

1 Inhaltsverzeichnis

1 Inhaltsverzeichnis	1
2 Allgemeine Spezifikation	2
Typografische Konventionen	2
2.1 Kommunikationsprinzip	2
2.2 Adressierung	2
3 Protokollrahmen	3
3.1 Allgemeine Bedingungen	3
3.2 Adressierung	3
3.3 Fehlermeldungen (Exception Response)	3
4 Detaillierte Beschreibung der Protokollfunktionen	4
4.1 Liste der Befehle	4
4.1.1 Read n Bits (Function Code 01 _{DEZ})	4
4.1.2 Read n Words (Function Code 03 _{DEZ})	6
4.1.3 Write 1 Bit (Function Code 05 _{DEZ})	7
4.1.4 Write 1 Word (Function Code 06 _{DEZ})	7
4.1.5 Loopback Test (Function Code 08 _{DEZ})	8
4.1.6 Write n Bits (Function Code 15 _{DEZ})	9
4.1.7 Write n Words (Function Code 16 _{DEZ})	10
4.1.8 Report Slave-ID (Function Code 17 _{DEZ})	12
5 Anhang	14
5.1 Versionshistorie	14

2 Allgemeine Spezifikation

Das Modbus/TCP-Protokoll steht für folgende PSG Regler zur Verfügung

Mehrkanalregler
flexotemp ® MCU 128
flexotemp ® PCU 128
flexotemp ® PCU 048
flexotemp ® PCU 024

Heißkanalregler
profiTEMP

Das Modbus-Protokoll ist für eine Client/Server Anwendung konzipiert. Damit werden sämtliche Konfigurationsparameter und auch der Zustand der Regelzonen für den Benutzer transparent gemacht.

Typografische Konventionen

In diesem Handbuch finden sich Symbole und Konventionen, die Ihnen zur schnelleren Orientierung dienen.



Achtung

Mit diesem Symbol werden Hinweise und Informationen angezeigt, die entscheidend für den Betrieb des Gerätes sind.

Bei Nichtbefolgen oder ungenauem Befolgen kann es zu Schäden am Gerät oder zu Personenschäden kommen.



Hinweis

Das Symbol weist auf zusätzliche Informationen und Erklärungen hin, die zum besseren Verständnis dienen.



Beispiel

Bei dem Symbol wird eine Funktion anhand eines Beispiels erläutert.



Verweis

Bei diesem Symbol wird auf Informationen in einem anderen Dokument verwiesen.



FAQ

Hier werden FAQ (frequently asked questions) beantwortet.

Gleichungen

Berechnungsvorschriften und Berechnungsbeispiele werden so dargestellt.

2.1 Kommunikationsprinzip

Die Kommunikation erfolgt nach dem Client/Server Modell. Ein Client kommuniziert mit einem oder mehreren Servern. Der Server wird nur dann aktiv, wenn er vom Client angesprochen wird.

Modbus/TCP nutzt Standard Ethernet mit TCP/IP. Als Transportprotokoll wird TCP genutzt. Bevor die entsprechenden Operationen/Befehle übermittelt werden können, muss eine TCP-Verbindung mit dem Server (MCU/PCU) aufgebaut werden. Für die Verbindung wird die IP-Adresse des Reglers und der Destination-Port „502“ verwendet. Auf dem Regler muss mindestens einer der verfügbaren Ports auf „502“ eingestellt sein, damit die Verbindung über das Modbus/TCP Protokoll aufgebaut werden kann.

Der Verbindungsauf- und -abbau erfolgt durch den Client. Erfolgt für 30 sec nach dem Verbindungsaufbau keine Kommunikation zwischen Client und Server, schließt der Server automatisch die Verbindung.

2.2 Adressierung

Siehe 3.2 Adressierung

3 Protokollrahmen

3.1 Allgemeine Bedingungen

Die Client/Server-Operationen werden in so genannten Modbus-PDUs transportiert. Diese sind unabhängig vom unterlagerten Kommunikationssystem. Bei Modbus/TCP wird vor der Modbus-PDU ein so genannter MBAP-Header (Modbus Application Protocol Header) platziert:

Datenteil des TCP-Segments					
MBAP-Header				Modbus-PDU	
2 Bytes	2 Bytes	2 Bytes	1 Byte	1 Byte	0...253 Bytes
Transaction Identifier (TId)	Protocol Identifier (PId)	Länge	Unit Identifier (Zone)	Function Code (FC)	Daten

Die Länge der Nutzdaten einer Modbus-PDU ist aus Kompatibilitätsgründen auf 253 Bytes begrenzt. Pro TCP-Segment ist nur eine Modbus-PDU zulässig. Der MBAP-Header enthält folgende Felder:

Element	Beschreibung
Transaction Identifier (TId)	Dient der Zuordnung zwischen Request und Response. Der Client belegt das Feld, der Server kopiert den Wert in die Antwort. Über diesen Identifier kann der Client die Antwort seine Anfrage zuordnen.
Protocol Identifier (PId)	Enthält bei Modbus/TCP immer den Wert 0.
Länge	Länge der folgenden Daten in Bytes (ab Unit Identifier)
Unit Identifier	Enthält die anzusprechende Regelzone (1-n). Der Wert 1 entspricht der 1. Regelzone auf dem Regler.
Function Code (FU)	Siehe Kapitel 4.1

Die Werte werden in der Byte Order Big Endian kodiert.

3.2 Adressierung

Im Modbus/TCP-Protokoll besitzt jede Zone eine eigene Adresse. Die erste Regelzone auf einem Regler hat immer die Adresse 1. Die Regelzone n hat immer die Adresse n. Der Regler wird über seine IP-Adresse beim Verbindungsaufbau spezifiziert. Da nur ein Byte für die Adresse zur Verfügung steht, können maximal 256 Zonen pro Regler adressiert werden. Da die Adresse 00H für Broadcast-Messages reserviert ist, reduziert sich die Anzahl der maximal über den Modbus erreichbaren Zonen auf 255.

3.3 Fehlermeldungen (Exception Response)

Es werden folgende Fehlercodes unterstützt

Fehlercode	Name	Bedeutung
01	Illegale Funktion	Funktionsnummer außerhalb des zulässigen Bereiches
02	Illegale Datenadresse	Parameter wird nicht unterstützt
03	Illegaler Datenwert	Datenwert falsch / Funktion nicht ausgeführt

4 Detaillierte Beschreibung der Protokollfunktionen

Nähere Informationen über die angesprochenen **Bit- / Worttabellen** sind den reglerspezifischen Parameter- und Objektlisten zu entnehmen.

4.1 Liste der Befehle

Es werden folgende Befehle unterstützt

Funktionscode	Bedeutung	Aktion
01 _{DEZ}	Read n Bits	Liest n Bits
03 _{DEZ}	Read n Words	Liest n Worte
05 _{DEZ}	Write 1 Bit	Setzt oder löscht 1 Bit
06 _{DEZ}	Write 1 Word	Schreibt 1 Wort
08 _{DEZ}	Loopback Test	Prüft die Kommunikation
15 _{DEZ}	Write n Bits	Löscht oder setzt n Bits
16 _{DEZ}	Write n Words	Schreibt n Worte
17 _{DEZ}	Report Slave-ID	Gibt die Versionsnummer aus

4.1.1 Read n Bits (Function Code 01_{DEZ})

Dieser Befehl ermöglicht das Lesen eines einzigen oder mehrerer Bits.

Für den Sendebefehl ergibt sich folgender allgemeiner Aufbau

Client	Tid		PId		Länge	
Bytenummer	1	2	3	4	5	6
HEX	HI	LO	HI	LO	HI	LO
Client	Zone	Funktion	Adresse des 1. Bits		Anzahl Bits	
Bytenummer	7	8	9	10	11	12
HEX		01	HI	LO	HI	LO

Dabei bezeichnet der Parameter „Adresse des 1. Bits“ den Tabellenindex eines in der **Bittabelle** aufgelisteten Bits. Der Parameter „Anzahl der Bits“ bezeichnet die Anzahl an Bits, die ab dem angegebenen Tabellenindex fortlaufend eingelesen werden sollen.

Der Regler gibt folgende Antwort zurück

Server	Tid		PId		Länge	
Bytenummer	1	2	3	4	5	6
HEX	HI	LO	HI	LO	HI	LO
Server	Zone	Funktion	Byte Count	Bit 1...8	Bit ...	Letztes Bit
Bytenummer	7	8	9	10	?	?
HEX		01				

Der Parameter „Byte-Count“ bezeichnet die Anzahl der übertragenen Datenbytes.

Die einzelnen Bits werden folgendermaßen in die einzelnen Datenbytes einmaskiert

Bit des Datenbytes Nr. ...	Adresse des Bits in der Tabelle
Bit 0 des Datenbytes 1	x
Bit 1 des Datenbytes 1	X + 1
Bit 2 des Datenbytes 1	X + 2
Bit 3 des Datenbytes 1	X + 3
Bit 4 des Datenbytes 1	X + 4
Bit 5 des Datenbytes 1	X + 5
Bit 6 des Datenbytes 1	X + 6
Bit 7 des Datenbytes 1	X + 7
Bit 0 des Datenbytes 2	X + 8
Bit 1 des Datenbytes 2	X + 9
(...)	(...)

wobei x = Adresse des 1. Bits in der Tabelle ist

Pro „Datenbyte“ werden so mit 8 Bits übertragen. Werden in einem „Datenbyte“ weniger als 8 Bits übertragen, so werden die restlichen Bits auf 0 gesetzt.



Es sollen von Kanal 6 ab dem Tabellenindex 2 die Zustände von 3 Bits gelesen werden.

Der Sendebefehl hierzu sieht folgendermaßen aus

Client	Tid		Pid		Länge	
Bytenummer	1	2	3	4	5	6
HEX	00	9E	00	00	00	06
Client	Zone	Funktion	Adresse des 1. Bits		Anzahl Bits	
Bytenummer	7	8	9	10	11	12
HEX	06	01	00	02	00	03

Antwort des Reglers

Server	Tid		Pid		Länge	
Bytenummer	1	2	3	4	5	6
HEX	00	9E	00	00	00	04
Server	Zone	Funktion	Byte Count	Bit 1...8		
Bytenummer	7	8	9	10		
HEX	06	01	01	05		

Byte-Count = 01H (es wird genau 1 Datenbyte übertragen)
 Datenfeld Bit 1-8 = 05H

Bit des Datenfeldes	Zustand des Bits	Adresse (Index) des Bits in der Tabelle
Bit 0 des Datenbytes 1	1	Tabellenindex 2
Bit 1 des Datenbytes 1	0	Tabellenindex 2 + 1
Bit 2 des Datenbytes 1	1	Tabellenindex 2 + 2
Bit 3 des Datenbytes 1	0	Tabellenindex 2 + 3
Bit 4 des Datenbytes 1	0	Tabellenindex 2 + 4
Bit 5 des Datenbytes 1	0	Tabellenindex 2 + 5
Bit 6 des Datenbytes 1	0	Tabellenindex 2 + 6
Bit 7 des Datenbytes 1	0	Tabellenindex 2 + 7

4.1.2 Read n Words (Function Code 03_{DEZ})

Dieser Befehl ermöglicht das Lesen eines einzigen Wortes oder auch mehrerer Worte.

Für den Sendebefehl ergibt sich folgender allgemeiner Aufbau

Client	Tid		Pid		Länge	
Bytenummer	1	2	3	4	5	6
HEX	HI	LO	HI	LO	HI	LO
Client	Zone	Funktion	Adresse des 1. Wortes		Anzahl Worte	
Bytenummer	7	8	9	10	11	12
HEX		03	HI	LO	HI	LO

Der Parameter „Adresse des 1. Wortes“ bezeichnet den Tabellenindex eines in der **Worttabelle** aufgelisteten Wortes und der Parameter „Anzahl der Worte“ die Anzahl an Worten, die ab dem angegebenen Tabellenindex fortlaufend eingelesen werden sollen.

Der Regler gibt folgende Antwort zurück

Server	Tid		Pid		Länge				
Bytenummer	1	2	3	4	5	6			
HEX	HI	LO	HI	LO	HI	LO			
Server	Zone	Funktion	Byte Count	Wort 1		Wort...		Letztes Wort	
Bytenummer	7	8	9	10	11	?	?	?	?
HEX		03		HI	LO	HI	LO	HI	LO

Der Parameter „Byte-Count“ bezeichnet die Anzahl der übertragenen Datenbytes, also genau der doppelte Wert der angeforderten Worte. In der Antwort des Reglers werden zunächst das High-Byte und anschließend das Low-Byte des Wortes übertragen.



Es soll von Kanal 3 ab dem Tabellenindex 2 ein Wort (SOLL = 100°C) gelesen werden.

Der Sendebefehl hierzu sieht folgendermaßen aus

Client	Tid		Pid		Länge	
Bytenummer	1	2	3	4	5	6
HEX	02	83	00	00	00	06
Client	Zone	Funktion	Adresse des 1. Wortes		Anzahl Worte	
Bytenummer	7	8	9	10	11	12
HEX	03	03	00	02	00	01

Antwort des Reglers

Server	Tid		Pid		Länge				
Bytenummer	1	2	3	4	5	6			
HEX	02	83	00	00	00	05			
Server	Zone	Funktion	Byte Count	Wort 1					
Bytenummer	7	8	9	10	11				
HEX	03	03	02	03	E8				

Byte-Count = 02H

(es wird genau 1 Datenwort übertragen)

Wort 1 = 03E8H

= 1000_{DEZ}

(1000, da der Sollwert in 0,1°C angegeben wird)

4.1.3 Write 1 Bit (Function Code 05_{DEZ})

Dieser Befehl ermöglicht das Setzen oder Löschen eines einzigen Bits.

Für den Sendebefehl ergibt sich folgender allgemeiner Aufbau

Client	Tid		PId		Länge	
Bytenummer	1	2	3	4	5	6
HEX	HI	LO	HI	LO	HI	LO
Client	Zone	Funktion	Adresse des Bits		Set / Reset Datenindex	
Bytenummer	7	8	9	10	11	12
HEX		05	HI	LO	HI	LO

Der Parameter „Adresse des Bits“ bezeichnet den Tabellenindex eines in der **Bittabelle** aufgelisteten Bits und der Parameter „Set / Reset Datenindex“ zeigt an, ob das betreffende Bit gesetzt oder gelöscht werden soll. Wird der „Set / Reset Datenindex“ mit FF00H belegt, so wird das Bit auf logisch 1 gesetzt, wird er mit 0000H belegt, so wird das Bit auf logisch 0 gesetzt.

Der Regler wiederholt als Antwort den empfangenen Befehl nach der Ausführung desselben.



Es soll von Kanal 32 das Bit FBA auf „ON“ (logisch 1) gesetzt werden. Der Tabellenindex des Bits FBA ist 6.

Der Befehl und die Antwort sind

Client	Tid		PId		Länge	
Bytenummer	1	2	3	4	5	6
HEX	03	50	00	00	00	06
Client	Zone	Funktion	Adresse des Bits		Set / Reset Datenindex	
Bytenummer	7	8	9	10	11	12
HEX	20	05	00	06	FF	00

4.1.4 Write 1 Word (Function Code 06_{DEZ})

Dieser Befehl ermöglicht einen Wert in einen Wortparameter zu schreiben.

Für den Sendebefehl ergibt sich folgender allgemeiner Aufbau

Client	Tid		PId		Länge	
Bytenummer	1	2	3	4	5	6
HEX	HI	LO	HI	LO	HI	LO
Client	Zone	Funktion	Adresse des Wortes		Datenwert	
Bytenummer	7	8	9	10	11	12
HEX		06	HI	LO	HI	LO

Der Parameter „Adresse des Wortes“ bezeichnet den Tabellenindex eines in der **Worttabelle** aufgelisteten Reglerparameters. Der „Datenwert“ enthält den Wert, auf den der Reglerparameter gesetzt werden soll. Beim Setzen des „Datenwertes“ ist auf den Wertebereich des jeweiligen Reglerparameters zu achten.

Der Regler wiederholt als Antwort den empfangenen Befehl nach der Ausführung desselben.



Es soll vom Kanal 1 der Sollwert auf 10°C gesetzt werden. Der Sollwert hat den Tabellenindex 1 und wird in 0,1°C angegeben.

Der Befehl und die Antwort sind

Client	Tid		PId		Länge	
Bytenummer	1	2	3	4	5	6
HEX	03	76	00	00	00	06
Client	Zone	Funktion	Adresse des Wortes		Datenwert	
Bytenummer	7	8	9	10	11	12
HEX	01	06	00	01	00	64

4.1.5 Loopback Test (Function Code 08_{DEZ})

Dieser Befehl ermöglicht es auf einfachste Art die Kommunikation zu testen.

Für den Sendebefehl ergibt sich folgender allgemeiner Aufbau

Client	Tid		PId		Länge	
Bytenummer	1	2	3	4	5	6
HEX	HI	LO	HI	LO	HI	LO
Client	Zone	Funktion	Diagnosecode		Daten	
Bytenummer	7	8	9	10	11	12
HEX		08	00	00	HI	LO

Über den Parameter „Diagnose-Code“ wird bestimmt, welche Daten der Regler zurücksenden soll. Vom Regler wird nur der „Diagnose Code“ 0000H unterstützt. Dieser gibt an, dass die gesendeten Daten im Datenfeld 1:1 zurückgesendet werden. Als „Daten“ dürfen beliebige Worte eingesetzt werden.

Der Regler soll die gesamte Anfrage ohne jede weitere Aktion wiederholen.



Es soll von Kanal 5 der Datenwert 1234H zurückgesendet werden.

Der Befehl und die Antwort sind

Client	Tid		PId		Länge	
Bytenummer	1	2	3	4	5	6
HEX	03	89	00	00	00	06
Client	Adresse	Funktion	Diagnosecode		Daten	
Bytenummer	7	8	9	10	11	12
HEX	05	08	00	00	12	34

4.1.6 Write n Bits (Function Code 15_{DEZ})

Dieser Befehl ermöglicht das Setzen oder Löschen mehrerer Bits. Die Bits müssen allerdings in der **Bittabelle** fortlaufend hintereinander stehen.

Für den Sendebefehl ergibt sich folgender allgemeiner Aufbau

Client	Tid		Pid				Länge		
Bytenummer	1	2	3		4		5	6	
HEX	HI	LO	HI		LO		HI	LO	
Client	Zone	Funktion	Adresse des 1. Bits		Quantity		Byte Count	Daten Byte1	Daten Byte ...
Bytenummer	7	8	9	10	11	12	13	14	?
HEX		0F	HI	LO	HI	LO			

Der Parameter „Adresse des 1. Bits“ entspricht dem Index der Bittabelle, ab dem Bits gesetzt / gelöscht werden sollen.

Der Parameter „Quantity“ zeigt die Anzahl an Bits, die ab dem oben beschriebenen Index in der Bittabelle fortlaufend gesetzt / gelöscht werden sollen.

„Byte-Count“ gibt die Anzahl der übertragenen „Datenbytes“ an.

In den „Datenbytes“ wird die Information übertragen, ob ein Bit gesetzt oder gelöscht werden soll. Steht eine logisch 0 im entsprechenden Bit eines Datenbytes, so wird das Bit gelöscht. Durch eine 1 wird es gesetzt.

Die Zuordnung der Datenbyteinformation zu den Bits in der Bittabelle erfolgt folgendermaßen

Bit des Datenbytes Nr. ...	Adresse des Bits in der Tabelle
Bit 0 des Datenbytes 1	x
Bit 1 des Datenbytes 1	X + 1
Bit 2 des Datenbytes 1	X + 2
Bit 3 des Datenbytes 1	X + 3
(...)	(...)
Bit 14 des Datenbytes 1	X + 14
Bit 15 des Datenbytes 1	X + 15
Bit 0 des Datenbytes 2	X + 16
Bit 1 des Datenbytes 2	X + 17
(...)	(...)

wobei x = Adresse des 1. Bits in der Tabelle ist

Pro „Datenbyte“ werden somit 8 Bits übertragen. Werden in einem „Datenbyte“ weniger als 8 Bits übertragen, so werden die restlichen Bits auf 0 gesetzt.

Der Regler gibt folgende Antwort zurück

Server	Tid		Pid				Länge		
Bytenummer	1	2	3		4		5	6	
HEX	HI	LO	HI		LO		HI	LO	
Server	Zone	Funktion	Adresse des 1. Bits		Quantity				
Bytenummer	7	8	9	10	11	12			
HEX		0F	HI	LO	HI	LO			

Detaillierte Beschreibung der Protokollfunktionen

Der Regler wiederholt bis auf die „Datenbytes“ die Anfrage.



Es sollen von Kanal 2 ab dem Tabellenindex 2 8 Bits unabhängig von ihrem vorherigen Zustand gelöscht werden und das 9. Bit soll gesetzt werden.

Der Sendebefehl hierzu sieht folgendermaßen aus

Client	TId		PId				Länge			
Bytenummer	1	2	3		4		5		6	
HEX	03	E8	00		00		00		09	
Client	Zone	Funktion	Adresse des 1. Bits		Quantity		Byte Count	Daten Byte1	Daten Byte 2	
Bytenummer	7	8	9	10	11	12	13	14	15	
HEX	02	0F	00	02	00	09	02	00	01	

Antwort des Reglers

Server	TId		PId				Länge			
Bytenummer	1	2	3		4		5		6	
HEX	03	E8	00		00		00		06	
Server	Zone	Funktion	Adresse des 1. Bits		Quantity					
Bytenummer	7	8	9	10	11	12				
HEX	02	0F	00	02	00	09				

4.1.7 Write n Words (Function Code 16_{DEZ})

Dieser Befehl ermöglicht das Schreiben mehrerer Worte mit unterschiedlichen Datenwerten.

Für den Sendebefehl ergibt sich folgender allgemeiner Aufbau

Client	TId		PId				Länge				
Bytenummer	1	2	3		4		5		6		
HEX	HI	LO	HI		LO		HI		LO		
Client	Zone	Funktion	Adresse des 1. Wortes		Quantity		Byte Count	Daten Wort1		Daten Wort ...	
Bytenummer	7	8	9	10	11	12	13	14	15	?	?
HEX		10	HI	LO	HI	LO		HI	LO	HI	LO

Der Parameter „Adresse des 1. Wortes“ entspricht dem Index eines in der **Worttabelle** aufgelisteten Reglerparameters.

Der Parameter „Quantity“ gibt die Anzahl an Worten, die ab dem oben beschriebenen Index in der Worttabelle geschrieben werden sollen.

Im Parameter „Byte-Count“ wird die Anzahl der übertragenen „Datenworte“ angegeben.

In den „Datenworten“ wird die Information übertragen, mit welchem Wert ein Reglerparameter jeweils geschrieben werden soll; dabei ist wieder auf den Wertebereich des jeweiligen Reglerparameters zu achten.

Der Regler gibt folgende Antwort zurück

Server	Tid		Pid		Länge	
Bytenummer	1	2	3	4	5	6
HEX	HI	LO	HI	LO	HI	LO
Server	Zone	Funktion	Adresse des 1. Wortes		Quantity	
Bytenummer	7	8	9	10	?	?
HEX		10	HI	LO	HI	LO

Der Regler wiederholt also bis auf die „Datenworte“ die Anfrage.



Es sollen auf Kanal 4 ab dem Tabellenindex 11 genau 2 Worte beschrieben werden (SPLO=10.0°C und SPHI = 200.0°C).

Der Sendebefehl hierzu sieht folgendermaßen aus

Client	Tid		Pid		Länge						
Bytenummer	1	2	3	4	5			6			
HEX	05	21	00	00	00			0B			
Client	Zone	Funktion	Adresse des 1. Wortes		Quantity		Byte Count	Daten Wort1		Daten Wort ...	
Bytenummer	7	8	9	10	11	12	13	14	15	16	17
HEX	04	10	00	0B	00	02	04	00	64	07	D0

Antwort des Reglers

Server	Tid		Pid		Länge	
Bytenummer	1	2	3	4	5	6
HEX	05	21	00	00	00	06
Server	Zone	Funktion	Adresse des 1. Wortes		Quantity	
Bytenummer	7	8	9	10	11	12
HEX	04	10	00	0B	00	02

4.1.8 Report Slave-ID (Function Code 17_{DEZ})

Mit diesem Befehl wird die Versionsnummer ausgelesen. Zusätzlich kann der Systemzustand abgefragt werden.

Für den Sendebefehl ergibt sich folgender allgemeiner Aufbau

Client	Tid		Pid		Länge	
Bytenummer	1	2	3	4	5	6
HEX	HI	LO	HI	LO	HI	LO
Client	Zone	Funktion				
Bytenummer	7	8				
HEX		11				

Der Regler gibt folgende Antwort zurück

Server	Tid		Pid		Länge						
Bytenummer	1	2	3	4	5		6				
HEX	HI	LO	HI	LO	HI		LO				
Server	Zone	Funktion	Byte Count	Kennung		1. Wort		2. Wort		3. Wort	
Bytenummer	7	8	9	10	11	12	13	14	15	16	17
HEX		11	06	10	FF	HI	LO	HI	LO	HI	LO

Der Parameter „Byte-Count“ bezeichnet die Anzahl der übertragenen Bytes. Es werden bei diesem Befehl immer 6 Datenbytes übertragen.

Im High-Byte des 1. Wortes wird bei einer PCU ‚P‘ und bei einer MCU ‚M‘ als ASCII-Wert eingetragen. Im Low-Byte des 1. Wortes steht die maximale Zonenanzahl, die auf dem Regler verfügbar ist. Bei einer PCU048 wird hier 48 eingetragen.

Im „zweiten-“ sowie im „dritten Wort“ wird die Softwareversionsnummer des Reglers übertragen.



Es soll die Versionsnummer ausgelesen werden.

Der Sendebefehl hierzu sieht folgendermaßen aus

Client	Tid		PId		Länge	
Bytenummer	1	2	3	4	5	6
HEX	05	34	00	00	00	02
Client	Zone	Funktion				
Bytenummer	7	8				
HEX	01	11				

Antwort des Reglers

Server	Tid		PId		Länge						
Bytenummer	1	2	3	4	5	6					
HEX	05	34	00	00	00	0B					
Server	Zone	Funktion	Byte Count	Kennung		1. Wort		2. Wort		3. Wort	
Bytenummer	7	8	9	10	11	12	13	14	15	16	17
HEX	01	11	08	10	FF	50	80	01	2B	09	0A
						(,P')	(128)	(01)	(43)	(09)	(A)

5 Anhang

5.1 Versionshistorie

Version	Datum	Änderung
1.00.02	06.03.2014	Heißkanalregler profiTEMP ergänzt
1.00.01	29.01.2010	Zeichenverschiebungen (pdf) korrigiert
1.00.00	23.11.2009	Erstveröffentlichung