ELECTRONIC PART OF AUTONOMOUS PARKING SYSTEM

Michał Jaros, Bartosz Owsiany, Witold Surdej, Bartosz Setlak
AGH Akademia Górniczo-Hutnicza im. Stanisława Staszica
Wydział Inżynierii Mechanicznej i Robotyki,
Al. Mickiewicza 30, 30-059 Kraków

0. ABSTRACT

In this paper, we will present the details of the creation of an electronic system capable of autonomous parking. It is an autonomous car-manoeuvering system that moves a vehicle into a parking spot without human intervention. It allows a vehicle to perform parking maneuvers at different angles. It aims to enhance the comfort and safety of driving. The parking maneuver is achieved by coordinating steering angle and speed which takes into account the actual situation in the environment to ensure collision-free motion within the available space[1]
Keywords: Automatic parking, scientific paper, vehicles, electronics

1. INTRODUCTION

In summer semester 2020/2021 students of mechatronic engineering were tasked with designing and building a functional robot picked by them from the list provided by dr Daniel Prusak. We split into 3 groups and our group picked the autonomous parking system i.e. a self-parking car. Because of the pandemic situation in Poland, we were unable to work traditionally. Most of us could only work from our homes and communicate through the internet which made everything more challenging. Under normal circumstances, we would design and build the robot on the site of the AGH university with physical lego bricks. Because of the aforementioned problems we were only supposed to design the car digitally, we were not obliged to create a physical prototype of it using the lego bricks. We ended up creating a prototype that was similar to the digital project but with a different system.
Our subgroup was tasked with constructing an electronic part of an autonomous parking system. We needed to choose and connect elements to gather data, process it, and steer the engines with this data. Additionally, we had to choose a type of power supply capable of supporting the whole system.

2. PURPOSE AND SCOPE OF THE WORK.

The purpose of our project was to design the electronics for the car to be capable of self-parking. To do this, we overview the existing solutions and ways in which the main problems can be solved and picking the options that are the most suitable for us. The

purposes of the used components were making the car move and turn, detecting the distance between the car and the obstacles, and measuring the velocity of the car. All these values are needed for calculating the proper way for the car to move.
Scope:

3. OVERVIEW OF EXISTING SOLUTIONS

Parallel parking has long been a struggle for licensed drivers and one of the most dreaded parts of a driving test. Many drivers labor through the maneuver, often damaging their car or deciding to look for non-parallel parking options elsewhere. Though modern technologies such as parking sensors or reversing cameras can help with parallel parking, many drivers prefer avoiding parallel parking altogether.

Luckily for today's drivers, self-parking cars are no longer a thing of the future as many car brands offer this feature in several of their models. Car manufacturers have finally provided a long-awaited solution for drivers who have avoided parallel parking through the years or worried about their parallel parking efforts time and time again.

figure 1.1 Audi A8 with self-parking system.

At present, there are two main methods of automatic parking systems: the methods based on ultrasonic sensors and those based on visual sensors. We decided to choose based on ultrasonic sensors because analyzing the image by visual sensors is too complicated. Ultrasonic parking sensors use high-frequency sound waves to detect objects. These sensors emit sound pulses that reflect off of nearby objects. A receiver detects the reflected waves and calculates the distance from your vehicle to the object. We used 6 of these: 3 in the front and 3 on the back (left side, right side, and back).

The second most important device in the car, while parking, is the speedometer. It gives the processor information about how fast the vehicle moves, that it can use when making decisions. In our case, we used an infrared sensor that detects speed.

4. MORPHOLOGICAL CHART

Morphological Chart of the electronic components of Autonomous parking system										
sub-functions	Solutions									
processing signals from sensors	Arduino Uno [2x]		Arduino Mega [2560 Rev3 - A000067]		Raspberry pi		Banana Pi			
Criterium	2	3	2	3	2	1	2	1		
	0	3	0	3	0	2	0	1		
detection of the obstacles	Magnetic proximity sensor		Capacitive proximity sensor		Ultrasonic proximity sensor [HC-SR04 6x]		Inductive proximity sensor		Optical ultrasonic sensor	
Criterium	3	2	2	1	3	3	1	1	1	1
	0	1	0	1	0	3	0	2	0	2
supplying energy to the microcontroller	shared supply for shield & controller		5V battery		USB connection		5V adapter			
Criterium	3	2	3	3	2	2	2	3		
	0	3	0	3	0	1	0	2		
measuring the speed of the car	Infrared sensor module [Arduino 51 AVR pic]		Arduino Speed sensor module		Magnetic speed sensor		Gears detecting speed sensor			

	3	3	2	2	2	2	2	2		
Criterium	0	3	0	3	0	2	0	1		
piloting of the steering axle	Servo		Stepper motor [23byj-48]							
	3	3	3	3						
Criterium	2	2	3	3						
wireless communication with controller	Radio Frequency module		Wi-Fi		Cellular		Bluetooth [SPP2.0]			
	2	2	2	2	1	1	3	3		
Criterium	3	2	3	2	3	1	3	3		
piloting of the motors and engine	Servo		Arduino Motor Shield [L293D]							
	3	3	3	3						
Criterium	2	2	3	3						
supplying energy to the stepper motor	Lipo battery [12V]		12V adapter		Battery basket [3x 18650 (12V)]					
	3	2	3	2	3	3				
Criterium	0	3	0	2	0	3				
propelling the car	Integrated system of Wheel+engine [65x26mm, 5V, 48:1 gear x2]									
	3	3								
Criterium	3	3								
Criterium					Final Choice					
Price	Availability									

durability	Ease of use

Criterium Scale			
0- unable to judge	1- nonsatisfaction	2- partial satisfaction	3- full satisfaction

[5]

5. MODEL AND SIMULATIONS:

When designing the electronics system, we took into account the most important functions the self-parking car should possess and we split them into smaller projects. We picked the most fitting components, i.e. sensors, motors, plates, etc.

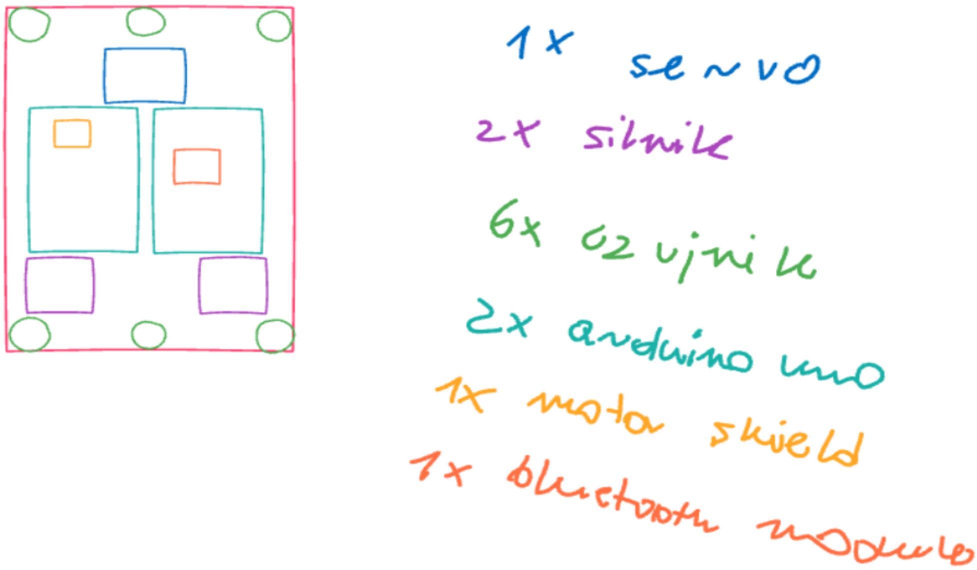


Figure 1.3 original theoretical placement of electrical components

We have started by creating a rough sketch of a complete system (figure 1.3). Using a method of rapid prototyping, we have assembled an incomplete system (figure 1.4) to check if the elements chosen by us will be able to support the final project.

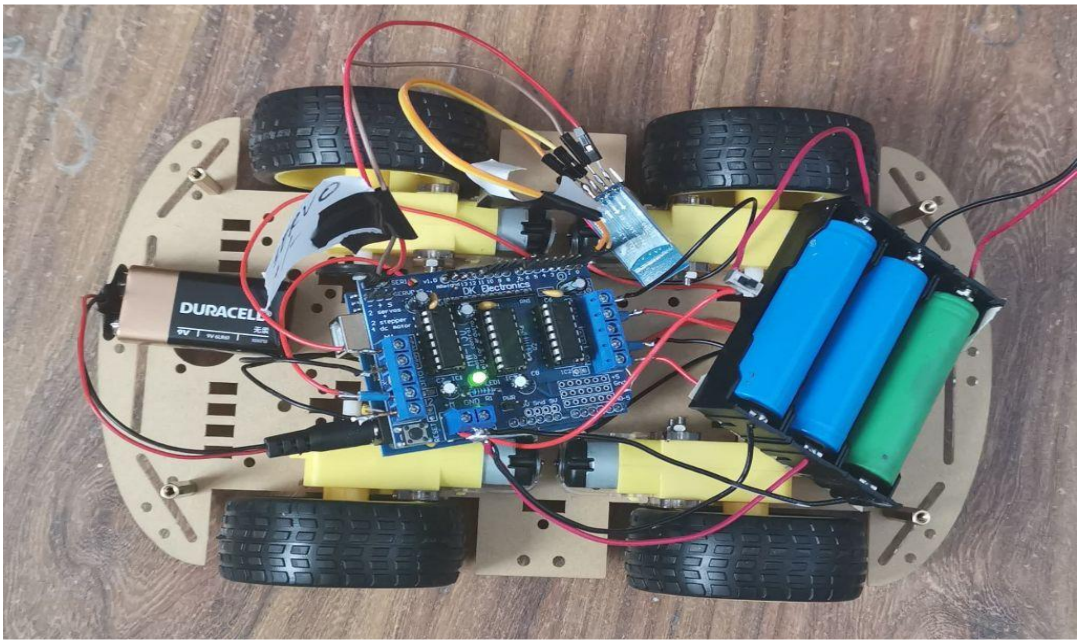


figure 1.4 1st attempt at steering car (system of engines + Bluetooth controller). Firstly motion of our vehicle was controlled by 4 engines using a tank-like method of turning.

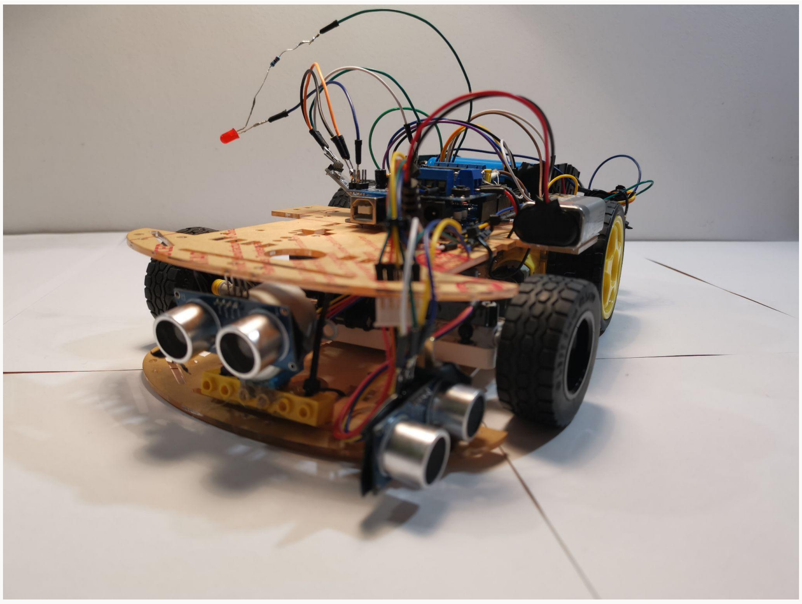
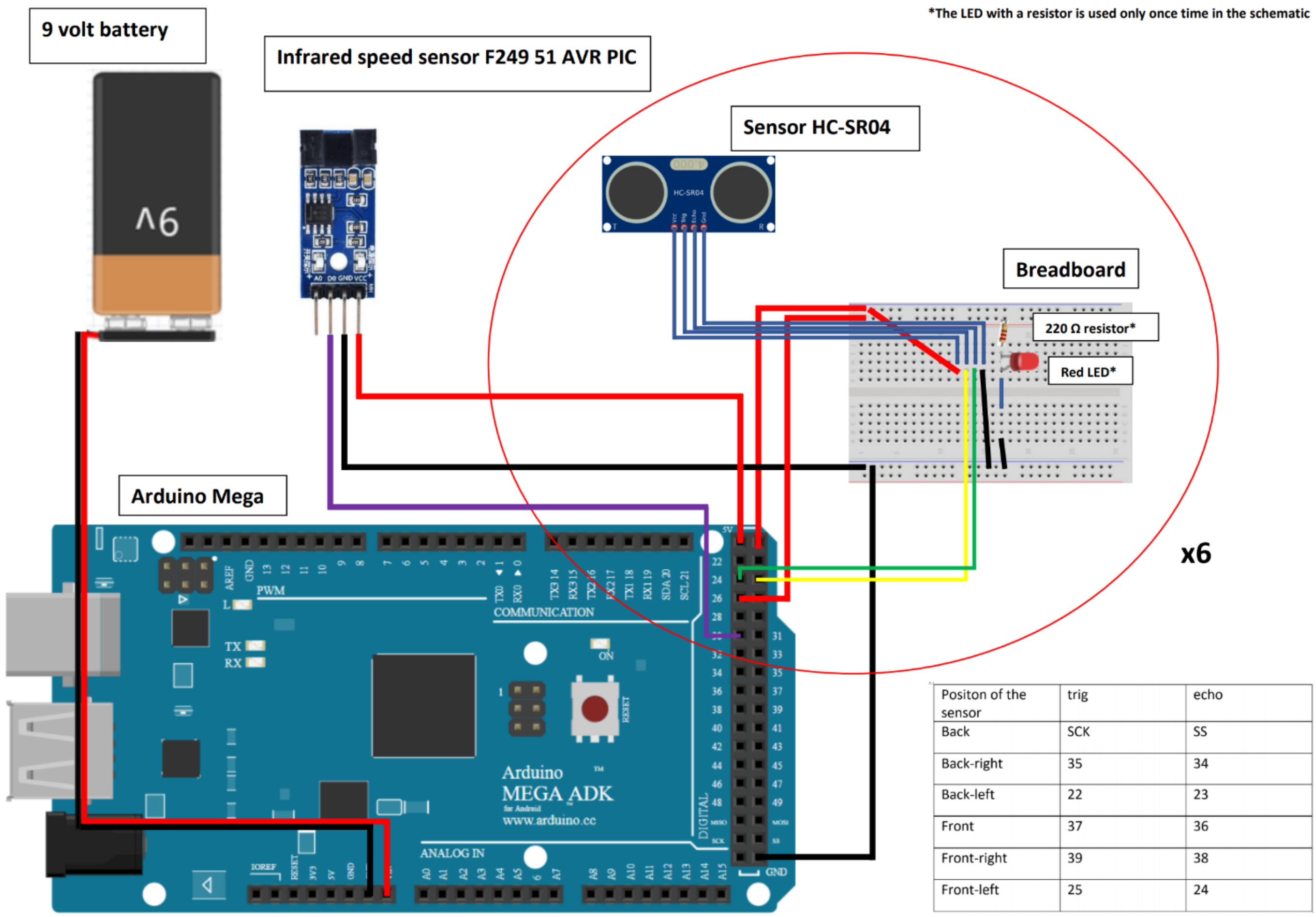


figure 1.5 2nd attempt at steering car (system of engines + Bluetooth controller + steering axle). In the second attempt, we replaced one pair of engines with a steering axle connected to the servo.

Then we proceeded to create a base for the system detecting the empty parking spot based on ultrasonic sensors.
One of the most important decisions we made was using Arduino Mega for our project. We picked it because of the large number of pins that were needed for plugging in all the sensors. We also decided to use Arduino Motor Shield for powering the motors connected to the rear wheels used to set the car in motion. The Motor Shield is also powering the stepper motor which makes it possible for the car to turn. The energy consumption turned out to be too big for Arduino alone, hence the use of Motor Shield. Arduino Mega can take up to 9V of

voltage and Motor Shield up to 12V. Because of this, we decided to use two power supplies to power these components separately to provide safety and to prevent the plates from burning.



Positon of the sensor	trig	echo
Back	SCK	SS
Back-right	35	34
Back-left	22	23
Front	37	36
Front-right	39	38
Front-left	25	24

figure 1.6 the final design of Arduino Mega circuit
Arduino Mega is powered by one 9V battery [4]

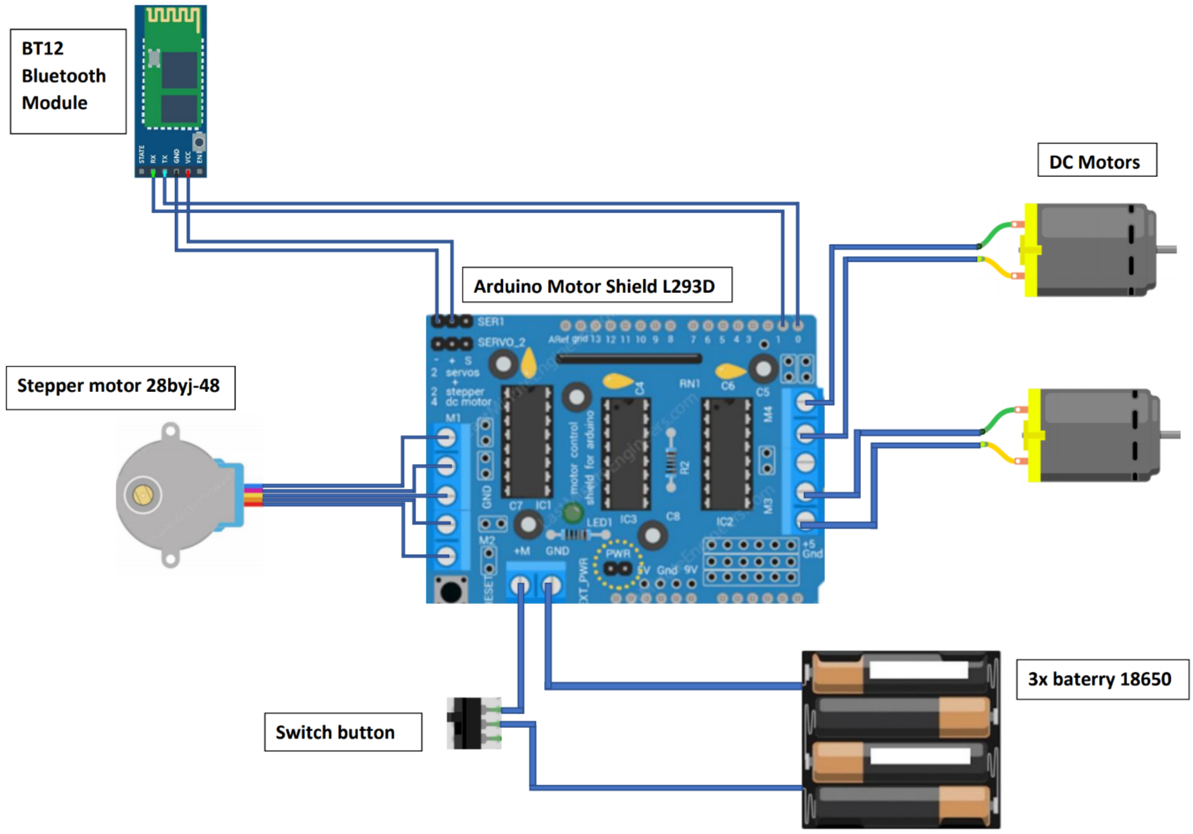


figure 1.7 the final design of Arduino Motor Shield circuit
Motor Shield is powered by three 16850 cells, a little bit more than 12V in total [4]

The component turned out to be very easy to assemble. We did it following the Arduino guidelines.

6. PROTOTYPE AND TEST

After assembling the base of a vehicle (figure 1.4). We have tested that we can steer the engines using a Bluetooth app, and the system was functioning. We have been testing everything on a physical model. Firstly we tested steering using 4 motors but then we added a solution with a movable front steering axle connected to the stepper motor and the programming team adjusted code for that solution, then we tested the mechanism again. A solution including steering axles enables more accurate control of the vehicle's path.

7. SUMMARY & CONCLUSIONS

1) We have managed to choose the correct electronic parts, we have connected them and successfully created the basic model of a vehicle.
2) The most important decision we’ve made was choosing Arduino Mega over two Arduino Unos.
3) We have created an online prototype of an ultrasonic sensor system.

4) The most interesting part of the project was the choice of sufficiently accurate motors. We had to consider multiple options to finally find the correct ones. It was also the biggest problem that we have encountered so far.
5) We've completed the design with satisfactory results. All of the components cooperate successfully, there are no issues.

8.BIBLIOGRAPHY:

[1] https://en.wikipedia.org/wiki/Automatic_parking
[2] DM TECHTRIX Self Parking Car using Arduino Uno/ Node MCU - YouTube 2019
[3]https://www.google.pl/url?sa=i&url=https%3A%2F%2Fautokult.pl%2F34164%2Cvolvo-xc90-z-fabrycznym-opancerzeniem-bezpieczenstwo-musi-swoje-wazyc&psig=AOvVaw0sCTF85kGtIBmFvLPmIub_&ust=1620289391794000&source=images&cd=vfe&ved=0CAIQjRxqFwoTCLCmhsCOsvACFQAAAAAdAAAAABAD
[4] List of graphics used during creating schematic:
https://lastminuteengineers.com/l293d-motor-driver-shield-arduino-tutorial/
http://neruo.pl/drv8835.html
https://digitalmaestro.org/articles/buttons-and-switches-in-tinkercad-circuit-projects
https://forbot.pl/blog/kurs-qt-4-aplikacja-mobilna-lacznosc-bluetooth-z-arduino-id35603
https://paulvollmer.net/FritzingParts/parts/Arduino_ADK_MEGA_2560-Rev3(icsp).html
https://lastminuteengineers.com/wp-content/uploads/arduino/Wiring-28BYJ48-Stepper-Motor-with-ULN2003-Driver-and-Arduino.png
https://maker.pro/custom/tutorial/an-introduction-to-circuit-design-with-fritzing
https://arduibots.wordpress.com/2015/10/12/component-hc-sr04-for-fritzing/
https://en.aliradar.com/item/32999009393-tzt-f249-4-pin-infrared-speed-sensor-module-for-arduino-51-avr-pic-3.3v-5v-high-quality
https://www.quora.com/How-do-I-connect-a-12V-battery-pack-to-an-Arduino-Mega-2560
[5] list of links used to create the morphological chart:

https://circuitdigest.com/microcontroller-projects/speed-distance-and-angle-measurement-for-mobile-robots-using-arduino-and-lm393-h206

https://www.theengineeringprojects.com/2015/02/interfacing-multiple-ultrasonic-sensor-arduino.html

http://www.electronicsblog.net/arduino-and-ultra-sonic-range-measurement-module-or-how-to-measure-the-pulse-time-with-a-hardware-timer-and-an-interrupt/

https://www.tutorialspoint.com/arduino/arduino_stepper_motor.htm

https://forum.arduino.cc/t/running-two-arduinos-in-sync/314352

https://www.tutorialspoint.com/arduino/arduino_dc_motor.htm https://www.orientalmotor.com/brushless-dc-motors-gear-motors/technology/brushless-dc-motors-servo-motors-inverter.html

https://www.arduino.cc/reference/en/libraries/servo/

additionally information from google (such as prices of the components)

and YouTube tutorials & guides