# Architecture of the system of 1:10 scale autonomous car - requirements-based design and implementation

Mateusz Perciński, Mikołaj Marcinkiewicz
Students' Robotics Association, Faculty of Power and Aeronautical Engineering
Warsaw University of Technology
Warsaw, Poland
Contact e-mail: mateusz.percinski@gmail.com

***Abstract*—This paper presents the architecture of the system of 1:10 scale autonomous car. It emphasizes how requirements-based approach was followed in project design phase not only to meet formal requirements of Carolo-Cup 2018 competition, but also to maximize vehicle performance. In competition events vehicles are supposed to operate in road-simulating environment, particularly to handle real-time navigation, static and dynamic obstacles avoiding, parking and following right of the road in intersections. In dynamic events quality of behavior and pace of performed actions are graded. Additional factors influencing final assessment are safety engineering, cost efficiency and knowledge management in team.**

***Keywords*—carolo-cup, autonomous car, system integration, computer vision.**

## I. Introduction

The main purpose of this paper is to introduce the architecture of the system of 1:10 scale autonomous car - requirements-based design approach and implementation of ready-to-go system.

As market introduction of automated vehicles is coming closer and closer, questions regarding safety and system validation of algorithm are getting more and more important. Many approaches need to be tested properly in non-impacting environment. Challenges concerns marking detection [3], obstacles avoiding with sensors [5], task handling reliability [4]. Technologies used in autonomous cars are very likely to become break-thoughts in other fields [6].

1:10 scaled vehicles are low-cost platforms that enable testing most of aspects of localization and navigation algorithms for real scale autonomous cars. Carolo-Cup provides a space for ideas testing, competing and know-how sharing for students team.

## II. System requirements

Main goal of project was to design and build robotic agent that will be allowed to compete and achieve strong result in Basic-Cup (competition class in Carolo-Cup event taking place yearly at Braunschweig University of Technology, Germany). Vehicles that are allowed to compete in Carolo-Cup are highly specified by competition's regulations [1]. Requirements cover both mechanical construction and behavior of vehicle in specific scenarios. Additional requirements were outlined during design phase of project based on factors that are supposed to improve car performance in particular tasks specified in regulations of Basic-Cup.

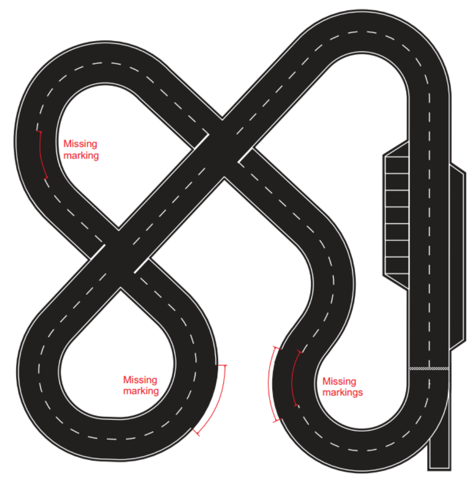



Figure 1. Sample track in Carolo-Cup competition (Basic-Cup) [1]

Below are specified requirements taken into consideration in design phase.

**Dimension of vehicle**

The vehicles must be based on four-wheeled 1:10 scale chassis. Only two axles are permitted. The wheel base must measure at least 200 mm. The track width (measured from the center of the wheels) must measure at least 160 mm. The vehicle, including possible extensions and bodywork, must not be wider than 300 mm. The height of fixed installations must not exceed a height of 300 mm above the track surface. For the acceptance test, the car must be driven through a fixed gate (inner dimensions: height 300 mm, width 300 mm) in RC-mode.

**Car lights**

The vehicle must be equipped in car lights which shall signal different driving maneuvers. This installation include three braking lights at the rear of the vehicle signaling active braking, yellow / orange direction indicators at each corner of car.

Research done cooperating with the Faculty of Power and Aeronautical Engineering, Warsaw University of Technology within the framework of Najlepsi z Najlepszych 2.0!" program.

Additionally, a clearly visible blue light is to be installed at the highest point of the vehicle, which flashes to signal the activation of RC-mode.

**Following Road**
In every event car is supposed to follow the road, driving in the right lane. This scenario shall imitate a rural road environment, consisting of long straight sections, tight turns, intersections and also containing a parking lot. All markings are white done on black background. Vehicle should be able to handle **missing lines**. Both right and center lines can be absent by 1 meter. Figure 1 presents sample track in Carolo-Cup competition.

**Obstalces overtaking**
During the event "Obstacle Evasion Course" the vehicle is suppose to handle static and dynamic obstacles. Obstacles could be located in right and left lane. Every lane change performed by car must be signaled with lights similarly to real traffic regulations. Correctness of maneuvers is narrowly described in Carolo-Cup regulations [1].

**Intersections**
The vehicle should be able to recognize and act properly in road intersections, especially to respect the right of way according to markings on the roads.

**Parking**
Vehicle is supposed to be able to recognize parking area on find free slot of proper size and perform parallel parking avoiding collision with adjoining obstacles and crossing slot boarding lines. Relevant light signalization during parking maneuver is required.

**Remote control overtaking**
In emergency situations, the vehicle must be stoppable and maneuverable using a remote control. Remote takeover is supposed to happen immediately on demand.

**Safety Engineering**
Ensuring safe and deterministic behavior of the vehicle is factor graded in static events.

**Cost Efficiency**
Cost and Power efficiency is taken into consideration in grading system.

**Knowledge management**
Providing continuous integration for car systems and knowledge management for future developments is additional graded factor

## III. General Design

### A. Functional decomposition to subsystems

In design phase of project requirements-based approach was followed. As there are many various functions that have to be necessarily realized by car as an autonomous agent and competitor in Carolo-Cup, mapping between identified functions and elements that are to realize these functions was provided. It is briefly presented in table I. Elements distilled are complete subsystems with interfaces designed for this project or sensors and actuators with communication interfaces specified by producers.

Table I
Functional decomposition of system

| Function | System elements |
|---|---|
| Power Supply and Management | NiMH batteries, Power Management Unit (custom PCB) |
| Road lane perception | one-board computer (Odroid XU4), USB camera (Kurokesu C1) |
| Speed Control | anyFC7 controller (STM32 F7), servomotor |
| Direction Control | anyFC7 controller (STM32 F7), magnetic encoders, DC motor |
| Remote Overtaking | RC equipment Futaba (T6K + R2006GS), proper handling by anyFC7 controller |
| Obstacles identification | laser sensors (ST VL53L0X) |
| Parking | laser sensors (ST VL53L0X), Gyroscope (IMU MPU6000) |
| Car lights | PWM controled LED strips |

### B. Overall sensor setup

Goal of project was to minimize number and cost of sensors used maintaining vehicle ability to perform in competition events. The Camera in charge of road lane perception is mount in highest point of car. Particular model of camera and lenses was chosen compromising captured width (min. 60cm) and distortion. Minimal number of distance sensor to enable secured parallel parking is 3 (front, rear and the right one). Additional equipment for parking is gyroscope (anyFC built-in IMU is used). For actual speed measurement magnetic encoders output is taken. As they are mounted directory in wheels of the car, control unit is able to compensate clearances of differential mechanism and gear shifts. Overall sensor setup is presented in figure 2.

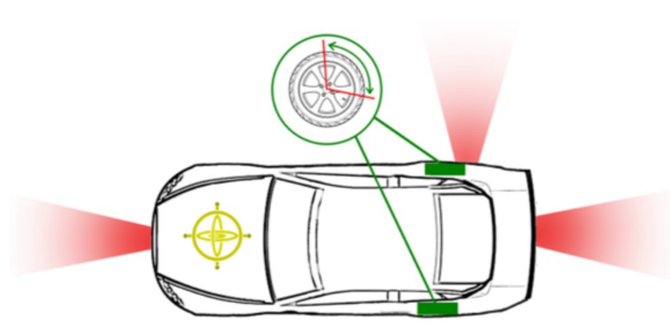


Figure 2. Placement of sensors used in vehicle

## IV. System Description

### A. Mechanical aspects of car

As specified in competition's regulation vehicle is based on 1:10 chassis of RC car. Default mechanism connecting servo and wheels of front axle realizes Ackermann steering geometry. Developing and testing on 1:10 chassis with one working axle enable scalability of system for possible future reimplementations in bigger-scaled projects. All the mounting elements and electronics grippers (also custom wheel rims to mount encoders) are printed in 3D technology.

Dimensions of car after all modifications are:

- height: 300mm (maximum allowed)
- width: 200mm
- length: 390mm

Figure 3 presents design of vehicle: 3D model3a, ready car without bodywork3b and final robot appearance3c.

(a) Model prepared with CAD software

(b) Vechicle without bodywork

(c) Final appearance of built car

Figure 3. Vehicle built for Carolo-Cup competition without bodywork

## B. Hardware architecture

From high-level perspective Vehicle system is divided into 2 main processing subsystems and other peripherals. Figure 4 presents connections schema and points peripherals used for handling communication between main controller particular system elements.

As main controller AnyFCF7 board is utilized. It is equipped with 32-bit micro-controller STM32F745 (processor ARM® Cortex-M7®, 216Mhz). AnyFC board was chosen because of its small size, relatively low price and ready-to-use connectors. Also built-in gyroscope is used.

Computer vision equipment:

1) Computer ODROID XU4 – Hardkernel, CPU: Samsung Exynos5422 Octa-Core(4 x 2,0 GHz + 4 x 1,4 GHz), RAM:2 GB (700 MHZ) LPDDR3,
2) Camera Kurokesu C1 - Resolution: 1920x1080, Max fps: 30,
3) Lens - focal length: 2.8 –12 mm, Angle of view: 93 degrees

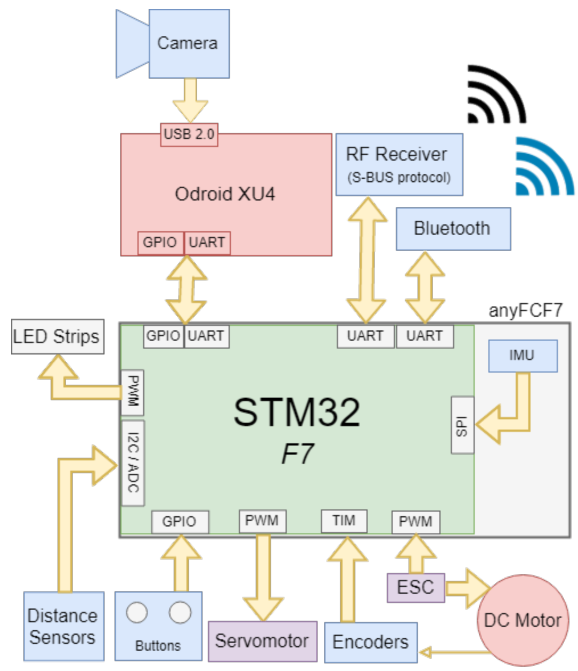



Figure 4. Map of communication between system elements

## C. Power Management

Power Management is realized by custom PCB designed for vehicle. PCB dimensions are adjusted to be of size of Main Controller Board (36x36mm) - it enables easy mounting. PMU realizes voltage stabilization, overvoltage, overcurrent and reverse polarization protection . Figure 5 presents power distribution into subsystems.

## D. Software architecture

The main assumption on software design was clear role separation between Computer Vision Processing Unit and anyFC board as main controller and communication manager. Odroid is capturing camera frames, process them and send informations to controller as fast as possible. Application running on anyFC board is scheduling the tasks, handle peripherals and use information gathered as they are needed.

As in case of autonomous cars driving in real traffic also in case of 1:10 agent there is strong demand to operate fulfilling hard real-time conditions. To provide such deterministic and safe behavior Real-Time Operating system is used. Implementation of FreeRTOS, open-source real-time operating system, is installed on STM32 micro-controller. Application running in FreeRTOS integrates and governs all data gatherings, processings and setting signals for actuators. It also realizes controllers loops.

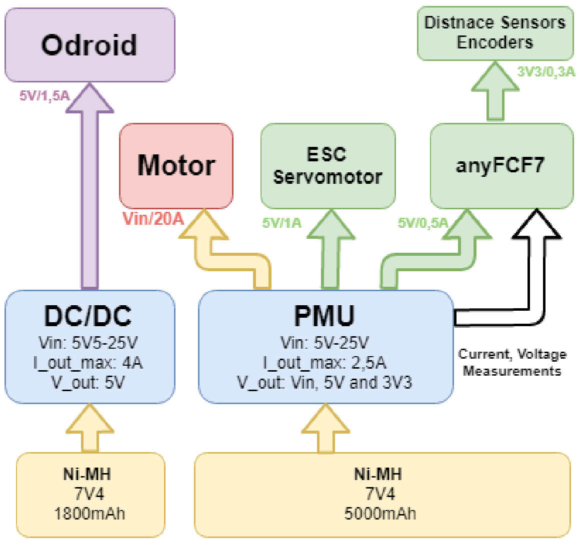



Figure 5. Power distribution into subsystems

FreeRTOS enables setting priorities for particular tasks. Figure 6 presents hierarchy of tasks realized in design application in regards to priority. The top priority is assigned to receiving information from remote control equipment. It enables to take over control on car in every situation. Task responsible for remote battery diagnostic has the lowest priority. If few tasks have the same priority the Round Robin algorithm for their threads scheduling and time sharing.

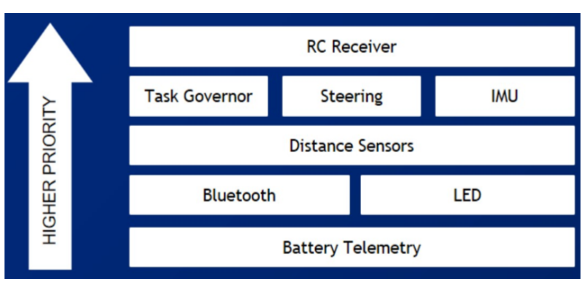



Figure 6. Prioritisation of tasks realized in FreeRTOS

Implementations of data reading and writing differ across the peripherals used for particular systems element. All of them are using DMA (Direct Memory Access) mechanism to access.

To ensure synchronization of access to data (read/write operations) Interrupts Manager is utilized.

### E. Human-Machine Interfaces

In competition's events it is forbidden to interact with car apart from emergencies, but for development, testing and pre-starting calibration phases it is crucial to have possibility to monitor car activities and set parameters remotely.

Vehicle communicates to outside world through 3 channels.

**2,4GHz Remote Control**
: It required by regulations of competition to have possibility to take over control remotely. All the values are send using SBUS protocol.

**Bluetooth**
: Bluetooth communication is designed to enable global variables (battery voltage, distance driven) monitoring and controller terms setting.

**Wifi streaming**
: Camera captured frames streaming for testing and presentation purposes. Streaming is not typically used as it is big-size load for computer vision computation unit

## V. Selected concepts description

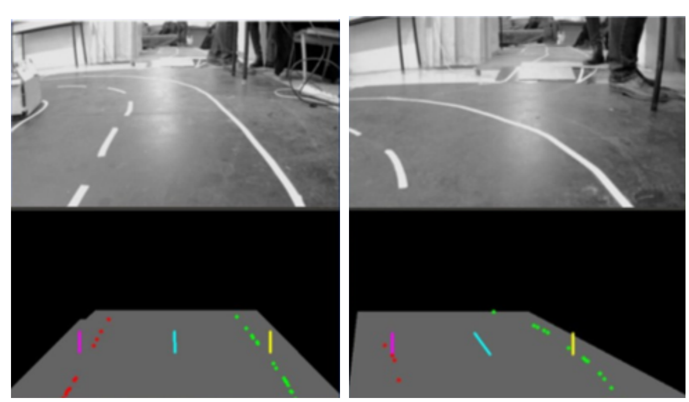
### A. Road lane perception

The main task of computer vision application vehicle localization on road lane and providing information about lane curvature. Additionally, it is in charge of delivering information about requested behavior in intersections (right of the law analysis), start of parking area and handling lane changing.

In general every captured frame is processed in following steps:

1) Camera frame capturing in YUYV2 format
2) Getting Y channel (gray-scaled image)
3) Applying Gaussian blur for noise elimination
4) Binary thresholding to distinct candidates for lines
5) Applying dynamic mask to shrink Region of Interest basing on lines captured in previous frame
6) Finding short white lines and calculating slope for each one
7) Dividing lines to associated with right and left edge of the road basing on calculated slopes
8) Determining horizontal position of car on road in regards to road edges
9) Calculating slope of the road as average of slope of its edges
10) Sending informations to main controller

Figure 7 is presenting input (top) and output (down) frames of algorithm in straight lane 7a and turning lane 7b scenarios.



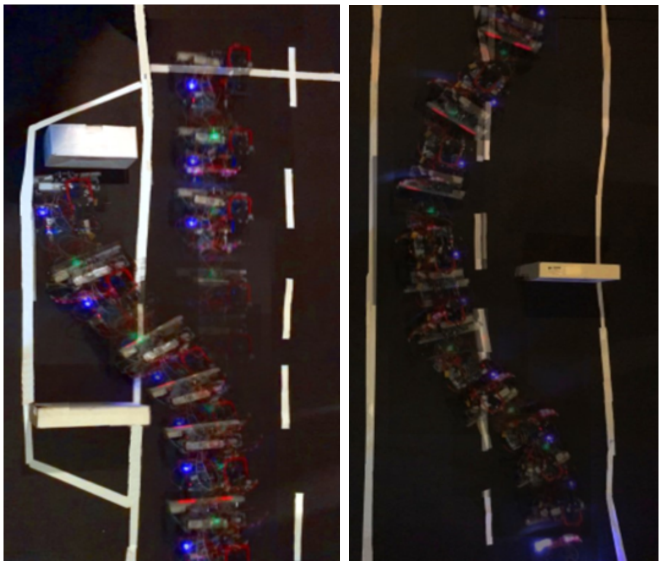
(a) Straight road scenario (b) Turn scenario

Figure 7. input (top) and output (down) frames of road lane perception algorithm

Above algorithm enable to extract proper informations from captured frames despite the fact that road edge can be continuous, dashed or broken

Apart from above sequence, algorithm is detecting checked line, which starts parking area basing on white pixel density inside the road line.

Intersections detection and proper handling is based on finding horizontal lines which indicates perpendicular roads and order to yield in part of cases. Waiting in intersection application is detecting any movements in front of the car



(a) Parallel parking sequence (b) Lane changing sequence

Figure 8. Top view for parking and lane changing sequences

### B. Parking

For parking maneuver all the sensor are used. Firstly horizontal checked line indicating beginning of parking area is detected by vision analysis. Then car detects parking spot with right sensor and measures length of spot based on encoders output. If parking spot is long enough, parking sequence starts. Car drives back using maximal turning radius till achieving desired angular position. Then drives back straightway. After set distance, drives back using maximal turning radius in opposite direction. sequence stops when car achieves parallel position or possible collision is detected. All possible values (distances and angles) are parametrized and adjusted to car actuator characteristics. To leave parking spot reverse sequence is used. Every road maneuver is properly indicated by car light.

Figure 8a shows parking sequence from top view.

### C. Collision Avoidance

In Carolo-Cup event car is supposed to handle static and dynamic obstacles occurring on right or left lane of the road. Front distance sensor is used for obstacles detection. Computer Visions Unit is informed to switch the lane edge assignment. Left line start to be treated as right one and car is changing road lane. After detection end of obstacles with right distance sensor reverse sequence is performed.

Figure 8b shows obstacles evasion sequence from top view.

## VI. Results

The designed systems are successfully implemented in vehicle. Car built with total budget of €900 is capable to perform all the task defined in requirements.

Assuming minimal turn radius is 1 meter length (accordingly to competition regulations) and no obstacle on the road vehicle is able to operate continuously at speed of 0.8 m/s. Taking into consideration scale it corresponds to almost 30 km/h. High quality direction control above that speed is impossible as camera-based information are too low interval - camera is capturing up to 20 frames per second in typically illuminated interiors.

In similar course it consumes average current of 12A with maximum value of 21A. Processors running computer vision algorithm are utilizing 50% to 60% of their resources (in average).

The car was able to successfully finish all the events of Carolo-Cup 2018 (Basic-Cup). In graded courses it was configured to operate slowly to minimize amount of penalties. It results in 4. result in Free Drive and Parking event and 5. result in Obstacle Evasion Course. Comparing with competitors' constructions, the vehicle achieved shorter distance in given time, but smaller amount of penalties than majority of teams. Additionally, the team achieved 6. best result in static events, so presentation of concepts behind designed construction.

## VII. Discussion

Requirements-based approach worked very effectively in case of designing car for Carolo-Cup. This approach shows its robustness with scenarios with very detailed requirements given. It also enables keeping project budget low and avoiding dispensable costs. Designed system perform as it was expected. Only minor changes in initial design occurred in implementation phase.

Although the vehicle is able to perform all required actions, there is a lot of space for improvements, especially in increasing operating speed. The solutions for that can involve moving to higher quality hardware, e.g. high speed camera or changing vision processing algorithms for more effective ones. The other solution that will enable vehicle performance is higher amount of distance sensors or using Lidar sensor for continuous environment mapping (high cost solution). Alternatively, determination of distances to obstacles could be performed by processing camera frames [10]. For using more sophisticated motion planning algorithms it is crucial to move most of tasks running on micro-controlled to more powerful computer (same computing vision or dedicated one).

## Acknowledgment

Authors are members of Students' Robotics Association (Koło Naukowe Robotyków) at the Faculty of Power and Aeronautical Engineering, Warsaw University of Technology.

Research done cooperating with the Faculty of Power and Aeronautical Engineering within the framework of "Najlepsi z Najlepszych 2.0!" program.

## References


[1] Basic Cup Regulations 2018, July 15, 2017. https://wiki.ifr.ing.tu-bs.de/carolocup/en/system/files/Basic%20Cup%20Regulations.pdf, Available on-line December 27, 2017

[2] S. Zug, C. Steup. J-B. Scholle, C. Berger, O. Landsiedel, F. Schuldt, J. Rieken, R. Matthaei, T. Form *Technical Evaluation of the Carolo-Cup 2014 - A Competition for Self-Driving Miniature Cars*, 2014 IEEE International Symposium on Robotic and Sensors Environments (ROSE) Proceedings, 2014, pp. 100 - 105

[3] B. S Khan, M. Hanafi, S. Mashohor *Automated Road Marking Detection System for Autonomous Car*. 2015 IEEE Student Conference on Research and Development (SCOReD), 2015, pp. 398 - 401

[4] Shimil Jose, Sajith Variyar V V, Soman K.P *Effective Utilization And Analysis Of ROS On Embedded Platform For Implementing Autonomous Car Vision And Navigation Modules*, 2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI), 2017, pp. 877 - 882

[5] A. Iqbal, S. S. Ahmed, M. D. Tauqeer, A. Sultan, S. Y. Abbas *Design of Multifunctional Autonomous Car using Ultrasonic and Infrared Sensors*, 2017 International Symposium on Wireless Systems and Networks (ISWSN), 2017, pp. 1 - 5

[6] M. Martinez, A. Roitberg, D. Koester, B. Schauerte, R. Stiefelhagen *Using Technology Developed for Autonomous Cars to Help Navigate Blind People*, 2017 IEEE International Conference on Computer Vision Workshops (ICCVW), 2017, pp. 1424 - 1432

[7] G. S. Pannu, M. D. Ansari, P. Gupta *Design and Implementation of Autonomous Car using Raspberry Pi*, International Journal of Computer Applications (0975–8887) Volume 113-No.9, March 2015

[8] T. Wankhade, P. Shriwas, *Design of Lane Detecting and Following Autonomous Robot*, IOSR Journal of Computer Engineering (IOSRJCE) ISSN: 2278-0661 Volume 2, Issue 2 (July-Aug. 2012), pp. 45-48.

[9] X. Miao, S. Li, H. Shen, *On-Board lane detection system for intelligent vehicle based on monocular vision*, International Journal on Smart Sensing and Intelligent Systems, vol. 5, no. 4, December 2012, pp. 957-972.

[10] S. Tuohy, D. O'Cualain, E. Jones, M. Glavin, *Distance determination for an automobile environment using inverse perspective mapping in OpenCV*, Irish Signals and Systems Conference 2010.