

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
СУМСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ
Кафедра електроніки і комп'ютерної техніки

ПОЯСНЮВАЛЬНА ЗАПИСКА

до кваліфікаційної роботи

магістра на тему:

**ПРОГРАМНО-УПРАВЛЯЮЧА СИСТЕМА ВИМІРЮВАННЯ
ЕЛЕКТРИЧНИХ ПАРАМЕТРІВ**

Завідуючий кафедрою

Опанасюк А. С.

Керівник кваліфікаційної роботи

магістра

Кулик І. А.

Виконала студентка

Грахольська А.В.

Суми - 2021

РЕФЕРАТ

Пояснювальна записка містить 100 сторінок, 23 рисунка, 12 таблиць, 16 джерел.

В наш час є актуальним питання розробки великої кількості типів мікро-ЕОМ що мають різноманітні функціональні можливості і вартість, призначені для різного застосування, що може бути як окремими спеціалізованими приладами, так і системи автоматизованого управління технологічними об'єктами.

Актуальністю даної тематики є розробка вимірювача індуктивності та ємності із застосуванням мікроконтролера, характеристики якого в балансі якість-ціна-габарити. Для зменшення габаритів системи доцільним є використання індикатора з послідовним введенням інформації.

Об'єкт розробки – програмно-управляюча система вимірювання електричних параметрів, принцип роботи якої ґрунтується на вимірі енергії, що накопичується в електричному полі конденсатора і магнітному полі котушки.

Метою розробки даної кваліфікаційної роботи магістра є обґрунтування та створення оптимального вимірювача індуктивності та ємності, який буде відповідати всім заданим вимогам.

В даній конструкції передбачено застосування мікроконтролера та ЖКІ індикатора, що забезпечує створення простої, економічної і зручної в експлуатації системи, що характеризуються високою точністю вимірювання. В даній роботі передбачено застосування програмної корекції нуля, що здійснює компенсацію ємності, індуктивності клем і з'єднуючих дротів [1].

Для роботи проектованої системи досить простим є підключення вимірюваного елемента і зняття показань з індикатора.

На основі роботи програмно-управляючої системи вимірювання електричних параметрів, було розроблено алгоритм роботи, структурну, функціональну, принципальну електричну схеми.

Ключові слова: мікроконтролерна система, вимірювач індуктивності, вимірювач ємності.

ЗМІСТ

РЕФЕРАТ.....	2
ЗМІСТ.....	3
ВСТУП.....	4
1 ОГЛЯД ЛІТЕРАТУРИ ТА ПОСТАНОВКА ЗАВДАННЯ ПРОЕКТУВАННЯ	7
1.1 Огляд літератури.....	7
1.2 Постановка завдання проектування.....	14
2 ОБГРУНТУВАННЯ АЛГОРИТМУ РОБОТИ І СТРУКТУРНОЇ СХЕМИ ПРОГРАМНО-УПРАВЛЯЮЧОЇ СИСТЕМИ ВИМІРЮВАННЯ ЕЛЕКТРИЧНИХ ПАРАМЕТРІВ.....	17
2.1 Розробка алгоритму проектованої системи	17
2.2 Розробка структурної схеми проектованої системи	37
3 РОЗРОБКА СХЕМИ ЕЛЕКТРИЧНОЇ ФУНКЦІОНАЛЬНОЇ ПРОЕКТОВАНОЇ СИСТЕМИ.....	41
4 РОЗРОБКА І РОЗРАХУНОК ПРИНЦИПОВИХ ЕЛЕКТРИЧНИХ СХЕМ ВУЗЛІВ І БЛОКІВ СИСТЕМИ.....	46
4.1 Вибір елементної бази.....	46
4.2 Розрахунки та синтез основних електричних вузлів, блоків проектованої системи	52
5 НАУКОВО – ДОСЛІДНА ЧАСТИНА.....	55
6 ЕКОНОМІЧНА ЧАСТИНА.....	70
6.1 Розрахунок трудомісткості та заробітної плати	70
6.2 Розрахунок собівартості проектованої системи	73
6.3 Розрахунок матеріальних витрат	74
6.4 Калькуляція собівартості виробу	76
6.5 Визначення гуртової ціни виробу	76
7 РОЗРОБКА ПРОГРАМ ОБРОБКИ ДАНИХ І УПРАВЛІННЯ СИСТЕМОЮ	79
7.1 Висновки.....	Ошибка! Закладка не определена.
НАУКОВІ ПРАЦІ СТУДЕНТА.....	95
ВИСНОВОК.....	96
СПИСОК ЛІТЕРАТУРИ	97
ДОДАТОК А	99

					ЕЛІТ 8.171.00.10.376 ПЗ			
<i>Зм.</i>	<i>Лист</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>	Програмно-управляюча система вимірювання електричних параметрів. Пояснювальна записка	<i>Лит.</i>	<i>Лист</i>	<i>Листов</i>
Розроб.		Грахольська А.В.					3	100
Перевір.		Кулик І.А.						
Реценз.								
Н. Контр.		Гапич Н.В.						
Затверд.		Опанасюк А.С.				СумДУ ЕС.м-01		

ВСТУП

Зараз на планеті настав дивовижний час, учасником якого є всі ми без винятку. Кожен житель нашої планети є свідком небувалого технологічного зростання, стрімкого розвитку науки та технологій. Це за своєю суттю не просто дивно, а й феноменально. Якщо посудити – людська цивілізація існує вже кілька тисяч років, і лише за останні кілька сотень років стався такий раптовий стрибок. Зараз багато істориків, філософів та інших науковців намагаються розгадати загадку і дати цьому наукове пояснення.

Однак ніхто з них так і не дав чіткої відповіді, з чим може бути пов'язаний такий стрімкий стрибок. Щоправда, є кілька гіпотез, що пояснюють його, але жодна з них при більш детальному розгляді не дає повної та обґрунтованої відповіді на поставлене запитання. У другу половину століття людство винайшло електронні мікросхеми, що послужило поштовхом до розвитку електроніки та винаходу найрізноманітніших побутових та промислових приладів. Також було винайдено річ, яка кардинально перевернула все наше життя, а саме — персональний комп'ютер. Зараз майже жоден житель нашої планети не може уявити своє життя без цього вірного помічника.

Важливо також відзначити, що у зв'язку зі стрімкістю розвитку технологій зараз дуже проблематично встежити за всіма технологічними новинками, що з'являються на світовому ринку. Більше того, вже мало хто зможе розібратися, скажімо, яка модель ноутбука з величезної кількості існуючих варіантів є найякіснішою, надійнішою та має більш високу швидкість. Адже від того, як швидко працює ноутбук або персональний комп'ютер, багато що залежить. Наприклад, при грі на віртуальній біржі Форекс однією з головних вимог успіху є наявність потужного комп'ютера, а також якісної лінії зв'язку. Крім персональних комп'ютерів і ноутбуків, численні компанії світу випускають різну периферійну техніку, і в її різноманітності також зможе розібратися далеко не кожна людина.

Мікропроцесор - процесор (пристрій, що відповідає за виконання арифметичних, логічних операцій та операцій управління, записаних у машинному коді), реалізований у вигляді однієї мікросхеми або комплекту з декількох спеціалізованих мікросхем (на відміну від реалізації процесора у вигляді електричної схеми на елементній базі загального призначення або як програмної моделі).

					ЕЛІТ 8.171.00.10.376 ПЗ	Лист
Вим.	Лист	№ докум.	Підпис	Дата		4

До складу мікропроцесора входять: арифметико-логічний пристрій, блок управління та синхронізації, запам'ятовуючий пристрій, регістри, шини передачі даних та команд. Деякі автори відносять до мікропроцесорів лише пристрої, реалізовані лише на мікросхемі. Таке визначення розходиться як з академічними джерелами, і з комерційної практикою (наприклад, варіанти мікропроцесорів Intel і AMD у корпусах типу SECC і таких, як Pentium II, було реалізовано кількох мікросхемах).

Нині, у зв'язку з дуже незначним поширенням процесорів, які є мікропроцесорами, у побутової лексиці терміни «мікропроцесор» і «процесор» практично рівнозначні. У той же час, мікропроцесор зазвичай не має інтегрованих у мікросхему пристроїв введення-виведення, таймерів та інших периферійних пристроїв, чим відрізняється від мікроконтролера.

Мікроконтролер (англ. Micro Controller Unit, MCU) - мікросхема, призначена для управління електронними пристроями. Типовий мікроконтролер поєднує одному кристалі функції процесора і периферійних пристроїв, містить ОЗУ і (або) ПЗУ. По суті це однокристальний комп'ютер, здатний виконувати відносно прості завдання. Відрізняється від мікропроцесора інтегрованими мікросхему пристроями вводу-виводу, таймерами та іншими периферійними пристроями.

Важливим напрямом застосування мікроконтролерів є побудова пристроїв та систем вимірювання електричних параметрів схем та електронних компонентів.

Вимірювачі індуктивності (L) та ємності (C) знаходять широке застосування на виробництві, в лабораторіях технічних навчальних закладів, домашньому використанні. Як правило, вони мають вигляд малогабаритних приладів та систем, що мають низьку точність або цифрових приладів та систем, що мають високу точність вимірювання і при цьому великі габарити [1].

Актуальність теми полягає саме в розробці програмно-управляючої системи вимірювання електричних параметрів із застосуванням мікроконтролера, який має баланс якості, ціни та розмірів. Зменшення розмірів даної системи є можливим завдяки доцільності застосування індикатора, що здійснює послідовне введення інформації.

Принцип роботи проектованої системи полягає в вимірюванні енергії, яка накопичується в електричному полі конденсатора і магнітному полі котушки.

					ЕЛІТ 8.171.00.10.376 ПЗ	Лист
Вим.	Лист	№ докум.	Підпис	Дата		5

В даній конструкції міститься мікроконтролер і ЖКІ індикатор, що спонукає створенню простої, малогабаритної системи, яка є економічною та зручною в застосування, здійснює вимірювання з високою точністю. В даній системі передбачена програмна корекції нуля, що здійснює компенсацію ємності та індуктивності клем та дротів, що виконують функцію з'єднання [2].

Для роботи даної системи достатньо підключити вимірюваний елемент і зняти показання з індикатора.

Метою даної кваліфікаційної роботи магістра є обґрунтування та створення програмно-управляючої система вимірювання електричних параметрів , що здійснює вимірювання індуктивності та ємності. В теперішній час випускається велика кількість типів мікро-ЕОМ з необхідними функціональними можливостями та вартістю, прикладом яких є спеціалізовані прилади та системи автоматизованого керування технологічними об'єктами [2].

					ЕЛІТ 8.171.00.10.376 ПЗ	Лист
Вим.	Лист	№ докум.	Підпис	Дата		6

1 ОГЛЯД ЛІТЕРАТУРИ ТА ПОСТАНОВКА ЗАВДАННЯ ПРОЕКТУВАННЯ

1.1 Огляд літератури

Мікропроцесорна система є функціонально закінченим виробом, що складається з одного або декількох пристроїв, головним чином з мікропроцесора та/або мікроконтролера, являє собою електронну систему, яка призначена для обробки вхідних сигналів і видачі вихідних сигналів.

Мікропроцесорна система складається з наступних компонентів:

1. Генератор тактових імпульсів, який є одиницею вимірювання (Шергіном) тривалості виконання команди. Чим вища частота, тим за інших рівних умов швидкодійнішою є МПС.
2. Мікропроцесор / мікроконтролер. Мікропроцесор - процесор (пристрій, що відповідає за виконання арифметичних, логічних операцій та операцій управління, записаних у машинному коді), реалізований у вигляді однієї мікросхеми або комплекту з декількох спеціалізованих мікросхем (на відміну від реалізації процесора у вигляді електричної схеми на елементній базі загального призначення або як програмної моделі).
3. Запам'ятовувачі (ОЗУ, ПЗУ) - це невід'ємні частини системи. Інтерфейси введення та виведення - пристрої сполучення МПС з блоками введення та виведення інформації. Усі блоки МПС пов'язані між собою шинами передачі цифрової інформації. У МПС використовують магістральний принцип зв'язку, у якому блоки обмінюються інформацією з єдиної шини даних. Кількість ліній у шині даних зазвичай відповідає розрядності МПС (кількості біт у слові даних).
4. Пристрої введення-виводу, периферійні пристрої.
5. Шини. Шина адреси застосовується для вказівки напряму передачі даних - по ній передається адреса осередку пам'яті або блоку вводу-виводу, які отримують або передають інформацію в даний момент. Шина управління служить передачі сигналів, синхронізуючих всю роботу МПС.

В якості вхідних і вихідних сигналів при цьому можуть використовуватися аналогові сигнали (вхідні аналогові сигнали перетворюються в послідовності кодів вибірок за допомогою АЦП, вихідні аналогові сигнали формуються з

					ЕЛІТ 8.171.00.10.376 ПЗ	Лист
Вим.	Лист	№ докум.	Підпис	Дата		7

послідовності кодів вибірок за допомогою ЦАП), поодинокі цифрові сигнали, цифрові коди, послідовності цифрових кодів.

У середині цифрової системи здійснюється зберігання, накопичення сигналів (або інформації). Обробка та зберігання інформації здійснюється в цифровому вигляді. У цифровій системі алгоритми обробки та зберігання інформації тісно пов'язані з схемотехнікою (зміна алгоритмів можлива лише методом зміни структури системи, заміни електронних вузлів, що входять в систему та зв'язків між ними). Цифрова система називається системою на "жорсткій логіці", що являє собою спеціалізовану систему, яка налаштована одну чи декілька встановлених задач.

Цифрові системи мають наступні переваги:

1. Відсутність апаратної надмірності, що полягає в ефективній роботі кожного з елементів.
2. Максимальна швидкість роботи (швидкість виконання алгоритмів обробки інформації так як логічні елементи мають максимальну швидкість).

Цифрові системи мають наступні недоліки:

3. Необхідність проектування і виготовлення системи для кожної нової задачі.
4. Тривалість виготовлення та висока вартість.

З вищеприведеного бачимо, що актуальним є питання створення системи, здатної адаптуватися до будь-якого завдання, переходу з одного алгоритму роботи на інший з постійним складом апаратури. Задавати алгоритм роботи такої системи можливо шляхом введення програми. Дана система матиме властивість універсальності. Дані критерії забезпечить проектована мікропроцесорна система.

Метою використання універсальної системи є забезпечення вирішення поставлених завдань, при чому у разі вирішення простого завдання система працюватиме, використовуючи не всі свої ресурси, тому збільшення надмірності, робить не виправданою універсальність системи, тому в даному разі актуальним є зниження швидкості універсальних систем. Оптимізація універсальної системи для вирішення кожної задачі з максимальною швидкістю не є можливою, так як зі зростанням універсальності, швидкість зменшується.

Для управління простими технологічними об'єктами застосовуються прості мікро-ЕОМ з малим обсягом пам'яті , які виробляються на одному кристалі в вигляді однієї мікросхеми і називаються мікроконтролерами.

					ЕЛІТ 8.171.00.10.376 ПЗ	Лист
Вим.	Лист	№ докум.	Підпис	Дата		8

Мікроконтролер (англ. Micro Controller Unit, MCU) - мікросхема, призначена для управління електронними пристроями.

Типовий мікроконтролер поєднує одному кристалі функції процесора і периферійних пристроїв, містить ОЗУ і (або) ПЗУ. По суті це однокристальний комп'ютер, здатний виконувати відносно прості завдання.

Відрізняється від мікропроцесора інтегрованими в мікросхему пристроями вводу-виводу, таймерами та іншими периферійними пристроями.

При проектуванні мікроконтролерів доводиться дотримуватися компромісу між розмірами та вартістю з одного боку та гнучкістю та продуктивністю з іншого. Для різних програм оптимальне співвідношення цих та інших параметрів може відрізнятися дуже сильно. Тому існує безліч типів мікроконтролерів, що відрізняються архітектурою процесорного модуля, розміром і типом вбудованої пам'яті, набором периферійних пристроїв, типом корпусу і т.д.

На відміну від звичайних комп'ютерних мікропроцесорів, мікроконтролери часто використовують гарвардську архітектуру пам'яті, тобто роздільне зберігання даних в ОЗУ, а команд — у ПЗУ.

Крім ОЗУ, мікроконтролер може мати вбудовану енергонезалежну пам'ять для зберігання програми та даних. Багато моделей контролерів взагалі немає шин для підключення зовнішньої пам'яті.

Найбільш дешеві типи пам'яті допускають лише одноразовий запис, або програма записується в кристал на етапі виготовлення (конфігурацією набору технологічних масок). Такі пристрої підходять для виробництва в тих випадках, коли програма контролера не буде оновлюватися. Інші модифікації контролерів мають можливість багаторазового перезапису програми в енергонезалежній пам'яті.

Неповний список периферійних пристроїв, які можуть використовуватися в мікроконтролерах, включає:

- універсальні цифрові порти, які можна налаштовувати як у введення, і на вивод;
- різні інтерфейси введення-виводу, такі як UART, I²C, SPI, CAN, USB, IEEE 1394, Ethernet;
- аналого-цифрові та цифро-аналогові перетворювачі;
- компаратори;

					ЕЛІТ 8.171.00.10.376 ПЗ	Лист
Вим.	Лист	№ докум.	Підпис	Дата		9

- широтно-імпульсні модулятори (ШІМ-контролер);
- таймери;
- контролери безколекторних двигунів, у тому числі крокових;
- контролери дисплеїв та клавіатур;
- радіочастотні приймачі та передавачі;
- масиви вбудованої флеш-пам'яті;
- вбудовані тактовий генератор та сторожовий таймер.

Обмеження за ціною та енергоспоживанням обмежує тактову частоту контролерів. Хоча виробники прагнуть забезпечити роботу своїх виробів на високих частотах, вони водночас надають замовникам вибір, випускаючи модифікації, розраховані на різні частоти та напруги живлення. У багатьох моделях мікроконтролерів використовується статична пам'ять для ОЗП та внутрішніх регістрів. Це дає можливість контролеру працювати на менших частотах і навіть не втрачати дані при повній зупинці тактового генератора. Часто передбачені різні режими енергозбереження, в яких відключається частина периферійних пристроїв та обчислювальний модуль [3].

Відомі сімейства:

- MCS 51 (Intel)
- ESP8266 и ESP32 (Espressif)
- MSP430 (TI)
- ARM (ARM Limited)
 - ST Microelectronics STM32 ARM-based MCUs
 - ARM Cortex, ARM7 и ARM9-based MCUs
 - Texas Instruments Stellaris MCUs
 - NXP ARM-based LPC MCUs
 - Toshiba ARM-based MCUs
 - Analog Devices ARM7-based MCUs
 - Cirrus Logic ARM7-based MCUs
 - Freescale Semiconductor ARM9-based MCUs
 - Silicon Labs EFM32 ARM-based MCUs
- AVR (Atmel)

					ЕЛІТ 8.171.00.10.376 ПЗ	Лист
Вим.	Лист	№ докум.	Підпис	Дата		10

Відомі компілятори Сі для МК:

- GNU Compiler Collection - підтримує ARM, AVR, MSP430 та багато інших архітектур
- Small Device C Compiler - підтримує безліч архітектур
- CodeVisionAVR (для AVR)
- IAR (для будь-яких МК)
- WinAVR (для AVR та AVR32)
- Keil (для архітектури 8051 та ARM)
- HiTECH (для архітектури 8051 та PIC від Microchip)

Відомі компілятори бейсика для МК:

- MikroBasic (архітектури PIC, AVR, 8051 та ARM)
- Bascom (архітектури AVR та 8051)
- FastAVR (для архітектури AVR)
- PICBasic (для архітектури PIC)
- Swordfish (для архітектури PIC)

Для налагодження програм використовуються програмні симулятори (спеціальні програми для персональних комп'ютерів, що імітують роботу мікроконтролера), внутрішньосхемні емулятори (електронні пристрої, що імітують мікроконтролер, які можна підключити замість нього до вбудованого пристрою, що розробляється) і налагоджувальний інтерфейс, наприклад, JTAG.

Застосування "жорсткої" логіки в мікропроцесорних пристроях і системах виправдане у двох основних випадках:

- 1) за необхідності отримання гранично високої швидкодії;
- 2) при побудові щодо нескладних пристроїв на інтегральних схемах малого та середнього ступеня інтеграції.

Пристрої та системи з жорсткою структурою зазвичай містять велику кількість інтегральних схем (ІС) малого та середнього ступеня інтеграції. Ці схеми встановлюються на платах, які висновки з'єднуються відповідно до реалізованими функціями. Характерна риса традиційної цифрової системи полягає в тому, що алгоритми обробки та зберігання інформації в ній жорстко пов'язані із схемотехнікою системи. Тобто зміна цих алгоритмів можлива лише шляхом зміни структури системи, заміни електронних вузлів, що входять до системи та/або зв'язків між ними. Наприклад, якщо нам потрібна додаткова

						ЕЛІТ 8.171.00.10.376 ПЗ	Лист
Вим.	Лист	№ докум.	Підпис	Дата			12

операція підсумовування, необхідно додати до структури системи зайвий суматор. Або якщо потрібна додаткова функція зберігання коду протягом одного такту, ми повинні додати структуру ще один регістр. Звичайно, це практично неможливо зробити в процесі експлуатації, обов'язково потрібен новий виробничий цикл проектування, виготовлення, налагодження всієї системи. Саме тому традиційна цифрова система часто називається системою на "жорсткій логіці".

Останнім часом актуальним є застосування програмованих систем, призначенням яких є вирішення однієї або декількох з близьких задач (ПЛІС - програмовані логічні інтегральні мікросхеми), яким властиві переваги цифрових та програмованих систем, серед яких є висока швидкодія і необхідна гнучкість.

Програма- це набір команд (цифрових кодів) , які визначають дії процесора (після розшифровки).

Всі пристрої мікропроцесорної системи об'єднуються загальною системною шиною (магістраль).

Системна магістраль включає в себе чотири основні шини нижнього рівня:

- Шина адреси (Address Bus) - служить для визначення адреси (номеру) пристрою, з яким процесор здійснює обмін інформацією в даний момент, бувають односпрямовані та двобічні;
- Шина даних (Data Bus) - це основна шина, необхідна для передачі інформаційних кодів між усіма пристроями мікропроцесорної системи, завжди двонаправлена;
- Шина управління (Control Bus) складається з окремих керуючих сигналів, кожен з яких під час обміну інформацією має свою функцію. (наприклад, стробування переданих або отриманих даних- визначення моментів часу подання інформаційного коду на шину даних), лінії одно-, дво- направлені;
- Шина живлення (Power Bus) - необхідна для живлення системи, яка містить лінії живлення і загальний провід.

У мікропроцесорної системі може бути одне джерело живлення (частіше +5 В) або декілька (зазвичай ще -5 В, +12 В і -12 В).

					ЕЛІТ 8.171.00.10.376 ПЗ	Лист
Вим.	Лист	№ докум.	Підпис	Дата		13

1.2 Постановка завдання проектування

В даній кваліфікаційній роботі магістра актуальним є питання розгляду вимірювачів індуктивності і ємності, які являють собою малогабаритні прилади, що мають низьку точність, або цифрові прилади великих габаритів і високої точності.

Розробка, що розглядається в даній роботі, містить мікроконтролер, з балансом якості і габаритів, що є можливим за рахунок використання індикатора з послідовним введенням інформації.

Принцип роботи вимірювача LC полягає в вимірюванні енергії, що накопичується в електричному полі конденсатора і магнітному полі котушки.

Застосування в даній конструкції мікроконтролеру і ЖК індикатора дозволяє отримати простий, малогабаритний, економічний та простий в експлуатації прилад з високою точністю вимірювання.

В даній системі передбачена програмна корекція нуля, що дозволяє компенсувати ємність індуктивності клем, з'єднувальних дротів і перемикача.

Для проектуваного вимірювача ємності було поставлено наступні вимоги: прямий відлік та лінійна шкала, широкий динамічний діапазон, мала похибка вимірювань, висока надійність.

Основні технічні характеристики проекрованої системи:

- 1) діапазон вимірюваних ємностей від 1 пФ до 1000 мкФ розбитий на 12 піддіапазонів (6 положень перемикача з двома положеннями перемикача множника відліку);
- 2) похибка вимірювання, при використанні:
 - індикатора стрілочного вимірювача класу точності 0,5 – не більше 2%,
 - цифрового вольтметра – 1,5%;
 - час встановлення показників на кожному з піддіапазонів – не більше 30 с;
 - дрейф початкової установки стрілки вимірювача за час роботи не менше 2 годин при зміні температури навколишнього середовища – від +10 до +30 ° С на всіх межах вимірювання не більше 1,5 поділки шкали;

						ЕЛІТ 8.171.00.10.376 ПЗ	Лист
Вим.	Лист	№ докум.	Підпис	Дата			14

- лінійність шкали приладу не гірше 1% (обмежена лінійністю застосовуваного індикатора);
- споживана потужність від мережі змінного струму – не більше 5 Вт;
- габарити системи – 236x155x100 мм; маса – 2 кг.



Рисунок 1.1– Зовнішній вигляд проектованої системи

Принцип роботи проектованої системи характеризується зарядкою вимірюваного конденсатора від джерела сигналу змінного струму стабільної форми, амплітуди і частоти з подальшим диференціюванням, виділенням постійної складової на синхронному детекторі і фіксацією її вимірювальними системами [3].

Основа конструкції – передня панель, на внутрішній стороні якої розміщена друкована плата та блок живлення, а з зовнішньої сторони встановлений мікроамперметр, перемикачі та гнізда для підключення цифрового вольтметра.

Для під'єднання вимірювача конденсаторів застосовують гнізда чи нероз'ємні виводи з ізольованих провідників .

Проектована система компактна та зручна при практичному використанні. На відміну від багатьох інших, цей вимірювач спеціально спроектований для вимірювання малих значень ємності.

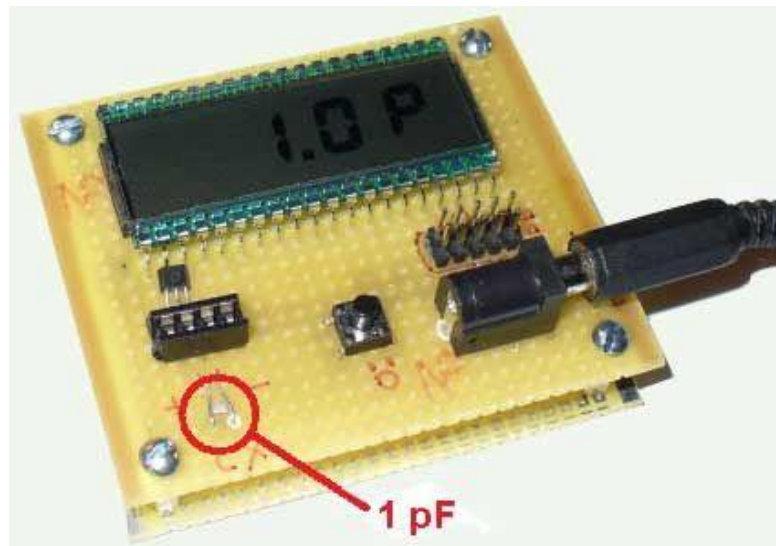


Рисунок 1.2– Вимірювач індуктивності та ємності

При проектуванні даної системи необхідно врахувати наступні її параметри:

- схемне вирішення та вибір елементної бази;
- поєднання вимірюваних х параметрів (індуктивність, ємність);
- діапазон вимірювання;
- додаткові можливості;
- похибка вимірювання;
- габарити;
- ціна.

Дана система має наступні додаткові можливості та застосування:

- зв'язок системи з комп'ютером для документування результатів вимірювання , можливість програмної обробки результатів;
- можливість вимірювання індуктивності (рідкість, що дана система вимірює різні параметри, як правило вимірювач індуктивності-окремий прилад);
- можливість сортування і відбракування компонентів.
- система може бути використана як генератор фіксованих частот з декадних поділом, а також як генератор позначок для радіотехнічних вимірювальних приладів.
- діапазон вимірювань розбитий подекадно на шість піддіапазонів з граничними значеннями ємності 100 пФ – 10 мкФ для конденсаторів й індуктивності 10 мкГн – 1 Гн для котушок індуктивності.

										Лист
										16
Вим.	Лист	№ докум.	Підпис	Дата						

2 ОБГРУНТУВАННЯ АЛГОРИТМУ РОБОТИ І СТРУКТУРНОЇ СХЕМИ ПРОГРАМНО-УПРАВЛЯЮЧОЇ СИСТЕМИ ВИМІРЮВАННЯ ЕЛЕКТРИЧНИХ ПАРАМЕТРІВ

2.1 Розробка алгоритму проекрованої системи

Мікропроцесорні системи на основі мікроконтролера часто мають застосування вбудованих систем для вирішення завдань керування об'єктом, отримали назву контролерів. Характерною для даного застосування є робота в реальному часі, що являє собою забезпечення певної реакції на зовнішні події в заданому часовому інтервалі.

Технологія проектування контролерів на базі мікроконтролера використовує принцип нерозривного проектування, налагодження апаратних і програмних засобів. При використанні даної технології, перед виробником стоїть задача реалізації повного циклу проектування, що починається розробкою алгоритму функціонування та закінчується комплексними іспитами виробу, а іноді супроводом при виробництві системи.

Технічне завдання містить вимоги до контролера з погляду реалізації визначеної функції управління, має вигляд текстового опису.

На основі побажань користувача складається функціональна специфікація, яка містить функції, що виконуються контролером після завершення проектування, визначаючи наскільки проектована система відповідає пропонованим вимогам.

Етап розробки алгоритму управління є найбільш важливим, поєднує декілька можливих варіанти алгоритмів, які відрізняються кількістю програмного забезпечення й апаратних засобів.

Максимальне використання апаратних засобів робить більш простим процес розробки та збільшує швидкодію контролера, що впливає на збільшення вартості і споживаної потужності. Збільшення об'єму апаратних засобів здійснюється за рахунок того, що обирається більш складний мікроконтролер, та використання спеціалізованих інтерфейсних схем. Дані фактори впливають на зростання вартості й енергоспоживання [4].

Зростання обсягу програмного забезпечення впливає на зниження кількості елементів контролера та вартості апаратних засобів, що впливає на зниження швидкодії, можливості зростання об'єму внутрішньої пам'яті мікроконтролеру, і

					ЕЛІТ 8.171.00.10.376 ПЗ	Лист
Вим.	Лист	№ докум.	Підпис	Дата		17

це впливає на розширення термінів розробки та строку налагодження програмного забезпечення. Важливою є можливість максимальної реалізації необхідних функцій програмними засобами при мінімальних апаратних витратах з дотриманням необхідної швидкодії та забезпечення. Прикладом вимог, які необхідні для дотримання є можливість захисту інформації (з використанням програмного коду) контролера, та дотримання максимально можливої тривалості роботи в автономному режимі. У результаті виконання даного етапу є наявність певних вимог, які висуваються до параметрів мікроконтролера, що використовується.

При виборі типу мікроконтролеру, що використовується для створення проектованої системи враховуються наступні характеристики:

- розрядність МК;
- швидкодія МК;
- набір команд і способів адресації МК;
- вимоги до джерела живлення МУ;
- споживана потужність у різних режимах МК;
- об'єм ПЗП програм і ОЗП даних, можливість їх розширення;
- наявність і можливості периферійних пристроїв, засоби підтримки роботи в реальному часі (таймери, процесори подій і т.п.);
- можливість перепрограмування МК в складі системи;
- засоби захисту внутрішньої інформації МК;
- можливість застосування в різних варіантах конструктивного виконання;
- вартість;
- наявність повної документації;
- наявність і доступність ефективних засобів програмування і налагодження МК;
- можливість заміни виробами інших фірм.

Список характеристик може бути розширеним, так як специфіка проектованої системи може зробити актуальними інші необхідні параметри мікроконтролеру.

Різноманітність фірм виробників, що випускають МК в наш час, відкриває перед виробником можливість вибору оптимального мікроконтролеру, що не містить функціональної надлишковості, а тим самим мінімізує вартість комплектуючих елементів.

						ЕЛІТ 8.171.00.10.376 ПЗ	Лист
Вим.	Лист	№ докум.	Підпис	Дата			18

Допущені на етапі створення алгоритму недоліки можуть призвести до необхідності заміни моделі і здійснення повторного розведення друкованої плати макета контролера, отже доцільним є моделювання основних елементів прикладної програми з використанням програмно-логічної моделі обраного мікроконтролеру.

В вигляду на те, що пріоритетною вимогою є зниження собівартості готової серійної електронної системи, мною здійснюється розробка системи, що містить бюджетний мікроконтролер типу сучасного сімейства мікроконтролерів Atmel AVR, Microchip PIC і Holtek, і це дозволяє оптимізувати розробку та виготовлення електронної системи з необхідною ціною.

Кожен із зазначених мікроконтролерів має деякі особливості, що дозволяють зробити вибір певного типу мікроконтролера для розробки нового електронного пристрою, здатного відповідати поставленим вимогам замовника.

Для розробки системи, що запрограмована на мові C, доцільним є використання мікроконтролеру ATtiny45, що є компактним та економічним, підходящим для виконання невеликих програм. ATtiny85 має необхідний об'єм пам'яті. Працює при живленні 1.8 Вольт до 5.5 Вольт. В разі необхідності використання більш економічного та компактного мікроконтролеру, ніж плата Arduino, що не передбачає підключення великої кількості компонентів, то використання ATtiny45 буде оптимальним.

Arduino IDE- це програмне забезпечення, яке необхідне для програмування плати Arduino. За замовчуванням програмування мікроконтролерів ATtiny не підтримується, але це виправляється завдяки проекту з відкритим вихідним кодом Arduino-Tiny, що розширює список мікроконтролерів, які можна запрограмувати. Скачайте версію, яка відповідає версії встановленої Arduino IDE (версію можна побачити в заголовку вікна).

Інструкція щодо установки Arduino-Tiny:

- Запустити Arduino IDE.
- Перейти до папки, в якій зберігаються скетчі для Arduino. Місцезнаходження цієї папки легко перевірити шляхом завантаження Arduino IDE і вибором в меню "Файл → Зберегти як ...". У діалоговому вікні буде відкрита шукана папка: "C: \ Projects \ Arduino \". Після чого треба закрити Arduino IDE.
- Необхідно переконатися, що в папці зі скетчами існує папка "hardware".

						ЕЛІТ 8.171.00.10.376 ПЗ	Лист
Вим.	Лист	№ докум.	Підпис	Дата			19

Якщо такої папки немає, але необхідно її створити. Якщо папка з скетчами "C: \ Projects \ Arduino \", то результат повинен бути наступним: "C: \ Projects \ Arduino \ hardware \"

- Розпакувати архів, що завантажений з сайту Arduino-Tiny в папку "hardware". Якщо папка з скетчами "C: \ Projects \ Arduino \", то структура файлів має :

"C: \ Projects \ Arduino \ hardware \ tiny \ license.txt"

"C: \ Projects \ Arduino \ hardware \ tiny \ Prospective Boards.txt"

"C: \ Projects \ Arduino \ hardware \ tiny \ README"

"C: \ Projects \ Arduino \ hardware \ tiny \ bootloaders"

"C: \ Projects \ Arduino \ hardware \ tiny \ cores"

Папка, що містить вихідні файли для ядра Arduino-Tiny:

"C: \ Projects \ Arduino \ hardware \ tiny \ cores \ tiny"

- Створення текстового файлу "boards.txt" в папці "tiny". Файл знаходиться за наступним посиланням:

"C: \ Projects \ Arduino \ hardware \ tiny \ boards.txt".

- Відкрити створений файл "boards.txt", розмістити в ньому вміст файлу "Prospective Boards.txt". Здійснити збереження модифікованого файлу "boards.txt".
- Здійснити запуск Arduino IDE і перевірити, щоб в підміню "Сервіс → Плата" були наявні мікроконтролери ATtiny.
- Найпростішим прикладом для завантаження на ATtiny84 / ATtiny45, що може бути використаний для перевірки роботи мікроконтролера є миготливий світлодіод. Даний скетч має назву Blink та може бути завантажений з "Файл → Приклади → 01.Basics".
- Для використання даного скетчу з ATtiny, треба здійснити зміну номеру ніжки, до якої буде підключатися світлодіод. Як правило використовується вивід під номером 13, але в МК серії ATtiny немає такої кількості доступних виводів, тому для проектованої системи буде застосовано вивід з номером 0. Спочатку проводиться ініціалізація цифрового виходу під номером 0, після чого в циклі повторюється подання високого значення на одну секунду, після чого здійснюється встановлення низького значення на одну секунду.

						ЕЛІТ 8.171.00.10.376 ПЗ	Лист
Вим.	Лист	№ докум.	Підпис	Дата			20

```

/* Blink
   Turns on an LED on for one second, then off for one second, repeatedly.
   This example code is in the public domain.
*/

// ATtiny84/ATtiny44 does not have Pin 13, so we use pin 0 instead.
// A current limiting resistor should be connected in line with the LED.
int led = 0;

// the setup routine runs once when you press reset:

void setup() {
  // initialize the digital pin as an output.
  pinMode(led, OUTPUT);
}

// the loop routine runs over and over again forever:
void loop() {
  digitalWrite(led, HIGH); // turn the LED on (HIGH is the voltage level)
  delay(1000);             // wait for a second
  digitalWrite(led, LOW);  // turn the LED off by making the voltage LOW
  delay(1000);             // wait for a second
}

```

Рисунок 2.1 – Приклад налаштування Arduino IDE для прошивки Attiny

Програма для завантаження на ATtiny отримана і далі треба здійснити підготовку Arduino до використання в якості програматора:

В підміню "Сервіс → Плата" обираємо мікроконтролер, який треба запрограмувати, як приклад, "Attiny45 @ 1Mhz (internal oscillator, BOD disabled)".

Схема підключення ATtiny84 / ATtiny45 до плати Arduino зображено на рисунку 2.2.

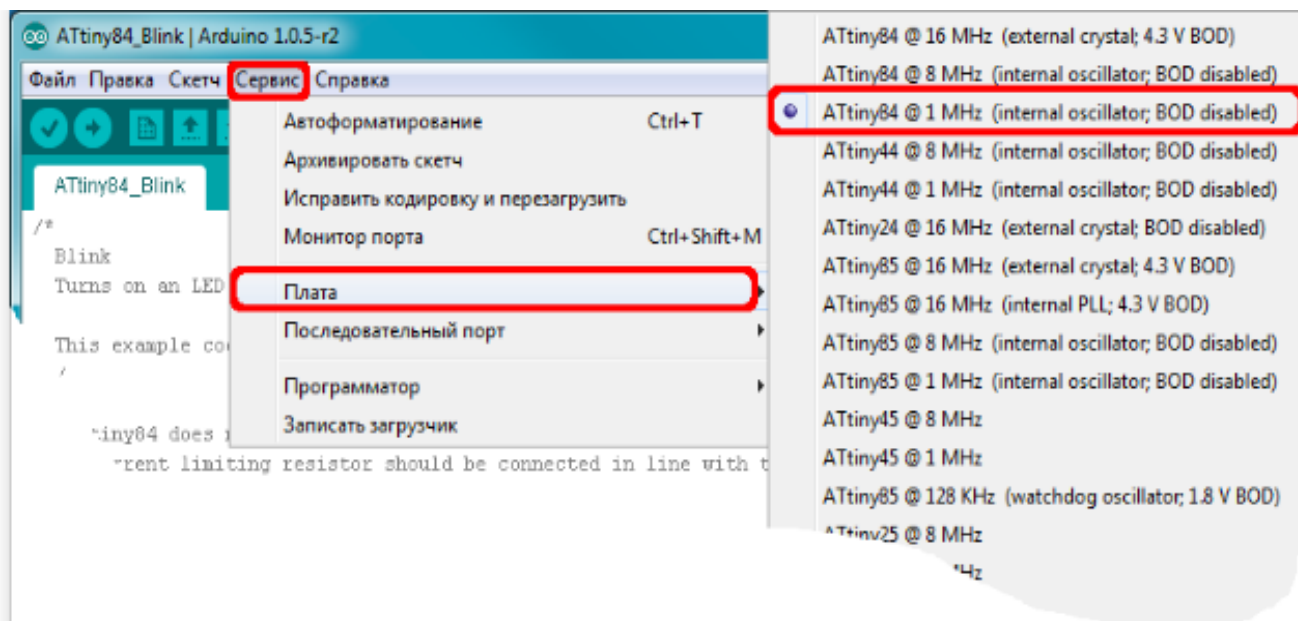


Рисунок 2.2 – Підготовка Arduino

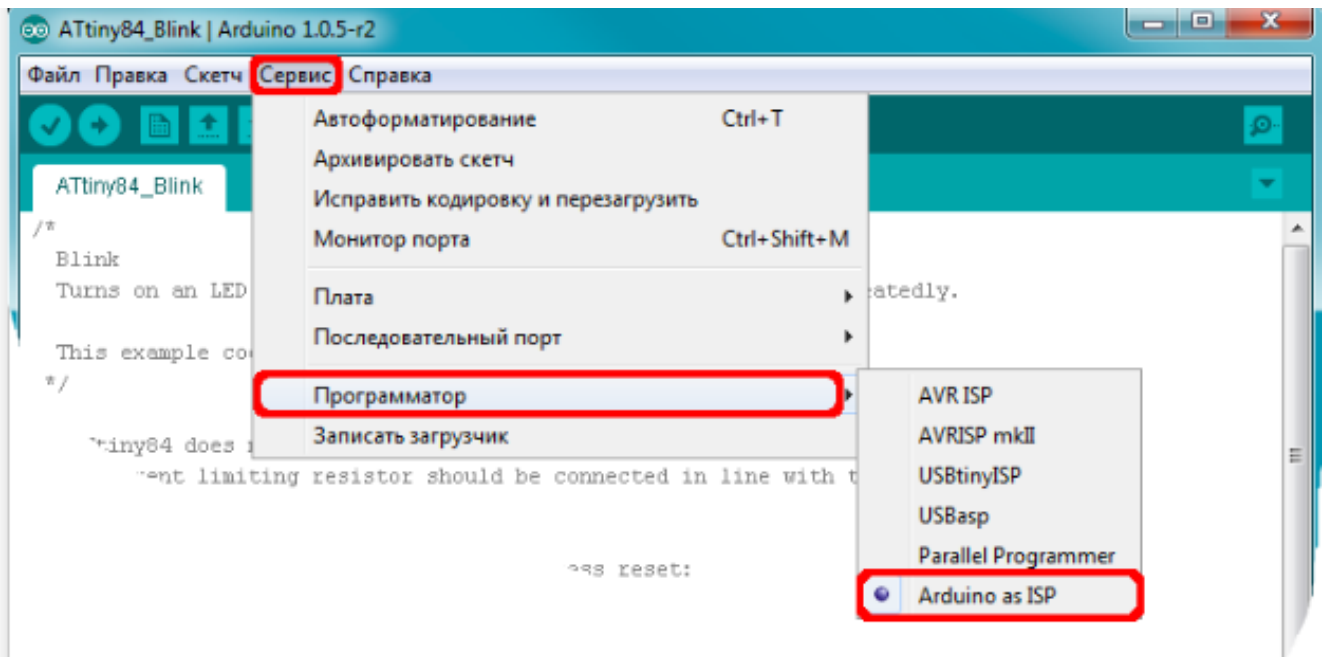


Рисунок 2.3 – Підготовка Arduino (Сервіс)

Для роботи з мікроконтролером треба розуміти призначення виходів. На рисунку 2.4 вказано, яким чином повинні бути підключені виводи плати Arduino до ATtiny45.

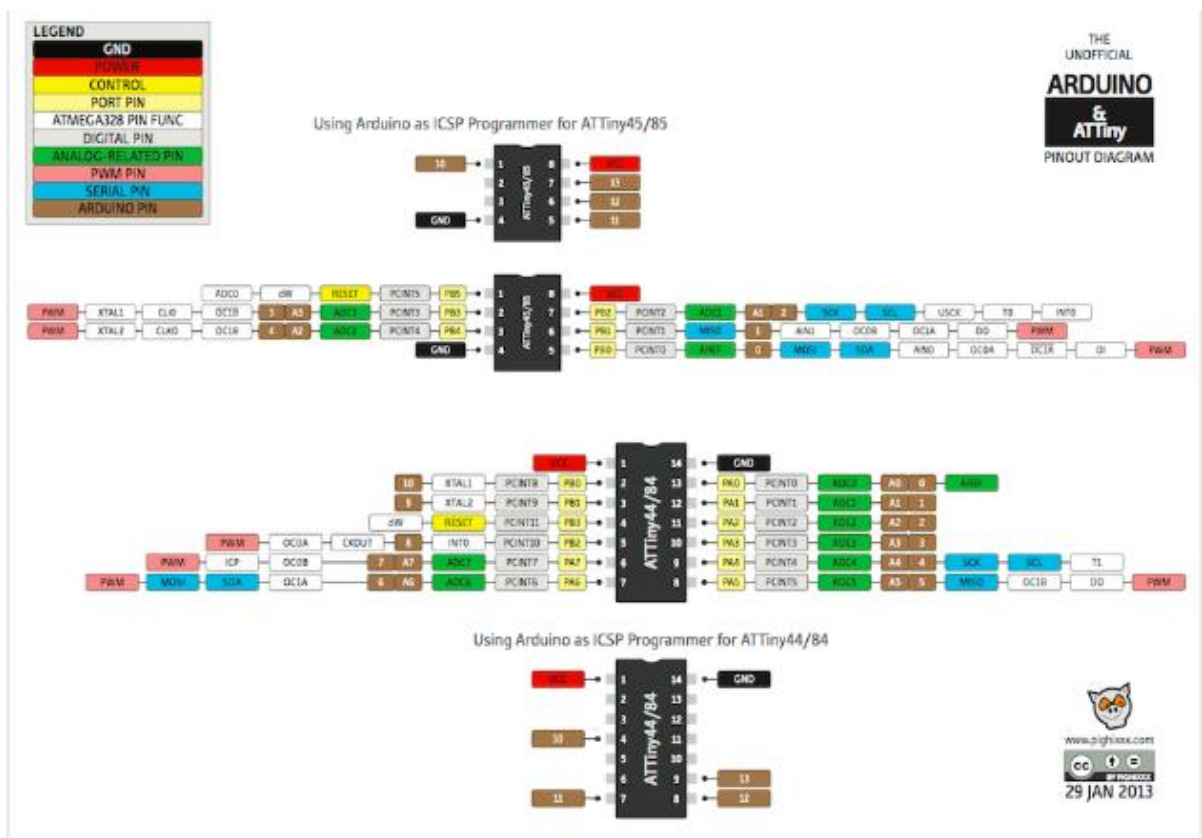


Рисунок 2.4 – Підготовка Arduino (Схема)

Після підготовки та настройки програмного забезпечення треба підключити апаратне забезпечення, для цього:

1. Відключити плату Arduino від джерела живлення, для уникнення можливого пошкодження плати.
2. Живлення (5V) від Arduino має бути підключене до ніжки під номером 1 мікроконтролера ATtiny45 - чорний провід.
3. Вивід 10 від Arduino повинен бути з'єднаний з виводом 4 (RESET) мікроконтролеру / ATtiny45 - зелений провід.
4. Вивід 11 від Arduino має бути з'єднаний з виводом 7 (MOSI) мікроконтролеру ATtiny45 - помаранчевий.
5. Вивід 12 від Arduino має бути з'єднаний з виводом 8 (MISO) мікроконтролеру ATtiny45 - білий провід.
6. Вивід 13 від Arduino має бути з'єднаний з виводом 9 (CLOCK) мікроконтролеру ATtiny45 - жовтий провід.
7. Вихід Земля (GND) від Arduino має бути з'єднаний з ніжкою під номером 14 мікроконтролеру / ATtiny45 - червоний провід.
8. Конденсатор ємністю 10мФ повинен з'єднувати вивід Земля (GND) і вивід Скидання (RESET), і це допоможе запобігти перезавантаження плати Arduino, гарантує, зв'язок Arduino IDE з ArduinoISP під час завантаження скетчу.
9. Підключення плати Arduino до комп'ютера за допомогою USB кабелю.

					ЕЛІТ 8.171.00.10.376 ПЗ	Лист
Вим.	Лист	№ докум.	Підпис	Дата		23

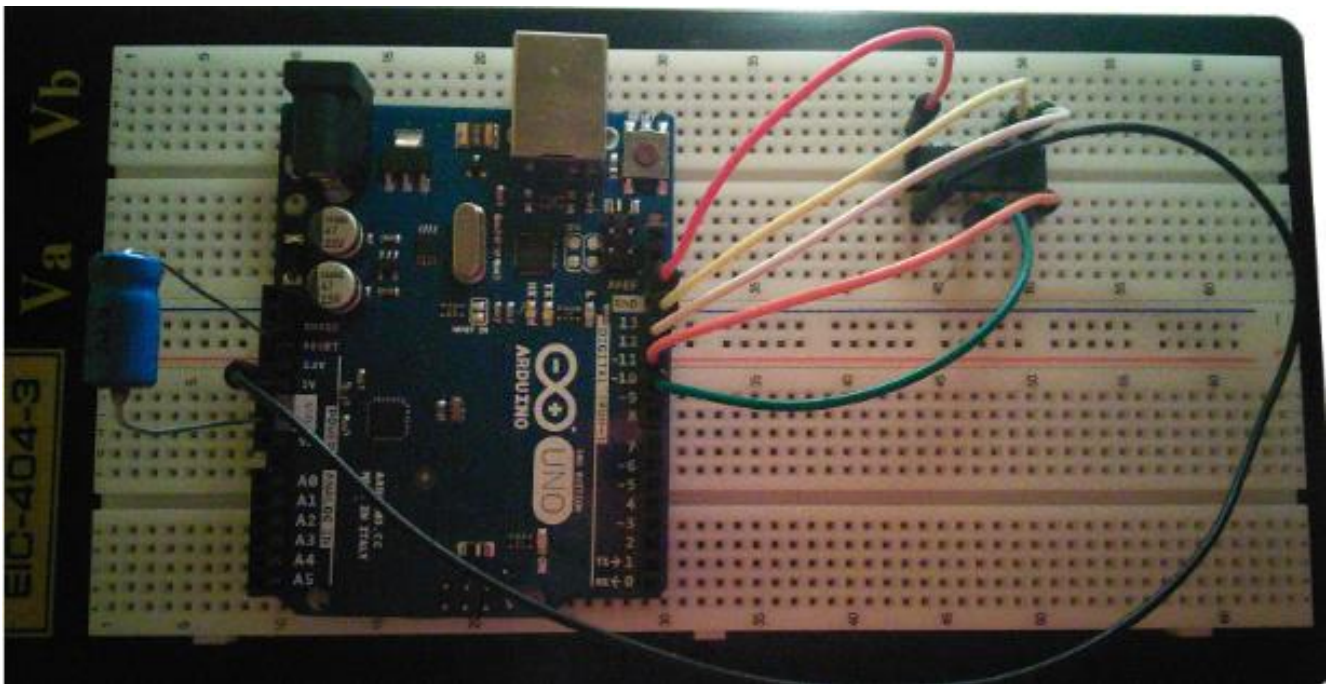


Рисунок 2.5 – Підключення Arduino

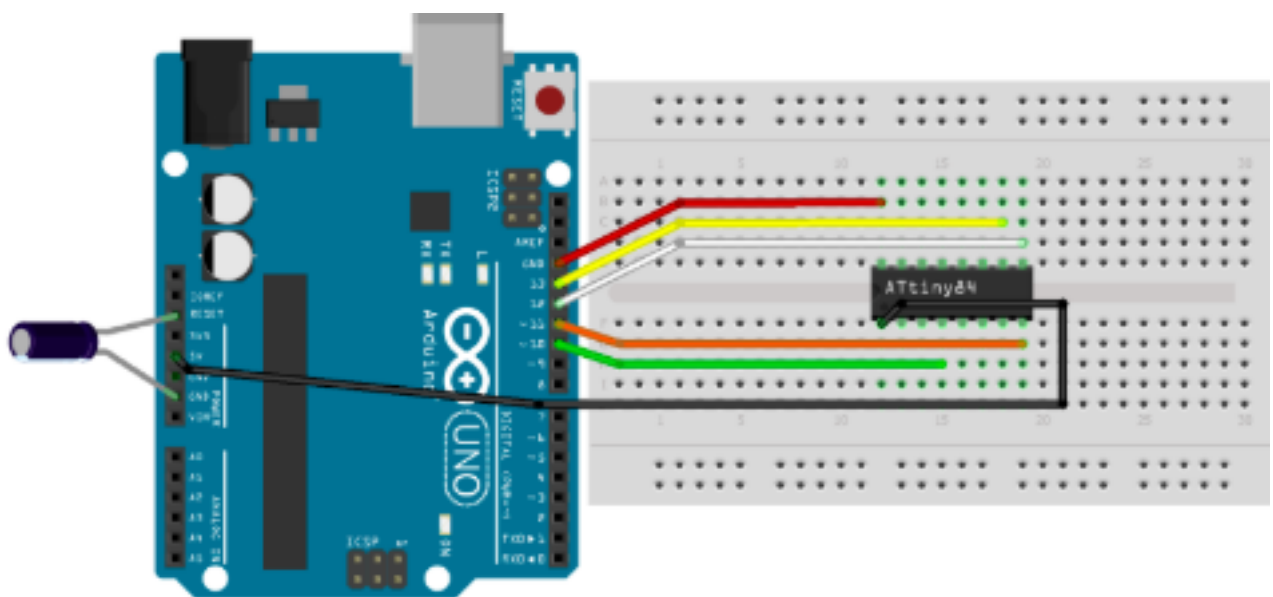


Рисунок 2.6 – Підключення Arduino (для завантаження скетчу)

Після завершення всіх необхідних підключень, можна виконати завантаження скетчу на мікроконтролер ATtiny (натиснути кнопку "Завантажити").

Якщо завантаження скетчу на ATtiny84 / ATtiny45 пройшла успішно, то можна приступити до тестування:

					ЕЛІТ 8.171.00.10.376 ПЗ	Лист
Вим.	Лист	№ докум.	Підпис	Дата		24

1. Вимкнути плату Arduino від живлення. Відповідно до завантажених скетчів, необхідно підключити анод (+) світлодіода до виходу під номером 2 мікроконтролера ATtiny45. Катод (-) світлодіода - коротка ніжка - повинен бути підключений до резистора (в даному випадку з опором 180 Ом, що буде виконувати функцію обмежувача струму. Інша ніжка резистора підключається до Землі (GND) плати Arduino.
2. Для живлення мікроконтролера використовується плата Arduino Uno, а саме вивід 3.3V, який підключено до виводу під номером 1 мікроконтролера ATtiny45.
3. Вивід Земля (GND) з плати Arduino повинен бути підключений до виходу під номером 14 мікроконтролера ATtiny45.
4. Підключення плати Arduino до живлення.

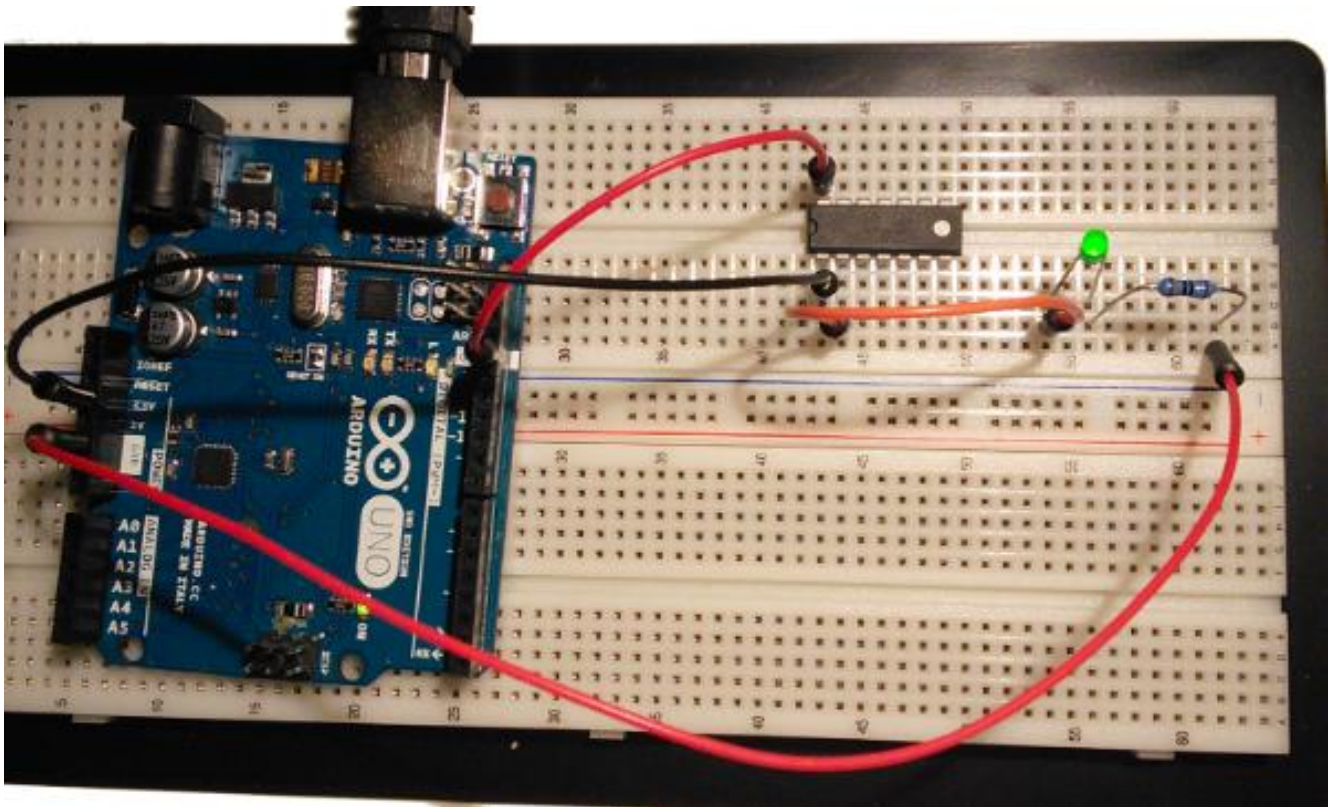


Рисунок 2.7 – Підключення Arduino (Кінцевий етап)

При роботі з мікроконтролерами ATtiny необхідно бути обережним, так як не всі бібліотеки можуть бути використані, треба перевірити форум Arduino на наявність інформації про сумісність і можливі проблеми.

У всіх мікроконтролерах сімейства mega є апаратний модуль UART, чого немає в мікроконтролерах сімейства tiny. Програмний UART може бути

						ЕЛІТ 8.171.00.10.376 ПЗ	Лист
Вим.	Лист	№ докум.	Підпис	Дата			25

використаний для молодшого сімейства.

Проекти написані як для ATtiny45 так і для ATmega16. Відмінності між ними незначні, стосуються ініціалізації та переривання задіяного таймера.

Важливим є створення заголовків softUart.h. Включення заголовків compilers_3.h, забезпечує сумісність з трьома AVR компіляторами (IAR, CodeVision, GNU GCC), макроозначеннями та прототипами функцій користувача. Прототипи, слідують своєму стилю, з використанням приставки ("SUART_"), що визначає до яких файлів віднести необхідні функції.

```
#ifndef SOFT_UART_H
```

```
#define SOFT_UART_H
```

```
#include "compilers_3.h"
```

```
/ * _____ * /
```

Налаштування

```
/ * Тактова частота мікроконтролера * /
```

```
#ifndef F_CPU
```

```
#define F_CPU 8000000L
```

```
#endif
```

```
/ * Прескалер таймеру 1L, 8L, 64L, 256L, 1024L * /
```

```
#define SUART_PRESCALER 1L
```

```
/ * Швидкість обміну даними * /
```

```
#define SUART_BAUD_RATE 19200L
```

```
/ * Місткість приймаючого буферу * /
```

```
#define SUART_BUF_SIZE 32
```

```
/ * Пін приймача * /
```

```
#define RX_PINX PINB
```

```
#define RX_PORTX PORTB
```

```
#define RX_DDRX DDRB
```

									Лист
									26
Вим.	Лист	№ докум.	Підпис	Дата				ЕЛІТ 8.171.00.10.376 ПЗ	

```

#define RX_PIN 1

/* Пін передавача */
#define TX_PORTX PORTB
#define TX_DDRX DDRB
#define TX_PIN 0

/* Функція або макрос, що виконується під час
очікування надходження даних в приймальний буфер.
SUART_GetChar () */
#define SUART_Idle ()

/* _____ Функції для _____ */

void SUART_Init (void);
char SUART_GetChar (void);
void SUART_PutChar (char ch);
void SUART_PutStr (char * str);
void SUART_PutStrFl (char __flash * str);
void SUART_FlushInBuf (void);
char SUART_Kbhit (void);
void SUART_TurnRxOn (void);
void SUART_TurnRxOff (void);

/
_____ */

#endif // SOFT_UART_H

```

В даній роботі було взято приклад ескізу Servo -> Sweep з заміною рядку, в якому міститься серво-змінна, з зміною номеру штиря, здійснене додавання декількох запитів до SoftwareServo :: refresh (), з урахуванням рекомендацій в документації Software Servo. Маємо наступний код:

```

#include <SoftwareServo.h>

```

						ЕЛІТ 8.171.00.10.376 ПЗ	Лист
Вим.	Лист	№ докум.	Підпис	Дата			27

```

SoftwareServo myservo; // створення сервооб'єкту для керування сервоприводом
// можливим є створення максимум 8 сервооб'єктів
int pos = 0; // змінна для збереження положення сервоприводу
void setup ()
{
myservo.attach (1); // прикріплення сервоприводу на контакті 1
}

void loop ()
{
для (pos = 0; pos <180; pos += 1) // йде від 0 - 180 градусів
{ // з кроком в 1 градус
myservo.write (пози); // передача сервокерування в положення в змінної 'pos'
затримка (15); // очікування 15 мс, щоб сервопривід досяг позиції
SoftwareServo :: оновлення ();
}
для (pos = 180; pos >= 1; pos = 1) // йде від 180 градусів до 0 градусів
{
myservo.write (пози); // передаємо сервокерування в положення в змінної 'pos'
затримка (15); // очікування 15 мс, щоб сервопривід досяг позиції
SoftwareServo :: оновлення ();
}
}

```

Для отримання коду на чіпі, доцільним є використання програмного забезпечення від MIT, що здійснює підтримку ATTiny 45 і ATTiny 85 в середовищі Arduino. Це робить можливим вибір ATTiny 85 з меню «Сервіс ->» в Arduino. Файли підтримки розташовані за наступним посиланням: <http://hlt.media.mit.edu/?p=1229>

З використанням програмного коду AVR, стало можливим завантаження прямо в 45, обравши Tools -> Board -> AT Tiny 45 (w USB Tiny ISP).

Для завантаження коду, треба натиснути кнопку «Перевірити», а потім натиснути кнопку «Завантажити».

Необхідним є підключення сервоприводу до 5 В та заземлення, перед підключенням контрольного проводу (жовтого / білого проводу, що йде від

									Лист
									28
Вим.	Лист	№ докум.	Підпис	Дата					

сервомотора), до контакту 6 на ATtiny 85 (PB1 - той самий вивід з жовтим кабелем MISO).

Макроозначення F_CPU, SUART_PRESCALER і SUART_BAUD_RATE використовуються в макросі для розрахунку таймерної константи, що забезпечує задану швидкість обміну по UART`у. Константа SUART_PRESCALER може приймати тільки ті значення, що записані в коментарі - 1L, 8L, 64L, 256L, 1024L. При заданні інших значень, компілятор буде видавати помилку, також якщо і при розрахунку таймерної константи будуть отримані неприпустимі значення - нуль і значення більше 256, і буде отримано попередження, в випадку коли швидкість програмного UART`а буде на 1% більше або менше ніж необхідне значення.

Також в заголовки біли додані ті визначення, які задають приймальний і передавальний вивід мікроконтролера і макроси для розрахунку і перевірки таймерної константи. Макроси дозволяють швидко підібрати оптимальне значення частоти мікроконтролера, переддільника таймера та швидкості передачі даних [6].

```
#define TIME_VALUE (F_CPU / (SUART_BAUD_RATE * NUM_OF_CYCLES *  
SUART_PRESCALER))  
#define TIMER_TCNT_VALUE (0xff - (TIME_VALUE - 1))  
#define ERROR_VALUE ((F_CPU * 100) / (TIME_VALUE * SUART_BAUD_RATE  
* NUM_OF_CYCLES * SUART_PRESCALER))  
#if ((TIME_VALUE > 256) || (TIME_VALUE == 0))  
  #error "TIMER_TCNT_VALUE is not correct. Change PRESCALER, BAUD_RATE  
or F_CPU"  
#endif  
  
#if ((ERROR_VALUE > 100) || (ERROR_VALUE < 100))  
  #warning "Baud rate error > 1% "  
#endif  
  
#if SUART_PRESCALER == 1L  
  #define CSXX (0 << CS02) | (0 << CS01) | (1 << CS00)  
#elif SUART_PRESCALER == 8L  
  #define CSXX (0 << CS02) | (1 << CS01) | (0 << CS00)  
#elif SUART_PRESCALER == 64L
```

					ЕЛІТ 8.171.00.10.376 ПЗ	Лист
Вим.	Лист	№ докум.	Підпис	Дата		29

```

#define CSXX (0 << CS02) | (1 << CS01) | (1 << CS00)
#elif SUART_PRESCALER == 256L
#define CSXX (1 << CS02) | (0 << CS01) | (0 << CS00)
#elif SUART_PRESCALER == 1024L
#define CSXX (1 << CS02) | (1 << CS01) | (1 << CS00)
#else
#error "prescaller not correct"
#endif

```

Функції void timer_set (int BAUD_RATE) і void set_timer_interrupt (timer_isr) було видалено, а функцію ініціалізації програмного UART`а, за допомогою якої здійснюється їх виклик, було підправлено наступним чином.

```

void SUART_Init (void)
{
/* Ініціалізація прапорів */
flag_tx_busy = FALSE;
flag_rx_ready = FALSE;
flag_rx_waiting_for_stop_bit = FALSE;
flag_rx_off = FALSE;

/* Налаштування виводів */
RX_DDRX &= ~ (1 << RX_PIN);
RX_PORTX |= (1 << RX_PIN);
TX_DDRX |= (1 << TX_PIN);
TX_PORTX |= (1 << TX_PIN);

/* Налаштування таймеру T0 */
TCCR0B = CSXX;
TCNT0 = TIMER_TCNT_VALUE;
TIMSK |= (1 << TOIE0);
}

```

В даній роботі було задіяно 8-ми розрядний таймер T0 в режимі NORMAL з перериванням за подією «переповнення». Переддільник таймеру (CSXX) визначається константою SUART_PRESCALER, а таймерна константа TIMER_TCNT_VALUE обчислюється за допомогою макросу.

						ЕЛІТ 8.171.00.10.376 ПЗ	Лист
Вим.	Лист	№ докум.	Підпис	Дата			30

Основний код програмного UART`а було розміщено в оброблювачу переривання таймеру T0, шляхом додавання в його початок коду перезапису значення лічильного регістру TCNT0. Оптимальним є використання режиму CTC, який у 8-ми разрядних таймерів іноді відсутній.

```
ISR (TIM0_OVF_VECTOR)
{
    uint8_t start_bit, flag_in;
    uint8_t rx_data;
    static uint16_t internal_rx_buffer;

    TCNT0 += TIMER_TCNT_VALUE;
    ... ..
    // основний код
    ....
}
```

Наступним оновленням коду програмного UART`а – є додавання кваліфікатора `volatile` до змінних, що отримав використання як в перериванні так і в звичайних функціях.

```
static volatile uint8_t inbuf [SUART_BUF_SIZE];
static volatile uint8_t qin = 0;
static volatile uint8_t qout = 0;

static volatile uint8_t flag_rx_ready;
static uint8_t flag_rx_waiting_for_stop_bit;
static uint8_t flag_rx_off;
static uint8_t rx_mask;
static uint8_t timer_rx_ctr;
static uint8_t bits_left_in_rx;

static volatile uint8_t flag_tx_busy;
static volatile uint8_t timer_tx_ctr;
static volatile uint8_t bits_left_in_tx;
```

					ЕЛІТ 8.171.00.10.376 ПЗ	Лист
Вим.	Лист	№ докум.	Підпис	Дата		31

```
static volatile uint16_t internal_tx_buffer;
```

Наступним етапом було додавання функції виведення рядків з ОЗУ і з flash пам'яті.

```
void SUART_PutStr (char * str)
{
    char ch;
    while (* str) {
        ch = * str ++;
        SUART_PutChar (ch);
    }
}
```

```
void SUART_PutStrFl (char __flash * str)
{
    char ch;
    ch = read_byte_flash (* str);
    while (ch) {
        SUART_PutChar (ch);
        str ++;
        ch = read_byte_flash (* str);
    }
}
```

Наступним кроком було створення проекту в IAR`е, шляхом підключення до нього отриманої «ліби» програмного UART`а та написання пари рядків коду для перевірки отриманих результатів. Решта змін, що були внесені до коду були пов`язані з невеликою оптимізацією, про них ми говорили.

Апаратне / програмне забезпечення UART працює наступним чином:

- При 9600 бодах, з тактовою частотою 8 МГц, тривалість одного біту складає $8000000/9600$ або 833,3 тактових циклу.

						ЕЛІТ 8.171.00.10.376 ПЗ	Лист
Вим.	Лист	№ докум.	Підпис	Дата			32

- Спочатку було здійснене відключення USI. Вхід в регістр зсуву USI було підключено до PB0, тому нами було визначено його як вхід та налаштовано переривання зміни пінів на ньому:

```
void InitialiseUSI (void) {
pinMode (DataIn, INPUT); // Визначення DI як вхідного
USICR = 0; // Відключення USI.
GIFR = 1 << PCIF; // Очищення прапора переривання зміни знаку.
GIMSK |= 1 << PCIE; // Дозвіл переривання зміни контактів
PCMSK |= 1 << PCINT0; // Включення зміни виводів на виведення 0
}
```

Початок байту викликає переривання зміни знаку. У процедурі обслуговування переривання зміни пін-коду робиться перевірка та визначення падаючого фронту, та налаштування таймеру / лічильнику в режимі CTC. Важливим є встановлення затримки в пів-біті, для потрапляння дов середини стартового біта, який складає 416,7 такту. Найближче, що можна донести - це попередній дільник 8 і порівняння, що дорівнює 52. Робимо очистку та здійснюємо дозвіл переривання порівняння виведення:

```
ISR (PCINT0_vect) {
if (! (PINB & 1 << PINB0)) { // Ігнорування, якщо DI високий
GIMSK &= ~ (1 << PCIE); // Відключення переривання зміни
TCCR0A = 2 << WGM00; // Режим CTC

TCCR0B = 0 << WGM02 | 2 << CS00; // Встановлення переддільнику на / 8
TCNT0 = 0; // Підрахунок від 0
OCR0A = 51; // Затримка (51 + 1) * 8 циклів
TIFR |= 1 << OCF0A; // Очищення прапора порівняння результатів
TIMSK |= 1 << OCIE0A; // Включення переривання порівняння виведення
}
}
```

Переривання збігу порівняння відбувається в середині стартового біта. В процедурі обслуговування переривання порівняння збігів повертається збіг порівняння з тривалістю одного біту, 104, здійснюється дозвіл USI початку

						ЕЛІТ 8.171.00.10.376 ПЗ	Лист
Вим.	Лист	№ докум.	Підпис	Дата			33

зміщення в бітах даних з подальшим зіставленням порівняння та вирішенням переривання переповнення USI:

```
ISR (TIMER0_COMPA_vect) {  
    TIMSK &= ~(1 << OCIE0A); // Відключення переривання COMPA  
    TCNT0 = 0; // Підрахунок від 0  
    OCR0A = 103; // Зрушення кожно з (103 + 1) * 8 циклів
```

// Включаємо переривання USI OVF та обираємо порівняння часу по Timer0 як джерело синхронізації USI:

```
    USICR = 1 << USIOIE | 0 << USIWM0 | 1 << USICS0;  
    USISR = 1 << USIOIF | 8; // Очищення прапора USI OVF та встановлення лічильнику  
}
```

Встановлюємо режим проводки на 0 з $0 << USIWM0$, що гарантує, що вихід зсувного регістру USI не матиме впливу виведення даних, PB1.

Коли 8 біт був зрушений в перериванні, тоді переповнення USI відбувається. Процедура обслуговування переривань відключає USI, зчитує регістр зсуву USI та здійснює дозвіл переривання, зміни переривання для наступного байта:

```
ISR (USI_OVF_vect) {  
    USICR = 0; // Відключення USI  
    int temp = USIDR;  
    Дисплей (ReverseByte (темп));  
    GIFR = 1 << PCIF; // Очищення прапора переривання зміни знака.  
    GIMSK |= 1 << PCIE; // Включення переривання зміни знаку  
}
```

UART відправляє біт LSB, а USI показує, що MSB є першим, є потреба в скасуванні порядку біту після прийому, і це можна отримати з використанням короткої програмної програми ReverseByte ():

```
unsigned char ReverseByte (unsigned char x) {  
    x = ((x >> 1) & 0x55) | ((X << 1) & 0xaa);
```


З додаванням параметру ATtiny45 @ 8MHz (зовнішній кристал, BOD вимкнено) в підменю Board. Необхідно обрати це та Burn Bootloader, для правильного встановлення плавких запобіжників з використанням Tiny AVR завантаження програми до ATtiny45.

Для використання даної процедури з різними частотами кристалів та швидкостями передавання, необхідною буде зміна значення попереднього розподілу та порівняння для таймеру / лічильнику в процедурі переривання зсуву піну та підпрограми переривання USI.

Для високих швидкостей передачі даних доцільним є врахування затримки обробки переривання. Програма ATtiny45 USI UART: проста програма ATTII USI UART.

Було змінено процедуру переривання порівняння вихідних даних, ISR (PCINT0_vect), для відсутності впливу на біти переривання таймера / Counter1. Це дає можливість використання функції таймеру Arduino millis () і delay (), з використанням таймеру / лічильнику1. Метою є очищення прапора порівняння виводу перед включенням переривання порівняння виведення, було додано рядок: TIFR |= 1 << OCF0A в підпрограму PCINT0_vect, а потім затримано половину бітів, в середині старшого біта, змінивши налаштування OCR0A на:

```
OCR0A = 51; // Затримка (51 + 1) * 8 циклів
```

Це гарантує, що кожен приклад зчитується посередині кожного біту.

2.2 Розробка структурної схеми проектованої системи

Конструктивна ємність клем та роз'ємів для підключення виводів елементів даної системи визначає мінімальні значення ємності, індуктивності, що вимірюються з точністю вимірювання параметрів в межах від 100 ПФ до 10 мкГн. В інших межах, вимірювання заданих параметрів визначається класом точності стрілочної вимірювальної головки, а струм, що споживається системою має максимальне значення 25 мА.

Принцип роботи проектованої системи базується на вимірі середнього значення розрядного струму ємності конденсатора і ЕРС самоіндукції індуктивності, що входять до складу даної системи. Вимірювач, принципова схема якого наведена на рис.1, в своєму складі містить генератор, що зображений на елементах DD1.5, DD1.6, і характеризується кварцовою стабілізацією частоти, наявністю лінійки дільників частоти на мікросхемах DD2 – DD6 та буферних інверторів (DD1.1 – DD1.4).

Резистор R4 здійснює обмеження вихідного струму інверторів. Ланцюг, що містить елементи VD7, VD8, R6, C4 - застосовується для вимірювання ємності, а ланцюг VD6, R5, R6, C4 – для вимірювання індуктивності.

Діод VD9 здійснює захист мікроамперметру РА1, запобігає можливному перевантаженню.

Ємність конденсатора, що зображений на схемі позначкою С4 досить великий для зменшення тремтіння стрілки, коли вона знаходиться на максимальній межі вимірювання, при якій тактова частота є мінімальною і дорівнює 10 Гц.

Отже, якщо напруга живлення постійна разом з частотою сигналу, то напруга на виході ОУ є прямопропорційною величинам ємності або індуктивності, що вимірюються. Дане твердження має місце в випадку, коли заряд конденсатора повний протягом половини періоду збудливої напруги і протягом іншої половини періоду здійснилось його повне розрядження.

Аналогічна закономірність існує і для котушки індуктивності, струм в якій повинен встигнути зрости до максимального значення і після знизитись до нуля. Дані умови можливо виконати шляхом вибору резисторів R1-R3 і частоти збудливої напруги.

					ЕЛІТ 8.171.00.10.376 ПЗ	Лист
						37
Вим.	Лист	№ докум.	Підпис	Дата		

Напругу, значення якої є пропорційним значенню параметру елемента, що вимірюється, подають з виходу ОУ через фільтр R4C2 на вбудований десятирозрядний АЦП мікроконтролера DD1.

Конденсатор С1 – є фільтром внутрішнього джерела напруги АЦП. Три елемента DD1, що знаходяться на схемі сверху разом з VD2, VD4, C3, C4 відповідають за підтримання напруги мінус 5 В, що є необхідним для роботи ОУ.

Проектована система результат вимірювань відображає на десятирозрядному семисегментному рідкокристалічному індикаторі HT1613.

З метою підвищення точності проектувана система має 9 піддіапазонів вимірювання, частота збудливої напруги на 1-ому з яких дорівнює 800 кГц, що забезпечує якісне вимірювання для конденсаторів з ємністю до 90 пФ та котушки, індуктивність яких до 90 мкГн.

На кожному з наступних піддіапазонів частота знижується в 4 рази, з відповідним розширенням меж вимірювання.

На 9 піддіапазоні частота дорівнює 12 Гц, що є необхідною для вимірювання конденсаторів ємністю до 5 мкФ і котушок з індуктивністю до 5 Гн.

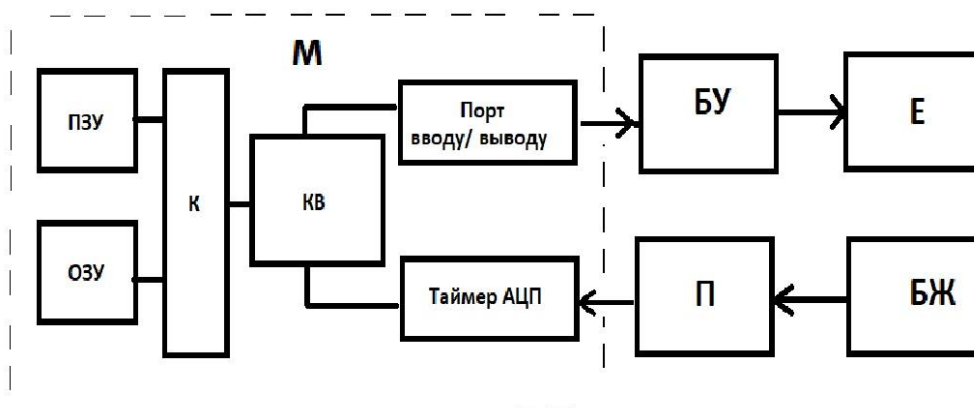


Рисунок 2.8 - Структурна схема вимірювача ємності та індуктивності

На даній схемі зображені наступні компоненти:

- БЖ- блок живлення ;

- П- підсилювач, компаратор;
- М- мікросхема ;
- КВ-кварцовий резонатор;
- К- контроллер пам'яті;
- Таймер АЦП;
- Порт Вводу / Виводу інформації;
- ПЗП – постійний запам'ятовуючий пристрій;
- ОЗП – оперативний запам'ятовуючий пристрій ;
- БУ – блок управління (мікроконтроллер);
- Е – екран для виведення інформації.

БЖ передає сигнал, який рухається до підсилювачу, де здійснюється його порівняння та підсилювання з використанням компаратора. Далі сигнал потрапляє на мікросхему, а після до блоку управління, і там здійснюється запам'ятовування та з використанням програми, виведення інформації на екран.

Вимірювач індуктивності та ємності складається з наступних компонентів:

- 1) індикатор HG1 - здійснює відображення показаників вимірювання;
- 2) мікроконтролер ATtiny45, що управляється за допомогою програми, що додається;
- 3) операційний підсилювач DA1;
- 4) мікросхема DD2;
- 5) світлодіод HL1, що виконує функцію стабілітрона на 1,5 В для живлення індикатору;
- 6) змінний резистор R14 - регулює напругу на підсилювачі (+1...+5В);
- 7) змінний резистор R11 – відповідає за налаштування точності вимірювання ємності;
- 8) змінний резистор R12- відповідає за налаштування вимірювання індуктивності;
- 9) кнопка SB1 - програмно корегує нуль;
- 10) інші EPE необхідні для роботи приладу.

Необхідний діапазон проектована система обирає автоматично, після відкриття живлення вимірювання починається з дев'ятого піддіапазону, а в

					ЕЛІТ 8.171.00.10.376 ПЗ	Лист
						39
Вим.	Лист	№ докум.	Підпис	Дата		

процесі перемикання, номер піддіапазону виводиться на індикатор, і це дозволяє визначити частоту виконуваного вимірювання.

За допомогою кнопки SB1 здійснюється програмна корекції нуля, що компенсує ємність і індуктивність клем і перемикача SA1.

Резистор R13 здійснює швидку розрядку конденсаторів C10 і C11 при вимикненому живленні.

Після вибору необхідного піддіапазону, результат вимірювання ємності в пФ та індуктивності в мкГн виводиться на індикатор. Для зчитування десяти частки пФ (мкГн) і одиниці мкФ (Гн) відокремлені порожнім знакомісцем, результат округляється до трьох значущих цифр.

В вимірюючих механізмах електромагнітної системи обертаючий момент зумовлюється електромеханічною дією магнітного поля вимірюваного струму в нерухливій котушці системи на рухливий феромагнітний яркір.

Механічні сили переміщують яркір, що втягується в котушку, включену в ланцюг вимірюваного струму таким чином, щоб енергія магнітного поля механізму зростає.

З метою посилення магнітного поля та регулювання значення обертаючого моменту був використан нерухливий феромагнітний сердечник.

Протидіючий момент створюється спіральною пружиною. З метою досягнення захисту від зовнішніх магнітних полів, вимірювальний механізм розміщується в феромагнітному екрані. Обертаючий момент, що діє на рухливу частину, пропорційний похідній енергії магнітного поля по координаті переміщення індуктивності і струму котушки.

Магнітне поле котушки з вимірюваним струмом є слабким, в зв'язку з проходженням значної частини відстані магнітних ліній в повітрі. Через це в зв'язку з низькою чутливістю вимірювального механізму електромагнітної системи вона потребує захисту від зовнішніх магнітних полів.

В астатичному вимірювальному механізмі число котушок подвоюється, а її магнітні поля мають протилежні напрямки. Зовнішнє однорідне магнітне поле підсилює магнітне поле однієї котушки, і пропорційно послабляє магнітне поле другої котушки, залишаючи постійним загальний обертаючий момент вимірювального механізму.

Під час проектування складних систем, необхідно враховувати всі змінні які мають вплив на елементи електричного кола, яке складається з конкретних

					ЕЛІТ 8.171.00.10.376 ПЗ	Лист
Вим.	Лист	№ докум.	Підпис	Дата		40

ЕРЕ, з певними значеннями значеннями, визначення яких здійснюється вимірювальними приладами та системами.

Для вимірювання параметрів індуктивності та ємності проекрованої системи та їх порівняння доцільним є використання вимірювачу LC.

3 РОЗРОБКА СХЕМИ ЕЛЕКТРИЧНОЇ ФУНКЦІОНАЛЬНОЇ ПРОЕКТОВАНОЇ СИСТЕМИ

Розробка схеми електричної функціональної пристрою проекрованої системи необхідна для вивчення принципу роботи при виробленні, налагодженні, контролі та ремонті. На ній позначені функціональні частини системи та їх взаємозв'язок. Вони зображені в вигляді умовних графічних позначень відповідно до вимог державних стандартів, є дозволені окремі функціональні частини, що не містять умовні графічні позначення зобразити в вигляді прямокутників, і розвивати до рівня принципів схем, з привоєною назвою та умовним позначенням. Допустими є об'єднання функціональних частин в функціональні групи, що виділені на схемі штрих-пунктирними лініями.

Наступні вказівки необхідні бути на функціональній схемі:

- 1) для кожної функціональної частини, зображеної прямокутником, необхідна бути присутня її назва або умовне позначення, вписана в прямокутник;
- 2) для кожної функціональної частини або елемента умовне графічне позначення позиційного значення.

В разі використання функціональної схеми разом з принциповою, позиційне позначення елементів та функціональних частин повинні бути однаковими. Перелік елементів складають, використовуючи дані принципової схеми. В разі наявності лише функціональної схеми (без принципової), позиційне позначення елементів і функціональних частин вказані за загальними правилами і розробляється перелік елементів. На функціональних схемах рекомендується вказувати поряд технічні характеристики функціональних частин.

Конструкторська документація необхідна для виготовлення вимірювача індуктивності та ємності, виконана в даній роботі за допомогою програм Splan та Sprint–Layout, що є простим і зручним інструментом для креслення електронних і

						ЕЛІТ 8.171.00.10.376 ПЗ	Лист
Вим.	Лист	№ докум.	Підпис	Дата			41

електричних схем, передбачає перенесення стандартних позначень ЕРЕ та символів з бібліотеки елементів на схему, що прив'язані до координатної сітки.

Splan містить інструменти для креслення і редагування, що дозволяють розробити схеми зручно та ефективно, серед яких є автонумерація елементів, складання переліку елементів.

Зобразимо функціональну схему блоку управління тобто мікроконтроллера АТtinу45 проектованої системи

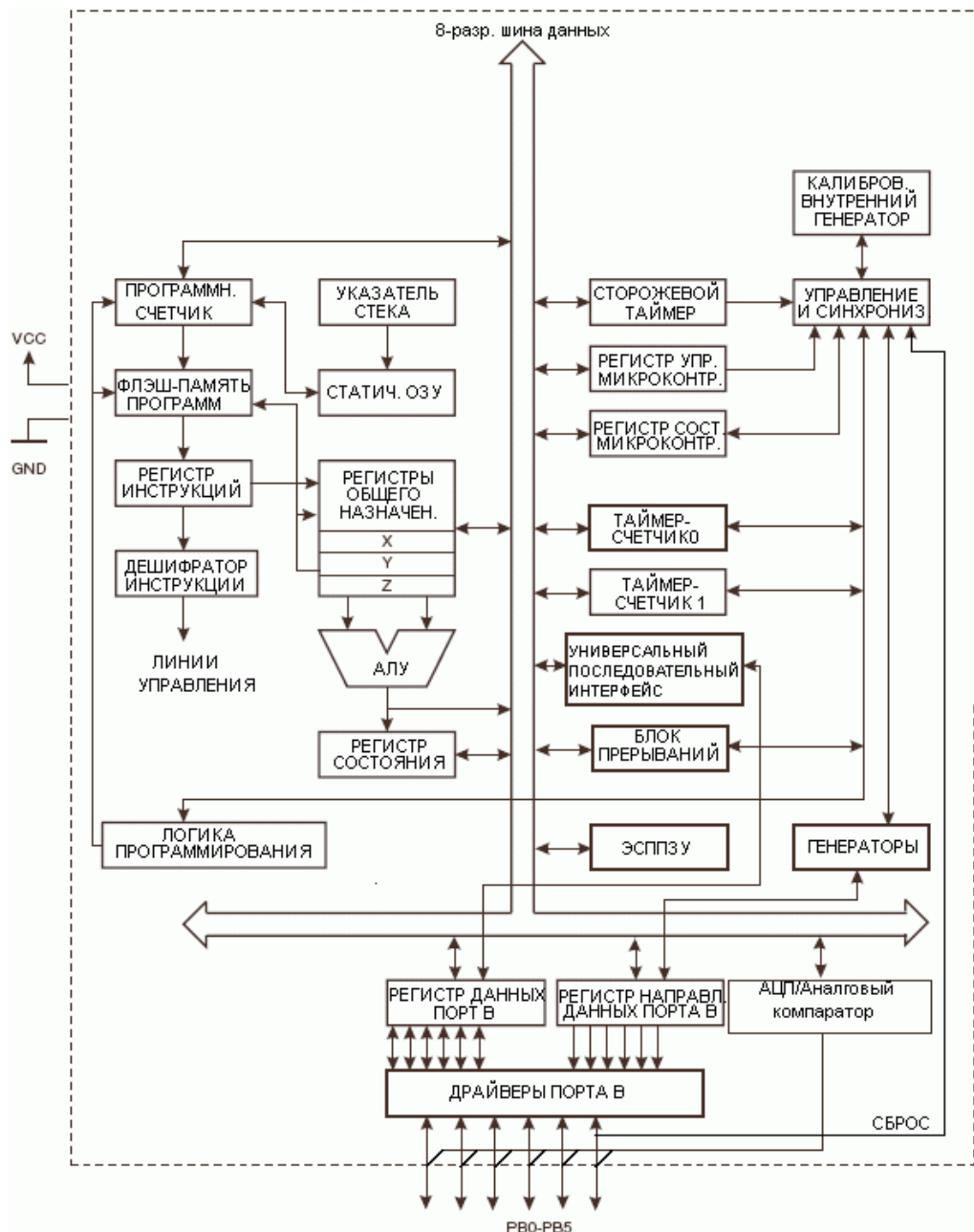


Рисунок 3.1 – Функціональна схема блоку управління АТtinу45

Особливості блоку:

- Висока продуктивність, економічність, 8-розрядний AVR-мікроконтролер.
- Удосконалена RISC-архітектура.
- 120 інструкцій, більшість яких виконуються за один цикл.
- 32 x 8 універсальні регістри загального призначення.
- Статична робота.
- Енергонезалежність пам'яті, програм, даних.
- Внутрішня програмована флеш-пам'ять програм розміром 2, 4, 8 Кбайт Atiny45, стійкість до зносу 10 тис. циклів запис / зтирання.
- 128/256/512 байт внутрішньої системної програмованої ЕСППЗУ (ATtiny25 / 45/85), стійкість до зносу 100 тис. циклів запису / зтирання.
- 128/256/512 байтів внутрішньої статичної ОЗП (ATtiny25 / 45/85).
- Програмовані біти захисту флеш-пам'яті і ЕСППЗУ.

Периферійні пристрої мають наступні відмінні риси:

- 8-розрядний таймер-лічильник з переддільника та двох каналів ШІМ.
- 8-розрядний високошвидкісний таймер-лічильник що має окремий попередній дільник, двоє високочастотні виходи ШІМ з окремими регістрами з заданим порогом порівняння.

Програмований генератор паузи має наступні складові:

- 1) універсальний послідовний інтерфейс;
- 2) 10-розрядний АЦП несиметричного каналу;
- 3) 2 диференціальних каналів з програмованим посиленням (1x, 20x);
- 4) програмований сторожовий таймер з встроєним генератором;
- 5) вбудований аналоговий компаратор.

Мікроконтролер, що був використаний для даної роботи має наступні спеціальні функції:

- Вбудована налагоджувана система debugWIRE.
- Внутрішнє системне програмування через порт SPI.
- Зовнішні та внутрішні джерела переривань.
- Економічні режими:
 - ✓ холостий хід (Idle);
 - ✓ зменшення шумів АЦП (ADC Noise Reduction);
 - ✓ знижена потужність (Power-down);

						ЕЛІТ 8.171.00.10.376 ПЗ	Лист
Вим.	Лист	№ докум.	Підпис	Дата			43

- Вдосконалена схема скиду при поданні живлення.
- Програмована схема контролю напруги живлення.
- Вбудований калібрований генератор.
- Ввід-вивід корпусу.
- 6 програмованих ліній введення-виведення.
- 8-вивідний корпус PDIP і 8-вивідний SOIC.
- Робоча напруга:
 - 1.8 - 5.5В для ATtiny25 / 45 / 85V
 - 2.7 - 5.5В для ATtiny25 / 45/85
- Швидкодія:
 - ✓ ATtiny25 / 45 / 85V: 0 - 4 МГц при 1.8 - 5.5В, 0 - 10 МГц при 2.7 - 5.5В
 - ✓ ATtiny25 / 45/85: 0 - 10 МГц при 2.7 - 5.5В, 0 - 20 МГц при 4.5 - 5.5В
- Промисловий температурний діапазон
- Низьке споживання струму:
 - ✓ Активний режим: 1 МГц, 1.8В: 450 мкА
 - ✓ Режим зниженої потужності: 0.1 мкА при 1.8В

Загальний опис мікроконтролеру, що було застосовано до проектованої системи:

ATtiny45 - економічний 8-розрядні КМОП мікроконтролери, зроблені за вдосконаленою AVR RISC-архітектурою. З виконанням багатьох інструкцій, за один машинний цикл мікроконтролери ATtiny25 / 45/85 мають продуктивність 1 млн. операцій в секунду з тактовою частотою 1МГц, і це забезпечує високу споживану потужність і швидкодію.

Ядро AVR здійснює комбінування багатьох інструкцій з 32 регістрами загального призначення, що підключені до арифметико-логічного пристрою (АЛП), що забезпечує доступ при застосуванні інструкції відразу до 2 регістрів і виконати її за 1машинний цикл. Результуюча архітектура характеризується високою ефективністю, збільшуючи продуктивність 10 разів порівняно з CISC-мікроконтролерами.

ATtiny45 містить наступні елементи: 2/4/8 кбайт внутрішньої системного-програмованої флеш-пам'яті, 128/256/512 байт ЕСППЗУ, 128/256/256 байт статичного ОЗУ, 6 ліній введення-виведення загального призначення, 32 універсальних робочих регістрів загального призначення, 1 8-розрядний таймер-

						ЕЛІТ 8.171.00.10.376 ПЗ	Лист
Вим.	Лист	№ докум.	Підпис	Дата			44

лічильник з режимами порівняння, 1 8-розрядний високошвидкісний таймер-лічильник, універсальний послідовний інтерфейс, внутрішні і зовнішні переривання, 4-канальний 10-розрядний АЦП, програмований сторожовий таймер з внутрішнім генератором, 3 програмні економічні режими.

В режимі холостого ходу (Idle) зупиняється ЦПУ, але працюють статичний ОЗУ, таймер-лічильник, АЦП, аналоговий компаратор і система переривань. В режимі зниженої потужності (Power-down) зберігається вміст регістрів, здійснюється відключення всіх вбудованих функції до наступного переривання або апаратного скидання. В режимі зниження шумів АЦП (ADC Noise Reduction) зупиняється робота ЦПУ та всіх модулів введення-виведення, окрім АЦП, за допомогою цього знижуються цифрові шуми під час перетворення АЦП.

Мікроконтролер має високу незалежну пам'ять, вбудована флеш-пам'ять перепрограмується через послідовний інтерфейс SPI за допомогою бюджетного програматора або за самопрограмування.

					ЕЛІТ 8.171.00.10.376 ПЗ	Лист
Вим.	Лист	№ докум.	Підпис	Дата		45

4 РОЗРОБКА І РОЗРАХУНОК ПРИНЦИПОВИХ ЕЛЕКТРИЧНИХ СХЕМ ВУЗЛІВ І БЛОКІВ СИСТЕМИ

4.1 Вибір елементної бази

При розробці проектованої системи необхідно:

- обрати радіоелементи, які зроблять роботу схеми надійною і стійкою;
- здійснити скорочення номенклатури деталей та елементів, щоб конструкція була легкою та компактною;
- ціна- якість ЕРЕ;
- аналіз очікувань від системи, розробка її .

Вимірювач індуктивності та ємності має наступну елементну базу:

- індикатор,
- мікроконтролер,
- операційний підсилювач,
- мікросхему,
- світлодіод,
- резистори постійні та змінні,
- конденсатори,
- діоди,
- перемикач,
- кнопку.

Дані ЕРЕ стандартні, що підвищує технологічність виробу.

Під час розробки друкованої плати береться до уваги ширина доріжок, отвори для свердління, враховується розмір кожного з елементів, для слідування правилам оптимальної конфігурації.

Деталі даного вимірювача індуктивності та ємності є змонтованими на односторонній друкованій платі довжиною 12 см та шириною 5,1 см.

Довжина дротів вхідних клем повинна бути не вище 2-3 см.

Діоди VD3-VD6 – є високочастотними і мають мале падіння напруги - ДЗ11, Д18, Д20.

Підстроювальні резистори R11, R12, R14 є малогабаритними, типу СПЗ-19. Заміна R11 на дротяний резистор може привести до зниження точності вимірювання. Заміна DA1 140УД1208 можлива на інший ОП, що має ланцюг установки нуля та здатен працювати від напруги 5 В. Номінал ємностей С4, С5, С11 знижувати небажаном на DD2, але можна замінити на будь-який КМОП

					ЕЛІТ 8.171.00.10.376 ПЗ	Лист
Вим.	Лист	№ докум.	Підпис	Дата		46

мікросхем серій 1561, 1561, 1554, 74НС, 74АС, з 6 інверторів, як варіант на 74НС14. Застосування ТТЛ серій 155, 555, 1533 не є бажаним.

При розробці проекрованої системи необхідно:

- обрати радіоелементи, які зроблять роботу схеми надійною і стійкою;
- здійснити скорочення номенклатури деталей та елементів, щоб конструкція була легкою та компактною;
- ціна- якість ЕРЕ;
- аналіз очікувань від системи, розробка її .

Виходячи з даних вимог, в конструкції друкованого вузла вимірювача містяться наступні радіоелементи:

Конденсатори К10-176 - конденсатори керамічні, що є ущільненими, постійної ємності, низьковольтними, що призначені для експлуатації в вигляді вбудованих елементів в ланцюгах постійного та змінного струмів, в імпульсному режимі.

Експлуатаційні характеристики:

- вібраційний діапазон частот 1 - 5000 Гц;
- діапазон робочих температур від - 40 - + 85 ° С;
- відносна вологість не більше 98%;
- допустимі відхилення ємності $\pm 10\%$.

Наступні показники надійності:

- напрацювання на відмову не нижче 15000 год;
- інтенсивність відмов не перевищує $1 \cdot 10^{-6}$ (1 / год);
- термін зберігання не нижче 15 років.

Конденсатор К10-176 був обраний, через малі габарити, низьку інтенсивність відмов, бюджетність.

Конденсатори К50-68- оксидно-електролітичні, алюмінієві, призначенням яких є робота в ланцюгах постійного, пульсуючого струму та в робота імпульсному режимі.

Експлуатаційні характеристики:

діапазон робочих температур від -40 до + 85 ° С;
відносна вологість не вища 98% ;
допустимі відхилення ємності від -10 до +50%.

Показники надійності:

- напрацювання на відмову не нижче 15000 год;

									Лист
									47
Вим.	Лист	№ докум.	Підпис	Дата	ЕЛІТ 8.171.00.10.376 ПЗ				

- інтенсивність відмов не вища $2 \cdot 10^{-6}$ (1 / год);
- термін зберігання не нижче 15 років.

Конденсатор К50-68 був використаний через компактні габарити, низьку інтенсивність відмов, економічність.

Мікросхема КР140УД1208 - операційний підсилювач загального застосування.

Експлуатаційні характеристики:

- коефіцієнт посилення 20000;
- напруга зсуву 6 мВ;
- вхідний струм 500 нА;
- напруга живлення ± 5 В;
- вхідний опір 0,4 МОм;
- споживаний струм 3 мА;
- швидкість наростання 0,3 В / мкс;
- діапазон робочих температур 0 - + 70 ° С;
- -відносна вологість не вища 98%.

Показники надійності:

- напрацювання на відмову не нижче 20000 год;
- інтенсивність відмов не вища $2 \cdot 10^{-10}$ (1 / год);
- термін зберігання не менше 18 років.

Мікросхема була використана, через високий коефіцієнт посилення і низьке енергоспоживання.

Мікросхема КР1157ЕН502А- стабілізатор напруги.

Експлуатаційні характеристики:

- напруга стабілізації 5 В;
- максимальний струм стабілізації навантаження 0,1 А;
- розсіюється максимальна потужність 0,5 Вт;
- діапазон робочих температур від мінус 60 до плюс 80 ° С;
- відносна вологість не більше 98%.

Показники надійності:

- напрацювання на відмову не нижче 15000 год;
- інтенсивність відмов не вища $2 \cdot 10^{-8}$ (1 / год);
- термін зберігання до 12 років.

					ЕЛІТ 8.171.00.10.376 ПЗ	Лист
Вим.	Лист	№ докум.	Підпис	Дата		48

Мікросхема обрана тому, що має низький коефіцієнт нестабільності по напрузі.

Мікросхема K561ЛН2- містить 6 буферних інверторів.

Експлуатаційні характеристики:

- напруга стабілізації 3- 15 В;
- розсіювання потужності 0,4 мВт;
- діапазон робочих температур від -10 до +70 ° С;
- відносна вологість не вища 98%.

Показники надійності:

- напрацювання на відмову до 40000 год;
- інтенсивність відмов не вища $2 \cdot 10^{-8}$ (1 / год);
- термін зберігання від 15 років.

Мікросхема серії K561 більш сучасна порівняно з серією K176 і має вдосконалені параметри.

Мікросхема АТtinu45- мікроконтролер.

Експлуатаційні характеристики:

- напруга живлення 2,7 -5,5 В;
- робоча частота 4 МГц;
- 8 біт;
- ємність пам'яті 1 Кбайт;
- тип пам'яті FLASH;
- діапазон робочих температур 0 - + 70 ° С;
- відносна вологість не вища 98%.

Показники надійності:

- напрацювання на відмову від 40000 год;
- інтенсивність відмов не вища $2 \cdot 10^{-8}$ (1 / год);
- термін зберігання до 18 років.

Індикатор символний НТ1613- десятирозрядний, семисегментний рідкокристалічний індикатор.

Експлуатаційні характеристики:

- напруга живлення 1,2 - 1,7 В;
- споживаний струм 10 мкА.

Показники надійності:

					ЕЛІТ 8.171.00.10.376 ПЗ	Лист
Вим.	Лист	№ докум.	Підпис	Дата		49

- напрацювання на відмову від 50000 год;
- інтенсивність відмов не вища $2 \cdot 10^{-8}$ (1 / год);
- термін зберігання до 15 років.

Резистори С2-33Н-0125- резистори постійні, металодіелектричні, не ізольовані, використовуються для роботи в ланцюгах постійного, змінного та імпульсного, є елементами навісного монтажу.

Експлуатаційні характеристики:

- допуск 10%;
- номінальна потужність 0,125 Вт;
- температура при номінальній потужності $+ 70^{\circ} \text{C}$;
- рівень власних шумів до 5 дБ;
- опір ізоляції $1 \cdot 10^4$ Мом;
- діапазон температур від -60 до $+155^{\circ} \text{C}$;
- відносна вологість повітря при $+35^{\circ} \text{C} - 98^{\circ} \text{C}$;
- гранична робоча напруга змінного чи постійного струму- 200 В.

Показники надійності:

- напрацювання на відмову від 30000 год;
- інтенсивність відмов до $1 \cdot 10^{-7}$ (1 / год);
- термін зберігання до 15 років.

Резистор використаний, так як має велике напрацювання на відмову, високу стабільність параметрів, низький рівень шумів і економічність.

Резистори СПЗ-19а- застосовуються для роботи з колами постійного та змінного струму в безперервному і імпульсному режимах, використовуються для монтажу на поверхню.

Експлуатаційні характеристики:

- потужність (при $t = 40^{\circ} \text{C}$) 0,5 Вт;
- гранична робоча напруга до 150 В;
- допустиме відхилення опору від номінального $\pm 20\%$;
- діапазон температур від $- 60$ до $+125^{\circ} \text{C}$;
- відносна вологість повітря при $+35^{\circ} \text{C}$ 98%;

					ЕЛІТ 8.171.00.10.376 ПЗ	Лист
Вим.	Лист	№ докум.	Підпис	Дата		50

- число циклів переміщення рухомої системи дорівнює 500;

Показники надійності:

- напрацювання на відмову від 20000 год;
- інтенсивність відмов до $1 \cdot 10^{-7}$ (1 / год);
- термін зберігання від 10 років.

Резистор використали, через високий рівень напрацювання на відмову, високу стабільність параметрів, низький рівень шумів.

Діод ДЗ11А- кремнієвий.

Експлуатаційні характеристики:

- прямий струм = 0,3 А;
- зворотний струм = 10 мкА;
- пряма напруга = 1 В;
- зворотна напруга = 100 В;
- діапазон робочих температур від -60 до +85 ° С;
- відносна вологість до 98%.

Показники надійності:

- напрацювання на відмову від 30000 год;
- інтенсивність відмов до $2 \cdot 10^{-7}$ (1 / год);
- термін зберігання від 12 років.

Діод було використано через компактність, широкий діапазон робочих температур, високий рівень ударостійкості.

Діод КД522Б- кремнієвий.

Експлуатаційні характеристики:

- прямий струм 50 мА;
- зворотний струм 1 мкА;
- пряма напруга 1 В;
- зворотна напруга 75 В;
- діапазон робочих температур від -60 до +85 ° С;
- відносна вологість до 98%.

Показники надійності:

- напрацювання на відмову від 30000 год;
- інтенсивність відмов не до $2 \cdot 10^{-7}$ (1/ год)
- термін зберігання від 12 років.

					ЕЛІТ 8.171.00.10.376 ПЗ	Лист
Вим.	Лист	№ докум.	Підпис	Дата		51

4.2 Розрахунки та синтез основних електричних вузлів, блоків проектованої системи

Розглянемо електричний розрахунок баластного резистора.

Початкові дані:

- напруга живлення $U_{п} = 5 \text{ В}$;
- пряме напруга світлодіода $U_{HLпр} = 1,5 \text{ В}$;
- прямиий струм світлодіода $I_{HLпр} = 10 \text{ мА}$.

Розрахуємо опір і потужність резистора $R_б$.

Розрахунок ведеться за схемою наведеною на рисунку 4.1.

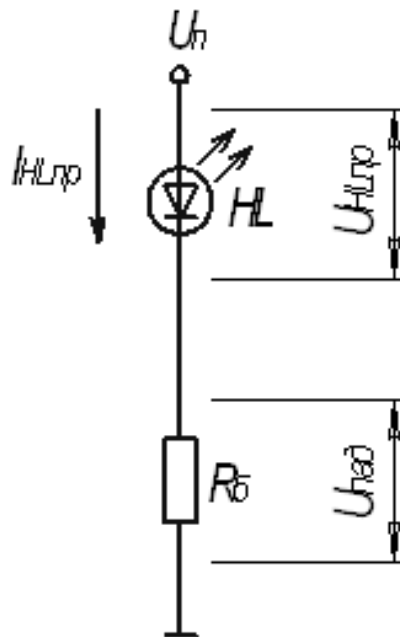


Рисунок 4.1 - Схема включення

Визначемо падіння напруги на резисторі $R_б$, $U_{пад}$, за формулою:

$$U_{пад} = U_{п} - U_{HLпр}; \quad (4.1)$$

$$U_{пад} = 5 - 1,5 = 3,5 \text{ В.}$$

Визначемо опір резистора, $R_б$, Ом, за формулою:

					ЕЛІТ 8.171.00.10.376 ПЗ	Лист
Вим.	Лист	№ докум.	Підпис	Дата		52

$$R_{\delta} = \frac{U_{\text{пад}}}{I_{\text{НЛпр}}} ; \quad (4.2)$$

$$R_{\delta} = \frac{3,5}{0,01} = 350 \text{ Ом}$$

Визначемо потужність резистора R_{δ} , за формулою:

$$P = U_{\text{пад}} \times I_{\text{НЛпр}} ; \quad (4.3)$$

$$P = 3,5 \times 0,01 = 0,035 \text{ Вт.}$$

На основі отриманих даних обираємо резистор R_{δ} з ряду E24 C2-33H-0,125-360 Ом.

Розглянемо електричний розрахунок стабілізатора напруги

Початкові дані:

- вихідна напруга $U = 5 \text{ В}$
- струм навантаження $I_{\text{н}} = 15 \text{ мА}$;
- частота мережі $f = 50 \text{ Гц}$;

Потрібно визначити:

- тип стабілізатора напруги;
- номінали ємностей C_9 і C_{10} .

Враховуючи вихідну напругу, що дорівнює 5 В і стан сучасної елементної бази, обираємо стабілізатор серії КР1157ЕН502А.

Розглянемо тепловий розрахунок інтегрального стабілізатора

Початкові дані:

- потужність розсіювання $P_{\text{рас}} = 0,5 \text{ Вт}$;
- температура навколишнього середовища $T_a = 55 \text{ }^\circ \text{С}$;
- максимальна допустима температура $T_j = 75 \text{ }^\circ \text{С}$.

Визначемо температуру нагрівання корпусу інтегрального стабілізатора КР1157ЕН502А без радіатора.

Визначемо температуру нагрівання корпусу без радіатора, T_j , $^\circ \text{С}$, за наступною формулою:

					ЕЛІТ 8.171.00.10.376 ПЗ	Лист
						53
Вим.	Лист	№ докум.	Підпис	Дата		

$$T_j = T_A + P_{\text{рас}} \times R_{Tj-A} ; \quad (4.4)$$

де R_{Tj-A} - тепловий опір корпус / повітря, $R_{Tj-A} = 5 \text{ }^\circ\text{C} / \text{Вт}$,

$$T_j = 55 + 0.5 \times 5 = 57.5^\circ\text{C}.$$

Струм навантаження, який може забезпечити цей стабілізатор дорівнює приблизно 15 мА. Він розрахований на струм 250 мА.

Напруга стабілізації даної мікросхеми $5 \text{ В} \pm 0,2 \text{ В}$, при $U_{\text{вх.}} = 8 \text{ В}$, що підходить згідно з параметрами напруги живлення.

З метою забезпечення стійкості роботи стабілізатора при імпульсній зміні струму навантаження на виході, необхідно використати конденсатор з ємністю не від 10 мкФ, тому вибираємо ємність 47 мкФ на виході і 100 мкФ на вході.

Конденсатори обираємо марки К50-68, що мають наступні основні параметри:

- Діапазон номінальних значень ємності: 0.1 - 47000 мкФ
- Номінальна напруга: 6.3 - 450 В
- Діапазон робочих температур: від -55 до + 105 $^\circ\text{C}$
- Допустиме відхилення ємності від номіналу: $\pm 20\%$

Дані конденсатори відповідають необхідним умовам: компактність, сучасна елементна база і економічність.

Робоча напруга конденсатора має бути більшою ніж напруга стабілізації, отже обираємо конденсатори з напругою 16 В на вході і виході.

С10 - К50-68-16В-47 мкФ;

С11 - К50-68-16В-100 мкФ.

Для нормальної роботи стабілізатора на його вході має бути напруга не менше 7 В. А так як струм споживання самого стабілізатору не більший 5 мА, то струм навантаження випрямлювача має бути не менший 20 мА.

					ЕЛІТ 8.171.00.10.376 ПЗ	Лист
						54
Вим.	Лист	№ докум.	Підпис	Дата		

5 НАУКОВО – ДОСЛІДНА ЧАСТИНА

Для побудови проектованої системи було використано МК ATtiny45, аналізуючи доцільність його використання, порівнюючи з аналогами. Для цього необхідно проаналізувати його параметри та принцип роботи.

Мікроконтролери ATmega328P / ATtiny45 – програмування були створені фірмою «Atmel», було випущено два мікроконтролера, що мали якість та визнання в даній сфері. Мікроконтролер ATmega328P та «Atmel» - чіп ATtiny45, це дві мікросхеми, що мають певні відмінності. Для запрограмування мікроконтролеру ATmega328P та прошивки чіпу ATtiny45 програмним кодом, було використано однакові схематичний та і програмний підходи.

Розглянемо більш детально технічні способи програмування ATmega328P та ATtiny45, та необхідне для прошивки даних МК програмне забезпечення.

Розглянемо мікроконтролери ATmega328P і ATtiny45 сімейства AVR. ATtiny25 / 45/85 – економічні 8-розрядні КМОП мікроконтролери, для виконання яких використовувалась вдосконалена AVR RISC-архітектура. Враховуючи, що були виконані більшість інструкцій, отримали що за один машинний цикл мікроконтролери ATtiny25 / 45/85 досягають продуктивності 1 млн перацій в секунду з тактирование частотою 1МГц, що дозволяє розробнику оптимізувати споживану потужність і швидкодія [7].

ATtiny25 / 45/85 містить наступні елементи:

- 2/4/8 Кбайт внутрішньо-програмованої флеш-пам'яті,
- 128/256/512 байт ЕСПЗУ,
- 128/256/256 байт статичного ОЗУ,
- ліній введення-виведення загального призначення,
- 32 універсальні робочі регістри загального призначення,
- один 8-рзрядний таймер-лічильник з режимами порівняння,
- один 8-розрядний високошвидкісний таймер-лічильник,
- універсальний послідовний інтерфейс,
- внутрішні і зовнішні переривання,
- 4-канальний 10-разр. АЦП, програмований сторожовий таймер з внутрішнім генератором, а також три програмно обираних економічних режиму. У режимі холостого ходу (Idle) зупиняється ЦПУ при цьому

						ЕЛІТ 8.171.00.10.376 ПЗ	Лист
Вим.	Лист	№ докум.	Підпис	Дата			55

продовжують роботу статичне ОЗУ, таймер-лічильник, АЦП, аналоговий компаратор і система переривань. У режимі зниженої потужності (Power-down) зберігається вміст регістрів, відключаються всі вбудовані функції до наступного переривання або апаратного скидання. У режимі зниження шумів АЦП (ADC Noise Reduction) зупиняється ЦПУ і всі модулі введення-виведення, крім АЦП, за рахунок чого досягається зниження цифрових шумів під час перетворення АЦП.

Параметри високопродуктивного, економічного 8-розрядного AVR-мікроконтролеру:

- Удосконалена RISC-архітектура.
- Набір з 120 інструкцій, основна кількість яких виконуються за 1 цикл, 32 x 8 універсальних регістрів загального призначення.
- Статична робота.
- Енергонезалежна пам'ять програм і даних.
- Внутрішня системна програмована флеш-пам'ять програм розміром 2, 4, 8 Кбайт (Atiny25, Atiny45, Atiny85) з зносостійкістю 10 тисяч циклів (запис / зтирання).
- 128/256/512 байтів внутрішнього системно-програмованого ЕСППЗУ (Attiny25 / 45/85) з зносостійкістю 100 тисяч циклів (запису / зтирання)
- 128/256/512 байтів внутрішньої статичної ОЗП (ATtiny25 / 45/85).
- Програмовані біти захисту флеш-пам'яті і ЕСППЗУ.

Розглянемо відмінні риси периферійних пристроїв:

- 8-розрядний таймер-лічильник з переддільником і 2 каналами ШІМ.
- 8-розрядний високошвидкісний таймер-лічильник з окремим переддільником.
- 2 високочастотних виходи ШІМ з окремими регістрами задання порогу порівняння.
- Програмований генератор паузи.
- Універсальний послідовний інтерфейс з окремим детектором умови старту
- 10-розрядний АЦП 4 несиметричними каналами, 2 диференційні канали з програмованим посиленням (1x, 20x).
- Програмований сторожовий таймер з окремим вбудованим генератором.
- Вбудований аналоговий компаратор.

										Лист
										56
Вим.	Лист	№ докум.	Підпис	Дата						

Розглянемо спеціальні функції мікроконтролеру, що був використаний в проєктованій системі:

- Вбудована відладна система debugWIRE.
- Внутрішнє системне програмування через порт SPI.
- Зовнішні та внутрішні джерела переривань.
- Економічні режими: холостий хід (Idle), зменшення шумів АЦП (ADC Noise Reduction) і знижена потужність (Power-down).
- Вдосконалена схема скидання при поданні живлення.
- Програмована схема контролю напруги живлення.
- Вбудований калібрований генератор вводу- виводу і корпусу.
- 6 програмованих ліній введення-виведення.
- 8-вивідний корпус PDIP і 8-вивідний SOIC.
- Робоча напруга V-1.8 - 5.5В для ATtiny25 / 45 / 85
V- 2.7 - 5.5В для ATtiny25 / 45/85

- Швидкодія

- Attiny45 / 45 / 85: 0 - 4 МГц при 1.8 - 5.5В,

0 - 10 МГц при 2.7 - 5.5В

- ATtiny25 / 45/85: 0 - 10 МГц при 2.7 - 5.5В,

0 - 20 МГц при 4.5 - 5.5В

- Промисловий температурний діапазон.
- Низький споживаний струм.
- Активний режим: 1 МГц, 1.8В: 450 мкА
- Режим зниженої потужності: 0.1 мкА при 1.8В

Програмований чіп ATmega328P являє собою 28-контактний мікроконтролер, що належить до сімейства AVR. Він застосований для конструктора «Ардуіно», в якості основного компонента електронного набору.

Мікроконтролери сімейства AVR викорисовуються для конструювання електронних пристроїв різної складності. Серія ATmega і ATiny часто застосовуються для апаратного програмування.

Мікроконтролер ATtiny45 має 8-контктний форм-фактор.

Мікросхема ATtiny45 оснащена функціями, властивими AT328P, однак функціональність ATtiny45 обмежена через малу кількість контактів введення-виведення [8].

									Лист
									57
Вим.	Лист	№ докум.	Підпис	Дата					

Програмування ATmega328P і ATtiny45 здійснюється через послідовний периферійний інтерфейс (SPI). Контактна шина містить 3 провідника, плюс «земля». По шині даних слідує сигнали:

- Вхід прийому даних (MOSI)
- Вихід передачі даних (MISO)
- Вхід синхронізації прийому (SCK)

Сигнал SCK (SCLK) генерує ведучий пристрій (програматор), цим сигналом забезпечується синхронна прийом / передача між ведучим і веденим пристроями.

Інтерфейс SPI слід розглядати як «синхронну» комунікаційну шину.

Існує маса програматорів, в яких застосовані ATmega328P і ATtiny45. Наприклад, професійна розробка «Atmel-ICE» є застосуванням чіпів компанії «Atmel».

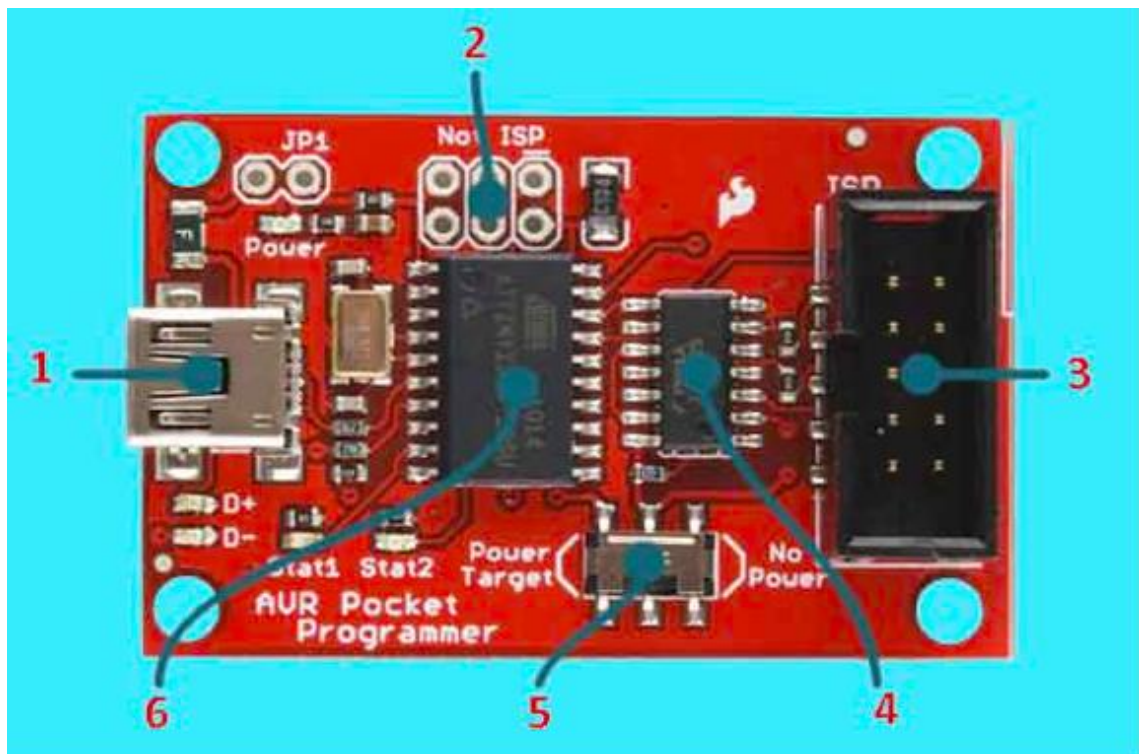


Рисунок 5.1 – Програматор AVR Pocket Programmer

AVR Pocket Programmer:

- 1 - вхідний інтерфейс USB;
- 2 - ISP колектор;
- 3 - вихідний інтерфейс ISP;
- 4 - буферні елементи 74AC125;

5 - перемикач режиму живлення;

6 - основний чіп Attiny45

Сигнальні лінії (MISO, MOSI, SCK) необхідно буферизувати, тому необхідно доповнити схему буфером, використовуючи мікросхему 74AC125.

Даний буфер застосовується на фірмовій збірці. Можливим є використання і транзисторного варіанту буфера.

Для роботи схеми програматора необхідним є використання драйверу під USB. Також може знадобитися використання програмного продукту Zadig, універсального інсталятора під Windows для установки загальних драйверів USB (версія Windows 7 і вище):

- WinUSB
- Libusb-Win32
- Libusb0
- LibusbK

Прошивка та утиліти для «AVR Pocket Programmer», драйвер програматора доцільніш ставити в систему Windows до підключення фізичної схеми до персонального комп'ютеру.

Мікроконтролери AVR програмуються через інтерфейс SPI за допомогою маршрутизації сигналів через шість ліній:

- VCC (напруга живлення),
- GND (сигнальна земля),
- RST (програмний скидання),
- MOSI (прийом даних),
- MISO (передача даних),
- SCK (сигнал синхронізації).

Для маршрутизації сигналів між пристроями, використовується один з двох стандартизованих роз'ємів:

- 6-контактний роз'єм (2 × 3).
- 10-контактний роз'єм (2 × 5).

Дані роз'єми з'єднуються з платою програматора та програмованим пристроєм через шлейф-кабель.

На фірмовій платі програматора присутній перемикач режиму живлення, і якщо він знаходиться в положенні «Power Target», плата має напругу 5В від USB.

						ЕЛІТ 8.171.00.10.376 ПЗ	Лист
Вим.	Лист	№ докум.	Підпис	Дата			59

В положенні перемикача «No Power», живлення на контакті 5В відсутнє. «No Power», використовується для програмування чіпів з низькою напругою (3,3 або 1,8 В).

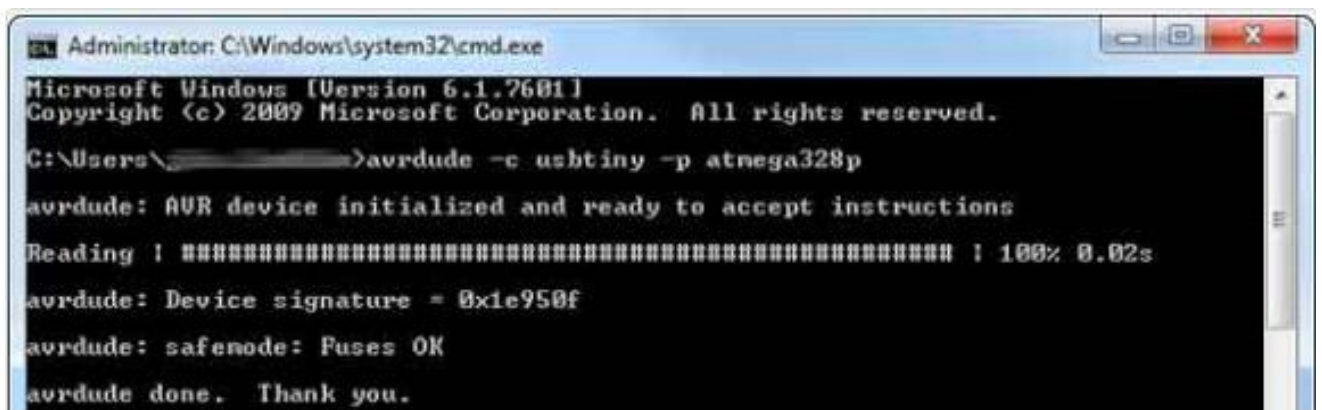
Утиліта AVRDUDE - це інструмент для командного рядка Windows, для його застосування необхідно запусити «командний рядок» Windows (Пуск -> Виконати), після чого перевірити готовність утилітів введенням команди:

```
avrdude -c usbtiny -p atmega328p
```

Командою передбачено наявність програматора мікроконтролера АТmega328Р, який підключено до системного роз'єму, але для чіпа іншої серії команда зміниться. Для МК АТtiny45 командний рядок буде виглядати наступним набором:

```
avrdude -c usbtiny -p t45
```

Якщо схематично підключити програматор і програмована мікросхема в нормі, то утиліта відобразить наступний термінальний текст:



```
Administrator: C:\Windows\system32\cmd.exe
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Users\>avrdude -c usbtiny -p atmega328p

avrdude: AVR device initialized and ready to accept instructions
Reading | ##### | 100% 0.02s
avrdude: Device signature = 0x1e950f
avrdude: safenode: Fuses OK
avrdude done. Thank you.
```

Рисунок 5.3 – Системне підключення програматора

Дана базова команда утиліти забезпечує ідентифікацію підключеного мікроконтролеру AVR. Інструмент AVRDUDE в процесі перевірки зчитує ідентифікатор чіпу.

Ідентифікатори програмованих мікросхем відрізняються в залежності від типу AVR. Мікроконтролер АТmega328Р ідентифікується як 0x1E950F, що і зазначено у вікні видачі [9].

Після перевірки робочого стану схеми, можна читати та записувати програмний код за допомогою утилітів AVRDUDE. Частина пам'яті чіпа, що прошивається даними, це Flash - незалежна область, область зберігання програм

Мікроконтролери сімейства F2MC-16LX / FR є повноцінною заміною широко поширених чіпів ATmega128, ADuC812, PIC16C923 (924), відповідно мікроконтролери Fujitsu необхідно застосовувати в широкому діапазоні електронних розробок.

В результаті використання стандартного інтерфейсу RS-232C та кабелю, розробленого для цього типу комунікації з персональним комп'ютером, проходить процес програмування мікроконтролерів F2MC-16LX / FR сімейства. Дані флеш-пам'яті мікроконтролера в разі необхідності легко можна перепрограмувати. Розробка повинна містити драйвер RS-232C для зв'язку з мікроконтролером по каналу UART.

Виводи мікроконтролера, позначені MD2, MD1, MD0 і Pxx, і вони не підтримують управління системою ПК, в зв'язку з тим, що їх конфігурація розробляється на стороні користувальницької конструкції.

Перед тим, як здійснити послідовне програмування, необхідно вставити для виводу HSTX значення «високого потенціалу» (для мікроконтролерів з виводом HSTX).

В разі переведення контакту RSTX від «низького потенціалу» до «високого потенціалу», після встановлення всіх рівнів для виводів MD2, MD1, MD0, а також Pх0, Pх1, мікроконтролер F2MC переходить до режиму послідовного перепрограмування. Після необхідно запустити програмне забезпечення на ПК. Стандартна програма послідовного програмування від Fujitsu характеризується асинхронним методом програмування мікроконтролера [10].

Для того, щоб підключити чіп до утилітарного програмного забезпечення, необхідне застосування двох ліній послідовного інтерфейсу:

- SOT (Transmit);
- SIN (Receive).

Прошивку Flash-пам'яті здійснює наступна команда:

```
avrdude c usbtiny p atmega328p U flash: w: test.hex
```

або для варіанту з ATtiny45:

```
avrdude c usbtiny p t45 U flash: w: test.hex
```

Примітка: ім'я файлу «test.hex» можна змінити.

									Лист
									61
Вим.	Лист	№ докум.	Підпис	Дата					

Процедура запису Flash-пам'яті вимагає більшого часу, ніж необхідно для читання даних, рядок стану командного терміналу завжди змінюється при читанні, записі, перевірці системи.

```
D:\mega16_kitCD\sample code\ex01_simple_input_output\Exe>avrdude -c usbtiny -p m16 -u -U flash:w:io.hex
found 5 busses

avrdude: AVR device initialized and ready to accept instructions

Reading : ##### | 100% 0.03s

avrdude: Device signature = 0x1e9403
avrdude: NOTE: FLASH memory has been specified, an erase cycle will be performed
        To disable this feature, specify the -D option.
avrdude: erasing chip
avrdude: reading input file "io.hex"
avrdude: input file io.hex auto detected as Intel Hex
avrdude: writing flash (322 bytes):

Writing : ##### | 100% 1.55s

avrdude: 322 bytes of flash written
avrdude: verifying flash memory against io.hex:
avrdude: load data flash data from input file io.hex:
avrdude: input file io.hex auto detected as Intel Hex
avrdude: input file io.hex contains 322 bytes
avrdude: reading on-chip flash data:

Reading : ##### | 100% 1.42s
```

Рисунок 5.4 - Процедура запису Flash-пам'яті

- Опція «U» команди AVRDUDE здійснює управління читанням і записом пам'яті мікропроцесора, нею користувач вказує на роботу з флеш-пам'яттю.

Символом «w» вказується функція запису, а далі через двокрапку вказується ім'я (і розташування) файлу (в прикладі test.hex), вміст якого потрібно записати в пам'ять.

Опція «U» також використовується для читання вмісту пам'яті AVR. Наприклад, наступна команда дозволяє вміст пам'яті чіпа AVR і зберігаємо в файлі під назвою «test.hex»:

```
avrdude c usbtiny p atmega328p U flash: r: test.hex: r
```

або для варіанту ATtiny45:

```
avrdude c usbtiny p t45 U flash: r: test.hex: r
```

Даний інструмент підтримує роботу тільки з файлами, що мають розширення «*.hex».

Для роботи мікроконтролеру необхідно підвести живлення. Напруга живлення для МК Atmel AVR знаходиться в межах 1.8- 5 В, в залежності від серії і моделі. AVR можуть працювати при напрузі від 5 В, тому в подальшому будемо вважати напругу живлення мікроконтролеру завжди рівною 5 В. Плюс напруги живлення зазвичай позначається як Vcc. Нульовий вивід позначають GND. Беручи до уваги приклад компоного блоку живлення, чорний провід це GND (так як провід заземлення в основному фарбують в чорний колір), а червоний це +5, буде нашим Vcc.

- Operating Voltages

- 1.8 - 5.5V (ATtiny45V)

- 2.7 - 5.5V (ATtiny45)

- Speed Grades

- ATtiny453V: 0 - 4 MHz @ 1.8 - 5.5V, 0 - 10 MHz @ 2.7 - 5.5V

- ATtiny45: 0 - 10 MHz @ 2.7 - 5.5V, 0 - 20 MHz @ 4.5 - 5.5V

Для роботи контролерів серії AVR достатньо живлення. На всі входи Vcc необхідно здійснити подання напруги 5 В, а всі входи GND заземлити

. МК може мати безліч входів Vcc і багато входів GND. Велика кількість виводів необхідна з метою рівномірного заживлення кристала з усіх боків, для запобігання перевантаження внутрішнього ланцюга живлення.

AGND і AVCC - це аналогові земля і живлення для аналого-цифрового перетворювача, що є дуже точним вимірювачем напруги, тому існує необхідність здійснювати живлення через додаткові фільтри, для уникнення впливу перешкод на якість вимірювання. Для цього в точних схемах проводять поділ землі на цифрову і аналогову (що з'єднані в одній точці), а на AVCC здійснюється подання напруги через фільтруючий дросель.

В разі відсутності необхідності використання АЦП та відсутності необхідності проведення точних вимірювань, цілком допустимим є подання на AVCC напруги 5 В, що і на Vcc, та заземлення AGND . Їх підключення є обов'язковим.

В чіпу Attiny45 є помилка на рівні топології - Vcc і AVcc зв'язані між собою всередині кристалу, з опором між ними 5 Ом. В Attiny45 і Attiny85 між Vcc і AVcc опір більший. Провід Vcc до програматора наведено пунктирною лінією, так як він не є обов'язковим.

										Лист
										63
Вим.	Лист	№ докум.	Підпис	Дата						

При включенні схеми, якщо конденсатор розряджений та напруга на RST близька до нуля - мікроконтролер не запускається, відбувається безперервне скидання. Після цього через деякий час, через резистор, відбувається зарядка конденсатора після чого напруга на RST досягне лог1 - мікроконтролер запуститься. За допомогою кнопки, в разі необхідності, можна зробити примусове скидання.

Затримка розраховується таким чином: $T = R * C$ для даної системи і дорівнює приблизно 1 с. Затримка потрібна для того, щоб мікроконтролер не стартував раніше ніж всі девайси плати живлення вийдуть на постійний режим. Для AVR затримку старту, можна зробити.

Необхідним є джерело тактового сигналу. Тактовий генератор - це серце мікроконтролера, в зв'язку з тим, що при кожному імпульсі відбувається якась операція всередині і чим швидша тактова частота, тим швидше МК виконує свої дії і та використовує електроенергії (на перемикання логічних вентилів потрібна енергія).

Імпульси задаються тактовим генератором, що вбудований в мікроконтролер, але генератор може бути зовнішнім, швидкість якого залежить від налаштувань мікроконтролера і вимог.

Генератор може бути:

- Внутрішнім з внутрішньої задається RC ланцюжком.

Виводи XTAL1 та XTAL2 можна використовувати як звичайні порти введення-виведення, як правило обирається одне з 4-х значень внутрішньої частоти.

- Внутрішнім з зовнішньої задається RC ланцюжком.

Зовні підключається мікроконтролер, конденсатор, резистор, можлива зміна на ходу тактової частоти, змінюючи значення резистора.

- Внутрішнім із зовнішнім задається кварцом.

Зовні ставиться кварцовий резонатор і пара конденсаторів (в випадку коли кварц низькочастотний (до 1МГц) то конденсатори не ставлять.

- Зовнішнім.

Необхідний в випадку коли кілька мікроконтролерів повинні працювати в жорсткій синхронізації від одного генератора [11].

У різних схем є різні переваги:

						ЕЛІТ 8.171.00.10.376 ПЗ	Лист
Вим.	Лист	№ докум.	Підпис	Дата			64

світлодіод може згоріти, можливе запалення виводу мікроконтролеру, який може витримати струм 20-30мА. А для нормального світіння світлодіоду необхідний струм 3 - 15мА.

Напруга на виході ноги МК близько 5 В, а падіння напруги на світлодіоді максимум 2,5 В. Таким чином, напруга яка повинна бути під контролем обмежувального резистору дорівнює $5 - 2.5 = 2.5\text{В}$.

Струм нам необхідний 5мА. Відповідно опір $R = U / I = 2.5 / 5\text{E-3} = 500\text{Ом}$. Найближчий по ряду це 510 Ом. Його і будемо використовувати.

Іноді є необхідність в створенні пристрою, який може працювати не тільки автономно, але і під управлінням комп'ютера. Іноді існує потреба в оперативному зчитуванні інформації з пристрою на мікроконтролері або задача полягає в розробці мікроконтролерної приставки для сполучення комп'ютера і та зовнішньою системою, якою він керує. Завжди є можливість організувати обмін інформацією між двома комп'ютерами на основі мікропроцесорних технологій (комп'ютер і мікроконтролер).

Проблема з'єднання мікроконтролера і комп'ютера завжди була актуальною. Щодо комп'ютерів PC XT, одним з простих і ефективних способів підключення до нього пристрою, було самостійне виготовлення нестандартного внутрішнього модуля, що був встановлений в шину ISA всередині корпусу комп'ютера на материнську плату. Шина ISA застосовувалась тоді для підключення всіх пристроїв розширення, таких як відеокарти, модулі додаткової пам'яті, контролери жорстких дисків і т.д. Роз'єм для підключення до такої шини можна було виготовити самостійно, щоявляв собою просто виступ друкованої плати вставного модуля з доріжками замість контактів. Логіка роботи шини була простою і дозволяла підключитися до неї, з використанням простих схем на цифрових елементах середньої інтеграції. В наш час на зміну шині ISA прийшла шина, під назвою PCI. В ній використовуються дрібні контактні ділянки та логіка її роботи була більш складною. Для підключення зовнішніх пристроїв необхідним стало використання одного із стандартних інтерфейсів введення- виведення. Наступні три варіанти, завдяки яким можливе підключення: підключення по LPT, підключення за допомогою COM-порту, підключення по USB.

- Розглянемо підключення за допомогою LPT порту.

Це найпростіший з точки зору програмної і апаратної реалізації спосіб. LPT порт комп'ютера організован таким чином, що є можливість програмним шляхом

						ЕЛІТ 8.171.00.10.376 ПЗ	Лист
Вим.	Лист	№ докум.	Підпис	Дата			66

мікросхеми розташовані 2 перетворювача, необхідних для отримання необхідних робочих напруг + 12В і 12В.

- Підключення за допомогою USB порту

Це найперспективніший спосіб підключення, а ще і найскладніший з точки зору програмної реалізації. USB порт - універсальний послідовний порт для підключення всіх видів зовнішніх периферійних пристроїв, за допомогою якого до комп'ютера можливе підключення зовнішніх накопичуваних на жорстких дисках та флеш-пам'яті, джойстики, мишки, звукові системи, WEB-камери, MP3 плеєри і навіть зовнішні TV-тюнери. Стандарт USB версії 2.0 підтримує передачу даних зі швидкістю від 1.5 Мбіт / сек до 480 Мбіт / сек. Протокол USB - це багаторівневий протокол передачі інформації, найнижчий рівень якого визначає правила щодо формування імпульсів: тривалість, розмах, спосіб кодування даних, метод синхронізації, метод перевірки помилок. Протокол USB високого рівня - це набір команд, що дозволяють опитувати підключені до порту зовнішні пристрої, отримувати у цих них інформацію щодо їх типу, назви, виробника, режимах роботи. Важливою проблемою для даного способу підключення є занадто висока швидкість роботи USB інтерфейсу, що створює обмеження.

Для підключення мікроконтролера до комп'ютера по каналу USB існує три способи.

- 1) Використання додаткової мікросхеми - перетворювача USB – FIFO.

Дана мікросхема є подальшим вдосконаленням попереднього способу, наприклад мікросхема FT245R фірми Future Technology Devices International Ltd, яку зручно застосовувати в випадку, коли мікроконтролер не має вбудованого послідовного інтерфейсу, що сумісний з RS232. FT245R емулює віртуальний COM порт в комп'ютері, надіслані дані комп'ютером в цей порт записуються всередині мікросхеми в буфер, що пораджує за принципом черги. Такий буфер є FIFO (скорочення від «First In, First Out»: перший увійшов, перший вийшов). Інформація зберігається в буфері до тих пір, доки не буде зчитана звідти мікроконтролером.

- 2) Апаратна реалізація USB порту.

Це найдешевший спосіб підключення мікроконтролера до комп'ютера за допомогою USB, який передбачає наявність лише самого контролера. USB протокол реалізується програмним шляхом. Даний спосіб підключення має обмеження. По-перше, це примусове розділення його роботи на цикл обміну

					ЕЛІТ 8.171.00.10.376 ПЗ	Лист
Вим.	Лист	№ докум.	Підпис	Дата		68

інформацією по USB та цикл виконання основного завдання. По-друге, асинхрон за рівнями сигналів. Рівень сигналу логічної одиниці порту USB повинен бути в межах 3-3,6, а порту вводу-виводу мікроконтролера дрівнює напрузі живлення +5 В.

3) Мікроконтролер містить вбудований апаратний USB інтерфейс.

Використання мікросхеми з вбудованим USB каналом - це самий доцільний підхід з точки зору якості і надійності роботи всієї системи. Недолік його – висока вартість [12].

					ЕЛІТ 8.171.00.10.376 ПЗ	Лист
Вим.	Лист	№ докум.	Підпис	Дата		69

6 ЕКОНОМІЧНА ЧАСТИНА

6.1 Розрахунок трудомісткості та заробітної плати

Трудомісткість – це кількість робочого часу, що необхідний на виготовлення одиниці продукції або послуг. Показник трудомісткості показує прямопропорційний зв'язок між обсягом виробництва та трудовитратами.

Продуктивність праці характеризує ефективність, результативність, визначається кількістю продукції, виробленої в одиницю робочого часу, витраченою працею на одиницю виготовленої продукції, або виконаних робіт.

Класифікують на продуктивність живої і продуктивність суспільної праці.

Продуктивність живої праці визначається витратами робочого часу в кожному окремому виробництві. Продуктивність суспільної праці - витратами живої і уречевленої праці.

Для підприємства, продуктивність праці - ефективність витрат тільки живої праці і обчислюється через показник виробітку і трудомісткості продукції, між якими є зворотнопропорційна залежність.

Трудомісткість визначається витратами живої праці на виготовлення продукції. Враховується не лише час виконання роботи, але й час перерв, що регламентуються на відпочинок та перерви, зумовлені технологією і організацією виробництва.

Існують наступні види трудомісткості:

- технологічна трудомісткість (витрати праці основних робітників, відряджених і погодинних);
- трудомісткість обслуговування (витрати праці допоміжних робітників);
- виробнича трудомісткість;
- трудомісткість управління (витрати праці керівників, фахівців, технічних виконавців).

Розрахунок трудомісткості виготовлення проектованої системи наведено в таблиці 6.2.

					ЕЛІТ 8.171.00.10.376 ПЗ	Лист
						70
Вим.	Лист	№ докум.	Підпис	Дата		

Таблиця 6.1- Трудоміскість виготовлення проектованої системи

№ операції	Назва операцій	Штучний час, н-год	Програма випуску, шт.	Об'єм, н-год
5	Заготівельна	0,002483	1	0,002483
10	Контрольна	0,0028		0,0028
15	Отримання заготовок	0,00405		0,00405
20	Термостабілізація заготовок	0,00312		0,00312
25	Свердлування	0,0101		0,0101
30	Контрольна	0,00465		0,00465
35	Підготовка поверхні	0,00653		0,00653
40	Нанесення світлочутливого шару	0,00155		0,00155
45	Експонування	0,0147		0,0147
50	Проявлення рисунку	0,00218		0,00218
55	Хімічна металізація	0,0259		0,0259
60	Контрольна	0,00623		0,00623
65	Обрізання заготовки	0,00375		0,00375
70	Контрольна	0,00299		0,00299
75	Обробка контура друкованої плати	0,00243		0,00243
80	Контрольна	0,01144		0,01144
Разом		0,104903		0,104903

Заробітна плата – це винагорода , яка вимірюється , як правило у грошовому виразі і яку за трудовим договором власник виплачує працівнику за виконуваним ним роботу.

Фонд основної заробітної плати включає нарахування за виконану роботу відповідно до встановлених норм праці (норми часу, вироблення, обслуговування) по тарифних ставках, відрядних розцінках або посадових окладах (для керівників, фахівців, технічних службовців), включаючи в повному об'ємі внутрішню сумісницю.

Фонд додаткової заробітної плати включає доплати, надбавки до тарифних ставок (окладам) за працю понад встановлену норму в розмірах, передбачених чинним законодавством, премії і винагороди, зокрема за вислугу років, що мають систематичний характер, оплата роботи в наднормовий час, в святкові і вихідні дні (у розмірах, встановлених чинним законодавством) і ін.

										Лист
										71
Вим.	Лист	№ докум.	Підпис	Дата						

Таблиця 6.2 – Розрахунок основної заробітної плати виробничих робітників

№ операції	Назва операції	Розряд	Трудомісткість, н-год	Годинна тарифна ставка, грн.	Сума, грн.
5	Заготівельна	3	0,002483	28	0,06952
10	Контрольна	3	0,0028	28	0,0784
15	Отримання заготівок	3	0,00405	28	0,1134
20	Термостабілізація заготівок	3	0,00312	28	0,08736
25	Свердлування	3	0,0101	28	0,2828
30	Контрольна	3	0,00465	28	0,1302
35	Підготовка поверхні	3	0,00653	28	0,18284
40	Нанесення світлочутливого шару	3	0,00155	28	0,0434
45	Експонування	3	0,0147	28	0,4116
50	Проявлення рисунку	3	0,00218	28	0,06104
55	Хімічна металізація	3	0,0259	28	0,7252
60	Контрольна	3	0,00623	28	0,17444
65	Обрізання заготівки	3	0,00375	28	0,105
70	Контрольна	3	0,00299	28	0,08372
75	Обробка контура друкованої плати	3	0,00243	28	0,06804
80	Контрольна	3	0,01144	28	0,32032
Разом					1,04903
Тарифна заробітна плата					2,937284
Доплати до тарифної заробітної плати					2,9
Основна заробітна плата					4

Додаткова зарплата $ЗП_{д}$, грн. (оплата за відпустоку, виконання додаткових функцій та ін.) обчислюється за наступною формулою:

$$ЗП_{д} = \frac{ЗП_{осн} \times K_{д\%}}{100}, \quad (6.1)$$

де $ЗП_{осн}$ – основна заробітна плата, грн.;

$K_{д\%}$ - процент витрат на додаткову заробітну плату, %.

$$ЗП_{д} = \frac{4 \times 18}{100} = 0,72 \text{ грн.}$$

					ЕЛІТ 8.171.00.10.376 ПЗ	Лист
Вим.	Лист	№ докум.	Підпис	Дата		72

6.2 Розрахунок собівартості проектованої системи

Собівартість – це сума витрат на виробництво і реалізацію одиниці продукції.

Вона відбиває вартість спожитих фондів (матеріали, що комплектують, амортизації будинків і фондів).

Собівартість – зовнішній показник, тому що від неї залежить прибуток і рентабельність.

Розрахунок собівартості може вироблятися декількома способами:

- калькулювання;
- укрупнені методи (параметричний, питомих ваг, агрегатний і інші).

Стаття «Сировина й матеріали» містить витрати на сировину, основні, допоміжні матеріали, тобто витрати, які розраховуються на одиницю продукції на основі цін.

Також беруться до уваги транспортно-заготівельні витрати (плата за транспортування, вантажно-розвантажувальні роботи, комісійні виплати заготівельним організаціям та ін.)

Таблиця 6.3 – Розрахунок транспортно-заготівельних витрат

Назва витрат	Сума, грн.
Основні та допоміжні матеріали	269,8
Покупні та комплектуючі вироби	1562,0
Разом	1831,8
Транспортно-заготівельні витрати %	201,5
Зворотні відходи %	5,7
Всього	2027,6

6.3 Розрахунок матеріальних витрат

Витрати на енергію, що є використаною для технологічного процесу для зміни стану, та розраховується за нормами витрат та тарифами на енергію.

Витрати на енергію C_E , грн., визначаються за формулою:

$$C_E = \sum_{i=1}^n H_E \times C_E, \quad (6.2)$$

де n – вид енергії;

C_E – витрати на енергію, грн.;

H_E – норми витрат окремих видів енергії;

C_E – середня ціна одиниці енергії.

$$C_E = 6,341 \times 9,467 = 60 \text{ грн}$$

Відрахування на соціальні заходи $C_{\text{соц}}$, грн., розраховуються за формулою:

$$C_{\text{соц}} = \frac{(ЗП_{\text{осн}} + ЗП_{\text{д}}) \times K_{\text{сз}\%}}{100}, \quad (6.3)$$

де $K_{\text{сз}\%}$ - процент відрахувань на соціальні заходи, %.

$$C_{\text{соц}} = \frac{(4 + 0,72) \times 36,76}{100} = 1,74 \text{ грн.}$$

На одиницю кожного різновиду продукції витрати на утримання та експлуатацію машин і устаткування обчислюються методом розподілу, зазвичай, пропорційно основній зарплаті робітників.

Витрати на експлуатацію машин і устаткування $ВУЕО$, грн., обчислюються за формулою:

$$ВУЕО = \frac{ЗП_{\text{осн}} \times \%_{\text{ВУЕО}}}{100}, \quad (6.4)$$

						Лист
					ЕЛІТ 8.171.00.10.376 ПЗ	74
Вим.	Лист	№ докум.	Підпис	Дата		

де $\%_{\text{ВУЕО}}$ – процент витрат по утриманню і експлуатації обладнання, %.

$$\text{ВУЕО} = \frac{4 \times 95}{100} = 3,8 \text{ грн.}$$

Загальновиробничі витрати — це витрати на управління, виробниче й господарське обслуговування цеху.

Загальновиробничі витрати ЗВВ, грн., розраховуються за формулою:

$$\text{ЗВВ} = \frac{\text{ЗП}_{\text{осн}} \times \%_{\text{ЗВВ}}}{100}, \quad (6.5)$$

де $\%_{\text{ЗВВ}}$ – процент загальновиробничих витрат.

$$\text{ЗВВ} = \frac{4 \times 72}{100} = 2,88 \text{ грн.}$$

Загальногосподарські витрати обчислюються за наступною формулою:

$$\text{ЗГВ} = \frac{\text{ЗП}_{\text{осн}} \times \%_{\text{ЗГВ}}}{100}, \quad (6.6)$$

де $\%_{\text{ЗГВ}}$ – процент загальногосподарських витрат, %.

$$\text{ЗГВ} = \frac{4 \times 58}{100} = 2,32 \text{ грн.}$$

					ЕЛІТ 8.171.00.10.376 ПЗ	Лист
Вим.	Лист	№ докум.	Підпис	Дата		75

6.4 Калькуляція собівартості виробу

Калькуляцію собівартості виробу заносимо в таблицю 5.

Таблиця 5 – Калькуляція виробу

№	Назва статті	Сума, грн.
1	Сировина та матеріали	270
2	Купівельні напівфабрикати, комплектуючі вироби	1562
3	Транспортно-заготівельні витрати	202
4	Зворотні відходи (вираховуються)	20
5	Енергія технологічна	60
Разом матеріальні витрати		2094
6	Основна заробітна плата	4
7	Додаткова заробітна плата	0,72
8	Відрахування на соціальні заходи	0,74
9	Витрати на утримання та експлуатацію машин і обладнання	3,8
10	Загальновиробничі витрати	2,32
Цехова собівартість		2106,5
11	Загальногосподарські (адміністративні) витрати	2,32
Виробнича собівартість		2207,6
12	Позавиробничі витрати (витрати на збут)	20
Повна собівартість		2227,6

6.5 Визначення гуртової ціни виробу

Ціна – це грошовий вираз вартості товару, кількість грошей, що сплачується за одиницю товару або послуги.

Величина прибутку розраховується за формулою:

$$П = СВ_{п} \times \frac{R}{100'} \quad (6.7)$$

де Π – величина прибутку, грн.

$СВ_{\pi}$ – сумарні витрати на виготовлення і реалізацію продукції, грн.

R – рентабельність продукції, %.

$$\Pi = 2106,46 \times \frac{19}{100} = 400,23 \text{ грн.}$$

Гуртова ціна випуску виробу без ПДВ $\Pi_{\text{гурт}}$, грн., розраховується за формулою:

$$\Pi_{\text{гурт}} = СВ_{\pi} + \Pi, \quad (6.8)$$

$$\Pi_{\text{гурт}} = 2106,46 + 400,23 = 2506,7 \text{ грн.}$$

Грошове вираження ПДВ, грн., розраховується за формулою:

$$\text{ПДВ} = \Pi_{\text{гурт}} \times \frac{\%_{\text{ПДВ}}}{100}, \quad (6.9)$$

де $\%_{\text{ПДВ}}$ – податок на додану вартість, %.

$$\text{ПДВ} = 2506,7 \times \frac{20}{100} = 501,34 \text{ грн.}$$

Вільна відпускна ціна з урахуванням ПДВ:

$$\Pi_{\text{гурт}} (\text{з ПДВ}) = \Pi_{\text{гурт}} + \text{ПДВ}, \quad (6.10)$$

$$\Pi_{\text{гурт}} (\text{з ПДВ}) = 2506,7 + 501,34 = 3008 \text{ грн.}$$

Отже для встановлення відповідної норми рентабельності потрібно встановити ціну в 3008,16 грн. за одну систему.

					ЕЛІТ 8.171.00.10.376 ПЗ	Лист
Вим.	Лист	№ докум.	Підпис	Дата		77

Висновок: в ході роботи, була розрахована собівартість проектованої системи, складена зведена таблиця калькуляції собівартості з якої видно що для того щоб зменшити собівартість продукції потрібно зменшити виробничі витрати і позавиробничі витрати.

					ЕЛІТ 8.171.00.10.376 ПЗ	Лист
Вим.	Лист	№ докум.	Підпис	Дата		78

7 РОЗРОБКА ПРОГРАМ ОБРОБКИ ДАНИХ І УПРАВЛІННЯ СИСТЕМОЮ

```
/* Blink
   Turns on an LED on for one second, then off for one second, repeatedly.
   This example code is in the public domain.
*/

// ATtiny84/ATtiny44 does not have Pin 13, so we use pin 0 instead.
// A current limiting resistor should be connected in line with the LED.
int led = 0;

// the setup routine runs once when you press reset:

void setup() {
  // initialize the digital pin as an output.
  pinMode(led, OUTPUT);
}

// the loop routine runs over and over again forever:
void loop() {
  digitalWrite(led, HIGH); // turn the LED on (HIGH is the voltage level)
  delay(1000);             // wait for a second
  digitalWrite(led, LOW);  // turn the LED off by making the voltage LOW
  delay(1000);             // wait for a second
}
```

Рисунок 7.1 – Приклад налаштування Arduino IDE для прошивки Attiny

Програма для завантаження на ATtiny отримана і далі треба здійснити підготовку Arduino до використання в якості програматора:

В підміню "Сервіс → Плата" обираємо мікроконтролер, який треба запрограмувати, як приклад, "Attiny45 @ 1Mhz (internal oscillator, BOD disabled)".

Схема підключення ATtiny84 / ATtiny45 до плати Arduino зображено на рисунку 7.2.

									Лист
									79
Вим.	Лист	№ докум.	Підпис	Дата	ЕЛІТ 8.171.00.10.376 ПЗ				

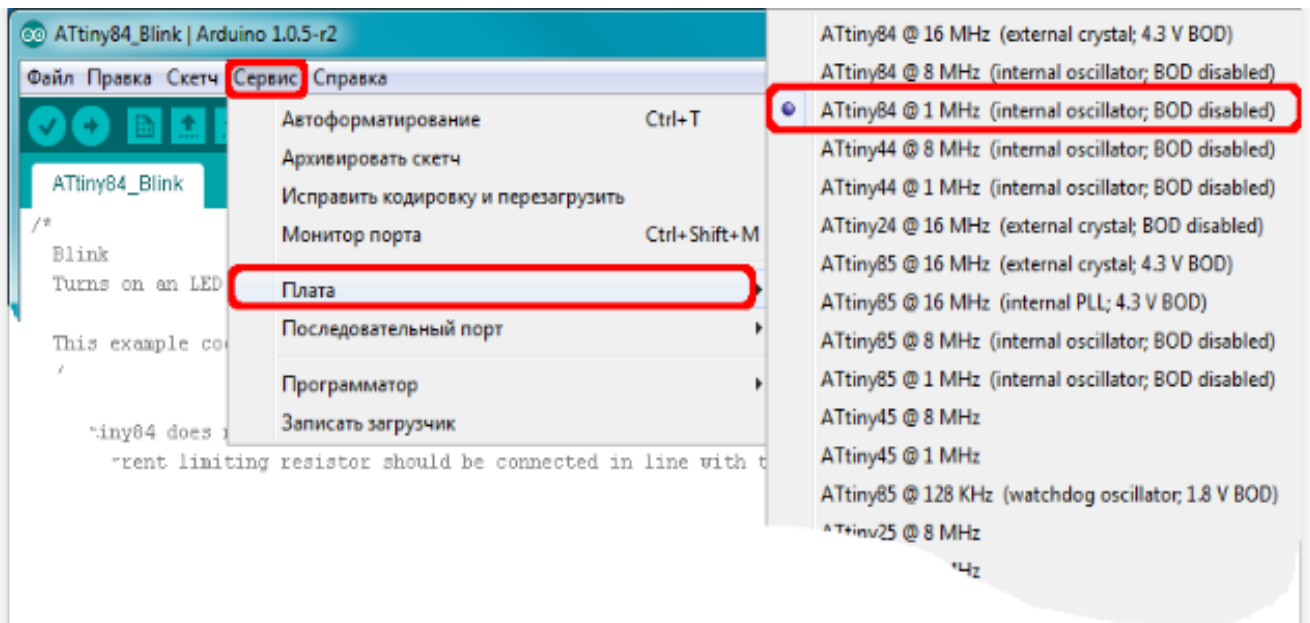


Рисунок 7.2 – Підготовка Arduino

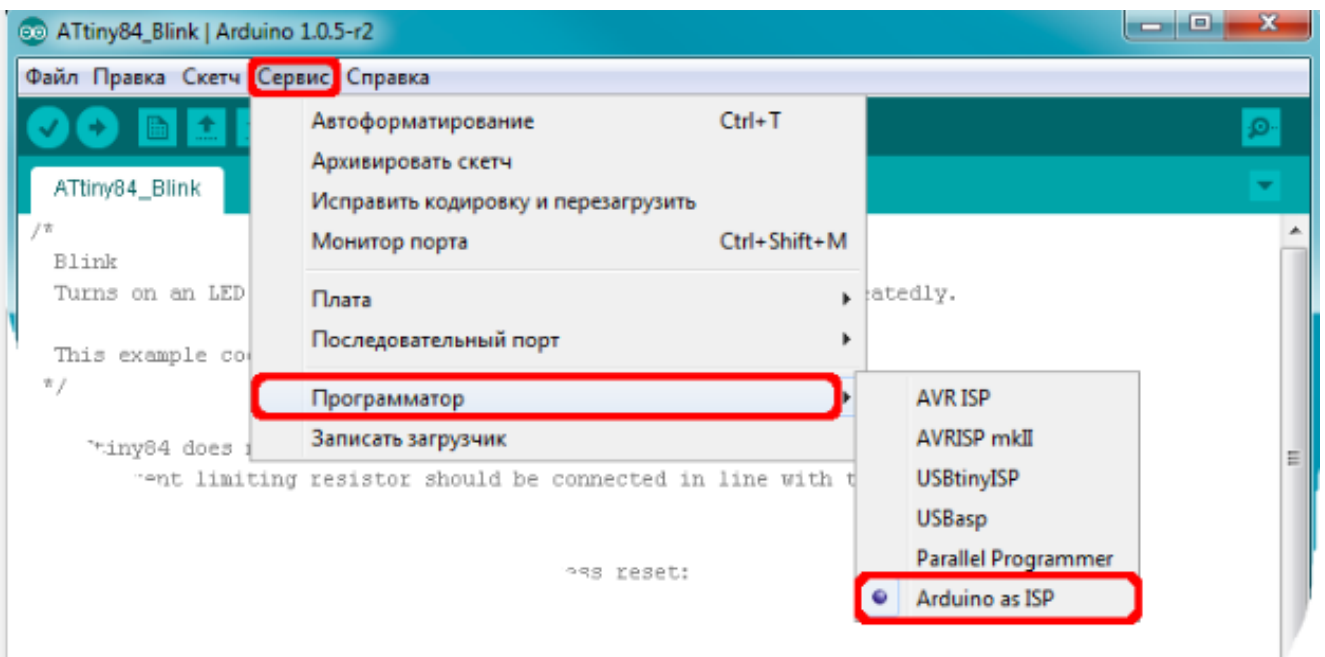


Рисунок 7.3 – Підготовка Arduino (Сервіс)

Для роботи з мікроконтролером треба розуміти призначення виходів. На рисунку 7.4 вказано, яким чином повинні бути підключені виводи плати Arduino до ATtiny45.

Після завершення всіх необхідних підключень, можна виконати завантаження скетчу на мікроконтролер ATtiny (натиснути кнопку "Завантажити").

Якщо завантаження скетчу на ATtiny84 / ATtiny45 пройшла успішно, то можна приступити до тестування:

5. Вимкнути плату Arduino від живлення. Відповідно до завантажених скетчів, необхідно підключити анод (+) світлодіода до виходу під номером 2 мікроконтролера ATtiny45. Катод (-) світлодіода - коротка ніжка - повинен бути підключений до резистора (в даному випадку з опором 180 Ом, що буде виконувати функцію обмежувача струму. Інша ніжка резистора підключається до Землі (GND) плати Arduino.
6. Для живлення мікроконтролера використовується плата Arduino Uno, а саме вивід 3.3V, який підключено до виводу під номером 1 мікроконтролера ATtiny45.
7. Вивід Земля (GND) з плати Arduino повинен бути підключений до виходу під номером 14 мікроконтролера ATtiny45.
8. Підключення плати Arduino до живлення.

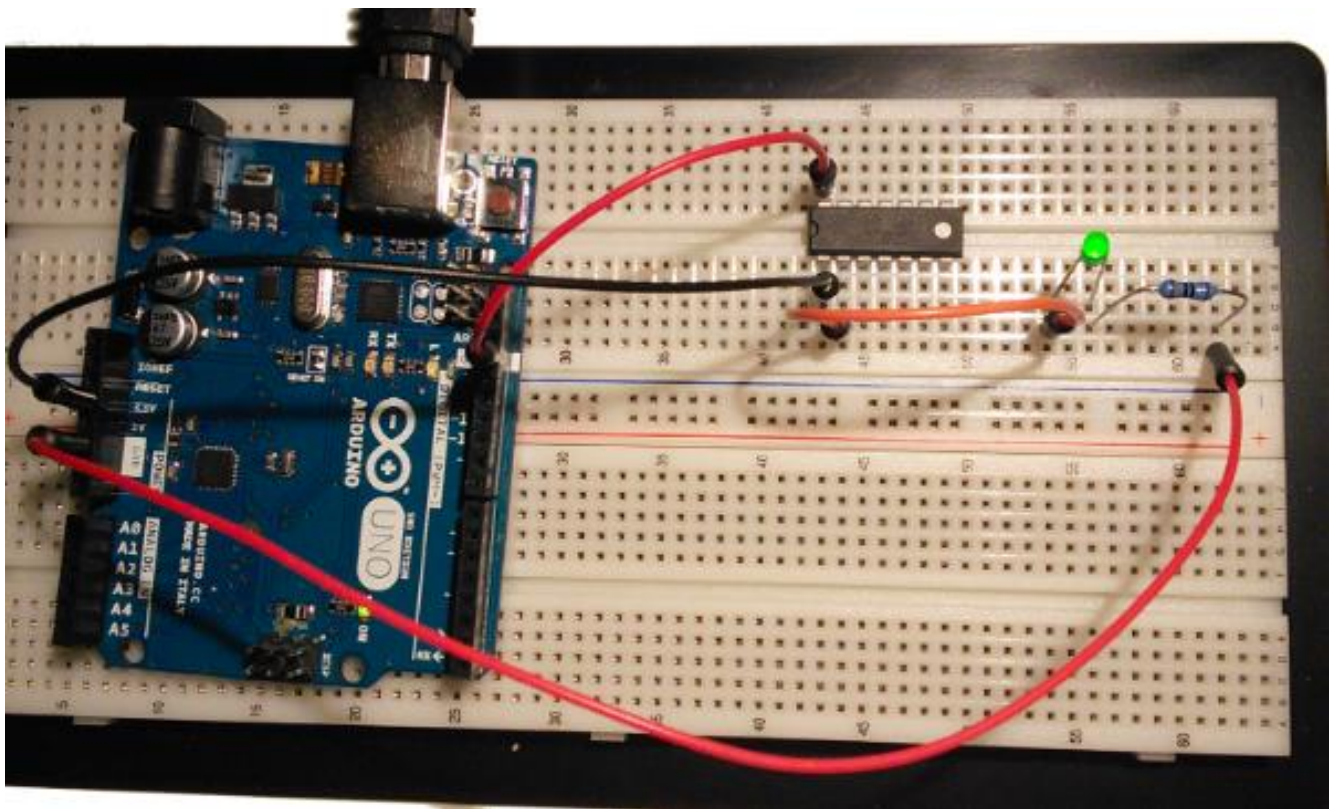


Рисунок 7.7 – Підключення Arduino (Кінцевий етап)

					ЕЛІТ 8.171.00.10.376 ПЗ	Лист
Вим.	Лист	№ докум.	Підпис	Дата		83

При роботі з мікроконтролерами ATtiny необхідно бути обережним, так як не всі бібліотеки можуть бути використані, треба перевірити форум Arduino на наявність інформації про сумісність і можливі проблеми.

У всіх мікроконтролерах сімейства mega є апаратний модуль UART, чого немає в мікроконтролерах сімейства tiny. Програмний UART може бути використаний для молодшого сімейства.

Проекти написані як для ATtiny45 так і для ATmega16. Відмінності між ними незначні, стосуються ініціалізації та переривання задіяного таймера.

Важливим є створення заголовків softUart.h. Включення заголовків compilers_3.h, забезпечує сумісність з трьома AVR компіляторами (IAR, CodeVision, GNU GCC), макроозначеннями та прототипами функцій користувача. Прототипи, слідуєть своєму стилю, з використанням приставки ("SUART_"), що визначає до яких файлів віднести необхідні функції.

```
#ifndef SOFT_UART_H
```

```
#define SOFT_UART_H
```

```
#include "compilers_3.h"
```

```
/ * _____ * /
```

Налаштування

```
/* Тактова частота мікроконтролера */
```

```
#ifndef F_CPU
```

```
#define F_CPU 8000000L
```

```
#endif
```

```
/* Прескалер таймеру 1L, 8L, 64L, 256L, 1024L */
```

```
#define SUART_PRESCALER 1L
```

```
/* Швидкість обміну даними */
```

```
#define SUART_BAUD_RATE 19200L
```

```
/* Місткість приймаючого буферу */
```

```
#define SUART_BUF_SIZE 32
```

									Лист
									84
Вим.	Лист	№ докум.	Підпис	Дата					

```

/* Пін приймача */
#define RX_PINX PINB
#define RX_PORTX PORTB
#define RX_DDRX DDRB
#define RX_PIN 1

/* Пін передавача */
#define TX_PORTX PORTB
#define TX_DDRX DDRB
#define TX_PIN 0

/* Функція або макрос, що виконується під час
очікування надходження даних в приймальний буфер.
SUART_GetChar () */
#define SUART_Idle ()

/* _____ Функції для _____ */

void SUART_Init (void);
char SUART_GetChar (void);
void SUART_PutChar (char ch);
void SUART_PutStr (char * str);
void SUART_PutStrFl (char __flash * str);
void SUART_FlushInBuf (void);
char SUART_Kbhit (void);
void SUART_TurnRxOn (void);
void SUART_TurnRxOff (void);

/
_____ */

#endif // SOFT_UART_H

```

В даній роботі було взято приклад ескізу Servo -> Sweep з заміною рядку, в

						ЕЛІТ 8.171.00.10.376 ПЗ	Лист
Вим.	Лист	№ докум.	Підпис	Дата			85

якому міститься серво-змінна, з зміною номеру штиря, здійснене додавання декількох запитів до SoftwareServo :: refresh (), з урахуванням рекомендацій в документації Software Servo. Маємо наступний код:

```
#include <SoftwareServo.h>
SoftwareServo myservo; // створення сервооб'єкту для керування сервоприводом
// можливим є створення максимум 8 сервооб'єктів
int pos = 0; // змінна для збереження положення сервоприводу
void setup ()
{
myservo.attach (1); // прикріплення сервоприводу на контакт 1
}

void loop ()
{
для (pos = 0; pos <180; pos += 1) // йде від 0 - 180 градусів
{// з кроком в 1 градус
myservo.write (пози); // передача сервокерування в положення в змінної 'pos'
затримка (15); // очікування 15 мс, щоб сервопривід досяг позиції
SoftwareServo :: оновлення ();
}
для (pos = 180; pos >= 1; pos = 1) // йде від 180 градусів до 0 градусів
{
myservo.write (пози); // передаємо сервокерування в положення в змінної 'pos'
затримка (15); // очікування 15 мс, щоб сервопривід досяг позиції
SoftwareServo :: оновлення ();
}
}
```

Для отримання коду на чіпі, доцільним є використання програмного забезпечення від MIT, що здійснює підтримку ATtiny 45 і ATtiny 85 в середовище Arduino. Це робить можливим вибір ATtiny 85 з меню «Сервіс ->» в Arduino. Файли підтримки розташовані за наступним посиланням: <http://hlt.media.mit.edu/?p=1229>

З використанням програмного коду AVR, стало можливим завантаження

						ЕЛІТ 8.171.00.10.376 ПЗ	Лист
Вим.	Лист	№ докум.	Підпис	Дата			86

прямо в 45, обравши Tools -> Board -> AT Tiny 45 (w USB Tiny ISP).

Для завантаження коду, треба натиснути кнопку «Перевірити», а потім натиснути кнопку «Завантажити».

Необхідним є підключення сервоприводу до 5 В та заземлення, перед підключенням контрольного проводу (жовтого / білого проводу, що йде від сервомотора), до контакту 6 на ATTiny 85 (PB1 - той самий вивід з жовтим кабелем MISO).

Макроозначення F_CPU, SUART_PRESCALER і SUART_BAUD_RATE використовуються в макросі для розрахунку таймерної константи, що забезпечує задану швидкість обміну по UART'у. Константа SUART_PRESCALER може приймати тільки ті значення, що записані в коментарі - 1L, 8L, 64L, 256L, 1024L. При заданні інших значень, компілятор буде видавати помилку, також якщо і при розрахунку таймерної константи будуть отримані неприпустимі значення - нуль і значення більше 256, і буде отримано попередження, в випадку коли швидкість програмного UART`а буде на 1% більше або менше ніж необхідне значення.

Також в заголовки біли додані ті визначення, які задають приймальний і передавальний вивід мікроконтролера і макроси для розрахунку і перевірки таймерної константи. Макроси дозволяють швидко підібрати оптимальне значення частоти мікроконтролера, переддільника таймера та швидкості передачі даних.

```
#define TIME_VALUE (F_CPU / (SUART_BAUD_RATE * NUM_OF_CYCLES *  
SUART_PRESCALER))  
#define TIMER_TCNT_VALUE (0xff - (TIME_VALUE - 1))  
#define ERROR_VALUE ((F_CPU * 100) / (TIME_VALUE * SUART_BAUD_RATE  
* NUM_OF_CYCLES * SUART_PRESCALER))  
#if ((TIME_VALUE > 256) || (TIME_VALUE == 0))  
  #error "TIMER_TCNT_VALUE is not correct. Change PRESCALER, BAUD_RATE  
or F_CPU"  
#endif  
  
#if ((ERROR_VALUE > 100) || (ERROR_VALUE < 100))  
  #warning "Baud rate error > 1% "  
#endif
```

						ЕЛІТ 8.171.00.10.376 ПЗ	Лист
Вим.	Лист	№ докум.	Підпис	Дата			87


```
}
```

В даній роботі було задіяно 8-ми розрядний таймер T0 в режимі NORMAL з перериванням за подією «переповнення». Переддільник таймеру (CSXX) визначається константою SUART_PRESCALER, а таймерна константа TIMER_TCNT_VALUE обчислюється за допомогою макросу.

Основний код програмного UART`а було розміщено в оброблювачу переривання таймеру T0, шляхом додавання в його початок коду перезапису значення лічильного регістру TCNT0. Оптимальним є використання режиму CTC, який у 8-ми разрядних таймерів іноді відсутній.

```
ISR (TIM0_OVF_VECTOR)
```

```
{
```

```
uint8_t start_bit, flag_in;  
uint8_t rx_data;  
static uint16_t internal_rx_buffer;
```

```
TCNT0 += TIMER_TCNT_VALUE;
```

```
... ..
```

```
// основний код
```

```
....
```

```
}
```

Наступним оновленням коду програмного UART`а – є додавання кваліфікатора volatile до змінних, що отримав використання як в перериванні так і в звичайних функціях.

```
static volatile uint8_t inbuf [SUART_BUF_SIZE];
```

```
static volatile uint8_t qin = 0;
```

```
static volatile uint8_t qout = 0;
```

```
static volatile uint8_t flag_rx_ready;
```

```
static uint8_t flag_rx_waiting_for_stop_bit;
```

```
static uint8_t flag_rx_off;
```

```
static uint8_t rx_mask;
```

```
static uint8_t timer_rx_ctr;
```

									Лист
									89
Вим.	Лист	№ докум.	Підпис	Дата					

```

static uint8_t bits_left_in_rx;

static volatile uint8_t flag_tx_busy;
static volatile uint8_t timer_tx_ctr;
static volatile uint8_t bits_left_in_tx;
static volatile uint16_t internal_tx_buffer;

```

Наступним етапом було додавання функції виведення рядків з ОЗУ і з flash пам'яті.

```

void SUART_PutStr (char * str)
{
    char ch;
    while (* str) {
        ch = * str ++;
        SUART_PutChar (ch);
    }
}

```

```

void SUART_PutStrFl (char __flash * str)
{
    char ch;
    ch = read_byte_flash (* str);
    while (ch) {
        SUART_PutChar (ch);
        str ++;
        ch = read_byte_flash (* str);
    }
}

```

Наступним кроком було створення проекту в IAR`е, шляхом підключення до нього отриманої «ліби» програмного UART`а та написання пари рядків коду для перевірки отриманих результатів. Решта змін, що були внесені до коду були

					ЕЛІТ 8.171.00.10.376 ПЗ	Лист
Вим.	Лист	№ докум.	Підпис	Дата		90

пов`язані з невеликою оптимізацією, про них ми говорили.

Апаратне / програмне забезпечення UART працює наступним чином:

- При 9600 бодах, з тактовою частотою 8 МГц, тривалість одного біту складає $8000000/9600$ або 833,3 тактових циклу.
- Спочатку було здійснене відключення USI. Вхід в регістр зсуву USI було підключено до PB0, тому нами було визначено його як вхід та налаштовано переривання зміни пінів на ньому:

```
void InitialiseUSI (void) {  
    pinMode (DataIn, INPUT); // Визначення DI як вхідного  
    USICR = 0; // Відключення USI.  
    GIFR = 1 << PCIF; // Очищення прапору переривання зміни знаку.  
    GIMSK |= 1 << PCIE; // Дозвіл переривання зміни контактів  
    PCMSK |= 1 << PCINT0; // Включення зміни виводів на виведення 0  
}
```

Початок байту викликає переривання зміни знаку. У процедурі обслуговування переривання зміни пін-коду робиться перевірка та визначення падаючого фронту, та налаштування таймеру / лічильнику в режимі СТС. Важливим є встановлення затримки в пів-біті, для потрапляння дов середини стартового біта, який складає 416,7 такту. Найближче, що можна донести - це попередній дільник 8 і порівняння, що дорівнює 52. Робимо очистку та здійснюємо дозвіл переривання порівняння виведення:

```
ISR (PCINT0_vect) {  
    if (! (PINB & 1 << PINB0)) { // Ігнорування, якщо DI високий  
        GIMSK &= ~ (1 << PCIE); // Відключення переривання зміни  
        TCCR0A = 2 << WGM00; // Режим СТС  
  
        TCCR0B = 0 << WGM02 | 2 << CS00; // Встановлення переддільнику на / 8  
        TCNT0 = 0; // Підрахунок від 0  
        OCR0A = 51; // Затримка (51 + 1) * 8 циклів  
        TIFR |= 1 << OCF0A; // Очищення прапору порівняння результатів  
        TIMSK |= 1 << OCIE0A; // Включення переривання порівняння виведення  
    }
```

									Лист
									91
Вим.	Лист	№ докум.	Підпис	Дата					

```
}
```

Переривання збігу порівняння відбувається в середині стартового біта. В процедурі обслуговування переривання порівняння збігів повертається збіг порівняння з тривалістю одного біту, 104, здійснюється дозвіл USI початку зміщення в бітах даних з подальшим зіставленням порівняння та вирішенням переривання переповнення USI:

```
ISR (TIMER0_COMPA_vect) {  
    TIMSK &= ~(1 << OCIE0A); // Відключення переривання COMPA  
    TCNT0 = 0; // Підрахуноквід 0  
    OCR0A = 103; // Зрушення кожно з (103 + 1) * 8 циклів
```

// Включаємо переривання USI OVF та обираємо порівняння часу по Timer0 як джерело синхронізації USI:

```
    USICR = 1 << USIOIE | 0 << USIWM0 | 1 << USICS0;  
    USISR = 1 << USIOIF | 8; // Очищення прапору USI OVF та встановлення  
лічильнику  
}
```

Встановлюємо режим проводки на 0 з $0 << USIWM0$, що гарантує, що вихід зсувного регістру USI не матиме впливувиведення даних, PB1.

Коли 8 біт був зрушений в перериванні, тоді переповнення USI відбувається. Процедура обслуговування переривань відключає USI, зчитує регістр зсуву USI та здійснює дозвіл переривання, зміни переривання для наступного байта:

```
ISR (USI_OVF_vect) {  
    USICR = 0; // Відключення USI  
    int temp = USIDR;  
    Дисплей (ReverseByte (темп));  
    GIFR = 1 << PCIF; // Очищення прапору переривання зміни знака.  
    GIMSK |= 1 << PCIE; // Включення переривання зміни знаку  
}
```

UART відправляє біт LSB, а USI показує, що MSB є першим, є потреба в скасуванні порядку біту після прийому, і це можна отримати з використанням короткої програмної програми ReverseByte ():

```
unsigned char ReverseByte (unsigned char x) {  
  x = ((x >> 1) & 0x55) | ((X << 1) & 0xaa);  
  x = ((x >> 2) & 0x33) | ((X << 2) & 0xcc);  
  x = ((x >> 4) & 0x0f) | ((X << 4) & 0xf0);  
  return x;  
}
```

Дана програма є ефективною, це полягає в перестановці сусідніх окремих бітів, та перестановці суміжних 2-бітових полів, обміні двома 4-бітними полями.

Далі запускаємо функцію Display (), для відображення байту на семисегментному дисплеї з метою налагодження. Код для обробки байту, що буде отримано, має бути в залишковій версії.

Компіляція програми, з використанням розширення ядра ATtiny для Arduino IDE, не включає налаштування для ATtiny85 з кристалом 8 МГц, тому тому було додано в файл boards.txt:

```
attiny45at8x.name = ATtiny45 @ 8 МГц (зовнішній кристалл, BOD відключений)
```

```
attiny45at8x.upload.using = Arduino: arduinoisp
```

```
attiny45at8x.upload.maximum_size = 8192
```

```
# Ext. Кристалічна ось 8,0 МГц; Час запуску: 16К СК / 14 СК + 65 мс; [CKSEL = 1111 SUT = 11]
```

```
# Відміна виявлення коричневого кольору; [BODLEVEL = 111]
```

```
# Зберігання пам'яті EEPROM через цикл очищення чипу; [EESAVE = 0]
```

```
# Послідовна загрузка програми (SPI) вимкнена; [SPIEN = 0]
```

```
attiny45at8x.bootloader.low_fuses = 0xFF
```

```
attiny45at8x.bootloader.high_fuses = 0xD7
```

```
attiny45at8x.bootloader.extended_fuses = 0xFF
```

```
attiny45at8x.bootloader.path = немає
```

					ЕЛІТ 8.171.00.10.376 ПЗ	Лист
Вим.	Лист	№ докум.	Підпис	Дата		93

```
attiny45at8x.bootloader.file = empty85at16.hex
```

```
attiny45at8x.build.mcu = attiny85
```

```
attiny45at8x.build.f_cpu = 8000000L
```

```
attiny45at8x.build.core = малий
```

З додаванням параметру ATtiny45 @ 8MHz (зовнішній кристал, BOD вимкнено) в підменю Board. Необхідно обрати це та Burn Bootloader, для правильного встановлення плавких запобіжників з використанням Tiny AVR завантаження програми до ATtiny45.

Для використання даної процедури з різними частотами кристалів та швидкостями передавання, необхідною буде зміна значення попереднього розподілу та порівняння для таймеру / лічильнику в процедурі переривання зсуву піну та підпрограми переривання USI.

Для високих швидкостей передачі даних доцільним є врахування затримки обробки переривання. Програма ATtiny45 USI UART: проста програма ATtiny45 USI UART.

Було змінено процедуру переривання порівняння вихідних даних, ISR (PCINT0_vect), для відсутності впливу на біти переривання таймера / Counter1. Це дає можливість використання функції таймеру Arduino millis () і delay (), з використанням таймеру / лічильнику1. Метою є очищення прапора порівняння виводу перед включенням переривання порівняння виведення, було додано рядок: TIFR |= 1 << OCF0A в підпрограму PCINT0_vect, а потім затримано половину бітів, в середині старшого біта, змінивши налаштування OCR0A на:

```
OCR0A = 51; // Затримка (51 + 1) * 8 циклів
```

Це гарантує, що кожен приклад зчитується посередині кожного біту.

НАУКОВІ ПРАЦІ СТУДЕНТА

Підвищення ефективності систем передачі даних на основі площинних кодів

Шевченко М.С., *аспірант*; Грахольська А.В., *студент гр. ЕСз-51с*
Сумський державний університет, м. Суми, Україна

Ефективність системи зв'язку визначається значною мірою забезпеченням високої вірності передачі даних. При цьому викликають науково-практичний інтерес такі методи збільшення вірності передачі, які не потребують, по-перше, суттєвих апаратно-програмних витрат при практичній реалізації, по-друге, розширюють функціональні можливості систем передачі щодо адаптації помилковиявляючої та корегувальної здатності застосованих кодів до рівня завад у каналі зв'язку. Досить перспективними з точки зору регулювання помилковиявляючої та корегувальної здатності та підвищення ефективності систем зв'язку при невеликій кількості витрат є площинні коди.

Кодуючі та декодуючі пристрої на основі площинних кодів досить легко можуть перебудовуватися в залежності від кількості помилок, які виникають у кодових послідовностях. Змінюючи число k контрольних розрядів, котрі охоплюють m інформаційних символів, можна вводити у систему передачі даних різні за рівнем помилковиявляючої та корегувальної здатності режими обміну інформацією, наприклад для каналів з низьким або високим рівнем завад. Залежність m інформаційних розрядів від k перевірочних досить проста і виглядає наступним чином $m = 1/2 \cdot k(k - 1)$.

Більш того, пропонується збільшити ефективність застосування площинних кодів, підсилити адаптивність систем передачі на їх основі за рахунок використання сумісно з площинним кодуванням інших завадостійких кодів, наприклад рівноважних, квазірівноважних і біноміальних кодів. Це ще більше підвищить вірність передачі даних, надасть можливість більш точно підлаштовуватися під рівень завад у каналі зв'язку та розширить клас помилок, які виявляються. Крім того, враховуючи те, що у структурі рівноважних, квазірівноважних і біноміальних кодів містяться двійкові біноміальні числа, мається можливість додатково проводити обчислювальну обробку даних з метою їх оцінки, порівняння, сортування тощо.

					ЕЛІТ 8.171.00.10.376 ПЗ	Лист
Вим.	Лист	№ докум.	Підпис	Дата		95

ВИСНОВОК

У процесі виконання кваліфікаційної роботи магістра було розроблено програмно-управляючу систему вимірювання таких електричних параметрів як вимірювач індуктивності та ємності на мікроконтролері, який є оптимальним при врахуванні якості, ціни та розмірів.

В результаті проведеної роботи було отримано оптимальний варіант системи, який відповідає задовільному класу точності та всім заданим вимогам.

За час роботи над кваліфікаційною роботою було вивчено і застосовано на практиці прогресивні методи конструювання, використано довідкову та спеціальну літературу, у тому числі ряд стандартів, інтернет ресурсів.

					ЕЛІТ 8.171.00.10.376 ПЗ	Лист
Вим.	Лист	№ докум.	Підпис	Дата		96

СПИСОК ЛІТЕРАТУРИ

- 1 Мортон Дж. Микроконтроллеры AVR. Вводный курс/ Мортон Дж. – Москва: Додэка-XX1, 2006. - 272 с.
- 2 Степанов А. Простой LC-метр/ Степанов А.// Радио, 1982. – №3. – с.47.
- 3 Семенов Б.Ю. Силовая электроника/ Семенов Б.Ю. – М: СОЛОН-Р, 2001. –321с.
- 4 Усатенко С.Т. и др. Выполнение электрических схем по ЕСКД/ Каченюк Т.К., Терехова М.В. – М: 1989. – 320с.
- 5 Хлюпин Н.П. Измеритель индуктивности и емкости/ Хлюпин Н.П.// Радио, 2004. – №7. – с.25-26.
- 6 Digital Integrated Circuits., 2-е изд. — М.: Вильямс, 2018. — 912 с.
- 7 Godse A. Micriprocessors and Interfacing Techniques./ D. Godse, A. Godse — Pune : Technical Publications, 2019.
- 8 Морелос-Сарагоса Р.. Мистецтво перешкодостійкого кодування / Р. Морелос-Сарагоса — Світ зв'язку, 2018.
- 9 Сендзюк М.А. Інформаційні системи і технології в економіці: навч.-метод. посіб. для самост. вивч. дисципліни / М.А. Сендзюк – М-во освіти і науки України, ДВНЗ “Київ. нац. екон. ун-т ім. В. Гетьмана”. – К.: КНЕУ, 2013. – 68 с.
- 10 Пономаренко В. С. Інформаційні системи в економіці: навч. посібник / В. С. Пономаренко, І. О. Золотарьова, Р. К. Бутова – Вид. ХНЕУ, 2016. – 176 с.
- 11 Colbourn Charles J. Handbook of Combinatorial Designs/ Н. Dinitz Jeffrey, J. Colbourn Charles — 2nd. — Boca Raton: Chapman & Hall/ CRC, 2017.
- 12 Jain, D., Tech, M., Engineering, C., Engineering, C., 2017. A Modified Method for Sound Compression Using Intrinsic Mode Functions 97–102.
- 13 Kalajdzic, K., Ali, S.H., Patel, A., 2015. Rapid lossless compression of short textmessages. Comput. Stand. Interfaces 37, 53–59. Kavitha, P., 2019. A Survey on Lossless and Lossy Data Compression Methods 7,110–114.
- 14 Н. В. Kekre, Tanuja Sarode and Prachi Natu, Performance Comparison of Column Hybrid Row Hybrid and full Hybrid Wavelet Transform on Image compression using Kekre Transform as Base Transform, International Journal of Computer Science and Information Security, (IJCSIS), vol. 12(2), pp. 5–17, (2016).
- 15 М. А. Kabir, А. М. Khan, М. Т. Islam and М. L. Hossain, Image Compression using Lifting Based Wavelet Transform Coupled with SPIHT Algorithm, In

					ЕЛІТ 8.171.00.10.376 ПЗ	Лист
Вим.	Лист	№ докум.	Підпис	Дата		97

Proceedings of IEEE International Conference on Informatics, Electronics & Vision (ICIEV), pp. 1–4, (2018).

16 Maksimova T., Shapran E., 2019.: Approach to diagnostics of marketing complex of industrial enterprise / TEKA Kom. Mot. I Energ. Roin. – OL PAN, 2010, 10 B, V XB. – Lublin, 2010. – P. 5-11.

17 Rad, R.M., Attar, A., Shahbahrami, A., 2019. A predictive algorithm for multimedidata compression. *Multimed. Syst.* 19, 103–115.

					ЕЛІТ 8.171.00.10.376 ПЗ	Лист
Вим.	Лист	№ докум.	Підпис	Дата		98

ДОДАТОК А

Подпрограмма обмена с МТ-10Т7-7Т :

sub_0_40:

;П/Прог вывода на ЖКИ МТ-10Т7, бит 5 всегда 1 для WR1
;бит 4=0-запись адреса, бит 4=1-запись данных
;в R22-готовое число для вывода

```
cbi PORTB, 3
cbi PORTB, 4
ldi r22, 0x2f ;адрес разблокировки индикатора 0xf
rcall shift
ldi r22, 0x3f ;код разблокировки индикатора 0xf
rcall shift
ldi r22, 0x20 ;адрес 1-го знакоместа 0x0
rcall shift
```

```
ldi r16, 0xA ;10 символов
ldi r30, 6 ;символы в регистрах R6..R15
```

loc_0_42:

```
ld r17, Z ;грузим через Z косвенно
mov r20, r30 ;сохраним указатель Z
mov r21, r31
ldi ZL, low (FONT*2) ;начало таблицы знакогенератора
ldi ZH, high (FONT*2)
add ZL, r17 ;добавить смещение (Z + код цифры)
brcc M1
inc ZH ;увы-портиМ r0-если что надо сохранить!
```

M1:

```
lpm ;чтение таблицы знакогенератора в r0
mov r30, r20 ;восстановим Z
mov r31, r21
mov r17, r0 ;размножим код знакогенератора из FONT
mov r22, r0
swap r17 ;старшую тетраду готовим для сдвига- перемещ в
```

младш

```
andi r17, 0x0f
andi r22, 0x0f
ori r17, 0x30
ori r22, 0x30
rcall shift
mov r22, r17
rcall shift ;обе тетрады передали, разберемся со счетчиками
inc r30 ;указатель Z на след-й символ
dec r16
brne loc_0_42
ret
```

; End of function sub_0_40

```
shift: ldi r23, 0x07 ;7 стробов до сброса
lsl r22
lsl r22 ;выдвинули WR1 бит в 7-й разряд
```

M2:

```
lsl r22
brcs M3 ;переход, если флаг C уст
cbi PORTB, 4
rjmp M4
```

M3:

```
sbi PORTB, 4
nop
```

M4:

```
sbi PORTB, 3 ;Строб
nop ;для
cbi PORTB, 3 ;74hc164
dec r23
brne M2
ret
```

									Лист
									99
Вим.	Лист	№ докум.	Підпис	Дата					

