

КВАЛІФІКАЦІЙНА РОБОТА

бакалавр

Освітній рівень

Система автоматичного визначення розмірів в процесі токарної обробки деталей з використанням мікроконтролера ESP32-CAM

Назва теми

КвРКІ 200102.20.01.02.ПЗ

Шифр

Галузь знань 12 «Інформаційні технології»

Шифр, назва

Спеціальність 123 «Комп'ютерна інженерія»

Шифр, назва

Освітня програма «Комп'ютерна інженерія та програмування»

Назва

Виконав: студент IV курсу, група KI2-20-1

Підпис

С.С. Баришніков

Ініціали, прізвище

Керівник

Підпис, дата

I.O. Засорнова

Ініціали, прізвище

Нормоконтролер

Підпис, дата

I.O. Засорнова

Ініціали, прізвище

До захисту допускаю:

Зав. кафедри комп'ютерної інженерії та інформаційних систем

Підпис

T.O. Говорущенко

Ініціали, прізвище

24 червня 2024 р.

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

Факультет ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

Кафедра КОМП'ЮТЕРНОЇ ІНЖЕНЕРІЇ ТА ІНФОРМАЦІЙНИХ СИСТЕМ

Освітній рівень БАКАЛАВР

Галузь знань 12 ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ

Спеціальність 123 КОМП'ЮТЕРНА ІНЖЕНЕРІЯ

Освітня програма «КОМП'ЮТЕРНА ІНЖЕНЕРІЯ ТА ПРОГРАМУВАННЯ»

ЗАТВЕРДЖУЮ

Зав. кафедри Т.О. Говорушенко

" 15 " 02 2024 р.

ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ БАКАЛАВРА

Баришнікову Сергію Сергійовичу

Прізвище, ім'я, по батькові студента

1. Тема проекту (роботи) Система автоматичного визначення розмірів в процесі токарної обробки деталей з використанням мікроконтролера ESP32-SAM

Керівник проекту (роботи) к.т.н., доцент Засорнова І.О.

Прізвище, ім'я, по батькові, науковий ступінь, весце звання

Затверджена наказом ректора університету від 15.02.2024 р. № 8

2. Строк подання студентом проекту (роботи) на кафедру 01.06.2024 р.

3. Вихідні дані до проекту (роботи) Завдання на кваліфікаційну роботу

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

Огляд відомих методів та їх рішення _____

Вибір наявного програмного забезпечення системи автоматичного визначення розмірів в процесі токарної обробки деталей _____

Програмна реалізація та тестування програмно-технічного засобу та висновки


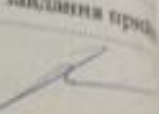


5. Перелік графічного матеріалу (із зазначенням обов'язкових креслень) _____

Блок-схема роботи програми _____

Схема з'єднань мікроконтролера з камерою _____

Електрична схема мікроконтролера _____

6. Консультанти розділів дипломного проекту

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Нормоконтроль	Засорнова І.О., доцент кафедри КПС		
Антиплагіат	Нічепорук А.О., доцент кафедри КПС		

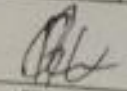
7. Дата видачі завдання « 15 » 02 2024 р.

КАЛЕНДАРНИЙ ПЛАН

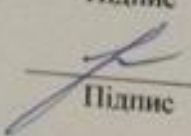
№х/п	Назва етапів (розділів) дипломного проекту (роботи)	Термін виконання етапів проекту (роботи)	Примітки
1	Вибір напрямку дослідження та узгодження тематики кваліфікаційної роботи з керівником	15.02.2024	виконано
2	Ознайомлення з предметною областю; формулювання мети та задач дослідження; визначення об'єкта та предмета дослідження	15.02.2024	виконано
3	Робота над розділом 1 – Огляд відомих методів та їх рішення	15.03.2024	виконано
4	Робота над розділом 2 – вибір наявного програмного забезпечення системи автоматичного визначення розмірів в процесі токарної обробки деталей	15.04.2024	виконано
5	Робота над розділом 3 – Програмна реалізація та тестування програмно-технічного засобу та висновки	1.05.2024	виконано
6	Оформлення пояснювальної записки згідно вимог	25.05.2024	виконано
7	Попередній захист ВКР	30.05.2024	виконано
8	Захист ВКР на засіданні ЕК	Червень 2024 року	

Студент

Керівник роботи


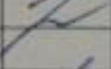
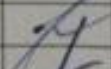
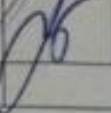

Підпис

С.С. Барішнікова
Ініціали, прізвище


Підпис

І.О. Засорнова
Ініціали, прізвище

№ р я д к а	ф о р м а т	Позначення	Найменування	К і л - л и с т і в	№ с к з	П р и м і т к а
			<u>Текстові документи</u>			
1		КвРКІ 200102.20.01.02.ПЗ	Пояснювальна записка	61		
			<u>Графічні матеріали</u>			
2		КвРКІ 200102.20.01.02. E8	Блок-схема роботи	1		
3		КвРКІ 200102.20.01.02.E8	Схема з'єднань мікроконтролера з камерою	1		
4		КвРКІ 200102.20.01.02.E2	Електрична схема мікроконтролера	1		

КвРКІ 200102.20.01.02.ВП							
Зм	Арж	№ докум	Підпис	Дата			
Розробив		Баршніков					
Перевір.		Засорнова					
Н. контр.		Засорнова					
Затв.		Головченко		24.08			
Система автоматичного визначення розмірів в процесі токарної обробки деталей з використанням мікроконтролера ESP32-CAM					Літера	Аркуш	Аркушів
					У	1	1
					ХНУ, КІ2-20-1		

АНОТАЦІЯ

Тема кваліфікаційної роботи: «Система автоматичного визначення розмірів в процесі токарної обробки деталей з використанням мікроконтролера ESP32-CAM».

Автор роботи: Баришніков Сергій Сергійович

Керівник роботи: Засорнова Ірина Олександрівна

Пояснювальна записка: 61 с., 32 рис., 1 табл., 4 дод., 57 джерел.

Графічна частина: 3 креслення.

**ESP32CAM, СИСТЕМА АВТОМАТИЧНОГО ВИЗНАЧЕННЯ РОЗМІРІВ,
ТОКАРНА ОБРОБКА.**

Метою дипломної роботи є забезпечення автоматичного визначення розмірів в процесі токарної обробки деталей з використанням мікроконтролера ESP32-CAM, а також оцінка механізмів обробки інформації у програмно-технічному засобі для забезпечення достатньої точності та ефективності обробки.

Об'єктом дослідження у цьому проєкті є процес автоматичного визначення розмірів деталей під час токарної обробки з використанням мікроконтролера ESP32-CAM.

Предмет дослідження у цьому проєкті є автоматична система для визначення розмірів деталей під час токарної обробки з використанням мікроконтролера ESP32-CAM.

Практичне значення має спроектована та реалізована система автоматичного визначення розмірів в процесі токарної обробки деталей з використанням мікроконтролера ESP32-CAM







Підпис студента

30.05.2024

Дата

ЗМІСТ

ВСТУП	4
1 ОГЛЯД ВІДОМИХ МЕТОДІВ ТА ЇХ РІШЕННЯ.....	6
1.1 Поняття мікроконтролеру. Складові мікроконтролерів та сфери їх використання.....	6
1.2 Загальні відомості про токарні верстати. Компоненти токарних верстатів. Основні типи токарних верстатів.....	9
1.3 Використання мікроконтролерів у токарних верстатах.....	17
1.4 Висновки та постановка задачі.....	19
2 ВИБІР НАЯВНОГО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ СИСТЕМИ АВТОМАТИЧНОГО ВИЗНАЧЕННЯ РОЗМІРІВ В ПРОЦЕСІ ТОКАРНОЇ ОБРОБКИ ДЕТАЛЕЙ.....	22
2.1 Технічні характеристики мікроконтролеру	22
2.2. Вибір камери для виконання роботи.....	24
2.3. Загальні відомості про мікроконтролер ESP32-CAM.....	27
2.4 Сфери використання мікроконтролеру ESP32-CAM.....	28
2.5 Порівняння мікроконтролеру ESP32-CAM з іншими за використанням у сфері автоматичної обробки деталей.....	30
2.6 Використання мікроконтролеру ESP32-CAM в системах автоматичної обробки деталей. Переваги та недоліки у порівнянні з іншими існуючими системами	36
2.7 Існуючі аналоги систем з використанням ESP32-CAM та інших мікроконтролерів.....	37
2.8 Висновки.....	39

КвРКІ 200102.20.01.02.ПЗ				
Зм	Арк.	№ док.ум.	Підпис	Дата
Виконав		Баричніков С.С.		
Перевір.		Засорнова І.О.		
Н.контр.		Засорнова І.О.		
Затвер.		Гливорубенко Т.О.		24.05
Система автоматичного визначення розмірів в процесі токарної обробки деталей з використанням мікроконтролера ESP32-CAM				
		Літера	Аркуш	Аркушів
		у	2	61
ХНУ К12-20-1				

3	ПРОГРАМНА РЕАЛІЗАЦІЯ ТА ТЕСТУВАННЯ	
	ПРОГРАМНОТЕХНІЧНОГО ЗАСОБУ ТА ВИСНОВКИ	40
3.1	Встановлення необхідних ресурсів та ArduinoIDE.....	40
3.2	Реалізація функцій для виконання алгоритму.....	49
3.3	Опис блок-схеми, схеми з'єднань мікроконтролера з камерою, електричної схеми	57
3.4	Розрахунок матеріальних витрат та напрямки вдосконалення.....	59
3.5	Висновки.....	60
	ВИСНОВКИ	61
	ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ	62
	ДОДАТОК А Блок-схема роботи програми.....	68
	ДОДАТОК Б Схема з'єднань мікроконтролера з камерою.....	69
	ДОДАТОК В Електрична схема мікроконтролера	70
	ДОДАТОК Г Лістинг коду	71

ВСТУП

У сучасному виробництві все більшу увагу приділяється автоматизації та контролю якості продукції. Це дозволяє підвищити продуктивність, точність та економити час. Одним із важливих етапів токарної обробки деталей є визначення їх розмірів. Традиційно це робиться вручну за допомогою штангенциркуля або інших вимірювальних інструментів. Цей процес може бути трудомістким, схильним до помилок та забирати багато часу.

Система автоматичного визначення розмірів в процесі токарної обробки деталей з використанням мікроконтролера ESP32-CAM - це інноваційне рішення, яке дозволяє автоматизувати цей процес та значно підвищити його ефективність. Основні завдання включають в себе вивчення принципів роботи технології, визначення її переваг порівняно з традиційними методами вимірювань розмірів, а також розробку та оптимізацію алгоритмів для точного та надійного визначення геометричних параметрів деталей

Це дослідження є актуальним у зв'язку з постійним стрімким розвитком технологій у виробництві та підвищеним попитом на автоматизовані системи контролю та вимірювань. У сучасному виробництві важливим завданням є автоматизація процесів та підвищення точності технологічних операцій. Однією з ключових галузей, що відіграє важливу роль у вирішенні цих завдань, є токарна обробка деталей. Сучасні технології дозволяють не лише здійснювати обробку з високою продуктивністю, але і впроваджувати в системи автоматичного контролю для забезпечення високої якості продукції.

Однією з інноваційних технологій у цьому контексті є система автоматичного визначення розмірів в процесі токарної обробки з використанням мікроконтролера ESP32-CAM. Ця технологія поєднує в собі переваги мікроконтролерної обробки даних та високоякісного зображення, що забезпечує точність і надійність визначення розмірів оброблювальних деталей.

					КВРКІ 200102.20.01.02.ПЗ	Арк. 2
Зм.	Арк.	№ докум.	Підпис	Дата		

ESP32-CAM – це компактний мікроконтролер, який об'єднує в собі потужний мікроконтролер ESP32-CAM та камеру . Використання цього пристрою дозволяє отримувати візуальні дані в реальному часі та використовувати їх для визначення розмірів оброблювальних деталей . Це відкриває нові перспективи для автоматизації та вдосконалення процесів токарної обробки.

Метою дипломної роботи є забезпечення автоматичного визначення розмірів в процесі токарної обробки деталей з використанням мікроконтролера ESP32-CAM, а також оцінка механізмів обробки інформації у програмно-технічному засобі для забезпечення достатньої точності та ефективності обробки.

Об'єктом дослідження у цьому проєкті є процес автоматичного визначення розмірів деталей під час токарної обробки з використанням мікроконтролера ESP32-CAM.

Предмет дослідження у цьому проєкті є автоматична система для визначення розмірів деталей під час токарної обробки з використанням мікроконтролера ESP32-CAM.

Практичне значення має спроектована та реалізована система автоматичного визначення розмірів в процесі токарної обробки деталей з використанням мікроконтролера ESP32-CAM, яка передбачає визначення розмірів та надсилання результатів на пошту.

					КВРКІ 200102.20.01.02.ПЗ	Арк.
						3
Зм.	Арк.	№ докум.	Підпис	Дата		

1 ОГЛЯД ВІДОМИХ МЕТОДІВ ТА ЇХ РІШЕННЯ

1.1 Поняття мікроконтролеру. Складові мікроконтролерів та сфери їх використання

Мікроконтролер є компактною інтегральною схемою, яка об'єднує функції мікропроцесора, пам'яті та периферійних пристроїв (таких як таймери, засоби введення-виведення, аналого-цифрові і цифро-аналогові перетворювачі, комунікаційні інтерфейси тощо). Зазвичай мікроконтролери використовуються для керування різноманітними пристроями або системами, де вони виконують завдання вимірювання, обробки сигналів, керування введенням-виведенням та інших функцій.

Перший мікроконтролер був створений у 1971 році Г.Буном. У 1980 році компанія Intel представила мікроконтролер i8048, а трохи пізніше того ж року - i8051 [1]. Завдяки вдало підбраному набору периферійних пристроїв, можливості гнучкого вибору між зовнішньою або внутрішньою програмною пам'яттю та прийнятній ціні, цей мікроконтролер здобув популярність на ринку. З технологічного погляду мікроконтролер i8051 був дуже складним для свого часу – його кристал містив 128 тисяч транзисторів, що у чотири рази перевищувало кількість транзисторів у 16-розрядному мікропроцесорі i8086 [2].

Завдяки вбудованій пам'яті та периферійним пристроям, мікроконтролери можуть працювати автономно і виконувати завдання без потреби у зовнішньому з'єднанні з комп'ютером або іншими пристроями. Їх невеликі розміри, низька вартість і низький рівень енергоспоживання роблять їх ідеальними для застосувань у середовищах з обмеженими ресурсами, потужністю або простором.

Незалежно від своїх функціональних можливостей, мікроконтролер зазвичай включає такі основні компоненти [3,4]:

					КВРКІ 200102.20.01.02.ПЗ	Арк.
						4
Зм.	Арк.	№ докум.	Підпис	Дата		

- мікропроцесор : обчислювальний блок мікроконтролера, який відповідає за виконання операцій обробки даних і керування. Його можливості включають виконання арифметичних, логічних та керуючих операцій;

- вбудована пам'ять: мікроконтролер має вбудовану пам'ять для збереження програмного коду, даних та конфігураційної інформації. Ця пам'ять поділяється на два типи: програмна пам'ять для зберігання програмного коду і змінних даних, а також оперативна пам'ять для тимчасового зберігання даних під час виконання програми;

- периферійні пристрої: мікроконтролер має різноманітні периферійні пристрої для забезпечення зовнішнього зв'язку та взаємодії з навколишнім середовищем. Серед них можуть бути таймери, лічильники, пристрої введення-виведення, аналого-цифрові та цифро-аналогові перетворювачі, універсальні асинхронні/синхронні приймачі-передавачі, шина взаємозв'язку, прямий доступ до пам'яті та інші;

- годинник/таймер: деякі мікроконтролери можуть мати вбудований годинник або таймер, що дозволяє відстежувати час та генерувати події з заданою частотою або після певного інтервалу часу;

- інтерфейси зв'язку: деякі мікроконтролери підтримують різні інтерфейси зв'язку, такі як USB, CAN, Ethernet, Wi-Fi, Bluetooth тощо, що дозволяє їм спілкуватися з іншими пристроями або мережами.

Мікроконтролери широко використовуються у різних галузях сучасного життя і можуть мати різні додаткові складові, залежно від їхніх можливостей та призначення [5-8].

- медичні пристрої: у медичних пристроях, таких як монітори серцевого ритму, імплантовані пристрої та пристрої для вимірювання кров'яного тиску, мікроконтролери використовуються для контролю за роботою пристрою та обробки зібраних даних;

					КВРКІ 200102.20.01.02.ПЗ	Арк.
						5
Зм.	Арк.	№ докум.	Підпис	Дата		

- керування роботами: від простих побутових роботів до складних промислових систем, мікроконтролери використовуються для керування рухом, збору даних з датчиків та прийняття рішень;

- системи відеоспостереження: мікроконтролери використовуються в системах відеоспостереження для управління камерами, обробки та аналізу отриманих даних, запису та зберігання відеоданих та взаємодії з іншими підсистемами в системі;

- системи автоматизації в промисловості: мікроконтролери використовуються у системах автоматизації та керування в промисловості для моніторингу параметрів, керування процесами виробництва та збору даних;

- пристрої інтернету речей (IoT): у розробці пристроїв IoT мікроконтролери використовуються для забезпечення підключення до мережі Інтернет, обміну даними з хмарними сервісами та керування підключеними пристроями;

- системи керування енергопостачанням: у системах керування енергопостачанням, таких як сонячні панелі, енергозберігаючі пристрої та системи "розумний будинок", мікроконтролери використовуються для ефективного використання енергії та моніторингу споживання;

- системи автомобілів: мікроконтролери використовуються у системах автомобілів для управління двигуном, електронною панеллю приладів, системами безпеки, керування передачами та іншими важливими функціями автомобіля;

- споживчі електронні пристрої: мікроконтролери використовуються у різних споживчих електронних пристроях, таких як смартфони, телевізори, музичні програвачі та побутові прилади. Вони забезпечують управління функціями пристрою, обробку введення та виведення даних та взаємодію з користувачем. Мікроконтролери знаходять широке застосування в різних галузях завдяки своїм можливостям та доступності, і їхнє використання.

					КВРКІ 200102.20.01.02.ПЗ	Арк.
						6
Зм.	Арк.	№ докум.	Підпис	Дата		

1.2 Загальні відомості про токарні верстати. Компоненти токарних верстатів. Основні типи токарних верстатів

Токарний верстат - це машина, в основному використовує інструмент для повороту обертального заготовок. Токарно-фрезерні верстати, шпінгалети, шпінгалети, крани, штампи та шліфувальні інструменти також можуть використовуватися для обробки на токарних верстатах. Металообробні верстати використовуються переважно для верстатів, дисків, рукавів та інших заготовок із обертовими поверхнями. Вони є найбільш широко використовуваним типом верстатів на механічних виробничих і ремонтних заводах [9].

Токарні верстати були винайдені і застосовувалися ще в глибоку давнину. Вони були дуже прості по пристрою, дуже недосконалі в роботі і мали спочатку ручної, а згодом ніжний привід [10]. Одним із головних модернізаторів токарного верстата у середньовіччі був Леонардо да Вінчі, який винайшов токарний верстат у якого рух шпинделя відбувався за допомогою ножної педалі. У 1571 Жак Бессон сконструював токарний верстат із власним самостійним механічним подаванням ходового гвинта. У 17 ст. з'явилися токарні верстати, у яких виріб, що оброблявся, приводився до руху уже не м'язовою силою токаря, а за допомогою водяного колеса, але різець як і раніше, тримав у руці токар. На початку 18 ст. токарні верстати все частіше стали використовувати для різання металів, а не дерева, отже проблема твердого кріплення різця та переміщення його уздовж оброблюваної поверхні, була досить важливою. Вже починаючи з 1750-го токарні верстати мали всі сучасні складові, такі як ходовий гвинт, колеса зміни частот обертання, хрещатий супорт, відомий також копіювальний токарний верстат 1741. Промислова революція 19 ст. викликала бурхливий розвиток металообробних верстатів, у тому числі й токарних верстатів. ерший токарний автомат з магазином, із розподільчим валом, з плоскими і циліндричними кулачками був створений Спенсером (англ. Spenser) у 1873. Перший револьверний прутковий автомат був створений 1880 року, який став прототипом для появи багатьох

					КВРКІ 200102.20.01.02.ПЗ	Арк.
						7
Зм.	Арк.	№ докум.	Підпис	Дата		

револьверних автоматів. У XIX столітті токарні верстати, завдяки винаходу Генрі Модслі перетворили на токарно-гвинторізні. Це значно розширило їх технологічні можливості [11].

За класифікацією група токарних верстатів поділяється на 9 підгруп (типів) [12]:

- автомати і напівавтомати одношпиндельні;
- автомати і напівавтомати багатшпиндельні;
- револьверні;
- центрувальні-відрізні;
- карусельні;
- токарні й лобові;
- багаторізцеві;
- спеціалізовані;
- різні.

Основні вузли токарного верстату [13]:

- станина є основою і призначена для монтажу всіх основних складальних одиниць верстата. По напрямних станини переміщуються каретка супорта і задня бабка;
- передня бабка встановлюється в лівій частині станини для закріплення заготовки і надання їй обертового руху. У верстатах ТВ-4 у передній бабці розміщена коробка зміни швидкостей;
- задня бабка підтримує вільний кінець довгих заготовок, а також кріпить різальні інструменти (свердла, зенкери, розвертки). Вона складається з трьох основних частин: корпусу, пінолі і плити. У конічний отвір пінолі можна встановити центр або інструмент. В разі потреби корпус задньої бабки зміщують у поперечному напрямі для обточування конусних отворів;
- супорт призначений для кріплення різців у різцетримачі і переміщення їх у подовжньому й поперечному напрямках відносно оброблюваної заготовки вручну або механічно;

					КВРКІ 200102.20.01.02.ПЗ	Арк.
						8
Зм.	Арк.	№ докум.	Підпис	Дата		

Токарні верстати за часом їхнього винайдення:

- токарно-лобові верстати;

Лобові токарні верстати застосовуються для виготовлення деталей великого діаметра і малої довжини. До недоліків лобових верстатів відносяться труднощі установки, вивірки і закріплення заготовки, а також виникнення вібрацій внаслідок великої ваги заготовки. У підсумку зазначених причин, лобові верстати поступаються за поширенням карусельним верстатам [15];



Рисунок 1.2 - Токарно-лобовий верстат [16]

- токарно-карусельні верстати;

Токарно-карусельні верстати призначені для виготовлення деталей великого діаметра і малої довжини. У токарно-карусельних верстатах вісь шпинделя розташовано вертикально, а торцева площина планшайби розташована в горизонтальній площині, внаслідок чого значною мірою полегшується установка, вивірення і закріплення заготовок. Вага заготовки і сили різання сприймаються круговими напрямними планшайби, тому шпиндель розвантажений від згинальних напружень і напружень крутіння, на відміну від лобових верстатів.

					КВРКІ 200102.20.01.02.ПЗ	Арк. 10
Зм.	Арк.	№ докум.	Підпис	Дата		

Унікальні карусельні верстати для виготовлення деталей турбін мають планшайбу діаметром до 18 метрів [17];



Рисунок 1.3 - Токарно-карусельний верстат [18]

- токарно-револьверний верстат;

Токарно-револьверний верстат — верстат токарної групи з револьверною головкою(замість задньої бабки), застосовується для багатоінструментальної обробки складних за конфігурацією поверхонь з пруткового матеріалу і штучних заготовок.

На револьверних верстатах виконують операції точіння, розточування, свердління, зенкерування, накатування різьб тощо.

Широке застосування дістали револьверні напівавтомати, головним чином обробні патрони з числовим керуванням [19];

					КВРКІ 200102.20.01.02.ПЗ	Арк. 11
Зм.	Арк.	№ докум.	Підпис	Дата		

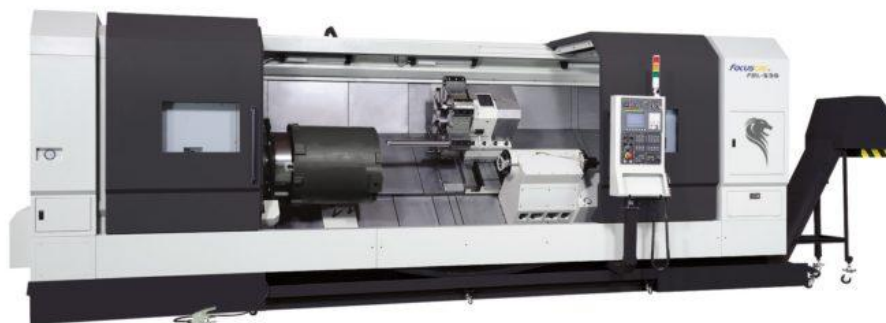


Рисунок 1.4 - Токарно-револьверний верстат [20]

- центрувально-відрізні верстати;

Верстати даного типу призначені для одночасної розрізки та центрування заготовки. До переваг верстатів даного типу можна віднести, те що під час обробки заготовки, відбувається поєднання двох операцій в одну, це дозволяє значно скоротити технологічний час [21];

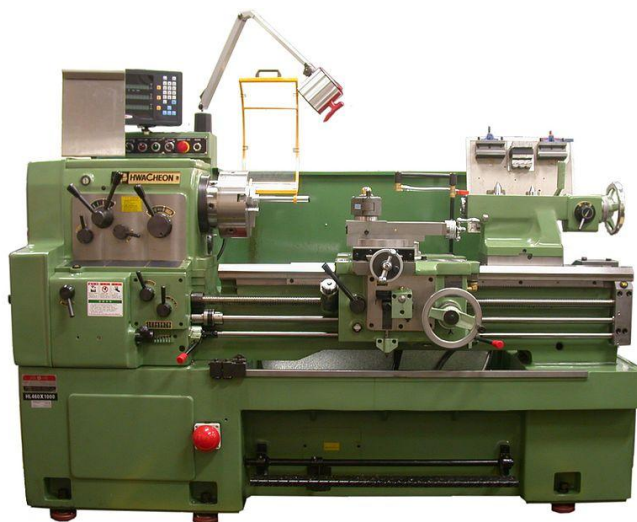


Рисунок 1.5 - Центрувально-відрізний верстат [22]

- багаторізцевий токарний верстат;

Багаторізцевий токарний верстат — верстат токарної групи, на котрому відбувається обробка заготовок одночасно декількома різцями, встановленими на

					КВРКІ 200102.20.01.02.ПЗ	Арк. 12
Зм.	Арк.	№ докум.	Підпис	Дата		

поздовжньому і поперечному супортах. Кожен із різців оброблює певну ділянку деталі, що значно скорочує час роботи супорта. Існують багаторізцеві токарні автомати і напівавтомати [23];



Рисунок 1.6 - Багаторізцевий токарний верстат [24]

- токарні автомати і напівавтомати.

Автоматами називаються верстати, у яких автоматизовані всі робочі і допоміжні рухи, потрібні для виконання технологічного циклу обробки деталі. До обов'язків робітника, що обслуговує верстат, входять періодичне завантаження заготовками, час від часу контроль розмірів і якості оброблених деталей, під налагодження верстата, а також загальне спостереження за його роботою. Токарні автомати поділяються на одношпиндельні і багатошпиндельні, застосовуються для виготовлення деталей із прутка, але в деяких випадках зі штучних заготовок [25]. Напівавтоматами називаються верстати, у яких процес обробки здійснюється без участі робітника. Установку і закріплення заготовки, а також зняття готової деталі робить робітник. Токарні напівавтомати поділяються на одношпиндельні і багатошпиндельні, на горизонтальні і вертикальні, застосовуються для обробки штучних заготовок [26];

					КВРКІ 200102.20.01.02.ПЗ	Арк.
						13
Зм.	Арк.	№ докум.	Підпис	Дата		



Рисунок 1.7 - Токарний автомат [27]



Рисунок 1.8 – Токарний напівавтомат [28]

1.3 Використання мікроконтролерів у токарних верстатах

Винахідником першого верстата з числовим керуванням є Джон Персонс , котрий працював інженером в компанії свого батька Parsons Inc, що випускала в кінці Другої світової війни пропелери для вертольотів. Він вперше запропонував

					КВРКІ 200102.20.01.02.ПЗ	Арк. 14
Зм.	Арк.	№ докум.	Підпис	Дата		

використовувати для обробки пропелерів верстат, який працює за програмою, що вводиться з перфокарт [29].

1949 року, ВПС США профінансували Parsons Inc розробку верстата для контурного фрезерування складних за формою деталей авіаційної техніки [30]. Але компанія не змогла самостійно працювати, тому звернулася до Массачусетського технологічного інституту. Але їх співпраця тривало не довго, МІТ придбав компанію з виробництва фрезерних верстатів Hydro-Tel і відмовився від співпраці з Parsons Inc [31]. Перший серійний пристрій числового програмного керування було створено компанією Bendix Corp. в 1954 році, і з 1955 року він став встановлюватися на верстатах. [32].

Завдяки постійному розвитку мікроконтролерів, сьогодні токарні верстата можуть працювати набагато точніше та продуктивніше. Вони здатні виконувати складні рухи та обробляти матеріали з високою швидкістю та ефективністю. Сучасні мікроконтролери дозволяють інтегрувати різноманітні функції у токарні верстата, такі як автоматична зміна інструментів, системи вимірювання, системи контролю якості. Це робить їх більш гнучкими та універсальними у використанні. Деякі сучасні токарні верстата використовують мікроконтролери з можливістю мережевого підключення, що дозволяє віддалено керувати процесом виробництва та здійснювати моніторинг через інтернет. Крім того, мікроконтролери забезпечують ефективну обробку даних, що дозволяє збільшити точність та якість обробки деталей.

Найпоширенішою мовою програмування числового програмного керування для металорізального устаткування є G-код. G-код - це стандартний мовний формат, що використовується в області обробки матеріалів, зокрема в верстатах з числовим управлінням, для керування рухом інструментів та обробки деталей. Він представляє собою послідовність команд, що вказують верстату, які рухи та операції потрібно виконати. Основні компоненти G-коду включають в себе команди для переміщення інструменту по робочій області, управління швидкістю руху, вибір інструментів, зміна параметрів обробки (таких як глибина

					КВРКІ 200102.20.01.02.ПЗ	Арк. 15
Зм.	Арк.	№ докум.	Підпис	Дата		

різу, швидкість подачі і інші), а також команди для включення і вимкнення різних функцій верстата. G-код може бути створений вручну або згенерований за допомогою спеціалізованих програм для моделювання та обробки деталей. Він є основним інструментом для програмування числового програмного керування верстатів і дозволяє точно керувати процесом виробництва, забезпечуючи високу точність і ефективність обробки [33].

1.4 Висновки та постановка задачі

Після аналізу інформаційної бази вдалося визначити основні характеристики мікроконтролерів та встановити їх рамки поняття. Для більш глибокого розуміння цієї теми було проведено огляд особливостей мікроконтролерів, їх складових елементів, а також розглянуто конкретні приклади та сфери постійного використання у виробництві або інших галузях.

Також, було опрацьовано поняття верстату токарної обробки. Для отримання більш глибокого розуміння з даною темою, було розглянуто типи верстатів за принципом їх роботи. Після цього визначили їхні переваги та умови для їх використання. Для глибшого розуміння цих систем як цілісних структур, були визначені основні компоненти та функціонал кожної з них.

Відповідно до опрацьованої інформації, була сформована постановка необхідних етапів алгоритму для виконання поставленої задачі.

Основним завданням наступного розділу є розроблення основного принципу роботи необхідного алгоритму. Додатково, необхідно дослідити технічні особливості мікроконтролера ESP32-CAM, які необхідні для виконання поставленого завдання. Це включає вибір необхідних компонентів у разі наявності можливих альтернатив.

На завершення, потрібно дослідити системи, які використовують мікроконтролер ESP32-CAM або можуть з ним взаємодіяти, щоб визначити відмінності цієї системи від існуючих аналогів, знайти їх переваги та недоліки, а

					КВРКІ 200102.20.01.02.ПЗ	Арк. 16
Зм.	Арк.	№ докум.	Підпис	Дата		

також з'ясувати можливі способи реалізації певних функцій з кращою ефективністю або оптимізацією.

					КВРКІ 200102.20.01.02.ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		17

2 ВИБІР НАЯВНОГО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ СИСТЕМИ АВТОМАТИЧНОГО ВИЗНАЧЕННЯ РОЗМІРІВ В ПРОЦЕСІ ТОКАРНОЇ ОБРОБКИ ДЕТАЛЕЙ

2.1 Технічні характеристики мікроконтролеру

Модуль ESP32-CAM — це компактний і потужний мікроконтролерний модуль, який включає камеру, Wi-Fi та Bluetooth, що робить його ідеальним для проєктів з відеоспостереження, інтернету речей (IoT), а також інших застосувань, де потрібні можливості захоплення зображення та передачі даних [34]. Нижче можемо побачити його зовнішній вигляд на рисунку 2.1.



Рисунок 2.1 – Зовнішній вигляд мікроконтролеру ESP32-CAM з під'єднаною камерою OV2640 [35]

Основні характеристики модулю [36]:

- двоядерний 32-бітний процесор з низьким енергоспоживанням;
- основна частота до 240 МГц, обчислювальна потужність до 600 DMIPS;

					КВРКІ 200102.20.01.02.ПЗ	Арк. 18
Зм.	Арк.	№ докум.	Підпис	Дата		

- обудований SRAM 520 КБ, зовнішній PSRAM 4М;
- підтримка інтерфейсів, таких як UART/SPI/I2C/PWM/ADC/DAC;
- підтримка камер OV2640 і OV7670, вбудована спалах;
- підтримка зображень Wi-Fi завантаження;
- підтримка TF карти;
- підтримує кілька режимів сну;
- вбудований Lwip і FreeRTOS;
- підтримка STA/AP/STA + AP Робочий режим;
- підтримка Smart Config/AirKiss розподільна мережа в один клік;
- габарити 27 × 41 × 5 mm ;
- напруга живлення: 5;
- процесор: 32-розрядний Tensilica Xtensa LX6;
- тактова частота процесора: 2 ядра × 160 МГц;
- пам'ять: 520 КБ SRAM, 4МБ PSRAM;
- бездротовий зв'язок: Bluetooth 4.2 BR/EDR, Wi-Fi 802.11 b/g/n;
- інтерфейси: UART, I2C, SPI, PWM.

Згідно цих технічних характеристик мікроконтролеру, варто зазначити його сильні сторони у порівнянні з іншими схожими мікроконтролерами. Сильні сторони мікроконтролера [37]:

- висока обчислювальна потужність: модуль оснащений потужним мікроконтролером ESP32, який має два ядра Tensilica LX6, що працюють на частоті до 240 МГц. Це забезпечує високу продуктивність для обробки зображень та виконання інших складних завдань;

- інтегрована камера: вбудована камера OV2640 підтримує роздільну здатність до 1600x1200 (UXGA) та може знімати відео до VGA (640x480) при 30 кадрах в секунду. Це дозволяє модулю захоплювати високоякісні зображення та відео;

- Wi-Fi та bluetooth: вбудовані модулі Wi-Fi (802.11 b/g/n) і Bluetooth (v4.2 BR/EDR і BLE) забезпечують гнучкість у підключенні до мережі та інших

					КВРКІ 200102.20.01.02.ПЗ	Арк. 19
Зм.	Арк.	№ докум.	Підпис	Дата		

пристроїв. Це робить ESP32-CAM ідеальним для IoT-проектів та бездротових систем;

- компактний розмір: модуль має невеликі габарити (27x40.5 мм), що дозволяє легко інтегрувати його в обмежений простір і робить його придатним для портативних та вбудованих систем;

- низька вартість: модуль ESP32-CAM є доступним за ціною, що робить його привабливим для хобістів, ентузіастів та розробників з обмеженим бюджетом;

- енергозбереження: ESP32-CAM підтримує різні режими енергозбереження, що дозволяє зменшити споживання енергії і використовувати модуль в автономних системах на батарейках.

Але цей модуль не є ідеальним, тому також варто розглянути його недоліки та це може спричинити зміни вибору мікроконтролера. Але у нашому проекті це неможливо, але на недоліки також варто звернути увагу.

- обмежена пам'ять: обсяг оперативної пам'яті на ESP32-CAM може бути недостатнім для деяких складних задач з обробки зображень або роботи з великими даними. Це обмежує використання складних алгоритмів машинного навчання без додаткових оптимізацій;

- складність у відладці: модуль не має вбудованого інтерфейсу для зручного відладження, що ускладнює процес розробки та виправлення помилок;

- високе енергоспоживання: незважаючи на наявність режимів енергозбереження, ESP32-CAM може споживати значну кількість енергії під час активної роботи, особливо при використанні Wi-Fi або камери. Це може бути проблемою для автономних систем, які працюють від батарейок;

- тепловиділення: при тривалій інтенсивній роботі модуль може нагріватися, що може вимагати додаткових заходів для охолодження або теплоотведення;

- обмежена документація та підтримка: Для деяких функцій модуля може бути недостатньо документації або прикладів використання, що може ускладнити процес розробки для новачків або в специфічних застосуваннях.

2.2 Вибір камери для виконання роботи

Як було зазначено у попередньому підрозділі, мікроконтролер ESP32-CAM підтримує дві моделі камер – OV2640 та OV7670. Тому необхідно порівняти ці дві моделі та визначити, яка з них найбільше підходить для реалізації необхідного функціоналу для виконання поставленого завдання. На рисунку 2.2,2.3 представлено зовнішній вигляд цих камер для візуального ознайомлення та оцінки їх розмірів.

Почнемо з технічних характеристик камери OV7670 [38] :

- однокристальний CMOS датчик зображення;
- контролюється за допомогою шини Serial Camera Control Bus ;
- максимальна роздільна здатність 640x480 пікселів ;
- доступні дозволи: QVGA (320 x 240); CIF (352 x 240); QCIF (176 × 144) ручне масштабування;
- варіанти кодування: RGB565, RGB555, RGB444, YUV / YCbCr 4: 2: 2, GRB 4: 2: 2, Raw RGB Data;
- напруга на CMOS сенсора: 1,8 В;
- напруга на аналогових елементах: 2,45 - 3 В;
- напруга живлення модуля: 3 - 5 В;
- споживання струму: 20 мА;
- розмір лінзи: 1/6 дюйма;
- кут огляду: 24 градуси;
- швидкість видаваного зображення: до 30 кадрів в секунду;
- чутливість: 1,1 В / Люкс-сек;
- відношення сигнал / шум: 46Дб;
- розміри: 35 x 35 x 31 мм;
- вага: 12 г.

					КВРКІ 200102.20.01.02.ПЗ	Арк. 21
Зм.	Арк.	№ докум.	Підпис	Дата		



Рисунок 2.3 – Зовнішній вигляд камери OV2640 [41]

Відповідно до отриманої інформації, варто зазначити, що кожна з цих камер має свої переваги та недоліки, які варто враховувати для кращого розуміння, яка саме камера найкраще підходить для виконання необхідного завдання.

Почнемо з переваг моделі OV7670: ця камера зазвичай є більш доступним варіантом та широко використовується у проектах з обмеженим бюджетом завдяки її низькій ціні. Вона легко доступна в багатьох електронних магазинах. Також варто зазначити, що фізичні розміри цієї камери досить малі, що полегшує її інтеграцію у різні пристрої та системи.

Однак, у моделі OV7670 є й недоліки: максимальна роздільна здатність досить низька, що може бути недостатнім для певних додатків або вимог до якості зображення. Спектр функцій, доступних у цій моделі, досить обмежений, що значно звужує її можливості у використанні. Крім того, можливості налаштування

камери та параметрів зображення теж обмежені, що може не відповідати вимогам конкретного проекту.

На відміну від OV7670, камера OV2640 має ряд переваг: максимальна роздільна здатність в два мегапікселі дозволяє отримувати значно вищу якість зображення та більше деталей. Набір функцій у цій камері значно ширший, включаючи автоматичний баланс білого, автоекспозицію та підтримку різних режимів фокусування, що робить її більш гнучкою для різних застосувань.

Однак перед вибором цієї моделі слід враховувати, що вона є дорожчою, ніж OV7670, що може вплинути на бюджет проекту. Також розміри камери OV2640 значно більші, що може ускладнити її інтеграцію у пристрої з обмеженим простором.

Зважаючи на вищезазначені характеристики, для кращої реалізації поставленого завдання рекомендується використовувати камеру OV2640. Незважаючи на те, що це призведе до збільшення витрат та може ускладнити інтеграцію, OV2640 має значно кращий функціонал та можливості, що дозволить не лише покращити якість відеоспостереження, а й розширити функціональність алгоритму новими можливостями.

2.3 Загальні відомості про мікроконтролер ESP32-CAM

Мікроконтролер ESP32-CAM має кілька переваг порівняно з іншими мікроконтролерами зі схожим функціоналом. Завдяки вбудованим модулям Wi-Fi та Bluetooth, мікроконтролер може бездротово підключатися до інтернету та інших пристроїв. Незалежно від обраної камери, її інтеграція в ESP32-CAM спрощує процес з'єднання та налаштування. Двоядерний процесор з високою тактовою частотою забезпечує достатню обчислювальну потужність для виконання складних функцій. Підтримка широкого спектру інтерфейсів, таких як GPIO, UART, SPI, I2C, ADC та інших, дозволяє підключати різні сенсори та пристрої, розширюючи функціонал системи відповідно до потреб користувача.

					КВРКІ 200102.20.01.02.ПЗ	Арк. 24
Зм.	Арк.	№ докум.	Підпис	Дата		

Однак, окрім переваг, ESP32-CAM має й недоліки, які варто враховувати. Обмежена кількість флеш-пам'яті та оперативної пам'яті може бути проблемою при обробці великої кількості інформації. Рівень споживання енергії мікроконтролером ESP32-CAM є досить високий порівняно з іншими моделями, що слід враховувати, особливо якщо передбачається живлення від батареї. Вартість цієї моделі також може перевищувати ціну деяких інших мікроконтролерів, що потрібно мати на увазі при оцінюванні бюджету майбутнього проекту.

2.4 Сфери використання мікроконтролеру ESP32-CAM

Мікроконтролер ESP32-CAM знаходить своє застосування в багатьох сферах завдяки своїм потужним функціональним можливостям. Ось декілька основних напрямків його використання [42]:

- системи відеоспостереження: завдяки вбудованій камері та можливостям бездротового зв'язку, ESP32-CAM широко використовується для створення автономних систем відеоспостереження. Він дозволяє передавати відео в режимі реального часу через Wi-Fi, що робить його ідеальним для охорони будинків, офісів та інших об'єктів;

- розумний дім: ESP32-CAM можна інтегрувати в систему розумного дому для моніторингу приміщень, автоматизації освітлення, контролю температури та інших функцій. Можливість підключення до мережі дозволяє дистанційно керувати різними пристроями через інтернет;

- інтернет речей : завдяки компактному розміру, низькому енергоспоживанню та широкому набору інтерфейсів, ESP32-CAM є відмінним вибором для IoT-проектів. Він може збирати та передавати дані від сенсорів, забезпечувати зв'язок між різними пристроями та виконувати локальну обробку даних;

					КВРКІ 200102.20.01.02.ПЗ	Арк. 25
Зм.	Арк.	№ докум.	Підпис	Дата		

- промислова автоматизація: у промислових умовах ESP32-CAM може використовуватись для моніторингу обладнання, контролю процесів та збору даних. Вбудовані інтерфейси дозволяють підключати різні датчики та актуатори для забезпечення повної автоматизації виробничих процесів;

- медичні пристрої: ESP32-CAM може бути використаний у медичних приладах для моніторингу пацієнтів, збору медичних даних та дистанційного контролю за станом здоров'я. Можливість передачі даних через інтернет робить його корисним для телемедицини;

- розробка роботів: у сфері робототехніки ESP32-CAM можна використовувати для створення систем візуального сприйняття, що дозволяє роботам орієнтуватися в просторі, виявляти об'єкти та взаємодіяти з навколишнім середовищем;

- освітні проекти: завдяки своїй доступності та широкому функціоналу.

2.5 Порівняння мікроконтролера ESP32-CAM з іншими за використанням у сфері автоматичної обробки деталей

Мікроконтролер ESP32-CAM відзначається низкою особливостей, які роблять його цікавим для використання в автоматичній обробці деталей. Проте, порівнюючи його з іншими мікроконтролерами, важливо звернути увагу на специфічні вимоги цієї сфери, а також на переваги та недоліки кожного варіанту.

1) Arduino Mega 2560

Arduino Mega 2560 - це мікроконтролерна плата, що базується на мікроконтролері ATmega2560. Вона має 54 цифрові GPIO порти, 16 аналогових входів, 4 UART порти, 16 МГц кварцовий резонатор, USB з'єднання та інші роз'єми та інтерфейси для розширення можливостей плати [43]. Даний мікроконтролер зображено на рисунку 2.4.

					КВРКІ 200102.20.01.02.ПЗ	Арк. 26
Зм.	Арк.	№ докум.	Підпис	Дата		



Рисунок 2.4 – Зовнішній вигляд Arduino Mega 2560 [44]

У кожної плати є свої переваги та недоліки і ця плата не виключення.

Переваги:

- велика кількість GPIO портів для підключення багатьох пристроїв;
- великий обсяг флеш-пам'яті для зберігання програмного коду та даних;
- зручний інтерфейс користувача та велика спільнота користувачів Arduino;
- підтримка різних інтерфейсів для підключення зовнішніх пристроїв.

Недоліки:

- обмежена швидкість та продуктивність порівняно з більш потужними мікроконтролерами;
- великі розміри та вага, що можуть ускладнювати інтеграцію в обмеженому просторі;
- високе енергоспоживання, що може бути проблемою для пристроїв з батарейним живленням.

2) Raspberry Pi Pico

Raspberry Pi Pico - це мікроконтролерна плата, яка використовує мікроконтролер RP2040. Вона має 26 GPIO портів, 3 аналогових входи, 2 UART

порти, 2 SPI і 2 I2C інтерфейси, 16 МБ вбудованої флеш пам'яті, USB з'єднання та інші роз'єми для підключення різноманітних пристроїв [45] .Для ознайомлення з ним роглянемо рисунок 2.5.



Рисунок 2.5 – Зовнішній вигляд Raspberry Pi Pico [46]

Переваги:

- компактний розмір та низька вартість, що робить його доступним для багатьох проектів;
- висока продуктивність та ефективність завдяки потужному мікроконтролеру RP2040;
- зручний інтерфейс для роботи з Raspberry Pi та велика спільнота користувачів;
- широкі можливості розширення завдяки великій кількості GPIO портів та підтримки різних інтерфейсів.

Недоліки:

- обмежені можливості порівняно з більш потужними одноплатними комп'ютерами Raspberry Pi;

- відсутність вбудованих засобів бездротового зв'язку (Wi-Fi, Bluetooth), що може обмежувати деякі застосування;
- обмежена кількість аналогових входів порівняно з іншими мікроконтролерами.

3) Raspberry Pi Zero W

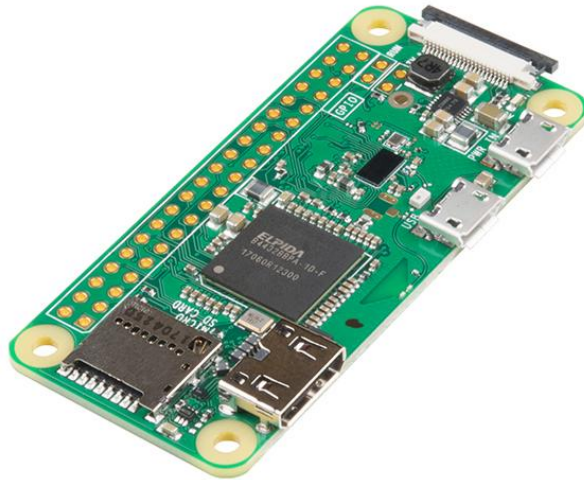


Рисунок 2.6 – Зовнішній вигляд Raspberry Pi Zero W [47]

Raspberry Pi Zero W - це одноплатний комп'ютер, який базується на процесорі Broadcom BCM2835 з тактовою частотою 1 ГГц. Він має 512 МБ оперативної

пам'яті, підтримку Wi-Fi та Bluetooth, один HDMI порт, один micro-USB порт, а також GPIO роз'єми для підключення різних пристроїв [48]. Для того щоб побачити його зовнішній вигляд розглянемо рисунок 2.6.

Переваги:

- компактний розмір та низька вартість, що робить його ідеальним для вбудованих та розширених проєктів;
- вбудована підтримка Wi-Fi та Bluetooth, що дозволяє бездротове підключення до мережі та інших пристроїв;

- наявність GPIO портів для підключення різних сенсорів, актуаторів та інших пристроїв;
- підтримка різних операційних систем, таких як Raspbian, Ubuntu, а також спеціалізовані ОС для IoT проектів.

Недоліки:

- обмежена обчислювальна потужність порівняно з більш потужними моделями Raspberry Pi, що може обмежити деякі застосування;
- обмежена кількість оперативної пам'яті, що може вплинути на продуктивність деяких додатків;
- відсутність додаткових портів, таких як Ethernet, що може вимагати використання додаткових переходників для підключення до мережі.

4) STM32F4 Discovery Kit

STM32F4 Discovery Kit - це платформа для розробки, що базується на мікроконтролері серії STM32F4 від STMicroelectronics. Вона має широкі можливості для розробки різноманітних програмних та апаратних проектів [49]. Даний мікроконтролер зображено на рисунку 2.7.

Переваги:

- висока продуктивність та потужність обробки завдяки використанню мікроконтролера STM32F4 з ядром ARM Cortex-M4 з FPU;
- наявність багатофункціональних портів, таких як GPIO, UART, SPI, I2C, ADC тощо, що дозволяє підключати різноманітні пристрої та сенсори;
- широкі можливості для розробки програмного забезпечення завдяки наявності різноманітних інтерфейсів та розширювальних слотів;
- вбудовані інструменти для відлагодження та програмування, такі як відлагоджувальник ST-Link та інтегровані датчики.

Недоліки:

- вища складність програмування порівняно з більш простими платформами, що може вимагати більшої експертизи у використанні;

- вища вартість порівняно з іншими одноплатними комп'ютерами, що може обмежувати доступність для окремих проектів;
- обмежена підтримка спільноти порівняно з іншими популярними платформами, що може вплинути на доступність додаткових ресурсів та бібліотек.

					КВРКІ 200102.20.01.02.ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		31



Рисунок 2.7 – Зовнішній вигляд STM32F4 Discovery Kit [50]

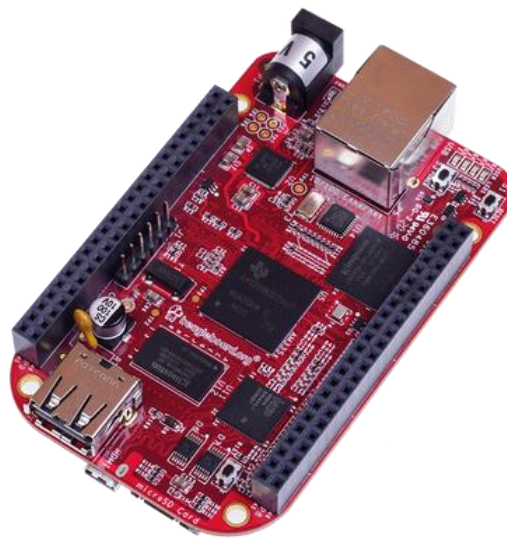


Рисунок 2.8 – Зовнішній вигляд BeagleBone Black [51]

Зм.	Арк.	№ докум.	Підпис	Дата

1) BeagleBone Black

BeagleBone Black - це одноплатний комп'ютер на базі процесора ARM Cortex-A8 від Texas Instruments. Він має ряд унікальних можливостей, які роблять його популярним для широкого спектру проектів [52]. Для ознайомлення розглянемо рисунок 2.8.

Переваги:

- потужний процесор Cortex-A8, що забезпечує високу продуктивність та швидкодію;
- широкий набір роз'ємів та інтерфейсів, включаючи GPIO, UART, SPI, I2C, HDMI, USB та інші, для підключення різноманітних пристроїв та датчиків;
- вбудований Ethernet та підтримка Wi-Fi через додаткові модулі, що робить його ідеальним для мережевих застосувань;
- підтримка великої кількості операційних систем, включаючи Linux та Android, що робить його універсальним для різних типів програмних проектів;
- активна спільнота користувачів та розробників, що надає доступ до багатьох корисних ресурсів та бібліотек.

Недоліки:

- вища вартість порівняно з деякими іншими одноплатними комп'ютерами, що може обмежувати його доступність для деяких проектів;
- обмежена кількість вбудованої пам'яті та об'ємів зберігання, що може бути не достатньо для деяких вимог;
- складність використання для початківців без досвіду в програмуванні та роботі з апаратурою.

					КВРКІ 200102.20.01.02.ПЗ	Арк. 33
Зм.	Арк.	№ докум.	Підпис	Дата		

2.6 Використання мікроконтролера ESP32-CAM в системах автоматичної обробки деталей. Переваги та недоліки у порівнянні з іншими існуючими системами.

Мікроконтролер ESP32-CAM часто використовується в системах автоматичної обробки деталей, особливо в проектах, які потребують невеликого розміру, низької вартості та низького енергоспоживання.

Основна причина цього полягає у вдалій комбінації мікроконтролера ESP32 з модулем камери OV2640. Хоча дана модель також підтримує камеру OV7670, модель OV2640 є більш розповсюдженою.

Технічні особливості мікроконтролера ESP32-CAM та камери, яку він використовує, забезпечують йому широке застосування.

Проте, для кращого розуміння його функціоналу як частини системи автоматичної обробки деталей, необхідно враховувати як сильні, так і слабкі сторони. Крім того, слід звернути увагу на особливості, що виникають при його використанні у порівнянні з іншими існуючими системами.

Серед переваг ESP32-CAM у сфері автоматичної обробки деталей слід зазначити наступні особливості: мікроконтролер пропонує потужний набір функцій, таких як бездротове підключення Wi-Fi, можливість запису відео та фотографування, підтримка мережевих протоколів і додаткових модулів зв'язку, що забезпечує широкий функціонал.

Наявність вже інтегрованої камери може значно зменшити фінансові витрати при проектуванні системи, хоча ці витрати навіть так є досить малими у порівнянні з іншими системами автоматичної обробки деталей.

Мікроконтролер має вбудовану здатність до обробки зображень, включаючи можливість розпізнавання об'єктів, виявлення дефектів та інші функції. Розробка систем з використанням ESP32-CAM є досить простою завдяки широкій підтримці програмних бібліотек та середовищ розробки.

					КВРКІ 200102.20.01.02.ПЗ	Арк. 34
Зм.	Арк.	№ докум.	Підпис	Дата		

Основні недоліки ESP32-CAM в системах автоматичної обробки деталей залишаються майже незмінними порівняно з іншими мікроконтролерами. Роздільна здатність камери OV2640 залишається досить низькою для деяких завдань, що ускладнює використання цього мікроконтролера для проектів, які потребують високої якості зображення.

Відсутність вбудованого сховища даних є ключовим недоліком, що змушує користувачів або розробників враховувати необхідність додаткової покупки і налаштування зовнішнього сховища. Це може призвести до труднощів у розробці системи та обмежити сферу його застосування.

Крім того, потреба у стабільному джерелі живлення для безперервної роботи не дозволяє використовувати даний мікроконтролер у автономних системах автоматичної обробки деталей, а також обмежує його здатність до роботи у разі аварійного вимкнення енергії.

2.7 Існуючі аналоги систем з використанням ESP32-CAM та інших мікроконтролерів

Існує багато аналогів систем з використанням ESP32-CAM та інших мікроконтролерів в процесі токарної обробки. Ось декілька прикладів:

- система на основі ESP32-CAM та Arduino Uno:

Ця система використовує ESP32-CAM для керування токарним верстатом з ЧПК, а Arduino Uno використовується як додатковий контролер для підключення сенсорів та виконавчих механізмів [53].

Приклад взаємодії цих двох плат на рисунку 2.9.

Переваги: доступна ціна, простота налаштування, широкий спектр доступних компонентів.

Недоліки: обмежена обчислювальна потужність, складність розробки складних алгоритмів.

					КВРКІ 200102.20.01.02.ПЗ	Арк. 35
Зм.	Арк.	№ докум.	Підпис	Дата		

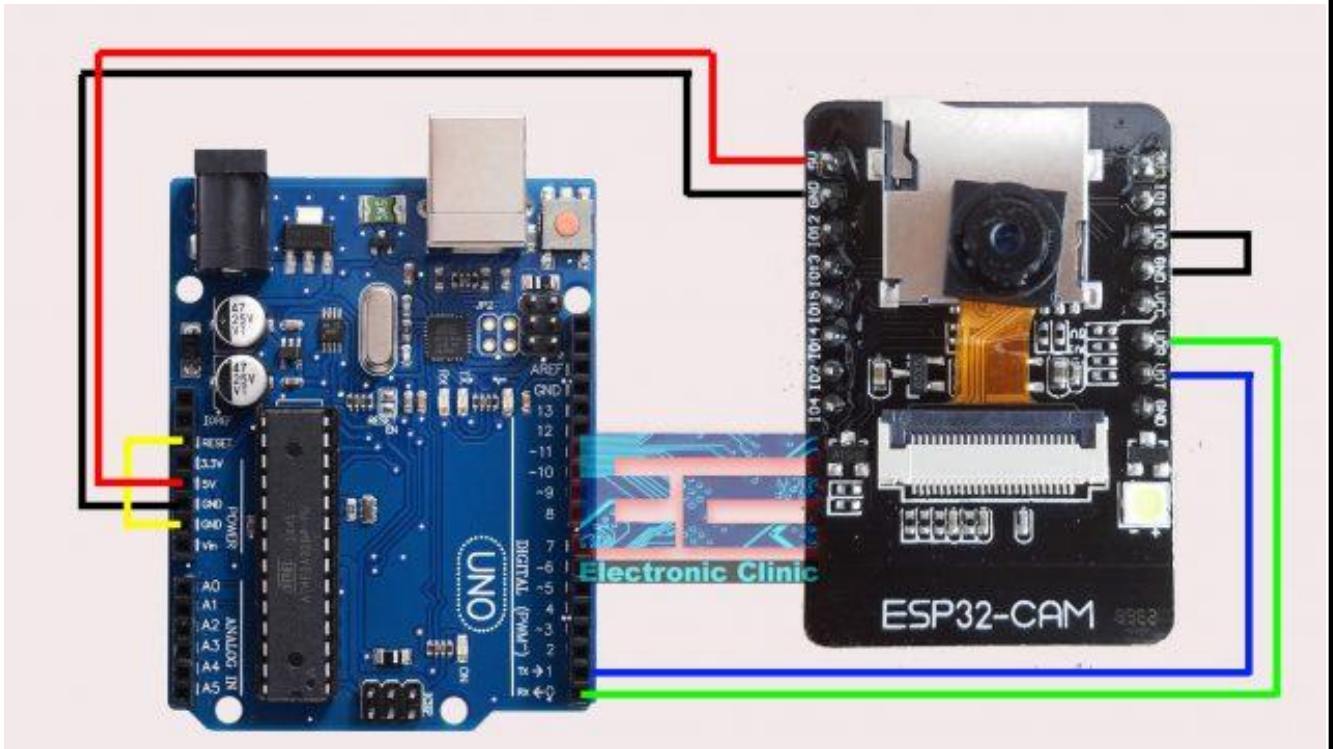


Рисунок 2.9 – Приклад взаємодії ESP32-CAM та Arduino Uno [54]

- система на основі ESP32-CAM та Raspberry Pi:

Ця система використовує ESP32-CAM для збору зображень з токарного верстата, а Raspberry Pi використовується для обробки зображень та керування токарним верстатом [55].

Переваги: висока обчислювальна потужність, можливість розробки складних алгоритмів, можливість підключення до хмари.

Недоліки: вища ціна, складність налаштування, потреба в додатковому обладнанні.

- системи з використанням інших мікроконтролерів:

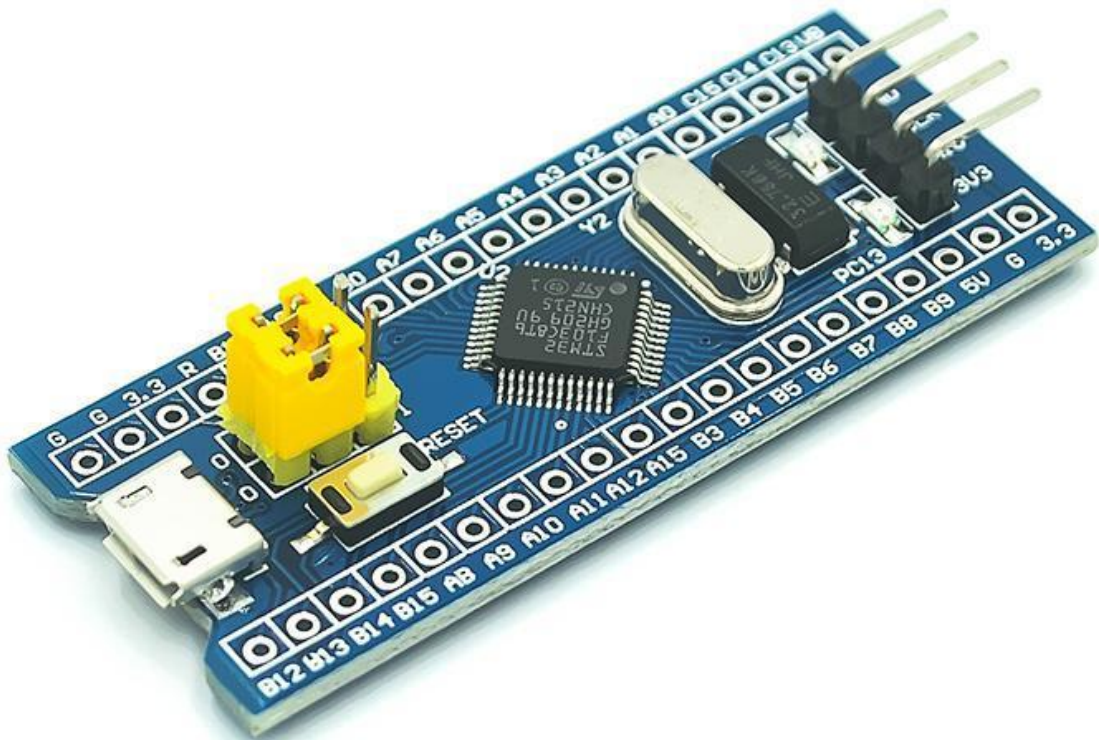


Рисунок 2.10 – Мікроконтролер STM32 [56]

Крім ESP32-CAM, для керування токарними верстатами також можна використовувати інші мікроконтролери, такі як STM32, Atmel AVR, PIC тощо.

Вибір мікроконтролера залежить від конкретних потреб проекту.

При виборі системи з використанням ESP32-CAM або іншого мікроконтролера для токарної обробки важливо врахувати такі фактори: бюджет проекту, рівень складності проекту, необхідні функції, доступні навички та знання.

2.8 Висновок

У цьому розділі ми дослідили технічні характеристики мікроконтролера ESP32-CAM, а також розглянули його можливості для використання в системі автоматичного визначення розмірів деталей в процесі токарної обробки. Технічні характеристики мікроконтролера ESP32-CAM має потужний двоядерний процесор Tensilica Xtensa LX32, 520 КБ SRAM, 8 МБ SPI флеш-пам'яті,

вбудований Wi-Fi та Bluetooth, а також 12-бітний АЦП. Вибір камери для виконання роботи, для даної задачі може використовуватися камера OV2640, яка має роздільну здатність 2 Мп і частоту кадрів 30 кадрів в секунду. Загальні відомості про мікроконтролер ESP32-CAM є компактним та потужним мікроконтролером з вбудованою камерою, що робить його ідеальним для використання в системах машинного зору. Сфери використання мікроконтролера ESP32-CAM може використовуватися в широкому спектрі проектів, включаючи системи відеоспостереження, системи розпізнавання об'єктів та системи автоматичного керування. Порівняння мікроконтролера ESP32-CAM з іншими, цей мікроконтролер має ряд переваг перед іншими мікроконтролерами, таких як потужний процесор, вбудована камера, низьке споживання енергії та доступна ціна. Переваги та недоліки використання ESP32-CAM в системах автоматичної обробки деталей, ESP32-CAM пропонує ряд переваг для систем автоматичної обробки деталей, таких як потужний мікроконтролер, вбудована камера, компактний розмір, низьке споживання енергії та доступна ціна. Однак, перед використанням ESP32-CAM важливо враховувати його обмеження, такі як роздільна здатність камери, відсутність вбудованого сховища, високе споживання енергії при роботі Wi-Fi та обмежена обчислювальна потужність. Існуючі аналоги систем з використанням ESP32-CAM та інших мікроконтролерів, існує багато аналогів систем з використанням ESP32-CAM та інших мікроконтролерів в процесі токарної обробки, які можна використовувати як зразки для розробки власної системи.

В наступному розділі ми опишемо алгоритм роботи системи автоматичного визначення розмірів деталей в процесі токарної обробки, а також наведемо приклад його реалізації на основі мікроконтролера ESP32-CAM.

					КВРКІ 200102.20.01.02.ПЗ	Арк. 38
Зм.	Арк.	№ докум.	Підпис	Дата		

3 ПРОГРАМНО-АПАРАТНА РЕАЛІЗАЦІЯ ПРОГРАМНО-ТЕХНІЧНОГО ЗАСОБУ ДЛЯ СИСТЕМИ АВТОМАТИЧНОГО ВИЗНАЧЕННЯ РОЗМІРІВ В ТОКАРНІЙ ОБРОБЦІ З МІКРОКОНТРОЛЕРОМ ESP32-CAM

3.1 Встановлення необхідних ресурсів та ArduinoIDE

Щоб створити систему автоматичного визначення розмірів в процесі токарної обробки деталей з використанням мікроконтролера ESP32-CAM, необхідно встановити та налаштувати Arduino IDE, а також додаткові бібліотеки та ресурси. Для початку щоб розпочати нашу роботу слід встановити середовище для розробки Arduino IDE. Його ми можемо встановити з офіційного сайту [56]. Це ми можемо побачити на рисунку 3.1.

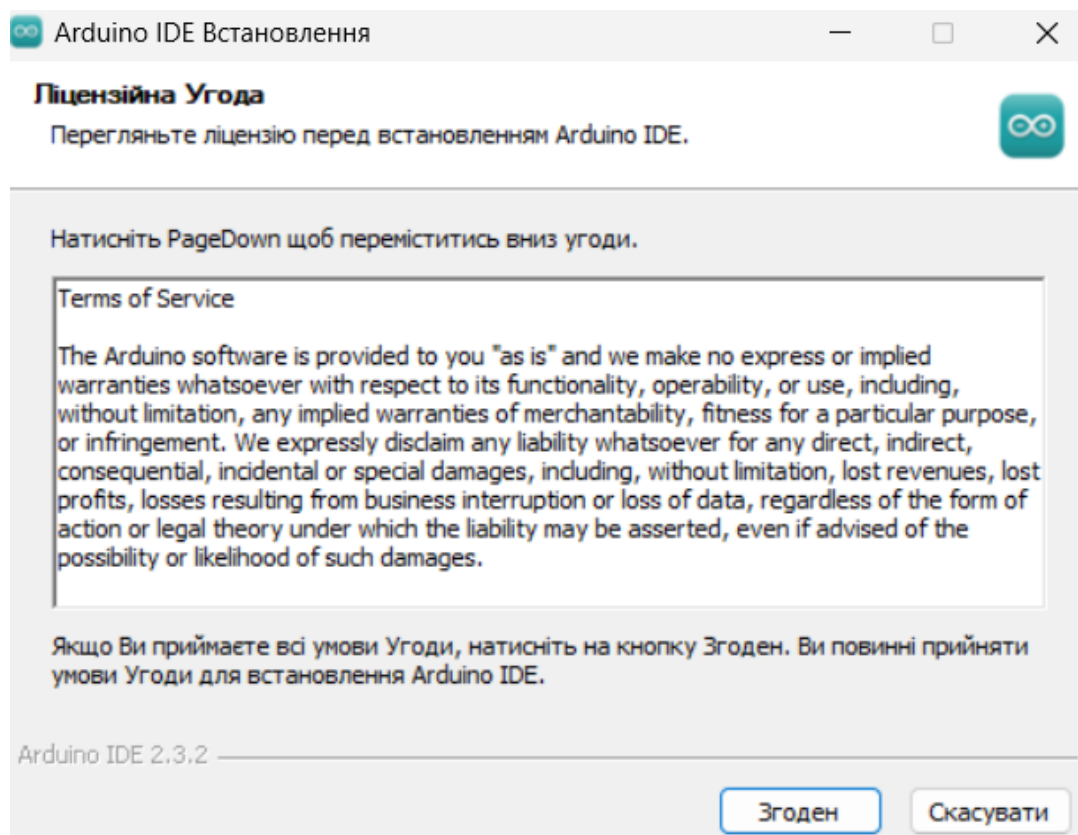


Рисунок 3.1- Встановлення Arduino IDE

Після того як ми встановили Arduino IDE нам потрібно налаштувати середовище розробки для програмування модуля ESP32-CAM. Для початку слід зайти у вікно «Параметри» та найти вкладку «URL менеджерів додаткових плат» далі потрібно в цьому полі ввести таку URL-адресу: https://dl.espressif.com/dl/package_esp32_index.json та натиснути клавішу Гаразд. Ці дії ми можемо побачити на рисунку 3.2.

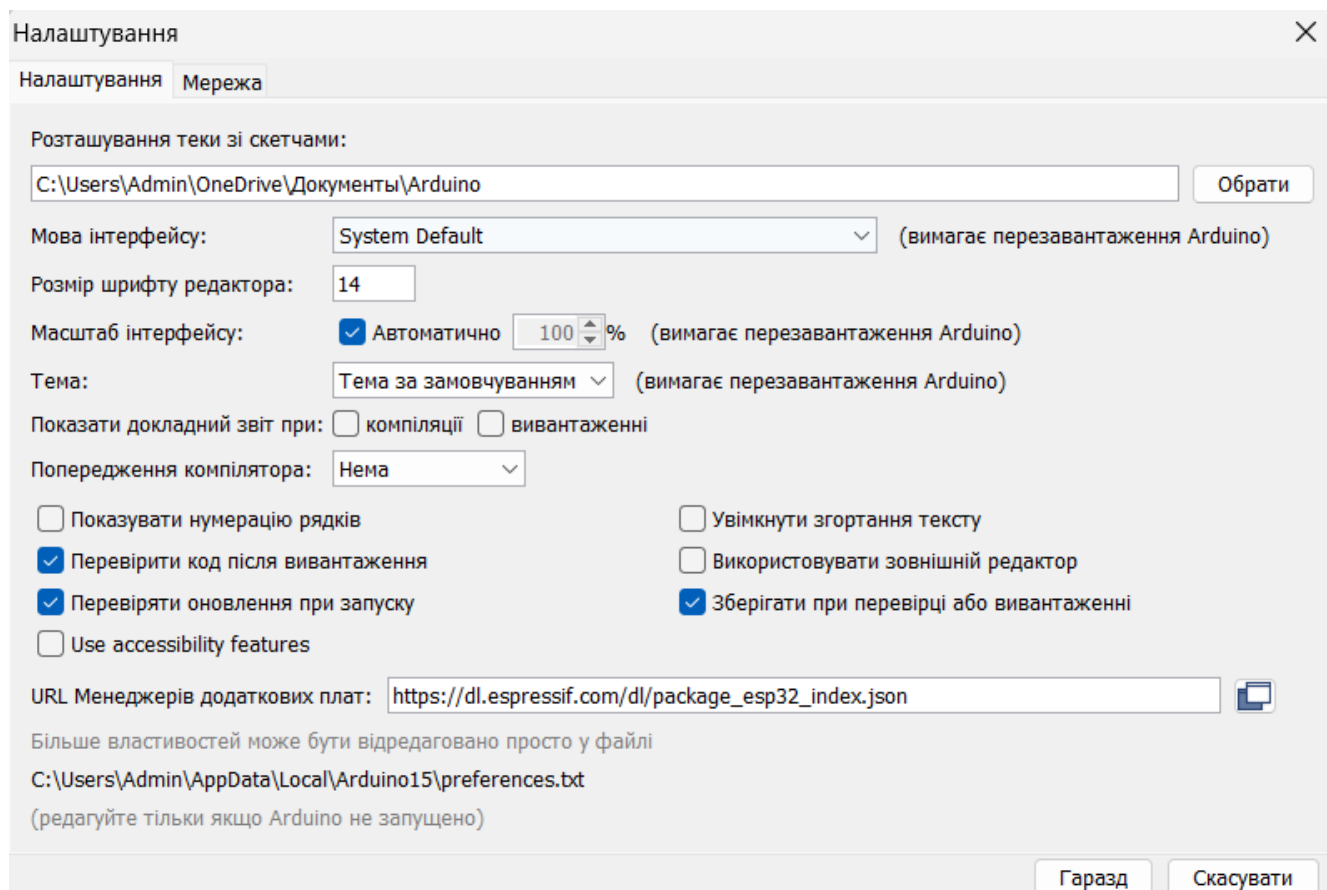


Рисунок 3.2 – Налаштування середовища

Далі нам потрібно зайти «Інструменти» > «Плати» > «Менеджер плат» та встановити бібліотеку ES32 для того щоб ми могли працювати далі з нашою платою. Це ми можемо спостерігати на рисунку 3.3. Там показано як ми встановлювали дану бібліотеку.

					КВРКІ 200102.20.01.02.ПЗ	Арк. 40
Зм.	Арк.	№ докум.	Підпис	Дата		

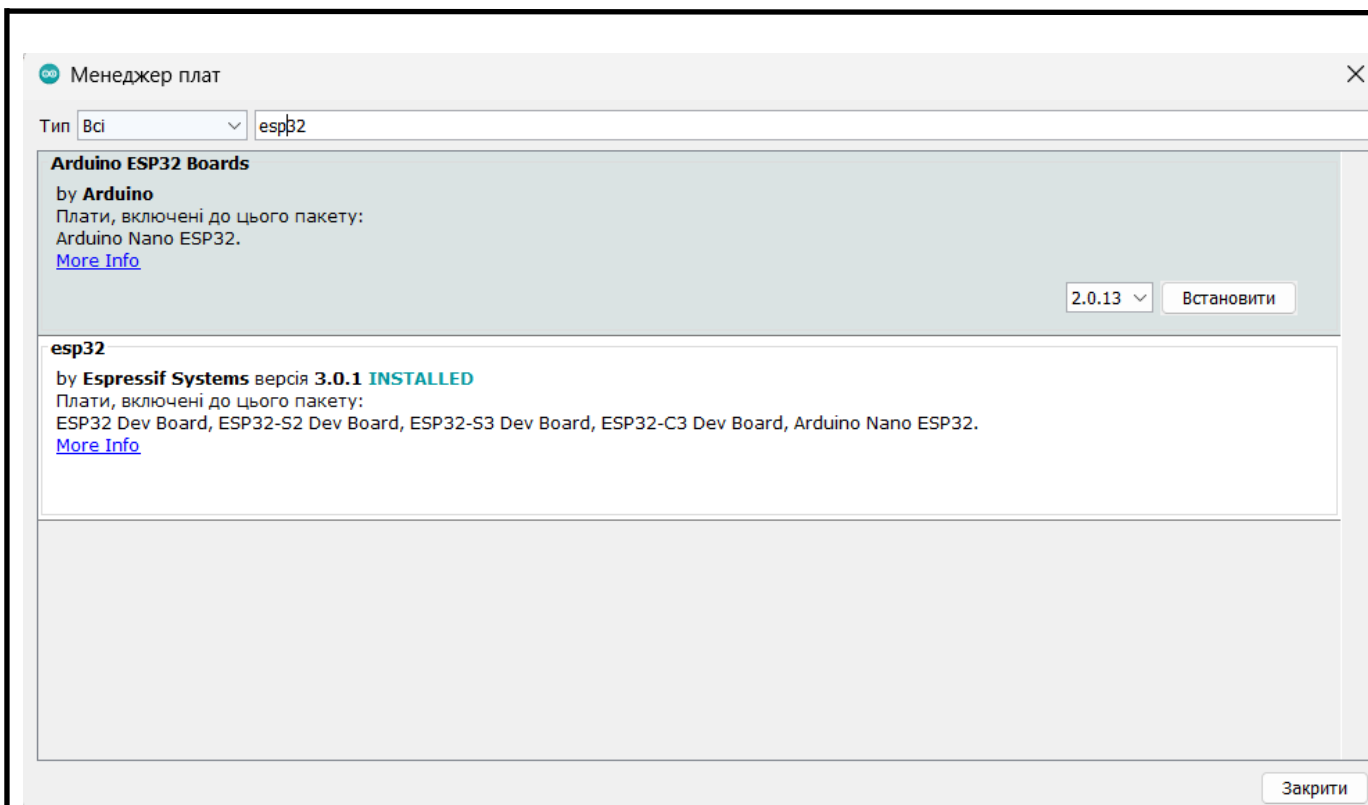


Рисунок 3.3 – Встановлення бібліотеки ESP32

.Далі нам потрібно підключити нашу плату до комп'ютера за USB-кабелю через UART-адаптер.Підключену плату до нашого ноутбука ми можемо побачити на рисунку 3.4.

					КВРКІ 200102.20.01.02.ПЗ	Арк.
						41
Зм.	Арк.	№ докум.	Підпис	Дата		

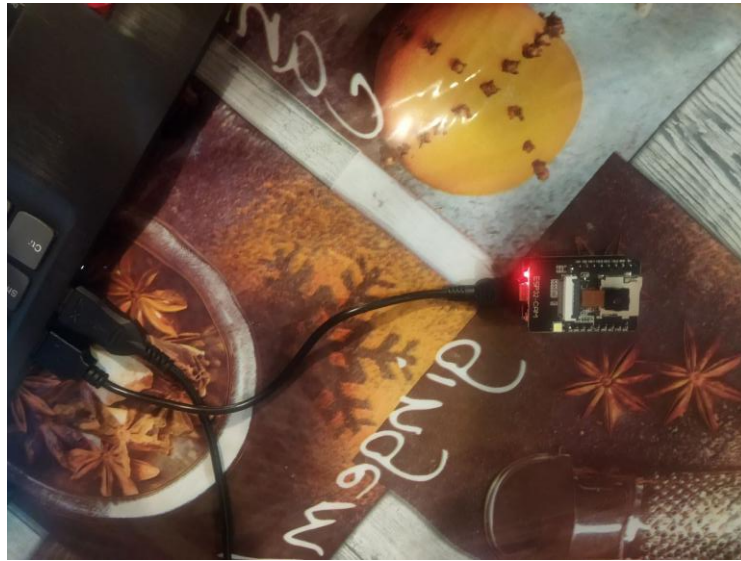


Рисунок 3.4 – Підключена плата до ноутбука

Після того як ми підключили плату до комп'ютера нам потрібно перевірити до якого саме порту вона підключена, це ми можемо зробити за допомогою диспетчера пристроїв.

Далі нам потрібно обрати порт в налаштуваннях. Це ми можемо зробити в вкладці «Інструменти».

Вибір порту до якого підключена плата ми можемо спостерігати на рисунку 3.5.

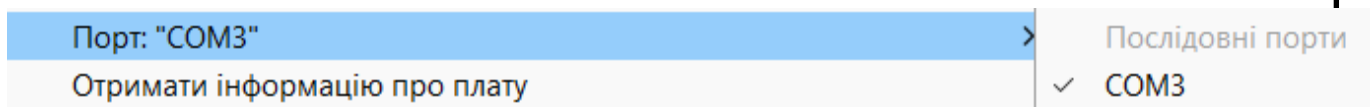


Рисунок 3.5 – Вибір порту до якого підключена плата

Після цього нам потрібно обрати нашу плату в інструментах, це ми можемо зробити в вкладці «Інструменти» > «Плата» > «ESP32 Arduino» і нам слід вибрати

нашу плату, а саме «AI Thinker ESP32-CAM». Ці дії ви можете побачити на рисунку 3.6.

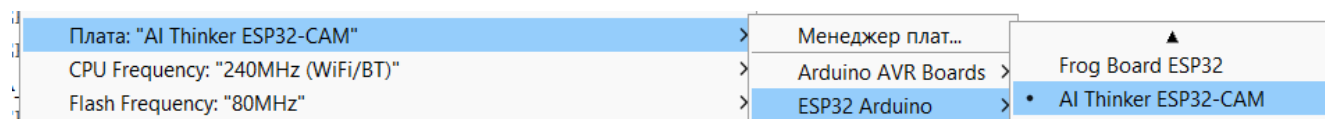


Рисунок 3.6 – Вибір плати в середовищі

Далі нам потрібно встановити усі необхідні бібліотеки такі як: Wi-Fi, ArduinoJson, ESP32-Mail-Client. Бібліотека Wi-Fi для ESP32 (ESP32-CAM) є однією з найважливіших для реалізації проєктів, які потребують бездротового з'єднання.

Вона надає функції для підключення до Wi-Fi мережі, управління з'єднанням і обміном даними через мережу.

Бібліотека ArduinoJson допоможе у нашому проєкті для обробки даних та зображень з нашої камери. ESP32-Mail-Client ця бібліотека дозволяє надсилати електронні листи з ESP32, що може бути корисним для надсилання зображень або повідомлень про події.

Приклад встановлення бібліотеки ми можемо побачити на рисунку 3.7.

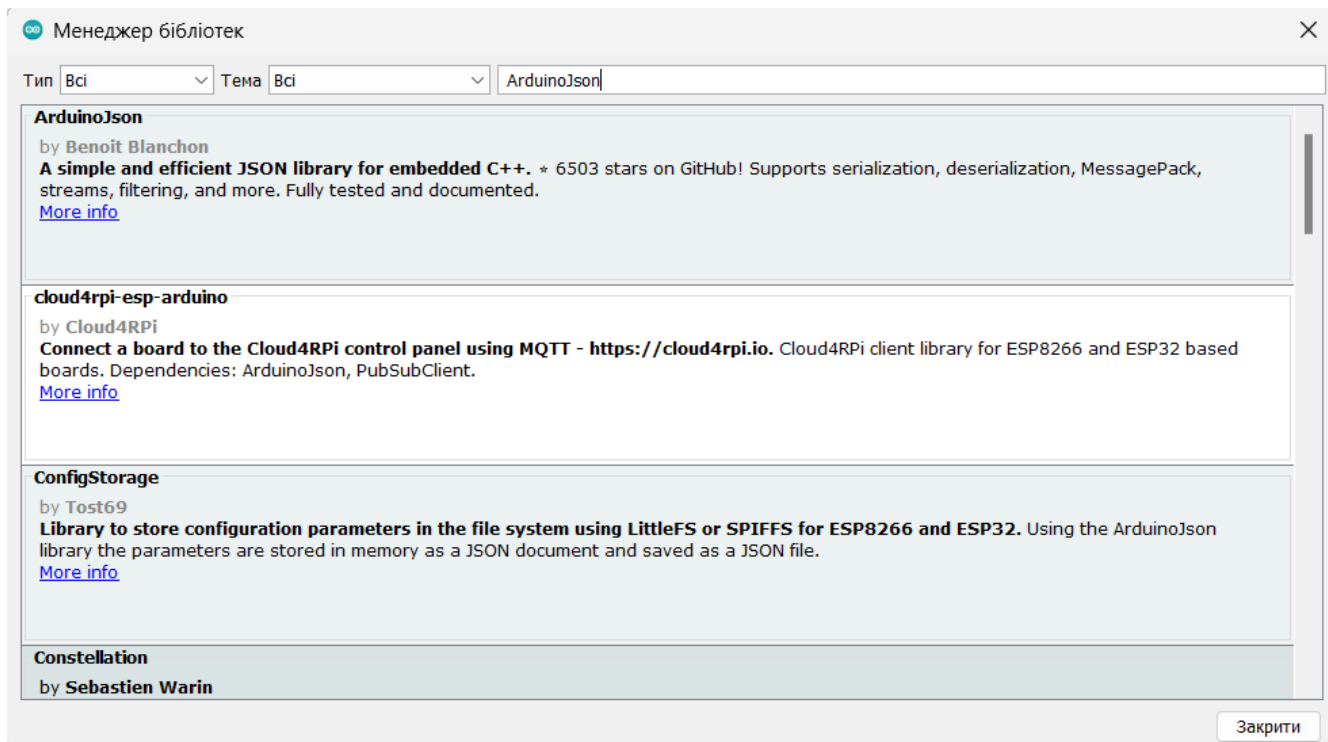


Рисунок 3.7 –Встановлення бібліотеки

Після завершення всіх попередніх кроків настав час протестувати плату ESP32-CAM, щоб переконатися, що вона працює належним чином.

Для цього ми можемо використовувати приклад CameraWebServer, який постачається разом із середовищем розробки Arduino, його ми можемо знайти в вкладці «Файл» > «Приклади» > «ESP32» > «Camera» там нам сліді обрати CameraWebServer.

Як це зробити ми можемо побачити на рисунку 3.8, на ньому показано як виконати ці дії.

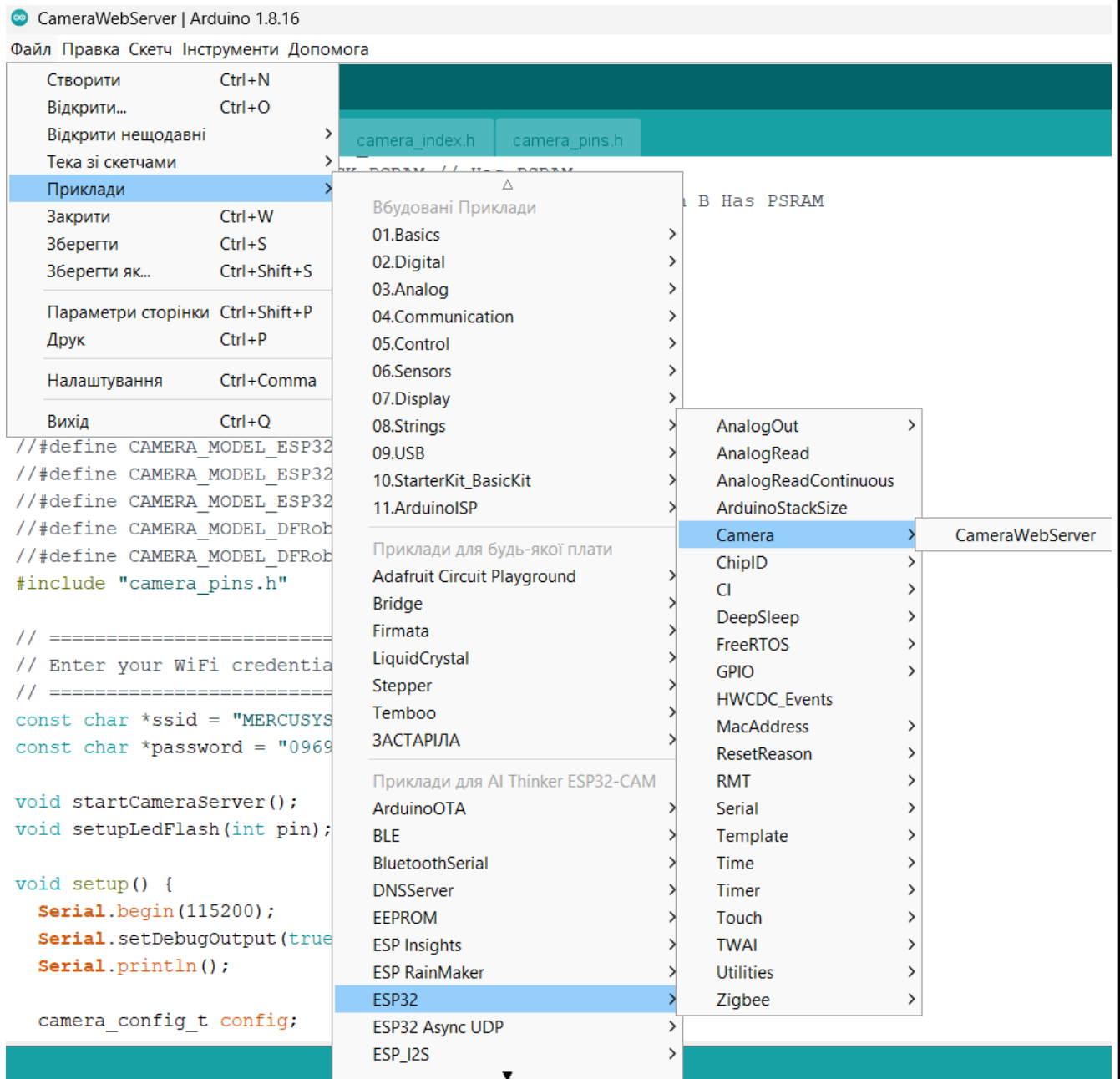


Рисунок 3.8 – Вибір прикладу для тесту плати

Після того як відкриється приклад нам потрібно змінити код, а саме вибрати необхідну плату та ввести логін і пароль від нашого WiFi. Після цих дій ми можемо спробувати скопіювати наш приклад, після того як скопіювати приклад нам потрібно його зберегти у папку та далі ми можемо спробувати завантажити його на нашу плату за допомогою кнопки. Цю кнопку ми можемо побачити на рисунку 3.9.

Рисунок 3.9-Кнопка прошивки плати

Після цих дій нам слід зайти в монітор послідовного порту за допомогою вкладки інструменти або за допомогою комбінації клавіш `ctrl+shift+M`, після цього на платі нам потрібно натиснути кнопку `reset`. Цю кнопку на самій платі ми можемо спостергати на рисунку 3.10.



Рисунок 3.10 - Кнопка reset на платі

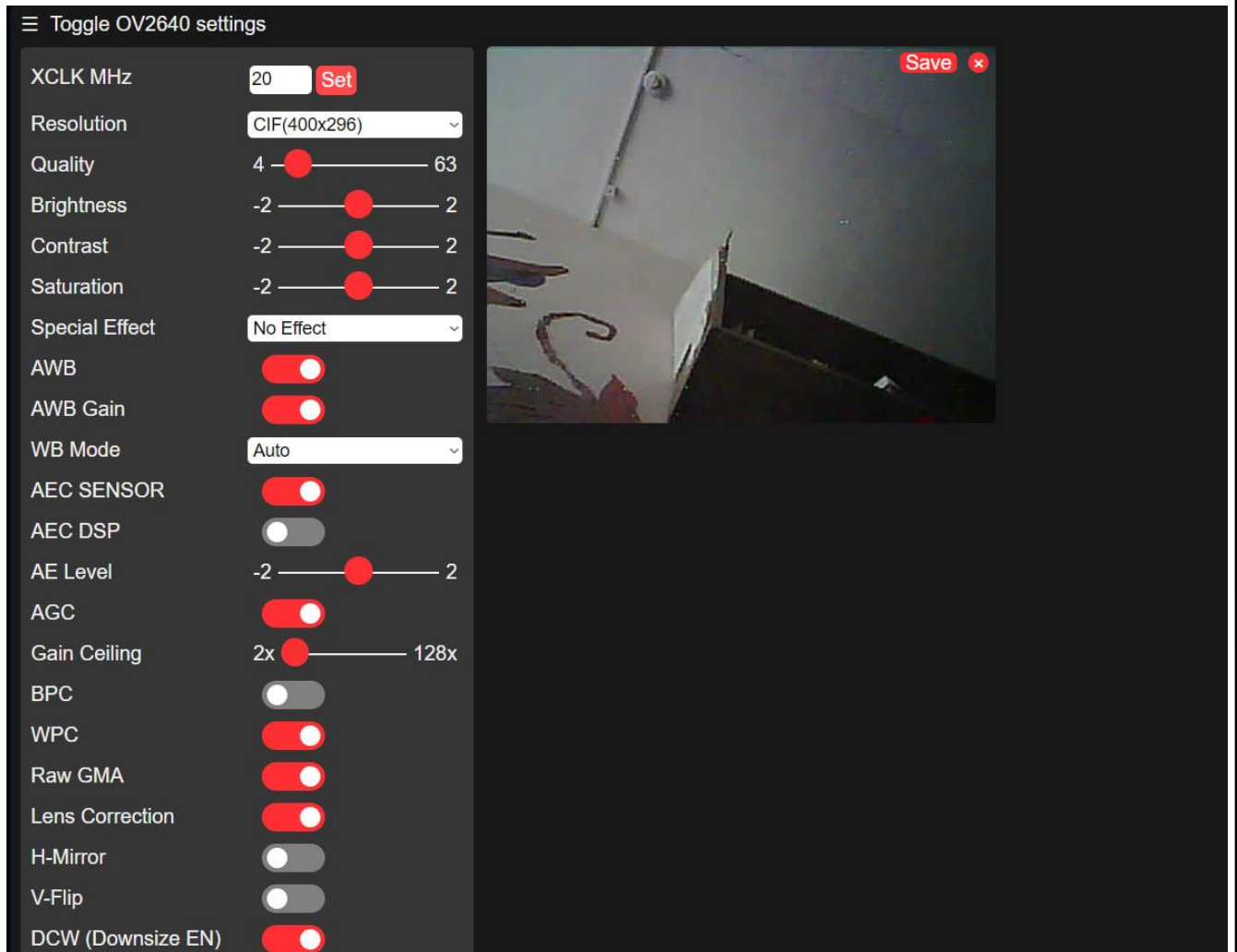


Рисунок 3.12 - Зображення з плати

3.2 Реалізація функцій для виконання алгоритму

Для створення системи автоматичного визначення розмірів в процесі токарної обробки деталей з використанням мікроконтролера ESP32-CAM, будемо реалізовувати кілька основних функцій. Система має виконувати захоплення зображення, обробку зображення, визначення розмірів та повідомлення у разі виявлення дефектів чи завершення обробки.

Для початку реалізації нашого проекту в нашому коді нам спочатку потрібно підключити усі необхідні бібліотеки, підключення всіх бібліотек показано в коді нижче.

```
#include <Arduino.h>
```

					КВРКІ 200102.20.01.02.ПЗ	Арк. 48
Зм.	Арк.	№ докум.	Підпис	Дата		

```
#include <WiFi.h>
#include <esp_camera.h>
#include <ESP_Mail_Client.h>
```

Arduino.h - це основна бібліотека для роботи з платформою Arduino. Вона забезпечує доступ до базових функцій мікроконтролера, таких як робота з цифровими і аналоговими входами/виходами, таймерами, інтерфейсами I2C, SPI і UART. WiFi.h – це бібліотека для роботи з Wi-Fi на платформі ESP32. Вона дозволяє підключатися до Wi-Fi мереж, встановлювати з'єднання з сервером, відправляти та отримувати дані через інтернет. esp_camera.h – це бібліотека для роботи з камерою ESP32-CAM. Вона забезпечує всі необхідні функції для ініціалізації, налаштування і захоплення зображень з камери. ESP_Mail_Client.h – ця бібліотека потрібна для роботи з електронною поштою на платформі ESP32. Вона дозволяє налаштувати SMTP-з'єднання, відправляти електронні листи з вкладеннями і текстом, що робить її корисною для відправки результатів обробки зображень на електронну пошту.

Далі нам потрібно визначити контакти GPIO, пов'язані з нашою моделлю камери:

```
#if defined(CAMERA_MODEL_AI_THINKER)
#define PWDN_GPIO_NUM    -1
#define RESET_GPIO_NUM  -1
#define XCLK_GPIO_NUM    0
#define SIOD_GPIO_NUM    26
#define SIOC_GPIO_NUM    27
#define Y9_GPIO_NUM      35
#define Y8_GPIO_NUM      34
#define Y7_GPIO_NUM      39
#define Y6_GPIO_NUM      36
#define Y5_GPIO_NUM      21
#define Y4_GPIO_NUM      19
```

```
#define Y3_GPIO_NUM    18
#define Y2_GPIO_NUM    5
#define VSYNC_GPIO_NUM 25
#define HREF_GPIO_NUM  23
#define PCLK_GPIO_NUM  22
```

Оскільки проєкт використовує Wi-Fi, потрібно вказати код для підключення до мережі. Для цього необхідно задати SSID і пароль Wi-Fi мережі, до якої підключається плата ESP32-CAM. Це можна зробити за допомогою наступного коду:

```
const char* ssid = "YOUR_SSID";
const char* password = "YOUR_PASSWORD";
```

В першому полі ми вказуємо назву нашого Wi-Fi, а в другому пароль. Далі нам потрібно ввести параметри пошти, куди буде надсилатися наші результати.

```
#define SMTP_HOST "smtp.gmail.com"
#define SMTP_PORT 465
#define AUTHOR_EMAIL "your_email@example.com"
#define AUTHOR_PASSWORD "your_password"
#define RECIPIENT_EMAIL "recipient_email@example.com"
```

Рядок `#define SMTP_HOST "smtp.example.com"` вказує адресу SMTP-сервера, який буде використовуватися для відправки електронної пошти. Другий рядок вказує номер порту, який використовується для підключення до SMTP-сервера. Далі нам потрібно вказати нашу пошту з якої ми будемо надсилати наше повідомлення та пароль від неї. Та в останньому рядку ми вказуємо пошту на яку будуть приходити наші результати.

Після цього нам слід зробити ініціалізацію камери на ESP32-CAM.

```
void initCamera() {
    camera_config_t config;
    config.ledc_channel = LEDC_CHANNEL_0;
    config.ledc_timer = LEDC_TIMER_0;
```

```
config.pin_d0 = Y2_GPIO_NUM;
config.pin_d1 = Y3_GPIO_NUM;
config.pin_d2 = Y4_GPIO_NUM;
config.pin_d3 = Y5_GPIO_NUM;
config.pin_d4 = Y6_GPIO_NUM;
config.pin_d5 = Y7_GPIO_NUM;
config.pin_d6 = Y8_GPIO_NUM;
config.pin_d7 = Y9_GPIO_NUM;
config.pin_xclk = XCLK_GPIO_NUM;
config.pin_pclk = PCLK_GPIO_NUM;
config.pin_vsync = VSYNC_GPIO_NUM;
config.pin_href = HREF_GPIO_NUM;
config.pin_sscb_sda = SIOD_GPIO_NUM;
config.pin_sscb_scl = SIOC_GPIO_NUM;
config.pin_pwdn = PWDN_GPIO_NUM;
config.pin_reset = RESET_GPIO_NUM;
config.xclk_freq_hz = 20000000;
config.pixel_format = PIXFORMAT_JPEG;
```

```
if(psramFound()){
    config.frame_size = FRAMESIZE_UXGA;
    config.jpeg_quality = 10;
    config.fb_count = 2;
} else {
    config.frame_size = FRAMESIZE_SVGA;
    config.jpeg_quality = 12;
    config.fb_count = 1;
}
```

```

// Ініціалізація камери
if (!esp_camera_init(&config)) {
    Serial.println("Камера ініціалізована успішно");
} else {
    Serial.println("Помилка ініціалізації камери");
    return;
}
}

```

Ця функція забезпечує належну роботу камери, що дозволяє ESP32-CAM захоплювати зображення відповідно до встановлених параметрів.

Далі нам потрібно налаштувати підключення до нашого Wi-Fi, це ми можемо зробити за допомогою наступного коду:

```

void connectToWiFi() {
    WiFi.begin(ssid, password);
    while (WiFi.status() != WL_CONNECTED) {
        delay(500);
        Serial.print(".");
    }
    Serial.println("WiFi підключено");
}

```

Після цього ми налаштуємо відправку електронної пошти з результатами наступним кодом :

```

void sendEmail(String subject, String message) {
    SMTPData smtpData;

    smtpData.setLogin(SMTP_HOST, SMTP_PORT, AUTHOR_EMAIL,
AUTHOR_PASSWORD);
    smtpData.setSender("ESP32-CAM", AUTHOR_EMAIL);
    smtpData.setPriority("High");
}

```

```
smtpData.setSubject(subject);
smtpData.setMessage(message, false);
smtpData.addRecipient(RECIPIENT_EMAIL);
```

```
if (!MailClient.sendMail(smtp, smtpData)) {
    Serial.println("Помилка відправки пошти: " +
MailClient.smtpErrorReason());
}
```

```
smtpData.empty();
}
```

Далі нам потрібно написати код для обробки зображень. Перетворення зображення у відтінки сірого, визначення країв деталі та обробки зображення для визначення розмірів.

```
void rgbToGrayscale(const uint8_t* rgb, uint8_t* grayscale, int width, int height)
{
    for (int i = 0; i < width * height; i++) {
        int r = rgb[i * 3];
        int g = rgb[i * 3 + 1];
        int b = rgb[i * 3 + 2];
        grayscale[i] = (r * 30 + g * 59 + b * 11) / 100;
    }
}
```

В цій частині коду ми перетворюємо наше зображення у відтінки сірого. Це потрібно нам для того щоб спростити обчислення та зменшувати розмір даних обробки. Кольорові зображення зазвичай мають три канали (червоний, зелений, синій – RGB), кожен з яких містить інформацію про інтенсивність кольору в кожному пікселі. Перетворення в градації сірого зменшує кількість даних, які потрібно обробляти, оскільки для кожного пікселя потрібно лише одне значення

замість трьох. Багато алгоритмів обробки зображень, таких як виявлення країв, розпізнавання контурів та сегментація зображень, працюють ефективніше і швидше на зображеннях у відтінках сірого. Це пояснюється тим, що вони можуть зосередитися на інтенсивності світла, а не на кольорах.

Далі нам потрібно визначити краї нашої заготовки. Це ми можемо зробити за допомогою наступної функції:

```
void edgeDetection(const uint8_t* grayscale, uint8_t* edges, int width, int height) {
    for (int y = 1; y < height - 1; y++) {
        for (int x = 1; x < width - 1; x++) {
            int gx = -grayscale[(y-1) * width + (x-1)] - 2 * grayscale[y * width + (x-1)] -
grayscale[(y+1) * width + (x-1)]
                + grayscale[(y-1) * width + (x+1)] + 2 * grayscale[y * width + (x+1)] +
grayscale[(y+1) * width + (x+1)];
            int gy = -grayscale[(y-1) * width + (x-1)] - 2 * grayscale[(y-1) * width + x] -
grayscale[(y-1) * width + (x+1)]
                + grayscale[(y+1) * width + (x-1)] + 2 * grayscale[(y+1) * width + x] +
grayscale[(y+1) * width + (x+1)];
            int g = abs(gx) + abs(gy);
            edges[y * width + x] = g > 128 ? 255 : 0; // Простий поріг
        }
    }
}
```

Далі ми створюємо функцію для обробки зображення та визначення розміру деталі.

```
void calculateDimensions(camera_fb_t* fb) {
    int width = fb->width;
    int height = fb->height;
    std::vector<uint8_t> grayscale(width * height);
```

```
std::vector<uint8_t> edges(width * height);
```

Далі аналізуємо контури та визначаємо розміри, обробляєм кожен з країв, та знаходимо максимальні і мінімальні значення для визначення розмірів.

```
std::vector<int> dimensions;
```

```
for (int y = 0; y < height; y++) {
```

```
    for (int x = 0; x < width; x++) {
```

```
        if (edges[y * width + x] == 255) {
```

```
            // Обробка кожного знайденого краю (це спрощено для прикладу)
```

```
            dimensions.push_back(y);
```

```
            dimensions.push_back(x);
```

```
        }
```

```
    }
```

```
}
```

```
int min_x = *std::min_element(dimensions.begin(), dimensions.end());
```

```
int max_x = *std::max_element(dimensions.begin(), dimensions.end());
```

```
int min_y = *std::min_element(dimensions.begin(), dimensions.end());
```

```
int max_y = *std::max_element(dimensions.begin(), dimensions.end());
```

```
int width_dim = max_x - min_x;
```

```
int height_dim = max_y - min_y;
```

Та після цього відправляємо результати на пошту

```
String result = "Width: " + String(width_dim) + " px, Height: " +  
String(height_dim) + " px";
```

```
    sendEmail(result);
```

```
}
```

3.3 Опис блок-схеми, схеми з'єднань мікроконтролера з камерою, електричної схеми

Роботу блок-схеми системи автоматичного визначення розмірів деталей в процесі токарної обробки з використанням мікроконтролера ESP32-CAM ми можемо спостерігати на рисунку 3.13.

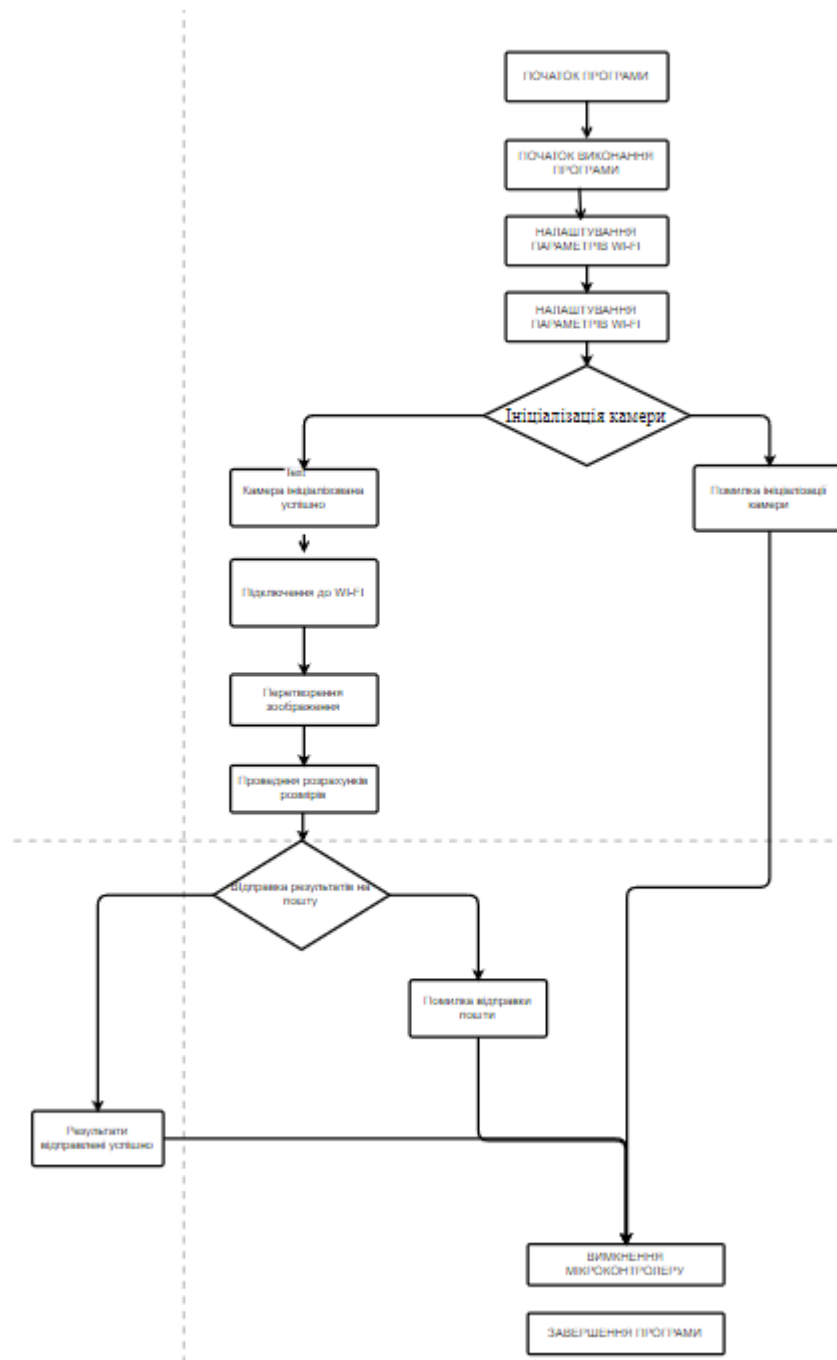


Рисунок 3.13 – Блок-схема роботи програми

Схема з'єднань мікроконтролеру з камерою та для виконання прошивки мікроконтролеру ми можемо побачити на рисунку 3.14 .

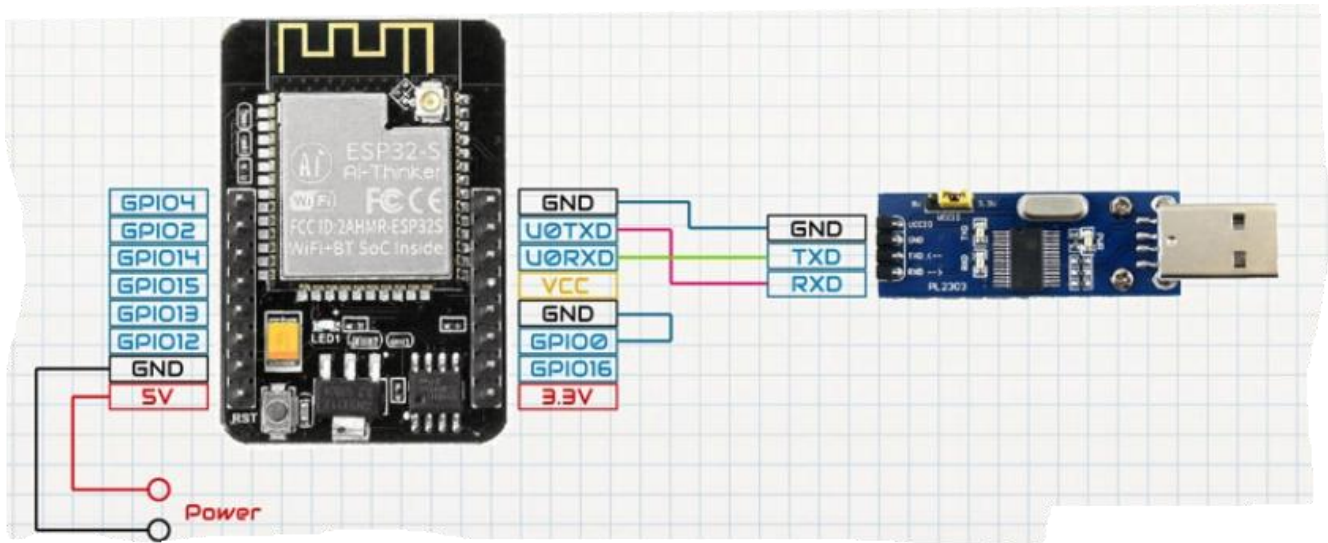


Рисунок 3.14 - Схема з'єднань мікроконтролеру з камерою та для виконання прошивки мікроконтролеру

Електрична схема модуля ESP32-CAM для нашого проекту представлена на рисунку 3.15 .

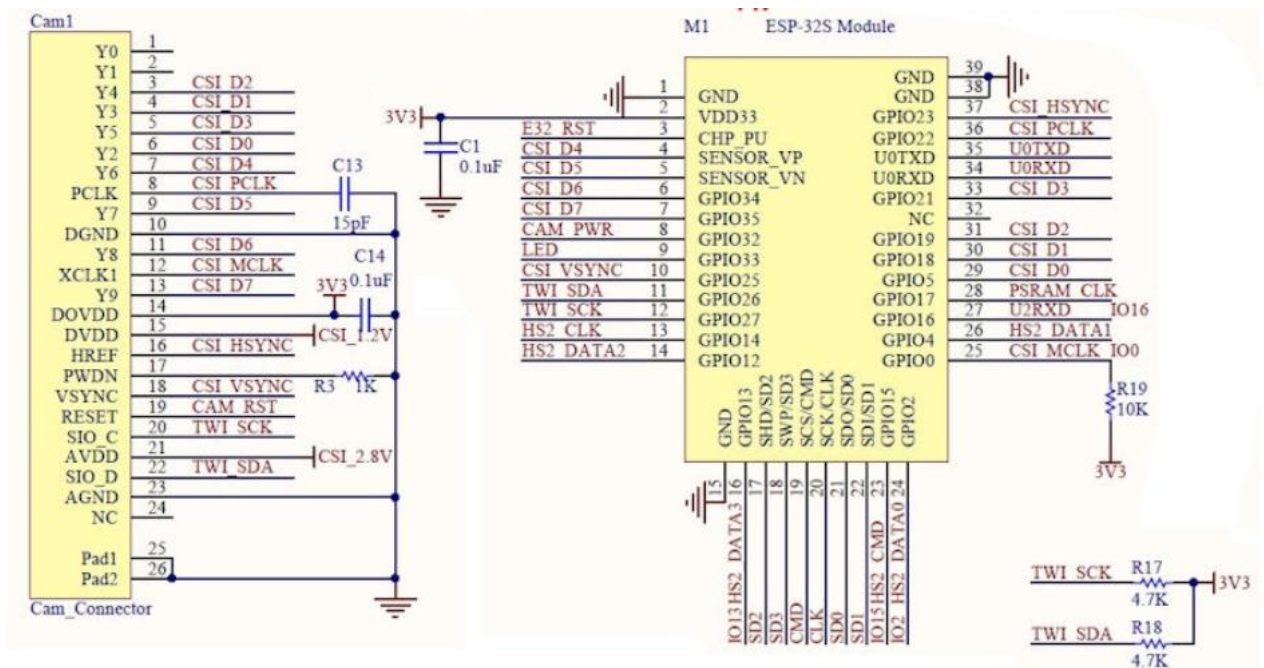


Рисунок 3.15 - Електрична схема модуля ESP32-CAM

3.4 Розрахунок матеріальних витрат та напрямки вдосконалення

Для створення системи автоматичного визначення розмірів деталей в процесі токарної обробки з використанням мікроконтролера ESP32-CAM необхідно врахувати вартість всіх компонентів та витратних матеріалів.

Розглянемо основні складові проекту:

- плата з мікроконтролером ESP32CAM, приблизна вартість приблизно 400-550 гривень;
- модуль Wi-Fi, вартість включена у плату;
- кабель живлення Micro-USB, приблизно 100-150 гривень;
- середовище для розробки Arduino IDE, безкоштовне програмне забезпечення для програмування ESP32-CAM.

Таблиця 3.1 – Вартість компонентів системи автоматичного визначення розмірів деталей

№	Назви компонентів	Ціна, грн
1	плата з мікроконтролером ESP32CAM	500
2	Модуль Wi-Fi	0
3	Кабель живлення Micro-USB	100
4	Середовище для розробки Arduino IDE	0

Цей розрахунок охоплює основні витрати на компоненти та матеріали для створення системи автоматичного визначення розмірів деталей в процесі токарної обробки з використанням ESP32-CAM. Додаткові витрати можуть виникнути в залежності від специфічних вимог проекту та наявності деяких компонентів. Тобто весь готовий проект обійшовся нам приблизно в 600 гривень.

Для вдосконалення системи автоматичного визначення розмірів деталей в процесі токарної обробки з використанням ESP32-CAM можна зробити кілька кроків. По-перше, необхідно покращити точність вимірювань, що можливо

завдяки калібруванню камери та використанню камер з вищою роздільною здатністю. Це дозволить зменшити спотворення об'єктива та підвищити якість зображень, що, в свою чергу, підвищить точність вимірювань.

По-друге, можна розширити функціональні можливості системи. Наприклад, інтеграція з системами ЧПУ дозволить автоматично налаштовувати параметри токарної обробки на основі результатів вимірювань. Також використання технологій машинного навчання дозволить автоматично розпізнавати та класифікувати деталі, що значно підвищить ефективність системи.

Крім того, покращення комунікації та інтерфейсу користувача може бути досягнуто за рахунок додавання веб-інтерфейсу для віддаленого моніторингу та керування системою. Вдосконалення системи сповіщення шляхом додавання можливості відправки повідомлень через різні месенджери (Telegram, WhatsApp) також може бути корисним.

Щоб підвищити надійність та стійкість системи, варто розглянути використання захисних корпусів для камери та інших електронних компонентів для захисту від пилу та вібрацій. Також додавання резервних джерел живлення забезпечить безперебійну роботу системи у випадку збоїв в електропостачанні.

Оптимізація обробки зображень є ще одним напрямом для вдосконалення. Використання багатопоточності або паралельної обробки дозволить пришвидшити аналіз зображень. Крім того, оптимізація алгоритмів обробки зображень зменшить навантаження на мікроконтролер та підвищить швидкість обробки.

Інтеграція системи з іншими системами управління підприємством (ERP-системами) дозволить автоматично оновлювати інформацію про стан обробки деталей. Також інтеграція у мережу Інтернет речей (IoT) дасть можливість централізовано збирати та аналізувати дані з декількох токарних верстатів.

Нарешті, підвищення безпеки системи може бути досягнуто шляхом впровадження шифрування даних при передачі через мережу та реалізації

системи аутентифікації та авторизації користувачів для захисту від несанкціонованого доступу. Всі ці вдосконалення сприятимуть підвищенню ефективності, точності та надійності системи в умовах реального виробництва.

3.5 Висновки

Після детального опису встановлення необхідних компонентів для Arduino IDE було представлено програмну частину ESP32CAM. Система визначення розмірів деталей в процесі токарної обробки на основі ESP32CAM зібрана з компонентів таких як: плата мікроконтролера ESP32CAM та камера OV2640.

Процес розробки програми почався з скачування та налаштування Arduino IDE після всіх налаштувань ми створили код для нашої програми системи визначення розмірів деталей в процесі токарної обробки. Також були створені схеми для нашого проекту та описана робота функцій нашого коду.

Були розраховані витрати на створення системи визначення розмірів деталей в процесі токарної обробки. Система включає в себе модуль ESP32CAM та кабель живлення Micro-USB Після проведення розрахунків ми дійшли до висновки що наш проект обійшовся нам приблизно у 600 гривень.

Також ми спробували роглянути варіанти вдосконалення нашої системи. Для вдосконалення системи автоматичного визначення розмірів деталей у процесі токарної обробки з використанням ESP32-CAM можна здійснити кілька важливих кроків. Підвищення точності вимірювань шляхом калібрування камери та використання камер з вищою роздільною здатністю значно покращить якість результатів. Розширення функціональних можливостей, наприклад, через інтеграцію з системами ЧПУ та застосування технологій машинного навчання, дозволить автоматично налаштовувати параметри обробки та розпізнавати різні деталі. Крім того, підвищення безпеки через шифрування даних та впровадження аутентифікації користувачів забезпечить захист від несанкціонованого доступу.

					КВРКІ 200102.20.01.02.ПЗ	Арк. 60
Зм.	Арк.	№ докум.	Підпис	Дата		

ВИСНОВОК

У ході виконання даного проєкту було проведено всебічне дослідження, розробку та тестування системи автоматичного визначення розмірів деталей у процесі токарної обробки з використанням мікроконтролеру ESP32-CAM.

У першому розділі зроблено огляд відомих методів та рішень у сфері мікроконтролерів і токарних верстатів, що дозволило зрозуміти, як використовуються мікроконтролери у токарних верстатах та які існують типи токарних верстатів. На основі цього аналізу було сформульовано завдання проєкту.

У другому розділі був вибір програмного забезпечення та обладнання для реалізації системи. Було детально розглянуто технічні характеристики мікроконтролеру ESP32-CAM, переваги та недоліки його використання у порівнянні з іншими мікроконтролерами, а також вибір відповідної камери для виконання завдання.

У третьому розділі була проведена програмна реалізація та тестування програмно-технічного засобу. Було здійснено встановлення необхідних ресурсів та середовища Arduino IDE, реалізацію функцій для виконання алгоритму автоматичного визначення розмірів деталей, а також створення блок-схеми, схеми з'єднань мікроконтролеру з камерою та електричної схеми.

Було також розраховано матеріальні витрати на реалізацію проєкту та визначено можливі напрямки вдосконалення системи, такі як підвищення точності вимірювань, розширення функціональних можливостей та інтеграція з іншими системами.

Загалом, проєкт продемонстрував ефективність використання мікроконтролеру ESP32-CAM для автоматичного визначення розмірів деталей під час токарної обробки, що може бути корисним у промисловості для підвищення точності та ефективності виробничих процесів.

					КВРКІ 200102.20.01.02.ПЗ	Арк. 61
Зм.	Арк.	№ докум.	Підпис	Дата		

Мета роботи досягнута.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1. Історія створення та розвитку мікропроцесорних засобів автоматизації. URL: <https://elprivod.nmu.org.ua/ua/interesting/hmpa/Історія%20створення%20та%20розвитку%20МП%20засобів%20авт.pdf> (дата звернення: 16.02.2024)
2. Архітектура мікроконтролерів з CISC архітектурою. URL: http://radiovtc.inf.ua/mps/lesson/L16_2012.htm (дата звернення: 16.02.2024)
3. Що таке мікропроцесор, мікроконтролер та програмований логічний контролер. URL: https://elprivod.nmu.org.ua/ua/interesting/what_is_mp_mc_plc.php(дата звернення: 17.02.2024).
4. Все про мікроконтролерах. URL: https://elektrovesti.net/interesting/642_vseo-mikrokontrollerakh. (дата звернення: 17.02.2024).
5. Глухов О.В., Кравчук О.О., Левченко Є.В. Вивчення властивостей мікроконтролерів і електронних систем на базі платформи Ардуіно: навч. посіб. Харків : ХНУРЕ, 2019. 192 с.
6. Конспект лекцій з дисципліни «Мікропроцесорна техніка». URL: <http://eprints.kname.edu.ua/43462/1/92Л%20Конспект%20МПТ%202015.pdf>. (дата звернення: 18.02.2024).
7. Переваги використання мікроконтролерів у портативних медичнихпристроях. URL: https://mts.nure.ua/wp-content/uploads/2021/03/aert-2019_web_14-16.pdf (дата звернення: 18.02.2024).
8. Про системи керування промисловими роботами. URL: <https://ela.kpi.ua/bitstream/123456789/46800/1/Page%20171-174.pdf>. (дата звернення: 18.02.2024).
9. Історія токарного верстата. URL: <http://ua.eomachinetool.com/news/the-history-of-lathe-14263657.html>(дата звернення: 20.02.2024).

					КВРКІ 200102.20.01.02.ПЗ	Арк. 62
Зм.	Арк.	№ докум.	Підпис	Дата		

10. Історія розвитку токарного верстата. URL: <https://www.istoriya.in.ua/istoriya-rozvitku-tokarnogo-verstata.html>(дата звернення: 20.02.2024).
11. Історія розвитку токарного верстата. URL: https://uk.wikipedia.org/wiki/Токарна_група_верстатів(дата звернення: 20.02.2024).
12. Типи токарних верстатів. URL: <https://studfile.net/preview/9699030/page:2/>(дата звернення: 23.02.2024)
13. Основні вузли токарного верстату. URL: http://www.tsatu.edu.ua/tkm/wp-content/uploads/sites/11/mehanych_obrob_tokr_verstat.pdf (дата звернення: 23.02.2024)
14. Токарно-гвинторізний верстат 16K20 фото. URL: <https://kievstanko.com.ua/ua/p4363910-tokarno-vintoreznyj-16k20.html>(дата звернення: 23.02.2024)
15. Токарно-лобові верстати. URL: <https://eurotools.ua/uk/111-tokarnye-stanki>(дата звернення: 25.02.2024).
16. Токарно-лобовий верстат фото. URL: <https://ua.all.biz/uk/verstaty-lobotokarni-ukrayina-eksport-g1939921>(дата звернення: 25.02.2024).
17. Токарно-карусельні верстати. URL: <https://buklib.net/books/36306>(дата звернення: 27.02.2024)
18. Токарно-карусельний верстат фото. URL: <https://stankocentr.com.ua/uk/purchase/tokarno-karuselnye-stanki/tokarno-karuselnyi-standok-1516-1516f1>(дата звернення: 27.02.2024)
19. Токарно-револьверний верстат. URL: <https://buklib.net/books/36305/>(дата звернення: 27.02.2024)
20. Токарно-револьверний верстат фото. URL: <https://www.met.ua/p/tokarno-revolvorny-verstat-z-chpk-fbl-530/>(дата звернення: 01.03.2024).

					КВРКІ 200102.20.01.02.ПЗ	Арк. 63
Зм.	Арк.	№ докум.	Підпис	Дата		

21. Центрувально-відрізні верстати. URL: https://publish.com.ua/nashi-dni/vidrizni-verstati-vidi-kharakteristiki-priznachennya.html#google_vignette(дата звернення: 01.03.2024).
22. Центрувально-відрізні верстати фото. URL:<https://ntc-melita.com.ua/ua/p1132008748-stanok-tokarno-vintoreznyj.html>(дата звернення: 01.03.2024).
23. Багаторізцевий токарний верстат. URL: https://www.wikidata.uk-ua.nina.az/Багаторізцевий_верстат.html(дата звернення: 05.03.2024).
24. Багаторізцевий токарний верстат фото. URL: <https://9craft.com.ua/ua/p1755825842-nastolnyj-tokarno-vintoreznyj.html>(дата звернення: 05.03.2024).
25. Токарні автомати. URL: <https://buklib.net/books/36308/>(дата звернення: 05.03.2024).
26. Токарні напівавтомати. URL: <https://buklib.net/books/36307/> (дата звернення: 10.03.2024).
27. Токарні автомати фото. URL:<https://abplanalp.ua/verstat-tokarnij-avtomat-shvejcarskogo-tipu-tornos-ct20>(дата звернення: 10.03.2024).
28. Токарні напівавтомати фото. URL:<https://kievstanko.com.ua/ua/p233791996-mnogoshpindelnye-tokarnye-avtomaty.html>(дата звернення: 10.03.2024).
29. Історія числового програмного управління. URL: <https://sitelab-15.dss-bi.com.ua/index.php/statti/6-istoriia-chyslovoho-prohramnoho-upravlinnia>(дата звернення: 18.03.2024)
30. Історія компанії Parsons Inc. URL: <https://www.wikidata.uk-ua.nina.az/ЧПК.html>(дата звернення: 18.03.2024)
31. Історія Массачусетського технологічного інституту. URL: https://uk.wikipedia.org/wiki/Массачусетський_технологічний_інститут(дата звернення: 18.03.2024)
32. Перший серійний пристрій. URL: https://www.wikidata.uk-ua.nina.az/Програмне_керування.html(дата звернення: 20.03.2024).

					КВРКІ 200102.20.01.02.ПЗ	Арк. 64
Зм.	Арк.	№ докум.	Підпис	Дата		

33. Інформація про G-Code. URL: <https://raptorcnc.com.ua/ua/a476957-spisok-code-code.html>(дата звернення: 20.03.2024).
34. Інформація про модуль. URL: <https://1wire.com.ua/modul-esp32-cam-s-kameroj-ov2640-2mp-ch340.html>(дата звернення: 20.03.2024).
35. ESP32-CAM з під'єднаною камерою фото. URL: <https://electronica.in.ua/ua/p1530391836-modul-esp32-cam.html>(дата звернення:: 12.04.2024).
36. Основні характеристики модулю. URL: <https://radiostore.com.ua/ua/p1110434996-modul-kamery-esp32.html>(дата звернення:: 12.04.2024).
37. Переваги мікроконтролера. URL: <https://radiostore.ua/products/esp32-cam-wi-fi-esp32-bluetooth-s-ov2640>(дата звернення:: 12.04.2024).
38. Камера OV7670. URL: <https://beegreen.com.ua/modul-vga-kameri-ov7670-arduino-avr-pic-10120>(дата звернення:: 17.04.2024).
39. Зовнішній вигляд камери OV7670. URL: <https://ardushop.in.ua/arduino/vga-camera-module-ov7670>(дата звернення:: 17.04.2024).
40. Камера OV2640. URL: <https://arduino.ua/prod4501-modyl-kameri-2mp-ov2640-dlya-esp32-cam-fov160>(дата звернення:: 17.04.2024).
41. Зовнішній вигляд камери OV2640. URL: <https://arduino.ua/prod5194-modyl-kameri-ov2640-2mpfov160-nv-dlya-esp-cam>(дата звернення:: 17.04.2024).
42. Сфери використання мікроконтролеру. URL: <https://cyberdid.medium.com/що-таке-esp-d1267d5877cf>(дата звернення:: 24.04.2024).
43. Мікроконтролер Arduino Mega 2560. URL: <https://arduinokit.com.ua/ua/p1080109370-arduino-mega-2560.html#:~:text=Arduino%20Mega%202560%20-%20это%20улучшенная,выходы%2C%20входы%20-%2054%20шт.> (дата звернення:: 24.04.2024).

					КВРКІ 200102.20.01.02.ПЗ	Арк. 65
Зм.	Арк.	№ докум.	Підпис	Дата		

44. Зовнішній вигляд Arduino Mega 2560. URL: <https://sxema.com.ua/ua/p745466128-arduino-mega-2560.html> (дата звернення: 24.04.2024).
45. Мікроконтролер raspberry pi pico. URL: <https://arduino.ua/prod4358-raspberry-pi-pico> (дата звернення: 1.05.2024).
46. Зовнішній вигляд Raspberry Pi Pico. URL: <https://picockpit.com/raspberry-pi/everything-about-the-raspberry-pi-pico/> (дата звернення: 1.05.2024).
47. Зовнішній вигляд Raspberry Pi Zero W. URL: <https://evo.net.ua/raspberry-pi-zero-w/> (дата звернення: 1.05.2024).
48. Мікроконтролер Raspberry Pi Zero W. URL: <https://arduino.ua/prod2345-raspberry-pi-zero-w> (дата звернення: 1.05.2024).
49. Мікроконтролер STM32F4 Discovery Kit. URL: <https://arduino.ua/prod1122-plata-razrabotchika-stm32f407vg-discovery> (дата звернення: 4.05.2024).
50. Зовнішній вигляд STM32F4 Discovery Kit URL: <https://www.st.com/en/evaluation-tools/stm32f4discovery.html> (дата звернення: 4.05.2024).
51. Зовнішній вигляд BeagleBone Black. URL : <https://evo.net.ua/mikrokomputer-beaglebone-black-industrial/> (дата звернення: 4.05.2024).
52. Мікроконтролер BeagleBone Black. URL: <https://diylab.com.ua/p244164517-beaglebone-black-revc.html> (дата звернення: 4.05.2024).
53. ESP32-CAM та Arduino Uno. URL: <https://www.electronicclinic.com/esp32-cam-esp32-camera-programming-using-arduino-issues-fixed/> (дата звернення: 7.05.2024).
54. Приклад взаємодії ESP32-CAM та Arduino Uno фото. URL: <https://www.electronicclinic.com/esp32-cam-esp32-camera-programming-using-arduino-issues-fixed/> (дата звернення: 7.05.2024).
55. ESP32-CAM та Raspberry Pi основі. URL: https://www.youtube.com/watch?v=Vql_2CnUYw4 (дата звернення: 12.05.2024).

					КВРКІ 200102.20.01.02.ПЗ	Арк. 66
Зм.	Арк.	№ докум.	Підпис	Дата		

56. Мікроконтролер STM32 фото. URL: <https://detalipro.in.ua/kontroler-dlja-rozrobki-stm32f103c8t6-arm-32-cortex-m3-stm32-diymore>(дата звернення: 12.05.2024).

57. Arduino IDE. URL: <https://www.arduino.cc>(дата звернення: 19.05.2024).

					КВРКІ 200102.20.01.02.ПЗ	Арк. 67
Зм.	Арк.	№ докум.	Підпис	Дата		


```
#define RESET_GPIO_NUM  -1
#define XCLK_GPIO_NUM    0
#define SIOD_GPIO_NUM    26
#define SIOC_GPIO_NUM    27
#define Y9_GPIO_NUM      35
#define Y8_GPIO_NUM      34
#define Y7_GPIO_NUM      39
#define Y6_GPIO_NUM      36
#define Y5_GPIO_NUM      21
#define Y4_GPIO_NUM      19
#define Y3_GPIO_NUM      18
#define Y2_GPIO_NUM       5
#define VSYNC_GPIO_NUM   25
#define HREF_GPIO_NUM    23
#define PCLK_GPIO_NUM    22

// Параметри WiFi
const char* ssid = "YOUR_SSID";
const char* password = "YOUR_PASSWORD";

// Параметри пошти
#define SMTP_HOST "smtp.example.com"
#define SMTP_PORT 465
#define AUTHOR_EMAIL "your_email@example.com"
#define AUTHOR_PASSWORD "your_password"
#define RECIPIENT_EMAIL "recipient_email@example.com"

// Ініціалізація об'єкту поштового клієнта
SMTPSession smtp;
```

```

// Ініціалізація камери
void initCamera() {
    camera_config_t config;
    config.ledc_channel = LEDC_CHANNEL_0;
    config.ledc_timer = LEDC_TIMER_0;
    config.pin_d0 = Y2_GPIO_NUM;
    config.pin_d1 = Y3_GPIO_NUM;
    config.pin_d2 = Y4_GPIO_NUM;
    config.pin_d3 = Y5_GPIO_NUM;
    config.pin_d4 = Y6_GPIO_NUM;
    config.pin_d5 = Y7_GPIO_NUM;
    config.pin_d6 = Y8_GPIO_NUM;
    config.pin_d7 = Y9_GPIO_NUM;
    config.pin_xclk = XCLK_GPIO_NUM;
    config.pin_pclk = PCLK_GPIO_NUM;
    config.pin_vsync = VSYNC_GPIO_NUM;
    config.pin_href = HREF_GPIO_NUM;
    config.pin_sscb_sda = SIOD_GPIO_NUM;
    config.pin_sscb_scl = SIOC_GPIO_NUM;
    config.pin_pwdn = PWDN_GPIO_NUM;
    config.pin_reset = RESET_GPIO_NUM;
    config.xclk_freq_hz = 20000000;
    config.pixel_format = PIXFORMAT_JPEG;

    if(psramFound()){
        config.frame_size = FRAMESIZE_UXGA;
        config.jpeg_quality = 10;
        config.fb_count = 2;
    }
}

```

```

} else {
    config.frame_size = FRAMESIZE_SVGA;
    config.jpeg_quality = 12;
    config.fb_count = 1;
}

// Ініціалізація камери
if (!esp_camera_init(&config)) {
    Serial.println("Камера ініціалізована успішно");
} else {
    Serial.println("Помилка ініціалізації камери");
    return;
}

// Підключення до WiFi
void connectToWiFi() {
    WiFi.begin(ssid, password);
    while (WiFi.status() != WL_CONNECTED) {
        delay(500);
        Serial.print(".");
    }
    Serial.println("WiFi підключено");
}

// Перетворення RGB-зображення у відтінки сірого
void rgbToGrayscale(const uint8_t* rgb, uint8_t* grayscale, int width, int height)
{
    for (int i = 0; i < width * height; i++) {

```

```

int r = rgb[i * 3];
int g = rgb[i * 3 + 1];
int b = rgb[i * 3 + 2];
grayscale[i] = (r * 30 + g * 59 + b * 11) / 100;
}
}
// Проста функція для виявлення країв (Собель)
void edgeDetection(const uint8_t* grayscale, uint8_t* edges, int width, int
height) {
    for (int y = 1; y < height - 1; y++) {
        for (int x = 1; x < width - 1; x++) {
            int gx = -grayscale[(y-1) * width + (x-1)] - 2 * grayscale[y * width + (x-1)] -
grayscale[(y+1) * width + (x-1)]
                + grayscale[(y-1) * width + (x+1)] + 2 * grayscale[y * width + (x+1)] +
grayscale[(y+1) * width + (x+1)];
            int gy = -grayscale[(y-1) * width + (x-1)] - 2 * grayscale[(y-1) * width + x] -
grayscale[(y-1) * width + (x+1)]
                + grayscale[(y+1) * width + (x-1)] + 2 * grayscale[(y+1) * width + x] +
grayscale[(y+1) * width + (x+1)];
            int g = abs(gx) + abs(gy);
            edges[y * width + x] = g > 128 ? 255 : 0; // Простий поріг
        }
    }
}

// Функція для обробки зображення та визначення розмірів об'єктів
void calculateDimensions(camera_fb_t* fb) {
    int width = fb->width;
    int height = fb->height;

```

```

std::vector<uint8_t> grayscale(width * height);
std::vector<uint8_t> edges(width * height);

// Перетворення зображення у відтінки сірого
rgbToGrayscale(fb->buf, grayscale.data(), width, height);

// Виявлення країв
edgeDetection(grayscale.data(), edges.data(), width, height);

// Аналіз контурів та визначення розмірів
std::vector<int> dimensions;
for (int y = 0; y < height; y++) {
    for (int x = 0; x < width; x++) {
        if (edges[y * width + x] == 255) {
            // Обробка кожного знайденого краю (це спрощено для прикладу)
            dimensions.push_back(y);
            dimensions.push_back(x);
        }
    }
}

// Визначення максимальних і мінімальних значень для визначення
розмірів
int min_x = *std::min_element(dimensions.begin(), dimensions.end());
int max_x = *std::max_element(dimensions.begin(), dimensions.end());
int min_y = *std::min_element(dimensions.begin(), dimensions.end());
int max_y = *std::max_element(dimensions.begin(), dimensions.end());

int width_dim = max_x - min_x;

```

```

int height_dim = max_y - min_y;
// Відправка електронної пошти з результатами
void sendEmail(String subject, String message) {
    SMTPData smtpData;

    smtpData.setLogin(SMTP_HOST, SMTP_PORT, AUTHOR_EMAIL,
AUTHOR_PASSWORD);
    smtpData.setSender("ESP32-CAM", AUTHOR_EMAIL);
    smtpData.setPriority("High");
    smtpData.setSubject(subject);
    smtpData.setMessage(message, false);
    smtpData.addRecipient(RECIPIENT_EMAIL);

    if (!MailClient.sendMail(smtp, smtpData)) {
        Serial.println("Помилка відправки пошти: " +
MailClient.smtpErrorReason());
    }

    smtpData.empty();
}

// Відправка результатів на пошту
String result = "Width: " + String(width_dim) + " px, Height: " +
String(height_dim) + " px";
    sendEmail(result);
}
void setup() {
    Serial.begin(115200);
}

```

```
initCamera();
connectToWiFi();
}

void loop() {
  camera_fb_t* fb = esp_camera_fb_get();
  if (!fb) {
    Serial.println("Помилка захоплення кадру");
    return;
  }

  // Обробка кадру для визначення розмірів
  calculateDimensions(fb);

  // Звільнення пам'яті
  esp_camera_fb_return(fb);

  // Затримка перед наступним кадром
  delay(1000);
}
```

Ім'я користувача:
Кафедра КІ

ID перевірки:
1016383158

Дата перевірки:
23.06.2024 09:27:42 EEST

Тип перевірки:
Doc vs Internet + Library

Дата звіту:
23.06.2024 09:28:21 EEST

ID користувача:
100005591

Назва документа: Баришніков_Система автоматичного визначення розмірів в процесі токарної обробки дета...

Кількість сторінок: 74 Кількість слів: 10276 Кількість символів: 77614 Розмір файлу: 6.79 MB ID файлу: 1016193419

Виявлено модифікації тексту (можуть впливати на відсоток схожості)

21.1% Схожість

Найбільша схожість: 5.17% з Інтернет-джерелом (<https://uk.wikipedia.org/wiki?curid=225672>)

20% Джерела з Інтернету 286 Сторінка 76

8.83% Джерела з Бібліотеки 135 Сторінка 78

0% Цитат

Не знайдено жодних цитат

Посилання 1 Сторінка 78

0% Вилучень

Немає вилучених джерел

Модифікації

Виявлено модифікації тексту. Детальна інформація доступна в онлайн-звіті.

Замінені символи 5

Підозріле форматування 19 сторінок

Anti-Plagiarism v-15.257

Максимальне співпадіння з одним документом 3.0%

Словники перевірки: en_US, ru_RU, ua_UA. Помилки в документах: 14%

ID: 132224 Назва: БКР Система автоматичного визначення розмірів в процесі токарної обробки деталей з використанням мікроконтролера ESP32-CAM Додано в БД: 2024-06-23 Автора: С.С. Баршніков Керівник: І.О. Засорнова Консультанти: Опоненти:	Документ		Сумарний збіг по Базі Даних	
	Символи	Лексеми	Символи	Лексеми
	64861	542	3692 (6%)	44 (8%)

Джерело плагіату

ID	Опис	Наявність плагіату в документі	
		Символи	Лексеми

РЕЦЕНЗІЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

Дисплемант: Барановіч Сергій Сергійович

Тема: Система автоматичного визначення розмірів в процесі токарної обробки
деталей з використанням мікроконтролера ESP32-CAM

Спеціальність: 123 «Комп'ютерна інженерія»

Обсяг кваліфікаційної роботи:

Кількість листів креслення 3 Кількість сторінок записки 51

1. Короткий зміст роботи та прийнятих рішень: Метою дипломної роботи є забезпечення автоматичного визначення розмірів в процесі токарної обробки деталей з використанням мікроконтролера ESP32-CAM, а також оцінка механізмів обробки інформації у програмно-технічному засобі для забезпечення достатньої точності та ефективності обробки

2. Висновок про відповідність роботи дипломному завданню: Робота повністю відповідає дипломному завданню.

3. Характеристика виконання кожного розділу, ступінь використання останніх досягнень науки і техніки і передових методів роботи: В першому розділі кваліфікаційної роботи було проведено дослідження предметної області остат відомий рішень та було розглянуто загальні відомості про наш мікроконтролер. В другому розділі кваліфікаційної роботи було проведено вибір компонента та проведено порівняння з можливими аналогами, підбрана кванта для нашого мікроконтролера розглянуто загальні відомості про наш мікроконтролер розглянуто можливі подібні до нашій системи. В третьому розділі кваліфікаційної роботи виконано детальну реалізацію нашій системи автоматичного визначення розміру деталей в процесі токарної обробки, а саме: встановлення ArduinoIDE та налаштування всіх необхідних ресурсів для роботи з неореалізація програмного коду для нашій програми та створення усіх необхідних функцій, також було підключено всі компоненти на наш проект було створено базу-даних роботи програми та конкретну схему.

4. Позитивні сторони роботи: висока практична цінність роботи.

5. Негативні сторони роботи: можливо покращити роботу за допомогою інтерфейсу.

6. Оцінка графічного оформлення та пояснювальної записки роботи: Пояснювальна записка оформлена коректно, згідно діючих стандартів оформлення документації.

7. Відгук про роботу в цілому: Робота виконана на належному науково-технічному рівні.

8. Інші зауваження: _____

9. Оцінка дипломної роботи: 3,5 (D) задовільно

Рецензент (прізвище, ім'я, по батькові, посада, місце роботи) _____

Тарасенко Леонід Петрович, зов аапра іпз

ХІУ

24 червня 2024 р.

[Signature] (підпис)

Завідувачу кафедри КПС
д-р.техн.наук, проф. Говорушенко Т. О.

Бараннікова Сергія Сергійовича

ПІІІ здобувача вищої освіти

ФІТ, 4 курсу, групи КІ2-20-1

ЗАЯВА

З правилами чинного Положення «Про систему забезпечення академічної доброчесності у Хмельницькому національному університеті» від 01.07.2022, згідно з яким виявлення плагіату є підставою для відмови в допуску кваліфікаційної роботи до захисту та застосування заходів дисциплінарної та академічної відповідальності, ознайомлений(а). Про використання програмно-технічних засобів для перевірки кваліфікаційних робіт здобувачів вищої освіти на плагіат оповіщений(а) та надаю свою згоду на обробку та збереження університетом моєї роботи в інституційному репозитарії університету.

Також надаю університету право на передачу моєї роботи для обробки та збереження в базах даних програмно-технічних засобів (Unicheck та Anti-Plagiarism) та використання роботи для виявлення плагіату в інших роботах, які перевіряються програмно-технічними засобами та користувачами, що мають доступ до цих програмно-технічних засобів, виключно в обмежених цілях для виявлення плагіату в текстах робіт.

Робота для перевірки університетом надається в друкованому та електронному варіанті. Електронна версія моєї роботи збігається (ідентична) з друкованою.

22 квітня 2024 року

**РІШЕННЯ ЕКСПЕРТНОЇ КОМІСІЇ
КАФЕДРИ КОМП'ЮТЕРНОЇ ІНЖЕНЕРІЇ ТА ІНФОРМАЦІЙНИХ СИСТЕМ
ПРО ДОПУСК КВАЛІФІКАЦІЙНОЇ РОБОТИ ДО ЗАХИСТУ**

Підтверджуємо ознайомлення з результатом звіту подібності щодо роботи, генерованого системою виявлення текстових збігів/ідентичності/схожості:

Назва: Система автоматичного визначення розмірів в процесі токарної обробки деталей з використанням мікроконтролера ESP32-CAM

Автор: Баришніков Сергій Сергійович

Спеціальність: 123- Комп'ютерна інженерія

Освітня програма: освітньо-професійна

Науковий керівник: Засорнова Ірина Олександрівна, к.т.н, доцент

Після аналізу звіту подібності зроблено такий висновок:

№	Висновок	Позначка про відповідність
1	Запозичення, виявлені в роботі, є законними і не є плагіатом. Робота приймається до захисту.	відповідає
2	Виявлені запозичення не є плагіатом, розміщені в розділах, які не описують безпосередньо авторське дослідження, але кількість цитат перевищує обсяг, виправданий поставленою метою роботи. Робота приймається до захисту, але має бути відкоригована. Відкоригований варіант має бути поданий на кафедру за 2 дні до захисту, разом із заявою щодо самостійності виконання письмової роботи та ідентичності друкованої та електронної версії роботи	
3	Виявлені запозичення не є плагіатом, але частково розміщені в розділах, які описують безпосередньо авторське дослідження, а кількість цитат перевищує обсяг, виправданий поставленою метою роботи. В зв'язку з цим мета роботи та поставлені завдання не були досягнені. Робота може бути допущена до захисту (наступного року) після того як буде відкоригована та допрацьована і успішно пройде повторну перевірку на академічний плагіат.	
4	Робота містить наявні текстові спотворення, передбачувані спроби укривтя запозичень або інші прояви академічного плагіату. Робота містить фабрикацію або фальсифікацію даних. Робота не допускається до захисту.	

Підтвердження:

Запозичення, виявлені в роботі, є законними і не є плагіатом, оскільки:

- 1) запозичення розміщені в розділах аналізу існуючих аналогів та прототипів, які не описують безпосередньо авторське дослідження і не стосуються результатів роботи;
- 2) усі запозичення фрагментарні, або мають належним чином оформлені посилання;
- 3) окремі виявлені збіги є загальноживаними фразами або виразами, про що свідчить посилання системи на збіг з 3-4 джерелами на один фрагмент речення;
- 4) всі зафіксовані системою ознаки модифікації тексту відносяться до комбінування латинських символів зі українськомовними скороченнями індексів в формулах, що не є модифікацією тексту.

Сумарний обсяг всіх запозичень, визначений системою виявлення збігів/ідентичності/схожості, складає 21.1% і адресується до 421 першоджерела, що, з урахуванням наведених обґрунтувань, відповідає характеру наукового дослідження і свідчить на користь кваліфікаційної роботи.

Керівник роботи

Гарант ОП

Завідувач кафедри КІС



І.О. Засорнова

С.М. Лисенко

Т. О. Говорущенко