

Funktion	CMD + Daten (->RS485)	Antwort (<- RS485)	
WriteI2C	DS + '77' + SA + 'XXYYZZ....' + CS +<CR>	'77' + DS + SA + '01' + CS +<CR>	Daten geschrieben
		'77' + DS + SA + '00' + CS +<CR>	Slave nicht gefunden
ReadI2C	DS + '72' + SA + Cnt + CS +<CR>	'72' + DS + SA + '01' + CS +<CR>	Slave gefunden
		'64' + DS + SA + Cnt + 'XXYYZZ....' + CS +<CR>	Daten gelesen
		'72' + DS + SA + '00' + CS +<CR>	Slave nicht gefunden
CheckSlvAdr	DS + '63' + SA + CS <CR>	'63' + DS + SA + '01' + CS +<CR>	Slave gefunden
		'63' + DS + SA + '00' + CS +<CR>	Slave nicht gefunden
SetSCLFreq	DS + '65' + IH + IL + CS +<CR>	'65' + DS + IH + IL + CS +<CR>	SCL-Freq. geschrieben
GetSCLFreq	DS + '69' + CS +<CR>	'69' + DS + IH + IL + CS +<CR>	SCL-Freq. gelesen
ReSetIO1	DS + '6D01' + CS +<CR>	'6D' + DS + '01' + CS +<CR>	IO-Pin 1 HIGH
	DS + '6D00' + CS +<CR>	'6D' + DS + '00' + CS +<CR>	IO-Pin 1 LOW
ReSetIO2	DS + '6E01' + CS +<CR>	'6E' + DS + '01' + CS +<CR>	IO-Pin 2 HIGH
	DS + '6E00' + CS +<CR>	'6E' + DS + '00' + CS +<CR>	IO-Pin 2 LOW
GetIOState	DS + '6F' + CS +<CR>	'6F' + DS + 00 + CS +<CR>	IO1 LOW – IO2 LOW
		'6F' + DS + 01 + CS +<CR>	IO1 HIGH – IO2 LOW
		'6F' + DS + 02 + CS +<CR>	IO1 LOW – IO2 HIGH
		'6F' + DS + 03 + CS +<CR>	IO1 HIGH – IO2 HIGH

Weitere Rückmeldungen	Beschreibung	
ChkSumERROR	'73' + DS + '01' + CS +<CR>	Checksum Error
UnCMD	'FF' + DS + '00' + CS +<CR>	Befehl unbekannt

DS: DIPSwitch, Adapter Adresse

SA: Slave Adresse.

CS: Prüfsumme (Checksum)

CS = 0x0100 – (Summe MOD 0x0100); Dabei ist Summe, die Summe aller zu sendenden Bytes ohne CS und CR.

Cnt: Anzahl der zu lesenden Bytes (s. ReadI2C)

IH: SCL-Frequenz - High

IL: SCL-Frequenz - Low

$f_{SCL} = 12.000.000 / (2 * (IL + IH))$; wenn **IL** und **IH** gleich sind, dann beträgt das Tastverhältnis 50%. (400kHz Adapter)

XXYYZZ...: sind die zu übertragene I2C – Daten.

<CR> : Alle gesendeten bzw. empfangenen Daten werden immer mit einem CarriageReturn (**<CR>** = 0x0D HEX) abgeschlossen.

XXYYZZ... sind die zu übertragene Daten. Es dürfen mindesten 1 Byte und maximal 128 Bytes übertragen werden.

Beispiel: es sollen die 5 Bytes 0xA1, 0x1F, 0x22, 0x5C, 0xB0 an die Adapter Adresse 0xFE an die Slave Adresse 0xC4 gesendet werden, dann wird folgende Zeichenkette abgeschlossen mit einem CarriageReturn über die Serielle

Schnittstelle gesendet: **'FE77C4A11F225CB0DB' + <CR>**

'FE' → 0xFE; Adapter Adresse, DIP-Switch

'77' → 0x77; Command / WRITE

'C4' → 0xC4; Slave Adresse

'A11F225CB0' → Daten: 0xA1, 0x1F, 0x22, 0x5C, 0xB0

'59' → 0x59; Checksumme aus **'FE77C4A11F225CB0'**

Summe = 0x46+0x45+0x37+0x37+0x43+0x34+0x41+0x31+0x31+0x46+0x32+0x32+0x35+0x43+0x42+0x30 = 0x03A7

(F) (E) (7) (7) (C) (4) (A) (1) (1) (F) (2) (2) (5) (C) (B) (0)

CS = 0x0100 – (Summe MOD 0x0100) = 0x0100 – (0x03A7 MOD 0x0100) = 0x0100 – 0xA7 = 0x59

<CR> → 0x0D; CarriageReturn

Die Rückmeldung vom Adapter kann folgendes sein:

'77FEC4012F' + <CR> oder **'77FEC40030' + <CR>**

Die empfangenen Daten werden als Zeichenkette in Hexdezimaler Form empfangen:

'77' → 0x77; Command

'FE' → 0xFE; Adapter Adresse ; DIP-Switch

'C4' → 0xC4; Slave Adresse

'01' → 0x01; Slave wurde gefunden, Datenübertragung erfolgreich.

'2F' → 0x2F; Checksumme aus **'FE77C401'**

<CR> → 0x0D; CarriageReturn

oder

'77' → 0x77; Command

'FE' → 0xFE; Adapter Adresse ; DIP-Switch

'C4' → 0xC4; Slave Adresse

'00' → 0x01; Slave nicht angeschlossen, Datenübertragung fehlgeschlagen.

'30' → 0x30; Checksumme aus '**FE77C400**'

<CR> → 0x0D; CarriageReturn

RS232 – Einstellungen:

Baud: 19200

DataBits: 8

StopBits: 1

Parity: None