Tatrabot - a mobile robotic platform for teaching programming

Pavel Petrovič, *ppetrovic@acm.org*
Dept of Applied Informatics, Comenius University, Bratislava

We have developed a low-cost robotic platform that we use to teach first grade undergraduate students programming in C language. After the study program in Applied Informatics has been redesigned, our students learn programming in high-level programming language Python, which has automatic memory management, advanced data-structures and algorithms. However, some of our more advanced courses, such as Operating Systems show that the students lack the understanding for the bits and bytes, low level programming and miss the connection between the actual processes that occur in hardware and their software being executed on a machine. We think it is important to understand this level too, if the student is to program efficiently and correctly. Furthermore, playing with robots stimulates and motivates freshmen students, makes the study program more attractive, and allows more playful learning, improves the overall experience and diminishes the shock from entering the University. In this demonstration, we will show you the robot, its capabilities and how we have used it in our course on Computer Hardware.

Figure 1: Student exercise, participants programmed their robots to follow line and navigate a maze.

Keywords

robotics platform, constructionism, C language

The platform

The base of the robot consists of two plastic plates with many small holes that allow easy mounting of additional equipment. The plates are connected with usual metal spacers giving the construction a solid and robust appearance. Simple wheels are propelled by two DC motors, base is supported by a supporting rolling sphere. All of this comes as 2WD Beginner Robot Chassis V2 from DAGU China, available from RobotShop for about 15 Eur. The control platform is a tiny board with 32-bit microcontroller STM32F103 with a bootloader. It provides many more features than a popular 8-bit Arduino platform, it is available for less than 4 Eur, its CPU runs on more than 3-times higher frequency, and it can be programmed in standard C using free, but very convenient development environment ChibiStudio based on Eclipse IDE. The included library ChibiOS – provides interfacing with most of the devices that are already available on the single chip microcontroller such as timers, PWM controllers, input capture units, I2C bus, CAN bus, SPI, serial ports and more, making it very easy to attach sensors and actuators of all kinds with less technical effort. In our case, we have connected the following: SHARP distance sensors, two IR obstacle sensors, 4-channel bottom line sensor, motor encoders with wheels cut by laser in a local FABLab since those provided by DAGU are IR transparent, a panel with six signal LEDs, a reset and program button, simple piezo sound output, two user buttons, and an inertial unit that includes 3DOF accelerometer, gyroscope and magnetometer, temperature sensor and barometer. Programs are downloaded over USB cable that allows communication in terminal or any other application that can communicate over serial port such as Imagine Logo. The robot also has a BlueTooth allowing for remote control, easy debugging, and remote data collection. A small-size breadboard allows easy plug & play attachment of further sensors, such as ultrasonic distance sensor. Everything is powered by a set of 4 AA cells, giving at least 5 hours of operation. The complete system is built for about 65 Eur requiring less than 10 hours for assembly. We have purchased parts for 30 units, and we are about half through the assembly process, allowing for a use in a group of 20 students.

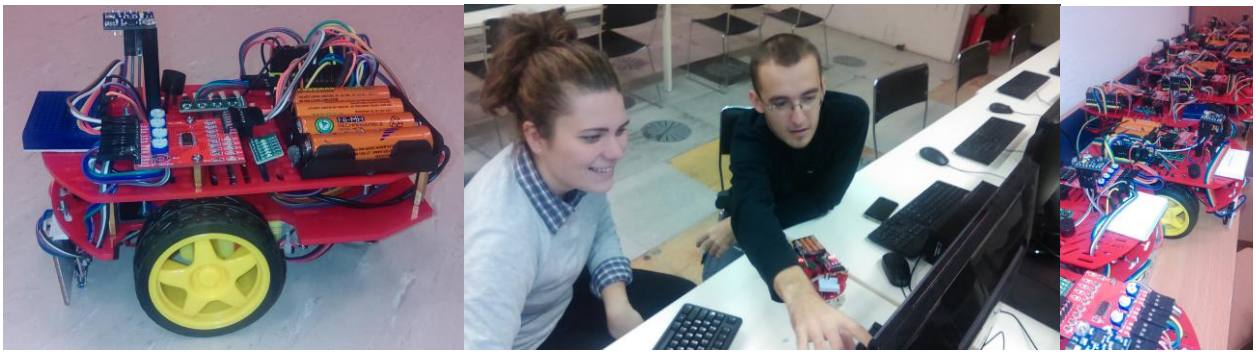

Figure 2: Tatrabot from side (left), students programming Tatrabot (middle), Tatrabots resting (right).

Experiences and Conclusions

The robots have been used in one freshmen course, a robotics seminar, where students made a program to solve sokoban puzzles and in an individual work with talented students. We have a set of functions that provide easy interfacing with the installed sensors. For instance, the current value of the left and right wheel encoders are in the global variables countA, countB, the current reading of the SHARP distance sensor is always available in the variable distance, to control speed of motors, one just needs to call a function set_motor(motor, speed), where the speed is 0-10000. We find the platform easy to use, easy to understand, but its advantage is that it is not a toy. We give the students a real thing. Both software and hardware is completely open, it is built from parts that can be used in industrial applications. All the learned material allows students to immediately continue developing their own embedded applications for a very low price. Yet, we adhere to the principles of constructionism. Students learn by doing and playing, they can come with creative designs, solutions, projects, interact in groups and have fun doing it. The documentation site for this robot is dai.fmph.uniba.sk/projects/tatrabot/