Flipdots display control protocols - October 2023

SERIAL (FOR LAN – see next page)

Frame for serial

0x80	Command	Address	Data	0x8F

Command –

Command	Numer of data bytes	Refresh?	Configuration
0x82	0	YES	Format (0x80, 0x82, 0x8F) – refresh of all connected displays
0x83	28	YES	28x7
0x84	28	NO	28x7
0x87	7	YES	7x7
0x92	14	YES	14x7
0x93	14	NO	14x7

Address – device address , 255 (0xFF) is a broadcast address – all devices are receiving this transmission

Data – transmitted content of a display; number of data bytes depends on size of a display. One byte is one strip of dots (7 dots). LSB is upper dot, MSB (the 7th) is lower dot. The most significant byte (8th) is ignored and should be set to zero.

Remarks:

Refresh = YES: DATA bytes are being shown on displays as soon as they are received

Refresh = NO: DATA bytes are being stored in a memory and shown on displays as soon as 0x80 / 0x82 / 0x8F sequence is received. This option helps to synchronize presentation of data. This is not supported by 7x7 displays.

LAN

There is a software using which you can set IP address / port of the panel. By default it is DHCP and after running the software you can see IP address / port. Ask for link to download the software.

In order to set content you must connect with UDP protocol to programmed IP / port.

Data frame is the following:

[**protocol command][32 bit request id][pix command][pix data lenght][… data …][crc]**

protocol command -> 102 (0x66).

32 bit request id -> 4 arbitarry bytes – allows the controller to determine which request the response is referring to. You can put here for example [0xaa][0xaa][0xaa][0xaa]

pix command -> 0x92 to show immediately the content.

pix data lenght -> for 7x14 display it is 14 (0xe), for 7x28 it is 28.

data -> data to show . Together with [pix command] and [pix data length] there must be a total of 255 bytes , the first [pix data length] bytes are used to show content, the rest is just filling up the qty and can be whatever, for example 0x00.

crc -> Control sum of the frame, calculated from the field [command] [requestId] and [data]. It is calculated from 0xFFFF polynomial.

This website is explaining how to do it. https://www.lammertbies.nl/comm/info/crc-calculation
This is a sum from the field CRC-CCITT (0xFFFF), with bytes swapped in place like in Modbus.

Example to show all dots black on 7x14 panel:

66 e3 af dc 32 92 0e 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 bd 5b

All dots white for 7x14 panel:

66 e3 af dc 32 92 0e 7f 7f 7f 7f 7f 7f 7f 7f 7f 7f 7f 7f 7f 7f 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 18 b0