

IOCORE1 MANUAL



Revision History

| Revision | Date | Author(s) | Description |
|----------|------------|-----------|---|
| 1 | 26.04.2016 | ME | Initial version. |
| 2 | 19.12.2017 | ME | Added: Rackmount accessory. Replaced VisualTouch info by Kiosc. vManager chapter added. |

Contents

| | | |
|-----------|----------------------|-----------|
| 1 | Introduction | 6 |
| 2 | Protocols | 9 |
| 3 | Quickstart | 13 |
| 4 | Setting up | 22 |
| 5 | Network | 24 |
| 6 | Show Control | 27 |
| 7 | Monitors | 32 |
| 8 | Settings | 34 |
| 9 | vManager | 40 |
| 10 | Kiosk | 44 |
| | Appendices | 48 |
| A | Trigger Types | 49 |
| B | Task Types | 56 |
| C | Templates | 62 |
| D | API | 63 |

©2017 Visual Productions BV. All rights reserved.

No parts of this work may be reproduced in any form or by any means - graphic, electronic, or mechanical, including photocopying, recording, taping, or information storage and retrieval systems - without the written permission of the publisher.

While every precaution has been taken in the preparation of this document, the publisher and the author assume no responsibility for errors or omissions, or for damages resulting from the use of information contained in this document or from the use of programs and source code that may accompany it. In no event shall the publisher and the author be liable for any loss of profit or any other commercial damage caused or alleged to have been caused directly or indirectly by this document.

Due to the dynamic nature of product design, the information contained in this document is subject to change without notice. Revisions of this information or new editions may be issued to incorporate such changes.

Products that are referred to in this document may be either trademarks and/or registered trademarks of the respective owners. The publisher and the author make no claim to these trademarks.



Declaration of Conformity

We, manufacturer Visual Productions BV, hereby declare under sole responsibility, that the following device:

IoCore

Is in conformity with the following EC Directives, including all amendments:
EMC Directive 2004/108/EG

And the following harmonized standards have been applied:
NEN-EN-IEC 61000-6-1:2007
NEN-EN-IEC 61000-6-3:2007

Full name and identification of the person responsible for product quality and accordance with standards on behalf of the manufacturer

Date:
April 29th, 2013

Place:
Haarlem, The Netherlands



ing. Maarten Engels
Managing Director
Visual Productions BV

WWW.VISUALPRODUCTIONS.NL
INFO@VISUALPRODUCTIONS.NL
TEL +31 (0)23 551 20 30
FAX +31 (0)23 551 68 30
ROLLANDSTRAAT 55ZW
NL-2013 SM HAARLEM
THE NETHERLANDS
COG 54497795
VAT NL851328477B01
ABN-AMRO BANK 53.22.22.261
BIC ABNANL2A
IBAN NL18ABNA053222261

Chapter 1

Introduction

Thank you for choosing the IoCore1. The IoCore1 is expansion module for the CueCore and QuadCore range of solid-state lighting controllers. The IoCore1 can add extra GPI, GPO, RS-232 and DMX ports to a CueCore based installation.

Furthermore, the IoCore is perfectly capable of working stand-alone when there is no CueCore attached. It can be programmed to convert and trigger any of its protocols autonomously. Given the open nature of the IoCore's protocol suite (UDP, OSC, Art-Net) this unit can easily be integrated with third-party systems.

An internal web-server provides the web-interface through which you can program the IoCore1. A modern browser is required to access this web-interface during set-up. A browser or computer is not required for standalone use after the initial set-up.



Figure 1.1: IoCore1

This manual discusses setting up and programming the unit. Chapter 2 provides background information on the communication protocols used the IoCore1. Chapters 4 and 5 cover how to set up the unit and configure the network connection.

Programming the triggering and converting functionality is done in the 6 chapter as discussed on page 27.

At the time of writing this manual the IoCore1's firmware was at version 1.39.

The IoCore1 is end-of-life and is not recommended for new designs. It has been superseded by IoCore2.

1.1 Features

The feature set of the IoCore1 includes:

- 8 GPI ports (can be set as contact-closure or 0-10V input)
- 8 GPO relay ports
- Ethernet based, supporting protocols like UDP, OSC and Art-Net
- DMX-512 port (can be set as input or output)
- RS-232 port
- Web-based user-interface for programming
- PoE (Power Over Ethernet) Class I
- Compatible with CueCore, QuadCore and B-Station
- Bundled with vManager and Kiosc software

1.2 What's in the box?

The IoCore1 packaging contains the following items (see figure 1.2):

- CD-ROM
- IoCore1
- Power supply (including international plug set)



Figure 1.2: IoCore1 box contents

1.3 Saving data to memory

This manual will describe how to configure the IoCore1 and program conversions, action, etc. The unit's web-interface is used for editing these kinds of elements. When changes are made, these changes are directly stored in the RAM memory of the IoCore1 and the programming will directly influence the behaviour of the unit. RAM memory is, however, volatile and its content will be lost through a power cycle. For this reason the IoCore1 will copy any changes in the RAM memory to its onboard flash memory. Flash memory retains its data even when not powered. The IoCore1 will load all its data back from the flash memory upon startup.

This memory copy process is conducted automatically by the IoCore1 and should not be of any concern of the user. One point of consideration is, however, that after making a change the unit should be given time to perform the copy to flash. As a rule of thumb, do not disconnect the power from the device within 30 seconds from making a programming change.

1.4 Further Help

If, after reading this manual, you have further questions then please consult the online forum at <http://forum.visualproductions.nl> for more technical support.

Chapter 2

Protocols

The IoCore1 is fitted with several communication ports and supports various protocols. This chapter describes these protocols and to which extent they are implemented in the IoCore1.

2.1 GPI

The IoCore1 features eight General Purpose Inputs (GPI) ports that can be connected to external equipment, switches and sensors. State changes on these GPI ports can be used to trigger programmed events inside the IoCore1.

Each GPI port can be switched between 'digital' and 'analog'. In the digital mode the signal is held up by an internal pull-up resistor and results in a logic '0'. The external equipment is intended to short the port's pin to the provided ground pin. This short will create a logic '1'. All four ports share one common ground pin.

When set to analog the external equipment is supposed to supply a voltage between 0V and 10V to the port's pin. Please refer to figure 2.1 for the pinout of the GPI connector. **Be careful not to supply more than 10V to the GPI port as that might cause permanent damage.**

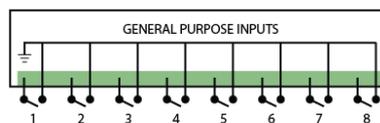


Figure 2.1: GPI Pinout

Please refer to page 36 for more information on configuring the GPI ports. Programming events based on GPI activity is done in the Show Control page, which is discussed on page 27.

2.2 GPO

Each GPO port features a normally-open (NO) relay for switching external equipment. The relay can switch 250VAC/30VDC up to 2A. Please refer to figure 2.2 for the pinout of the GPO connector.

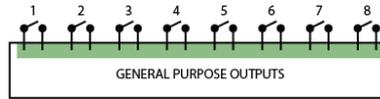


Figure 2.2: GPO Pinout

2.3 RS-232

The bi-directional RS-232 port can be used to communicate with external equipment such as video projectors and matrix switchers. Figure 2.3 shows the pinout of the connector which shared with the DMX port. Please refer to page 36 for more information on configuring the RS-232 port.



Figure 2.3: RS-232 Pinout

2.4 DMX-512

DMX-512 is the standard communication protocol for stage lighting. Its official name is E1.11-2008 USITT DMX512-A. Nowadays the reach of the DMX protocol has extended beyond entertainment lighting and is also used for architectural lighting. Originally one DMX network contained 512 channels which is called a 'universe'. With the growing size and complexity of lighting systems it is now very common for a system to compose of multiple universes, each conveying 512 channels. It is advised to use a shielded twisted pair cable for DMX cabling. The cable should be terminated with an 120 Ohm resistor.

DMX-512 is a very successful protocol with, however, a few limitations. The maximum number of attached devices is limited to 32 and they all have to be

connected in bus-topology having one cable running via each device. Furthermore, a DMX-512 cable should not be longer than 300 meters.



Figure 2.4: Visual Productions' RdmSplitter

The DIN Rail RdmSplitter from Visual Productions (See figure 2.4) helps tackle those inconvenient limitations. The Splitter takes a DMX signal and sends it out again on its 6 DMX output ports for scaling group topology. Each output port is capable of driving 32 more devices. The Splitter can also function as a signal booster as each port supports another 300 meter long connection.

The IoCore1 has one DMX port and is therefore able to control 512 channels. The port can be configured either as input or output. Figure 2.5 shows the pinout of the connector.

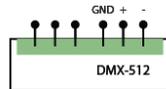


Figure 2.5: DMX Pinout

2.5 Art-Net

The Art-Net protocol primarily transfers DMX-512 data over Ethernet. The high bandwidth of an Ethernet connection allows Art-Net to transfer up to 256 universes. The data sent out for Art-Net does put a certain load on the network, therefore it is recommended to disable Art-Net when not in use. The IoCore1 can receive and send out one DMX universe over Art-Net. Art-Net can be used for creating triggers in the show control programming.

2.6 TCP

The Transmission Control Protocol (TCP) is a core protocol of the Internet Protocol Suite. It is used for its reliable, ordered and error checked delivery of a stream of bytes between applications and hosts over IP networks. It is considered 'reliable' because the protocol itself checks to see if everything that was transmitted was delivered at the receiving end. TCP allows for the retransmission of lost packets, thereby making sure that all data transmitted is received.

The IoCore1 supports reception of TCP message.

2.7 UDP

User Datagram Protocol (UDP) is a simple protocol for sending messages across the network. It is supported by various media devices like video projectors and Show Controllers. It does not incorporate error checking, therefore it is faster than TCP but less reliable.

There are two ways how to have the IoCore1 respond to incoming UDP messages. The API (see page 64) makes typical IoCore1 functions available through UDP. Furthermore, custom messages can be programmed in the Show Control page (see page 27). This is also the place where to program outgoing UDP messages.

2.8 OSC

Open Sound Control (OSC) is a protocol for communicating between software and various multi-media type devices. OSC uses the network to send and receive messages, it can contain MIDI and custom information.

There are apps available for creating custom-made user interfaces on iOS (iPod, iPhone, iPad) and Android. These tools allow to program fool-proof user-interfaces for controlling the device. E.g. Kiosc from Visual Productions.

There are two ways how to have the IoCore1 respond to incoming OSC messages. Firstly, the API (see page 63) makes typical IoCore1 functions available through OSC. Secondly, custom messages can be programmed in the Show Control page (see page 27).

Chapter 3

Quickstart

This chapter provides step by step tutorials on how to program your IoCore1 for some typical tasks:

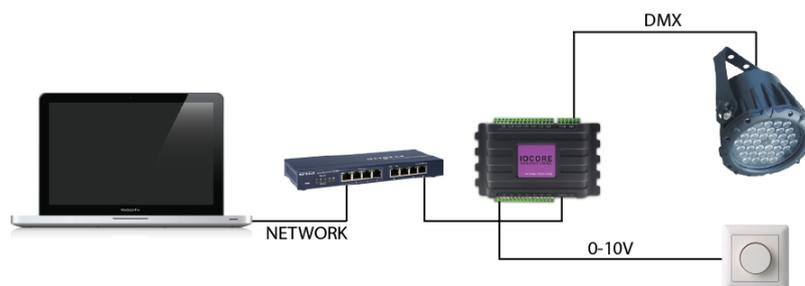
- Convert 0-10V levels to DMX
- Use DMX to trigger a RS-232 lamp strike command to a video projector

3.1 Convert 0-10V levels to DMX

This tutorial shows how to use the IoCore1 to convert 0-10V levels to DMX. The 0-10V levels could be generated by sensors (e.g. ambient lighting sensor) or a potentiometer (1-10V wall mount rotary dimmer). The DMX levels can then be used to directly control lighting fixtures or it can be fed into an external lighting console to control the intensity of complete scenes. Please follow the next few steps.

1. Setup network and connections

Connect the IoCore1 and the computer to a network switch. Connect the rotary dimmer to GPI port 1. Connect the fixture to the DMX port and set it to starting address 1.



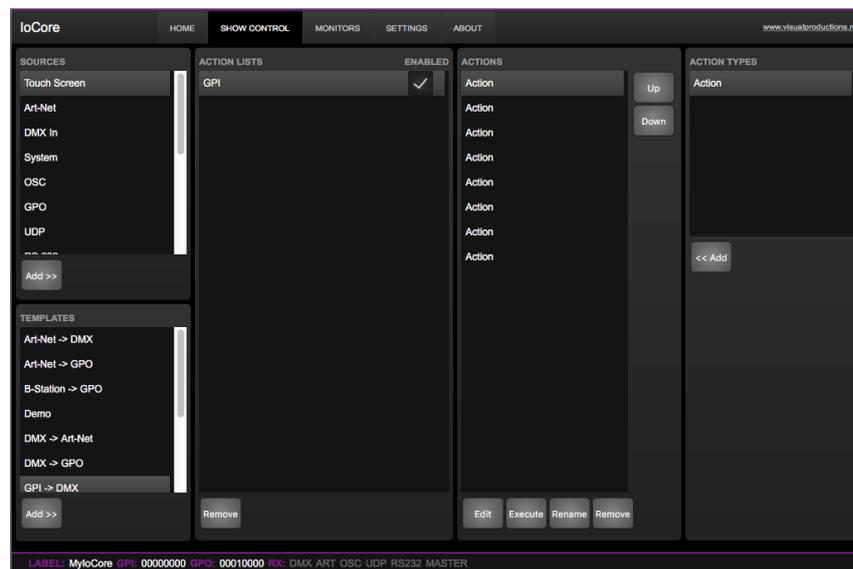
2. Connect to the network

Configure the equipment to have the following static IP addresses:

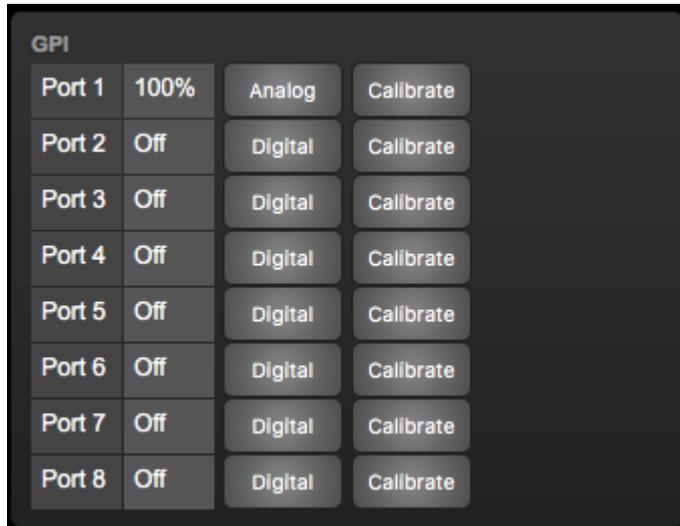
| | IP | Subnet |
|----------|--------------|---------------------------------|
| Computer | 192.168.1.9 | 255.255.255.0 |
| IoCore1 | 192.168.1.10 | 255.255.255.0 (Default setting) |

3. Add template

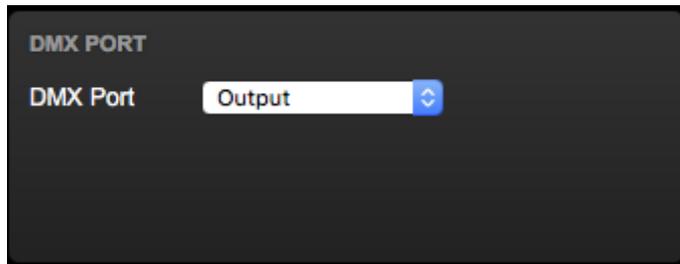
Open a browser on the computer and browse to the web-interface of the IoCore1 by typing '192.168.1.10' in the browser's address field. Go to the Show Control page and select the 'GPI ->DMX' template. Press the 'Add >>' button in the lower left corner of the screen.



4. Configure GPI port
Go to the IoCore1 Settings page and set GPI port 1 to 'analog'.



5. Configure the DMX port
In the settings page, set the DMX port direction to 'output'



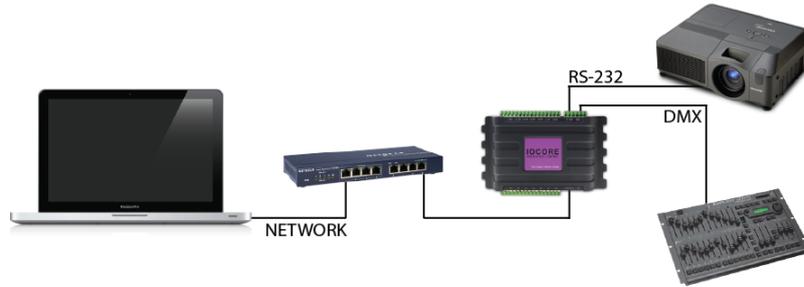
Changing the rotary encoder should now control DMX channel 1 for the fixture.

3.2 Use DMX to trigger a RS-232 lamp strike command to a video projector

This example enables an external DMX source to trigger a RS-232 message that will be send to a video projector; the RS-232 message could be used to strike the lamp within the projector. Please take the following steps:

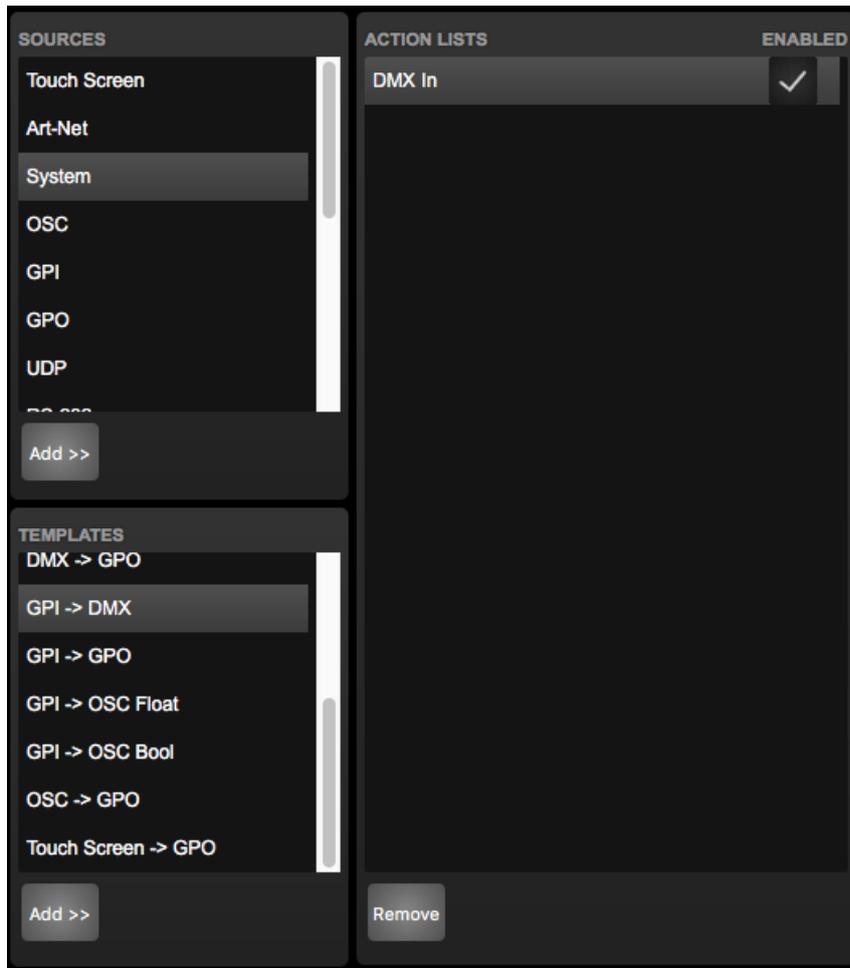
1. Setup network and connect equipment

Connect the IoCore1 and the computer to a network switch. Connect the the external DMX board to the DMX port. Connect the projector to the RS-232 port.



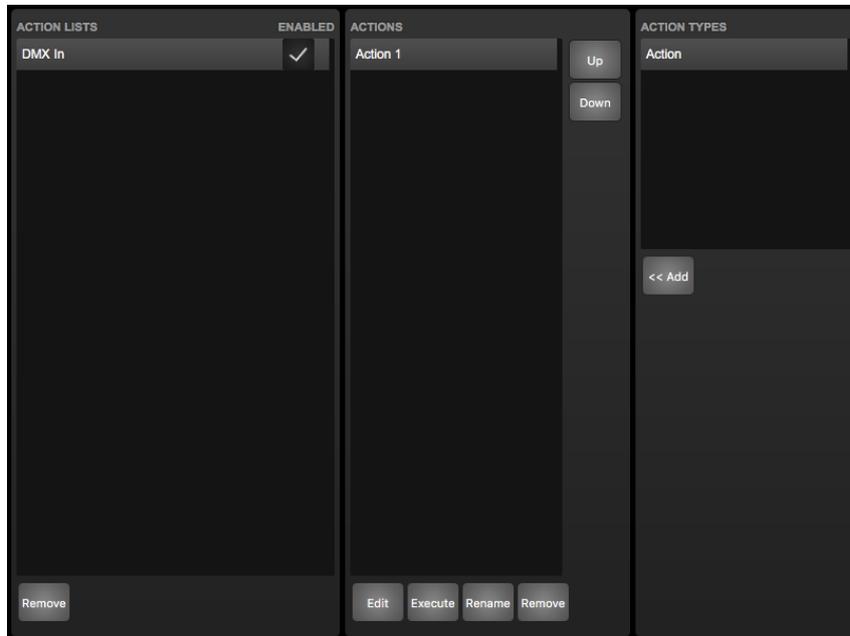
2. Add source

Open a browser on the computer and browse to the web-interface of the IoCore1 by typing '192.168.1.10' in the browser's address field. Go to the Show Control page and select the 'DMX In' source. Press the 'Add >>' button below the table.



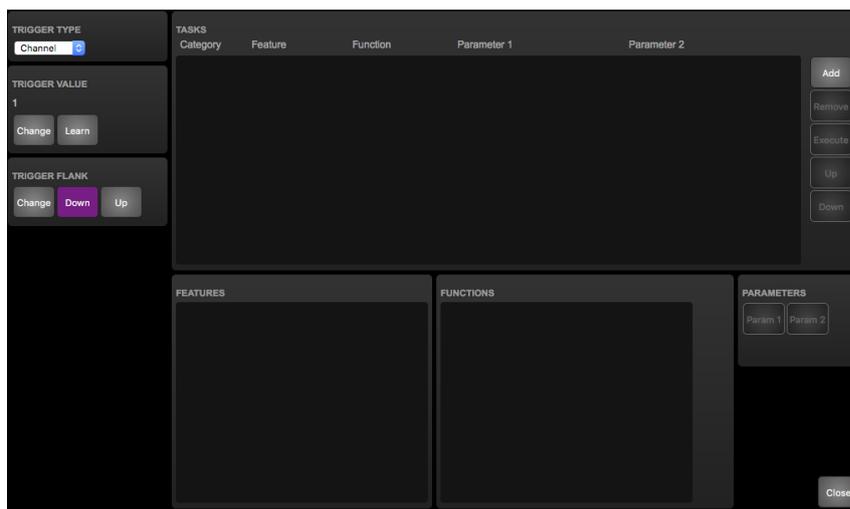
3. Add action

Make sure the 'DMX In' actionlist is selected and press the '<<Add' button below the 'Action Types' table.



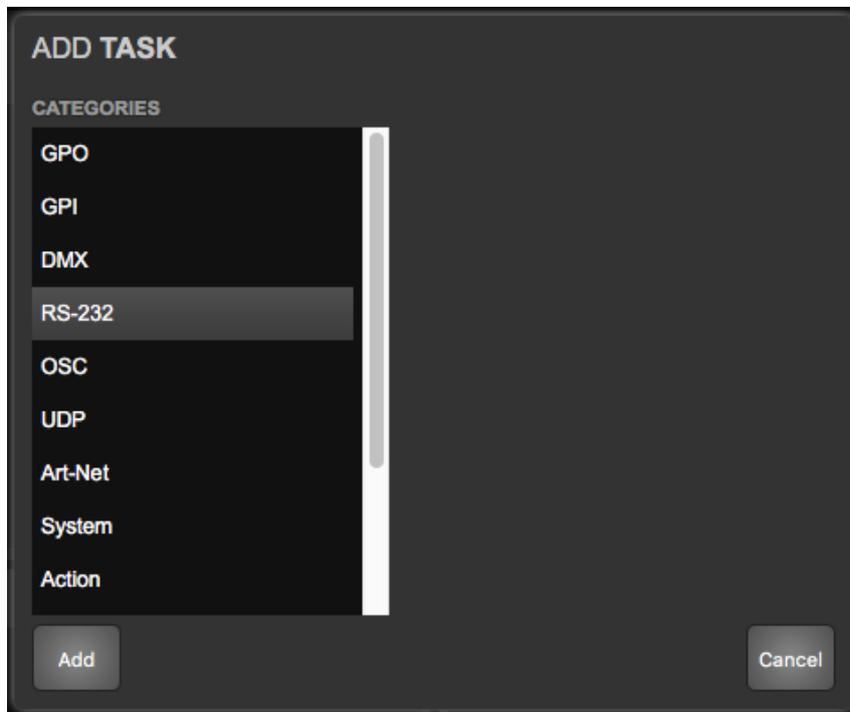
4. Edit action

Make sure the new action is selected and press the 'Edit' button. Set the flank to 'Down'. This will make sure that the action is only triggered when DMX channel 1 goes from 0% to non-zero.



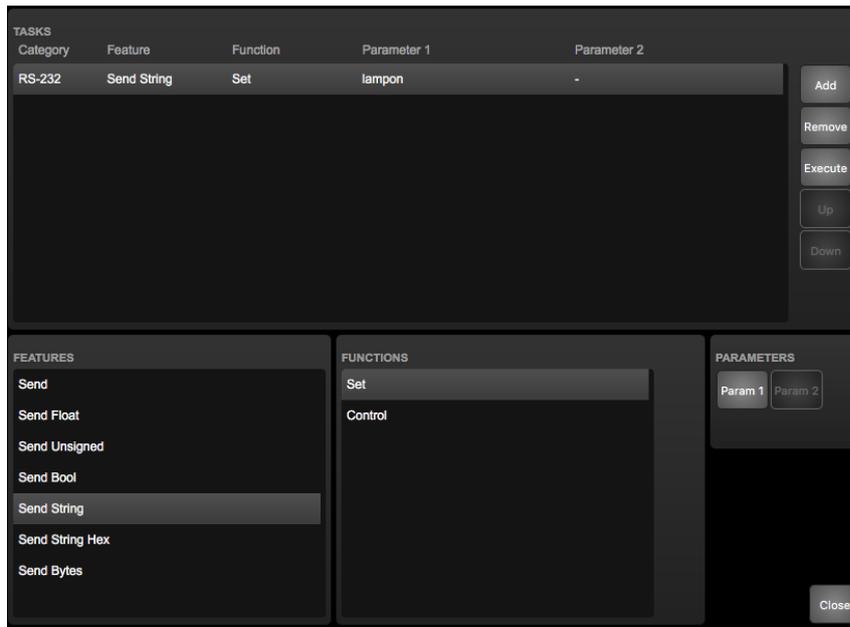
5. Add task

Press the 'Add' button at the top left of the screen and choose 'RS-232' from the task categories.



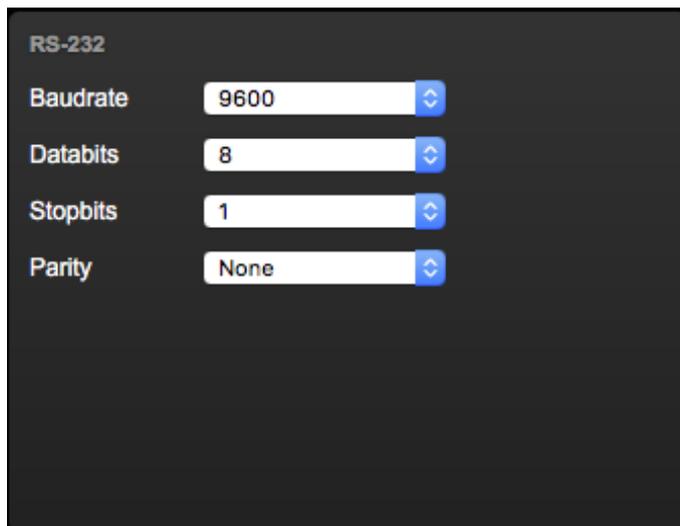
6. Edit task

Select the task and change the feature to 'Send String'. Set the function to 'Set'. Enter e.g. 'lampon' in parameter 1. You can now exit the dialog by pressing the 'Close' button. Please consult the projector's documentation and replace the 'lampon' string accordingly.



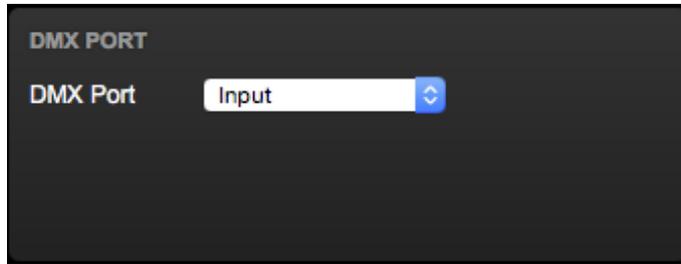
7. Configure RS-232 port

Go to the IoCore1 Settings page. Set the RS-232 details to match specifications that would be found in the projector's documentation.



8. Configure the DMX port

In the settings page, set the DMX port direction to 'input'



Bringing channel 1 up from 0% will now make the IoCore1 send the RS-232 string to the projector. The DMX console should bring channel back to 0% so it is ready to triggered again.

Chapter 4

Setting up

This chapter discusses how to set up the IoCore2.

4.1 Mounting

The device can be placed desktop or it can be DIN Rail mounted. The device is prepared for DIN Rail mounting by using the 'DIN rail holder TSH 35' from Bopla (Product no. 22035000).

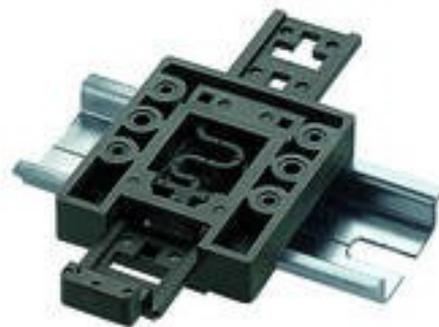


Figure 4.1: Bopla DIN rail adapter

This adapter is - amongst others - available from:

- Farnell / Newark (order code 4189991)
- Conrad (order code 539775 - 89)
- Distrelec (order code 300060)

4.2 Rackmount

There is an adapter available for mounting the IoCore1 into a 19" rack . The rackmount adapter is 1 HE and is sold separately. It fits two units, however, it is supplied with one position closed by a blind panel, see figure 4.2.



Figure 4.2: Rackmount adapter

4.3 Kensington Lock

The device can be secured by using a Kensington style laptop lock.



Figure 4.3: Kensington lock

4.4 Power

The IoCore2 requires a DC power supply of 12 Volt with a minimum of 500mA. The 2,1 mm DC connector is center-positive.

The IoCore2 is also Power-over-Ethernet (PoE) enabled (starting from hardware version v1.6). It requires PoE Class I.



Figure 4.4: DC polarity

Chapter 5

Network

The IoCore1 is a network capable device. A network connection between a computer and the unit is required to configure and program the IoCore1, however, once the device is programmed then it is not necessary anymore for the IoCore1 to be connected to an Ethernet network.

There are multiple arrangements possible for connecting the computer and the IoCore1. They can be connected peer-to-peer, via a network switch or via Wi-Fi. Figure 5.1 illustrates these different arrangements.

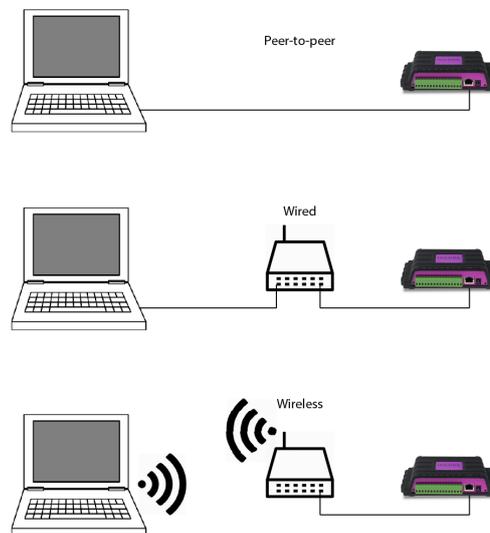


Figure 5.1: Network arrangements

The Ethernet port on the IoCore1 is auto-sensing; it does not matter whether a cross or straight network-cable is being used.

5.1 IP Address

The IoCore1 only supports static IP addresses. By default, the IoCore1 is set to 192.168.1.10.

There are three ways to change the IP address setting of the IoCore1.

- **vManager** can be used to detect a IoCore1 on the network. Once found, the vManager software (figure ??) allows for changing the IP address and subnet mask.
- If the IP address is already known then browsing to this address using the computer's browser will show the IoCore1's **web-interface**. The Settings page on this web-interface enables changing the IP address and subnet mask settings.
- By pressing the **reset button** on the device for 3 seconds, it will reconfigure the unit to the factory default IP address and subnet mask. No other settings will be changed. The default IP address is 192.168.1.10 with the subnet mask set to 255.255.255.0.



Figure 5.2: Reset button

5.2 Access via Internet

The IoCore1 can be accessed through the Internet. There are two ways to achieve this: Port-Forwarding and VPN.

- **Port-Forwarding** Is relatively easy to setup in the router. Each router is different so it is advised to consult the router's documentation (sometimes it is referred to as NAT or Port-Redirecting). Please note that port forwarding is not secure, since anybody could access the IoCore1 this way.
- Accessing via a **Virtual Private Network** (VPN) tunnel requires more setup efforts, also the router needs to support the VPN feature. Once set up, this is a very secure way to communicate with the IoCore1. A VPN is a network technology that creates a secure network connection over a public network such as the Internet or a private network owned by a service provider. Large corporations, educational institutions, and government

agencies use VPN technology to enable remote users to securely connect to a private network. For further information about VPN please refer to <http://whatismyipaddress.com/vpn>.

Chapter 6

Show Control

The IoCore1 can interact with the outside world; it can receive messages and values through various protocols and it can send out many protocols. It is possible to automate the IoCore1 by having it respond automatically to incoming signals. An example of this would be to control the GPO relays by receiving a specific UDP network message. The Show Control page (See figure 6.1) enables this kind of programming to be made.

When using the IoCore1 as an expansion to the CueCore controller then it is advised to use OSC messages to establish communication between IoCore1 and CueCore. To setup the IoCore1 quickly for this purpose it is recommended to add a template. Templates are discussed on page 30.

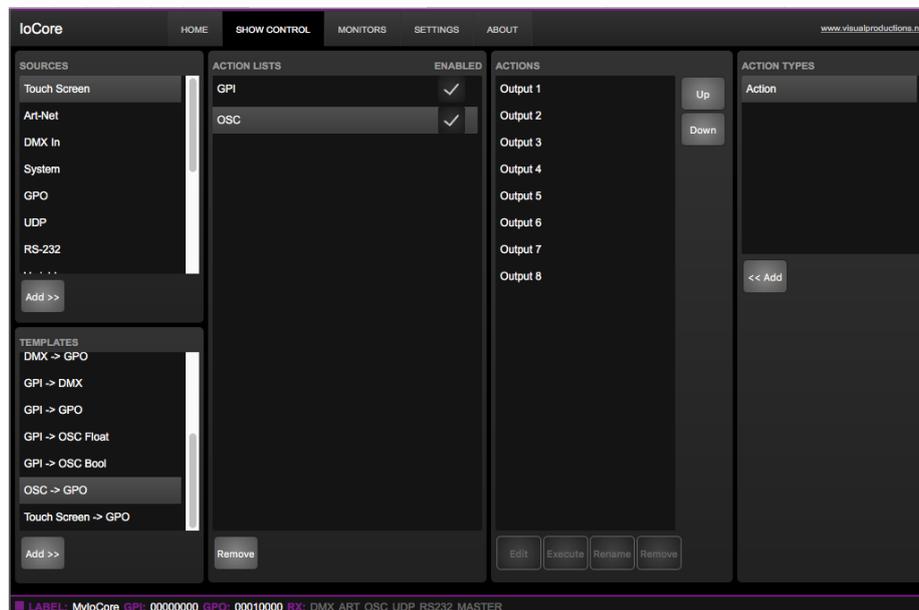


Figure 6.1: Show Control page

The Show Control page presents a system of 'actions'. A signal that the Io-

Core1 needs to respond to or perhaps convert into some other signal, needs to be expressed in an actions. Before programming actions please consider the Show Control structure in figure 6.2.

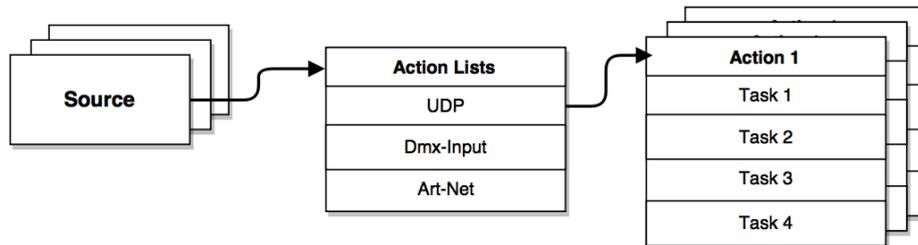


Figure 6.2: Show Control structure

The IoCore1 is capable of listening to various protocols. These available protocols are listed in Sources, however, the IoCore1 can only actively listen to 8 protocols at once. The active protocols are listed in 'Action Lists'. Each action list can contain actions. Within a protocol/source each individual signal requires its own action. For example, when listening to channel 1 and 2 on the incoming DMX, the DMX action list needs two actions; one for each channel.

Inside the action we define the trigger and tasks. The trigger specifies for which signal to filter. In the above DMX example the trigger would be set to 'channel 1' and 'channel 2' respectively. The tasks determine what the IoCore1 will do when this action is triggered. Several tasks can be placed in the action. There are tasks available for a wide range of IoCore1 features and external protocols. Task types are detailed in Appendix B on page 56.

6.1 Sources and Action Lists

The Sources listing presents all protocols that the IoCore1 is capable of receiving. It also includes internal features that can create events that can be used for triggering actions, such as the GPI ports. These sources are available, however, they will only be actively listened to once moved to the action-list table.

| Source | Description |
|---------------|---|
| Touch Screen | Triggers from Kiosc. For each Action various controls can be chosen such as buttons and sliders, colour picker etc. The order of the actions will control the arrangement in Kiosc. |
| Art-Net | Art-Net DMX data |
| DMX In | Receiving DMX data when the DMX port is set as input |
| System | Events such as 'Power on' |
| OSC | OSC network message |
| GPI | Changes on one of the GPI ports |
| GPO | A GPO port changes state |
| UDP | UDP network messages |
| RS-232 | Receiving RS-232 serial messages |
| Variable | The Variable source works in combination with the variable task (For more information about the Variable task please refer to Task Types). The Variable task will set a value of which an enabled action-list type with Variable as Source will use as a trigger. |
| Randomiser | The randomiser can generate a random number |
| User List 1-4 | These action-lists will never trigger an event, however, they are useful for advanced programming. |

Action-lists can be temporarily suspended by disabling their checkbox in the Show Control page. There is also a task available to automate changing the state of this checkbox.

6.2 Actions

Actions are executed when a certain signal is received. This signal is defined by the trigger. A trigger is always relative to the action-list the action belongs to. For example, when the trigger-type is set to 'Channel' then it refers to a single DMX channel if the action is placed inside a 'DMX Input' list and it means a single Art-Net channel if the action resides in an Art-Net action-list.

A trigger is determined by the trigger-type, trigger-value and trigger-flank fields. Although these fields are not applicable for all action-lists and are therefore sometimes omitted in the web GUI. The trigger-type field specifies what kind of signal the action will be triggered by. For example, when making an action in the Art-Net list there is the choice between 'Channel' and 'Receiving' trigger-types. The trigger-value specifies the actual signal value. In the Art-Net example the trigger-value could be set to 'channel 1' or 'channel 2'.

In some action-lists actions do also need to specify the trigger-flank. The flank

further specifies the value that the signal should have before triggering the action. For example, when an action is triggered from a Touch Screen list and it is linked to a button in the Kiosk software, the flank will determine whether to trigger only when the button goes down or only when it goes up. Appendix A provides an overview of the available trigger-types.

An action-list can have up to 48 actions, system-wide there is a maximum of 64 actions.

6.3 Tasks

Tasks are added to an action in order to specify what to do when it gets executed. Up to 8 tasks can be included in an action, systemwide there is a maximum of 128 tasks. The tasks are executed in the order of the list. There is a wide selection of tasks available to choose from, they include altering any of the internal software features like GPOs but also sending out messages through any of the supported protocols. The tasks are organised in categories. Once a task is chosen from these categories each task allows for further choice between several 'Features' and 'Functions'. Tasks contain up to two parameters that might be required for its execution.

If the event that triggers the action passes a parameter along then this parameter can be used in a task. The 'set' function makes a task use a fixed value, however, when using the 'control' function then the trigger's parameter is used. This is very useful for conversions between protocols.

For example when converting 0-10V to DMX the GPI action specifies the port (e.g. #1) and flank (e.g. OnChange) on which it will trigger. The actual 0-10V level sampled on the GPI port will be passed along and fed into the action. Then when a task (e.g. DMX) uses the function 'control' this 0-10V level will be used for setting the DMX value.

A task can be tested by selecting it and pressing the 'execute' button in the action-edit dialog. The complete action can also be tested; go to the Show Control page, select the action and press the 'execute' button.

Appendix B provides a detailed overview of the available tasks, features, functions and parameters.

6.4 Templates

The Show Control page presents a list of templates. A template is a set of action-list, actions and task. These templates configure the IoCore1 to perform typical functions; for example convert Art-Net to DMX or control the 8 relays through OSC. The templates thus save time; otherwise actions should have been set up manually. They can also function as a guide to soften the learning curve on actions; a lot can be learned from adding a template and then exploring the actions and tasks it created.

Please note that some templates require settings to be changed in the settings page; for example the 'Receiving Art-Net' template needs the Art-Net subnet/universe to be specified in order to achieve an Art-Net to DMX conversion. Appendix C gives an overview of the available templates.

6.5 Variables

Variables are internal memories that can hold a value; a number in the range of [0,255]. There are 8 variables and they are typically used for advanced show control programming. In the IoCore1, the content of the variable is not stored between power cycles.

Variables can be set by tasks. Variables can be added as sources in order to have actions triggered when a variable changes value.

6.6 Randomizer

The randomizer is an internal software feature that can generate a (pseudo-)random number. This is useful for having an event trigger a random lighting scene in a themed environment. The randomizer is activated by the Randomizer-task. The result of the randomizer's calculation can be obtained by catching the event in the Randomizer-actionlist.

Chapter 7

Monitors

This page allows the user to inspect the incoming and outgoing data, both DMX-type data (See figure 7.1) as well as control messages (See figure 7.2). Monitoring incoming and outgoing data can help the user troubleshoot during programming.

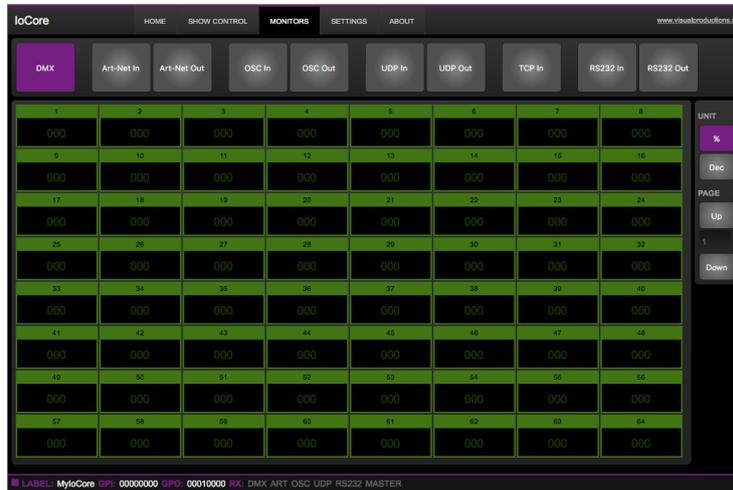


Figure 7.1: DMX Monitor page

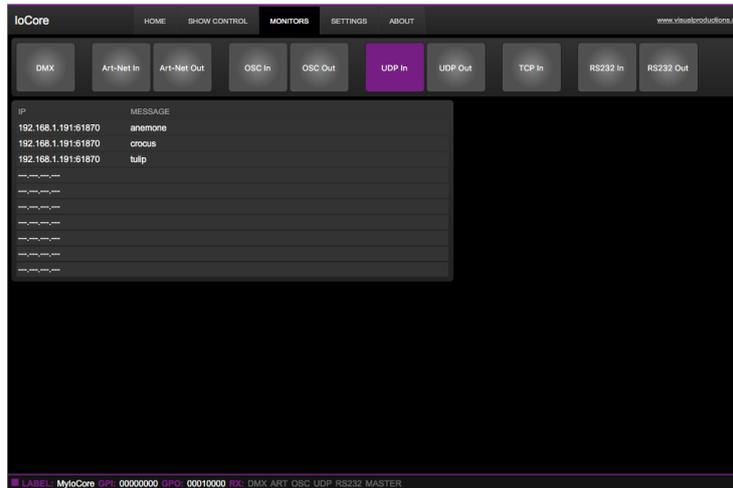


Figure 7.2: UDP Monitor page

Chapter 8

Settings

The IoCore1's settings are organised into sections, see the Settings page figure 8.1. This chapter will discuss each section.

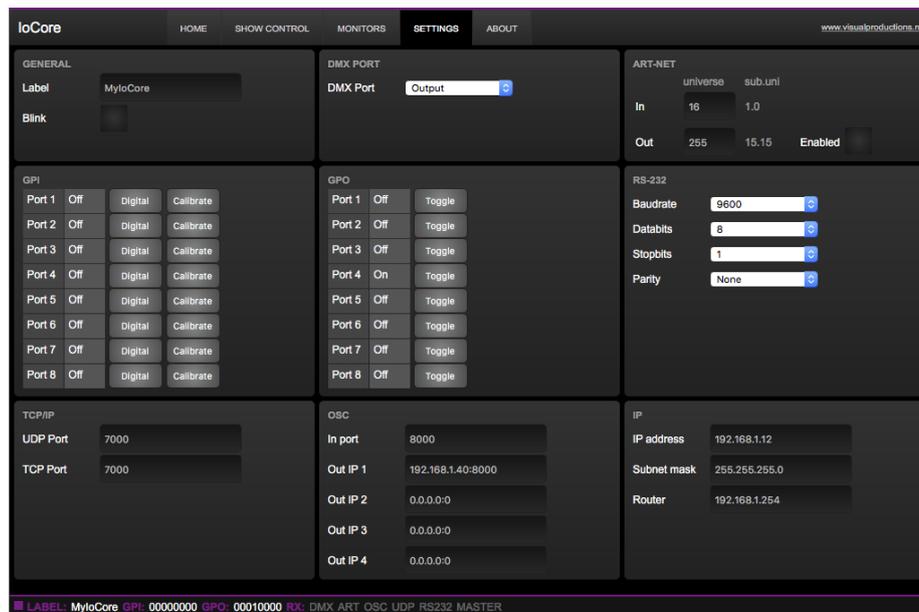


Figure 8.1: Settings page

8.1 General

You can change the IoCore1's label. This label can be used to distinguish the unit in a set-up with multiple devices. By enabling the 'Blink' checkbox the device's LED will blink to help to identify it amongst multiple devices.

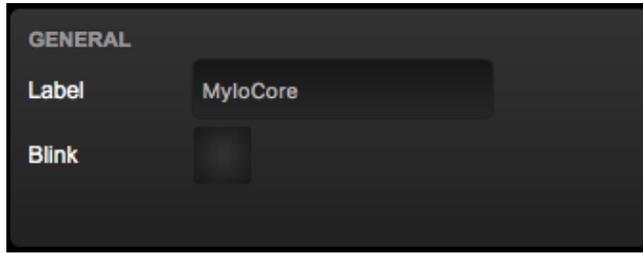


Figure 8.2: General Settings

8.2 DMX

This section allow for changing the direction of the DMX port.

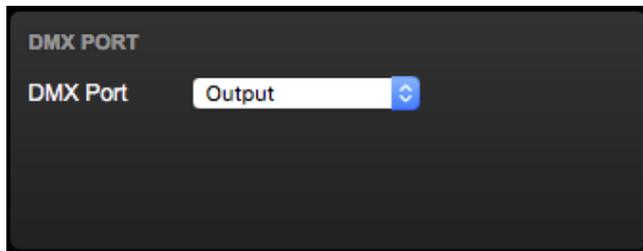


Figure 8.3: DMX Settings

8.3 Art-Net

The IoCore1 supports sending out 1 universe or and receiving 1 universe of Art-Net. These universes can be mapped to any of the 256 available universes in the Art-Net protocol. The universe is entered in the 'subnet.universe' format, i.e. the lowest universe number is written as '0.0' and the highest universe number is denoted as '15.15'. The outgoing Art-Net transmission can be disabled by disabling the checkbox.

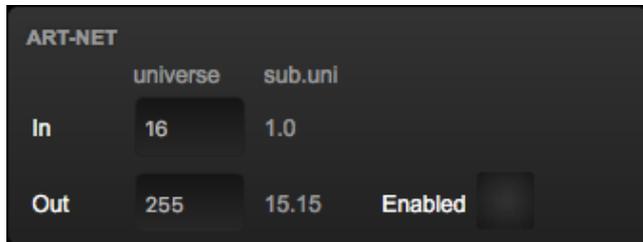


Figure 8.4: Art-Net settings

The vManager software also allows you to change the Art-Net destination IP. The destination IP determines where the outgoing Art-Net data will be send to.

Typically, this field contains a broadcast address like 2.255.255.255 which will send the Art-Net data to the 2.x.x.x IP range. Another typical Art-Net broadcast address is 10.255.255.255. When using broadcast address 255.255.255.255 then all the devices on the network will receive the Art-Net data.

It is also possible to fill in a unicast address like 192.168.1.11; in this case the Art-Net data will be send to one IP address only. This keeps the rest of the network clean of any Art-Net network messages.

8.4 GPI

The GPI ports can be configured either as digital (contact-closure) or as analog (0-10V) input.

In case a port is set as analog and the signal supplied falls within the 0-10V range (e.g. 1-5V) then it is recommended to calibrate to port in order to map the minimal voltage to a logical 0% and map the maximum voltage to 100%.

To perform a calibration first enable the Calibrate button, then bring the incoming signal to the minimum level, bring the signal to the maximum level and then close the calibration process by clicking the Calibrate button again.

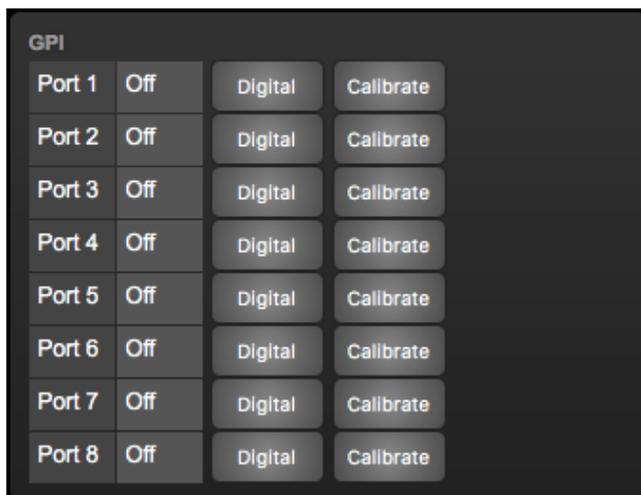


Figure 8.5: GPI Settings

Caution: Applying a voltage >10V could damage the IoCore.

8.5 GPO

This section shows the status of the GPO relays and provides buttons to switch them directly.

8.6 RS-232

Settings for RS-232 communication like baudrate and stopbits.

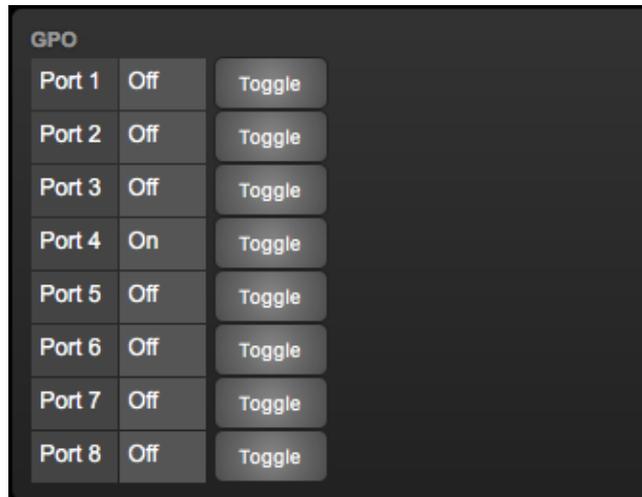


Figure 8.6: GPO Settings

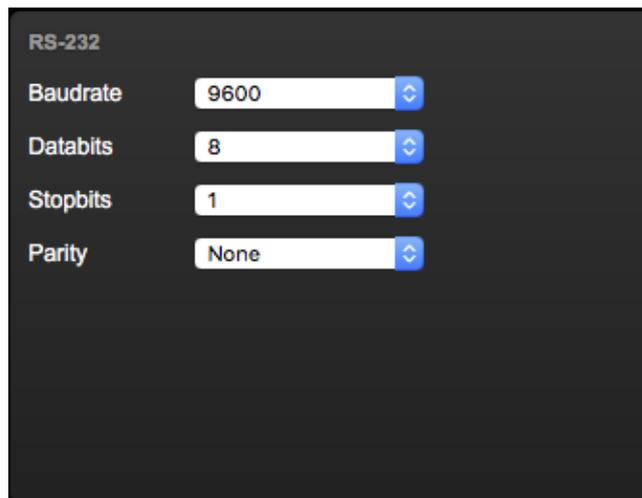


Figure 8.7: RS-232 Settings

8.7 TCP/IP

Defines the listening ports for TCP and UDP messages. External system intending to send TCP or UDP message to the IoCore1 should need to know the unit's IP address and this port number. By default both ports are set to 7000.

8.8 OSC

External equipment sending OSC messages to the IoCore1 need to be aware of the number specified in the 'Port' field. This is the port the IoCore1 listens to for incoming messages.

| TCP/IP | |
|----------|------|
| UDP Port | 7000 |
| TCP Port | 7000 |

Figure 8.8: TCP/IP settings

| OSC | |
|----------|-------------------|
| In port | 8000 |
| Out IP 1 | 192.168.1.40:8000 |
| Out IP 2 | 0.0.0.0:0 |
| Out IP 3 | 0.0.0.0:0 |
| Out IP 4 | 0.0.0.0:0 |

Figure 8.9: OSC Settings

The IoCore1 will send its outgoing OSC messages to the IP addresses specified in the 'Out IP' fields. Up to four IPs can be specified here. Use the 'ipaddress:port' format in these fields, e.g. "192.168.1.11:9000". If a field should not be used that it can be filled with IP 0.0.0.0:0. It is possible to enter a broadcast IP address like 192.168.1.255 in order to reach more than four recipients.

8.9 IP

The IP fields are for setting up the IP address and subnet mask of the IoCore1. The 'Router' field is only required when Port Forwarding is used.

IP

| | |
|-------------|---------------|
| IP address | 192.168.1.12 |
| Subnet mask | 255.255.255.0 |
| Router | 192.168.1.254 |

Figure 8.10: IP Settings

Chapter 9

vManager

A free-of-charge software tool called vManager has been developed to manage the devices. vManager allows for:

- Setup the IP address, subnet mask, router and DHCP
- Backup and restore the device's internal data and settings
- Perform firmware updates
- Set the real-time clock of the IoCore1 (The computer's date and time will be used)
- Identify a specific device (in a multi device set-up) by blinking its LED
- Revert to factory defaults

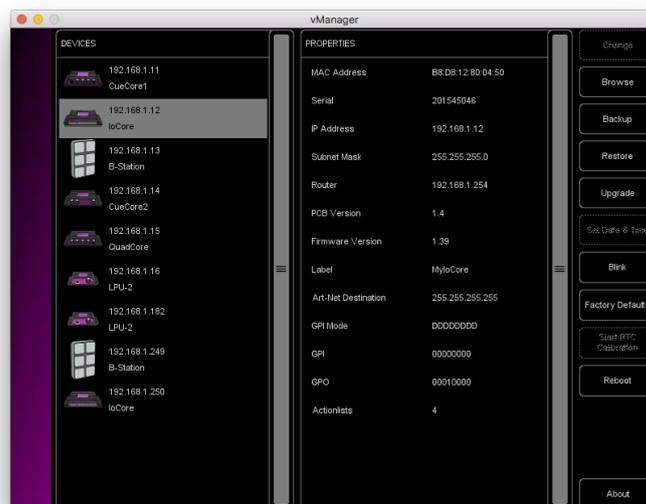


Figure 9.1: vManager

The following section explain the buttons in the vManger, as seen in figure 9.1.

9.1 Backup

Backups of all the programming data inside the device can be made. This backup file (an XML) is saved on the computer's hard-disk and can be easily transferred via e-mail or USB stick. The data of the backup can be restored via the Restore button.

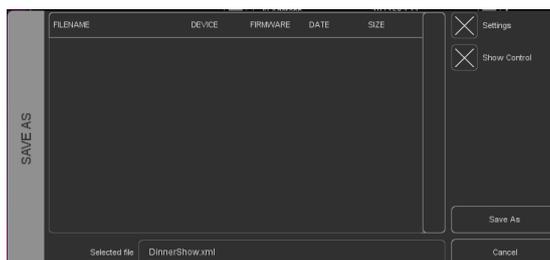


Figure 9.2: Creating a backup

The backup files created by vManager can be found at the following locations:

| | | |
|-------------------|--|---|
| Microsoft Windows | | \Users\[username]\Documents\Visual Productions\Common\Backups |
| macOS | | /Users/[username]/Visual Productions/Common/Backups |
| Ubuntu Linux | | /home/[username]/Visual Productions/Common/Backups |

9.2 Upgrade Firmware

To upgrade the firmware, first select the device and press the Upgrade Firmware button. The dialogue allows for selecting from the list of firmware versions available.



Figure 9.3: Firmware upgrade

Warning: Make sure the power to the device is not interrupted during the upgrade process.

9.3 Set Date & Time

The computer's date and time can be quickly copied to the IoCore1 by selecting a device and clicking the Set Date & Time button. Not all Visual Productions devices feature an internal real-time clock.

9.4 Blink

The device's LED can be set to blink fast for identifying the particular unit amongst multiple devices. The blinking is enabled by double-clicking on a device in the Devices list or by selecting a device and then clicking the Blink button.

9.5 Factory Defaults

All the user data like cues, tracks and actions are stored on the memory. They will be completely erased and all settings will be reverted to their defaults by pressing the Factory Defaults button. This action does not affect the device's IP settings.

9.6 RTC Calibration

The IoCore1 features an internal real-time clock (RTC) that is used for generating scheduler triggers (date, time, sunrise, etc). In case that the clock is drifting, i.e. slowly falling behind or getting ahead of the real time, then it can be calibrated again using the vManager. The procedure is as follows:

1. Select the IoCore1
2. Click on the 'Start RTC Calibration' button
3. wait approximately 30 minutes
4. Click on the 'Stop RTC Calibration' button
5. Apply the recommended calibration value in the dialog (figure 9.4)

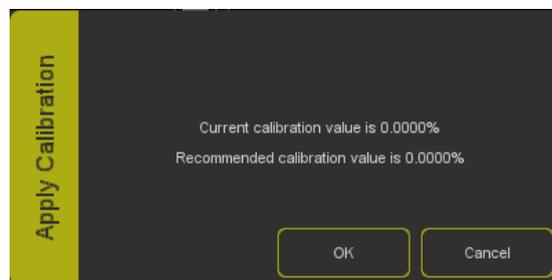


Figure 9.4: Apply calibration value

9.7 Reboot

The Reboot button allows you to remotely restart the device. This is useful for testing the unit's behaviour after a power-cycle.

9.8 Installing vManager

vManager is available on:

- Windows 7, 8 & 10
- macOS 10.11 or higher
- Ubuntu Linux 16.04 amd64
- iOS
- Android 5.0 or higher

Please download the vManager from <http://www.visualproductions.nl/downloads> or obtain it via the appropriate appstore.

Chapter 10

Kiosc

Kiosc is a simple app for creating custom user-interface. This app (see figure 10.1) is available on many operating systems: (iOS, Android, Windows, macOS & Ubuntu Linux). It can remote control lighting controllers from Visual Productions such as CueluxPro, CueCore, IoCore and the B-Station.

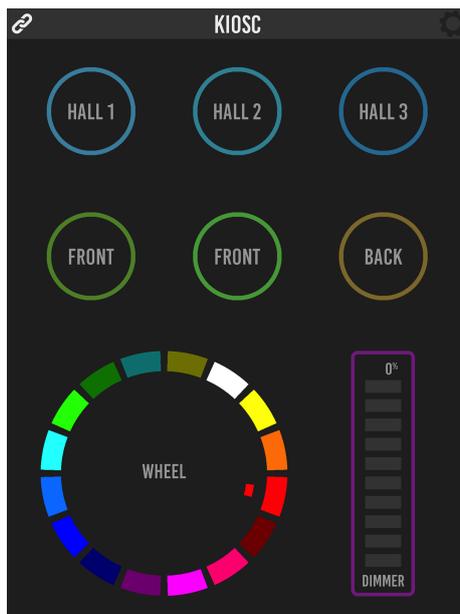


Figure 10.1: Kiosc

10.1 Connecting

The Kiosc app connects to a host (e.g. CueluxPro or CueCore) by Ethernet. Make sure all devices are within the same IP range. The Kiosc app is able to discover the hosts on the network, as seen in figure 10.2.

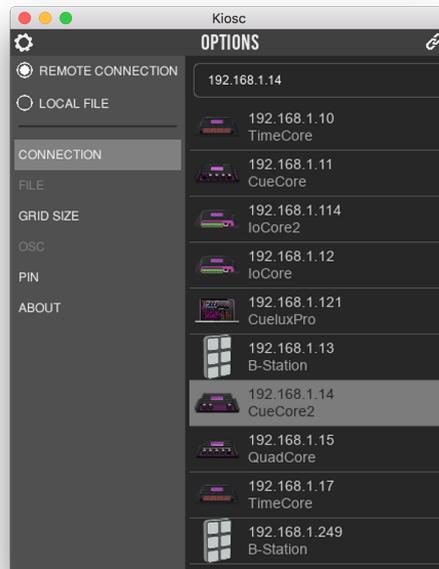


Figure 10.2: Connecting to a host

10.2 Designing the layout

The app features control elements like buttons, sliders, colour-pickers, clocks, etc. The layout of Kiosc is programmed by adding an actionlist with source 'Touch Screen'. This actionlist allows the control elements to be inserted. The order of the elements determines the location on the Kiosc screen. Also, the Kiosc app has a option that allows you to set the 'grid size' of the screen (see figure 10.3).

10.3 Password protect

Kiosc can be used to provide a fool-proof user-interface to people that should choose lighting presets without being able to alter settings in the system. For this purpose a password PIN code can be enabled. This setting is found in Kiosc's settings page, see figure 10.4.

10.4 Requirements

The minimal system requirements for Kiosc are:

- iOS 8.0 or higher
- Android 5.0 or higher
- Windows 10

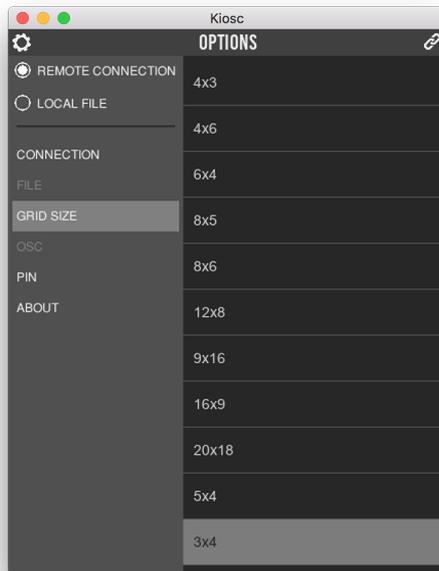


Figure 10.3: Choosing in Grid size

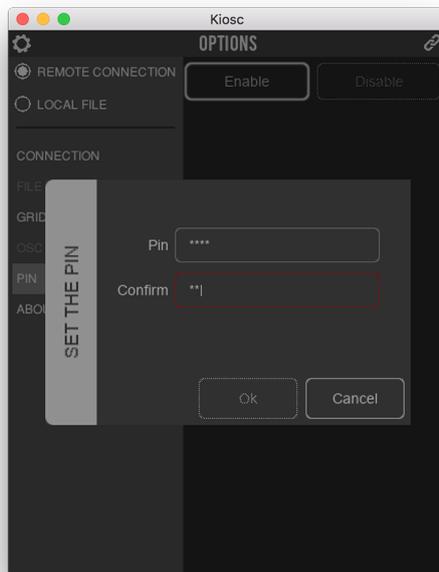


Figure 10.4: Enabling the PIN

- macOS 10.11 or higher
- Ubuntu Linux 16.04 (amd64)

10.5 Installing Kiosc

You can obtain Kiosc via the iOS Appstore, Google Playstore, Windows Appstore and macOS Appstore

On Ubuntu, you can acquire the Kiosc app from the uApp Explorer or install it by using the command-line:

```
snap find kiosc  
snap install kiosc
```

Appendices

Appendix A

Trigger Types

The following tables list the different types of triggers that can be used in the IoCore1. The different types are accompanied with values and flanks.

A.1 Touch Screen

| Trigger Type | Trigger Value | Flank | Description |
|--------------|---------------|--------|------------------------------|
| - | - | Change | Button/Fader goes up or down |
| - | - | Down | Button is pressed |
| - | - | Up | Button is released |

When editing the Touch Screen actionlist it will be possible to add different kind of actions such as Button, Fader and Colour Picker. These elements will be displayed in the VisualTouch software which is available free-of-charge from the Visual Productions website.

A.2 Art-Net

| Trigger Type | Trigger Value | Flank | Description |
|--------------|---------------|--------|---|
| Channel | DMX address | Change | Channel changes |
| Channel | DMX address | Down | Channel becomes non-zero |
| Channel | DMX address | Up | Channel becomes zero |
| UniverseA | - | - | A DMX level change in the universe |
| Receiving | - | Change | Start receiving or loose Art-Net signal |
| Receiving | - | Down | Lost Art-Net signal |
| Receiving | - | Up | Start receiving Art-Net signal |

A.3 DMX In

| Trigger Type | Trigger Value | Flank | Description |
|--------------|---------------|--------|---|
| Channel | DMX address | Change | Channel changes |
| Channel | DMX address | Down | Channel becomes non-zero |
| Channel | DMX address | Up | Channel becomes zero |
| UniverseA | - | - | A DMX level change in the universe |
| Receiving | - | Change | Start receiving or loose Art-Net signal |
| Receiving | - | Down | Lost Art-Net signal |
| Receiving | - | Up | Start receiving Art-Net signal |

A.4 System

| Trigger Type | Trigger Value | Flank | Description |
|--------------------|---------------|--------|---|
| Startup | - | - | The IoCore1 has been power up |
| Network Connection | - | Change | Network connection established or lost |
| Network Connection | - | Stop | Network connection lost |
| Network Connection | - | Start | Network connection established |
| ReleasedByMaster | - | Change | Master (e.g. CueluxPro) released or obtained connection |
| ReleasedByMaster | - | Stop | Master released connection |
| ReleasedByMaster | - | Start | Master obtained connection |

A.5 OSC

| Trigger Type | Trigger Value | Flank | Description |
|--------------|---------------|--------|---|
| Message | URI | Change | Receive a message that matches the URI |
| Message | URI | Down | Receive a message that matches the URI and the value non-zero |
| Message | URI | Up | Receive a message that matches the URI and the value is zero |
| Receiving | - | - | Receive any message |

The user can define his own URI as the trigger value of a message, however, the OSC specification dictate this string must start with a '/' sign. Please note that this string has a maximum length of 25 characters, including the '/'.

A.6 GPI

| Trigger Type | Trigger Value | Flank | Description |
|-----------------|-------------------|--------|---|
| Channel | Port number | Change | Port state changes |
| Channel | Port number | Down | Port is closed |
| Channel | Port number | Up | Port is opened |
| Channel 0-19% | Port number | Change | Analog level enters or leaves the range |
| Channel 0-19% | Port number | Down | Analog level enters the range |
| Channel 0-19% | Port number | Up | Analog level leaves the range |
| Channel 20-39% | Port number | Change | Analog level enters or leaves the range |
| Channel 20-39% | Port number | Down | Analog level enters the range |
| Channel 20-39% | Port number | Up | Analog level leaves the range |
| Channel 40-59% | Port number | Change | Analog level enters or leaves the range |
| Channel 40-59% | Port number | Down | Analog level enters the range |
| Channel 40-59% | Port number | Up | Analog level leaves the range |
| Channel 60-79% | Port number | Change | Analog level enters or leaves the range |
| Channel 60-79% | Port number | Down | Analog level enters the range |
| Channel 60-79% | Port number | Up | Analog level leaves the range |
| Channel 80-100% | Port number | Change | Analog level enters or leaves the range |
| Channel 80-100% | Port number | Down | Analog level enters the range |
| Channel 80-100% | Port number | Up | Analog level leaves the range |
| Binary | Combination value | - | A combination of ports being closed |

Use the Binary trigger type to catch port combinations when they are set as digital. Specify the port combination by adding the values that correspond to the ports (see table below). This value is entered as the trigger value.

| Port | Value |
|------|-------|
| 1 | 1 |
| 2 | 2 |
| 3 | 4 |
| 4 | 8 |
| 5 | 16 |
| 6 | 32 |
| 7 | 64 |
| 8 | 128 |

For example, to trigger on both port 1 and 2 being closed fill in trigger value 3 (1+2). To trigger on port 5 and 6, fill in value 48 (16+32).

A.7 GPO

| Trigger Type | Trigger Value | Flank | Description |
|--------------|-------------------|--------|-------------------------------------|
| Channel | Port number | Change | Port state changes |
| Channel | Port number | Down | Port is closed |
| Channel | Port number | Up | Port is opened |
| Binary | Combination value | - | A combination of ports being closed |

Use the Binary trigger type to catch port combinations. Specify the port combination by adding the values that correspond to the ports (see table below). This value is entered as the trigger value.

| Port | Value |
|------|-------|
| 1 | 1 |
| 2 | 2 |
| 3 | 4 |
| 4 | 8 |
| 5 | 16 |
| 6 | 32 |
| 7 | 64 |
| 8 | 128 |

For example, to trigger on both port 1 and 2 being closed fill in trigger value 3 (1+2). To trigger on port 5 and 6, fill in value 48 (16+32).

A.8 UDP

| Trigger Type | Trigger Value | Flank | Description |
|--------------|---------------|-------|--|
| Message | String | - | Receive a message that matches the trigger-value |
| Receiving | - | - | Receive any message |

The user can define his own string as the trigger value of a message. Please note that this string has a maximum length of 25 characters.

It is possible to pass a parameter along with a message. In order to do this use the syntax `trigger=value`. For example when the trigger type is set to 'message' and the trigger value is set to `intensity` then the transmitting equipment can pass an intensity level by sending `intensity=255`, where 255 can be any number in the range [0,255].

A.9 RS-232

| Trigger Type | Trigger Value | Flank | Description |
|--------------|---------------|-------|--|
| Message | String | - | Receive a message that matches the trigger-value |
| Receiving | - | - | Receive any message |

The user can define his own string as the trigger value of a message. Please note that this string has a maximum length of 25 characters.

It is possible to pass a parameter along with a message. In order to do this use the syntax `trigger=value`. For example when the trigger type is set to 'message' and the trigger value is set to `intensity` then the transmitting equipment can pass an intensity level by sending `intensity=255`, where 255 can be any number in the range [0,255].

A.10 Variable

| Trigger Type | Trigger Value | Flank | Description |
|--------------|----------------|--------|--|
| Channel | Variable Index | - | The specified variable changes |
| Variable 1 | Number [0,255] | Change | Variable 1 becomes = or # to the value |
| Variable 1 | Number [0,255] | Down | Variable 1 becomes = to the value |
| Variable 1 | Number [0,255] | Up | Variable 1 becomes # to the value |
| Variable 2 | Number [0,255] | Change | Variable 2 becomes = or # to the value |
| Variable 2 | Number [0,255] | Down | Variable 2 becomes = to the value |
| Variable 2 | Number [0,255] | Up | Variable 2 becomes # to the value |
| Variable 3 | Number [0,255] | Change | Variable 3 becomes = or # to the value |
| Variable 3 | Number [0,255] | Down | Variable 3 becomes = to the value |
| Variable 3 | Number [0,255] | Up | Variable 3 becomes # to the value |
| Variable 4 | Number [0,255] | Change | Variable 4 becomes = or # to the value |
| Variable 4 | Number [0,255] | Down | Variable 4 becomes = to the value |
| Variable 4 | Number [0,255] | Up | Variable 4 becomes # to the value |
| Variable 5 | Number [0,255] | Change | Variable 5 becomes = or # to the value |
| Variable 5 | Number [0,255] | Down | Variable 5 becomes = to the value |
| Variable 5 | Number [0,255] | Up | Variable 5 becomes # to the value |
| Variable 6 | Number [0,255] | Change | Variable 6 becomes = or # to the value |
| Variable 6 | Number [0,255] | Down | Variable 6 becomes = to the value |
| Variable 6 | Number [0,255] | Up | Variable 6 becomes # to the value |
| Variable 7 | Number [0,255] | Change | Variable 7 becomes = or # to the value |
| Variable 7 | Number [0,255] | Down | Variable 7 becomes = to the value |
| Variable 7 | Number [0,255] | Up | Variable 7 becomes # to the value |
| Variable 8 | Number [0,255] | Change | Variable 8 becomes = or # to the value |
| Variable 8 | Number [0,255] | Down | Variable 8 becomes = to the value |
| Variable 8 | Number [0,255] | Up | Variable 8 becomes # to the value |

A.11 Randomizer

| Trigger Type | Trigger Value | Flank | Description |
|----------------|--------------------------------|-------|--|
| Result | - | - | The Randomizer made a new value |
| Specific Value | Number in the range of [0,255] | - | The Randomizer made a value that matches |

A.12 User List (1-4)

User lists have no triggers. Actions inside user lists can only be activated by other actions through 'Action' task with the 'Link' feature.

A.13 Timer

| Trigger Type | Trigger Value | Flank | Description |
|--------------|---------------|--------|---------------------------|
| - | Timer Index | Change | The timer starts or stops |
| - | Timer Index | Stop | The timer stops |
| - | Timer Index | Start | The timer starts |

A.14 TCP

| Trigger Type | Trigger Value | Flank | Description |
|--------------|---------------|-------|--|
| Message | String | - | Receive a message that matches the trigger-value |
| Receiving | - | - | Receive any message |

The user can define his own string as the trigger value of a message. Please note that this string has a maximum length of 25 characters.

Appendix B

Task Types

Tasks allow you to automate the functionality in the IoCore1. All this functionality is categorized in task-types. This appendix provides a listing of the various task-types. The tables present an overview of all available features and functions per task-type.

B.1 GPO

Control the GPO relays.

| Feature | Function | Parameter 1 | Parameter 2 |
|-------------|----------|-------------|-------------|
| Set Value | Set | Port [1,8] | On/Off |
| Set Value | Toggle | Port [1,8] | - |
| Set Value | Control | Port [1,8] | - |
| Set Value | Pulse | Port [1,8] | - |
| Solo | Set | Port [1,8] | - |
| Solo | Control | Port [1,8] | - |
| Random Solo | Set | - | - |
| Clear All | Set | - | - |

B.2 GPI

Force the GPI actions to be triggered.

| Feature | Function | Parameter 1 | Parameter 2 |
|---------------|----------|-------------|-------------|
| Sample Binary | Set | - | - |
| Refresh | Set | - | - |

B.3 DMX

Manipulate the DMX levels. These are the levels that can also be send out via Art-Net.

| Feature | Function | Parameter 1 | Parameter 2 |
|--------------|------------------|-----------------|----------------------|
| Set Value | Set | Channel [1,512] | Value [0,255] |
| Set Value | Toggle | Channel [1,512] | - |
| Set Value | Control | Channel [1,512] | - |
| Set Value | Inverted Control | Channel [1,512] | - |
| Set Value | Decrement | Channel [1,512] | - |
| Set Value | Increment | Channel [1,512] | - |
| Set Value | Universe A | - | - |
| Set Value | Switch | DMX Channel | - |
| Set Value | Control scaled | Channel [1,512] | Percentage [0%,100%] |
| Set Value | Control offset | Channel [1,512] | Offset [0,255] |
| Bump Channel | Set | Channel [1,512] | Value [0,255] |
| Bump Channel | Control | Channel [1,512] | - |
| Clear All | Set | - | - |
| Spectrum | Control | Channel [1,510] | - |

B.4 RS-232

Send an message via the RS-232 port. Specify the port settings (baudrate, stop-bits, etc) in the settings page.

| Feature | Function | Parameter 1 | Parameter 2 |
|-----------------|----------|-----------------------|-------------|
| Send | Set | text string | - |
| Send | Control | - | - |
| Send Float | Set | floating point number | - |
| Send Float | Control | - | - |
| Send Unsigned | Set | positive number | - |
| Send Unsigned | Control | - | - |
| Send Bool | Set | On or Off | - |
| Send Bool | Control | - | - |
| Send String | Set | text string | - |
| Send String | Control | - | - |
| Send String Hex | Set | hex string | - |
| Send String Hex | Control | String | - |
| Send Bytes | Set | Hex string | - |
| Send Bytes | Control | - | - |

Please note that string in parameter 1 has a maximum length of 25 characters.

The Send Bytes features allows for sending ASCII codes. For example, in order to send the string 'Visual' followed by a line feed parameter 1 should be '56697375616C0A'.

B.5 OSC

Send an OSC message via the network. The OSC recipients are specified in the Settings page.

| Feature | Function | Parameter 1 | Parameter 2 |
|---------------|----------|-------------|-----------------------|
| Send | Set | URI | floating point number |
| Send | Control | URI | - |
| Send Float | Set | URI | floating point number |
| Send Float | Control | URI | - |
| Send Unsigned | Set | URI | positive number |
| Send Unsigned | Control | URI | - |
| Send Bool | Set | URI | true or false |
| Send Bool | Control | URI | - |
| Send String | Set | URI | String of characters |
| Send String | Control | URI | - |
| Colour | Set | URI | RGB colour |
| Colour | Control | URI | - |

Please note that string in parameter 1 has a maximum length of 25 characters, including the compulsory leading '/' sign.

B.6 UDP

Send an UDP message via the network. Specify the recipient in Parameter 2. For example "192.168.1.11:7000".

| Feature | Function | Parameter 1 | Parameter 2 |
|-----------------|----------|-----------------------|-------------------|
| Send | Set | text string | IP address & port |
| Send | Control | - | IP address & port |
| Send Float | Set | floating point number | IP address & port |
| Send Float | Control | - | IP address & port |
| Send Unsigned | Set | positive number | IP address & port |
| Send Unsigned | Control | - | IP address & port |
| Send Bool | Set | true or false | IP address & port |
| Send Bool | Control | - | IP address & port |
| Send String | Set | text string | IP address & port |
| Send String | Control | - | IP address & port |
| Send String Hex | Set | hex string | IP address & port |
| Send String Hex | Control | String | IP address & port |
| Send Bytes | Set | Hex string | IP address & port |

Please note that string in parameter 1 has a maximum length of 25 characters.

The Send Bytes features allows for sending ASCII codes. For example, in order to send the string 'Visual' followed by a line feed parameter 1 should be '56697375616C0A'.

B.7 System

Miscellaneous tasks.

| Feature | Function | Parameter 1 | Parameter 2 |
|------------------------------|----------|-------------|-------------|
| Blink | Set | On or Off | - |
| Blink | Toggle | - | - |
| Blink | Control | - | - |
| Show Control Routing Art-Net | Set | On or Off | - |
| Show Control Routing Art-Net | Toggle | - | - |
| Show Control Routing Art-Net | Control | - | - |

B.8 Action

Trigger another action.

| Feature | Function | Parameter 1 | Parameter 2 |
|---------|----------|-------------|-------------|
| Link | Set | Action | - |

B.9 Action List

Manipulate an action-list.

| Feature | Function | Parameter 1 | Parameter 2 |
|---------|------------------|-------------|-------------|
| Enable | Set | Action-list | On or Off |
| Enable | Toggle | Action-list | - |
| Enable | Control | Action-list | - |
| Enable | Inverted Control | Action-list | - |

B.10 Variable

Manipulate one of the eight variables.

| Feature | Function | Parameter 1 | Parameter 2 |
|-----------|----------------------|----------------|----------------------|
| Set Value | Set | Variable [1,8] | Value [0,255] |
| Set Value | Toggle | Variable [1,8] | Value [0,255] |
| Set Value | Control | Variable [1,8] | - |
| Set Value | Inverted Control | Variable [1,8] | - |
| Set Value | Decrement | Variable [1,8] | - |
| Set Value | Increment | Variable [1,8] | - |
| Set Value | Continuous Decrement | Variable [1,8] | Delta [1,255] |
| Set Value | Continuous Increment | Variable [1,8] | Delta [1,255] |
| Set Value | Stop Continuous | Variable [1,8] | - |
| Set Value | Control Scaled | Variable [1,8] | Percentage [0%,100%] |
| Set Value | Control Offset | Variable [1,8] | Offset [0,255] |
| Refresh | Set | Variable [1,8] | - |

ables are further explained on page 31

B.11 Randomiser

Trigger the Randomizer to generate a new random number.

| Feature | Function | Parameter 1 | Parameter 2 |
|---------|----------|---------------|---------------|
| Refresh | Set | Minimum value | Maximum value |

B.12 Timer

Manipulate on of the four internal timers.

| Feature | Function | Parameter 1 | Parameter 2 |
|-----------|----------|-------------|-------------|
| Playstate | Start | Timer # | - |
| Playstate | Stop | Timer # | - |
| Playstate | Restart | Timer # | - |
| Time | Set | Timer # | Time |

Appendix C

Templates

This appendix discusses the templates provided in the Show Control page.

| Template | Description |
|--------------------|--|
| Art-Net ->DMX | Converts the incoming Art-Net universe to DMX. The Art-Net universe number is determined on the settings page. The direction of the DMX port should be set to output. |
| Art-Net ->GPO | Control the GPO relays by Art-Net (channel 1-8). The Art-Net universe number is determined on the settings page. |
| DMX ->Art-Net | Converts incoming DMX to Art-Net. Art-Net should be enabled and the Art-Net universe number chosen on the settings page. The direction of the DMX port should be set to input. |
| DMX ->GPO | Control the GPO relays by DMX (channel 1-8). The direction of the DMX port should be set to input. |
| GPI ->DMX | Each GPI port controls a DMX channel (1-8). The direction of the DMX port should be set to output. |
| GPI ->GPO | Each GPI port controls a GPO port. |
| GPI ->OSC Float | Each GPI port converts to a OSC message using a floating point parameter. This template is intended for GPI set to analog. Specify the OSC target IPs on the settings page. |
| GPI ->OSC Bool | Each GPI port converts to a OSC message using a boolean parameter. This template is intended for GPI set to digital. Specify the OSC target IPs on the settings page. |
| OSC ->GPO | The GPO ports are controlled by OSC messages. |
| Touch Screen ->GPO | Creates a VisualTouch layout for controlling the GPO ports. |

Appendix D

API

The IoCore1 is pre-programmed to make its internal functionality available via OSC and UDP. There is a simple API implemented for each protocol. Notwithstanding these API's, it is possible to create your own OSC and UDP implementation in the Show Control page.

D.1 OSC

The following table uses GPO #1 as an example. The number '1' can be replaced by any number in the range of [1,6].

| URI | Parameter | Description |
|-------------|-----------|---------------------------------|
| /core/gpo/1 | bool | Switch the relay on GPO port #1 |

The following table uses actionlist #1 as an example. The number '1' can be replaced by any number in the range of [1,8]. The table also uses action #2 as an example. The number '1' can be replaced by any number in the range of [1,48].

| URI | Parameter | Description |
|----------------------|--------------------|--|
| /core/al/1/2/execute | bool/float/integer | Execute action #2 inside action list #1 |
| /core/al/1/enable | bool | Set the 'enable' checkbox for action list #1 |

The following table shows how to directly control the DMX values. The DMX port needs to be switched to output. The table uses dmx channel #1 as an example. The number '1' can be replaced by any number in the range of [1,512].

| URI | Parameter | Description |
|-------------|-----------|--------------------------------|
| /core/dmx/1 | integer | Set the value of DMX channel 1 |

The following table shows how to active miscellaneous functions.

| URI | Parameter | Description |
|-------------|-----------|---------------------------------------|
| /core/blink | - | Momentarily flashes the IoCore1's LED |

D.2 UDP

The following table uses GPO #1 as an example. The number '1' can be replaced by any number in the range of [1,6].

| String | Description |
|------------------|---------------------------------|
| core-gpo-1=<0/1> | Switch the relay on GPO port #1 |

The following table uses actionlist #1 as an example. The number '1' can be replaced by any number in the range of [1,8]. The table also uses action #2 as an example. The number '1' can be replaced by any number in the range of [1,48].

| String | Description |
|---------------------------|--|
| core-al-1-1-execute=<arg> | Execute action #2 inside action list #1 |
| core-al-1-enable=<bool> | Set the 'enable' checkbox for action list #1 |

The following table shows how to directly control the DMX values. The DMX port needs to be switched to output. The table uses dmx channel #1 as an example. The number '1' can be replaced by any number in the range of [1,512].

| String | Description |
|----------------------|---------------------------------|
| core-dmx-1=<integer> | Set the value of DMX channel #1 |

The following table shows how to active miscellaneous functions.

| String | Description |
|------------|---------------------------------------|
| core-blink | Momentarily flashes the IoCore1's LED |

Index

- vManager, 40
 - 19 rack, 22
- Action, 27, 29
- Action list, 28
- Actionlist, 28
- API, 63
- Art-Net, 11, 35
- Auto-sensing, 24
- Backup, 41
- Baudrate, 36
- Blink, 42, 60
- Box contents, 7
- Broadcast, 36
- Calibrate, 36
- Control, 30
- Date, 42
- Destination IP, 35
- DIN Rail, 22
- DMX, 35
- DMX-512, 10
- Execute, 30
- Factory Defaults, 42
- Feature, 56
- Firmware, 41
- Flank, 29, 49
- Forum, 8
- Function, 56
- General, 34
- GPI, 9, 36, 56
- GPO, 10, 36, 56
- IP, 38
- IP Address, 25
- Kensington, 23
- Kiosk, 29, 30, 44
- Memory, 8
- Monitor, 32
- OSC, 12, 37, 50, 58
- Password, 45
- PIN, 45
- PoE, 23
- Port Forwarding, 38
- Port-Forwarding, 25
- Quickstart, 13
- Rackmount, 22
- Randomizer, 31, 55, 61
- RdmSplitter, 11
- Reboot, 43
- Relay, 36
- Reset button, 25
- Router, 38
- RS-232, 10, 36, 53, 57
- RTC Calibration, 42
- Settings, 34
- Show Control, 27
- Source, 28
- Splitter, 11
- Startup, 50
- Subnet mask, 25
- System, 50
- Task, 28, 30
- Task type, 56
- TCP, 12, 37, 55
- TCP/IP, 37
- Template, 30, 62
- Templates, 62
- Time, 42
- Timer, 55, 61
- Touch Screen, 49

Trigger, 28
Trigger Type, 49
Trigger type, 49
Tutorial, 13

UDP, 12, 37, 52, 59
Unicast, 36
URI, 50
User List, 55

Variable, 54
Variables, 31, 60
VisualTouch, 49
VPN, 25