DXM-0808/DXM-1616 Matrix Controller

Communication protocol V1.0I, last modified 2009-07-16

One. The basic parameters and communication data format

Basic parameters: BAUD = 38400,8 bit data bits, 1 stop bit, no parity

Data Format: The frame format. Each time you send all the data needs to be finished his frame.

Note: The output mode changes, for example:

4 * 4, input channel 4 over the order, the above command output channel 4 invalid.

8 * 4, 8 or more input channels, the order, the above command output channel 4 invalid.

1: Transmit Frame: PC -----> DMX16

Structure: The frame start flag + frame + frame end mark the data segment

Start of frame flag: 0x41 0x52 0x63

Frame end marker: 0XA8 0XB9 0XC3

Frame data segment:

P_ID (ID address code)

P_Com_Hig (command high byte)

P_Com_Low (command low byte)

P_CH (channel logo)

P_N (Data number)

P_Date 1

... ...

P_Date N

CRC (parity bit: P_ID + P_Com_Hig + P_Com_Lowl + P_CH + P_N + P_Data1 .. n)

Address code + command low byte high byte + command + data + channel number + data symbol 1 .. N + check data.

Example:

Address code is 0X01, command high byte 0Xe6, command low byte 0X64, channel logo is 0x00, the total number of data is 0X01, the data 1 is 0X64, the entire frame is as follows:

41 52 63 01 E6 64 00 01 64 B0 A8 B9 C3

2. Receive frame: DMX16 ---> PC

Start of frame flag: 0xf5 0xfa 0xf7

M_ID (consistent with the P_ID)

M_Com_Hig (consistent with the P_Com_Hig)

M_Com_Low (consistent with the P_Com_Low)

M_Count

M_0;

M_Mode

M_Data1

...

M_Date n

Note: n = M_Count

Detailed command:

DMX16 mainly in the following order: online / offline, enter the volume adjustment, matrix volume control, PEQ, XOVER, delay, phase, Limit / Compressor / Noise-Gate, output volume control. Program calls and so on.

One. Online / Offline:

Note: All of the parameter adjustment and program management functions will need the machine online.

P_Com_Hig = 230; P_Com_Low = 100; P_CH = 0; P_N = 1; P_Date1 = 100;

P_Com_Low = 100: Online

P_Com_Low = 80: Offline

Example: all examples are the corresponding ID number is 1. Example, all data is 16 hexadecimal

Online: 41 52 63 01 E6 64 00 01 64 B0 A8 B9 C3

Offline: 41 52 63 01 E6 50 00 01 00 38 A8 B9 C3

Returns: M_Mode = 210; M_Count = 16;

II. Input Volume Control:

Sent:

P_Com_Hig = 21; P_Com_Low = 1 .. 16 (1 represents the input 1 channel, 16, said input of 16 channels);

P_CH = P_Com_Low

P_N = 2; P_Data1 = gain high byte; P_Data2 = gain low byte;

Range: Mute,-40dB ~ 6dB

Mute_flag, Gain_Value;

Mute: Mute_flag = 0x8000 not mute: Mute_flag = 0x0000;

Gain_Value = 200 + (db value * 10 +400) + Mute_flag;

Example:

-40dB when: Gain_Value = 200;

0dB time: Gain_Value = 600; 0dB mute: Gain_Value = 600 +32768 = 33368;

+6 DB when: Gain_Value = 660;

After successfully sending the machine to receive orders:

M_Cout = 1;

M_0 = 0;

M_Mode = 0;

M_Data1 = 0;

Example:

Input 2 channel volume value is-6.9dB

41 52 63 01 15 02 02 02 02 13 31 A8 B9 C3

Enter the 13-channel volume value is +2.1 dB

41 52 63 01 15 0D 0D 02 02 6D A1 A8 B9 C3

Enter the 16-channel volume value is +0.0 dB, Mute

41 52 63 01 15 10 10 02 82 58 12 A8 B9 C3

Add. Output MASTER volume (all of the output channels are controlled):

Sent:

P_Com_Hig = 53; P_Com_Mid = 0;

P_CH =;

P_N = 2; P_Data1 = gain high byte; P_Data2 = gain low byte;

Gain formula consistent with the input volume. Range: Mute,-40dB ~ 6dB

After successfully sending the machine to receive orders:

M_Cout = 1;

M_0 = 0;

M_Mode = 0;

M_Data1 = 0;

III. Output Volume Control:

Sent:

P_Com_Hig = 22; P_Com_Mid = 1 .. 16 (1, said output 1 channel, 16, said output of 16 channels);

P_CH = P_Com_Mid;

P_N = 2; P_Data1 = gain high byte; P_Data2 = gain low byte;

Gain formula consistent with the input volume. Range: Mute,-40dB ~ 6dB

After successfully sending the machine to receive orders:

M_Cout = 1;

M_0 = 0;

M_Mode = 0;

M_Data1 = 0;

Example:

Output 2-channel volume value is +0.0 dB

41 52 63 01 16 02 02 02 02 58 77 A8 B9 C3

IV. Matrix volume control.

Sent:

P_Com_Hig = 32; P_Com_Low = 1 .. 16 (output channel 1 channel 1 represents the input, the input 16-channel 16);

P_CH = 1 .. 16 (input channel 1, said output 1 channel, 16, said output 16 channels)

P_N = 2; P_Data1 = gain high byte; P_Data2 = gain low byte;

Gain formula consistent with the input volume, range: Mute,-40dB ~ 6dB

After successfully sending the machine to receive orders:

M_Cout = 1;

M_0 = 0;

M_Mode = 0;

M_Data1 = 0;

Example:

Input 2-channel 4-channel output signal matrix, the volume value is-2.6dB

41 52 63 01 20 02 04 02 02 3E 69 A8 B9 C3

5-channel input 7-channel output signal matrix, the volume value is-25.8Db, MUTE

41 52 63 01 20 05 07 02 81 56 06 A8 B9 C3

Five. PEQ

Sent:

P_Com_Hig = 27;

P_Com_Low = 1 .. 16 (output channel 1, said output 1 channel, 16, said output 16 channels)

P_CH = P_Com_Low

P_N = 6

PEQ_FREQ = frequency index table [P_Data1 * 256 + P_Data2];

PEQ_GAIN: P_Data3 = gain high byte; P_Data4 = low byte of the gain, the gain formula consistent with the input volume. Range:-12dB ~ +12 dB. Only when the PEQ_GAIN to 0dB PEQ mode can be changed.

PEQ_Q = Q value of the index table [P_Data5];

P_Data6 (bit7, bit6, bit5, bit4, bit3, bit2, bit1, bit0)

Bit7: BYPASS ON/OFF.1 said that BYPASS ON, 0 said BYPASS OFF

bit6, bit4, bit3: PEQ current location, 001 said PEQ1, 010 that PEQ2, 011 that PEQ3. . . 101 said PEQ5.

bit2, bit1, bit0: PEQ mode, 000, said PEQ, 001 that HiShelf, 010 that LoShelf, 011 that AllPass.

Example:

Output 5-channel PEQ2 FREQ = 4.00KHz, GAIN = +10.2 dB, Q = 62.0

41 52 63 01 1B 05 05 06 01 36 02 BE 47 A0 0A A8 B9 C3

After successfully sending the machine to receive orders:

M_Cout = 1;

M_0 = 0;

M_Mode = 0;

M_Data1 = 0;

VI. Phase

P_Com_Hig = 23; P_Com_Low = 1 .. 16 (1, said output 1 channel, 16, said output of 16 channels);

P_CH = P_Com_Low;

P_N = 1; P_Data1 = phase;

P_Data1 = 0 that the phase is 0, P_Data1 = 0x80 that phase is 180;

Example:

180 5-channel output

41 52 6301 17 05 05 01 80 A3 A8 B9 C3

After successfully sending the machine to receive orders:

M_Cout = 1;

M_0 = 0;

M_Mode = 0;

M_Data1 = 0;

VII. Delay

P_Com_Hig = 24; P_Com_Low = 1 .. 16 (1, said output 1 channel, 16, said output of 16 channels);

P_CH = P_Com_Low;

P_N = 3; P_Data1 = 0 (delay time used for future expansion);

P_Data2 high delay

P_Data3 low delay

Delay_Value = P_Data2 * 256 + P_Data3

Delay_Value:0~65520,0:0ms, 1:0.02ms (1/48ms),..., 65520:1365.00ms

Example: when Delay=1000.00ms, Delay_Value=48000

example：

Output 5 channel delay is set to 1000ms

41 52 63 01 18 05 05 03 00 BB 80 61 A8 B9 C3

After successfully sending the machine to receive orders:

M_Cout = 1;

M_0 = 0;

M_Mode = 0;

M_Data1 = 0;

VIII. HPF

P_Com_Hig = 25; P_Com_Low = 1 .. 16 (1, said output 1 channel, 16, said output of 16 channels);

P_CH = P_Com_Low;

P_N = 3;

HPF_Slope = slope of the index table [P_Data1];

HPF_Freq = frequency index table [P_Data2 * 256 + P_Data3]; 10Hz ~ 16KHz

After successfully sending the machine to receive orders:

M_Cout = 1;

M_0 = 0;

M_Mode = 0;

M_Data1 = 0;

Note: the same output channel HPF_Freq less than LPF_Freq.

IX. LPF

P_Com_Hig = 26; P_Com_Low = 1 .. 16 (1, said output 1 channel, 16, said output of 16 channels);

P_CH = P_Com_Low;

P_N = 3;

LPF_Slope = slope of the index table [P_Data1];

LPF_Freq = frequency index table [P_Data2 * 256 + P_Data3]; 35Hz ~ 20KHz

After successfully sending the machine to receive orders:

M_Cout = 1;

M_0 = 0;

M_Mode = 0;

M_Data1 = 0;

Ten. LIMIT ON / OFF

P_Com_Hig = 28; P_ Com_Low = 1 .. 16 (1, said output 1 channel, 16, said output of 16 channels);

P_CH = P_ Com_Low;

P_N = 1; P_Data1 = 0 that BYPASS OFF; P_Data1 = 1, said BYPASS ON

After successfully sending the machine to receive orders:

M_Cout = 1;

M_0 = 0;

M_Mode = 0;

M_Data1 = 0;

XI. Limit-Attack

P_Com_Hig = 29; P_Com_Low = 1 .. 16 (1, said output 1 channel, 16, said output of 16 channels);

P_CH = P_Com_Low;

P_N = 1; P_Data1 range: 0 ~ 127

After successfully sending the machine to receive orders:

M_Cout = 1;

M_0 = 0;

M_Mode = 0;

M_Data1 = 0;

XII. Limit-Release

P_Com_Hig = 31; P_Com_Low = 1 .. 16 (1, said output 1 channel, 16, said output of 16 channels);

P_CH = P_Com_Low;

P_N = 1; P_Data1 range: 0 ~ 127

After successfully sending the machine to receive orders:

M_Cout = 1;

M_0 = 0;

M_Mode = 0;

M_Data1 = 0;

XIII. Limit-Threshold

P_Com_Hig = 33; P_Com_Low = 1 .. 16 (1, said output 1 channel, 16, said output of 16 channels);

P_CH = P_Com_Low;

P_N = 1; P_Data1 range: 0 ~ 127

After successfully sending the machine to receive orders:

M_Cout = 1;

M_0 = 0;

M_Mode = 0;

M_Data1 = 0;

Fourteen. Compressor ON / OFF

P_Com_Hig = 34; P_Com_Low = 1 .. 16 (1, said output 1 channel, 16, said output of 16 channels);

P_CH = P_Com_Low;

P_N = 1; P_Data1 = 0 expressed OFF; P_Data1 = 1, said ON

After successfully sending the machine to receive orders:

M_Cout = 1;

M_0 = 0;

M_Mode = 0;

M_Data1 = 0;

Fifth. Compressor-Attack

P_Com_Hig = 35; P_Com_Low = 1 .. 16 (1, said output 1 channel, 16, said output of 16 channels);

P_CH = P_Com_Low;

P_N = 1; P_Data1 = 0 ~ 127

After successfully sending the machine to receive orders:

M_Cout = 1;

M_0 = 0;

M_Mode = 0;

M_Data1 = 0;

Sixteen. Compressor-Release

P_Com_Hig = 36; P_Com_Low = 1 .. 16 (1, said output 1 channel, 16, said output of 16 channels);

P_CH = P_Com_Low;

P_N = 1; P_Data1 = 0 ~ 127

After successfully sending the machine to receive orders:

M_Cout = 1;

M_0 = 0;

M_Mode = 0;

M_Data1 = 0;

XVII. Comressor-threshold

P_Com_Hig = 37; P_Com_Low = 1 .. 16 (1, said output 1 channel, 16, said output of 16 channels);

P_CH = P_Com_Low;

P_N = 1; P_Data1 = 0 ~ 29

After successfully sending the machine to receive orders:

M_Cout = 1;

M_0 = 0;

M_Mode = 0;

M_Data1 = 0;

XVIII. Compressor-ratio

P_Com_Hig = 38; P_Com_Low = 1 .. 16 (1, said output 1 channel, 16, said output of 16 channels);

P_CH = P_Com_Low;

P_N = 1; P_Data1 = 0 ~ 127

After successfully sending the machine to receive orders:

M_Cout = 1;

M_0 = 0;

M_Mode = 0;

M_Data1 = 0;

XIX. Compressor-Boost

P_Com_Hig = 39; P_Com_Low = 1 .. 16 (1, said output 1 channel, 16, said output of 16 channels);

P_CH = P_Com_Low;

P_N = 1; P_Data1 = 0 ~ 3

After successfully sending the machine to receive orders:

M_Cout = 1;

M_0 = 0;

M_Mode = 0;

M_Data1 = 0;

Twenty Noise-Attack

P_Com_Hig = 40; P_Com_Low = 1 .. 16 (1, said output 1 channel, 16, said output of 16 channels);

P_CH = P_Com_Low;

P_N = 1; P_Data1 = 0 ~ 127

After successfully sending the machine to receive orders:

M_Cout = 1;

M_0 = 0;

M_Mode = 0;

M_Data1 = 0;

The twenty-first Noise-Release

P_Com_Hig = 41; P_Com_Low = 1 .. 16 (1, said output 1 channel, 16, said output of 16 channels);

P_CH = P_Com_Low;

P_N = 1; P_Data1 = 0 ~ 127

After successfully sending the machine to receive orders:

M_Cout = 1;

M_0 = 0;

M_Mode = 0;

M_Data1 = 0;

Twenty-two Noise-Threshold

P_Com_Hig = 42; P_Com_Low = 1 .. 16 (1, said output 1 channel, 16, said output of 16 channels);

P_CH = P_Com_Low;

P_N = 1; P_Data1 = 0 ~ 28

After successfully sending the machine to receive orders:

M_Cout = 1;

M_0 = 0;

M_Mode = 0;

M_Data1 = 0;

Xxiii. Program calls.

P_Com_Hig = 202; P_Com_Low = 1 .. 16 (program number);

P_CH = P_Com_Low;

P_N = 1; P_Data1 = 1;

After successfully sending the machine to receive orders:

M_Cout = 1;

M_0 = 0;

M_Mode = 0;

M_Data1 = 0;

Call the program, wait 2 minutes and then the other command operation, invoke the program automatically after the output channel mute.

Crossover Slope Form:

1: FLAT THRU

2:6 dB Butterworth

3:12 dB/0.5

4:12 dB/0.6

5:12 dB/0.7

6:12 dB/0.8

7:12 dB/0.9

8:12 dB/1.0

9:12 dB/1.1

10:12 dB/1.2

11:12 dB/1.3

12:12 dB/1.4

13:12 dB/1.5

14:12 dB/1.6

15:12 dB/1.7

16:12 dB/1.8

17:12 dB/1.9

18:12 dB/2.0

19:12 dB Bessel

20:12 dB Butterworth

21:18 dB Butterworth

22:18 dB Bessel

23:24 dB Butterworth

24:24 dB Bessel

25:24 dB Link_Riley

26:48 dB Butterworth

27:48 dB Bessel

28:48 dB Link_Riley

Q value table:

Q range: 0.4 ~ 128 [4 104]

0.4 ~ 1.0: step 0.1 4 ~ 10

1.2 ~ 5.0: step 0.2 11 ~ 30

5.5 ~ 10.0: step 0.5 31 ~ 40

11 ~ 20: step 1 41 ~ 50

22 ~ 128: step 2 51 ~ 104

```
Double Q_LABEL [] = {
0,1,2,3, / /    PEQ, hishelf, Loshelf, ALLPASS
 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0,
 1.2, 1.4, 1.6, 1.8, 2.0, 2.2, 2.4,
 2.6, 2.8, 3.0, 3.2, 3.4, 3.6, 3.8,
 4.0, 4.2, 4.4, 4.6, 4.8, 5.0, 5.5,
 6.0, 6.5, 7.0, 7.5, 8.0, 8.5, 9.0,9.5,
 10, 11, 12, 13, 14, 15, 16,
 17, 18, 19, 20, 22, 24, 26,
 28, 30, 32, 34, 36, 38, 40,
 42, 44, 46, 48, 50, 52, 54,
 56, 58, 60, 62, 64, 66, 68,
 70, 72, 74, 76, 78, 80, 82,
 84, 86, 88, 90, 92, 94, 96,
 98,100,102,104,106,108,110,
112,114,116,118,120,122,124,
126,128}
```

Frequency Table:

/ / 0 ~ 118 freq 10 ~ 100

/ / 119 ~ 237 freq 100 ~ 1000

/ / 238 ~ 357 freq 1k ~ 10k

/ / 358 ~ 394 freq 10k ~ 20k

/ / XOVER 0 ~ 395

/ / PEQ 35 ~ 395

```
double FREQ_LABEL [395] = {
 10.0,10.2,10.4,10.6,10.8,11.0,11.3
, 11.5,11.7,11.9,12.2,12.4,12.6,12.9
```

, 13.1,13.4,13.7,13.9,14.2,14.5,14.7
, 15.0,15.3,15.6,15.9,16.2,16.6,16.9
, 17.2,17.5,17.9,18.2,18.6,18.9,19.3
, 19.7,20.1,20.5,20.9,21.3,21.7,22.1
, 22.5,23.0,23.4,23.9,24.3,24.8,25.3
, 25.8,26.3,26.8,27.3,27.8,28.4,28.9
, 29.5,30.1,30.7,31.3,31.9,32.5,33.1
, 33.8,34.4,35.1,35.8,36.5,37.2,37.9
, 38.6,39.4,40.1,40.9,41.7,42.5,43.4
, 44.2,45.1,45.9,46.8,47.7,48.7,49.6
, 50.6,51.6,52.6,53.6,54.6,55.7,56.8
, 57.9,59.0,60.1,61.3,62.5,63.7,65.0
, 66.2,67.5,68.8,70.2,71.5,72.9,74.3
, 75.8,77.2,78.7,80.3,81.8,83.4,85.0
, 86.7,88.4,90.1,94.0,95.0,97.0,99.0

, 101,103,105,107,109,111,114
, 116,118,120,123,125,127,130
, 132,135,138,140,143,146,149
, 152,154,157,161,164,167,170
, 173,177,180,184,187,191,195
, 198,202,206,210,214,218,223
, 227,231,236,241,245,250,255
, 260,265,270,275,281,286,292
, 297,303,309,315,321,327,334
, 340,347,354,360,367,375,382
, 389,397,405,412,420,429,437
, 445,454,463,472,481,490,500