Controllable self-balancing robot

Łukasz Maliszewski

February 2, 2017

Intermediate Projectas
under instructions of
Witold Paluszyński Ph.D.

Embedded Robotics,
Chair of Cybernetics and Robotics,
Faculty of Electronics,
Wrocław University of Technology

Abstract

The main goal of this project is to design and construct two-wheel controllable self-balancing mobile robot. Robot will be based on Discovery F3 board with IMU sensor to mesure position in space. Overall there should be possibility to control robot via wireless gamepad. All asumptions of the project have been fulfilled

1. Introduction

The main point of this project is to design and construct two-wheel controllable self-balancing mobile robot. Robot will be based on Discovery F3 board with IMU sensor to mesure position in space. Robot should be able to hold vertical orientation with as small as possible oscilation. Overall there will be possibility to control movement via wireless gamepad.

1.1. Goals

Main goals of the project:

- design and construct two-wheel mobile platforms
- design electronic part of robot
- program microcontroller such that robot will be able to self-balancing
- connect robot with gamepad to control trajectory of movement

1.2. Tools

Tools, which were used in project:

- KiCad EDA - to design PCB with motor driver
- CubeMX - to configure $DiscoveryF3$ board with $STM32$ microcontroller
- Keil uVision5 - to develop software

2. Mechanical and electronic part

As we can observe on the Figure 1. and Figure 2., robot consist of:

- $DiscoveryF3$ board
- two micro motors - Pololu HPCB 30:1 with wheel
- DC-DC step-down module with $5V$ output
- self made motor driver module based on MC33932
- self made aluminum construction
- 7.4V Li-Pol battery
- HC-06 bluetooth module

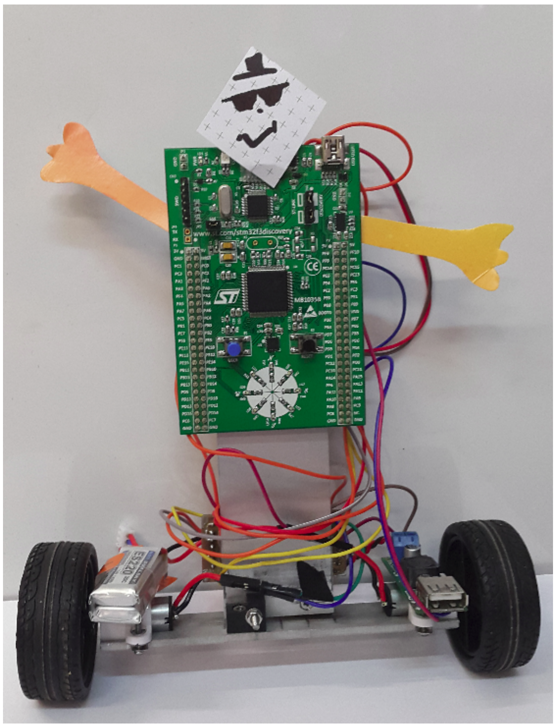

Figure 1: Self-balancing two-wheel robot "bAtek" - front view

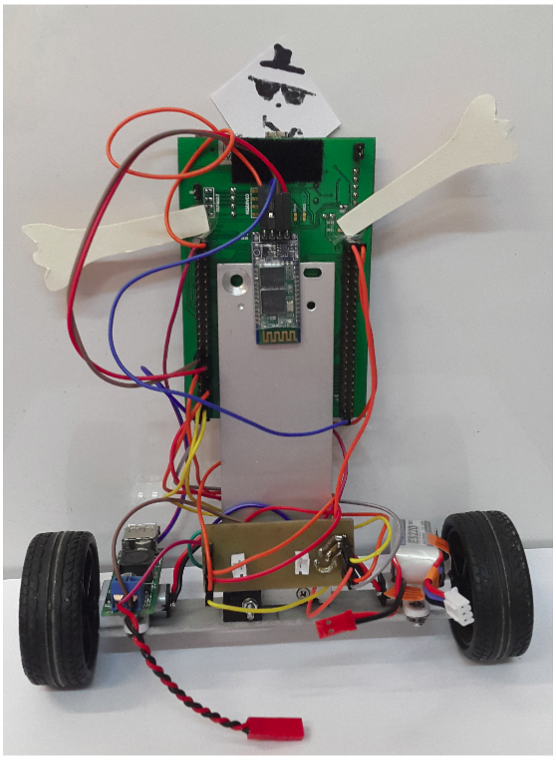

Figure 2: Self-balancing two-wheel robot "bAtek" - back view

Table 1: Chosen paramenters of *STM32 Discovery F3* by *ST*

Microcontroller	*STM32F303VCT6*
Range of microcontroller voltage	$2,0$–$3,6\ V$
Frequency	$72\ MHz$
Flash memory	$256\ KB$
RAM memory	$48\ KB$
Microcontroller's case	$LQFP100$
Maximum output current	$25mA$
Board power supply voltage	$5V$ *lub* $3,3V$

Table 2: Chosen paramenters of *Pololu HPCB* micro motors

Rated voltage	$6V$
Gear	30:1
Rotation speed	$440RPM$
Stall current	$1800mA$
Torque	$0,28kg * cm\ (0,027Nm)$
Size	$24 \times 10 \times 12mm$
Mass	$10g$

2.1. DiscoveryF3 board

The main control unit is *STM32 Discovery F3* evaluation board. Chosen parameters are shown in Table 1. This board contains accelerometer and gyroscope so we are able to mesure angle of balance. Communication to this peripherials is possible by using SPI and I^2C interface. Microcontroller generate also PWM signal to control velocity of motors and digital signal to controll direction.

2.2. Motors

In project used Pololu HPCB micro motors with gear 30:1. Parameters of motors are shown in Table 2. Motors have been chosen because of them price and availability. Unfortunately first motors - Faulhaber 1524 were destroyed because of misuse so it was necessary to quick find replacement.

2.3. Power supply

As power supply Discovery board was used DC-DC module with 5V output. Robots is powered by 7.4V Li-Pol battery.

2.4. Motor driver

Motor driver based on MC33932 was design in KiCad software. On Figure 3. and Figure 4. shown the schema and PCB project.

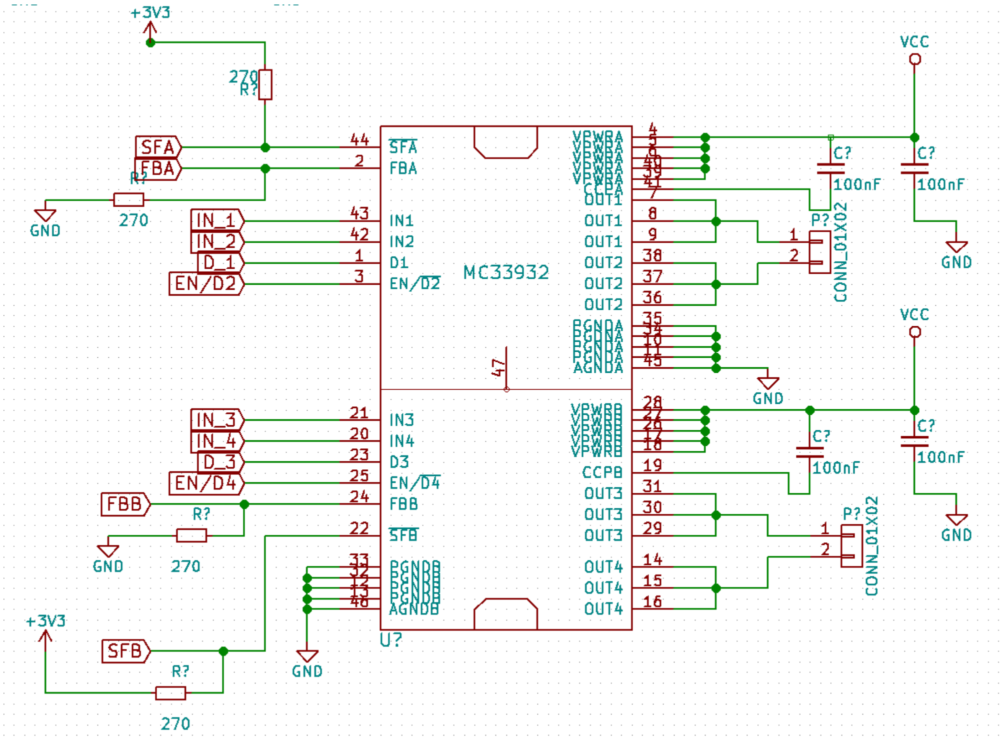

Figure 3: Scheme of motor driver

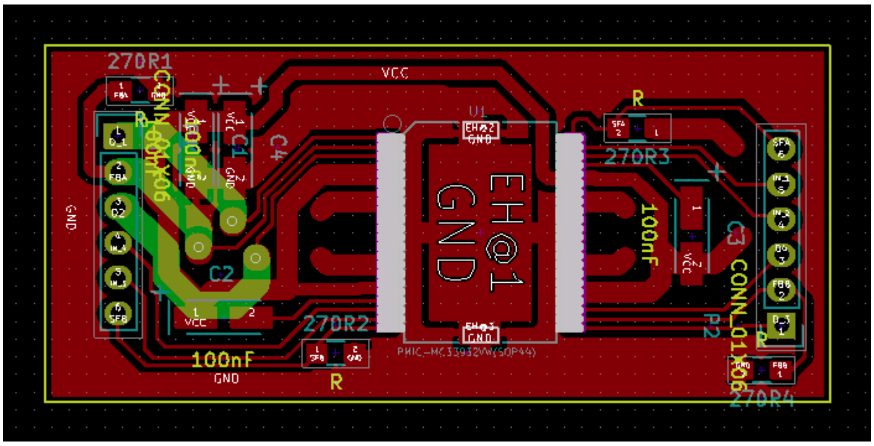

Figure 4: PCB project of motor driver

2.5. Bluetooth module

Connection with robot is possible using bluetooth via HC-06 module. It is very popular ready to use module with UART interface.

2.6. Aluminum construction

Body of robot is a self made construction cosnsist of two aluminum plates. Parts are conected using $M2.5$ screws forming inverted letter "T"

3. Software part

3.1. Self-balancing task

Control loop for balance task consist of three main parts:

1. get data from accelerometer and gyroscope
2. filter data using complementary filter
3. drive motors based on filtered data using PID controller

There is also implemented interrupt from UART, such that when some message is recive, program leave main loop and operate this interrupt.

3.1..1 Complementary filter

Using single sensor as accelerometer or gyroscope is not enough to obtain real angle because of noises and disturbances. To get precise information about actual angle of robot it was necessary to implement complementary filter using accelerometer and gyroscope. Implementation of this filter is as follows:

angle = 0,98 · (angle + gyroData · dt) + 0,02 · accData

Thanks of this filtering, actual data from accelerometer, which often have a lot of disturbances have less meaning to obtain actual angle. Values *0,98* and *0,02* are constant and can be change to tune filter.

3.1..2 PID controller

Self balancing was carried out based on classical PID controller which in this case is absolutely enough. Constatns K_P, K_I, K_D was tune experimentally during test. General scheme of PID controller shown on the Figure 5.

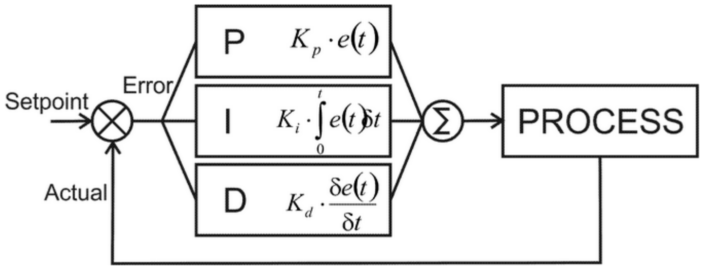

Figure 5: General scheme of PID control loop

3.2. Control via gamepad

To control robot via gamepad have been used *python* module - *pygame*. This module allows to get data about buttons directly from gamepad. In this project was used *Logitech Gamepad F710*. Data from joypad are converted into simple frame consist of one character such that:

- for go forward - 'W'
- for go back - 'S'
- for turn left - 'A'
- for turn right - 'D'

Character is sended into HC-06 module via bluetooth, and then HC-06 through UART sent message to microcontroller. When microcontroller recieve message from UART, then interrupt is generated and for 50ms motors accomplish given movement. Thanks of that we are able to control robot.

4. Summary

The project has been completely finished. Main assumptions of self balancing and control via joypad are fulfilled. In class of balancing robots it is very important to get precise information about actual angle. So filtering data from accelerometer and gyroscope is necessary, without proper processing it is impossible to actuated robots. Second thing worth of attention is mass of robot and torque if motors. It should be noted that too heavy robots in combination with to small torque of motors have no chance to self balance.

Next step of developing this project could be implement autonomus behavior using sonars to avoid obstacles or wall. Also adaptation robots to follow the line as classical linefollower can be a good idea.

References

[1] Ye Ding, Joshua Gafford, Mie Kunio *Modeling, Simulation and Fabrication of a Balancing Robot* . Harvard University, Massachusettes Institute of Technology, 2012

[2] Brian Bonafilia, Nicklas Gustafsson, Per Nyman and Sebastian Nilsson *Self-balancing two-wheeled robot.*

[3] Mikael Arvidsson and Jonas Karlsson *Design, construction and verification of a self-balancing vehicle.*. Department of Signals and Systems Chalmers University of Technology Göteborg, Sweden, 2012

[4] Hellman Hanna, Sunnerman Henrik *Two-Wheeled Self-Balancing Robot.* KTH Royal Institute of Technology in Stockholm, 2015

[5] Pieter-Jan Van de Maele *Reading a IMU Without Kalman: The Complementary Filter*
`http://www.pieter-jan.com/node/11`

[6] Datasheet of Discover F3 board
`http://www.st.com/en/evaluation-tools/stm32f3discovery.html`