23rd ABCM International Congress of Mechanical Engineering
December 6-11, 2015, Rio de Janeiro, RJ, Brazil

# LOW-COST EMBEDDED COMPUTER FOR MOBILE ROBOT PLATFORM BASED ON RASPBERRY BOARD

**Vitor Akihiro Hisano Higuti**
**Henry B. Guerrero**
University of São Paulo, Av. Trabalhador São-carlense, 400, São Carlos, São Paulo, Brazil.
vitor.higuti@usp.br, h_borrelo@sc.usp.br

**Andrés Eduardo Baquero Velasquez**
**Renan Moreira**
**Livia Martinelli Tinelli**
**Daniel Varela Magalhães**
**Marcelo Becker**
University of São Paulo, Av. Trabalhador São-carlense, 400, São Carlos, São Paulo, Brazil.
andresbaquero@sc.usp.br, renanmoreira@usp.br, tinelli@sc.usp.br, daniel@sc.usp.br, becker@sc.usp.br

**Juan Fajardo Barrero**
Universidad de los Llanos, Km. 12 Vía Puerto López, Villvicencio - Meta, Colombia.
jfajardo@unillanos.edu.co

**Débora Marcondes Bastos Pereira Milori**
Embrapa Instrumentation, Rua XV de Novembro, 1452, São Carlos, São Paulo, Brazil.
debora.milori@embrapa.br

***Abstract.*** *Nowadays embedded computers are becoming more popular and available for everybody. Due to this, computational costs may be reduced. Based on this, we decided to evaluate the use of a Raspberry Pi B+ as the main control unit for an experimental mobile platform. In order to carry out the tests we developed a controller for a differential drive mobile robot to achieve wall-following navigation which will keep it at a desired distance to the wall nearby. The locomotion of the mobile robot is supported on the actuation of two DC Maxon motors which are managed by two EPOS2 boards. A LIDAR sensor (Laser Imaging Detection and Ranging) has been implemented with the aim of measuring the distance to wall. First, all device drivers were configured for this work under Raspbian operational system. Then, a fuzzy controller was implemented in C++ for the wall-following strategy. Finally, we conducted tests in indoor environment. The results enlightened the influence of the parameters in the fuzzy sets with regards to overshoot and settling time. Also, they demonstrated the potential of the mobile robot for further use in control strategy learning.*

***Keywords:*** *wall-following controller, fuzzy logic controller, differential drive mobile robot*

## 1. INTRODUCTION

Embedded systems have been widespread used where a computer is needed since the 1970s. But they were seen simply as small computers for most of their history. Thus the main engineering concern was to deal with limited resources (limited processing power, limited energy sources, small memories, etc.). Just recently, it is noted that the principal challenge arises from the interaction with physical processes (Lee and Seshia, 2014). Nowadays,

Investing effort in the deployment of embedded systems on mobile robots became highly attractive with the availability on the market of small-size computers with not only high processing capabilities but also with General Purpose Input/Output (GPIO). For instance, the Raspberry Pi B+ is a credit-card sized powerful computer. Initially, the Raspberry Pi was projected with the aim of motivating children to learn computer science using a low cost computer reachable for a school budget and hence avoiding the requirement of high cost computers (Globo News, 2012).

Hardware resources available on the Raspberry PI B+ board allowed its application on the control of a differential drive mobile robot. The main difficulty to command the exposed prototype consisted on the requirement to interchange information with two EPOS2 24/5 boards and a Laser Range Finder which usually was done using at least a home laptop because it was necessary to use computer software and libraries. According to requirements, the research group has found that the Raspberry Pi B+ offered the same capabilities as a laptop. Hence, this paper is intended to expose about the usage of a Raspberry PI B+ as a low-cost embedded computer for a mobile robot platform. Also description of the robot prototype is provided.

Section 2 provides a basic description of the devices and about the kinematics of the mobile robot. Section 3 focuses on the fuzzy controller development. Then section 4 presents and discusses the experimental results. Finally, conclusions, acknowledgments and references are presented.

## 2. GENERAL DESCRIPTION OF MOBILE ROBOT

The prototype structure was built with the aim of supporting three planar platforms (two magenta acrylic layers and one black wood layer) such that they carry diverse components and devices that were embedded in the mobile robot. Fundamentally the prototype structure is 0.3 m long, 0.27 m width and 0.33 m high.

Figure 1 shows the components present in the prototype: (1) Raspberry Pi B+, (2) Wi-Fi dongle, (3) EPOS2 board, (4) remote control toy car's wheel, (5) caster wheel, (6) power supply unit and (7) 12V battery. And Fig. 2 presents a bottom view with (8) Hokuyo URG-04LX laser rangefinder and the (9) DC motors manufactured by *maxon motors Company*.

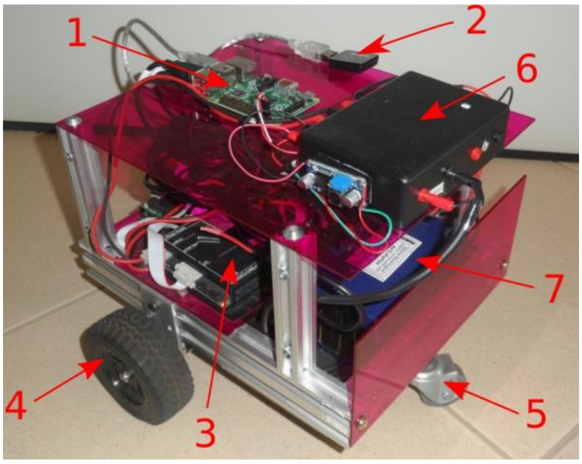


Figure 1. Mobile robot main components

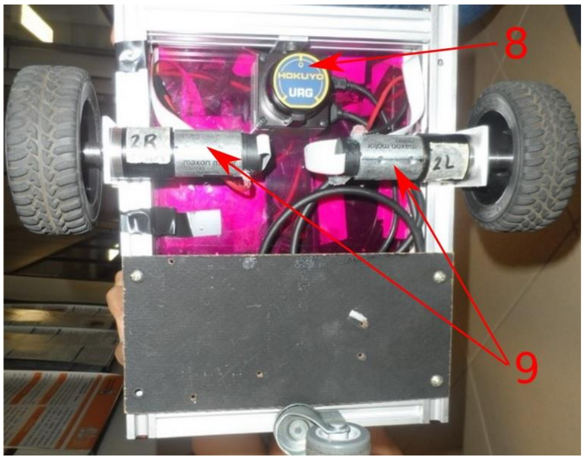


Figure 2. Mobile robot's bottom view

### 2.1 Locomotion System

The locomotion system of differential-drive mobile robot consists of two *motor-gearbox-encoder* sets manufactured by *maxon motors* Company. The chosen motor is a 22 W DC motor REmax29 (reference 226802) which has to be supplied with 12 V and has two outputs shafts (rear and front) each. In order to track the motor speed (rpm) and the rotation direction, the rear shaft is coupled to an incremental encoder (reference 225805). And the front shaft is mounted to the gearbox GP 32C (reference 166938) for a reduction of 33:1. The output shaft of the gearbox is connected to the wheel. The interaction between the Raspberry Pi B+ with the motors occurs by two EPOS2 24/5 Controller. A block diagram of the locomotion system is presented in Fig. 3.

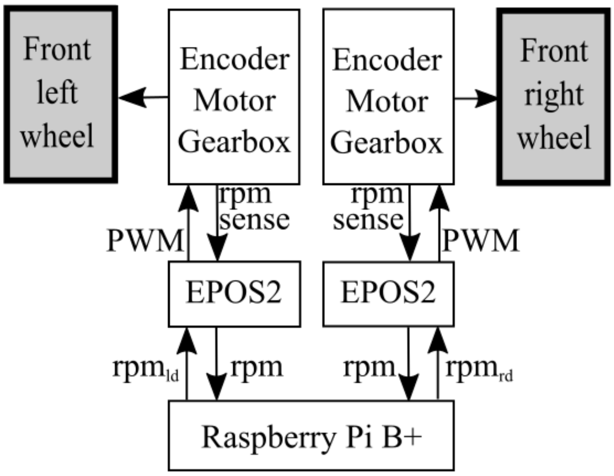


Figure 3. Block diagram of the locomotion system

#### 2.1.1 Maxon EPOS2 24/5 Board and Motor Set

Figure 4. EPOS2 24/5 board

Figure 5. URG-04LX

Figure 6. Raspberry Pi B+

The EPOS2 24/5 board (Fig. 4) is a small-sized, full digital, smart motion controller. Due to its flexible and high efficient power stage, the EPOS2 24/5 drives brushed DC motors with digital encoder as well as brushless EC motors with digital Hall sensors and encoder. EPOS2 24/5 has integrated functionalities to work on position, velocity and current control allowing sophisticated positioning applications. The configuration of EPOS2 24/5 allows its control as a slave node such that it drives directly the connected DC motor obeying the desired conditions for velocity or position coming from a master computer via any USB or RS232 built-in interface (maxon motors ag, 2013).

Besides the Linux library obtained from maxon product website, the usage of EPOS2 boards with Raspberry Pi also required the installation of FTDI drivers, available in FTDI Chip website. The initial configuration steps followed those available in (Amato *et al.*, 2012). Furthermore, a compact library was generated for EPOS handling: functions to prepare, open and close the board communication, and to drive the motor with velocity parameters.

### 2.2 Hokuyo URG-04LX Laser Rangefinder (LRF)

The Hokuyo URG-04LX (Fig. 5) is an LRF categorized as an Amplitude Modulated Continuous Wave (AMCW) sensor. The description of its working principles can be found in (Okubo *et al.*, 2009). With an angular resolution of 0.36°, a rotating mirror sweeps the laser beam horizontally over a range of 240°. The scan rate is about 0.1 s as the mirror rotates at about 600rpm. The data transfer rate can be configured at 19.2, 57.6, 115.2, 250, 500, or 750 Kbps when connected to an RS-232C port, or it can be set at 12 Mbps when connected via USB. Therefore, the USB connection is chosen for the data transfer in out tests. The LRF has a guaranteed accuracy distance range between 0.060 and 4.095 m. The quoted measurement error is ±0.01 m for distances of less than 1 m. For greater distances, the error is quoted as ±2%, assuming a target of 0.070×0.070 m white Kent paper (Hokuyo Automatic Co, 2008).

The URG-04LX can be configured and read using the library available in (Fukuda *et al.*, 2014). For this application, the 683 distance readings that cover 240° full scan are stored into an array every 0.1 s. From this array, just one element was used as the measured distance to a lateral reference.

### 2.3 Raspberry Pi B+

The Raspberry Pi is a small low cost credit-card sized computer designed initial and specifically for education but providing interesting resources for embedded systems. Raspberry Pi Model B+ (Fig. 6) is based around an ARM processor which can run the open source Linux operating system well known as Raspbian which is a Debian-based Linux distribution specifically designed for the Raspberry Pi. Further description, including the specification of the Raspberry Pi B+ board, can be found in (Sachdeva and Katchii, 2014).

### 2.4 Differential Drive Kinematics

The presented mobile robot uses the drive mechanism known as differential drive (Dudek and Jenkin, 2010). Therefore each wheel is independently driven either forward or backward by its own electric motor. The wheels have a radius $R_w$ of 0.098 m and the distance L between their centers is 0.293 m. The midpoint between the wheels along y axis defines a point M. Left and right wheel velocities in x-axis are $v_l$ and $v_r$, respectively. Thus they rotate with angular velocities $\omega_l = -v_l / R_w$ and $\omega_r = v_r / R_w$. Figure 7 shows the positive direction for wheel velocities ($v_l$ and $v_r$), translational velocity of the point M ($v$), rate of rotation ($\omega$) that the mobile robot performs about the ICC – *Instantaneous Center of Curvature*, and signed distance R from point M to the ICC.

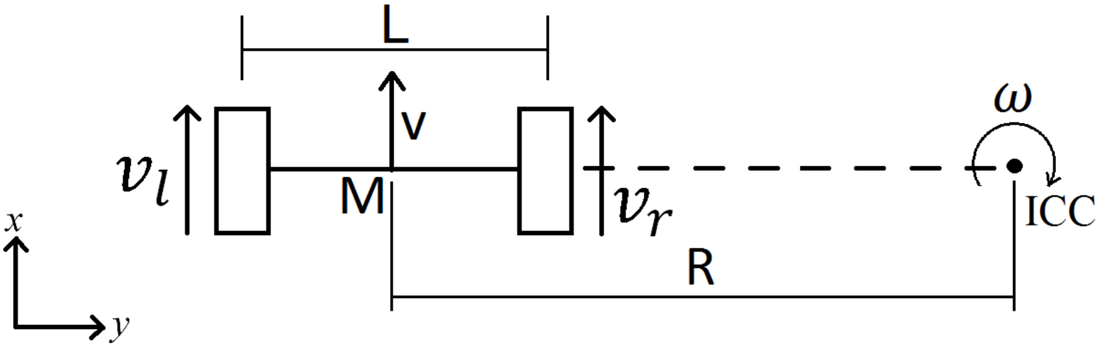


Figure 7. Differential drive kinematics

The following differential drive kinematics' equations were derived from (Dudek and Jenkin, 2010):

$$R = 0.5 * L * (\omega_l - \omega_r) / (\omega_l + \omega_r) \quad (1)$$

$$v = R_w * (\omega_l - \omega_r) /2 \quad (2)$$

From (Dudek and Jenkin, 2010), an infinite R makes the robot to move in a straight line. Other values of R lead to curved trajectories and in the case of zero R, the robot rotates about a point midway between the two wheels. Thus it is expected that smaller values of R leads faster variations in global y-axis over time.

With the goal to control motors to follow a specific trajectory, we must know the angular velocities $\omega_l$ and $\omega_r$ that will achieve a desired translational *v* and a signed distance R to the ICC. Thus rearranging the Eq. (1) and Eq. (2):

$$\omega_l = v*(R + 0.5*L)/(R*R_w) \quad (3)$$

$$\omega_r = -v*(R - 0.5*L)/(R*R_w) \quad (4)$$

And we obtain Eq. (3) and Eq. (4), which were further used for implementation purpose. After these values are converted from International System units to revolutions per minute, they can be sent as arguments of the functions that communicate with EPOS2 board.

## 3. CONTROLLER DEVELOPMENT

In order to assess the capabilities of the proposed mobile robot, a wall-follower fuzzy logic controller was developed and implemented. The choice of fuzzy controller (Sivanadam *et al,* 2007) is based on three factors: first, it is an initial work with the main goal of assessing the hardware and software setup; there is little knowledge about the robot's kinematics and dynamics model; and finally a fuzzy controller can take advantage of team members' previous experience with mobile robotics. Figure 8 shows the general description of the system to achieve wall-following. The LRF sensor URG 04LX provides the distance (d) between the robot and a lateral reference. The distance (d) is sent to a Raspberry Pi B+ which is the control unit. This board provides the needed rotational speeds to each EPOS2 board which will drive the motor.

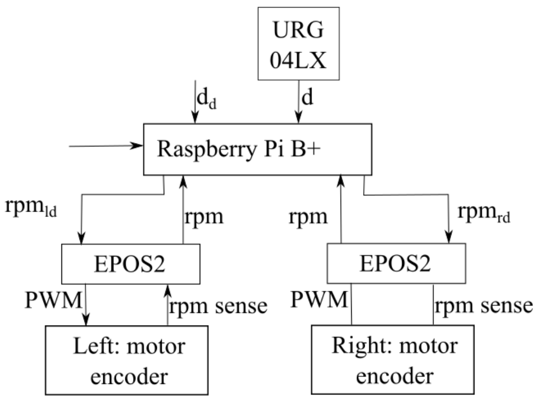


Figure 8. Data flow between mobile robot's components

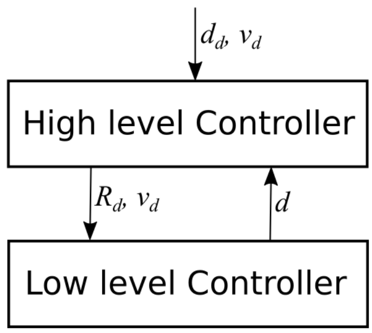


Figure 9. Low and High Level Controllers' Diagram

According to Fig. 9, the system can be divided in two parts: the high-level controller and the low-level controller. The first is the wall-follower controller itself which receives the desired distance to lateral reference $d_d$ and desired translational speed $v_d$ from user, and a feedback of the current distance $d$ to the right wall from the low-level controller. Then it outputs the desired distance $R_d$ from midpoint of the wheels common axis to the ICC and the desired translational speed $v_d$. And the low-level controller assumes these outputs from high-level controller in order to drive actuators using the embedded EPOS2 boards.

### 3.1 Wall-following controller

For the validation of the platform, a wall-follower was developed and experimented in indoor environment. Figure 10 depicts the conducted experiments: while in forward movement with constant speed $v$, the mobile robot should keep itself at a desired distance to the right wall. This is achieved giving the required signals to each one of the locomotion system's motors based on the distance measurements at angle θ supplied by the LRF sensor.

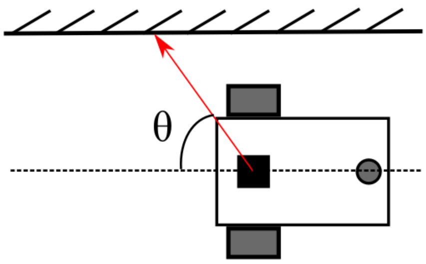


Figure 10. Wall-following strategy

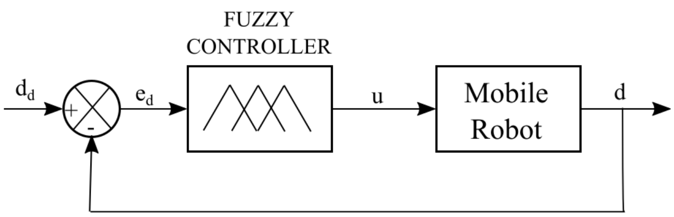


Figure 11. Block Diagram of the High-level controller

Figure 11 shows a basic closed-loop control system block diagram, which represents the wall-following controller. An error detector comparator receives two signals: desired and current distance. With them, the comparator is able to generate the error signal ($e_d$) that enters the fuzzy controller. Its output is a control signal (u) which is applied to the plant, mobile robot itself.

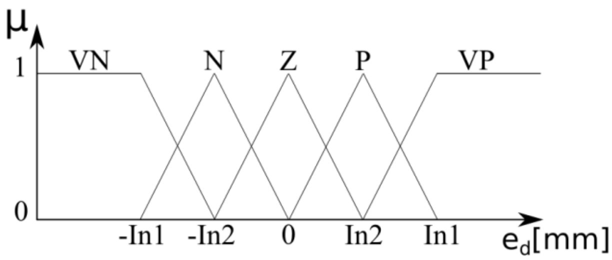


Figure 12. Input fuzzy set

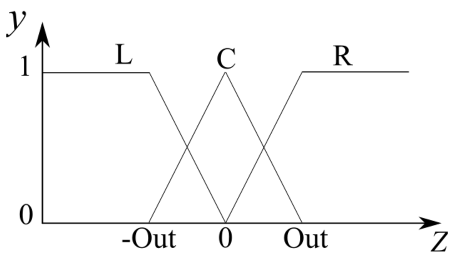


Figure 13. Output fuzzy set

The fuzzy controller's implementation follows (Guerrero *et al.*, 2014; Velasquez *et al.*, 2014). It has an input set with five triangular pertinence functions and an output set with three triangular pertinence functions as shown in Fig. 12 and Fig. 13. As different sets were tested, the parameters *In1, In2* and *Out* are used for their definition. Dealing with the error signal ($e_d$) in the horizontal axis in Fig. 12, the pertinence may fall into at least one of the five input sets, named Very Negative (VN), Negative (N), Neutral (Z), Positive (P) and Very Positive (VP), and generate two values $\mu(e_d)$. According to rules defined in Eq.5, the position of $e_d$ in the input set defines where the values μ cut in the output set, comprised by three sets named Left (L), Central (C) and Right (R).

If $e_d$ = VN or if $e_d$ = N, then μ = L
If $e_d$ = Z, then μ = C (5)
If $e_d$ = P or if $e_d$ = VP, then μ = R

Finally, the centroid method is used to calculate the defuzzified output Z*. As discussed in section 2.4, smaller values of R lead faster turns while the robot moves in a straight line for infinity R. Thus it is required an inverse relationship between Z* and $R_d$. In general, three responses are expected:

If $Z^* < Out$, then turn left
If $-Out < Z^* < Out$ then go ahead (6)
If $Z^* > Out$, then turn right

## 4. EXPERIMENTAL RESULTS

The mentioned wall-following controller was implemented using C++ routines. For this set of experiments, the desired distance to keep is 0.5 m with respect to the right wall. Considering θ the angle between the robot's longitudinal axis and LIDAR reading, as the right wheel blocks a θ = 90° LIDAR reading, the sine projection of θ = 60° reading is used. The experimentation site is an indoor environment as shown in Fig. 14. The track is approximately 6 m length. The wall is made of seven 1.05 m x 1.2 m white boards divided by six 0.019 m width black stripes. From previous experience, the initial values for fuzzy parameters were chosen as *In1* = 100, *In2* = 10 and *Out* = 1. With these values, we expected an average response in terms of overshoot and settling time.

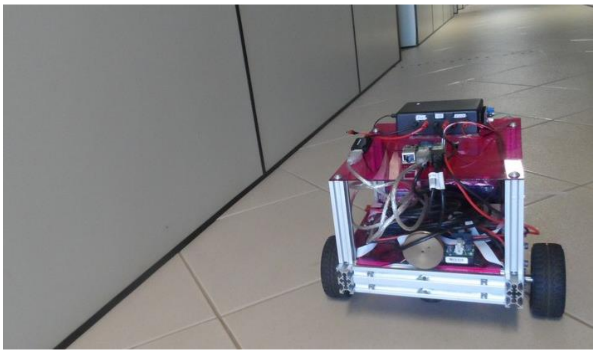

Figure 14. Experimentation site

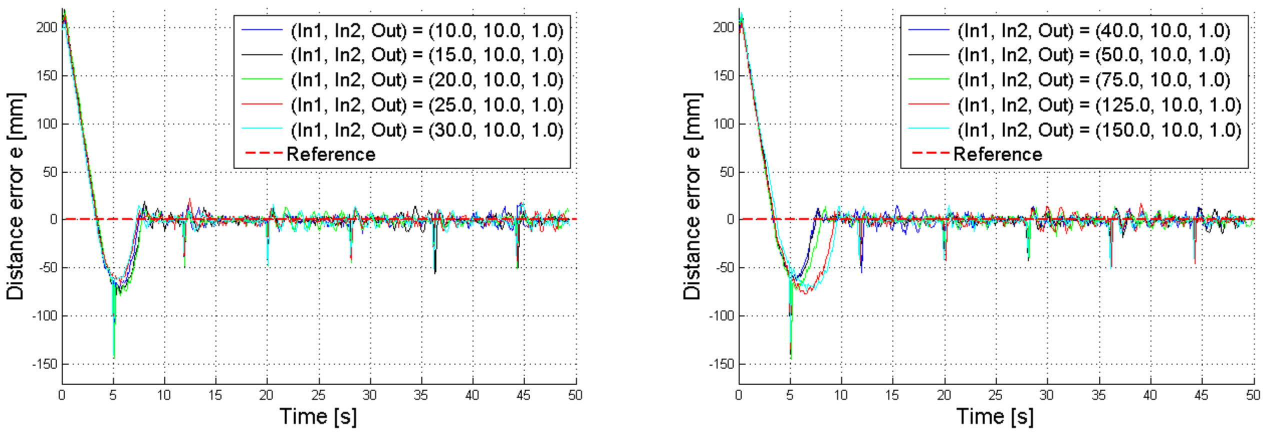


Figure 15. Experiments changing parameter *In1* in the input fuzzy set

Figure 15 depicts several tests with different values for *In1* while *In2* and *Out* were constant and equal to 10 and 1.0, respectively. As it can be seen, overshoot is around 0.07 m and settling time about 8 s, and little changes happens as *In1* increases until 75. After that, the overshoot and settling time increases significantly.

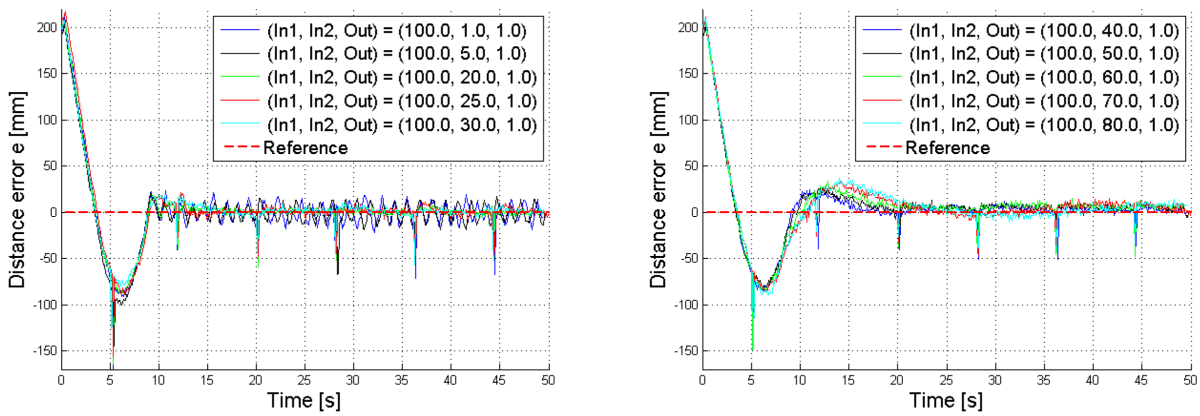


Figure 16. Experiments changing parameter *In2* in the input fuzzy set

Figure 16 shows the influence of *In2*. Hence In1 and Out were kept constants at values, respectively, 100 and 1.0. Smaller values of *In2*, such as 1 and 5, have more oscillatory responses. On the other hand, bigger values of *In2*, such as 70 and 80, have slower settling time and increased undershoot. Moreover, a low frequency oscillatory response also appears.

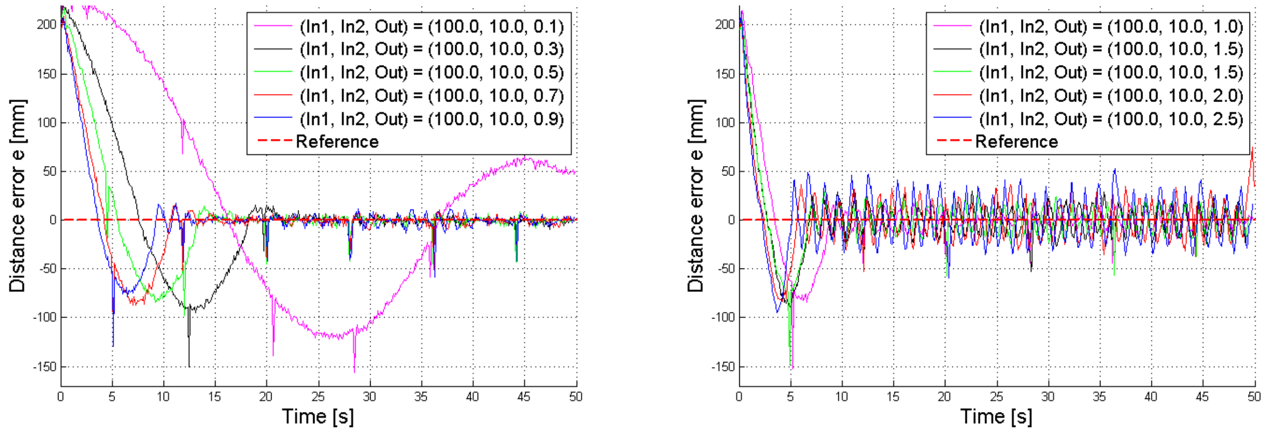


Figure 17. Experiments changing parameter *Out* in the output fuzzy set

Finally, some experiments with different responses varying the parameter *Out* in the fuzzy output set and with a fixed fuzzy input set. From Fig. 17, we can verify that the settling time decreases as *Out* is increased. But, for values bigger than 1, the oscillatory response is clearly present.

It is important to explain the existence of misreading in all plots at times around 5 s, 12 s 20 s, 28 s, 36 s and 44 s. As presented in (Kneip *et al.*, 2009), besides the color, the reading is also dependent on the material properties, such as the surface brightness, and the incidence angle, which is non-zero. The combination of the two led to missing functional compliance and faulty measurements. Further discussion is not the purpose of this work.

## 5. CONCLUSIONS AND FUTURE WORKS

The mobile robot prototype presented in this paper was successfully assembled, both in hardware and software perspective. Picking components as maxon motors, EPOS boards and Hokuyo LIDAR sensor reduced considerate efforts in handling the hardware and ensuring that it would work the expected way. The challenge itself remained in the task of integrating all devices under Raspberry Pi B+, which we accomplished adapting default libraries from manufacturers and C++ programming.

Without further time-consuming modelling of the vehicle, a simple differential drive kinematics was used to relate naturally high-level signals (translational speed and distance to ICC) with low-level signals (rotational speed of each of the motors). As the goal is the evaluation of the Raspberry Bi+ as main unit control, experimental analysis was not conducted to evaluate the correctness of this model. Nevertheless the tests demonstrated the expected behavior and thus the model will be further used as a starting point for studies with differential drive mobile robots. And that relationship between signals was made with fuzzy controller, another tool which facilitates a quick implementation.

For this platform and the employed fuzzy sets, it can be expected that higher values of input parameters *In1* and *In2* have more impact on settling time. But changes are much more perceptible when output parameter *Out* is increased: the frequency of oscillatory behavior decreases while overshoot and settling time visibly increases. Besides enlightenment about fuzzy controller usage, experiments also showed no interference from the Raspberry Pi B+ acting as the main unit control.

An unexpected result was the interference of the black stripes in the experimentation site. For further indoor tests, a completely white wall in length may be required to avoid such disturbance. Nevertheless, the analysis of the Hokuyo URG 04-LX in outdoor environment, like in the middle of crops, may be an incoming topic of study as the small size of the mobile robot makes it appealing to initial outdoor tests, where the principles of algorithms can be analyzed and developed before moving on to usually bulky outdoor mobile robots.

According to the shown experimental reached results, the exposed prototype can be considerate as a learning platform to tackle deeper researches with regard to navigation and control of mobile robot. Thus a next step consists onto the definition of the kinematic model, enabling us to implement classic and modern control strategies applied on mobile robots like exposed, still for the wall-following task. Furthermore, Raspberry Pi has GPIO pins which allow the connection of 3.3 V devices directly to the board. Examples may be GPS and Inertial Measurement Unit (IMU) modules. Therefore the board itself is suitable for further use in learning mobile robotics in indoor environment and with low budget.

## 6. ACKNOWLEDGEMENTS

Authors would like to thank CNPq and CAPES for the financial support which enabled the construction of the mobile robot. Also thanks are given to COLCIENCIAS (PhD Scholarship), Universidad de los Llanos, EESC USP and Embrapa Instrumentation for the financial and technical support to maintain the research team in this work.

## 7. REFERENCES

AMATO, J., ANDERSON, J., CARLONE, T., FAGAN, M., PADIR, T., STAFFORD, K. and MILLER, B., 2012. “Worcester Polytechnic Institute’s 2012 RASC-AL Exploration Robo-Ops Competition Team”. Jan 2015 < https://wpirover.files.wordpress.com/2011/09/maxon-ubuntu-instructions2.pdf>.

DUDEK, G. and JENKIN, M., 2010. *Computational principles of mobile robotics*. Cambridge university press, New York, 2nd edition.

FUKUDA, H., AOKI, T., FUJIMOTO, J., KIMOTO, K., KRISTOU, M., NAGATA, M., MUSASHI, M., MOMODORI, T. and WADA, F., 2014. “Welcome to URG Network”. 02 Feb. 2015 <http://sourceforge.net/p/urgnetwork/wiki/Home/>.

GLOBO NEWS, 2012. “Cientistas criam computador para ensinar programação às crianças”. 02 Feb. 2015 <https://www.youtube.com/watch?v=8_xVCB68msY>.

GUERRERO, H. B., VELASQUEZ, A. E. B., BARRERO, J. F., CÔCO, D. Z., RISARDI, J. C., MAGALHÃES, D. V. and BECKER, M., 2014. “Orientation (Yaw) Fuzzy Controller Applied to a Car-Like Mobile Robot Prototype”. In *Proceedings of the IEEE 5th Colombian Workshop on Circuits and Systems - CWCAS2014*. Bogotá, Colombia.

HOKUYO AUTOMATIC CO, 2008. “Scanning Laser Range Finder URG-04LX: Specifications”. 02 Feb. 2015 <https://www.hokuyo-aut.jp/02sensor/07scanner/download/pdf/URG-04LX_UG01_spec_en.pdf>.

KNEIP, L., TÂCHE, F., CAPRARI, G. and SIEGWART, R., 2009. “Characterization of the compact Hokuyo URG-04LX 2D laser range scanner”. In *Proceedings of the IEEE International Conference on Robotics and Automation – ICRA2009*. Kobe, Japan.

LEE, E. A. and SESHIA, S. A., 2014. *Introduction to Embedded Systems – A Cyber-Physical Systems Approach.* Lee & Seshia, Edition 1.5.

MAXON MOTORS AG, 2013. “EPOS2 P 24/5 Programmable Positioning Controller: Hardware Reference”. 02 Feb. 2015 < http://www.maxonmotor.com/medias/sys_master/8815046721566/378308-Hardware-Reference-En.pdf >.

OKUBO, Y., YE, C. and BORENSTEIN, J., 2009. “Characterization of the Hokuyo URG-04LX Laser Rangefinder for Mobile Robot Obstacle Negotiation”. *Unmanned Systems Technology XI*, Vol. 7332.

SACHDEVA, P. and KATCHII, S., 2014. “A Review Paper on Raspberry Pi”. *Journal of Current Engineering and Technology*, Vol. 4, No.6, p.3818-3819 – INPRESSCO2014.

SIVANANDAM, S., SUMATHI, S., DEEPA, S., 2007, *Introduction to Fuzzy Logic using MATLAB*. Springer-Verlag, Berlin, Germany.

VELASQUEZ, A. E. B., GUERRERO, H. B., BARRERO, J. F., TIBERTI, A. J., IBARRA, J. C. P., BECKER, M., MAGALHÃES, D. V., RISARDI, J. C., 2014. “Mobile robot prototype used to the study of the trajectory control strategies and auto-localization”. In *Proceedings of the VIII National Congress of Mechanical Engineering – CONEM2014.* Uberlândia, Brazil.

## 8. RESPONSIBILITY NOTICE

The authors are the only responsible for the printed material included in this paper.