

VERSION 1.2
NOVEMBER 16, 2018

IOFIREBUG ENGINE FOR UNIVERSAL ROBOTS

INSTRUCTION MANUAL



4EACH S.R.O.
WWW.4EACH.CZ

TABLE OF CONTENTS

Instruction manual.....	0
Intended audience	2
Disclaimer.....	2
Copyright.....	2
Power supply.....	4
RS-422 Interface.....	4
RS-422 line Termination:	4
RS-485 Compatibility.....	4
Binary input connection.....	5
Binary output connection	6
Analog input connection.....	7
Wagon 16I expansion module.....	8
Wagon 8O expansion module.....	9
Compatibility.....	10
Installation.....	10
Configuration	12
Using IOFIREBUG with your robot application.....	14
Set output program node	14
Available variables.....	15
Available functions.....	15
iofb_get_digital_input.....	15
iofb_get_analog_input.....	15
iofb_get_analog_inputF.....	16
iofb_set_digital_output	16
iofb_get_digital_output	16
Device initialisation.....	17
Device default values	17

Notes on the documentation

INTENDED AUDIENCE

This description is only intended for the use of trained specialists in control and automation engineering who are familiar with the applicable national standards.

It is essential that the following notes and explanations are followed when installing and commissioning these components.

The responsible staff must ensure that the application or use of the products described satisfy all the requirements for safety, including all the relevant laws, regulations, guidelines and standards.

DISCLAIMER

The documentation has been prepared with care. The product described is, however, constantly under development. For that reason, the documentation is not in every case checked for consistency with performance data, standards or other characteristics. If it contains technical or editorial errors, we retain the right to make alterations at any time and without warning. No claims for the modification of products that have already been supplied may be made based on the data, diagrams and descriptions in this documentation.

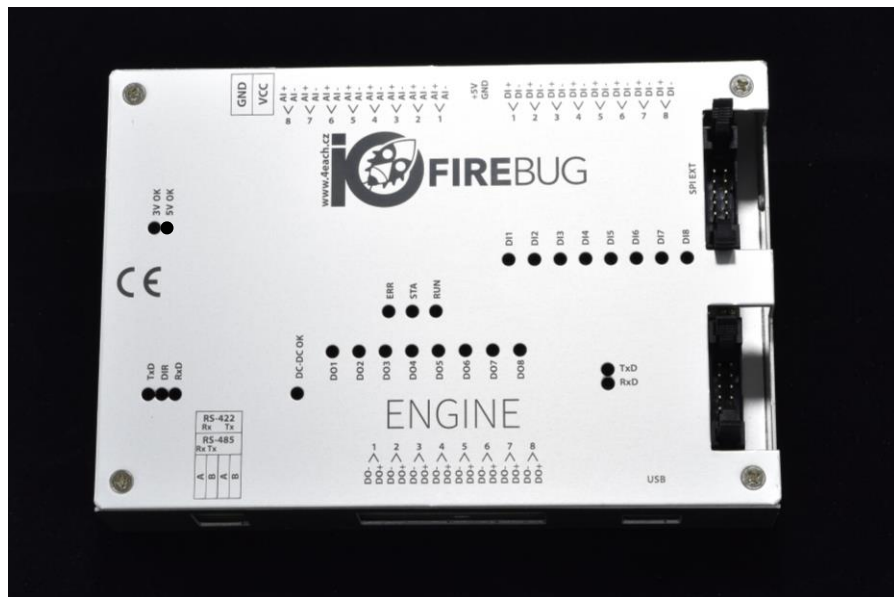
COPYRIGHT

4Each and IOFireBug are registered trademarks of and licensed by 4Each s. r. o., Czech Republic

Product overview

IOFireBug Engine block diagram is shown on following image. Device is equipped with following interfaces:

- 8x analog input
- 8x binary input (galvanically isolated)
- 8x switching N-FET output (galvanically isolated, all channels have common ground)
- RS485 / RS422 Interface (switchable, galvanically isolated)
- USB to Serial Interface (FTDI)
- SPI Interface for optional expanders
- I2C Intentionally unused



POWER SUPPLY

Device can be powered using DC power supply, using 10V to 30V.

Power LED indication:

LED	3V OK	5V OK	DC-DC OK
Meaning	Voltage present 3,3V from power supply	Voltage present 5,0V from power supply	Voltage present 5,0V from galvanically isolated power supply
Color	Red	Red	Red

If everything is ok, all three LED shine continuously.

USB TO COMPUTER INTERFACE

The device is equipped with a USB interface for direct communication with robot control box unit. Status of device communication is indicated by LEDs near USB connector.

Communication status LED:

LED	TxD	RxD
Meaning	TxD (unit is sending data)	RxD (unit is receiving data)
Color	Green	Red

RS-422 INTERFACE

IOFirebug Engine for Universal Robots supports RS-422 communication interface. Robot CB can communicate with IOFirebug using additional USB to RS-422 converter (ie. SB485s from papouch.com). Both interfaces (USB and RS-422) are completely interchangeable but make you sure to use only one.

RS-422 LINE TERMINATION:

Using jumper on P33 and P34, P35 and P36 port terminates the RS422 line (part of delivery). Please ensure, that these jumpers are inserted before connecting RS-422 interface.

RS-485 COMPATIBILITY

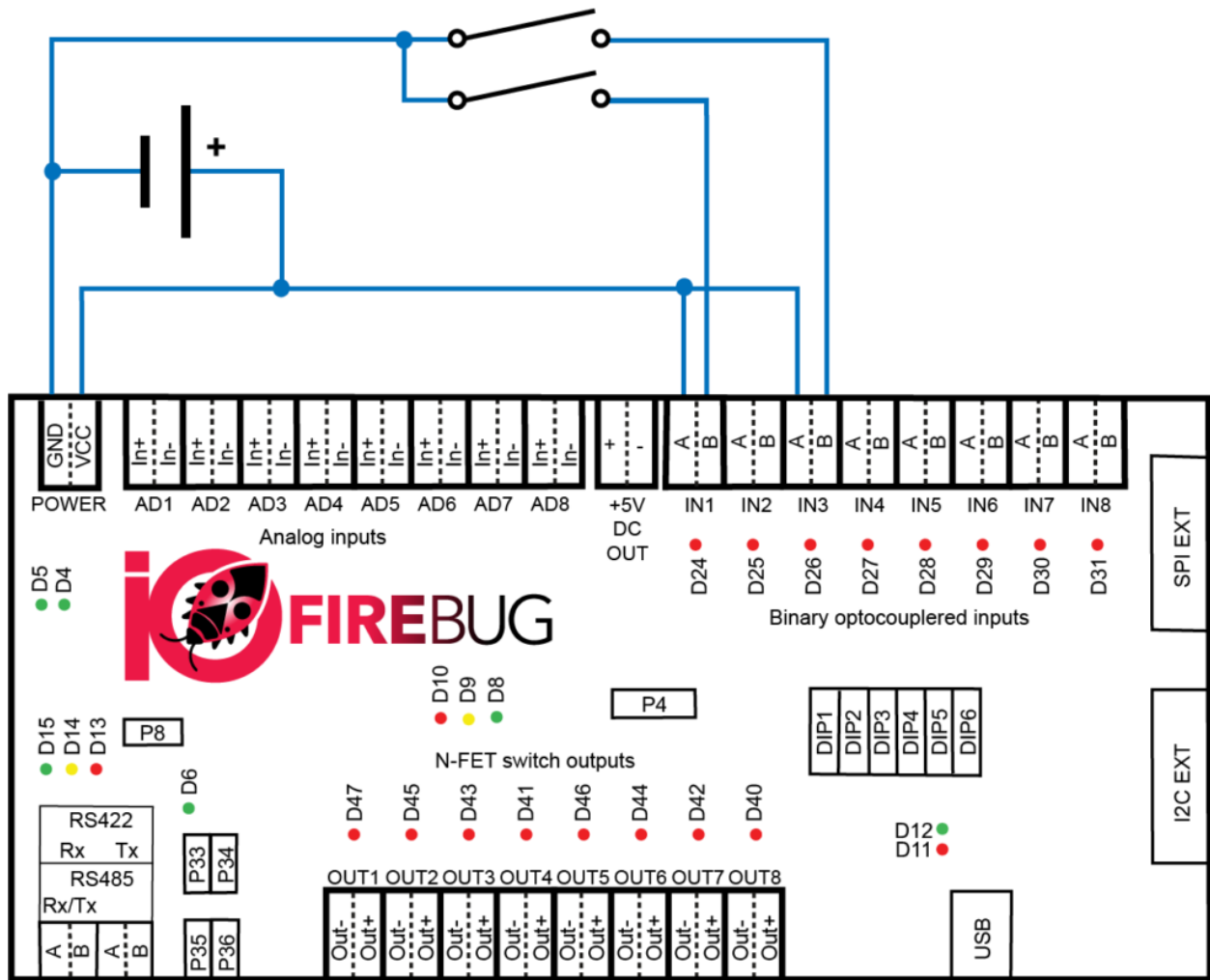
Though IOFirebug Pro supports the RS-485 interface, the IOFirebug Engine for Universal Robots currently **doesn't** support RS-485 communication.

Please do not insert RS-485 jumper (P8) into the Engine board.

I/O connection

BINARY INPUT CONNECTION

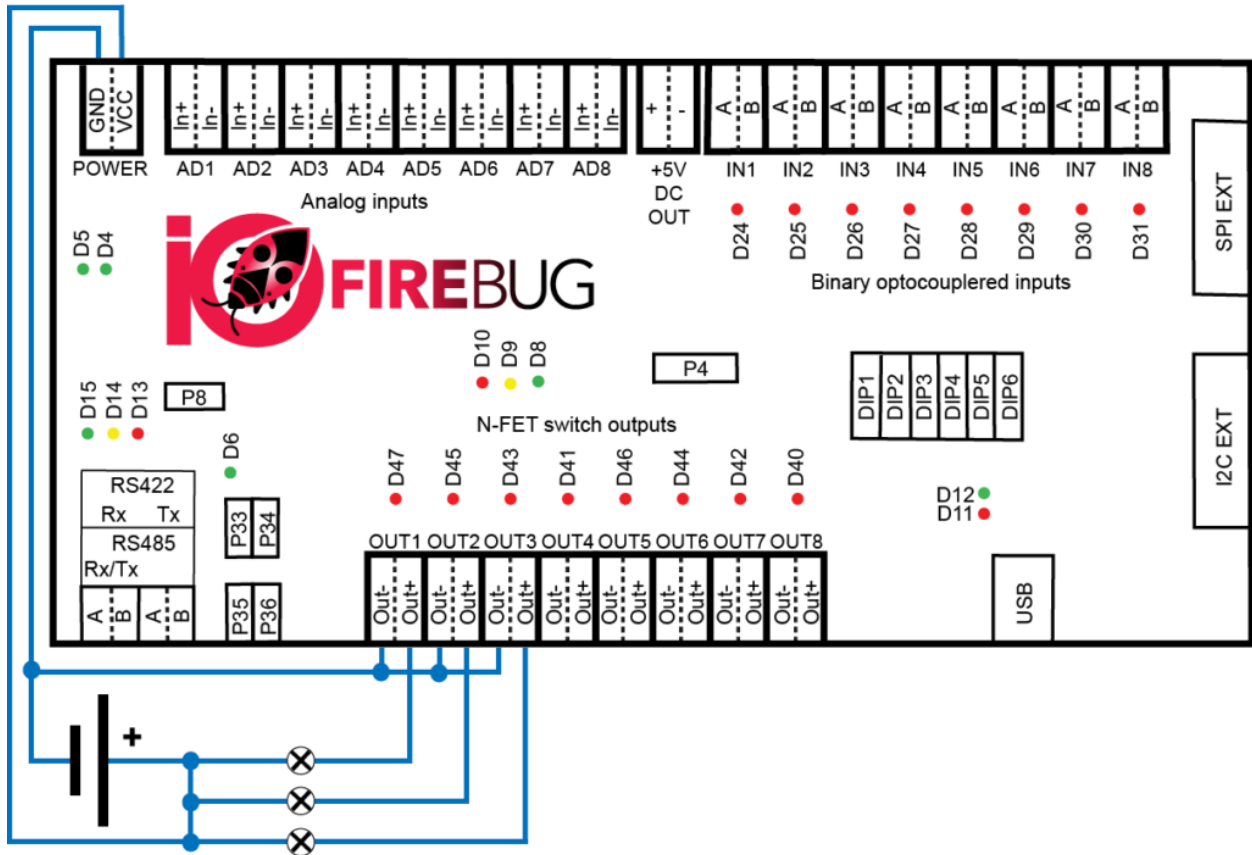
Binary inputs are galvanically isolated using opto isolators and independent of input voltage. Input voltage range is from 10V to 30V. Every input has its own indication LED, which is lid on as output is in HIGH state.



1 BINARY INPUT WIRING EXAMPLE

BINARY OUTPUT CONNECTION

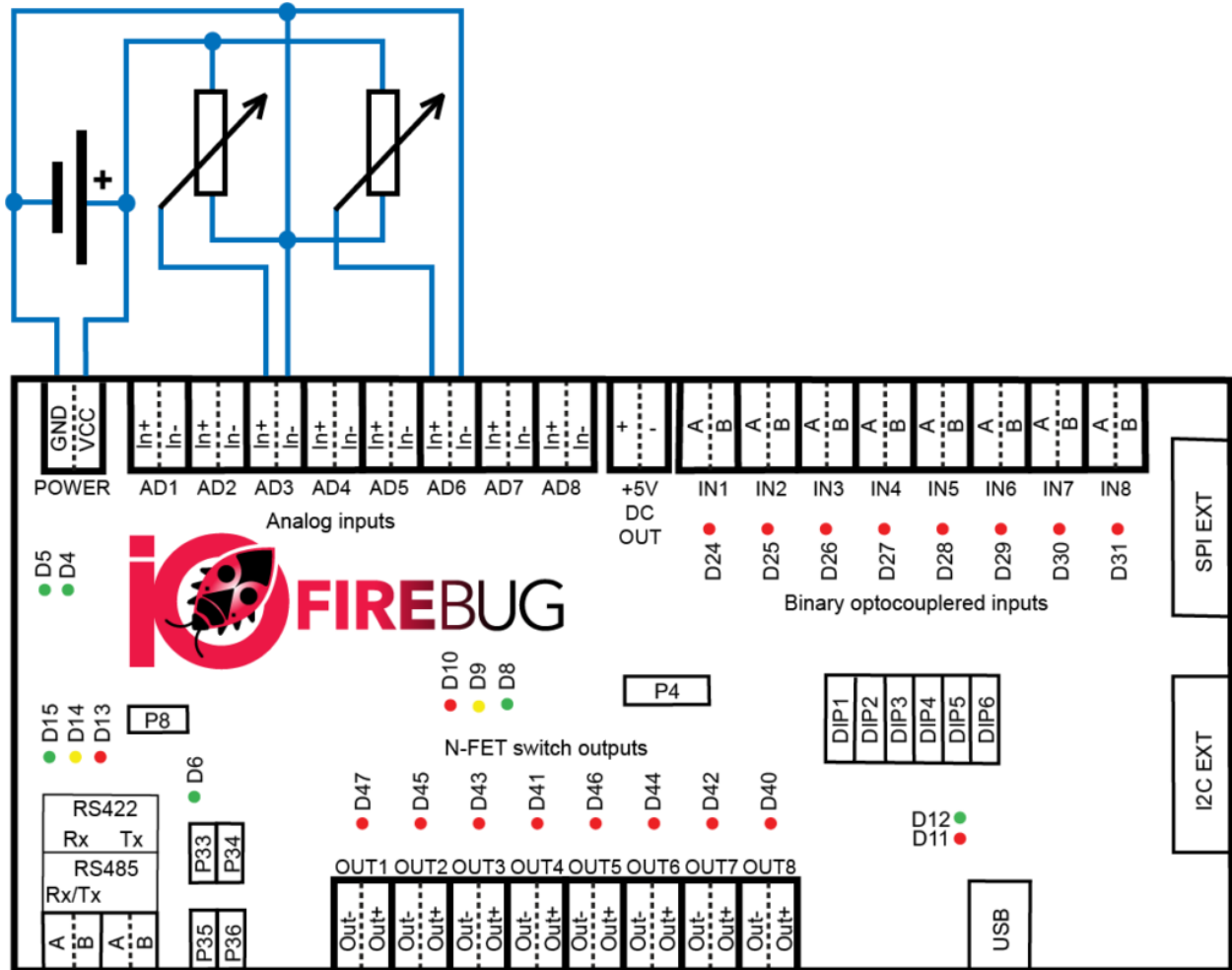
The binary outputs consist of N-FET switching transistor. Load switching is realized by grounding (connecting negative wire). The transistor output is rated for switching 3A of continuous current, in short-term can switch up to 4A. Maximum switching voltage is 50V. Each output has a LED indicator that shines red when the transistor (output) is closed.



2 BINARY N-FET OUTPUT WIRING EXAMPLE

ANALOG INPUT CONNECTION

Analog inputs are used to precisely determine input voltage.



3 ANALOG INPUT WIRING EXAMPLE

IOFirebug is equipped by 8-channel 12bit A/D converter. Analogue inputs are designed to measure DC voltage 0 to 10V. Minimum value is 0 for 0V, maximum safe value is 3413 for 10V (input can accept 12V for short time, in this case the value is 4095).

- Use `iofb_get_analog_input` function for reading a value in the integer format
- Use `iofb_get_analog_inputF` function for reading a value in the float format

Wagon expansion modules

Modules are connected to the IOFireBug Engine device with 10 pin ribbon cable (part of Wagon delivery), using SPI interface. When using extension modules, IOFireBug Engine must be properly configured by URCap. Expansion modules can be connected in chain in any order.

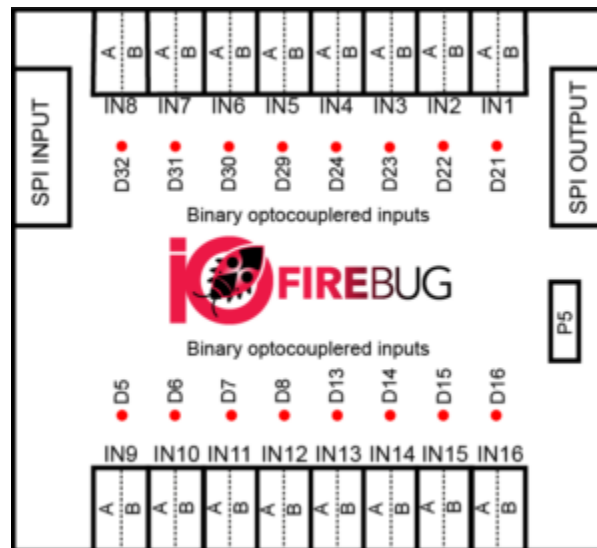
Only the last module must be terminated by jumper.

The Engine can use a maximum of 8 expansion modules at once.

WAGON 16I EXPANSION MODULE

Wagon 16I allows you to connect additional 16 digital inputs to the IOFireBug Engine. Inputs are electrically identical to those on the Engine device. Connection circuitry is therefore analogous. If module is last in the chain, bus must be terminated by P5 jumper.

The module is powered by the Engine device. Each input has its own LED that shines red when the input is in HIGH state.

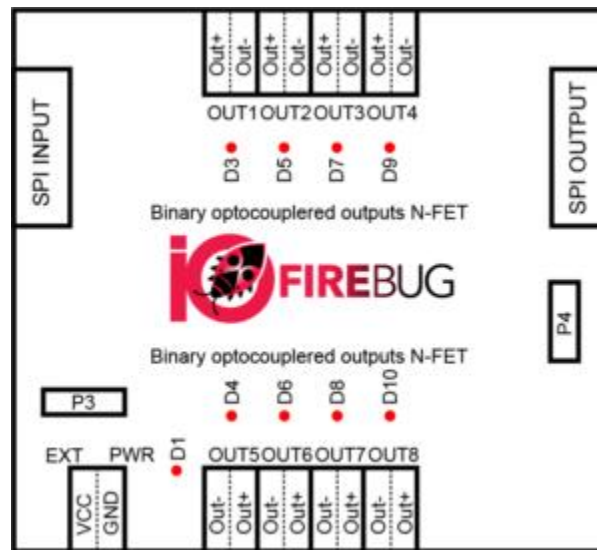


4 BINARY INPUT EXPANDER

WAGON 80 EXPANSION MODULE

Expansion Module allows you to connect additional 8 binary outputs (N-FET) to the IOFireBug Engine. Outputs are electrically identical to those on the Engine device. Connection circuitry is therefore analogous. If module is last in the chain, bus must be terminated by P4 jumper.

By default, the module is powered by the Engine device, but it may also be powered from an external power supply. To select how the module is powered, use jumper P3. When the module is powered correctly, green LED D1 shines. Each output has its own LED that shines red when the output is in HIGH state.



5 BINARY N-FET OUTPUT EXPANDER

URCap plug-in

IOFirebug Engine for Universal robots is designed to use with URCap robot plug-in.

Please note, that other versions of IOFirebug products are not compatible with URCap.

COMPATIBILITY

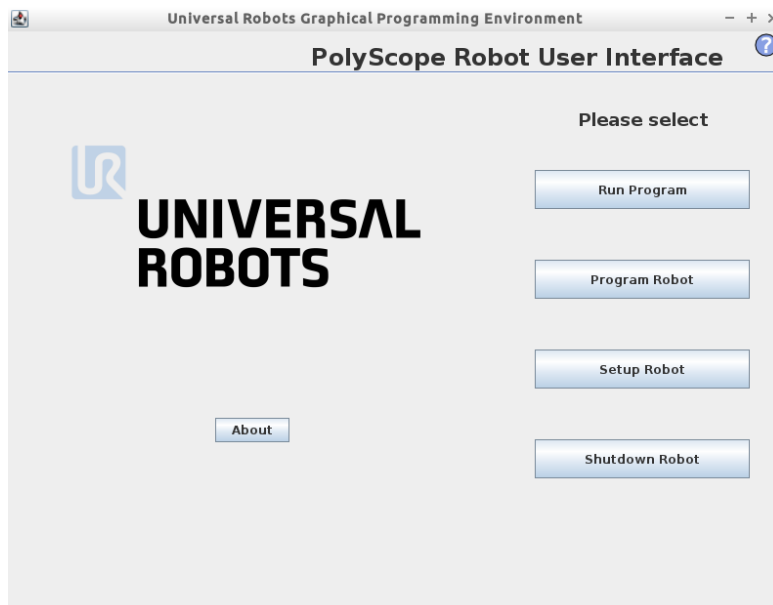
IOFirebug for Universal robots and URCap is working with CB3.1 and Polyscope 3.4 and higher (including e-series).

Following instructions are demonstrated on Polyscope version 3.4, but all principals are same in new versions of PolyScope as well.

INSTALLATION

Plug-in installation procedure is following:

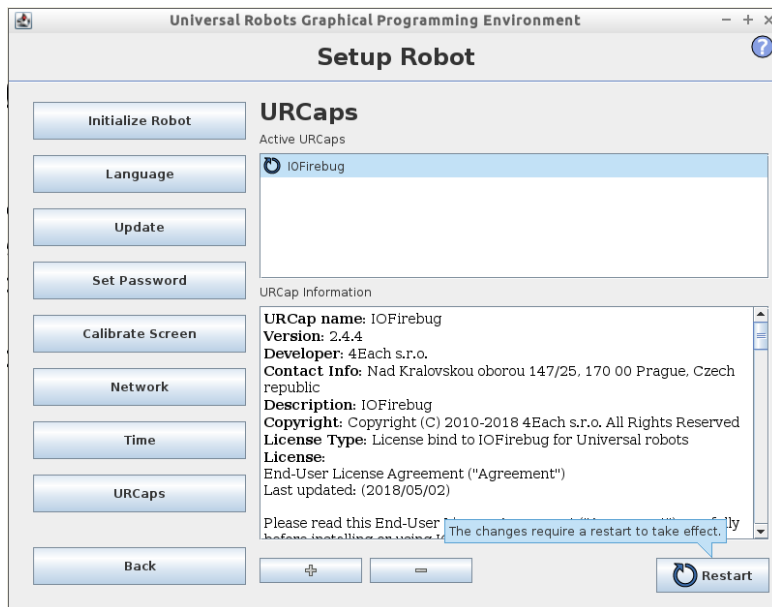
- 1) Download the URCap at: <https://4each.cz/iofirebug-engine-for-universal-robots>
- 2) Copy .urcap file on the flash disk and insert it in the teach pedant of the robot
- 3) Connect the IOFirebug Engine for Universal Robots to the robot and provide correct power supply.
- 4) Choose Setup Robot



5) Choose URCaps



6) Click “+” button for adding the plug-in and find the .urcap file on the flash disk.



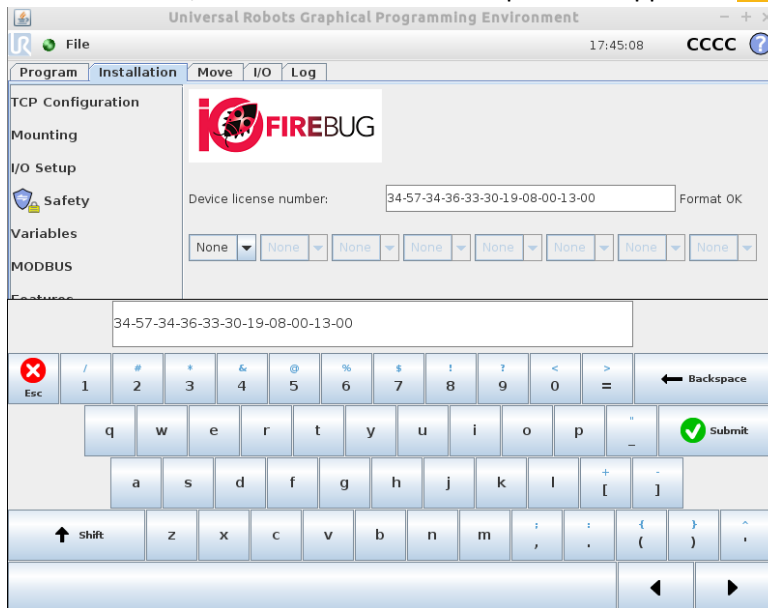
7) Click on “Restart” button and wait until robot starts again

CONFIGURATION

- 1) Go to the Program Robot screen and choose the Installation tab



- 2) If the URCap installation is success you will find IOFirebug setting
- 3) Type the device license number which is part of product delivery (if you feel that you miss your license number, don't hesitate to contact product support at 4each@4each.cz)



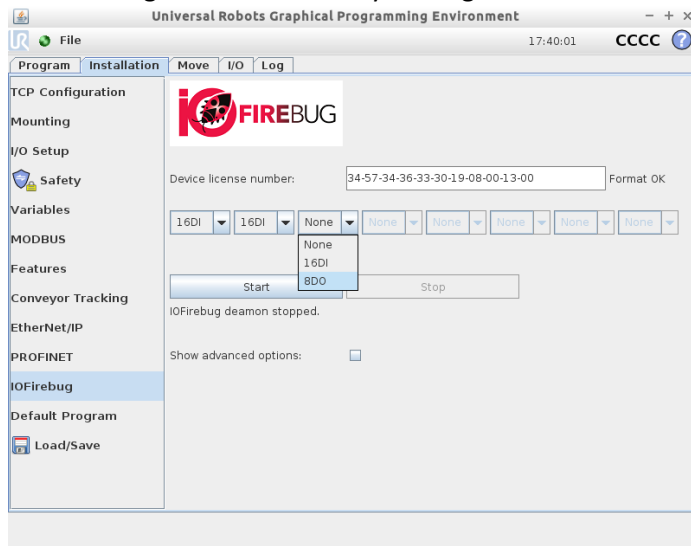
Note: Correct format is like 00-01-02-03-04-05-06-07-08-09-10. These values are in hexadecimal format, so it can contain character a to f as well.

- 4) Define the interface for connection. If there are no other USB accessories connected to the robot it will always be "ttyUSB0".

Note: Number of displayed ports doesn't correspond with physical interfaces of the robot. This setting is only for wider compatibility in case of use the UR with other similar accessories from other UR+ vendors.

Note: In newer versions of IOFireBug URCap interface may be hidden under "advanced options".

5) Choose Wagons connected to your Engine in correct order (left to right)



6) Click "Start" button

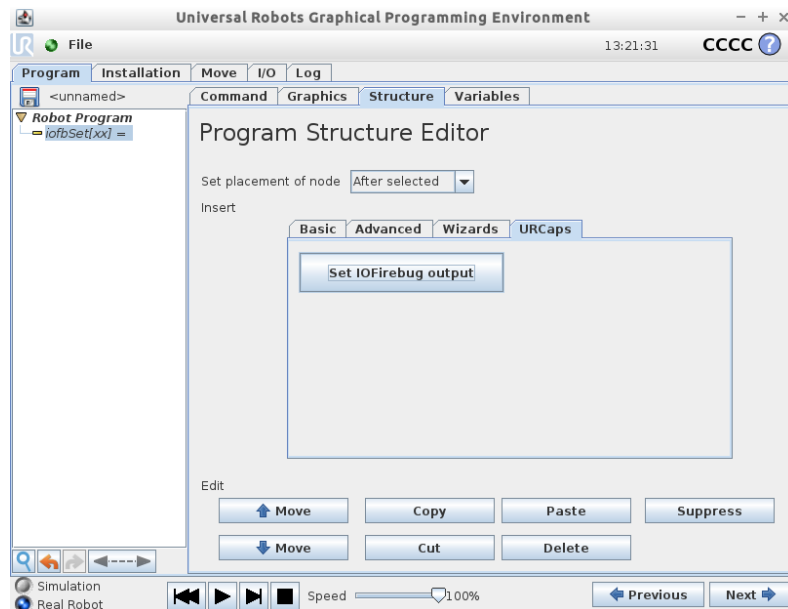
USING IOFIREBUG WITH YOUR ROBOT APPLICATION

IOFirebug products for Universal robots allows to expand robot I/O interface depending on count of Wagon modules connected.

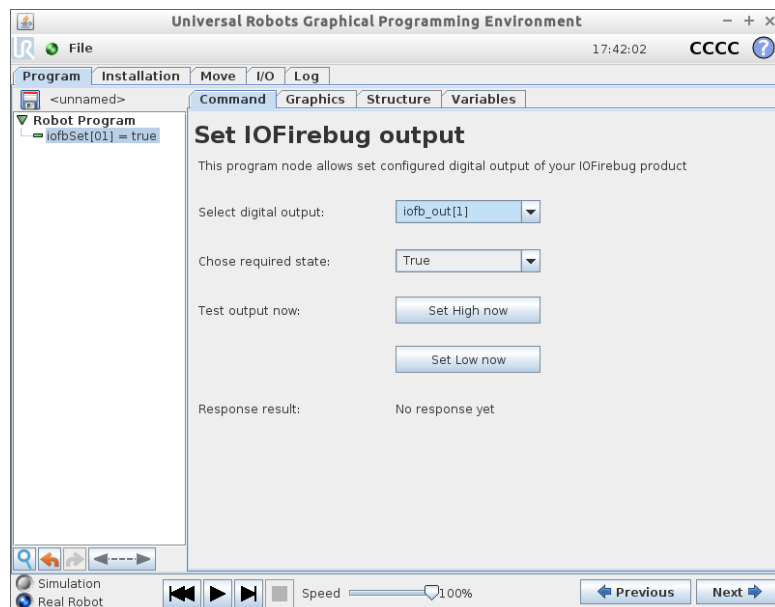
In this document we assume using one Wagon 16I and one Wagon 8O. The IOFirebug system has 24 digital inputs and 16 digital outputs configured. Engine is equipped with 8 analogue inputs without any expanding capability.

SET OUTPUT PROGRAM NODE

It is possible to put Set IOFirebug output function to the UR program from “Structure” Tab.



When this function is inserted, a program node with undefined output will appear in the program tree. Command Tab contains user interface for defining the index of the output and its value.



In drop-down lists can be defined requested value for a specific digital output. By the “Perform action now” button is possible to set requested value to the output instantly.

Because the IOFirebug system is connected externally, URcap supports the confirmation of commands sent to the output. Result of this confirmation is shown next to the button (OK/ERR).

This program node is very useful during the wire check. It is not necessary to create any application logic to check the wiring.

Disadvantage of the “Set IOFirebug output” node is, that program node doesn’t return the confirmation status to the UR application. For that case URcap adds some functions to UR Script.

AVAILABLE VARIABLES

URcap defines globally available variables with count of configured digital inputs and outputs

iofb_configured_di_cnt contains the number of configured digital inputs

iofb_configured_do_cnt contains the number of configured digital outputs

AVAILABLE FUNCTIONS

URcap adds following function to the expression editor:

```
IOFirebug - 4Each s.r.o.
iofb_get_digital_input(<iofbinput>)
iofb_get_analog_input(<iofbinput>)
iofb_get_analog_inputF(<iofbinput>)
iofb_set_digital_output(<iofboutput>, <value>)
iofb_get_digital_output(<iofboutput>)
```

IOFB_GET_DIGITAL_INPUT

This function allows UR program to get state of selected digital input. Function expects the index of configured digital input and returns its state. If the function *iofb_get_digital_input* is called with higher index of digital input than configured count, the return value will always be False. If the communication with the device is broken or device is turned off the function returns False.

Arguments

uint iofbinput: index of configured digital input.

Return value

bool: State of digital input with *iofbinput* index

IOFB_GET_ANALOG_INPUT

This function allows UR program to get state of selected analogue input. Function expects the index of analogue input and returns its value. If the function *iofb_get_analog_input* is called with index higher than 8 function returns value -1. If the communication with the device is broken or device is turned off the function returns 0.

Arguments

uint iofbinput: index of analogue input. (1...8)

Return value

uint: Value of analogue input with *iofbinput* index

IOFB_GET_ANALOG_INPUTF

This function calls internally `iofb_get_analog_input` method and converts return value to decimal volts using float number.

Arguments

uint iofbinput: index of analogue input. (1...8)

Return value

float: Value of analog input in volts with *iofbinput* index

IOFB_SET_DIGITAL_OUTPUT

This function allows UR program to set the state of selected digital output. Function expects the index of configured digital output and requested value. This function returns confirmation status of the operation (disconnected or powered off device returns False). If the function `iofb_get_digital_input` is called with higher index of digital output than configured count, the return value will always be False.

Arguments

uint iofboutput: index of configured digital output.

bool value: requested value.

Return value

bool: Confirmation status of the operation

IOFB_GET_DIGITAL_OUTPUT

This function allows UR program to get back the state of selected digital output. Function expects the index of configured digital output. This function returns the state of the output with a small delay since the output is set by the function `iofb_set_digital_output` or `iofb_set_digital_outall`.

Arguments

uint iofboutput: index of configured digital output.

Return value

bool: status of the digital output

Troubleshooting

DEVICE INITIALISATION

During the initialization process some issues can happen. This chapter will try to help you what to do in case of some error messages.

Message: USB port error or device disconnected!

Recommended steps:

In case of USB connection: Be sure that USB cable is not broken and no longer than 1.8m and the device has correct power supply (10 – 30 V DC).

In case of RS-422 connection using converter: Check the USB cable or try to use another converter.

Message: Communication broken or invalid address!

Recommended steps:

In case of USB connection: Be sure that all DIP switches of the device are set to OFF except DIP6, which should be ON.

In case of RS-422 connection using converter: Check the state of DIP switches states. Be sure you are using double twisted-pair cable no longer than 100m and the RS-422 line is properly terminated (converter setting, All IOFirebug Engine jumpers P33-P36 are present).

Message: Invalid license number or device version!

Recommended steps: Stop the daemon by pressing “Stop” button and check if license number is correct. This message can be shown in case of misconfiguration of the device. In that case, try to use the “DEVICE DEFAULT VALUES” procedure.

Message: This is not UR+ Certified device!

Recommended steps: Buy UR+ certified product. Other IOFirebug products are not compatible with URCap.

Message: Daemon internal error!

Recommended steps: Remove installed URCap and install the version from the vendor site.

DEVICE DEFAULT VALUES

If you set all DIP switches (inside the box) to position ON and reboot the device (cut off power supply), the unit clears the last configuration of EEPROM memory, this sets the default values. The unit indicates memory erase by flashing red LED RUN.

To return to normal operational state, you need to switch DIP1 to DIP5 switches to an OFF state and DIP6 to an ON state. Don't forget to restart unit one more time before use.