Mobile Robotics as a Tool for Teaching and Learning Embedded Systems Design

Hugo Vieira Neto, João Felipe Calvo Nunes, Alexandre Gabriel Simas and Fábio Kurt Schneider
Post-graduation Program in Electrical Engineering and Industrial Informatics
Federal University of Technology – Paraná
Avenida Sete de Setembro, 3165
Curitiba-PR, CEP 80.230-910, Brazil
hvieir@utfpr.edu.br, jfcn88@gmail.com, simas061287@gmail.com, fabioks@utfpr.edu.br

***Abstract*—In this paper we describe our experience at the Federal University of Technology of Paraná (UTFPR) in using mobile robotics as a tool for teaching and learning embedded systems design in Electronics and Computer Engineering undergraduate courses. We argue that developing a complete working autonomous mobile robot that integrates the concepts from analog and digital electronics, sensors and actuators, control systems and real-time programming has several advantages over working on isolated and disconnected laboratory experiments. Our experience shows that the challenges posed to students in this context result in a higher level of motivation and stimulate their creativity towards generating different solutions to *real-world* problems in embedded systems design.**

I. INTRODUCTION

Autonomous mobile robots are an excellent example of embedded control systems operating in real-time. Even the simplest robot behaviours, such as obstacle avoidance or wall following, demand a series of low-level processes in order to acquire data from sensors and control actuators. There is a rich variety of sensor modalities to be chosen and many different flavours of control strategies to be used. Issues about power autonomy are also present, as mobile robots need to be battery operated. All of these characteristics make the development of autonomous mobile robots a very attractive subject for a course on embedded systems design.

However, using small commercial mobile robots for education, such as the Khepera [1] or the the Lego Mindstorms NXT kit [2] would result in minimising most of the engineering challenges of building an entire robotic system from scratch. On the other hand, developing a mechanical platform for such robots from scratch is not a trivial task, especially for undergraduate electronic and computer engineering students. Our solution for this problem was achieved by delegating the design of the mechanical platform of the robots to students in Mechatronics.

The specifications of our mobile robot platform were initially set to meet the requirements for the Robocup small-size league, but quickly expanded to incorporate distance measuring devices based on infra-red [3] and ultrasound [4] sensing. The sensor modality was chosen to be simple and inexpensive, while still enabling experimentation with basic mobile robot behaviours and the difficulties that arise from using real and noisy sensor readings to achieve the desired robot behaviour. The sensors were chosen also to provide opportunities for students to work with different interfacing techniques, such as analog to digital conversion, pulse width measurements and asynchronous serial communication.

In our mobile robot design, we opted for an omnidirectional drive system using three especially designed wheels [5] angled at 120 degrees, each driven by a DC motor and sensed by a home-made incremental encoder, which provides speed feedback and odometry information. The omnidirectional drive system was chosen to provide students with a range of possibilities for controlling robot motion, as it can be used in full omnidirectional or differential steering drive modes [6].

The processing unit of our mobile robots is based on the ARM core [7], which is a powerful 32-bit architecture that is licensed to several semiconductor manufacturers, such as Atmel, Freescale and NXP (former Philips Semiconductors), among others. Microcontrollers produced by these manufacturers usually include a myriad of on-chip peripherals that facilitate the development of real-time systems [8], [9] and interfacing to sensors and actuators. We have based the processing unit of our mobile robot in development kits from eSysTech (http://www.esystech.com.br/), which is a spin-off company from UTFPR that specialises in developing hardware and software solutions for embedded systems.

The main argument in this paper is that providing students with a large and challenging project to be developed throughout the academic term is far more motivating and yields better learning results than conducting a series of isolated and disconnected laboratory assignments. In our opinion, it is very different for the students to work on several practical assignments about the numerous issues in embedded systems design — concurrent programming, scheduling, device driver development — than *integrating* all or most of these issues in a single design of a complex embedded system, intended to operate in a *real-world* scenario [10]. The latter approach seems to situate students better in the context of embedded systems design due to the very fact of being actively exposed to a problem that demands integration of concepts.

In the next sections we give further details about the construction of our mobile robot platform and describe how we have been using this platform to teach and learn embedded systems design during the past few academic terms.

II. THE UTFPR MOBILE ROBOT

A. Mechanics

The first mechanical platform of the UTFPR mobile robot was designed to comply with the Robocup small-size league rules, i.e. it was cylindrical in shape, with an 180 mm diameter base and 150 mm in height. The first prototypes were completely designed by four teams of mechatronics students during the activities of their "Integration Workshop I" (EL54S) module in the first semester of 2008. Among the best solutions obtained by the students, there was a mobile robot with omnidirectional drive using three wheels specially designed for this purpose. A CAD model of the first prototype is shown in figure 1.

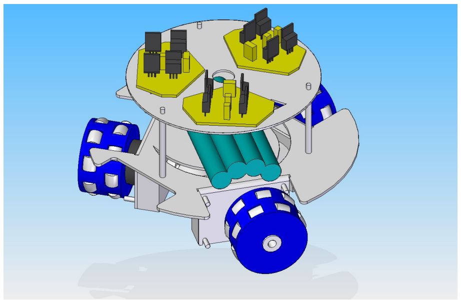
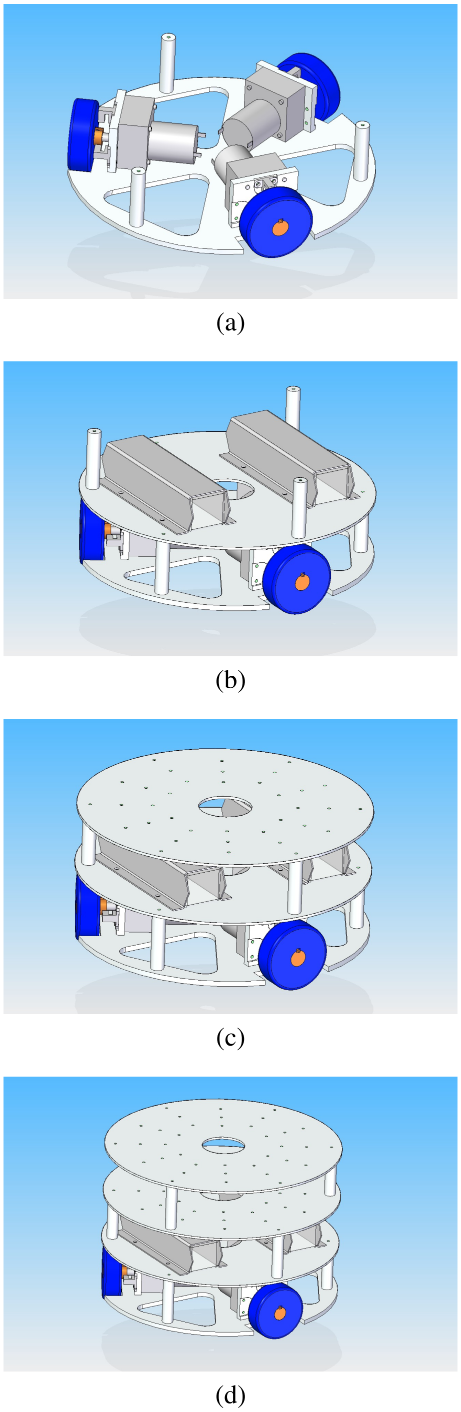


Fig. 1. First mechanical platform of the UTFPR mobile robot, showing the omnidirectional drive system at the bottom, battery packs in the middle and power electronics at the top.

The prototype shown in figure 1 was later used as the mechanical platform by electronics engineering students for the development of the control electronics and programming in order to implement a mobile robot with obstacle avoidance behaviour, as part of the activities of the "Embedded Systems" (J7D480) module. This pilot study of teaching embedded systems design using mobile robotics as a tool to motivate and challenge electronics engineering students was conducted in the second semester of 2008 and was considered very successful — all of the ten student teams involved in the project were able to satisfactorily develop working versions of hardware and software that controlled the mechanical platform to achieve the expected behaviour. More details of what was suggested to be implemented by the students in terms of electronics hardware will be given in subsection II-B.

Due to the success of the experience of using mobile robotics in the "Embedded Systems" (J7D480) module, we decided to build more improved prototypes and keep using them for teaching during the first semester of 2009. During this academic term we are developing and building four new prototypes using the acquired know-how from the past terms. The new prototypes no longer comply with the Robocup small-size league rules, as they are now larger (220 mm diameter base). However, the new prototypes are now designed having the specific use for teaching embedded systems design in mind. A CAD model of our new mechanical design is shown in figure 2.

The mechanical parts are made of aluminium, brass and stainless steel, most of them machined using UTFPR's Mechanics Department machining facilities. Only large mechanical parts, such as the round bases and levels were laser cut by third party.



(a)

(b)

(c)

(d)

Fig. 2. Current mechanical platform of the UTFPR mobile robot: (a) detail of the omnidirectional drive system; (b) detail of the first level, designed to contain the battery packs; (c) detail of the second level, which is designed to contain the power electronics circuit and distance sensors; (d) third level, which is designed as a base for the control electronics circuit.

In the first prototype, students specified double row transwheels from Kornylak Corporation, but in our current mechanical platform, we decided to use single row transwheels. Both models use synthetic rubber coated polypropylene rollers (cat-trak model) in order to provide good grip to the floor and are shown in figure 3.

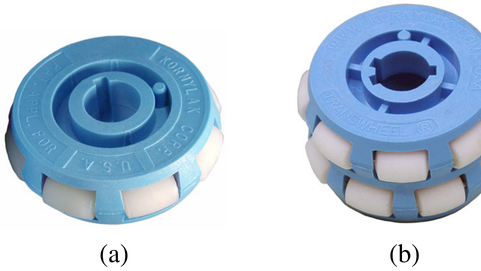

(a) (b)

Fig. 3. Kornylak cat-trak transwheels: (a) 2051KX - single row model; (b) 2052KX - double row model.

The main advantage of adopting a three wheel omnidirectional drive system is its versatility of use. Students are free to implement their drive system as full omnidirectional, which is able to move the robot in every direction without the need of reorientation, or a simpler differential drive system based on two active wheels and one steering wheel, which needs reorientation of the "front" of the robot in order to switch to different directions [6].

Our design philosophy intends to provide students with different implementation options and expose them to the advantages and limitations of each choice — it is extremely important and interesting that engineering students learn the implications of their design trade-off decisions. Therefore, we try to provide room for the emergence of different approaches as much as possible. Versatility in the use of the prototype is what has driven its hardware design, not only for mechanics but also electronics, as will become evident later on.

B. Electronics

The electronics hardware of the UTFPR mobile robot is divided in two main circuits — power electronics designed to drive the actuators, and control electronics comprising the processing unit and sensors. The core of the system is in its processing unit, which is currently based on eSysTech's eAT55 development kit for the Atmel AT91M55800A microcontroller [11], which consists of an ARM7TDMI core and several on-chip peripheral devices. Some of the most important features of this microcontroller for the development of our mobile robot are:

- 8-level priority vectored interrupt controller
- 58 programmable I/O lines
- 6-channel 16-bit timer/counter
- 3 USARTs
- Master/slave SPI interface
- Programmable watchdog timer
- 8-channel 10-bit analog-to-digital converter
- 2-channel 10-bit digital-to-analog converter
- Advanced power management controller

The eAT55 development kit also includes other several important features [12]:

- External SRAM memory (up to 1 MB)
- External FLASH memory (up to 8 MB)
- JTAG interface for programming and debugging
- Availability of the microcontroller's external bus

The existence of a wide range of peripheral devices is a very important characteristic of the specified processing unit, having in mind the idea of offering students several different implementation possibilities of sensing and control strategies necessary to achieve the desired robot behaviour. Having more peripheral devices available increases their chances of implementing creative solutions and exercising trade-off decisions, which are both crucial for education in engineering.

Figure 4 depicts a block diagram of the electronics hardware of UTFPR's mobile robot, showing the connections to sensors and actuators. There are three DC motors individually driven by H-bridges and powered by a 14.4 V NiMH battery pack. Concerning sensors, there are three incremental encoders, each mechanically attached to the axis of each DC motor, and up to six distance measuring devices available to be interfaced to the processing unit. The control electronics is powered by a 4.8 V battery pack, aided by a DC-DC power management integrated circuit.

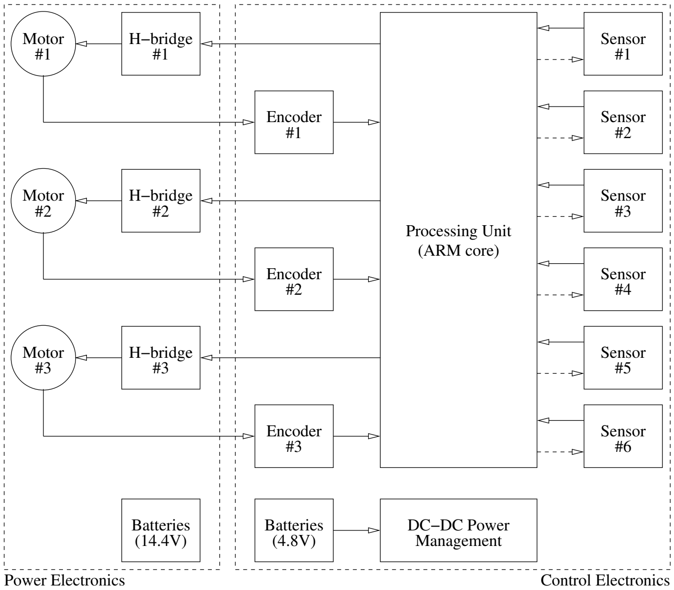


Fig. 4. Block diagram of the electronics of the UTFPR mobile robot. Dashed lines separate the power electronics circuit (actuators) from the control electronics circuit (sensors and processing unit), which have two isolated power sources.

Motors. The DC motors used in our prototype were specified to provide enough torque to move the robot at relatively high speeds. The motors used are equipped with gear reduction boxes that yield 0.6 kgf.cm at 538.6 rpm when powered at the nominal operating voltage of 12 V. Figure 5 shows a picture of the motor model we have used, which is manufactured by Inmepe Maia (http://www.maia.ind.br/).

Fig. 5. Inmepe Maia's motor model MN37-5655 (1:22.28 gear reduction).

In order to drive the specified DC motors, we have used the MC33932 integrated circuit manufactured by Freescale Semiconductor [13], which consists of two H-bridges that are able to drive currents of up to 5.0 A in a single device. The use of integrated H-bridges is extremely convenient and makes DC motor control — direction and speed — possible through pulse width modulated (PWM) signals, which can be easily generated by the Atmel AT91M55800A microcontroller's 16-bit timer/counter operating in waveform mode or simply by software. The rotational speed of the motor will be proportional to the duty cycle of the PWM signal applied to the H-bridge controlling it.

Encoders. Incremental encoders were designed in order to provide speed feedback from each motor. For that, we have used GP1A51HRJ00F transmissive photointerrupters from Sharp Microelectronics [14] and home-made rotary encoders, which were laser printed in overhead transparencies (acetate film), in a similar fashion to what is suggested in [15]. Figure 6 shows a picture of the components of our home-made incremental encoder.

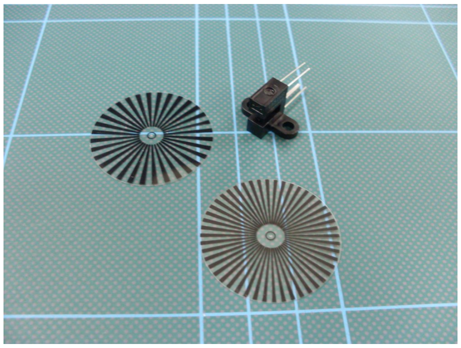


Fig. 6. Home-made incremental encoder: the picture shows two circular patches of acetate film — one with 64 and the other with 96 alternated transparent and laser printed opaque radial strips — and a Sharp GP1A51HRJ00F transmissive photointerrupter.

Our home-made encoders consist of a circular patch of acetate film with alternated transparent and opaque radial strips — either 64 or 96 alternating strips, depending of the desired resolution. A circular patch is attached to the axis of each motor and inserted into the slit of a transmissive photointerrupter, as illustrated in the CAD model shown in figure 7.

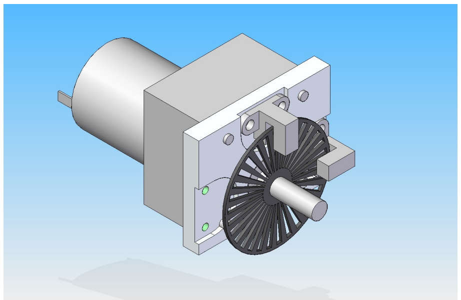


Fig. 7. CAD model illustrating our incremental encoder attached to a motor. Our mechanical design is versitile enough to allow the use of two transmissive photointerrupters per motor, in case the implementation of a quadrature encoder is desired. However, the standard configuration in our prototype uses only one transmissive photointerrupter per motor, implementing incremental encoders that are not able to sense motor direction.

When a transparent radial strip is aligned with the sensor, light is transmitted through the slit, causing the sensor to output a low logic level, but when an opaque radial strip aligns with the sensor, the light beam is interrupted, causing the sensor to output a high logic level. Therefore, when the motor is powered, the photointerrupter outputs a rectangular wave, whose frequency is proportional to the rotational speed of the axis. The frequency (or the period) of this rectangular wave can be easily measured by the 16-bit timer/counter available in the Atmel AT91M55800A microcontroller or by software through interrupt handling.

Sensors. Two low-cost distance measuring devices were specified for use in our mobile robot, one of them based on infrared light reflections — the Sharp GP2Y0A02YK sensor — and the other based on ultrasound echoes — the MaxBotix MaxSonar-EZ1 sonar. Both sensors are shown in figure 8.

(a) (b)

Fig. 8. Distance measuring devices: (a) Sharp GP2Y0A02YK infrared sensor; (b) MaxBotix MaxSonar-EZ1 ultrasound sensor.

The Sharp GP2Y0A02YK infrared sensor is able to measure distances from 0.2 to 1.5 m and provides an analog output signal, which can be easily read using the Atmel AT91M55800A microcontroller's analog-to-digital converter channels. The analog output voltage of the GP2Y0A02YK infrared sensor decreases non-linearly as the distance being measured increases, demanding the use of an interpolation function or look-up-table in the software to interpret the readings correctly. According to the manufacturer, the use of a triangulation method to determine distance makes the sensor very robust to variations in the reflectivity of the object and environmental temperature changes [3].

The other available choice for distance measuring device is the MaxBotix MaxSonar-EZ1 sensor, which is a very reliable and versatile device that provides the output reading in three different ways. The distance measured by the sonar — in the range of 0.15 to 6.5 m — can be read as a PWM signal, as an analog voltage or as an asynchronous serial digital output [4].

If the analog voltage output of the MaxSonar-EZ1 is used, the distance measurements can be read using the Atmel AT91M55800A microcontroller's analog-to-digital converter channels, in a similar fashion to how the GP2Y0A02YK infrared distance sensor is interfaced. However, the voltage levels provided by the MaxSonar-EZ1 are significantly low and an extra external amplifier may be required to increase the distance resolution of the system. The distance measurements can also be read using the PWM output, whose duty cycle can be measured by the Atmel AT91M55800A microcontroller's 16-bit timer/counter operating in capture mode. Alternatively, the MaxSonar-EZ1 can also be read by the Atmel AT91M55800A microcontroller through asynchronous serial communication. The MaxSonar-EZ1 can operate in free run mode (continuous operation) or triggered mode, which allows the multiplexation of several devices in a single USART channel of the microcontroller with a minimum of extra hardware.

The MaxSonar-EZ1 sonar is itself a complete embedded system and the MaxBotix website (http://www.maxbotix.com/) provides interesting literature regarding the objectives and trade-off decisions taken in its design, which is always worth being brought to the attention of the students as a plus to an embedded systems course module.

C. Power

A mobile robot's power system is of utmost importance to its autonomy. In our prototype, we designed separate power sources for the power electronics circuit and the control electronics circuit (figure 4) — the robot's omnidirectional actuator system is powered by a 14.4 V NiMH battery pack (12 Sony 2500 mAH AA cells) and the processing unit is powered by a 4.8 V battery pack (4 Sony 2500 mAH AA cells) and a DC-DC power management integrated circuit.

Discharge of the 14.4 V battery pack does not have a significant influence in the overall operation of the robot, except for its average navigation speed, which is compensated up to certain limits by the feedback control system provided by the incremental encoders. However, the existence of precisely regulated voltages are crucial for the correct operation of digital circuits, which are powered by the 4.8 V battery pack. For that reason, switching DC-DC converters must be present in the power source for the control circuit in order to assure adequate power supply voltages, compensating battery discharge.

In our power source design, we employ the MC34704 power management integrated circuit manufactured by Freescale Semiconductor. This quite remarkable integrated circuit is able to operate with input voltages ranging from 2.7 to 5.5 V and provide up to eight independent and regulated power outputs with significant current capacity. Table I lists the specifications of each regulator available in the MC34704 integrated circuit, along with their typical target applications [16].

TABLE I
MC34704 DC-DC POWER MANAGEMENT CHANNEL OUTPUTS.

Regulator	V_{OUT} typ (V)	I_{OUT} max (mA)	Target application
REG1	5.0	500	+5V ref
REG2	2.8 / 3.3	500	μC I/O
REG3	1.2 / 1.5 / 1.8	550	μC core
REG4	1.8 / 2.5	300	DDR
REG5	3.3	500	μC I/O
REG6	15.0	60	+ref
REG7	-7.0	60	-ref
REG8	15.0	30	Backlight display

The components from Freescale Semiconductor that we are currently using, namely the MC34704 power management integrated circuit and the MC33932 dual H-bridge integrated circuit (described in subsection II-B), are surface mount devices (SMD), which make them difficult to be used in student assemblies within class laboratories. In order to minimise these difficulties, we have developed the adaptor boards shown in figure 9, which can be easily connected to protoboards or custom printed circuit boards and become more accessible to the students.

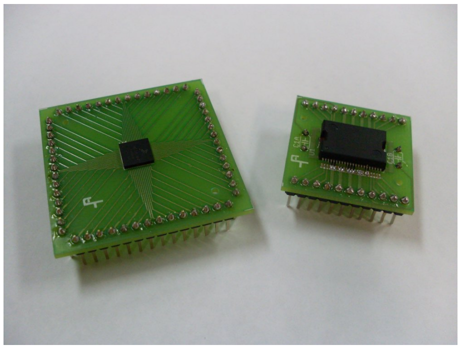


Fig. 9. SMD adaptor boards: a 56-pin QFN adaptor for the MC34704 is shown on the left and a 44-pin HSOP adaptor for the MC33932 is shown on the right.

III. EMBODIMENT AND BEHAVIOUR DESIGN

Having mechanical platforms and processing units available, the task we have been assigning to the students enrolled in "Embedded Systems" is to design a mobile robot that is able to navigate as fast as possible while avoiding obstacles in its operating environment. Although this behaviour seems simple at first, students are required to design the necessary hardware interface to sensors and actuators, and also to implement software in terms of device drivers and higher level control routines to generate the desired robot behaviour. This task can be posed as some sort of competition between student teams in order to motivate their best efforts to obtain a fast moving robot, smooth trajectories, full use of the omnidirectional drive system or any other possible distinctive characteristics that may aggregate value to their final solution.

Students are given the choice to use the earlier described infrared or ultrasound distance sensors (or both) in their robot design. We recommend that they use from three to six sensors spatially distributed over the robot's mechanical platform in order to make the desired behaviour feasible. The choice of how to arrange sensors spatially is what we call the "embodiment design" of the robot. Some of the ideas that are initially given to the students regarding the arrangement of sensors over the robot are shown in figure 10.

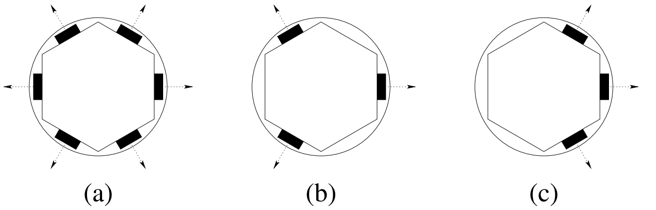


Fig. 10. Suggested sensor configuration arrangements: (a) six distance sensors angled at 60 degrees; (b) three distance sensors angled at 120 degrees; (c) three distance sensors angled at 60 degrees. Dotted arrows indicate the direction of distance measurements, either by infrared or ultrasound sensing. All robot configurations are viewed from the top.

Sensor arrangements (a) and (b) in figure 10 tend to facilitate the implementation of full omnidirectional drive robots, while arrangement (c) practically imposes that the robot has a frontal side facing towards the direction where the sensors are pointing and in this case a differential steering drive seems to make more sense. In case arrangement (b) is chosen, one must decide if the sensors are to be mounted aligned with the robot's wheels or not.

Student teams — usually a group of three to four students — are free to choose the kind of sensors they wish to use and their physical distribution over the mechanical platform of the robot, which are not limited to the examples given in figure 10. Their decisions are encouraged to be geared towards achieving the best possible performance for the desired robot behaviour, be it navigation speed, smoothness of the resulting trajectories, omnidirectional navigation or any other relevant criteria. Obviously, trade-off engineering decisions will naturally emerge in this matter.

The embodiment design will naturally bias what we call the "behavioural design" of the robot, which we understand to be the software implementation of low and high-level control routines. The lowest software level involves the development of device drivers to control actuators and read sensors. The highest software level involves implementing the final robot behaviour, usually through a reactive approach by tightly connecting sensor readings to actuator response [15] for simplicity.

The behavioural design is usually divided into milestones along the academic term, in order to give student teams a rough guideline of the activities to be executed. Some of the major milestones are listed as follows:

1) Development of a software device driver to set motor speed and direction. Generation of PWM signals using the microcontroller's timer/counter in waveform mode is encouraged.
2) Development of a software device driver to read incremental encoders. Estimation of speed using the microcontroller's interrupt inputs or timer/counter in capture mode is encouraged.
3) Development of a software device driver to control motor speed using the two previously developed device drivers. The use of proportional control is encouraged.
4) Development of a software module to control robot navigation (translation and rotation). The use of full omnidirectional or differential steering drive navigation are possible depending on the embodiment design of the robot.
5) Development of a software device driver to read distance sensors. The Sharp GP2Y0A02YK infrared sensors present no other option than using the microcontroller's analog-to-digital converter, while the MaxBotix MaxSonar-EZ1 sensor is much more versatile in this sense — the multiplexation of one of the microcontroller's USART channels to communicate with multiple MaxSonar-EZ1 sensors is encouraged.
6) Development of the high-level behavioural routine of the robot, coupling sensor readings to motor responses to achieve the desired robot behaviour. The use of reactive approaches is encouraged.

Our embedded systems design classes are organised in four-hour theoretical or practical laboratory weekly sessions. Most of the theoretical sessions are concentrated in the beginning of the academic term and include a written examination biased towards the software implementation of relevant device drivers for the mobile robot project. The final ten to twelve weeks (about three quarters) of the academic semester are reserved to practical sessions involving the activities related to the milestones listed earlier. The final examination consists in a practical presentation of a working prototype of the mobile robot by each student team and the delivery of written documentation.

IV. CONCLUSION

We presented an approach for teaching and learning embedded systems design using mobile robotics as a tool and reported our recent experience using this approach in the "Embedded Systems" (J7D480) module, in the context of UTFPR's Electronics Engineering undergraduate course. This approach stems from a similar experience that was more biased towards mechanical aspects, which we previously applied to the "Integration Workshop I" (EL54S) module in UTFPR's Mechatronics undergraduate course. In fact, a mechanical platform developed by a team of mechatronics students served as the basis for the development of an enhanced version of the robot, which is currently in use.

Our approach is founded on exposing students to a relatively large and complex, but motivating mobile robotics project rather than assigning small isolated laboratory experiments that only focus on localised and limited aspects of embedded systems design. The main intention of our approach is to expose students to trade-off engineering decisions that only arise when one is working on a relatively large scale design, which seems to stimulate their creativity to find solutions to engineering problems and provides them with a sneak preview of the issues involved in real-life designs.

Details of the teaching method we developed during the past year — which is constantly being improved — were presented, as well as most technical details of the UTFPR mobile robot design. The results of this semester's activities are intended to be extended in the "Integration Workshop III" (IF66J) module, in the context of UTFPR's Computer Engineering undergraduate course. Next semester we are planning to assign the computer engineering students the task of controlling our fleet of four robots to cooperatively build maps of their operating environments using swarm intelligence concepts and wireless communication. For that, we intend to have our ARM microcontroller running eSysTech's X real-time operating system.

ACKNOWLEDGEMENTS

The authors thank Freescale Semiconductor for providing free component samples. Financial resources for third party services and material for the construction of the mobile robot prototypes were provided by UTFPR's Electronics Department, whose support is gratefully acknowledged. Finally, the authors also thank UTFPR's Mechanics Department for supporting the construction of our mechanical prototypes.

REFERENCES

[1] F. Mondada, E. Franzi, and A. Guignard, "The development of Khepera," in *1st International Khepera Workshop*, Paderborn, Germany, December 1999, pp. 7–14.

[2] D. Benedettelli, *Creating Cool Mindstorms NXT Robots*. Apress, 2008.

[3] Sharp Microelectronics, "GP2Y0A02YK – Long distance measuring sensor," product datasheet, January 2006.

[4] MaxBotix Inc., "LV-MaxSonar-EZ1 – High performance sonar range finder," product datasheet, July 2007.

[5] P. F. Muir and C. P. Neuman, "Kinematic modeling of wheeled mobile robots," *Journal of Robotic Systems*, vol. 4, no. 2, pp. 281–340, 1987.

[6] H. R. Everett, *Sensors for Mobile Robots: Theory and Application*. A. K. Peters, 1995.

[7] D. Seal, *ARM Architecture Reference Manual*, 2nd ed. Addison-Wesley Professional, 2001.

[8] A. Burns and A. Wellings, *Real-Time Systems and Their Programming Languages*, 2nd ed. Addison-Wesley Longman, 1996.

[9] A. C. Shaw, *Real-time Systems and Software*. Wiley, 2001.

[10] T. Bräunl, *Embedded Robotics: Mobile Robot Design and Applications with Embedded Systems*, 3rd ed. Springer, 2008.

[11] Atmel Corporation, "AT91M55800A – AT91 ARM Thumb microcontrollers," product datasheet, July 2001.

[12] eSysTech, *eAT55 Evaluation Board User's Manual*, 4th ed., eSysTech - Embedded Systems Technologies, Curitiba, Brazil, March 2006.

[13] Freescale Semiconductor, "MC33932 – 5.0 A throttle control H-bridge," product datasheet, January 2009.

[14] Sharp Microelectronics, "GP1A51HRJ00F – Case package transmissive photointerrupter," product datasheet, October 2005.

[15] J. L. Jones, B. A. Seiger, and A. M. Flynn, *Mobile Robots: Inspiration to Implementation*, 2nd ed. A. K. Peters, 1999.

[16] Freescale Semiconductor, "MC34704 – Multiple channel DC-DC power management IC," product datasheet, July 2008.