*Int. J. Engng Ed.* Vol. 23, No. 3, pp. 491–498, 2007
Printed in Great Britain.

0949-149X/91 $3.00+0.00


# Control of Mobile Robots using Mobile Technologies*

ANGEL VALERA
*Dpto. Ingeniería de Sistemas y Automática, Universidad Politécnica de Valencia, 46022 Valencia, Spain. E-mail: giuprog@isa.upv.es*

MARTIN WEISS
*Institute for System Dynamics, University of Stuttgart, Germany*

MARINA VALLÉS
*Dpto. Ingeniería de Sistemas y Automática, Universidad Politécnica de Valencia, 46022 Valencia, Spain*

JOSÉ LUIS DÍEZ
*Dpto. Ingeniería de Sistemas y Automática, Universidad Politécnica de Valencia, 46022 Valencia, Spain*

*Nowadays, many educational and research objectives can be achieved through the use of configurable, small, low-cost mobile robot kits. Using these systems, students must learn to work in teams and deal with topics such as real-world issues, integrated systems building and multidisciplinary information. This paper deals with mobile robot control. It presents a low cost laboratory experiment based on LEGO Mindstorms. In order to avoid the limitations of the original communication system, a new one is proposed. This system is based on Bluetooth and establishes communication between a host computer and/or mobile robots. With this environment a wide variety of robot activities can be developed due to its flexibility, power, and simplicity of use. The paper also presents examples of these activities related with robot control design, artificial vision, trajectories planning, etc.*

**Keywords:** mobile robots; Bluetooth; LEGO

## INTRODUCTION

ROBOTICS is closely related to control applications for education. The success of robots in education (from secondary school level to undergraduate courses) has come about as a result of a combination of factors [1]. First, students are very motivated when they use robotics because they can physically experience their work. Nowadays, a second factor in the success of robots is the decreasing cost.

Despite the fact that educational equipment is generally very expensive, it is now possible to acquire several cheap platforms based on robotics (ActivMedia's Pioneer robot [2], MIT's HandyBoard and Cricket controller cards [3], the LEGO Group's LEGO Mindstorms [4], etc.). These platforms usually consist of controllers, electronic sensors, low-cost mechanical parts and/or small robots. They do not provide the same precision as industrial robots, but they are adequate for educational purposes. In addition, due to their multidisciplinary nature, these platforms are very good for promoting teamwork: each person is responsible for developing a specific part of the project or work—physical robot development, robot programming, behavior planning of strategies, etc.

Although these platforms are very suitable for the development of robot activities, because they are cheap they have reduced communication capabilities. For example, the work presented here will focus on the mobile robot control laboratory based on the LEGO Mindstorms platform [5, 6]: this is cheap, flexible, and has communication ability.

LEGO kits, with more than 700 construction pieces, sensors, and communication and microcontroller hardware costs approximately 200; thus, they provide a very economical solution compared with other robot platforms. LEGO's flexibility is mainly due to the platform that supports a suite of sensors and actuators, and the programmable control unit that is able to serve as the basis for a wide variety of programming projects. However, the LEGO communication system is infrared-based, and some problems appear if long distances are involved or there are obstacles between the emitter and the receiver.

This may only be a slight problem if someone can "help" the robots if communication is lost due to distance, obstacles, or the device is out of the emission angle. However, if a telerobotics platform is required, the problem is greater and someone should be assigned to this task if a reliable platform is to be obtained.

* Accepted 18 March 2007.

This paper presents a simple methodology for enhancing the communication abilities of an educational mobile robot platform by means of Bluetooth ©.

## LABORATORY ACCESS MODE APPROACHES

Information technologies were originally used in the field of education [7], where the Internet increases flexibility in the education process with regard to schedules and physical spaces [8] and facilitates remote control applications [9]. Currently, the creation of applications such as monitoring and control through the Internet [10], as well as teleworking, and telemedicine are quite common.

Based on the Internet, there are two different options for setting up laboratories in learning environments: virtual laboratories and remote laboratories. A virtual laboratory allows, for example, continuous access to a simulated process through a computer.

Halfway between traditional and virtual laboratories are remote laboratories, which offer allow remote users to perform real experiments. Most of the equipment needed for setting up a virtual/remote laboratory is available in traditional laboratories. The only additional element required is an interface between the local application and the Web server [9].

In addition to the well-studied advantages (related to time and space flexibility) of the three approaches to laboratory work (i.e., traditional hands-on, simulations/virtual labs and remote labs), the learning outcomes of each of them are quite different [11]. For example, students using hands-on laboratories become experts in hardware problems, while students using simulations become more familiar with the theoretical contents, and those doing remote experiments are more likely to identify non-idealities in the experimental results.

With the proviso that alternative student access modes may improve some learning outcomes while reducing others, the mobile robotics platform proposed in the paper has been designed in a flexible way, by allowing all kinds of laboratories as explained above. This allows teachers to choose one option or a combination of laboratory methods to get the best possible combination for the required learning outcomes.

The greatest technical problems that were encountered were with the telerobotic (through a network or the Internet) platform [12, 13], because the low-cost approach has poor communication capabilities and a human supervisor is required. The proposed system avoids this problem, because it uses a communication system based on radiofrequency (Bluetooth), thus overcoming the limitations of the LEGO infrared system.

## THE LEGO SYSTEM

*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 over. Apart from the familiar beams, bricks and gears, the kit contains DC motor actuators, a range of sensors and, most importantly, the RCX component. The RCX is LEGO's programmable brick that allows 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 16 kB of internal ROM and 33 kB 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 speaker.

The RCX component was initially developed as an educational tool through the collaboration of LEGO and MIT [14, 15]. 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 RCX system was initially intended 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 [16, 17]. Recent issues of *IEEE Robotics and Automation Magazine* [1, 18, 19] and *IEEE Control Systems Magazine* [20] argue 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 elicits enthusiasm and innovation from the students.

*LEGO Components*

The components of the LEGO Mindstorms Robotic Invention System are described in detail in [16]. In this paper, components are categorized as sensors or actuators.

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

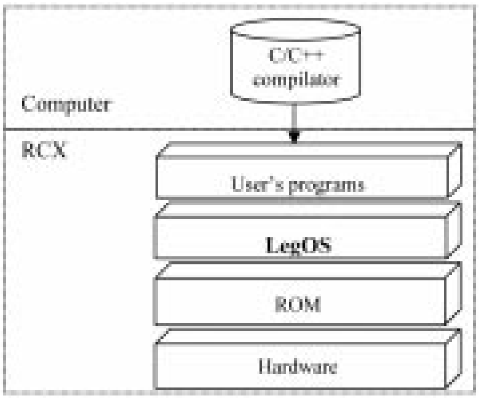


Fig. 1. Programming environment used: users must program the robot applications in C/C++. The programs are then downloaded to the RCX.

temperature sensors. The sensors' values are stored so that a program can read them later. 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). Varying the pulse width creates different levels of power (0–255 in each direction): at low power levels, the pulse is very brief, while at full power the pulse never stops.

*LEGO programming environment*

LegOS is the real-time kernel used in this work [21]. It was designed by M. L. Noga around the standard C language. As shown in Fig. 1, 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 four main categories are output functions (that allow users to interact with the motors and the LCD), input functions (that control both the sensors and the RCX buttons), program control, and communication functions.

By using the program control functions users are able to let a program sleep for a specified number of milliseconds, or make a program wait until a particular event has occurred with the wait_event() function. In addition, legOS also allows thread executions and it has some functions to control the interrupts and semaphores

For the communications, 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, one can communicate with a number of different RCXs at the same time. In addition, there are two available types of transmission: broadcast (to transmit data that should be receivable by everyone within range) and addressed (to transmit data to a specific host).

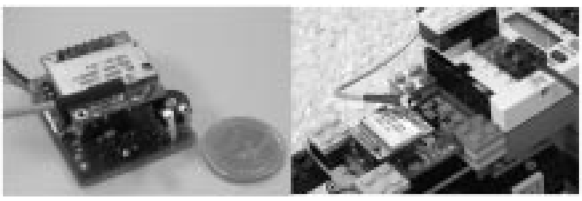

Fig. 2. Communications system, placed in front of each RCX to establish communication.

*Development of a Bluetooth communication system for the LEGO*

The LEGO Mindstorms system includes an infrared (IR) tower for communication between a PC and RCXs. This IR tower allows one to download the operating system and the programs to be run into RCXs.

Although IRs are good for short distances and specific communications, they have some limitations related to the maximum angle between devices and the maximum distance between them. These limitations lead to problems when IR is used for devices (mobile robots) that have freedom of movement, where distances are variable and/or visual contact is lost, or where a communication system between different RCXs is required.

In order to overcome the IR limitations, a radio-frequency solution has been developed for the LEGO system. Bluetooth © has been chosen, giving simple solutions to the problems of interference, bands to be used, protocols, ranges, compatibilities, etc.

The communications system that has been developed is a Bluetooth emitter–receiver, situated in front of the RCX as shown in the Fig. 2. In this way, after the IR serial signal has been adapted to the logical levels of the Bluetooth chip (UART standard with TTL-CMOS compatible logical levels), communication between RCXs and a PC or between RCXs can be performed. Although the system size is small and can be fixed easily to an RCX, smaller versions are being developed.

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 (which fits exactly with the serial frequency of 2400 bauds) with one start bit, 8 data bits, and one stop bit. An 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 (0 volt).

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

This system also includes two stages: the emitter and the receiver. In the first stage, it provides the output of the Bluetooth controls (using a switch

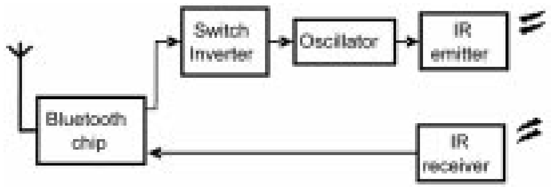


Fig. 3. Communications system architecture. Block diagram of the IR emitter and receiver connections with the Bluetooth chip.

inverter and an 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 3 shows the communications system blocks.

This particular Bluetooth device (Promi-ESD chip, by Initium) was chosen because of 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 was used. The IR receiver must also be very precise. Although there are other more economical solutions, the TSOP-34838 (by Vishay) IR receiver has been chosen because of its robustness against noise and ambient light interference.

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

Finally, a precise voltage regulator must be used to provide the TTL tension level (3.3 V) 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.

## MOBILE ROBOT LABORATORY SETUP

*Introduction*

Equipped with this Bluetooth communication system, LEGO vehicles can be used for laboratory experiments in robotics, mechatronics, real-time systems or control engineering courses. Here LEGO and Bluetooth provide a cheap and flexible solution, which allows many different problem formulations and widens the operation range by using wireless connections.

In the following section we will propose a laboratory setup and different levels of remits that can be used for student projects and laboratory work.

*Laboratory setup*

A standard PC is used, equipped with a Conceptronics USB Bluetooth adapter, which can communicate with several devices at a time using the

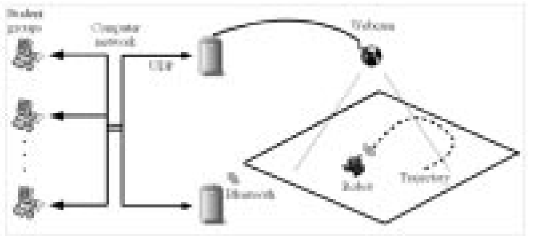

Fig. 4. Laboratory setup: two computers are monitoring the Webcam and RCX; they transmit processed data via UPD to workstations/lab PCs.

"Serial Port Profile" Bluetooth profile. So for every vehicle an emulated COM-Port is created for exchanging data. In order to enable the Bluetooth functions for the Windows operating system, BlueSoleil software is used (www.bluesoleil.com).

To detect the position of the vehicle a second PC, or in our case the same one, is connected to a Logitech Webcam, which is positioned 2–2.6 m above the ground. In order to develop the image processing algorithms, the students can use MATLAB and Video For Matlab (VFM), a free image acquisition toolbox for MATLAB. Subsequently the algorithms can be integrated in a prewritten JAVA framework for higher computational efficiency. Thus the students need only a basic knowledge of JAVA programming.

For the experiments, a JAVA application is run on both PCs; they continuously send the processed data via UDP to all other laboratory PCs. In this way information about the robot's position, images from the Webcam and received data from the RCX are available to all student groups. They can use a JAVA-client, which has to be started in MATLAB, to access the information and can send data back to the RCX. This results in the overall communication scheme showed in Fig. 4.

*Basic robot control*

Based on this setup at first some basic feedforward control actions can be realized by the student groups. Therefore they have to write C programs for the RCX, that control its behavior based on sensor data from touch, light or rotation sensors for example.

For programming the RCX, the Brick Command Center (BricxCC) with a cygwin cross-compiler for legOS (see section above: LEGO Programming Environment) is proposed. BricxCC is a free development environment that supports programming RCX bricks in C, C++, Pascal, Forth, and Java using the legOS, pbForth, and leJOS alternate firmwares (http://bricxcc.sourceforge.net/).

For possible problem formulations, some examples can be devised:

- Linetracking: Using light sensors, the robot must be programmed to follow a line. The track is created on a bright surface using (for example) black electrical tape. The contrast

between the bright background and the black line should be sufficiently large so that the light sensor can easily determine if the sensor is on the line. Different line tracking tests can be suggested, like a timed race, head-to-head racing, etc.
- Preprogrammed movements using encoders: Basic movements such as move forward, go back, turn left/right, etc. can be implemented using encoders included in the LEGO Mindstorms set using a simple controller for the robot position. Stringing together these basic movements, the robot must execute a more complex movement. It is very simple to verify the theoretical and the final robot end position. The less position deviation is obtained, the better is the robot control.
- Obstacle avoidance: Additionally, for planning these movements, the student groups can use MATLAB and VFM to program image processing routines for obstacle detection. Once obstacles are recognized in the image, a possible path to a target location can be calculated with MATLAB and it can be transmitted to the RCX.

For the proposed cases Bluetooth only is used to send such commands as start/stop, move/turn and needed parameters like the angle to turn. Based on this class of tasks, many competitions can be held for student teams, using a variety of problems to be solved (e.g. www.ist.uni-stuttgart.de/robolab).

Figure 5 shows a possible way in which line-tracking can be accomplished. The left illustration shows a mobile robot following a path; on the right there is more detail, showing how the light sensors have been attached to the front of the vehicle. It is therefore possible to determine if and in what direction there was deviation from the line and counter measures can be applied.

This kind of control activity is very easy to realize but the students must take into account that LEGO pieces are not very precise and wheels or tracks always produce slip, so the real movement differs from that measured. The encoders sum up these errors; it results in a significant error over time. After a series of basic movements the measured position will differ from the real position and a correction movement has to be applied. For example, it is possible to use position data from the Webcam to determine the result of a point-to-point movement and calculate the corrective step needed to solve this problem.

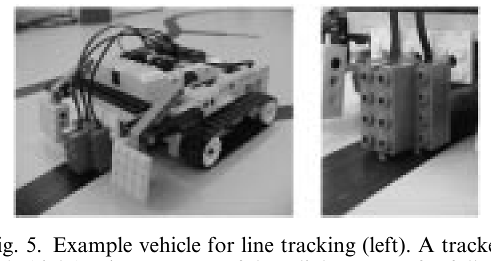

Fig. 5. Example vehicle for line tracking (left). A tracked vehicle (right) using an array of three light sensors for following a path of black tape

*Advanced real-time robot control*

As demonstrated in [22] the laboratory setup shown in Fig. 4 also can be used for continuous control, applying a different system structure.

In the case of advanced control tasks, some problems can appear because the RCX provides only limited computational capabilities due to a missing arithmetic coprocessor. Trigonometric functions or floating point calculations, which are needed for better control algorithms are not possible in an acceptable time. To overcome this drawback and to allow more complex control actions the following setup was proposed, with only the RCX acting as a passive component.

In contrast to the previous section, where the RCX controlled the whole movement, here only velocity controllers are used to ensure a given reference speed for left and right wheels/tracks. Furthermore a more complex controller runs on a PC generating the reference values by using sensor data from the Webcam and the robot sensors. The references and sensor data are exchanged using the Bluetooth communication system.

To enable this, the legOS firmware was extended by a new, faster protocol called LCP (Lego Control Protocol), to guarantee at least 15 samples per second. Furthermore, image processing algorithms developed with MATLAB and VFM were implemented in JAVA for higher sampling rates.

In this way there are almost no restrictions on the complexity of the controllers or applications and the Webcam can be used online and in real-time.

To control the velocity of the LEGO vehicle a common discrete PID controller was implemented on the RCX— in the case of our tracked vehicle, one for each track.

The reference trajectory can be generated using cubic splines. For example the students can rebuild their favorite Formula 1 circuit or simple shapes as in Fig. 6.

For more accurate position and orientation data and to cope with transmission delays, one has to develop observation and filtering algorithms that provide more detailed information about the system state, as for example the orientation that is essential for trajectory control. In control engineering such algorithms are called observers. For the experiment described here, the use of an Extended Kalman Filter is proposed due to its robustness with regard to measurement noise. With the additional data the students can then develop, test and compare different controller types.

For establishing a real-time control in a simple and understandable environment, the graphical tool MATLAB/Simulink was chosen. To force Simulink to run in real-time, a free library called RT Blockset [23] was used.

In addition, in this case, the Webcam can also be

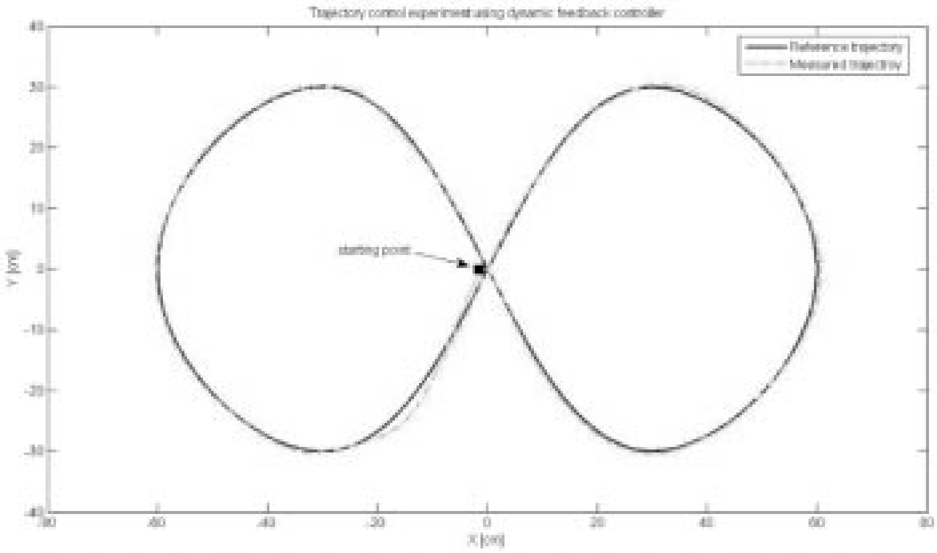


Fig. 6. Example trajectory control experiment using the control algorithms.

used for obstacle detection and a subsequent path-search and trajectory generation [22].

Figure 6 shows the result of a trajectory control experiment using the described setup. An Extended Kalman Filter in combination with a dynamic feedback controller was used to follow the reference trajectory.

*Interacting robot control*

The new communications system allows different kind of activities in addition to independent robots control: those activities where mobile robots must interact in order to perform a task. Examples of these kinds of applications are common in extraterrestrial robotics, where general-purpose devices are programmed for different tasks: exploring, assembling, maintaining and repairing space hardware, human assistance in space, etc.

In order to show such robot interactions, a system for exploring an area has also been developed. The application consists of a marsupial robot (supply ship) that transports some mobile robots. Theoretically, each mobile robot is equipped with a specific sensor, in addition to general purpose sensors such as encoders, dynamometers and navigation cameras. Specific sensors can be for temperature, pressure, humidity, wind velocity sensors, microscope, or X-ray or Mössbauer spectrometers situated in an articulated arm for instrumentation and abrasion tools.

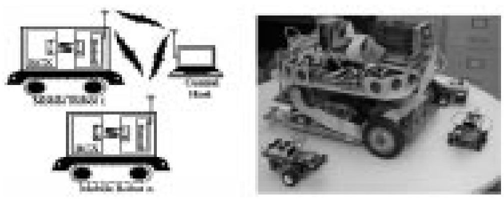

Fig. 7. Possible search mission: a marsupial robot sends out small drones to explore the target region. The drones can interact among themselves and with the PC on the host/marsupial robot (schema on left).

The marsupial robot has to transport mobile robots (see Fig. 7) to a previously defined search zone. Once there, mobile robots should search their allotted area. If a robot detects a zone of special interest, all the other robots are informed of the zone's coordinates. Once the robots have obtained their specified sensor information, they come back to the marsupial robot to perform a new task.

Using the results from the activities of the previous sections, students can further develop the drone's movements to accomplish the mission tasks.

Additionally, students must take into account factors such as low battery levels, and develop interactions between the drones and the marsupial robot to react to such situations. For example, if the battery level is very low, the drone must return to its charging station at the marsupial robot. The host computer then can change the missions of the remaining drones.

Because of the complexity of these problems, they provide a variety of topics for student projects and theses.

## CONCLUSIONS

This paper has described a mobile robot laboratory based on LEGO and Bluetooth. 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 controls communications between RCXs and a PC.

Some frameworks and problem formulations have been presented for use in various engineering courses. Different activities can be performed, depending of the level of studies.

Introductory control courses can work with basic robot control activities. In this case, students must program the RCX in C to perform the proposed tasks. Robot-design competitions are a popular and effective way of motivating students, especially in the first years where the courses are normally very theoretical.

For more advanced courses, students can use the real-time control framework. At this level, students must first implement image processing routines with JAVA and/or MATLAB. After that, they have to develop controllers and signal filters with Simulink schemes.

Finally, for interacting robot control, students can program both the RCX and the host computer to accomplish more complex missions.

Currently, the described experimental set-up is being used for two undergraduate level subjects. Learning is assessed through the evaluation of reports delivered by students, and an increase in the knowledge, depth and quality of the reports have been noticed. Additionally, students are much more motivated than they have been in previous years because of their keeness to achieving good solutions to the problem proposed, the commentaries on the course and the better valuation of this part of the course on an internal teaching poll.

*Acknowledgments*—The authors gratefully acknowledge the support from the Comisión Interministerial de Ciencia y Tecnología (CICYT) and the European Commission for partial funding of this work, under projects number DPI2005 08732-C02-02 and N/04/B/PP 165.011 AutoTech and Fondo Europeo Desarrollo Regional (FEDER). The authors also thank Alberto Encinas for his help in the development of part of this work.

## REFERENCES

1. G. T. McKee, The maturing discipline of robotics. *Int. J. Eng. Educ.*, **22**(4), 2006, pp. 692–701.
2. ActivMedia Robotics, http://www.activrobots.com.
3. The Handy Board homepage, http://handyboard.com.
4. LEGO Mindstorms homepage, http://mindstorms.lego.com.
5. A. Valera, M. Vallés, J. L. Díez and C. García, Development of Bluetooth communications for LEGO-based mobile robot laboratories, *44th IEEE Conference on Decision and Control and European Control Conference ECC'05*, Seville Spain, 2005.
6. T. A. Spedding, A working model of a manufacturing enterprise with internet control. *Int. J. Eng. Educ.*, **17**(2), 2001, pp. 197–206.
7. M. Johansson, M. Gäfvert and K. J. Aström, Interactive tools for education in automatic control, *IEEE Control Syst. Mag.*, 18(3), 1998, pp. 33–40.
8. S. E. Poindexter and B.S. Heck, Using the Web in your courses: What can you do?. What should you do?. *IEEE Contol. Syst. Mag.*, 19(1), 1999, pp. 83–92.
9. A. Valera, J. L. Díez, M. Vallés and P. Albertos, Virtual and remote control laboratory development, *IEEE Control Syst. Mag.*, **25**(1), 2005, pp. 35–39.
10. K. K. Tan , and D. Gillet (guest editors), Innovative approaches to control engineering education, *Int. J. Eng. Educ.,* **21**(6), 2005.
11. E. D. Lindsay, and M. C. Good, Effects of laboratory access modes upon learning outcomes, *IEEE Trans. Educ.,* **48**(4), 2005, pp. 619–631.
12. S. You, T. M. Wang, R. Eagleson *et al.,* A low-cost internet-based telerobotic system for access to remote laboratories, *Artif. Intell. Eng.,* **15**(3), 2001, pp. 265-279.
13. R. Safaric, M. Debevc, R. M. Parkin *et al.*, Telerobotics experiments via Internet, *IEEE Trans. Ind. Electron.,* **48**(2), 2001, pp. 424–431.
14. M. Resnick, F. Martin, R. Sargent and B. Silverman, Programmable bricks: Toys to think with. *IBM Syst. J.,* **35**(3&4), 1996, pp. 443–452.
15. S. Papert, What's the big idea? Towards a pedagogy of idea power, *IBM Syst. J.,* **39**(3&4), (2000) pp. 720–729.
16. D. Baum, *Definitive Guide to Lego Mindstorms*, Apress. Berkley, CA, (2000).
17. D. Baum, M. Gasperi, R. Hempel and L. Villa. *Extreme Mindstorms*, Apress, Berkley, CA, (2000).
18. F. Klassner and S.D. Anderson, LEGO mindstorms; Not just K-12 anymore, *IEEE Robot. Automat. Mag.*, **10**(2), 2003, pp. 12–18.
19. J. B. Weinberg and X. Yu, Robotics in education: Low-cost platforms for teaching integrated systems, *IEEE Robot. Automat. Mag.*, 10(2), 2003, pp. 4–5.
20. P. Gawthrop, and E. McGookin, A LEGO-based control experiment, *IEEE Contr. Syst. Mag.*, **24**(5), 2004, pp. 43–56.
21. brickOS home page http://brickos.sourceforge.net.
22. M. Weiss, Bluetooth-networked trajectory control of an autonomous LEGO-Mindstorms vehicle, Diploma Thesis, Inst. for System Dynamics, University of Stuttgart, (2006).
23. http://digilander.libero.it/LeoDaga/Simulink/RTBlockset.htm

**Angel Valera** received his first degree in computer science in 1990 and his Ph.D. in 1998 from the Universidad Politécnica de Valencia (UPV), Spain. He joined the Department of Systems Engineering and Control (DISA) in 1989. Since then, he has worked as a lecturer and assistant professor. He currently teaches automatic control, robotics and mechatronics. His research interests include robot control, real-time systems and control education.

**Martin Weiss** received his degree in engineering cybernetics from the University of Stuttgart in 2006. He developed the continuous control experiment as part of his diploma thesis. His research interests are mobile robotics, systems biology and nonlinear control.

**Marina Vallés** received her degree in computer science in 1996, completed a Master's Degree on CAD/CAM/CIM in 1997 and her Ph.D. in 2004 from the Universidad Politécnica de Valencia (UPV), Spain. Since 1999, she has been a lecturer at the Department of Systems Engineering and Control (UPV). She currently teaches automatic control and computer simulation. Her research interests include real-time systems, automatic code generation and control education.

**José Luis Díez** received an MSc. in industrial Engineering in 1995, and his Ph.D. in control engineering in 2003, both from the UPV. At the Department of Systems Engineering and Control (UPV) he has taught courses on automation, linear systems control theory and digital signal processing. His interests include complex systems modeling, clustering techniques, intelligent control and control education.