IOCORE2 MANUAL

© VISUAL PRODUCTIONS BV

WWW.VISUALPRODUCTIONS.NL

Revision History

Revision	Date	Author(s)	Description
5	25.07.2023	ME	Updated relay power rating.
4	04.07.2023	ME	Added Timer and Userlist examples. Updated screen shots. Inserted FCC declaration.
3	07.06.2018	ME	Updated vManager chapter to reflect app-store distribution. Moved majority of Kiosc information to a dedicated Kiosc manual. Added discussion on password and share analytics.
2	19.10.2017	ME	Rackmount adapter added. Quick-start system diagrams improved.
1	04.09.2017	ME	Initial version.

Contents

1 Introduction 7

2 Protocols 11

3 Quickstart 15

4 Installation 23

5 Network 27

6 Show Control 31

7 Monitors 38

8 Settings 40

9 vManager 47

10 Kiosc 51

Appendices 53

A Trigger Types 54

B Task Types 62

C Templates 68

D API 70

©2023 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, herby declare under sole responsibility, that the following device:

IoCore2

Conforms to the following EC Directives, including all amendments:
EMC Directive 2014/30/EU

And the following harmonized standards have been applied:
NEN-EN-IEC 61000-6-1:2019

The object of the declaration is in conformity with the relevant Union harmonisation Legislation.

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

Date: November 24th, 2022	Place: Haarlem, The Netherlands

ing. Maarten Engels
Managing Director
Visual Productions BV

SUPPLIER'S DECLARATION OF CONFORMITY (No. 2023/202-02)

This Declaration of Conformity is issued under the sole responsibility of the manufacturer

MANUFACTURER

Company name	Visual Productions BV
Full address	Izaak Enschedeweg 38A 2031 CR Haarlem
Country	The Netherlands

RESPONSIBLE PARTY – U.S. CONTACT INFORMATION

Company name	ACT Entertainment
Full address	3581 Larch Lane Jackson, MO 63755
Country	United States of America
Contact details (Phone)	+ 1 800 255 9822

DESCRIPTION AND IDENTIFICATION OF THE EQUIPMENT

Generic denomination	Lighting control system
Function/intended use	Architectural lighting controller
Models	CueCore2, QuadCore, IoCore2, TimeCore, LPU-1, LPU-2

The object of the Declaration described above is in conformity with all relevant provisions of:

FCC (47 CFR Part 15B)

Supplementary information

This equipment has been tested and found to comply with the limits for a Class B digital device, pursuant to part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation. This equipment generates, uses and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation.

Signed for and on behalf of:

Place of issue:	Haarlem The Netherlands	**Identity:** **Function:**	Maarten Engels Managing Director
Date of issue:	13-06-2023	**Signature:**	

Chapter 1

Introduction

Thank you for choosing the IoCore2. The IoCore2 is expansion module for the CueCore and QuadCore range of solid-state lighting controllers. The IoCore2 can add extra GPI, GPO, RS-232 and DMX ports to a CueCore based installation. The IoCore2 is the successor of the original IoCore1.

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 IoCore. 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.

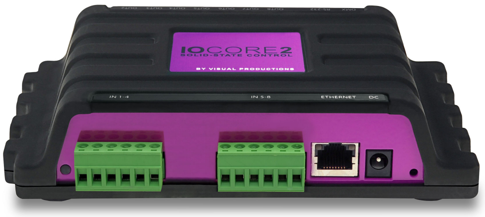


Figure 1.1: IoCore2

This manual discusses setting up and programming the unit. Chapter 2 provides background information on the communication protocols used the IoCore2. Chapters **??** 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 31.

At the time of writing this manual the IoCore2's firmware was at version 1.06.

1.1 Compliance

This device is in compliance with CE, UKCA and FCC regulations.

1.2 Feature comparison

The following table visualises the difference between the IoCore2 and IoCore1. This overview might prove to be helpful to IoCore users considering choosing the model for their new designs.

	IoCore2	IoCore1
CPU	180MHz	120MHz
Flash memory	16MB	8MB
GPI	8x digital/analog	8x digital/analog
GPO	8x relay	8x relay
RS-232	yes	yes
DMX	bi-directional	bi-directional
Art-Net	input+output	input+output
sACN	input+output	-
TCP	input	input
UDP	input+output	input+output
OSC	input+output	input+output
POE	class I	class I
DC	12-24V	12V
DHCP	yes	-
CueluxPro Licence	1 universe	-
Kiosc compatible	yes	yes

1.3 What's in the box?

The IoCore2 packaging contains the following items (see figure 1.2):

- IoCore2
- Power supply (including international plug set)

- Network cable
- 16-pin + 3x 6-pin terminals
- Info card

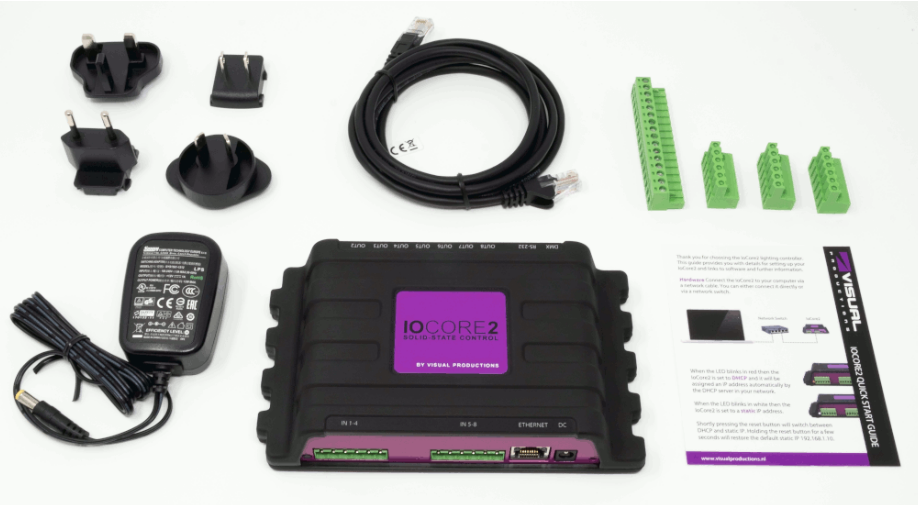


Figure 1.2: IoCore2 box contents

1.4 Saving data to memory

This manual will describe how to configure the IoCore2 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 IoCore2 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 IoCore2 will copy any changes in the RAM memory to its onboard flash memory. Flash memory retains its data even when not powered. The IoCore2 will load all its data back from the flash memory upon startup.

This memory copy process is conducted automatically by the IoCore2 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.

1.5 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 IoCore2 is fitted with several communication ports and supports various protocols. This chapter describes these protocols and to which extent they are implemented in the IoCore2

2.1 GPI

The IoCore2 features eight General Purpose Input (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 IoCore2.

Each GPI port can be switched between *Digital* and *Analog*. Set to *Digital* the port works as a contact-closure. In *Analog* mode the port is a 0-10V level input.

2.2 GPO

Each GPO port features a normally-open (NO) relay for switching external equipment. The relay can switch 50VAC/30VDC up to 3A. Please refer to figure 2.1 for the pinout of the GPO connector.

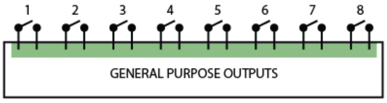


Figure 2.1: GPO Pinout

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 2.2 shows the pinout of the connector which shared with the DMX port. Please refer to page 42 for more information on configuring the RS-232 port.

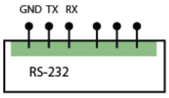


Figure 2.2: RS-232 Pinout

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 2.3: Visual Productions' RdmSplitter

The DIN Rail RdmSplitter from Visual Productions (See figure 2.3) 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.

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 IoCore2 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 sACN

The streaming Architecture of Control Networks (sACN) protocol uses a method of transporting DMX-512 information over TCP/IP networks. The protocol is specified in the ANSI E1.31-2009 standard.

The sACN protocol supports multi-cast in order to take efficient use of the network's bandwidth.

The IoCore2 supports sending and receiving of one sACN universe.

2.7 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 IoCore2 supports receiving TCP messages.

2.8 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, therefor it is faster than TCP but less reliable.

There are two ways how to have the IoCore2 respond to incoming UDP messages. The API (see page 71) makes typical IoCore2 functions available through UDP. Furthermore, custom messages can be programmed in the Show Control page (see page 31). This is also the place where to program outgoing UDP messages.

2.9 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 various 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 IoCore2 respond to incoming OSC messages. Firstly, the API (see page 70) makes typical IoCore2 functions available through OSC. Secondly, custom messages can be programmed in the Show Control page (see page 31).

2.10 DHCP

The Dynamic Host Configuration Protocol (DHCP) is a standardised network protocol used on Internet Protocol (IP) networks for dynamically distributing network configuration parameters, such as IP addresses.

The IoCore2 is a DHCP client.

Chapter 3

Quickstart

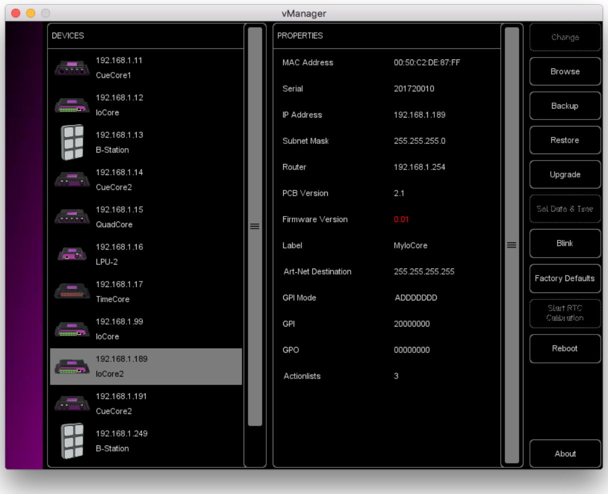
This chapter provides step by step tutorials on how to program your IoCore2 for some typical tasks:

- 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 IoCore2 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. Connect to the network
   Connect the IoCore2 with an Ethernet cable to the router. It is required that the network is managed by a router that features a DHCP server. If the network router is not DHCP capable then read the network chapter on page 27 for alternative setups.



vManager

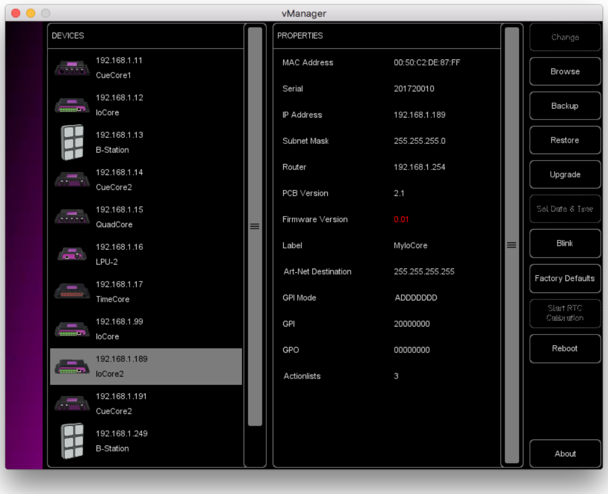
DEVICES

192.168.1.11	CueCore1
192.168.1.12	IoCore
192.168.1.13	B-Station
192.168.1.14	CueCore2
192.168.1.15	QuadCore
192.168.1.16	LPU-2
192.168.1.17	TimeCore
192.168.1.99	IoCore
192.168.1.189	IoCore2
192.168.1.191	CueCore2
192.168.1.249	B-Station

PROPERTIES

MAC Address	00:50:C2:DE:87:FF
Serial	201720010
IP Address	192.168.1.189
Subnet Mask	255.255.255.0
Router	192.168.1.254
PCB Version	2.1
Firmware Version	0.01
Label	MyIoCore
Art-Net Destination	255.255.255.255
GPI Mode	ADDDDDDD
GPI	20000000
GPO	00000000
Actionlists	3

2. Install the vManager
   To access the web-interface of the IoCore2, the vManager tool is recommended. This tool can be downloaded from the Visual Productions website and installed. Once the installation is complete, run the vManager to discover the IP address of the IoCore2.



vManager

DEVICES

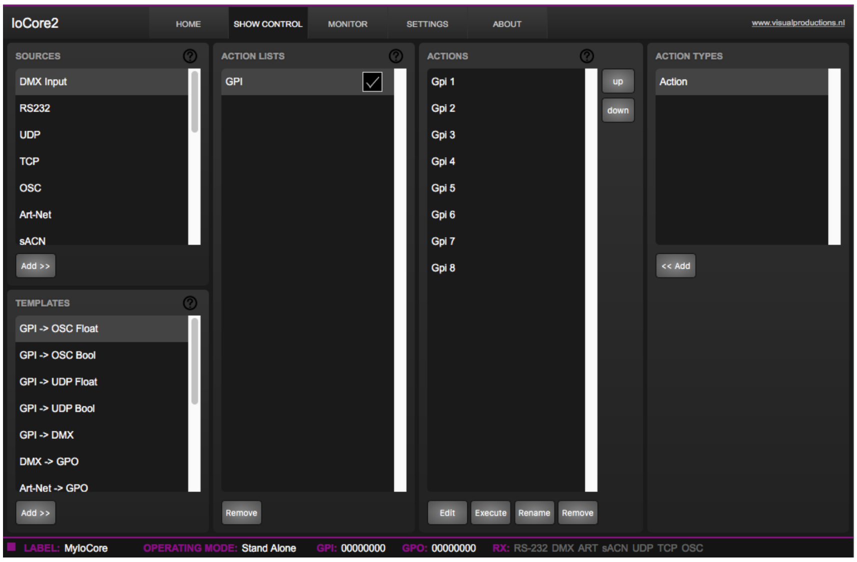
192.168.1.11	CueCore1
192.168.1.12	IoCore
192.168.1.13	B-Station
192.168.1.14	CueCore2
192.168.1.15	QuadCore
192.168.1.16	LPU-2
192.168.1.17	TimeCore
192.168.1.99	IoCore
192.168.1.189	IoCore2
192.168.1.191	CueCore2
192.168.1.249	B-Station

PROPERTIES

MAC Address	00:50:C2:DE:87:FF
Serial	201720010
IP Address	192.168.1.189
Subnet Mask	255.255.255.0
Router	192.168.1.254
PCB Version	2.1
Firmware Version	0.01
Label	MyIoCore
Art-Net Destination	255.255.255.255
GPI Mode	ADDDDDDD
GPI	20000000
GPO	00000000
Actionlists	3

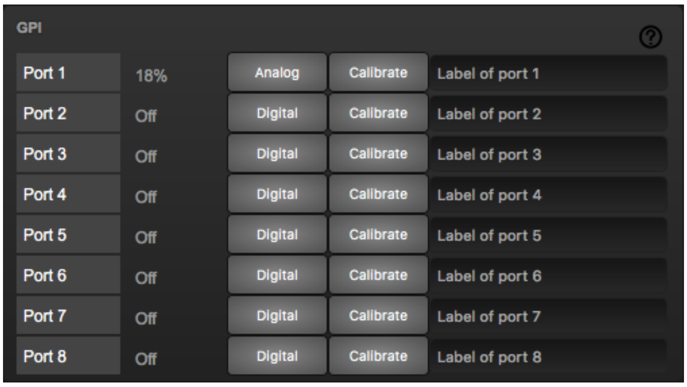
3. Open the web-interface
   Choose the IoCore2 from the device list and click on the 'Browse' button to open the web-interface.

4. Add template
   Go to the Show Control page and select the 'GPI ->DMX' template. Press the 'Add >>' button in the lower left corner of the screen.



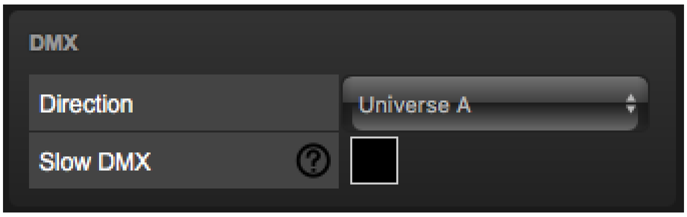
LABEL: MyloCore OPERATING MODE: Stand Alone GPI: 00000000 GPO: 00000000 RX: RS-232 DMX ART sACN UDP TCP OSC

5. Configure GPI port
   Go to the IoCore2 Settings page and set GPI port 1 to 'analog'.



Port 1	18%	Analog	Calibrate	Label of port 1
Port 2	Off	Digital	Calibrate	Label of port 2
Port 3	Off	Digital	Calibrate	Label of port 3
Port 4	Off	Digital	Calibrate	Label of port 4
Port 5	Off	Digital	Calibrate	Label of port 5
Port 6	Off	Digital	Calibrate	Label of port 6
Port 7	Off	Digital	Calibrate	Label of port 7
Port 8	Off	Digital	Calibrate	Label of port 8

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



Direction: Universe A

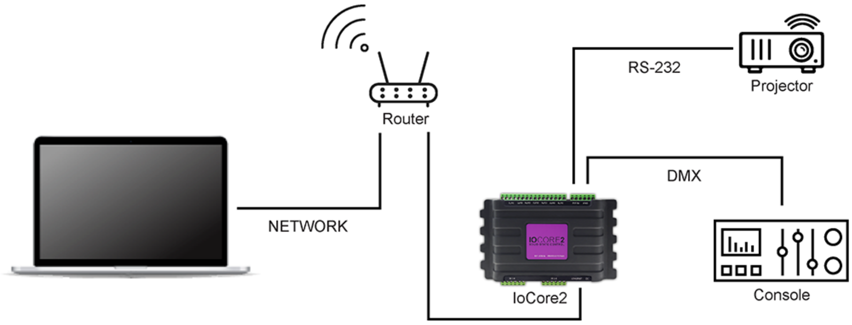
Slow DMX ? □

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

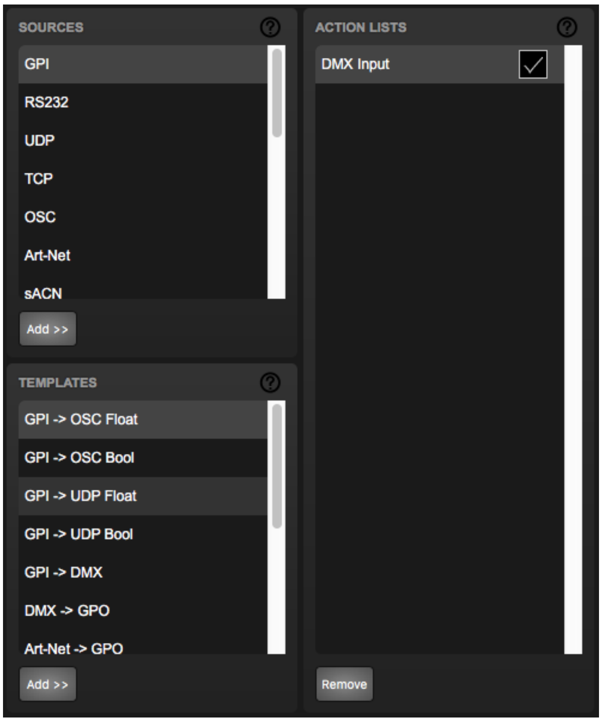
3.2 Use DMX to trigger a RS-232 lamp strike command to a video projector

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:

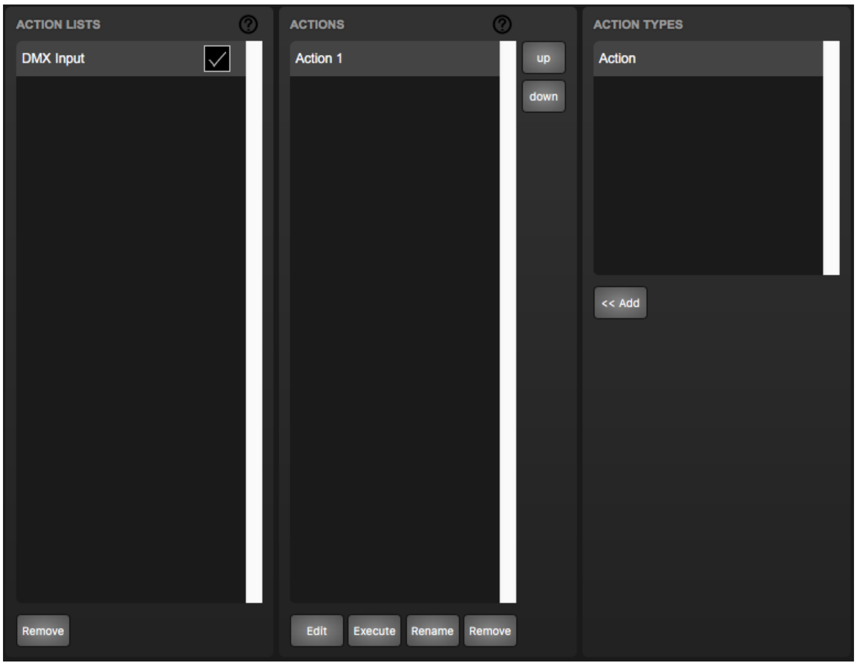
1. Setup network and connect equipment
   Connect the IoCore2 and the computer to a DHCP capable network router. Connect the the external DMX board to the DMX port. Connect the projector to the RS-232 port.



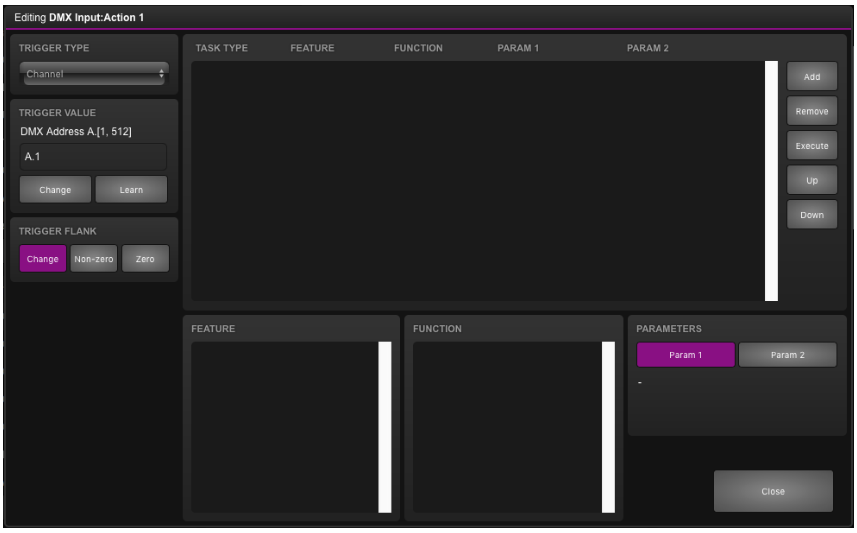
2. Add source
   Open a browser on the computer and browse to the web-interface of the IoCore2 by typing '192.168.1.10' in the browser's address field. Go to the Show Control page and select the 'DMX In' source. Press the 'Add >>' button below the table.



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



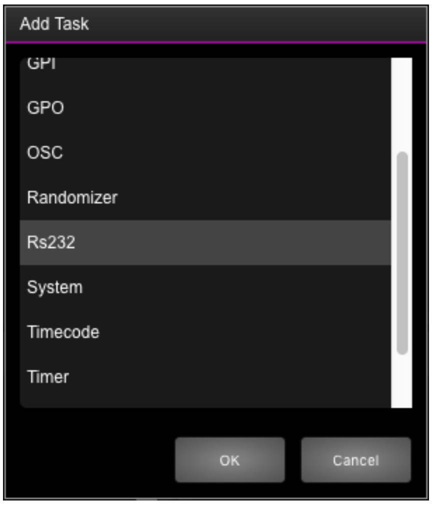
4. Edit action
   Make sure the new action is selected and press the 'Edit' button. Set the flank to 'Down'. This will make sure that the action is only triggered when DMX channel 1 goes from 0% to non-zero.


Editing **DMX Input:Action 1**

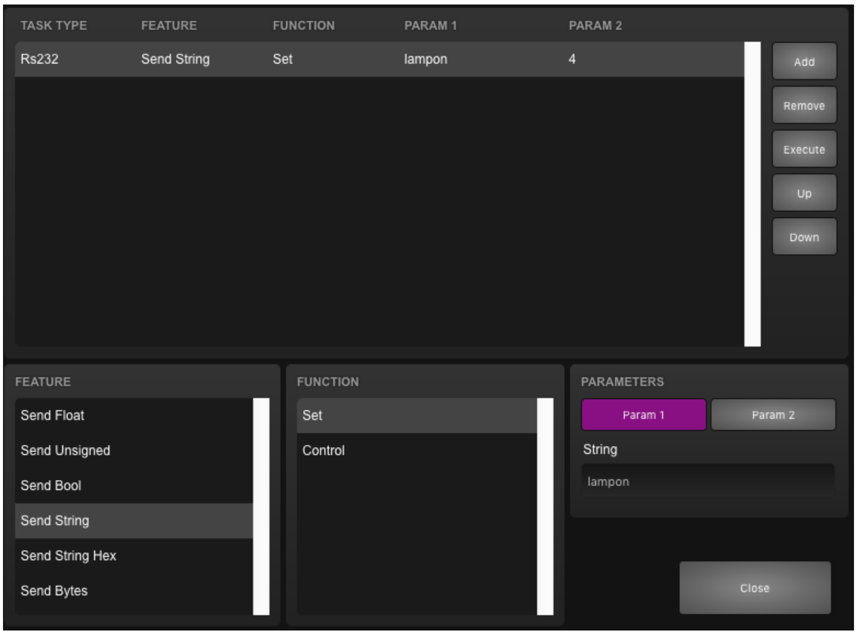
TRIGGER TYPE: Channel

TRIGGER VALUE: DMX Address A.[1, 512]: A.1

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



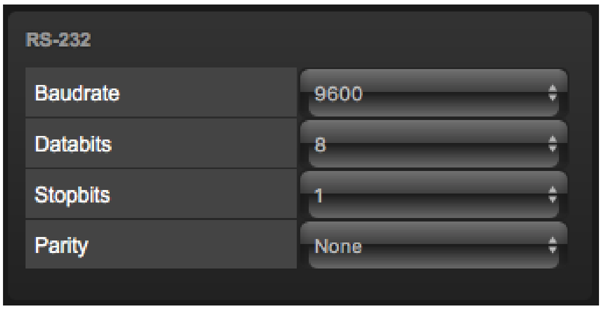
6. Edit task
   Select the task and change the feature to 'Send String'. Set the function to 'Set'. Enter e.g. 'lampon' in parameter 1. You can now exit the dialog by pressing the 'Close' button. Please consult the projector's documentation and replace the 'lampon' string accordingly.



TASK TYPE	FEATURE	FUNCTION	PARAM 1	PARAM 2
Rs232	Send String	Set	lampon	4

String: lampon

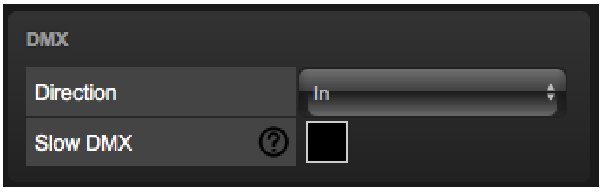
7. Configure RS-232 port
   Go to the IoCore2 Settings page. Set the RS-232 details to match specifications that would be found in the projector's documentation.



RS-232

Baudrate	9600
Databits	8
Stopbits	1
Parity	None

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



DMX

Direction	In
Slow DMX ?	□

Bringing channel 1 up from 0% will now make the IoCore2 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

Installation

This chapter discusses how to set up the IoCore2.

4.1 DIN Rail Mounting

The device 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).

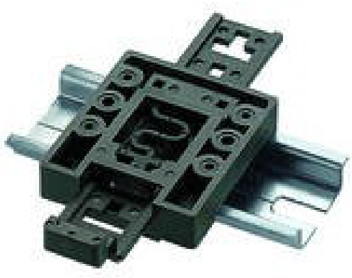


Figure 4.1: Bopla DIN rail adapter

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 IoCore2 into a 19” rack . The rackmount adapter is 1U and is sold separately. It fits two units, however, it is supplied with one position closed by a blind panel, see figure 4.2.

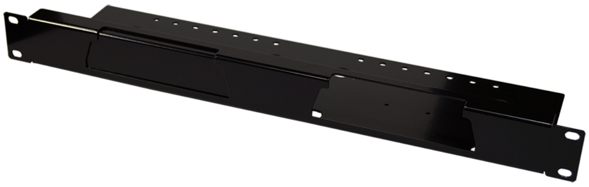


Figure 4.2: Rackmount adapter

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 between 12 and 24 Volt with a minimum of 500mA. The 2,1 mm DC connector is center-positive. The IoCore2 is also Power-over-Ethernet (PoE) enabled. It requires PoE Class I.

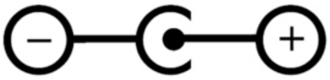


Figure 4.4: DC polarity

4.5 GPI

The IoCore2 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 IoCore2.

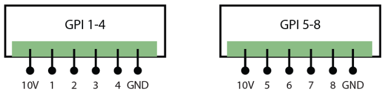
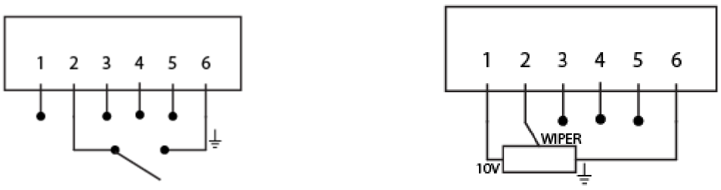


Figure 4.5: GPI Pinout

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. For convenience, a 10V supply is available on one of the pins of the GPI connector. Please refer to figure 4.5 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 4.6 shows examples of how to wire a contact-closure to a GPI port set to digital. And it shows an example of potentiometer connected to a GPI port set to analog.



(a) Contact-closure

(b) Potentiometer

Figure 4.6: GPI Wiring examples

Please refer to page 42 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 31.

The green connector used for the GPI port has part number CTB9200/6A and replacements can be ordered from Farnell/Newark (order code 3881880).

Figure 4.7: GPI Connector

4.6 DMX

The IoCore2 has one DMX port and is therefor able control 512 channels. The port can be configured either as input or output. Figure 4.8 shows the pinout of the connector.

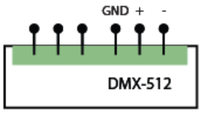


Figure 4.8: DMX Pinout

Chapter 5

Network

The IoCore2 is a network capable device. A network connection between a computer and the unit is required to configure and program the IoCore2, however, once the device is programmed then it is not necessary anymore for the IoCore2 to be connected to an Ethernet network.

There are multiple arrangements possible for connecting the computer and the IoCore2. They can be connected peer-to-peer, via a network switch or via Wi-Fi. Figure 5.1 illustrates these different arrangements.

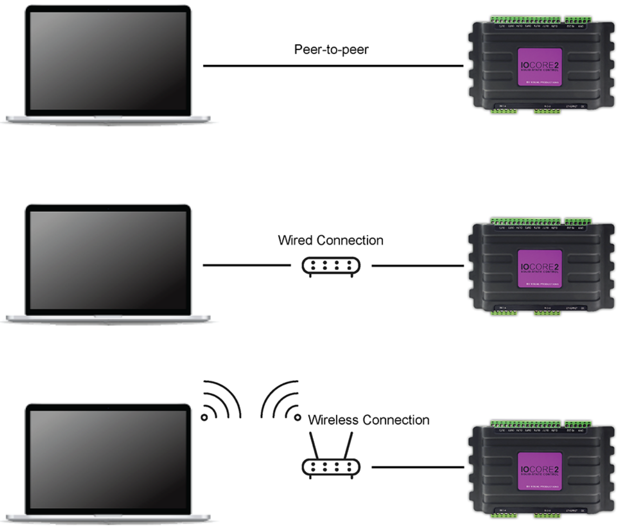


Figure 5.1: Network arrangements

The Ethernet port on the IoCore2 is auto-sensing; it does not matter whether a cross or straight network-cable is being used. Although the Ethernet port is classified as 100 Mbps, buffer limits may apply for specific tasks as API messages.

5.1 IP Address

The IoCore2 supports both static IP addresses and automatic IP addresses. By default, the IoCore2 is set DHCP in which it will be automatically assigned an IP address by the DHCP server in the network. The 'DHCP server' is typically part of the router's functionality.

Static IP addresses are useful when there is no DHCP server in the network, for instance when there is a direct peer-to-peer connection between a IoCore2 and a computer. It is also useful in permanent installations where the IP address of the IoCore2 is known by other equipment and therefor should not change. When using DHCP there is always the risk of automatically being given a new IP address in the event that the DHCP server is replaced. When using static IP addresses make sure that all equipment on the network have unique IP addresses.

The IoCore2's LED helps to determine which kind of IP address is set. The LED will indicate red when using DHCP and it will indicate white in the case of a static IP address.

There are three ways to change the IP address setting of the IoCore2.

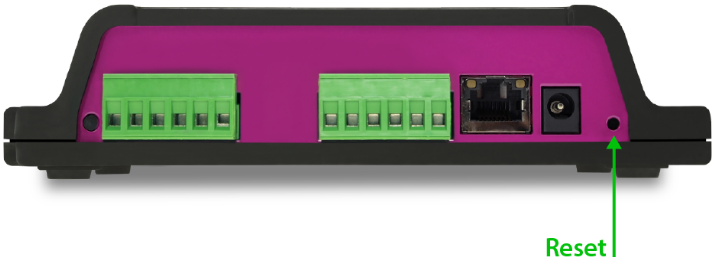


Figure 5.2: Reset button

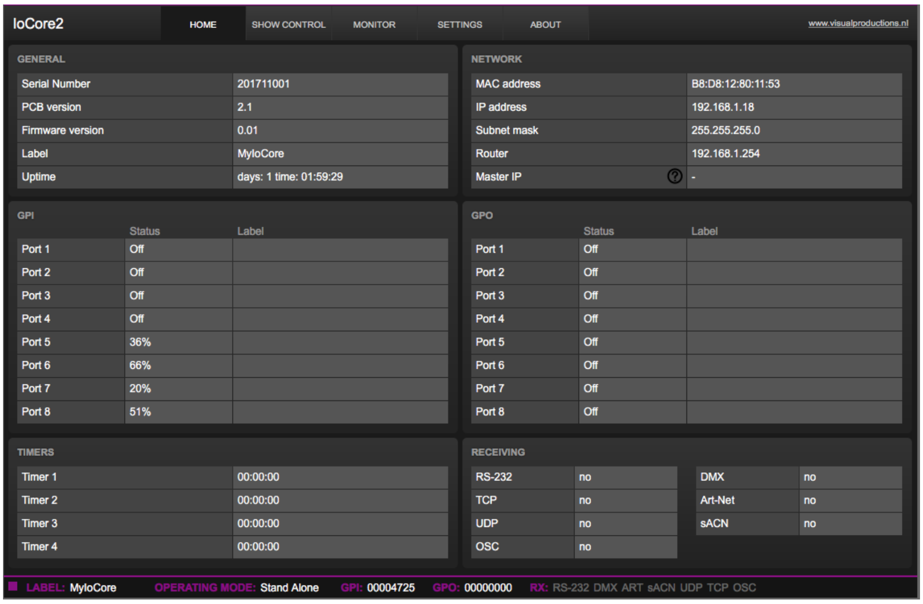
- **vManager** can be used to detect a IoCore2 on the network. Once found, the vManager software (figure chapter 9) allows for changing the IP address, subnet mask and DHCP settings.
- If the IP address is already known then browsing to this address using the computer's browser will show the IoCore2's **web-interface**. The Settings page on this web-interface enables changing the IP address, subnet mask and DHCP settings.
- By briefly pressing the **reset button** on the device it toggles between static and automatic IP addresses. By pressing and holding the reset button (see figure 5.2) 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.

5.2 Web-interface

The IoCore2 features an inbuilt web-server. This web-interface can be accessed via a standard browser. It is recommended to use any of the following browsers:

- Microsoft Edge
- Google Chrome (v102 or higher)
- Apple Safari (v15 or higher)
- Mozilla Firefox (v54 or higher)

The web-interface enables you to configure and program the IoCore2. When browsing to the unit the home page (figure 5.3) will appear first. The home page is read-only; it provides information but does not allow for changing any setting. The other pages present many settings that can be edited. These pages will be discussed in the subsequent chapters.



GENERAL

Serial Number	201711001
PCB version	2.1
Firmware version	0.01
Label	MyloCore
Uptime	days: 1 time: 01:59:29

NETWORK

MAC address	B8:D8:12:80:11:53
IP address	192.168.1.18
Subnet mask	255.255.255.0
Router	192.168.1.254
Master IP	-

GPI

	Status	Label
Port 1	Off	
Port 2	Off	
Port 3	Off	
Port 4	Off	
Port 5	36%	
Port 6	66%	
Port 7	20%	
Port 8	51%	

GPO

	Status	Label
Port 1	Off	
Port 2	Off	
Port 3	Off	
Port 4	Off	
Port 5	Off	
Port 6	Off	
Port 7	Off	
Port 8	Off	

TIMERS

Timer 1	00:00:00
Timer 2	00:00:00
Timer 3	00:00:00
Timer 4	00:00:00

RECEIVING

RS-232	no	DMX	no
TCP	no	Art-Net	no
UDP	no	sACN	no
OSC	no		

LABEL: MyloCore OPERATING MODE: Stand Alone GPI: 00004725 GPO: 00000000 RX: RS-232 DMX ART sACN UDP TCP OSC

Figure 5.3: Home page

5.2.1 Uptime

This field indicates how long the unit has been alive since its last reboot.

5.2.2 Master IP

When the unit is not in Stand Alone mode, then this field displays the IP number of system that is mastering the IoCore2. Refer to chapter **??** for more information on operating modes.

5.3 Access via Internet

The IoCore2 can be accessed through the Internet. There are two ways to achieve this: Port Forwarding and VPN.

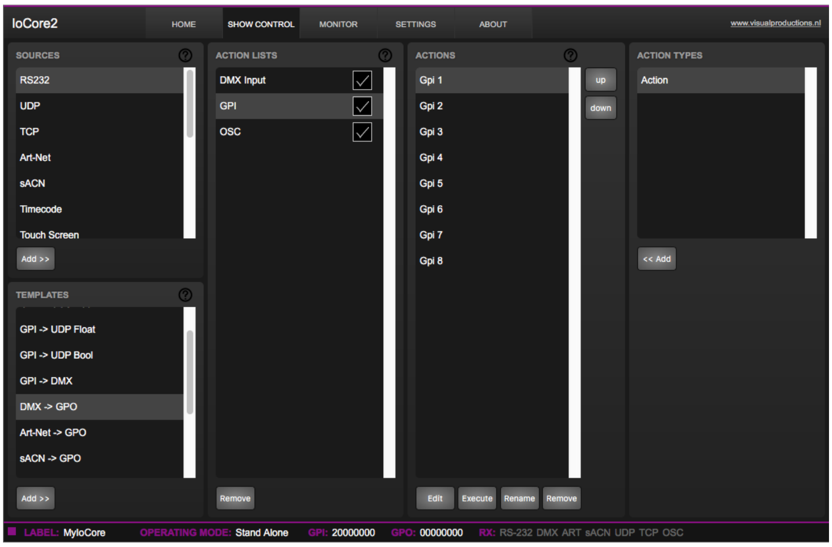
- **Port Forwarding** Is relatively easy to setup in the router. Each router is different so it is advised to consult the router's documentation (sometimes it is referred to as NAT or Port-Redirecting). Please note that port forwarding is not secure, since anybody could access the IoCore2 this way.
- 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 IoCore2. 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 IoCore2 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 IoCore2 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.

Another option is integrating the IoCore2 with other systems, by using the various protocols it can receive to trigger its functionality. The *Show Control* page (See figure 6.1) enables this kind of programming to be made.



LABEL: MyIoCore OPERATING MODE: Stand Alone GPI: 20000000 GPO: 00000000 RX: RS-232 DMX ART sACN UDP TCP OSC

Figure 6.1: Show Control page

The *Show Control* page presents a system of *actions*. A signal that the IoCore2 should to respond to or perhaps convert into some other signal, needs to be expressed in an action. Before programming actions please consider the show control structure in figure 6.2.

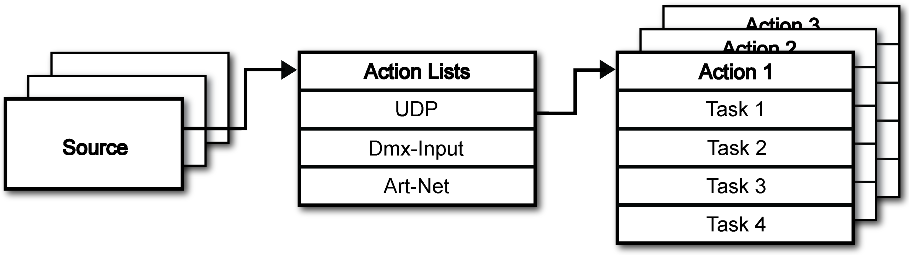


Figure 6.2: Show Control structure

The IoCore2 is capable of listening to various protocols. These available protocols are listed in *Sources*, however, the IoCore2 can only actively listen to 8 protocols at once. The active protocols are listed in *Actionlists*. Each actionlist 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 actionlist 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 IoCore2 will do when this action is triggered. Several tasks can be placed in the action. There are tasks available for a wide range of IoCore2 features and external protocols. Task types are detailed in Appendix B on page 62.

Please consult the API appendix on page 70 before implementing incoming OSC or UDP messages; the API already exposes typical functionality through OSC and UDP and therefor it might not be necessary to implement custom messages.

6.1 Sources and Actionlists

The Sources listing presents all protocols that the IoCore2 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
GPI	Changes on one of the GPI ports
GPO	Changes on one of the GPO ports
DMX Input	Receiving DMX data when the DMX port is set as input
RS-232	Receiving RS-232 serial messages
UDP	UDP network messages
TCP	TCP network messages
OSC	OSC network message
Art-Net	Art-Net DMX data
sACN	sACN DMX data
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.
Randomiser	The randomiser can generate a random number
System	Events such as 'Power on'
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.
Timer	Internal timers expiring
Actionlist	Actionlist's enable-checkboxes changing state
User List 1-4	These action-lists will never trigger an event, however, they are useful for advanced programming.

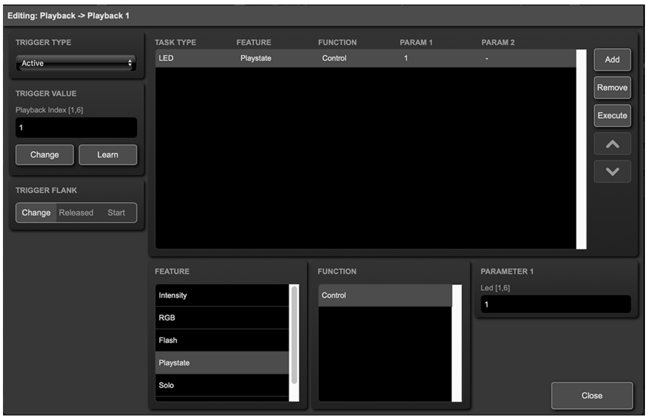
Action-lists can be temporarily suspended by disabling their checkbox in the Show Control page. There is also a task available to automate changing the state of this checkbox.

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 actionlist 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 actionlist.

Figure 6.3 shows the screen when editing an action.

A trigger is determined by the trigger-type, trigger-value and trigger-flank fields.



Editing: Playback -> Playback 1

TRIGGER TYPE: Active

TRIGGER VALUE

Playback Index [1,6]: 1

TRIGGER FLANK: Change

TASK TYPE	FEATURE	FUNCTION	PARAM 1	PARAM 2
LED	Playstate	Control	1	-

FEATURE: Playstate

FUNCTION: Control

PARAMETER 1

Led [1,6]: 1

Figure 6.3: Edit Action dialog

Although these fields are not applicable for all actionlists and are therefor 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 actionlists 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 the Kiosc list, 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 actionlist 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 playbacks and recorder 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 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 *Edit Action* dialog. The complete action can also be tested; go to the *Show Control* page, select the action and press the *Execute* button.When either of these *Execute* buttons are used, the source of the *Control* value for tasks will be the *Execute* button. The result will depend on the chosen task and feature, but will most likely be 100%, 1.0, or 255 when pressed and 0%, 0.0, or 0 when depressed.

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 actionlists, actions and task. These templates configure the IoCore2 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 DMX outlets to be set to outputs in order to achieve an Art-Net to DMX conversion.

Appendix C gives an overview of the available templates.

6.5 Variables

Variables are part of the show control system in the IoCore2. There are variables and each can hold a value in the range of [0,255]. These values can be manipulated by tasks and can be used for advanced action programming. Variables can be added as sources in order to have actions triggered when a variable changes value.

You can see the status of the variables in the monitor page, as discussed on page 38.

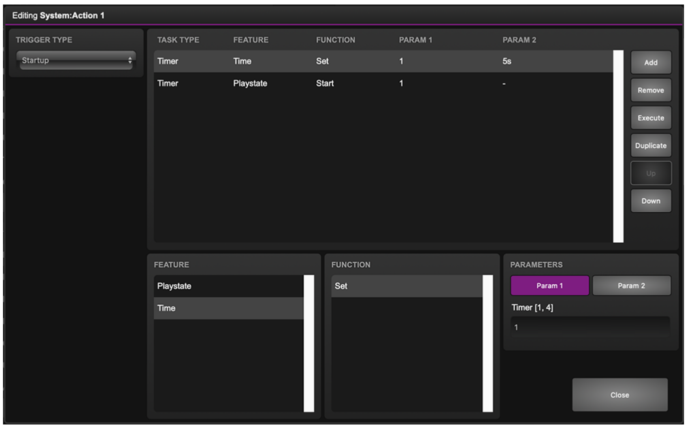
Please note that the variable values are not stored between power cycles.

6.6 Timers

The show control system of the IoCore2 features four internal timers. By using tasks, the timers can be set to certain durations and they can be started. Once started the timers will countdown to zero. When the timer reaches zero it will generate an event that can be captured by using the *Timer* actionlist. Please note that the timer values are not stored between power cycles. You can see the status of the four timers in the monitor page, under *Timers.*

The next example will show a timer being used to regularly send a UDP message to an external system in order to notify that the IoCore2 is still 'alive'.

- Setup and start the timer at power-on. This is done by creating an action in the *System* actionlist.



Editing System:Action 1

TRIGGER TYPE: Startup

TASK TYPE	FEATURE	FUNCTION	PARAM 1	PARAM 2
Timer	Time	Set	1	5s
Timer	Playstate	Start	1	-

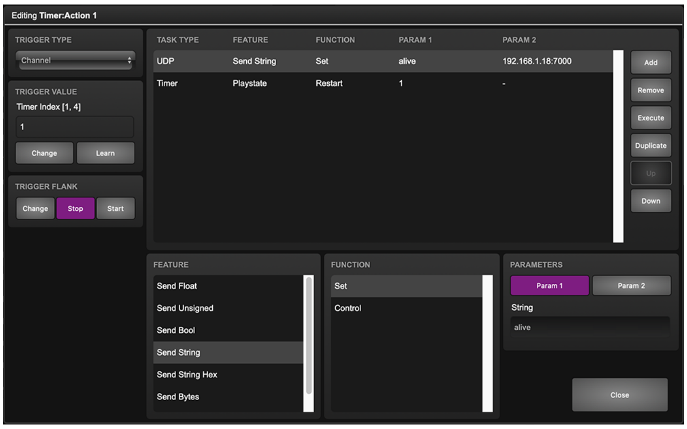
FEATURE: Time

FUNCTION: Set

PARAMETERS: Param 1

Timer [1, 4]: 1

- If the timer expires then send the UDP message and restart the timer. This is done by creating an action in the *Timer* actionlist.



Editing Timer:Action 1

TRIGGER TYPE: Channel

TRIGGER VALUE

Timer Index [1, 4]: 1

TRIGGER FLANK: Stop

TASK TYPE	FEATURE	FUNCTION	PARAM 1	PARAM 2
UDP	Send String	Set	alive	192.168.1.18:7000
Timer	Playstate	Restart	1	-

FEATURE: Send String

FUNCTION: Set

PARAMETERS: Param 1

String: alive

6.7 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.

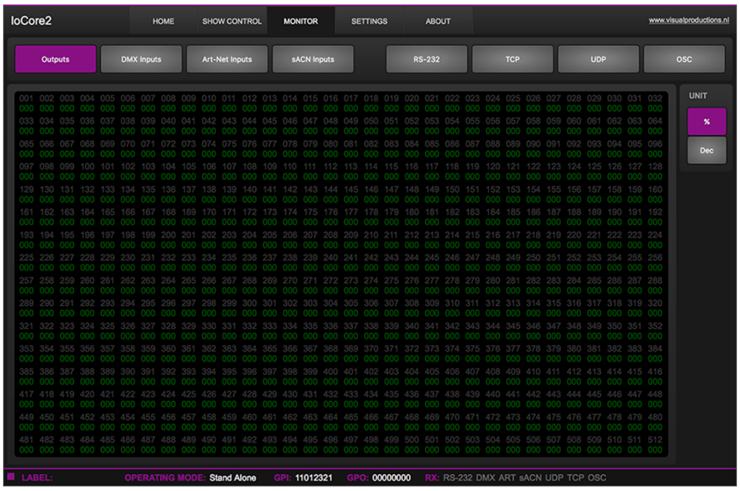
6.8 User Lists

Normally a source is connected to a communication port, protocol or software feature. Actionlists will be triggered as soon as an event happens in a source. There is an exception; the User Lists are not connected to any source and therefor will not be triggered by any communication or other event. The purpose of User Lists is to have extra actionlists containing actions which can be triggered by explicitly linking to it. See page **??** for details of the *Link* feature in the *Action* task.

Chapter 7

Monitors

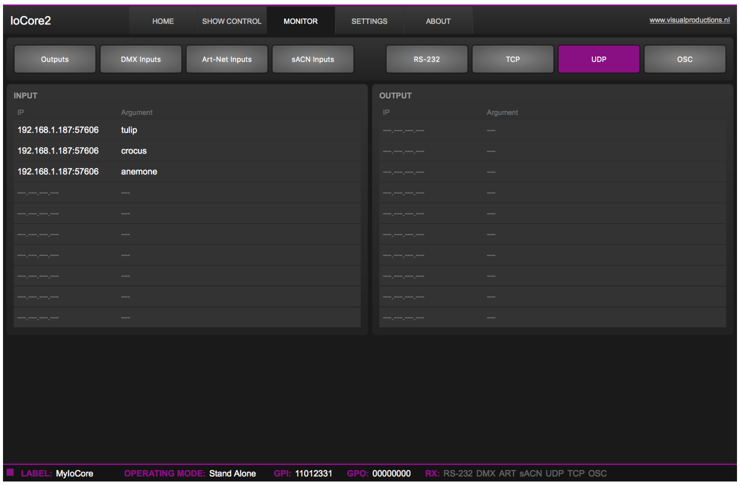
This page allows the user to inspect the incoming and outgoing data, both DMX-type data (See figure 7.1) as well as control messages (See figure 7.2). Monitoring incoming and outgoing data can help the user troubleshoot during programming.



```
001 002 003 004 005 006 007 008 009 010 011 012 013 014 015 016 017 018 019 020 021 022 023 024 025 026 027 028 029 030 031 032
000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000
033 034 035 036 037 038 039 040 041 042 043 044 045 046 047 048 049 050 051 052 053 054 055 056 057 058 059 060 061 062 063 064
000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000
065 066 067 068 069 070 071 072 073 074 075 076 077 078 079 080 081 082 083 084 085 086 087 088 089 090 091 092 093 094 095 096
000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000
097 098 099 100 101 102 103 104 105 106 107 108 109 110 111 112 113 114 115 116 117 118 119 120 121 122 123 124 125 126 127 128
000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000
129 130 131 132 133 134 135 136 137 138 139 140 141 142 143 144 145 146 147 148 149 150 151 152 153 154 155 156 157 158 159 160
000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000
161 162 163 164 165 166 167 168 169 170 171 172 173 174 175 176 177 178 179 180 181 182 183 184 185 186 187 188 189 190 191 192
000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000
193 194 195 196 197 198 199 200 201 202 203 204 205 206 207 208 209 210 211 212 213 214 215 216 217 218 219 220 221 222 223 224
000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000
225 226 227 228 229 230 231 232 233 234 235 236 237 238 239 240 241 242 243 244 245 246 247 248 249 250 251 252 253 254 255 256
000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000
257 258 259 260 261 262 263 264 265 266 267 268 269 270 271 272 273 274 275 276 277 278 279 280 281 282 283 284 285 286 287 288
000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000
289 290 291 292 293 294 295 296 297 298 299 300 301 302 303 304 305 306 307 308 309 310 311 312 313 314 315 316 317 318 319 320
000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000
321 322 323 324 325 326 327 328 329 330 331 332 333 334 335 336 337 338 339 340 341 342 343 344 345 346 347 348 349 350 351 352
000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000
353 354 355 356 357 358 359 360 361 362 363 364 365 366 367 368 369 370 371 372 373 374 375 376 377 378 379 380 381 382 383 384
000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000
385 386 387 388 389 390 391 392 393 394 395 396 397 398 399 400 401 402 403 404 405 406 407 408 409 410 411 412 413 414 415 416
000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000
417 418 419 420 421 422 423 424 425 426 427 428 429 430 431 432 433 434 435 436 437 438 439 440 441 442 443 444 445 446 447 448
000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000
449 450 451 452 453 454 455 456 457 458 459 460 461 462 463 464 465 466 467 468 469 470 471 472 473 474 475 476 477 478 479 480
000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000
481 482 483 484 485 486 487 488 489 490 491 492 493 494 495 496 497 498 499 500 501 502 503 504 505 506 507 508 509 510 511 512
000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000
```

LABEL: OPERATING MODE: Stand Alone GPI: 11012321 GPO: 00000000 RX: RS-232 DMX ART sACN UDP TCP OSC

Figure 7.1: DMX Monitor page



INPUT

IP	Argument
192.168.1.187:57606	tulip
192.168.1.187:57606	crocus
192.168.1.187:57606	anemone
---.---.---.---	---
---.---.---.---	---
---.---.---.---	---
---.---.---.---	---
---.---.---.---	---
---.---.---.---	---
---.---.---.---	---

OUTPUT

IP	Argument
---.---.---.---	---
---.---.---.---	---
---.---.---.---	---
---.---.---.---	---
---.---.---.---	---
---.---.---.---	---
---.---.---.---	---
---.---.---.---	---
---.---.---.---	---
---.---.---.---	---

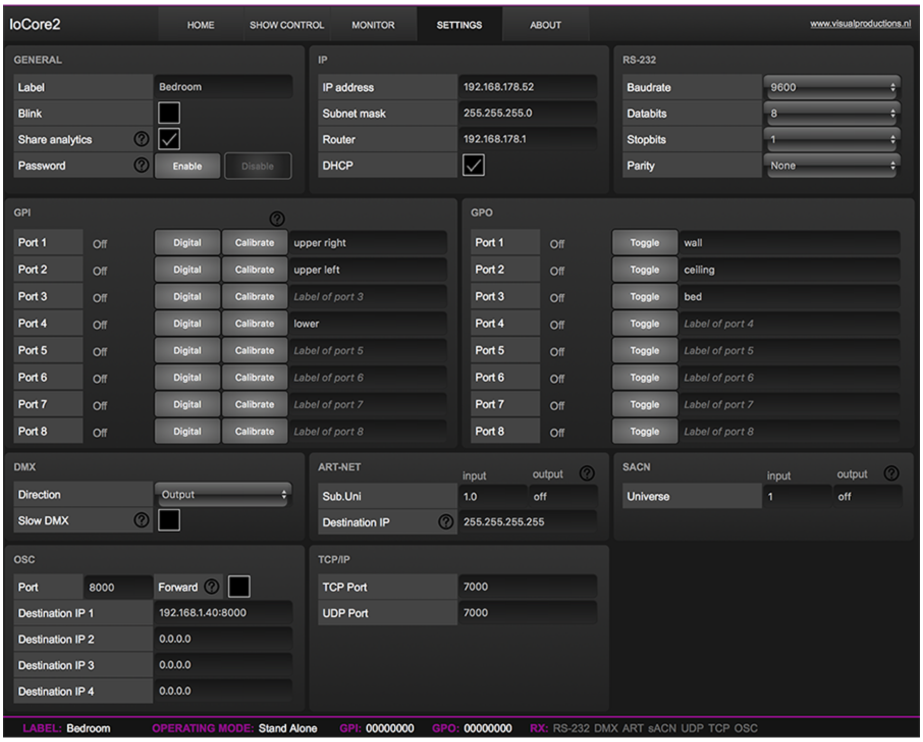
LABEL: MyIoCore OPERATING MODE: Stand Alone GPI: 11012331 GPO: 00000000 RX: RS-232 DMX ART sACN UDP TCP OSC

Figure 7.2: UDP Monitor page

Chapter 8

Settings

The IoCore2's settings are organised into sections, see the Settings page figure 8.1. This chapter will discuss each section.



GENERAL

Label	Bedroom	
Blink	■	
Share analytics	☑	
Password	Enable	Disable

IP

IP address	192.168.178.52
Subnet mask	255.255.255.0
Router	192.168.178.1
DHCP	☑

RS-232

Baudrate	9600
Databits	8
Stopbits	1
Parity	None

GPI

Port 1	Off	Digital	Calibrate	upper right
Port 2	Off	Digital	Calibrate	upper left
Port 3	Off	Digital	Calibrate	*Label of port 3*
Port 4	Off	Digital	Calibrate	lower
Port 5	Off	Digital	Calibrate	*Label of port 5*
Port 6	Off	Digital	Calibrate	*Label of port 6*
Port 7	Off	Digital	Calibrate	*Label of port 7*
Port 8	Off	Digital	Calibrate	*Label of port 8*

GPO

Port 1	Off	Toggle	wall
Port 2	Off	Toggle	ceiling
Port 3	Off	Toggle	bed
Port 4	Off	Toggle	*Label of port 4*
Port 5	Off	Toggle	*Label of port 5*
Port 6	Off	Toggle	*Label of port 6*
Port 7	Off	Toggle	*Label of port 7*
Port 8	Off	Toggle	*Label of port 8*

DMX

Direction	Output
Slow DMX	■

ART-NET

	input	output
Sub.Uni	1.0	off
Destination IP	255.255.255.255	

SACN

	input	output
Universe	1	off

OSC

Port	8000	Forward	■
Destination IP 1	192.168.1.40:8000		
Destination IP 2	0.0.0.0		
Destination IP 3	0.0.0.0		
Destination IP 4	0.0.0.0		

TCP/IP

TCP Port	7000
UDP Port	7000

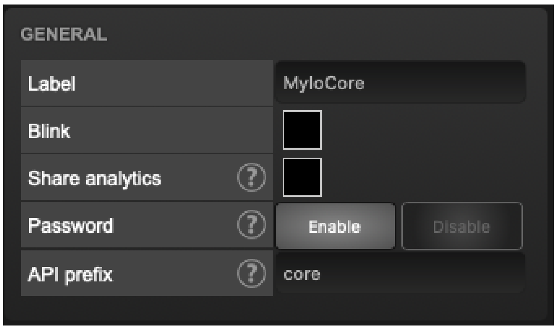
LABEL: Bedroom OPERATING MODE: Stand Alone GPI: 00000000 GPO: 00000000 RX: RS-232 DMX ART sACN UDP TCP OSC

Figure 8.1: Settings page

8.1 General

You can change the IoCore2'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.



GENERAL

Label	MyIoCore
Blink	□
Share analytics	□
Password	Enable Disable
API prefix	core

Figure 8.2: General Settings

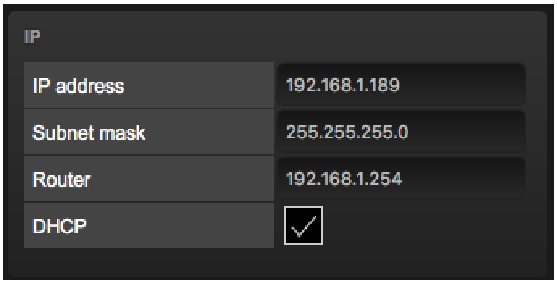
By enabling the *Share Analytics* options, you will help Visual Productions improve its products. When enabled, the IoCore2 will send daily diagnostics and usage data to Visual Productions. This information remains anonymous.

The API commands discussed in appendix D start with a prefix that is set to *core* by default. When using multiple devices from Visual Productions it can be useful to assign unique labels to these prefix, especially when using broadcasted messages. Read more about feedback loops in paragraph D.3.1.

Unauthorised users can be prevented from making changes to the IoCore2 by enabling the *Password* protection. Once enabled, the password can be disabled via the web-interface (using the *Disable* button) and the reset button (see figure 5.2). Long-press the reset button to disable the password protection; this will also revert the unit's static IP back to the default factory settings.

8.2 IP

The IP fields are for setting up the IP address and subnet mask of the IoCore2. The *Router* field is only required when Port Forwarding is used. You can also enable or disable the DHCP feature (For more information see chapter 5 at page 27).



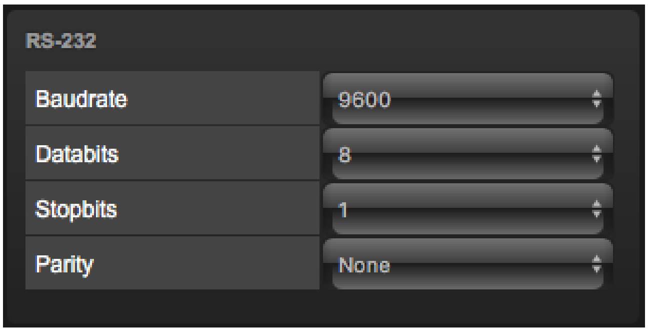
IP

IP address	192.168.1.189
Subnet mask	255.255.255.0
Router	192.168.1.254
DHCP	☑

Figure 8.3: IP Settings

8.3 RS-232

Settings for RS-232 communication like baudrate and stopbits.



RS-232

Baudrate	9600
Databits	8
Stopbits	1
Parity	None

Figure 8.4: RS-232 Settings

8.4 GPI

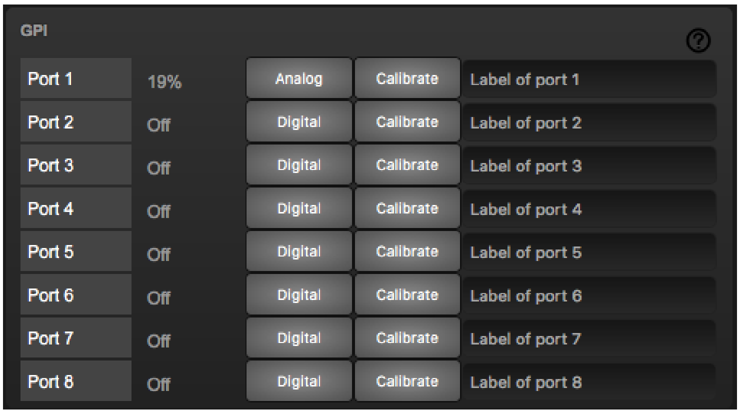
The GPI ports can be configured either as digital (contact-closure) or as analog (0-10V) input.

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.

Do not apply more than 10V to the GPI port, as this could cause permanent damage.

Each port has a label field that can contain a user-description.

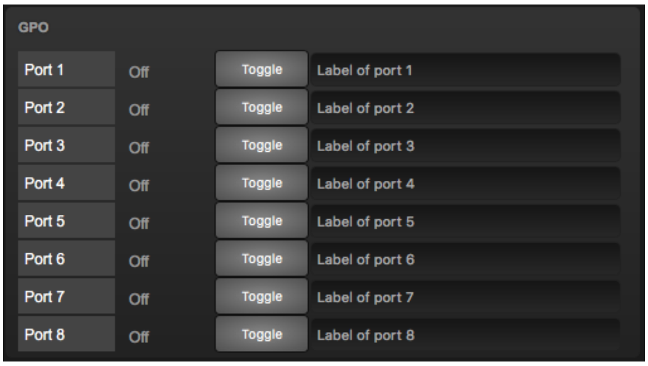


Port 1	19%	Analog	Calibrate	Label of port 1
Port 2	Off	Digital	Calibrate	Label of port 2
Port 3	Off	Digital	Calibrate	Label of port 3
Port 4	Off	Digital	Calibrate	Label of port 4
Port 5	Off	Digital	Calibrate	Label of port 5
Port 6	Off	Digital	Calibrate	Label of port 6
Port 7	Off	Digital	Calibrate	Label of port 7
Port 8	Off	Digital	Calibrate	Label of port 8

Figure 8.5: GPI Settings

8.5 GPO

This section shows the status of the GPO relays and provides buttons to switch them directly. Each port has a label field that can contain a user-description.

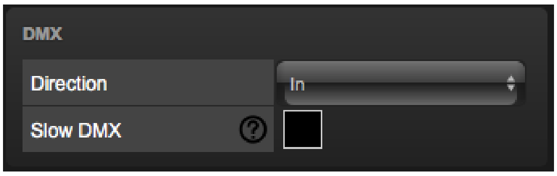


Port 1	Off	Toggle	Label of port 1
Port 2	Off	Toggle	Label of port 2
Port 3	Off	Toggle	Label of port 3
Port 4	Off	Toggle	Label of port 4
Port 5	Off	Toggle	Label of port 5
Port 6	Off	Toggle	Label of port 6
Port 7	Off	Toggle	Label of port 7
Port 8	Off	Toggle	Label of port 8

Figure 8.6: GPO Settings

8.6 DMX

The *DMX* settings specify whether the DMX port is an *Input* or *output*.



DMX

Direction	In
Slow DMX	□

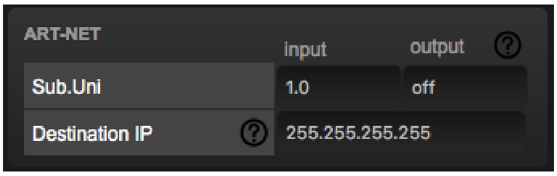
Figure 8.7: DMX settings

When the *Slow DMX* checkbox is enabled, the IoCore2 will slowdown the rate at which it sends out DMX. This is done to facilitate DMX fixtures that have difficulties keeping up with the optimal DMX transmission rate.

8.7 Art-Net

The *Art-Net* feature in the IoCore2 supports 1 universe out and 1 universes in.

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 entering 'off' in the output fields.



ART-NET

	input	output
Sub.Uni	1.0	off
Destination IP	255.255.255.255	

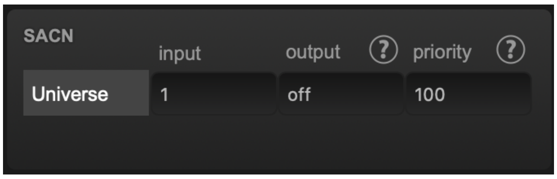
Figure 8.8: Art-Net settings

The *Destination IP* determines where the outgoing Art-Net data will be sent 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.

8.8 sACN

The IoCore2 supports 1 incoming *sACN* universe and 1 outgoing universe. The universe field should hold a number in the range of [1,63999]. Outgoing sACN transmission can be disabled by entering *off* into the sACN output field.



SACN

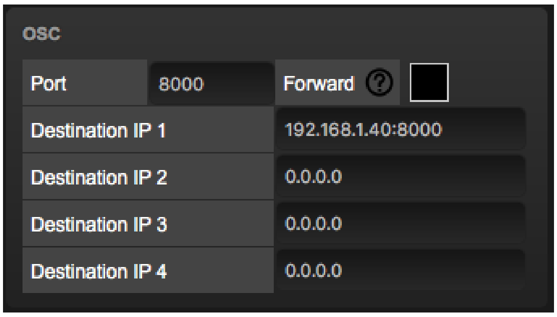
	input	output	priority
Universe	1	off	100

Figure 8.9: sACN settings

Set priority field to control the priority level of the output universe. The priority can be set from 0 (lowest) to 200 (highest).

8.9 OSC

External equipment sending OSC messages to the IoCore2 need to be aware of the number specified in the 'Port' field. This is the port the IoCore2 listens to for incoming messages.



OSC

Port	8000	Forward	□
Destination IP 1	192.168.1.40:8000		
Destination IP 2	0.0.0.0		
Destination IP 3	0.0.0.0		
Destination IP 4	0.0.0.0		

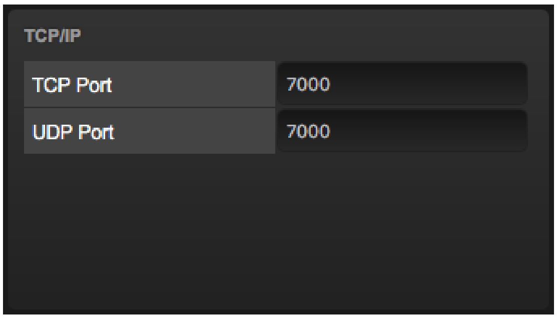
Figure 8.10: OSC Settings

The IoCore2 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 'ip-address:port' format in these fields, e.g. "192.168.1.11:9000". If a field should not be used then 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.

Enabling the *Forward* checkbox will have the IoCore2 copy every incoming OSC message and send it the addresses specified in the 'Out IP' fields.

8.10 TCP/IP

Defines the listening ports for TCP and UDP messages. External systems intending to send TCP or UDP message to the IoCore2 need to know the unit's IP address and this port number. By default both ports are set to 7000.



TCP/IP

TCP Port	7000
UDP Port	7000

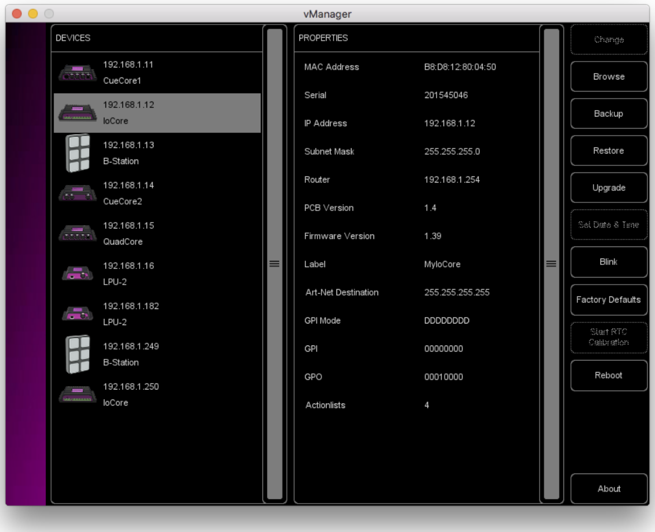
Figure 8.11: TCP/IP settings

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 upgrades
- Identify a specific device (in a multi device set-up) by blinking its LED
- Revert to factory defaults



DEVICES	
192.168.1.11	CueCore1
192.168.1.12	IoCore
192.168.1.13	B-Station
192.168.1.14	CueCore2
192.168.1.15	QuadCore
192.168.1.16	LPU-2
192.168.1.182	LPU-2
192.168.1.249	B-Station
192.168.1.250	IoCore

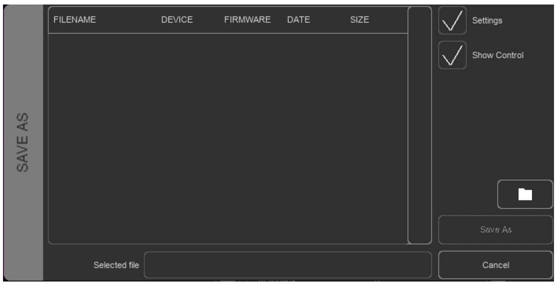
PROPERTIES	
MAC Address	B8:D8:12:80:04:50
Serial	201545046
IP Address	192.168.1.12
Subnet Mask	255.255.255.0
Router	192.168.1.254
PCB Version	1.4
Firmware Version	1.39
Label	MyIoCore
Art-Net Destination	255.255.255.255
GPI Mode	DDDDDDDD
GPI	00000000
GPO	00010000
Actionlists	4

Figure 9.1: vManager

The following section explain the buttons in the vManager, 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.



FILENAME	DEVICE	FIRMWARE	DATE	SIZE

☑ Settings

☑ Show Control

Selected file

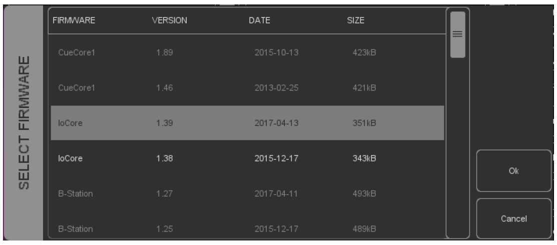
Figure 9.2: Creating a backup

Apps distributed by app stores are not allowed to access files outside this designated location. It is important to know where vManager is storing its files, in case you wish transfer a backup file to memory stick or dropbox.

The designated file location differs per operating system and is likely to be a long and obscure path. For this reason, vManager provides you with a shortcut to the correct file location. A *Folder* button can be found in the file related dialogs. Clicking this button will open a file browser at appropriate folder.

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.



FIRMWARE	VERSION	DATE	SIZE
CueCore1	1.89	2015-10-13	423kB
CueCore1	1.46	2013-02-25	421kB
IoCore	1.39	2017-04-13	351kB
IoCore	1.38	2015-12-17	343kB
B-Station	1.27	2017-04-11	493kB
B-Station	1.25	2015-12-17	489kB

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 unit by selecting a device and clicking the *Set Date & Time* button. Not all Visual Productions devices feature an internal real-time clock. The IoCore2 does not have such a RTC.

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 in the on-board flash memory. They will be completely erased and all settings will reverted to their defaults by pressing the *Factory Defaults* button. This action does not affect the device's IP settings.

9.6 Reboot

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

9.7 Installing vManager

The vManager app is available on a wide range of operating systems, both mobile and desktop.

The softwares is distributed through app-stores to take advantage of receiving future software updates automatically.

9.7.1 iOS

vManager can be downloaded from the Apple iOS app-store at `https://itunes.apple.com/us/app/vman/id1133961541`.

9.7.2 Android

vManager can be found on the Google Play store at `https://play.google.com/store/apps/details?id=org.visualproductions.manager`.

Android 5.0 or higher is required.

9.7.3 Windows

Visit the Microsoft store at `https://www.microsoft.com/en-us/p/vmanager/9nblggh4s758`.

Windows 10 is required.

9.7.4 macOS

Visit the Apple macOS app store at `https://apps.apple.com/us/app/vmanager/id1074004019`.

macOS 11.3 is recommended.

9.7.5 Ubuntu

You can acquire the vManager from Snapcraft at `https://snapcraft.io/vmanager`.

Alternatively, it can be installed by using the command-line:

```
snap find vmanager
snap install vmanager
```

To update the apps later on via the command-line type:

```
snap refresh vmanager
```

Ubuntu 22.04 LTS is recommended. The software is only available for the amd64 architecture.

Chapter 10

Kiosc

Kiosc is an application for creating custom touch screen user-interfaces for the range of lighting controllers from Visual Productions. Kiosc is designed to have no editing capability, making it a fool-proof interface that can safely be presented to non-technical operators.

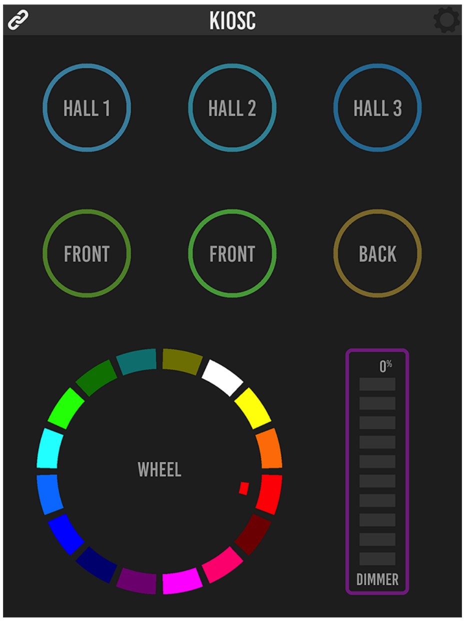


Figure 10.1: Kiosc

Kiosc is the ideal way to remote control our solid-state lighting controllers like CueluxPro, CueCore1, CueCore2, QuadCore, IoCore1, IoCore2, LPU-2, DaliCore, B-Station1 and the IoCore2. Kiosc enables you to choose scenes or presets, set intensity levels or choose RGB colours.

You can also use it to control third-party AV equipment. Kiosc speaks OSC, UDP and TCP.

Kiosc is available as software app and as a physical product. The hardware version of Kiosc is a wall-mount 7” touch screen with Kiosc pre-installed. It is powered by PoE and requires only a RJ-45 connection.

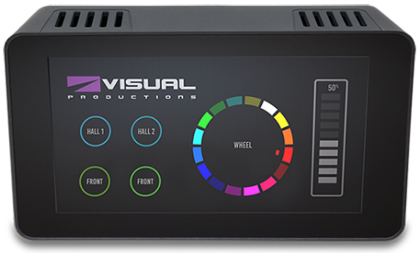


Figure 10.2: Kiosc

Please read the Kiosc manual, available from `http://www.visualproductions.nl/downloads` for more details.

Appendices

Appendix A

Trigger Types

The following tables list the different types of triggers that can be used in the IoCore2. The different types are accompanied with values and flanks.

A.1 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	Enter	Analog level enters the range
Channel 0-19%	Port number	Leave	Analog level leaves the range
Channel 20-39%	Port number	Change	Analog level enters or leaves the range
Channel 20-39%	Port number	Enter	Analog level enters the range
Channel 20-39%	Port number	Leave	Analog level leaves the range
Channel 40-59%	Port number	Change	Analog level enters or leaves the range
Channel 40-59%	Port number	Enter	Analog level enters the range
Channel 40-59%	Port number	Leave	Analog level leaves the range
Channel 60-79%	Port number	Change	Analog level enters or leaves the range
Channel 60-79%	Port number	Enter	Analog level enters the range
Channel 60-79%	Port number	Leave	Analog level leaves the range
Channel 80-100%	Port number	Change	Analog level enters or leaves the range
Channel 80-100%	Port number	Enter	Analog level enters the range
Channel 80-100%	Port number	Leave	Analog level leaves the range
Binary	Combination value	-	A combination of ports being closed
Short press	Port number	-	Short closure on port
Long press	Port number	-	Long closure on port

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 3 and 4, fill in value 12 (4+8).

A.2 DMX Input

Trigger Type	Trigger Value	Flank	Description
Channel	DMX Address	Change	Channel changes
Channel	DMX Address	Non-zero	Channel becomes non-zero
Channel	DMX Address	Zero	Channel becomes zero
UniverseA	-	-	A DMX level change in the first universe
UniverseB	-	-	A DMX level change in the second universe
Receiving	-	Change	Start receiving or loose DMX signal
Receiving	-	Stop	Lost DMX signal
Receiving	-	Start	Start receiving DMX signal

A.3 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.4 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 31 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.5 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 31 characters.

A.6 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 31 characters, including the '/'.

A.7 Art-Net

Trigger Type	Trigger Value	Flank	Description
Channel	DMX Channel	Change	Channel changes
Channel	DMX Channel	Non-zero	Channel becomes non-zero
Channel	DMX Channel	Zero	Channel becomes zero
UniverseA	-	-	A DMX level change in the first universe
UniverseB	-	-	A DMX level change in the second universe
Receiving	-	Change	Start receiving or loose Art-Net signal
Receiving	-	Stop	Lost Art-Net signal
Receiving	-	Start	Start receiving Art-Net signal

A.8 sACN

Trigger Type	Trigger Value	Flank	Description
Channel	DMX Channel	Change	Channel changes
Channel	DMX Channel	Non-zero	Channel becomes non-zero
Channel	DMX Channel	Zero	Channel becomes zero
UniverseA	-	-	A DMX level change in the first universe
UniverseB	-	-	A DMX level change in the second universe
Receiving	-	Change	Start receiving or loose sACN signal
Receiving	-	Stop	Lost sACN signal
Receiving	-	Start	Start receiving sACN signal

A.9 Kiosc

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

When editing the Kiosc 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 Kiosc software.

More details on Kiosc can be found on page 51.

A.10 Randomizer

The *Randomizer* will generate a pseudo-random number, this functionality is discussed on page 37.

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.11 System

Trigger Type	Trigger Value	Flank	Description
Startup	-	-	The IoCore2 has been powered 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

Use *Startup* if something needs to happen as soon as the IoCore2 is powered on. Please note that ethernet network might not be online yet.

Use the *Network Connection* trigger if something needs to happen as soon as the ethernet network becomes online after a power-cycle.

ReleasedByMaster is a trigger that will happen when the IoCore2 is disconnected from CueluxPro.

A.12 Variable

Variables can contains a number between 0 and 255. This functionality is explained on page 35.

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 specified number
Variable 1	Number [0,255]	Equal	Variable 1 = number
Variable 1	Number [0,255]	Stop Equal	Variable 1 stops to be = to number
Variable 2	Number [0,255]	Change	Variable 2 becomes = or # to the specified number
Variable 2	Number [0,255]	Equal	Variable 2 = number
Variable 2	Number [0,255]	Stop Equal	Variable 1 stops to be = to number
Variable 3	Number [0,255]	Change	Variable 3 becomes = or # to the specified number
Variable 3	Number [0,255]	Equal	Variable 3 = number
Variable 3	Number [0,255]	Stop Equal	Variable 1 stops to be = to number
...			
Variable 9	Number [0,255]	Change	Variable 9 becomes = or # to the specified number
Variable 9	Number [0,255]	Equal	Variable 9 = number
Variable 9	Number [0,255]	Stop Equal	Variable 9 stops to be equal to number
Variable 10	Number [0,255]	Change	Variable 10 becomes = or # to the specified number
Variable 10	Number [0,255]	Equal	Variable 10 = number
Variable 10	Number [0,255]	Stop Equal	Variable 10 stops to be = to number

A.13 Timer

There are four timers in the IoCore2. This functionality is explained on page 36.

Trigger Type	Trigger Value	Flank	Description
Channel	Timer Index	Change	The timer starts or stops
Channel	Timer Index	Stop	The timer stops
Channel	Timer Index	Start	The timer starts
Time	Timer Index	-	A Stream of the current time of that timer

The timer will automatically stop when it reaches 00:00.0. An *Action* needs to be programmed in order to restart the timer and make it loop.

A.14 Actionlist

Actionslists are discussed in detail on page **??**. The Actionlists have a checkbox for enabling/disabled. Changing this checkbox generates a trigger.

Trigger Type	Trigger Value	Flank	Description
-	Actionlist Index	Change	The actionlist enable checkbox changes
-	Actionlist Index	Down	The actionlist is enabled
-	Actionlist Index	Up	The actionlist is disabled

A.15 User List (1-4)

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

Appendix B

Task Types

Tasks allow you to automate the functionality in the IoCore2. 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 Action

Trigger another action.

Feature	Function	Parameter 1	Parameter 2
Link	Set	Action	-

B.2 Actionlist

Manipulate an actionlist.

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.3 DMX

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

Feature	Function	Parameter 1	Parameter 2
Universe	Control HTP	Universe #	-
Universe	Control LTP	Universe #	-
Universe	Control Priority	Universe #	-
Universe	Clear	Universe #	-
Channel	Set	DMX Channel	DMX Value
Channel	Toggle	DMX Channel	-
Channel	Control	DMX Channel	-
Channel	Inverted Control	DMX Channel	-
Channel	Decrement	DMX Channel	-
Channel	Increment	DMX Channel	-
Bump	Set	DMX Channel	DMX Value
Bump	Control	DMX Channel	-
Clear	Set	-	-
RGB	Set	DMX Address	RGB Colour Value
RGB	Control	DMX Address	-
RGBA	Control	DMX Address	-
XY	Control	DMX Address	-
XxYy	Control	DMX Address	-

B.4 GPI

Force the GPI actions to be triggered.

Feature	Function	Parameter 1	Parameter 2
Refresh	Set	-	-

B.5 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.6 OSC

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

Feature	Function	Parameter 1	Parameter 2
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	-
Send Colour	Set	URI	RGB colour
Send Colour	Control	URI	-

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

B.7 Randomiser

Trigger the Randomizer to generate a new random number.

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

B.8 RS-232

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

Feature	Function	Parameter 1	Parameter 2
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.9 System

Miscellaneous tasks.

Feature	Function	Parameter 1	Parameter 2
Blink	Set	On or Off	-
Blink	Toggle	-	-
Blink	Control	-	-

B.10 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

B.11 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 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
Wake On Lan	Set	MAC Address	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'.

When using the Wake On Lan feature parameter 1 should contain the MAC Address of system's NIC (Network Interface Controller) you wish to wake up. The recommended value for parameter 2 is `255.255.255.255:7`. This broadcasts the message to the whole network at port 7 which is most commonly used for Wake On Lan.

B.12 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]	-
Single Dimmer	Control	Variable #	Delta

Variables are further explained on page 35.

The Single Dimmer feature is used to increase or decrease a level by using only one switch. When controlling this task via GPI action, then closing the GPI will increase or decrease the level. Opening the GPI port will freeze on the current level. This feature is useful for controlling an intensity will just one button.

Appendix C

Templates

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

Template	Description
GPI ->OSC Float	Each GPI port converts to an 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 an OSC message using a boolean parameter. This template is intended for GPI set to digital. Specify the OSC target IPs on the settings page.
GPI ->UDP Float	Each GPI port converts to an UDP message using a floating point parameter. This template is intended for GPI set to analog. Specify the UDP target IPs in parameter 2 of each task.
GPI ->UDP Bool	Each GPI port converts to an UDP message using a boolean parameter. This template is intended for GPI set to digital. Specify the UDP target IPs in parameter 2 of each task.
GPI ->DMX	Each GPI port controls a DMX channel (1-8). The direction of the DMX port should be set to output.
DMX ->GPO	Control the GPO relays by DMX (channel 1-8). The direction of the DMX port should be set to input.
Art-Net ->GPO	Control the GPO relays by Art-Net (channel 1-8). The Art-Net universe number is determined on the settings page.
sACN ->GPO	Control the GPO relays by sACN (channel 1-8). The sACN universe number is determined on the settings page.
Receiving DMX	Read in the DMX universe. It can be outputted by Art-Net and/or sACN.
Receiving Art-Net	Read in the Art-Net universe. It can be outputted by DMX and/or sACN.
Receiving sACN	Read in the sACN universe. It can be outputted by DMX and/or Art-Net.
GPI ->CueCore2	Each GPI port controls a playback on a CueCore2. The CueCore2 is controlled via its fixed OSC mapping. Specify the OSC target IP on the settings page.
GPI ->LPU-2	Each GPI port controls a playback on a LPU-2. The LPU-2 is controlled via its fixed OSC mapping. Specify the OSC target IP on the settings page.

Appendix D

API

The IoCore2 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 uses timer #1 as an example. The number '1' can be replaced by any number in the range of [1,4].

URI	Parameter	Description
/core/tm/1/start	-	Start timer #1
/core/tm/1/stop	-	Stop timer #1
/core/tm/1/restart	-	Restart timer #1
/core/tm/1/pause	-	Pause timer #1
/core/tm/1/set	time-string	Set timer #1 at the time-string

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

URI	Parameter	Description
/core/va/1/set	integer	Set the value of variable #1
/core/va/1/refresh	-	Refresh variable #1; a trigger will be generated as if the variable changed value
/core/va/refresh	-	Refresh all variables; triggers will be generated

The following table shows how to active miscellaneous functions.

URI	Parameter	Description
/core/blink	-	Momentarily flashes the IoCore2's LED
/core/hello	-	The unit will reply with the same Hello message

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 uses timer #1 as an example. The number '1' can be replaced by any number in the range of [1,4].

String	Description
core-tm-1-start	Start timer #1
core-tm-1-stop	Stop timer #1
core-tm-1-restart	Restart timer #1
core-tm-1-pause	Pause timer #1
core-tm-1-set=<text>	Set timer #1 at the time-string

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

String	Description
core-va-1-set=<integer>	Set the value of variable #1
core-va-1-refresh	Refresh variable #1; a trigger will be generated as if the variable changed value
core-va-refresh	Refresh all variables; triggers will be generated

The following table shows how to active miscellaneous functions.

String	Description
core-blink	Momentarily flashes the IoCore2's LED
core-hello	The unit will reply with the same Hello message

D.3 Feedback

The IoCore2 is able to send feedback to external equipment using its API, so called 'clients'. The IoCore2 keeps a memory of the last four OSC clients and last four UDP clients. The clients will automatically receive updates on several playback related state changes.

Below is a table listing the messages the IoCore2 will send back to its clients.

OSC	UDP
/core/al/1/enable	core-al-1-enable
/core/hello	core-hello

The `hello` command is ideal for polling the device; it allows you to verify that the IoCore2 is online at the IP address and port that you expect.

A power-cycle will clear the internal client lists. Send `/core/goodbye` or `core-goodbye` to explicitly be removed from the client list.

Consider programming custom action in the show control when additional feedback functionality is required.

D.3.1 Preventing a feedback loop

Feedback is automatically send to a device which uses the OSC or UDP API. If the external device is also a Visual Productions unit then the feedback message could be interpreted by the external unit a new command. This can result in another feedback message being generated. An endless stream of feedback messages can stall the units involved.

This feedback loop can be prevented by assign a unique label the device's API prefix. This setting is discussed on page 41.

Index

vManager, 47
19 rack, 24

Action, 31, 33
Actionlist, 32, 61
Android, 50
API, 70
API prefix, 41, 73
Art-Net, 13, 44, 58
Auto-sensing, 27

Backup, 48
Baudrate, 42
Blink, 41, 49
Bopla, 23
Box contents, 8

Calibrate, 42
CE, 8
Comparison, 8
Contact-closure, 25
Control, 34

Date, 49
Destination IP, 44
DHCP, 14, 28, 41
DIN Rail, 12, 23
DMX, 12, 26, 43
DMX Input, 56

Execute, 35

Factory Defaults, 49
FCC, 8
Feature, 62
Features, 8
Feedback, 73
Firmware, 48
Flank, 33, 54
Forum, 9
Forward, 45
Function, 62

General, 41
GPI, 11, 25, 42
GPO, 11, 43

IoCore1, 8
iOS, 49
IP, 41

Kensington, 24
Kiosc, 33, 51, 59

Label, 41
LED, 28
Lock, 24

macOS, 50
Master IP, 29
Memory, 9
Mode, 29
Monitor, 38
Mounting, 23

Network Connection, 59
Network Interface Controller, 66

OSC, 14, 45, 57

Password, 41
PoE, 24
Polarity, 24
Port, 45
Port Forwarding, 30, 41
Potentiometer, 25
Power, 24
Prefix, 41, 73

Quickstart, 15

Rackmount, 24
Randomizer, 37, 59, 64
RdmSplitter, 12
Reboot, 49

Relay, 43
ReleasedByMaster, 59
Reset button, 28
Router, 41
RS-232, 12, 42, 56

sACN, 13, 44, 58
Settings, 40
Share Analytics, 41
Show Control, 31
Single Dimmer, 67
Slow DMX, 44
Snap, 50
Source, 32
Startup, 59
Subnet mask, 28
System, 59

Task, 32, 34
Task type, 62
TCP, 13, 45, 57
TCP/IP, 45
Template, 35, 68
Templates, 68
Time, 49
Timer, 60, 66
Timers, 36
Trigger, 32
Trigger Type, 54
Trigger type, 54
Tutorial, 15

Ubuntu, 50
UDP, 13, 45, 57
Uptime, 29
URI, 57
User List, 37, 61

Variable, 59
Variables, 35, 67
VPN, 30

Wake On Lan, 66
Windows, 50