Applied Physics Projects Using the Arduino Platform

Marin Oprea, Cristina Miron

Faculty of Physics, University of Bucharest, Bucharest-Magurele, Romania
E-mail: opreamarin2007@yahoo.com

Abstract

The element of novelty and innovation is an extremely important factor for any educational endeavour. In this respect, the Arduino open-source platform could be considered an extremely useful resource for the setup of innovative applications or projects in the physics class. Designed as a small-sized microcontroller, it is capable to collect data from the surrounding environment through a series of sensors, such as distance, pressure, vibration, temperature, humidity, acceleration, force, light, PIR sensors etc. The present article focuses on the creative uses of the Arduino Uno platform in the physics class. This platform is based on the microcontroller ATM328 and has an architecture compatible with a wide range of tasks in automation: from the control of the servomotors inside a robot to the analysis of environmental parameters through a weather station. With a minimal investment and a substantial economy in terms of space, time and materials needed for certain projects, the teacher and the students are able to perform a series of exciting applications in the physics class. Moreover, through the use of this Arduino platform, the students are invited to improve their computing skills, as basic notions of C/C++ programming are needed to operate the platform. The class can thus be engaged in attractive projects of applied physics, because Arduino provides a nearly limitless array of applications which create a fun learning atmosphere.

Keywords: Arduino, microcontroler, sensors, physics projects

1 Introduction

The advancement of the scientifical research in the field of micro and nanotechnologies has led to the worldwide spread of processors and microcontrolers. The microcontroler operated platforms have revolutionized the way we design and construct automatic systems in the industrial and domestic sector. An example of a microcontroler-based platform is Arduino. This platform can receive data from the environment through an extended use of sensors, such as temperature, humidity, atmospheric pressure, distance, acceleration, rotation, light sensors etc. (*see* Figure 1).

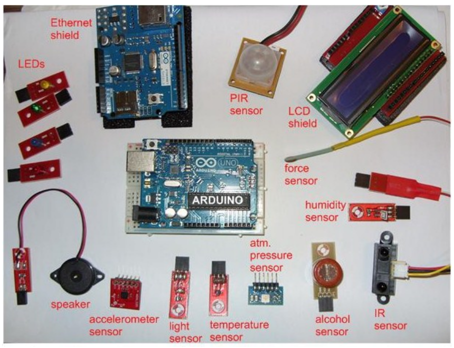

Figure 1. Arduino platform with sensors and other devices

The Arduino board can communicate data through the Internet, using specific devices: Ethernet shields, WiFi shields, GSM devices and can connect to a computer or a mobile phone through a Bluetooth connector. The platform can control with a high degree of accuracy the DC electric motors used in robotics, the servomotors and the stepper motors used in the industrial sector. Some authors have already started reporting about their experience with Arduino-based projects (Margolis, 2013; Ribaric and Younker, 2013), but literature in this field is still in its infancy.

At the same time, Arduino is a powerful didactic instrument that creatively appeals to the student's knowledge in the field of physics, informatics, electronics and technology. Using this platform in the science classes gives the students skills that can become important premises for choosing a profession in the robot design and programming sector.

In this paper we described just a few of the didactically relevant projects that we developed using Arduino: a flower watering alarm system triggered by a soil moisture sensor, a weather station which can send data on GoogleDocs and a robot-car using a motor driver.

2 Flower watering alarm system triggered by a soil moisture sensor

The accomplishment of this project was quite easy. We used an Arduino board, a moisture sensor made of two metallic electrodes introduced into a flower pot and an optical alarm device: a LED which turned on every time the plants needed water (*see* Figure 2).

The metallic electrodes made of copper (or stainless steel, such as the telescopic antenna segments) were placed in the flower pot at a distance of 1cm from each other. With the help of an ohmmeter we measured the electrical resistance between the two ground electrodes. One could notice that the value of the electrical resistance depended on the level of humidity in the soil: the wetter the soil got, the lower the resistance of the electrodes system.

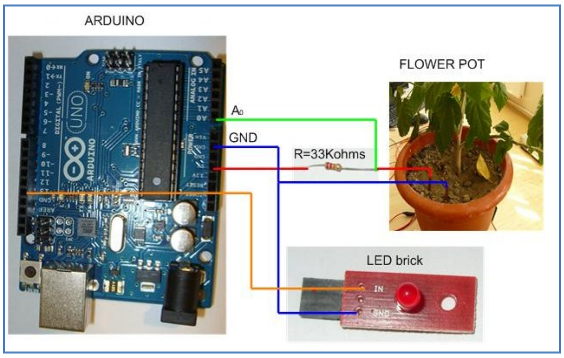

Figure 2. Experimental scheme

We performed resistance measurement tests using different flower pots. For the resistor R connected to the system we selected a medium value R=33kohm, between the value associated with the dry soil and the value measured for the humid soil.

By assembling the system Arduino-resistor-electrodes-flower pot we created a voltage divider. One can notice that the second resistor, other than the one placed in the scheme, is determined by the electrodes grounded in the flower pot. When the soil becomes more dry or more humid, the value of the resistance changes, which leads to a shift in the value read by Arduino.

The experimental measurements performed with the ohmmeter have marked out values of the soil resistance between 10 and 100kohm, depending on its degree of humidity. In such circumstances, we decided to use a resistor with the value of 33kohm for this project. In order to monitor the data in *Serial Monitor*, we used the following source code (see Figure 3).

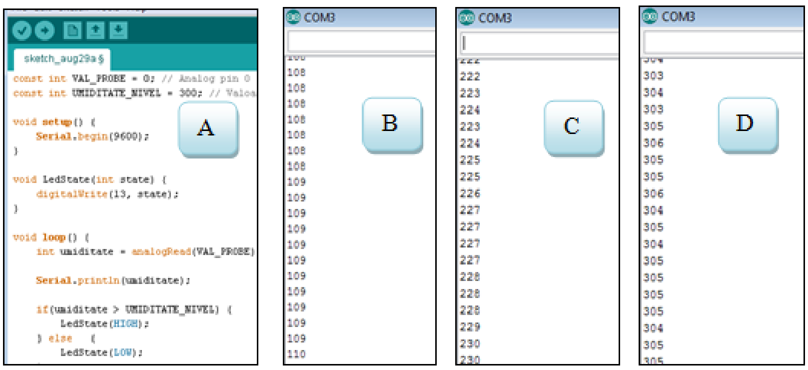

Figure 3. Source code (A) and data in *Serial Monitor* for:
humid soil values (B), threshold humidity values (C), dry soil values (D)

The threshold of humidity was established at the value of 230 units after repeated data monitoring tests from *Serial Monitor* were performed. Above this value the humidity level of the soil is reduced and the LED turns on and under this values the LED stays off.

The project can be perfected by adding an acoustic alarm instead of the optical one or by installing a GSM shield that enables the user to receive an alert either through SMS or through e-mail.

3 Weather station

Using the Arduino board, a BMP085 sensor and an Ethernet shield for Arduino we assembled a weather station for the collection of temperature and atmospheric pressure data which we stored online on a Google server as an Excel document. The BMP085 sensor is a high precision sensor capable of measuring atmospheric pressure and temperature. Given the fact that the atmospheric pressure varies with altitude, based on the measured values, the altitude can be calculated. The Ethernet shield was placed on top of the Arduino board, making sure that the pins would fit correctly into the board (*see* Figure 4).

With the help of the RJ45 cable, the shield was connected to a switch in the local network. After installing the Ethernet library which accepts the DHCP protocol, we stopped and reopened the developement environment in order to update the data. The following program was uploaded on Arduino (*see* Figure 5).

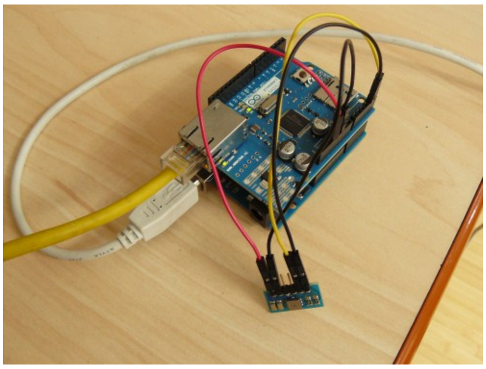

Figure 4. Arduino Shield Ethernet

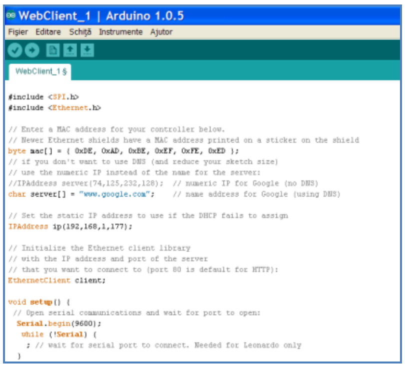

Figure 5. Arduino code – Weather station

In *Serial Monitor* we received a confirmation message for connecting the Arduino board to the Internet through this shield, which we verified by sending a ping to the google.com server (*see* Figure 6).

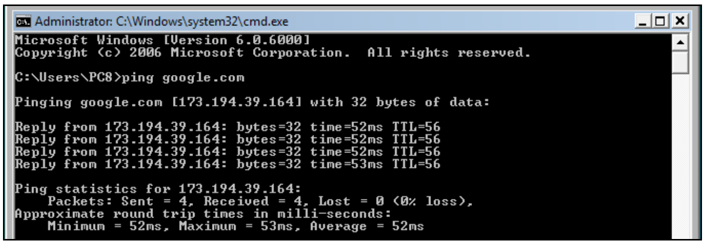

Figure 6. Pinging the Google server

Using the computer on which we installed the Arduino folder we accessed the address: http://www.docs.google.com. It's here that we created a spreadsheet form containing two text fields, one for temperature and one for atmospheric pressure (*see* Figure 7).

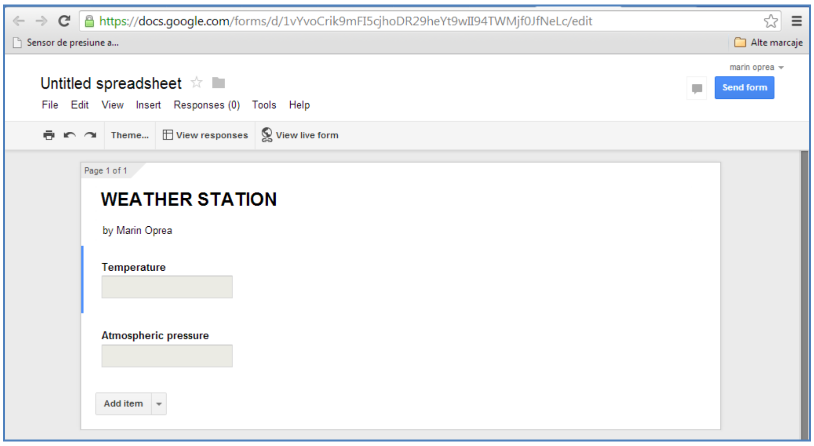

Figure 7. Spreadsheet created on GoogleDocs

From the menu *Form* we selected *Go to Live Form* in order to see the form created by GoogleDocs for the gathering of data. By analysing the source code of the page (View Page Source) we localized the *Form Key*, which is an unique series of characters associated with the form and the codes associated with the text fields which we introduced in the code lines used by Arduino for the Internet connection (*see* Figure 8).

The form can now receive the data offered by the BMP085 sensor, whose activity was monitored at the same time on *Serial Monitor*. Finally, the data, which was updated every 15 seconds, is to be found listed in the GoogleDocs form, where a user can monitor it in real time (*see* Figure 9).

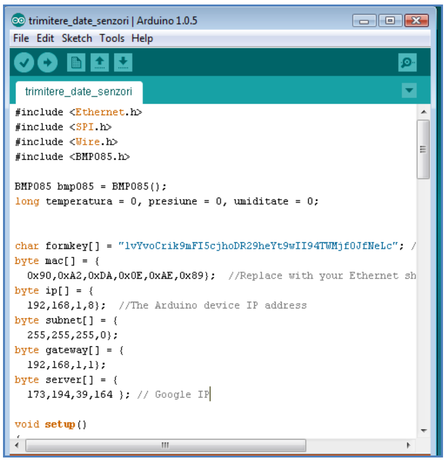

Figure 8. Weather station Arduino code

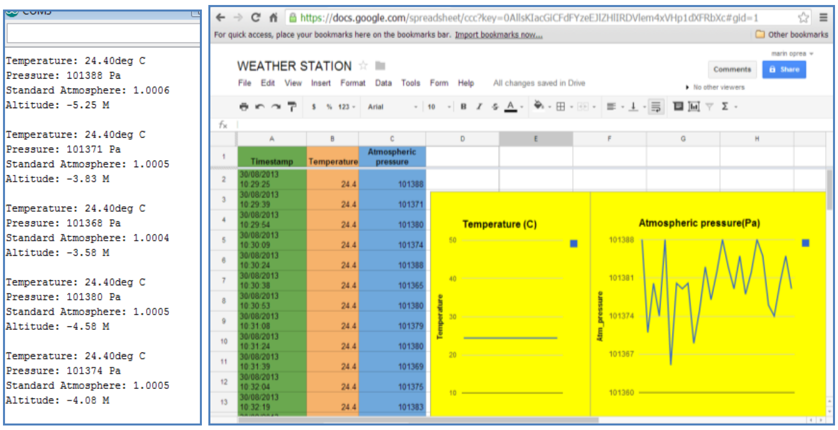

Figure 9. Data transferred on GoogleDocs

4 Robot-platform using Arduino and a motor driver

The assemblying of the robot was performed using a kit which included two DC current motors, two specially designed wheels, a chassis made of plexiglass and a set of connectors – bolts and nut bolts (*see* Figure 10).

Figure 10. Arduino kit for the construction of a robot

For powering up the robot we used a battery container that delivers a tension of 6V required by the motors. The free run speed of the motors is 85 rotations/minute and the values of the free run current amounts to 70mA. The stall current absorbed by the motors is 800mA and the stall torque that they generate is 5,4kgf * cm. The robot assembly procedure began by covering the wheels with the provided rubber bands and continued with the coupling of the wheels and the motors (*see* Figure 11).

Figure 11. Assemblying the wheels

Securing the assembly wheels-motors on the chasis was done with the help of some plexiglass plates (*see* Figure 12).

Figure 12. Securing the assembly wheels-motors on the chassis

After installing the metallic support wheel, we placed the Arduino board on the chassis, on top of which we inserted the driver motor shield L298N. The motors were connected to the pins named Motor1 and Motor2, while the powering of the motors was done through the pins VIN and GND.

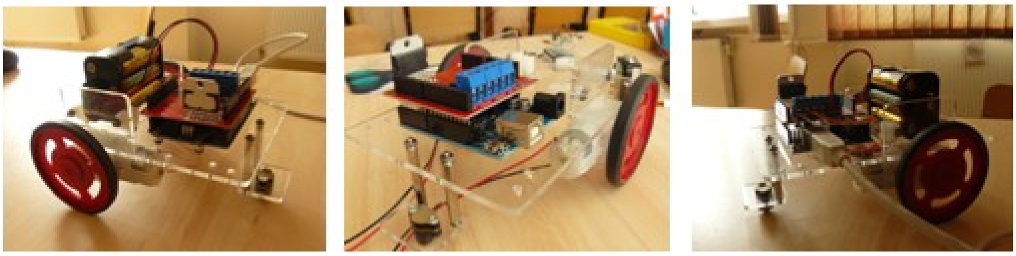

Figure 13. Assembled Arduino robot

We performed several tests for the movement of the assembled robot (*see* Figure 13), modifying in the source code of the motor drivers the values of the rectilinear displacement times between two trajectory changes (*see* Figure 14).

```
go(255,-255);
delay(1000);
go(-255,-255);
delay(1000);
go(-255,255);
delay(1000);
go(255,255);
delay(1000);
```

```
go(255,-255);
delay(1000);
go(-255,-255);
delay(5000);
go(-255,255);
delay(1000);
go(255,255);
delay(5000);
```

Figure 14. Altering the Arduino code

The next step in developing this platform resides in the automatic control of the displacement direction on an obstacle course using proximity sensors, either ultrasonic (Parallax) or infrared (Sharp) (*see* Figure 15).

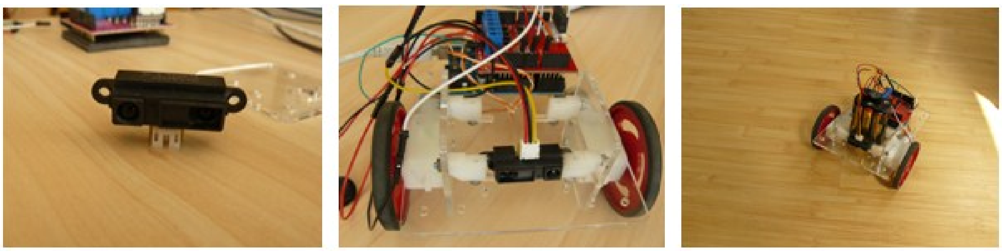

Figure 15. Testing the Sharp sensor

At the same time, the platform can be controlled through Bluetooth with the help of a mobile phone using application Amarino, which can be downloaded for free on an Adroid smartphone.

5 Conclusions

The projects of applied physics using the Arduino platform have boosted the students' motivation for the study of the scientifical and technological subjects. In the field of physics, we noticed an important increase in the understanding of the main physical quantities – distance, force, temperature, pressure, acceleration etc. – through the study of the sensors' functioning manner. In the field of technology, the students have developed working abilities with motors, electric devices and mechanical assemblies. In the field of electronics, they understood the functioning principles of simple electric devices – resistors, capacitors, LEDs, transistors, but also of integrated circuits with specific functions used in the constructed assembly. With respect to informatics, the necessary C/C++ notions required to program the Arduino board were studied, together with terms from the field of information and communication technology, which were understood by studying the Intenet connection protocols of the platform.

During the course of these projects, the students showed an innovative, creative and collaborative spirit, along with an ongoing commitment in the development and accomplishment of the projects. The outcomes were reflected in the improvement of their school performances in all scientific subjects. In conclusion, we are convinced that such projects give physics education a greater quality and efficiency.

References

Margolis, M. (2013): *Make an Arduino-Controlled Robot*, Sebastopol: O'Reilly, Inc.

Ribaric, T. and Younker, J. (2013): Arduino-enabled Patron Interaction Counting, *Code4Lib Journal*, 20, available at: http://journal.code4lib.org/articles/8200 [20.08.2013].

Internet Sources:

http://www.instructables.com/id/Arduino-Projects/
https://learn.sparkfun.com/tutorials
http://playground.arduino.cc/projects/ideas
http://quarkstream.wordpress.com/lessons-in-arduino/
http://www.buildcircuit.com/how-to-make-an-easy-robot-using-arduino-android-amarino-ardumoto-and-magician-chassis/
http://robofun.ro/