Revista Perspectiva em Gestão, Educação & Tecnologia, v.3, n.6, julho-dezembro /2014

AUTODRIVE PROJECT

Kleber Moreti de Camargo

kleber.camargo@fatec.sp.gov.br

Rodrigo Diniz

rodrigo.diniz@fatec.sp.gov.br

FATEC Itapetininga

TRANSLATION: Gilcéia Goularte de Oliveira Garcia

FATEC Itapetininga

ABSTRACT: This article presents the AutoDrive project as a way to make a safer ground transportation. This is an automated low cost solution that aims to reduce or even eliminate collision accident in traffic. Therefore, it employs the open source Arduino technology, coupled with sensors, in recognition of obstacles and control of a vehicle on a reduced scale, thus aiming to demonstrate its applicability. Hardware constraints directly impacted the results, however, these have demonstrated the potential that the project has, even with these restrictions. With appropriate changes, the AutoDrive project can indeed assist and help in reducing accidents.

Keywords: Arduino. Microcontroller. Autopilot.

1 INTRODUCTION

With the steady increase in the number of accidents throughout the country and an increasingly chaotic traffic in large cities, the need to create mechanisms that can reduce accidents and optimize traffic in major centers arises.

According to Sciesleski (1982), traffic accidents, either occur due to the lack of vehicles and roads maintenance, or they are caused by pedestrians and drivers, where human flaws stand out among other determining factors of accidents.

According to data from the National Department of Transit (Denatran), traffic accidents by collision or impact represent more than half of the accidents with victims in Brazil (BRASIL, s.d.).

There are, currently, only high cost vehicles, which have solutions that increase safety during the transportation by motor vehicles, as for example, proximity sensors in the vehicle ahead that allow a vehicle to slow down while maintaining a safe distance, and also, sensors that verify the existence of another vehicle in the so-called blind spot.

On this issue, the project development objective of AutoDrive presents a low-cost device able to drive (control) a vehicle, so, as to avoid collisions and even drive it in a certain way without the need of human intervention, keeping compatible speed on the path to be taken.

The implementation of such a device in a vehicle must be able to reduce the occurrence of accidents by collision.

Brief History

With recent onset, in 2005, Arduino technology emerged at the Design Institute in Italy by Professor Massimo Banzi, who sought to make it easier for his design students to learn technology. In discussion with David Cuartielles, researcher at the University of Malmo, in Sweden, who sought the same solution, Arduino came up.

As existing products at the time were expensive and difficult to handle, Banzi and Cuartiellis eventually developed a microcontroller that could be used with some ease by their students. Thus, David Cuartielles designed the board and a pupil of Massimo, programmed the software that could control the board. Gian Martino was an engineer hired to help students, who turned out to produce an initial run of 200 boards.

These boards were sold in the form of kits so that each student could develop their own project. Their popularity grew rapidly, as a result of their easy handling and their low cost. The Arduino name was reference to a bar where the pupils and the teachers used to go to at the time. There are, currently, a number of models of Arduino that eventually had their initial technology improved (EVAN, et al, 2013).

2 METHODOLOGY

For the project, implementation AutoDrive technology called Arduino was used, an open source platform built on a single board (Figure 1), able to interact with the outside world by means of several sensors. This platform offers a software suitable for the creation of source code that must be written in language similar to the C programming language, and allows its recording on Arduino microcontroller board by means of a USB cable (ARDUINO, 2005a).

Figure 1 – Arduino Uno Board.

Source:http://arduino.cc/en/Main/ArduinoBoardUno

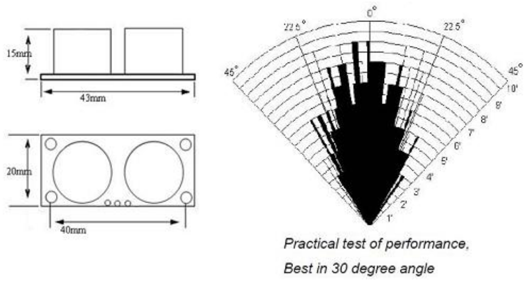
A remote control car in 1:18 scale was used, its control board was extracted and its function coming under the responsibility of Arduino.

To identify obstacles 4 ultrasonic distance HC-SR04 sensors were used (Figure 2 and Figure 3) integrated with Arduino, also, for engine control, a CI L298n was used, capable of controlling the two engines of the car (1 to control the direction of the car simulating the steering wheel and 1 for car speed control), as described in Arduino (2005b).

Figure 2 – Sensor HC-SR04.

Source: http://blog.oscarliang.net/how-to-use-ultra-sonic-sensor-arduino-hc-sr04

Figure 3 – Angle of sensor operation HC-SR04.



Source: http://www.elecrow.com/download/HC_SR04%20Datasheet.pdf

The control software was developed in C language, in its own IDE offered by Arduino, and making use of the "Ultrasonic" library. It was necessary to make a small change in this library in regard to the waiting time of the ultrasonic signal return. This change was necessary because the library, in its original code, determines that the sender pole of the

HC-SR04 sensor to only repeat the procedure for issuing after receiving, by the pole receiver, the last signal issued. This behavior was causing lockups in code developed for control of the vehicle. Since the implementation of the programme was conditional to the release of processing by the "Ultrasonic" library in every iteration of the process of issuing and receiving of ultrasonic signal, it prevented the continuation of the normal flow of the algorithm until its completion, and as consequence, the vehicle would not make decisions to change directions, causing it to collide.

Figure 4 – Original code library "Ultrasonic".

```
duration = pulseIn(Echo_pin,HIGH);
```

Source: Drawn by author.

Figure 5 – "Ultrasonic" library code changed.

```
duration = pulseIn(Echo_pin,HIGH,5000);
```

Source: Drawn by author.

The change required was to establish a maximum amount of time for which the library would wait the signal return. As soon as the time is over, the "new" library code disposes the need to wait for the last signal and sends another signal to a new attempt to get the current distance between the vehicle and the possible obstacles around it.

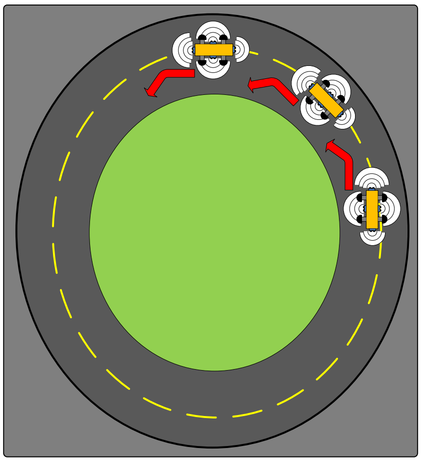
Fundamentally, the purpose of the control code was to provide the vehicle an algorithm capable of, by identifying obstacles, trajectory-altering decisions thus avoiding the risk of collisions, keeping the vehicle within a given route.

To demonstrate the ability of the project to avoid collisions, the prototype remote control was not used to guide it, as the demonstration would be hampered on its efficiency and effectiveness. Thus, we chose to allow the vehicle to take its own decisions of conduct within a predetermined path.

3 RESULTS AND DISCUSSIONS

The efficiency of integration of hardware and software developed has been tested by placing the scaled-down vehicle in an oval path as shown in Figure 6.

Figure 6-Oval Path.



Source: Drawn by author.

Along the path curbs were placed so that the vehicle would interpret them as walls, which defined the route to be fulfilled.

Ten tests were carried out, where in each of them the vehicle had its initial position on the path randomly determined.

It was found that in eight attempts, the vehicle completed seven laps without the occurrence of collisions. On the other two attempts, the vehicle was able to complete five laps without any incident. At the end of these, the vehicle ended up bumping into one of the curbs.

The overweight, from the installation of the necessary hardware for the implementation of the project, caused an inefficiency in the engine that controls the change of direction of the vehicle.

Another point to be noted is the fact that the Ultrasonic sensors were unable to perform correct readings at certain points of the route due to not being on a favorable angular position to the reflection process of the sent signals.

After the tests performed and the knowledge acquired throughout the process of construction and development of hardware and software used was possible to detect some restrictions that gave cause to collisions after five or seven laps around the course, they are:

a) Restrictions of the vehicle: the small-scale vehicle used, after being fitted with the necessary hardware (Arduino, protoboard, battery power supply, sensors, CI l298 and cables) had an overweight, which influenced the performance of vehicle engines negatively, mainly the engine responsible for changing the trajectory of the vehicle. This overweight dramatically reduced the efficiency of the front engine to perform full movements in change of direction to the right and left. To reduce this effect, adjustments were made in the source code of the project to force the wheel to the desired direction. Consisting in creating repetitive routines that repeatedly intersect command to change direction to the right or left

Revista Perspectiva em Gestão. Educacão & Tecnologia. v.3. n.6. julho-dezembro /2014

until it reaches the correct position. As a result, there has been an increase in the time required for the ideal positioning of the vehicle corresponding to its current state on the path.

b) Restrictions of the Ultrasonic sensors: these sensors feature as basic principle of operation a signal emission and the return reception of this same signal. HC- SR04 sensors have a standard emission and reception angle of 30º. Due to various facts like the steady sensor reading, the vehicle being in motion, and, as a result of this movement, the vehicle at various times keeps a different angle from 90° in relation to the side curbs, the result was that by reflection, signals sent constantly ended up lost, not returning any information to control software.

To reduce the negative effects from this fact, a routine that keeps the value of the last reading received by the sensor was created within the algorithm, until a new reading returns a value that represents the current state of the vehicle in relation to the curbs, and thus allowing the algorithm not to lose the required parameters for the adequacy of its trajectory. Obviously that impairs the accuracy in decision-making in the process of adjustment carried out by trajectory algorithm, contributing sharply to the reduction of the success in a larger number of laps without the occurrence of accidents.

Figure 7 – routine that maintains the values of the last valid reading of sensors.

```
//Procedure que atribui os valores de distância obtidos dos sensores às variáveis
void verificaObstaculo(){
  frente = verificaFrente();
  esquerda = verificaEsquerda();
  direita = verificaDireita();
  re = verificaRe();
  if (esquerda != 0){
    esq = esquerda;
  }
  if (direita != 0){
    dir = direita;
  }
}
```

Source: Drawn by author.

As a way to solve this problem, infrared sensors could be used, in association with Ultrasonic sensors, they would not suffer the inconvenience caused by the phenomenon of reflection as a function of angle, being sensors that use infrared light. However, it is noteworthy that neither of them alone would not represent the perfect solution to the project, considering that these sensors are unable to perform readings when directed against translucent objects, such as glass. So, the possible solution for this restriction would be the combined use of the two types of sensors.

4 FINAL CONSIDERATIONS

Even with the occurrence of collisions after a given number of cycles with success, the AutoDrive project presented a positive performance in the face of the serious restrictions imposed by hardware used. With the use of a vehicle that offers a better infrastructure and support as well as integration of Ultrasonic and infrared sensors, and also, the improvement of the control software, it can greatly be improved in efficiency and effectiveness.

The project AutoDrive, with the necessary adjustments and improvement can give rise to a device capable of making overland transport more efficient and secure, moreover, being deployed on real vehicles, it can be integrated with other technologies, such as GPS to feed information about the path to be traversed. By making use of open source technology, it can still present itself as a low cost solution and thus be offered to consumers at an affordable price.

BIBLIOGRAPHICAL REFERENCES

ARDUINO. Disponível em <http://www.arduino.cc/>. Acesso em 17/11/12. 2005a.

ARDUINO. **Topic: How to connect L298N Motor Driver Board [SOLVED]**. Disponível em <http://forum.arduino.cc/index.php?topic=84439.0>. Acesso em 17/11/12. 2005b.

BRASIL, MINISTERIO DAS CIDADES. **O trânsito em números**. Disponível em <http://www.brasil.gov.br/sobre/cidadania/gentileza-urbana/paz-no-transito/o-transito-em-numeros>. Acesso em 24/11/12. (S.D.).

EVAN M. et al. **Arduino em Ação**. Novatec, São Paulo, S.P, 2013.

SCIESLESKI, A.J. Aspectos psicopatológicos do homem no trânsito. **Rev. Bras. Med. Tráf.**, 1982.