Journal of King Saud University – Engineering Sciences (2011) **23**, 131–138

King Saud University

Journal of King Saud University – Engineering Sciences

www.ksu.edu.sa
www.sciencedirect.com

TECHNICAL NOTE

A semi-autonomous mobile robot for education and research

Waheid Gharieb Ali

Electrical Engineering Department, College of Engineering, King Saud University, P.O. Box 800, Riyadh 11421, Saudi Arabia

Received 24 May 2009; accepted 7 April 2010
Available online 5 April 2011

KEYWORDS

Mobile robots;
Microcontroller applications;
ATmega32;
I2C bus

Abstract This paper presents the development, implementation, and testing of a semi-autonomous robotic platform, which may potentially be used for educational and research purposes. Educational purposes include: teaching the student how to design a stable electromechanical platform, exploring different types of sensors to navigate around any obstacles, interfacing different electronic components to a microcontroller, and demonstrating how to program the microcontroller chip in order to control a robotic platform. Research purposes include: developing and investigating the performance of different control algorithms to achieve behaviour analysis and obstacles avoidance. A modular hardware design is implemented using I2C bus to interface different sensors and motor drivers to the ATMEL microcontroller chip (AVR ATmega32). The hardware is integrated in one application board as embedded system design. The software is developed using C-compiler (ImageCraft) and a top-down approach is adopted to design different software modules. Experimental results are given to demonstrate the potential of the developed hardware and software modules.

1. Introduction

Mobile robotics is a discipline that is concerned with designing the hardware and software of such that the robots are able to perform their task in the presence of noise, contradictory and inconsistent sensor information, and possibly in dynamic environments. Mobile robots may be remotely controlled, guided by specially designed environments (beacons, bar codes, induction loops, etc.) or fully autonomous, i.e. independent from any links to the outside world (Nehmzow, 2006). Mobile robots are widely used in industrial applications, including transportation, inspection, exploration, surveillance, health care (Dubowsky et al., 2000; Rentschler et al., 2003), entertainment robots or even museum tour guides (Graf, 2001; Graf et al., 2004). What makes them interesting to scientific applications is the fact that they close the loop between perception and action, and can therefore be used as tools to investigate task achieving (intelligent) behaviour. The behaviour of a mobile robot is not the result of the robot's programming alone, but results from the interaction of three fundamental components:

- The program running on the robot (the "task").
- The physical hardware of the robot (the way its sensors and motors work, battery charge, etc.).

E-mail addresses: wahied@ksu.edu.sa, wahied@hotmail.com
URL: http://faculty.ksu.edu.sa/wahied

Peer review under responsibility of King Saud University.
doi:10.1016/j.jksues.2011.03.007

Production and hosting by Elsevier

- The environment itself (how visible objects are to the robot's sensors, how good the wheel grip is, etc.).

The robot's behaviour emerges from the interaction between these three fundamental components (see Fig. 1). The robot's behaviour will change if the robot's hardware is changed, or if the control program (the task) is changed, or if the environment is changed. For example, an unsuccessful wall following robot can be changed into a successful one by either changing the robot's sensors, by improving the control code, or by placing reflective strips on the walls! The fundamental principles that govern this interaction between robot, task and environment are, at the moment, only partially understood. For this reason it is currently not possible to design mobile robot controllers off line, i.e. without testing the real robot in the target environment, and fine tuning the interaction through trial and error. One aim in mobile robotics research is to analyze the interaction between robot, task and environment quantitatively, to gain a theoretical understanding of this interaction which would ultimately allow off-line design of robot controllers, as well as a quantitative description of experiments and their results. Therefore, the development in this paper aims to highlight the hardware and software implementation to design an autonomous platform. It serves to achieve research purposes such as:

- Developing software modules to achieve different tasks and obstacle avoidance.
- Investigating the interaction between the robot-task-environment to improve the system performance.

This platform serves also to achieve educational purposes such as:

- Teaching the student how to design a stable electromechanical platform.
- Helping the student to know the technology for different sensors and actuators that used in a mobile robot design such as: IR detectors, ultrasonic sensors, digital compass, and DC motor derivers.
- Giving the opportunity to learn how to program the microcontroller chip to acquire sensors data and react autonomously with environment in order to perform pre-specified tasks.

The remainder of the paper is organized as follows: Section 2 presents the specifications and practical considerations in the mechanical model design. Section 3 includes the details for the electrical design and implementation. Section 4 involves the functional description for the developed software modules for different tasks. Section 5 is dedicated for testing and evaluation. Finally, Section 6 presents the discussion and conclusions.

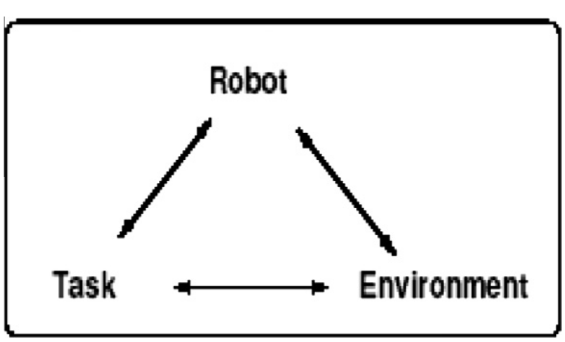

Figure 1 Robot-environment interaction (Nehmzow, 2006).

2. Mechanical design

The robot platform is made from a wood with dimension of 40 cm length and 30 cm width. The robot moves in a corridor of width 240 cm. Therefore, the selected dimension of the mobile base is suitable to install sensors to interact with the environment. The wheel selection has to provide sufficient traction and stability for the robot to cover all of the desired terrain, and enable sufficient control over the velocity of the robot. There are four major wheel classes, as shown in Fig. 2:

(a) Standard wheel with two degrees of freedom (rotation around the motorized axle, rotation around the contact point).
(b) Castor wheel with two degrees of freedom (rotation around the offset steering joint, rotation around the contact point).
(c) Swedish wheel with three degrees of freedom (rotation around the motorized wheel axle, rotation around rollers, rotation around the contact point).
(d) Ball or spherical wheel that can spin along any direction.

They differ widely in their kinematics, and therefore the choice of wheel type has a large effect on the overall kinematics of the mobile robot. The standard wheel and the castor wheel have a primary axis of rotation and are thus highly directional. To move in a different direction, the wheel must be steered first along a vertical axis. The key difference between these two wheels is that the standard wheel can accomplish this steering motion with no side effects, as the center of rotation passes through the contact patch with the ground, whereas the castor wheel rotates around an offset axis, causing a force to be imparted to the robot chassis during steering. Conventionally, static stability requires a minimum of three wheels, with the additional caveat that the center of gravity must be contained within the triangle formed by the ground contact points of the wheels. Stability can be further improved by adding more wheels, although once the number of contact points exceeds three, the hyper static nature of the geometry will require some form of flexible suspension on uneven terrain (Siegwart and Nourbakhsh, 2004).The adopted approach to design the mobile platform is to use differential derive wheel as shown in Fig. 3.

The differential drive design has two motors mounted in fixed positions on the left and right side of the robot, independently driving one wheel each. Since a minimum of three contact points to ground are necessary, this design requires one or two additional passive caster wheels or sliders, depending on the location of the driven wheels. Differential drive is mechanically simpler than the single wheel drive, because it does not require rotation of a driven axis. However, driving control for differential drive is more complex than for single wheel drive, because it requires the coordination of two driven wheels (Braeunl, 2006). Fig. 3 demonstrates the driving actions of a differential drive robot. If both motors run at the same speed, the robot drives straight forward or backward, if one motor is

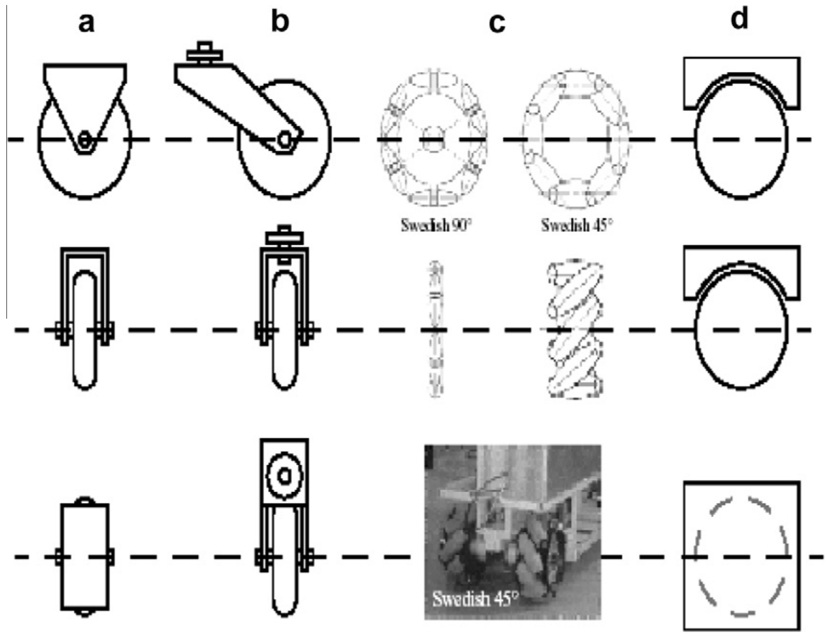

Figure 2 Four basic wheel types (Siegwart and Nourbakhsh, 2004).

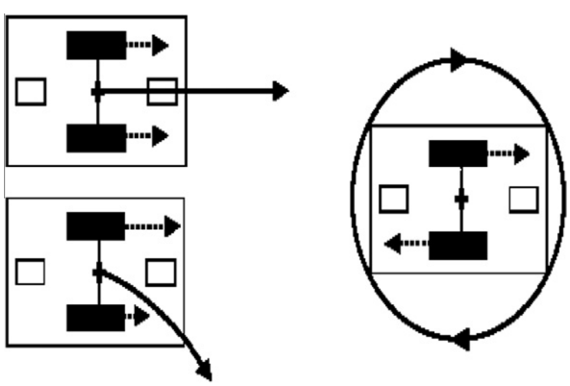

Figure 3 Driving and rotation of the differential drive.

running faster than the other, the robot drives in a curve along the arc of a circle, and if both motors are run at the same speed in opposite directions, the robot turns on the spot (minimum rotation space).

The overall robot specifications are given as (see Fig. 4):

- Dimension with 40 cm length, 30 cm width and 36 cm height.
- Weight has to be less than 10 kg.
- Power supply, 12 VDC rechargeable batteries.
- User interface, keypad, LCD, and IR remote controller.
- Control system, embedded using a microcontroller.
- Sensors, 3 Infra-Red sensor (front, right, and left), 2 Ultrasonic sensors (right front and left front), and a digital compass to measure the front heading angle.

3. Electrical design

A schematic drawing of the mobile platform is given in Fig. 5. The hardware consists of:

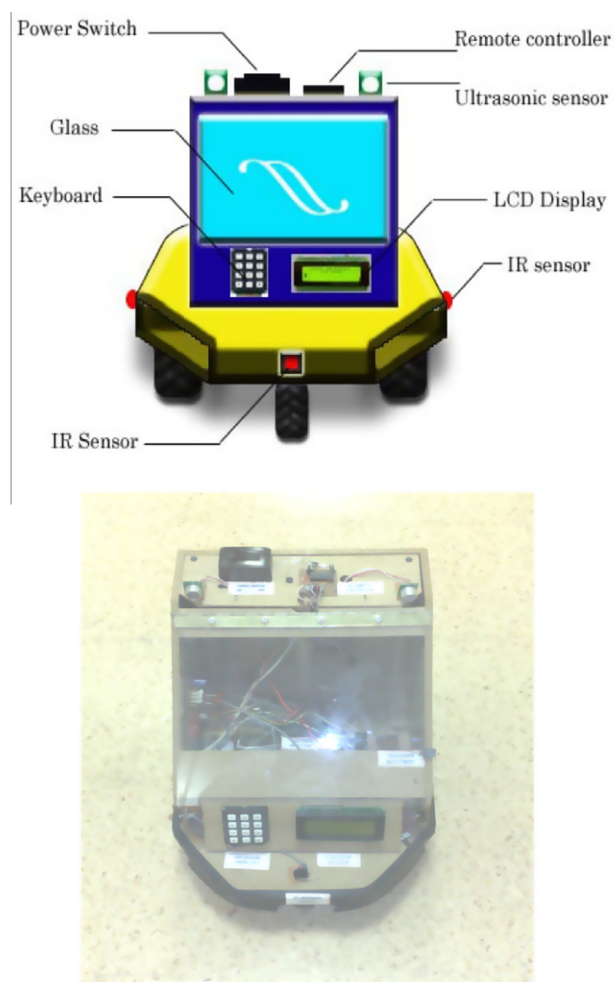

Figure 4 Mobile robot design.

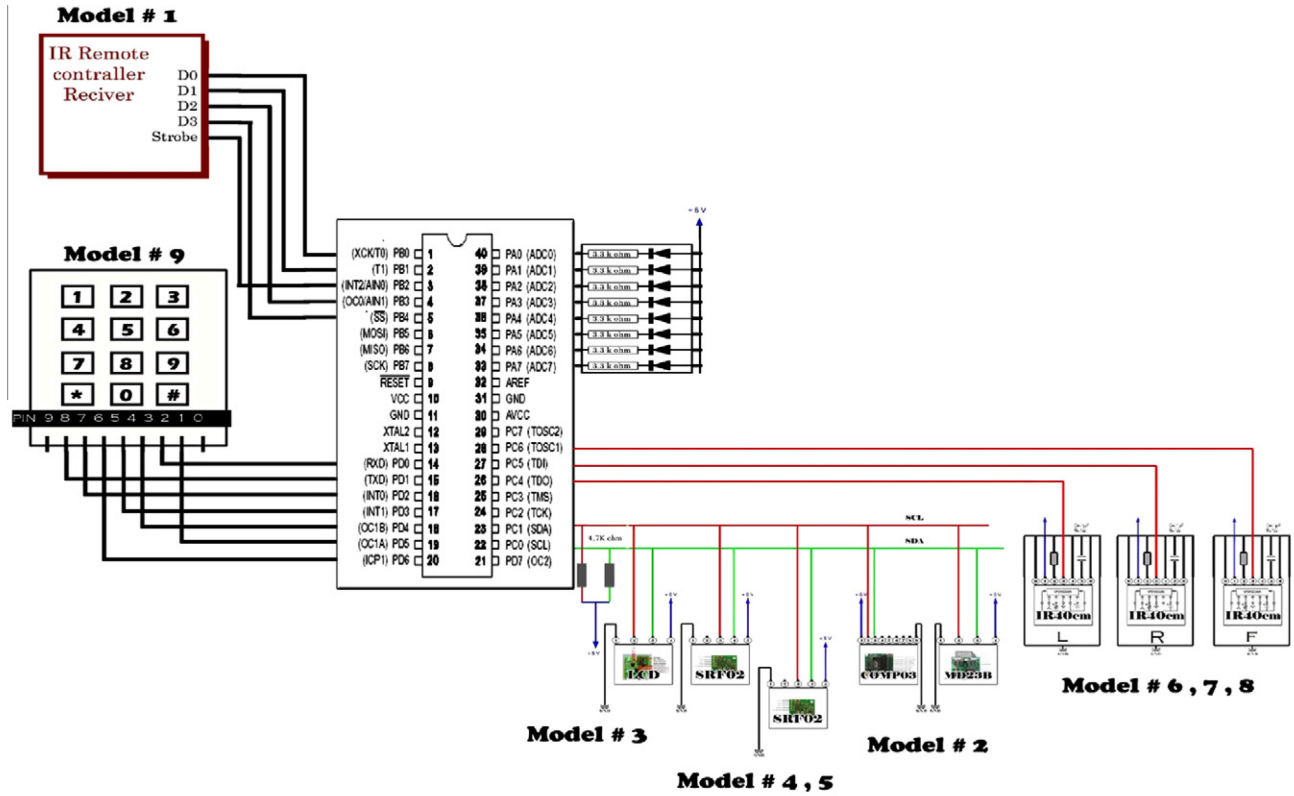

Figure 5 Electrical hardware.

- Microcontroller Atmel ATmega32 (Datasheet, 2009).
- IR receiver (remote control) circuit # 1.
- Two DC motors as differential deriving system + MD23 dual H-bridge controller board #2 to control the motors using I2C bus.
- One digital compass to detect the current direction #2 (CMPS03 connected to I2C bus).
- LCD03 connected to I2C bus #3
- Two Ultrasonic sensors SRF02 #4 and #5 (analog measurement using I2C bus).
- Three digital IR sensors #6, #7, and #8 (SHARP GP2Y0D340K with 40 cm pre-defined value to detect obstacles).
- Keypad 4X3 matrix # 9.

The microcontroller ATMEGA32 is the central brain of the mobile robot. ATmega32 chip is a powerful controller with many capabilities such as:

- It can handle the I2C bus communication using a so-called two-wire interface TWI.
- In circuit programming using flash memory.
- 32 KB flash memory.
- 1 KB EEPROM.
- 2 KB internal SRAM.
- External interrupts.
- Two 8-bit Timer/Counter.
- One 16-bit Timer/Counter.
- Four PWM channels
- 8 channels analog inputs with 10 bit A/D built in converter
- Built in clock generator with a pre-scalar.

The IR receiver module converts the received command to an equivalent binary code according to pressed key. Then it interrupts the master controller ATmega32 with the help of its strobe signal and delivers the binary number to the microcontroller. It has four data lines D_o, D_1, D_2 and D_3 are connected with PB0, PB1, PB3, and PB4 respectively. By pressing any key on the remote control, an appropriate binary number appears on these data lines and then strobe line goes low for short period (1 ms) which causes an external interrupt and the ATmega32 will fetch the binary number under the control of software. The Sharp IR detectors are connected with PC6 (front), PC5 (right side) and PC4 (left side) I/O pins of ATMEGA32. If any object is located at a distance less then 40 cm from the robot frame, the sensor output will be fired to alert the microcontroller that an obstacle is detected. Other components are connected to the standard I2C bus. The I2C (Inter-Integrated Circuit) bus is a two-wire serial interface, low to medium speed communication bus. It is developed by Philips Semiconductors Company in early 1980s (Philips, 1995). The I2C is called a two-wire interface (TWI) for Atmel AVR (Atmel, 2004). The strength of the TWI bus includes the capability of addressing up to 128 devices on the same bus. This two-wire serial interface includes SCL and SDA. SCL is the clock line. SCL is used to synchronize the rate of data transfer over the bus. SDA is the data line. The SCL & SDA lines are connected to all devices on the TWI bus. There is also 0 volt and 5 volt wires to power up the connected components on the bus. Both SCL and SDA lines are "open drain" drivers. What this means is that the chip can drive its output low, but it cannot drive it high. There should be a pull-up resistor from the SCL line to the 5v line and another one from the SDA line to the 5v line. You only need one set of pull-up resistors (R_P = 1.8 KΩ) for the whole TWI bus, not for each device, as shown in Fig. 6. The hardware architecture is open to allow the integration of any new modules in a quick and robust manner using TWI bus.

The modularity and general purpose of this robotic platform makes it very useful in education and research (Tur and Pfeiffer, 2006).

4. Software design

The software is written using ImageCraft C-compiler and development environment for Atmel AVR (ImageCraft, 2005). The compiler has a powerful capabilities and user-friendly integrated development environment (IDE). Once the compiled file (Hex code) is obtained, it can be downloaded in the microcontroller flash memory using the available Micro master programmer kit for Atmel AVR. The application software implements ten commands where each one gives a special movement mode to the robot as shown in Fig. 7.

The program continuously scans the keypad and cheek if any key is pressed to give a command. If no command is given the command No will be 15, which means do nothing just wait for a new command. If a command from 0 to 9 will be observed then the software will start that particular movement mode against the command. The program starts with initialization to declare the different ports and registers especially for the TWI.

The TWI software protocol consists of basic four functions to do the following tasks:

- Start (Initiate the TWI sequence), this will alert all the slave devices on the bus that a transaction is starting and they should listen in; next the master will send out the device address. The slave that matches this address will continue with the transaction, any other devices will ignore the rest of the transaction and wait for the next.
- Send (Write to slave), the master has to send out the register number inside the slave that it wishes to write to.
- Receive (Read from slave), the master has to send out the register number inside the slave that it wishes to read from.
- Stop (End the TWI sequence), this will complete the transaction.

5. Testing and evaluation

5.1. Right wall tracking movement (Command no. 1)

The first movement can be initiated by pressing key No.1 either from keypad in front of the robot (see Fig. 4) or from

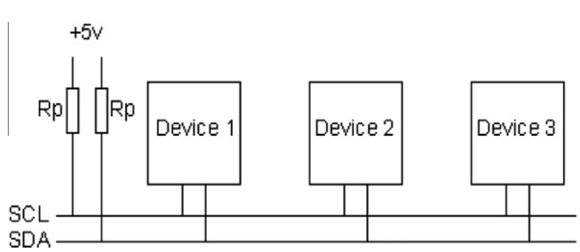

Figure 6 TWI bus interconnection.

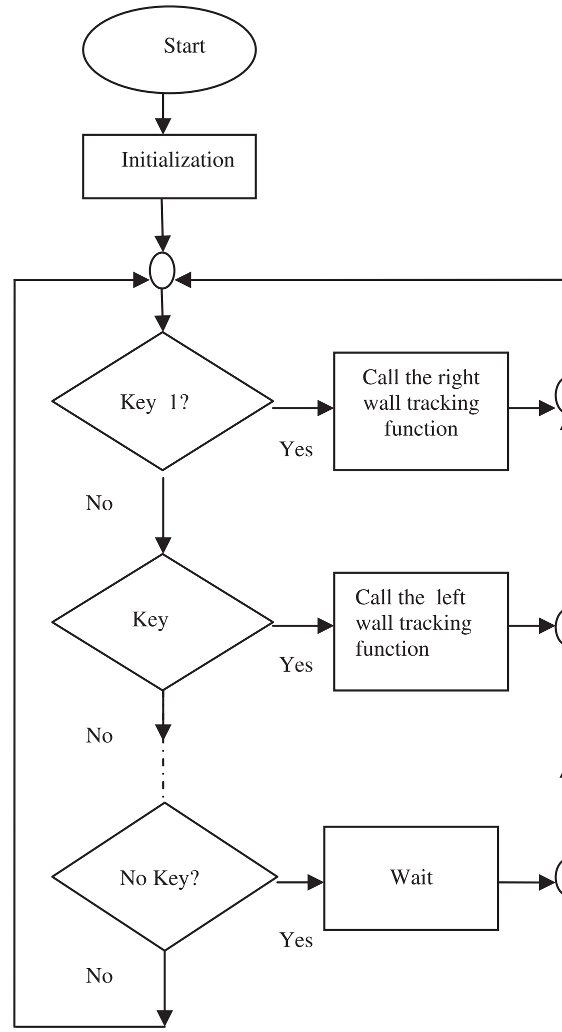

Figure 7 Main program flowchart.

a remote control. In this movement, the robot will track the right wall in keeping the safety distance 40 cm. If the robot moves closer to wall then the software will drive it away from the wall (see Fig. 8). In this mode, if the robot faced any front obstacle; it will switch on display light and stop for a while; then the display light will switched off and starts to turn left in expecting that the facing obstacle is a wall corner. IR sensors are used only in this mode. This mode can be interrupted by pressing Ok key in remote control.

In this case, the robot speed is low (0.1 m/s) to avoid unstable navigation. The robot motion takes a zigzag form to follow the wall. Fig. 9 shows the robot distance measured by ultrasonic sensor PS-2103 mounted on the top of the robot and connected to a data logger Pasco Explorer GLX to investigate the robot performance.

The switching to zero values is due to the zigzag motion of the robot. The zero measurements indicate that the robot moves towards the wall and the distance is less than 15 cm. In Fig. 9, the robot starts to move forward to track the right wall. After 3 s, the distance to the wall is less than the safe limit (40 cm). Therefore, the control program turns the robot left to avoid the collision. Once, the IR indicates that the distance is more than 40 cm; the robot turns right while moving. Different

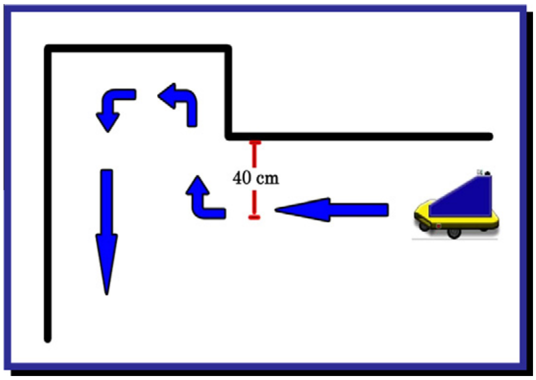

Figure 8 Right wall tracking movement.

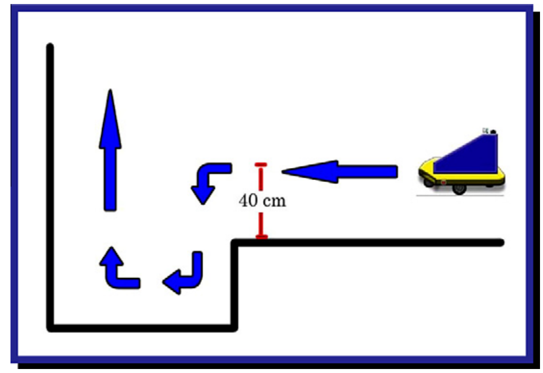

Figure 10 Left wall tracking movement.

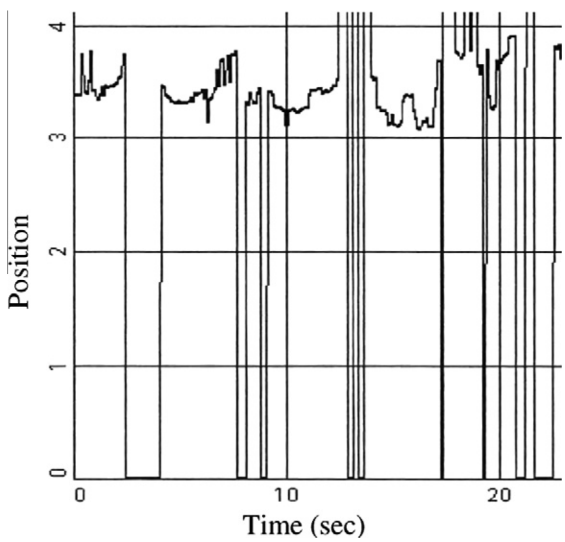

Figure 9 Right wall tracking.

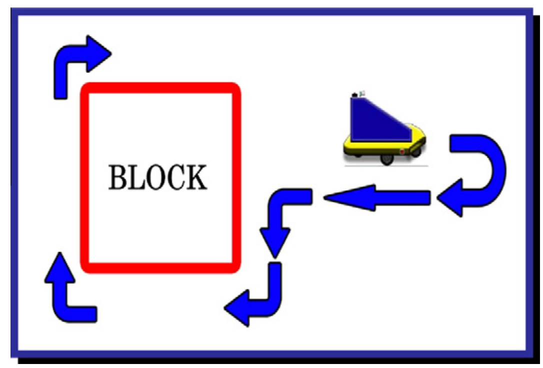

Figure 11 Circular movement.

trials are applied experimentally and the results affirmed the tracking objective without any collision.

5.2. Left wall tracking movement (Command no. 2)

The mode is similar to the previous one, but the difference is that the robot will track the wall to the left direction. By pressing key no. 2 either on keypad or in remote control, this mode will be displayed on the LCD and the movement to track the wall will start (see Fig. 10). Similarly this mode can also be interrupted by pressing Ok key on the remote control. The robot will stop and displays ''waiting for command'' on the LCD. The obtained performance affirmed the potential of the software development.

5.3. Circular movement (Command no. 3)

By pressing key no. 3 in keypad or using wireless remote control, this mode will be displayed on the LCD and the robot will start rotating in the clockwise direction (see Fig. 11).

At the same time it will observe the distance of the objects in the front of it, if it will find any object in the range of 150 cm using ultrasonic sensors RF02. Then it will stop rotating, switch in light for a while and then it will start revolving around the block. This mode can also be interrupted by pressing Ok key in the wireless remote control.

5.4. Low speed forward movement (Command no. 4)

This mode can be initiated by pressing key no. 4 either on keypad or in the remote control. In this mode, the robot moves behind a person which is in the range of the 30 cm to 125 cm in its front. If the person will turn to right or left the robot will track the moving target (see Fig. 12). It is not like a missile, which is move with high speed towards a moving target. It will also switch on display light.

But, if the person will move outside its range then it will stop and switch off the display light. This mode uses both IR detectors and SRF02 sensors. This mode can also be interrupted by pressing Ok key on keyboard.

5.5. High speed forward movement (Command no. 5)

In this mode, the robot will follow the target at a higher speed than the low speed mode (see Fig. 13).

5.6. Backward movement (Command no. 6)

In this mode, the robot will move back when the target (i.e., the person) approaches it from the front. Robot's speed will

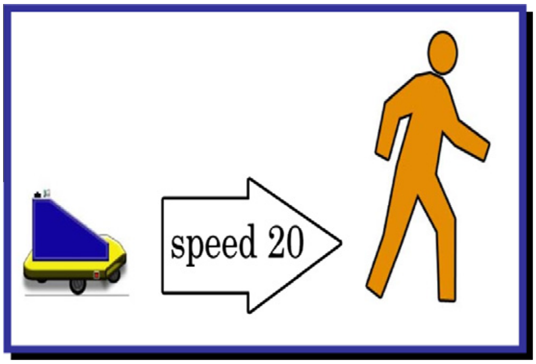

Figure 12 Low speed forward movement.

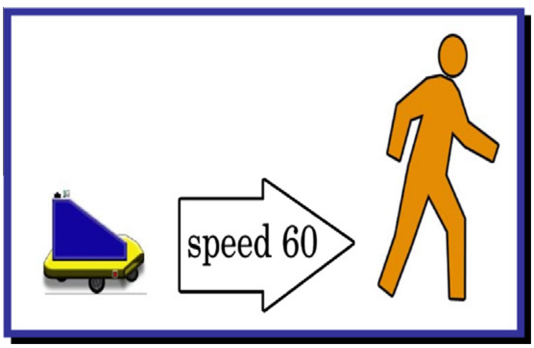

Figure 13 High speed forward movement.

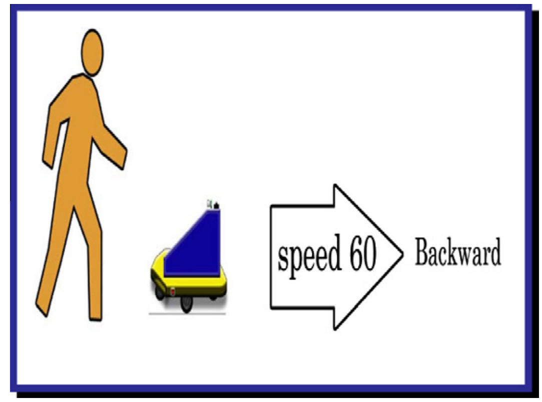

Figure 14 Backward movement.

increase as the person comes closer to it (see Fig. 14). This mode can be initiated by pressing key no. 6 either on the keypad or in remote control. The program can be interrupted by pressing OK key on remote control.

5.7. Open loop movements (Commands nos. 0, 8, 9, 7)

These are small movements, to move robot in any direction using the wireless IR remote control. The robot can move with commands as follows

- Command 0: to move a little back ward
- Command 8: to move a little forward
- Command 9: for right turn
- Command 7: for left turn

All the above commands are tested in the real time environment using either the keypad or the IR remote control. The obtained results were satisfactory to realize the expected behaviors and the investigated performance proved the potential of the proposed design and implementation.

6. Discussion and conclusions

The developed robot platform was not designed for a specific task but to be a general wheeled autonomous platform from which several applications: educational, research, or industrial can be implemented. Different indoor applications are implemented: wall tracking, moving object tracking, obstacle avoidance, and others. Students can learn different tasks: microcontroller programming using C compiler, sensor characteristics, motor driving circuits and signal condition circuit design. Practical work on robotics at the university level can help engineering students develop the needed communication skills for teamwork.

The hardware design is open using I2C bus that helps to integrate more sensors and test them on the same platform. The Platform design is achieved taken into account the stability of movements and the ability to navigate in a small work space. Therefore, the driving wheels are installed in the middle of the platform to obtain which is so-called on spot rotation. The use of TWI bus reduced the hardware effort while it requires a great effort to write a successful software protocol. In this design, the TWI protocol is developed in C-programming language from scratch as function modules to achieve different tasks as: start, send, receive, and stop sequences. Vision system using camera and GPS module can be integrated in the design for the future to build a highly autonomous obstacle avoidance and trajectory planning.

References

Atmel, 2004. Application Note AVR311: Using the TWI module as I2C slave. Available online at: <http://www.atmel.com/dyn/resources/prod_documents/doc2565.pdf>, Last update 2004.

Braeunl, Th., 2006. Embedded Robotics: Mobile Robot Design and Applications with Embedded Systems. Springer-Verlag, Berlin, Heidelberg.

Datasheet, 2009. Atmel ATmega32. Available online at <http://www.atmel.com/dyn/resources/prod_documents/doc2503.pdf>, Last update 2503O–AVR–07/09, 2009.

Dubowsky, S., Genot, F., Godding, S., Kozono, H., Skwersky, Yu H., Yu, L.S., 2000. PAWM – a robotic aid to the elderly for mobility assistance and monitoring: a helping-hand for the elderly. In: Proceedings of International IEEE Conference on Robotics and Automation, vol. 1/4, pp. 570–576.

Graf, B., 2001. Reactive navigation of an intelligent robotic walking aid. In: Proceedings of IEEE International Workshop on Robotics and Human Intelligence, Paris, France, 2001, pp. 353–358.

Graf, B., Hans, M., Schraft, R.D., 2004. Mobile robot assistants. IEEE Robot. Automat. Mag., pp. 67–77.

ImageCraft, 2005. C-Compiler Manual “ICCV7 for AVR”. Available on line at: < http://www.imagecraft.com > , Last update 2005.

Nehmzow, U., 2006. Scientific Methods in Mobile Robotics: Quantitative Analysis of Agent Behaviour. Springer-Verlag, London Limited.

Philips, The I2C-bus and how to use it (including specifications). Available on line at: < http://www.datsi.fi.upm.es/docencia/Micro_C/i2c.pdf > , April 1995.

Rentschler, A.J., Cooper, R.A., Blasch, B., Boninger, M.L., 2003. Intelligent walkers for elderly: performance and safety testing of VA-PAMAID robotic walker. J. Rehab. Res. Develop. 40(5), 423–432.

Siegwart, R., Nourbakhsh, I., 2004. Introduction to autonomous mobile robots, Massachusetts Institute of Technology, USA.

Tur, J.M., Pfeiffer, C.F., 2006. Mobile robot design in education. IEEE Robot. Automat. Mag., pp. 69–75.