

The KIPR Wallaby/Link in comparison to the Hedgehog

Tiefenbach Jaqueline, Waldherr Julian, Killer Lorenz, Hantich Tobias, Beisteiner Michael, Höller Dominik

Department for Computer Science
Federal Technical Secondary College in Wiener Neustadt
Austria
s0urce@outlook.at

Abstract— This publication outlines the Differences between the Controllers KIPR Link, KIPR Wallaby and AndriX v3 alias Hedgehog, both Hardware- and Software-related, so that the Controller, which fits best to your claims can be chosen for Botball.

Keywords—Controller Comparison, AndriX v3, Hedgehog, KIPR Link, KIPR Wallaby

I. INTRODUCTION

The intention of this publication is to inform about 3 important Controllers for Botball-applications. The benefits and disadvantages will be outlined and an overview of the Controllers will be given. The content refers to the Hardware-Specs and Ports as well as the Software-Peculiarities of the Control-Units in comparison to each other.

II. HEDGEHOG LIGHT

A. Tech Specs

- STM32F4 + Raspberry Pi 3 Model B (inclusive WIFI)
- 16 sensor ports total
 - 8 analog ports
 - 8 digital ports
- 4 Motor-Ports
- 4 Servo-Ports
- Battery-Port
- I²C Port
- SPI Port
- 1 serial USART-Port
- Ethernet-Port
- 4 USB-Ports
- Micro-SD Slot

B. General Information

The Hedgehog Light is the new Version of the old Hedgehog-Controller. The primary Communication-Interface changed from USB to WLAN-Connection. It is primarily designed to use it with a web IDE, but the Controller can also work with other Mobile- and Desktop-Operating Systems. Although hosting its own WiFi network is possible with the Hedgehog Light, support is currently focused on connecting to existing networks. The Hedgehog Light uses the WiFi built into the Raspberry Pi 3, lifting the protocol restriction of Hedgehog/AndriXv3. Thanks to the WLAN Connection, the Smartphone/Computer is a complete Development-Tool which can monitor the robot quite good. On the other Hand, the direct control over WLAN can lead to slower reaction-times. The WLAN-Controller is also restricted to the Hedgehog-Protocol. To bypass the Reaction Time Problem, the programs can be executed with the on-board Raspberry-Pi 3 Model B. The Hedgehog Light also has a serial interface, to which peripheral hardware can be connected, if they communicate via USART (Universal Synchronous Receiver Transmitter), which means hardware like the iRobot Create can be used and controlled with the Hedgehog Light.



Fig. 1. Hedgehog Light



Fig. 2. Raspberry Pi 3 Model B

III. KIPR LINK

A. Tech Specs

- 8 Digital I/O-Ports
- 8 Analog Input-Ports
- 4 Motor Ports
- 4 Servo-Ports
- 2 USB-Ports
- 1 Micro-USB Port
- 1 TTL-Serial-Port
- Color touch screen (320x240 dpi)



Fig. 3. KIPR-Link Controller

B. General Information

The KIPR Link is the predecessor model of the KIPR Wallaby. It is developed by the KISS Institute for Practical Robotics and the programs get compiled to the Link via a Program, the KISS IDE. The primary language, in which you program in the Development Environment is C. This Controller can be used from beginners and newcomers to Robotic, who want a user-friendly device, as well as from experts, who want a device with many features. In addition to the controller, an USB-Cable, a Power Adapter and a USB-Camera is included. Left the KIPR Link on charge for a longer time, like the whole night, may damage the battery. In about 90 Minutes, the accumulator of the Link should be fully charged, when the

Controller is turned off. It is not possible to create a WiFi-Network with the Link, so you have to use an existing one.

IV. KIPR WALLABY

A. Tech Specs

- 10 Digital Ports
- 6 Analog Ports
- 4 Motor Ports
- 4 Servo Ports
- 1 Micro-USB Port
- Color Touch Screen



Fig. 4. KIPR-Wallaby Controller

B. General Information

The KIPR Wallaby is the 4th and latest Version of the KIPR Controllers. His successor Model is the KIPR Link, so everything mentioned with the Link applies to the Wallaby equally. To write Programs and compile them to the Controller, the KISS Web IDE is used. Serial Hardware, like the iRobot Roomba, can be connected via the (...) Serial-Port.

V. COMPARISON

A. Programming Languages:

If we take a look at the programming languages, which are supported on the three controllers, there are the first differences. On the Wallaby and the Link, ANSI C is used in the IDEs. All three controllers have in common that each of them supports C++ and JavaScript. It is also possible to write programs in Python on the Wallaby and the Hedgehog Light.

B. Development Environment:

In order to compile a Program to the KIPR Link, a download and installation of the KISS IDE (KIPR's Instructional Software System) is necessary. The Link supports no Web-IDE. The Hedgehog Light although has an integrated

Web IDE, which can be used. For the Wallaby controller, you can use both Environments, either the KISS IDE as a Program or as a Web-Application. In this matter, the Wallaby is the best controller, because it unites the possibilities of the other two.

C. Connection

To connect to the Hedgehog Light and compile the Program, a WiFi connection is necessary. The Hedgehog Light is configured to connect to a WiFi named “hedgehog” by default, but can be configured for any number of networks by plugging in a flash drive with a configuration file at startup. The KIPR Wallaby and Link are also able to connect to any WLAN, no matter which name it has. The Hedgehog is able to connect to password protected Networks, but by default the password has to be “hedgehog”. For the Link and the Wallaby, it is not recommended to use password protected networks. In addition, the Hedgehog and the Wallaby have the opportunity to create their own Network, to which a Computer or Smartphone can be connected. Another Feature, which is just supported by the KIPR Controllers, is a wired Connection via USB. A useful alternative, because Wireless Connections are more error-prone. The Hedgehog Light supports wired connections as well, but via Ethernet. However, it is necessary to provide the Hedgehog Light with an address via DHCP.

D. Ports

All three Controllers do have 16 ports. The sensor ports of the Hedgehog Light can be configured as digital outputs as well. The Hedgehog Light and the Link have got 8 Digital ports and 8 Analog ports. The Wallaby is different to the other Controllers. It has 10 Digital ports and only 6 Analog ports. The Hedgehog Light has the most USB Ports, but no Micro-USB Port.

E. Miscellaneous

All the three Controllers do have some technical problems. With the Hedgehog Light we experienced some issues, while using the Botball Servos. If you do not include a waiting command for some milliseconds after the “moveServo” statement, it will not completely end the Servo movement. One problem for the KIPR Link is the battery charge display as it will not be very precise. We also managed to destroy the touchscreen of the Controller, once we tried to calibrate it. The Wallaby has several smaller issues such as an on/off switch you almost can’t reach with the finger, unreliable touchscreen or weak WiFi-signal sometimes.

F. Code Samples

Because the syntax for the Wallaby and the Link is equal, we only compare the code for Hedgehog Light to the one for the Link. Here is a very small example on how to activate two motors to drive forward then turn a servo to a position and turn right with the robot afterwards on the Link or Wallaby.

```

mav(1, 1000);
mav(0, 1000);
msleep(1500);

set_servo_position(1, 500);

mav(1, 1000);
mav(0, -1000);
msleep(500);

```

Fig. 5. KIPR-Link Code

On the next image, you can see how the same commands would look like for the Hedgehog Light.

```

hedgehog.move(0, 300)
hedgehog.move(1, 300)
sleep(1.5)

hedgehog.set_servo(0, True, 500)

hedgehog.move(0, 300)
hedgehog.move(1, -300)
sleep(0.5)

```

Fig. 6. Hedgehog Code

CONCLUSION

Altogether, every Controller has its benefits and is designed for their Usages. The Hedgehog Light is a very good Controller for Beginners, because of the Web IDE. On the other Hand, the Display of the KIPR Controllers allow to easily check Values from Servos or the Battery without a Computer. The KIPR Wallaby has of course more software opportunities than the Link, because it is the successor model.

ACKNOWLEDGEMENT

The authors would like to thank Dr. Michael Stifter for his support during the work on this publication. We would also like to thank the members of Team items for their useful tips and the PRIA Staff for documenting and creating Information Pages for their Controller, the Hedgehog.

REFERENCES

- [1] B. McDorman and J. Southerland, “A Look inside the KIPR Link”, http://files.kipr.org/gcer/2013/proceedings/McDorman_A_Look_Inside_the_KIPR_Link.pdf
- [2] KISS Institute for Practical Robotics, “KIPR Link Getting Started Manual”, http://www.kipr.org/sites/default/files/KIPR_Link_Getting_Started_Manual_0.1.pdf
- [3] R. Aroca, L. Goncalves, and P. Oliveira, “Towards smarter robots with smartphones”, Robocontrol 2012, 2012
- [4] KISS Institute for Practical Robotics, <http://www.kipr.org/kiss-web-wallaby-firmware>
- [5] J. Southerland, <https://github.com/kipr/kovan-fpga>, 2013
- [6] Koza C., Lepuschitz W., Wolff M., Frank D., Koppensteiner G. (2017) Hedgehog Light – A Versatile, White Box Educational Robotics Controller” https://link.springer.com/chapter/10.1007/978-3-319-55553-9_19

- [7] KISS Institute for Practical Robotics,
<http://www.kipr.org/hardware-software>
- [8] B. McDorman, N. Zaman, J. Villatoro,
<http://github.com/kipr/kiss>, 2013.
- [9] KISS Institute for Practical Robotics,
<http://www.kipr.org/products/link>
- [10] Score! By Pria,
<https://pria.at/research/score/>
- [11] 2017 GitHub, Inc
<https://github.com/PRIARobotics/HedgehogLightSetup/tree/master/doc/de>