

# Development of Bluetooth Communications for LEGO-Based Mobile Robot Laboratories

A. Valera, M. Vallés, J.L. Díez, C. García

Dpto. Ingeniería de Sistemas y Automática, Universidad Politécnica de Valencia

**Abstract**—Nowadays, a lot of educational and research objectives can be achieved through the use of configurable, small, low-cost mobile robot kits. Using these systems, students must deal with topics like teamwork, real-world issues, integrated systems building, multidiscipline information, etc.

This paper will present a mobile robot laboratory. The laboratory is based on the LEGO Mindstorms programmed with LegOS. With this environment a wide variety of robot activities can be developed due to its flexibility, power, and simple use.

For the communication between the computer and the LEGO controller, the standard solution is the use of an infrared tower. With this kind of communication, several problems can occur because long distances between the emitter and the receiver are not allowed. In addition, the communication can fail if there is some obstacle between them. In order to avoid these limitations, the paper proposes a new communication system based on a Bluetooth chip.

## I. INTRODUCTION

The relationship between information technologies and process control has reached a new stage [1], encouraging the creation of applications such as monitoring and control through the Internet [2], as well as teleworking, telemedicine, and telerobotics. With regard to process control, information technologies were originally used in the field of education [3], where the Internet increases flexibility in the education process regarding schedules and physical spaces [4] and facilitates remote control applications.

Robotics is closely related with control applications for education. The success of robots in education (from secondary school to undergraduate courses) has come about as a result of a combination of factors [5]. First, students are very motivated with robotics because they can experience

The authors wish to acknowledge the support provided by the Plan Nacional de I+D, Comisión Interministerial de Ciencia y Tecnología (CICYT), and the Vicerrectorado para el Espacio Europeo de Educación Superior (Universidad Politécnica de Valencia) for partial funding of this work, under projects number DPI2002 04432-C03-02, DPI2002 04432-C04-04 and PAEEES-04-115

physically their work.

Nowadays, a second factor in the success of robots is the decreasing cost. Despite in general all the educational equipment is very expensive, now it is possible to acquire several cheap platforms based on robotics (ActivMedia's Pioneer robot [6], MIT's HandyBoard and Cricket controller cards [7], The LEGO Group's LEGO Mindstorms [8], etc.). These platforms usually consist of controllers, electronic sensors, low-cost mechanical parts and/or small robots. They do not provide the same precision than industrial robots, but they are sufficient for educational purposes. In addition, due to their multidisciplinary nature, these platforms are very interesting to promote the teamwork. In this way, each person is responsible for developing a specific part of the project or work: physical robot development, robot programming, behavior planning of strategies, etc.

This paper will present a mobile robot laboratory based on The LEGO Mindstorms platform. For several reasons, this system is very suitable for the development of robot activities. The first reason is the price. LEGO kits, with more than 700 construction pieces, sensors, and communication and microcontroller hardware costs approximately 200 €; thus, it is a very economical solution compared with other robot platforms. Another reason is the flexibility: the LEGO platform supports a suite of sensors and actuators, and the programmable control unit can serve as the basis for a wide variety of programming projects.

Under authors' point of view, LEGO kit has only one limitation: its communication system. Because it uses an infrared emitter, some problems can appear if the distance is very long or there are obstacles between the emitter and the receiver (the computer and the LEGO controller).

In order to avoid this, the paper proposes a new communication system based on radiofrequency. Because the new system uses a Bluetooth chip, the limitations of the infrared system have been solved. In addition, the final system price is also very low.

The paper will also present several examples of mobile robot activities developed using this environment.

## II. THE LEGO SYSTEM

### A. Introduction

In 1998 LEGO released the first Mindstorms set, the Robotics Invention System (RIS 1.0). It is an educational toy for children aged 12 years and older. Apart from those familiar beams, bricks and gears, the kit contains dc motor actuators, a range of sensors, and, most importantly, the RCX component. The RCX is the LEGO's programmable brick that allowed models not just to move, but to sense and respond to their environment. It is based on a Hitachi H8 series microprocessor. This 8-bit CPU provides most of the control logic for the RCX, including serial I/O, Analog to Digital Converter and built-in timers. It even contains 16KB of internal ROM and 33KB of static RAM. In addition, there is an interface to three actuators and three sensors, an infrared (IR) communications interface (to communicate with a desktop computer or another RXC) as well as an LCD, four pushbuttons and a small speaker.

The RCX component was initially developed as an educational tool through the collaboration of LEGO and MIT [9], [10]. On the first version of the RCX allowed six input and six output blocks to be connected to the H8 microprocessor. When LEGO developed the commercial version of the RCX, the number of inputs and outputs was reduced to three of each. Although this change reduced the flexibility of the RCX utilization, it reduced the drain of the batteries that power the system.

The initial intended use of the RCX system was for research and educational activities. Combining the versatile LEGO construction blocks with the easy-to-use programming and I/O interfacing of the RCX provided a fast prototyping system to support these activities. Although the commercialization of this product has focused on the recreational and K-12 educational markets, the flexible and expanding world of LEGO Mindstorms is widely accepted as a tool for research and higher education.

Enthusiasts have extended the hardware and software in various ways [11], [12]. Recent issues of IEEE Robotics and Automation Magazine [13]-[15] or IEEE Control Systems Magazine [16] argues that these extensions show that the LEGO Mindstorms kit can be used to good effect in an education context. In particular, the kit is relatively cheap, robust, reconfigurable, reprogrammable, and induces enthusiasm and innovation in students.

### B. LEGO Components

The components of the LEGO Mindstorms Robotic Invention System are described in detail in [11]; for this article, components are categorized under the headings of sensors, actuators, and information technology.

#### Sensors

The RCX has three sensor ports, each of which can accommodate one of four different LEGO sensors: touch, light, rotational, and temperature sensor.

There are two different classes of sensors: passive and active sensors. Passive sensors operate as simple resistances and require no power from the RCX in order to function. Touch and temperature sensors are examples of passive sensors. Active sensors require power from the RCX and involve slightly more sophisticated electronics. Light and rotation sensors are active sensors.

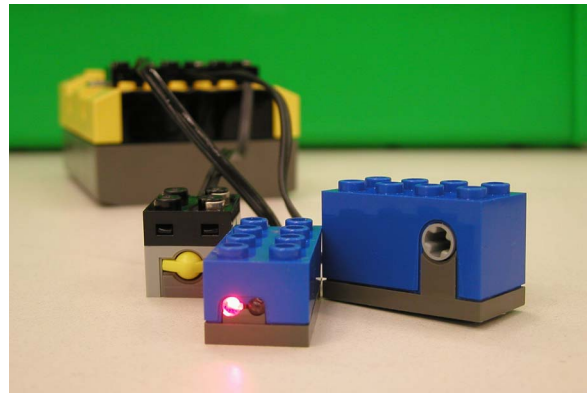


Fig. 1. LEGO sensors. The sensors used in the mobile robot laboratory are: contact sensor (left), light sensor (middle) and rotation sensor (right).

The RCX has no way of knowing what kind of sensor has been hooked up, so at some point a program must tell the RCX what kind of sensor (sensor type) is attached and how to read it (sensor mode). The sensor type determines how the RCX interacts with the sensor. The sensor mode tells the RCX how to interpret a sensor's values.

Every sensor has three separate values associated with it: the raw value, boolean value and processed value. The raw value is the actual ADC reading, which represents the voltage of the sensor and ranges from 0 (0volts) to 1023 (5volts). The boolean value is always either 0 or 1 and is often useful in cases where a simple yes/no decision needs to be made. The range of a processed value is dependent on the sensor's mode. For example, in percentage mode, the processed value ranges from 0 to 100.

All three values (raw, processed, and boolean) are stored so that a program can read them later on. For example, the standard firmware automatically samples each sensor every

3 milliseconds.

### Actuators

The standard LEGO actuator component is a good quality permanent-magnet dc motor with high inertia and low friction. It has an internal flywheel, which acts as a sort of energy storage tank. The actuator is powered using pulse-width modulation (PWM). PWM rapidly switches between on and off. The time that is spent in on is called a pulse, and the duration of this pulse is called its width. The percentage of time that the signal is on is called its duty-cycle. Varying the pulse width creates different levels of power: at low power levels, the pulse is very brief, while at full power the pulse never stops.

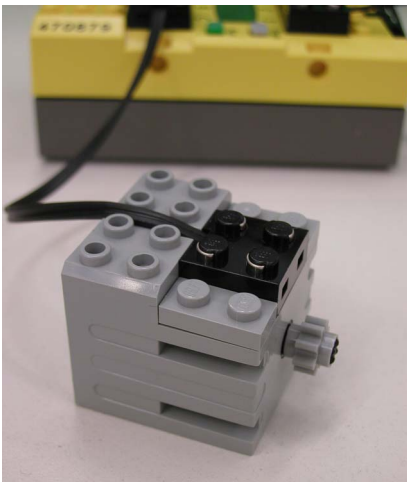


Fig. 2. LEGO motors. The robots are two-wheeled vehicle. Wheels are driven by two LEGO dc motor.

### C. Programming Environment

LegOS is the real-time kernel used in this work [17]. LegOS was designed by M.L. Noga around the standard C language. As shown in Figure 3, the host computer compiles a legOS program (written in C) and generates binary code that can be executed natively on the RCX. This binary is then downloaded to the robot, where the legOS operating system executes the program and provides an interface to the RCX's hardware.

The core of legOS is the library of functions that provide access to the various features of the RCX and the OS. The three main categories are:

- *Output functions:* these functions allow users to interact (primarily) with the motors and the LCD. There are two sets of functions to control the motors: one set that controls the direction of the motors (off, fwd, rev and brake) and another set that controls the speed (between 0 and 255). The LCD functions allow direct control of the LCD to send, for example, a string (or an integer).

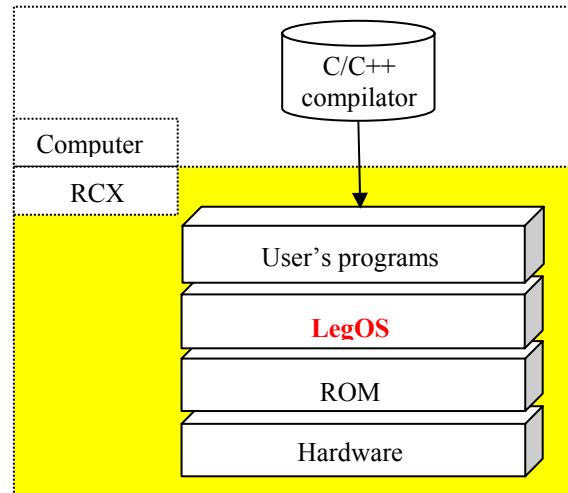


Fig. 3. Programming environment used: users must program in C/C++ language the robot applications. After that, programs are downloaded to the RCX to execute the task.

- *Input functions:* the input functions control both the sensors and the RCX buttons. There are three constants to access directly to the raw sensor values (Sensor\_x, x=1..3). The legOS kernel updates all three of these values virtually continuously. Depending on the sensor kind, there are additional constants to access to touch sensors (TOUCH\_x), light sensors (LIGHT\_x) or rotation sensors (ROTATION\_x).
- *Program control:* with the program control functions users can put the program to sleep for a specified number of seconds or milliseconds with the *sleep()* and *msleep()* functions. Another time management function is *wait\_event()*. This function is used to make a program wait until a particular event has occurred. LegOS also allows the thread executions. A thread is basically a separate function or set of functions, which run side by side with other functions. To use the threading functionality in legOS, users can execute the *execi()* and *kill(pid\_t)* functions. Finally, legOS offers some functions that can be used during critical phases in the main program. There are some functions to control the interrupts (*disable\_irqs()* or *enable\_irqs()*) and the semaphores (*sem\_init()*, *sem\_t()*, *sem\_destroy()*, etc.).
- *Communication functions:* LegOS has its own networking protocol (LNP) to communicate back and forth with the PC. Because every legOS kernel can have its own LNP address, different RCXs can be communicated at once. There are two available types of transmission: broadcast and addressed. The broadcast transmission is used to transmit data that

should be received by everyone who is within range. In the addressed transmission is used for addressing a specific host, and is received only by the host with the specific address. In both cases a packet handler must be declared and registered.

### III. DEVELOPMENT OF A BLUETOOTH COMMUNICATION SYSTEM FOR THE LEGO MINDSTORMS

As previously said, LEGO Mindstorms system includes an infrared (IR) tower for communication between PC and RCXs. This IR tower allows downloading into RCXs the operating system and the programs to be run.

Although IRs are good for short distances and specific communications, they have some limitations related to the maximum angle between devices and maximum distance.

This leads to some problems when IR is used for devices (mobile robots) with freedom of movement, where distances are variable and/or visual contact is missed, or in the case that a communication system among different RCXs is required.

In order to overcome IR limitations a radio-frequency solution has been developed for the LEGO system. Bluetooth<sup>®</sup> has been chosen given the simple solutions that provides for interferences, bands to be used, protocols, ranges, compatibilities, etc.

The developed communications system is a Bluetooth emitter-receiver to be situated in front of the RCX as shown in the next figure. In this way, after and adaptation of IR serial signal to logical levels of Bluetooth chip (UART standard with TTL-CMOS compatible logical levels), the communication between RCXs and PC or among RCXs can be performed.

Although system size is small and can be fixed easily to RCX, smaller versions are being developed.

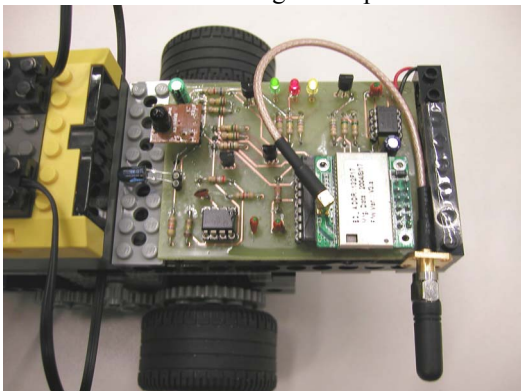


Fig. 4. Developed communications system. The system includes an IR emitter and receiver that communicate with RCX.

The serial communication protocol used by LEGO, works at 2400 bauds, odd parity and one stop bit. In the LEGO IR protocol, the bit encoding is 470 $\mu$ S (that fits exactly with the serial frequency of 2400 bauds) with one start bit, 8 data bits, and one stop bit. A IR '0' bit is encoded as a 470 $\mu$ S pulse train of 38 KHz, and a '1' bit as 470 $\mu$ S of nothing (0v).

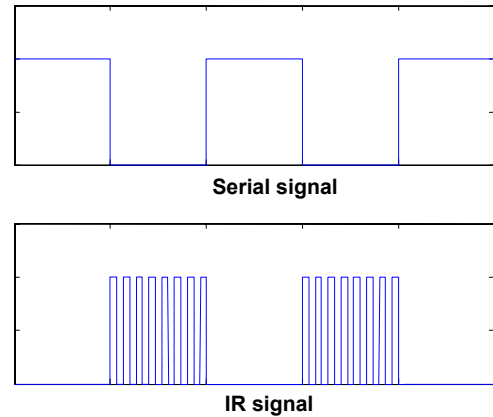


Fig. 5. LEGO serial and IR signals. The IR bits have duration of 470 $\mu$ S. The low level is a pulse train of 38 KHz. The high level is an empty pulse.

The communications system developed consists of a commercial Bluetooth device. This chip is in charge of sending and receiving (via radio-frequency) information from/to PC and/or other RCXs.

This system also includes two stages: the emitter and the receiver. In the first one, the output of the Bluetooth controls (using a switch inverter) the oscillator. When the Bluetooth chip receives (from the PC) a low level bit, the oscillator generates a pulse signal of 38 KHz. If the Bluetooth chip receives a high level bit, the oscillator does not work. The oscillator output is connected to an IR emitter. The reception stage is simpler: the output of the IR receiver is connected directly to the Bluetooth chip. Figure 6 shows the communications system blocks.

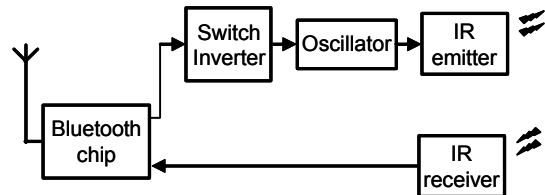


Fig. 6. Communications system architecture. The system has the Bluetooth chip and the emitter and receiver stages.

The selection of this particular Bluetooth device (Promi-ESD chip, by Initium) was motivated for its low cost, its small dimensions and also because Bluetooth Serial Protocol was embedded.

For the oscillator, a precise and very low power

consumption timer TLC555CDR (by Texas Instruments) chip has been used. The IR receiver must be also very precise. Despite there are others more economical solutions, the TSOP-34838 (by Vishay) IR receiver has been chosen because its robustness with the noise and ambient light interferences.

For carrying out the switch/inverter block, also different solutions can be employed. Different versions of this communications system have been implemented using the MAX4561 or MAX3233 (by Dallas-Maxim) chips.

Finally, a precise voltage regulator must be used to provide the TTL tension level (3.3volts) to feed different elements of the communications system (Bluetooth chip, IR receiver, etc.). In this work the MAX884 (by Dallas-Maxim) linear regulator has been chosen.

#### IV. MOBILE ROBOTS LABORATORY PROPOSAL

##### A. Introduction

Several LEGO vehicles with the Bluetooth communication system have been developed for a Mobile Robots Laboratory. Additionally, a PC with a Bluetooth standard adapter is available and it is used as central host. Figure 7 shows a schema of the laboratory configuration.

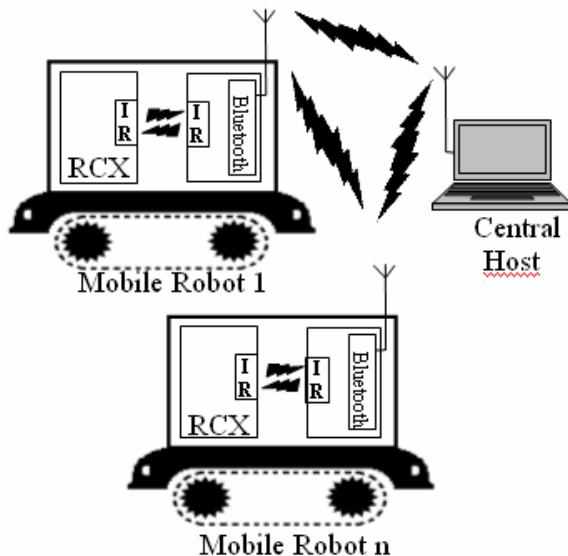


Fig. 7. Mobile Robots Laboratory. Each vehicle has a Bluetooth communication system, then being possible a communication with other robots and central host in a simple, fast, and reliable way.

Different activities have been developed with this laboratory, but all of them can be classified into 2 types: independent and cooperating robots.

##### B. Independent Robots Control

This kind of activities allows the robots to perform typical tasks as: line tracking, pre-programmed movements, avoiding obstacles, etc. In this case, students should write a C/C++ program that, accessing to robot sensors and actuators, perform the required task

BrixCC is used in these activities. It is an environment that includes a text editor, a number of tools that allows, for example, choosing the programming language, and the type and port number to be used when transferring the program. After compilation, the program is transferred to RCX to perform the task.

##### C. Cooperating Robots Control

The communications systems allow a new kind of activities additionally to independent robots control: those activities where mobile robots must interact in order to perform a task. These kinds of applications are common in space robots, where general purpose devices are programmed for different tasks: exploring, assembling, maintenance and repairing space hardware, human assistance in the space, etc.

In order to demonstrate system capabilities, an exploring system have been developed. The application consists of a marsupial robot (supply ship) transporting three mobile robots. Each mobile robot belongs, additionally to general purpose sensors (cheap), a specific sensor (expensive). For example, we can assume that each robot has sensors for temperature, pressure, humidity, wind velocity, and panoramic and navigation cameras. Specific sensors consist of an articulated arm with instrumentation (microscope, X-ray and Mössbauer spectrometers), and abrasion tools, etc.

The marsupial robot must transport mobile robots to a previously defined search zone. Once there, mobile robots should search in their corresponding area. Figure 9 shows and scheme of developed application.

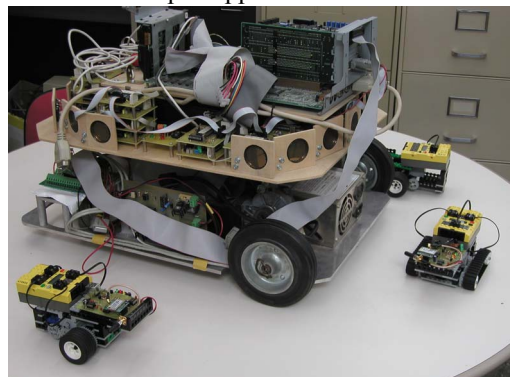


Fig. 8. Explorer system. It is composed by three mobile robots and a marsupial robot in charge of transporting them. The marsupial robot is equipped by an ordinary PC with radio-frequency communications.

If a robot detects a special interest zone, all the other robots are informed about zone co-ordinates. Once all robots have their specific sensors information, they come back to marsupial robot to perform a new task.

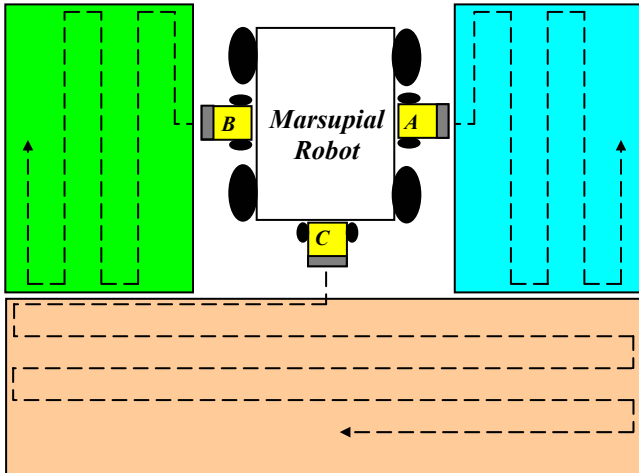


Fig. 9. Explorer system. When the marsupial robot arrives in the searching zone the three exploring vehicles (A, B, and C), are assigned to a searching sub-zone. Once mission is finished, they come back to marsupial robot to perform a new task

Additionally, as far as mobile robots can check their own batteries level, one of them can come back to marsupial robot if its level is very low and then marsupial robot can reorganize the task to be performed among the other robots.

## V. CONCLUSIONS

Robot-design competitions are a popular and effective way to teach robotics, control and real-time systems, computer vision, etc. This work has presented a mobile robot laboratory based on LEGO Mindstorms working with LegOS real-time kernel.

In order to avoid the problems of the original communication system, a new one has been developed. It is based on a Bluetooth chip that is in charge of the communications between RCXs and PC.

Finally, an explorer system based on mobile robots has been presented for testing and analyzing the facilities of the platform used.

## ACKNOWLEDGMENT

Authors thank Alberto Encinas and Joaquin Molck the help to the development of part of this work.

## REFERENCES

- [1] S.H. Yang, X. Chen, and J.L. Alty. "Design issues and implementation of Internet-based process control systems," *Contr. Eng. Practice*, vol. 11, pp.709-720, 2003.
- [2] A. Valera, J.L. Diez, M. Vallés, and P. Albertos. "Virtual and Remote Control Laboratory Development," *IEEE Contr. Syst. Mag.*, vol. 25, no. 1, pp. 35-39, 2005.
- [3] M. Johansson, M. Gäfvert, and K.J. Aström, "Interactive tools for education in automatic control," *IEEE Contr. Syst. Mag.*, vol. 18, no. 3, pp. 33-40, 1998.
- [4] S.E. Poindexter and B.S. Heck, "Using the Web in your courses: What can you do?. What should you do?," *IEEE Contr. Syst. Mag.*, vol. 19, no. 1, pp. 83-92, 1999.
- [5] J.B. Weinberg, and X. Yu, "Robotics in Education: Low-Cost Platforms for Teaching Integrated Systems", *IEEE Robot. Automat. Mag.*, vol. 10, no. 2, pp. 4-5, 2003.
- [6] ActivMedia Robotics Web site [Online]. Available: <http://www.activrobots.com>
- [7] The Handy Board home page [Online]. Available: <http://handyboard.com/>
- [8] LEGO Mindstorms home page [Online]. Available: <http://mindstorms.lego.com/>
- [9] M. Resnick, F. Martin, R. Sargent, and B. Silverman, "Programmable bricks: Toys to think with", *IBM Syst. J.*, vol. 35, no. 3&4, pp. 443-452, 1996.
- [10] S. Papert, "What's the big idea? Towards a pedagogy of idea power", *IBM Syst. J.*, vol. 39, no. 3&4, pp. 720-729, 2000.
- [11] D. Baum. *Definitive Guide to Lego Mindstorms*. Berkley, Ca: Apress, 2000.
- [12] D. Baum, M. Gasperi, R.Hempel, and L. Villa. *Extreme Mindstorms*. Berkley, CA: Apress, 2000.
- [13] J.B. Weinberg and X. Yu, "Robotics in education: Low-cost platforms for teaching integrated systems", *IEEE Robot. Automat. Mag.*, vol. 10, no. 2, pp. 4-6, 2003.
- [14] F. Klassner and S.D. Anderson, "LEGO mindstorms; Not just K-12 anymore", *IEEE Robot. Automat. Mag.*, vol. 10, no. 2, pp. 12-18, 2003.
- [15] L. Greenwald and J. Kopena, "Mobile robot labs", *IEEE Robot. Automat. Mag.*, vol. 10, no. 2, pp. 25-32, 2003.
- [16] P. Gawthrop, and E. McGooin, "A LEGO-Based Control Experiment," *IEEE Contr. Syst. Mag.*, vol. 24, no. 5, pp. 43-56, 2004.
- [17] brickOS home page [Online]. Available: <http://brickos.sourceforge.net>