

USB / DMX512 Generator

Betriebsanleitung für ab Mai 2009 gelieferte Geräte

Status 25. Juli 2009

Einsatzbereiche:

Der USB / DMX512 Generator ist konzipiert um einen digitalen Lichtsteuerbus über eine USB-Schnittstelle mit kurzen ASCII-Texten zu steuern.

Die DMX-Steuersequenz kann in ihrem gesamten Umfang von 512 Bytes gesendet werden. Die Sequenzlänge kann aber auch nach Bedarf verkürzt werden, wodurch sich eine höhere Wiederholgeschwindigkeit ergibt.

Der USB / DMX Generator kann für jeden beliebigen DMX-Offset einen kompletten Fade-Vorgang ausführen - ausgelöst durch einen kurzen Textbefehl. Es können auf allen DMX-Kanälen gleichzeitig Fade-Vorgänge ausgeführt werden. Hierauf basierend ist auch beim Laden von Presets (Lichtszenen) das Überblenden auf die neue Szene mit einer Fade-Zeit zwischen 0 und 31,9 Sekunden möglich.

Bis zu 115 Presets können fest abgespeichert und mit einem kurzen ASCII-Befehl aktiviert werden. Auf diese Art können z.B. Lichtszenen unproblematisch abgerufen werden.

Großer Wert wurde darauf gelegt, dass **jeder Steuerbefehl als möglichst kompakter ASCII-Text dargestellt werden kann**, der sich gut sowohl zur manuellen Eingabe per Tastatur als auch zur automatischen Erzeugung / Auswertung in einem Anwenderprogramm, einer Medien- oder Industriesteuerung eignet.

Darüber hinaus ist **ein Satz kompakter binärer Befehle verfügbar zum schnellen Setzen des Werts beliebiger DMX-Kanäle bzw. Blocks von DMX-Kanälen** sowie anderer wichtiger Systemparameter. Das gesamte Kommunikationsprotokoll ist offengelegt.

Das Gerät ist **NICHT zugelassen** für sicherheitskritische Anwendungen, bei denen Personen gefährdet werden oder nennenswerter Sachschaden entstehen könnte !

Hardware

Elemente an der Frontseite:



Links an der Frontplatte befindet sich die **USB – Buchse**:

Hier wird mit einem handelsüblichen USB- Kabel ("Typ A"-Stecker zu "Typ B"-Stecker) die Verbindung zum Steuer-Rechner hergestellt. Zur Datenkommunikation wird auf dem Steuer-Rechner eine virtuelle serielle Schnittstelle installiert. Der USB / DMX Generator funktioniert baudraten-unabhängig, dh. mit beliebiger am PC eingestellter Baudrate. Neben der Datenübertragung erfolgt auch die Stromversorgung des USB / DMX Generators über das USB-Kabel (5 Volt, max 100mA).

Die grüne **Leuchtdiode** zeigt das Vorhandensein der Betriebsspannung an. Bei Kommando-Transfer auf der Steuerschnittstelle erlischt die LED kurzzeitig.

DMX512 - Ausgangsbuchse rechts an der Frontplatte:

Die DMX-Schnittstelle ist ausgelegt nach den Empfehlungen des Standard DMX512-A:
5-polige XLR Einbaubuchse (optional 3-polige XLR-Einbaubuchse)

DMX OUT	XLR Buchse
Abschirmung,Signalmasse	Pin 1
DMX- Sender (TX-)	Pin 2
DMX+ Sender (TX+)	Pin 3

Der DMX-Sender ist nicht galvanisch getrennt, also zwangsläufig über das USB-Kabel auch mit der PC-Masse und im Normalfall auch mit dem Schutzleiter der 230V-Hausinstallation verbunden.

Inbetriebnahme des USB / DMX512 Generators

Sicherheitshinweis: Da die Datenübertragung via DMX512 informatorisch als "unzuverlässig" einzustufen ist, **ist es ausdrücklich VERBOTEN, den USB / DMX512 Generator einzusetzen bei allen sicherheitskritischen Anwendungen**, bei denen es im Fehlerfall zu Personen- oder größeren Sachschäden kommen könnte !

Vor der eigentlichen Inbetriebnahme müssen auf dem Steuer-PC geeignete **USB-Treiber installiert** werden. Dazu wird der USB / DMX Generator ohne angeschlossene DMX-Busleitung mit einem USB-Kabel ("Typ A"-Stecker zu "Typ B"-Stecker) an den Steuer-PC angeschlossen. Neben der Datenübertragung erfolgt auch die Stromversorgung des USB / DMX Generators über das USB-Kabel (5 Volt, max 100mA). In vielen Fällen kann der USB / DMX Generator über einen passiven Hub angeschlossen werden (z.B. USB-Buchse an der Tastatur), arbeitet jedoch dann an der Grenze der Stromversorgungskapazität. Für maximale Betriebssicherheit und Stabilität des DMX-Signals empfehlen wir daher, das Gerät direkt an eine USB-Buchse am Steuer-Rechner anzuschließen.

Im Normalfall wird der USB / DMX Generator vom PC über eine "virtuelle" serielle Schnittstelle angesteuert, dieser Fall wird nachfolgend beschrieben. Manche Anwendungsprogramme verwenden aber auch andere USB-Ansteuerungstechniken. Im Zweifelsfall sollte hierzu die Dokumentation der vorgesehenen Steuer-Anwendung studiert werden.

Auf dem Bildschirm wird (je nach verwendetem Betriebssystem) eine Meldung erscheinen, dass eine neue Hardware-Erweiterung gefunden wurde. Dabei wird in der Regel nachgefragt, von welcher Quelle die dazu erforderlichen Treiber kopiert werden sollen.

Der USB / DMX Generator ist mit USB-Chips FT245BL des Herstellers FTDI ausgerüstet. Die **mitgelieferte Support-CD** (Verzeichnis "USB") **enthält "VCP"** (= Virtual Com Port)- **Treiber** für gängige Windows-Betriebssysteme ab Windows 2000. Treiber für andere Betriebssysteme oder evtl neuere Versionen müssen von der FTDI Webseite (www.ftdichip.com) heruntergeladen und dekomprimiert werden.

Die Installation der Treiber erfolgt entsprechend dem Bildschirm -Dialog.

Auf folgende **Spezial-Situation** wird hingewiesen: Die USB-Charakteristik des USB / DMX Generators wird durch ein auf der Platine steckbares EEPROM "individualisiert". Er bekommt z.B. eine eigene Port-Nummer zugewiesen - dadurch ist er von anderen USB-Geräten mit gleichartigen USB-Chips unterscheidbar. In PC-Konfigurationen, bei denen abwechselnd verschiedene externe Geräte mit FTDI-Chips angeschlossen werden, kann die hohe Zahl reservierter Port-Nummern zu Problemen oder umständlicher Handhabung führen. In dem Fall empfehlen wir, das Individualisierungs-EEPROM auszubauen: USB-Kabel trennen, 2 Schrauben am Gehäuseboden lösen, oberes Gehäuseteil abziehen und das EEPROM (8-polig mit Lack-Markierung) aus dem Sockel zu ziehen. Alternativ kann mit dem

von FTDI zur Verfügung gestellten Programm "MPROG" das EEPROM via USB ohne Demontage des Geräts gelöscht und ggf. später neu beschrieben werden. Hierzu vorher unbedingt die Dokumentation von "MPROG" studieren !

Windows-98 Rechner sind zunächst nach Anschluss des USB / DMX Generators etwa für 30 Sekunden blockiert. Ausserdem sollte nach dem Entfernen des USB Kabels der Generator möglichst erst nach ca. 30 Sekunden wieder angeschlossen werden, weil sonst die USB-Verbindung gelegentlich erst nach Neustart des Rechners wieder hergestellt werden kann.

Der USB / DMX Generator arbeitet baudraten-unabhängig zwischen 2400 und 115200 Baud. Daher kann eine beliebige Standard-Baudrate am Steuerprogramm eingestellt sein.

Allerdings **muss** folgendes Datenformat eingehalten werden: **8Bit Daten, KEIN Paritätsbit, 1 Stopbit (= 8N1)**. Handshake wird vom USB / DMX Generator nicht unterstützt und ist dank interner Pufferspeicher und hoher Verarbeitungsgeschwindigkeit auch nicht notwendig.

Im nächsten Schritt wird das DMX-Equipment untereinander verkabelt. Dabei sollte die Betriebsspannung aller Geräte zunächst ausgeschaltet sein. Die Adressierschalter an allen Dimmern etc. werden so eingestellt, dass sich eine dem Projekt entsprechende sinnvolle Anordnung der DMX-Adressen ergibt. Der 5-polige Stecker des DMX-Kabels wird in die 5-polige Buchse "DMX512" and der Frontplatte des Generators gesteckt.

Falls man sich nicht sicher ist und keine DMX-Erfahrung hat, empfehlen wir zum Start einen einzigen DMX-Dimmer mit Lampe für DMX-Kanal Nr. 1 zu adressieren und anzuschliessen.

Nach Anschluss des USB-Kabels wird die LED an der Frontplatte grün leuchten. Alle DMX-Ausgangskanäle sollten bei einem neuen Gerät auf Pegel "0" gesetzt sein. War das Gerät bereits in Betrieb, sollten sich der DMX-Ausgang entsprechend einem evtl. gespeicherten Preset Nr.0 einstellen.

Für einen ersten Funktionstest empfiehlt es sich, den USB / DMX Generator mit einem Terminalprogramm anzusteuern (z.B. mit „Hyperterm“, das mit Windows installiert wird).

Um eine Befehlszeile zu beenden erwartet der DMX Generator nur ein "Carriage Return" (=dez13 =hex D). Als Bestätigung sendet er dieses Zeichen zurück. "Line Feed" wird empfangsseitig ignoriert und niemals gesendet. Wenn der Cursor auf dem PC-Bildschirm stets an den Anfang der vorherigen Zeile springt, muss das Terminalprogramm so konfiguriert werden, dass es beim Datenempfang selber ein "Line Feed" einfügt.

Auf dem Terminalprogramm sollte die Meldung "DMX Generator" erscheinen. Wird die "Return" Taste am PC gedrückt, sollte der DMX Generator die vorherige Bildschirmzeile mit einem "Carriage Return" (=dez13 =hexD) die nächste Kommando-Zeile eröffnen. Weitere Tips für die ersten Schritte siehe folgende Beschreibung "Schnellstart und elementare Befehle".

ASCII-Protokoll des USB / DMX512 Generators

Zur Steuerung des USB / DMX Generators sind zwei Steuerprotokolle verfügbar: das ASCII-Protokoll und das binäre Protokoll (siehe Teil 2 des Manuals). Beide Varianten sind gegeneinander transparent, d.h. es ist keine explizite Modus-Umschaltung erforderlich, die Befehle können untereinander gemischt werden.

Zum Betrieb zusammen mit PC-Anwendungen oder anderen USB Host-Controllern mit geringen Möglichkeiten der Datenumwandlung erweist sich das weiter unten beschriebene binäre Protokoll häufig als geeigneter. Darüber hinaus ist es kompakter und schneller, erfordert aber zu seiner Handhabung steuerungsseitig häufig professionelle Programmiertechniken.

Hingegen kann das ASCII-Protokoll auch per Hand mit einem Terminalprogramm ausgeführt werden, was wir zum Kennenlernen des USB / DMX Generators oder zum DMX-Bus Debuggen auch empfehlen.

Jeder Steuerbefehl und jede Zustandsmeldung beginnt mit einem einzelnen charakteristischen Buchstaben, danach folgt, wenn notwendig, eine Zahl als Wert-Parameter. Alle Zeichen werden unabhängig von Groß- und Kleinschreibung interpretiert. Rückmeldungen kommen immer in Großbuchstaben.

Schnellstart und elementare Befehle:

Die **Nummer des zu verändernden DMX-Kanals** (1 - 512) wird eingestellt mit dem Befehl "S" (spricht das interne SLOT-Register an, **siehe Anhang A**), danach folgt die DMX-Kanal-Nummer als 1- bis 3-stellige Dezimalzahl.

Danach wird der an diesem DMX-Kanal **einzustellende DMX-Pegel** mit dem Befehl "V" (spricht das interne VALUE-Register an) übertragen, gefolgt von einer als ASCII-Text dargestellten Zahl im Bereich 0 bis 255: Folgt nicht unmittelbar darauf ein neuer Befehlscode, muss die Zahl dreistellig eingegeben werden oder bei weniger Stellen mit <return> abgeschlossen werden.

Beispiel: S1V45 <return>

ist gleichbedeutend mit **S001V045**

Wenn mit einem Befehl eine komplette **Überblendung** vom aktuellen DMX-Pegel zum neu gewünschten Endwert gestartet werden soll, **muss zuvor mit dem Befehl "T" die Fade-Zeit eingestellt werden** (Parameter in Sekunden, optional können getrennt durch einen Punkt Zehntelsekunden hinzugefügt werden).

Die eingestellte Fade-Zeit wirkt auf alle nachfolgenden Befehle bis sie geändert wird. Sie kann unmittelbar nach jedem "V"- Befehl geändert werden ohne Rückwirkung auf bereits erteilte Befehle. **Fade-Zeit = 0 heisst sofort auf Endwert schalten.**

Die maximale Fade-Zeit beträgt 31,9 Sekunden.

Beispiel: T3.5S1V045

Wenn ein Block aufeinanderfolgender DMX-Kanäle neu eingestellt werden soll, wird mit dem "," (Komma)-Befehl die DMX-Adresse automatisch um eins inkrementiert (erhöht), **bevor dort** die als Parameter angehängte Zahl als neuer DMX-Wert eingetragen wird.

Beispiel: S1V45,0,255,136....

modifiziert die DMX-Kanäle 1,2,3,4, und überblendet mit der eingestellten Fade-Zeit

Der Befehl "Q" stellt nichts ein, aber er informiert über die aktuellen Einstellungen des zuvor mit dem Befehl "S" adressierten DMX-Kanals. Typisches Beispiel einer Rückmeldung:

Tx: CH=1 OUT=13 TX=26 MF=50% CY=0 L=512 T=3.2

OUT ist der aktuelle Pegel für den derzeit adressierten DMX-Kanal CH.TX beschreibt den Inhalt des Sendepuffers dieses DMX-Kanals. Beide unterscheiden sich, wenn der globale Masterfader (Anzeige MF) nicht auf 100% eingestellt ist. CY gibt die aktuelle Länge der Lauflicht-Schleife ("Chaser") an, im Lieferzustand = 0. L gibt die aktuell eingestellte Länge des DMX-Zyklus an .T gibt die aktuell eingestellte Fade-Zeit für Überblendungen an (Sekunden.Zehntel).

Grundsätzlicher Aufbau der ASCII-Befehle:

Befehlscode Parameter [Befehlscode Parameter] <CR=hexD>

Bei jedem Befehl wird der mit dem Befehlscode verknüpfte Parameter in das Entsprechende der oben beschriebenen Register eingetragen und je nach Befehlstyp die entsprechende Aktion mit den dann vorhandenen Daten ALLER internen Register ausgeführt. Daher beeinflusst die REIHENFOLGE der Befehle u.U. die ausgeführte Aktion. Generell **sollte die Reihenfolge FADETIME, SLOT, VALUE**

eingehalten werden, wobei aber oft nicht alle Register neu beschrieben werden müssen. (Genauere Beschreibung der internen Struktur und der Register- Funktionen siehe Anhang A)

Jeder Befehl wird ausgeführt, sobald die notwendige Anzahl von ASCII-Zeichen empfangen wurde. Zahlen werden automatisch abgeschlossen, **sobald die im jeweiligen Kontext maximal mögliche Anzahl ASCII-Stellen empfangen wurde**. Bei kleineren Zahlen muss entweder einfach mit der Eingabe des nächsten Befehls begonnen werden, ein "Return" eingegeben werden oder die Zahl muss mit vorangestellten Nullen eingegeben werden

Beispiel: S1V23<return>

ist gleichbedeutend mit **S1V023** oder **S001V23Q**.

Parameter (d.h. Zahlenwerte) für SLOT, FADETIME und LOOP müssen grundsätzlich im **Dezimalformat** angegeben werden und werden auch in diesem Format zurück gemeldet. Der Parameter des **Masterfader** wird stets in Prozent eingetragen (ohne nachgestelltes % Zeichen) im Bereich 0 bis 200.

Lediglich die **Parameter für den gewünschten DMX-Pegel (VALUE)** können als **Dezimalzahl, Hexadezimalzahl oder in Prozent** angegeben werden (siehe "System-Befehle") und werden auch im einstellten Format zurück gemeldet.

VALUE wird intern immer als 8-Bit-Wert, d.h. als Zahl im Bereich 0 bis 255 gespeichert.

Wird das Prozentformat ausgewählt, dann werden die Prozentangaben (0 bis 100%) auf den Bereich 0-255 umgerechnet, in diesem Bereich werden auch alle internen Operationen des DMX Generators ausgeführt.

Daher liegt die im Prozentformat erzielbare Präzision stets deutlich unter dem technisch Möglichen.

Das Steuerzeichen für Sprung an Zeilenanfang "**CR=Carriage Return**" = **dez13** = **hexD** führt den vorherigen Befehl sofort aus und reorganisiert interne Strukturen des Kommandointerpreters. Das Zeilenvorschub-Zeichen "**LF=Line Feed**" = **hexA** = **dez10** wird ignoriert. Dies ist bei der Konfiguration von Anwendungs-Programmen zu beachten. Es können beliebig viele Befehle pro Kommando-Zeile gegeben werden. Obwohl nicht dringend erforderlich, erhöht ein gelegentlich eingefügter Zeilenumbruch die Systemstabilität.

Leerstellen werden nicht interpretiert, d.h. können zur grafischen Auflockerung verwendet werden.

Bemerkt man während der Eingabe einen **Tippfehler**, sollte der angefangene Befehl mit der Return-Taste gelöscht werden. Die Rücktaste ist nicht wirksam.

Rückmeldungen:

Das Gerät sendet kein "Echo" der empfangenen Daten. Falls gewünscht, am Anwendungsprogramm "lokales Echo" einstellen.

Jede korrekt ausgeführte und mit "Carriage Return" beendete Befehlszeile bestätigt das Gerät ebenfalls mit einem "Carriage Return". Wenn der Befehl (meistens wegen eines Syntax-Fehlers) nicht interpretiert werden kann, antwortet der Generator sofort mit '?' plus "Carriage Return". Wurden bereits weitere nachfolgende Befehle in den Kommandopuffer geschrieben, so werden diese gelöscht und müssen ggf. erneut eingegeben werden.

Zum Auslesen angeforderte Daten sendet der Generator unmittelbar nach Befehlseingabe (Q- oder Z- Befehl).

Jede angeforderte Datenausgabe wird ebenfalls mit einem "Carriage Return" beendet. Hieran können evtl eingesetzte Analyseprogramme das Ende eines syntaktisch zusammengehörenden Datenblocks erkennen.

Datenformat der Geräte-Kennung beim Einschalten bzw. Reset:

DMX Generator <CR>

Diese Meldung wird automatisch beim Einschalten oder nach einem Reset gesendet. Beim Einstecken des USB-Kabels kann die Meldung gelegentlich durch USB-interne Verzögerung verloren gehen. Die wiederholte spontane Sendung dieser Meldung während des Betriebs deutet auf eine ungenügende Stromversorgung oder auf das Gerät wirkende extrem starke Störimpulse hin.

Kurzreferenz aller ASCII-Befehle

Sn	DMX-Kanal (SLOT-Register) für spätere Aktion adressieren (n=1 bis 512)	S.6
Vn	DMX-Pegel @ SLOT einstellen (n=0 bis 255)	S.7
,n	(Komma) SLOT inkrementieren, dann DMX-Pegel @ SLOT einstellen (n=0 - 255)	S.7
=n	DMX-Block n Offsets ab SLOT+1 mit DMX-Pegel von SLOT füllen (n=1 bis 512)	S.7
+	DMX-Sendepuffer @ SLOT inkrementieren (um eins erhöhen)	S.8
-	DMX-Sendepuffer @ SLOT dekrementieren (um eins verringern)	S.8
^n	addiert n zum DMX Preload Puffer @ SLOT (n=0 bis 255)	S.8
_n	subtrahiert n vom DMX Preload Puffer @ SLOT (n=0 bis 255)	S.8
\$	ab jetzt DMX-Pegel in HEX (gilt nur für V , Komma , ^, _ , R , Z und Q- Befehl)	S.8
&	ab jetzt DMX-Pegel DEZIMAL (nur für V , Komma , ^, _ , R , Z und Q- Befehl)	S.9
%	ab jetzt DMX-Pegel in PROZENT (nur V , Komma , ^, _ , R , Z und Q- Befehl)	S.9
(DMX-Sendepuffer nicht nachladen, im Hintergrund neue Lichtszene aufbauen	S.9
)	DMX-Sendepuffer wieder nachladen	S.9
Ts.t	FADETIME einstellen s=Sekunden t=Zehntel	S.9
!	alle Blendvorgänge abbrechen und auf momentanten Wert einfrieren	S.10
Zn	n Bytes ab DMX-Kanal = "SLOT" aus dem DMX-Sendepuffer auslesen	S.10
Mn	stellt den Masterfader ein: n=0 bis 200 (in Prozent, siehe Detailbeschreibung)	S.10
in	Schleife der Lauflichtfunktion einstellen (n=2 - 255, 0=SLOT Register ändern)	S.10
>	alle DMX-Werte in der Lauflichtschleife zum nächst höheren Kanal schieben	S.11
	alternativ: wenn i=0: SLOT inkrementieren (um 1 erhöhen)	
<	alle DMX-Werte in der Lauflichtschleife zum nächst niedrigeren Kanal schieben	S.11
	alternativ: wenn i=0: SLOT dekrementieren (um 1 verringern)	
Ln	DMX- Zykluslänge einstellen (n=24 bis 512)	S.11
Q	Inhalte aller DMX-Register @ SLOT anzeigen	S.11
~n	Sendepuffer und Konfiguration als Preset Nr. n sichern	S.11
@n	Preset Nr. n in Puffer und Konfiguration laden	S.12
 	alle Puffer und Konfiguration auf Lieferzustand zurückstellen	S.12
'	(Apostroph) Firmware-Version über USB ausgeben	S.12

Beschreibung aller ASCII-Befehle

Jeder Steuerbefehl und jede Zustandsmeldung ist mit einem einzigen, charakteristischen Buchstaben gekennzeichnet. Wenn ein Befehl einen Parameter erfordert, so ist dieser in eckigen Klammern <...> hinter dem Befehls-Buchstaben verzeichnet. Dieses kompakte Format eignet sich gut sowohl zur manuellen Eingabe per Tastatur als auch zur automatischen Erzeugung/Auswertung in einem Anwenderprogramm.

Zusätzlich gibt es einen Satz kompakter binärer Befehle zum schnellen Setzen des Werts beliebiger einzelner sowie kompletter Blocks von DMX-Kanälen, die im 2. Teil des Manuals beschrieben werden.

Adressierung des zu bearbeitenden DMX-Kanals (SLOT):

S <Kanal-Nummer>

Der Befehl **adressiert den DMX-Kanal <Kanal-Nummer>**, auf den viele der nachfolgend beschriebenen Befehle wirken. Intern wird dieser Wert im SLOT Register gespeichert.

Im DMX-Jargon wird oft das Wort "Slot" für DMX-Kanal gebraucht, da beim DMX-Sendevorgang jedem DMX-Kanal ein bestimmtes Zeitintervall (time slot) im Sendezyklus zugeordnet ist.

Parameter: DMX-Kanal (Bereich 1 bis 512), der mit nachfolgenden Befehlen manipuliert werden soll.

Anmerkung: Dieser Befehl löst keine unmittelbare Aktion aus. Aber nachfolgende Befehle werden auf den durch SLOT adressierten DMX-Kanal angewandt.

Beispiel: S127 schreibt 127 in Register SLOT

DMX-Pegel in Sendepuffer laden (mit Zwischenspeicherung im Preload-Puffer):

V <Pegel>

Parameter in den Preload-Puffer (und in den Sendepuffer) **für den DMX-Kanal = SLOT** eintragen.

Parameter: Pegel (Bereich 0 bis 255) ist der DMX-Pegel (z.B. Lampenhelligkeit), der nachfolgend im durch SLOT adressierten DMX-Kanal gesendet wird.

Anmerkung: Beim nächsten DMX-Resetimpuls wird der Preload-Puffer in den DMX-Sendepuffer übertragen unter Berücksichtigung einer mit FADETIME eingestellten Blendzeit und des globalen MASTERFADER Korrekturfaktors. Siehe auch Blockier-Befehl "(".

Wenn FADETIME gleich 0 ist, wird der durch SLOT adressierte Speicherplatz des Preload-Puffers sofort auf <Pegel> gesetzt.

Wenn FADETIME ungleich 0 ist, wird ein Fade-Prozess gestartet. Der Blendvorgang startet mit dem aktuellen Wert des mit SLOT adressierten Sendepuffers und endet, wenn dieser den Wert <Pegel> erreicht hat.

Beispiel: V34 stellt den Pegel des Sendepuffers im vorher mit SLOT adressierten DMX-Kanal auf 34. Der Parameter wird interpretiert in der vorher eingestellten Zahlenbasis (dezimal=Standardeinstellung, Hex oder Prozent).

, (Komma) <Pegel>

Zuerst inkrementiert dieser Befehl automatisch das SLOT-Register (addiert 1 hinzu), dann schreibt er den Parameter <Pegel> in das VALUE - Register und trägt ihn gleichzeitig in den Preload-Puffer für den DMX-Kanal = SLOT ein.

Parameter: Pegel (Bereich 0 bis 255) ist der DMX-Pegel (z.B. Lampenhelligkeit), der nachfolgend im durch SLOT (inkrementiert) adressierten DMX-Kanal gesendet wird.

Anmerkung: abgesehen davon, dass das SLOT- Register zuerst inkrementiert wird, hat der Komma-Befehl die gleiche Funktion wie der **V-Befehl**. Er eignet sich zur kompakten Einstellung eines Blocks aufeinander folgender DMX-Kanäle.

W <Pegel>

Dieser Befehl ist identisch mit obigem Befehl , (Komma). Er wird z. Zt. noch unterstützt zwecks Rückwärtskompatibilität mit älteren Firmware-Versionen. **Nicht empfohlen für Neuentwicklungen !**

= <Blocklänge>

Dieser Befehl schreibt den aktuellen Wert von DMX-Kanal = SLOT in <Blocklänge> DMX Kanäle ab dem mit (SLOT+1) adressieren Kanal in den Preload-Puffer und beim nächsten DMX-Reset-Impuls weiter in den Sendepuffer.

Parameter: Blocklänge (Bereich 1 bis 512) ist die Anzahl der DMX-Kanäle in die der gleiche DMX-Wert eingetragen wird. Unabhängig vom vorgegebenen Wert von Blocklänge wird DMX-Kanal 512 nicht überschritten. Die Überblendung vom jeweiligen Ist- Zustand jedes Kanals zum vorgegebenen End-Zustand wird auf allen angesprochenen DMX-Kanälen durch den aktuellen Wert von FADETIME bestimmt.

+ (kein Parameter)

Der Wert des aktuell mit SLOT adressierten DMX-Kanals des Preload-Puffers wird inkrementiert (1 wird addiert)

Anmerkung: Der Wert kann nicht größer als 255 werden. Wenn er bereits gleich 255 ist, wird der Befehl ignoriert.

Das VALUE-Register wird durch diesen Befehl nicht verändert. Die Manipulation findet direkt im Preload-Puffer statt. Ein eventuell auf diesem DMX-Kanal aktiver Fade-Prozess wird abgebrochen.

- (Minus, kein Parameter)

Der Wert des aktuell mit SLOT adressierten DMX-Kanals Preload-Puffers wird dekrementiert (1 wird abgezogen)

Anmerkung: Der Wert kann nicht kleiner als 0 werden. Wenn er bereits gleich 0 ist, wird der Befehl ignoriert.

Das VALUE-Register wird durch diesen Befehl nicht verändert. Die Manipulation findet direkt im Preload-Puffer statt. Ein eventuell auf diesem DMX-Kanal aktiver Fade-Prozess wird abgebrochen.

^ <Summand>

Addiert <Summand> zum Preload Puffer im aktuell durch SLOT adressierten DMX-Kanal (und startet einen Fade-Prozess)

Anmerkung: Der Endwert kann nicht grösser als dezimal 255 eingestellt werden. Falls die Addition einen Überlauf verursachen würde, wird der Endwert auf 255 begrenzt.

Hat etwa die gleiche Funktion wie der V-Befehl. Statt eines absoluten DMX- Werts wird jedoch die Summe aus (bisherigem Eintrags von VALUE <Summand>) neu in VALUE eingetragen und als Sollwert eines neuen Fade-Prozesses genommen. Jeder auf diesem DMX-Kanal aktive Fade-Prozess wird mit dem neuen Sollwert und FADETIME überschrieben und neu gestartet.

_ <Subtrahend>

Subtrahiert <Subtrahend> vom Preload Puffer im aktuell durch SLOT adressierten DMX-Kanal (und startet einen Fade-Prozess)

Anmerkung: Der Endwert kann nicht kleiner als 0 eingestellt werden. Falls die Subtraktion ein "Borgen" verursachen würde, wird der Endwert auf 0 begrenzt.

Hat etwa die gleiche Funktion wie der V-Befehl. Statt eines absoluten DMX- Werts wird jedoch die Differenz aus (bisherigem Eintrags von VALUE minus <Subtrahend>) neu in VALUE eingetragen und als Sollwert eines neu startenden Fade-Prozesses genommen.

\$ (kein Parameter)

Setzt die Zahlenbasis für DMX-Pegel auf Hexadezimal.

Anmerkung: Alle Parameterwerte der Befehle V und ,(Komma) werden von nun an als Hexadezimalzahlen interpretiert. Dieser Zustand bleibt aktiv, bis durch einen entsprechenden Befehl eine andere Zahlenbasis eingestellt wird. Da die Zahlenbasis auch in Presets 0 bis 3 gespeichert ist, wird evtl. auch durch das Laden dieser Presets oder durch ein Reset der Box die Zahlenbasis verändert.

Alle an den PC gemeldeten DMX-Pegelwerte sind ebenfalls als Hexadezimalzahl mit vorangestelltem "\$" codiert.

& (kein Parameter)

Setzt die Zahlenbasis für DMX-Pegel auf Dezimal.

Anmerkung: Alle Parameterwerte der Befehle V und ,(Komma) werden von nun an als Dezimalzahlen interpretiert. Dieser Zustand bleibt aktiv, bis durch einen entsprechenden Befehl eine andere Zahlenbasis eingestellt wird. Da die Zahlenbasis auch in Presets 0 bis 3 gespeichert ist, wird evtl. auch durch das Laden dieser Presets oder durch ein Reset der Box die Zahlenbasis verändert.

Alle an den PC gemeldeten DMX-Pegelwerte sind ebenfalls als Dezimalzahl (ohne vorangestelltes oder angehängtes Symbol) codiert.

% (kein Parameter)

Setzt die Zahlenbasis für DMX-Pegel auf Prozent.

Anmerkung: Alle Parameterwerte der Befehle V und ,(Komma) werden von nun an als Prozentwerte 0--100 interpretiert. Dieser Zustand bleibt aktiv, bis durch einen entsprechenden Befehl eine andere Zahlenbasis eingestellt wird. Da die Zahlenbasis auch in Presets 0 bis 3 gespeichert ist, wird evtl. auch durch das Laden dieser Presets oder durch ein Reset der Box die Zahlenbasis verändert.

Alle an den PC gemeldeten DMX-Pegelwerte sind ebenfalls in Prozent mit angehängtem %-Symbol codiert.

Die Verwendung der Prozentskala ist **deutlich ungenauer als die Verwendung der Dezimal- oder Hexadezimalskala**. Intern arbeitet der USB / DMX Generator stets mit voller Auflösung 0-255. Daher kann die mehrfache Anwendung der Prozentskala zu erheblichen Abweichungen vom exakt berechneten Prozentwert führen.

((kein Parameter)

Blockiert den Preload-Puffer d.h. er wird nicht in den Sendepuffer kopiert

Anmerkung: Die Blockierung des Preload-Puffers wird eingesetzt, um Daten für eine neue Szene vom PC zu laden, ohne dass Teile der halbfertig aufgebauten Szene bereits gesendet werden.

Diese Methode ist speziell dann einzusetzen, wenn komplexe, langsam übertragene oder per Hand eingegebene Daten übertragen werden oder wenn 16 bit Controller konsistent nachgeladen werden sollen und die neue Konfiguration mit einem "Bang" aktiviert werden soll.

Während der Blockierung werden bereits vorher aktive Fade-Prozesse direkt in den Sendepuffer übertragen, ansonsten wird der vorherige Inhalt zyklisch aus dem Sendepuffer wiederholt. Nachdem alle Daten in den Preload-Puffer übertragen sind, wird die Blockierung mit dem Befehl ")" wieder aufgehoben.

) (kein Parameter)

Beendet die Blockierung des Preload-Puffers. Ab dem nächsten DMX-Reset wird sein Inhalt wieder in den Sendepuffer kopiert.

Unmittelbar danach wird das Zeichen ^ über die USB Schnittstelle ausgegeben. Dies ermöglicht die optimale Synchronisation von zeitkritischen Übertragungen mit Blockierung (z.B. Lauflicht-Effekte).

T <Sekunden.Zehntel>

Der Parameter wird ins FADETIME-Register eingetragen. Keine Aktion wird unmittelbar ausgelöst.

Parameter: Für Sekunden wird ein Wert zwischen 0 und 31 eingetragen. **Optional** - durch einen Punkt getrennt - können Zehntel-Sekunden angehängt werden. Die Blendzeit kann also maximal 31,9 Sekunden betragen.

Beispiel: T13.4 stellt FADETIME auf 13,4 Sekunden ein.

! (kein Parameter)

Alle Blendvorgänge werden sofort abgebrochen und auf den momentanen Wert eingefroren (FREEZE)

Sendepuffer auslesen:

Z<Anzahl Bytes>

Soviele Bytes wie im Parameter angegeben aus dem SENDEpuffer ab DMX-Kanal "SLOT" auslesen und via USB als ASCII Nachricht senden.

Parameter: Anzahl der auszulesenden Bytes (1 bis max. 128)

Aufbau einer Zustandsnachricht als Antwort auf den Z- Befehl:

s <1.Kanal-Nr> v [\$]Zustandswert [, \$Zustandswert] <CR =dez13 =hexD >

Als Dezimalzahl gemeldete Zustandswerte werden ohne Zusatz gesendet

Als Hexadezimalzahl gemeldete Zustandswerte werden mit vorangestelltem '\$' gesendet

In Prozent gemeldete Zustandswerte werden mit nachgestelltem '%' gesendet

Systembefehle:

M <Prozent>

Trägt den Parameter ein in Masterfader. Die Werte aller DMX-Kanäle werden unmittelbar moduliert.

Parameter: Der Masterfader wird grundsätzlich mit einem Prozentwert eingestellt (ohne nachgestelltes % Zeichen und unabhängig von der aktiven Zahlenbasis).

Voreinstellung =100, Maximum = 200, Minimum = 0.

Anmerkung: Der Masterfader arbeitet als digitaler Signal-Prozessor während der Sendepuffer in die DMX-Transmitter Hardware übertragen wird. Er ist nützlich für die globale Helligkeitsjustierung von Lichtszenen. **Interne Daten der DMX-Generators werden hierdurch nicht verändert.**

Der **momentan gesendete Pegel eines jeden DMX- Kanals** ist der aktuelle Wert des Sendepuffers multipliziert mit dem Masterfader-Faktor,d.h. bis zu 200%. Bedingt durch die intern ausgeführte schnelle Integer-Arithmetik kann der tatsächliche DMX-Pegel geringfügig kleiner ausfallen als eine exakte Berechnung ergeben würde (interne Zwischenrechnungen werden stets nach unten gerundet).

Änderungen des Parameters wirken unmittelbar, FADETIME hat hierauf keinen Einfluss.

i <Schleifenlänge>

stellt die Schleifenlänge einer Lauflichtkette ("Chaser") ein

Anmerkung: Die Lauflichtkette beginnt immer mit DMX-Kanal 1 und ist mindestens 2, maximal 255 DMX-Kanäle lang. Mit den Befehlen > und < werden alle aktuellen DMX-Werte zum nächsten bzw. vorherigen Kanal übertragen. Das Ende der Kette verbindet sich mit Kanal 1 zu einem geschlossenen Kreis. Alle DMX-Kanäle ausserhalb der Kette funktionieren weiterhin normal. Die Befehle zum Einstellen eines DMX-Pegels funktionieren auch innerhalb der Lauflichtkette, aktive Fade-Prozesse werden mit dem Chaser weitergereicht.

Für DMX-Scheinwerfer mit mehreren Kanälen (z.B. RGB-LEDs) werden die Schiebebefehle mehrmals unmittelbar hintereinander eingegeben. Um dabei eventuell auftretende Übergangsstörungen zu

vermeiden, kann das Nachladen des DMX-Senders kurzfristig unterbunden werden mit einer Befehlskette der Art (>>>).

Alternativ können Lauflicht- und Flackereffekte durch wiederholtes Laden von Presets erzeugt werden.

< (kein Parameter)

Lauflichtkette einen Schritt nach niedrigeren DMX-Kanälen verschieben

Alternativ wenn i=0 eingegeben: SLOT dekrementieren (um 1 verringern)

> (kein Parameter)

Lauflichtkette einen Schritt nach höheren DMX-Kanälen verschieben

Alternativ wenn i=0 eingegeben: SLOT inkrementieren (um 1 erhöhen)

L <Länge>

Stellt ein, wieviele Kanäle (Slots) in jedem DMX-Zyklus gesendet werden.

Anmerkung: Die Pegel der höheren DMX-Kanäle bleiben gespeichert, werden aber nicht gesendet. Wenn zu einem späteren Zeitpunkt irgendein DMX-Kanal oberhalb der aktuell eingestellten Zyklus-Länge adressiert oder beschrieben wird, passt sich die Zyklus-Länge automatisch an. Mit einem danach gegebenen L-Befehl kann die Zykluslänge jederzeit wieder reduziert werden. Entsprechend den Vorgaben des DMX512 Standards kann die Zyklus-Länge nicht kleiner als 24 Slots eingestellt werden.

Beispiel: L87 begrenzt den DMX-Sendezyklus auf Kanäle 1 bis 87.

Q (kein Parameter)

Gibt die aktuellen Werte aller relevanten Register des mit dem 'S'-Befehl adressierten DMX-Kanals ("SLOT") aus. Die Meldung erscheint als klarschrift-lesbarer ASCII -Text.

Beispiel einer typischen Meldung:

Tx: CH=1 OUT=13 TX=27 MF=50% CY=0 L=512 T=3.2

Anmerkung: OUT ist der aktuelle Pegel für den derzeit adressierten DMX-Kanal CH.TX beschreibt den Inhalt des Sendepuffers dieses DMX-Kanals. Beide unterscheiden sich, wenn der globale Masterfader (Anzeige MF) nicht auf 100% eingestellt ist. CY gibt die aktuelle Länge der Lauflicht-Schleife ("Chaser") an. L zeigt die aktuell eingestellte Länge des DMX-Zyklus und die Zahl nach T ist die aktuell eingestellte Fade-Zeit für Überblendungen (Sekunden.Zehntel).

~ <preset#>

Speichert den aktuellen Inhalt des Sendepuffers permanent als Preset (=Lichtszene) Nummer "preset#".

Parameter: preset# (Bereich 0 bis 114)

Anmerkung: Die "Systemkonfiguration" wird nur in den Presets 0 bis 3 gespeichert. Das betrifft den aktuellen Wert von LOOP, die Zahlenbasis für DMX-Werte und die Lauflicht-Schleife. In den Presets Nr. 0 bis 3 werden daher vorzugsweise unterschiedliche System-Konfigurationen zum schnellen Wechsel abgelegt. **Der Parameterwert von FADETIME wird nur in Preset 0 gespeichert. Da dieses Preset automatisch beim Einschalten geladen wird, kann so ein "soft Start" konfiguriert werden** (neu ab Revisionsnummer 67).

Bei den Presets 4 bis 114 wird nur die aktuelle Lichtszene des Sendepuffers gespeichert, diese lässt sich damit universell aus verschiedenen Systemkonfigurationen aufrufen.

Der Parameterwert des Masterfader wird nicht in Presets gespeichert.

Beispiel: ~23 speichert den Zustand des DMX Generators als Preset Nummer 23

@ <preset#>

Lädt Preset (=Lichtszene) Nummer "preset#" in den Sendepuffer

Parameter: preset# (Bereich 0 bis 114)

Anmerkung: Beim Laden der Presets Nr. 0 bis 3 wird das System entsprechend den gespeicherten Vorgaben neu konfiguriert. Bei den anderen Presets wird nur der Inhalt des Sendepuffers ausgetauscht. Im Lieferzustand sind alle Presets so formatiert, dass der DMX Generator beim Laden eines anwenderseitig unbeschriebenen Preset die Werkseinstellungen übernimmt, Details siehe Befehl "|"

Beim Einschalten oder Reset des DMX Generators wird automatisch stets Preset Nr. 0 geladen.

Wenn FADETIME ungleich 0 eingestellt ist, wird beim Laden des Presets mit dieser Zeitkonstante vom aktuellen Zustand in die neue Lichtszene übergeblendet. **Ausnahme:** (neu ab Revisionsnummer 67) wenn Preset Nr. 0 geladen wird (auch beim Einschalten des Geräts), dann wird mit dem dort gespeicherten Wert von FADETIME geblendet.

Beispiel: @23 lädt Preset N.r 23 und stellt den DMX-Generator entsprechend ein.

| (kein Parameter)

"clear all memory": Alle Puffer und Einstellungen werden auf die Werkseinstellung gesetzt.

Alle DMX-Pegelwerte des Sendepuffers werden auf "0" gesetzt. Zahlenbasis "Dezimal".

Masterfader = 100% Das Lauflicht ("Chaser") wird abgeschaltet. LOOP = 512, FADETIME = 0.0

Presets werden nicht gelöscht oder verändert.

' (Apostroph, kein Parameter)

sendet den Buchstaben 'r' gefolgt von einer 2-stelligen Revisions-Nummer über die USB-Schnittstelle zurück.

Anmerkung: Dieser Befehl dient Anwenderprogrammen zur automatischen Suche der USB-Schnittstelle, an der die der DMX Generator angeschlossen ist. Wird im normalen Betrieb nicht benötigt.

} <Impulsdauer> } <Impulsdauer>

Spezialbefehl zur Modifikation der Dauer des DMX-Reset-Impuls

Anmerkung: Dieser Befehl ist vorgesehen zur Lösung extrem seltener Problemfälle und **sollte nur nach Rücksprache mit unserer Serviceabteilung angewandt werden.**

Nach der rechten geschweiften Klammer wird ein Byte bestehend aus 2 Hex-Ziffern 0...F in Textform eingegeben. **Der Befehl muss unmittelbar mit gleichem Parameter wiederholt werden**, es darf weder eine Leerstelle noch ein Zeilenumbruch dazwischen gefügt werden. Die Parameter werden - unabhängig von Presets - permanent gespeichert und bei jedem Systemstart geladen.

1.Hex-Ziffer (High-Nibble): Dauer des DMX-Reset Impuls: ca. 90us Grundwert + (Nibblewert * 8 us).

Der Einstellbereich beträgt also etwa 90us bis 210 us.

2.Hex-Ziffer (Low-Nibble): Dauer des Mark-Impuls zwischen DMX-Reset und Startbyte:

ca. 10us + Nibblewert. Der Einstellbereich beträgt also etwa 10us bis 25us.

Ausnahme: Der Befehl }00}00 setzt das System dauerhaft zurück auf die Werkseinstellung.

"USB / DMX512 Generator" Betriebsanleitung Teil 2

Binäres Protokoll des USB / DMX512 Generators

Zwar sind ASCII-Befehle einfach und angenehm mit einem Terminalprogramm oder einem komfortabel ausgerüsteten Anwenderprogramm einzugeben. **Häufig jedoch macht es jedoch bei Anwendungsprogrammen mit sehr einfacher Skriptsprache oder bei SPS-Steuerungen Schwierigkeiten, die einzustellenden Zahlenwerte in einen ASCII-Text umzuwandeln.** Ein anderes Problem bei ASCII-Befehlen besteht darin, dass pro Befehl wesentlich mehr Daten übertragen werden müssen als für die eigentliche Nutzinformation nötig. Binäre Übertragung - speziell von Datenblöcken - ist deutlich schneller.

Obwohl weniger anschaulich und auf den ersten Blick schwieriger in der Anwendung, bieten wir daher für die wichtigsten Aufgaben einen Satz binärer Befehle an.

Die binären Befehle sind vollständig transparent koexistent mit dem ASCII Befehlssatz. Alle Befehlstypen können - korrekte Codierung vorausgesetzt - beliebig gemischt werden. Eine explizite Modus-Umschaltung ist nicht erforderlich.

NEU: es werden zwei unterschiedlich konzipierte binäre Befehlsätze angeboten.

Zuerst wird der auch in den vorherigen Firmware-Versionen vorhandene Befehlssatz beschrieben, der **"nicht druckbare" ASCII Codes** (ASCII Code < 32(hex20) **zur Synchronisation** und Unterscheidung zu normalen ASCII Befehlen verwendet. Die bekannten DMX-Lichtsteuerprogramme "Freestyler" und "DMX-Control" steuern Cinetix-Geräte auf der Basis dieses Befehlssatzes an.

Anschliessend wird ein neu eingeführter - zu **MIDI-Kanalnachrichten** kompatibler binärer Befehlssatz beschrieben. Eine Steuerung hierfür ist zwar etwas komplizierter zu programmieren, dafür ist setzt er sich dank seiner Konstruktion aber robuster von ASCII-Befehlen ab und re-synchronisiert sich bei Übertragungsfehlern schneller und zuverlässiger.

Binärer Befehlssatz mit "nicht druckbaren" ASCII Zeichen

Binäre Befehle zum Setzen einzelner DMX-Kanäle lösen Fade-Prozesse aus - ebenso wie ihr ASCII-Äquivalent.

Binäre Befehle zum Übertragen von Datenblöcken schreiben hingegen direkt in den Preload-Puffer und brechen bereits aktivierte Fade-Prozesse sofort ab. Als alternative Blockübertragung mit Fadeprozessen bietet sich der zum MIDI CHANNEL PRESSURE kompatible Binärbefehl an.

Mit "druckbaren" ASCII-Zeichen, so wie sie auf der PC-Tastatur eingetippt werden können, ist es nicht möglich, in den binären Befehls-Modus zu gelangen. **Der binäre Modus wird automatisch verlassen, sobald die dafür notwendige Anzahl Bytes über die USB-Schnittstelle übertragen wurde.** Falls einzelne Bytes verloren gehen sollten, wird jede binäre Übertragung automatisch nach 0,5 Sekunden beendet um das "Aufhängen" des DMX Generators zu vermeiden.

Jeder binäre Befehl beginnt mit einem typischen Befehlscode, der den Typ des Befehls charakterisiert. Hiermit ist indirekt auch die Anzahl der nachfolgenden Bytes festgelegt. Diese werden als rohe 8-Bit Binärdaten interpretiert, auch wenn sie "druckbar" sind.

Wenn ein binärer Befehl **falsch eingegeben** wurde oder sonstwie nicht ausgeführt werden konnte, sendet er genau wie ein ASCII-Befehl ein **Fragezeichen** zurück.

Im Folgenden bedeuten in spitze Klammern gesetzte Zahlen rohe Binärwerte, keine ASCII-Codes. (Die Klammern sind nicht Bestandteil der Befehlscodes.)

Codierung der binären Befehle:

Der Befehlscode in spitzen Klammern charakterisiert die auszuführende Operation

<2> DMX-Pegel im Kanal-Bereich 1 bis 255 einstellen, (ferner Sonderfall Kanal 512).

gefolgt von 2 Bytes

Datenbyte 1: Vorgabe Kanal 1 - 255.

Wenn der Parameter = 0 ist, **dann wird DMX-Kanal 512 eingestellt.**

Datenbyte 2: DMX-Pegel, der an diesem DMX-Kanal eingestellt werden soll.

Nach diesen 3 Bytes wird das Steuer-Interface automatisch wieder auf ASCII Befehle geschaltet.

<3> DMX-Wert im Kanal-Bereich 256 bis 511 einstellen.

gefolgt von 2 Bytes

Datenbyte 1: (Kanal 256 bis 511) **minus 256**, also Byte1 = 0 bis 255

Datenbyte 2: DMX-Pegel, der an diesem DMX-Kanal eingestellt werden soll.

Nach diesen 3 Bytes wird das Steuer-Interface automatisch wieder auf ASCII Befehle geschaltet.

<4> stellt DMX-Werte in einem Block aufeinander folgender Kanäle ein.

Der Start-Kanal des Blocks kann im Bereich 1 bis 255 liegen.

Hierauf folgen **zwei Headerbytes** und eine **festgelegte Anzahl Datenbytes**

1. Headerbyte: Start-Kanal 1 bis 255

2. Headerbyte: Anzahl der nachfolgend in den Sendepuffer einzutragenden DMX-Bytes. Um 256 Bytes auszulesen, trägt man hier 0 ein.

Datenbytes: gezählte Folge beliebiger Bytes.

Es müssen exakt so viele DMX-Bytes geschickt werden, wie im 2. Befehlsbyte eingetragen sind. Werden zu viele Bytes übertragen, können diese als ASCII Zeichen interpretiert werden und ungewollte Aktionen auslösen. **Werden zu wenige Bytes übertragen,** bleibt das Gerät in einer Endlosschleife hängen, die nach ca 0,5 Sekunden automatisch abgebrochen wird. Alle bis dahin übertragenen Datenbytes werden in den Preload Puffer eingetragen

Nach Übertragung aller im Befehl vorgesehenen Bytes wird das Steuer-Interface automatisch wieder auf ASCII Befehle geschaltet.

<5> stellt DMX-Werte in einem Block aufeinander folgender Kanäle ein.

Der Start-Kanal des Blocks kann im Bereich 256 bis 511 liegen.

Hierauf folgen **zwei Headerbytes** und eine **festgelegte Anzahl Datenbytes**

1. Headerbyte: (Startkanal 256 bis 511) **minus 256** (=Bytewert 0 bis 255).

2. Headerbyte: Anzahl der nachfolgend in den Sendepuffer einzutragenden DMX-Bytes. Wird hier 0 eingetragen, folgen 256 Bytes.

Startkanal 512 ist mit dem Befehl nicht erreichbar, da ein Block-Befehl dann keinen Sinn macht.

Datenbytes: gezählte Folge beliebiger Bytes

Wurde durch falsche Berechnung des 2.Headerbytes Kanal 512 überschritten, werden die zuviel empfangenen Bytes ignoriert, jedoch wird der binäre Datenempfang aufrecht erhalten, bis die vorgegebene Anzahl Bytes empfangen wurde. Siehe ansonsten Kommentar zu Befehl <4>.

<7> stellt den Masterfader und die FADETIME ein.

gefolgt von 2 Datenbytes

Datenbyte 1: Vorgabe Masterfader 0 bis 200 (hexC8).

Wenn der Parameter >200 ist, dann wird der Masterfader nicht verändert.

Datenbyte 2: FADETIME in 1/10 Sekunden Einheiten, zulässige Werte 0 bis 254.

Abweichend vom entsprechenden ASCII-Befehl können also binär nur Fade-Zeiten von 0 bis 25,4 Sekunden eingestellt werden

Wenn der Parameter = 255 (hexFF) ist, dann wird FADETIME nicht verändert.

Nach diesen 3 Bytes wird das Steuer-Interface automatisch wieder auf ASCII Befehle geschaltet.

<12> lädt ein Preset (=Lichtszene)

mit Überblendung entsprechend dem aktuellen Wert von FADETIME

gefolgt von 2 Datenbytes:

Datenbyte 1: zwecks Kompatibilität mit anderen Cinetix Produkten **immer = 0**

Datenbyte 2: zu ladende Preset-Nr 0 bis 239 (hexEF) - **siehe Anmerkung oben.**

Nach diesen 3 Bytes wird das Steuer-Interface automatisch wieder auf ASCII Befehle geschaltet.

Binärer Befehlssatz mit MIDI-kompatiblen Steuerbefehlen

Eine MIDI-Kanalnachricht beginnt mit einem Statusbyte, darauf folgen je nach Nachrichtentyp 1 oder 2 Datenbytes.

Das erste Byte jeder Nachricht ist das **Statusbyte**, **nur bei diesem ist das höchstwertige Bit 7 gesetzt**. Damit ist es eindeutig von allen ASCII-Befehlen unterscheidbar.

Bei allen danach folgenden Datenbytes ist das höchstwertige Bit7 gelöscht, der sonstige Inhalt ist kontext-abhängig.

Jeder MIDI-Befehl wird erst akzeptiert und ausgewertet, wenn die Anzahl der zum Statusbyte gehörenden Datenbytes empfangen wurde. Unvollständige Befehle und überschüssige Datenbytes werden ignoriert. Bei fehlenden Datenbytes wird ab dem nächsten Statusbyte mit der Decodierung eines neuen Befehls begonnen.

Damit wird gegenüber dem zuvor beschriebenen binären Befehlssatz zusätzliche Datensicherheit gewonnen.

Nachteilig an den MIDI-kompatiblen Befehlen ist die auf 7 Bit beschränkte Nutzlänge eines Datenbytes. Zur Übertragung von 8-Bit Daten ist somit immer eine Verteilung von Nutz-Bits auf mehrere MIDI-Bytes notwendig.

Beispiel zur Umrechnung des gewünschten DMX-Pegels in einen Binär-Befehl siehe unten.

DMX-Kanal (=SLOT) und Preload-Puffer einstellen

Die Division **./.** durch 2 wird nach den mathematischen Regeln der "Ganzzahl-Division" ausgeführt, d.h. es gibt keine Nachkommastellen, der Teilungsrest wird ignoriert, es wird immer auf die nächst kleinere Ganzzahl "abgerundet".

DMX-Kanal zwischen 1 und 127

DMX-Pegel gerade Zahl

Statusbyte= 144 (hex 90)

1.Datenbyte = DMX-Kanal

2.Datenbyte = DMX-Pegel **./.** 2

DMX-Pegel ungerade Zahl

Statusbyte= 176(hex B0)

1.Datenbyte = DMX-Kanal

2.Datenbyte = DMX-Pegel **./.** 2

DMX-Kanal zwischen 128 und 255

DMX-Pegel gerade Zahl

Statusbyte = 145 (hex 91)

1.Datenbyte = DMX-Kanal - **128**

2.Datenbyte = DMX-Pegel **./.** 2

DMX-Pegel ungerade Zahl

Statusbyte = 177 (hex B1)

1.Datenbyte = DMX-Kanal - **128**

2.Datenbyte = DMX-Pegel **./.** 2

DMX-Kanal zwischen 256 und 383

DMX-Pegel gerade Zahl

Statusbyte = 146 (hex 92)

1.Datenbyte = DMX-Kanal - **256**

2.Datenbyte = DMX-Pegel **./.** 2

DMX-Pegel ungerade Zahl

Statusbyte = 178 (hex B2)

1.Datenbyte = DMX-Kanal - **256**

2.Datenbyte = DMX-Pegel **./.** 2

DMX-Kanal zwischen 384 und 511

DMX-Pegel gerade Zahl

Statusbyte = 147 (hex 93)
1.Datenbyte = DMX-Kanal - **384**
2.Datenbyte = DMX-Pegel **./. 2**

DMX-Pegel ungerade Zahl

Statusbyte = 179 (hex B3)
1.Datenbyte = DMX-Kanal - **384**
2.Datenbyte = DMX-Pegel **./. 2**

DMX-Kanal gleich 512

DMX-Pegel gerade Zahl

Statusbyte = 144 (hex 90)
1.Datenbyte = 0
2.Datenbyte = DMX-Pegel **./. 2**

DMX-Pegel ungerade Zahl

Statusbyte = 176 (hex B0)
1.Datenbyte = 0
2.Datenbyte = DMX-Pegel **./. 2**

Beispiel: nehmen wir an, dass DMX-Kanal 450 auf den Pegel 101 eingestellt werden soll. Dazu wird in obiger Tabelle nachgesehen im Block "DMX-Kanal zwischen 384 und 511" und da der einzustellende DMX-Pegel eine ungerade Zahl ist, in der rechten Spalte.

Daraus ergibt sich: Statusbyte = 179,

1.Datenbyte= (450-384) = 66,

2. Datenbyte = (101 ganzzahlig dividiert durch 2) = 50

Adressierten DMX-Kanal ("SLOT") inkrementieren und Pegel einstellen

(geeignet zur **blockweisen Einstellung** von DMX-Kanälen, entspricht dem ASCII Komma-Befehl)

DMX-Pegel gerade Zahl

Statusbyte = 208 (hex D0)
Datenbyte = DMX-Pegel **./. 2**

DMX-Pegel ungerade Zahl

Statusbyte = 209 (hex D1)
Datenbyte = DMX-Pegel **./. 2**

Ergänzend gibt es die Möglichkeit, nacheinander folgende Befehle dieser Art ohne Wiederholung des Statusbytes zu übertragen ("Running State").

Während dies im MIDI-Umfeld üblich ist, **muss der "Running State" hier unbedingt mit einem letzten Statusbyte beendet werden, um Fehlinterpretation nachfolgender ASCII-Befehle zu vermeiden.** Dies ergibt eine effiziente Methode zum Einstellen mehrerer aufeinanderfolgender DMX-Kanäle **mit Überblendung.**

Die folgenden ansonsten äquivalenten Befehle aktivieren den Running State:

DMX-Pegel gerade Zahl

Statusbyte = 210 (hex D2)
Datenbyte = DMX-Pegel **./. 2**

DMX-Pegel ungerade Zahl

Statusbyte = 211 (hex D3)
Datenbyte = DMX-Pegel **./. 2**

Alle anderen MIDI-kompatiblen Statusbytes beenden automatisch den Running State, insbesondere alle einzelnen Bytes im Bereich 246 (hex F6) bis 255 (hex FF) .

Beispiel:

Um die DMX-Kanäle 1 bis 5 mit den DMX-Pegeln 10,20,30,40,50, die Kanäle 6,7,8 mit den Pegeln 55,65,75 und Kanal 9 mit Pegel 0 zu beschreiben, wäre folgende dezimale Bytefolge zu senden:
144,1,5,210,10,15,20,25,211,27,32,37,210,0,255.

Mit dem letzten Byte 255 wird hier die Akzeptanz für Running State wieder abgeschaltet, somit werden nachfolgende ASCII-Befehle korrekt interpretiert.

Gleichwertig ist aber auch die Byte-Folge: 144,1,5,210,10,15,20,25,211,27,32,37,208,0.

FADETIME einstellen

Statusbyte = 160 (hex A0)

1. Datenbyte = Blend-Dauer in Sekunden (0 bis 31)

2. Datenbyte = zusätzliche Blenddauer in Zehntelsekunden (0 bis 9).

Anmerkung: Der momentane Wert von FADETIME wird beim Start des Blendvorgangs in die jeweilige Fader-Ressource kopiert. Unmittelbar danach kann FADETIME ohne Rückwirkung auf laufende Blendprozesse geändert werden.
Normalerweise wird der Ausgang der Fader-Ressourcen in den Preload-Puffer geschrieben. Wenn der Preload-Puffer jedoch "blockiert" ist - siehe vorstehend beschriebene POLY KEY PRESSURE Befehle - werden die Ausgangsdaten zusätzlich asynchron direkt in den Sendepuffer geschrieben. Blockieren des Preload-Puffers verursacht also keine Störung laufender Blendprozesse.

Presets (= Lichtszenen) laden mit :

- Statusbyte = 160 (hex A0)
 - 1. Datenbyte = 96 (hex 60):
 - 2. Datenbyte = 0 bis 99 (hex 63): Nr. des zu ladenden Presets
-

Presets (= Lichtszenen) abschaltfest speichern mit :

- Statusbyte = 160 (hex A0)
 - 1. Datenbyte = 112 (hex 70):
 - 2. Datenbyte = 0 bis 99 (hex 63): Nr. des zu speichernden Presets
-

Masterfader einstellen mit :

- Statusbyte = 160 (hex A0)
 - 1. Datenbyte = 120 (hex 78)
 - dann 2. Datenbyte = 0-127 (hex 7F) Masterfader -Einstellung in %
 - 1. Datenbyte = 121 (hex 79)
 - dann 2. Datenbyte = 0-100 (hex 64) Masterfader -Einstellung 100-200%
(100 höher als im Datenbyte eingetragen)
- siehe **Anmerkung** zum äquivalenten ASCII-Befehl "M"
-

Lauflichtfunktion ("Chaser") einstellen mit :

- Statusbyte = 160 (hex A0)
- 1. Datenbyte = 122 (hex 7A)
 - dann 2. Datenbyte = 0 Lauflicht ("Chaser") ausschalten
 - oder** = 2-127 (hex 7F) dann Lauflicht Schleife 2-127 einstellen
(Datenbyte=1 macht keinen Sinn und wird nicht akzeptiert)
- 1. Datenbyte = 123 (hex 7B)
 - dann 2. Datenbyte = 0-127 (hex 7F) Lauflicht Schleife 100 bis 227 einstellen
(100 höher als im Datenbyte eingetragen)
- 1. Datenbyte = 124 (hex 7C)
 - dann 2. Datenbyte = 0-55 (hex 37) Lauflicht Schleife 200 bis 255 einstellen
(200 höher als im Datenbyte eingetragen)

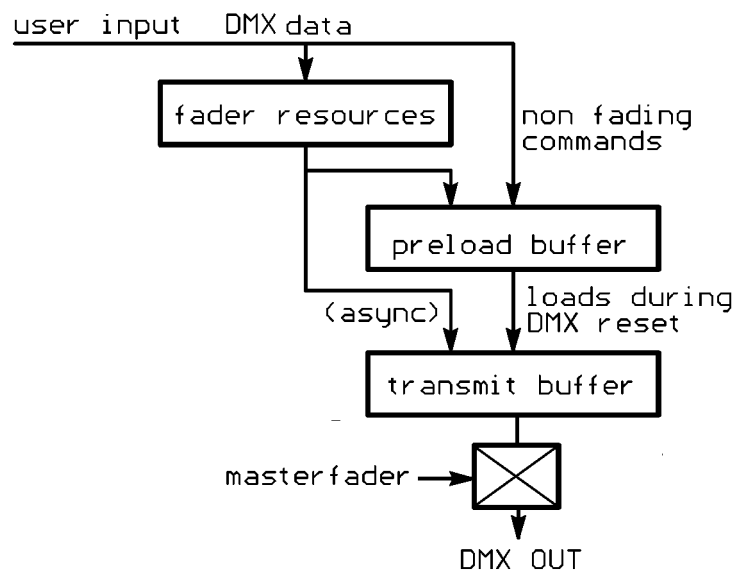
Anhang A

Konzept und Datenformat als DMX512 Sender

Um die Funktion des USB / DMX Generators und der darauf zugreifenden Befehle besser zu verstehen, sollten Sie eine grobe Kenntnis der internen Datenaufbereitung und Zwischenspeicherung für das gesendete DMX-Signal haben.

Für jeden der 512 DMX-Kanäle (DMX-Slang: "Slots") sind zwei spezifische Speicherzellen vorhanden: Preload-Puffer und Sendepuffer.

Der Preload-Puffer ist eine Art Montagespeicher. Im Normalfall wird der Preload-Puffer bei jedem DMX-Zyklus in den Sendepuffer kopiert, d.h. muss vom Anwender nicht weiter beachtet werden. Der Kopiervorgang kann aber mit den Befehlen " (" und ") " blockiert werden, so dass komplexe Szenen unsichtbar aufgebaut werden können, die dann mit einem Schlag zur Sendung an DMX OUT freigegeben werden.



Die beim Beschreiben des Preload-Puffers angewandte Befehls-Philosophie beruht auf dem Zusammenwirken dreier Register ("Register" = Computer-Fachausdruck für einen Speicherplatz mit spezieller Funktion):

- dem **SLOT-Register** (beschreiben mit Befehl "S"),
- dem **VALUE (Pegel)-Register** (beschreiben mit Befehlen V, Komma, + , - , ^ , _) und
- dem **FADETIME -Register** (beschreiben mit Befehl "T").

Das **SLOT-Register** speichert den **DMX-Kanal**, den die **USB- Kommandoschnittstelle** als nächstes beschreiben oder auslesen soll.

Das **VALUE-Register** speichert den hier einzutragenden **DMX-Pegel** als Byte-Wert. Zugleich wird der Wert an der Position SLOT in den Preload-Puffer eingetragen.

Für jeden DMX-Kanal kann mit einem ASCII-Befehl ein kompletter, vom USB / DMX Generator intern automatisch gesteuerter **Blendvorgang** ausgelöst werden. Hierzu wird die Blendzeit im **FADETIME-Register** voreingestellt. **Fade-Prozesse können auf allen DMX-Kanälen gleichzeitig aktiv sein** und es werden auch Presets (= abgespeicherte Lichtszenen) beim Laden mit FADETIME in die zuvor bestehende Lichtszene übergeblendet. Der momentane Wert von FADETIME wird beim Start des Blendvorgangs in die jeweilige Fader-Ressource kopiert. Unmittelbar danach kann FADETIME ohne Rückwirkung auf laufende Blendprozesse geändert werden.

Normalerweise wird der Ausgang der Fader-Ressourcen in den Preload-Puffer geschrieben. Wenn der Preload-Puffer jedoch "blockiert" ist - siehe Befehle "(" und ")" - werden die Ausgangsdaten zusätzlich asynchron direkt

in den Sendepuffer geschrieben. Blockieren des Preload-Puffers verursacht also keine Störung laufender Blendprozesse.

Der **Masterfader** moduliert die gesendeten DMX-Daten wie ein einfacher Signalprozessor wenn sie vom Sendepuffer in die DMX-Transmitter Hardware übertragen werden. Auf diese Weise kann eine komplette Lichtszene auf einfache Art "gestretcht" werden. Intern gespeicherte DMX-Daten werden durch den Masterfader nicht manipuliert.

Ferner gibt es ein weiteres Register von untergeordneter Bedeutung: **LOOP** speichert die aktuelle Länge des DMX-Zyklus (Anzahl gesendeter DMX-Kanäle).

Nach Einschalten oder Reset des Generators ist der aktuell adressierte DMX-Kanal ("SLOT") initialisiert mit "1", VALUE und FADETIME sind auf "0" gesetzt, der Masterfader wird mit 100% initialisiert und LOOP wird mit "512" oder dem im Preset gespeicherten Wert gestartet.

Anhang B

Das Datenformat DMX512

Der Standard "DMX512" bzw. dessen Neufassung ANSI E1.11-2004 ("DMX512-A") sieht vor, dass ein Bus-Master digitale Steuerdaten auf eine Busleitung sendet. Diese Busleitung wird von Empfänger zu Empfänger durchgeschleift. **Alle am Bus angeschlossenen DMX-Empfänger werden zyklisch mit allen aktuellen Steuerdaten versorgt**, unabhängig davon ob sich diese Daten in der Zwischenzeit verändert haben und ob sie für diesen Empfänger bestimmt sind.

Jeder **DMX-Zyklus beginnt mit einem speziellen Reset-Impuls und dem Start-Byte** als Header. Anschliessend werden nacheinander alle Datenbytes im 8-Bit Format an alle potentiellen Empfänger gesendet. Somit ist jedem DMX-Kanal eine eindeutig bestimmte Zeitscheibe (time slot) in jedem DMX-Zyklus zugeordnet. Daher werden die DMX-Kanäle im DMX512-A Standard als "Slots" bezeichnet. Pro DMX-Slot wird also ein Datenbyte übertragen. Der Wert dieses Datenbytes (im Bereich 0 bis 255) beschreibt dann z.B. die Helligkeit einer Lampe oder die Stellung eines "Moving Head". **In diesem Manual verwenden wir für den Wert eines DMX-Datenbytes die Bezeichnung "DMX-Pegel"**.

Die **Adressierung der Empfänger** erfolgt durch die Position ("Slot") ihrer Datenbytes im DMX-Zyklus relativ zum Startbyte. Mittels eines Codierschalters o.ä. kann am DMX-Empfänger eine Zahl zwischen 1 und 512 eingestellt werden. Der Empfänger tastet in jedem DMX-Zyklus die ab diesem Slot gesendeten Datenbytes aus dem DMX-Datenstrom ab. Die Anzahl der ausgewerteten Bytes hängt ab von der spezifischen Funktion des Empfängers. Wird z.B. am Codierschalter des Empfängers die 28 eingestellt, so kopiert dieser Empfänger das 28., 29., 30. ... Byte nach dem Startbyte aus dem DMX-Datenstrom. Die Daten werden nicht aus dem DMX-Datenstrom entfernt, alle anders eingestellten Empfänger ignorieren sie jedoch. Je nach Einstellung des Basis-Slots können also mehrere Empfänger - gewollt oder versehentlich - die gleichen Daten empfangen.

Der "DMX512" Standard erlaubt aufgrund seiner Konstruktion keine Fehlerprüfung und -Korrektur der Datenübertragung und bietet keinen eindeutig genormten Rückkanal vom Empfänger zum Sender. Bei der Ansteuerung träger Aktuatoren wie Glühlampendimmer oder Servomotoren wirken sich einzelne fehlerhaft empfangene Bytes kaum aus, weil sie ja im nächsten Datenpaket mit hoher Wahrscheinlichkeit wieder korrigiert werden. **Sehr kritisch ist hingegen das Auslösen einmaliger Schaltvorgänge über den DMX-Bus.**

<p>Aus diesen Gründen ist der Einsatz von DMX ausdrücklich VERBOTEN bei allen sicherheitskritischen Anwendungen, bei denen es im Fehlerfall zu Personen- oder größeren Sachschäden kommen könnte !</p>

Als Installations-Kabel für den DMX512 Bus verwendet man am besten ein spezielles 2-polig verdrehtes und abgeschirmtes "RS-485" Datenkabel, das allerdings recht teuer und nur im Spezialhandel für industrielle Elektronik erhältlich ist. CAT5 Ethernetkabel hat sich ebenfalls gut bewährt. Für Festinstallationen bis ca. 300m eignet sich abgeschirmte ISDN-Leitung JY(St)Y, wobei eine Ausführung mit 0,8mm Aderdurchmesser vorzuziehen ist. Bei Buslängen bis insgesamt 100 m genügt auch ein gutes 2-adriges Mikrofonkabel. Die Abschirmung des Kabels sollte an beiden XLR-Steckverbindern - nur - mit Pin1 verbunden werden.

Lange Busleitungen müssen am empfängerseitigen Ende "abgeschlossen" bzw "terminiert" werden:
DMX+ und DMX- werden mit einem 120 Ohm - Widerstand verbunden.

In der Regel wird die Busleitung von einem Empfänger zum nächsten "durchgeschleift". Bei den meisten Dimmern etc ist DMX OUT intern direkt mit DMX IN verbunden. Bei solchen Installationen hat man am besten einen 5-poligen XLR-Stecker mit eingebautem Widerstand parat, den man in DMX OUT des **letzten** am Bus angeschlossenen Gerätes steckt.



Cinetix Medien und Interface GmbH
Gemündenerstr. 27 D-60599 Frankfurt/Main
Fon: +49-69-68 51 05 Fax +49-69-68 600 409
<http://www.cinetix.de/interface/>

* Irrtum, technische Änderungen und Lieferbarkeit vorbehalten.

* Diese Beschreibung ist informativ und sichert keine Produkteigenschaften im rechtlichen Sinne zu.

* Im Text zitierte Warenzeichen und Produktnamen sind geschütztes Eigentum ihrer Eigentümer