**Slave**

DMX

Instructions

Michel Keijzers, © 2017

# History

Table : History

|  |  |
| --- | --- |
| **Date/period** | **Actions** |
| Nov 8, 2017 | Initial Version |

# Table of Contents

Contents

[History 1](#_Toc498978467)

[Table of Contents 2](#_Toc498978468)

[List of Tables 3](#_Toc498978469)

[List of Figures 4](#_Toc498978470)

[1 Introduction 5](#_Toc498978471)

[2 Requirements 6](#_Toc498978472)

[2.1 Design 8](#_Toc498978473)

# List of Tables

[Table 1: History 1](#_Toc498978474)

[Table 15: Requirements GS 5](#_Toc498978475)

# List of Figures

# Introduction

This document describes the DMX instructions.

**DMX Channels**

It should be possible to use 2,048 DMX channels (4 universes). However, commands do not have to be optimized for this.

**Pattern**

For e.g. fading effects, it needs to be prevented that continuous changes/messages need to be sent from the Controller to the DMX device. Therefore, patterns will be used. These patterns should be programmable (e.g. speed of the fading effect).

A pattern can be defined per DMX channel.

**DMX Program**

A program is a set of patterns, thus actually the pattern for multiple DMX channels. There can be multiple programs active at the same moment, with or without overlapping DMX channels.

**Scene**

A scene is a set of programs which

The instructions chapter has not been changed according to the information above.

# Instructions

**Code examples**

send DMX Channel 128 Value 10

Send DMX Channel 128 Var V1

Send DMX Channel 128 Prop NoteNumber // Eg from MIDI

send DMX Channel 128 Values 10 20 30 40 50 60 70 80

send DMX Scenes 0 1 2 3 5 7 Mult 16 Value 10

send DMX OffsetChannel 128 Scenes 0 1 2 3 5 7 Mult 16 Value 10

send DMX OffsetChannel 128 Scenes 0 1 2 3 5 7 Mult 16 Values 10 20 30 40 50 60 70 80

**Code instructions**

0: Single channel

“send” “DMX” <channels\_str> <channel> <values\_str> <value>

1: Multiple channels

“send” “DMX” <channels\_str> <channel> <values\_str> (<value>)+

2: Multiple scenes, single value

“send” “DMX” <offset\_channel\_str> <offset\_channel> <scenes\_str> <scene>+ [<mult\_str> (8 | 16)] <values\_str> <value>

3: Multiple scenes, multiple values

“send” “DMX” <offset\_channel\_str> <offset\_channel> <scenes\_str> <scene>+ [<mult\_str> (8 | 16)] <values\_str> <value>+

<channels\_str> = (“channels” | “ch”), case independent

<channel> = <integer 0..255>

<value\_str> = (“values” | “val” | “v”), case independent

<value> = <integer 0..255>

<scenes\_str> = (“scenes” | “sc”), case independent

<scene> = <integer 0..6>

<mult\_str> = (“mult” | “m”), case independent

<mult> = <”8” | “16”>

<offset\_channel> = (“offset\_channel” | “offset\_ch” | “offch” | “och” | “oc”), case independent

<offset\_channel = <integer 0..2040 or 4080>, multiple of <mult>

**Instruction opcodes**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Byte** | **Bits** | **Meaning** | **Values** | **Meaning** | **Description/remarks** |
| 0 | 7~4 | Instruction Type | ???? | Send DMX |  |
| 0 | 3~2 | Sub command | 00 | Single Channel |  |
|  |  |  | 01 | Multiple Channels |  |
|  |  |  | 10 | Multiple scenes, single value |  |
|  |  |  | 11 | Multiple scenes, multiple values |  |
| 0 | 1~0 |  |  |  | Depending on Sub command |

**00: Single Channel**

0: Single channel

“send” “DMX” <channels\_str> <channel> <values\_str> <value>

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Byte** | **Bits** | **Meaning** | **Values** | **Meaning** | **Description/remarks** |
| 0 | 1~0 | Value Type | 00 | Value |  |
|  |  |  | 01 | Variable |  |
|  |  |  | 01 | Property |  |
| 1 |  | Channel | 0-255 | Channel |  |
| 2 |  | Value / Property / Variable | 0-255 | Depending on b0.1~0 | See Values chapter |

**01: Multiple channels**

1: “send” “DMX” <channels\_str> <channel> <values\_str> (<value>)+

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Byte** | **Bits** | **Meaning** | **Values** | **Meaning** | **Description/remarks** |
| 0 | 0~1 | MSB Channel | 00-11 | MSB Channel) | 0, 256, 512 or 1024 |
|  |  |  | 1 | MSB Channel (255-511) |  |
| 1 |  | Channel | 0-255 | Channel |  |
| 2 |  | NrOfValues | 0-255 | Number of values |  |
| 3..NrOfValues |  | Values | 0-255 |  | See Values chapter |

**10: Multiple channels, single value**

2: “send” “DMX” <offset\_channel\_str> <offset\_channel> <scenes\_str> <scene>+ [<mult\_str> (8 | 16)] <values\_str> <value>

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Byte** | **Bits** | **Meaning** | **Values** | **Meaning** | **Description/remarks** |
| 0 | 1~0 | Value Type | 00 | Value |  |
|  |  |  | 01 | Variable |  |
|  |  |  | 01 | Property |  |
| 1 | 7 | Multiplication factor | 0 | 8 |  |
|  |  |  | 1 | 16 |  |
| 1 | 6 | Offset channel or scenes | 0 | Byte 1 are scenes |  |
|  |  |  | 1 | Byte 1 is offset channel |  |
| 1 | 5~0 | If bit 1.6=0: Scenes |  | 6 scenes | (first) 6 scenes |
|  |  | If bit 1.6=1: Offset Channel | 0-255 | Offset channel multiplied by multiplication factor | If mult factor = 8: 0-512  If mult factor = 16: 0-1024 |
| 2 |  | Scenes |  | (LSB) scenes | (last) 8 scenes |
| 3 |  | Value | 0-255 | Value | See Values chapter |

**11: Multiple channels, multiple values**

3: “send” “DMX” <offset\_channel\_str> <offset\_channel> <scenes\_str> <scene>+ [<mult\_str> (8 | 16)] <values\_str> <value> +

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Byte** | **Bits** | **Meaning** | **Values** | **Meaning** | **Description/remarks** |
| 0 | 1 | Multiplication factor | 0 | 8 |  |
|  |  |  | 1 | 16 |  |
| 0 | 0 | Offset channel or scenes | 0 | Byte 1 are scenes |  |
|  |  |  | 1 | Byte 1 is offset channel |  |
| 1 |  | If 0.0=0: Scenes |  | 8 scenes | (first) 8 scenes |
|  |  | If 0.0=1: Offset Channel | 0-255 | Offset channel multiplied by multiplication factor | If mult factor = 8: 0-2048  If mult factor = 16: 0-4096 |
| 2 |  | Scenes |  | (LSB) scenes | (last) 8 scenes |
| 3 |  | Number of Values | 0-255 | Number of values |  |
| 4..x |  | Values | 0-255 | Values | See Values chapter |

# Requirements

## Generic

TODO

## Hardware

TODO

## Software

TODO

# Inputs/Outputs

TODO

# Design

## Diagnostics LEDs

Table 25: Audio Diagnostics LEDs

|  |  |  |
| --- | --- | --- |
| **Function** | **LED Color** | **Description** |
| Power | Blue (generic) | Off: Power off  On: Power on |
| RF | Yellow (generic) | Off: empty message transmitting/receiving  Slow blinking: contact with controller  Double fast blinking per second: no contact with slave  Triple fast blinking per second: problem with RF  On: non empty message transmitting/receiving |

Note that if the GUI Device shows errors whenever possible.

## Breadboard Layout

TODO

## Proto Layout

TODO

## Component List

TODO

# Software

## Design

TODO

## Memory Usage

TODO

## Timing Performance

TODO

# Testing

## Unit Tests

TODO

## Integration Tests

TODO