Описание системы

«Автономный гексапод»

Подготовил Yauheni Liauchuk [unstutby@gmail.com](mailto:unstutby@gmail.com) 24.01.2019

Содержание

[Вступление 2](#_Toc536407934)

[Обзор системы 2](#_Toc536407935)

[Механическая платформа 2](#_Toc536407936)

[Плата управления сервомашинками 2](#_Toc536407937)

[Компьютер 3](#_Toc536407938)

[Лидар 3](#_Toc536407939)

[Питание 3](#_Toc536407940)

[Программное обеспечение 3](#_Toc536407941)

[Структурная схема 4](#_Toc536407942)

[Механическая платформа 4](#_Toc536407943)

[Используемые термины 4](#_Toc536407944)

[Размеры и углы 5](#_Toc536407945)

[Плата управления сервомашинками 6](#_Toc536407946)

[Сервомашинки 6](#_Toc536407947)

[Калибровка 6](#_Toc536407948)

[Лидар 7](#_Toc536407949)

[Компьютер 7](#_Toc536407950)

[Вход по VNC 7](#_Toc536407951)

[Программное обеспечение 7](#_Toc536407952)

[Координатная система 7](#_Toc536407953)

[Исходный код 7](#_Toc536407954)

[Краткое описание классов 8](#_Toc536407955)

[Полезная информация 8](#_Toc536407956)

[Принцип работы сервомашинок 8](#_Toc536407957)

[Прошивка платы управления 8](#_Toc536407958)

# Вступление

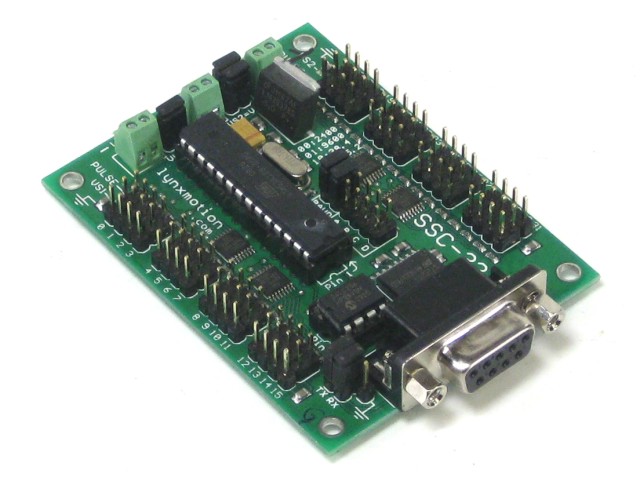
Система «Автономный гексапод» (далее просто «Паук», несмотря на количество ног) предназначена для обучения робототехнике. На момент написания этой документации система представляет собой работающее сочетание механической, электронной и программной частей, которое позволяет «Пауку» самостоятельно перемещаться по помещению и избегать препятствий. Возможна модификация любой из этих составляющих – добавление новых и замена старых частей системы, изменение алгоритмов. Для успешной работы над дальнейшей модификацией «Паука» заинтересованным студентам желательно представлять, как работает вся система в целом, и как работает часть, которую вы хотели бы модифицировать. Цель этого документа – дать такое представление, чтобы упростить и ускорить начало работы с «Пауком». Описаны составляющие системы, их взаимодействие, принципы работы, а также некоторые наблюдения и советы автора; даны ссылки на инструкции, справочные материалы и полезные статьи.

# Обзор системы

## Механическая платформа

В системе используется платформа [Lynxmotion CH3-R](http://www.lynxmotion.com/c-101-ch3-r.aspx) – гексапод с тремя степенями свободы в каждой конечности (т.е., 6 ног с 3 суставами, всего 18 суставов). Каждый из суставов может быть повёрнут на произвольный угол, за счёт чего можно реализовать различные алгоритмы походки, поворотов и т.п. Суставы представляет собой сервомашинки модели HS-475/[HS-485](https://servodatabase.com/servo/hitec/hs-485hb). Внутри платформы имеется свободное пространство, в котором располагается электроника и аккумуляторы. Для удобного и аккуратного крепления всех внутренних частей изготовлена пластиковая рама. Сверху закреплён [лидар](#_Лидар).

## Плата управления сервомашинками

Управление 18-ю сервомашинками одновременно – достаточно нетривиальная задача. Плата управления сервоприводами [SSC-32](http://www.lynxmotion.com/p-395-ssc-32-servo-controller.aspx) берёт на себя эту задачу и существенно упрощает программирование «Паука». Плата управления SSC-32 принимает команды через последовательный интерфейс (UART, 5v). В команде, в частности, можно указать, на какой угол должен быть повёрнут конкретный сустав и с какой скоростью он должен повернуться в это положение. Плата может управлять 32 сервомашинками, для конечностей «Паука» использовано только 18, так что возможно простое добавление дополнительных механических устройств на основе сервомашинок (например, клешни-захвата).

## Компьютер

Основные алгоритмы (обработка данных с лидара, походка) выполняются на одноплатном компьютере [Raspberry Pi 3 B](https://www.raspberrypi.org/products/raspberry-pi-3-model-b/). Компьютер работает под управлением операционной системы Raspbian (Linux). Мощность компьютера избыточна для текущих задач и позволяет даже подключить к нему камеру (имеется в лаборатории, использовалась в предыдущей версии системы «Паук») и обрабатывать видео в реальном времени. К компьютеру можно подключить монитор, клавиатуру, мышь, но это может потребовать разборки корпуса «Паука», так что для разработки удобнее использовать удалённый доступ через VNC.

## Лидар

Сверху платформы закреплён лидар [SLAMTEC RPLIDAR A1](http://www.slamtec.com/en/lidar/a1). Лидар имеет вращающуюся часть, в которой расположен инфракрасный лазер и фотодиодная линейка, с помощью которых лидар может измерять расстояние до препятствия в которое попдает луч. Максимальное расстояние, измеряемое данным лидаром – 12 метров. Связь с компьютером осуществляется через UART интерфейс (через переходник UART<->USB).

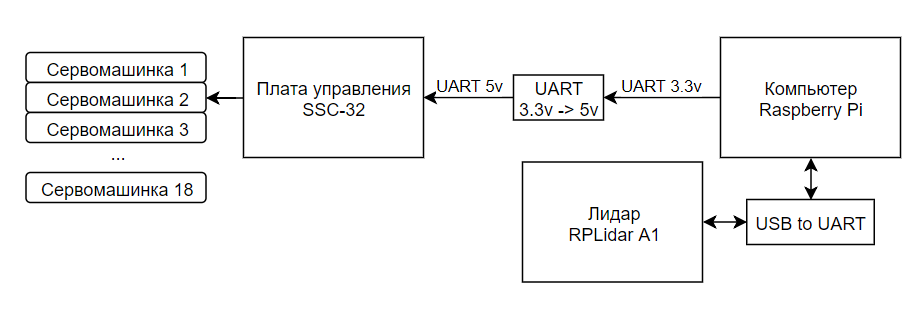
## Питание

Питание системы осуществяется от LiPo аккумулятора 11.1 В. Плата управления SSC-32 требует питания напряжением 6 В (определяется сервомашинками, при использовании других сервомашинок напряжение может быть другим). Компьютер Raspberry Pi и лидар питаются напряжением 5 В. Для эффективного понижения напряжения аккумулятора в систему включен DC-DC преобразователь.

## Программное обеспечение

Программа, управляющая «Пауком», написана на языке C++. Разработка происходит в IDE Code::Blocks непосредственно в компьютере Raspberry Pi. При работе программа постоянно принимает данные от лидара и вырабатывает команды для движения «Паука».

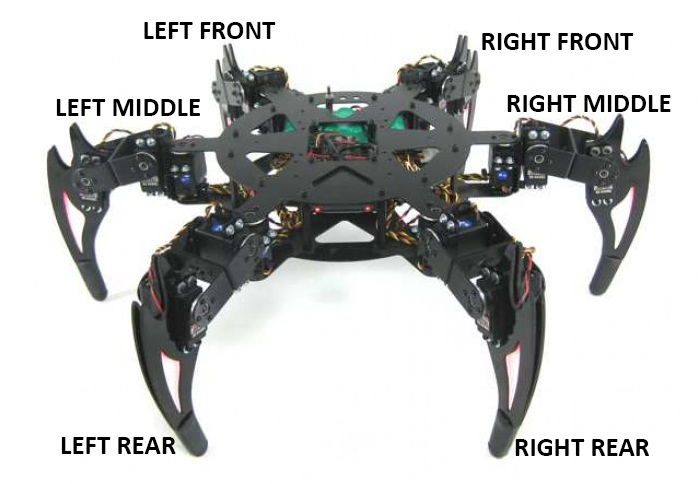
# Структурная схема



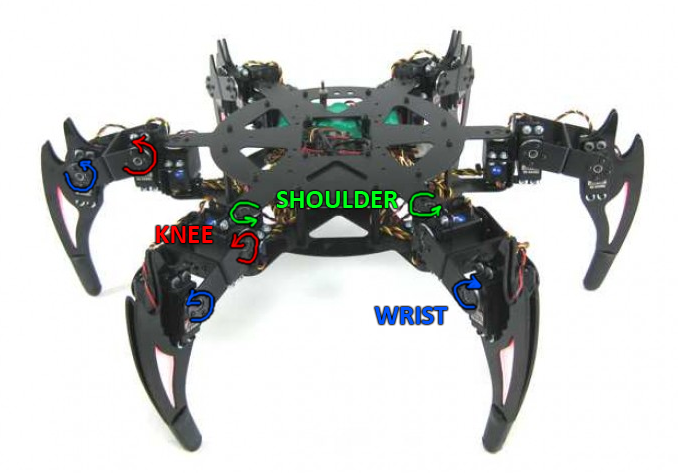
# Механическая платформа

## Используемые термины

В коде программы и конфигурационном файле используется следующие названия для ног «Паука»:

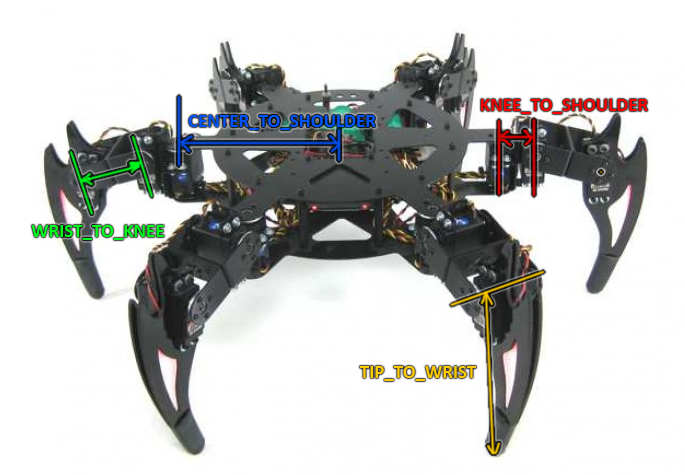


Для обозначения конкретных суставов используются следующие названия:



## Размеры и углы

Чтобы переместить конец ноги (tip) в заданную точку пространства, необходимо вычислить углы поворота всех суставов этой ноги. Для этого необходимо знать некоторые размеры. Эти размеры были измерены и сохранены в конфигурационном файле (config.txt в папке программы).



Размеры в милиметрах:

TIP\_TO\_WRIST=140  
WRIST\_TO\_KNEE=56  
KNEE\_TO\_SHOULDER=28  
CENTER\_TO\_SHOULDER=166  
KNEE\_TO\_BOTTOM=50

# Плата управления сервомашинками

Исчерпывающую документацию по плате SSC-32 можно найти на сайте производителя: <http://www.lynxmotion.com/images/data/ssc-32.pdf> . Важно: в документации отсутсвует информация о бинарных командах, которые используются в «Пауке» для экономии пропускной способности UART-интерфейса. Информация об этих командах находится тут: <http://www.lynxmotion.com/images/html/build177.htm>

Каналы управления нумеруются от 0 до 31 (подписаны на плате). Соответствие между номером канала и конкретным суставом задаётся в конфигурационном файле, например:

LEFT\_FRONT\_WRIST\_CHANNEL=0  
LEFT\_FRONT\_KNEE\_CHANNEL=1  
LEFT\_FRONT\_SHOULDER\_CHANNEL=2

# Сервомашинки

## Калибровка

Сервомашинки могут быть закреплены внутри ног «Паука» в произвольном положении, поэтому для корректной работы необходима калибровка, т.е. установление соответствия длительностей импульсов (см. [принципы работы сервомашинок](#_Принцип_работы_сервомашинок)) и углов поворота суставов. Также каждая сервомашинка может быть подключена к любому из 32 каналов платы управления. Соответствие задаётся в конфигурационном файле. Максимальные и минимальные углы поворота суставов задаются для всех конечностей сразу:

KNEE\_MIN\_ANGLE=35  
KNEE\_MAX\_ANGLE=135  
WRIST\_MIN\_ANGLE=20  
WRIST\_MAX\_ANGLE=100  
SHOULDER\_MIN\_ANGLE=-90  
SHOULDER\_MAX\_ANGLE=90

Далее для каждого сустава(сервомашинки) указываются соответствующие длительности и номер канала:

LEFT\_FRONT\_WRIST\_CHANNEL=0  
LEFT\_FRONT\_WRIST\_MIN=900  
LEFT\_FRONT\_WRIST\_MAX=1700

LEFT\_FRONT\_KNEE\_CHANNEL=1  
LEFT\_FRONT\_KNEE\_MIN=2000  
LEFT\_FRONT\_KNEE\_MAX=1000

LEFT\_FRONT\_SHOULDER\_CHANNEL=2  
LEFT\_FRONT\_SHOULDER\_MIN=550  
LEFT\_FRONT\_SHOULDER\_MAX=2370

При замене неисправной сервомашинки на новую необходимо опытным путём установить новые длительности (отправляя команды на плату управления добиться нужного угла) и скорректировать информацию в конфигурационном файле.

# Лидар

Общее описание, характеристики:  
<http://bucket.download.slamtec.com/8e7a1f4490a235717b43fccaf7dcae325dda7dc8/LD108_SLAMTEC_rplidar_datasheet_A1M8_v2.1_en.pdf>

Инструкция по использованию:  
<http://bucket.download.slamtec.com/b42b54878a603e13c76a0a0500b53595846614c6/LR001_SLAMTEC_rplidar_protocol_v1.1_en.pdf>

# Компьютер

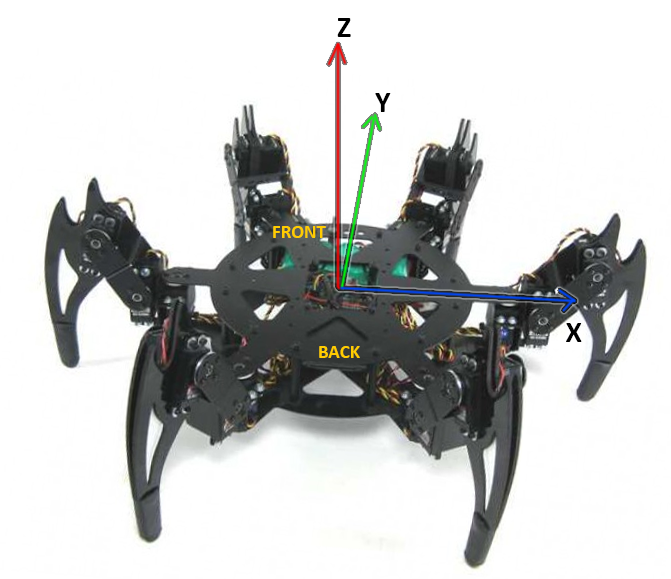
## Вход по VNC

Для входа на удалённый рабочий стол через VNC используется логин root и пароль по умолчанию raspberry. В случае каких-либо проблем с подключением через сеть для разрешения ситуации можно подключить к Raspberry Pi клавиатуру и монитор.

# Программное обеспечение

## Координатная система

Все размеры внутри программы исчисляются в милиметрах. Углы могут исчисляться в радианах (при любых тригонометрических рассчётах), градусах (**только** в конфигурационном файле) и в нормализованном диапазоне от 0 до 1 (минимальный/максимальный возможный угол поворота сустава). Направления отсчёта нормализованных углов поворота суставов соответсвует [рисунку с названиями суставов](#_Используемые_термины). Координатная система, принятая в программе, выглядит следующим образом:



## Исходный код

Все исходные коды доступны тут: <https://github.com/levevg/hexapod>

В паке Firmware находится код программы, выполняемой непосредственно на компьютере «Паука». Для работы с этим проектом используется IDE Code::Blocks.

В папке Simulation находится код программы, которая была создана для возможности проверки алгоритмов «Паука» на персональном компьютере (без самого «Паука»). Симулятор состоит из двух частей – Hexapod (практически полностью совпадает с проектом Firmware, симулирует ПО «Паука») и Environment (симулирует «железо» «Паука» и помещение). Для работа с этими проектами используется IDE CLion или любая другая, поддерживающая CMake. Для успешной компиляции проекта Environment необходимо скачать и скомпилировать библиотеку freeglut.

## Краткое описание классов

Config – класс для загрузки данных из конфигурационного файла

Serial – класс для передачи и приёма данных через последовательный интерфейс

Joint – класс сустава (сервомашинки) с возможностью задавать угол поворота в диапазоне от 0 до 1 (калибровочные данные загружаются в классе Config)

Limb – класс конечности с возможностью перемещать конец ноги в заданные декартовы координаты (расчёт необходимых для этого углов).

Hexapod – класс, собирающий вместе все суставы и конечности, и предоставляющий функции для ходьбы, подъёма и т.п.

Behavior – базовый класс алгоритма поведения «Паука»

SquareTestBehavior – простой пример класса поведения. «Паук» перемещается по траектории в виде квадрата.

LidarBehavior – класс поведения на основе информации от лидара. Пытается идти вперёд избегая препятствий.

LidarAPI – функции для приёма и передачи данных лидара

# Полезная информация

## Принцип работы сервомашинок

<http://www.lynxmotion.com/images/data/ssc-32.pdf>, страница 4

<http://easyelectronics.ru/servomashinka-hs-311.html>

## Прошивка платы управления

Исходные коды прошивки микроконтроллера платы SSC-32 отсутсвуют в свободном доступе. В случае выхода микроконтроллера из строя (такое уже однажды случалось) необходимо заменить его и прошить в него bootloader (его можно найти в папке SSC-32 Bootloader репозитория с иходным кодом проекта). После этого перемычки на плате нужно установить для связи с компьютером через COM порт, подключить её к компьютеру и прошить firmware используя программу производителя (её, а так же прошивку можно найти на сайте <http://www.lynxmotion.com/c-15-software.aspx> ). Помните, что версия прошивки должна поддерживать режим двоичных команд (на сайте производителя имеются прошивки не поддерживающие этот режим).