



## Modular smart control system architecture for the mobile robot platform

Raivo Sell\*, Eero Väljaots, Tengiz Pataraiia, and Ehsan Malayjerdi

Department of Mechanical and Industrial Engineering, Tallinn University of Technology, Ehitajate tee 5, 19086 Tallinn, Estonia

Received 12 April 2019, accepted 16 May 2019, available online 19 November 2019

© 2019 Authors. This is an Open Access article distributed under the terms and conditions of the Creative Commons Attribution-NonCommercial 4.0 International License (<http://creativecommons.org/licenses/by-nc/4.0/>).

**Abstract.** The paper is focusing on the research of current technologies of multi-purpose mobile robots, including implementation of modular layout, open software frameworks, self-organization with plug and play (PnP) capability and adaptive sensor fusion. The hardware layout is studied in detail and modular architecture is proposed as an optimal solution of module interconnection. Software modular architecture is developed in the similar principle, taking plug and play connectivity into account. The framework of the software consists of implementing middleware for the software modularization and three-level hierarchical structure. Practical implementation platforms are introduced and future developments discussed.

**Key words:** mobile robot, unmanned ground vehicle, system architecture.

### 1. INTRODUCTION

Unmanned ground vehicles (UGVs) used to be operated by the remote control in performing hazardous tasks over the distance. Nowadays, UGVs, are developed for various purposes, e.g., for research and industrial use, and these are sophisticated vehicles having semi-autonomous or full autonomous modes. For autonomous navigation and safe driving more sophisticated software and hardware is required, including high performance sensors like light detection and ranging (LiDAR); and cameras. Autonomous driving requires different sub-tasks to be solved, e.g. localization, base-navigation, local and global planning, obstacle avoidance, etc. [1]. Although these tasks are similar for most of the mobile robots, they still are very dependent on the selected sensors and robot hardware specifics. To overcome this issue, modular structure of the system is the most obvious choice, especially if the similar functionality has to be applied for different mobile robots. The modular concept of the

mobile robots has been studied in many researches and specific solutions have been offered, e.g. in [2,3]. This research is based on the knowledge and expertise acquired from the first Estonian self-driving car – ISEAUTO [4,5], which was put on the road after less than a year-long development. The success of the project motivates to apply similar software and hardware concept also to smaller off-road vehicles. The conceptual solution is an open source modular smart control system for mid-size off-road mobile robotic platforms. Two different robots were used for the experiments and implementation of the concept. The manuscript is organized so that three following chapters introduce two mobile robots as a base platform for the concept implementation. Chapters 5 to 7 describe the concept of the modular architecture and the implementation.

### 2. IMPLEMENTATION PLATFORMS

The unmanned robotic platforms where the concept of modular architecture is applied, are universal mobile

\* Corresponding author, [raivo.sell@taltech.ee](mailto:raivo.sell@taltech.ee)

robotic platform UKU – developed by the students of Tallinn University of Technology (TalTech); and Agronaut – a robot for the agriculture, developed by the Estonian company Hecada. Both vehicles are in the same size but use different concepts of steering and power system.

### 3. UNIVERSAL MOBILE ROBOT UKU

The all-terrain mobile robot UKU, shown in Fig. 1, is powered by the electric motor and Li-Io batteries. The robot is with a rear-wheel drive without differential and mechanical transmission. It has off road tires suitable for climbing over the obstacles or for example, plowing snow autonomously on the parking lot in winter. The robot has a special self-contained measurement system to measure the efficiency, similar to [6] and dynamics of energy consumption. Robot weights 260 kg and has nominal speed of 4 m/s. The main electric motor is a permanent magnet DC motor producing 4 kW. Basic

navigation sensors are SICK 2D laser scanners (measuring range up to 80 m), ultrasonic rangars on both sides and long-range 360-degree LiDAR on the top of the vehicle. The more detailed architecture of the robot is described in [7].

### 4. AGRICULTURAL MOBILE ROBOT AGRONAUT

The Agronaut [8], shown in Fig. 2, is a universal mobile robotic platform, which purpose is practical testing of unmanned technologies and navigation in agricultural conditions. Its physical layout is symmetric and modular, consisting of identical modules that are connected to each other through hydraulic steering linkage. One module accommodates the 15-kW power plant that powers the hydraulic system. The other module is free for transporting necessary task-specific equipment. It is also possible to connect the third and the fourth module and actuators using the same physical interface on both



Fig. 1. Universal mobile robot UKU.

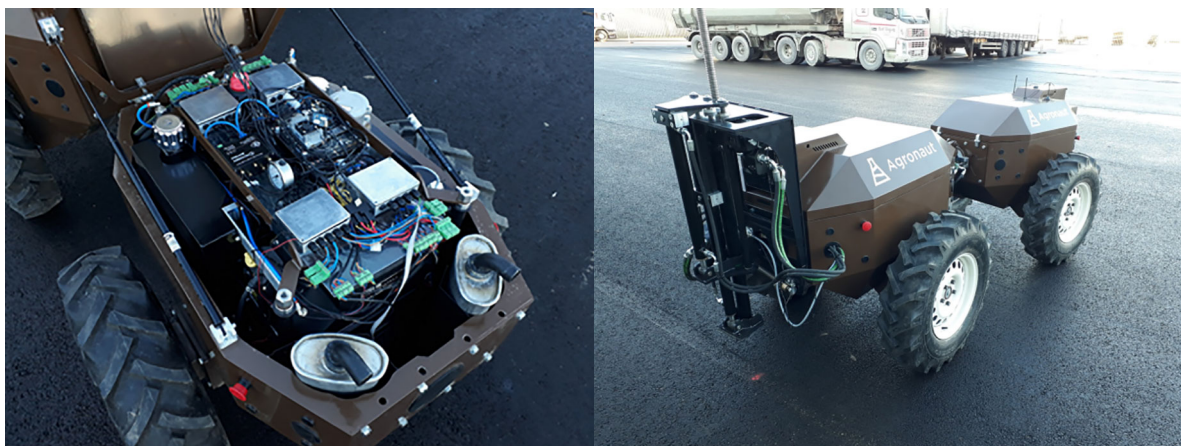


Fig. 2. Agronaut UGV platform, modular electronic units on top of the hydraulic system.

ends of the body control module. Agronaut is all-wheel drive robot and has robust design to suit for the agricultural field of use. Therefore, unlike robot UKU, all actuators outside the body are hydraulically powered and hydraulic lines are routed from one unit to another. As the vehicle can be assembled with task-specific modules, the control system also has to be modular and easily configurable for the required task. Each hydraulic actuator has its own electronic control, implementing PID-regulators for controlling hardware, connected with central computer through controller area network (CAN). As the platform weighs 470 kg, it suits perfectly for automating repetitive simple tasks usually carried out by humans, e.g., automated soil sampling of the cultivated land.

### 5. CONCEPT OF MODULAR ARCHITECTURE

Hardware and software architecture were developed based on technical requirements which were set initially according to the level of functionality and robustness of the system. Key priority of software and hardware architecture was to create a safe, easily reconfigurable and scalable system where each module/microcontroller has a task to control each locomotion unit (steering, driving motors, brakes) and different sensors across the platform. The whole control system is based on two communication protocols – Controller Area Network (CAN) bus and Universal Datagram Protocol (UDP). These protocols were chosen based on several key priorities: speed, reliability, and robustness. Intermodule communication was developed based on the CAN bus. Choice of CAN bus was made due to its high-transmission reliability, real-time capabilities, and robustness. Communication between the master controller and the computer was done through UDP messages due to its speed, reliability, and efficiency. Figure 3 shows SysML block definition diagram (bdd) and internal block diagrams (ibd) for general modular architecture. These

diagrams describe how messages are delivered from the main computing unit to the lower level controllers and main parameter values of the data flow.

### 6. IMPLEMENTATIONS OF MODULAR ARCHITECTURE

The concept of modular architecture is applied to the base platforms described earlier. The hardware architecture implemented in mobile robot UKU has two-level software control, and three-level hardware control architecture. Figure 4 shows implementation of the hardware architecture in specific UGV platform. It has high-level sensors dealing with navigation and obstacle avoidance. High-level sensors are directly connected to the main computing unit, which runs the Robotic Operating System (ROS) middleware and open source components for its main tasks. ROS-based system control algorithms produce driving commands to low-level controllers through the master controller. Master controllers prioritize and translate messages between high-level and low-level controllers. Low-level controllers are dealing with direct control of motor drivers, consisting of proportional-integral-derivative (PID) and ground safety algorithms. There is a separate safety controller and it is independent from both high-level and low-level controllers. It monitors output signals as well as CAN network messages and can stop the vehicle in case anomalies occur. The system has also remote link over the 4G/5G mobile network to provide online data stream between server and operator.

Control software for each control unit of the mobile robot was designed in a similar manner to make it simple, easily understood and configurable in the future for different systems, e.g. front and rear locomotion units of the robot have the same PID regulators with different data inputs and controller coefficients.

Let’s take an example of a steering controller where desired wheel angle (setpoint) is sent from the ROS

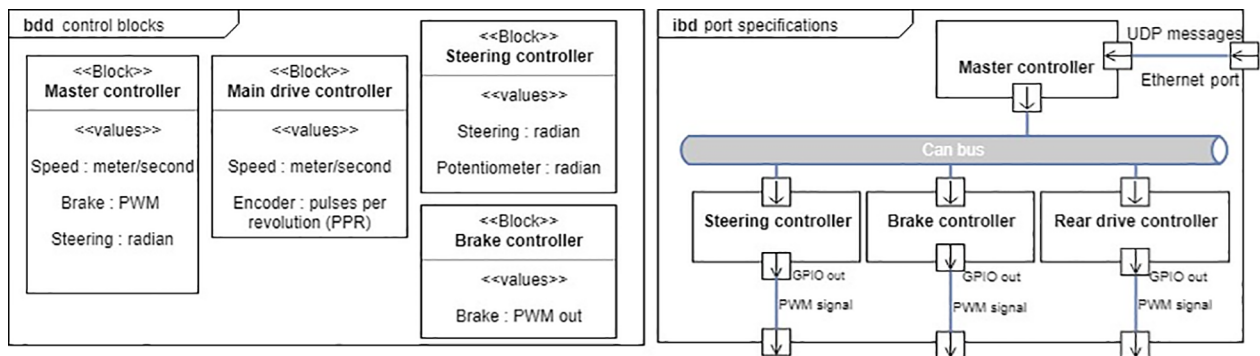


Fig. 3. General modular architecture of UGV.

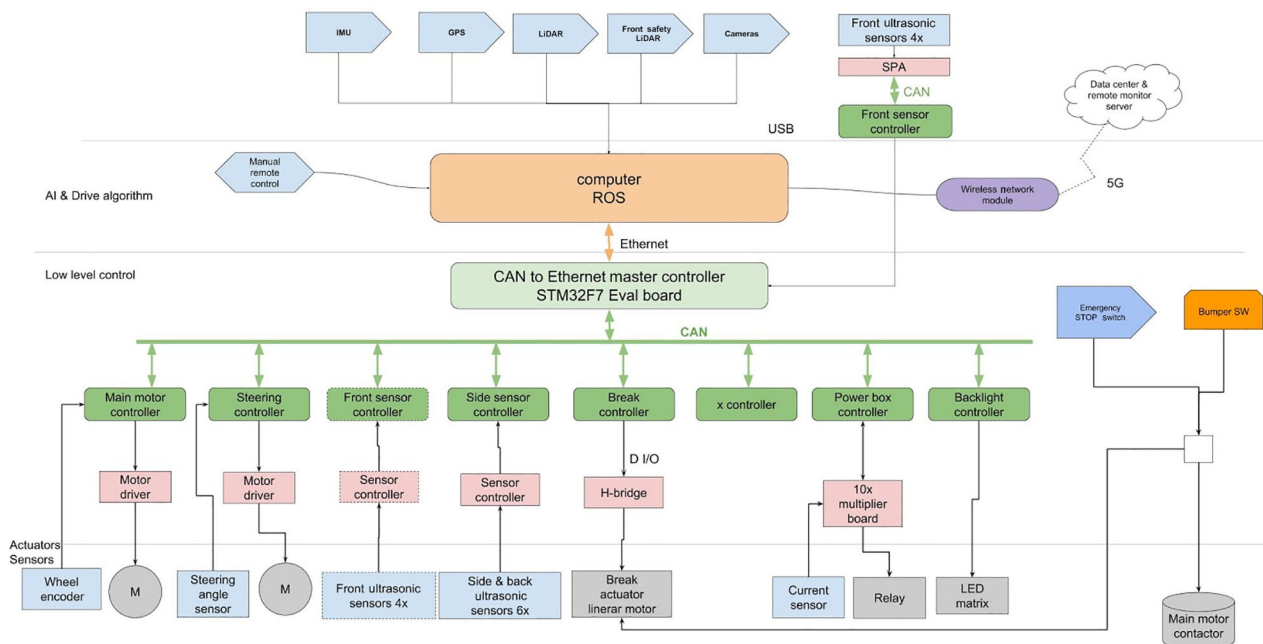


Fig. 4. Hardware architecture implemented on the UGV UKU.

computer with the CAN bus protocol and the controller’s task is to calculate the most optimal motor speed to reach the desired steering angle.

Control object, in this case, is H-bridge type motor driver, which is changing the polarity of the voltage based on the pulse width modulation (PWM) signals, which are generated from the controller. Feedback device for the steering motor is a simple analog sensor (potentiometer) due to its simplicity of integration and accuracy. Control process of the PID is shown in Fig. 5.

PID controller of both actuators was tuned and validated separately by step tests. This process clearly shows the dynamical characteristic of each actuator. Testing and fine-tuning of PID controller defines the overall performance of the mobile robot in the future and is therefore crucial. Figure 6 shows the most optimal test result for front steering. Setpoint of the experiment was steering angle input from the computer varying from  $-1.0$  to  $1.0$  radians, and potentiometer value as a controller feedback. This test case where the

practical method of [9] was followed shows step response for controller with an angle input from  $0.9$  rad to  $-0.9$  rad, which is one full rotation of the steering axle of the UGV. Proportional derivative (PD) action of the PID controller was enough to reach the optimal results for both steering and rear-wheel drive. The advantage of using only PD characteristics is rapid output and short time required to return process value to setpoint. PD formula can be seen in Eq. (1):

$$u = k_p + e + k_d \frac{d_e}{d_t}, \quad (1)$$

where  $u$  – static characteristics,  $k_p$  – proportional gain,  $k_d$  – derivative gain,  $e$  – error,  $d_e$  – change in error; and  $d_t$  – change in time. After tuning the PD parameters for steering motor we got the most optimal result using coefficient values:  $k_p = 0.1$  and  $k_d = 0.9$ . The graph shows that the steady state is reached after 8 seconds. This means that the robot will do one full turn of the

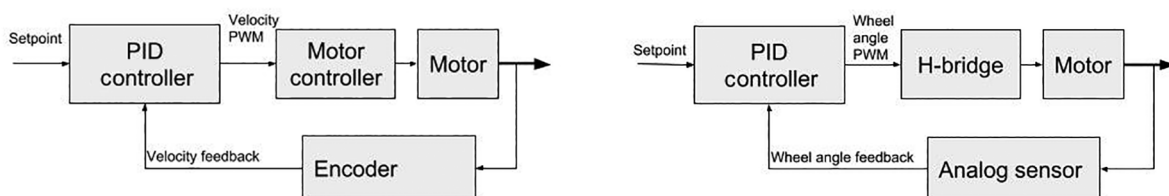


Fig. 5. Steering and main motor control diagram.



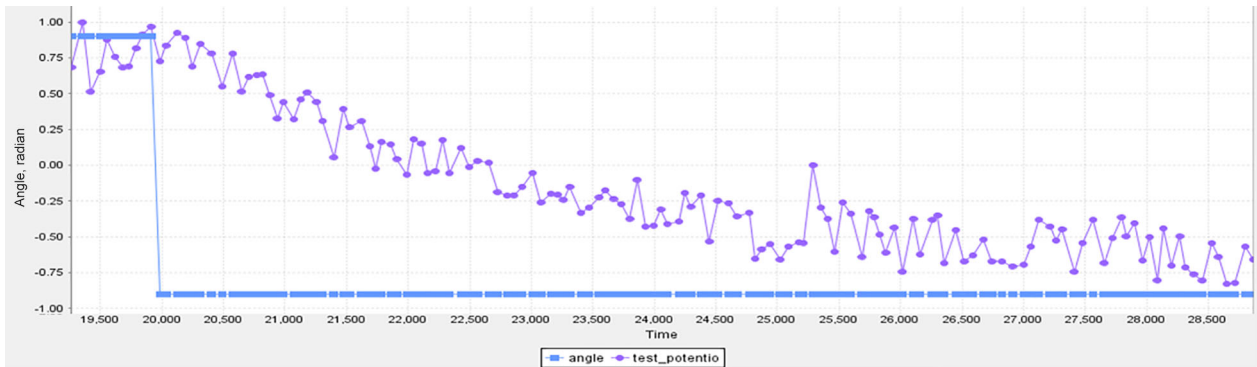


Fig. 6. PD tuning test results.

steering axle in 8.5 seconds, which is a satisfactory result for the research if we take into account the fact that the experiment was done on a standing vehicle.

### 7. SOFTWARE ARCHITECTURE

The high-level software architecture of the system, shown in Fig. 7, is based on the Robot Operating System (ROS). The reasons behind this decision were open-source drivers for multiple sensors and simplicity of integration of third-party software like Autoware, Yolo and multiple device drivers. According to the current architecture, the mobile robot takes several inputs from sensors. The global positioning system (GPS) is used for localization and path following, which is defined by a human. 2D (LiDAR), and camera inputs are used for obstacles' detection and safety. Output commands are steering angle, brake and linear velocity, which are sent to the low-level controllers over the UDP messages. Main software for computation of current architecture is Autoware [10], which is an open source library for

self-driving cars and thus has many advanced software capabilities like lane following, obstacle avoidance, traffic light detection, lane detection etc.

The ROS platform itself is based on high modularity and scalability due to its master/slave architecture. ROS communication protocol is based on the publish-subscribe method and therefore it allows us to use external libraries and run separate individual nodes that will easily interact with each other even on multiple platforms. The ROS is a middleware and operates well on multiple cross platforms however software architecture described in this section does not use ROS on lower level controllers due to its lack of real-time capabilities. To merge ROS and lower-level controllers, software bridge was built, which converts custom ROS messages to the UDP messages. Modularity of this particular high-level software architecture is mainly the result of key principles of ROS and its approach of implementing multiple software libraries as building blocks of the primary product.

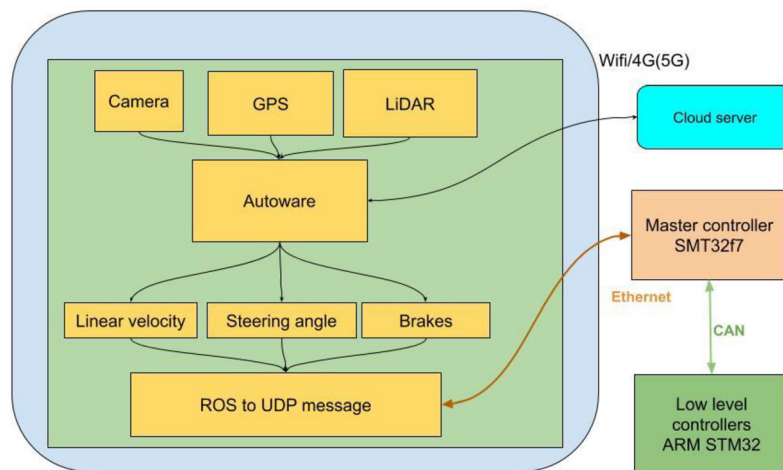


Fig. 7. Software architecture and message flow.

## 8. CONCLUSIONS

A modular architecture was proposed in this study to achieve flexibility and fast implementation process for off-road unmanned ground vehicles (UGVs). The architecture proposes hierarchical hardware and software structure and in particular, three-level hardware and two-level software architecture, which can be adapted according to the target platform specifics. System design and modular concept was implemented by taking into account an early stage mechatronic methodology proposed by [11]. The implementation example relies on the off-road universal robot platform UKU described in the first part of the paper. In addition, another similar robot platform is considered to reach comparable results of implementation. Both UGVs have the same size and similar purpose but they are different in their power and locomotion concept. The proposed concept for the hardware controller was evaluated in more detail using a test drive of UKU. Results showed that the proposed architecture guarantees the main driving requirements achieving stable navigation in dynamic and unknown environments, having fast and flexible implementation at the same time. The experiment of the implementation of proposed modular control system for UKU confirms that the proposed modular architecture can be easily implemented for similar unmanned robots, e.g. Agronaut. The future works include advancements in modular architecture, in particular, the high-level control algorithms including AI-based mission planning and navigation.

## ACKNOWLEDGEMENTS

The research was supported by the European Commission through the H2020 project Finest Twins (grant No. 856602). The publication costs of this article were covered by the Estonian Academy of Sciences.

## REFERENCES

1. Saska, M., et al. Navigation, localization and stabilization of formations of unmanned aerial and ground vehicles. In *Proceedings of the International Conference on Unmanned Aircraft Systems (ICUAS), Atlanta, GA, USA, May 28–31, 2013*. IEEE, 2013. <https://ieeexplore.ieee.org/document/6564767>
2. Ahmadzadeh, H., Masehian, E., and Asadpour, M. Modular Robotic Systems: Characteristics and Applications. *J. Intell. Rob. Syst.*, 2016, **81**(3–4), 317–357.
3. Wang, Y., et al. Building Unmanned Plant Factory with Modular Robotic Manipulation and Logistics Systems. In *Recent Developments in Intelligent Computing, Communication and Devices. Advances in Intelligent Systems and Computing, Vol. 752* (Patnaik, S. and Jain, V., eds). Springer, Singapore, 2019, 11–19.
4. Rassõlkin, A., Sell, R., and Leier, M. Development Case Study of the First Estonian Self-Driving Car, ISEAUTO. *Electr. Control Commun. Eng.*, 2018, **14**(1), 81–88.
5. Sell, R., Leier, M., Rassõlkin, A., and Ernits, J.-P. Self-driving car ISEAUTO for research and education. In *Proceedings of the 19th International Conference on Research and Education in Mechatronics (REM), Delft, Netherlands, June 7–8, 2018*. IEEE, 2018. <https://ieeexplore.ieee.org/document/8421793>
6. Rassõlkin, A., Hõimoja, H., and Teemets, R. Energy saving possibilities in the industrial robot IRB 1600 control. In *Proceedings of the 7th International Conference-Workshop Compatibility and Power Electronics (CPE), Tallinn, Estonia, June 1–3, 2011*. IEEE, 2011. <https://ieeexplore.ieee.org/document/5942236>
7. Väljaots, E., Sell, R., and Rimasauskas, M. Unmanned Ground Vehicle Energy Efficiency Validation in Territory Surveillance Mission. *Solid State Phenom.*, 2016, **251**, 164–170.
8. Väljaots, E., Lehist, H., Kiik, M., and Leemet, T. Soil sampling automation using mobile robotic platform. *Agron. Res.*, 2018, **16**(3), 917–922.
9. Rassõlkin, A. and Vodovozov, V. A test bench to study propulsion drives of electric vehicles. In *Proceedings of the 8th International Conference-Workshop Compatibility and Power Electronics (CPE), Ljubljana, Slovenia, June 5–7, 2013*. IEEE, 2013. <https://ieeexplore.ieee.org/document/6601169>
10. Kato, S., et al. Autoware on board: enabling autonomous vehicles with embedded systems. In *Proceedings of the 9th ACM/IEEE International Conference on Cyber-Physical Systems (ICCPs), Porto, Portugal, April 11–13, 2018*. IEEE, 2018. <https://ieeexplore.ieee.org/document/8443742>
11. Sell, R., Coatanéa, E., and Christophe, F. Important aspects of early design in mechatronic. In *Proceedings of the 6th International Conference of DAAAM Baltic, Industrial Engineering, Tallinn, Estonia, April 24–26, 2008*, 177–182. <http://innomet.ttu.ee/daaam08/Online/Design%20Engineering/Coatanea-Sell.pdf>

## Mobilsete robotite modulaarne arhitektuur

Raivo Sell, Eero Väljaots, Tengiz Pataraija ja Ehsan Malayjerdi

Modulaarne arhitektuur on võtmetähtsusega mobiilsete robotite arendusel, kui tegemist on multiotstarbelise robotisõidukiga. Artiklis on tutvustatud modulaarsuse kontseptsiooni keskklassi mobiilsele robotile, kus peamine fookus on tarkvaraline modulaarsus, isorganiseeruvus koos lihtsalt ühendatavate lisaseadmetega, andurite väljundite kombineerimine ja riistvaraline modulaarsus. Väljatöötatud kolmetasandiline tarkvara raamistik sisaldab vahevara ja seda on rakendatud kahe erineva mobiilse roboti juhtsüsteemides.