



Technical data sheet

AGEDI

Document version: AGEDI-man-en
Data of last revision: 2012-07-30
Valid from firmware version v3.01.x

**ATTENTION! THIS CONFIDENTIAL DOCUMENT IS PROPERTY OF
NETRONIX SP. Z O. O.**

**DISTRIBUTION OF THIS DOCUMENT IN ANY WAY WITHOUT SPECIAL
PERMISSION OF ITS OWNER IS STRICTLY FORBIDDEN**

Content

1	Introduction.....	4
2	Wiring and electrical specification.....	5
2.1	Element location in housing.....	5
2.2	Signals in device screw connector.....	6
2.3	Electrical specifications.....	6
3	First start-up.....	7
3.1	Finding device in network with active DHCP server.....	8
3.1.1	Reference to device's network name.....	8
3.1.2	Finding device with Discoverer application.....	8
3.2	Connecting.....	9
4	“Discoverer” finding mechanism.....	10
5	Configuration panel.....	11
5.1	Log in to administrative panel.....	11
5.2	Network settings configuration.....	12
5.3	Configuration of security settings.....	14
5.4	Configuration of serial interface settings.....	16
5.4.1	Mode „Device monitor”.....	17
5.4.2	Mode „TCP/IP converter RAW”.....	17
5.5	Configuration of SNMP settings.....	18
5.6	Port input/output configuration.....	20
5.6.1	Configuration port selection.....	21
5.6.2	Port configuration.....	21
5.7	Monitored device settings configuration.....	22
5.7.1	Device and parameter selection configuration form.....	23
5.7.2	Device configuration form.....	23
5.7.3	Parameter configuration form.....	24
5.7.4	Configuration file of monitored device.....	26
5.8	Configuration files.....	27
6	Verification of monitored device state.....	27
6.1	MIB database read.....	27
6.2	Listening to SNMP trap notifications.....	28
6.3	Device state preview on WWW page.....	29
6.4	Communication status results interpretation.....	30
7	Firmware update.....	31
8	File system.....	32
9	Return to factory settings.....	33
10	Trap notifications „Specific” values.....	33
11	MIB file compatible with ASN.1 standard.....	34

Index of tables

Table 1:	Housing elements.....	5
Table 2:	Signals in screwed connector.....	6
Table 3:	Electrical parameters.....	6
Table 4:	Default network interface settings.....	7
Table 5:	Default user name and password settings.....	11
Table 6:	Default network interface settings.....	13

Table 7: Default settings of session parameters.....	15
Table 8: Default settings of parameters access list ACL.....	16
Table 9: Default serial interface settings.....	17
Table 10: Default SNMP settings.....	19
Table 11: Default device settings.....	23
Table 12: Default parameter settings.....	24
Table 13: Types of supported parameters.....	25
Table 14: “Specific” values send in notifications.....	33

Index of illustrations

Illustration 1: Device housing diagram.....	5
Illustration 2: Answer for ping request with host name.....	8
Illustration 3: Discoverer application.....	9
Illustration 4: Home page.....	10
Illustration 5: Log in form to configuration panel.....	12
Illustration 6: Welcome page on configuration panel.....	12
Illustration 7: Network settings configuration panel.....	13
Illustration 8: Security settings configuration panel.....	15
Illustration 9: Serial interface settings configuration panel.....	17
Illustration 10: SNMP settings configuration panel.....	19
Illustration 11: Configuration panel of I/O port.....	21
Illustration 12: Monitored device settings configuration panel.....	22
Illustration 13: Parameter and device selection form.....	23
Illustration 14: Device configuration form.....	23
Illustration 15: Device's parameter configuration form.....	24
Illustration 16: Configuration files upload/download panel.....	27
Illustration 17: MIB database preview via MIB Browser application.....	28
Illustration 18: Parameters results preview on WWW page.....	30
Illustration 19: Panel used to activate bootloader.....	32
Illustration 20: File system upload form.....	33

1 Introduction

AGEDI is device used to monitor parameters read from third party devices via serial interface with MODBUS RTU protocol. Device is able to cyclically read environment temperature measure by two temperature sensors from DS1920 family connected to 1-Wire interface. Device is also equipped with 8 I/O ports allowing for remote read/modification their state.

Agent is able simultaneously monitor maximally up to 4 devices with defined 120 parameters at each, and also analyze their state and for defined circumstances send notifications to monitoring devices called NMS.

Device is equipped with:

- Ethernet interface 10Mbit/s.
- Serial interface RS-232 configurable up to 230400 bps.
- Serial interface RS-485 configurable up to 230400 bps.
- 1-Wire interface used to read temperature measures.
- 8 digital I/O ports.
- Green LED diode as a device supply power indicator.
- Red LED diode as a status indicator of activity on serial interfaces RS-485 and RS-232.
- Return to default settings button.

Device features:

- WWW server allowing configuration, remote reset and device state preview.
- SNMP server
- SNMP server used to read conditions of monitored device with ability to send notifications to selected IP addresses form NMS Tramps devices list. Access to read measures are restricted by list of IP addresses of NMS devices with consideration of groups names to read and write operation.
- DHCP client.
- Support NBNS protocol.
- Mechanism of finding devices in network with dynamic network settings assignment.
- TCP ↔ Serial conversion server working interchangeably with machine used to monitor device state via MODBUS RTU.
- TFTP server for remote firmware update.
- ACL access control list to WWW and conversion TCP ↔ Serial servers to restrict access to device to selected IP addresses only.

2 Wiring and electrical specification

2.1 Element location in housing

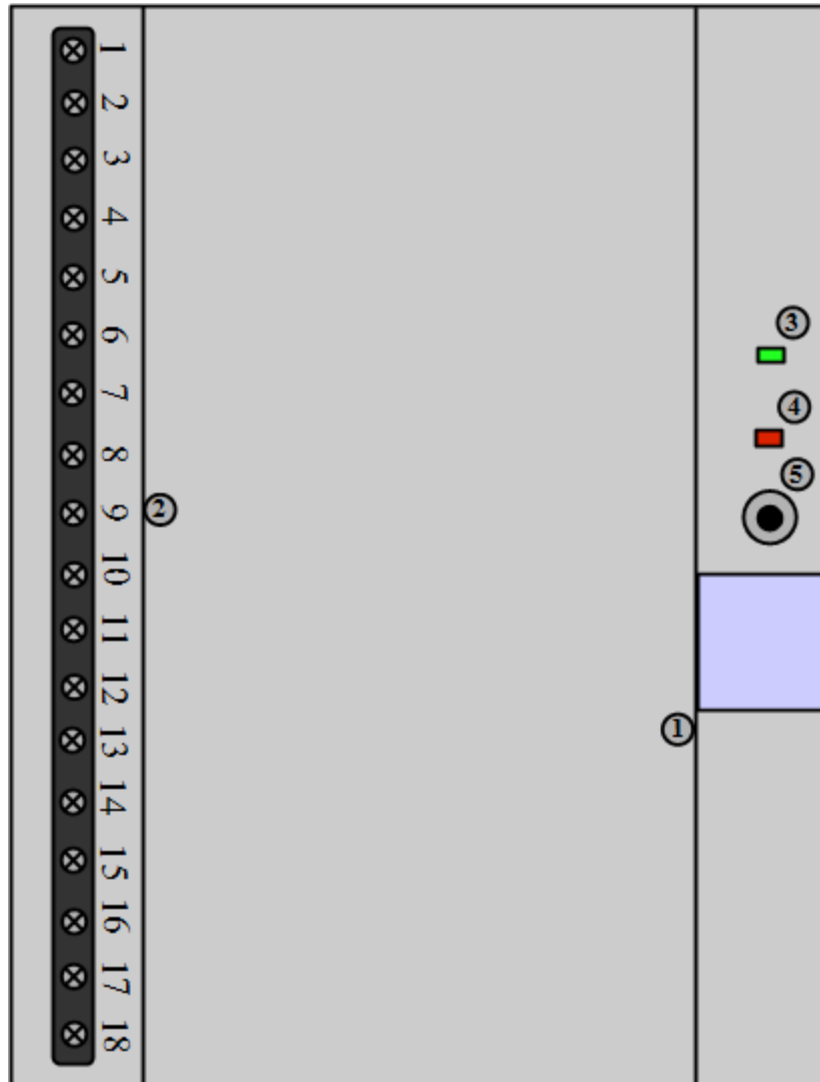


Illustration 1: Device housing diagram.

Table 1: Housing elements.

No.	Name	Description
1	Socket RJ45	Connection to Ethernet network.
2	Screw connector 18 ports	Power supply, RS-485 and RS-232 signals, 1-Wire, 8 I/O digital ports.
3	LED - Green	Diode indicating connection of power supply.
4	LED - Red	Diode indicating activity on serial interface.
5	Reset button	Return to factory settings button.

2.2 Signals in device screw connector

Table 2: Signals in screwed connector.

Pin no.	Name	Description
1	I/O1	Port input/output
2	I/O2	Port input/output
3	I/O3	Port input/output
4	I/O4	Port input/output
5	I/O5	Port input/output
6	I/O6	Port input/output
7	I/O7	Port input/output
8	I/O8	Port input/output
9	RS232 - TXD	Output
10	RS232 - RXD	Input
11	RS-485 – „A”	Input/output
12	RS-485 – „B”	Input/output
13	GND	Ground reference point for serial interfaces
14	GND	Ground reference point for serial interfaces
15	1-Wire	Data line of 1-Wire
16	+5V	1-Wire devices power supply output
17	GND	Supply power input
18	+V	Supply power input

2.3 Electrical specifications

Table 3: Electrical parameters

No.	Symbol	Characteristic	Value			Units	Comment
			Min.	Typ.	Max.		
Power supply parameters							
1	VPower	Supply voltage	+7.0	-	+26.0	V	
2	IPower	Supply current	60	80	300	mA	Depending from load of serial interfaces and supply voltage level
Interface RS-232 parameters							
3	Vrs232_out	Voltage range on output line TXD	-	+/-10.0	-	V	Without load
4	Vrs232_in	Maximum allowed voltage range on	-	-	+/-30.0	V	

No.	Symbol	Characteristic	Value			Units	Comment
			Min.	Typ.	Max.		
		Input line RXD					
Interface RS-485 parameters							
5	Vrs485_out	Differential output voltage range abs(A,B)	-	3.0	-	V	
6	Vrs485_in	Differential input voltage range A, B	-7.5	-	+12.5	V	
I/O ports parameters							
7	Rs	Serial resistance	90	100	110	Ω	
8	Ru	Pull up resistance	4,23	4,7	5,17	kΩ	Port pull up voltage level 3,3V
10	Imax	Maximum output current	-2,0	-	2,0	mA	
11	Vih	Input voltage high	1,6	-	5	V	
12	Vil	Input voltage low	0	-	0,45	V	
13	Ipu	Weak pull up current	0,5	-	0,75	mA	At Vpin = 0V

3 First start-up

A device delivered is equipped with factory settings. The Ethernet interface has been set by default using following values.

Table 4: Default network interface settings.

Parameter name	Description	Value
MAC	Hardware device address.	00:04:A3:XX:XX:XX ⁽¹⁾
Host Name	Host network name.	AGEDI
DHCP	Dynamic network configuration client.	Enabled
IP Address	Network address.	10.0.0.205 ⁽²⁾
Gateway	Gateway address.	10.0.0.1 ⁽²⁾
Subnet Mask	Subnet mask.	255.255.255.0 ⁽²⁾
Primary DNS	First DNS server address.	10.0.0.1 ⁽²⁾
Secondary DNS	Second DNS server address.	10.0.0.1 ⁽²⁾

Note:

- (1) – Unique value of address MAC for each of devices and can not be changed during process of configuration.
- (2) – Setting is valid when DHCP server are inactive in network, or client of dynamic configuration are disabled.

A procedure for establishing a connection with SNMP agent is different depending on network to which the agent is connected.

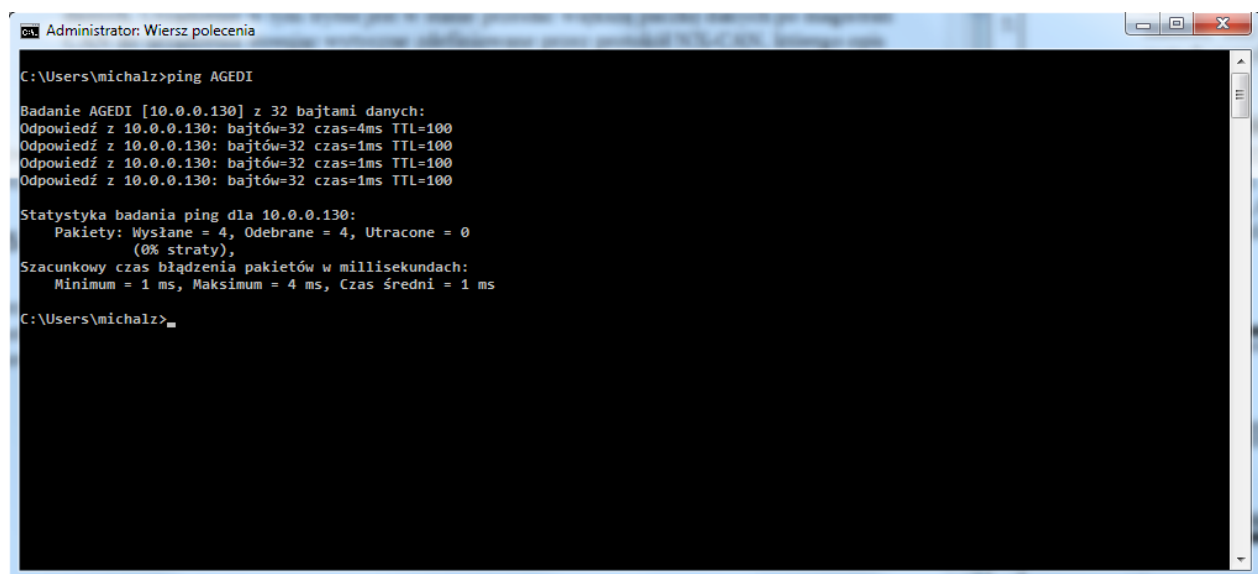
If a device is to be connected to Ethernet network, and its DHCP server is active, additional measures should be taken to find a device connected to network. If our network does not have an active DHCP server or network connection with PC computer is arranged in one-to-one way, you can omit successive sub-point “ searching the device with an active DHCP server in network”. If the agent is connected to network without DHCP server active, special care should be taken as the network settings will comply the device settings.

3.1 Finding device in network with active DHCP server

In case of network with dynamic assignment of the device network configuration, we are not able to anticipate which configuration will be set for given agent without defining the additional rules for sever which sets the configurations. There are two methods to verify IP address which given device has.

3.1.1 Reference to device's network name

The AGEDI provides the NetBios Name service, thanks to it we can establish a connection with a device using its network name. If network name is known („AGEDI” by default), it will be possible to send ping packet with network name of the device, to receive information on IP address assigned.



```
Administrator: Wiersz polecenia
C:\Users\michalz>ping AGEDI

Badanie AGEDI [10.0.0.130] z 32 bajtami danych:
Odpowiedź z 10.0.0.130: bajtów=32 czas=4ms TTL=100
Odpowiedź z 10.0.0.130: bajtów=32 czas=1ms TTL=100
Odpowiedź z 10.0.0.130: bajtów=32 czas=1ms TTL=100
Odpowiedź z 10.0.0.130: bajtów=32 czas=1ms TTL=100

Statystyka badania ping dla 10.0.0.130:
    Pakiety: Wysłane = 4, Odebrane = 4, Utracone = 0
            (0% straty),
Szacunkowy czas błędzenia pakietów w milisekundach:
    Minimum = 1 ms, Maksimum = 4 ms, Czas średni = 1 ms

C:\Users\michalz>
```

Illustration 2: Answer for ping request with host name.

3.1.2 Finding device with Discoverer application

A device has a mechanism integrated for fast finding it in network, so we can get the IP address which has been assigned to it dynamically.

To locate the agent, use the Discoverer software of NETRONIX company. The program is intended for the .NET 3.5 platform, and in case of old version of the platform, a start-up error will

occur. Make sure that the platform has been installed at workplace at which the program is to be run.

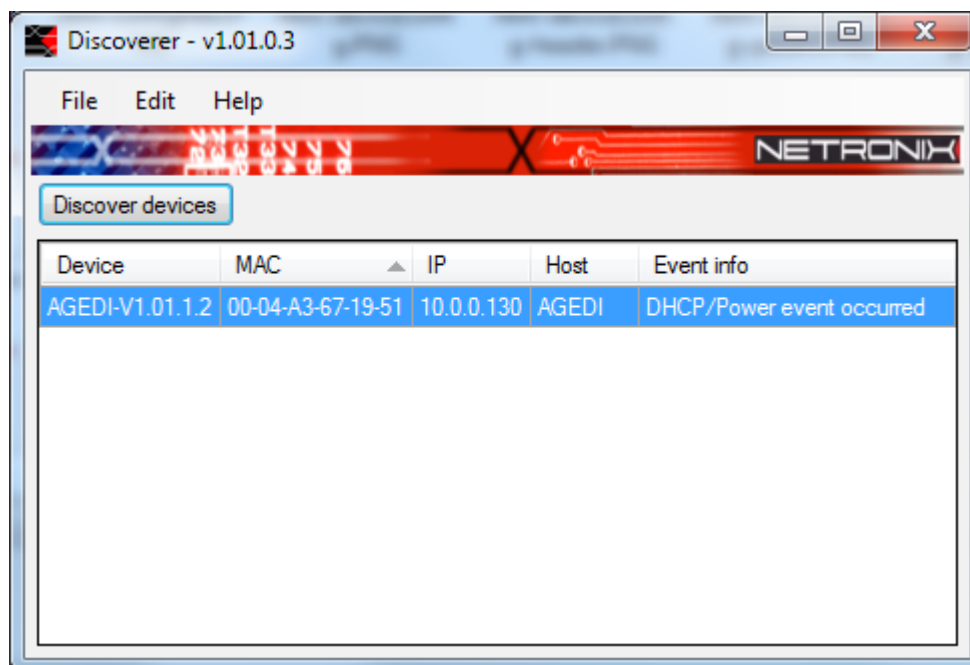


Illustration 3: Discoverer application.

To find a NETRONIX device in local network, click with the left mouse button the „Discover devices” key in the top left corner of the program window. In table of program window all network NETRONIX devices will be displayed in table of program window which are active in the network. In a list of devices detected, there is information such as IP address, type, physical MAC address and network name of device.

If a device we are searching does not show its presence, but this device has been connected properly (LED which signals “ON” state of a supply and green LED in RJ45 terminal are lit), we can suppose that network settings of the device have been modified and do not allow data to be transmitted in a local network.

A procedure for finding of NETRONIX devices in network will be described later in details in chapter: „Finding devices with Discoverer application”.

3.2 Connecting

If a device already has IP address which is known, we can begin to establish a connection. To do this, open any internet browser and then input IP address of the device or its network name in the address bar.

In browser window the device home page will be displayed with information such as firmware version, compilation date and monitored devices data. In top right corner of home page there is a link to login page. Through this page we can get to configuration panel of the device.

NETRONIX [Login](#)

Netronix SNMP Agent

Status!

Firmware Version: AGENT-SNMP-v3.01.1.1
Build Date: Jun 20 2012 11:52:45

Below you'll see the current status of the devices monitored by Agent. Data viewed on this page will refresh in 60 second period. To achive higher time resolution you will have to refresh page manual clicking refresh button in your browser.

Temperature sensors:

Sensor ID	Temperature mesasurement
229F1B13000000BC	26.600
28AB58B200000087	26.500

IO ports:

Port direction	State
Input	High
Input	High
Input	High
Input	High
Input	High
Input	High
Input	High
Input	High

Device 1 status:

State	Device Name	Address	Status
Disabled	-	-	-

Device 2 status:

State	Device Name	Address	Status
Disabled	-	-	-

Device 3 status:

State	Device Name	Address	Status
Disabled	-	-	-

Device 4 status:

State	Device Name	Address	Status
Disabled	-	-	-

Copyright © 2012 Netronix

Illustration 4: Home page.

4 “Discoverer” finding mechanism

A mechanism of finding the devices in network is based on constant listening to the UDP on 30303 port. To receive information on the device, simply send to given port a broadcast message with “Discover” content in front of data packet. After receiving of such packet, the devices send UTP responses to sender to port from which the query has been sent.

A device can send broadcast messages automatically to the 30303 port in case of two following events appear:

- Reset of a device.
- Network device settings has changed as a result of receiving the parameters of new configuration from DHCP server.

In case the one of above events appears, an information on the event which caused packet sending is added in the Discover program.

Automatic sending of messages with information on event occurrence has an advantage that in this case there is no need to query the device continuously, and messages are sent in broadcast way, so they can reach the listening programs even in case the network settings are not configured correctly.

5 Configuration panel

To make settings, a user should pass verification procedure. This procedure and individual configuration panels are described in subsequent sections of the chapter.

5.1 Log in to administrative panel

To make settings, a user should pass verification procedure. The configuration panel is protected with password and user name. To go to login page:

- Click “Login” link on the top right corner of the device home website.
- In the address field, input „login.htm” file name.
- In the address file, input a path to any file which is in “protect” catalog e.g.: „protect/index.htm”. When server detects that browser (from which endeavor to get to “protect” catalog occurs) did not send an actual session identifier with the query, it will re-route the browser to the login website automatically.

Table 5: Default user name and password settings.

Parameter name	Default value
Password	„1234”
Login	„admin”

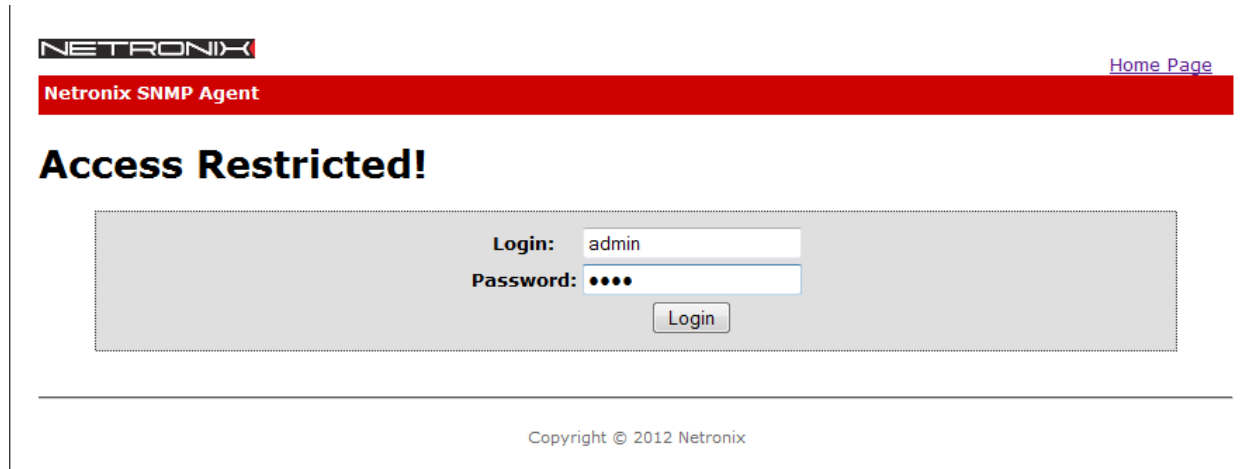


Illustration 5: Log in form to configuration panel.

If a user completes the form by entering the login website password and login properly, the server as a response for the query will send the session identifier and re-route the browser to the “protect/index.htm” welcome page of configuration panel.



Illustration 6: Welcome page on configuration panel.

In the same time, one person can be logged only, so it is important to finish the session by clicking the “logout” link, after finishing the operation of configuration panel. If the session is not closed properly, server will complete it, when maximal inactivity time elapses. We can set this time in the „Security” tab of administration panel. Time default value is 5 minutes.

5.2 Network settings configuration

We can configure network settings in the „Network” tab of administration panel. Starting the configuration be sure that after having confirmed new settings, we can re-connect with a

device. If settings entered cause that establishing of connection is not possible, and there is no other method known, we can recover network settings to their factory defaults. But, it will remove other changes done during device configuration.

NETRONIX [Logout](#)

Netronix SNMP Agent

Network Configuration

This page allows the configuration of the device's network settings.

CAUTION: Incorrect settings may cause the device to lose network connectivity. If provided settings make device inaccessible you will be able to restore factory settings with reset button.

Enter the new settings for the device below:

MAC Address: 00:04:A3:67:19:81
Host Name: AGENT-SNMP
 Enable DHCP
IP Address: 10.0.0.130
Gateway: 10.0.0.1
Subnet Mask: 255.255.255.0
Primary DNS: 10.0.0.3
Secondary DNS: 10.0.0.3

Copyright © 2012 Netronix

Illustration 7: Network settings configuration panel.

Table 6: Default network interface settings.

Parameter name	Description	Value
MAC	Hardware device address.	00:04:A3:XX:XX:XX ⁽¹⁾
Host Name	Host network name.	AGEDI
DHCP	Dynamic network configuration client.	Enabled
IP Address	Network address.	10.0.0.205 ⁽²⁾
Gateway	Gateway address.	10.0.0.1 ⁽²⁾
Subnet Mask	Subnet mask.	255.255.255.0 ⁽²⁾
Primary DNS	First DNS server address.	10.0.0.1 ⁽²⁾
Secondary DNS	Second DNS server address.	10.0.0.1 ⁽²⁾

Note:

- (1) – Unique value of address MAC for each of devices and can not be changed during process of configuration.
- (2) – Setting is valid when DHCP server are inactive in network, or client of dynamic configuration are disabled.

In case the DHCP server is enabled, the parameters in the form will conform to values obtained from the server and with which a device actually operates. It means that, if a device has operated until now with an active DHCP client and obtained a configuration from the server, this configuration will be returned to the form by the device. Otherwise, when DHCP client is disabled, the device will return network parameters, which are stored in non-volatile memory and are being used as alternative configuration in case of lack of DHCP server in network.

„Reset Config” form buttons restores parameter values from before editing.

New network settings will be valid after next switch-on of a device, so it is recommended, to make a remote reset of a device just after configuration modifications have been done.

5.3 Configuration of security settings

A device is equipped with two levels of security against unauthorized access. First level is an access restriction to configuration panel on website, by providing an authentication session basing on login and password of a user. Second kind of security is the ACL list with addresses accepted by web server and TCP ↔ RS conversion server (if it is enabled). Connections from address which not present in the list are rejected immediately just after a connection has been rejected. Operation of address verifying from ACL list can be deactivated by assignment to all IP addresses on IP list zero values („0.0.0.0”). The ACL list does not confine connections with SNMP server, because this server has its own list of NMS addresses which can have an access to data stored in MIB base.

The configuration of security settings are available in the „Security” tab of configuration panel. New parameters should be entered with special care, just not to have gained a restricted access to configuration panel. Otherwise, restoring of physical settings of the device will be necessary.

NETRONIX [Logout](#)

Netronix SNMP Agent

Network
Security
Serial
SNMP
I/O Ports
Device Config
Configuration Files
Bootloader
Reboot

Security Configuration

This page allows the configuration of the device's security settings.

CAUTION: Incorrect settings may cause the device will be inaccessible for some users and network devices. If provided settings make device inaccessible you will be able to restore factory settings with reset button.

Authentication

Login and password are use for restrict configuration pages form unwanted persons. It is important to know that only one user can be loged in at a time. So when you finish your job remember to logout before closing the web browser. Otherwise session will stil be active and no one will be able to log in, until idleness timeout will elapse or device will be reset.

Enter the new settings for the device below:

Login:
Password:
Password*:
Timeout time [min.] (max. 255):

Access control list

This form allows to configure list of network device IP address, which are able to connect with this device. Other connections initiated beyond list will instantly disconnected. If list is empty this functionality will be disabled.

Enter the new settings for the device below:

ACL 1:
ACL 2:
ACL 3:
ACL 4:

Copyright © 2012 Netronix

Illustration 8: Security settings configuration panel.

The main form in the panel is used for configuring the parameters of user authorization session. Entering and saving the settings done in this form causes, that the changes will be entered immediately without necessity to reset a device.

Table 7: Default settings of session parameters.

Parameter name	Description	Default value
Login	User name.	„admin“ ⁽¹⁾

Parameter name	Description	Default value
Password	User password.	„1234“ ⁽¹⁾
Password*	Confirmation of new password.	-
Timeout time	Session inactivity time, after which session will be automatically closed, if in configuration panel weren't made any moves.	5 minutes

Note:

(1) - Up to 8 characters

The bottom form in the panel is used for configuring the settings of access control list. Parameter values which have been modified and saved during this configuration will be valid in a moment of next reset of a device.

Table 8: Default settings of parameters access list ACL.

Parameter name	Description	Default value
ACL1	Host's 1 IP address, which may open connection with device.	0.0.0.0
ACL2	Host's 2 IP address, which may open connection with device.	0.0.0.0
ACL3	Host's 3 IP address, which may open connection with device.	0.0.0.0
ACL4	Host's 4 IP address, which may open connection with device.	0.0.0.0

5.4 Configuration of serial interface settings

A device is equipped with two serial interfaces RS-485 and RS-232. The interfaces are connected in the way that data which outputs from the SNMP agent inputs to the TX RS-232 and RS-485 data lines at once. Data approaching from serial buses are separated from each other, so data received from the RS-485 is not transmitted to RS-232 and the other way round.

The serial port can perform two functions depending on configuration method used. First mode of operation is used for gathering data from devices connected to bus by using the MODBUS RTU protocol. The second mode and the third one allow to send data from defined TCP port to serial port. One function only from functions of serial port can be active in same the moment.

The serial port configuration is available in the “Serial” tab of configuration panel.

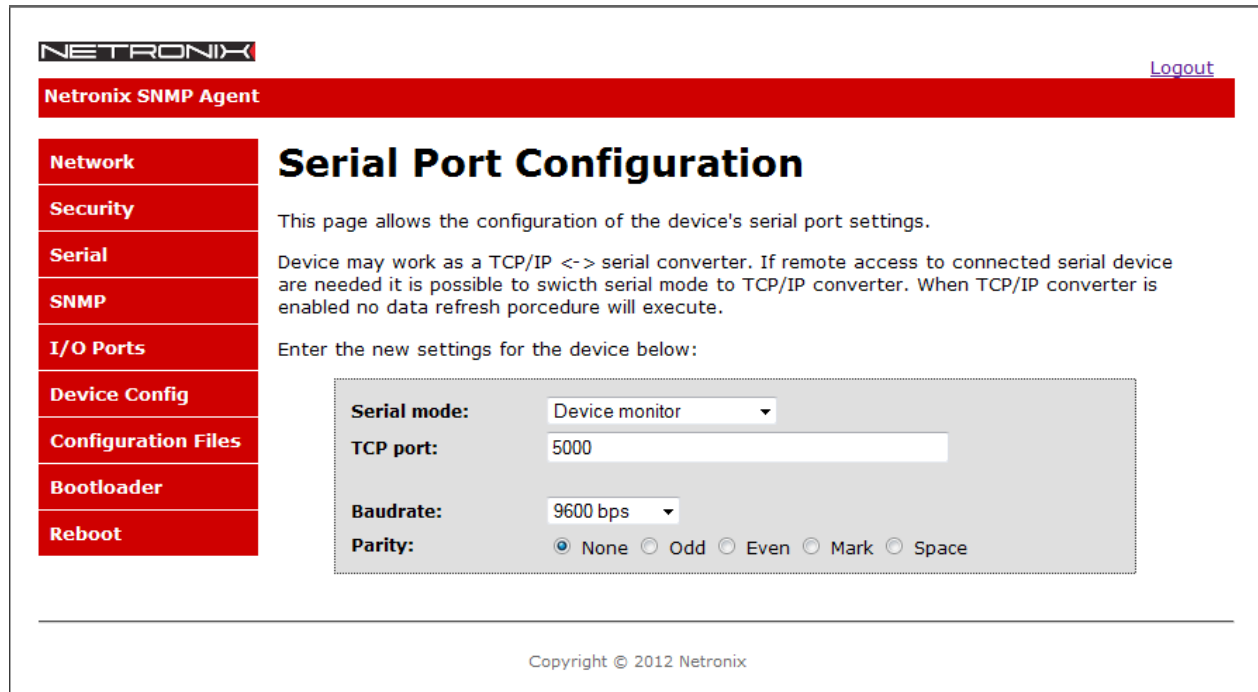


Illustration 9: Serial interface settings configuration panel.

Table 9: Default serial interface settings.

Parameter name	Description	Default value
Serial mode	Work mode of serial interface.	„Device monitor”
TCP Port	TCP port number of conversion server.	5000 ⁽¹⁾⁽²⁾
Baudrate	Data transmission speed of serial interface.	9600 bps
Parity	Parity transmission bit of serial interface.	None

Note:

- (1) - Server is active only when „Serial mode” is set in „TCP/IP converter RAW”.
- (2) - Change of this parameter need device reset.

5.4.1 Mode „Device monitor”

A „Device monitor” mode is a basic one for operation of the agent. In this mode, the agent uses serial port to query on the device status by means of MODBUS RTU protocol (functions no. “1” and “3”), and conversion between TCP port and serial interface is nonactive. In this mode, a temperature measurement is performed via sensors connected to the 1-Wire bus.

5.4.2 Mode „TCP/IP converter RAW”

A „TCP/IP converter RAW” is used for transmitting data directly from configured TCP port to a serial port and the other way round. For this mode, no additional drivers are needed. To take the control over the agent serial port, an user can use any program providing connection option to a TCP port. Please note, that conversion server verifies, if the address from which given connection has been established is mentioned in a list of ACL addresses. If the list is active and


device address which is trying to connect is not defined in it, the device breaks the connection immediately and do not allow for unauthorized persons to take control over the serial port.

In this mode, temperature measurement as well as cyclic parameter read-out of devices is nonactive.

5.5 Configuration of SNMP settings

The configuration of the SNMP agent parameters is available in the “SNMP” tab of configuration panel. In this panel, we can make following settings:

- Refreshing time of data being read from devices.
- IP address list of NMS devices which can be read from and save to MIB base of agent.
- IP address list of trap receivers to which messages should be sent in case the defined events occur.
- Community name with which messages are to be sent.
- Community names which have the read-out rights of parameters from MIB base.
- Community names which have the save rights of parameters into MIB base.


Logout

Netronix SNMP Agent

Network
 Security
 Serial
 SNMP
 I/O Ports
 Device Config
 Configuration Files
 Bootloader
 Reboot

SNMP Community Configuration

Read/Write Community String configuration for SNMPv2c Agent.

Configure multiple community names if you want the SNMP agent to respond to the NMS/SNMP manager with different read and write community names. If less than three communities are needed, leave extra fields blank to disable them.

Refresh Time [s]:

NMS 1:

NMS 2:

NMS 3:

NMS 4:

NMS 5:

NMS 6:

NMS 7:

NMS 8:

NMS Trap 1:

NMS Trap 2:

NMS Trap 3:

NMS Trap 4:

NMS Trap 5:

NMS Trap 6:

NMS Trap 7:

NMS Trap 8:

Trap Comm :

Read Comm1 :

Read Comm2 :

Read Comm3 :

Write Comm1:

Write Comm2:

Write Comm3:

Copyright © 2012 Netronix

Illustration 10: SNMP settings configuration panel.

Table 10: Default SNMP settings.

Parameter name	Description	Default value
Refresh Time	Period of time in which data collected from monitored devices should be refreshed.	10 s
NMS 1 ... NMS 8	NMS hosts IP addresses which has access to database.	0.0.0.0 ⁽¹⁾

Parameter name	Description	Default value
NMS Trap 1 ... NMS Trap 8	IP addresses of devices to which should be send notification in case defined event came out.	0.0.0.0
Trap Comm	Community name with which notifications should be send.	„public”
Read Comm1	Community name 1 authorize to read from MIB base.	„private”
Read Comm2	Community name 2 authorize to read from MIB base.	„read”
Read Comm3	Community name 3 authorize to read from MIB base.	„public”
Write Comm1	Community name 1 authorize to write to MIB base.	„private”
Write Comm2	Community name 2 authorize to write to MIB base.	„write”
Write Comm3	Community name 3 authorize to write to MIB base.	

Note:

- (1) - When all IP addresses are zeroed, access to database is unrestricted.

5.6 Port input/output configuration

Depending on operation direction, the input/output ports can perform remote control function of devices, when they are set as an output or can be used for monitoring, when they are set as an input. Configuration of operation direction for given port can be performed from the configuration page level as well as by means of SNMP protocol. Main difference is that settings which are sent to a device using the SNM protocol will be of momentary nature. Settings entered by means of port configuration panel will be initial ones which are assigned during reset of a device. The I/O ports can be a source of interruptions depending on configurations connected with low and high state or port state change. Analysis of the port status value is performed when this port is configured as an input only.

Configuration panel of I/O ports is available in the “I/O Ports” tab of configuration panel.

Illustration 11: Configuration panel of I/O port.

5.6.1 Configuration port selection

Choose from “I/O Index” port selector in the top form of configuration page, a port which is to be configured. Pushing „Submit” button of that form will cause that information will be sent to a device, which port is to be configured. The device responds by returning the port configuration which is placed in the bottom form of the page. The „Trap Receivers” flags inform an user which IP addresses of trap receivers will obtain information, in a moment, when an event will be generated on input port which will cause information sending. Contrary to other settings described in following sections of this documentations, the flags are assigned to all ports simultaneously.

5.6.2 Port configuration

The lower form of configuration panel displays actual settings of a port selected. The most important parameter is the port operation direction („Port Direction”). When we set this parameter as an „Input”, the port will read state on input terminal of a device. The „Initial State” parameter will be ignored and the „State Changed”, „State Low” and „State High” flags will mean “input” in operation. When we set one of mentioned flag, and the event we defined occurs, a message will be generated, and sent to the trap receivers under IP addresses defined formerly. When we set the port direction as an input, operation direction is changed immediately and its state is set in accordance with the „Initial State” field. This value will be assigned to an output during next start-up of the device.

5.7 Monitored device settings configuration

Setting configuration of queried devices is available in „Device Config” of configuration panel.

NETRONIX [Logout](#)

Netronix SNMP Agent

Network
Security
Serial
SNMP
I/O Ports
Device Config
Configuration Files
Bootloader
Reboot

Monitored Device Configuration

This page allows the configuration of the devices supported by SNMP agent.

Upload Configuration File:

To download configuration file for this device use mouse context menu on following link:
[Configuration File.](#)

Enter the new settings for the device below:

Device Index: Dev 1 ▾
Parameter Index: 0

Enable Device
Modbus Addr: 0
Timeout [ms]: 300
Delay [ms]: 20
Name:
Traps Receivers: IP1 IP2 IP3 IP4 IP5 IP6 IP7 IP8
Param Count: 0

Param Type: Disabled ▾
Register Address: 0
Name:
Traps Events: Value Changed
 Value Above Limit Value Under Limit
Value Mask: h00000000
Value Min: 0
Value Max: 0

Copyright © 2012 Netronix

Illustration 12: Monitored device settings configuration panel.

Configuration of monitored devices has been divided into three forms, which will be described in details in following sections of the chapter.

5.7.1 Device and parameter selection configuration form

The AGEDI can monitor up to 120 parameters of four devices connected to serial bus. Before start of configuring, a user must choose which parameter and of what device he wants to configure. Then he can start the configuration process.

Illustration 13: Parameter and device selection form.

To choose defined device, a user should set required „Device Index” corresponding to configured device a confirm selection by pressing „Submit” button located on the left bottom corner of the form. After confirming the selection, the device index will be sent to the device, and actual configuration of the device will be displayed in the form below as a response.

To choose defined parameter, a user should input required index number of parameter by selecting it from range 0 to 119 in the „Parameter Index” field, and next confirm the selection by pressing the „Submit” button. To go to the next or previous parameter, use the „Next” or „Previous” buttons, located on bottom of the form. As a response for parameter changing, the device will return actual parameter setting by placing it in the lowest form of the panel.

5.7.2 Device configuration form

Basic parameters of device are the device name and the MODBUS protocol address, which are used for communication via serial bus. In the form described above, we can define also a maximum time of waiting for a response and a delay time between last response of the device and subsequent query. The maximum time of waiting for a response can be defined for each device individually, but delay time between the last response and successive query is determined by slowest device and cannot be lower than highest time set for all devices.

Illustration 14: Device configuration form.

Table 11: Default device settings.

Parameter name	Description	Default value
Enable Device	Flag activating read of device parameters.	Inactive

Parameter name	Description	Default value
Modbus Addr	Device address used in MODBUS RTU protocol in communication with device.	0
Timeout	Maximum wait time for answer from device.	300 ms
Delay	Minimum time between last answer and next request.	20 ms
Name	Device name.	Empty
Trap Receivers	Flags activating corresponding IP addresses trap receivers defined in NMS Tram 1..8 list in SNMP configuration panel.	Inactive ⁽¹⁾
Param count	Number of active parameters assigned to configured device.	0 ⁽²⁾

Note:

- (1) - If NMS Trap address corresponding given flag is zeroed (0.0.0.0), then none notification will be send.
- (2) - Read only parameter.

5.7.3 Parameter configuration form

The first step in configuration of given parameter is choosing its type. The device supports 15 parameter types actually, which will be described in following sections of the chapter. The next step is to define correctly an address of register assigned to this parameter in device and its name. If such request appears, check please in the form on which event the agent is to generate messages and define generation conditions of the messages.

The screenshot shows a configuration form with the following elements:

- Param Type:** A dropdown menu currently showing "Disabled".
- Register Address:** A text input field containing the value "0".
- Name:** A text input field that is currently empty.
- Traps Events:** Three checkboxes: "Value Changed" (unchecked), "Value Above Limit" (unchecked), and "Value Under Limit" (unchecked).
- Value Mask:** A text input field containing the hexadecimal value "h00000000".
- Value Min:** A text input field containing the value "0".
- Value Max:** A text input field containing the value "0".
- Buttons:** Two buttons at the bottom: "Set" and "Reset Config".

Illustration 15: Device's parameter configuration form.

Table 12: Default parameter settings.

Parameter name	Description	Default value
Param Type	Parameter type.	Disabled
Register Address	Parameter's address in device. This address is directly used in MODBUS RTU transmission.	0
Name	Parameter name.	Empty
Traps Events	Flags activating notifications in case specific circumstances	Inactive

Parameter name	Description	Default value
	occur: „ ValueChanged ” – Notification is generated in case of value change occur. Event may be generated for each type of variables except floating point types. „ Value Above Limit ” – Upper limit of value has been crossed. Flag is ignored for bit type variables. „ Value Under Limit ” – Lower limit of value has been crossed. Flag is ignored for bit type variables.	
Value Mask	Mask covering changes of bits in parameter's value. To generate event notifying about value change, value of parameter should change on bit fields on witch value of mask fields is set on 1.	0x00000000(hex)
Value Min	Lower value limit crossing witch should trigger generation of notification.	0
Value Max	Upper value limit crossing witch should trigger generation of notification.	0

Because there are many different types of parameters implemented, the values are represented in MIB base in form of the ASCII character sequence. Presentation manner of values depends on type of parameter tested e.g. bit values will be presented by means of the flag sequence with a letter “b” in front. Integer parameters are presented in decimal form with “-” sign in front, when MSB value is equal „1” and parameter is of integer type with a sign.

Table 13: Types of supported parameters.

Parameter type	MODBUS RTU function	Register count	Description	Generated notifications
Bit	1	1	Parameter is bit flag.	- Change of value
Byte	3	1	Parameter is 8 bit number with a sign.	- Change of value - Lower limit crossed - Upper limit crossed
UByte	3	1	Parameter is 8 bit number without a sign.	- Change of value - Lower limit crossed - Upper limit crossed
Word	3	1	Parameter is 16 bit number with a sign.	- Change of value - Lower limit crossed - Upper limit crossed
UWord	3	1	Parameter is 16 bit number without a sign.	- Change of value - Lower limit crossed - Upper limit crossed
Long	3	2	Parameter is 32 bit number with a sign.	- Change of value - Lower limit crossed - Upper limit crossed
ULong	3	2	Parameter is 32 bit number without a sign.	- Change of value - Lower limit crossed - Upper limit crossed
8Bit_package	3	1	Parameter is representing	- Change of value

Parameter type	MODBUS RTU function	Register count	Description	Generated notifications
			a pack of 8 bit flags.	
16Bit_package	3	1	Parameter is representing a pack of 16 bit flags.	- Change of value
32Bit_package	3	2	Parameter is representing a pack of 32 bit flags.	- Change of value
Float	3	2	Parameter is representing floating point number in device registers in byte sequence 2,1,4,3.	- Lower limit crossed - Upper limit crossed
Float_B	3	2	Parameter is representing floating point number in device registers in byte sequence 4,3,2,1.	- Lower limit crossed - Upper limit crossed
Float_BB	3	2	Parameter is representing floating point number in device registers in byte sequence 3,4,1,2.	- Lower limit crossed - Upper limit crossed
Float_L	3	2	Parameter is representing floating point number in device registers in byte sequence 1,2,3,4.	- Lower limit crossed - Upper limit crossed
Float_LB	3	2	Parameter is representing floating point number in device registers in byte sequence 2,1,4,3.	- Lower limit crossed - Upper limit crossed

5.7.4 Configuration file of monitored device

If all parameters have been configured properly, you can read configuration file of a device defined just now, and then save or duplicate it by assigning to other indexes of devices. The configuration file may be used in future frequently as a definition of registers for a device of given type.

All data will be duplicated, so it is required to remember to change the MODBUS address of device, and if it is important to change the device name during copying a register settings.

To read a content of configuration file, click with right mouse button the „Configuration File” link located just above the form of device selection, and then choose “Save a target element as...” from context menu.

To save the configuration file, press “Preview...” button located at the top of the device configuration panel. In web browser, a window of file to be send will be displayed. Be sure that the file to be sent includes proper configuration of the device. Otherwise, data will be overwritten with wrong values.

Manual edition of files is not recommended, because there is possible to input wrong configuration, which in extreme case can cause breaking of communication with devices.

5.8 Configuration files

The AGEDI has two summary configuration files, where the agent configuration is stored. A first file comprises summary configuration of all devices. Second one consists of configuration of other agent settings such as: network settings, security settings, serial port settings and SNMP parameters. In case of device configuration file, the changes are made immediately with no need to reset. When second configuration file is uploaded, all changes will be performed, just after reset of a device.

Access to configuration files is possible in the „Configuration Files” tab of configuration panel.

The screenshot shows the 'Agent Configuration Files' section of the Netronix SNMP Agent configuration panel. On the left is a red sidebar menu with options: Network, Security, Serial, SNMP, I/O Ports, Device Config, Configuration Files (selected), Bootloader, and Reboot. The main content area has a title 'Agent Configuration Files' and a description: 'This page allows the configuration of the agent using a configuration files.' It features two sections for file uploads. The first section, 'Upload Devices Configuration File:', includes a yellow note: 'Note: When upload file is complete settings will apply instantly without need to reboot device.' Below this is a text input field with 'Przełączaj...' and 'Upload' buttons. A link 'Devices Configuration File' is provided for downloading. The second section, 'Upload Other Settings Configuration File:', includes another yellow note: 'Note: When upload file is complete device have to be rebooted to apply all new settings.' It also has a text input field with 'Przełączaj...' and 'Upload' buttons, and a link 'Other Settings Configuration File' for downloading. The top of the page shows the 'NETRONIX' logo and a 'Logout' link.

Copyright © 2012 Netronix

Illustration 16: Configuration files upload/download panel.

6 Verification of monitored device state

A status of queried devices you can verify by using some following methods:

- By reading the parameters from MIB base of device.
- By listening to messages sent by agent during occurrence of specified events.
- By monitoring a device status on main page of the device.

6.1 MIB database read

Parameters of monitored devices which are placed in the MIB base which is updated periodically. Read-out of this base can be performed by means of SNMP protocol queries. Base structure has been described in the last chapter of this document by using for it a MIB file. By using any program e.g. free MIB-Browser of IReasoning company, it is possible to upload a file

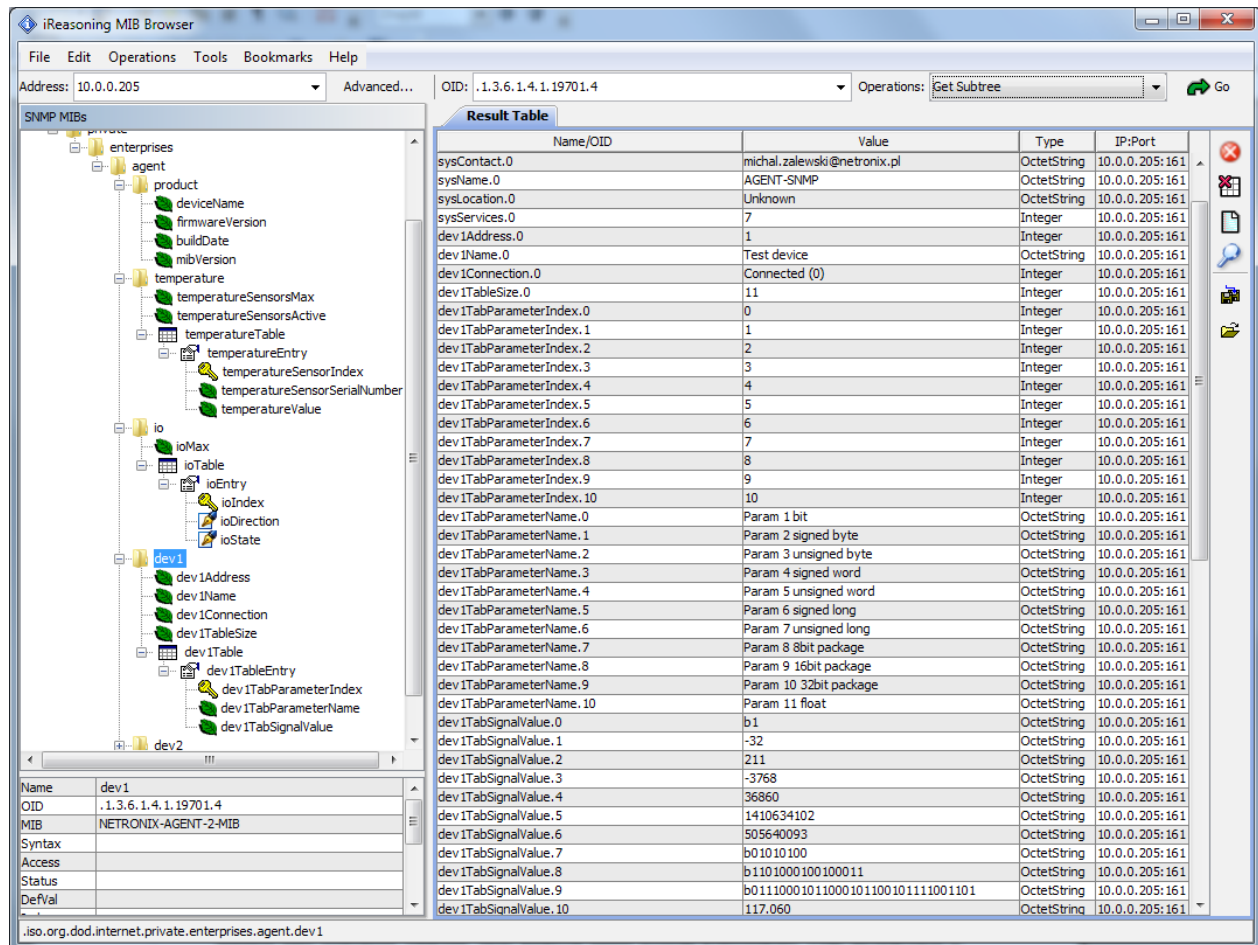


Illustration 17: MIB database preview via MIB Browser application.

which describes base structure, define IP address of agent and next read all parameters situated in the base.

6.2 Listening to SNMP trap notifications

There are four types of events on which occurrence the agent sends Trap messages informing user that specified event has happened.


1. When status of agent communication with monitored device has been changed. In case of detection of incorrect communication with a device, the agent sends the Trap message under the IP TrapNMSx address informing on incorrect status of communication. The Vendor Specific Trap Notification Type value for that event is 1.
2. A value of tested parameter has been changed. The parameters of monitored devices which are verified can be of parameter types and other than floating ones. In case of changing the value when the Value Changed flag for monitored parameter is active, the agent sends a message with the Specific Trap Notification Type equal 2.
3. A value of tested parameter has dropped below minimum value defined during configuring the monitored device. Verified parameters of monitored device can be of all types except the bit ones. In case the minimum value of the parameter is crossed, and if the Value Under flag is active, the agent will send a message with the Vendor Specific Trap Notification Type equal 3.

4. A value of tested parameter has increased above maximum value defined during configuring the monitored device. Verified parameters of monitored device can be of all types except the bit ones. In case the maximum value of the parameter is crossed, and if the Value Above flag is active, the agent sends a message with the Vendor Specific Trap Notification Type equal 4.

Basing on returned messages, it is possible to presume, if monitored device operates correctly.

6.3 Device state preview on WWW page

On the main page, an information on the monitored device status is displayed generated by the AGEDI. The data is updated periodically with interval of 60 s, by sending a query on the status.xml file generated by means of the AJAX script. Next, the file received is parsed and finally a table with the parameter results read is generated on page.


[Login](#)

Netronix SNMP Agent

Status!

Firmware Version: AGENT-SNMP-v3.01.1.1
Build Date: Jun 20 2012 16:49:30

Below you'll see the current status of the devices monitored by Agent. Data viewed on this page will refresh in 60 second period. To achive higher time resolution you will have to refresh page manual clicking refresh button in your browser.

Temperature sensors:

Sensor ID	Temperature mesasurement
229F1B13000000BC	29.400
28AB58B200000087	27.200

IO ports:

Port direction	State
Input	High
Input	High
Input	High
Input	High
Input	High
Input	High
Input	High
Input	High

Device 1 status:

State	Device Name	Address	Status
Enabled	Test device	1	Connected

Parameter Name	Type	Register	Value
Param 1 bit	Bit	0	b1
Param 2 signed byte	Byte	1	-32
Param 3 unsigned byte	UByte	2	211
Param 4 signed word	Word	4	-3768
Param 5 unsigned word	UWord	5	36860
Param 6 signed long	Long	7	1410634102
Param 7 unsigned long	ULong	9	505640093
Param 8 8bit package	8Bit_package	3	b01010100
Param 9 16bit package	16Bit_package	6	b1101000100100011
Param 10 32bit package	32Bit_package	11	b01110001011000101100101111001101
Param 11 float	Float	13	117.060

Device 2 status:

State	Device Name	Address	Status
Disabled	-	-	-

Device 3 status:

State	Device Name	Address	Status
Disabled	-	-	-

Illustration 18: Parameters results preview on WWW page.

6.4 Communication status results interpretation

The AGEDI can return three values which signal status of communication with queried devices. We can read this status by sending a query to appropriate OID of MIB base or review information displayed on home page of device.

Values of communication status:

- **Connected (0)** – Transmission to queried devices performed successfully.

- **Not-Connected (1)** – Transmission between agent and device encountered difficulties. This status means that no parameter has been read out. The agent has tried to read first three parameters defined during configuration, but none attempt has completed successfully.
- **Improper-Settings (2)** – A device returned error message. It means value was greater than or equal to 128 during read-out of parameter. When such response is encountered, other parameters of device are not queried in the same iteration.

7 Firmware update

A device can reload a firmware remotely using the TFTP client. To reload a firmware, go to the „Bootloader” tab in configuration panel. Press the „Run Bootloader” button, located at the very bottom of the panel. At this moment, the device stops executing of normal program and goes to bootloader mode for next 10 seconds. If uploading of new firmware is not initiated for 10 seconds, the agent will leave this mode of operation and return to normal mode. During start-up of the bootloader, an application delivers network addresses to it with which the application has worked till now. This behavior is very important in case of dynamic assignment of IP addresses in network.

To send a new firmware, you can use any TFTP client which is available in most of operating systems in command lines. An example of such call in case of MS Windows 2000/XP system is shown below:

```
TFTP 10.0.0.205 put agent_snmp-v3.01.1.1.nhex.
```

If reload procedure of firmware is initiated, the agent will remain in bootloader state till new program will be saved successfully. After saving is completed, the agent leaves the bootloader and switches to execute the new program.

If for some unknown reason a connection has been broken, the saving will not be completed any more and we can retry to save a new program. When the reason of transmission interruption is a power shutdown or when user wants to reset a device, after failure of saving; the agent switches to the bootloader mode again, but its network settings are changed for factory ones i.e. IP: 10.0.0.205 and address: MAC 00:04:A3:00:00:00. When the device has unique MAC address assigned, the settings will be changed because the bootloader application is different than target one. The bootloader program has no access to memory, where the unique MAC address is stored, so can appear a necessity to reset the ARP table of system.

NETRONIX [Logout](#)

Netronix SNMP Agent

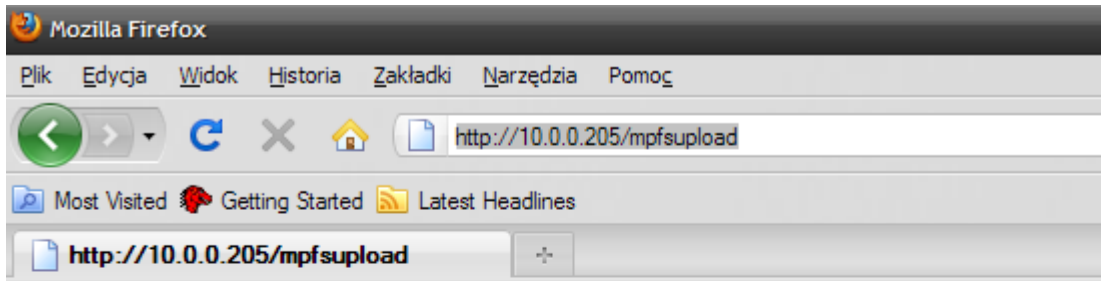
Network	<h2>Bootloader</h2> <p>Device will be accesible at the same address as currently (10.0.0.205), but only TFTP server will be active.</p> <p>-----</p> <h3>Uploading Instructions</h3> <p>To upload firmware file delivered by manufacturer you may use any of TFTP client instaled on your sytem. After device will get into bootloader state you have only 10 second to initiate upload procedure. After that time if no upload were initiated, device will be reset and return to normal work. If upload procedure were initiated, but with some reason upload didn't succed device will stay in bootloader state until firmware file upload properly. If file uploading were interupted caused by supply peak device will reset and get into bootloader state but with default IP address 10.0.0.205.</p> <p><input type="button" value="Run Bootloader"/></p>
Security	
Serial	
SNMP	
I/O Ports	
Device Config	
Configuration Files	
Bootloader	
Reboot	

Copyright © 2012 Netronix

Illustration 19: Panel used to activate bootloader.

8 File system

An agent has file system built in. In the system, data is stored such as: MIB base structure, web page and other script flies required to control a device correctly. The files are stored in external EEPROM memory, so there in possible to reload them together during writing a new program. To reload the old file system with a new one, write in a device address and „mpfsupload” i.e.: <http://10.0.0.205/mpfsupload> in address bar of browser. In browser window , the form will be displayed where we can write in an image of new file system.



MPFS Image Upload

Illustration 20: File system upload form.

Please notice that not every change of firmware must be connected with necessity of changing the file system. It is required in situations, when the GGI scripts which are performed during page display are different than ones used in previous version. Because of the file system signature, the file image should be saved complying with actual firmware version. If application signature does not comply with version of the image used and one attempts to open given web page, a message „File not Found” will be displayed.

9 Return to factory settings

In case the settings of a device delivered have been modified and establishing of connection is not possible, a user can restore factory defaults. To restore the defaults, press by means of a pin and hold for ca. 5 second the “Reset” button which is hidden under housing of the device. The device will signal the beginning of procedure for clearing the actual settings by turning on status red LED. In the moment, you can already release the switch. The device will save the factory defaults, and then will be rest.

10 Trap notifications „Specific” values

Table 14: “Specific” values send in notifications.

Value	Meaning
1	Communication status with monitored device changed.

Value	Meaning
2	Parameter value changed.
3	Parameter value under lower limit.
4	Parameter value above upper limit.

11 MIB file compatible with ASN.1 standard

```
-- MIB for "Netronix AGEDI".
-- Author: Zalewski Michal <michal.zalewski@netronix.pl>
-- Version: 1.00.01
-- Last modification date: 2012-06-05

NETRONIX-AGENT-3-MIB DEFINITIONS ::= BEGIN

IMPORTS
    enterprises, OBJECT-TYPE, MODULE-IDENTITY
        FROM SNMPv2-SMI
    TEXTUAL-CONVENTION
        FROM SNMPv2-TC;

-- 1.3.6.1.4.1.PRIVATE
agent MODULE-IDENTITY
    LAST-UPDATED "201206050900Z"
    ORGANIZATION "Netronix sp. z o.o., Poland"
    CONTACT-INFO
        "
            Build by Netronix
            WWW: http://www.netronix.pl
            E-mail: netronix@netronix.pl
        "
    DESCRIPTION "Agent Netronix model 3." ::= { enterprises
19701 }

-- Textual conventions.

PositiveInteger ::= TEXTUAL-CONVENTION
    STATUS current
    DESCRIPTION "Number >= 1."
    SYNTAX Unsigned16 (1..65535)

NonNegativeInteger ::= TEXTUAL-CONVENTION
    STATUS current
    DESCRIPTION "Number >= 0."
    SYNTAX Unsigned16 (0..65535)

ConnectionState ::= TEXTUAL-CONVENTION
    STATUS current
    DESCRIPTION "Device connection states."
```

```

SYNTAX INTEGER { Connected(0), Not-Connected(1), Improper-
Settings(2) }

```

```

IODirection ::= TEXTUAL-CONVENTION
    STATUS current
    DESCRIPTION "Port I/O TTL direction."
    SYNTAX INTEGER { output(0), input(1) }

```

```

IOState ::= TEXTUAL-CONVENTION
    STATUS current
    DESCRIPTION "Port I/O TTL states."
    SYNTAX INTEGER { low(0), high(1) }

```

```

-- Rozpoczęcie definicji struktury drzewa.
-- 1.3.6.1.4.1.PRIVATE.1
product OBJECT IDENTIFIER ::= { agent 1 }

```

```

-- 1.3.6.1.4.1.PRIVATE.1.1
deviceName OBJECT-TYPE
    SYNTAX OCTET STRING (SIZE (0..127))
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION "Device name."
    ::= { product 1 }

```

```

-- 1.3.6.1.4.1.PRIVATE.1.2
firmwareVersion OBJECT-TYPE
    SYNTAX OCTET STRING (SIZE (0..127))
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION "Software version."
    ::= { product 2 }

```

```

-- 1.3.6.1.4.1.PRIVATE.1.3
buildDate OBJECT-TYPE
    SYNTAX OCTET STRING (SIZE (0..127))
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION "Build date."
    ::= { product 3 }

```

```

-- 1.3.6.1.4.1.PRIVATE.1.4
mibVersion OBJECT-TYPE
    SYNTAX OCTET STRING (SIZE (0..127))
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION "MIB file version."
    ::= { product 4 }

```

```

-- 1.3.6.1.4.1.PRIVATE.2

```

```

temperature OBJECT IDENTIFIER ::= { agent 2 }

-- 1.3.6.1.4.1.PRIVATE.2.1
temperatureSensorsMax OBJECT-TYPE
    SYNTAX NonNegativeInteger
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION "Number of temperature sensors supported by
agent."
    ::= { temperature 1 }

-- 1.3.6.1.4.1.PRIVATE.2.2
temperatureSensorsActive OBJECT-TYPE
    SYNTAX NonNegativeInteger
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION "Number of temperature sensors found on 1-WIRE
bus after restart."
    ::= { temperature 2 }

-- 1.3.6.1.4.1.PRIVATE.2.3
temperatureTable OBJECT-TYPE
    SYNTAX SEQUENCE OF TemperatureEntry
    MAX-ACCESS not-accessible
    STATUS current
    DESCRIPTION "Temperature sensors table."
    ::= { temperature 3 }

-- 1.3.6.1.4.1.PRIVATE.2.3.1
temperatureEntry OBJECT-TYPE
    SYNTAX TemperatureEntry
    MAX-ACCESS not-accessible
    STATUS current
    DESCRIPTION "Table entry."
    INDEX { temperatureSensorIndex }
    ::= { temperatureTable 1 }

TemperatureEntry ::= SEQUENCE {
    temperatureSensorIndex          PositiveInteger,
    temperatureSensorSerialNumber  OCTET STRING,
    temperatureValue                INTEGER
}

-- 1.3.6.1.4.1.PRIVATE.2.3.1.1
temperatureSensorIndex OBJECT-TYPE
    SYNTAX PositiveInteger
    MAX-ACCESS not-accessible
    STATUS current
    DESCRIPTION "Sensor index."
    ::= { temperatureEntry 1 }

```

```

-- 1.3.6.1.4.1.PRIVATE.2.3.1.2
temperatureSensorSerialNumber OBJECT-TYPE
    SYNTAX OCTET STRING (SIZE (8))
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION "Temperature sensor unique serial number."
    ::= { temperatureEntry 2 }

-- 1.3.6.1.4.1.PRIVATE.2.3.1.3
temperatureValue OBJECT-TYPE
    SYNTAX INTEGER
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION "Temperature value."
    ::= { temperatureEntry 3 }

-- 1.3.6.1.4.1.PRIVATE.3
io OBJECT IDENTIFIER ::= { agent 3 }

-- 1.3.6.1.4.1.PRIVATE.3.1
ioMax OBJECT-TYPE
    SYNTAX NonNegativeInteger
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION "Number of inputs/outputs TTL supported by
agent."
    ::= { io 1 }

-- 1.3.6.1.4.1.PRIVATE.3.2
ioTable OBJECT-TYPE
    SYNTAX SEQUENCE OF IOEntry
    MAX-ACCESS not-accessible
    STATUS current
    DESCRIPTION "I/O table."
    ::= { io 2 }

-- 1.3.6.1.4.1.PRIVATE.3.2.1
ioEntry OBJECT-TYPE
    SYNTAX IOEntry
    MAX-ACCESS not-accessible
    STATUS current
    DESCRIPTION "Table entry."
    INDEX { ioIndex }
    ::= { ioTable 1 }

IOEntry ::= SEQUENCE {
    ioIndex                PositiveInteger,
    ioDirection            IODirection,

```

```

        ioState                IOState
    }

-- 1.3.6.1.4.1.PRIVATE.3.2.1.1
ioIndex OBJECT-TYPE
    SYNTAX PositiveInteger
    MAX-ACCESS not-accessible
    STATUS current
    DESCRIPTION "I/O index."
    ::= { ioEntry 1 }

-- 1.3.6.1.4.1.PRIVATE.3.2.1.2
ioDirection OBJECT-TYPE
    SYNTAX IODirection
    MAX-ACCESS read-write
    STATUS current
    DESCRIPTION "I/O direction (input or output)."
```

```

    ::= { ioEntry 2 }

-- 1.3.6.1.4.1.PRIVATE.3.2.1.3
ioState OBJECT-TYPE
    SYNTAX IOState
    MAX-ACCESS read-write
    STATUS current
    DESCRIPTION "I/O state (low or high)."
```

```

    ::= { ioEntry 3 }

-- 1.3.6.1.4.1.PRIVATE.4
dev1 OBJECT IDENTIFIER ::= { agent 4 }

-- 1.3.6.1.4.1.PRIVATE.4.1
dev1Address OBJECT-TYPE
    SYNTAX NonNegativeInteger
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION "Modbus device 1 address."
    ::= { dev1 1 }

-- 1.3.6.1.4.1.PRIVATE.4.2
dev1Name OBJECT-TYPE
    SYNTAX OCTET STRING (SIZE (0..127))
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION "Device 1 name."
    ::= { dev1 2 }

-- 1.3.6.1.4.1.PRIVATE.4.3
dev1Connection OBJECT-TYPE
    SYNTAX ConnectionState
    MAX-ACCESS read-only
```

```

        STATUS current
        DESCRIPTION "Device connection status."
        ::= { dev1 3 }

-- 1.3.6.1.4.1.PRIVATE.4.4
dev1TableSize OBJECT-TYPE
    SYNTAX NonNegativeInteger
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION "Size of parameters table."
    ::= { dev1 4 }

-- 1.3.6.1.4.1.PRIVATE.4.5
dev1Table OBJECT-TYPE
    SYNTAX SEQUENCE OF Dev1TableEntry
    MAX-ACCESS not-accessible
    STATUS current
    DESCRIPTION "Parameters table."
    ::= { dev1 5 }

-- 1.3.6.1.4.1.PRIVATE.4.5.1
dev1TableEntry OBJECT-TYPE
    SYNTAX Dev1TableEntry
    MAX-ACCESS not-accessible
    STATUS current
    DESCRIPTION "Table entry."
    INDEX { dev1TabParameterIndex }
    ::= { dev1Table 1 }

-- Definicja struktury tabeli.
Dev1TableEntry ::= SEQUENCE {
    dev1TabParameterIndex                NonNegativeInteger,
    dev1TabParameterName                 OCTET STRING,
    dev1TabSignalValue                   OCTET STRING,
}

dev1TabParameterIndex OBJECT-TYPE
    SYNTAX NonNegativeInteger
    MAX-ACCESS not-accessible
    STATUS current
    DESCRIPTION "Parameter index."
    ::= { dev1TableEntry 1 }

dev1TabParameterName OBJECT-TYPE
    SYNTAX OCTET STRING (SIZE (0..127))
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION "Parameter name."
    ::= { dev1TableEntry 2 }

```

```

dev1TabSignalValue OBJECT-TYPE
    SYNTAX OCTET STRING (SIZE (0..127))
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION "Signal value."
    ::= { dev1TableEntry 3 }

-- 1.3.6.1.4.1.PRIVATE.5
dev2 OBJECT IDENTIFIER ::= { agent 5 }

-- 1.3.6.1.4.1.PRIVATE.5.1
dev2Address OBJECT-TYPE
    SYNTAX NonNegativeInteger
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION "Modbus device 2 address."
    ::= { dev2 1 }

-- 1.3.6.1.4.1.PRIVATE.5.2
dev2Name OBJECT-TYPE
    SYNTAX OCTET STRING (SIZE (0..127))
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION "Device 2 name."
    ::= { dev2 2 }

-- 1.3.6.1.4.1.PRIVATE.5.3
dev2Connection OBJECT-TYPE
    SYNTAX ConnectionState
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION "Device connection status."
    ::= { dev2 3 }

-- 1.3.6.1.4.1.PRIVATE.5.4
dev2TableSize OBJECT-TYPE
    SYNTAX NonNegativeInteger
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION "Size of parameters table."
    ::= { dev2 4 }

-- 1.3.6.1.4.1.PRIVATE.5.5
dev2Table OBJECT-TYPE
    SYNTAX SEQUENCE OF Dev2TableEntry
    MAX-ACCESS not-accessible
    STATUS current
    DESCRIPTION "Parameters table."
    ::= { dev2 5 }

```



```

-- 1.3.6.1.4.1.PRIVATE.5.5.1
dev2TableEntry OBJECT-TYPE
    SYNTAX Dev2TableEntry
    MAX-ACCESS not-accessible
    STATUS current
    DESCRIPTION "Table entry."
    INDEX { dev2TabParameterIndex }
    ::= { dev2Table 1 }

-- Definicja struktury tabeli.
Dev2TableEntry ::= SEQUENCE {
    dev2TabParameterIndex          NonNegativeInteger,
    dev2TabParameterName          OCTET STRING,
    dev2TabSignalValue            OCTET STRING,
}

dev2TabParameterIndex OBJECT-TYPE
    SYNTAX NonNegativeInteger
    MAX-ACCESS not-accessible
    STATUS current
    DESCRIPTION "Parameter index."
    ::= { dev2TableEntry 1 }

dev2TabParameterName OBJECT-TYPE
    SYNTAX OCTET STRING (SIZE (0..127))
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION "Parameter name."
    ::= { dev2TableEntry 2 }

dev2TabSignalValue OBJECT-TYPE
    SYNTAX OCTET STRING (SIZE (0..127))
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION "Signal value."
    ::= { dev2TableEntry 3 }

-- 1.3.6.1.4.1.PRIVATE.6
dev3 OBJECT IDENTIFIER ::= { agent 6 }

-- 1.3.6.1.4.1.PRIVATE.6.1
dev3Address OBJECT-TYPE
    SYNTAX NonNegativeInteger
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION "Modbus device 3 address."
    ::= { dev3 1 }

```

```

-- 1.3.6.1.4.1.PRIVATE.6.2
dev3Name OBJECT-TYPE
    SYNTAX OCTET STRING (SIZE (0..127))
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION "Device 3 name."
    ::= { dev3 2 }

-- 1.3.6.1.4.1.PRIVATE.6.3
dev3Connection OBJECT-TYPE
    SYNTAX ConnectionState
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION "Device connection status."
    ::= { dev3 3 }

-- 1.3.6.1.4.1.PRIVATE.6.4
dev3TableSize OBJECT-TYPE
    SYNTAX NonNegativeInteger
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION "Size of parameters table."
    ::= { dev3 4 }

-- 1.3.6.1.4.1.PRIVATE.6.5
dev3Table OBJECT-TYPE
    SYNTAX SEQUENCE OF Dev3TableEntry
    MAX-ACCESS not-accessible
    STATUS current
    DESCRIPTION "Parameters table."
    ::= { dev3 5 }

-- 1.3.6.1.4.1.PRIVATE.6.5.1
dev3TableEntry OBJECT-TYPE
    SYNTAX Dev3TableEntry
    MAX-ACCESS not-accessible
    STATUS current
    DESCRIPTION "Table entry."
    INDEX { dev3TabParameterIndex }
    ::= { dev3Table 1 }

-- Definicja struktury tabeli.
Dev3TableEntry ::= SEQUENCE {
    dev3TabParameterIndex          NonNegativeInteger,
    dev3TabParameterName          OCTET STRING,
    dev3TabSignalValue            OCTET STRING,
}

dev3TabParameterIndex OBJECT-TYPE
    SYNTAX NonNegativeInteger

```

```
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION "Parameter index."
 ::= { dev3TableEntry 1 }

dev3TabParameterName OBJECT-TYPE
SYNTAX OCTET STRING (SIZE (0..127))
MAX-ACCESS read-only
STATUS current
DESCRIPTION "Parameter name."
 ::= { dev3TableEntry 2 }

dev3TabSignalValue OBJECT-TYPE
SYNTAX OCTET STRING (SIZE (0..127))
MAX-ACCESS read-only
STATUS current
DESCRIPTION "Signal value."
 ::= { dev3TableEntry 3 }

-- 1.3.6.1.4.1.PRIVATE.7
dev4 OBJECT IDENTIFIER ::= { agent 7 }

-- 1.3.6.1.4.1.PRIVATE.7.1
dev4Address OBJECT-TYPE
SYNTAX NonNegativeInteger
MAX-ACCESS read-only
STATUS current
DESCRIPTION "Modbus device 4 address."
 ::= { dev4 1 }

-- 1.3.6.1.4.1.PRIVATE.7.2
dev4Name OBJECT-TYPE
SYNTAX OCTET STRING (SIZE (0..127))
MAX-ACCESS read-only
STATUS current
DESCRIPTION "Device 4 name."
 ::= { dev4 2 }

-- 1.3.6.1.4.1.PRIVATE.7.3
dev4Connection OBJECT-TYPE
SYNTAX ConnectionState
MAX-ACCESS read-only
STATUS current
DESCRIPTION "Device connection status."
 ::= { dev4 3 }

-- 1.3.6.1.4.1.PRIVATE.7.4
dev4TableSize OBJECT-TYPE
SYNTAX NonNegativeInteger
MAX-ACCESS read-only
```

```

        STATUS current
        DESCRIPTION "Size of parameters table."
        ::= { dev4 4 }

-- 1.3.6.1.4.1.PRIVATE.7.5
dev4Table OBJECT-TYPE
    SYNTAX SEQUENCE OF Dev4TableEntry
    MAX-ACCESS not-accessible
    STATUS current
    DESCRIPTION "Parameters table."
    ::= { dev4 5 }

-- 1.3.6.1.4.1.PRIVATE.7.5.1
dev4TableEntry OBJECT-TYPE
    SYNTAX Dev4TableEntry
    MAX-ACCESS not-accessible
    STATUS current
    DESCRIPTION "Table entry."
    INDEX { dev4TabParameterIndex }
    ::= { dev4Table 1 }

-- Definicja struktury tabeli.
Dev4TableEntry ::= SEQUENCE {
    dev4TabParameterIndex          NonNegativeInteger,
    dev4TabParameterName          OCTET STRING,
    dev4TabSignalValue            OCTET STRING,
}

dev4TabParameterIndex OBJECT-TYPE
    SYNTAX NonNegativeInteger
    MAX-ACCESS not-accessible
    STATUS current
    DESCRIPTION "Parameter index."
    ::= { dev4TableEntry 1 }

dev4TabParameterName OBJECT-TYPE
    SYNTAX OCTET STRING (SIZE (0..127))
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION "Parameter name."
    ::= { dev4TableEntry 2 }

dev4TabSignalValue OBJECT-TYPE
    SYNTAX OCTET STRING (SIZE (0..127))
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION "Signal value."
    ::= { dev4TableEntry 3 }
END

```