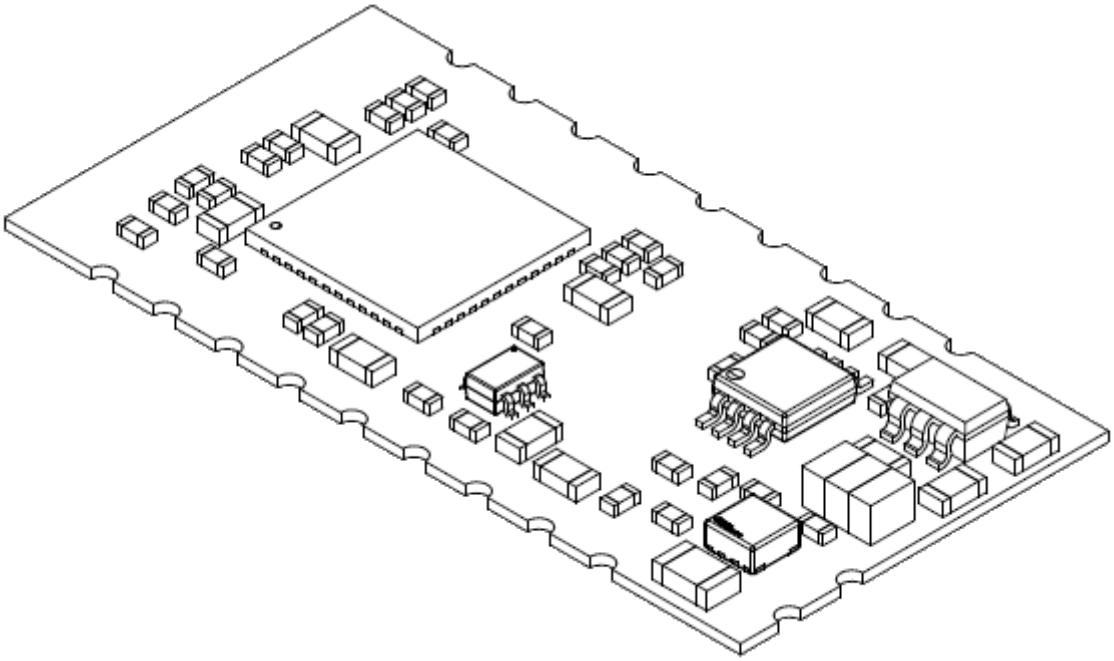
DiGidot

PUSHING CONTROL BOUNDARIES

PxLNet Smart Receiver



5-24 VDC | 8-16 BIT | ARM | DMX RDM | SPI | CE

THANK YOU FOR CHOOSING DIGIDOT!

We strive to offer you the latest technology in LED lighting control and we're constantly innovating and expanding our control platform. We're convinced that you have everything you need to create a truly mesmerizing and innovative piece of light art.

Are you proud of your end result? Please share it with us, we love to see what you can do with our products!

Your DiGidot team

/Digidot-Technologies-b.v.

/DigidotTechnologies

TABLE OF CONTENTS

INTRODUCTION

The DiGidot PxLNet Smart Receiver brings light fixtures to a whole new level. The day has come that you don't have to think about which protocol you are going to use. You can send any signal, PxLNet, DMX or one of many different SPI protocols to the Smart Receiver and it will work. The device also works with the RDM standard, allowing you to remotely configure the device, when everything is connected. And with the auto-address feature, every fixture knows it's position in the chain so minimal configuration is required.

The Smart Receiver enables manufacturers to fully specify what features they want to use and how. You can add both analog and digital sensors to measure voltage, current, temperature, and much more. The built-in power limiter can automatically dim the fixture if the temperature is too high. When there Is no signal from the main controller, it can turn off the power supply after a timeout.

The Smart Receiver also contains a powerful pixel processing engine which can duplicate pixels, group, remove/add colors, apply gamma correction and much more! This can save a lot of output channels from your main controller.

All these settings can be configured from a web portal. Manufacturers have the freedom to select which parameters can be adjusted by the end-user, with RDM, and which one are locked. You can create Product Profiles, is similar how DMX personalities work, but much more advanced.

BEFORE INSTALLATION

Before installing DiGidot products it is important to consider following safety and installation instructions.

Safety instructions

- Before installation and use of this product, read this manual carefully.
- When handling this product, ensure you're fully discharged and take counter-ESD measures, like an anti-static bracelet, mat and chair and make sure they're properly grounded.
- Local electrical and safety rules and guidelines always take precedence over this manual.
- DiGidot Technologies B.V. cannot be held liable for improper handling, product installation, usage or storage.
- Installation should only be carried out by a professional and certified installer that is qualified to work on the electric installation.
- Do not conduct any repairs of the device (there are no user serviceable parts inside). Any unapproved repairs and/or product modifications will void product warranty. DiGidot Technologies B.V. cannot be held liable for any consequences.
- Repairs of this product may only be carried out by the manufacturer DiGidot Technologies B.V.
- Repairs and maintenance on the installation may only be carried out by qualified technicians.
- Always disconnect the mains power when working on a high voltage electric installation, not doing so may result in product damage or personal injuries.
- Do not connect or modify this product other than described in this manual.
- Never use a product that is damaged or does not work correctly or when the product is visibly damaged or when the product starts to smoke, or when a crackling/sizzling noise is audible. If this is the case in any way, turn of the power of the devices and please contact your supplier immediately.
- The Smart Receiver is designed for indoor use (dry locations) only. Exposure to rain or moisture may cause fatal damage to the controller and increase risk of electrical shock.

Where to start reading?

We, as DiGidot know that this manual isn't just a regular manual. With more than 14.000 words, it really needs a small guide where to start reading. And this section will help you to navigate through the manual!

1. **Explore**

This section, consists of 2 chapters: Feature Set Overview and Check Application. This is a great point to get started as it goes through every functionality of the Smart Receiver, first in shortlist form, and then more in-dept. And it should give you an idea about how you can utilize this device. When you think this might be suitable for your fixture or project, you can read the Check Application chapter. This will cover some fundamental requirements to your fixture/project and make sure it's going to work for this case.

2. **Integration**

It must be said that unless you're 100% sure that the Smart Receiver will be useful in your project/fixture, it's probably a good idea to start with the **Explore** section. After that, this section will cover 3 chapters: Integrate into a fixture/project, Use and Apply Product Profiles and Configuring a Product Profiles. The first chapter will get your project/fixture ready with all the possibilities of the Smart Receiver: How can you connect your LED's, sensors or buttons, how do you define your topology, basically, how you can utilize every feature and what additional hardware you should add to make it work. After that, in the chapter Use and Apply Product Profiles, an introduction is made to the online web portal, where you can manage your Product Profiles. This determines how the Smart Receiver should operate. It also explains how to change the Product Profile in the future. Finally, the chapter Configuring a Product Profile covers every setting and possibility out there. And every parameter is explained what it does and how you can use it. This is shown when you're creating or editing a Product Profile.

3. **Using in the field**

When you have your Product Profile and additional hardware ready, we can now talk about how the Smart Receiver should be handled in the field. This section covers 3 chapters: Connect to the Smart-Receiver, Tips & Tricks, and Troubleshooting. First, we explain how the device can communicate to DMX / RDM / SPI controllers and how the available functionality is different when you use third-party controllers. An installation should never be considered complete without checking the Tips & Tricks section to make full advantage of the Smart Receiver in your fixture/project. Finally, problems could always be on the rise so a chapter, Troubleshooting, is fully dedicated to solving all kinds of problems: from not finding any device, to fixing low performance issues.

Appendixes

Finally, this manual has some appendixes which shows things like the Pin-out of the Smart Receiver, Supported RDM parameters and Supported LED IC's.

Last word

This manual is an All-in-One version. Only the specification sheet is a separate document. Because of it, a lot of references are made to different sections. That's the reason you should use this manual digitally: Every piece of text that is Orange with underline, is clickable. Only this piece is an exclusion!

Enjoy reading and good luck with your Project or Fixture integration with the Smart Receiver!

Feature set overview

Short list

In- /Output

- **Flexible** Input Support: Choose from 30 different SPI protocols and DMX
- **Flexible** dual output support: Send DMX (with high-speed options), SPI and PxLNet over long-range
- **Long range** Input by design
- Works with **any third-party DMX/RDM** controller
- Up to 6 universes of Input
- Up to 2x 4 universes of Output

Sensors

- Support for **2 analog** sensors
- Support for **digital sensors** that use the I²C protocol

Pixel Manipulation

- Built-In **upscaling** from 8 to 16 bit
- Pixel/channels **grouping**
- Covert RGB to RGBW, vice versa, add or remove channels with the flexible color order
- Show a **fallback color** when there is no signal received
- External **Status LED** support

Automation

- Automatic adjust output power based on current temperature
- Automatic addressing
- Automatic power supply control when there is no signal received, to turn off

For the fixture manufacturer

- Fully configurable form our web portal for manufacturers
- Parameters are fully **RDM compatible**
- Determine which parameters are **user configurable**

Extra

- Over-the-Air Updates

Detailed feature list

Universal Long-Range Input

Universal PxLNet/DMX input Connection: Supports DMX input directly or PxLNet through our accessory, the PxLNet Transmitter, from a DiGidot or another SPI controller.

Process more than 3000 channels, output more than 4000

The input handles up to 6 universes and the output can do 4 universes each so in total, the Smart Receiver can send up to 8 universes.

Dual outputs that can send various protocols

The Dual Output enables you to send up 30 different SPI protocols, along with DMX. Both outputs can independently configured to do their own thing. So, both DMX, both SPI or a mix between them. It also supports sending higher speeds of DMX to increase performance on the output.

Configure with RDM controllers

You can either use the built-in RDM functionality on a DiGidot controller, or any other RDM capable device to configure the Smart Receiver, SET/GET parameters and read sensor data.

Powerful Pixel Manipulation engine

Our built-in pixel processor can do various things to the input data. The data can be upscaled from 8 to 16 bit so will double the performance. The grouping of pixels is also a powerful feature as this won't reduce the channels of your license from any SPI controller. Further, you can freely set the color order to match the LED IC In the fixture. It's also possible to bypass, mask or add zero channels so you can for example, convert a RGB signal to RGBW and vice versa.

Integration of sensors

The Smart Receiver is able to communicate to external sensors, like temperature, voltage, current, basically anything that the RDM standard describes. There are 2 inputs for analog sensors where you can connect any sensor that works on changing the resistance or voltage. It also features an I²C Interface to communicate to digital sensors.

Automatic addressing of devices

When you're only using Smart Receiver enabled fixtures, you can automatically address them by using the auto-order functionality. This makes the patching process a lot easier and faster.

Automatic power supply switching

Either when there is no signal received on the input, or it only shows 'blackout frames', the Smart Receiver can automatically turn off a power supply. If the connection is restored, the power supply will be turned on again.

Adaptive power limiter

With the adaptive power limiter, the Smart Receiver can dynamically adjust the maximum brightness of the fixture in relation with the temperature. If the fixture gets hotter, the maximum brightness will decrease or even completely turn off the fixture to prevent any damage.

Fully customizable functionality with our web Interface

Manufacturers are in full control when it comes to all previously mentioned functionality. Our online web portal enables them to create Product Profiles to fully customize the inner working of the Smart Receiver.

Compatible with DiGidot and third-party RDM controllers

All RDM parameters can be SET or GET by any DMX/RDM controller. The DiGidot controllers can also communicate with the Smart Receiver, if a SPI protocol is used.

Operation logging

The Smart Receiver features a built-in logging system that keeps track of the power-on time, burning time, and the number of times the device has been switched on. This information can also be retrieved with RDM.

Over the air updates

Regular firmware updates will be available in the future to add even more functionality, improving the existing feature set and to fix any bugs.

Small in size

This whole feature set comes all packed in a very tiny device which is only 30 x 15 x 2 mm (length, width, height) so it’s easy to integrate this into a fixture.

Check application

Requirements / checks

Because the Smart Receiver is a component and not an end-user product, it's important to know which requirements the project or product has so you can determine if this product can be used as intended.

What type of LED IC do you want to control?
The most important question is: What type of LED IC do you want to control? The Smart Receiver is built to output various digital LED IC's, like DMX and SPI variants. The full list of all supported LED IC's for the output, is shown in the appendix Supported LED IC's. The device has 2 outputs which can be configured individually so outputting different protocols is possible. SPI can be used directly. For DMX outputs, keep in mind an additional IC or a PxLNet Transceiver is required.

What main SPI/DMX controller do you want to use
Second question would be: what kind of main SPI/DMX controller are you're planning to use with? The Smart Receiver is a DMX/SPI-processor so an external input is always required. You can either use our DiGidot line-up, or any other third-party DMX/SPI controllers. The full feature set is always available if you're using our controller. With third-party controllers, take a look at the following chapter to see what the possibilities are: "Compatible modes with Third-party controllers" as functionality may be limited.

How many channels do you want to control in total
The device supports both 8 and 16 it output. With 16-bit LED IC's, it can send out 2 universes of data. With a 8-bit LED IC, it can send 4 universes of data per output. In total, you can connect 32 Smart Receivers on one single data output from a DMX/RDM or DiGidot controller.

How many channels do you want to send from every data line form the SPI/DMX controller?
Up to 6 universes of data (8-bit) can be send to a network of Smart Receivers. Keep in mind that you can also group/repeat channels on the output so this may reduce the number of channels required from the main SPI/DMX controller.

Space and power required
The device it's maximum power it can use is 1.5 Watt. And the device itself is 30 x 15 x 2 mm (L x W x H) in terms of dimensions.

Make up the results

When these questions are asked and no problem arises, then using of the Smart Receiver should be a valuable addition for your fixture or project! But if you want to make sure the specific feature(s) that you require will be handled as expected, continue to read on the next page, which goes into every functionality in detail. If you're still have questions or need further assistance, we are happy to help!

Integrate into a fixture or a project

Introduction

This chapter provides comprehensive guidance for manufacturers or customers looking to successfully integrate a PxLNet Smart Receiver into their fixture or project. It also gives information about the power rating, LED IC connection and how to connect various inputs and outputs like sensors, buttons, auto addressing, PSU switch and much more.

This chapter is divided into 2 sections: **Essential** and **Optional** connections. The essential connections should always be done, regardless of which setup the Smart Receiver is used in. The optional connections are extra things you can add to extend its functionality.

Essential connections

First, we will go through the connections that are mandatory if you're going to use the device:

Power

To connect power to the Smart Receiver, attach a power supply with a voltage range of 5 to 48 Volts DC. The maximum power consumption is 1.5 Watts. The V+ (red) should be connected to the positive terminal and the V- (black) should be connected to the negative terminal of the power supply.

Main data input connection

The main port for input (and output in case of RDM traffic) is called **D1**. For the data, 3 pins should be connected: Pin #12 (D+) and Pin #13 (D-) and Pin #14 to the cable it's shield (GND).

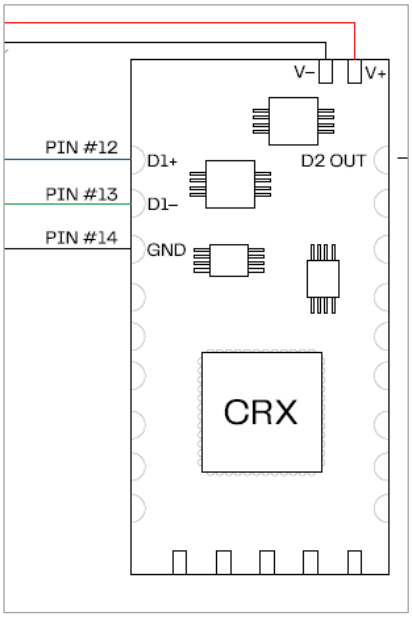


Figure 1 - CRX power & main data port connection

> Note: the wiring of the data connection follows the same rules as the DMX standard.

Topology

The Smart Receiver can be integrated in different kind of network topologies. These are the 2 variants to choose from:

- Daisy chain
- Bus style

The difference is how multiple of Smart Receivers are connected together. To better understand how they differ we will go through both of them.

> Note: This chapter will focus on data connection diagrams. Remember that power connections to the fixture/LED tape and the Smart Receiver must also be established.

Daisy chain style

In daisy-chain style, all the fixtures/lights will be connected to each other in a chain, very similar to how DMX fixtures are connected: You will have a data input and an output. Both sides of the fixture will have 5 connections: D+, D-, DC+, DC- and data ground. If you plan to use the "Auto-addressing" feature, an additional pin will be required.

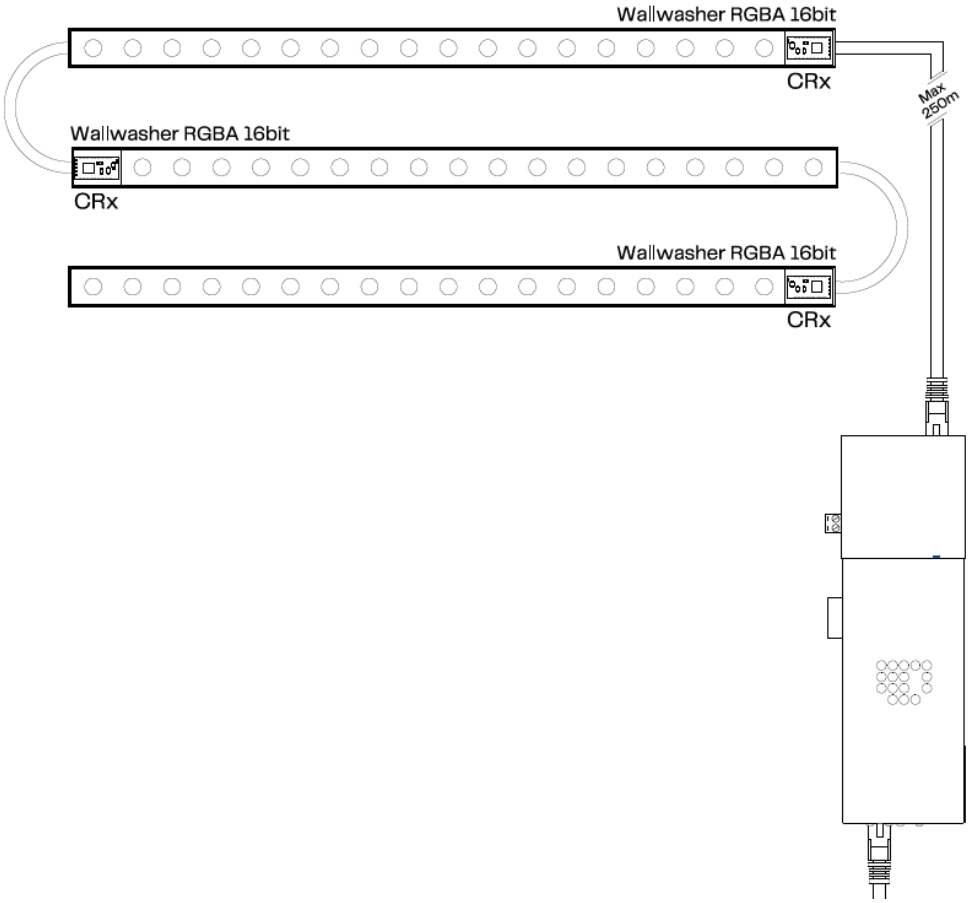


Figure 2 - Daisy-chain topology

Bus style

When you want to use the bus style, generally, the Smart Receiver, is not integrated into the fixture itself, but in a small breakout board to utilize both outputs. This data signal has a tree structure: one tribe with many branches: The main data line goes in one direction, and it has multiple T-split junctions, where the Smart Receiver is connected. This approach is very useful as it completely lacks the daisy chain part: Every branch has a Smart Receiver with two available outputs, and at the end of the line, it doesn't have to be connected to the start of the next line. This increases reliability because if one Smart Receiver, or line fails, only that line will fail but everything on a different branch or data output will still work. And it can save a lot of cables.

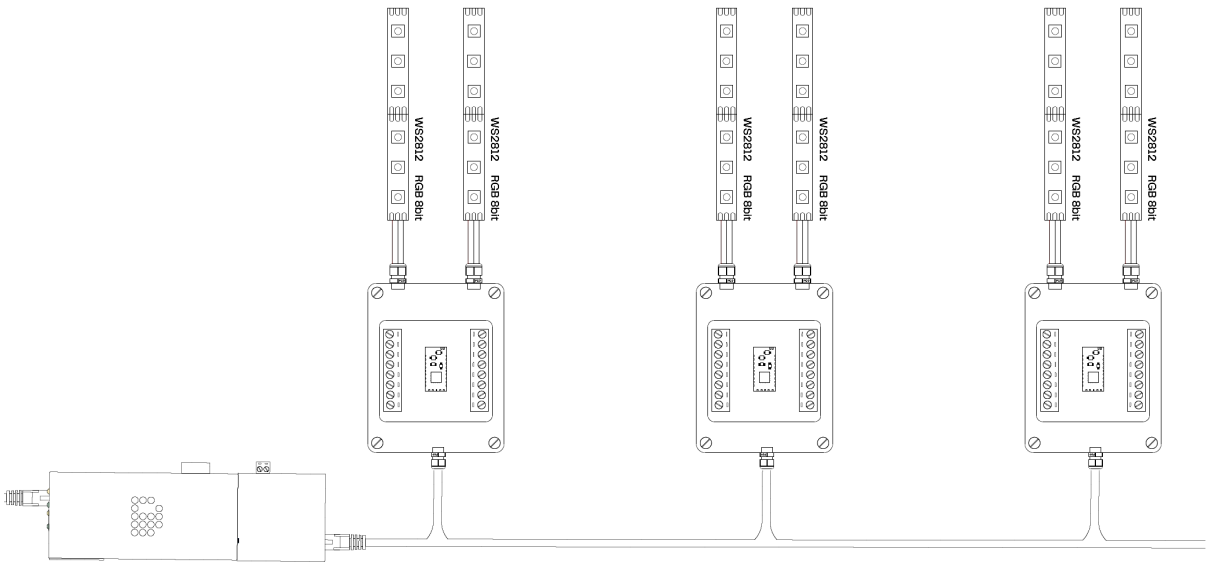


Figure 3 - Breakout Topology

Limitations

Regardless of which topology you're going to use: keep in mind it follows these guidelines:

- **Maximum Receivers per output:** No more than **32 Smart Receivers** connected on one single data output
- **Maximum Distance:** The maximum distance from the controller to **any Smart Receiver**, should be less than **250 meter** as it doesn't act as a repeater here.

Connect output(s) to the LED IC

The Smart Receiver has two available outputs. They are called **D2** and **D3**. Both outputs can be configured independently so both DMX-TTL, both SPI or one can output DMX-TTL and the other one SPI. Be sure to check the list of Supported LED IC's if the Smart Receiver supports the LED IC. If you're only want to use a single output, use **D2**.

> Note: When you're planning to use SPI, you can connect the LED IC it's data input, often referred as D or DI, directly to one of the output pins D2 or D3. If you're planning to use DMX however, you will need to add an additional IC (RS485 driver) to convert to a proper DMX.

Adding a DMX output

If you want to output a proper DMX signal, you will need additional hardware to make it work. The IC that you want to search for is a **RS485 Transceiver IC.**

Generally, any RS485 Transceiver IC will do. Its data input should be connected to the Smart Receiver its output (D2 or D3), the A and B signals, should be connected to the DMX IC and the IC should be in **Drive/Send** mode all the time. The capable speed/bandwidth should be at least 0.5 Mbps. When you're DMX IC supports higher speeds of DMX (like DMX1024 or DMX2048), we recommend 5 Mbps to be supported.

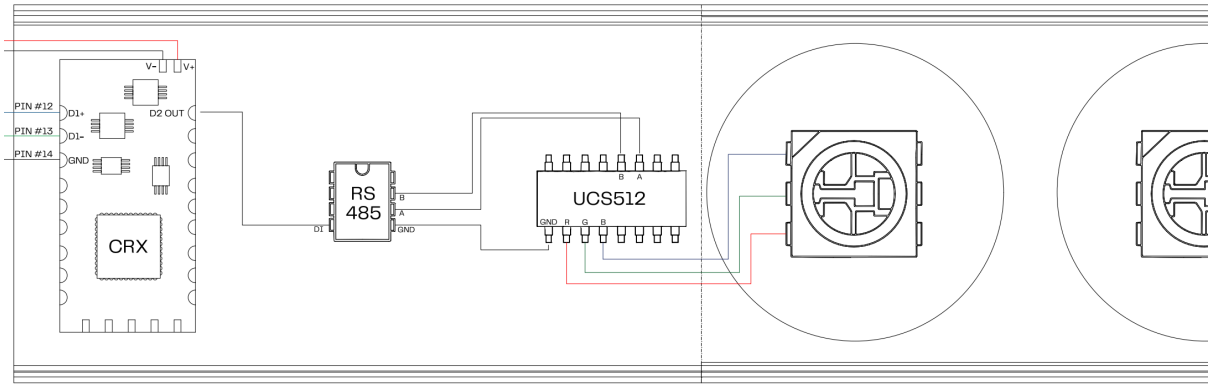


Figure 4 - Smart Receiver integrate in DMX fixture

Adding a SPI output

To connect the Smart Receiver to an existing SPI LED IC, the data input of the SPI LED IC, for example the UCS1904, can be connected directly to **D2** or **D3** out. The signal coming from the Smart Receiver is between 0 and 5 volt DC.

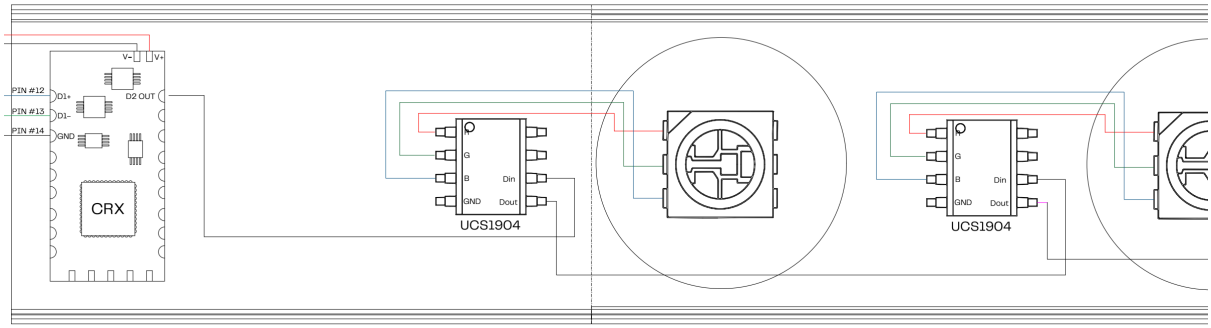


Figure 5 - Smart Receiver integrate in SPI fixture

Extra connections

Adding analog sensors

The Smart Receiver has two analog inputs: **Pin #2** and **Pin #3**, labelled **Sense 1** and **Sense 2**, which accepts any voltage range between 0 and 3.3 volt. That means that any device that can output a constant voltage, will work. You can use sensors that will vary their resistance, based on what it measures, so called “Resistive Sensors”. For example, you can use NTC or PTC resistors that will change its resistance with temperature. It’s also possible to connect analog 0–10 volt devices but with some additional hardware, to avoid damaging the Smart Receiver.

> Warning: To make resistive sensors work, they need a resistor divider network which contains an additional resistor as well as an external voltage. Make sure the value on the input pin doesn’t exceed 3.3 volt under any circumstances.

Example: For connecting a 0-10 volt device, you can use two resistors: a 2.2 kΩ and 1 kΩ ohm that are connected in series and where the mid-point is connected to the analog input (top image)

Example 2: To add a potentiometer that has a range of 0-10 kΩ, you can add an external 5 volt power supply and a 10k resistor to connect it to the analog input (bottom image)

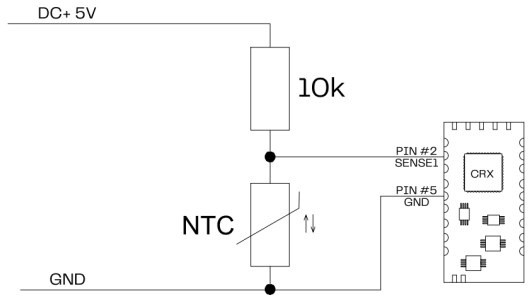


Figure 6 - NTC connection scheme

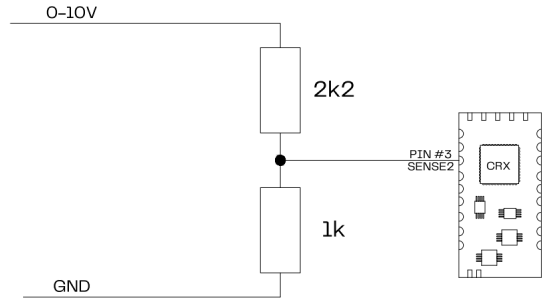


Figure 7 - 0-10 volt connection scheme

For analog sensors: when you define this sensor on the Product Profile page, the value being read on the input will be between 0 and 4095. Now, 0 means 0 volt is measured, and 4095 means 3.3 volt. You can do two different things with the raw value that's measured: offset and multiply. The offset parameter will add or subtract the value while the multiply value, as the name suggest, multiplies the value.

> Note: For every 1 volt increase on the input, the raw value will increase with 1240

> Note: Try to aim for "milli" values for your sensors: millivolt, milliamps etc. to have a decent resolution and precision. Comma values can't be displayed.

Example 1: Show the raw voltage level at the input
The offset parameter can be 0 (0 as raw value = 0 volt) and the multiply should be 0.813 because 1240 x 0.813 = 1000mV (for the multiplier, 1000 means multiply by 1.000, so you will need to fill in 813). When 3.3 volt is measured on the input pin, its raw value is 4095. This, multiplied with 0.813 makes 3300 which represents the actual voltage in millivolt.

Example 2: Current sensor with linear voltage response
Given is a current sensor, that can measure current. The output of this sensor is a voltage level. When measuring zero current, the output of the sensor is 1.5 volt and for every 1 amperage, the output level is increased with 100 millivolts.

First the offset value: At 1.5 volt, it should show 0 current. For every voltage, the raw value increase 1240. This means: 1.5 Volt x 1240 = 1860. This should be 0 amperage so we set the offset parameter to **-1860** (the minus will subtract the value from its raw value) so this will result in 0.

Then the multiplier: For every amperage, 100 millivolts will be added to the sensor its output. Again, for every volt increase, the raw value is increased with 1240. It's best to get to the milli representation because comma values are not allowed. So again, for every 100 millivolts, it's increased by 1240, divided by 10 (as 100mV fits 10 times in 1 volt) = 124. This number should be 1000 as for every 100mV, the current rises 1000mA. So, the multiplier will be: 1000 / 124 = 8,064 so this will result in the final value of **8064** for the multiplier.

Example calculation Question
The raw input is showing 3090. How much amps is going through?

Answer
3090 + -1860 (**offset value**, filled in with the minus) = 1230
1230 x 8,064 (**multiplier value**, 8064 is the actual value for the multiplier) = 9918 milliamps or 9.9 Amps.

These sensor definitions can be filled in on the Product Profile configuration page.

Adding digital (I²C) sensors

In addition to analog sensors, the Smart Receiver also supports digital sensors using the I²C protocol. Note that using I²C sensors will **disable** the PSU switches on the Smart Receiver.

To use I²C devices, make sure the necessary pull-up resistors are there on the circuit and they have a unique address. Connect the **SDA** to Pin #17 and **SCL** to Pin #18.

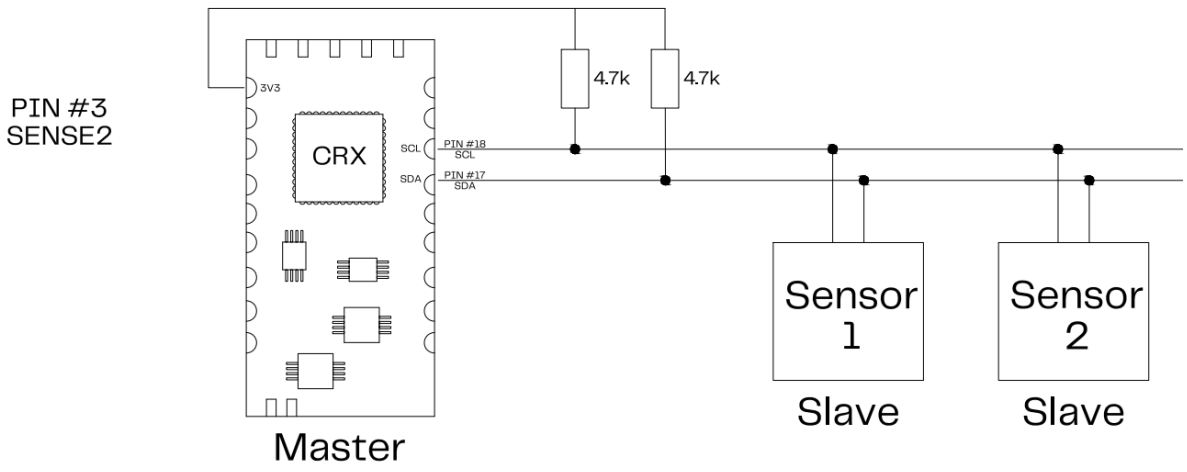


Figure 8 - I²C connection scheme

Currently, they're 2 devices supported:

- SHT40 (temperature)
- INA237 (temperature / current / voltage / power consumption)

The sensors can be configured on the Product Profile configuration page at the Sensor section.

In the future, more sensors can be added with a firmware update. If a specific sensor is required for your project, you can always contact us and we will check what we can do and if we're able to add it.

Auto addressing

The Smart Receiver contains an automatic addressing feature which can be used to know the order of the fixtures on a data line. For this, two pins are required from the device. The “Program-OUT” pin must be connected to the next device it’s “Program-IN” pin. The first device is determined by having the “Program-IN” pin left unconnected.

Wiring of the pins

Two pins are required from Smart Receiver to make this work. When all the connections are made from one side (Breakout topology), both Program-IN and Program-OUT function, will add a wire so the total count or wires will be 7. If your topology has an in- / output (daisy-chain topology), the connectors on both sides, should have 6 pins. The output connection should go to the next fixture it’s input connection every time a fixture is added.

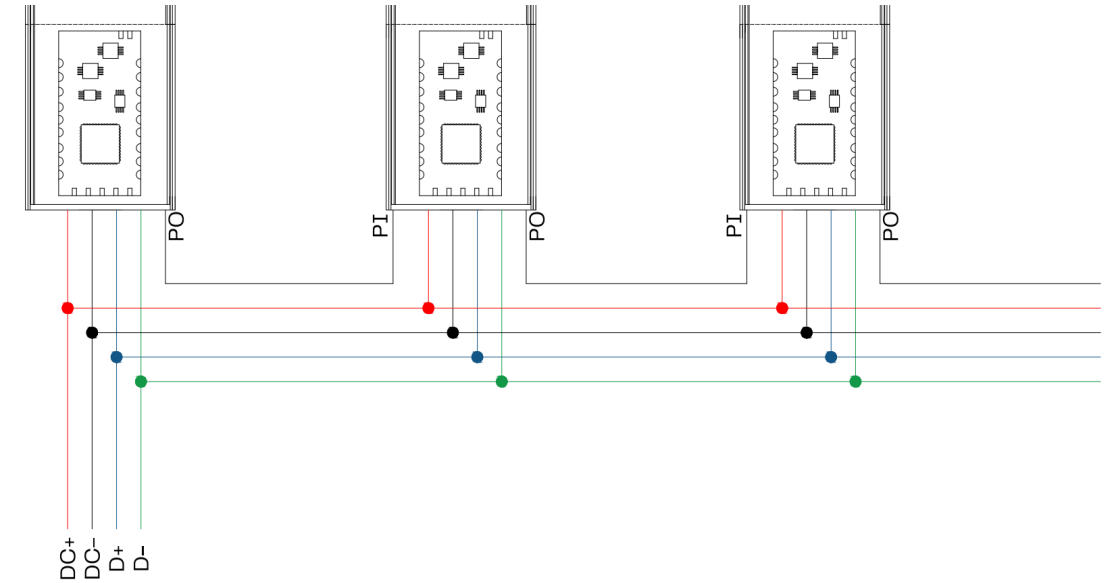


Figure 9 - Auto-addressing connections

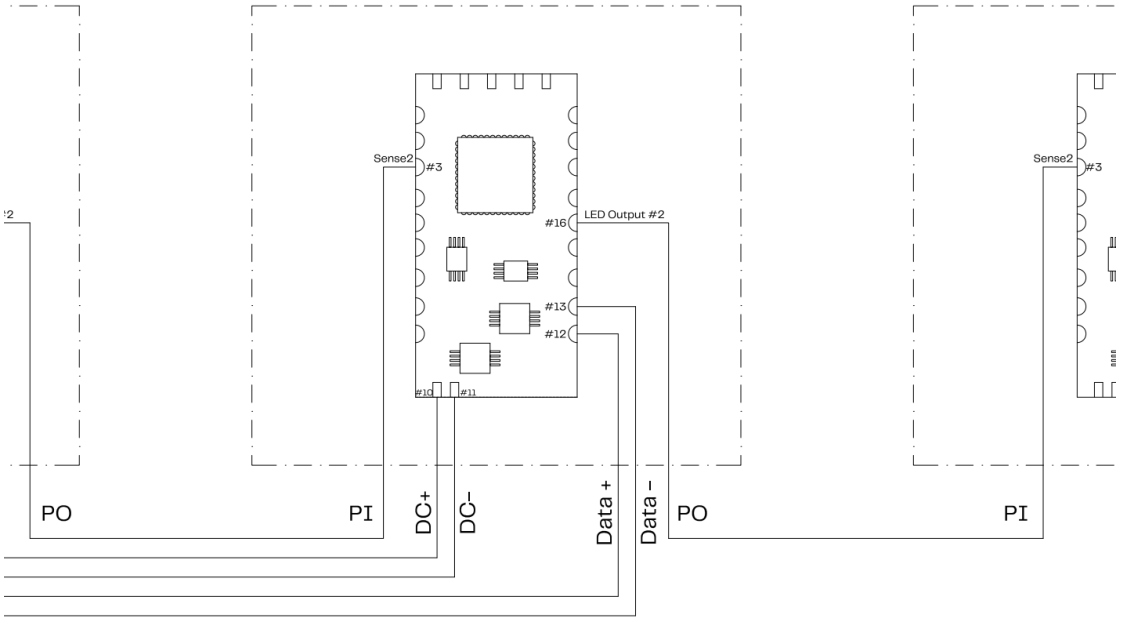


Figure 10 - Close-up auto addressing connection. Note: the GND connection is not shown in this image

Adding in the Product Profile Configurator
You can use the automatic addressing feature by configuring the “Program-IN” and “Program-OUT” pins. The input can only be configured on **Sense 2 (Pin #3),** which can be configured from the AUX Connections (analog). The output can be only be on **LED output 2 (Pin #16)** which can be configured from the AUX connections (digital).

PSU switch

The Smart Receiver features an automatically switched GPIO, when there is no signal, or the incoming signal is “Blackout” for a period of time. After the timeout, the power supply is turned off and when the signal resumes, it's turned on again. This can save power as standby power consumption of the power supply is eliminated.

> Note: When using the PSU switch feature, the I²C bus will be unavailable

> Note: For this feature to use, the power supply of the actual LED's or fixture, must be separate to the power supply of the Smart Receiver so the device will stay on all the time.

The Power Supply Switch can be added in 3 steps

1. Go to AUX connections (digital), and choose for **PSU SW 1 (Pin #17)** or **PSU SW 2 (Pin #18)**
2. Scroll down to the “Power Supply Switch Config” and choose on which pin number it's configured, how the GPIO works (see "Adding GPIO" to get more info) and, if the pin should be HIGH by default
3. Finally, you can set 2 timeouts: one for loss of data signal, the other for the time the input is fully "Blackout". Both timeouts are in milliseconds.

Adaptive power limiter

The Adaptive Power Limiter is a feature to link the temperature value, to the brightness level. By default, the internal temperature sensor is used, but you can change which sensor value should be used for measurement. There are 4 setpoints that you can configure to adjust the brightness accordingly to the temperature. For every setpoint that it reaches, a different temperature and brightness level can be set. The region between the setpoint is linear interpolated.

Example:
For this example, the following values are being used:

	Temperature	Brightness
Start point	70	90
Setpoint #1	80	75
Setpoint #2	90	25
Setpoint #3	100	0

The start point is set to 70°C with a brightness of 90%. This means, till 70°C, the fixture it's brightness will be 90%. When it reaches 80°C, the brightness will be 75%. When it reaches 90°C, the brightness is 25% and when it reaches 100°C, the brightness is 0% as the fixture is getting too hot so it should turn off right away. Between the different setpoints, the temperature is interpolated with the brightness linearly so there's always a smooth transition.

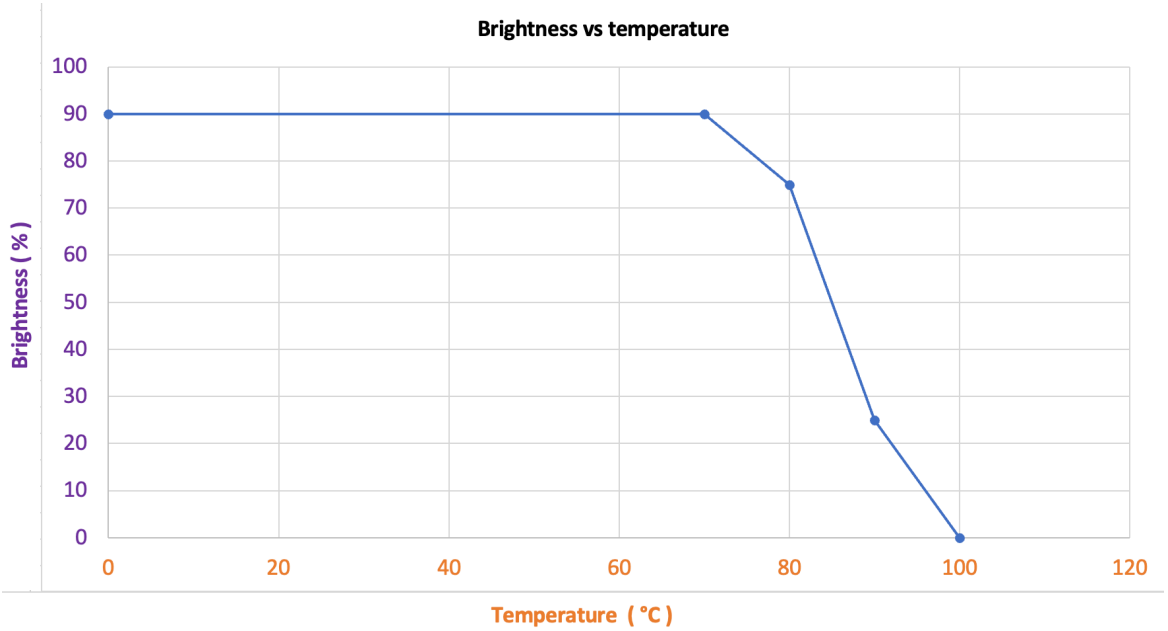


Figure 11 - Temperature vs Brightness response

Configuring
These parameters can be filled in with the Adaptive Power Limiter section.

Status LED

You can display the system status of the Smart Receiver by connecting two **WS2812** digital LEDs, in series, to one of output pins. You can either use a small piece of LED tape, or use the LED IC package like the **WS2812B**. For the last one, you will need to add some small components like a resistor and capacitor, to make it work properly. Refer to the datasheet to get the exact values.

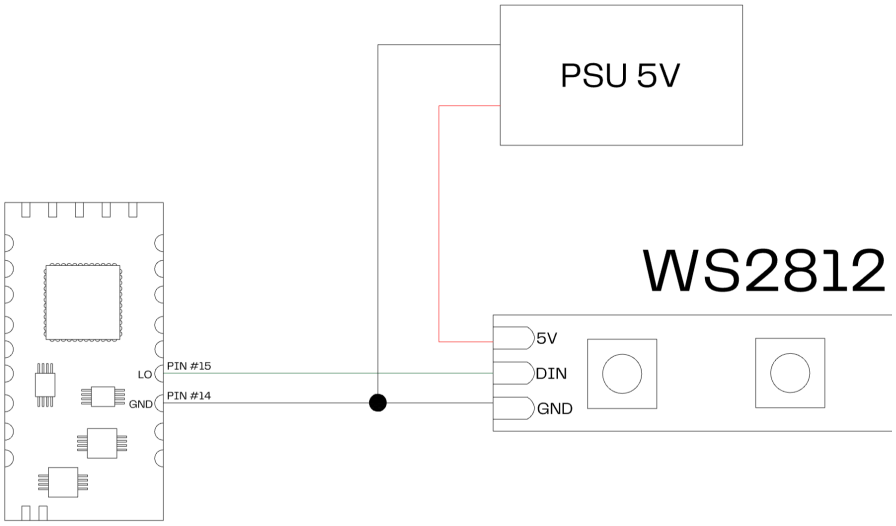


Figure 12 - Connect an external LED tape, to use as a status LED

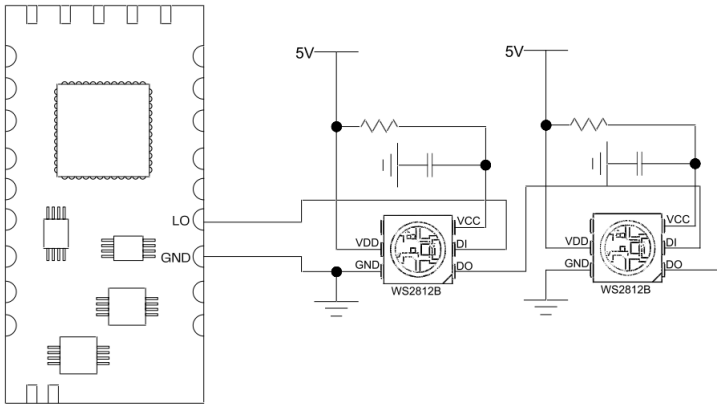


Figure 13 - Connect WS2812B IC's to use as a status LED

These are all the different status LED colors that are shown:

Colors	Status
Fading Blue / Cyan	Idle
Fading Blue / Purple	Playing a built-in effect/scene
Fading Yellow / Orange	Communicating with RDM
Flashing Red / Blue	Device has crashed, restarting
Fading green	Highlight feature in RDM to identify device

Configure the LED output

To make use of the external status LED feature, assign **LED output 1 (Pin #16)** to “Status LED Output” in the AUX connections (digital).

Adding GPIO

The Smart Receiver is able to interface to other devices with the help of GPIO’s (General Purpose Input Output). It can be used as an input or output. The digital voltage level of the input, or the value that need to be set on the output, can be SET/GET with RDM. For the output, you can choose between **Push-Pull** and **Open-Drain.**

With Push-Pull, you can output either 0 or 3.3 volt from the device. When you need a higher voltage (up to 5), you can use the Open-Drain method where you use an external pull up, connected to 5 volts, which is then shorted to ground.

The values set or retrieved via RDM correspond to the voltage levels on the pin: 0 for LOW (or 0 volt) and 1 for HIGH (or 3.3 volt).

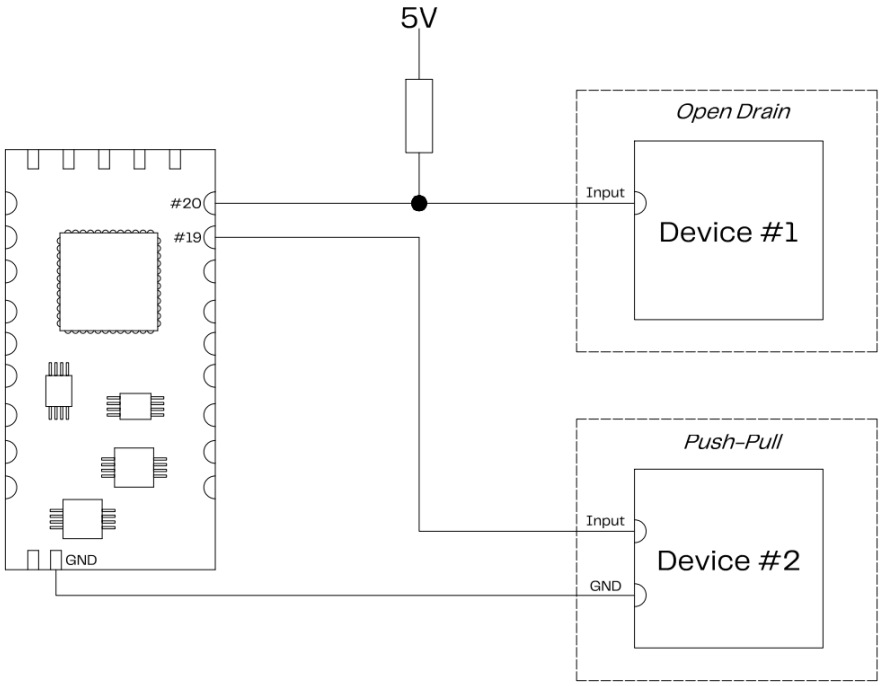


Figure 14 - Use GPIO's as an output, in Open-Drain and Push-Pull style

Configure the GPIO
To the GPIO functionality, select the GPIO function on four different pins, from the AUX connections (digital). section.

Add a button

It's also possible to connect a button or switch, to the Smart Receiver to, play some test scenes or factory reset the device. To get a full overview of the actions, checkout the Configuring a Product Profile chapter.

You can connect a button to **SENSE 1 (Pin #2)**. Make sure the configuration is setup properly in the AUX Connections (analog input) and button. The button is considered pressed when the input is 0 volt so a **Pull-up** operation mode is required which is shown in this image:

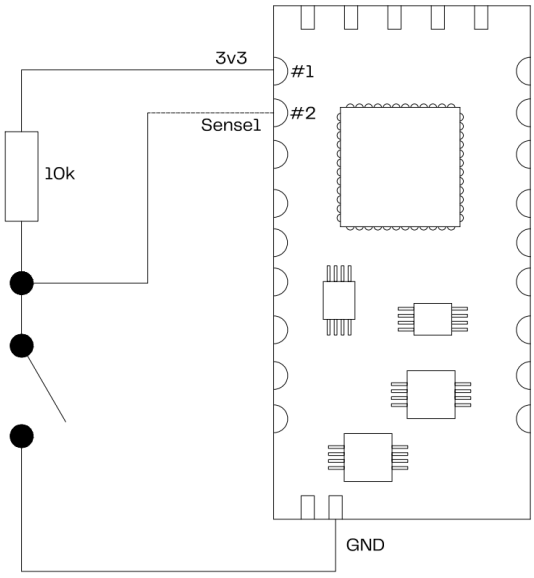


Figure 15 - Connect a button

PCB layout guidelines

Introduction

This chapter provides steps for mounting the Smart Receiver on a PCB using either SMD or through-hole method. The Smart Receiver is a high-speed communication device so design guidelines should be followed when producing a PCB with these speeds. An Altium Library is available on our website, if you want to use it as SMD or Trough-Hole on your PCB.

Required connections

Depending on the exact configuration and use-case for the device, not all pins have to be connected. The pins that always should be connected are the following:

- Power pins DC+ (Pin #10) and DC- (Pin #11)
- Main input connection: D1+ (Pin #12) and D1- (Pin #13)
- Output pints: D2 (Pin #6), D3 (Pin #9)
- Ground connections: GND: (Pin #5, Pin #8, Pin #14)

Design practices

- For D1+ and D1-, make sure the trace is the same length as well as the same impedance. Connect the cable its Shield Ground to #14. Follow the differential pair design practices for 5 Mhz.
- For D2 and D3, also route the Ground trace next to it (#5 or #8) to your LED IC so this track is equally as long as the main signal trace (#6 or #9)
- Pin #1 is directly connected to the main 3.3 volt rail. This can be used for very low current applications (<10 mA), like: pull-up circuit for I²C sensors or a small LED that indicates power good.

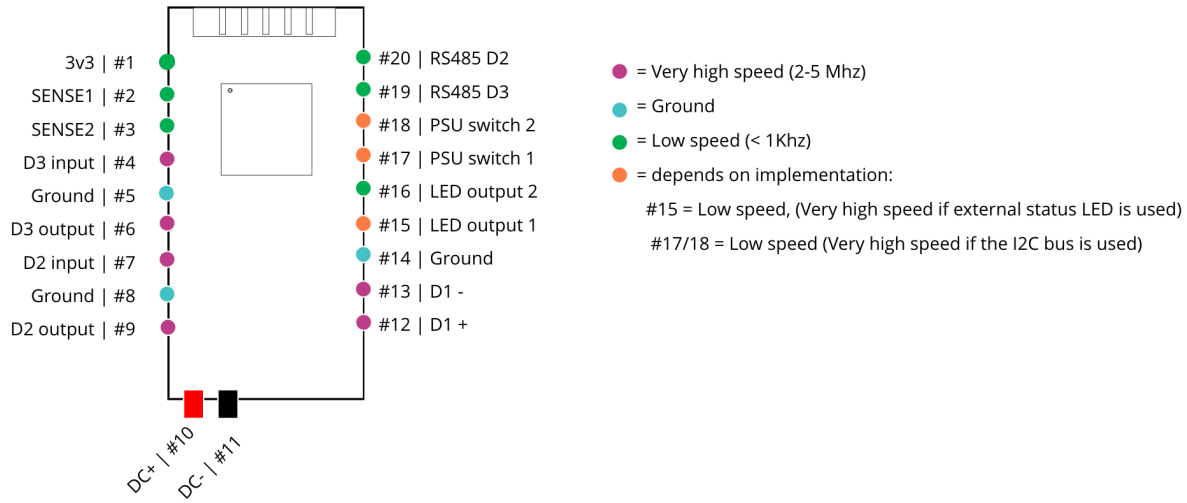


Figure 16 - Pinout speed requirements

Use and apply Product Profiles

General

This chapter describes how to access our online web portal to create identities for your Smart Receiver. The portal is available at: https://products.digidot.eu/ . This portal stores a list with fixture identities that you can manage. You can add, edit, remove the items at any time. You can also download these configurations as a file to upload to the Smart Receiver.

The identities are called **Product Profiles**. They describe how the fixture is configured and how it should operate. It contains information on how the different outputs and sensors are connected and how various functionality works, like the adaptive power limiter, PSU switch, grouping pixels feature, basically everything from the Feature set.

Some configuration parameters may be changed by the user, after the Smart Receiver has been delivered. You can define which parameters can be modified by the user in Locked Parameters.

All the possible options on the Product Profile page, are explained in the Configuring a Product Profile chapter.

Account

To use our web portal, you need to create a manufacturer account. This can be done by visiting our registration page:

https://products.digidot.eu/register-manufacturer/

Overview page

After creating an account, you're presented with the overview page. For every Product Profile you've created, 4 buttons will be shown:

- **Edit** - Takes you to the Edit Product Profile page
- **Duplicate** - Duplicate's the profile to a new one
- **Download** - Download the configuration file for the Smart Receiver
- **Delete** - Permanently delete the profile from your account

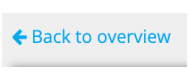
Adding a new item

If you want to add a new product profile, you can click the 

button on the top. This opens the editor page where you can configure all settings. See the chapter Configuring a Product Profile to walk through all the different options.

When you want to save this Product Profile as an item, you can go all the way down and click the 

button.

If you want to go back, go to the top of the page top, and click 

in the upper-left corner

Apply a new Product Profile to the Smart Receiver

Overview

To apply a new Product Profile to the Smart Receiver, two steps should be performed

- Download the Product Profile to your computer
- Upload this file trough the RDM interface of a DiGidot system

Update the Smart Receiver with a new Product Profile

To download the Product Profile configuration file (.crxt), click the **download** button at the overview page. After the file is downloaded, go to the interface of the DiGidot system and follow these steps

- Go to **Settings**
- Go to **Remote Device Manager**
- Press the **blue discovery button** to search for Smart Receivers
- Wait till the message "x RDM devices found" at the bottom of the screen or check the DiGidot Controller it's status LED is showing the idle status (blue/cyan) and not yellow
- On the top left, click the gear icon 
- Choose **Update Devices**
- Select the devices that you want to change the Product Profile of and press **Next Step**
- Click **Select File** and choose the file that you've just downloaded which has the ".crxt" extension
- Click **Upload**

After a few seconds, the update should be complete, with the message that shows how many devices were successfully upgraded. If any device failed to upgrade, you can try again or refer to the Troubleshooting chapter.

Configuring a Product Profile

Introduction

This chapter covers everything that you can configure, while editing or creating a new Product Profile, from our web portal.

General info

Product name

Name of the product. This is only used on the **Overview** page of the **Product Profiles.**

Product default RDM label

This will be the **default label** (RDM parameter: “DEVICE_LABEL”) of the device when it's discovered by any RDM controller.

Product Image

An image that represents the product. This image is only used on the **Overview** page of the **Product Profiles.**

Product version

Version label to be used only in the **Overview** page of the **Product Profiles.**

Properties

Physical properties

These fields, in **millimeters**, represent the dimensions of the product where the Smart Receiver is fitted in. These are read-only parameters.

Electrical properties

These fields, in **millivolt**, represent the operating voltage, **minimum** and **maximum**, the **current draw** and the **total power** consumption. These fields are read-only.

Additional product information

Overview

The additional product information is a list of parameters which you can add to show information about anything relevant to the product. There can be up to 16 parameters; with a description, and a value (number or text). These parameters are read-only.

Input connection

Pins

This will determine which pins represents the main input. At the moment, "Pin #12 + #13" is the only option to choose from.

Protocol

This determines what input protocol should be accepted. You have three options:

- **Auto** (default) - Accept both DMX and PxLNet
- **DMX** - Only DMX is accepted
- **PxLNet** - Only a SPI signal, combined with our PxLNet Transmitter is accepted

Regardless of this setting, the Smart Receiver will still be configurable at any time, with RDM.

Bits

This specifies how many bits need to be taken from the input. Normally set to 8 also if a 16-bit fixture is connected. If you want to have a 1:1 conversion (16 bit in to 16 bit out), you can set this to 16.

Out connection

About

This section allows configuration of the two LED outputs available on the Smart Receiver. The first item of the table is for output **D2**. When using only one of the two outputs, you want to use **D2**. If you scroll down, before **AUX Connections**, there is a button with **Add Output.** Click this one if you also want to use the second (**D3**) output.

Protocol

This is a list of available output protocols that can be send out.

Color order

The color order input field can be used to re-arrange channels from the input, to the ouput. The possible character you can enter are: 'r', 'g', 'b', 'w'. You can either have 3 or 4 different color values. All the letters should be comma separated. No comma is required at the very end of the value. You can also re-use the same letter if you want to duplicate the channel.

There are also 3 different symbols you can use to change the channels, for example, to convert RGB to RGBW or to use a RGBW signal to only control the Red LED of a fixture by skipping the G,B and W.

- **-** : Remove output channel
- **+** : Skip output channel
- **0** : Blank output channel

> Note: Keep in mind the total number of characters in this field (without the comma's) must be 3 or 4.

Example
The incoming signal has 8 channels, 2 pixels with RGBW data: **1**, **2**, **3**, **4**, **5**, **6**, **7**, **8**. And the output is also connected to a RGBW fixture unless other is specified in the Use Case

> Note: When channel 0 is shown on the output, it means it sends out a value of 0

Use case	Color order value	Output
Normal 1:1 conversion	**R,G,B,W**	**1**, **2**, **3**, **4**, **5**, **6**, **7**, **8**
RGBW to RGB (with RGB fixture)	**R,G,B,-**	**1**, **2**, **3**, **4**, **5**, **6**
RGB to RGBW	**R,G,B,0**	**1**, **2**, **3**, **0**, **5**, **6**, **7**, **0**
RGBW to RGBX (w channel skipped)	**R,G,B,+**	**1**, **2**, **3**, **0**, **4**, **5**, **6**, **0**,
Only use R (skip G,B,W)	**R,+,+,+**	**1**, **0**, **0**, **0**, **2**, **0**, **0**, **0**

Grouping channels

Take x amount of channels

The first parameter specifies how many channels need to be copied.

Duplicate x amount of channels

The second parameters specify how many copy operations there need to be done.

Example
You have a RGBW fixture within total of 10 pixels which can be individually controlled. You want to save input channels so you want to combine all those 10 pixels to do the same thing. Then **Take x number of channels** would be 4 (because it's RGBW) and **Duplicate x number of channels** would be 10 so the first pixel is copied on the remaining 9 pixels of the fixture.

Example

The incoming signal has 12 channels and represents data for 3 RGBW pixels: **1**, **2**, **3**, 4, **5**, **6**, **7**, 8, **9**, **10**, **11**, 12

Use case	Take x amount	Duplicate x amount	Output
Normal operation	**1**	**1**	**1**, **2**, **3**, 4, **5**, **6**, **7**, 8, **9**, **10**, **11**, 12
Duplicate channel	**1**	**4**	**1**, **1**, **1**, **1**, **2**, **2**, **2**, **2**, **3**, **3**, **3**, **3**, 4, 4, 4, 4
Group channels	**4**	**3**	**1**, **2**, **3**, 4, **1**, **2**, **3**, 4, **1**, **2**, **3**, 4

Fallback

For both outputs, you can configure a fallback color. When there is no signal, or the signal is “Blackout” for a certain amount of time that's specified in the fallback timeout field (in milliseconds) the color in the **Fallback colors** field will be shown. This field should have 3 or 4 values from 0 to 255 and separated by commas, just like the **Color order** field.

AUX Connections (analog input)

About

The Smart Receiver contains two analog inputs, on Pin #2 and Pin #3. In total, you can choose between three different functionalities:

Analog input	Connect an analog sensor with a voltage range from 0 to 3.3 volt
Button	Connect a button or contact enclosure to perform actions
Auto addressing input	Input for the Program in for Auto addressing
None	Input is disabled

You can enter a description for each input; however, this is only for the **Product profile** page. If you want to connect a sensor to one of these inputs, the description can be configured at the **Sensor configuration** section which is further down the page.

Not all pins can support all modes. This table shows the available modes per pin.

	Analog input	Button	Auto addressing input
Sense #1 (Pin #2)	YES	YES	No
Sense #2 (Pin #3)	YES	No	YES

AUX Connections (digital IO)

Overview

The Smart Receiver contains four digital IO that can be assigned with different functionality.

The available options are these:

GPIO	Use the pin as a simple in- or output.
PSU switch	Automatically switched output if there is no signal or black frames on the input
Status LED output	Connect 2x WS2812 (or similar) LEDs to show the status of the device
Auto addressing output	Output pin used to automatically address the devices
I²C interface	I²C bus to communicate with various I²C enabled devices

You can also enter a description for every digital IO, but that's only for the **Product profile** page.

GPIO

You can use any of the digital pins as an input or output.

For the output there are 2 ways of output types: **Push-pull** and **Open-Drain**.

The **Push-pull** acts like a voltage source: It either sends out 3.3 volt or 0 volt.
The **Open-drain** acts more like a switch. You can connect up to 5 volt, with the help of a current limiting resistor. Check the datasheet for the specific values. The state of the pin (HIGH/LOW) can be set with RDM. Besides that, you can also specify the state at startup. To get more information about the wiring, go to the integration part of the GPIO.

When configured as an input, you can get the value with RDM. When it's 1, its HIGH (3.3 volt) and if it's 0, its LOW (0 volt).

PSU Switch

With the PSU Switch option, the Smart Receiver will automatically switch the power supply on or off when the is no signal on the input or the input is "Blackout" for a given period of time. You can configure these parameters further down at the **Power Supply Switch Config**

Status LED Output

This option allows the Smart Receivers to drive 2 pixels of WS2812 LEDs to represents the device's status. The meaning of every color can be found in the Adding a Status LED section.

Auto addressing output

This pin should be connected to an **Auto Addressing input** of the next device to make the automatic ordering process work correctly.

I²C interface

In addition to analog sensors, it's also possible to connect digital sensors using the I²C protocol. Not all sensors are supported so you should check the support I²C sensor list. It's also worth mentioning this feature uses 2 pins (Pin #17 and Pin #18).

Available functions per pin

	Status LED out	Auto addressing output	I²C	PSU switch	GPIO
LED #1 - (Pin #15)	YES	No	No	No	YES
LED #2 - (Pin #16)	No	YES	No	No	YES
PSU SW #1 - (Pin #17)	No	No	YES*	YES	YES
PSU SW #2 - (Pin #18)	No	No		YES	YES

* = Uses 2 pins to operate

Sensors

Overview

You can add both analog and digital sensors to the Smart Receiver. In total, up to 15 sensors can be added. These sensors will appear in any RDM monitoring software and follow the RDM standard. Next to the **current value** of the sensor, the **min** and **max** will also be measured and are updated every second, for each sensor. You can also **reset** the min/max value.

Description

RDM parameter: The description of the sensor.

Type

RDM parameter: the type of the measured value

Min/max range

RDM parameter: Specify the range where the value is considered to be valid.

Min/max normal range

RDM parameter: Specify the range where the value should be between in a normal operation.

Multiplier

RDM parameter: A multiplication factor that's applied to the value before it's being sent out to the RDM controller.

Internal Value Multiplier & Offset

The **internal Value Multiplier** is very similar to the previous one except this is primarily for the analog input sensors to change the skew rate of the raw value when it's rising or falling. The **internal value Offset** is the 'zero' point for the sensor value which is primarily used with the analog input sensors. This value will be added or subtracted from the raw sensor value, before the multiplier is applied.

For more information on setting these values, refer to the Adding analog sensors section.

Connection

Specify the pin or protocol that the sensor uses which can be one of the 2 analog pins, or I²C.

Extra parameters (I²C)

When using I²C, you can specify the sensor that you've connected, the address and also which parameter you want to retrieve from the sensor.

> Note: When you're using an I²C sensor, make sure you've selected **I²C interface** at AUX connections (digital IO).

Power Supply Switch Config

Overview

When you selected PSU Switch on Pin #17 or #18, at the AUX Connections (digital IO) section, you can do the configuration in this section.

Pin

Select which pin, #17 or #18 you want to use this function.

Open Drain (not checked = push-pull)

Check the checkbox if you plan to use the **Open-drain** method. Leave unchecked to use the **Push-Pull** method.

GPIO set when switched on

Inverts the output behavior.

Turn off after no signal for x milliseconds

Specify a timeout for when there is no signal, the power supply should turn off.

Turn off Power Supply after black frames for more than x milliseconds

Specify a timeout for when the signal is present, but it's a ”Blackout”, the power supply should turn off.

Adaptive Power Limiter

Overview

With the **Adaptive Power Limiter,** you can dynamically change the output brightness based on a sensor value. You can choose either of the self-configured sensors or use the internal temperature sensor. Currently, this feature consists of a 3-stage brightness limiter:

If you **activate** the feature, you can select a **sensor index** to retrieve the temperature from. By default, it's a 0 which means the internal temperature is used.

'**Default behavior**' field. This determines the brightness (slider name = normal brightness) when the temperature is below the **start temperature (first threshold).**

Then, there are 3 different setpoints you can configure. The operation mode is best to explain with an example:

	Start point
Start temperature	**40**
Max Brightness	**100**

	Threshold 1	Threshold 2	Threshold 3
Activate temperature	**50**	**80**	**90**
Max brightness	**90**	**50**	**0**

This configuration can be summarized with this:

- Up to 40 degrees, the brightness is 100%
- From 40-50 degrees, the brightness will scale down to 90%
- From 50-80 degrees, the brightness will scale down to 50%
- From 80-90 degrees, the brightness will scale down to 0%

> Note: When the temperature reaches the final threshold of 90°C, the brightness will instantly drop to 0%.

Button config (if button at Analog in, Sense #1 is selected)

Overview

You can connect a button to Sense 1 (Pin #2) to perform a number of actions. The button is considered pressed when the signal is LOW.

Triggering

Because only one button can be connected, you can press and hold the button to perform up to 4 different actions. You can set a different action if the button is held for **0.2 sec**, **3 sec**, **10 sec** and **30 seconds**. While pressing the button, the device it's status LED will show a **white, fading effect** to show the next checkpoint of the timeout hasn't been reached yet. When it **flashes green**, you're in a valid time window to perform the action so if you release the button, the action will be executed.

Actions

Here is a list of the API commands you can use when a certain time has passed when the button is pressed. You can enter an API command here and then it will be executed.

test_playlist	Switches between 5 scenes: R,G,B,W, full white, fading red, and normal input
restart	Restarts the device
factory_reset	Erases all the user settings. Product Profile will be kept.

Internal sensors

Overview

Next to the sensors you can add yourself, and the internal temperature sensor, there are 4 additional sensors you can add to be monitored with RDM. These sensors are separate from the user-defined sensors and don't take any place of the 15 available spots of sensors, you can add yourself.

The sensors that are available are:

- **Uptime in seconds**
- **Input FPS**
- **Output FPS**
- **Invalid Frames**

The sensors can be added or removed with the help of a single number. Instructions for this are provided in the section Add/Change Internal Sensors. The visibility of these sensors can be changed by the user if it's not **locked**.

Lock parameter for end user

Overview

With these checkboxes, you can decide if a parameter is read-only or can be (re)set by the user with RDM.

Lock template

Overview

By locking the template, it's possible to determine what new template files the Smart Receiver should accept after the last one has been uploaded:

Anyone	Any template file from any manufacturer can be uploaded
Manufacturer only (default)	Only allow template files to be uploaded if it's from the same manufacturer
Finalize	After uploading the template file, it will be NO longer possible to change it!

Caution: If you use the **Finalize** option, you will be no longer be able to upload a new template file, even as a manufacturer. If it's really necessary to change the template afterwards, please contact DiGidot for further assistance.

Connect to the Smart Receiver(s)

Operating modes

Introduction

This chapter describes the different ways to connect to the Smart Receiver with a DiGidot system and a third-party controller. It also explains how to use the Smart Receiver can be used in normal, configuration, and update mode.

Mode's overview

There are 3 operating modes the Smart Receiver can be in: Normal, configuration and update mode.

In **normal mode**, the Smart Receiver receives data as a normal LED IC, like the WS2812 or DMX product.

In **configuration mode**, the Smart Receiver is compatible with RDM.

In **update mode**, it's possible to change the **Product Profile** of the device and to upgrade the firmware.

	Configured IC	Normal mode	Configuration mode	Update mode
DiGidot Controller	DMX / SPI	Supported	Supported	Supported
Third-party controller	DMX	Supported	Supported	Not Supported
Third-party controller	SPI	Supported	Not Supported	Not Supported

In summary: you have the full capabilities if you using a DiGidot Controller. When using a third-party controller with DMX/RDM, you will lose updating firmware functionality and changing its Product Profile. And with any other SPI controller, you can still use the Smart Receiver but you won't be able to set or get any parameters with RDM as well as updating.

Warning: It is highly recommended to use an SPI-configured IC when pairing the Smart Receiver with a DiGidot Controller for optimal performance.

Best practices for wiring

Wiring scheme

To establish a proper connection between the Smart Receivers and any DiGidot or Third-party control system, you need to connect three wires: **Data+** (D+) and **Data-** (D-) and the **Ground**. On the Smart Receiver, these are Pin #12, Pin #13 and Pin #14 respectively.

Cabling

To create a solid connection from the controller to the Smart Receiver, we advise to use a **RS485** or **DMX** cable. An ethernet cable can also be used. **UTP** is good enough for short lengths but for longer runs, a **F/UTP** cable is better as it has additional shielding around the cable and is harder to snap.

Other requirements

The limitations of a Smart Receiver network are comparable to a DMX system in terms of wiring and topology

- A maximum of **32 Smart Receivers** can be connected to one single data output
- Make sure to terminate the D+ and D- with **a 120 ohm resistor**, at the end of the line
- When using the Breakout topology, the stub length (from the T-split to the Breakout Box, we recommended to not exceed 1 meter

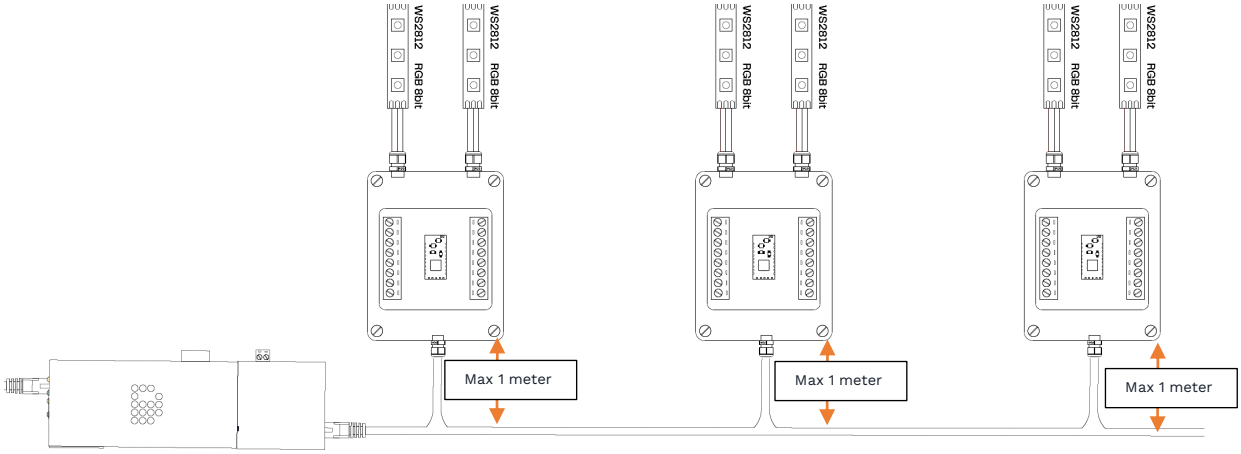


Figure 17 - Breakout topology with stub limits

Communication (DMX/RDM/SPI)

DiGidot C4 (with DMX)

> Warning: As DMX mode is quite slow, it's highly recommended to use SPI mode with a DiGidot C4, with the help of a Transmitter. But for small installations of testing, this is fine. Contact us if you're not sure

When using DMX mode, you can operate the Smart Receiver in any mode. To configure the DiGidot Controller, in DMX mode, you need the following:

- DiGidot C4 (Live or Extended)
- Smart Receiver

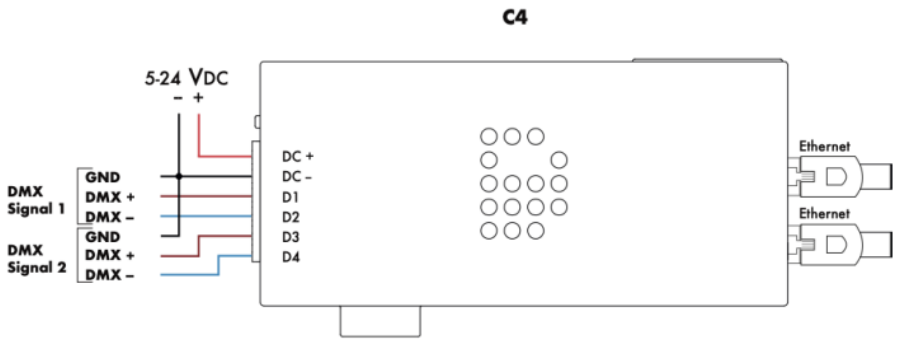


Figure 18 - DMX pinout from a DiGidot C4

To set up the DiGidot Controller, access the web interface, and configure a DMX universe on one or multiple ports. With the DiGidot C4, you can configure 2 DMX ports, with a PxLNet node, you will have 4 DMX ports.

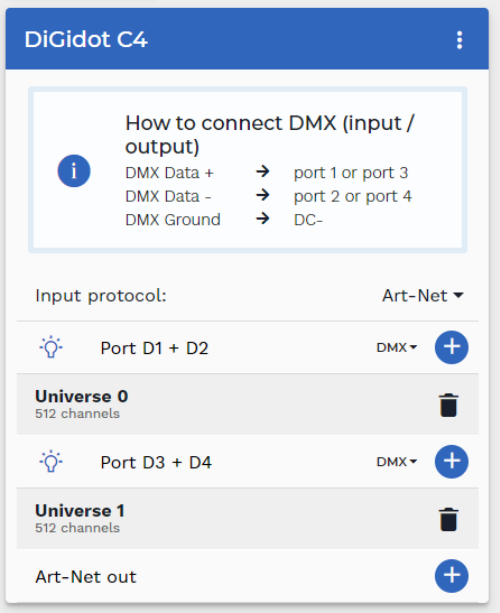


Figure 19 - Use with DMX configuration

DiGidot C4 + Transmitter (with SPI)

When using this method, you can operate the Smart Receiver in any mode. To configure the DiGidot Controller with SPI, you need the following materials:

- DiGidot C4 (Live or Extended) - or - DiGidot PxLNet Node
- DiGidot PxLNet Transmitter (only when using the DiGidot C4)
- Smart Receiver

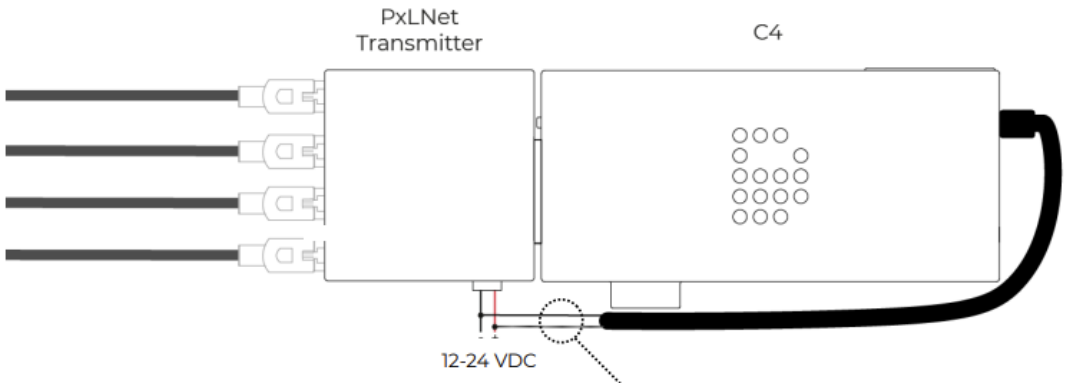


Figure 20 - DiGidot C4 usage with Transmitter

To setup the DiGidot Controller, access the web interface, and configure one universe or more (up to 6) with any supported LED IC listed in the Supported LED IC's table (**input** section) , on one or multiple ports.

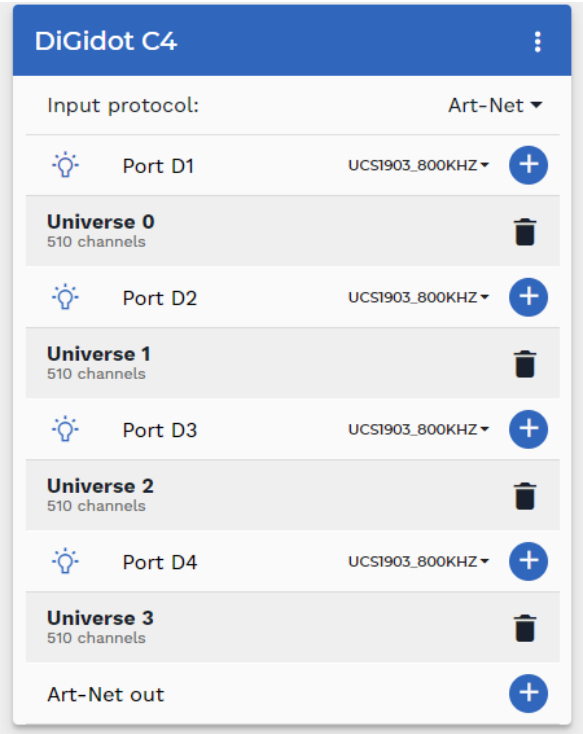


Figure 21 - Use with SPI configuration

Third-party DMX/RDM Controller (with DMX)

When using this method, you can use the Smart Receiver in normal and configuration mode. To configure the DiGidot Controller, in DMX mode, to communicate with a Smart Receiver system, you need the following materials:

- The Third-party DMX/RDM controller
- Smart Receiver

After wiring, ensure the controller is sending a full DMX universe and has RDM enabled on this universe if you want to use configuration mode.

Third-party SPI Controller (with SPI)

If you're using this method, you can only use the normal mode. To configure the Third-party controller, with SPI, to communicate with a Smart Receiver system, you need the following materials:

- The Third-party SPI controller
- DiGidot PxLNet Transceiver
- Smart Receiver

After wiring, set up one or more universes (up to 6) with any supported LED IC from the input section in this table. Then, connect the DiGidot Transceiver in **Transmit** mode. Refer to the PxLNet Transceiver manual on our website for instructions on how to do this. This is typically done by removing the jumper or changing the switch position.

Discover- and configuring with a DiGidot Controller

Introduction

If the wiring is completed, you can discover and configure your Smart Receiver(s). To do this, go to the web interface of the DiGidot System -> **Settings** -> **Remote Device Manager** and press the **blue button** with the refresh icon to start the discovery for devices. Depending on the number of devices, the discovery may take a while. Generally, it should take 10 seconds per device. If it takes longer, check out the Troubleshooting section and the tips & tricks page to increase the discovery speed.

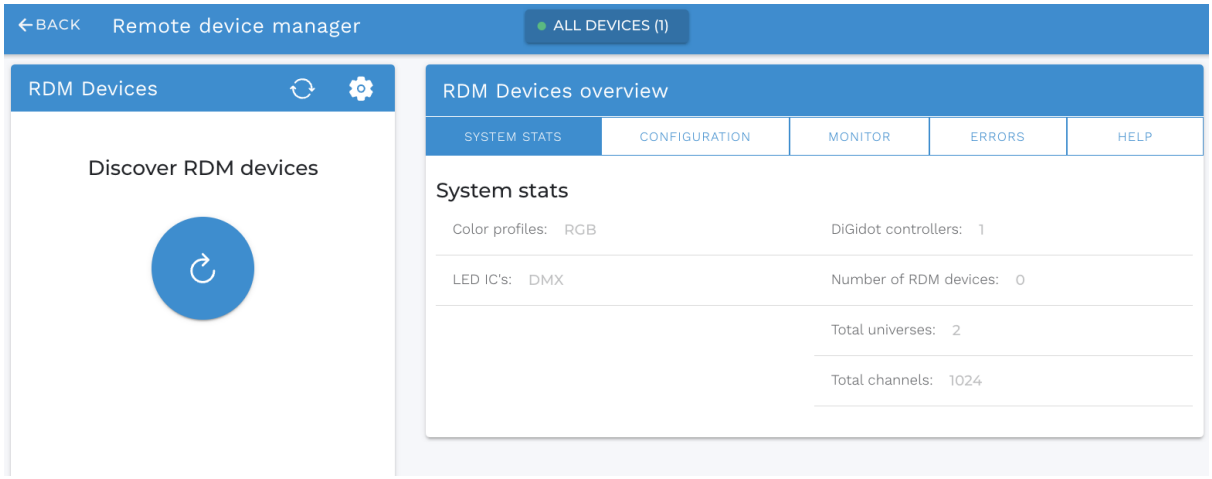


Figure 22 - Remote device manager page

Once discovery is complete, indicated by the status LED of the DiGidot Controller, changing from yellow to blue, the page will look something like this:

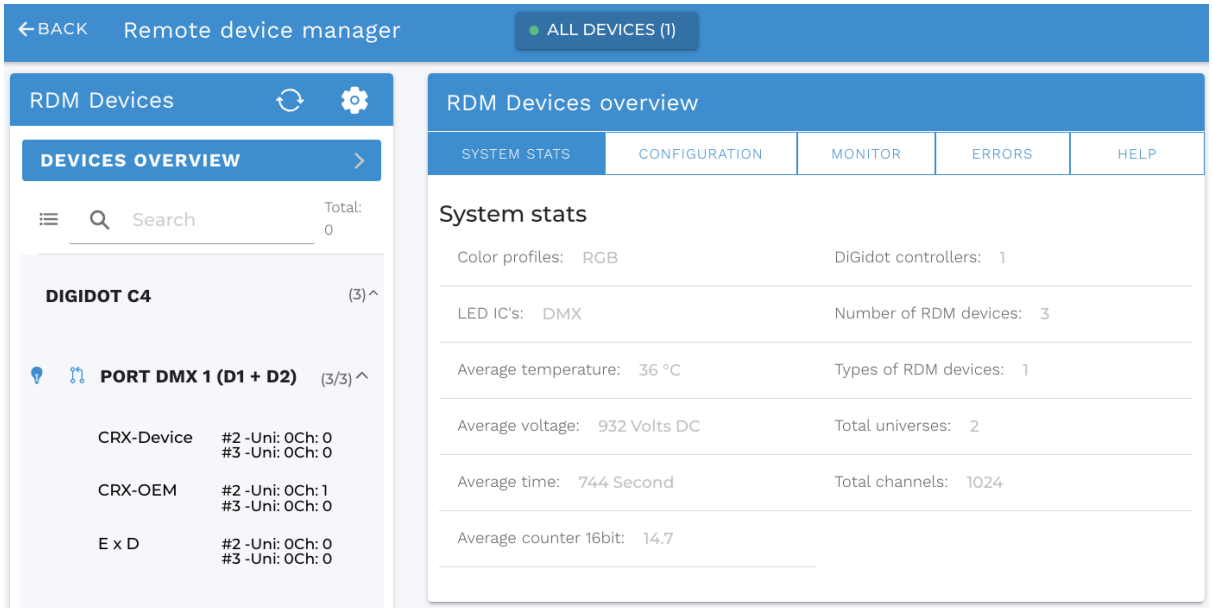


Figure 23 - RDM page after a discovery

In this case, one DiGidot C4 is used and three Smart Receivers are found on port D1+D2 while in DMX mode. When you click on the left side, on the name of the device, in this case, "DiGidot C4", the list will expand and shows on which port devices are found. After this, you can click on the port itself, in this case "PORT DMX 1 (D1+D2" to view the individual Smart Receivers. When you click on one of these devices, it will turn blue to indicate this is the current selected device.

When you've selected a Smart Receiver, the view on the right side will change so only information will be shown for that specific device.

This view on the right has 4 categories:

- **Information** – RDM parameters that are GET-only
- **Configuration** – RDM parameters can both be SET or GET
- **Sensors** – All the sensors configured for this Smart Receiver
- **Actions** – RDM parameters that can be SET only as well as some basic actions like highlight, factory-reset and restart

Information page

The first view to appear is the **Information** page. At the top, next to the device its name, there will be a lightbulb icon visible next to the device name. This can be used to highlight (RDM parameter: IDENTIFY_DEVICE) the device. For the Smart Receiver, this has 2 effects:

- All the LED outputs will be on 100%
- If an external status LED is connected, it will show a fading green effect for 5 seconds

Further down, the page displays parameters that can only be GET (read-only) but not SET. These parameters include information about the device itself, its ID, software version, and GET-only parameters from the Product Profile.

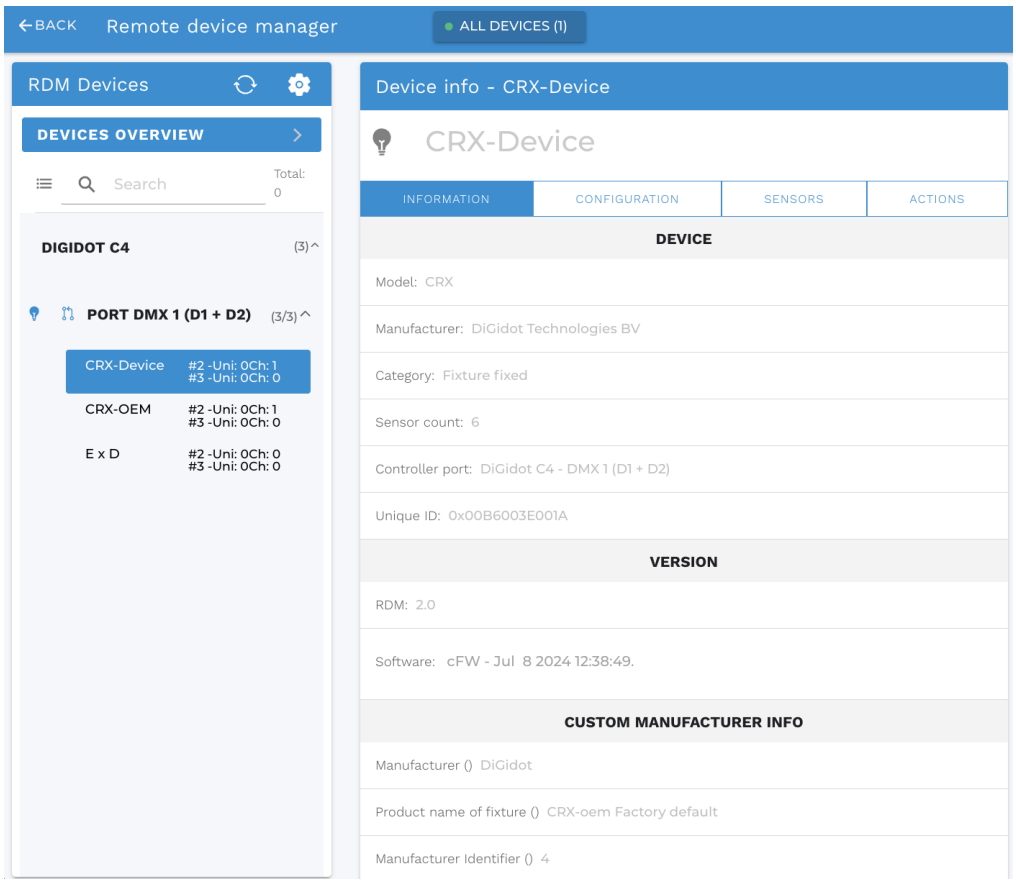


Figure 24 - Information view

Configuration page

This page displays all parameters that can be SET or GET using RDM. It contains most **Product Profile** it's parameters as well as the **user settings** like the start channel and the number of channels for each output. All values are retrieved from the device after a full discovery. When you want to refresh the value, you can press **GET**. To change a value, modify the number or text shown in the field and press **SET**.

If you want to get more information about each parameter, you can go to the RDM reference table over here.

You can also change the name (RDM parameter: DEVICE_LABEL) at the top here which is, in this case, "CRX-Device". Changes to the name are automatically saved in the background.

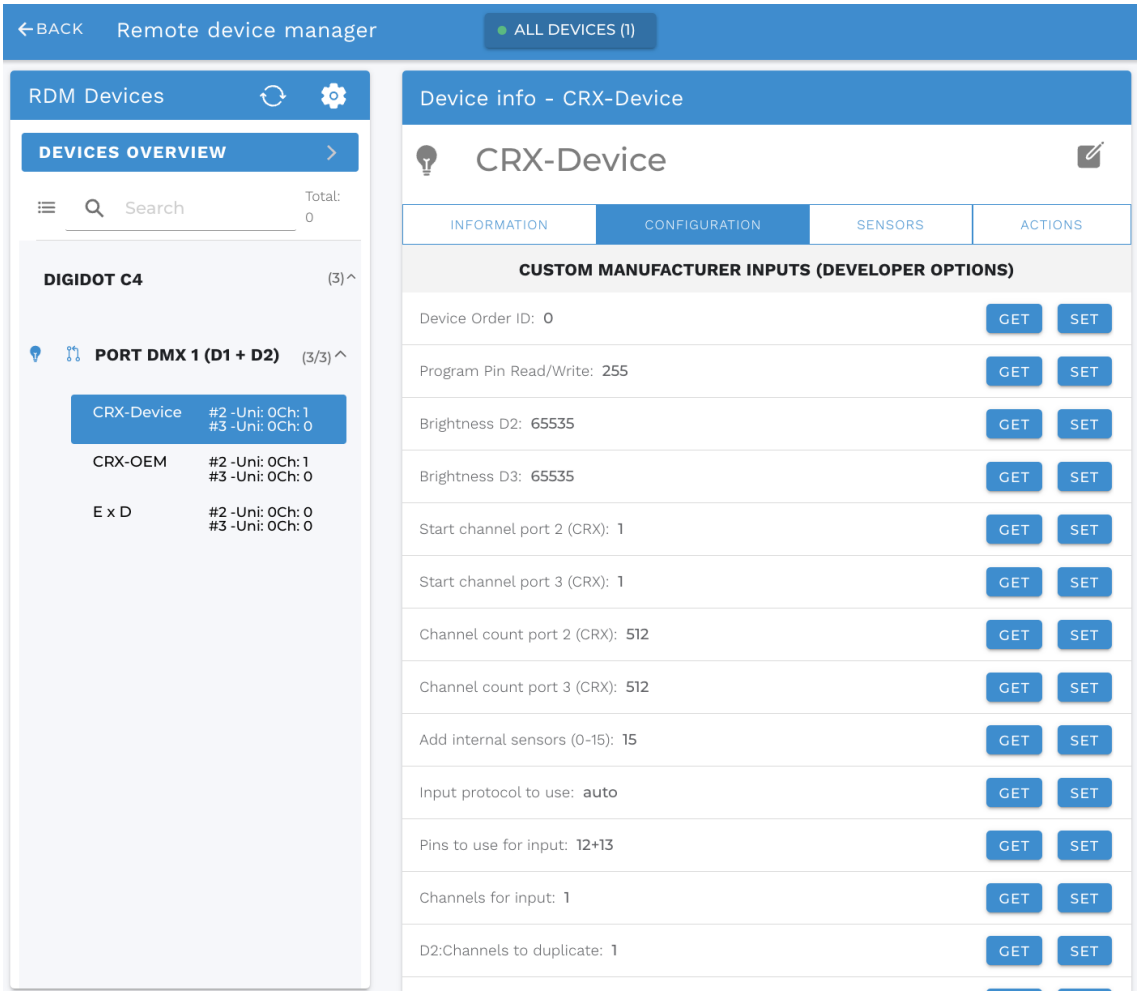


Figure 25 - Configuration view

Sensor page

This page shows all the configured sensors of the Smart Receiver. It also keeps track of the lowest and highest possible value ever recorded for each sensor. You can reset these lowest/highest registered values, by clicking the **RESET** button next to the specific sensor.

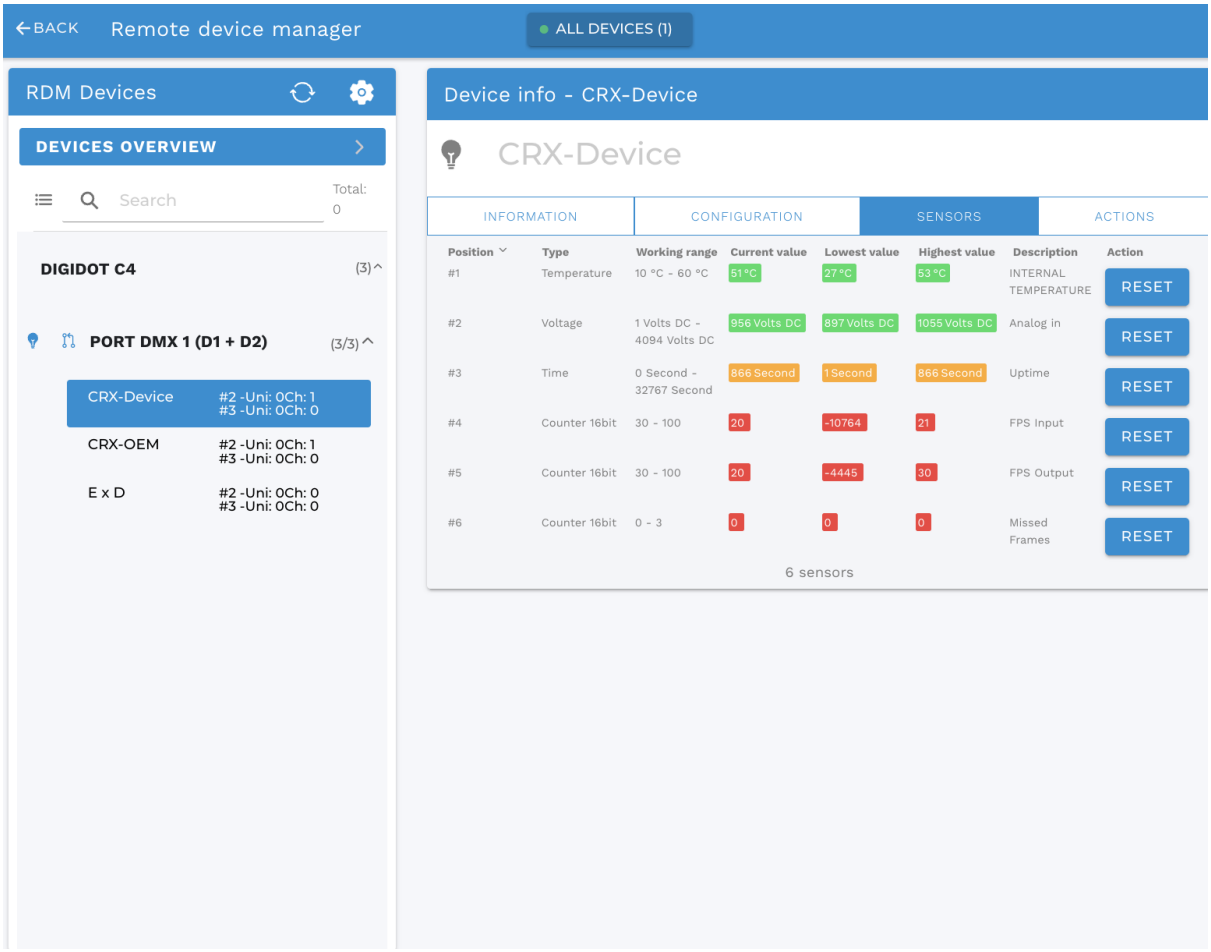


Figure 26 - Sensor view

Actions page

This page shows the RDM parameters which are SET-only. You can change the value in the text/number field and press **SET** to set the parameter on the device.

Additional options at this page:

- Restart the Smart receiver or the DiGidot controller
- Highlight specific device or the port of the DiGidot controller
- Factory-Reset (user settings **only**, Product Profile will be **kept**)

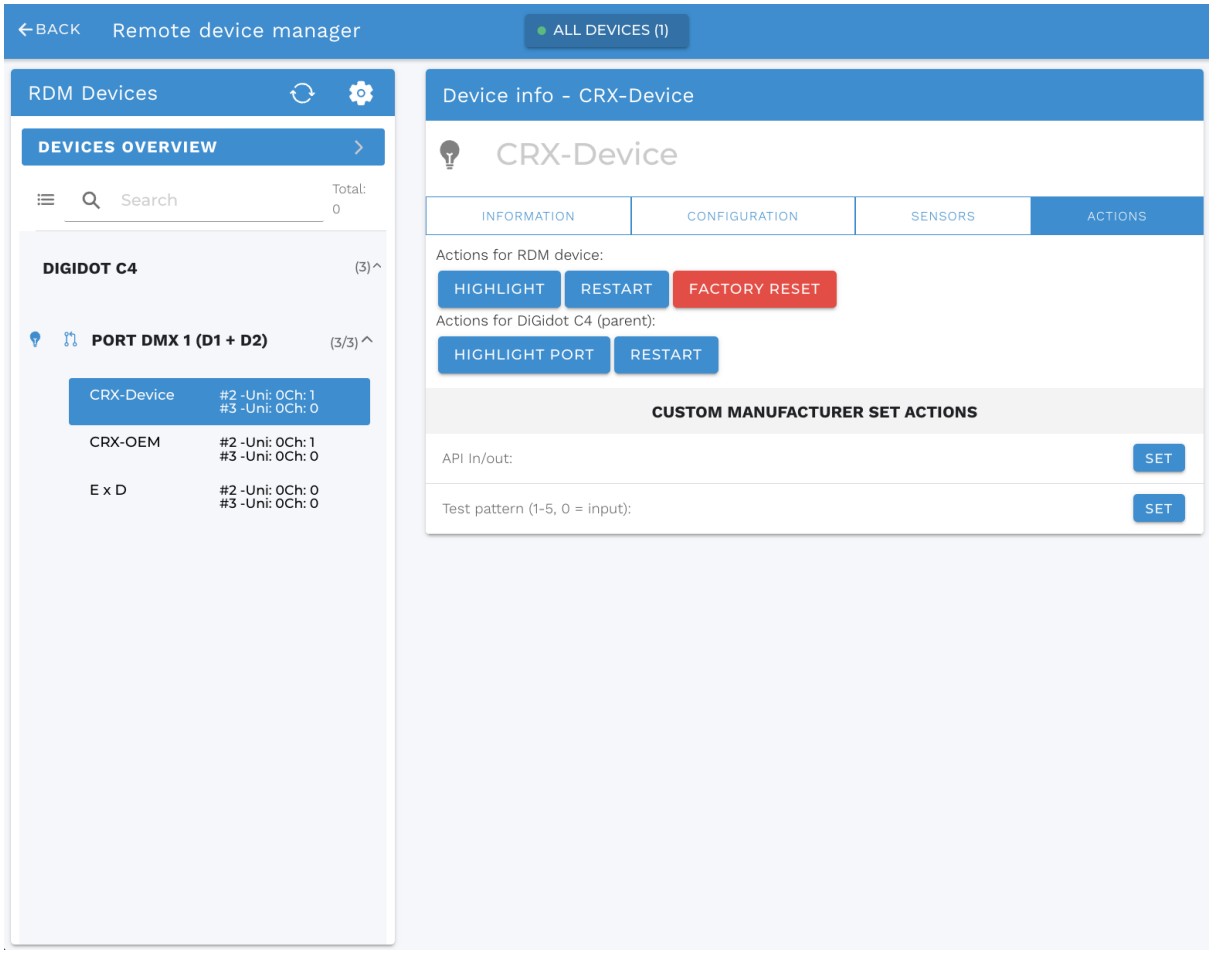


Figure 27 - Actions view

Tips & Tricks

Increase discovery speed

Tip: To speed up the discovery process, set the RDM parameter **Discovery Response Mode** (0x8F02) to **0**

Use the DiGidot to configure/update and after that, you can use any DMX /RDM /SPI Controller

Tip: When you want to use the Smart Receiver with other SPI controllers, you use the DiGidot controller first, to connect and fully configure the Smart Receiver and then connect it to the third-party SPI controller. If you want to change the Product Profile, while working with a third-party DMX/RDM controller, connect it to a DiGidot Controller to perform the update, and then put it back into your system.

Troubleshooting

Introduction

This chapter covers the most common problems if you're using a Smart Receiver. It shows a step-by-step approach on how to solve these issues.

No devices show up after discovering

If no Smart Receivers are detected by a DiGidot Controller or a third-party controller, consider the following possible issues:

- Either the SPI controller, the PxLNet Transmitter of the Smart Receiver **doesn't receive power**
- The cable **isn't connected properly** or makes a **bad** connection
- If a DiGidot system is used: The **wrong protocol** may be selected

Our advice would be to first start with the obvious and simplest mistakes that happen. After that, we go deeper if the issue is still there.

Check power to every device
Make sure both the DiGidot or third-party controller and Smart Receiver receives **power**. When using a DiGidot system, you can check if there is a **blue status LED** flashing on top or on the side. If you're also using a DiGidot PxLNet Transmitter, a **blinking blue LED** should be visible on the **side**. The best way to check if the Smart Receiver gets power, it is to get a multimeter and check voltages on the **3.3-volt power rail**. You can do this by connecting the Red (or +) probe to **Pin #1** and the Black (or -) probe to **Pin #11**. It should show 3.3 volt.

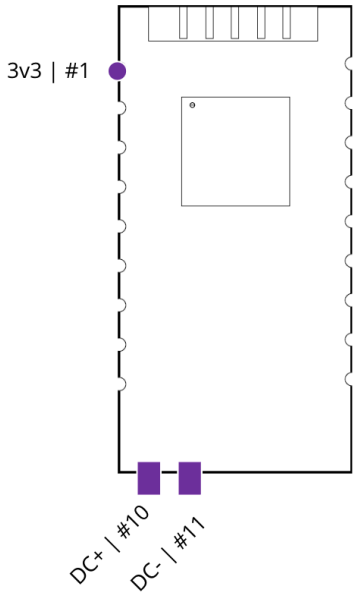


Figure 28 - Power pins

If it shows a number below 3.2 volt, leave the Black probe at **Pin #11** and move the Red probe to **Pin #10.** It should read anything between **5 - 48 volt** as this is the power input from the power supply that you've chosen. If it measures a correct voltage, the Smart Receiver may be faulty. Please contact

DiGidot in this case. When no voltage is measured, the device isn't receiving any power and you should diagnose/replace the power supply.

Check hardware and their connections
After verifying the power, it's time to check the equipment and the connection. When a PxLNet Transmitter is used, does it show **any lights** (green and orange) near the RJ45 output ports? If no lights are shown at all, the Transmitter might be dead. You can also temporary **eliminate** the PxLNet Transmitter and connect the data line straight to the DiGidot C4 and **configure DMX** as the output protocol so it uses 2 outputs like D1 and D2 and try again. If that's working, the Transmitter might be dead. Please contact us.

When this also has no effect, verify the cable and the connections

- Check the D+ and D- if they're connected **properly** and, check every potential **weak** connection point (like screw terminals, WAGO's etc.)
- When using a RJ45 cable we follow ESTA standard in terms of wiring

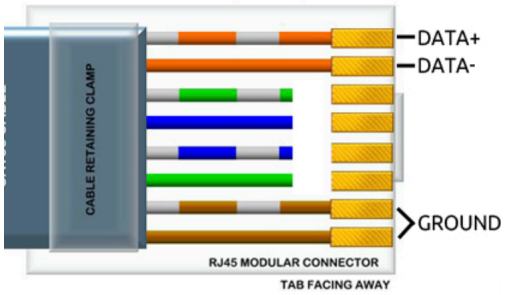


Figure 29 - RJ45 Pinout for PxLNet/DMX

- Check if the **GROUND** is connected properly
- Try to use a **different** or **shorter** cable

Check configuration
If you're using a DiGidot system, make sure it's in-/out configuration is correctly configured. When you're not using a Transmitter, you should configure DMX as output protocol, on D1+D2 or D3+D4 and connect the D+ and D- of the Smart Receiver system accordingly as how you would connect a DMX system.

If you do use a Transmitter, make sure on the in/out configuration page, one of the Supported LED IC's (input section) is selected and the D+ and D- from the Transmitter is correctly connected to the corresponding connections of the Smart Receiver.

Discovery/RDM traffic is very slow

The initial discovery could take some time as all the Smart Receivers are communicating at the same time on the data line to the main controller. If you have only one Smart Receiver, a full discovery, which includes retrieving all parameters, should take about 10 seconds. For every additional device, it can take up to an additional 10 seconds per device, before all the information is retrieved and the discovery completes. If you're using a DiGidot system, you can recognize that a full discover has been completed in two ways:

- The status LED on the DiGidot Controller is going from **yellow** to **blue**
- A blue toast message in the interface on the bottom with "x RDM devices found" is shown

The total expected discovery time = 10 sec x number of devices

If the process takes longer than this calculated time, there is a chance the data connection is not ideal. Check the previous chapter to make sure the data connection is correct.

> Tip: You can increase the speed of the discovery by setting the **RDM parameter discovery response mode** (0x8F02) to 0.

Uploading gives an error

If you encounter an "UploadFailed" error after uploading a new firmware update or Product Profile, consider the following potential causes:

- You're not allowed to upload a new product profile
- An update packet was dropped

To check if you're allowed to upgrade the Product Profile, request RDM parameter 0x8604 from the Smart Receiver and check its value.

- If it's 0; It's okay, you're allowed to update the Product Profile
- If it's 1; You can only upgrade it if you also uploaded the previous Product Profile
- If it's 2; You can't upgrade this device

If upgrading is allowed, you can always try again and see if it works. After 3 attempts, you should check other things: Make sure there is **no scene is played** on the DiGidot Controller and **no Art-Net signal** is being sent to the device.

Verify the reliability of the data connection

When the update process is still failing, it could also be a bad data connection. To verify that the data connection is stable, there are 2 ways to do this;

- Check how long it takes to do a full discovery
- Add the internal sensor called "Invalid Frames"

To check if the discovery time is as expected, check the previous page. For the internal sensor, you can add this by with the help of this chapter: Appendix B – Adding internal sensors.

Appendixes

This chapter contains all the appendixes of the manual. It includes tables with a large amount of data for easy reference.

A: Pinout Smart Receiver

Below is the pinout shown of the Smart Receiver if you want to use the different functionality. Check the datasheet, if you want to use a pin as input or output, and if you want to know, what the absolute maximum ratings are.

> Warning: This device doesn't come with an enclosure. When touching or handling the device, ensure you follow ESD guidelines to avoid damaging the device.

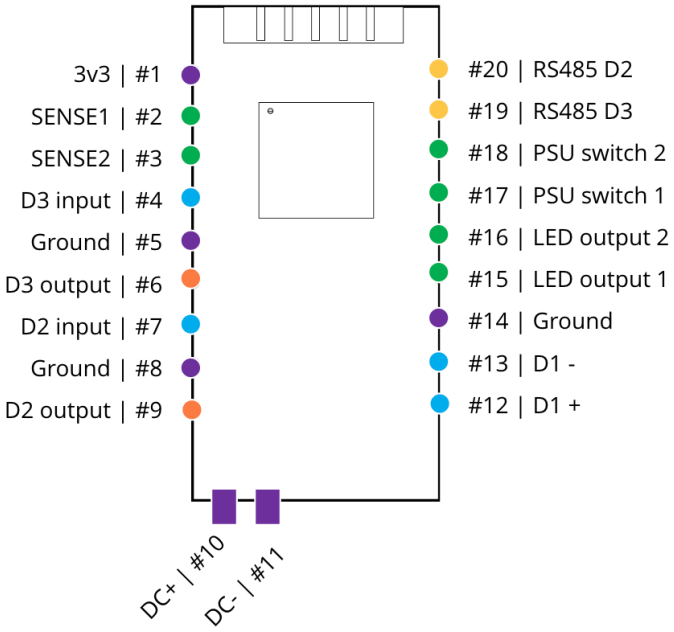


Figure 30 - Pinout Smart Receiver

Pin	Name	Main function	Extra function #1	Extra function #2
1	3v3	3.3 volt output (Power Good signal)		
2	SENSE1	Analog sensor input 1	Button input	
3	SENSE2	Analog sensor input 2	Auto addressing input	
4	D3 input	(future use)		
5	Ground	Shield from data cable		
6	D3 output	Output signal DMX-TTL/SPI		
7	D2 input	(future use)		
8	Ground	Ground reference for D2 output		
9	D2 output	Output signal DMX-TTL/SPI		
10	DC+	Power input positive		
11	DC-	Power input negative		
12	D1+	Main input port, positive, with RDM		
13	D1-	Main input port, negative, with RDM		
14	Ground	Ground reference for D1 input		
15	LED output 1	Status LED output	GPIO	
16	LED output 2	Auto addressing output	GPIO	
17	PSU Switch 1	Power supply switch	I²C interface	GPIO
18	PSU Switch 2	Power supply switch	I²C interface	GPIO
19	RS485 select D2	Set direction external RS485 chip for D2		
20	RS485 select D3	Set direction external RS485 chip for D3		

B. RDM parameter list

Introduction

This chapter contains a full table of all the available parameters the Smart Receiver supports. The device is compliant with this RDM standard:

https://getdlight.com/media/kunena/attachments/42/ANSI_E1-20_2010.pdf

How to read the table

Name
Name of the parameter, also the description of the parameter when retrieving it with RDM

Parameter
16-bit number in HEX format of the specific parameter.

R/W
R = Read (GET), W = Write (SET), RW = Read & Write (GET & SET). Sometimes an asterisk (*) is shown. This means that normally this parameter is RW, but the manufacturer may have configured this parameter to be Read-only.

Range
Type of parameter (number of text) and what valid values can be set.

Some words are used to indicate a number:
INT16: 32,767
UINT16: 65,535
INT32: 2,147,483,648

Sometimes, a minus symbol (-) is used to indicate the number should be read in negative. Example:
-INT16 = -32,767

Info
Provides an explanation of the parameter and what the values represent.

Full parameter table

General RDM (E1.20) parameters

These parameters are supported form the main E1.20 RDM standard

Name	Parameter	R/W	Range	Info
Device Model Description	0x0080	R	-	Returns model info, always "CRX"
Manufacturer Label	0x0081	R	-	Returns manufacturer of CRX, returns "DiGidot"
Device Label	0x0082	RW	Max 32 ASCII	Name of the device
Identify Device	0x1000	RW	0 or 1	When 1, it will highlight the outputs and show the status led (if connected) in a fading green color
Reset Device	0x1001	W	0 or 1	Restarts device when 1 isset
Sensor Definition	0x0200	R	“object”	Returns sensor definition info in E1.20 format
Sensor Value	0x0201	RW	“object”	Returns sensor value info in E1.20 format
Device Hours	0x0400	R	0-INT32	Returns operation hours
Lamp Hours	0x0401	R	0-INT32	Returns number of hours the input is not "Blackout"
Lamp Strikes	0x0402	R	0-INT32	Returns power-on count of the device
Factory Defaults	0x0090	W	0 or 1	When 1 is SET, resets all user settings. Product Profile will be kept

User settings

These settings are always available for the user, to read or write. It defines how many channels it has on the output, which start channel should be used, and what brightness level the outputs should have.

Name	Parameter	R/W	Range	Info
Order ID	0x8001	RW	0-255	Number to show which it is in line. Can be set manually or, if Auto-addressing is used, automatically.
Auto Program ID	0x8002	RW	(-3)-255	Used internally for auto addressing
Brightness D2	0x8003	RW	0-UINT16	16-bit brightness control for output D2
Brightness D3	0x8004	RW	0-UINT16	16-bit brightness control for output D3
Start channel D2	0x8201	RW	0-UINT16	Start channel of output D2
Start channel D3	0x8202	RW	0-UINT16	Start channel of output D3
Channels D2	0x8301	RW	0-UINT16	Number of channels to output on D2
Channels D3	0x8302	RW	0-UINT16	Number of channels to output on D3

General fixture identity information

These parameters are general information about the fixture, how it should be identified in terms of name and manufacturer. It also has an option to enable/disable internal sensors.

Name	Parameter	R/W	Range	Info
Manufacturer label	0x8601	R	40 ASCII	Manufacturer name
Product name	0x8602	R	40 ASCII	Name of product
Manufacturer ID	0x8603	R	0-INT32	Manufacturer Identifier
Lock level	0x8604	R	0-2	Returns current Lock level: 0 = none, 1 = manufacturer only, 2 = fully locked
Internal sensors	0x8605	RW*	0-15	Add/remove internal sensors to the main sensor list. See this section for full explanation.

Fixture general properties

These parameters are used to show the general properties (in terms of physical and electrical) of the fixture like the height or the max current usage.

Name	Parameter	R/W	Range	Info
Fixture length	0x8611	R	0-INT32	Returns fixture length in mm
Fixture height	0x8612	R	0-INT32	Returns fixture height in mm
Fixture width	0x8613	R	0-INT32	Returns fixture width in mm
Minimum operating voltage	0x8621	R	(-INT32)-INT32	Returns minimum operating voltage in mV
Maximum operating voltage	0x8622	R	(-INT32)-INT32	Returns maximum operating voltage in mV
Max current usage	0x8623	R	(-INT32)-INT32	Returns max current usage in uA
Normal power rating	0x8624	R	(-INT32)-INT32	Returns max power usage in mW

Main input port settings

These parameters are used to configure the main input port, if it should handle DMX, PxLNet or both. It also defines if the incoming signal is 1 byte (= 8 bit) or 2 bytes (= 16 bit).

Name	Parameter	RW	Range	Info
Input protocol filter	0x8631	RW*	16 ASCII	Defines allowed input protocols. For more info, check Configuring a Product Profile
Input port	0x8632	R		<Future use>
Input channels per parameter	0x8633	RW*	1 or 2	Defines the size of a single channel: 1 = 8 bit input and 2 = 16 bit input

Output settings D2

These parameters are used to configure output D2

Name	Parameter	RW	Range	Info
D2 Repeat channels take	0x8642	RW*	1-1024	Define how many channels the repeat operation should take. See Repeat section for more info.
D2 Repeat channels count	0x8643	RW*	1-1024	Define how many times the repeat operation should duplicate. See Repeat section for more info.
D2 Protocol base	0x8644	RW*	16 ASCII	Defines the main output protocol. Check the Output protocol section for more info
D2 Protocol period time	0x8645	RW*	200-1,000,000	Defines the output timer its period in ns
D2 Protocol one time	0x8646	RW*	0-10000	Defines the dutycycle for 1 (in SPI mode) in ns
D2 Protocol zero time	0x8647	RW*	0-10000	Defines the dutycycle for 0 (in SPI mode) in ns
D2 Protocol latch time	0x8648	RW*	0-1000	Defines the time for latch (in SPI mode) in us
D2 Color order	0x8649	RW*	16 ASCII	Defines the color order, check the color order section for more info
D2 Channel width	0x864A	RW*	8 or 16	Defines the number of bits for the output, 8 or 16
D2 Fallback color	0x864B	RW*	16 ASCII	Define the color when in fallback mode. Check the Fallback section for more info.
D2 Fallback timeout	0x864C	RW*	0-INT32	Define the time before the fallback color is shown in ms. Check the Fallback section for more info.

Output settings D3

These parameters are used to configure output D3

Name	Parameter	RW	Range	Info
D3 Repeat channels take	0x8652	RW*	1-1024	Define how many channels the repeat operation should take. See Repeat section for more info.
D3 Repeat channels count	0x8653	RW*	1-1024	Define how many times the repeat operation should duplicate. See Repeat section for more info.
D3 Protocol base	0x8654	RW*	16 ASCII	Defines the main output protocol. Check the Output protocol section for more info
D3 Protocol period time	0x8655	RW*	200-1,000,000	Defines the output timer its period in ns
D3 Protocol one time	0x8656	RW*	0-10000	Defines the dutycycle for 1 (in SPI mode) in ns
D3 Protocol zero time	0x8657	RW*	0-10000	Defines the dutycycle for 0 (in SPI mode) in ns
D3 Protocol latch time	0x8658	RW*	0-1000	Defines the time for latch (in SPI mode)
D3 Color order	0x8659	RW*	16 ASCII	Defines the color order, check the color order section for more info
D3 Channel width	0x865A	RW*	8 or 16	Defines the number of bits for the output, 8 or 16
D3 Fallback color	0x865B	RW*	16 ASCII	Define the color when in fallback mode. Check the Fallback section for more info.
D3 Fallback timeout	0x865C	RW*	0-INT32	Define the time before the fallback color is shown in ms. Check the Fallback section for more info.

Automatic Power Supply Unit witch configuration

These parameters are used to view/configure the behaviour of the Power Supply that needs to be controlled.

Name	Parameter	RW	Range	Info
PSU Switch Pin	0x8661	R	0-255	Defines which output pin is used for controlling the PSU. 0 = unused.
PSU Switch Open Drain	0x8662	R	0 or 1	0 = Using Push-Pull / 1 = Using Open-Drain
PSU Switch Default Set	0x8663	R	0 or 1	0 = Normally not set / 1 = Normally set
PSU Switch Timeout Blackout	0x8664	RW*	0-INT32	Time in ms, from when a blackout is detected, until the PSU turns off
PSU Switch Timeout No Signal	0x8665	RW*	0-INT32	Time in ms, from when no signal is detected, until the PSU turns off

Additional commands

Name	Parameter	RW	Range	Info
Discovery response mode	0x8F02	R/W	0 or 1	0 = hide template configure parameters (faster discovery) 1 = show all parameters (slower)
Set Effect	0x8F04	R/W	0-5	0 = input, 1-4 are static colors and 5 is fading red

Add/change Internal sensors

The internal sensor RDM parameter is a number between 0 and 15. To configure this, determine which sensors you want to add and sum their associated numbers. This resulting value can then be set using RDM.

For example, if you want to have the **Input FPS** and **Output FPS**, you need to add the numbers 2 and 4 together so you get 6 in the end. If you don't want to have any internal sensors, the value can be set to 0.

ADD THIS NUMBER	SENSOR NAME	DESCRIPTION
1	Uptime in seconds	Gives the uptime of the device in seconds. After 9 hours, this value will be frozen.
2	Input FPS	Show the current input FPS
4	Output FPS	Show the current output FPS (both outputs are rendered at the same FPS)
8	Invalid frames	Show the number of frames that are dropped due to bad connection in the wiring

C. Supported LED IC's

This chapter is about which LED IC's the Smart Receiver supports on its input and output. Besides SPI, DMX512 is also supported on both the input and outputs. On the output, it also supports the faster variant of DMX like DMX1024, DMX1536 and DMX2048. The table below indicates the support status with **Red** indicating "not supported" and **Green** indicating "supported".

IC	Input	Output
APA104		
APA106		
CS8812		
DMX512		
DMX1024		
DMX1536		
DMX2048		
INK1003		
SK6812 (+RGBW)		
SK6822		
SM16703		
SM16704		
TM1803		
TM1804		
TM1809		
TM1812		
TM1934		
UCS1903		
UCS1904		
UCS2903		
UCS2904		
UCS2909		
UCS2912		
UCS512B3		
UCS8903		
UCS8904		
WES9412		
WES943		
WS2811		
WS2812		
WS2812B		
WS2812S		
WS2813		
WS2815		
WS2818		

VARIOUS

Online resources

For technical specifications, latest documentation, manuals, product information, support and upgrades, please visit digidot.eu.

Remarks

We've put great care in writing this manual. However, in case you encounter any discrepancies or unclarities, please contact us. This manual and feature specific instructions are based on firmware and interface versions mentioned in the chapter Manual version.

Terminology

CRX	Different name for Smart Receiver
SPI	Serial Peripheral Interface: this is an interface bus that is used to send data between microcontrollers and 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 as one DMX control network consisting of 512 control channels, also typically associated with one DMX output. DiGidot also refers to a universe as one set of 512 control Channels that belong to specific Art-Net, sACN or DMX universes. But when we mention more than one universe on one output for example, we always refer to them as amount of control channels and not necessarily as DMX universes.
I/O configuration	Input/Output configuration, determines how and which signals the controller are received and how the ports of the controllers are configured for sending a specific signal.
DMX	**D**igital **M**ultiple**X** is a universal lighting control protocol which is standardized by USITT and ESTA/Plasa.
RDM	**R**emote **D**evice **M**anagement: Is the implementation of Remote Device Management based on ANSI E1.20
Product Profile	A configuration that describes how the Smart Receiver should work in terms of input/output protocol / sensors / pixel transformation etc.
Blackout	A data signal where all the values are zero.

Compliances & EU declaration of Conformity

DiGidot Technologies BV is a registered DMX-RDM manufacturer at tsp.esta.org, under ID 0x00B6
DiGidot Technologies BV registered Art-Net manufacturer at artisticlicence.com, under ID 0x0FD0.

Warranty

This product is covered by a carry-in manufacturer's warranty of 1 years which covers any design faults, production faults and component failures.
Warranty voids if the product was installed or used incorrectly or not in accordance with this manual, and/or if the product was damaged due to external factors, opened, modified or electrically overloaded. Warranty conditions of DiGidot Technologies B.V. apply. Warranty claims have to be issued by email: support@digidot.eu.

Disposal and recycling

This product should not be disposed with other household waste. When you decide to dispose this product and/or its battery, do so in accordance with local environmental and recycling regulations.

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 TG - Amsterdam
The Netherlands
Phone: +31 (0)20 820 1849

info@digidot.eu
www.digidot.eu

/Digidot-Technologies-b.v.

/DigidotTechnologies

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

Trademarks

Used trademarks and product names belong to their respective owners and are only being used for illustrational purposes.

Copyright

© 2024 DiGidot Technologies BV. All rights reserved.
Errors and omissions excepted. The information in this document is subject to change at any time without prior notice. Visit our website digidot.eu for the most recent version of this document. It's not allowed to copy or reproduce, translate or publish the information within this document without prior written approval from DiGidot Technologies BV.

Disclaimer

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 its 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.

Manual version

This manual version is written for the DiGidot PxLNet Smart Receiver with the following versions:

- 1.0.0
- 1.0.1

Some functions may be different and new features may have been added. Please make sure that you're using the manual that matches your firmware version.