Register NAME	MODBUS Register	Register VALUE	DATA TYPE	DO WRITE
ASCII COMMANDS				
HEART BEAT	ASCII READ COMMAND	#HB<CR> Result: #HB<CR>	ASCII	
	TX	#255,HB<CR>		
	RX	#255,HB<CR>		
Sends an Heartbeat to test the communication				
GET VERSION	ASCII READ COMMAND	#VERSION<CR> Result: #VERSION:<VersionHi>,<VersionMed>,<VersionLo><CR>	ASCII	
	TX	#255,VERSION<CR>		
	RX	#255,VERSION:1.1.0<CR>		
		Actual SW version:1.1.0		
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	#255,TYPE<CR>		
	RX	#255,TYPE:RESI-C4-A-16DI12RO4AIOX<CR>		
		Actual module type:RESI-C4-A-16DI12RO4AIOX		
Returns the actual module type				
GET FEATURES	ASCII READ COMMAND	#FTRS<CR> Result: #FTRS:<Type><CR>	ASCII	
	TX	#255,FTRS<CR>		
	RX	#255,FTRS:RESI-C4-A-16DI12RO4AIOX,RS485,DI:16,RO:12,AIOX:4<CR>		
		Actual module type:N/A		
		Number of digital inputs:N/A		
		Type of digital inputs:N/A		
Returns the actual module features				
GET OWNER	ASCII READ COMMAND	#OWNER<CR> Result: #OWNER:<Owner><CR>	ASCII	
	TX	#255,OWNER<CR>		
	RX	#255,OWNER:RESI<CR>		
		Actual owner:RESI		
Returns the actual owner of the module				

GET CREATOR	ASCII READ COMMAND	#CREATOR<CR> Result: #CREATOR:<Creator><CR>	ASCII	
	TX	#255,CREATOR<CR>		
	RX	#255,CREATOR:DI HC SIGL,MSC<CR>		
		Actual creator:DI HC SIGL,MSC		
Returns the actual creator of the module				
GET COPYRIGHT	ASCII READ COMMAND	#COPYRIGHT<CR> Result: #COPYRIGHT:<Copyright><CR>	ASCII	
	TX	#255,COPYRIGHT<CR>		
	RX	#255,COPYRIGHT:2015-24 BY RESI AND DI HC SIGL,MSC WWW.RESI.CC<CR>		
		Actual copyright:2015-24 BY RESI AND DI HC SIGL,MSC WWW.RESI.CC		
Returns the actual copyright of the module				
GET SERIAL NUMBER	ASCII READ COMMAND	#SN<CR> Result: #SN:<Serial><CR>	ASCII	
	TX	#255,SN<CR>		
	RX	#255,SN:310018000D57435535333920<CR>		
		Actual serial number:310018000D57435535333920		
Returns the actual serial number of the module				
SET BOX NAME	ASCII WRITE COMMAND	#SETBOXNAME:<BOXNAME><CR> Result: #OK<CR>	ASCII	YES
	BOXNAME	MYBOX		
	TX	#255,SETBOXNAME:MYBOX<CR>		
	RX	N/A		
Sets a new box name for the controller				
GET BOX NAME	ASCII READ COMMAND	#BOXNAME<CR> Result: #BOXNAME:<BoxName><CR>	ASCII	
	TX	#255,BOXNAME<CR>		
	RX	#255,BOXNAME:NONAME<CR>		
		Actual box name:NONAME		
Returns the actual box name of the module. If no box name is defined, the value NONAME is returned				
GET INTERNAL STATUS	ASCII READ COMMAND	#INTSTAT<CR> Result: #INTSTAT:<Status><CR>	ASCII	
	TX	#255,INTSTAT<CR>		
	RX	#255,INTSTAT:I2C1:0,I2C2:0<CR>		
		Actual internal status:I2C1		
Returns the device specific internal status				

GET DIP SWITCH	ASCII READ COMMAND	#GDIP<CR> Result: #GDIP:<DIPSwitchDec>,<DIPSwitchHex><CR>	ASCII	
	TX	#255,GDIP<CR>		
	RX	#255,GDIP:0,0x0<CR>		
		Actual DIP SWITCH settings:00000000		
Returns the actual 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 (=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)				
SYSTEM COMMANDS				
RESET	ASCII WRITE COMMAND	#RST<CR> Result: #OK<CR>	ASCII	NO
	TX	#255,RST<CR>		
	RX	N/A		
Executes a software reset (Reboot) of the module. Be aware that you will losse all serial interfaces in USB!!!				
FACTORY RESET	ASCII WRITE COMMAND	#FRST<CR> Result: #OK<CR>	ASCII	NO
	TX	#255,FRST<CR>		
	RX	N/A		
Executes a factory reset of the module				
WATCHDOG TIMER	ASCII WRITE COMMAND	#WD:<WDTIME><CR> Result: #OK<CR>	ASCII	NO
	WDTIME	10		
	TX	#255,WD:10<CR>		
	RX	N/A		
Enables or disables the WATCHDOG Timer for the Raspberry Pi module. WDTIME: 1..3600000: Time for Watchdog in Milliseconds (Maximum 60 Minutes) =0: No Watchdog is generated HINT: The Watchdog is internally handled every 10ms, so every value below 10 will reset immediately the Raspberry Pi computer.				
SET IO WATCHDOG TIMER	ASCII WRITE COMMAND	#SIOWATCHDOG:<IOWDTIME><CR> Result: #OK<CR>	ASCII	NO
	IOWDTIME	10		
	TX	#255,SIOWATCHDOG:10<CR>		

	RX	N/A		
Sets a new time for the internal IO WATCHDOG Timer. <IOWDTIME> is a time in 100ms. =0: No IO Watchdog is used HINT: The Watchdog is internally handled every 100ms, if the Timer reaches 0, all internal IOS will be set to a preconfigured state. Every ASCII command or MODBUS request will reset this timer.				
GET IO WATCHDOG TIMER	ASCII READ COMMAND	#GIOWATCHDOG<CR> Result: #GIOWATCHDOG::<IOWDTIME><CR>	ASCII	
	TX	#255,GIOWATCHDOG<CR>		
	RX	#255,GIOWATCHDOG:0,0x0<CR>		
Returns the actual time for the internal IO WATCHDOG Timer. <IOWDTIME> is a time in 100ms. =0: No IO Watchdog is used HINT: The Watchdog is internally handled every 100ms, if the Timer reaches 0, all internal IOS will be set to a preconfigured state. Every ASCII command or MODBUS request will reset this timer.				
CPU PARAMETERS				
GET CPU VOLTAGE	ASCII READ COMMAND	#GCPUTEMP<CR> Result: #GCPUTEMP:<CPUTemp><CR>	ASCII	
	TX	#255,GCPUTEMP<CR>		
	RX	#255,GCPUTEMP:40.6938<CR>		
		Actual internal temperature of CPU:40.6938°C		
Current internal temperature of CPU in ° Celsius.				
GET CPU VOLTAGE	ASCII READ COMMAND	#GCPUVOLT<CR> Result: #GCPUVOLT:<CPUVoltage><CR>	ASCII	
	TX	#255,GCPUVOLT<CR>		
	RX	#255,GCPUVOLT:3.3009<CR>		
		Actual supply voltage of CPU:3.3009V		
Current internal supply voltage of CPU in Volt.				
GET CPU BACKUP	ASCII READ COMMAND	#GCPUBACK<CR> Result: #GCPUBACK:<CPUBackupVoltage><CR>	ASCII	
	TX	#255,GCPUBACK<CR>		
	RX	#255,GCPUBACK:3.0390<CR>		
		Actual backup voltage of CPU for RTC:3.0390V		
Current internal backup voltage of CPU for the RTC in Volt.				

Register NAME	MODBUS Register	Register VALUE	DATA TYPE	DO WRITE
ASCII COMMANDS:REAL TIME CLOCK				
GET REAL TIME CLOCK	ASCII READ COMMAND	#GRTC<CR> Result: #GRTC:YMD,<YEAR>,<MONTH>,<DAY>,HMS,<HOUR>,<MINUTE>,<SECOND>,<WEEKDAY> ,DOK,<DATEOK>,TOK,<TIMEOK><CR>	ASCII	
	TX	#255,GRTC<CR>		
	RX	#255,GRTC:YMD,24,1,1,HMS,0,3,3,MON,DOK,1,TOK,1<CR>		
		Actual date DD.MM.YYYY:1.1.2024		
		Actual time HH.MM.SS (24h):00:03:03		
		Actual Weekday:MON		
		Battery buffered date is ok:YES		
		Battery buffered time is ok:YES		
Shows current RTC time of battery backup RTC on module				
ASCII COMMANDS:REAL TIME CLOCK				
SET REAL TIME CLOCK	ASCII WRITE COMMAND	#SRTC:YMD,<YEAR>,<MONTH>,<DAY>,HMS,<HOUR>,<MINUTE>,<SECOND>, <WEEKDAY><CR> Result: #OK<CR>	ASCII	YES
	YEAR	2026		
	MONTH	04		
	DAY	01		
	HOUR	11		
	MINUTE	51		
	SECOND	16		
	WEEKDAY	WED		
	TX	#255,SRTC:YMD,26,04,01,HMS,11,51,16,WED<CR>		
	RX	N/A		
Executes a software reset (Reboot) of the module.				

Register NAME	MODBUS Register	Register VALUE	DATA TYPE	DO WRITE
ASCII COMMANDS:FRAM				
GET FRAMSIZE	ASCII READ COMMAND	#GFRAMSIZE<CR> Result: #GFRAMSIZE:<FRAMType>,<FRAMSize>,<UsedSizeDEC>,<UsedSizeHEX><CR>	ASCII	
	TX	#255,GFRAMSIZE<CR>		
	RX	#255,GFRAMSIZE:FM25L16B_G,2kB,2,0x2<CR>		
Reads the actual type and size of the used FRAM. The <UsedSize> describes the internal used space in bytes of the FRAM				
ASCII COMMANDS:FRAM				
GET FRAM16	ASCII READ COMMAND	#GFRAM16:<INDEX><CR> Result: #GFRAM16:<INDEXDEC>,<VALUEDEC>,<INDEXHEX>,<VALUEHEX><CR> or #GFRAM16:<INDEXDEC>,ERR,<INDEXHEX>,ERR<CR>	ASCII	
	INDEX	350		
	TX	#255,GFRAM16:350<CR>		
	RX	#255,GFRAM16:350,0,0x15E,0x0<CR>		
		FRAM Index in bytes:350		
		FRAM Value in decimal:0		
Reads the actual UINT16 value (2 bytes) of FRAM memory <INDEX>. <INDEX> is a BYTE index in the FRAM strogae starting with 0.				
GET FRAM32	ASCII READ COMMAND	#GFRAM32:<INDEX><CR> Result: #GFRAM32:<INDEXDEC>,<VALUEDEC>,<INDEXHEX>,<VALUEHEX><CR> or #GFRAM32:<INDEXDEC>,ERR,<INDEXHEX>,ERR<CR>	ASCII	
	INDEX	350		
	TX	#255,GFRAM32:350<CR>		
	RX	#255,GFRAM32:350,0,0x15E,0x0<CR>		
		FRAM Index in bytes:350		
		FRAM Value in decimal:0		
Reads the actual UINT32 value 4 bytes) of FRAM memory <INDEX>. <INDEX> is a BYTE index in the FRAM strogae starting with 0.				
GET FRAMDBL	ASCII READ COMMAND	#GFRAMDBL:<INDEX><CR> Result: #GFRAMDBL:<INDEXDEC>,<VALUEDBL>,<INDEXHEX>,<VALUEDBL><CR> or #GFRAMDBL:<INDEXDEC>,ERR,<INDEXHEX>,ERR<CR>	ASCII	
	INDEX	400		
	TX	#255,GFRAMDBL:400<CR>		
	RX	#255,GFRAMDBL:400,0,0x190,0<CR>		
		FRAM Index in bytes:400		
		FRAM Value in decimal:0		
Reads the actual DOUBLE value 8 bytes) of FRAM memory <INDEX>. <INDEX> is a BYTE index in the FRAM strogae starting with 0.				
ASCII COMMANDS:FRAM				

SET FRAM16	ASCII WRITE COMMAND	#SFRAM16:<INDEX>,<VALUE><CR> Result: #SFRAM16:OK<CR> or #SFRAM16:ERR<CR>	ASCII	YES
	INDEX	350		
	VALUE	1234		
	TX	#255,SFRAM16:350,1234<CR>		
	RX	N/A		
Writes a new UINT16 value (2 byte) into FRAM memory <INDEX>. <INDEX> is a BYTE index in the FRAM strogae starting with 0.				
SET FRAM32	ASCII WRITE COMMAND	#SFRAM32:<INDEX>,<VALUE><CR> Result: #SFRAM32:OK<CR> or #SFRAM32:ERR<CR>	ASCII	YES
	INDEX	350		
	VALUE	123456		
	TX	#255,SFRAM32:350,123456<CR>		
	RX	N/A		
Writes a new UINT32 value (4 byte) into FRAM memory <INDEX>. <INDEX> is a BYTE index in the FRAM strogae starting with 0.				
SET FRAMDBL	ASCII WRITE COMMAND	#SFRAMDBL:<INDEX>,<DOUBLEVALUE><CR> Result: #SFRAMDBL:OK<CR> or #SFRAMDBL:ERR<CR>	ASCII	YES
	INDEX	400		
	DOUBLEVALUE	3,1415926		
	TX	#255,SFRAMDBL:400,3.1415926<CR>		
	RX	N/A		
Writes a new DOUBLE value (8 byte) into FRAM memory <INDEX>. <INDEX> is a BYTE index in the FRAM strogae starting with 0.				

Register NAME	MODBUS Register	Register VALUE	DATA TYPE	DO WRITE
LED STATUS:LED1:GREEN				
GET LED1	ASCII READ COMMAND	#GLED1<CR> Result: #GLED1:<LEDMode>,<LEDStateDec>,<LEDStateHex><CR>	ASCII	
	TX	#255,GLED1<CR>		
	RX	#255,GLED1:OFF,0,0x0<CR>		
		Actual LED state:OFF LED ist currently 0		
Returns the actual state of the LED1:GREEN on the cover of module				
LED COMMANDS:LED1:GREEN				
SET LED1 OFF	ASCII WRITE COMMAND	#SL1OFF<CR> Result: #OK<CR>	ASCII	YES
	TX	#255,SL1OFF<CR>		
	RX	N/A		
Sets the current state of the LED1:GREEN on the cover of module to OFF				
SET LED1 ON	ASCII WRITE COMMAND	#SL1ON<CR> Result: #OK<CR>	ASCII	YES
	TX	#255,SL1ON<CR>		
	RX	N/A		
Sets the current state of the LED1:GREEN on the cover of module to ON				
SET LED1 INVERT	ASCII WRITE COMMAND	#SL1INV<CR> Result: #OK<CR>	ASCII	YES
	TX	#255,SL1INV<CR>		
	RX	N/A		
Inverts the current state of the LED1:GREEN on the cover of module from ON to OFF or from OFF to ON				
SET LED1 PULSE	ASCII WRITE COMMAND	#SL1PULSE:<PULSETIME><CR> Result: #OK<CR>	ASCII	YES
	PULSETIME	1000		
	TX	#255,SL1PULSE:1000<CR>		
	RX	N/A		
Sets the current state of the LED1:GREEN on the cover of module to PULSE and defines the one time pulse duration in Milliseconds between 1 and 60000				
SET LED1 BLINK	ASCII WRITE COMMAND	#SL1BLINK:<BLINKTIME><CR> Result: #OK<CR>	ASCII	YES
	BLINKTIME	1000		
	TX	#255,SL1BLINK:1000<CR>		
	RX	N/A		

Sets the current state of the LED1:GREEN on the cover of module to PULSE and defines the one time pulse duration in Milliseconds between 1 and 60000				
SET LED1 FLASH	ASCII WRITE COMMAND	#SL1FLASH:<ONTIME>,<OFFTIME><CR> Result: #OK<CR>	ASCII	YES
	ONTIME	200		
	OFFTIME	3000		
	TX	#255,SL1FLASH:200,3000<CR>		
	RX	N/A		
Sets the current state of the LED1:GREEN on the cover of module to FLASH and defines the on and off intervals in Milliseconds between 20 and 600000				

Register NAME	MODBUS Register	Register VALUE	DATA TYPE	DO WRITE
LED STATUS:LED2:WHITE				
GET LED2	ASCII READ COMMAND	#GLED2<CR> Result: #GLED2:<LEDMode>,<LEDStateDec>,<LEDStateHex><CR>	ASCII	
	TX	#255,GLED2<CR>		
	RX	#255,GLED2:OFF,0,0x0<CR>		
		Actual LED state:OFF LED ist currently 0		
Returns the actual state of the LED2:WHITE on the cover of module				
LED COMMANDS:LED2:WHITE				
SET LED2 OFF	ASCII WRITE COMMAND	#SL2OFF<CR> Result: #OK<CR>	ASCII	YES
	TX	#255,SL2OFF<CR>		
	RX	N/A		
Sets the current state of the LED2:WHITE on the cover of module to OFF				
SET LED2 ON	ASCII WRITE COMMAND	#SL2ON<CR> Result: #OK<CR>	ASCII	YES
	TX	#255,SL2ON<CR>		
	RX	N/A		
Sets the current state of the LED2:WHITE on the cover of module to ON				
SET LED2 INVERT	ASCII WRITE COMMAND	#SL2INV<CR> Result: #OK<CR>	ASCII	YES
	TX	#255,SL2INV<CR>		
	RX	N/A		
Inverts the current state of the LED2:WHITE on the cover of module from ON to OFF or from OFF to ON				
SET LED2 PULSE	ASCII WRITE COMMAND	#SL2PULSE:<PULSETIME><CR> Result: #OK<CR>	ASCII	YES
	PULSETIME	1000		
	TX	#255,SL2PULSE:1000<CR>		
	RX	N/A		
Sets the current state of the LED2:WHITE on the cover of module to PULSE and defines the one time pulse duration in Milliseconds between 1 and 60000				
SET LED2 BLINK	ASCII WRITE COMMAND	#SL2BLINK:<BLINKTIME><CR> Result: #OK<CR>	ASCII	YES
	BLINKTIME	1000		
	TX	#255,SL2BLINK:1000<CR>		
	RX	N/A		

Sets the current state of the LED2:WHITE on the cover of module to PULSE and defines the one time pulse duration in Milliseconds between 1 and 60000				
SET LED2 FLASH	ASCII WRITE COMMAND	#SL2FLASH:<ONTIME>,<OFFTIME><CR> Result: #OK<CR>	ASCII	YES
	ONTIME	200		
	OFFTIME	3000		
	TX	#255,SL2FLASH:200,3000<CR>		
	RX	N/A		
Sets the current state of the LED2:WHITE on the cover of module to FLASH and defines the on and off intervals in Milliseconds between 20 and 600000				

Register NAME	MODBUS Register	Register VALUE	DATA TYPE	DO WRITE
LED STATUS:LED3:RED				
GET LED3	ASCII READ COMMAND	#GLED3<CR> Result: #GLED3:<LEDMode>,<LEDStateDec>,<LEDStateHex><CR>	ASCII	
	TX	#255,GLED3<CR>		
	RX	#255,GLED3:OFF,0,0x0<CR>		
		Actual LED state:OFF LED ist currently 0		
Returns the actual state of the LED3:RED on the cover of module				
LED COMMANDS:LED3:RED				
SET LED3 OFF	ASCII WRITE COMMAND	#SL3OFF<CR> Result: #OK<CR>	ASCII	YES
	TX	#255,SL3OFF<CR>		
	RX	N/A		
Sets the current state of the LED3:RED on the cover of module to OFF				
SET LED3 ON	ASCII WRITE COMMAND	#SL3ON<CR> Result: #OK<CR>	ASCII	YES
	TX	#255,SL3ON<CR>		
	RX	N/A		
Sets the current state of the LED3:RED on the cover of module to ON				
SET LED3 INVERT	ASCII WRITE COMMAND	#SL3INV<CR> Result: #OK<CR>	ASCII	YES
	TX	#255,SL3INV<CR>		
	RX	N/A		
Inverts the current state of the LED3:RED on the cover of module from ON to OFF or from OFF to ON				
SET LED3 PULSE	ASCII WRITE COMMAND	#SL3PULSE:<PULSETIME><CR> Result: #OK<CR>	ASCII	YES
	PULSETIME	1000		
	TX	#255,SL3PULSE:1000<CR>		
	RX	N/A		
Sets the current state of the LED3:RED on the cover of module to PULSE and defines the one time pulse duration in Milliseconds between 1 and 60000				
SET LED3 BLINK	ASCII WRITE COMMAND	#SL3BLINK:<BLINKTIME><CR> Result: #OK<CR>	ASCII	YES
	BLINKTIME	1000		
	TX	#255,SL3BLINK:1000<CR>		
	RX	N/A		

Sets the current state of the LED3:RED on the cover of module to PULSE and defines the one time pulse duration in Milliseconds between 1 and 60000				
SET LED3 FLASH	ASCII WRITE COMMAND	#SL3FLASH:<ONTIME>,<OFFTIME><CR> Result: #OK<CR>	ASCII	YES
	ONTIME	200		
	OFFTIME	3000		
	TX	#255,SL3FLASH:200,3000<CR>		
	RX	N/A		
Sets the current state of the LED3:RED on the cover of module to FLASH and defines the on and off intervals in Milliseconds between 20 and 600000				

Register NAME	MODBUS Register	Register VALUE	DATA TYPE	DO WRITE
LED STATUS:LED4:YELLOW				
GET LED4	ASCII READ COMMAND	#GLED4<CR> Result: #GLED4:<LEDMode>,<LEDStateDec>,<LEDStateHex><CR>	ASCII	
	TX	#255,GLED4<CR>		
	RX	#255,GLED4:OFF,0,0x0<CR>		
		Actual LED state:OFF LED ist currently 0		
Returns the actual state of the LED4:YELLOW on the cover of module				
LED COMMANDS:LED4:YELLOW				
SET LED4 OFF	ASCII WRITE COMMAND	#SL4OFF<CR> Result: #OK<CR>	ASCII	YES
	TX	#255,SL4OFF<CR>		
	RX	N/A		
Sets the current state of the LED4:YELLOW on the cover of module to OFF				
SET LED4 ON	ASCII WRITE COMMAND	#SL4ON<CR> Result: #OK<CR>	ASCII	YES
	TX	#255,SL4ON<CR>		
	RX	N/A		
Sets the current state of the LED4:YELLOW on the cover of module to ON				
SET LED4 INVERT	ASCII WRITE COMMAND	#SL4INV<CR> Result: #OK<CR>	ASCII	YES
	TX	#255,SL4INV<CR>		
	RX	N/A		
Inverts the current state of the LED4:YELLOW on the cover of module from ON to OFF or from OFF to ON				
SET LED4 PULSE	ASCII WRITE COMMAND	#SL4PULSE:<PULSETIME><CR> Result: #OK<CR>	ASCII	YES
	PULSETIME	1000		
	TX	#255,SL4PULSE:1000<CR>		
	RX	N/A		
Sets the current state of the LED4:YELLOW on the cover of module to PULSE and defines the one time pulse duration in Milliseconds between 1 and 60000				
SET LED4 BLINK	ASCII WRITE COMMAND	#SL4BLINK:<BLINKTIME><CR> Result: #OK<CR>	ASCII	YES
	BLINKTIME	1000		
	TX	#255,SL4BLINK:1000<CR>		
	RX	N/A		

Sets the current state of the LED4:YELLOW on the cover of module to PULSE and defines the one time pulse duration in Milliseconds between 1 and 60000				
SET LED4 FLASH	ASCII WRITE COMMAND	#SL4FLASH:<ONTIME>,<OFFTIME><CR> Result: #OK<CR>	ASCII	YES
	ONTIME	200		
	OFFTIME	3000		
	TX	#255,SL4FLASH:200,3000<CR>		
	RX	N/A		
Sets the current state of the LED4:YELLOW on the cover of module to FLASH and defines the on and off intervals in Milliseconds between 20 and 600000				

Register NAME	MODBUS Register	Register VALUE	NEW REAL VALUE	NEW VALUE	DATA TYPE	DO WRITE
PRODUCT INFO						
HW_GROUP	3x65201 4x65201 I:65200	50176,0xC400 B:C4 00			UINT16 R/O	
This is the group of hardware of the current product						
HW_TYPE	3x65202 4x65202 I:65201	148,0x0094 B:00 94			UINT16 R/O	
This is the type of hardware of the current product						
SW_VERSION	3x65203 4x65203 I:65202	272,0x0110 B:01 10			UINT16 R/O	
		SW VERSION:0.1.0				
This is the current software version of the firmware						
SW_AUTHOR	3x65204 4x65204 I:65203	21321,0x5349 B:53 49			UINT16 R/O	
This is the current software author of the firmware						
MANUFACTURER	3x65205 4x65205 I:65204	1380275017,0x52455349 B:52 45 53 49			UINT32 R/O	
This is the current software author of the firmware						
NUMBER OF DIGITAL INPUTS	3x65207 4x65207 I:65206	16,0x0010 B:00 10			UINT16 R/O	
		Number of DIS:16				
This is the current software version of the firmware						
NUMBER OF DIGITAL OUTPUTS	3x65208 4x65208 I:65207	12,0x000C B:00 0C			UINT16 R/O	
		Number of DOS:12				
This is the current software version of the firmware						
NUMBER OF ANALOG INPUTS	3x65209 4x65209 I:65208	0,0x0000 B:00 00			UINT16 R/O	
		Number of AIS:0				
This is the current software version of the firmware						
NUMBER OF ANALOG OUTPUTS	3x65210 4x65210 I:65209	0,0x0000 B:00 00			UINT16 R/O	
		Number of AOS:0				

This is the current software version of the firmware						
NUMBER OF UNIVERSAL IN/OUTPUTS	3x65211 4x65211 I:65210	4,0x0004 B:00 04			UINT16 R/O	
		Number of AIOX:4				
This is the current software version of the firmware						
NUMBER OF SPECIAL INPUTS	3x65212 4x65212 I:65211	0,0x0000 B:00 00			UINT16 R/O	
		Number of special inputs:0				
This is the current software version of the firmware						
NUMBER OF SPECIAL OUTPUTS	3x65213 4x65213 I:65212	0,0x0000 B:00 00			UINT16 R/O	
		Number of special outputs:0				
This is the current software version of the firmware						
FEATURE1	3x65214 4x65214 I:65213	2,0x0002 B:00 02			UINT16 R/O	
		Feature:RS485				
This is the feature list of the controller: 0:NONE, 1:RS232, 2:RS485, 3:KNX, 4:DALI, 5:MBUS, 6:LORA, 7:LTE, 8:2xETHERNET						
FEATURE2	3x65215 4x65215 I:65214	0,0x0000 B:00 00			UINT16 R/O	
		Feature:NONE				
FEATURE3	3x65216 4x65216 I:65215	0,0x0000 B:00 00			UINT16 R/O	
		Feature:NONE				
FEATURE4	3x65217 4x65217 I:65216	0,0x0000 B:00 00			UINT16 R/O	
		Feature:NONE				
FEATURE5	3x65218 4x65218 I:65217	0,0x0000 B:00 00			UINT16 R/O	
		Feature:NONE				
FEATURE6	3x65219 4x65219 I:65218	0,0x0000 B:00 00			UINT16 R/O	
		Feature:NONE				
FEATURE7	3x65220 4x65220 I:65219	0,0x0000 B:00 00			UINT16 R/O	

		Feature:NONE				
FEATURE8	3x65221 4x65221 I:65220	0,0x0000 B:00 00			UINT16 R/O	
		Feature:NONE				
MODBUS WATCHDOG						
MODBUS WATCHDOG TIME	3x65222 4x65222 I:65221	0,0x0000 B:00 00		50	UINT16 R/W	NO
		Actual watchdog time in 1/100s:0 -> 0,0s				
Writing a value onto this register defines a new time for the internal communication watchdog timer. The value is a timespan in 1/100s. =0: The communication watchdog is disabled =1..65535: Communication watchdog will be trigegred after x 1/100s pause on communication line In case of an communication watchdog, the module sets all outputs to the states defined in the configuration output registers Reading this register will return the current stored time from the internal FRAM						
RASPBERRY PI WATCHDOG TIMER	3x65223 4x65223 I:65222	0,0x0000 B:00 00		50	UINT16 R/W	NO
		remaining watchdog time in 10ms:0 -> 0,000s				
Enables or disables the WATCHDOG Timer for the Raspberry Pi module. 1..65535: Time for Watchdog in x10 Milliseconds (Maximum 655,35 seconds) =0: No Watchdog is generated						
FRAM						
GET FRAM TYPE	3x65224 4x65224 I:65223	2,0x0002 B:00 02			UINT16 R/O	
		FRAM size & type:FM25L16 2kB				
Returns the current ytpe of the FRAM and its total size =2:FM25L16B_G, 2kB =64:FM25V05, 64kB =128:FM25V10, 128kB						
GET FRAM USED BYTES	3x65225 4x65225 I:65224	46,0x002E B:00 2E			UINT16 R/O	
		FRAM used bytes:46				
Returns the amount of used bytes from system in FRAM						
RTC REAL TIME CLOCK						
RTC YEAR	3x65231 4x65231 I:65230	20,0x0014 B:00 14		24	UINT16 R/W	NO
		Actual RTC year:20				
Returns the actual year of the internal real time clock in the range of 24 to 99. Writing to this register prepares the setting of a new time.						

RTC MONTH	3x65232 4x65232 I:65231	1,0x0001 B:00 01		1	UINT16 R/W	NO
		Actual RTC month:1				
Returns the actual month of the internal real time clock in the range of 1 to 12 Writing to this register prepares the setting of a new time.						
RTC DAY	3x65233 4x65233 I:65232	7,0x0007 B:00 07		1	UINT16 R/W	NO
		Actual RTC day:7				
Returns the actual day of the internal real time clock in the range of 1 to 31 Writing to this register prepares the setting of a new time.						
RTC HOUR	3x65234 4x65234 I:65233	21,0x0015 B:00 15		12	UINT16 R/W	NO
		Actual RTC month:21				
Returns the actual hour of the internal real time clock in the range of 0 to 23 Writing to this register prepares the setting of a new time.						
RTC MINUTE	3x65235 4x65235 I:65234	47,0x002F B:00 2F		45	UINT16 R/W	NO
		Actual RTC hour:47				
Returns the actual minute of the internal real time clock in the range of 0 to 59 Writing to this register prepares the setting of a new time.						
RTC SECOND	3x65236 4x65236 I:65235	49,0x0031 B:00 31		30	UINT16 R/W	NO
		Actual RTC second:49				
Returns the actual second of the internal real time clock in the range of 0 to 59 Writing to this register prepares the setting of a new time.						
RTC DAY OF WEEK	3x65237 4x65237 I:65236	2,0x0002 B:00 02		5:FRIDAY	UINT16 R/W	NO
		Actual RTC week day:TUE		SELECT DAY OF WEEK		
Returns the actual day of week in the range 1 to 7 1:MON, 2:TUE, 3:WED, 4:THU, 5:FRI, 6:SAT, 7:SUN Writing to this register writes a new date and time and weekday to the RTC						
DIP SWITCH STATUS						
DIP SWITCH	3x65501 4x65501 I:65500	0,0x0000 B:00 00			UINT16 R/O	

Returns the actual setting 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)						
CPU DATA						
SERIAL1	3x65521 4x65521 I:65520	12544,0x3100 B:31 00			UINT16 R/O	
Serial number of module as 96 bit unsigned integer number						
SERIAL2	3x65522 4x65522 I:65521	6144,0x1800 B:18 00			UINT16 R/O	
SERIAL3	3x65523 4x65523 I:65522	3415,0x0D57 B:0D 57			UINT16 R/O	
SERIAL4	3x65524 4x65524 I:65523	17237,0x4355 B:43 55			UINT16 R/O	
SERIAL5	3x65525 4x65525 I:65524	13619,0x3533 B:35 33			UINT16 R/O	
SERIAL6	3x65526 4x65526 I:65525	14624,0x3920 B:39 20			UINT16 R/O	
		SERIAL:00310018570D554333352039				
Serial number of module as 96 bit unsigned integer number						
CPU TEMPERATURE	3x65527 4x65527 I:65526	4607,0x11FF B:11 FF			UINT16 R/O	
		Actual internal temperature of CPU:46,07°C				
Current internal temperature of CPU in ° Celsius multiplied by 100.						
CPU VOLTAGE	3x65528 4x65528 I:65527	330,0x014A B:01 4A			UINT16 R/O	
		Actual supply voltage of CPU:3,30V				
Current internal supply voltage of CPU in Volt multiplied by 100.						
CPU BACKUP	3x65529 4x65529 I:65528	309,0x0135 B:01 35			UINT16 R/O	
		Actual backup voltage of CPU for RTC:3,09V				
Current internal backup voltage of CPU for RTC in Volt multiplied by 100.						
RESETs						

RASPBERRY PI RESET	3x65534 4x65534 I:65533	0,0x00 B:00 00		1:PERFORM RASPBERRY PI RESET	BIT R/W	NO
Resets the Raspberry Pi						
RASPBERRY PI RESET	3x65534 4x65534 I:65533	0,0x0000 B:00 00		1:PERFORM RASPBERRY PI RESET	UINT16 R/W	NO
Resets the Raspberry Pi						
FACTORY RESET	1x65535 2x65535 I:65534	0,0x00 B:00		1:PERFORM FACTORY RESET	BIT R/W	NO
Performs a factory reset of all internal saved parameters						
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						
RESET	1x65536 2x65536 I:65535	0,0x00 B:00		1:PERFORM ARM RESET	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). Be aware that you will losse all serial interfaces in USB!!!						
RESET	3x65536 4x65535 I:65535	0,0x0000 B:00 00		1:PERFORM ARM RESET	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). Be aware that you will losse all serial interfaces in USB!!!						

Register NAME	MODBUS Register	Register VALUE	NEW REAL VALUE	NEW VALUE	DATA TYPE	DO WRITE
DIP SWITCH STATUS						
DIP SWITCH DIP1	1x65001 2x65001 I:65000	0,0x00 B:00			BIT R/O	
Returns the actual setting of the Dip switches. =0: DIP is OFF =1: DIP is ON						
DIP SWITCH DIP2	1x65002 2x65002 I:65001	0,0x00 B:00			BIT R/O	
DIP SWITCH DIP3	1x65003 2x65003 I:65002	0,0x00 B:00			BIT R/O	
DIP SWITCH DIP4	1x65004 2x65004 I:65003	0,0x00 B:00			BIT R/O	
DIP SWITCH DIP5	1x65005 2x65005 I:65004	0,0x00 B:00			BIT R/O	
DIP SWITCH DIP6	1x65006 2x65006 I:65005	0,0x00 B:00			BIT R/O	
DIP SWITCH DIP7	1x65007 2x65007 I:65006	0,0x00 B:00			BIT R/O	
DIP SWITCH DIP8	1x65008 2x65008 I:65007	0,0x00 B:00			BIT R/O	
LED1:GREEN						
LED1:GREEN SET TO OFF	1x65009 2x65009 I:65008	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil sets the LED to OFF						
LED1:GREEN SET TO ON	1x65010 2x65010 I:65009	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil sets the LED to ON						
LED1:GREEN INVERT LED STATE	1x65011 2x65011 I:65010	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil inverts the actual LED state						
LED1:GREEN BLINK	1x65012 2x65012 I:65011	????		N/A:DO NOTHING	BIT W/O	NO

Writing 1 to this coil start symmetrical blinking of LED with last defined time						
LED1:GREEN FLASH	1x65013 2x65013 I:65012	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil start asymmetrical flashing of LED with last defined times						
LED1:GREEN PULSE	1x65014 2x65014 I:65013	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil start one time pulse of LED with last defined time						
LED1:GREEN BLINK 5s	1x65015 2x65015 I:65014	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil start symmetrical blinking of LED with 5s ON-5s OFF cycle						
LED1:GREEN BLINK 1s	1x65016 2x65016 I:65015	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil start symmetrical blinking of LED with 1s ON-1s OFF cycle						
LED1:GREEN BLINK 250ms	1x65017 2x65017 I:65016	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil start symmetrical blinking of LED with 250ms ON-250ms OFF cycle						
LED1:GREEN BLINK 50ms	1x65018 2x65018 I:65017	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil start symmetrical blinking of LED with 50ms ON-50ms OFF cycle						
LED1:GREEN FLASH 5s-1s	1x65019 2x65019 I:65018	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil start asymmetrical flashing of LED with 5s ON-1s OFF cycle						
LED1:GREEN FLASH 1s-250ms	1x65020 2x65020 I:65019	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil start asymmetrical flashing of LED with 1s ON-250ms OFF cycle						
LED1:GREEN FLASH 500ms-100ms	1x65021 2x65021 I:65020	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil start asymmetrical flashing of LED with 500ms ON-100ms OFF cycle						
LED1:GREEN FLASH 300ms-50ms	1x65022 2x65022 I:65021	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil start asymmetrical flashing of LED with 300ms ON-50ms OFF cycle						
LED1:GREEN PULSE 1s	1x65023 2x65023 I:65022	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil start one time pulse of LED with 1s ON						

LED1:GREEN PULSE 500ms	1x65024 2x65024 I:65023	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil start one time pulse of LED with 500ms ON						
LED1:GREEN PULSE 250ms	1x65025 2x65025 I:65024	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil start one time pulse of LED with 250ms ON						
LED1:GREEN PULSE 100ms	1x65026 2x65026 I:65025	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil start one time pulse of LED with 100ms ON						
LED1:GREEN PULSE 20ms	1x65027 2x65027 I:65026	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil start one time pulse of LED with 20ms ON						
LED2:WHITE						
LED2:WHITE SET TO OFF	1x65029 2x65029 I:65028	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil sets the LED to OFF						
LED2:WHITE SET TO ON	1x65030 2x65030 I:65029	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil sets the LED to ON						
LED2:WHITE INVERT LED STATE	1x65031 2x65031 I:65030	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil inverts the actual LED state						
LED2:WHITE BLINK	1x65032 2x65032 I:65031	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil start symmetrical blinking of LED with last defined time						
LED2:WHITE FLASH	1x65033 2x65033 I:65032	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil start asymmetrical flashing of LED with last defined times						
LED2:WHITE PULSE	1x65034 2x65034 I:65033	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil start one time pulse of LED with last defined time						
LED2:WHITE BLINK 5s	1x65035 2x65035 I:65034	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil start symmetrical blinking of LED with 5s ON-5s OFF cycle						

LED2:WHITE BLINK 1s	1x65036 2x65036 I:65035	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil start symmetrical blinking of LED with 1s ON-1s OFF cycle						
LED2:WHITE BLINK 250ms	1x65037 2x65037 I:65036	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil start symmetrical blinking of LED with 250ms ON-250ms OFF cycle						
LED2:WHITE BLINK 50ms	1x65038 2x65038 I:65037	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil start symmetrical blinking of LED with 50ms ON-50ms OFF cycle						
LED2:WHITE FLASH 5s-1s	1x65039 2x65039 I:65038	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil start asymmetrical flashing of LED with 5s ON-1s OFF cycle						
LED2:WHITE FLASH 1s-250ms	1x65040 2x65040 I:65039	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil start asymmetrical flashing of LED with 1s ON-250ms OFF cycle						
LED2:WHITE FLASH 500ms-100ms	1x65041 2x65041 I:65040	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil start asymmetrical flashing of LED with 500ms ON-100ms OFF cycle						
LED2:WHITE FLASH 300ms-50ms	1x65042 2x65042 I:65041	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil start asymmetrical flashing of LED with 300ms ON-50ms OFF cycle						
LED2:WHITE PULSE 1s	1x65043 2x65043 I:65042	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil start one time pulse of LED with 1s ON						
LED2:WHITE PULSE 500ms	1x65044 2x65044 I:65043	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil start one time pulse of LED with 500ms ON						
LED2:WHITE PULSE 250ms	1x65045 2x65045 I:65044	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil start one time pulse of LED with 250ms ON						
LED2:WHITE PULSE 100ms	1x65046 2x65046 I:65045	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil start one time pulse of LED with 100ms ON						

LED2:WHITE PULSE 20ms	1x65047 2x65047 I:65046	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil start one time pulse of LED with 20ms ON						
LED3:RED						
LED3:RED SET TO OFF	1x65049 2x65049 I:65048	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil sets the LED to OFF						
LED3:RED SET TO ON	1x65050 2x65050 I:65049	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil sets the LED to ON						
LED3:RED INVERT LED STATE	1x65051 2x65051 I:65050	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil inverts the actual LED state						
LED3:RED BLINK	1x65052 2x65052 I:65051	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil start symmetrical blinking of LED with last defined time						
LED3:RED FLASH	1x65053 2x65053 I:65052	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil start asymmetrical flashing of LED with last defined times						
LED3:RED PULSE	1x65054 2x65054 I:65053	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil start one time pulse of LED with last defined time						
LED3:RED BLINK 5s	1x65055 2x65055 I:65054	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil start symmetrical blinking of LED with 5s ON-5s OFF cycle						
LED3:RED BLINK 1s	1x65056 2x65056 I:65055	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil start symmetrical blinking of LED with 1s ON-1s OFF cycle						
LED3:RED BLINK 250ms	1x65057 2x65057 I:65056	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil start symmetrical blinking of LED with 250ms ON-250ms OFF cycle						
LED3:RED BLINK 50ms	1x65058 2x65058 I:65057	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil start symmetrical blinking of LED with 50ms ON-50ms OFF cycle						

LED3:RED FLASH 5s-1s	1x65059 2x65059 I:65058	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil start asymmetrical flashing of LED with 5s ON-1s OFF cycle						
LED3:RED FLASH 1s-250ms	1x65060 2x65060 I:65059	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil start asymmetrical flashing of LED with 1s ON-250ms OFF cycle						
LED3:RED FLASH 500ms-100ms	1x65061 2x65061 I:65060	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil start asymmetrical flashing of LED with 500ms ON-100ms OFF cycle						
LED3:RED FLASH 300ms-50ms	1x65062 2x65062 I:65061	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil start asymmetrical flashing of LED with 300ms ON-50ms OFF cycle						
LED3:RED PULSE 1s	1x65063 2x65063 I:65062	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil start one time pulse of LED with 1s ON						
LED3:RED PULSE 500ms	1x65064 2x65064 I:65063	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil start one time pulse of LED with 500ms ON						
LED3:RED PULSE 250ms	1x65065 2x65065 I:65064	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil start one time pulse of LED with 250ms ON						
LED3:RED PULSE 100ms	1x65066 2x65066 I:65065	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil start one time pulse of LED with 100ms ON						
LED3:RED PULSE 20ms	1x65067 2x65067 I:65066	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil start one time pulse of LED with 20ms ON						
LED4:YELLOW						
LED4:YELLOW SET TO OFF	1x65069 2x65069 I:65068	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil sets the LED to OFF						
LED4:YELLOW SET TO ON	1x65070 2x65070 I:65069	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil sets the LED to ON						

LED4:YELLOW INVERT LED STATE	1x65071 2x65071 I:65070	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil inverts the actual LED state						
LED4:YELLOW BLINK	1x65072 2x65072 I:65071	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil start symmetrical blinking of LED with last defined time						
LED4:YELLOW FLASH	1x65073 2x65073 I:65072	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil start asymmetrical flashing of LED with last defined times						
LED4:YELLOW PULSE	1x65074 2x65074 I:65073	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil start one time pulse of LED with last defined time						
LED4:YELLOW BLINK 5s	1x65075 2x65075 I:65074	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil start symmetrical blinking of LED with 5s ON-5s OFF cycle						
LED4:YELLOW BLINK 1s	1x65076 2x65076 I:65075	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil start symmetrical blinking of LED with 1s ON-1s OFF cycle						
LED4:YELLOW BLINK 250ms	1x65077 2x65077 I:65076	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil start symmetrical blinking of LED with 250ms ON-250ms OFF cycle						
LED4:YELLOW BLINK 50ms	1x65078 2x65078 I:65077	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil start symmetrical blinking of LED with 50ms ON-50ms OFF cycle						
LED4:YELLOW FLASH 5s-1s	1x65079 2x65079 I:65078	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil start asymmetrical flashing of LED with 5s ON-1s OFF cycle						
LED4:YELLOW FLASH 1s-250ms	1x65080 2x65080 I:65079	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil start asymmetrical flashing of LED with 1s ON-250ms OFF cycle						
LED4:YELLOW FLASH 500ms-100ms	1x65081 2x65081 I:65080	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil start asymmetrical flashing of LED with 500ms ON-100ms OFF cycle						

LED4:YELLOW FLASH 300ms-50ms	1x65082 2x65082 I:65081	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil start asymmetrical flashing of LED with 300ms ON-50ms OFF cycle						
LED4:YELLOW PULSE 1s	1x65083 2x65083 I:65082	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil start one time pulse of LED with 1s ON						
LED4:YELLOW PULSE 500ms	1x65084 2x65084 I:65083	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil start one time pulse of LED with 500ms ON						
LED4:YELLOW PULSE 250ms	1x65085 2x65085 I:65084	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil start one time pulse of LED with 250ms ON						
LED4:YELLOW PULSE 100ms	1x65086 2x65086 I:65085	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil start one time pulse of LED with 100ms ON						
LED4:YELLOW PULSE 20ms	1x65087 2x65087 I:65086	????		N/A:DO NOTHING	BIT W/O	NO
Writing 1 to this coil start one time pulse of LED with 20ms ON						
DIP SWITCH STATUS						
DIP SWITCH	3x65501 4x65501 I:65500	0,0x0000 B:00 00			UINT16 R/O	
Returns the actual setting 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)						
LED1:GREEN						
LED1:GREEN STATE	3x65502 4x65502 I:65501	3,0x0003 B:00 03		1:SET TO ON	UINT16 R/W	NO
		State of LED:BLINK				

Returns the actual state of the LED Writing to this register will set a new state for the LED 0: Switch LED permanent OFF 1: Switch LED permanent ON 2: Invert last state of LED 3: Start symmetrical blinking of LED with TIME1 ON and TIME1 OFF 4: Start asymmetrical flashing of LED with TIME1 ON and TIME2 OFF 5: Start one time pulse of LED with TIME1 ON and inifinite OFF						
LED1:GREEN TIME1	3x65503 4x65503 I:65502	1000,0x03E8 B:03 E8		1000	UINT16 R/W	YES
		Actual time 1 in ms:1000				
Returns the actual time1 for blink,flash and pulse ON time in Milliseconds Writing to this register sets a new time in the range 20-65534ms						
LED1:GREEN TIME2	3x65504 4x65504 I:65503	2000,0x07D0 B:07 D0		2000	UINT16 R/W	YES
		Actual time 2 in ms:2000				
Returns the actual time2 for blink and flash OFF time in Milliseconds Writing to this register sets a new time in the range 20-65534ms						
LED2:WHITE						
LED2:WHITE STATE	3x65505 4x65505 I:65504	1,0x0001 B:00 01		1:SET TO ON	UINT16 R/W	NO
		State of LED:ON				
Returns the actual state of the LED Writing to this register will set a new state for the LED 0: Switch LED permanent OFF 1: Switch LED permanent ON 2: Invert last state of LED 3: Start symmetrical blinking of LED with TIME1 ON and TIME1 OFF 4: Start asymmetrical flashing of LED with TIME1 ON and TIME2 OFF 5: Start one time pulse of LED with TIME1 ON and inifinite OFF						
LED2:WHITE TIME1	3x65506 4x65506 I:65505	1000,0x03E8 B:03 E8		1000	UINT16 R/W	YES
		Actual time 1 in ms:1000				
Returns the actual time1 for blink,flash and pulse ON time in Milliseconds Writing to this register sets a new time in the range 20-65534ms						
LED2:WHITE TIME2	3x65507 4x65507 I:65506	2000,0x07D0 B:07 D0		2000	UINT16 R/W	YES
		Actual time 2 in ms:2000				
Returns the actual time2 for blink and flash OFF time in Milliseconds Writing to this register sets a new time in the range 20-65534ms						
LED3:RED						
LED3:RED STATE	3x65508 4x65508 I:65507	0,0x0000 B:00 00		1:SET TO ON	UINT16 R/W	NO

		State of LED:OFF				
Returns the actual state of the LED Writing to this register will set a new state for the LED 0: Switch LED permanent OFF 1: Switch LED permanent ON 2: Invert last state of LED 3: Start symmetrical blinking of LED with TIME1 ON and TIME1 OFF 4: Start asymmetrical flashing of LED with TIME1 ON and TIME2 OFF 5: Start one time pulse of LED with TIME1 ON and inifinite OFF						
LED3:RED TIME1	3x65509 4x65509 I:65508	1000,0x03E8 B:03 E8		1000	UINT16 R/W	YES
		Actual time 1 in ms:1000				
Returns the actual time1 for blink,flash and pulse ON time in Milliseconds Writing to this register sets a new time in the range 20-65534ms						
LED3:RED TIME2	3x65510 4x65510 I:65509	2000,0x07D0 B:07 D0		2000	UINT16 R/W	YES
		Actual time 2 in ms:2000				
Returns the actual time2 for blink and flash OFF time in Milliseconds Writing to this register sets a new time in the range 20-65534ms						
LED4:YELLOW						
LED4:YELLOW STATE	3x65511 4x65511 I:65510	0,0x0000 B:00 00		1:SET TO ON	UINT16 R/W	NO
		State of LED:OFF				
Returns the actual state of the LED Writing to this register will set a new state for the LED 0: Switch LED permanent OFF 1: Switch LED permanent ON 2: Invert last state of LED 3: Start symmetrical blinking of LED with TIME1 ON and TIME1 OFF 4: Start asymmetrical flashing of LED with TIME1 ON and TIME2 OFF 5: Start one time pulse of LED with TIME1 ON and inifinite OFF						
LED4:YELLOW TIME1	3x65512 4x65512 I:65511	1000,0x03E8 B:03 E8		1000	UINT16 R/W	YES
		Actual time 1 in ms:1000				
Returns the actual time1 for blink,flash and pulse ON time in Milliseconds Writing to this register sets a new time in the range 20-65534ms						
LED4:YELLOW TIME2	3x65513 4x65513 I:65512	2000,0x07D0 B:07 D0		2000	UINT16 R/W	YES
		Actual time 2 in ms:2000				
Returns the actual time2 for blink and flash OFF time in Milliseconds Writing to this register sets a new time in the range 20-65534ms						

Command NAME	ASCII command type	ASCII command structure	NEW REAL VALUE	NEW VALUE	DATA TYPE	DO WRITE
ASCII COMMANDS						
DIGITAL INPUTS						
GET DIGITAL INPUTS	ASCII READ COMMAND	#GDIS<CR> Result: #GDIS:<DISDec>,<DISHex><CR>			ASCII	
	TX	#1,GDIS<CR>				
	RX	#1,GDIS:0,0x0<CR>				
		Actual status of digital inputs:00.0000.0000				
Returns the actual state of all 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 2: State of DI3 (=0:OFF, =1:ON) ... Bit 7: State of DI8 (=0:OFF, =1:ON) Bit 8: State of DI9 (=0:OFF, =1:ON) Bit 9: State of DI10 (=0:OFF, =1:ON)							
GET DIGITAL INPUT DIx	ASCII READ COMMAND	#GDI<DINR><CR> Result: #GDI<DINR>:<DIxDec>,<DIxHex><CR>			ASCII	
	DINR	1				
	TX	#1,GDI1<CR>				
	RX	#1,GDI1:0,0x0<CR>				
		Actual status of digital input DI1:0=OFF				
<DINR>: 1=DI1..10=DI10						
Returns the actual state of the digital input DIx as decimal number and as hexadecimal number. 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	#1,GAC<CR>				
	RX	#1,GAC:0,0x0<CR>				
		Actual 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 PART x	ASCII READ COMMAND	#CADISP<PART><CR> Result: #CADISP<PART>:<ChangeDInDec>,...,<ChangeDIn+9Dec>, <ChangeDInHex>,...,<ChangeDIn+9Hex><CR>	ASCII	
	PART	1		
	TX	#1,CADISP1<CR>		
	RX	#1,CADISP1:0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0<CR>		
		Actual counter for changes on DI1:0		
		Actual counter for changes on DI2:0		
		Actual counter for changes on DI3:0		
		Actual counter for changes on DI4:0		
		Actual counter for changes on DI5:0		
		Actual counter for changes on DI6:0		
		Actual counter for changes on DI7:0		
		Actual counter for changes on DI8:0		
		Actual counter for changes on DI9:0		
		Actual counter for changes on DI10:0		
<PART>: 1, 1=DI1-DI10				
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 The parameter <PART> defines the part of the digital inputs. The command returns maximal 16 digital inputs.				
CHANGE DIx	ASCII READ COMMAND	#CDI<DINR><CR> Result: #CDI<DINR>:<ChangesDec>,<ChangesHex><CR>	ASCII	
	DINR	1		
	TX	#1,CDI1<CR>		
	RX	#1,CDI1:0,0x0<CR>		
		Actual counter for changes on digital input DI1:0		
<DINR>: 1=DI1..10=DI10				
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 PART x	ASCII READ COMMAND	#SKADISP<PART><CR> Result: #SKADISP<PART>:<ShortKeyDInDec>,...,<ShortKeyDIn+9Dec>, <ShortKeyDInHex>,...,<ShortKeyDIn+9Hex><CR>	ASCII	
	PART	1		
	TX	#1,SKADISP1<CR>		

	RX	#1,SKADISP1:0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0<CR>		
		Actual counter for short keypress events on DI1:0		
		Actual counter for short keypress events on DI2:0		
		Actual counter for short keypress events on DI3:0		
		Actual counter for short keypress events on DI4:0		
		Actual counter for short keypress events on DI5:0		
		Actual counter for short keypress events on DI6:0		
		Actual counter for short keypress events on DI7:0		
		Actual counter for short keypress events on DI8:0		
		Actual counter for short keypress events on DI9:0		
		Actual counter for short keypress events on DI10:0		
<PART>: 1, 1=DI1-DI10				
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. The parameter <PART> defines the part of the digital inputs. The command returns maximal 16 digital inputs.				
SHORT KEY DIx	ASCII READ COMMAND	#SKDI<DINR><CR> Result: #SKDI<DINR>:<ShortKeyDec>,<ShortKeyHex><CR>	ASCII	
	DINR	1		
	TX	#1,SKDI1<CR>		
	RX	#1,SKDI1:0,0x0<CR>		
		Actual counter for short keypress events on digital input DI1:0		
<DINR>: 1=DI1..10=DI10				
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 PART x	ASCII READ COMMAND	#LKSADISP<PART><CR> Result: #LKSADISP<PART>:<LongKeyStartDInDec>,...,<LongKeyStartDIn+9Dec>, <LongKeyStartDInHex>,...,<LongKeyStartDIn+9Hex><CR>	ASCII	
	PART	1		
	TX	#1,LKSADISP1<CR>		
	RX	#1,LKSADISP1:0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0<CR>		
		Actual counter for long keypress start events on DI1:0		
		Actual counter for long keypress start events on DI2:0		
		Actual counter for long keypress start events on DI3:0		
		Actual counter for long keypress start events on DI4:0		
		Actual counter for long keypress start events on DI5:0		
		Actual counter for long keypress start events on DI6:0		
		Actual counter for long keypress start events on DI7:0		
		Actual counter for long keypress start events on DI8:0		
		Actual counter for long keypress start events on DI9:0		
		Actual counter for long keypress start events on DI10:0		

<PART>: 1, 1=DI1-DI10				
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. The parameter <PART> defines the part of the digital inputs. The command returns maximal 16 digital inputs.				
LONG KEY START DIx	ASCII READ COMMAND	#LKSDI<DINR><CR> Result: #LKSDI<DINR>:<LongKeyStartDec>,<LongKeyStartHex><CR>	ASCII	
	DINR	1		
	TX	#1,LKSDI1<CR>		
	RX	#1,LKSDI1:0,0x0<CR>		
		Actual counter for long keypress start events on digital input DI1:0		
<DINR>: 1=DI1..10=DI10				
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 PART x	ASCII READ COMMAND	#LKEADISP<PART><CR> Result: #LKEADISP<PART>:<LongKeyEndDInDec>,...,<LongKeyEndDIn+9Dec>, <LongKeyEndDInHex>,...,<LongKeyEndDIn+9Hex><CR>	ASCII	
	PART	1		
	TX	#1,LKEADISP1<CR>		
	RX	#1,LKEADISP1:0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0<CR>		
		Actual counter for long keypress end events on DI1:0		
		Actual counter for long keypress end events on DI2:0		
		Actual counter for long keypress end events on DI3:0		
		Actual counter for long keypress end events on DI4:0		
		Actual counter for long keypress end events on DI5:0		
		Actual counter for long keypress end events on DI6:0		
		Actual counter for long keypress end events on DI7:0		
		Actual counter for long keypress end events on DI8:0		
		Actual counter for long keypress end events on DI9:0		
		Actual counter for long keypress end events on DI10:0		
<PART>: 1, 1=DI1-DI10				
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. The parameter <PART> defines the part of the digital inputs. The command returns maximal 16 digital inputs.				
LONG KEY END DIx	ASCII READ COMMAND	#LKEDI<DINR><CR> Result: #LKEDI<DINR>:<LongKeyEndDec>,<LongKeyEndHex><CR>	ASCII	
	DINR	1		
	TX	#1,LKEDI1<CR>		
	RX	#1,LKEDI1:0,0x0<CR>		
		Actual counter for long keypress end events on digital input DI1:0		
<DINR>: 1=DI1..10=DI10				

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 PART x	ASCII READ COMMAND	#RADISP<PART><CR> Result: #RADISP<PART>:<RiseDInDec>,...,<RiseDIn+9Dec>, <RiseDInHex>,...,<RiseDIn+9Hex><CR>	ASCII	
	PART	1		
	TX	#1,RADISP1<CR>		
	RX	#1,RADISP1:0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x 0<CR>		
		Actual counter for rising edges on DI1:0		
		Actual counter for rising edges on DI2:0		
		Actual counter for rising edges on DI3:0		
		Actual counter for rising edges on DI4:0		
		Actual counter for rising edges on DI5:0		
		Actual counter for rising edges on DI6:0		
		Actual counter for rising edges on DI7:0		
		Actual counter for rising edges on DI8:0		
		Actual counter for rising edges on DI9:0		
		Actual counter for rising edges on DI10:0		
<PART>: 1, 1=DI1-DI10				
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. The parameter <PART> defines the part of the digital inputs. The command returns maximal 16 digital inputs.				
RISE DIx	ASCII READ COMMAND	#RDI<DINR><CR> Result: #RDI<DINR>:<RiseDec>,<RiseHex><CR>	ASCII	
	DINR	1		
	TX	#1,RDI1<CR>		
	RX	#1,RDI1:0,0x0<CR>		
		Actual counter for rising edges on digital input DI1:0		
<DINR>: 1=DI1..10=DI10				
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 PART x	ASCII READ COMMAND	#FADISP<PART><CR> Result: #FADISP<PART>:<FallDInDec>,...,<FallDIn+9Dec>, <FallDInHex>,...,<FallDIn+9Hex><CR>	ASCII	
	PART	1		
	TX	#1,FADISP1<CR>		
	RX	#1,FADISP1:0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x 0<CR>		
		Actual counter for falling edges on DI1:0		
		Actual counter for falling edges on DI2:0		
		Actual counter for falling edges on DI3:0		

		Actual counter for falling edges on DI4:0		
		Actual counter for falling edges on DI5:0		
		Actual counter for falling edges on DI6:0		
		Actual counter for falling edges on DI7:0		
		Actual counter for falling edges on DI8:0		
		Actual counter for falling edges on DI9:0		
		Actual counter for falling edges on DI10:0		
<PART>: 1, 1=DI1-DI10				
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. The parameter <PART> defines the part of the digital inputs. The command returns maximal 16 digital inputs.				
FALL DIx	ASCII READ COMMAND	#FDI<DINR><CR> Result: #FDI<DINR>:<FallDec>,<FallHex><CR>	ASCII	
	DINR	1		
	TX	#1,FDI1<CR>		
	RX	#1,FDI1:0,0x0<CR>		
		Actual counter for falling edges on digital input DI1:0		
<DINR>: 1=DI1..10=DI10				
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	NO
	TX	#1,RC<CR>		
	RX	N/A		
Resets all internal counters for digital inputs and events on this digital inputs to 0.				
DIGITAL INPUTS EVENTs				
EVENTS ON	ASCII WRITE COMMAND	#EVTON<CR> Result: #OK<CR>	ASCII	NO
	TX	#1,EVTON<CR>		
	RX	#1,OK<CR>		
Activates event sending of changes on digital inputs Whenever a change is detected on the digital inputs, the IO module sends immediately #<BusAdr>,EVT:DIS:<AllDISasDec>,<AllDISasHex><CR>				
EVENTS OFF	ASCII WRITE COMMAND	#EVTOFF<CR> Result: #OK<CR>	ASCII	NO
	TX	#1,EVTOFF<CR>		
	RX	#1,OK<CR>		
Deactivates event sending of changes on digital inputs Whenever a change is detected on the digital inputs, the IO module sends immediately #<BusAdr>,EVT:DIS:<AllDISasDec>,<AllDISasHex><CR>				

Command NAME	ASCII command type	ASCII command structure	NEW REAL VALUE	NEW VALUE	DATA TYPE	DO WRITE
ASCII COMMANDS						
DIGITAL OUTPUTS						
UPDATE DIGITAL INPUTS AND OUTPUTS	ASCII WRITE COMMAND	#UDIOS:<OutAllDOS><CR> Result: #UDIOS:0,0x0<CR>			ASCII	YES
	DO1	0:OFF				
	DO2	0:OFF				
	DO3	0:OFF				
	DO4	0:OFF				
	DO5	0:OFF				
	DO6	0:OFF				
	DO7	0:OFF				
	DO8	0:OFF				
	TX	#255,UDIOS:0<CR>				
	RX	#255,UDIOS:1,0x1<CR>				
		Actual status of digital inputs:00.0000.0001				
Sets all digital outputs to the new state OutAllDOS and gives back the current status of all digital inputs InAllDIS as decimal and hexadecimal value OutAllDOS: The new state for all digital outputs Bit 0: State of DO1 (=0:OFF, =1:ON) Bit 1: State of DO2 (=0:OFF, =1:ON) Bit 2: State of DO3 (=0:OFF, =1:ON) ... Bit 5: State of DO6 (=0:OFF, =1:ON) Bit 6: State of DO7 (=0:OFF, =1:ON) Bit 7: State of DO8 (=0:OFF, =1:ON)							
SET DIGITAL OUTPUTS	ASCII WRITE COMMAND	#SDOS:<OutAllDOS><CR> Result: #OK<CR>			ASCII	YES
	DO1	0:OFF				
	DO2	0:OFF				
	DO3	0:OFF				
	DO4	0:OFF				
	DO5	0:OFF				
	DO6	0:OFF				
	DO7	0:OFF				
	DO8	0:OFF				
	TX	#255,SDOS:0<CR>				
	RX	#255,OK<CR>				

Sets all digital outputs to the new state OutAllDOS The new state for all digital outputs Bit 0: State of DO1 (=0:OFF, =1:ON) Bit 1: State of DO2 (=0:OFF, =1:ON) Bit 2: State of DO3 (=0:OFF, =1:ON) ... Bit 5: State of DO6 (=0:OFF, =1:ON) Bit 6: State of DO7 (=0:OFF, =1:ON) Bit 7: State of DO8 (=0:OFF, =1:ON)				
SET DIGITAL OUTPUT DOx	ASCII WRITE COMMAND	#SDO<DONR>:<Out><CR> Result: #OK<CR>	ASCII	NO
	DONR	2		
	DOx	0:OFF		
	TX	#255,SDO2:0<CR>		
	RX	N/A		
<DONR>: 1=DO1..8=DO8				
Sets the new state for digital output DOx. The state is defined with <Out>. Out The new state of the digital output DOx: =0: digital output is OFF =1: digital output is ON				
GET DIGITAL OUTPUTS	ASCII READ COMMAND	#GDOS<CR> Result: #GDOS:<DOSDec>,<DOSHex><CR>	ASCII	
	TX	#255,GDOS<CR>		
	RX	#255,GDOS:0,0x0<CR>		
		Actual status of digital outputs:0000.0000		
Returns the actual state of the digital outputs as decimal number and as hexadecimal number. DOSDec, DOSHex The current state of the digital outputs: Bit 0: State of DO1 (=0:OFF, =1:ON) Bit 1: State of DO2 (=0:OFF, =1:ON) Bit 2: State of DO3 (=0:OFF, =1:ON) ... Bit 5: State of DO6 (=0:OFF, =1:ON) Bit 6: State of DO7 (=0:OFF, =1:ON) Bit 7: State of DO8 (=0:OFF, =1:ON)				
GET DIGITAL OUTPUT DOx	ASCII READ COMMAND	#GDO<DONR><CR> Result: #GDO<DONR>:<DOxDec>,<DOxHex><CR>	ASCII	
	DONR	2		
	TX	#255,GDO2<CR>		
	RX	#255,GDO2:0,0x0<CR>		
		Actual status of digital output DO2:0=OFF		

Returns the actual state of the digital output DOx as decimal number and as hexadecimal number.
DOxDec, DOxHex
The current state of the digital output DOx:
=0: relay output is OFF
=1: relay output is ON

DIGITAL OUTPUTS: PULSE OUTPUT

PULSE DOx	ASCII WRITE COMMAND	#PDO<DONR>:<Time><CR> Result: #OK<CR>	ASCII	YES
	DONR	2		
	TIME	200		
	TX	#255,PDO2:200<CR>		
	RX	#255,OK<CR>		

<DONR>: 1=DO1..8=DO8

<Time>: 0..65535*100ms

This command switches the digital output DOx on for the pulse duration <PulseTimeIn100ms>*100ms.
PulseTimeIn100ms: A duration in 100ms units.
The corresponding digital output is switched on for this time period.

GET PULSE TIMER DOx	ASCII READ COMMAND	#GPT<DONR><CR> Result: #GPT:<TimeDec>,<TimeHex><CR>	ASCII	
	DONR	2		
	TX	#255,GPT2<CR>		
	RX	#255,GPT2:0,0x0<CR>		
		Actual pulse time for DO2:0,0s		

<DONR>: 1=DO1..8=DO8

Returns the remaining timer value of the pulse for digital output DOx in ms.
PulseTimeInMSDec, PulseTimeInMSHex
The remaining time of the pulse in Milliseconds

FAN COIL #1-#2

SET FAN COIL FCx	ASCII WRITE COMMAND	#SFC<FCNR>:<Mode><CR> Result: #OK<CR>	ASCII	YES
	FCNR	1		
	MODE	9999:DEACTIVATED		
	TX	#255,SFC1:9999<CR>		
	RX	#1,OK<CR>		

Sets a new mode for FAN COIL functionality on RO1, RO2 and RO3 of fan coil group.
Fan coul groups: 1:RO1,RO2,RO3, 2:RO5,RO6,RO7, 3:RO9,RO10,RO11
=9999: This function is not used
=0: All three ROs aure OFF
=1: STAGE 1: RO1 ist ON, RO2,RO3 are OFF
=2: STAGE 2: RO2 is ON, RO1,RO3 are OFF
=3: STAGE 3: RO3 is ON, RO1,RO2 are OFF

In this mode the module inserts a pause with no outputs on, when swtiching from one stage to another stage. Also a minimum time for each stage is maintained by the module

GET FAN COIL FCx	ASCII READ COMMAND	#GFC<FCNR><CR> Result: #GFC<FCNR>:<ModeDec>,<ModeHex><CR>	ASCII	
	FCNR	1		
	TX	#255,GFC1<CR>		
	RX	#1,GFC1:9999,0x270F<CR>		
		Current mode fo FC:9999->DEACTIVATED		
Current mode for FAN COIL functionality on RO1, RO2 and RO3 of fan coil group: =9999: This function is not used =0: All three ROs aure OFF =1: STAGE 1: RO1 ist ON, RO2,RO3 are OFF =2: STAGE 2: RO2 is ON, RO1,RO3 are OFF =3: STAGE 3: RO3 is ON, RO1,RO2 are OFF In this mode the module inserts a pause with no outputs on, when swtiching from one stage to another stage. Also a minimum time for each stage is maintained by the module				
SET PAUSE TIME FCx	ASCII WRITE COMMAND	#SPTFC<FCNR>:<Time><CR> Result: #OK<CR>	ASCII	YES
	FCNR	1		
	TIME	3,123		
	TX	#255,SPTFC1:3123<CR>		
	RX	#1,OK<CR>		
Sets a new pause time with no relays ON between stage switching. Time is defined in 1ms units (0 to 65,535 Seconds selectable)				
GET PAUSE TIME FCx	ASCII READ COMMAND	#GPTFC<FCNR><CR> Result: #GPTFC<FCNR>:<TimeDec>,<TimeHex><CR>	ASCII	
	FCNR	1		
	TX	#255,GPTFC1<CR>		
	RX	#1,GPTFC1:3123,0xC33<CR>		
		Current pause time for FC1:3,123s		
Returns the pause time with no relays ON between stage switching. Time is defined in 1ms units (0 to 65,535 Seconds selectable)				
SET STAGE TIME FCx	ASCII WRITE COMMAND	#SSTFC<FCNR>:<Time><CR> Result: #OK<CR>	ASCII	YES
	FCNR	1		
	TIME	7,250		
	TX	#255,SSTFC1:7250<CR>		
	RX	#1,OK<CR>		
Sets the minimum time for a stage in 1ms units. If the module activates a new stage 1 to 3, it keeps this stage at least for this time span, before changing to another stage.				
GET STAGE TIME FCx	ASCII READ COMMAND	#GSTFC<FCNR><CR> Result: #GSTFC<FCNR>:<TimeDec>,<TimeHex><CR>	ASCII	
	FCNR	1		

	TX	#255,GSTFC1<CR>		
	RX	#1,GSTFC1:7250,0x1C52<CR>		
		Current stage time for FC1:7,250s		
Returns the minimum time for a stage in 1ms units. If the module activates a new stage 1 to 3, it keeps this stage at least for this time span, before changing to another stage.				

Command NAME	ASCII command type	ASCII command structure	NEW REAL VALUE	NEW VALUE	DATA TYPE	DO WRITE
ASCII COMMANDS						
DIGITAL OUTPUTS						
INITIAL & WATCHDOG STATE FOR DIGITAL OUTPUTS						
SET INITIAL & WATCHDOG STATE FOR DIGITAL OUTPUTS	ASCII WRITE COMMAND	#SCDOS:<OutAllDOS><CR> Result: #OK<CR>			ASCII	YES
	DO1	0:OFF				
	DO2	0:OFF				
	DO3	0:OFF				
	DO4	0:OFF				
	DO5	0:OFF				
	DO6	0:OFF				
	DO7	0:OFF				
	DO8	0:OFF				
	TX	#255,SCDOS:0<CR>				
	RX	#255,OK<CR>				
This command sets all digital outputs to a new state for controller restart and watchdog function. The state is saved in FRAM. This state is used after power on and after a communication watchdog timeout, if a watchdog time is configured. OutAllDOS The new state for the digital outputs: Bit 0: New state of DO1 (=0:OFF, =1:ON) Bit 1: New state of DO2 (=0:OFF, =1:ON) ... Bit 6: New state of DO7 (=0:OFF, =1:ON) Bit 7: New state of DO8 (=0:OFF, =1:ON)						
GET INITIAL & WATCHDOG STATE FOR DIGITAL OUTPUTS	ASCII READ COMMAND	#GCDOS<CR> Result: #GCDOS:<DOSDec>,<DOSHex><CR>			ASCII	
	TX	#255,GCDOS<CR>				
	RX	#255,GCDOS:0,0x0<CR>				
		Init & watchdog configuration for digital outputs:				
		DO1-DO8:0000.0000				
Returns the actual initial and watchdog state of the digital outputs as decimal number and as hexadecimal number. This state is used after power on and after a communication watchdog timeout, if a watchdog time is configured DOSDec, DOSHex The current state of the digital outputs: Bit 0: State of DO1 (=0:OFF, =1:ON) Bit 1: State of DO2 (=0:OFF, =1:ON) ... Bit 6: State of DO7 (=0:OFF, =1:ON) Bit 7: State of DO8 (=0:OFF, =1:ON)						

Register NAME Command NAME	MODBUS Register ASCII Command	Register VALUE ASCII Command	NEW REAL VALUE	NEW VALUE	DATA TYPE	DO WRITE
STATUS DIGITAL INPUTS						
DI1	1x00001 2x00001 I:0	0,0x00 B:00			BIT R/O	
		Actual 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	
		Actual state of DI2:0=OFF				
DI3	1x00003 2x00003 I:2	0,0x00 B:00			BIT R/O	
		Actual state of DI3:0=OFF				
DI4	1x00004 2x00004 I:3	0,0x00 B:00			BIT R/O	
		Actual state of DI4:0=OFF				
DI5	1x00005 2x00005 I:4	0,0x00 B:00			BIT R/O	
		Actual state of DI5:0=OFF				
DI6	1x00006 2x00006 I:5	0,0x00 B:00			BIT R/O	
		Actual state of DI6:0=OFF				
DI7	1x00007 2x00007 I:6	0,0x00 B:00			BIT R/O	
		Actual state of DI7:0=OFF				
DI8	1x00008 2x00008 I:7	0,0x00 B:00			BIT R/O	
		Actual state of DI8:0=OFF				
DI9	1x00009 2x00009 I:8	0,0x00 B:00			BIT R/O	
		Actual state of DI9:0=OFF				

DI10	1x00010 2x00010 I:9	0,0x00 B:00			BIT R/O	
		Actual state of DI10:0=OFF				
STATUS DIGITAL OUTPUTS						
DO1	1x00011 2x00011 I:10	0,0x00 B:00		1	BIT R/W	NO
		Actual state of DO1:0=OFF		ENTER NEW STATE (0 or 1)		
Current state of the digital output DOx =0:DO is OFF, =1:DO is ON Writing on this register changes the state of the digital output						
DO2	1x00012 2x00012 I:11	0,0x00 B:00		0	BIT R/W	NO
		Actual state of DO2:0=OFF		ENTER NEW STATE (0 or 1)		
DO3	1x00013 2x00013 I:12	0,0x00 B:00		0	BIT R/W	NO
		Actual state of DO3:0=OFF		ENTER NEW STATE (0 or 1)		
DO4	1x00014 2x00014 I:13	0,0x00 B:00		0	BIT R/W	NO
		Actual state of DO4:0=OFF		ENTER NEW STATE (0 or 1)		
DO5	1x00015 2x00015 I:14	0,0x00 B:00		0	BIT R/W	NO
		Actual state of DO5:0=OFF		ENTER NEW STATE (0 or 1)		
DO6	1x00016 2x00016 I:15	0,0x00 B:00		0	BIT R/W	NO
		Actual state of DO6:0=OFF		ENTER NEW STATE (0 or 1)		
DO7	1x00017 2x00017 I:16	0,0x00 B:00		0	BIT R/W	NO
		Actual state of DO7:0=OFF		ENTER NEW STATE (0 or 1)		
DO8	1x00018 2x00018 I:17	0,0x00 B:00		0	BIT R/W	NO
		Actual state of DO8:0=OFF		ENTER NEW STATE (0 or 1)		
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 REAL DIGITAL INPUTS						
DI1	1x15001 2x14001 I:15000	0,0x00 B:00			BIT R/O	
		Actual state of DI1:0=OFF				
Current state of the digital input DIx with the internal software filter to suppress glitches or spike on thie line =0:DI is OFF, =1:DI is ON						
DI2	1x15002 2x15002 I:15001	0,0x00 B:00			BIT R/O	
		Actual state of DI2:0=OFF				
DI3	1x15003 2x15003 I:15002	0,0x00 B:00			BIT R/O	
		Actual state of DI3:0=OFF				
DI4	1x15004 2x15004 I:15003	0,0x00 B:00			BIT R/O	
		Actual state of DI4:0=OFF				
DI5	1x15005 2x15005 I:15004	0,0x00 B:00			BIT R/O	
		Actual state of DI5:0=OFF				
DI6	1x15006 2x15006 I:15005	0,0x00 B:00			BIT R/O	
		Actual state of DI6:0=OFF				
DI7	1x15007 2x15007 I:15006	0,0x00 B:00			BIT R/O	
		Actual state of DI7:0=OFF				
DI8	1x15008 2x15008 I:15007	0,0x00 B:00			BIT R/O	
		Actual state of DI8:0=OFF				
DI9	1x15009 2x15009 I:15008	0,0x00 B:00			BIT R/O	
		Actual state of DI9:0=OFF				
DI10	1x15010 2x15010 I:15009	0,0x00 B:00			BIT R/O	
		Actual state of DI10:0=OFF				
STATUS DIGITAL INPUTS						

UNFILTERED DI1	1x15011 2x15011 I:15010	0,0x00 B:00			BIT R/O	
		Actual state of UNFILTERED DI1:0=OFF				
Current state of the real digital input DIx without the internal software filter to suppress glitches or spike on thie line =0:DI is OFF, =1:DI is ON						
UNFILTERED DI2	1x15012 2x15012 I:15011	0,0x00 B:00			BIT R/O	
		Actual state of UNFILTERED DI2:0=OFF				
UNFILTERED DI3	1x15013 2x15013 I:15012	0,0x00 B:00			BIT R/O	
		Actual state of UNFILTERED DI3:0=OFF				
UNFILTERED DI4	1x15014 2x15014 I:15013	0,0x00 B:00			BIT R/O	
		Actual state of UNFILTERED DI4:0=OFF				
UNFILTERED DI5	1x15015 2x15015 I:15014	0,0x00 B:00			BIT R/O	
		Actual state of UNFILTERED DI5:0=OFF				
UNFILTERED DI6	1x15016 2x15016 I:15015	0,0x00 B:00			BIT R/O	
		Actual state of UNFILTERED DI6:0=OFF				
UNFILTERED DI7	1x15017 2x15017 I:15016	0,0x00 B:00			BIT R/O	
		Actual state of UNFILTERED DI7:0=OFF				
UNFILTERED DI8	1x15018 2x15018 I:15017	0,0x00 B:00			BIT R/O	
		Actual state of UNFILTERED DI8:0=OFF				
UNFILTERED DI9	1x15019 2x15019 I:15018	0,0x00 B:00			BIT R/O	
		Actual state of UNFILTERED DI9:0=OFF				
UNFILTERED DI10	1x15020 2x15020 I:15019	0,0x00 B:00			BIT R/O	
		Actual state of UNFILTERED DI10:0=OFF				
STATUS DIGITAL OUTPUTS						
DO1	1x16001 2x16001 I:16000	0,0x00 B:00		1	BIT R/W	NO

		Actual state of DO1:0=OFF		ENTER NEW STATE (0 or 1)		
Current state of the digital output DOx =0:DO is OFF, =1:DO is ON Writing on this register changes the state of the digital output						
DO2	1x16002 2x16002 I:16001	0,0x00 B:00		0	BIT R/W	NO
		Actual state of DO2:0=OFF		ENTER NEW STATE (0 or 1)		
DO3	1x16003 2x16003 I:16002	0,0x00 B:00		0	BIT R/W	NO
		Actual state of DO3:0=OFF		ENTER NEW STATE (0 or 1)		
DO4	1x16004 2x16004 I:16003	0,0x00 B:00		0	BIT R/W	NO
		Actual state of DO4:0=OFF		ENTER NEW STATE (0 or 1)		
DO5	1x16005 2x16005 I:16004	0,0x00 B:00		0	BIT R/W	NO
		Actual state of DO5:0=OFF		ENTER NEW STATE (0 or 1)		
DO6	1x16006 2x16006 I:16005	0,0x00 B:00		0	BIT R/W	NO
		Actual state of DO6:0=OFF		ENTER NEW STATE (0 or 1)		
DO7	1x16007 2x16007 I:16006	0,0x00 B:00		0	BIT R/W	NO
		Actual state of DO7:0=OFF		ENTER NEW STATE (0 or 1)		
DO8	1x16008 2x16008 I:16007	0,0x00 B:00		0	BIT R/W	NO
		Actual state of DO8:0=OFF		ENTER NEW STATE (0 or 1)		

Register NAME Command NAME	MODBUS Register ASCII Command	Register VALUE ASCII Command	NEW REAL VALUE	NEW VALUE	DATA TYPE	DO WRITE
STATUS DIGITAL INPUTS						
DI1	3x00001 4x00001 I:0	0,0x0000 B:00 00			UINT16 R/O	
		Actual 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	
		Actual state of DI2:0=OFF				
DI3	3x00003 4x00003 I:2	0,0x0000 B:00 00			UINT16 R/O	
		Actual state of DI3:0=OFF				
DI4	3x00004 4x00004 I:3	0,0x0000 B:00 00			UINT16 R/O	
		Actual state of DI4:0=OFF				
DI5	3x00005 4x00005 I:4	0,0x0000 B:00 00			UINT16 R/O	
		Actual state of DI5:0=OFF				
DI6	3x00006 4x00006 I:5	0,0x0000 B:00 00			UINT16 R/O	
		Actual state of DI6:0=OFF				
DI7	3x00007 4x00007 I:6	0,0x0000 B:00 00			UINT16 R/O	
		Actual state of DI7:0=OFF				
DI8	3x00008 4x00008 I:7	0,0x0000 B:00 00			UINT16 R/O	
		Actual state of DI8:0=OFF				
DI9	3x00009 4x00009 I:8	0,0x0000 B:00 00			UINT16 R/O	
		Actual state of DI9:0=OFF				
DI10	3x00010 4x00010 I:9	0,0x0000 B:00 00			UINT16 R/O	

		Actual state of DI10:0=OFF				
STATUS DIGITAL OUTPUTS						
DO1	3x00011 4x00011 I:10	0,0x0000 B:00 00		1	UINT16 R/W	NO
		Actual state of DO1:0=OFF		ENTER NEW STATE (0 or 1)		
Current state of the digital output DOx =0:DO is OFF, =1:DO is ON Writing on this register changes the state of the digital output						
DO2	3x00012 4x00012 I:11	0,0x0000 B:00 00		0	UINT16 R/W	NO
		Actual state of DO2:0=OFF		ENTER NEW STATE (0 or 1)		
DO3	3x00013 4x00013 I:12	0,0x0000 B:00 00		0	UINT16 R/W	NO
		Actual state of DO3:0=OFF		ENTER NEW STATE (0 or 1)		
DO4	3x00014 4x00014 I:13	0,0x0000 B:00 00		0	UINT16 R/W	NO
		Actual state of DO4:0=OFF		ENTER NEW STATE (0 or 1)		
DO5	3x00015 4x00015 I:14	0,0x0000 B:00 00		0	UINT16 R/W	NO
		Actual state of DO5:0=OFF		ENTER NEW STATE (0 or 1)		
DO6	3x00016 4x00016 I:15	0,0x0000 B:00 00		0	UINT16 R/W	NO
		Actual state of DO6:0=OFF		ENTER NEW STATE (0 or 1)		
DO7	3x00017 4x00017 I:16	0,0x0000 B:00 00		0	UINT16 R/W	NO
		Actual state of DO7:0=OFF		ENTER NEW STATE (0 or 1)		
DO8	3x00018 4x00018 I:17	0,0x0000 B:00 00		0	UINT16 R/W	NO
		Actual state of DO8:0=OFF		ENTER NEW STATE (0 or 1)		
DIGITAL INPUTS: RESET						
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..DI10	3x10002 4x10002 I:10001	0,0x0000 B:00 00			UINT16 R/O	
		Actual state of DI1:0=OFF				
		Actual state of DI2:0=OFF				
		Actual state of DI3:0=OFF				
		Actual state of DI4:0=OFF				
		Actual state of DI5:0=OFF				
		Actual state of DI6:0=OFF				
		Actual state of DI7:0=OFF				
		Actual state of DI8:0=OFF				
		Actual state of DI9:0=OFF				
		Actual state of DI10:0=OFF				
Actual state of all digital inputs DI1..DI12 Bit 0: =0:DI1 is OFF, =1:DI1 is ON Bit 1: =0:DI2 is OFF, =1:DI2 is ON ... Bit 8: =0:DI9 is OFF, =1:DI9 is ON Bit 9: =0:DI10 is OFF, =1:DI10 is ON						
STATUS OF DIGITAL OUTPUTS						
STATUS OF ALL DOS DO1-DO8	3x10003 4x10003 I:10002	0,0x0000 B:00 00		0x00FF	UINT16 R/W	NO
		Actual state of DO1:0=OFF		1		
		Actual state of DO2:0=OFF		1		
		Actual state of DO3:0=OFF		1		
		Actual state of DO4:0=OFF		1		
		Actual state of DO5:0=OFF		1		
		Actual state of DO6:0=OFF		1		
		Actual state of DO7:0=OFF		1		
		Actual state of DO8:0=OFF		1		
Actual state of all digital outputs Bit 0: =0:DO1 is OFF, =1:DO1 is ON Bit 1: =0:DO2 is OFF, =1:DO2 is ON ... Bit 6: =0:DO7 is OFF, =1:DO7 is ON Bit 7: =0:DO8 is OFF, =1:DO8 is ON Write on this register sets all digital outputs to a new state						
STATUS OF DIGITAL OUTPUTS						
REAL STATUS OF ALL DOS DO1-DO8	3x10501 4x10501 I:10500	0,0x0000 B:00 00			UINT16 R/O	
		Real state of DO1:0=OFF				

		Real state of DO2:0=OFF				
		Real state of DO3:0=OFF				
		Real state of DO4:0=OFF				
		Real state of DO5:0=OFF				
		Real state of DO6:0=OFF				
		Real state of DO7:0=OFF				
		Real state of DO8:0=OFF				
Actual state of all digital outputs in the DO chips Bit 0: =0:DO1 is OFF, =1:DO1 is ON Bit 1: =0:DO2 is OFF, =1:DO2 is ON ... Bit 6: =0:DO7 is OFF, =1:DO8 is ON Bit 7: =0:DO8 is OFF, =1:DO7 is ON Write on this register sets all digital outputs to a new state						
STATUS REAL DIGITAL INPUTS						
DI1	3x15001 4x15001 I:15000	0,0x0000 B:00 00			UINT16 R/O	
		Actual state of DI1:0=OFF				
Current state of the digital input DIx with the internal software filter to suppress glitches or spike on thie line =0:DI is OFF, =1:DI is ON						
DI2	3x15002 4x15002 I:15001	0,0x0000 B:00 00			UINT16 R/O	
		Actual state of DI2:0=OFF				
DI3	3x15003 4x15003 I:15002	0,0x0000 B:00 00			UINT16 R/O	
		Actual state of DI3:0=OFF				
DI4	3x15004 4x15004 I:15003	0,0x0000 B:00 00			UINT16 R/O	
		Actual state of DI4:0=OFF				
DI5	3x15005 4x15005 I:15004	0,0x0000 B:00 00			UINT16 R/O	
		Actual state of DI5:0=OFF				
DI6	3x15006 4x15006 I:15005	0,0x0000 B:00 00			UINT16 R/O	
		Actual state of DI6:0=OFF				
DI7	3x15007 4x15007 I:15006	0,0x0000 B:00 00			UINT16 R/O	
		Actual state of DI7:0=OFF				

DI8	3x15008 4x15008 I:15007	0,0x0000 B:00 00			UINT16 R/O	
		Actual state of DI8:0=OFF				
DI9	3x15009 4x15009 I:15008	0,0x0000 B:00 00			UINT16 R/O	
		Actual state of DI9:0=OFF				
DI10	3x15010 4x15010 I:15009	0,0x0000 B:00 00			UINT16 R/O	
		Actual state of DI10:0=OFF				
STATUS DIGITAL INPUTS						
UNFILTERED DI1	3x15011 4x15011 I:15010	0,0x0000 B:00 00			UINT16 R/O	
		Actual state of UNFILTERED DI1:0=OFF				
Current state of the real digital input DIx without the internal software filter to suppress glitches or spike on thie line =0:DI is OFF, =1:DI is ON						
UNFILTERED DI2	3x15012 4x15012 I:15011	0,0x0000 B:00 00			UINT16 R/O	
		Actual state of UNFILTERED DI2:0=OFF				
UNFILTERED DI3	3x15013 4x15013 I:15012	0,0x0000 B:00 00			UINT16 R/O	
		Actual state of UNFILTERED DI3:0=OFF				
UNFILTERED DI4	3x15014 4x15014 I:15013	0,0x0000 B:00 00			UINT16 R/O	
		Actual state of UNFILTERED DI4:0=OFF				
UNFILTERED DI5	3x15015 4x15015 I:15014	0,0x0000 B:00 00			UINT16 R/O	
		Actual state of UNFILTERED DI5:0=OFF				
UNFILTERED DI6	3x15016 4x15016 I:15015	0,0x0000 B:00 00			UINT16 R/O	
		Actual state of UNFILTERED DI6:0=OFF				
UNFILTERED DI7	3x15017 4x15017 I:15016	0,0x0000 B:00 00			UINT16 R/O	
		Actual state of UNFILTERED DI7:0=OFF				
UNFILTERED DI8	3x15018 4x15018 I:15017	0,0x0000 B:00 00			UINT16 R/O	
		Actual state of UNFILTERED DI8:0=OFF				

UNFILTERED DI9	3x15019 4x15019 I:15018	0,0x0000 B:00 00			UINT16 R/O	
		Actual state of UNFILTERED DI9:0=OFF				
UNFILTERED DI10	3x15020 4x15020 I:15019	0,0x0000 B:00 00			UINT16 R/O	
		Actual state of UNFILTERED DI10:0=OFF				
STATUS DIGITAL OUTPUTS						
DO1	3x16001 4x16001 I:16000	0,0x0000 B:00 00		1	UINT16 R/W	NO
		Actual state of DO1:0=OFF		ENTER NEW STATE (0 or 1)		
Current state of the digital output DOx =0:DO is OFF, =1:DO is ON Writing on this register changes the state of the digital output						
DO2	3x16002 4x16002 I:16001	0,0x0000 B:00 00		0	UINT16 R/W	NO
		Actual state of DO2:0=OFF		ENTER NEW STATE (0 or 1)		
DO3	3x16003 4x16003 I:16002	0,0x0000 B:00 00		0	UINT16 R/W	NO
		Actual state of DO3:0=OFF		ENTER NEW STATE (0 or 1)		
DO4	3x16004 4x16004 I:16003	0,0x0000 B:00 00		0	UINT16 R/W	NO
		Actual state of DO4:0=OFF		ENTER NEW STATE (0 or 1)		
DO5	3x16005 4x16005 I:16004	0,0x0000 B:00 00		0	UINT16 R/W	NO
		Actual state of DO5:0=OFF		ENTER NEW STATE (0 or 1)		
DO6	3x16006 4x16006 I:16005	0,0x0000 B:00 00		0	UINT16 R/W	NO
		Actual state of DO6:0=OFF		ENTER NEW STATE (0 or 1)		
DO7	3x16007 4x16007 I:16006	0,0x0000 B:00 00		0	UINT16 R/W	NO
		Actual state of DO7:0=OFF		ENTER NEW STATE (0 or 1)		
DO8	3x16008 4x16008 I:16007	0,0x0000 B:00 00		0	UINT16 R/W	NO
		Actual state of DO8:0=OFF		ENTER NEW STATE (0 or 1)		

Register NAME Command NAME	MODBUS Register ASCII Command	Register VALUE ASCII Command	NEW REAL VALUE	NEW VALUE	DATA TYPE	DO WRITE
DIGITAL INPUTS: DIGITAL INPUT HAS CHANGED IT'S STATE						
DI HAS CHANGED DI1	1x20001 2x20001 I:20000	0,0x00 B:00			BIT R/O	
If the digital input has changed this bit inverts its last state						
DI HAS CHANGED DI2	1x20002 2x20002 I:20001	0,0x00 B:00			BIT R/O	
DI HAS CHANGED DI3	1x20003 2x20003 I:20002	0,0x00 B:00			BIT R/O	
DI HAS CHANGED DI4	1x20004 2x20004 I:20003	0,0x00 B:00			BIT R/O	
DI HAS CHANGED DI5	1x20005 2x20005 I:20004	0,0x00 B:00			BIT R/O	
DI HAS CHANGED DI6	1x20006 2x20006 I:20005	0,0x00 B:00			BIT R/O	
DI HAS CHANGED DI7	1x20007 2x20007 I:20006	0,0x00 B:00			BIT R/O	
DI HAS CHANGED DI8	1x20008 2x20008 I:20007	0,0x00 B:00			BIT R/O	
DI HAS CHANGED DI9	1x20009 2x20009 I:20008	0,0x00 B:00			BIT R/O	
DI HAS CHANGED DI10	1x20010 2x20010 I:20009	0,0x00 B:00			BIT R/O	
DIGITAL INPUTS: SHORT KEYPRESS EVENT ON DIGITAL INPUT DETECTED						
SHORT KEYPRESS ON DI1	1x20011 2x20011 I:20010	0,0x00 B:00			BIT R/O	
If a short keypress event was detected on the digital input this bit inverts its last state						
SHORT KEYPRESS ON DI2	1x20012 2x20012 I:20011	0,0x00 B:00			BIT R/O	

SHORT KEYPRESS ON DI3	1x20013 2x20013 I:20012	0,0x00 B:00			BIT R/O	
SHORT KEYPRESS ON DI4	1x20014 2x20014 I:20013	0,0x00 B:00			BIT R/O	
SHORT KEYPRESS ON DI5	1x20015 2x20015 I:20014	0,0x00 B:00			BIT R/O	
SHORT KEYPRESS ON DI6	1x20016 2x20016 I:20015	0,0x00 B:00			BIT R/O	
SHORT KEYPRESS ON DI7	1x20017 2x20017 I:20016	0,0x00 B:00			BIT R/O	
SHORT KEYPRESS ON DI8	1x20018 2x20018 I:20017	0,0x00 B:00			BIT R/O	
SHORT KEYPRESS ON DI9	1x20019 2x20019 I:20018	0,0x00 B:00			BIT R/O	
SHORT KEYPRESS ON DI10	1x20020 2x20020 I:20019	0,0x00 B:00			BIT R/O	
DIGITAL INPUTS: LONG KEYPRESS START EVENT ON DIGITAL INPUT DETECTED						
LONG KEYPRESS START ON DI1	1x20021 2x20021 I:20020	0,0x00 B:00			BIT R/O	
If a long keypress start event was detected on the digital input this bit inverts its last state						
LONG KEYPRESS START ON DI2	1x20022 2x20022 I:20021	0,0x00 B:00			BIT R/O	
LONG KEYPRESS START ON DI3	1x20023 2x20023 I:20022	0,0x00 B:00			BIT R/O	
LONG KEYPRESS START ON DI4	1x20024 2x20024 I:20023	0,0x00 B:00			BIT R/O	
LONG KEYPRESS START ON DI5	1x20025 2x20025 I:20024	0,0x00 B:00			BIT R/O	
LONG KEYPRESS START ON DI6	1x20026 2x20026 I:20025	0,0x00 B:00			BIT R/O	

LONG KEYPRESS START ON DI7	1x20027 2x20027 I:20026	0,0x00 B:00			BIT R/O	
LONG KEYPRESS START ON DI8	1x20028 2x20028 I:20027	0,0x00 B:00			BIT R/O	
LONG KEYPRESS START ON DI9	1x20029 2x20029 I:20028	0,0x00 B:00			BIT R/O	
LONG KEYPRESS START ON DI10	1x20030 2x20030 I:20029	0,0x00 B:00			BIT R/O	
DIGITAL INPUTS: LONG KEYPRESS END EVENT ON DIGITAL INPUT DETECTED						
LONG KEYPRESS END ON DI1	1x20031 2x20031 I:20030	0,0x00 B:00			BIT R/O	
If a long keypress end event was detected on the digital input this bit inverts its last state						
LONG KEYPRESS END ON DI2	1x20032 2x20032 I:20031	0,0x00 B:00			BIT R/O	
LONG KEYPRESS END ON DI3	1x20033 2x20033 I:20032	0,0x00 B:00			BIT R/O	
LONG KEYPRESS END ON DI4	1x20034 2x20034 I:20033	0,0x00 B:00			BIT R/O	
LONG KEYPRESS END ON DI5	1x20035 2x20035 I:20034	0,0x00 B:00			BIT R/O	
LONG KEYPRESS END ON DI6	1x20036 2x20036 I:20035	0,0x00 B:00			BIT R/O	
LONG KEYPRESS END ON DI7	1x20037 2x20037 I:20036	0,0x00 B:00			BIT R/O	
LONG KEYPRESS END ON DI8	1x20038 2x20038 I:20037	0,0x00 B:00			BIT R/O	
LONG KEYPRESS END ON DI9	1x20039 2x20039 I:20038	0,0x00 B:00			BIT R/O	
LONG KEYPRESS END ON DI10	1x20040 2x20040 I:20039	0,0x00 B:00			BIT R/O	
DIGITAL INPUTS: RISING EDGE ON DIGITAL INPUT DETECTED						

RISING EDGE ON DI1	1x20041 2x20041 I:20040	0,0x00 B:00			BIT R/O	
If a rising edge was detected on the digital input this bit inverts its last state						
RISING EDGE ON DI2	1x20042 2x20042 I:20041	0,0x00 B:00			BIT R/O	
RISING EDGE ON DI3	1x20043 2x20043 I:20042	0,0x00 B:00			BIT R/O	
RISING EDGE ON DI4	1x20044 2x20044 I:20043	0,0x00 B:00			BIT R/O	
RISING EDGE ON DI5	1x20045 2x20045 I:20044	0,0x00 B:00			BIT R/O	
RISING EDGE ON DI6	1x20046 2x20046 I:20045	0,0x00 B:00			BIT R/O	
RISING EDGE ON DI7	1x20047 2x20047 I:20046	0,0x00 B:00			BIT R/O	
RISING EDGE ON DI8	1x20048 2x20048 I:20047	0,0x00 B:00			BIT R/O	
RISING EDGE ON DI9	1x20049 2x20049 I:20048	0,0x00 B:00			BIT R/O	
RISING EDGE ON DI10	1x20050 2x20050 I:20049	0,0x00 B:00			BIT R/O	
DIGITAL INPUTS: FALLING EDGE ON DIGITAL INPUT DETECTED						
FALLING EDGE ON DI1	1x20051 2x20051 I:20050	0,0x00 B:00			BIT R/O	
If a falling edge was detected on the digital input this bit inverts its last state						
FALLING EDGE ON DI2	1x20052 2x20052 I:20051	0,0x00 B:00			BIT R/O	
FALLING EDGE ON DI3	1x20053 2x20053 I:20052	0,0x00 B:00			BIT R/O	
FALLING EDGE ON DI4	1x20054 2x20054 I:20053	0,0x00 B:00			BIT R/O	

FALLING EDGE ON DI5	1x20055 2x20055 I:20054	0,0x00 B:00			BIT R/O	
FALLING EDGE ON DI6	1x20056 2x20056 I:20055	0,0x00 B:00			BIT R/O	
FALLING EDGE ON DI7	1x20057 2x20057 I:20056	0,0x00 B:00			BIT R/O	
FALLING EDGE ON DI8	1x20058 2x20058 I:20057	0,0x00 B:00			BIT R/O	
FALLING EDGE ON DI9	1x20059 2x20059 I:20058	0,0x00 B:00			BIT R/O	
FALLING EDGE ON DI10	1x20060 2x20060 I:20059	0,0x00 B:00			BIT R/O	

Register NAME Command NAME	MODBUS Register ASCII Command	Register VALUE ASCII Command	NEW REAL VALUE	NEW VALUE	DATA TYPE	DO WRITE
PULSE TIME FOR DIGITAL OUTPUTS						
PULSE TIME DO1	3x20001 4x20001 I:20000	0,0x0000 B:00 00	200	20,0	UINT16 R/W	YES
Generate a pulse on digital output x in 100ms units (0,1 to 6553,5 Seconds selectable) If you write onto this register, the digital output will be switched on for the desired time in 100ms units.						
PULSE TIME DO2	3x20002 4x20002 I:20001	0,0x0000 B:00 00	300	30,0	UINT16 R/W	NO
PULSE TIME DO3	3x20003 4x20003 I:20002	0,0x0000 B:00 00	400	40,0	UINT16 R/W	NO
PULSE TIME DO4	3x20004 4x20004 I:20003	0,0x0000 B:00 00	500	50,0	UINT16 R/W	NO
PULSE TIME DO5	3x20005 4x20005 I:20004	0,0x0000 B:00 00	20	2,0	UINT16 R/W	NO
PULSE TIME DO6	3x20006 4x20006 I:20005	0,0x0000 B:00 00	20	2,0	UINT16 R/W	NO
PULSE TIME DO7	3x20007 4x20007 I:20006	0,0x0000 B:00 00	20	2,0	UINT16 R/W	NO
PULSE TIME DO8	3x20008 4x20008 I:20007	0,0x0000 B:00 00	20	2,0	UINT16 R/W	NO
PULSE STATUS FOR DIGITAL OUTPUTS						
PULSE TIMER DO1	3x21001 4x21001 I:21000	0,0x00000000 B:00 00 00 00			UINT32 R/O	
		0,0 seconds				
Remaining time of the pulse on digital output x in Milliseconds.						
PULSE TIMER DO2	3x21003 4x21003 I:21002	0,0x00000000 B:00 00 00 00			UINT32 R/O	
		0,0 seconds				
PULSE TIMER DO3	3x21005 4x21005 I:21004	0,0x00000000 B:00 00 00 00			UINT32 R/O	

		0,0 seconds				
PULSE TIMER DO4	3x21007 4x21007 I:21006	0,0x00000000 B:00 00 00 00			UINT32 R/O	
		0,0 seconds				
PULSE TIMER DO5	3x21009 4x21009 I:21008	0,0x00000000 B:00 00 00 00			UINT32 R/O	
		0,0 seconds				
PULSE TIMER DO6	3x21011 4x21011 I:21010	0,0x00000000 B:00 00 00 00			UINT32 R/O	
		0,0 seconds				
PULSE TIMER DO7	3x21013 4x21013 I:21012	0,0x00000000 B:00 00 00 00			UINT32 R/O	
		0,0 seconds				
PULSE TIMER DO8	3x21015 4x21015 I:21014	0,0x00000000 B:00 00 00 00			UINT32 R/O	
		0,0 seconds				
PULSE STATUS FOR DIGITAL OUTPUTS						
PULSE TIMER DO1	3x21017 4x21017 I:21016	0,0x00000000 B:00 00 00 00			UINT32R R/O	
		0,0 seconds				
Remaining time of the pulse on digital output x in Milliseconds.						
PULSE TIMER DO2	3x21019 4x21019 I:21018	0,0x00000000 B:00 00 00 00			UINT32R R/O	
		0,0 seconds				
PULSE TIMER DO3	3x21021 4x21021 I:21020	0,0x00000000 B:00 00 00 00			UINT32R R/O	
		0,0 seconds				
PULSE TIMER DO4	3x21023 4x21023 I:21022	0,0x00000000 B:00 00 00 00			UINT32R R/O	
		0,0 seconds				
PULSE TIMER DO5	3x21025 4x21025 I:21024	0,0x00000000 B:00 00 00 00			UINT32R R/O	
		0,0 seconds				

PULSE TIMER DO6	3x21027 4x21027 I:21026	0,0x00000000 B:00 00 00 00			UINT32R R/O	
		0,0 seconds				
PULSE TIMER DO7	3x21029 4x21029 I:21028	0,0x00000000 B:00 00 00 00			UINT32R R/O	
		0,0 seconds				
PULSE TIMER DO8	3x21031 4x21031 I:21030	0,0x00000000 B:00 00 00 00			UINT32R R/O	
		0,0 seconds				
FAN COIL #1						
FC1 MODE	3x40001 4x40001 I:40000	9999,0x270F B:27 0F		3:LEVEL 3	UINT16 R/W	YES
		Current mode of FC:9999->DEACTIVATED				
Current mode for FAN COIL functionality on RO1, RO2 and RO3: =9999: This function is not used =0: All three ROs are OFF =1: STAGE 1: RO1 is ON, RO2,RO3 are OFF =2: STAGE 2: RO2 is ON, RO1,RO3 are OFF =3: STAGE 3: RO3 is ON, RO1,RO2 are OFF In this mode the module inserts a pause with no outputs on, when switching from one stage to another stage. Also a minimum time for each stage is maintained by the module						
FC1 PAUSE TIME	3x40002 4x40002 I:40001	3123,0x0C33 B:0C 33	5000	5,0	UINT16 R/W	NO
		3,123 seconds				
Sets and returns the pause time with no relays ON between stage switching. Time is defined in 1ms units (0 to 65,535 Seconds selectable)						
FC1 STAGE TIME	3x40003 4x40003 I:40002	7250,0x00001C52 B:00 00 1C 52	10000	10,0	UINT32 R/W	NO
		7,250 seconds				
Sets and returns the minimum time for a stage in 1ms units. If the module activates a new stage 1 to 3, it keeps this stage at least for this time span, before changing to another stage.						
FC1 STAGE TIME	3x40005 4x40005 I:40004	7250,0x00001C52 B:1C 52 00 00	10000	10,0	UINT32R R/W	NO
		7,250 seconds				
Sets and returns the minimum time for a stage in 1ms units. If the module activates a new stage 1 to 3, it keeps this stage at least for this time span, before changing to another stage.						
FAN COIL #2						
FC2 MODE	3x40012 4x40021 I:40020	1000,0x03E8 B:03 E8		9999:DEACTIVATED	UINT16 R/W	YES
		Current mode of FC:1000->N/A				

Current mode for FAN COIL functionality on RO5, RO6 and RO7: =9999: This function is not used =0: All three ROs are OFF =1: STAGE 1: RO9 is ON, RO10,RO11 are OFF =2: STAGE 2: RO10 is ON, RO9,RO11 are OFF =3: STAGE 3: RO11 is ON, RO9,RO10 are OFF In this mode the module inserts a pause with no outputs on, when switching from one stage to another stage. Also a minimum time for each stage is maintained by the module						
FC2 PAUSE TIME	3x40013 4x40013 I:40012	0,0x0000 B:00 00	5000	5,0	UINT16 R/W	NO
		0,000 seconds				
Sets and returns the pause time with no relays ON between stage switching. Time is defined in 1ms units (0 to 65,535 Seconds selectable)						
FC2 STAGE TIME	3x40014 4x40014 I:40013	983055000,0x3A983A98 B:3A 98 3A 98	10000	10,0	UINT32 R/W	YES
		983055,000 seconds				
Sets and returns the minimum time for a stage in 1ms units. If the module activates a new stage 1 to 3, it keeps this stage at least for this time span, before changing to another stage.						
FC2 STAGE TIME	3x40016 4x40016 I:40015	0,0x00000000 B:00 00 00 00	10000	10,0	UINT32R R/W	NO
		0,000 seconds				
Sets and returns the minimum time for a stage in 1ms units. If the module activates a new stage 1 to 3, it keeps this stage at least for this time span, before changing to another stage.						

Register NAME Command NAME	MODBUS Register ASCII Command	Register VALUE ASCII Command	NEW REAL VALUE	NEW VALUE	DATA TYPE	DO WRITE
DIGITAL INPUTS						
STATUS DI1 A	3x05001 4x05001 I:5000	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	3x05002 4x05002 I:5001	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	3x05003 4x05003 I:5002	0,0x0000 B:00 00			UINT16 R/O	
		DI:0,CC:0,REC:0,FEC:0				
STATUS DI2 B	3x05004 4x05004 I:5003	0,0x0000 B:00 00			UINT16 R/O	
		DI:0,SKE:0,LKSE:0,LKEE:0				
STATUS DI3 A	3x05005 4x05005 I:5004	0,0x0000 B:00 00			UINT16 R/O	
		DI:0,CC:0,REC:0,FEC:0				
STATUS DI3 B	3x05006 4x05006 I:5005	0,0x0000 B:00 00			UINT16 R/O	
		DI:0,SKE:0,LKSE:0,LKEE:0				
STATUS DI4 A	3x05007 4x05007 I:5006	0,0x0000 B:00 00			UINT16 R/O	
		DI:0,CC:0,REC:0,FEC:0				
STATUS DI4 B	3x05008 4x05008 I:5007	0,0x0000 B:00 00			UINT16 R/O	

		DI:0,SKE:0,LKSE:0,LKEE:0			
STATUS DI5 A	3x05009 4x05009 I:5008	0,0x0000 B:00 00		UINT16 R/O	
		DI:0,CC:0,REC:0,FEC:0			
STATUS DI5 B	3x05010 4x05010 I:5009	0,0x0000 B:00 00		UINT16 R/O	
		DI:0,SKE:0,LKSE:0,LKEE:0			
STATUS DI6 A	3x05011 4x05011 I:5010	0,0x0000 B:00 00		UINT16 R/O	
		DI:0,CC:0,REC:0,FEC:0			
STATUS DI6 B	3x05012 4x05012 I:5011	0,0x0000 B:00 00		UINT16 R/O	
		DI:0,SKE:0,LKSE:0,LKEE:0			
STATUS DI7 A	3x05013 4x05013 I:5012	0,0x0000 B:00 00		UINT16 R/O	
		DI:0,CC:0,REC:0,FEC:0			
STATUS DI7 B	3x05014 4x05014 I:5013	0,0x0000 B:00 00		UINT16 R/O	
		DI:0,SKE:0,LKSE:0,LKEE:0			
STATUS DI8 A	3x05015 4x05015 I:5014	0,0x0000 B:00 00		UINT16 R/O	
		DI:0,CC:0,REC:0,FEC:0			
STATUS DI8 B	3x05016 4x05016 I:5015	0,0x0000 B:00 00		UINT16 R/O	
		DI:0,SKE:0,LKSE:0,LKEE:0			
STATUS DI9 A	3x05017 4x05017 I:5016	0,0x0000 B:00 00		UINT16 R/O	
		DI:0,CC:0,REC:0,FEC:0			
STATUS DI9 B	3x05018 4x05018 I:5017	0,0x0000 B:00 00		UINT16 R/O	
		DI:0,SKE:0,LKSE:0,LKEE:0			
STATUS DI10 A	3x05019 4x05019 I:5018	0,0x0000 B:00 00		UINT16 R/O	
		DI:0,CC:0,REC:0,FEC:0			

STATUS DI10 B	3x05020 4x05020 I:5019	0,0x0000 B:00 00			UINT16 R/O	
		DI:0,SKE:0,LKSE:0,LKEE:0				
STATUS						
FILTER PATTERN DI1	3x05021 4x05021 I:5020	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	3x05023 4x05023 I:5022	0,0x00000000 B:00 00 00 00			UINT32 R/O	
FILTER PATTERN DI3	3x05025 4x05025 I:5024	0,0x00000000 B:00 00 00 00			UINT32 R/O	
FILTER PATTERN DI4	3x05027 4x05027 I:5026	0,0x00000000 B:00 00 00 00			UINT32 R/O	
FILTER PATTERN DI5	3x05029 4x05029 I:5028	0,0x00000000 B:00 00 00 00			UINT32 R/O	
FILTER PATTERN DI6	3x05031 4x05031 I:5030	0,0x00000000 B:00 00 00 00			UINT32 R/O	
FILTER PATTERN DI7	3x05033 4x05033 I:5032	0,0x00000000 B:00 00 00 00			UINT32 R/O	
FILTER PATTERN DI8	3x05035 4x05035 I:5034	0,0x00000000 B:00 00 00 00			UINT32 R/O	
FILTER PATTERN DI9	3x05037 4x05037 I:5036	0,0x00000000 B:00 00 00 00			UINT32 R/O	
FILTER PATTERN DI10	3x05039 4x05039 I:5038	0,0x00000000 B:00 00 00 00			UINT32 R/O	
DIGITAL INPUTS: STATUS FOR DIGITAL INPUT DI1						
RISE DI1	3x07001 4x07001 I:7000	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	3x07002 4x07002 I:7001	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	3x07003 4x07003 I:7002	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	3x07004 4x07004 I:7003	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	3x07005 4x07005 I:7004	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	3x07006 4x07006 I:7005	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	3x07011 4x07011 I:7010	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
FALL DI2	3x07012 4x07012 I:7011	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				

CHANGE DI2	3x07013 4x07013 I:7012	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
SHORT KEYPRESS DI2	3x07014 4x07014 I:7013	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
LONG KEYPRESS START DI2	3x07015 4x07015 I:7014	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
LONG KEYPRESS END DI2	3x07016 4x07016 I:7015	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
DIGITAL INPUTS: STATUS FOR DIGITAL INPUT DI3						
RISE DI3	3x07021 4x07021 I:7020	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
FALL DI3	3x07022 4x07022 I:7021	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
CHANGE DI3	3x07023 4x07023 I:7022	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
SHORT KEYPRESS DI3	3x07024 4x07024 I:7023	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
LONG KEYPRESS START DI3	3x07025 4x07025 I:7024	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
LONG KEYPRESS END DI3	3x07026 4x07026 I:7025	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
DIGITAL INPUTS: STATUS FOR DIGITAL INPUT DI4						
RISE DI4	3x07031 4x07031 I:7030	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				

FALL DI4	3x07032 4x07032 I:7031	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
CHANGE DI4	3x07033 4x07033 I:7032	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
SHORT KEYPRESS DI4	3x07034 4x07034 I:7033	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
LONG KEYPRESS START DI4	3x07035 4x07035 I:7034	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
LONG KEYPRESS END DI4	3x07036 4x07036 I:7035	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
DIGITAL INPUTS: STATUS FOR DIGITAL INPUT DI5						
RISE DI5	3x07041 4x07041 I:7040	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
FALL DI5	3x07042 4x07042 I:7041	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
CHANGE DI5	3x07043 4x07043 I:7042	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
SHORT KEYPRESS DI5	3x07044 4x07044 I:7043	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
LONG KEYPRESS START DI5	3x07045 4x07045 I:7044	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
LONG KEYPRESS END DI5	3x07046 4x07046 I:7045	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
DIGITAL INPUTS: STATUS FOR DIGITAL INPUT DI6						

RISE DI6	3x07051 4x07051 I:7050	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
FALL DI6	3x07052 4x07052 I:7051	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
CHANGE DI6	3x07053 4x07053 I:7052	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
SHORT KEYPRESS DI6	3x07054 4x07054 I:7053	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
LONG KEYPRESS START DI6	3x07055 4x07055 I:7054	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
LONG KEYPRESS END DI6	3x07056 4x07056 I:7055	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
DIGITAL INPUTS: STATUS FOR DIGITAL INPUT DI7						
RISE DI7	3x07061 4x07061 I:7060	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
FALL DI7	3x07062 4x07062 I:7061	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
CHANGE DI7	3x07063 4x07063 I:7062	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
SHORT KEYPRESS DI7	3x07064 4x07064 I:7063	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
LONG KEYPRESS START DI7	3x07065 4x07065 I:7064	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				

LONG KEYPRESS END DI7	3x07066 4x07066 I:7065	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
DIGITAL INPUTS: STATUS FOR DIGITAL INPUT DI8						
RISE DI8	3x07071 4x07071 I:7070	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
FALL DI8	3x07072 4x07072 I:7071	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
CHANGE DI8	3x07073 4x07073 I:7072	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
SHORT KEYPRESS DI8	3x07074 4x07074 I:7073	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
LONG KEYPRESS START DI8	3x07075 4x07075 I:7074	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
LONG KEYPRESS END DI8	3x07076 4x07076 I:7075	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
DIGITAL INPUTS: STATUS FOR DIGITAL INPUT DI9						
RISE DI9	3x07081 4x07081 I:7080	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
FALL DI9	3x07082 4x07082 I:7081	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
CHANGE DI9	3x07083 4x07083 I:7082	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
SHORT KEYPRESS DI9	3x07084 4x07084 I:7083	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				

LONG KEYPRESS START DI9	3x07085 4x07085 I:7084	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
LONG KEYPRESS END DI9	3x07086 4x07086 I:7085	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
DIGITAL INPUTS: STATUS FOR DIGITAL INPUT DI10						
RISE DI10	3x07091 4x07091 I:7090	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
FALL DI10	3x07092 4x07092 I:7091	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
CHANGE DI10	3x07093 4x07093 I:7092	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
SHORT KEYPRESS DI10	3x07094 4x07094 I:7093	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
LONG KEYPRESS START DI10	3x07095 4x07095 I:7094	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				
LONG KEYPRESS END DI10	3x07096 4x07096 I:7095	0,0x0000 B:00 00			UINT16 R/O	
		0 event(s)				

Register NAME Command NAME	MODBUS Register ASCII Command	Register VALUE ASCII Command	NEW REAL VALUE	NEW VALUE	DATA TYPE	DO WRITE
INITIAL & WATCHDOG STATUS FOR ALL DIGITAL OUTPUTS						
FRAM INTIAL & WATCHDOG STATUS OF DO1-DO8	3x59001 4x59001 I:59000	0,0x0000 B:00 00		0x00FF	UINT16 R/W	YES
		Actual init & watchdog state of DO1:0=OFF		1		
		Actual init & watchdog state of DO2:0=OFF		1		
		Actual init & watchdog state of DO3:0=OFF		1		
		Actual init & watchdog state of DO4:0=OFF		1		
		Actual init & watchdog state of DO5:0=OFF		1		
		Actual init & watchdog state of DO6:0=OFF		1		
		Actual init & watchdog state of DO7:0=OFF		1		
		Actual init & watchdog state of DO8:0=OFF		1		

Current FRAM setting of initial and watchdog state of all digital outputs. This state is used after power on and after a communication watchdog timeout, if a watchdog time is configured
Bit 0: =0:DO1 is OFF, =1:DO1 is ON
Bit 1: =0:DO2 is OFF, =1:DO2 is ON
...
Bit 6: =0:DO7 is OFF, =1:DO7 is ON
Bit 7: =0:DO8 is OFF, =1:DO8 is ON

Write on this register sets all digital outputs to a new state for module restart and watchdog function. The state is saved in FRAM