MICRO-ROBOT SOCCER TEAM - MECHANICAL AND HARDWARE IMPLEMENTATION

Frederico Carvalho Vieira
DCA – CT – Universidade Federal do Rio Grande do Norte, Campus Universitário – Lagoa Nova – 59072-970 – Natal - RN
fred@leca.ufrn.br

Pablo Javier Alsina
DCA – CT – Universidade Federal do Rio Grande do Norte, Campus Universitário – Lagoa Nova – 59072-970 – Natal - RN
pablo@leca.ufrn.br

Adelardo Adelino Dantas de Medeiros
DCA – CT – Universidade Federal do Rio Grande do Norte, Campus Universitário – Lagoa Nova – 59072-970 – Natal - RN
adelardo@leca.ufrn.br

Abstract: *This paper describes the hardware implementation of a micro-robot soccer team. The robotic systems were developed to meet the FIRA specifications. A modular development methodology was adopted. Thus, each robot system was divided into three main subsystems. The mechanical sub-system includes all mechanical characteristics related to the robot, such as the body and wheels dimensions, motors specifications and motion characteristics. The robot body and motors arrangement were designed using Auto-CAD. A differential drive system was adopted, allowing the robot to execute straight forward/backward trajectories or circular ones and also to rotate around itself. The motors were chosen to reach a 2 m/s maximum velocity, with high acceleration and torque. The communication sub-system involves the hardware implementation and protocol specification used for data communication between a host computer and the micro-robots. The digital processing of the data protocol was implemented using PIC microcontrollers. The power subsystem was designed considering the motors specifications, such as the peak current and motor torque. This sub-system was implemented using a L293 IC, which is capable to drive two bi-directional DC motors. The proposed system was designed using components acquired in the national market, resulting on a low cost, robust and compact team.*

Keywords: *Mobile Robot, Micro-Robot, Robot Soccer.*

1. Introduction

The Robot-Soccer Championship , as proposed by FIRA (Federation of International Robot-Soccer Association), is a competition created to promote significant developments on robotics, sensor fusion, intelligent control, data communication, vision, mechatronics, computation, artificial life and related areas of engineering and technology (Veloso et all, 1998; Wyeth and Brown, 2000; Wyeth et all, 2000).

Several Educational and Research Institutions around the world have developed efficient robot soccer teams (Veloso et all, 1998; Wyeth et all, 2000), although, such implementations involves top technologies and, consequently, high costs. In Brazil, a few Research Institutions have proposed their own teams (Tavares, 1997; Costa et all, 1999). Due to limited resources, such Institutions were forced to look for low cost solutions.

According to the MiroSot-FIRA rules (FIRA, 1999) , each team is constituted by three robots, whose dimensions can not be greater than 7.5 x 7.5 x 7.5 cm^3. Most of the information processing can be done in a host PC and commands can be transmitted, through radio link, to the robots, by means of unique carrier frequency. These hard constraints constitutes a technological challenging problem. In order to achieve such objectives, a robot soccer team that allies reduced dimensions, robustness and extremely low-cost was developed and its implementation details are presented in this paper.

The robots were developed in a modular fashion, composed by three integrated subsystems, namely: the mechanical subsystem, the communication subsystem and the power subsystem. In this paper, Section 2 describes the Global System focusing on the functional integration of these subsystems. In Section 3, implementation details of the mechanical subsystem are presented. Considerations about the power supply and driving motors designs are presented in Section 4. Aspects of the communication between the host PC and the robot team, part of the communication subsystem, are described in Section 5. In Section 6 is presented the communication and driving subsystems integration through a microcontroller. Finally, the conclusions and future works are presented in Section 6.

2. Architecture Overview

The proposed system follows the general configuration established by the MiroSot-FIRA rules: A vision system, an information processing system, a radio link and the robotic system, as shown in Fig.(1). The vision system consists of a top CCD camera plus a frame grabber connected to the host PC, which processes the captured image of both robot-soccer teams and the ball in order to compute their positions. The information processing system is a software module, running also in the host PC, that computes the global team strategy and the robot motion commands from the positions given by the vision system. The radio link provides the communication between the host PC and the robot team by means of a radio transmitter module connected to the host PC and receivers modules embedded in each robot. The robotic system consists of the three robots of the team, each one including the mentioned receiver module, a microcontroller, and a motor driving system, allowing the robots to execute the commands sent by the host PC.

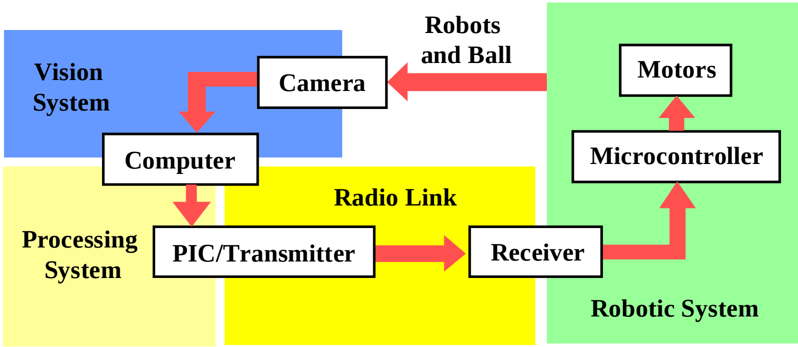

Figure 1. General System Architecture (PIC is a micrcontroller).

3. The Mechanical Subsystem

The mechanical sub-system includes all mechanical characteristics related to the robot, such as the body and wheel dimensions, motors specifications and motion characteristics. The robot body and motors arrangements were designed using Auto-CAD. A differential drive system was adopted, allowing the robot to execute straight forward/backward trajectories or circular ones and also to rotate around its own body. The motors were chosen to reach a 2 m/s maximum velocity, with high acceleration and torque. No mechanical reductions were needed, since the motors have shown a good torque response.

The locomotion system was designed to allow the robots to execute fast and simple movements, like circular trajectories. The locomotion system presents two wheels, each one coupled to one motor .The locomotion system and robot body are shown in fig.(2).

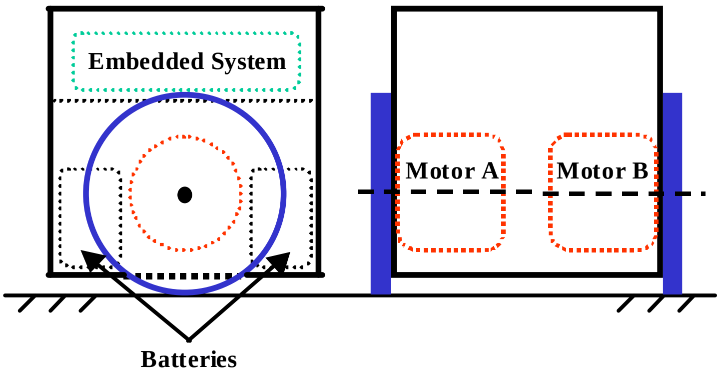

Figure 2. The robot locomotion system.

The robot body was built in acrylic material, with 2mm of thickness, which is very resistant for this purpose. The motors (screwed to the robot body) are localized between the tow batteries, and the embedded system is above an acrylic platform, to avoid direct contact to the motors. The wheels were made using a CAD-CAM system, in order to avoid different wheel sizes. Some mechanical characteristics, like the wheel size, body momentum and weight distribution, have a heavy contribution to the system dynamics and must be considered by the control algorithm. Another problem to face is that the system is time variant, due to the batteries discharge. This characteristic requires a continuous dynamic model adaptation.

4. The Driving Subsystem

On reduced scale mobile robot design, the most common actuators are DC motors. This kind of motor presents several advantages, like: easy to drive, easy to control, and mainly, low cost. On this project, DC motors were used to actuate on the robot wheels, allowing the robot to produce movements. DC Motors are “easy” to drive, because when you apply a positive DC component to it terminals, it produces a direct way rotation. The angular velocity of this rotation is proportional to the DC component applied, that is, as greater it is the DC component, greater will be the angular velocity. The reverse way rotation is analog to the direct way rotation.

There are several ways to drive a DC motor, however, some of them offer a greater loss of the power delivered to the motor. To implement a driving system with low loss of power, it is common to use a well-known driver circuit, the chopper or “H-Bridge”. The H-Bridge is constituted of four power transistors, which are switched from on to off accordingly to an input signal. This specific kind of signal is called PWM (Pulse Width Modulation). The PWM is a logic control signal that provides the information about how the transistors must be switched to produce a desired output. This output applied to the motor will rotate the motor on direct or reverse way, and with different velocity levels. The PWM is a rectangular wave, with constant frequency (constant period) and variable duty-cycle. The duty-cycle determines how much time the signal must hold on high level during one period. This value is specified by period percentages, that is, if the signal holds on high level during half-period, the duty-cycle is said to be of 50%. The input signal is a switched signal that switches between two states, on and off (or logic levels, ‘1’ and ‘0’). Such input signal will produce a similar output switched signal, however, the output signal can assume positive or negative values and it delivers the power needed to drive the motor. Figure (3) shows an example of a H-Bridge output signal, generated by a PWM input signal.

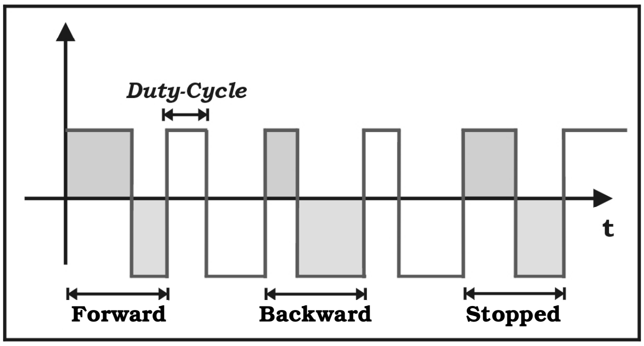

Figure 3. H-Bridge output signal.

Looking at fig.(3) it can be observed that when the green area is greater than the gray one (i.e., the duty-cycle is greater than 50%), the output signal produces a positive DC component. If a positive DC component is applied to a DC motor it will cause the motor to rotate on direct way (i.e., the motor will produce a forward movement). The backward movement is analog to the forward movement. When the green area equals to the gray area, there is no DC component, so the motor stops.

The driving subsystem was developed based on integrated circuits that implements tow H-Bridge. This IC was the L293. The L293 is capable of delivering output currents to 1A per channel. Since the motor peak current is about 700mA, the L293 fits on this requirement. The use of the L293 IC reduced the intern space that would be occupied by a discrete implementation of this circuit. Thus, the implementation of the driving subsystem hardware was simplified by the use of the L293, which requires only the addition of some free-wheeling diodes to protect the motors.

To generate the PWM control signals used to drive the motors, there was used a microcontroller PIC16F84. For each motor a different control signal is needed, in order to allow the independent driving. As the microcontroller is shared with the communication subsystem, the integration of both subsystems will be described in section 7. Figure(4) shows the driving system circuit scheme.

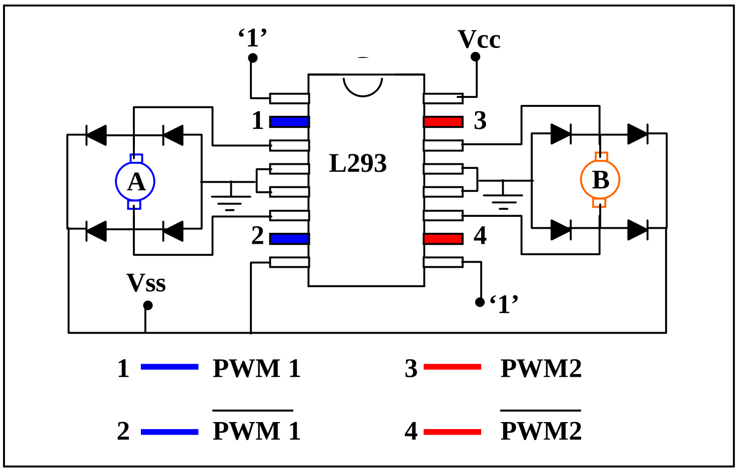

Figure 4. Driving circuit scheme.

On fig.(4) there are two DC motors ‘**A**’ and ‘**B**’, the L293 IC, the protection diodes, and it shows how the components are linked. The colored pins indicate the PWM and its complement input signals for each motor. The **Vcc** symbol indicates the L293 supply voltage. The **Vss** indicates the maximum voltage that can be delivered to the motors. One 9Volts battery is used as power supply to the driving subsystem and another one to the others devices on the embedded system.

5. The Communication Subsystem

In order to allow the host computer to remotely control the robot team, it is necessary a way of communication between them. There are many ways to achieve this goal, however, depending on the transmission rates and data length to be transmitted, some simple protocols can be adopted. After defining the communication protocol, it is necessary to define a protocol data unit, that is, a structure that defines how the data must be allocated to be transmitted and interpreted on the right way. After all, there must be defined the hardware necessary to achieve the communication. All those aspects will be considered in this section.

5. 1 The RS-232C Protocol

The transmission rates requirement is not the main obstacle on the design of this subsystem, since the image capture rate is not so fast (30 frames per second). Thus, we can opt for a simple communication protocol, like the RS-232C, that allows communication rates up to 115.000 BPS. The RS-232C is the most common protocol used for digital communication through the personal computers serial interface. The computers and microcontrollers that implement this protocol have a built-in chip denominated UART (Universal Asynchronous Receiver Transmitter), that manages all information and configurations related to the protocol. The UART defines the maximum transmission rates that can be implemented.

The RS-232C is capable to realize serial bytes transfers, where each byte is preceded by a start bit, that indicates the transmission beginning, and succeeded by a stop bit, that indicates the end of the transmission. This kind of transmission is denominated “8N1”. There are others transmission configurations that allow, for example, to define a parity bit, or even two stop bits.

5.2. The Protocol Data Unit (PDU)

Now that the communication protocol is defined, a protocol data unit (PDU) has to be specified. Since the RS-232C can transfer only one byte per transmission, the definition of a PDU whose length is greater than one byte would require more processing time, and techniques to recompose the PDU. The solution was to define a one-byte length PDU, containing all the information needed to interpret and execute a determinate command. The defined PDU is shown on fig.(5).

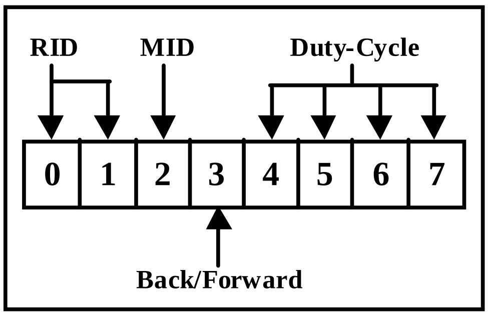

Figure 5. PDU specification.

The fig.(5) shows how the information needed to be transmitted is allocated in fields inside the PDU. The **RID** field indicates the bits used to identify each robot, that is, as each robot uses the same transmission frequency, it is necessary a way to distinguish if a data must be interpreted by a determinate robot. Since the teams have three players (robots), two identification bits are sufficient to distinguish each robot. The **MID** field is the bit used to identify which motor should be actuated. The **Back/Forward** field indicates if the specified motor should rotates on the direct way or on the reverse one. This operation requires just one bit. The reminding bits, 4 to 7, are assigned to motor velocity variation, given by the duty-cycle (as shown in section 4). Using four bits to specify the duty-cycle is equivalent to have sixteen velocity levels for each motor rotation way. This information should be read and interpreted by the embedded communication system (in each robot of the team). After the PDU definition it is possible to know how the programs for transmission and reception should be done, in order to interpret correctly the information transmitted.

5. 3 Hardware Implementation Considerations

On the development of the hardware for transmission and reception, there were used specific communication modules. These modules are radio transmitters and receivers that allow the radio link between the host PC and the robot team. Due to the limited robot dimension (7.5x 7.5x7.5 cm^3), the receiver module dimension should be very reduced. Another requirement was the disposability of equivalent modules with different work frequency, since there are two teams and they must work on different frequencies. After analyzing several modules, the communication modules adopted were the RT-4 (Transmitter) and RR-3 (Receiver) from Telecontrolli. Two sets of different frequencies (315 MHz and 433 MHz) were used.

The next decision was the choice of how the transfer should be done from the computer to the transmitter module. Two possibilities were considered: through the parallel interface, or through the serial interface. Each one presents advantages and disadvantages. Only the first method was implemented and tests are still being taken to verify the adequability of this method. Future works will include the serial interface implementation for comparison ends. To implement the parallel interface method, it was necessary a microcontroller in order to serialize the parallel data sent through the parallel interface.

The transmission software at the host PC was written in C and is responsible for the PDU construction and transmission through the parallel interface. A microcontroller reads that data and serialize it as the RS-232C would do (start-bit, data-bits, stop-bit), and send it to the radio transmitter module. This procedure is shown on fig.(6).

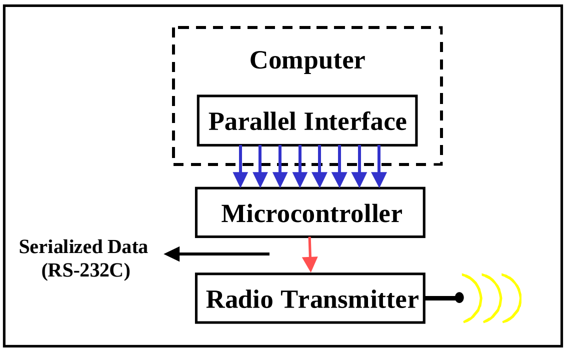

Figure 6. Transmission scheme.

In order to receive the transmitted data, the embedded communication system (in the robots) detects the start-bit, and starts to read the transmitted data. This embedded communication system is constituted of a radio receiver module and a microcontroller. There is another embedded system, the driving system, which shares the same mircrontroller used by the communication system. After receiving the data transmitted, a program in the mircrontroller verifies (through RID PDU field verification) if the received data should be interpreted by that robot, if so, it interprets the data and store them, for further use by the driving system. If RID PDU field verification fails, the system discards the received data. Figure (7) shows the transmitter hardware implementation.

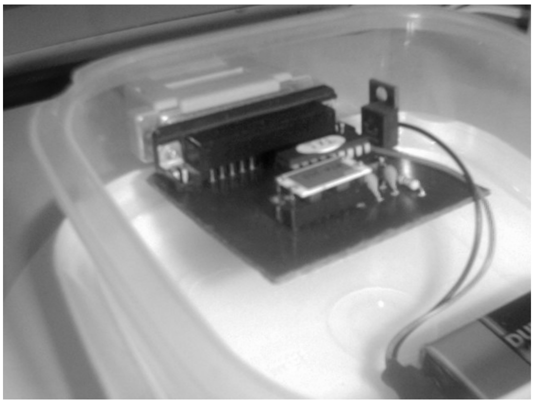

Figure 7. Transmitter hardware implementation.

6. Integrating Subsystems Using a Microcontroller

As mentioned in earlier sections, the embedded systems on each the robot is an integration of part of the communication subsystem and the driving subsystem. This integration is possible through a shared microcontroller that gets information from one subsystem and passes it to the other. The microcontroller tasks are: reception of data, interpretation of the received data and execution of the commands. These tasks can be classified into two groups:

1. Driving Task;
2. Communication Task.

These tasks will be detailed in the subsections that follow.

6.1 Driving Task

The driving task generates the PWM control signal to drive the motors ‘A’ and ‘B’, on an independent way. The algorithm that describes this task is not so complex. The algorithm just needs information about the duty-cycle, that is, the velocity level, for each motor, and the desired PWM work frequency. The frequency must be constant (a 500Hz frequency was adopted). There were defined variables to store these values and four output pins (two for each motor). The communication task passes these values and indicates which motor should actuate and the duty-cycle to be used. This task makes use of the microcontroller timer, and here is the point where the program gets complex. Some microntrollers have a built-in PWM generator, which makes the implementation much simpler. For cost reasons, the microcontroller adopted doesn’t offer those features, requiring the ability to program these functions.

6.2 Communication Task

The main programming obstacle takes place here, where the communication task also requires the use of the micrconcontroller timer. The communication task must receive correctly a RS-232C transmission and interpret the data received. This task makes use of the micrcontroller interrupts. The interrupts allow a program to stop whenever they are called and to jump to an interrupt treatment routine. This feature was used to implement the reception task. When a start-bit is detected, it generates an interrupt, causing the system to jump to the interrupt routine that configures the program to sample the receiver pin in specific times, in order to correctly read the data transmitted. After the complete reception, the program starts the data verification, and signals that the reception task is over, and the system goes back to the normal driving task. When a data is being received, the driving task doesn’t stop, the program schedules the tasks in order to execute both of them without system faults or lost of data. Due to the micrcontroller limited resources (just one timer), the scheduler programming was a little complex, but it works perfectly.

7. Conclusions

The developed robot-soccer team was implemented with a minimum set of integrated circuits, resulting on a very compact onboard system (only three IC’s in an unique printed circuit board of 50 x 40 mm^2). This compactness allied to the absence of any mechanical reduction, leads to a mechanical robustness of the system. Furthermore, it guarantees sufficient free space for future expansions, such as more batteries, embedded sensors or a kick device (solenoid). Figure (8) shows a picture of the developed robot teams.

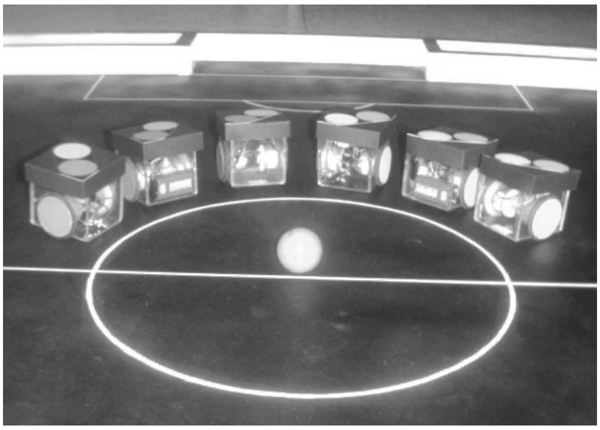

Figure 8. Picture of the developed robot-soccer teams.

The mechanical and driving characteristics of the developed robot system have achieved the Mirosot-FIRA requirements and, besides, resulting in a good torque-velocity relation, which is very important from the competition view point.

The communication protocol makes use of one byte frame size, leading to a simple implementation, with low transmission delays, which is fundamental for this real-time application. In spite of this simplicity, this arrangement can be easily modified to include a limited amount of additional information.

All the mechanical and electronic parts were acquired in the national market (Brazil). The search for modular integrated components results on a extremely low cost implementation (less than US$ 50,00 for each robot). This feature is one of the most important results of this project and is attractive for public research institutions, with limited financial resources.

8. References

Costa, A. H. R., Pegoraro, R., Stolfi, G., Sichman, J. S., Pait, F. M. and Ferasoli, F. H., 1999. "GUARANÁ Robot-Soccer Team: Some Architectural Issues", Proceedings of IV Simpósio Brasileiro de Automação Inteligente, São Paulo, Brazil, pp. 457-462.

FIRA, 1999. Fira Mirosot Game Rules. Approved by the FIRA Executive Comittee and disponible at http://www.fira.net.

Kim, J. H. and Vadakkepat, P., 2000. "Multi-Agent Systems: A Survey from the Robot-Soccer Perspective". Intelligent Automation and Soft Computing, 6(1).

Tavares, R. F., 1997. "Um Breve Histórico dos Robôs no País do Futebol". Disponible at http://www.ia.cti.br/~leopard.

Veloso, M., Stone, P. and Han, K., 1998. "The CMUnited-97 Robot-Soccer Team: Perception and Multi-Agent Control", Proceedings of the Second International Conference on Autonomous Agents.

Wyeth, G. and Brown, B., 2000. "Robust Adaptive Vision for Robot-Soccer". Mechatronics and Machine Vision in Practice.

Wyeth, G., Brown, B. and Tews, A., 2000. "UQ RoboRoos: Prelimary Design of a Robot Soccer Team". Lectures Notes in AI: RoboCup'98.