MiniDelta - Educational Robot with Parallel Kinematic Structure

Pavel Dolejší*, Václav Krys*, Vladimír Mostýn*
* VŠB – Technical University of Ostrava, Department of Robotics, Ostrava, Czech Republic
pavel.dolejsi@vsb.cz; vaclav.krys@vsb.cz; vladimir.mostyn@vsb.cz

Abstract—**Educational robot with parallel kinematic structure consisting mostly of commercial off-the-shelf components controlled remotely from a computer using a touchscreen. The robot is intended for education and presentation of technology and automation. The control system and user interface have been designed with emphasis on simplicity and intuitiveness, while offering a wide range of options and a gradually increasing learning curve.**

I. INTRODUCTION

Modern technologies, automation and robotics became a normal part of the modern world in the past decades. They are applied to various degrees across all fields, not only engineering. This growth is connected with a need of qualified professionals in the field, as well as of educated people with general understanding of the principles. The problem is that even though there has a been a slight improvement, automation and robotics in particular still remain a very expensive area that is commonly not available to general public.

Nowadays, technical universities are usually not equipped with specialized laboratories with industrial robots or other advanced tools that would prepare future generations of experts in the field of robotics. The capacity limitations prevent a more extensive education of students across specializations or even fields. These students then enter work processes with only shallow knowledge of the principles and possibilities of automation. In some cases, the students may at least attend brief introductory courses but for most of them, robotics remains a mysterious branch bordering on fiction.

The situation is not much better for students who decided to devote their professional life to robotics and who were given the opportunity to work on specialized machines. Industrial robots are designed to be operated mostly in practice, which is why their main criteria are functionality and efficiency. Even though the situation has been changing slowly in recent years, user-friendliness has never been a key parameter. Many students of robotics then experience a certain "cultural shock" when working in laboratories, as they not only need to learn to work with the robotic system but also understand the logic of the control system and often also an independent programming language. This represents a steep difficulty curve which leads to a waste of time and resources.

Both these problems could be solved if students in lower levels of their studies at least came in contact with robotics and automation and could learn the very basics.

This would require an educational robotic system with a simple technology, robust yet illustrative and with a well-designed control system accessible even to beginners. Such a system could be used for general introduction of university students, as well as lower degree students, to the world of technology and technical fields.

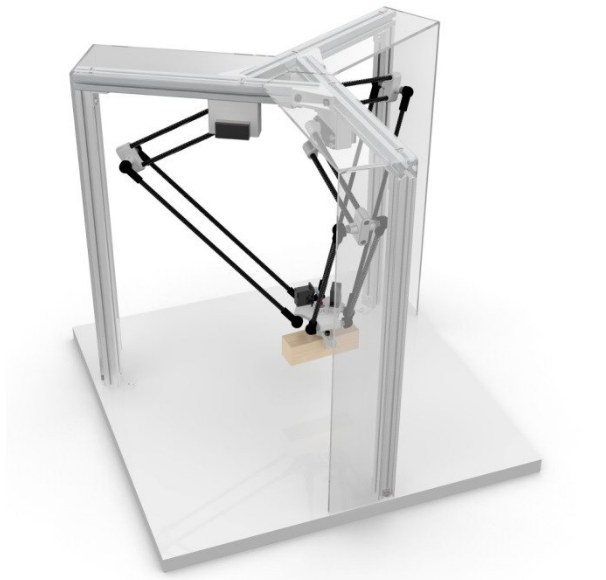

Figure 1. MiniDelta robot

This educational robot, named MiniDelta, was built in 2013 at Technical University of Ostrava.

II. MECHANICAL SUBSYSTEM

MiniDelta is a purpose-designed educational robot with parallel kinematic structure. Modular blocks of serial robotic arms are already commercially available on the market. However, one new trend of industrial robotics are parallel robots, for which there are no commercial educational variants. This is why a completely new Delta robot was built with three arms and four degrees of freedom. The main requirements on the design were low purchase costs, which resulted in the wide application of commercial off-the-shelf (COTS) components intended namely for models, as well as low spatial requirements and accuracy of the robot. These requirements determined the final size and weight of the robotic system.

Built robot (Fig. 1) takes up space corresponding to a cube with a side length of 0.5 m. The robot weighs about 5.2 kg, whereas movable parts weigh only 230g (the handled object is not included). The complex robot workspace is limited by computer into a cylinder with a diameter of 300 mm and height of 80 mm (Fig. 2).

978-1-4799-8221-9/15/$31.00 ©2015 IEEE

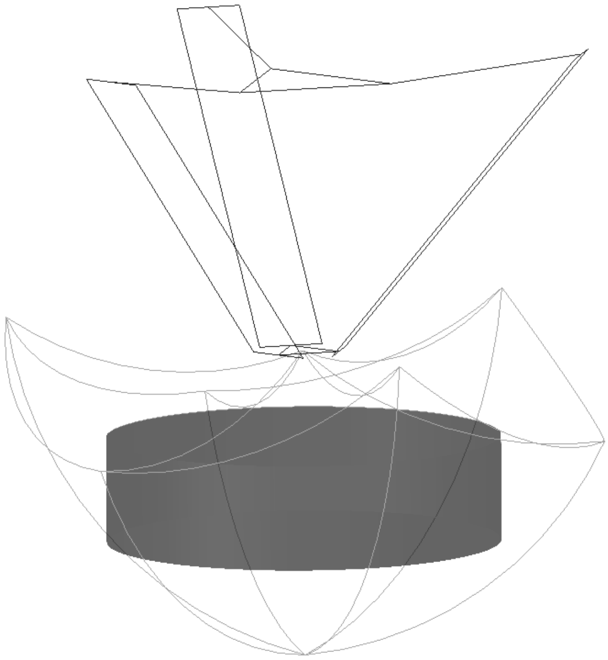

Figure 2. Robot workspace

The maximum nominal load of the robot is 100 g, whereas the main limitation is the load capacity of the end gripper. Just like other commercial delta robots, MiniDelta has 4 degrees of freedom, three parallel translations and one serial rotation of the end effector. The first prototype had a simple mechanical gripper as the end effector; other prototypes were provided with a lifting electromagnet.

The supporting structure of the robot consists of aluminum profiles complemented with 3D printed (polycarbonate) parts. Movable arms and the end platform are mostly made of carbon composites with 3D printed parts. All five drives of the robot consist of model servos of different sizes and parameters. Availability of 3D print made the designing of the robot much easier; however, due to certain disadvantages, an alternative construction with a lower share of 3D printed parts, which were replaced namely by simple steel sheet parts, was also created and tested on the second prototype.

III. Electric Design And HW Layer of Control System

The fundamental decision in the design was the use of COTS modelling servos as drives of the robot. Servos represent a compact block consisting of the motor, gear box and control electronics. This decision resulted in a simplified control system and had a positive impact on the price of the whole system, as it was not necessary to develop our own, less compact solution or purchase significantly more expensive professional systems. Nonetheless, the cost of this solution is relatively high. It consists of a lower accuracy of the robot and most importantly absence of any feedback. Servo circuits autonomously control the servo motion, yet are not programmed to provide feedback. A variant solution was designed for modification of servos that would partially solve the problem; however, at the end we decided against this solution in favour of compatibility with other servos.

The solution with modelling servos removes the whole motor control block from the design. Instead it suffices to send the desired rotation position to servos in regular intervals. The input of servos is a time-coded signal. This signal can be generated by almost any microprocessor with appropriate libraries; however, we decided for another COTS product, concretely the Pololu MiniMaestro 12 controller, due to its accuracy and simple integration. This board is based on the PIC microprocessor and is designed directly for controlling a larger number of servos. In accordance with data provided by the manufacturer, it can generate a signal with the accuracy of 0.25 ms at a refresh rate of 333Hz. In addition, it also provides the option to connect to a PC by a USB cable or UART serial port. Both can be used to control the robot from a PC, either via a cable (USB) or a wireless module (UART). The wireless module consists of a simple converter from a serial port to Bluetooth via the SPP protocol. The SPP protocol ensures that the Bluetooth module is used as a bridge and that devices on each side see only the standard serial line.

All devices in the robot have been designed for a uniform power supply of 5V and are powered from a shared adapter. In case of wireless connection with the computer, the robot is connected only by a single cable to the power network. A solution variant with a battery has also been designed, resulting in a completely wireless robot, but it has not been realized in the final design.

The MiniMaestro board only generates control signals for servos on the basis of input data from the superordinate system. The original robot design included additional Atmel 8bit processor that would ensure the internal logic of the robot. However, this solution was rejected due to insufficient performance; instead, the external control computer is used directly as the superordinate system for the MiniMaestro board. Reconsideration of this design and it realization with a higher-performance processor is an open possibility for future versions of the robot.

Due to the absence of an independent logic process in the robot, the control system needs to also include the control computer, which provides not only the graphical user interface of the robot but also all robot motion control logic. Basically any device with an USB port or Bluetooth SPP can be used as the control computer. For practical reasons, we decided for a PC platform with Windows operating system. Use of other platforms, such as Android, is an open possibility for future versions of the robot.

IV. Logic Level of The Control System

Logic control is done exclusively via software on the superordinate computer. In practice, it consists of several consecutive steps. The first step is the receipt of the target coordinates in the workspace. These coordinates may be submitted either directly by the user, via user interface, or by the function generating a sequence of coordinates. The target coordinates represent a point, to which the end effector is to move at a concrete moment. In the next step, a transformation function is applied to the coordinate, which is divided into three partial coordinates relevant to individual arm motors. Possible calibration corrections are also applied within this step. From this moment on, transformations for individual arms are independent of each other.

The third step is calculation of the inverse function. Its output is the required rotation of servos in degrees. The inverse function is realized by the vector method, which has proven to be the most suitable for the Delta robot. The inverse task calculation is based on [1]. With certain simplifications, the calculation of the inverse function represents the calculation of a quadratic equation with coordinates and arm lengths as input parameters. In ideal conditions, the output of this quadratic equation may be two real numbers; however, for singularities or when the point is outside the reach of the arms, there may be one or none solution in the set of real numbers. Normally, such sub-standard solutions could constitute a problem; however, the workspace of the robot has been designed so that valid points in the workspace correspond only to two real numbers. Points in the workspace always have one solution in the allowed range of servo rotation (+10°/-80°) and one outside the range. The value in the allowed range is then sent to the next step.

The fourth step is the conversion of calculated angles of servo rotation to input variables of the MiniMaestro controller and creation of a data string for sending. MiniMaestro accepts numerical values between 2800 and 9200; simple linear transformation suffices for the conversion. The resulting calculated value can be sent directly to the robot. However, to ensure simultaneous movement of all three arms and efficient use of system tools at the same time, the calculations of all three arms are again joined together at this point and the values are sent as a single string. One example of such a string is:

0x9F, 0x02, 0x03, 0x00, 0x00, 0x70, 0x2E

This 7-byte string would change the position of two servos. Each position is determined by two bytes and additional three control bytes. For three servos the string would be 9 bytes. MiniMaestro verifies the form and range of input data and executes the order only if all three values were received correctly. From the perspective of the control program, the values are sent to a virtual serial (COM) port, where they are accepted by the operating system and sent either to a USB port or via Bluetooth, depending on the circumstances.

The whole logic control string occurs periodically in an infinite loop. The selected call period is 3ms, which represents the shortest interval that can be processed by the combination of MiniMaestro and servos. This short interval was selected in an effort to achieve the maximum apparent fluency of the robot movement and short reaction time. At the same time, the time represents a fixed limit, during which the calculation needs to be completed. This does not constitute a fundamental problem for a PC. Practical tests have shown that the whole loop can take place in a couple of microseconds. However, this has proven to be an insurmountable obstacle for the considered Atmel 8bit on-board processor, since these time limits could not be met even after optimizations. This is why it was decided to leave the logic level to an external computer.

In the context of PC running on Windows, one problematic area needs to be pointed out. As explained in the previous paragraph, meeting exact time intervals is necessary for controlling the robot. The Windows system is in no way a RTOS (Real-time operating system) and meeting of time limits is therefore impossible in principle. This would be unacceptable for an industrial robotic system. However, for the MiniDelta robot it was accepted as the price for simplicity and availability. Slightly worse functionality when the computer is busy is not perceived as a critical error of the system. Practical tests have shown that a modern PC offers ample performance and does not cause any noticeable negative impact on the robot functionality.

V. User Interface And User Modes

The graphical user interface is part of the complex computer software, which also provides logic control. Since the logic loop is called in regular intervals as the top-priority function, it is relatively independent of the rest of the program. At present it can be separated from the rest only at the source code level. The control program is written in the C# language for the Windows Desktop platform, so that it can be operated on classic PCs without a touchscreen, when required. However, the interface has been designed for working on touchscreens, especially on tablets. This adaptation consists namely in larger control elements. With regards to the increasing popularity of the Windows 8 system - and Windows 10 in the future - it is an open question whether the application should be transferred to the native touch interface of Windows 8 within future development.

So far, two versions of the program have been developed. The older version was used mostly for development and testing of the robot and consisted of a single window that contained all control elements of the robot. It included the option of direct control of the point in the workspace based on cartesian and cylindrical coordinates, as well as a basic menu for creation of movement sequences. The problem of this program was its complexity and user-unfriendliness. Even though after training the students were able to control the robot, they complained about certain limitations and general non-intuitiveness of the program. This is why the comments were implemented into a new, user-friendlier version of the user interface.

The new program was based on the same calculation core, therefore the resulting movement options of the robot did not significantly change. The interface was divided into several different modules that offer new modes of user inputs. Each module has an independent screen with minimized number of control elements and intuitive interface adapted for specific tasks. Individual modules can be also clearly scaled based on their complexity allowing students to increase their knowledge while working interactively with the robot and gradually progress to more complex control methods. The simplest modules are also available to lower level students, e.g. at elementary schools.

The current program version is available only in Czech language. It can be easily localized to other languages.

The home screen contains a simple menu, through which the robot is paired with the computer and then one of the created modules is selected. Currently there are four modules. Modules Chessboard, Teleoperation, Direct Control and GCode.

The first and simplest module of the program is the Chessboard module. The dominant element of this module window is a matrix of 64 buttons arranged as a chessboard. The module assumes the presence of a printed chessboard in the robot's workspace. After pressing any of the chessboard on the screen, the robot moves automatically to the selected field and performs a picking-up/laying-down operation of the object. The whole sequence of movements is fully automatic and no other input is required from the user. This represents a limitation of the robot's options to only one simple handling operation; on the other hand, it is a very user-friendly solution suitable namely as a first introduction with the robot or for elementary schools.

The second module is the Teleoperation module. Its dominant feature is a large coordinate shape, which corresponds to the shape printed on the base of the robot. This module simulates an industrial teleoperator, as it allows controlling the final point in real time by moving the cursor around the coordinate shape. This is a very efficient control method, specifically on a tablet screen. Just like the previous module, this module has a very limited functionality. Controlling a point in real time with decent robot dynamics is very attractive and this module is very popular for simpler tasks. In the robot variant with an electromagnet, several physical tasks have been designed in addition to the simple handling tasks in order to extend the application options of the robot.

The third module is called Direct Control. Its concept stems from an older single-window version of the program, and just like this version, it provides direct control of all coordinates of the robot. Main translation movements may be controlled in the Cartesian or cylindrical coordinate system.

In addition to control of axes, this module also allows creation of a sequence of movements and their replaying. Due to high complexity of this module and a large number

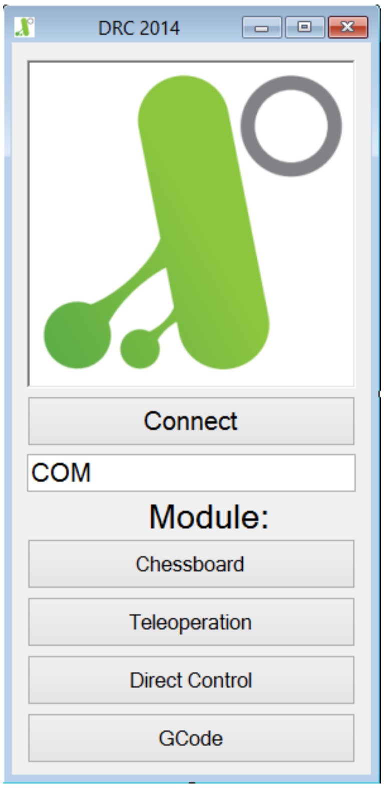

Figure 3. Module selection menu

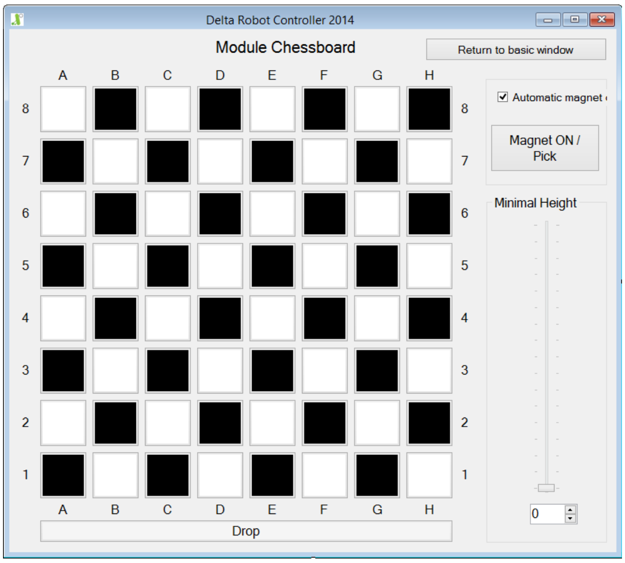

Figure 4. Module Chessboard

of elements, Direct Control is not as intuitive as the previous two modules and is intended for more experienced students. On the other hand, it offers the widest range of options and its controlling is similar to industrial robots in certain aspects.

The fourth and last module is the GCode module. It was created on the basis of our experience that industrial robots and automated machines are often programmed using external tools and specific programming languages. This module works as a simple interpreter of the G-Code language (ISO 6983). G-Code was selected for its simplicity, as well its frequent use in the field of CNC machines and lately also with 3D printers. The module offers only basic control elements for running and stopping the external code and is currently able to interpret only selected movement commands. The development of this module is still in progress and it is planned to add other functions suitable for the MiniDelta robot.

The four described modules are currently being realized

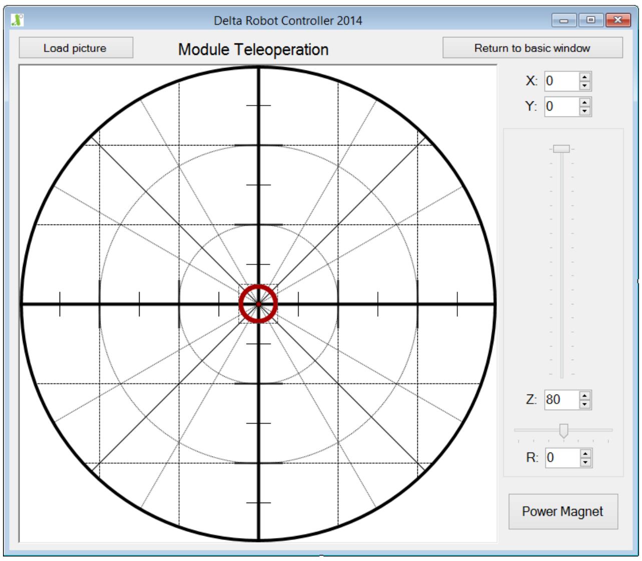

Figure 5. Module Teleoperation

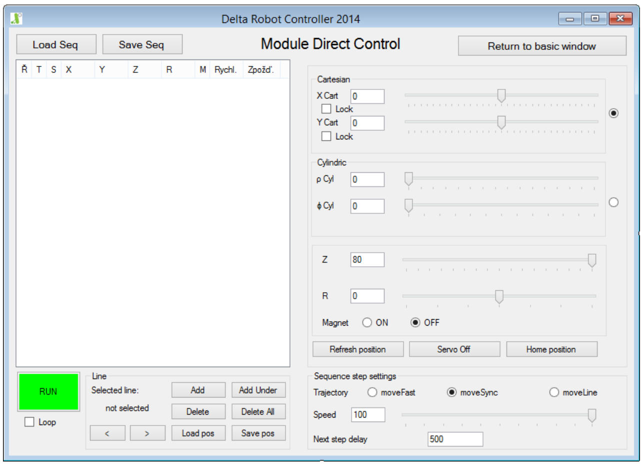

Figure 6. Module Direct Control

and are running in a trial operation. Based on experience with the robot operation, there may be further modifications and extensions, or new modules may be added. All modules represent only a graphical (or functional) extension of the shared control loop running in background.

VI. Application

Concrete applications of robots in education are still an open question, which will be evaluated on the basis of practical experience from operation. The proposed methodology, however, assumes using one robot in a smaller group of students. One option is that the robot will be connected by a cable to a classic desktop computer and the students will take turn operating the PC. However, a more efficient and attractive way is controlling the robot from a lighter tablet with a touchscreen using wireless connection. At present, the program is limited to the Windows Desktop platform, which narrows down suitable tablets to the more expensive models. Rewriting the program to the more available Android platform is currently an open possibility.

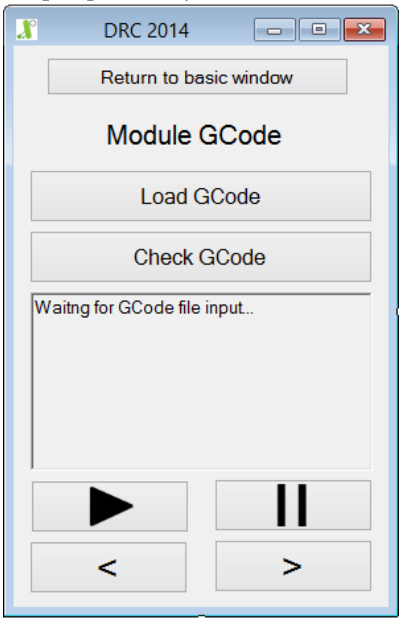

Figure 7. Module GCode

As a part the educational program, a methodology [2] has already been processed for the entry of the robot into test operation on selected Czech high and elementary schools. The methodology includes general information about the robot, sample technical documentation and examples of tasks. Further development of the MiniDelta robot will be determined on the basis of results from the operation.

VII. Conclusion

The MiniDelta robot built a year ago at the Technical University of Ostrava has met all expectations. A risky approach focusing on COTS components even at the price of slightly worse parameters has paid off and resulted in a robot with low purchase costs and high appeal.

Observed mechanical parameters of the prototype are full movement capability within the projected cylindrical workspace, with repeatable positioning precision of 1 mm. Translational velocity remains above 1 m/s even under full load, which corresponds to 100 g. With reduced velocity, loads up to 500 g are deemed possible.

Based on the experience with operating the first prototype, a number of improvements have been designed and some of them were implemented to the second robot prototype. Other suggestions are still being processed and the mechanical construction of the MiniDelta robot may still undergo partial changes.

In terms of electronics, the used simple structure has proven to be most suitable. A weak point of the robot is the lack of feedback from servos and absence of an independent logic processor. Removal of these weak points is the aim of further development of the robot. The current solution with logic control on an external computer works fine despite the lack of RTOS and no significant problems have been discovered. This is probably caused by the excess of performance of modern computers.

When evaluating the robot from the point of view of expected parameters and usability, MiniDelta is considered a success, even though in few points results proved to be below expectations. Namely problematic was found the repeatable precision of the robot. Expected precision was within tenths of mm, while significant plays and imprecisions only barely allow 1 mm precision. However for expected purpose was this found to be still acceptable, as high precision is not required for educational purposes and is justified by low cost of the robot. All other parameters allow use of the robot for proposed role as the educational tool without relevant limitations.

The current version of the control program of the robot meets all requirements on simplicity and user-friendliness, while being able to offer complex control methods. The program development is not yet completely finished and slight changes may be expected. However, it is ready for the first phase of the trial operation with students of primary and secondary education. Even after completion of the current program version, the robot software will remain the most actively developed part of the robot. A new module may be added to the four current modules with control modes or the robot may be applied to a completely different field through SW. Detailed options are being investigated and will be thoroughly documented if they are realized.

In addition to the already mentioned first prototype, there is also a group of MiniDelta robots with technology corresponding to the second, improved prototype. These robots will be tested in trial operation among students. It is hard to predict concrete conclusions from the trial operation but it is certain that they will have a significant impact on the development of the MiniDelta robot. The work of students with robots lends the originally technical issue an interesting pedagogical overlap. The impact of robots on efficiency of education would probably deserve its own study, which would, however, be beyond the scope of the technological development of the robot.

ACKNOWLEDGMENT

This article has been elaborated in the framework of the project Opportunity for young researchers, reg. no. CZ.1.07/2.3.00/30.0016, supported by Operational Programme Education for Competitiveness and co-financed by the European Social Fund and the state budget of the Czech Republic.

This article has been also supported by specific research project SP2014/176 and financed by the state budget of the Czech Republic.

REFERENCES

[1] J. Merlet, “Parallel Robots,” Dordrecht: Kluwer Academic Publishers, 2000, 355 p.

[2] Z. Duffek, P. Kopec and P Dolejší, “Metodika práce s robotem MiniDelta pro SŠ,” VSB TU: Ostrava, 2014 [Czech]

[3] P. Dolejší, *Design of the Parallel Manipulator with Four Degrees of Freedom*. Ostrava, 2013. Diploma thesis. VŠB –Technical University Ostrava, Faculty of mechanical Engineering, Department of Robotics. Supervisor: prof. Dr. Ing. Vladimír Mostýn

[4] M. Stock and K. Miller, "Optimal kinematic design of spatial parallel manipulators: Application to Linear Delta robot." Journal of mechanical design 125.2 (2003): 292-301.

[5] A. Khouki, “An optimal time-energy control design for a prototype educational robot.” Robotica, 2002, 20.6: 661-671.

[6] L. Tsai. “Robot analysis: the mechanics of serial and parallel manipulators.” New York: Wiley, 1999.

[7] F. Pierrot, C. Reynaud and A. Fournier. "DELTA: a simple and efficient parallel robot." Robotica 8 (1990): 105-109.