



Befehlsbeschreibung

zum

Digitalen Funktionsgenerator 13654.99

Aktuelle Version 03.00 SP

04.06.14 / 04.06.2014 09:43:00 / REV. 03.00

Inhaltsverzeichnis

SCHNITTSTELLENPARAMETER	3
BENÖTIGTE PARAMETER/MESSWERTE/MODES	3
KOMMUNIKATION	5
KOMMUNIKATIONSRAHMEN.....	5
<i>Generelles Rahmenformat der seriellen Schnittstelle</i>	5
RAHMENARTEN.....	5
BEISPIELE DER KOMMUNIKATION	6
<i>Lesen und schreiben der Konfiguration (Grundgerät)</i>	6
<i>Lesen und schreiben der Konfiguration (Sensorfunktion)</i>	6
<i>Lesen und der SU Konfiguration (virtuelle Sensor)</i>	7
<i>Nutzdaten von Datenrahmen im IDLE Modus</i>	7
<i>Nutzdaten von Datenrahmen im Messmodus (Paketrahmenantwort)</i>	8
<i>Start einer Messung</i>	8
<i>Messung Stopp</i>	9
FIRMWAREUPDATE	9
<i>Geräteklassen</i>	9
<i>Kommandos zur Steuerung eines Firmwareupdates</i>	9
<i>Hinweise zum Ablauf und der Implementierung der Steuerung der PC-Software</i>	12
<i>CRC-32 Berechnung</i>	13
BEISPIELKOMMUNIKATIONEN FÜR FUNKTIONSGENERATOR	15
PC-MODE EIN	15
FREQUENZWERT SETZEN:	15
FREQUENZWERT LESEN:	15

Schnittstellenparameter

Zur Kommunikation zwischen dem Funktionsgenerator und einem PC wird der serielle UART FT232RL der Fa. FTDI benutzt.

Vendor-ID = 0x0304 (Fa. FTDI)

Product-ID = 0xA303

Die Übertragungsparameter sind wie folgt festgelegt:

921600 Bd

8 Datenbits

1 Stopbit

Keine Parität

Kein Handshake

Benötigte Parameter/Messwerte/Modes

Zur Kommunikation mit dem Funktionsgenerator muss folgende Kommunikation möglich sein:

In der Version 3 besitzt der Digitale Funktionsgenerator zusätzlich zur Möglichkeit der Einstellung noch einen Messkanal (Frequenz/Hz). D.h. das Gerät verhält sich wie ein Sensor.

Sensor-ID = 0xFE;

Modes:

PC-Mode ein/ausschalten (Anzeige „PC remote“ im Display des Gerätes signalisiert den PC-Mode)

ACHTUNG: Damit das Gerät im Remote-Modus bleibt muss der Parameters „Hardware-Version“ der Grundparameter oder eine Datenabfrage Typ B (0x0B) innerhalb von 3s erfolgen.

Parameter Grundgerät (Geberfunktionsbetrieb):

Index	Bezeichnung	Anzahl Bytes	Minimum	Maximum	Default
0x00	Hardware-Version	1	0x00	0xFF	
0x01	Firmware-Version	2	0x0000	0xFFFF	0x0100= 1.00
0x02	Geräteklasse	1			0x0A
0x03	Mode (0 = Leistungsausgang, 1= Kopfhörer)	1	0x00	0x01	0x00
0x04	Signalform (1=Sin,2=Dreieck,3=Rechteck, 4=Frequenzrampe, 5=Spannungsrampe)	1			
0x05	Frequenzwert (Hz*10)	4	0	9999990	2000 = 200,0Hz
0x06	Amplitude (mV)	2			
0x07	Offset (mV)	2			
0x08	Startfrequenz f1 (Hz*10)	4			
0x09	Stopfrequenz f2 (Hz*10)	4			
0x0A	Frequenzpause bei Rampe (ms)	4	1	9999	
0x0B	Schrittweite Frequenz (Hz*10)	4			
0x0C	Signalform Frequenzrampe (0=sin,1=dreieck, 2=rechteck, 3= F-Rampe, 4=U-Rampe)	1	0	4	
0x0D	Startspannung U1 (mV)	2			
0x0E	Stoppspannung U2 (mV)	2			
0x0F	Spannungspause bei Rampe (ms)	4	1	9999	
0x10	Schrittweite Spannung (mV)	2			
0x11	Gesamtdauer Frequenzrampe (ms)	4			
0x12	Gesamtdauer Spannungsrampe (ms)	4			
0x13	Frequenzrampe zyklisch=1 , einfach=0 abarbeiten	1	0	1	0
0x14	Spannungsrampe zyklisch, einfach abarbeiten	1	0	1	0
0x15	Lineare/logarithmische Frequenzrampe	1	0	1	0=linear,1=log
0x16	Sweepfaktor (wert*1000)	2	0	1250	0.000..1,250

Index	Bezeichnung	Anzahl Bytes	Minimum	Maximum	Default
0x17	U~f Ausgangsskalierung 0= 0..1Mhz 1=0..100kHz 2 = 0..10kHz, 3= 0..1Khz, 4= 0..100Hz 5= f1..f2	1	0	5	0
0x18	Amplitude - Frequenzrampe (mV)	2			
0x19	Offset - Frequenzrampe (mV)	2			
0x20	Offset - Offsetfaktor	2	-16000	16000	-23
0x21	Offset - Verstärkungsfaktor	2	0	200	835 = 0.835
0x22	Amplitude -Offsetfaktor	2	-16000	16000	2180
0x23	Amplitude – Verstärkungsfaktor 0	2	0	2000	850 = 0.850
0x24	Amplitude – Verstärkungsfaktor 1	2	0	2000	800 = 0.800

Parameter Sensorfunktion (Sensorfunktionsbetrieb):

Index	Bezeichnung	Anzahl Bytes	Minimum	Maximum	Default
0x00	Hardware-Version	2	0x00	0xFFFF	
0x01	Firmware-Version	2	0x0000	0xFFFF	0x0200= 2.00
0x02	Geräteklasse	2			0x000A
0x03	Extended Adresse	8			Not used
0x04	Kurzadresse (*)	2	0x0001	0xFFFD	0x0001
0x05	Aktuelles Netzwerk	2	0x0000	0xFFFE	0x0000
0x08	Samplerate IDLE Modus	2			Not used
0x09	Samplerate Measure Modus	4			
0x14	Zeitbasis	1	0	4	0 = ns, 1=us, 2=ms, 3=s, 4=min
0x1A	Paketgröße	1	1	50	
0x1C	Bitmaske der zu erfassenden Messgrößen	1	Bitmaske	Bitmaske	0xFF

Parameter 0x01 bis 0x02 besitzen nur Lesestatus, alle anderen Parameter besitzen Lese- und Schreibstatus. Die Firmwareversion wird durch aufspielen eines neuen Updates verändert. Die Hardware-Version muss bei der Kalibrierung eingegeben werden (wird ebenfalls im EEPROM abgelegt).

*Standardmäßig wird die Adresse nicht auf dem Display dargestellt. Sobald ein Wert PC-Seitig übermittelt wird, wird dieser auf dem Display dargestellt.

Parameter Sensoreinheit (virtuelle SU)

Index	Bezeichnung	Anzahl Bytes	Minimum	Maximum
0x00	EEPROM bereits programmiert	1	255	
0x01	Sensor-ID	1	0,..., 255	
0x02	Seriennummer	4	ASCII	
0x03	Prüfsumme (nicht verwendet)	2		
0x04	Hersteller-ID	1	0	255
0x05	Max. Datenrate	4		
0x06	Typ. Datenrate	4		
0x07	Bitmaske der verfügbaren Sensorkanäle	1	Bitmaske	Bitmaske
0x08	Bitmaske der Burstkanäle	1	Bitmaske	Bitmaske
0x09	Kalibrierung erlaubt 1= 1Punkt, 2= 2Punkt, 0 = keine	1	0	2
0x0a	Sampleunit 0=ns, 1=µs, 2=ms, 3= s, 4=h	1	0	4
0x0b	Revisionsnummer	1		
0x0c	Warm Up Zeit (100ms-Schritte) 1= 1x100ms, 2=2+100ms	1		
0x0d	Gleichungstyp 1= linear (m*B0 +A0)	1		
0x0e	Anz. Kalibrierseiten	1		
0x0f	Aktuelle Kalibrierseite	1		

Kommunikationsrahmen

Generelles Rahmenformat der seriellen Schnittstelle

Die Kommunikationsrahmen sind wie folgt aufgebaut:

Start	Länge	Art	Adresse	Daten	Stopp
0x7D	1 Byte	1 Byte	2 Bytes	x Bytes	0x7E

Abbildung 1 Rahmenformat der seriellen Kommunikation

- Der Rahmenstart ist durch das Byte vom Wert 0x7D gekennzeichnet.
- Längensymbol, 1 Byte. Start, Länge und Stopp – Zeichen werden nicht mitgezählt, das entspricht also der gesamten Rahmenlänge – 3.
- Die Rahmenart entsprechend Tabelle 1.
- Adresse, 2 Bytes. Bei ausgehenden Rahmen (PC -> Gerät) wird die Kurzadresse des Geräts angegeben, für das der Rahmen bestimmt ist. Bei empfangenen Rahmen entspricht die Adresse der Quelladresse. Hinweis: die Übertragung ist Little – Endian, das entspricht der Least Significant Byte First Ordnung. Beispiel: Ein Rahmen eines Geräts mit Adresse 0x0001 wird als 0x0100 übertragen.
DER FUNKTIONSGENERATOR BESITZT DIE ADRESSE =0x100 FÜR PARAMETER DES GRUNDGERÄTES UND 0x101 FÜR PARAMETER DER SENSORFUNKTION.
- Die Nutzdaten richten sich nach der Rahmenart und sind in den entsprechenden Kapiteln beschrieben
- Rahmenstopp (0x7E)

ACHTUNG: die Übertragung ist Little – Endian, das entspricht der Least Significant Byte First Ordnung. Beispiel: Ein Rahmen eines Geräts mit Adresse 0x0100 wird als 0x0001 übertragen.

Rahmenarten

Die in der nachfolgenden Tabelle aufgeführten Rahmenarten sind definiert. Jeder Rahmen hat einen Rahmenindex, über den die Art der Daten entschieden wird.

Die Liste der Rahmenarten kann in Absprache mit der Phywe entsprechend der Erfordernissen der Geräte erweitert werden.

Rahmen-index	Beschreibung
0x01	Anfrage Daten
0x0A	Daten Paket Typ A (Default für Pakete im E_MODE_MEASURE)
0x0B	Daten Paket Typ B (Default für Pakete im E_MODE_IDLE_DATA1)
0x11	Konfiguration schreiben
0x12	Konfiguration lesen
0x18	Konfiguration OK
0x19	Konfiguration Fehler
0x1A	Konfiguration Antwort (Wert)
0x1B	Kommando OK
0x1F	Anfrage/Antwort Liste angemeldeter Wireless Geräte
0x21	Firmware: Start Übertragung neuer Firmware
0x22	Firmware: neue Daten schreiben
0x23	Firmware: Abschluss mit Checksumme
0x24	Firmware: Nächster Teil der Firmware anfordern
0x28	Firmware OK
0x29	Firmware Fehler
0x31	Start Messung
0x32	Stop Messung
0x42	Konfiguration Virtuelle SU lesen
0x4A	Konfiguration Virtuelle SU Antwort (Wert)

Rahmen-index	Beschreibung
0x4D	PC-Mode ein
0x4E	PC-Mode aus
0x4F	Rampe ein/aus bzw. Werte übergeben
0x50	Kalibrierdaten in EEPROM schreiben
0xFF	Fehler in der Kommunikation

Tabelle 1 Rahmenarten

Beispiele der Kommunikation

In den folgenden Unterkapiteln sind einige Beispiele der Kommunikation dargestellt. Die dargestellten Kommunikationsrahmen beziehen sich immer auf Rahmen, die auf der seriellen Schnittstelle übertragen werden. Zur besseren Übersichtlichkeit wurde auf die Umkehrung der Adressen von Little Endian verzichtet

Lesen und schreiben der Konfiguration (Grundgerät)

- Jedes Gerät verfügt über Zwei Konfigurationsparameter, die zur Laufzeit gelesen und beschrieben werden können.
- Die Geräte werden über die 2-Byte Kurzadresse unterschieden. Geräte die nur einzeln verwendet werden bekommen die Kurzadresse 0x0100.
- Ein Telegramm mit Konfigurationsdaten kann immer ein oder mehrere Werte enthalten. Der erste 8b Wert repräsentiert den Eintrag der Konfiguration, danach folgen die Konfigurations-Werte. Die Konfigurations-Werte müssen genau die Anzahl Bytes enthalten, die für das Gerät festgelegt wurden.
- Ein Fehler wird zurückgegeben, wenn
 - einer der angeforderten Parameter nicht verfügbar ist
- Wird Paketart 0xFF zurückgegeben, ist ein globaler Fehler aufgetreten.

Wurden alle Parameter korrekt geschrieben wird OK zurückgegeben

Beispiel 1 - Anfrage des Parameter 0x01 (Firmwareversion):

Start	Länge	Art	Adresse	Param.	Stopp
0x7D	0x04	0x12	0x0100	0x01	0x7E

Antwort: (angefragten Parameter hat 2 Byte -Firmwareversion = 2.10)

Start	Länge	Art	Adresse	Param	Daten	Stopp
0x7D	0x07	0x1A	0x0100	0x01	0x02 0x0A	0x7E

Beispiel 2 - Schreiben von Konfigurationsdaten (hier ein Frequenzwert = 0x0105)

Start	Länge	Art	Adresse	ID	Daten	Stopp
0x7D	0x06	0x11	0x0100	0x05	0x05 0x01	0x7E

Antwort OK:

Start	Länge	Art	Adresse	Stopp
0x7D	0x03	0x18	0x0100	0x7E

Antwort im Fehlerfall:

Start	Länge	Art	Adresse	Stopp
0x7D	0x03	0x19	0x0100	0x7E

Lesen und schreiben der Konfiguration (Sensorfunktion)

Beispiel - Anfrage des Parameter 0x1A (Paketgröße):

Start	Länge	Art	Adresse	Param.	Stopp
0x7D	0x04	0x12	0x0101	0x1A	0x7E

Antwort: (hier Paketgröße = 5)

Start	Länge	Art	Adresse	Param.	Daten	Stopp
0x7D	0x05	0x1A	0x0101	0x1A	0x05	0x7E

Lesen und der SU Konfiguration (virtuelle Sensor)

Beispiel - Anfrage des Parameter 0x01 (Sensor-ID):

Start	Länge	Art	Adresse	Param.	Stopp
0x7D	0x04	0x42	0x0101	0x01	0x7E

Antwort: (hier Sensor-ID = FE -> Funktionsgenerator)

Start	Länge	Art	Adresse	Param.	Daten	Stopp
0x7D	0x05	0x4A	0x0101	0x01	0xFE	0x7E

Nutzdaten von Datenrahmen im IDLE Modus

Anfrage:

Start	Länge	Art	Adresse	Stopp
0x7D	0x03	0x01	0x0100	0x7E

Antwort:

Start	Länge	Art	Adresse	Messkanalgröße	Bitmaske Messkanalgröße	Messkanal Ch.1 24 Bit	Stopp
0x7D	0x0F	0x0B	0x0100	0x02	0x01	0x000000	0x7E

Messkanalgröße	
Wert (8-Bit)	Größe der Messwerte in Bit
0x00	8
0x01	16
0x02	24
0x03	32
0x04	40
0x05	48
0x06	56
0x07	64

Messgrößen/ Messkanal	Bitmaske der zu erfassenden Messgrößen
Ch. 1	Bit 0 (0x01)
Ch. 2	Bit 1 (0x02)
Ch. 3	Bit 2 (0x04)
Ch. 4	Bit 3 (0x08)

Nutzdaten von Datenrahmen im Messmodus (Paketrahmenantwort)

- Im Messmodus werden die Daten zum PC hin so gesendet, wie sie im internen Speicher Funktionsgenerators sind.
- Ist eine Paketrahmenantwort durch den Funktionsgenerator zum PC gesendet, folgt eine weitere Paketrahmenantwort mit einem um die Sampleanzahl inkrementierten Zeitstempel.
- Ist die Funktionsart „Frequenzrampe“ gewählt. Wird mit „Messung Start“ auch automatisch die Frequenzrampe gestartet. Ist die Rampe durchfahren, wird die Messung automatisch gestoppt, und das Kommando „Messung Stopp“ (0x32) zum PC geschickt.
- Durch senden des Signals „Messung Stopp“ kann die Rampe jederzeit beendet werden.

Bei eingestellter Paketgröße von 10

Start	Länge	Art	Adresse	Messkanal- größe	Bitmaske Mess- kanalgröße	Zeitstempel	Ch.1 24Bit	...	Ch.1 24Bit	Stopp
0x7D	0x47	0x0A	0x0100	0x02	0x01	0x00 00 00 00				0x7E

Nachfolgende Übertragung:

Start	Länge	Art	Adresse	Messkanal- größe	Bitmaske Mess- kanalgröße	Zeitstempel	Ch.1 24Bit	...	Ch.1 24Bit	Stopp
0x7D	0x47	0x0A	0x0100	0x02	0x01	0x09 00 00 00				0x7E

Start einer Messung

- Die Messung wird mit dem Kommando 0x31 und der Zieladresse des Funktionsgenerators gestartet.
- Auf das Kommando folgt sofort eine Bestätigung vom Typ 0x38, um zu signalisieren, dass der Funktionsgenerator den Messmodus gestartet hat.
- Ist die Funktionsart „Frequenzrampe“ gewählt. Wird mit „Messung Start“ auch automatisch die Frequenzrampe gestartet.
- Ist der Start des Messmodus nicht möglich, wird ein Fehler vom Typ 0x39 an den PC geschickt. Mögliche Fehlerfälle sind
 - Der Funktionsgenerator ist nicht im IDLE-Modus
 -
- Beispiel: Messung Start mit positiver Bestätigung

PC -> Funktionsgenerator

Start	Länge	Art	Adresse	Stopp
0x7D	0x3	0x31	0x0100	0x7E

Funktionsgenerator-> PC

Start	Länge	Art	Adresse	Stopp
0x7D	0x3	0x38	0x0100	0x7E

Fehlerfall:

Start	Länge	Art	Adresse	Fehlercode 16 Bit	Stopp
0x7D	0x3	0x38	0x0100	0xabcd	0x7E

Messung Stopp

- Bei „Messung Stopp“ wird immer das Kommando „Bestätigung auf Start oder Stopp“ (0x38) durch den Funktionsgenerator an die PC-Applikation gesendet
- Eine Messung wird von der PC-Applikation mit dem Kommando „Messung Stopp“ (0x32) gestoppt.
- Eine Antwort durch den Funktionsgenerator wird erst dann an den PC gesendet, wenn der Funktionsgenerator die verbleibenden Daten aus dem internen Messwertspeicher übertragen hat.
- Als Antwort wird ein Rahmen des Typs 0x38 mit der Funktionsgenerator-Adresse (0x0100) gesendet.

C4- Xpert -Link -> PC

Start	Länge	Art	Adresse	Stopp
0x7D	0x03	0x32	0x0100	0x7E

Firmwareupdate

Geräteklassen

- Beim Upload einer Firmware muss die Geräteklasse angegeben werden, um sicherzustellen, dass das richtige Firmware-Image passend zum Gerät übertragen wird.
- Folgende Geräteklassen sind definiert:

Gerät	Geräteklasse
Funktionsgenerator	0x0A
Röntgengerät	0x0B
C4 Junior Link	0x0C

Tabelle 2 Geräteklassen

Kommandos zur Steuerung eines Firmwareupdates

0x28: Firmware OK

- Diese Rahmenart wird verwendet, um eine Bestätigung des eines vorangegangenen Kommandos zu signalisieren.
- Unterschiedliche Parameter können vorkommen; das richtet sich nach dem vorangehenden Kommando.

0x29: Firmware Fehler

- Kann ein Kommando nicht ausgeführt werden, oder liegt ein Fehler vor, wird ein Rahmen mit der ID 0x29 gesendet.
- Ein 8-Bit Fehlercode gibt Aufschluss über die Ursache. Folgende Codes sind definiert:

Fehlercode	Beschreibung	Betrifft Kommandos
0x01	Ungültige Geräte-ID	0x21
0x02	Ungültige Hardware-Version	0x21
0x03	Firmware-Image zu groß	0x21
0x04	Ungültige Sequenznummer, die letzte gültige Sequenznummer wird nach dem Fehlercode übertragen.	0x22
0x05	Checksummenprüfung nach Übertragung fehlgeschlagen.	0x23

Tabelle 3 Firmware Update Fehlercodes

0x21: Start Übertragen neuer Firmware von PC zum Gerät

Folgende Parameter müssen dem Paket 0x21 mitgegeben werden:

Parameter	Länge	Beschreibung
1	2	Firmware-Version
2	1	Firmware Image Nummer.
3	1	Gesamte Anzahl Images dieser Firmware
4	4	Länge des Firmware-Image in Bytes. Falls das Firmware-Image in mehrere Teile unterteilt ist, dann bezieht sich die Länge nur auf den zu übertragene Teil.
5	2	Gerätetyp. Siehe dazu Tabelle 2 auf Seite 9.
6	1	Länge der Hardwarekompatibilitätsliste in Bytes
7	1...x	Hardwarekompatibilitätsliste: Falls eine Firmware für mehrere Hardwarerevisionen gilt, dann werden diese hier aneinander gereiht.

Beispiel:

- Firmware Version 0x0124 = 01.36
- Teil 1 von 2
- Länge des Teil-Images: 10kBytes
- Firmware für Geräte-Typ: 0x001
- Hardware-Revisionen: 0x001

		Art	Adresse		Version		Teil 1/2		Länge				Ger. Klasse		Länge HW Liste		Liste HW		
7D	10	21	00	01	24	01	01	02	00	28	00	00	07	00	02	01	00	7E	

Antwort: OK

		Art	Adresse		
7D	03	28	00	01	0x7E

Antwort Fehler, falsche Geräte-ID:

		Art	Adresse		Fehlercode	
7D	03	29	00	01	01	7E

0x22 Neue Daten schreiben

- Das Firmware-Image wird paketweise an das Gerät gesendet
- Jedes Paket verfügt über eine Kennzeichnung des Abschnitts. Der Abschnitt wird durch einen Zähler des ersten Bytes innerhalb des Firmware-Image bestimmt. Das erste Byte des Firmware-Image hat die Abschnittsnummer 0, das letzte Byte die Länge des Firmware-Image – 1.
- Die Abschnittsnummer bezieht sich auf das erste Byte im aktuell übertragenen Paket.
- Das Gerät bestätigt jedes empfangene Paket mit der übertragenen Abschnittsnummer.
- Die Abschnitte müssen der Reihenfolge nach übertragen werden. Falls die Reihenfolge nicht eingehalten wird, wird der Fehlercode 0x04 gefolgt von der zuletzt empfangenen Abschnittsnummer zurückgegeben.
- Die Parameter sind wie folgt:

Parameter	Länge	Beschreibung
1	4	Abschnittsnummer, Zähler des ersten Bytes in dem Firmware-Paket.
2	beliebig	Firmware-Daten

- Ablauf:

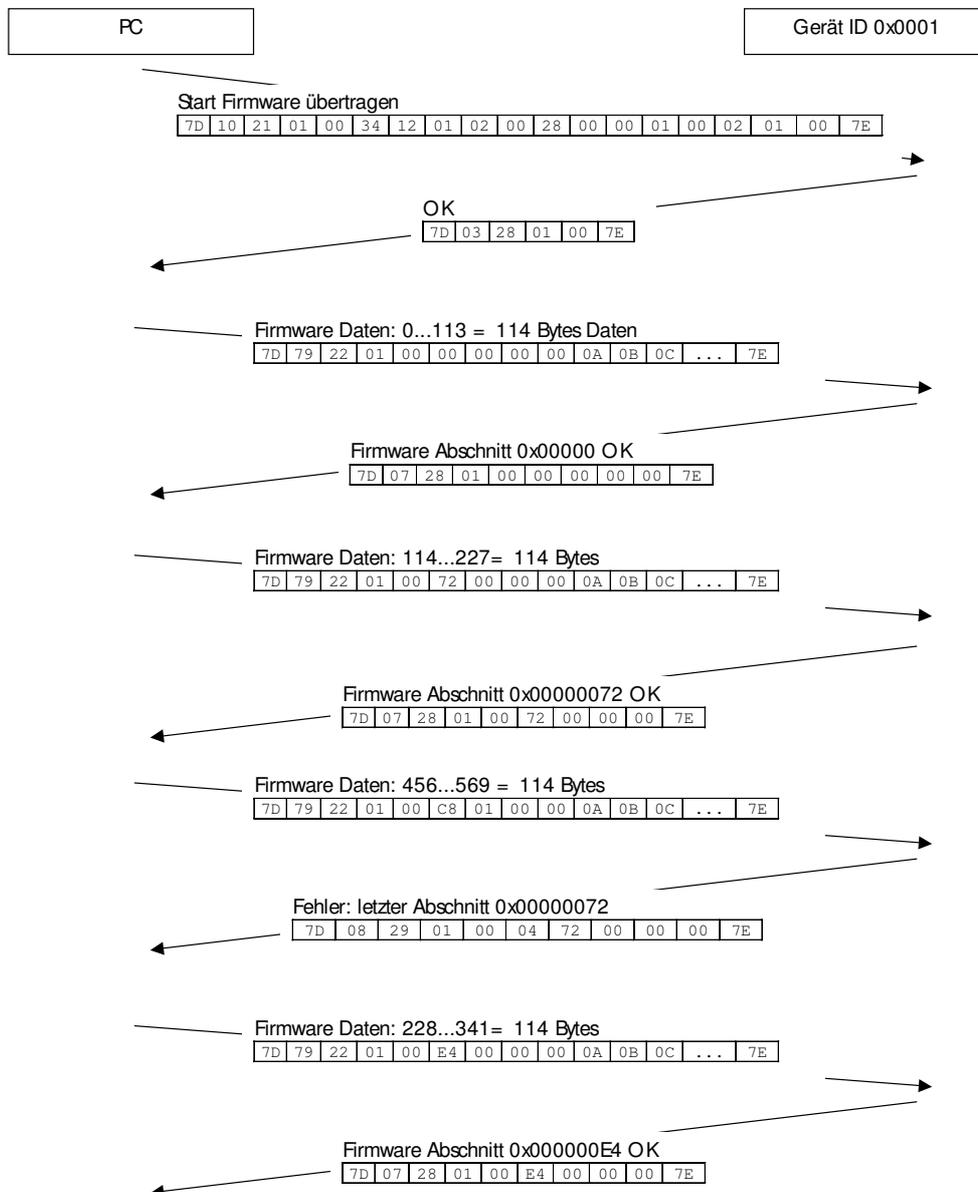


Abbildung 2 Ablauf Firmware Daten senden

0x23: Firmware-Upload abschließen

- Der Firmware-Upload wird abgeschlossen, indem die CRC-32-Prüfsumme des Firmware-Images übertragen wird.

Parameter	Länge	Beschreibung
1	4	CRC-32 Checksumme Firmware Image

Beispiel: 10kB Firmware-Image

		Art	Adresse	CRC-32 Checksumme	
7D	07	23	00 01	00 28 00 AB	7E

Antwort OK:

		Art	Adresse	
7D	03	28	00 01	0x7E

Antwort Checksumme ungültig:

		Art	Adresse	Fehlercode	
7D	04	29	00 01	05	7E

0x24 Nächsten Teil des Firmware-Image anfordern

- Wird das Firmware-Image nicht in einem ganzen Teil übertragen, so wird automatisch vom Gerät nach erfolgreichem Update von einem Teil ein weiterer Teil angefragt.
- Hintergrund dabei ist, dass nach einer Übertragung eines Image-Teilstücks ein Neustart des Geräts erfolgt.
- Die Parameter sind wie folgt:

Parameter	Länge	Beschreibung
1	2	Firmware Version
2	1	Nächster zu übertragener Firmware-Image Teil
3	1	Gesamte Anzahl Image Teile

Beispiel: Upload von FW Version 0x1234 Teil 1/2 erfolgreich, das Gerät sendet nach dem Neustart die folgende Meldung zum PC, um Teil 2/2 anzufordern:

		Art	Adresse	Version	Teil 2/2
7D	10	24	00 01	34 12	02 02

Hinweise zum Ablauf und der Implementierung der Steuerung der PC-Software

Generierung der Firmware-Beschreibung

- Um den Firmware-Upload zu starten, ist ein Paket vom Typ 0x21 erforderlich. Die Daten müssen bei der Erstellung der Firmware festgelegt werden:
 - Firmware-Image Nummer und Anzahl Firmware-Images: sind beim Universalzähler immer 1, da die Firmware nicht unterteilt ist.
 - Länge des Firmware-Image in Bytes: die exakte Größe der *.bin – Datei mit dem Firmware-Image in Bytes.
 - Der Gerätetyp ist beim Lichtgeschwindigkeitsmessgerät 0x07.
 - Hardwarekompatibilitätsliste: diese wird ab dem Zeitpunkt wichtig, zu dem verschiedene Hardware-Revisionen vorhanden sind. Vermutlich wird neuere Firmware abwärtskompatibel gehalten, somit auf mehrere Revisionen anwendbar sein. Durch diese Liste kann aber auch eine bestimmte Revision ausgeschlossen werden. Somit wird die Möglichkeit unterbunden, eine inkompatible Firmware auf ein Gerät zu laden.

Start des Firmware-Uploads

- Es wird so lange das Paket 0x21 gesendet, bis das Gerät „OK“ oder einen Fehler zurück gibt.

Übertragen der Firmware

- Das Binärfile kann, muss jedoch nicht in einzelne Teile unterteilt werden. Und die Länge der Pakete kann während der Übertragung unterschiedlich sein.
- Jeder Abschnitt beginnt mit einem 32b großen Zähler, der nach jedem Paket um die übertragene Länge inkrementiert werden muss.
- Um die Integrität der Datenübertragung sicherzustellen, wird jedes übertragene Paket bestätigt.
- Trifft keine Bestätigung ein, kann das letzte Paket unaufgefordert noch einmal gesendet werden. Für das Timeout empfiehlt sich ein Wert zwischen 0,5 und 1s.
- Das letzte Paket darf nicht mit leeren Bytes aufgefüllt werden, sondern hat entsprechend der verbleibenden Bytes eine geringere Länge.

Abschluss des Firmware-Uploads

- Der Upload der Firmware wird mit einem eigenen Paket abgeschlossen, das eine 32bit CRC der eben übertragenen Firmware enthält.
- Die CRC wird einzig über den Inhalt der Binärdatei gebildet.
- Um sicherzustellen, dass das Gerät das letzte Paket erhalten hat, ist eine zweifache Bestätigung implementiert:
 - PC > 0x23 + CRC
 - Gerät > OK
 - PC > OK
- Im Fall, dass ein Paket verloren geht, wird die Bestätigung vom Gerät wiederholt übertragen, solange der PC das Paket 0x23 sendet. Somit wird ausgeschlossen, dass das Gerät das letzte Paket nicht erhält oder der PC die Rückmeldung auf den Upload nicht empfängt.

Übertragen von zwei Teilen eines Firmware-Image

- Sind zwei Teile der Firmware zu übertragen, tritt zwischen dem ersten Teil und dem zweiten Teil eine Pause ein, während das Gerät einen Neustart.
- Das Gerät erkennt, dass sie den ersten Teil des Firmware-Image geladen hat, und fordert nach einem Neustart den zweiten Teil mit einem Paket vom Typ 0x24 an.
- Die Reihenfolge ist bindend: der erste Teil muss vor dem zweiten Teil übertragen werden.
- Wird somit einem Gerät nur der erste Teil übertragen wird dies bei jedem Neustart in den Zustand gehen, um den zweiten Teil der Firmware anzufordern.

Ablaufdiagramm

- Das nachfolgende Diagramm zeigt den Firmware – Upload aus der Seite der PC – Applikation als Flussdiagramm.
- Übergänge werden als „/“ mit einem Bezeichner der Bedingung dargestellt (wie in Zustandsdiagrammen).

CRC-32 Berechnung

- Die CRC-32 Berechnung verwendet das Polynom 0x04c11db7
- Der Initialwert der CRC ist 0xffffffff
- In die Berechnung der CRC werden die gesamten Daten des Firmware-Images einbezogen.

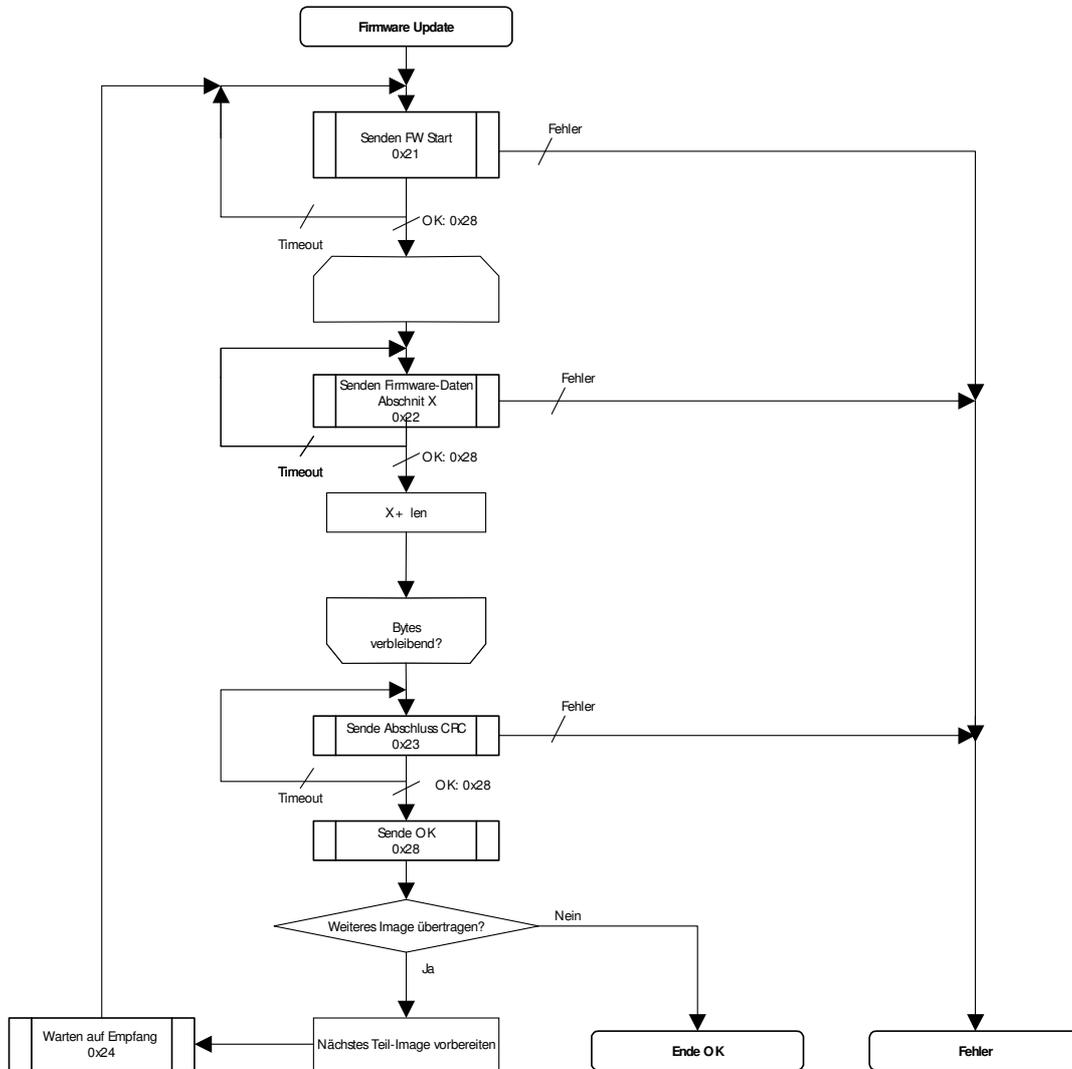


Abbildung 3 Firmware-Upload Flussdiagramm

Beispielkommunikationen für Funktionsgenerator

PC-Mode ein

PC sendet:

Start	Länge	Art	Adresse	Stopp
0x7D	0x03	0x4C	0x0100	0x7E

Antwort OK:

Start	Länge	Art	Adresse	Stopp
0x7D	0x03	0x1B	0x0100	0x7E

Frequenzwert setzen:

PC sendet

Start	Länge	Art	Adresse	Param	Daten	Stopp
0x7D	0x06	0x11	0x0100	0x05	0x01 0x07	0x7E

Antwort OK:

Start	Länge	Art	Adresse	Stopp
0x7D	0x03	0x18	0x0100	0x7E

Frequenzwert lesen:

PC sendet:

Start	Länge	Art	Adresse	Param.	Stopp
0x7D	0x04	0x12	0x0100	0x05	0x7E

Antwort:

Start	Länge	Art	Adresse	Param	Daten	Stopp
0x7D	0x07	0x1A	0x0100	0x05	0x01 0x07	0x7E