Design of Mobile Holonomic Robot with Wireless Inertial Measurement Control System

Projekt mobilnego robota holonomicznego z bezprzewodowym inercyjnym systemem sterowania

PIOTR RYCHLIK
WOJCIECH KACZMAREK *

DOI: https://doi.org/10.17814/mechanik.2017.7.96

The article presents the design of mobile holonomic robot, where special attention was given to the robot's control method. It discusses the way of processing values from an inertial measurement unit to actuator signals. In order to determine the correct functioning of the robot, a number of tests were carried out covering its software, and followed by the determining of basic parameters characterizing robot's mobility.

KEYWORDS: mobile robot, holonomy, control, inertial measurement unit, IMU, swedish wheel, omni wheel

Holonomy is a concept that applies to all objects that can change their position in space by themselves, and specifically refers to mobile robots - is one of the parameters that characterize the division of these robots by the kinematic model.

If the n-dimensional space is assumed to be the reference space, then it must have at least n degrees of freedom to be called an object in that holographic space. However, if an object can change the orientation of its coordinate system without changing its position in space, then such a system, even if it cannot move along one axis, is also called holonomic.

Inertia control reads the current (instantaneous) acceleration and momentum values of the accelerometer and gyro respectively from each of the three axes of the Cartesian coordinate system. Inertial sensors have been used especially in inertial navigation systems that are used in jet propulsion and are autonomous.

Robot construction

One of the ways to ensure a mobile holomotor is the use of Swedish (omni-directional) wheels, which, independently or in combination, increase the number of degrees of freedom of the robot (fig. 1). The characteristic feature of the Swedish wheels is the placement of rollers on their periphery, which allows movement along the main (active) axis of rotation of the wheel.

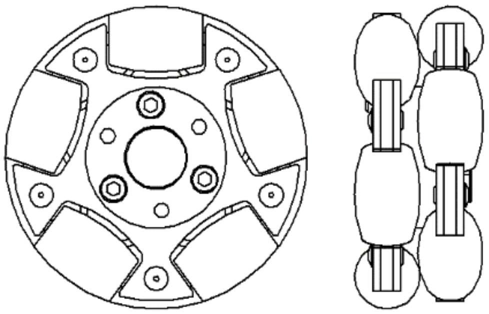

Fig. 1. Twin omni-directional wheel with a diameter of ⌀60 mm (based on the 3D model available at: www.robotshop.com/en/60mm-aluminum-omni-wheel.html)

Brushless DC motors with 70:1 integrated planetary gear ratio, torque of 1.41 Nm and rotational speed of 150 rpm (fig. 2) are used for driving the robot.

The smartphone's inertial output is sent wirelessly to the high-performance computing unit of the Raspberry Pi Zero microprocessor, which is responsible for processing data. Due to the small capacity of the hardware generated rectangular signal by the microcomputer an additional ATMega2560 microcontroller based unit and 16 hardware ports supporting pulse width modulation were used (this was sufficiently sufficient for the project).

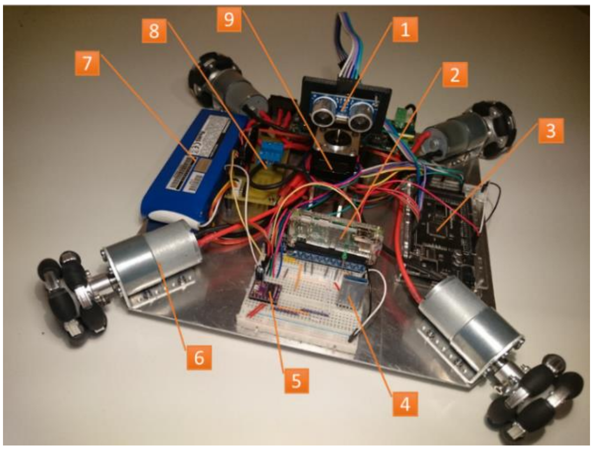

Fig. 2. Robot view: 1 - distance sensor, 2 - master unit, 3 – sub-unit, 4 – bluetooth module, 5 - stepper motor controller, 6 - DC motor, 7 - battery, 8 - power module, 9 – stepper

---

* Piotr Rychlik, ppłk dr inż. Wojciech Kaczmarek (wojciech.kaczmarek@wat.edu.pl) – Wydział Mechatroniki i Lotnictwa, Wojskowa Akademia Techniczna

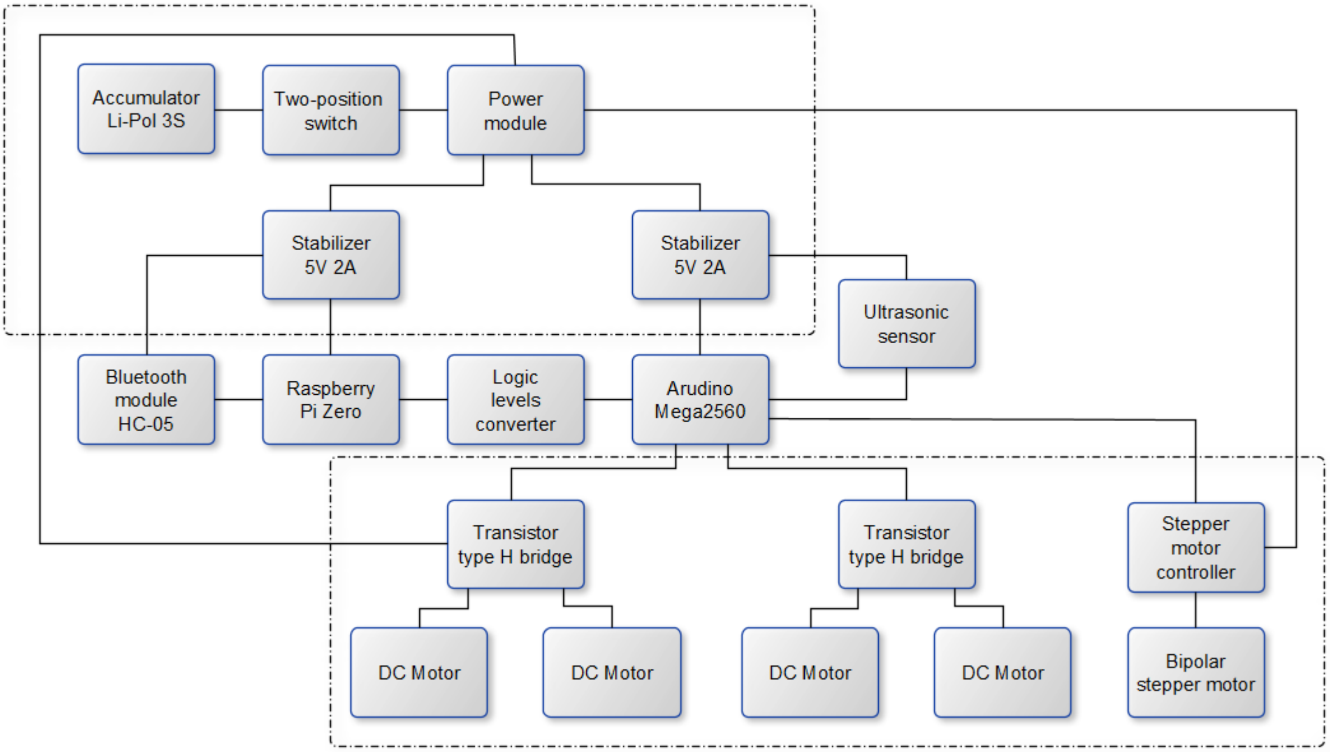

Fig. 3. Block diagram of electrical connections

The robot was equipped with an ultrasonic distance sensor with the ability to rotate with respect to the vertical axis, allowing it to rotate in the direction of the robot's motion. The simplified block diagram of electrical connections is shown in fig. 3.

Communication between the microcomputer and the microcontroller is through an I^2C interface, where the correctness of the transmitted data is checked by means of cyclic redundant CRC control.

Software platform and workflow algorithm

To control the direction and speed of movement of the robot will use pulse width modulation PWM control signals and the direction of rotation of the shafts of engines that will be given to the H-bridge transistor connected to the DC motors. Determining the robot's motion parameters will take place in the parent unit based on data from the smartphone accelerometer, and then these parameters will be sent to the slave.

The frame data on the direction and speed of movement is sent from the phone using the following scheme:

$$47_{(16)}\ Acc_{x(16)}\ 2C_{(16)}\ Acc_{y(16)}\ 2C_{(16)}\ Acc_{z(16)}\ 0A_{(16)} \qquad (1)$$

The frame starts with the letter "*G*" written as $47_{(16)}$ in the ASCII hexadecimal system [5], followed by $Acc_{x(16)}$, $Acc_{y(16)}$, and $Acc_{z(16)}$ acceleration values (respectively along the *X*, *Y* and *Z* axis) are ASCII characters, each separated by a comma ($2C_{(16)}$) for easier and faster processing of the received values by the parent. Data frame ends with newline "\ n" - $0A_{(16)}$.

Taking into account only the acceleration of the earth's acceleration values recorded by an inertial sensor should be between 0 and approximately 9.81 m/s^2 depending on the current orientation of the accelerometer relative to the acceleration. In the event of additional acceleration in the Earth acceleration compensation system, the Acceleration value may be greater (or less - if reversed) than the above, so to compensate for undesirable factors potentially affecting the robot's motion, then the acceleration of gravity and acceptable, predetermined tolerance of:

$$g - \Delta Acc \le \sqrt{Acc_x^2 + Acc_y^2 + Acc_z^2} \le g + \Delta Acc \qquad (2)$$

where: Acc_x, Acc_y, Acc_z - acceleration along the *X, Y* and *Z* axes [$m \cdot s^{-2}$], ΔAcc - acceleration deviation [$m \cdot s^{-2}$], *g* - earth acceleration ≅ 9.81 [$m \cdot s^{-2}$].

The idea of processing the data from the inertial sensor to the kinematics of the robot is shown in fig. 4. As can be seen, the function is even, i.e. the mean value of the generated signal is the same for accelerations with the same values, but opposite ones. The designed characteristic is directly proportional to the acceleration module in compartments II and IV and can be written as a system of equations:

$$PWM(Acc) = \begin{cases} PWM_{max}, & Acc_{xvy} > |Acc_{max}| \\ a \cdot |Acc| + b, & Acc_{xvy} \in [\pm Acc_{min}; \pm Acc_{max}] \\ 0, & Acc_{xvy} < |Acc_{min}| \end{cases} \qquad (3)$$

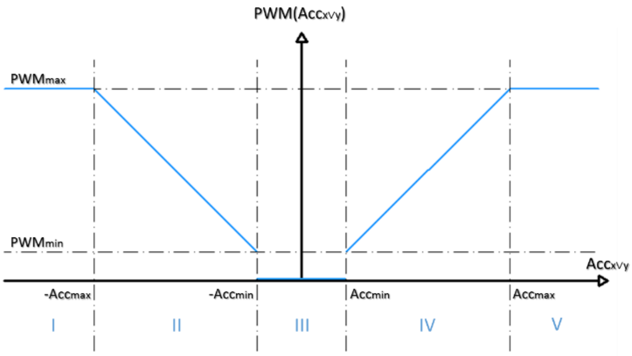

Fig. 4. Speed control function. Assignments: Acc_{xvy} - acceleration value along the *X* or *Y* axis of the smartphone; $PWM(Acc_{xvy})$ - value of pulse width modulation in acceleration function, which is sent to corresponding microcontroller register; $|Acc_{max}|$, $|Acc_{min}|$ - the acceleration characteristics (points), resulting in a change in the monotonous nature of the function; PWM_{min} - minimum average signal value causing robot movement; PWM_{max} - the maximum value of the octal PWM register, equivalent to a fill factor of 100%

The minimum value of the modulated PWM_{min} pulse width, which causes the motion of the robot, has been determined empirically. The high inertia of the robot does not allow it to move away at values smaller than PWM_{min} and greater than zero, which results in the motors striving for maximum current draw - this lasts for a period of high signal generation and repeats with a period inversely proportional to frequency signal. That is why the characteristic shown in fig. 4 does not include the range along the ordinate axis greater than zero and less than PWM_{min}.

Fig. 5 shows the algorithms for the master unit, implemented in Python, and for the slave algorithm in C language.

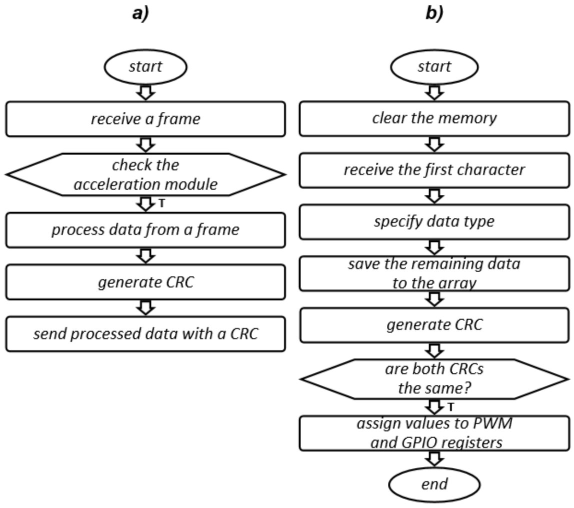

Fig. 5. Algorithms: a) main program of the master unit, b) program interrupt of the slave unit

In both of the algorithms in fig. 5, failure of one of the test conditions contained in the hexagons causes the algorithm to continue to run and return to its origin. In addition, in a simple software loop, the slave unit checks the distance to the nearest obstacle - if it is too small, the unit does not allow further movement of the robot in that direction.

Rotation of the stepper motor shaft is done autonomously based on the acceleration values along the *X* and *Y* axis of the accelerometer. To determine the angle of rotation, the complex number *z* was used, from which the real part is equal to the acceleration along the axis of the accelerometer *X* and the imaginary part acceleration along the axis *Y*:

$$z = Acc_x + iAcc_y \tag{4}$$

Determination of the angle of rotation consists in calculating the argument of the main complex number with, for example according to formula [2]:

$$\varphi = arg\, z = arctg \frac{Im(z)}{Re(z)} = arctg \frac{Acc_y}{Acc_x} \tag{5}$$

Then, it should be converted to the number of steps that the stepper must perform to achieve the desired position, taking into account the increment that occurred with respect to the previous angle value.

Tests

Prior to the road test, the following were checked: correctness of the connections of individual components, values of voltage and polarity. Software testing has also been carried out on the communication between the components and has verified whether the data processing proceeds correctly and has the desired effect. Within the framework of the data transmission, deliberately introduced an error to calculate the CRC to verify that for different checksums erroneous data will be ignored.

During basic tests, the accuracy of object detection was determined by the ultrasonic distance sensor used, whose measurement accuracy is approximately 3 mm. If an obstacle is detected at a distance of 40 cm from the sensor software properly preventing further movement in that direction. The maximum speed of the robot is about 0.4 m/s and the sloping angle of the robot can be approx. 26° when using one pair of motors and 34° for two pairs of motors. Speed and angle measurements were determined indirectly, i.e., based on the functional relationship between the value obtained directly and the measured value [3].

Conclusions

The use of omni-directional wheels increased the maneuverability of the robot - allowing it to travel sideways and diagonally and rotate around its axis. These wheels, however, have their drawbacks, which appear even when entering the sloping plain. For both instances of the resulting inclination angle, the motors have a power reserve to overcome the elevation, but the wheels do not provide sufficient motion and adequate traction. One way to improve the grip and smoothness of a robot's motion when climbing to an elevated slope is to use more wheels (appropriately located) on one shaft. With cyclic redundancy, the risk of incorrect data processing was reduced.

REFERENCES

1. Chwaleba A., Moeschke B., Płoszajski G., Elektronika, Warszawa: WSiP, 2008, pages 304-324.
2. Leitner R., Zarys matematyki wyższej dla studentów cz. II, Warszawa: WNT, 1987, pages 327-343.

3. Chwaleba A., Poniński M., Siedlecki A., Metrologia elektryczna, Warszawa: WNT, 2010, page. 21.
4. Nota katalogowa stabilizatora L78S05CV, [dostęp: 25.01.2017]http://pdf1.alldatasheet.com/datasheet-pdf/view/22636/ STMICROELECTRONICS/L78S05CV.html
5. Tabela znaków ASCII, [dostęp: 12.01.2017] http://www.asciitable.com ■