КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

ІМЕНІ ТАРАСА ШЕВЧЕНКА

ФАКУЛЬТЕТ РАДІОФІЗИКИ, ЕЛЕКТРОНІКИ ТА КОМП'ЮТЕРНИХ СИСТЕМ

Кафедра радіотехніки та радіоелектронних систем

До захисту допущено: «На правах рукопису»
Завідувач кафедри _____ Ігор АНІСІМОВ
« __ » червня 2023 р.

КВАЛІФІКАЦІЙНА РОБОТА БАКАЛАВРА

на тему:

«ЦИФРОВА ПАЯЛЬНА СТАНЦІЯ З АКУМУЛЯТОРНИМ ЖИВЛЕННЯМ»

Виконав:
студентка 4-го курсу
денної форми навчання
спеціальності 172 - Телекомунікації та радіотехніка
ОП «Інформаційна безпека телекомунікаційних систем і мереж»
Баумґартен Ґрета Ігорівна _____

Науковий керівник:
к.ф.-м.н., ас. Фесенко Сергій Олександрович _____

Рецензент:
д.ф.-м.н., проф. Веклич Анатолій Миколайович _____

Засвідчую, що у цій бакалаврській роботі
немає запозичень з праць інших авторів без
відповідних посилань
Студент _____

Робота допущена до захисту в ЕК рішенням кафедри радіотехніки та радіоелектронних систем від «22» червня 2023 р., протокол № 21.

Завідувач кафедри радіотехніки та радіоелектронних систем,
доктор фіз.-мат. наук, професор
Анісімов Ігор Олексійович _____

Київ – 2023

ЗМІСТ

ВСТУП ..…… 3
РОЗДІЛ 1. ОГЛЯД АПАРАТНОЇ ЧАСТИНИ …… 4
1.1. Принцип роботи струмових ШІМ-контролерів …….4
1.2. Опис мікроконтролеру Atmega88-16PU...…….9
РОЗДІЛ 2. ПРИНЦИП РОБОТИ, ПЕРЕВАГИ Й НЕДОЛІКИ ІНШИХ ТЕХНІЧНИХ РІШЕНЬ ЦИФРОВИХ ПАЯЛЬНИХ СТАНЦІЙ……....14
2.1. Огляд наявних технічних рішень ……14
2.2. Аналіз схемотехнічних рішень паяльної станції SL-10……..16
РОЗДІЛ 3. СИНТЕЗ ЕЛЕКТРИЧНОЇ ПРИНЦИПОВОЇ СХЕМИ ЦИФРОВОЇ ПАЯЛЬНОЇ СТАНЦІЇ З АКУМУЛЯТОРНИМ ЖИВЛЕННЯМ. ……..20
3.1. Синтезована електрична схема ……...……..…20
ВИСНОВКИ ……. 26
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ ……...…… 27
ДОДАТОК. А. ПРОГРАМНИЙ КОД МІКРОКОНТРОЛЕРА…… 28

ВСТУП

Життя сучасного суспільства практично неможливе без радіоелектронних засобів. Радіомовлення (broadcasting), телебачення (television), телефон (telephone), телеграф (telegraph), телетайп, радіолокація, промислова, медична та аграрна електроніка – далеко не повний перелік електронних засобів, які широко використовуються у нашому житті. Але будь-яка система забезпечується сукупністю електричних і електронних пристроїв різної складності, які складаються з елементів, до яких прикладені електричні напруги або протікають електричні струми. Будь-які складні радіоелектронні пристрої складаються з різноманітних електричних кіл, які мають певні електричні властивості. Кожен такий пристрій будується шляхом з'єднання пасивних та активних компонентів у єдину систему.

Оскільки основним методом з'єднання електричних провідників є пайка, то існує ціла низка паяльного обладнання, яке використовується для різних задач. Звичайно, для пайки провідників і радіокомпонентів можна використовувати звичайний паяльник, але він має певні недоліки, зокрема - неможливість контролювати температуру. Для усунення цього недоліку і були розроблені паяльники з точним підтриманням температури. Такий паяльник з блоком керування можна назвати паяльною станцією. Звичайно, найпростіше придбати готовий пристрій, але як правило, якісна паяльна станція коштує досить дорого. Окрім того, саморобний пристрій можна налаштувати під свої потреби, що не завжди можна зробити із промисловими зразками. Також, як правило, промислові зразки регулюють температуру в ключовому режимі - що може зменшити час життя нагрівального елемента. Також, не варто принижувати отриманий досвід розробки радіоелектронного пристрою, який отримує студент під час виконання поставленого завдання. З огляду на вищесказане:

Темою дипломного проекту є: цифрова портативна паяльна станція з можливістю акумуляторного живлення, плавною зміною напруги

нагрівального елемента (для збільшення часу його роботи) та наявністю додаткового режиму швидкої зміни температури на задану величину (дуже зручна функція для паяння полігонів і масивних провідників, яка, як правило, відсутня у промислових зразках).

РОЗДІЛ 1. ОГЛЯД АПАРАТНОЇ ЧАСТИНИ

1.1. Принцип роботи струмових ШІМ-контролерів

ШІМ-контролер (англ. Pulse Width Modulation controller, PWM) - це електронний пристрій, який генерує сигнал зі змінною шириною імпульсів з високою точністю і частотою. Цей сигнал використовується для керування потужністю електронних пристроїв, таких як двигуни, світлодіоди, нагрівальні елементи тощо.

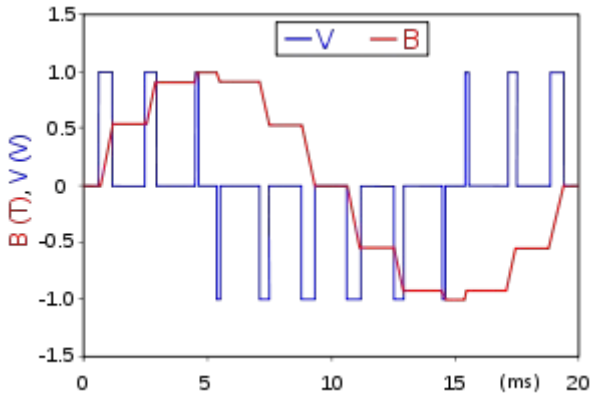

Рис.1.1 Принцип широтно-імпульсної модуляції

ШІМ-контролери широко застосовуються в різних галузях, де потрібно керування потужністю електричних пристроїв з високою точністю і ефективністю. Наприклад, вони використовуються в автомобільній промисловості для керування електродвигунами, в електроніці для керування світлодіодами, в електричних нагрівальних системах для керування температурою та інших пристроях.

Одним з основних переваг ШІМ-контролерів є їх ефективність. Завдяки змінній ширині імпульсів, контролер може керувати потужністю електричного пристрою з дуже великим значенням ККД. Це дозволяє зменшити витрати на електроенергію та покращити надійність пристрою.

ШІМ-контролери можуть бути програмовані з використанням мікроконтролерів або FPGA, що дозволяє налаштувати їх для конкретних завдань та вимог. Окрім того, на ринку існує безліч ШІМ-контролерів, що мають різні параметри та функції, що дозволяє вибрати оптимальний контролер для конкретного застосування [1].

Основні типи ШІМ-контролерів:

1.Аналогові ШІМ-контролери: Генерують аналоговий сигнал зі змінною шириною імпульсів. Вони можуть бути реалізовані з використанням аналогових компонентів, таких як операційні підсилювачі, резистори, конденсатори тощо.

2. Цифрові ШІМ-контролери: Генерують цифровий сигнал зі змінною шириною імпульсів. Вони можуть бути програмовані з використанням мікроконтролерів або FPGA, що дозволяє налаштувати їх для конкретних завдань та вимог.

3. Єдиний ШІМ-контролер: Контролер, що забезпечує керування потужністю всіх пристроїв в системі за допомогою єдиного ШІМ-сигналу.

4. Мультиканальні ШІМ-контролери: Контролер, що забезпечує керування потужністю декількох пристроїв в системі за допомогою різних ШІМ-сигналів.

5. Інтегровані ШІМ-контролери: Контролери, що вбудовуються безпосередньо в інші електронні пристрої, такі як мікроконтролери, FPGA, драйвери моторів, і т.д.

Крім цього, можуть існувати ШІМ-контролери з різною кількістю каналів керування, різним діапазоном частот та іншими параметрами, що залежать від конкретних вимог до системи керування.

До окремого підтипу ШІМ - контролерів варто віднести струмові. Так називають контролери, які постійно слідкують за струмом силового транзистора і якщо він перевищує заданий ліміт - вимикають транзистор до закінчення періоду внутрішнього тактового генератора. Відкривається транзистор з початком наступного періоду генератора. Отже, одним із обов'язкових елементів такого типу ШІМ контролерів є тригер-защіпка, який блокує силовий транзистор до закінчення періоду. На початку наступного періоду формується короткий імпульс, який перекидає тригер у початковий стан - відриваючи силовий транзистор. Яскравим і дуже популярним представником струмових ШІМ контролерів є: UC3843. Ця ІМС дуже часто використовується для керування різноманітними типами вимикачів, наприклад, польовими транзисторами або IGBT, зокрема, вона використовується і в даній роботі. Ця схема забезпечує стабільний та точний сигнал широтно-імпульсної модуляції (ШІМ), що дозволяє регулювати величину вихідного сигналу від силового елемента.

Основні технічні характеристики UC3843 включають:

1. Напруга живлення (VCC): у нормі, цей чіп працює з напругою живлення в діапазоні від 7 до 30 В.
2. Струм споживання: споживаний струм UC3843 залежить від режиму роботи і підключених навантажень. Він зазвичай становить кілька міліампер.
3. Вихідні напруги: UC3843 має два вихідних піна - COMP (компенсація) і OUT (вихід). Вихідний сигнал компенсації зазвичай знаходиться в діапазоні 0-3 В, а вихідний сигнал визначається зовнішніми компонентами, підключеними до виводу OUT.
4. Частота ШІМ: UC3843 має внутрішній осцилятор, який генерує ШІМ-сигнал. Частота цього осцилятора може бути налаштована зовнішніми компонентами і зазвичай становить кілька десятків кілогерц.
5. Функції захисту: UC3843 може мати різні функції захисту, такі як захист від короткого замикання, захист від перевантаження та захист

від перегріву[2]. Ці функції захисту допомагають забезпечити безпеку і надійність схеми.

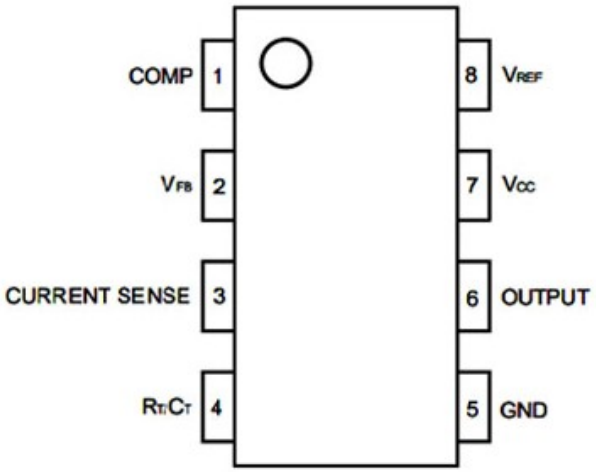

Рис.1.2. Виводи IMC UC3843.

UC3843 має широкі можливості настройки параметрів, таких як частота перемикання, величина пульсу, час спадання та час наростання сигналу. Крім того, вона має вбудовані захисні функції, такі як захист від перевантаження, замикання та перегріву [3].

Контролер може бути використаний в різних додатках, таких як силові джерела живлення, перетворювачі постійного струму (DC-DC), інвертори, схеми заряду батарей та інші.

Ця схема має просту конструкцію, що дозволяє легко підключати її до інших елементів електронної схеми.

Вона використовується для керування імпульсними блоками живлення, які забезпечують стабільну вихідну напругу та струм для електронних пристроїв, має вбудований операційний підсилювач, який дозволяє регулювати частоту та ширину імпульсів ШІМ, що надходять на ключовий транзистор імпульсного блоку живлення. Ця інтегральна схема має широкий діапазон напруг живлення від 8 В до 35 В.

Для зміни скважності імпульсів, які виводяться з цієї схеми, доречно використовувати різні методи, описані нижче:

Один з методів полягає у зміні опору відповідного резистора на вході COMP (компенсаційного входу) схеми UC3843.

Змінюючи значення цього опору, можна налаштувати скважність імпульсів. Більший опір призводить до зменшення скважності, а менший опір - до збільшення скважності.

Інший метод полягає у використанні зовнішнього напругового поділювача для зміни відношення між опором компенсації і опором регулювання на вході COMP. Змінюючи це відношення, можна вплинути на скважність імпульсів.

Необхідно враховувати, що зміна скважності імпульсів може вимагати налагодження інших параметрів схеми, таких як частота ШІМ і рівень компенсації, для досягнення оптимального результату. Деталі налаштування залежать від конкретної застосовної схеми та вимог проекту.

UC3843 має 8 виводів для підключення до інших елементів електронної схеми. Ось деякі з них та їх функції:

VCC: Вхідне живлення (позитивна напруга). Зазвичай підключається до джерела живлення напругою від 8 до 35 Вольт.

GND: Земля або негативна напруга.

COMP: Вихід підсилювача помилки. Використовується для порівняння вихідної напруги з референсною напругою та формування сигналу ПШМ (пульсово-широтної модуляції).

FB: Зворотний зв'язок (Feedback). Підключається до точки на вихідному фільтрі, щоб забезпечити стабілізацію вихідної напруги або струму.

RT / CT: Зовнішє RC - коло, що встановлює частоту роботи контролера.

DIS: Вивід для вимкнення контролера (Shutdown). Підключення до землі призводить до вимкнення контролера.

OUT: Вихідний ШІМ сигнал контролера. Слід зауважити, що вихідний драйвер може витримувати струми до 1А, що, як правило, достатньо для швидкої перезарядки затворів навіть потужних силових транзисторів.

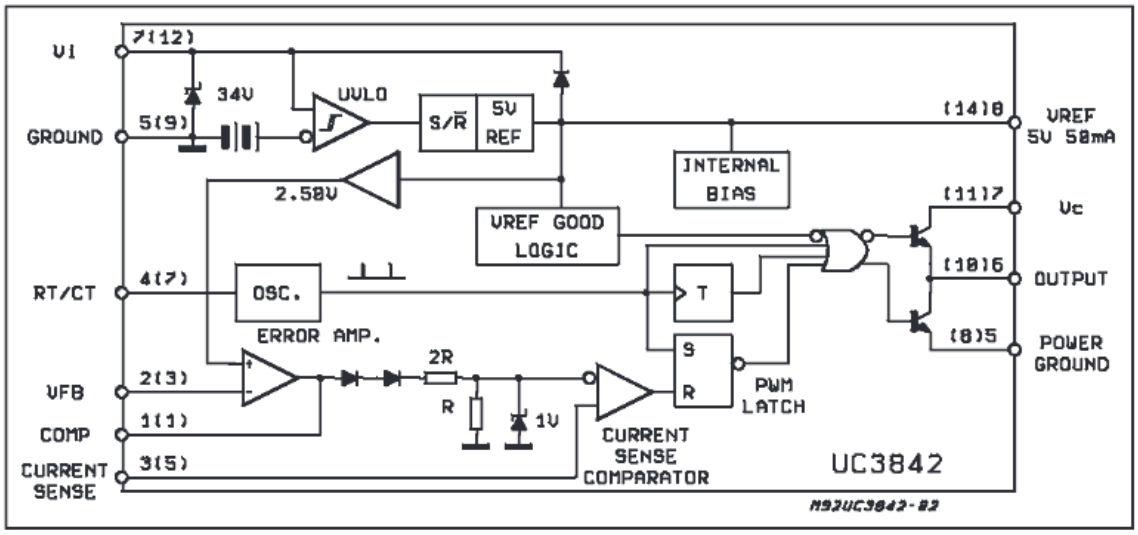

Рис. 1.2. Внутрішня блок-схема ШІМ-контролера UC3843

З рис. 1.1. видно, що блок-схема контролера UC3843 включає в себе наступні основні блоки:

Компаратор напруги: зрівнює опорну напругу VREF з напругою зворотного зв'язку від виходу джерела живлення.

Вхідна логіка: забезпечує інтерфейс контролера із зовнішньою середою, включаючи управління вихідним сигналом, вхідним напругою та вхідними струмами.

Генератор імпульсів: генерує широкий імпульсний сигнал на основі зрівняння опорної напруги і напруги зворотного зв'язку, який керує вихідним транзистором джерела живлення.

Широтно-імпульсна модуляція (ШИМ): керує шириною імпульсів на виході контролера, що дозволяє регулювати вихідну напругу джерела живлення.

Управління генератором імпульсів: керує параметрами генератора імпульсів, включаючи частоту, рівень сигналу та інші параметри.

Вихідний транзистор: керує вихідною напругою джерела живлення шляхом включення і виключення із заданою шириною імпульсів.

Осцилятор: генерує внутрішній опорний сигнал для контролю частоти ШІМ.

Джерело опорної напруги: генерує стабільну опорну напругу VREF для використання в поєднанні з напругою зворотного зв'язку.

У цілому UC3843 являє собою високопродуктивний ШІМ контролер, який дозволяє регулювати вихідну напругу джерела живлення з високою точністю та ефективністю.

ШІМ-контролери можна класифікувати за різними ознаками, наприклад, за типом сигналу, який вони генерують, або за кількістю каналів керування, які вони підтримують.

1.2 Опис мікроконтролера Atmega88-16PU

ATmega88-16PU є мікроконтролером, що виробляється компанією Atmel (зараз Microchip Technology). Він належить до серії AVR мікроконтролерів і має 8-бітову архітектуру. ATmega88-16PU має вбудовану флеш-пам'ять ємністю 16 кілобайт, оперативну пам'ять (SRAM) об'ємом 1 кілобайт та набір периферійних пристроїв, таких як таймери, АЦП, порти вводу-виводу та інші. Цей мікроконтролер працює з напругою живлення в діапазоні від 2,7 до 5,5 вольт і має 32 контакти в корпусі DIP. ATmega88-16PU широко використовується в різних вбудованих пристроях, таких як автомобільні електронні системи, системи керування побутовою технікою, мікроконтролерні пристрої для промисловості та інші.

Мікроконтроллер має 8-бітний RISC-процесор з тактовою частотою до 16 МГц, 8 Кб внутрішньої флеш-пам'яті для програм, 1 Кб SRAM і 512 байт EEPROM.

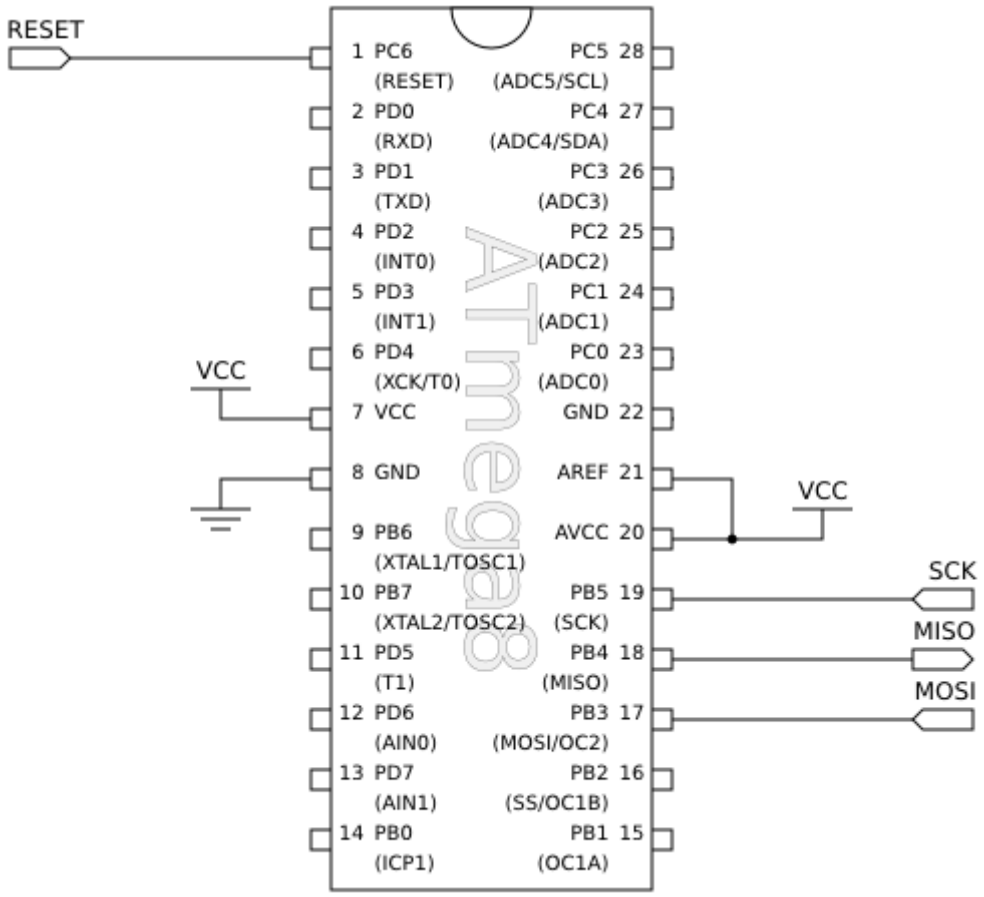

Рис.1.3 Назви виводів мікроконтролера ATmega88-16PU.

Крім того, мікроконтролер ATmega88-16PU має вбудовані апаратні таймери / лічильники, порти введення / виведення (GPIO), а також аналого-цифровий перетворювач (ADC) з аналоговим мультиплексором.

ATmega88-16PU має 28 виводів для з'єднання з іншими елементами електронної схеми. Ось опис розташування та функціонального призначення кожного виводу:

1. PCINT14 / T0 / PD0 - вхід / вихід для переривання програмного забезпечення 14, вхід таймера 0 / порту D;

2. PCINT15 / T1 / PD1 - вхід / вихід для переривання програмного забезпечення 15, вхід таймера 1 / порту D;
3. PCINT16 / INT0 / PD2 - вхід / вихід для переривання програмного забезпечення 16, вхід переривання 0 / порту D;
4. PCINT17 / INT1 / PD3 - вхід / вихід для переривання програмного забезпечення 17, вхід переривання 1 / порту D;
5. PCINT18 / OC2B / PD4 - вхід / вихід для переривання програмного забезпечення 18, вихід каналу B таймера 2 / порту D;
6. PCINT19 / OC2A / PD5 - вхід / вихід для переривання програмного забезпечення 19, вихід каналу A таймера 2 / порту D;
7. PCINT20 / ICP1 / PD6 - вхід / вихід для переривання програмного забезпечення 20, вхід збору збоїв таймера 1 / порту D;
8. PCINT21 / XTAL1 / PD7 - вхід / вихід для переривання програмного забезпечення 21, вхід кварцового генератора;
9. PCINT22 / XTAL2 / PB0 - вхід / вихід для переривання програмного забезпечення 22, вихід кварцового генератора;
10. PCINT23 / TOSC1 / PB1 - вхід / вихід для переривання програмного забезпечення 23, вхід зовнішнього резонатора;
11. PCINT0 / CLK0 / PB2 - вхід / вихід для переривання програмного забезпечення 0, вхід годинника / порту B;

Пристрій можна підключити до інших елементів електронної схеми, використовуючи різноманітні інтерфейси та протоколи зв'язку. Зауважимо деякі особливості ATmega88-16PU:

1. Для правильної роботи ATmega88-16PU потрібно підключити живлення ві діапазоні від 2,7 В до 5,5 В. Для забезпечення стабільної роботи рекомендується використовувати стабілізоване джерело живлення. Слід зауважити, що мікроконтролер має вбудовану систему контролю напруги живлення (BOD), яка вимкнута за замовчеванням.

2. Для підключення до кварцового резонатора ATmega88-16PU має спеціальні входи. Для правильної роботи мікроконтролера потрібно підключити кварц з частотою до 16 МГц до виводів XTAL1 та XTAL2.

3. Підключення до периферійних пристроїв: ATmega88-16PU має багато вихідних та вхідних портів, які можна використовувати для підключення до різних периферійних пристроїв, таких як дисплеї, сенсорні екрани, датчики, актуатори тощо.

4. Підключення до ПК: для програмування та отримання даних з ATmega88-16PU можна використовувати спеціальний програматор та програмне забезпечення, яке дозволяє писати та завантажувати програми на мікроконтролер. Для підключення до ПК можна використовувати USB-порт або інтерфейси, такі як RS232, SPI, I^2C тощо.

5. За допомогою UART: Мікроконтролер ATmega88-16PU має вбудований UART і може підключатися до інших пристроїв, що підтримують цей протокол зв'язку. Для підключення до UART потрібно використовувати виводи PD0 (RX) та PD1 (TX).

6. За допомогою I^2C: ATmega88-16PU підтримує протокол зв'язку I^2C, який дозволяє підключати до нього різні пристрої, такі як датчики, дисплеї та інші. Для підключення до I^2C потрібно використовувати виводи PC4 (SDA) та PC5 (SCL).

7. За допомогою SPI: ATmega88-16PU підтримує протокол зв'язку SPI, який використовується для підключення до нього різних пристроїв, таких як дисплеї, сенсори та інші. Для підключення до SPI потрібно використовувати ноги PB5 (MOSI), PB6 (MISO) та PB7 (SCK).

8. За допомогою портів введення / виведення: ATmega88-16PU має 3 порти введення / виведення (PORTB, PORTC та PORTD), які можна використовувати для підключення до нього різних елементів схеми, таких як світлодіоди, датчики та інші. Для підключення до портів введення / виведення потрібно використовувати відповідні виводи, які відповідають за кожен порт [5].

РОЗДІЛ 2. ПРИНЦИП РОБОТИ, ПЕРЕВАГИ Й НЕДОЛІКИ ІНШИХ ТЕХНІЧНИХ РІШЕНЬ ЦИФРОВИХ ПАЯЛЬНИХ СТАНЦІЙ

Паяльна станція – це пристрій, що використовується для паяння компонентів на електронних платах. Вона складається з двох основних компонентів: паяльника і блока живлення.

Паяльник – це ручний інструмент з металевим наконечником (жалом), який використовується для нагрівання місця з'єднання компонентів і паяльного олов'яно-свинцевого (або безсвинцевого) припою. Важливі характеристики паяльника включають потужність, температурний діапазон і можливість заміни жала.

Блок живлення – це джерело живлення, яке забезпечує електричну енергію для паяльника. Воно може мати регульовану потужність і можливість налаштування температури. Деякі моделі паяльних станцій можуть мати цифрові дисплеї, які відображають поточну температуру паяльника.

Додаткові функції та характеристики паяльних станцій можуть включати автоматичне вимкнення для збереження енергії та захисту від перегріву, регульовану повітряну подачу для використання з гарячим повітрям, антистатичний захист для роботи з чутливими електронними компонентами тощо.

2.1 Огляд наявних технічних рішень

Для прикладу розглянемо один з типових паяльників, який використовується у паяльних станціях - TCP 12 (рис. 2.1).

Рисунок 2.1 — Зовнішній вигляд паяльника TCP 12

Характеристики паяльника TCP 12:

1. Напруга живлення: 12 В постійного струму
2. Максимальна потужність: 35 Вт
3. Триметровий подовжувач із затискачами (крокодили)
4. Контроль: 2-х точковий контролер з датчиком температури (використовується ефект Кюрі)
5. Температурні режими: 5 = 260°C 6 = 310°C 7 = 370°C 8 = 425°C
6. Тип жала: з жалом LT M та перехідником для тис PT 7 (370°C)

Паяльник має вихідну потужність 30/40 Вт, 12/14 В. Паяльник поставляється з клемою акумулятора, щоб його можна було підключити, наприклад, до автомобільного акумулятора.

Іншим типовим прикладом паяльника є SanErYiGo SH72 (рис. 2.2).

Рисунок 2.2 Зовнішній вигляд паяльника SanErYiGo SH72.

Характеристики паяльника SanErYiGo SH72:

1. Робоча напруга: 12-24В;
2. Потужність: 18-65Вт;
3. Діапазон температур: 220-400 °С;
4. Сумісність жала: SH-BC2/SH-C4/SH-D24/SH-I/SH-KU;
5. Роз'єм живлення: 5.5х2.5мм;
6. Габаритні розміри ручки: 110х16х15мм;
7. Розміри жала: 115х5.5мм;
8. Вага: 28г.

Паяльник SanErYiGo SH72 живиться від 12В до 24В постійної напруги. Для підключення живлення паяльника використовується роз'єм 2.5х5.5мм, для підключення блока живлення або зовнішнього акумулятора LiFePo4 за допомогою перехідника.

Максимальна потужність 65Вт дозволяє без проблем паяти як малі радіокомпоненти так і проводи і контакти великих діаметрів (наприклад провід 10AWG на роз'єм XT90), а можливість живлення від зовнішнього акумулятора дозволяє робити це навіть у польових умовах. Потужність паяльника та споживання струму залежать від вхідної напруги живлення:

- Напруга живлення: 12В, Потужність: **18Вт**, Струм: >1.5А;
- Напруга живлення: 16В, Потужність: **32Вт**, Струм: >1.9А;
- Напруга живлення: 19В, Потужність: **40Вт**, Струм: >2.1А;
- Напруга живлення: 24В, Потужність: **65Вт**, Струм: >2.7А [7].

2.2 Аналіз схемотехнічних рішень паяльної станції SL-10

У даному підрозділі зроблено аналіз паяльної станції SOLOMON SL-10 - виробника високоякісного професійного паяльного обладнання [4].

Паяльна станція SL-10 має наступні характеристики та функції:

1. Регульована температура: Паяльна станція SL-10 зазвичай має можливість налаштування температури паяльника для відповідності вимогам конкретного завдання паяння.

2. Швидке нагрівання: Модель SL-10 зазвичай має високоефективний нагрівальний елемент, що дозволяє швидко досягти бажаної робочої температури.

3. Компактний дизайн: Паяльна станція зазвичай має компактний і зручний дизайн, що дозволяє зручне використання та зберігання.

4. Додаткові функції: Деякі моделі SL-10 можуть мати додаткові функції, такі як автоматичне вимкнення після певного періоду бездіяльності, екран для відображення налаштувань та інші.

5. Паяльні станції вільні від традиційних недоліків звичайних електропаяльників і мають, по-перше, швидке нагрівання до робочого стану, а, по-друге, мають температурний контроль за нагріванням і вбудовану систему підтримки оптимального теплового режиму з ефективними засобами стабілізації робочої температури під час паяння. Наявність широкого набору змінних жал та допоміжних пристроїв для паяння дозволяє значно розширити межі традиційного процесу паяння.

Широке коло фірм-виробників (від європейських та американських – до тайванських) надає сьогодні великі можливості вибору паяльних станцій, що технічно підходять для виконання будь-яких радіомонтажних робіт. Ціни на такі станції (у доларовому обчисленні) коливаються у досить широких межах.

Паяльні станції SL-10, SL-20, SL-30 відрізняються типами індикаторних пристроїв, що відображають температуру паяльника на даний момент часу. На рис. 2.4 можемо розглянути зовнішній вигляд SL-10.

Рис.2.4 Зовнішній вигляд паяльної станції SL-10

У станції SL-10 індикатор відсутній: температура жала паяльника визначається за допомогою движкового потенціометра за лінійною шкалою. Індикація нагріву паяльника здійснюється світлодіодом, який конструктивно

розташований в ручці потенціометра. У станції SL-20 температура жала також задається потенціометром, але у якості індикатора використовується лінійна світлодіодна шкала.

Принципова схема паяльної станції SL-10 наведено на рис. 2.5. Напруга з термопари ТП, розташованої в нагрівальному елементі паяльника, подається на вхід, що не інвертує, операційного підсилювача, виконаного на одній половині мікросхеми DA1 LM358. Коефіцієнт посилення каскаду визначається ланцюгом НЗЗ (резистори R3-R5) і регулюється підстроювальним резистором R3. Вихідна напруга DA1.1, пропорційна температурі жала паяльника, подається на вхід DA1.2. На інвертуючий вхід DA1.2 одночасно подається напруга з RV1 потенціометра, що виконує роль задатчика температури. DA1.2, виконаний на другій половині МС LM358, у цій схемі виконує функції компаратора напруги.

Вихідна напруга DA1.2 управляє схемою комутації транзисторах TR2, TR3. Стабілізація напруги задатчика здійснюється за допомогою параметричного стабілізатора, виконаного на транзисторі TR1.

Керуючим елементом, що комутує струм нагрівального елемента паяльника, є симистор TR4 типу BT136-600E. Після включення паяльної станції температура паяльника невелика, напруга на виході DA1.1 близько до нуля; при цьому напруга на виході DA1.2 відсутня. Транзистор TR2 закритий, транзистор TR3 відкритий, що призводить до відкривання симістора TR4 та нагрівання паяльника.

Для прикладу розглянемо принципову схему паяльної станції на рис. 2.5.

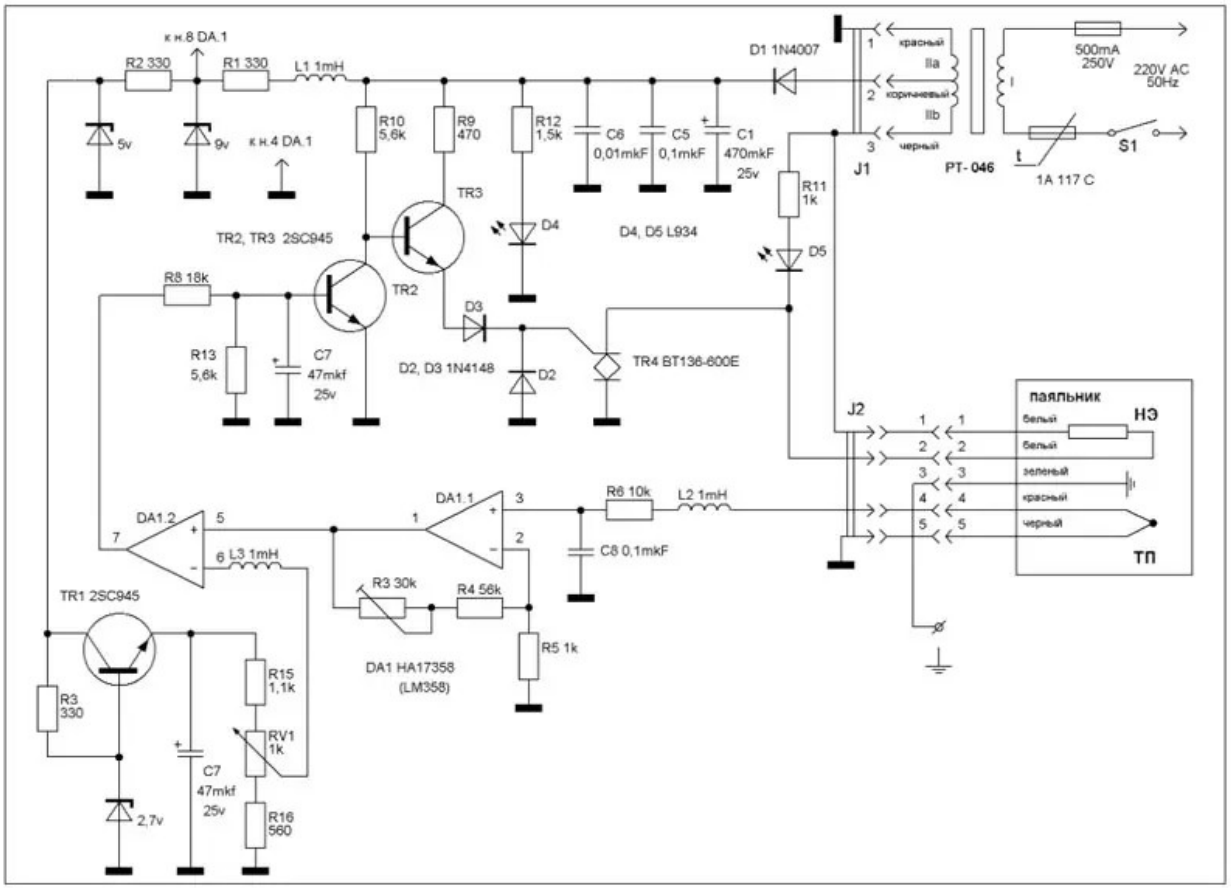

Рис. 2.5. SOLOMON-10. Принципова схема паяльної станції

У міру нагріву жала паяльника напруга на виході DA1.1 збільшується до тих пір, поки не перевищить напругу, що задає, на інвертуючому вході DA1.2. З'являється при цьому на виході DA1.2 позитивна напруга відкриває транзистор TR2; транзистор TR3 закривається – нагрівання паяльника припиняється. Далі схема працює в режимі автоматичного підтримання заданої температури.

Для живлення паяльної станції застосовується мережевий трансформатор типу PT-045, призначений як станції SL-10, так схожої за електронною схемою SL-20. Застосування понижуючого трансформатора забезпечує гальванічну розв'язку між електричною мережею 220 і низьковольтним паяльником, що забезпечує необхідну електробезпеку при роботі зі станцією і дозволяє включити в схему елементи заземлення жала паяльника. Потужність трансформатора - 60 Вт, вторинна обмотка забезпечує напругу 2 х 12В при максимальному струмі 2 А.

Живлення паяльника станції здійснюється напругою 24 В; для живлення електронної схеми напруга 12 знімається з однієї з обмоток трансформатора і випрямляється діодом D1.

Конструктивно паяльна станція SL-10 виконана в корпусі із твердої пластмаси. Органи керування та індикації розташовані на передній панелі станції. Паяльник підключається до станції за допомогою 5-штиркового роз'єму, розташованого на лівій бічній панелі. Усі елементи принципової схеми змонтовано на друкованій платі, там же встановлений змінний повзунковий резистор RV1, движок якого виведений на передню панель станції.

Керуючий симистор TR4 встановлений на компактному пластинчастому радіаторі HS314. Друкована плата з'єднується із трансформатором живлення за допомогою роз'єму J1; вихідний роз'єм з'єднаний з роз'ємом J2 друкованої плати шлейфом.

Істотним недоліком паяльної станції SL-10 є відсутність індикатора температури жала паяльника. Світлодіод D5, що індикує подачу напруги на нагрівальний елемент, не дає уявлення про температуру паяльника. Тим часом інформацію про температуру тиску легко отримати, вимірюючи вихідну напругу на виході підсилювача термопари [8].

РОЗДІЛ 3. СИНТЕЗ ЕЛЕКТРИЧНОЇ ПРИНЦИПОВОЇ СХЕМИ ЦИФРОВОЇ ПАЯЛЬНОЇ СТАНЦІЇ З АКУМУЛЯТОРНИМ ЖИВЛЕННЯМ.

3.1. Синтезована електрична схема

На рисунку 3.1 показана принципова електрична схема цифрової паяльної станції з мікроконтролерним керуванням:

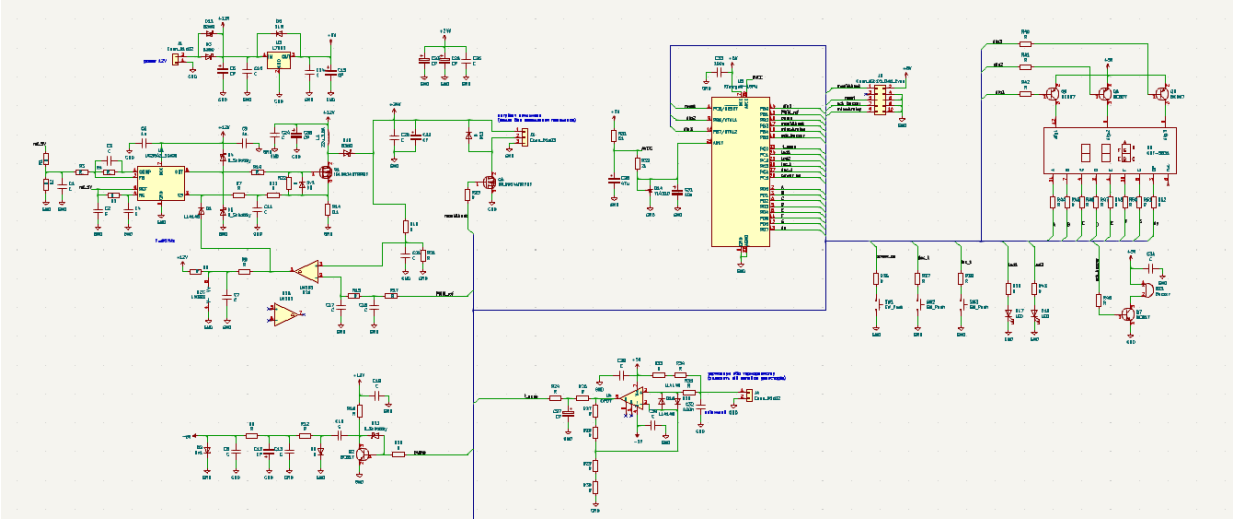

Рис. 3.1. Схема принципова електрична схема цифрової паяльної станції з мікроконтролерним керуванням.

Її основою є восьмирозрядний мікроконтролер ATMEGA8-16PU. Розглянемо роботу різних блоків схеми.

На рис. 3.2. зображено блок ШІМ-контролера з DC-DC перетворювачем, його обв’язку з опорним компаратором та силовим ключем на польовому транзисторі IRLR024NTRPBF.

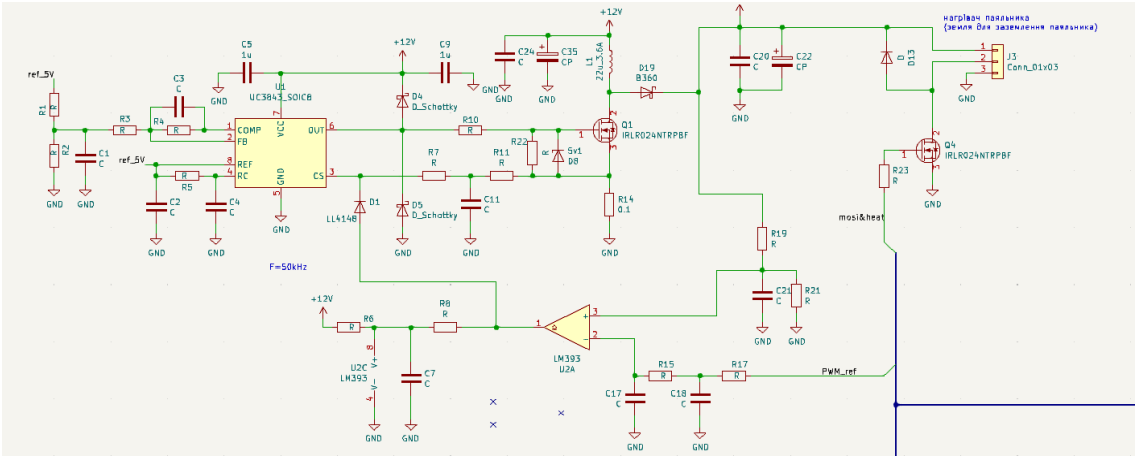

Рис. 3.2. Блок ШІМ-контролера з DC-DC перетворювачем.

З рис. 3.2. видно, що даний модуль представляє із себе генератор керованих імпульсів, що подаються на затвор польового транзистора IRLR024NTRPBF, що закривають або відкривають ключ. Під час відкриття ключа, струм дроселя L1 накопичується, а під час закриття дросель віддає енергію через діод D19 у навантаження (нагрівач паяльника) та фільтр C20, C22.

Діоди D4, D5 використовуються для захисту ШІМ-контролера від перенапруги та ефекту «защіпування мікросхеми».

Після цього, напруга зменшується дільником напруги R19,R21 та фільтрується синтезованим ФНЧ на базі C21 та дільником напруги описаним вище.

Вихід з подільника надходить на інвертуючий вхід компаратора LM393. З іншого боку, на неінвертуючий вхід подається опорна напруга. Після перевищення рівня опорної напруги, на виході компаратора з’явиться високий рівень напруги. Вихідна напруга компаратора через діод D1 подається на контакт «CS» ШІМ-контролера. Скважність (duty cycle) контролера дорівнює 95%, однак при перевищенні опорної напруги (в нашому випадку-1В),транзистор закривається до кінця періоду.

Транзистор Q1, у колі витоку має резистор R14, який використовується як шунт. Після цього виміряна напруга надходить на ФНЧ R11,C11 для відбиття високочастотних пульсацій та шумів. Стабілітрон D8 включений паралельно з резистором R22 для захисту від пробою Q1. Резистор R10 обмежує струм затвору в момент перемикання контактів.

C20, C22 використовуються у якості фільтрації лінії 24В,після цього напруга подається на нагрівач паяльника, тобто на роз’єм J3. Зворотній діод D13 увімкнений в розрив нагрівача, що усуває від індуктивних складових та захищає транзистор Q4.

На рис. 3.3 зображено блок термопари:

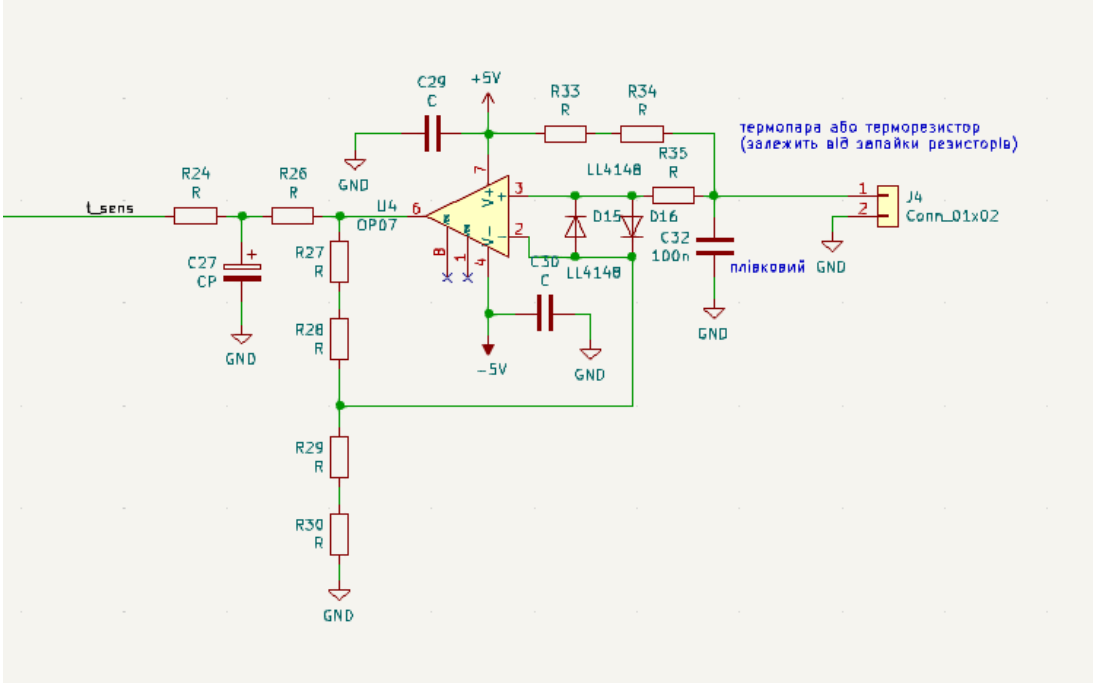

Рис. 3.3. Блок термопари.

Термопара паяльника підключається через роз’єм J4. В основі блоку, лежить прецизійний операційний підсилювач, увімкнений по схемі неінвертуючого підсилювача. Використання прецизійного операційного підсилювача для вимірювання сигналів від термопари є досить поширеним підходом у багатьох додатках, де потрібно виміряти температуру з високою точністю.

Термопара - це пристрій для вимірювання температури, який працює на принципі термоелектричного ефекту. Вона складається з двох провідників різних матеріалів, з'єднаних в одній точці. При наявності різниці температур між точкою з'єднання та кінцями провідників утворюється електрична напруга, яка залежить від температури.

Прецизійний операційний підсилювач використовується для підсилення слабкого сигналу термопари та його подальшої обробки. Він має високу точність, стабільність і низький рівень шуму, що дозволяє отримати достовірні вимірювання температури.

Операційний підсилювач може бути налаштований для підсилення сигналу термопари відповідно до потреб конкретного додатку [9]. Крім того,

він може мати вбудовані функції компенсації температурних ефектів, такі як зміщення нуля, що дозволяє забезпечити більш точні вимірювання навіть при зміні умов роботи. Існує кілька способів підключення термопари до операційного підсилювача, включаючи наступні:

1. Зворотний зв'язок через резистор зміщення: В цій схемі термопара підключається між вхідними пінами підсилювача, а резистор зміщення підключається між цими пінами і землею. Це дозволяє забезпечити стабільну роботу підсилювача при зміні температури.

2. Зсув напруги зміщення: В цій схемі термопара підключається між вхідними пінами підсилювача, а резистор зміщення підключається між входом підсилювача і землею. Це дозволяє зсунути напругу зміщення підсилювача, що поліпшує точність вимірювання.

3. Внутрішнє підсилення: Деякі операційні підсилювачі мають вбудовані функції підсилення, спеціально призначені для термопар. В цьому випадку термопара підключається безпосередньо до входів підсилювача.

На рис. 3.4. зображений блок зарядового насосу К1.

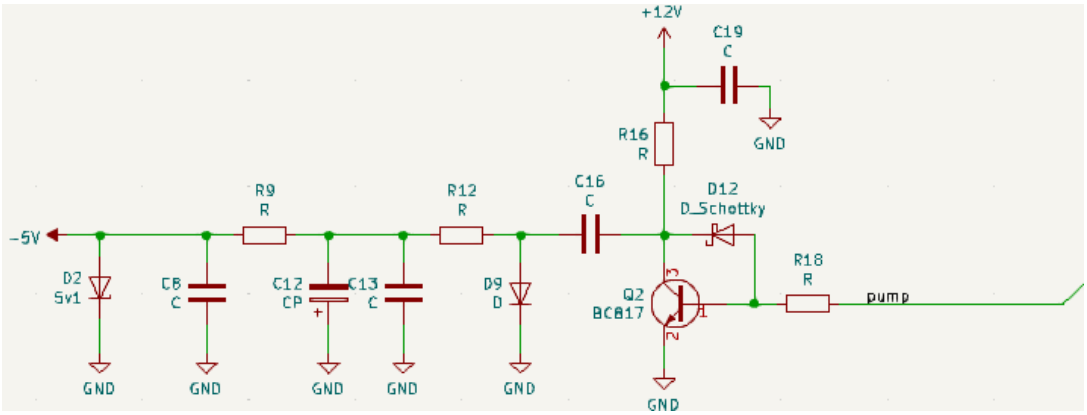

Рис. 3.4. Блок зарядового насосу

З рис. 3.4. видно, що для забезпечення напруги -5В використовується схема зарядового насосу на біполярному транзисторі Q2.

Коли транзистор закритий, через опір R16 лінією +12В тече струм, що заряджає конденсатор С16 через діод D9 майже до рівня напруги живлення.

При відкритому транзисторі, конденсатор C16 розряджається, через R12, який встановлений для обмеження струму перехідного процесу.

Обмежувальний резистор R13 захищає порт мікроконтролера, якщо лінія PB2 випадково сконфігурована на вихід.

На рис. 3.5. зображено лінійний стабілізатор LM1117 3.3В.

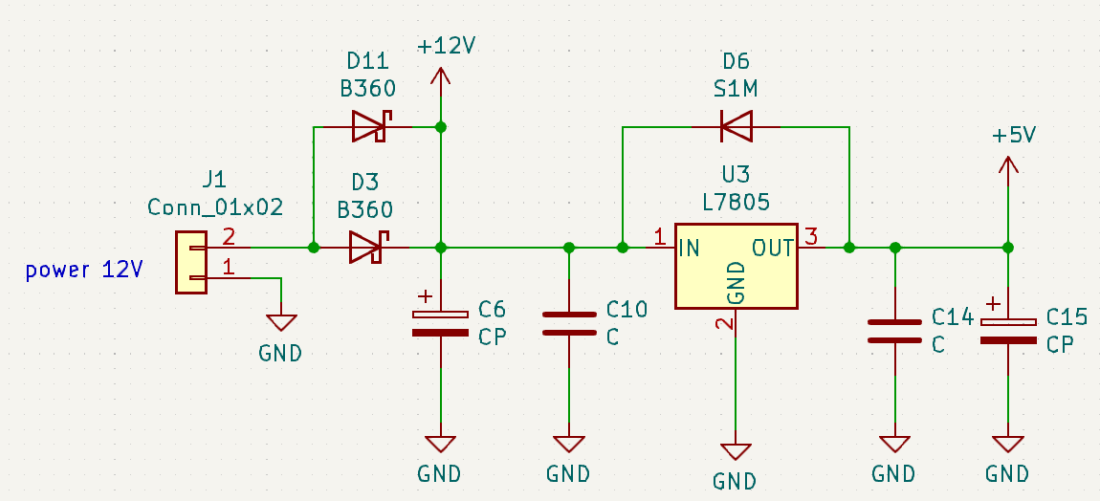

Рис. 3.5. Лінійний стабілізатор LM1117 3.3В

З рис. 3.5 видно, що стабілізатор використовується для стабілізації рівня вихідної напруги до 5 В. Вхідна напруга приходить на клеми J1, захисні діоди Шотткі D11, D3 забезпечують захист від «переполюсовки».

ВИСНОВКИ

В даній роботі розроблена цифрова паяльна станція на основі мікроконтролера ATmega8 (ATmega88). Як і промислові зразки, розроблена станція містить цифровий дисплей, який показує як параметри налаштування, так і поточну температуру паяльника. Головною відмінністю даної паяльної станції від промислових зразків є плавне регулювання напруги нагрівача паяльника, що, теоретично, має збільшити час його роботи, адже нагрівальний елемент не зазнає постійних термічних напружень. Звичайно, для кількісної оцінки ефекту від даного нововведення треба проводити додаткові випробування, які виходять за рамки дипломної роботи.

Окрім того, відмінністю даної розробки є наявність режиму швидкої (шляхом однократного натискання кнопки) зміни температури на задану величину. Даної функції, яка збільшує зручність пропайки полігонів і масивних провідників, не вистачає у промислових зразках.

СПИСКИ ВИКОРИСТАНИХ ДЖЕРЕЛ

1. "Trends in the Semiconductor Industry". Semiconductor History Museum of Japan. Archived from the original on 2019-06-27. Retrieved 2019-06-27.

2. Грайворовський М. В., Новіков О. М. Безпека інформаційно-комунікаційних систем: підручник для вищих навчальних закладів. Київ: видавнича група BHV, 2009. 607 с.,

3. AVR 8/16-bit XMMEGA A4 Microcontroller. Datasheet, ,2010. p 108

4. ATtiny24/44/84/ 8-bit AVR Microcontroller with 2/4/8K Bytes in-system Programmable flash: datasheet. 2015. p. 196

5. RScomponents. Каталог продукції [Електронний ресурс]. – Режим доступу до ресурсу:

https://www.rcscomponents.kiev.ua/modules.php?name=stocks&query=DC-

6. Паяльник Saneryigo SH72 65Вт 24В [Електронний ресурс]. – Режим доступу до ресурсу:

https://arduino.ua/ru/prod4393-payalnik-saneryigo-sh72-65w-24v

7. Паяльник акумуляторний, швидкого нагріву, 4.8V, HandsKit [Електронний ресурс] . – Режим доступу до ресурсу:

https://cables.kiev.ua/pajalnik-handskit-akumuljatornii-48v.html

8. Розбір схем та принцип роботи паяльників серії S-L10 [Електронний ресурс] . – Режим доступу до ресурсу:

https://www.aks.ua/item/view/79885/payalnaya-stanciya-salomon-pensol-sl10-n-48w-230vac.html

9. Журнал «Радіо» [Електронний ресурс] / A. Pyattaev, D. Solomitckii, A. Ometov – Режим доступу до ресурсу: radio_092022.pdf

ДОДАТОК. А. ПРОГРАМНИЙ КОД МІКРОКОНТРОЛЕРА

```
#include <avr/io.h>
#include <avr/wdt.h>
#include <avr/interrupt.h>  /* for sei() */
#include <util/delay.h>     /* for _delay_ms() */
#include "eeprom.h"
//**********************CONSTANTS
#define ON 0xFF
#define OFF 0
#define def_min_temperture 150 //мінімальна температура
#define def_max_temperature 450 //максимальна температура в нормальному режимі
#define def_max_temperature_for_students 280 // максимальна температура в обмеженому режимі
#define def_EEPROM_struct_addr 0x0001
//************************
#define set_flag(struct_name,field_name,flag) struct_name.field_name[((int)(flag/8))]|=(1<<(flag%8)) //
встановлення флагу
#define clear_flag(struct_name,field_name,flag) struct_name.field_name[((int)(flag/8))]&=~(1<<(flag
%8)) //очищення флагу
#define xor_flag(struct_name,field_name,flag) struct_name.field_name[((int)(flag/8))]^=(1<<(flag%8)) //
зміна флагу на протилежний
#define get_flag(struct_name,field_name,flag) (struct_name.field_name[((int)(flag/8))]&(1<<(flag
%8))) //читання флагу
#define buzzer_on() PORTB|=(1<<PB5)
#define buzzer_off() PORTB&=~(1<<PB5)
#define led1(x) PORTC=((PORTC&(~(1<<PC2)))|(x&(1<<PC2)))
#define led2(x) PORTC=((PORTC&(~(1<<PC3)))|(x&(1<<PC3)))
#define buzzer_on() PORTB|=(1<<PB5) //включили пищалку
#define buzzer_off() PORTB&=~(1<<PB5) //виключили пищалку
#define heat_on() PORTB&=~(1<<PB3) //включеня нагрівача польовиком
#define heat_off() PORTB|=(1<<PB3) //виключеня нагрівача польовиком
#define relay_on() PORTB|=(1<<PB4) //вмикаємо реле
#define relay_off() PORTB&=~(1<<PB4) //вимикаємо реле
#define start_ADC() ADCSRA|=(1<<ADSC)
#define incT PC5
#define decT PC4
#define forceT PC1
//************************
void display(unsigned char ,unsigned char );
```

```
//*************************
typedef struct
{unsigned char digit1;
unsigned char digit2;
unsigned char digit3;}
digits;
typedef struct
{unsigned int adc_data;//дані з АЦП
unsigned int temperature;//поточна температура жала
unsigned int desired_temperature;//бажана температура жала
unsigned int display;//число, яке буде відображатись на дисплеї
unsigned char count_digit;//номер відображуваної цифри при динамічній індикації
unsigned char flags[2];//масив флагів
#define MHT 0 //Main Heat Trigger тригер вмикання основного нагрівача, необхідно для системи підігріву
#define PT 1 //Preheater Trig - тригер підігрівача
#define BPT 2 //Ban Preheater Trig - тригер заборони зменшення напруги підігрівача
#define SH 3 //Soft Heat - повільний підігрів - використовується при ввімкненні паяльника - він забороняє включення основного нагрівача
#define PK 4 //Push Key - натиснута якась кнопка
#define LMT 5 //Limited Max Temperature - режим обмеженості максимальної температури}
global_variables;
//**********************глобальні змінні
global_variables gv;
//**********************
digits num_to_digit(unsigned int num)
{digits dg;
dg.digit1=(unsigned char)(num/100);
num=num%100;
dg.digit2=(unsigned char)(num/10);
num=num%10;
dg.digit3=(unsigned char)num;
return dg;}
//**********************
ISR(TIMER2_OVF_vect)
{digits dg;
dg=num_to_digit(gv.display);//отримуємо числа для відображення
gv.count_digit++;
```

```
if(gv.count_digit>3)
{gv.count_digit=1;
gv.adc_data=ADCW;//для 10 бітного режиму не працює окремо через ADCH та ADCL
start_ADC();}
if(gv.count_digit==1)
{display(1,dg.digit1);}
if(gv.count_digit==2)
{display(2,dg.digit2);}
if(gv.count_digit==3)
{display(3,dg.digit3);}}
//**************************
void display(unsigned char dig,unsigned char num)
{PORTB|=(1<<PB0)|(1<<PB6)|(1<<PB7);//гасимо сегменти A,E,F
PORTD|=(1<<PD0)|(1<<PD1)|(1<<PD2)|(1<<PD4)|(1<<PD7);//гасимо сегменти C, G, dp, B, D
PORTD|=(1<<PD3)|(1<<PD5)|(1<<PD6);//вимикаємо транзистори розрядів
if(num>127)
{PORTD&=~(1<<PD2);//засвічуємо точку
num=255-num;}
switch(dig)
{case 1:
PORTD&=~(1<<PD5);//вмикаємо першу цифру
break;
case 2:
PORTD&=~(1<<PD6);//вмикаємо другу цифру
break;
case 3:
PORTD&=~(1<<PD3);//вмикаємо третю цифру
break;}
switch(num)
{case 0:
PORTB&=~((1<<PB0)|(1<<PB6)|(1<<PB7));
PORTD&=~((1<<PD0)|(1<<PD4)|(1<<PD7));
break;
case 1:
PORTD&=~((1<<PD0)|(1<<PD4));
break;
case 2:
PORTB&=~((1<<PB0)|(1<<PB7));
```

```
PORTD&=~((1<<PD1)|(1<<PD4)|(1<<PD7));
break;
case 3:
PORTB&=~(1<<PB7);
PORTD&=~((1<<PD0)|(1<<PD1)|(1<<PD7)|(1<<PD4));
break;
case 4:
PORTB&=~(1<<PB6);
PORTD&=~((1<<PD0)|(1<<PD1)|(1<<PD4));
break;
case 5:
PORTB&=~((1<<PB6)|(1<<PB7));
PORTD&=~((1<<PD0)|(1<<PD1)|(1<<PD7));
break;
case 6:
PORTB&=~((1<<PB0)|(1<<PB6)|(1<<PB7));
PORTD&=~((1<<PD0)|(1<<PD1)|(1<<PD7));
break;
case 7:
PORTB&=~(1<<PB7);
PORTD&=~((1<<PD4)|(1<<PD0));
break;
case 8:
PORTB&=~((1<<PB0)|(1<<PB6)|(1<<PB7));
PORTD&=~((1<<PD0)|(1<<PD1)|(1<<PD4)|(1<<PD7));
break;
case 9:
PORTB&=~((1<<PB6)|(1<<PB7));
PORTD&=~((1<<PD0)|(1<<PD1)|(1<<PD4)|(1<<PD7));
break;
case 10:
//A
PORTB&=~((1<<PB0)|(1<<PB6)|(1<<PB7));
PORTD&=~((1<<PD0)|(1<<PD1)|(1<<PD4));
break;
case 11:
//b
PORTB&=~((1<<PB0)|(1<<PB6));
```

```
PORTD&=~((1<<PD0)|(1<<PD1)|(1<<PD7));
break;
case 12:
//C
PORTB&=~((1<<PB0)|(1<<PB6)|(1<<PB7));
PORTD&=~(1<<PD7);
break;
case 13:
//d
PORTB&=~(1<<PB0);
PORTD&=~((1<<PD0)|(1<<PD1)|(1<<PD4)|(1<<PD7));
break;
case 14:
//E
PORTB&=~((1<<PB0)|(1<<PB6)|(1<<PB7));
PORTD&=~((1<<PD1)|(1<<PD7));
break;
case 15:
//F
PORTB&=~((1<<PB0)|(1<<PB6)|(1<<PB7));
PORTD&=~(1<<PD1);
break;
case 16:
//-
PORTD&=~(1<<PD1);
break;}}
void timer1_init()
{TCCR1A=(1<<COM1A1)|(1<<COM1B1);//Clear OC1A/OC1B on Compare Match, set OC1A/OC1B
at BOTTOM, (non-inverting mode)
TCCR1A|=(1<<WGM10);//Fast PWM, 8-bit
TCCR1B=(1<<WGM12);//Fast PWM, 8-bit
TCCR1B|=(1<<CS10);//clk I/O /1 (No prescaling)
TCNT1H=0;
TCNT1L=0;
OCR1AH=0;
OCR1AL=0;
OCR1BH=0;
OCR1BL=0;}
```

```
void timer2_init()
{TCCR2=(1<<CS22)|(1<<CS20);//clk T2S /128 (From prescaler)
TIMSK|=(1<<TOIE2);
OCR2=127;}
void ADC_init()
{ADMUX=0;//AREF, Internal V ref turned off; ADC0
ADCSRA=(1<<ADEN)|(1<<ADSC)|(1<<ADPS2)|(1<<ADPS1);}
//**************************
void EEPROM_write_struct(global_variables gvst)
{unsigned char i;
for(i=0;i<sizeof(gvst);i++)
{EEPROM_write_byte(def_EEPROM_struct_addr+i,*(&(gvst.adc_data)+i));}}
//***********
void EEPROM_read_struct(global_variables* gvst)
{unsigned char i;
for(i=0;i<sizeof(*gvst);i++)
{*(&((*gvst).adc_data)+i)=EEPROM_read_byte(def_EEPROM_struct_addr+i);}}
//**************************
int main()
{unsigned char preheater;
unsigned char keys;//клавіші
unsigned char akselerator;//прискорювач підбору напруги підігрівача
unsigned long ulong_temp;
unsigned char counter_for_display;
//************************INITIALIZATION
DDRB=(1<<PB0)|(1<<PB1)|(1<<PB2)|(1<<PB3)|(1<<PB4)|(1<<PB5)|(1<<PB6)|(1<<PB7);
DDRC=(1<<PC2)|(1<<PC3);
PORTC|=(1<<PC1)|(1<<PC4)|(1<<PC5);
DDRD=0xFF;
buzzer_off();
timer1_init();
timer2_init();
ADC_init();
heat_off();
relay_off();
OCR1BH=0;
OCR1BL=127;
led1(OFF);
```

```
led2(OFF);
//set_flag(gv,flags,LMT);
//************************************************
relay_on();
sei();
OCR1AL=255;//вимикаємо підігрівач
start_ADC();

akselerator=1;
set_flag(gv,flags,SH); //встановлюємо флаг повільного підігріву паяльника - це забороняє включення основного нагрівача
preheater=0;// встановлюємо максимальну напругу підігрівача
counter_for_display=0;
//*********************************режим обмеження макисмальної температури
//clear_flag(gv,flags,LMT);//очищаємо флаг режиму обмеження температури згори
EEPROM_read_struct(&gv);
if( ((PINC&(1<<incT))==0) && ((PINC&(1<<decT))==0) )
{//будемо тут, якщо затиснуті обидві кнопки одночасно
xor_flag(gv,flags,LMT);//інвертуємо флаг режиму обмеження температури згори
buzzer_on();
EEPROM_write_struct(gv);//записуємо ЄЄПРОМ
_delay_ms(300);
buzzer_off();}
EEPROM_read_struct(&gv);//читаємо ЄЄПРОМ
if(get_flag(gv,flags,LMT)==0)
{//встановлений звичайний режим макисмальної температури
led2(ON);//вмикаємо зелений світлодіод, що означає нормальний режим
_delay_ms(1000);}
else
{//встановлений обмежений режим макисмальної температури
led1(ON);//вмикаємо червоний світлодіод, що означає обмежений режим
if(gv.desired_temperature>def_max_temperature_for_students)
{gv.desired_temperature=230;}
_delay_ms(1000);}
led1(OFF);
led2(OFF);
//*********************************************************************
while(1)
```

```
{
//----------------------------------------------------------------------------------------------------//перетворення значення АЦП в реальну температуру
ulong_temp=gv.adc_data*63;
ulong_temp=(unsigned long)(ulong_temp/100);
gv.temperature=(unsigned int)ulong_temp+48;
//************************************************************
if(get_flag(gv,flags,SH)!=0)
{//будемо тут, якщо встановлений флаг повільного підігріву, тобто немає заборони на основний нагрів
if( (gv.temperature>gv.desired_temperature) || (gv.temperature>250) )
{//будемо тут, коли досягнута бажана температура або 250 С
clear_flag(gv,flags,SH); //скидаємо флаг повільного підігріву паяльника
//preheater=127;//середня напруга підігрівача}}
//----------------------------------------------------------------------------------------------------підбір напруги підігрівача
if(gv.temperature<gv.desired_temperature-5)
{//будемо тут, коли необхідно ввімкнути основний нагрівач
if(get_flag(gv,flags,SH)==0)
{//будемо тут, якщо скинутий флаг повільного підігріву, тобто немає заборони на основний нагрів
heat_on();//вмикаємо основний нагріач
led1(ON);}
if(get_flag(gv,flags,MHT)==0)
{//будемо тут, якщо скинутий тригер основного нагрівача (щоб цикл не виконувався декілька
 разів)
//************************************************************БЛОК ПРИСКОРЕННЯ
if(get_flag(gv,flags,BPT)!=0)
{//будемо тут, якщо встановлений тригер заборони зменшення температури підігрівача
//це означає, що між включеннями основного нагрівача не була досягнута температура,
//при якій зменшується напруга підігрівача, а тому вмикаємо прискорення
akselerator++;
if(akselerator>10)
{akselerator=10;}}
else
{akselerator=1;}
//***
if(preheater>akselerator)
{preheater=preheater-akselerator;}
```

```
else
{if(preheater>0)
{preheater--;}}}
set_flag(gv,flags,MHT);//встанвлюємо тригер основного нагрівача
set_flag(gv,flags,BPT);//встанвлюємо тригер заборони зменшення температури підігрівача}
else
{heat_off();
led1(OFF);
clear_flag(gv,flags,MHT);//скидаємо тригер основного нагрівача}
//*****************************************************************************
if(gv.temperature>gv.desired_temperature+5)
{//будемо тут, коли температура перевищить верхню межу, коли треба зменшувати напругу
підігрівача
OCR1AL=255;//вимикаємо підігрівач (0 -максимальна напруга, 255 - нульова напруга)
if( ( get_flag(gv,flags,PT)==0 ) && ( get_flag(gv,flags,BPT)==0 ) )
{//будемо тут, коли скинуті тригери підігрівача та заборони зміни напруги підігрівача
//перший, щоб зменшення апруги не повторювалась багаторазово
//другий забороняє зменшувати напругу підігрівача, якщо перед цим був включений основний
нагрівач
//****************************************БЛОК ПРИСКОРЕННЯ
if(get_flag(gv,flags,BPT)==0)
{//будемо тут, якщо скинутий тригер заборони зменшення температури підігрівача
//це означає, що не відбувалось включення основного нагрівача, а тому вмикаємо прискорення
akselerator++;
if(akselerator>10)
{akselerator=10;}}
else
{akselerator=1;}
//*********************************K
if( (preheater+akselerator)<255 )
{//ця умова, щоб не вийти за межі 255
preheater=preheater+akselerator;}
else
{if(preheater!=255)
{preheater++;}}
//*****************************
}set_flag(gv,flags,PT);//встанвлюємо тригер підігрівача
clear_flag(gv,flags,BPT);//скидаємо тригер заборони зменшення температури підігрівача}
```

```
else
{clear_flag(gv,flags,PT);//скидаємо тригер підігрівача
OCR1AL=preheater;}
//----------------------------------------------------------------------кінець підбору напруги підігрівача
//-------------------------------------------------------------------------------------блок обробки кнопок
if(counter_for_display==0)
{gv.display=gv.temperature;
led2(OFF);}
else
{gv.display=gv.desired_temperature;
led2(ON);
counter_for_display--;
if(counter_for_display==0)
{//процедура запису в ЄЄПРОМ
EEPROM_write_struct(gv);}}
keys=PINC;
if( ((keys&(1<<incT))!=0) && ((keys&(1<<decT))!=0) && ((keys&(1<<forceT))!=0) )
{//будемо тут, якщо відпущені всі кнопки
clear_flag(gv,flags,PK);//очищаємо флаг натисненості якоїсь кнопки}
else
{set_flag(gv,flags,PK);//встановлюємо флаг натисненості якоїсь кнопки}
//********інкремент температури
if((keys&(1<<incT))==0)
{//натиснута кнопка інкременту температури
gv.desired_temperature++;//інкрементуємо температуру
if(get_flag(gv,flags,LMT)==0)
{//будемо тут, якщо виключений режим обмеження температури
if(gv.desired_temperature>def_max_temperature)
{//перевищена верхня межа температури
gv.desired_temperature=def_max_temperature;//обмежуємо температуру}}
else
{//будемо тут, якщо включений режим обмеження температури
if(gv.desired_temperature>def_max_temperature_for_students)
{//перевищена верхня межа температури
gv.desired_temperature=def_max_temperature_for_students;//максимальна температура в обмеженому
режимі}}
counter_for_display=10;//затримка для показів бажаної температури                    }
//*******декремент температури
```

```
if((keys&(1<<decT))==0)
{//натиснута кнопка декременту температури
gv.desired_temperature--;//інкрементуємо температуру
if(gv.desired_temperature<def_min_temperture)
{gv.desired_temperature=def_min_temperture;//обмеження температури знизу}
counter_for_display=10;//затримка для показів бажаної температури}
_delay_ms(200);}
return 0;}
```