PAN9320 Communication Specification

Firmware V1.9.0.1

Table of Contents

1 Abstract 7
2 Commands 8
2.1 WLAN-Service (module: wlan) 9
2.1.1 Infrastructure SSID (variable: infra_ssid) 9
2.1.2 Infrastructure Mode (variable: infra_mode) 9
2.1.3 Infrastructure Status (variable: infra_status) 10
2.1.4 Infrastructure Security (variable: infra_sec) 10
2.1.5 Infrastructure Configuration (variable: infra_cfg) 11
2.1.6 Access Point SSID (variable: ap_ssid) 11
2.1.7 Access Point Mode (variable: ap_mode) 12
2.1.8 Access Point Security (variable: ap_sec) 12
2.1.9 Access Point Configuration (variable: ap_cfg) 13
2.1.10 Access Point Station list (variable: ap_stlist) 13
2.1.11 Wireless Network Scan (variable: scan) 14
2.1.12 Available Wireless Networks (variable: list) 14
2.1.13 WLAN Region Code (variable: rcode) 15
2.1.14 WLAN BSSID (variable: infra_bssid) 15
2.1.15 WLAN Channel (variable: infra_rch) 16
2.1.16 WLAN RSSI (variable: infra_rssi) 16
2.2 Name-Services (module: name) 16
2.2.1 Device Name (variable: device) 16
2.2.2 MDNS Domain (variable: mdns_domain) 17
2.2.3 MDNS Mode (variable: mdns_mode) 18
2.2.4 MDNS Server 1/2 (variable: mdns_serv1 || mdns_serv2) 18
2.2.5 NetBIOS Name Service Name (variable: nbns_name) 19
2.2.6 NetBIOS Mode (variable: nbns_mode) 20
2.3 NET-Service (module: net) 20
2.3.1 IP-Configuration (variable: ipcfg) 20
2.3.2 Fixed IP-Address (variable: fixedip) 21
2.3.3 IP Mode (variable: mode) 23
2.3.4 PING Request (variable: ping_req) 23
2.3.5 PING Status (variable: ping_status) 24
2.4 Email (module: smail) 24
2.4.1 User mail address (variable: sender) 24
2.4.2 Server configuration (variable: server) 25
2.4.3 Mail sending (variable: send) 26
2.4.4 Mail status (variable: status) 26
2.4.5 Mail error (variable: error) 27

2.5 System (module: system) 27
2.5.1 Firmware version (variable: firmware) 27
2.5.2 MAC address (variable: macaddr)....27
2.5.3 Serial number (variable: serialnum)....28
2.5.4 WiFi firmware version (variable: wifi_ver) 28
2.5.5 Bootloader version (variable: bootl_ver)....28
2.5.6 Hardware revision (variable: hwrev) 29
2.5.7 Restart (variable: restart)....29
2.5.8 Reset (variable: factory)....29
2.5.9 Save mode (variable: savemode)....30
2.5.10 Save the configuration (variable: savecfg)....30
2.5.11 Power save mode (variable: psm)....30
2.6 User Managment (module: user) 31
2.6.1 User edit/add (variable: edit)....31
2.6.2 User name (variable: name) 32
2.6.3 User delete (variable: delete) 32
2.6.4 Default login active (variable: dlogin) 33
2.6.5 Default Rights (variable: drights) 33
2.6.6 My rights (variable: myrights)....34
2.7 Telnet (module: telnet)....34
2.7.1 Mode (variable: mode) 34
2.7.2 Auth(variable: auth)....35
2.8 Firmware Update (module: fwu) 35
2.8.1 Progress (variable: progress) 35
2.8.2 Status (variable: status) 36
2.8.3 Check for firmware version (variable: checkfw) 36
2.8.4 Firmware Update (variable: fwupdate) 37
2.9 Commando UART (module: cmduart) 37
2.9.1 Send (variable: send)....37
2.9.2 Request (variable: request)....38
2.9.3 Uart configuration (variable: cfg)....38
2.9.4 Send user rights via UART (variable: sendauth)....39
2.10 GPIO (module: gpio) 39
2.10.1 Mode (variable: mode) 39
2.10.2 State (variable: state)....40
2.11 HTTP-Client (module: httpc) 41
2.11.1 Status (variable: status) 41
2.11.2 HTTP-Server (variable: server) 41
2.11.3 POST (variable: post)....42
2.11.4 GET (variable: get)....42

2.11.5 CMDAPI(variable: cmdapi) 43
2.11.6 TOKEN (variable: token) 43
2.12 Netcat (module: netcat) 44
2.12.1 State (variable: state) 44
2.12.2 Server IP (variable: serv_ip) 44
2.12.3 Server Login (variable: serv_login) 45
2.12.4 Telnet option (variable: telopt) 46
2.12.5 TCP Port (variable: tcp_port) 46
2.12.6 Mode(variable: mode) 47
2.12.7 Exclusive lock(variable: lock) 47
2.12.8 Release lock(variable: release) 48
2.12.9 Authentication(variable: auth) 48
2.12.10 DTR-Pin(variable: dtrpin) 49
2.13 Uart Binary (Module: binuart) 49
2.13.1 Uart configuration (variable: cfg) 49
2.13.2 Read Data(variable: data) 50
2.13.3 Write Data(variable: data) 51
2.14 Time (Module: time) 51
2.14.1 UTC (variable: utc) 51
2.14.2 Time Sync (variable: sync) 52
3 Communication with the host-controller 53
3.1 COMMAND-UART 53
3.1.1 Additional user rights 53
3.2 BINARY-UART 53
3.2.1 Supported baudrates 53
4 Status Information 54
4.1 Telegram Return Code 54
4.2 WiFi-Parameter 55
4.2.1 WiFi status information 55
4.3 Mail-Service 56
4.3.1 Mail module status 56
4.4 User-Managment 56
4.4.1 User-Rights 56
4.5 Firmware Update 57
4.5.1 Firmware Update Status list 57
4.6 HTTP-Client 57
4.6.1 HTTP-Client Status 57
4.7 Uart configuration 57
4.7.1 Data bit type 57
4.7.2 Parity types 58

4.7.3 Stop bit type....58
4.8 GPIO....59
5 Change History....60
6 General information....61

1 ABSTRACT

The PAN9320 is designed to fulfill the main requirements of customers, who are not familiar with the WiFi technology, but needs Wifi as an added value in a product.
It provides a full embedded webstack implemented on a Marvell MC200 combined with Marvells 88W8782 WiFi SoC.

This document specifies the communication via UART and HTTP/JSON for the various features of the PAN9320.

Please contact your local sales office for further details on additional options and services, by visiting www.panasonic.com/rfmodules for U.S. and http://industrial.panasonic.com/eu/ for Europe or write an e-mail to wireless@eu.panasonic.com

2 COMMANDS

There are three types of commands. The format is ascii based ended with CR-LF.

- **Request**
- **Response**
- **Error**

Request commands are used to request data, set values and trigger various functionalities. The structure is as follows:

Command	Module	Variable	Parameter

Command: Either "get" or "set"
Module: Defines the functional software module, which shall be addressed
Variable: A subset of the selected module
Parameter: Dependent on the module and variable, different parameters can be used.

Each request will trigger a *Response* message of the following structure:

Command	Module	Variable	Return Code	Parameter

Command: Either "get" or "set"
Module: Defines the functional software module, which shall be addressed
Variable: A subset of the selected module
Return Code: For the codes please see Chapter 4.1
Parameter: Dependent on the module and variable, different parameters can be used.

Error messages will only have the following structure:

"Error"	Return Code

Return Code: For the codes please see Chapter 4.1

Example of the architecture:

HTTP/JSON Request	["Command","Module","Variable"]
HTTP/JSON Response	["Command","Module","Variable","Return-Code","Parameter"]
CMD-UART Request	Command Module Variable\x0d\x0a
CMD-UART Response	Command Module Variable Return-Code Parameter\x0d\x0a

2.1 WLAN-SERVICE (MODULE: WLAN)

2.1.1 Infrastructure SSID (variable: infra_ssid)

Command option:	get
Parameter 1:	<ssid> - SSID of the infastructure network to which the application is connected. Default value: - Range: String with ASCII printable characters with a length of (1-32).
Description :	Returns the ssid of the wlan network as a ASCII-encoded string with max. length 32 characters to which the application is connected.
GET-Rights:	0x00

Examples:

HTTP/JSON Get-Request	["get","wlan","infra_ssid"]
HTTP/JSON Get-Response	["get","wlan","infra_ssid","0","testnetwork"]
CMD-UART Get-Request	get wlan infra_ssid\x0d\x0a
CMD-UART Get-Response	get wlan infra_ssid 0 testnetwork\x0d\x0a

2.1.2 Infrastructure Mode (variable: infra_mode)

Command option:	get / set
Parameter 1:	<mode> - Enables/Disables infrastructure mode. Default value: off Range: [on, off] on: mode enabled off: mode disabled
Description:	The variable return the status of the infrastructure mode, the variable can also enable or disable the infrastructure mode.
GET-Rights:	0x00
SET-Rights:	0x02

Examples:

HTTP/JSON Get-Request	["get","wlan","infra_mode"]
HTTP/JSON Get-Response	["get","wlan","infra_mode","0","on"]
HTTP/JSON Set-Request	["set","wlan","infra_mode","off"]
HTTP/JSON Set-Response	["set","wlan","infra_mode","0","off"]

CMD-UART Get-Request	get wlan infra_mode\x0d\x0a
CMD-UART Get-Response	get wlan infra_mode 0 on\x0d\x0a
CMD-UART Set-Request	set wlan infra_mode off\x0d\x0a
CMD-UART Set-Response	set wlan infra_mode 0 off\x0d\x0a

2.1.3 Infrastructure Status (variable: infra_status)

Command option:	get
Parameter 1:	<status> - Status of the infrastructure connection. Default value: 0 Range: 4.2.1
Description:	Returns the status of the infrastructure connection. See list 4.2.1
GET-Rights:	0x00

Examples:

HTTP/JSON Get-Request	["get","wlan","infra_status"]
HTTP/JSON Get-Response	["get","wlan","infra_status","0","4"]
CMD-UART Get-Request	get wlan infra_status\x0d\x0a
CMD-UART Get-Response	get wlan infra_status 0 4\x0d\x0a

2.1.4 Infrastructure Security (variable: infra_sec)

Command option:	get
Parameter 1:	<security> - Security setting of the infrastructure network. Default value: - Range: [1..5] 1: OPEN 2: WEP 3: WPA 4: WPA2 5: WPA2_MIX
Description:	Returns the security setting of the infrastructure network to which the application is connected.
GET-Rights:	0x00

Examples:

HTTP/JSON Get-Request	["get","wlan","infra_sec"]
HTTP/JSON Get-Response	["get","wlan","infra_sec","0","2"]
CMD-UART Get-Request	get wlan infra_sec\x0d\x0a
CMD-UART Get-Response	get wlan infra_sec 0 2\x0d\x0a

2.1.5 Infrastructure Configuration (variable: infra_cfg)

Command option:	set
Parameter 1:	<ssid> - SSID of the chosen infrastructure network. Default value: - Range: String with ASCII printable characters with a length of (1-32).
Parameter 2:	<psk> - Pass phrase of the chosen infrastructure network. Default value: -
Parameter 3:	<security> - Security setting of the infrastructure network. Default value: - Range: [1..5] 1: OPEN 2: WEP 3: WPA 4: WPA2 5: WPA2_MIX
Description:	With the variable, a connection to a chosen infrastructure network could be established. The pass phrase and the security level must match the settings of the infrastructure network.
SET-Rights:	0x02

Examples:

HTTP/JSON Set-Request	["set","wlan","infra_cfg","testnetwork","password","2"]
HTTP/JSON Set-Response	["set","wlan","infra_cfg","0","testnetwork","password","2"]
CMD-UART Set-Request	set wlan infra_cfg testnetwork password 2\x0d\x0a
CMD-UART Set-Response	set wlan infra_cfg 0 testnetwork password 2\x0d\x0a

2.1.6 Access Point SSID (variable: ap_ssid)

Command option:	get
Parameter 1:	<ssid> - SSID of the access point. Default value: PAN9320_AP Default password: PAN_9320 Range: String with ASCII printable characters with a length of (1-32).
Description :	Returns the ssid of the access point as a ASCII-encoded string with max. length 32 characters.
GET-Rights:	0x00

Examples:

HTTP/JSON Get-Request	["get","wlan","ap_ssid"]
HTTP/JSON Get-Response	["get","wlan","ap_ssid","0","apwlannetz"]

CMD-UART Get-Request	get wlan ap_ssid\x0d\x0a
CMD-UART Get-Response	get wlan ap_ssid 0 apwlannetz\x0d\x0a

2.1.7 Access Point Mode (variable: ap_mode)

Command option:	get / set
Parameter 1:	<mode> - Enables/Disables access point mode. Default value: on Range: [on, off] on: mode enabled off: mode disabled
Description:	The variable return the status of the access point mode, the variable can also enable or disable the access point mode.
GET-Rights:	0x00
SET-Rights:	0x04

Examples:

HTTP/JSON Get-Request	["get","wlan","ap_mode"]
HTTP/JSON Get-Response	["get","wlan","ap_mode","0","on"]
HTTP/JSON Set-Request	["set","wlan","ap_mode","off"]
HTTP/JSON Set-Response	["set","wlan","ap_mode","0","off"]
CMD-UART Get-Request	get wlan ap_mode\x0d\x0a
CMD-UART Get-Response	get wlan ap_mode 0 on\x0d\x0a
CMD-UART Set-Request	set wlan ap_mode off\x0d\x0a
CMD-UART Set-Response	set wlan ap_mode 0 off\x0d\x0a

2.1.8 Access Point Security (variable: ap_sec)

Command option:	get
Parameter 1:	<security> - Security setting of the access point network. Default value: 4 Range: [1..5] 1: OPEN 2: WEP 3: WPA 4: WPA2 5: WPA2_MIX
Description:	Returns the security setting of the access point network.
GET-Rights:	0x00

Examples:

HTTP/JSON Get-Request	["get","wlan","ap_sec"]
HTTP/JSON Get-Response	["get","wlan","ap_sec","0","2"]
CMD-UART Get-Request	get wlan ap_sec\x0d\x0a
CMD-UART Get-Response	get wlan ap_sec 0 2\x0d\x0a

2.1.9 Access Point Configuration (variable: ap_cfg)

Command option:	set
Parameter 1:	<ssid> - SSID of the access point. Default value: - Range: String with ASCII printable characters with a length of (1-32).
Parameter 2:	<psk> - Pass phrase of the access point. Default value: - Range: String with ASCII printable characters with a length of (8-63).
Parameter 3:	<security> - Security level of the access point. Default value: - 1: OPEN 2: WEP /* Not supproted */ 3: WPA 4: WPA2 5: WPA2_MIX
Parameter 4:	<radio channel> - Radio channel for the AP, the radio channel will be automatic adapted if the wlan infrastructure interface is connecting to a wlan network.
Description:	Configures the access point.
SET-Rights:	0x04

Examples:

HTTP/JSON Set-Request	["set","wlan","ap_cfg","apwlan","psk","4","0"]
HTTP/JSON Set-Response	["set","wlan","ap_cfg","0","apwlan","psk","4","0"]
CMD-UART Set-Request	set wlan ap_cfg apwlan psk 4 0\x0d\x0a
CMD-UART Set-Response	set wlan ap_cfg 0 apwlan psk 4 0\x0d\x0a

2.1.10 Access Point Station list (variable: ap_stlist)

Command option:	get
Parameter 1:	<stlist> - List of connected devices (MAC-Address)
Description:	With this parameter a list of connected devices can be requested. The MAC-Address of the device is listed
GET-Rights:	0x00

Examples:

HTTP/JSON Set-Request	["get","wlan","ap_stlist"]
HTTP/JSON Set-Response	["get","wlan","ap_stlist","0","01:02:03:04:05:06","11:12:13:14:15:16"]
CMD-UART Set-Request	get wlan ap_stlist\x0d\x0a
CMD-UART Set-Response	get wlan ap_stlist 0 01:02:03:04:05:06 11:12:13:14:15:16 \x0d\x0a

2.1.11 Wireless Network Scan (variable: scan)

Command option:	set
Description:	Triggers a scan for available Wireless networks
SET-Rights:	0x02

Examples:

HTTP/JSON Set-Request	["set","wlan","scan"]
HTTP/JSON Set-Response	["set","wlan","scan","0"]
CMD-UART Set-Request	set wlan scan\x0d\x0a
CMD-UART Set-Response	set wlan scan 0\x0d\x0a

2.1.12 Available Wireless Networks (variable: list)

Command option:	get
Parameter 1:	<ssid> - SSID of of the found wireless network. Default value: - Range: String with ASCII printable characters with a length of (1-32).
Parameter 2:	<security> - Security level of the found wireless network. Default value: - Range: [1..5] 1: OPEN 2: WEP 3: WPA 4: WPA2 5: WPA2_MIX
Parameter 3:	<signal> - Signal strength of the found wireless network. Default value: - Range: 0-100
Description:	Returns a list of available wireless networks which were found by the network scan. All found networks up to 8 will be returned in a single Get-Respond. Each found network is described with its SSID, security type and signal strength. The possibly first three

	parameters (1-3) describe the first network in the list. The possibly next three parameters (4-6) describe the second network in the list and so on, till the maximum possibly last three parameters (16-18) for the eights and last network in the list.
GET-Rights:	0x00

Examples:

HTTP/JSON Get-Request	["get","wlan","list"]
HTTP/JSON Get-Response	["get","wlan","list","0","net1","2","50","net2","1","80"]
CMD-UART Get-Request	get wlan list\x0d\x0a
CMD-UART Get-Response	get wlan list 0 net1 2 50 net2 1 80\x0d\x0a

2.1.13 WLAN Region Code (variable: rcode)

Command option:	get
Parameter 1:	<code> - region code Default value: EU Range [EU, US]
Description:	Region code for the radio chip
GET-Rights:	0x00

Examples:

HTTP/JSON Get-Request	["get","wlan","rcode"]
HTTP/JSON Get-Response	["get","wlan","rcode","0",”EU”]
CMD-UART Get-Request	get wlan rcode\x0d\x0a
CMD-UART Get-Response	get wlan rcode 0 EU\x0d\x0a

2.1.14 WLAN BSSID (variable: infra_bssid)

Command option:	get
Parameter 1:	<bssid> - Current BSSID Default value: -
Description:	BSSID of the connected WiFi
GET-Rights:	0x00

Examples:

HTTP/JSON Get-Request	["get","wlan","infra_bssid"]
HTTP/JSON Get-Response	["get","wlan","infra_bssid","0",”80:70:60:50:40:30”]
CMD-UART Get-Request	get wlan infra_bssid\x0d\x0a
CMD-UART Get-Response	get wlan infra_bssid 0 80:70:60:50:40:30\x0d\x0a

2.1.15 WLAN Channel (variable: infra_rch)

Command option:	get
Parameter 1:	<channel> - Current radio channel Default value: -
Description:	Radio channel of the connected WiFi
GET-Rights:	0x00

Examples:

HTTP/JSON Get-Request	["get","wlan","infra_rch"]
HTTP/JSON Get-Response	["get","wlan","infra_rch","0",”5”]
CMD-UART Get-Request	get wlan infra_rch\x0d\x0a
CMD-UART Get-Response	get wlan infra_bss rch id 0 5\x0d\x0a

2.1.16 WLAN RSSI (variable: infra_rssi)

Command option:	get
Parameter 1:	<rssi> - Current RSSI Default value: - Range:
Description:	RSSI of the connected WiFi
GET-Rights:	0x00

Examples:

HTTP/JSON Get-Request	["get","wlan","infra_rssi"]
HTTP/JSON Get-Response	["get","wlan","infra_rssi","0",”80”]
CMD-UART Get-Request	get wlan infra_rssi\x0d\x0a
CMD-UART Get-Response	get wlan infra_ rssi rch id 0 80\x0d\x0a

2.2 NAME-SERVICES (MODULE: NAME)

2.2.1 Device Name (variable: device)

Command option:	set
Parameter 1:	<name> - Device name Default value: - Range: String with maximum of 15 characters
Description:	The device name is universal name and is taken for the following services: - MDNS-Domain - MDNS-Serv1-Inst-Name

	- MDNS-Serv2-Inst-Name - NBNS-Name The device name is limited to just twenty-six letters, ten digits and the hyphen character, it is not allowed to use spaces or other punctuation. Get-Command not useful, because the set device name can be differ, if one or more name services detect name conflicts. Nonetheless it is possible to get and also set the service name separately from each name service. See 2.2.4 and 2.2.5 .
SET-Rights:	0x02

Examples:

HTTP/JSON Set-Request	["set","name","device","pan9320"]
HTTP/JSON Set-Response	["set","name","device","0","pan9320"]
CMD-UART Set-Request	set name device pan9320\x0d\x0a
CMD-UART Set-Response	set name device 0 pan9320\x0d\x0a

2.2.2 MDNS Domain (variable: mdns_domain)

Command option:	get / set
Parameter 1:	<name> - MDNS domain. Default value: pan9320 Range: Limit by software with a maximum of 31 characters
Description:	The domain is for the mdns service. The mdns domain is limited to just twenty-six letters, ten digits and the hyphen character, it is not allowed to use spaces or other punctuation.
GET-Rights:	0x00
SET-Rights:	0x04

Examples:

HTTP/JSON Get-Request	["get","name","mdns_domain"]
HTTP/JSON Get-Response	["get","name","mdns_domain","0","pan9320"]
HTTP/JSON Set-Request	["set","name","mdns_domain","pan9320"]
HTTP/JSON Set-Response	["set","name","mdns_domain","0","pan9320"]
CMD-UART Get-Request	get name mdns_domain\x0d\x0a
CMD-UART Get-Response	get name mdns_domain 0 pan9320\x0d\x0a
CMD-UART Set-Request	set name mdns_domain pan9320\x0d\x0a
CMD-UART Set-Response	set name mdns_domain 0 pan9320\x0d\x0a

2.2.3 MDNS Mode (variable: mdns_mode)

Command option:	get / set
Parameter 1:	<mode> - Enables/Disables MDNS mode. Default value: on Range: [on, off] on: mode enabled off: mode disabled
Description:	The MDNS mode is to switch on or off the MDNS service.
GET-Rights:	0x00
SET-Rights:	0x04

Examples:

HTTP/JSON Get-Request	["get","name","mdns_mode"]
HTTP/JSON Get-Response	["get","name","mdns_mode","0","on"]
HTTP/JSON Set-Request	["set","name","mdns_mode","on"]
HTTP/JSON Set-Response	["set","name","mdns_mode","0","on"]
CMD-UART Get-Request	get name mdns_mode\x0d\x0a
CMD-UART Get-Response	get name mdns_mode 0 on\x0d\x0a
CMD-UART Set-Request	set name mdns_mode on\x0d\x0a
CMD-UART Set-Response	set name mdns_mode 0 on\x0d\x0a

2.2.4 MDNS Server 1/2 (variable: mdns_serv1 || mdns_serv2)

Command option:	get / set
Parameter 1:	<dev_name> - Device name. Default value: pan9320 Range: String with maximum of 31 characters The device name is limited to just twenty-six letters, ten digits and the hyphen character, it is not allowed to use spaces or other punctuation.
Parameter 2:	<prot_name> - Protocol name. Default value: _http._tcp Range: String with maximum of 31 characters
Parameter 3:	<port> - Port of the service. Default value: 80 Range: 0-65535
Parameter 4:	<ttl> - Time to live of the service. Default value: 120 Range: 0-4294967295
Parameter 5:	<txt> - Additional information of the service.

	Default value: pan9320 Range: String with maximum of 31 characters
Description:	The mdns service can be configured for a individual use. By setting the <prot_name> every device can propagate his own service in the network.
GET-Rights:	0x00
SET-Rights:	0x04

Examples:

HTTP/JSON Get-Request	["get","name","mdns_serv1"]
HTTP/JSON Get-Response	["get","name","mdns_serv1","0","pan9320","_http._tcp","80","120", "PAN9320text"]
HTTP/JSON Set-Request	["set","name","mdns_serv1","pan9320","_http._tcp","80","120", "PAN9320text"]
HTTP/JSON Set-Response	["set","name","mdns_serv1","0","pan9320","_http._tcp","80","120", "PAN9320text"]
CMD-UART Get-Request	get name mdns_serv1\x0d\x0a
CMD-UART Get-Response	get name mdns_serv1 0 pan9320 _http._top 80 120 PAN9320text\x0d\x0a
CMD-UART Set-Request	set name mdns_serv1 pan9320 _http._top 80 120 PAN9320text\x0d\x0a
CMD-UART Set-Response	set name mdns_serv1 0 pan9320 _http._top 80 120 PAN9320text\x0d\x0a

2.2.5 NetBIOS Name Service Name (variable: nbns_name)

Command option:	get / set
Parameter 1:	<name> - NetBIOS Name Service name. Default value: pan9320 Range: String with a maximum size of 15 characters
Description:	The NetBIOS Name Service Name is limited to just 15 characters
GET-Rights:	0x00
SET-Rights:	0x04

Examples:

HTTP/JSON Get-Request	["get","name","nbns_name"]
HTTP/JSON Get-Response	["get","name","nbns_name","0","pan9320"]
HTTP/JSON Set-Request	["set","name","nbns_name","pan9320"]
HTTP/JSON Set-Response	["set","name","nbns_name","0","pan9320"]
CMD-UART Get-Request	get name nbns_name\x0d\x0a
CMD-UART Get-Response	get name nbns_name 0 pan9320\x0d\x0a
CMD-UART Set-Request	set name nbns_name pan9320\x0d\x0a

CMD-UART Set-Response	set name nbns_name 0 pan9320\x0d\x0a

2.2.6 NetBIOS Mode (variable: nbns_mode)

Command option:	get / set
Parameter 1:	<mode> - Enables/Disables NBNS mode. Default value: on Range: [on, off] on: mode enabled off: mode disabled
Description:	The NBNS mode is to switch on or off the NBNS service.
GET-Rights:	0x00
SET-Rights:	0x04

Examples:

HTTP/JSON Get-Request	["get","name","nbns_mode"]
HTTP/JSON Get-Response	["get","name","nbns_mode","0","on"]
HTTP/JSON Set-Request	["set","name","nbns_mode","on"]
HTTP/JSON Set-Response	["set","name","nbns_mode","0","on"]
CMD-UART Get-Request	get name nbns_mode\x0d\x0a
CMD-UART Get-Response	get name nbns_mode 0 on\x0d\x0a
CMD-UART Set-Request	set name nbns_mode on\x0d\x0a
CMD-UART Set-Response	set name nbns_mode 0 on\x0d\x0a

2.3 NET-SERVICE (MODULE: NET)

2.3.1 IP-Configuration (variable: ipcfg)

Command option:	get
Parameter 1:	<interface> - Network interface. Default value: - Range: [0, 1]
Parameter 2:	<local_ip> - IP address. Default value: - Range: -
Parameter 3:	<subnet_ip>- Subnet mask. Default value: - Range: -
Parameter 4:	<gateway_ip> - Gateway address. Default value: -

	Range: -
Parameter 5:	<dns1_ip> - DNS-Server address. Default value: - Range: -
Description:	IP configuration of an interface (0, 1). For a get-request has only the first parameter (interface) to be indicated. The get-respond in turn includes all 5 parameters.
GET-Rights:	0x00

Examples:

HTTP/JSON Get-Request	["get","net","ipcfg","0"]
HTTP/JSON Get-Response	["get","net","ipcfg","0","0","192.168.130.30","255.255.255.0", "192.168.130.1","192.168.130.1"]
CMD-UART Get-Request	get net ipcfg 0\x0d\x0a
CMD-UART Get-Response	get net ipcfg 0 0 192.168.130.30 255.255.255.0 192.168.130.1 192.168.130.1\x0d\x0a

2.3.2 Fixed IP-Address (variable: fixedip)

ommand option:	get/set
Parameter 1:	<interface> - Network interface. Default value: 1 Range: [0, 1] 0: currently only setting the ' fixedip' of interface 1 supported 1: -
Parameter 2:	<local_ip> - IP address. Default value: - Range: -
Parameter 3:	<subnet_ip> - Subnet mask. Default value: - Range: -
Parameter 4:	<gateway_ip> - Gateway address. Default value: - Range: -
Parameter 5:	<dns1_ip> - DNS-Server address. Default value: - Range: -
Description:	IP configuration of an interface(0, 1) for the fixed IP mode. 0 = Access Point 1 = Station For a get-request has only the first parameter (interface) to be indicated. The get-respond in turn includes all 5 parameters.
GET-Rights:	0x00
SET-Rights:	0x04

Examples:

HTTP/JSON Get-Request	["get","net","fixedip","1"]
HTTP/JSON Get-Response	["get","net","fixedip","0","1","192.168.130.30","255.255.255.0", "192.168.130.1","192.168.130.1"]
HTTP/JSON Set-Request	["set","net","fixedip","1","192.168.130.30","255.255.255.0", "192.168.130.1","192.168.130.1"]
HTTP/JSON Set-Response	["set","net","fixedip","0","1","192.168.130.30","255.255.255.0", "192.168.130.1","192.168.130.1"]
CMD-UART Get-Request	get net fixedip 1\x0d\x0a
CMD-UART Get-Response	get net fixedip 0 1 192.168.130.30 255.255.255.0 192.168.130.1 192.168.130.1\x0d\x0a
CMD-UART Set-Request	set net fixedip 1 192.168.130.30 255.255.255.0 192.168.130.1 192.168.130.1\x0d\x0a

CMD-UART Set-Response	set net fixedip 0 1 192.168.130.30 255.255.255.0 192.168.130.1 192.168.130.1\x0d\x0a

2.3.3 IP Mode (variable: mode)

Command option:	get/set
Parameter 1:	<interface> - Network interface. Default value: 1 Range: [0, 1] 0: Currently not supported 1: -
Parameter 2:	<mode> - IP Mode. Default value: off without any connection aip+dhcp for the first connection Range: [aip+dhcp, fip] aip+dhcp: AutoIP and DHCP fip: Fixed-IP
Description:	IP mode for the network interface.
GET-Rights:	0x00
SET-Rights:	0x04

Examples:

HTTP/JSON Get-Request	["get","net","mode","1"]
HTTP/JSON Get-Response	["get","net","mode","0","1","aip+dhcp"]
HTTP/JSON Set-Request	["set","net","mode","1","aip+dhcp"]
HTTP/JSON Set-Response	["set","net","mode","0","1","aip+dhcp"]
CMD-UART Get-Request	get net mode 1\x0d\x0a
CMD-UART Get-Response	get net mode 0 1 aip+dhcp\x0d\x0a
CMD-UART Set-Request	set net mode 1 aip+dhcp\x0d\x0a
CMD-UART Set-Response	set net mode 0 1 aip+dhcp\x0d\x0a

2.3.4 PING Request (variable: ping_req)

Command option:	set
Parameter 1:	<ip-adresse> - Destination IP-Address Default value: - Range: -
Description:	The Ping request is only the command to trigger a ping to an ip-address. The status of this request must be requested seperatly.
SET-Rights:	0x04

Examples:

HTTP/JSON Set-Request	["set","net","ping_req","192.168.1.1"]
HTTP/JSON Set-Response	["set","net","ping_req","0","192.168.1.1"]
CMD-UART Set-Request	set net ping_req 192.168.1.1\x0d\x0a
CMD-UART Set-Response	set net ping_req 0 192.168.1.1\x0d\x0a

2.3.5 PING Status (variable: ping_status)

Command option:	get
Parameter 1:	<ip-adresse> - Destination IP-Address Default value: - Range: -
Parameter 2:	<status> - Status of the ping request Default value: - Range: [0-] 0: Idle 1: Ping sending 2: Ping Success 3: Ping Error
Parameter 3:	<time> - Responce time for the ping request in [ms] Default value: - Range: -
Description:	Status for a ping request.
GET-Rights:	0x01

Examples:

HTTP/JSON Get-Request	["get","net","ping_status"]
HTTP/JSON Get-Response	["get","net","ping_status","0","192.168.1.1",”1”,”65”]
CMD-UART Get-Request	get net ping_status\x0d\x0a
CMD-UART Get-Response	get net ping_status 0 192.168.1.1 1 65\x0d\x0a

2.4 EMAIL (MODULE: SMAIL)

2.4.1 User mail address (variable: sender)

Command option:	get/set
Parameter 1:	<address> - Email address of the sender. Default value: - Range: String with maximum of 60 characters
Description:	Mail address of the sender.

GET-Rights:	0x01
SET-Rights:	0x04

Examples:

HTTP/JSON Get-Request	["get","smail","sender"]
HTTP/JSON Get-Response	["get","smail","sender","0","maxmuster@muster.de"]
HTTP/JSON Set-Request	["set","smail","sender","maxmuster@muster.de"]
HTTP/JSON Set-Response	["set","smail","sender","0","maxmuster@muster.de"]
CMD-UART Get-Request	get smail sender\x0d\x0a
CMD-UART Get-Response	get smail sender 0 maxmuster@muster.de\x0d\x0a
CMD-UART Set-Request	set smail sender maxmuster@muster.de\x0d\x0a
CMD-UART Set-Response	set smail sender 0 maxmuster@muster.de\x0d\x0a

2.4.2 Server configuration (variable: server)

Command option:	get/set
Parameter 1:	<server_name> - Name of the email server. Default value: - Range: String with maximum of 31 characters and a minimum of 4 characters
Parameter 2:	<port> - Port of the server. Default value: 25 Range: 0-65535
Parameter 3:	<login> - Login for the server. Default value: - Range: String with maximum of 31 characters
Parameter 4:	<password> Password for the login. Default value: - Range: String with maximum of 15 characters
Description:	Server configuration for email.
GET-Rights:	0x01
SET-Rights:	0x04

Examples:

HTTP/JSON Get-Request	["get","smail","server"]
HTTP/JSON Get-Response	["get","smail","server","0","MyServ","25","MyLogin"]
HTTP/JSON Set-Request	["set","smail","server","MyServ","25","MyLogin","MyPassword"]
HTTP/JSON Set-Response	["set","smail","server","0","MyServ","25","MyLogin","MyPassword"]
CMD-UART Get-Request	get smail server\x0d\x0a

CMD-UART Get-Response	get smail server 0 MyServ 25 MyLogin\x0d\x0a
CMD-UART Set-Request	set smail server MyServ 25 MyLogin MyPassword\x0d\x0a
CMD-UART Set-Response	set smail server 0 MyServ 25 MyLogin MyPassword\x0d\x0a

2.4.3 Mail sending (variable: send)

Command option:	set
Parameter 1:	<recipient> - Recipient for the email. Default value: - Range: String with maximum of 48 characters
Parameter 2:	<subject> - Subject of the email. Default value: - Range: String with maximum of 60 characters
Parameter 3:	<text> - Text of the email. Default value: - Range: String with maximum of 250 characters
Description:	Server configuration for email. The mailtext does only support printable ascii characters. A currage return new line command is not supported.
SET-Rights:	0x04

Examples:

HTTP/JSON Set-Request	["set","smail","send","receiver@web.de","subject","mail"]
HTTP/JSON Set-Response	["set","smail","send","0","receiver@web.de","subject","mail"]
CMD-UART Set-Request	set smail send receiver@web.de subject mail\x0d\x0a
CMD-UART Set-Response	set smail send 0 receiver@web.de subject mail\x0d\x0a

2.4.4 Mail status (variable: status)

Command option:	get
Parameter:	<status> - Status of the mail module Default value: 0 Range: See list 4.3.1 .
Description:	Status of the mail module
GET-Rights:	0x01

Examples:

HTTP/JSON Get-Request	["get","smail","status"]
HTTP/JSON Get-Response	["get","smail","status","0","1"]
CMD-UART Get-Request	get smail status\x0d\x0a

CMD-UART Get-Response	get smail status 0 1\x0d\x0a

2.4.5 Mail error (variable: error)

Command option:	get
Parameter:	<value> - Value of the error. Default value: 0 Range: [0-1] 0: No error active 1: Error active
Description:	Error value of the email module.
GET-Rights:	0x01

Examples:

HTTP/JSON Get-Request	["get","smail","error"]
HTTP/JSON Get-Response	["get","smail","error","0","0"]
CMD-UART Get-Request	get smail error\x0d\x0a
CMD-UART Get-Response	get smail error 0 0\x0d\x0a

2.5 SYSTEM (MODULE: SYSTEM)

2.5.1 Firmware version (variable: firmware)

Command option:	get
Description:	Returns the firmware version
GET-Rights:	0x00

Examples:

HTTP/JSON Get-Request	["get","system","firmware"]
HTTP/JSON Get-Response	["get","system","firmware","0","V_01_01_02"]
CMD-UART Get-Request	get system firmware\x0d\x0a
CMD-UART Get-Response	get system firmware 0 V_01_01_02\x0d\x0a

2.5.2 MAC address (variable: macaddr)

Command option:	get
Description:	Returns the MAC address.
GET-Rights:	0x00

Examples:

HTTP/JSON Get-Request	["get","system","macaddr"]
HTTP/JSON Get-Response	["get","system","macaddr","0","0f:0e:03:02:01:00"]
CMD-UART Get-Request	get system macaddr\x0d\x0a
CMD-UART Get-Response	get system macaddr 0 0f:0e:03:02:01:00\x0d\x0a

2.5.3 Serial number (variable: serialnum)

Command option:	get
Description:	Returns the serial number
GET-Rights:	0x00

Examples:

HTTP/JSON Get-Request	["get","system","serialnum"]
HTTP/JSON Get-Response	["get","system","serialnum","0","Ser00001"]
CMD-UART Get-Request	get system serialnum\x0d\x0a
CMD-UART Get-Response	get system serialnum 0 Ser00001\x0d\x0a

2.5.4 WiFi firmware version (variable: wifi_ver)

Command option:	get
Description:	Returns the firmware version of the WiFi module
GET-Rights:	0x00

Examples:

HTTP/JSON Get-Request	["get","system","wifi_ver"]
HTTP/JSON Get-Response	["get","system","wifi_ver","0","V_03_02_01"]
CMD-UART Get-Request	get system wifi_ver\x0d\x0a
CMD-UART Get-Response	get system wifi_ver 0 V_03_02_01\x0d\x0a

2.5.5 Bootloader version (variable: bootl_ver)

Command option:	get
Description:	Returns the bootloader version of the WiFi module
GET-Rights:	0x00

Examples:

HTTP/JSON Get-Request	["get","system","bootl_ver"]
HTTP/JSON Get-Response	["get","system","bootl_ver","0","V_01_02_01"]
CMD-UART Get-Request	get system bootl_ver\x0d\x0a
CMD-UART Get-Response	get system bootl_ver 0 V_01_02_01\x0d\x0a

2.5.6 Hardware revision (variable: hwrev)

Command option:	get
Description:	Returns the hardware revision number
GET-Rights:	0x00

Examples:

HTTP/JSON Get-Request	["get","system","hwrev"]
HTTP/JSON Get-Response	["get","system","hwrev","0","Rev.01"]
CMD-UART Get-Request	get system hwrev\x0d\x0a
CMD-UART Get-Response	get system hwrev 0 Rev.01\x0d\x0a

2.5.7 Restart (variable: restart)

Command option:	set
Parameter:	-
Range	-
Description:	This parameter triggers a restart of the Hardware
SET-Rights:	0x04

Examples:

HTTP/JSON Set-Request	["set","system","restart"]
HTTP/JSON Set-Response	["set","system","restart","0"]
CMD-UART Set-Request	set system restart\x0d\x0a
CMD-UART Set-Response	set system restart 0\x0d\x0a

2.5.8 Reset (variable: factory)

Command option:	set
Parameter:	-
Range	-
Description:	This parameter makes a factory reset of the software. After the factory reset the hardware will be restarted
SET-Rights:	0x04

Examples:

HTTP/JSON Set-Request	["set","system","factory"]
HTTP/JSON Set-Response	["set","system","factory","0"]
CMD-UART Set-Request	set system factory\x0d\x0a
CMD-UART Set-Response	set system factory 0\x0d\x0a

2.5.9 Save mode (variable: savemode)

Command option:	set
Parameter 1:	<savemode> - The save mode can set to manual or automatic. Default value: auto Range: auto or manual
Description:	Manual means the user has to trigger a save of the current configuration. Automatic means, the configuration will be saved after every change automatically.
SET-Rights:	0x04

Examples:

HTTP/JSON Set-Request	["set","system","savemode",“auto”]
HTTP/JSON Set-Response	["set","system","savemode”,"0",”auto”]
HTTP/JSON Get-Request	["get","system","savemode"]
HTTP/JSON Get-Response	["get","system","savemode","0","auto"]
CMD-UART Set-Request	set system savemode auto\x0d\x0a
CMD-UART Set-Response	set system savemode 0 auto\x0d\x0a
CMD-UART Get-Request	get system savemode\x0d\x0a
CMD-UART Get-Response	get system savemode 0 auto\x0d\x0a

2.5.10 Save the configuration (variable: savecfg)

Command option:	set
Parameter:	-
Range	-
Description:	This command triggers saving of the configuration
SET-Rights:	0x02

Examples:

HTTP/JSON Set-Request	["set","system","savecfg"]
HTTP/JSON Set-Response	["set","system","savecfg”,"0"]
CMD-UART Set-Request	set system savecfg\x0d\x0a
CMD-UART Set-Response	set system savecfg 0\x0d\x0a

2.5.11 Power save mode (variable: psm)

Command option:	set
Parameter 1:	<mode> - Power save mode Default value: - Range: [0,1,2]

	0 = Disable power save mode 1 = IEEE power save modus 2 = Power down
Parameter 2: (**only for mode 1)**	<dtims> - Multiple dtims (16Bit) Default value: - Range: [0...65535] 65534: CLOSEST_DTIM_TO_LISTEN_INTERVAL)
Parameter 3: **(only for mode 1)**	<BeaconTimeout> - Beacon miss timeout (16Bit) Default value: - Range: [0...65535]
Parameter 4: **(only for mode 1)**	<ListInt>- Local listen interval (16Bit) Default value: - Range: [0...65535]
Parameter 5: **(only for mode 1)**	<AdhocWakePeriod> - Adhoc wake period (16Bit) Default value: - Range: [0...65535]
Parameter 6: **(only for mode 1)**	<delay> - Delay to powersave (16Bit) Default value: - Range: [0...65535]
Description:	Sets the MC200 and the radio chips power save mode
SET-Rights:	0x04

Examples:

HTTP/JSON Set-Request	["set","system","psm",”1”]
HTTP/JSON Set-Response	["set","system","psm”,"0",”1”]
CMD-UART Set-Request	set system psm 1\x0d\x0a
CMD-UART Set-Response	set system psm 0 1\x0d\x0a
CMD-UART Set-Request	set system psm 1 65534 5 0 0 0\x0d\x0a
CMD-UART Set-Response	set system psm 0 1 65534 5 0 0 0\x0d\x0a

2.6 USER MANAGMENT (MODULE: USER)

2.6.1 User edit/add (variable: edit)

Command option:	set
Parameter 1:	<index#> - User index number. Default value: - Range: [0-4]
Parameter 2:	<name> - Login name. Default value: - Range: String with maximum of 29 characters
Parameter 3:	<password> - Password for login.

	Default value: - Range: String with maximum of 29 characters
Parameter 4:	<rights> - Rights of the user. Default value: - Range: [0..255] See list 4.4.1
Description:	Sets the user with login name, password and rights.
SET-Rights:	0x80

Examples:

HTTP/JSON Set-Request	["set","user","edit","1","Max","password","61"]
HTTP/JSON Set-Response	["set","user","edit","0","1","Max","password","61"]
CMD-UART Set-Request	set user edit 1 Max password 61\x0d\x0a
CMD-UART Set-Response	set user edit 0 1 Max password 61\x0d\x0a

2.6.2 User name (variable: name)

Command option:	get
Parameter 1:	<index#> - User index number. Default value: - Range: [0..4]
Parameter 2:	<name#> - User name Default value: - Range: -
Parameter 3:	<rights#> - User rights Default value: - Range: [0..255] See list 4.4.1
Description:	Returns the user name for the login.
GET-Rights:	0x01

Examples:

HTTP/JSON Get-Request	["get","user","name","1"]
HTTP/JSON Get-Response	["get","user","name","0","1","Max","61"]
CMD-UART Get-Request	get user name 1\x0d\x0a
CMD-UART Get-Response	get user name 0 1 Max 61\x0d\x0a

2.6.3 User delete (variable: delete)

Command option:	set
Parameter:	<index#> - User index number. Default value: -

	Range: [1-4]
Description:	Deletes an user. The user 0 can not be delete.
SET-Rights:	0x80

Examples:

HTTP/JSON Set-Request	["set","user","delete","1"]
HTTP/JSON Set-Response	["set","user","delete","0","1"]
CMD-UART Set-Request	set user delete 1\x0d\x0a
CMD-UART Set-Response	set user delete 0 1\x0d\x0a

2.6.4 Default login active (variable: dlogin)

Command option:	get
Description:	Returns '0' for default login is still set. Returns '1' for login has been changed.
GET-Rights:	0x01

Examples:

HTTP/JSON Get-Request	["get","user","dlogin"]
HTTP/JSON Get-Response	["get","user","dlogin","0","1"]
CMD-UART Get-Request	get user dlogin\x0d\x0a
CMD-UART Get-Response	get user dlogin 0 1\x0d\x0a

2.6.5 Default Rights (variable: drights)

Command option:	get/set
Parameter 1:	<drights> - Parameter default rights if no authentification was requested Default value: 0 Range: 0 - 255
Description:	This parameter is used for default rights, if no authentification is need for the files.
GET-Rights:	0x01
SET-Rights:	0x80

Examples:

HTTP/JSON Get-Request	["get","user","drights"]
HTTP/JSON Get-Response	["get","user","drights","0","0"]
CMD-UART Get-Request	get user drights\x0d\x0a
CMD-UART Get-Response	get user drights 0 0\x0d\x0a

HTTP/JSON Set-Request	["set","user","drights","1"]
HTTP/JSON Set-Response	["set","user","drights","0","1"]
CMD-UART Set-Request	set user drights 1\x0d\x0a
CMD-UART Set-Response	set user drights 0 1\x0d\x0a

2.6.6 My rights (variable: myrights)

Command option:	get
Parameter 1:	<myrights> - Returns the current rights Default value: 0 Range: 0 - 255
Description:	This parameter is used to display the current rights for a loged in user
GET-Rights:	0x00

Examples:

HTTP/JSON Get-Request	["get","user","myrights"]
HTTP/JSON Get-Response	["get","user","myrights","0","3"]
CMD-UART Get-Request	get user myrights\x0d\x0a
CMD-UART Get-Response	get user myrights 0 3\x0d\x0a

2.7 TELNET (MODULE: TELNET)

2.7.1 Mode (variable: mode)

Command option:	get/set
Parameter 1:	<mode> - Mode for telnet Default value: on Range: 'on' or 'off'
Description:	Returns the telnet mode and this parameter can set the telnet mode to 'on' or 'off'
GET-Rights:	0x01
SET-Rights:	0x04

Examples:

HTTP/JSON Get-Request	["get","telnet","mode"]
HTTP/JSON Get-Response	["get","telnet","mode","0","on"]
CMD-UART Get-Request	get telnet mode\x0d\x0a

CMD-UART Get-Response	get telnet mode 0 on\x0d\x0a
HTTP/JSON Set-Request	["set","telnet","mode","off"]
HTTP/JSON Set-Response	["set","telnet","mode","0","off"]
CMD-UART Set-Request	set telnet mode off\x0d\x0a
CMD-UART Set-Response	set telnet mode 0 off\x0d\x0a

2.7.2 Auth(variable: auth)

Command option:	get/set
Parameter 1:	<auth> - Authentication for telnet enabled / disabled Default value: on Range: 'on' or 'off'
Description:	With this parameter the telnet authentication can be enabled or disabled
GET-Rights:	0x01
SET-Rights:	0x80

Examples:

HTTP/JSON Get-Request	["get","telnet","auth"]
HTTP/JSON Get-Response	["get","telnet","auth","0","on"]
CMD-UART Get-Request	get telnet auth\x0d\x0a
CMD-UART Get-Response	get telnet auth 0 on\x0d\x0a
HTTP/JSON Set-Request	["set","telnet","auth","off"]
HTTP/JSON Set-Response	["set","telnet","auth","0","off"]
CMD-UART Set-Request	set telnet auth off\x0d\x0a
CMD-UART Set-Response	set telnet auth 0 off\x0d\x0a

2.8 FIRMWARE UPDATE (MODULE: FWU)

2.8.1 Progress (variable: progress)

Command option:	get
Parameter:	<progress> - The firmware update progress. Default value: - Range: [0..100] in %
Description:	The command returns the progress of the firmware update process.

GET-Rights:	0x01

Examples:

HTTP/JSON Get-Request	["get","fwu","progress"]
HTTP/JSON Get-Response	["get","fwu","progress","0","50"]
CMD-UART Get-Request	get fwu progress\x0d\x0a
CMD-UART Get-Response	get fwu progress 0 50\x0d\x0a

2.8.2 Status (variable: status)

Command option:	get
Parameter:	<status> - The firmware update status. Default value: 0 Range: See list 4.5.1
Description:	Returns the status of the firmware update process.
GET-Rights:	0x01

Examples:

HTTP/JSON Get-Request	["get","fwu","status"]
HTTP/JSON Get-Response	["get","fwu","status","0","0"]
CMD-UART Get-Request	get fwu status\x0d\x0a
CMD-UART Get-Response	get fwu status 0 0\x0d\x0a

2.8.3 Check for firmware version (variable: checkfw)

Command option:	get / set
Parameter:	<firmware> - Is only returned in a get-request, is the answer from the firmware update server Default value: - Range: -
Description:	The command returns a information from the firmware update server. The firmware update server is responsible for the answer. With the Set-Request a request is triggered to get the current firmware version on the server. With the Get-Request the answer from the server can be requested.
GET-Rights:	0x01
SET-Rights:	0x04

Examples:

HTTP/JSON Get-Request	["get","fwu","checkfw"]

HTTP/JSON Get-Response	["get","fwu","checkfw",”0”,"V_01_00_07"]
HTTP/JSON Set-Request	["set","fwu","checkfw"]
HTTP/JSON Set-Response	["set","fwu","checkfw","0"]
CMD-UART Get-Request	get fwu checkfw\x0d\x0a
CMD-UART Get-Response	get fwu checkfw 0 V_01_00_07\x0d\x0a
CMD-UART Set-Request	set fwu checkfw\x0d\x0a
CMD-UART Set-Response	set fwu checkfw 0\x0d\x0a

2.8.4 Firmware Update (variable: fwupdate)

Command option:	set
Parameter:	<version> - Version number for the update. If this parameter is emtpy, the latest version will be installed. Default value: - Range: -
Description:	With this parameter a firmware update via HTTP-Server is executed.
SET-Rights:	0x40

Examples:

HTTP/JSON Set-Request	["set","fwu","fwupdate"]
HTTP/JSON Set-Response	["set","fwu","fwupdate","0"]
CMD-UART Set-Request	set fwu fwupdate\x0d\x0a
CMD-UART Set-Response	set fwu fwupdate 0\x0d\x0a

2.9 COMMANDO UART (MODULE: CMDUART)

2.9.1 Send (variable: send)

Command option:	set
Parameter:	<data> - Data which are send over the uart to set something Default value: - Range:
Description:	The data are send over the uart
SET-Rights:	0x08 or 0x10

Examples:

HTTP/JSON Get-Request	["set","cmduart","send",”example”]
HTTP/JSON Get-Response	["set","cmduart","send","0","example"]
CMD-UART Get-Request	set cmduart send example\x0d\x0a
CMD-UART Get-Response	set cmduart send 0 example\x0d\x0a

2.9.2 Request (variable: request)

Command option:	get
Parameter:	<data> - Data which are send over the uart to request something Default value: - Range: -
Description:	The data are send over the uart
GET-Rights:	0x08 or 0x10

Examples:

HTTP/JSON Get-Request	["get","cmduart","request",”temp”]
HTTP/JSON Get-Response	["get","cmduart","request","0","temp=100"]
CMD-UART Get-Request	get cmduart request temp\x0d\x0a
CMD-UART Get-Response	get cmduart request 0 temp=100\x0d\x0a

2.9.3 Uart configuration (variable: cfg)

Command option:	set/get
Parameter 1:	<baudrate> - Baudrate for the uart Default value: 115200 Range: see list 3.2.1
Parameter 2:	<data_bits> - Data bit for the uart, this parameter can not be set, it will always be 8. Default value: 8 Range: [5-8] see list 4.7.1
Parameter 3:	<parity> - Parity bit for the uart, this parameter can not be set, it will always be 0. Default value: 0 Range: [0-4] see list 4.7.2
Parameter 4:	<stopbit> - Stopbits for the uart, this parameter can not be set, it will always be 1. Default value: 1 Range: [1-2] see list 4.7.3
Parameter 5	<hwhandshake> - Hardware handshake enabled or disabled. For the command uart the hardware handshake will be always disabled (0) Default : 0
Description:	Uart configuration for the netcat module
GET-Rights:	0x01
SET-Rights:	0x04

Examples:

HTTP/JSON Get-Request	["get","cmduart","cfg"]
HTTP/JSON Get-Response	["get","cmduart","cfg","0","115200","8","0","1","0"]
HTTP/JSON Set-Request	["set","cmduart","cfg","115200","8","0","1","0"]
HTTP/JSON Set-Response	["set","cmduart","cfg","0","115200","8","0","1","0"]
CMD-UART Get-Request	get cmduart cfg\x0d\x0a
CMD-UART Get-Response	get cmduart cfg 0 115200 8 0 1 0\x0d\x0a
CMD-UART Set-Request	set cmduart cfg 115200 8 0 1 0\x0d\x0a
CMD-UART Set-Response	set cmduart cfg 0 115200 8 0 1 0\x0d\x0a

2.9.4 Send user rights via UART (variable: sendauth)

Command option:	set/get
Parameter 1:	<sendauth> - Flag for sending additional the current user rights via UART Default value: off Range: on / off
Description:	If this flag is on the command request and send for the command uart are additional send the current user rights. See 3.1.1
GET-Rights:	0x01
SET-Rights:	0x80

Examples:

HTTP/JSON Get-Request	["get","cmduart","sendauth"]
HTTP/JSON Get-Response	["get","cmduart","sendauth","0","off"]
HTTP/JSON Set-Request	["set","cmduart","sendauth","off"]
HTTP/JSON Set-Response	["set","cmduart","sendauth","0","off"]
CMD-UART Get-Request	get cmduart sendauth \x0d\x0a
CMD-UART Get-Response	get cmduart sendauth 0 off\x0d\x0a
CMD-UART Set-Request	set cmduart sendauth off\x0d\x0a
CMD-UART Set-Response	set cmduart sendauth 0 off\x0d\x0a

2.10 GPIO (MODULE: GPIO)

2.10.1 Mode (variable: mode)

Command option:	get/set
Parameter 1:	<pin#> - A pin of the GPIO.

	Default value: - Range: See list 4.8
Parameter 2:	<mode> - Mode of a GPIO pin. Default value: out Range: [in, out] in: input out: output
Description:	Gets/Sets the mode of a specified GPIO pin.
GET-Rights:	0x01
SET-Rights:	0x04

Examples:

HTTP/JSON Get-Request	["get","gpio","mode","4"]
HTTP/JSON Get-Response	["get","gpio","mode","0","4","in"]
HTTP/JSON Set-Request	["set","gpio","mode","4","in"]
HTTP/JSON Set-Response	["set","gpio","mode","0","4","in"]
CMD-UART Get-Request	get gpio mode 4\x0d\x0a
CMD-UART Get-Response	get gpio mode 0 4 in\x0d\x0a
CMD-UART Set-Request	set gpio mode 4 in\x0d\x0a
CMD-UART Set-Response	set gpio mode 0 4 in\x0d\x0a

2.10.2 State (variable: state)

Command option:	get/set
Parameter 1:	<pin#> - A pin of the GPIO. Default value: - Range: See list 4.8
Parameter 2:	<state> - Enables/Disables a GPIO pin. Default value: off Range: [on, off] on: Enabled off: Disabled
Description:	Enables/Disables a specified GPIO pin.
GET-Rights:	0x01
SET-Rights:	0x04

Examples:

HTTP/JSON Get-Request	["get","gpio","state","4"]
HTTP/JSON Get-Response	["get","gpio","state","0","4","off"]

HTTP/JSON Set-Request	["set","gpio","state","4","off"]
HTTP/JSON Set-Response	["set","gpio","state","0","4","off"]
CMD-UART Get-Request	get gpio state 4\x0d\x0a
CMD-UART Get-Response	get gpio state 0 4 off\x0d\x0a
CMD-UART Set-Request	set gpio state 4 off\x0d\x0a
CMD-UART Set-Response	set gpio state 0 4 off\x0d\x0a

2.11 HTTP-CLIENT (MODULE: HTTPC)

2.11.1 Status (variable: status)

Command option:	get
Parameter 1:	<status> - HTTP-Client status. Default value: 0 Range: [0-2] See 4.6.1
Description:	Returns the status of the HTTP-Client.
GET-Rights:	0x01

Examples:

HTTP/JSON Get-Request	["get","httpc","status"]
HTTP/JSON Get-Response	["get","httpc","status","0","1"]
CMD-UART Get-Request	get httpc status\x0d\x0a
CMD-UART Get-Response	get httpc status 0 1\x0d\x0a

2.11.2 HTTP-Server (variable: server)

Command option:	get/set
Parameter 1:	<server-adr> - HTTP-Server address. Default value: - Range: String with maximum of 39 characters
Parameter 2:	<port> - HTTP-Server port number. Default value: - Range: [0-65535]
Parameter 3:	<ressource> - HTTP-Server ressource. Default value: - Range: String with maximum of 99 characters
Parameter 4:	<username> - HTTP-Server username. Default value: - Range: String with maximum of 29 characters
Parameter 5:	<password> - HTTP-Server password.

	Default value: - Range: String with maximum of 19 characters
Description:	Returns the status of the HTTP-Client.
GET-Rights:	0x01
SET-Rights:	0x04

Examples:

HTTP/JSON Get-Request	["get","httpc","server"]
HTTP/JSON Get-Response	["get","httpc","server","0","ServerAdr","80","ressource","username"]
HTTP/JSON Set-Request	["set","httpc","server","ServerAdr","80","ressource","username", "password"]
HTTP/JSON Set-Response	["set","httpc","server","0","ServerAdr","80","ressource","username", "password"]
CMD-UART Get-Request	get httpc server\x0d\x0a
CMD-UART Get-Response	get httpc server 0 ServerAdr 80 ressource username\x0d\x0a
CMD-UART Set-Request	set httpc server ServerAdr 80 ressource username password\x0d\x0a
CMD-UART Set-Response	set httpc server 0 ServerAdr 80 ressource username password\x0d\x0a

2.11.3 POST (variable: post)

Command option:	set
Parameter 1:	<data> - Data which should be send Default value: - Range: String with maximum of 299 characters
Description:	Data which should be send to the server. To Send the data a vaild server must be configured. Also a connection to this server must be avaiable
GET-Rights:	0x01
SET-Rights:	0x04

Examples:

HTTP/JSON Set-Request	["set","httpc","post","data"]
HTTP/JSON Set-Response	["set","httpc","post","0","data"]
CMD-UART Set-Request	set httpc post data\x0d\x0a
CMD-UART Set-Response	set httpc post 0 data\x0d\x0a

2.11.4 GET (variable: get)

Command option:	get / set

Parameter 1:	<data> - Data which should be send Default value: - Range: String with maximum of 299 characters
Description:	Data which should be send to the server
GET-Rights:	0x01
SET-Rights:	0x04

Examples:

HTTP/JSON Get-Request	["get","httpc","get"]
HTTP/JSON Get-Response	["get","httpc","get","0","data"]
HTTP/JSON Set-Request	["set","httpc","get"]
HTTP/JSON Set-Response	["set","httpc","get","0"]
CMD-UART Get-Request	get httpc get\x0d\x0a
CMD-UART Get-Response	get httpc get 0\x0d\x0a
CMD-UART Set-Request	set httpc get\x0d\x0a
CMD-UART Set-Response	set httpc get 0\x0d\x0a

2.11.5 CMDAPI(variable: cmdapi)

Command option:	set
GET-Rights:	0x01
SET-Rights:	0x04

Examples:

HTTP/JSON Get-Request	["set","httpc","cmdapi"]
HTTP/JSON Get-Response	["set","httpc","cmdapi","0"]
CMD-UART Get-Request	set httpc cmdapi\x0d\x0a
CMD-UART Get-Response	set httpc cmdapi 0\x0d\x0a

2.11.6 TOKEN (variable: token)

Command option:	get/set
Parameter 1:	<token> - server token for the client Default value: - Range: 19 characters
Description:	Token from the server for the client
GET-Rights:	0x01
SET-Rights:	0x04

Examples:

HTTP/JSON Get-Request	["get","httpc","token"]
HTTP/JSON Get-Response	["get","httpc","token","0",”12345abc”]
CMD-UART Get-Request	get httpc token\x0d\x0a
CMD-UART Get-Response	get httpc token 0 12345abc\x0d\x0a
HTTP/JSON Set-Request	["set","httpc","token","12345abc"]
HTTP/JSON Set-Response	["set","httpc","token","0","12345abc"]
CMD-UART Set-Request	set httpc token 12345abc\x0d\x0a
CMD-UART Set-Response	set httpc token 0 12345abc\x0d\x0a

2.12 NETCAT (MODULE: NETCAT)

2.12.1 State (variable: state)

Command option:	get / set
Parameter 1:	<state> - Starts or Stops a TCP-Listen or a TCP-Connect Default value: off Range: [on, off]
Description:	The variable can start or stop a TCP-Listen or a TCP-Connect depends on the mode parameter.
GET-Rights:	0x01
SET-Rights:	0x04

Examples:

HTTP/JSON Get-Request	["get","netcat","state"]
HTTP/JSON Get-Response	["get","netcat","state","0","on"]
HTTP/JSON Set-Request	["set","netcat","state","off"]
HTTP/JSON Set-Response	["set","netcat","post","0","off"]
CMD-UART Get-Request	get netcat state\x0d\x0a
CMD-UART Get-Response	get netcat state 0 on\x0d\x0a
CMD-UART Set-Request	set netcat state off\x0d\x0a
CMD-UART Set-Response	set netcat state 0 off\x0d\x0a

2.12.2 Server IP (variable: serv_ip)

Command option:	get/set
Parameter 1:	<server-adr> - Server IP address Default value: - Range: -

Description:	Is the server IP address for a TCP-Connect with the netcat module.
GET-Rights:	0x01
SET-Rights:	0x04

Examples:

HTTP/JSON Get-Request	["get","netcat","serv_ip"]
HTTP/JSON Get-Response	["get","netcat","serv_ip","0","125.69.12.1"]
HTTP/JSON Set-Request	["set","netcat","serv_ip","125.69.12.1"]
HTTP/JSON Set-Response	["set","netcat","serv_ip","0","125.69.12.1"]
CMD-UART Get-Request	get netcat serv_ip\x0d\x0a
CMD-UART Get-Response	get netcat serv_ip 0 125.69.12.1\x0d\x0a
CMD-UART Set-Request	set netcat serv_ip 125.69.12.1\x0d\x0a
CMD-UART Set-Response	set netcat serv_ip 0 125.69.12.1\x0d\x0a

2.12.3 Server Login (variable: serv_login)

Command option:	set/get
Parameter 1:	<username> - Username on the server for the login Default value: - Range: String with maximum of 29 characters
Parameter 2:	<password> - Password on the server for the login. The password will not be return with a get-request. Default value: - Range: String with maximum of 29 characters
Description:	Login for the Server for netcat
GET-Rights:	0x01
SET-Rights:	0x04

Examples:

HTTP/JSON Get-Request	["get","netcat","serv_login"]
HTTP/JSON Get-Response	["get","netcat","serv_login","0","username"]
HTTP/JSON Set-Request	["set","netcat","serv_login","username","password"]
HTTP/JSON Set-Response	["set","netcat","serv_login","0","username","password"]
CMD-UART Get-Request	get netcat serv_login\x0d\x0a
CMD-UART Get-Response	get netcat serv_login 0 username\x0d\x0a
CMD-UART Set-Request	set netcat serv_login username password\x0d\x0a
CMD-UART Set-Response	set netcat serv_login 0 username password\x0d\x0a

2.12.4 Telnet option (variable: telopt)

Command option:	set/get
Parameter 1:	<option> - on or off Default value: off Range: on/off
Description:	Enables or disables the telnet option for netcat. All telnet options can be found at chapter **Fehler! Verweisquelle konnte nicht gefunden werden. Fehler! Verweisquelle konnte nicht gefunden werden.**
GET-Rights:	0x01
SET-Rights:	0x04

Examples:

HTTP/JSON Get-Request	["get","netcat","telopt]
HTTP/JSON Get-Response	["get","netcat","telopt","0",”on”]
HTTP/JSON Set-Request	["set","netcat","telopt","off”]
HTTP/JSON Set-Response	["set","netcat","telopt,”0”,"off”]
CMD-UART Get-Request	get netcat telopt\x0d\x0a
CMD-UART Get-Response	get netcat telopt 0 on\x0d\x0a
CMD-UART Set-Request	set netcat telopt off\x0d\x0a
CMD-UART Set-Response	set netcat telopt 0 off\x0d\x0a

2.12.5 TCP Port (variable: tcp_port)

Command option:	set/get
Parameter 1:	<port> - port for a tcp connection (listen and connect) Default value: - Range: 1-65535
Description:	TCP port for a TCP-Listen or a TCP-Connect
GET-Rights:	0x01
SET-Rights:	0x04

Examples:

HTTP/JSON Get-Request	["get","netcat","tcp_port]
HTTP/JSON Get-Response	["get","netcat","tcp_port","0",”12345”]
HTTP/JSON Set-Request	["set","netcat","tcp_port","12345”]
HTTP/JSON Set-Response	["set","netcat","tcp_port,”0”,"12345”]

CMD-UART Get-Request	get netcat tcp_port\x0d\x0a
CMD-UART Get-Response	get netcat tcp_port 0 12345\x0d\x0a
CMD-UART Set-Request	set netcat tcp_port 12345\x0d\x0a
CMD-UART Set-Response	set netcat tcp_port 0 12345\x0d\x0a

2.12.6 Mode(variable: mode)

Command option:	set/get
Parameter 1:	<mode> - client or server Default value: client Range: client or server
Description:	The mode can be set to "client" or "server", In the client mode the netcat module is connecting to a server. In the server mode the netcat module is the server and a client can connect establish a connection.
GET-Rights:	0x01
SET-Rights:	0x04

Examples:

HTTP/JSON Get-Request	["get","netcat","mode"]
HTTP/JSON Get-Response	["get","netcat","mode","0","client"]
HTTP/JSON Set-Request	["set","netcat","mode","server"]
HTTP/JSON Set-Response	["set","netcat","mode,"0","server"]
CMD-UART Get-Request	get netcat mode\x0d\x0a
CMD-UART Get-Response	get netcat mode 0 client\x0d\x0a
CMD-UART Set-Request	set netcat mode server\x0d\x0a
CMD-UART Set-Response	set netcat mode 0 server\x0d\x0a

2.12.7 Exclusive lock(variable: lock)

Command option:	get
Parameter 1:	<lock ID> - Handle ID for the exclusive lock Default value: - Range: 0 - 65535
Description:	To lock the communication a user can request a handle ID, this handle ID must be added to the wirte-request or the read-request.
GET-Rights:	0x01

Examples:

HTTP/JSON Get-Request	["get","netcat","lock"]
HTTP/JSON Get-Response	["get","netcat","lock","0","15252"]
CMD-UART Get-Request	get netcat lock\x0d\x0a
CMD-UART Get-Response	get netcat lock 0 15252\x0d\x0a

2.12.8 Release lock(variable: release)

Command option:	set
Parameter 1:	<lock ID> - Handle ID to remove the exclusive lock Default value: - Range: 0 - 65535
Description:	Removes the exclusive lock, which was created with the "lock" parameter. The lock will also be release after 5 minutes without any data activity.
SET-Rights:	0x04

Examples:

HTTP/JSON Set-Request	["set","netcat","release","15252"]
HTTP/JSON Set-Response	["set","netcat","release","0","15252"]
CMD-UART Set-Request	set netcat release 15252\x0d\x0a
CMD-UART Set-Response	set netcat release 0 15252\x0d\x0a

2.12.9 Authentication(variable: auth)

Command option:	set/get
Parameter 1:	<auth> - on/off Default value: on Range: on or off
Description:	Enables or disables the authentication for netcat
GET-Rights:	0x01
SET-Rights:	0x80

Examples:

HTTP/JSON Get-Request	["get","netcat","auth"]
HTTP/JSON Get-Response	["get","netcat","auth","0","on"]
HTTP/JSON Set-Request	["set","netcat","auth","off"]

HTTP/JSON Set-Response	["set","netcat","auth,"0","off"]
CMD-UART Get-Request	get netcat auth\x0d\x0a
CMD-UART Get-Response	get netcat auth 0 on\x0d\x0a
CMD-UART Set-Request	set netcat auth off\x0d\x0a
CMD-UART Set-Response	set netcat auth 0 off\x0d\x0a

2.12.10 DTR-Pin(variable: dtrpin)

Command option:	set/get
Parameter 1:	<pin> - DTR pin for the netcat uart Default value: - Range: Available pins see 4.8
Description:	Set the DTR pin for the netcat UART. This Pin set set to low or high The dtr-pin is a pin which can be configured via telnet-option. The dtr pin can be set to high or low via a telnet command. For example see **Fehler! Verweisquelle konnte nicht gefunden werden.**
GET-Rights:	0x01
SET-Rights:	0x04

Examples:

HTTP/JSON Get-Request	["get","netcat","dtrpin"]
HTTP/JSON Get-Response	["get","netcat","dtrpin","0","4"]
HTTP/JSON Set-Request	["set","netcat","dtrpin","4"]
HTTP/JSON Set-Response	["set","netcat","dtrpin,"0","4"]
CMD-UART Get-Request	get netcat dtrpin\x0d\x0a
CMD-UART Get-Response	get netcat dtrpin 0 4\x0d\x0a
CMD-UART Set-Request	set netcat dtrpin 4\x0d\x0a
CMD-UART Set-Response	set netcat dtrpin 0 4\x0d\x0a

2.13 UART BINARY (MODULE: BINUART)

2.13.1 Uart configuration (variable: cfg)

Command option:	set/get
Parameter 1:	<baudrate> - Baudrate for the uart Default value: 115200 Range: See baudrate list 3.2
Parameter 2:	<data_bits> - Data bit for the uart, this parameter can not be

	set, it will always be 8. Default value: 8 Range: [5-8] see list 4.7.1
Parameter 3:	<parity> - Parity bit for the uart, this parameter can not be set, it will always be 0. Default value: 0 Range: [0-4] see list 4.7.2
Parameter 4:	<stopbit> - Stopbits for the uart, this parameter can not be set, it will always be 1. Default value: 1 Range: [1-2] see list 4.7.3
Parameter 5	<hwhandshake> - Hardware handshake enabled or disabled Default : 1 Range: [0-1]
Description:	Uart configuration for the netcat module.
GET-Rights:	0x01
SET-Rights:	0x04

Examples:

HTTP/JSON Get-Request	["get","binuart","cfg"]
HTTP/JSON Get-Response	["get","binuart","cfg","0","115200","8","0","1","1"]
HTTP/JSON Set-Request	["set","binuart","cfg","115200","8","0","1","1"]
HTTP/JSON Set-Response	["set","binuart","cfg","0","115200","8","0","1","1"]
CMD-UART Get-Request	get binuart cfg\x0d\x0a
CMD-UART Get-Response	get binuart cfg 0 115200 8 0 1 1\x0d\x0a
CMD-UART Set-Request	set binuart cfg 115200 8 0 1 1\x0d\x0a
CMD-UART Set-Response	set binuart cfg 0 115200 8 0 1 1\x0d\x0a

2.13.2 Read Data(variable: data)

Command option:	read
Parameter 2:	<lock ID> - Handle ID Default value: - Range: 0 - 65535
Parameter 1:	<data> - data which is received Default value: - Range: Maximum 150 binary values as two ascii-hex values
Description:	Reads out the binary uart. If there is no exclusive lock (no handle ID) the parameter could be set to "0" or "".

READ-Rights:	0x20

Examples:

HTTP/JSON Read-Request	["read","binuart","data",”1234”]
HTTP/JSON Read-Response	["read","binuart","data","0",1234.”data”]
CMD-UART Read-Request	read binuart data 1234\x0d\x0a
CMD-UART Read-Response	read binuart data 0 1234 data\x0d\x0a

2.13.3 Write Data(variable: data)

Command option:	write
Parameter 1:	<lock ID> - Handle ID Default value: - Range: 0 - 65535
Parameter 2:	<data> - data to write Default value: - Range: Maximum 300 characters
Description:	Write data to the uart. If there is no exclusive lock (no handle ID) the parameter must be set to “0”.
WRITE-Rights:	0x20

Examples:

HTTP/JSON Write-Request	["write","binuart","data",”1234”,”64656667”]
HTTP/JSON Write-Response	["write","binuart","data",”1234”,"0",”64656667”]
CMD-UART Write-Request	write binuart data 1234 64656667\x0d\x0a
CMD-UART Write-Response	write binuart data 0 1234 64656667\x0d\x0a

2.14 TIME (MODULE: TIME)

2.14.1 UTC (variable: utc)

Command option:	get
Parameter 1:	<utc> - date and time of the utc Default value: - Range: UTC
Description:	UTC with the ISO 8601 time format Year-Month-DayTHour:Minute:Seconds 2015-08-31T11:12:13
GET-Rights:	0x00

Examples:

HTTP/JSON Get-Request	["get","time","utc"]
HTTP/JSON Get-Response	["get","time","utc","0","2015-08-31T11:12:13"]
CMD-UART Get-Request	get time utc\x0d\x0a
CMD-UART Get-Response	get time utc 2015-08-31T11:12:13\x0d\x0a

2.14.2 Time Sync (variable: sync)

Command option:	get/set
Parameter 1:	<sync> - timeout value for NTP time update Default value: 3600 seconds Range: 0 -> no NTP update Range: Minimum: 3600 seconds Maximum: 4294967294 seconds
Description:	Setting the timeout for NTP updates. Setting the value to 0 results in no updates for the time.
GET-Rights:	0x01

Examples:

HTTP/JSON Get-Request	["get","time","sync"]
HTTP/JSON Get-Response	["get","time","sync","0","3600"]
HTTP/JSON Set-Request	["set","time","sync","3600"]
HTTP/JSON Set-Response	["set","time","sync","0","3600"]
CMD-UART Get-Request	get time utc\x0d\x0a
CMD-UART Get-Response	get time utc 0 3600\x0d\x0a
CMD-UART Set-Request	set time utc 3600\x0d\x0a
CMD-UART Set-Response	set time utc 0 3600\x0d\x0a

3 COMMUNICATION WITH THE HOST-CONTROLLER

The command uart is the UART 0. The command uart knows two commands, the request and the send command.

3.1 COMMAND-UART

3.1.1 Additional user rights

With the command a application can decide to send the current user rights via a command uart command.
The user rights + '#' in front of the rights, will be added into the data string, in front of the data divided by a space from the original data.
Example:
Original data (binary)

```
54 65 73 74
```

Original data (ASCII)

```
Test
```

Added user rights to the data (binary)

```
23 66 66 20 54 65 73 74
```

Added user rights to the data (ASCII)

```
#ff Test
```

3.2 BINARY-UART

3.2.1 Supported baudrates

The following baudrates are supported:
300, 600, 1200, 2400, 4800, 9600, 14400, 19200, 28800, 38400, 56000, 57600, 115200, 128000, 256000, 520000, 780000, 1500000

4 STATUS INFORMATION

4.1 TELEGRAM RETURN CODE

With every GET or SET-Request the telegram is returning a return code. The return code could delivered in two different ways.

- The return code is delivered with a ERROR telegram see Table 2: Return-Code Error-telegram
- The return code is delivered in the the normal telegram behind the variable, see return code description in Table 1: Return-Code Info and Table 3: Return-Code for normal telegram

Return Code	***Description***
0	No error detected
10	Info, that the binary uart got still data left
11	Warning, Data overflow on the binary uart

Table 1: Return-Code Info

Return Code	***Description***
100	Empty telegram
101	Comand not found. Only get or set command is valid
102	Module not found
103	Variable not found
104	Data error
105	Internal Buffer error

Table 2: Return-Code Error-telegram

Return Code	***Description***
201	No permission to get or set this parameter
202	Get Request is not possible for this variable
203	Set Request is not possible for this variable
204	Parameter error
205	Format error
206	Variable not found

207	Communication error with the UART
208	Internal Storage error
301	The parameter length is to long for the variable
302	The application is not yet ready
303	Netcat wrong exclusive lock ID

Table 3: Return-Code for normal telegram

Example for Error-telegram:

HTTP/JSON Get-Request	["test","system","firmware"]
HTTP/JSON Get-Response	["ERROR","101"]
CMD-UART Get-Request	test system firmware\x0d\x0a
CMD-UART Get-Response	ERROR 101 "Command not found"\x0d\x0a

Example for a normal telegram return code.

HTTP/JSON Set-Request	["set","wlan","infra_mode","test"]
HTTP/JSON Set-Response	["set","wlan","infra_mode","204","test"]
CMD-UART Set-Request	set wlan infra_mode test\x0d\x0a
CMD-UART Set-Response	set wlan infra_mode 204 test\x0d\x0a

4.2 WIFI-PARAMETER

4.2.1 WiFi status information

Status - Number	***Description***
0	Not initialized
1	Not connected to a network
2	Establishing a connection
3	IP address is obtained
4	Connected to the network
5	Searching for networks
6	Entered network not found
7	Authentication failed

4.3 MAIL-SERVICE

4.3.1 Mail module status

Status - Number	***Description***
0	Not configured
1	Configuration ok - Ready to send
2	Mail is sending
3	Failed to connect to server
4	Authentification failed

4.4 USER-MANAGMENT

4.4.1 User-Rights

Rights	***Description***
0x80	Rights to change the user configuration update
0x40	Rights to do a firmware update
0x20	Rights to SET/GET data via BINUART
0x10	Rights to SET/GET data via CMDUART HIGH
0x08	Rights to SET/GET data via CMDUART LOW
0x04	Rights to SET parameters with a high priority
0x02	Rights to SET parameters with a low priority
0x01	Rights to GET parameters

The PAN9320 got three demo users

Group	***Description***
Admin (0xFF)	- Rights to change the user configuration update - Rights to do a firmware - Rights to SET/GET data via BINUART - Rights to SET/GET data via CMDUART HIGH - Rights to SET/GET data via CMDUART LOW - Rights to SET parameters with a high priority - Rights to SET parameters with a low priority - Rights to GET parameters with a low priority
Technician (0x3F)	- Rights to SET/GET data via BINUART - Rights to SET/GET data via CMDUART HIGH - Rights to SET/GET data via CMDUART LOW - Rights to SET parameters with a high priority

	- Rights to SET parameters with a low priority - Rights to GET parameters with a high priority - Rights to GET parameters with a low priority
User (0x0B)	- Rights to SET/GET data via CMDUART LOW - Rights to SET parameters with a low priority - Rights to GET parameters with a low priority

4.5 FIRMWARE UPDATE

4.5.1 Firmware Update Status list

Number	***Description***
0	Ready for Update
1	Update active
2	Checking for new firmware
3	Update done
4	Update failed

4.6 HTTP-CLIENT

4.6.1 HTTP-Client Status

Number	***Description***
0	Not configured
1	Ready to send data
2	Data is sending
3	Error

4.7 UART CONFIGURATION

4.7.1 Data bit type

Number	***Description***
8	8 bit data mode

4.7.2 Parity types

Number	***Description***
0	No parity

4.7.3 Stop bit type

Number	***Description***
1	One Stop Bits Select

4.8 GPIO

The PAN9320 module offers some GPIO´s which can be controlled via the command API. The following table shows all available pins.

GPIO-CMD-Index	***HW-Pin Name***	***HW-Pin Description***
4	GPIO4	Digital I/O #4
5	GPIO5	Digital I/O #5
6	GPIO6	Digital I/O #6
7	GPIO7	Digital I/O #7
28	GPIO28	Digital I/O #28
44	GPIO44	Digital I/O #44
45	GPIO45	Digital I/O #45

Tabelle 4: gpio pin table

5 CHANGE HISTORY

Vers.	***Date***	***by***	***Change description***
1.01	22/09/14	bme	Base version
1.02	22.01.15	osu	- Revised entire chapter "2 ".
1.03	26.01.15	Bme,ch	- Adapted the commands
1.04	19.02.15	Bme,ch	- Added netcat
1.05	23.02.15	Ch	- Added range of the Parameter 1 "baudrate" in netcat uart_cfg
1.06	10.3.15	Ch	-Changed layout
1.07	20.3.15	Ch	-Added Modul Information
1.1	13.7.15	Ch	-minor changes
1.2	27.11.15	CH	-corrected ranges,detail

6 GENERAL INFORMATION

© Panasonic Electronic Devices Europe GmbH 2010.

All rights reserved.

Panasonic does not warranty and accepts no liability for the information contained herein. The information contained in this document is subject to change without notice. Modules containing "ES" in the series number are Engineering Samples -- i.e. PANxxxxES. This means, the design of this product is not yet concluded. Engineering Samples may be partially or fully functional, and there may be differences to be published Data Sheet.

Engineering Samples are not qualified and are not to be used for reliability testing or series production.

Disclaimer:

Customer acknowledges that samples may deviate from the Data Sheet and may bear defects due to their status of development and the lack of qualification mentioned above.

Panasonic rejects any liability or product warranty for Engineering Samples. In particular, Panasonic disclaims liability for damages caused by

the use of the Engineering Sample other than for Evaluation Purposes, particularly the installation or integration in another product to be sold by Customer,

deviation or lapse in function of Engineering Sample,

improper use of Engineering Samples.

Panasonic disclaims any liability for consequential and incidental damages.

In case of any questions, please contact your local sales partner or the related product manager.