



Ⓒ Operating Instructions

## Hexapod Robobug Complete Set

Item no. 1664151

CE

	Page
1. Introduction.....	4
2. Explanation of symbols.....	4
3. Intended Use.....	4
4. Package Contents.....	5
5. Product Description.....	5
6. Features.....	6
7. Safety Instructions.....	7
8. Required accessories.....	8
9. General notes.....	9
10. Preparatory work.....	10
a) Application of the screw locking lacquer.....	10
b) Ball bearings and cylindrical pins.....	10
c) Power Supply.....	11
d) Installation of the software and firmware.....	11
11. Transferring the Firmware.....	12
a) Installing the driver.....	12
b) Install Arduino IDE.....	12
c) Setting up Arduino IDE.....	12
12. Preparing the leg servos for installation.....	15
13. Assembling the mechanical parts.....	18
a) Mounting servos.....	18
b) Mounting of bearings.....	20
c) Mounting the servo disks.....	22
d) Mounting the spacer.....	24
e) Mounting the servos on the top plate.....	25
f) Mount bottom plate.....	27
g) Mounting the Receiver and Hexapod Robot Board.....	29
h) Mounting the legs.....	32
i) Laying the cables.....	35
j) Installing the battery.....	39
k) Mounting the rubber feet.....	40
l) Setting jumpers.....	40
14. Transferring the Firmware.....	41
15. Controlling with the Gamepad.....	42
16. Calibrate legs.....	44
17. Using user boards.....	47
18. Demo Programmes.....	50
19. Overview of the connections and components of the robot board.....	53
20. Schematic system overview (block diagram).....	55
21. Pan unit.....	56
22. Pin assignment of the user boards.....	57
23. Parts list.....	58
24. FAQ.....	63
25. Cleaning.....	64
26. Declaration of Conformity (DOC).....	64

	<b>Page</b>
27. Disposal.....	64
a) Product.....	64
b) Batteries/Rechargeable Batteries .....	64
28. Technical Data .....	65
a) General information.....	65
b) Control electronics .....	65
c) Gamepad .....	65

# 1. Introduction

---

Dear customer,

Thank you for purchasing this product.

This product meets the requirements of current European and national guidelines.

We kindly request the user to follow the operating instructions to preserve this condition and to ensure safe operation!



This user manual is part of the product. It contains important information on starting up and handling the device. Bear this in mind if you pass this product to a third party.

For this reason, keep these operating instructions for future reference!

All names of companies and products are the trademarks of the respective owners. All rights reserved.

If there are any technical questions, please contact:

International: [www.conrad.com/contact](http://www.conrad.com/contact)

United Kingdom: [www.conrad-electronic.co.uk/contact](http://www.conrad-electronic.co.uk/contact)

## 2. Explanation of symbols

---



An exclamation mark in a triangle indicates important instructions in this operating manual which must be adhered to.



The arrow symbol indicates specific tips and advice on operation.

## 3. Intended Use

---

The "Hexapod Robobug" complete set is solely designed for training, research and private use in the hobby or model construction area and the operating times associated with it.

The product is not a toy and should be kept out of reach of children under 14 years of age.

The product is aimed at advanced users who already have experience with Arduino and the programming language C/C++ and also in electronics as well as in the construction of mechanical kits.

The product must not get damp or wet! The electronics are designed for operation at an ambient temperature between 0 °C and +40 °C. The electronics are designed for use in a wide range of applications.

Observe all the safety instructions in these operating instructions. They contain important information regarding the handling of the product. You are the sole responsible for the safe operation of the Hexapod Robobug complete set!

Mechanical components as well as servo motors are subject to natural wear and can wear during long operating times. The servos can be reordered as spare parts. The servo motors have a high cycle stability of up to 100,000 cycles. However, the Robobug is not designed to carry out permanent movements. The motors in the servo require a break of at least 10 minutes after a longer run (about 10 minutes of continuous movement) in order to cool down. This ensures a long service life.

If you use the product for other purposes than those described above, the product may be damaged. Moreover, improper use involves risks such as short-circuits, fire, electric shocks, etc. Please read the operating instructions carefully and do not discard them. Please include these operating instructions when you pass the product on to a third party.

## 4. Package Contents

---

- Hexapod Robobug Mechanical Kit
- 18x Servos
- Hexapod Robot Board
- Gamepad for control
- Small parts (e.g., jumpers, battery plug XT30)
- Quick guide

### Latest Operating Instructions

Download the latest operating instructions via the link [www.conrad.com/downloads](http://www.conrad.com/downloads) or scan the QR code shown here. Follow the instructions on the website.

On the product website you will find all sample programmes, firmware updates and further instructions for extensions.



## 5. Product Description

---

A running robot for beginners as well as experienced developers: With the "Hexapod Robobug" complete set, consisting of electronics, mechanics and digital high-performance servos, you can build your own Hexapod running robot and enter the exciting world of robotics!

The "Hexapod Robobug" complete set is the ideal basis for entering the fields of electronics, mechanics and programming: it enables a large number of extensions by using additional user boards such as Arduino, Raspberry Pi or NodeMCU and offers space for many additional components.

The robot is designed in such a way that it can be controlled without programming using the controller supplied, similar to an RC model. Thanks to its expandability and flexibility, the robot is perfect for hobby, research, schools, education and training and encourages independent development and expansion.

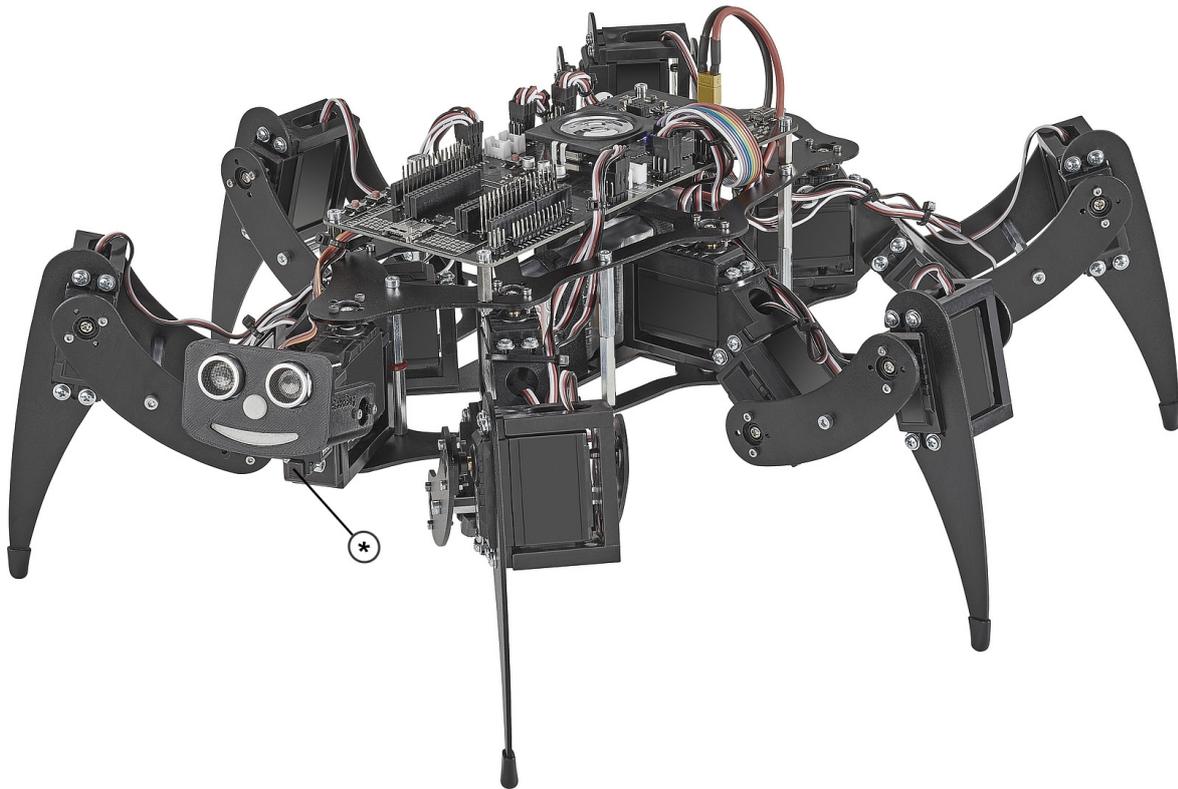


Figure 1

→ (\*) Not included in the delivery:

The 3D print data of the Pan & Tilt unit and the required components can be found on the product's website. The product also requires a rechargeable battery and a suitable charger.

## 6. Features

---

- Simple construction
- Easy programming with Arduino™ (Arduino™ library available)
- Firmware based on Arduino™
- Onboard locomotion controller ATmega2560 for leg control (Arduino™ MEGA compatible)
- 18x high performance digital servos with metal gear, double ball bearings
- High-quality aluminium and plastic parts
- Legs with double ball bearings
- Control via gamepad (included in delivery) and user board commands
- Easily accessible connections for your own expansions
- Serial data connection between user board and locomotion controller
- Compatible with the following additional boards: Arduino, NodeMCU, SBC (e.g., Raspberry Pi)
- USB programming interface for Locomotion controller
- Loudspeaker with amplifier for sound output
- MicroSD card reader
- Infrared receiver
- Freely usable buttons
- Power management
- I<sup>2</sup>C Sseeed Grove connector for extensions
- Automatic level adjustment for 5 V and 3.3 V user boards
- Versatile expandability thanks to open architecture and open source software
- Complete kit to set up the robot (batteries, rechargeable batteries and charger not included)

# 7. Safety Instructions

---



Please read the operating instructions carefully and pay particular attention to the safety instructions. We do not assume liability for any injuries/material damages resulting from failure to observe the safety instructions and the information in these operating instructions regarding the proper use of the product. Furthermore, in such cases, the warranty / guarantee will be null and void.

Dear customer,

These safety instructions are not only for the protection of the product, but also for your own safety and the safety of other people. Therefore, please read this chapter very carefully before using the product for the first time!

- The product is aimed at advanced users who already have experience with Arduino and the C/C ++ programming language as well as in electronics and in the construction of mechanical kits. If you do not have sufficient experience, please contact an experienced developer, a local builders club or our support.
- For safety and approval reasons, unauthorised conversion and/or modification of the product beyond what is described, is not permitted.
- The product is not a toy. Keep out of the reach of children and pets.
- Do not carelessly cast aside the packaging material. It may become a dangerous plaything for children.
- Protect the product from extreme temperatures, direct sunlight, strong vibrations, high humidity, moisture, combustible gases, vapours and solvents.
- Never expose the product to mechanical stress.
- When connecting the servos, other components and their connecting cables, ensure that the contacts are secure. Loose or wobbly connectors may cause interference or damage.
- If soldering is required, be sure to avoid short circuits occur during soldering. Before you perform a soldering operation, all parts must have zero potential.
- To avoid short circuits, make sure that components or solder contacts do not come into contact with metal parts, when installing the board. Short circuits will ruin the product and void the warranty!
- If safe operation is no longer possible, take the device out of service and secure it against unintended use. Safe operation is no longer possible, if the product:
  - shows visible damage,
  - no longer functions properly,
  - has been stored under adverse ambient conditions for an extended period of time or
  - has been exposed to considerable strain during transport.
- Please handle the product with care. The product can be damaged if crushed, struck or dropped, even from a low height.
- If you have doubts about how the equipment should be operated or how to safely connect it, consult a trained technician.
- Maintenance, adjustment and repair work should only be carried out by an expert or a specialised workshop.
- If you have any questions that are not answered in these operating instructions, please contact our technical customer service or other professionals.

## 8. Required accessories

---

- NiMH battery; 5 cells, min. 3700 mAh (e.g., Conrad item no. 1784857)
- Charger (e.g., Conrad item no. 1413029 or 1413030)
- Adapter cable XT30 (e.g., Conrad item no. 1785258)
- Batteries (2x AAA/Micro) (e.g., Conrad item no. 658010)
- Screw locking (e.g., Conrad item no. 826389)
- Mini-USB cable (e.g., Conrad item no. 1365369)

### Optional:

- Servo for Pan unit \* (e.g., Conrad item no. 1762878)
- Servo for tilt unit \* (e.g., Conrad item no. 1762877)
- Arduino UNO (e.g., Conrad item no. 191789)
- Screw locking (e.g., Conrad item no. 1613301)

—> \* The Pan & Tilt unit is the head mechanism (see Figure 1) and can optionally be printed on a 3D printer.

### Required tools and material:

- Small side cutter (electronic side cutter)
- Scissors
- Open-end wrench SW 5.5 mm
- Open-end wrench SW 4 mm
- Phillips screwdriver PH1
- Phillips screwdriver PH2
- Hexagon socket 6 mm
- Hexagon wrench 1.5 mm
- Hexagon wrench 2.5 mm
- Screw locking (medium strength) e.g., LOCTITE® 243 (Conrad item no. 1370555)
- 90° Stop angle or straight edge
- Work paper

## 9. General notes

---

- Take enough time for the assembly. Too much hurry leads often to mistakes which can damage the components or cancel out the time gained by extensive reworking.
- The workplace should be sufficiently large and clean so that the various components and assemblies can be easily removed and assembled.
- It is essential that you observe the pictures during assembly. Here the assembly locations and the correct alignment of the components are shown.
- All mechanical components of the kit are manufactured very precisely. Do not use force during assembly under any circumstances. All parts can be assembled without much effort. If this is not the case, reconsider the assembly step again and read through the corresponding description in this manual again.
- When tightening the screws, be careful not to tighten them too tightly. This can lead to damage to the plastic parts, loss of warranty/guarantee!
- The screws are already delivered pre-sorted in small bags. You can already open these before assembly and place them sorted by type on the worktable. This facilitates assembly, as you do not have to search for individual screws.
- Some screws, nuts and other small parts are included in the scope of delivery in larger quantities than is needed. These serve as a replacement if, for example, a screw or similar is lost during assembly.
- The Pan & Tilt unit is the head mechanism (see picture 1) and can optionally be printed on a 3D printer. You will find the data for this as a download on the product's website (as well as all other available downloads).
- The Locomotion-Controller is located under the loudspeaker. It is responsible for leg control and is programmed via the mini-USB connection. An overview of the board can be found in Chapter 19 in Figure 79.

# 10. Preparatory work

→ Before the robot is assembled, some parts are prepared. This makes the actual assembly easier and faster.

## a) Application of the screw locking lacquer

In the following instructions, some screws should optionally be secured with a screw lock. This is pointed out accordingly in the text. We recommend that you only apply the screw lock after the robot has been completely assembled and calibrated. Afterwards the screws to be secured are removed again and provided with safety lacquer. When assembling and calibrating the legs, it might be necessary to loosen them again for adjustment tasks.

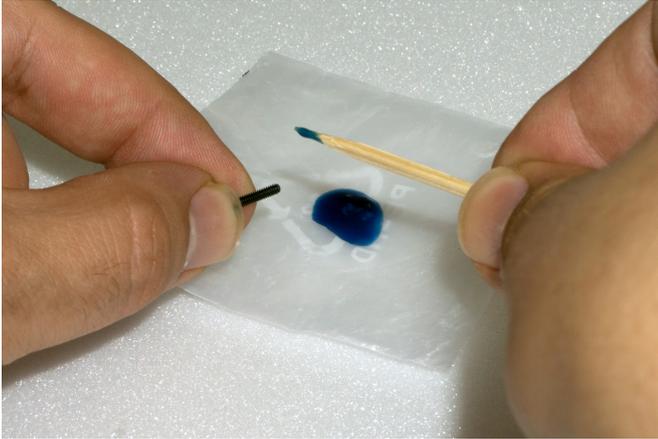


Figure 2

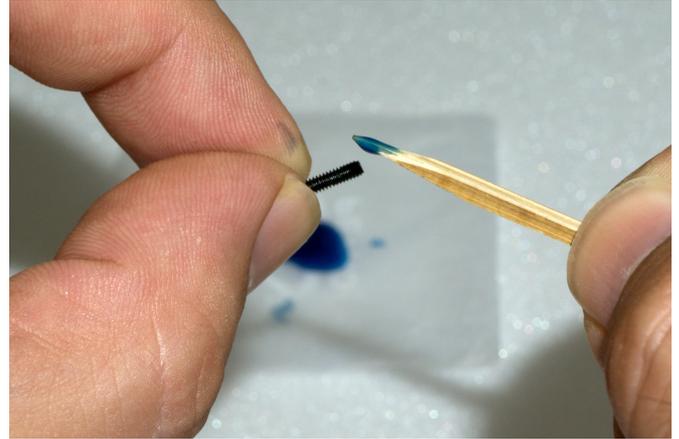


Figure 3

The easiest way to apply the screw lock adhesive is to apply a larger drop of the screw lock onto a base (e.g., screw set). With the help of a toothpick or a small screwdriver, the screw lock can be proportionally applied to the thread. Do not use too much screw lock, a small drop is sufficient.

## b) Ball bearings and cylindrical pins

The leg servos are additionally stabilised by ball bearings with pressed-in cylindrical pins. This significantly reduces the mechanical load on the servos and significantly increases the service life of the servos. The ball bearings are supplied pre-assembled. If a pin is pushed out of the ball bearing during assembly, it can be pushed back again. If the pin is no longer firmly seated in the ball bearing, you can solve the problem with a small trick.

Notch the pin at the point where it sits in the ball bearing with a side cutter, see picture 4.



Figure 4

Then push the pin back into the ball bearing (if necessary, carefully knock it in with a small hammer).

Figure 5 shows the ball bearing with the cylindrical pin.

→ Brown marks on the ball bearing are not rust! They represent grease and glue, which are used with the assembly of the two components in the manufacturing. You can remove these residues simply with alcohol (spirit).



Figure 5

## c) Power Supply

In assembling the Robobug you will need the battery for the power supply, in order to be able to move the leg servos in the middle position, to align the legs and later to install the firmware, which is responsible for running the robot in the next work steps. For all this work, the Robobug needs a power supply.

The right power supply is crucial for the safe and trouble-free operation of the Robobug. The Robobug is designed for a NiMH battery with 5 cells and a capacity between 3500 - 5000 mAh.

We recommend using the optionally available rechargeable battery. This can be used directly with the Robobug without soldering.

### Prepare your own battery

The "Hexapod Robobug" complete set comes with a type XT30 battery plug. This is soldered to the battery selected by you. The cable length between battery and plug must not exceed 20 cm! For the connection cable, we recommend using a highly flexible silicone cable with a cross-section of between 2 and 2.5 mm<sup>2</sup>.

Pay attention to the polarity (plus/+ and minus/-), and that no short circuit is caused by soldering.



In the event of a short-circuit, rechargeable batteries can emit a lot of current, which can lead to burns and fire damage as well as to dangerous injuries. An explosion of the battery is also possible!

The ready-made battery is provided with a battery shrink tube. This way the battery looks clean at the end of the soldering work and indicates that there is no danger of a short circuit.

In figure 7 you can see how the battery is plugged into the Robobug. The plug is protected against reverse polarity and cannot therefore be plugged in the wrong way.

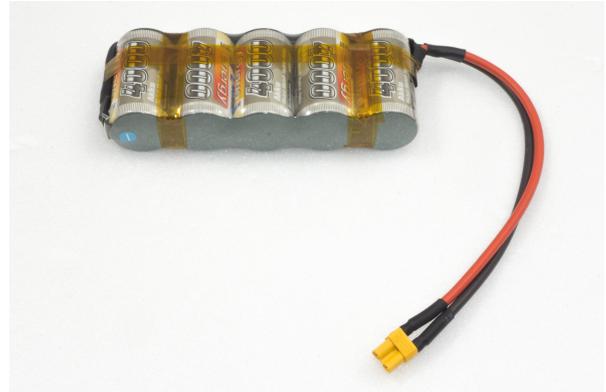


Figure 6

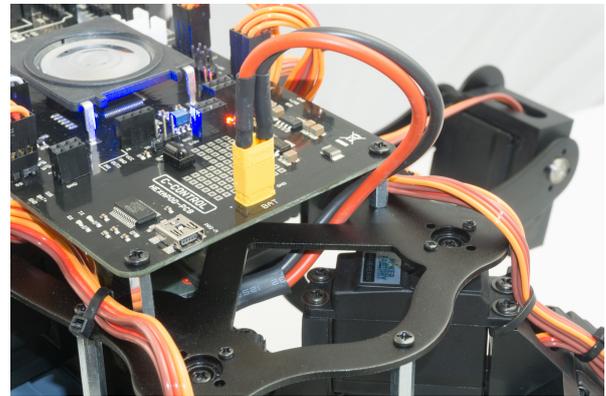


Figure 7

## d) Installation of the software and firmware

The software packages, firmware and tools required for the "Hexapod Robobug" are available online as a download package. This package is updated and extended regularly. Please check occasionally whether a new version or useful extensions are available.

Open your web browser and navigate either via our shop website [www.conrad.com](http://www.conrad.com) to the product page of the "Hexapod Robobug" complete set, or visit our download centre [www.conrad.com/downloads](http://www.conrad.com/downloads) directly.

Here you can download the package "001664151-up-01-en-DOWNLOAD\_BUNDLE\_Vx\_x" (x\_x is the version of the package). The highest version number corresponds to the latest version! Unpack the downloaded ZIP file on your hard disk.

### Here is a short explanation of the included folders:

"Arduino"	Here is a text file with the download link for the software Arduino™ IDE included. This is needed, for example, to install the firmware without an updater.
"Datasheets"	Data sheets and sketches
"Driver"	Driver for the FTDI USB chip
"Library demos"	Contains the Arduino library, demo's and the Locomotion firmware as a ZIP file. Will be installed in the Arduino IDE.
"Schematic"	Schematic of the Hexapod Robot Board as PDF file
"Terminal"	Calibration software for setting the Hexapod

# 11. Transferring the Firmware

On delivery, no firmware is installed on the Hexapod Robot Board. The firmware must first be transferred during initial commissioning. This is done via the Arduino IDE.

The following description shows you the procedure for transferring the firmware. In the following chapters you will transfer different programmes to the Hexapod Robot Board. The procedure is identical except for the selection of the respective program.

→ When transferring the firmware, the Hexapod Robot Board must be connected to the power supply (battery)! Make sure that the battery is fully charged before transferring the firmware or programmes.

## a) Installing the driver

Connect the "PRG-M" port of the Hexapod Robot Board to a free USB port on your computer. Windows will now attempt to install a new driver. Typically, Windows will automatically download and install the drivers from the Internet, since the driver for the "FTDI" USB bridge chip used is available in the Windows driver download (the computer must be connected to the Internet for this to work).

If this method does not work, refer to the driver manually during installation. The driver is located in the unpacked download bundle, in the "Driver" directory.

After installing the driver, check in the device manager whether the driver for the Hexapod Robot Board has been installed and which COM port number has been assigned. Then select them in the Arduino IDE.

You can also download the drivers directly from <http://www.ftdichip.com>.

## b) Install Arduino IDE

You can download the latest version of the Arduino IDE from [www.arduino.cc](http://www.arduino.cc). The Arduino IDE is available as "installer version" and as "ZIP archive". Both versions can be used.

With the installer version, you install the Arduino IDE as you would install any other programmes. With the ZIP version, you must unpack the ZIP archive to the desired location on your computer after the download. Then start the programme by double-clicking on the file "arduino.exe".

→ The current firmware version was developed with Arduino version 1.8.3. If there are problems with newer versions, download an older version of the Arduino IDE from [www.arduino.cc](http://www.arduino.cc) and transfer the firmware with it.

Also familiarize yourself with the board and its connectors. An overview of the board can be found in Chapter 19 Figure 79.

## c) Setting up Arduino IDE

Start the Arduino IDE and follow the instructions in this manual.

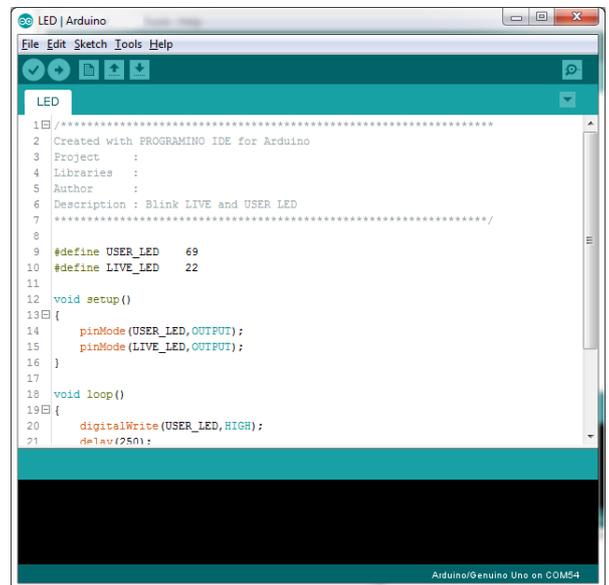


Figure 8

Now install the Hexapod library. It can be found in the download bundle under "Library-Demos\Maker-Factory-Hexapod-V1.1.zip". Select the menu item "Sketch\Include Library\Add .ZIP Library..." in the Arduino IDE.

→ If you update the library later, you have to delete the old library first, otherwise the Arduino IDE will report an error!

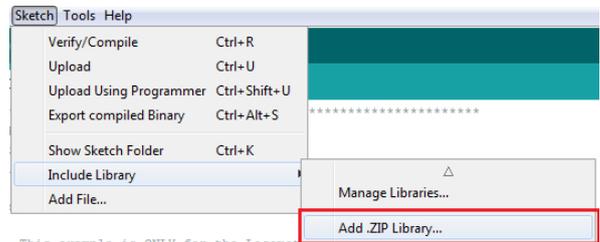


Figure 9

After you have installed the library, you will find two folders named "Locomotion" and "User-Boards" in the menu item "File\Examples\Maker-Factory-Hexapod-Vx.x".

The folder "Locomotion" contains programmes for the Locomotion controller, which is fixed on the Hexapod Robot Board and is responsible for the operation.

The folder "User-Boards" contains programmes for the Arduino compatible user boards like Arduino UNO and NodeMCU.

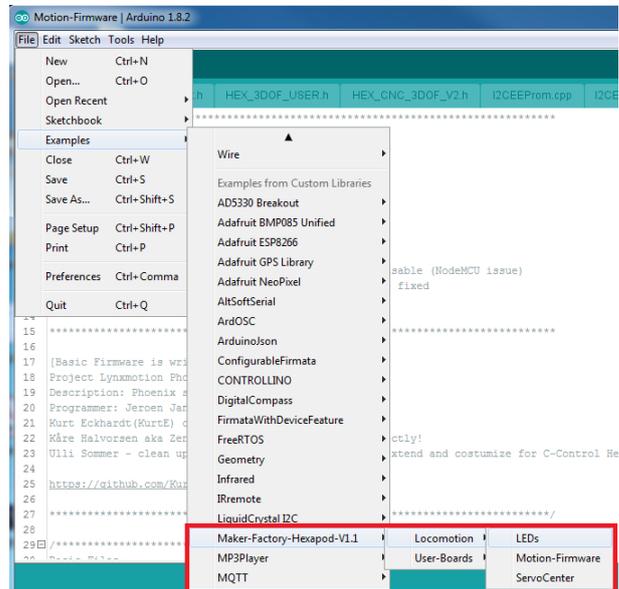


Figure 10

Select the microcontroller used under the menu item "Tools\Board\Arduino/Genuino Mega".

The locomotion controller is compatible with the Arduino Mega 2560.

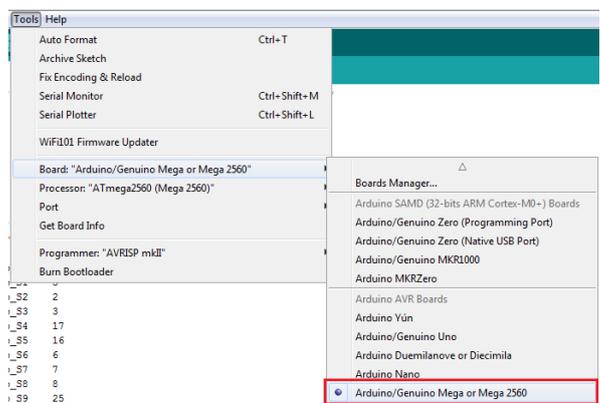


Figure 11

Select the COM port under "Tools\Port" that was assigned to the Hexapod Robot Board during driver installation.

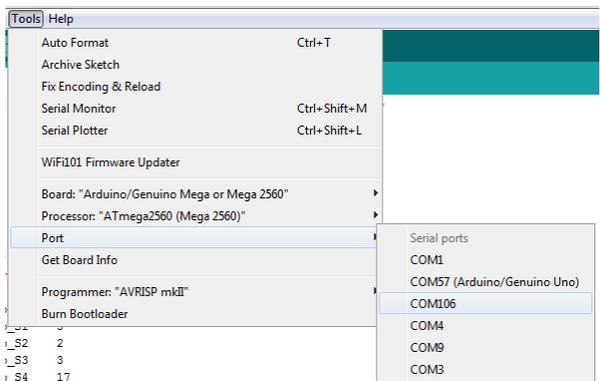


Figure 12

To test if everything works, transfer the sample programme "\User-Board\Locomotion\LEDs". After successful transmission, the programme causes the blue "LIVE-LED" and the red "USER-LED" to flash alternately.

→ To transfer the programmes, the battery must be plugged into the Hexapod Robot Board!

After selecting the "LEDs" programme (also called Sketch in Arduino) a new Arduino IDE window opens with the "LEDs" Sketch.

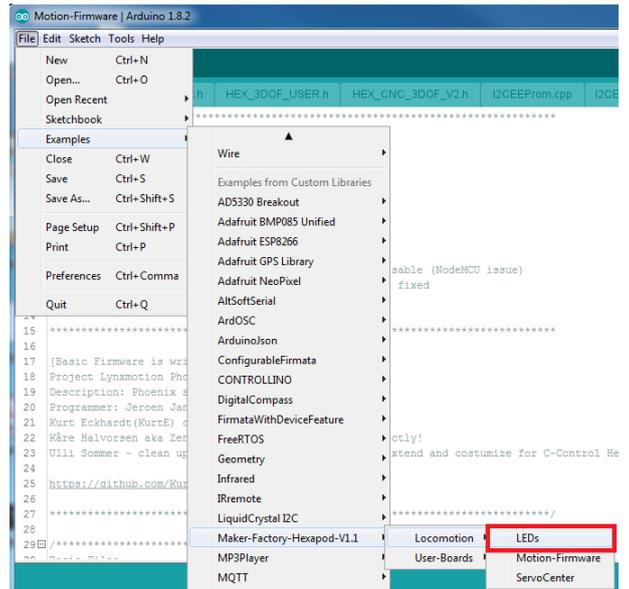


Figure 13

With the "arrow to the right" symbol (see picture 14) in the menu you transfer the programme to the Hexapod Robot Board. Alternatively, you can also use the shortcut "CTRL+U".



Figure 14

This procedure applies to all programmes to be transferred (called sketches in the Arduino IDE) that are intended for the locomotion controller. For the user board samples, you only change the applied user board in use and the "COM-Port".

→ Please read the respective initial comment in the source code of the examples before transferring. This contains important information about the program!

If the transfer does not work, please check your settings for board and connection and also check if the board has been entered and installed correctly in the device manager and if the power supply is connected.

Here you can find more information about installing the Arduino IDE: <http://arduino.cc/en/Guide/Windows>

Leave the board plugged in for the next step, as you will need to move the leg servos to the centre position in this step.

## 12. Preparing the leg servos for installation

Before you install the leg servos, you must move them to the centre position. This means moving the servos to the mechanical middle position using the Hexapod Robot Board and the "ServoCenter" program. This is the starting position for assembly.

First unpack all 18 servos; then straighten all connecting cables.

Pin assignment:

Labelling in Figure 15	Function	Imprints on PCB
P	Control pulse	SIG
+	Positive pole/+	BAT
-	Negative pole/-	GND

To set the leg servos, transfer the "ServoCenter" programme (located in the "Locomotion" directory) to the Hexapod Robot Board. The programme automatically moves the servos from "S0" to "S17" to the middle position.

Connect the servos to the pin headers as shown in the figure (pay attention to the polarity!). All servo connections (S0 to S17) of the robot board are controlled with a pulse width of 1500  $\mu$ s.

At this point you do not have to connect all 18 servos to the Hexapod Robot Board at the same time. In practice, it has proven to be much easier and faster to use only one connection (e.g., S0) and then move one servo after the other to the middle position.

The brown wire of the servo cable points to the edge of the board. The board is also labelled SIG (pulse), BAT+ (+), GND (-).

→ The battery must be connected for this procedure!

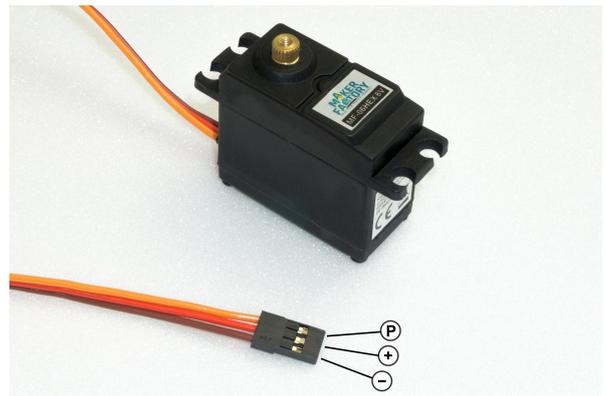


Figure 15

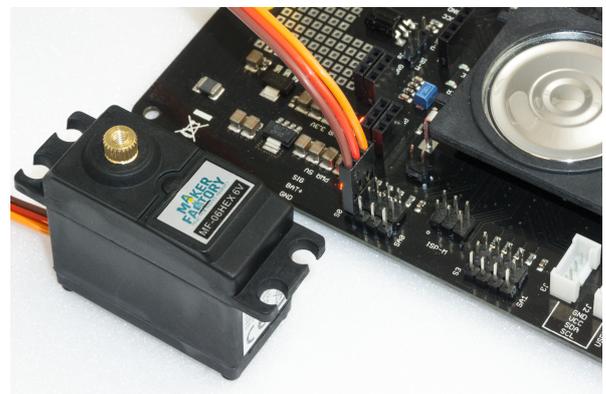


Figure 16

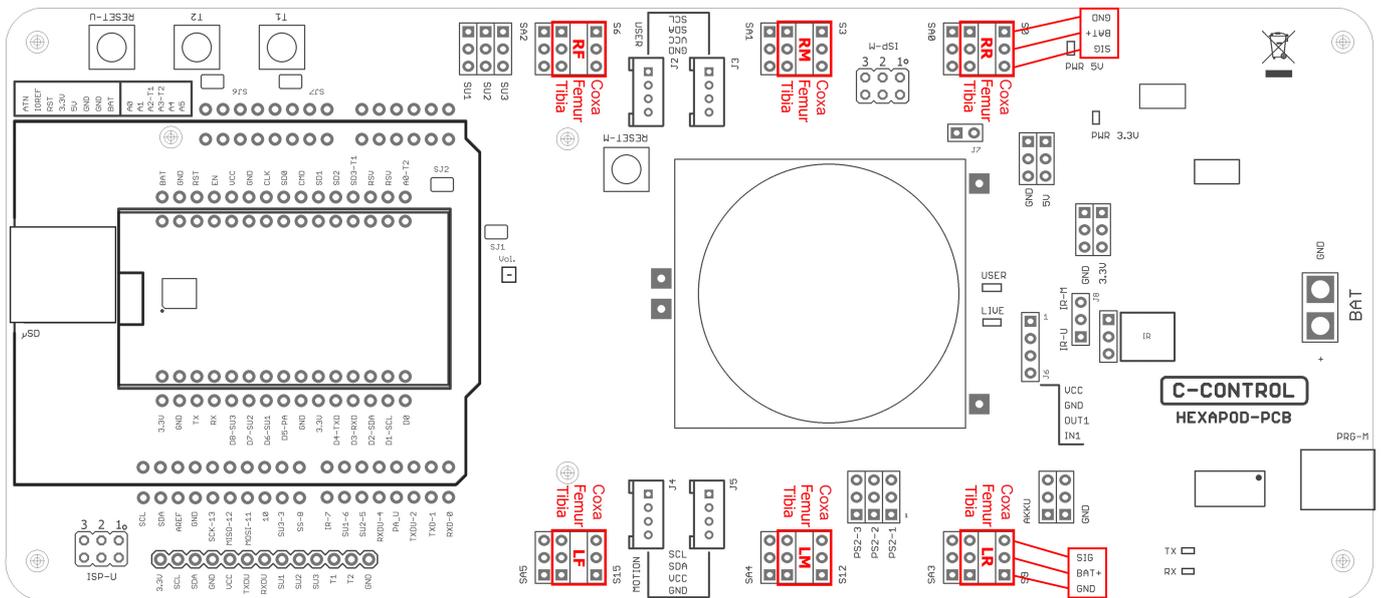


Figure 17

Figure 17 shows the pin headers to which the servos are connected.

→ The pin strips labelled SA0 to SA5 do not carry a control signal. This connection remains unassigned and can later be used for proper extensions such as leg contact sensors.

The joint details are given anatomically in Latin and have the following meaning:

- Coxa = hip
- Femur = thigh
- Tibia = leg

The connections are also indicated in Figure 17 with the leg name as an abbreviation:

- RR = Right rear leg ("Right Rear")
- RM = Right Middle leg ("Right Middle")
- RF = Right front leg ("Right Front")
- LR = Left rear leg ("Left Rear")
- LM = Left middle leg ("Left Middle")
- LF = Left front leg ("Left Front")

→ Leave the programme on the Hexapod Robot Board until the robot is completely assembled. If you accidentally twist a servo mechanically during assembly, you can plug it in again and align it.

If the servo is turned from the middle position during installation, this has a negative effect on the running characteristics. This can also damage the servo and the mechanics.

Disconnect the robot board from the power supply in case you won't be using it for a long time to avoid accidental short circuits!

### Functional description of a servo

Servos have electronics, a motor and a potentiometer.

The potentiometer picks up the mechanical position of the servo and forwards it to the servo electronics.

The electronics in the servo compares the given position (nominal value, in our case the pulse width given by the locomotion controller) with the pulse width of the servo electronics (set via the potentiometer, actual value).

If there is a difference between the two pulse widths, the servo electronics regulates the motor to the target position.

The graphic illustrates the pulse/pause ratio required to control a servo. In this case, the pulse has a length of  $1.5 \text{ ms} = 1500 \mu\text{s}$ , which corresponds to the centre position of the servo.

A servo receives as control signal a positive pulse with a pulse width between  $500 \mu\text{s}$  (0.5 ms) and  $2500 \mu\text{s}$  (2.5 ms). The total period duration of the control signal is 20 ms.

When the servo is in the middle position, the pulse width is  $1500 \mu\text{s}$  (1.5 ms).

If the pulse width is less than 1.5 ms, the servo rotates in one direction. If the pulse width is greater than 1.5 ms, it rotates in the other direction. The angle of rotation is proportional to the pulse width, i.e. the longer or shorter the pulse, the more or less the servo rotates in the corresponding direction.

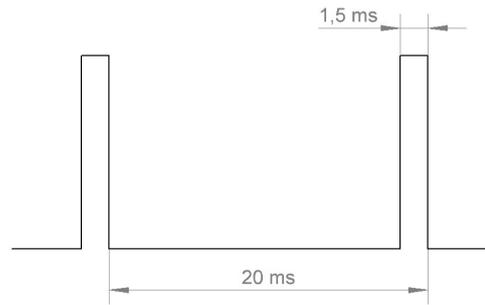


Figure 18

# 13. Assembling the mechanical parts

The following illustrations illustrate the mechanical assembly.



### Caution!

When tightening the plastic screws, make sure that they are not screwed too tightly into the plastic and that the plastic holders are not damaged!

If you remove a screw from a plastic part and then screw it in again, first turn the screw anticlockwise until you notice that the screw "engages". At this point the screw is "engaged" in the existing thread. The screw can then be easily screwed into the thread that has already been created during the first screwing in. If the self-tapping plastic screw is screwed in without using this method, a new thread may occur. This weakens the material and the tensile strength as well as the hold are no longer guaranteed.

## a) Mounting servos

Mount the servos as shown in the following illustrations. You will need 3 pairs of left and 3 pairs of right legs.

Most aluminium parts have a slightly rounded side due to the manufacturing process (punching process). You can use the parts in both directions during assembly. The nicer side should point forward in the direction of travel. The rounded side of the top and bottom plates should point in the same direction (top or bottom).

**Required tool:** 1x PH2 Cross Screwdriver

The screws are packed in plastic bags in the kit. Figure 19 shows the type of screws used for servo mounting.

A total of 6x legs must be assembled (3 left and 3 right).



Figure 19

Figure 20 shows a leg of the Robobug. The leg elements are called "Coxa", "Femur" and "Tibia".

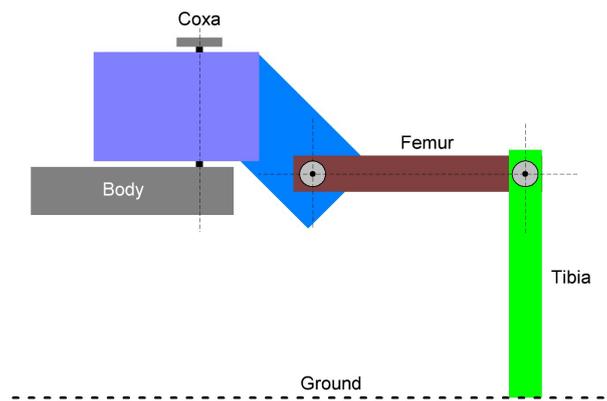


Figure 20

Place the servo in the plastic U-angle and the aluminium leg thereon (pay attention to the left and right leg and the round edge).

The servo cable is pushed through the oval opening in the U-angle.

Screw the leg together with four plastic screws.

When screwing the parts together, align them so that they are centred in the U-angle, the aluminium legs lie against the servo and ensure that nothing is strained or warped. If necessary, loosen the screws a little to relax the mechanics and then carefully tighten the screws crosswise.



Figure 21

The Coxa servo and femur servo holder must be mounted 6 times (3 left and 3 right).

First insert a servo into the holder and screw it on. Only then is the second servo plugged in and screwed in.

The servo cable of the Coxa servo is inserted through the round opening at the top and the servo cable for the femur servo is inserted through the large oval opening as with the U-angle mounting.

Screw the servos together with four plastic screws.

When screwing the servos together, align them so that they are centred in the plastic holder and are not tensed. If necessary, loosen the screws a little to release the mechanical tension and then carefully tighten the screws crosswise.



Figure 22



Figure 23

## b) Mounting of bearings

**Required tool:** 10 mm hexagon socket (from a socket wrench set) or comparable tool as a tool for mounting the bearings

Push the bearings into the aluminium parts as shown in the following illustrations. Here a hexagon socket wrench or a comparable tool can be helpful. Do not use rough force, this can damage the base plate and the ball bearings! Avoid tilting, with some sensing the position can be pressed in without great resistance.

→ Make sure that the edge of the bearing is flush with the aluminium part.

If a bearing cannot be pressed in, it may be caused by a protruding burr or too much paint. You can easily rework the bore with a knife (e.g., cutter knife for wallpaper or floors) or with a file. If the bore has become too large, the bearing can be glued to the flange of the bearing with a small drop of glue (e.g., Uhu® Endfest or comparable).

The bearings may come loose during final assembly due to uneven pressing of the cylindrical pins into the plastic holders. This is, however, not too bad, as the pins can easily be pushed back into the intended hole.

Sticking should be a last resort. The bearings clamp sufficiently strongly in the holes for the operation of the robot even without glue.

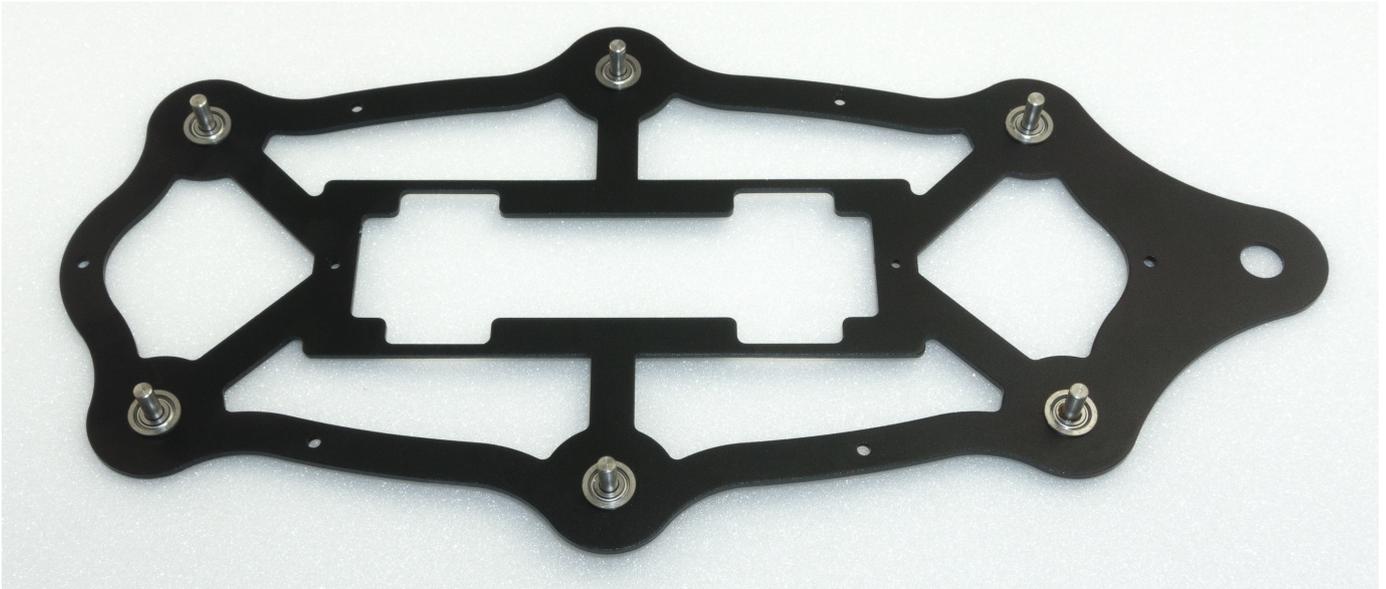


Figure 24

→ Figure 24 shows the bottom plate from above!

Figure 25 shows how you can press the bearings into the base plate using the femur aluminium parts as a support.

The hexagon wrench must press against the edge of the bearing. Masking the aluminium parts with painter tape protects them from scratching. Other parts that can be found in the hobby workshop can also be used as underlay.

→ Do not use adhesive tape that cannot be removed without leaving a residue.

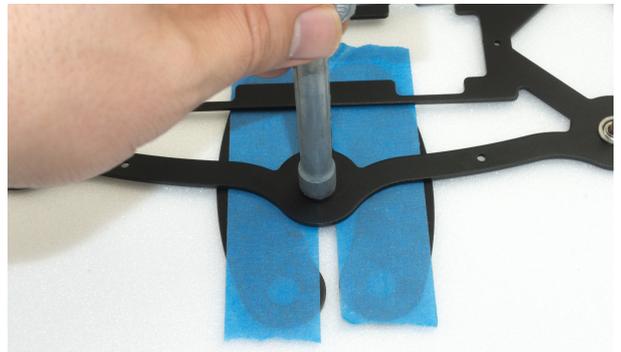


Figure 25

The femur counter bearing must be assembled a total of 6 times.



Figure 26

Figure 27 shows how you can push the bearing into the aluminium part by hand. The bearing should snap into the bore provided with a bold push.



Figure 27

Make sure that the edge of the bearing is flush with the aluminium part.



Figure 28

### c) Mounting the servo disks

Mount the servo disks as described in the following illustrations. Use two M2 screws + nut for each servo disk. The other holes remain free!



These screws (all M2) are now already provided with a screw lock.

**Required tool:** PH1 Phillips screwdriver  
Small flat nose pliers or wrench for M2 nut (SW4)  
Screw locking (medium strength)

#### Mounting the servo discs on the upper plate:

→ Figure 29 shows the top plate from above!

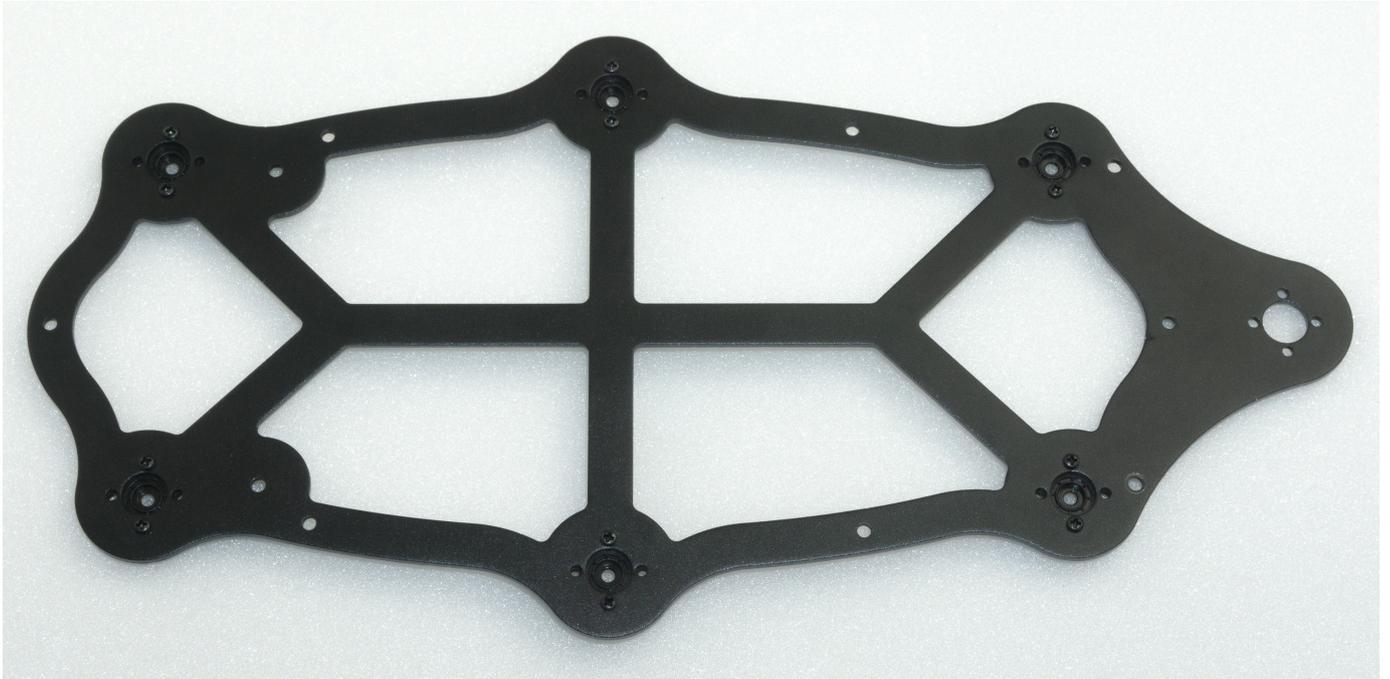


Figure 29

→ Figure 30 shows the top plate from below!

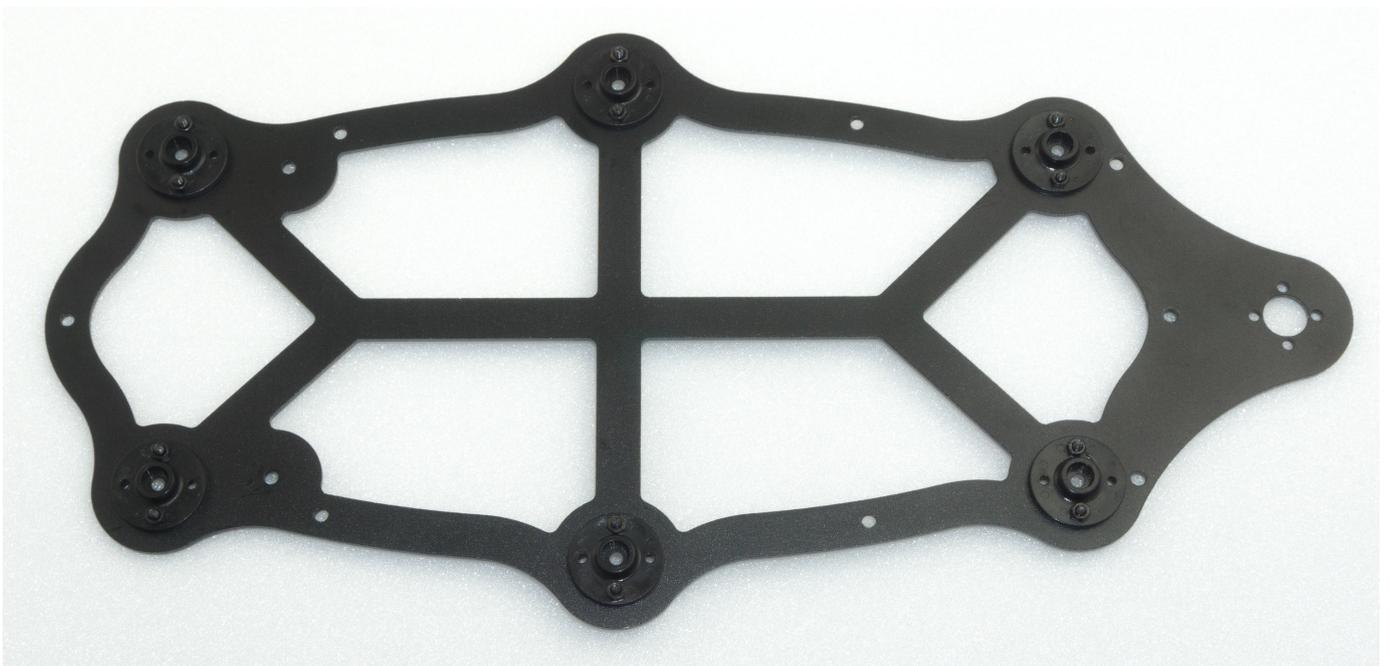


Figure 30

The femur servo mount must be mounted a total of 6 times.



Figure 31



Figure 32

#### d) Mounting the spacer

Next, screw together the spacers (30 mm long) for the legs, top and bottom plates as shown in Figure 33.

A total of 12 screwed spacers are required.

**Required tool:** 2x SW 5.5mm open-end wrench



Figure 33

Then fit a screwed spacer to each femur counter bearing. Insert the threaded side through the counter bearing and fix it with an M3 nut. It is sufficient if the spacers are only tightened by hand.

Do not tighten these screws until the assembly is complete. This way you still have the possibility to easily adjust everything during assembly.



Figure 34

Do the same with the base plate.

→ Figure 35 shows the bottom plate from above!

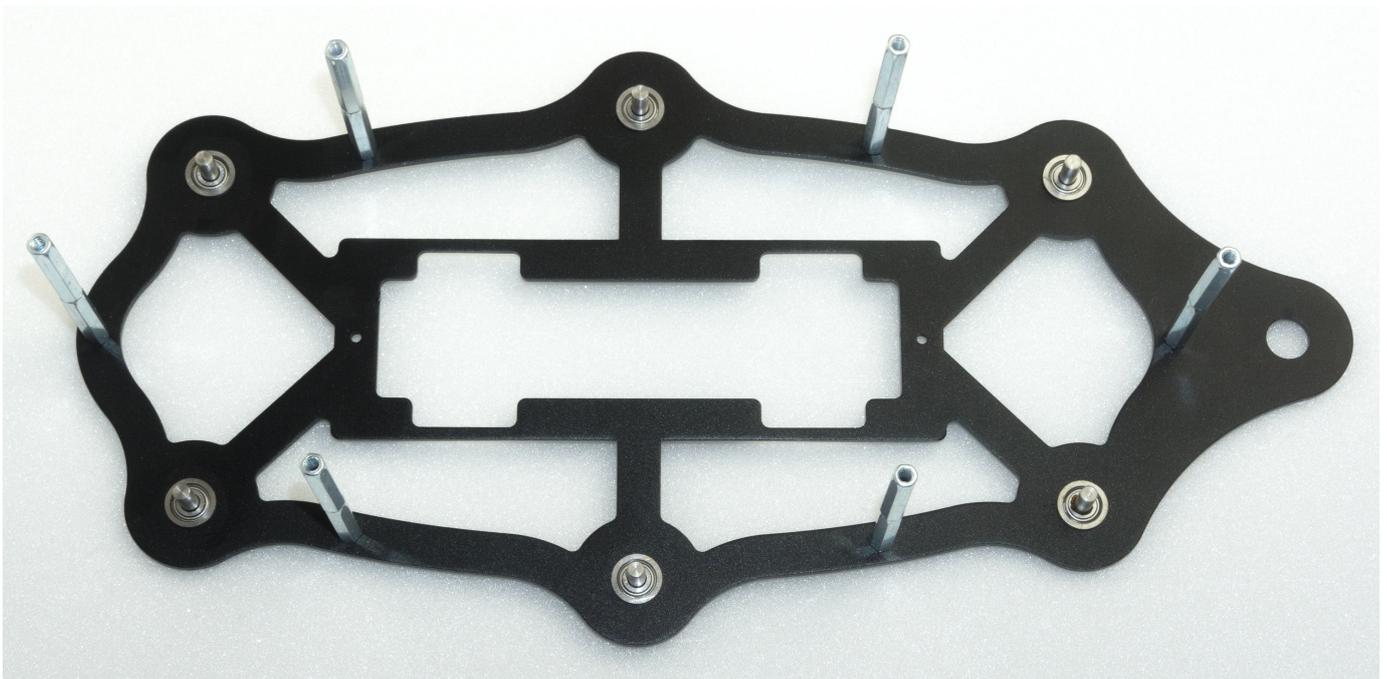


Figure 35

### e) Mounting the servos on the top plate

Mount the servos with the M3 screws (M3 x 6 mm, large head diameter) as shown in Figure 36.

→ For mounting the servos with the servo discs, always use the M3 x 6 mm screws (servo mounting screws) with the larger 6 mm head is used. The M3 x 6 mm screws with the smaller 5 mm head are used for the spacers!

Insert the servos into the teeth of the previously mounted servo discs. Proceed carefully so that the servos do not twist during insertion. If it should happen that a servo has rotated from the middle position during assembly, then bring the servo back to the middle position with the aid of the Hexapod Robot Board and the "ServoCenter" software. This procedure has already been described in Chapter 12.

If, due to the teeth of the servo disk or the general mechanical tolerances, you are not yet able to align the legs too precisely, this is not a problem. Adjust the leg as mechanically as possible, as shown in the illustration. An exact calibration is done after the assembly with the help of the software.

→ The servo mounting screws (PH2 (M3 x 6 mm, large head diameter) are not yet equipped with a screw lock! Only when the robot has been completely assembled and calibrated can these screws be removed and secured!

**Required tool:** PH2 Phillips screwdriver

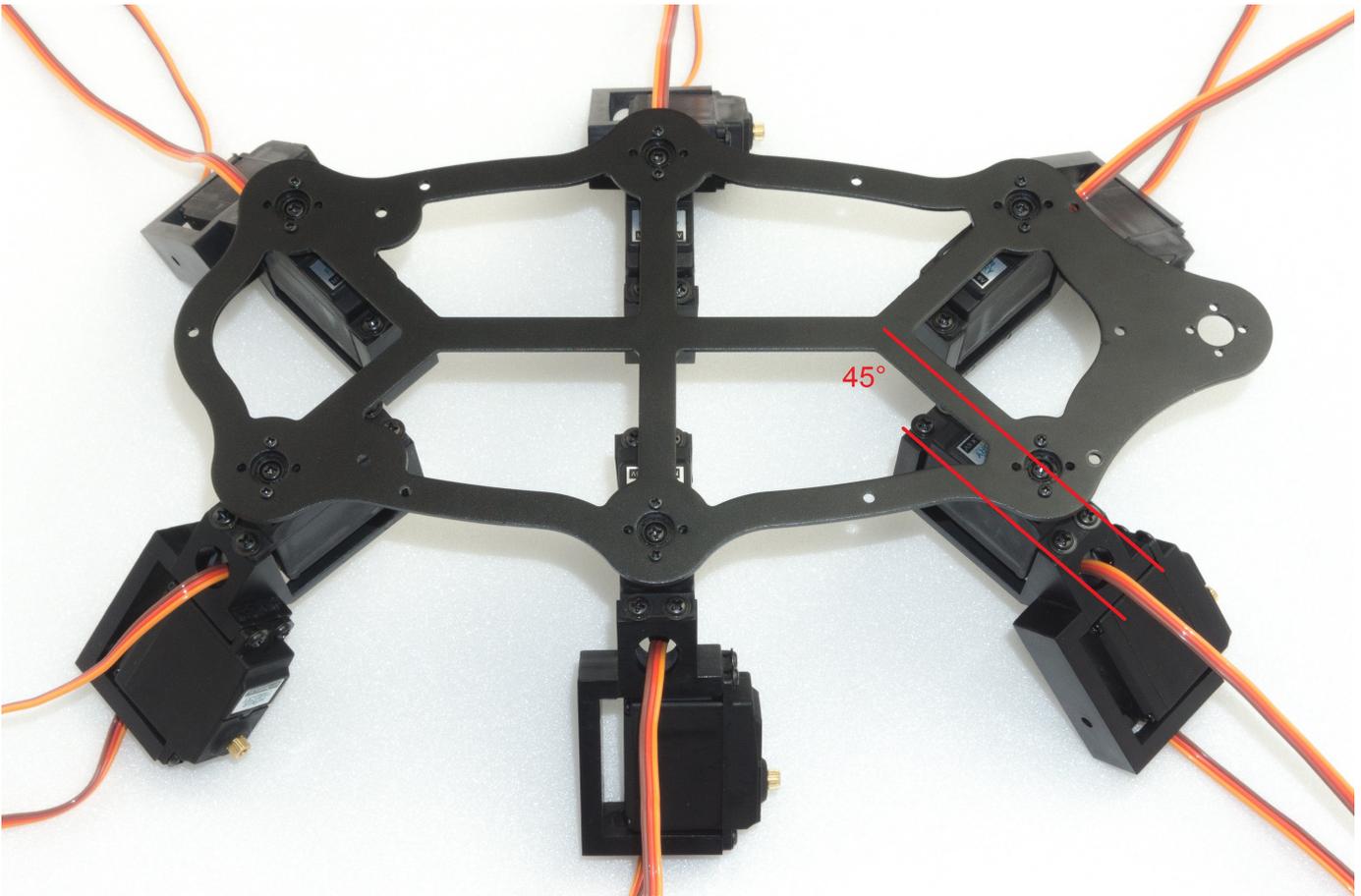


Figure 36

Figure 36 illustrates the Hexapod body viewed from "above". The two front and the two rear legs are installed in such a way that they point outwards at an angle of 45° without twisting the centre position of the servo (centre position). You can orient yourself on the cross-paths in the upper plate at an angle of 45°.

The two middle legs (left and right side) are installed straight outwards at an angle of 90° to the body.

→ Please also refer to Figure 37 on the next page.

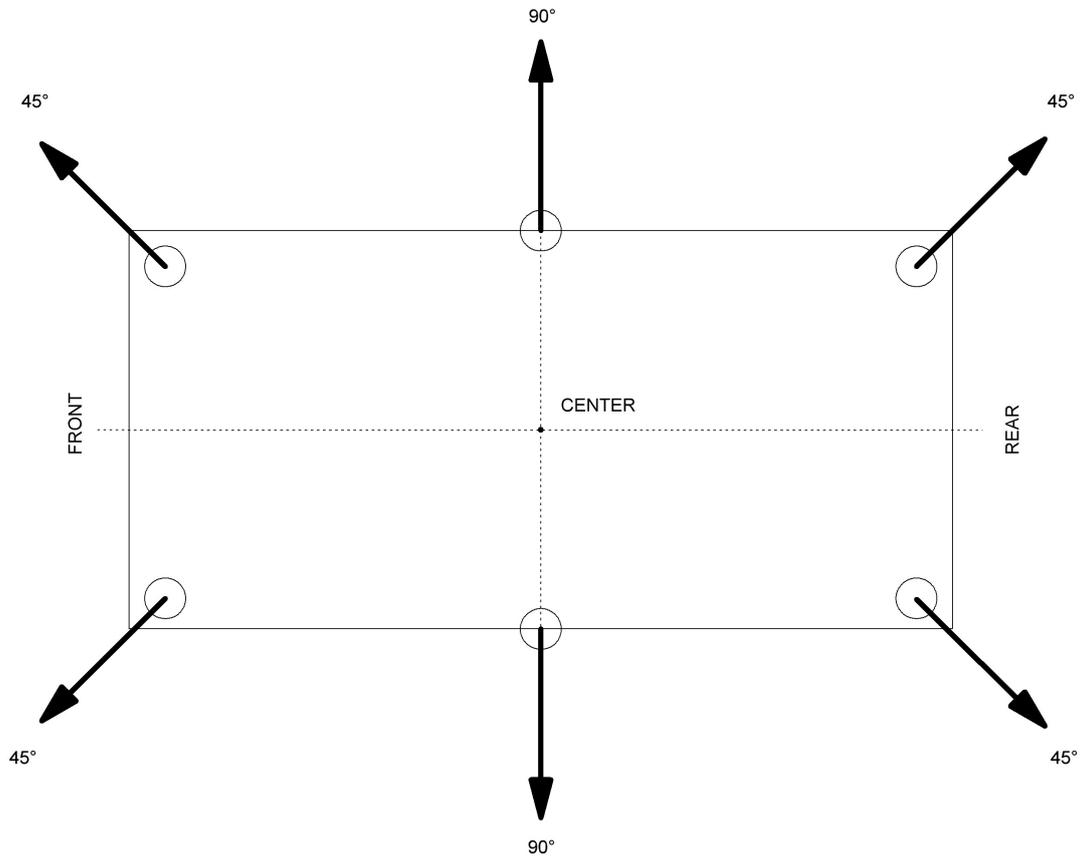


Figure 37

## f) Mount bottom plate

Mounting the bottom plate requires some skill. Place the top plate with its back on the table as shown in Figure 38.

Push the cylinder pins of the base plate into the plastic holders of the top plate, starting from the front or rear, all the way round. Do not push the pins completely in the first time, but repeat the working step and press the cylinder pin only a few millimetres into the plastic each time it circulates. This prevents the component from becoming strained!

→ Place an underlay, such as a cardboard box, between the component and the table so that you do not scratch the table with the screw heads! The foam from the packaging for instance is very suitable for this purpose.

Do not use force. This can damage the servos, the base plate and the ball bearings!

You can set the correct distance between the bearing and the plastic holder using a 2.5 mm Hexagon key. The Hexagon key serves as a distance gauge, see Figure 41.

Finally, the upper plate is screwed together with spacers. Use the M3 x 6 screws (small head diameter 5 mm) for this purpose.

**Required tool:** PH2 Phillips screwdriver  
Underlay e.g., cardboard box

→ Figure 38 shows the bottom side of the robot!

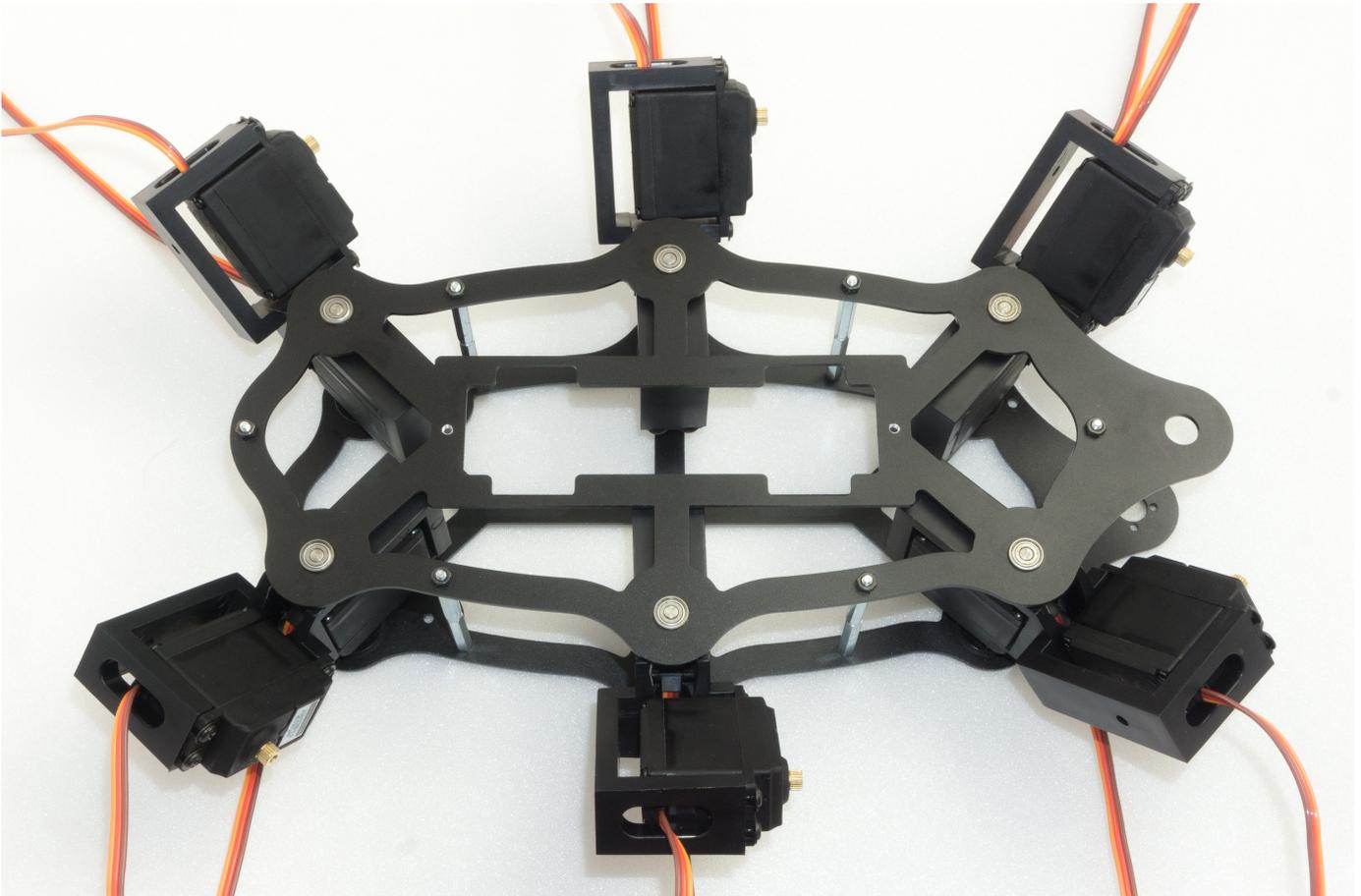


Figure 38

→ Figure 39 shows the top side of the robot!

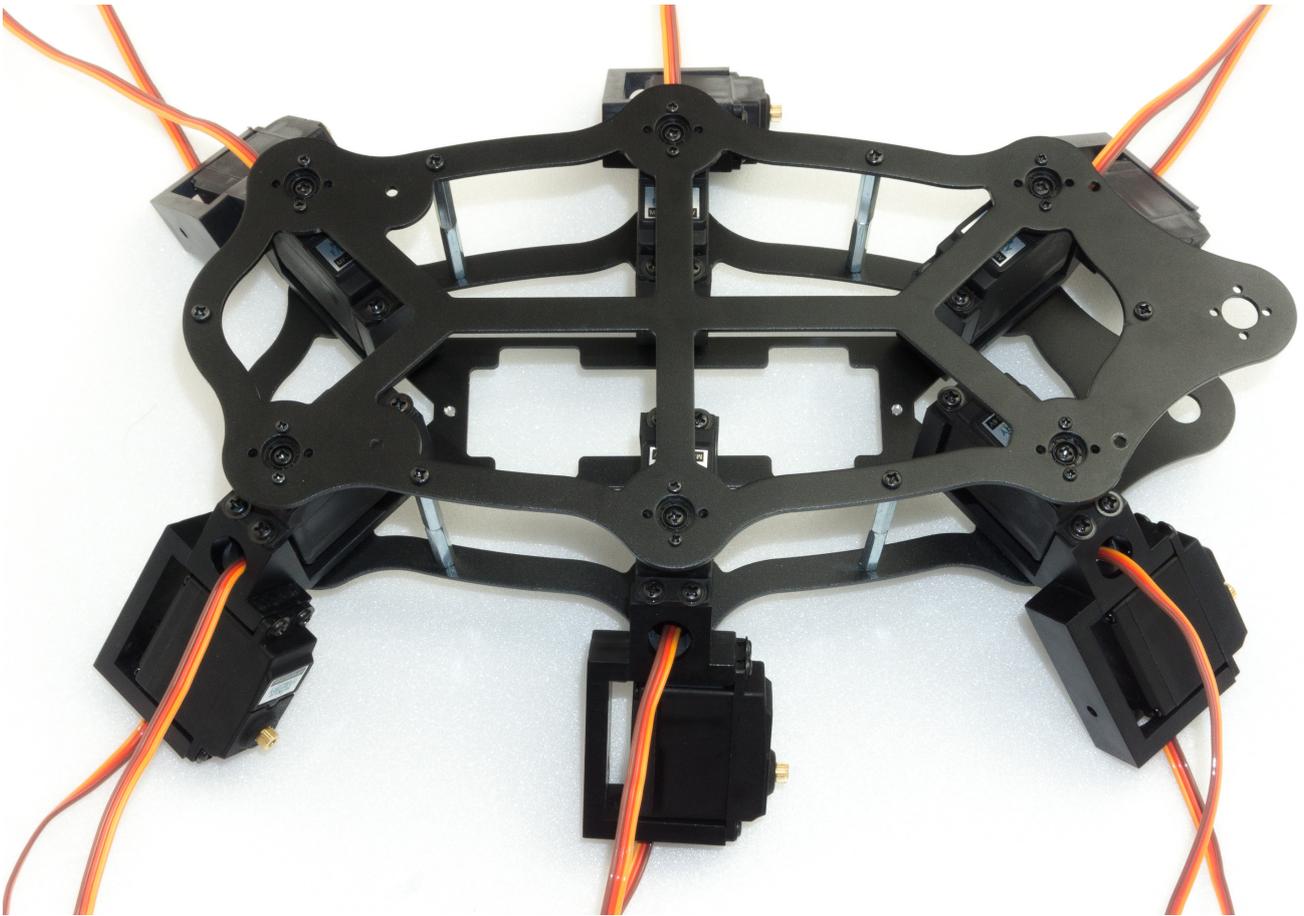


Figure 39

Figure 40 shows how the dowel pins are only slightly pressed in to fix them in the plastic.

Only when all the dowel pins have been fixed in the holes of the plastic holders are they pressed in at a distance (2.5 mm).

To achieve the correct distance, place the Hexagon key between the bearing and the plastic holder and press the cylinder pins into the plastic holder with moderate force.

If a cylindrical pin has been pressed in too far by mistake, you can correct this with a wide slotted screwdriver by gently levering it out. Make sure, however, that the aluminium parts are not damaged (scratched).

After all bearings have been pressed in and adjusted to the correct distance, screw them together with the spacers.

Figure 41 shows how to set the correct distance using the 2.5 mm hexagon wrench as the spacer.



Figure 40



Figure 41

## g) Mounting the Receiver and Hexapod Robot Board

Now the radio receiver for the gamepad is connected to the robot board and mounted. The receiver is fixed to the underside of the robot board with the supplied double-sided adhesive tape.

→ The receiver and the connection cable must first be connected to each other. Connect the wide receiver plug to the receiver. When connecting the receiver to the robot board, make sure that the polarity of the connectors is correct.

**Required tool:** Scissors  
PH2 Phillips screwdriver  
Open-end wrench SW 5.5 mm

Plug the cable into the receiver as shown in Figure 42.

The "POWER" and "RX" labels point upwards in the illustration. Use the same colour sequence as shown in the illustration.



Figure 42

Shorten the double-sided adhesive tape to approx. 25 mm using scissors.

Then peel off one side of the protective film of the adhesive tape and stick this side onto the robot board.

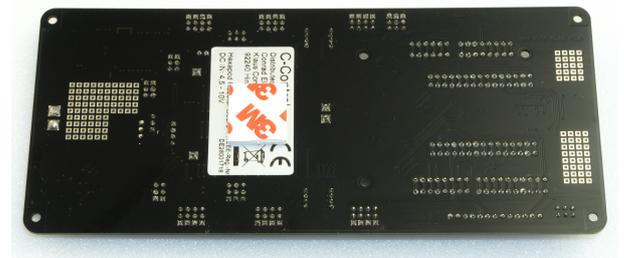


Figure 43

Now peel off the other side of the protective film and affix the receiver.

Make sure that the housing side with the two red LEDs locks with the "thick" contacts (see arrow in the illustration) of the loudspeaker.

If the housing protrudes too far into the contacts of the loudspeaker, the housing will press against the contacts of the loudspeaker when mounting the circuit board.



Figure 44

Now connect the cable to the robot board. The colours are combined in the plugs to 3 wires each.

From left (MicroSD card holder position) to right (battery plug position):

White, grey, purple -> White points to outer edge of board

Blue, green, yellow -> Blue points to the outer edge of the board

Orange, red brown -> Orange points to the outer edge of the board

→ Figure 45b shows a detail enlargement of Figure 45a.



Figure 45a

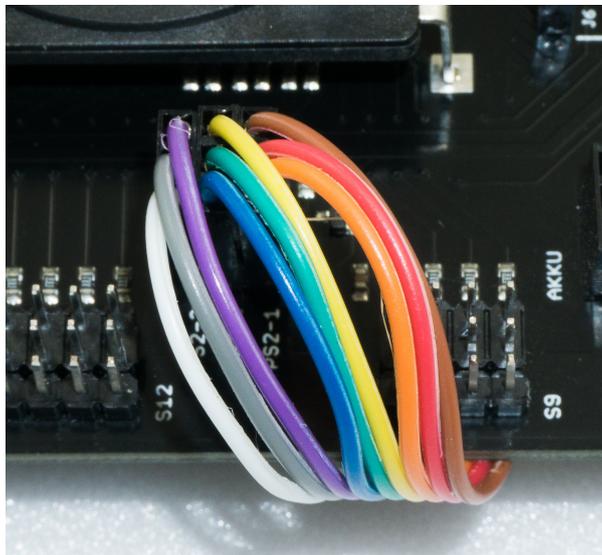


Figure 45b



## h) Mounting the legs

Try to mount the legs the as well as you can. Minor deviations do not play a role and cannot be avoided by the mechanical tolerances and by the coarse teeth of the servo disks.

→ The legs will be exactly adjusted later by software!

Figure 48 (see also Figure 20) shows the basic configuration of a Hexapod leg. It should be noted that the thigh (femur) is aligned parallel to the ground ("ground" in the picture).

The pivot point of the shin ("tibia") is parallel to the body ("body"). When the leg is fully adjusted, there is a 90° angle between the femur and the tibia. The arrangement between "body" and "femur" is parallel in one line. The difference in height does not play a role here.

Figure 49 (see also Figure 17) once again shows the pin assignment of the robot board for the leg servos. The following list shows wherein each servo is connected.

Basically, the connection sequence "COXA", "FEMUR", "TIBA" always applies; then follows the plug contact "Sx", which remains free!

Now connect the servos to the board. With the "ServoCenter" programme you can always check during assembly whether the leg is still in the basic position.

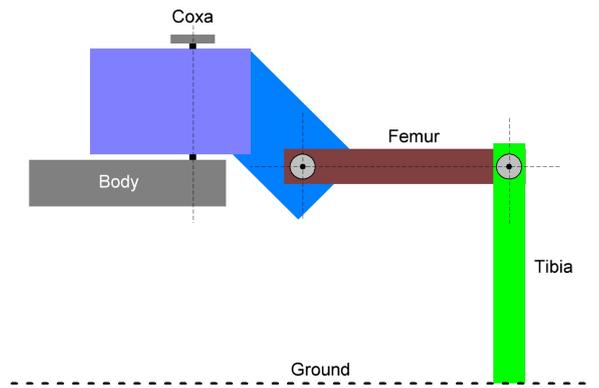


Figure 48

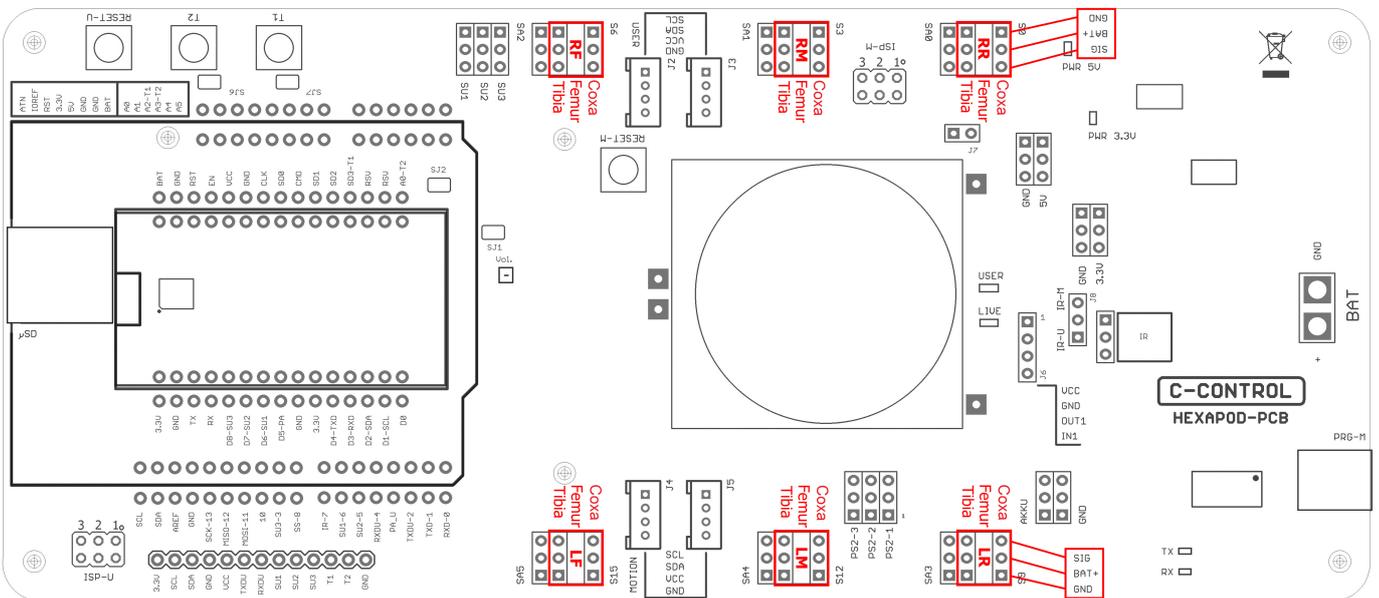


Figure 49

→ After mounting each leg, use the "ServoCenter" programme to check whether the leg is in the basic configuration or whether an error has crept in due to turning the servo while mounting!

### RR (right rear)

- S0 = Coxa
- S1 = Femur
- S2 = Tibia

### RM (right middle)

- S3 = Coxa
- S4 = Femur
- S5 = Tibia

### RF (right front)

- S6 = Coxa
- S7 = Femur
- S8 = Tibia

### LR (left rear)

- S9 = Coxa
- S10 = Femur
- S11 = Tibia

### LM (left middle)

- S12 = Coxa
- S13 = Femur
- S14 = Tibia

### LF (left front)

- S15 = Coxa
- S16 = Femur
- S17 = Tibia

Coxa = hip

Femur = thigh

Tibia = leg or shin-bone

First mount the thighs ("femur") as shown in Figure 50. Screw the thighs to the servos (M3 x 6 mm screws, large head diameter 6 mm). Then assemble the legs "tibia" and fix them with the M3 x 6 screws (large head diameter 6 mm).

It is helpful to lift the robot with the help of a cardboard box. If you have a 3D printer, you can also print out the robot stand.

The files can be found free of charge at [www.conrad.com](http://www.conrad.com) on the respective product website.

→ After the robot is completely assembled and adjusted, you should provide the screws, which connect the servo disk with the servo, with a small drop of screw lock. During operation they will loosen quickly and may get lost!

**Required tool:** PH2 Phillips screwdriver  
Open-end wrench SW 5.5 mm

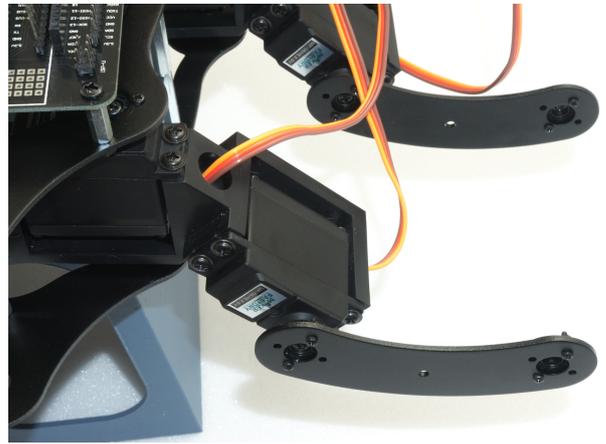


Figure 50

Despite the angled tibia servos, "femur" must be parallel to the floor at the end and "tibia" parallel to the body.

The "femur" and "tibia" make a 90° angle; the servos themselves are still in the middle position!

Figure 52 shows the Robobug when all legs are fully assembled and in the home position. Ideally, the red line (Figure 51) would run exactly in the middle of the screw heads. Since the servo disks do not allow this due to the tothing and the mounting tolerance, it may be slightly off the mark with some legs. This is not bad, the exact alignment to the middle will be done later by software.

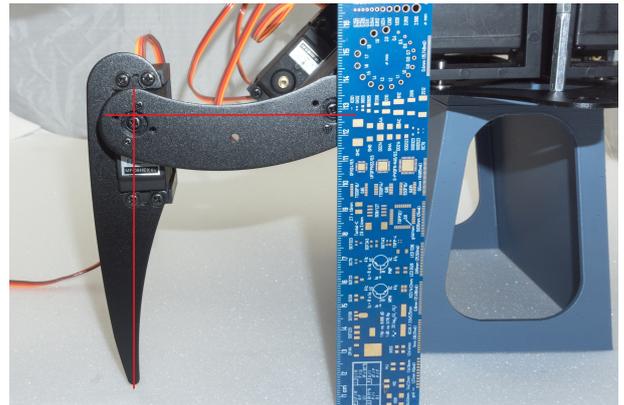


Figure 51

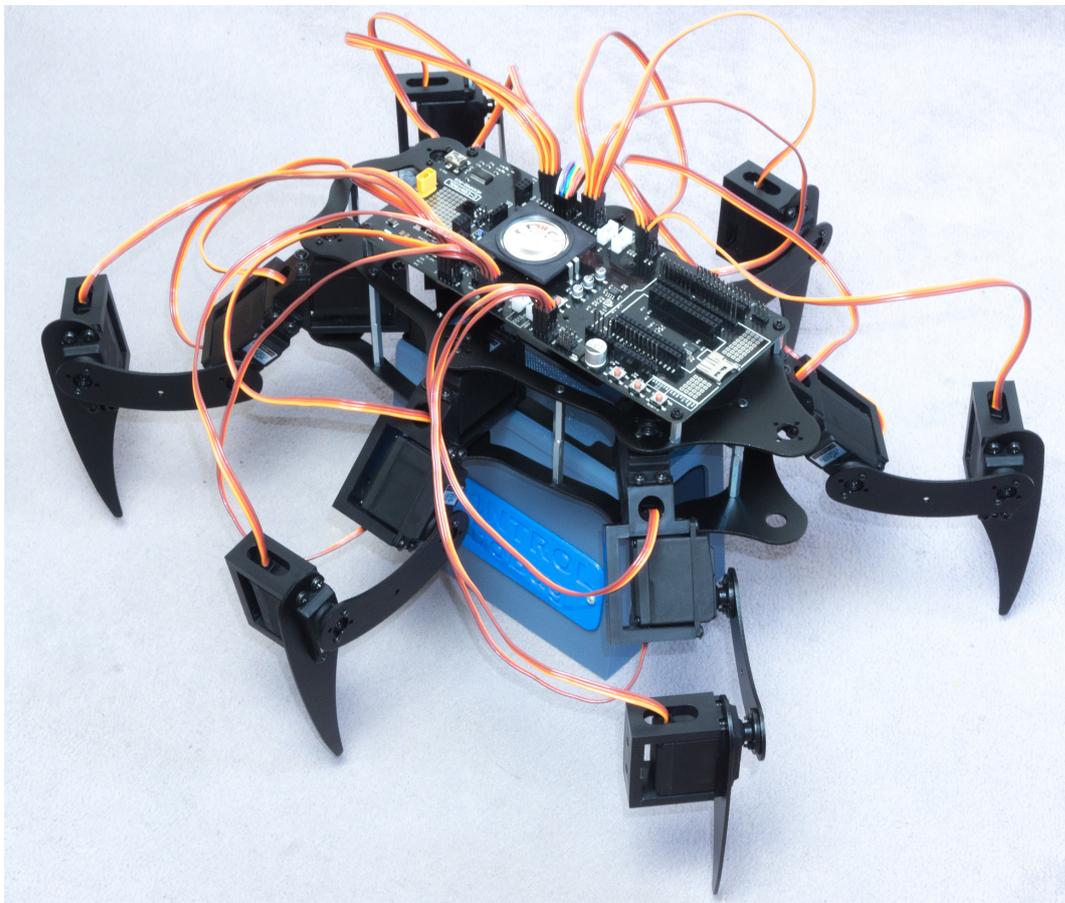


Figure 52

Now mount the counter bearings.

→ The U holder has two holes, use the hole facing the cable opening.

When pressing in the dowel pins, proceed in the same way as when pressing in the dowel pins of the base plate.

The distance between the bearing and the plastic part here is 1.5 mm. You can also use an Hexagon key as a distance gauge here.

Then screw the spacer on each leg with a M3 x 6 mm screw (small head diameter 5 mm) as shown in Figure 53.

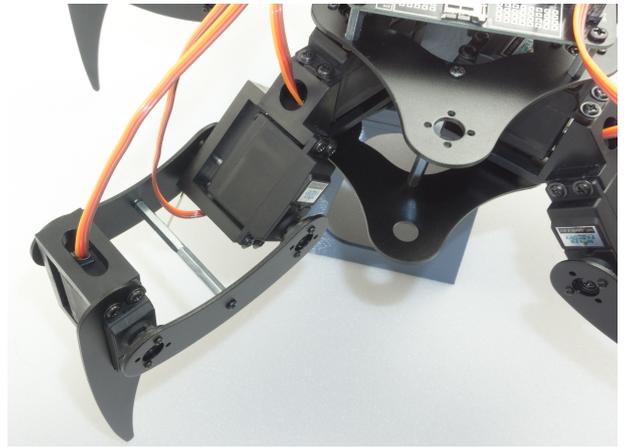


Figure 53

The 1.5 mm hexagon key is placed between the ball bearing and the plastic part and the cylindrical pin is pressed in until it stops.

Also, in this case, press the cylindrical pins only alternately (first one side then the other) piece by piece into the plastic so as not to tilt the holder.

→ If you hold the robot at an angle, you can hang the Hexagon key over the cylinder pin without it falling down.

Figure 55 shows the Robobug with the legs fully assembled and in the basic position.

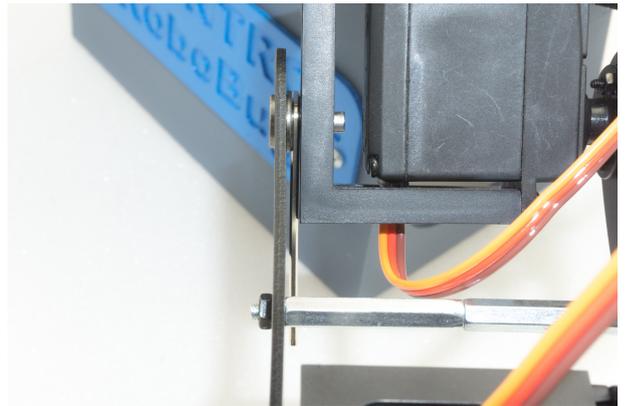


Figure 54

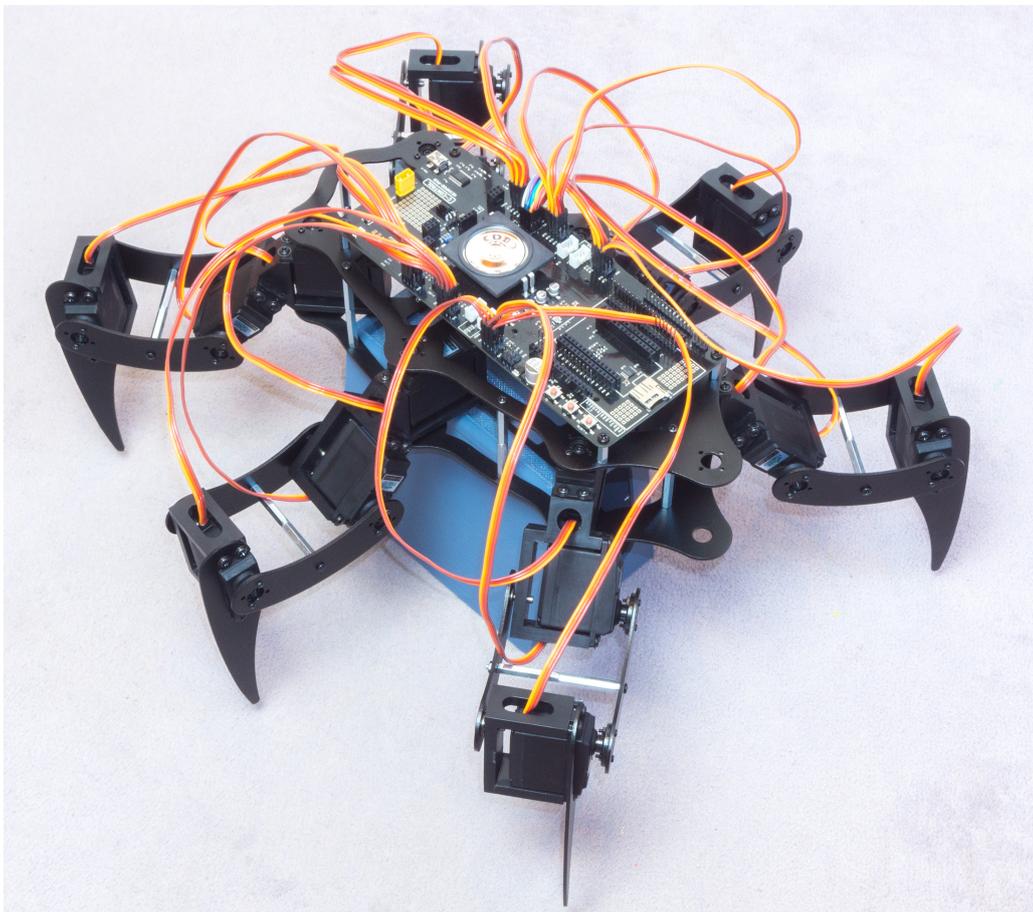


Figure 55

## i) Laying the cables

Lay the cables as shown in the illustrations. Use the supplied cable ties for this purpose. After the cable ties have been tightened, cut off the protruding part of the cable tie. A clean cable laying gives the robot a very high-quality appearance!

→ As long as you have not finished aligning the cables, you should not tighten the cable ties too tightly. This way you can make small corrections easily.

**Required tool:** Electronic side cutter

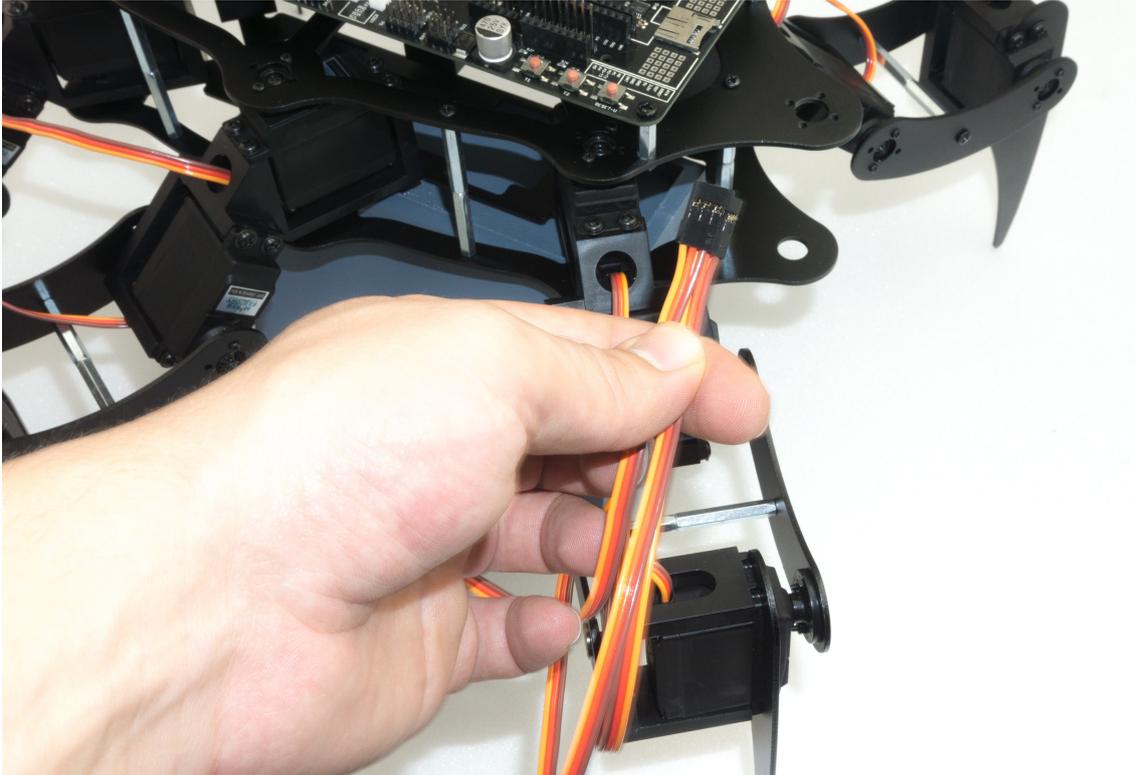


Figure 56

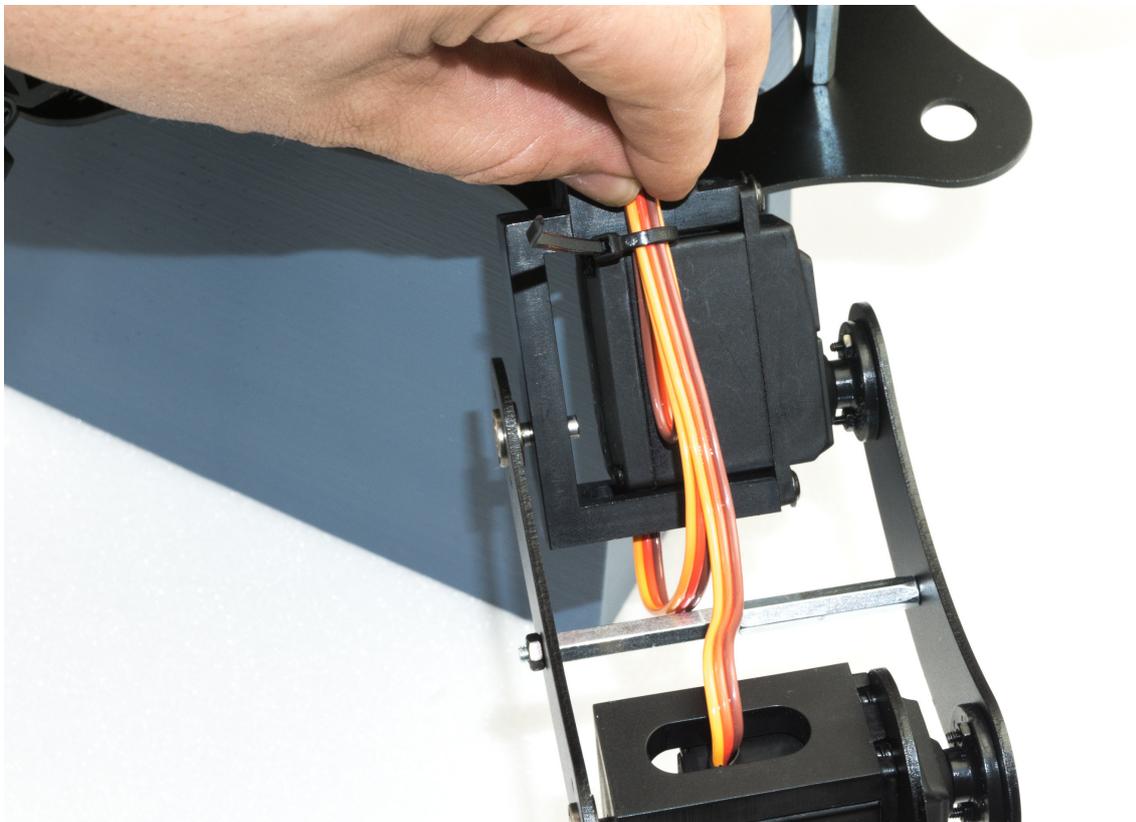


Figure 57

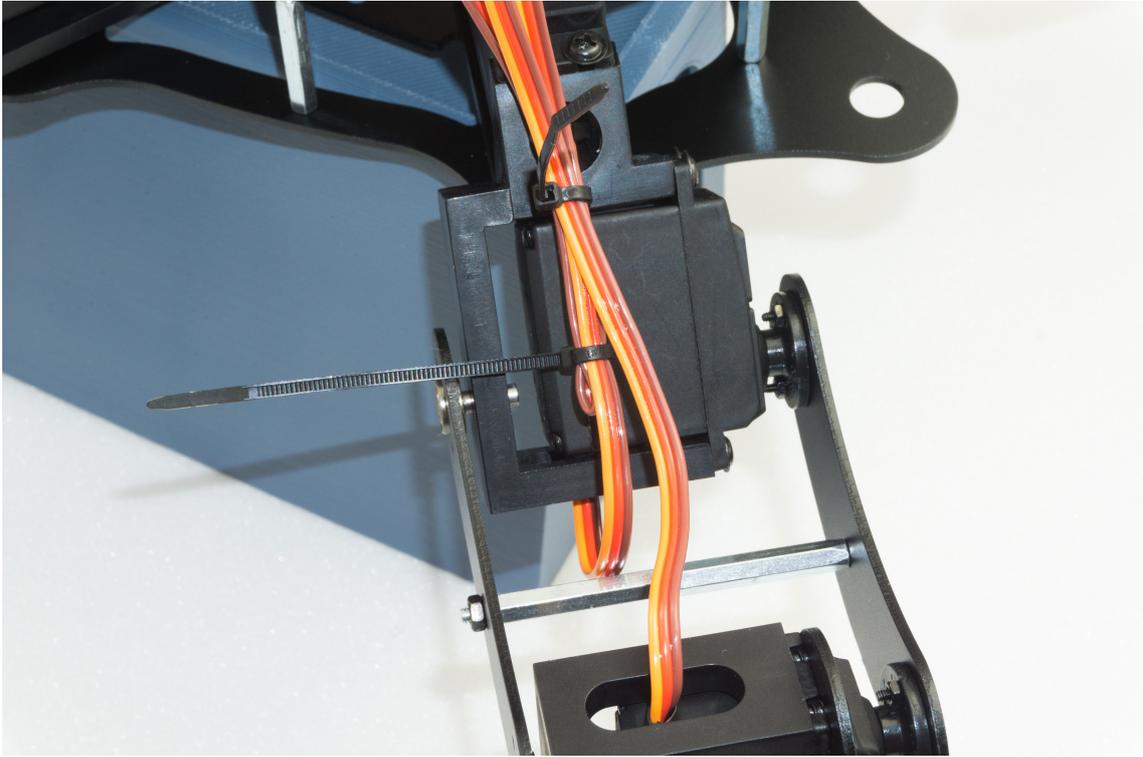


Figure 58

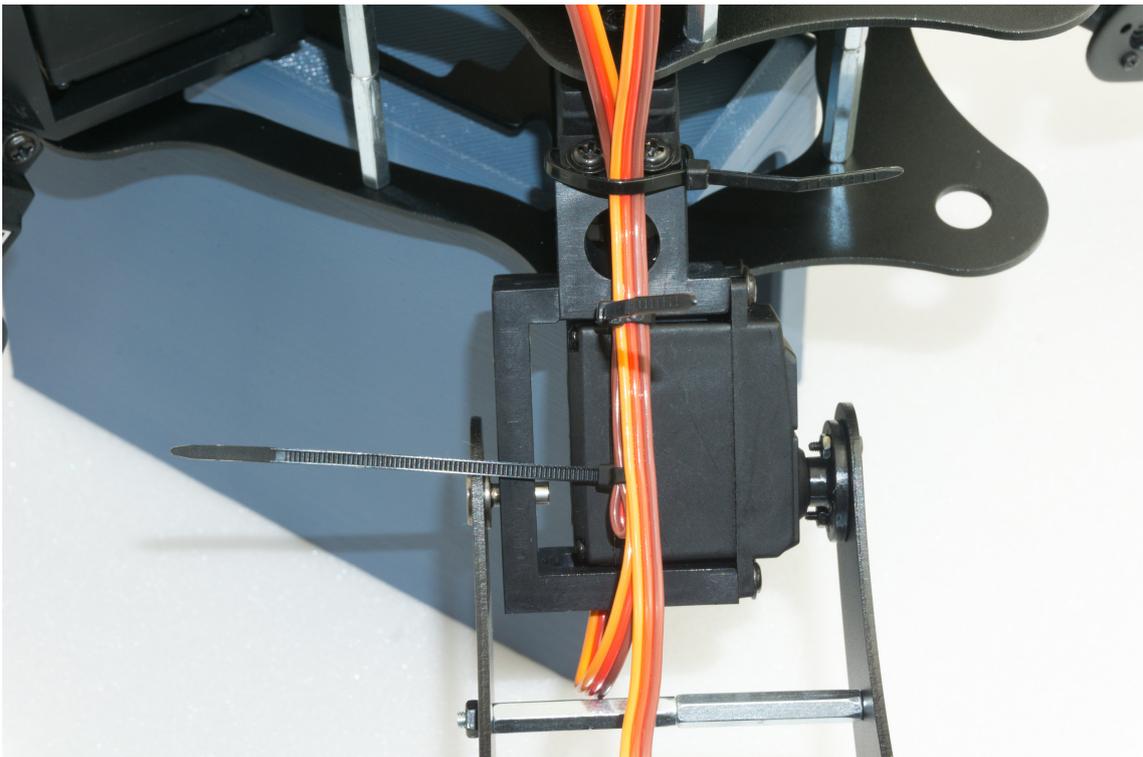


Figure 59

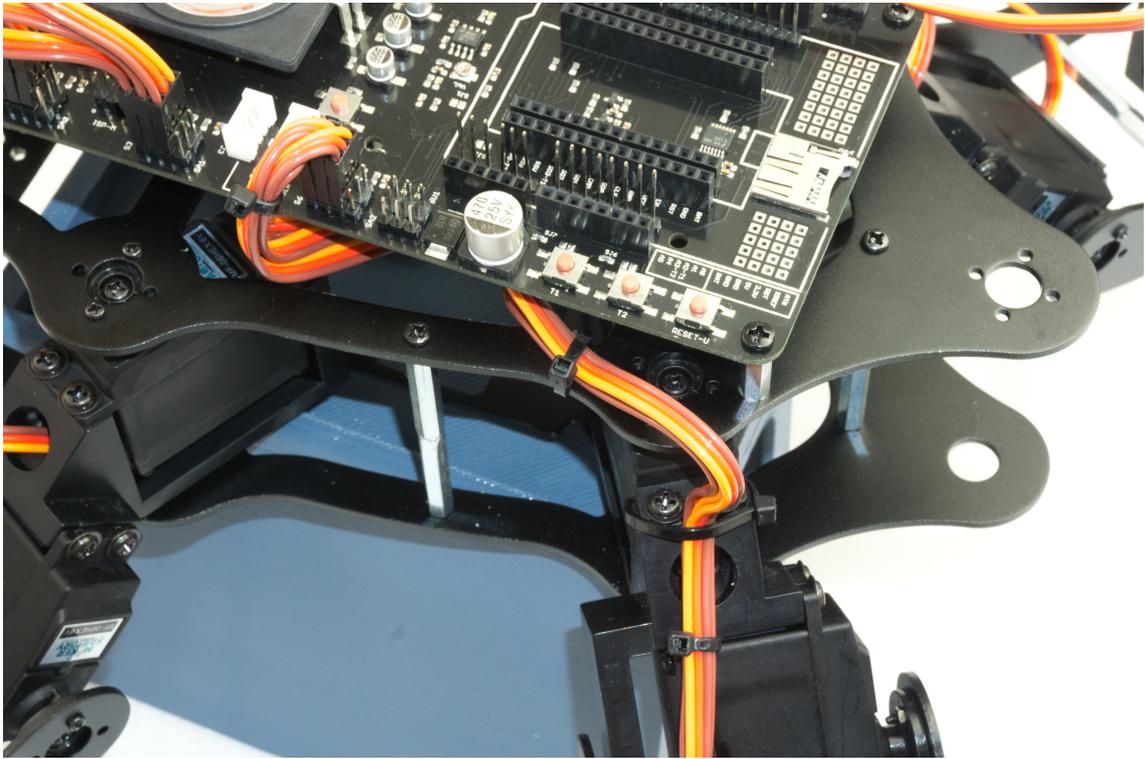


Figure 60

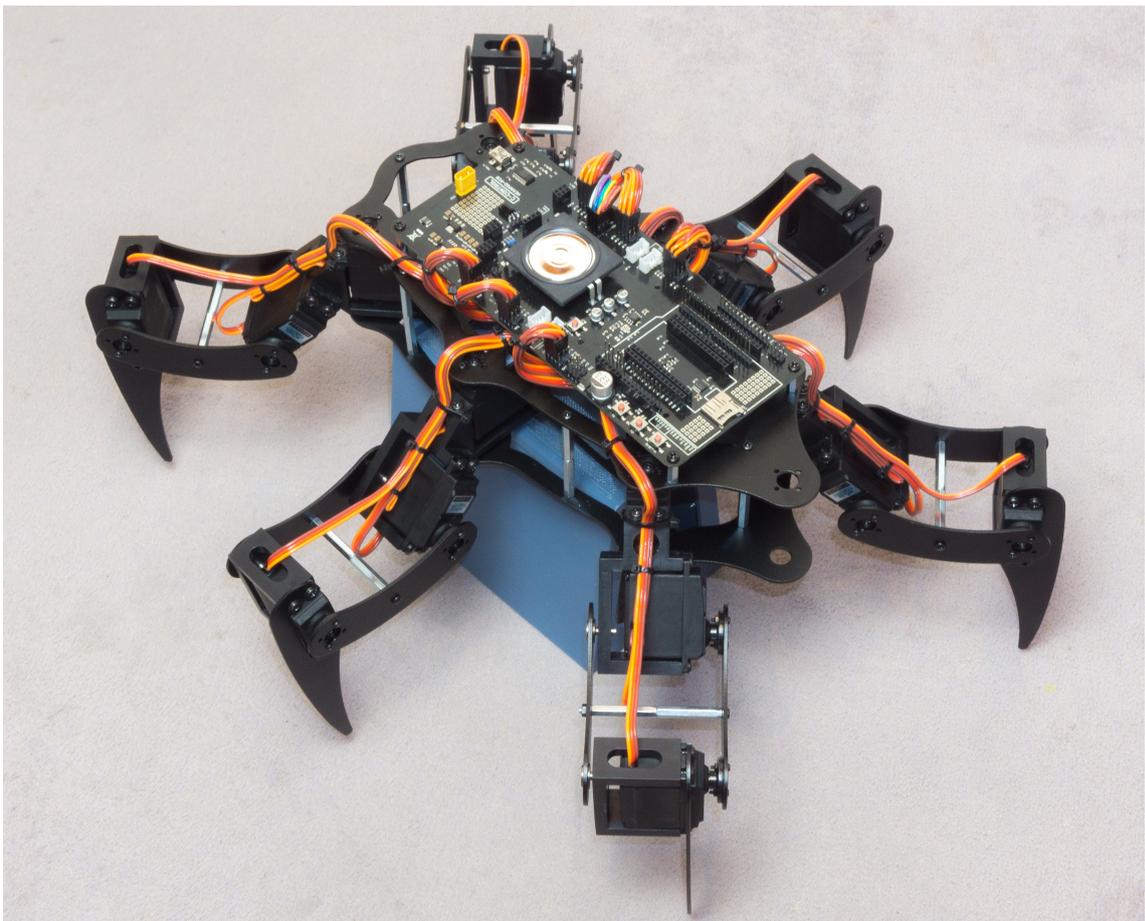


Figure 61

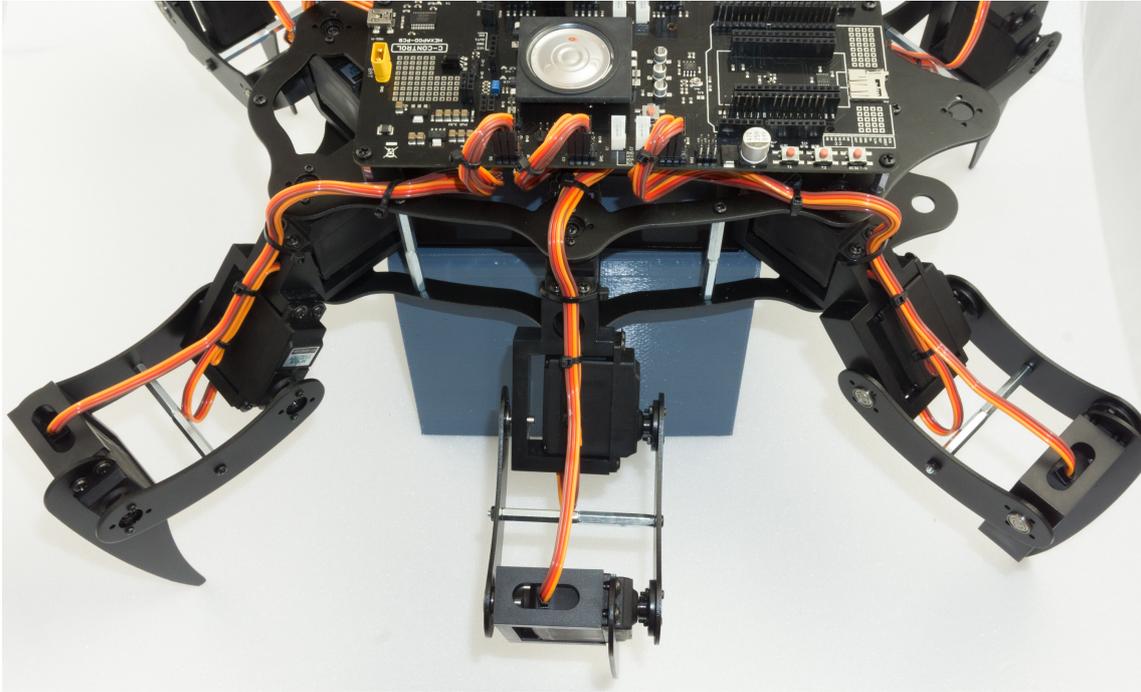


Figure 62

## j) Installing the battery

The battery is mounted edgewise with the supplied Velcro fastener as shown in the illustrations. To do this, shorten the Velcro fastener to approx. 25 cm. Tighten the Velcro so that the battery has a firm grip on the battery plate and cannot slip during operation.

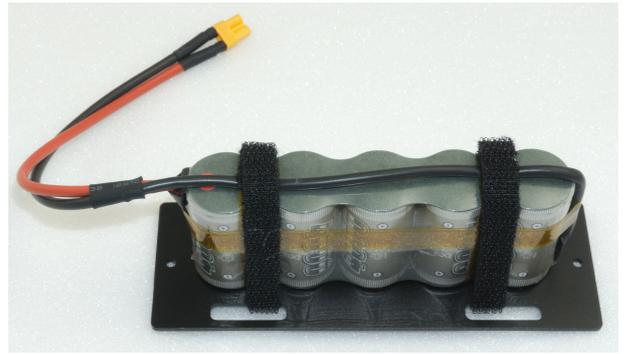


Figure 63

Then run the cable through the top plate as shown in Figure 64a. Screw the battery retaining plate to the lower plate. Use the screws M4 x 5 mm for this purpose.



Figure 64a

In Figure 64b (see also Figure 7) you can see how the battery is plugged into the Robobug. The plug is protected against reverse polarity and cannot therefore be plugged in the wrong way.

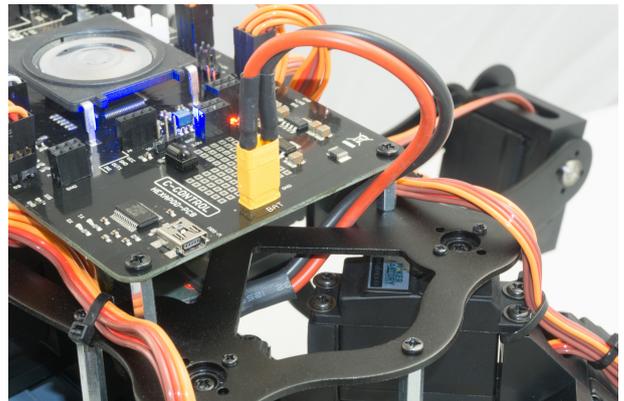


Figure 64b

### k) Mounting the rubber feet

Slide the supplied rubber feet onto the ends of the legs.

You can also secure the rubber caps with some glue (e.g., Uhu® All Purpose Glue or similar) so that they are not lost during operation.

→ The rubber feet not only secure a good grip on the ground, but also ensure that the aluminium legs do not scratch the ground (floor, table etc.).

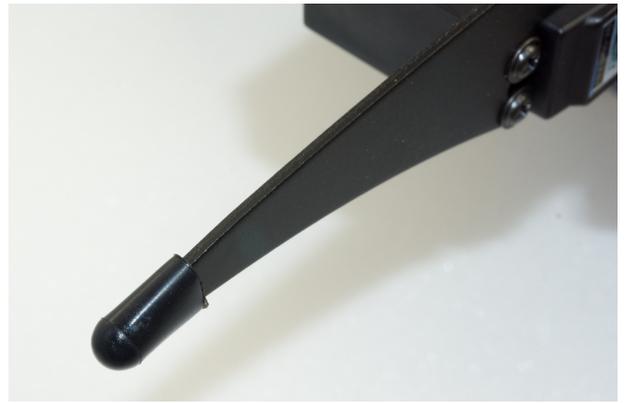


Figure 65

### l) Setting jumpers

Finally, the two jumpers are plugged in. The jumper for the infrared receiver is plugged on "IR-U"; the jumper J7 is plugged only on one pin.

→ If you connect jumper J7 to both pins, the gamepad will be deactivated!

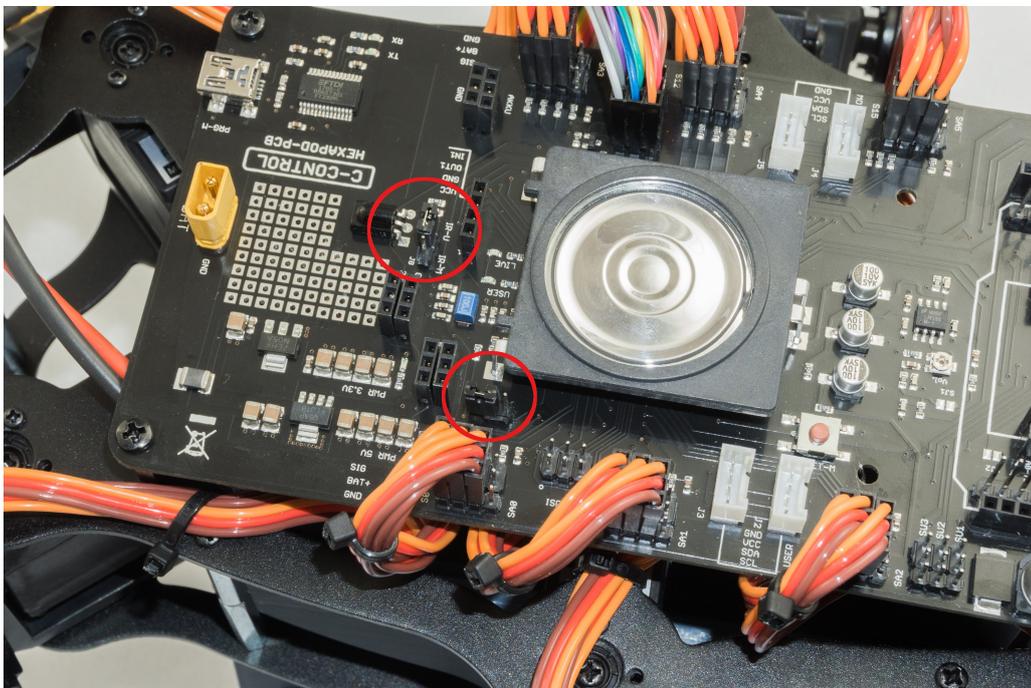


Figure 66

# 14. Transferring the Firmware

After you have finished the mechanical assembly of the robot, the "Locomotion Firmware" can be installed on the robot board. This is responsible for the running behaviour of the robot.

**Check the following before transferring and switching on the robot board:**

- Are all servos connected correctly (polarity)?
- Are all servos mechanically in the middle?
- Is the Gamepad controller connected?
- Is the battery sufficiently charged?
- Is there enough space available for the robot, even in the event of a malfunction?

Now connect the battery to the robot board, if not already done.

Connect the robot board (connection "PRG-M") to your computer as already done.

Then select in the Arduino IDE in the Hexapod examples the programme "Motion Firmware" (located under "Locomotion"). Transfer the programme to the robot board as usual.

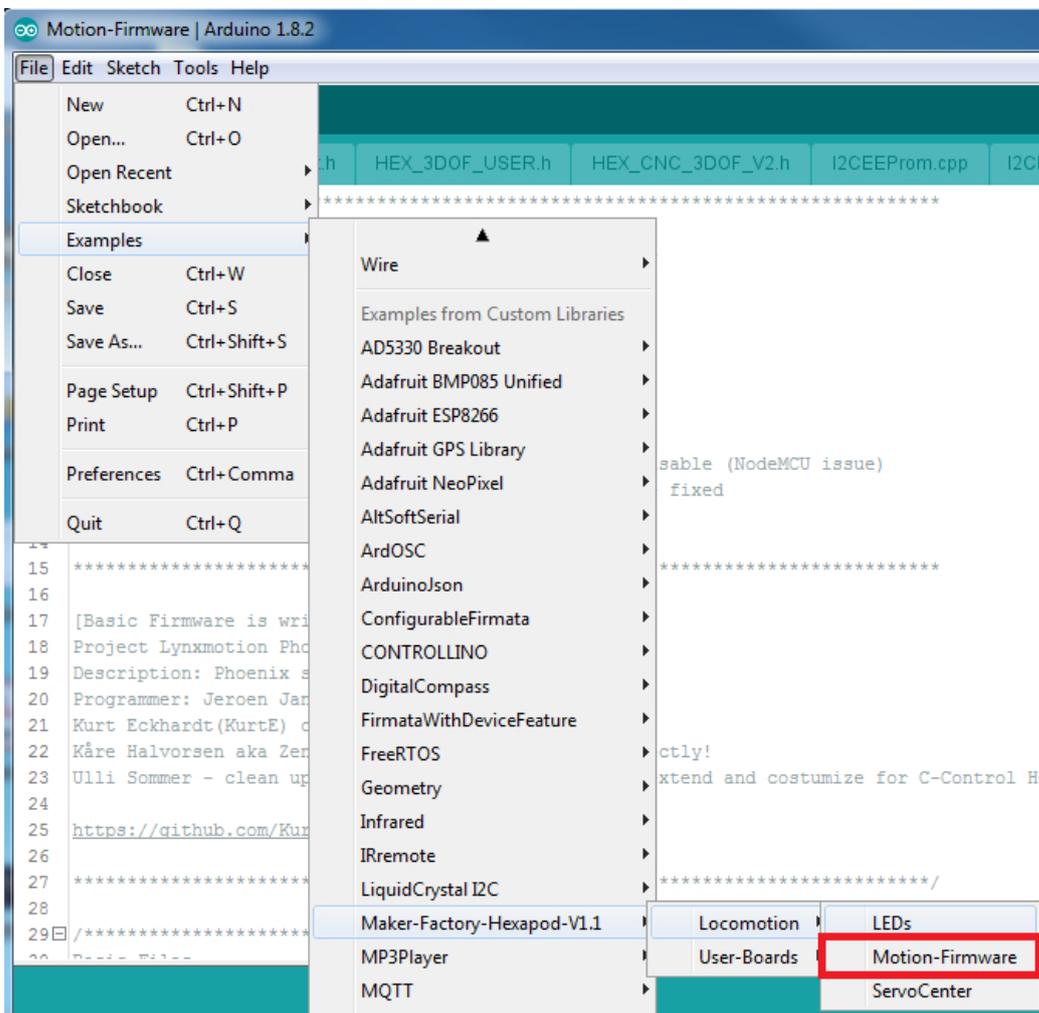


Figure 67

# 15. Controlling with the Gamepad

After you have transferred the "motion firmware" to the Locomotion controller (see chapter 14), you can control the robot with the gamepad. With the included gamepad, all functions of the Locomotion controller are immediately available to you after commissioning and the robot can be moved similar to a remote-controlled car without you having to create your own programme beforehand.

→ To check the settings comfortably and quickly, it is advantageous to operate the robot with the gamepad. Alternatively, you can also control it directly via a user board. In practice, however, this has turned out to be less convenient for initial commissioning.

In addition, you can later intervene in the sequence of your programme with the help of the gamepad when using a user board. It is also possible to read the control commands (joystick values and buttons) from the user board and use them in your own program.



### Important!

Make sure that the receiver is connected correctly and that the plugs are not twisted on the pin strips, otherwise the receiver may be damaged! The jumper "J7" must not be plugged in, otherwise the gamepad is deactivated (see chapter 13. I).



Figure 68

### Quick Start:

- Disconnect the power supply from the Hexapod Robot Board.
- Switch on the gamepad with the switch (6). The two LEDs (12) "Green" and "Red" flash.
- Connect the battery to the Hexapod Robot Board. The LEDs (12) "Green" and "Red" light up permanently after a few seconds. The transmitter (gamepad) is now connected to the receiver.
- Press the "START" button (15). The robot beeps and the red "USER-LED" on the Hexapod Robot Board lights up permanently.
- Switch on the robot with the key (3). The robot beeps and the blue "LIVE-LED" on the Hexapod Robot Board flashes.
- Let the robot stand up by pressing button (1).
- Use the thumb joystick (7) to move the robot forwards/backwards and left/right. With the thumb joystick (5) you can turn the robot on the spot.

#### Detailed description of the gamepad functions:

- 1 Make the robot stand up and sit down (toggle function)
- 2 System reset
- 3 Switch the robot on/off after the gamepad has been activated with the key (15)
- 4 Switch on/off balance mode (toggle function)
- 5 Right Thumb joystick (turn on the spot, turn in rotate and translate mode and change the height of the robot)
- 6 LED lights green when the connection to the robot has been established
- 7 Left thumb joystick (forward/backward, in rotate and translate mode forward/backward and lateral movement).
- 8 D-Pad up/down adjusts the height of the robot (up = higher, down = lower).
- 9 D-Pad left/right adjusts the running speed (left = slower, right = faster)
- 10 Upper key = translate mode (toggle function)
- 11 Lower key = rotate mode (toggle function)
- 12 On/off switch
- 13 Select gait (Tripod-6, Tripod-8, Tripple-12, Tripple-16, Ripple-12, Wave-24). Each time the button is pressed again, the gait is shifted through as described in the list. If the list starts from the beginning (Tripod-6), the Hexapod Robot Board signals this with a different tone sequence.
- 14 No function
- 15 Activate gamepad (after this the robot reacts to control commands)
- 16 Activates double step height (toggle function)
- 17 Activates double step size (toggle function)

→ All buttons with toggle function (renewed pressing, changes the state) makes the robot board beep in different tone sequence. If you press the key for the first time, the Hexapod Robot Board acknowledges this with a beep and when you press it again (original state) with a tone sequence. This makes it easy to find out what state the robot is in. After the tone sequence has sounded, the original state (start state) is always active.

## 16. Calibrate legs

To adjust the legs exactly, use the programme "Terminal" (if you work with the Windows operating system) or alternatively, the Arduino IDE Terminal.

The aim now is to adjust the legs into an exact basic configuration. When assembling the mechanics, you will not be able to bring the legs 100% into the basic configuration due to the teeth of the servo disks and the mechanical tolerances. You can use this tool to compensate for these deviations.

Figure 69 shows how the basic configuration should look for each leg.

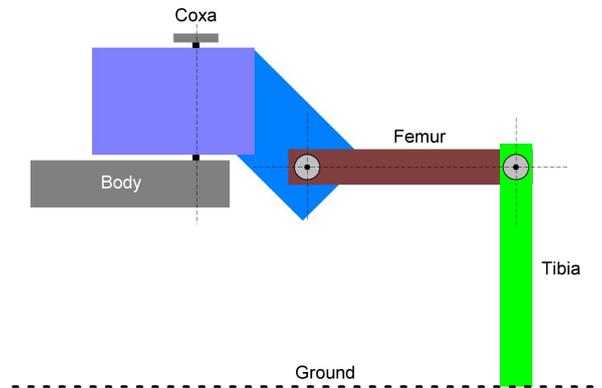


Figure 69

The fictitious red lines show how to align the leg. The thigh "femur" must be aligned so that it is parallel to the floor.

The tibia is perpendicular to the floor and makes a 90° angle at the thigh.

If the robot is placed on a flat surface with a mounting aid as shown in the illustration, the distance between the table and the centre of the M3 cross screws can easily be measured with a ruler. The angle can also be easily checked with a ruler by using it as an angle as shown in the illustration. The ruler stands with the short edge on the table and the long side must coincide with the vertical red line.

Figure 70 shows a ready mounted leg in the basic position. Ideally, the red line would run exactly in the middle of the screw heads. Since the servo disks do not allow this due to the tothing and the mounting tolerance, it may be slightly off the mark on some legs. You can now correct this with the calibration.

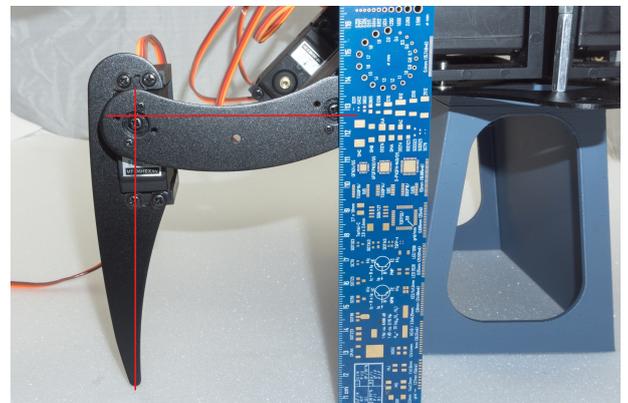


Figure 70

### → Use a mounting aid:

Our mounting aid is ideal for calibrating the legs.

The mounting aid is available as a 3D model for download at [www.conrad.com](http://www.conrad.com) under the product. For printing you need a suitable 3D printer. Alternatively, you can of course also use any other base such as a pot, a sturdy box or similar. It is important that the robot stands stable on it and does not wobble; the legs must not be obstructed in the movement.

## Calibrating legs with PC

If you are using Windows, you can use the "Terminal" program. The programme is located in the "Terminal" folder.

If you are using MAC OS or Linux, you can skip this part and continue with the section "Leg calibration with the Arduino IDE".

- Start the programme with the file "Terminal.exe" and perform the calibration. Elevate the robot upwards so that the legs can move freely in the air. A stable box can, for example, serve for the elevation.
- Connect the Hexapod Robot Board (connector "PRG-M") to your computer as already done when installing the firmware.
- Connect the battery to the Hexapod Robot Board.
- Start the programme "Terminal.exe".
- Now select the serial port (same port as for programming). To do this, click on "Select Port". If more than one device is displayed, simply unplug the motherboard and plug it back into the USB port. The device which then disappeared briefly is the correct one.
- Select 38400 Baud under "Baud rate".
- Click on "Connect". The robot appears in the terminal window, as shown in Figure 71.
- Click on "Servo offset mode" in the upper left corner to enter the calibration mode.
- The robot will now "wobble" with the servo selected for adjustment.
- By clicking on "+" and "-" several times, you can adjust the selected servo exactly in small steps.
- Once you have finished setting the servo, you can click on "Next" to set the next servo until you have set all 18 servos. You can also select a specific leg by clicking on the "Leg RR", "Leg RM", "Leg RF", "Leg LR", "Leg LM" and "Leg LF" buttons.
- Once all servos have been set, click on "Exit". The programme now asks whether the settings should be saved. Click on "Yes" to save the settings or "No" to discard the settings.
- With "Clear Terminal" you can delete the contents of the terminal window.

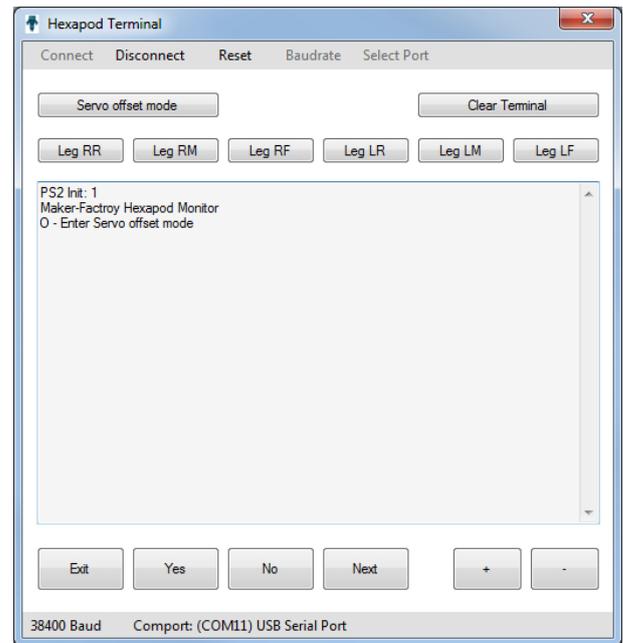


Figure 71

## Calibrating legs with the Arduino IDE

If you are using a MAC or Linux, you cannot use the "Terminal" programme (only suitable for Windows). However, you can still adjust the legs by using another terminal programme of your choice.

The easiest way is to use the Arduino Terminal. You can invoke this in the Arduino IDE.

The figure shows the Arduino terminal after the terminal has connected to the robot board.

### Proceed as follows:

- Connect the Hexapod Robot Board (connector "PRG-M") to your computer as already done when installing the firmware.
- Connect the battery to the Hexapod Robot Board.
- Start the terminal program.
- If you are using a terminal other than the Arduino terminal, select the serial port (same port as for programming).
- Select a baud rate of 38400 baud. No LF and CR or "No line encoding" in the Arduino terminal.
- Connect the terminal to the robot. The robot appears in the terminal window, as shown in Figure 72. Please note: The Arduino terminal connects immediately after opening!
- Press the "RESET-M" button on the Hexapod Robot Board once. This is not necessary when using the Arduino terminal.

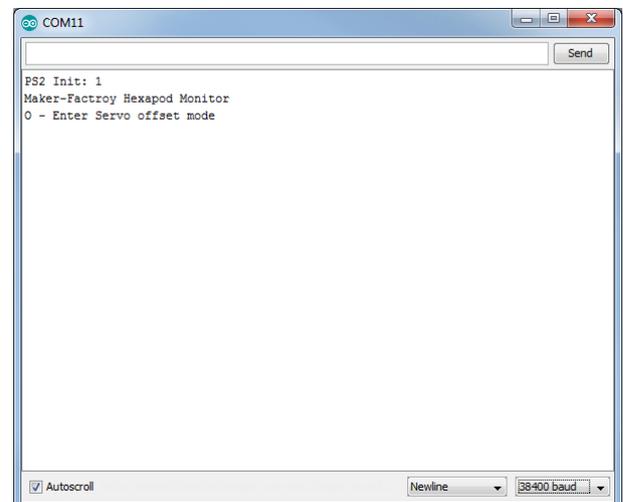


Figure 72

The following text appears in the terminal window.

```
PS2 Init: 0
Maker Factory Hexapod Monitor
O - Enter servo offset mode
```

Send an "O" to the robot board by writing "O" in the transmission line and then pressing the Enter key. The robot will now "wiggle" with the chosen servo in the setting.

→ The characters can be written in small or capital letters ("o" as well as "O" is possible).

The following message then appears in the terminal:

```
Serial Cmd Line:o<eol>
Find Servo Zeros.
$-Exit, +- changes, *-change servo
0-5 Chooses a leg, C-Coxa, F-Femur, T-Tibia
Servo: RR Coxa(0)
```

The following input options are available:

- With "\$" you leave the calibration -> Exit
- With "+" or "-" you can precisely adjust the selected servo in small steps. The characters must be sent to the Hexapod Robot Board until the leg is correctly adjusted.
- Use "\*" to select the next servo.
- With a number from 0 to 5 you can select the legs individually.

```
0 = Leg RR (right rear)
1 = Leg RM (right middle)
2 = Leg RF (right front)
3 = Leg LR (left rear)
4 = Leg LM (left middle)
5 = Leg LF (left front)
```

When all servos are set, send a "\$" character. The programme will now ask if you want to save the data. Send "y" for save or "n" to discard the settings.

→ You may need to perform this calibration procedure again after some time. The reason for this is that after a certain operating time the servos get a larger gear backlash and the first calibration no longer proves to be ideal.

After the Robobug has been assembled and calibrated, you can unscrew the screws M3 x 6 mm (large head diameter 6 mm) and secure them with a screw lock. These are then screwed in again. Also check all other screws for tightness.

# 17. Using user boards

The robot board can be equipped with different user boards to extend the functionality. Possible user boards are e.g., Arduino UNO, NodeMCU or compatible boards. These can be plugged directly onto the robot board.

With the software library described in the next chapter you can send control commands from the user board to the locomotion controller or read values from the locomotion controller. More about this later in the next chapter "Demo Programmes".

- **Arduino-UNO**

The two outer longer pin headers are used for a compatible user board "Arduino-UNO". All pins are led out at socket connectors next to it. This allows you to easily access the pins of the board. The pin assignment is printed on the board next to the pin headers and corresponds to that of the board used. Have a look at the software examples.

- **NodeMCU**

The popular WiFi board with the designation "NodeMCU" and the employed ESP8266 WiFi chip can be used on the inner slot as a user board. Also, here all pins are led out on the adjacent socket strips, which makes experimenting easier. Have a look at the software examples.

- **SBC (Single Board Computer)**

A Raspberry Pi 2 or 3 or any other SBC can be connected to the pin header next to the "ISP-U" connection. It is also possible to use radio modules or Bluetooth® modules as user boards via this connection. This pin header should be regarded universal and represents an interface for proper extensions and developments.

- **Arduino UNO**

Plug an "Arduino UNO" board onto the robot board as shown in Figure 73. Proceed carefully and make sure that the longer pin strips, which are plugged into the sockets of the "Arduino UNO" board, are not bent.

Also, make sure that all pin header pins are plugged into the sockets of the "Arduino UNO" board.

It is possible to use almost all Arduino UNO compatible boards.

After plugging in the board, you can jump directly to the chapter "Demo Programmes".

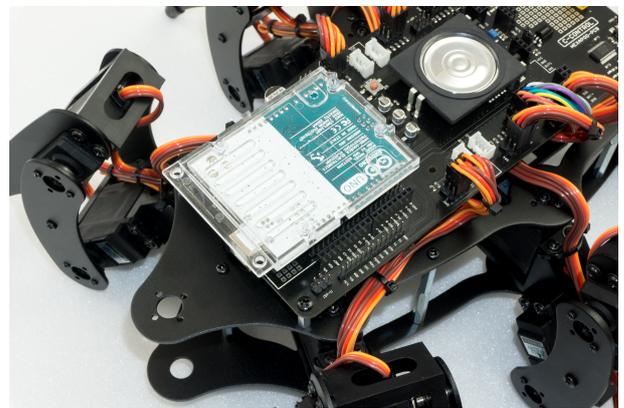


Figure 73

- **NodeMCU WiFi Board**

The WiFi board called "NodeMCU" and the ESP8266 WiFi chip used can be used as a user board on the inner slot.

If you want to extend the robot board with WiFi, you can easily do so with the help of this user board. The examples already include programmes for the NodeMCU board that allow control view from a web browser.

Plug a NodeMCU board onto the Hexapod Robot Board as shown in Figure 74. Proceed carefully and make sure that the pin strips are not bent and that all pins are plugged in correctly.

After plugging in the board, you can jump directly to the chapter "Demo Programmes".

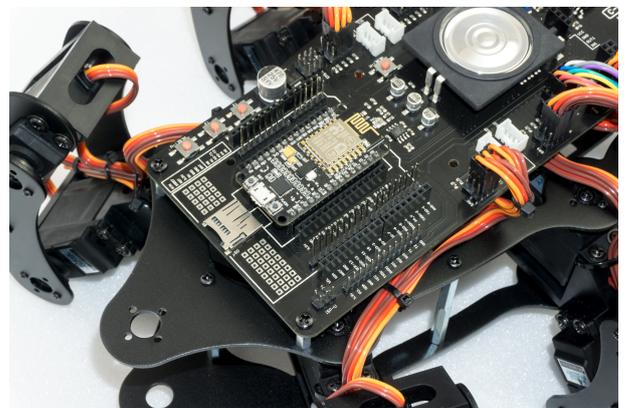


Figure 74



When buying, make sure that the distance between the pin strips is correct! There are two different versions on the market. One has a Silabs CP2102 as USB chip while the other version has a CH340G.

The version with the CH340G has a wider pin spacing and does not fit into the robot board! The NodeMCU board to be used must have a pin strip spacing of 23 mm.

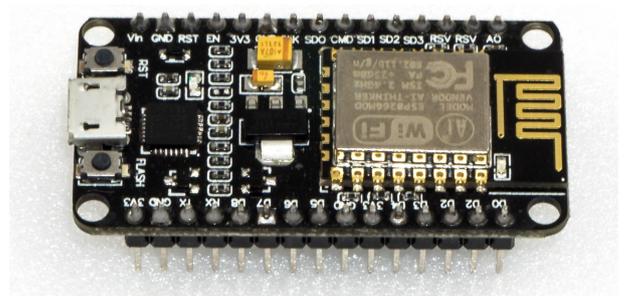


Figure 75

• **Single-Board-Computer (SBCs) and other Boards**

The robot board offers the possibility to mount an SBC like e.g., a Raspberry Pi 2/3 or compatible via board spacers. SBCs with a different hole spacing can be easily mounted with a self-made adapter board. Here the robot board offers enough space for proper extensions.

The pin header shown next to the "ISP-U" connection is a universal connection port for SBC or your own extensions. For the wiring between the SBC and the robot board, it is best to use the so-called "jumpwire" (socket - socket).

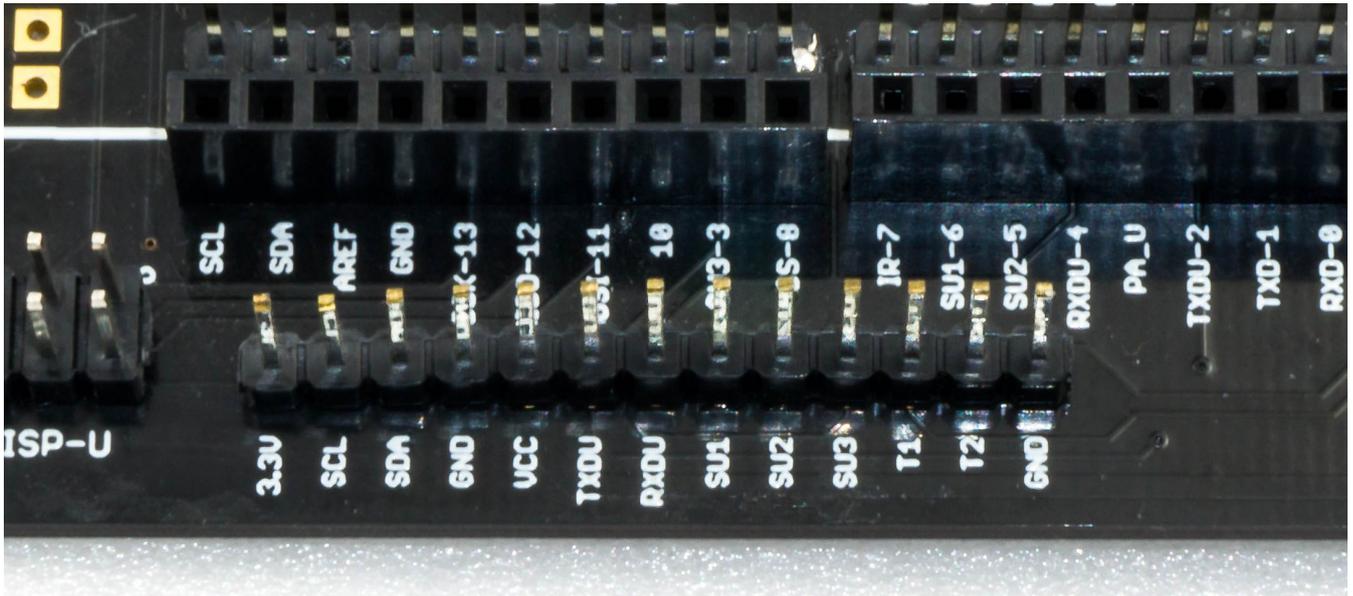


Figure 76

**Pin description (see inscription below the pins in Figure 76):**

3,3V	3.3 V/DC Supply input for the internal level converters of the robot board (Connect 3.3 V from the SBC in this case). If you use a board with a level of 5 V, then connect 5 V instead of 3.3 V (always connect the respective supply voltage; the SBC then supplies the level converters of the robot board with the correct voltage).
SCL	I <sup>2</sup> C clock line (connect SCL of the SBC here, the I <sup>2</sup> C bus is then available at sockets J2 and J3)
SDA	I <sup>2</sup> C data cable (here connect SCL of the SBC, the I <sup>2</sup> C bus is then available at sockets J2 and J3)
GND	Ground/GND (negative pole)
VCC	Output +5 V/DC, 1000 mA from the voltage regulator, here you can connect the supply of the SBC
TXDU	UART transmission line of the locomotion controller (connect the UART RxD of the SBC here)
RXDU	UART receiver line of the locomotion controller (connect the UART TxD of the SBC here)
SU1 - SU3	These connections lead directly to the pin header SU1, SU2, SU2. These pins can be used freely, e.g., for servos or digital signals.
T1 - T2	These pins are connected to the buttons T1 and T2. The buttons are externally equipped with pull-up resistors and provide a high signal when they are not actuated. The pull-up resistors can be deactivated with the board jumpers SJ6 and SJ7.
GND	Ground/GND (negative pole)

Figure 77 shows the robot and a Raspberry Pi 2 mounted on three 25 mm long circuit board spacers. The holes for mounting are already present in the robot board. The connection between the Raspberry Pi and the robot board was made using jumpwire connectors. The cables of the jumpers were placed under the Raspberry Pi to get a clean wiring.



Figure 77

→ It's possible that your SBC reacts very sensitively to voltage fluctuations and safe operation is not possible via the battery, which also supplies the leg servos.

In this case we recommend to supply the SBC via a small separate power bank. In practice, a battery capacity with a Powerbank of 2000 - 3000 mAh is sufficient to supply the SBC until the Hexapod battery is empty. Sometimes it is also sufficient to support the power supply directly at the SBC with a large capacitor of 2200  $\mu$ F (or more).

The reason for the unstable operation might come from a battery that is too small with high internal resistance, or from servos with very high current consumption or large current peaks.

# 18. Demo Programmes

To use the user boards easily and comfortably, use the Maker-Factory Hexapod library. This library was already installed at the beginning of this manual. The provided examples show the use of the user boards and the functionality of the Arduino Hexapod library.

When using user boards that are not Arduino compatible (like SBC), look at the file "Hexapod\_Lib.h". This file shows how the communication between the user board and the locomotion controller works. This will serve you as an example for developing your own function library. To do this, unpack the ZIP file "Library-Demos\Maker-Factory-Hexapod-Vx.1.zip" from the download bundle and look at the source code of the library in an editor such as Notepad or a development environment of your choice.

 The supplied examples are designed for "Arduino UNO" and "NodeMCU" boards programmed with the Arduino IDE.

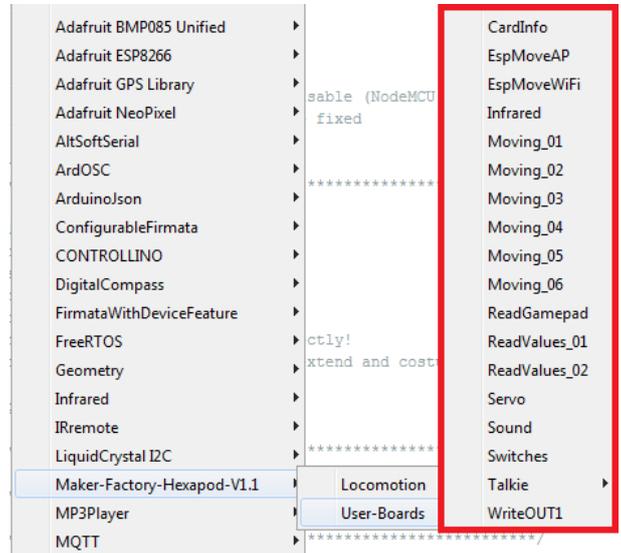


Figure 78

The Arduino examples show how to use the library. These can be found in the Arduino IDE under "Examples\Maker-Factory-Hexapod-Vx.x\...". Take your time to look at all the examples and test them on your Hexapod. Once you have understood the examples, you can start and create your own programmes with them. Take a closer look at the "Hexapod\_Lib.h". With some experience you can easily extend it with your own functions.

### Short description of the provided examples for Locomotion:

In the folder "Locomotion" you find the examples for the Locomotion controller (Arduino MEGA2560).

- **"LEDs"**  
Small test programme that alternately flashes the two LEDs (User and Live).
- **"Motion-Firmware"**  
Firmware of the locomotion controller.
- **"ServoCenter"**  
This programme moves the servos to the middle position. As needed for mechanical assembly of the Hexapod robot.

## Brief description of the supplied examples for user boards:

In the folder "User-Boards" you will find the examples for Arduino UNO and NodeMCU or compatible boards.



Before use, change the Arduino IDE to the correct Arduino board! Also, read the information in the comments to the respective examples!

→ The "Hexapod\_Lib.h" contains all functions that are sent and received from the UserBoard to the Locomotion-Controller. For the advanced developer it is worthwhile to have a look inside or to extend these with proper functions. The functions contain a detailed comment which parameters can be passed. The programme code for receiving the data, sending the data to the UserBoard and evaluating the gamepad can be found in the locomotion firmware in the file "Data\_Input.h".

The "Hexapod\_Lib.h" can also be downloaded from the product's website.

### • "CardInfo"

→ This example is intended only for Arduino UNO or compatible boards!

With this example you can check the function of the MicroSD card reader. Insert a MicroSD memory card with arbitrary content into the card slot and start the example. The terminal displays the contents of the memory card (file name). All original Arduino SD card examples work. You only have to change the pin for "CS" (Chipselect) as in this example. The terminal baud rate is 19200 baud.

### • "EspMoveAP"

→ This example is intended for NodeMCU boards only!

The example shows control via an HTTP server using a direct WiFi connection to the NodeMCU board. The programme sets up an access point with which you can connect to a smartphone, for example. The assigned IP address is displayed in the terminal. Type it into your browser. Then the browser loads the web interface to control the robot. You can then control the robot via WiFi by tapping or clicking the buttons. The example can easily be extended by you into other functionalities. The terminal baud rate is 115200 baud.

### • "EspMoveWiFi"

→ This example is intended for NodeMCU boards only!

The example shows the control via a web browser using a WiFi connection to the NodeMCU board. The programme establishes a connection to your router and the robot is then accessible within your network. The assigned IP address is displayed in the terminal. Type it into your browser. Then the browser loads the web interface to control the robot. You can then control the robot via WiFi by tapping or clicking the buttons. The example can easily be extended by you into other functionalities. The terminal baud rate is 115200 baud.

### • "Infrared"

→ This example is intended only for Arduino UNO or compatible boards!

The example reads an RC5 infrared code and outputs the values at the terminal. Plug the Jumper J8 in the direction of the label "IR-U". The universal remote control can serve as an IR transmitter set to RC5 (many Philips devices work with RC5), which is available for TV and audio equipment in stores. The terminal baud rate is 19200 baud.

### • "Moving\_01"

This example shows how data is sent to the locomotion controller without the Hexapod library. This example is well suited as a basis for using SBC. However, experience in programming is necessary! Have a closer look at the function SendData(). Here you can see how data is sent from the user board to the locomotion controller. The programme does not make an acknowledgement evaluation, but sends the data only to the locomotion controller without evaluating an answer from the locomotion controller.

### • "Moving\_02"

This example is ideal for getting started with the programming of user boards and shows the basic functions to control the Hexapod with the help of the library.

### • "Moving\_03"

This example shows how to use the ROBOT\_MOVE() function.

### • "Moving\_04"

This example shows how to use the ROBOT\_ROTATE\_MODE() and ROBOT\_TRANSLATE\_MODE() functions.

### • "Moving\_05"

This example shows how to use the ROBOT\_SINGLE\_LEG() function.

### • "Moving\_06"

This example shows how to interrupt a user programme using the gamepad. In this example, use the terminal to receive the messages from the robot. Set the robot to an elevation so that the legs move in the air without touching the ground. This allows you to observe the output when controlling the robot on the terminal. The terminal baud rate is 115200 baud.

### • "ReadGamepad"

The example shows how you can read the control data of the gamepad from the locomotion controller. If you plug in the jumper "J7", the locomotion controller no longer reacts to the gamepad. It is then possible to use the data only on the user board and to realise your own controls. The terminal baud rate is 115200 baud.

- **"ReadValues\_01"**

The example shows how you can send and read data between locomotion controller and user board without a library. This example is well suited as a basis for using SBC. However, experience in programming is necessary! The terminal baud rate is 115200 baud.

- **"ReadValues02"**

This example shows how to read data from the locomotion controller using the library. The terminal baud rate is 115200 baud.

- **"Servo"**

The example shows the use of additional servos which are controlled via the user board. The connectors SU1 to SU3 are used for this purpose. It is also possible to use the connectors SU1 to SU3 for other components such as sensors, buttons, switches, etc. The terminal baud rate is 115200 baud.

- **"Sound"**

The example shows a simple sound output via the loudspeaker.

- **"Switches"**

The example shows how the user buttons T1 and T2 are queried.

- **"Talkie"**

Talkie shows how speech output can be realised with the help of the Talkie library. Talkie is an implementation of the Texas Instruments ICs speech synthesiser on an Arduino board. This IC was very often used in the early 80s. This speech output was often found in expensive clocks, game computers, learning computers and many other devices. The unmistakable sound from the 80's is also available on the Hexapod Robot Board. The examples contain over 1000 prefabricated phrases that can be assembled to sentences. If the voice output is too quiet for you, you can increase the volume at the trimmer "Vol." on the Hexapod Robot Board. In addition, you can build a small "speaker box" for the speaker, this increases the volume significantly! A simple small box can do wonders here.

- **"WriteOUT1"**

This example shows how to read and write the IN1 and OUT1 inputs/outputs of the locomotion controller. This gives you an additional input and output to the user board. The pins are directly connected to the locomotion controller and can supply a maximum of 20 mA current. The maximum input voltage for the input is 5 V/DC.



5V & GND	Stabilised 5 V/DC/1000 mA are available at these pins. This connector is available for proper extensions and experiments ("5V" = positive pole, "GND" = negative pole).
3.3V PWR-LED	The red LED indicates that the 3.3 V voltage/power supply is working. It lights up when a power supply is connected to the BAT connector.
5V PWR-LED	The red LED indicates that the 5 V voltage/power supply is working. It lights up when a power supply is connected to the BAT connector.
J7	The jumper "J7" deactivates the evaluation of the PS2-compatible gamepad in the locomotion controller firmware (the robot can then no longer be controlled via the gamepad). The gamepad control commands can then only be read out via the user board!
ISP-M	ISP connector ("in-system programming") of the Locomotion controller. Via this connector the locomotion controller can be programmed via an ISP programming device. It is also possible to use this connector to integrate your own components with SPI interface.
PS2-1 - PS2-3	A PS2-compatible gamepad, available as an accessory, can be connected to these pins. With this controller the robot can be controlled manually, similar to a remote controlled car.
J2 & J3	The two sockets are connected to the I2C-BUS connector of the Locomotion controller and can be used for proper extensions. The connectors are compatible with the "SEEED-GROVE" components.
J4 & J5	The two sockets are connected to the I2C-BUS connector of the user board and can be used for proper extensions. The connectors are compatible with the "SEEED-GROVE" components.
RESET-M	The button triggers a hardware reset of the Locomotion controller when pressed.
S0 - S17	The "leg servos" of the Hexapod are connected to these connections. The pin headers are always arranged in blocks, which are assigned to the respective leg. The connections for "Coxa - Hip", "Femur - Thigh" and "Tibia - Shin" are counted from the respective label (S...). The pins are compatible with the most common servos. Make sure the polarity is correct. The polarity is marked on the board with "SIG" for the signal line, "BAT+" for the positive pole and "GND" for the negative pole.
SA0 - SA5	The analogue inputs SA0 to SA5 can be used for proper extensions in the locomotion firmware. In the "Motion Firmware" these can also be queried by the user board via a function.
SJ1	The PCB jumper "SJ1" connects the audio amplifier with the locomotion controller. In most practical cases, this connection remains in place. If necessary, you can carefully cut the jumper with a small wallpaper knife. With a soldering iron and some solder you can re-establish the connection.
Loudspeaker	The loudspeaker is used to output the signal tones and is connected to the audio amplifier. The signal tones can be generated by the locomotion controller and also by the user board.
Vol.	The trimmer "Vol." is used to adjust the volume of the signal tones of the locomotion controller and the user board. If you turn the trimmer in the direction of the buttons, the volume is increased; in the opposite direction it becomes quieter. In practice, a small watchmaker's screwdriver has been used to adjust the small trimmer. Be careful not to damage the trimmer when adjusting it! The basic setting is the middle position of the trimmer.
T1 & T2	The buttons "T1" and "T2" can be used with the user board. Please take a look at the software examples.
SJ6 & SJ7	The PCB jumpers "SJ6" and "SJ7" connect the outputs of the buttons "T1" and "T2" with a 22 kOhm pull-up resistor, so that no pull-up resistors have to be activated in the user board program. In most practical cases, this connection remains in place. If necessary, you can carefully cut the jumpers with a small wallpaper knife. With a soldering iron and some solder you can re-establish the connection.
RESET-U	The "RESET-U" button resets the user board.
SJ2	The PCB jumper "SJ2" connects the button "T2" with the NodeMCU user board. The used pin of the NodeMCU is the analogue input and can be disconnected from the push-button "T2" by this jumper if used otherwise. Normally this connection remains in place. If necessary, you can cut the jumper carefully with a small wallpaper knife. With a soldering iron and some solder you can re-establish the connection.
SU1 - SU3	The connectors SU1, SU2 and SU3 are connected to the user board slots and can be used for proper applications. They have the same polarity as the servo connectors of the leg servos. Please take a look at the software examples.
µSD	The card slot for a MicroSD card is connected to the Arduino-UNO compatible user board slot and can be used for own applications. Please take a look at the software examples.
ISP-U	ISP port ("In-System Programming") of the Arduino UNO compatible user board slot. This port is used to programme the user board using an ISP programmer. It is also possible to use this connector to integrate your own components with SPI interface.  → The Locomotion-Controller is located under the loudspeaker.

# 20. Schematic system overview (block diagram)

The diagram shows schematically the internal wiring and the interaction of the individual components of the robot board and serves as an aid for programming.

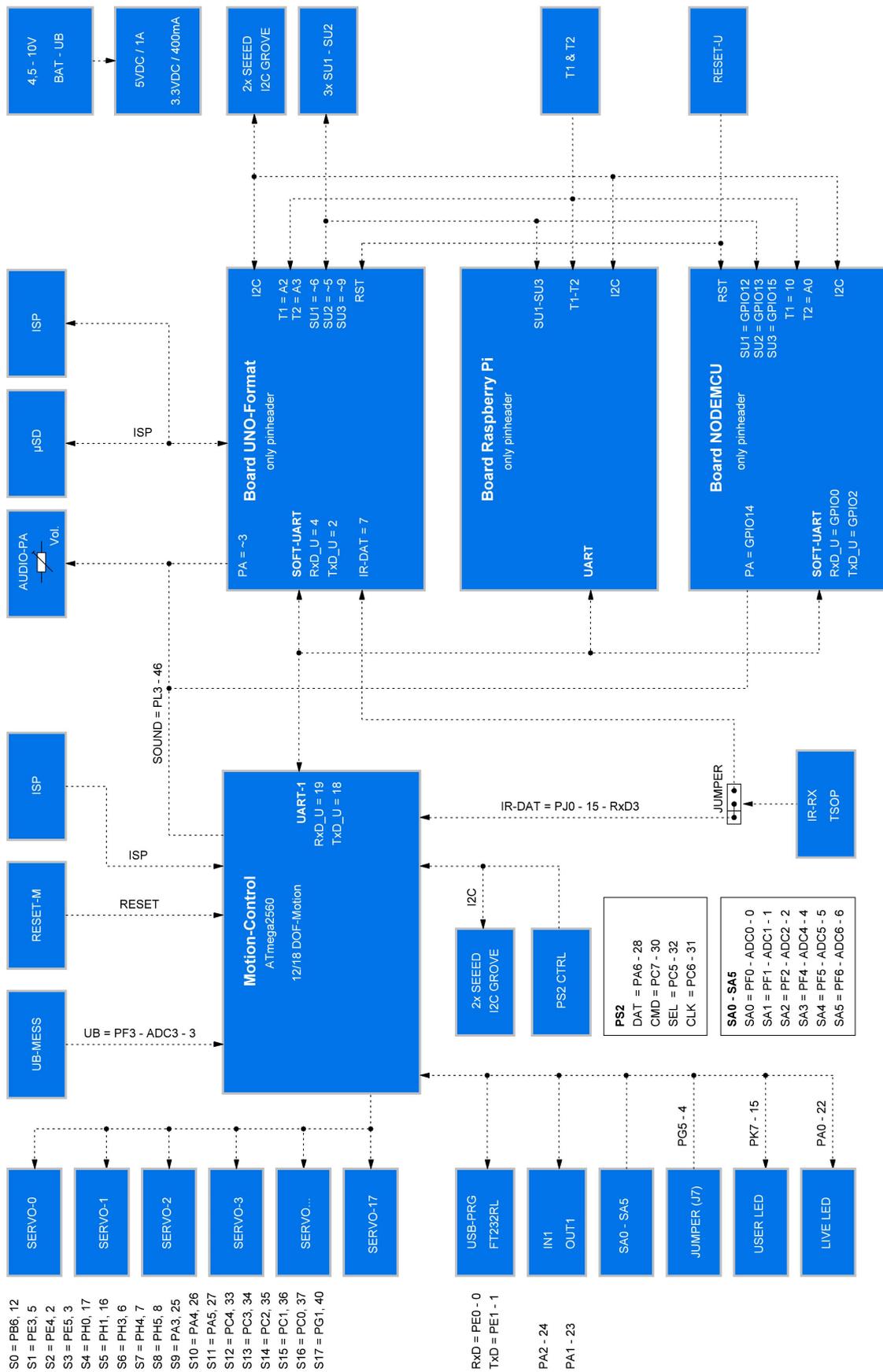


Figure 80

## 21. Pan unit

---

Optionally, an additional servo can be purchased, and the robot can be extended by a pan unit (swivel unit). The complete kit already includes an additional U-holder with accessories. Sensors or faces, for example, can be mounted on this. Please also have a look at the additional downloads for the Robobug. In the download area there are various 3D printable files available to expand your robot. The Pan unit is mounted in the same way as the legs are. Figure 81 shows the fully assembled Pan unit on the robot.

The servo can then be controlled via a user board, e.g. via a **Arduino™ UNO**. The servo is connected to one of the pin headers **SU1** to **SU3**.

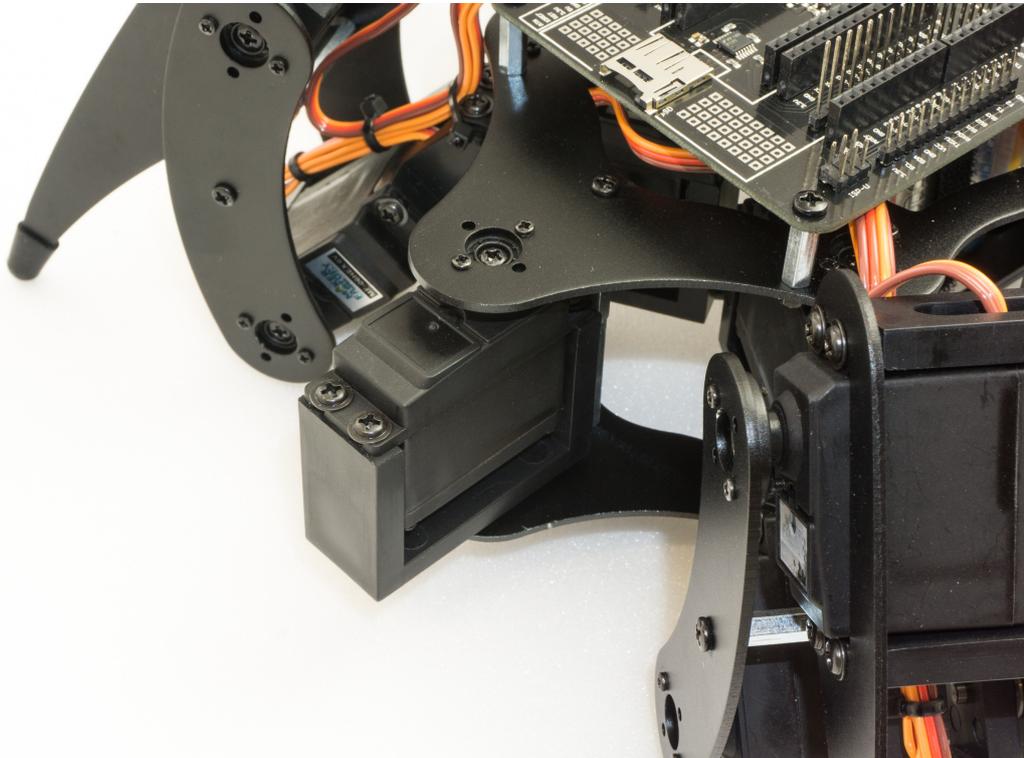


Figure 81

## 22. Pin assignment of the user boards

Locomotion controller (ATmega2560)		
	Port	Arduino
S0	PB6	12
S1	PE3	5
S2	PE4	2
S3	PE5	3
S4	PH0	17
S5	PH1	16
S6	PH3	6
S7	PH4	7
S8	PH5	8
S9	PA3	25
S10	PA4	26
S11	PA5	27
S12	PC4	33
S13	PC3	34
S14	PC2	35
S15	PC1	36
S16	PC0	37
S17	PG1	40
SA0	PF0	A0
SA1	PF1	A1
SA2	PF2	A2
SA3	PF4	A4
SA4	PF5	A5
SA5	PF6	A6
PS2_DAT	PA6	28
PS2_CMD	PC7	30
PS2_SEL	PC5	32
PS2_CLK	PC6	31
IR	PJ0	15
UB-MESS	PF3	A3
SOUND	PL3	46
Jumper J7	PG5	4
IN1	PA2	24
OUT1	PA1	23
RxD-U	PD2/RxD1	Serial1
TxD-U	PD3/TxD1	Serial1
LIVE-LED	PA0	22
USER-LED	PK7	69

Arduino user board		
	Port	Arduino
T1	ADC2	A2
T2	ADC3	A3
SCK	PB5	13
MISO	PB4	12
MOSI	PB3	11
SS_SD	PB0	8
PA	PD3	3
IR-DAT	PD7	7
SU1	PD6	6
SU2	PD5	5
SU3	PB1	9
RxD_U	PD4	4
TxD_U	PD2	2

NodeMCU user board		
	Port	Arduino
T1		10
T2	ADC0	A0
PA	D5	GPIO14
SU1	D6	GPIO12
SU2	D7	GPIO13
SU3	D8	GPIO15
RxD_U	D3	GPIO0
TxD_U	D4	GPIO2

## 23. Parts list

---

→ Some components such as screws, nuts and other small parts are included in the scope of delivery in larger quantities than necessary. These are used as replacements if, for example, a screw is lost during assembly.



Figure 82: 1x upper plate



Figure 83: 1x lower plate



Figure 84: 1x battery plate



Figure 85: 3x leg "Tibia" left + 3x leg "Tibia" right (always take care to round off the edges)



Figure 86: 12x leg "Femur"

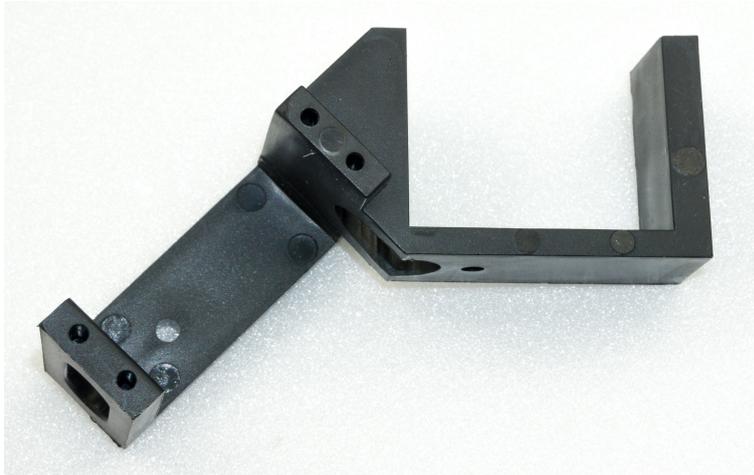


Figure 87: 3x "Coxa/Femur" servo mount right



Figure 88: 3x "Coxa/Femur" servo mount left



Figure 89: 7x U-angle "Tibia", servo holder



Figure 90: 18x servo discs



Figure 91: 18x leg servos

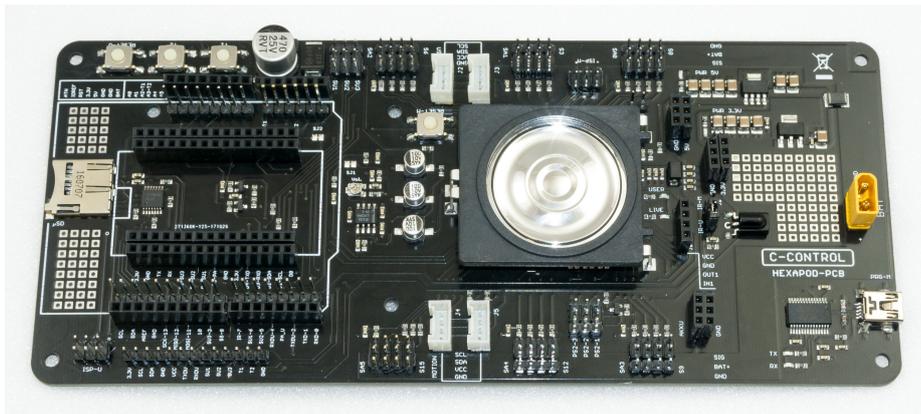


Figure 92: 1x Robobug board



Figure 93: 1x PS2 compatible gamepad



Figure 94: 1x PS2 Gamepad compatible receiver

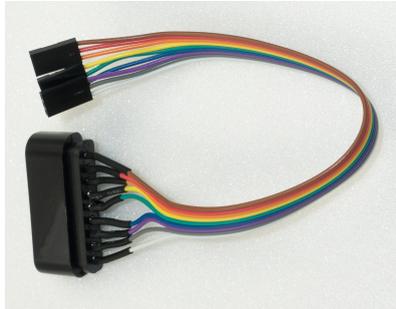


Figure 95: 1x 9-pin connection cable for receiver

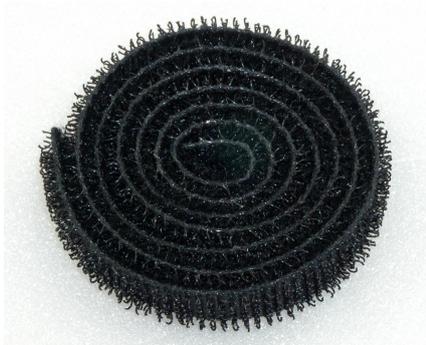


Figure 96: 1x Velcro fastener for fastening the battery pack



Figure 97: 1x Double-sided adhesive tape (servo tape) for mounting the receiver



Figure 98: 19x ball bearing with dowel pin for leg counter bearing



Figure 99: 25x cable ties for cable fixing

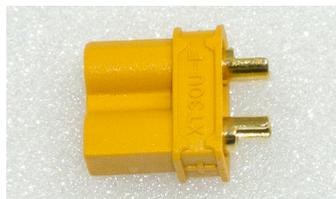


Figure 100: 1x XT30 battery plug to assemble your own battery



Figure 101: 24x 30 mm M3 spacer for legs and upper and lower plate



Figure 102: 4x 15 mm M3 spacer for PCB mounting



Figure 103a: 16 x Screw M3 x 6 mm PH2 (small head diameter 5 mm) for spacers and PCB mounting



Figure 103b: 18 x Screw M3 x 6 mm PH2 (large head diameter 5 mm) for spacers and PCB mounting



Figure 104: 72x Self-tapping screw M2.9 x 10 mm for servo mounting



Figure 105: 2x M4 x 6 mm PH2 Screw for battery mounting



Figure 106: 36x M2 x 8 mm screw for fixing the servo discs



Figure 107: 36x M2 Nut for mounting the servo discs



Figure 108: 16x M3 nut for fixing the spacers



Figure 109: 6x rubber caps for leg

## 24. FAQ

---

### Error compiling the examples

If there are problems compiling the firmware: The provided firmware as well as the examples were developed with Arduino version 1.8.3. Alternatively, you can download it from [www.arduino.cc](http://www.arduino.cc).

### Sound output too low

Check the setting of the "Vol." trimmer. Build a small "speaker box" into the robot board speaker. This increases the volume and improves the sound considerably.

### There is no communication with SPI devices and the user board

Set the Chip Select (CS) pin of the user board to "High" and check your connections at the ISP-U connector and if the used SPI component is supplied with power.

```
const int chipSelect = 8;
pinMode(chipSelect,OUTPUT);
digitalWrite(chipSelect,HIGH);
```

### The Gamepad does not connect or show any reaction

- Check the batteries in the transmitter.
- Is the receiver correctly connected to the robot board?
- Is the "motion firmware" installed on the robot board?
- Disconnect the power supply of the robot board, switch on the Gamepad and then re-establish the connection to the power supply of the Hexapod (plug in the battery).
- Is the Gamepad PS2 compatible?
- Jumper J7 must not be plugged in.

### The robot does not react to the gamepad

- Check the batteries in the transmitter.
- Is the receiver correctly connected to the robot board?
- Is the "motion firmware" installed on the robot board?
- Disconnect the power supply of the robot board, switch on the Gamepad and then re-establish the connection to the power supply of the Hexapod (plug in the battery).
- Is the Gamepad PS2 compatible?
- Is the jumper J7 removed?

### The robot is not running cleanly

- Have the legs been calibrated?
- Are the batteries sufficiently strong and charged?
- Is the robot too heavy?
- Have the servos been set to the middle position before installation?
- Are the legs slipping on the ground?
- Are all screws tightened?
- Does the mechanics wobble?
- Is the mechanics too stiff?

### Where can I find the circuit diagram?

The circuit diagram can be found in the "Schematic" folder in the download bundle or can be downloaded from [www.conrad.com](http://www.conrad.com).

**The robot carries out a reset after switching on or getting up.**

- Make sure that the battery is fully charged.
- Is the battery connection cable too thin (< 1.5 mm<sup>2</sup>) or longer than 20 cm?
- Is the battery plug soldered on cleanly?
- If the battery is too small, it cannot supply the necessary operating current for the servos.
- Is/are there any servo(s) blocked?
- Does the mechanics move too sluggishly?

## 25. Cleaning

---

The outside of the robot should only be cleaned with a soft, dry cloth or brush. Never use aggressive cleaning agents or chemical solutions as these may damage the surface of circuit board.

Protect the circuit board from dirt and humidity.

## 26. Declaration of Conformity (DOC)

---

We, Conrad Electronic SE, Klaus-Conrad-Straße 1, D-92240 Hirschau, hereby declare that this product conforms to the 2014/53/EU directive.

—> The full text of the EU conformity declaration is available via the following Internet address:

[www.conrad.com/downloads](http://www.conrad.com/downloads)

Select a language by clicking a flag symbol and enter the order number of the product in the search field; then you will be able to download the EU declaration of conformity in PDF format.

## 27. Disposal

---

### a) Product



Electronic devices are recyclable and do not belong in the household waste. Dispose of an unserviceable product in accordance with the relevant statutory regulations.



Remove any inserted batteries/rechargeable batteries and dispose of them separately from the product.

### b) Batteries/Rechargeable Batteries

As the end user, you are required, by law, (Battery Ordinance) to return all spent batteries/rechargeable batteries; disposal of them in the household waste is prohibited.



Batteries/rechargeable batteries contain harmful materials and are labelled with the symbol shown to indicate that disposal in the household waste is forbidden. The symbols of the relevant heavy metals are: Cd = cadmium, Hg = mercury, Pb = lead (the marking can be seen on the battery/rechargeable battery, e.g., underneath the refuse bin symbol shown on the left).

You can return your used batteries/rechargeable batteries free of charge at the official collection points of your community, in our stores, or at places where batteries or rechargeable batteries are sold.

You thereby fulfil your statutory obligations and contribute to the protection of the environment.

## 28. Technical Data

---

### a) General information

Body length.....	280 mm
Body height without PCB.....	65 mm
Body width without PCB.....	150 mm
Coxa length.....	50 mm
Femur length.....	75 mm
Tibia length.....	115 mm
Weight.....	with battery approx. 2.7 kg
Servo actuating force.....	at 6 V/DC: 13 kg
Number of degrees of freedom DOF (servos).....	18
Recommended battery pack.....	5 cells NiMH, 3700 mAh (connection cable 2.5 mm <sup>2</sup> , plug XT30)

### b) Control electronics

Operating voltage.....	4.5 - 10 V/DC
Current consumption in robot board only.....	approx. 80 mA
Number of servo connections.....	18 locomotion controllers + 3 user boards
Programming.....	Arduino IDE (C/C++)
Programming interface.....	USB
User board slots.....	3
I2C connectors Locomotion controller.....	2
I2C connectors User-Board.....	2
IR receiver.....	1x 38 kHz (only usable in combination with Arduino UNO slot or compatible boards)
Audio amplifier with loudspeaker.....	1
MicroSD card slot (*).....	1 (can only be used in conjunction with Arduino UNO slot or compatible boards)
Locomotion Controller.....	1 (Atmel ATmega2560 - Arduino MEGA compatible)
Reset button Locomotion controller.....	1
User board reset button.....	1
ISP connection Locomotion controller.....	1
User board ISP connector.....	1
PS2 compatible gamepad connector.....	1
Internal power supply.....	3.3 V/DC and 5 V/DC
Current carrying capacity 3.3 V/DC.....	400 mA
Current carrying capacity 5 V/DC.....	1000 mA
Dimensions (L x W x H).....	200 x 88 x 23 mm
Board weight of circuit board only.....	approx. 100 g
Working temperature range.....	0 °C to +40 °C

### c) Gamepad

Frequency range.....	2.402 – 2.521 GHz
Transmission power.....	10 mW
Power supply.....	integrated Li-Ion battery, rechargeable via USB







This is a publication by Conrad Electronic SE, Klaus-Conrad-Str. 1, D-92240 Hirschau ([www.conrad.com](http://www.conrad.com)).

All rights including translation reserved. Reproduction by any method, e.g. photocopy, microfilming, or the capture in electronic data processing systems require the prior written approval by the editor. Reprinting, also in part, is prohibited. This publication represent the technical status at the time of printing.

Copyright 2018 by Conrad Electronic SE.

1661451\_V2\_1018\_01\_VTP\_m\_en