

DMX4ALL Commands

- ENGLISH -

Overview

<u>INTRODUCTION.....</u>	<u>3</u>
COM PORT CONNECTION	4
USB DRIVER CONNECTION.....	5
TCP CONNECTION.....	6
ASCII COMMANDS.....	7
Check connection.....	7
Get Informations	7
Set Channel	8
Set Channel ALL	8
Set Channel with fade	9
Set Channel ALL with fade.....	9
Stop fade	10
Stop fade ALL.....	10
Get Channel	11
Get Channel from DMX-INPUT.....	11
Run Program (1).....	12
Run Program (2).....	12
Get Program.....	12
Set MasterDimmer	13
Get MasterDimmer	13
SET RED BRIGHTNESS.....	14
GET RED BRIGHTNESS	14
SET GREEN BRIGHTNESS	15
GET GREEN BRIGHTNESS.....	15
SET BLUE BRIGHTNESS	16
GET BLUE BRIGHTNESS.....	16
Set BlackOut	17
Get BlackOut	17
SET OUTPUT ON/OFF (BLACKOUT)	18
GET OUTPUT ON/OFF (BLACKOUT).....	18
Set Speed.....	19
Get Speed	19
Set Freeze.....	20
Get Freeze	20
Set number of output channels.....	21
Get number of output channels	21
GET SETTINGS	22
SAMPLES FOR ASCII COMMANDS	23
ARRAY COMMANDS	24
Write DMX-OUT (ArrayTransfer)	24
Write DMX-OUT Channel 1-256 (Fast Mode).....	25
Write DMX-OUT Channel 257-512 (Fast Mode).....	25
Read back DMX-OUT	26
Read DMX-IN.....	26
MULTIUNIVERSE ARRAY TRANSFER	27
Write DMX-OUT (MultiUniverse).....	27
Read DMX-OUT.....	28
Read DMX-IN.....	28
STAND-ALONE MEMORY.....	29
Write Stand-Alone-Memory	29

Introduction

To communicate with the DMX4ALL Interfaces it is necessary to open a connection. This can be made with different ports:

- COM port
- Virtual COM port (VCP)
- USB connection with FTDI D2XX driver
- TCP connection

After connecting to the DMX Interface you can use the DMX4ALL commands.

COM port connection

The COM port connection is possible with the following DMX4ALL interfaces:

- USB-DMX-Interfaces with VCP (FTDI USB-VCP driver)
- RS232 connected interfaces (DMX Player XS)
- LAN-DMX-Interface (using Lantronix Com Port Redirector)
- ArtNet-DMX STAGE-PROFI 1.1 (using LAN-VCP)
- ArtNet-DMX UNIVERSE DR1.1 (using LAN-VCP)
- ArtNet-DMX UNIVERSE 4.1 (using LAN-VCP)
- ArtNet-DMX UNIVERSE DR4.1 (using LAN-VCP)
- ArtNet-LED Dimmer 4/4R (using LAN-VCP)
- ArtNet-LED Dimmer 6/6R (using LAN-VCP)
- ArtNet-DMX TWIN PoE (using LAN-VCP)

Open a COM port with the following parameters:

Baudrate: 38400 Baud (Some old interface can be switched to 19200 Baud)
Bytes: 8
Stop-Bits: 1
Parity: None
Handshake: None

After you open the connection and have the COM handle, you can use the standard transfer functions to write and read the data from the interface.

USB driver connection

The direct USB driver connection is possible with the following DMX4ALL interfaces:

USB-DMX-Interfaces with FTDI D2XX driver or FTDI combined driver

- Mini-USB-DMX-Interface
- DMX Player S
- DMX Player L
- DMX Player ONE
- USB-DMX STAGE-PROFI MK2

To communicate with the USB-Driver, you have to use the FTDI D2XX driver. More informations you can find in the "D2XX Programmer's Guide" from FTDI.

To send data to the interface, you have to open the device with FT_Open. After you got a handle to the device, you can send the data via FT_Write and read back data with FT_Read.

Example:

```
FT_HANDLE    ftHandle;
FT_STATUS    ftStatus;
DWORD        RxBytes, TxBytes;
DWORD        BytesReceived, BytesWritten;
char         RxBuffer[256], TxBuffer[256];

ftStatus = FT_Open(0,&ftHandle);
if(ftStatus != FT_OK)
    return;    // FT_Open failed

FT_SetBaudRate(ftHandle, FT_BAUD_38400);

TxBuffer = "C000L255";
TxBytes = 8;
ftStatus = FT_Write(ftHandle, TxBuffer, TxBytes, &BytesWritten);
if(ftStatus != FT_OK)
    return;    // FT_Write failed

RxBytes = 10;
ftStatus = FT_Read(ftHandle, RxBuffer, RxBytes, &BytesReceived);
if (ftStatus == FT_OK)
{
    // FT_Read OK
    // Check received chars
}else{
    // FT_Read Failed
}
FT_Close(ftHandle);
```

TCP connection

The TCP connection is possible with the following DMX4ALL interfaces:

- LAN-DMX-Interface
- ArtNet-DMX STAGE-PROFI 1.1
- ArtNet-DMX UNIVERSE DR1.1
- ArtNet-DMX UNIVERSE 4.1
- ArtNet-DMX UNIVERSE DR4.1
- ArtNet-LED Dimmer 4/4R
- ArtNet-LED Dimmer 6/6R
- ArtNet-DMX TWIN PoE

For control the interface over the TCP connection please use the IP-address of the device and use the port 10001 to open the TCP connection.

The sent and received data are RAW-data packets with the DMX4ALL command data inside.

ASCII commands

Check connection

Description

Check if the connection is OK.

Definition

Transmit C?
Receive G

Devices

- ALL -

Get Informations

Description

Get informations about the interface

Definition

Transmit I
Receive *char-array with informations about the interface*

```
ArtNet-Player L - V1.0  
(c) 2012 DMX4ALL GmbH  
>>> www.DMX4ALL.eu <<<  
MAC: 00-60-35-64-00-00
```

```
Programs:  
BLUE  
GREEN  
RED  
DROPS  
ROTATE  
MAGIAR  
CIRCLE  
STRIPES  
FIRE  
RADIAL  
TETRIS
```

Devices

- ALL -

Set Channel

Description

Write a Value (Level) to one Channel

Definition

Transmit

CxxxLyyy

xxx = Channel (000 ... 511)

yyy = Level (000 ... 255)

Receive

G

Devices

- ALL -

Definition

Transmit

CxxxxxLyyy

xxxxx = Channel (000 ... 65535)

Value depends on device output universes

Channel 0 - 511 → Universe 1

Channel 512-1023 → Universe 2

: : : : : : : : : : :

yyy = Level (000 ... 255)

Receive

G

Devices

- ArtNet-DMX TWIN PoE

V1.02 or higher

Set Channel ALL

Description

Write a value (Level) to all Channels

Definition

Transmit

CALLLyyy

yyy = Level (000 ... 255)

Receive

G

Devices

- ArtNet-LED-Dimmer 4

V2.12 or higher

- ArtNet-LED-Dimmer 6

V2.12 or higher

- ArtNet-DMX STAGE-PROFI 1.1

V2.15 or higher

- ArtNet-DMX TWIN PoE

V1.02 or higher

Set Channel with fade

Description

Schreibt einen Wert (Level) auf einen Kanal (Channel) der über einen Fade mit der vorgegebenen Zeit erreicht wird.

Definition

Transmit

CxxxTzzzzzLyyy

xxx = Channel (000 ... 511)

yyy = Level (000 ... 255)

zzzzz = Time, Unit 0,1s (00000 ... 65535)

Receive

G

Devices

- ArtNet-LED-Dimmer 4 V2.12 or higher
- ArtNet-LED-Dimmer 6 V2.12 or higher
- ArtNet-DMX STAGE-PROFI 1.1 V2.15 or higher

Set Channel ALL with fade

Description

Schreibt einen Wert (Level) auf alle Kanäle (Channel) der über einen Fade mit der vorgegebenen Zeit erreicht wird.

Definition

Transmit

CALLTzzzzzLyyy

yyy = Level (000 ... 255)

zzzzz = Time, Unit 0,1s (00000 ... 65535)

Receive

G

Devices

- ArtNet-LED-Dimmer 4 V2.12 or higher
- ArtNet-LED-Dimmer 6 V2.12 or higher
- ArtNet-DMX STAGE-PROFI 1.1 V2.15 or higher

Stop fade

Description

Stoppt einen Fade für den angegebenen Kanal (Channel) der mit *Set CHANNEL with fade* oder *Set CHANNEL ALL with fade* gestartet wurde.

Definition

Transmit **CxxxT00000**
xxx = Channel (000 ... 511)

Receive *G*

Devices

- ArtNet-LED-Dimmer 4 V2.12 or higher
- ArtNet-LED-Dimmer 6 V2.12 or higher
- ArtNet-DMX STAGE-PROFI 1.1 V2.15 or higher

Stop fade ALL

Description

Stoppt alle Fades die mit *Set CHANNEL with fade* oder *Set CHANNEL ALL with fade* gestartet wurde.

Definition

Transmit **CALLT00000**

Receive *G*

Devices

- ArtNet-LED-Dimmer 4 V2.12 or higher
- ArtNet-LED-Dimmer 6 V2.12 or higher
- ArtNet-DMX STAGE-PROFI 1.1 V2.15 or higher

Get Channel

Description

Read back a DMX output channel value.

Definition

Transmit **Cxxx?**
xxx = Channel (000 ... 511)

Receive **yyyG**
yyy = Level (000 ... 255)

Devices

- ALL -

Get Channel from DMX-INPUT

Description

Read a DMX input channel value.

Definition

Transmit **Vxxx?**
xxx = Channel (000 ... 511)

Receive **yyyG**
yyy = Level (000 ... 255)

Devices

- USB-DMX STAGE PROFI MK3
- ArtNet-DMX STAGE-PROFI 1.1
- ArtNet-DMX UNIVERSE DR1.1

Run Program (1)

Description

Start a program.

Definition

Transmit

Raaa

aaa = Program (001...max. program)

Receive

G

Devices

- DMX-Player S / L / ONE

Run Program (2)

Description

Start a program.

Definition

Transmit

Paaa

aaa = Program (001...max. program)

Receive

G

Devices

- ArtNet-LED-Dimmer 4
- ArtNet-LED-Dimmer 6
- ArtNet Player 4
- ArtNet MultiPixxPlayer 16
- DMX-Configurator V2.1 or higher

Get Program

Description

Request the actual executed program.

Definition

Transmit

P?

Receive

yyyG

yyy = Actual program

Devices

- ArtNet-LED-Dimmer 4
- ArtNet-LED-Dimmer 6
- ArtNet Player 4
- ArtNet MultiPixxPlayer 16
- DMX-Configurator V2.1 or higher

Set MasterDimmer

Description

Write the MasterDimmer value.

Definition

Transmit **Laaa**
aaa = MasterDimmer value (000 ... 255)

Receive **G**

Devices

- ArtNet-LED-Dimmer 4
- ArtNet-LED-Dimmer 6
- ArtNet Player 4
- ArtNet MultiPixxPlayer 16
- DMX-Configurator V2.1 or higher

Get MasterDimmer

Description

Read the MasterDimmer value

Definition

Transmit **L?**

Receive **bbbG**
bbb = actual read MasterDimmer value

Devices

- ArtNet-LED-Dimmer 4
- ArtNet-LED-Dimmer 6
- ArtNet Player 4
- ArtNet MultiPixxPlayer 16
- DMX-Configurator V2.1 or higher

Set RED Brightness

Description

Write the brightness value for RED channels

Definition

Transmit

Raaa

aaa = RED brightness value (000 ... 255)

Receive

G

Devices

- ArtNet-LED-Dimmer 4
- ArtNet-LED-Dimmer 6
- ArtNet Player 4
- ArtNet MultiPixxPlayer 16
- DMX-Configurator V2.1 or higher

Get RED Brightness

Beschreibung

Read the RED brightness value.

Definition

Transmit

R?

Receive

bbbG

bbb = actual read RED brightness value

Devices

- ArtNet-LED-Dimmer 4
- ArtNet-LED-Dimmer 6
- ArtNet Player 4
- ArtNet MultiPixxPlayer 16
- DMX-Configurator V2.1 or higher

Set GREEN Brightness

Description

Write the brightness value for GREEN channels

Definition

Transmit

Gaaa

aaa = GREEN brightness value (000 ... 255)

Receive

G

Devices

- ArtNet-LED-Dimmer 4
- ArtNet-LED-Dimmer 6
- ArtNet Player 4
- ArtNet MultiPixxPlayer 16
- DMX-Configurator V2.1 or higher

Get GREEN Brightness

Beschreibung

Read the GREEN brightness value.

Definition

Transmit

G?

Receive

bbbG

bbb = actual read GREEN brightness value

Devices

- ArtNet-LED-Dimmer 4
- ArtNet-LED-Dimmer 6
- ArtNet Player 4
- ArtNet MultiPixxPlayer 16
- DMX-Configurator V2.1 or higher

Set BLUE Brightness

Description

Write the brightness value for BLUE channels

Definition

Transmit

Raaa

aaa = BLUE brightness value (000 ... 255)

Receive

G

Devices

- ArtNet-LED-Dimmer 4
- ArtNet-LED-Dimmer 6
- ArtNet Player 4
- ArtNet MultiPixxPlayer 16
- DMX-Configurator V2.1 or higher

Get BLUE Brightness

Beschreibung

Read the BLUE brightness value.

Definition

Transmit

R?

Receive

bbbG

bbb = actual read BLUE brightness value

Devices

- ArtNet-LED-Dimmer 4
- ArtNet-LED-Dimmer 6
- ArtNet Player 4
- ArtNet MultiPixxPlayer 16
- DMX-Configurator V2.1 or higher

Set BlackOut

Description

Switch Blackout ON/OFF.

Definition

Transmit

Ba

a = 0 (BLACKOUT off) / a = 1 (BLACKOUT on)

Receive

G

Devices

- DMX-Player S / L / ONE

Get BlackOut

Description

Get the actual BlackOut state.

Definition

Transmit

B?

Receive

bG

b = 0 (BLACKOUT is off)

b = 1 (BLACKOUT is on)

Devices

- DMX-Player S / L / ONE

Set Output On/Off (BlackOut)

Description

Switch output ON/OFF.

Definition

Transmit **Oa**
a = 0 (OUTPUT off) / a = 1 (OUTPUT on)

Receive **G**

Devices

- ArtNet-LED-Dimmer 4
- ArtNet-LED-Dimmer 6
- ArtNet Player 4
- ArtNet MultiPixxPlayer 16
- DMX-Configurator ab V2.1

Get Output On/Off (BlackOut)

Description

Get the actual Output state.

Definition

Transmit **O?**

Receive **bG**
b = 0 (OUTPUT off)
b = 1 (OUTPUT on)

Devices

- ArtNet-LED-Dimmer 4
- ArtNet-LED-Dimmer 6
- ArtNet Player 4
- ArtNet MultiPixxPlayer 16
- DMX-Configurator ab V2.1

Set Speed

Description

Write the speed value.

Definition

Transmit

Saaa

aaa = Speed value (slow 000 ... 255 fast)

Receive

G

Devices

- ArtNet-LED-Dimmer 4
- ArtNet-LED-Dimmer 6
- ArtNet Player 4
- ArtNet MultiPixxPlayer 16

Get Speed

Description

Read the actual speed value.

Definition

Transmit

S?

Receive

bbbG

bbb = actual read speed value

Devices

- ArtNet-LED-Dimmer 4
- ArtNet-LED-Dimmer 6
- ArtNet Player 4
- ArtNet MultiPixxPlayer 16

Set Freeze

Description

Stop the program execution.

Definition

Transmit

Fa

a = 0 (Freeze off) / a = 1 (Freeze on)

Receive

G

Devices

- ArtNet-LED-Dimmer 4
- ArtNet-LED-Dimmer 6
- ArtNet Player 4
- ArtNet MultiPixxPlayer 16

Get Freeze

Description

Read the actual Freeze-State.

Definition

Transmit

F?

Receive

bG

b = 0 (Freeze off)

b = 1 (Freeze on)

Devices

- ArtNet-LED-Dimmer 4
- ArtNet-LED-Dimmer 6
- ArtNet Player 4
- ArtNet MultiPixxPlayer 16

Set number of output channels

Description

Set the number of channels used for the DMX output.

Definition

Transmit **Nxxx**
xxx= = Number of channels (000 ... 511)

Receive **G**

Devices

- DMX-Player S / L / ONE
- Mini USB DMX Interface
- USB-DMX STAGE-PROFI MK2

Get number of output channels

Description

Get the actual number of output channels

Definition

Transmit **N?**

Receive **yyyG**
yyy = Number of channels (000 ... 511)

Devices

- DMX-Player S / L / ONE
- Mini USB DMX Interface
- USB-DMX STAGE-PROFI MK2

Get Settings

Description

Read the settings from the device into one string

Definition

Transmit	X?
Receive	<i>Rxxx;Gxxx;Bxxx;Lxxx;Sxxx;Pxxx;Fb;Ob;G</i>

xxx = Current value
b = Current state

Devices

- ArtNet-LED-Dimmer 4
- ArtNet-LED-Dimmer 6
- ArtNet Player 4
- ArtNet MultiPixxPlayer 16
- ArtNet Analog/Relais Interface
- DMX-Configurator V2.1 or higher

Samples for ASCII commands

Check connection to interface:

Transmit character: C?

Receive character: G

Set DMX channel 1 to value 255:

Transmit character: C000L255

Receive character: G

Set DMX channel 123 to value 47:

Transmit character: C122L047

Receive character: G

Read DMX-OUT buffer channel 1:

Transmit character: C000?

Receive character: 255G (actual value)

Read DMX-IN channel 1:

Transmit character: V000?

Receive character: 255G (actual value)

Read number of channels on DMX-IN

Transmit character: V?

Receive character: 512G (actual value)

Array commands

The array transfer is used to transfer more than one data byte (array) in one command. Each command begins with a header byte that defines the command. After this command the control bytes are transferred. After the last control byte the interface send the answer packet (Receive).



Please note, that the array of the transmitted data are maximum 255 bytes long. Also the sum of the start channel and the changes channels must be a maximum of 255 (0xFF) !

Write DMX-OUT (ArrayTransfer)

Transmit			Receive
0xFF		(Block transfer header)	
<i>start channel</i>	<i>L</i>	(0x00 ... 0xFF)	
<i>start channel</i>	<i>H</i>	(0x00 / 0x01)	
<i>number of channels</i>		(0x00 ... 0xFF)	
<i>first data byte</i>		(0x00 ... 0xFF)	
..			
..			
<i>last data byte</i>			G

Description

The start channel defines the destination of the data bytes. The number of channels gives the count of data bytes.



Don't write data to different universes in one command.

Devices

- Mini USB DMX Interface
- USB-DMX STAGE PROFI MK2
- NanoUSB DMX Interface
- USB-DMX STAGE PROFI MK3
- LAN-DMX STAGE-PROFI
- DMX-Player XS / S / L / ONE
- ArtNet-DMX STAGE-PROFI 1.1
- ArtNet-DMX UNIVERSE DR1.1

Example: To set the channels 10-15 to the value 100,120,140,150,255,10, please transmit the following BYTES:

FF 09 00 06 64 78 8C 96 FF 0A (Transmitted bytes)

The interface send back the ASCII char „G“ if the command is OK and executed

Write DMX-OUT Channel 1-256 (Fast Mode)

Transmit		Receive
0xE2	(Block transfer header)	
<i>channel low byte</i>	(0x00 ... 0xFF)	
<i>data byte</i>	(0x00 ... 0xFF)	

Description

Write one DMX value to output buffer.

Devices

- MiniUSB DMX Interface
- NanoUSB DMX Interface
- LAN-DMX STAGE PROF1
- USB-DMX STAGE PROF1 MK2
- USB-DMX STAGE PROF1 MK3
- ArtNet-DMX STAGE-PROFI 1.1
- ArtNet-DMX UNIVERSE DR1.1

Write DMX-OUT Channel 257-512 (Fast Mode)

Transmit		Receive
0xE3	(Block transfer header)	
<i>channel low byte</i>	(0x00 ... 0xFF)	
<i>data byte</i>	(0x00 ... 0xFF)	

Description

Write one DMX value to output buffer.

Devices

- MiniUSB DMX Interface
- NanoUSB DMX Interface
- LAN-DMX STAGE PROF1
- USB-DMX STAGE PROF1 MK2
- USB-DMX STAGE PROF1 MK3
- ArtNet-DMX STAGE-PROFI 1.1
- ArtNet-DMX UNIVERSE DR1.1

Read back DMX-OUT

Transmit			Receive
0xFE		(Block transfer header)	
<i>start channel</i>	<i>L</i>	(0x00 ... 0xFF)	
<i>start channel</i>	<i>H</i>	(0x00 / 0x01)	
<i>number of channels</i>		(0x00 ... 0xFF)	
			<i>DATA[0]</i>
			..
			..
			<i>DATA[n]</i>
			<i>G</i>

Description

Read back the DMX values from output buffer.

Devices

- MiniUSB DMX Interface
- NanoUSB DMX Interface
- LAN-DMX STAGE PROFI
- USB-DMX STAGE PROFI MK2
- USB-DMX STAGE PROFI MK3
- ArtNet-DMX STAGE-PROFI 1.1
- ArtNet-DMX UNIVERSE DR1.1

Read DMX-IN

Transmit			Receive
0xFC		(Block transfer header)	
<i>start channel</i>	<i>L</i>	(0x00 ... 0xFF)	
<i>start channel</i>	<i>H</i>	(0x00 / 0x01)	
<i>number of channels</i>		(0x00 ... 0xFF)	
			<i>DATA[0]</i>
			..
			..
			<i>DATA[n]</i>
			<i>G</i>

Description

Read DMX values from the input signal.

Devices

- USB-DMX STAGE PROFI MK3
- ArtNet-DMX STAGE-PROFI 1.1
- ArtNet-DMX UNIVERSE DR1.1

MultiUniverse array transfer

The **MultiUniverse array transfer** commands can handle more than 512 channels (1 universe).

Write DMX-OUT (MultiUniverse)

Transmit			Receive
0xD1		(Block transfer header)	
<i>start channel</i>	<i>H</i>	(0x00 ... 0xFF)	
<i>start channel</i>		(0x00 ... 0xFF)	
<i>start channel</i>		(0x00 ... 0xFF)	
<i>start channel</i>	<i>L</i>	(0x00 ... 0xFF)	
<i>number of channels</i>	<i>H</i>	(0x00 ... 0xFF)	
<i>number of channels</i>	<i>L</i>	(0x00 ... 0xFF)	
<i>first data byte</i>		(0x00 ... 0xFF)	
..			
..			
<i>last data byte</i>			G

Description

The start channel defines the destination of the data bytes, also the universe. To access universe 1 the destination channel must be 0 to 511, to access universe 2 the destination channel must be 512 to 1023, to access universe 3 the destination channel must be 1024 to 1535 and so on.



Don't write data to different universes in one command.

Devices

- AvengerSD Player
- MultiPixx DMX-Controller
- ArtNet-DMX STAGE-PROFI 1.1
- ArtNet-DMX UNIVERSE DR1.1

Read DMX-OUT

Transmit			Receive
0xD2		(Block transfer header)	
<i>start channel</i>	<i>H</i>	(0x00 ... 0xFF)	
<i>start channel</i>		(0x00 ... 0xFF)	
<i>start channel</i>		(0x00 ... 0xFF)	
<i>start channel</i>	<i>L</i>	(0x00 ... 0xFF)	
<i>number of channels</i>	<i>H</i>	(0x00 ... 0xFF)	
<i>number of channels</i>	<i>L</i>	(0x00 ... 0xFF)	
			<i>DATA[0]</i>
			<i>..</i>
			<i>..</i>
			<i>DATA[n]</i>

Description

The start channel defines the source of the data bytes, also the universe.
 To access universe 1 the source channel must be 0 to 511,
 to access universe 2 the source channel must be 512 to 1023,
 to access universe 3 the source channel must be 1024 to 1535 and so on.



Don't read data from different universes in one command.

Devices

- AvengerSD Player
- MultiPixx DMX-Controller
- ArtNet-DMX STAGE-PROFI 1.1
- ArtNet-DMX UNIVERSE DR1.1

Read DMX-IN

Transmit			Receive
0xD8		(Block transfer header)	
<i>universe</i>		(0x00 ... 0xFF)	
<i>start channel</i>	<i>H</i>	(0x00 ... 0xFF)	
<i>start channel</i>	<i>L</i>	(0x00 ... 0xFF)	
<i>number of channels</i>	<i>H</i>	(0x00 ... 0xFF)	
<i>number of channels</i>	<i>L</i>	(0x00 ... 0xFF)	
			<i>CHANNELS H</i>
			<i>CHANNELS L</i>
			<i>DATA[0]</i>
			<i>..</i>
			<i>..</i>
			<i>DATA[n]</i>

Devices

- AvengerSD Player
- MultiPixx DMX-Controller
- ArtNet-DMX STAGE-PROFI 1.1
- ArtNet-DMX UNIVERSE DR1.1

Stand-Alone memory

The device internal Stand-Alone memory is written by the command **Write Stand-Alone-Memory**. Devices which use a SD-Card for Stand-Alone it is necessary to use a SD-Card-Writer to store the data to the card.

Write Stand-Alone-Memory

Transmit		Receive
0xFD	(Block transfer header)	
address	L (0x00 ... 0xFF)	
address	H (0x00 ... 0xFF)	
data byte	(0x00 ... 0xFF)	
checksum	(address low xor address high xor data byte)	
		G

Description

To write data to the stand alone memory you need the array transfer.

Devices

- DMX-Player XS / S / M / L / ONE
- LAN-DMX STAGE-PROFI
- USB-DMX STAGE-PROFI MK2

Stand-Alone memory organisation (DataVersion 2)

```

    BYTE    0xF0 (StartByte)
    BYTE    0x02 (DataVersion)
    BYTE    20 (ProgramTriggerFrequenz) (not used)
    WORD    UsedChannels (not used)
    BYTE    DefaultProgramm 0=no selection / 1=Program[0]
    BYTE    NrOfProgramms

Program[0]
    WORD    NrOfAllProgramBytes
    BYTE    Speed
    WORD    PointerToFirstStep

InitStep
    BYTE    NrOfData (=n) (not used yet, must be 0)
    WORD    Channel[1]
    BYTE    Data[1]
    :
    WORD    Channel[n]
    BYTE    Data[n]

ProgramName
    BYTE[15]

Step (FirstStep)
    WORD    PointerToNextStep
    WORD    FadeTime
    WORD    DelayTime
    BYTE    NrOfData (=n)
    WORD    Channel[1]
    BYTE    Data[1]
    :
    WORD    Channel[n]
    BYTE    Data[n]

Step
    WORD    PointerToNextStep
    WORD    FadeTime
    WORD    DelayTime
    BYTE    NrOfData (=n)
    WORD    Channel[1]
    BYTE    Data[1]
    :
    WORD    Channel[n]
    BYTE    Data[n]

:::
Step
    WORD    PointerToFirstStep
    WORD    FadeTime
    WORD    DelayTime
    BYTE    NrOfData (=n)
    WORD    Channel[1]
    BYTE    Data[1]

```

```

:
WORD Channel[n]
BYTE Data[n]

Program[1]
:::
Program[n]

BYTE Anzahl der Event-Datensätze (0-16)
Event[0]
BYTE Event-Data0
BYTE Event-Data1
BYTE Event-Data2
BYTE Event-Data3
BYTE Event-Data4
BYTE Event-Data5
BYTE Event-Data6
:::
Event[n]

```

Timer Event

```

Triggerdata0 = Minute
Triggerdata1 = Stunde
Triggerdata2 = Days
Triggerdata3 = TIMER_EVENT ( =0x01 )
Triggerdata4 = Action ( 0x00 - Start Scene )
Triggerdata5 = free
Triggerdata6 = Scene to start (1...x)

```

MIDI Event

```

Triggerdata0 = MIDI_Command
Triggerdata1 = MIDI_Data1
Triggerdata2 = MIDI_Data2
Triggerdata3 = MIDI_EVENT ( =0x02 )
Triggerdata4 = Action
Triggerdata5 = ActionParam1
Triggerdata6 = ActionParam2

```

Stand-Alone memory organisation (DataVersion 3)

Only to use with DMX-Player ONE !

```

        BYTE    0xF0 (StartByte)
        BYTE    0x03 (DataVersion)
        BYTE    Config
        BYTE    DefaultProgramm          0=no selection / 1=Program[0] / ...
        BYTE    NrOfProgramms
Program[0]
        WORD    NrOfAllProgramBytes
        BYTE    Speed
        3BYTES  PointerToFirstStep
ProgramName
        BYTE[15]          ASCII chars
Step (FirstStep)
        3BYTES  PointerToNextStep
        WORD    FadeTime
        WORD    DelayTime
        WORD    NrOfData Ch0-255 (=nLow)    Only channels from 0-255 / 0=no Data
        BYTE    Channel[0] / High = 0x00
        BYTE    Data[0]
        :
        :
        BYTE    Channel[nLow-1]
        BYTE    Data[nLow-1]
        WORD    NrOfData Ch256-511 (=nHigh)  Only channels from 256-511 / 0=no Data
        BYTE    Channel[0] / High = 0x01
        BYTE    Data[0]
        :
        :
        BYTE    Channel[nHigh-1]
        BYTE    Data[nHigh-1]
Step
        3BYTES  PointerToNextStep
        WORD    FadeTime
        WORD    DelayTime
        WORD    NrOfData Ch0-255 (=nLow)    Only channels from 0-255 / 0=no Data
        BYTE    Channel[0] / High = 0x00
        BYTE    Data[0]
        :
        :
        BYTE    Channel[nLow-1]
        BYTE    Data[nLow-1]
        WORD    NrOfData Ch256-511 (=nHigh)  Only channels from 256-511 / 0=no Data
        BYTE    Channel[0] / High = 0x01
        BYTE    Data[0]
        :
        :
        BYTE    Channel[nHigh-1]
        BYTE    Data[nHigh-1]
:::
Step
        3BYTES  PointerToNextStep
        WORD    FadeTime
        WORD    DelayTime
        WORD    NrOfData Ch0-255 (=nLow)    Only channels from 0-255 / 0=no Data
        BYTE    Channel[0] / High = 0x00
        BYTE    Data[0]
        :
        :
        BYTE    Channel[nLow-1]
        BYTE    Data[nLow-1]
        WORD    NrOfData Ch256-511 (=nHigh)  Only channels from 256-511 / 0=no Data
        BYTE    Channel[0] / High = 0x01
        BYTE    Data[0]
        :
        :
        BYTE    Channel[nHigh-1]
        BYTE    Data[nHigh-1]
Program[1]
:::
Program[n]
        BYTE    Anzahl der Event-Datensätze    (0-16)
        ARRAY  Event-Datensätze

        BYTE    Anzahl der RGB devices        (0-170)
        WORD    RGB-Device Channel R [0]
        :
        :
        WORD    RGB-Device Channel R [n]

```

DMX 4 ALL

WWW.DMX4ALL.EU

DMX4ALL GmbH
Reiterweg 2A
D-44869 Bochum
Germany

Last change: 27.02.2018

© Copyright DMX4ALL GmbH

All rights reserved. No part of this manual may be reproduced in any form (photocopy, pressure, microfilm or in another procedure) without written permission or processed, multiplied or spread using electronic systems.

All information contained in this manual was arranged with largest care and after best knowledge. Nevertheless errors are to be excluded not completely. For this reason I see myself compelled to point out that I can take over neither a warranty nor the legal responsibility or any adhesion for consequences, which decrease/go back to incorrect data. This document does not contain assured characteristics. The guidance and the characteristics can be changed at any time and without previous announcement.