The 8th TSME International Conference on Mechanical Engineering
12-15 December 2017
Bangkok, Thailand

BME0006

Kinematic analysis and trajectory of a dog-like robot

Amornphun Phunopas[1,*], and Narisara Suwichien[2]
[1] Department of Production Engineering, King Mongkut's University of Technology North Bangkok, Faculty of Engineering 1518 Pracharat 1 Road Bangsue, Bangkok 10800, Thailand
[2] Department of Industrial Technology, Pibulsongkram Rajabhat University, Faculty of Industrial Technology 156 Mu 5 Plaichumpol Sub-district, Muang district, Phitsanulok 65000, Thailand

[*]Corresponding Author: amornphun.p@eng.kmutnb.ac.th, Tel. +668 8560 2994, Fax. +66 2 587 0029

Abstract A dog-like robot is a platform that is useful as a social partner. It can help with certain simple tasks or even take care of children and elderly people. The first step in development is to design the robotic structure and confirm its locomotion. The robot is a small-sized pet that will be mimicked by a mechanical system. The robot has four legs. There are three joints in each leg. The present work demonstrates the kinematic analysis and trajectory of the four legs. It verifies the results in simulation and then implements them in real robot hardware, using a digital servo motor for each robotic joint.

1. Introduction

The dog has been a beloved pet throughout history. The dog-like robot platform is easy to interact with for everyone as it is familiar to humans. However, a dog-like robot is harder to build and control than a wheeled platform because the former has four legs and many actuators. Advanced research was performed by the Boston Dynamics Laboratory [1]. They have developed SpotMini, which has four legs with fully autonomous motion control. The size is equal to that of a big dog, but it has a very long neck. It can walk and behave like an animal; for example, it plays with people. Moreover, it can step over a ladder and stand up when it falls on the floor. However, the idea of a social robot has been explored for many years. In 1999, the first autonomous robot AIBO was commercialized by Sony [2]. Nevertheless, Sony discontinued the AIBO product in 2006. AIBO had four legs with three degrees of freedom per leg and was a behavior programmable dog-like robot. Eniko et al. (2003) experimented to compare the social behavior of real dogs and AIBO [3]. AIBO could not realistically represent the behavior of a dog, but it showed the future trend that the robot would have indistinguishable animal-like behavior.

The structure of the dog-like robot was designed using only one degree of the freedom of the legged-robot mechanism, which mimics the anatomy and anthropometric proportions of a dog. It used a low number of actuators [4]. The primary analysis of a multi-legged robot is kinematics and trajectory generation. Lopez et al. researched how the hexapod robot performed the movements of mammals, insects or reptiles and showed the simulation using OpenGL [5]. For another kind of animal, the gait planning is inspired by the biologically correct, multi-legged, crablike robot [6].

To control the multi-legged robot is more complicated than controlling the conventional wheeled robot. The multi-legged robot must be stable and balanced under dynamic modeling [7]. Moreover, it is known that the legged robot has many links and consumes more energy than the wheeled robot.

In a study focused on energy consumption analysis, the first approach was a minimization of the norm of feet forces, and the second was a minimization of the norm of joint torques [8]. However, it showed only a mathematical model without real energy consumption measurement. Agheli and Nestinger proposed measuring the force-based stability margin of a multi-legged robot on irregular terrain by attaching the force sensor to the foot [9]. Gor et al. used the bond graph technique that can be generated from a three-dimensional dynamic model of the legged quadruped robot, which has two revolute joints and a prismatic compliant link per one leg. The bond graph can interface with various controller models [10-11]. Besides, with a walking legged robot, there are other legged-robot physical behaviors, such as a dynamic model of the jumping robot [12], and the automated walking and running [13]. The robot needs to have a feedback sensor to control its motion smoothly and provide stability. Zhiguo Shi et al. collected the motion parameters with data from the vision system to solve the stability problem by using a software-based algorithm to search an optimized motion parameter. It calculated new input control parameters for the four-legged robots' motion [14].

The parallel genetic algorithm is used to design the walking gait for the four-legged robot. The idea is that a legged-robot should continuously walk autonomously in unexpected situations. For example, if the robot gets damaged, i.e., a loss of operation of one or two legs, the robot should adopt a new walking gait [15]. The fault-tolerant gait is analyzed on an inclined plane [16]. Moreover, the multi-legged robot, which has high performance in locomotion, is developed to localize its position using SLAM (Simultaneous Localization and Mapping). Xuehe et al. showed that the multi-legged robot could explore unknown indoor environments using the matching feature points in 2D images with high accuracy [17].

From the overview, there are many techniques to develop the multi-legged robot to have physical and social behavior like an animal. This research aims to develop a small-sized dog-like robot platform for further research.

2. Dog-like Robot Structure

The robot mimics a puppy or small-sized dog. The dog-like robot has four legs, and one leg has three degrees of freedom, as shown in **Figure**. 1. The robot is a prototype and used as a testbed. The body frames are made from ABS filament from 3D rapid-prototype printing, but the structure is strong enough to test the robot walking.

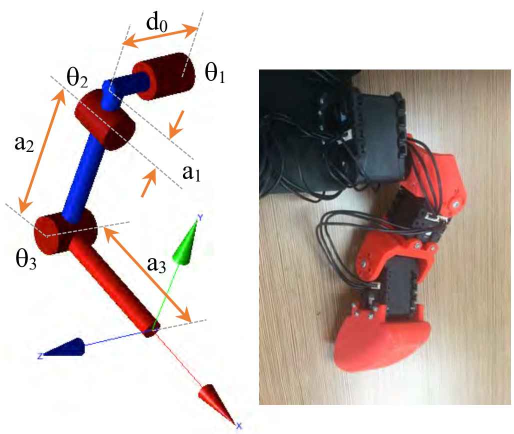

Figure. 1 (Left) The robot configuration has joint and link parameters for kinematic calculation. (Right) A robot's leg structure has three revolution joints using a digital servo motor in real implementation.

The robot has a workspace, and the limit angle of actuators for each joint is 300 degrees, as shown in **Figure**. 2. The mechanical frames limit the joint motion differently for each joint position to perform such behaviors as sit, sleep, and expand its legs in an X-shape.

The dog-like robot has coordinate frames defined by the global coordinate frame for the whole body and local coordinate frame for each leg, as shown in **Figure**. 3. The body is rectangular, with width of 130 mm and length of 150 mm. The center of the robot body is at the (0,0,0) position, and below the robot body is a minus value in the z-axis.

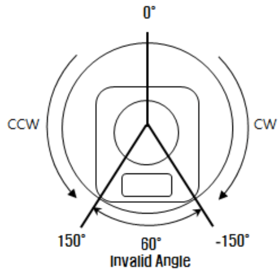

Figure. 2 The motor defined angle has a total of 300 degrees. The origin is at the center 0 degrees; turning clockwise is a minus angle and counterclockwise is a plus angle. The invalid angle is 60 degrees.

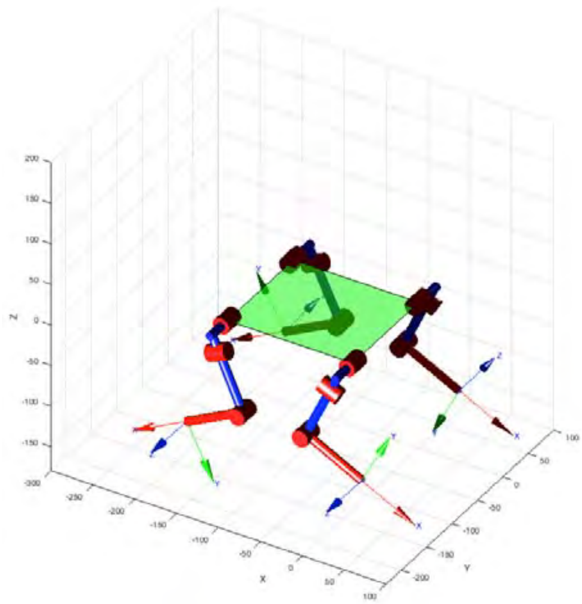

Figure. 3 The robot default configuration is ready to move to other gestures and actions. It can naturally perform dog-like behavior.

From Figure. 3, the robot has a default configuration that has a joint angle as in Table 1, where FR is the Front Right leg, FL is the Front Left leg, RR is the Rear Right leg, and RL is the Rear Left leg. θ_1 is the shoulder joint, θ_2 is the wing joint, and θ_3 is the knee joint.

Table. 1 Total of 12 joints' angle for the default standing configuration.

Joint's angle	Leg (degree)			
	FR	FL	RR	RL
θ_1	-45	45	45	-45
θ_2	1	1	1	1
θ_3	90	-90	-90	90

2.1 Kinematics

The robot has a modular leg that has the same forward kinematic equation for every leg. The robot can move and act by controlling three joints. The kinematic is derived through Equation (1-3). Even though the equation is the same for each leg, the installed positions are different. The left and right side of the robot's leg are mirrors, and the initial angles are defined by a plus and a minus sign.

$$x = -a_3 sin\theta_1 cos\theta_2 cos\theta_3 - a_3 cos\theta_1 sin\theta_3 - a_2 sin\theta_1 cos\theta_2 - a_1 sin\theta_1 \quad (1)$$
$$y = a_3 sin\,\theta_2 cos\theta_3 + a_2 sin\theta_2 + d_0 \quad (2)$$
$$z = -a_3 cos\theta_1 cos\theta_2 cos\theta_3 + a_3 sin\,\theta_1 sin\,\theta_3 - a_2 cos\theta_1 cos\theta_2 - a_1 cos\theta_1 \quad (3)$$

Where, $a_1 = 20$mm, $a_2 = 70$mm, $a_3 = 75$mm and $d_0 = 30$mm.

One leg has been defined in the local coordinate frame, and the origin is at the shoulder or the first joint. From the kinematic equation, the pace trajectory path planning can be created in various shapes, such as a semi-ellipse in Figure 4-5.

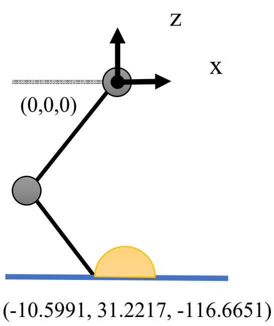

Figure. 4 The end-effector of the robot leg is the foot tip of the robot. It was defined by XYZ coordinates to control the digital servo motor of the shoulder joint and knee joint.

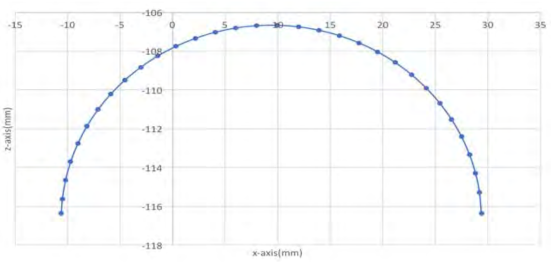

Figure. 5 The pace trajectory path is designed in a semi-ellipse to facilitate stepping forward of the foot tip on the ground. Also, the foot moves forward by a pace height of 10 mm and a pace distance of 40 mm.

The pace trajectory can be inverted to control the position of the robots' leg. Figure 6 shows the front right leg control angle position after about six seconds.

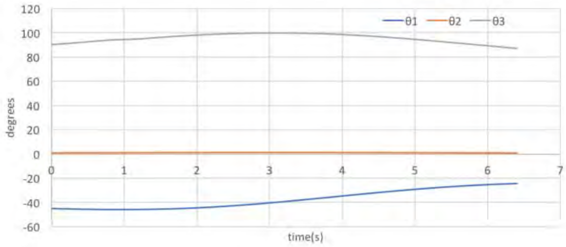

Figure. 6 Front Right Leg moved on a semi-ellipse pace path and the θ_1, θ_2, and θ_3 have changed the rotation related to the semi-ellipse pace trajectory.

3. Four-legged walking gait

The gait can be generated by the kinematic equations. The physical features of the dog-like robot are similar to the real dog, which simplifies generating a gait pattern (Figure 7). It uses a shoulder joint and knee joints, neglecting the wing joint when the robot walks forward. The wing joint is used when turning and spreading its legs.

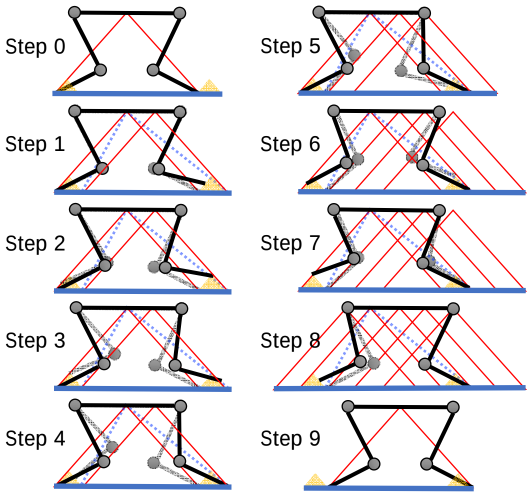

Figure. 7 Side view of gait pattern in nine steps while moving forward on a smooth floor

The robot moves sequentially forward in the X-axis and lifts its legs in the Z-axis of the global frame. It orderly starts moving in step 1 with FR. While FR moves, RL starts moving in step 2. Through step 5, FR and RL have already moved to the new pace. Next, FL starts moving, and the last leg RR start moving. Finally, all legs move to the next position by one pace in step 9.

4. Design and System

4.1 Design

The robot uses 12 digital servo motors as the joints and skeleton of the four legs, similar to a puppy. All motors control the position simultaneously according to the gait. The robot's frame is made from plastic, and the controller is put on its body. The rear legs are shorter than the front legs because the feet were designed in different sizes and shapes. Both front legs can take longer paces than the rear legs. The whole-body robot structure is shown in Figure 8.

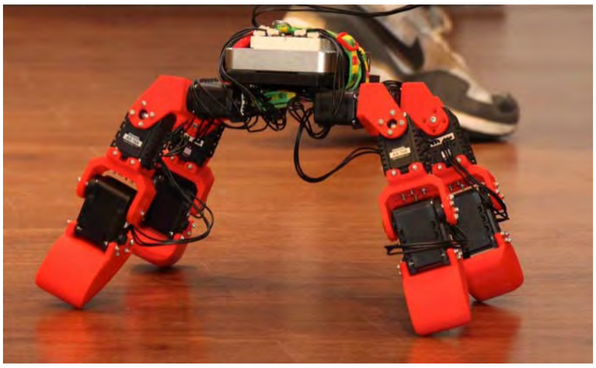

Figure 8. The dog robot standing and expanding its four legs to the side of its body

In fact, each motor has a different static payload, but the robot just uses the same motor model for every joint that has the same maximum torque. The robot has a total weight (W), and the appropriate useable torque should not be over the maximum torque (τ_{max}). The robot has four legs, and its weight can distribute to each leg equally. Figure 9 shows a free-body diagrams of a one-leg torque calculation.

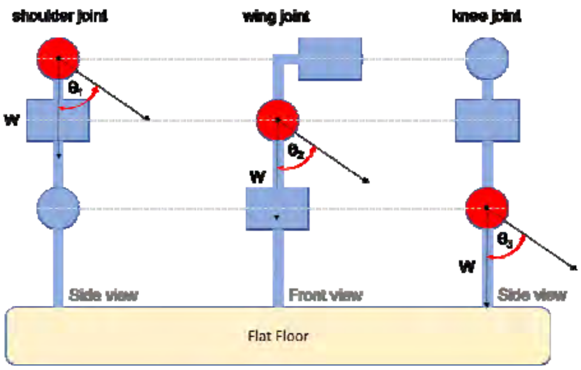

Figure. 9 One-leg robot in the different view. For the torque calculation, the robot stands on a flat floor. The leg stretches out at its most extended length.

The robot is small and lightweight. It has a high-performance motor that is used in the small-size humanoid robot. The calculation helps to know the robot weight limitation in Equations 4 to 6.

$$\tau(\theta_1) = (0.25w)\cdot \sin(\theta_1)\cdot(a_1 + a_2 + a_3) \quad (4)$$
$$\tau(\theta_2) = (0.25w)\cdot \sin(\theta_2)\cdot(a_2 + a_3) \quad (5)$$
$$\tau(\theta_3) = (0.25w)\cdot \sin(\theta_3)\cdot a_3 \quad (6)$$

From the equations, the maximum load is at the shoulder joint. The condition is $\tau(\theta_1)\cap\tau(\theta_2)\cap\tau(\theta_3)<\tau_{max}$. The robot's maximum weight should not be over 4.36 kgf. All the robot's legs do not share the weight all the time while the robot is walking. The motor will stop working when it is overloaded by the dynamic external force. Therefore, the recommended robot weight should not be over 2 kgf by more than 50% of the safety factor.

4.2 System

The robot is operated by an OpenCM ARM Cortex-M3 microcontroller that is powered by a 32-bit CPU [18]. The microcontroller needs an extension Dynamixel motor driver board through a 485 serial bus protocol [19] for the 12 motors paired using a three-wire connector as in Figure 10. The Dynamixel refers to each motor using an ID number. The motor model is AX-18A [20], and it has maximum torque of 18kgf/cm, stall torque of 1.8 Nm, and a speed of 97 rpm, using operation voltage of 9-12 V.

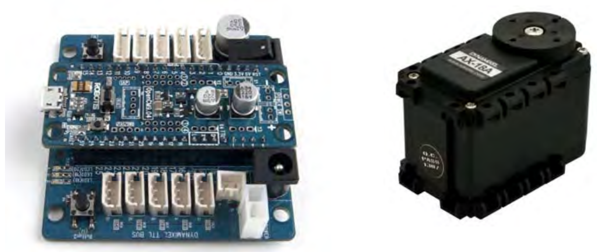

Figure. 10 (Left) The ARM Cortex-M3 is placed on the OpenCM 485 extension board. (Right) The digital servo model is Dynamixel AX-18A.

4.3 Software

The robot's legs move together concurrently when the robot walks. The robot control hardware and software must support the walking behavior. OpenCM IDE [21] is similar to Arduino programming, and the MCU has high efficiency. It supports a multitasking programming that is necessary for the four-legged robot moving coincidently using FreeRTOS to perform tasks as in Figure 11. The robot moves FR, RL, FL, and finally RR in parallel time steps.

The robot walks by using the multitasking program in Table 2. The program begins with a connection task to the 485 bus protocol for communicating with the Dynamixel motors in line 3. The setup function has created 12 tasks to control the motors in lines 4-20. There is nothing run in the loop function, but there are 12 synchronous multitasks defined to control 12 motors individually in walking step series angles in lines 24-35. However, each motor has to rotate in the trajectory walking pattern.

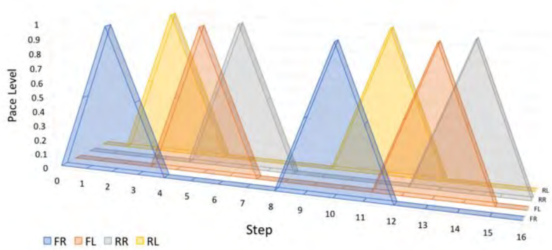

Figure. 11 The pace level is normalized. Four-legged parallel logic control to lift the robot feet up and down in walking steps to move the robot forward.

Table. 2 Pseudocode by Arduino Coding

```
void setup()
{
      Set connection to 485 Bus
      // Front Right Leg
      Create task_id3 for move θ1 // motor id3
      Create task_id2 for move θ2 // motor id2
      Create task_id1 for move θ3 // motor id1
      // Front Left Leg
      Create task_id4 for move θ1 // motor id4
      Create task_id5 for move θ2 // motor id5
      Create task_id6 for move θ3 // motor id6
      // Rear Left Leg
      Create task_id9 for move θ1 // motor id9
      Create task_id8 for move θ2 // motor id8
      Create task_id7 for move θ3 // motor id7
      // Rear Right Leg
      Create task_id10 for move θ1 // motor id10
      Create task_id11 for move θ2 // motor id11
      Create task_id12 for move θ3 // motor id12
}
// Multitasking Trajectory Path Profile
void loop() {}
void task_id1(){ Set knee angles of FR }
void task_id2(){ Set wing angles of FR }
void task_id3(){ Set shoulder angles of FR }
void task_id4(){ Set shoulder angles of FL }
void task_id5(){ Set wing angles of FL }
void task_id6(){ Set knee angles of FL }
void task_id7(){ Set knee angles of RL }
void task_id8(){ Set wing angles of RL }
void task_id9(){ Set shoulder angles of RL }
void task_id10(){ Set shoulder angles of RR }
void task_id11(){Set wing angles of RR }
void task_id12(){Set knee angles of RR }
```

5. Experiment

The robot is tested by forward walking to verify the kinematic equation with a pure walking gait. It has a walking gait trajectory without an algorithm to balance the robot when it takes a walk. The robot uses the digital servo motors that already have a positioned control firmware. The environment used for testing is a flat floor.

5.1 Result

The result of the robot walking is seen as image sequences. The graphs are not precisely like the planned walking gait because it is modified to keep balancing the robot to walk straight forward in Figure 12. The left and right side legs have a different motion because the control values of the motor have been compensated.

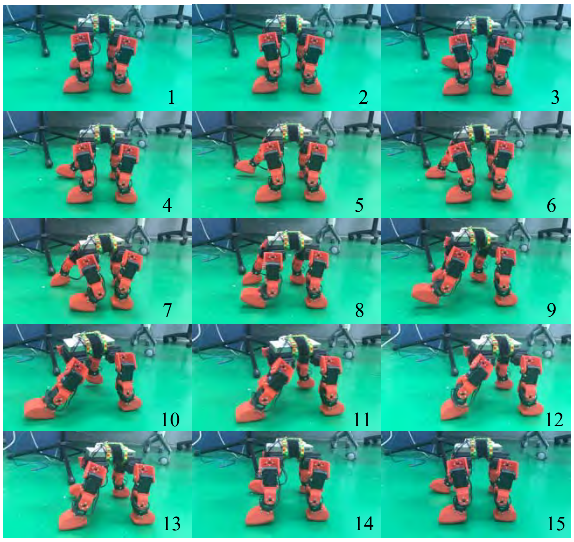

Figure. 12 The image sequence from 1 to 15 images in five seconds on the walking one pace of each leg

The robot stood walked without expanding its legs to the side. Thus, if the wing joint angle was fixed by one degree, then it is not shown in Figure. 13.

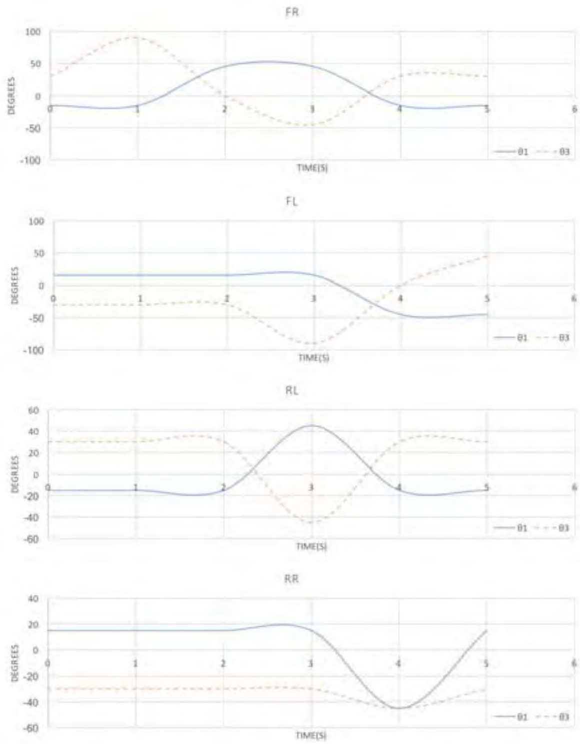

Figure. 13 The series of θ_1 and θ_3 angle to control the robot, which concurrently walked forward, using four legs as FR, FL, RL, and RR, in five seconds per round.

6. Discussion

The robot did not move smoothly, and it did not walk in a well-balanced fashion because it did not have additional sensors. For example, the IMU sensor is necessary for detecting the center of gravity and the body balance, when the robot walks. The robot needs a sensor at its feet to detect whether it has already contacted the ground. Moreover, the feet are made of a non-adhesive plastic filament, so the robot slips while walking. Therefore, the design of the walking gait pattern does not work well.

7. Conclusion

The dog-like robot can walk forward by relying on the walking gait trajectory from the validated kinematic equation. In many patterns, the walking gait and trajectory pace can be designed independently for proper stability and balance. The multi-task program can control the four-legged walking gait well. However, it is better to use the high performance that is embedded in the computer or mini-pc, to develop its intelligent perception of, and response to, the environment.

Acknowledgement

This work is supported by the Green University Innovation Project 2015 and a New Researcher Grant 2016 from the Faculty of Engineering, King Mongkut University of Technology, North Bangkok.

References

[1] The Boston dynamics laboratory, USA, URL: https://www.bostondynamics.com/spot-mini, accessed on 18/09/2017
[2] The SONY company, Japan, http://www.sony-aibo.com/, accessed on 18/09/2017

[3] Kubinyi, E., Miklósi, Á., Kaplan, F., Gácsi, M., Topál, J., & Csányi, V. (2004). Social behaviour of dogs encountering AIBO, an animal-like robot in a neutral and in a feeding situation. Behavioural processes, 65(3), 231-239.
[4] Na, B., & Kong, K. (2016). Design of a One Degree-of-Freedom Quadruped Robot Based on a Mechanical Link System: Cheetaroid-II. IFAC-PapersOnLine, 49(21), 409-415.
[5] GarcaLpez, M. C., Gorrostieta-Hurtado, E., Vargas-Soto, E., Ramos-ArreguÃ-n, J. M., Sotomayor-Olmedo, A., & Morales, J. M. (2012). Kinematic analysis for trajectory generation in one leg of a hexapod robot, Procedia Technology. Volume, 3, 342-350.
[6] Chen, X., Wang, L. Q., Ye, X. F., Wang, G., & Wang, H. L. (2013). Prototype development and gait planning of biologically inspired multi-legged crablike robot. Mechatronics, 23(4), 429-444.
[7] Shah, S. V., Saha, S. K., & Dutt, J. K. (2012). Modular framework for dynamic modeling and analyses of legged robots. Mechanism and Machine Theory, 49, 234-255.
[8] Roy, Shibendu Shekhar, and Dilip Kumar Pratihar. "Dynamic modeling, stability and energy consumption analysis of a realistic six-legged walking robot." Robotics and Computer-Integrated Manufacturing 29.2 (2013): 400-416.
[9] Agheli, M., & Nestinger, S. S. (2016). Force-based stability margin for multi-legged robots. Robotics and Autonomous Systems, 83, 138-149.
[10] Gor, M. M., Pathak, P. M., Samantaray, A. K., Yang, J. M., & Kwak, S. W. (2015). Control oriented model-based simulation and experimental studies on a compliant legged quadruped robot. Robotics and Autonomous Systems, 72, 217-234.
[11] Ragusila, V., & Emami, M. R. (2016). Mechatronics by analogy and application to legged locomotion. Mechatronics, 35, 173-191.
[12] Kumar, G., & Pathak, P. M. (2013). Dynamic modelling & simulation of a four legged jumping robot with compliant legs. Robotics and Autonomous Systems, 61(3), 221-228.
[13] Shahbazi, M., Lopes, G. A. D., & Babuska, R. (2014). Automated transitions between walking and running in legged robots. IFAC Proceedings Volumes, 47(3), 2171-2176.
[14] Shi, Z., Zhang, Q., Tu, J., Wang, Z., & Zhang, X. (2012). Vision Stability of Four-legged Robot Based on the Feedback of Motion Parameters. Procedia Engineering, 29, 3250-3255.
[15] Park, H., & Kim, K. J. (2013). The automated fault-recovery for four-legged robots using parallel genetic algorithm. Procedia Computer Science, 24, 158-166.
[16] Sarkar, D., Dubey, S. K., Mahapatra, A., & Roy, S. S. (2014). Modeling and analysis of fault tolerant gait of a multi-legged robot moving on an inclined plane. Procedia Technology, 14, 93-99.
[17] Xuehe, Z., Ge, L., Gangfeng, L., Jie, Z., & ZhenXiu, H. (2016). GPU based real-time SLAM of six-legged robot. Microprocessors and Microsystems, 47, 104-111.
[18] OpenCM 9.04 , The ROBOTIS company, Korea, http://support.robotis.com/en/product/controller/opencm9.04.htm, accessed on 18/09/2017
[19] OpenCM 485 expansion board, ROBOTIS company, Korea, http://www.robotis.us/opencm-485-expansion-boar/, accessed on 18/09/2017
[20] The digital servo motor model AX-18A, ROBOTIS company, Korea, URL: http://www.robotis.us/dynamixel-ax-18a/, accessed on 18/09/2017
[21] OpenCM integrated development environment (IDE), ROBOTIS company, Korea, URL: http://www.robotis.us/opencmide/, accessed on 18/09/2017