

МАГІСТЕРСЬКА РОБОТА  
МР.ПМК<sub>М</sub>-032.00.00.000 ПЗ

Група ПМК<sub>М</sub>-21-1

Веркалець Анатолій

Андрійович

2022



(повне найменування закладу вищої освіти)

Інститут інженерної механіки

Кафедра комп'ютеризованого машинобудування

Освітній рівень магістр

Спеціальність 131 – Прикладна механіка

(шифр і назва)

**ЗАТВЕРДЖУЮ**

Завідувач кафедри \_\_\_\_\_

« \_\_\_\_ » \_\_\_\_\_ 20\_\_ року

## **ЗАВДАННЯ НА МАГІСТЕРСЬКУ РОБОТУ СТУДЕНТОВІ**

Мельнику Олександрю Богдановичу

(прізвище, ім'я, по батькові)

1. Тема роботи: Розробка системи керування транспортної платформи для обслуговування навчально-дослідницької дільниці механічної обробки

керівник роботи Лукань Т.В., асистент кафедри КМВ

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом закладу вищої освіти від “29” 10 2021 року № 546/7

2. Строки подання студентом роботи 23 грудня 2022р.

3. Вихідні дані до роботи: креслення складових елементів конвеєра та умови експлуатації маніпулятора; середовище програм SolidWorks

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити)

1. Теоретична частина. 2. Конструкторська частина. 3. Технологічна частина. 4.

Програмування конвеєра

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)

1. Гнучка автоматизована система – 1 лист А1. 2. Будова стрічкового конвеєра – 1 лист А1. 3. Траєкторія руху інструменту при обробці деталі – 1 лист А1. 4. Імітація обробки деталі – 1 лист А1. 5. Деталювання- 3 листи А3, 9 листів А4.

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
1	Лукань Т.В., асистент кафедри КМВ		
2	Лукань Т.В., асистент кафедри КМВ		
3	Лукань Т.В., асистент кафедри КМВ		

7. Дата видачі завдання \_\_\_\_\_

**КАЛЕНДАРНИЙ ПЛАН**

№ з/п	Назва етапів магістерської роботи	Термін виконання етапів роботи	Примітки
1	Теоретична частина		
2	Конструкторська частина		
3	Технологічна частина		
4	Програмування конвеєра		
5	Оформлення графічної частини		
6	Оформлення пояснювальної записки		

Студент \_\_\_\_\_ Веркалець А.А.  
( підпис ) ( прізвище та ініціали )

Керівник роботи \_\_\_\_\_ Лукань Т.В.  
( підпис ) ( прізвище та ініціали )

“ 01 ” 09 2022 р.

## Реферат

Магістерська кваліфікаційна робота на тему «Розробка системи керування транспортної платформи для обслуговування навчально-дослідницької ділянки механічної обробки» складається зі 75 аркушів. До неї входять 51 рисунок, 5 таблиць, 2 додатки. В роботі використано 10 літературних джерел.

Об'єкт дослідження – Транспортна платформа для обслуговування навчально-дослідницької ділянки.

Предмет дослідження – Програма транспортної платформи для обслуговування навчально дослідницької ділянки.

Мета роботи – Програмування транспортної платформи для обслуговування навчально дослідницької ділянки.

Для проектування конструкції роботизованого візка було проаналізовано його призначення та умови роботи. Були вибрані комплектуючі та в середовищі SolidWorks розроблені конструкції роботизованого візка.

Для виготовлення деталі «Верхня основа основа» було розроблено технологічний процес, вибрано верстатне обладнання, різальний інструмент, вибрано режими різання та виконана симуляція обробки в середовищі CamWorks.

Програмування проводилось в середовищі розробки Arduino IDE з використанням стандартних бібліотек.

Ключові слова: транспортна платформа, SolidWorks, механічна обробка, технологічний процес, режим різання, тривимірна модель, CamWorks, Arduino.

Студент Веркалець А.А.

## Summary

Master's thesis on the topic: Development of a transport platform control system for servicing the training and research department of mechanical processing. This work consists of 75 sheets. It includes 51 figures, 5 tables, 2 appendices. 10 bibliographic titles were used in the work.

The object of study - Transport platform for servicing the educational and research station.

The subject of research - The program of the transport platform for the service of the educational research station.

The purpose of the work - Programming of the transport platform for the service of the educational research station.

To design the robotic cart, its purpose and operating conditions were analyzed. The components were selected and the designs of the robotic cart were developed in the SolidWorks environment. For the manufacture of the "Upper basic" part, a technological process was developed, machine tools, cutting tools were selected, cutting modes were selected, and machining simulation was performed in the CamWorks environment.

Programming was carried out in the Arduino IDE development environment using standard libraries.

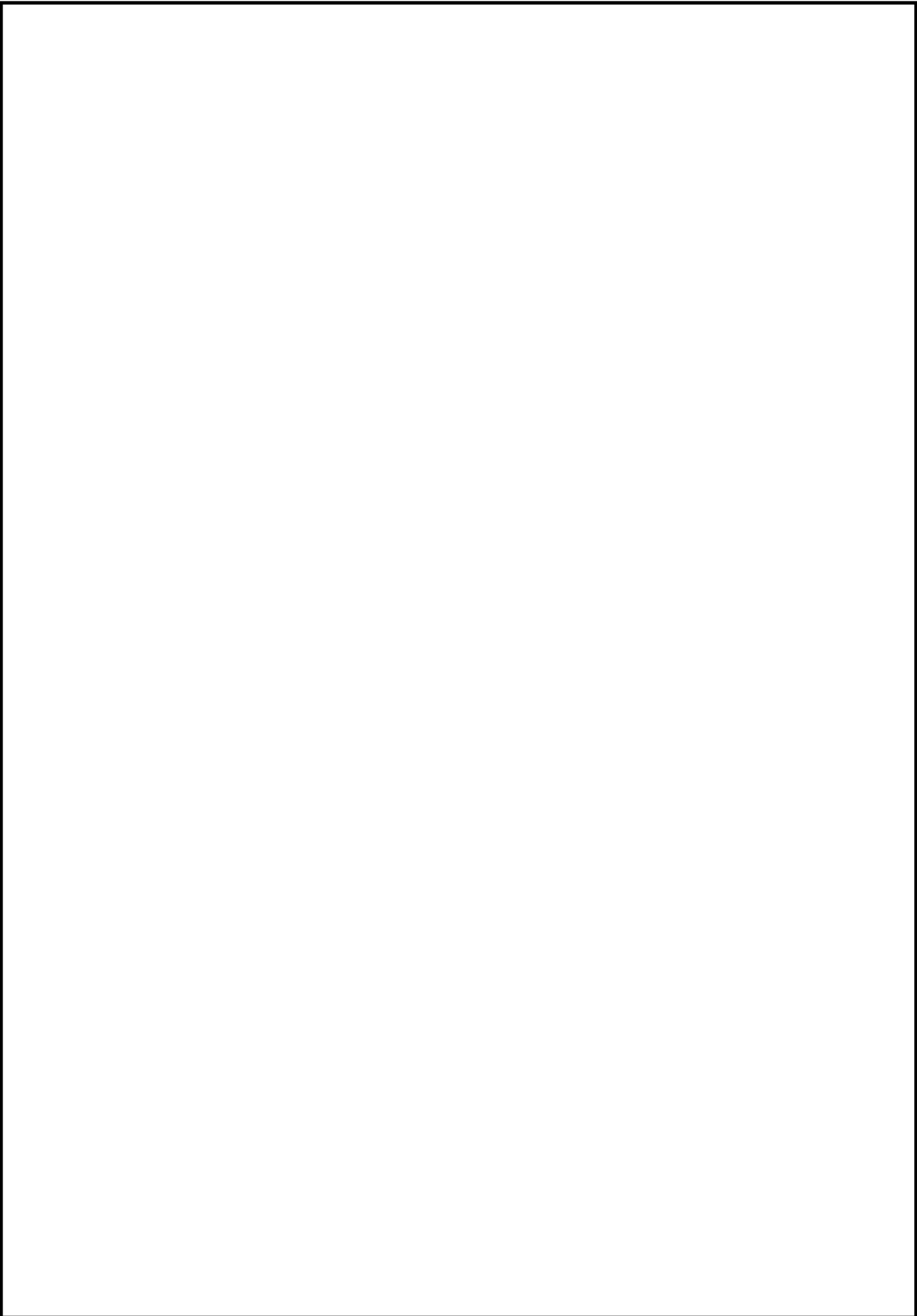
Keywords: Transport platform, SolidWorks, machining, technological process, cutting model, three-dimensional model, CamWorks, Arduino.

Student Verkalets A.A.

## ЗМІСТ

Вступ.....	8
1. Теоретичні відомості.....	9
1.1 Гнучке автоматизоване виробництво.....	9
1.2 Транспортні роботи.....	19
1.3 Приклади використання транспортних роботів.....	23
2. Конструкторська частина.....	25
2.1 Опис конструкції спроектованої платформи та принцип роботи....	25
2.2 Підбір комплектуючих.....	25
2.3 Опис середовища 3D моделювання системи SolidWorks.....	35
2.4 Створення 3D моделей комплектуючих та загальної збірки моделі.....	37
3. Технологічна частина.....	39
3.1 Опис конструкції та призначення не стандартної деталі.....	39
3.2 Аналіз технологічності конструкції деталі.....	41
3.3 Опис і аналіз технологічного процесу.....	43
3.4 Вибір різального інструменту та режимів різання.....	45
3.5 Створення керуючої програми.....	48
4. Програмування платформи.....	55
4.1 Загальна інформація про програмування Arduino.....	55
4.2 Редактор програми Arduino IDE.....	56
4.3 Програма для пересування платформи.....	59
4.3.1 Константи підключення входів Arduino UNO.....	59
4.3.2 Константи встановлених значень.....	61
4.3.3 Встановлення обраховуваних значень.....	62
4.3.4 Основа програми.....	64
4.3.5 Програма для крокового двигуна.....	71
4.4 Калібрування платформи.....	72
4.5 Симуляція роботи платформи.....	73
Висновки.....	74
Список літератури.....	75

					<i>МР.ПМКм-032.00.00.000 ПЗ</i>		
Зм.	Арк.	№ Докум.	Підпис	Дата			
Розроб.		Веркалець А А			Літ.	Арк.	Аркуші
Перевір.		Лукань Т. В.				7	75
Затверд.					<b>Пояснювальна записка</b>		
					<b>ІФНТУНГ</b>		



					<i>MP.ПМКМ-032.00.00.000 ПЗ</i>	<i>Арк.</i>
<i>Зм.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		2

## ВСТУП

Розвинене машинобудування є важливою складовою і вінцем технологічної і економічної бази держави, в основі якої лежить сконцентрований інтелект минулих і сьогоденних поколінь. Машина повинна відповідати технологічним вимогам, вимогам по обмеженню матеріаломісткості та енерговитратам, і разом з тим повинна бути високопродуктивною, надійною в експлуатації. Для успішного розв'язання такого комплексу вимог, спеціалісти галузі машинобудування повинні знати основи цілого ряду наук і в тому числі основ автоматичної. Курсове проектування сприяє закріпленню, поглибленню і узагальненню теоретичних знань, а також застосуванню цих знань до комплексного розв'язку конкретної інженерної задачі дослідження і розрахунку. Кількість стадій передбачена єдиною системою конструкторської документації (ЄСКД): технічне завдання, технічна пропозиція, ескізний проект, технічний і робочий проекти. Виконання цих стадій ґрунтується на методах автоматизованого проектування машин (САПР), базою яких є логічне формулювання завдання, математичні моделі процесів і об'єктів, вибір і обґрунтування системних моделей, обчислення на ЕОМ і аналіз результатів, складання прогнозів досягнення мети. Можливість виконання розрахунків ґрунтується на формалізації задач і зв'язків, алгоритмах і програмах розрахунків та програмному забезпеченні, що дозволяє проектанту вести діалог з ЕОМ.

					<i>МР.ПМКм-032.00.00.000 ПЗ</i>	<i>Арк.</i>
<i>Зм.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		8

# 1. ТЕОРЕТИЧНІ ВІДОМОСТІ

## 1.1 Гнучке автоматизоване виробництво

Гнучке автоматизоване виробництво (ГАВ) являє собою організаційно-технічну виробничу систему, що функціонує на основі комплексної автоматизації і здатна (у діапазоні технічних можливостей) з мінімальними витратами й у короткі терміни, не припиняючи виробничого процесу і не зупиняючи устаткування, переходити на випуск нової продукції довільної номенклатури шляхом перебудови технологічного процесу (у межах наявного станочного парку й обслуговуючого комплексу) за рахунок заміни програм управління [2]. Основний показник – ступінь гнучкості – визначається витратами часу на розширення номенклатури продукції, що випускається, та необхідними додатковими витратами при переході на випуск нової продукції. Поняття гнучкості виробничої системи багатокритеріальне. Залежно від конкретно розв'язуваних завдань системою висуваються різноманітні аспекти гнучкості:

- машинна гнучкість – простота перебудови технологічного устаткування для виробництва заданої множини деталей;
- технологічна гнучкість – спроможність устаткування виробляти задану множину деталей різними способами; 29
- структурна гнучкість – можливість розширення гнучкої виробничої системи (ГВС) за рахунок уведення нових технологічних модулів;
- виробнича гнучкість – спроможність системи продовжувати обробку деталей у разі відмови окремих технологічних елементів;
- маршрутна гнучкість – можливість зміни порядку виконання операцій без перепланування устаткування;
- гнучкість за обсягом – спроможність системи ефективно функціонувати при різних обсягах виробництва;
- гнучкість за номенклатурою – спроможність системи виготовляти різноманітні деталі. У гнучкому автоматизованому виробництві робота

					<i>МР.ПМКм-032.00.00.000 ПЗ</i>	<i>Арк.</i>
<i>Зм.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		9



рахунок включення нового обладнання); гнучкість технології (оцінюється числом операцій, реалізованих в даній виробничій системі); організаційну гнучкість (оцінюється складністю переходу на обробку будь-якого виробу даної номенклатури);

- Поняття гнучкості ієрархічне: від окремих одиниць обладнання і операційних технологій до заводського виробничого процесу в цілому;
- ГВ базується на безлюдній (малолюдній) технології і дозволяє відмовитись від традиційної технічної і супроводжувачої документації(без паперові технології);
- ГВ здатні обробляти вироби заданої номенклатури в будь-якому порядку їх прямування;
- ГВ функціонують на основі принципу централізації обробки – об'єкт виробництва не потребує додаткової доробки на іншому обладнанні;
- ГВ функціонують, як правило на основі уніфікованих технологій.

Систематизація понять в області гнучкого виробництва

УС - універсальні верстати з ручним управлінням; УГОД - ділянки групової обробки деталей; ОЦ - оброблювальний центр; РТК - робототехнічний комплекс; СЧПУ - верстати з числовим програмним забезпеченням; ГПМ - гнучкий виробничий модуль; АССИ - автоматична система слідкування за інструментом; АСПИ - автоматична система налагоджування інструмента; АСАБ - автоматична система аналізу браку; АСИО - автоматизована система інструментального забезпечення; АТСС - автоматизована транспортно-складальна система; АСОН - автоматизована система забезпечення надійності; АСУО - автоматизована система видалення відходів; АСОК - автоматизована система забезпечення якості; СОФ - система забезпечення функціонування обладнання; ГПС - гнучка виробнича система; ГАЛ - гнучка автоматизована лінія; ГАУ - гнучка автоматизована ділянка; САПР - система автоматизованого проектування; АСТПП - автоматизована система технологічної підготовки виробництва; АСОПП - автоматизована система оперативного планування

					<i>МР.ПМКМ-032.00.00.000 ПЗ</i>	<i>Арк.</i>
<i>Зм.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		<i>11</i>

виробництва; АСМП - автоматизована система моделювання виробництва; АССОО - автоматизована система змісту і обслуговування обладнання; АССОЕС - автоматизована система енергозабезпечення енергозберігання; АСМСПО - автоматизована система матеріалозбереження і переробки відходів; АРМ - автоматизоване робоче місце; ГАП - гнучке автоматизоване виробництво; АЛ - автоматизовані лінії; ГАЦ - гнучкий автоматизований цех; АСНИ - автоматизована система наукових досліджень; АСУП - автоматизована система управління виробництвом; АСПП - автоматизована система планування виробництвом; АСУЕП - автоматизована система управління економікою виробництва; АСДП - автоматизована система діловодства; АСУСНАБ - автоматизована система управління постачанням; ЦУОБ - цехи з універсальним обладнанням; АСУСБИТ - автоматизована система управління збутом; АСУКАДРИ - автоматизована система управління кадрами; ГАЗ - гнучкий автоматизований завод.

#### Гнучке виробництво в народному господарстві

Значення наукових і технічних проблем гнучких виробничих систем для народного господарства полягає у створенні й удосконаленні методів і засобів технологічного, інформаційного та математичного забезпечення, які підвищують рівень гнучкості й автоматизації виробничих процесів, забезпечують їх автономне функціонування у різних галузях народного господарства. Як наслідок, досягається підвищення продуктивності, надійності, ритмічності та поліпшення інших показників діяльності як окремих автоматизованих технологічних одиниць, так і інтегрованих систем загалом, а також створюються умови для усунення людини зі сфери працевітких і небезпечних робіт та інтелектуалізації її діяльності.

#### Пропорційна варіація факторів виробництва

У довгостроковому періоді всі фактори виробництва є змінними. Підприємство, намагаючись збільшити обсяги виробництва продукції, залучає все більше ресурсів, тобто збільшує масштаб виробництва. При цьому

					<i>МР.ПМКм-032.00.00.000 ПЗ</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		12

спостерігається різна віддача (ефект) від зміни масштабу виробництва.

Зростаюча віддача від масштабу відбувається тоді, коли обсяг виробництва продукції зростає відчутніше, ніж обсяги використання ресурсів. Наприклад, при подвоєнні факторів виробництва обсяг випуску продукції зростає більш ніж у два рази.

**Зростаючий ефект масштабу виробництва може досягатися завдяки впливу таких факторів:**

1. *Поділ праці.* На більших підприємствах можлива спеціалізація, що призводить до підвищення продуктивності праці і зниження витрат.

2. *Поліпшення управління.* На більших підприємствах виділяють спеціалістів, які безпосередньо займаються маркетингом, рекламою, постачанням, науково-технічною роботою і т. д. Це дає змогу збільшити ефективність діяльності підприємства.

3. *Збільшення масштабів виробництва не вимагає пропорційного збільшення всіх ресурсів.* Наприклад, збільшення вдвічі кількості верстатів на заводі не вимагає такого ж збільшення механіків, електриків, охоронців, бухгалтерів, а також витрат на освітлення, опалення, вентиляцію і т. д.

Постійна (стала) віддача від масштабу виробництва спостерігається тоді, коли обсяги виробництва продукції та обсяги використання ресурсів зростають пропорційно. Збільшення вдвічі ресурсів виробництва призводить до подвоєння обсягів випуску продукції.

Спадна віддача від масштабу виробництва відбувається тоді, коли випуск продукції зростає менш відчутніше, ніж зростають обсяги використаних факторів виробництва. Наприклад, збільшення вдвічі ресурсів призводить до підвищення випуску продукції лише в півтора рази.

**Спадаючий масштаб виробництва виникає через вплив таких факторів:**

1. Значна інерція великих систем, втрата ними гнучкості, необхідної в умовах нестабільного ринку.

					<i>МР.ПМКм-032.00.00.000 ПЗ</i>	<i>Арк.</i>
<i>Зм.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		13

2. Вихід підприємства за межі порога керованості (великі розміри підприємства створюють громіздку систему управління. "а-трудняється координація проміжних ланок, обмін інформацією, а це приводить до зниження ефективності управлінських рішень).

#### Оцінка технологічності об'єктів гнучкого виробництва

У гнучкому виробництві, організованому за принципами групової технології, особливого значення набуває обробки деталей на технологічність. При цьому розглядаються вимоги технологічності не окремої деталі, а цілої групи. Особливе значення мають такі характеристики: вимоги до розмірів і допусків на обробку, спільність розмірних ланцюгів, вибір баз, матеріал заготовок, метод отримання заготовок, зручність обробки з одного встановлення.

Конструкція і технологія виготовлення деталі, вузла чи виробу технологічна, якщо відповідає всім експлуатаційним вимогам і для її виготовлення витрачається мінімальна кількість суспільно-корисної праці. В умовах гнучкого виробництва, де конструкція виробу і технологія його виготовлення проектується в рамках автоматизованої системи технологічної підготовки виробництва (АСТПВ), технологічність конструкції виробів впливає на вихідні параметри ГВС (гнучкість, продуктивність, надійність, якість обробки).

При обробці деталей на технологічність до них ставлять такі вимоги:

1. симетричні конструкції деталей, але з деякими не симетричними елементами (виступами, проточками, бур тинками, отворами та ін..) Мають бути приведені до повної симетрії введенням аналогічних конструктивних елементів;
2. асиметричні деталі повинні мати різко виражену асиметричність, причому більша перевага надається асиметрії за зовнішнім контуром, ніж за внутрішнім;
3. при внутрішній асиметричності і необхідності орієнтувати деталі

					<i>МР.ПМКм-032.00.00.000 ПЗ</i>	<i>Арк.</i>
<i>Зм.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		14

за внутрішнім контуром на їх зовнішній поверхні мав бути відмітний елемент, розташований у відповідності з внутрішнім контуром;

1. для зручності орієнтування деталей складної конфігурації на їх поверхню вводяться додаткові конструктивні елементи, що не потребуються при роботі;
2. для запобігання зчеплювання деталей типу спіральних пружин при автоматизації процесу їх транспортування і орієнтації вводять деякі конструктивні зміни;
3. розміри пазів, заглиблень, отворів великого діаметра та інших елементів конструкції деталей мають бути такими, щоб запобігти можливості западання однієї деталі в іншу;
4. для складальних автоматів нетехнологічними є деталі з плоского тонкого листа; вироби мають забезпечити складання з одного боку;
5. вимоги до технологічності конструкції визначають введення елементів, що забезпечують задану точність відносно розташування складових частин виробу при складанні (напрямні фаски, забірні конуси тощо), а також фіксують і компенсують елементи.

Для підвищення технологічності деталей і виробів необхідно враховувати таке; конструкція деталі та інструмент, що застосовується для неї, мають сприяти скороченню обсягу механічної обробки, зменшенню кількості інструмента, необхідного для обробки групи деталей. Для цього доцільно віддавати перевагу централізованому постачанню ГВС інструментом, оптимізувати кількість інструмента, в інструментальних магазинах застосовувати переважно інструмент простої форми; конструкція деталей, що входять до групи для обробки в ГВС, має передбачати ідентичність базування та забезпечувати обробку деталей без перестановлень на супутники різної форми чи з мінімальною кількістю перестановлень: слід намагатися обмежити ряд розмірів і посадок; введення класних розмірів має суворо обґрунтовуватися, а їх кількість мав бути мінімальною; необхідно підвищувати;ступінь уніфікації,

					<i>MP.ПМКм-032.00.00.000 ПЗ</i>	<i>Арк.</i>
<i>Зм.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		15

нормалізації та стандартизації форм і розмірів деталей.

При використанні верстатів з ЧПУ традиційні поняття про технологічність дещо змінюються. Так, виготовлення валу зі змінним кроком гвинтової поверхні, що нарізається на універсальних токарно-гвинторізних верстатах, нетехнологічне. Ті самі операції на верстатах з ЧПУ і Гнучких модулях виконувати не важко і деталь переходить до . розряду технологічних. Оскільки технологія обробки складних поверхонь на "оброблювальних центрах" спрощена, то складність геометричних форм поверхонь деталей не впливає негативно на оцінку технологічності.

Можна сформулювати деякі основні вимоги до технологічності деталей, що обробляються на програмно-керованому обладнанні, вмонтованому в ГВС.

При конструюванні деталей типу тіл обертання необхідно намагатись об'єднати декілька простих елементів в одну складну форму, наближення форми заготовки до остаточної форми деталі, уніфікації і виточок під ущільнення, застосування канавок спеціальної форми для виходу шліфувального кола, розташування одностипних канавок з одного боку деталі, узгодження розмірів торцевих поверхонь деталей з номенклатурою centruючих пристроїв, зменшення довжини елементів, що надходять, на поверхні деталі;

При конструюванні корпусних і плоских деталей, що потребують, фрезерної обробки, необхідно домагатися уніфікації ресурсів поєднання елементів деталі, вибору співвідношення радіусів взаємно перпендикулярних поверхонь, що стискаються (цей вибір забезпечує найбільшу торцеву поверхню інструмента), застосування симетричних конструкцій деталі.

Обробка деталей на ГВМ характеризується багато інструментальністю і централізацією операцій. Це зумовлює підвищені вимоги до базування деталі і стану базової поверхні. При цьому конструкція деталі має забезпечувати вільний доступ кінцевого інструмента до поверхонь, що обробляються, при її незмінному встановленні, високу точність при консольній обробці, оскільки борштанги не можна використовувати через незручності їх зберігання в

					<i>MP.ПМКм-032.00.00.000 ПЗ</i>	<i>Арк.</i>
<i>Зм.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		16

інструментальному магазині.

При розробці креслень, деталей, призначених для виготовлення в ГВС, необхідно виконувати деякі специфічні вимоги: розміри деталей задаються у прямокутній системі координат, яка прив'язується до координат верстата; базові поверхні деталей, особливо установчі, мають суміщуватися з координатними площинами; кріпильні отвори, розташовані на тій чи іншій відстані від центра основного отвору і задані центральним кутом дуги між їх осями і діаметром, задаються координатами осей кожного отвору; розміри окремих поверхонь, груп отворів або елементів деталі можна ув'язувати з початком координат і задавати в місцевій системі координат.

#### Якісна оцінка технологічності

Якісна оцінка технологічності виробництва утруднює процес автоматизації обробки конструкції на технологічність. Кількісну оцінку технологічності найчастіше виконують за трудомісткістю і собівартістю виготовлення виробів, а іноді за їх матеріаломісткістю. За трудомісткістю технологічність оцінюють так:

$$T=A \cdot K \cdot P$$

T- базовий показник за трудомісткістю; A- трудомісткість; K - коефіцієнт складу виробу; P- коефіцієнт зниження трудомісткості

Аналогічно розраховується технологічність за собівартістю. Коефіцієнт складності має перевищувати одиницю. Його визначають як відношення значень технічних параметрів виробу, то проектується, і виробу-аналога. Базові показники розраховують за абсолютним і відносним значеннями та за масою, питомою матеріаломісткістю, трудомісткістю, питомою трудомісткістю виготовлення виробу, технологічною собівартістю і питомою технологічною собівартістю. При визначенні базових показників виникають значні труднощі, оскільки складно порівнювати трудомісткість нової і базової конструкцій через відміни умов їх виготовлення. Тому базові показники зручно порівнювати з еталоном, що являє собою ідеальну конструкцію виробу, складальної одиниці,

					<i>MP.ПМКм-032.00.00.000 ПЗ</i>	<i>Арк.</i>
<i>Зм.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		17

деталі. Показник технологічності еталону і трудомісткість його виготовлення приймаються за одиницю. Проте складно зберігати еталони для різноманітних типів деталей, які постійно вдосконалюються в конструктивному виконанні. Питання кількісної оцінки технологічності конструкції, перебувають у стадії розробки і потребують глибокої методологічної проробки і подальшого вдосконалення. (рис. 1.1).



Рисунок 1.1 — Схема конструкторсько-технологічної документації

## 1.2 Транспортні роботи

Транспортний робот - це програмно-керований механізм із засобами завантаження і розвантаження транспортуючих виробів, автоматично переміщається по програмі управління в межах ділянки, цеху, заводу.

Характерними ознаками промислового робота є:

- автоматичне керування;
- здатність до швидкого і відносно легкому перепрограмуванню (зміні послідовності, системи і змісту команд);
- здатність до виконання трудових дій.

У першому випадку параметри траєкторії руху вантажу вибирають оператори, система робота реалізує і підтримує їх значення.

У другому випадку зазначення параметрів руху робота здійснюється повністю його автономною системою автоматичним управлінням.

У третьому - всі параметри руху (швидкість, шлях, зупинка і ін.) формуються під впливом зовнішніх сигналів, які передаються безконтактним способом. Система управління робота побудована на комбінованих способах завдання програми, забезпечує розшифровку цих сигналів і вироблення команд на виконання.

Транспортні роботи класифікують:

1. За місцем переміщення вантажів:
  1. на підлогові;
  2. підвісні.
2. За способом переміщення вантажів:
  1. транспортують вантаж захватом маніпулятора;
  2. буксируючі транспортні засоби з вантажем і переміщують вантаж без захоплення.
3. За функціональними можливостями:
  1. виконують тільки одну операцію, зв'язану з переміщенням вантажу;

					<i>MP.ПМКм-032.00.00.000 ПЗ</i>	<i>Арк.</i>
<i>Зм.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		19

2. виконують декілька операцій (переміщення , завантаження , вивантаження і т.п.) .
4. За типом ходового пристрою:
  1. пневмоколісні;
  2. на рейковому ходу;
  3. гусеничні та крокуючі .
5. За способом управління:
  1. з ручною системою;
  2. автономні;
  3. працюючі по зовнішніх сигналам;
  4. з комбінованою системою .

Підлогові транспортні роботи можуть бути рейковими і безрейковими, а підвісні - монорейковими, консольно-крановими і порталними. Роботи вертикального переміщення можуть бути з дискретним і безперервним рухом.

**Підлогові безрейкові транспортні роботи забезпечують:**

- значне зниження капіталовкладень на монтаж траси;
- усунення захаращення виробничих площ стаціонарними транспортними пристроями;
- економію виробничих площ внаслідок суміщення трас робіт з внутрішньоцехових проїздами та проходами;
- оптимальний розподіл вантажопотоків завдяки простоті трас і можливості зміни маршрутних схем і парку робіт - машин при перебудові виробництва або збільшенні випуску продукції;
- простоту взаємодії з допоміжним технологічним обладнанням на робочих місцях.

Але потрібно, щоб траса була чистою.

Оскільки промислові транспортні роботи функціонують у виробничій обстановці, де обладнання ГАВ часто виступає як перешкода, виникає необхідність попередньої прокладки безпечного маршруту транспортування

					<i>MP.ПМКм-032.00.00.000 ПЗ</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		20

вантажів з метою запобігання зіткнень.

### Недолік транспортних роботів з індукційним самонаведенням

Головним недоліком транспортних роботів з індукційним самонаведенням є необхідність попередньої прокладки струмопровідного кабелю вздовж траси руху. В умовах ГАВ технологічні маршрути і пов'язані з ними траси для транспортування вантажів часто змінюються . Тому на практиці не вдається , як правило , заздалегідь розрахувати і прокласти всі траси , потреба в яких може виникнути в майбутньому. Водночас перекладка кабелю наведення при зміні траси трудомістка і вимагає втручання людини. Все це знижує продуктивність і автономність ГАВ .

### Призначення транспортних роботів

Транспортні роботи призначені для автоматизованого транспортування об'єктів , а також для управління різними транспортними системами . Дослідження і розробки по створенню транспортних роботів інтенсивно ведуться в усьому світі. При цьому виділяються чотири принципово різних типи - наземні , повітреплаваючі , водоплавні і підземні . Теорія і практика трьох останніх типів досягли ще в цілому того рівня , щоб говорити про них сьогодні як про загальні реальності.

Практичний розвиток отримали нині наземні транспортні роботи , які можуть бути колісними , крокуючими і гусеничними . Найбільший розвиток і поширення в даний час отримали колісні транспортні роботи , використовувані досить широко в промислових автоматизованих транспортно - складських системах і гнучких автоматизованих виробництвах у вигляді мобільних автоматичних кранів , автоматичних керованих візків ( АУТ ) , робокар та ін , що оснащуються в багатьох випадках різними маніпуляційними пристроями. У найпростішому вигляді такі роботи йдуть по рейках або за маршрутом над кабелем , прокладеним під поверхнею підлоги. Генератор частоти , подаючи струм по кабелю , створює магнітне поле , яке вловлюється двома датчиками приймального пристрою візка , напрямними її по необхідному маршруту. У

					<i>МР.ПМКм-032.00.00.000 ПЗ</i>	<i>Арк.</i>
<i>Зм.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		21

більш складному варіанті візок обладнується автономною керуючою ЕОМ .

### Розширення функціональних можливостей

Для розширення функціональних можливостей транспортних роботів на їх борту іноді встановлюється один або декілька маніпуляторів. У результаті виходять комбіновані маніпуляційно - транспортні роботи , які можуть не тільки транспортувати вантажі , але й самостійно завантажуватися і розвантажуватися , а також маніпулювати вантажами. Розробка таких універсальних роботів для ГАВ становить інтерес з різних точок зору.

У маніпуляційно - транспортних роботах сконцентровані багато проблем механіки , теорії адаптивного управління , навігації та штучного інтелекту.

**З точки зору механіки** рухова система цих роботів являє собою комплекс виконавчих механізмів з голономних і неголономними зв'язками , що дозволяє автоматизувати широкий спектр ручних і транспортних операцій .

**З позицій теорії управління** ці роботи є складною багатовимірною системою , активно взаємодіють із зовнішнім середовищем. Організація автономного функціонування таких роботів в змінній виробничій обстановці неможлива без розвиненої інформаційно- навігаційної системи і пов'язаної з нею адаптивної системи управління .

Нарешті, з точки зору теорії штучного інтелекту маніпуляційно - транспортні роботи цікаві тим , що вони функціонують в змінних умовах , де частина обладнання ГАВ грає роль перешкод , а об'єкти маніпулювання і вантажі , що підлягають транспортуванню , можуть мати довільне розташування та орієнтацію. Тому виникає необхідність надати адаптивній системі управління такі інтелектуальні функції , як розпізнавання об'єктів , аналіз обстановки , формування понять і моделювання навколишнього середовища.

### Переваги

Промислові роботи зарекомендували себе як гнучкі автоматизовані засоби реалізації внутрішньоцехових і міжопераційних матеріальних зв'язків,

					<i>МР.ПМКм-032.00.00.000 ПЗ</i>	<i>Арк.</i>
<i>Зм.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		22

що володіють цілим рядом переваг в порівнянні з іншими пристроями:

- малі габаритні розміри рухомого органу;
- великий діапазон регулювання швидкості переміщення;
- автоматичні переміщення;
- повне вивільнення проїздів після проходження транспортного робота для транспорту інших видів;
- автономність.

### **1.3 Приклади використання транспортних роботів**

Нині у світі багато стартапів, які розробляють складські роботи. Один із них – стартап Covariant. 2018 року до команди звернувся відомий постачальник складських рішень Knapp. Компанія хотіла створити роботизовану руку, яка б сортувала товар.

Проект намагалися розробити понад 6 років. Проблема в тому, що роботизована рука повинна вміти поводитися з різними типами товарів. Наприклад, акуратно брати упаковку з ліками, щоби не пошкодити її.

Для створення робота для автоматичного сортування на складі, Covariant використовували технологію штучного інтелекту, який може навчатися самостійно.

Алгоритми робота аналізують положення предмета у просторі, обчислюють, з якого кута краще взяти його, адаптує послідовність рухів кожної операції. Спочатку роботизованою рукою керувала людина. Це робили у тому, щоб штучний інтелект навчався. Після кожної маніпуляції система записувала послідовність дій та аналізувала їх. Також роботизована рука самостійно намагалася сортувати понад 1000 різних предметів. Система записувала вдалі та невдалі спроби.

Технологія штучного інтелекту популярна у сфері складської логістики. Наприклад, автомобільний концерн Škoda навіть розробив AI-рішення, яке допомагає завантажити контейнер із мінімальними втратами. Про це в іншій нашій статті.

					<i>МР.ПМКм-032.00.00.000 ПЗ</i>	<i>Арк.</i>
<i>Зм.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		23

Чи використовують складські роботи в Україні?

Коли говорять про інновації на складах, найчастіше згадують розвинуті американські компанії, такі як Amazon. Проте українські підприємства також автоматизують свої склади, де роботи допомагають людям.

Роботи на логістичних комплексах "Нової Пошти"

У 2020 році великий поштовий оператор "Нова Пошта" почав використовувати роботів на своїх логістичних.

					<i>MP.ПМКМ-032.00.00.000 ПЗ</i>	<i>Арк.</i>
<i>Зм.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		24

## 2. КОНСТРУКТОРСЬКА ЧАСТИНА

### 2.1 Опис конструкції спроектованої платформи та принцип роботи

Модель рухомої платформи реалізовано на базі Arduino UNO і стандартних комплектуючих деталей (motor shield, мотор редуктор з колесом, ультразвуковий дальномір, датчик лінії, кроковий двигун, сервопривід).

Робот оснащений датчиками дані яких він аналізує, керуючими модулями для виводу результатів роботи, та скетчем по алгоритму якого він буде працювати. В нашому випадку платформа оснащена трьома датчиками лінії (TCRT5000), ультразвуковим дальноміром (HC-SR04), домкратом та поворотним механізмом. Управляючими модулями стали два мотор-редуктори з колесом, кроковий двигун та сервопривід. Вони підключені до arduino uno через motor shield. Кроковий двигун відповідатиме за переміщення домкрату. За допомогою кінцевих вимикачів отримаємо корекну зупинку підняття чи опускання домкрату. Сервопривід відповідатиме за поворот столу з деталями.

Для того, щоб платформа не пересувалась постійно було спроектовано відділення, яке внеможливлуватиме пересування платформи допоки робот, який забирає деталь не подасть сигнал, щоб платформа рушила. Для цього використовуватимуться кнопки, які робот перед початком роботи натискатиме тим самим подаючи сигнал платформі про необхідність пересування.

### 2.2 Підбір комплектуючих

#### Arduino UNO

Arduino Uno – це пристрій на основі мікроконтролера ATmega328 (datasheet). До його складу входить все необхідне для зручної роботи з мікроконтролером: 14 цифрових входів/виходів (з них 6 можуть використовуватися як ШИМ-виходи), 6 аналогових входів, кварцовий резонатор на 16 МГц, роз'єм USB, роз'єм живлення, роз'єм для внутрішньосхемного програмування (ICSP) та кнопка скидання. Для початку роботи з пристроєм досить просто подати живлення від AC/DC-адаптера або батарейки, або підключити його до комп'ютера за допомогою кабелю USB.(рис. 2.1).

					<i>MP.ПМКМ-032.00.00.000 ПЗ</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		25

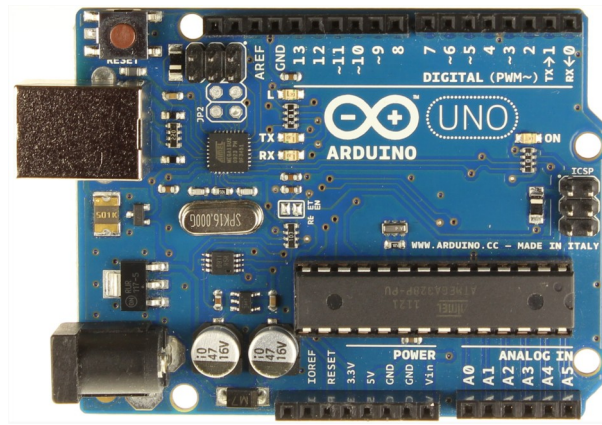


Рисунок 2.1 — Плата Arduino UNO

## HC-SO04



Рисунок 2.2 — HC-SR04

Робота модуля заснована на принципі ехолокації. Модуль посилає ультразвуковий сигнал і приймає його відображення від об'єкта. Вимірявши час між відправленням і отриманням імпульсу, неважко вирахувати відстань до перешкоди.

Підключення ультразвукового далекоміра до Arduino

Модуль підключається чотирма проводами. Контакти VCC і GND служать для підключення харчування, а Trig і Echo- для відправки і прийому сигналів далекоміра. Підключимо їх до пінам 10 і 11 відповідно.





### Область застосування:

- Виявлення перешкод
- Виявлення білих і чорних ліній

### Характеристики:

- Використаний інфрачервоний датчик відображення TCRT5000
- Впевнене визначення перешкоди від 1 до 25мм
- Цифровий вихід компаратора з максимальним навантаженням 15мА
- Регулювання чутливості потенціометром
- Робоча напруга від 3.3В до 5В
- Розміри модуля: 3.2 x 1.4см
- Граничний компаратор напруги LM393



Рисунок 2.6 – Датчик лінії TCRT5000

### Мотор Shield (L298P)

L298P Motor Shield - це плата драйвера двигунів постійного струму, що

					<i>MP.ПМКм-032.00.00.000 ПЗ</i>	<i>Арк.</i>
<i>Зм.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		29

використовує мікросхему потужного драйвера електродвигунів L298P, яка може безпосередньо управляти двома двигунами постійного струму; струм через навантаження - до 2 ампер. Вихідні інтерфейси управління двигунами використовують 8 високошвидкісних діодів Шоттки в якості захисту. Дана плата може бути встановлена безпосередньо на плату Arduino.

Цей Шилд особливо стане в нагоді тим, хто збирає роботів і робоплатформи з використанням колекторних двигунів. На платі, крім самого драйвера двигунів L298, зручно розміщені роз'єми для підключення Bluetooth модуля, сервоприводу і датчиків. Зручне підключення, гарна якість монтажу вигідно відрізняють даний Шилд від аналогічних.



Рисунок 2.7 — Моторшилд

### **Мотор редуктори**

Коефіцієнт передачі: 48: 1

Робоча напруга: 3 - 12 В (рекомендований 6 - 8 В)

Число оборотів холостого ходу (3 В): 100 об / хв

Число оборотів холостого ходу (6 В): 200 об / хв

Струм холостого ходу (3 В): 60 мА

					<i>МР.ПМКм-032.00.00.000 ПЗ</i>	<i>Арк.</i>
<i>Зм.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		30

Струм холостого ходу (6 В): 71 мА

Струм: 70-470 мА (в залежності від навантаження)

Крутний момент (3 В): 1,2 кг х см

Крутний момент (6 В): 1,92 кг х см

Максимальна швидкість: 70 см / с

Матеріал шини: гума

Діаметр колеса з шиною: 6,6 см

Ширина колеса: 2,66 см

Вага комплекту: 45 г



Рисунок 2.8 — Мотор редуктор

### Кроковий двигун

Кроковий двигун — електричний двигун, в якому імпульсне живлення електричним струмом призводить до того, що його ротор не обертається неперервно, а виконує щоразу обертальний рух на заданий кут. Завдяки цьому, кут повороту ротора залежить від числа поданих імпульсів струму, а кутова швидкість ротора точно рівна частоті імпульсів помноженій на кут повороту ротора за один цикл роботи двигуна.



кількість еквівалентних полюсів, на відміну від основних полюсів, на яких розташовані обмотки. Зазвичай використовуються 4 основні полюси для 3,6 градусних двигунів і 8 основних полюсів для 1,8-0,9 градусних двигунів. Зубці ротора забезпечують менший опір магнітного ланцюга у певних положеннях ротора, що покращує статичний і динамічний момент. Це забезпечується відповідним розташуванням зубців, коли частина зубців ротора знаходиться строго навпроти зубців статора, а частина між ними.

Ротор гібридного двигуна має зубці, розташовані в осьовому напрямку. Ротор розділений на дві частини, між якими розташований циліндричний постійний магніт. Таким чином, зубці верхньої половинки ротора є північними полюсами, а зубці нижньої половинки — південними. Крім того, верхня і нижня половинки ротора повернуті один відносно одного на половину кута кроку зубців. Число пар полюсів ротора дорівнює кількості зубців на одній з його половинок. Зубчасті полюсні наконечники ротора, як і статор, набрані з окремих пластин для зменшення втрат на вихрові струми.

### **Драйвер крокового двигуна**

Драйвер ULN2003 призначений для використання у парі з кроковими двигунами. Як результат, дана модель активно застосовується при складанні та тестуванні макетів бюджетної офісної техніки (3D - принтерів, факсів, копіїв), а також недорогих виробничих верстатів з ЧПУ, яким необхідна висока точність переміщень. Передатне відношення пристрою становить 1/64, кут кроку – 5,625 x 1/64. Завдяки адаптованому введенню модель сумісна з будь-якими видами контролерів. Для керування використовуються спеціальні програми, записані в керуючому пристрої. Для захисту елементів ланцюгів від викиду напруги ULN2003 застосовуються спеціальні діоди, встановлені на виходах. Світлодіоди забезпечують індикацію працюючих фаз.

					<i>МР.ПМКМ-032.00.00.000 ПЗ</i>	<i>Арк.</i>
<i>Зм.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		33



Кулька опора - <https://beegreen.com.ua/uk-ua/opornij-rolik-dviguna-n20-mini-3pi-15024>



Рисунок 2.12 — Опорний ролик

### 2.3 Опис середовища 3D моделювання системи SolidWorks

Створені ескізи використовуються для створення елементів 3-х вимірної твердотільної моделі.

Елементи панелі інструментів поділені на 2 види, а саме:

- Для утворення
- Ті, що корегують

Тривимірний об'єкт ґрунтується на плоскому ескізі.

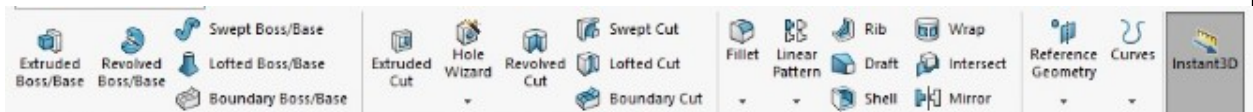






Рисунок 2.13 – Вигляд панелі інструментів


Для створення 3 - D моделі використовують спеціальну панель інструментів (рис. 2.1) до неї входять багато елементів, деякі з них:

Витягнути  - Ескізи повинні бути замкненими для нетонкостінних елементів, а також профілі повинні не перетинатися для нетонкостінних елементів.

Витягнутий виріз  - Ескізи повинні бути замкнуті, не перетинатися.

Обертання  - Ескізи повинні бути замкнуті для нетонкостінних елементів. Профілі повинні не перетинатися для нетонкостінних елементів. Профілі не можуть перетинати осьову лінію. Якщо ескіз містить кілька осьових ліній, виберіть осьову лінію, яку потрібно використовувати як вісь обертання.

Обертаний виріз  - Ескізи повинні бути замкнуті, непересічні. Ескізи не можуть перетинати осьову лінію. Якщо ескіз містить кілька осьових ліній, виберіть осьову лінію, яку потрібно використовувати як вісь обертання.

Круговий масив  - Створення масиву елементів, розташованих по колу навколо зазначеної осі.

Отвір під кріплення - Можна створити наступні типи отворів під кріплення:



- цековка,



- зенковка,



- отвір,



- мітчик,



- трубна конічна різьба.

В геометричному моделюванні розрізняють поверхневе (каркасно-поверхове) і твердотільне моделювання. В двох випадках, результатом моделювання являється оболонка, (або декілька оболонок) яка описує поверхню моделюючого об'єкту. В поверхневому моделюванні спочатку створюються і модифікуються необхідним чином поверхні, що описують окремі елементи моделюючого об'єкту. Ці поверхні обрізають по лінії пересічення, сполучають один з одним поверхнями скруглення чи переходу, а також виконують над ними інші операції. Потім із отриманих поверхонь складають оболонку. В поверхневому моделюванні результуюча оболонка не обов'язково повинна бути замкнутою, вона може відображати лиш частину (головну частину) моделюючого об'єкту.

В твердотільному моделюванні з початку, робота йде з оболонками тіл, а

										Арк.
										36
Зм.	Арк.	№ докум.	Підпис	Дата	МР.ПМКМ-032.00.00.000 ПЗ					

не з окремими поверхнями. Оболонки повністю описують поверхні моделюючих об'єктів відокремлюючи їх внутрішній об'єм від іншої частини простору. Процес побудови оболонки тіла в даному випадку аналогічний процесу створення моделюючого об'єкту. Спочатку створюється оболонка деякої простої форми, надалі оболонка форми змінюється необхідним чином, для цього використовуються Булеві операції над тілами, операція побудови тонкостінного тіла, операція скруглення та інші операції. За допомогою операцій, оболонці тіла надають необхідну форму.

#### **2.4 Створення 3D моделей комплектуючих та загальної збірки моделі**

В середовищі розробки SolidWorks створюємо 3D модель описаної платформи (рис. 2.14). З окремих 3D моделей комплектуючих складаємо загальний вигляд. На ньому зображені:

- Arduino Uno R3;
- Моторшилд L298P;
- Датчики лінії TCRT5000;
- Ехолокатор HC-SR04;
- Батарейний відсік;
- Мотор-редуктори;
- Роликова опора;
- Домкрат;
- Поворотний стіл;
- Нестандартні деталі;

					<i>MP.ПМКм-032.00.00.000 ПЗ</i>	<i>Арк.</i>
<i>Зм.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		37

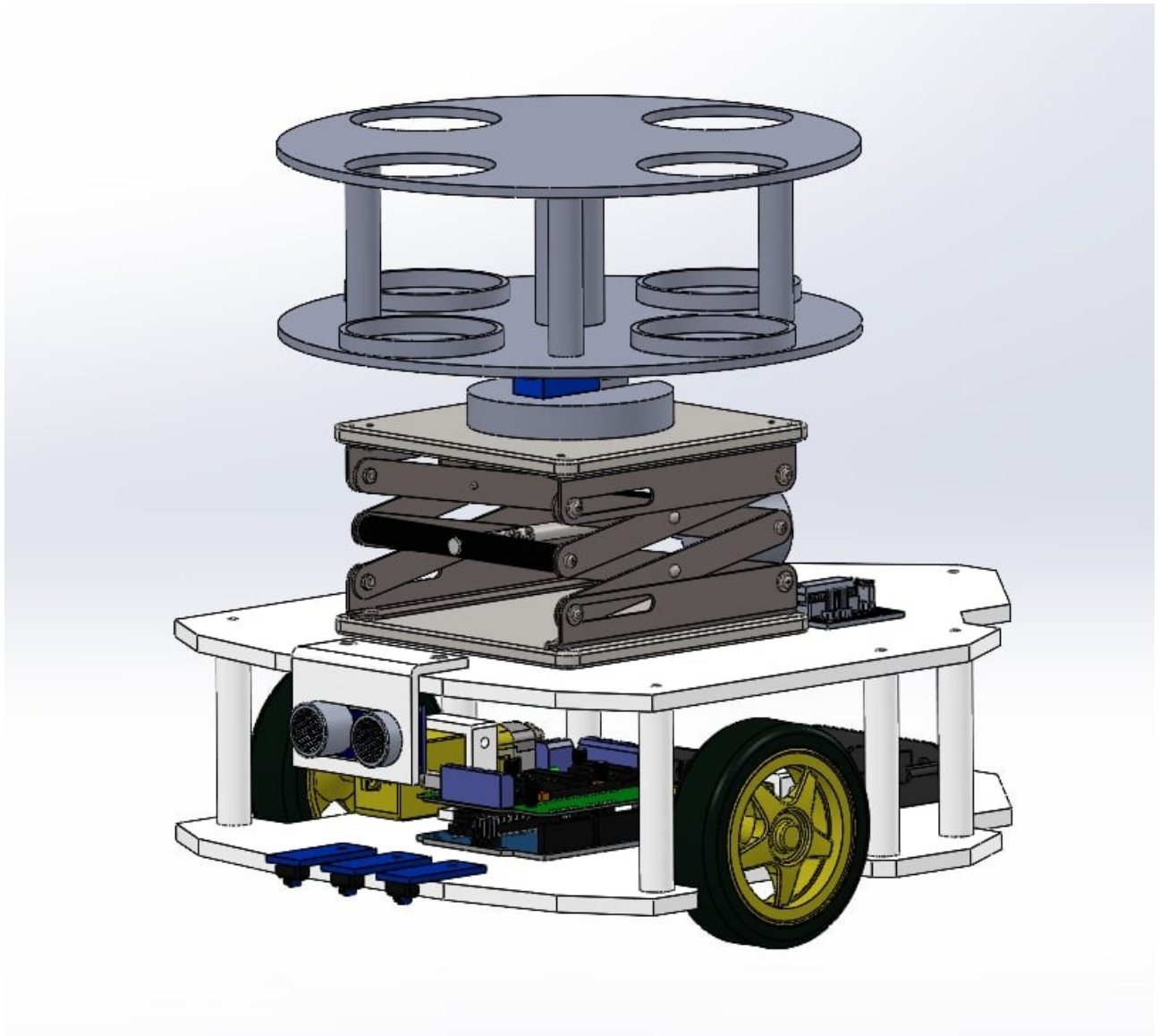


Рисунок 2.14 — 3D модель платформи

Зм.	Арк.	№ докум.	Підпис	Дата

МР.ПМКМ-032.00.00.000 ПЗ

Арк.

38

### 3 ТЕХНОЛОГІЧНА ЧАСТИНА

#### 3.1 Опис конструкції та призначення не стандартної деталі

Основною нашої платформи являється корпус. Складається з нижньої основи, верхньої основи, опор та додаткового відділа для ехолокатора. В даному розділі розглянемо виготовлення верхньої основи.

Розміри деталі: 250x200 мм;

Маса деталі: 0,3 кг;

Матеріал деталі: Алюмінієвий сплав Д1т;

Форма деталі: пластина з фігурним контуром;

Виготовлена “Верхня плита” з алюмінієвого сплаву Д1т.

Алюміній технічної чистоти, що має високу корозійну стійкість. Технічний алюміній (АД0, АД1) застосовують для елементів конструкції та деталей, що не несуть навантаження, коли потрібна висока пластичність, хороша зварюваність, опір корозії та високі тепло- та електропровідність. З них виготовляють різні трубопроводи, палубні надбудови суден, кабелі, електропроводи, шини, фольгу, перегородки в кімнатах, двері, рами, посуд, цистерни для молока.

Таблиця 3.1 — Хімічний склад в % матеріалу

Fe	Si	Mn	Cr	Ti	Al	Cu	Mg	Zn	Домішки
до 0.7	0.2 -	0.4 - 1	До 0.1	до 0.15	91.7 -	3.5 - 4.5	0.4 -	до 0.25	решту кожна 0.05; всього 0.15

Таблиця 3.2 — Механічні властивості матеріалу

Алюміній	σв, МПа	НВ	Густина ρ, кг/м <sup>3</sup>	Модуль пружності, Е·10 <sup>-3</sup> , МПа	Питома теплоємність, Дж/(кг·град)	Коефіцієнт лінійного розширення, (1/Град)	Теплопро- відність, Вт/(м·град)	Питомий електроопір, Ом·м
Д1	315 - 370	95	2800	0.72	922	22.9·10 <sup>6</sup>	130	54·10 <sup>9</sup>

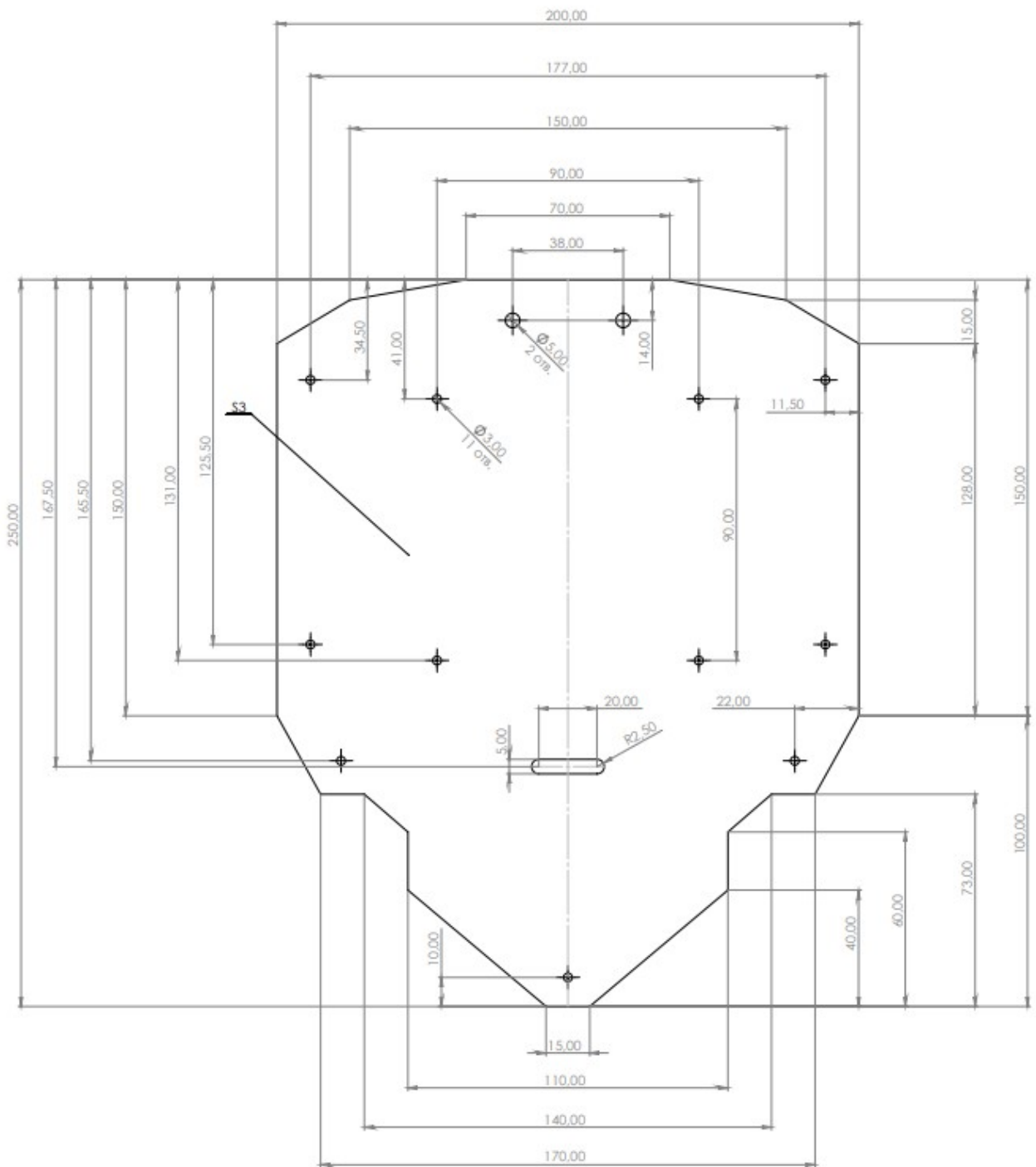


Рисунок 3.1 — Креслення деталі

Зм.	Арк.	№ докум.	Підпис	Дата

МР.ПМКМ-032.00.00.000 ПЗ

Арк.

40

### 3.2 Аналіз технологічності конструкції деталі

Деталь виготовляється з алюмінією, прокатом.

Коефіцієнт оброблюваності матеріалу:

$$KV_M = 11;$$

Деталь постачається вже порізана в необхідний розмір.

Всі поверхні деталі будуть оброблятися на фрезерному верстаті з ЧПК. За точністю обробки - верстат нормальної точності.

Поверхні:

- поверхні 1 отвори, для яких використовуватиметься свердло;
- поверхні 2 отвори, для яких використовуватиметься свердло;
- поверхня 3 отвір, для якого використовуватиметься фреза;
- поверхня 4 торець, для якого використовуватиметься фреза;

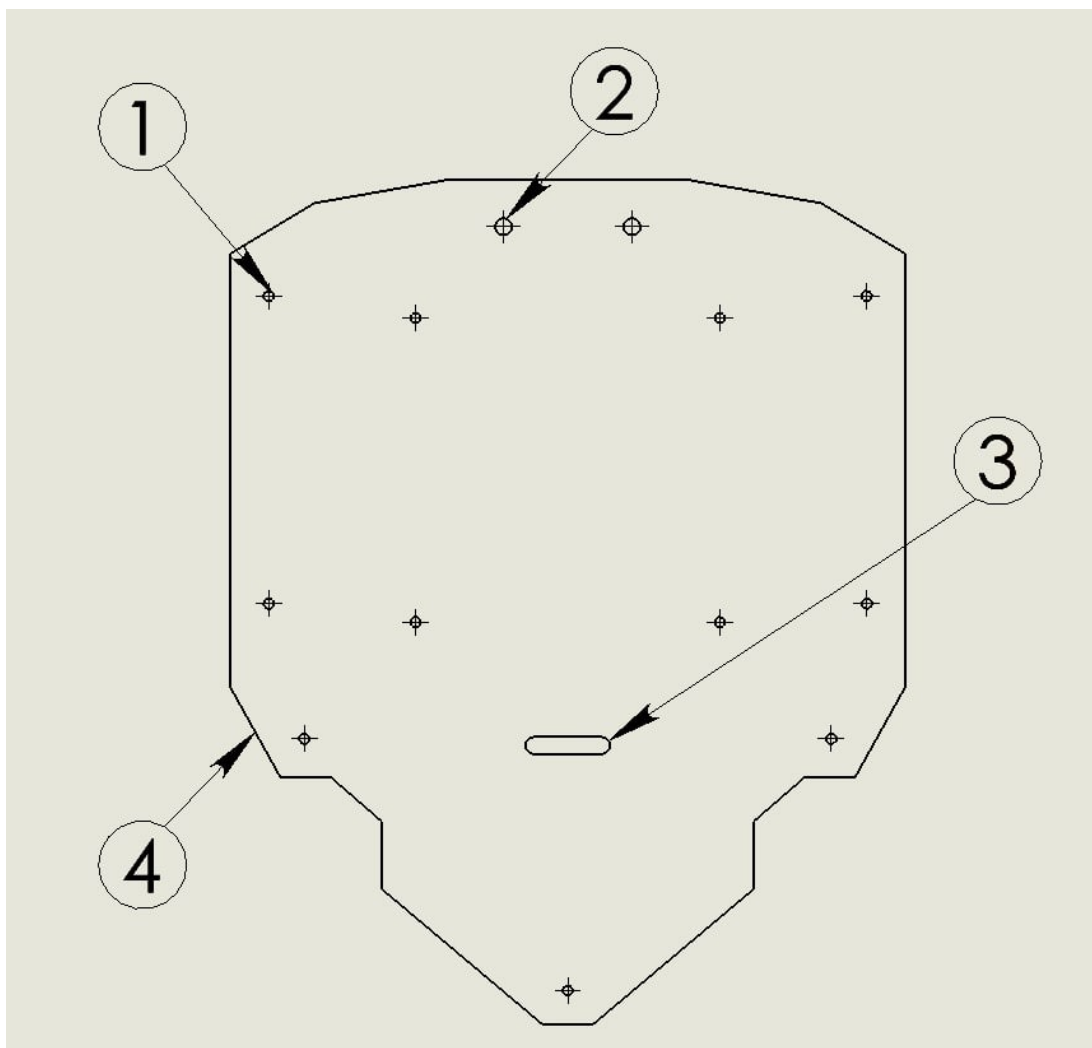


Рисунок 3.2 — Поверхні деталі

Зм.	Арк.	№ докум.	Підпис	Дата

МР.ПМКМ-032.00.00.000 ПЗ

Арк.

41

Таблиця 3.3 — Конструкція деталі

№ Поверхні	Конфігурація та службове призначення	Розміри, мм	Квалітет точності, допус, мм	Шорсткість $R_a$ , мкм
1	Внутрішня циліндрична поверхня	11 отв. Ø3 мм	H9	2,5
2	Внутрішня циліндрична поверхня	2 отв. Ø5 мм	H9	2,5
3	Внутрішня циліндрична поверхня	Паз 5 x 25 мм	H9	2,5
4	Контур	200 x 250 мм	H9	2,5

### 3.3 Опис і аналіз технологічного процесу

Обробка деталі відбувається на фрезерному верстаті з ЧПК 800\*1200\*240 (фрезер чпу), який входить в парк обладнання Центру інноваційного розвитку.



Рисунок 3.3 - 3-х координатний фрезерний верстат з ЧПК

Таблиця 3.5 Характеристики фрезерного верстату ЧПК

Назва параметра	Значення
Робочий діапазон	800x1200x300
Відстань від торця шпинделя до столу, мм	300
Матеріал та конфігурація стола	сталевий з Т-подібними пазами
Шпиндель	3,2 кВт з постійним моментом
Тип цанги	ER20
Максимальні оберти шпинделя, об / хв	22000-28000
Двигун приводу верстата	Кроковий двигун Nema 34 з енкодером для зворотного зв'язку (базова комплектація) комплект на 3 осі
Привід переміщення	ШВП Hiwin (X- 2005, Y- 2005, Z- 1605)
Напрявні	Hiwin HGR (X - 20, Y - 20, Z - 15)

Швидкість холостого ходу	3000мм/хв±20%
Датчик висоти (калібратор)	у комплекті
Датчики «0» верстата (Home)	у комплекті
Програмне забезпечення / Операційна система	NCStudio, Mach3
Комп'ютерний інтерфейс	Ethernet i/або USB
Сумісність програмного забезпечення	Type3 / PowerMill / Artcam
Напруга, В	220
Максимальна споживана потужність, кВт	4
Охолодження інструменту	Система подачі MOP у комплекті

### **Особливості:**

- Безконтактні індуктивні датчики по всіх осях
- Датчик висоти інструменту
- Гофрозахист всіх напрямних
- Всі посадочні місця під направляючі фрезеровані, точність ±0,05 мм
- Власне виробництво

### **Жорсткість конструкції**

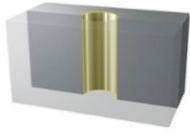
Види оброблюваних матеріалів: деревина будь-яких порід, композитні матеріали (ДСП, ДВП, МДФ, фанера і т. д.), «Alucobond», «Dibond», будь-полістирол (у тому числі ПВХ і полістирол з полікарбонатом), гетинакс, штучний камінь, м'які кольорові метали, акрилове скло, оргскло, модельний пластик і т. п.

Таблиця 3.6 Проектний варіант технологічного процесу

№ оп.	Назва та зміст операції, інструмент	Тип і модель верстата	Характеристик а пристрою, інструмент	Операційний ескіз
010	Фрезерна з ЧПК 1. Встановити заготовку 2. Свердлити групу отворів Ø3Н9	Фрезерний верстат з ЧПК	1. Свердло CoroDrill® 860 Ø3 2. Свердло CoroDrill® 860	



СВЕРЛЕНИЕ СВЕРЛАМИ С  
МОНОЛИТНОЙ РЕЖУЩЕЙ  
ЧАСТЬЮ / ЦЕЛЫМ.



ПОЯСНЕНИЕ

1 Сверление сверлами с симметричной вершиной

CoroDrill 860

860.1-0300-009A1-NM  
H10F  
Инструмент

Соединение  
Cylindrical shank (DIN 1835-A / DIN 6535-  
HA) -metric: 6  
СОЖ  
 Внутренний  
 Эмульсия 10%

Цилиндрическое отверстие в  
цельном материале

N N1.3.C.AG 90 нв

Универсальный  
высокопроизводительный станок  
 ● 200 kW, 10000 1/min  
 ♀ 200 kW, 500000 1/min

ЭКОНОМИЧЕСКАЯ ЭФФЕКТИВНОСТЬ РЕЖИМЫ РЕЗАНИЯ ИЗМЕНИТЕ РЕЖИМЫ РЕЗАНИЯ ПРЕДЕЛЫ  
 CO<sub>2</sub> EMISSIONS **NEW**

VC [m/min] СКОРОСТЬ РЕЗАНИЯ	FN [mm] ПОДАЧА НА ОБОРОТ	N [1/min] ЧАСТОТА ВРАЩЕНИЯ ШПИНДЕЛЯ
1 309	0.25	32800
VF [mm/min] МИНИМАЛЬНАЯ ПОДАЧА У ЦЕНТРА ИНСТРУМЕНТА	PPC [kW] МОЩНОСТЬ РЕЗАНИЯ	MMC [Nm] КРУТЯЩИЙ МОМЕНТ
1 8200	0.922	0.269
FFF [N] УСИЛИЕ ПОДАЧИ	DEPTH [mm] ГЛУБИНА	
1 135	5	

ПОЯСНЕНИЕ

1 Сверление сверлами с симметричной вершиной

Рисунок 3.5 — Режимы резания для сверления

## 2. Сверление группы отверстий Ø5

Твердосплавное сверло CoroDrill® 860

[860.1-0300-009A1-NM H10F \(coromant.com\)](https://www.coromant.com/860.1-0300-009A1-NM-H10F)

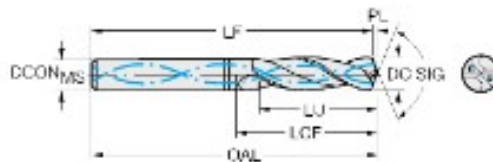


Рисунок 3.6 — Техническая иллюстрация

Диаметр резания (DC) 5 mm

Рабочая длина (LU) 15,7 mm

Точность отвору (ТСНА) H7

Отвѣр наскрѣзный. Глубина отвору 3мм

Режимы резания выбираемо за рекомендацѣями сайта.

Зм.	Арк.	№ докум.	Підпис	Дата

MP.ПМКМ-032.00.00.000 ПЗ

Арк.

46

СВЕРЛЕНИЕ СВЕРЛАМИ С  
МОНОЛИТНОЙ РЕЖУЩЕЙ  
ЧАСТЬЮ / ЦЕЛЬН.



ПОЯСНЕНИЕ

1 Сверление сверлами с симметричной вершиной

CoroDrill 860

860.1-0500-015A1-NM  
N10F  
Инструмент

Соединение  
Cylindrical shank (DIN1835-A / DIN6535-  
NA) - metric: 6

СОЖ

Внутренний  
Эмульсия 10%

Цилиндрическое отверстие в  
цельном материале

N N1.3.C.AG 90 HB

Универсальный  
высокопроизводительный станок  
200 kW, 10000 1/min  
200 kW, 50000 1/min

ЭКОНОМИЧЕСКАЯ ЭФФЕКТИВН...

РЕЖИМЫ РЕЗАНИЯ

ИЗМЕНИТЕ РЕЖИМЫ РЕЗАНИЯ

ПРЕДЕЛЫ

CO<sub>2</sub> EMISSIONS **NEW**

VC [m/min] СКОРОСТЬ РЕЗАНИЯ	FN [mm] ПОДАЧА НА ОБОРОТ	N [1/min] ЧАСТОТА ВРАЩЕНИЯ ШПИНДЕЛЯ
308	0.325	19600

VF [mm/min] МИНУТНАЯ ПОДАЧА У ЦЕНТРА ИНСТРУМЕНТА	PPC [kW] МОЩНОСТЬ РЕЗАНИЯ	MMC [Nm] КРУТЯЩИЙ МОМЕНТ
6360	1.84	0.9

FFF [N] УСИЛИЕ ПОДНИ	DEPTH [mm] ГЛУБИНА
272	5

ПОЯСНЕНИЕ

1 Сверление сверлами с симметричной вершиной

Рисунок 3.7 — Режимы різання для сведління

### 3. Фрезерування контура 250x200 мм

Інструмент 1K335-0400-050-XC 1730 CoroMill® Dura

[1K335-0400-050-XC 1730 \(coromant.com\)](http://1K335-0400-050-XC 1730 (coromant.com))

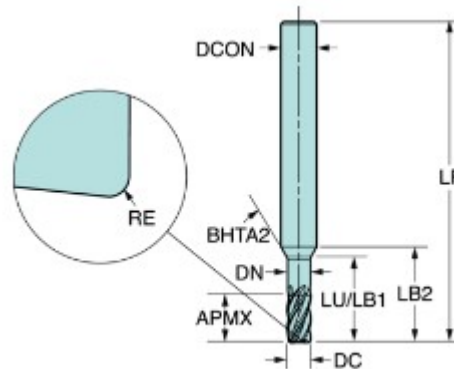


Рисунок 3.8 — Технічна ілюстрація

Діаметр різання ( ) - 4мм;

Діаметр різання по торцю ( ) - 3мм;

Мах грубина різання — 8 мм;

Глибина фрезерування 3мм;

Режими різання підбираєм за рекомендаціями сайту;

Зм.	Арк.	№ докум.	Підпис	Дата

MP.ПМКМ-032.00.00.000 ПЗ

Арк.

47



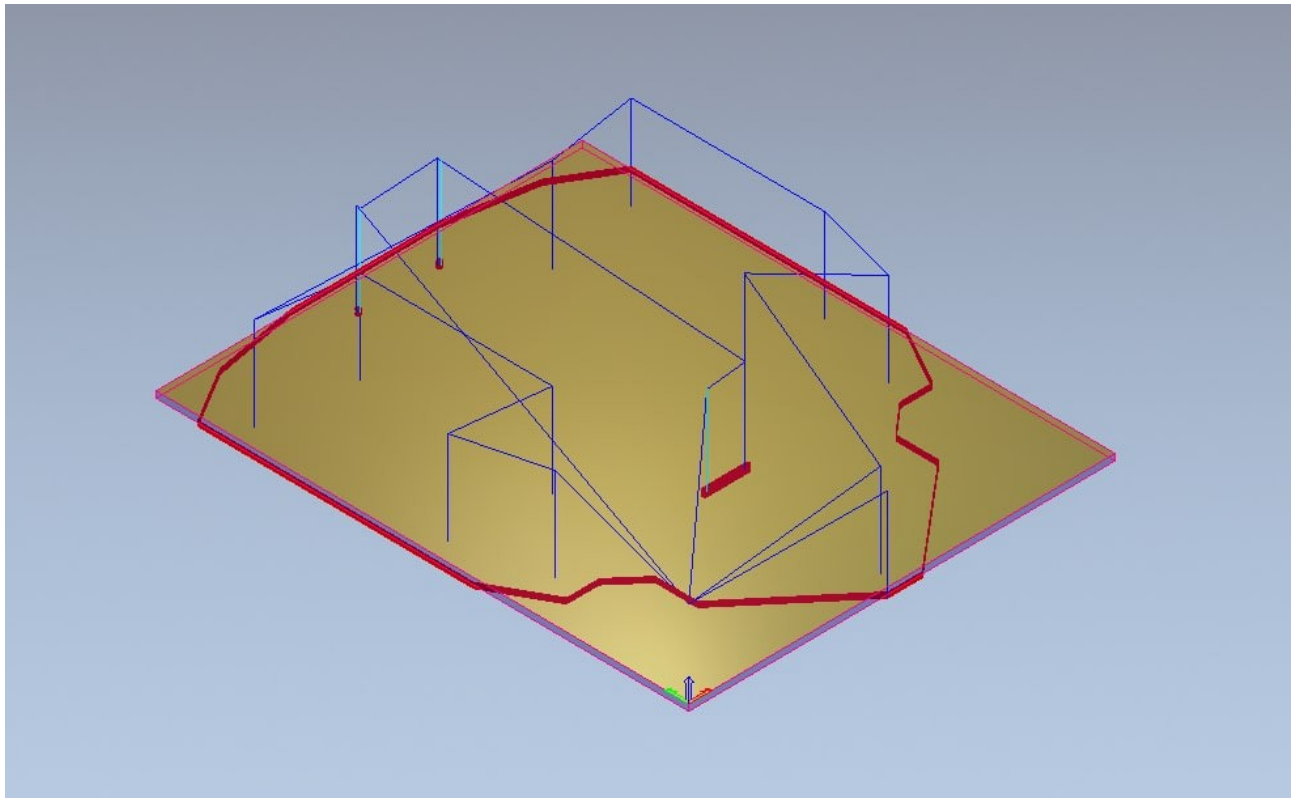


Рисунок 3.10 — Траекторія руху обробки деталі

Програма ЧПК:

%	G0Z41.011	X89.289Y83.204	X110.866Y82.001
TIM6	G0X178.000Y84.500	X89.498Y83.366	X89.134
G0Z41.011	G0Z2.000	X89.742Y83.468	X89.034Y82.242
G0X0.000Y0.000S15000M3	G1Z-3.200F120.0	X89.996Y83.501	X89.000Y82.501
G0X22.000Y84.500Z41.011	G0Z41.011	X110.004	X111.000
G0Z2.000	G0X145.000Y119.000	X110.000Y83.501	X110.966Y82.760
G1Z-3.200F120.0	G0Z2.000	X90.000	X110.866Y83.001
G0Z41.011	G1Z-3.200F120.0	X89.706Y83.457	X89.134
G0X11.500Y124.500	G0Z41.011	X89.419Y83.315	X89.289Y83.204
G0Z2.000	G0X100.000Y10.000	X89.186Y83.082	X89.498Y83.366
G1Z-3.200F120.0	G0Z2.000	X89.044Y82.795	X89.742Y83.468
G0Z41.011	G1Z-3.200F120.0	X89.000Y82.501	X89.996Y83.501
G0X55.000Y119.000	G0Z41.011	X89.044Y82.207	X110.004
G0Z2.000	G0X0.000Y0.000	X89.186Y81.920	X110.000Y83.501
G1Z-3.200F120.0	G0Z41.011	X89.419Y81.687	X90.000
G0Z41.011	G0X0Y0	X89.706Y81.545	X89.706Y83.457
G0Y209.000	M30	X90.000Y81.501	X89.419Y83.315
G0Z2.000	%	X110.000Y81.501	X89.186Y83.082
G1Z-3.200F120.0	TIM6	X110.294Y81.545	X89.044Y82.795
G0Z41.011	G0Z41.011	X110.581Y81.687	X89.000Y82.501
G0X11.500Y215.500	G0X0.000Y0.000S15000M3	X110.814Y81.920	X89.044Y82.207
G0Z2.000	G0X89.996Y81.501Z41.011	X110.956Y82.207	X89.186Y81.920
G1Z-3.200F120.0	G1Z-0.457F120.0	X111.000Y82.501	X89.419Y81.687
G0Z41.011	G1X110.004F720.0	X110.956Y82.794	X89.706Y81.545
G0X145.000Y209.000	X110.258Y81.535	X110.815Y83.081	X90.000Y81.501
G0Z2.000	X110.502Y81.636	X110.583Y83.314	X110.000Y81.501
G1Z-3.200F120.0	X110.711Y81.798	X110.297Y83.456	X110.294Y81.545
G0Z41.011	X110.866Y82.001	X110.004Y83.501	X110.581Y81.687
G0X188.500Y215.500	X89.134	G0Z41.011	X110.814Y81.920
G0Z2.000	X89.034Y82.242	G0X89.996Y81.501	X110.956Y82.207
G1Z-3.200F120.0	X89.000Y82.501	G1Z-0.914F120.0	X111.000Y82.501
G0Z41.011	X111.000	G1X110.004F720.0	X110.956Y82.794
G0Y124.500	X110.966Y82.760	X110.258Y81.535	X110.815Y83.081
G0Z2.000	X110.866Y83.001	X110.502Y81.636	X110.583Y83.314
G1Z-3.200F120.0	X89.134	X110.711Y81.798	X110.297Y83.456

Зм.	Арк.	№ докум.	Підпис	Дата

МР.ПМКМ-032.00.00.000 ПЗ

Арк.

49

X110.004Y83.501	X89.044Y82.795	X89.134	X110.581Y81.687
G0Z41.011	X89.000Y82.501	X89.034Y82.242	X110.814Y81.920
G0X89.996Y81.501	X89.044Y82.207	X89.000Y82.501	X110.956Y82.207
G1Z-1.371F120.0	X89.186Y81.920	X111.000	X111.000Y82.501
G1X110.004F720.0	X89.419Y81.687	X110.966Y82.760	X110.956Y82.794
X110.258Y81.535	X89.706Y81.545	X110.866Y83.001	X110.815Y83.081
X110.502Y81.636	X90.000Y81.501	X89.134	X110.583Y83.314
X110.711Y81.798	X110.000Y81.501	X89.289Y83.204	X110.297Y83.456
X110.866Y82.001	X110.294Y81.545	X89.498Y83.366	X110.004Y83.501
X89.134	X110.581Y81.687	X89.742Y83.468	G0Z41.011
X89.034Y82.242	X110.814Y81.920	X89.996Y83.501	G0X89.996Y81.501
X89.000Y82.501	X110.956Y82.207	X110.004	G1Z-0.457F120.0
X111.000	X111.000Y82.501	X110.000Y83.501	G1X119.141F720.0
X110.966Y82.760	X110.956Y82.794	X90.000	X119.427Y235.096
X110.866Y83.001	X110.815Y83.081	X89.706Y83.457	X119.680Y235.267
X89.134	X110.583Y83.314	X89.419Y83.315	X119.866Y235.500
X89.289Y83.204	X110.297Y83.456	X89.186Y83.082	X118.134
X89.498Y83.366	X110.004Y83.501	X89.044Y82.795	X118.034Y235.741
X89.742Y83.468	G0Z41.011	X89.000Y82.501	X118.000Y236.000
X89.996Y83.501	G0X89.996Y81.501	X89.044Y82.207	X120.000
X110.004	G1Z-2.286F120.0	X89.186Y81.920	X119.966Y236.259
X110.000Y83.501	G1X110.004F720.0	X89.419Y81.687	X119.866Y236.500
X90.000	X110.258Y81.535	X89.706Y81.545	X118.134
X89.706Y83.457	X110.502Y81.636	X90.000Y81.501	X118.320Y236.733
X89.419Y83.315	X110.711Y81.798	X110.000Y81.501	X118.573Y236.904
X89.186Y83.082	X110.866Y82.001	X110.294Y81.545	X118.859Y236.990
X89.044Y82.795	X89.134	X110.581Y81.687	X119.141
X89.000Y82.501	X89.034Y82.242	X110.814Y81.920	X119.071Y236.997
X89.044Y82.207	X89.000Y82.501	X110.956Y82.207	X119.000Y237.000
X89.186Y81.920	X111.000	X111.000Y82.501	X118.706Y236.956
X89.419Y81.687	X110.966Y82.760	X110.956Y82.794	X118.419Y236.814
X89.706Y81.545	X110.866Y83.001	X110.815Y83.081	X118.186Y236.581
X90.000Y81.501	X89.134	X110.583Y83.314	X118.044Y236.294
X110.000Y81.501	X89.289Y83.204	X110.297Y83.456	X118.000Y236.000
X110.294Y81.545	X89.498Y83.366	X110.004Y83.501	X118.044Y235.706
X110.581Y81.687	X89.742Y83.468	G0Z41.011	X118.186Y235.419
X110.814Y81.920	X89.996Y83.501	G0X89.996Y81.501	X118.419Y235.186
X110.956Y82.207	X110.004	G1Z-3.200F120.0	X118.706Y235.044
X111.000Y82.501	X110.000Y83.501	G1X110.004F720.0	X119.000Y235.000
X110.956Y82.794	X90.000	X110.258Y81.535	X119.294Y235.044
X110.815Y83.081	X89.706Y83.457	X110.502Y81.636	X119.581Y235.186
X110.583Y83.314	X89.419Y83.315	X110.711Y81.798	X119.814Y235.419
X110.297Y83.456	X89.186Y83.082	X110.866Y82.001	X119.956Y235.706
X110.004Y83.501	X89.044Y82.795	X89.134	X120.000Y236.000
G0Z41.011	X89.000Y82.501	X89.034Y82.242	X119.963Y236.270
G0X89.996Y81.501	X89.044Y82.207	X89.000Y82.501	X119.844Y236.536
G1Z-1.829F120.0	X89.186Y81.920	X111.000	X119.650Y236.760
G1X110.004F720.0	X89.419Y81.687	X110.966Y82.760	X119.404Y236.915
X110.258Y81.535	X89.706Y81.545	X110.866Y83.001	X119.141Y236.990
X110.502Y81.636	X90.000Y81.501	X89.134	G0Z41.011
X110.711Y81.798	X110.000Y81.501	X89.289Y83.204	G0X118.859Y235.010
X110.866Y82.001	X110.294Y81.545	X89.498Y83.366	G1Z-0.914F120.0
X89.134	X110.581Y81.687	X89.742Y83.468	G1X119.141F720.0
X89.034Y82.242	X110.814Y81.920	X89.996Y83.501	X119.427Y235.096
X89.000Y82.501	X110.956Y82.207	X110.004	X119.680Y235.267
X111.000	X111.000Y82.501	X110.000Y83.501	X119.866Y235.500
X110.966Y82.760	X110.956Y82.794	X90.000	X118.134
X110.866Y83.001	X110.815Y83.081	X89.706Y83.457	X118.034Y235.741
X89.134	X110.583Y83.314	X89.419Y83.315	X118.000Y236.000
X89.289Y83.204	X110.297Y83.456	X89.186Y83.082	X120.000
X89.498Y83.366	X110.004Y83.501	X89.044Y82.795	X119.966Y236.259
X89.742Y83.468	G0Z41.011	X89.000Y82.501	X119.866Y236.500
X89.996Y83.501	G0X89.996Y81.501	X89.044Y82.207	X118.134
X110.004	G1Z-2.743F120.0	X89.186Y81.920	X118.320Y236.733
X110.000Y83.501	G1X110.004F720.0	X89.419Y81.687	X118.573Y236.904
X90.000	X110.258Y81.535	X89.706Y81.545	X118.859Y236.990
X89.706Y83.457	X110.502Y81.636	X90.000Y81.501	X119.141
X89.419Y83.315	X110.711Y81.798	X110.000Y81.501	X119.071Y236.997
X89.186Y83.082	X110.866Y82.001	X110.294Y81.545	X119.000Y237.000

<i>Зм.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>

*МР.ПМКМ-032.00.00.000 ПЗ*

*Арк.*

*50*

X118.706Y236.956	X118.000Y236.000	X119.404Y236.915	X118.419Y235.186
X118.419Y236.814	X120.000	X119.141Y236.990	X118.706Y235.044
X118.186Y236.581	X119.966Y236.259	G0Z41.011	X119.000Y235.000
X118.044Y236.294	X119.866Y236.500	G0X118.859Y235.010	X119.294Y235.044
X118.000Y236.000	X118.134	G1Z-2.743F120.0	X119.581Y235.186
X118.044Y235.706	X118.320Y236.733	G1X119.141F720.0	X119.814Y235.419
X118.186Y235.419	X118.573Y236.904	X119.427Y235.096	X119.956Y235.706
X118.419Y235.186	X118.859Y236.990	X119.680Y235.267	X120.000Y236.000
X118.706Y235.044	X119.141	X119.866Y235.500	X119.963Y236.270
X119.000Y235.000	X119.071Y236.997	X118.134	X119.844Y236.536
X119.294Y235.044	X119.000Y237.000	X118.034Y235.741	X119.650Y236.760
X119.581Y235.186	X118.706Y236.956	X118.000Y236.000	X119.404Y236.915
X119.814Y235.419	X118.419Y236.814	X120.000	X119.141Y236.990
X119.956Y235.706	X118.186Y236.581	X119.966Y236.259	G0Z41.011
X120.000Y236.000	X118.044Y236.294	X119.866Y236.500	G0X80.859Y235.010
X119.963Y236.270	X118.000Y236.000	X118.134	G1Z-0.457F120.0
X119.844Y236.536	X118.044Y235.706	X118.320Y236.733	G1X81.141F720.0
X119.650Y236.760	X118.186Y235.419	X118.573Y236.904	X81.427Y235.096
X119.404Y236.915	X118.419Y235.186	X118.859Y236.990	X81.680Y235.267
X119.141Y236.990	X118.706Y235.044	X119.141	X81.866Y235.500
G0Z41.011	X119.000Y235.000	X119.071Y236.997	X80.134
G0X118.859Y235.010	X119.294Y235.044	X119.000Y237.000	X80.034Y235.741
G1Z-1.371F120.0	X119.581Y235.186	X118.706Y236.956	X80.000Y236.000
G1X119.141F720.0	X119.814Y235.419	X118.419Y236.814	X82.000
X119.427Y235.096	X119.956Y235.706	X118.186Y236.581	X81.966Y236.259
X119.680Y235.267	X120.000Y236.000	X118.044Y236.294	X81.866Y236.500
X119.866Y235.500	X119.963Y236.270	X118.000Y236.000	X80.134
X118.134	X119.844Y236.536	X118.044Y235.706	X80.320Y236.733
X118.034Y235.741	X119.650Y236.760	X118.186Y235.419	X80.573Y236.904
X118.000Y236.000	X119.404Y236.915	X118.419Y235.186	X80.859Y236.990
X120.000	X119.141Y236.990	X118.706Y235.044	X81.141
X119.966Y236.259	G0Z41.011	X119.000Y235.000	X81.071Y236.997
X119.866Y236.500	G0X118.859Y235.010	X119.294Y235.044	X81.000Y237.000
X118.134	G1Z-2.286F120.0	X119.581Y235.186	X80.706Y236.956
X118.320Y236.733	G1X119.141F720.0	X119.814Y235.419	X80.419Y236.814
X118.573Y236.904	X119.427Y235.096	X119.956Y235.706	X80.186Y236.581
X118.859Y236.990	X119.680Y235.267	X120.000Y236.000	X80.044Y236.294
X119.141	X119.866Y235.500	X119.963Y236.270	X80.000Y236.000
X119.071Y236.997	X118.134	X119.844Y236.536	X80.044Y235.706
X119.000Y237.000	X118.034Y235.741	X119.650Y236.760	X80.186Y235.419
X118.706Y236.956	X118.000Y236.000	X119.404Y236.915	X80.419Y235.186
X118.419Y236.814	X120.000	X119.141Y236.990	X80.706Y235.044
X118.186Y236.581	X119.966Y236.259	G0Z41.011	X81.000Y235.000
X118.044Y236.294	X119.866Y236.500	G0X118.859Y235.010	X81.294Y235.044
X118.000Y236.000	X118.134	G1Z-3.200F120.0	X81.581Y235.186
X118.044Y235.706	X118.320Y236.733	G1X119.141F720.0	X81.814Y235.419
X118.186Y235.419	X118.573Y236.904	X119.427Y235.096	X81.956Y235.706
X118.419Y235.186	X118.859Y236.990	X119.680Y235.267	X82.000Y236.000
X118.706Y235.044	X119.141	X119.866Y235.500	X81.963Y236.270
X119.000Y235.000	X119.071Y236.997	X118.134	X81.844Y236.536
X119.294Y235.044	X119.000Y237.000	X118.034Y235.741	X81.650Y236.760
X119.581Y235.186	X118.706Y236.956	X118.000Y236.000	X81.404Y236.915
X119.814Y235.419	X118.419Y236.814	X120.000	X81.141Y236.990
X119.956Y235.706	X118.186Y236.581	X119.966Y236.259	G0Z41.011
X120.000Y236.000	X118.044Y236.294	X119.866Y236.500	G0X80.859Y235.010
X119.963Y236.270	X118.000Y236.000	X118.134	G1Z-0.914F120.0
X119.844Y236.536	X118.044Y235.706	X118.320Y236.733	G1X81.141F720.0
X119.650Y236.760	X118.186Y235.419	X118.573Y236.904	X81.427Y235.096
X119.404Y236.915	X118.419Y235.186	X118.859Y236.990	X81.680Y235.267
X119.141Y236.990	X118.706Y235.044	X119.141	X81.866Y235.500
G0Z41.011	X119.000Y235.000	X119.071Y236.997	X80.134
G0X118.859Y235.010	X119.294Y235.044	X119.000Y237.000	X80.034Y235.741
G1Z-1.829F120.0	X119.581Y235.186	X118.706Y236.956	X80.000Y236.000
G1X119.141F720.0	X119.814Y235.419	X118.419Y236.814	X82.000
X119.427Y235.096	X119.956Y235.706	X118.186Y236.581	X81.966Y236.259
X119.680Y235.267	X120.000Y236.000	X118.044Y236.294	X81.866Y236.500
X119.866Y235.500	X119.963Y236.270	X118.000Y236.000	X80.134
X118.134	X119.844Y236.536	X118.044Y235.706	X80.320Y236.733
X118.034Y235.741	X119.650Y236.760	X118.186Y235.419	X80.573Y236.904

Зм.	Арк.	№ докум.	Підпис	Дата

МР.ПМКМ-032.00.00.000 ПЗ

Арк.

51

X80.859Y236.990	X81.680Y235.267	X82.000Y236.000	X80.044Y236.294
X81.141	X81.866Y235.500	X81.963Y236.270	X80.000Y236.000
X81.071Y236.997	X80.134	X81.844Y236.536	X80.044Y235.706
X81.000Y237.000	X80.034Y235.741	X81.650Y236.760	X80.186Y235.419
X80.706Y236.956	X80.000Y236.000	X81.404Y236.915	X80.419Y235.186
X80.419Y236.814	X82.000	X81.141Y236.990	X80.706Y235.044
X80.186Y236.581	X81.966Y236.259	G0Z41.011	X81.000Y235.000
X80.044Y236.294	X81.866Y236.500	G0X80.859Y235.010	X81.294Y235.044
X80.000Y236.000	X80.134	G1Z-2.743F120.0	X81.581Y235.186
X80.044Y235.706	X80.320Y236.733	G1X81.141F720.0	X81.814Y235.419
X80.186Y235.419	X80.573Y236.904	X81.427Y235.096	X81.956Y235.706
X80.419Y235.186	X80.859Y236.990	X81.680Y235.267	X82.000Y236.000
X80.706Y235.044	X81.141	X81.866Y235.500	X81.963Y236.270
X81.000Y235.000	X81.071Y236.997	X80.134	X81.844Y236.536
X81.294Y235.044	X81.000Y237.000	X80.034Y235.741	X81.650Y236.760
X81.581Y235.186	X80.706Y236.956	X80.000Y236.000	X81.404Y236.915
X81.814Y235.419	X80.419Y236.814	X82.000	X81.141Y236.990
X81.956Y235.706	X80.186Y236.581	X81.966Y236.259	G0Z41.011
X82.000Y236.000	X80.044Y236.294	X81.866Y236.500	G0X0.000Y0.000
X81.963Y236.270	X80.000Y236.000	X80.134	G0Z41.011
X81.844Y236.536	X80.044Y235.706	X80.320Y236.733	G0X0Y0
X81.650Y236.760	X80.186Y235.419	X80.573Y236.904	M30
X81.404Y236.915	X80.419Y235.186	X80.859Y236.990	%
X81.141Y236.990	X80.706Y235.044	X81.141	T1M6
G0Z41.011	X81.000Y235.000	X81.071Y236.997	G0Z41.011
G0X80.859Y235.010	X81.294Y235.044	X81.000Y237.000	G0X0.000Y0.000S15000M3
G1Z-1.371F120.0	X81.581Y235.186	X80.706Y236.956	G0X91.986Y-1.409Z41.011
G1X81.141F720.0	X81.814Y235.419	X80.419Y236.814	G1Z-0.500F120.0
X81.427Y235.096	X81.956Y235.706	X80.186Y236.581	G1X91.748Y-1.298F720.0
X81.680Y235.267	X82.000Y236.000	X80.044Y236.294	X91.534Y-1.147
X81.866Y235.500	X81.963Y236.270	X80.000Y236.000	X44.034Y38.853
X80.134	X81.844Y236.536	X80.044Y235.706	X43.811Y39.085
X80.034Y235.741	X81.650Y236.760	X80.186Y235.419	X43.640Y39.367
X80.000Y236.000	X81.404Y236.915	X80.419Y235.186	X43.535Y39.680
X82.000	X81.141Y236.990	X80.706Y235.044	X43.500Y40.000
X81.966Y236.259	G0Z41.011	X81.000Y235.000	Y59.315
X81.866Y236.500	G0X80.859Y235.010	X81.294Y235.044	X29.440Y71.500
X80.134	G1Z-2.286F120.0	X81.581Y235.186	X15.000
X80.320Y236.733	G1X81.141F720.0	X81.814Y235.419	X14.690Y71.532
X80.573Y236.904	X81.427Y235.096	X81.956Y235.706	X14.384Y71.632
X80.859Y236.990	X81.680Y235.267	X82.000Y236.000	X14.103Y71.797
X81.141	X81.866Y235.500	X81.963Y236.270	X13.867Y72.017
X81.071Y236.997	X80.134	X81.844Y236.536	X13.689Y72.272
X81.000Y237.000	X80.034Y235.741	X81.650Y236.760	X-1.311Y99.272
X80.706Y236.956	X80.000Y236.000	X81.404Y236.915	X-1.452Y99.624
X80.419Y236.814	X82.000	X81.141Y236.990	X-1.500Y100.000
X80.186Y236.581	X81.966Y236.259	G0Z41.011	Y228.000
X80.044Y236.294	X81.866Y236.500	G0X80.859Y235.010	X-1.452Y228.376
X80.000Y236.000	X80.134	G1Z-3.200F120.0	X-1.305Y228.739
X80.044Y235.706	X80.320Y236.733	G1X81.141F720.0	X-1.069Y229.052
X80.186Y235.419	X80.573Y236.904	X81.427Y235.096	X-0.772Y229.286
X80.419Y235.186	X80.859Y236.990	X81.680Y235.267	X24.228Y244.286
X80.706Y235.044	X81.141	X81.866Y235.500	X24.476Y244.406
X81.000Y235.000	X81.071Y236.997	X80.134	X24.741Y244.478
X81.294Y235.044	X81.000Y237.000	X80.034Y235.741	X64.741Y251.478
X81.581Y235.186	X80.706Y236.956	X80.000Y236.000	X135.000Y251.500
X81.814Y235.419	X80.419Y236.814	X82.000	X175.259Y244.478
X81.956Y235.706	X80.186Y236.581	X81.966Y236.259	X175.524Y244.406
X82.000Y236.000	X80.044Y236.294	X81.866Y236.500	X175.772Y244.286
X81.963Y236.270	X80.000Y236.000	X80.134	X200.772Y229.286
X81.844Y236.536	X80.044Y235.706	X80.320Y236.733	X201.069Y229.052
X81.650Y236.760	X80.186Y235.419	X80.573Y236.904	X201.305Y228.739
X81.404Y236.915	X80.419Y235.186	X80.859Y236.990	X201.452Y228.376
X81.141Y236.990	X80.706Y235.044	X81.141	X201.500Y228.000
G0Z41.011	X81.000Y235.000	X81.071Y236.997	Y100.000
G0X80.859Y235.010	X81.294Y235.044	X81.000Y237.000	X201.452Y99.624
G1Z-1.829F120.0	X81.581Y235.186	X80.706Y236.956	X201.311Y99.272
G1X81.141F720.0	X81.814Y235.419	X80.419Y236.814	X186.311Y72.272
X81.427Y235.096	X81.956Y235.706	X80.186Y236.581	X186.133Y72.017

Зм.	Арк.	№ докум.	Підпис	Дата

МР.ПМКМ-032.00.00.000 ПЗ

Арк.

52

X185.897Y71.797	X156.189Y39.085	G1Z-2.000F120.0	X29.440Y71.500
X185.616Y71.632	X155.966Y38.853	G1X91.748Y-1.298F720.0	X15.000
X185.310Y71.532	X108.466Y-1.147	X91.534Y-1.147	X14.690Y71.532
X185.000Y71.500	X108.176Y-1.339	X44.034Y38.853	X14.384Y71.632
X170.560	X107.845Y-1.460	X43.811Y39.085	X14.103Y71.797
X156.500Y59.315	X107.500Y-1.500	X43.640Y39.367	X13.867Y72.017
Y40.000	X92.500	X43.535Y39.680	X13.689Y72.272
X156.465Y39.680	X92.239Y-1.477	X43.500Y40.000	X-1.311Y99.272
X156.360Y39.367	X91.986Y-1.409	Y59.315	X-1.452Y99.624
X156.189Y39.085	G1Z-1.500F120.0	X29.440Y71.500	X-1.500Y100.000
X155.966Y38.853	G1X91.748Y-1.298F720.0	X15.000	Y228.000
X108.466Y-1.147	X91.534Y-1.147	X14.690Y71.532	X-1.452Y228.376
X108.176Y-1.339	X44.034Y38.853	X14.384Y71.632	X-1.305Y228.739
X107.845Y-1.460	X43.811Y39.085	X14.103Y71.797	X-1.069Y229.052
X107.500Y-1.500	X43.640Y39.367	X13.867Y72.017	X-0.772Y229.286
X92.500	X43.535Y39.680	X13.689Y72.272	X24.228Y244.286
X92.239Y-1.477	X43.500Y40.000	X-1.311Y99.272	X24.476Y244.406
X91.986Y-1.409	Y59.315	X-1.452Y99.624	X24.741Y244.478
G1Z-1.000F120.0	X29.440Y71.500	X-1.500Y100.000	X64.741Y251.478
G1X91.748Y-1.298F720.0	X15.000	Y228.000	X135.000Y251.500
X91.534Y-1.147	X14.690Y71.532	X-1.452Y228.376	X175.259Y244.478
X44.034Y38.853	X14.384Y71.632	X-1.305Y228.739	X175.524Y244.406
X43.811Y39.085	X14.103Y71.797	X-1.069Y229.052	X175.772Y244.286
X43.640Y39.367	X13.867Y72.017	X-0.772Y229.286	X200.772Y229.286
X43.535Y39.680	X13.689Y72.272	X24.228Y244.286	X201.069Y229.052
X43.500Y40.000	X-1.311Y99.272	X24.476Y244.406	X201.305Y228.739
Y59.315	X-1.452Y99.624	X24.741Y244.478	X201.452Y228.376
X29.440Y71.500	X-1.500Y100.000	X64.741Y251.478	X201.500Y228.000
X15.000	Y228.000	X135.000Y251.500	Y100.000
X14.690Y71.532	X-1.452Y228.376	X175.259Y244.478	X201.452Y99.624
X14.384Y71.632	X-1.305Y228.739	X175.524Y244.406	X201.311Y99.272
X14.103Y71.797	X-1.069Y229.052	X175.772Y244.286	X186.311Y72.272
X13.867Y72.017	X-0.772Y229.286	X200.772Y229.286	X186.133Y72.017
X13.689Y72.272	X24.228Y244.286	X201.069Y229.052	X185.897Y71.797
X-1.311Y99.272	X24.476Y244.406	X201.305Y228.739	X185.616Y71.632
X-1.452Y99.624	X24.741Y244.478	X201.452Y228.376	X185.310Y71.532
X-1.500Y100.000	X64.741Y251.478	X201.500Y228.000	X185.000Y71.500
Y228.000	X135.000Y251.500	Y100.000	X170.560
X-1.452Y228.376	X175.259Y244.478	X201.452Y99.624	X156.500Y59.315
X-1.305Y228.739	X175.524Y244.406	X201.311Y99.272	Y40.000
X-1.069Y229.052	X175.772Y244.286	X186.311Y72.272	X156.465Y39.680
X-0.772Y229.286	X200.772Y229.286	X186.133Y72.017	X156.360Y39.367
X24.228Y244.286	X201.069Y229.052	X185.897Y71.797	X156.189Y39.085
X24.476Y244.406	X201.305Y228.739	X185.616Y71.632	X155.966Y38.853
X24.741Y244.478	X201.452Y228.376	X185.310Y71.532	X108.466Y-1.147
X64.741Y251.478	X201.500Y228.000	X185.000Y71.500	X108.176Y-1.339
X135.000Y251.500	Y100.000	X170.560	X107.845Y-1.460
X175.259Y244.478	X201.452Y99.624	X156.500Y59.315	X107.500Y-1.500
X175.524Y244.406	X201.311Y99.272	Y40.000	X92.500
X175.772Y244.286	X186.311Y72.272	X156.465Y39.680	X92.239Y-1.477
X200.772Y229.286	X186.133Y72.017	X156.360Y39.367	X91.986Y-1.409
X201.069Y229.052	X185.897Y71.797	X156.189Y39.085	G1Z-3.000F120.0
X201.305Y228.739	X185.616Y71.632	X155.966Y38.853	G1X91.748Y-1.298F720.0
X201.452Y228.376	X185.310Y71.532	X108.466Y-1.147	X91.534Y-1.147
X201.500Y228.000	X185.000Y71.500	X108.176Y-1.339	X44.034Y38.853
Y100.000	X170.560	X107.845Y-1.460	X43.811Y39.085
X201.452Y99.624	X156.500Y59.315	X107.500Y-1.500	X43.640Y39.367
X201.311Y99.272	Y40.000	X92.500	X43.535Y39.680
X186.311Y72.272	X156.465Y39.680	X92.239Y-1.477	X43.500Y40.000
X186.133Y72.017	X156.360Y39.367	X91.986Y-1.409	Y59.315
X185.897Y71.797	X156.189Y39.085	G1Z-2.500F120.0	X29.440Y71.500
X185.616Y71.632	X155.966Y38.853	G1X91.748Y-1.298F720.0	X15.000
X185.310Y71.532	X108.466Y-1.147	X91.534Y-1.147	X14.690Y71.532
X185.000Y71.500	X108.176Y-1.339	X44.034Y38.853	X14.384Y71.632
X170.560	X107.845Y-1.460	X43.811Y39.085	X14.103Y71.797
X156.500Y59.315	X107.500Y-1.500	X43.640Y39.367	X13.867Y72.017
Y40.000	X92.500	X43.535Y39.680	X13.689Y72.272
X156.465Y39.680	X92.239Y-1.477	X43.500Y40.000	X-1.311Y99.272
X156.360Y39.367	X91.986Y-1.409	Y59.315	X-1.452Y99.624

X-1.500Y100.000  
Y228.000  
X-1.452Y228.376  
X-1.305Y228.739  
X-1.069Y229.052  
X-0.772Y229.286  
X24.228Y244.286  
X24.476Y244.406  
X24.741Y244.478  
X64.741Y251.478  
X135.000Y251.500  
X175.259Y244.478

X175.524Y244.406  
X175.772Y244.286  
X200.772Y229.286  
X201.069Y229.052  
X201.305Y228.739  
X201.452Y228.376  
X201.500Y228.000  
Y100.000  
X201.452Y99.624  
X201.311Y99.272  
X186.311Y72.272  
X186.133Y72.017

X185.897Y71.797  
X185.616Y71.632  
X185.310Y71.532  
X185.000Y71.500  
X170.560  
X156.500Y59.315  
Y40.000  
X156.465Y39.680  
X156.360Y39.367  
X156.189Y39.085  
X155.966Y38.853  
X108.466Y-1.147

X108.176Y-1.339  
X107.845Y-1.460  
X107.500Y-1.500  
X92.500  
X92.239Y-1.477  
X91.986Y-1.409  
G0Z41.011  
G0X0.000Y0.000  
G0Z41.011  
G0X0Y0  
M30

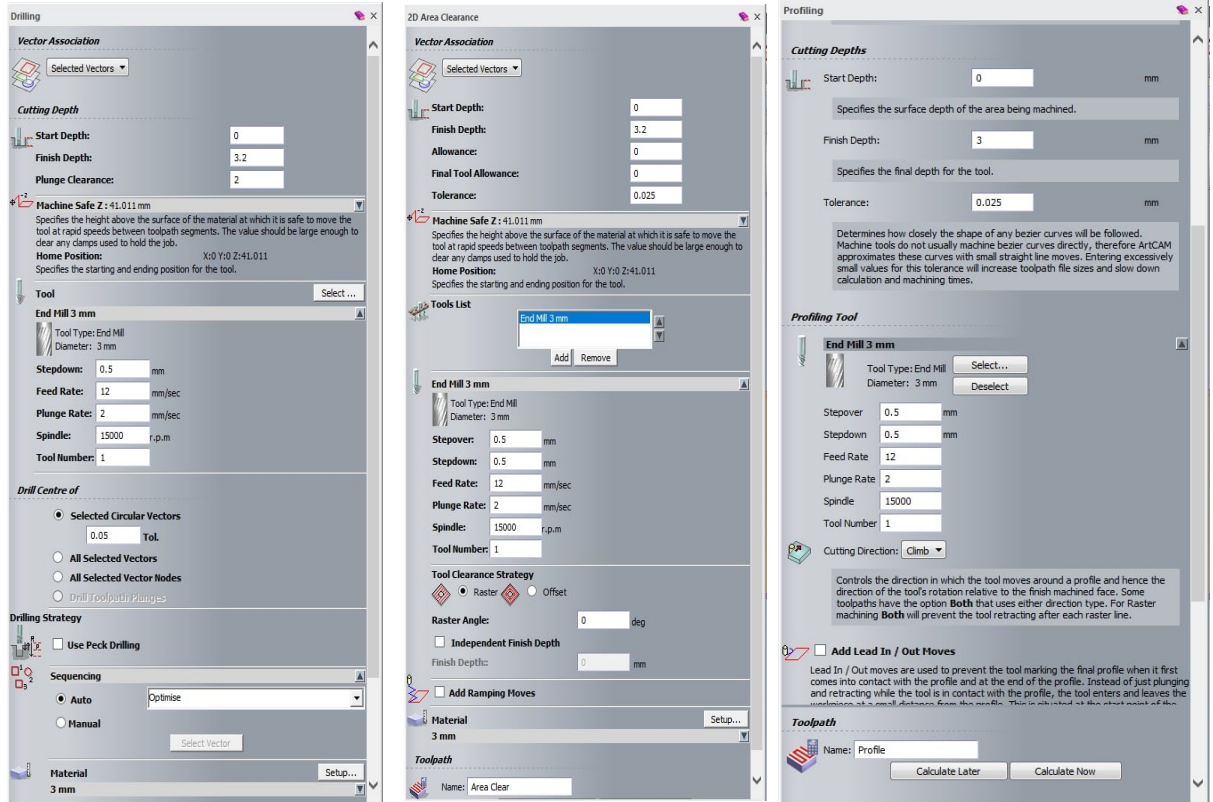


Рисунок 3.11 — Режими різання для всіх операцій

Зм.	Арк.	№ докум.	Підпис	Дата

## 4 ПРОГРАМУВАННЯ ПЛАТФОРМИ

### 4.1 Загальна інформація про програмування Arduino

В мові Arduino не потрібно задавати відправну точку, як в мові Сі, де ви змушені визначити основну програму. Дійсно, під час запуску програми, вона завантажує те, що називається об'єкт в об'єктно-орієнтованому програмуванні ООП в пам'ять. Об'єкти в ООП належать класам. Для кожного об'єкта того ж класу стан визначається у вигляді набору атрибутів або елементів, котрі можна розглядати як змінні, представлені як набір даних різних типів в пам'яті. Стан об'єкта може управлятись методами: набір станів, призначених для виконання заданої операції по зміні або забезпеченню стану об'єкта. Якщо ви не знайомі з ООП, ви можете роз'яснити для себе, що стан це набір змінних та методи, це набір функцій.

Скетч Arduino складається з самостійного файла, в якому, на відміну від мови С, Ви повинні визначити принаймні, дві секції: перша називається `setup()`, а друга `loop()`. Змінні, доступні з обох секцій програми, повинні бути оголошені за їх межами, як глобальні змінні.

Як тільки програма запуститься, виконуються операнди, розміщені в блоці `setup()`: вони призначені для ініціалізації значень змінних на початку запуску, а також для налаштування портів периферії Arduino. Після закінчення обробки `setup()` Arduino починає циклічне виконання інструкцій в блоці `loop()`. Після виконання всіх операндів, цикл повторюється знову і знову.

### Ввод-вивід даних з Arduino.

В Arduino нема портів для підключення до монітора чи клавіатури: хоча вони і не обов'язкові на таких типах пристроїв. Контакти вводу / виводу призначені для забезпечення обміну даними та сигналами вводу / виводу з зовнішнім світом. Однак це не самодостатні порти, до яких ми звикли, працюючи з комп'ютерами, а окремі сигнальні виводи. А в тих випадках, коли

					<i>MP.ПМКМ-032.00.00.000 ПЗ</i>	<i>Арк.</i>
<i>Зм.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		55



- Завантажити - завантажити програму в Arduino;
- Завантажити за допомогою програматора - завантажити програму за допомогою програматора;
- Налаштування друку – налаштування принтера;
- Друк - вивід на друк коду програми;
- Налаштування – налаштування редактора;
- Вихід - вихід з Arduino IDE.

Пункт меню Правка містить команди, пов'язані з редагування текст програми, включаючи копіювання, вставку, настройку відступів і пошук за ключовим словом.

У розділі Скетч розміщуються команди для контролю за процесом компіляції програми.

- Перевірити / Компілювати - компілювати програму;
- Показати папку скетчів - відкрити системну папку з програмами;
- Додати файл - додати до проекту файл з даними або програмою;
- Імпортувати бібліотеку - підключити до програми бібліотеку зі списку встановлених.






Пункт меню Сервіс включає в себе допоміжні функції для роботи з самим мікроконтролером.

- Автоформатування - автоматична розстановка відступів, переносів рядків і т.п .;

- Архівувати скетч - архівація папки з програмою, і збереження архіву в вказане місце;
- Виправити - кодування і перезавантажити
- Монітор порту - відкрити вікно для обміну даними з мікроконтролером;
- Плата - вибір поточної плати (в даному випадку Arduino Uno);
- Послідовний порт - вибір порту, до якого підключений;

Нарешті, меню Довідка містить докладний опис всіх функцій самого редактора Arduino IDE, а також всілякі команди і прийоми роботи з платформою Arduino.

### Меню іконок

-  Перевірити / Компілювати програму
-  Завантажити програму в Arduino
-  Створити нову програму
-  Відкрити існуючу програму
-  Зберегти програму
-  Монітор послідовного порту

### Вкладки

Кожна програма для Arduino може складатися з декількох файлів. Для перемикання між цими файлами служить система вкладок в редакторі. Там же, можна створити нову вкладку, і асоціювати з нею файл в папці з проектом.

### Вікно програми

Безпосередньо, текст програми створюється і редагується в головному

					<i>MP.ПМКм-032.00.00.000 ПЗ</i>	<i>Арк.</i>
<i>Зм.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		58

вікні редактора. По суті, вікно редактора являє собою типовий текстовий редактор, з підсвічуванням конструкцій коду.

### Вікно повідомлень

У самому низу редактора Arduino IDE є невелике вікно, що служить для виведення повідомлень про проблеми, які виникають в процесі компіляції програми, або під час завантаження програми в мікроконтролер.

## 4.3 Програма для пересування платформи

Як і в будь якій мові програмування для початку потрібно описати константи та будь які змінні які будуть використовуватись. Для опису констант потрібно враховувати існуючі входи в мікросхемі Arduino UNO. Також звісно необхідно пам'ятати про додаткові бібліотеки які допомагають використовувати додаткові команди для роботи з комплектуючими.

### 4.3.1 Константи для підключення до входів Arduino UNO

Так першою командою в нашому коді ми підключаємо додаткову бібліотеку для роботи з ехолокатором або як ми називаємо його в нашому проєкті, датчиком виміру відстані (рис. 4.1).

```
//          БІБЛІОТЕКА для роботи датчика ехо локатора  
#include   <iarduino_HC_SR04_int.h>
```

Рисунок 4.1 — Підключення бібліотеки ехолокатора

### Датчики лінії

Так як у нас присутні три датчики лінії потрібно на кожен з них описати свою константу. Наприклад датчик який знаходиться зліва називатиметься “datLL”. Дані датчики повинні підключатись до аналогових виходів мікросхеми тому присвоємо їм значення будь яких окремих виходів (рис. 4.2).

```

const uint8_t    datLL = A3;           // Датчик лінії, який знаходиться зліва
const uint8_t    datLC = A4;           // Датчик лінії, який знаходиться посередині
const uint8_t    datLR = A5;           // Датчик лінії, який знаходиться зправа

```

Рисунок 4.2 — Константи для датчиків лінії (A3, A4, A5 — номери конкретних аналогових виходів)

### Датчик відстані (ехолокатор)

Для підключення до мікросхеми потрібно враховувати два виходи які присутні на даному датчику, а саме “ECHO” та “TRIG”. Ці виходи і визначають принцип роботи датчика. Їх ми підключаємо до цифрових входів мікросхеми. На платі Arduino UNO цифровими входами являються 2 та 3. Отримуємо дві константи для ехолотатора (рис. 4.3).

```

const uint8_t    datV_TRIG = 2;        // Вихід ехолотатора TRIG
const uint8_t    datV_ECHO = 3;        // Вихід ехолотатора ECHO

```

Рисунок 4.3— Константи для виходів “TRIG” та “ECHO”

Також потрібно об’явити об’єкт з яким ми працюємо, а саме ехолотатор. Для цього підтягуємо уже підключену бібліотеку та вказуємо які значення відносяться до даного об’єкту (рис. 4.4).

```

//
#include <ArduinoHCSR04.h>              // Для роботи з бібліотекою потрібно вказати об’єкт HCSR (і виходи ехолотатора)

```

Рисунок 4.4 — Оголошення об’єкту для роботи з бібліотекою

### Моторшилд

Даний компонент включає в собі чотири виходи і потрібний він для коректного керування мотор-редукторами. Тому буде використано два виходи на лівий мотор та два виходи на правий мотор. Для обох мотор один з виходів відповідатиме за ШІМ (широко імпульсну модуляцію). Всі виходи підключаємо до будь яких входів плати. В результаті отримуємо чотири константи (рис. 4.5).

```

const uint8_t    motSh_LH = 7;           // Вихід напрямлення до якого підключений лівий мотор
const uint8_t    motSh_LE = 6;           // Вихід ШІМ(широтна імпульсна модуляція) до якого підключений лівий мотор
const uint8_t    motSh_RE = 5;           // Вихід ШІМ до якого підключений правий мотор
const uint8_t    motSh_RH = 4;           // Вихід напрямлення до якого підключений правий мотор

```

Рисунок 4.5 — Константи для виходів моторшилда

### 4.3.2 Константи для конкретно встановлених значень

Для початку потрібно встановити показники датчиків лінії. Потрібно підключити до плати Arduino UNO один датчик, а саму плату під'єднати до комп'ютера. Наводячи датчик на лінію траєкторії руху та навколишнє середовище, що оточує лінію, встановлюємо два значення які відповідатимуть за пересування платформи відповідно до траєкторії (п. 4 “Калібрування”). Результат записуємо в константи (рис. 4.6).

```

const uint16_t   datLOL = 930;           // Дані які датчик показує на лінії (це значення потрібно настроїти)
const uint16_t   datLOOL = 730;         // Дані які датчик показує поза лінією (це значення потрібно настроїти)

```

Рисунок 4.6 — Показники датчика лінії

Встановлюємо максимальна швидкість переміщення платформи. Вказується в значенні від 0 до 255 задля плавного пришвидшення або сповільнення (рис. 4.7).

```

const uint8_t    MAXSP = 255;

```

Рисунок 4.7 — Максимальна швидкість переміщення

Задля коректної та більш гнучкої роботи платформи потрібно додатково встановити значення яке допомагатиме платформі продовжувати роботу навіть при виїзді поза лінію траєкторії для можливості повернення її на траєкторію. Для цього встановимо час який відповідатиме за роботу платформи поза траєкторією (рис. 4.8).

```
const uint32_t    TimerST = 2000;
```

Рисунок 4.8 — Час затримки перед повною зупинкою платформи

Також потрібно вказати значення яке відповідатиме за швидкість реагування на поворот. Це робиться задля плавності руху, щоб платформа не реагувала на маневр занадто різко. Значення встановлюється від 0 до 255 шляхом підбирання значення. Для нашої платформи оптимальним значенням являється 10 (рис. 4.9).

```
const uint8_t    Povорот = 10;
```

Рисунок 4.9 — Коефіцієнт плавності повороту

Потрібно встановити відстань яка відповідатиме за зупинку платформи перед перешкодою. Значення також підбирається шляхом досліду. Нехай даний коефіцієнт становитиме 10 см (рис. 4.10).

```
const uint8_t    Vidst = 10;
```

Рисунок 4.10 — Коефіцієнт відстані до об'єкту

Далі потрібно встановити напрямок руху двигунів. Це значення залежить від полярності тобто від підключення двигунів. Для нього доцільно використовувати булеві значення (рис. 4.11).

```
const bool    OborotMot[2] = {1,1};
```

Рисунок 4.11 — Константа напрямку руху двигунів

### 4.3.3 Встановлення значень, які обраховуватимуться

Почнемо зі значень швидкості на двигунах. Для цього використовуватимемо масив з двох значень де 0 означатиме правий мотор, а 1

					<i>МР.ПМКМ-032.00.00.000 ПЗ</i>	<i>Арк.</i>
<i>Зм.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		62

— лівий мотор (рис. 4.12).

```
uint8_t      MotorSP[2];
```

Рисунок 4.12— Значення швидкості двигунів

Середнє значення та константа побітного виводу значень датчиків лінії.  
(рис. 4.13).

```
uint16_t     datLMID;  
uint8_t      datLBIT;
```

Рисунок 4.13— Середнє значення та константа побітного виводу значень датчиків лінії

Далі потрібно становити так звані прапорці тобто триггери, які викликатимуть спрацювання коду, який в свою чергу регулюватиме роботу платформи. Наприклад

- Прапорець, який характеризуватиме лінію, а саме її колір (0 — темна лінія, 1 - світла лінії);
- Прапорець, який характеризуватиме ШІМ
- Прапорець, який характеризуватиме зупинку (0 - без зупинки, 1 — зупинка)
- Прапорець, який характеризуватиме відстань до перешкоди (0 - нема перешкоди, 1 - є перешкода)

Всі ці значення використовуватимуться в булевій формі (рис. 4.14).

```
bool      flgLine;
bool      flgPWM;
bool      flgStop;
bool      flgDistance;
```

Рисунок 4.14 — Значення прапорців які використовуються в булевій формі

Потрібен також прапорець для розпізнавання напрямку повороту. Для цього скористаємось простими числами, які характеризуватимуть напрям повороту (0 - прямо їдем, -1 - наліво, +1 — направо) (рис. 4.15).

```
int8_t    flgTurning;
```

Рисунок 4.15 — Значення, яке характеризуватиме напрям повороту

Залишається тільки константа, яка відобразатиме час впродовж якого була проведена попередня операція (рис. 4.16).

```
uint32_t  TimerML;
```

Рисунок 4.16 — Час впродовж якого проведена попередня операція

### 4.3.4 Основа програми

Як ми знаєм програма для ардуїно ділиться на два основних модулі. Перший модуль який винонується лиш один раз та модуль який являє собою неперервний цикл. Тож після закінчення опису констант потрібно почати з модулю який виконується лиш один раз. Цей модуль відповідає за встановлення та присвоєння значень до конкретних відділів програми.

#### Модуль Void\_Setup

Почнем з основи, а саме розпізнавання лінії. Для того, щоб зробити

платформу більш автоматичну та адаптивну було вирішено написати складніший код, щоб платформа могла пересуватись визначаючи як темну лінію на світлому фоні так і світлу лінію на темному фоні. Використаємо для визначення просте порівняння та встановимо значення прапорця кольору лінії відповідно до порівняння значень датчика поза лінією та на лінії, якщо значення за лінією більше тоді буде встановлюватись 1, а це означає, що ми пересуваємся по світлій лінії на темному фоні, якщо навпаки, то буде скидатись на 0 і це означатиме темна лінія на світлому фоні. В залежності від цього значення буде обраховуватись середнє значення показників датчика тому потрібні різні формули для обрахування (рис. 4.17).

```
void setup(){
// Розпізнавання кольору лінії
// flgLine = (datLOOL>datLOL); // Якщо значення за лінією
// // Якщо навпаки, то буде 0
// Визначаєм середнє значення показників датчика лінії (на ній і поза нею)
// Так як у нас може їздити по темній лінії на світлому фоні і навпаки потрібні різні
if(flgLine){datLMID = datLOL + (datLOOL - datLOL)/2;} // Для світлої лінії
else {datLMID = datLOOL + (datLOL - datLOOL)/2;} // Для темної лінії
```

Рисунок 4.17 — Визначення контрасту на якому пересувається платформа

Далі потрібно присвоїти конкретні значення швидкості мотор редукторам (рис. 4.18).

```
// Присвоюєм моторам конкретні значення швидкості (1 - лівий мотор; 0 - правий мотор)
MotorSP[1] = MAXSP;
MotorSP[0] = MAXSP;
```

Рисунок 4.18 — Присвоєння значення швидкостей мотор-редукторами

Для коректної роботи потрібно налаштувати початкові значення. Для цього присвоїмо значення для окремих прапорців:

- для значення ШІМ — 1; щоб показати, що потрібно змінити значення

										Арк.
										65
Зм.	Арк.	№ докум.	Підпис	Дата	MP.ПМКм-032.00.00.000 ПЗ					

ШИМ;

- для значення повороту — 0;
- для значення відстані — 0;
- для значення зупинки — 0;

Скинувши попередні дані потрібно налаштувати спрацювання виходів плати. Це робиться для отримання чи зчитування даних.

Спочатку виходи на датчики лінії. Для них всі виходи налаштовуємо як вхід. Для отримання інформації з них.

Далі виходи мотор-редукторів встановлюємо на вихід. Для керування напрямленням обертів та швидкості.

Для направлення мотор-редукторів потрібно ще встановити значення яке подаватиме напругу і обертатиме в якусь сторону двигуни. Для цього використовується команда “digitalWrite”.

Залишилось встановити затримку. Це робиться для того щоб платформа не починала працювати зразу як тільки була підключена до живлення. В нашому випадку встановлюється значення в 2 секунди. А також запишимо час за який виконувалась попередня операція. Це значення в подальшому використовуватиметься в програмі.



```
datLBIT = 0;
datLBIT |= ((analogRead(datLL)>datLMID)^flgLine)<<2;
datLBIT |= ((analogRead(datLC)>datLMID)^flgLine)<<1;
datLBIT |= ((analogRead(datLR)>datLMID)^flgLine)<<0;
```

Рисунок 4. 20 — Присвоєння бітових значень конкретним датчикам

Провівши попередні операції ми отримуємо варіацію різних випадків:

- 000 — Всі датчики поза лінією;
- 010 — Середні датчик на лінії, інші поза лінією;
- 011 — Лівий поза лінією, інші на лінії;
- 111 — Всі датчки на лінії;
- 100 — Лівий датчик на лінії, інші поза лінією;
- 001 — Правий датчик на лінії;

Опишемо всі ці випадки в програмі та запишемо дії які, платформа повинна виконувати при якомусь з цих випадків. Наприклад, середній датчик на лінії, а інші поза лінією. В цьому випадку потрібно скинути прапорець повороту та прапорець зупинки, а також встановити прапорець ШІМ, якщо до цього відбувався поворот. Далі випадки коли правий, правий і середній на лінії або лівий, лівий і середній на лінії. Ці випадки схожі в дії відрізнятиметься лише прапорець повороту, коли поворот відбуватиметься вліво прапорець встановлюється 1, якщо вправо, то -1. Також в обох випадках прапорець зупинки дорівнюватиме нулю, та прапорець ШІМ дорівнюватиме 1, так як відбувається поворот.

Залишаються випадки в яких дії є невизначеними це 000 та 101. При першому потрібно всього навсього визначити чи відбувався поворот до цього чи ні. Якщо не відбувався поворот, то просто прапорець зупинки встановити

					<i>MP.ПМКм-032.00.00.000 ПЗ</i>	<i>Арк.</i>
<i>Зм.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		68

нулю. Якщо поворот відбувався тоді не потрібно зупиняти, бо в цьому випадку платформа могла просто випадково виїхати за лінію і тоді робот буде шукати лінію по якій йому їхати.

Також є додаткові випадки які не залежать від значення показників датчиків. До цих випадків відносять

- Платформа перебувала поза лінією більше двох секунд;
- Ультразвуковий датчик бачить перешкоду;

В першому випадку потрібно порівняти значення часу проведення попередньої дії з часом який є сталим, тобто дві секунди. Якщо цей час більший тоді потрібно провести зупинку.

В другому випадку порівняти значення показника ехолотатора з сталим значенням в 10 сантиметрів. Якщо значення ехолотатора менше за стале тоді проводитиметься зупинка. Додатково добавимо відгалуження, яке означатиме, якщо перешкода віддалилась, або пропала тоді продовжити рух. (рис. 4.21).

```
switch(datLBIT){
  case 0b000:          flgPWM=flgTurning;          flgStop=!flgTurning; break;
  case 0b010:          TimerML=millis();          flgPWM=flgTurning; flgTurning=0;          flgStop=0;          break;
  case 0b100:
  case 0b110:          TimerML=millis();          flgPWM=1;          flgTurning=-1;          flgStop=0;          break;
  case 0b001:
  case 0b011:          TimerML=millis();          flgPWM=1;          flgTurning=1;          flgStop=0;          break;
  default:            flgPWM=1;          break;
}
if(TimerML>millis())    { TimerML=0;}
if(TimerML+TimerST<millis()){          flgPWM=1;          flgTurning=0;          flgStop=1;          }
if(HCSR.distance()<Vidst){ TimerML=millis(); flgPWM=1;          flgDistance=1;          }
else if(flDistance)    { TimerML=millis(); flgPWM=1;          flgDistance=0;          }
```

Рисунок 4.21 — Випадки за яких платформа матиме різну поведінку

Залишилось описати керування моторами з використанням ШІМ. Всі випадки будуть відбуватись тільки тоді коли прапорець ШІМ встановлений,

тобто дорівнюватиме одиниці.

Опишемо поведінку моторів відповідно значення прапорця повороту.

Якщо прапорець дорівнюватиме

- -1 — Тоді зменшуватимемо значення максимальної швидкості на лівому моторі з кожним виконанням циклом, а значення на правому моторі встановлюємо максимальним. Відповідно поворот відбуватиметься вліво.
- 0 — Тоді вставляємо на обох моторах максимальне значення. Відповідно рух відбуватиметься прямо.
- 1 — Тоді зменшуємо значення максимальної швидкості на правому моторі з кожним виконанням циклом, а значення на лівому моторі встановлюємо максимальним. Відповідно поворот відбуватиметься вправо.

Також потрібно додати відношення швидкостей до прапорців зупинки та відстані. Тобто якщо встановлений будь який з цих прапорців тоді швидкість на обох моторах дорівнюватиме нулю.

На завершення просто виводимо отримані значення на мотори. (рис. 4.22).

```
// Встановлюємо ШІМ для моторів
if(flgPWM){flgPWM=0;
  switch(flgTurning){ // Швидкість лівого мотору:           Швидкість правого мотору:
    case -1:          if(MotorSP[1]>0){MotorSP[1]--;}      MotorSP[0]=MAXSP;           break;
    case 0:           MotorSP[1]=MAXSP;                    MotorSP[0]=MAXSP;           break;
    case 1:           MotorSP[1]=MAXSP;                    if(MotorSP[0]>0){MotorSP[0]--;} break;
  } if(flgStop){      MotorSP[1]=0;                        MotorSP[0]=0;}
  if(flgDistance){   MotorSP[1]=0;                        MotorSP[0]=0;}
// Обраховані значення передаємо на вхід до моторів
  analogWrite(motSh_LE, MotorSP[1]);
  analogWrite(motSh_RE, MotorSP[0]);
```

Рисунок 4.22 — Встановлення значень ШІМ для мотор-редукторів

									Арк.
									70
Зм.	Арк.	№ докум.	Підпис	Дата					



Для модуля **Void\_Loop** потрібно описати яку кількість кроків робитиме наш двигун. Для користуватимемось відправними точками. Нам потрібно, щоб двигун починав працювати тільки тоді коли платформа зупинена. Для цього введемо умову, що коли швидкості на обох мотор-редукторах, прапорець зупинки дорівнюють нулю, тоді виконувати піднімання на задану кількість кроків. (рис. 4.24). Використаємо команду, яка задає швидкість обертання та запишемо кількість кроків яку двигун повинен виконати. Також встановимо затримку в 2 секунди.

```
void loop() {  
  // put your main code here, to run repeatedly:  
  steppermotor.setSpeed(1);  
  StepsRequired = 4;  
  steppermotor.step(StepsRequired);  
  delay(2000);  
}
```

Рисунок 4.24 — Підняття крокового двигуна на задану кількість кроків

Загальний вигляд програми див. Додаток Б.

#### 4.4 Калібрування платформи

Щоб платформа працювала коректно потрібно налаштувати вихідні дані, тобто відкалібрувати вносячи дані отримані з оточення по якому платформа пересуватиметься.

Створивши траєкторію руху потрібно зняти показники середнього датчика лінії. Навівши датчик на лінії яка являє собою траєкторію потрібно підключити плату Arduino UNO до комп'ютера та запустити в середовищі Arduino IDE монітор послідовного порту в якому буде виведено значення датчика лінії. Дане значення потрібно занести в константу, яка описана в програмі. Далі проводим ті ж операції тільки для середовища, яке оточує лінію.

									Арк.
									72
Зм.	Арк.	№ докум.	Підпис	Дата	MP.ПМКм-032.00.00.000 ПЗ				

Дані занесені значення потрібно помістити в пам'ять Arduino Uno для цього потрібно прошити плату з новими показниками.

#### 4.5 Симуляція роботи платформи

Задля імітації процесу взаємодії роботів захватів та перевірки суміжності їхньої роботи, створений макет розташування роботів захваті та транспортного робота (рис. 4.25).

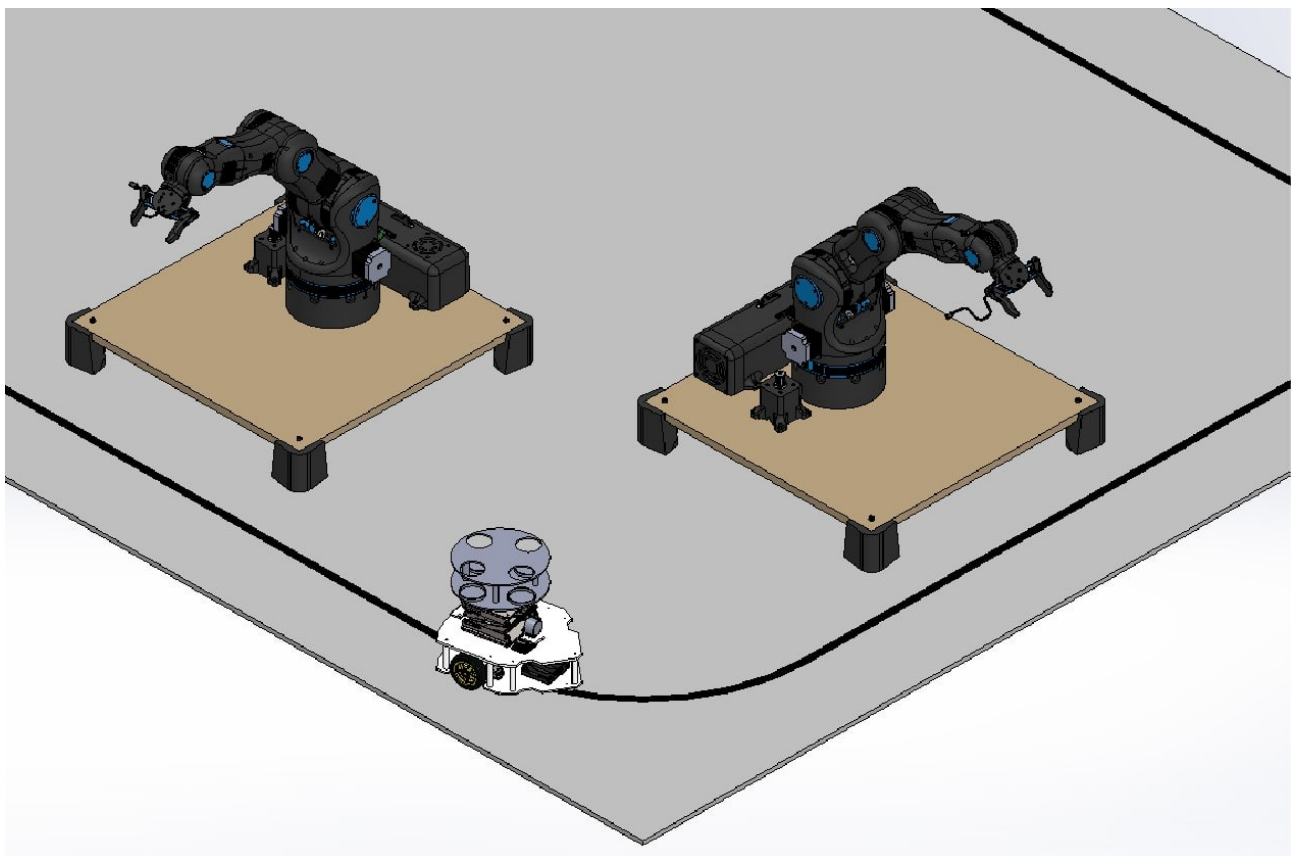


Рисунок 4.25 — Симуляція взаємодії між собою роботів

Зм.	Арк.	№ докум.	Підпис	Дата



## Список використаної літератури

1. О. Р. Онисько, В. Г. Панчук “Основи програмного керування технологічними системами” Навчальний посібник. Івано- Франківськ 2014. (302 ст.).
2. Панчук, В. Г. Засоби програмного керування. Програмування ПЛК FATEK у середовищі WinProLadder : лабораторний практикум / В. Г. Панчук, О. Р. Онисько, Т. В. Лукань. - Івано-Франківськ : ІФНТУНГ, 2018. - 65 с
3. [Джереми Блум Изучаем Arduino- инструменты и методы технического волшебства](#)
4. Петин В.А., Биняковский А.А. Практическая энциклопедия Arduino
5. "Робототехнические комплексы" Під редакцією Б.І.Черпакова, 2002 р.
6. [Транспортний робот — Wiki ТНТУ \(tntu.edu.ua\)](#)
7. [Arduino основи програмування \(geekmatic.in.ua\)](#)
8. [Програмування Arduino - Digit Logic \(digit-logic.com.ua\)](#)
9. [Програмування ARDUINO. Створюємо практичні пристрої віртуальний диск. купити в Києві та Україні](#)
10. [Arduino для школярів. Програмування \(naurok.com.ua\)](#)

					<i>МР.ПМКм-032.00.00.000 ПЗ</i>	<i>Арк.</i>
<i>Зм.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		75

# Додатки

Форм.	Зона	Поз.	Обозначение	Наименование	Кол.	Примеч.
				<u>Документація</u>		
A1			MP.ПМКм-032.02.00.000 СК	Складальне креслення	1	
				<u>Складальні одиниці</u>		
		1	MP.ПМКм-032.02.01.000	Домкрат	1	
		2	MP.ПМКм-032.02.02.000	Поворотний стіл	1	
				<u>Деталі</u>		
		3	MP.ПМКм-032.02.00.003	Основа нижня	1	
		4	MP.ПМКм-032.02.00.004	Основа верхня	1	
A4		5	MP.ПМКм-032.02.00.005	Кріплення ехолокатора	1	
A4		6	MP.ПМКм-032.02.00.006	Кріплення мотора	2	
A4		7	MP.ПМКм-032.02.00.007	Опора	11	
				<u>Стандартні деталі</u>		
		8		Плата Arduino UNO R3	1	
		9		Motor Shield L298P	1	
		10		HC-SR04	1	

<b>MP.ПМКм-032.02.00.000</b>				
Изм	Лист	№ докум	Подп.	Дата
Разраб.	Веркалець А.А.			
Пров.	Лукаць Т.В.			
Н.контр.				
Утв.				
<b>Транспортна платформа</b>			Лит.	Лист
				Листов
				1
			<b>ІФНТУНГ</b>	
			<b>ПМКм-21-1</b>	



## Додаток Б

```
// БІБЛІОТЕКА для роботи датчика ехо локатора
#include <iarduino_НС_SR04_int.h>
// БІБЛІОТЕКА для роботи з кроковим двигуном
#include <Stepper.h>
// Номера виходів до яких підключені датчики (лінії, відстані) і
MotorShield (дані введені в константи потрібно буде змінити при підключенні
візка)
const uint8_t datLL = A3; // Датчик лінії, який знаходиться зліва
const uint8_t datLC = A4; // Датчик лінії, який знаходиться посередині
const uint8_t datLR = A5; // Датчик лінії, який знаходиться зправа
const uint8_t datV_TRIG = 2; // Вихід ехолокатора TRIG
const uint8_t datV_ECHO = 3; // Вихід ехолокатора ECHO
const uint8_t motSh_LH = 12; // Вихід направлення до якого підключений
лівий мотор
const uint8_t motSh_LE = 11; // Вихід ШІМ(широтна імпульсна модуляція)
до якого підключений лівий мотор
const uint8_t motSh_RE = 10; // Вихід ШІМ до якого підключений правий
мотор
const uint8_t motSh_RH = 13; // Вихід направлення до якого підключений
правий мотор
const float STEP_PER_REV = 32;
const float GEAR_RED = 64; //potribno glanytu na krokDV GEAR
REDACTION
const float STEPS_PER_OUT_REV = STEP_PER_REV * GEAR_RED;
int StepsRequired;
//
Stepper steppermotor(STEP_PER_REV, 6, 8, 7, 9); // TREBA PIDIBRATU
SVOII VUHODU
//
iarduino_НС_SR04_int HCSR(datV_TRIG, datV_ECHO); // Для роботи з
бібліотекою потрібно вказати об'єкт HCSR (і виходи еходатчика)
// Значення які ми встановлюємо:
const uint16_t datLOL = 681; // Дані які датчик показує на лінії (це
значення потрібно настроїти)
const uint16_t datLOOL = 174; // Дані які датчик показує поза лінією (це
значення потрібно настроїти)
const uint8_t MAXSP = 255; // Масимальна швидкість
const uint32_t TimerST = 2000; // Впородовж записаного часу візок не
буде зупинятись коли він поза лінією
const uint8_t Povорот = 10; // Значення швидкості реакції на поворот (чим
більше значення тим різкіше він реагує)
const uint8_t Vidst = 10; // Мінімальне значення в сантиметрах, коли візок
буде реагувати на перешкоду
const bool OborotMot[2] = {1,1}; // В яку сторону будуть крутитися
```

```

двигуни (залежить від полярності, тобто від під'єднання моторів)
    // Значення які будуть рахуватись:
    uint8_t MotorSP[2]; // Швидкість на моторах (0 - правий мотор, 1 - лівий
мотор)
    uint16_t datLMID; // Середнє значення показу датчиків лінії
    uint8_t datLBIT; // Константа, потрібна буде для побітного виводу
значення датчиків лінії
    bool flgLine; // Прапорець, який характеризуватиме лінію (0 - темна лінія,
1 - світла лінії)
    int8_t flgTurning; // Прапорець, який характеризуватиме поворот (0 -
прямо їдем, -1 - наліво, +1 - направо)
    bool flgPWM; // Прапорець, який характеризуватиме ШІМ
    bool flgStop; // Прапорець, який характеризуватиме зупинку (0 - без
зупинки, 1 - зупинка)
    bool flgDistance; // Прапорець, який характеризуватиме відстань до
перешкоди (0 - нема перешкоди, 1 - є перешкода)
    uint32_t TimerML; // Час впродовж якого була проведена попередня
операція
    void setup(){
    // Розпізнавання кольору лінії
    flgLine = (datLOOL>datLOL); // Якщо значення за лінією більше тоді буде
встановлюватись 1, а це означає, що ми пересуваємося по світлій лінії на
темному фоні
    // Якщо навпаки, то буде скидатись на 0 і це означатиме темна лінія на
світлому фоні
    // Визначаєм середнє значення показників датчика лінії (на ній і поза нею)
    // Так як у нас може їздити по темній лінії на світлому фоні і навпаки
потрібні різні формули для обрахування
    if(flgLine){datLMID = datLOL + (datLOOL - datLOL)/2;} // Для світлої лінії
    else {datLMID = datLOOL + (datLOL - datLOOL)/2;} // Для темної лінії
    // Присвоюєм моторам конкретні значення швидкості (1 - лівий мотор; 0 -
правий мотор)
    MotorSP[1] = MAXSP;
    MotorSP[0] = MAXSP;
    // Встановлюєм прапорець ШІМ, скидаєм значення прапорця повороту,
зупинки і відстані
    flgPWM=1; flgTurning=0; flgStop=0; flgDistance=0;
    // Встановлюєм режими роботи виходів і направлення обертів моторів
    pinMode (datLL, INPUT );    // Встановлюєм вихід лівого датчика лінії
як вхід (для того щоб получить дані які він нам покаже)
    pinMode (datLC, INPUT );    // Встановлюєм вихід лівого датчика лінії
як вхід
    pinMode (datLR, INPUT);    // Встановлюєм вихід лівого датчика лінії
як вхід
    pinMode (motSh_LH, OUTPUT );    // Встановлюєм вихід motSh_LH як

```

```

вхід (для керування напрямленням оборотів лівого мотору)
    pinMode (motSh_LE, OUTPUT );    // Встановлюємо вихід motSh_LE як
вхід (для керування швидкістю обертів лівого мотору)
    pinMode (motSh_RE, OUTPUT );    // Встановлюємо вихід motSh_RE як
вхід
    pinMode (motSh_RH, OUTPUT );    // Встановлюємо вихід motSh_RH як
вхід
    digitalWrite(motSh_LH, OborotMot[1]);    // Встановлюємо на виході
motSh_LH значення OborotMot[1] (направлення обертів лівого мотору)
    digitalWrite(motSh_RH, OborotMot[0]);    // Встановлюємо на виході
motSh_RH значення OborotMot[0]
    // Записуємо показання центрального датчика лінії в константи datLOL і
datLOOL
    Serial.begin(9600); while(!Serial){}    // Зчитування/Передача даних через
порт на швидкості 9600 б/с
    Serial.println(analogRead(datLC));    // Виводим показання датчика на
екран для запису в константи
    // Встановлюємо затримку 2 сек і поновлюємо час з моменту виконання
останньої операції
    delay(2000);
    TimerML = millis();
}
void loop(){
    if (MotorSP[1] = 0, MotorSP[0] = 0, flgDistance = 0, flgStop = 0){
        steppermotor.setSpeed(1);
        StepsRequired = 4;
        steppermotor.step(StepsRequired);
        delay(2000);
    }
    // Serial.println(HCSR.distance());//
    // delay(1000);//
    // Заносим показання трьох датчиків в одну змінну встановлюючи прості
значення (1 - датчик на лінії, 0 - датчик поза лінією)
    datLBIT = 0;    // Спочатку скидаємо значення даної змінної
    datLBIT |= ((analogRead(datLL)>datLMID)^flgLine)<<2; // Встановлюємо 2
біт змінної datLBIT в 1, якщо лівий датчик знаходиться на лінії,якщо ні то
залишаємо біт в 0
    datLBIT |= ((analogRead(datLC)>datLMID)^flgLine)<<1; // Встановлюємо 1
біт змінної datLBIT в 1, якщо центральний датчик знаходиться на лінії,якщо ні
то залишаємо біт в 0
    datLBIT |= ((analogRead(datLR)>datLMID)^flgLine)<<0; // Встановлюємо 0
біт змінної datLBIT в 1, якщо правий датчик знаходиться на лінії,якщо ні то
залишаємо біт в 0
    // Як візок буде себе поводити відносно значення datLBIT
    switch(datLBIT){

```

```

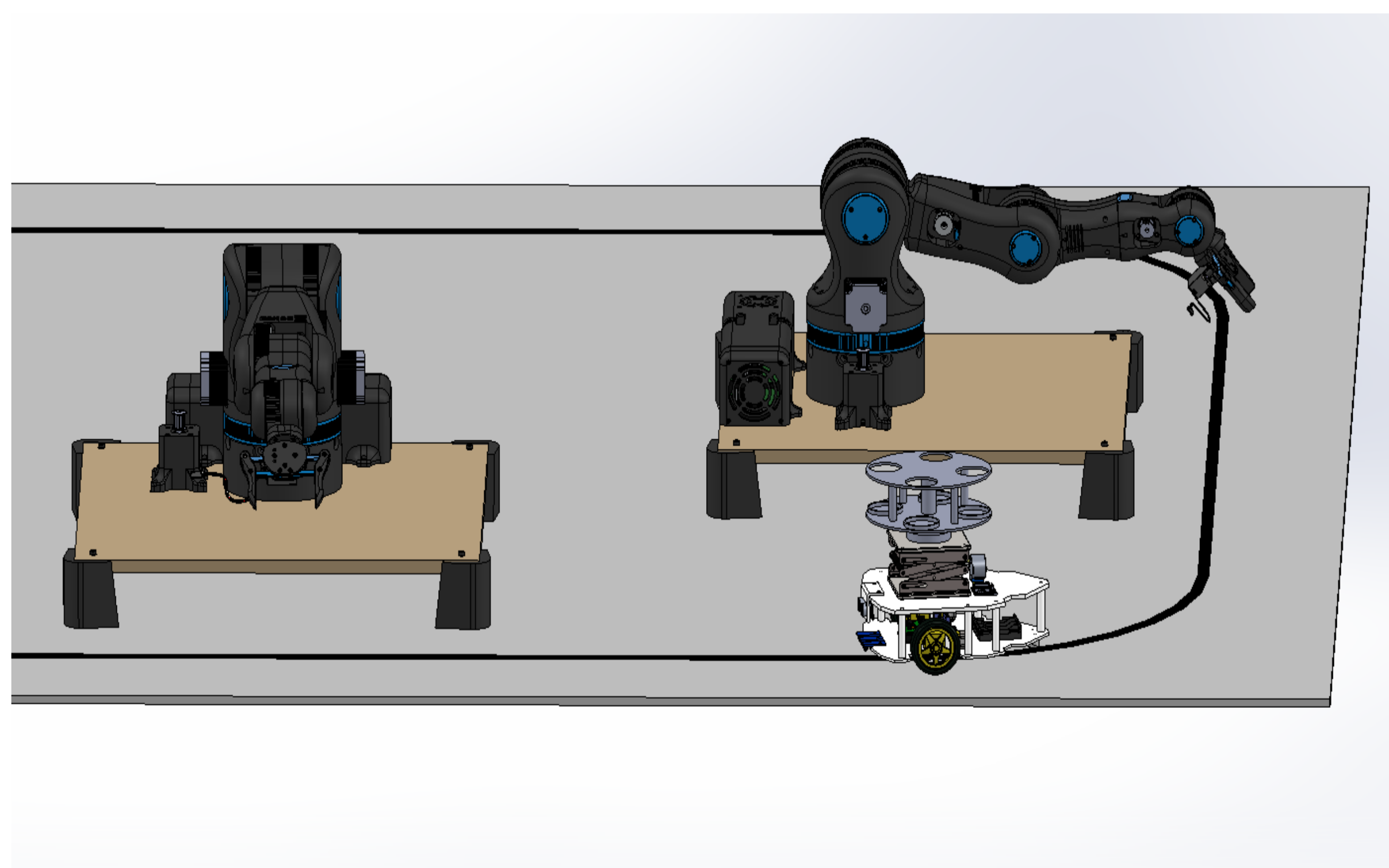
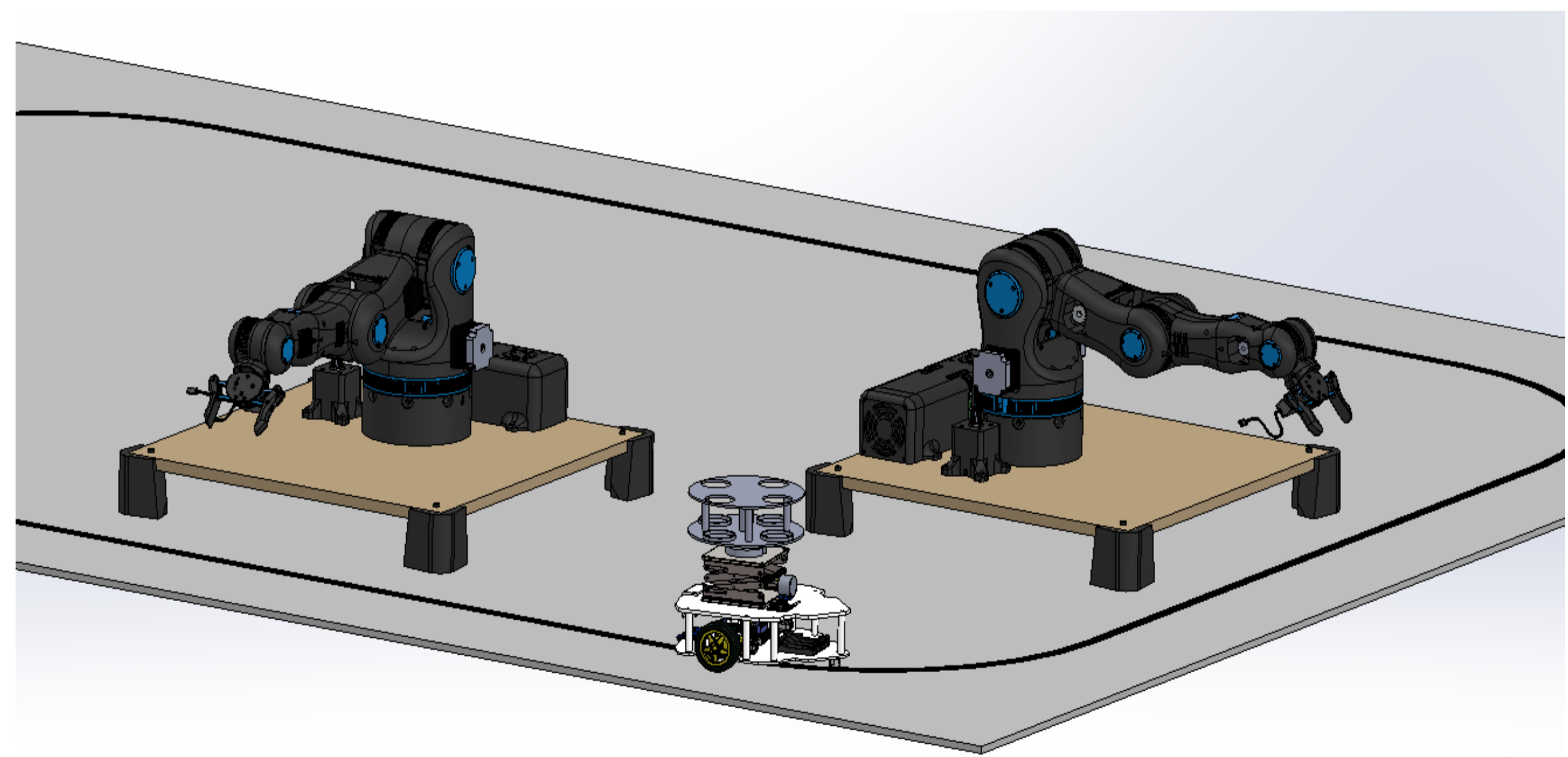
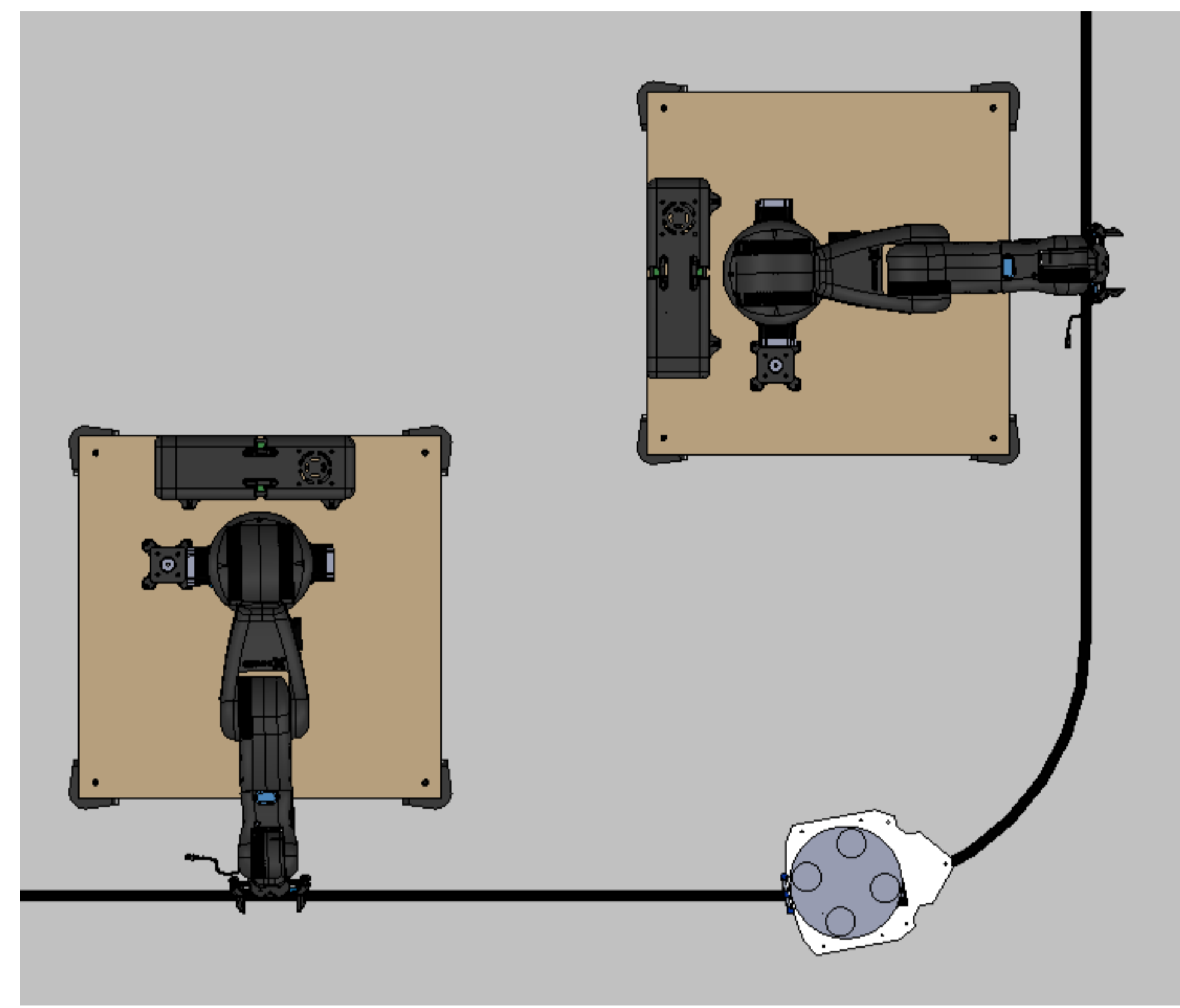
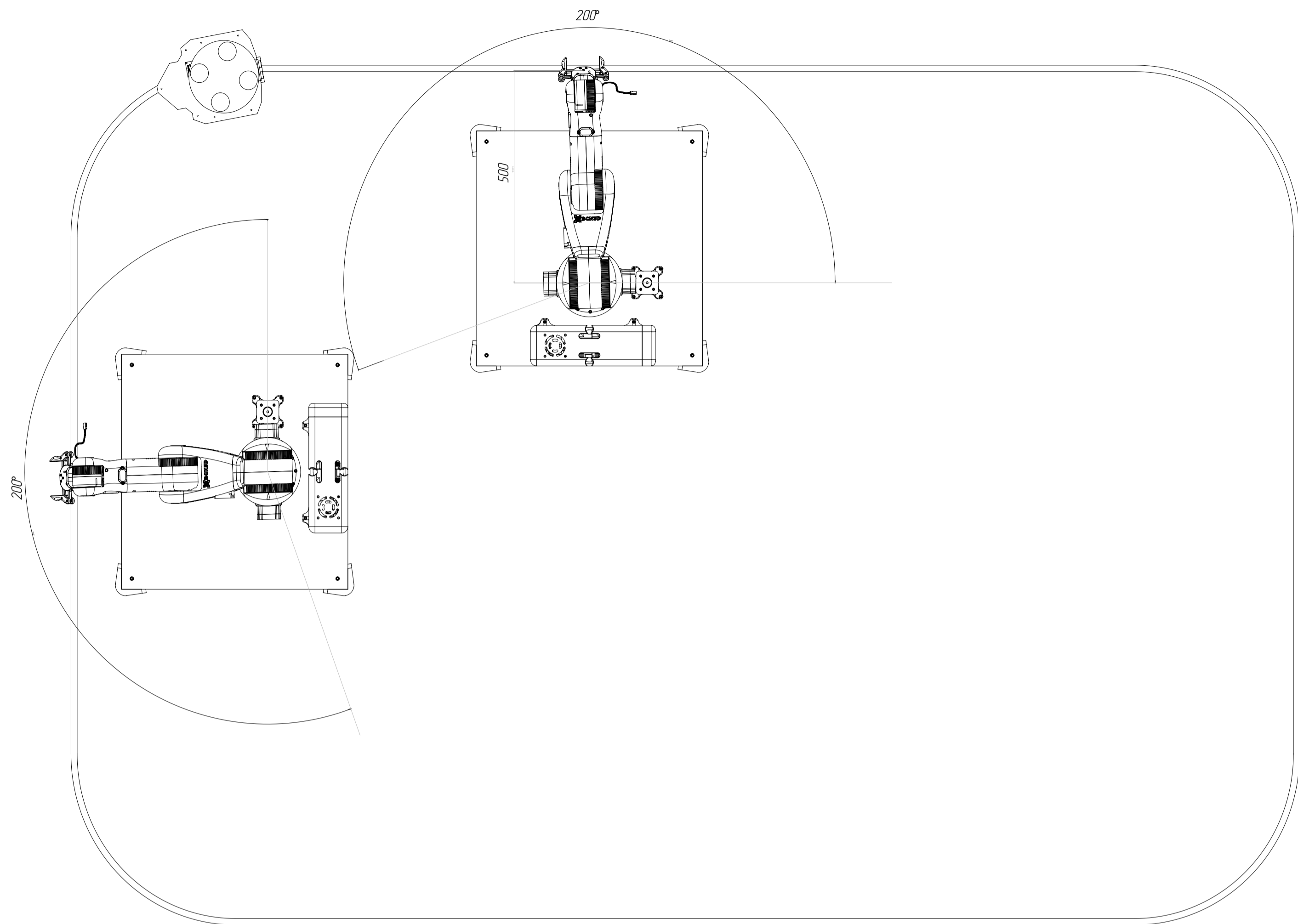
        case 0b000:    flgPWM=flgTurning;    flgStop=!flgTurning; break; // 000
- Всі датчики поза лінією
        case 0b010:    TimerML=millis(); flgPWM=flgTurning; flgTurning=0;
flgStop=0;    break; // 010 - Тільки центральний датчик на лінії
        case 0b100:    // 100 - Тільки лівий датчик на лінії
        case 0b110:    TimerML=millis(); flgPWM=1;    flgTurning=-1; flgStop=0;
break; // 110 - Центральний і лівий датчик на лінії
        case 0b001:    // 001 - Тільки правий датчик на лінії
        case 0b011:    TimerML=millis(); flgPWM=1;    flgTurning=1; flgStop=0;
break; // 011 - Правий і центральний датчик на лінії
        default:    lgPWM=1;    break;
    }
    if(TimerML>millis()) { TimerML=0;} // Позбавляємося
переповнення mills()
    if(TimerML+TimerST<millis()){    flgPWM=1; flgTurning=0; flgStop=1;
} // Зупиняємося якщо лінія втрачена на більше часу ніж 2 сек
    if(HCSR.distance()<Vidst){ TimerML=millis(); flgPWM=1; flgDistance=1;
} // Зупиняємося якщо спереди перешкода
    else if(flDistance) { TimerML=millis(); flgPWM=1;
flDistance=0;    } // Продовжуємо рух якщо перешкода пропала
    // Встановлюємо ШІМ для моторів
    if(flPWM){flPWM=0;
    switch(flTurning){ // Швидкість лівого мотору: Швидкість правого
мотору:
        case -1: if(MotorSP[1]>0){MotorSP[1]--;} MotorSP[0]=MAXSP;
break; // Зменшуємо швидкість лівого мотору коли повертаєм
        case 0: MotorSP[1]=MAXSP; MotorSP[0]=MAXSP;    break; //
Однакова швидкість коли їдемо прямо
        case 1: MotorSP[1]=MAXSP; if(MotorSP[0]>0){MotorSP[0]--;}
break; // Зменшуємо швидкість правого мотору коли повертаєм
    } if(flStop){ MotorSP[1]=0; MotorSP[0]=0;} // Зупиняємося якщо
flStop = 1
    if(flDistance){ MotorSP[1]=0; MotorSP[0]=0;} // зупиняємося
якщо flDistance = 1
    // Обраховані значення передаємо на вхід до моторів
    analogWrite(motSh_LE, MotorSP[1]);
    analogWrite(motSh_RE, MotorSP[0]);
    }
}
}

```

Форм.	Зона	Поз.	Обозначение	Наименование	Кол.	Примеч.
				<u>Документація</u>		
A1			MP.ПМКм-032.02.00.000 СК	Складальне креслення	1	
				<u>Складальні одиниці</u>		
		1	MP.ПМКм-032.02.01.000	Домкрат	1	
		2	MP.ПМКм-032.02.02.000	Поворотний стіл	1	
				<u>Деталі</u>		
		3	MP.ПМКм-032.02.00.003	Основа нижня	1	
		4	MP.ПМКм-032.02.00.004	Основа верхня	1	
A4		5	MP.ПМКм-032.02.00.005	Кріплення ехолокатора	1	
A4		6	MP.ПМКм-032.02.00.006	Кріплення мотора	2	
A4		7	MP.ПМКм-032.02.00.007	Опора	11	
				<u>Стандартні деталі</u>		
		8		Плата Arduino UNO R3	1	
		9		Motor Shield L298P	1	
		10		HC-SR04	1	

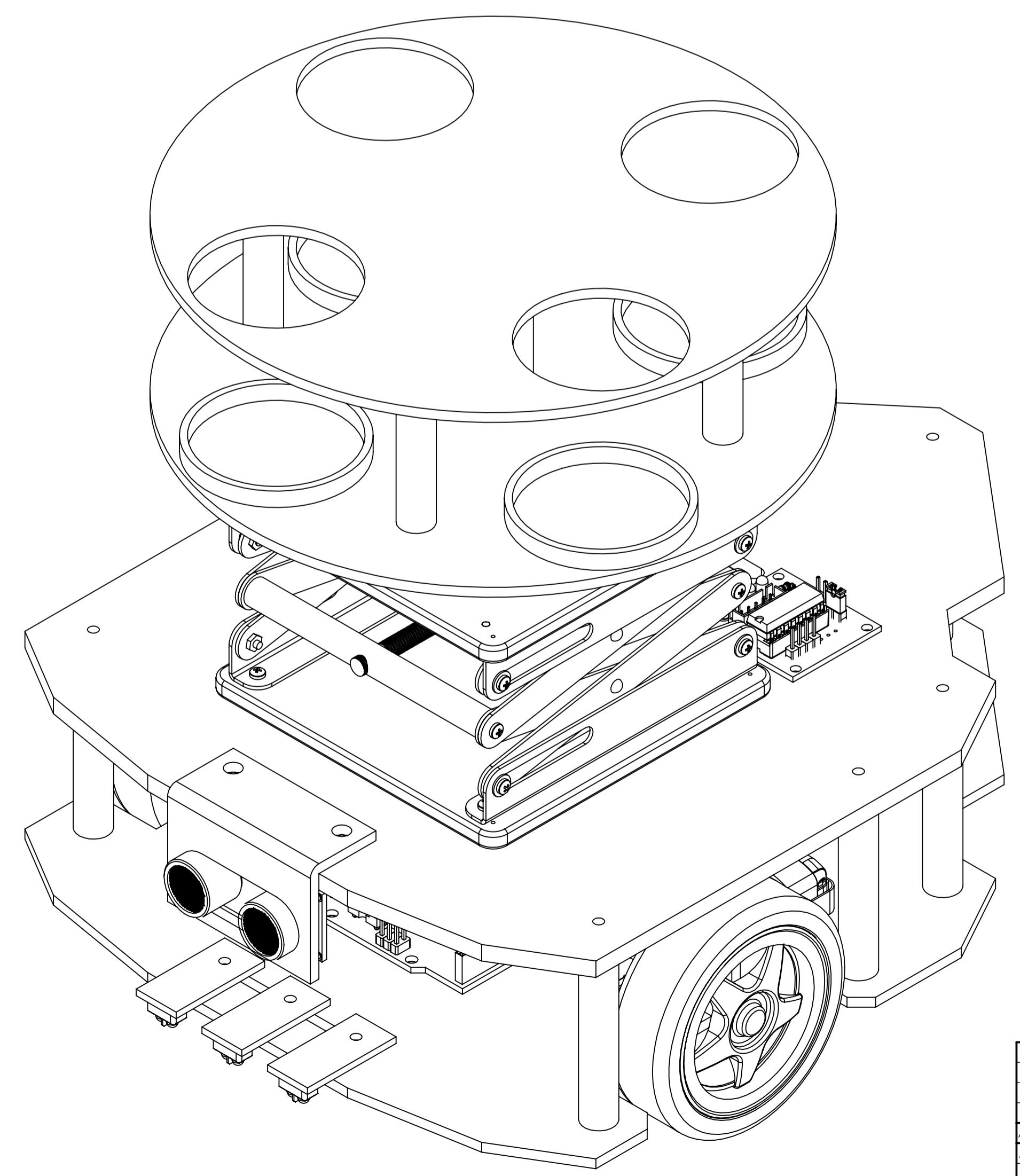
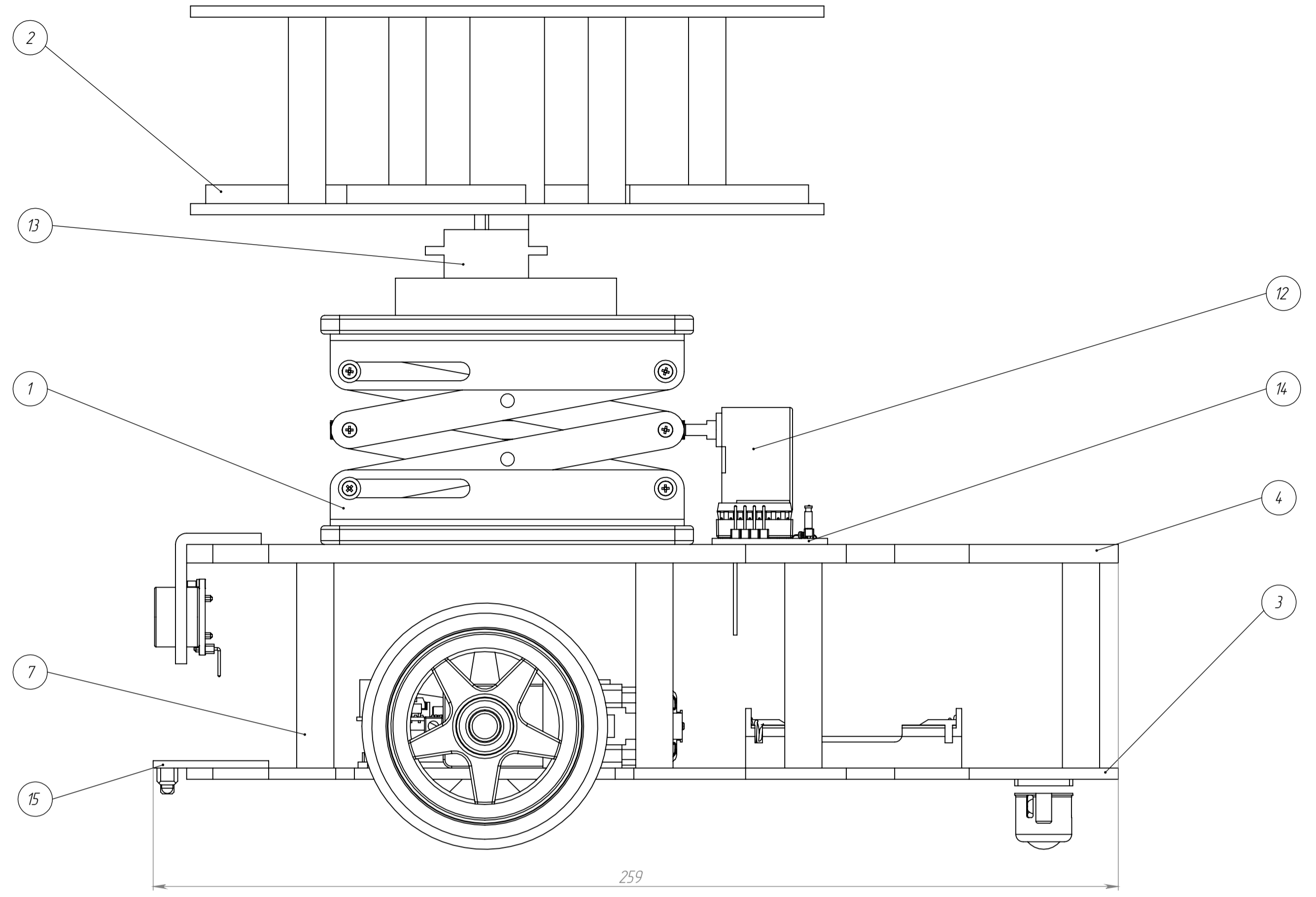
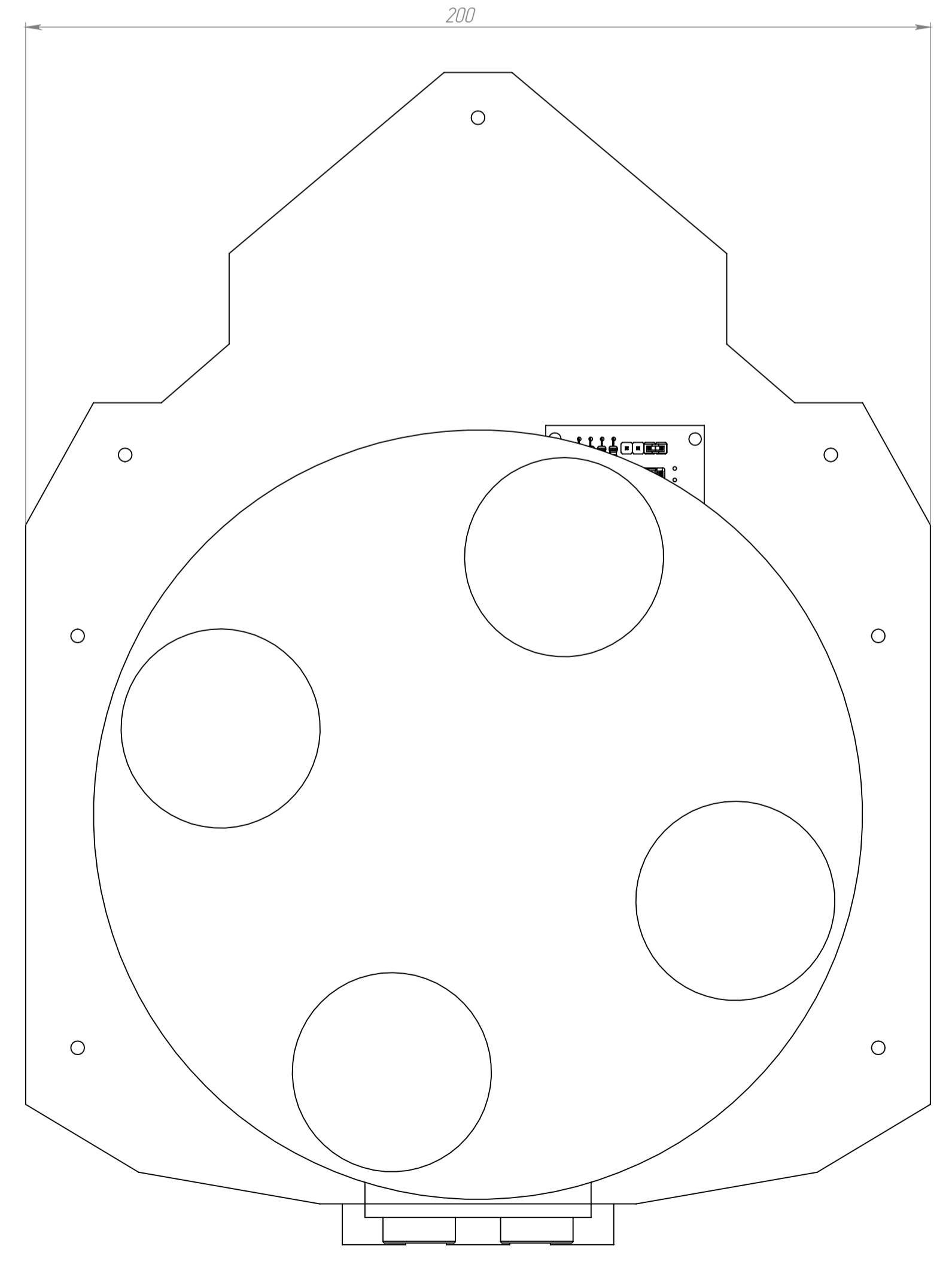
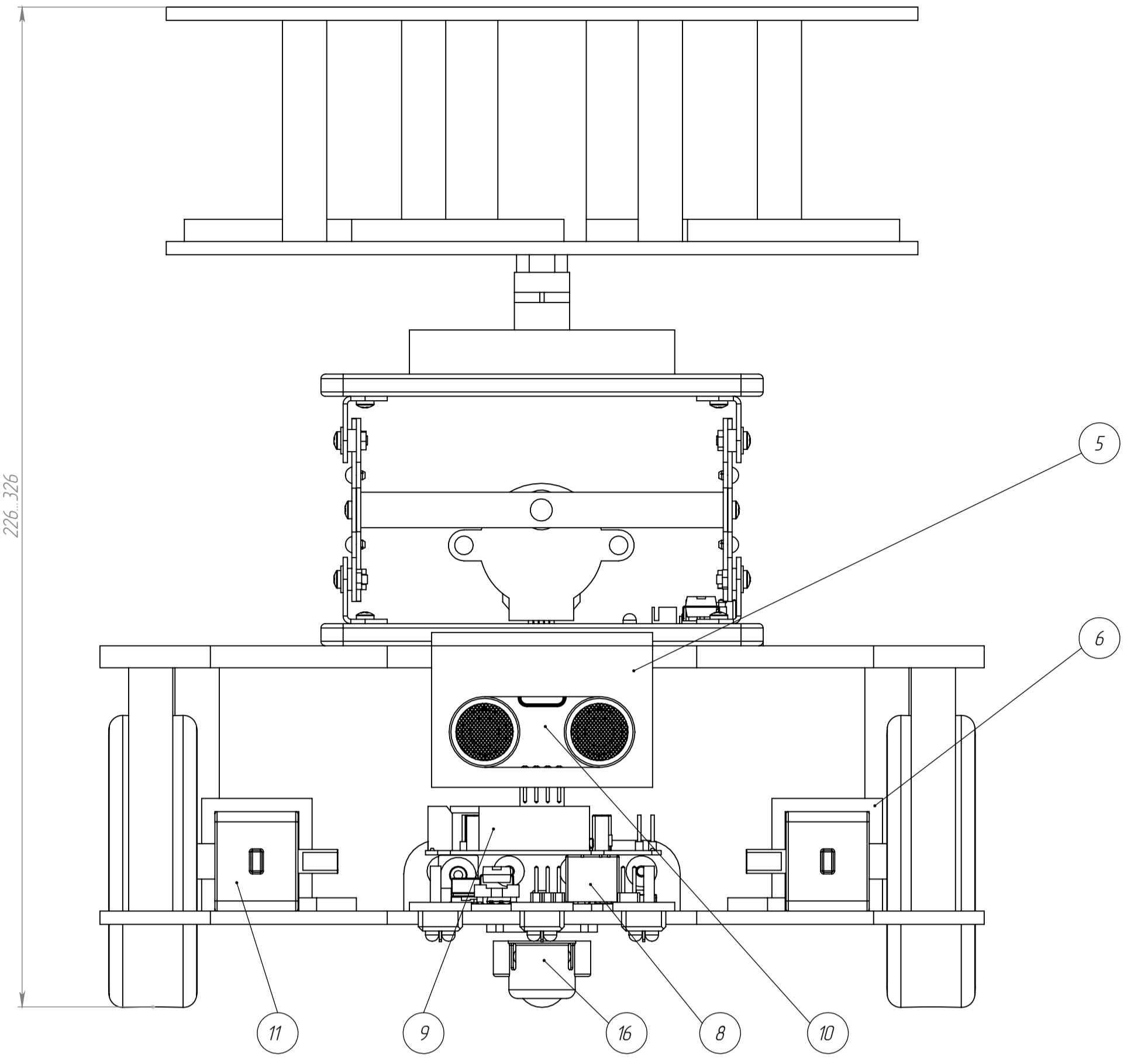
<b>MP.ПМКм-032.02.00.000</b>				
Изм	Лист	№ докум	Подп.	Дата
Разраб.	Веркалець А.А.			
Пров.	Лукаць Т.В.			
Н.контр.				
Утв.				
<b>Транспортна платформа</b>			Лит.	Лист
				Листов
				1
			<b>ІФНТУНГ</b>	
			<b>ПМКм-21-1</b>	





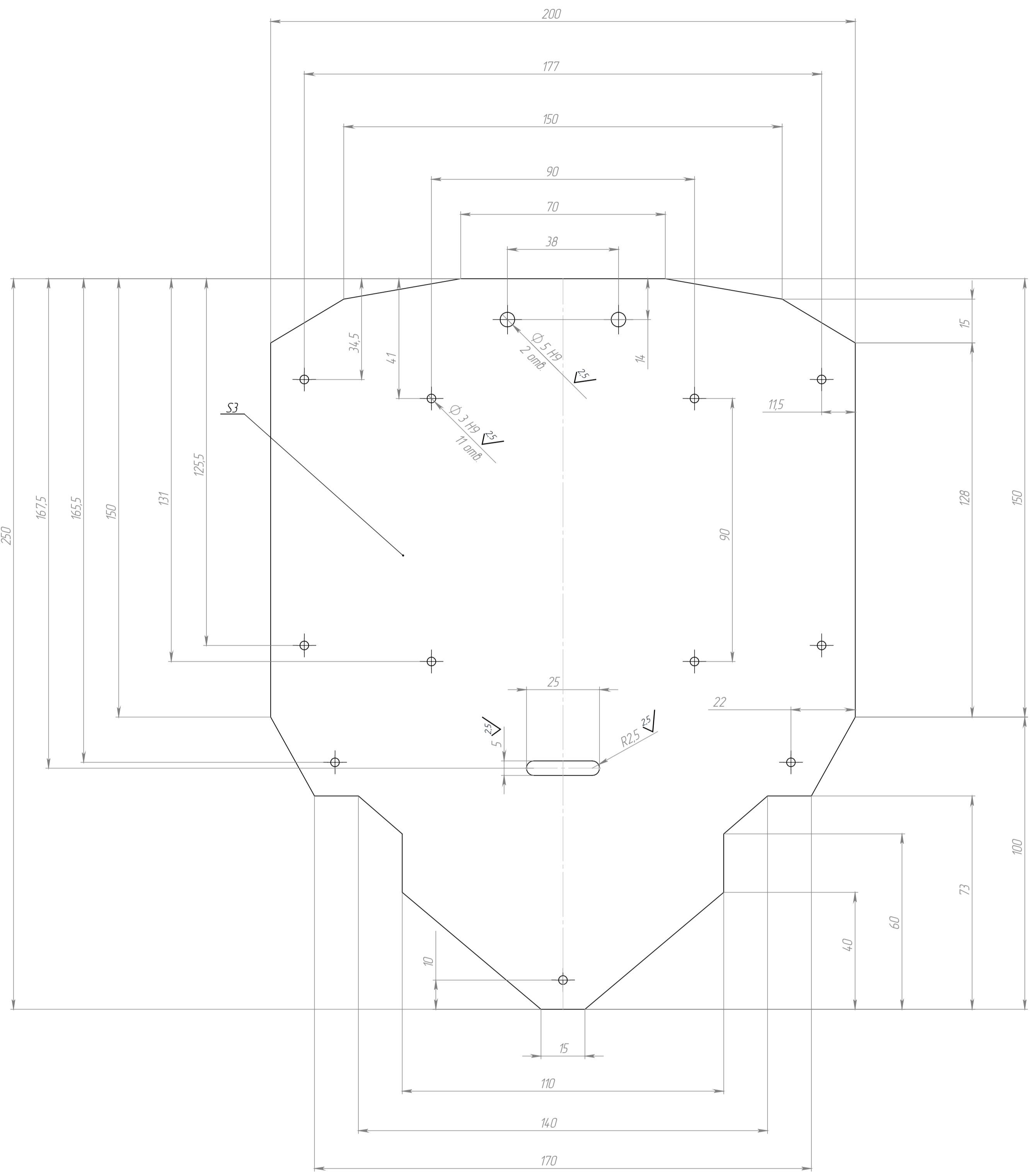
					<b>MP.ПМКМ-032.01.00.000</b>		
Имя	Лист	№ докум.	Год	Дата	Проект транспортной платформы для обслуживания дільниці		
Разработ	Веркавель А.А.						
Проб	Льцань Т.В.						
Г. констр							
И. констр					Лист	Листов	
Синд							ФНТУНГ ПМКМ-21-1
					1	Копировать	Формат А1

Лист 11  
Лист 12  
Лист 13  
Лист 14  
Лист 15  
Лист 16  
Лист 17  
Лист 18  
Лист 19  
Лист 20  
Лист 21  
Лист 22  
Лист 23  
Лист 24  
Лист 25  
Лист 26  
Лист 27  
Лист 28  
Лист 29  
Лист 30  
Лист 31  
Лист 32  
Лист 33  
Лист 34  
Лист 35  
Лист 36  
Лист 37  
Лист 38  
Лист 39  
Лист 40  
Лист 41  
Лист 42  
Лист 43  
Лист 44  
Лист 45  
Лист 46  
Лист 47  
Лист 48  
Лист 49  
Лист 50  
Лист 51  
Лист 52  
Лист 53  
Лист 54  
Лист 55  
Лист 56  
Лист 57  
Лист 58  
Лист 59  
Лист 60  
Лист 61  
Лист 62  
Лист 63  
Лист 64  
Лист 65  
Лист 66  
Лист 67  
Лист 68  
Лист 69  
Лист 70  
Лист 71  
Лист 72  
Лист 73  
Лист 74  
Лист 75  
Лист 76  
Лист 77  
Лист 78  
Лист 79  
Лист 80  
Лист 81  
Лист 82  
Лист 83  
Лист 84  
Лист 85  
Лист 86  
Лист 87  
Лист 88  
Лист 89  
Лист 90  
Лист 91  
Лист 92  
Лист 93  
Лист 94  
Лист 95  
Лист 96  
Лист 97  
Лист 98  
Лист 99  
Лист 100



					МР.ПМКМ-032.02.00.000 СК		
Изм	Лист	№ докум	Дата	Лист	Итого	Транспортная платформа	
		Веркалец А.А.				Итого	11
		Льваць Т.В.				Листов	
						ИФНТУНГ	
						ПМКМ-21-1	
						Формат А1	

Лист 11 из 11  
 Веркалец А.А.  
 Льваць Т.В.  
 ИФНТУНГ  
 ПМКМ-21-1  
 Формат А1

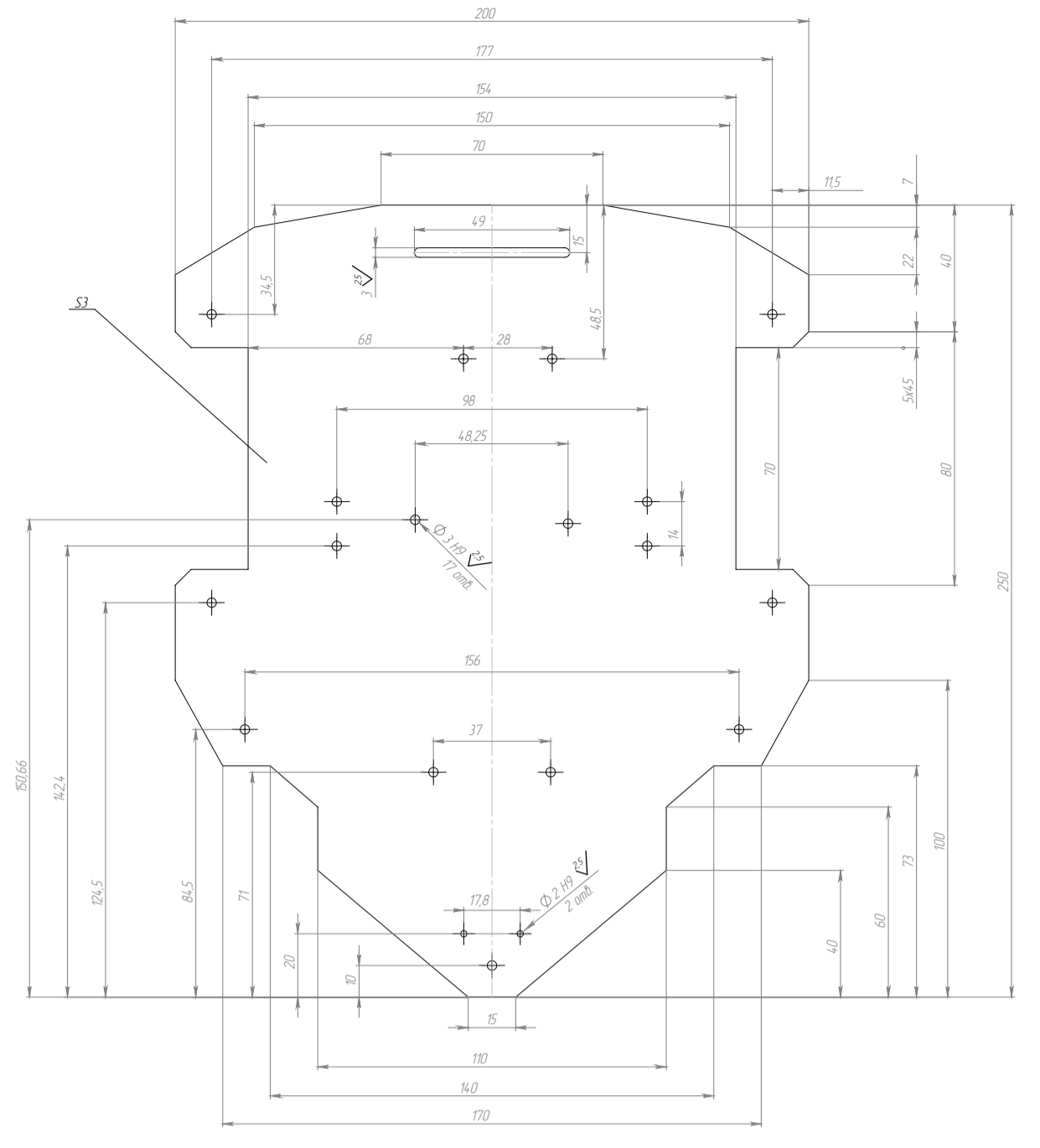


1 h14

MP.ПМКМ-032.02.00.004					Лист	Масса	Масштаб
Основа верхняя					Н		1:1
Д1 ГОСТ 17232-99					Лист	Листов	
Изм.	Лист	№ докум.	Подп.	Дата			
Разраб.	Веркалець А.А.						
Проб.	Лукаць Т.В.						
Т.контр.	Лукаць Т.В.						
Н.контр.	Лукаць Т.В.						
Читб.	Панчук В.Г.						

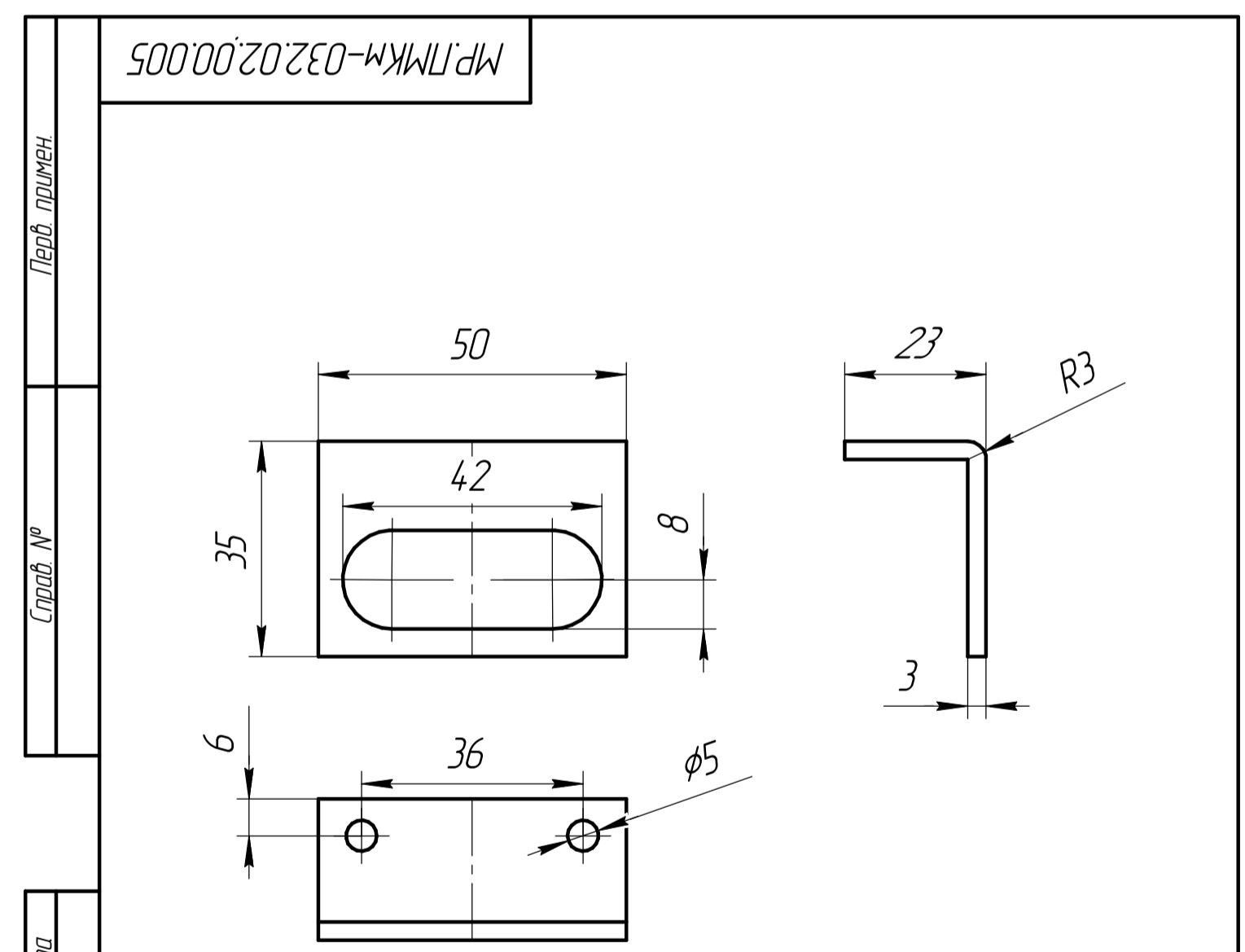
12.5/11

MP.ПМКМ-032.02.00.004

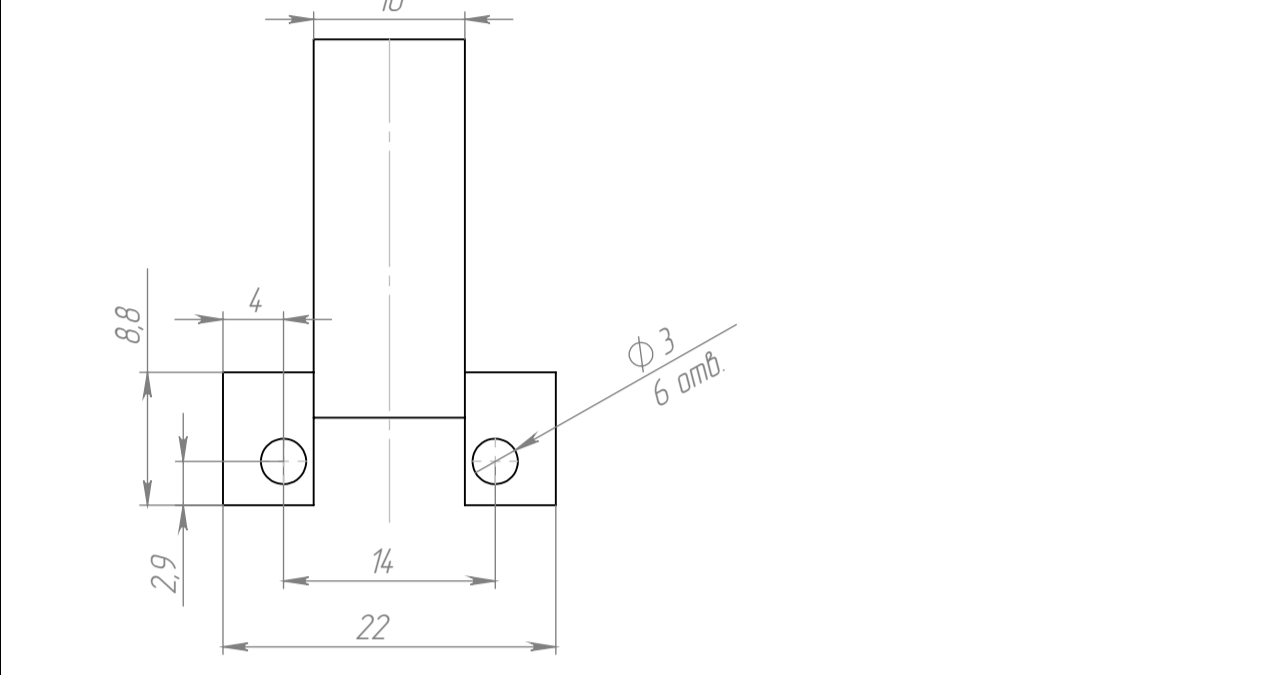
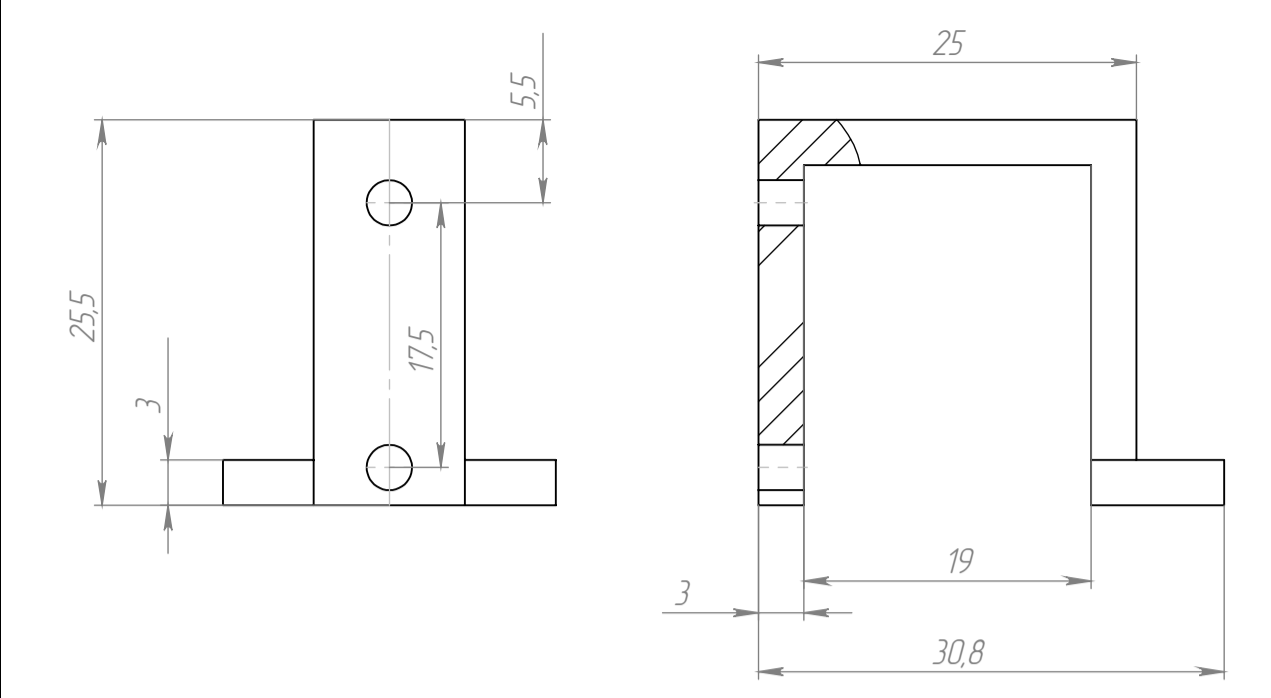


1 h14

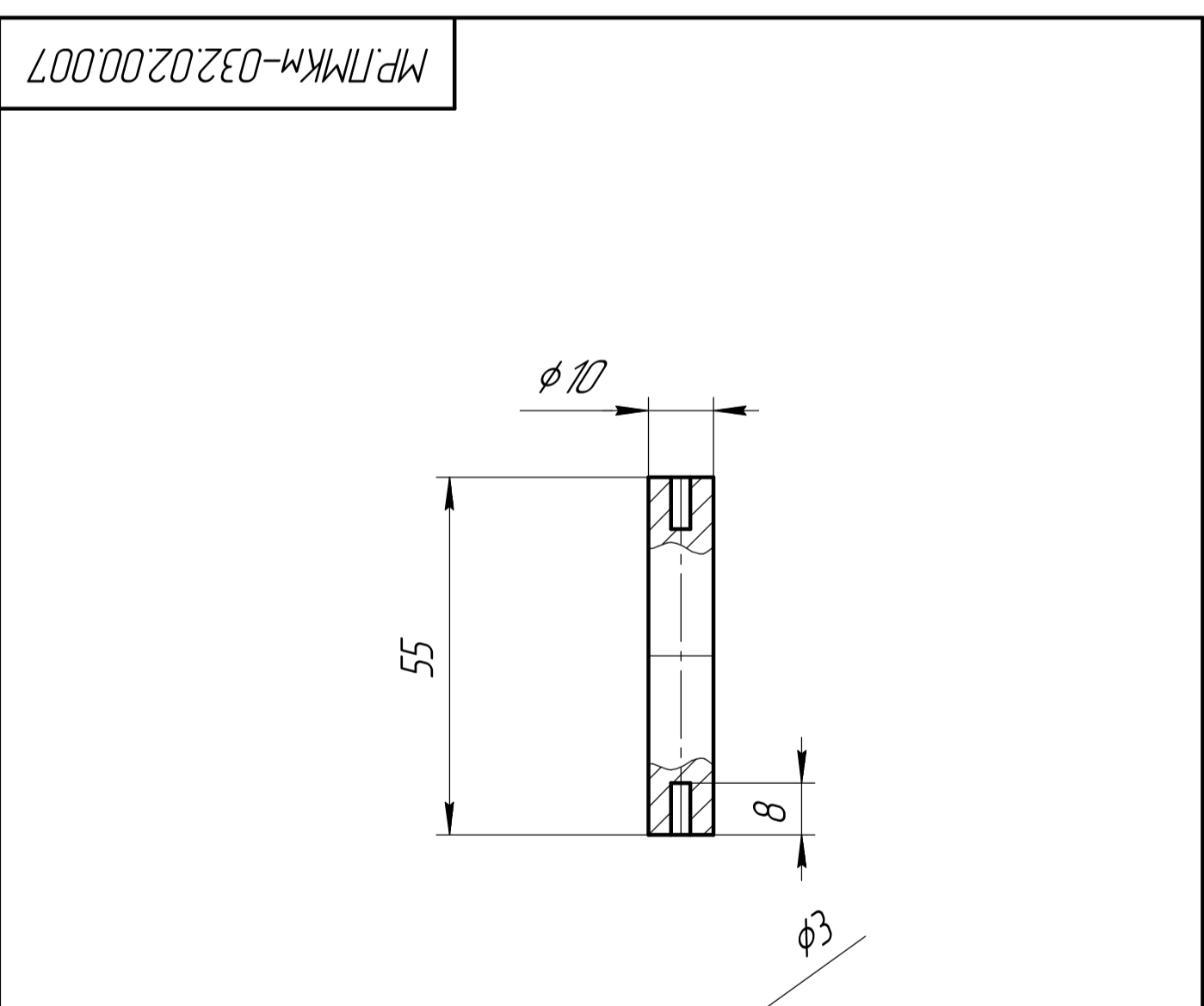
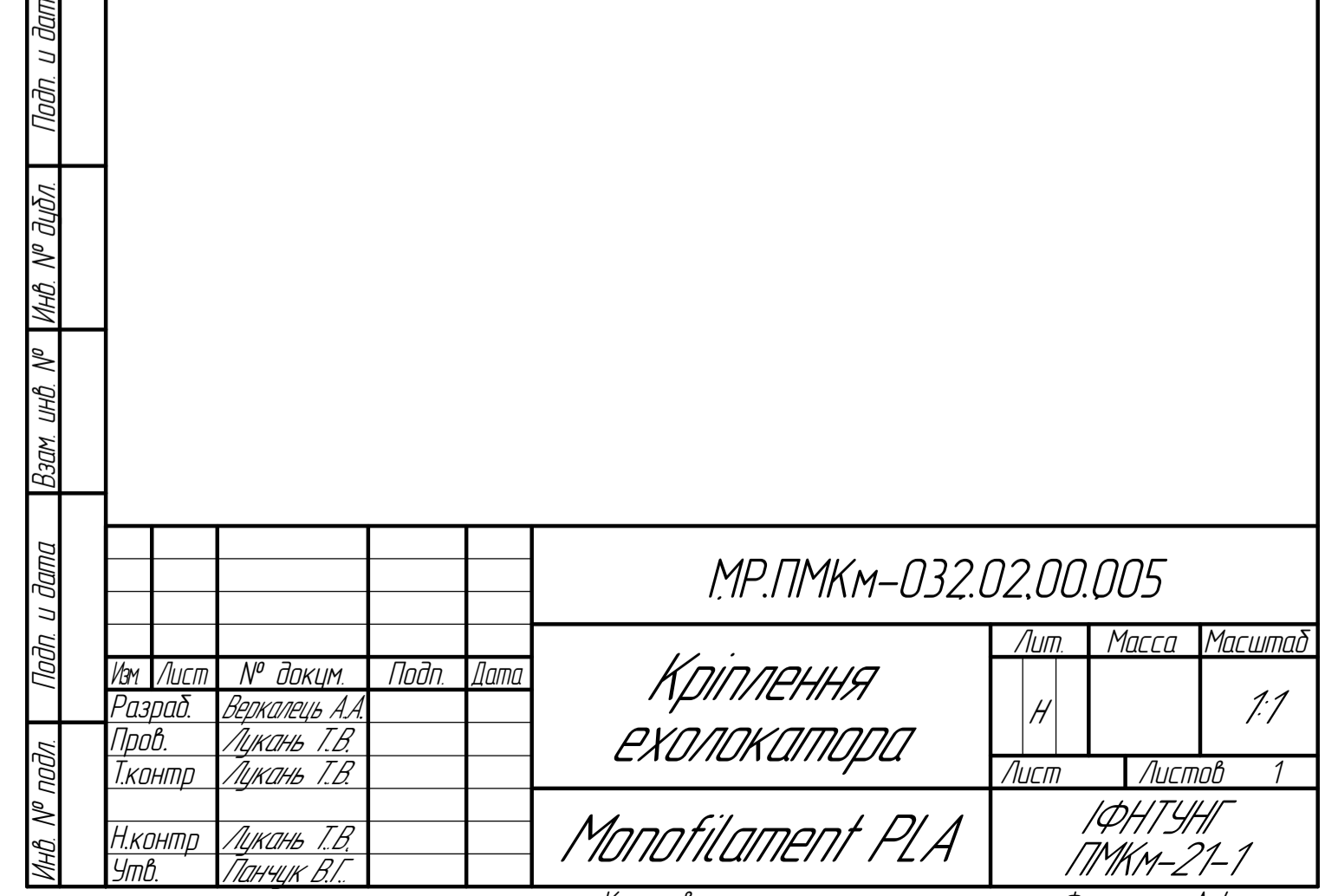
MP.ПМКМ-032.02.00.003					Лист	Масса	Масштаб
Основа нижняя					Н		1:1
Д1 ГОСТ 17232-99					Лист	Листов	
Изм.	Лист	№ докум.	Подп.	Дата			
Разраб.	Веркалець А.А.						
Проб.	Лукаць Т.В.						
Т.контр.	Лукаць Т.В.						
Н.контр.	Лукаць Т.В.						
Читб.	Панчук В.Г.						



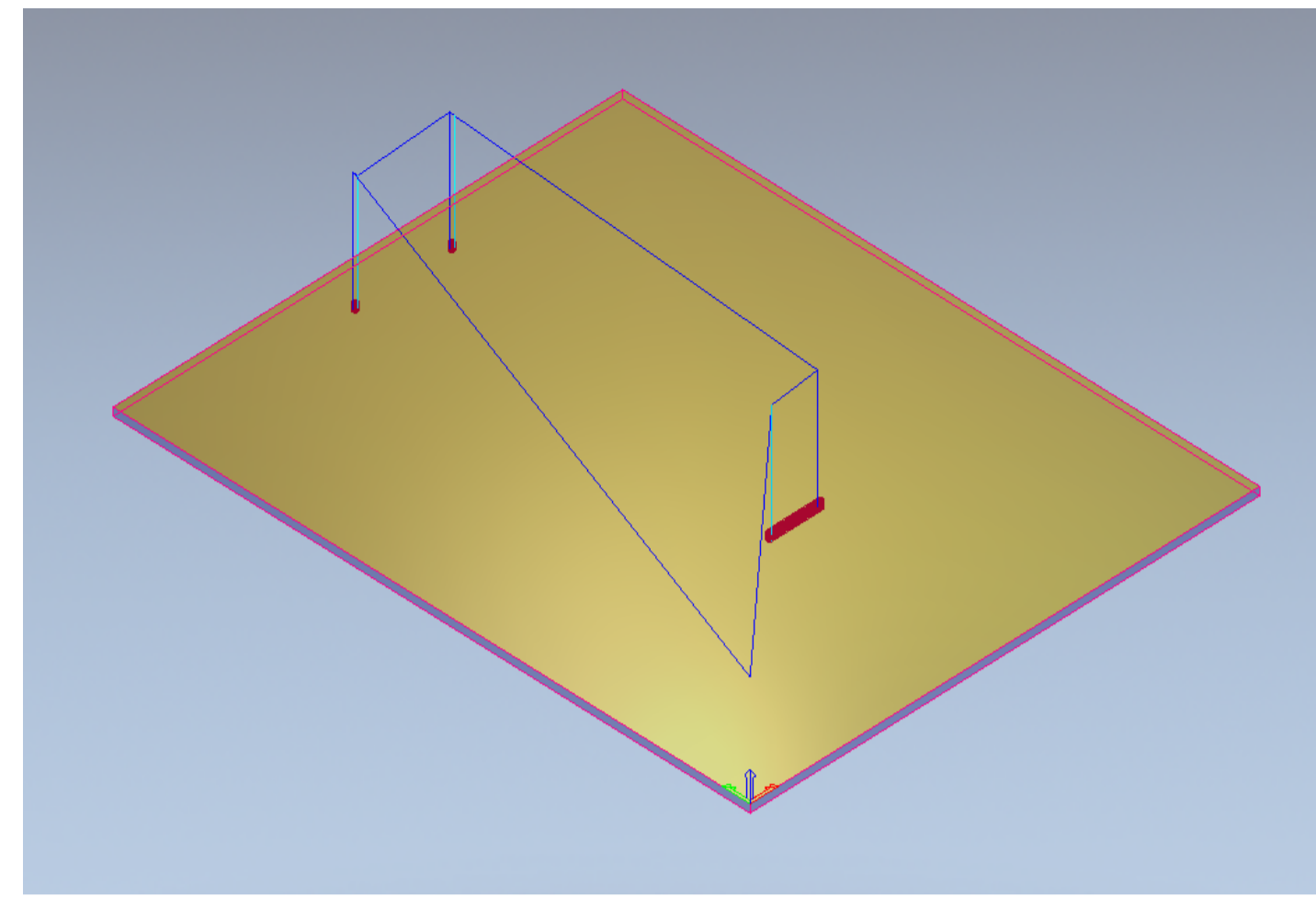
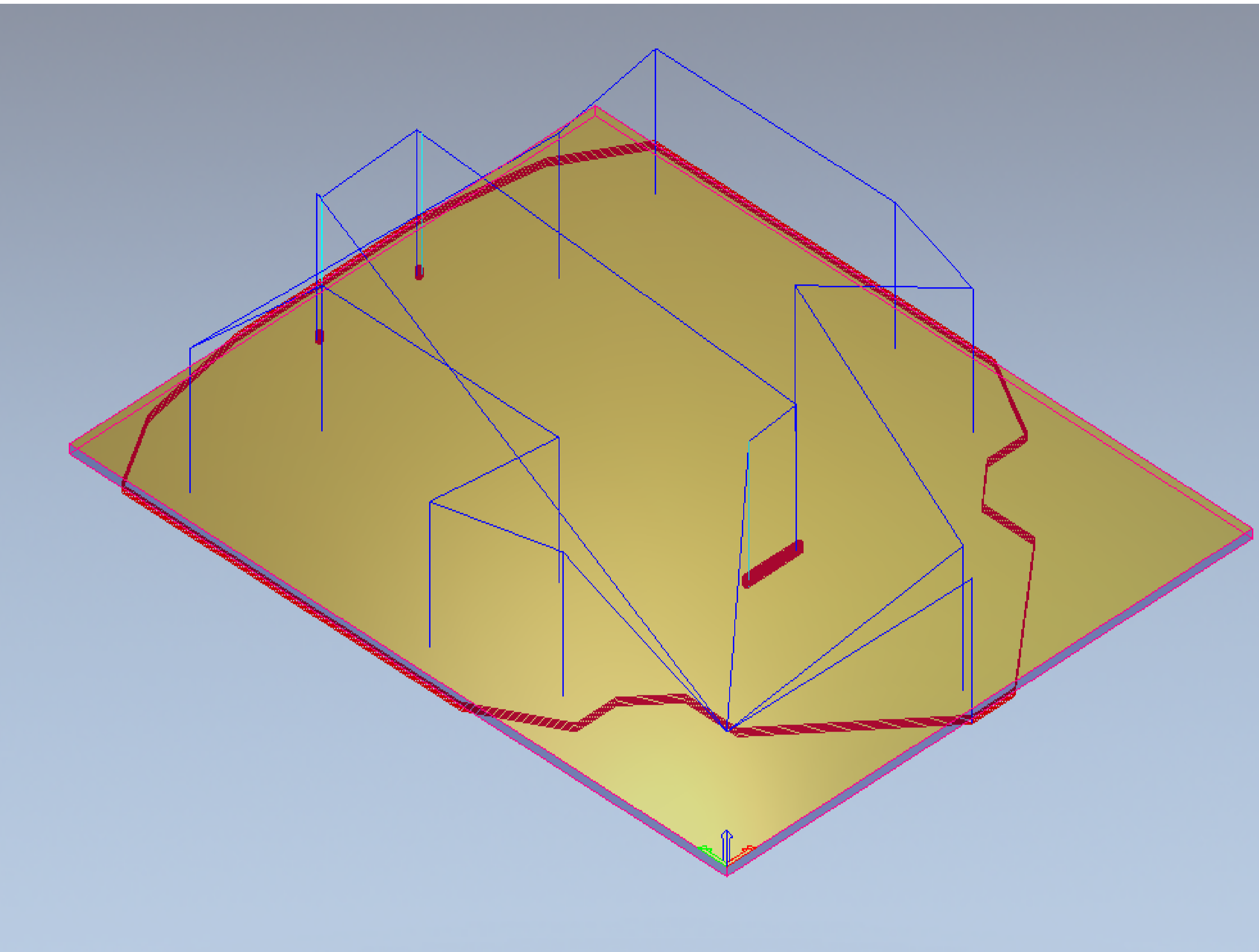
MP.ПМКМ-032.02.00.005					Лист	Масса	Масштаб
Кріплення ехолотатора					Н		1:1
Monofilament PLA					Лист	Листов	1
Изм.	Лист	№ докум.	Подп.	Дата			
Разраб.	Веркалець А.А.						
Проб.	Лукаць Т.В.						
Т.контр.	Лукаць Т.В.						
Н.контр.	Лукаць Т.В.						
Читб.	Панчук В.Г.						



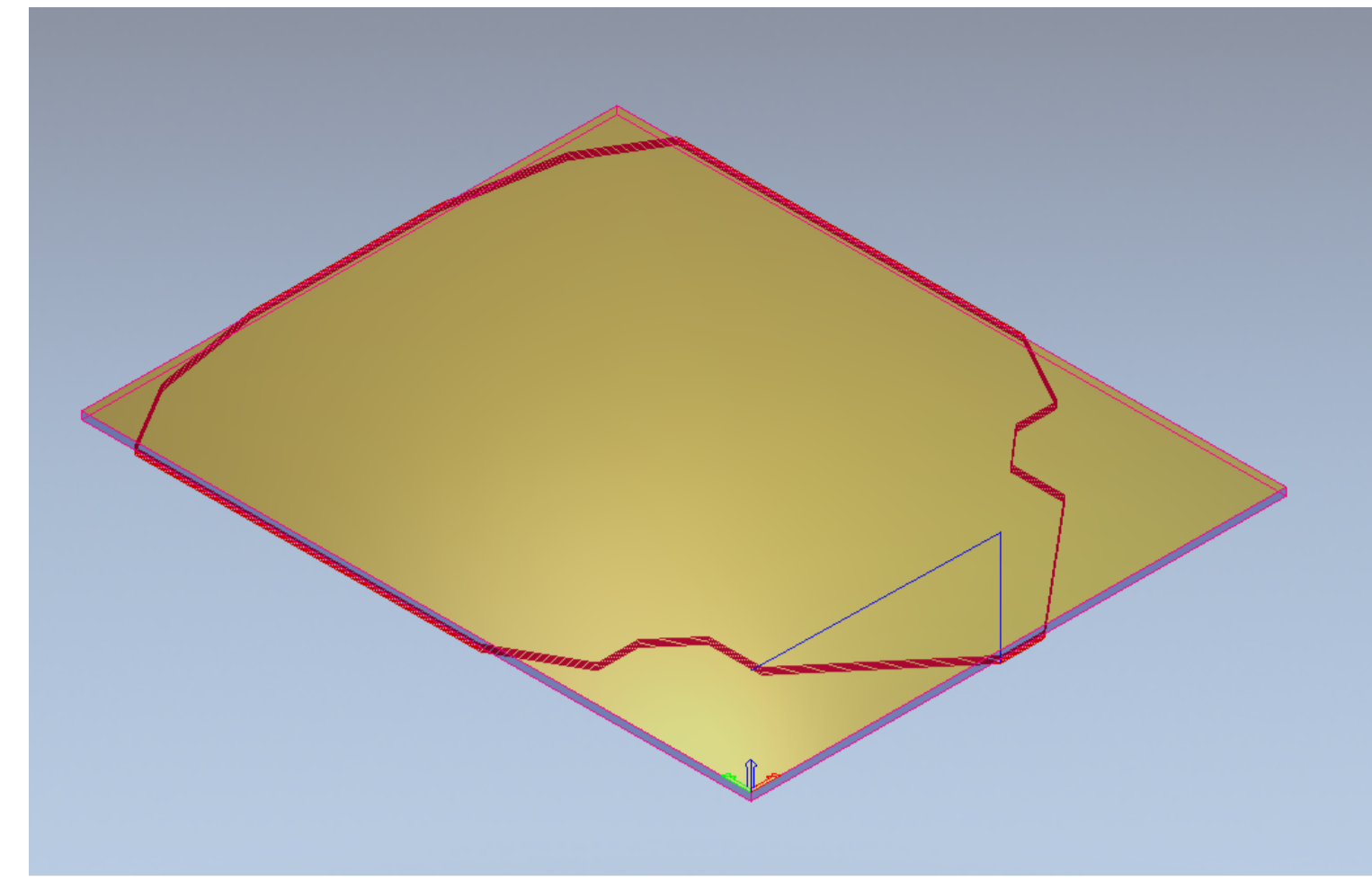
MP.ПМКМ-032.02.00.006					Лист	Масса	Масштаб
Кріплення мотору					Н		1:1
Monofilament PLA					Лист	Листов	1
Изм.	Лист	№ докум.	Подп.	Дата			
Разраб.	Веркалець А.А.						
Проб.	Лукаць Т.В.						
Т.контр.	Лукаць Т.В.						
Н.контр.	Лукаць Т.В.						
Читб.	Панчук В.Г.						



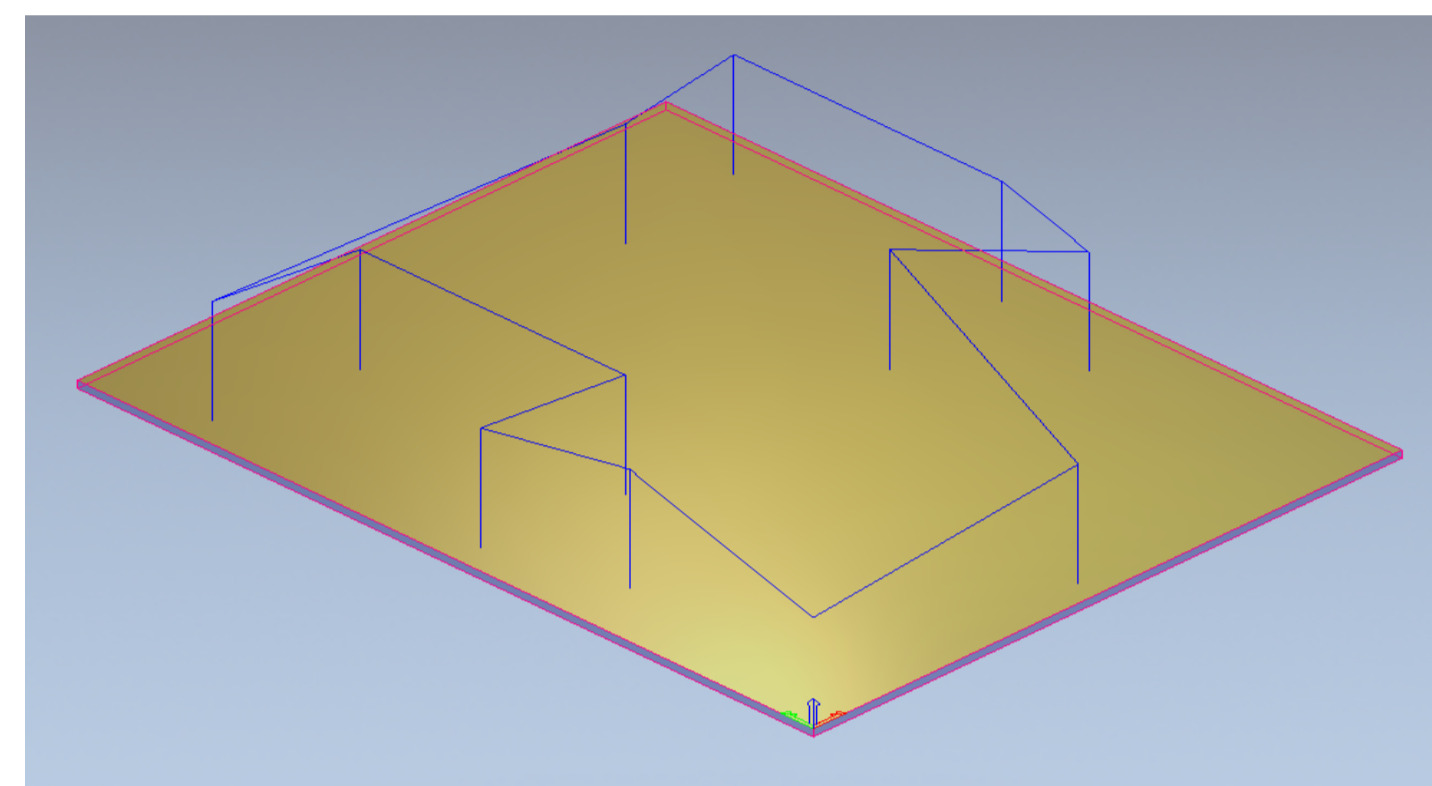
MP.ПМКМ-032.02.00.007					Лист	Масса	Масштаб
Опора					Н		1:1
Monofilament PLA					Лист	Листов	1
Изм.	Лист	№ докум.	Подп.	Дата			
Разраб.	Веркалець А.А.						
Проб.	Лукаць Т.В.						
Т.контр.	Лукаць Т.В.						
Н.контр.	Лукаць Т.В.						
Читб.	Панчук В.Г.						



Операція 010 Фрезерування пазу



Операція 010 Фрезерування контуру



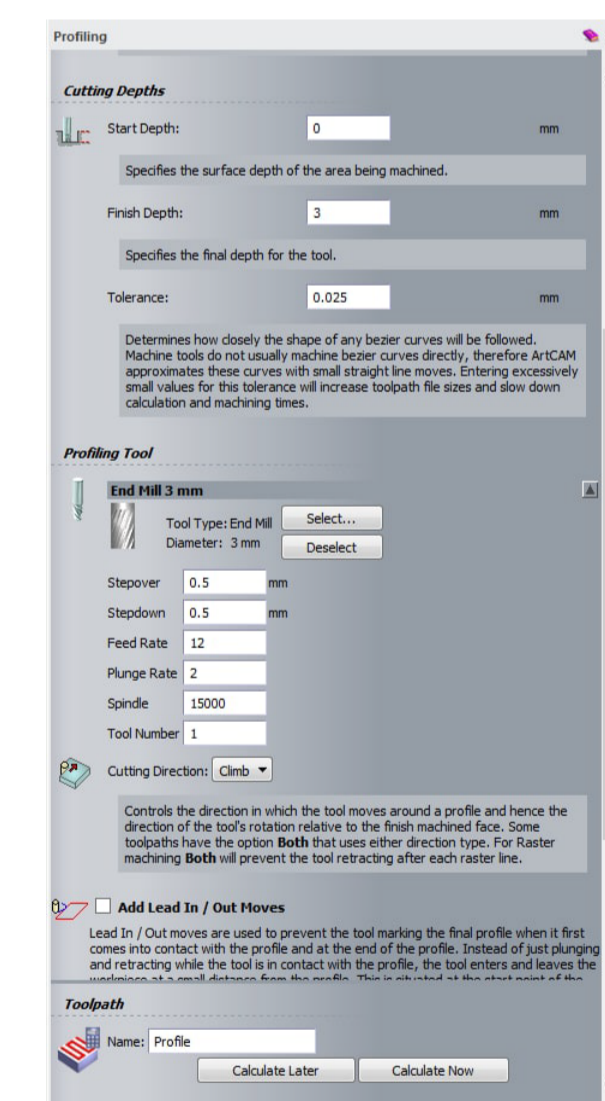
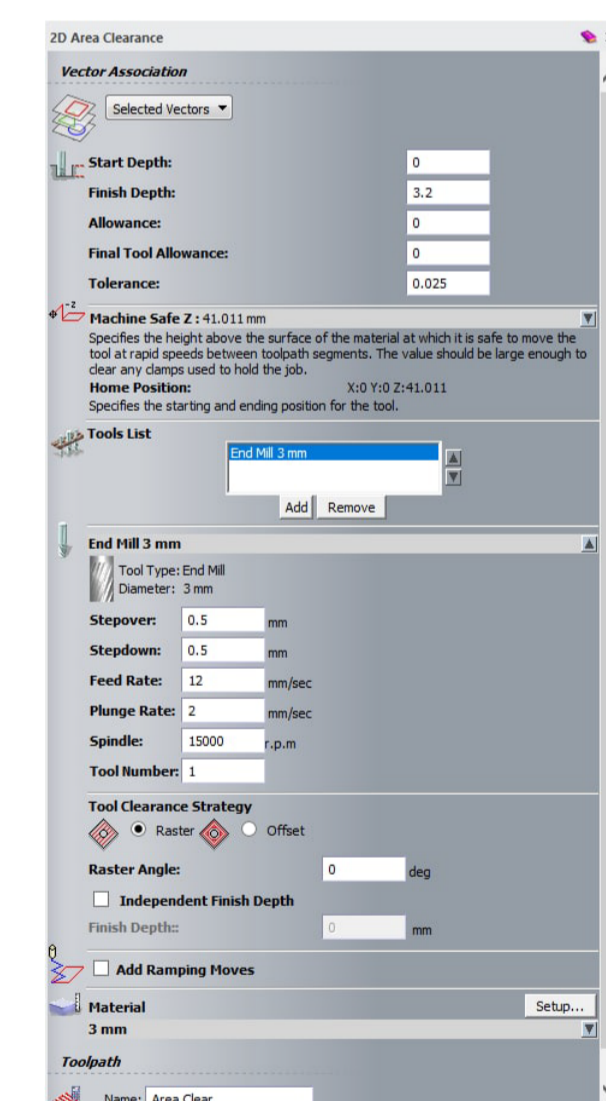
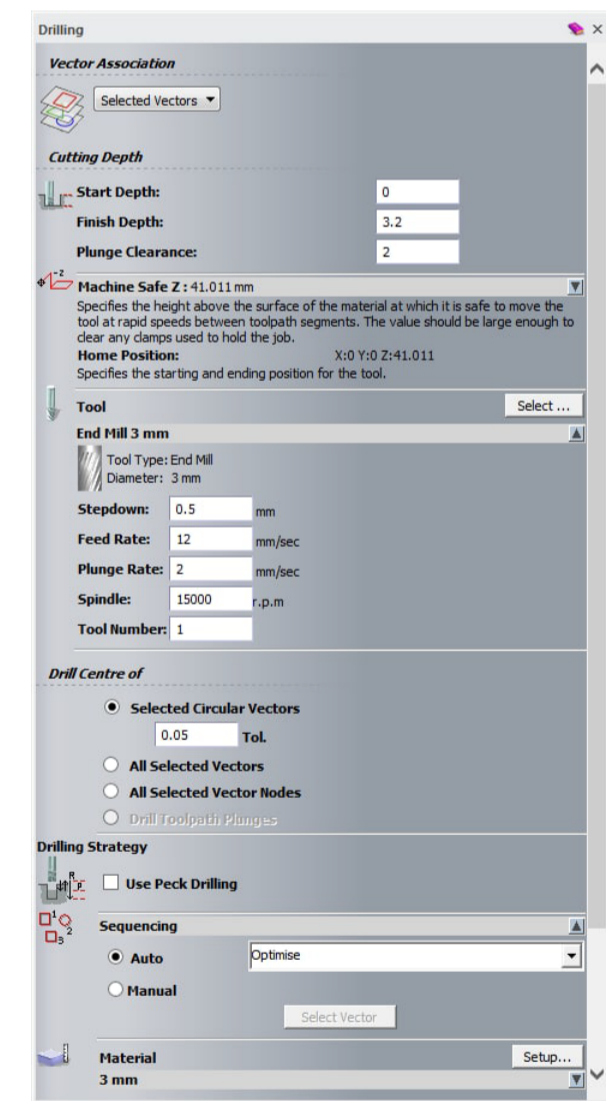
Операція 010 Свердління групи отворів

Уловите своє

Зображення траєкторій руху всіх інструментів

Table with 10 columns and 100 rows of numerical data representing tool movement trajectories.

Керуюча програма для верстата



Налаштування режимів різання в середовищі ArtCAM

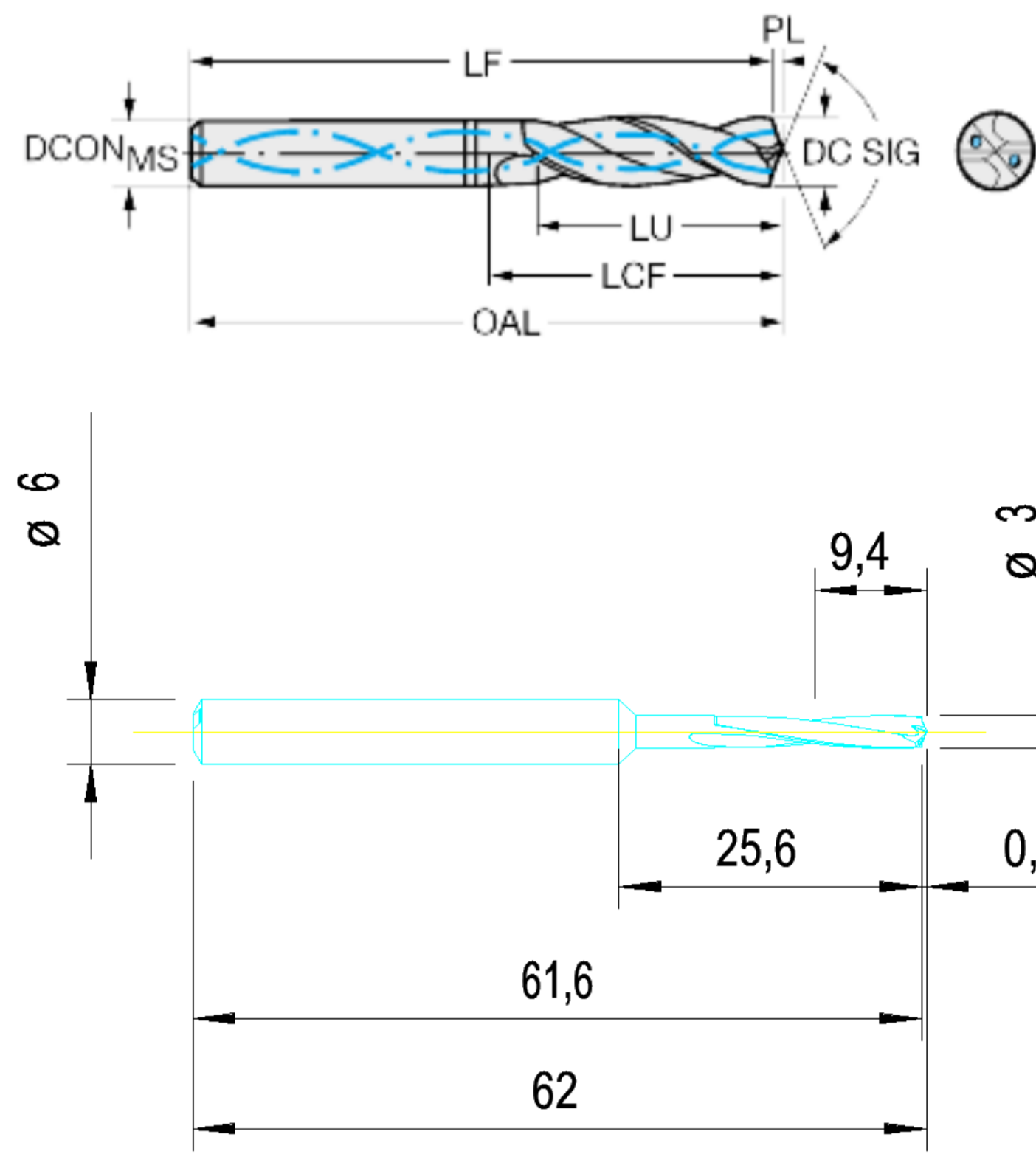
Technical drawing header and table with fields for part name, material, and other specifications.

МР.ПМКМ-032.04.00.000

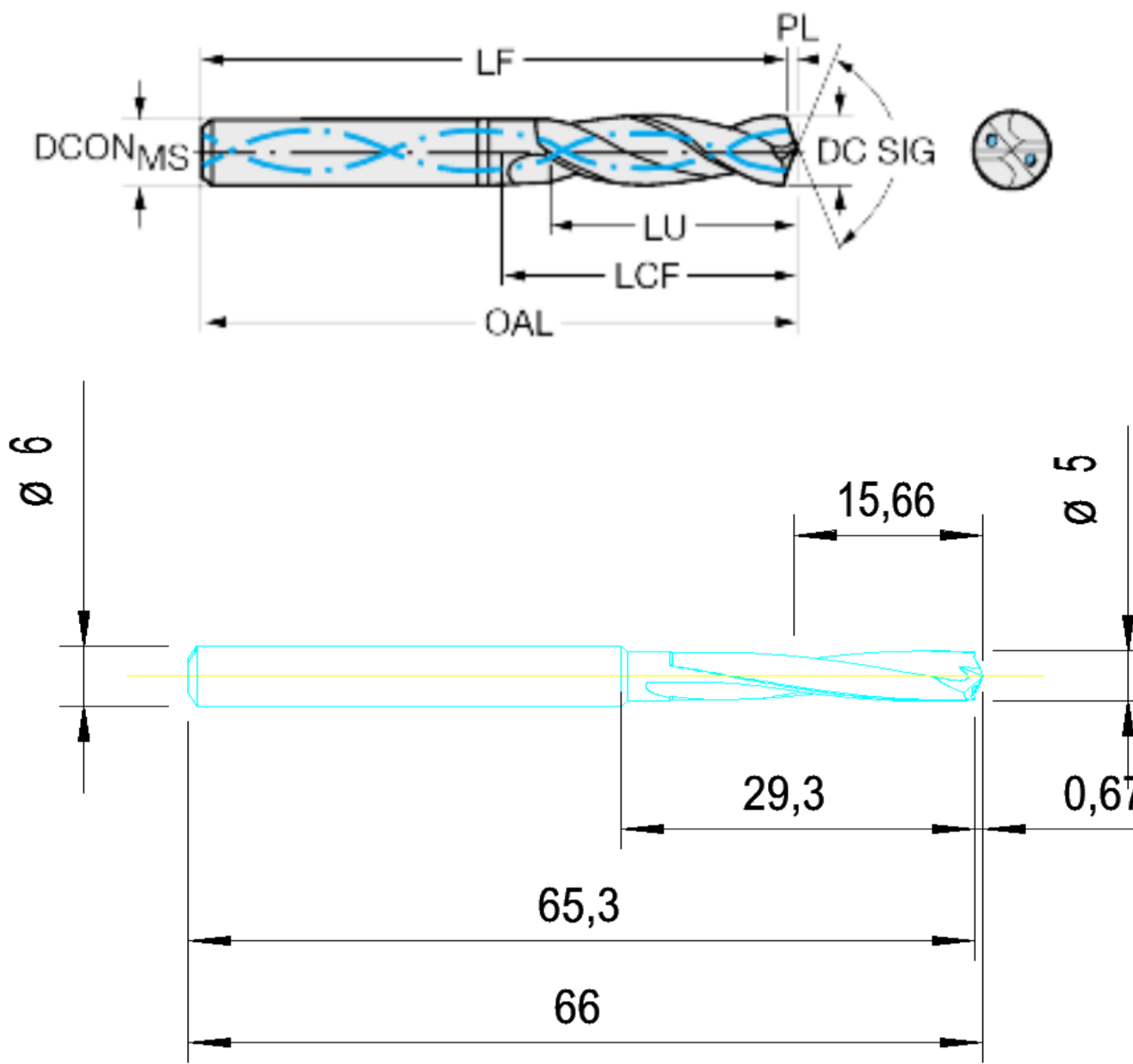
Траєкторія обробки деталі «Слобода верня»

Table with columns for 'Лист' (Sheet), 'Маса' (Mass), and 'Масштаб' (Scale).

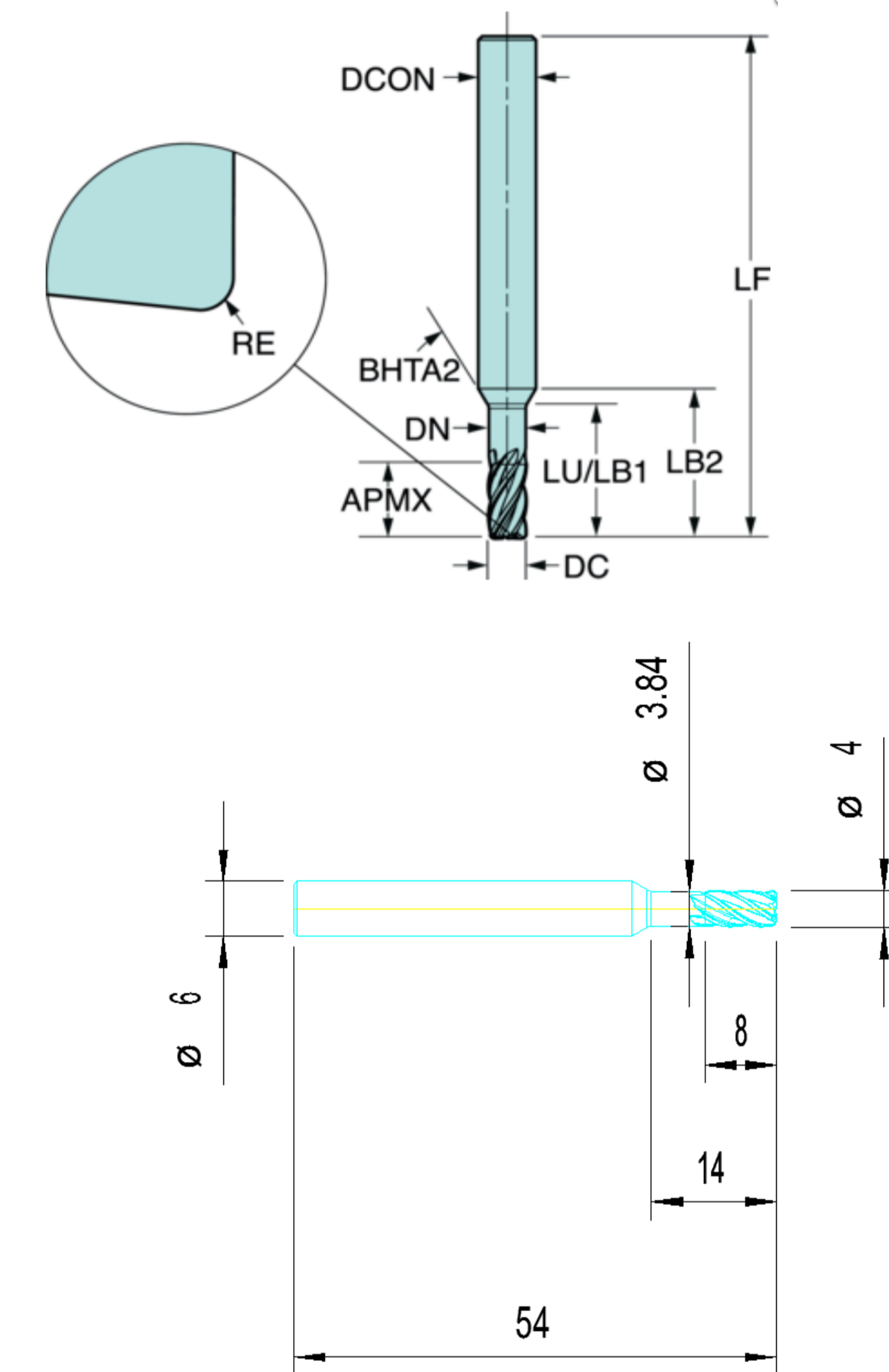
ПМКМ-21-1



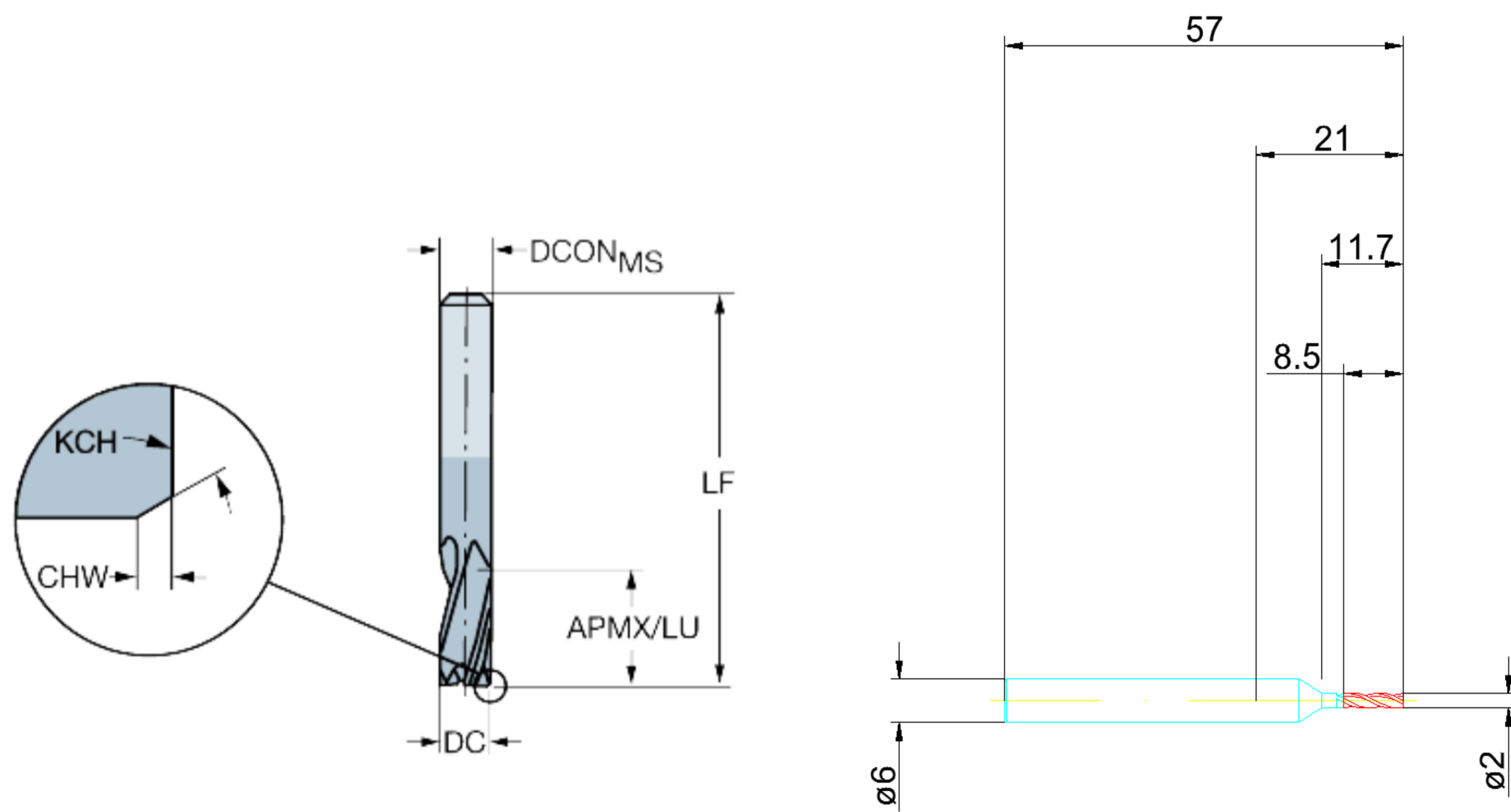
Свердло Coro-Drill 860.1-0300-009A1-NM H10F



Свердло Coro-Drill 860.1-0500-015A1-NM H10F



Фреза CoroMill 1K335-0400-050-XC 1730



Фреза CoroMill 1P251-0200-XA 1630

					MP.ПМКМ-038.05.00.000			
Имя	Лист	№ документа	Дата	Лист	Инструментальное	Лист	Масса	Масштаб
Разработ	Верхалець А.А.				оснащення верстату	Н		
Проб	Луцкань Т.В.					Лист		
Т. констр								
Н. констр								
Синд								
					ФНТУНГ			
					ПМКМ-21-1			

# Керуюча програма в середовищі Arduino IDE

```
// БІБЛІОТЕКА для роботи датчика ехо локатора
#include <iarduino_HC_SR04_int.h>
// БІБЛІОТЕКА для роботи з кроковим двигуном #include <Stepper.h> //
Номера виходів до яких підключені датчики (лінії, відстані) і MotorShield (дані введени в констан-
ти потрібно буде змінити при підключенні візка)
const uint8_t datLL = A3; // Датчик лінії, який знаходиться зліва
const uint8_t datLC = A4; // Датчик лінії, який знаходиться посередині
const uint8_t datLR = A5; // Датчик лінії, який знаходиться зправа const
uint8_t datV_TRIG = 2; //
Вихід ехолокатора TRIG const uint8_t datV_ECHO = 3; // Вихід ехолокатора ECHO
const uint8_t motSh_LH = 12; // Вихід направлення до якого підключений
лівий мотор
const uint8_t motSh_LE = 11; // Вихід ШІМ(широтна імпульсна модуляція) до якого підключений
лівий мотор
const uint8_t motSh_RE = 10; // Вихід ШІМ до якого підключений правий
мотор
const uint8_t motSh_RH = 13; // Вихід напрвлення до якого підключений
правий мотор
const float STEP_PER_REV = 32;
const float GEAR_RED = 64; //potribno glanytu na krokDV GEAR REDACTION
const float STEPS_PER_OUT_REV = STEP_PER_REV * GEAR_RED; int StepsRequired;
//
Stepper steppermotor(STEP_PER_REV, 6, 8, 7, 9); // TREBA PIDIBRATU SVOII VUHODU
// iarduino_HC_SR04_int HCSR(datV_TRIG, datV_ECHO); // Для роботи з дідліотекою потрібно
вказати об'єкт HCSR (і виходи еходатчика) // Значення які ми встановлюєм:
const uint16_t datLOL = 681; // Дані які датчик показує на лінії (це
значення потрібно настроїти)
const uint16_t datLOOL = 174; // Дані які датчик показує поза лінією (це значення потрібно на-
строїти)
const uint8_t MAXSP = 255; // Максимальна швидкість
const uint32_t TimerST = 2000; // Впродовж записаного часу візок не буде зупинятись коли
він поза лінією
const uint8_t Povorot = 10; // Значення швидкості реакції на поворот (чим більше значення
тим різкіше він реагує) const uint8_t Vidst = 10; // Мінімальне значення в сантиметрах, коли
візок буде реагувати на перешкоду
const bool OboratMot[2] = {1,1}; // В яку сторону будуть крутитися
двигуни (залежить від полярності, тобто від під'єднання моторів)
// Значення які будуть рахуватись:
uint8_t MotorSP[2]; // Швидкість на моторах (0 - правий мотор, 1 - лівий
мотор)
uint16_t datLMID; // Середнє значення показу датчиків лінії uint8_t datLBIT; // Константа, по-
трібна буде для подітного виводу
значення датчиків лінії bool flgLine; // Прапорець, який характеризуватиме лінію (0 - темна
лінія,
1 - світла лінії)
int8_t flgTurning; // Прапорець, який характеризуватиме поворот (0 - прямо ідем, -1 - наліво,
+1 - направо)
bool flgPWM; // Прапорець, який характеризуватиме ШІМ
bool flgStop; // Прапорець, який характеризуватиме зупинку (0 - без зупинки, 1 - зупинка)
bool flgDistance; // Прапорець, який характеризуватиме відстань до
перешкоди (0 - нема перешкоди, 1 - є перешкода)
uint32_t TimerML; // Час впродовж якого була проведена попередня операція void setup(){
// Розпізнавання кольору лінії
flgLine = (datLOOL>datLOL); // Якщо значення за лінією більше тоді буде
встановлюватись 1, а це означає,що ми пересуваємся по світлій лінії на
темному фоні
```

```
// Якщо навпаки, то буде скидатись на 0 і це означатиме темна лінія на
світлому фоні
// Визначаєм середнє значення показників датчика лінії (на ній і поза нею)
// Так як у нас може іздити по темній лінії на світлому фоні і навпаки
потрібні різні формули для обрахування
if(flglLine){datLMID = datLOL + (datLOOL - datLOL)/2;} // Для світлої лінії
else {datLMID = datLOOL + (datLOL - datLOOL)/2;} // Для темної лінії
// Присвоюєм моторам конкретні значення швидкості (1 - лівий мотор, 0 - правий мотор)
MotorSP[1] = MAXSP; MotorSP[0] = MAXSP; // Встановлюєм прапорець ШІМ, скидаєм значення
прапорця повороту,
зупинки і відстані
flgPWM=1; flgTurning=0; flgStop=0; flgDistance=0;
// Встановлюєм режими роботи виходів і направлення обертів моторів
pinMode (datLL, INPUT ); // Встановлюєм вихід лівого датчика лінії як вхід (для того щоб полу-
чити дані які він нам покаже)
pinMode (datLC, INPUT ); // Встановлюєм вихід лівого датчика лінії як вхід pinMode (datLR, INPUT);
// Встановлюєм вихід лівого датчика лінії як вхід pinMode (motSh_LH, OUTPUT ); // Встаноб-
люєм вихід motSh_LH як
вхід (для керування направленням оборотів лівого мотору)
pinMode (motSh_LE, OUTPUT ); // Встановлюєм вихід motSh_LE як
вхід (для керування швидкістю обертів лівого мотору)
pinMode (motSh_RE, OUTPUT ); // Встановлюєм вихід motSh_RE як
вхід pinMode (motSh_RH, OUTPUT ); // Встановлюєм вихід motSh_RH як
вхід digitalWrite(motSh_LH, OboratMot[1]); // Встановлюєм на виході motSh_LH значення
OboratMot[1] (направлення обертів лівого мотору)
digitalWrite(motSh_RH, OboratMot[0]); // Встановлюєм на виході motSh_RH значення OboratMot[0]
// Записуєм показання центрального датчика лінії в константи datLOL і datLOOL
Serial.begin(9600); while(!Serial){} // Зчитування/Передача даних через
порт на швидкості 9600 б/с
Serial.println(analogRead(datLC)); // Виводим показання датчика на
екран для запису в константи
// Встановлюєм затримку 2 сек і поновлюєм час з моменту виконання останньої операції
delay(2000);
TimerML = millis();
}
void loop(){
if (MotorSP[1] = 0, MotorSP[0] = 0, flgDistance = 0, flgStop = 0){
steppermotor.setSpeed(1);
StepsRequired = 4;
steppermotor.step(StepsRequired);
delay(2000);
}
/ Serial.println(HCSR.distance()); //
// delay(1000); //
// Заносим показання трьох датчиків в одну змінну встановлюючи прості значення (1 - датчик
на лінії, 0 - датчик поза лінією)
datLBIT = 0; // Спочатку скидаєм значення даної змінної
datLBIT |= ((analogRead(datLL)>datLMID)^flgLine)<<2; // Встановлюєм 2
біт змінної datLBIT в 1, якщо лівий датчик знаходиться на лінії,якщо ні то
залишаєм біт в 0
datLBIT |= ((analogRead(datLC)>datLMID)^flgLine)<<1; // Встановлюєм 1
біт змінної datLBIT в 1, якщо центральний датчик знаходиться на лінії,якщо ні то залишаєм біт
в 0
datLBIT |= ((analogRead(datLR)>datLMID)^flgLine)<<0; // Встановлюєм 0
біт змінної datLBIT в 1, якщо правий датчик знаходиться на лінії,якщо ні то
залишаєм біт в 0
```

```
// Як візок буде себе поводити відносно значення datLBIT
switch(datLBIT){
case 0b000: flgPWM=flgTurning; flgStop=!flgTurning; break; // 000
- Всі датчики поза лінією case 0b010: TimerML=millis(); flgPWM=flgTurning; flgTurning=0; flgStop=0;
break; // 010 - Тільки центральний датчик на лінії case 0b100: // 100 - Тільки лівий датчик
на лінії case 0b110: TimerML=millis(); flgPWM=1; flgTurning=-1; flgStop=0; break; // 110 - Цен-
тральний і лівий датчик на лінії case 0b001: // 001 - Тільки правий датчик на лінії
case 0b011: TimerML=millis(); flgPWM=1; flgTurning=1; flgStop=0; break; // 011 - Правий і цен-
тральний датчик на лінії
default: lgPWM=1; break; }
if(TimerML>millis()) { TimerML=0;} // Позабляємся переповнення millis()
if(TimerML+TimerST<millis()){ flgPWM=1; flgTurning=0; flgStop=1; } // Зупиняємся якщо лінія втра-
чена на дільше часу ніж 2 сек
if(HCSR.distance()<Vidst){ TimerML=millis(); flgPWM=1; flgDistance=1; } // Зупиняємся якщо спереди
перешкода
else if(flgDistance) { TimerML=millis(); flgPWM=1; flgDistance=0; } // Продовжуєм рух якщо переш-
кода пропала
// Встановлюєм ШІМ для моторів
if(flgPWM){flgPWM=0; switch(flgTurning){ // Швидкість лівого мотору: Швидкість правого
мотору:
case -1: if(MotorSP[1]>0){MotorSP[1]--;} MotorSP[0]=MAXSP; break; // Зменшуєм швидкість лівого
мотору коли повертаєм
case 0: MotorSP[1]=MAXSP; MotorSP[0]=MAXSP; break; // Однакова швидкість коли ідем прямо
case 1: MotorSP[1]=MAXSP; if(MotorSP[0]>0){MotorSP[0]--;}
break; // Зменшуєм швидкість правого мотору коли повертаєм
} if(flgStop){ MotorSP[1]=0; MotorSP[0]=0;} // Зупиняємся якщо
flgStop = 1
if(flgDistance){ MotorSP[1]=0; MotorSP[0]=0;} // зупиняємся якщо flgDistance = 1
// Обраховані значення передаєм на вхід до моторів
analogWrite(motSh_LE, MotorSP[1]);
analogWrite(motSh_RE, MotorSP[0]);
}
}
```

MP.ПМКМ-032.06.00.000					
Керуюча програма для Arduino UNO				Лист	Масштаб
Лист	Масштаб				
Лист	Листов	ІФНТУНГ ПМКМ-21-1			

Лист

Лист

Лист

Лист

Лист

Лист

Лист