Development of an exploration land robot using low-cost and Open Source platforms for educational purposes.

D Hernández, H Trejo and E Ordoñez.
Instituto de Ingeniería y Tecnología. Departamento de Física y Matemáticas.
Universidad Autónoma de Ciudad Juárez.
Av. del Charro no. 450 Nte. Col. Partido Romero, CP 32310, Tel. +52 656 688-4801
E-mail: hdz.david@me.com

Abstract. In this paper we present the didactic experience of building a low-cost robot composed of sensors, actuators, general electronics and already available frameworks. The control of the robot is through the usage of commercial Open Source platforms as Arduino; and the Raspberry Pi. The experience ranges from general conceptualization, mechanical, electric and electronic design, microcontroller programming and communications.

1. Introduction
Engineering physics can be defined as the direct application of mathematics and physics in the resolution of engineering problems. A direct example of application is robotics. Robots are devices composed of actuators, sensors and electronic circuits, which find their foundation in electrodynamics, classical mechanics and solid state physics. Another important subject around robots that has a direct relationship with physics engineering is control theory, where physics and advanced mathematics are used to solve robot control problems. This two approaches make robots a good supporting tool for teaching and demonstrating physics in class. It motivates and assist students to develop problem-solving skills, as well as help them understand the real life applications of the concepts they are learning.

A strong drawback of the usage of robots as a tool to teach physics is their high cost, which range from $350 USD (Lego Mindstorms [1]) up to $10,100 USD (Robotino [2]). This can make them inaccessible to low-budget schools and universities. Many authors have tackled this problem in several ways, for example, with the use of hobbyist equipment for their projects [3-6].

In this paper we illustrate the experience of building an inexpensive robot with the main objective of explaining the details and shortcuts for the general idea to be implemented in any school. Even if the cost of this robot is not as low as other options ($14 USD, KiloBot [7]), it provides the student the experience of building a robot, experience that he will never get only by experimenting with a pre-assembled robotic kit.

2. Development platforms.
In order to create the systems that monitor and control the robot, we need to use microcontrollers and computers. For this purpose, we used already available development platforms. They were
chosen by their low cost, availability, ease of use, large user base, support, example code, but mostly by the usage of free and/or open source hardware of software [8].

2.1. Raspberry Pi.
The Raspberry Pi is a $35 USD, credit-card sized computer that uses the Broadcom BCM2835 System-On-a-Chip. With a power consumption of 5.0W and the equivalent computing power of the original X-Box, it supports video outputs in RCA and HDMI formats, has an Ethernet port, two USB ports, up to 512MB of RAM, a GPIO port and solid state disk drive [9]. It runs on Linux operating systems, being Raspbian (based on Debian) the default option [9].

2.2. Arduino.
Arduino is a family of boards oriented to prototyping based on Open Source hardware and software [8]. It consist of a board with two microcontrollers. The principal microcontroller is where user’s programs are loaded and executed. Carries a set of digital and ADC inputs besides digital and PWM outputs. The secondary microcontroller acts as a Serial to USB converter that interacts with the computer. The IDE includes the means for writing programs (referred as sketches), compile and upload them to the board [10].

In this project we used an Arduino Uno board and four extra ATMEGA328P microcontrollers preloaded with the Arduino bootloader. Many authors have used Arduino to operate their robots, usually employing the whole board in their project [3-4, 6], having the inconveniences of false connections, inefficient use of the ports, space and cost increment I. The present robot design requires four microcontrollers, making the use of four Arduino boards ($25 USD each) unsuitable. Our solution was the deployment of the ATMEGA328P microcontrollers ($1.00 USD each) to a standalone PCBs using resonators II, surrounding electronics and power supply.

3. Architecture and modules
The robot developed in this project is shown in Figure 1, where the main components are highlighted. Figure 2 illustrates the main blocks of architecture of the robot. It shows the major components, modules and the means of communication between them. An ATMEGA328P microcontroller is used in every module except the Control module.

3.1. Frame and support
The robot is manufactured over a recovered electric scooter, where the front wheel was eliminated in favour of a welded Ackermann steering geometry [11], in which the two wheels came from an old bicycle. The main stand was removed and four pillars added to mount acrylic platforms to accommodate modules, components and cables.

3.2. Steering and traction.
The displacement of the robot is achieved through the management of a traction motor that powers a rear wheel. The original electric motor, gears, chain and rear wheel were kept, but a rotary encoder for motion measurement, adapted from a PS/2 Mouse board and acceded with a microcontroller using a library [12], was attached to the wheel. With the help of this encoder, the microcontroller monitors the number of turns that the traction wheel makes, becoming an odometer for the robot.

The motor is driven by an H-bridge using a PWM signal that is generated by means of a PID controller implemented from a library [13] also running in the microcontroller. Finally, direction control is fulfilled by governing the steering mechanism with a digital servomotor. Figure 3 illustrates this module.
The most expensive component in this module is the L298 H-bridge costing $2.45 USD. The price of manufacturing the PCB was $6 USD. The total amount was approximately $10 USD.

3.3. Power.
The main source of energy is a 4Ah, 12V, lead-acid battery. A microcontroller monitors the currents that are flowing through the modules, and control the charge rate of the battery with a PWM signal that manages a switching-mode power supply. The module also contains a Step-Down Power Converter that feeds all the electronic circuits in the robot, and a LM7805 circuit that powers servomotors. Every component is protected by a fuse to avoid over currents or short-circuit damage. Figure 4 shows the steps required to regulate and condition the power from the battery in order to be used by the different components in the robot.

The most expensive components in this module were two capacitors and a power transistor ($2.00 USD each, used to build the switching-mode power supply), six fuse mounts ($0.50 USD each) and the Step-Down Power Converter ($1.55 USD). The price of the board was approximately $14 USD for a total cost of about $27 USD.

3.4. Sensors.
In order of the robot to navigate and monitor its internal conditions, it needs to obtain physical information of its environment via sensors. Some sensors need to be implemented in a board with a microcontroller in order for their output, which can be analog (as the temperature sensor) or in the form of an I2C bus (like the IMU and the Magnetometer), to be read and adapted to a format the control module can understand.

Sensors implemented in a board with an ATMEGA328P microcontroller are: four Ultrasonic Range sensors, a Temperature sensor, an Inertial Measurement Unit (IMU; combines Accelerometer and Gyroscope), and a Magnetometer. Figure 5 illustrates the sensors module, its components and interfaces. The total cost of the sensors is about $11 USD, plus $1 USD for the microcontroller and $3 USD for the board. Total is about $15 USD.

Other sensors are connected directly to the Raspberry Pi via an USB cable. Sensors connected directly to the Raspberry Pi are: two video cameras ($5 USD each) and a GPS module ($20 USD).

3.5. Communication.
A microcontroller is used to interact with the Movement, Power Source and Sensors modules. This microcontroller acts as a "network switch", receiving packages of data and retransmitting them to.
the adequate port, from the modules to the computer and vice versa. The price of this module is very low (approx. $3 USD), as only a microcontroller, terminals and a very small PCB are needed.

3.6. Control.
The control module is composed of the Raspberry Pi and the devices directly connected to it. The computer seize data from the modules, the GPS and the cameras; processes and stores information, send orders to the microcontrollers and receive instructions over the internet through the Wi-Fi module. The serial port information is analysed and catalogued by Python scripts. Data from the cameras is processed with the OpenCV library and the GPS data is shown in an interactive map. The robot can stream live video, and can be controlled with an X-Box controller connected to a remote computer.

The most expensive components are: the Raspberry Pi ($35 USD) and the Wi-Fi module ($14 USD). Total price is around $50 USD.

4. Results and future work
The robot has been completely assembled and the different modules were independently tested with satisfactory results. All the sensors and actuators are working properly within their intrinsic error and the computer communicates effectively with the operator. What follows is the theoretical mode of operation for the simple case of sending the robot to a specific point on the planet.

An automatic-travel algorithm is yet to be implemented. It would use information obtained from GPS to know a very good approximate of its position at a fixed moment, and estimate from there by using the odometer, accelerometer, gyroscope and magnetometer.

Obstacle avoidance and path finder algorithms will obtain their data from GPS, ultrasonic range sensors and stereo reconstruction from the cameras. Obstacle avoidance will have precedence over path finder, and top speed will be limited in unknown environments. Stereo reconstruction will focus in mapping the floor in front of the robot in furthearance of evading holes and other obstacles.
5. Conclusion

Using recycled parts from an electric scooter and a bicycle, microcontrollers and a computer, the full system for an exploration terrestrial robot for educational purposes was constructed. The total cost of the bought components is less than $200 USD, whereas many parts were already available (scooter, battery, mouse, cameras, Arduino) so the total price starting from zero could raise up to $300 USD. Even if this is the case, the experience of:

- gathering information from various sources to implement different components and sensors;
- application of theoretical physics, mathematics and statistics for movement control, sensor output interpretation and calculations;
- development of microcontroller and computer software in several programming languages;
- design and implementation of communications schemes, mechanical parts, electric interfaces, electronic circuits and PCBs;
- putting it all together to accomplish the building of a robot; leaves more to the student than just experimenting with a pre-assembled kit.

Notes

I Nevertheless, one can state that Arduino is a single electronic board which provides convenient communication, ways of programming, surrounding electronics and electrical terminals for inputs and outputs, that make it ideal as a development platform.

II The ATMEGA328P can work at several frequencies, but 16MHz is what timing functions are optimized for in the Arduino Uno. This means that if other frequency is used, functions as serial communication will not work unless the code for them is compensated for the frequency change.

III This sensor outputs 10mV/K and can be read with the ADC in the ATMEGA328P microcontroller with a precision of ±0.45K.

IV The ATMEGA328P microcontroller can connect to an I^2C bus with the use of the wire library.

V The Digital Motion Processor in the IMU is not very easy to use, and is still not completely open. The maker of the sensor is not clear on how to program it, and its use is almost not documented at all. Up to now, the only available source is a compiled Demo Code that Jeff Rowberg managed to reverse-engineer to create a very extensive library for the use of this sensor [14].

VI The Raspberry Pi computer has only two USB ports. In order to accommodate all this devices, an externally-powered USB HUB had to be used. It is recommended to be externally-powered for the current of the connected devices don’t go through the Raspberry Pi.

References

[1] The LEGO Group. “LEGO MINDSTORMS EV3 LEGO Shop”. Accessed 07-Jun-2014. 2013. URL: "http://shop.lego.com/en-US/LEGO-MINDSTORMS-EV3-31313”.

[2] RobotShop. “Festo Robotino Mobile Robotic Development Platform”. Accessed 07-Jun-2014. 2013. URL: "http://www.robotshop.com/ca/en/festo-robotino-mobile-robotic.html”.

[3] Luiz A. Junior et al. “A Low-Cost and Simple Arduino-Based Educational Robotics Kit”. In: Cyber Journals: Multidisciplinary Journals in Science and Technology, Journal of Selected Areas in Robotics and Control (JSRC), December edition, Volume 3, Issue 12 (2013).

[4] Wan Muhamad Hanif Wan Kadir, Reza Ezuan Samin, and Babul Salam Kader Ibrahim. “Internet Controlled Robotic Arm”. In: International Symposium on Robotics and Intelligent Sensors (IRIS 2012) (2012), pp. 1061-1071.

[5] Ricardo Neves and Aníbal C. Matos. “Raspberry Pi based stereo vision for small size ASVs”. In: Oceans – San Diego. (2013), pp. 1-6.

[6] Wojciech Maślak and Bohdan S. Butkiewicz. “Autonomous vehicle with fuzzy control”. In:
[7] Kteam. “Kilobot”. Accessed 11-Jun-2014. 2014. URL: "http://www.k-team.com/mobile-robotics-products/kilobot".

[8] Joshua M. Pearce. “Building Research Equipment with Free, Open-Source Hardware”. In: Science, 14 September 2012: Vol. 337 no. 6100 pp. 1303-1304.

[9] The Raspberry Pi Foundation. “FAQs – Raspberry Pi”. Accessed 27-May-2014. 2014. URL: "http://www.raspberrypi.org/help/faq/".

[10] Arduino SA. “Arduino”. Accessed 16-May-2014. 2014. URL: "http://arduino.cc/".

[11] Wikipedia contributors. “Ackermann steering geometry, Wikipedia, The Free Encyclopedia”. Accessed 07-Jun-2014. 2014. URL: "http://en.wikipedia.org/wiki/Ackermann_steering_geometry".

[12] Chris J. Kiek. “PS2 mouse interface for Arduino”. Accessed 02-May-2014. 2008. URL: "http://playground.arduino.cc/ComponentLib/Ps2mouse".

[13] Brett Beauregard. “Arduino PID Library”. Accessed 28-May-2014. 2008. URL: "http://playground.arduino.cc/Code/PIDLibrary".

[14] Jeff Rowberg. “I2Cdevlib. MPU-6050 6-axis accelerometer/gyroscope”. Accessed 28-May-2014. 2013. URL: "http://www.i2cdevlib.com/devices/mpu6050#source".