ANNALS of Faculty Engineering Hunedoara
– International Journal of Engineering
Tome XIII [2015] – Fascicule 4 [November]
ISSN: 1584-2673 [CD-Rom; online]
a free-access multidisciplinary publication of the Faculty of Engineering Hunedoara

[1.] Gelu-Ovidiu TIRIAN

MAZE SOLVING MOBILE ROBOT

[1.] University Politehnica Timişoara, Faculty of Engineering Hunedoara, ROMANIA

Abstract: The paper introduces the concept of an autonomous mobile robot capable of solving a labyrinth. The mobile robot is a combination of devices, servomotors and sensors, under a control system that operates in real space, and which is designed to perform the operation of crossing a labyrinth, irrespective of its shape, by processing the information received from the sensors. The robot was designed by means of the Fritzing software, while the programming was done in Arduino. In designing the robot, the criteria under consideration were the regulations of the maze-type robot competitions, so that it can take part in such competitions.
Keywords: robot, labyrinth, the development board, sensors, programming language

1. INTRODUCTION

The mobile robot is a complex system capable of performing various activities in real world specific situations. It is a combination of devices equipped with servomotors and sensors under the control of a hierarchical computing system operating in real space, influenced by a series of physical properties (for instance, gravitation, which influences the movement of all robots moving on the ground) and which has to plan the movements so as the robot can perform a task according both to the initial state of the system and to the actual information related to the work environment. The success in performing such tasks depends both on the knowledge the robot has on the initial configuration of the work space and on the information obtained during its evolution [1] [2] [8].

2. ROBOT CONSTRUCTION

Herein after is given the description of the robot and its main hardware components. This is an autonomous mobile robot that does not need its remote control by means of various applications that can be installed on mobile phones with operation system.

The feeding is done by a set of 6 batteries mounted in a special case and connected to the GND (ground) and the 5V (Volt) pins [12] [13].

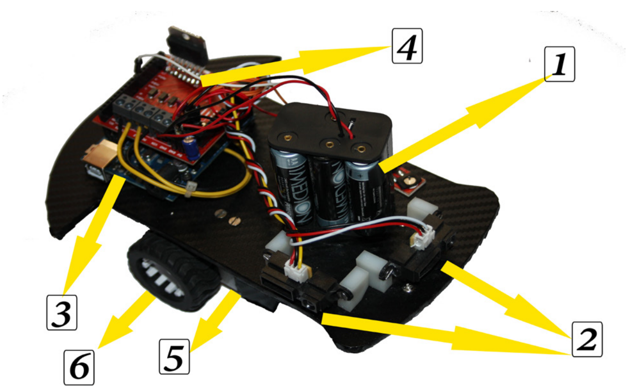

Figure 1. Overall look of the robot

The main hardware components used in building this robot are: 1 - Case 6 R6 Batteries, 2 - distance sensor, 3 - Arduino Uno board, 4 - L298N shield; 5 - c.c. motors, 6 – running wheels. This mobile system is built first by assembling the pieces, and feeding the motor driver and the microcontroller by means of 6 R6 batteries. The platform allowing the assembling of the devices is a plastic plate, highly efficient in mounting the hardware devices of the mobile system. The second step is to connect the Arduino board to a computer by means of a USB cable, in order to transfer the data. Finally, the Arduino programming software is used; its programming editor is offered by the site www.arduino.cc. The programming language is a usual one, similar to C++[5].

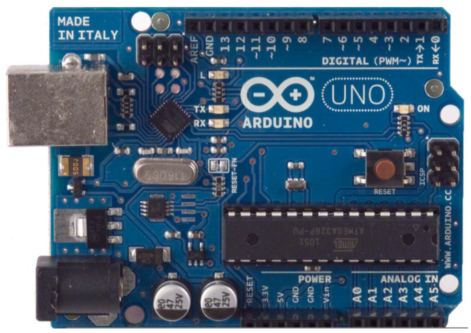

Figure 2. The ARDUINO microcontroller

2.1. ARDUINO UNO

The robot needs a brain. The best option in this sense is the Arduino UNO board. Arduino UNO is an open-source processing platform, based on flexible and user-friendly software and hardware. It consists of a small platform (6.8 cm / 5.3 cm – in the most widely used variant) built around a signal processor and capable of receiving data from the environment through a series of sensors and of performing actions upon the environment by means of lights, motors, servomotors and other types of mechanical devices. The processor is capable of running a code written in a programming language that is very similar to C++[10] [11].

2.2. The distance sensors

The Sharp distance sensor is a component that can be used in order to measure the distance to various objects in the surrounding environment. This type of sensors is the most suitable for this kind of robot, as it has a very fast response time. The ultrasound sensors are not efficient, as their response is relatively slow.

Sharp distance sensors are available in three variants, according to the zone where they are efficient. There are sensors that are efficient between 3 and 40 cm, between 10 and 80 cm, and between 15 and 150 cm. These sensors cannot be used out of their active zone.

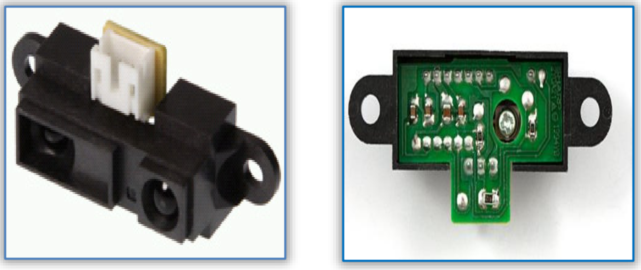

Figure 3. SHARP distance sensor

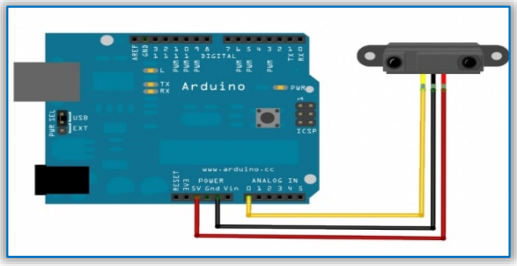

Figure 4. The connection to the Arduino board

The Sharp GP2D12 infrared sensor has the following technical characteristics: it uses the principle of infrared light reflection in order to measure distance, its action range being between 10 and 80 cm; the sensor is fed by a 4,5 to 5 V source, with a current of 33 mA, the output voltage ranging between 0.4-2.4 V when fed by a 5 V source [9].

Its interface has 3 wires: feeding (Vcc)- red wire, ground (GND)- black wire and the output voltage (Vout)- yellow wire.

The principle underlying the gauging method used by the Sharp sensor consists in a triangle formed between the infrared Emitter, the reflectionpoint and the infrared detector. Thus, the emitter emits a pulse of infrared light [3]. The light travels until it meets an obstacle, or, if not, it continues its movement. In case it hits no obstacle, the light will not be reflected and no object will be detected. If the light is reflected by an object, it will return to the detector, creating a triangle among the emitter, the reflection point and the detector, like in the next figure 5.

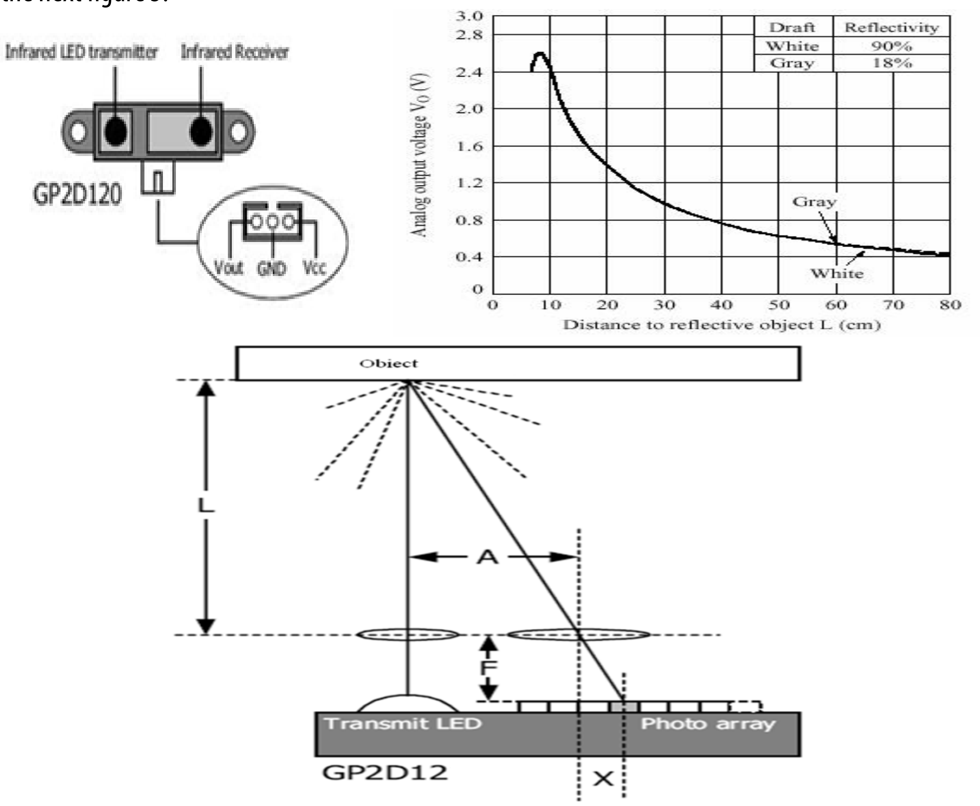

Figure 5. The gauging principle underlying the SHARP distance sensor

2.3. Direct current motors

This compact gear, (low DC current brush motor with a reduction gear of 120:1) is most suitable for small robots. At 4,5 V, it has a free-run speed of 120 rpm and a standard torque of about 20 oz-in; the D-shaped axle has a diameter of 3 mm. This axle is perpendicular to the motor shaft [6].

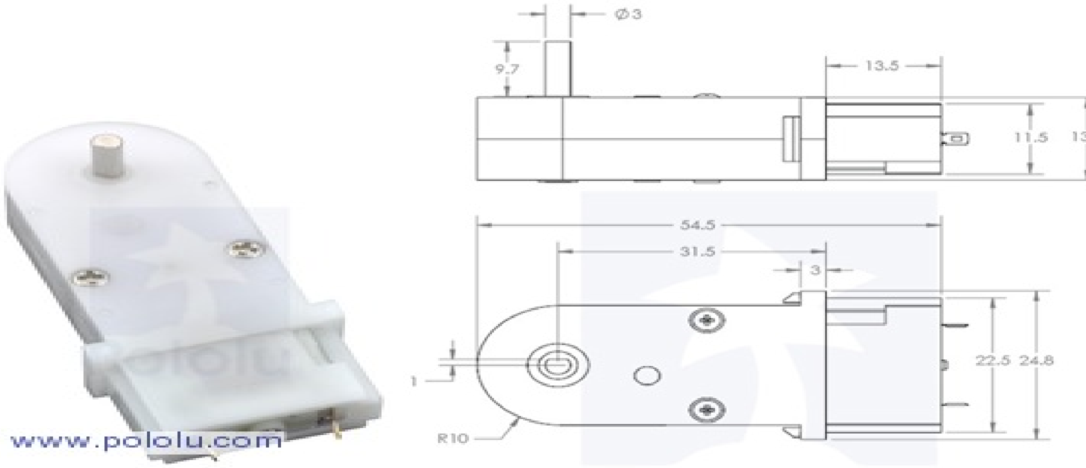

Figure 6. DC motor

2.4. L298N motor driver

Arduino is a very capable brain for this type of robots, but, as any brain it can only perform precise operations. Arduino has not been designed to put out high power for motors, it only puts out precise control signals. The motor driver is connected directly to the energy source (battery) and it controls the motors according to the control signals received from Arduino.

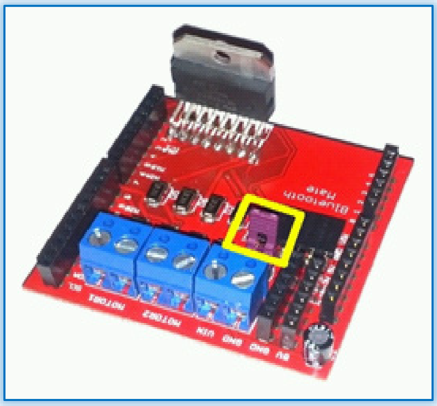

Figure 7. Motor driver

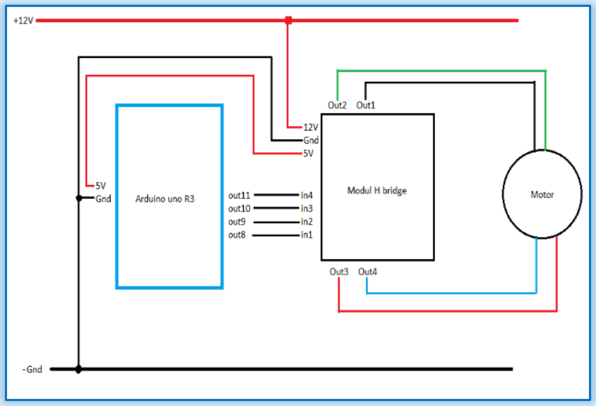

Figure 8. L298N to Arduino connection diagram

This motor driver is based on the L298N circuit, and it can control 2 DC motors, the maximum current being 2 amperes. The driver is completely assembled as an Arduino shield, which simplifies its use. The connection to the Arduino is done by coupling the shield board to the Arduino board and plugging the VIN and GND pins to the motor feeder. The PWM pins, controlling the L298 drive are: 3, 5, 6 and 9 (see also the testing program given herein after). The two motors are connected by the screw pins marked "MOTOR1" and "MOTOR2". For this robot, the power continuously dissipated must be rather high and this is why a radiator was used [12] [13].

Further on is given the way of connecting the components that form the autonomous system that solves the labyrinth.

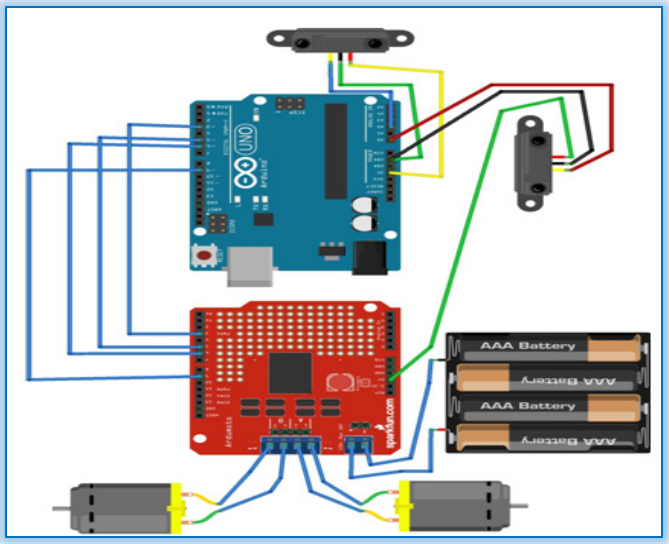

Figure 9. Robot component connection diagram

ISSN: 1584-2673 [CD-ROM]; ISSN: 1584-2673 [online]

3. THE ROBOT FUNCTIONING PRINCIPLE

The robot advances the moment the distance read by the infrared sensor ranges between 12 and 14 centimeters. Certain disturbances may cause the robot to slightly turn right or left. This is why the position has to be corrected, but also slightly, in order to gain time in running the course. The correction is done to the left when the distance between the right wall and the robot is between 10 and 12 centimeters and to the right when the distance between the right wall and the robot is between 14 and 16 centimeters. In the worst scenario, the robot might not perform the left correction accordingly and the distance between the robot and the right wall could get to be too small (<10 centimeters in this case), risking to touch the wall, which is not desirable. In order to avoid this situation, the robot will turn sharp to the left, until the distance to the right wall is a little higher than 10 centimeters, after which it will do another correction to the left.

At the moment the robot can turn right, the distance between the right wall and the robot is more than 16 centimeters. This is why the robot does not turn sharp right, which might block the robot in the wall corner. The right turn is achieved by revolving the left wheel faster than the right one, thus the robot turning in a semi-circle to the right.

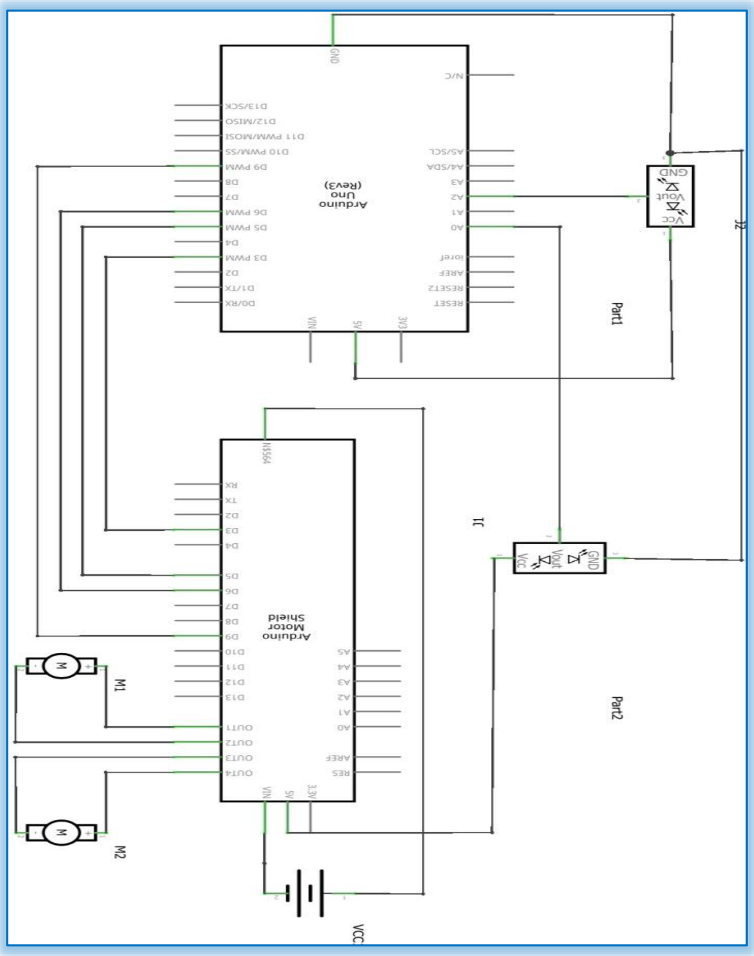

Figure 10. Connection diagram

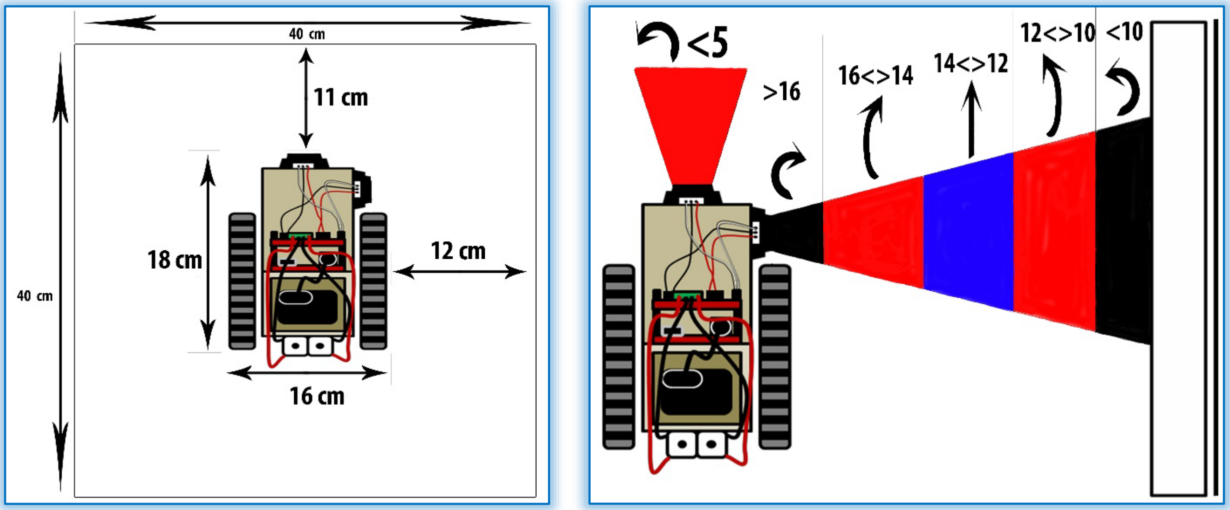

Figure 11. The robot functioning principle

All the movements analyzed so far are done only if the distance to the front wall is above 5 centimeters. If the robot reaches the front wall and the distance is below 5 centimeters, the robot turns left. The left turn is done on the spot, by turning the right wheel forewards and the left wheel backwards at the same speed. Thus, the robot can turn 360 degrees with no problem, due to its size, which is suitable for the labyrinth (40 centimeters / 40 centimeters).

4. SOFTWARE IMPLEMENTATION

The programming language used for a correct functioning is similar to C++ and it is called Arduino.

The window shown below is a programming editor dedicated to Arduino boards. One has to enter the Tools menu and choose the board in use, as well as the port selected by the computer.

Any program line written has to be checked by pressing the „OK" button, which checks the errors that might apear and after the message " Done compiling" is displayed, the program can be run by pressing the „UPLOAD" button, which sends by the COM3 port, the instructions to the microcontroller, where they are further processed and sent to the output pins.

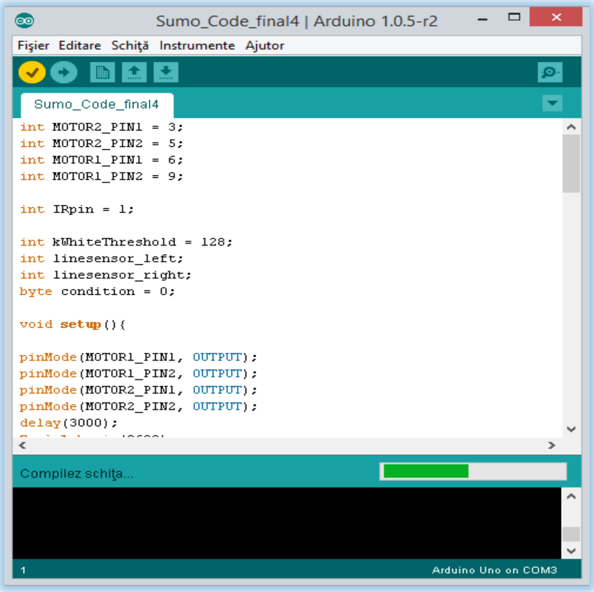

Figure 12. Program compiling

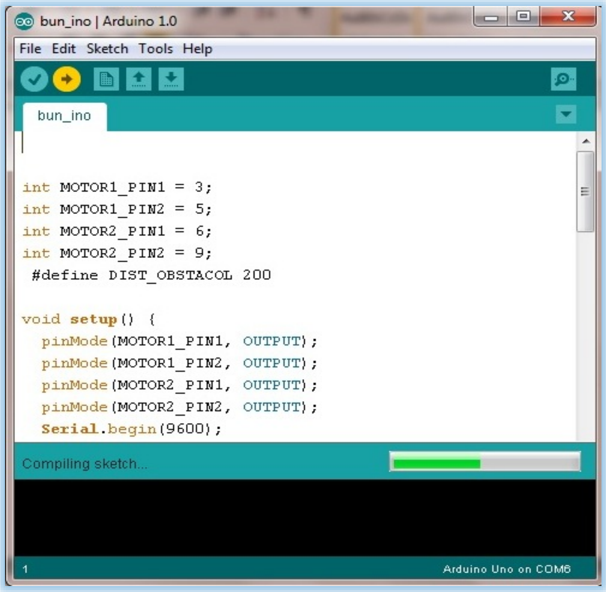

Figure 13. Program uploading

Further on, some print screens are given, showing the code lines of the program.

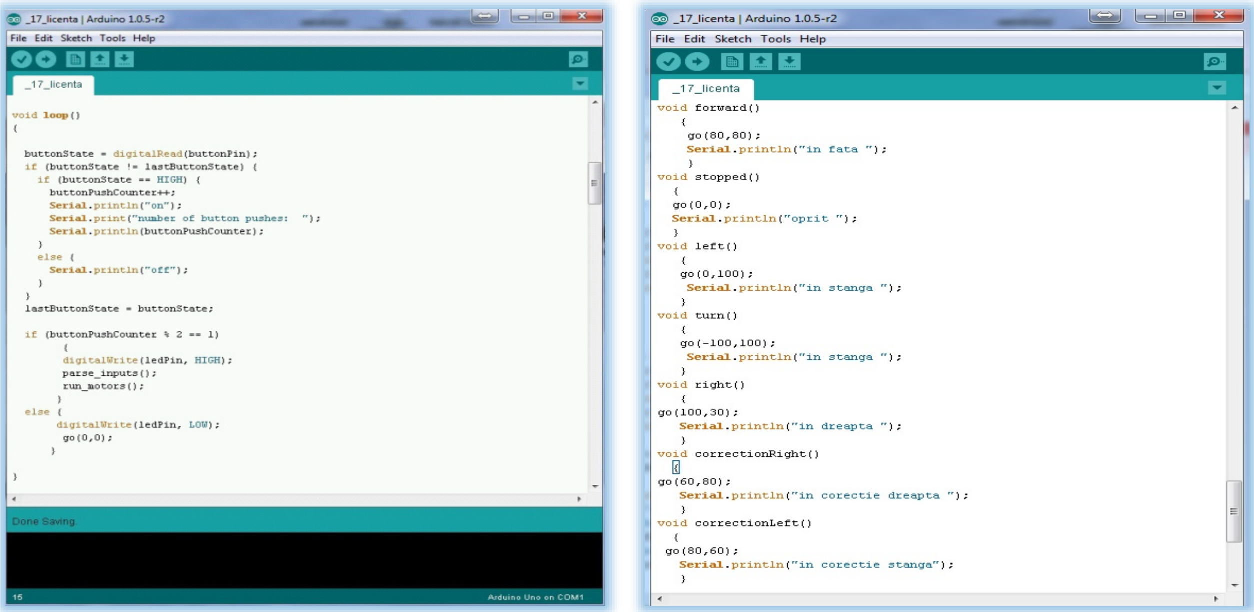

Figure 14. Program structure

The algorithm of the robot is based on the following basic rules: if the robot can turn right, it turns right; if the robot can go ahead, it goes ahead; if the robot can turn left, it turns left. The decisions are made in the order they have been enumerated.

5. CONCLUSIONS

The paper presented an autonomous mobile robot capable of running along a labyrinth. It was designed using a Fritzing software and the programming was done in Arduino language.

This robot benefits from a series of important advantages and it can be used as didactical equipment, or adapted for its use in various domains, such as: the exploration of narrow or dangerous spaces, military or spatial applications, etc. The mobile robot is a combination of devices, servomotors and sensors under a control system, operating in real space; it can plan its moves so as to be capable of running along a labyrinth, according to its shape and to the information received from the sensors.

References

[1.] Dumitriu A., Bucşan C., Demian T., “Sisteme senzoriale pentru roboţi”, Editura MEDRO, Bucureşti, 1996.

[2.] Ivanescu Mihai, Robotica - Editura Universitaria, 1995

[3.] Ispas Victor, Roboți pentru aplicații speciale - Editura Dacia, 2001

[4.] Iovine J., „Robots, androids and animatrons”, McGraw-Hill, USA, 2002.

[5.] Laumond J.P., “Robot motion planning and control”, Springer-Verlag, London, U.K., 1998.

[6.] PEDRE S., P DE CRISTFORIS, CACCAVELLI J., “A MOBILE MINI ROBOT ARCHITECTURE FOR RESEARCH, EDUCATION AND POPULARIZATION OF SCIENCE”, 2010 - ROBOTICA.DC.UBA.AR

[7.] Tirian G. O., “ Robotică”, Editura Politehnica Timisoara, 2013.

[8.] Tirian G. O., “Automation of a warehouse by means of a robotic arm” - Annals of Faculty Engineering Hundeoara – International Journal of Engineering Tome XI, Fascicule 4, 2013.

[9.] Tirian G. O., AM. Nitescu, C.Chioncel, “The design and construction of an autonomous mobile mini-sumo robot” – ACTA TECHNICA CORVINIENSIS– Bulletin of Engineering Tome VIII, Fascicule 2, 2015.

[10.] Sharp GP2Y0A21YK0F Distance Measuring Sensor datasheet,http://www.pololu.com/le/download/gp2y0a21yk0f.pdf?

[11.] http://www.roboter-info.de/index_en.htm

[12.] http://www.robofun.ro

[13.] http://www.sparkfun.com

ANNALS of Faculty Engineering Hunedoara – International Journal of Engineering