Disconnect Time out | 42 | B42 | 1 | 0=Disabled, 1...255 seconds |
| Reserved | 43 | | 5 | |
| SNMP Trap receiver IP | 48 | B48-B51 | 4 | 5.5 |
| UDP info send | 52 | B52-B55 | 4 | 5.4 |
| UDP command port | 56 | W56 | 2 | 5.4 |
| UDP dest. port | 58 | W58 | 2 | 5.4 |

| Parameter | Pos | DHTML Name | L | Info/ link |
|---|---|---|---|---|
| UDP interval | 60 | W60 | 2 | 5.4 |
| TCP command port | 62 | W62 | 2 | 5.4 |
| Pullup configuration | 64 | B64 | 1 | Pullup Resistor for 8 Inputs (see below) |
| Pullup Input 1 | 64 | B64b0 | b | 0=On, 1=Off |
| Pullup Input 2 | 64 | B64b1 | b | 0=On, 1=Off |
| Pullup Input 3 | 64 | B64b2 | b | 0=On, 1=Off |
| Pullup Input 4 | 64 | B64b3 | b | 0=On, 1=Off |
| Pullup Input 5 | 64 | B64b4 | b | 0=On, 1=Off |
| Pullup Input 6 | 64 | B64b5 | b | 0=On, 1=Off |
| Pullup Input 7 | 64 | B64b6 | b | 0=On, 1=Off |
| Pullup Input 8 | 64 | B64b7 | b | 0=On, 1=Off |
| SNMP Trap Generation | 65 | B65 | 1 | Trap generation for 8 inputs (see below) |
| Trap Input 1 | 65 | B65b0 | b | 0=No, 1=Send Trap |
| Trap Input 2 | 65 | B65b1 | b | 0=No, 1=Send Trap |
| Trap Input 3 | 65 | B65b2 | b | 0=No, 1=Send Trap |
| Trap Input 4 | 65 | B65b3 | b | 0=No, 1=Send Trap |
| Trap Input 5 | 65 | B65b4 | b | 0=No, 1=Send Trap |
| Trap Input 6 | 65 | B65b5 | b | 0=No, 1=Send Trap |
| Trap Input 7 | 65 | B65b6 | b | 0=No, 1=Send Trap |
| Trap Input 8 | 65 | B65b7 | b | 0=No, 1=Send Trap |
| Input Polarity | 66 | B66 | 1 | Polarity for 8 Inputs Low Active for Contact Closures use High Active for Driving Voltage use (see below) |
| Polarity Input 1 | 66 | B66b0 | b | 0=Low, 1=High Act. |
| Polarity Input 2 | 66 | B66b1 | b | 0=Low, 1=High Act. |
| Polarity Input 3 | 66 | B66b2 | b | 0=Low, 1=High Act. |

| Parameter | Pos | DHTML Name | L | Info/ link |
|---|---|---|---|---|
| Polarity Input 4 | 66 | B66b3 | b | 0=Low, 1=High Act. |
| Polarity Input 5 | 66 | B66b4 | b | 0=Low, 1=High Act. |
| Polarity Input 6 | 66 | B66b5 | b | 0=Low, 1=High Act. |
| Polarity Input 7 | 66 | B66b6 | b | 0=Low, 1=High Act. |
| Polarity Input 8 | 66 | B66b7 | b | 0=Low, 1=High Act. |
| Input Usage A/D | 67 | B67 | 1 | Input 1...4 selection Either Analog values in Info send but no Traps or Digital state & Traps (see below) |
| Usage Input 1 | 67 | B67b0 | b | 0=Digital, 1=Analog |
| Usage Input 2 | 67 | B67b1 | b | 0=Digital, 1=Analog |
| Usage Input 3 | 67 | B67b2 | b | 0=Digital, 1=Analog |
| Usage Input 4 | 67 | B67b3 | b | 0=Digital, 1=Analog |
| SNMP Trap Repeat Time | 68 | W68 | 2 | 0=Disabled, 1...65535 seconds |
| I/O Protocol | 70 | B70 | 1 | 0=1-wire 1=Wiegand |
| Reserved | 71 | B71 | 9 | Reserved for OEM |
| Syslog Server | 80 | B80-B83 | 4 | 5.4 |
| NTP Server | 84 | B84-B87 | 4 | 5.6 |
| DNS Server | 88 | B88-B91 | 4 | 5.1 |
| Syslog Debug Level | 92 | B92 | 1 | 5.4 |
| Debug mode/flags | 93 | B93 | 1 | For Barix internal use only |
| Time Zone | 94 | B94 | 1 | Time zone in 30 minutes intervals (signed, +-24) |
| Reserved | 95 | B95 | 1 | DST rule flags |
| TCP command port Disconnect Time out | 96 | B96 | 1 | Inactivity Disconnect 0=None, 1...255 seconds |

| Parameter | Pos | DHTML Name | L | Info/ link |
|---|---|---|---|---|
| Reserved | 97 | | 3 | |
| BCL Program Name | 100 | S100 | 9 | Filename of BCL programm without .tok extension, 8 characters & 0x00 string termination |

# 14 Dictionary

**DHCP**

Short for Dynamic Host Configuration Protocol, a protocol used to assign an IP address to a device connected to a Network.

**IP**

Short for Internet Protocol, the IP is an address of a computer or other network device on a network using IP or TCP/IP. Every device on an IP-based network requires an IP address to identify its location or address on the network. Example: **`192.168.2.10`**
See also chapter 15 IP Address, Netmask etc.

**IPzator**

Barix IPzator technology is designed for the purpose that the Barix device can create its own IP address according to the network structure in case it can t receive one from your network. If DHCP, AUTOIP or BOOTP fail, IPzator will create an IP address within the subnet and test it (starting with **`x.x.x.168`** and if occupied incrementing by one). If the address works and is not being used by another device on the network, it will give the address to the Barix device.

**MAC address**

Abbreviation for Medium Access Control, a MAC is a unique address number formatted in hexadecimal format and given to each computer and/or network device on a computer network. Because a MAC address is a unique address a computer network will not have the same MAC address assigned to more than one computer or network device. Example: **`A1:B2:C3:D4:E5:F6`**

**Netmask**

A number used to identify a sub network so that an IP address can be shared on a LAN (Local Area Network). A mask is used to determine what subnet an IP address belongs to. An IP address has two components, the network address and the host address. For example, consider the IP address `150.215.017.009`. Assuming this is part of a Class B network, the first two numbers (`150.2.`) represent the Class B network address, and the second two numbers (`.017.009`) identify a particular host on this network.
The Netmask would then be `255.255.0.0` .
See also chapter 15 IP Address, Netmask etc.

**Ping**

Ping is a basic Internet program that lets you verify that a particular IP address exists and can accept requests.
Example: ping `192.168.2.10`

**Static IP**

A Static IP is a fixed IP address that you assign manually to a device on the network. It remains valid until you disable it.

**Telnet**

Telnet is a user command and an underlying TCP/IP protocol for accessing remote computers. On the Web, HTTP and FTP protocols allow you to request specific files from remote computers, but not to actually be logged on as a user of that computer. With Telnet, you log on as a regular user with whatever privileges you may have been granted to the specific application and data on that computer. Example: telnet `192.168.2.10`

# 15 IP Address, Netmask etc.

### IP Addressing

An IP address is a 32 bit value, divided into four octets of eight bits each. The standard representation is four decimal numbers (in the range of 0..255), divided by dots.

- Example: `192.2.1.123`

This is called decimal-dot notation. The IP address is divided in two parts: a network and a host part. To support different needs, five network classes have been defined. Depending on the network class, the last one, two or last three bytes define the host, while the remaining part defines the network. In the following text, `x` stands for the host part of the IP address.

### Class A network

- IP address `1.x.x.x` to `127.x.x.x`

Only 127 different networks of this class exist. These have a very large number of potential connected devices (up to 16'777'216)

- Example: `10.0.0.1` (network `10`, host `0.0.1`)

### Class B network

- IP address `128.0.x.x` to `191.255.x.x`

These networks are used for large company networks. Every network can consist of up to 65534 devices.

- Example: `172.1.3.2` (network `172.1`, host `3.2`)

### Class C network

- IP address **192.0.0.x** to **223.255.255.x**

Class C networks are most common and for smaller companies. These networks can consist of a maximum number of 254hosts.

- Example: **192.7.1.9** (network **192.7.1**, host **9**)

### Class D network

The remaining addresses **224.x.x.x - 239.x.x.x** are defined as Class D and are used as multicast addresses.

### Class E network

No addresses are allowed with the four highest order bits set to 1 (**240.x.x.x 254.x.x.x**).
These addresses, called "class E", are reserved.

### Network Address

The host address with all host bits set to "0" is used to address the network as a whole (for example in routing entries).

- Example: **192.168.0.0**

Network addresses can not be used as a host address!

### Broadcast Address

The address with the host part bits all set to 1 is the broadcast address, meaning for every host .

- Example: **192.168.0.255**

Broadcast addresses can not be used as a host address!

## IP Netmask

The Netmask is used to divide the IP address differently from the standard defined by the classes A,B and C.

Entering a Netmask, it is possible to define how many bits from the IP address are to be taken as the network part and how many bits are to be taken as the host part.

### Standard IP network Netmask:

| Class | Network bits | Host bits | Netmask |
|---|---|---|---|
| A | 8 | 24 | 255.0.0.0 |
| B | 16 | 16 | 255.255.0.0 |
| C | 24 | 8 | 255.255.255.0 |

### Netmask examples:

| Netmask | Host bits |
|---|---|
| 255.255.255.252 | 2 |
| 255.255.255.248 | 3 |
| 255.255.255.240 | 4 |
| 255.255.255.224 | 5 |
| 255.255.255.192 | 6 |
| 255.255.255.128 | 7 |
| 255.255.255.0 | 8 |
| 255.255.254.0 | 9 |
| 255.255.252.0 | 10 |
| 255.255.248.0 | 11 |
| . | . |
| . | . |
| 255.128.0.0 | 23 |
| 255.0.0.0 | 24 |

## Private IP Networks and the Internet

If your network is not connected to the Internet and there are no plans to make such a connection you may use any IP address you wish.

However if you are not connected to the Internet and have plans to connect to the Internet or you are connected to the Internet and want to operate your Barix Barionet on an intranet you should use one of the sub-networks below for your network. These network numbers have been reserved for such networks. If you have any questions about IP assignment ask your Network Administrator.

### Private IP networks by class:

| Class | Network |
|---|---|
| A | 10.x.x.x |
| B | 172.16.x.x |
| C | 192.168.0.x |

## Network RFC s

For more information regarding IP addressing see the following documents. They can be found on the Internet:

| RFC | Description |
|---|---|
| 950 | Internet Standard Subnetting Procedure |
| 1700 | Assigned Numbers |
| 1117 | Internet Numbers |
| 1597 | Address Allocation for Private Internets |

# 16 BIN / DEC / HEX conversion

Hexadecimal digits have values from 0..15, represented as 0..9 and as A (for 10) to F (for 15).

The following table can serve as a conversion chart:

### Bin /DEC / Hex Table

| Decimal | Binary | Hexadecimal |
|---|---|---|
| 0 | 0000 | 0 |
| 1 | 0001 | 1 |
| 2 | 0010 | 2 |
| 3 | 0011 | 3 |
| 4 | 0100 | 4 |
| 5 | 0101 | 5 |
| 6 | 0110 | 6 |
| 7 | 0111 | 7 |
| 8 | 1000 | 8 |
| 9 | 1001 | 9 |
| 10 | 1010 | A |
| 11 | 1011 | B |
| 12 | 1100 | C |
| 13 | 1101 | D |
| 14 | 1110 | E |
| 15 | 1111 | F |

To convert a binary value in a hexadecimal representation, the upper and lower four bits are treated separately, resulting in a two-digit hexadecimal number.

# 17 Legal Information



For information about our devices and the latest version of this manual please visit www.barix.com.

Barix AG
Seefeldstrasse 303
8008 Zurich

SWITZERLAND

Phone: +41 43 433 22 11
Fax: +41 44 274 28 49

Internet

web: www.barix.com

email: sales@barix.com

support: support@barix.com