EMBEDDED CONTROL SYSTEM FOR MOBILE ROBOTS WITH DIFFERENTIAL DRIVE

Michal KOPČÍK, Ján JADLOVSKÝ
Department of Cybernetics and Artificial Intelligence, Faculty of Electrical Engineering and Informatics,
Technical University of Košice, Letná 9, 042 00 Košice, Slovak Republic, Tel.: +421 55 602 4218,
E-mail: michal.kopcik@tuke.sk, jan.jadlovsky@tuke.sk

ABSTRACT

This article deals with design and implementation of control system for mobile robots with differential drive using embedded system. This designed embedded system consists of single control board featuring ARM based microcontroller which control the peripherals in real time and perform all low-level motion control. Designed embedded system can be easily expanded with additional sensors, actuators or control units to enhance applicability of mobile robot. Designed embedded system also features build-in communication module, which can be used for data for data acquisition and control of the mobile robot. Control board was implemented on two different types of mobile robots with differential drive, one of which was wheeled and other was tracked. These mobile robots serve as testing platform for Fault Detection and Isolation using hardware and analytical redundancy using Multisensor Data Fusion based on Kalman filters.

Keywords: *embedded system, mobile robot, motion control, microcontroller, fault detection and diagnostics*

1. INTRODUCTION

Mobile robotics these days experiences huge expansion in many different fields such as service robotics, military, education or recreation. There are many different types of mobile robots, like wheeled or tracked, walking robots or flying drones.

Mobile robots with wheels and tracks are still the most widespread among other types, mainly because relatively simple control system and very effective and precise performance of motion. These mobile robots are used in service robotics as home appliances (robotics vacuum cleaners), in industry they are used as robotics storekeepers (Kiva robots) or in education (Khepera).

There are some open source mobile robotics platforms such as e-puck [1], or mobile robots based on low-cost Arduino modules, their disadvantage that they are either designed for specific chassis or they require multiple modules to fulfill basic needs to control mobile robot. Many existing types of mobile robots uses distributed control architecture, where low level control is performed using microcontrollers and high level control is covered by computers with operating system.

The main motivation for this work was to design universal control system for small mobile robots with differential driven chassis which can be easily expanded with additional sensors such as Inertial Measurement Units (IMU), magnetometers or different types of distance measuring sensors. Huge advantage is ability to expand designed system with minicomputer such as Raspberry Pi to perform more computing power demanding task such as image processing or Simultaneous Localization and Mapping (SLAM).

Primary the mobile robots described in this article was designed for the purpose to test Fault Detection and Identification (FDI) algorithms based on hardware and analytical redundancy, where measures from different sensors and mathematical models are fused together using Multi Sensor Data Fusion (MSDF) algorithms based on weighted averaging or Kalman filter (KF). Similar approach of FDI with application in mobile robotics was described in [2]. Some other applications are maze mapping and shortest path finding using image processing, line following or supervisory control.

For designed embedded electronics the Robot Operating System (ROS) communication package was created, which is capable to control motion of the robot and can read multiple parameters from mobile robot. This ROS package was successfully tested on remote computer connected via Bluetooth and on minicomputer Raspberry Pi placed directly on the mobile robot.

The first part of this paper deals with design of the control system for mobile robots with differential chassis based on embedded system with 32-bit ARM microcontroller. Second part describes Diagnostic System, which will be implemented in mobile robot control system software in future work. The last part is devoted to sample applications that use described mobile robots with control system in several different applications.

2. STRUCTURE OF MOBILE ROBOT CONTROL SYSTEM

Whole mobile robot consists of electronics (microcontroller, drivers and sensors), mechanics (chassis, actuators and wheels or tracks) and control system (program for microcontroller). Given parts may vary depending on specific application. In this chapter we present mobile robots developed in our Department of Cybernetics and Artificial Intelligence, Faculty of Electrical Engineering and Informatics, Technical University of Košice.

Block diagram of sample mobile robot with its internal connections is shown in Fig. 1, where essential parts are Control board and mechanics and optional parts are sensor board and minicomputer Raspberry Pi.

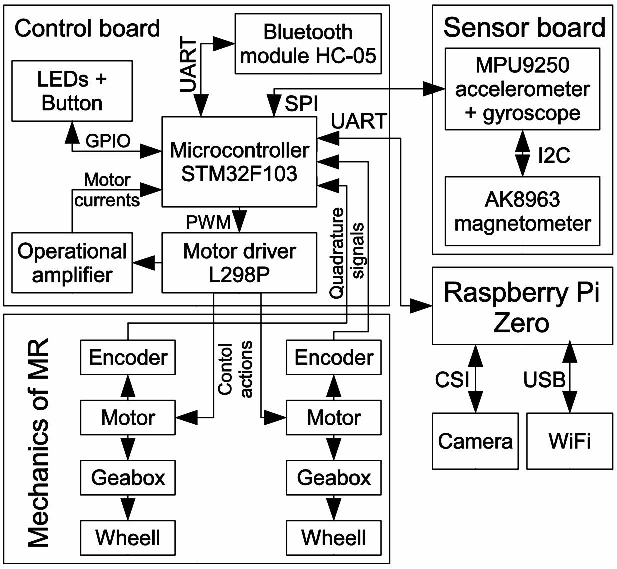

Fig. 1 Block diagram of mobile robot control system with extensions

2.1. Electronics

Electronics consists of single double sided Printed Circuit Board (PCB) with ARM based microcontroller STM32F103. This microcontroller has 32-bit ARM M3 core with clock frequency up to 72 MHz. It features multiple high speed counters with automatic quadrature signal decoders that are used to measure speed of the motors equipped with incremental encoders. Given microcontroller can be programmed and debugged using Serial Wire Debug (SWD) interface, using Joint Test Action Group (JTAG) interface or using bult-in bootloader.

As motor driver there is L298P dual full H-bridge integrated circuit, with shunt resistors for current sensing. Sensed voltages from shunt resistors are amplified using operational amplifiers with RC low pass filter to eliminate noise caused by PWM input signal. The motor driver is capable of driving motors with continuous current up to 2 A when using appropriate heat sink.

On this board there is communication Bluetooth module HC-05, which serves to two purposes. One purpose is establishing connection with computer or smartphone to control mobile robot and second purpose is to wirelessly upload program to microcontroller.

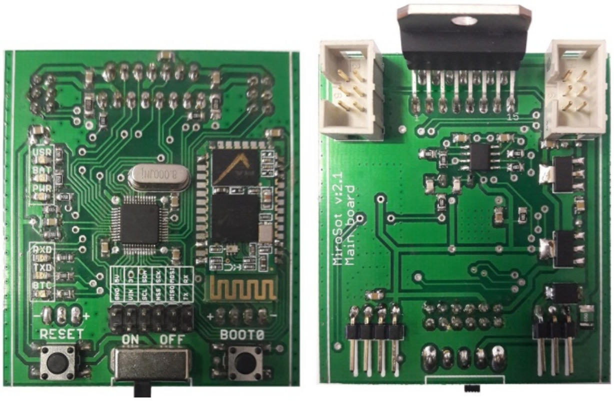

Fig. 2 Image of designed electronics of mobile robots

This circuit board also features expansion connector which supplies power and communication interfaces to daughter boards. Communication interfaces available through this expansion connector are Serial Peripheral Interface (SPI), Inter Integrated Circuit (I2C) and Universal Asynchronous Receiver and Transmitter (UART). Picture of mobile robot control board is shown in Fig. 2.

2.2. Mechanics of mobile robots

Designed electronics has been tested with two different chassis, one with wheels and one with tracks. In both types of robots we used the same type of Lithium Polymer (Li-Pol) battery with nominal voltage of 7.4 V and capacity 850 mAh. Both types of mobile robots are shown in Fig. 3.

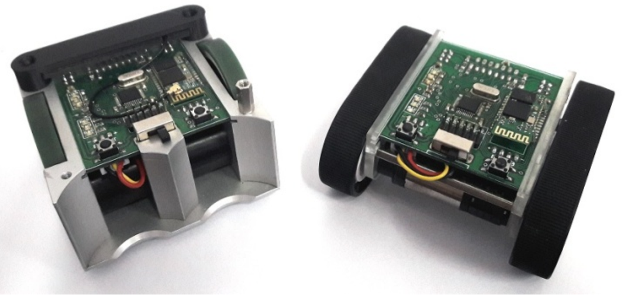

Fig. 3 Mobile robots MiroSot (left) and TrackBot (right) without any extensions

2.2.1. Mobile robot for robotic soccer of MiroSot category

It is wheeled mobile robot with dimensions 75 x 75 x 50 mm and weight about 400 g. As actuators there are two DC Faulhaber micromotors 2224U006SR with built-in high resolution encoders IEH2-4096. Chassis consists of aluminium frame joined together using machine screws. The shaft of the motor is directly connected to the wheel using spur gears, where one gear is on the shaft of the motor and second one is fitted in wheel of the robot itself from the inside.

With given encoders, wheels and gear ratio, the robot has resolution of 724 pulses per millimeter (ppm) and can reach top speeds about 2.5 m/s.

2.2.2. Tracked mobile robot TrackBot

TrackBot is small tracked robot with dimensions about 90 x 90 x 40 mm. Robot weights approximately 300 g and features two ESCAP 16 DC motors with built-in encoder and planetary gearbox. Shaft of the motors is directly connected to the drive wheels. As tracks we used POLOLU-1415 rubber tracks.

This mobile robot has resolution 119 ppm, top linear speed is about 400 mm/s and angular speed is about 8 rad/s.

2.3. Mobile Robot Control System (MRCS)

MRCS implemented in microcontroller was divided into several function parts. Whole control system is written in native C language designed especially for microcontrollers, but with some minor changes this MRCS can be implemented under Windows or Linux machine.

Whole mobile control system is executed and managed using interrupts from timers and peripherals and some functionality is managed using Direct Memory Access (DMA) to relieve computational power when transferring large amount of data. On Fig. 4 is shown conceptual diagram of whole MRCS.

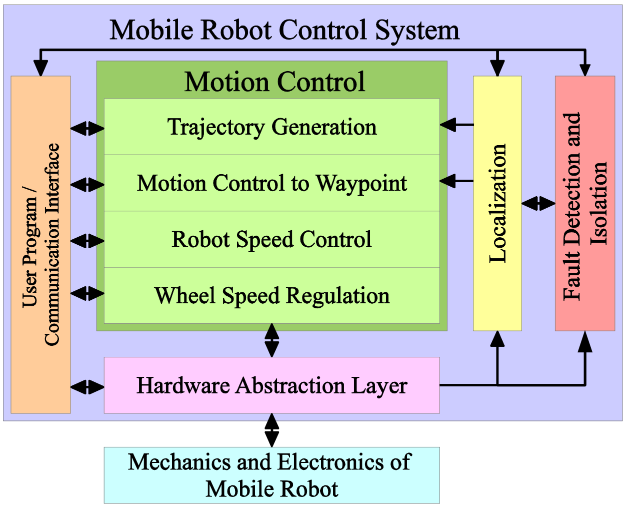
Mobile Robot Control System
Motion Control
Trajectory Generation
Motion Control to Waypoint
Robot Speed Control
Wheel Speed Regulation
User Program / Communication Interface
Localization
Fault Detection and Isolation
Hardware Abstraction Layer
Mechanics and Electronics of Mobile Robot

Fig. 4 Conceptual diagram of structure and interconnections between modules of mobile robot control system

2.3.1. Hardware Abstraction Layer (HAL)

This part of control system serves as bridge between hardware of mobile robot and other parts of control system. Its purpose is to relive programmer of the higher level program from the need to know exact structure of electronics or given microcontroller.

HAL performs all necessary measurements from sensors and initializes necessary peripherals. It executes time critical functions from other parts of the MRCS like computation of wheel speed regulators, motion control, readout of data from sensors or communication interface.

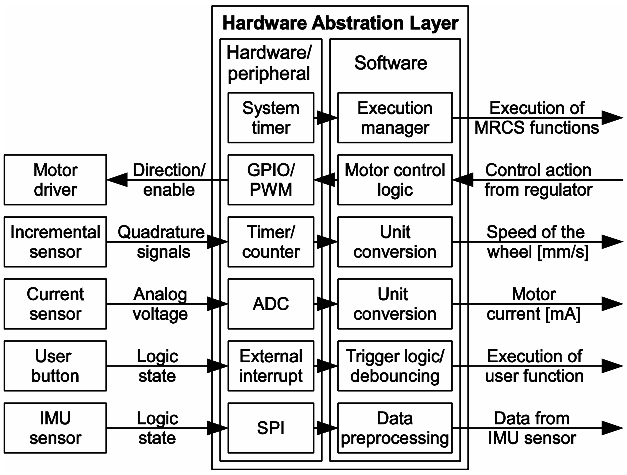

Fig. 5 Block diagram of Hardware Abstraction Layer

2.3.2. Motion control

This part of control system handles all motion control of mobile robot from regulation of the speed of wheels (tracks) to generation of waypoints. It consists of several parts (layers) where every higher layer depends on functionality of lower layers. Whole motion control system was described in more detail in [3].

Wheel speed regulation

This layer performs feedback regulation of the speed of wheels. Wheel speed regulation consists of two nested feedback PSD controllers, where one is used to control current flowing through armature of the motor and second one control overall speed of the wheel.

The PSD controllers can be easily enhanced with feedforward component as described in [4]

Robot speed control

This layer is used to translate linear and angular velocities of the robot to speed of the wheels using kinematic model of the robot. It also handles acceleration and deceleration in angular and linear direction.

Motion control to waypoint

Motion of the robot is based on set of waypoints generated by user or higher layer of motion control. There are two types of waypoint. First is for linear motion, where the robot follows straight line between two waypoints and second is for rotation along the axis of robot. This layer also features function which smoothly switches between reference lines, when there are multiple waypoints set to linear motion, to ensure jerk free motion.

Trajectory generation

The highest level of motion control is used to generate set of waypoints based on input parameters of curve. To generate waypoints we use Bézier curves, where input parameters are actual position and speed and desired end location and speed. From these parameters there is generated set of waypoints, which are passed lo lower level of motion control system.

2.4. Communication interface

This part of firmware is used as a gateway between MRCS and other systems like Robot Operating System (ROS), or custom control program. Using communication interface other system have access to all functionality of mobile robot starting from motion control to all data from sensors.

Packets have form of strings which represents data and commands with variable length. Transmitted data packets from control system can be requested or unrequested. Requested data packet follows every received command packet and unrequested data packets are generated on events or periodically using system for subscribing of parameters. Data packets generated on events have three levels. First levels of events are messages with information about achieved waypoints or press of the user button. Higher two event levels carry information about non critical and critical errors in system, like discharged battery or communication bus problem.

As default as interface is used UART connected to PC via Bluetooth module, but this interface can be redirected to other peripheral or module capable of transceiving data.

Communication interface features function used for subscribing desired parameters like position, data from sensors or control actions. These subscribed parameters are then sent via communication interface with defined period of time. This is main tool for collecting of the data, which are then used for offline or online data analysis.

2.5. User program

There is program space reserved for user program, where user can control robot using string commands just like control via communication interface, or directly by functions from firmware libraries of control system, User

When writing user program, programmer doesn't have to worry about timings or delays of this program, because all control system functionality is executed in parallel with user program using interrupts of microcontroller.

2.6. Localization

For the purpose of feedback motion control, motion control system needs to know position of robot in plane or space. In designed MRCS there is currently implemented relative localization based on odometry, where information from incremental sensors transformed to relative position using kinematic model of the robot.

Obtained position is however burdened with error, which increases with any motion op the robot. This localization error is caused by slippage between robot wheels and surface.

To minimize this error, we are planning to enhance mobile robots with sensors of relative and absolute motion, specifically gyroscope, accelerometer and magnetometer based on Micro-Electro-Mechanical System (MEMS). Information from these sensors and from odometry will be processed using MSDF to obtain more precise relative position as position gained only from odometry.

3. DIAGNOSTIC SYSTEM (DS)

Diagnostic system is based on hardware and analytical redundancy, where information from multiple sensors and mathematical models are put together using MSDF.

Proposed diagnostic system is currently under testing on data measured from sensors and other relevant parts of MRCS. Diagnostic system was described in more detail in [5]. Whole diagnostic system is spreaded over multiple parts of MRCS, especially HAL, FDI and localization.

3.1. Data preprocessing

This part of DS is used to prepare data from individual sensors in the way of collecting, transforming and filtering. Collecting of the data means receiving data from physical sensor using peripheral of microcontroller. This is done in HAL of MRCS. Transforming of the data is calculation of the raw data to physical units based on calibration data.

This part is also used to generate analytical redundancy based on mathematical models of mobile robot and its system. Example of this analytical redundancy is calculation of linear and angular speed of the robot based on current flowing through armature of DC motors.

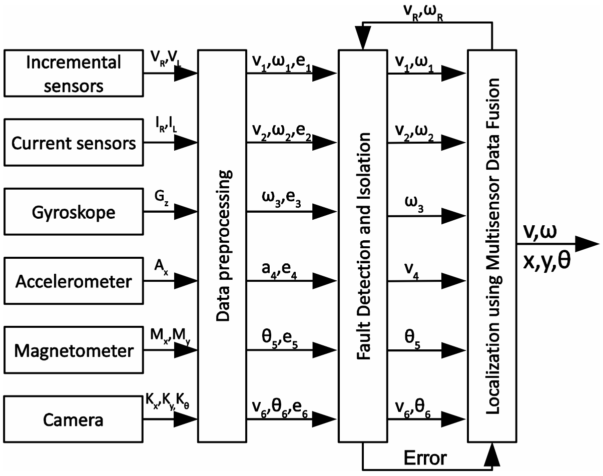

Fig. 6 Block diagram of proposed diagnostic system

Data preprocessing is also first stage of FDI by capturing faults of data caused by communication bus problems and intermittent single measure faults in data. Some of the functionality is handled by HAL, like unit conversion and data acquisition using peripherals of microcontroller.

Sample block diagram of data preprocessing for current sensors to linear and angular speed conversion is shown in Fig. 7, where LPF stands for Low Pass Filter and Units conversion block is used to transform voltages measured by Analog to Digital converters to currents.

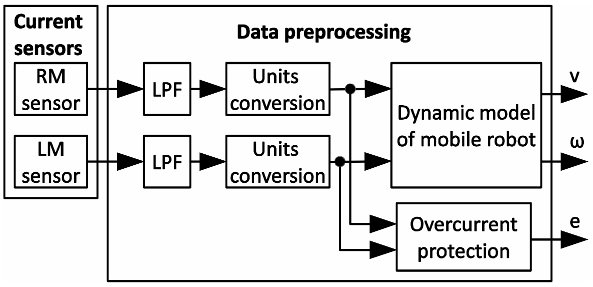

Fig. 7 Block diagram of sample data pre-processing

3.2. Fault Detection and Isolation

FDI consists of residual generation, which is done by combining of the data using MSDF. For each sensor there is independent residual generator that combines output from all other sensors using KF, whose output is subtracted from the output of given sensor.

This type of FDI is also capable of identifying unusual or unwanted states of the system like skid of the robot or bump to the obstacle. These unusual states may show similar signs as faults of the sensors, some of them can be distinguished by taking in count information from multiple residuals. For example error of the magnetometer can be mistaken with unusual state caused by magnetic field from metal object or permanent magnet placed near mobile robot.

All residual are processed using simple expert system to determine actual state of the sensors and mobile robot.

Simplified block diagram of designed FDI for four sensors is shown in Fig. 8, where $\omega_{1\text{-}4}$ are measures from sensors after data preprocessing, $e_{1\text{-}4}$ are error attributes from data preprocessing, $r_{1\text{-}4}$ are residuals and E is information about faults and state of the system. There is separate FDI for linear and angular speed.

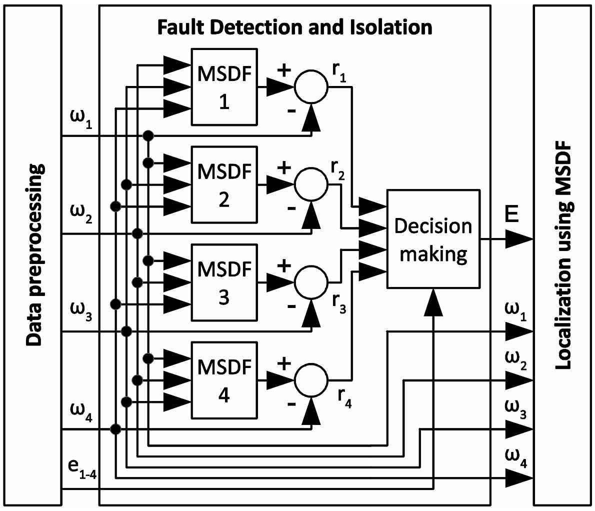

Fig. 8 Block diagram of Fault Detection and Isolation

3.3. Localization using Multisensor Data Fusion

There are several ways how to fuse data from multiple sensors to form one more precise and robust result. In our case we decided to use KF like in the FDI part. MSDF is sensitive to errors of the input data, but in this case the final MSDF uses information about errors and state of the system from FDI part, to take inputs only from correct sources. This is done by selecting appropriate matrix from database of steady gain Kalman matrices calculated for every state of the system. MSDF with application in mobile is described in [6].

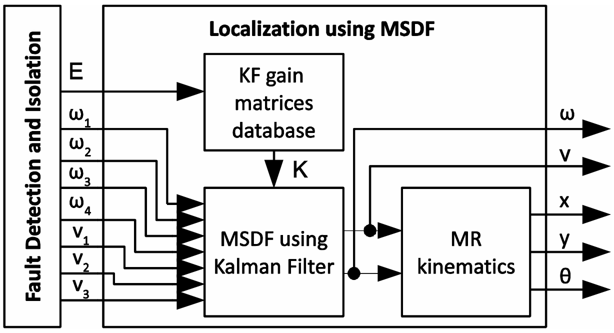

Fig. 9 Block diagram of localization system using MSDF

4. SAMPLE APPLICATIONS

Proposed MRCS was implemented and tested in multiple applications, which are part of bachelor and diploma theses and semestral projects. In this chapter we present some of them.

4.1. Mobile robotic soccer of category MiroSot

Mobile robot soccer of category MiroSot is competition where two teams of small mobile robots with restricted dimensions and weight compete against each other. The whole system consists of three, five or ten mobile robots per team (depending on league), camera placed above gaming pitch and central computer.

Each player has color sticker with team color and other colors for identifying of the robots and their position. Central computer runs ROS with several nodes to control robots. Main nodes are in this application are image processing node to determine positions of all robots, strategy planning node to calculate reference trajectories for robots, motion control node to control motion of mobile robots and communication node to transfer commands between computer and robots. Image processing and MiroSot category was described in [7].

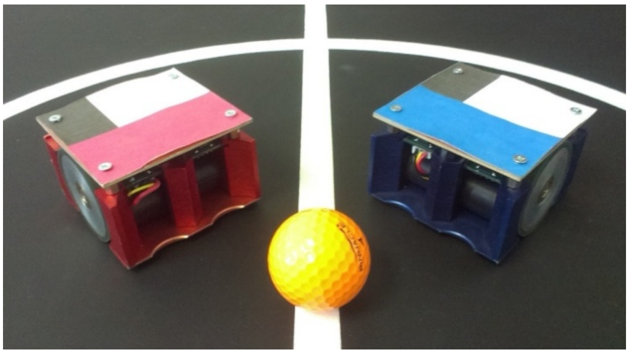

Fig. 10 Mobile robots in robotic soccer application

4.2. Maze solving

For the purpose of maze solving, the standard MiroSot mobile robot was enhanced by minicomputer Raspberry Pi and camera. Main control board of the robot communicates with Raspberry Pi using UART interface, bypassing Bluetooth module. Raspberry Pi is used for image processing based on output from camera and for controlling of the motion of the robot. Output from image processing algorithm is information about relative position of the robot in current cell of maze and presence of nearby walls. From relative position based on image processing and information about position from mobile robot, the correction of robots position is calculated and sent back to the MRCS. This way the error of the localization calculated by the MRCS is eliminated. Information about nearby walls from image processing is passed to maze solving algorithm that decides which way the robot should go and sends relative coordinates directly to motion control system of robot as waypoints. For maze solving we implemented and tested Flood Fill algorithm, which is described in more detail in [8].

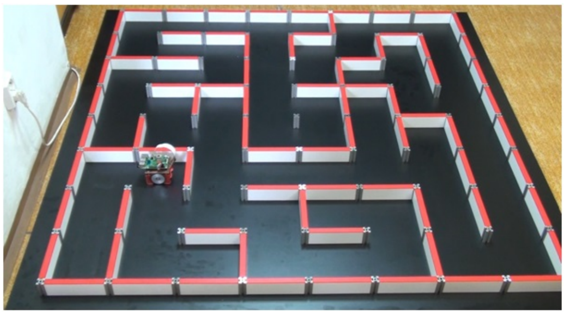

Fig. 11 Mobile robot in maze solving application

4.3. Line following using camera

In this application was used TrackBot mobile robot with Raspberry Pi zero and camera. The line is detected using image processing on images from camera. To follow line, robot uses algorithm, which determines distance of the line from the center of the robot and calculates linear and angular velocities, which are sent to MRCS.

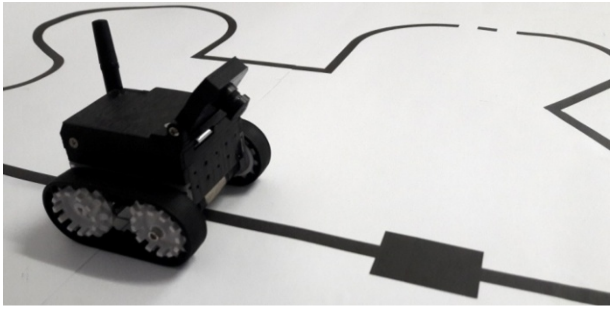

Fig. 12 TrackBot in line following application

5. CONCLUSIONS

In this article we presented control system for mobile robots with differential drive, using embedded electronics with 32-bit ARM based microcontroller. Proposed mobile robot control system was implemented and tested in various applications using two types of chassis with minimal software changes. In this article the Fault Detection and Isolation and MultiSensor Data Fusion was proposed, which will be implemented in control system in the future work.

ACKNOWLEDGMENTS

This publication arose thanks to the support of the Operational Programme Research and development for the project "Centre of Information and Communication Technologies for Knowledge Systems" (ITMS code 26220120020), co-financed by the European Regional Development Fund (20%), by the Research and Development Operational Program for project: University Science Park Technicom for innovative applications with knowledge technology support – 2nd phase, ITMS code 313011D232, co-financed by the ERDF (40%), grant TUKE FEI-2015-33: Research Laboratory of Nonlinear Underactuated Systems (30%) and also this publications also arose thanks to grant KEGA - 001TUKE-4/2015 (10%).

REFERENCES

[1] MONDADA, F. et al.: The e-puck, a Robot Designed for Education in Engineering, *Proceedings of the 9th Conference on Autonomous Robot Systems and Competitions*, Castelo Branco, Portugal, 2009, vol. 1, num. 1, pp. 59–65.

[2] STERGIOS, S. I. – SUKHATME, G. S. – BEKEY, G. A.: Sensor fault detection and identification in a mobile robot, *Proceedings of the 1998 IEEE/RSJ Intl. Conference on Intelligent Robots and Systems*, Victoria, B.C., Canada, October 1998, pp. 1383–1388, ISBN 0-7803-4465-0.

[3] JADLOVSKÝ, J. – KOPČÍK, M.: Distributed Control System for Mobile Robots with Differential Drive, *2016 Cybernetics & Informatics (K&I)*, Levoča, 2016, pp. 1–5, ISBN 9781-5090-1832-1.

[4] JADLOVSKÝ, J. – KOPČÍK, M.: Basic Motion Control of Differential-Wheeled Mobile Robot ALFRED, *Advances in Intelligent Systems and Computing. - Switzerland : Springer*, 2014, Vol. 316, 2015, pp. 73–80, ISSN 2194-5357.

[5] KOPČÍK, M.: Diagnostics of Sensors and Actuators Within Distributed Control System, *SCYR 2015 - 15th Scientific Conference of Young Researchers: Proceedings from conference*, Herľany, 2014, pp. 214–215, ISBN 978-80-553-2130-1.

[6] KOPČÍK, M.: Multisensor data fusion for differential wheeled mobile robots, *SCYR 2014 - 14th Scientific Conference of Young Researchers: Proceedings from conference*, Herľany, 2015, pp. 223–226, ISBN 978-80-553-1714-4.

[7] JADLOVSKÝ, J. – VARGA, M. – KOPČÍK, M.: Image Processing for Localization of Mobile Robots, *Electrical Engineering and Informatics 6: Proceedings of the Faculty of Electrical Engineering and Informatics of the Technical University of Košice*, 2015, pp. 574–577, ISBN 978-80-553-2178-3.

[8] JADLOVSKÝ, J. – KOPČÍK, M. – SEGIŇÁKOVÁ, S.: Introduction into maze mapping and the shortest path finding, *Electrical Engineering and Informatics 6: Proceedings of the Faculty of Electrical Engineering and Informatics of the Technical University of Košice*, 2015, pp. 621–625, ISBN 978-80-553-2178-3.

Received March 2, 2017, accepted July 11, 2017

BIOGRAPHIES

Michal Kopčík was born on 17.1.1989. In 2013 he graduated (MSc) with distinction at the Department of Cybernetics and Artificial Intelligence of the Faculty of Electrical Engineering and Informatics at Technical University of Košice. Since September 2013 he works as PhD. student at the Department of Cybernetics and Artificial Intelligence. His scientific research is focusing on mobile robotics in connection to embedded systems, fault detection and isolation and multisensor data fusion. He also participates in various research projects.

Ján Jadlovský works at the Department of Cybernetics and Artificial intelligence of Technical University of Košice as a Assoc prof. He is a graduate of Technical University of Košice, Faculty of Electrical Engineering. In terms of pedagogy he focuses on the issues of proposal and implementation of distributed systems that control production processes. In his science-research based activities he is oriented towards distributed control systems, image recognition, complex functional diagnostics of single purpose regulators, diagnostics of production control systems, creation of information and control systems with application of the latest information technology. He is a chief executive of company KYBERNETIKA, s.r.o., Košice, that is oriented towards design engineering, implementation and operation of production and diagnostic systems in the electrical engineering, mechanical and metallurgical production.