



Kostenlos: Demo-Software  
und Firmware-Update

## DDS130/DDS30 – das eigene Softwareinterface programmieren

Die beiden mikroprozessorgesteuerten Funktionsgeneratoren DDS130 und DDS30 sind nicht nur über eine PC-Schnittstelle bzw. die internen Bedienelemente steuerbar, ihre Kommunikationsschnittstelle bietet sich für weitere Steuerungsmöglichkeiten an. Mit der hier aufgeführten Offenlegung und Beschreibung der Übertragungsprotokolle eröffnen sich dem ambitionierten Programmierer genau diese Möglichkeiten. Eine Demosoftware ermöglicht die ersten praktischen Schritte dahin.

### Schnittstelle offengelegt

Der Funktionsgenerator DDS130 wurde so konzipiert, dass die gesamte Bedienung des Geräts über eine PC-Software erfolgt. Mit Kenntnis des Kommunikationsprotokolls ist aber auch eine Integration des Funktionsgenerators in ein eigenes Projekt bzw. in eine eigene Anwendung möglich, und zwar unabhängig von der beiliegenden PC-Software. Diese Möglichkeit ergibt sich ebenso für den DDS30, mit beiden lässt sich über dasselbe Protokoll kommunizieren. Da der DDS30 als autarkes Gerät konzipiert wurde, muss zur Ansteuerung durch ein externes Gerät zuerst am DDS30 eine USB-Schnittstelle installiert werden. Zur Ansteuerung beider Geräte steht eine Demosoftware und die Protokollbeschreibung [1] zur Verfügung.

### USB-Schnittstelle für DDS30

Anders als der DDS130 verfügt der DDS30 nicht über eine eigene USB-Schnittstelle. Jedoch befinden sich an der Stiftleiste ST2 des DDS30 alle notwendigen Daten- und Versorgungsleitungen, um eine USB-Schnittstelle mit einfachen Mitteln nachzurüsten.

Für die Nachrüstung einer USB-Schnittstelle eignen sich hervorragend das Mini-USB-Modul UM2102 [2] sowie das optisch getrennte USB-Modul U02102 [3], welches zusätzlich eine galvanische Trennung zwischen dem PC und dem DDS30 erzeugt. Durch die galvanische Trennung werden zusätzlich eventuelle Störsignale verhindert, die über das USB-Kabel in den Signalpfad des DDS30 gelangen könnten.

In den Tabellen 1 bis 3 sind die für die Datenkommunikation der einzelnen Komponenten relevanten Stiftleisten und deren Pins aufgelistet. In den zugehörigen Bildern 1 und 2 sind die jeweiligen Anschlussverbindungen des DDS30 mit dem UM2102 bzw. dem U02102 dargestellt.

### Firmware-Update für DDS30

Damit der DDS30 mit der Demosoftware verwendet werden kann, muss zunächst ein Firmware-Update auf die Version 2.0 oder höher eingespielt werden. Erst durch dieses Update werden die benötigten Kommunikationsprotokolle über die UART-Schnittstelle aktiviert.



Dazu ist der Bootloader des DDS30 über den Menüpunkt 5.5 *Firmwareupdate* zu starten und das auf der Produktseite bereitgestellte Firmware-Update über die Updatesoftware zu installieren.

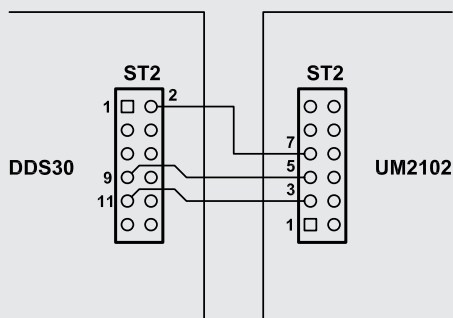
### Verbindung des DDS30 mit dem PC

Der Zugriff auf den DDS30 erfolgt mittels virtuellem COM-Port (VCP). Hierfür muss zuvor der entsprechende Treiber von Silabs auf dem System installiert werden. Die Verbindung des DDS30 mit dem PC über den VCP kann mit einem beliebigen Terminalprogramm getestet werden, beispielsweise HTerm [4]. Die Verbindungseinstellungen für die Verbindung zum COM-Port finden sich in [Tabelle 4](#). Der entsprechende Treiber für den DDS130 für die USB-Kommunikation liegt sowohl der beiliegenden PC-Software als auch der Demosoftware bei. Die benötigten Kommunikationsparameter für die USB-Verbindung finden sich in [Tabelle 5](#).

### Das Kommunikationsprotokoll

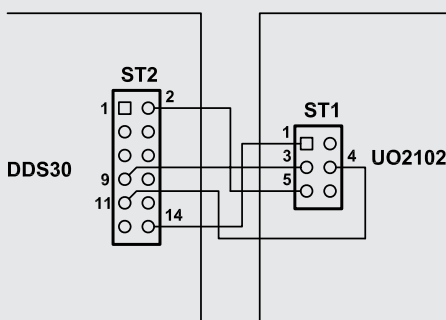
Beide Funktionsgeneratoren nutzen ein Kommunikationsprotokoll, welches dem des 5x5x5-RGB-Cube oder des ID100 stark ähnelt. Die Datenübertragung an den DDS30 bzw. DDS130 erfolgt in einem festen Kommunikationsrahmen ([siehe Tabelle 6](#)).

Bei der Übertragung gibt es zwei Steuerzeichen, die in den zu übertragenden Daten nicht vorkommen dürfen: Das Start-Zeichen (0x02) und das Escape-Zeichen (0x10). Wenn eines dieser Steuerzeichen in der Länge, den Nutzdaten oder der Prüfsumme



Kommunikationsverbindung DDS30-UM2102

Bild 1: Anschluss des DDS30 an das USB-Interface UM2102



Kommunikationsverbindung DDS30-UO2102

Bild 2: Anschluss des DDS30 an das USB-Interface UO2102

Tabelle 1

#### Pinbelegung der relevanten Daten- und Versorgungsleitungen beim DDS30

Stiftleiste	Pin	Funktion
ST2	1 & 2	Masse
ST2	9	RxD out
ST2	11	TxD in
ST2	13 & 14	+5 V

Tabelle 2

#### Pinbelegung der relevanten Daten- und Versorgungsleitungen beim UM2102

Stiftleiste	Pin	Funktion
ST2	3 & 4	TxD out
ST2	5 & 6	RxD in
ST2	7 & 8	Masse

Tabelle 3

#### Pinbelegung der relevanten Daten- und Versorgungsleitungen beim UO2102

Stiftleiste	Pin	Funktion
ST1	1 & 2	+5 V
ST1	3	RxD in
ST1	4	TxD out
ST1	5 & 6	Masse

Tabelle 4

#### Kommunikationsparameter für VCP-Verbindung mit DDS30

Baudrate	115200
Datenbits	8
Parität	none/keine
Stoppbits	1

Tabelle 5

#### Kommunikationsparameter für USB-Verbindungen mit DDS130

VID	0x18EF
PID	0xE010
Baudrate	76800
Datenbits	8
Parität	none/keine
Stoppbits	1

Tabelle 6

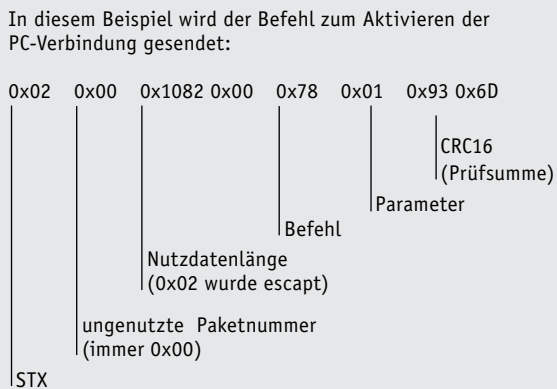
#### Kommunikationsrahmen

Inhalt	Bytezahl	Beschreibung
Startzeichen	1	STX (0x02), kennzeichnet Befehlsbeginn
Paketnummer	1	ungenutztes Byte des Protokolls, immer 0x00
Nutzdatenlänge	2	Anzahl der Nutzdaten (Befehl und Parameter) im Rahmen
Befehl	1	Befehl ( <a href="#">siehe Tabelle 4</a> )
Parameter	0 bis n	Parameter des Befehls ( <a href="#">siehe Tabelle 4</a> )
Prüfsumme	2	CRC16 (Prüfsumme) aller vorigen Byte

enthalten ist, wird diesem das Escape-Zeichen vorangestellt und das höchstwertige Bit (0x80) gesetzt. Hieraus ergeben sich folgende Umsetzungen: 0x02 wird zu 0x1082 und 0x10 wird zu 0x1090. Die einzelnen Befehle mit Parameterbeschreibungen können den Tabellen 7a–c entnommen werden. Bei allen 16-Bit- (2 Byte) und 32-Bit-Werten (4 Byte) wird das jeweils höchstwertige Byte zuerst übertragen (Big Endian).

Das folgende Beispiel zeigt die zu sendenden Byte, die die Verbindung des DDS mit einem PC aktiviert:

Beispiel Protokollrahmen



Zur Sicherung der Übertragung wird aus den zu übertragenden Daten eine Prüfsumme (CRC) mit einer Länge von 16 Bit (2 Byte) berechnet. Die Parameter für die Berechnung sind der Initialwert (0xFFFF) und das CRC-Polynom (0x8005). Details zur Prüfsummenberechnung sind beispielsweise auf Wikipedia [5] zu finden. In der Demosoftware wird für die Berechnung eine eigene „CRC16“-Klasse verwendet, die die CRC-Berechnung letztlich durchführt. Nach der Initialisierung wird die Prüfsumme durch Übergabe der Daten an die Shift-Methoden berechnet. Hierbei ist zu beachten, dass zuvor die Steuerzeichen aus der Länge und den Nutzdaten ersetzt werden müssen, bevor diese an die Shift-Methoden übergeben werden. Anschließend werden eventuelle Steuerzeichen aus dem errechneten CRC entfernt und zusammen mit den übrigen Daten an das Gerät übertragen. Die Antwort des Geräts setzt sich nach demselben Muster zusammen, einzig das Feld für die Parameter wird durch die Nutzdaten der Antwort ersetzt. In der Demosoftware sind Routinen definiert, die die Byte eines Befehls automatisch anhand des Befehlszeichens und der Parameter erzeugen und die die Nutzdaten aus der Antwort auf einen gesendeten Befehl extrahieren. Die Kommunikationsprotokolle des 5x5x5-RGB-Cube oder des ID100 unterscheiden sich nur durch das Fehlen der Paketnummer.

### Verfügbare Befehlszeichen

Bevor mit dem DDS30 bzw. DDS130 über eine PC-Software kommuniziert werden kann, muss die Verbindung mittels des Befehlszeichens „x“ und des Parameters 0x01 (siehe Befehlszeichenübersicht in den Tabellen 7a–c) initialisiert bzw. aktiviert werden. Das DDS130 besitzt auf der Vorderseite unterhalb der LED zur Energieversorgung eine LED, die den aktuellen Status der Geräteverbindung anzeigt. Das DDS30 zeigt die aktive Verbindung im Display an. Nach An-

steuerung des DDS muss mit demselben Befehl und dem Parameter 0x00 die Verbindung zwischen DDS und PC-Software wieder getrennt werden, um einen konsistenten Zustand des Geräts zu gewährleisten. Das erfolgreiche Umschalten des Verbindungsstatus wird mit einem 0x06 (ASCII-Zeichen für ein Acknowledge/Bestätigung) quittiert.

Beide Geräte besitzen drei Betriebsmodi: einen Standard-, einen Wobble- und einen Modulationsmodus. Der Betriebsmodus kann mit dem Befehl „e“ (0x65) gesetzt und mit „E“ (0x45) ausgelesen werden.

Bei den Befehlen zum Setzen („f“/0x66) und Auslesen („F“/0x46) der Frequenz wird der jeweilige Wert in Form von vier Byte ausgedrückt. Der DDS30 und der DDS130 erzeugen Frequenzen zwischen 0,25 Hz bis 30 MHz in einem Intervall von 0,25 Hz. Die zu setzende Frequenz wird zuerst mit 100 multipliziert, um sich möglicher Dezimalstellen zu entledigen. Dieser Wert wird, ausgedrückt als 32-Bit-Zahl (Big Endian), zur Kommunikation genutzt. Infolgedessen ergibt sich beispielsweise für die Frequenz 2.150.300,75 Hz folgendes Muster:

- Parameter 1:  $(215030075 \gg (0 * 8)) \& 0xFF = 0x0C$
- Parameter 2:  $(215030075 \gg (1 * 8)) \& 0xFF = 0xD1$
- Parameter 3:  $(215030075 \gg (2 * 8)) \& 0xFF = 0x19$
- Parameter 4:  $(215030075 \gg (3 * 8)) \& 0xFF = 0x3B$

Diese Vorgehensweise kann beispielhaft für alle Parameter angewandt werden, die mehr als 1 Byte benötigen.

### Ansteuerung mit der Demosoftware

Die mit ihrem Quelltext veröffentlichte Demosoftware zeigt eine Ansteuerung des DDS30 bzw. DDS130 anhand des zuvor erläuterten Protokolls. Sie ist eine Konsolenanwendung und wurde in der Sprache C# für das .NET-Framework 3.5 erstellt. Die Bearbeitung und das Starten können beispielsweise über eine der kostenlos verfügbaren Visual-Studio-Express-Versionen [6] erfolgen. Sowohl der DDS30 als auch der DDS130 lassen sich über die Demosoftware ansprechen, eine entsprechende Auswahl erfolgt beim Starten der Software. Nach einer erfolgreichen Verbindung kann einer von vier Menüpunkten ausgewählt werden, die jeweils einzelne mögliche Befehle der Befehlstabelle verdeutlichen. Hierzu gehören die explizite Aktivierung und Deaktivierung der PC-Verbindung zum DDS (Befehl „x“), jeweils die Ausgabe und Änderung der aktuellen Frequenz (Befehle „f“ und „F“) und der aktuellen Signalform (Befehl „s“ und „S“) sowie eine Ausgabe aller aktuell gesetzten Einstellungen des Geräts (Befehl „P“). Die genutzten Klassen für die Gerätekommunikation und der unter Umständen dazugehörigen DLL für die USB-Verbindung, zur Prüfsummenberechnung und der Generierung der Ansteuerungsbefehle können in eigene Projekte übernommen und genutzt werden. **ELV**



### Weitere Infos:

- [1] Download der Demosoftware mit folgenden Web-Codes:  
DDS130: #1282      DDS30: #1283
- [2] UM2102 – mehr unter Web-Code: #1269
- [3] UO2102 – mehr unter Web-Code: #1270
- [4] HTerm zur Kommunikation des DDS30 über VSP ohne separate Software: [www.der-hammer.info/terminal](http://www.der-hammer.info/terminal)
- [5] Wikipedia-Artikel zur Prüfsummenberechnung:  
[http://de.wikipedia.org/wiki/Zyklische\\_Redundanzprüfung](http://de.wikipedia.org/wiki/Zyklische_Redundanzprüfung)
- [6] Kostenlose Entwicklungsumgebung:  
Microsoft Visual Studio 20XX Express unter:  
[www.elv.de/l/?l=visualstudio](http://www.elv.de/l/?l=visualstudio)



Tabelle 7a: Verfügbare Befehlszeichen mit Parametern für DDS30 und DDS130

Gemeinsame Befehle					
Beschreibung	Befehl	Parameter (Anzahl der Byte)	Parameterinformationen		Antwort
PC-Verbindung aktivieren	x (0x78)	Status (1 Byte)	0x00:	PC-Verbindung deaktivieren	ACK NAK (*1)
			0x01:	PC-Verbindung aktivieren	
Geräteversion auslesen	V (0x56)	kein Parameter notwendig	–	–	Version (2 Byte) 100 bedeutet v1.00
Frequenz einstellen	f (0x66)	Frequenz (4 Byte)	Auflösung:	0,01 Hz	ACK NAK (*1)
			Bereich:	0,25 Hz – 30 MHz	
Frequenz auslesen	F (0x46)	kein Parameter notwendig	–	–	Frequenz (4 Byte)
Signalform einstellen	s (0x73)	Signalform (1 Byte)	0x00:	Sinus	ACK NAK (*1)
			0x01:	Dreieck	
			0x02:	Rechteck	
			0x03:	Rechteck halbe Frequenz	
Signalform auslesen	S (0x53)	kein Parameter notwendig	–	–	Signalform (1 Byte)
Sinusfilter schalten	a (0x61)	Filterstatus (1 Byte)	0x00:	Filter ausschalten	ACK NAK (*1)
			0x01:	Filter einschalten	
Status des Sinusfilters auslesen	A (0x41)	kein Parameter notwendig	–	–	Filterstatus (1 Byte)
PLL-Teilerfaktor einstellen	t (0x74)	Teilerfaktor (2 Byte)	Bereich:	1–2048	ACK NAK (*1)
PLL-Teilerfaktor auslesen	T (0x54)	kein Parameter notwendig	–	–	Teilerfaktor (2 Byte)
PLL-Offset einstellen	o (0x6f)	Offset (4 Byte vorzeichenbehaftet)	Auflösung:	0,1 kHz	ACK NAK (*1)
			Bereich:	-2,0 GHz – +2,0 GHz	
PLL-Offset auslesen	O (0x4f)	kein Parameter notwendig	–	–	Offset (4 Byte vorzeichenbehaftet)
Kalibrierwert des Oszillators einstellen	k (0x6b)	Kalibrierwert (8 Byte)	Auflösung:	0,01 Hz	ACK NAK (*1)
			Bereich:	66,109 MHz – 68,109 MHz	
Kalibrierwert des Oszillators auslesen	K (0x4b)	kein Parameter notwendig	–	–	Kalibrierwert (8 Byte)
Wobble-Modus konfigurieren	w (0x77)	Startfrequenz (4 Byte)	Auflösung:	0,01 Hz	ACK NAK (*1)
		Stoppfrequenz (4 Byte)	Bereich:	0,25 Hz – 30 MHz	
			Auflösung:	0,01 Hz	
Wobblefrequenz (2 Byte)	Bereich:	0,25 Hz – 30 MHz			
Betriebsmodus einstellen	e (0x65)	Modus (1 Byte)	0x00:	normaler Betriebsmodus	ACK NAK (*1)
			0x01:	Wobble-Modus	
			0x02:	Modulations-Modus	
Betriebsmodus auslesen	E (0x45)	kein Parameter notwendig	–	–	Modus (1 Byte)
Minimal-/Maximalfrequenz einstellen	c (0x63)	Minimalfrequenz (4 Byte)	Auflösung:	0,01 Hz	ACK (*2)
		Maximalfrequenz (4 Byte)	Bereich:	0,25 Hz – 30 MHz	
Frequenzschrittweite einstellen	z (0x7a)		Frequenzschrittweite (4 Byte)	Auflösung:	0,01 Hz
		Bereich:		0,25 Hz – 30 MHz	
Frequenzschrittweite auslesen	Z (0x5a)	kein Parameter notwendig	–	–	Frequenzschrittweite (4 Byte)
Kalibrierte Amplitude einstellen	g (0x67)	Spitze-Spitze-Spannung (2 Byte)	Auflösung:	1 mV	ACK (*2)
			Bereich:	max. 1000 mV	
Amplitude digital einstellen	d (0x64)	Amplitude digital (2 Byte)	Bereich:	0 - 16.383 Dieser Wert steuert direkt den DAC	ACK (*2)
Reset auf Werkseinstellungen	r (0x72)	kein Parameter notwendig	–	–	ACK

### Spezielle Befehle des DDS30

Beschreibung	Befehl	Parameter (Anzahl der Byte)	Parameterinformationen		Antwort
Modulations-Mode konfigurieren	m (0x6d)	Konfigurationswert (1 Byte)	Bit 0:	AM-Mode 0: inaktiv 1: aktiv	ACK NAK (*1)
			Bit 1:	Modulationsquelle 0: intern 1: extern	
			Bit 2 & Bit 3	Bit 3 = 0; Bit 2 = 0: FSK Bit 3 = 0; Bit 2 = 1: PSK Bit 3 = 1; Bit 2 = X: ASK	
		Frequenzhub (8 Byte)	Auflösung:	0,01 Hz	
			Bereich:	-30 MHz – +30 MHz	
		Phasenhub (2 Byte)	Auflösung:	1°	
			Bereich:	0 – 359°	
		Amplitudenhub (1 Byte)	Auflösung:	1 %	
			Bereich:	0 % – 100 %	
		Modulationsfrequenz (2 Byte)	Auflösung:	0,1 Hz	
			Bereich:	1 – 1000 Hz	
Aktuelle Konfiguration oder Speicherplatz programmieren	p (0x70)	Speicherplatz (1 Byte)	0:	aktuelle Konfiguration	ACK (*2)
			1–10:	Speicherplatz	
		Frequenz (4 Byte)	Auflösung:	0,01 Hz	
			Bereich:	0,25 Hz – 30 MHz	
		Minimalfrequenz (4 Byte)	Auflösung:	0,01 Hz	
			Bereich:	0,25 Hz – 30 MHz	
		Maximalfrequenz (4 Byte)	Auflösung:	0,01 Hz	
			Bereich:	Min.-Freq. – 30 MHz	
		Schrittweite (4 Byte)	Auflösung:	0,01 Hz	
			Bereich:	0,25 Hz – 30 MHz	
		Wobble-Startfrequenz (4 Byte)	Auflösung:	0,01 Hz	
			Bereich:	0,25 Hz – 30 MHz	
		Wobble-Stoppfrequenz (4 Byte)	Auflösung:	0,01 Hz	
			Bereich:	0,25 Hz – 30 MHz	
		Wobble-Frequenz (2 Byte)	Auflösung:	0,1 Hz	
			Bereich:	0,1 – 20 Hz	
		Signalform (1 Byte)	siehe Signalform einstellen		
Modulationskonfiguration (1 Byte)	siehe Modulation-Modus konfigurieren				
FSK-Frequenzhub (8 Byte)	Auflösung:	0,01 Hz			
	Bereich:	-30 MHz – +30 MHz			
PSK-Phasenhub (2 Byte)	Auflösung:	1°			
	Bereich:	0 – 359°			
Modulationsfrequenz (2 Byte)	Auflösung:	0,1 Hz			
	Bereich:	1 – 1.000 Hz			
	nur für interne Modulation				
	Bit 0:	Status des Sinusfilters			
		Bit 0 = 0: inaktiv Bit 0 = 1: aktiv			
Konfigurationswert (1 Byte)		Betriebsmodus			
	Bit 2 & Bit 1:	Bit 2 = 0; Bit 1 = 0: Standard-Modus Bit 2 = 0; Bit 1 = 1: Wobble-Modus Bit 2 = 1; Bit 1 = 0: Modulations-Modus			
PLL-Faktor (2 Byte)	Bereich:	1 – 2048			
PLL-Offset (4 Byte)	Auflösung:	0,1 kHz			
	Bereich:	-2,0 GHz – +2,0 GHz			
Aktuelle Konfiguration oder Speicherplatz auslesen	P (0x50)	Speicherplatz (1 Byte)	0: aktuelle Konfiguration		Frequenz (4 Byte)
			1–10: Speicherplatz		Minimalfrequenz (4 Byte)
					Maximalfrequenz (4 Byte)
					Schrittweite (4 Byte)
					Amplitude (2 Byte)
					Wobble-Startfrequenz (4 Byte)
					Wobble-Stoppfrequenz (4 Byte)
					Wobble-Frequenz (2 Byte)
					Signalform (1 Byte)
					Modulationskonfiguration (1 Byte)
					FSK-Frequenzhub (8 Byte)
					PSK-Phasenhub (2 Byte)
		Modulationsfrequenz (2 Byte)			
		Konfigurationswert (1 Byte)			
		PLL-Faktor (2 Byte)			
		PLL-Offset (4 Byte)			
PC-Verbindung manuell beendet	B (0x42)	kein Parameter notwendig	-	-	DDS30 erwartet keine Antwort

Tabelle 7b: Verfügbare Befehlszeichen mit Parametern für DDS30



Tabelle 7c: Verfügbare Befehlszeichen mit Parametern für DDS130

Spezielle Befehle des DDS130						
Beschreibung	Befehl	Parameter (Anzahl der Byte)	Parameterinformationen		Antwort	
Modulations-Mode konfigurieren	m (0x6d)	Konfigurationswert (1 Byte)	Bit 0:	AM-Mode 0: inaktiv 1: aktiv	ACK NAK (*1)	
			Bit 1:	Modulationsquelle 0: intern 1: extern		
			Bit 2:	Amplitudenmoduliert 0: nein 1: ja		
			Bit 2 & Bit 3:	Bit 3 = 0; Bit 2 = 0: FSK Bit 3 = 0; Bit 2 = 1: PSK Bit 3 = 1; Bit 2 = X: ASK		
		Frequenzhub (8 Byte)	Auflösung:	0,01 Hz		ACK NAK (*1)
			Bereich:	-30 MHz – +30 MHz		
		Phasenhub (2 Byte)	Auflösung:	1°		
		Bereich:	0 – 359°	ACK NAK (*1)		
Amplitudenhub (1 Byte)	Auflösung:	1 %				
	Bereich:	0 % – 100 %				
Modulationsfrequenz (2 Byte)	Auflösung:	0,1 Hz	ACK NAK (*1)			
	Bereich:	1 – 5000 Hz				
Aktuelle Konfiguration programmieren	p (0x70)	Frequenz (4 Byte)	Auflösung:	0,01 Hz	ACK (*2)	
			Bereich:	0,25 Hz – 30 MHz		
		Minimalfrequenz (4 Byte)	Auflösung:	0,01 Hz		
			Bereich:	0,25 Hz – 30 MHz		
		Maximalfrequenz (4 Byte)	Auflösung:	0,01 Hz		
			Bereich:	Min.-Freq. – 30 MHz		
		Schrittweite (4 Byte)	Auflösung:	0,01 Hz		
			Bereich:	0,25 Hz – 30 MHz		
		Wobble-Startfrequenz (4 Byte)	Auflösung:	0,01 Hz		
			Bereich:	0,25 Hz – 30 MHz		
		Wobble-Stoppfrequenz (4 Byte)	Auflösung:	0,01 Hz		
			Bereich:	0,25 Hz – 30 MHz		
		Wobble-Frequenz (2 Byte)	Auflösung:	0,1 Hz		
			Bereich:	0,1 – 20 Hz		
		Signalform (1 Byte)	siehe Signalform einstellen			
		Modulations-konfiguration (1 Byte)	siehe Modulation-Mode konfigurieren			
		FSK-Frequenzhub (8 Bytes)	Auflösung:	0,01 Hz		ACK (*2)
	Bereich:	-30 MHz – +30 MHz				
PSK-Phasenhub (2 Bytes)	Auflösung:	1°				
	Bereich:	0° – 359°				
Modulationsfrequenz (2 Bytes)	Auflösung:	0,1 Hz				
	Bereich:	1 Hz – 5000 Hz				
	nur für interne Modulation					
Konfigurationswert (1 Byte)	Bit 0:	Status des Sinusfilters Bit 0 = 0: inaktiv Bit 0 = 1: aktiv				
	Bit 2 & Bit 1:	Betriebsmodus Bit 2 = 0: Bit 1 = 0: Standard-Modus Bit 2 = 0: Bit 1 = 1: Wobble-Modus Bit 2 = 1: Bit 1 = 0: Modulations-Modus				
PLL-Faktor (2 Byte)	Bereich:	1–2048				
PLL-Offset (4 Byte)	Auflösung:	0,1 kHz				
	Bereich:	-2,0 GHz – +2,0 GHz				
Aktuelle Konfiguration auslesen	P (0x50)	kein Parameter notwendig			Version (2 Byte) Frequenz (4 Byte) Minimalfrequenz (4 Byte) Maximalfrequenz (4 Byte) Schrittweite (4 Byte) Amplitude (2 Byte) Wobble-Startfrequenz (4 Byte) Wobble-Stoppfrequenz (4 Byte) Wobble-Frequenz (2 Byte) Signalform (1 Byte) Modulations-konfiguration (1 Byte) FSK-Frequenzhub (8 Byte) PSK-Phasenhub (2 Byte) Modulationsfrequenz (2 Byte) Konfigurationswert (1 Byte) PLL-Faktor (2 Byte) PLL-Offset (4 Byte)	

\*1: Parameter falsch oder außerhalb des Wertes

\*2: Es erfolgt keine Bereichskontrolle der einzelnen Parameter!