

Abstracts of Dissertations

Institute of Information and
Communication Technologies

BULGARIAN ACADEMY OF
SCIENCES



8 / 2024



MULTIPURPOSE
TELE-OPERATED
SERVICE ROBOT

Petko Stoev

МНОГОЦЕЛЕВИ
ТЕЛЕ-УПРАВЛЯЕМ
СЕРВИЗЕН РОБОТ

Петко Стоев

Автореферати на дисертации

Институт по информационни и
комуникационни технологии

БЪЛГАРСКА АКАДЕМИЯ НА НАУКИТЕ

ISSN: 1314-6351

Поредицата „Автореферати на дисертации на Института по информационни и комуникационни технологии при Българската академия на науките“ представя в електронен формат автореферати на дисертации за получаване на научната степен „Доктор на науките“ или на образователната и научната степен „Доктор“, защитени в Института по информационни и комуникационни технологии при Българската академия на науките. Представените трудове отразяват нови научни и научно-приложни приноси в редица области на информационните и комуникационните технологии като Компютърни мрежи и архитектури, Паралелни алгоритми, Научни пресмятания, Лингвистично моделиране, Математически методи за обработка на сензорна информация, Информационни технологии в сигурността, Технологии за управление и обработка на знания, Грид-технологии и приложения, Оптимизация и вземане на решения, Обработка на сигнали и разпознаване на образи, Интелигентни системи, Информационни процеси и системи, Вградени интелигентни технологии, Йерархични системи, Комуникационни системи и услуги и др.

Редактори

Геннадий Агре

Институт по информационни и комуникационни технологии, Българска академия на науките
E-mail: agre@iinf.bas.bg

Райна Георгиева

Институт по информационни и комуникационни технологии, Българска академия на науките
E-mail: rayna@parallel.bas.bg

Даниела Борисова

Институт по информационни и комуникационни технологии, Българска академия на науките
E-mail: dborissova@iit.bas.bg

Настоящото издание е обект на авторско право. Всички права са запазени при превод, разпечатване, използване на илюстрации, цитирания, разпространение, възпроизвеждане на микрофилми или по други начини, както и съхранение в бази от данни на всички или част от материалите в настоящето издание. Копирането на изданието или на част от съдържанието му е разрешено само със съгласието на авторите и/или редакторите

The series Abstracts of Dissertations of the Institute of Information and Communication Technologies at the Bulgarian Academy of Sciences presents in an electronic format the abstracts of Doctor of Sciences and PhD dissertations defended in the Institute of Information and Communication Technologies at the Bulgarian Academy of Sciences. The studies provide new original results in such areas of Information and Communication Technologies as Computer Networks and Architectures, Parallel Algorithms, Scientific Computations, Linguistic Modelling, Mathematical Methods for Sensor Data Processing, Information Technologies for Security, Technologies for Knowledge management and processing, Grid Technologies and Applications, Optimization and Decision Making, Signal Processing and Pattern Recognition, Information Processing and Systems, Intelligent Systems, Embedded Intelligent Technologies, Hierarchical Systems, Communication Systems and Services, etc.

Editors

Gennady Agre

Institute of Information and Communication Technologies, Bulgarian Academy of Sciences
E-mail: agre@iinf.bas.bg

Rayna Georgieva

Institute of Information and Communication Technologies, Bulgarian Academy of Sciences
E-mail: rayna@parallel.bas.bg

Daniela Borissova

Institute of Information and Communication Technologies, Bulgarian Academy of Sciences
E-mail: dborissova@iit.bas.bg

This work is subjected to copyright. All rights are reserved, whether the whole or part of the materials is concerned, specifically the rights of translation, reprinting, re-use of illustrations, recitation, broadcasting, reproduction on microfilms or in other ways, and storage in data banks. Duplication of this work or part thereof is only permitted under the provisions of the authors and/or editor.



BULGARIAN ACADEMY OF SCIENCES

Abstract of PhD Thesis

MULTIPURPOSE TELE-OPERATED SERVICE ROBOT

Petko Ivanov Stoev

Supervisor: Prof. Nayden Shivarov

Approved by Supervising Committee:

Assoc. Prof. Martin Kambushev

Assoc. Prof. Stanislav Gyoshev

Assoc. Prof. Stefan Biliderov

Prof. Marin Marinov

Corr. Member Lyubka Doukovska



**INSTITUTE OF INFORMATION AND
COMMUNICATION TECHNOLOGIES**

Department of Cyber-Physical Systems

Relevance of the dissertation topic

Robotics and, in particular, service robots are an important part of strategic policies and documents for the development of the technological sector: "Society 5.0", "Industry 4.0", "Industry 5.0", PROGRAM 2030 of the United Nations, Strategic Program of the European Commission for the period 2019-2024, "2030 Digital Compass: Europe's Path to the Digital Decade", "European Digital Strategy", "Europe Fit for the Digital Age", Horizon Europe Programme, European Green Deal, EU Circular Economy Action Plan. Their development and implementation is a major factor in advancing technology, improving safety, increasing efficiency, and addressing important societal, economic, and environmental challenges. As a rapidly developing field, multi-purpose tele-operated service robots will have an increasing presence and play an increasingly important role in the modern world.

Purpose and tasks of the dissertation

The aim of the dissertation is the development of a multi-purpose tele-operated service robot, designed to assist people in their work related to patrolling, monitoring, guarding and signaling objects, cleaning and sanitizing premises, lifting and carrying loads, distribution of medicines, transportation and storage of products, performing manipulation operations with elements.

The development addresses the following problem areas of multipurpose service robots: mechanical design, sensor system, industrial design and ergonomics, actuators and actuators, communication and telecontrol, human-robot interaction.

The achievement of the set goal is related to the implementation of the following specific tasks:

1. Getting to know the main characteristics and specific features of tele-operated service robots
2. Designing the main components of the mechanical design of the robot:
 - Universal mobile platform
 - Module with lifting platform
 - Rotary table module
 - Cleaning module
 - Scissor lift module
 - Niryo anthropomorphic robot module
3. Choice of drive, actuators and sensor system of the robot
4. Development of industrial design of the robot
5. Development of remote control of the tele-controlled multifunctional mobile robotic platform
6. Creating robot operating modes
 - Development of control algorithms
 - Compilation and implementation of computer simulations and verification of control algorithms

PUBLICATIONS OF THE PHD STUDENT ON THE THEME OF THE DISSERTATION

At international conferences abroad:

Stoev, P., Chikurtev, D., Stefanov, T., Dimitrov, D. & Vitanova, D. (**2023**). Remote Control of a Teleoperated Multi-Purpose Mobile Robot Platform Using a Web-Based Graphical Interface, via MQTT and Web Sockets. The 3rd International Conference on Electrical, Computer, Communications and Mechatronics Engineering (ICECCME) 2023

Stoev, P. , Yovchev S., Chikurtev, D., Chivarov, N. Development of a Teleoperated Mobile Robot with Module for Cleaning Animal Husbandry Complexes. TechSys **2022** , Accepted for publishing

Nayden Chivarov, Denis Chikurtev, Petko Stoev, Vasil Lozanov, Stefan Chivarov. ROBCO Drone - Service Robot for Transport and Delivery of Grocery Products, 7th International Conference on Engineering and Emerging Technologies **2021** – ISTANBUL – TURKEY

At international conferences in our country:

Stoev, P. Development of a Lifting Module of a Mobile Robot for Warehouses. ADP **2022** , Pages 259-261, ISSN: 2682-9584

Vasil Lozanov, Nayden Shivarov, Denis Chikurtev, Stefan Shivarov, Petko Stoev. Industrial design of service warehouse robot "ROBKO", Annual international scientific conference of BAFA "Georgi Benkovski" **2021**

Citations :

Nayden Chivarov, Denis Chikurtev, Petko Stoev, Vasil Lozanov, Stefan Chivarov. ROBCO Drone - Service Robot for Transport and Delivery of Grocery Products, 7th International Conference on Engineering and Emerging Technologies **2021** – ISTANBUL – TURKEY

INTRODUCTION

The multi-purpose tele-controlled service robot is a versatile robotic system designed to perform a wide range of tasks and services in different fields and areas through human-to-operator remote control.

This dissertation discusses fundamental problems related to mechanics, industrial design, ergonomics, mechanical compatibility and telecontrol of multipurpose tele-controlled mobile robots. The development of a multi-purpose tele-controlled service robot is presented, through which can be carried out, monitoring, security and signaling of objects, cleaning and sanitation of premises, lifting and carrying cargo, distribution of medicines, transportation and storage of products, performing manipulation operations.

The present dissertation work is structured as follows:

CHAPTER 1: The main aspects of the mechanics and design of multi-purpose tele-controlled service robots are discussed, including the specific features and systems of service robots, mechanical compatibility and mechanical systems for their movement, the design and ergonomics of robots, communication in service tele-controlled robots. The specificity of human-robot interaction in the context of multi-purpose tele-guided service robots is indicated.

CHAPTER 2: In Chapter Two, the design of the developed multi-purpose tele-controlled service robot is presented. The mechanical system of the robot, its actuator and actuators, its sensor system and the realized design are described in detail.

CHAPTER 3: Chapter Three presents the control of the multi-purpose robot, and for each mode and selected module are developed by the doctoral student algorithm of operation, software and hardware control. The remote control of the teleoperated multifunctional mobile robot-platform is considered. Simulations were conducted in a computer environment, on the basis of which the control algorithms were verified.

CHAPTER I: THEORETICAL OVERVIEW

1.3 Main features of multipurpose robots

A key feature of multi-purpose robots is their modular design, which enables users to easily add or remove components. This modularity allows the robot to be customized for specific tasks and makes the implementation of improvements or adaptations easier. Multi-purpose robots are equipped with a variety of handling elements, such as robotic arms with multiple degrees of freedom and adaptive grippers. Through this flexibility, they can handle different objects and perform a variety of tasks. Depending on the applications envisaged, multi-purpose robots may have different mobility systems. The choice of mobility depends on the terrain and the tasks that the robot will face. Multi-purpose robots include a variety of sensors for perception, navigation and interaction with the environment. Safety is paramount, especially when robots work alongside humans. Advanced algorithms for navigation, obstacle avoidance and decision making contribute to their autonomy. Energy-efficient components,

power management systems and battery technologies are integrated into mobile robots to optimize the robot's energy consumption. Multi-purpose teleoperated robots have advanced communication capabilities, including Wi-Fi, cellular and satellite connectivity, to achieve remote monitoring, control and data transmission. A new aspect related to the design of robots and in particular multi-purpose teleoperated robots is the integration of AI in order to improve their adaptability and decision-making capabilities. AI allows robots to learn from experience and improve their performance over time.

1.5 Mechanical compatibility

Mechanical compatibility is an essential aspect in the design of modular robots. It refers to the compatibility between the mechanical parts of the robot and the modules that attach or remove. Mechanical compatibility includes dimensional compatibility, strength and load-bearing capacity, standardized interfaces, service.

1.7 Design, ergonomics and regulatory requirements

As a mediator, design plays an essential role in building successful human-robot communication. It represents that specific link without which the perception, and hence the approach of man to the robot, would be different. Robot design is a responsible task, with numerous parameters to be tailored taking into account the environment in which the robot will be placed in order for human-robot communication to be carried out correctly. To unify and lay the foundations of standardization in the design of robots, ergonomic requirements, psychological aspects of shapes and colors, their psychophysiological impact and the emotions and actions of the user generated on this basis are considered.

1.9 Tele-control of service robots

Through telecontrol, service robots are expanding their implementation environments and the functionalities they can have. In the context of service robots, tele-control refers to the remote control of robot movements and actions by operator-human using a control interface. There is a wide range of remote control and guidance methods for robots, each method having its own specific advantages and challenges, contributing to the diverse range of telecontrol options.

1.10 Human-Robot Interaction in the Context of Multi-Purpose Tele-Operated Service Robots

Human-robot interaction is a multidisciplinary field whose goal is to achieve safe, productive and seamless collaboration between humans and robots. In the context of multi-purpose tele-operated service robots, the main directions for achieving effective human-robot interaction are related to the design of intuitive control interfaces that are convenient for the operator and require minimal effort; reducing the operator's cognitive load by introducing shared control modes where the robot autonomously assists in navigation and obstacle avoidance; providing real-time feedback to operators, including live video images from the robot's cameras, sensor data and status updates, haptic feedbacks; implementing safety mechanisms, including emergency stop buttons and obstacle detection systems, to ensure the safety of both the operator and the robot; implementation of encryption and access control, especially in cases where the robot captures sensitive data or works in a special environment; responsible use of robots in the context of ethical aspects of human-robot interaction, such as robot behavior, decision-making, and accountability.

CHAPTER II: DESIGN OF A MULTIPURPOSE TELE-OPERATED SERVICE ROBOT

2.1 Mechanical design

2.1.1 Universal mobile platform of the multi-purpose robot

The developed platform (Fig. 12) aims to enable the easy installation of additional modules that placed on the robot to allow the execution of specific tasks in different environments and to expand its functional capabilities. Depending on the module that connects to the platform, it can be with two, three or four plates. They are made of high density polyethylene (HDPE). The thickness of each of the plates is 9.5 mm and the end points for each plate describe a circle of 450 mm diameter. Each of the plates is fixed to the next plate by means of a group of four rods, 12.5 mm in diameter on each of the rods.

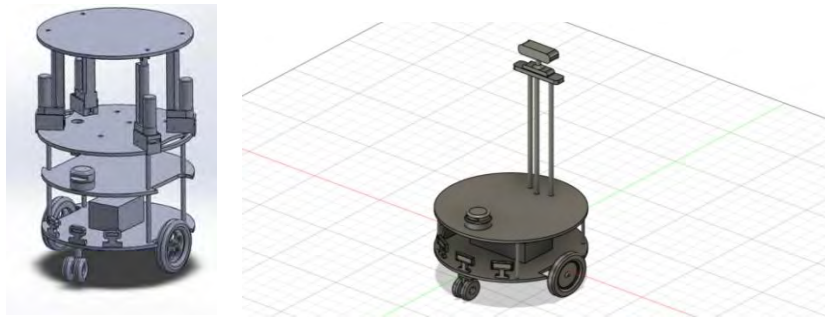


Fig.12 3D image of the designed mobile robotic platform

The distance the bars provide between the plates is as follows: 152 mm between the first and second and 127.5 mm between the second and third, the plates being numbered in order starting from the surface on which the robot is placed. Below the first plate is an additional plate measuring 336.6 mm x 177.8 mm with a thickness of 9.5 mm. The distance between the additional plate and the first plate is 78 mm measured from the bottom surface.

The height measured from the surface on which the robot is placed to the top of the first plate is 106 mm. In the first and second plate, an incision measuring 165 mm x 38 mm is made externally, tangentially to each of the plates. The cut in the first plate is used to place the drive wheels in an additional smaller plate attached under the first plate, on which the electric motors of the driving wheels are mounted, as well as the drive battery.

The platform is equipped with four wheels for moving the robot: two differentially driven wheels with a diameter of 155 mm and two auxiliary wheels located at the front and rear ends, with a diameter of 76 mm. The wheels are attached to an axle with a diameter of 12.75 mm. The constructed platform has a differential drive, which makes it possible to control the movements of the robot by adjusting the rotational speed of both wheels. The two driving wheels, as well as the battery that powers them, and the auxiliary wheels are attached under the first plate. The drive wheels are driven by two brush motors of 96 W each, with worm gear, operating at a voltage of 12V, equipped with 36-position encoders. The motor assembly generates approximately 85 inches/Lbs of torque. A chamber and a LIDAR are mounted on the upper plate, through which the telecontrol is carried out. Additional obstacle detection sensors are placed on the bottom plate.

2.1.2 Multi-Purpose Robot Modules

2.1.2.1 Lifting platform module

Through this module, the robot can deliver and store goods in shops and warehouses. For the purpose of lifting packed supplies, a 4th plate is added to the top plate of the mobile robotic platform. The fourth plate is mounted on four linear motors through 8 L-shaped profiles - 2 for each linear motor. The linear actuators are connected to the 3rd platform of the robot using specially designed slots, which are fixed on the 3rd platform by means of bolts. The dimensions of the slot designed for the linear actuator are 80mm x 46mm x 45mm (length/width/height). An additional build was made from the slot to the 3rd plate, cocentric to the plate, for further stability of the slot. A channel of 18.4 mm width and 18 mm height was designed in which the lower part of the actuator was inserted, as well as a hole at the bottom of the 12.5 mm diameter slot used for attachment to the 3rd plate by means of a bolted connection. On both sides of the slot, holes measuring 6 mm are designed and are used to fix the linear actuator in place. In addition, an incision measuring 40mm x 12mm is made at the rear of the slot, through which the power cord of the actuators passes.

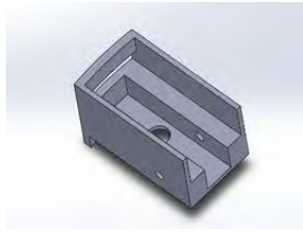


Fig.13 The designed slots for the linear actuators



Fig.14 Actuator of the lifting platform

The linear actuator used in the development is 12 volts with 90 kg maximum load capacity. The actuator uses a potentiometer for feedback. The dimensions of the actuator [81] in the retracted condition are 75 x 40 x 258. The stroke of the actuator is 100 mm. At the bottom and top are designed 6 mm holes through which it is fastened. L-shaped profiles for attaching the actuator to the 4th plate are 20mm x 20mm x 25mm with a wall thickness of 2 mm.

2.1.2.2 Rotary table module

The rotary table consists of two common parts - a main body in which the control and drive components are placed and a rotary disc with slits for storing and transporting products. The designed module is fastened to the robot by a frame which is mounted to the top plate. The frame design consists of two pairs of components for easier installation to the main body of the rotating table. The main body has dimensions 177mm x 177mm x 65mm / width x length x height /. The thickness of the sheet metal used is 2 mm. Attached to its bottom, located 6 mm from each of 4 rubber feet with a diameter of 20 mm and a height of 11 mm for additional stability.

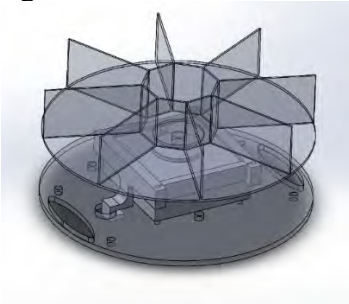


Fig.15 The rotary table and the way of mounting to the robot plate

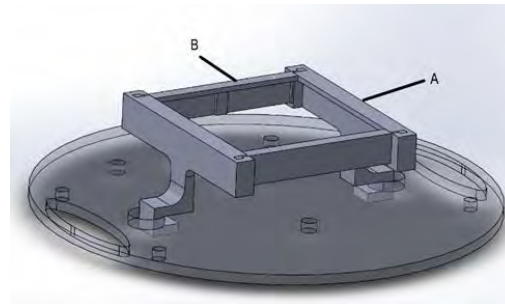


Fig.16 The frame for fixing the the rotary table- details A and B

Between the main body and the rotating disc the gear for rotational movement is located externally, with large and small gears having respective diameters of 85 mm and 13 mm. The distance between the bottom of the rotating disc and the top of the main body is 35 mm. The diameter of the rotating disc is 450 mm. In the center of the rotating disc is placed a regular octagon with a side of 50 mm. Each of the 8 slots, located 45 degrees apart, has an external arch length of 225 mm and a length measured from the center of the arch to the corresponding side of the octagon of 164.6 mm. The slots are also made of sheet metal with a sheet thickness of 1.5 mm. The height of each slot is 90 mm.

The frame for mounting the rotary table to the top plate of the robot was built using a 3D printer. It consists of two pairs of identical elements: part A and part B, which slide into each other. Part A has a U-shaped profile. The frame fixing the rotary table is attached to the top plate of the mobile robot by means of the two U-shaped profiles, the inner sides of which are tangential to the holes into which they enter. Part B enters the specially designed channels of part A, thus tightening the opposite parts and assembling the frame. Detail A is a combination of a T-shaped profile that is placed in the main body of the rotary table and a U-shaped profile for attachment to the upper plate. The T-shaped part has the following dimensions: 213mm x 20mm x 35mm. The thickness of the profile is 5 mm. The U-shaped part has a radius of 17.5 mm in the part, tangent to the hole and a total width of 28.5 mm. The frame assembly channels are 15mm x 6.5mm.

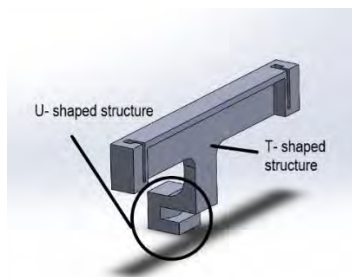


Fig.17 Detail A

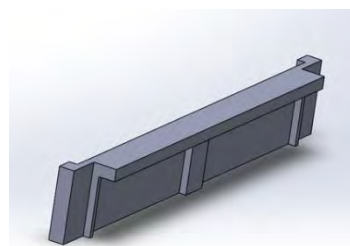


Fig.18 Detail B

Detail B has the following dimensions, described by its end points: 187 x 16,5 x 30 mm /length x width x height/. Three additional ribs are designed on the inside of the workpiece to further support the horizontal fixation of the rotary table. The thickness of the part profile is 3 mm, and for the ribbed parts with additional support it is 6 mm.

By designing such a 3D printed frame, in both cases of its operation and robot movement, the desired stability and fixing of the rotary table is achieved.

2.1.2.3 Cleaning module

The cleaning module consists of several separate systems that can operate simultaneously or independently. The first system is a combination of two identical rotating brushes [84] connected by a mechanism which allows them to be lowered or raised depending on whether the robot is only moving or in the sweep mode. A second auxiliary system is a combination of a portable tank with a tank pump which, if necessary, discharges detergent or water through nozzles [85][86] located in front of the rotating brushes. The other subsystem that contributes to the sanitation process is an adjustable rubber brush that, like rotating brushes, moves vertically using a linear actuator, and a second UV lamp is mounted to the tank, at the rear of the robot.

The mechanical system with rotating brushes (Fig. 19) is designed to provide contact with the surface, which is cleaned at any time when it is active and the robot performs the sanitation process. At the lowest position, when the brushes are active, the design provides sufficient pressure to carry out the technological procedure.

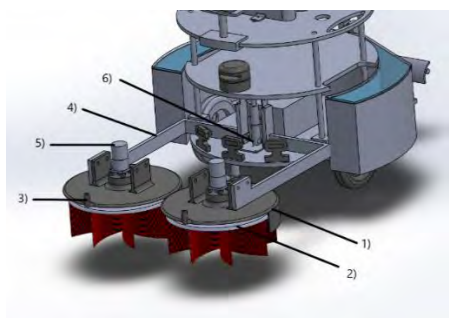


Fig.19 1- brush cover, 2- cleaning disc, 3- nozzle for detergent, 4- Structure of lifting arms, 5- electric motor, 6- linear actuator

The activation of the detergent discharge system is carried out by an operator and is determined by the type of room being sanitized. Feedback in this case is based on the telecontrol of the robot and the mounted camera. On the top plate, the robot is equipped with an antibacterial UV lamp to remove various pathogens in the environment, which is not in contact with the brushes of the developed module. An adjustable rubber brush, which, like the rotating brushes, moves vertically using a linear actuator, and a second UV lamp are mounted to the tank, at the back of the robot. The rubber brush is used to push the used detergent from the surface cleaned with rotating brushes. The additional second UV lamp is directed directly to the surface of the floor, which the robot cleans, using as the last technological operation in the cleaning process.

The rotary brush system consists of the following few components. The first component is the

cleaning disc with a diameter of 250 mm. On it is placed the brush cover with a diameter of 270 mm. From the brush cover, a semicircle with a height of 55 mm is extruded, which is facing the mobile robot to protect the robot from contamination during operation of the rotary brush. A nozzle is designed at the front end of the lid, through which, if necessary, the detergent is discharged from the tank. In the center of the bonnet are the electric motor and the plates for attaching the brushes to the lifting arms. A 12V with a speed reducer with a power of 150 rpm was chosen for the electric motor. The lifting and lowering of the brush is carried out by means of two lifting arms connected by a lever forming a rocker, fastened to a linear actuator. The actuator is fixed with a holder and placed between the sensors and the battery of the first platform.

When cleaning with brushes, their direction of rotation is determined by which side of the robot is closer to a wall of the room. Cleaning is carried out from the near to the far wall, that is, the brushes push the waste to the side, after which the robot moves in the opposite direction and the rotation of the brushes changes. This continues until it reaches the other end of the room. The principle is illustrated in Figure 20.

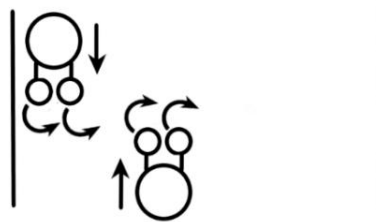


Fig.20 Principle of rotation of the cleaning brushes and the robot movement in the room subjected to a cleaning procedure

During the design of the module, the brushes are designed for use on hard surfaces. The movement of the robot in the premises of the complex and the implementation of the technological operations of the hygiene module are carried out both on the basis of internal navigation and telecontrol.

The second system of the hygiene module consists of a detergent tank and a tank pump, which are activated if necessary. The tank pump selected is 12V with a maximum flow rate of 5 l/min, which is mounted on the third plate. The detergent tank is placed on the bottom plate. Attaches to the mobile robot through 4 plates with holes that enter the rods between the first and second plate.

The rotary brush system consists of the following few components. The first component is the disc with the cleaning feathers. The disc has a diameter of 250mm and 5mm thickness of the plate, the total height is 105mm. In the center of the disc is an extruded bushing with a diameter of 9 mm and a height of 15, which is mounted to the flange transmitting the rotational movement from the electric motor by means of two bolts M3 passing transversely through the assembly (Fig. 22).

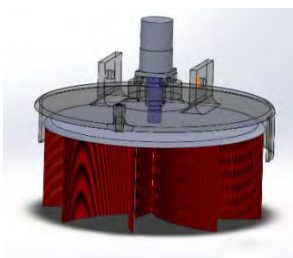


Fig.21 Rotary brush assembly

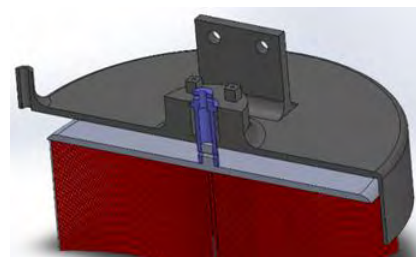


Fig.22 Rotary brush section cut

The flange connecting the brush with the electric lifter has a length of 55 mm. The small diameter is 15mm, and the large one, which rests on a bearing and covers it as a cover with a diameter of 21mm. The part that is on the bearing is 4 mm thick. For bearing, a needle bearing with an internal diameter of 15 mm, an outer diameter of 21 mm and a thickness of 22 mm is selected, which is placed in a socket designed for this purpose contained in the structure of the cover. The brush cover has a diameter of 270 mm and a thickness of 5 mm. The semicircle of the lid with a length of 55 mm, which faces the mobile robot, has been extruded in order to protect the robot from contamination when working with the rotary

brush. In the center of the lid there is a cylindrical body with an outer and inner diameter of 60 and 21 mm respectively, and a height of 26 mm. To the upper surface of the cylindrical body are constructed 4 sockets measuring 8mm x 8mm x 9mm. In the sockets are placed holes with a length of 30 mm for bolts M3, through which the electric motor is installed. A 12V with a 150 rpm reducer was chosen for electric motor [87]. The attachment of the rotary brushes to the lifting arm is carried out by plates measuring 60 mm x 65 mm x 7,5 mm, 115 mm apart, which contain holes with a diameter of 10 mm. Depending on whether the rotary brush is clamped on the left or right side, the corresponding fastening plate is used.

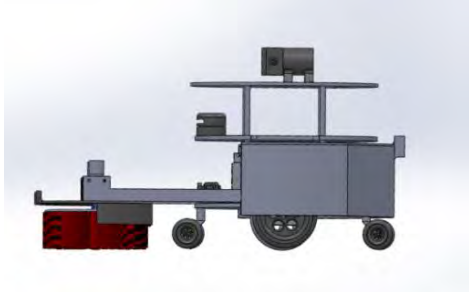


Fig.23 Side view of the robot with the rotary brush

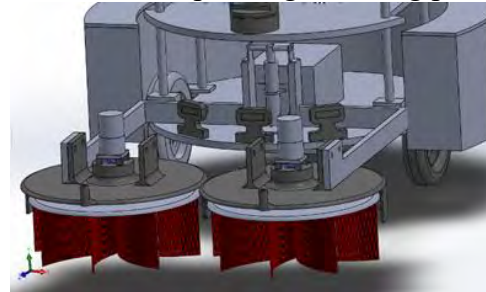


Fig.24 Assembly of a module with rotary brushes lowered, mounted to the lifting mechanism

The lifting and lowering of the brush is carried out by means of two lifting arms (Fig. 23.), connected by a lever forming a rocker, clamped to a linear actuator. The actuator[88] was fixed by means of a specially designed plate holder (Fig.25), being placed between the sensors and the battery on the first platform. The arms are made of 30mm x 10mm x 1.5mm profiles. The arms are fastened to linear plugging bearings, which are mounted to the front two rods (Fig. 24). In order for the brush to make contact with the cleaning surface, it is necessary that the piston of the linear actuator reaches the extreme lower position. In this position, the actuator is 113.5 mm from the plate from the hole in the piston. When lifting in the extreme upper position of the piston, a 30 mm stroke is set. This provides enough space for the robot to move without attaching the brush blades when the cleaning mode is not active.

An auxiliary system was added to the cleaning rotary brush by means of a pump [89] and a tank to be switched on if necessary (Fig.26a and Fig.26b). 12V was chosen as the pump with a maximum flow rate of 5l/min. The pump is mounted on the third plate by means of 4 M3 bolts. The water tank has an inner side cocentric and equal to the plates of the mobile robot, with the end points describing a circle with a diameter of 600 mm.

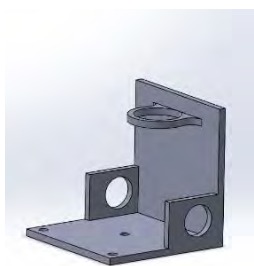


Fig.25 Holder for linear actuator

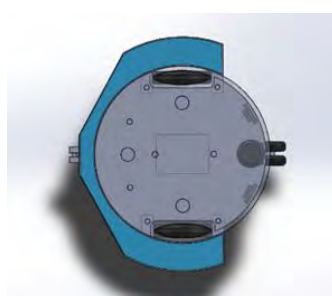


Fig.26a. Water tank, mounted to the robot platform

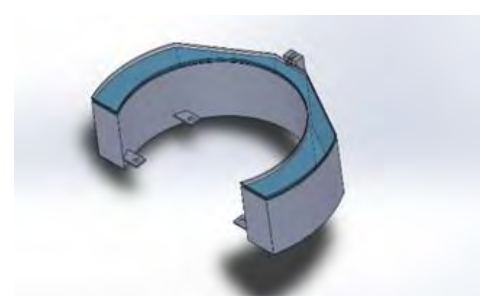


Fig.26b. Water tank

The height of the tank is 170 mm and 30 mm from the base are welded 4 plates with holes that enter the bars between the first and second plates and thus attach it to the entire mobile platform. At the rear, the tank has an element through which the lid of the tank is attached.

2.1.2.4 Scissors lift module

The hoist is made of two frame structures - frame A and B, with identical external dimensions and profiles that make them up, and the other components of the overall assembly are fixed to them. The profiles used for the outer frames are 20mm x30mm x2mm. The frames measure 400mm x 240mm and their corners are beveled in order for the frame to be as close as possible along the contour of the robot

plate. In the folded position, the loader is 60 mm high from the robot platform to which it is mounted, and in the maximum raised almost 172 mm. Frame A is attached to the upper plate of the mobile robot by means of 4 L-shaped profiles measuring 25x 35 x 3, each of the 4 profiles being clamped with 3 bolts M6.

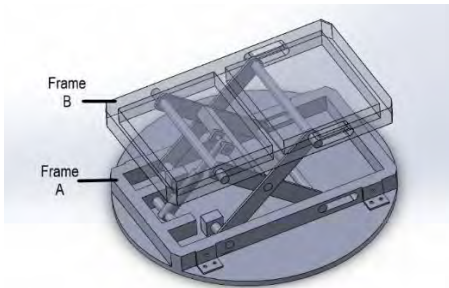


Fig.27 Assembly of the scissors lift

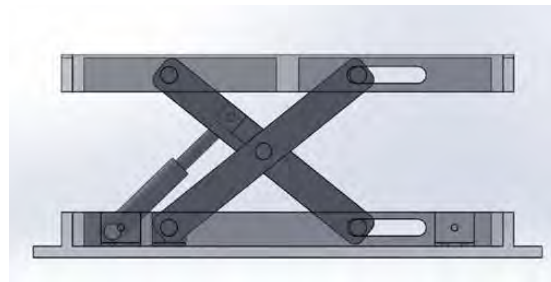


Fig.28 Front view of the mechanism with components of the scissors lift

Identical for both frames at one end of each of them in the profiles are milled grooves, which accommodate the bearing bodies and axles for which the corresponding two sliding arms are clamped for lifting frame B, and at the other end holes with a diameter of 15 mm are drilled. The length of the milled channel is 70mm, in which a needle bearing with an outer and inner diameter of 21mm and 15mm respectively is impaled. At the other end of frame A, internal to the structure, the linear actuator [90] is clamped. The actuator piston has a diameter of 12 mm and a stroke of 100 mm. The linear actuator is attached to two profiles - 20mm x 30mm with a length of 50mm, the center of grip being 13mm from the robot plate i.e. under the central axis of the profiles to ensure a larger bevel when the scissors are retracted. An additional profile is welded to frame B transversely in the middle to increase the strength and strength of the structure when lifting loads. In the upper part on frame B is welded sheet material with the same shape and thickness of the sheet 3 mm, on which to be laid the different products. The lifting arms measure 235mm x 30mm x 4mm with R8 edge rounding. In each of the arms are drilled 3 holes with a diameter of 15 mm - one central and two at each of the ends at a distance of 214.80 mm from each other. To two of the arms is welded profile 20mm x 20mm, to which the piston of the linear actuator is attached. Weight sensors [74] are placed on the upper surface in contact with the products raised by the robot to protect the mobile robot from overloading

2.1.2.5 Module with anthropomorphic robot Niryo

For the purpose of the developed service robot the anthropomorphic robot Niryo One [92] was chosen with the following specification: 6-axis robot with six degrees of freedom. The dimensions of the lengths of the components constituting the structure are as follows: base - 103 mm, arm - 80 mm, first joint - 210 mm, second joint - 180 mm.



Fig.29 Anthropomorphic robot Niryo One



Fig.30 Robot Postures

The back panel of the Niryo One contains many different physical interfaces:

- Top button;
- Switch output 12 V;

- LED;
- 4 USB ports;
- GPIO panel (Total 6 digital pins);
- Power adapter connector.
- Ethernet port on Raspberry Pi 3B;
- Dynamixel XL-320 connector;
- Dynamixel XL-430 connector;
- Power switch;
- CAN busbar connection for Niryo Steppers;

The robot weighs 3.3 kg. The possible maximum reach with a mounted gripper is up to 440 mm. At the bottom of the Niryo One are placed 4 suction cups and rubber feet, through which it is connected to a specific work surface. The maximum rotation angle for each of the joints is as follows: J1: -175° to 175°, J2: -90° to 36.7°, J3: -80° to 90°, J4: -175° to 175°, J5: -100° to 110°, J6: -147.5° to 147.5. The location of the joints is shown in Fig. 30.

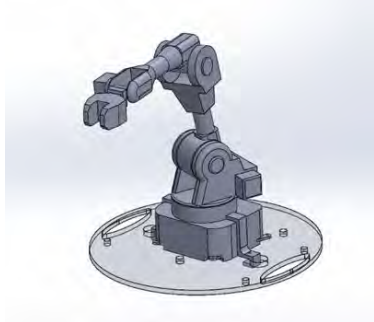


Fig.31 The fastening principle used to fix Niryo One to the top platform

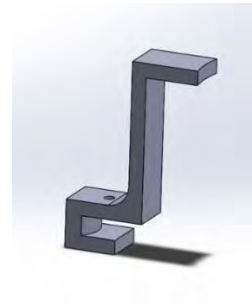


Fig.32 Niryo One fixing holder

The holder for attaching and providing the extra stability required for the Niryo One is a combination of a U-shaped profile that enters a hole in the upper plate, and an L-shaped profile, the upper part of which is corradial and in contact with the base at joint J1, thus fixing the robot in a stable position. The holder is mounted to the plate through its U-shaped part, which is clamped with M5 using the vise principle. The dimensions of the U-shaped part are 23.5 mm x 25 mm / width x length / and thickness 7 mm. The L-shaped profile has the following dimensions: 71mm x 32mm x 22.5mm / height x length x width / and 7mm thickness. The designed holder in combination with the suction cups and rubber feet of the anthropomorphic robot ensures the necessary stability during operation.

2.2 Movement and power-driven mechanisms

The mobile robotic platform chosen is of the differential control type [49]. This type of platform has two motors located on the left and right sides, which drive independently of each other the two side wheels. To achieve stability of the platform we use two auxiliary wheels at the front and rear. This wheel arrangement allows rotation in place, but on uneven surfaces contact between a wheel and the ground may be lost.

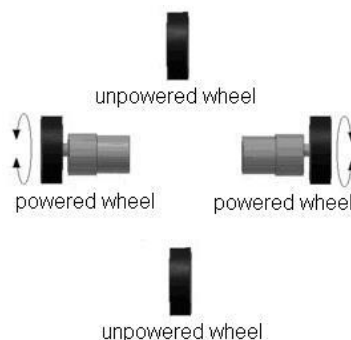


Fig.33 Movement of the mobile robo-platform

- Forward movement of both driven wheels - achieves forward movement of the robot.
- Movement of the two driven wheels back - achieves movement of the robot backwards.
- Movement of one wheel forward and the other backward - achieves rotation of the robot in a small circle, the center of which is located between the two driven wheels.
- Movement of one wheel slower than the other - achieves rotations of the robot in the direction of the

slower wheel. How fast it rotates depends on how big the difference between the two speeds is. The control of a differential mobile platform is complex because coordination and cooperation between two separately driven wheels is required [50].

2.3 Sensor system

Sensor systems are a critically important part of any mobile robot project. They help to perform all the functions of interaction with the environment – monitoring the spatial position and orientation of the robot, maintaining a trajectory, detecting obstacles, safety of the robot and the people working with it. Depending on their main function, the sensors that are used in mobile robots can be divided into several categories: tactile sensors, proximity sensors, acceleration and orientation sensors, laser scanner (LIDAR), visually based systems, wheel encoders, ultrasonic sensors.

2.4 Industrial design of the robot

2.4.1 Fundamental aspects

With a multi-purpose mobile robot, the goal should be the cleanest and neutral design that fits into the working environment of different types of equipment. The structure building the robot's appearance is segmented into separate panels. Compared to the specific segment that makes up the plastic of the design, there are specific accents that correspond to the motor units, sensors and modules of the multi-purpose robot. The graphic lines depicted around the openings of the ultrasonic sensors serve as an accent for their utilitarian placement in the design. At the rear of the structure is positioned a hole with a lid that provides access to the jack for charging the robot as well as for a view of the display showing the percentage of battery charge. In the realization of the vision and construction of the multi-purpose robot, the molds of the plastic are realized in a way that ensures maximum possible protection of the components housed inside from solid and dust particles, which will ensure its proper functioning.

CHAPTER III: CONTROL OF THE MULTI-PURPOSE SERVICE ROBOT

3.1 Tele-operation of the multi-purpose mobile robot platform through a web-based graphical interface, via MQTT and web sockets

3.1.4 Implementation of MQTT over WebSocket in JavaScript

In this development, an interface for controlling a two-wheeled mobile robot with a DC motor using MQTT.js and WebSockets is presented. [45] [70] The interface includes a text box for theme selection, a speed control slider, and four directional control buttons. The theme chosen by the user is the channel through which the interface publishes speed and motion values.



Fig.36 The overall process of communication between the elements

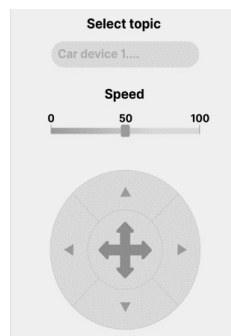


Fig.37 The developed user interface

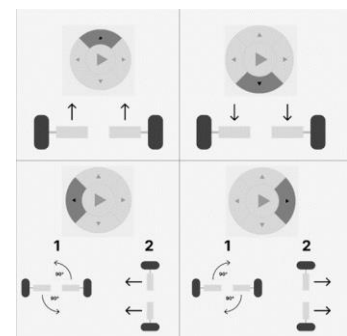


Fig.38 Movement of wheels

The movement speed is set through the JavaScript "Change" event, which takes the value from the slider field and triggers a POST query directed to the "publisher" route. A JSON object is sent that contains the subject name and a message that is an integer value determining the speed. Similarly, the direction of movement is determined by one of four buttons, each of which is indicated by a different arrow indicating the desired direction. Each button contains a command with a template for the required movement of the engine in the given direction. When one of the buttons is pressed, a POST request is

sent to the selected topic during the "MouseDown" event. After the button is released, the "MouseUp" event is triggered, which sends a command to stop the motors, and in response, the device stops moving. Figure 37 illustrates the created user interface for the program. The interface allows users to manage the speed and direction of the robot. The topic chosen by the user is used to publish the speed and movement values. The robot subscribes to the a topic and receives the values sent by the interface. As a message protocol MQTT is used, and WebSockets provides a two-way communication channel between the interface and the robot. Figure 38 illustrates how the engine wheels move depending on the button pressed. Pressing the "up" button causes both wheels to rotate forward, resulting in the robot moving forward. Pressing the "down" button causes both wheels to rotate in the opposite direction, resulting in the robot moving backwards. Pressing the "left" button causes the right wheel to rotate forward, while the left wheel rotates backwards, as a result of which the robot turns left. Conversely, pressing the "right" button causes the left wheel to rotate forward while the right wheel rotates backwards, resulting in the robot turning right [44].

The flow chart (Fig.39) illustrates the main steps related to the control of a mobile robot through the provided interface, including setting a theme, speed regulation, driving direction control, posting values and acting on these values to control the robot wheels. As shown in Figure 40, after starting the robot the controller automatically establishes a connection with the MQTT broker. Once connected, when a subscribed topic message is received from the broker, the specified Callback function is performed.

As a result of the input data received, a command is sent to execute the movement of the engine as needed.

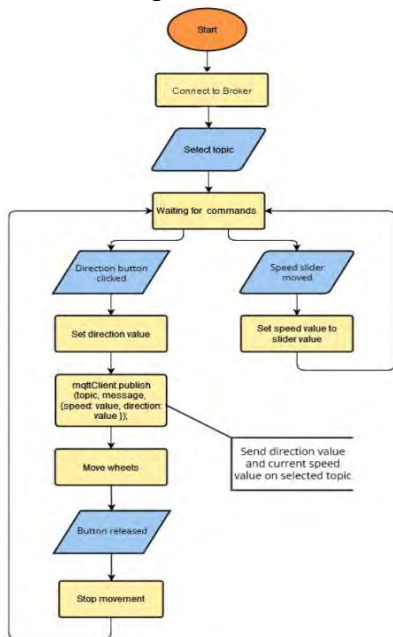


Fig.39 Workflow of the process

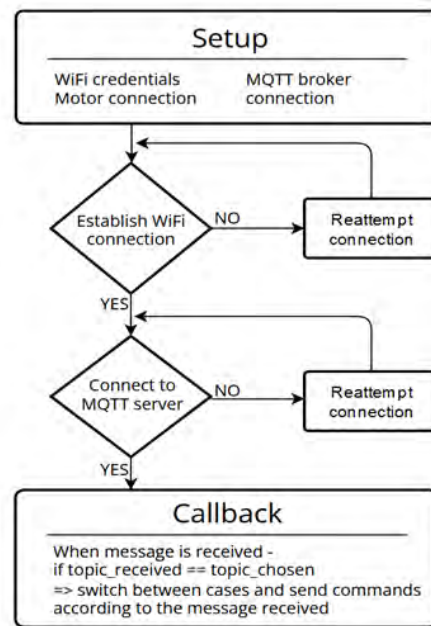


Fig.40 Firmware diagram

Actions such as contacting a broker, subscribing to a topic, and posting messages are saved and displayed in a console. Figure 41 shows the logs for subscribing to a particular topic, pressing the "Forward" button and running it after a few seconds. The time of the action is also displayed.

```

    Console:
    14:34:24: Connected with MQTT Broker: "ws://192.168.41.129:9001/mqtt"
    14:34:41: Subscribed to MQTT Topic: "home/car1"
    14:35:07: Published "FORWARD, 50" to MQTT Topic: "home/car1"
    14:35:07: MQTT Message Received. Message: "FORWARD, 50" MQTT Topic:
    "home/car1" QoS Value: "0"
    14:35:21: Published "STOP, 0" to MQTT Topic: "home/car1"
    14:35:21: MQTT Message Received. Message: "STOP, 0" MQTT Topic:
    "home/car1" QoS Value: "0"
  
```

Fig. 41. Console Register

3.2 Control of the multi-purpose robot in drug delivery mode

The rotary table is a module that can be mounted to the mobile robot-platform with tele-control (Fig. 42) and is an important component for the distribution of items such as medicines for different patients relative to the given hospital room.

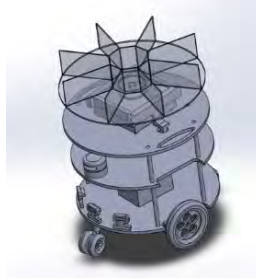


Fig.42 View of the rotation module attached to the mobile robot-platform

3.2.1 Algorithm of robot operation

The robot operating algorithm in drug delivery mode is presented in Figure 43 and includes the following steps:

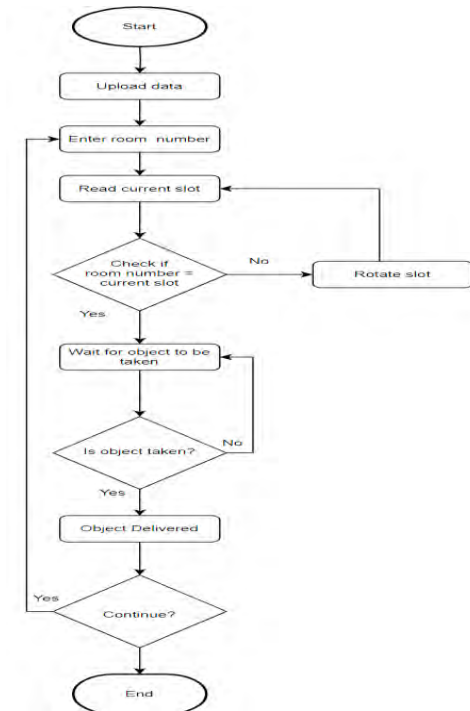


Fig.43 Flowchart of the operation of the module

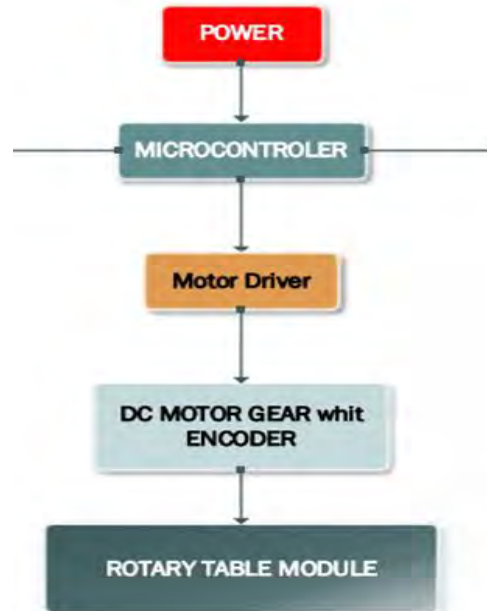


Fig.44 Scheme of the main components of the distribution module

- Installation and startup of the module:

After the successful installation of the rotary table to the mobile robot-platform and the power supply, the module control program is started.

- Loading items:

The operator places the drugs or other items to be carried and distributed in the individual slots of the rotary table.

- Select Mode:

The program interface provides a choice of different modes of operation, depending on the objectives of the operation. For example, the operator can choose an automatic mode for the distribution of objects based on preset parameters or a manual mode, where the operator has control over the process.

- Start the work cycle:

After the operator selects the desired mode, he starts the duty cycle by pressing the corresponding button in the program.

- Rotation and distribution:

The way the rotary table works is to rotate until the selected objects in the given slot reach in front of the robot. On the basis of an encoder it is known about the position of the given slot. Through the interface, it is saved in which slot objects for which room and which patient are intended.

- Submission of the items:

The robot moves towards the patient or the place where the objects are to be provided. The rotary table feeds the objects. After the patient takes their pills using the tele-controlled robot's camera, the operator validates that the pills have been taken..

- Repetition of the process:

If there are more slots for the patient after handing over the item, the rotary table continues to rotate and feed the next item until all items assigned to the given room and patient are allocated.

- Completion of the work cycle:

After all subjects have been successfully allocated, the program outputs a message to complete the operation. The operator sets the robot to move to the next location on the basis of semi-autonomous tele-control or directly navigates it via direct tele-control

- Turn off the module:

The operator can disconnect the module after the successful completion of the duty cycle.

3.2.2. Simulations and results of verification of the module and algorithm of operation

The simulation setup aims to verify the management and behavior of the module. The simulation was carried out on the basis of several softwares, through their paralytic and synchronized implementation. In the simulation, the following softwares are bound:

- SolidWorks-CAD: Software used to construct the robot platform and module
- Proteus 8.16: Software for electronic design and automation (EDA/ ECAD). Proteus allows creating circuits, designing printed circuit boards, etc. It is used in this simulation by providing powerful functionality when simulating microcontrollers by applying hex files to already built circuits.
- Arduino IDE: Environment used to write the code illustrating the algorithm
- HHD Software's Virtual Serial Port Tools: It is a comprehensive set of software tools designed to manage and emulate virtual serial ports. Virtual serial port tools can create virtual serial devices and connections that fully emulate the functionality of hardware physical devices.
- Blender 3D: 3D modeling software that allows writing scripts using the Python API. Provides the ability to communicate with Proteus.

The first step is to create the code in the Arduino IDE. In order to be able to realize the simulation in the code, the BlendixSerial.h library is introduced, which allows the communication between the simulation of the hardware in Proteus with the 3D environment of Blender 3D, in which the constructed robot is placed.

```

#include <BlendixSerial.h>
#include <Servo.h>

BlendixSerial blendix;
Servo myservo; // create servo object to control a servo
int timeOn = 50;
int timeOff = 50;
String servoPrompt = "Enter sequence of servo degrees (1-8): ";
bool autonomousMode = false;

void setup() {
  Serial.begin(9600);
  myservo.attach(10);
}

void loop() {
  Serial.println("Enter 1 for Autonomous mode or 0 for Manual mode: ");
  while (Serial.available() == 0) {}
}

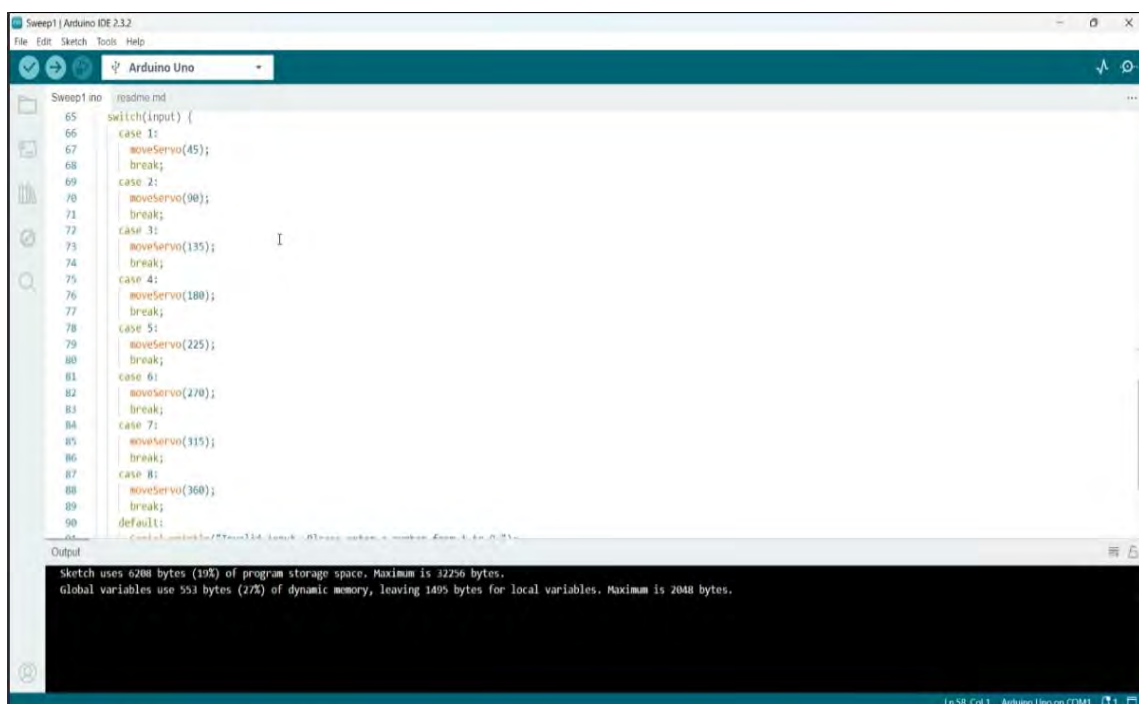
int modeInput = Serial.parseInt();

if (modeInput == 1) {
  autonomousMode = true;
  autonomousLoop();
} else if (modeInput == 0) {

```

Fig.45 The BlendixSerial.h library implemented in the IDE created in Arduino

For manual mode, 0 is entered in the Proteus virtual terminal, and for autonomous distribution mode, 1 is entered. Each of the slots that are loaded by the operator in drug distribution corresponds to a specific angle based on the number of slots that are on the rotation disk. Figure 46 illustrates the value of the angle corresponding to the specific slot.



```

Sweep1.ino  readme.md
65  switch(input) {
66  case 1:
67    moveServo(45);
68    break;
69  case 2:
70    moveServo(90);
71    break;
72  case 3:
73    moveServo(135);
74    break;
75  case 4:
76    moveServo(180);
77    break;
78  case 5:
79    moveServo(225);
80    break;
81  case 6:
82    moveServo(270);
83    break;
84  case 7:
85    moveServo(315);
86    break;
87  case 8:
88    moveServo(360);
89    break;
90  default:
91    // do nothing
92  }
93  }

Output
Sketch uses 6288 bytes (19% of program storage space. Maximum is 32256 bytes.
Global variables use 553 bytes (27% of dynamic memory, leaving 1495 bytes for local variables. Maximum is 2048 bytes.
Ln 58, Col 1  Arduino Uno on COM1

```

Fig.46 Slot angle values

A second step after setting up the program in the Arduino IDE environment is the construction of the hardware system and its wiring in the Proteus environment. For this purpose, Proteus libraries are used, in which there are ready-made components - motors, sensors, controllers, etc. Figure 47 shows the created hardware scheme connecting the components that make up the module as well as a virtual terminal for entering and visualizing the data.

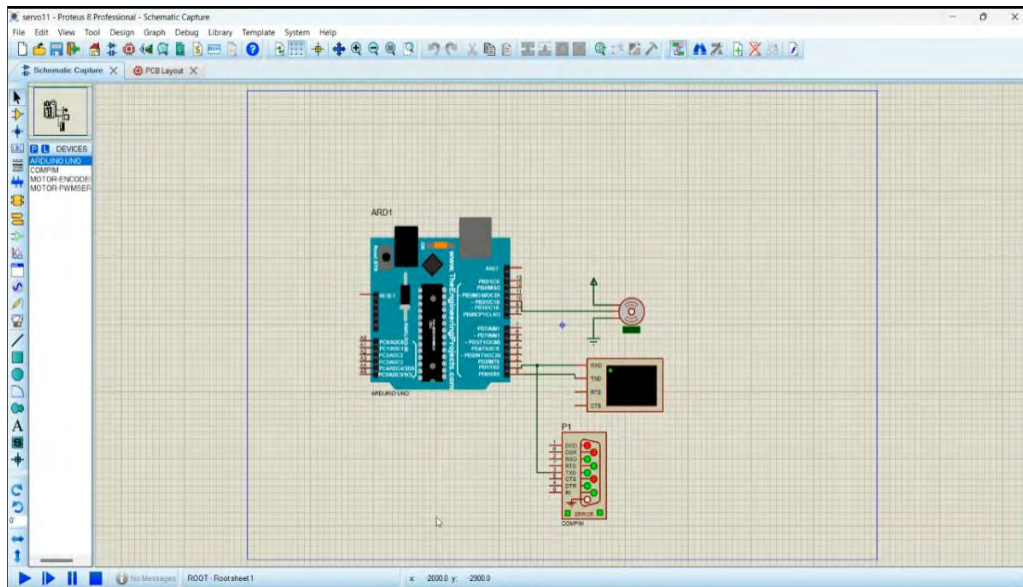


Fig.47 Hardware scheme of the drug distribution module

Once all the components are connected properly and the hardware system building the module is ready, the next step is to load the program written in the Arduino IDE in the Proteus environment, where we perform the simulation. Loading the program is done by pressing the right click on the microcontroller in the environment of Proteus. As a result, the Edit Component menu comes out, through which we load the arduino file.

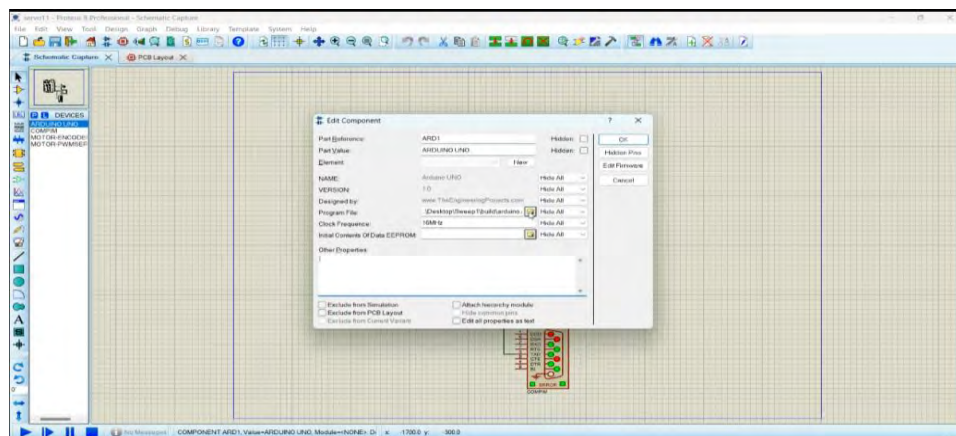


Fig. 48: Uploading the program written in the Arduino IDE in the Proteus environment

The next step is to open the 3D model of the robot in the 3D environment of Blender. In order for the model to move in the way we want, the constructed parts in Solidworks must be exported separately and assembled into a new one in the Blender 3D environment where the necessary constraints to the 3D objects are set and the objects that will be managed by the program and will move in the specified way are specified. The rigging steps for the model to move along a certain trajectory based on the control program that is implemented in the Proteus simulation are the following:

In object mode, an "armature" is created (a skeletal structure that manages a 3D object).

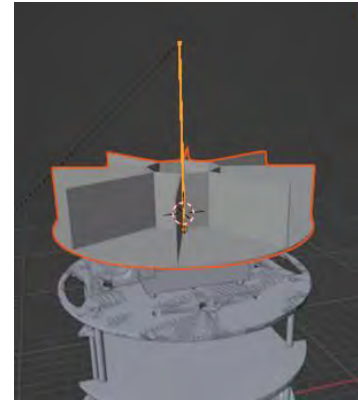
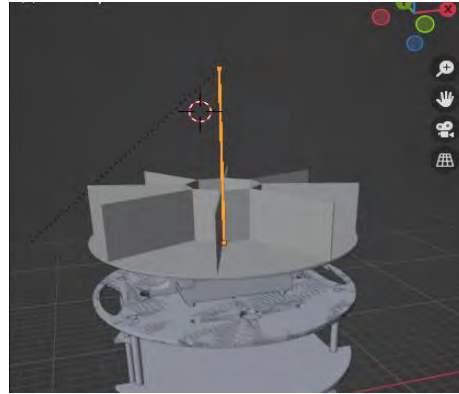
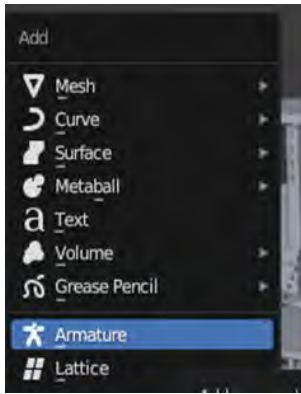


Fig. 49 Menu for generating an armature structure

Fig.50 The generated armature for controlling the 3D object

Fig.51 Positioning of the armature in the 3D object

Once the fittings are created, it is moved to the place where we want the movement to take place and placed towards the overall construction of the model we wish to be rigged (Figure 50 and Figure 51). In object mode, once the desired model is selected, the armature is switched to the pose mode (Figure 52). A choice of bone to be erupted, i.e. attached to the model (Figure 53), is then selected. Finally, ctrl+p is pressed and a "bone" is selected(Figure 54)

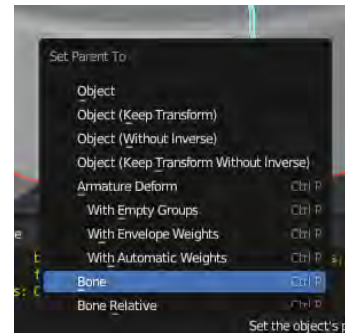
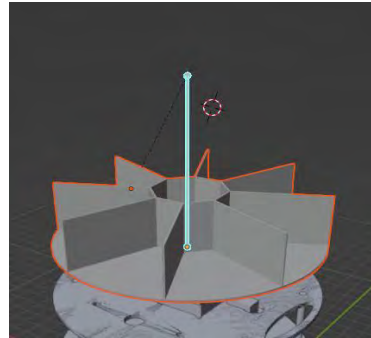
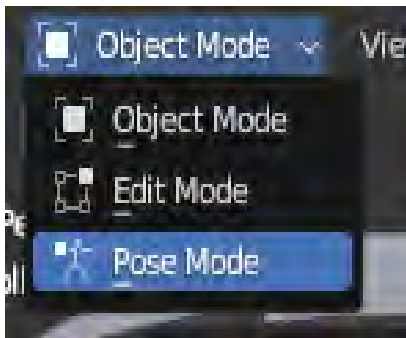


Fig.52 Choosing Pose menu

Fig.53 Choosing a bone which is to be rigged

Fig.54 Menu for choosing a "bone"

After the model is rigged, we add the application to Blender BlendixSerial. The BlendixSerial library is designed to work together with the blendixserial add-on in Blender. It allows controlling objects in Blender through serial communication with Arduino or other microcontrollers. By emulating the COM ports from HHD Software's Virtual Serial Port Tools, together with BlendixSerial begins the connection between the simulation in Proteus and the 3D constructed robot with attached module in Blender

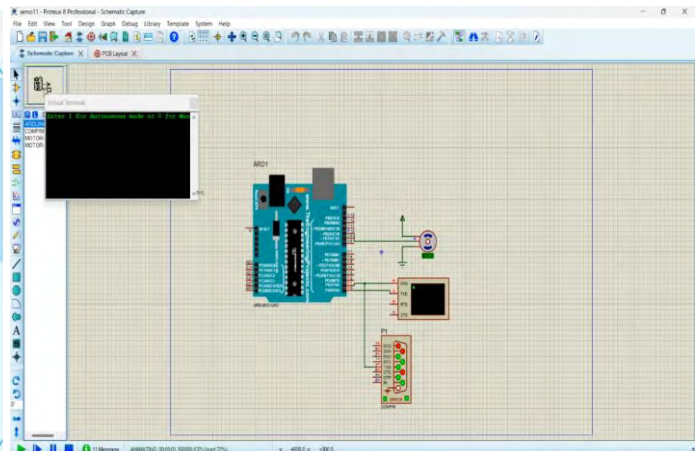
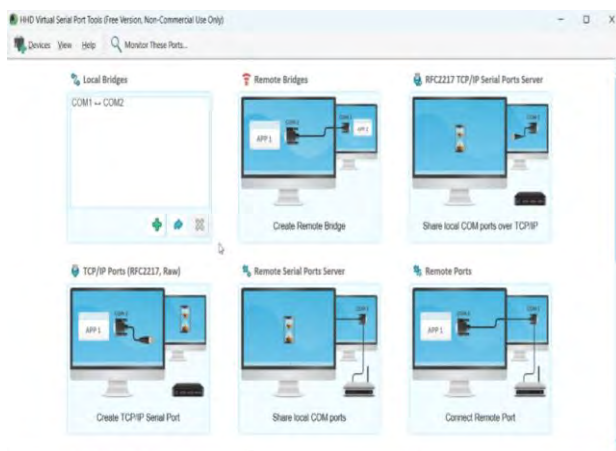


Fig.55 The emulation of the COM ports

Fig.56 Start the simulation

The simulation in Proteus is started. (Figure 56). Using the virtual terminal, the running program is visualized, which is written in Arduino and implemented in Proteus through the generated hex file. The terminal waits for 0 or 1 to be entered for manual or autonomous distribution mode. In order to visualize the changes in the 3D model of the constructed robot in the BlendixSerial drop-down menu, we must have pressed Connect Serial and Start Movement, as well as indicate the emulated port, which is illustrated in Figures 57, 58 and 59.

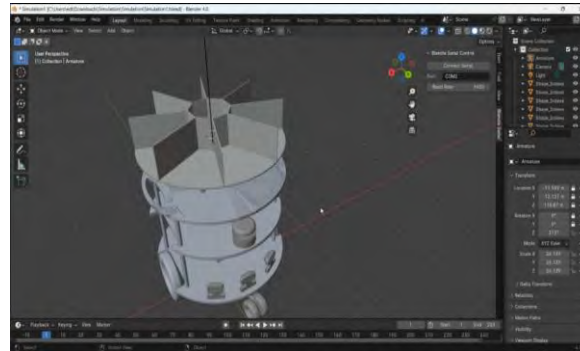


Fig.57 Selection of COM port in Blender



Fig.58 BlendixSerial before startup

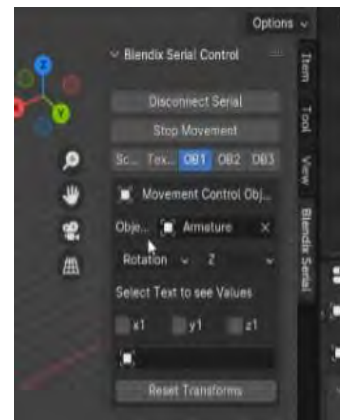


Fig.59 BlendixSerial after startup

After entering 1 and starting the autonomous allocation the program waits for the introduction of the sequence of slots with drugs loaded into them, which are for the patient to which the operator sends the robot. (illustrated in Figure 60).

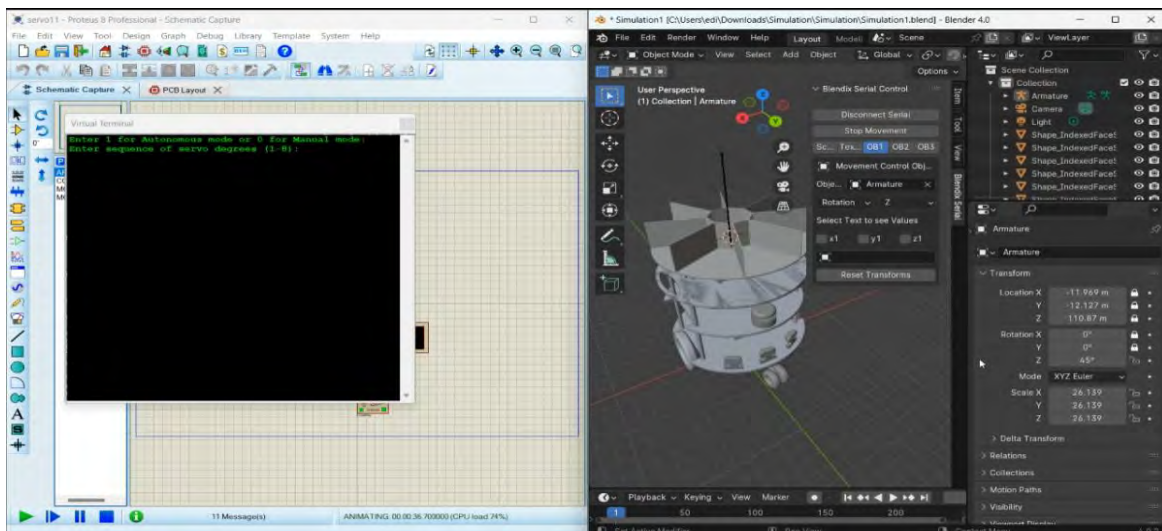


Fig.60 The synchronized environments of Proteus and Blender

Figure 61 shows the autonomous distribution mode for 3 slots. Once we have entered the numbers of the slots in which we have put the drugs, the program executes them consistently. The virtual terminal shows the sequential execution of the 3rd, 6th and 1st slots, and after each spin to the respective slot terminal it indicates that the movement is successful / completed. 45 degrees corresponds to the first slot and can be seen both on the terminal on the left side of the screenshots and on the right side in the 3D environment of the blender in the z unlocked rotation of the object.

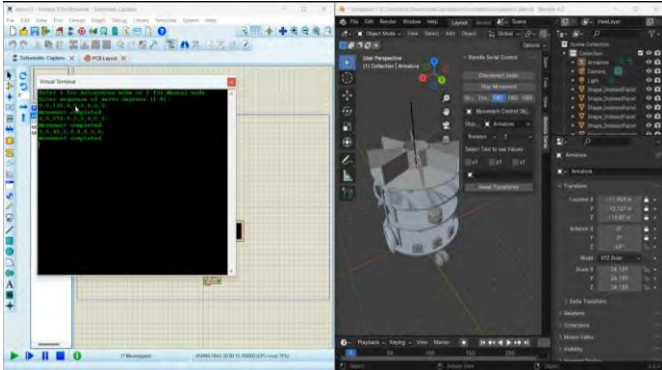


Fig.61 Autonomous distribution mode

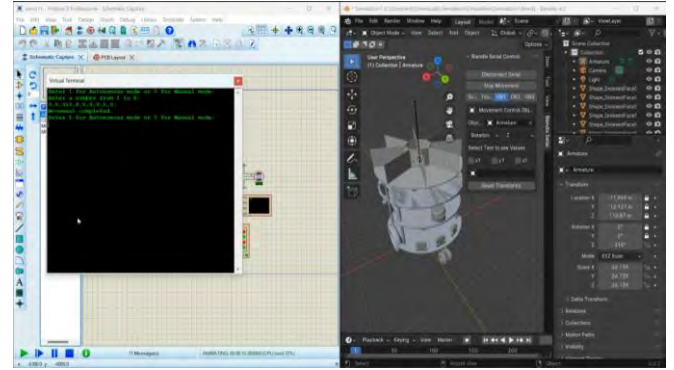


Figure 62: Manual distribution mode

In manual mode (which is activated when entering 0 in the terminal) the slot is entered, which we want to rotate in the front of the robot so that the patient can access it. In the image Fig. a slot 7 has been introduced, which corresponds to 315 degrees, which are visualized in the virtual terminal and in the menu responsible for rotation of the distributor table in the 3D blender environment. Once the movement is successfully completed the terminal awaits the next input for mode and then for slot number.

3.3 Control of the multi-purpose robot in surveillance, security and signalling mode

The surveillance and security module is an important component of the tele-controlled mobile robot platform, which provides the operator with the ability to choose between surveillance mode, video or photo recording mode, and active security mode..



Fig. 63 Mobile robot-platform with mounted module for surveillance and security

The developed algorithm of robot operation in bypass, surveillance, security and signaling mode is presented in Figure 64, and Figure 65 presents the main components that make up the module.

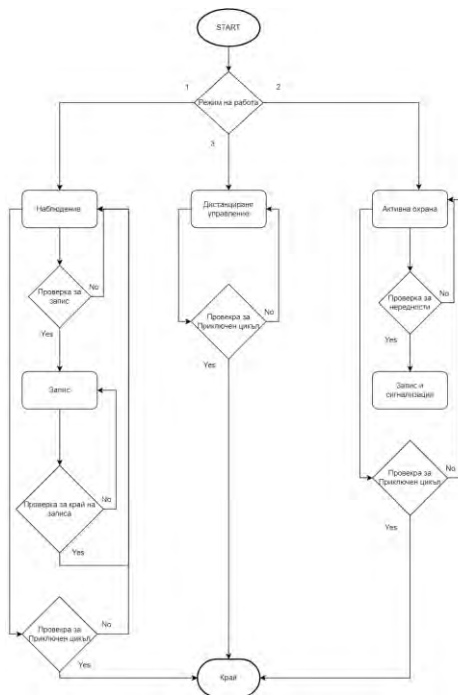


Fig.64 Flowchart of the operation of the module

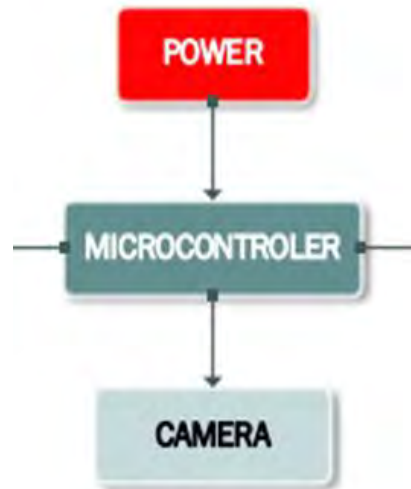


Fig.65 Scheme of the main components of surveillance module

Algorithm of robot operation:

- Installation and startup of the module:

The surveillance and security module contains a built-in camera that is mounted on the top plate of the mobile robot platform. After the successful installation of the module and the power supply, the module control program is started.

- Select mode:

The interface of the program provides a choice of different modes of operation. The operator can choose between surveillance mode, video or photo recording mode, and active guard mode.

- Start the work cycle:

After the operator selects the desired mode, he starts the duty cycle by pressing the corresponding button in the program.

- Surveillance and recording:

The camera of the surveillance and security module begins to broadcast a video signal that can be monitored in real time by the operator. In recording mode, the program can start recording video or photos that are saved for future use.

- Remote control:

The operator can control the tilt and rotation of the camera through the program interface. This allows the operator to point the camera at specific objects or areas.

- Active security:

In active guard mode, the robot can use additional sensors or algorithms to detect movement or illegal activities in the surveillance area. In case of detection of illegal activities, the module can provide a warning to the operator and start recording the incident.

- Completion of the work cycle:

After the operator has finished monitoring or recording, a message is displayed to complete the operation.

- Turn off the module:

The operator can turn off the module after the successful completion of the operating cycle. This module provides capabilities for monitoring and security at different locations and situations, providing real time to the operator or recording important data for future use.

3.4 Control of the multi-purpose robot in mode for transporting and lifting loads by a means of scissor lift

The general appearance of the robot in the mode for transport and lifting by means of a notch hoist is shown in Figure 66.

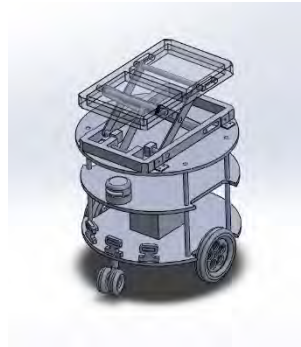


Fig.66 General appearance of the robot with a plate hoist module mounted to it

3.4.1 Algorithm of robot operation

The developed algorithm of operation is presented in Figure 67 and includes the following steps:

- It starts with the installation and startup of the module. After the operator brings the platform into operation and turns on the power, the program is started and the controller sends a signal to the driver of the linear motor (actuator).
- If the lifting platform is not in its starting position, the linear engine begins to lower the arms on which the lifting platform is attached. After performing this procedure, the platform is in "anticipation" mode, until a command is given by the operator. This procedure must also be carried out in case the platform has stopped its work in an emergency and the lifting platform has remained in the working position.
- When the robot receives a command "lift the load", the controller sends a signal to the linear motor driver. The motor begins to raise the arms on which the lifting platform is mounted until the weight sensor touches the load. It takes into account the weight of the load, and if it exceeds the norm, the program reports an error "Attention, excess weight!" Then the mechanism returns the lifting platform to "standby" mode.
- When the program receives a command "lift the load" and the weight sensor detects a weight in the norm, the mechanism raises the load to the blinded height and the robot can proceed to transport the load.

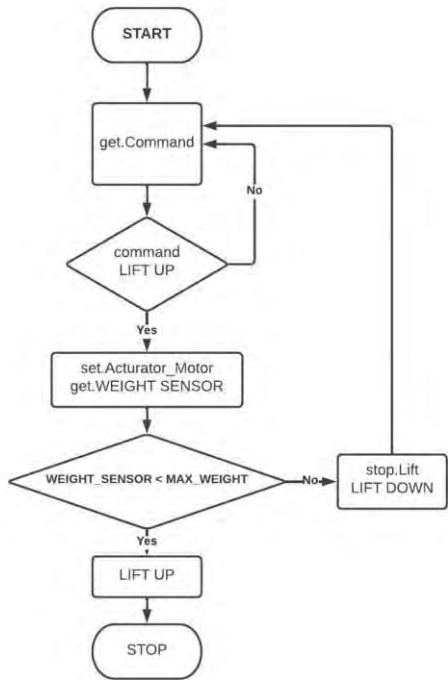


Fig.67 Flowchart of the operation of the module

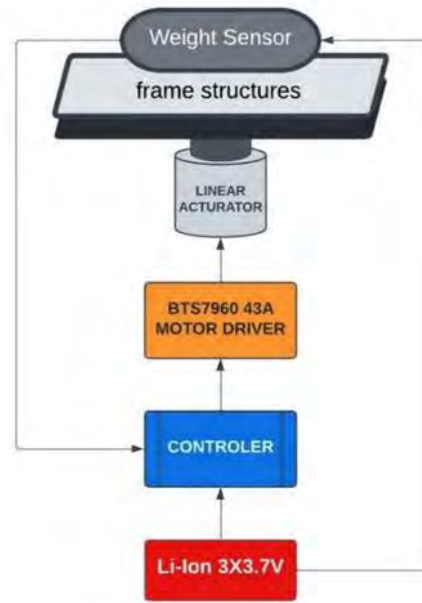


Fig.68 Scheme of the main components of lifting module

Figure 68 presents the hardware system of the lifting robot. The lifting module is powered by a Li-ion battery block consisting of 3 series connected cells with a voltage of 3.7V and a maximum current of 5A.

3.4.2. Simulation and results of the verification of the module and algorithm of operation

Through the simulation setup, the control and behavior of the module is verified. The simulation is performed on the basis of several softwares through their parallel and synchronized implementation. The following software are bound in the simulation: SolidWorks-CAD, Proteus 8.16, Arduino IDE, HHD Software's Virtual Serial Port Tools, Blender 3D.

The first step is to create the code in the Arduino IDE. In order to be able to realize the simulation in the code, the BlendixSerial.h library is introduced, which allows the communication between the simulation of the hardware in Proteus with the 3D environment of Blender 3D, in which the constructed robot is placed.

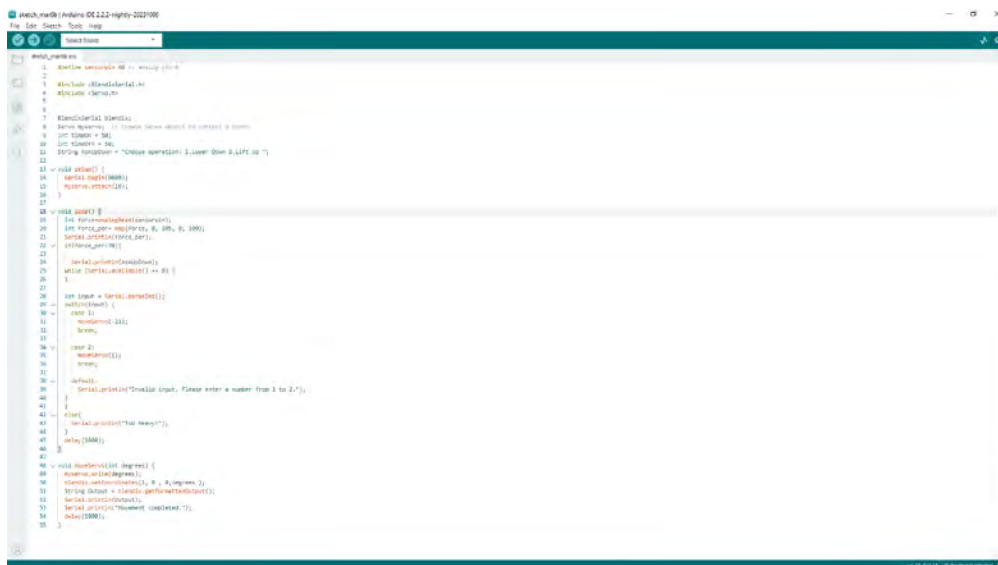


Fig.69 Creation of the code in the Arduino IDE

The code provides 2 user options, one for lifting and one for lowering.

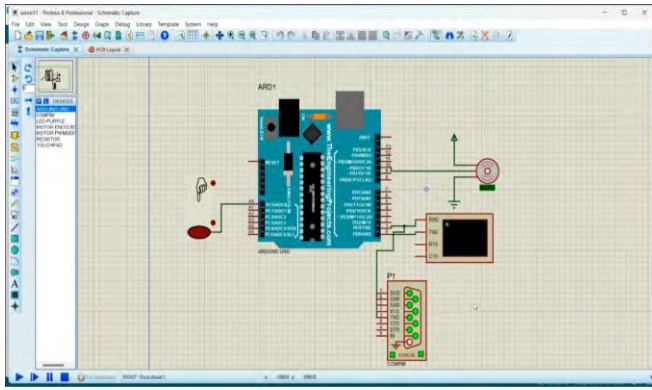


Fig.70 Hardware scheme in Proteus

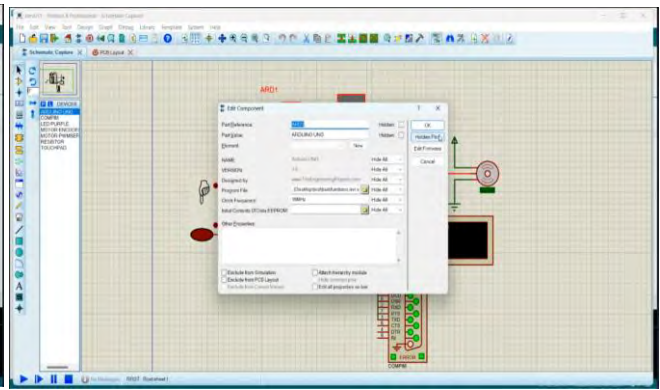


Fig.71 Uploading the program written, in the Arduino IDE in Proteus environment

A second step after setting up the program in the Arduino IDE environment is the construction of the hardware system and its wiring in the Proteus environment. For this purpose, Proteus libraries are used, in which there are ready-made components - motors, sensors, controllers, etc. Figure 70 shows the created hardware scheme connecting the components that make up the module as well as a virtual terminal for entering and visualizing the data. Once all components are connected correctly and the hardware system building the module is ready the next step is to load the program written in the Arduino IDE in the Proteus environment, where we perform the simulation. (figure71). Loading the program is done by pressing the right click on the microcontroller in the environment of Proteus. As a result, the Edit Component menu comes out, through which we load the Arduino file. The next step is to open the 3D model of the robot in the 3D environment of Blender. In order for the model to move in the way we want, the constructed parts in Solidworks must be exported separately and assembled into a new one in the Blender 3D environment where the necessary constraints to the 3D objects are set and the objects that will be managed by the program and will move in the specified way are specified. The rigging steps for the model to move along a certain trajectory based on the control program that is implemented in the Proteus simulation are the following:

In object mode, an "armature" is created (a skeletal structure that manages a 3D object). Once the armature is created, it is moved to the place where we want the movement to take place and placed to the overall construction of the model we want to be rigged. (Figure 72, Figure73 and Figure74).



Fig. 72 Menu for generating an armature structure



Fig.73 The generated armature for controlling the 3D object



Fig.74 Positioning of the armature in the 3D object

In object mode, after selecting the desired model, the armature is switched to pose mode (figure 75). This is followed by a choice of bone to be erupted, i.e. attached to the model (Figure 76). Finally, ctrl+p is pressed and "bone" is selected(Figure 77)

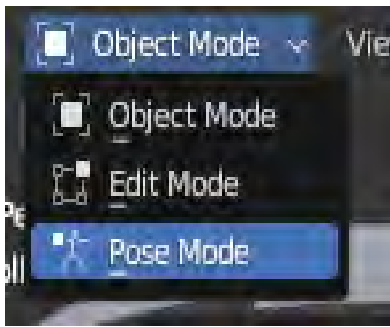


Fig.75 Choosing Pose menu



Fig.76 Choosing a bone which is to be rigged

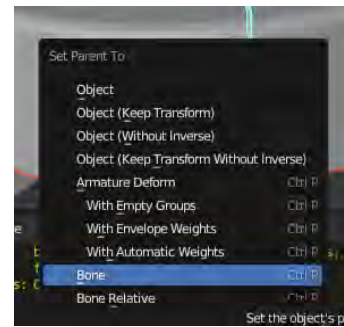


Fig.77 Menu for choosing a "bone"

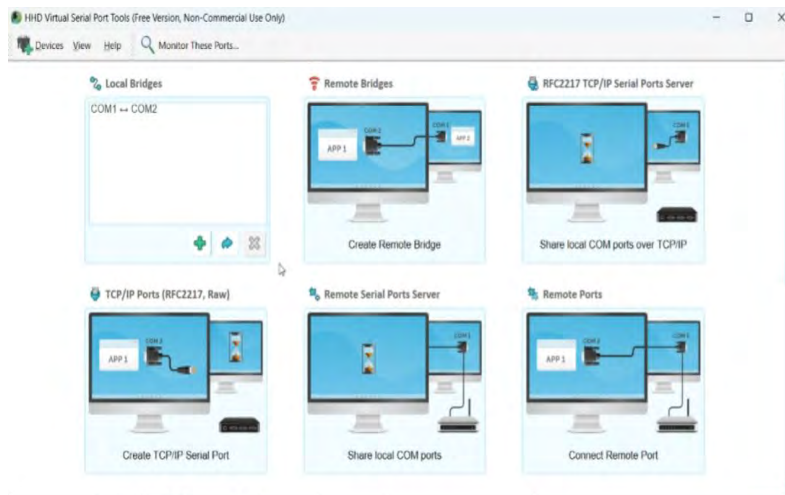


Fig.78 The emulation of the COM ports

After the model is rigged, we add the application to Blender BlendixSerial. The BlendixSerial library is designed to work alongside the blendixserial add-on in Blender. It allows controlling objects in Blender through serial communication with Arduino or other microcontrollers. By emulating the COM ports from HHD Software's Virtual Serial Port Tools, together with BlendixSerial begins the connection between the simulation in Proteus and the 3D-engineered robot with attached module in Blender.

We run the simulation in Proteus (figure79). Using the virtual terminal, the running program is visualized, which is written in Arduino and implemented in Proteus through the generated hex file. The terminal is waiting for us to enter 1 or 2 to remove or lift the platform.

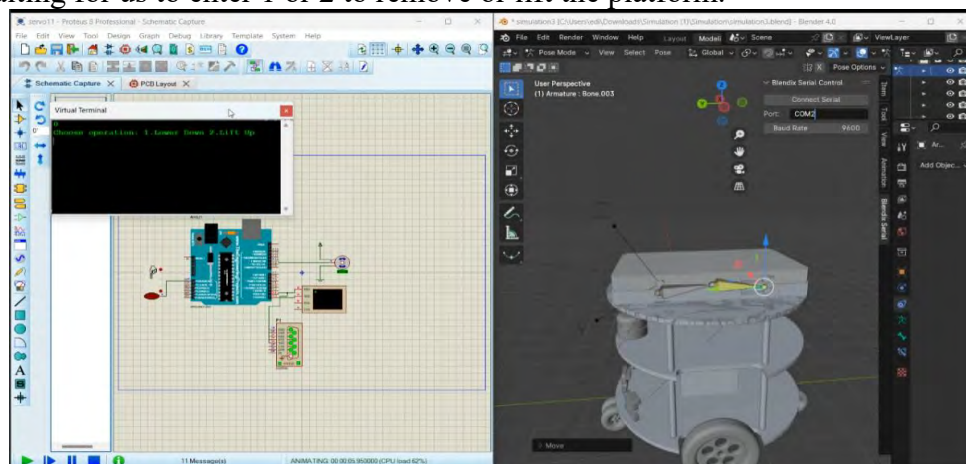


Fig.79 Running the simulation in Proteus and selecting a COM port in Blender

In order to visualize the changes in the 3D model of the constructed robot in the drop-down menu of BlendixSerial we must have pressed Connect Serial and Start Movement, as well as specify the emulated port.

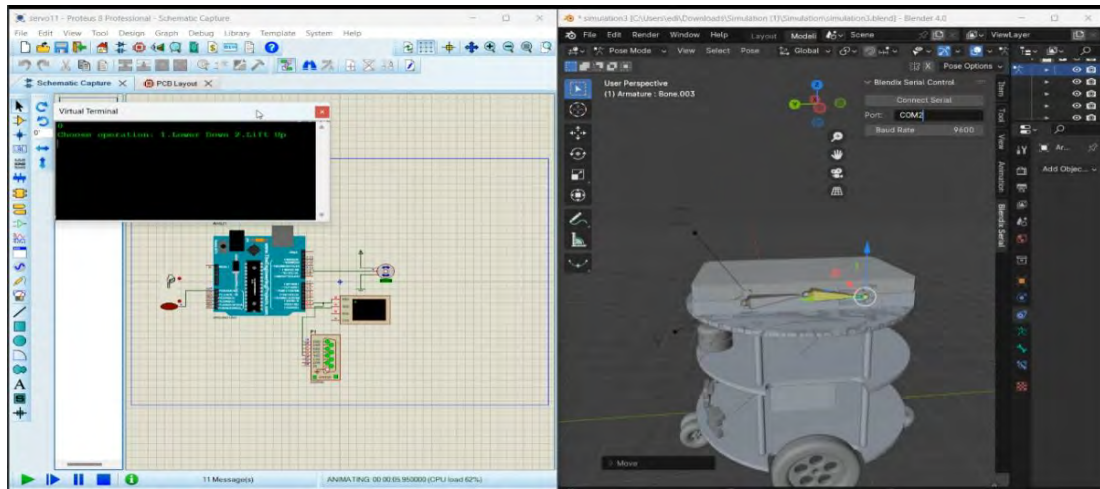


Fig.82 Starting position waiting for the introduction into the terminal

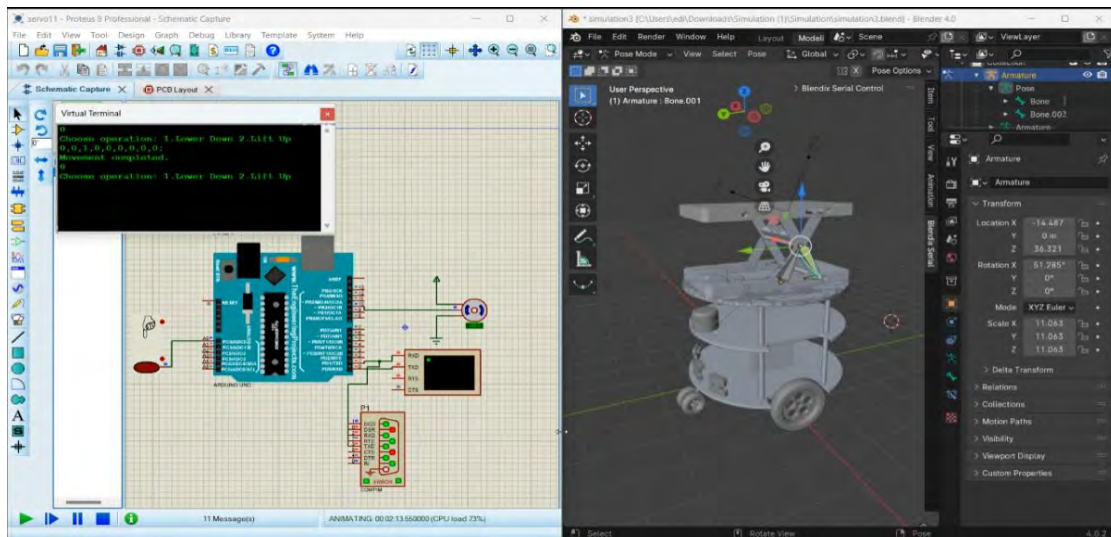


Fig.83 The robot with the lifted module

Figures 82 and Figure 83 depict the two possible robot positions, namely downthrown and raised platform. When the permissible weight is exceeded, the user receives a warning that the load placed on the platform is too much and the program temporarily stops its function until the excess load is removed – Figure 84.

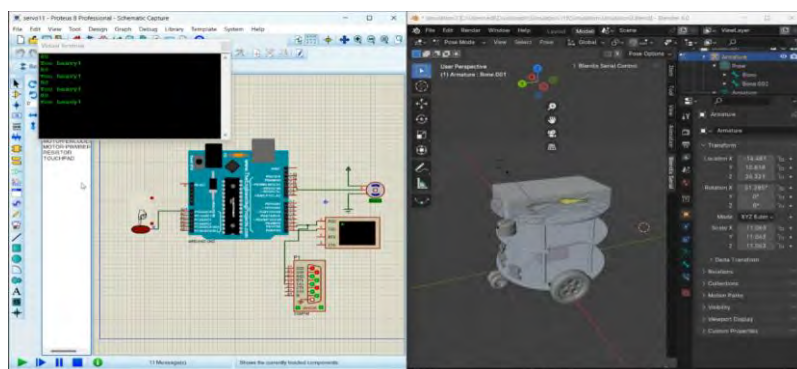


Fig.84 Terminated robot function until excess load is removed

Once the movement is successfully completed the terminal awaits the next command input by the user.

3.5 Control of the multi-purpose robot in storage and delivery mode via a lifting platform

The general appearance of the robot in the mode for storage and/or delivery of finished products is shown in Figure 86.

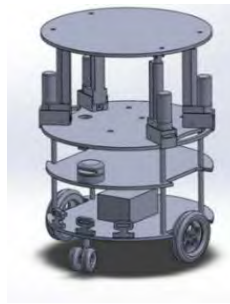


Fig.86 Robot with mounted module with lifting platform for the supply of finished products

3.5.1 Algorithm of robot operation

The developed algorithm of robot operation in production delivery mode by means of a lifting platform module is presented in Figure 87. Similar to the plate hoist, the module with lifting platform for storage and delivery goes through the same functional steps.

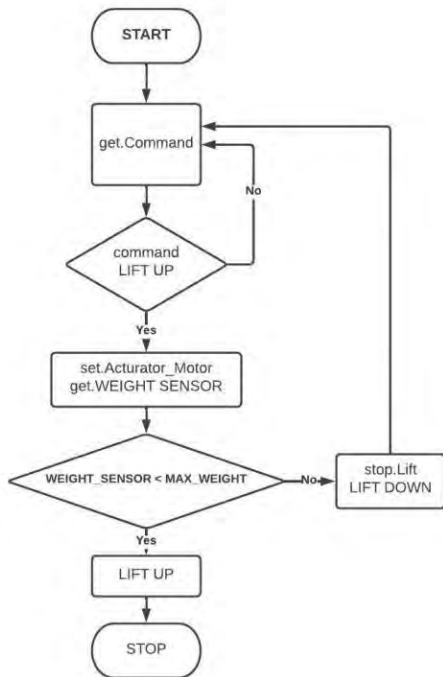


Fig.87Flowchart for controlling the module

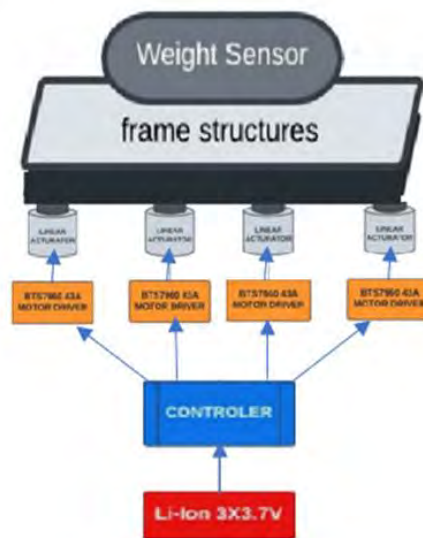


Fig.88 Diagram with the basic components building the module

Figure 88 presents the hardware system of the module. The service robot in storage and/or delivery mode includes a differential mobile platform, an industrial computer and controllers, an electromechanical drive system (DC motors with reducers and encoders), a 12V LiPo rechargeable battery and a sensor system. The linear actuator used was 12 volts with a maximum load capacity of 90 kg.

3.5.2Simulation and results of the verification of the module and algorithm of operation

The verification of the module and the algorithm of work was realized through the implementation of a computer simulation, for which the synchronized application of the following software was performed in parallel: SolidWorks-CAD, Proteus 8.16, Arduino IDE, HHD Software's Virtual Serial Port Tools and Blender 3D.

The first step is to create the code in the Arduino IDE. In order to be able to implement the simulation in

the code, the BlendixSerial.h library is introduced, which allows the communication between the simulation of the hardware in Proteus with the 3D environment of the Blender 3D in which the constructed robot is placed.

When starting the simulation part in Proteus in the virtual terminal of the simulation, the operator chooses the type of operation / movement of the platform /: 1- for downloading 2- for lifting.

In the program, when the weight limit that can be taken by the platform, the terminal returns the message "Too heavy!" and does not allow the operation to be performed until the weight reduction in the permissible values.



```

1 #include <Arduino.h>
2 #include <BlendixSerial.h>
3 #include <Servo.h>
4
5 BlendixSerial blendix;
6 Servo myservo, myservo2, myservo3;
7 int channel = 0;
8 int timeout = 50;
9
10 string message = "Close operation: 1,lower down 2,lift up ";
11
12
13 void setup() {
14   Serial.begin(9600);
15   myservo.attach(9);
16   myservo2.attach(10);
17   myservo3.attach(11);
18 }
19
20
21 void loop() {
22   int force_param = map(force, 0, 100, 0, 180);
23   Serial.println(force_param);
24   if(force_param > 90){
25

```

Fig.89 The written program in Arduino IDE

A second step after the creation of the program in the environment of the Arduino IDE is the construction of the hardware system and its wiring in the Proteus environment. For this purpose, the Proteus libraries are used, in which there are ready-made components - motors, sensors, controllers, etc. - Figure 90 shows the created hardware scheme connecting the components building the module as well as a virtual terminal for entering and visualizing the data.

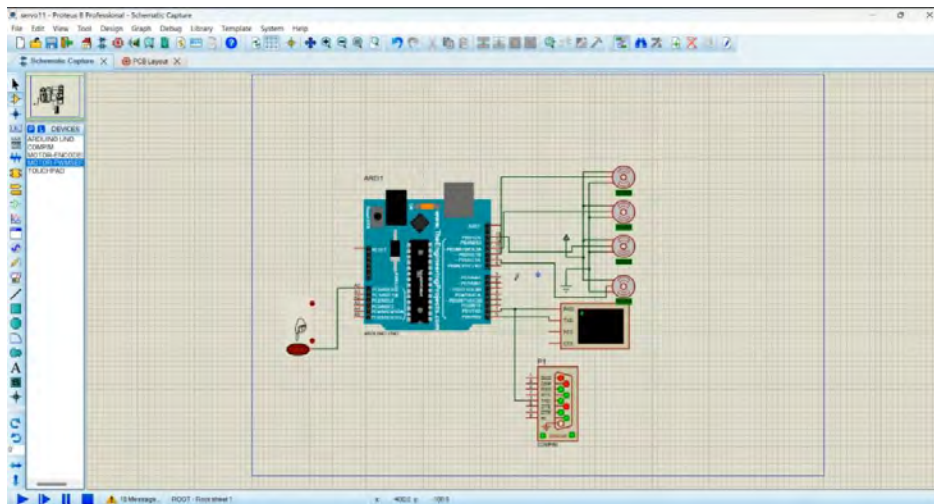


Fig.90 Hardware scheme

Once all the components are connected properly and the hardware system building the module is ready, the next step is to load the program written in the Arduino IDE in the Proteus environment, where we perform the simulation. Loading the program is done by pressing the right click on the microcontroller in the environment of Proteus. As a result, the Edit Component menu comes out, through which we load the Arduino file.

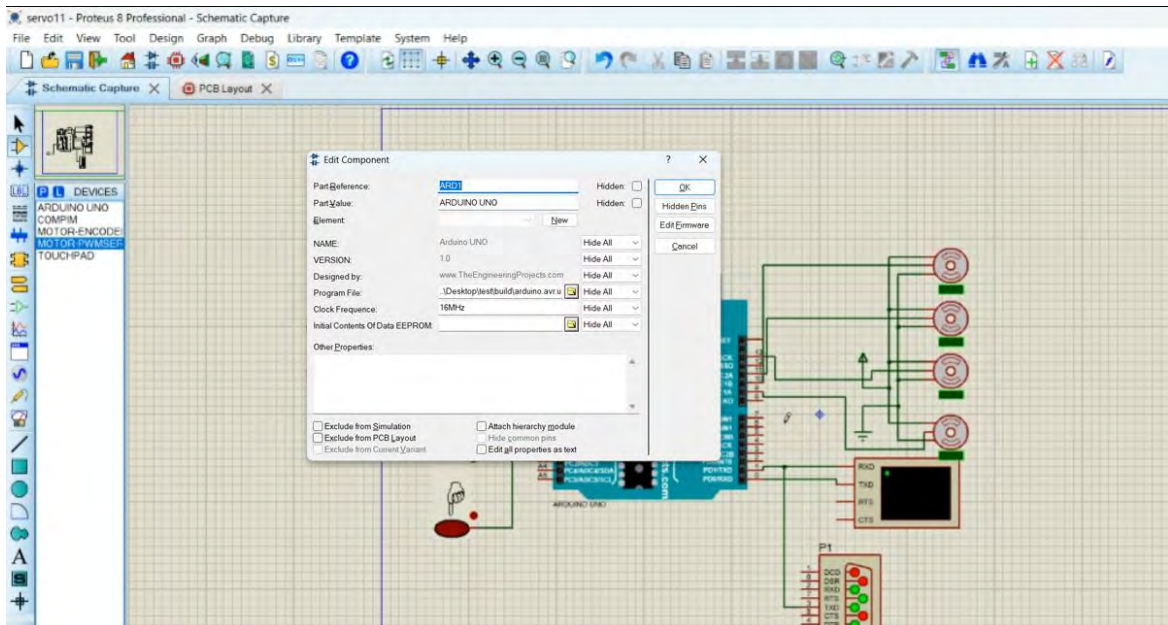


Fig.91 Uploading the Arduino program in Proteus environment

The next step is to open the 3D model of the robot in the 3D environment of Blender. In order for the model to move in the way we want, the constructed parts in Solidworks must be exported separately and assembled in a new way in the Blender 3D environment where the necessary constraints to the 3D objects are set and the objects that will be managed by the program and will move in the specified way are set. The rigging steps for the model to move along a certain trajectory based on the control program that is implemented in the Proteus simulation are the following:

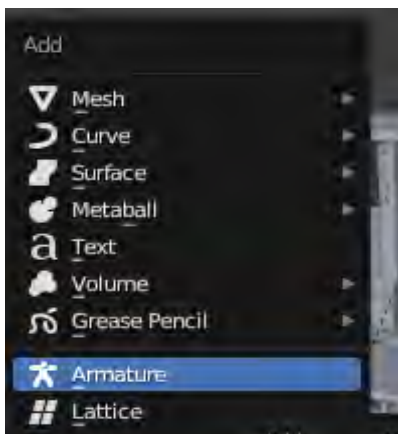


Fig.92 Menu for generating a bone structure



Fig.93 The generated armature for controlling the 3D object



Fig.94 Positioning of the armature in the 3D object

In object mode, "armature" are created. (Fig. 92). Once the fittings are created, it is moved to the place where we want the movement to take place and placed to the overall construction of the model we want to be rigged. (Fig. 93 and Fig. 94).

In object mode, after selecting the desired model, the armature is switched to pose mode (figure 95). This is followed by a choice of bone to be rigged, i.e. attached to the model (Figure 96). Finally, ctrl+p is pressed and "bone" is selected(Figure 97)

After the model is rigged, we add the application to Blender BlendixSerial. The BlendixSerial library is designed to work alongside the blendixserial add-on in Blender. It allows controlling objects in Blender through serial communication with Arduino or other microcontrollers.

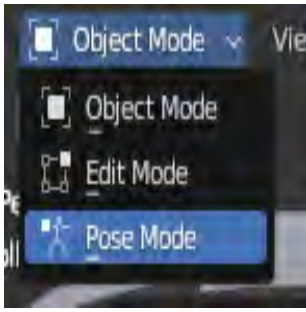


Fig.95 Choosing Pose menu



Fig.96 Choosing a bone which is to be rigged

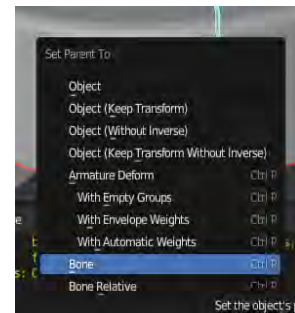


Fig.97 Menu for choosing a "bone"

By emulating the COM ports from HHD Software's Virtual Serial Port Tools, together with BlendixSerial begins the connection between the simulation in Proteus and the 3D-engineered robot with attached module in Blender

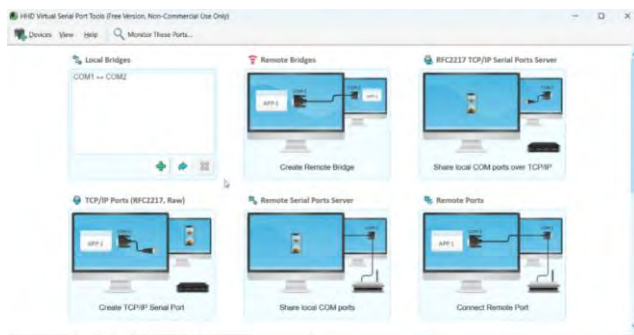


Fig.98 Emulating the COM ports

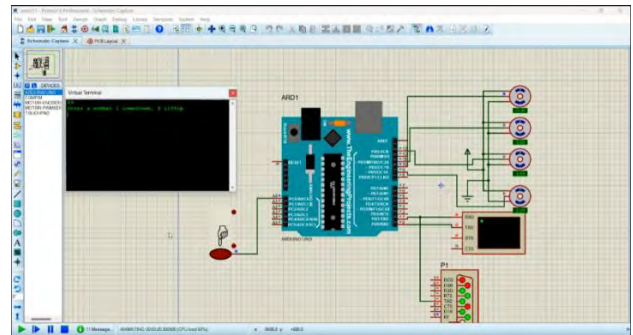


Fig.99 Start the simulation from Proteus

We run the simulation in Proteus. Using the virtual terminal, the running program is visualized, which is written in Arduino and implemented in Proteus through the generated hex file. The terminal waits for us to enter 1 or 2 to remove or erect a lifting platform.

In order to visualize the changes in the 3D model of the constructed robot in the drop-down menu of BlendixSerial we must have pressed Connect Serial and Start Movement, as well as specified the emulated port. By introducing "2" in a virtual terminal, the position of the lifting platform rises up the Z axis (in the case where the reported weight is less than 70% of the prepostima). Once the movement is successfully completed, the terminal awaits the next entry. Figure 103 and Figure 104 show by visualizing in the virtual terminal a situation where the weight is above the allowable and the operation is not allowed and the next situation in which the weight is in the set limit values and the movement is carried out.

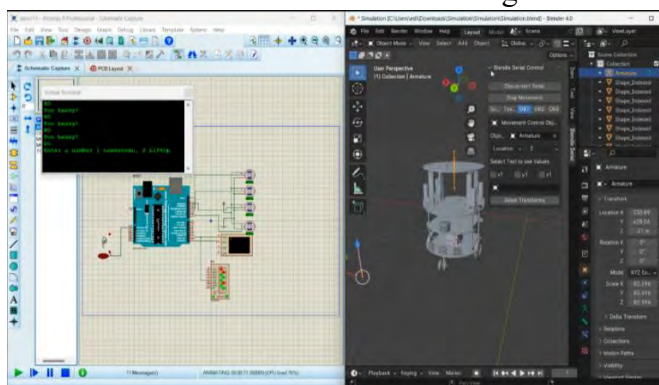


Fig.103 A situation in which the load is above allowed values and the operation is not permitted

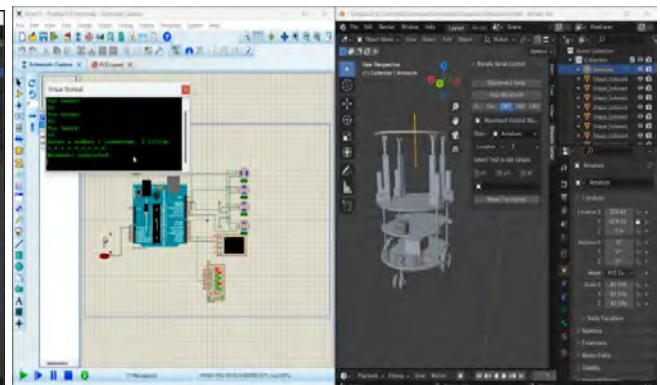


Fig.104 The movement is carried out- the load is in the allowed values

By introducing a "1" in the virtual terminal the position of the lifting platform returns to its original position. Once the movement has been successfully completed the terminal awaits the next input (Fig. 105)

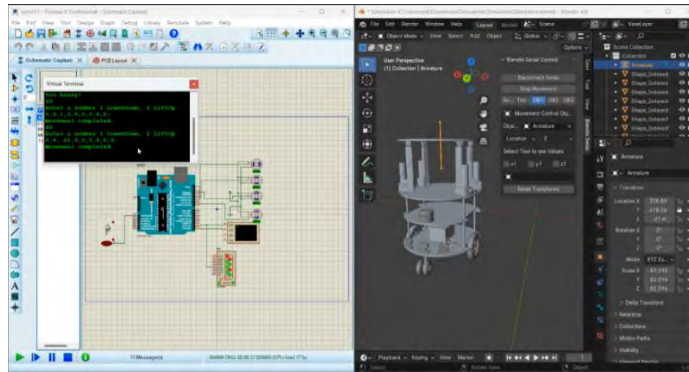


Fig.105 The terminal awaiting the next introduction

3.6 Control of the multi-purpose robot in cleaning mode

The general appearance of the robot with a cleaning module connected to the universal mobile platform is shown in Figure 107.

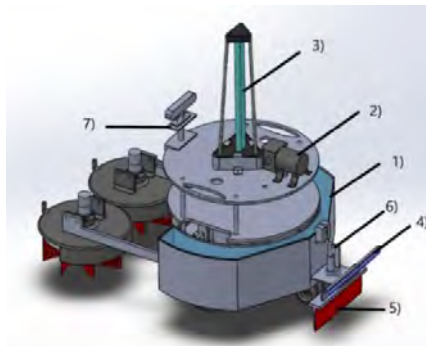


Fig. 107: 1- detergent tank with cover, 2- tank pump, 3- antibacterial UV lamp for sanitation of the surroundings, 4- antibacterial UV lamp for floor sanitation, 5- rear adjustable brush, 6- linear actuator for moving rear brush, 7- teleoperation camera

3.6.1 Algorithm of robot operation

The developed algorithm of operation in cleaning mode, presented in Figure 108, includes the following steps:

- It starts with the installation and startup of the module. After the operator has installed the cleaning module and put the platform into operation (power on), the program is started and the controller sends a signal to the actuator. The motor begins to lift the structure on which the rotating brushes are attached until the angle set by the program is reached. After performing this procedure, the platform is in "standby" mode until a command is issued by the operator. This procedure must also be carried out in case the platform has interrupted its operation in an emergency situation and the structure carrying the rotating brushes has remained in the working position.
- When the program receives the command "start cleaning", the controller sends a signal to the driver [93] of the motor. The actuator begins to lower the arm on which the rotary motors are mounted until the brushes touch the working surface. After the structure is lowered, the controller sends a signal to the drivers of the motors driving the rotating brushes. They are driven, reaching their maximum speed in 5 seconds. This prevents the load that would result from the sudden start of the engines.
- When the program receives the command "start cleaning with water", if the platform was in "standby" mode, the procedure in step 2 is repeated, and after the motors driving the rotating brushes reach their maximum speed, the controller sends a signal to the pump motor driver of the tank.

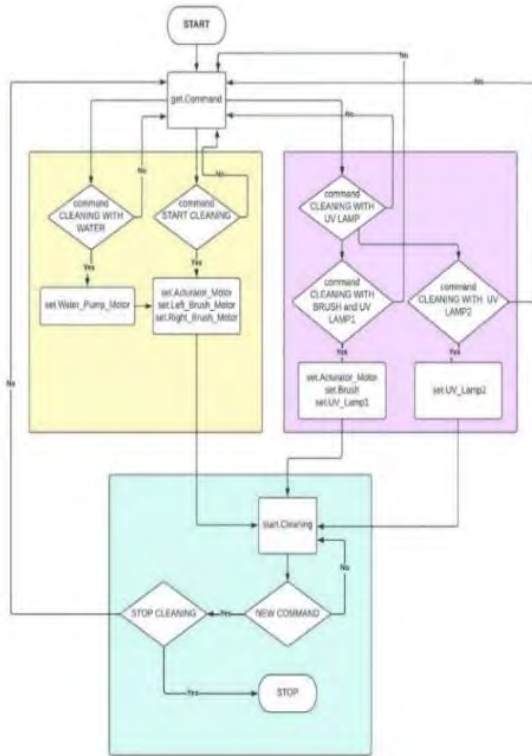


Fig.108 Flowchart of the operation of the module

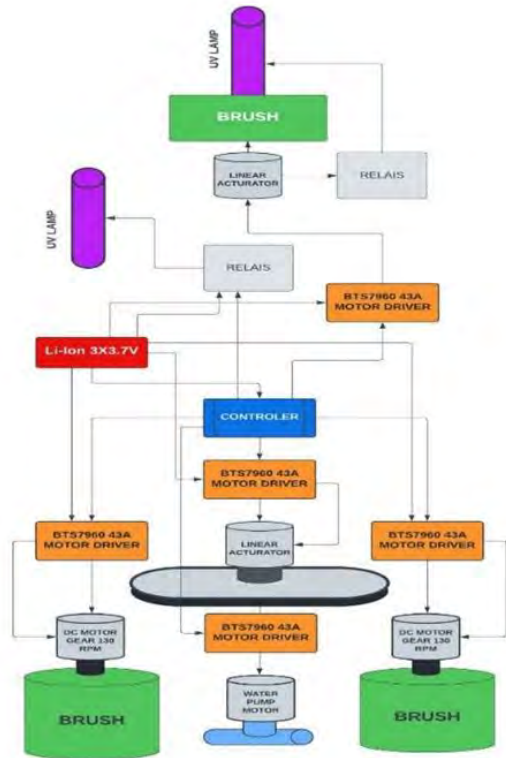


Fig.109 Scheme of the main components of cleaning module

- If the platform was in "start cleaning" mode when the command "start water cleaning" was given, the controller directly sends a signal only to the motor driver controlling the water pump.
- When the program receives the command "start cleaning the floor with a UV lamp", the controller sends a signal to the linear motor driver. The motor starts lowering the arm, to which a floor cleaning brush and a UV lamp are mounted. After the arm is lowered to the floor, a signal is sent to the relay supplying the UV lamp.
- When issuing the command "Start cleaning the environment with a UV lamp", the controller sends a signal to the relay power supply controlling the UV lamp cleaning the environment.
- At the "end of cleaning" command the controller delivers successively the following signals:
 - a signal to the engine driver driving the pump of the stop-up tank,
 - signal to the motors drivers driving the rotating brushes to stop operation,
 - signal to the linear motor driver to lift the structure to the specified position set by the program,
 - signal to the linear motor driver to lift the rear brush and command to turn off the UV lamp for floor hygiene and the central UV lamp cleaning the surroundings in the room.

After this command, the platform will again be in standby mode and the power can be interrupted.

In Figure 109 the hardware system of the cleaning module is represented. The cleaning module is powered by a lithium-ion battery consisting of 3 series-connected cells with a voltage of 3.7V and a maximum current of 5A.

3.6.2 Tele-operation of the developed cleaning module

The control of the cleaning module is added to the web-based graphical interface for teleoperation of the robot. The connection between the robot and the user is via Wi-Fi. The user can use a mobile device or laptop- the only condition is to connect to the robot's network and have a web browser. Teleoperation is obtained using the technology of the web socket. They transmit data from the robot to the operator and vice versa.

Buttons and tools have been added to the interface to manage the various modules and the operations they perform. Based on the teleoperation, the operator sets the creation of a map and the preservation of the map of the current room, and according to the situation he observes in the room, sets which systems to turn on when the robot cleans it. In addition, at any time the operator has access to the robot and the

remote control option.

The cleaning functions built into the graphical interface are the following:

- Create a map: the operator manually walks the robot around the room until a complete map is created. Each procedure then uses the map made [16]. A sequential map scanning algorithm is used to ensure that cleaning procedures are carried out everywhere
- Start cleaning only with brushes
- Start cleaning with brushes and spraying with detergent
- Switching on and activating the disinfecting subsystem with UV lamps. Since there is a danger to humans and animals during this procedure, an algorithm of radiation protection is activated before starting the lamps. In this mode, the robot detects the presence of humans or animals. If detected, the UV lamps shall be switched off and signalled at the operator's graphical interface. For better perception, the robot is able to recognize markers and other objects [17].
- Cleaning including all systems / brushes, washing, UV lamps / of the sanitary module: In this mode, all systems of the cleaning module are activated, as well as all algorithms.

The feedback from the robot to the operator is realized by visualization of data from the sensors and devices of the robot. The operator can monitor the video stream from the robot's camera, track the location of the robot according to the navigation map, see if there is a set operation and to what extent it is completed. When a problem occurs while performing an operation or a malfunction occurs, it is also reported to the operator.

In addition to automated operations, buttons have been added to the graphical interface to control the individual functions of the modules. In this way, the operator can use the robot in manual teleoperation mode, to diagnose systems, etc.

3.6.3. Simulation and results of verification of the module and algorithm of operation

To verify the module and the algorithm of work, a computer simulation is compiled and implemented through the synchronized application of the following softwares: SolidWorks-CAD, Proteus 8.16, Arduino IDE, HHD Software's Virtual Serial Port Tools and Blender 3D.

The first step is to create the code in the Arduino IDE. In order to be able to implement the simulation in the code, the `BlendixSerial.h` library is introduced, which allows the communication between the simulation of the hardware in Proteus with the 3D environment of the Blender 3D in which the constructed robot is placed.

When starting the simulation part in Proteus in the virtual terminal of the simulation, the operator selects which modes of operation of the cleaning module to activate. The program has 4 modes, which are entered with a digit in the terminal:

- 1-start of cleaning (starts cleaning with brushes after the shoulder structure is removed);
- 2-UV cleaning (cleaning with the brush positioned at the back of the robot together with the UV lamp attached to it);
- 3-cleaning with water (starts the electric pump for cleaning with water through the nozzles positioned at the front of the rotary brushes);
- 4- UV lamp mode (cleaning the surrounding environment using the UV lamp mounted on the top plate of the mobile robot)

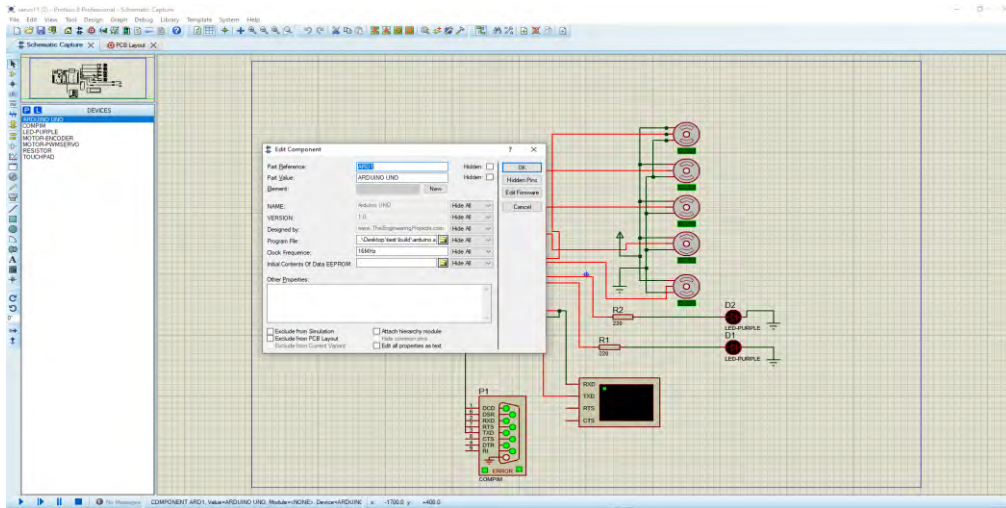


Fig.112 The loading of the Arduino file in the environment of Proteus

The rigging steps for the model to move along a certain trajectory based on the control program that is implemented in the Proteus simulation are as follows:

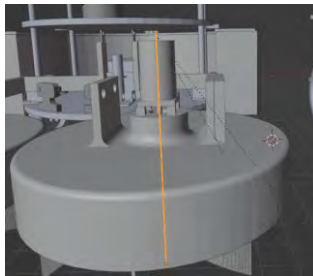


Fig.114 The generated armature For controlling the 3D object

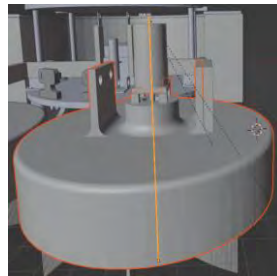


Fig.115 Positioning of the armature in the 3d object

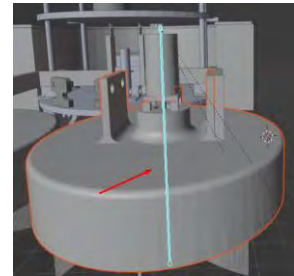


Fig.117 Choosing a bone which is to be rigged

In object mode, an "armature" (a skeletal structure that controls a 3D object) is created. Once the armature is created, it is moved to the place where we want the movement to take place and placed to the overall design of the model we wish to be rigged (Fig. 114 and Fig. 115). In object mode, once the desired pattern is selected, the armature is switched to pose mode. This is followed by a choice of bone to be erupted, i.e. attached to the model (Figure 117). Finally, ctrl + p is pressed and "bone" is selected. After the model is rigged, we add the application to Blender BlendixSerial. The BlendixSerial library is designed to work together with the blendixserial add-on in Blender. It allows controlling objects in Blender through serial communication with Arduino or other microcontrollers. By emulating the COM ports from HHD Software's Virtual Serial Port Tools, together with BlendixSerial begins the connection between the simulation in Proteus and the 3D-engineered robot with attached module in Blender

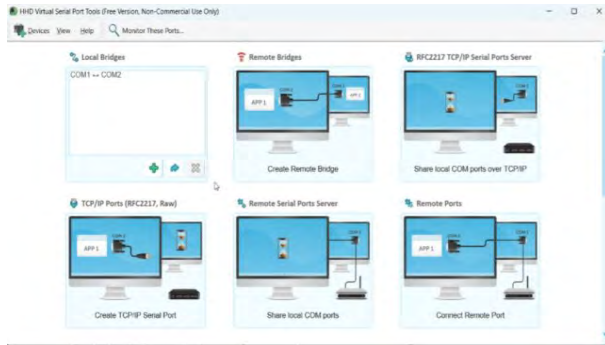


Fig.119 Emulation of the COM ports

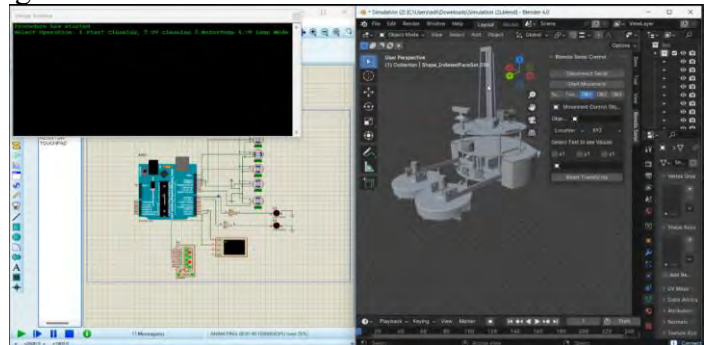


Fig.120 Start the simulation in Proteus

We run the simulation in Proteus. Using the virtual terminal, the running program is visualized, which is written in Arduino and implemented in Proteus through the generated hex file. The terminal writes that

the procedure has started and is waiting for us to introduce the cleaning mode we want to start. Possible modes of operation are:

start of cleaning, 2- UV cleaning, 3- cleaning with water, 4- mode UV lamp

Before starting the program, in accordance with the cleaning algorithm, the arms on which the cleaning brushes are mounted are automatically raised. After entering a cleaning command in the terminal, they are lowered to the extreme lower position, in which it is intended to start the rotation of the brushes and cleaning on the basis of a geometric analysis carried out in the process of designing the product. In order to visualize the changes in the 3D model of the constructed robot in the drop-down menu of BlendixSerial we must have pressed Connect Serial and Start Movement, as well as specified the emulated port. – (Fig.123)



Figure 123 Selection of COM port in the environment of Blender

After entering 1 and starting the cleaning through the front brushes, the shoulder structure is lowered and the rotation of the brushes begins. The terminal visualizes the moving of the arms and indicates that the action has been successfully performed. The move is also displayed in the menu in Blender, showing the move in height. After that, the next execution command is expected. (Fig 126) The image in figure 127 corresponds to the introduction of the cleaning mode with the brush positioned at the rear of the robot and the activated UV lamp.

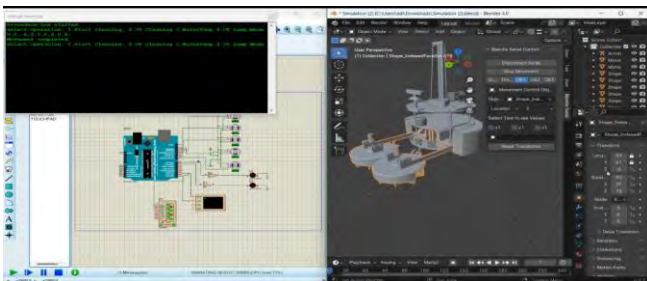


Fig.126 The executed mode in the simulation environment and anticipation of next command

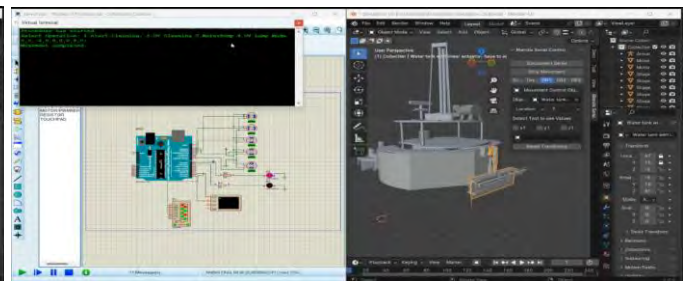


Fig.127 Start cleaning with brush and UV lamp in the simulation environment

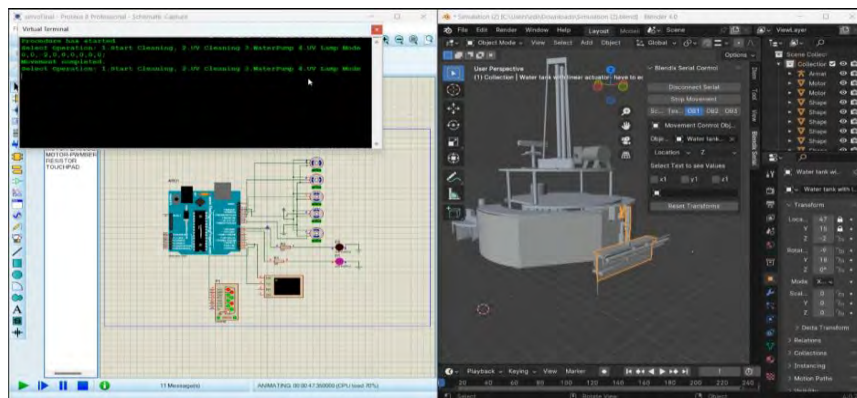


Fig.128 The activated mode for cleaning the ambient environment in the simulation

In the simulation part of Proteus, the lamp corresponding to the UV lamp in the 3D model is included

based on the code, and the visualization of the change is done with a change of color.

The introduction into the Proteus virtual terminal the mode for cleaning the environment by UV lamp is visualized as a corresponding lamp in the simulation, mounted on the top plate in Blender 3D changing color. In the case of water cleaning, the engine responsible for viewing the pump in the environment of the Proteus is activated. (Fig.128)

3.7 Control of the multi-purpose robot with attached module with antropomorphic robot

For operations requiring a certain level of precision, an anthropomorphic robot module is attached to the mobile robot platform.

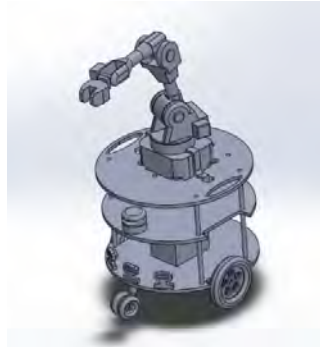


Fig.129 The module with the attached Niryo One to the mobile robot

3.7.1 Controlling software

For the functioning of the anthropomorphic robot, the Niryo One Studio control software [94] running under different operating systems (Windows, MacOS, Linux) is used. The software shown in Figure 130 is used for setting up the robot, moving the robot, programming a sequence of commands, etc.

a) The left menu of the software window has the following robot control panels:

- Niryo blocks;
- Reserved positions;
- Saved sequences (with the possibility of automatic starting);
- Settings (Wi-Fi, Raspberry Pi, software versions);
- Calibration;
- Hardware status; Troubleshoot errors and logs.

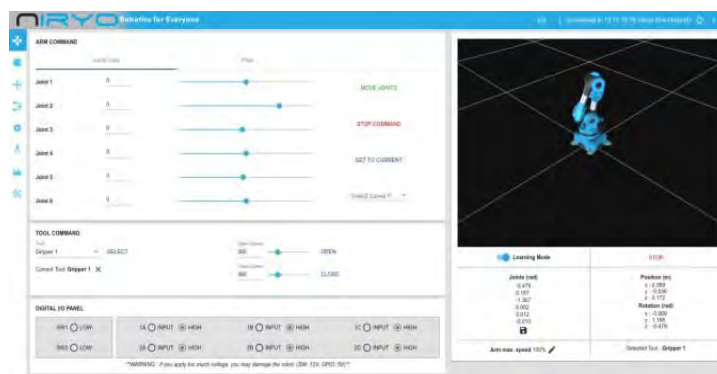


Fig.130 Niryo One control software window

b) Robot control

When proceeding to the control of the robot, a calibration must be carried out in advance, after which an indicative LED (green or blue) indicates that the robot has been successfully started. The control software of the anthropomorphic robot Niryo One Studio, has a section providing information about the robot and contains the training mode shown in Figure 131.

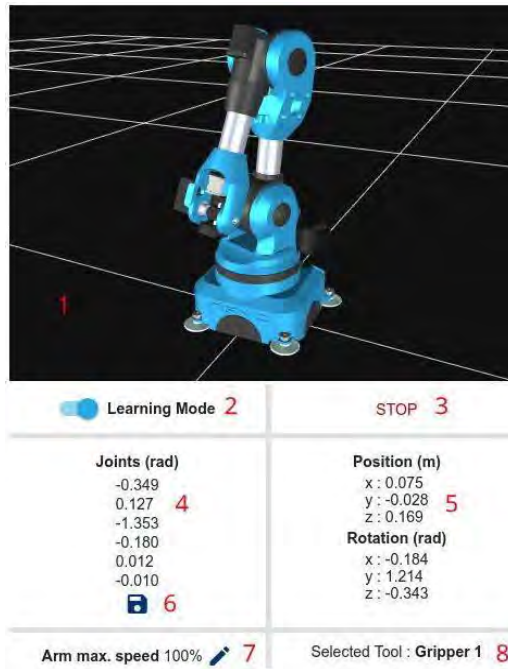


Fig.131 Permanent right section in the software window

- The software provides a 3D view of the robot. The 3D view helps to detect whether the manual/automatic calibration has been done correctly. If the physical positions of the robot differ and what is visible on the screen, the automatic calibration must be done again.
- When activating the "learning mode" button, the torque of all engines is deactivated. When the robot is in "learning mode", it can be moved freely. When the training mode is turned off, all motors are activated and the robot cannot move freely. When the training mode is turned off, all motors are activated and the robot cannot move freely
- At any time, when the "stop" button is activated, the current program of the robot is stopped.
- Current state of joints (J1-J6).
- Current position and orientation of TCP (Tool Center Point – the center of the specialized gripper - dispenser).
- A "keep current position" button allows to keep the current position, after which it can be reused later.
- Current maximum joint speed can be changed at any time (0-100%).
- Currently selected tool. C. Robot drive There are 3 tabs in the command panel of the robot control software. On the "Arm command" tab [94], an action can be selected from the "joints" tab or the "posture" tab. The "joints" tab allows to move each joint independently. The "pose" tab allows to move the position of the TCP relative to the initial x, y, z coordinates of the robot base.

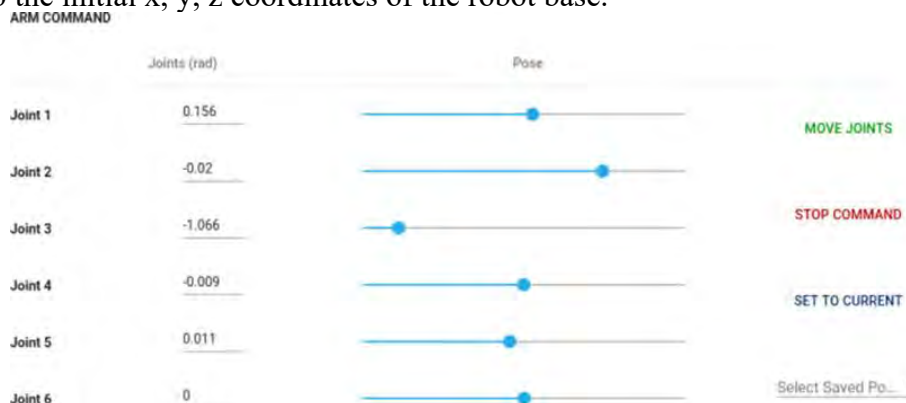


Fig.132 Arm command section in the control software

To execute a command, the following procedure is followed:

- Press the "set current position" button. The command will adjust the axes, position and orientation to the current state of the robot. This makes it possible to subsequently change the values from the current state (e.g. if desired, move only axis 5 or move TCP along the axis and keep the same orientation).
- Modification of joint or posture values (position + orientation).
- Pressing the "Move Joints" or "Move Pose" button to execute the move command.
- Once the command is executed, a notification is received at the bottom of the screen.
- At any time there is an option to cancel the current execution of the command by clicking on the "Stop" button. While running a current move command, a new command cannot be run. If you want to start a new command, you must first stop the previous one. There is also the possibility to choose a position that was previously reserved with the selection field "select a reserved position", thus replacing the first and second steps.

CONCLUSION

In this dissertation is presented the development of a multi-purpose tele-controlled service robot, through which can be carried out, monitoring, security and signaling of objects, cleaning and sanitation of premises, lifting and carrying cargo, distribution of medicines, transportation and storage of products, performing manipulation operations with elements.

The device of the robot includes a universal mobile platform and six modules, through which the robot performs completely different in nature and way of performing activities and operations. Its principle of operation is based on telecontrol.

The developed multi-purpose tele-control robot is used in completely different working environments – health facilities, retail chains, warehouses, livestock farms, expanding the range of application of service robots and contributing to their development and wide distribution. It is part of the service robots, which will increasingly improve working conditions and facilitate people's work.

In the future, the research on the service robot presented in the dissertation will be continued, and for this purpose a physical model of the robot will be realized and verified in real operating conditions.

CONTRIBUTIONS

- Major international documents, programmes and policies related to global development in the context of robotics and service robots within them have been studied and analysed.
- For each one of the robot operating modes, depending on the module used, a control algorithm has been developed and the execution sequence of each one is described.
- 4 softwares are tied - Proteus 8.16, Arduino, HHD Software's Virtual Serial Port Tools and Blender 3D for synchronized data transmission between them.
- Based on the parallel, collaborative and synchronized application of the five software (Proteus 8.16, SolidWorks, Arduino IDE, HHD Software's Virtual Serial Port Tools and Blender 3D), simulations have been compiled and implemented, through which the algorithms of the robot's work in its different modes are verified by the PhD student.
- A mobile robotic platform has been developed with the possibility of attaching different design and purpose modules.
- Six separate modules have been developed, with different functions, design and purpose, through which the robot can work in heterogeneous environments and perform completely different in nature and way of performing activities and operations.
- Tele-operation of a multifunctional mobile robotic platform through web-based graphical interface, via MQTT and web sockets has been developed

ACKNOWLEDGEMENTS

I would like to thank my supervisor prof. Dr. Nayden Shivarov, for his assistance in conducting the current research and the preparation of the dissertation.

REFERENCES

- [16] D. Chikurtev, K. Yovchev. Marker-based Automatic Dataset Collection for Robotic Vision System. *Mechanisms and Machine Science*, 102, Springer Science and Business Media B.V., 2021
- [17] D. Chikurtev, N. Chivarov, S.Chivarov, A. Chikurteva. Mobile robot localization and navigation using LIDAR and indoor GPS. *IFAC papers online*, 54, 13, Elsevier, 2021
- [44] MD Moniruzzaman, Alexander Rassau, Douglas Chai, Syed Mohammed Shamsul Islam, *Teleoperation methods and enhancement techniques for mobile robots: A comprehensive survey, Robotics and Autonomous Systems*, Volume 150, 2022
- [45] M. Mukhandi, D. Portugal, S. Pereira, MS Couceiro. A novel solution for securing robot communications based on the MQTT protocol and ROS. In 2019 IEEE/SICE International Symposium on System Integration (SII) 2019 Jan 14 (pp. 608-613). IEEE.
- [49] N. Chivarov, D. Chikurtev, I. Rangelov, A. Gigov and N. Shivarov. Educational Mobile Robot Platform for Line Following, 6th International Scientific Conference "Education, Science, Innovations", Pernik, Bulgaria, p.p. 290 – 298, 2016
- [50] N. Chivarov, D. Chikurtev, N. Shivarov, K. Yovchev, I. Rangelov, A. Gigov, M. Konstantinov, M. Marinov, V. Lazarov, E. Markov. Telecontrolled Service Robot For Increasing The Quality Of Life Of Elderly And Disabled, ADP 2017, June, Sozopol, Bulgaria; p.p. 171- 175
- [70] Z. Kapić, A. Crnkic, E. Mujčić, J. Hamzabegović. A web application for remote control of ROS robot based on WebSocket protocol and Django development environment. In IOP Conference Series: Materials Science and Engineering 2021 Nov 1 (Vol. 1208, No. 1, p. 012035). IOP Publishing.
- [74] <https://digital-strategy.ec.europa.eu/en/library/digital-economy-and-society-index-desi-2021>
- [81] <https://www.aliexpress.com/item/32813245169.html?spm=2114.12010612.8148356.6.56a9fee63LCph6>
- [84] Technical documents – Lely
- [85] Cleaning farm robot - ProCleaner X100 - WASHPOWER - autonomous / pig barn
- [86] Self-learning, mobile cleaning robot ensures absolute cleanliness in production line - Inceptive Mind
- [87] <https://vikiwat.com/elektrodivigatel-postoyannotokov-s-reduktor-12vdc-150rpm-2w-vfga37fg-32i-a601.html>
- [88] 12V Linear Actuator With Bracket Aluminum Alloy Small Electric Push Rod Mini Linear Motor|DC Motor|
- [89] Durable DC 12V 130PSI Agricultural Electric Water Pump Black Micro High Pressure Diaphragm Water Sprayer Car Wash 12 V|Pumps|
- [90] 12v Small Acuator Motor Stroke Length 30mm/50mm/100mm/150mm Load Capacity 150n Mirco Electric Linear Servos - Dc Motor
- [92] <https://niryo.com/docs/niryo-one/user-manual/mechanical-specifications>
- [93] <https://www.handsontec.com/dataspecs/module/BTS7960%20Motor%20Driver.pdf>
- [94] <https://niryo.com/docs/niryo-one/user-manual/complete-user-manual>



БЪЛГАРСКА АКАДЕМИЯ НА НАУКИТЕ

АВТОРЕФЕРАТ НА ДИСЕРТАЦИЯ

за присъждане на образователна и научна степен “доктор” по докторска програма
„„Приложение на принципите и методите
на кибернетиката в различни области на науката”

МНОГОЦЕЛЕВИ ТЕЛЕ-УПРАВЛЯЕМ СЕРВИЗЕН РОБОТ

Петко Иванов Стое

Ръководител: Проф. Найден Шиваров

Научно жури:

Доц. Мартин Камбушев
Доц. Станислав Гьошев
Доц. Стефан Билидеров
Проф. Марин Маринов
Чл.кор. Любка Дуковска



**Институт по информационни и
комуникационни технологии
Секция „Кибер-физични системи”**

Актуалност на темата на дисертацията

Роботиката и в частност сервизните работи са важна част от стратегическите политики и документи за развитие на технологичния отрасъл: „Общество 5.0“, „Индустрия 4.0“, „Индустрия 5.0“, ПРОГРАМА 2030 на ООН, Стратегическа програма на Европейската комисия за периода 2019-2024, „2030 Цифров компас: европейският път за цифровото десетилетие“, "Европейска цифрова стратегия", "Европа, пригодена за цифровата ера", Програма "Хоризонт Европа", Европейската зелена сделка, План за действие за кръговата икономика на ЕС. Тяхното развитие и внедряване е основен фактор за напредъка на технологиите, подобряването на безопасността, повишаването на ефективността и справянето с важни обществени, икономически и екологични предизвикателства. Като бързоразвиваща се сфера многоцелевите теле-управляеми сервизни работи ще имат все по-голямо присъствие и ще играят все по-важна роля в съвременния свят.

Цел и задачи на дисертацията

Целта на дисертацията е разработване на многоцелеви телеуправляем сервизен робот, предназначен да подпомогне хората в тяхната трудова дейност, свързана с обход, наблюдение, охрана и сигнализация на обекти, почистване и хигиенизиране на помещения, повдигане и пренасяне на товари, дистрибуция на лекарства, транспортиране и складиране на продукти, извършване на манипулационни операции с елементи.

Разработката разглежда следните проблемни области на многоцелевите сервизни работи: механичен дизайн, сензорна система, промишлен дизайн и ергономия, задвижване и изпълнителни механизми, комуникация и телеуправление, взаимодействие човек-робот.

Постигането на поставената цел е свързано с изпълнението на следните конкретни задачи:

1. Запознаване с основните характеристики и специфични особености на телеуправляемите сервизни работи
2. Проектиране на основните компоненти на механичния дизайн на робота:
 - Универсална мобилна платформа
 - Модул с повдигаща платформа
 - Модул с ротационна маса
 - Почистващ модул
 - Модул с ножичест подежник
 - Модул с антропоморфен робот Niryu
3. Избор на задвижване, изпълнителни механизми и сензорна система на робота
4. Разработване на промишлен дизайн на робота
5. Разработване на дистанционно управление на телеуправляемата многофункционална мобилна роботизирана платформа
6. Създаване на работни режими на робота
 - Разработване на алгоритми за управление
 - Съставяне и реализиране на компютърни симулации и верифициране на алгоритмите за управление

Публикации на докторанта по темата на дисертацията

На международни конференции в чужбина:

1. Stoev, P., Chikurtev, D., Stefanov, T., Dimitrov, D. & Vitanova, D. (2023). Remote Control of a Teleoperated Multi-Purpose Mobile Robot Platform Using a Web-Based Graphical Interface, via MQTT and Web Sockets. The 3rd International Conference on Electrical, Computer, Communications and Mechatronics Engineering (ICECCME) 2023
2. Stoev, P., Yovchev S., Chikurtev, D., Chivarov, N. Development of a Teleoperated Mobile Robot with Module for Cleaning Animal Husbandry Complexes. TechSys 2022, Accepted for

publishing

3. Nayden Chivarov, Denis Chikurtev, Petko Stoev, Vasil Lozanov, Stefan Chivarov. ROBCO Drone - Service Robot for Transport and Delivery of Grocery Products, 7th International Conference on Engineering and Emerging Technologies **2021** – ISTANBUL – TURKEY

На международни конференции у нас:

1. Stoev, P. Development of a Lifting Module of a Mobile Robot for Warehouses. ADP **2022**, Pages 259-261, ISSN: 2682-9584
2. Васил Лозанов, Найден Шиваров, Денис Чикуртев, Стефан Шиваров, Петко Стоев. Индустиален дизайн на сервизен складов робот „РОБКО“, Годишна международна научна конференция на ВВВУ "Георги Бенковски" **2021**

Цитати:

1. Nayden Chivarov, Denis Chikurtev, Petko Stoev, Vasil Lozanov, Stefan Chivarov. ROBCO Drone - Service Robot for Transport and Delivery of Grocery Products, 7th International Conference on Engineering and Emerging Technologies **2021** – ISTANBUL – TURKEY

УВОД

Многоцелевият теле-управляем сервизен робот е универсална роботизирана система, предназначена да изпълнява широк спектър от задачи и услуги в различни области и сфери, посредством дистанционно управление от човек - оператор.

В настоящия дисертационен труд се разглеждат фундаментални проблеми свързани с механиката, индустриалния дизайн, ергономията, механичната съвместимост и телеуправлението на многоцелевите теле-управляеми мобилни роботи. Представена е разработка на многоцелеви теле-управляем сервизен робот, чрез който могат да бъдат осъществени обход, наблюдение, охрана и сигнализация на обекти, почистване и хигиенизиране на помещения, повдигане и пренасяне на товари, дистрибуция на лекарства, транспортиране и складиране на продукти, извършване на манипулационни операции.

Настоящата дисертационна работа е структурирана както следва:

ГЛАВА 1: Разгледани са основните аспекти на механиката и дизайна на многоцелевите теле-управляеми сервизни роботи, в това число специфичните особености и системите на сервизните роботи, механична съвместимост и механичните системи за тяхното придвижване, дизайнът и ергономията на роботите, комуникация при сервизните теле-управляеми роботи. Посочена е спецификата на взаимодействието човек-робот в контекста на многоцелевите теле-управляеми сервизни роботи.

ГЛАВА 2: В Глава втора е представен дизайнът на разработения многоцелеви теле-управляем сервизен робот. Подробно са описани механичната система на робота, неговото задвижване и изпълнителни механизми, сензорната му система и реализирания дизайн.

ГЛАВА 3: В Глава трета е представено управлението на многоцелевия робот, като за всеки режим и избран модул са разработени от докторанта алгоритъм на работа, софтуер и хардуер на управление. Разгледано е дистанционното управление на телеуправляемата многофункционална мобилна робот-платформа. Проведени са симулации в компютърна среда, на базата на които са верифицирани алгоритмите за управление.

ГЛАВА I: ТЕОРЕТИЧЕН ОБЗОР

1.3 Основни характеристики на многоцелевите роботи

Основна характеристика на многоцелевите роботи е модулният им дизайн, който дава възможност на потребителите лесно да добавят или премахват компоненти. Тази модулност

позволява на робота да бъде персонализиран за конкретни задачи и прави имплементацията на подобрения или адаптациите по-лесни. Многоцелевите роботи са оборудвани с разнообразни елементи за манипулация, като роботизирани ръце с множество степени на свобода и адаптивни хващачи. Посредством тази гъвкавост те могат да боравят с различни обекти и да изпълняват разнородни задачи. В зависимост от предвидените приложения, многоцелевите роботи могат да имат различни системи за мобилност. Изборът на мобилност зависи от терена и задачите, с които роботът ще се сблъска. Многоцелевите роботи включват разнообразни сензори за възприятие, навигация и взаимодействие с околната среда. Безопасността е от първостепенно значение, особено когато роботите работят заедно с хората. Усъвършенстваните алгоритми за навигация, избягване на препятствия и вземане на решения допринасят за тяхната автономност. Енергийно ефективните компоненти, системите за управление на храненето и технологиите за батерии се интегрират при мобилните роботи, за да оптимизират консумацията на енергия на робота. Многоцелевите телеуправляеми роботи разполагат с разширени комуникационни възможности, включително Wi-Fi, клетъчна и сателитна свързаност, посредством които се постига дистанционно наблюдение, контрол и предаване на данни. Нов аспект, свързан с проектирането на роботи и в частност на многоцелевите телеуправляеми роботи е интегрирането на AI с цел подобряване на тяхната адаптивност и способности за вземане на решения. AI позволява на роботите да се учат от опита и да подобряват ефективността си с течение на времето.

1.5 Механична съвместимост

Механичната съвместимост е съществен аспект при проектирането на модулни роботи. Тя се отнася до съвместимостта между механичните части на робота и модулите, които се прикрепят или свалят. Механичната съвместимост включва съвместимост на размерите, здравина и носимост, стандартизирани интерфейси, обслужване.

1.7 Дизайн, ергономия и нормативни изисквания

Като медиатор дизайнът играе съществена роля в изграждането на успешна комуникация между човека и робота. Той представлява онова специфично звено, без което възприемането, а от там и подходът на човека спрямо робота би бил различен. Дизайнът на роботи е отговорна задача, с множество параметри, които трябва да бъдат съобразени, като се вземе предвид средата, в която робота ще бъде поставен, за да може комуникацията човек - робот да бъде осъществена правилно. За унифициране и полагане на основите на стандартизация в дизайна на роботи, се визират ергономични изисквания, психологични аспекти на формите и цветовете, тяхното психофизиологично въздействие и породените на тази база емоции и действия на потребителя.

1.9 Теле-управление на сервизните роботи

Посредством телеуправлението сервизни роботи разширяват своите среди на имплементация и функционалностите, които могат да имат. В контекста на сервизните роботи теле-управлението се отнася до дистанционното управление на движенията и действията на робота от оператор-човек с помощта на интерфейс за управление. Има широк спектър от методи за дистанционно управление и насочване на роботи, като всеки метод има своите специфични предимства и предизвикателства, допринасяйки за разнообразния спектър от възможности за телеуправление.

1.10 Взаимодействие човек-робот в контекста на многоцелеви теле-управляеми сервизни роботи

Взаимодействието човек-робот е мултидисциплинарна развиваща се област, чиято цел е постигането на безопасно, продуктивно и безпроблемно сътрудничеството между хората и роботите. В контекста на многоцелеви теле-управляеми сервизни роботи главните насоки за постигане на ефективно взаимодействие човек-робот са свързани с проектиране на интуитивни интерфейси за управление, които са удобни за оператора и изискват минимално усилие; намаляване на когнитивното натоварване на оператора чрез въвеждане на споделени режими на управление, при които робота автономно подпомага навигацията и избягването на препятствия; предоставяне на обратна връзка в реално време на операторите, включително видео изображения на живо от камерите на робота, данни от сензори и актуализации на състоянието, хиптични

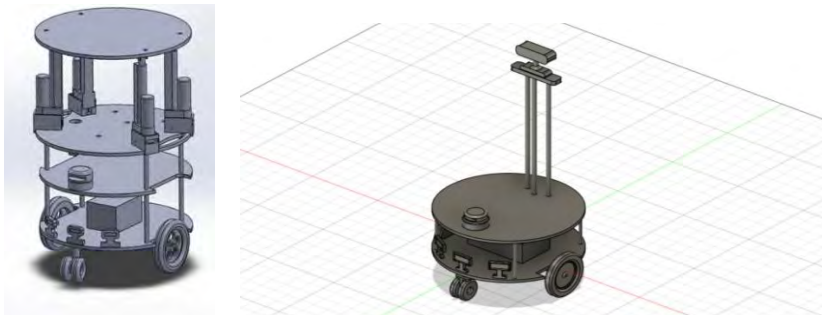
обратни връзки; прилагане на механизми за безопасност, включително бутони за аварийно спиране и системи за откриване на препятствия, за да се гарантира безопасността както на оператора, така и на робота; внедряване на криптиране и контрол на достъпа, особено в случаите, при които роботът улавя чувствителни данни или работи в специална среда; отговорно използване на роботите в контекста на етичните аспекти на взаимодействието човек-робот, като поведение на робота, вземане на решения и отчетност.

ГЛАВА II: ДИЗАЙН НА МНОГОЦЕЛЕВИ ТЕЛЕ-УПРАВЛЯЕМ СЕРВИЗЕН РОБОТ

2.1 Механичен дизайн

2.1.1 Универсална мобилна платформа на многоцелевия робот

Разработената платформа (фиг. 12) има за цел да даде възможност за лесно монтиране на допълнителни модули, които поставени на робота да позволяват изпълнението на специфични задачи в различни среди и да разширят неговите функционални възможности. В зависимост от модула, който се свързва към платформата тя може да бъде с две, три или четири плочи. Те са изработени от полиетилен с висока плътност (HDPE). Дебелината на всяка от плочите е 9,5 мм, а крайните точки за всяка плоча описват кръг с диаметър 450 мм. Всяка от плочите се фиксира към следващата плоча с помощта на група от четири пръта, с диаметър 12,5 мм на всеки от прътите.



Фиг.12 3D изображение на проектираната мобилна роботизирана платформа

Разстоянието, което прътите осигуряват между плочите, е както следва: 152 мм между първата и втората и 127,5 мм между втората и третата, като плочите се номерират по ред, започвайки от повърхността, върху която е поставен роботът. Под първата плоча е разположена допълнителна плоча с размери 336,6 мм x 177,8 мм с дебелина 9,5 мм. Разстоянието между допълнителната плоча и първата плоча е 78 мм, измерено от долната повърхност.

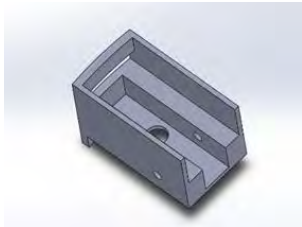
Височината, измерена от повърхността, върху която е поставен роботът, до върха на първата плоча е 106 мм. В първата и втората плоча се прави разрез с размери 165 мм x 38 мм външно, допирателно към всяка от плочите. Разрезът в първата плоча се използва за поставяне на задвижващите колела в допълнителна по-малка плоча, прикрепена под първата плоча, върху която са монтирани електрическите двигатели на задвижващите колела, както и задвижващата батерия.

Платформата е оборудвана с четири колела за придвижване на робота: две диференциално задвижвани колела с диаметър 155 mm и две помощни колела, разположени в предния и задния край, с диаметър 76 мм. Колелата са прикрепени към ос с диаметър 12,75 мм. Конструиранията платформа е с диференциално задвижване, което дава възможност за управление на движенията на робота чрез регулиране на скоростта на въртене на двете колела. Двете задвижващи колела, както и батерията, която ги захранва, и помощните колелата са прикрепени под първата плоча. Задвижващите колела се задвижват от два четкови мотора с мощност 96 W всеки, с червячна предавка, работещи при напрежение 12V, оборудвани с 36-позиционни енкодери. Моторният възел генерира приблизително 85 инча/Lbs въртящ момент. Върху горната плоча са монтирани камера и LIDAR, чрез които се осъществява телеуправлението. Допълнителни сензори за откриване на препятствия са поставени на долната плоча.

2.1.2 Модули на многоцелевия робот

2.1.2.1 Модул с повдигаща платформа

Чрез този модул роботът може извършва доставка и складиране на стоки в магазини и складове. За целите на повдигането на опаковани доставки се добавя 4-та плоча към горната плоча на мобилната роботизирана платформа. Четвъртата плоча е монтирана върху четири линейни мотора чрез 8 L-образни профила - по 2 за всеки линеен двигател. Линейните задвижващи механизми са свързани към 3-та платформа на робота с помощта на специално проектирани слотове, които са фиксирани върху 3-та платформа посредством болтове. Размерите на слота, предназначен за линейния задвижващ механизъм, са 80mm x 46mm x 45mm (дължина / ширина / височина). Направена е допълнително построение от слота до 3-та плоча, коцентрично към плочата, за по-нататъшна стабилност на слота. Проектиран е канал с ширина 18,4 мм и височина 18 мм, в който е поставена долната част на задвижващия механизъм, както и отвор в долната част на слота с диаметър 12,5 мм, използван за закрепване към 3-та плоча чрез болтова връзка. От двете страни на слота са проектирани отвори с размери 6 мм, които се използват за фиксиране на линейния задвижващ механизъм на място. Освен това в задната част на слота е направен разрез с размери 40mm x 12mm, през който преминава захранващият кабел на задвижващите механизми.



Фиг.13 Проектираните слотове за линейните задвижващи механизми

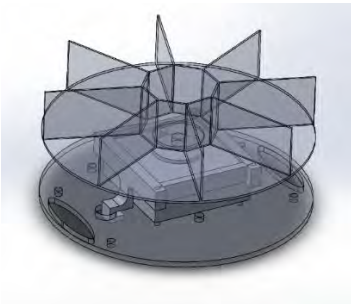


Фиг.14 Задвижващ механизъм на повдигащата платформа

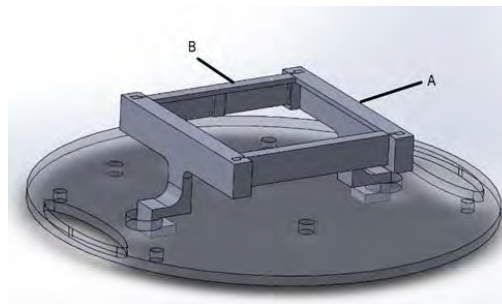
Линейният задвижващ механизъм, използван в разработката, е 12 волта с 90 кг максимална товароносимост. Задвижващият механизъм използва потенциометър за обратна връзка. Размерите на задвижващия механизъм [81] в прибрано състояние са 75 x 40 x 258. Ходът на задвижващия механизъм е 100 мм. В долната и горната част са проектирани 6 мм отвори, през които се закрепва. L-образните профили за закрепване на задвижващия механизъм към 4-та плоча са 20mm x 20mm x 25mm с дебелина на стената 2 mm.

2.1.2.2 Модул с ротационна маса

Въртящата се маса се състои от две общи части - основно тяло, в което са поставени компонентите за управление и задвижване и ротационен диск с прорези за съхранение и транспортиране на продукти. Проектираният модул се закрепва към робота чрез рамка, която се монтира към горната плоча. Конструкцията на рамката се състои от две двойки компоненти за по-лесно монтиране към основното тяло на въртящата се маса. Основното тяло е с размери 177mm x 177mm x 65mm /ширина x дължина x височина/. Дебелината на използваната ламарина е 2 мм. Прикрепени към дъното му, разположени на 6 мм от всеки от 4 гумени крачета с диаметър 20 мм и височина 11 мм за допълнителна стабилност.



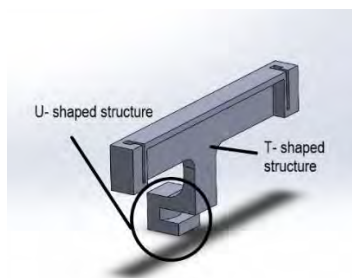
Фиг.15 Ротационната маса и начинът на захващане към плочата на робота



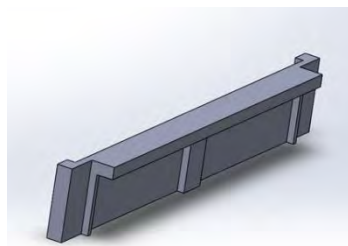
Фиг.16 Рамката за фиксиране на ротационната маса- детайли А и Б

Между основното тяло и въртящия се диск предавката за въртеливо движение е разположена външно, като големите и малките зъбни колела са със съответни диаметри от 85 мм и 13 мм. Разстоянието между дъното на въртящия се диск и горната част на основното тяло е 35 мм. Диаметърът на въртящия се диск е 450 мм. В центъра на въртящия се диск е поставен правилен осмоъгълник със страна 50 мм. Всеки от 8-те слота, разположени на 45 градуса един от друг, има външна дължина на свода 225 мм и дължина, измерена от центъра на арката до съответната страна на осмоъгълника от 164,6 мм. Слотовете също са изработени от ламарина с дебелина на листа 1,5 мм. Височината на всеки слот е 90 мм.

Рамката за монтиране на въртящата се маса към горната плоча на робота е изградена с помощта на 3D принтер. Състои се от две двойки идентични елементи: част А и част Б, които се плъзгат един в друг. Част А има U-образен профил. Рамката, фиксираща въртящата се маса, е прикрепена към горната плоча на мобилния робот с помощта на двата U-образни профила, чиито вътрешни страни са допирателни към отворите, в които влизат. Част Б влиза в специално проектираните канали на част А, като по този начин затяга противоположните части и сглобява рамката. Детайл А е комбинация от T-образен профил, който се поставя в основното тяло на въртящата се маса и U-образен профил за закрепване към горната плоча. T-образната част има следните размери: 213мм x 20мм x 35мм. Дебелината на профила е 5 мм. U-образната част има радиус 17,5 мм в частта, допирателна към отвора и обща ширина 28,5 мм. Каналите за сглобяване на рамката са 15мм x 6,5мм.



Фиг.17 Детайл А



Фиг.18 Детайл Б

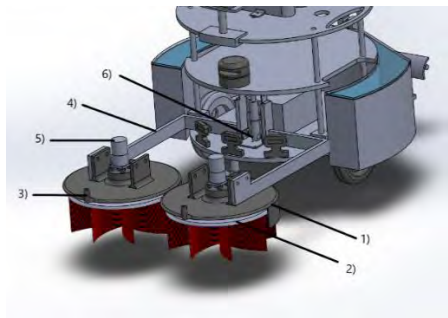
Детайл Б има следните размери, описани от крайните му точки: 187 x 16,5 x 30 мм /дължина x ширина x височина/. Три допълнителни ребра са проектирани от вътрешната страна на детайла, за да поддържат допълнително хоризонталното фиксиране на въртящата се маса. Дебелината на профила на частта е 3 мм, а за оребрените части с допълнителна опора е 6 мм.

Чрез проектирането на такава 3D отпечатана рамка, и в двата случая на нейната работа и движение на робота, се постига желаната стабилност и фиксиране на въртящата се маса.

2.1.2.3 Почистващ модул

Модулът за почистване се състои от няколко отделни системи, които могат да работят едновременно или независимо. Първата система е комбинация от две еднакви въртящи се четки [84], свързани с механизъм, който им позволява да се спускат или повдигат в зависимост от това дали роботът само се движи или е в режим на почистване. Втора спомагателна система е комбинация от преносим резервоар с резервоарна помпа, която при необходимост изпуска почистващ препарат или вода през дюзи [85][86], разположени пред въртящите се четки. Другата подсистема, която допринася за процеса на хигиенизиране представлява регулируема гумена четка, която подобно на въртящите се четки се движи вертикално с помощта на линеен задвижващ механизъм, и втора UV лампа са монтирани към резервоара, в задната част на робота

Механичната система с въртящи се четки (фиг. 19) е проектирана така, че да осигурява контакт с повърхността, която се почиства по всяко време, когато е активна и роботът извършва процеса на хигиенизиране. В най-ниско положение, когато четките са активни, конструкцията осигурява достатъчно налягане за извършване на технологичната процедура.

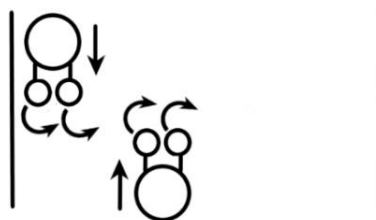


Фиг.19 1- капак на четката, 2- почистващ диск, 3- дюза за препарат, 4- структура на повдигащи рамена, 5- електрически мотор, 6- линеен задвижващ механизъм

Активирането на системата за изпускане на препаратите се извършва от оператор и се определя от вида на помещението, което се хигиенизира. Обратната връзка в този случай се базира на телеуправлението на робота и монтираната камера. На горната плоча роботът е оборудван с антибактериална UV лампа за отстраняване на различни патогени в околната среда, която не е в контакт с четките на разработения модул. Регулируема гумена четка, която подобно на въртящите се четки се движи вертикално с помощта на линеен задвижващ механизъм, и втора UV лампа са монтирани към резервоара, в задната част на робота. Гумената четка се използва за избутване на използвания препарат от почистваната с въртящи се четки повърхност. Допълнителната втора UV лампа е насочена директно към повърхността на пода, който роботът почиства, като се използва като последна технологична операция в процеса на саниране.

Системата с ротационни четки се състои от следните няколко компонента. Първият компонент е почистващият диск с диаметър 250 мм. Върху него се поставя капакът на четката с диаметър 270 мм. От капака на четката се екструдира полукръг с височина 55 мм, който е обърнат към мобилния робот, за да предпази робота от замърсяване по време на работа на ротационната четка. В предния край на капака е проектирана дюза, през която при необходимост се изпуска препаратите от резервоара. В центъра на капака са разположени електродвигателят и планките за закрепване на четките към повдигащите рамена. За електродвигател е избран 12V с редуктор на скоростта с мощност 150 об./мин. Повдигането и спускането на четката се осъществява с помощта на две повдигащи рамена, свързани с лост, образуващ кобилица, закрепена към линеен задвижващ механизъм. Актуаторът се фиксира с държач и се поставя между сензорите и батерията на първата платформа.

При почистване с четки посоката им на въртене се определя от това коя страна на робота е по-близо до стена на помещението. Почистването се извършва от близката към далечната стена, т.е. четките избутват отпадъците настрани, след което роботът се движи в обратна посока и въртенето на четките се променя. Това продължава, докато стигне до другия край на стаята. Принципатът е илюстриран на фигура 20.



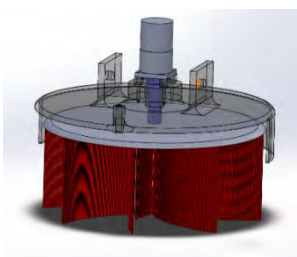
Фиг.20 Принцип на въртене на почистващите четки и движението на робота в стаята, подложена на почистваща процедура

По време на проектирането на модула четките са предназначени за използване за твърди повърхности. Движението на робота в помещенията на комплекса и изпълнението на технологичните операции на хигиенния модул се осъществяват както на базата на вътрешна навигация, така и на телеуправление.

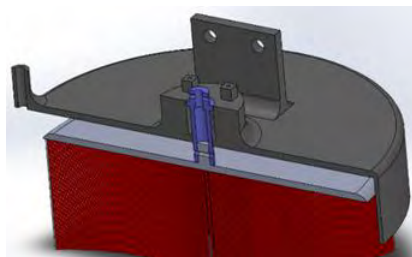
Втората система на хигиенния модул се състои от резервоар за препарат и помпа за резервоар,

които се активират при необходимост. Избраната помпа за резервоар е 12V с максимален дебит от 5 l/min, която е монтирана на третата плоча. Резервоарът за препарат се поставя на долната плоча. Закрепва се към мобилния робот чрез 4 пластини с отвори, които влизат в прътите между първата и втората пластина.

Системата с ротационната четка се състои от следните няколко компонента. Първият компонент е дискът с почистващите пера. Дискът е с диаметър 250мм и 5мм дебелина на плочата, като общата височина е 105мм. В центъра на диска е екструдирана втулка с диаметър 9мм и височина 15, която се монтира към фланеца, предаващ въртеливото движение от електродвигателя, посредством два болта М3 минаващ напречно през сглобката (фиг. 22).

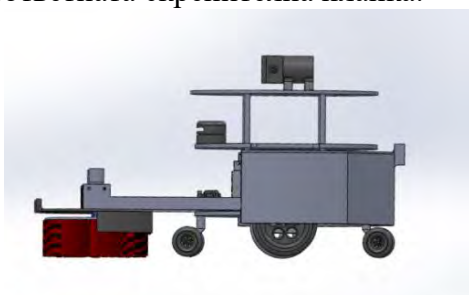


Фиг.21 Сглобка на ротационна четка

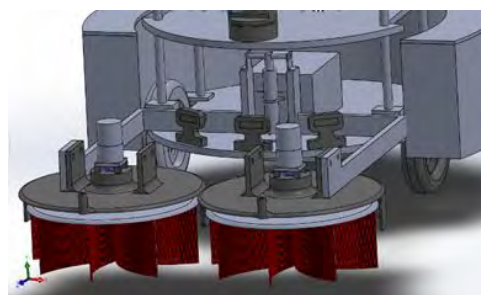


Фиг.22 Разрез на ротационната четка

Свързващият четката с електродвигателя фланец е с дължина 55мм. Малкият диаметър е 15мм, а големият, който ляга върху лагер и го покрива като капак с диаметър 21мм. Часта която е върху лагера е с дебелина 4мм. За лагер е избран иглен лагер с вътрешен диаметър 15мм, външен 21мм и дебелина 22мм, който се поставя в проектирано за целта гнездо поместено в конструкцията на капака. Капакът на четката е с диаметър 270мм и дебелина 5мм. Екструдирана е полуокръжността от капака с дължина 55мм, която е с лице към мобилния робот, с цел предпазване на робота от замърсяване при работа на ротационната четка. В центъра на капака поместено цилиндрично тяло с външен и вътрешен диаметър съответно 60 и 21мм, и височина 26мм. Към горната повърхност на цилиндричното тяло са конструирани 4 гнезда с размери 8мм x 8мм x 9мм. В гнездата са поместени отвори с дължина 30мм за болтове М3, посредством които се монтира електродвигателя. За електродвигател [87] е избран 12V с редуктор с 150 об./мин. Захващането на ротационните четки към повдигащото рамо се осъществява чрез планки с размери 60мм x 65мм x 7,5мм, на отстой 115мм една от друга, в които са поместени отвори с диаметър 10мм. Спрямо това дали ротационната четка се захваща от лявата или дясна страна се ползва съответната скрепителна планка.



Фиг.23 Страничен изглед на робота със спусната ротационна четка

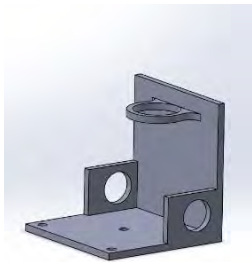


Фиг.24 Сглобка на модул с ротационни четки монтирани към повдигащия механизъм

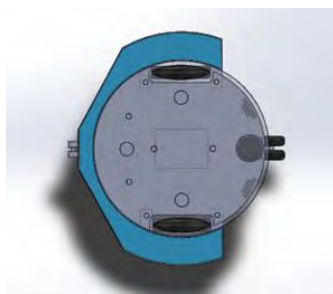
Вдигането и спускането на четката се осъществява посредством две повдигащи рамена (фиг. 23.), свързани посредством лост образувайки кобилица, захванати към линейен актуатор. Актуаторът [88] е фиксиран посредством специално проектирана планка-държач (фиг.25), като е поместен между сензорите и батерията на първата платформа. Рамената са изградени от 30мм x 10мм x 1,5мм профили. Рамената са скрепени към линейни плъсгащи лагери, които са монтирани към предните два пръта (фиг.24). За да осъществи четката контакт с повърхността за почистване е необходимо буталото на линейния актуатор да достигне крайно долно положение. В тази

позиция актуаторът е на 113,5 мм от плочата считано от отвора в буталото. При повдигане в крайно горно положение на буталото се задава 30мм ход. Така се осигурява достатъчно пространство, за да може роботът да се придвижва без да закачат перата на четката когато режимът за почистване не е активен.

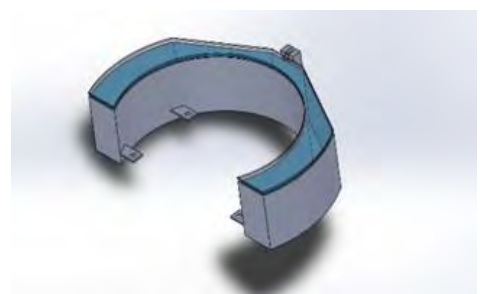
Към почистващата ротационна четка е добавена спомагателна система посредством помпа [89] и резервоар, която при необходимост да се включва (фиг.26а и фиг.26б). За помпа е избрана 12V с максимален дебит от 5л/мин. Помпата се монтира на третата плоча посредством 4 М3 болта. Резервоарът за вода е с вътрешна страна коцентрична и равна на плочите на мобилният робот, като крайните точки описват окръжност с диаметър 600мм.



Фиг.25 Държач за
линеен актуатор



Фиг.26а. Резервоар за вода,
монтиран към робот платформата

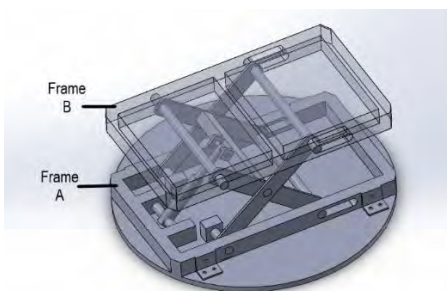


Фиг.26б. Резервоар за вода

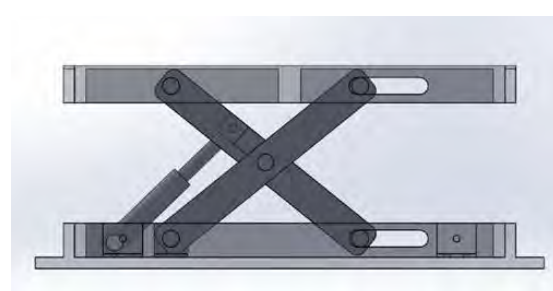
Височината на резервоара е 170мм като на 30мм от основата са заварени 4 планки с отвори, които влизат в прътовете между първата и втора плочи и така го скрепят към цялата мобилна платформа. В задната си част резервоара има елемент, посредством който се закача капака на резервоара.

2.1.2.4 Модул с ножичест подежник

Ножичестият подежник е изграден от две рамкови конструкции/ frame structures/ - рама А и Б, с идентични външни размери и профили изграждащи ги, като към тях са фиксирани другите компоненти на цялостната сглобка. Профилите ползвани за външните рамки са 20мм х30мм х2мм. Рамките са с размери 400мм х 240мм като ъглите им са скосени с цел рамата да бъде възможно най-близко по контура на плочата на робота. В сгънато положение ножичестият повдигач е на 60мм височина от платформата на робота, към която е монтиран, а в максимално повдигнато почти 172мм. Рама А се захваща към горната плоча на мобилния робот посредством 4 г-образни профила с размери 25х 35 х 3, като всеки от 4-те профила се захваща с по 3 болта М6.



Фиг.27 Сглобка на ножичест подежник



Фиг.28 Изглед на механизма с
компонентите на ножичестия подежник

Идентично и за двете рами в единия край на всяка от тях в профилите са фрезирани канали, в който се помества лагерните тела и осите, за които са захванати съответните две плъзгащи рамена за повдигане на рама Б, а в другия край са разпробити отвори с диаметър 15мм. Дължината на фрезирания канал е 70мм, в който е набит иглен лагер с външен и вътрешен диаметър съответно 21мм и 15мм. В другия край на рама А, вътрешно на конструкцията, е захванат линейния актуатор [90]. Буталото на актуатора е с диаметър 12мм и ход 100мм. Линейния актуатор е закрепен към два профила- 20мм х 30мм с дължина 50мм, като центъра на захващане е на 13мм от плочата на

робота т.е. под централната ос на профилите, за да се гарантира по-голямо скосение при прибрано положение на ножиците. Към рама Б напречно по средата е заварен допълнителен профил, който да повиши здравината и якостта на конструкцията при повдигане на товари. В горната част върху рама Б е заварен листов материал със същата форма и дебелина на листа 3мм, на който да бъдат полагани различните продукти. Повдигащите рамена са с размери 235мм x 30мм x 4мм със закръгление на ръба R8. Във всяко от рамената са разпробити 3 отвора с диаметър 15мм- един централно и два във всеки от краищата на разстояние 214.80 мм един от друг. Към две от рамената е заварен профил 20мм x 20мм, към който се закрепя буталото на линейния актуатор. Върху горната повърхност, контактуваща с повдиганите от робота изделия, са поставени сензори [74] за тежест, които да предпазват мобилния робот от претоварване

2.1.2.5 Модул с антропоморфен робот Нирио

За целите на разработения сервизен робот беше избран антропоморфният робот Niryu One [92] със следната спецификация: 6-осен робот с шест степени на свобода. Размерите на дължините на компонентите, съставляващи конструкцията, са както следва: основа - 103 мм, рамо - 80 мм, първа фуга - 210 мм, втора става - 180 мм.



Фиг.29 Антропоморфен робот Niryu One

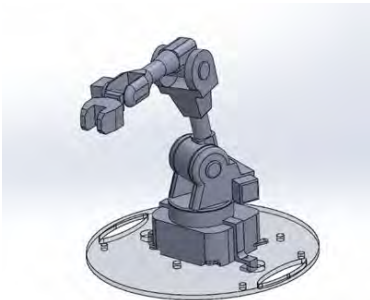


Фиг.30 Ставите на робота

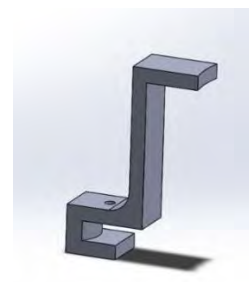
В задния панел на Niryu One са поместени множество различни физически интерфейси:

- Горен бутон;
- LED;
- 4 броя USB порт;
- GPIO панел (Общо 6 цифрови щифта);
- Съединител на захранващия адаптер.
- Ethernet порт на Raspberry Pi 3B;
- Изход за превключвател 12 V;
- Конектор Dynamixel XL-320;
- Конектор Dynamixel XL-430;
- Превключвател на захранването;
- CAN шинна връзка за Niryu Steppers;

Роботът тежи 3,3 кг. Възможният максимален обсег с монтиран хващач е до 440 мм. В долната част на Niryu One са поставени 4 вендузи и гумени крачета, през които е свързан към специфична работна повърхност. Максималният ъгъл на въртене за всяка от ставите е както следва: J1: -175 ° до 175 °, J2: -90 ° до 36.7 °, J3: -80 ° до 90 °, J4: -175 ° до 175 °, J5: от -100 ° до 110 °, J6: от -147.5 ° до 147.5. Местоположението на ставите е показано на фиг.30.



Фиг.31 Принципът на закрепване, използван за фиксиране на Niryu One към горната платформа



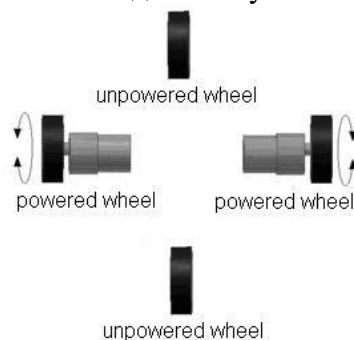
Фиг.32 Държач за фиксиране Niryu One

Държачът за закрепване и осигуряване на допълнителната стабилност, необходима за Niryu One е комбинация от U-образен профил, който влиза в отвор в горната плоча, и L-образен

профил, чиято горна част е корадилна и в контакт с основата при съединение J1, като по този начин роботът се фиксира в стабилно положение. Държачът се монтира към плочата чрез нейната U-образна част, която се затяга с M5, като се използва принципът на менгемето. Размерите на U-образната част са 23,5 мм x 25 мм /ширина x дължина/ и дебелина 7 мм. Г-образният профил е със следните размери: 71мм x 32мм x 22,5мм /височина x дължина x ширина/ и 7мм дебелина. Чрез проектирания държач в комбинация със смукателните вендузи и гумените крачета на антропоморфния робот се гарантира необходимата стабилност по време на работа.

2.2 Задвижване и изпълнителни механизми

Избраната мобилна роботизирана платформа е от типа с диференциално управление [49]. Този тип платформи имат два мотора, разположени от лявата и дясната страна, които задвижват независимо един от друг двете странични колела. За да се постигне стабилност на платформата използваме две помощни колела отпред и отзад. Това разположение на колелата позволява въртене на място, но при неравни повърхности може да се изгуби контакт между някое колело и земята.



Фиг.33 Движение на мобилната робо-платформа

- Движение на двете задвижвани колела напред - постига движение на робота напред.
- Движение на двете задвижвани колела назад - постига движение на робота назад.
- Движение е на едното колело напред, а другото назад - постига завъртане на робота в малък кръг, чийто център се намира между двете задвижвани колела.
- Движение на едно колело по-бавно от другото - постига завъртания на робота в посоката на по-бавното колело. Колко бързо се завърта зависи от това колко голяма е разликата между двете скорости

Управлението на диференциална мобилна платформа е сложно, понеже се изисква координация и сътрудничество между две отделно задвижвани колела [50].

2.3 Сензорна система

Сензорните системи са критично важна част от всеки проект за мобилен робот. Те помагат за осъществяване на всички функции на взаимодействие с околната среда – следене на пространственото положение и ориентация на робота, поддържане на траектория, откриване на препятствия, безопасност на робота и хората, работещи с него. В зависимост от основната си функция сензорите, които са използвани в мобилните роботи, могат да се разделят на няколко категории: тактилни сензори, сензори за близост, сензори за ускорение и ориентация, лазерен скенер (LIDAR), визуално базирани системи, енкодери на колелата, ултразвукови сензори.

2.4 Промишлен дизайн на робота

2.4.1 Основни положения

При многоцелеви мобилен робот целта би следвало да бъде максимално изчистен и неутрален дизайн, който да се вписва в работната среда на различни по вид съоръжения. Конструкцията, изграждаща външния вид на робота, е сегментирана на отделни панели. Спрямо конкретния сегмент, изграждащ пластиката на дизайна, са поместени конкретни акценти, които кореспондират с двигателните звена, сензорите и модулите на многоцелевия робот. Графичните линии, изобразени около отворите на ултразвуковите сензори служат като акцент за тяхното утилитарно поместване в дизайна. В задната част на конструкцията е позициониран отвор с капак, който предоставя достъпа до буксата за зареждане на робота както и за изглед към дисплея,

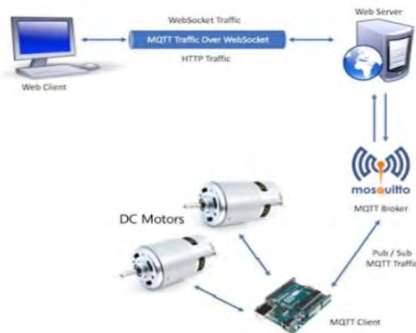
показващ процента заряд на батерията. При реализацията на визията и конструкцията на многоцелевия робот формите на пластиката са реализирани по начин, който да гарантира максимална възможна защитеност на компонентите, поместени във вътрешността, от твърди и прахови частици, което да подсури правилното му функциониране.

ГЛАВА III: УПРАВЛЕНИЕ НА МНОГОЦЕЛЕВИЯ СЕРВИЗЕН РОБОТ

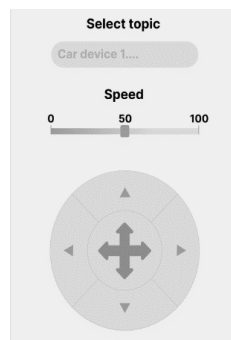
3.1 Теле-управление на многофункционална мобилна роботизирана платформа чрез уеб базиран графичен интерфейс, посредством MQTT и уеб сокети

3.1.4 Реализация на MQTT през WebSocket в JavaScript

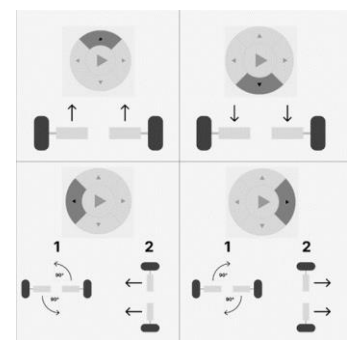
В тази разработка е представен интерфейс за управление на мобилен робот с две колела с постоянен ток двигател с помощта на MQTT.js и WebSockets. [45][70] Интерфейсът включва текстово поле за избор на тема, плъзгач за управление на скоростта и четири бутона за управление на посоката. Темата, избрана от потребителя, е каналът, чрез който интерфейсът публикува стойностите за скоростта и движението.



Фиг.36 Цялостният процес на комуникация между елементите



Фиг.37 Разработеният потребителски интерфейс



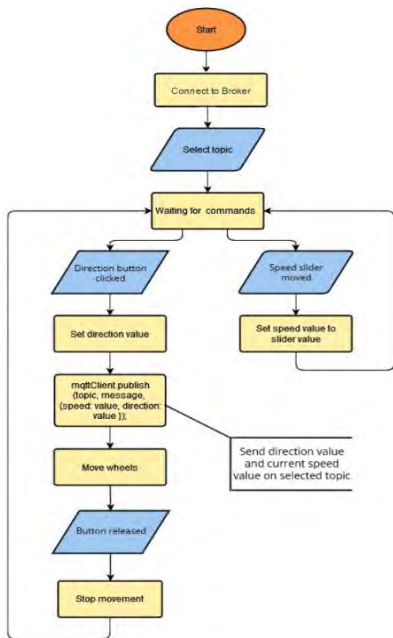
Фиг.38 Движение на колелата

Скоростта на движение се задава чрез събитието "Промяна" на JavaScript, което взема стойността от полето на плъзгача и задейства POST заявка, насочена към маршрута "издател". Изпраща се JSON обект, който съдържа името на темата и съобщение, което е целочислена стойност, определяща скоростта. По същия начин посоката на движение се определя от един от четирите бутона, всеки от които е обозначен с различна стрелка, указваща желаната посока. Всеки бутон съдържа команда с шаблон за необходимото движение на двигателя в дадената посока. Когато се натисне един от бутоните, при събитието "MouseDown" се изпраща POST заявка към избраната тема. След като бутонът бъде освободен се задейства събитието "MouseUp", което изпраща команда за спиране на двигателите и в отговор устройството спира да се движи.

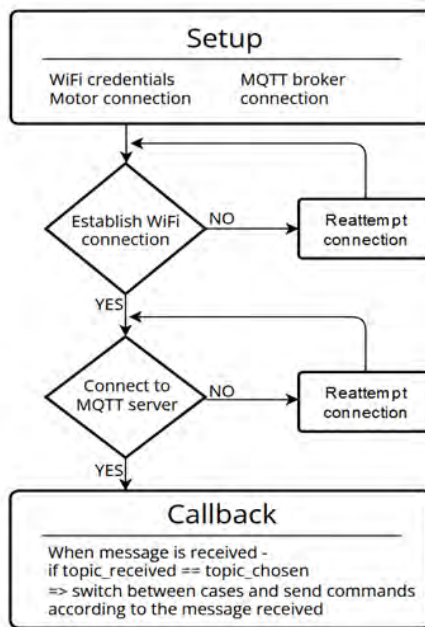
Фигура 37 илюстрира създадения потребителски интерфейс за програмата. Интерфейсът позволява на потребителите да управляват скоростта и посоката на робота. Темата, избрана от потребителя, се използва за публикуване на стойностите за скоростта и движението. Роботът се абонира за а темата и получава стойностите, изпратени от интерфейса. Като протокол за изпращане на съобщения се използва MQTT, а WebSockets осигурява двупосочен канал за комуникация между интерфейса и робота. Фигура 38 илюстрира как колелата на двигателя се движат в зависимост от натиснатия бутон. Натискането на бутона "нагоре" кара двете колела да се въртят напред, в резултат на което роботът се движи напред. Натискането на бутона "надолу" кара двете колела да се въртят в обратна посока, в резултат на което роботът се движи назад. Натискането на бутона "наляво" кара дясното колело да се върти напред, докато лявото колело се върти назад, в резултат на което роботът завива наляво. И обратното, натискането на бутона "дясно" кара лявото колело да се върти напред, докато дясното се върти назад, в резултат на което роботът завива надясно [44].

Блок-схемата (фиг.39) илюстрира основните стъпки, свързани с управлението на мобилен робот

чрез предоставения интерфейс, включително задаване на тема, регулиране на скоростта, управление на посоката на движение, публикуване на стойности и действие върху тези стойности за управление на колелата на робота. Както е показано на фигура 40, след стартирането на робота контролерът автоматично установява връзка с MQTT брокера. След като се свърже, когато се получи съобщение за абонирана тема от брокера, се изпълнява определената функция Callback. В резултат на получените входни данни се изпраща команда, за да се изпълни движението на двигателя, както е необходимо.



Фиг.39 Работен поток на процеса



Фиг.40 Диаграма на фърмуера

Действия като свързване с брокер, абониране за тема и публикуване на съобщения се записват и показват в конзола. На фигура 41 са показани дневниците за абониране за определена тема, натискане на бутона "Forward" (Предай напред) и пускането му след няколко секунди. Показва се и времето на действието.

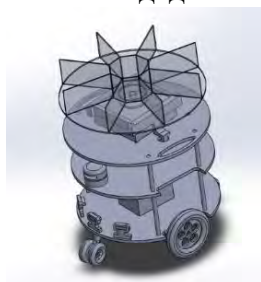
```

Console:
14:34:24: Connected with MQTT Broker: "ws://192.168.41.129:9001/mqtt"
14:34:41: Subscribed to MQTT Topic: "home/car1"
14:35:07: Published "FORWARD, 50" to MQTT Topic: "home/car1"
14:35:07: MQTT Message Received. Message: "FORWARD, 50" MQTT Topic:
"home/car1" QoS Value: "0"
14:35:21: Published "STOP, 0" to MQTT Topic: "home/car1"
14:35:21: MQTT Message Received. Message: "STOP, 0" MQTT Topic:
"home/car1" QoS Value: "0"
    
```

Фиг. 41. Регистър на конзолата

3.2 Управление на многоцелевия робот в режим за доставяне на лекарства

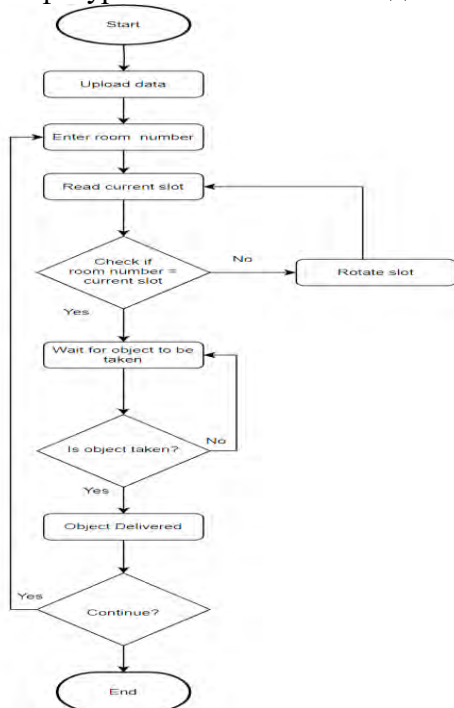
Ротационната маса е модул, който може да се монтира към мобилната робот-платформа с теле-управление (фиг. 42) и представлява важен компонент за дистрибуция на предмети като лекарства за различни пациенти съотнесени към дадената болнична стая.



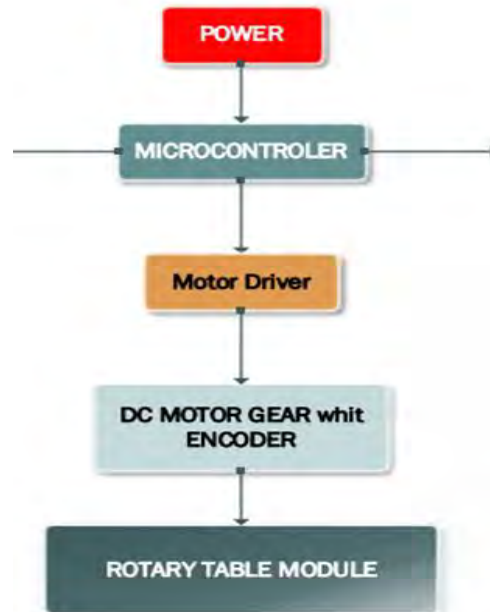
Фиг.42 Изглед на ротационния модул, скрепен към мобилната робот-платформа

3.2.1 Алгоритъм на работа на работа

Разработеният алгоритъм на работа на работа в режим доставяне на лекарства е представен нагледно на фигура 43 и включва следната стъпки:



Фиг.43 Блок схема за управлението на модула



Фиг.44 Схема на основните компоненти, изграждащи дистрибуторния модул

- Инсталация и стартиране на модула:

След успешната инсталация на ротационната маса към мобилната робот-платформа и включването на захранването, програмата за управление на модула се стартира.

- Зареждане на предмети:

Операторът поставя лекарствата или другите предмети, които трябва да бъдат пренесени и разпределени, в отделните слотове на ротационната маса.

- Избор на режим:

Интерфейсът на програмата предоставя възможност за избор на различни режими на работа, в зависимост от целите на операцията. Например, операторът може да избере автоматичен режим за разпределение на предмети на базата на предварително зададени параметри или ръчен режим, където операторът има контрол върху процеса.

- Стартиране на работния цикъл:

След като операторът избере желанния режим, той стартира работния цикъл чрез натискане на съответния бутон в програмата.

- Ротация и разпределение:

Начинът на работа на ротационната маса е да се върти, докато избраните предмети в дадения слот достигнат пред робота. На база на енкодер се знае за позицията на дадения слот. Посредством интерфейсът се запазва в кой слот обектите за коя стая и кой пациент са предназначени.

- Подаване на предметите:

Роботът се придвижва към пациента или мястото, където трябва да бъдат предоставени предметите. Ротационната маса подава предметите. След като пациентът вземе хапчетата си, посредством камерата на теле-управляемия робот, операторът валидира, че хапчетата са взети.

- Повторение на процеса:

Ако има повече слотове за дадения пациент след предаване на предмета, ротационната маса продължава да се върти и подава следващия предмет, докато не бъдат разпределени всички предмети зачислени към дадената стая и пациент.

- Завършване на работния цикъл:

След като всички предмети бъдат успешно разпределени, програмата извежда съобщение за завършване на операцията. Операторът задава предвижването на робота към следващото място на база на полу-автономно теле-управление или директно го навигира посредством директно теле-управление

- Изключване на модула:

Операторът може да изключи модула след успешното приключване на работния цикъл.

3.2.2. Симулация и резултати от верифицирането на модула и алгоритъма на работа

Симулационната постановка има за цел да верифицира управлението и поведението на модула. Симулацията е извършена на базата на няколко софтуера, посредством тяхното паралелно и синхронизирано изпълнение. В симулацията са обвързани следните софтуери:

- SolidWorks-CAD: Софтуер, използван за конструиране на робот-платформата и модула
- Proteus 8.16: Софтуер за електронно проектиране и автоматизация (EDA/ ECAD). Proteus позволява създаване на схеми, проектиране на печатни платки и др. Използван е в настоящата симулация, тъй като предоставя мощна функционалност при симулиране на микроконтролери посредством прилагане на hex файлове към вече изградени схеми.
- Arduino IDE: Среда, използвана за написването на кода, онагледяващ алгоритъма
- NHD Software's Virtual Serial Port Tools: Представява цялостен набор от софтуерни инструменти, предназначени за управление и емуляция на виртуални серийни портове. Посредством инструментите за виртуален серийен порт може да се създават виртуални серийни устройства и връзки, които напълно подражават на функционалността на хардуерните физически устройства.
- Blender 3D: Софтуер за 3D моделиране, който позволява писане на скриптове посредством Python API. Предоставя възможността за комуникиране с Proteus.

Първа стъпка е създаване на кода в Arduino IDE. За да може да се осъществи симулацията в кода се въвежда BlendixSerial.h библиотеката, която позволява комуникацията между симулацията на хардуера в Proteus с 3D средата на Blender 3D, в която е поставен конструирания робот.

```
#include <BlendixSerial.h>
#include <Servo.h>

BlendixSerial blendix;
Servo myservo; // create servo object to control a servo
int timeOn = 50;
int timeOff = 50;
String servoPrompt = "Enter sequence of servo degrees (1-8): ";
bool autonomousMode = false;

void setup() {
  Serial.begin(9600);
  myservo.attach(10);
}

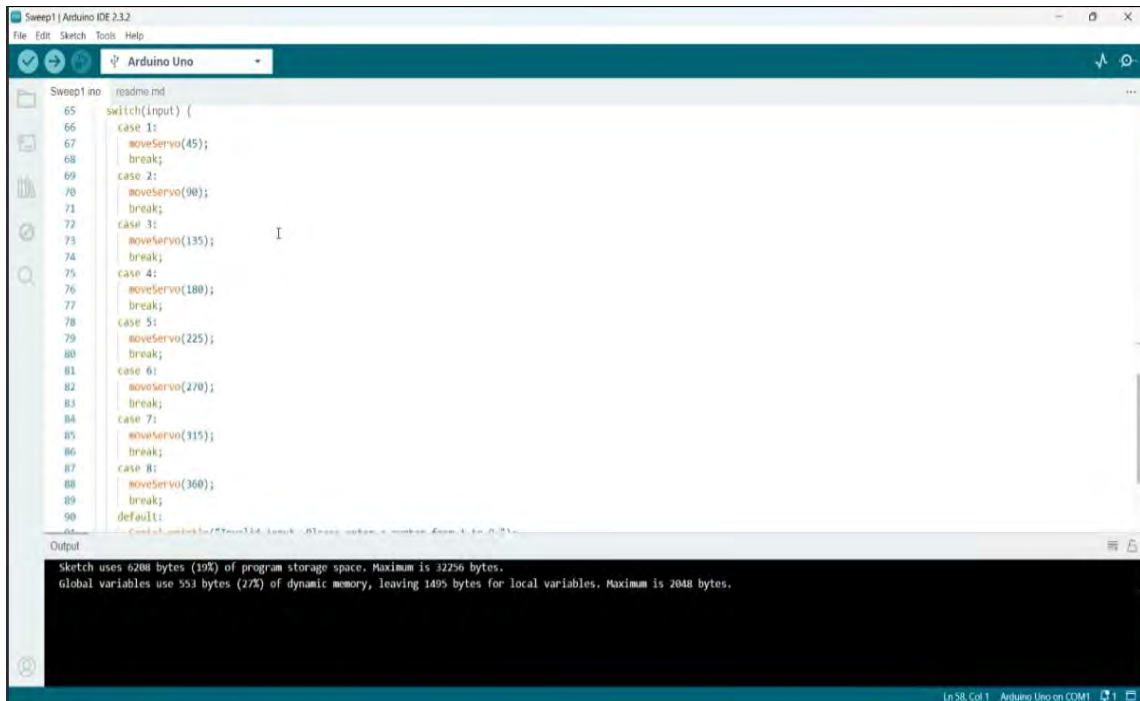
void loop() {
  Serial.println("Enter 1 for Autonomous mode or 0 for Manual mode: ");
  while (Serial.available() == 0) {}
}

int modeInput = Serial.parseInt();

if (modeInput == 1) {
  autonomousMode = true;
  autonomousloop();
} else if (modeInput == 0) {
```

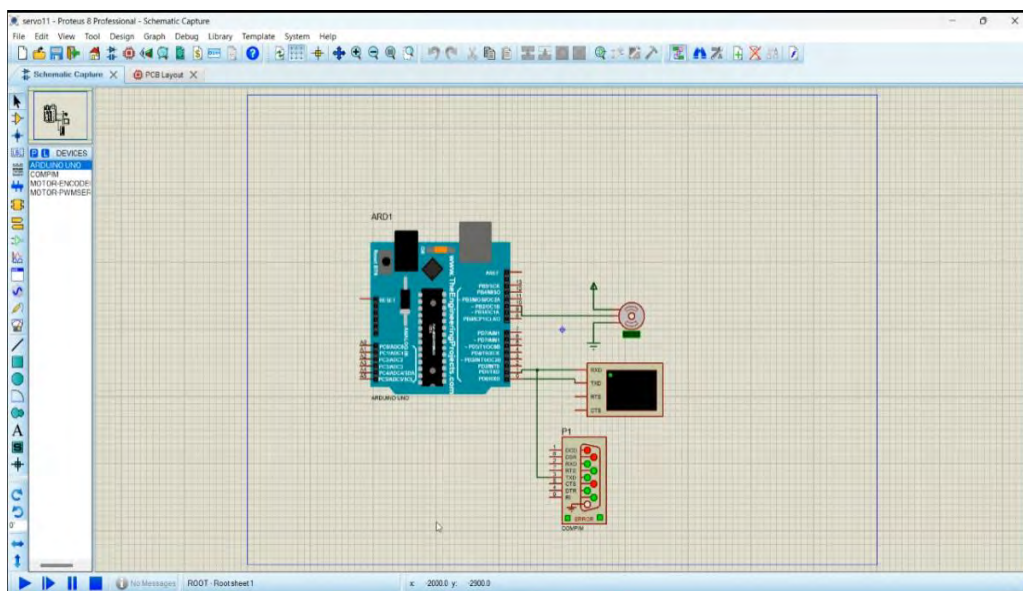
Фиг. 45 Въведената BlendixSerial.h библиотека в създадения в Arduino IDE код

За ръчен режим във виртуалния терминал на Proteus се въвежда 0, а за автономен режим при дистрибуция се въвежда 1. Всеки от слотове, които се зареждат от оператора при дистрибуция на лекарства отговаря на конкретен ъгъл на база на броя на слотовете, които са на ротационния диск. Фигура 46 онагледява стойността на ъгъла, кореспондиращ с конкретния слот.



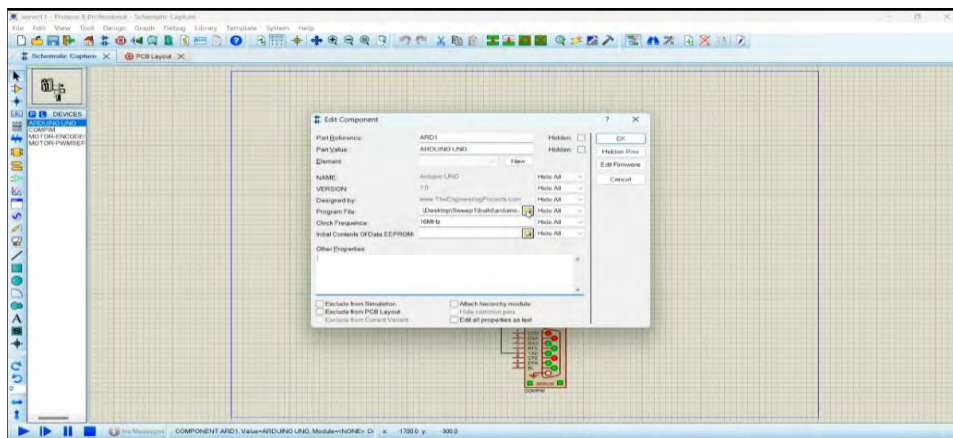
Фиг.46 Стойности на ъгъла на слотовете

Втора стъпка след създаването на програмата в средата на Arduino IDE е изграждането на хардуерната система и нейното окабеляване в средата на Proteus. За тази цел се използват библиотеките на Proteus, в които има готови компоненти - мотори, сензори, контролери и т.н. На фигура 47 е показана създадената хардуерна схема, свързваща компонентите, изграждащи модула както и виртуален терминал за въвеждане и визуализиране на данните.



Фиг.47 Хардуерна схема на модула за дистрибуция на лекарства

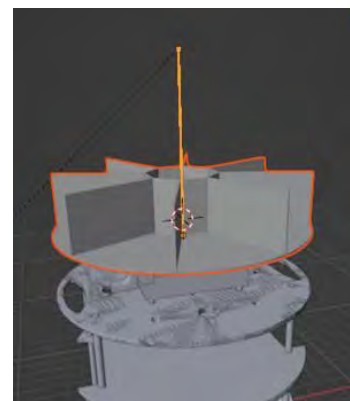
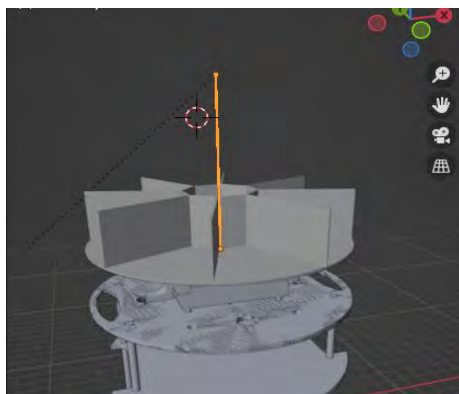
След като всички компоненти са свързани правилно и хардуерната система, изграждаща модула е готова следващата стъпка е да се зареди програмата написана в Arduino IDE в средата на Proteus, където извършваме симулацията. Зареждането на програмата става като в средата на Proteus върху микроконтролера се натисне с десен бутон. В резултат излиза менюто Edit Component, през което зареждаме ардуино файла.



Фигура 48: Зареждане на програмата написана в Arduino IDE в средата на Proteus

Следваща стъпка е отварянето на 3D модела на робота в 3D средата на Blender. За да може да се движи модела по определения желан от нас начин конструираните части в Solidworks трябва да бъдат експортирани отделно и асемблирани на ново в средата на Blender 3D където се задават необходимите ограничения към 3D обектите и се ригват обектите, които ще бъдат управлявани от програмата и ще се движат по определения начин. Стъпките на ригване, за да може модела да се движи по определена траектория на база на управляващата програма, която е имплементирана в симулацията Proteus са следните:

В обектен режим се създава „арматура“(скелетна структура, която управлява даден 3D обект).

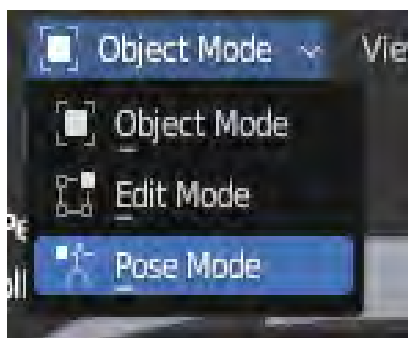


Фиг. 49 Меню за създаване на скелетна структура

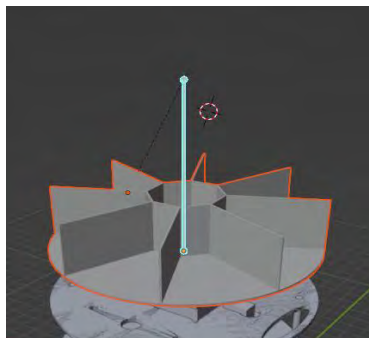
Фиг.50 Генерираната арматура за управление на 3D обекта

Фиг.51 Позициониране на арматурата в 3D обекта

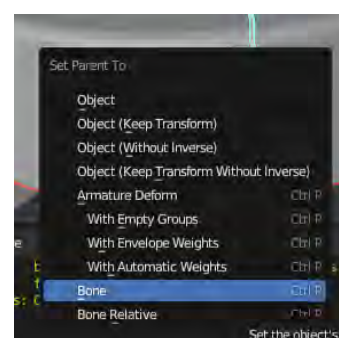
След като се създаде арматурата, тя се премества на мястото, на което искаме да се осъществи движението и се поставя към цялостната конструкция на модела, който желаем да бъде ригнат (фигура 50 и фигура 51). В обектен режим, след като се избере желания модел, арматурата се превключва в режим на поза (фигура 52). Следва избор на кост, която да бъде ригната, т.е. скрепена към модела (фигура53). И накрая се натиска ctrl + p и се избира "кост"(фигура 54)



Фиг.52 Избор на меню поза

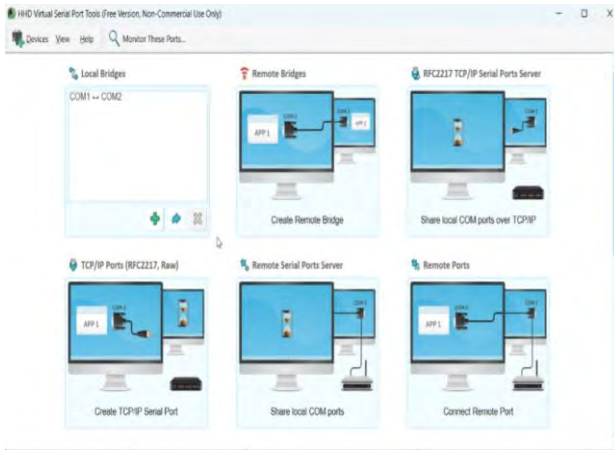


Фиг.53 Избор на кост, която да бъде ригната

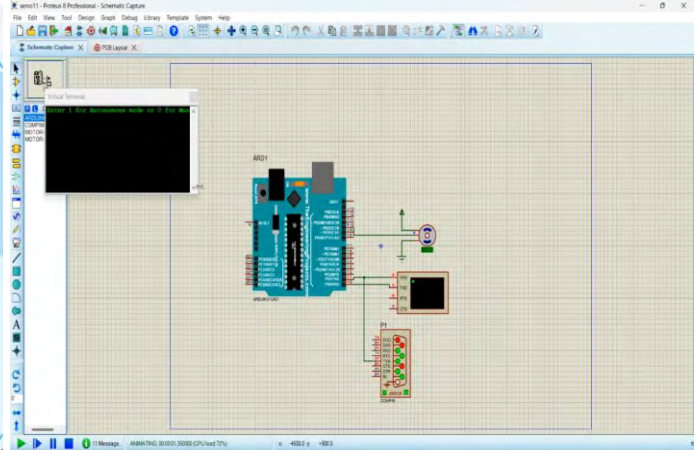


Фиг.54 Меню за избор на „кост“

След ригването на модела добавяме приложението към Blender BlendixSerial. Библиотеката BlendixSerial е проектирана да работи заедно с добавката blendixserial в Blender. То позволява контролиране на обекти в Blender чрез серийна комуникация с Arduino или други микроконтролери. Чрез емулирането на COM портовете от HHD Software's Virtual Serial Port Tools заедно с BlendixSerial започва осъществяването на връзката между симулацията в Proteus и 3D конструирания робот с прикачен модул в Blender

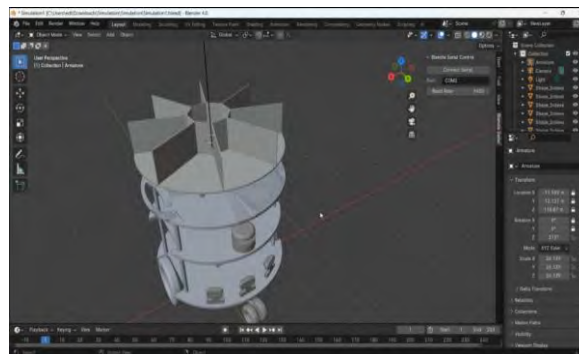


Фиг.55 Емулирането на COM портовете



Фиг.56 Стартиране на симулацията

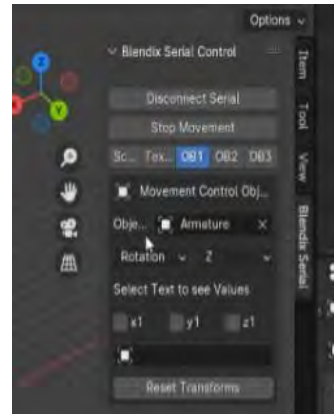
Стартира се симулацията в Proteus. (фигура 56). Посредством виртуалният терминал се визуализира стартираната програма, която е написана в Arduino и имплементирана в Proteus чрез генерирания hex файл. Терминалът изчаква да се въведе 0 или 1 за ръчен или автономен режим на дистрибуция. За да се визуализират промените в 3D модела на конструирания робот в падащото меню на BlendixSerial задължително трябва да сме натиснали Connect Serial и Start Movement, както и да сме посочили емулирания порт, което е онагледено на фигури 57, 58 и 59.



Фиг.57 Избор на COM порт в Blender

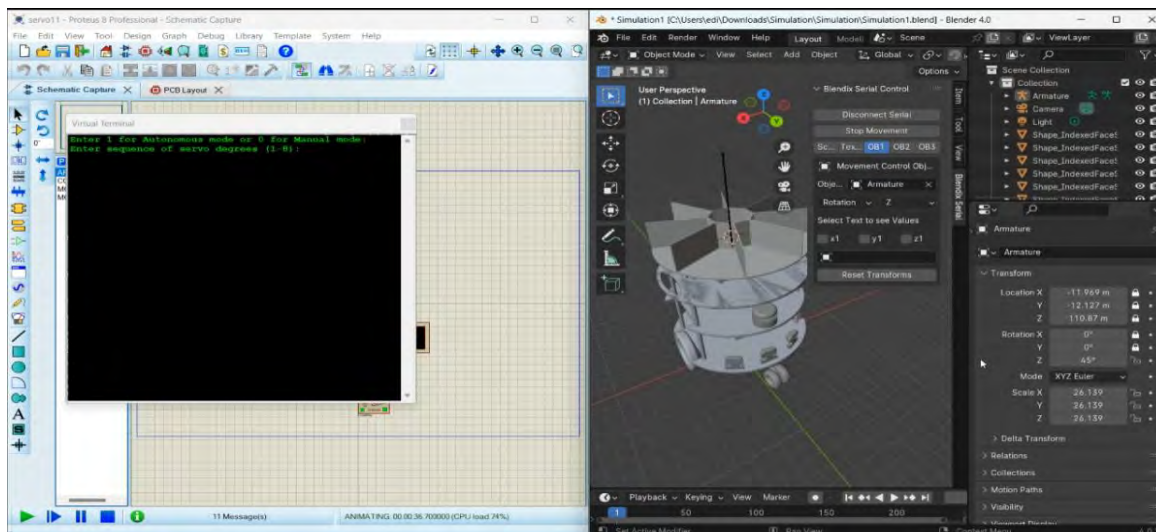


Фиг.58 BlendixSerial преди стартиране



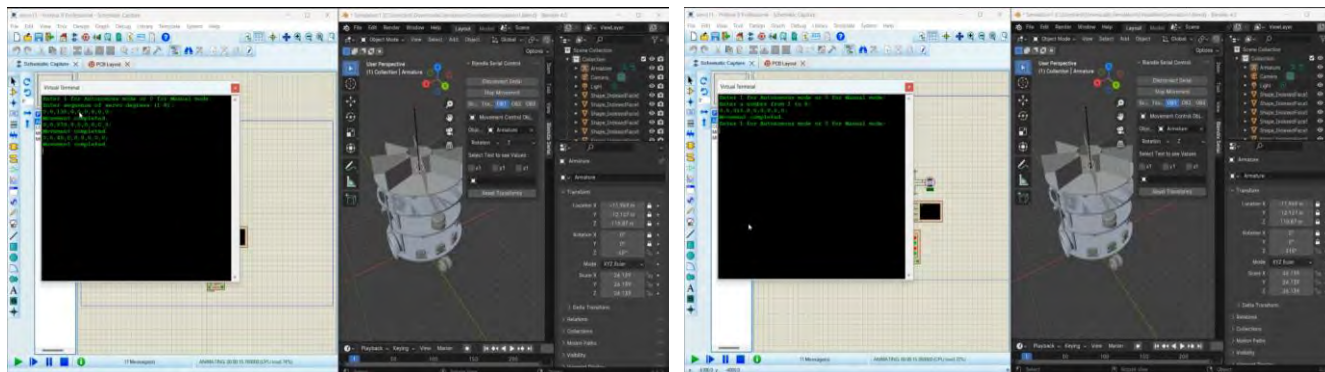
Фиг.59 BlendixSerial след стартиране

След като въведем 1 и стартираме автономното разпределяне програмата изчаква въвеждането на последователно от слотове със заредени в тях лекарства, които са за пациента, към който оператора изпраща робота. (онагледено на фигурата 60).



Фиг.60 Синхронизираните среди на Proteus и Blender

На фигурата 61 е показан автономният режим на дистрибуция за 3 слота. След като сме въвели номера на слотовете, в които сме сложили лекарствата, програмата ги изпълнява последователно. На виртуалния терминал е показано последователното изпълнение на 3ти, 6ти и 1ви слот като след всяко завъртане до съответния слот терминала изписва, че движението е успешно/завършено. 45 градуса отговаря на първи слот и може да бъде видян както на терминала от лявата част на скрийншота така и отдясната страна в 3D средата на блендър в отключената по z ротация на обекта.



Фиг.61 Автономен режим на дистрибуция

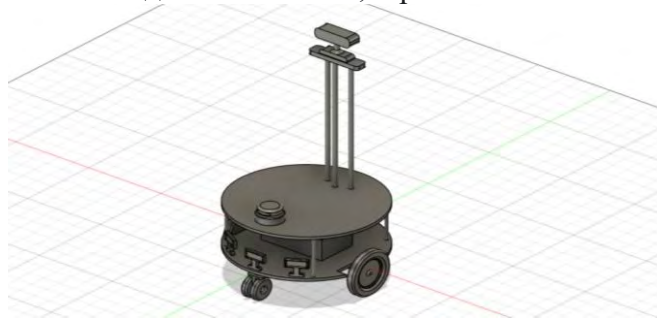
Фигура 62: Ръчен режим дистрибуция

В ръчен режим (който се активира при въвеждане на 0 в терминала) е въведен слота, който искаме да завъртим в предната страна на робота, за да може пациента да има достъп до него. На изображението фиг. е въведен слот 7, който отговаря на 315 градуса, които са визуализирани във виртуалния терминал и в менюто, отговарящо за ротация на дистрибуторната маса в 3D средата на блендър. След като движението е успешно завършено терминалът очаква следващото въвеждане за режим и след това за номер на слота.

3.3 Управление на многоцелевия робот в режим на обход, наблюдение, охрана и сигнализация

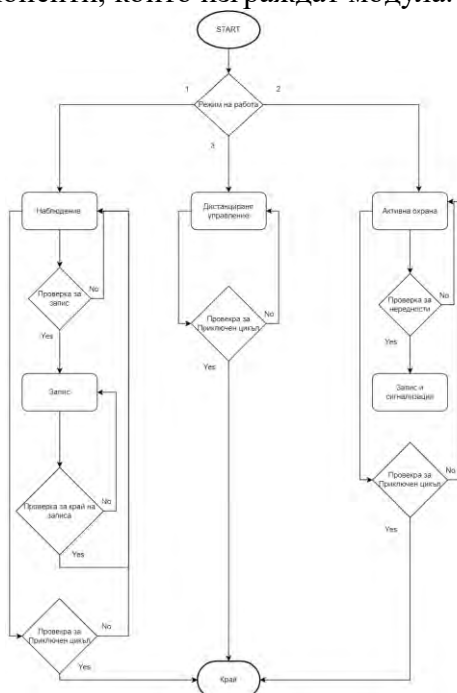
Модулът за наблюдение и охрана е важен компонент на мобилната робот-платформа с теле-управление, който предоставя възможности на оператора да избира между режим на

наблюдение, режим на запис на видео или снимки, и режим на активна охрана.

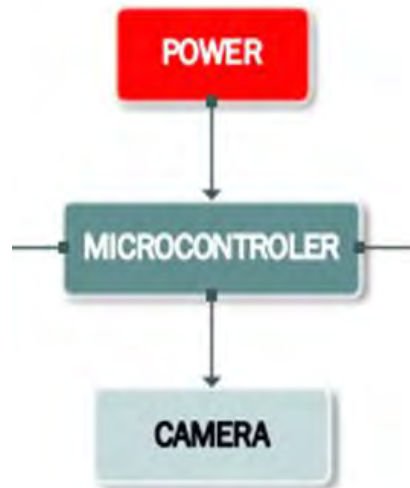


Фиг. 63 Мобилна робот-платформа с монтиран модул за наблюдение и охрана

Разработеният алгоритъм на работа на робота в режим на обход, наблюдение, охрана и сигнализация е представен нагледно на фигура 64, а на фигура 65 са представени основните компоненти, които изграждат модула.



Фиг.64 Блок схема за управлението на модула



Фиг.65 Схема на основните компоненти изграждащи модула за наблюдение

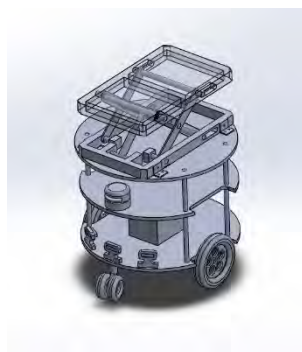
Алгоритъмът на работа включва следната стъпки:

- **Инсталация и стартиране на модула:**
Модулът за наблюдение и охрана съдържа вградена камера, която е монтирана на най-горната плоча на мобилната робот платформа. След успешната инсталация на модула и включването на захранването, програмата за управление на модула се стартира.
- **Избор на режим:**
Интерфейсът на програмата предоставя възможност за избор на различни режими на работа. Операторът може да избере между режим на наблюдение, режим на запис на видео или снимки, и режим на активна охрана.
- **Стартиране на работния цикъл:**
След като операторът избере желанния режим, той стартира работния цикъл чрез натискане на съответния бутон в програмата.
- **Наблюдение и запис:**
Камерата на модула за наблюдение и охрана започва да излъчва видео сигнал, който може да бъде наблюдаван в реално време от оператора. В режим на запис, програмата може да започне запис на видео или снимки, които се запазват за бъдеща употреба.

- **Дистанцирано управление:**
Операторът може да управлява наклона и завъртането на камерата чрез интерфейса на програмата. Това позволява на оператора да насочи камерата към конкретни обекти или области.
- **Активна охрана:**
В режим на активна охрана, роботът може да използва допълнителни сензори или алгоритми за детекция на движение или неправомерни дейности в зоната на наблюдение. При засичане на неправомерни дейности, модулът може да предостави предупреждение на оператора и да започне запис на инцидента.
- **Завършване на работния цикъл:**
След като операторът приключи с наблюдението или записа се извежда съобщение за завършване на операцията.
- **Изключване на модула:**
Операторът може да изключи модула след успешното приключване на работния цикъл. Този модул предоставя възможности за наблюдение и охрана на различни места и ситуации, като предоставя реално време на оператора или записва важни данни за бъдеща употреба.

3.4 Управление на многоцелевия робот в режим за транспортиране и издигане на товари чрез ножичест подежник

Общият вид на робота в режим за транспортиране и издигане посредством ножичест подежник е показан на фигура 66.

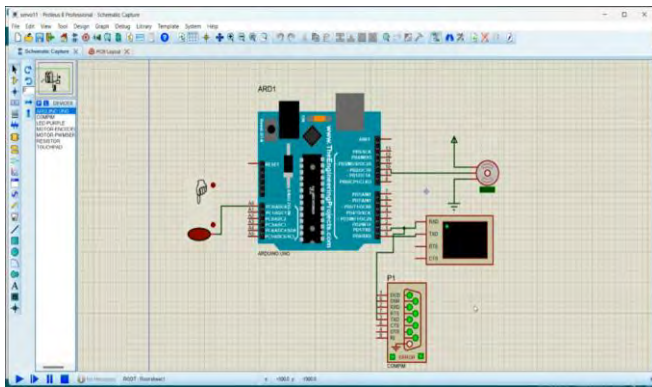


Фиг.66 Общ вид на робота с монтиран към него модул с ножичест подежник

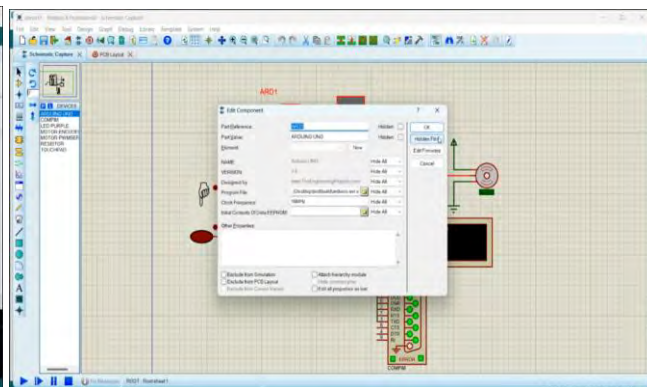
3.4.1 Алгоритъм на работа на робота

Разработеният алгоритъм на работа е представен на фигура 67 и включва следните стъпки:

- Започва се с инсталация и стартиране на модула. След като операторът приведе платформата в работен режим и включи захранването, програмата се стартира и контролерът подава сигнал към драйвера на линейния мотор (актуатора).
- Ако повдигащата платформа не е на стартовата си позиция линейния двигател започва да сваля рамената, върху които е прикрепена повдигащата платформа. След извършване на тази процедура платформата се намира в режим „очакване“, до подаване на команда от оператора. Тази процедура се налага да се извърши и в случай, че платформата е прекъснала работата си аварийно и повдигащата платформа е останала в работно положение.
- Когато роботът получи команда „повдигни товара“, контролера подава сигнал към драйвера на линейния мотор. Моторът започва да повдига рамената, върху които е монтирана повдигащата платформа, докато датчика за тежест не докосне товара. Той отчита тежестта на товара и ако той надвишава нормата, програмата отчита грешка „Внимание, привишено тегло!“ Тогава механизмът връща повдигащата платформа в режим „очакване“.
- Когато програмата получи команда „повдигни товара“ и датчика за тежест отчете тежест в нормата, механизмът повдига товара до зададената височина и роботът може да пристъпи към транспортиране на товара.



Фиг. 70 Хардуерна схема в Proteus



Фиг. 71 Зареждане на програмата написана, в Arduino IDE в средата на Proteus

Втора стъпка след създаването на програмата в средата на Arduino IDE е изграждането на хардуерната система и нейното окабеляване в средата на Proteus. За тази цел се използват библиотеките на Proteus, в които има готови компоненти - мотори, сензори, контролери и т.н. На фигура 70 е показана създадената хардуерна схема, свързваща компонентите, изграждащи модула както и виртуален терминал за въвеждане и визуализиране на данните. След като всички компоненти са свързани правилно и хардуерната система, изграждаща модула е готова следващата стъпка е да се зареди програмата написана в Arduino IDE в средата на Proteus, където извършваме симулацията. (фигура 71). Зареждането на програмата става като в средата на Proteus върху микроконтролера се натисне с десен бутон. В резултат излиза менюто Edit Component, през което зареждаме Arduino файла. Следваща стъпка е отварянето на 3D модела на робота в 3D средата на Blender. За да може да се движи модела по определения желан от нас начин конструирания части в Solidworks трябва да бъдат експортирани отделно и асемблирани на ново в средата на Blender 3D където се задават необходимите ограничения към 3D обектите и се ригват обектите, които ще бъдат управлявани от програмата и ще се движат по определен начин. Стъпките на ригване, за да може модела да се движи по определена траектория на база на управляващата програма, която е имплементирана в симулацията Proteus са следните:

В обектен режим се създава „арматура“ (скелетна структура, която управлява даден 3D обект). След като се създаде арматурата, тя се премества на мястото, на което искаме да се осъществи движението и се поставя към цялостната конструкция на модела, който желаем да бъде ригнат. (фигура 72, фигура 73 и фигура 74).



Фиг. 72 Меню за създаване на скелетна структура

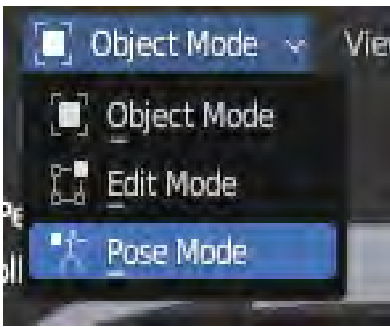


Фиг. 73 Генерираната арматура за управление на 3D обекта



Фиг. 74 Позициониране на арматурата в 3D обекта

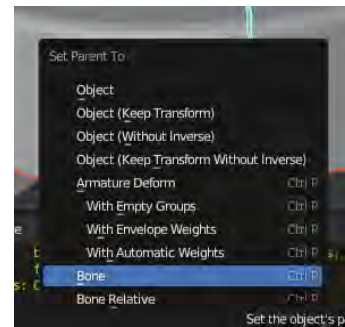
В обектен режим, след като се избере желания модел, арматурата се превключва в режим на поза (фигура 75). Следва избор на кост, която да бъде ригната, т.е. скрепена към модела (фигура 76). И накрая се натиска ctrl + p и се избира "кост" (фигура 77)



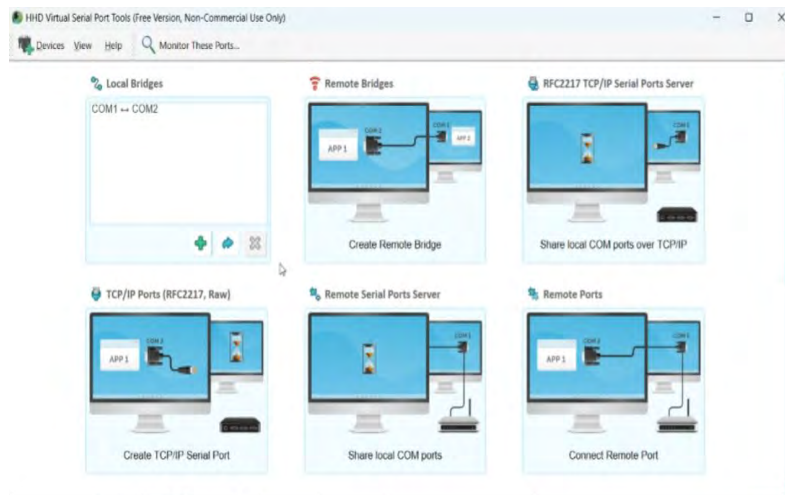
Фиг.75 Избор на меню поза



Фиг.76 Избор на кост, която да бъде ригната



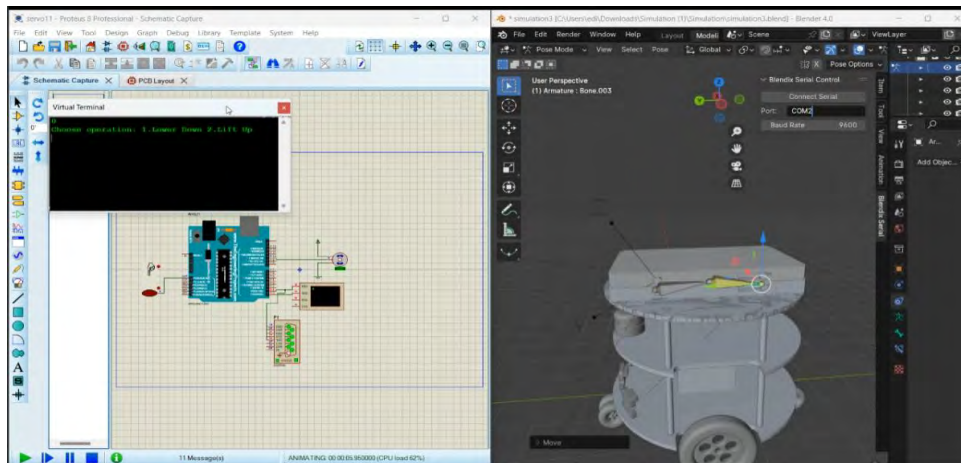
Фиг.77 Меню за избор на „кост“



Фиг.78 Емулирането на COM портовете

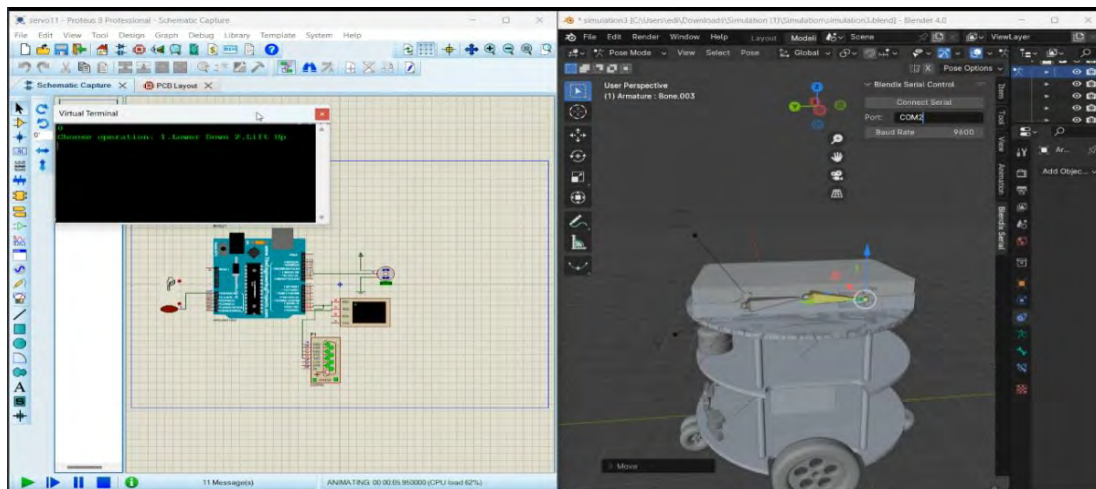
След ригването на модела добавяме приложението към Blender BlendixSerial. Библиотеката BlendixSerial е проектирана да работи заедно с добавката blendixserial в Blender. То позволява контролиране на обекти в Blender чрез серийна комуникация с Arduino или други микроконтролери. Чрез емулирането на COM портовете от NHD Software's Virtual Serial Port Tools заедно с BlendixSerial започва осъществяването на връзката между симулацията в Proteus и 3D конструирания робот с прикачен модул в Blender.

Стартираме симулацията в Proteus (фигура79). Посредством виртуалният терминал се визуализира стартираната програма, която е написана в Arduino и имплементирана в Proteus чрез генерирания hex файл. Терминалът изчаква да въведем 1 или 2 за сваляне или повдигане на платформата.

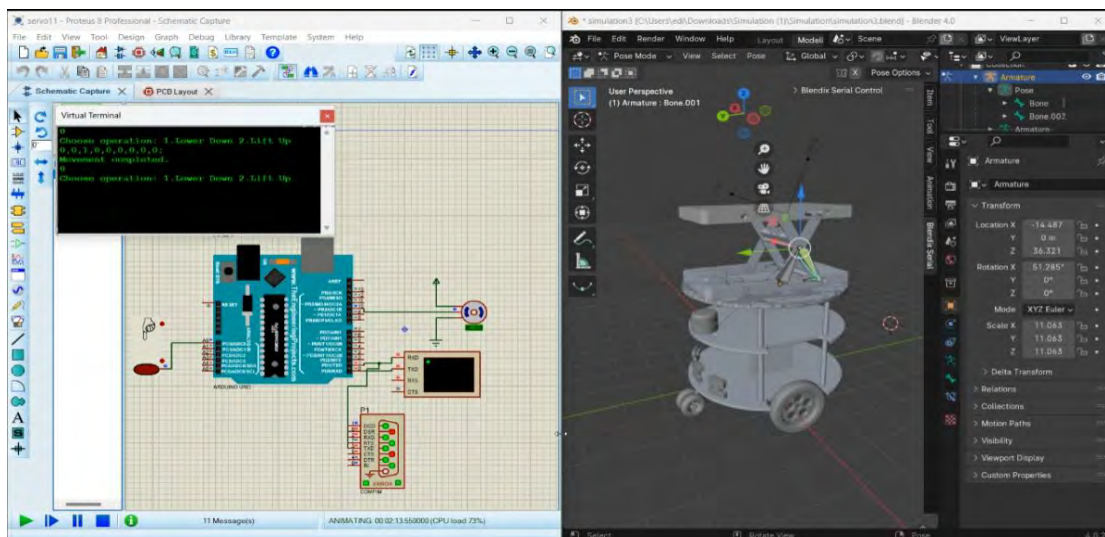


Фиг.79 Стартиране на симулацията в Proteus и избор на COM порт в Blender

За да се визуализират промените в 3D модела на конструирания робот в падащото меню на BlendixSerial задължително трябва да сме натиснали Connect Serial и Start Movement, както и да сме посочили емулирания порт.

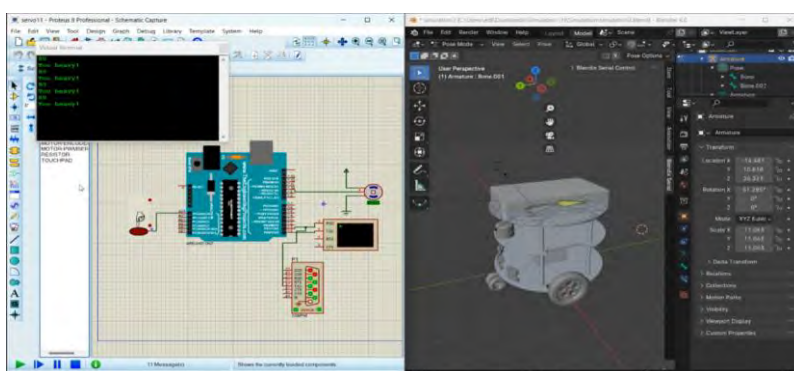


Фиг.82 Стартова позиция, изчакваща въвеждането в терминала



Фиг.83 Роботът с повдигнат ножсичест подежник

На фигурите 82 и фигура 83 са изобразени двете възможни позиции на робота, а именно свалена и повдигната платформа. При превишаване на допустимото тегло, потребителят получава предупреждение че товара поставен на платформата е прекалено много и програмата временно прекратява функцията си до отнемане на излишния товар – фигура 84.

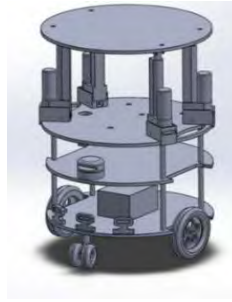


Фиг.84 Прекратена функцията на робота до отнемане на излишния товар

След като движението е успешно завършено терминалът очаква следващото въвеждане на команда при нужда от потребителя.

3.5 Управление на многоцелевия робот в режим за складиране и доставка посредством повдигаща платформа

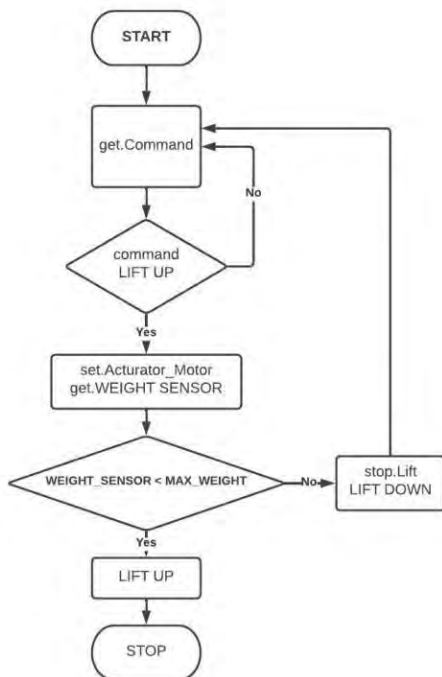
Общият вид на робота в режим в режим за складиране и/или доставка на готова продукция е показан на фигура 86.



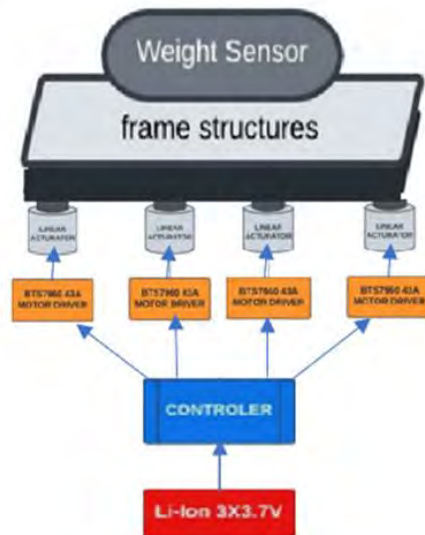
Фиг.86 Робот с монтиран модул с повдигаща платформа за доставка на готова продукция

3.5.1 Алгоритъм на работа на робота

Разработеният алгоритъм на работа на робота в режим за доставка на продукция посредством модул с повдигаща платформа е представен на фигура 87. Подобно на ножичестия подемник модулът с повдигаща платформа за складиране и доставка преминава през същите функционални стъпки.



Фиг.87 Блок схема за управлението на модула



Фиг.88 Схема на основните компоненти, изграждащи модула

На фигура 88 е представена хардуерната система на модула. Сервизният робот в режим за складиране и/или доставка на готова продукция включва диференциална мобилна платформа, индустриален компютър и контролери, електромеханична задвижваща система (постоянно токови двигатели с редуктори и енкодери), 12V LiPo акумулаторна батерия и сензорна система. Използваният линеен задвижващ механизъм е 12 волта с максимална товароносимост 90 кг.

3.5.2. Симулация и резултати от верифицирането на модула и алгоритъма на работа

Верифицирането на модула и алгоритъма на работа се осъществи чрез реализирането на

компютърна симулация, за която се извърши паралелно и синхронизираното прилагане на следните софтуери: SolidWorks-CAD, Proteus 8.16, Arduino IDE, HHD Software's Virtual Serial Port Tools и Blender 3D.

Първа стъпка е създаване на кода в Arduino IDE. За да може да се осъществи симулацията в кода се въвежда `BlendixSerial.h` библиотеката, която позволява комуникацията между симулацията на хардуера в Proteus с 3D средата на Blender 3D, в която е поставен конструираният робот.

При стартиране на симулационната част в Proteus във виртуалния терминал на симулацията оператора избира вида операция/движение на платформата/: 1- за сваляне 2- за вдигане.

В програмата при превишаване на лимита на тегло, което може да поеме платформата терминалът връща съобщение "Too heavy!" и не позволява изпълнението на операцията до намаляването на килограмите в допустимите стойности.



```

1 #include <Arduino.h> // using (a-b)
2
3 #include <BlendixSerial.h>
4 #include <Servo.h>
5
6
7 #include <BlendixSerial.h>
8 #include <Servo.h>
9
10 int timeout = 50;
11 int timeout2 = 50;
12
13 string message = "Close operation: 1,lower down 2, lift up ";
14
15
16 void setup() {
17   Serial.begin(9600);
18   servo1.attach(9);
19   servo2.attach(11);
20 }
21
22 void loop() {
23   int force_sensor_value = analogRead(A0);
24   int force_sensor_value_scaled = map(force_sensor_value, 0, 1023, 0, 1000);
25   Serial.println(force_sensor_value_scaled);
26   if (force_sensor_value_scaled > 700) {
27     message = "Too heavy!";
28   }
29 }

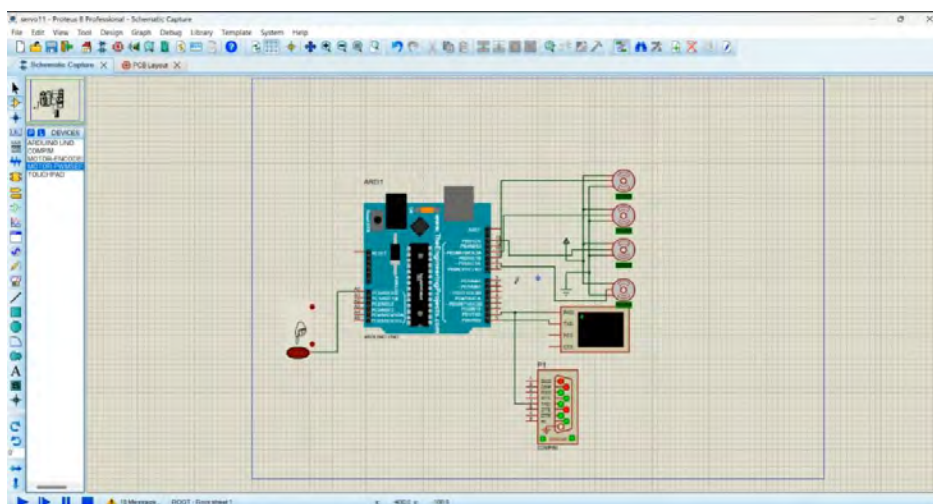
```

Output:

Sketch uses 6080 bytes (46%) of program storage space. Maximum is 13226 bytes.
Global variables use 404 bytes (10%) of dynamic memory, leaving 3644 bytes for local variables. Maximum is 3648 bytes.

Фиг.89 Създадената програма в средата на Arduino IDE

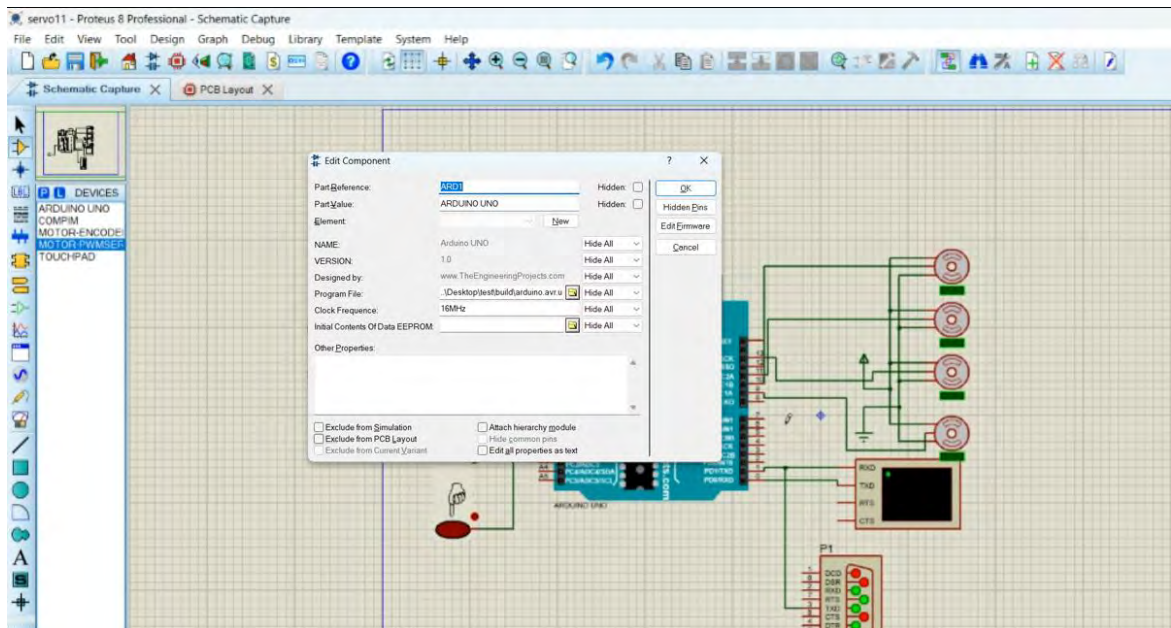
Втора стъпка след създаването на програмата в средата на Arduino IDE е изграждането на хардуерната система и нейното окабеляване в средата на Proteus. За тази цел се използват библиотеките на Proteus, в които има готови компоненти - мотори, сензори, контролери и т.н. На фигура 90 е показана създадената хардуерна схема, свързваща компонентите, изграждащи модула както и виртуален терминал за въвеждане и визуализиране на данните.



Фиг.90 Хардуерна схема

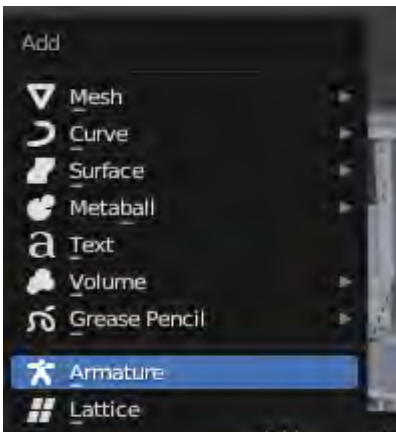
След като всички компоненти са свързани правилно и хардуерната система, изграждаща модула е готова следващата стъпка е да се зареди програмата написана в Arduino IDE в средата на Proteus, където извършваме симулацията. Зареждането на програмата става като в средата на Proteus върху

микроконтролера се натисне с десен бутон. В резултат излиза менюто Edit Component, през което зареждаме Arduino файла.



Фиг.91 Зареждане на Arduino програмата в средата на Proteus

Следваща стъпка е отварянето на 3D модела на робота в 3D средата на Blender. За да може да се движи моделът по определения желан от нас начин конструиранияте части в Solidworks трябва да бъдат експортирани отделно и асемблирани на ново в средата на Blender 3D където се задават необходимите ограничения към 3D обектите и се ригват обектите, които ще бъдат управлявани от програмата и ще се движат по определения начин. Стъпките на ригване, за да може моделът да се движи по определена траектория на база на управляващата програма, която е имплементирана в симулацията Proteus са следните:



Фиг.92 Меню за създаване на скелетна структура



Фиг.93 Генерираната арматура за управление на 3D обекта



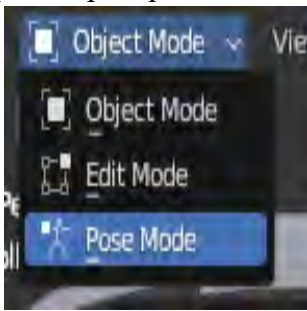
Фиг.94 Позициониране на арматурата в 3D обекта

В обектен режим се създава „арматура“. (фиг. 92). След като се създаде арматурата, тя се премества на мястото, на което искаме да се осъществи движението и се поставя към цялостната конструкция на модела, който желаем да бъде ригнат. (фиг. 93 и фиг.94).

В обектен режим, след като се избере желания модел, арматурата се превключва в режим на поза (фигура 95). Следва избор на кост, която да бъде ригната, т.е. скрепена към модела (фигура 96). И накрая се натиска ctrl + p и се избира "кост"(фигура 97)

След ригването на модела добавяме приложението към Blender BlendixSerial. Библиотеката

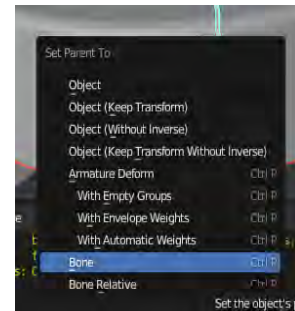
BlendixSerial е проектирана да работи заедно с добавката blendixserial в Blender. То позволява контролиране на обекти в Blender чрез серийна комуникация с Arduino или други микроконтролери.



Фиг.95 Избор на меню поза

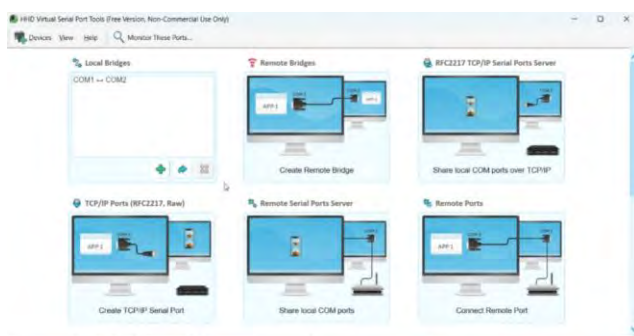


Фиг.96 Избор на кост, която да бъде ригната

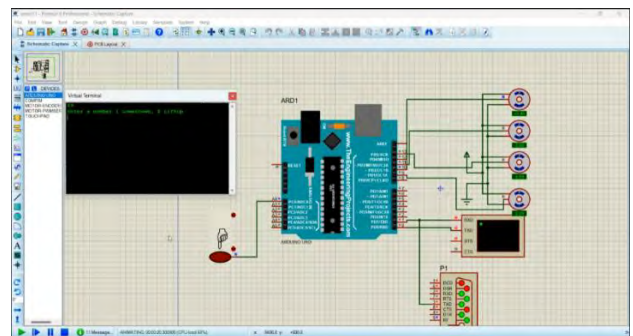


Фиг.97 Меню за избор на „кост“

Чрез емулирането на COM портовете от HND Software's Virtual Serial Port Tools заедно с BlendixSerial започва осъществяването на връзката между симулацията в Proteus и 3D конструирания робот с прикачен модул в Blender

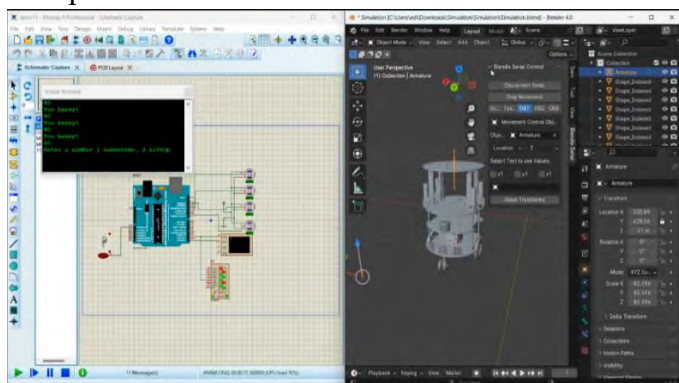


Фиг.98 Емулиране на COM портовете

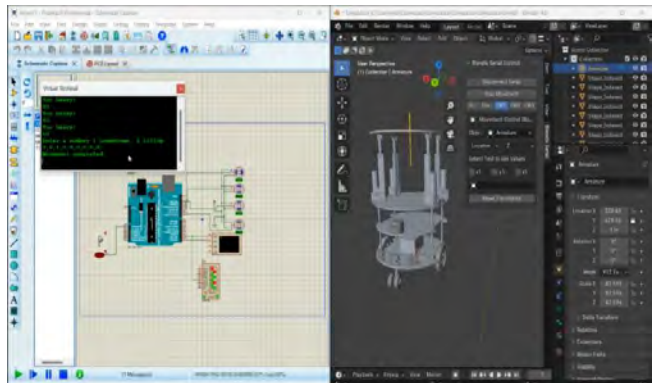


Фиг.99 Стартиране на симулацията от Proteus

Стартираме симулацията в Proteus. Посредством виртуалния терминал се визуализира стартираната програма, която е написана в Arduino и имплементирана в Proteus чрез генерирания hex файл. Терминала изчаква да въведем 1 или 2 за сваляне или издигане на повдигаща платформа. За да се визуализират промените в 3D модела на конструирания робот в падащото меню на BlendixSerial задължително трябва да сме натиснали Connect Serial и Start Movement, както и да сме посочили емулирания порт. Чрез въвеждане на „2“ във виртуален терминал позицията на повдигащата платформа се издига нагоре по Z axis (в случаите когато отчетената тежест е под 70% от допустимата). След като движението е успешно завършено терминалът очаква следващото въвеждане. На фигура 103 и фигура 104 е показано посредством визуализирането във виртуалния терминал на ситуация, в която тежестта е над допустимото и операцията не е допусната и следващата ситуация, в която тежестта е в зададените допустими стойности и движението е извършено.

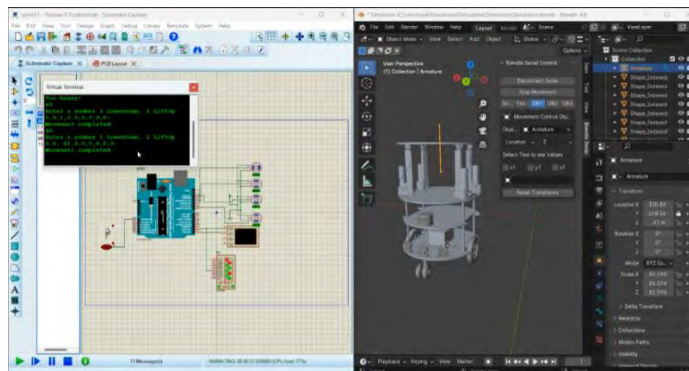


Фиг.103 Ситуация, в която тежестта е над допустимото и операцията не е допусната



Фиг.104 Движението е извършено- тежестта е в зададените допустими стойности

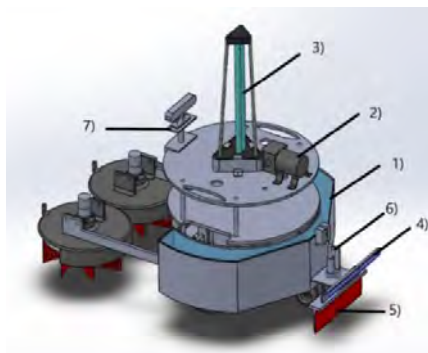
Чрез въвеждане на „1“ във виртуалния терминал позицията на повдигащата платформа се връща в първоначалната си позиция. След като движението е успешно завършено терминалът очаква следващото въвеждане (фиг. 105)



Фиг.105 Терминалът в очакване следващото въвеждане

3.6 Управление на многоцелевия робот в режим на почистване

Общият вид на робота със свързан към универсалната мобилна платформа модул за почистване е показан на фигура 107.



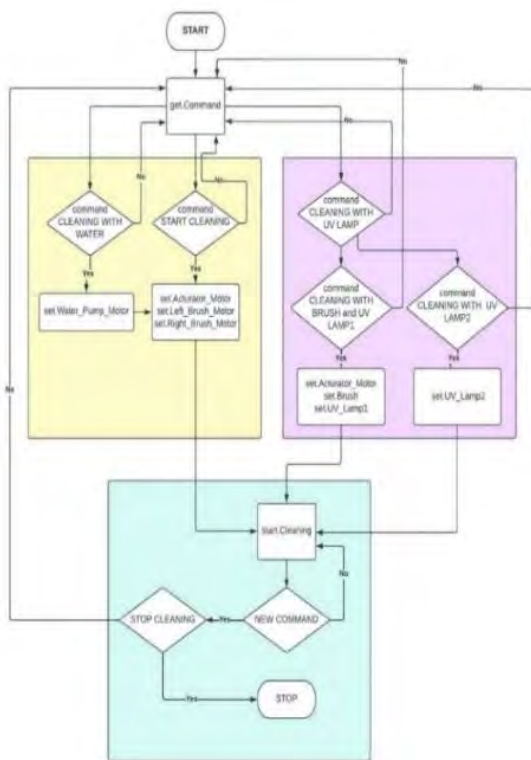
Фиг. 107: 1- резервоар за препарат с капак, 2- помпа за резервоар, 3- антибактериална UV лампа за хигиенизиране на околностите, 4- антибактериална UV лампа за хигиенизиране на пода, 5- задна регулируема четка, 6- линеен задвижващ механизъм за преместване на заден брус, 7- камера за телеоперация

3.6.1 Алгоритъм на работа на робота

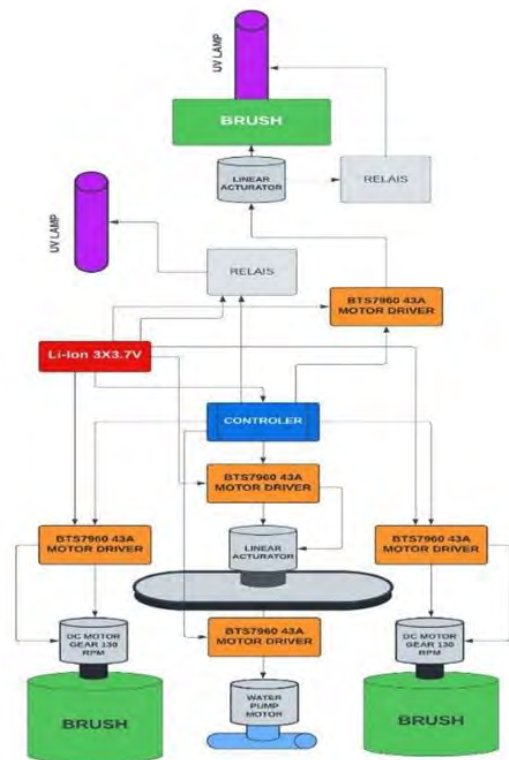
Разработеният алгоритъм на работа в режим на почистване, представен на фигура 108, включва следните стъпки:

- Започва се с инсталация и стартиране на модула. След като операторът е инсталирал почистващия модул и е въвел платформата в експлоатация (включване на захранването), програмата се стартира и контролерът изпраща сигнал до задвижващия механизъм. Моторът започва да повдига конструкцията, на която са прикрепени въртящите се четки, докато се достигне ъгълът, зададен от програмата. След извършване на тази процедура, платформата е в режим "в готовност", докато не бъде издадена команда от оператора. Тази процедура трябва да се извърши и в случай, че платформата е прекъснала работата си при аварийна ситуация и конструкцията, носеща въртящите се четки, е останала в работно положение.
- Когато програмата получи командата "стартиране на почистването", контролерът изпраща сигнал до драйвера [93] на двигателя. Задвижващият механизъм започва да спуска рамото, на което са монтирани въртящите се двигатели, докато четките докоснат работната повърхност. След като конструкцията е спусната, контролерът изпраща сигнал до драйверите на двигателите, задвижващи въртящите се четки. Те се задвижват, като достигат максималната си скорост за 5 секунди. Това предотвратява натоварване, което би се получило от внезапното стартиране на двигателите.

- Когато програмата получи командата "започнете почистването с вода", ако платформата е била в режим "изчакване", процедурата в стъпка 2 се повтаря и след като двигателите, задвижващи въртящите се четки, достигнат максималната си скорост, контролерът изпраща сигнал до драйвера на двигателя на помпата на резервоара.



Фиг.108 Блокова схема на алгоритъма за управление



Фиг.109 Хардуерна система на хигиенния модул

- Ако платформата е била в режим "стартиране на почистването", когато е била дадена командата "стартиране на почистването на водата", контролерът директно изпраща сигнал само на драйвера на двигателя, управляващ водната помпа.
- Когато програмата получи командата "започнете почистване на пода с UV лампа", контролерът изпраща сигнал до драйвера на линейния мотор. Моторът започва спуска рамото, към което е монтирана четка за почистване на пода и UV лампа. След като рамото се спусне на пода, се изпраща сигнал към релето, захранващо UV лампата.
- При издаване на командата "Започнете почистването на околната среда с UV лампа", контролерът изпраща сигнал към релейното захранване, контролиращо UV лампата, почистваща околната среда.
- При командата "край на почистването" контролерът подава последователно следните сигнали:
 - сигнал към водача на двигателя, задвижващ помпата на резервоара за спиране на работа,
 - сигнал към драйверите на двигателите, задвижващи въртящите се четки за спиране на работата,
 - сигнал към драйвера на линейния мотор да вдигне конструкцията до определената позиция, зададена от програмата,
 - сигнал към драйвера на линейния мотор за повдигане на задната четка и команда за изключване на UV лампата за хигиена на пода и централната UV лампа, почистваща околностите в стаята.

На фигура 109 е представена хардуерната система на почистващия модул. Почистващият модул се захранва от литиево-йонна батерия, състояща се от 3 серийно свързани клетки с напрежение 3.7V и максимален ток 5A.

3.6.2 Телеуправление на разработения модул за почистване

Управлението на почистващия модул е добавено към уеб-базирания графичен интерфейс

за телеоперация на робота. Връзката между робота и потребителя е чрез Wi-Fi. Потребителят може да използва мобилно устройство или лаптоп- единственото условие е да се свърже с мрежата на робота и да има уеб браузър. Телеуправлението се получава с помощта на технологията на уеб сокета. Те предават данни от робота към оператора и обратно.

Към интерфейса са добавени бутони и инструменти за управление на различните модули и операциите, които изпълняват. Въз основа на телеоперацията операторът задава създаването на карта и запазването на картата на текущата стая, а според ситуацията, която наблюдава в стаята, задава кои системи да се включат, когато роботът я почиства. Освен това по всяко време операторът има достъп до робота и опцията за дистанционно управление.

Функциите за почистване, вградени в графичния интерфейс, са следните:

- Създаване на карта: операторът ръчно разхожда робота из стаята, докато се създаде пълна карта. След това всяка процедура използва направената карта [16]. Използва се алгоритъм за последователно сканиране на картата, за да се гарантира, че процедурите за почистване се извършват навсякъде

- Започнете да почиствате само с четки

- Започнете почистване с четки и пръскане с препарат

- Включване и активиране на дезинфекциращата подсистема с UV лампи. Тъй като има опасност за хората и животните по време на тази процедура, преди стартиране на лампите се активира алгоритъм за защита от радиация. В този режим роботът засича присъствието на хора или животни. Ако такива бъдат открити, UV лампите се изключват и сигнализират в графичния интерфейс на оператора. За по-добро възприятие роботът е в състояние да разпознава маркери и други обекти [17].

- Почистване включително всички системи /четки, миене, UV лампи/ на санитарния модул: В този режим се активират всички системи на почистващия модул, както и всички алгоритми.

Обратната връзка от робота към оператора се реализира чрез визуализация на данни от сензорите и устройствата на робота. Операторът може да следи видео потока от камерата на робота, да проследява местоположението на робота според навигационната карта, да види дали има зададена операция и до каква степен тя е завършена. Когато възникне проблем при извършване на операция или възникне неизправност, той също се съобщава на оператора.

В допълнение към автоматизираните операции, към графичния интерфейс са добавени бутони за управление на отделните функции на модулите. По този начин операторът може да използва робота в режим на ръчна телеоперация, за диагностициране на системи и др.

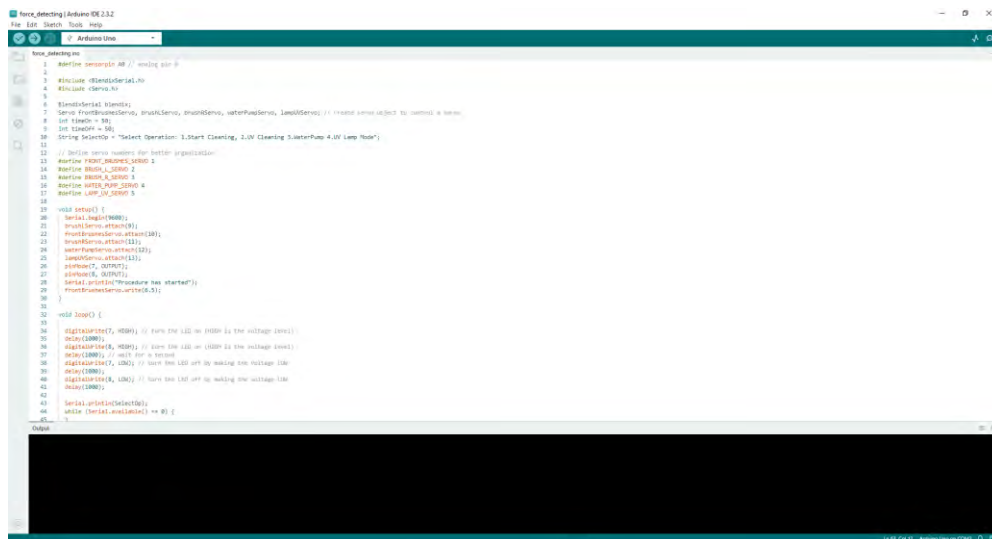
3.6.3. Симулация и резултати от верифицирането на модула и алгоритъма на работа

За верифициране на модула и алгоритъма на работа се състави и реализира компютърна симулация посредством синхронизираното прилагане на следните софтуери: SolidWorks-CAD, Proteus 8.16, Arduino IDE, HND Software's Virtual Serial Port Tools и Blender 3D.

Първата стъпка е създаване на кода в Arduino IDE. За да може да се осъществи симулацията в кода се въвежда `BlendixSerial.h` библиотеката, която позволява комуникацията между симулацията на хардуера в Proteus с 3D средата на Blender 3D, в която е поставен конструираният робот.

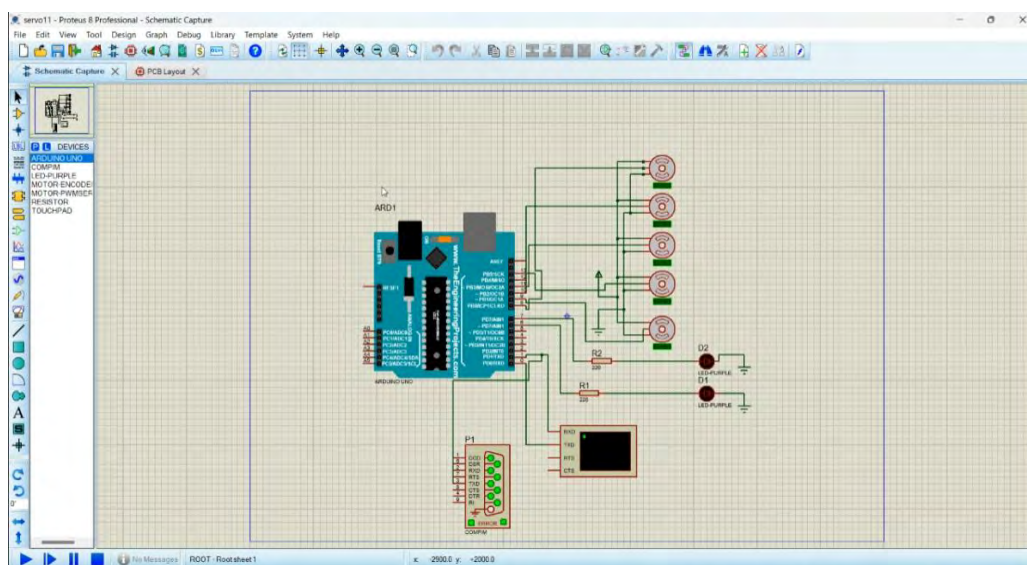
При стартиране на симулационната част в Proteus във виртуалния терминал на симулацията операторът избира кои режими на работа на почистващия модул да активира. В програмата са заложили 4 режима, които се въвеждат с цифра в терминала:

- 1- начало на почистването (започва да почиства с четки след като раменната конструкция бъде свалена);
- 2- UV почистване (почистване с четката, позиционирана в задната част на робота заедно с UV лампата прикачена към нея);
- 3- почистване с вода (стартира електрическата помпа за почистване с вода посредством дюзите, позиционирани в предната част на ротационните четки);
- 4- режим UV лампа (почистване на околната среда посредством UV лампата монтирана върху най-горната плоча на мобилния робот)



Фиг.110 Създаването на програмата в средата на Arduino IDE

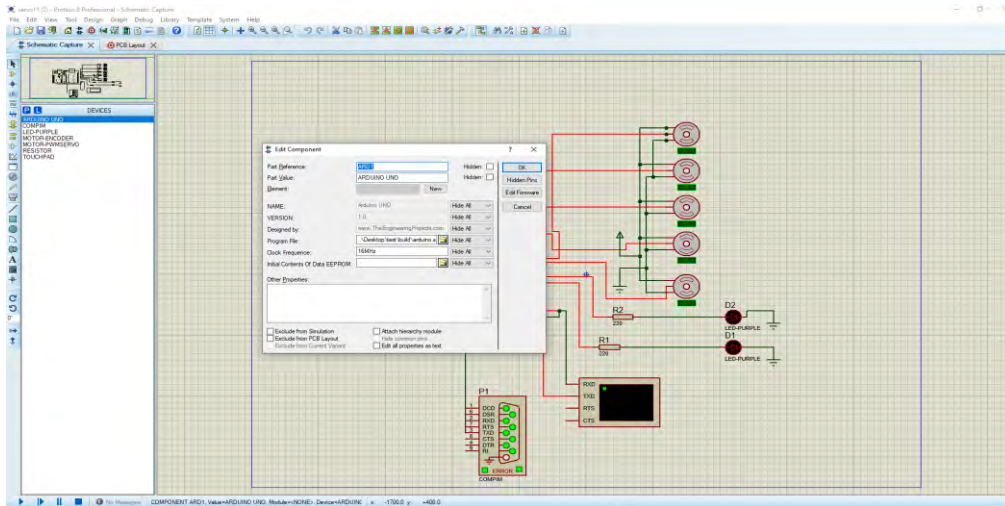
Втора стъпка след създаването на програмата в средата на Arduino IDE е изграждането на хардуерната система и нейното окабеляване в средата на Proteus. За тази цел се използват библиотеките на Proteus, в които има готови компоненти - мотори, сензори, контролери и т.н. На фигура 111 е показана създадената хардуерна схема, свързваща компонентите, изграждащи модула както и виртуален терминал за въвеждане и визуализиране на данните.



Фиг.111 Хардуерна схема на почистващия модул

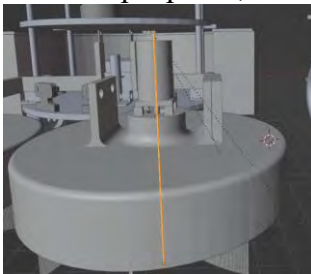
След като всички компоненти са свързани правилно и хардуерната система, изграждаща модула е готова следващата стъпка е да се зареди програмата написана в Arduino IDE в средата на Proteus, където извършваме симулацията. Зареждането на програмата става като в средата на Proteus върху микроконтролера се натисне с десен бутон. В резултат излиза менюто Edit Component, през което зареждаме Arduino файла.

Следваща стъпка е отварянето на 3D модела на робота в 3D средата на Blender. За да може да се движи модела по определения желан от нас начин конструираните части в Solidworks трябва да бъдат експортирани отделно и асемблирани на ново в средата на Blender 3D където се задават необходимите ограничения към 3D обектите и се ригват обектите, които ще бъдат управлявани от програмата и ще се движат по определения начин.

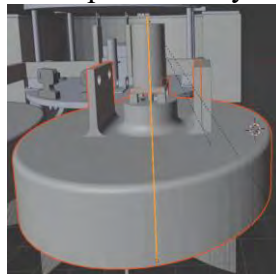


Фиг.112 Зареждането на Arduino файла в средата на Proteus

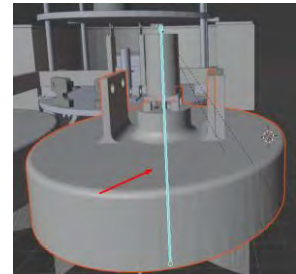
Стъпките на ригване, за да може моделът да се движи по определена траектория на база на управляващата програма , която е имплементирана в симулацията Proteus са следните:



Фиг.114 Генерираната арматура за управление на 3D обекта



Фиг.115 Позициониране на арматурата в 3D обекта

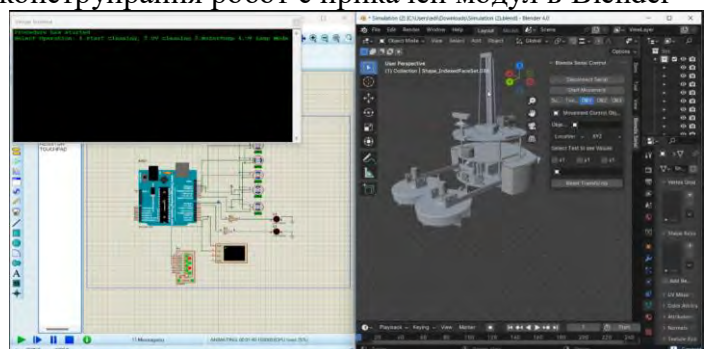


Фиг.117 Избор на кост която да бъде ригната

В обектен режим се създава „арматура“ (скелетна структура, която управлява даден 3D обект). След като се създаде арматурата, тя се премества на мястото, на което искаме да се осъществи движението и се поставя към цялостната конструкция на модела, който желаем да бъде ригнат (фиг.114 и фиг.115). В обектен режим, след като се избере желанния модел, арматурата се превключва в режим на поза. Следва избор на кост, която да бъде ригната, т.е. скрепена към модела (фигура 117). И накрая се натиска ctrl + p и се избира "кост". След ригването на модела добавяме приложението към Blender BlendixSerial. Библиотеката BlendixSerial е проектирана да работи заедно с добавката blendixserial в Blender. То позволява контролиране на обекти в Blender чрез серийна комуникация с Arduino или други микроконтролери. Чрез емулирането на COM портовете от HND Software's Virtual Serial Port Tools заедно с BlendixSerial започва осъществяването на връзката между симулацията в Proteus и 3D конструирания робот с прикачен модул в Blender



Фиг.119 Емулиране на COM портовете



Фиг.120 Стартиране на симулацията в Proteus

Стартираме симулацията в Proteus. Посредством виртуалният терминал се визуализира стартираната програма, която е написана в Arduino и имплементирана в Proteus чрез генерирания

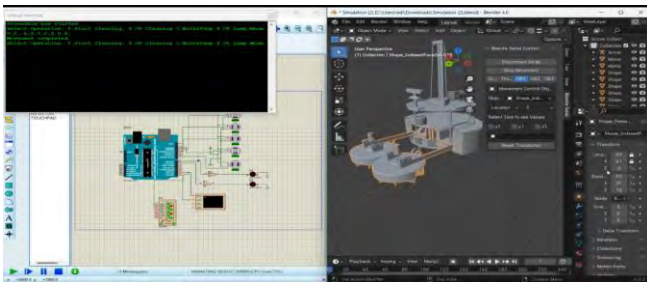
hex файл. Терминалът изписва, че процедурата е започнала и изчаква да въведем режима на почистване, който искаме да стартираме. Възможните режими на работа са:

1- начало на почистването, 2- UV почистване, 3- почистване с вода, 4- режим UV лампа
 Преди започване на програмата, в съответствие с алгоритъма за почистване, рамената, на които са монтирани почистващите четки се вдигат автоматично. След въвеждане на команда за почистване в терминала, те се спускат до крайно долно положение, в което е предвидено да започне ротацията на четките и почистването на база на извършен геометричен анализ в процеса на проектиране на изделието. За да се визуализират промените в 3D модела на конструирания робот в падащото меню на BlendixSerial задължително трябва да сме натиснали Connect Serial и Start Movement, както и да сме посочили емулацияния порт. – (фиг.123)

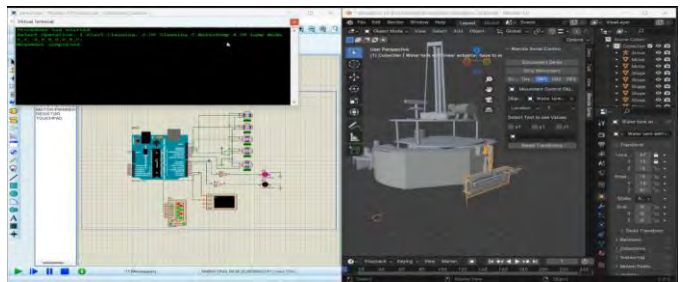


Фигура 123 Избор на COM порт в средата на Blender

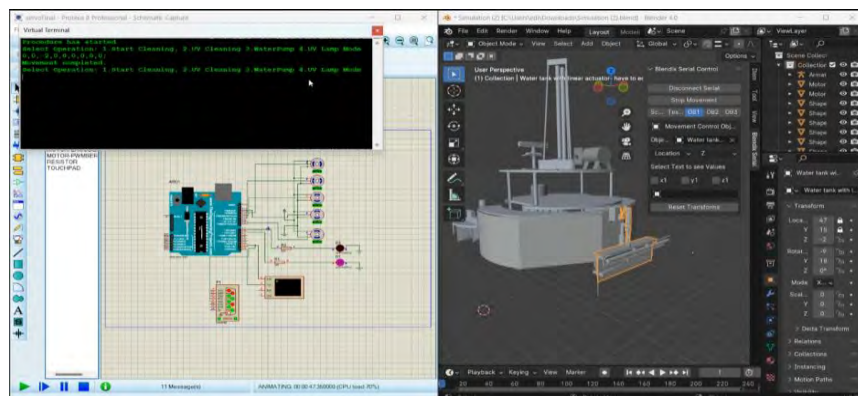
След като въведем 1 и стартираме почистването посредством предните четки, раменната конструкция се смъква и започва въртенето на четките. Терминалът визуализира преместването на рамената и изписва, че действието е успешно извършено. Преместването е визуализирано и в менюто в Blender, показващо преместването във височина. След което се очаква въвеждане на следваща команда за изпълнение. (Фиг. 126) Изображението на фигурата 127 съответства на въвеждането на режима за почистване с четката, позиционирана в задната част на робота и активираната UV лампа.



Фиг.126 Изпълненият режим в симулационната среда и очакване на следваща команда



Фиг.127 Стартиране на почистване с четка и UV лампа в симулационната среда



Фиг.128 Активираният режим за почистване на околната среда в симулацията

В симулационната част на Proteus лампата, съответстваща на UV лампата в 3D модела се включва на база на кода, а визуализацията на изменението се осъществява с промяна на цвета.

Въвеждането във виртуалния терминал на Proteus режима за почистване на околната среда посредством UV лампа се визуализира като съответстваща лампа в симулацията, монтирана на най-горната плоча в Blender 3D променя цвета си. В случая на почистването с вода се активира дадения двигател отговарящ за онагледаването на помпата в средата на Proteus.(фиг.128)

3.7 Управление на многоцелевия робот с прикрепен модул с антропоморфен робот

За операции, изискващи определено ниво на прецизност, към платформата на мобилния робот е прикрепен модул с антропоморфен робот.



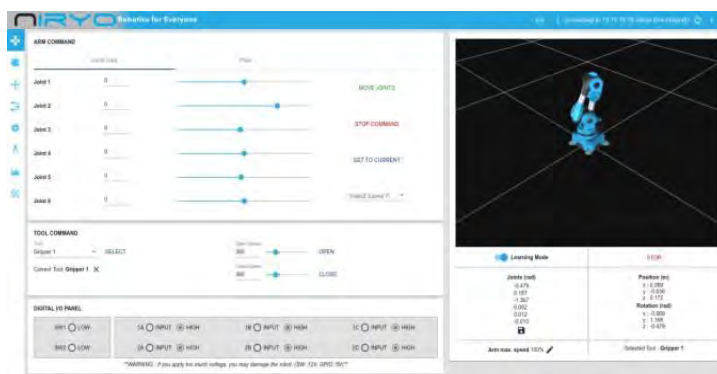
Фиг.129 Модулът с прикрепения Niryo One към мобилния робот

3.7.1 Софтуер за управление

За функционирането на антропоморфния робот се използва управляващия софтуер Niryo One Studio [94], работещ под различни операционни системи (Windows, MacOS, Linux). Софтуерът, показан на фигура 130 се използва за настройка на робота, преместване на робота, програмиране на последователност от команди и др.

а) Лявото меню на прозореца на софтуера разполага със следните панели за управление на робота:

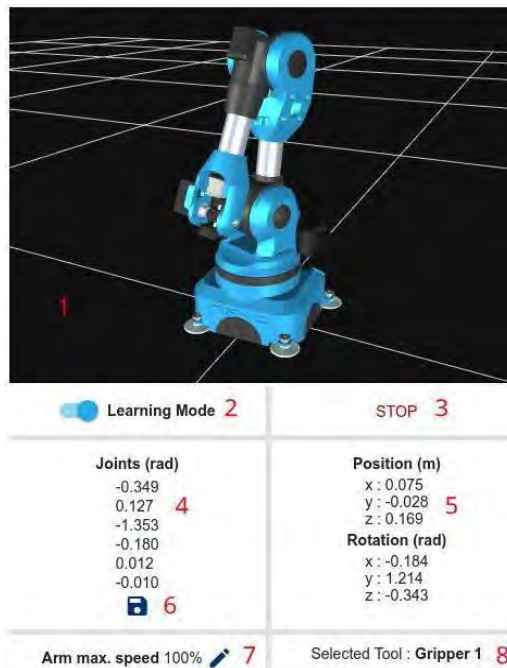
- Нирио блокове;
- Запазени позиции;
- Запазени последователности (с възможност за автоматично стартиране);
- Настройки (Wi-Fi, Raspberry Pi, версии на софтуера);
- Калибриране;
- Състояние на хардуера; Отстраняване на грешки и регистрационни файлове.



Фиг.130 Прозорец на софтуера за управление на Niryo One

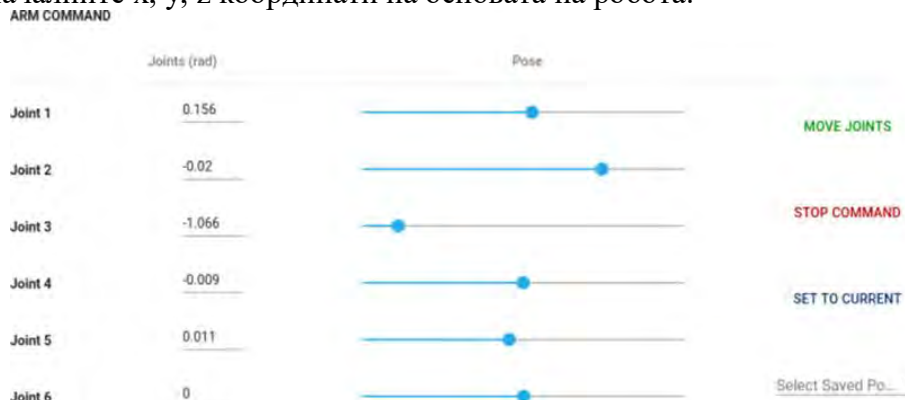
б) Управление на робота

При пристъпване към управлението на робота, предварително трябва да бъде извършено калибриране, след което посредством индикиращ LED светодиод (зелен или син) се отчита, че роботът е успешно стартиран. Софтуерът за управление на антропоморфният робот Niryo One Studio, има секция, предоставяща информация за робота и съдържа режима на обучение, показано на фигура 131.



Фиг.131 Постоянна дясна секция в прозореца на софтуера

- Софтуера предоставя 3D изглед на робота. 3D изгледът помага за отчитането дали е направено правилно ръчното/автоматичното калибриране. При разлика във физическите позиции на робота и това което се вижда на екрана се налага да се направи автоматичното калибриране отново.
- При активиране на бутон „режим на обучение“ се деактивира въртящия момент на всички двигатели. Когато роботът е в „режим на обучение“, може да бъде преместван свободно. Когато режимът на обучение е изключен, всички двигатели се активират и робота не може да се движи свободно. Когато режимът на обучение е изключен, всички двигатели се активират и робота не може да се движи свободно
- По всяко време при активиране на „стоп“ бутона се спира текущата програма на робота.
- Текущо състояние на ставите (J1-J6).
- Текуща позиция и ориентация на TCP (Tool Center Point – центъра на специализирания хващач - диспенсър).
- Бутон „запазване на текущата позиция“ позволява да се запази текущата позиция, след което може да се използва повторно по-късно.
- Текуща максимална скорост на ставата може да се промени по всяко време (0-100%).
- Текущо избран инструмент. С. Задвижване на робота В командния панел на управляващия робота софтуер има 3 раздела. В раздела „Arm command“ [94] може да се избира действие от раздела „стави“ или раздела „поза“. Разделът „стави“ позволява да се премества всяка става независимо. Разделът „поза“ позволява да се премести позицията на TCP спрямо началните x, y, z координати на основата на робота.



Фиг.132 Arm command раздел в управляващия софтуер

За да се изпълни команда, се следва следната процедура:

- Натискане на бутона „задаване на текуща позиция“. Командата ще настрои осите, позицията и ориентацията към текущото състояние на робота. Това дава възможност за последваща промяна на стойностите от текущото състояние (напр. при желание да се премести само ос 5 или да се премести ТСР по оста и да се запази същата ориентация).
- Модифициране на стойностите на ставите или поза (позиция + ориентация).
- Натискане върху бутона „Move Joints“ или „Move Pose“, за да се изпълни командата за преместване.
- След като командата бъде изпълнена, се получава известие в долната част на екрана.
- По всяко време има опция да се отмени текущото изпълнение на командата, като се кликне върху бутона „Стоп“. Докато се изпълнява текуща команда за преместване не може да се изпълнява нова команда. При желание да се стартира нова команда, първо трябва да се спре предишната. Има възможност също да се избере позиция, която преди това е била запазена с полето за избор „избор на запазена позиция“, като така се заместват първата и втората стъпки

ЗАКЛЮЧЕНИЕ

В настоящия дисертационен труд е представена разработка на многоцелеви теле-управляем сервизен робот, чрез който могат да бъдат осъществени обход, наблюдение, охрана и сигнализация на обекти, почистване и хигиенизиране на помещения, повдигане и пренасяне на товари, дистрибуция на лекарства, транспортиране и складиране на продукти, извършване на манипулационни операции с елементи.

Устройството на робота включва универсална мобилна платформа и шест модула, посредством които робота извършва напълно различни по естество и начин на изпълнение дейности и операции. Принципът му на работа се базира на телеуправлението.

Разработеният многоцелеви теле-управляем робот намира приложение в напълно различни работни среди – здравни заведения, търговски вериги, складови помещения, животновъдни ферми като разширява диапазона на приложение на сервизните роботи и допринася за тяхното развитие и широко разпространение. Той е част от сервизните роботи, посредством които все повече ще се подобряват условията на работа и ще се облекчава трудовата дейност на хората.

За в бъдеще научните изследвания върху представения в дисертацията сервизен робот ще бъдат продължени като за целта ще се реализира физически модел на робота и ще се верифицира в реални условия на работа.

ПРИНОСИ

- Проучени и анализирани са основни международни документи, програми и политики, свързани с глобалното развитие, в контекста на роботиката и сервизните работи в тях.
- За всеки един от режимите на работа на робота, в зависимост от използвания модул, е разработен алгоритъм на управление и е описана последователността на изпълнение на всеки един от тях.
- Обвързани са 4 софтуера - Proteus 8.16, Arduino, HND Software's Virtual Serial Port Tools и Blender 3D за синхронизирано предаване на данни между тях.
- На базата на паралелното, съвместно и синхронизирано приложение на петте софтуера (Proteus 8.16, SolidWorks, Arduino IDE , HND Software's Virtual Serial Port Tools и Blender 3D) са съставени и реализирани симулации, посредством които са верифицирани съставените от докторанта алгоритми на работа на робота в различните му режими.
- Разработена е мобилна роботизирана платформа с възможност за закрепване на различни по конструкция и предназначение модули.
- Разработени са шест отделни модула, с различни функции, дизайн и предназначение, посредством които роботът може да работи в разнородни среди и да извършва напълно различни по естество и начин на изпълнение дейности и операции.
- Разработено е телеуправление на многофункционална мобилна роботизирана платформа чрез уеб базиран графичен интерфейс, посредством MQTT и уеб сокети

Благодарности

Благодаря на своя научен ръководител проф. д-р Найден Шиваров, за оказаното съдействие при провеждане на настоящите изследвания и подготовката на дисертацията.

ЛИТЕРАТУРА

- [16] D. Chikurtev, K. Yovchev. Marker-based Automatic Dataset Collection for Robotic Vision System. Mechanisms and Machine Science, 102, Springer Science and Business Media B.V., 2021
- [17] D. Chikurtev, N. Chivarov, S.Chivarov, A. Chikurteva. Mobile robot localization and navigation using LIDAR and indoor GPS. IFAC papers online, 54, 13, Elsevier, 2021
- [44] MD Moniruzzaman, Alexander Rassau, Douglas Chai, Syed Mohammed Shamsul Islam, Teleoperation methods and enhancement techniques for mobile robots: A comprehensive survey, Robotics and Autonomous Systems, Volume 150, 2022
- [45] M. Mukhandi, D. Portugal, S. Pereira, MS Couceiro. A novel solution for securing robot communications based on the MQTT protocol and ROS. In 2019 IEEE/SICE International Symposium on System Integration (SII) 2019 Jan 14 (pp. 608-613). IEEE.
- [49] N. Chivarov, D. Chikurtev, I. Rangelov, A. Gigov and N. Shivarov. Educational Mobile Robot Platform for Line Following, 6th International Scientific Conference "Education, Science, Innovations", Pernik, Bulgaria, p.p. 290 – 298, 2016
- [50] N. Chivarov, D. Chikurtev, N. Shivarov, K. Yovchev, I. Rangelov, A. Gigov, M. Konstantinov, M. Marinov, V. Lazarov, E. Markov. Telecontrolled Service Robot For Increasing The Quality Of Life Of Elderly And Disabled, ADP 2017, June, Sozopol, Bulgaria; p.p. 171- 175
- [70] Z. Kapić, A. Crnković, E. Mujčić, J. Hamzabegović. A web application for remote control of ROS robot based on WebSocket protocol and Django development environment. In IOP Conference Series: Materials Science and Engineering 2021 Nov 1 (Vol. 1208, No. 1, p. 012035). IOP Publishing.
- [74] <https://digital-strategy.ec.europa.eu/en/library/digital-economy-and-society-index-desi-2021>
- [81] <https://www.aliexpress.com/item/32813245169.html?spm=2114.12010612.8148356.6.56a9fee63LCph6>
- [84] Technical documents – Lely
- [85] Cleaning farm robot - ProCleaner X100 - WASHPOWER - autonomous / pig barn
- [86] Self-learning, mobile cleaning robot ensures absolute cleanliness in production line - Inceptive Mind
- [87] <https://vikiwat.com/elektrodvigatel-postoyannotokov-s-reduktor-12vdc-150rpm-2w-vfga37fg-32i-a601.html>
- [88] 12V Linear Actuator With Bracket Aluminum Alloy Small Electric Push Rod Mini Linear Motor|DC Motor|
- [89] Durable DC 12V 130PSI Agricultural Electric Water Pump Black Micro High Pressure Diaphragm Water Sprayer Car Wash 12 V|Pumps|
- [90] 12v Small Acuator Motor Stroke Length 30mm/50mm/100mm/150mm Load Capacity 150n Mirco Electric Linear Servos - Dc Motor
- [92] <https://niryo.com/docs/niryo-one/user-manual/mechanical-specifications>
- [93] <https://www.handsontec.com/dataspecs/module/BTS7960%20Motor%20Driver.pdf>
- [94] <https://niryo.com/docs/niryo-one/user-manual/complete-user-manual>

Abstracts of Dissertations

Number 8, 2024

INSTITUTE OF INFORMATION AND COMMUNICATION TECHNOLOGIES
BULGARIAN ACADEMY OF SCIENCES

БЪЛГАРСКА АКАДЕМИЯ НА НАУКИТЕ

ИНСТИТУТ ПО ИНФОРМАЦИОННИ И КОМУНИКАЦИОННИ ТЕХНОЛОГИИ

Брой 8, 2024

Автореферати на дисертации