

Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя

Прикладних інформаційних технологій та електроінженерії

(повна назва факультету)

Комп'ютерно-інтегрованих технологій

(повна назва кафедри)

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття освітнього ступеня

бакалавр

(назва освітнього ступеня)

на тему: Розробка автоматизованого програмно-апаратного
комплексу для програмування мікроконтролерів STM32

Виконав(ла): студент(ка) 4 курсу, групи КТЗс-41
спеціальності 151 – Автоматизація та комп'ютерно-
інтегровані технології

(шифр і назва спеціальності)

(підпис) Довжанин В.В.
(прізвище та ініціали)

Керівник _____
(підпис) Стухляк Д.П.
(прізвище та ініціали)

Нормоконтроль _____
(підпис) Левицький В.В.
(прізвище та ініціали)

Завідувач кафедри _____
(підпис) Микитишин А.Г.
(прізвище та ініціали)

Рецензент _____
(підпис) Капаціла Ю.Б.
(прізвище та ініціали)

Тернопіль
2022

Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя

Факультет прикладних інформаційних технологій та електроінженерії
(повна назва факультету)

Кафедра комп'ютерно-інтегрованих технологій
(повна назва кафедри)

ЗАТВЕРДЖУЮ

Завідувач кафедри

Микитишин А.Г.

(підпис)

(прізвище та ініціали)

« »

20__ р.

ЗАВДАННЯ

НА КВАЛІФІКАЦІЙНУ РОБОТУ

на здобуття освітнього ступеня бакалавр
(назва освітнього ступеня)

за спеціальністю 151 Автоматизація та комп'ютерно-інтегровані технології
(шифр і назва спеціальності)

студенту Довжанин Вадим Володимирович
(прізвище, ім'я, по батькові)

1. Тема роботи Розробка автоматизованого програмно-апаратного комплексу для програмування мікроконтролерів STM32

Керівник роботи Стухляк Данило Петрович, к.т.н.
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

Затверджені наказом ректора від « 28 » 01 2022 року № 4/7-58

2. Термін подання студентом завершеної роботи 15.06.2022 р.

3. Вихідні дані до роботи технічна документація

4. Зміст роботи (перелік питань, які потрібно розробити)

1. Аналітична частина 2. Проектна частина 3. Спеціальна частина 4. Безпека життєдіяльності, основи охорони праці

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

1. Актуальність роботи. 2. Мета роботи. 3. Цілі і завдання які необхідно було вирішити.

4. Робототехніка і програмування. 5. TRIK Studio. 6. Мікроконтролери STM32. 7. Варіанти рішення задач. 8. Архітектура. 9. Процес генерації в TRIK Studio. 10. Архітектура генераторів.

11. Генерації коду на мові C++. 12. Компіляція і завантаження програми. 13. Практичне застосування. 14. Висновки.

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Безпека життєдіяльності, основи охорони праці	Гурик О.Я., доцент кафедри МТ		

7. Дата видачі завдання _____

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1	Аналітична частина	29.01-26.02	
2	Проектна частина	27.02-28.03	
3	Спеціальна частина	29.03-01.05	
4	Безпека життєдіяльності, основи охорони праці	02.05-01.06	

Студент

_____ (підпис)

Довжанин В.В.

_____ (прізвище та ініціали)

Керівник роботи

_____ (підпис)

Стухляк Д.П.

_____ (прізвище та ініціали)

АНОТАЦІЯ

Кваліфікаційна робота бакалавра має такі складові: - пояснювальна записки та графічна частини (ілюстративний матеріал – слайди).

Об'єм графічної частини роботи становить 14 слайдів.

Об'єм пояснювальної записки складає 60 друкованих сторінок формату А4 (210×297), об'єм додатків – - друкованих сторінок формату А4.

Робота складається з чотирьох розділів, в яких нараховується 30 рисунків та 4 таблиць з даними.

В роботі використано 36 літературних джерел.

Метою даної роботи є розробка розробка автоматизованого програмно-апаратного комплексу для програмування мікроконтролерів STM32

У даній роботі розроблено програмно-апаратний комплекс для програмування мікроконтролерів ST32. В результаті роботи були отримані такі результати:

- розроблено архітектуру системи програмування мікроконтролера STM32 у середовищі TRIK Studio;
- реалізований модуль, що підключається для мікроконтролера STM32 в TRIK Studio;
- реалізований генератор коду з програми мовою візуальних діаграм у програму мовою C++

Ключові слова: **МІКРОКОНТРОЛЕР, ПРОГРАМУВАННЯ, АРХІТЕКТУРА СИСТЕМИ , КОД ПРОГРАМИ.**

ЗМІСТ

ВСТУП	6
1. АНАЛІТИЧНА ЧАСТИНА	9
1.1. <i>Постановка задачі</i>	9
1.2 <i>Огляд існуючих аналогів</i>	9
1.3 <i>Апаратні елементи підтримки мікроконтролерів STM32</i>	12
1.4 <i>Режими роботи ЦПУ</i>	17
2 ПРОЕКТНА ЧАСТИНА	22
2.1. <i>Мікроконтролер STM32 Discovery</i>	22
2.2. <i>Середовище розробки</i>	26
3 СПЕЦІАЛЬНА ЧАСТИНА	34
3.1 <i>Архітектура системи</i>	33
3.2. <i>Плагін для мікроконтролера STM32 TRIK Studio</i>	36
3.3. <i>Генератор коду на мові C++</i>	36
3.4 <i>Інтеграція компілятора та завантажувача з TRIK Studio</i>	37
3.5 <i>Використання мікроконтролера STM32</i>	39
4. БЕЗПЕКА ЖИТТЄДІЯЛЬНОСТІ, ОСНОВИ ОХОРОНИ ПРАЦІ	44
4.1. <i>Етапи створення безпечних умов праці</i>	44
4.2. <i>Аналіз потенційних небезпек і шкідливостей виробничого середовища</i>	46
4.3. <i>Методи реалізації нормальних умов роботи</i>	51
4.4. <i>Розрахунок заземлення пристрою для заземлення системи контролю температури</i>	54

Висновки..... 57

ПЕРЕЛІК ПОСИЛАНЬ..... 58

ВСТУП

У багатьох сферах діяльності для виконання різноманітних завдань впроваджуються роботи, які здатні замінити людську працю машинною. У зв'язку з цим більш масштабнішим стає розвиток робототехнічної галузі, одним із напрямків діяльності якої є створення програмних засобів, які дадуть можливість користувачам зручно та швидко програмувати різноманітні роботи.

Одним із засобів, здатних спростити та прискорити процес створення програм, є візуальне програмування [24]. Візуальне програмування – це спосіб задавати програму в термінах графічних об'єктів замість звичного для багатьох програмістів застосування абстракції текстового програмування. Цей спосіб дозволяє спростити представлення об'єктів, з якими доводиться оперувати програмісту при розробці програми, що може заощадити час, що витрачається на оновлення програмного продукту, і покращити вміння навчання програмуванню, тим самим розширивши аудиторію розробників програмного забезпечення.

Для управління різноманітними електронними пристроями, зокрема роботами, використовуються певні мікросхеми — мікроконтролери. Мікроконтроллер є обчислювальним пристроєм невеликих розмірів, який містить у собі одне або кілька процесорних ядер, пам'ять, програмовані пристрої введення/виводу і інші периферійних пристроїв.

З недавнього часу в робототехнічній галузі зріс інтерес щодо нової серії мікроконтролерів STM32. Широке поширення дана лінійка набула завдяки своїм перевагам порівняно з аналогами: високої продуктивності, відносно низької вартості та роботі в умовах невеликого електроспоживання. Однокристальні мікроконтролери доступні для вимірювальних приладів, камер, відеокамер, принтерів, Широко використовується в широкому діапазоні галузей, від сканерів і копіювальних пристроїв до електронних розважальних продуктів і різних товарів побутової електроніки.

ARM-процесори використовуються в найрізноманітніших пристроях. Використовуючи архітектуру RISC (архітектуру, яка скорочує набір команд), ARM зміг створити ядра з меншою кількістю транзисторів, які добре вплинули на споживання енергії та ціну. Широке прийняття цієї архітектури пов'язано з тим, що ARM займається тільки дизайном процесора і продає ліцензії сторонніми компаніями на використання технології. При цьому сумісність коду гарантується на рівні інструкції процесора, точніше такий самий же код працює на мікроконтролерах що виробляються багатьма фірмами. Однією з таких компаній є сімейство STM32 ST мікроелектроніки на основі архітектури Cortex-M.

Загалом ARM діє у трьох напрямках:

A ("Application") - Кернел для класичних додатків

R ("Real-time") – ядро для вбудованих систем, що реалізуються в реальному часі.

M ("мікроконтролер") — ядро мікроконтролера.

При проектуванні мікроконтролера потрібно йти на компроміс за розміром і вартістю з одного боку, а з іншого — між гнучкістю і продуктивністю. Оптимальне співвідношення між цими та іншими параметрами сильно варіюється від створення до застосування. Тому архітектура модулів процесора, розмір і тип внутрішньої пам'яті, набір периферійних пристроїв, тип шини відрізняються. На відміну від традиційних комп'ютерних мікропроцесорів, архітектура пам'яті має величезну різноманітність мікроконтролерів, часто використовується для зберігання даних і команд окремо в оперативній пам'яті і ROM.

Мікроконтролери часто містять вбудовані периферійні пристрої, які дозволяють використовувати одну маленьку мікросхему замість великої кількості окремих пристроїв.

На різних навчальних та науково-виробничих відділах проводиться

дослідження платформи QReal [27]. QReal являє собою meta CASE-інструментарій, тобто середовище для створення нових візуальних мов та інтегрованих середовищ програмування для них. Програмування у вигляді окремих об'єктів можна використовувати як засіб управління роботами [22], у зв'язку з чим на основі середовища QReal [15] була створена технологія візуального програмування роботів TRIK Studio [25], яка дозволяє створювати графічні програми для різних роботів та виконувати їх.

У зв'язку з цим з метою збільшення можливостей середовища TRIK Studio було поставлено завдання здійснити підтримку програмування мікроконтролера STM32 у TRIK Studio.

.

1. АНАЛІТИЧНА ЧАСТИНА

1.1 Постановка задачі

Мета даної роботи є розробка автоматизованого програмно-апаратного комплексу для написання програм у мікроконтролерів STM32 у TRIK Studio.

Для реалізації поставленої мети потрібно:

- розробити архітектуру системи програмування мікроконтролера STM32 у середовищі TRIK Studio;
- розробити під'єднання модуля для мікроконтролера STM32 у TRIK Studio;
- реалізувати генератор коду із програми мовою візуальних діаграм у програму мовою C++;
- розробити підсистему інтеграції компілятора та завантажувача програми з TRIK Studio;
- провести апробацію засобу програмування на платі мікро-контролера STM32F4-Discovery

1.2 Огляд існуючих аналогів

Сімейство мікроконтролерів STM32

STM32 - це сімейство 32-бітних мікроконтролерів від компанії STMicroelectronics [21]. Мікроконтролери STM32 згруповані в серії, засновані на ядрах 32-бітних ARM процесорів серії Cortex-M. На рівні ядра

процесора робота з мікроконтролером відбувається переважно з використанням мов C, C++, також використовується мова асемблера.

ARM [1] - Це архітектура процесорів, в основі якої міститься концепція RISC (Reduced Instruction Set Computer). Дана стратегія організації обчислювальної системи передбачає використання спрощеного набору інструкцій, тобто кожна окрема операція процесора виконує фіксований невеликий обсяг зробленої роботи, для прикладу, може бути читання даних з пам'яті в регістр, запис даних з реєстру в пам'ять, операції над даними, що містяться в регістрах (арифметичні, логічні тощо) та інші. Як правило, інструкції RISC-процесорів мають фіксовану довжину і простий формат кодування, що спрощує вибірку та декодування команди. Спрощення набору команд дозволяє збільшити продуктивність системи.

Cortex-M є серією 32-бітових ядер ARM процесорів, які в основному використовуються для роботи в мікроконтролерах. До серії входять ядра Cortex-M0, Cortex-M0+, Cortex-M1, Cortex-M3, Cortex-M7(F), Cortex-M23, Cortex-M33(F). Приставка «F» вказує на наявність модуля операцій з змінною дробовою частиною (FPU — Floating Point Unit), однак у деяких комплектаціях його може не бути.

Ядра Cortex-M0, Cortex M0+ та Cortex-M23 відрізняють невеликий розмір та відносно низька ціна. Ядра Cortex-M3, Cortex-M4 та Cortex-M33 підтримують оптимальний баланс між виробником.

ністю та низькими енерговитратами. Ядро Cortex-M7 призначене для забезпечення функціонування систем, що потребують високої продуктивності.

Ядра ARM Cortex-M4, Cortex-M7 та Cortex-M33 підтримують обробку цифрових сигналів та операції зі змінною крапкою, що дають більшу продуктивність та енергоефективність обчислень на цих процесорах.

На рисунку 1.1 представлено зображення плати мікроконтролера STM32F429I-Discovery [20], яка використовувалася у межах даної роботи для апробації. Плата містить такі компоненти:

- мікроконтролер STM32F429ZIT6 з ядром Cortex-M4F з 2 Мб флеш-збереження даних, 256 Кб тимчасової пам'яті (RAM) у

корпусі LQFP144;

- програматор ST-LINK/V2;
- живлення плати через USB або від другого джерела 3V чи 5V;
- рідкокристалічний дисплей 2.4”;
- 64 Мб SRAM;
- датчик руху ST-MEMS L3GD20 та цифровий вихід триосьового гіроскопа;
- шість світлодіодів:
 - LD1 (червоний/зелений) для підключення USB;
 - LD2 (червоний) для живлення 3.3 V;
 - два користувальницьких світлодіода: LD3 (зелений) і LD4 (червоний);
 - два світлодіоди для USB OTG¹: LD5 (зелений) та LD6 (червоний).
- дві кнопки: користувальницька та кнопка перезапуску;
- USB OTG із роз'ємом micro-AB.



Рис. 1.1 Плата мікроконтролера STM32F429I-Discovery.

1.3 Апаратні елементи підтримки мікроконтролерів STM32

Мікроконтролер — мікросхема, призначена для управління різноманітним електронним пристроєм. Мікроконтролер реалізує як процесорні, так і периферійні функції і включає в себе постійний блок зберігання (ROM) і оперативну пам'ять. Це комп'ютер, який може виконувати найпростіші завдання.

Найпоширеніші периферійні пристрої:

- Вбудований генератор годинників та таймер сторожової собаки.
- Аналогові, цифрові (кодування інформації), та перетворювачі аналогово-цифрові;
- Таймер і широкий імпульсний модулятор;

- Універсальний цифровий порт вводу-виводу;
- Радіочастотний приймач і передавач;
- Аррай внутрішньої флеш-пам'яті;
- Контролер дисплея та клавіатура;
- Послідовний інтерфейс;
- Контролер двигуна;
- Порівняння;
- контролери двигунів;
- компаратори;

Назва мікроконтролера вражає, тому говорять про сім'ю, яка має специфічну архітектуру і атрибути, притаманні їй. Найпопулярніші на сьогодні родини — PIC (Microchip), MCS51, AVR (Atmel), Cortex, ADSP21XX, ARM7 .. 9, 15.

ARMHoldings розробила ряд мікроконтролерів, заснованих на моделі ARM. Цей мікроконтролер призначений для різноманітних програм.

Одним з їх новітніх продуктів є плеяда процесорів Cortex: Cortex-A (додатки), Cortex-R (в реальному часі), Cortex-M (мікроконтролери)

Ядра Cortex-A і Cortex-R можуть бути використані з високопродуктивними технологіями (ноутбуками, планшетами тощо). Те ж саме стосується і галузей промисловості (медичної, автомобільної, безпекової тощо). Мікроконтролери на основі цих ядер дорогі і не підходять для малих і середніх проектів.

Cortex-M розроблений спеціально для цих проектів. Що говорить про використання у різного розміру проектах, а також для програм що розроблені більш раніше (Cortex-A і Cortex-R).

Сімейство Cortex-M має такі ядра: Cortex-M0...M7, рис. 2.1 – 2.3. Сімейство Cortex-M має перспективи для подальшого використання [12].

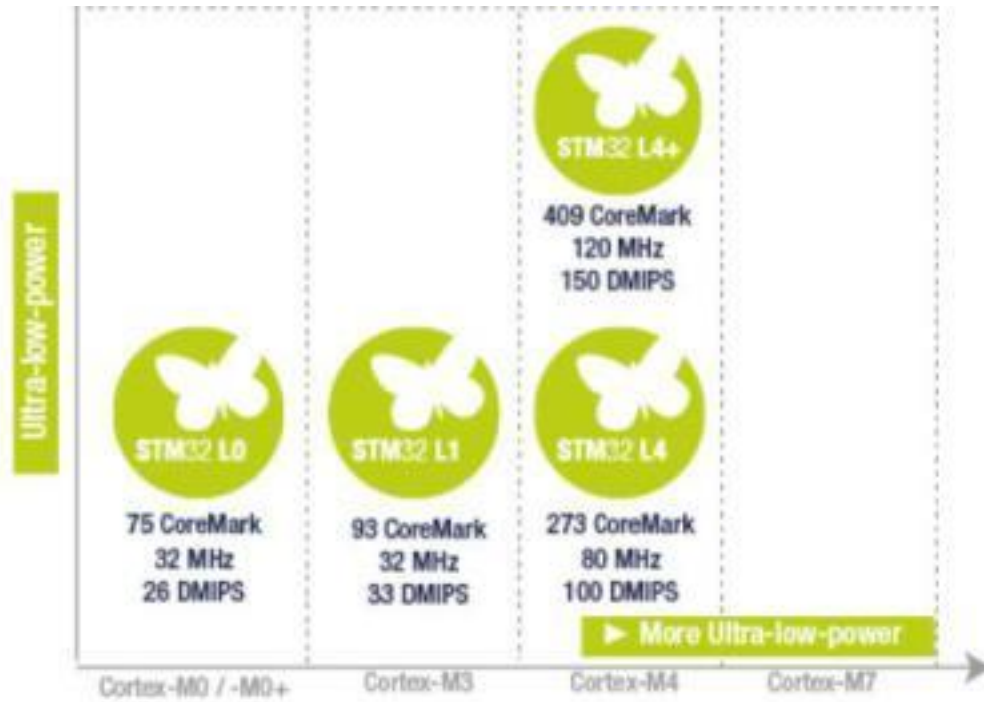


Рисунок 1.2 – Мікроконтролери типу «Ultra low power»

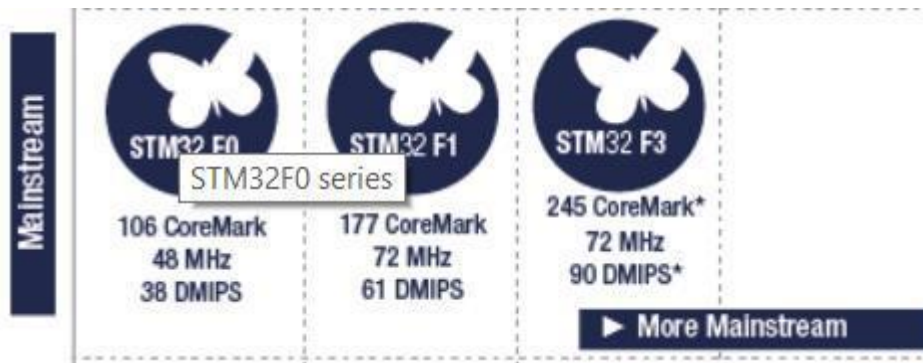


Рисунок 1.3 – Мікроконтролери типу «Mainstream»

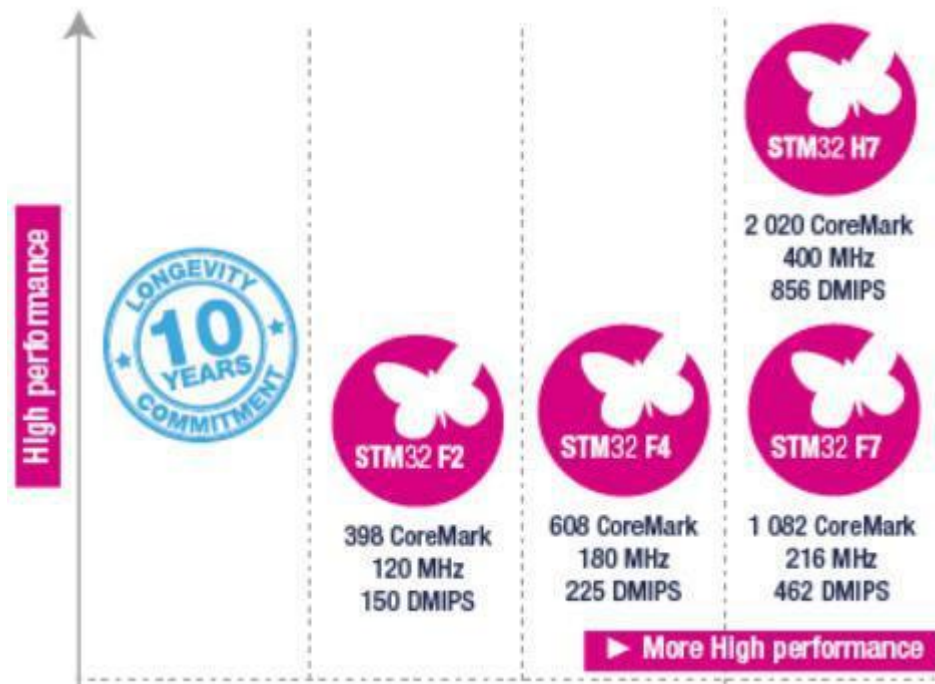


Рисунок 1.4 – Мікроконтролери типу «High performance»

Найкраща цінність для грошей сьогодні — 32-бітна компанія мікроконтролера STM32F4 серії STMicroelectronics. Крім написання продуктивності, до складу STM32F4 входить великий набір периферійних пристроїв.

Мікроконтролер STM32F4 в даний час є лідером ринку по продуктивності мікроконтролера Cortex-M. Основними параметрами лінійки STM32F4 є:

- 32 - кортикальне - M4F бітове ядро, DSP інструкції і FPU модуль;
- Тактова швидкість 168 МГц, 210 DMIP при виконанні коду зі спалахом;
- До 1 МБ флеш-пам'яті;
- 174 КБ SRAM + 58 КБ SRAM, присвяченого ядру;
- 16 каналів DMA, підтримка FIFO, подвійна буферизація;
- Контролер зовнішньої пам'яті та РК-дисплей, FSMC;
- Контролер карт пам'яті SDIO;
- 4 Кб ОП та резервної енергетичної зони;
- 0 рік. Повномасштабний РТК з 5-секундними інкрементами;
- До 17 32/16-розрядних таймерів, керування двигуном

- До трьох 12-бітних АЦП, 24 каналів, 24 μ с, 7 в потрібному режимі 2 μ с і 2 12-бітні АЦП;
- USB / fs OTG ;
- 10/100 MAC Ethernet;
- До 6 USART та додатково 3 x SPI (з послідовним інтерфейсом I2S);
- інтерфейс “CMOS” 8/10/12/14
- кодування AES128 .. 256, // Hash (MD5, SHA-1);
- Рандом-генератор чисел.

STMicroelectronics надає STM32 Discovery Debugging Kit з серії зневадження Discovery, щоб швидко розпочати роботу з STM32.

STM32F4 Discovery включає акселерометр і цифровий мікрофон, але без цільового контролера і вмістних програм ST-link. Ці заходи деох позитивно впливають на розповсюдження мікроконтролерів серії STM32F4.

ЦПУ Cortex

Основою процесора Cortex є 32 – х, RISC. ЦП задією простіший метод програмування ARM7 і ARM 9. Існує більший набір команд, який підтримує цілочисельну арифметику та покращену роботу бітів. Більш жорсткі тимчасові характеристики реального світу.

Конвеєр.

CPU Cortex може виконувати команди за один цикл. Також ARM7 та ARM9. Це було досягнуто за допомогою триступінчастим конвеєром.

Процесори Cortex-M3, такі як ARM7/ARM9, використовують триступеневий трубопровід, але Cortex-M3 підтримує прогнозування переходу для мінімізації конвеєрних перезавантажень.

Одна функція розшифровує наступну функцію, і читається третя, рис.1 5.

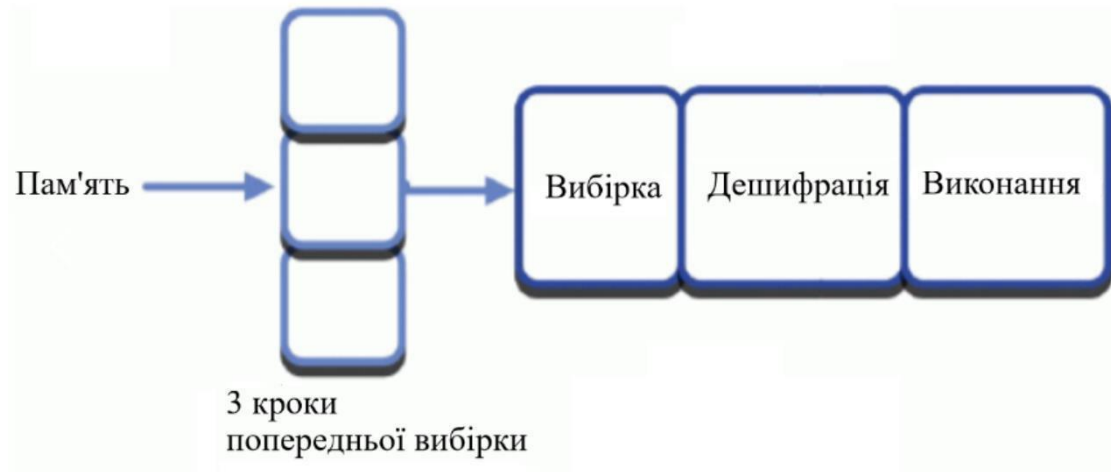


Рисунок 1.5 – Конвеєр

Цей механізм працює добре, але тільки з лінійними шнурами. Якщо потрібний перехід, то трубопровід очищають і перезапускають.

У ARM7 та ARM9 цей перехід обмежує продуктивність. CPU Cortex містить логіку, що передбачає переходи. Іншими словами, при досягнутому умовному переході виконується попередня вибірка, і в результаті для виконання будуть доступні дві команди для умовного переходу. Результат не є втратою продуктивності.

Тому конвеєр є інструментом, який є залежним від продуктивності і не вимагає ніяких дій в частині програмного коду [13].

1.4 Режими роботи ЦПУ

Незважаючи на те, що процесори Cortex працюють швидко, легко використовуються, і були розроблені з меншою кількістю логічних елементів. В основі мікроконтролера розглядається підтримка DSO.

Cortex володіє двома режимами: Режим потоків (або режим потокової передачі) і режим процесора (або режим процесора).

ЦП запускається в режимі потоку як безперервна фонові інструкція і переходить в режим обробника при обробці винятків.

CPU Cortex також може запускати програмний код в привілейованих або несанкціонованих режимах. У привілейованому режимі ЦП має доступ до більшості вибору інструкцій, а в несанкціонованому режимі деякі команди відключені. Цей режим також вимикає доступ до багатьох регістрів керування системою процесорів Cortex. Крім того, ви можете налагодити задіювання стека. Головний стек доступний як у режимах thread, так і handler. Крім того, режим відпрацювання можна настроїти на використання стека процесів.



Рисунок 1.6 – Режимы работы ЦПУ

Процесори Cortex-M3 доступні в звичайному режимі ('flat'), а також підтримують операційні системи реального часу.

Відразу після скидання процесор Cortex починає в «плоскій» конфігурації.

- Режим «як нитка, так і обробник» мають повні обмеження доступу до ресурсів процесора через те, що інструкції виконуються в привілейованому режимі.

Зник, режим ниті та обробника використовують основний стек. Для початку виконання команд можна вказати напрямок скидання і початкову адресу стека процесору для виконання коду C програми. Однак, якщо використовувати DSO або розробляти програми, що критикують безпеку, то можна використовувати процесор у складнішій конфігурації.

При даній реалізації, великий код програми можна розділити на системний та прикладний, Коли виникають помилки, то в прикладному кодї не запуститься збїй ОСРЧ.

Карта пам'ятї

Процесор Cortex-M3 є стандартизованим ядром мікроконтролера, тому карта пам'ятї явно реалізована. Коли будемо використовувати незначну кількість внутрішніх шин, отримаємо адресний простір одинарний, який має об'єм 4 Гб. Перша 1 Гб пам'ятї рівномірно розподілена по двох частинах, а саме кодом програми і частиною статичної оперативної пам'ятї. Простір програмного коду оптимізовано для роботи шини I-Code. Аналогічно, простір в статичній оперативній пам'ятї досягається через шину Dcode. Якщо не звартати увагу, що використовується завантаження і запуск команд в статичній області оперативної пам'ятї, дискретизація виконується областю системної шини, яке потребує додаткового режиму очікування.

Тому запуск програмного коду в статичній оперативній пам'ятї йде більш повільніше, ніж вбудована флеш-пам'ять в області коду програми. Наступного 0 року. 5 Гб пам'ятї — внутрішня область UVV. Ця область містить все, що надає користувачеві мікроконтролер UVV виробника. Перші 1 МБ статичної оперативної пам'ятї і регінів UVV обробляються в бітах. Для цього використайте спосіб bit banding. Отже, всі дані, що розміщуються в цих частинах, можуть бути оброблені як вербально, так і бітовим шляхом. До зовнішньої статичної оперативної пам'ятї та UVV виділено такі 2 ГБ адресного простору: Останній 0 рік. 5 Гб зарезервовано для ресурсів системи процесорів Cortex і розширення процесорів майбутнього Cortex. Всі регістри процесора Cortex є фіксованими. Це полегшує портування застосунків між більшістю мікроконтролерами STM32.

Для процесорів Cortex-M3 встановлена конкретна частина пам'яті на 4 Гб, що має специфічні ділянки, виділені для зберігання програмних кодів, статичної оперативної пам'яті, елементів вводу-виводу, не внутрішньою пам'яті і пристроїв, ще додатково і внутрішніх регістрів Cortex. Отриманий рисунок розміщення пам'яті однаковий для всіх Cortex рис.1.7.



Рисунок 1.7 – Карта пам'яті

Доступ до фрагментованих даних.

Набори команд ARM7 та ARM9 мають доступ до символічних змінних, таких як байти, півслова та слова. Це дозволяє ЦП підтримувати цілочисельні змінні без бібліотеки програмного забезпечення, як у випадку 8-бітних і 16-бітних мікроконтролерів. Але старий ЦП ARM має один недолік: Він також це додатково має можливість працювати тільки з нефрагментарними словами. В результаті, компілятор LINER розміщує дані неефективно в статичній оперативній пам'яті, а деякі з них втрачаються. (В залежності від комбінації змінних, що використовуються, втрата може досягати 25%.), рис.1 8.

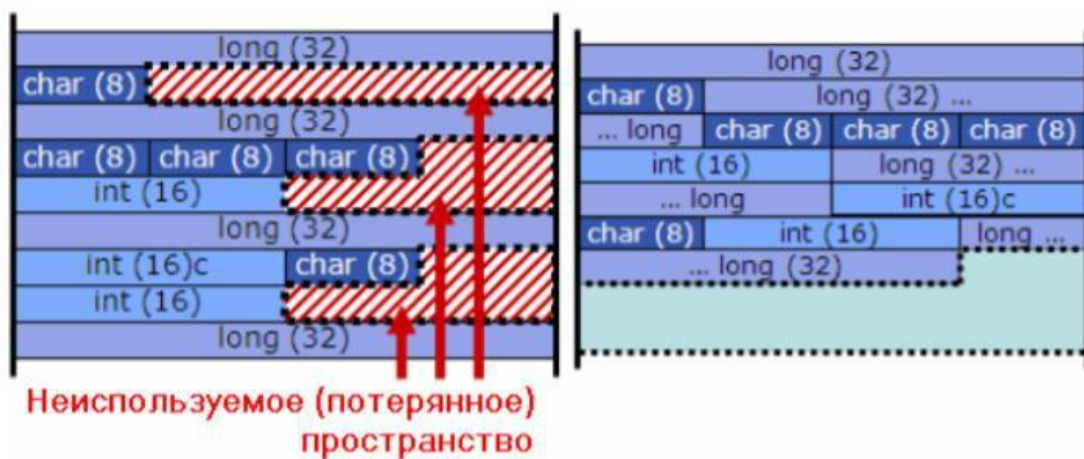


Рисунок 1.8 – Доступ до фрагментованих даних

CPU Cortex має можливість зберігати метод адресації слів, напівслів і байтів, і дозволяє фрагментований доступ до даних. Це дає компілятору liner повну свободу при розміщенні даних в змінній пам'яті. Крім того, функція прив'язування бітів Cortex, що підтримується CPU, дозволяє групувати програмні прапорці в слова або змінні типу напівслова замість використання окремих байтів для кожного прапорця.

2 ПРОЕКТНА ЧАСТИНА

2.1 Мікроконтролер STM32 Discovery.

Комплекти STM32 Discovery виробництва STMicroelectronics. Комплект розрахований на дослідження і розробку 32 мікроконтролерів в популярному сімействі ARM Cortex-M3. Бажано використовувати лінійку Arm Cortex M3 на основі мікроконтролера STM32F100RBT6B, в результаті чого доступний набір функцій STMicroelectronics A. Універсальний вихід цього борту і всі виходи силового ланцюга і скидання знаходяться по окружності 2 Керується штифтом (62) з кроком 54мм. Це дозволяє вбудовувати схему в прототип пристрою, рис. 2.1.

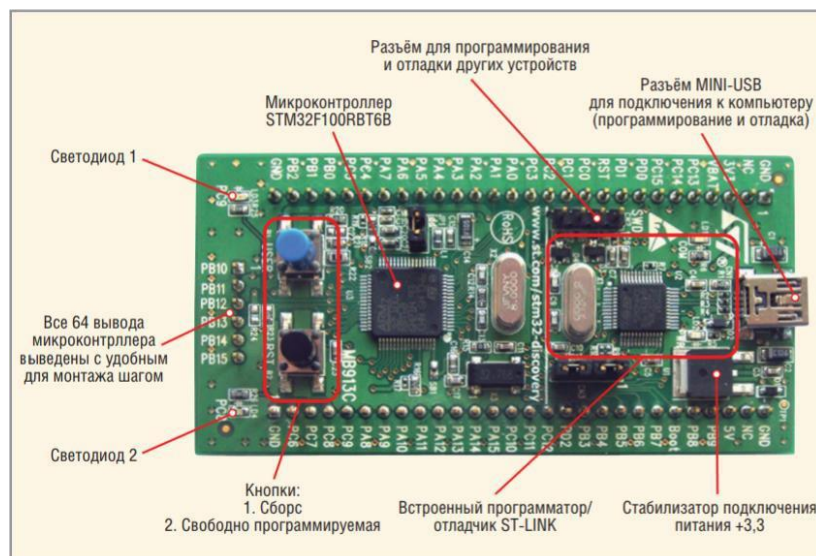


Рисунок 2.1 – Зовнішній вигляд Discovery

Крім мікроконтролера, на платі є програмна програма - debugger ST link (рис.2.2). Це дозволяє почати оновлення свого першого застосування без придбання додаткових пристроїв для програмування та переналагодження. Для початку потрібно, це персональний комп'ютер і USB-кабель (MicroUSB).

Великим плюсом є те, що програми для налаштування можна використовувати як незалежні пристрої для програмування та налагодження інших пристроїв на основі

мікроконтролера STM32. Для цього на платі встановлено чотириконтактний SWD-роз'єм і джампер для вибору якого мікроконтролера регулювати (встановлюється зовні або на платі) [14].

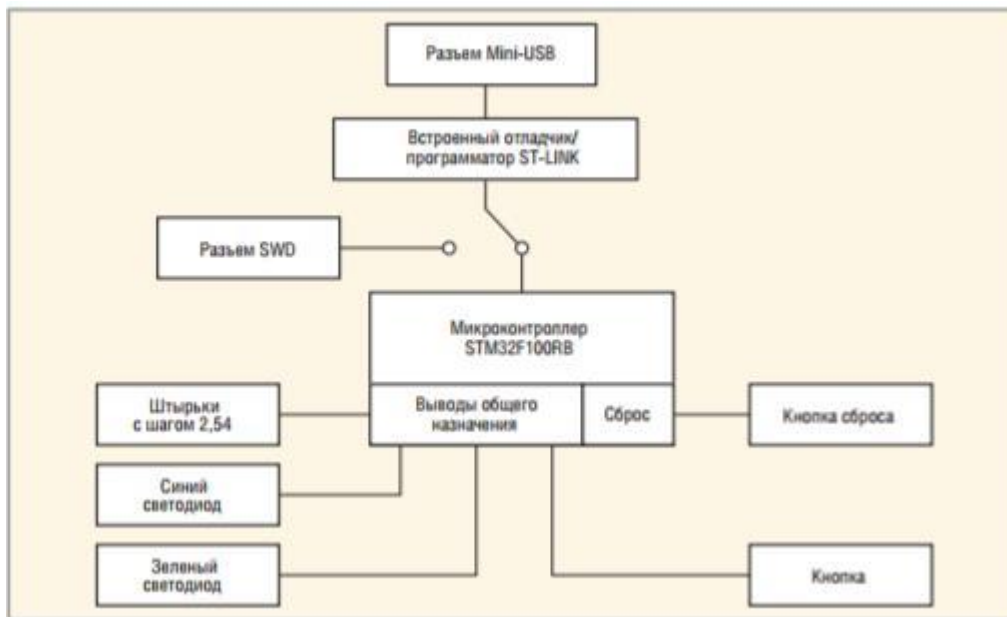


Рисунок 2.2 – Функціональна схема плати Discovery

Open746I-C Standart на основі STM32F746ICT6

Відлагоджувач Open746I-C Standart заснований на мікроконтролері ARM Cortex-M7 STM32F746ICT6 і має різноманітну периферію [15].

Продукт структурований материнською платою і мікроконтролером core746i. Дошки для зневадження підтримують широкий спектр розширень функцій з дочірніми платами для спеціалізованих програм. Завдяки модульності та відкритому дизайну ця дошка для налагодження ідеально підходить для початку розробки застосунків на мікроконтролерах лінійки використання STM32.

Материнська плата

Материнська плата являє собою надруковану карту з роз'ємами для зовнішніх адаптерів і деяких пасивних компонентів, Рисунок 2.3.

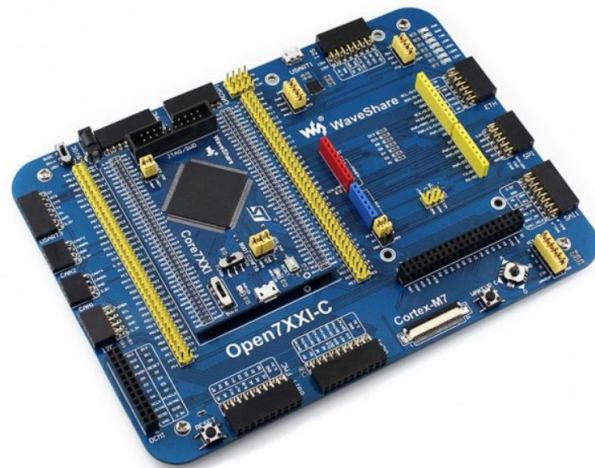


Рисунок 2.3 – Плата для налагодження Open746I-C Standart

Плата мікроконтролера

Фізичне розміщення компонентів мікроконтролера є обох боків.

Рисунок — рис. 2.4

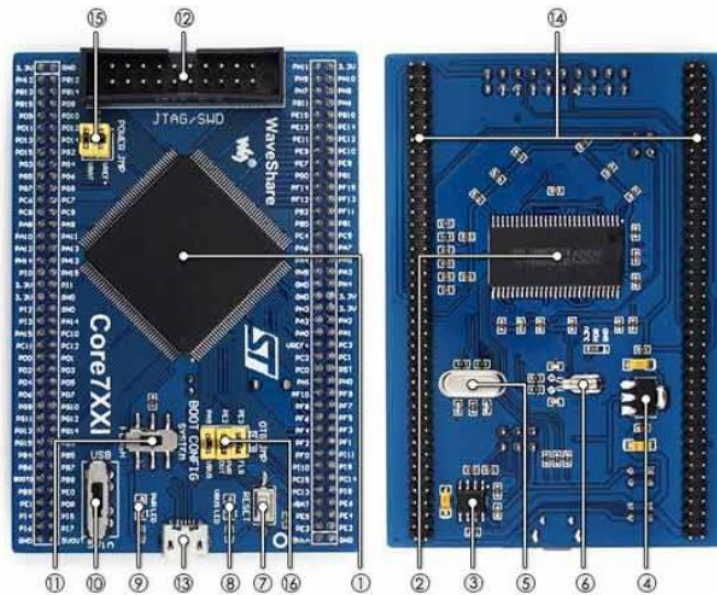


Рисунок 2.4 – Плата мікроконтролера

Maple mini

Arduino відрізняється оригінальністю, легкістю розробки, доступністю обладнання, практичним застосуванням у багатьох випадках. Ця вартість налагодження протягом свого існування була величезною перевагою в плані навчання.

У зв'язку з розробкою і високооб'ємним виробництвом дешевих плат для налаштування Maple mini, існує ще один спосіб швидко розробити мікроконтролери в сімействі STM32.

Модель Maple Mini базується на мікроконтролері STM32F103. Плата має інтерфейс USB, вбудований в мікроконтролер і не вимагає пристрою під'єднання USB-to-UART для з'єднання із комп'ютером, рис. 2.5.

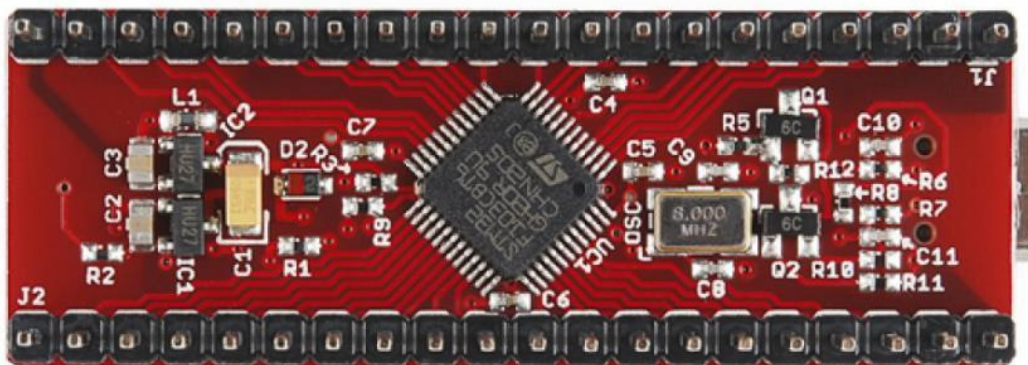


Рисунок 2.5 – Плата для настройки Maple mini

Будова мікроконтролера STM32 робить його доступним для широкого кола галузей промисловості. Через те, що мікроконтролери мають різні ядерні моделі, вони ідеально підходять для вирішення як простих проблем, так і складніших промислових розмірів. Є багато програмістів, які можуть зневажувати і налаштовувати мікроконтролер. Також можна за допомогою програміста протестувати мікроконтролер відповідно до своїх вимог.

2.2. Середовище розробки

Утворення дій по порівняльному аналізу існуючих засобів, що надають можливість створювати програми для мікроконтролерів STM32, з метою розглянути вже досягнуті результати в галузі програмування плат серії STM32, дає можливість знайти способи організації таких інструментаріїв та обґрунтувати доцільність розробки засобів підтримки STM32 у TRIK Studio .

Як критеріїв для порівняння було обрано: по-перше, вільне поширення продукту, т.к. у рамках дослідницької діяльності бажано працювати з інструментами, що передбачають відкрите та вільне використання програмного забезпечення; по-друге, кросплатформність засобу, т.к. для універсального використання необхідно мати інструментарій, доступний на багатьох операційних системах, а саме як Windows, Linux, MacOS; і, по-третє, підтримка візуальних середовищ для програмування та створення програм, т.к. є бажання застосувати методологію об'єктно-орієнтованого програмування до створення програм для мікроконтролера STM32.

IAR Embedded Workbench for ARM

IAR Embedded Workbench for ARM [11] являє собою багатофункціональне середовище розробки додатків для цілого ряду мікроконтролерів під керуванням процесорів ARM. Розробка програми може вестись користувачем мовами C, C++ та мовою асемблера.

Середовище включає наступні інструменти: компілятор C/C++, бібліотеки C/C++, транслятор асемблера, лінковник, текстовий редактор, відладчик у кодах C, C++ та асемблера, менеджер проектів.

Ця програма працює лише на пристроях, що використовують операційну систему Windows та є комерційним продуктом.

Keil MDK

Keil MDK (Microcontroller Development Kit) [12] — інструментарій для реалізації програмного забезпечення для широкого спектру мікроконтролерів на основі процесорів ARM Cortex-M. MDK включає середовище розробки μ Vision, C/C++ компілятор для архітектури ARM, відладчик та інші компоненти, необхідні розробки, які можна побачити на малюнку 2.6

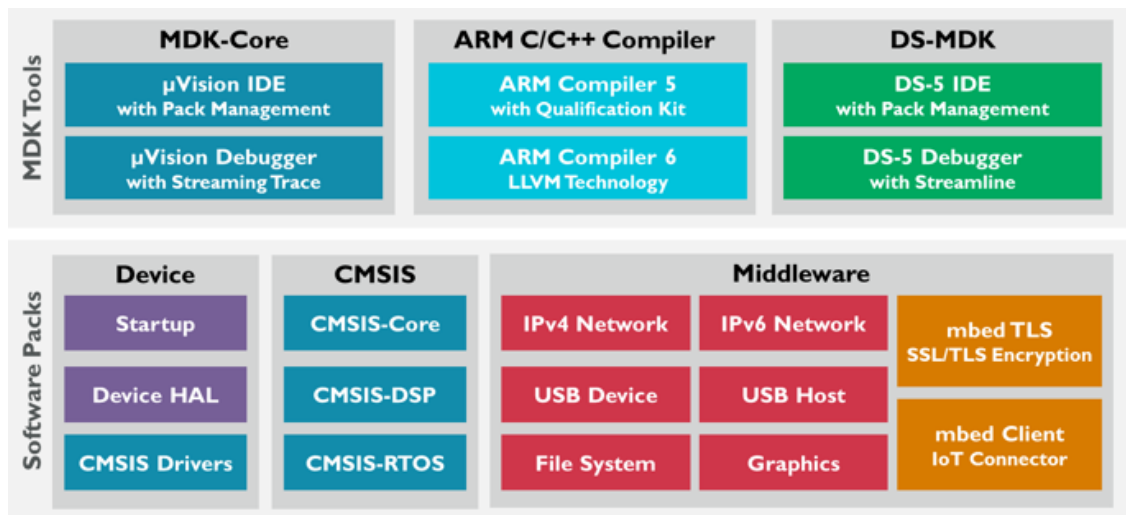


Рис. 2.6 Компоненти Keil Microcontroller Development Kit. Зображення взято з джерела [13].

Важливо відзначити, що середовище розробки Keil μ Vision можна використовувати лише на пристроях, що використовують операційну систему Windows. На додаток до цього засіб програмування є комерційним продуктом.

Eclipse ARM plugin Eclipse [6] — це кросплатформове вбудоване середовище розробки, яке дозволяє розробляти додатки різними мовами програмування, таких як Java, C, C++, JavaScript, Python та інші. Таке різноманіття підтримуваних можливостей досягається завдяки системі модулів (плагінів), що підключаються, які дозволяють налаштувати роботу середовища залежно від своїх потреб. Плагін GNU ARM Eclipse [8] дозволяє в середовищі програмування Eclipse здійснювати розробку та налагодження програм для процесорів ARM мовами C, C++. Плагін, як і середовище Eclipse, знаходиться у відкритому доступі та надає право на вільне розповсюдження та

використання. Зображення вікна Eclipse з прикладом проекту для мікроконтролера STM32F4 можна проаналізувати на рисунку 2.7

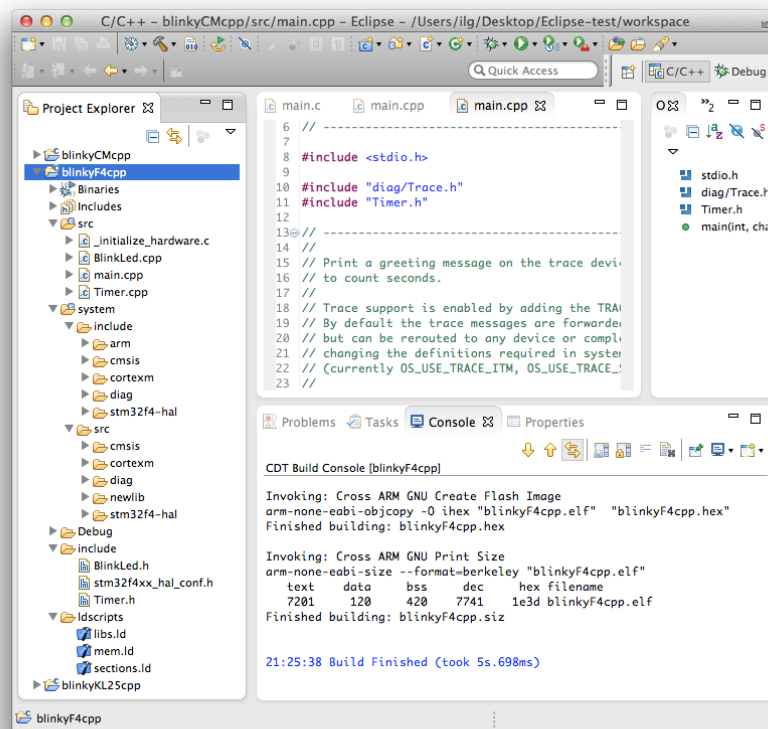


Рис. 2.7 Вікно Eclipse з прикладом проекту для мікроконтролера STM32F4. Зображення взято з джерела [7].

Atollic TrueStudio

Atollic TrueStudio [2] являє собою кросплатформенну інте-середовище розробки програмного забезпечення для процесорів ARM [31-34]. Засіб розробки створено з урахуванням платформи Eclipse, розглянутої раніше. Інструментарій включає редактор коду з підтримкою мов C, C++ і асемблера, C/C++ компілятор, транслятор асемблера, лінковник, текстовий редактор коду, налагоджувач, систему складання вихідного коду, менеджер проектів; підтримує такі системи контролю версій як Git, SVN, CVS.

Програма Atollic TrueStudio є комерційною. Є безкоштовна версія продукту, проте вона має низку обмежень, що мають повноцінну розробку програмного забезпечення.

CooCox

CooCox [4] — інструментарій для розробки програмного забезпечення для процесорів ARM Cortex-M [29,30]. До складу засобу програмування входять: адаптер налагодження CoLinkEx, інтегроване середовище розробки CoIDE, графічний інструмент генерації коду CoSmart. Інструментарій CooCox передбачає вільне поширення та використання.

CooCox CoIDE — середовище розробки, що базується на інструменті Eclipse та компіляторі GCC (інструментарій GCC-ARM-Embedded).

Цей засіб підтримує лише операційну систему Windows для розробки програмного забезпечення.

На основі проведеного огляду можна зробити висновок, що всі розглянуті середовища програмування мають ряд переваг та недоліків. Результати порівняльного аналізу можна побачити в таблиці на рисунку 2.8. При цьому всі вищевикладені засоби передбачають створення програм текстовою мовою програмування, і жоден із наведених інструментаріїв не надає можливості створювати програми візуальною мовою. Тому доцільно здійснити підтримку програмування мікроконтролерів STM32 у TRIK Studio.

Критерій порівняння	Свободное распространение	Кроссплатформенность	Поддержка визуальных языков
Технология			
IAR Embedded Workbench for ARM	—	—	—
Keil MDK	—	—	—
Eclipse ARM plugin	+	+	—
Atollic TrueStudio	—	+	—
CooCox	+	—	—

Рис. 2.8 Порівняльна таблиця засобів програмування мікроконтролера STM32.

TRIK Studio

Загальна архітектура

TRIK Studio [5] – це територія для програмування роботів, яке дозволяє створювати графічні програми для роботів Lego Mindstorms NXT, та її модифікації, а саме Lego Mindstorms EV3, TRIK. Створені програми є можливість виконувати прямо на комп'ютері, посилаючи команди роботу через Wi-Fi, Bluetooth або USB-інтерфейс, можна виконувати на двовірній моделі робота, яка симулює поведінку реальної роботи в процесі виконання програми, а також можна генерувати по графічним діаграмам код різними стрічковими мовами програмування та надсилати його для виконання на роботі. До складу архітектури TRIK Studio входять такі модулі, як ядро технології QReal, ядро системи TRIK Studio, набори моделей роботів для підтримуваних середовищем робототехнічних конструкторів, інтерпретатори графічних програм, генератори вихідного коду програм текстовими мовами програмування за графічними діаграмами, а також засоби комунікації з роботами та інші компоненти. Загальну архітектуру програми TRIK Studio можна побачити малюнку 2.9.



Рис. 2.9 Спільна архітектура TRIK Studio.

Генератори

Процес генерації коду текстовою мовою програмування на основі графічної програми в TRIK Studio відбувається так: спочатку по візуальній діаграмі, побудованій користувачем, будується граф потоку управління програми, який шляхом обходу перевіряється на коректність. У разі синтаксичної коректності програми будується семантичне дерево програми. Для кожного текстового мови програмування, що підтримується, заданий набір шаблонів генерації, які дозволяють сконструювати код програми в процесі обходу семантичного дерева. Архітектура підсистеми генерації коду в TRIK Studio представлена малюнку 2.10.



Рис. 2.10 Архітектура підсистеми генерації коду у TRIK Studio.

Всі генератори коду програми TRIK Studio оформлені у вигляді модулів, що підключаються. Основним класом для всіх генераторів є Robots Generator PluginBase. Від нього успадковуються всі конкретні генератори для кожної з підтримуваних платформ роботів і для кожного мови програмування, що підтримується програмою TRIK Studio. Ієрархію успадкування та приклад генераторів коду програми TRIK Studio можна побачити на малюнку 2.11.

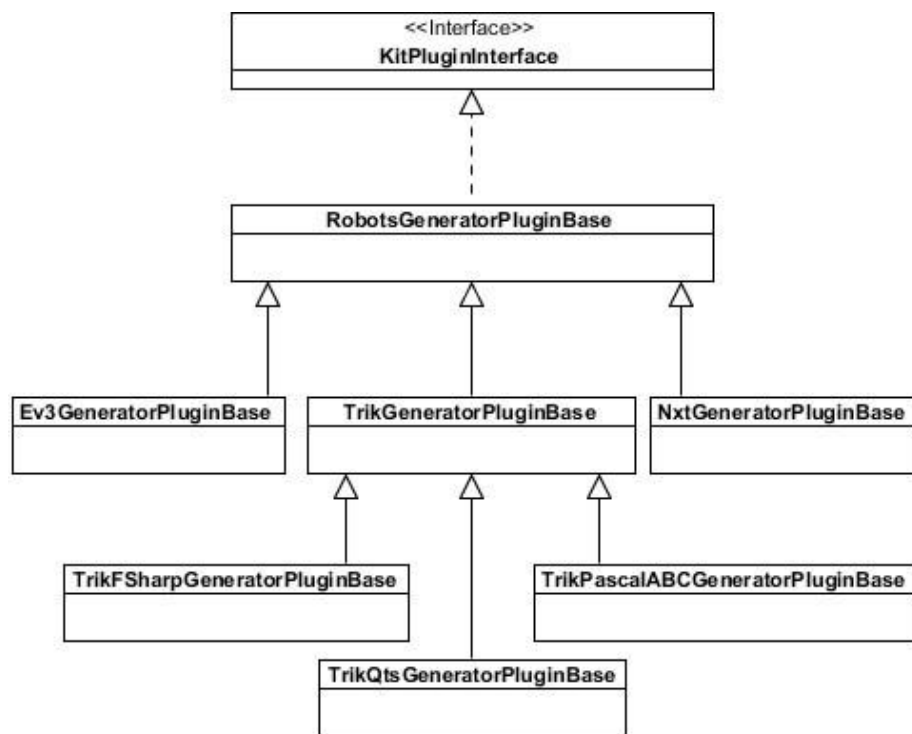


Рис. 2.11 Генераторы коду у TRIK Studio.

3 СПЕЦІАЛЬНА ЧАСТИНА

3.1 Архітектура системи

Розроблена у процесі цієї роботи система і двох модулів. Перший модуль міститься всередині середовища TRIK Studio і є підсистемою отримання на основі графічної діаграми, створеної користувачем, коду програми мовою C++. Генерація коду запускається при натисканні користувачем відповідної кнопки у меню програми TRIK Studio. У вигляді схеми зображення даного модуля є можливість побачити на рисунку 3.1.

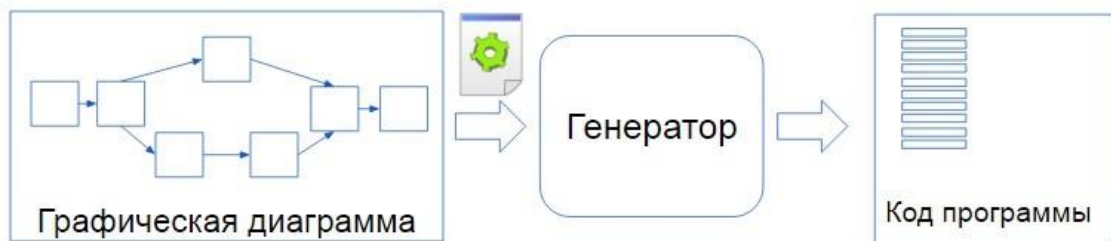


Рис. 3.1 Схема генерації коду мовою C++ за графічною діаграмою.

Для підтримки нової моделі мікроконтролерів було розширено архітектуру середовища програмування TRIK Studio, а саме було додано модель роботи STM32 і було реалізовано генератор коду мовою C++. Загальну архітектуру програми TRIK Studio з модулем програмування мікроконтролера STM32 можна побачити на рис.

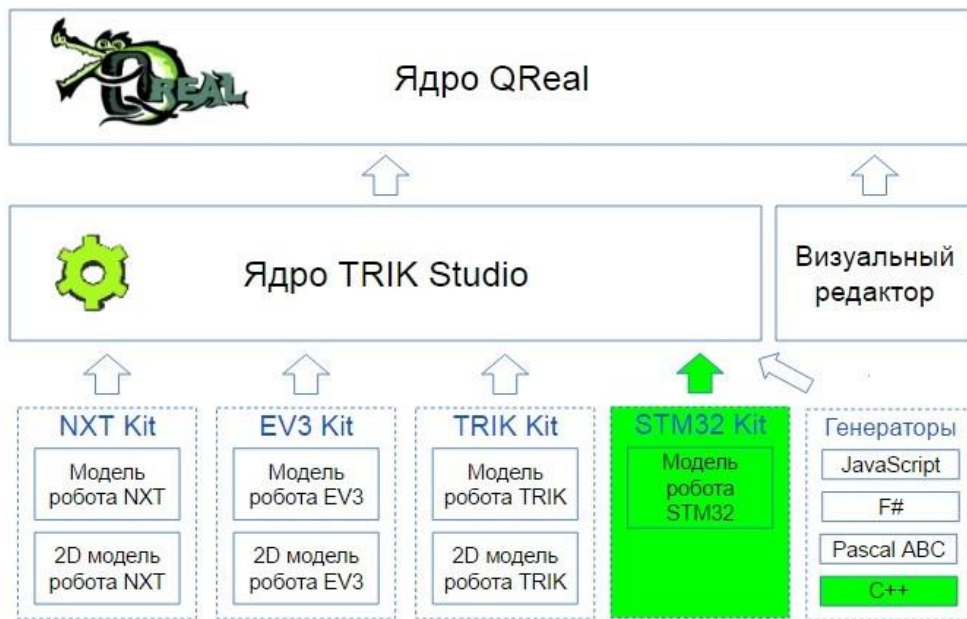


Рис. 3.2 Архітектура TRIK Studio з модулем програмування мікроконтролера STM32.

Для здійснення генерації коду мовою програмування C++ було розширено архітектуру підсистеми генерації коду TRIK Studio, а саме були реалізовані шаблони блоків, які використовуються в процесі конструювання коду текстовою мовою програмування. Архітектура підсистеми генерації коду в TRIK Studio з модулем генерації C++ коду представлена на рисунку 3.3. Зеленим кольором відзначені компоненти, додані в процесі виконання даної роботи.

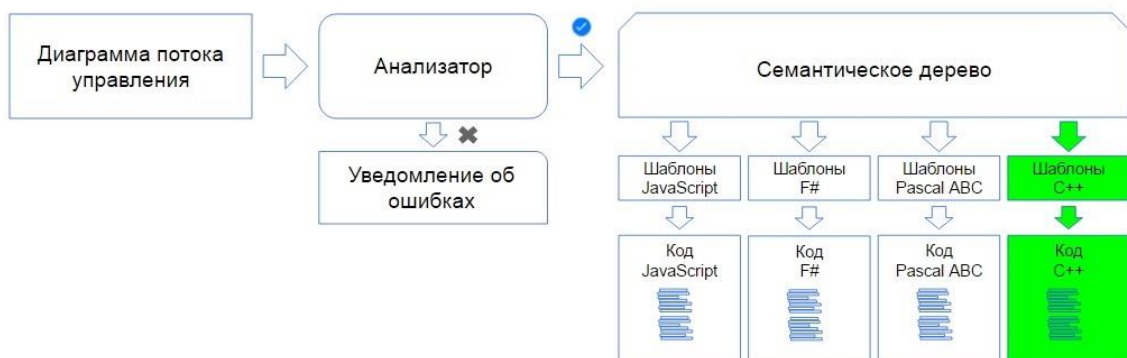


Рис. 3.3 Архітектура підсистеми генерації коду в TRIK Studio з модулем генерації C++ коду.

Ієрархію класів генераторів коду за візуальними діаграмами в TRIK Studio можна проаналізувати на рисунку 3.4. Зеленим кольором зафарбовані блоки, додані в архітектуру середовища програмування роботів TRIK Studio у процесі роботи над системою.

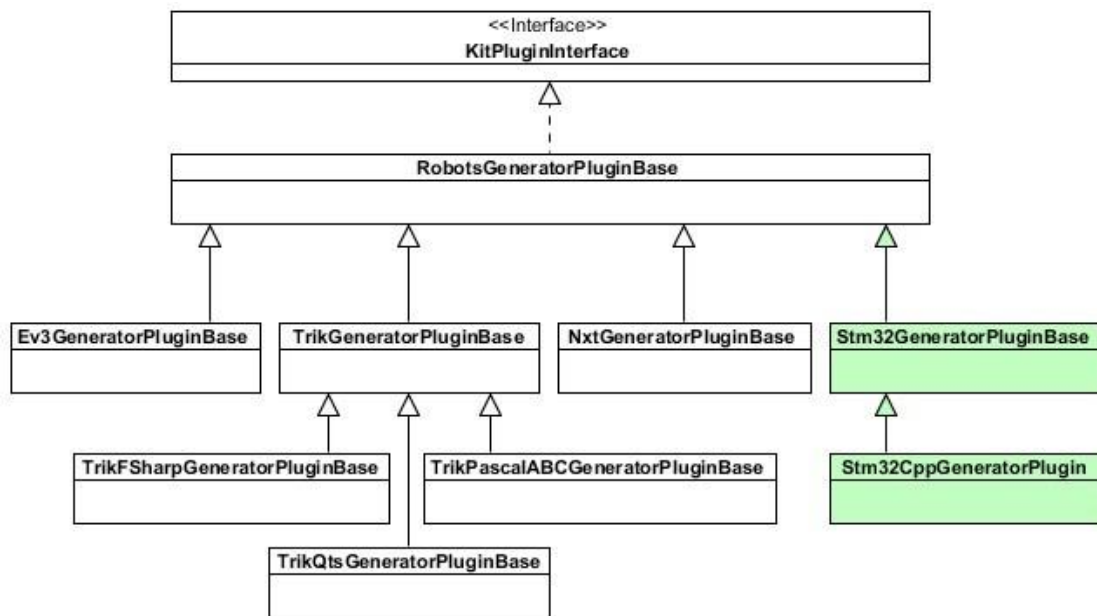


Рис. 3.4 Архітектура генераторів у TRIK Studio з генератором коду мовою C++.

Другим модулем є підсистема, основною метою якої є компіляція вихідного коду програми мовою C++ разом з багатьма потрібними бібліотеками в двійковий код та завантаження отриманого коду на плату мікроконтролера. Компіляція та завантаження програми починається після запуску конкретного елемента програми користувачем. Схему цієї підсистеми можна проаналізувати рисунку 3.5.

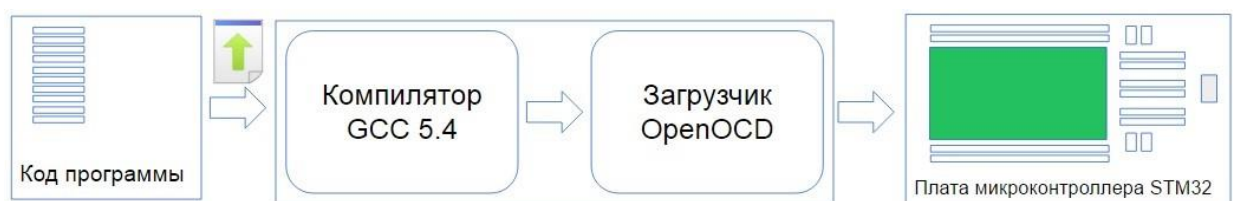


Рис. 3.5 Схема процесу компіляції та завантаження програми на плату мікроконтролера STM32.

3.2 Плагін для мікроконтролера STM32 TRIK Studio

У процесі виконання роботи був створений новий модуль, що дозволяє створювати графічні програми для мікроконтролера STM32 в TRIK Studio. Цей модуль відповідає за створення нової моделі робота, що називається «STM32», і додавання всіх необхідних можливостей для роботи, а саме використання режиму генерації програмного коду мовою C++.

Розглянемо ієрархію класів плагінної системи TRIK Studio, зображення якої можна побачити на малюнку 11. Базовим інтерфейсом для всіх модулів, що підключаються, чи то генератор або інтерпретатор, є `KitPluginInterface`. Він повідомляє найзагальніші для всіх плагінів компоненти, такі як унікальний ідентифікатор, метод ініціалізації, ім'я для відображення користувачеві і т.д. Застосування наперед квазаного елемента можна у класі `Robots Generator PluginBase`, який є базовим для всіх генераторів. Від нього успадковуються всі модулі, що відповідають за генерацію коду для кожної з підтримуваних моделей роботів, а саме `Ev3 Generator Plugin Base`, `Nxt Generator Plugin Base`, `Trik Generator Plugin Base` і доданий до них в рамках даної роботи `Stm32GeneratorPluginBase`. На цьому рівні ієрархії створюється модель робота, з якою надалі відбуваються необхідні маніпуляції. Для даного класу був реалізований нащадок `Stm32CppGeneratorPlugin`, відповідальний за генерацію коду мовою C++ та запуск процесу компіляції та завантаження програми на плату мікроконтролера.

3.3 Генератор коду на мові C++

Для здійснення процесу генерації коду на мові програмування, що раніше не підтримується середовищем TRIK Studio, потрібно, як говорилося раніше, сформувати структуру генерованого коду шляхом завдання шаблонів генерації. Для цього було оголошено та реалізовано шаблони генерації для таких блоків, як блок читання значень аналогового сенсора — були реалізовані блоки для інфрачервоного сенсора, сенсора світла, датчика відстані, датчика торкання; блок очікування в мілісекундах, блок встановлення кольору фону на екрані плати мікроконтролера, блок встановлення кольору світлодіода на платі

мікроконтролера (світлодіоди LD3 та LD4), блок виведення переданого користувачем тексту на екрані мікроконтролера. Дані шаблони генерують виклики відповідних функцій, оголошених у виділеному модулі Brick, який підключається до отриманої в процесі генерації програми. Мета даного модуля полягає в наданні інтерфейсу між програмою користувача C++ і драйверами мікроконтролера, які є частиною даної бібліотеки STM32Cube. Цей модуль інтегрований у програму TRIK Studio разом із необхідними бібліотеками для використання у процесі з мікроконтролером.

Отриманий у результаті генерації код програми відображається користувачеві в окремому вікні, в якому він може вносити зміни до програмного коду перед відправкою програми на виконання на платі мікроконтролера.

Як цільова мова програмування використано версію стандарту мови C++, однією з основних можливостей якої є безпосереднє виведення типів, який можан використати у процесі генерації. Ключове слово «auto» дозволяє зменшити надмірність та громіздкість програмного коду, проте ніщо не заважає користувачеві створювати програму у звичному стилі у вікні редагування коду та виконувати створений код на платі мікроконтролера.

3.4 Інтеграція компілятора та завантажувача з TRIK Studio

Існує безліч різних посібників та методичних посібників з програмування мікроконтролерів STM32 [3], [18], [23]. Основні ідеї, що стосуються компіляції та завантаження програми, були взяті з посібника з розробки програм для STM32 розробника Енді Брауна.

Для запуску програми мовою C++ на виконання необхідно спочатку отримати бінарний код програми, а потім отриманий двійковий код відправити на виконання на плату мікроконтролера. Для цього розроблена підсистема інтеграції компілятора та завантажувача програми з TRIK Studio.

Для роботи підсистеми використовується інструментарій GNU Tools ARM Embedded [9], до якого входить компілятор GCC

5.4 (GNU Compiler Collection) [10] для архітектури ARM, драйвер вбудованого

в плату мікроконтролера програматора ST-LINK/V2 [16], який працює за інтерфейсом JTAG, та бібліотека OpenOCD (Open On-Chip Debugger) [14] для завантаження прошивки на плату мікроконтролера.

JTAG (Joint Test Action Group) — це відомий у використанні промисловий апаратний інтерфейс, який використовується в багатьох мікросхемах для прошивки, налагодження та тестування. Назва походить від групи розробників стандарту IEEE 1149.1-1990, де були описані правила дій над створенням даного інтерфейсу. Стандарт передбачає виділення в мікросхемі спеціального набору блоків, стан працездатності яких можна оцінити незалежно від мікросхеми за допомогою окремих висновків. При цьому є можливість не тільки перевіряти роботу блоків, а й керувати ними. У більшості випадків використовують чотири або п'ять виділених сигнальних ліній.

При запуску сформованої програми мовою C++ на виконання відбувається виклик спроектованого у межах даної роботи пакетного файлу. При цьому дивлячись яку систему використовуєте, на якій запущено середовище програмування TRIK Studio, буде запущено пакетний файл, що відповідає своїй операційній системі. Підтримуються операційні системи Windows, Linux, MacOS X. Кожен пакетний файл містить набір команд для підготовки оточення для запуску процесу компіляції, а саме запис у змінну оточення шляхів до інструментів, необхідних для здійснення компіляції та завантаження програми. Користувачеві достатньо лише вказати у пакетному файлі шляху до встановлених ним бібліотек. Також пакетний файл містить команду для запуску виконання файлу Makefile.

За основу Makefile для підсистеми інтеграції компілятора та завантажувача з TRIK Studio було взято файл, створений Стефаном Вогелем — розробником з Німеччини, який розмістив посібник з компіляції та прошивки плати STM32 для вільного використання [17]. За допомогою практичного використання користувачами цей файл був оптимізований для роботи з платою мікроконтролера STM32F429I-Discovery. Також було користувачами реалізовано прототип засобу програмування мікроконтролера.

Makefile здійснює запуск компіляції на пристрої користувача та може додавати для виконання програматору плати ST-LINK/V2. Makefile забезпечує

коректне лінування з багатьма потрібними бібліотеками та драйверами мікроконтролера STM32, які інтегровані в середовище програмування TRIK Studio.

3.5 Використання мікроконтролера STM32

Було проведено апробацію створеного засобу програмування мікроконтролера STM32 на платі мікроконтролера STM32F429I-Discovery.

На рисунку 3.6 представлений інтерфейс середовища програмування роботів TRIK Studio.

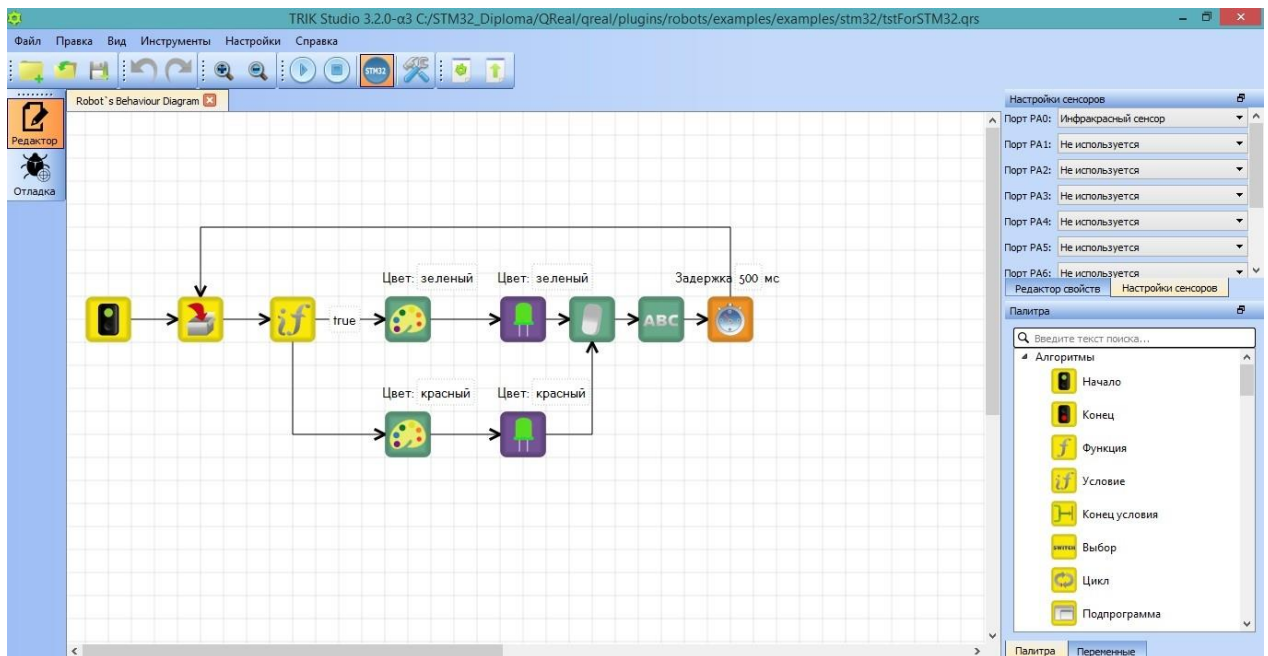


Рис. 3.6 Інтерфейс середовища TRIK Studio.

Апробація проводилася на прикладі графічної програми, наведеної на рисунку 3.7.

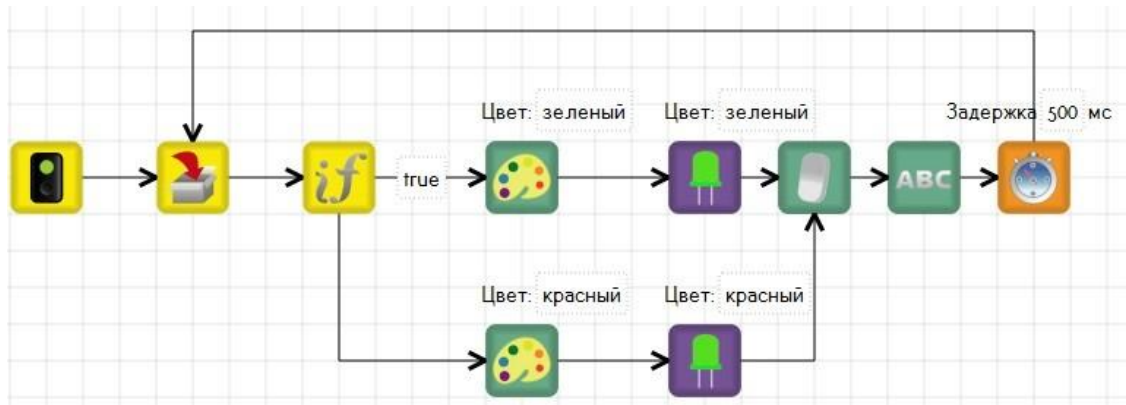
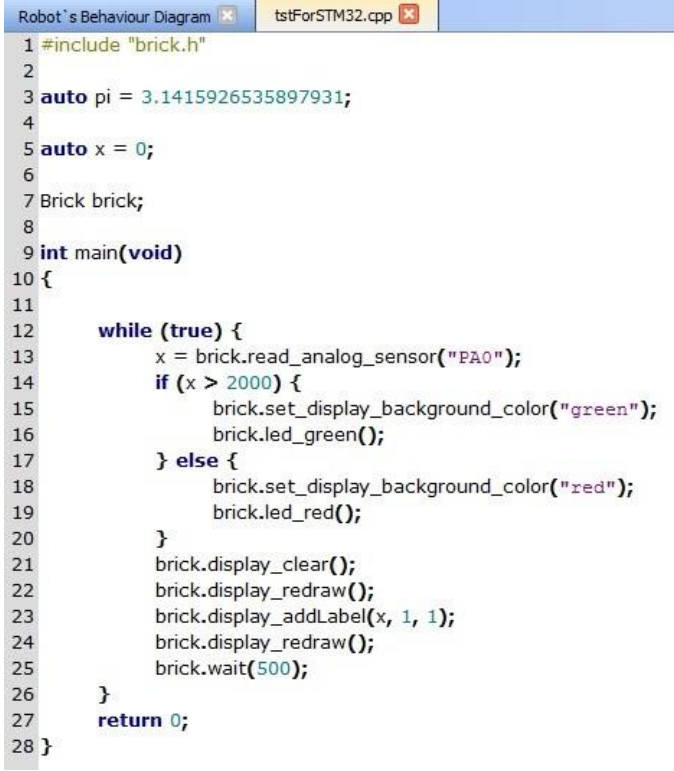


Рис. 3.7 Тестова програма візуальною мовою.

Ця програма в нескінченному циклі кожні півсекунди враховує значення інфрачервоного сенсора, підключеного до порту PA0, і в залежності від значення або зафарбовує фон дисплея плати мікроконтролера зеленим кольором і виставляє значення світлодіода в зелений колір (світлодіод LD3), або зафарбовує екран і включає червоний колір світлодіода плати (світлодіод LD4). Потім екран очищається, і на дисплеї відображається значення, яке показує датчик. Отже, відбувається визначення присутності об'єкта перед датчиком.

Для проведення апробації до плати мікроконтролера STM32F429I-Discovery був підключений інфрачервоний аналоговий датчик, значення якого вимірювалися в процесі проведеного експерименту. Мікроконтролер послідовно підключався до пристроїв під задіянням операційних систем Windows 10, Ubuntu 16.04, MacOS X. На кожному пристрої були встановлені всі необхідні для проведення апробації інструментарії, а саме були встановлені бібліотеки GNU Tools ARM Embedded та OpenOCD, а також були прописані шляхи до встановлених бібліотек у відповідній операційній системі пакетний файл. Для пристрою під використанням операційної системи Windows також було встановлено драйвер для програматора ST-LINK та інструментарій Cygwin для виконання команд середовища UNIX у системі Windows. На кожному пристрої було запущено програму TRIK Studio з підключеним плагіном для програмування мікроконтролера STM32. У середовищі програмування роботів був відкритий проект із графічною діаграмою програми, представленою та описаною вище. За цією діаграмою згенерувався

код програми мовою C++, яку можна проаналізувати на рисунку 3.8. Код відкривається в окремому вікні, в якому користувач може вносити зміни, що були перевірені під час апробації.



```
Robot's Behaviour Diagram x  tstForSTM32.cpp x
1 #include "brick.h"
2
3 auto pi = 3.1415926535897931;
4
5 auto x = 0;
6
7 Brick brick;
8
9 int main(void)
10 {
11
12     while (true) {
13         x = brick.read_analog_sensor("PA0");
14         if (x > 2000) {
15             brick.set_display_background_color("green");
16             brick.led_green();
17         } else {
18             brick.set_display_background_color("red");
19             brick.led_red();
20         }
21         brick.display_clear();
22         brick.display_redraw();
23         brick.display_addLabel(x, 1, 1);
24         brick.display_redraw();
25         brick.wait(500);
26     }
27     return 0;
28 }
```

Рис. 3.8 Тестова програма мовою C++.

Потім програму було запущено на виконання. Через деякий час (близько 15 секунд) на мікроконтролері замигала лампочка світлодіода LD1, що свідчило про прийом двійкових даних, т.к. Цей світлодіод є індикатором передачі даних через інтерфейс USB. У вікні програми TRIK Studio «Помилки» з'явилася інформація про попередження, а також налагоджувальна інформація, яка зібралася за час компіляції та завантаження програми. Стан налагоджувальної плати мікроконтролера можна проаналізувати на рисунку 3.9.

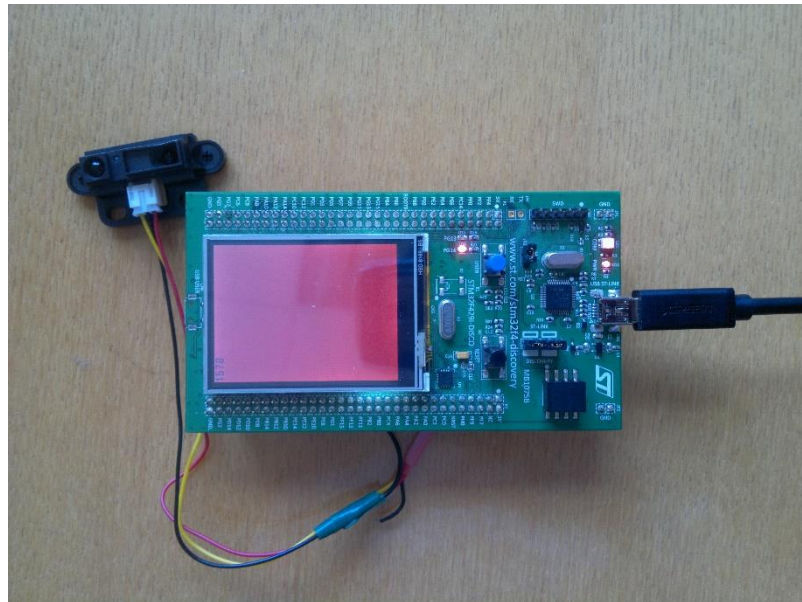


Рис. 3.9 Стан мікроконтролера за відсутності об'єкта перед інфрачервоним датчиком.

У разі наближення деякого об'єкта на близьку відстань до інфрачервоного датчика відбувається перехід налагоджувальної плати в інший стан, який можна побачити на рисунку 3.10. Дисплей фарбується в зелений колір, і світиться зелений світлодіод плати.

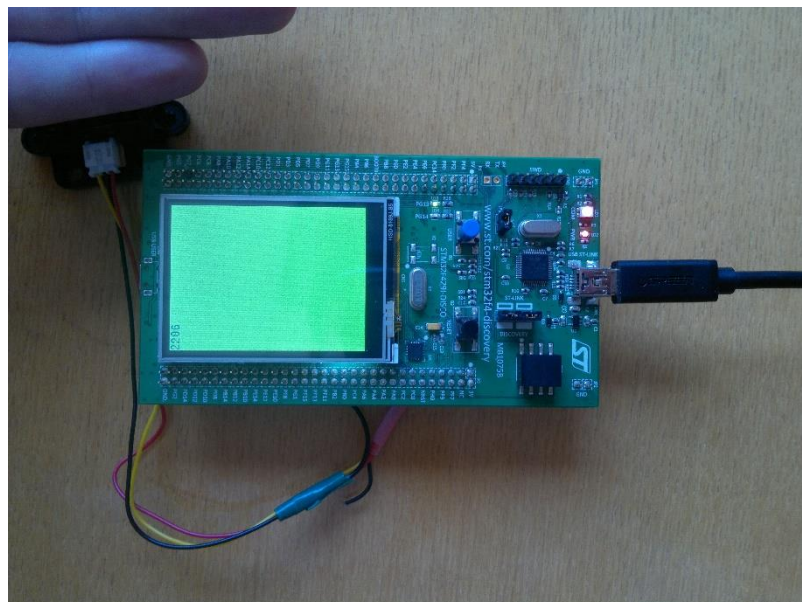


Рис. 3.10 Розпізнавання об'єкта перед інфрачервоним датчиком.

Якщо об'єкт видалити від датчика, то плата повернеться до свого початкового стану. Це можна побачити рисунку 3.11.

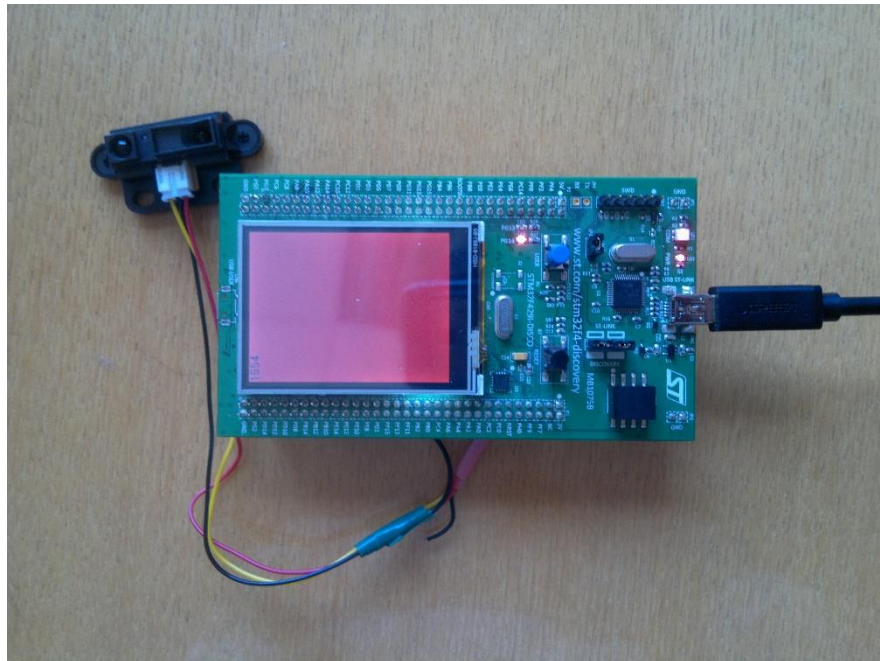


Рис. 3.11 Стан мікроконтролера при видаленні об'єкта від інфрачервоного датчика.

У разі вимкнення плати мікроконтролера від комп'ютера живлення вимикається. При цьому при подальшому вмиканні мікроконтролера виконуватиметься та програма, яка була завантажена на плату. Кожне наступне завантаження прошивки перезаписує робочу програму мікроконтролера.

4 БЕЗПЕКА ЖИТТЄДІЯЛЬНОСТІ, ОСНОВИ ОХОРОНИ ПРАЦІ

4.1 Етапи створення безпечних умов праці

Утворення безпечних, нешкідливих і сприятливих умов праці - одне з найважливіших задач у теперішньому виробництві, яке важливо не тільки в машинобудуванні, але і в соціальному плані.

На сучасному етапі охорона праці також є метод правових, соціально-економічних, лікувально-профілактичних дій на забезпечення здоров'я людини і вміння працювати.

Основним принципом політики в сфері безпеки є визнання важливості здоров'я та сфер життя працівників. Виходячи з цього, формуються основні завдання поля ОП:

- Розробка та впровадження високотехнологічних технологій.
- Підвищити рівень безпеки існуючого виробничого обладнання і технічних процесів за дією яка дає можливість усунути небезпечні і які можуть зашкодити виробничі елементи;
- Удосконалуйте обладнання підприємства новітніми технічними засобами безпеки та промислової гігієни.
- Стандарне впровадження (SSBP);
- Здійснення підготовки населення та покращення здібностей працівників ОП.
- Удосконалення технічних і виробничих площ.
- Середовища, яке оточує повинно бути захищене.

Заходи охорони здоров'я, безпеки та гігієни тісно пов'язані з технологіями. Зміни в технічних процесах та обладнанні. Необхідно вивчити їх з точки зору безпеки виробництва і знайти відповідні методи усунення факторів, що призводять до аварій.

Одне з головних завдань зробити свою роботу в компанії безпечною і здоровою. Це завдання реалізується шляхом комплексної механізації

виробничих процесів, більш повної автоматизації, технологічної досконалості, переходу від безпечної до безпечної технології і широкого впровадження методів покращення умов праці, безпеки і наукової організації праці.

У повній відповідності з чинним законодавством компанії створюють внутрішні правила та інструкції з безпеки, а також профілактичні заходи, докладні інструкції щодо безпечної практики роботи для кожної професії та кожного виду роботи. Багато наказів Міністерства промисловості реалізуються на виконання завдань з охорони праці та безпеки.

Контроль за дотриманням законодавства та здійсненням методів з охорони праці на багатьох підприємствах залишається за державними та громадськими органами моніторингу. Функції державного нагляду виконують профспілкові технічні інспекції, державні муніципальні технічні інспекції, державні санітарні інспекції тощо.

Контролюючі органи з охорони праці та державні інспектори відіграють важливу роль у створенні надійних та безпечних умов праці на підприємствах. Цей комітет формується з членів профспілки (робітників, інженерів, працівників) і очолюється членами місцевих комітетів підприємства, членами цехових комітетів всередині цеху.

Перевірка повинна: Стежити за дотриманням трудового договору, правил і норм з безпеки та промислової гігієни, а також заходів щодо запобігання критичним випадкам на виробництві та зменшення захворюваності, контролювати організацію та якість працівників у безпечній трудовій практиці, а також своєчасні комбінезони, спеціальне взуття та засоби індивідуального захисту.

Перевірка з охорони праці сприяє тому що на виробництві будуть використовувати більш сучасні технології та нову техніку, автоматизації і удосконалення виробничих процесів з метою ліквідування тяжких ручних робіт.

Мають місце номенклатури, які підпадають під план роботи підприємств: профілактика нещасних випадків, хвороб на робочому місці, загальне поліпшення умов праці. Розумне використання коштів, виділених на номенклатуру, допоможе зменшити кількість критичних випадків на підприємстві та полегшити роботу працівникам.

Правильне навчання співробітників правилам безпеки і безпечним методам роботи значною мірою сприятиме боротьбі з аваріями на підприємствах. Працівники зобов'язані вивчати управління та обслуговування підрозділів, машин і механізмів на спеціальних курсах, організованих спеціалізованими навчальними закладами або навчальними центрами.

На виробництві працівники отримують відповідне керівництво, обсяг і зміст яких і тривалість стажування працівників за кожною професією встановлюються керівником установи підприємства. Інструкції передбачають загальні небезпеки безпеки, виявлені на об'єкті, і конкретні небезпеки безпеки, пов'язані з виконанням певних видів робіт.

Всі працівники повинні бути повторно проінструктовані з безпеки не менше чим один раз на рік і перевірені комітетами. При впровадженні нових технічних процесів і методів роботи, а також при впровадженні нових правил і норм безпеки працівники зобов'язані пройти ретельніше навчання і керівництво. Результати всіх видів керівництва для перевірки знань співробітників, які отримали позитивні оцінки, будуть введені на карту керівництва з безпеки, підписану всіма членами комітету і тими, хто отримує вказівки.

4.2 Обґрунтування потенційних небезпек і шкідливої дії виробничого середовища

Розроблений пристрій являє собою автоматичну систему контролю температури в суворих умовах. До працівників небезпечних професій

відносяться інженери і чергові працівники, які використовують контейнери, що містять шкідливі речовини, а також оператори, які є користувачами комп'ютерів, з підвищеним опроміненнями у середовище. З урахуванням деталей роботи користувача представлені вимоги і рекомендації щодо режимів роботи і відпочинку, освітлення і вентиляції і т. д.

Список потенційно небезпечних і факторів, що можуть зашкодити для виробництва залежить від декількох умов. Найважливішими з них є технологія виробничого процесу, стан виробничого обладнання, обладнання секонд-хенду.

При цьому оператор має справу з електромагнітним випромінюванням, здатним впливати на організм людини дозами, що перевищують допустимий діапазон і викликати професійні захворювання.

Вібрації з частотою більше 20 000 Гц називаються ультразвуком. Розроблені ультразвукові термометри працюють на частоті 50 кГц - це низькі частоти. Довжина хвилі дорівнює 10-1 км.

Дія ультразвукового технологічного обладнання передбачає поширення звукових коливань в широкому спектральному діапазоні від 20 до 70 кГц і більше. При ультразвукових коливаннях також генерується звуковий шум з максимальною енергією від 8 до 10 кГц, інтенсивність якого досягає від 100 до 120 дБ. Іншими словами, шум може викликати звукову втому.

У просторі ультразвук швидко розпадається, але в твердих і рідких середовищах загасання сповільнюється. Коли ультразвук діє на твердий чи інший елемент, частинки вібрують зі швидкістю ультразвуку і температура підвищується. Коли ультразвукові хвилі діють на рідини, відбувається кавітація. Рідина утворює численні проміжки у вигляді дрібних бульбашок, що змушує рідину «ламатися». Коли ці кавітаційні бульбашки закриваються, утворюється великий (до декількох сотень атмів) тиск, що руйнує сусідні тверді поверхні.

Вплив ультразвуку малої потужності на людину в основному є термічним ефектом, так як кавітація руйнує окремі клітини організму і викликає локально сильне підвищення температури. Під впливом ультразвуку також відбуваються явища дисперсії і застигання, прискорюючи хімічний процес [22].

Дія ультразвуку сприяє головному білю, нездужання, втрату зору і тиску на очне яблуко, втрату апетиту, сухість у роті, затвердіння мови, болі в животі і т.д., які зникають після роботи. При цьому незначна зміна температури тіла (0,5...1С), частота серцевих скорочень 5 в хвилину. Спостерігаються фізіологічні зміни, що складаються зі зниження в 10 ударів, стійких порушень апетиту, тьмяної чутливості шкіри. втрата слуху і зору і т. д.

Ультразвукове контактне випромінювання небезпечніше, ніж в повітрі. ГОСТ 12. 1. 001-83 надає посилання на рівень звукового тиску, що виробляється вібрацією повітря на ультразвукових частотах. Нормалізація методів звукового тиску виконується в діапазоні частот по 3 октави для умов впливу ультразвуку на оператора на період 8-годинного дня. Значення рівня ультразвукового тиску наведені в табл. 4.1.

Таблиця 4.1 – Оцінювання впливу звукового тиску

Середньгеометричні частоти третьоктавних полос, Гц	12500	16000	20000 і вище
Поправка, дБ	75	85	110

Ультразвукові рівні тиску вимірюються після запуску, ремонту та експлуатації при перевірці обладнання хоча б раз у рік. Технічне обладнання повинно мати ультразвукові характеристики. Рівень звукового тиску третьої октавної смуги в допустимому діапазоні частот. Вимірюється в 4 точках по контуру приладу на відстані 1,5 м і менше та також на висоті 1,5 м від основи.

Відстань не менше 0,5 м від контуру приладу і більше 2 м від відбивної поверхні.

Оборона від ультразвуку полягає в тому, щоб звести до мінімуму інтенсивність ультразвукового випромінювання і його тривалість. Тому потрібно використовувати конкретну установку джерела живлення, необхідну для розстрочки, і включити її в міру необхідності.

Особи, які систематично страждають від ультразвукового випромінювання, втрачають концентрацію і втрачають рівновагу. Перебування в області ультразвукових генераторів може викликати слабкість, нездужання, головні болі і біль у вухах, змінення частоти серцебиття і погний сон [35].

При середній і високій інтенсивності його наслідки можуть бути оніміння і навіть фатальними. Методи захисту від негативної дії ультразвуку виконуються одноково, подібно до захисту від шуму.

Методи індивідуального захисту включають антишумові та рукавички.

Рукавички повинні бути в два шари. Верхній шар - гумовий, нижній шар - бавовняна або вовняна тканина.

Якщо в результаті аналізу було встановлено початкові етапи захворювання внаслідок дії УЗД потерпілому необхідно вчасно припинити роботу з УЗД, що дає можливість уникнути негативній дії да сприяти виздоровленню. Ультразвукове технологічне обладнання розміщене в корпусі, та має звукоізоляційну камеру. Матеріал і поверхня екрану такі ж, як і для захисту від шуму.

Комп'ютерні оператори, працівники з обробки даних, програмісти та інші люди, які регулярно піддаються впливу шкідливих виробничих факторів, таких як підвищений шум, підвищена температура навкруги у машинному відділенні, відсутність або недостатнє освітлення, небезпека від дії електричного струму, статична електрика і т. д. Електрика, електромагнітне випромінювання і т. д.

У час роботи з ВДТ (катодною променевою трубкою) на організм впливає безліч факторів фізичних властивостей, але всі вони знаходяться в межах нормальних значень за діючими нормами і набагато нижче. Дані вимірювань наведені в таблиці 4. 2.

Деталі освітлення на робочому місці VT викликані кількома особливостями. До них відносяться:

- Наявність екранів, клавіатур і документів пов'язана з різноманітністю освітлення об'єктів візуальної роботи користувача комп'ютера. Щоб розмістити ці об'єкти в різних зонах спостереження, необхідно при потребі переводити погляд з однієї лінії зору на іншу.
- Маніпулюйте пульсуючими самосвітлюючими об'єктами, які завжди знаходяться в центрі вашої області зору. Він не відповідає нормативним вимогам для обмеження пульсації та сліпоти.

Таблиця 4.2 – Аналіз потенційних небезпек користувача ЕОМ

Види випромінювань	Діапазон	Фактичні дані замірів	Нормовані значення
Рентгенівське	понад 1,2 Кев	10-12 мк р/год	75,0 мк р/год
Ультрафіолетове	230-280 нм	0-0,01	0,01Вт/м ²
Видимий діапазон	320-500 нм	2,0-3,0	10,0 Вт/м ²
ІК-випромінювання	700нм-1мм	0,05-4,0	100,0 Вт/м ²
Електростатичне поле	0Гц	15кВ/м	20-60 кВ/м ²
Електричний струм	—	U=220 В; I≤6А F=50 Гц	—

- На роботі яскравість в межах поля зору несприятливо розсіюється, тому що освітлена поверхня навколо поля зору може бути яскравіше центру поля зору - темна, тьмяна, а іноді і незаповнена знаками.

- Погляд користувача при роботі з екраном практично горизонтальний, тому сліпучий ефект світильника, який освітлює робочу кімнату, більше, ніж у інших освітлювальних приладів. Це призводить до зниження кута дії різних джерел сліпоти, яке є негативним та сприяє збільшенню сліпоти. Через наявність дзеркальної, не плоскої зовнішньої поверхні ЕПТ неможливо повністю видалити всі відбиті відблиски з поля зору.

Гучні шуми можуть ускладнити розпізнавання кольорів, зменшити реакцію розпізнавання образів і знизити продуктивність (на 5-12)%. Тривалий вплив промислового шуму з силою звукового тиску 90 дБ зазвичай призводить до (30-60)% зниження продуктивності. Отже, з останніми дослідженнями, рівень звукового тиску ОС не повинен перевищувати 50 дБА [36].

4.3 Методи реалізації нормальних умов роботи

Зроблено низку заходів для підтримки працездатності та реалізація нормальних умов праці. Нормальні умови праці можна охарактеризувати такими факторами, як температура і вологість, рівень звуку в приміщенні, освітлення і т. д. При зниженій увазі при високих температурах ви стаєте нетерплячими і недбалими. Низький - інтенсивний тепловіддача з організму знижує рухливість кінцівок. Впливає на тепловіддачу і вологість. Це негативно позначається на організмі людини. Сухе повітря збільшує випаровування, в результаті чого виникає сухість слизових оболонок і шкіри. І навпаки, дуже вологе повітря послаблює випаровування. Нормативні характеристики мікроклімату промислових об'єктів наведені в табл. 4. 3.

Таблиця 4.3 – Нормативні характеристики мікроклімату у виробничих приміщеннях

					Повітряна
--	--	--	--	--	-----------

Виробниче приміщення	Категорія робіт	Пора року	Температура повітря, °С	Відносна вологість, %	швидкість, м/с
Дисплейний клас	Легка (1-аА)	Холод	19 – 23	38-62	0,15
		Тепло	19 – 28	39-61	0,17

Оскільки зорова робота з ВДТ є специфічною, то завданням є забезпечення необхідних умов візуальної роботи користувачів ЕОМ за рахунок розподілу яскравостей у полі зору працюючого та максимально можливого зменшення засліпленості від прямого і відбитого блищання та відмежування від постійної пульсації зображення на ВТ, які посилюють загальну та зорову втому. Для цього необхідно забезпечити параметри освітлення, тобто правильно вибрати приміщення, розташувати робоче місце.

При виборі приміщення для робочого місця необхідно звернути увагу, що вікна при відповідному куті розміщення дають відблиски на екранах монітора, викликаючи при цьому засліпленість у користувачів. Необхідно обладнати сонцезахисний пристрій, приміщення, в якому влаштовано відкриття світла з одного боку, ідеально підходить для розміщення ВД. Площа засклення не повинна перевищувати 25% в площі стін з вікнами. Робочі місця доцільно розташовувати в глибині приміщення. При використанні в загальному освітленні світильників прямого світла, робочі території повинні бути обов'язково розташовані в ряди, паралельні до стіни з вікнами. Потрібно звернути увагу, що територія де працюєш з ВДТ вимагає значної площі ($4,6 \times 7 \text{ м}^2$), оскільки навколо нього повинен залишатись вільний простір не менше 1 метра з кожного боку.

Потрібно дотримуватись також гігієнічних та світлотехнічних рекомендацій. Системи освітлення має бути загальною і загальною локалізованою. Тип лампи в залежності від розподілу світла і того, як лампа

влаштована в приміщенні, врахувати і також висоту до стелі, місця роботи і кількості завдань. Рівні освітленості наведено в табл. 4.4.

Освітлювальні прилади повинні розташовуватися над проходами між колонами на роботі, або з суцільними лініями, або розставлені один від одного, в залежності від кількості колонних світильників, необхідних для забезпечення нормалізованого освітлення на роботі.

Для профілактики та підтримання працездатності необхідно використовувати перервидля відпочинку згідно регламенту. У період роботи за дисплеєм у режимі праці та відпочинку необхідно передбачити через кожні 40-45 хвилин трихвилинні та п'ятихвилинні перерви для відпочинку.

Таблиця 4.4 – Рівні освітленості у приміщеннях з ВДТ

Вид дисплея	Група напруженості зорової роботи	Нормована освітленість				Тип світиль- ника
		загальне		комбіноване		
		E_T , не менше (лк)	E_V (лк)	E_T , не менше (лк)	E_V (лк)	
Одноколірний (яскравий знак на темному фоні)	II	300	150- 100	400	150- 100	ЛПО-12- Кососвет
	I	400	200- 150	500	300- 150	ЛСП-12- Кососвет

Графічний багатоколірний та одноколірний (яскраві лінії на темному фоні)	Будь-яка	-	-	400	100- 75	
--	----------	---	---	-----	------------	--

Сумарний час праці з відеотерміналом (4 години) краще розподілити на дві частини і працювати по 2 години у першу та другу половини робочого дня. Під час праці чергового інженера через нервові навантаження знижується увага і зорове сприйняття, занижена м'язова активність. Для зниження монотонності в роботі, перерви відпочинку потрібно супроводжувати вправами для рук та розминки хребта, особливо шийних хребців і м'язів.

4.4 Розрахунок заземлення пристрою для заземлення системи контролю температури

Оскільки електроустановки мають великі потенційні небезпеки, необхідно приділити велику увагу створення умов електробезпеки, для того щоб зробити роботу персоналу безпечнішою.

Для уникнення шкідливого впливу завад на роботу схеми необхідно передбачити і розрахувати заземлення системи.

Вихідними даними для розрахунку заземлення є:

- напруга живлення приладу – 220В, потужність < 50 ВА;
- згідно “Правил устроювання електроустановок” опір заземлення має бути у межах $R < 4 \text{ Ом}$;

- в якості заземлення використовується труба;
- довжина – $L=2\text{м}$;
- діаметр – $d=0.075\text{ м}$;
- глибина заземлення – $t=1\text{м}$;
- обчислювальний питомий опір землі ρ складає для суглинків $150\text{ Ом}\cdot\text{м}$;
- коефіцієнт використання заземлювача $K_B=0.8$.

Визначимо опір зникаючого струму з тільки одного заземлючого пристрою, який розташовано вертикально за формулою:

$$R_1 = \frac{\rho}{2 \cdot \pi \cdot 4} \cdot \text{Ln} \frac{2 \cdot L}{d} + 0.5 \cdot \text{Ln} \frac{4 \cdot t + L}{4 \cdot t - L}, \quad (5.1)$$

де $L=2.5\text{ м}$ – довжина заземлювача;

$d=0.1\text{ м}$ – діаметр заземлювача;

$t=1\text{ м}$ – відстань заглиблення заземлювача.

З формули (5.1) маємо:

$$R_1 = \frac{150}{2 \cdot 3.14 \cdot 4} \text{Ln} \frac{2 \cdot 2.5}{0.1} + 0.5 \cdot \text{Ln} \frac{4 \cdot 1 + 2.5}{4 \cdot 1 - 2.5} = 25.52 \text{ Ом.}$$

визначимо потрібну частину заземлювачів, що паралельно з'єднані за формулою

$$n = \frac{R_g}{R \cdot k_b}, \quad (5.2)$$

де $k_B=0.8$ – коефіцієнт використання заземлювачів.

З формули (5.2) $n = \frac{25}{4 \cdot 0.8} = 7.81$. Вибираємо, що необхідно 8 заземлювачів.

Визначимо опір горизонтального заземлювача, який використовується для з'єднання електродів розміщених вертикально, враховуючи, що він має форму смуги прямокутного січення і виконаний із сталі:

$$R_r = \frac{\rho}{4 \cdot \pi \cdot 4} \cdot \text{Ln} \frac{2 \cdot l}{b \cdot t},$$

(5.3)

де $b=0,1$ м – ширина горизонтального заземлювача;

$l=1,05*1,5*9=12,6$ – довжина горизонтального електрода при розташуванні вертикальних електродів в один ряд:

$$R_r = \frac{150}{4 \cdot \pi \cdot 4} \cdot \ln \frac{2 \cdot 12,6}{0,1 \cdot 1} = 16,53 \text{ Ом.}$$

Визначимо загальний опір розтікання струму заземлювача, який використовується для забезпечення заземлення комплексу за формулою:

$$R_i = \frac{R_b \cdot R_{ri}}{R_r \cdot k_u + R_b \cdot n \cdot k_b}.$$

(5.4)

Розрахувавши, $R_i = \frac{25 \cdot 16,53}{16,53 \cdot 0,5 + 25 \cdot 8 \cdot 0,8} = 2,46$ Ом бачимо, що

необхідний рівень опору заземлення отримано.

ВИСНОВКИ

В результаті роботи були отримані такі результати:

- розроблено архітектуру системи програмування мікроконтролера STM32 у середовищі TRIK Studio;
- реалізований модуль, що підключається для мікроконтролера STM32 в TRIK Studio;
- реалізований генератор коду з програми мовою візуальних діаграм у програму мовою C++;
- розроблена підсистема інтеграції компілятора та завантажувача програми з TRIK Studio;
- проведено апробацію засобу програмування на платі мікроконтролера STM32F4-Discovery;

ПЕРЕЛІК ПОСИЛАНЬ

1. Цифровая схемотехника и архитектура компьютера. Дополнение по архитектуре ARM | Харрис Дэвид М., Харрис Сара Л.
2. Carmine Noviello. Mastering STM32 . Изд. Leanpub, 2018. – 852с
3. C++ STM32 Development Environment. – URL: <http://andybrown.me.uk/2015/03/22/stm32dev-windows/>
4. Лабораторный практикум для изучения микроконтроллеров архитектуры ARM Cortex-M4 на базе отладочного модуля STM32F4 Discovery / Бугаев В.И., Мусиенко М.П., Крайнык Я.М. – Москва-Николаев: МФТИ-ЧГУ, 2013. – 71 с.
5. Dmitry Mordvinov, Yurii Litvinov, Timofey Bryksin. TRIK Studio: Technical Introduction. – Proceedings of the FRUCT'20, 2017, ISSN 2305-7254, ISBN 978-952-68653-0-0. pp 296-308.
6. Дэвид Карлсон. Eclipse. Издательство Лори, 2008.-336с.
7. Машнин Т. С. Среда разработки ECLIPSE Руководство разработчика, 2014
8. Машнин Т. С. » Eclipse: разработка RCP-, Web-, Ajax- и Android-приложений на Java» БХВ-Петербург, 2013 год, 384 стр.
9. GNU ARM Embedded Toolchain. – URL: <https://developer.arm.com/open-source/gnu-toolchain/gnu-rm>
10. GNU Compiler Collection. – URL: <https://gcc.gnu.org/>
11. IAR Embedded Workbench. – URL: <https://www.iar.com/iar-embedded-workbench>
12. Keil Embedded Development Tools. – URL: <http://www.keil.com/>
13. Keil Microcontroller Development Kit. – URL: <http://www2.keil.com/>

com/mdk5/

14. OpenOCD. – URL: <http://openocd.org/>
15. QReal. – URL: <https://github.com/qreal/qreal>
16. ST-LINK USB driver for Windows. – URL: http://www.st.com/content/st_com/en/products/embedded-software/development-tool-software/stsw-link009.html
17. STM32 Cube-GCC. –URL: <https://github.com/stv0g/stm32cube-gcc>
18. STM32 Firmware and Debugging Manual. – URL:<https://github.com/mrsin/k-lab/wiki/STM32-прошивка-с-помощью-OpenOCD-и-отладка-в-Eclipse>
19. STM32 plugin for TRIK Studio. – URL: <https://github.com/PrikhodkoStanislav/qreal/tree/STM32>
20. STM32F429I-Discovery. – URL: <http://www.st.com/en/evaluation-tools/32f429idiscovery.html>
21. STMicroelectronics. – URL: http://www.st.com/content/st_com/en.html
22. Брыксин Т.А., Литвинов Ю.В. Среда визуального программирования роботов QReal:Robots. – Материалы международной конференции «Информационные технологии в образовании и науке». Самара. 2011. С. 332-334.
23. Бугаев В.И., Мусиенко М.П., Крайнык Я.М. Лабораторный практикум для изучения микроконтроллеров архитектуры ARM Cortex-M4 на базе отладочного модуля STM32F4 Discovery. – Москва-Николаев: МФТИ-ЧГУ, 2013.
24. Кознов Д.В. Основы визуального моделирования. – БИНОМ. Лаборатория знаний, Интернет-Университет Информационных Технологий, 2008.
25. Литвинов Ю.В., Кириленко Я.А. TRIK Studio: среда обучения программированию с применением роботов. – V Всероссийская конференция «Современное технологическое обучение: от компьютера к

- роботу» (сборник тезисов), СПб., ЗАО «Полиграфическое предприятие № 3», 2015, С. 5-7.
26. Приходько С.В., Литвинов Ю.В. Поддержка программирования микроконтроллера STM32 в TRIK Studio. – Материалы научно-практической конференции студентов, аспирантов и молодых ученых «Современные технологии в теории и практике программирования». СПб.: Изд-во Политехн. ун-та, 2017. С. 71-72.
 27. Терехов А.Н., Брыксин Т.А., Литвинов Ю.В. и др. Архитектура среды визуального моделирования QReal. – Системное программирование. Вып. 4. СПб.: Изд-во СПбГУ, 2009.
 28. «РадиоЛоцман» Журнал для тех, кто интересуется электроникой.
 29. И.Н.Огородников Микропроцессорная техника: Введение в Cortex-M. Учебное пособие, 2015-118 с.
 30. Микроконтроллеры Cortex-M0, M3, M4.
<https://mcusrp.ru/index.php/ucontrollers/mcu/113-mikrokontrollery-cortex-m0m3m4>
 31. ARM процессоры. <https://www.cnews.ru/book/ARM64>
 32. Веб-сайт компанії “STMicroelectronics”.
 33. Веб-сайт компанії “ARM Ltd.”.
 34. Stm32: Эпоха 32-битных микроконтроллеров наступила.
<https://www.compel.ru/lib/53953>
 35. ГОСТ 12.1.006 - 84.ССБТ. Електромагнітні поля радіочастот. Допустимі рівні на робочих місцях і вимоги до проведення контролю. - Введ.01.01.85.
 36. ГОСТ 12.1.003 - 83 *. ССБТ. Шум. Загальні вимоги безпеки. - Введ.01.07.84