

WIR RETTEN, WAS ANDERE
AUFGEBEN.

www.dmxface.at

DMXface

Communication manual

Für Version 5.12 bis 5.15

Änderungen zu Vorversionen	3
Allgemeines	4
Grundsätzlicher Aufbau eines Datenframes	5
Timeouts	5
Optionale Befehlsbytes	5
Command Reflection der Kommandos	6
Leere Antworten bzw. Leerframes	7
Die Kommunikation per USB oder integriertem LAN Controller beim DMXfaceXHn	7
Die Kommunikation per RS485	8
RS485 Datenframe mit Ziel - und Sender Adresse	8
RS485 Bus Adressbereiche	9
Kommunikation Syntax der Beschreibung	10
Basis Protokoll DMX Kommandos	12
DMX OUT Kanäle abfragen (Kanal 1-512 und interne 513-544)	12
DMX OUT Kanal abfragen mit zusätzlicher Text Antwort (Kanal 1-512 und interne 513-544)	12
DMX OUT Kanal setzen (auch mehrere in einem Befehl, Kanal 1-512 und interne 513-544)	12
DMX OUT Kanal setzen mit Text Rückantwort (Kanal 1-512 und interne 513-544)	12
DMX OUT Kanalwerte mit einer Fade Zeit übergeben	13
DMX OUT Kanal Namen abfragen (Kanal 1-512 und interne 513-544)	13
DMX OUT MASTER Wert setzen	13
DMX OUT MASTER Wert abfragen	13
DMX IN Eingangskanäle abfragen (KANAL 1-512)	14
Basis Protokoll Szenen / Programm Kommandos	15
Szene abrufen	15
Szenen Namen abfragen	15
Szene(n) bzw. Programm(e) aufrufen	15
Programm Name abfragen	15
Programm aufrufen	15
Szene vom DMX OUT speichern, optional Fadezeit und aktive Kanäle ändern	16
Szenenabruf mit anderem Helligkeitswert, optional mit alternativer Fadezeit	16
Szene mit alternativer Helligkeit abrufen optional Einstellung speichern	17
Szene aus Speicherplatz mit neuer Helligkeit abrufen	17
Szene aus Speicherplatz abrufen und Helligkeit um angegebenen Wert erhöhen	17
Szene aus Speicherplatz abrufen und Helligkeit um angegebenen Wert reduzieren	17
Basis Protokoll Trigger	18
Trigger Name abfragen	18
Trigger aufrufen	18
Trigger RTC Zeitschaltung abfragen	18
Trigger RTC Zeitschaltung setzen / schreiben	19
Einen oder mehrere Trigger auslösen	19
Basis Protokoll Ausgänge / OUTPUTS	20

- OUTPORT Wert setzen20
- OUTPORT Wert setzen (auch mehrere gleichzeitig)20
- OUTPORT Name abfragen20
- OUTPORT Wert abfragen20
- Basis Protokoll Eingänge / INPORTS21
 - INPORT Name abfragen21
 - INPORT inkl. Tabellenkonversionswerten abfragen21
 - BUS INPORT auf einen Wert setzen.....21
 - INPORT Analogwerte der ersten 8 Inports (lokale DMXface Eingänge)22
 - INPORT Analogwerte 24 Kanäle (DMXface + IO Extension).....22
 - INPORT Analogwerte der BUS Eingänge (CAN OPEN Erweiterung)22
 - INPORT alle Analogwerte abfragen 24 lokale + 32 BUS Eingänge23
- Basis Protokoll Status und Portabfragen24
 - Abfrage des aktuellen Gerätestatus24
 - Abfrage von 16 wählbaren DMX OUT Kanälen und dem Gerätestatus24
 - Portstatus + wählbare Anzahl von DMX OUT Kanälen ab Kanal 1 abfragen25
- Basis Protokoll Echtzeituhr lesen oder schreiben26
 - RTC Einstellung lesen.....26
 - RTC Einstellung schreiben.....26
- Weitere Befehle27
 - Interface Versionsstring abfragen.....27
 - Fehler Status Register abfragen27
- Befehle die zwar noch unterstützt aber nicht mehr empfohlen werden.28
 - DMX OUT Kanalwert abfragen (Kanal 1-255)28
 - DMX OUT auf einen Kanalwert setzen (Kanal 1-255)28
 - DMX OUT auf einen Kanalwert setzen (auch mehrere in einem Befehl, Kanal 1-256)28
 - DMX OUT Kanal inkl. Tabellenkonversionswerten abfragen (Kanal 1-255).....29
 - DMX OUT alle Kanäle abfragen.....29
 - DMX IN eine Anzahl von DMX Kanäle abfragen29
 - DMX OUT Kanal auf einen Wert setzen mit Rückantwort als Text (Kanal 1-255)30
 - INPORT Wert abfragen.....30

Änderungen zu Vorversionen

Ab Rev. 5.14

Antworten des DMXface werden grundsätzlich als lange Frames mit 16 Bit Längenwort und Startzeichen 0x01 ausgegeben.

Das DMXface ist auch mit einer PRO Firmware erhältlich die nicht nur 224 DMX Kanäle, sondern volle 512 DMX Kanäle am Ein- und Ausgang unterstützt.

Zusätzlich gibt es für interne Anwendungen weitere interne 32 DMX Kanäle.

Diese werden zwar nicht auf der DMX OUT Schnittstelle ausgegeben, sind aber innerhalb der DMXface Funktionen voll verwendbar. Somit insgesamt 544 DMX Kanäle

Um die Kanäle zu adressieren gibt es bei vielen Befehlen die bislang nur die Kanäle bis 256 adressieren konnten die Option durch zusätzliche Bytes auch die DMX Kanäle mit 16 Bit Adressierung zu erreichen.

Weiters auch einen neuen DMX16 Befehl der alle Funktionen abdeckt.

Ab Rev.5.12

wurde eine Master Channel Funktion für den DMX Out implementiert.

Diese Funktion bezieht sich **nur auf den DMX Output**, nicht auf die internen Verknüpfungen an DALI, KNX, CAN OPEN oder die lokalen PWM LED-Ausgänge.

Das ACTIVE Send wurde überarbeitet, die Steuerungsbits der Konfiguration haben neue Bedeutungen.

ACTIVE SEND Events werden nur noch auf **Serial PORT 1** ausgegeben.

Die Ausgabe am RS485 Port entfällt.

Zusätzlich ist die Ausgabe beim DMXfaceXHn (integrierter LAN-Controller) auch auf den LAN Sockets 6+7 möglich

Ab der DMXface Revision 5.04 sendet das DMXface zu jeder Anfrage eine „Command Reflection“

Das bedeutet das, das Kommando welches an DMXface gesendet wurde in der Antwort anhand der letzten 2 Bytes ausgewertet werden kann und somit empfangene Daten auch zugeordnet werden können, ohne dass die Abfrage unmittelbar zuvor und vom selben Host erfolgt ist.

Ab Revision 5.07 wurde das ACTS erweitert, sodass die Funktion auf für Serial Port 1 verfügbar ist.

Allgemeines

Der DMXface Controller bietet drei grundsätzliche Möglichkeiten der Kommunikation

- Die Kommunikation per lokalem USB-Anschluss
- Die Kommunikation über das im DMXfaceXHn integrierte LAN Modul (Protokollgleich mit USB) über ein Socket das im Modus **MAIN COMMUNICATION** läuft.
Es sind maximal 7 Socket gleichzeitig zur Kommunikation verfügbar.
- Die Kommunikation per RS485
Diese gilt auch für die am RS485 Bus angeschlossenen LAN Module **NET IO und NET DMX**

Für RS232 gibt es die vereinfachte Kommunikation über das [ACTIVE SEND](#) Protokoll welches auf dem Serialport1 des DMXface aktiviert werden kann.

Das Active Send Protokoll unterstützt Kommandos zum Abrufen von Szenen, Programmen, Setzen von DMX Kanälen und Outports, sowie DMX IN und OUT Abfragen und ist eigens dokumentiert.

Bei der Kommunikation mit RS485 ist der Zugriff über einen

RS485 Adapter mit einem IO NET oder NET DMX Zusatzmodul möglich. Die LAN-Schnittstellen dieser Zusatzmodule verbinden den RS485 BUS mit dem LAN. Da der RS485 Bus aber vorwiegend für die Kommunikation mit LCD Touch Displays genutzt wird, sollte für die Kommunikation per LAN vorzugsweise ein DMXfaceXHn mit einem integrierten LAN-Controller verwendet werden.

Der grundsätzliche Unterschied zwischen der Kommunikation per RS485 und USB bzw. integriertem LAN Modul besteht darin, dass am lokalen USB / internem LAN Port nur das angeschlossene Gerät angesprochen werden kann, bei der RS485 Kommunikation können am Bus bis zu 11 Geräte angeschlossen sein, die alle über die am DMXface vergebene BUS Adresse erreichbar sind.

Bei der RS485 Kommunikation ist daher neben den Nutzdaten auch die Weitergabe von Geräteadressen erforderlich. Dazu werden die BUS Adresse des Zielgerätes als auch die BUS Adresse des Senders an den ersten zwei Stellen im Datenbereich eingefügt.

Das DMXface, für das die Daten bestimmt sind, antwortet und verwendet als Zieladresse die empfangene Senderadresse.

Bei USB / internem LAN werden keine Geräteadressen benötigt, daher unterscheidet sich der Datenframe bei USB durch das Wegfallen der Zieladresse und Senderadresse im Datenbereich.

Grundsätzlicher Aufbau eines Datenframes

Es gibt zwei grundsätzliche Aufbauten eines Datenframes, welcher sich durch das Startbyte und die mögliche Datenlänge im Datenbereich unterscheidet.

Ein kurzer Datenframe verwendet das Startzeichen 0x02 und kann maximal 255 Bytes Daten transportieren.

Ein normaler Datenframe verwendet das Startzeichen 0x01, und sendet 2 Bytes zur Beschreibung der Datenlänge, daher sind theoretisch bis zu 65535 Bytes Datenlänge möglich.

Praktisch sind die Empfangspuffer mit einer Größe von 560Bytes begrenzt.

DMXface ab Rev. 5.14 antworten grundsätzlich mit normalen Datenframes. (Startzeichen 0x01 und 16 Bit Längenbeschreibung), Kommandos werden in beiden Varianten angenommen.

Auch Daten mit einer Länge unter 255 Bytes sollten als normaler Frame (16 Bit Länge) gesendet werden.

Grundstruktur	Startbyte	Länge Datenbereich		Datenbereich	End Byte
Kurzer Frame	0x02	0x00 bis 0xFF		0 bis 255 Bytes	0x03
Normaler Frame	0x01	Datenbereich Länge Low BYTE	Datenbereich Länge High BYTE	0 bis 65535 Bytes	0x03

Bei der Übertragung der Länge mit 16 Bit wird zuerst das LOW Byte, danach das HIGH Byte auf die Leitung geschickt.

Kommandos, BUS Adressen und Command Reflection sind immer Bestandteil des Datenbereiches auf dessen Länge sich auch die Längeninformation bezieht.

Timeouts

Die DMXface Controller antworten auf ein Kommando je nach Kommando und Auslastung typisch in 1-10ms. Zusätzlich fallen die Zeiten im LAN an, wenn der RS485 BUS über ein LAN Modul verbunden ist.

Daher sollte die Host Applikation nach ca. 100 – 200ms ohne Antwort den Kommunikationsprozess zum Gerät abbrechen und ein Timeout auslösen um nicht ewig auf eine Antwort zu warten.

Optionale Befehlsbytes

Beim Zugriff auf höhere DMX Kanäle (über 255) wird die Kanalnummer als 16 Bit übergeben.

Bei vielen Befehlen die bislang nur 8 Bit adressieren konnten, kann das High Byte der DMX Kanal Nummer optional am Ende gesendet werden.

Andere Befehle bieten zusätzliche Optionen die durch Hinzufügen von Informationen an das Kommando.

Optionale Bytes sind farblich gekennzeichnet.

Command Reflection der Kommandos

Jeder DMXface Befehl fängt mit einem Wert zwischen 0x00 und 0x7F an.

Bei der Antwort vom DMXface wird dieses Kommando Byte um den Wert 0x80 erhöht und am Ende der Antwort mit einem zusätzlichen Byte reflektiert.

Dies ermöglicht das asynchrone Empfangen von Antworten vom DMXface in Anwendungen, da Aufgrund der Auswertung der letzten 2 Bytes der Rückschluss auf die Zuordnung der Daten getroffen werden kann.

Als Beispiel das Statement Echtzeituhr lesen:

Kommando „ZR“ bzw. [0x5A], [0x52]

Das gesendete Kommando sieht wie folgt aus

USB / LAN

Startbyte	Länge LOW	Länge HIGH	CMD	CMD	ENDBYTE
0x01	0x02	0x00	0x5A	0x52	0x03

RS485

RS485 BUS Adresse

Startbyte	Länge LOW	Länge HIGH	An ADR	Von ADR	CMD	CMD	ENDBYTE
0x01	0x04	0x00	0x00 bis 0x0A DMXface	0xE0 bis 0xFE Rechner	0x5A	0x52	0x03

Bei der Antwort des DMXface wird das Kommando 0x5A um den Wert 0x80 erhöht und zusammen mit einem 2ten Byte das je nach Kommando unterschiedlich genutzt ist am Ende des Datenbereichs eingefügt.

Die Command Reflection ist immer 2 Byte lang und wird bei der Beschreibung der Kommandos blau dargestellt.

Antwort

USB

Startbyte	Länge 2 Byte	RTC DATEN	Reflection	Refl. Option	ENDBYTE
0x01	0x0B + 0x00	9 BYTE	0xDA	0x52	0x03

RS485

RS485 BUS Adresse

Startbyte	Länge 2 Byte	An ADR	Von ADR	RTC DATEN	Reflection	Refl. Option	ENDBYTE
0x02	0x0D + 0x00	0xE0- 0xFE Rechner	0x00-0x0A DMXface	9 BYTE	0xDA	0x52	0x03

Leere Antworten bzw. Leerframes

Jedes Kommando an einen DMXface Controller wird beantwortet.

Sind in der Antwort keine Daten erforderlich, so sendet das Gerät zumindest einen leeren Frame (Frame ohne Daten)

Bei USB oder integrierten LAN-Controller am DMXfaceXHn

Start Byte	Datenlänge		Befehl und Daten	Command Reflection	Command Reflection Option	End Byte
0x01	LOW=2	HIGH=0	0 Bytes leer	Command Byte + 0x80	1 Byte	0x03

Kürzeste Antwort bei der RS485 Kommunikation

Start Byte	Datenlänge		Befehl und Daten	BUS ADR Ziel	BUS ADR Quelle	Command Reflection	Command Reflection Option	End Byte
0x01	LOW=4	HIGH=0	0 Bytes	PC-Anwendung	DMXface Bus Adresse	Command Byte + 0x80	1 Byte	0x03

Die Kommunikation per USB oder integriertem LAN-Controller beim DMXfaceXP

Senden an ein DMXface per USB, oder integriertem LAN-Controller

Start Byte	Datenlänge		Befehl und Daten	End Byte
0x01	LOW(n)	HIGH(n)	n Bytes Kommando mit Daten	0x03

Empfang von einem DMXface an USB, oder mit integriertem LAN-Controller

Start Byte	Datenlänge		Daten	Command Reflection	Command Reflection Option	End Byte
0x01	LOW(n+2)	HIGH(n+2)	n Bytes Antwort mit Daten	Command Byte + 0x80	1 Byte	0x03

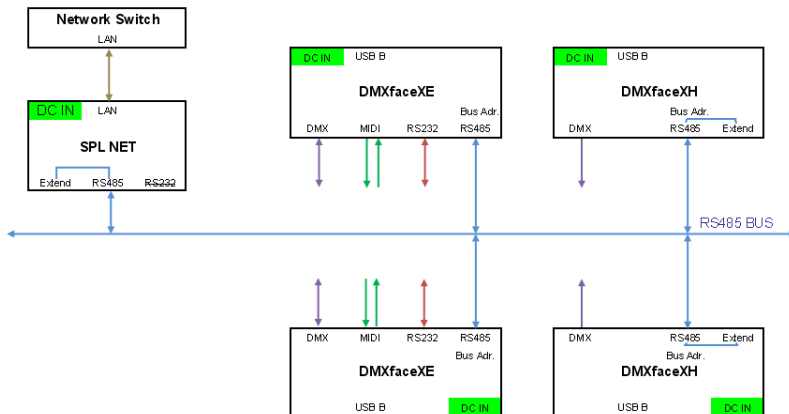
Die Kommunikation per RS485

Diese Kommunikation gilt auch für die **IO NET und LAN DMX Module** die den RS485 BUS mit dem LAN verbinden.

Bei einer RS485 Verbindung können bis zu 11 Geräte am selben Bus angeschlossen sein, jedes Gerät muss eine Busadresse besitzen welche nur einmal vorkommt.

Diese wird bei den DMXface Controller im Setup vergeben bzw. ist per Default auf Adresse Null eingestellt. Im Setup der Geräte kann auch ein Geräte name vergeben werden.

Beispiel Anbindung via BUS und einem SPL NET LAN Adapter



Die Übertragung findet mit 115200 Baud, 8 Datenbits, 1 Stopbit statt.

Bei RS485 fließen in den Datenbereich zusätzlich zu den Nutzdaten die BUS Adressen des Ziels und des Senders ein.

An Position 1 des Datenbereichs wird die Zielbusadresse des DMXface (Default 0x00, wenn nicht im Setup geändert) eingetragen.

An Position 2 des Datenbereiches erfolgt die Bus Adresse der Anwendung und kann im Bereich 0xE0 bis 0xFE frei gewählt werden.

Bei der Antwort des DMXface kommen im Datenbereich die 2 Adressen in umgekehrter Reihenfolge wieder zurück.

Die weiteren Daten sind gleich wie bei der Kommunikation über USB oder integriertem LAN Modul.

RS485 Datenframe mit Ziel - und Sender Adresse

Frame RS485	Startbyte	Länge Datenbereich		Zieladresse	Eigene Adresse	Nutzdaten	End Byte
Kurzer Frame	0x02	0x00 bis 0xFF		0x00 bis 0x0A	0xE0 bis 0xFE	0 bis 253 Bytes	0x03
Normaler Frame	0x01	Länge Low Byte	Länge High Byte	0x00 bis 0x0A	0xE0 bis 0xFE	0-65533 Bytes	0x03

Die Länge des Datenbereichs wird aus den grün markierten Feldern berechnet.

Vorzugsweise sollten zur Kommunikation normale Frames verwendet werden, das DMXface ab Rev. 5.14 antwortet immer im Format der normalen Frames. Verstehen kann es beide Formate.

Beispiel:

Senden an ein DMXface per RS485, NETIO Modul oder NET DMX Modul

Start Byte	Datenlänge		BUS ADR Ziel	BUS ADR Quelle	Befehl und Daten	End Byte
0x01	LOW(n+2)	HIGH(n+2)	DMXface BUS Adresse lt. DMXface Settings	Gewählte BUS Adr. Des Host Programmes	n Bytes Kommando mit Daten	0x03

Empfang von einem DMXface an RS485, NETIO Modul oder NET DMX Modul

Start Byte	Datenlänge		Daten	BUS ADR Ziel	BUS ADR Quelle	Command Reflection	Command Reflection Option	End Byte
0x01	LOW(n+2)	HIGH(n+2)	n Bytes Kommando mit Daten	DMXface BUS Adresse lt. DMXface Settings	Gewählte BUS Adr. Des Host Programmes	Command Byte + 0x80	1 Byte	0x03

RS485 Bus Adressbereiche

Bus Adresse	Reservierung / Verwendung
0x00-0x0A	Möglicher Adressbereich der DMXface Controller
0x0B bis 0xDF	Reserviert
0xE0-0xFE	PC-Anwendungen / Programme die DMXface Controller ansprechen
0xFF	Broadcast an alle Empfänger ohne Erwartung einer Antwort

Der Bereich 0xE0 bis 0xFE wird für die PC / Steuergeräte und deren Applikation verwendet.

Ein senden an die BUS Adresse 255 (0xFF) wird von allen am Bus befindlichen Controller empfangen und ausgeführt.

Es wird dann von keinem Controller eine Antwort gesendet da diese ja am Bus kollidieren würde.

Kommunikation Syntax der Beschreibung

[,A'] der Buchstabe „A“ = ASCII 65 wird gesendet.

[10] der Dezimalwert 10 bzw. das ASCII Zeichen 10 wird gesendet

[0x20] der Wert HEX 20 bzw. das ASCII Zeichen 32 wird gesendet

Szene / Programm Werte

	Das aufrufen einer Szene oder eines Programms erfolgt durch einen Wert 0-255
Wert 0	Das Interface führt ein ALL OFF aus (Alle DMX Out = 0, alle Outports = 0, alle Timer abbrechen)
Wert 1- 180	Szene 1-180
Wert 224-251	Zum Abruf der Programme 1-28
Wert 255	Keine Funktion, es wird keine Szene oder Programm abgerufen.

Inport / Output Byte

Wert 0-255	Der Wert entspricht dem Bitstatus des entsprechenden In oder Output Port Bit								
	8	7	6	5	4	3	2	1	
Wert	0x80	0x40	0x20	00x10	00x8	00x4	0x02	0x01	

Output Maske

Wert 0-255 Eine Outputmaske beschreibt mit einem gesetzten Bit, welches Port Bit von einer Änderung betroffen ist, und welche Port Bits unverändert bleiben.
Das Port Byte übergibt den neuen Wert, die Maske übergibt welche der acht Bits geändert werden sollen oder unverändert im aktuellen Zustand bleiben.

Timer status byte

Wert 0-255	Dieses Status Byte zeigt mit einem gesetzten Bit an das der entsprechende Timer gerade aktiv ist.								
Timer	8	7	6	5	4	3	2	1	
Wert	0x80	0x40	0x20	00x10	00x8	00x4	0x02	0x01	

Timer disable byte

Hat dieser Status einen Wert > 0 so sind alle Ausführungen von Timer und Programmen deaktiviert.

- 0x01 Timer Ausführung ist deaktiviert, da User das Bit gesetzt hat.
- 0x10 ACTIVE SEND ist temporär abgeschaltet
- 0x40 Timer Ausführung blockiert da ein **Excessive flash write** Fehler aufgetreten ist. Dieser Fehler tritt auf, wenn ein Userprogramm eine Szene permanent überschreibt und dadurch die Lebenszeit des Flashspeichers unzulässig verkürzt wird.
- 0x80 Das Betriebssystem hat eine Endlosschleife in den Programmblöcken festgestellt. Dies würde zum Absturz des Controllers führen. Wird eine Schleife entdeckt die nicht nach 5 Unterschleifen zurückkehrt, so wird das Bit 0x80 gesetzt und die weitere Ausführung von Programmen verhindert.

Kommunikation, Allgemeines

DMX IN / OUT Wert

Wert 0-255 Entspricht dem am DMXIN bzw. OUT anliegenden Kanalwert

DMX Mode Status Byte

Einige DMXface Typen unterstützen einen DMX Read Mode

Bei aktiviertem DMX Read Mode wird der DMX Out Ausgang auf einen DMX Eingang bzw. die Daten von einem DMX Eingang gelesen, wenn dieser vorhanden ist.

- 0x01 Bit = 1 Interface im DMX Read Mode
- 0x80 Bit = 1 Interface unterstützt DMX Read Mode

ACTIVE Send Kontrollbits

Bei den DMXfaceXH/XP und XE gibt es die Möglichkeit ein ACTIVE SEND zu aktivieren.

Das bedeutet, dass gewählte Events wie z.B. ein Infrarot Empfang oder eine Änderung an Inport Zuständen in einer Warteschlange gespeichert und wahlweise auf SERIAL Port 1, LAN SOCKET6 und LAN SOCKET 7 ausgegeben werden. (DMXfaceXP mit integriertem LAN-Controller)

Die Ausgabe auf dem RS485 Port entfällt ab Rev. 5.12

ACTIVE END MODE Bits

- 0x01 ACTIVE SEND PROTOKOLL EMPFANG am SERIAL PORT 1 aktivieren
- 0x02 ACTIVE SEND EVENTS an SERIAL PORT 1 senden
- 0x04 DMX OUTPUT INFO an SERIAL PORT 1 senden
- 0x08 keine Funktion

Mit integrierter LAN-Option

- 0x10 ACTIVE SEND EVENTS an LAN SOCKET 6 senden
 - 0x20 DMX OUTPUT INFO an LAN SOCKET 6 senden
 - 0x40 ACTIVE SEND EVENTS an LAN SOCKET 7 senden
 - 0x80 DMX OUTPUT INFO an LAN SOCKET 7 senden
- Details dazu im Bereich der Beschreibung ACTIVE SEND

Basis Protokoll DMX Kommandos

DMX OUT Kanäle abfragen (Kanal 1-512 und interne 513-544)

Einen oder ganze Blöcke von DMX OUT Kanälen abfragen.

Befehl: [0x55], [0x00], [DMX Adr. high], [DMX Adr. low], optional [Anzahl HIGH], [Anzahl LOW]

Antwort:

[DMX CHANNEL HIGH], [DMX CHANNEL LOW], [Wert1], [Wert2], ... [0xD5], [0x00]

DMX Kanäle werden als 16 Bit Wert ab Kanal 1 übergeben.

Bei Abfragen die mehr als die verfügbaren DMX Kanäle anfordern, werden nur die verfügbaren Kanäle ausgegeben.

Werden die optionalen Bytes für einen Abfrage von mehr Kanälen weggelassen so wird nur der abgefragte Kanal ausgegeben.

DMX OUT Kanal abfragen mit zusätzlicher Text Antwort (Kanal 1-512 und interne 513-544)

Einen DMX Kanal mit zusätzlichem Text Output optional aus der Tabellenkonversion abfragen.

Befehl: [0x55], [0x01], [DMX Adr. high], [DMX Adr. low]

Antwort:

[DMX CHANNEL HIGH], [DMX CHANNEL LOW], [Wert], [TEXT String], [0x00], [0xD5], [0x01]

DMX Kanal wird als 16 Bit Wert ab Kanal 1 übergeben.

Ist der DMX Kanal einer Übersetzungstabelle zugewiesen so wird der Text aus der Tabellenkonversion generiert. Ansonsten enthält der String den Dezimalwert der Einstellung des abgefragten DMX Kanals.

DMX OUT Kanal setzen (auch mehrere in einem Befehl, Kanal 1-512 und interne 513-544)

Einen oder mehrere DMX Kanäle setzen

Befehl: [0x55], [0x02], [DMX Adr.1 high], [DMX Adr.1 low], [DMX Wert1], [DMX Adr.2 high], ...

Antwort Leerframe: [0xD5], [0x02]

DMX Kanäle werden als 16 Bit Wert ab Kanal 1 übergeben.

Es können bis zu 180 Kanäle gleichzeitig gesetzt werden.

DMX OUT Kanal setzen mit Text Rückantwort (Kanal 1-512 und interne 513-544)

Einen DMX Kanal setzen und eine Antwort mit Wert und Text erhalten

Befehl: [0x55], [0x03], [DMX Adr. high], [DMX Adr. low], [DMX Wert]

Antwort:

[DMX CHANNEL HIGH], [DMX CHANNEL LOW], [DMX WERT], [TEXT STRING], [0x00], [0xD5], [0x03]

DMX Kanäle werden als 16 Bit Wert ab Kanal 1 übergeben.

Ist der DMX Kanal einer Übersetzungstabelle zugewiesen so wird der Text aus der Tabellenkonversion generiert. Ansonsten enthält der String den Dezimalwert der Einstellung des abgefragten DMX Kanals.

DMX OUT Kanalwerte mit einer Fade Zeit übergeben

Befehl [0x20], [0x09], [DMX KANAL high], [DMX KANAL low], [DMX Wert], [Fadezeit]
 Der DMX Kanal wird unter Anwendung der Fadezeit auf den neuen Wert geblendet.
 Es können auch mehrere DMX Kanäle gleichzeitig bedient werden. Max 512 Bytes
 DMX Kanalnummer wird ab 1 als 16 Bit Wert übergeben.

[DMX KANAL high] Wertbereich 0x00-0x02 DMX Kanalnummer ab 1 high Byte
 [DMX KANAL low] Wertbereich 0x00-0xFF DMX Kanalnummer ab 1 low Byte
 [DMX Wert] Wertbereich 0x00-0xFF
 [Fadezeit] Wertbereich 0x00-0xFF

Antwort 2 Bytes: [Command Reflection: \[0xA0\], \[0x09\]](#)

Es können auch mehrere DMX Kanäle gleichzeitig bedient werden. (max. 128 Kanäle / 514 Bytes)
 [0x20], [0x09], [DMX KANAL1 high], [DMX KANAL1 low], [DMX Wert1], [Fadezeit1], [DMX KANAL 2 high], [DMX KANAL2 low],
 [DMX Wert2], [Fadezeit2], ...

DMX OUT Kanal Namen abfragen (Kanal 1-512 und interne 513-544)

Befehl: [0x4F], [0x44], [DMX OUT Kanal ab 1], [0x4E] für DMX Kanal 1-255

Befehl: [0x4F], [0x44], [DMX OUT ab 1 high], [DMX OUT ab 1 low] [0x4E] für DMX Kanal 1-544

*Liefert den Namen des DMX Kanals der am DMXface programmiert wurde.
 (DMXface Console, Text and Overview oder Set Text)*

Antwort: 34 Bytes

[32 Bytes Name des DMX Kanals]

[Command Reflection \[0xCF\], \[0x00\]](#)

DMX OUT MASTER Wert setzen

Befehl [0x20] [0x0A], [NEUER MASTER WERT 0-255]

[MASTER WERT] Wertbereich 0x00-0xFF

Antwort 2 Bytes [Command Reflection: \[0xA0\], \[0x0A\]](#)

Dieser Befehl setzt einen **DMX OUT** Masterwert.

Alle am DMX OUT Anschluss ausgegebenen Kanäle werden über den DMX Master Wert gesteuert.

Nicht betroffen sind die internen Funktionen:

- Ausgaben auf den lokalen PWM Ausgängen
- Angekoppelte DALI, KNX oder CAN Open Gateways

DMX OUT MASTER Wert abfragen

Befehl [0x20] [0x0B]

Antwort 3 Bytes:

[DMX OUT Master Wert 0x00-0xFF]

[\[0xA0\], \[0x0B\] Command Reflection](#)

Der DMX OUT MASTER WERT wird auch bei der Statusabfrage [\[0x43\]](#) und [\[0x51\]](#) mitgesendet.

DMX IN Eingangskanäle abfragen (KANAL 1-512)

Einen oder ganze Blöcke von DMX IN Kanälen abfragen.

Befehl: [0x55], [0x05], [DMX Adr. high], [DMX Adr. low], optional [Anzahl HIGH], [Anzahl LOW]

Antwort:

[DMX CHANNEL HIGH], [DMX CHANNEL LOW], [Wert1], [Wert2], ... [0xD5], [0x05]

DMX Kanäle werden als 16 Bit Wert ab Kanal 1 übergeben.

Bei Abfragen die mehr als die verfügbaren DMX IN Kanäle anfordern, werden nur die verfügbaren Kanäle ausgegeben.

Werden die optionalen Bytes für einen Abfrage von mehr Kanälen weggelassen so wird nur der abgefragte Kanal ausgegeben.

Basis Protokoll Szenen / Programm Kommandos

Szene abrufen

Befehl: [0x4F], [0x53], [Szenennummer ab 1], [0x43]

Ruft die Szene auf.

Antwort: 2 Bytes

Command Reflection [0xCF], [0x00]

Szenen Namen abfragen

Befehl: [0x4F], [0x53], [Szenennummer ab 1], [0x4E]

Liefert den Namen der Szene der beim Speichern vergeben wurde. (DMXface Console, Text and Overview oder Set Text)

Antwort: 34 Bytes

[32 Bytes Name der Szene]

Command Reflection [0xCF], [0x00]

Szene(n) bzw. Programm(e) aufrufen

Befehl [0x50], [Szenen / PG Nummer], [Szenen / PG Nummer], ...

Befehlslänge 2 bis max. 512 Bytes

Durch den Befehl 0x50 kann eine oder auch mehrere Szenen bzw. Programme abgerufen werden.

[Szenen / PG Nummer]

Szenennummer 0 = ALL OFF

Szenennummer 1-180 sind die im DMXface gespeicherten Szenen

Szenennummer 224-251 entspricht den Programmen 1-28

Szenennummer 255 reserviert, keine Funktion

Antwort: 2 Byte

Command Reflection [0xD0] und [Szenen Nummer]

Bei mehreren Abrufen in einem Command steht an letzter Stelle die letzte Szenen- / Programm Nummer die abgerufen wurde.

Programm Name abfragen

Befehl: [0x4F], [0x50], [Programm 1-28], [0x4E]

Liefert den Namen des Programmblock der beim Speichern vergeben wurde. (DMXface Console)

Antwort: 34 Bytes

[32 Bytes Name des Programms]

Command Reflection [0xCF], [0x00]

Programm aufrufen

Befehl: [0x4F], [0x50], [Programm 1-31], [0x43]

Ruft das Angegebene Programm am DMXface auf

Antwort: 2Bytes

Command Reflection [0xCF], [0x00]

Szene vom DMX OUT speichern, optional Fadezeit und aktive Kanäle ändern

Befehl: [0x4F], [0x53], [Szenennummer ab 1], [0x53], optional [FADE Time], [DMX CH1 on/off], [DMX CH2 on/off], [DMX CH3 on/off], ...

Alle Kanäle die in der Szene aktiviert sind werden mit den, aktuell am DMX OUT, anliegenden Werten neu gespeichert.

Fade Zeiten, Output und sonstige Einstellungen der Szene bleiben unverändert.

Optional können zusätzliche Daten für die Fadezeit der Szenen sowie für jeden Kanal ein Byte mit der Information, ob dieser in der Szene aktiviert (Wert>0) oder deaktiviert (Wert =0) sein soll, angehängt werden.

Kanäle die nach dem letzten gesendeten Byte liegen behalten ihren ON/OFF Status in der Szene und werden nicht geändert.

z.B.

Befehl		FADE	CH1	CH2	CH3	CH4	CH5	CH6	CH7	CH8
[0x4F], [0x53], [0x01], [0x53], [0x20], [0x01], [0x01], [0x01], [0x01], [0x00], [0x00], [0x00], [0x00]										

Dieses Command aktiviert die Kanäle 1-4 und deaktiviert die Kanäle 5-8 in der Szene 1

Die Fadezeit der Szene wird auf 3,2 Sekunden gesetzt

Die Aktivierung dahinter liegender Kanäle für die keine Information gesendet wurde bleibt unverändert.

Antwort: 2 Bytes

Command Reflection [0xCF], [0x00]

Szenenabruf mit Helligkeitswert, optional mit alternativer Fadezeit

Befehl [0x20], [0x00], [Szene], [Helligkeit], optional [Fadezeit]

Dieser Befehl ruft die genannte Szene mit einer übergebenen Wiedergabehelligkeit ab.

Wird zusätzlich auch ein 5tes Byte gesendet, so wird dies anstelle der Programmierten Fadezeit der Szene herangezogen.

[Szene] Wertbereich 1-180

[Helligkeit] Wertbereich 0-255

[Fadezeit] Wertbereich 0-255, Byte ist optional, wird es nicht gesendet so wird die in der Szene programmierte Fadezeit herangezogen.

0-100 Fadezeit 0-10 Sek.

101-255 Fadezeit 11-165 Sek.

Antwort 2 Bytes

Command Reflection: [0xA0], [0x00]

DMXface stellt 32 Speicherplätze zur Verfügung an denen eine Szene und die Helligkeitseinstellung der Szene aus einem Abruf gespeichert werden kann.

Im Anschluss kann der Abruf von dem Speicherplatz mit einer erhöhten oder reduzierten Helligkeit erneut erfolgen. Die Speicher verlieren den Inhalt bei einem Neustart des Interface.

Szene mit alternativer Helligkeit abrufen optional Einstellung speichern

Befehl [0x20], [0x01], [Szene], [Helligkeit], [optional Speicherplatz 0-31]

[Szene] Wertbereich 1-180

[Helligkeit] Wertbereich 0-255

[Speicher Nr.] Wertbereich 0-31, Byte ist optional, wird es nicht gesendet so wird die in der Szene mit der Helligkeit abgerufen ohne dass etwas gespeichert wird.

Antwort 2 Bytes

Command Reflection: [0xA0], [0x01]

Szene aus Speicherplatz mit neuer Helligkeit abrufen

Befehl [0x20] [0x02], [Speicherplatz 0-31], [optional neue Helligkeit 0-255]

Wird einen neue Helligkeit mitgeschickt wird die Szene aus dem Speicher mit der gesendeten Helligkeit wiedergegeben und der Speicher aktualisiert, ansonsten erfolgt die Wiedergabe der Szene mit dem gespeicherten Wert.

[Szene] Wertbereich 1-180

[Speicher Nr.] Wertbereich 0-31

[Helligkeit] Wertbereich 0-255 optional

Antwort 2 Bytes

Command Reflection: [0xA0], [0x02]

Szene aus Speicherplatz abrufen und Helligkeit um angegebenen Wert erhöhen

Befehl [0x20] [0x03], [Speicherplatz 0-31], [Helligkeitserhöhung 1-255]

Die Helligkeit der gespeicherten Szene wird um, den angegebenen Wert erhöht und wiedergegeben. Der Speicher wird aktualisiert.

[Speicher Nr.] Wertbereich 0-31

[Helligkeitserhöhung] Wertbereich 1-255, dieser Wert wird zur gespeicherten Helligkeit addiert und die Szene abgerufen. Der Speicher wird mit der neuen Helligkeit aktualisiert.

Antwort 2 Bytes

Command Reflection: [0xA0], [0x03]

Szene aus Speicherplatz abrufen und Helligkeit um angegebenen Wert reduzieren

Befehl [0x20] [0x04], [Speicherplatz 0-31], [Helligkeitsreduktion 1-255]

Die Helligkeit der gespeicherten Szene wird um, den angegebenen Wert reduziert und wiedergegeben. Der Speicher wird aktualisiert.

[Speicher Nr.] Wertbereich 0-31

[Helligkeitserhöhung] Wertbereich 1-255, dieser Wert von der gespeicherten Helligkeit subtrahiert und die Szene abgerufen. Der Speicher wird mit der neuen Helligkeit aktualisiert.

Antwort 2 Bytes

Command Reflection: [0xA0], [0x04]

Basis Protokoll Trigger

Trigger Name abfragen

Befehl: [0x4F], [0x54], [Trigger Nummer 1-64], [0x4E]

Liefert den Namen des Triggers der beim Speichern vergeben wurde. (DMXface Console)

Antwort: 34 Byte

[32 Byte Trigger Name], *Command Reflection* [0xCF], [0x00]

Trigger aufrufen

Befehl: [0x4F], [0x54], [Trigger Nummer 1-48], [0x43]

Führt den Trigger aus, wie wenn eines der am Trigger programmierten Ereignisse eingetreten wäre.

Antwort: *Command Reflection* 0xCF, 0x00

Trigger RTC Zeitschaltung abfragen

Befehl: [0x4F], [0x54], [Trigger Nummer 1-48], [0x54]

Antwort: 10 Bytes

[Aktiv], [Sek], [Min], [Std], [WT], [Tag], [Mon], [Jahr], *Command Reflection* [0xCF], [0x00]

Bei den folgenden Registern, ist der dezimale Zeitwert als Hexcode eingesetzt.

Soll eine Prüfung des Teils der Zeit erfolgen, wird zusätzlich auch das Bit7(0x80) im jeweiligen Register gesetzt

[Aktiv] Zeitschaltfunktion aktiv 0x80 oder Deaktiviert = 0x00

[Sek] Sekundenprüfung, Bit 7 gesetzt, wenn aktiv, Bit 1-6 Sekunden Dezimal als Hex

[Min] Minutenprüfung, Bit 7 gesetzt, wenn aktiv, Bit 1-6 Sekunden Dezimal als Hex

[Std] Stundenprüfung, Bit 7 gesetzt, wenn aktiv, Bit 1-6 Sekunden Dezimal als Hex

[WT] Wochentags Prüfung, Bit 7 gesetzt, wenn aktiv, Bit 1-6 stehen jeweils für einen Tag

[Tag] Tagesprüfung, Bit 7 gesetzt, wenn aktiv, Bit 1-6 Sekunden Dezimal als Hex

[Mon] Monatsprüfung, Bit 7 gesetzt, wenn aktiv, Bit 1-6 Sekunden Dezimal als Hex

[Jahr] Jahresprüfung, Bit 7 gesetzt, wenn aktiv, Bit 1-6 Sekunden Dezimal als Hex

Beispiel für einen Sekunden, Minuten ... Wert

Bereich 0-59 Sekunden entspricht einem einzusetzenden Wert 0x00 bis 0x59 also nicht nach HEX umgerechnet, sondern 1:1 ins HEX übernommen.

Wird die Prüfung auf den entsprechenden Teil in der Zeitschaltfunktion aktiviert so wird zusätzlich das Bit x80 gesetzt. Weiters muss auch der Eintrag gesamt aktiviert werden und das [Aktiv] Byte auf den Wert 0x80 gesetzt werden.

Beispiel:

Hex 0x11 entspricht der elften Sekunde.

Soll auf Sekunden geprüft werden wird zusätzlich das Bit 7 gesetzt

Bereich 0x80 (Sekunde 0) bis 0xD9 (Sekunde 59)

Nicht verwendete / geprüfte Teile des Datums oder der Uhrzeit können auf 0x00 belassen werden.

Trigger RTC Zeitschaltung setzen / schreiben

Befehl: [0x4F], [0x54], [Trigger Nummer 1-64], [0x74], [8 Byte Zeitschaltdaten]

Deaktiviert oder setzt eine Trigger Zeitschaltfunktion auf einen neuen Wert

Zeitschaltdaten:

[Aktiv], [Sek], [Min], [Std], [WT], [Tag], [Mon], [Jahr]

Antwort: 2Bytes *Command Reflection* [0xCF], [0x00]

Bei den folgenden Registern, ist der dezimale Zeitwert als Hexcode eingesetzt.

Soll eine Prüfung des Teils der Zeit erfolgen, wird zusätzlich auch das Bit7(0x80) im jeweiligen Register gesetzt

[Aktiv]	Zeitschaltfunktion aktiv 0x80 oder Deaktiviert = 0x00
[Sek]	Sekundenprüfung, Bit 7 gesetzt, wenn aktiv, Bit 1-6 Sekunden Dezimal als Hex
[Min]	Minutenprüfung, Bit 7 gesetzt, wenn aktiv, Bit 1-6 Sekunden Dezimal als Hex
[Std]	Stundenprüfung, Bit 7 gesetzt, wenn aktiv, Bit 1-6 Sekunden Dezimal als Hex
[WT]	Wochentags Prüfung, Bit 7 gesetzt, wenn aktiv, Bit 1-6 stehen jeweils für einen Tag
[Tag]	Tagesprüfung, Bit 7 gesetzt, wenn aktiv, Bit 1-6 Sekunden Dezimal als Hex
[Mon]	Monatsprüfung, Bit 7 gesetzt, wenn aktiv, Bit 1-6 Sekunden Dezimal als Hex
[Jahr]	Jahresprüfung, Bit 7 gesetzt, wenn aktiv, Bit 1-6 Sekunden Dezimal als Hex

Beispiel Uhrzeit 18:00:00 Uhr täglich

Daten:

[Aktiv] = 0x80	RTC Timer aktiviert
[Sek] = 0x80	Prüfung Sekunde aktiv, Wert = Sekunde 0x00 + Aktiv 0x80
[Min] = 0x80	Prüfung Minute aktiv, Wert = Minute 0x00 + Aktiv 0x80
[Std] = 0x98	Prüfung Stunde aktiv, Wert = Stunde 0x18 + Aktiv 0x80
[WT] = 0x00	Keine Wochentags Prüfung, Bit 7/0x80 nicht gesetzt
[Tag] = 0x00	Keine Kalendertag Prüfung, Bit 7/0x80 nicht gesetzt
[Mon] = 0x00	Keine Monats Prüfung, Bit 7/0x80 nicht gesetzt
[Jahr] = 0x00	Keine Jahresprüfung, Bit 7/0x80 nicht gesetzt

Einen oder mehrere Trigger auslösen

Befehl [0x16], [Trigger Nummer 0-79], [Trigger Nummer 0-79], ...

Ein Trigger wird ausgelöst wie wenn eines seiner programmierten Ereignisse eingetreten wäre.

Beim DMXfaceXH(n) PRO stehen 80 Trigger zur Verfügung, sonst 64.

Antwort: 2 Byte

Command Reflection [0x96] und [Trigger Nummer]

Bei mehreren Abrufen in einem Command steht an letzter Stelle die letzte Trigger Nummer die abgerufen wurde.

Basis Protokoll Ausgänge / OUTPUTS

OUTPUT Wert setzen

Befehl: [0x4F], [0x6F], [Output 1-24], [Wert]

Setzt den Output auf den angegeben Wert 0x00 = Aus oder 0x01-0xFF = Ein

Antwort: 2 Bytes

Command Reflection [0xCF], [0x00]

OUTPUT Wert setzen (auch mehrere gleichzeitig)

Befehl [0x76], [OUTPUT n], [WERT x], [OUTPUT n], [WERT x],

[OUTPUT n]: Output Nummer ab 0 gezählt, max. 23 Outputs

[WERT x]: 0x00-0xFF, 0= Ausschalten 0x01-0xFF= Einschalten

Befehlslänge 3 – 73 Bytes

Antwort 2Byte:

Command Reflection 0xF6, 0x00

OUTPUT Name abfragen

Befehl: [0x4F], [0x4F], [Output 1-24], [0x4E]

Liefert den Namen des Outputs der zugewiesen wurde. (DMXface Console, Text and Overview)

Antwort: 34 Bytes

[32 Bytes Name des Output]

Command Reflection [0xCF], [0x00]

OUTPUT Wert abfragen

Befehl: [0x4F], [0x4F], [Output 1-24], [0x52]

Liefert den aktuellen Schaltzustand eines Outputs.

Antwort: 3 Byte

[Output Status 0x00 oder 0x01]

Command Reflection [0xCF], [0x00]

Basis Protokoll Eingänge / INPORTS

INPORT Name abfragen

Befehl: [0x4F], ['I'], [Inport1-56], ['N']

Liefert den Namen des Inport der vergeben wurde. (DMXface Console, Text and Overview)

Inport 1-8 sind die am DMXface lokalen Eingänge

Inport 9-24 sind die Eingänge an einer DMXface IO Extension

Inport 25-56 entsprechen den BUS Eingängen BUS1 – BUS 32

Antwort: 34 Bytes

[32 Byte Name des Inport]

Command Reflection [0xCF], [0x00]

INPORT inkl. Tabellenkonversionswerten abfragen

Befehl: [0x1B], [Inport Nummer 1-56]

Fragt den Inport Zustand ab und liefert den digitalen Zustand, den Analogwert, und den in einen Text konvertierten Wert mit 0x00 Terminierung zurück. Ist dem Inport eine Tabellenkonversion zugewiesen so beinhaltet der Textbereich das Ergebnis der Tabellenübersetzung mit einer 0x00 Terminierung.

Antwort: min. 8 Bytes

[Inport Digital], [Inport Analog], [Inport Text], [0x00]

Command Reflection [0x9B], [Inport Nummer]

[Inport Digital] Wert 0x00 oder 0x01

[Inport Analog] Wert 0x00 bis 0xFF entsprechend der anliegenden Spannung 0-5.1 Volt

[Inport Text], [0x00] z.B. der analoge Wert 0x39 (Dez. 57) wird als [0x35], [0x37], [0x20], [0x00] eingesetzt. Der Wert 0x00 steht als Endzeichen nach dem Ende des Strings.

BUS INPORT auf einen Wert setzen

Befehl: [0x1C], [Inport 0-55]

Inport Nummern 0-7 Eingänge lokal am DMXface, das Setzen des Eingangs wird sofort mit dem aktuell am DMXface anliegenden Wert überschrieben.

Inport Nummern 8-23 DMXface IO Extension. Wenn kein Erweiterungsmodul angeschlossen ist, können die Eingänge auch beschrieben werden.

Inport Nummern 24-55 BUS1 bis BUS 32
Sofern keine KNX oder CAN Open Erweiterung angeschlossen ist die auf die Eingänge schreibt, sind diese verwendbar.

Antwort 2 Bytes: *Command Reflection [0x9C], [Inport Nummer]*

INPORT Analogwerte der ersten 8 Inports (lokale DMXface Eingänge)

Befehl: [0x61], [0x00]

Antwort: 10 Bytes

AD Wert INPORT 1	AD Wert 0x00 – 0xFF	Byte 1
AD Wert INPORT 2	AD Wert 0x00 – 0xFF	Byte 2
AD Wert INPORT 3	AD Wert 0x00 – 0xFF	Byte 3
AD Wert INPORT 4	AD Wert 0x00 – 0xFF	Byte 4
AD Wert INPORT 5	AD Wert 0x00 – 0xFF	Byte 5
AD Wert INPORT 6	AD Wert 0x00 – 0xFF	Byte 6
AD Wert INPORT 7	AD Wert 0x00 – 0xFF	Byte 7
AD Wert INPORT 8	AD Wert 0x00 – 0xFF	Byte 8
Command Reflection	0xE1,0x00	Byte 9+10

INPORT Analogwerte 24 Kanäle (DMXface + IO Extension)

Befehl: [0x61], [0x01]

Antwort: 26 Bytes

AD Wert INPORT 1	AD Wert 0x00 – 0xFF	Byte 1
AD Wert INPORT 2	AD Wert 0x00 – 0xFF	Byte 2
AD Wert INPORT 6	AD Wert 0x00 – 0xFF
AD Wert INPORT 23	AD Wert 0x00 – 0xFF	Byte 23
AD Wert INPORT 24	AD Wert 0x00 – 0xFF	Byte 24
Command Reflection	0xE1,0x01	Byte 25+26

INPORT Analogwerte der BUS Eingänge (CAN OPEN Erweiterung)

Befehl: [0x61], [0x02]

Antwort: 34 Bytes

AD BUS 1.1	AD Wert 0x00 – 0xFF	Byte 1
AD BUS 1.2	AD Wert 0x00 – 0xFF	Byte 2
.....	
AD Bus 1.8	AD Wert 0x00 – 0xFF	Byte 8
AD Bus 2.1	AD Wert 0x00 – 0xFF	Byte 9
AD Bus 2.2	AD Wert 0x00 – 0xFF	Byte 10
.....	
AD Bus 2.8	AD Wert 0x00 – 0xFF	Byte 16
.....	
AD Bus 4.8	AD Wert 0x00 – 0xFF	Byte 32
Command Reflection	0xE1,0x02	Byte 33+34

INPORT alle Analogwerte abfragen 24 lokale + 32 BUS Eingänge

Befehl: [0x61], [0x03]

Antwort: 58 Bytes

AD Wert INPORT 1	AD Wert 0x00 – 0xFF	Byte 1
AD Wert INPORT 2	AD Wert 0x00 – 0xFF	Byte 2
AD Wert INPORT 6	AD Wert 0x00 – 0xFF
AD Wert INPORT 23	AD Wert 0x00 – 0xFF	Byte 23
AD Wert INPORT 24	AD Wert 0x00 – 0xFF	Byte 24
AD BUS 1.1	AD Wert 0x00 – 0xFF	Byte 25
AD BUS 1.2	AD Wert 0x00 – 0xFF	Byte 26
.....	
AD Bus 1.8	AD Wert 0x00 – 0xFF	Byte 31
AD Bus 2.1	AD Wert 0x00 – 0xFF	Byte 32
AD Bus 2.2	AD Wert 0x00 – 0xFF	Byte 40
.....	
AD Bus 2.8	AD Wert 0x00 – 0xFF	Byte 17
.....	
AD Bus 4.8	AD Wert 0x00 – 0xFF	Byte 56
Command Reflection	0xE1,0x03	Byte 57+58

Basis Protokoll Status und Portabfragen

Abfrage des aktuellen Gerätestatus

Befehl: [0x43]

Antwort: 14 Bytes

[Letzte aufgerufene Szene / Programm]	Wert 0x01-0xFE	Byte 1
[Inport Byte, Status Inport 1-8]	Wert 0x00-0xFF	Byte 2
[Outputport Byte, Status Outputport 1-8]	Wert 0x00-0xFF	Byte 3
[Timer Status Byte, aktive Timer 1-8]	Wert 0x00-0xFF	Byte 4
[Timer Disable Byte, Status]	siehe Allgemeines	Byte 5
[DMX Mode Status Byte]	siehe Allgemeines	Byte 6
[Letzte aufgerufene Szene durch Trigger]	Wert 0x01-0xFE	Byte 7
[Letzte aufgerufene Szene durch Programm]	Wert 0x01-0xFE	Byte 8
[Inport Byte, Status Inport 17-24]	Wert 0x00-0xFF	Byte 9
[Inport Byte, Status Inport 9-16]	Wert 0x00-0xFF	Byte 10
[Outputport Byte, Status Outputport 9-16]	Wert 0x00-0xFF	Byte 11
[DMX MASTER WERT]	Wert 0x00-0xFF	Byte 12
[COMMAND RELFECTION]	0xC3	Byte 13
	0x00	Byte 14

Abfrage von 16 wählbaren DMX OUT Kanälen und dem Gerätestatus

[0x51], [DMX Startkanal low Byte], [\[optional DMX Startkanal High Byte\]](#)

DMX Startkanal wird ab 0 übergeben d.h. 0x00 entspricht DMX Kanal 1

Startkanal Wertebereich 0-255 bei 8 Bit Übergabe

Startkanal Wertebereich 0-527 bei 16 Bit Übergabe

Antwort 30 Bytes

[DMX s]	Wert DMX Out Startkanal	Wert 0x00-0xFF	Byte 1
[DMX s+1]	Wert DMX Out Startkanal +1	Wert 0x00-0xFF	Byte 2
.....
[DMX s+15]	Wert DMX Out Startkanal +15	Wert 0x00-0xFF	Byte 16

Folgende Bytes sind gleich wie bei der Abfrage [0x43] oder [,C']

[Letzte aufgerufene Szene / Programm]	Wert 0x01-0xFE	Byte 17
[Inport Byte, Status Inport 1-8]	Wert 0x00-0xFF	Byte 18
[Outputport Byte, Status Outputport 1-8]	Wert 0x00-0xFF	Byte 19
[Timer Status Byte, aktive Timer 1-8]	Wert 0x00-0xFF	Byte 20
[Timer Disable Byte, Status]	siehe Allgemeines	Byte 21
[DMX Mode Status Byte]	siehe Allgemeines	Byte 22
[Letzte aufgerufene Szene durch Trigger]	Wert 0x01-0xFE	Byte 23
[Letzte aufgerufene Szene durch Programm]	Wert 0x01-0xFE	Byte 24
[Inport Byte, Status Inport 17-24]	Wert 0x00-0xFF	Byte 25
[Inport Byte, Status Inport 9-16]	Wert 0x00-0xFF	Byte 26
[Outputport Byte, Status Outputport 9-16]	Wert 0x00-0xFF	Byte 27
[DMX MASTER WERT]	Wert 0x00-0xFF	Byte 28
[COMMAND RELFECTION]	0xD1	Byte 29
[DMX Startkanal der Abfrage]	0x00-0xFF	Byte 30

Portstatus + wählbare Anzahl von DMX OUT Kanälen ab Kanal 1 abfragen

Befehl [0x14] [n = Anzahl der DMX OUT Kanäle zum Übertragen max. 255 Kanäle]

Alle Inport und Outport Zustände werden übertragen.

Immer 8 Ports sind in einem Byte Wert Bitcodiert enthalten.

zusätzlich auch eine wählbare Anzahl von DMX OUT Kanälen.

Antwort: 34 Bytes (0 DMX Kanäle) bis 258 Bytes (224 DMX Kanäle)

Reserve	0x00	Byte 1
Reserve	0x00	Byte 2
OUTPORT 9-16	0x00-0xFF	Byte 3
OUTPORT 1-8	0x00-0xFF	Byte 4
Reserve	0x00	Byte 5
Reserve	0x00	Byte 6
Reserve	0x00	Byte 7
Reserve	0x00	Byte 8
BUS INPORT 4.1-4.8	0x00-0xFF	Byte 9
BUS INPORT 3.1-3.8	0x00-0xFF	Byte 10
BUS INPORT 2.1-2.8	0x00-0xFF	Byte 11
BUS INPORT 1.1-1.8	0x00-0xFF	Byte 12
Reserve	0x00	Byte 13
INPORT 17-24	0x00-0xFF	Byte 14
INPORT 9-16	0x00-0xFF	Byte 15
INPORT 1-8	0x00-0xFF	Byte 16
AD Wert INPORT 1	0x00-0xFF	Byte 17
AD Wert INPORT 2	0x00-0xFF	Byte 18
AD Wert INPORT 3	0x00-0xFF	Byte 19
AD Wert INPORT 4	0x00-0xFF	Byte 20
AD Wert INPORT 5	0x00-0xFF	Byte 21
AD Wert INPORT 6	0x00-0xFF	Byte 22
AD Wert INPORT 7	0x00-0xFF	Byte 23
AD Wert INPORT 8	0x00-0xFF	Byte 24
AD Wert INPORT 9	0x00-0xFF	Byte 25
AD Wert INPORT 10	0x00-0xFF	Byte 26
AD Wert INPORT 11	0x00-0xFF	Byte 27
AD Wert INPORT 12	0x00-0xFF	Byte 28
AD Wert INPORT 13	0x00-0xFF	Byte 29
AD Wert INPORT 14	0x00-0xFF	Byte 30
AD Wert INPORT 15	0x00-0xFF	Byte 31
AD Wert INPORT 16	0x00-0xFF	Byte 32
DMXOUT 1 Kanalwert	0x00-0xFF	Byte 33
DMXOUT 2 Kanalwert	0x00-0xFF	Byte 34
DMXOUT 3 Kanalwert	0x00-0xFF	Byte 35
DMXOUT n Kanalwert	0x00-0xFF
Command Reflection	0x94, n	Letzte 2 Bytes

Basis Protokoll Echtzeituhr lesen oder schreiben

RTC Einstellung lesen

Befehl [0x5A], [0x52]

Antwort: [Sek], [Min], [Std], [WT], [Tag], [Mon], [Jahr], [0x00], [SZ AUTO], [0xDA], [0x52]
Command Reflection

RTC Einstellung schreiben

Befehl: [0x5A], [0x53], [Sek], [Min], [Std], [Wt], [Tag], [Mon], [Jahr], [0x00], [SZ AUTO]

Antwort: [0xDA], [0x52]

Der Uhrzeitchip am DMXface gibt die Dezimalwerte der Uhrzeit als HEX Byte aus

Die Zählung läuft wie folgt:

0x00, 0x01, 0x02, 0x03, 0x04, 0x05, 0x06, 0x07, 0x08, **0x09, 0x10**, 0x11, 0x12.....

Die ist bei allfälligen Umrechnungen zu berücksichtigen.

Echtzeituhr Bytes:

[Sek]	Wert 0x00 bis 0x59	Hex als Dezimal auswerten	Byte 1
[Min]	Wert 0x00 bis 0x59	Hex als Dezimal auswerten	Byte 2
[Std]	Wert 0x00 bis 0x23	Hex als Dezimal auswerten	Byte 3
[WT]	Wert 0x01	Montag	Byte 4
	Wert 0x02	Dienstag	
	Wert 0x03	Mittwoch	
		
	Wert 0x07	Sonntag	
[Tag]	Wert 0x1 bis 0x31	Hex als Dezimal auswerten	Byte 5
[Mon]	Wert 0x1 bis 0x12	Hex als Dezimal auswerten	Byte 6
[Jahr]	Wert 0x00 bis 0x99	Hex als Dezimal auswerten	Byte 7
[0x00]	Wert immer 0x00		Byte 8
[SZ AUTO]	Automatische Sommerzeitumstellung		Byte 9
	Bit 0 gesetzt = automatische Sommerzeitumstellen aktiv		
	Bit 7 (bei der Abfrage) = 1, aktuelle Zeit ist Sommerzeit		

Weitere Befehle

Interface Versionsstring abfragen

Befehl [0x56]

Antwort:

String mit Version und den Features des DMXface (variable Länge)

+ Command Reflection 2Bytes 0xD6 und 0x00

Fehler Status Register abfragen

Befehl [0x73]

Antwort: 8 Bytes

[ERR_SYSTEM]

0x01 PLL Start Fehler
 0x02 LCD Data ERR
 0x04 LCD CMD ERR
 0x10 I2C Timeout RTC
 0x20 EE Data set to default
 0x40 SIC CHIP timeout Uh3
 0x80 SIC CHIP timeout Uh4

[ERR_SCIBUFF]

0x01 DMX Buffer Overflow
 0x02 RS485 Buffer Overflow
 0x04 USB Buffer Overflow
 0x10 RS232 Buffer Overflow
 0x20 MIDI / DALI Buffer Overflow

[ERR_SCIDATA]

0x01 DMX Data Error
 0x02 RS485 Data Error
 0x04 USB Data Error
 0x10 RS232 Data Error
 0x20 MIDI / DALI Data Error

[FLASH WRITES Bit 16-23]

[FLASH WRITES Bit 8-15]

[FLASH WRITES Bit 0-7]

Anzahl der durch Programme verursachten Flash Writes

[Command Reflection]

0xF3

0x00

Befehle die zwar noch unterstützt aber nicht mehr empfohlen werden.

DMX OUT Kanalwert abfragen (Kanal 1-255)

Befehl: [0x4F], [0x44], [DMX OUT Kanal ab 1], [0x52]

Antwort: min. 7 Bytes

[DMX WERT], [TEXT WERT min 4 Bytes], *Command Reflection* [0xCF], [0x00]

Ist der DMX Kanal keiner Tabelle zugewiesen, dann wird der DMX Wert in den Textbereich mit 3 Zeichen + Terminierung 0x00 eingefügt.

z.B. beim DMX Wert 127

[0x7F], [0x31], [0x32], [0x37], [0x00], [0xCF], [0x00]

Ist dem DMX Kanal eine Tabellen Konversion zugeordnet so wird die komplette Ausgabe aus der Umrechnung in den Textbereich eingefügt.

DMX OUT auf einen Kanalwert setzen (Kanal 1-255)

Befehl: [0x4F], [0x64], [DMX OUT Kanal ab 1], [Wert 0-255]

Setzt den angegebenen DMX Kanal auf den Wert

Antwort: *Command Reflection* [0xCF], [0x00]

DMX OUT auf einen Kanalwert setzen (auch mehrere in einem Befehl, Kanal 1-256)

DMX Kanal Bereich bis max. 256 (alternativ siehe DMX16 Befehl)

Befehl: [0x4B], [Kanal], [Wert], [Kanal], [Wert], [Kanal], [Wert], [Kanal].....

Es können ein oder mehrere DMX Out Kanäle auf einen neuen Wert gesetzt werden.

[Kanal] DMX Kanal ab 0 gezählt 0-255 für die Kanäle 1-256 oder höchstem im DMXface verfügbaren Kanal

[Wert] Neuer DMX Wert für den Kanal

Befehlsängen bis zu 512 Bytes möglich

Antwort: 2 Bytes *Command Reflection* [0xCB], [0x00]

DMX OUT Kanal inkl. Tabellenkonversionswerten abfragen (Kanal 1-255)

Siehe DMX16 Befehl für den Zugriff auf die Kanäle >255

Befehl: [0x1D], [DMX Kanal Nummer 1-255]

Fragt Wert eines DMX OUT Kanals ab und liefert den aktuellen Wert und den in einen Text konvertierten Wert mit 0x00 Terminierung zurück. Ist dem DMX Kanal eine Tabellenkonversion zugewiesen so beinhaltet der Textbereich das Ergebnis der Tabellenübersetzung mit einer 0x00 Terminierung.

Antwort: min. 7 Bytes

[DMX Wert], [DMX Wert Text,0x00], Command Reflection [0x9B], [Inport Nummer]

[DMX Wert] Wert 0x00 bis 0xFF

[DMX Text,0x00] z.B. der DMX Wert 0x64 (Dez. 100) wird als [0x31], [0x30], [0x30], [0x00] eingesetzt.

DMX OUT alle Kanäle abfragen

Befehl: [0x44]

Fragt alle 224 oder 512 (PRO Version) DMX OUT Kanäle am Interface ab.

Beindet sich das DMXface im DMX Read Mode, dann werden die Werte vom DMX EINGANG geliefert.

DMX Read Mode wird durch den Umschaltbutton an der DMXface Console aktiviert und ist nur aktiv bis zum Neustart des Interface oder bis zum Schließen der DMXface Console.

Antwort:226 oder 514 Bytes

[224 oder 512 Bytes DMX Kanal Werte], Command Reflection [0xC4], [0x00]

DMX IN eine Anzahl von DMX Kanäle abfragen

Befehl [0x17], [DMX IN Startkanal low], [Anzahl der Kanäle low],

[optional DMX IN Startkanal high], [optional Anzahl der Kanäle high]

[DMX IN Startkanal] Erste Kanalnummer die abgefragt wird, Zählung beginnt bei 0 (=DMX Kanal1)

[Anzahl der Kanäle] Wertebereich 1 - Anzahl der unterstützen DMXface Kanäle (max. 224)

Optional können auch größere DMX Kanalnummern und Abfragebereiche durch Ausgabe der High Bytes abgefragt werden.

Befehlslänge 3,4 oder 5 Bytes

Antwortlänge Anzahl der Kanäle + 2 Bytes

[Wert DMX IN Kanal Startkanal]

[Wert DMX IN Kanal Startkanal+1]

[Wert DMX IN Kanal Startkanal+2]

.....

[Wert DMX IN Kanal Startkanal + Anzahl]

[Command Reflection 0x97], [Startkanal Low Byte]

DMX OUT Kanal auf einen Wert setzen mit Rückantwort als Text (Kanal 1-255)

DMX Kanal Bereich bis max. 255 (alternativ siehe DMX16 Befehl)

Befehl: [0x40], [Kanal 1-255], [Wert]

Es wird ein DMX OUT Kanal auf den Wert gesetzt und als Rückantwort der aktuelle DMX Wert mit einem Text gesendet. Ist der DMX Kanal einer Tabellenkonversion zugewiesen so beinhaltet der Text das Ergebnis aus der Tabellenkonvertierung

[Kanal] DMX Kanal ab 1 gezählt 1-255 oder höchstem im DMX verfügbaren Kanal

[Wert] Neuer DMX Wert für den Kanal

Antwort: min. 7 Bytes

[DMX Wert], [DMX Text + 0x00], Command Reflection [0xC0], [Kanal Nummer]

Beispiel:

Befehl [0x40], [0x01], [0xC8] setzt den DMX Kanal 1 auf den Wert 200

Antwort des DMXface:

[0xC8], [0x32], [0x30], [0x30], [0x00], [0xC0], [0x01]

Text: ,2' ,0' ,0' Terminierung

INPORT Wert abfragen

Befehl: [0x4F], [0x49], [Inport1-56], [0x52]

Befehl: [0x4F], ['I'], [Inport1-8], ['R']

Antwort: min. 7 Bytes

[Digitalwert], [Analogwert], [TEXT WERT min 4 Bytes], Command Reflection [0xCF], [0x00]

Ist der Eingang keiner Tabelle zugewiesen, dann wird der Analogwert in den Textbereich mit 3 Zeichen + Terminierung 0x00 eingefügt.

z.B. beim Analogwert Wert 57

[0x00], [0x39], [0x35], [0x37], [0x20], [0x00]

Command Reflection [0xCF], [0x00]

Ist dem Eingangskanal eine Tabellen Konversion zugeordnet so wird die komplette Ausgabe aus der Umrechnung in den Textbereich eingefügt.