Navigation Control System of Walking Hexapod Robot

Javier Ollervides, Jorge Orrante-Sakanassi, Víctor Santibáñez y Alejandro Dzul
Instituto Tecnológico de la Laguna, Blvd. Revolución y Cuauhtémoc, Torreón, Coah., 27000, México
e-mail: jollervi@itlalaguna.edu.mx, jaos@ieee.org, vsantiba@itlalaguna.edu.mx, dzul@faraday.itlalaguna.edu.mx
Phone: (52)-871-7051324

***Abstract*—This paper describes the implementation of an electronic control system applied to a multivariable system called hexapod robot walker. The application focuses on the navigation system for the movement of the robot, planning the movements of the robot legs. The measurement of physical variables of the robot is performed using electronic force sensors mounted on the end of the legs. Just as an inertial sensor system for feedback orientation and direction in the navigation system of the hexapod robot. The operation of this robot was implemented with DC servo closed-loop controlled by electric current, angular position and angular velocity feedback.**

Keywords: Mobile robots, hexapod robot, walking robots, legged robots, navigation robots.

I. INTRODUCTION

The main problem of an autonomous mobile robot is to perform the control of locomotion in rough terrain. A walking robot can be a compelling alternative in solving this problem. So to accomplish this task, we have implemented a six-legged walking robot called in modern literature as walking hexapod robot. The main reason for the choice of this type of robot, is the fact that kinematic model of a hexapod robot introduces the concept of static stability during locomotion of its legs when it made navigation with their legs in walking.

Control of a hexapod walking robot is a complex problem because the robot navigation is coordinated simultaneous movement of six legs, each with three degrees of freedom (DOF), with a total of eighteen DOF hexapod robot. Because the navigation of the robot is performed in an unstructured area, it is necessary to use an electronic system with feedback sensors for physical variables of the system to perform the hexapod robot interaction with the operating environment . However, the robot must be able to perform their tasks under high uncertainty in the electrical signals that can be measured by the sensors. It must also be considered that the main problem is to coordinate the angular motion of the robotic system eighteen joints during locomotion for the navigation of the robot, including the sequence of steps. This problem can be solved by implementing a dedicated electronic system with distributed control architecture [1].

II. MECHANICAL STRUCTURE OF THE ROBOT

Figure 1 shows the kinematic configuration of one leg of the robot. The kinematic model used is expressed in the coordinate system of the conventional notation Denavit-Hartenberg for joints of each leg of the robot. The six legs of the robot are mechanically identical. The main design parameter considered is the maximum electrical torque and the dimensions of the robot servo. Many of the parts of the structure are implemented with high density aluminum. The shape of the robot was specially chosen to keep the center of mass of the robot in its geometric center, and thus to maintain the balance of the structure during the dynamics of locomotion of the robot legs when running the tripod type walking sequence. The potential energy accumulated due to the physical form of the robot allows you to carry on board a large amount of weight that includes the servo actuators of the legs as well as the electronic system of onboard sensors and the electronic system with microcontroller that implements the algorithm of navigation of the robot. Figure 2 shows a full view of the robot. Each robot leg is equipped with a force sensor mounted on his foot. This force sensor is a variable resistor, *i.e.* the electrical resistance provided by the sensor varies nonlinearly as a function of the force applied in the sensor membrane, allowing it to detect robot footholds a surface, applied in each leg. Other hexapod robots with similar characteristics are mentioned in references [1], [2] and [3].

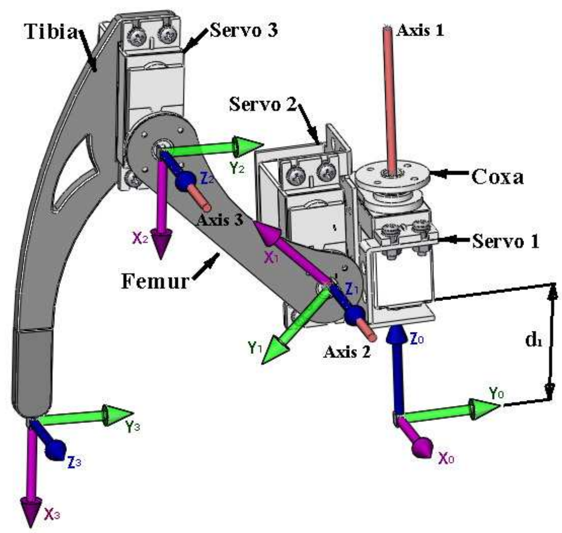

Fig. 1. Kinematic configuration of a robot leg.

978-0-7695-4878-4/12 $26.00 © 2012 IEEE
DOI 10.1109/CERMA.2012.18

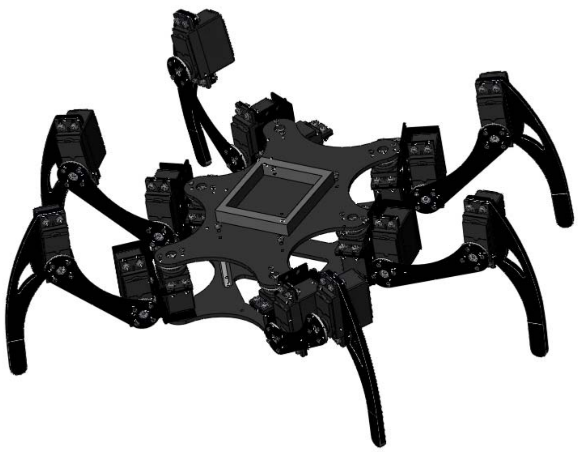

Fig. 2. Full view of hexapod robot.

III. Mathematical model of robot leg

1) Direct kinematic model: The performance of a legged robot depends largely on the design to choose, so it is important to select a mechanical design for the robot to maximize leg movement and impose fewer restrictions on the locomotion of walking. For the implementation of each of the legs of the hexapod robot has been used a kinematic chain of three revolutions or joints (RRR). The joint coordinate frame assignment algorithm uses the geometric model of Denavit-Hartenberg [4], [5]. Direct geometric model for the mechanism of each leg has been formulated through a mobile frame $O_i(x_i, y_i, z_i)$ at each joint, with $i = 1...3$ and a fixed frame $O_0(X_0, Y_0, Z_0)$. The coordinates of the robot frame are assigned as shown in Figure 1. The different links of the legs of the robot have been named as coxa, femur and tibia. The coordinate frames of the leg of the robot starts with a link to the point where the leg is anchored or mounted on the back of the robot, the link 1 is the coxa, the femur is the link two and link 3 is the tibia. The general form of the transformation matrix of link i to link i-1 using the Denavit-Hartenberg parameters is shown in (1).

$$\mathrm{T}_{\mathrm{i}}^{\mathrm{i-1}} = \begin{bmatrix} \cos\theta_i & -\sin\theta_i\cos\alpha_i & \sin\theta_i\sin\alpha_i & L_i\cos\theta_i \\ \sin\theta_i & \cos\theta_i\cos\alpha_i & -\cos\theta_i\sin\alpha_i & L_i\sin\theta_i \\ 0 & \sin\alpha_i & \cos\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{1}$$

The transformation matrix is calculated from the following series of transformations:

- 1) Translate d_i along z_{i-1} axis,
- 2) Rotate θ_i about z_{i-1} axis,
- 3) Translate L_i along x_{i-1} axis,
- 4) Rotate α_i about x_{i-1} axis.

The resulting transformation is obtained as a product of the three transformation matrix of the system:

$$\mathrm{T}_{\mathrm{base}}^{\mathrm{coxa}} = \mathrm{T}_{\mathrm{femur}}^{\mathrm{coxa}}\mathrm{T}_{\mathrm{tibia}}^{\mathrm{femur}}\mathrm{T}_{\mathrm{base}}^{\mathrm{tibia}} \tag{2}$$

where each of these tansformation matrix is defined as:

$$\mathrm{T}_{\mathrm{femur}}^{\mathrm{coxa}} = \begin{bmatrix} \cos\theta_1 & 0 & \sin\theta_1 & L_1\cos\theta_1 \\ \sin\theta_1 & 0 & -\cos\theta_1 & L_1\sin\theta_1 \\ 0 & 1 & 0 & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3}$$

$$\mathrm{T}_{\mathrm{tibia}}^{\mathrm{femur}} = \begin{bmatrix} \cos\theta_2 & -\sin\theta_2 & 0 & L_2\cos\theta_2 \\ \sin\theta_2 & \cos\theta_2 & 0 & L_2\sin\theta_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{4}$$

$$\mathrm{T}_{\mathrm{base}}^{\mathrm{tibia}} = \begin{bmatrix} \cos\theta_3 & -\sin\theta_3 & 0 & L_3\cos\theta_3 \\ \sin\theta_3 & \cos\theta_3 & 0 & L_3\sin\theta_3 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5}$$

Using (2) and considering the (3-5) the coordinates of the final support of a leg of the robot is defined as:

$$\begin{aligned} x &= \cos\theta_1(L_1 + L_3\cos(\theta_2+\theta_3) + L_2\cos\theta_2) \\ y &= \sin\theta_1(L_1 + L_3\cos(\theta_2+\theta_3) + L_2\cos\theta_2) \\ z &= d_1 + L_3\sin(\theta_2+\theta_3) + L_2\cos\theta_2) \end{aligned} \tag{6}$$

where d_1 is the distance from the surface to the joint of the coxa and the parameter L_i represents the length of the links in the leg.

2) Inverse kinematic model: The geometric model described above provides the connection between the joint variables and the position and orientation of the frame located at the far end of the leg. The problem of inverse kinematic model is to determine the joint variables from one position and orientation of the reference frame from the end of the leg. Obtain the solution to this problem is important to specify the paths of motion of the joint variables of the leg, which are obtained from the transformation of the trajectory of movement about the frame from the end of the leg, correspondingly with the desired motion of the reference frame from the end of the leg. Then the objective is to obtain the three joint variables θ_1, θ_2 and θ_3 corresponding to the desired position of the frame from the end of the leg. In this case, reference frame orientation analysis from the end of the leg was not performed, because we are only interested in their position. This procedure can be verified in the references [4], [5].

By applying the direct kinematic equation 6 and to considering the following limitations: all joints allow rotation only on one axis, the links of the femur and tibia always have a rotation on parallel axes, and the physical limitations that we can determine the angle of the joint. The angle of the articulation of the coxa may be obtained by the inverse tangent function as shown in Figure 3.A.

$$\theta_1 = \arctan\left(\frac{x_1}{y_1}\right) \tag{7}$$

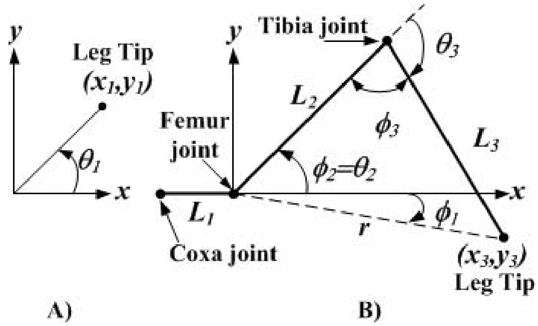

Fig. 3. Inverse kinematic model scheme.

In order to determine the other two joint variables we use a geometric approach. Also to simplify the analysis, the coordinates of the extreme end of the leg will be transformed into the coordinate frame of the coxa using the transformation matrix shown below:

$$\mathrm{T}_{\text{coxa}}^{\text{femur}} = \begin{bmatrix} (R_{\text{femur}}^{\text{coxa}})^T & -(R_{\text{femur}}^{\text{coxa}})^T * d_{\text{femur}}^{\text{coxa}} \\ 0 & 1 \end{bmatrix} \tag{8}$$

The varibale ϕ_2 represents the angle related to the position of the femur servo drive, then from the triangle shown in Figure 3.B can be derived directly that:

$$\theta_2 = \phi_2 \tag{9}$$

The variable ϕ_1 represents the angle between the x axis and the resulting vector r, and can be calculated using the arctangent function:

$$\phi_1 = \arctan\left(\frac{y_3}{x_3}\right) \tag{10}$$

where x_3 and y_3 are the coordinates of the end of the leg with respect to the reference frame of the coxa. If we consider the angle ϕ_t as the sum of the angles $\phi_1 + \phi_2$ and apply the law of cosines to the triangle formed by the sides L_2, L_3 and r (see the Figure 3.B) yields the equation:

$$\phi_t = \arccos\left(\frac{L_2^2 + r^2 - L_3^2}{2L_2 r}\right) \tag{11}$$

where $r = \sqrt{(x_3^2 + y_3^2)}$. Then the variable in the articulation of the femur is expressed as:

$$\theta_2 = \arccos\left(\frac{L_2^2 + r^2 - L_3^2}{2L_2 r}\right) + \arctan\left(\frac{y_3}{x_3}\right). \tag{12}$$

Finally applying the law of cosines to the triangle shown in Figure 3.B, the calculated value for the angle ϕ_3 is

$$\phi_3 = \arccos\left(\frac{L_2^2 + L_3^2 - r^2}{2L_2 L_3}\right) \tag{13}$$

observing Figure 3.B, it is possible to deduce the joint variable θ_3 as:

$$\theta_3 = \pi - \phi_3 \tag{14}$$

IV. Hexapod robot control arquitecture

The control system architecture is shown in Figures 4 and 5. The distributed control system is divided into 3 layers:

- Layer 1 - Core embedded microcontroller connected via USB to a PC (remote command station).
- Layer 2 - Master embedded system (microcontroller **ARM CORTEX M3** with **USART** and **I2C** com. ports).
- Layer 3 - Slave embedded servo (microcontroller with **I2C**).

The highest layer (layer 1) is located in the central command station and based on an embedded board with microcontroller. This layer enables remote communication with the robot. The microcontroller is the core of the central command station; it is dedicated to receiving signals representing physical variables of the robot, such as: the positions and angular velocities, currents and torques of the servo. The communication between the electronic system on board the robot and the central station microcontroller is implemented through a radio frequency (RF) link. The communication channel is located in the 2.4 GHz band and it is implemented with a transceiver module that handles the communication protocol with ZigBee standards.

The next two layers are part of the platform onboard of the mobile robot. The lower layer 2 is anchored on the platform of the robot. This layer is responsible for the planning of trajectories to calculate the desired setpoint paths of the servo motion, to generate the spatial displacement for robot navigation, so as to process the signals obtained from the electronic onboard sensors of the robot . For this purpose we used a embedded microcontroller board with a 32-bit data bus, this system has a integrated transceiver module with ZigBee standards. This module can be used bi-directionally, to receive and transmit data frames for the execution command in the allocation of specific tasks of the robot. By sending data packet frames in a fixed rate, it is permitted that the physical variables of the system may be plotted in real time on the display of the central station.

Lower layer (layer 3) is formed by the integrated electronic drive system, which implements a motion *PID-PI* controller applied to each CD servo. There are three servodrivers per leg. The master microcontroller have a single high speed communication port named **USART** (Universal Asynchronous Receiver Transmitter), which is dedicated specifically to perform the serial communication link with the six legs using the same data bus (see Figures 4 and 5). The real-time measurement of the physical state variables (position and angular velocity as well as the electric current of the servomotor) is performed by the microcontroller which is integrated in the servo module. Data packets communication of of physical variables is implemented using the bidirectional port with serial protocol **I2C** (Inter Integrated Circuit). The data in this communication link is transferred from the memory of the

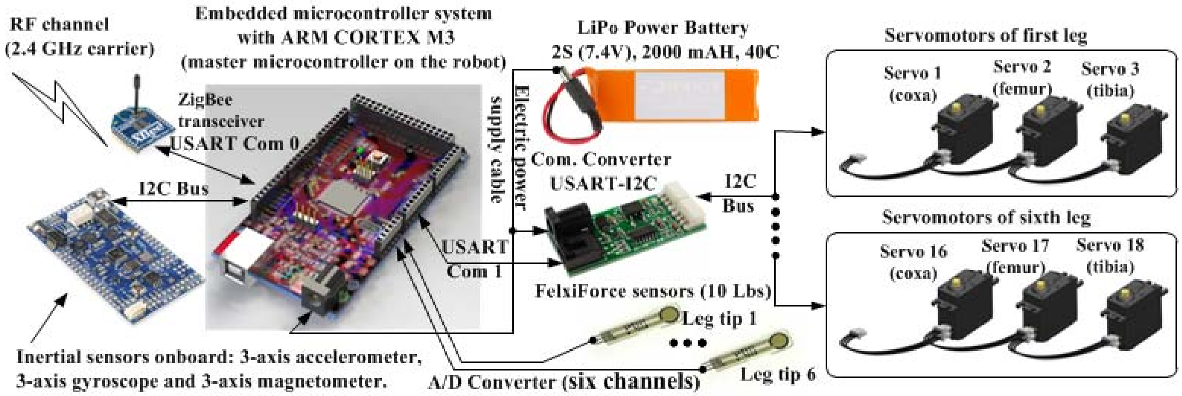

Fig. 4. Architecture of the electronic control system of the hexapod robot.

microcontroller which is integrated in the servodriver module to the master microcontroller on board of mobile robot. Once the data packets are transmitted by RF from the memory of master microcontroller to the memory of the microcontroller located in the central command station, it can be graphed and processed. In this control process, servodriver microcontroller receives the setpoint position and angular velocity, and as an immediate response returns the data measured by the position and angular sensors, as well as current of servo through **I2C** bus. This occurs in real-time algorithm of the electronic control system.

The control system has two channels of communication. The first one consists of two RF modules that implement the ZigBee communication protocol connected to the serial port **USART** implemented by the microcontroller of the central command station, and the other connected to the serial port **USART** implemented by the master microcontroller onboard platform of mobile robot (see Figures 4 and 5). The modules mentioned above are used to send command information from the central command station to the mobile robot, and send the information of the current state of the robot from the onboard microcontroller of mobile robot to the microcontroller of central command station. The second channel is implemented using the **I2C** communication bus, which connects each embedded microcontroller of servomotor module with the master microcontroller board.

V. FEEDBACK CONTROLLER OF ROBOT LEGS

The servo used for this application have a serial communication system for receiving the position and velocity setpoints in digital format, also have the ability to send data packets that represent the state variables of the actuator such as the position and angular velocity, as well as current and torque electric servomotor. Each servo drive has an integrated microcontroller running a feedback control algorithm *PID-PI*. The equation of this controller is shown below:

$$u_c(t) = K_{pc} i_e(t) + K_{ic} \int i_e(t) \tag{15}$$

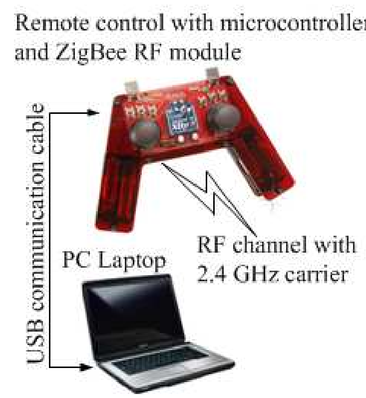

Fig. 5. Wireless remote central command station.

where $i_e(t) = i_d(t) - i_c(t)$ is the vector of the electric current errors of the servos of a leg, while $i_c(t)$ is the vector of electric currents sensed in the actuators of a leg, and $i_d(t)$ is the vector of the desired current of *PI* controller. The vector $i_d(t)$ represents the *PID* control law for the joint position of a leg, expressing as:

$$i_d(t) = K_p \theta_e(t) + K_i \int \theta_e(t) + K_v \dot{\theta}_e(t) \tag{16}$$

where $\theta_e(t) = \theta_d(t) - \theta(t)$ is the vector of the joint position error, $\theta_d(t)$ is the vector of the desired joint position and $\theta(t)$ is the vector of the current joint position. To calculate the path of transition from one leg of the robot in joint coordinates was necessary to use the inverse kinematic model. The desired setpoint is assigned to move sinusoidally with the purpose of planning a desired three-dimensional trajectory of parabolic type. The equations that produce a parabolic motion planning shown below.

$$x_d = \frac{x_f - x_i}{2}\left(1 - \cos\left(\frac{\pi t}{t_f - t_i}\right)\right), \quad z_d = -0.2x_d^2 + 2x_d \tag{17}$$

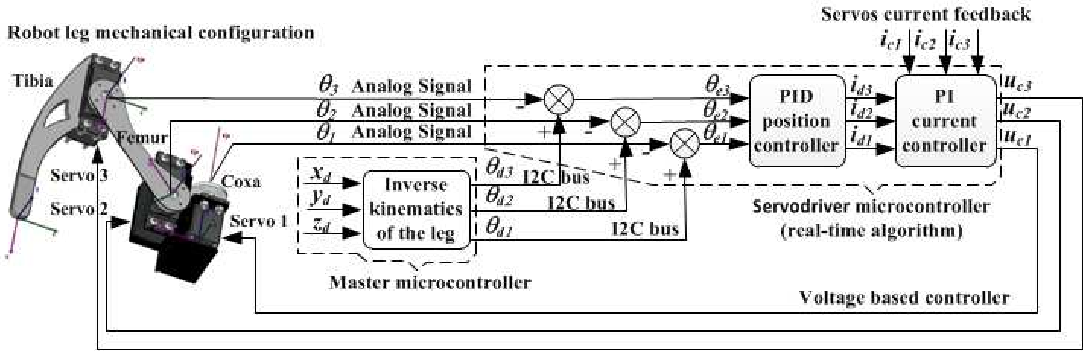

Fig. 6. Feedback control system of a leg servos of the robot.

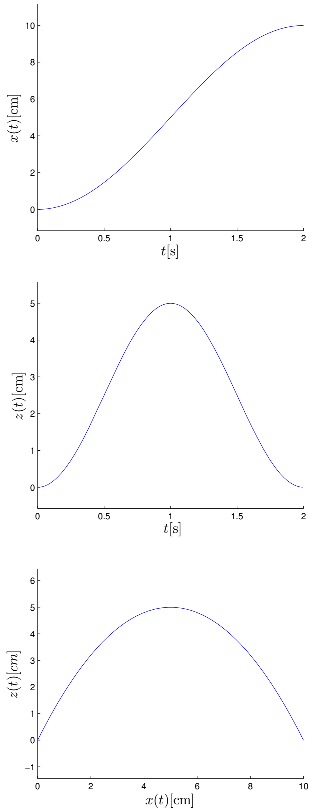

Fig. 7. Graphs of desired motion of the final extreme of leg.

where parameters x_i and x_f represent the initial and final distance on the x axis , while the constants t_i and t_f represent the time that goes the distance. Seen in Figure 7 the graphs for the desired motion trajectory.

The locomotion algorithm implemented to perform the navigation of the robot is programmed in $C++$ language for an **ARM CORTEX M3** 32 bit microcontroller of **ATMEL AVR**. To calculate the trajectories of setpoint position and angular velocity of the servo, we used one **USART** port of this microcontroller. **USART** port is used to communicate eighteen servodrivers connected in cascaded through a **I2C** circuit converter (see the Figure 6). There are three nested feedback loops in the distributed control system of the hexapod robot. The first is the internal feedback loop that implements a digital *PID-PI* controller that runs in real time using an embedded microcontroller of servo drive module, which is used to control the position and angular velocity of a robot joint. The second feedback loop is responsible for feedback the force applied at the extreme end of the leg of the robot to determine whether there is a strong support point on the ground surface. This loop is implemented so that the movement of the leg must be recalculated if any obstacle is found on the surface displacement. This loop can be considered as the motion controller of a full-leg of the robot, so that total have 6 motion software controllers, one for each leg of the robot. The third feedback loop is implemented by the onboard master microcontroller ARM Cortex M3 of the mobile robot. This feedback control loop is responsible for controlling the orientation of the robot structure to mantain the balance, as well as the direction of navigation of the robot. The sensors used for feedback direction and orientation of the robot are: a 3-axis accelerometer, 3-axis gyroscope, and a 3-axis magnetometer. These sensors work together to perform the measurement of orientation and direction of the robot. These data are processed using **I2C** bus of the master microcontroller.

VI. Locomotion navigation System

Forward movement of the hexapod robot path has been precalculated with plane $x - z$, by a parabolic trajectory as shown in Figure 7, so that the sequence of steps of the robot is performed by using means of motion control tripod with a setpoint tracking parabolic path. The robot can move forward or backward, move left and right, as well as rotation when moving clockwise or counterclockwise about its geometric center. For simultaneous movements with three legs, used the concept known as walking with tripod or triangle of balance, where the hexapod robot keeps its balance or equilibrium point, whether statically or performing this sustained forward movement or rotation. The concept of static stability is the fact that the center of gravity of the robot is located permanently in the area formed by tripod or triangle of support. The Figures 8

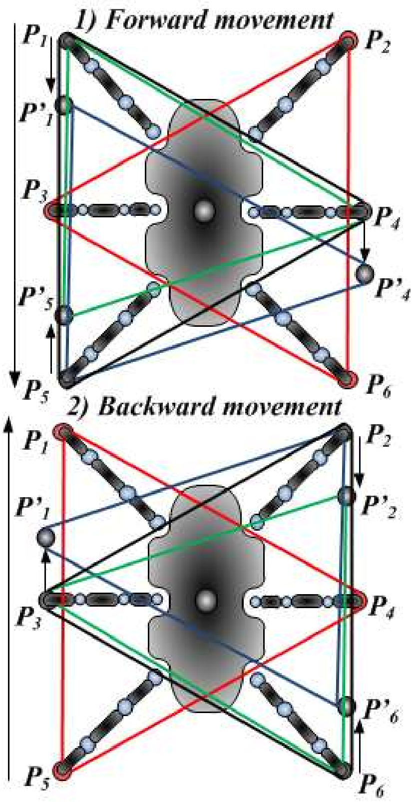

Fig. 8. Tripods of forward and reverse to walking robot.

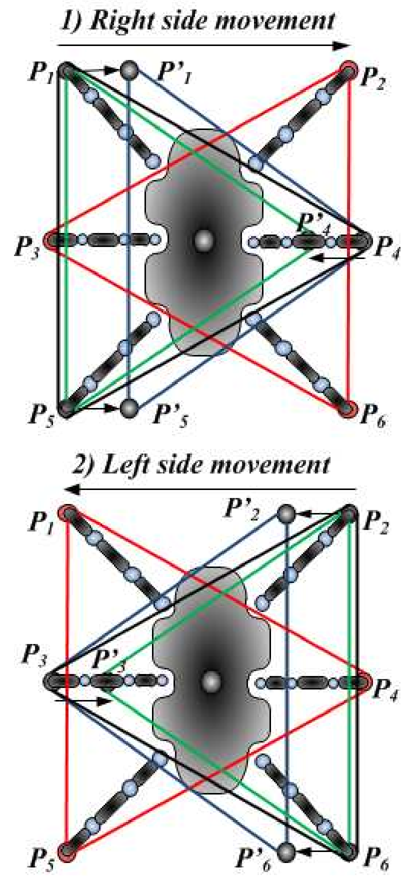

Fig. 9. Tripods of left side and right side to walking robot.

and Figure 9 shows the sequence of tripods (see Table I). For more information about hexapod robots locomotion methods, read the references [3] and [6].

TABLE I
TRIPODS SEQUENCE

Tripod	Forward	Backward	Right side	Left side
A	P_1, P_4, P_5	P_2, P_3, P_6	P_1, P_4, P_5	P_2, P_3, P_6
B	P_2, P_3, P_6	P_1, P_4, P_5	P_2, P_3, P_6	P_1, P_4, P_5
C	P_1', P_4', P_5	P_2, P_1', P_6'	P_1', P_4', P_5	P_2', P_3, P_6'
D	P_1, P_4, P_5'	P_2', P_3, P_6	P_1, P_4, P_5'	P_2, P_3', P_6

VII. CONCLUSION

This paper is about the implementation of a hexapod walking robot with open architecture, which can be reprogrammed for the evaluation of navigation control algorithms with sensorial feedback. Experimental results on the electronic control platform on the robot can be stored on a central computer and plotted in real time via wireless data communication. During the integration of the system components, such as: the communication subsystem, control subsystem and the subsystem of sensorial perception, we could test the feasibility of using electronic systems with embedded 32 bits microcontrollers, which allow further miniaturization of the navigation system mounted in the mobile robot.

ACKNOWLEDGMENT

This work was partially supported in part by DGEST and CONACyT (gant 134534).

REFERENCES

[1] D. Belter, K. Walas, and A. Kasinski, *Distributed Control System of DC Servomotors for Six Legged Walking Robot, 13th International Power Electronics and Motion Control Conference (EPE-PEMC 2008)*, Page(s): 1044 - 1049, 1-3 Sept. 2008.

[2] O.E.L. Botello, M.L.B. García, P. del Carmen Zambrano Robledo, A.R.R. Velázquez, *Design of a Hexapod Robot Based on Insects, 7th International Electronics, Robotics and Automotive Mechanics Conference (CERMA), 2010*, Page(s): 347 - 354, Sept. 28 2010-Oct. 1 2010.

[3] Mark Showalter, *Hexapod Work Space and Walking Algorithm: Work Space Analysis and Walking Algorithm Development for a Radially Symmetric Hexapod Robot*, VDM Verlag, 2009.

[4] S. Manoiu-Olaru, M. Nitulescu and V. Stoian, *Hexapod Robot. Mathematical Support for Modeling and Control 15th International Conference on System Theory, Control, and Computing (ICSTCC), 2011*, Page(s): 1–6, 14-16 Oct. 2011.

[5] Mark W. Spong, Seth Hutchinson, and M. Vidyasagar , *Robot Modeling and Control*, John Wiley & Sons, 2005.

[6] E. Celaya and J.M. Porta, *A Control Structure for the Locomotion of a Legged Robot on Difficult Terrain, IEEE Robotics and Automation Magazine*, Vol. 5, No. 2, Page(s) 43-51, June 1998.