

PAPER • OPEN ACCESS

Development of control systems kit for study of PID controller in the framework of cyber-physical systems

To cite this article: Andrej Škraba *et al* 2020 *IOP Conf. Ser.: Mater. Sci. Eng.* **734** 012105

View the [article online](#) for updates and enhancements.



240th ECS Meeting

Digital Meeting, Oct 10-14, 2021

We are going fully digital!

Attendees register for free!

REGISTER NOW



Development of control systems kit for study of PID controller in the framework of cyber-physical systems

Andrej Škraba¹, Vladimir Stanovov² and Eugene Semenko²

¹ University of Maribor, Cybernetics & Decision Support Systems Laboratory, Faculty of Organizational Sciences, Kidričeva cesta 55a, 4000 Kranj, Slovenia

² Reshetnev Siberian State University of Science and Technology, Institute of Computer Science and Telecommunications, 31 Krasnoyarskiy Rabochiy Ave., Krasnoyarsk, 660037, Russian Federation

E-mail: andrej.skraba@um.si

Abstract. Control systems kit for studying the PID principles was developed using affordable microcontroller Arduino UNO and ARM Mini PC Raspberry Pi 3 Model B as well as GK802V5LE. PID algorithm in JavaScript/ECMAScript is provided. The concept of the Cyber-physical System is addressed and the possibility to extend the classical control system paradigm. A detailed description of the system structure is provided as well as the layout of the user interface, which enables us to monitor the key parameters of the PID control algorithm in real-time. Node.js was used on the server-side as well as WebSockets for the communication between client and server. The proposed platform is suitable for rapid prototyping in the area of Cyber-physical Systems and the Internet of Things.

1. Introduction

PID control algorithm is one of the basic engineering principles, which is important to grasp in all areas of science. However, true understanding might be challenging for students, due to the lack of physical realization of control system in class [1]. Another motivation to study control principles on the physical plant is development of Cyber-physical systems, which could be understood as mechanism, that are controlled by computer algorithms tightly integrated by internet and its users.

Therefore, on one side, we have a motivation to address control systems and PID control and on the other hand the development of Cyber-physical system which means, that the technology used should be accessible over internet [2]. This is of special importance if one wants to connect the physical object with cloud intelligence [3, 4]. New possibilities for development in the area of control systems include self-driving cars, autonomous robotics, human-collaborative robotics etc. which are all tied to control system field [5-7]. Notion of the “Cyber” should also include the simulation model, lately known as digital twin of the system. Regarding the new technologies, [8] “Educators must have an open attitude towards new technologies. They should sensibly incorporate new technological development to avoid the risk of teaching the students of today, how to solve the problems of tomorrow, with the tools from yesterday.” One of the digital twin building blocks should consider the physical setup of the control system. Mastering of theoretical concepts is certainly important [9-11], however, some details are usually, linked to careful examination of the real system [12]. Idea of developing the control system over the internet is certainly engaging. Besides, connecting several of such systems provides us with the



opportunity to develop mass collaborative Cyber-physical environment exploiting the intelligence of the cloud [3, 4].

There are certainly several out of the box solutions to provide control system in class, however it is important, that new technologies are considered such as ESP8266 and ESP32 [13-16]. It is important, that students have access to affordable hardware to develop and test the control systems at home and potentially develop professional systems with the same components, which are nowadays found in many professional applications [17].

2. Methodology

The PID controller is widely used in industrial control systems as well as in many other areas of application [18]. PID control is abstracted by the following equation defining the input to the controller $u(t)$:

$$u(t) = K_p \varepsilon(t) + K_i \int_0^t \varepsilon(\tau) d\tau + K_d \frac{d\varepsilon(t)}{dt} \quad (1)$$

where K_p , K_i and K_d are Proportional, Integral and Differential coefficients for the proportional, integral, and derivative terms respectively. $\varepsilon(t)$ is difference between the reference and output value representing error on which is the input to the PID controller.

In order to realize the control principle provided by abstract relation in Eq. (1), one should develop appropriate physical setup. In our case, we will control the position of DC motor shaft [1, 12]. The structure of the control system is outlaid in figure 1. As the input to our control system the desired value i.e. position of the DC motor shaft is determined by input potentiometer (1). Output (2) is in our case the actual DC motor shaft position. (3) is H-bridge controller [21], (4) is plant, i.e. DC Motor, (5) represents the feedback, (6) comparator, that consists of Mini ARM PC and microcontroller Arduino, (7) is connection between comparator and controller and (8) information takeoff, where we measure the actual position of the DC motor shaft and feed it back to the input, in our case, this value goes to the comparator.

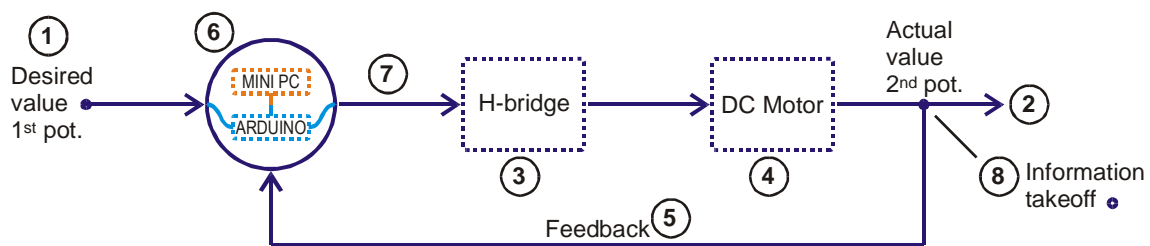


Figure 1. Control system structure.

Figure 2 shows control system structure, where PID controller is inserted between comparator and H-bridge controller. Input to the PID structure is $e(t)$ which represents the error, i.e. the difference between desired and actual value. According to the parameters of PID controller, i.e. K_p , K_i and K_d , the input to the H-bridge controller $u(t)$ is determined. In our case, $u(t)$ is the sum of proportional, integral and differential term.

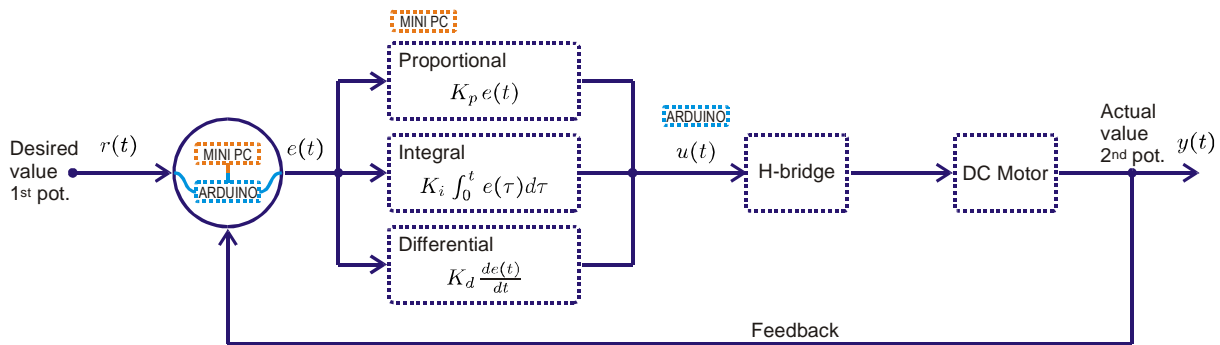


Figure 2. Control system structure with PID controller.

Additional development, which augments the classical control system paradigm is enhancement with the connection to the cloud information infrastructure, which is shown in figure 3. Under the dashed green line the classical control system is shown. Here multiple instances could be considered, which is marked with the dots. Important addition to the classical control system is interconnection to the Cloud – internet. This enables us, to harvest [4] cloud intelligence such as for example speech recognition in order to control Cyber-physical system. Important attribute of such system is, that it is open and by the large number of users also learnable. By combining several cloud information services (1, 2, ... N), one could get important improvement in performance [4]. The fact that the system is connected to the internet enables us easy programming of such platforms as well as remote control and monitoring.

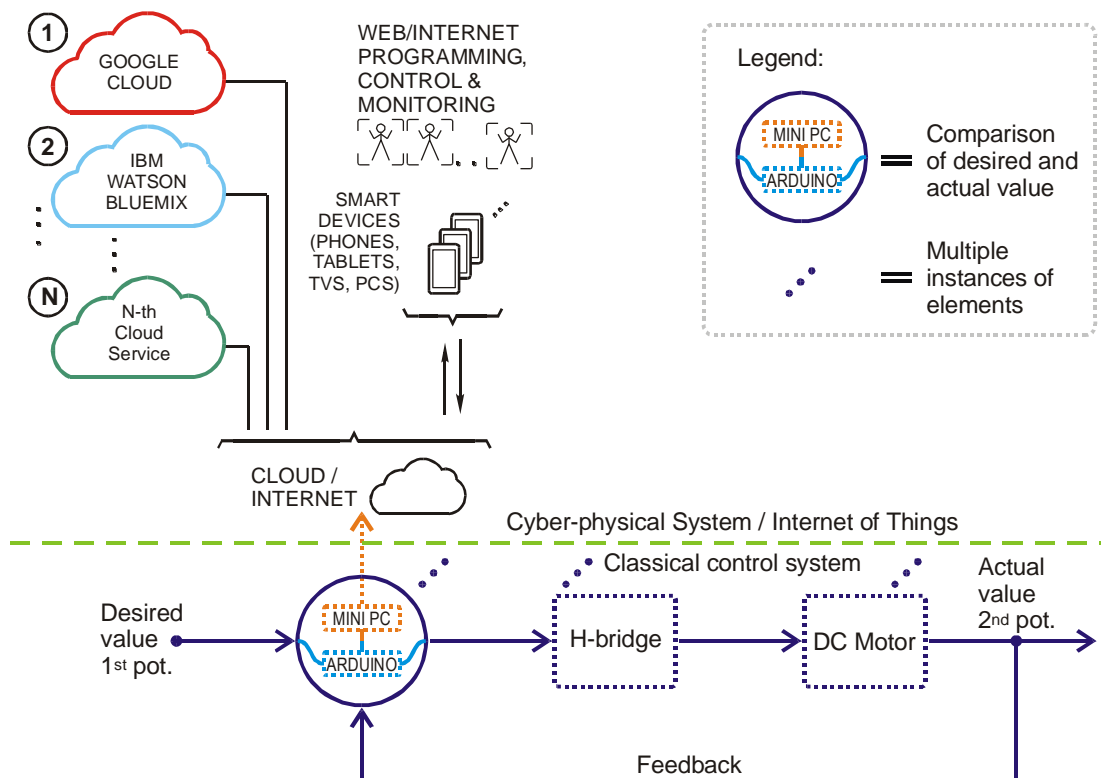


Figure 3. Control system in the framework of Cyber-physical Systems (CPS).

In order to realize such system, the following technologies were applied: Arduino microcontroller, Mini ARM based computer with Linux operating system, node.js for server side execution of

JavaScript/ECMAScript, firmata, socket.io and cloud9 IDE [2]. Cloud9 IDE is important part of the system since it enables us to access the platform for programing remotely over phone or laptop connected to the WiFi network.

3. Results

Figure 4 show the realization of the system, which corresponds figure 1 (left). The numbering is matching, therefore one could easily identify particular object from the abstract schematics. With (1) the input potentiometer is shown, (2) is the position of the DC motor output shaft, here the pointer is used. (3) shows the H-bridge on the breadboard, (4) is the small DC motor with gears, (5) represents the feedback, which is realized by white wire, (6) is ARM Mini PC GK802 with Arduino UNO microcontroller, (7) marks connection between Arduino UNO microcontroller and H-bridge, i.e. connection between comparator and controller and (8) marks the information takeoff, which is realized by the output potentiometer.

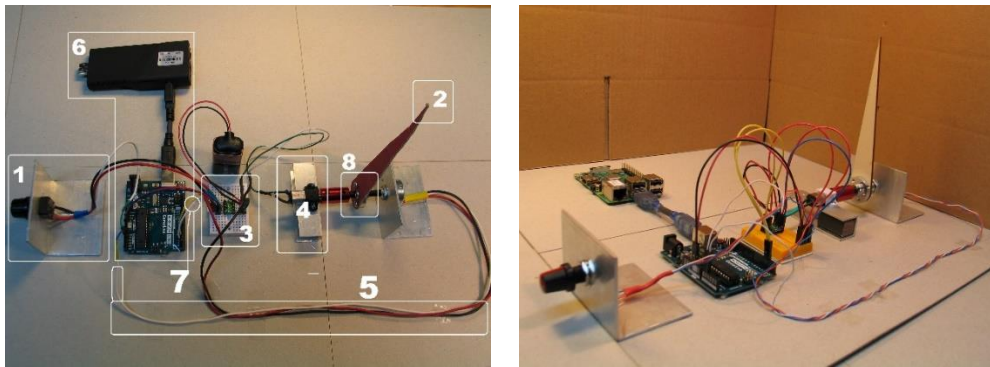


Figure 4. Realization of control system with Arduino UNO and ARM Mini PC GK805V5LE (left). On the (right) the realization with Raspberry Pi 3 Model B is shown.

In order to study the system dynamics, the MATLAB/Simulink [19] model has been developed [12]. The model structure is shown in figure 5. Important aspect of the modeling process is validation which has been performed by applying standard measures, such as Mean Absolute Percent Error (MAPE) and MSE (Mean Squared Error) [22, 23].

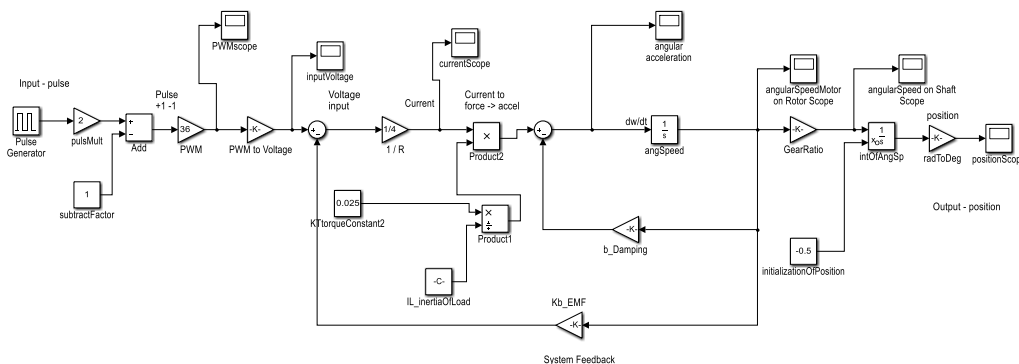


Figure 5. Simulation model of small DC motor in MATLA / Simulink.

The PID algorithm realized in the JavaScript/ECMAScript is shown in figure 6. The algorithm was executed in 30 ms loop using **setInterval** function on server side. It is important that the code of the control algorithm runs on the server side, in our case in node.js environment. One can observe the computation of the discrete integral as **errSum += err**. All their parts of the term defined by Eq. (1), i.e. $K_p E$, $K_i \text{ledt}$ and $K_d \text{De_dt}$ are summed together to provide the input to the H-bridge controller, i.e. $u(t)$. In the code this is **pwm** variable, which represents the Pulse Width Modulation input to the H-bridge controller. Logical statemets are used to determine the rotation orientation.

```
err = desiredValue - actualValue; // error as difference between desired and actual val.
errSum += err; // sum of errors | like integral
errSumAbs += Math.abs(err);
dErr = err - lastErr; // difference of error
KpE = parameters.Kp1*err;
KiLedt = parameters.Ki1*errSum;
KdDe_dt = parameters.Kd1*dErr;
pwm = KpE + KiLedt + KdDe_dt; // we use above parts
lastErr = err; // save the value of error for next cycle to estimate the derivative
if (pwm > pwmLimit) {pwm = pwmLimit}; // to limit pwm values
if (pwm < -pwmLimit) {pwm = -pwmLimit}; // to limit pwm values
if (pwm > 0) {board.digitalWrite(2,1); board.digitalWrite(4,0);}; // direction if > 0
if (pwm < 0) {board.digitalWrite(2,0); board.digitalWrite(4,1);}; // direction if < 0
board.analogWrite(3, Math.abs(pwm));
```

Figure 6. PID algorithm realization in JavaScript/ECMAScript.

Figure 7 shows the user interface, which is important in the process of system dynamics study. Important fact is, that the experiments are conducted on the live system. (1) represents the desired and actual value, i.e. input and output of the system. (2) shows the Error, i.e. difference between desired and actual value, (3) is integral part of the PID equation, (4) is the differential part of the PID equation. (5) shows $u(t)$, i.e. PWM input to the H-bridge controller and (6) shows the particular terms of PID equation, Eq. 1. One can observe, that by the proper values of K_p , K_i and K_d , the values of particular terms are normalized, having comparable range. (7) shows the integral of the absolute error which enables us to compare different parameter setups.

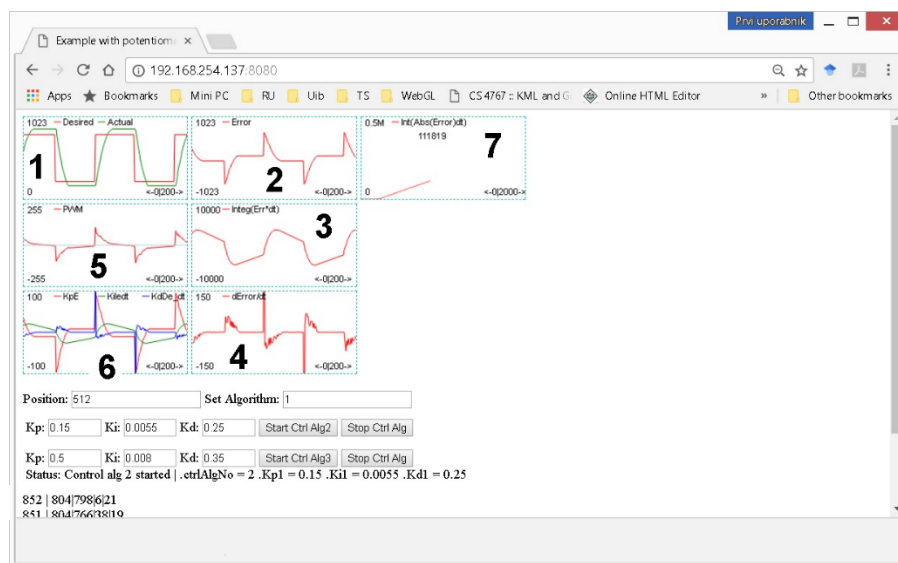


Figure 7. Graphical User Interface (GUI) of PID control setup in Web Browser Chrome.

One can set different values of the K_p , K_i and K_d parameters. Important part of the user interface are Start and Stop Control Algorithm buttons. Since the parameter values are not limited, the response of the system might be unpredictable, therefore it is important, that we are able to stop the control algorithm if necessary.

4. Discussion

Mastering PID control algorithm is a challenging task. New technologies, such as ARM Mini PC Raspberry Pi and GK802 with Arduino UNO enabled us to develop affordable control system study kit in the framework of Cyber-physical Systems. Interconnection with internet provides us with new opportunities to develop didactic environment and provide better understanding of the fundamental control system principles.

Important software technology that enabled us to developed described setup is node.js which provides us with the possibility to execute the code on the server side. Important fact is, that the entire control system together with the client side user interface was developed with one language, i.e. JavaScript/ECMAScript. This is important in order to reduce the complexity of the development task. The fact, that we have used widely adopted technology of Cyber-physical Systems and Internet of Things is also important, since the students should be familiar with the latest technology.

The new implementations will consider ESP8266 and ESP32 [13,14,15,17]. Important principle at the development of such student-oriented kits is affordability and novelty. This provides promising setup to study other algorithms as well as machine learning methods.

Acknowledgement

This work was supported by the Slovenian Research Agency (ARRS) within SI-MNE bilateral project “Development of Speech Controlled Wheelchair for Disabled Persons as Cyber-Physical System” Proj. No.: BI-ME/16-17-022, SI-RF bilateral project “Efficient Control of Cyber-physical Systems & Internet of Things by the Application of Evolutionary and Biologically Inspired Algorithms” Proj. No.: BI-RU/16-18-040, and Research program “Decision support systems in electronic commerce”, program No.: UNI-MB-0586-P5-0018.

References

- [1] Škraba A 2018 Development of educational cyber-physical Internet of Things platform study of the PID controller *7th Mediterranean Conf. on Embedded Computing (MECO)* pp 1-4
- [2] Škraba A, Stanovov V, Semekin E and Kofjac D 2016 Putting Cloud 9 IDE on the Wheels for Programming Cyber-Physical / Internet of Things Platforms - Providing Educational Prototypes *13th International Conf. on Informatics in Control, Automation and Robotics*
- [3] Škraba A., Stojanović R, Zupan A, Koložvari A and Kofjač D 2015 Speech-Controlled Cloud-Based Wheelchair Platform for Disabled Persons *Microprocessors and Microsystems* **39** 819-828
- [4] Koložvari A 2019 Speech-recognition cloud harvesting for improving the navigation of cyber-physical wheelchairs for disabled persons *Microprocessors and Microsystems* **69** 179-87
- [5] Sarhadi P, Noei A and Khosravi A 2016 Model reference adaptive PID control with anti-windup compensator for an autonomous underwater vehicle *Robotics and Autonomous Systems* **83** 87-93
- [6] Beschi M, Padula F and Visioli A 2016 Fractional robust PID control of a solar furnace *Control Engineering Practice* **56** 190-9
- [7] Wiliamowski B M and Irwin J D 2018 *Control and Mechatronics* (New York CRC Press)
- [8] Šafaric R, Truntic M, Hercog D and Pacnik G 2015 Control and robotics remote laboratory for engineering education *Int J Online Eng* **1** 1-8
- [9] Hale F 1988 *Introduction to control system analysis and design* (Englewood Cliffs NJ Prentice Hall)

- [10] Nise N 2015 *Control Systems Engineering* (Hoboken NJ Wiley)
- [11] Lyshevski S 2003 *Control systems theory with engineering applications* (Mumbai Jaico Pub. House)
- [12] Škraba A, Stanovov V and Semenkin E 2019 Modelling of DC motor and educational application in Cyberphysical systems *IOP Conference Series: Materials Science and Engineering* **537** 042008
- [13] Škraba A, Koložvari A, Kofjač D, Stojanović R, Stanovov V and Semenkin E 2016 Streaming pulse data to the cloud with bluetooth LE or NODEMCU ESP8266 *5th Mediterranean Conference on Embedded Computing (MECO)* pp 428-31
- [14] Škraba A, Koložvari A, Kofjač D, Stojanović R, Stanovov V and Semenkin E 2017 Prototype of group heart rate monitoring with NODEMCU ESP8266 *6th Mediterranean Conference on Embedded Computing (MECO)* pp 1-4
- [15] Škraba A, Koložvari A, Kofjač D, Stojanović R, Semenkin E and Stanovov V 2019 Prototype of Group Heart Rate Monitoring with ESP32 *8th Mediterranean Conference on Embedded Computing (MECO)* pp 1-4
- [16] Kofjač D, Stojanović R, Koložvari A and Škraba A 2018 Designing a low-cost real-time group heart rate monitoring system *Microprocessors and Microsystems* **63** 75-84
- [17] Alibaba's IoT Wrist Bands Based on ESP32 | Espressif Systems", Espressif.com, 2019 Retrieved from: https://www.espressif.com/en/media_overview/news/alibaba%E2%80%99s-iot-wrist-bands-based-esp32
- [18] Saeed K 2009 Can trend forecasting improve stability in supply chains? A response to Forrester's challenge in Appendix L of Industrial Dynamics *System Dynamics Review* **25** 63-78
- [19] 2014 *MATLAB and Simulink* The MathWorks (Inc., Natick, Massachusetts, United States)
- [20] Pololu DRV8838 Single Brushed DC Motor Driver Carrier, Retrieved from: <https://www.pololu.com/product/2990>
- [21] Oliva R and Vensim A 1995 Module to Calculate Summary Statistics for Historical Retrieved from: <http://metasd.com/wp-content/uploads/2010/03/D4584theil.pdf>
- [22] Sterman J D 1984 Appropriate Summary Statistics for Evaluating the Historical Fit of System Dynamics Models *Dynamica* **10(2)** 51-66