PUSHING CONTROL BOUNDARIES

Performance Guide
v3.0.1-1

05-04-2025

TABLE OF CONTENTS

INTRODUCTION

As pixel-controlled lighting gains more popularity, and as projects are getting bigger and bigger, but above all, more demanding, it becomes increasingly more important to pay attention to detail.

Working with SPI controlled LED products can sometimes be a bit challenging because it's less straight forward than commonly used lighting control protocols, such as DMX. We at DiGidot Technologies, as a manufacturer of professional pixel control hardware, feel that it's of great importance to share our knowledge with our customers in order to get the most out of our DiGidot products.

Buying a DiGidot Controller because it can control a lot of channels, may lead to disappointment because the end result often depends on entirely different aspects. In most cases our products are not a limiting factor when it comes to performance. The things that do have a major impact are; the content quality and control system like a pixel control software, console or media server, but also proper wiring, grounding, signal integrity, network performance, and last but not least the type of SPI protocol/LED IC.

In this document we will cover performance related to the SPI protocols and DiGidot hardware. In order to understand this fully, we also dig into the world of SPI. We assume that all other factors mentioned above are evidentially and are not within our sphere of influence, although they should not be discarded.

We believe that this document will help to engineer pixel-controlled projects a lot better and that it will contribute to better performing DiGidot systems.

PREFACE

Welcome to the DiGidot Performance Guide. This guide provides insights in SPI controlled lighting in general and lists maximum frame rates (FPS) of our controllers.

The maximum speeds in this document are expressed in FPS and relate to the number of full frames of data sent from the controller the first IC, per second.

It is important to be able achieve certain frame rates for specific applications.

E.g.: When fast moving video content is recorded at high frame rates (e.g. 60 fps) and then converted to Art-Net (using a media server for example), you want to be able to use all recorded frames on your LED pixels. In order to do this, the DiGidot controller must, ideally, send the same number of frames per second on its output as on it's receiving on it's input.

A common misunderstanding is that frame rates relate to the visual frames per second, same as we're used to apply this term to video. In the case of SPI however, we talk about the amount of full data sets for all controlled pixels per second. The visual frequency of light is determined by the PWM frequency of the LED chips itself. This guide will not cover any of the PWM driver related specifications.

Although we primarily talk about the speed of the data, this may still have an effect on visual performance.

The DiGidot controllers can be considered to be a tool, with very few limitations. Just like with any tool, if it's not used correctly, the end result will probably not quite be as expected. The only limits the controllers has are; The total amount of channels you can control (this is determined by the device license) and the physical port limit of 8 universes. The rest is basically up to the user and the laws of physics (sorry we did not make them).

USED TERMS

Here are some frequently used terms in respect to pixel controlled LED lights.

Controllers	Refers to a LED controller from DiGidot, which is either a DiGidot C1, C4 or the PxLNet Node.
FPS	Frames per second (output frames of data, not from the PWM driver of the LED itself)
Pixel	This is one LED or group of LED's of which every color can be controlled individually. E.g. RGB or RGBW.
DMX	This is a communication protocol between lighting consoles or PC's and fixtures.
Art-Net	This is a DMX over Ethernet communication protocol between Lighting Control software or consoles and LED controllers.
sACN	Basically the same as Art-Net, but different 'under the hood' properties.
SPI	Serial Peripheral Interface; This is an communication protocol that is used to send data between microcontrollers and other IC's.
IC	Integrated Circuit; this is an electronic component which houses a readymade circuit that is able to perform one specific task. An IC can be placed separately on a PCB or integrated in a LED package to control/drive a LED or segment/array of LEDs.
IC- /SPI protocol	Type of control protocol that is used by an IC.
Universe	Typically a universe is referred to as one DMX control network consisting of 512 control channels, also typically associated with one DMX output. We also refer to a universe as one set of 512 channels that belong to specific Art-Net / sACN or DMX universes. But when we mention more than one universes on one output for example, we always refer to them as amount of control channels and not necessarily as DMX universes.

SPI EXPLAINED

Serial Peripheral Interface is a type of synchronous serial communication protocol designed to communicate at high speeds between electronic components, such as processors, micro controllers, ICs, etc.

SPI was never designed to leave a PCB (Printed Circuit Board) and cover large distances through cables, hence it comes with some limitations that we have to take into account, such as a very limited cable lengths.

There are countless variants of the SPI protocol on the market and some of them are used to control LED pixels. The DiGidot controllers supports well over 60 of these industry standard SPI protocols. This allows us to control almost any SPI controlled lighting product on the market.

Every SPI controlled product that contains LED’s, has either an internal or external IC (integrated circuit). This controls the logic of the SPI communication protocol and also has a driver circuit integrated to send PWM signal to light the actual diode.

Similar to DMX, SPI is a line bus protocol of which the information is only send in one direction. Unlike DMX, SPI IC’s cannot be addressed. Instead, SPI IC’s can only do one thing and they all do it in the same way. They take the first data packet, process and execute its information, and send the rest of the information to the next IC in line. This process repeats itself until there are no more IC’s in line. This also means that the order of data packets sent to the LED IC’s must be in the exact right order as the pixels in your projects. This is done by pixel mapping (usually in your pixel software or console) and on the input/output configuration page of the DiGidot Controllers where you can assign specific control channels to the output ports.

Another big difference is that SPI is not limited to a certain number of channels such as DMX, where you only have 512 channels. SPI protocols / LED IC’s also work on different speeds in comparison to DMX. This usually varies between 400 kHz and 2 MHz. DMX works at 250 kHz and is therefore, in general, much slower than any SPI protocol on the market.

The basics of working with SPI controlled products is quite simple, the only thing that makes it more complex are the differences among the protocols.

DMX AT HIGHER SPEEDS

Since the release of DiGidot Controller Firmware 2.3.6 we also support faster variants of DMX: DMX at 500 kHz (sometimes referred as DMX1024), DMX at 750 kHz and DMX at 1 MHz. (sometimes referred as DMX2048). With this, you can archive much higher framerates, especially when you're using many universes on a single port. It's very important to check the datasheet of the manufacturer of the LED IC to check till which speed it supports DMX.

Currently, only DMX TTL supports these higher speeds. These signals are **TTL** which means: 1 wire and they can cover only a short distance (up to 2 meters). You can make it DMX by using our PxLNet Transmitter.

OVERVIEW OF SPI PROTOCOLS

In this chapter, we going to take a look at various SPI protocols and how they compare to each other. Most LED IC's can be divided into 3 major categories of SPI:

1. Data only (also called 'Data')
2. Data/Backup
3. Clock/Data.

The DiGidot Controllers supports all these protocols.

Data only

The most common used SPI protocol for LED's is **Data only** or just **Data**. This protocol uses a single wire to communicate. On this wire, the actual signal creates a digital bit that is either '1' or '0' depending on the 'duty cycle', which is the ratio of how long the signal stays **HIGH** (5 volt) and **LOW** (0 volt). A commonly used protocol of this type, is the **WS2812**. LED products with this protocol have 3 wires in total; 2 are for the power supply (usually on the sides, indicated with **GND** or **-** and **5V** / **12V** / **24V** or **+**), depending on how much voltage the IC needs) and one wire for the data signal (often marked as **D** or **Di**). The DiGidot C1 can output 2 lines of this type, the C4 can output 4 while our PxLNet Node can output 16 lines. Both will send this in parallel. The downside of this protocol is that, if only one LED/IC in the string breaks, all consecutive LED's will fail and show the last received color or stay completely off. This shortcoming is solved with the 'Data/Backup' protocols.

Data/Backup

The **'Data/Backup'** protocol is almost identical to the 'Data' protocol but with one major advantage: If one LED in the string fails, all consecutive LED's keep working but with one pixel shifted.

E.g.: You have a total of 100 LEDs on a pixel-controlled strip. If LED #50 fails, then the LED data of LED #50 will be shown on LED #51 and the data for LED #51 will be shown at LED #52 etc. So it creates a shift in the patch if any LED fails.

Most of the time, the shifted pixel is no problem unless you're using the LED strip for something like a screen/display where the LED's are patched with the 'snake' method. Then, the shifted pixel will shift every other pixel after that in the line of the LED screen after that which is why for this type of application, 'Data/Backup' is mostly avoided. If two adjacent pixels fail, the consecutive LED's still fails and will show the last received color or be completely off depending on the LED IC. The rest of the properties are exactly the same as the 'Data' protocol. Be aware that there are 2 types of 'Data/Backup' implementations. Most of them, you only need to connect the data line to the DiGidot controller (so it uses 1 output port). But some LED IC's (this applies to the SK6822 and the WES943) use 2 ports to output the protocol correctly. This will halve the pixel capacity that the DiGidot controller can output.

Clock/Data

The last major SPI protocol is **Clock/Data.** This protocol uses 2 wires, one for the data and the other for a clock signal. Instead of relying on the 'duty cycle' of the signal to determine if the sent bit is a '0' or '1', it uses an extra clock signal. Every time the clock signal goes from **LOW** to **HIGH**, the voltage is read at that moment on the data line. The addition of a clock line, has the effect it can transfer much more data in the same time compared to a Data (+ backup) protocol. The downside however, is that it needs 2 wires per output, which means that a DiGidot C1 can only have 1 line of this protocol, the C4 can have 2 lines and the PxLNet Node can output 8.

Identifying SPI protocols

Which SPI protocol do you have? There are 2 ways to identify which type of LED IC you have.

1. The model name could be printed on the strip itself but you may also find this information in the supplied datasheet or on the packaging.

2. If your product uses external IC's, most of the time, you will find the LED IC model printed on the IC chips themselves (usually the large black components on the strip). You may however need a magnifier to be able to read the small letters.

If you know the LED IC model, you can use the **Alphabetic LED IC table**.

If you don't know the LED IC, you can which letters and how many pins / solder pads the LED strip has on the start:

* If your LED product has 3 connections or solder pads, it's almost certainly a **'Data'** protocol.
* If your LED product has 4 connections, it's most likely a **'Clock/Data'** or **'Data/Backup'** protocol.

You can also check the letters next to the solder pads on your LED product. If you see the letters **'C'** or **'CLK'**, it's a **'Clock/Data'** protocol and if you see a **'B', 'D2' or 'FIN'** it's a 'Data/Backup' protocol.

PERFORMANCE GUIDELINES

Introduction

In order to calculate the expected performance of your system, there are some factors that are important to be aware of.

It is important to pay extra attention to the cable length, resistance, type of cable and the supply voltage to the LED IC's.

Because SPI is a very fast protocol, it is essential to use good quality cables in order to avoid signal distortion. We advise to use CAT6 network cable or better, or a good quality official DMX Cable (110Ω). Our recommendation is to not exceed wire gauges of 0,5mm^2 or AWG 20.

Keep in mind that, if the first pixel of the line is further than two meters (6.5 ft) away, you should use the DiGidot PxLNet Transceiver Kits to overcome this problem. The DiGidot Transceiver Kits can be used to send the SPI data up to 250 meters away and thus will increase the range of your system. The PxLNet Node, as the name implies, already has the long-range technology built in.

Determine the maximum frame rate

The maximum frame rate that can be achieved by a DiGidot Controller, is determined by the frame rate of the port that controls the greatest number of channels. For the PxLNet Node, it applies to the greatest number of channels on an output from a single Bay, that determines the performance of the other 3 ports of that same bay. So you could get different performance numbers for each bay of the PxLNet Node.

We recommend to spread out all channels as evenly as possible to get the best performance from your controller. Sometimes, it's more convenient to make a very long run instead of using multiple runs of equal length. This will save a lot of cables. But if you do this, it will dramatically decease the maximum FPS you can get on your DiGidot Controller.

Bits per channel

Most LED IC's have an 8 bit resolution/color depth per channel which gives you 256 possible dimming values for each channel (same as DMX). That's enough for most applications. Some LED IC's have 16 bits which gives you 65536 possible dimming values for each channel. The most important reason to choose 16 bit is to have much smoother dimming curve. This can be best observed in very slow dimming transitions and also, in lower brightness regions, when you're using the LED in a dark environment like a theatre.

Using a 16 bit LED IC, only makes sense for a DiGidot controller when the **input** (Art-Net or sACN signal) is also 16 bit, or if you enable **'gamma correction'** on the controller itself. The 'gamma correction' method is highly recommended as this significantly decreases the network load and increases the software-/console compatibility.

This option can be found by navigating to the web interface of the controller, go to **Settings** -> **Input/Output configuration** -> click on the top-right **3 dots,** -> **Global configuration** -> Enable **Gamma correction**.

> Sending 16 bit directly from Art-Net software or a console is not recommended as this significantly increases your network load.

If you want to use a 16 bit LED IC, you have to remember that, you're always sending twice the amount of bits (information) on the **output,** compared to an 8 bit LED IC with the same amount of channels/pixels. This means that it also takes twice the amount of time to send out all channels. Effectively, if you have 2 SPI IC's that are both identical, but one is 8 bit and the other is 16 bit, the 8 bit variant is twice as fast. For the same number of channels, you can double the FPS or you can output twice the number of channels with the same FPS, when using 8 bits instead of 16. The type of the LED protocol (like Data or Clock/Data) in this case doesn't matter. If you don't exceed 2 universes (or 1024 channels) on the port where you're sending the most channels from, 16 bit will still give you around 60 FPS most of the time for our supported SPI IC's.

> Please note that you will most likely not be able to tell the difference between 8 and 16 bit dimming when you are displaying fast moving content, or in daylight situations.

SPI protocol type

Another important factor regarding the performance is the SPI type. Every LED IC that a DiGidot controller can output, can be divided into 3 categories: **Data only**, **Data/Backup** and **Clock/Data**. Check out the previous chapter to get more technical information about these protocols. The performance difference isn't huge in comparison with 8 bit vs 16 bit. When you use a 'Clock/Data' protocol, the performance can be increased with 25 % on average.

Special case SPI IC's

A few SPI IC's require extra bits to be sent for each frame and in some cases for every channel. This greatly reduces the performance because it needs to send more information for every frame. This can affect the performance by -10 % up to -100 %, in comparison to a 'Data' protocol. Currently, there a 2 LED IC's that have these penalties: The **CX808** and the **MBI6120**.

SELECTING THE RIGHT LED OR SPI PROTOCOL/IC

For choosing the right LED/SPI IC and therefore the right product for your installation, it's important to first determine the required dimming resolution (8 or 16 bits).

1. Dimming resolution

Do you need 16 bit dimming (65536 steps) or is 8 bit (256 steps) enough? Most of the time, 16 bit is only used If you're using the LED strip in a (very) dark environment and/or, you have a very slow effect running. Also, the Art-Net/sACN software/control system should send out 16 bit, or you can use the 'gamma correction' option in the controller (see previous chapter on how to enable that feature). In almost every case, when comparing 2 'similar' SPI IC's where one is 8 bit and the other is 16 bit, the 8 bit will perform twice as fast when compared to the 16 bit variant.

So only use 16 bit if: If your content will be slowly changing and if you're using the LED strips in a low light environment (theatre, outside in the night, etc.),

For everything else: use a 8 bit LED IC.

2. Protocol type

The next step Is to choose between one of the two major SPI protocol types.

- Clock/Data
- Data only / Data backup

To figure out which protocol is the preferred one, here are some guidelines:

1. If you want to control 64 universes or more from a single DiGidot PxLNet Node, more than 8 universes on a C1 or more than 16 universes on a DiGidot C4, choose a **Data** protocol LED.
2. In case you want to make long cable runs and send between 6-8 universes per output, it makes sense to use a **Clock/Data** protocol.

If these things are not relevant, it doesn't really matter which protocol you choose. On average, the 'Clock/Data' will be about 25 % faster in most cases with the same number of channels.

PERFORMANCE OVERVIEWS

Table explanation

The tables in the next chapter of this document, display the maximum FPS you can achieve on a single bay of a PxLNet Node, or on a DiGidot C1 or C4 controller, based on the highest number of channels on a single output port.

E.g.: Here is an example of a single Bay of the PxLNet Node / DiGidot C1 /C4, I/O configuration where 8 universes are configured with a WS2812 LED IC.

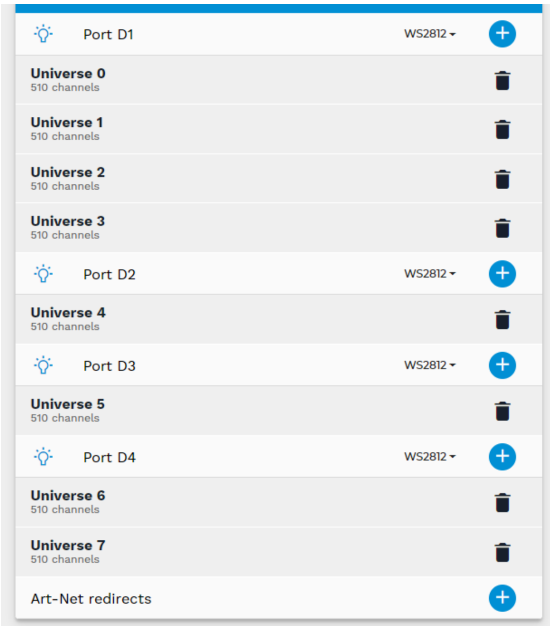

Figure 1a - Maximum FPS for all ports is 40

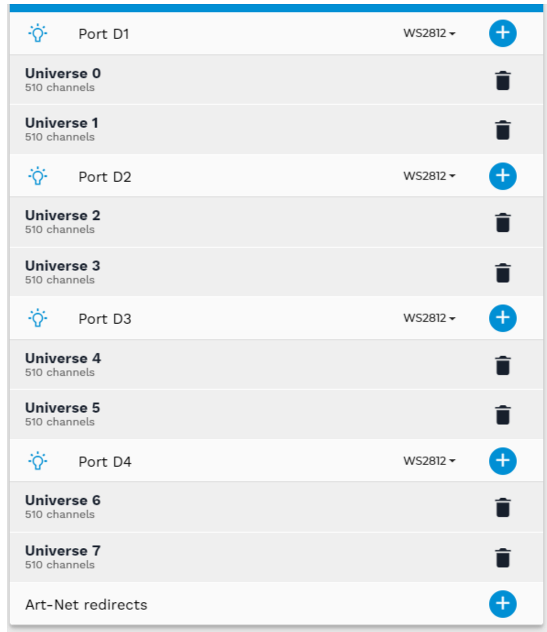

Figure 1b Maximum FPS for all ports is 75 !

From image **1a**, on port D1, there are 4 universes configured while on port D2, D3 there is 1 universe on each and on D4, there are 2 universes configured. The **"Highest universes on a single port"** in this case is 4. From the table on the last page, we can see that for 4 universes maximum on a single port, would result in an FPS of **40** for the DiGidot C1/C4 or for that specific bay.

If we would rearrange the universes so they are spread evenly, so 2 universes on each port, like on image **1b**, this number will be 2. A maximum of 2 universes means the maximum frame rate is **75** ! So about twice as fast! With this example, it should be clear that the best performance for every installation is always achieved with evenly spread universes or channels.

Please note that the displayed frame rates are measured and rounded to even numbers for easy comparison and include safety margins in order to assure that the indicated values are realistic.

In following tables, the first column always lists the LED IC's/SPI protocols.

For 'Data' and 'Data/Backup', a 'port' means a single output port (e.g. D1).

For Clock/Data, a 'port ' means the combination of two data lines. (e.g. D1 + D2).

All tables are sorted with the best preforming protocols/IC's on top and the lowest performing ones on the bottom.

The last table is an exception because it is sorted in alphabetical order.

Per unit

Every chapter with a table starts with "X universes per unit". "A unit" can be 2 things;

A DiGidot C1 / C4:
It means the whole device. So, you should see this text as "per device".

A PxLNet Node:
It means a single bay. The node has a total of 4 bays and each bay has Its own maximum performance. For example: If you configure P1 till P4 (so bay 1) with only 1 universe per port, and P5 till P16 (so bay 2, 3 and 4) with 6 universes per port, only the ports of bay 1 will operate at a fast FPS of around 100 while bay 2, 3 and 4, all the ports of those, will operate at the much slower 25 FPS. So all the bays have independent performance numbers.

Legend

To indicate the performance level you can expect with a certain number of channels, we use colors. The following colors are used in the tables:

Green	Expect no problems
Orange	You can use this but you may experience drop-outs while loading the interface or having the monitor page open 24/7.
Red	You're close or at 100% system load. We don't recommend to use this option for installations. Sometimes, the number of FPS in red can't be achieved at all depending on how many FPS you're sending Art-Net

If the cell in the table is empty, the output performance is less than 20 FPS. This should be avoided at all time.

'CLOCK/DATA' IC'S FOR 1-12 UNIVERSES PER UNIT

8 Bit

LED IC	Bits	1	2	3	4	5	6	7	8
LD1510	8	100	100	100	100	70	60	50	45
WS2801	8	100	100	100	100	70	60	50	45
WS2803	8	100	100	100	100	70	60	50	45
SM16716	8	100	100	100	100	70	60	50	40
SM16726	8	100	100	100	100	70	60	50	40
PC5XS301V0500	8	100	100	100	100	70	60	45	40
APA102 8bit	8	100	100	100	60	50	45	40	35

16 bit

LED IC	Bits	1	2	3	4	5	6	7	8
APA102	12	100	100	100	60	50	45	35	30
HD107S	12	100	100	100	60	50	45	40	35
DM412	16	100	80	60	45	35	30	25	20
MY9231	16	100	100	60	45	35	30	25	20
MY9291	16	100	100	60	45	35	30	25	20
MY9221	16	100	100	60	40	35	25	20	
MBI6024	16	100	45	25	20	20			

Please note that for 16 bit LED IC's, we calculate based on 16 bit color channels. This means that you don't need do half the number of universes listed above, you can calculate with these numbers.

'DATA' IC'S FOR 1-16 UNIVERSES PER UNIT

LED IC	Bits	Protocol	1	2	3	4	5	6	7
UCS512XX (1 Mhz)*	8/16	Data	100	75	50	40	30	25	20
DMX-TTL (1Mhz)*	8	Data	100	75	50	40	30	25	20
SK6805	8	Data	100	75	50	40	30	25	20
SK6812 (+RGBW)	8	Data	100	75	50	40	30	25	20
SK6813	8	Data	100	75	50	40	30	25	20
SK6815	8	Data	100	75	50	40	30	25	20
SM16703	8	Data	100	75	50	40	30	25	20
SM16704	8	Data	100	75	50	40	30	25	20
SM16712	8	Data	100	75	50	40	30	25	20
SM16824	16	Data	100	75	50	40	30	25	20
UCS7604*	16	Data/Backup	100	75	50	40	30	25	20
UCS7804*	16	Data/Backup	100	75	50	40	30	25	20
WES9412	8	Data	100	75	50	40	30	25	20
WS2808	8	Data/Backup	100	75	50	35	30	25	20
WS2812	8	Data	100	75	50	40	30	25	20
WS2813	8	Data/Backup	100	75	50	40	30	25	20
WS2814	8	Data/Backup	100	75	50	35	30	25	20
WS2815	8	Data/Backup	100	75	50	40	30	25	20
WS2851	8	Data	100	75	50	35	30	25	20
WS2852	8	Data	100	75	50	35	30	25	20
UCS2903	8	Data	100	75	50	35	30	25	20
UCS2904	8	Data	100	75	50	35	30	25	20
UCS2909	8	Data	100	75	50	35	30	25	20
WS2811 (800Khz)	8	Data	100	75	50	35	30	25	20
WS2812B	8	Data	100	75	50	35	30	25	20
TM1804 (800Khz)	8	Data	100	75	50	35	30	25	20
TM1812	8	Data	100	75	50	35	30	25	20
TM1814	8	Data	100	75	50	35	30	25	20
UCS1903 (800Khz)	8	Data	100	75	50	35	30	25	20
TM1903	8	Data	100	75	50	35	30	25	20
TM1914	8	Data/Backup	100	75	50	35	30	25	20
TX1818	8	Data	100	75	50	35	30	25	20
UCS1904	8	Data	100	75	50	35	30	25	20
UCS1909	8	Data	100	75	50	35	30	25	20
INK1003	8	Data	100	75	50	35	30	25	20
GS8206	8	Data/Backup	100	75	50	35	30	25	20

LED IC	Bits	Protocol	1	2	3	4	5	6	7
GS8208	8	Data/Backup	100	75	50	35	30	25	20
TM1824	8	Data	100	75	50	35	30	25	20
UCS1903 (800Khz)	8	Data	100	75	50	35	30	25	20
UCS1904	8	Data	100	75	50	35	30	25	20
UCS1909	8	Data	100	75	50	35	30	25	20
INK1003	8	Data	100	75	50	35	30	25	20
GS8206	8	Data/Backup	100	75	50	35	30	25	20
GS8208	8	Data/Backup	100	75	50	35	30	25	20
CS8812	8	Data/Backup	100	75	50	35	30	25	20
APA104	8	Data	100	75	50	35	30	25	20
APA105	8	Data	100	75	50	35	30	25	20
APA109	8	Data	100	75	50	35	30	25	20
UCS2912	8	Data	100	75	50	35	30	25	20
UCS2903	8	Data	100	75	40	30	25	20	
DMX TTL(750 kHZ)*	8	Data	100	75	40	30	25	20	
UCS512 (750 kHZ)*	8/16	Data	100	75	40	30	25	20	
UCS2909	8	Data	100	75	40	30	25	20	
WS2812S	8	Data	100	75	40	25	25	20	
APA106	8	Data	100	55	35	25	20		
SK6812	8	Data	100	55	35	25	20		
SK6822	8	Data/Backup	100	55	35	25	20		
TM1934	8	Data/Backup	100	55	35	25	20		
TM1809	8	Data	100	55	35	25	20		
TM1934	8	Data/Backup	100	55	35	25	20		
WS2818	8	Data/Backup	100	55	35	25	20		
WES943	8	Data/Backup	100	50	35	25	20		
TM1803	8	Data	75	40	25	20			
TM1804 (400Khz)	8	Data	75	40	25	20			
UCS1903 (400Khz)	8	Data	75	40	25	20			
WS2811 (400Khz)	8	Data	75	40	25	20			
DMX TTL (500 kHz)	8	Data	75	40	25	20			
UCS512 (500 kHz)*	8/16	Data	75	40	25	20			
CX808	8	Data	60	30	20				
DMX TTL(250 kHz)*	8	Data	40						
DMX	8	Data	40						
UCS512 (250Khz)*	8/16	Data	40						

16 bit 'Data'

LED IC	Bits	Protocol	1	2	3	4	5	6	7
UCS512XX* (1 Mhz)	16*	Data	100	75	50	40	30	25	20
UCS7604*	16*	Data/Backup	100	75	50	35	30	25	20
UCS7804*	16*	Data/Backup	100	75	50	35	30	25	20
GW6205 (800Khz)	12	Data	100	50	35	25	20		
UCS5603	12	Data/Backup	100	50	35	25	20		
BS0901	16	Data	80	50	30	20			
UCS9812	16	Data	70	40	25	20			
UCS8904	16	Data	70	40	25	20			
UCS8903	16	Data	70	40	25	20			
GW6205 (400Khz)	12	Data	50	25	20				
MBI6120	16	Data	45	20					

LED IC'S FOR 17-24 UNIVERSES PER UNIT

For this many universes you have to limit every port to about 6 universes (3072 channels) per port. Even distribution of the channels is highly recommended and necessary to get the best performance. Only 8 bits LED IC's are supported.

> **IMPORTANT**
> The Art-Net/sACN FPS rate of your software or console must be equal or lower than the FPS rates in this table.

LED IC	Bits	Protocol	4	5	6
DMX TTL (1 MHz)*	8	Data	40	30	25
UCS512XX (1 MHz)*	8/16	Data	40	30	25
SK6805	8	Data	40	30	25
SK6812 (+ RGBW)	8	Data	40	30	25
SK6813	8	Data	40	30	25
SK6815	8	Data	40	30	25
SK6851	8	Data	40	35	25
SM16703	8	Data	40	30	25
SM16704	8	Data	40	30	25
SM16712	8	Data	40	30	25
SM16824	16	Data	40	30	25
UCS7604*	16	Data/Backup	40	30	25
UCS7804*	16	Data/Backup	40	30	25
WES9412	8	Data	40	30	25
WS2813	8	Data/Backup	40	30	25
WS2815	8	Data/Backup	40	30	25
WS2812	8	Data	40	30	25
UCS2904	8	Data	35	30	25
WS2811 (800Khz)	8	Data	35	30	25
WS2808	8	Data/Backup	35	30	25
WS2812B	8	Data	35	30	25
WS2814	8	Data/Backup	35	30	25
WS2851	8	Data	35	30	25
WS2852	8	Data	35	30	25
TM1804 (800Khz)	8	Data	35	30	25
TM1812	8	Data	35	30	25
TM1814	8	Data	35	30	25
TM1903	8	Data	35	30	25
TM1914	8	Data/Backup	35	30	25

LED IC	Bits	Protocol	4	5	6
TX1818	8	Data	35	30	25
UCS1903 (800Khz)	8	Data	35	30	25
UCS1904	8	Data	35	30	25
UCS1909	8	Data	35	30	25
GS8206	8	Data/Backup	35	30	25
GS8208	8	Data/Backup	35	30	25
CS8812	8	Data/Backup	35	30	25
INK1003	8	Data	35	30	25
APA104	8	Data	35	30	25
APA105	8	Data	35	30	25
APA109	8	Data	35	30	25

RECOMMENDED LED IC'S

This is our recommendation list of LED IC's. We have 2 lists: 'Best overall' and 'Best in speed'. The first one is a great LED IC in terms of dimming, reliability and speed. The second category is more focused to push as many channels from 1 output as possible.

Best overall (speed & quality)

Top tier (16 bit and fast):
UCS7604* / UCS7804* / SM16724*
UCS512XX* (16 bit)

Good (8bit and fast):
UCS512XX* -> 8 bit
WS2813 / WS2814 / WS2815
SK6812RGBW
SM16704 / SM16704

Best in speed (for longer data runs)

8 bit (faster, normal dimming):
SM16716 / SM16724
WS2801 / WS2803

16 bit (better dimming but slower):
LD1510
MY9221 / MY9231 / MY9291

COMPLETE IC OVERVIEW (ALPHABETICAL)

LED IC	Volt	Bits	Protocol	1	2	3	4	5	6	7	8
APA102	5	12	Clock/Data	100	100	100	60	50	45	35	30
APA102 (8 bit)	5	8	Clock/Data	100	100	100	60	50	45	40	35
APA104	5	8	Data	100	75	50	35	30	25	20	
APA105	5-24	8	Data	100	75	50	35	30	25	20	
APA106	5	8	Data	100	55	35	25	20			
APA109	5	8	Data	100	70	50	35	30	25	20	
BS0901	24	12	Data	80	50	30	20				
CS8812	12	8	Data/Backup	100	75	50	35	30	25	20	
CX808	5	8	Data	60	35	20					
DM412	5	16	Clock/Data	100	80	60	45	35	30	25	20
DMX		8	Data	40							
DMX TTL* (250kHz)		8	Data	40	20						
DMX TTL* (500 kHz)		8	Data	80	40	20					
DMX TTL* (750 kHz)		8	Data	100	60	40	30	25	20		
DMX TTL* (1 MHz)		8	Data	100	75	50	40	30	25	20	
GS8206	12	8	Data/Backup	100	75	50	35	30	25	20	
GS8208	12	8	Data/Backup	100	75	50	35	30	25	20	
GW6205 (400Khz)	5	12	Data	50	25	20					
GW6205 (800Khz)	5	12	Data	100	50	35	25	20			
HD107S	5	12	Clock/Data	100	100	100	60	50	45	35	30
INK1003	5	8	Data	100	75	50	35	30	25	20	
LB1908	12	8	Data/Backup	100	70	50	40	30	25	20	
LD1510	12	8	Clock/Data	100	100	100	100	65	60	50	45
LPD1101	5	5	Clock/Data	100	100	100	100	70	70	65	60
LPD6803	5	5	Clock/Data	100	100	100	100	70	70	65	60
MBI6024	5-12	16	Clock/Data	100	45	25	20				
MBI6120	5-12	16	Data	45	20						
MY9221	5	16	Clock/Data	100	100	60	40	35	30	25	20
MY9231	5	16	Clock/Data	100	100	60	45	35	30	25	20
MY9291	5	16	Clock/Data	100	100	60	45	35	30	25	20
P9883	5	8	Data	100	70	50	40	30	25	20	
PC5XS301V0500	5	8	Clock/Data	100	100	100	100	65	60	45	40
SK6805	5	8	Data	100	70	50	40	30	25	20	
SK6812 (+ RGBW)	5	8	Data	100	70	50	40	30	25	20	
SK6813	5	8	Data	100	70	50	40	30	25	20	
SK6815	5	8	Data	100	70	50	40	30	25	20	
SK6818	5	8	Data	100	55	35	25	20			
SK6822	5	8	Data/Backup*	100	55	35	25	20			

LED IC	Volt	Bits	Protocol	1	2	3	4	5	6	7	8
SK6851	24	8	Data	100	70	50	40	30	25	20	
SM16703	12	8	Data	100	70	50	40	30	25	20	
SM16704	12	8	Data	100	70	50	40	30	25	20	
SM16712	5	8	Data	100	70	50	40	30	25	20	
SM16716	5	8	Clock/Data	100	100	100	100	70	60	50	40
SM16726	5-12	8	Clock/Data	100	100	100	100	70	60	50	40
SM16824	5-24	16*	Data	100	70	50	40	30	25	20	
TM1803	5	8	Data	70	40	25	20				
TM1804 (400Khz)	5-24	8	Data	70	40	25	20				
TM1804 (800Khz)	5-24	8	Data	100	70	50	35	30	25	20	
TM1809	5	8	Data	100	75	35	25	20			
TM1812	24	8	Data	100	75	50	35	30	25	20	
TM1814	24	8	Data	100	75	50	35	30	25	20	
TM1824	24	8	Data	100	75	50	35	30	25	20	
TM1903	5-24	8	Data	100	75	50	35	30	25	20	
TM1914	24	8	Data/Backup	100	75	50	35	30	25	20	
TM1934	5	8	Data/Backup	100	50	35	25	20			
TX1818	12	8	Data	100	75	50	35	30	25	20	
UCS1903 (400Khz)	12-24	8	Data	75	40	25	20				
UCS1903 (800Khz)	12-24	8	Data	100	75	50	35	30	25	20	
UCS1904	12-24	8	Data	100	75	50	35	30	25	20	
UCS1909	12-24	8	Data	100	75	50	35	30	25	20	
UCS2903	12-24	8	Data	100	50	50	35	30	20		
UCS2904	12-24	8	Data	100	75	50	35	30	25	20	
UCS2909	12-24	8	Data	100	50	50	35	30	20		
UCS2912	12-24	8	Data	100	75	50	35	30	25	20	
UCS512XX*- 250kHz	12-24	8/16*	Data	40							
UCS512XX*- 500kHz	12-24	8/16*	Data	80	40	20					
UCS512XX*- 750kHz	12-24	8/16*	Data	100	60	40	30	25	20		
UCS512XX*- 1 MHz	12-24	8/16*	Data	100	75	50	40	30	25	20	
UCS5603	12-24	12	Data/Backup	100	50	35	25	20			
UCS7604 (+RGB)*	24	16*	Data/Backup	100	75	50	40	30	25	20	
UCS7804*	24	16*	Data/Backup	100	75	50	40	30	25	20	
UCS8903	24	16	Data	70	40	25	20				
UCS8904	24	16	Data	70	40	25	20				
UCS9812	24	16	Data	75	40	25	20				
WES9412	5	8	Data	100	75	50	35	30	25	20	
WES943*	5	8	Data/Backup	100	50	35	25	20			
WS2801	5	8	Clock/Data	100	100	100	100	70	60	50	45
WS2803	5	8	Clock/Data	100	100	100	100	70	60	50	45
WS2808	12	8	Data/Backup	100	75	50	35	30	25	20	
WS2811 (400Khz)	5	8	Data	70	40	25	20				

LED IC	Volt	Bits	Protocol	1	2	3	4	5	6	7	8
WS2811 (800Khz)	5	8	Data	100	75	50	40	30	25	20	
WS2812	5	8	Data	100	75	50	40	30	25	20	
WS2812B	5	8	Data	100	75	50	40	30	25	20	
WS2812S	5	8	Data	100	60	40	30	25	20		
WS2813	5	8	Data/Backup	100	75	50	40	30	25	20	
WS2814	5-12	8	Data/Backup	100	75	50	35	30	25	20	
WS2815	12	8	Data/Backup	100	75	50	40	30	25	20	
WS2818	5-24	8	Data/Backup	100	50	35	25	20			
WS2851	5-12	8	Data	100	75	50	35	30	25	20	
WS2852	5-12	8	Data	100	75	50	35	30	25	20	

ASTERICS MEANING

DMX TTL variants* = DMX TTL signal can be sent out in four speeds: 250 kHz (default for DMX fixtures, 500 kHz, 750 kHz and 1 MHz). It can be different for the fixture, if it actually supports the higher speeds. Check the datasheet of the manufacturer to check which speeds are supported.

UCS7604/UCS7804/SM16724 = This IC requires 8 bit data but in the IC itself, it will be upscaled so it will have the benefits of a 16 bit LED IC.

UCS512XX = This IC covers the whole UCS512 family: UCS512B3, UCS512B4, UCS512C1, UCS512CN, UCS512C4, UCS512D, UCS512H4, UCS512G4. Covers the 3ch and 4ch. The UCS512B3, UCS512B4, and UCS512C4 are 8 bit. The rest have built-in upscaling to 16 bit. The speed that can be archived can vary from fixture to fixture. 750 KHz is 95% supported while most ones also support 1 Mhz.

UCS512XX (8 bit) - UCS512B3, UCS512B4, UCS512C4

UCS512XX (16 bit upscaling) - UCS512C1, UCS512CN, UCS512D, UCS512H4, UCS512H5, UCS512G4, UCS512G6

WES943 = This variant uses 2 data wires per line.

Please note that in case of 16 bits we calculate based on 16 bit color channels. This means that you don't need do half the number of universes listed above, you can calculate with these numbers.

Copyright

Disclaimer

No legal rights can be derived from this document. Errors and omissions are excepted.

The information in this document is subject to change at any time without prior notice. Visit our website www.digidot.eu for the most recent version of this document. The information in this document was put together with great care and is intended to serve as a helpful guideline.

DiGidot Technologies B.V. explicitly rejects any form of liability claims in advance, whether it concerns direct or indirect disadvantages, damage, injuries, loss of income or other financial, materialistic or personal consequences that are directly or indirectly caused by use of this product or it's owner or user. Any form of unintended or misuse of this information and/or DiGidot products indemnifies DiGidot Technologies BV from liability and warranty obligations.

Feedback

Tell us all about your experience with DiGidot!

The continuous development of the DiGidot control platform is only possible thanks to feedback from our users. If you have any suggestions, please contact us by email: info@digidot.eu

Imprint

DiGidot Technologies BV
Bromostraat 18-3
1067 TR Amsterdam
The Netherlands

Chamber of commerce registration: 67637825
VAT identification no.: NL857103581B01
Managing director: Stefan Verhoef

Phone: +31 (0)20 820 1849
info@digidot.eu
www.digidot.eu