Design, Construction and Control of Hexapod Walking Robot

Marek Zak
Faculty of Information Technology
Brno University of Technology
Brno 612 66
Email: izakmarek@fit.vutbr.cz

Jaroslav Rozman
Faculty of Information Technology
Brno University of Technology
Brno 612 66
Email: rozmanj@fit.vutbr.cz

***Abstract*—This paper deals with design, construction and control of a hexapod (i.e. six-legged) walking robot. Basic characteristics of legged robots, a few existing robots and their pros and cons are described in the introduction of this paper. Next, basic gaits, which are used by legged robots for their locomotion are mentioned here. Main part of the paper is focused on the result of our project - on the six-legged robot, which can walk using tripod, wave and ripple gait and which is equipped with sonars, force-sensitive resistors and encoders. This robot is controlled and monitored from an user interface program, which can display data from the sensors and the positions of robot legs. The robot can be used to test and verify algorithms, gaits and features of hexapod walking robots.**

I. Introduction - Why legged robots?

Robots can be found everywhere. One of the most important part of a robot is its chassis. There are several basic chassis types: wheeled, tracked and legged chassises. Wheeled chassises are fast, but not suitable for rough terrain. Tracked chassises are slower, but more suitable to rugged terrain. Legged chassises are quite slow and more difficult to control, but extremely robust in rough terrain. Legged chassises are capable to cross large holes and can operate even after losing a leg [1]. Many researches were performed in this field in past few years, because of its large potential. Legged chassises are especially ideal for space missions [2]. For example the ATHLETE developed by NASA [3]. There are also several projects in military research like LS3 by Boston Dynamics [4].

We aim to create a cheap legged platform, which would allow research and testing of gaits and walking chassis mechanisms. Create a system with many sensors that allows the chassis all known movement or behavior. The robot should be driven from wirelessly connected computer and should send all available data from sensors, which will be displayed on the computer in the user interface program. This platform should be universal, the user could connect to the robot and drive it and send his own data to the user interface program of the control computer.

Part of the problem is the design and implementation of hexapod gait algorithms such as tripod, wave or ripple (Figure 2) [2]. It is also important to create and program a system into the microcontroller unit (MCU) of the robot, which would be able to perform given algorithm by adjusting the servomotor angle at the right time.

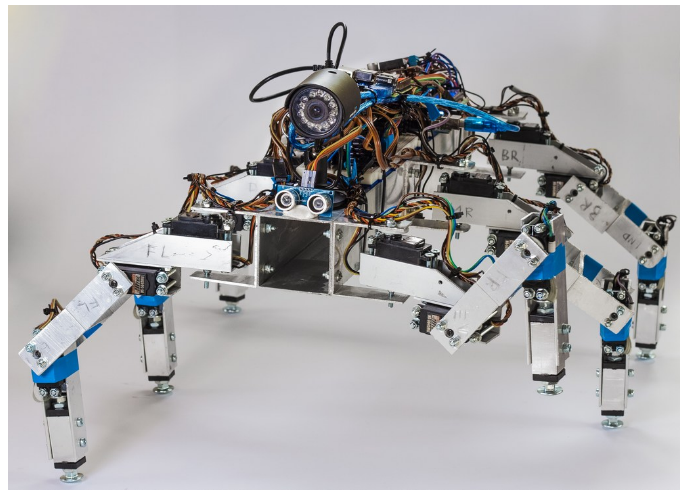

Fig. 1. Hexapod. This robot was designed and constructed during the project. It is equipped with sonars, camera, LCD display and more accessories.

There are several companies, which are producing hexapod robot models and platforms. Name Lynxmotion [5] or Trossen Robotics [6]. Both companies offer a variety of hobby and research level robot kits and parts. They also offer several types of hexapods. These robots differ in the body shape and leg construction. All robots come with software, which provides control of servomotors using inverse kinematics and creating custom gaits. Robot kits are sold for about $ 1,000, depending on the version.

Although several solutions already exist and have great potential, each one of them has some disadvantage. The first one is price, which is quite high, about $ 1,000 for robot with electronics, remote controller and motors. Another disadvantage is equipment of the robots. Most of the available robots have limited expansion options, like missing ground sensors, which are difficult to install later, or servomotor type with insufficient power or features. Also the batteries are often built in the body and it is difficult or even impossible to remove them.

Because of these disadvantages, we decided to build a new robot (Figure 1). This new robot is capable of the same movements like commercial versions and tries to remove their negatives. Robot is made of aluminium profiles, because of their easy availability and sufficient strength. Unlike commer-

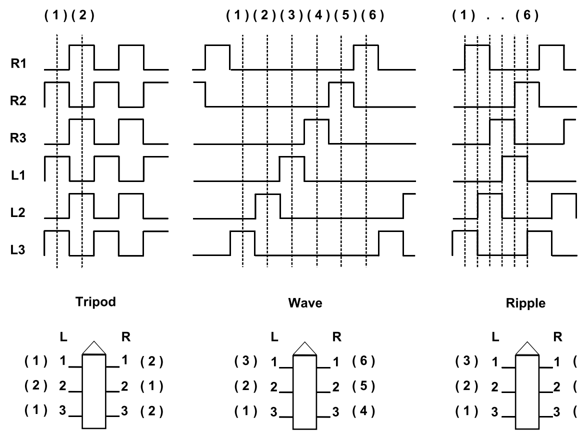

Fig. 2. Walking gaits. The chart shows the movement of each leg in time. A high value represents leg movement, low values means no movement. Tripod, wave and ripple gaits are shown in this figure. Tripod has two group of legs, all the legs in the same group move at once. In the wave gait only one leg is moving forward at any time. After all legs are set up to their new positions, step is completed. In the ripple gait all legs move the same way, but their moves are shifted.

cial versions, this hexapod has also a wide variety of sensors and equipment, like ultrasonic sonars, LCD display, encoders, foot sensors or camera. All the information from the sensors are sent to the computer and displayed in the UI program. It is also possible to use a library, which allows reading of all the data from the robot and sending commands to it.

This robot can walk using tripod, wave or ripple gait and is also capable of rotation. Each leg is equipped with a force-sensitive resistor to detect ground and each servomotor has an encoder to determine joint's current position. Sonars are able to detect obstacles to avoid collisions. All events and information can be monitored in Hexapod Control Room (Chapter V-A).

II. Characteristics of Legged Robots

This part of the paper focuses on several characteristics of walking robots. Some classifications of walking robots and most common walking gaits are described.

There are many ways how to classify walking robots – by a body shape [7], number of legs, number of degrees of freedom per leg or locomotion technique. Various options can be combined to achieve many different configurations.

At least two degrees of freedom are needed to construct a walking robot – the first for lifting the leg, second for rotating it. Nevertheless there should be three degrees of freedom for a good functioning chassis, because the legs move along a circle and the forward movement of the body causes slipping between the foot and the terrain, which can be compensated by third joint [8], [9].

Walking chassis movements can be divided into statically stable and dynamically stable [1], [9]. Static stability represents the ability of the chassis to remain in a stable position in every moment of movement. Static stability is typical for hexapod, which is always stable during its movement. Dynamically stable chassis is sometimes out of balance – balancing or falling. This is common for two-legged robots.

A. Walking Gaits

Walking gait refers to the locomotion achieved through the movement of robot legs. Compared to human gait, the legged chassis usually has more than two legs. Therefore, the locomotion of a robot is much more complicated. There are several basic gaits, such as tripod, wave or ripple (Figure 2).

Tripod gait is based on two groups of legs. During each step the first group of the legs is lifted and is rotated forward and is laid upon on the ground. Then the other group is lifted. Now both groups are moving, the first group backward, the second group forward and finally the second group is laid on the ground. It is obvious that both groups perform the same movement, but they are shifted by half a period. Tripod gait is very fast, but also very unstable. That is because at one moment half of the whole weight of the robot is only on one leg, which can lead to slip or even to fall.

Another gait is wave, which is the most stable gait, but also the slowest. Wave gait consists of a sequential adjustment of legs forward and only when all the legs are set to the new positions, the step is completed. In each phase of step maximally one leg is lifted up, which leads to high stability of this gait.

Ripple gait is inspired by insects. Each leg performs the same move – up, forward, down, backward. Leg moves partially overlap. In other words, the time when the first foot is lifted and begins to move forward, the second leg begins to lift up. In this way the robot cycles through all legs.

These are the most common gaits. But, theoretical number of different gaits N can be calculated using Equation 1. Not all of them are usable for effective locomotion.

$$N = (2K - 1)! \tag{1}$$

Where K is the number of the legs of the robot. For hexapod robot it is 11! = 39 916 800 possibilities of locomotion [10]. The number is quite large, because this equation calculates all possible motions, like motion up and down, which of course doesn't lead to an effective movement.

III. Body Construction

We constructed a hexapod robot during this project. This hexapod has rectangular body type – it has two groups of legs, three on each side. Each leg has three degrees of freedom and is powered by hobby servomotors HS-5485HB on coxa and tibia joints and HS-5645MG on femur joint (Figure 4). Servomotors are equipped with encoders and leg bases are equipped with force-sensitive sensors to detect ground (Figure 3).

Servomotors must be sufficiently powerful, depending on the desired behavior. If tripod gait is required, then each motor on the middle legs must be powerful enough to hold half of the weight of the robot.

It is important to choose a suitable material for the body. It must be solid enough, but not too heavy. Therefore aluminum profiles were chosen for the body construction. They are quite light-weighted and solid enough. In addition, they are available in various sizes and shapes and are easy to handle. Robot is made of 25 mm and 60 mm profiles. Structure of the leg is

Fig. 3. Force-sensitive resistors placement. The inner profile is movable in digits of singles a millimeter. This ensures perfect functionality of the force-sensitive resistor.

shown in Figure 4. The robot is 70 cm long, 47 cm wide and 6 cm high and weighs 4.3 kg.

The price of components on the robot (body, servomotors, MCU) is about $ 500, which is half the price of the commercial version. Another $ 80 costs Raspberry Pi [11], sensors and accessories.

IV. Robot Electronic System

In order to control the robot there must be some control unit. The open-source electronic platform Arduino Mega 2560 [12], which is based on MCU Atmega2560 [13], was chosen to drive the servomotors and sensors on this robot. Arduino board is connected to the Raspberry Pi via USB cable. The whole scheme is in Figure 5. All sensors like sonars, LCD display, memory card, GPS module or force-sensitive resistors are connected to Arduino. LCD display is connected by digital pins using integrated Hitachi HD44780 driver, which allows 4-bit or 8-bit mode. The 4-bit mode requires seven I/O pins from the Arduino, while the 8-bit mode requires 11 pins. There are also 18 servomotors connected to digital pins and driven by MCU's timers.

Raspberry Pi is a miniature computer the size of a credit card, to which a standard monitor, a keyboard and a mouse can be connected. It has extremely low power consumption (max. 3.5 W) and can run linux based operating system Raspbian. There are several models, which differ in RAM, the number of USB ports or GPIO pins. Raspberry Pi is equipped with a USB Wi-Fi dongle, which is connected to a wireless network, and runs Qt [14] client program, which is able to find the IP address of the server and gets connected to it. Sensor data are sent to a computer after successful connection. Client is also capable of reconnection after disconnection.

All sensors and servomotors are connected and driven by Arduino board. Servomotors are controlled by MCU's timer using VarSpeedServo library [15].

Robot is also equipped with ultrasonic distance sensors HC-SR04 – sonars. These sensors can measure distance from 2 cm to 400 cm. It has 4 pins – Vcc, ground, trigger and echo. Robot is equipped with eight sonars, three in front, three in back and two on sides. Sonars are connected using MCU's digital pins and a multiplexer.

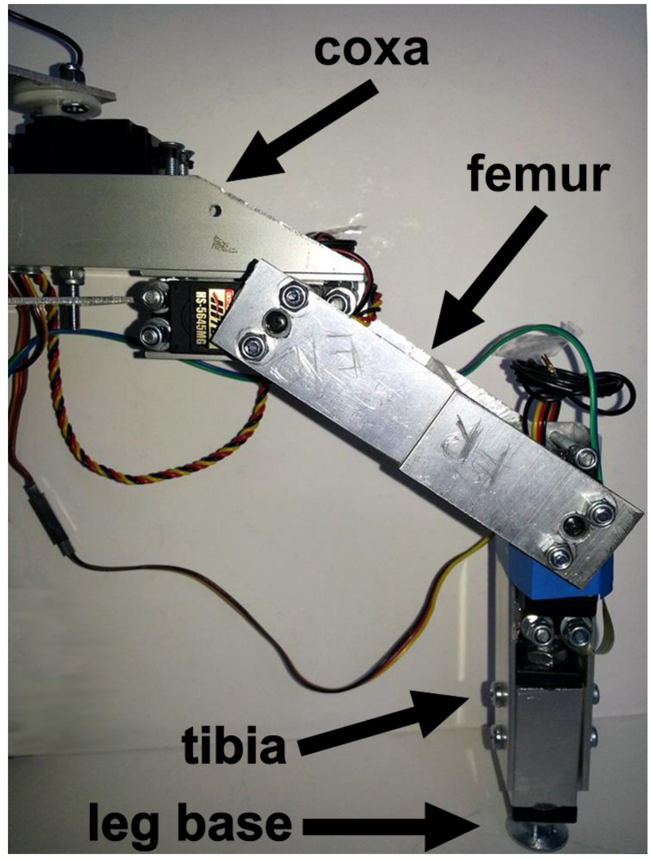

Fig. 4. Leg structure. From left: body is connected to the coxa using coxa joint, which allows forward and backward rotation. Coxa is then connected to the femur and femur is connected to the tibia, which allows lifting and laying. [8]

If the robot should walk across rugged terrain, it must be able to detect ground. Therefore each leg is equipped with force-sensitive resistors, which are better than tactile switches, because they can detect level of pressure on the leg. This is used for robot calibration. To determine the pressure force-sensitive resistor is connected to A/D converter using voltage divider.

The robot also has a GPS module to determine its position and a SD card to store logs and configuration files.

Besides of the sensors, the robot is equipped with a two-line LCD display, which displays basic information about the robot such as battery level, the name or the selected gait.

Energy to the entire system is supplied by one 11.1 V Li-Po battery, which can supply up to 60 A. This power is sufficient for all servomotors and electronics. But servomotors require 6 V power supply and electronics require 5 V power supply. Therefore, the robot is equipped with voltage regulators. Each servomotor has its own regulator and there is one more regulator for electronics. Switching voltage regulator were used to decrease power consumption.

V. Hexapod Control Software

The software of the robot can be divided into three parts – a computer Qt user interface program, a Raspberry Pi Qt

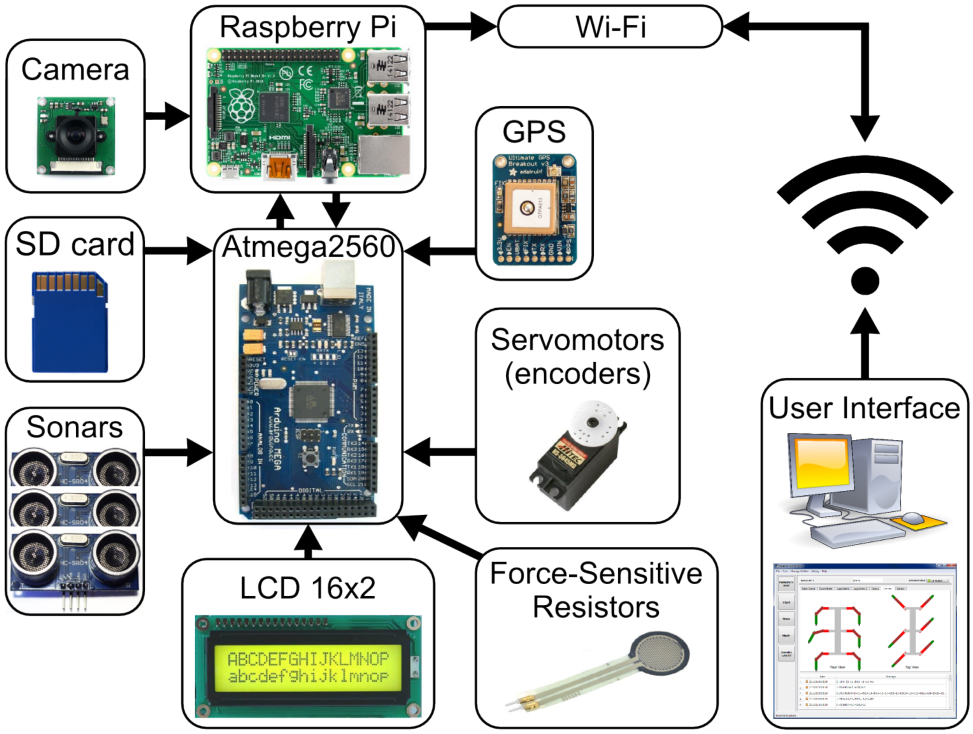

Fig. 5. The electronic system of the robot. In the center is a MCU Atmega2560 integrated on a Arduino Mega 2560. Most of the sensors like sonars, LCD display, memory card, GPS module or force-sensitive resistors are connected to it. There are also 18 servomotors connected to digital pins and driven by MCU's timers. Arduino board is connected to the Raspberry Pi via USB cable. Raspberry Pi is connected to the computer via wi-fi and provides data from the sensors to the computer and commands from the computer to the Arduino. Data from the sensors are visualized in the user interface on the computer.

program and an Arduino C++ program. Data from Arduino are sent to the Raspberry Pi through a serial port and then to a computer over Wi-Fi.

A. Hexapod Control Room

Hexapod Control Room is user interface program (Figure 6) designed in C++ and Qt [14], which is used to control servomotors, gaits and monitor sensors of the robot. It also shows actual position of the robot legs from the top and the back view and provides command and log windows. Server allows up to 10 connections. It is possible to switch among the connected robots. Inactive robots are automatically removed.

There was also developed Qt library for custom and external applications. It allows to create a server, which creates connection to the client and returns the raw data from the client. This is useful when developing custom control application for the hexapod robot.

B. Robot Client Program

This program runs on Raspberry Pi and communicates with Hexapod Control Room and Arduino MCU. User interface is shown in Figure 6. After the program starts, it tries to connect to the server. The IP address is discovered using broadcast messages and client asks for a new identifier. The server answers with a new identifier and asks for the type or refuses connection. Client sends the type and name eventually. After exchange of these messages the new robot is added into the server user interface and the client starts to send data from sensors, which are displayed in the Hexapod Control Room. For initial connection TCP sockets were used to prevent connection failures, however, sensor data are sent through UDP sockets, which have higher performance and allow real-time communication.

C. Arduino Program

MCU on Arduino runs program, which communicates with the Raspberry Pi over USB serial port. Once authorized from the Raspberry Pi, the program starts sending data from sensors over serial port. It also controls all servomotors, manages sonar measurements and receives commands from the computer.

It is also important to keep the main program loop non-blocking. Therefore individual steps are performed according to the calendar structure, which is similar to the next-event algorithm. The steps are planned and stored in the calendar structure according to their start times. When performing a step, first item from the calendar is selected and executed. So the non-blocking execution of the main loop is achieved, which allows other events to be performed.

The robot is equipped by ultrasonic sonars to discover obstacles around the robot. If distance measurement is needed, a short pulse is emitted to the trigger pin, which causes the formation of several ultrasonic waves. Waves are reflected in the environment and are detected by the receiver. Based on the time elapsed from sending waves the distance can be calculated. Because measuring the time using Arduino's pulseIn function [16] generates delay, interrupts were used.

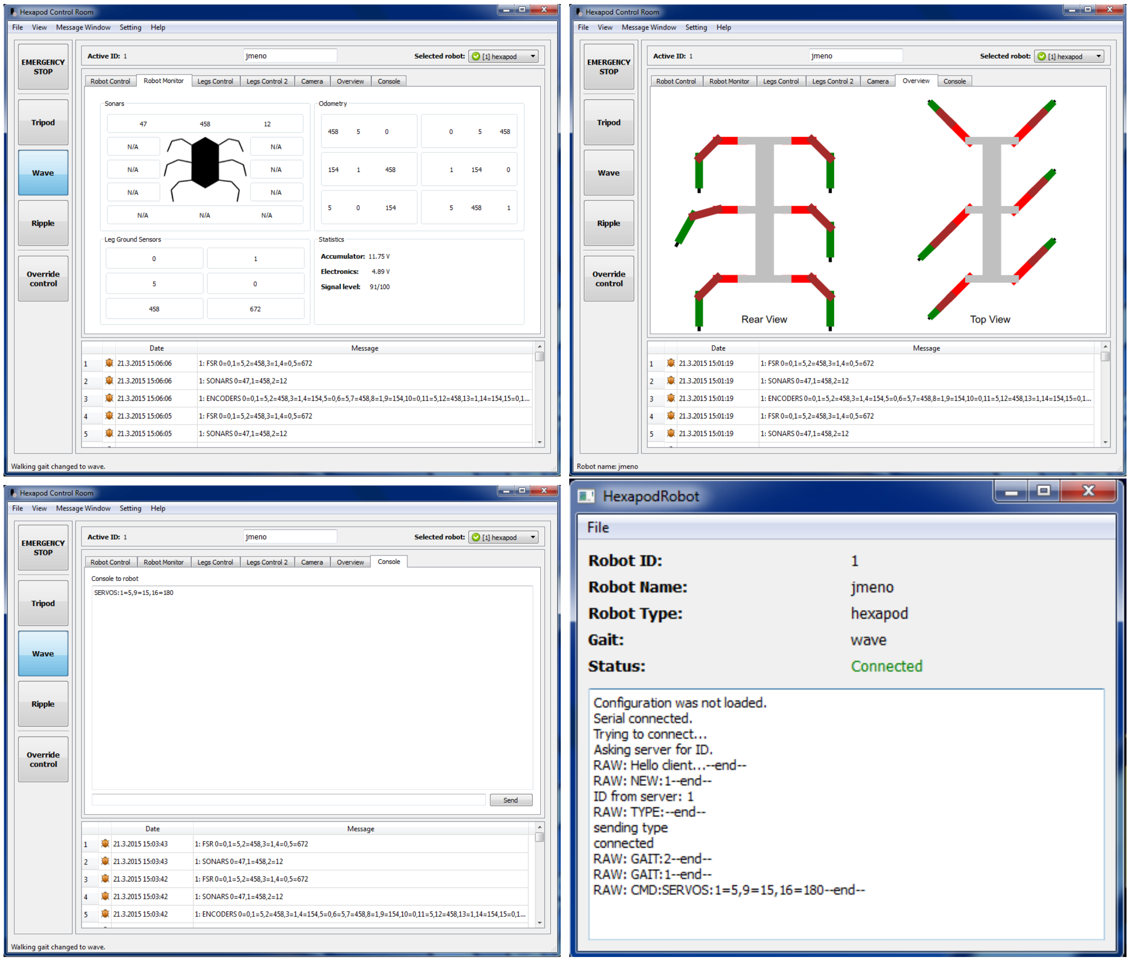

Fig. 6. UI screenshots of Hexapod Control Room. Top left figure shows the Monitor screen with the values of sonars, encoders and force-sensitive resistors. There is also information about battery and robot wi-fi signal strength. Top right image shows the actual position of the legs. Left model is from back view and right model is from top view. Bottom left image presents console, which allows sending command to the robot. Bottom right image shows the client window with the most important information and log window.

The time measurement is started by the arrival of rising signal and ended by falling signal. The distance can be calculated according to the time between these two events.

VI. Hexapod Motion Testing

Hexapod has been tested according to several criteria. One of them was the movement of the robot on different surfaces (carpet and wooden floor) and with different footers (bolt and bolt with paper tape). The robot was also tested in rugged terrain and speed and data transmission were monitored.

A. Walking Test

The robot was tested on various surfaces using various gaits and different materials on the footers and the stability of the robot and any unwanted behavior like slipping of the footers or tilting of the body were monitored. Test parameters are listed in Table I. In each test the robot walked using the given gait on the given surface equipped with bolted footers or bolted footers with tape. The test path did not contain any obstacles.

Tests showed, that carpet is much better surface than wooden floor, because carpet has higher friction than wooden floor. The robot slipped and fell during one test on wooden floor. Also paper tape on the footer's bolts improved stability.

TABLE I. Tests setup

ID	Surface	Footers	Gait
1	carpet	bolt	wave
2	carpet	bolt	tripod
3	carpet	bolt with tape	wave
4	carpet	bolt with tape	tripod
5	wooden floor	bolt	wave
6	wooden floor	bolt	tripod
7	wooden floor	bolt with tape	wave
8	wooden floor	bolt with tape	tripod

The wave gait was more stable compared to the tripod gait. It is because during the tripod gait robot stands only on three legs. The robot was able to walk almost in all tests.

B. Walk Testing in Rugged Terrain

These tests were focused on walking robot in rugged terrain, which was simulated by wooden obstacles. The robot walks through the testing area and the stability of the robot and foot sensors were tested. The robot was capable to walk through the area without any issues. Except for the inability

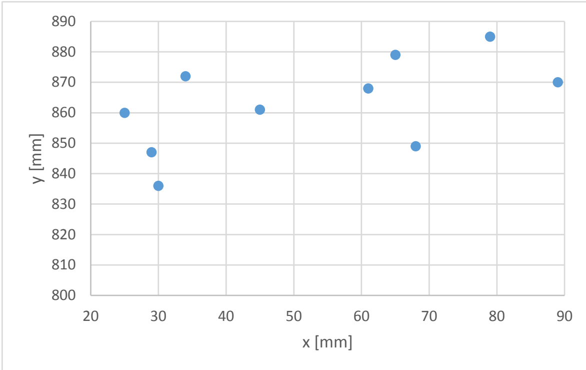

Fig. 7. Results of repeatability of movement tests. Each point represents the final position of the front right leg. The standard deviation of all tests for x is 21,72 and for y is 14,45. It can be observed, that points are distributed in two clusters, which have quite similar shape. This can be result of the decreasing voltage in the battery or because of the freedom in joints. We are preparing improvements to the control system, which should lead to decrease the variance of the final positions.

of the robot to detect obstacles at the leg level, which is also a precondition for climbing stairs.

The sensors were tested by comparing the values on the force-sensitive with thresholds using the test software on the computer which was connected via a USB serial link. If the thresholds were not achieved, it can lead to instability, which is not desirable and the test was unsuccessful.

It is also important, that the tibia is perpendicular to surface plane. Otherwise the servomotors are in constant load, which leads to higher power drain and can even lead to its malfunction. Therefore, the tibia angle was monitored.

C. Repeatability of Movement

The aim of these tests was to determine whether the robot is able to repeat the movement on a certain path. The robot walked ten times through the same path and the final positions of footers were marked. The robot walked using wave gait with active ground sensors. The wave gait was chosen, because it is the most stable gait and therefore the error should be minimal. During each test the robot made thirteen cycles of the gait.

The results of these tests are shown in Figure 7. Each point represents the final position of the leg. The standard deviation of all tests for x is 21,72 and for y is 14,45, which is quite good result compared to the size of the robot and the length of the path, which was about 80 cm.

VII. Conclusion

A Hexapod robot was designed, constructed and tested during this project. Robot can walk using tripod, wave and ripple gaits and it is equipped with ultrasonic sonars, force-sensitive resistors, encoders and LCD display. Ground sensors on legs allow ground detection, so the robot can walk in rugged terrain. An Arduino Mega 2560 board was selected as the main control unit. This board is equipped by Atmega2560 microcontroller and all peripherals are connected to it. Arduino is connected via USB serial port to Raspberry Pi, a miniature computer, which provides connection to the control computer via Wi-Fi.

There was designed user interface program in C++ and Qt, which allows to control and monitor robot. Program visualise legs positions, displays data from sensors and allows sending commands to robot. It also allows the user to create custom gait, which can be either tested on the robot or simulated in the user interface. Connection to control program is automatic.

Performed tests show, that the robot is capable of quite precise movement, even in rugged terrain.

The price of components on the robot (body, servomotors, MCU) is about $ 500. Another $ 80 cost Raspberry Pi, sensors and accessories.

In our future work we want to involve in the research of controlling hexapod robot using evolution techniques like central pattern generators.

Acknowledgment

This work was supported by the European Regional Development Fund in the IT4Innovations Centre of Excellence project CZ.1.05/1.1.00/02.0070 and by the project IGA FITS-14-2486.

References

[1] U. Saranlı, "Dynamic locomotion with a hexapod robot," Ph.D. dissertation, The University of Michigan, 2002.

[2] F. Tedeschi and G. Carbone, "Design issues for hexapod walking robots," *Robotics*, vol. 3, no. 2, pp. 181–206, 2014.

[3] NASA, "All-terrain hex-limbed extra-terrestrial explorer," http://athlete.jpl.nasa.gov/, 2009, [Online; visited 09-03-2015].

[4] B. Dynamics, "Boston dynamics: Dedicated to the science and art of how things move." http://www.bostondynamics.com/, 2015, [Online; visited 10-03-2015].

[5] Lynxmotion, "Lynxmotion robot kits," http://www.lynxmotion.com/, 2015, [Online; visited 10-03-2015].

[6] T. Robotics, "Trossen robotics," http://www.trossenrobotics.com/, 2015, [Online; visited 10-03-2015].

[7] E. Moore and M. Buehler, "Stable stair climbing in a simple hexapod robot," DTIC Document, Tech. Rep., 2001.

[8] X. Ding, A. Rovetta, J. Zhu, and Z. Wang, *Locomotion analysis of hexapod robot*. INTECH Open Access Publisher, 2010.

[9] S. Manoiu-Olaru, M. Nitulescu, and V. Stoian, "Hexapod robot. mathematical support for modeling and control," in *System Theory, Control, and Computing (ICSTCC), 2011 15th International Conference on*, Oct 2011, pp. 1–6.

[10] R. Siegwart, *Introduction to autonomous mobile robots*. Cambridge, Mass: MIT Press, 2004.

[11] R. P. Foundation, "Respberry pi," http://www.raspberrypi.org/, 2014, [Online; visited 09-04-2015].

[12] Arduino, "Arduino - arduinoboardmega2560," 2015.

[13] Atmel, "Atmega2560," 2015.

[14] Digia, "Qt project," http://qt-project.org/, 2015, [Online; visited 21-03-2015].

[15] N. T. Group, "Varspeedservo," https://github.com/netlabtoolkit/VarSpeedServo, 2013, [Online; visited 21-03-2015].

[16] Arduino, "Arduino - pulsein," http://arduino.cc/en/Reference/pulseIn, 2015, [Online; visited 21-03-2015].