DIP SWITCH	3x10100 4x10100 I:10099	65,0x0041 B:00 41			UINT16 R/O	
Returns the current setting of the Dip switches. For ULTRA SLIM IOs The current value of the DIP switches: Bit 0: DIP Switch 1 (=0:OFF, =1:ON) Bit 1: DIP Switch 2 (=0:OFF, =1:ON) Bit 2: DIP Switch 3 (=0:OFF, =1:ON) Bit 3: DIP Switch 4 (=0:OFF, =1:ON) For BIG IOs: The current value of the DIP switches: Bit 0: DIP Switch 1 (=0:OFF, =1:ON) Bit 1: DIP Switch 2 (=0:OFF, =1:ON) Bit 2: DIP Switch 3 (=0:OFF, =1:ON) Bit 3: DIP Switch 4 (=0:OFF, =1:ON) Bit 4: DIP Switch 5 (=0:OFF, =1:ON) Bit 5: DIP Switch 6 (=0:OFF, =1:ON) Bit 6: DIP Switch 7 (=0:OFF, =1:ON) Bit 7: DIP Switch 8 (=0:OFF, =1:ON)						
PRODUCT DATA						
HW_GROUP	3x65201 4x65201 I:65200	16384,0x4000 B:40 00			UINT16 R/O	
This is the group of hardware of the current product						
SW_GROUP	3x65202 4x65202 I:65201	32783,0x800F B:80 0F			UINT16 R/O	
This is the group of software of the current product						
SW_VERSION	3x65203 4x65203 I:65202	4608,0x1200 B:12 00			UINT16 R/O	
		SW VERSION:1.2.0				
This is the current software version of the firmware						
SW_AUTHOR	3x65204 4x65204 I:65203	18771,0x4953 B:49 53			UINT16 R/O	
This is the current software author of the firmware						
MODBUS SETTINGS						
UNIT_ID	3x65222 4x65222 I:65221	1,0x0001 B:00 01			UINT16 R/O	
		UNIT ID:1				
If the host reads this register, the current defined unit ID is returned.						

FLASH UNIT_ID	3x65223 4x65223 I:65222	15,0x000F B:00 0F		27	UINT16 R/W	NO
		UNIT ID:15				
If the host reads this register, the current defined unit ID from the FLASH is returned. This UnitID is used if DIP switch for UnitID is set to 15						
HINT:This settings will be active after you repower or reset your device !!						
BAUD_RATE	3x65224 4x65224 I:65223	115200,0x0001C200 B:00 01 C2 00	57600	57600	UINT32 R/W	NO
		115200Bd		ENTER BAUD RATE		
This is the current configured baud rate in the FLASH For ULTRA SLIM IOs RESI-xxx-SIO: This baudrate is only used, if DIP switch mode DIP1=ON+DIP2=ON (BR) (default is 57600bd) For BIG IOs RESI-xxx-SIO: This baudrate is only used, if DIP switch mode DIP7=ON (PARAMETER) (default is 57600bd)						
Valid baud rates are: 300bd, 600bd, 900bd, 1200bd, 2400bd, 4800bd, 9600bd, 19200bd, 38400bd, 57600bd, 115200bd, 128000bd 230400bd, 250000bd, 256000bd						
HINT:This settings will be active after you repower or reset your device !!						
PARITY	3x65226 4x65226 I:65225	0,0x0000 B:00 00		1:EVEN PARITY	UINT16 R/W	NO
		NO PARITY		SELECT PARITY		
If the register is read out, the currently set parity of the serial interface is returned. Writing a value to this register will change the new parity in FLASH. This will only take effect after a restart of the module. This can be triggered by writing to the RESET SYSTEM register. Parity values are 0: no parity 1: even parity 2: odd parity						
STOP BITS	3x65227 4x65227 I:65226	1,0x0001 B:00 01		2:TWO STOPBITS	UINT16 R/W	NO
		ONE STOPBIT		SELECT STOPBITS		
If the register is read out, the currently set number of stop bits of the serial interface is returned. Writing a value to this register will change the new number of stop bits in the FLASH. This will only take effect after a restart of the module. This can be triggered by writing to the RESET SYSTEM register. Values for stop bits are 1: one stop bit 2: two stop bits						
MODBUS TIMING	3x65228 4x65228 I:65227	0,0x0000 B:00 00		10	UINT16 R/W	NO
		Actual timing:0ms				
If the host reads this register, the current defined timing for MODBUS telegrams is returned. This timing is a time in ms which extends the standard 1.5 character timeout between two consecutive bytes on the serial line. If you write a new value to this register, the new settings are stored into the internal FLASH. Reboot the device to activate the new settings.						

MODBUS WATCHDOG TIME	3x65229 4x65229 I:65228	0,0x0000 B:00 00		50	UINT16 R/W	YES
		Actual watchdog time in 1/100s:0 -> 0,0s				
Writing a value onto this register defines a new time for the internal communcation watchdog timer. The value is a timespan in 1/100s. =0: The communcation watchdog is disabled =1..65535: Communcation watchdog will be trigegred after x 1/100s pause on communcation line In case of an communcation watchdog, the module sets all outputs to the states defined in the configuration output registers Reading this registe will return the current stored time from the internal FLASH						
CPU DATA						
SERIAL1	3x65521 4x65521 I:65520	49,0x0031 B:00 31			UINT16 R/O	
Serial number of module as 96 bit unsigned integer number						
SERIAL2	3x65522 4x65522 I:65521	62,0x003E B:00 3E			UINT16 R/O	
SERIAL3	3x65523 4x65523 I:65522	21258,0x530A B:53 0A			UINT16 R/O	
SERIAL4	3x65524 4x65524 I:65523	18005,0x4655 B:46 55			UINT16 R/O	
SERIAL5	3x65525 4x65525 I:65524	12343,0x3037 B:30 37			UINT16 R/O	
SERIAL6	3x65526 4x65526 I:65525	8248,0x2038 B:20 38			UINT16 R/O	
		SERIAL:31003E000A53554637303820				
Serial number of module as 96 bit unsigned integer number						
CPU TEMPERATURE	3x65527 4x65527 I:65526	3865,0x0F19 B:0F 19			UINT16 R/O	
		Current internal temperature of CPU:38,7°C				
Current internal temperature of CPU in ° Celsius multiplied by 10.						
CPU VOLTAGE	3x65528 4x65528 I:65527	336,0x0150 B:01 50			UINT16 R/O	
		Current supply voltage of CPU:3,36V				
Current internal supply voltage of CPU in Volt multiplied by 1000.						

CPU BATTERY	3x65529 4x65529 I:65528	317,0x013D B:01 3D			UINT16 R/O	
		Current battery voltage of CPU:3,17V				
Current internal backup battery voltage of CPU in Volt multiplied by 1000.						
CONVERTER STATUS						
CONVERTER STATUS	3x65534 4x65534 I:65533	0,0x0000 B:00 00			UINT16 R/O	
Current status of the converter						
FACTORY RESET	3x65535 4x65535 I:65534	0,0x0000 B:00 00		1:PERFORM FACTORY RESET	UINT16 R/W	NO
Performs a factory reset of all internal saved parameters						
SOFTWARE RESET						
RESET	1x65536 2x65536 I:65535	0,0x00 B:00		N/A:NO CHANGE	BIT R/W	NO
Performs a software reset, whenever 1 is written to this register. If the host writes to this register 1, the module executes a soft reset (reboot).						
RESET	3x65536 4x65535 I:65535	0,0x0000 B:00 00		N/A:NO CHANGE	UINT16 R/W	NO
Performs a software reset, whenever 1 is written to this register. If the host writes to this register 1, the module executes a soft reset (reboot).						

HEART BEAT	ASCII READ COMMAND	#HB<CR> Result: #HB<CR>	ASCII	
	TX	#HB<CR>		
	RX	#1,HB<CR>		
Sends an Heartbeat to test the communcation				
GET VERSION	ASCII READ COMMAND	#VERSION<CR> Result: #VERSION:<VersionHi>,<VersionMed>,<VersionLo><CR>	ASCII	
	TX	#VERSION<CR>		
	RX	#1,VERSION:1.2.00<CR>		
		Current SW version:1.2.00		
Returns the version number of the module VersionHi: Version number high (1..255) VersionMed: Version number medium (1..255) VersionLo: Version number low (1..255)				
GET TYPE	ASCII READ COMMAND	#TYPE<CR> Result: #TYPE:<Type><CR>	ASCII	
	TX	#TYPE<CR>		
	RX	#1,TYPE:RESI-16RO-SIO<CR>		
		Current module type:RESI-16RO-SIO		
Returns the current module type				
GET FEATURES	ASCII READ COMMAND	#FTRS<CR> Result: #FTRS:<Type><CR>	ASCII	
	TX	#FTRS<CR>		
	RX	#1,FTRS:RESI-16RO-SIO,16RO<CR>		
		Current module type:RESI-16RO-SIO		
Returns the current module features				
GET OWNER	ASCII READ COMMAND	#OWNER<CR> Result: #OWNER:<Owner><CR>	ASCII	
	TX	#OWNER<CR>		
	RX	#1,OWNER:RESI<CR>		
		Current owner:RESI		
Returns the current owner of the module				
GET CREATOR	ASCII READ COMMAND	#CREATOR<CR> Result: #CREATOR:<Creator><CR>	ASCII	
	TX	#CREATOR<CR>		
	RX	#1,CREATOR:DI HC SIGL,MSC<CR>		
		Current creator:DI HC SIGL,MSC		
Returns the current creator of the module				

GET COPYRIGHT	ASCII READ COMMAND	#COPYRIGHT<CR> Result: #COPYRIGHT:<Copyright><CR>	ASCII	
	TX	#COPYRIGHT<CR>		
	RX	#1,COPYRIGHT:2015-23 BY RESI AND DI HC SIGL,MSC WWW.RESI.CC<CR>		
		Current copyright:2015-23 BY RESI AND DI HC SIGL,MSC WWW.RESI.CC		
Returns the current copyright of the module				
GET SERIAL NUMBER	ASCII READ COMMAND	#SN<CR> Result: #SN:<Serial><CR>	ASCII	
	TX	#SN<CR>		
	RX	#1,SN:31003E000A53554637303820<CR>		
		Current serial number:31003E000A53554637303820		
Returns the current serial number of the module				
GET INTERNAL STATUS	ASCII READ COMMAND	#INTSTAT<CR> Result: #INTSTAT:<Status><CR>	ASCII	
	TX	#INTSTAT<CR>		
	RX	#1,INTSTAT:I2C1:0,I2C2:0,FRAM:28<CR>		
Returns the device specific internal status				
GET DIP SWITCH	ASCII READ COMMAND	#GDIP<CR> Result: #GDIP:<DIPSwitchDec>,<DIPSwitchHex><CR>	ASCII	
	TX	#GDIP<CR>		
	RX	#1,GDIP:65,0x41<CR>		
		Current DIP SWITCH settings:0100.0001		
Returns the current setting of the Dip switches as decimal number and as hexadecimal number. DIPSwitchDec DIPSwitchHex The current value of the DIP switches: Bit 0: DIP Switch 1 (=0:OFF, =1:ON) Bit 1: DIP Switch 2 (=0:OFF, =1:ON) Bit 2: DIP Switch 3 (=0:OFF, =1:ON) Bit 3: DIP Switch 4 (=0:OFF, =1:ON) Bit 4: DIP Switch 5, if available (=0:OFF, =1:ON) Bit 5: DIP Switch 6, if available (=0:OFF, =1:ON) Bit 6: DIP Switch 7, if available (=0:OFF, =1:ON) Bit 7: DIP Switch 8, if available (=0:OFF, =1:ON)				
ASCII COMMANDS				
SET MODBUS ADDRESS	ASCII WRITE COMMAND	#SMBADR:<UNITID><CR> Result: #OK<CR>	ASCII	NO
	UNITID	123		
	TX	#SMBADR:123<CR>		
	RX	N/A		

Redefines the unit ID of the module. This change will affect the MODBUS/RTU communication immediately. As a Unit IO you can use the values 0dec to 255dec.

HINT: The new settings are activated after a system reboot or power off on cycle!

SET MODBUS BAUDRATE	ASCII WRITE COMMAND	#SMBBAUD:<BAUD><CR> Result: #OK<CR>	ASCII	NO
	BAUD	128000:128000BD		
	TX	#SMBBAUD:128000<CR>		
	RX	N/A		

Sets a new baud rate in the FLASH
For ULTRA SLIM IOs RESI-xxx-SIO: This baudrate is only used, if DIP switch mode DIP1=ON+DIP2=ON (BR) (default is 57600bd)
For BIG IOs RESI-xxx-SIO: This baudrate is only used, if DIP switch mode DIP7=ON (PARAMETER) (default is 57600bd)
The following baudrates are allowed:
300bd, 600bd, 900bd, 1200bd, 2400bd, 4800bd,
9600bd, 19200bd, 38400bd, 57600bd, 115200bd, 128000bd
230400bd, 250000bd, 256000bd

HINT: The new setup parameters will be active after a restart of the module.

SET MODBUS PARITY	ASCII WRITE COMMAND	#SMBPAR:<PARITY><CR> Result: #OK<CR>	ASCII	NO
	PARITY	ODD:ODD PARITY		
	TX	#SMBPAR:ODD<CR>		
	RX	N/A		

Sets a new parity for the serial interface.
MBParity:
NONE: no parity
EVEN: even parity
ODD: odd parity

HINT: The new setup parameters will be active after a restart of the module.

SET MODBUS STOPS	ASCII WRITE COMMAND	#SMBSTOP:<STOPBIT><CR> Result: #OK<CR>	ASCII	NO
	STOPBIT	TWO:TWO STOPBITS		
	TX	#SMBSTOP:TWO<CR>		
	RX	N/A		

Sets a new amount of stop bits for the serial interface.
MBStops
ONE: one stop bit
TWO: two stop bits

HINT: The new setup parameters will be active after a restart of the module.

SET MODBUS PARAMS	ASCII WRITE COMMAND	#SMBPARAMS:<UNITID>,<BAUD>,<PARITY>,<STOPBIT><CR> Result: #OK<CR>	ASCII	NO
	UNITID	3		
	BAUD	115200:115200BD		
	PARITY	EVEN:EVEN PARITY		
	STOPBIT	TWO:TWO STOPBITS		
	TX	#SMBPARAMS:3,115200,EVEN,TWO<CR>		
	RX	N/A		
Sets all parameters for serrial interface				
GET MODBUS ADDRESS	ASCII READ COMMAND	#GMBADR<CR> Result: #GMBADR:<MBUnitDec>,<MBFLASHDec>,<MBUnitHex>,<MBFLASHHex><CR>	ASCII	
	TX	#GMBADR<CR>		
	RX	#1,GMBADR:1,15,0x1,0xF<CR>		
		Current MODBUS unit ID:1,15,0x1,0xF		
Shows the current used MODBUS/RTU or ASCII unit address and shows also the stored unit address in the FLASH memory, which is only used if the DIP switch for the bus address is set to 0. MBUnitDec,MBUnitHex The current used MODBUS/RTU unit or ASCII address for communication MBFLASHDec,MBFLASHHex The internal stored MODBUS/RTU unit address or ASCII address from the FLASH memory, if the DIP switch DIP3 is OFF.				
GET MODBUS BAUDRATE	ASCII READ COMMAND	#GMBBAUD<CR> Result: #GMBBAUD:<BaudRate><CR>	ASCII	
	TX	#GMBBAUD<CR>		
	RX	#1,GMBBAUD:115200,0x1C200<CR>		
		Current baudrate:115200,0x1C200		
This is the current configured baud rate in the FLASH For ULTRA SLIM IOs RESI-xxx-SIO: This baudrate is only used, if DIP switch mode DIP1=ON+DIP2=ON (BR) (default is 57600bd) For BIG IOs RESI-xxx-SIO: This baudrate is only used, if DIP switch mode DIP7=ON (PARAMETER) (default is 57600bd) The following baudrates are allowed: 300bd, 600bd, 900bd, 1200bd, 2400bd, 4800bd, 9600bd, 19200bd, 38400bd, 57600bd, 115200bd, 128000bd 230400bd, 250000bd, 256000bd				
GET MODBUS PARITY	ASCII READ COMMAND	#GMBPAR<CR> Result: #GMBPAR:<MBParity><CR>	ASCII	
	TX	#GMBPAR<CR>		
	RX	#1,GMBPAR:NONE<CR>		
		Current parity:NONE		
Shows the current configured parity of the serial interface. MBParity NONE: no parity EVEN: even parity ODD: odd parity				

GET MODBUS STOP	ASCII READ COMMAND	#GMBSTOP<CR> Result: #GMBSTOP:<MBStop><CR>	ASCII	
	TX	#GMBSTOP<CR>		
	RX	#1,GMBPAR:ONE<CR>		
		Current stopbit(s):ONE		
Shows the current configured parity of the serial interface. MBParity NONE: no parity EVEN: even parity ODD: odd parity				
GET MODBUS PARAMS	ASCII READ COMMAND	#GMBPARAMS<CR> Result: #GMBPARAMS:<MBUnitDec>,<MBFLASHDec>,<MBUnitHex>,<MBFLASHHex>, <MBBaudrateDec>,<MBBaudrateHex>,<MBParity>,<MBStops><CR>	ASCII	
	TX	#GMBPARAMS<CR>		
	RX	#1,GMBADR:1,0x1,15,0xF,115200,0x1C200,NONE,ONE<CR>		
		Current MODBUS unit ID used:1		
		Current MODBUS unit ID in FLASH:15		
		Current baudrate in FLASH:115200		
		Current parity in FLASH:NONE		
		Current stopbit(s) in FLASH:ONE		
Returns the complete settings for serial interface				
ASCII COMMANDS				
RESET	ASCII WRITE COMMAND	#RST<CR> Result: #OK<CR>	ASCII	NO
	TX	#RST<CR>		
	RX	N/A		
Executes a software reset (Reboot) of the module.				
FACTORY RESET	ASCII WRITE COMMAND	#FRST<CR> Result: #OK<CR>	ASCII	NO
	TX	#FRST<CR>		
	RX	N/A		
Performs a factory reset of all internal saved parameters				
SET MODBUS WATCHDOG TIMER	ASCII WRITE COMMAND	#SMBWATCHDOG:<WDTIME><CR> Result: #OK<CR>	ASCII	YES
	WDTIME	100		
	TX	#SMBWATCHDOG:100<CR>		
	RX	#1,OK<CR>		

Enables or disables the WATCHDOG Timer for the IO module. WDTIME: 1..65535: Time for Watchdog in 1/100s =0: Watchdog is deactivated HINT: The Watchdog is internally handled every 100ms. If the IO module receives no valid frame within this time period, the outputs are set to predefined values!				
GET MODBUS WATCHDOG TIMER	ASCII READ COMMAND	#GMBWATCHDOG<CR> Result: #GMBWATCHDOG:<WDTIME><CR>	ASCII	
	TX	#GMBWATCHDOG<CR>		
	RX	#1,GMBWATCHDOG:100,0x64<CR>		
		Current watchdog time:100 -> 10,0s		
Shows the actual configured time for the telegram watchdog function of the IO module. WDTIME: 1..65535: Time for Watchdog in 1/100s =0: Watchdog is deactivated HINT: The Watchdog is internally handled every 100ms. If the IO module receives no valid frame within this time period, the outputs are set to predefined values!				
CPU PARAMETERS				
GET CPU VOLTAGE	ASCII READ COMMAND	#GCPUTEMP<CR> Result: #GCPUTEMP:<CPUTemp><CR>	ASCII	
	TX	#GCPUTEMP<CR>		
	RX	#1,GCPUTEMP:38.7842<CR>		
		Current internal temperature of CPU:38.7842°C		
Current internal temperature of CPU in ° Celsius multiplied by 10.				
GET CPU VOLTAGE	ASCII READ COMMAND	#GCPUVOLT<CR> Result: #GCPUVOLT:<CPUVoltage><CR>	ASCII	
	TX	#GCPUVOLT<CR>		
	RX	#1,GCPUVOLT:3.3632<CR>		
		Current supply voltage of CPU:3.3632V		
Current internal supply voltage of CPU in Volt multiplied by 1000.				
GET CPU BATTERY	ASCII READ COMMAND	#GCPUBATT<CR> Result: #GCPUBATT:<CPUBatteryVoltage><CR>	ASCII	
	TX	#GCPUBATT<CR>		
	RX	#1,GCPUBATT:3.1793<CR>		
		Current backup battery voltage of CPU:3.1793V		
Current internal backup battery voltage of CPU in Volt multiplied by 1000.				

DI1	1x00001 2x00001 I:0	0,0x00 B:00			BIT R/O	
		Current state of DI1:0=OFF				
Current state of the digital input DIx =0:DI is OFF, =1:DI is ON						
DI2	1x00002 2x00002 I:1	0,0x00 B:00			BIT R/O	
		Current state of DI2:0=OFF				
DI3	1x00003 2x00003 I:2	0,0x00 B:00			BIT R/O	
		Current state of DI3:0=OFF				
DI4	1x00004 2x00004 I:3	0,0x00 B:00			BIT R/O	
		Current state of DI4:0=OFF				
DI5	1x00005 2x00005 I:4	0,0x00 B:00			BIT R/O	
		Current state of DI5:0=OFF				
DI6	1x00006 2x00006 I:5	0,0x00 B:00			BIT R/O	
		Current state of DI6:0=OFF				
DI7	1x00007 2x00007 I:6	0,0x00 B:00			BIT R/O	
		Current state of DI7:0=OFF				
DI8	1x00008 2x00008 I:7	0,0x00 B:00			BIT R/O	
		Current state of DI8:0=OFF				
DI9	1x00009 2x00009 I:8	0,0x00 B:00			BIT R/O	
		Current state of DI9:0=OFF				
DI10	1x00010 2x00010 I:9	0,0x00 B:00			BIT R/O	
		Current state of DI10:0=OFF				
DI11	1x00011 2x00011 I:10	0,0x00 B:00			BIT R/O	

		Current state of DI11:0=OFF				
DI12	1x00012 2x00012 I:11	0,0x00 B:00			BIT R/O	
		Current state of DI12:0=OFF				
DIGITAL INPUTS: RESET						
RESET COUNTERS	1x10000 2x10000 I:9999	0,0x00 B:00		1:PERFORM RESET	BIT R/W	NO
If this register is written to 1, all internal edge counters and event counters are set to 0. 0 is always returned when reading.						
STATUS DIGITAL INPUTS						
DI1	3x00001 4x00001 I:0	0,0x0000 B:00 00			UINT16 R/O	
		Current state of DI1:0=OFF				
Current state of the digital input DIx =0:DI is OFF, =1:DI is ON						
DI2	3x00002 4x00002 I:1	0,0x0000 B:00 00			UINT16 R/O	
		Current state of DI2:0=OFF				
DI3	3x00003 4x00003 I:2	0,0x0000 B:00 00			UINT16 R/O	
		Current state of DI3:0=OFF				
DI4	3x00004 4x00004 I:3	0,0x0000 B:00 00			UINT16 R/O	
		Current state of DI4:0=OFF				
DI5	3x00005 4x00005 I:4	0,0x0000 B:00 00			UINT16 R/O	
		Current state of DI5:0=OFF				
DI6	3x00006 4x00006 I:5	0,0x0000 B:00 00			UINT16 R/O	
		Current state of DI6:0=OFF				
DI7	3x00007 4x00007 I:6	0,0x0000 B:00 00			UINT16 R/O	
		Current state of DI7:0=OFF				
DI8	3x00008 4x00008 I:7	0,0x0000 B:00 00			UINT16 R/O	
		Current state of DI8:0=OFF				

DI9	3x00009 4x00009 I:8	0,0x0000 B:00 00			UINT16 R/O	
		Current state of DI9:0=OFF				
DI10	3x00010 4x00010 I:9	0,0x0000 B:00 00			UINT16 R/O	
		Current state of DI10:0=OFF				
DI11	3x00011 4x00011 I:10	0,0x0000 B:00 00			UINT16 R/O	
		Current state of DI11:0=OFF				
DI12	3x00012 4x00012 I:11	0,0x0000 B:00 00			UINT16 R/O	
		Current state of DI12:0=OFF				
DIGITAL INPUTS						
STATUS DI1 A	3x00013 4x00013 I:12	0,0x0000 B:00 00			UINT16 R/O	
		DI:0,CC:0,REC:0,FEC:0				
Status for the digital input DIx Bit 0-4: Lower 5 bits of CHANGE COUNTER Bit 5-9: Lower 5 bits of RISING EDGE COUNTER Bit 10-14: Lower 5 bits of FALLING EDGE COUNTER Bit 15: Current Status of DIx =0: DIx si OFF, =1: DIx is ON						
STATUS DI1 B	3x00014 4x00014 I:13	0,0x0000 B:00 00			UINT16 R/O	
		DI:0,SKE:0,LKSE:0,LKEE:0				
Status for the digital input DIx Bit 0-4: Lower 5 bits of SHORT KEYPRESS EVENTS Bit 5-9: Lower 5 bits of LONG KEYPRESS START EVENTS Bit 10-14: Lower 5 bits of LONG KEYPRESS END EVENTS Bit 15: Current Status of DIx =0: DIx si OFF, =1: DIx is ON						
STATUS DI2 A	3x00015 4x00015 I:14	0,0x0000 B:00 00			UINT16 R/O	
		DI:0,CC:0,REC:0,FEC:0				
STATUS DI2 B	3x00016 4x00016 I:15	0,0x0000 B:00 00			UINT16 R/O	
		DI:0,SKE:0,LKSE:0,LKEE:0				
STATUS DI3 A	3x00017 4x00017 I:16	0,0x0000 B:00 00			UINT16 R/O	

		DI:0,CC:0,REC:0,FEC:0			
STATUS DI3 B	3x00018 4x00018 I:17	0,0x0000 B:00 00		UINT16 R/O	
		DI:0,SKE:0,LKSE:0,LKEE:0			
STATUS DI4 A	3x00019 4x00019 I:18	0,0x0000 B:00 00		UINT16 R/O	
		DI:0,CC:0,REC:0,FEC:0			
STATUS DI4 B	3x00020 4x00020 I:19	0,0x0000 B:00 00		UINT16 R/O	
		DI:0,SKE:0,LKSE:0,LKEE:0			
STATUS DI5 A	3x00021 4x00021 I:20	0,0x0000 B:00 00		UINT16 R/O	
		DI:0,CC:0,REC:0,FEC:0			
STATUS DI5 B	3x00022 4x00022 I:21	0,0x0000 B:00 00		UINT16 R/O	
		DI:0,SKE:0,LKSE:0,LKEE:0			
STATUS DI6 A	3x00023 4x00023 I:22	0,0x0000 B:00 00		UINT16 R/O	
		DI:0,CC:0,REC:0,FEC:0			
STATUS DI6 B	3x00024 4x00024 I:23	0,0x0000 B:00 00		UINT16 R/O	
		DI:0,SKE:0,LKSE:0,LKEE:0			
STATUS DI7 A	3x00025 4x00025 I:24	0,0x0000 B:00 00		UINT16 R/O	
		DI:0,CC:0,REC:0,FEC:0			
STATUS DI7 B	3x00026 4x00026 I:25	0,0x0000 B:00 00		UINT16 R/O	
		DI:0,SKE:0,LKSE:0,LKEE:0			
STATUS DI8 A	3x00027 4x00027 I:26	0,0x0000 B:00 00		UINT16 R/O	
		DI:0,CC:0,REC:0,FEC:0			
STATUS DI8 B	3x00028 4x00028 I:27	0,0x0000 B:00 00		UINT16 R/O	
		DI:0,SKE:0,LKSE:0,LKEE:0			

STATUS DI9 A	3x00029 4x00029 I:28	0,0x0000 B:00 00			UINT16 R/O	
		DI:0,CC:0,REC:0,FEC:0				
STATUS DI9 B	3x00030 4x00030 I:29	0,0x0000 B:00 00			UINT16 R/O	
		DI:0,SKE:0,LKSE:0,LKEE:0				
STATUS DI10 A	3x00031 4x00031 I:30	0,0x0000 B:00 00			UINT16 R/O	
		DI:0,CC:0,REC:0,FEC:0				
STATUS DI10 B	3x00032 4x00032 I:31	0,0x0000 B:00 00			UINT16 R/O	
		DI:0,SKE:0,LKSE:0,LKEE:0				
STATUS DI11 A	3x00033 4x00033 I:32	0,0x0000 B:00 00			UINT16 R/O	
		DI:0,CC:0,REC:0,FEC:0				
STATUS DI11 B	3x00034 4x00034 I:33	0,0x0000 B:00 00			UINT16 R/O	
		DI:0,SKE:0,LKSE:0,LKEE:0				
STATUS DI12 A	3x00035 4x00035 I:34	0,0x0000 B:00 00			UINT16 R/O	
		DI:0,CC:0,REC:0,FEC:0				
STATUS DI12 B	3x00036 4x00036 I:35	0,0x0000 B:00 00			UINT16 R/O	
		DI:0,SKE:0,LKSE:0,LKEE:0				
STATUS						
FILTER PATTERN DI1	3x00037 4x00037 I:36	0,0x00000000 B:00 00 00 00			UINT32 R/O	
The internal pattern for corresponding digital input for AC/DC filtering. The internal used state is created out of this internal pattern via oversampling.						
FILTER PATTERN DI2	3x00039 4x00039 I:38	0,0x00000000 B:00 00 00 00			UINT32 R/O	
FILTER PATTERN DI3	3x00041 4x00041 I:40	0,0x00000000 B:00 00 00 00			UINT32 R/O	

FILTER PATTERN DI4	3x00043 4x00043 I:42	0,0x00000000 B:00 00 00 00			UINT32 R/O	
FILTER PATTERN DI5	3x00045 4x00045 I:44	0,0x00000000 B:00 00 00 00			UINT32 R/O	
FILTER PATTERN DI6	3x00047 4x00047 I:46	0,0x00000000 B:00 00 00 00			UINT32 R/O	
FILTER PATTERN DI7	3x00049 4x00049 I:48	0,0x00000000 B:00 00 00 00			UINT32 R/O	
FILTER PATTERN DI8	3x00051 4x00051 I:50	0,0x00000000 B:00 00 00 00			UINT32 R/O	
FILTER PATTERN DI9	3x00053 4x00053 I:52	0,0x00000000 B:00 00 00 00			UINT32 R/O	
FILTER PATTERN DI10	3x00055 4x00055 I:54	0,0x00000000 B:00 00 00 00			UINT32 R/O	
FILTER PATTERN DI11	3x00057 4x00057 I:56	0,0x00000000 B:00 00 00 00			UINT32 R/O	
FILTER PATTERN DI12	3x00059 4x00059 I:58	0,0x00000000 B:00 00 00 00			UINT32 R/O	
GENERAL STATUS OF DIS						
RESET COUNTERS	3x10000 4x10000 I:9999	0,0x0000 B:00 00		1:PERFORM RESET	UINT16 R/W	NO
If this register is written to 1, all internal edge counters and event counters are set to 0. 0 is always returned when reading.						
HAS DIS CHANGED	3x10001 4x10001 I:10000	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
As soon as the module registrates an event on one of the available digital inputs, this global event counter is incremented by 1. Possible events are: Detection of a short keypress Detection of the start of a long keypress Detection of the end of a long keypress						
STATUS OF ALL DIS DI1..DI12	3x10002 4x10002 I:10001	0,0x0000 B:00 00			UINT16 R/O	
		Current state of DI1:0=OFF				
		Current state of DI2:0=OFF				

		Current state of DI3:0=OFF				
		Current state of DI4:0=OFF				
		Current state of DI5:0=OFF				
		Current state of DI6:0=OFF				
		Current state of DI7:0=OFF				
		Current state of DI8:0=OFF				
		Current state of DI9:0=OFF				
		Current state of DI10:0=OFF				
		Current state of DI11:0=OFF				
		Current state of DI12:0=OFF				
Current state of all digital inputs DI1..DI16 Bit 0: =0:DI1 is OFF, =1:DI1 is ON Bit 1: =0:DI2 is OFF, =1:DI2 is ON ... Bit 10: =0:DI11 is OFF, =1:DI11 is ON Bit 11: =0:DI12 is OFF, =1:DI12 is ON						
DIGITAL INPUTS: STATUS FOR DIGITAL INPUT DI1						
RISE DI1	3x30001 4x30001 I:30000	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
Counter for rising edges on the digital input DIx. If the module detects a rising edge on the digital input, this counter is incremented by 1. After power on or a soft reset this counter is set always to 0. With the function RESET COUNTER this counter is also set to 0.						
FALL DI1	3x30002 4x30002 I:30001	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
Counter for falling edges on the digital input DIx. If the module detects a falling edge on the digital input, this counter is incremented by 1. After power on or a soft reset this counter is set always to 0. With the function RESET COUNTER this counter is also set to 0.						
CHANGE DI1	3x30003 4x30003 I:30002	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
Counter for events on the digital input DIx. If the module detects an event on the digital input, this counter is incremented by 1. After power on or a soft reset this counter is set always to 0. With the function RESET COUNTER this counter is also set to 0. The following events are available: Detecion of a short keypress Detection of the start of a long keypress Detection of the end of a long keypress						
SHORT KEYPRESS DI1	3x30004 4x30004 I:30003	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
Counter for short keypress events on the digital input DIx. If the module detects a short keypress on the digital input, this counter is incremented by 1. After power on or a soft reset this counter is set always to 0. With the function RESET COUNTER this counter is also set to 0.						

LONG KEYPRESS START DI1	3x30005 4x30005 I:30004	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
Counter for start events of long keypress actions on the digital input DIx. If the module detects the start of a long keypress action on the digital input, this counter is incremented by 1. After power on or a soft reset this counter is set always to 0. With the function RESET COUNTER this counter is also set to 0.						
LONG KEYPRESS END DI1	3x30006 4x30006 I:30005	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
Counter for end events of long keypress actions on the digital input DIx. If the module detects the end of a long keypress action on the digital input, this counter is incremented by 1. After power on or a soft reset this counter is set always to 0. With the function RESET COUNTER this counter is also set to 0.						
DIGITAL INPUTS: STATUS FOR DIGITAL INPUT DI2						
RISE DI2	3x30011 4x30011 I:30010	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
FALL DI2	3x30012 4x30012 I:30011	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
CHANGE DI2	3x30013 4x30013 I:30012	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
SHORT KEYPRESS DI2	3x30014 4x30014 I:30013	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
LONG KEYPRESS START DI2	3x30015 4x30015 I:30014	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
LONG KEYPRESS END DI2	3x30016 4x30016 I:30015	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
DIGITAL INPUTS: STATUS FOR DIGITAL INPUT DI3						
RISE DI3	3x30021 4x30021 I:30020	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				

FALL DI3	3x30022 4x30022 I:30021	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
CHANGE DI3	3x30023 4x30023 I:30022	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
SHORT KEYPRESS DI3	3x30024 4x30024 I:30023	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
LONG KEYPRESS START DI3	3x30025 4x30025 I:30024	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
LONG KEYPRESS END DI3	3x30026 4x30026 I:30025	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
DIGITAL INPUTS: STATUS FOR DIGITAL INPUT DI4						
RISE DI4	3x30031 4x30031 I:30030	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
FALL DI4	3x30032 4x30032 I:30031	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
CHANGE DI4	3x30033 4x30033 I:30032	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
SHORT KEYPRESS DI4	3x30034 4x30034 I:30033	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
LONG KEYPRESS START DI4	3x30035 4x30035 I:30034	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
LONG KEYPRESS END DI4	3x30036 4x30036 I:30035	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				

DIGITAL INPUTS: STATUS FOR DIGITAL INPUT DI5						
RISE DI5	3x30041 4x30041 I:30040	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
FALL DI5	3x30042 4x30042 I:30041	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
CHANGE DI5	3x30043 4x30043 I:30042	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
SHORT KEYPRESS DI5	3x30044 4x30044 I:30043	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
LONG KEYPRESS START DI5	3x30045 4x30045 I:30044	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
LONG KEYPRESS END DI5	3x30046 4x30046 I:30045	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
DIGITAL INPUTS: STATUS FOR DIGITAL INPUT DI6						
RISE DI6	3x30051 4x30051 I:30050	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
FALL DI6	3x30052 4x30052 I:30051	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
CHANGE DI6	3x30053 4x30053 I:30052	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
SHORT KEYPRESS DI6	3x30054 4x30054 I:30053	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
LONG KEYPRESS START DI6	3x30055 4x30055 I:30054	0,0x0000 B:00 00			UINT16 R/O	

		0 event(s)				
LONG KEYPRESS END DI6	3x30056 4x30056 I:30055	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
DIGITAL INPUTS: STATUS FOR DIGITAL INPUT DI7						
RISE DI7	3x30061 4x30061 I:30060	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
FALL DI7	3x30062 4x30062 I:30061	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
CHANGE DI7	3x30063 4x30063 I:30062	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
SHORT KEYPRESS DI7	3x30064 4x30064 I:30063	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
LONG KEYPRESS START DI7	3x30065 4x30065 I:30064	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
LONG KEYPRESS END DI7	3x30066 4x30066 I:30065	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
DIGITAL INPUTS: STATUS FOR DIGITAL INPUT DI8						
RISE DI8	3x30071 4x30071 I:30070	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
FALL DI8	3x30072 4x30072 I:30071	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
CHANGE DI8	3x30073 4x30073 I:30072	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				

SHORT KEYPRESS DI8	3x30074 4x30074 I:30073	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
LONG KEYPRESS START DI8	3x30075 4x30075 I:30074	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
LONG KEYPRESS END DI8	3x30076 4x30076 I:30075	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
DIGITAL INPUTS: STATUS FOR DIGITAL INPUT DI9						
RISE DI9	3x30081 4x30081 I:30080	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
FALL DI9	3x30082 4x30082 I:30081	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
CHANGE DI9	3x30083 4x30083 I:30082	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
SHORT KEYPRESS DI9	3x30084 4x30084 I:30083	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
LONG KEYPRESS START DI9	3x30085 4x30085 I:30084	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
LONG KEYPRESS END DI9	3x30086 4x30086 I:30085	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
DIGITAL INPUTS: STATUS FOR DIGITAL INPUT DI10						
RISE DI10	3x30091 4x30091 I:30090	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
FALL DI10	3x30092 4x30092 I:30091	0,0x0000 B:00 00			UINT16 R/O	

		0 event(s)			
CHANGE DI10	3x30093 4x30093 I:30092	0,0x0000 B:00 00		UINT16 R/O	
		0 event(s)			
SHORT KEYPRESS DI10	3x30094 4x30094 I:30093	0,0x0000 B:00 00		UINT16 R/O	
		0 event(s)			
LONG KEYPRESS START DI10	3x30095 4x30095 I:30094	0,0x0000 B:00 00		UINT16 R/O	
		0 event(s)			
LONG KEYPRESS END DI10	3x30096 4x30096 I:30095	0,0x0000 B:00 00		UINT16 R/O	
		0 event(s)			
DIGITAL INPUTS: STATUS FOR DIGITAL INPUT DI11					
RISE DI11	3x30101 4x30101 I:30100	0,0x0000 B:00 00		UINT16 R/O	
		0 event(s)			
FALL DI11	3x30102 4x30102 I:30101	0,0x0000 B:00 00		UINT16 R/O	
		0 event(s)			
CHANGE DI11	3x30103 4x30103 I:30102	0,0x0000 B:00 00		UINT16 R/O	
		0 event(s)			
SHORT KEYPRESS DI11	3x30104 4x30104 I:30103	0,0x0000 B:00 00		UINT16 R/O	
		0 event(s)			
LONG KEYPRESS START DI11	3x30105 4x30105 I:30104	0,0x0000 B:00 00		UINT16 R/O	
		0 event(s)			
LONG KEYPRESS END DI11	3x30106 4x30106 I:30105	0,0x0000 B:00 00		UINT16 R/O	
		0 event(s)			
DIGITAL INPUTS: STATUS FOR DIGITAL INPUT DI12					

RISE DI12	3x30111 4x30111 I:30110	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
FALL DI12	3x30112 4x30112 I:30111	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
CHANGE DI12	3x30113 4x30113 I:30112	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
SHORT KEYPRESS DI12	3x30114 4x30114 I:30113	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
LONG KEYPRESS START DI12	3x30115 4x30115 I:30114	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
LONG KEYPRESS END DI12	3x30116 4x30116 I:30115	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				

DIGITAL INPUTS				
GET DIGITAL INPUTS	ASCII READ COMMAND	#GDIS<CR> Result: #GDIS:<DISDec>,<DISHex><CR>	ASCII	
	TX	#GDIS<CR>		
	RX	#1,GDIS:0,0x0<CR>		
		Current status of digital inputs:0000.0000.0000		
Returns the current state of all 32 digital inputs as decimal number and as hexadecimal number. DISDec, DISHex The current state of all digital inputs: Bit 0: State of DI1 (=0:OFF, =1:ON) Bit 1: State of DI2 (=0:OFF, =1:ON) ... Bit 10: State of DI11 (=0:OFF, =1:ON) Bit 11: State of DI12 (=0:OFF, =1:ON)				
GET DIGITAL INPUT DIx	ASCII READ COMMAND	#GDI<DINR><CR> Result: #GDI<DINR>:<DIxDec>,<DIxHex><CR>	ASCII	
	DINR	12		
	TX	#GDI12<CR>		
	RX	#1,GDI12:0,0x0<CR>		
		Current status of digital input DI12:0=OFF		
Returns the current state of the digital input Dix as decimal number and as hexadecimal number. X stands for the desired digital input between 1 and 12. DIxDec, DIxHex: The current state of the digital input x: =0: Digital input is OFF =1: Digital input is ON				
GET ALL CHANGES	ASCII READ COMMAND	#GAC<CR> Result: #GAC:<ChangesDec>,<ChangesHex><CR>	ASCII	
	TX	#GAC<CR>		
	RX	#1,GAC:0,0x0<CR>		
		Current change counter:0		
Returns the counter for changes on all digital inputs. As soon as the module detects a short keypress or long key press or long key release event, this counter is incremented by 1. If this values has changed sience the last polling request, the host knows, that at least one digital input has changed its state.				
CHANGE ALL DIS	ASCII READ COMMAND	#CADIS<CR> Result: #CADIS:<ChangeDI1Dec>,...,<ChangeDIxDec>,<ChangeDI1Hex>,...,<ChangeDIxHex><CR>	ASCII	
	TX	#CADIS<CR>		
	RX	#1,CADIS:0,0,0,0,0,0,0,0,0,0,0,0,0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0<CR>		
		Current counter for changes on DI1:0		
		Current counter for changes on DI2:0		
		Current counter for changes on DI3:0		
		Current counter for changes on DI4:0		

		Current counter for changes on DI5:0		
		Current counter for changes on DI6:0		
		Current counter for changes on DI7:0		
		Current counter for changes on DI8:0		
		Current counter for changes on DI9:0		
		Current counter for changes on DI10:0		
		Current counter for changes on DI11:0		
		Current counter for changes on DI12:0		
Returns for each digital input the counter for changes. As soon as the module detects a signal change on a digital input, the change counter for the affected digital input is incremented by 1. A signal change can be: Detection of a short keypress Detection of the start of a long keypress Detection of a release of a long keypress				
CHANGE DIx	ASCII READ COMMAND	#CDI<DINR><CR> Result: #CDI<DINR>:<ChangesDec>,<ChangesHex><CR>	ASCII	
	DINR	12		
	TX	#CDI12<CR>		
	RX	#1,CDI12:0,0x0<CR>		
		Current counter for changes on digital input DI12:0		
Returns for digital input <DINR> the counter for signal changes. As soon as the module detects a signal change on a digital input, the change counter for the affected digital input is incremented by 1. A signal change can be: Detection of a short keypress Detection of the start of a long keypress Detection of a release of a long keypress				
SHORT KEY ALL DIS	ASCII READ COMMAND	#SKADIS<CR> Result: #SKADIS:<ShortKeyDI1Dec>,...,<ShortKeyDIxDec>, <ShortKeyDI1Hex>,...,<ShortKeyDIxHex><CR>	ASCII	
	TX	#SKADIS<CR>		
	RX	#1,SKADIS:0,0,0,0,0,0,0,0,0,0,0,0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0<CR>		
		Current counter for short keypress events on DI1:0		
		Current counter for short keypress events on DI2:0		
		Current counter for short keypress events on DI3:0		
		Current counter for short keypress events on DI4:0		
		Current counter for short keypress events on DI5:0		
		Current counter for short keypress events on DI6:0		
		Current counter for short keypress events on DI7:0		
		Current counter for short keypress events on DI8:0		
		Current counter for short keypress events on DI9:0		
		Current counter for short keypress events on DI10:0		
		Current counter for short keypress events on DI11:0		
		Current counter for short keypress events on DI12:0		
Returns for each digital input the counter for short keypress events. As soon as the module detects a short keypress on a digital input, the counter for the affected digital input is incremented by 1.				

SHORT KEY DIx	ASCII READ COMMAND	#SKDI<DINR><CR> Result: #SKDI<DINR>:<ShortKeyDec>,<ShortKeyHex><CR>	ASCII	
	DINR	12		
	TX	#SKDI12<CR>		
	RX	#1,SKDI12:0,0x0<CR>		
		Current counter for short keypress events on digital input DI12:0		
Returns for digital input <DINR> the counter for short keypress events. As soon as the module detects a short keypress on a digital input, the counter for the affected digital input is incremented by 1.				
LONG KEY START ALL DIS	ASCII READ COMMAND	#LKSADIS<CR> Result: #LKSADIS:<LongKeyStartDI1Dec>,...,<LongKeyStartDIxDec>, <LongKeyStartDI1Hex>,...,<LongKeyStartDIxHex><CR>	ASCII	
	TX	#LKSADIS<CR>		
	RX	#1,LKSADIS:0,0,0,0,0,0,0,0,0,0,0,0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0<CR>		
		Current counter for long keypress start events on DI1:0		
		Current counter for long keypress start events on DI2:0		
		Current counter for long keypress start events on DI3:0		
		Current counter for long keypress start events on DI4:0		
		Current counter for long keypress start events on DI5:0		
		Current counter for long keypress start events on DI6:0		
		Current counter for long keypress start events on DI7:0		
		Current counter for long keypress start events on DI8:0		
		Current counter for long keypress start events on DI9:0		
		Current counter for long keypress start events on DI10:0		
		Current counter for long keypress start events on DI11:0		
		Current counter for long keypress start events on DI12:0		
Returns for each digital input the counter for long keypress start events. As soon as the module detects the start of a long keypress on a digital input, the counter for the affected digital input is incremented by 1.				
LONG KEY START DIx	ASCII READ COMMAND	#LKSDI<DINR><CR> Result: #LKSDI<DINR>:<LongKeyStartDec>,<LongKeyStartHex><CR>	ASCII	
	DINR	12		
	TX	#LKSDI12<CR>		
	RX	#1,LKSDI12:0,0x0<CR>		
		Current counter for long keypress start events on digital input DI12:0		
Returns for digital input <DINR> the counter for long keypress start events. As soon as the module detects the start of a long keypress on a digital input, the counter for the affected digital input is incremented by 1.				
LONG KEY END ALL DIS	ASCII READ COMMAND	#LKEADIS<CR> Result: #LKEADIS:<LongKeyEndDI1Dec>,...,<LongKeyEndDIxDec>, <LongKeyEndDI1Hex>,...,<LongKeyEndDIxHex><CR>	ASCII	
	TX	#LKEADIS<CR>		
	RX	#1,LKEADIS:0,0,0,0,0,0,0,0,0,0,0,0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0<CR>		
		Current counter for long keypress end events on DI1:0		
		Current counter for long keypress end events on DI2:0		
		Current counter for long keypress end events on DI3:0		

		Current counter for long keypress end events on DI4:0		
		Current counter for long keypress end events on DI5:0		
		Current counter for long keypress end events on DI6:0		
		Current counter for long keypress end events on DI7:0		
		Current counter for long keypress end events on DI8:0		
		Current counter for long keypress end events on DI9:0		
		Current counter for long keypress end events on DI10:0		
		Current counter for long keypress end events on DI11:0		
		Current counter for long keypress end events on DI12:0		
Returns for each digital input the counter for long keypress end events. As soon as the module detects the end of a long keypress on a digital input, the counter for the affected digital input is incremented by 1.				
LONG KEY END DIx	ASCII READ COMMAND	#LKEDI<DINR><CR> Result: #LKEDI<DINR>:<LongKeyEndDec>,<LongKeyEndHex><CR>	ASCII	
	DINR	12		
	TX	#LKEDI12<CR>		
	RX	#1,LKEDI12:0,0x0<CR>		
		Current counter for long keypress end events on digital input DI12:0		
Returns for digital input <DINR> the counter for long keypress end events. As soon as the module detects the end of a long keypress on a digital input, the counter for the affected digital input is incremented by 1.				
RISE ALL DIS	ASCII READ COMMAND	#RADIS<CR> Result: #RADIS:<RiseDI1Dec>,...,<RiseDIxDec>,<RiseDI1Hex>,...,<RiseDIxHex><CR>	ASCII	
	TX	#RADIS<CR>		
	RX	#1,RADIS:0,0,0,0,0,0,0,0,0,0,0,0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0<CR>		
		Current counter for rising edges on DI1:0		
		Current counter for rising edges on DI2:0		
		Current counter for rising edges on DI3:0		
		Current counter for rising edges on DI4:0		
		Current counter for rising edges on DI5:0		
		Current counter for rising edges on DI6:0		
		Current counter for rising edges on DI7:0		
		Current counter for rising edges on DI8:0		
		Current counter for rising edges on DI9:0		
		Current counter for rising edges on DI10:0		
		Current counter for rising edges on DI11:0		
		Current counter for rising edges on DI12:0		
Returns for each digital input the counter for rising edges. As soon as the module detects a rising edge on a digital input, the rising edge counter for the affected digital input is incremented by 1.				
RISE DIx	ASCII READ COMMAND	#RDI<DINR><CR> Result: #RDI<DINR>:<RiseDec>,<RiseHex><CR>	ASCII	
	DINR	12		
	TX	#RDI12<CR>		
	RX	#1,RDI12:0,0x0<CR>		
		Current counter for rising edges on digital input DI12:0		

Returns for digital input <DINR> the counter for rising edges. As soon as the module detects a rising edge on a digital input, the rising edge counter for the affected digital input is incremented by 1.				
FALL ALL DIS	ASCII READ COMMAND	#FADIS<CR> Result: #FADIS:<FallDI1Dec>,...,<FallDIxDec>,<FallDI1Hex>,...,<FallDIxHex><CR>	ASCII	
	TX	#FADIS<CR>		
	RX	#1,FADIS:0,0,0,0,0,0,0,0,0,0,0,0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0<CR>		
		Current counter for falling edges on DI1:0		
		Current counter for falling edges on DI2:0		
		Current counter for falling edges on DI3:0		
		Current counter for falling edges on DI4:0		
		Current counter for falling edges on DI5:0		
		Current counter for falling edges on DI6:0		
		Current counter for falling edges on DI7:0		
		Current counter for falling edges on DI8:0		
		Current counter for falling edges on DI9:0		
		Current counter for falling edges on DI10:0		
		Current counter for falling edges on DI11:0		
		Current counter for falling edges on DI12:0		
Returns for each digital input the counter for falling edges. As soon as the module detects a falling edge on a digital input, the falling edge counter for the affected digital input is incremented by 1.				
FALL DIx	ASCII READ COMMAND	#FDI<DINR><CR> Result: #FDI<DINR>:<FallDec>,<FallHex><CR>	ASCII	
	DINR	12		
	TX	#FDI12<CR>		
	RX	#1,FDI12:0,0x0<CR>		
		Current counter for falling edges on digital input DI12:0		
Returns for digital input <DINR> the counter for falling edges. As soon as the module detects a falling edge on a digital input, the falling edge counter for the affected digital input is incremented by 1.				
RESET COUNTERS	ASCII WRITE COMMAND	#RC<CR> Result: #OK<CR>	ASCII	YES
	TX	#RC<CR>		
	RX	N/A		
Resets all internal counters for digital inputs and events on this digital inputs to 0.				