

**МІНІСТЕРСТВО ВНУТРІШНІХ СПРАВ УКРАЇНИ
ЛЬВІВСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ ВНУТРІШНІХ СПРАВ
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ УПРАВЛІННЯ, ПСИХОЛОГІЇ
ТА БЕЗПЕКИ**

Кафедра інформаційних технологій

**РОЗРОБЛЕННЯ ЦИФРОВОГО БАГАТОФУНКЦІОНАЛЬНОГО
ПРИСТРОЮ ДЛЯ ПЛАНУВАННЯ ТРИВАЛИХ ЧЕРГУВАНЬ ТА
ОЦІНКИ СТАНУ ОСОБОВОГО СКЛАДУ В ЕКСТРЕМАЛЬНИХ
УМОВАХ**

Кваліфікаційна робота
здобувача вищої освіти
4 курсу денної форми навчання
Олександра КОПАНЄВ

Науковий керівник:
доцент, кандидат технічних наук_
Ігор ФАРМАГА

Рецензент:

вчене звання, науковий ступінь

(Ім'я ПРИЗВИЩЕ рецензента)

Кваліфікаційна робота допущена до захисту

«___» _____ 2026 р., протокол № _____

Завідувач кафедри інформаційних технологій

Олег ЗАЧЕК

(підпис)

Львів
2026

ЗМІСТ

ВСТУП.....	3
РОЗДІЛ 1. СТАН ПРОБЛЕМНОЇ ОБЛАСТІ.....	9
1.1. Характеристики об'єктів вимірювання та способи відображення даних.....	9
1.2. Огляд існуючих розробок багатофункціональних цифрових пристроїв.....	14
РОЗДІЛ 2. ЗАСОБИ РОЗРОБКИ АПАРАТНОГО ТА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ЦИФРОВОГО ПРИСТРОЮ ДЛЯ ПЛАНУВАННЯ ТРИВАЛИХ ЧЕРГУВАНЬ ТА ОЦІНКИ СТАНУ ОСОБОВОГО СКЛАДУ.....	18
2.1. Система автоматизованого проектування і моделювання програмованих пристроїв Proteus VSM.....	18
2.2. Програмне забезпечення WinAVR та мова C як інструментарій проектування систем на базі контролерів AVR.....	21
РОЗДІЛ 3. АПАРАТНЕ ЗАБЕЗПЕЧЕННЯ ЦИФРОВОГО ПРИСТРОЮ ДЛЯ ПЛАНУВАННЯ ТРИВАЛИХ ЧЕРГУВАНЬ ТА ОЦІНКИ СТАНУ ОСОБОВОГО СКЛАДУ.....	25
3.1. Обґрунтування вибору апаратних засобів для реалізації цифрового пристрою для планування тривалих чергувань та оцінки стану особового складу.....	25
3.2. Проектування цифрового пристрою для планування тривалих чергувань та оцінки стану особового складу в екстремальних умовах.....	74
РОЗДІЛ 4. ПРОГРАМНО-АЛГОРИТМІЧНЕ ЗАБЕЗПЕЧЕННЯ ЦИФРОВОГО ПРИСТРОЮ ДЛЯ ПЛАНУВАННЯ ТРИВАЛИХ ЧЕРГУВАНЬ ТА ОЦІНКИ СТАНУ ОСОБОВОГО СКЛАДУ В ЕКСТРЕМАЛЬНИХ УМОВАХ.....	76
4.1. Алгоритм роботи цифрового пристрою для планування тривалих чергувань та оцінки особового складу.....	76
4.2. Розроблення програмного модуля інтеграції датчика температури DS18B20.....	81
4.3. Розроблення програмного модуля обробки даних з сенсора тиску MPX4115.....	82
4.4. Розроблення програмних засобів взаємодії з годинником реального часу DS1307.....	82
4.5. Розроблення драйвера для підключення алфавітно-цифрового РКД-модуля.....	83
4.6. Програмна реалізація алгоритму роботи цифрового пристрою для планування тривалих чергувань та оцінки особового складу.....	84
4.7. Моделювання цифрового пристрою для планування тривалих чергувань та оцінки стану особового складу в екстремальних умовах в Proteus ISIS.....	85
ВИСНОВКИ.....	89
СПИСОК ВИКОРИСТАНИХ ЛІТЕРАТУРНИХ ДЖЕРЕЛ.....	90
ДОДАТКИ.....	92

СПИСОК УМОВНИХ СКОРОЧЕНЬ

АЦП (ADC) – аналого-цифровий перетворювач;

АЗ – апаратне забезпечення;

ВС – вбудована система;

ЕК – електрична схема;

МК – мікроконтролер;

ОЗП (RAM) – оперативний запам'ятовуючий пристрій (абр. англ. Random Access Memory);

ПК – персональний комп'ютер;

ПЗ – програмне забезпечення;

ПЗП (ROM) – постійний запам'ятовуючий пристрій (абр. англ. Read-Only Memory);

РКД (LCD) – рідкокристалічний дисплей (абр. англ. Liquid Crystal Display);

САПР – система автоматизованого проектування;

ЦП – цифровий пристрій;

AVR – архітектура 8-розрядних мікроконтролерів компанії Atmel (Microchip Technology);

EEPROM – (англ. Electrically Erasable Programmable Read-Only Memory) – електрично стираєна перепрограмовувана постійна пам'ять;

FLASH – тип енергонезалежної напівпровідникової пам'яті з можливістю електричного перезапису;

I²C (TWI) – (англ. Inter-Integrated Circuit) – двопровідний послідовний інтерфейс обміну даними між інтегральними схемами;

SPI – (англ. Serial Peripheral Interface) – послідовний синхронний інтерфейс повнодуплексної передачі даних для зв'язку мікроконтролерів із периферією;

1-Wire – послідовний двопровідний (із врахуванням лінії заземлення) інтерфейс обміну даними, де передача сигналів та живлення периферії можуть здійснюватися по одній сигнальній лінії.

АНОТАЦІЯ

Копанєв О., Фармага І. (керівник). Розроблення цифрового багатофункціонального пристрою для планування тривалих чергувань та оцінки стану особового складу. Бакалаврська кваліфікаційна робота. – Львівський державний університет внутрішніх справ, Львів, 2025.

У кваліфікаційній роботі здійснено проектування апаратного і програмного забезпечення цифрового пристрою, призначеного для оптимізації планування тривалих чергувань та оперативного моніторингу навколишнього середовища під час виконання завдань особовим складом поліції. Пристрій поєднує в собі функції прецизійного реального часу (годинник-будильник), а також модулі цифрової термометрії та барометрії для контролю екстремальних кліматичних умов на місцях дислокації підрозділів.

Апаратно-програмну архітектуру пристрою побудовано на базі мікроконтролера AVR ATmega32, який координує роботу трьох вимірювальних каналів: хронометричного (RTC мікросхема DS1307 через інтерфейс I2C), термометричного (цифровий давач DS18B20 через шину 1-Wire) та барометричного (аналоговий давач MPX4115, підключений до вбудованого АЦП). Візуалізація даних здійснюється на алфавітно-цифровому РК-дисплеї формату 16×2. Програмне забезпечення МК реалізує циклічний збір телеметрії та підтримує інтерактивне меню для конфігурування часу, календаря, будильника та форматів відображення. Засобами САПР Proteus розроблено електричну принципову схему та комп'ютерну модель цифрового пристрою. Створено алгоритм функціонування системи та програмне забезпечення мікроконтролера ATmega32 в середовищі WinAVR. Комплексне моделювання, тестування та аналіз працездатності розробленого пристрою в різних режимах роботи проведено за допомогою симулятора Proteus ISIS.

Ключові слова: цифровий пристрій, мікроконтролер AVR, годинник реального часу DS1307, термометр DS18B20, барометр MPX4115A, РК-дисплей HD44780, вбудоване ПЗ, мова програмування C, САПР Proteus VSM, WinAVR.

ABSTRACT

Kopaniev O., Farmaha I. (supervisor). Development of a digital multifunctional device for planning long shifts and assessing the condition of personnel. Bachelor's thesis. – Lviv State University of Internal Affairs, Lviv, 2026.

The qualification work involved designing the hardware and software of a digital device designed to optimize the planning of long shifts and operational monitoring of the environment during the performance of tasks by police personnel. The device combines precision real-time functions (alarm clock), as well as digital thermometry and barometry modules to monitor extreme climatic conditions at the locations of units.

The hardware and software architecture of the device is based on the AVR ATmega32 microcontroller, which coordinates the operation of three measurement channels: chronometric (RTC chip DS1307 via I2C interface), thermometric (digital sensor DS18B20 via 1-Wire bus) and barometric (analog sensor MPX4115, connected to the built-in ADC). Data visualization is carried out on a 16×2 alphanumeric LCD display. The MK software implements cyclic telemetry collection and supports an interactive menu for configuring time, calendar, alarm clock and display formats.

An electrical schematic diagram and a computer model of a digital device were developed using Proteus CAD. The system operation algorithm and ATmega32 microcontroller software were created in the WinAVR environment. Comprehensive modeling, testing, and analysis of the performance of the developed device in various operating modes were carried out using the Proteus ISIS simulator.

Keywords: digital device, AVR ATmega32 microcontroller, DS1307 real-time clock, DS18B20 digital temperature sensor, MPX4115A analog pressure sensor, HD44780 alphanumeric LCD, Proteus VSM CAD, embedded C programming language, embedded software, WinAVR development toolkit.

ВСТУП

Сучасний прогрес у галузі мікроелектроніки, зокрема поява доступних мікроконтролерів та широкого спектру сенсорів, відкриває можливості для проектування економічно ефективних цифрових систем. Прикладом такої розробки є багатофункціональний цифровий пристрій, що інтегрує функції календаря, годинника-будильника, а також модулі вимірювання температури та атмосферного тиску. Даний пристрій орієнтований на оптимізацію графіків тривалих чергувань та моніторинг стану особового складу поліції під час роботи в екстремальних ситуаціях.

Мікроконтролери сьогодні є ключовим компонентом більшості цифрових систем. Їхня популярність серед інженерів пояснюється можливістю дешево та швидко реалізувати складні алгоритми керування. Оскільки більшість необхідних функціональних блоків уже вбудовані в мікросхему, потреба у додаткових вузлах зникає. У результаті це призводить до створення компактних пристроїв із низьким рівнем споживання енергії.

Стан атмосфери у конкретній локації та в певний момент часу безпосередньо впливає на умови несення служби, тому кількісна оцінка метеорологічних показників є критично важливою для планування тривалих чергувань і моніторингу стану особового складу в екстремальних ситуаціях. До ключових метеопараметрів, що потребують точного вимірювання, належать температура, відносна вологість, атмосферний тиск, а також параметри вітру (швидкість і напрямок), рівень хмарності та інтенсивність опадів. Сучасна метеорологія використовує дані, отримані за допомогою спеціалізованих технічних засобів, для прогнозування погоди, що має велике практичне значення. Пошук методів виявлення та передбачення погодних змін спонукав людство до постійного вдосконалення вимірювальних систем та інструментарію.

Мета роботи полягає у проектуванні апаратної частини та розробці мікропрограмного забезпечення для багатофункціонального цифрового комплексу. Пристрій, що поєднує в собі функції годинника-будильника,

термометра та барометра, призначений для оптимізації графіків тривалих чергувань та моніторингу психофізіологічного стану працівників поліції під час виконання завдань у складних умовах. Технічна реалізація апаратної платформи базується на використанні мікроконтролера архітектури AVR виробництва компанії Atmel.

Для реалізації поставленої мети визначено такі **завдання**:

- виконати схемотехнічне проєктування цифрового пристрою для планування тривалих чергувань та оцінки стану особового складу у середовищі САПР Proteus VSM;
- побудувати логічну схему (алгоритм) функціонування проєктованого пристрою;
- написати програмні компоненти (драйвери) низького рівня для взаємодії з модулем реального часу DS1307, термодатчиком DS18B20, аналоговим барометром MPX4115;
- реалізувати підпрограму керування LCD-індикатором для виведення текстової та цифрової інформації;
- об'єднати модулі в цілісне вбудоване програмне забезпечення (прошивку) мікроконтролера згідно з визначеною логікою роботи;
- здійснити комплексну верифікацію та аналіз роботи віртуального прототипу в інтерактивному симуляторі Proteus ISIS.

Об'єкт дослідження – процес автоматизованого збору телеметричної інформації, контролю параметрів навколишнього середовища та координації графіків несення служби особовим складом поліції.

Предмет дослідження – схемотехнічна структура, вбудоване програмне забезпечення та алгоритми взаємодії 8-розрядного мікроконтролера з цифровими й аналоговими сенсорами (температури, тиску) та модулем реального часу для оптимізації тривалих чергувань в екстремальних умовах.

Методи дослідження. Для досягнення поставленої мети та вирішення визначених завдань у роботі використано комплекс наукових та інженерних методів дослідження: методи системного аналізу та теорії проєктування

вбудованих систем, методи математичного моделювання, методи цифрової обробки сигналів та теорії інтерфейсів (1-Wire, I2C), методи комп'ютерного та схемотехнічного моделювання (САПР Proteus VSM/ISIS).

Структура роботи. Кваліфікаційна робота складається із вступу, чотирьох розділів, висновків, списку використаних джерел, додатків. Обсяг основного тексту роботи складає 89 сторінок, 41 рисунок, 18 таблиць, 1 додаток і 19 бібліографічних джерел. Загальний обсяг роботи – 94 сторінки.

РОЗДІЛ 1

СТАН ПРОБЛЕМНОЇ ОБЛАСТІ

1.1. Характеристики об'єктів вимірювання та способи відображення даних

Система 12-годинного відліку часу базується на розподілі повної доби на два рівні періоди. Для розрізнення ранкового та вечірнього часу використовуються латинські аббревіатури: a.m. (*ante meridiem* – перша половина дня) та p.m. (*post meridiem* – друга половина дня). Такий формат є основним у США, Австралії, Канаді (крім провінції Квебек), на Філіппінах та в Новій Зеландії. Водночас у низці країн, як-от Велика Британія, Бразилія, Греція та Франція, цей метод обчислення часу співіснує з традиційним 24-годинним форматом. Хоча у більшості країн світу офіційно домінує 24-годинна система, у побутовому мовленні часто віддають перевагу 12-годинному формату. Замість стандартних латинських аббревіатур (a.m. та p.m.) у таких випадках зазвичай використовують уточнювальні слова: «ранку», «дня», «вечора» або «ночі». Така популярність 12-годинного циклу в неформальному контексті зумовлена зручністю використання коротших числівників, а також традиційним виглядом класичного стрілкового циферблата [1].

Навіть за умови існування міжнародного стандарту ISO 8601, у світі досі немає єдиного підходу до позначення 12-ї години дня та ночі. Основна складність полягає в етимології латинських термінів *ante meridiem* та *post meridiem*, де слово *meridiem* трактується як “середина дня” або “полудень”. Через це виникає термінологічна неоднозначність: опівдні можна інтерпретувати і як “12 a.m.” (12 годин до полудня), і як “12 p.m.” (12 годин після полудня), що створює певні труднощі при проектуванні систем точного часу [1].

З логічної точки зору, опівночі також властива термінологічна подвійність: її можна класифікувати як “12 p.m.” (рахуючи 12 годин від

попереднього полудня) або як “12 a.m.” (за 12 годин до настання наступного). Щоб уникнути помилок у розрахунках, фахівці Національного морського музею в Грінвічі пропонують використовувати однозначні дефініції — “12 дня” та “12 ночі”. Аналогічного підходу дотримуються й укладачі словника The American Heritage Dictionary, вказуючи на доцільність описового позначення цих часових точок.

Таблиця 1.1. Зіставлення часових інтервалів у 24-годинній та 12-годинній системах числення

24-годинна шкала	12-годинна шкала	Вживана розмовна форма
00:00	12:00 a.m.	Північ (дванадцята година ночі)
01:00	1:00 a.m.	Перша година ночі
02:00	2:00 a.m.	Друга година ночі
03:00	3:00 a.m.	Третя година ночі
04:00	4:00 a.m.	Четверта година ранку
05:00	5:00 a.m.	П'ята ранку
06:00	6:00 a.m.	Шоста година ранку
07:00	7:00 a.m.	Сьома ранку
08:00	8:00 a.m.	Восьма година ранку
09:00	9:00 a.m.	Дев'ята ранку
10:00	10:00 a.m.	Десята година ранку
11:00	11:00 a.m.	Одинадцята ранку
12:00	12:00 p.m.	Полудень (дванадцята година дня)
13:00	1:00 p.m.	Перша година дня
14:00	2:00 p.m.	Друга година дня
15:00	3:00 p.m.	Третя година дня
16:00	4:00 p.m.	Четверта година дня
17:00	5:00 p.m.	П'ята година дня (після обіду)
18:00	6:00 p.m.	Шоста година вечора
19:00	7:00 p.m.	Сьома вечора
20:00	8:00 p.m.	Восьма година вечора
21:00	9:00 p.m.	Дев'ята вечора
22:00	10:00 p.m.	Десята година вечора
23:00	11:00 p.m.	Одинадцята вечора

Згідно з американськими стандартами ділового мовлення, для уникнення неоднозначності при визначенні межі доби рекомендується уникати терміна “північ”. Замість нього використовується позначення 11:59 p.m. для фіксації моменту завершення поточної доби та 12:01 a.m. для позначення початку нового дня. Такий підхід є загальноприйнятим у США при складанні юридичних угод, а також у транспортному плануванні (графіки руху потягів та автобусів), хоча поодинокі винятки з цього правила все ж існують.

У Сполучених Штатах 24-годинна система числення часу також має місце, проте вона відома під назвою “військовий час” (military time). Цей формат був офіційно впроваджений американськими збройними силами після завершення Другої світової війни. Головною метою такого переходу стало забезпечення ефективної координації та зручної взаємодії з арміями інших країн світу, де 24-годинний стандарт є домінуючим.

У структурах, де точність має критичне значення — зокрема в армії, авіації, медицині та рятувальних службах – 12-годинний відлік часу вважається неприпустимим через високі ризики помилок. Для гарантування максимальної безпеки та уникнення плутанини в цих галузях, а також у метрології, використовують виключно 24-годинний формат. Це дозволяє чітко координувати дії без ризику переплутати день із ніччю.

Таблиця 1.2. Регіональні особливості відображення часових та календарних даних

Регіон/Країна	Формат часу (основний)	Формат дати (прийнятий)	Приклад (14:30, 5 березня 2026)
США, Філіппіни	12-годинний (AM/PM)	ММ/ДД/РРРР	02:30 PM, 03/05/2026
Європа (Україна, Німеччина)	24-годинний	ДД.ММ.РРРР	14:30, 05.03.2026
Велика Британія	Змішаний (12/24)	ДД/ММ/РРРР	2:30 pm або 14:30, 05/03/2026
Китай, Японія, Корея	24-годинний	РРРР/ММ/ДД	14:30, 2026/03/05
Канада (Квебек)	24-годинний	РРРР-ММ-ДД	14:30, 2026-03-05

Температура (атмосферна, ґрунтова або водна) виступає фізичним показником теплового стану об’єкта, що відображає ступінь його нагрятості. Оскільки повітряні маси є фізичними тілами, вони завжди володіють певним тепловим запасом, який перевищує значення абсолютного нуля. Показники температури в атмосферних шарах перебувають у стані постійної динаміки; вони безперервно трансформуються в часі та суттєво різняться залежно від географічного розташування спостережного пункту.

На земній кулі температура повітря демонструє надзвичайну просторову мінливість. Історичні межі спостережень фіксують максимум трохи вище +50

°C у пустелях та мінімум близько -90 °C на південному полюсі. Важливо зазначити, що рух угору крізь атмосферні шари супроводжується неоднорідною зміною тепла. Якщо в початковому шарі (до 15 км) повітря стає холоднішим, то в наступному (до 60 км) воно знову прогрівається, що зумовлено особливостями поглинання сонячної радіації різними шарами атмосфери.

Для кількісного оцінювання теплового стану навколишнього середовища у фізиці та метеорології використовують загальноприйнятну шкалу Цельсія. В межах системи СІ саме ці одиниці є основними для фіксації температури води, землі чи атмосфери. Конструкція цієї вимірювальної системи спирається на два стабільні стани води за нормального тиску (1013 ГПа): початок танення льоду прийнято за нульову відмітку, тоді як процес кипіння маркується як 100°C.

Окрім системи Цельсія, у наукових розрахунках та теоретичній фізиці фундаментальне значення має абсолютна шкала температур (шкала Кельвіна). Її початкова точка — абсолютний нуль – теоретично відповідає стану повного спокою молекул, що є гранично низькою можливою температурою. У метриці Цельсія ця величина становить -273,15°C. Важливою особливістю є ідентичність ціни поділки обидвох систем: зміна на 1К еквівалентна зміні на 1°C. Температурні показники за цією шкалою завжди мають позитивне значення. Для розрізнення у формулах абсолютну температуру прийнято позначати великою літерою Т, тоді як значення за Цельсієм – малою t.

Окрім метричних систем, у деяких країнах (зокрема в США) досі активно використовується шкала Фаренгейта, розроблена німецьким фізиком Г. Фаренгейтом у 1724 році. В основі цієї системи лежить поділ інтервалу між фазовими переходами води (замерзанням та кипінням) на 180 рівних частин. Згідно з цим стандартом, температура танення льоду фіксується на позначці +32°F, що відповідає 0°C у загальноприйнятій системі Цельсія. Формули для переведення температури з Цельсія у Фаренгейти: $t_{\text{°F}} = 1,8 \cdot t_{\text{°C}} + 32$ і навпаки: $t_{\text{°C}} = 5/9 \cdot (t_{\text{°F}} - 32)$.

Атмосферний тиск визначається як сила гідростатичного тиску повітряних мас, що діє на одиницю площі поверхні. Фізично цей показник еквівалентний вазі вертикального стовпа повітря, який простягається від точки вимірювання до верхньої межі атмосфери. Сумарна маса газової оболонки нашої планети, яка створює цей постійний тиск на земну кору, оцінюється приблизно у $5,15 \times 10^{15}$ тонн.

Починаючи з XVII століття, за прикладом Е. Торрічеллі, атмосферний тиск традиційно вимірювали за допомогою ртутного стовпа, фіксуючи його висоту в міліметрах або дюймах. Проте з впровадженням математичних методів прогнозування та аналізу атмосферних процесів стало очевидним, що лінійні одиниці (міліметри) не відображають фізичну природу тиску як силової характеристики. Через незручність таких розрахунків у 1920-х роках норвезький вчений В. Б'єркнес запропонував використовувати спеціалізовану метеорологічну одиницю – мілібар (мбар).

Мілібар (мбар) – це одиниця вимірювання тиску, що визначається як сила в 1000 дин, яка діє на площу в 1 см^2 (де 1 дин надає масі в 1 г прискорення 1 см/с^2). Згідно з міжнародними метеорологічними стандартами, нормальний атмосферний тиск (зафіксований на рівні моря на широті 45° при 0°C) становить 1013,25 мбар, що еквівалентно 760 мм рт. ст. Натомість за стандартний тиск прийнято вважати 1000 гПа (750 мм рт. ст.). У сучасній системі SI основною одиницею є Паскаль (Па) – тиск, створюваний силою в 1 Н, яка рівномірно розподілена на площі 1 м^2 . У метеорології найчастіше використовують гектопаскаль (гПа), оскільки він чисельно ідентичний мілібару ($1 \text{ гПа} = 1 \text{ мбар}$).

Вимірювання атмосферного тиску з допомогою датчика MPX4115. Вихідний сигнал напівпровідникового датчика тиску MPX4115 є лінійною функцією від вимірюваного тиску. Математична залежність вихідної напруги (V_{out}) від тиску (P) та напруги живлення (V_s) описується рівнянням:

$$V_{out} = V_s \cdot (0,009 \cdot P - 0,095)$$

Для обробки цього сигналу використовується мікроконтролер ATmega32, оснащений 10-бітовим аналого-цифровим перетворювачем (АЦП). При опорній напрузі 5В крок квантування (ціна наймолодшого розряду, LSB) становить: $\Delta V = 5\text{В}/1023 \approx 0,00488\text{В}$, $\Delta V = \frac{5\text{В}}{1023} \approx 0,00488\text{В}$

Якщо N – це цифровий код, отриманий з регістрів АЦП, то відповідна йому вхідна напруга обчислюється як $V_{in} = N \cdot \Delta V$. $V_{in} = N \cdot \Delta V$. Таким чином, загальна формула зв'язку між кодом АЦП та тиском при напрузі живлення $V_s=5\text{В}$ набуває вигляду: $N \cdot 0,00488 = 5 \cdot (0,009 \cdot P - 0,095)$.
 $N \cdot 0,00488 = 5 \cdot (0,009 \cdot P - 0,095)$.

1.2. Огляд існуючих розробок багатофункціональних цифрових пристроїв

Аналіз науково-технічної літератури та сучасних інтернет-ресурсів дозволив систематизувати дані щодо існуючих розробок багатофункціональних цифрових систем. На сьогодні ринок та інженерна спільнота пропонують широкий спектр рішень, що поєднують у собі функції хронометражу, вимірювання температури та атмосферного тиску. Особлива увага в сучасних проектах приділяється використанню мікроконтролерної техніки як бази для побудови пристроїв, а також інтеграції модулів бездротової передачі даних для дистанційного моніторингу.

На Рис. 1.1–1.3 представлено приклади сучасних цифрових систем, що інтегрують функції годинника, вимірювання температури та атмосферного тиску з подальшою візуалізацією даних на рідкокристалічних або сегментних дисплеях [2]. Ці розробки демонструють різні підходи до компонування інтерфейсу користувача та відображення метеорологічних параметрів.



Рис. 1.1. Багатофункціональний цифровий модуль Hanksares, що поєднує в собі систему точного часу з будильником та комплексні датчики для моніторингу атмосферного тиску, температури й відносної вологості повітря



Рис. 1.2. Комбінована цифрова метеостанція з інтегрованими модулями барометрії, термометрії та гігрометрії. Пристрій забезпечує одночасну візуалізацію поточного часу та ключових параметрів стану навколишнього середовища на єдиному дисплеї

Таблиця 1.3. Технічні характеристики комбінованого цифрового пристрою

Параметр	Значення/Опис
Діапазон вимірювання t	0°C...+50°C (32...122°F)
Діапазон вологості	10%...99% RH
Похибка (темп./волог.)	±1°C (±2°F)/±5% RH
Роздільна здатність	0,1°C та 1% RH
Додаткові функції	Запис MAX/MIN, індикатор комфорту, вибір одиниць (°C/°F)
Габаритні розміри	110×100×20 мм
Елемент живлення	1,5 В (тип AAA)



Рис. 1.3. Сучасні цифрові модулі з інтегрованими функціями годинника та метеомоніторингу (температура, вологість)

Пристрій забезпечує моніторинг атмосферного тиску з візуалізацією динаміки його змін, а також надає дані про поточну температуру та час (Рис.1.3). Система формування барометричного прогнозу відображає гістограму тиску, що базується на поточному значенні та архівованих даних за останні 1, 3, 6 та 12 годин. Часові параметри виводяться у двох стандартах: 24-годинному та 12-годинному (з індикацією А.М./Р.М.). На основі отриманих метеоданих пристрій генерує короткостроковий прогноз на наступні 12 годин, використовуючи графічні символи: “ясно”, “мінлива хмарність”, “хмарно” та “опаді”. Пристрій має наступні експлуатаційні параметри. Робочий діапазон вимірювання атмосферного тиску становить від 800 до 1050 мБар із роздільною здатністю 1 мБар. Похибка вимірювання тиску в основному робочому діапазоні (900–1050 мБар) не перевищує ± 3 мБар. Термометричний модуль забезпечує контроль температури в межах від -9.5°C до $+70^{\circ}\text{C}$ з точністю $\pm 1.0^{\circ}\text{C}$ та кроком вимірювання 0.5°C . Візуалізація даних здійснюється за допомогою рідкокристалічного дисплея (LCD). Система підтримує подвійний формат відображення часу (12/24 години). Автономність приладу забезпечується одним елементом живлення типу ААА, а компактні габарити (приблизно 100x100x57 мм) дозволяють використовувати його як портативний засіб.

Аналізована модель WR-0100-020 китайського виробництва (фірма АТЕ) призначена для комплексного моніторингу кліматичних умов.

Внутрішній температурний датчик працює в межах від -10 до +60°C, тоді як виносний сенсор дозволяє проводити вимірювання в ширшому діапазоні: від -40 до +60°C. Похибка термометра становить $\pm 1^\circ\text{C}$. Пристрій також контролює відносну вологість повітря (20–99% RH) із точністю $\pm 5\%$. Важливою функціональною особливістю є можливість ручного калібрування показників та система тривожного сповіщення (аларм) при виході значень за встановлені межі. Живлення системи здійснюється від двох лужних елементів типу AA (1.5 В), а корпус має габарити 96x120x30 мм.

РОЗДІЛ 2

ЗАСОБИ РОЗРОБКИ АПАРАТНОГО ТА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ЦИФРОВОГО ПРИСТРОЮ ДЛЯ ПЛАНУВАННЯ ТРИВАЛИХ ЧЕРГУВАНЬ ТА ОЦІНКИ СТАНУ ОСОБОВОГО СКЛАДУ

2.1. Система автоматизованого проектування і моделювання програмованих пристроїв Proteus VSM

Програмний пакет Proteus VSM від розробника Labcenter Electronics є потужним інструментом автоматизованого проектування. Система дозволяє ефективно проектувати складні електронні модулі, включаючи цифрові пристрої на базі мікроконтролерів різних типів.

Proteus забезпечує високу точність схмотехнічного моделювання завдяки використанню детальних віртуальних аналогів реальних радіодеталей. Характерною особливістю пакета є інтегрована можливість верифікації прошивок для різних архітектур – від базових мікропроцесорів до спеціалізованих контролерів, що дозволяє проводити повну налагодження системи до створення фізичного прототипу.

Функціонал Proteus охоплює повний цикл створення електронного виробу: від креслення принципової електричної схеми до комплексної симуляції її роботи за допомогою контрольно-вимірювальних приладів. Система дозволяє проводити налагодження програмного коду безпосередньо у віртуальному середовищі. Окремою перевагою пакета є інтегрований модуль для проектування друкованих плат, що підтримує функцію об'ємної (3D) візуалізації готового пристрою.

Універсальність Proteus зумовлена наявністю потужної бібліотеки, яка включає понад 6000 віртуальних моделей електронних компонентів. Система забезпечує коректне моделювання роботи широкого спектра програмованих чипів, серед яких архітектури AVR, PIC, ARM7, а також серії 8051 та HC11. Завдяки наявності детальних довідкових даних для кожного елемента,

розробник може ефективно проектувати вузли на базі найбільш розповсюджених у промисловості та навчанні мікропроцесорів.

Однією з переваг Proteus VSM є можливість спільного відлагодження програмного та апаратного забезпечення для систем, побудованих на базі різних типів контролерів. Програма забезпечує достовірну імітацію роботи складних електронних пристроїв. Слід, проте, зважати на теоретичні обмеження віртуальної симуляції, яка не може врахувати всі фізичні чинники реального середовища. Проте для верифікації логіки роботи мікропроцесорних систем цей інструментарій є цілком ефективним і достатнім. Програмний комплекс дозволяє гнучко налаштовувати робоче середовище завдяки можливості додавання нових компонентів до базової бібліотеки. Підтримка промислового стандарту SPICE значно розширює межі моделювання, оскільки більшість сучасних виробників електронних пристроїв надають готові моделі своїх продуктів. Це вирішує проблему відсутності специфічних деталей: достатньо імпортувати завантажену модель у відповідну графічну оболонку (корпус) всередині середовища Proteus.

Структурно пакет Proteus поділений на два основні робочі модулі:

- ISIS: слугує ядром для схемотехнічного проектування та відлагодження пристрою. Після успішного тестування алгоритмів у цьому модулі, проект експортується в середовище розробки друкованих плат.
- ARES: відповідає за конструкторський етап розробки. Завдяки підтримці автоматичного трасування (ELECTRA) та інтелектуальній розстановці елементів згідно з існуючою схемою, модуль забезпечує високу швидкість проектування плат будь-якої складності.

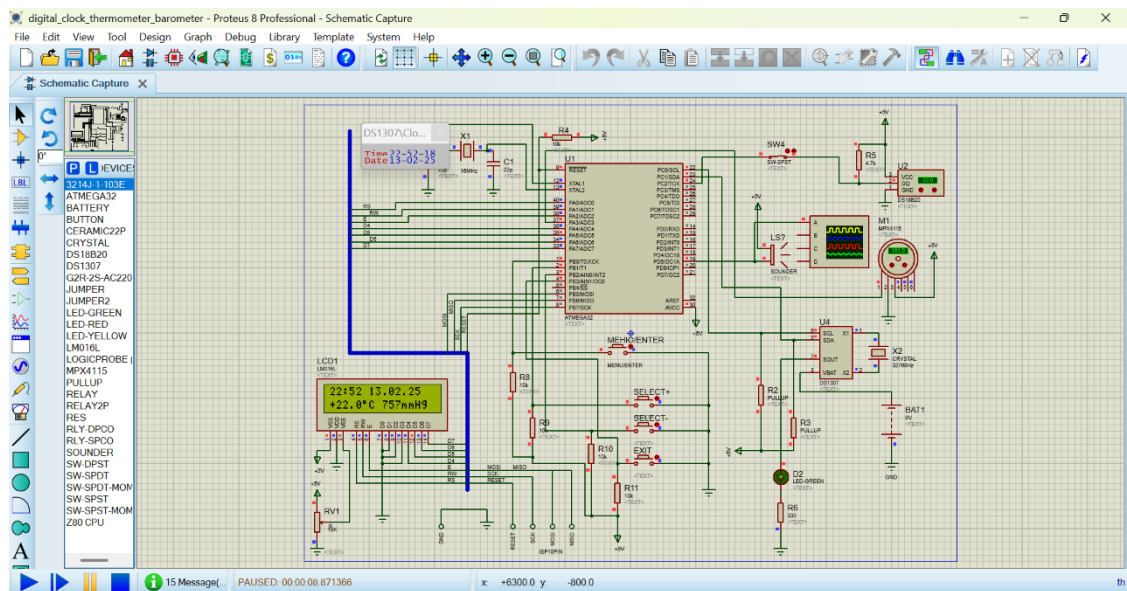


Рис. 2.1. Робочий простір середовища ISIS для проектування та симуляції схем

До переліку ключових переваг Proteus слід віднести можливість сполучення симуляції з реальними інтерфейсами введення-виведення:

1. Модуль COMPIM: виконує роль мосту для підключення віртуального мікроконтролера до фізичного COM-порту, що є критично важливим для перевірки RS-232 сумісних пристроїв.
2. Інструмент USBCONN: надає можливість інтеграції розробленої моделі з USB-портом хост-машини, забезпечуючи реалістичну перевірку обміну даними.

Використання Proteus VSM дозволяє здійснювати повний цикл розробки, налагодження та верифікації вбудованого програмного забезпечення на етапі, що передуює створенню фізичного прототипу. Система демонструє високу сумісність із провідними середовищами розробки (IDE), що підтримують різні мікропроцесорні архітектури:

- CodeVisionAVR та WinAVR (орієнтовані на сімейство AVR);
- IAR (універсальне рішення для широкого спектра контролерів);
- ICC (підтримка архітектур AVR, msp430 та ARM7);
- Keil (оптимізовано для роботи з ядрами 8051 та ARM);
- HiTECH (спеціалізоване ПЗ для мікроконтролерів PIC та 8051).

2.2. Програмне забезпечення WinAVR та мова C як інструментарій проектування систем на базі контролерів AVR

При написанні програмного коду в середовищі WinAVR використовується стандартний набір базових типів та зарезервованих слів мови ANSI C, доповнений специфічними розширеннями для роботи з архітектурою AVR (Табл.2.1, 2.2).

Таблиця 2.1. Характеристики базових типів даних мови C для мікроконтролерів AVR

Категорія типу	Назва в мові C	Розмір (біт)	Діапазон значень
Логічний	bit	1	0 ... 1
8-бітні (байтові)	char, signed char unsigned char	8	-128 ... 127 0 ... 255
16-бітні (слова)	int, short int unsigned int	16	-32 768 ... 32 767 0 ... 65 535
32-бітні (подвійні)	long int unsigned long int	32	$\pm 2,147 \cdot 10^9$ 0 ... 4 294 967 295
З плаваючою комою	float, double	32	$\pm 1,175 \cdot 10^{-38} \dots \pm 3,402 \cdot 10^{38}$

Таблиця 2.2. Класифікація ключових слів мови C в інструментарії WinAVR

Категорія	Зарезервовані слова
Типи даних	int, char, float, double, long, short, void, bit, signed, unsigned
Керуючі конструкції	if, else, switch, case, default, for, while, do, break, continue, goto, return
Специфікатори пам'яті	static, extern, register, volatile, const, auto, inline, flash, eeprom
Структури та типи	struct, union, enum, typedef, sizeof
Специфічні для МК	interrupt, sfrw, funcused

WinAVR – це універсальний пакет інструментів з відкритим вихідним кодом, призначений для розробки ПЗ для контролерів компанії Atmel. Завдяки інтеграції компілятора GNU GCC, середовище підтримує роботу з мовами C і C++, забезпечуючи ефективне проектування для платформ AVR/AVR32. Кожен компонент цього пакета виконує специфічне завдання, що в сукупності дозволяє реалізувати повний цикл створення надійного програмного забезпечення для вбудованих систем.

Основними компонентами інструментарію WinAVR є:

- Середовище редагування: Programmers Notepad – спеціалізований текстовий редактор, адаптований для написання та структурування вихідного коду.
- Програмний базис: AVR-LibC – фундаментальна бібліотека стандартних функцій, що забезпечує взаємодію з апаратними ресурсами контролерів AVR.
- Засоби прошивки: AVRDUDE – високоефективна утиліта для програмування пам'яті мікросхем, що реалізує обмін даними через SPI-інтерфейс.
- Інструменти налагодження: GNU Debugger (GDB) та його графічна оболонка Insight, що дозволяють здійснювати покроковий аналіз виконання програми.
- Верифікація та симуляція: Simulavr – програмний симулятор архітектури AVR, інтегрований із відладчиком GDB для віртуального тестування коду.
- Обробка бінарних даних: SRecord – багатофункціональний набір утиліт для маніпуляції завантажувальними файлами та образами пам'яті EPROM у різних форматах.

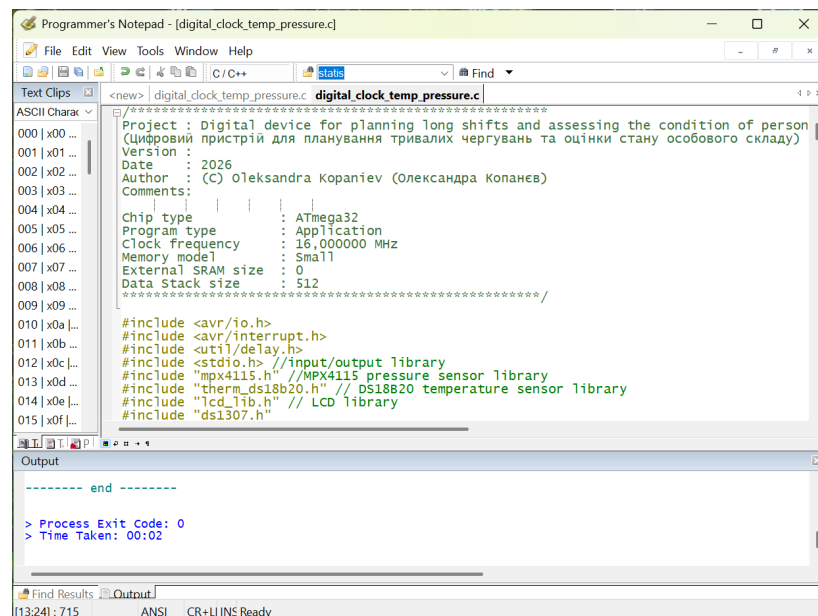


Рис. 2.2. Робоче середовище WinAVR із відкритим текстовим редактором Programmers Notepad

WinAVR характеризується низьким порогом входження завдяки наявності розгорнутої технічної документації, що полегшує процес освоєння інструментарію. Компілятор генерує оптимізований та компактний машинний код, суворо дотримуючись стандартів синтаксису класичної мови C. Проте суттєвим обмеженням середовища є відсутність вбудованих засобів автоматичної конфігурації периферійних модулів AVR та візуальних інструментів для налаштування інтерфейсів взаємодії, що вимагає від розробника ручного програмування регістрів.

Інструментарій WinAVR має можливість повної інтеграції з фірмовим середовищем Atmel Studio, що створює потужну екосистему для проектування та відлагодження програмних рішень. Таке поєднання забезпечує повну підтримку апаратних можливостей мікроконтролерів Atmel. Програмний пакет розповсюджується на умовах вільної ліцензії GNU, що дозволяє користувачам не лише безкоштовно експлуатувати ПЗ, а й модифікувати його вихідний код відповідно до власних потреб. Крім того, наявність портативної версії (portable) дозволяє використовувати інструментарій без попереднього встановлення в операційну систему.

При інсталяції WinAVR критично важливо дотримуватися правила іменування директорій: шлях до папки не повинен містити кириличних символів або пробілів, що забезпечує коректну роботу компілятора. Після завершення встановлення користувачеві стає доступним набір ярликів для запуску різних модулів системи, серед яких ключовим є текстовий редактор Programmers Notepad. Докладна технічна документація та приклади коду інтегровані в структуру пакета за шляхом `\doc\avr-libc`. Програмний комплекс має англomовний інтерфейс і демонструє стабільну роботу в усіх 32-бітних версіях ОС Windows, а також має відповідну реалізацію для платформ на базі Linux.

У Табл.2.3 наведено технічні характеристики та склад програмного інструментарію WinAVR.

*Таблиця 2.3. Технічні характеристики та склад програмного
інструментарію WinAVR*

Категорія	Характеристики та компоненти
Загальні параметри	Підтримка сімейства ОС Windows; мова інтерфейсу – англійська; ліцензія – Open Source (OSI).
Мовне забезпечення	Підтримка мов низького (Assembler) та високого рівнів (C, C+ +).
Компіляція та збірка	GNU GCC: безкоштовний C-компілятор; GNU Binutils: набір утиліт для асемблювання та лінування; MFile: засіб генерації керуючих Make-файлів.
Засоби розробки (IDE)	Programmers Notepad: основний редактор коду; повна інтеграція з середовищем Atmel Studio.
Налагодження та симуляція	GDB/Insight: консольний та графічний відладчики; simulavr: програмний симулятор МК; AVaRICE: інтерфейс для роботи з JTAG ICE.
Програмування та сервіс	avrdude/avrdude-gui: засоби прошивки МК; SRecord: інструментарій для маніпуляції файлами образів пам'яті (EPROM); avr-libc: стандартна бібліотека функцій.

РОЗДІЛ 3

АПАРATНЕ ЗАБЕЗПЕЧЕННЯ ЦИФРОВОГО ПРИСТРОЮ ДЛЯ ПЛАНУВАННЯ ТРИВАЛИХ ЧЕРГУВАНЬ ТА ОЦІНКИ СТАНУ ОСОБОВОГО СКЛАДУ

3.1. Обґрунтування вибору апаратних засобів для реалізації цифрового пристрою для планування тривалих чергувань та оцінки стану особового складу

Для реалізації функціональних можливостей проектного пристрою проведено порівняльний аналіз сучасної елементної бази. Вибір мікроконтролера та сенсорної периферії базується на специфіці експлуатації пристрою підрозділами поліції в екстремальних умовах, де ключову роль відіграють автономність живлення та оперативність обробки даних від метеорологічних модулів.

При проектуванні багатофункціонального цифрового пристрою для потреб особового складу поліції за основу було взято мікроконтролер ATmega32, який забезпечує гнучке керування всіма вузлами системи. Для автоматизації планування графіків чергувань інтегровано модуль реального часу DS1307. Метеорологічний функціонал (термометр та барометр) реалізовано на базі перевірених рішень – цифрового сенсора DS18B20 та аналогового датчика тиску MPX4115. Взаємодія з користувачем здійснюється через стандартний рідкокристалічний дисплей 16x2 та ергономічну клавіатуру з чотирьох тактових кнопок.

Огляд мікроконтролера Atmel ATmega32. Сучасний ринок мікропроцесорної техніки представлений тисячами модифікацій контролерів від десятків світових виробників. Втім, для реалізації даного проекту було обрано продукцію компанії Atmel. Цей вибір зумовлений оптимальним балансом між вартістю компонентів, обчислювальною потужністю та низьким рівнем енергоспоживання. Додатковим аргументом на користь архітектури

AVR стала розвинена інфраструктура засобів розробки та широка номенклатура периферійних модулів, що значно спрощує процес проектування вбудованих систем.

Сімейство 8-бітних мікроконтролерів AVR від компанії Atmel базується на прогресивній RISC-архітектурі, що дозволяє ефективно вирішувати широкий спектр завдань у сфері вбудованих систем. Порівняно з аналогічними рішеннями, дані МК вирізняються підвищеною продуктивністю та функціональною універсальністю. Важливою експлуатаційною перевагою є підтримка технології внутрішньосхемного програмування (ISP), що дозволяє багаторазово (до 1000 циклів) оновлювати програмне забезпечення без демонтажу мікросхеми з друкованої плати.

Архітектурна ієрархія мікроконтролерів AVR включає три основні категорії:

- TinyAVR: бюджетні рішення в ультракомпактних корпусах (від 6–8 виводів), оптимізовані для простих завдань із жорсткими обмеженнями щодо габаритів та вартості.
- ClassicAVR: базова лінія контролерів, що забезпечує продуктивність до 16 MIPS. Дана серія характеризується об'ємом Flash-пам'яті до 8 КБ, EEPROM до 512 байт та оперативною пам'яттю SRAM до 512 байт.
- MegaAVR: найбільш потужна підродина, призначена для реалізації складних алгоритмів. Вона вирізняється розширеним об'ємом Flash-пам'яті (до 256 КБ), наявністю апаратного помножувача, 10-розрядного багатоканального АЦП та значним обсягом SRAM (до 8-16 КБ у старших моделях).

Ключовою особливістю архітектури є кодова сумісність: розроблене програмне забезпечення легко масштабується при переході від молодших моделей до більш продуктивних у межах усього сімейства.

ATmega32 та його модифікація з низьким енергоспоживанням ATmega32L представляють собою високопродуктивні 8-бітні мікроконтролери сімейства AVR. Ключовою особливістю цих чіпів є

наявність 32 КБ внутрішньосхемно програмованої Flash-пам'яті, що дозволяє оперативно оновлювати програмний код. Поєднання архітектури RISC із низьким рівнем споживання струму робить ці мікроконтролери оптимальним рішенням для автономних пристроїв, де критично важливою є тривала робота від акумулятора при збереженні високої швидкодії.

Технічні можливості та архітектурні особливості ATmega32:

- Обчислювальне ядро та продуктивність: Прогресивна RISC-архітектура, що включає 131 інструкцію (більшість виконується за один такт). Швидкодія до 16 MIPS при тактовій частоті 16 МГц. Наявність 32 робочих регістрів загального призначення та апаратного помножувача (2 такти).
- Пам'ять та ресурси: 32 КБ Flash-пам'яті програм (до 10 000 циклів перезапису) з можливістю самопрограмування (Bootloader). 1024 байт енергонезалежної пам'яті EEPROM для зберігання констант та налаштувань. 2 КБ внутрішньої оперативної пам'яті SRAM.
- Периферійні модулі: Три таймери/лічильники (два 8-бітних та один 16-бітний) з режимами порівняння та ШІМ. 8-канальний 10-розрядний АЦП та вбудований аналоговий компаратор. Комунікаційні інтерфейси: USART, SPI та I2C (Two-wire Interface). Програмований сторожовий таймер (Watchdog) з власним осцилятором.
- Системні функції та енергоспоживання: Шість режимів енергозбереження (Idle, Power-down та інші). Підтримка інтерфейсу JTAG для відлагодження та граничного сканування. Діапазон напруги живлення: від 2,7 В до 5,5 В (для версії L).

На Рис. 3.1 представлено конфігурацію виводів мікроконтролера ATmega32 у двох найбільш поширених типах корпусів. Для макетування та розробки дослідного зразка пристрою обрано 40-вивідний корпус PDIP (Plastic Dual In-line Package), що дозволяє зручно встановлювати мікросхему в сокет або безпаячну макетну плату. У свою чергу, 44-вивідний корпус TQFP (Thin

Quad Flat Pack) є оптимальним для фінальної серійної збірки на друковану плату методом поверхневого монтажу (SMD).

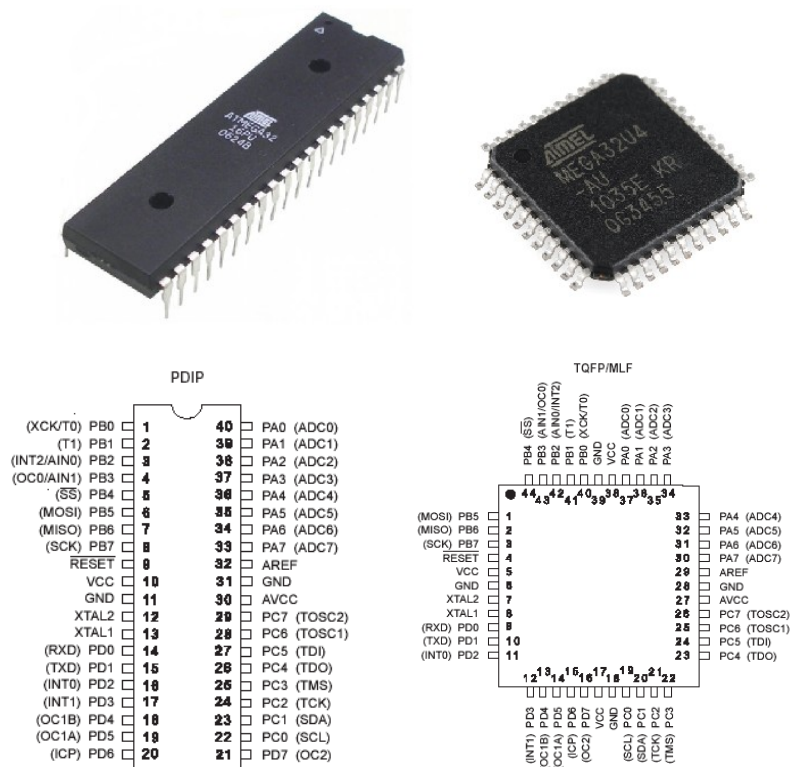


Рис. 3.1. Схема розташування та функціональне призначення виводів мікроконтролера ATmega32 у корпусах PDIP та TQFP

В основу функціонування ядра AVR покладено поєднання розширеної системи команд із блоком із 32-х 8-розрядних регістрів загального призначення (РЗП). Унікальність цієї архітектури полягає в тому, що всі регістри безпосередньо інтегровані з арифметико-логічним пристроєм (АЛП). Це дозволяє зчитувати два незалежних операнди, виконувати операцію та зберігати результат усього за один системний такт. Така організація обчислювального процесу забезпечує продуктивність, що майже вдесятеро перевищує показники традиційних CISC-мікроконтролерів при однаковій тактовій частоті.

Мікроконтролер ATmega32/L інтегрує в одному кристалі всі необхідні ресурси для побудови складних цифрових систем. Наявність 32 КБ Flash-пам'яті з можливістю самопрограмування та JTAG-інтерфейсу дозволяє ефективно виконувати налагодження коду безпосередньо в системі. Периферійний набір включає 8-канальний 10-бітний АЦП, що є критично

важливим для обробки сигналів від аналогових датчиків (наприклад, барометра MPX4115). Система живлення та шість режимів енергозбереження забезпечують гнучкість у керуванні споживанням, де режим Idle дозволяє значно знизити струм споживання шляхом зупинки процесорного ядра при збереженні активності таймерів та інтерфейсів.

Наведена блок-схема (Рис. 3.2) ілюструє Гарвардську архітектуру мікроконтролера, що характеризується роздільними шинами для пам'яті програм та даних. Це дозволяє центральному процесору (CPU) одночасно звертатися до Flash-пам'яті за наступною інструкцією та зчитувати/записувати дані в SRAM, що суттєво підвищує загальну швидкодію системи. Центральне місце займає 8-бітне RISC-ядро, інтегроване з 32 робочими регістрами та арифметико-логічним пристроєм (ALU).

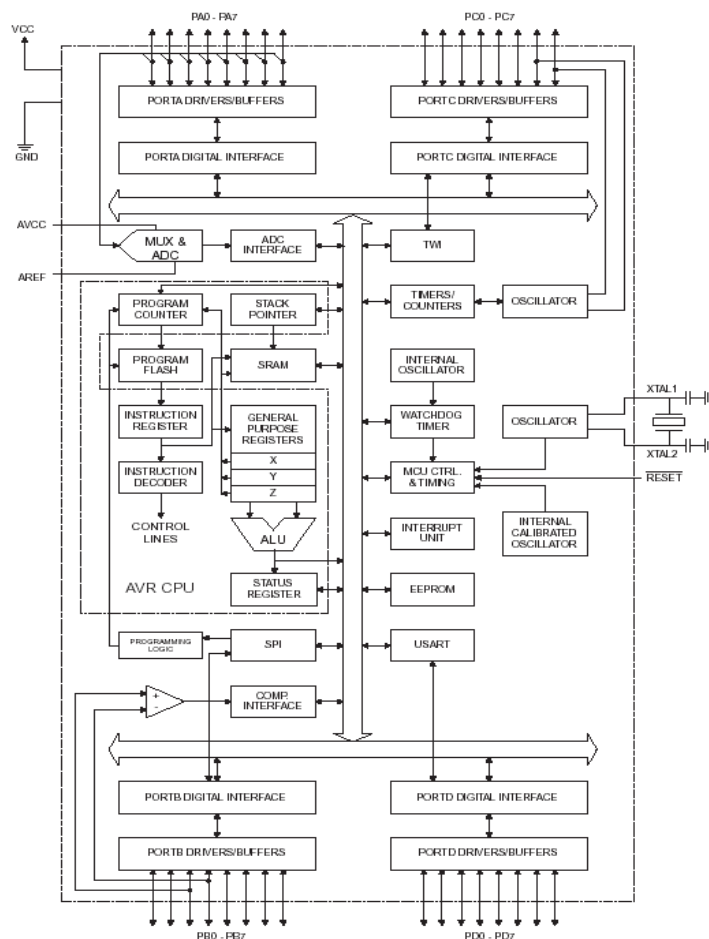


Рис. 3.2. Внутрішня структурна схема мікроконтролера ATmega32

У режимі очікування (Idle mode) припиняється тактування процесорного ядра, проте зберігається повна працездатність основних периферійних

модулів: USART, SPI, таймерів-лічильників, систем переривань та АЦП. Окрему увагу в архітектурі ATmega32 приділено режиму придушення шумів АЦП (ADC Noise Reduction). У цьому стані функціонують виключно аналого-цифровий перетворювач та асинхронний таймер, що мінімізує вплив цифрових завад від роботи логічних схем МК на точність квантування сигналу.

Режим Power-down забезпечує мінімально можливе споживання струму пристроєм. У цьому стані “заморожується” робота генератора тактових сигналів та зупиняються всі периферійні модулі, проте архітектура МК дозволяє зберегти вміст усіх внутрішніх регістрів. Такий підхід дає змогу системі миттєво повернутися до виконання алгоритму після отримання зовнішньої команди або переривання, що є критично важливим для автономних поліцейських пристроїв, які працюють у польових умовах.

Для забезпечення високої автономності пристрою в екстремальних умовах використано адаптивні режими роботи ATmega32. Завдяки здатності до швидкої реініціалізації з Idle-режиму (зокрема, за зовнішнім перериванням), мікроконтролер ефективно підлаштовується під зовнішні чинники, споживаючи енергію лише під час активної фази обчислень. У випадках, коли необхідна безперервна робота годинника реального часу при збереженні високої готовності системи, застосовується розширений режим очікування, де активними залишаються як основний, так і допоміжний тактові осцилятори.

Мікроконтролери даної серії виготовлені за запатентованою технологією енергонезалежної пам'яті високої щільності компанії Atmel. Наявність вбудованої ISP Flash-пам'яті забезпечує можливість внутрішньосхемного перепрограмування через стандартний послідовний інтерфейс SPI. Окрім використання зовнішніх програматорів, архітектура підтримує оновлення коду за допомогою завантажувальної програми (Bootloader), що функціонує безпосередньо в ядрі AVR. Поєднання прогресивної 8-бітної RISC-архітектури з гнучкими засобами керування

пам'яттю робить ATmega32 високоефективним та економічно вигідним рішенням для побудови сучасних вбудованих систем управління.

Аналого-цифровий перетворювач МК ATmega32. Стандартні цифрові інтерфейси ATmega32 розпізнають лише дискретні стани напруги (низький та високий рівні). Зокрема, логічний нуль фіксується в діапазоні 0...1,5В, а логічна одиниця – від 1,9В до 5В. Для розв'язання задач, що потребують вимірювання плавно змінних напруг (у всьому робочому діапазоні), розробники інтегрували в структуру МК спеціалізований модуль – аналого-цифровий перетворювач. Це дозволяє перетворювати вхідний аналоговий сигнал у цифровий код для подальшої програмної обробки.

На представленій схемі Рис.3.3 зображено спеціалізовані пінні порту PA (ADC0-ADC7), які відповідають за приймання аналогових сигналів. Окрім восьми вхідних каналів, модуль АЦП потребує підключення опорної напруги через вивід AREF та окремого живлення аналогової частини через вивід AVCC із заземленням AGND для мінімізації цифрових завад.

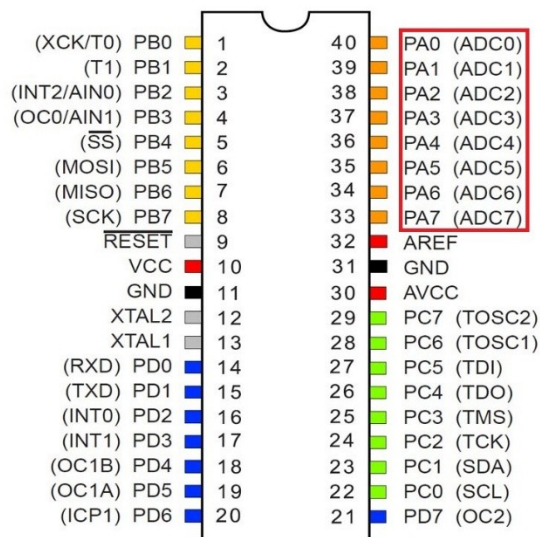


Рис. 3.3. Схема розташування та призначення виводів АЦП мікроконтролера ATmega32

Аналогові сигнали в мікроконтролері ATmega32 можуть бути подані на будь-який із виводів порту А (PORTA). Конфігурування та керування роботою інтегрованого модуля АЦП здійснюється за допомогою набору спеціалізованих регістрів:

- ADCSRA – основний регістр для налаштування режимів роботи, стану та керування перетворювачем.
- ADMUX – керуючий регістр, що відповідає за вибір вхідного каналу (мультиплексування) та джерела опорної напруги.
- SFIOR – регістр додаткових функцій, що використовується для налаштування тригерів автозапуску.
- ADCH/ADCL – пара регістрів, що зберігають відповідно старший та молодший байти отриманого цифрового значення.

Bit	7	6	5	4	3	2	1	0	
	REFS1	REFS0	ADLAR	MUX4	MUX3	MUX2	MUX1	MUX0	ADMUX
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

Таблиця 3.1. Конфігурація джерел опорної напруги (V_{ref}) АЦП

REFS1	REFS0	Опис обраного джерела опорної напруги
0	0	Зовнішній сигнал: використовується напруга, подана на вивід AREF; внутрішній стабілізатор вимкнений.
0	1	Напруга живлення (AVCC): опорним сигналом є живлення аналогової частини (потрібен фільтруючий конденсатор на AREF).
1	0	Не використовується (Зарезервовано).
1	1	Внутрішній еталон: джерело 2,56 В (рекомендується підключення зовнішньої ємності до виводу AREF).

Діапазон функціонування АЦП безпосередньо залежить від обраного джерела опорної напруги (U_{ref}). Якщо амплітуда вхідного однополярного сигналу перевищує встановлений поріг U_{ref} , вихідний результат перетворення досягне свого максимуму – 0x3FF (що відповідає 1023 у десятковій системі). В архітектурі ATmega32 передбачено три варіанти формування опорного рівня: використання напруги живлення AVCC, активація прецизійного внутрішнього стабілізатора на 2,56 В або подача еталонного сигналу на фізичний вивід AREF. Зокрема, зв'язок між AVCC та модулем АЦП реалізується за допомогою інтегрованого пасивного ключа.

Внутрішній опорний рівень напруги 2,56 В формується прецизійним джерелом еталонної напруги (U_{ref}), що підключене через інтегрований буферний підсилювач. Оскільки фізичний вивід AREF має пряме сполучення з входом АЦП, для мінімізації імпульсних завад та шумів у колі опорної напруги

рекомендується встановлювати фільтруючий конденсатор між цим виводом та шиною заземлення (GND). Контроль рівня напруги на AREF можливий за допомогою зовнішнього вольтметра з високим входним опором. Важливо враховувати, що внутрішнє джерело опорної напруги є високоомним, тому до нього допускається підключення лише ємнісного навантаження; будь-яке значне резистивне навантаження призведе до похибки вимірювання.

Результат 10-бітного перетворення АЦП зберігається у парі 8-розрядних регістрів ADCH та ADCL. За замовчуванням дані вирівнюються праворуч, займаючи молодші 10 бітів 16-бітного слова. Проте встановлення біта ADLAR у регістрі ADMUX дозволяє активувати лівостороннє вирівнювання. Ключовою особливістю архітектури є механізм захисту цілісності даних: після зчитування молодшого байта (ADCL) доступ модуля АЦП до обох регістрів тимчасово блокується. Це гарантує, що якщо нове перетворення завершиться до моменту читання старшого байта (ADCH), поточні значення не будуть перезаписані. Доступ для оновлення регістрів автоматично відновлюється лише після того, як програма звернеться до ADCH.

Таблиця 3.2. Розподіл бітів результату АЦП (ADLAR=0)

Регістр	Б7	Б6	Б5	Б4	Б3	Б2	Б1	Б0
ADCH	-	-	-	-	-	-	ADC9	ADC8
ADCL	ADC7	ADC6	ADC5	ADC4	ADC3	ADC2	ADC1	ADC0

Біти 2-7 регістру ADCH у цьому режимі не використовуються і при зчитуванні повертають “0”.

При встановленні біта ADLAR у стан логічної одиниці в регістрі ADMUX, мікроконтролер переходить у режим лівостороннього вирівнювання результату АЦП. У цьому випадку вісім старших розрядів (ADC9...ADC2) записуються безпосередньо в регістр ADCH, а два молодші розряди (ADC1 та ADC0) переміщуються у старші біти регістру ADCL.

Таблиця 3.3. Структура результату АЦП при ADLAR=1

Регістр	Б7	Б6	Б5	Б4	Б3	Б2	Б1	Б0
ADCH	ADC9	ADC8	ADC7	ADC6	ADC5	ADC4	ADC3	ADC2
ADCL	ADC1	ADC0	-	-	-	-	-	-

Біти 0-5 регістру ADCL у цьому режимі зарезервовані та повертають нуль при зчитуванні.

Вибір аналогового каналу в ATmega32 здійснюється шляхом запису відповідної комбінації у молодші п'ять бітів регістру ADMUX (MUX0–MUX4). Архітектура дозволяє реалізувати декілька режимів зчитування:

1. Стандартний режим: Опитування одного з восьми каналів (ADC0–ADC7). Це класична схема для більшості сенсорів.
2. Диференціальний режим з підсиленням: Використовується для вимірювання різниці потенціалів між двома входами. Особливістю ATmega32 є наявність програмованого підсилювача з коефіцієнтами 10x та 200x, що дозволяє обробляти слабкі сигнали без зовнішніх операційних підсилювачів.
3. Діагностичний режим: Дозволяє підключити вхід АЦП до внутрішнього еталона (Bandgap reference) або до «землі» для самокалібрування системи. У Табл.3.4. наведено конфігурацію входних каналів мультиплексора АЦП.

Таблиця 3.4. Конфігурація входних каналів мультиплексора АЦП

MUX4...0	Тип входного сигналу	Опис підключення	Коефіцієнт підсилення
00000 – 00111	Одиночний	Входи ADC0 – ADC7 (відносно GND)	1x
01000 – 01101	Диференціальний	Входи ADC0 та ADC1 або ADC2 та ADC3	10x/200x
01110 – 11101	Диференціальний	Комбінації ADC0 – ADC7 із різницеvim підсиленням	1x
11110	Тестовий	Внутрішнє джерело опорної напруги (V_{BG})	-
11111	Тестовий	Спільний вивід (GND)	-

Для керування входним мультиплексором АЦП у мікроконтролері ATmega32 використовується регістр ADMUX. Шляхом маніпуляції бітами MUX4:0 розробник може призначити один із восьми доступних аналогових входів (ADC0 – ADC7) як однополярне джерело сигналу. Крім того, підтримується можливість підключення входу перетворювача до опорної напруги 1,22 В (Bandgap reference) або до загального проводу, що дозволяє програмно контролювати зміщення нуля та точність квантування.

Архітектура АЦП у мікроконтролері ATmega32 дозволяє гнучко конфігурувати інвертуючі та неінвертуючі входи диференціального підсилювача. У разі активації диференціального режиму, підсилювальний каскад обробляє різницю потенціалів між обраною парою аналогових каналів, множачи її на встановлений коефіцієнт підсилення. Отриманий посилений сигнал надалі подається на вхід аналого-цифрового перетворювача. Важливо зауважити, що при переході до однополярного режиму зчитування каскад підсилення виключається із ланцюга обробки сигналу, і вимірювання проводиться безпосередньо.

Bit	7	6	5	4	3	2	1	0	
	ADEN	ADSC	ADATE	ADIF	ADIE	ADPS2	ADPS1	ADPS0	ADCSRA
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

Для коректної роботи АЦП у мікроконтролері ATmega32 частота тактування модуля повинна знаходитися в межах 50 – 200 кГц. Вибір коефіцієнта ділення системної частоти здійснюється за допомогою бітів ADPS2:0 (Табл.3.5).

Таблиця 3.5. Налаштування коефіцієнта ділення частоти АЦП

ADPS2	ADPS1	ADPS0	Коефіцієнт ділення
0	0	0	2
0	0	1	2
0	1	0	4
0	1	1	8
1	0	0	16
1	0	1	32
1	1	0	64
1	1	1	128

Структура модуля АЦП включає спеціалізований дільник частоти (прескалер), джерелом тактування для якого є системний генератор мікроконтролера. Коефіцієнт ділення змінюється дискретно за законом 2^n у діапазоні від 2 до 128. Налаштування необхідного значення реалізується через біти групи ADPS у керуючому регістрі ADCSRA. Робота дільника активується одночасно з увімкненням модуля АЦП шляхом встановлення прапорця ADEN. Прескалер функціонує безперервно, доки біт ADEN перебуває у стані логічної одиниці, і автоматично скидається при деактивації модуля (ADEN=0).

Для досягнення максимальної точності квантування (10 біт) тактова частота на вході схеми послідовного наближення повинна підтримуватися в межах 50 – 200 кГц. У випадках, коли пріоритетом є швидкість оцифрування, а роздільна здатність може бути нижчою за 10 розрядів, допускається підвищення частоти тактування понад 200 кГц. Слід враховувати, що стандартний цикл перетворення в мікроконтролері ATmega32 триває 13 тактів синхронізації модуля АЦП.

Початковий цикл перетворення, що слідує безпосередньо за активацією модуля АЦП (шляхом встановлення біта ADEN у регістрі ADCSRA), триває 25 тактів синхронізації. Така збільшена тривалість зумовлена необхідністю стабілізації та повної ініціалізації внутрішніх аналогових каскадів. Після завершення цього процесу сформований цифровий код записується у регістри даних (ADCL та ADCH), а апаратний прапорець переривання ADIF переходить у стан логічної одиниці, сигналізуючи про готовність даних.

У режимі одиничного перетворення (Single Conversion) завершення циклу оцифрування супроводжується автоматичним скиданням біта ADSC у стан логічного нуля. Для ініціалізації наступного вимірювання необхідно повторно встановити цей біт програмним шляхом; при цьому старт перетворення синхронізується з першим наростаючим фронтом тактового сигналу модуля АЦП. Натомість у режимі автоматичного перезапуску (Free Running) новий цикл квантування ініціюється негайно після закінчення попереднього, а прапорець ADSC постійно перебуває у стані логічної одиниці, забезпечуючи безперервний процес збору даних.

Ініціалізація одиничного циклу оцифрування здійснюється шляхом запису логічної одиниці у біт ADSC (ADC Start Conversion). Цей біт виконує роль індикатора активності: він зберігає високий рівень протягом усього часу роботи схеми послідовного наближення і автоматично скидається апаратно після фіксації результату. Важливою особливістю є захист поточного циклу: якщо програмно ініційовано зміну вхідного каналу під час активного

перетворення, АЦП завершить поточну ітерацію і лише після цього переключить мультиплексор.

Режим безперервного циклу (Free Running) активується встановленням відповідного біта (ADFR/ADATE). У цьому стані АЦП постійно оновлює регістри даних. Перший запуск у циклічному режимі також потребує встановлення біта ADSC, після чого подальші перетворення відбуваються автоматично, не залежачи від стану прапорця переривання ADIF.

Активація біта ADATE (ADC Auto Trigger Enable) переводить аналого-цифровий перетворювач у режим автоматичного запуску. У цій конфігурації ініціалізація кожного нового циклу оцифрування відбувається за позитивним фронтом вибраного тригерного сигналу. Конкретне джерело, що слугуватиме сигналом для старту (наприклад, переповнення таймера, зовнішнє переривання або завершення попереднього перетворення), визначається комбінацією бітів ADTS у регістрі спеціальних функцій SFIOR.

Прапорець переривання ADIF (ADC Interrupt Flag) автоматично переходить у стан логічної одиниці після завершення циклу квантування та оновлення вмісту регістрів результату (ADCH:ADCL). За умови активації дозволу переривань від АЦП (біт ADIE) та глобального дозволу переривань (біт I у регістрі SREG), система ініціює виклик відповідної підпрограми обробки. Апаратне скидання прапорця ADIF відбувається в момент переходу процесора на вектор переривання. Програмно цей прапорець можна очистити шляхом запису в нього логічної одиниці. Важливо враховувати, що використання команд типу «читання-модифікація-запис» (зокрема інструкцій SBI та CBI) щодо регістру ADCSRA може призвести до ненавмисного скидання прапорця та втрати запиту на переривання.

Біт ADIE (ADC Interrupt Enable) відповідає за програмний дозвіл генерації переривання після завершення циклу роботи АЦП. Для коректної обробки запиту необхідно, щоб цей біт був встановлений у стан логічної одиниці одночасно з глобальним прапорцем дозволу переривань (I) у регістрі статусу SREG. У разі виконання обох умов, мікроконтролер автоматично

переходить до виконання відповідного вектора переривання одразу після фіксації результату в регістрах даних.

Регістр SFIOR (Special Function IO Register) містить біти керування спеціальними функціями мікроконтролера, зокрема конфігурацію тригерів АЦП. Структура регістру SFIOR:

Bit	7	6	5	4	3	2	1	0	
	ADTS2	ADTS1	ADTS0	–	ACME	PUD	PSR2	PSR10	SFIOR
Read/Write	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

Таблиця 3.6. Вибір джерела автоматичного запуску АЦП

ADTS2	ADTS1	ADTS0	Джерело сигналу запуску (Trigger Source)
0	0	0	Режим вільного ходу (Free Running mode)
0	0	1	Аналоговий компаратор
0	1	0	Зовнішнє переривання INT0
0	1	1	Співпадіння Таймера/Лічильника 0 (Compare Match A)
1	0	0	Переповнення Таймера/Лічильника 0 (Overflow)
1	0	1	Співпадіння Таймера/Лічильника 1 (Compare Match B)
1	1	0	Переповнення Таймера/Лічильника 1 (Overflow)
1	1	1	Подія захоплення Таймера/Лічильника 1 (Capture Event)

Функціонування бітів ADTS у регістрі SFIOR безпосередньо залежить від стану біта ADATE у регістрі ADCSRA. Якщо ADATE встановлено у логічну одиницю, комбінація бітів ADTS визначає джерело сигналу для автоматичного запуску перетворення. У протилежному випадку (ADATE=0) стан цих бітів ігнорується системою. Ініціалізація циклу оцифрування відбувається за позитивним фронтом (наростанням) обраного сигналу тригера. Важливо враховувати, що перехід сигналу від джерела запуску зі стану “0” у стан “1” генерує керуючий фронт, який при активному модулі АЦП (ADEN=1) призводить до старту перетворення. Проте перемикання у режим вільного ходу (Free Running mode, код ADTS[2:0]=0) не розцінюється системою як подія запуску, навіть якщо прапорець переривання ADIF уже встановлений.

Годинник реального часу DS1307 з послідовним інтерфейсом I2C. DS1307 – це енергоефективна мікросхема годинника-календаря з підтримкою послідовного інтерфейсу. Пристрій використовує двійково-десятковий формат представлення даних і має у своєму розпорядженні 56 байтів

енергонезалежної статичної пам'яті (SRAM) для зберігання користувацьких налаштувань.

Обмін даними та адресними сигналами здійснюється в послідовному режимі за допомогою двопровідної двоспрямованої шини. Система забезпечує точний відлік часових параметрів: від секунд до року включно. Функціонал пристрою передбачає автоматичне визначення тривалості місяця (якщо в ньому менше 31 дня), а також коректну роботу з високосними роками.

Функціонал пристрою дозволяє відображати час у форматах 12 або 24 годин, причому у 12-годинному режимі передбачено маркування AM/PM. Мікросхема DS1307 оснащена інтегрованим контролером напруги, який у разі зникнення основного живлення миттєво переводить систему на роботу від резервної батареї.

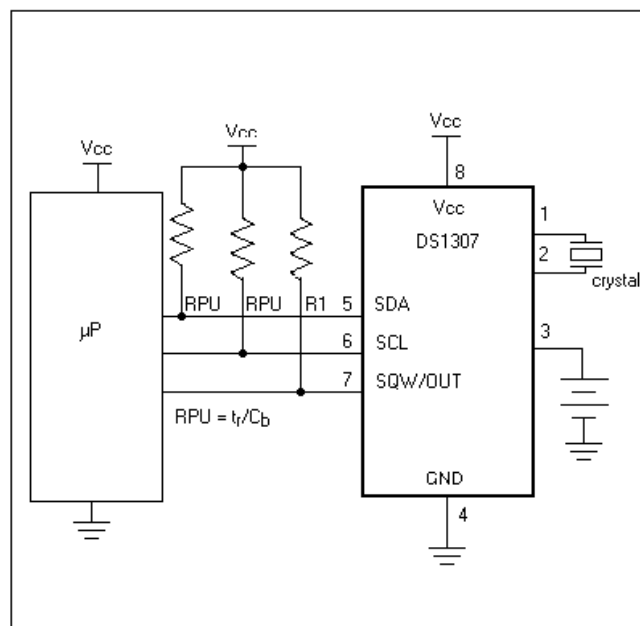


Рис. 3.4. Базова схема підключення модуля DS1307

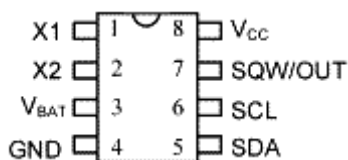
У розробленому пристрої використано мікросхему DS1307, яка виконує роль годинника реального часу. Вона забезпечує безперервну фіксацію часових координат (секунди, хвилини, години) та повної дати з урахуванням тривалості місяців та високосних років (корекція діє до 2100 року). Особливістю модуля є наявність 56 байтів SRAM, де можна зберігати критично важливу інформацію користувача за підтримки енергії від батареї. Зв'язок із контролером реалізовано через послідовний двопровідний

інтерфейс. Пристрій автоматично детектує перебої в основній лінії живлення, перемикаючись на резервне джерело зі споживанням менше 500 нА.

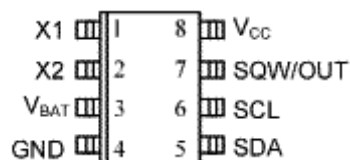
У Табл.3.7 та Рис.3.5 наведено основні модифікації мікросхеми залежно від типу корпусу та призначення.

Таблиця 3.7. Класифікація за типом корпусу та призначенням мікросхеми DS1307

Маркування	Тип корпусу	Розмір (mil/мм)	Робочий діапазон
DS1307	DIP (8 pin)	300/7,62	Стандартний
DS1307Z	SOIC (8 pin)	150/3,81	Стандартний
DS1307N	DIP (8 pin)	300/7,62	Промисловий
DS1307ZN	SOIC (8 pin)	150/3,81	Промисловий



а)



б)

Рис. 3.5. Конфігурація та розташування виводів мікросхеми DS1307: а) у 8-контактному корпусі DIP (шириною 300 mil/7,62 мм); б) у 8-контактному корпусі SOIC (шириною 150 mil/3,81 мм)

Функціональне призначення виводів мікросхеми DS1307 визначено наступним чином:

- VCC – вхід для підключення основної напруги живлення системи.
- X1, X2 – контакти, призначені для приєднання зовнішнього кварцового резонатора з частотою 32,768 кГц, що забезпечує тактування внутрішнього лічильника.
- VBAT – вхідний канал для підключення резервного джерела живлення (літієвого елемента на 3 В), що гарантує збереження ходу годинника при відключенні основної мережі.
- GND – спільний вивід (“земля”), що є від’ємним полюсом живлення.
- SDA (Serial Data) — лінія послідовної передачі даних, через яку відбувається обмін інформацією з мікроконтролером.

- SCL (Serial Clock) — вхід для подачі послідовних синхроімпульсів, що керують швидкістю передачі даних по шині.
- SQW/OUT — програмований вихід, що генерує прямокутні імпульси (меандр) заданої частоти для синхронізації зовнішніх пристроїв.

Функціонування DS1307 на шині I2C базується на архітектурі “ведучий-ведений”. Для доступу до внутрішньої пам’яті контролер формує сигнал START і надсилає адресу пристрою разом із номером початкового регістра. Завдяки автоматичному переходу до наступних комірок пам’яті, дані можуть передаватися безперервним потоком. Цикл обміну інформацією припиняється лише після подачі команди STOP.

У разі зниження напруги живлення VCC до рівня менше $1,25 \cdot V_{BAT}$, мікросхема DS1307 миттєво припиняє поточні операції обміну даними та обнуляє вбудований адресний лічильник. У цей період пристрій ігнорує будь-які зовнішні команди та сигнали на входах, що є необхідним заходом для запобігання некоректному запису інформації або спотворенню даних у регістрах.

Коли напруга VCC стає нижчою за рівень VBAT, мікросхема переходить у режим резервного живлення з мінімальним енергоспоживанням. Процес зворотного перемикавання з батареї на основну лінію VCC ініціюється, щойно напруга живлення перевищить поріг $V_{BAT} + 0,2$ В. Повноцінне ж відновлення обміну даними та сприйняття вхідних сигналів стає можливим лише після того, як значення VCC досягне рівня $1,25 \cdot V_{BAT}$.

На Рис. 3.6 зображено функціональну блок-схему мікросхеми DS1307 із послідовним інтерфейсом обміну даними.

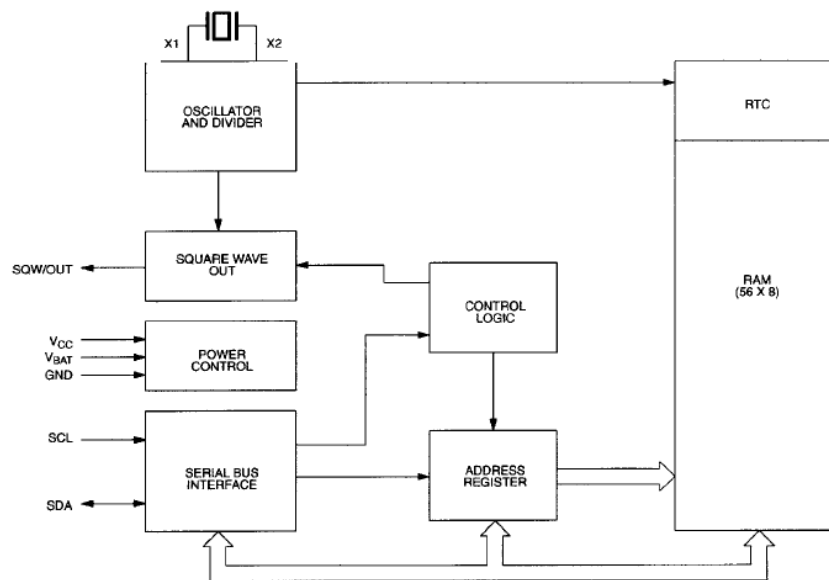


Рис. 3.6. Функціональна блок-схема мікросхеми DS1307 із послідовним інтерфейсом обміну даними

V_{CC} – основний вхід живлення напругою +5 В. Пристрій перебуває у стані повної функціональності, що дозволяє здійснювати операції зчитування та запису даних, доки рівень напруги перевищує поріг $1,25 \cdot V_{BAT}$. Якщо при підключеній 3-вольтовій батареї значення V_{CC} опускається нижче цього рівня, доступ до регістрів блокується для запобігання помилкам, хоча генерація часових міток не припиняється. У разі подальшого зниження напруги V_{CC} нижче потенціалу V_{BAT} , внутрішня пам'ять (SRAM) та блоки RTC автоматично переходять на споживання енергії від резервного елемента.

V_{BAT} – спеціалізований вхід, призначений для підключення резервного джерела живлення, як-от стандартного літієвого елемента на 3 В. Для стабільного функціонування мікросхеми DS1307 напруга на цьому контакті повинна перебувати в межах від 2,0 до 3,5 В. Використання літієвої батареї ємністю 48 мА·год або вище дозволяє підтримувати хід годинника в автономному режимі протягом понад 10 років (за умови експлуатації при температурі 25 °C).

SCL (Serial Clock Input) – вхід для подачі послідовних тактових імпульсів. Цей висновок використовується для синхронізації процесу передачі даних між ведучим пристроєм (мікроконтролером) та мікросхемою за послідовним інтерфейсом I2C.

SDA (Serial Data Input/Output) – двонаправлена лінія, призначена для послідовного обміну даними між компонентами системи. Оскільки вихід модуля реалізовано за схемою з відкритим стоком (open-drain), для коректної роботи інтерфейсу та формування високого логічного рівня необхідно використовувати зовнішній підтягувальний резистор (pull-up), підключений до шини живлення.

SQW/OUT (Square Wave/Output Driver) – багатофункціональний вихід, призначений для генерації сигналу прямокутної форми (меандру). Активація даного виводу відбувається шляхом встановлення біта SQWE у значення «1» у відповідному регістрі керування. Користувачеві доступний вибір однієї з чотирьох фіксованих частот тактування: 1 Гц, 4,096 кГц, 8,192 кГц або 32,768 кГц, що дозволяє використовувати мікросхему як джерело опорної частоти для зовнішніх пристроїв.

Вихідний каскад SQW/OUT реалізовано за схемою з відкритим стоком, що зумовлює необхідність встановлення зовнішнього підтягувального резистора до шини живлення. Важливою особливістю даного виводу є його здатність до генерації сигналу як при наявності основної напруги (V_{CC}), так і в режимі автономної роботи від резервної батареї (V_{BAT}). Це дозволяє використовувати модуль як незалежне джерело тактування навіть за відсутності зовнішнього живлення.

X1, X2 – вхідні контакти, призначені для підключення стандартного годинникового кварцового резонатора з номінальною частотою 32,768 кГц. Внутрішній тактовий генератор мікросхеми оптимізований для роботи з кристалами, що мають ємність навантаження (C_L) рівну 12,5 пФ. Дотримання цього параметра є необхідною умовою для забезпечення мінімальної часової похибки пристрою. Для мінімізації впливу паразитних ємностей та зовнішніх електромагнітних завад, кварцовий резонатор слід розташовувати на мінімальній відстані від висновків X1 та X2. Оптимальна топологія друкованої плати передбачає створення захисного кільця заземлення навколо контуру генератора (Рис.3.7).

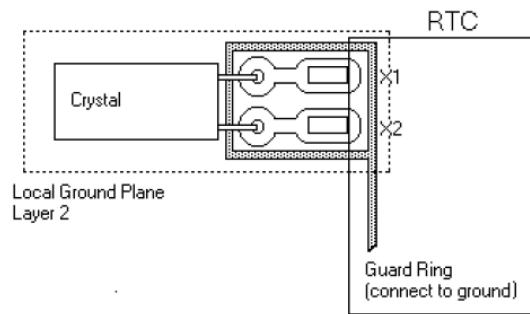


Рис. 3.7. Конструктивні рекомендації щодо трасування та розміщення кварцового резонатора

Точність функціонування модуля RTC безпосередньо залежить від метрологічних характеристик кварцового кристала та ємнісного балансу схеми осцилятора. Будь-яка невідповідність між внутрішньою ємністю резонатора та ємністю навантаження схеми тактування ініціює зсув номінальної частоти. В процесі експлуатації критичним фактором є температурна нестабільність кварцу, яка вносить нелінійну складову в загальну похибку. Також слід враховувати ризики наведення високочастотних шумів, які здатні викликати аномальне прискорення лічильників часу через паразитне спрацювання тригерів.

Організація адресного простору та пам'яті. Структура розподілу адрес для регістрів годинника реального часу (RTC) та статичного ОЗП (SRAM) наведена на рис. 3.8. Область керування часовими параметрами займає адресний простір від 00h до 07h. Решта доступної пам'яті (від 08h до 3Fh) відведена під потреби користувача для зберігання енергонезалежних даних. Механізм багатобайтового доступу реалізовано з використанням кільцевого адресного вказівника: при досягненні кінцевої адреси ОЗП (3Fh) наступний такт звернення автоматично перенаправляє вказівник на початкову адресу RTC (00h). На Рис.3.8 відображено логічну організацію внутрішньої пам'яті пристрою, розділену на сегменти часових регістрів (RTC) та енергонезалежного ОЗП користувача.

00H	SECONDS
	MINUTES
	HOURS
	DAY
	DATE
	MONTH
	YEAR
07H	CONTROL
08H	RAM
3FH	56 x 8

Рис. 3.8. Структура адресного простору та розподіл регістрів пам'яті DS1307

Програмний доступ до часових та календарних даних. Зчитування поточної інформації про час і дату здійснюється шляхом звернення до відповідних адресних комірок (Рис. 3.9). Процес встановлення або корекції часових параметрів реалізується через операцію запису значень у відповідні регістри. Важливою особливістю архітектури є представлення вмісту всіх регістрів у двійково-десятковому форматі (BCD). Для керування роботою осцилятора використовується 7-й біт нульового регістра (CH – Clock Halt): встановлення його в логічну “1” зупиняє тактовий генератор, що дозволяє знизити споживання енергії, тоді як скидання в “0” активує хід годинника.

	BIT7							BIT0
00H	CH	10 SECONDS			SECONDS			00-59
	0	10 MINUTES			MINUTES			00-59
	0	12 24	10 HR A/P	10 HR	HOURS			01-12 00-23
	0	0	0	0	0	DAY		1-7
	0	0	10 DATE		DATE			01-28/29 01-30 01-31
	0	0	0	10 MONTH	MONTH			01-12
	10 YEAR				YEAR			00-99
07H	OUT	0	0	SQWE	0	0	RS1	RS0

Рис. 3.9. Структурна організація та бітова маска регістрів хронометражу DS1307

На Рис. 3.9 наведено детальний розподіл розрядів у регістрах часу та календаря. Кожна комірка пам'яті (від 00h до 06h) відповідає за конкретний

параметр (секунди, хвилини, години, день тижня, число, місяць, рік), що дозволяє здійснювати точне налаштування та зчитування часових міток у форматі BCD.

Особливості початкової ініціалізації модуля. Необхідно враховувати, що під час подачі живлення на мікросхему DS1307 початковий стан внутрішніх регістрів є невизначеним. У зв'язку з цим, алгоритм ініціалізації системи повинен передбачати обов'язкове програмне скидання біта CH (біт 7 регістра 00h) у значення "0". Це забезпечує примусовий запуск тактового генератора та перехід пристрою в активний режим відліку часу.

Конфігурація форматів відображення часу. Мікросхема DS1307 підтримує два режими відліку годин: 12-годинний та 24-годинний. Вибір формату здійснюється за допомогою 6-го біта регістра годин (02h). Активація цього біта (логічна "1") переводить пристрій у 12-годинний режим, де 5-й біт функціонує як індикатор АМ/PM (високий рівень відповідає післяполудневому часу — PM). У разі використання 24-годинного формату (6-й біт дорівнює "0"), 5-й біт регістра використовується для представлення другого десятка годин (діапазон 20 – 23 години).

Механізм фіксації часових параметрів при зверненні до шини. Для забезпечення цілісності даних у мікросхемі DS1307 реалізовано метод подвійної буферизації. У момент виникнення умови START на двопровідній шині (I2C), актуальні значення з основних лічильників часу миттєво копіюються до допоміжного набору регістрів. Подальше зчитування інформації мікроконтролером відбувається саме з цих буферних комірок, тоді як основний годинниковий механізм продовжує безперервний відлік. Це усуває ризик отримання некоректних даних у разі оновлення вмісту регістрів безпосередньо в процесі їх зчитування.

Функціональне призначення регістра керування RTC. Регістр керування (адреса 07h) призначений для програмування режимів роботи універсального виходу SQW/OUT. За допомогою модифікації окремих бітів цього регістра встановлюється стан вихідного каскаду, активується генерація прямокутних

імпульсів та обирається їхня номінальна частота. Це дозволяє адаптувати модуль RTC для синхронізації зовнішніх периферійних пристроїв або тактування мікроконтролера (Табл.3.6).

Таблиця 3.6. Структура та функціональне призначення бітів регістра керування (07h)

Біт 7	Біт 6	Біт 5	Біт 4	Біт 3	Біт 2	Біт 1	Біт 0
OUT	0	0	SQWE	0	0	RS1	RS0

Опис функціонування:

- OUT (Output Control): Визначає логічний рівень на виході SQW/OUT, коли генерація прямокутних імпульсів вимкнена (SQWE=0).
- SQWE (Square Wave Enable): При встановленні в “1” активує вбудований генератор меандру.
- RS1, RS0 (Rate Select): Визначають частоту вихідного сигналу (1 Гц, 4,096 кГц, 8,192 кГц або 32,768 кГц). У Табл.3.7 наведено залежність частоти прямокутних імпульсів від комбінації керуючих бітів селекції швидкості (RS1 та RS0). Ці налаштування є актуальними лише за умови активації біта дозволу генерації (SQWE=1).
- Біти 1, 2, 4, 5, 6: Зарезервовані виробником і зазвичай зчитуються як “0”.

Таблиця 3.7. Конфігурація вихідної частоти сигналу SQW/OUT

RS1	RS0	Вихідна частота SQW/OUT
0	0	1 Гц
0	1	4,096 кГц
1	0	8,192 кГц
1	1	32,768 кГц

DS1307 підтримує обмін даними по протоколу I2C по двопровідній двонаправленій шині. Пристрій, який передає дані на шину, є передавачем (transmitter), а пристрій, що приймає дані – приймачем (receiver). Пристрій, що управляє передачею даних, називається ведучим. Пристрій, яким керує ведучий, називається веденим. Ведучий пристрій генерує синхроімпульси (serial clock - SCL), управляє доступом до шини і генерує умови START і STOP. DS1307 працює на шині як ведений пристрій. Типова конфігурація шини з використанням протоколу I2C показана на Рис.3.10.

Організація обміну даними RTC за протоколом I2C. Взаємодія мікросхеми DS1307 із зовнішніми пристроями реалізується через двопровідну двоспрямовану послідовну шину за протоколом I2C. У межах цієї архітектури пристрій, що ініціює передачу, визначається як передавач (transmitter), а пристрій-адресат — як приймач (receiver). Роль координатора обміну виконує ведучий пристрій (Master), який відповідає за формування синхросигналу (SCL), арбітраж доступу до шини та генерацію службових умов START і STOP. Модуль DS1307 функціонує виключно як ведений пристрій (Slave). Типова схема підключення компонентів до шини наведена на рис. 3.10.

Обмін інформацією здійснюється по шині I2C, яка складається з ліній даних (SDA) та синхронізації (SCL). Протокол передбачає чітку ієрархію: центральний мікроконтролер виступає в ролі ведучого (Master), генеруючи тактові імпульси та керуючи сесіями зв'язку. DS1307, будучи веденим пристроєм (Slave), очікує на адресу від ведучого та відповідає на запити зчитування або запису. Структурна схема такої конфігурації, що включає підтягувальні резистори та декілька вузлів, представлена на Рис.3.10.

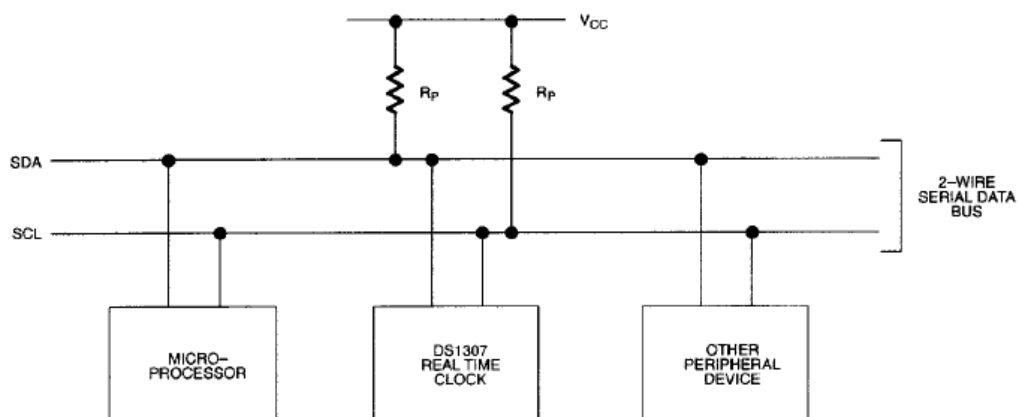


Рис. 3.10. Архітектура та схема підключення пристроїв до двопровідної послідовної шини I2C

Представлена на Рис.3.10 типова топологія мережі I2C включає ведучий пристрій (MCU) та ведений вузол (DS1307). Особливістю даної конфігурації є використання ліній з відкритим стоком, що потребує обов'язкового встановлення підтягувальних резисторів (R_p) до шини живлення V_{CC} для забезпечення коректних логічних рівнів.

На Рис. 3.11 – 3.13 зображено часові діаграми повного циклу передачі та запису байта даних по шині I2C. Основні принципи передачі даних по шині I2C:

1. Умова вільної шини: Ініціація обміну дозволяється лише тоді, коли лінії SDA та SCL перебувають у високому логічному стані.
2. Валідація даних: Зміна стану лінії SDA припустима виключно під час низького рівня сигналу SCL.
3. Керуючі сигнали: Будь-яка зміна рівня SDA, що відбувається при високому рівні SCL, інтерпретується логікою пристрою не як біт даних, а як службова умова (START або STOP).

Аналіз станів та режимів функціонування шини I2C. У межах реалізації протоколу обміну з модуля DS1307 виділяють наступні стаціонарні та перехідні стани комунікаційного середовища:

- Пасивний стан (Bus Not Busy): Обидві лінії (SDA, SCL) утримуються у високому логічному стані.
- Умова START: Ініціюється ведучим пристроєм шляхом переведення лінії SDA з високого рівня в низький при стабільному високому рівні SCL.
- Умова STOP: Формується при переході SDA з низького рівня у високий при високому стані SCL, що сигналізує про завершення поточної сесії.
- Верифікація даних: Коректність біта забезпечується незмінністю стану SDA протягом високої фази тактового імпульсу. Модифікація даних припустима лише під час низького рівня SCL.
- Механізм підтвердження (ACK/NACK): Кожен переданий байт супроводжується дев'ятим тактовим імпульсом, під час якого приймач зобов'язаний утримувати SDA на низькому рівні (ACK). Відсутність підтвердження на останньому байті при зчитуванні з DS1307 використовується ведучим для завершення передачі перед генерацією умови STOP.

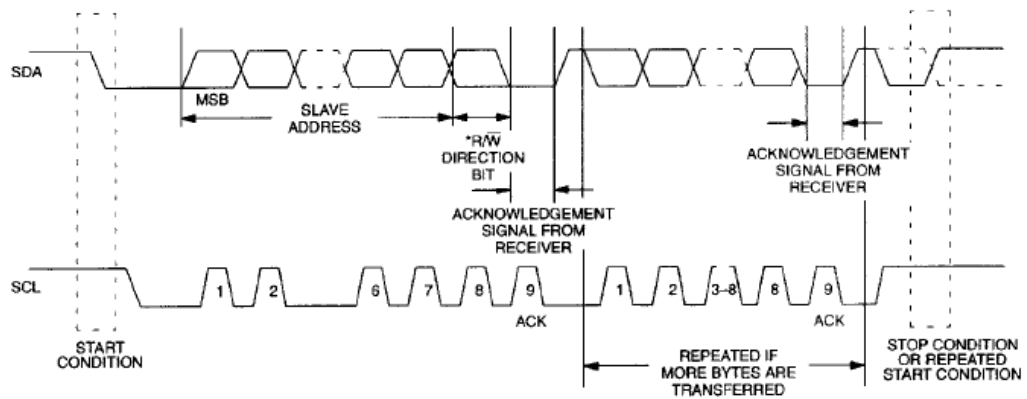


Рис. 3.11. Часова діаграма повного циклу передачі байта даних по шині I2C

Протоколи транзакцій шини I2C. Логіка керування обміном передбачає два основні типи передачі:

- Трансляція від МК до RTC: Ведучий пристрій формує кадр, що складається з адреси та даних. Важливою умовою є моніторинг біта ACK після кожного байта; відсутність підтвердження сигналізує про помилку на шині.
- Трансляція від RTC до МК: Ведучий пристрій забезпечує тактування для зчитування байтів з регістрів DS1307. Ключовим моментом алгоритму є вимкнення генерації ACK перед термінацією зв'язку умовою STOP. Це звільняє лінію SDA, дозволяючи ведучому коректно завершити цикл.

DS1307 може працювати у наступних двох режимах:

1. Функціонування DS1307 у режимі веденого приймача (запис даних). У режимі запису модуль DS1307 здійснює апаратне розпізнавання власної 7-бітної адреси (1101000) та біта напрямку обміну ($R/\overline{W}=0$), що слідує за умовою START. Після успішної ідентифікації пристрій генерує сигнал підтвердження (ACK). Перший байт даних, отриманий від ведучого після адресного кадру, автоматично завантажується у внутрішній вказівник адрес (Address Pointer), визначаючи регістр, з якого почнеться запис. Наступні байти записуються безпосередньо в пам'ять RTC, причому після кожного байта вказівник автоматично інкрементується. Завершення сеансу зв'язку ініціюється ведучим пристроєм через формування умови STOP.

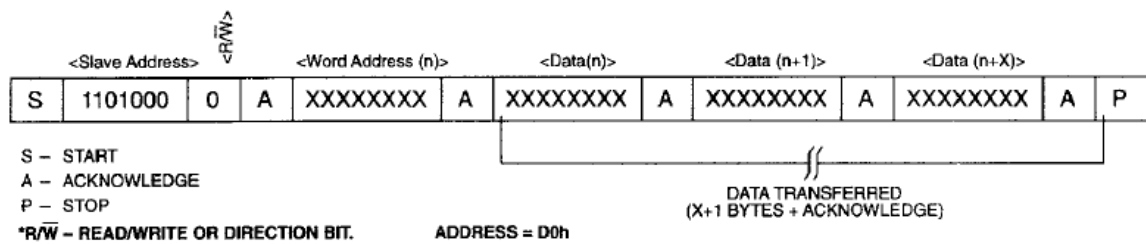


Рис. 3.12. Часова діаграма транзакції запису в режимі Slave Receiver

На Рис.3.12 наведено повну послідовність формування пакетів: після умови START передається байт адреси пристрою з бітом запису ($R/\overline{W}=0$), перший байт даних інтерпретується як адреса внутрішнього регістра (вказівник), а наступні байти – як інформація для збереження. Кожен етап супроводжується сигналом підтвердження (ACK) від веденого пристрою до моменту генерації умови STOP.

2. Функціонування DS1307 у режимі веденого передавача (зчитування даних). Протокол зчитування ініціюється передачею байта адреси пристрою, де біт напрямку встановлюється в логічну “1” ($R/\overline{W}=1$). Після апаратного розпізнавання адреси та формування сигналу підтвердження (ACK), DS1307 починає послідовну трансляцію вмісту регістрів, починаючи з поточної позиції внутрішнього вказівника. Дані передаються по лінії SDA синхронно з тактовими імпульсами SCL, що генеруються ведучим пристроєм. Процес триває до моменту, поки ведучий не сформує сигнал непідтвердження (NACK) на останньому байті, що є обов’язковою умовою перед генерацією сигналу STOP.

Формування адресного кадру та механізм доступу до даних. Першим байтом, що транслюється ведучим пристроєм після умови START, є адресний кадр. Він складається з унікальної 7-бітної адреси модуля (1101000) та біта вибору напрямку $R/\overline{W}$, який для операції читання встановлюється в логічну “1”. Після апаратного підтвердження (ACK) від DS1307, пристрій розпочинає послідовну видачу даних, починаючи з поточної позиції регістрового вказівника. Важливо враховувати, що якщо перед початком сесії не було виконано операцію встановлення адреси, зчитування почнеться з комірки, на якій зупинився вказівник під час попередньої транзакції. Завершення

процедури вимагає від ведучого генерації сигналу непідтвердження (NACK) на останньому байті.

На Рис.3.13 відображено процес передачі даних від DS1307 до ведучого пристрою. Після передачі адреси пристрою з бітом читання ($R/\bar{W}=1$), модуль RTC трансліює вміст регістрів, починаючи з поточної позиції вказівника. Особливістю даної діаграми є завершальний етап: ведучий пристрій формує сигнал непідтвердження (NACK) після останнього байта, що є необхідною умовою для коректної генерації сигналу STOP та звільнення шини.



Рис. 3.13. Часова діаграма транзакції читання в режимі Slave Transmitter

У Табл.3.8 наведено електричні характеристики інтерфейсу та живлення мікросхеми RTC DS1307. Дані параметри дозволяють узгодити рівні сигналів шини I2C між мікроконтролером ATmega328P та мікросхемою DS1307, а також розрахувати термін служби резервного елемента живлення.

Таблиця 3.8. Електричні характеристики інтерфейсу та живлення мікросхеми RTC DS1307

Характеристика	Позначення	Діапазон значень	Примітка
Напруга основного живлення	V_{CC}	4,5 В ... 5,5 В	Номінал 5,0 В
Напруга літієвого елемента	V_{BAT}	2,0 В ... 3,5 В	Тип CR2032
Вхідний рівень високий	V_{IH}	2,2 В ... $V_{CC}+0,3$ В	Для SDA, SCL
Вхідний рівень низький	V_{IL}	-0,3 В ... +0,8 В	Для SDA, SCL
Струм у режимі очікування	I_{CCS}	< 200 мкА	При $V_{CC}=5$ В

На Рис.3.14 наведено часові параметри, що визначають динаміку обміну: час встановлення та утримання даних ($t_{SU:DAT}$, $t_{HD:DAT}$), тривалість станів високого та низького рівнів тактового сигналу (t_{HIGH} , t_{LOW}), а також інтервали формування умов START та STOP. Дотримання цих часових обмежень є необхідною умовою для запобігання помилок синхронізації при роботі мікроконтролера з RTC на частоті 100 кГц.

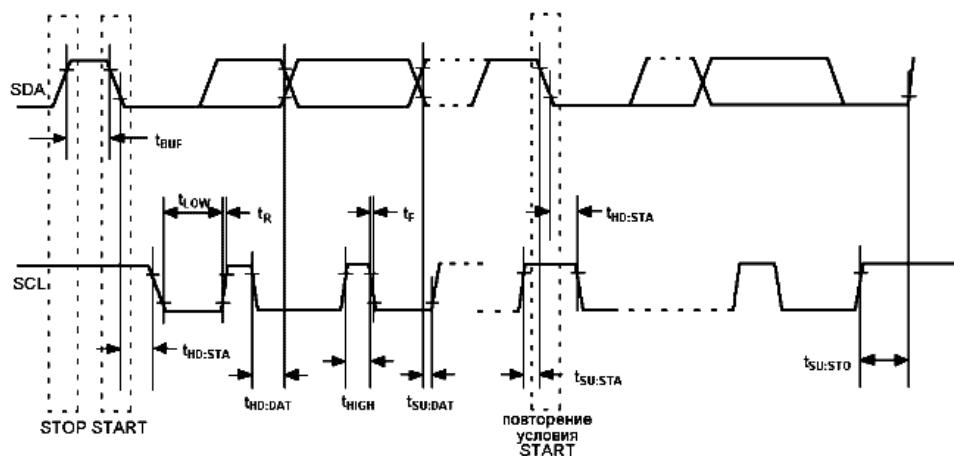


Рис. 3.14. Повна часова діаграма сигналів шини I2C для модуля DS1307

Обґрунтування вибору сенсора температурного моніторингу.

Сучасна номенклатура електронних компонентів надає розмаїття засобів вимірювання температури, що різняться за динамічним діапазоном, енергоспоживанням, типом інтерфейсу та конструктивним виконанням. У межах даної розробки пріоритет віддано цифровому термометру DS18B20 від компанії Dallas Semiconductor (Maxim Integrated). Ключовою перевагою даного сенсора є інтегрований аналого-цифровий перетворювач із програмованою розрядністю (від 9 до 12 біт), що дозволяє гнучко варіювати між швидкістю вимірювання та точністю представлення даних [12].

Ключові переваги використання DS18B20 у системі: 1. Протокол 1-Wire: Дозволяє зчитувати дані лише через один цифровий порт МК. 2. Унікальна ідентифікація: 64-бітний код забезпечує безпомилкову роботу в багатоточкових системах контролю. 3. Електрична сумісність: Підтримка логічних рівнів 3,3 В та 5 В без додаткових вузлів узгодження. 4. Точність та дискретність: Висока точність у робочому діапазоні та можливість вибору дискретності вимірювання (9–12 біт). 5. Автономна сигналізація: Вбудовані тригери для реалізації термостатичних функцій без постійної участі центрального процесора.

Вибір датчика DS18B20 обумовлений його ідентичністю протоколів обміну з серією DS1822, що значно спрощує розробку програмного забезпечення. Універсальність пристрою підтверджується його активним

впровадженням у промислові системи керування та побутову техніку. Використання таких сенсорів у термочутливих системах різного призначення гарантує стабільність метрологічних характеристик та стійкість до зовнішніх чинників, що є важливим аргументом для їх інтеграції у проєктований вимірювальний комплекс.

Конструктивні особливості та принципи функціонування DS18B20.

Цифровий термодатчик DS18B20 виготовляється у кількох варіантах корпусів, серед яких найпоширенішими є 8-вивідні SOIC та μ SOP, а також малогабаритний трививідний корпус TO-92, який найчастіше використовується у вбудованих системах. Обмін даними з центральним процесором реалізовано на базі пропрієтарного протоколу 1-Wire.

Унікальною архітектурною особливістю пристрою є підтримка режиму “паразитного живлення” (parasite power), що дозволяє сенсору отримувати необхідну енергію безпосередньо з шини даних через вбудований накопичувальний конденсатор. Завдяки інтегрованому 64-бітному серійному номеру, система дозволяє створювати розгалужені мережі моніторингу на базі всього однієї сигнальної лінії, забезпечуючи індивідуальну адресацію кожного вузла вимірювання.

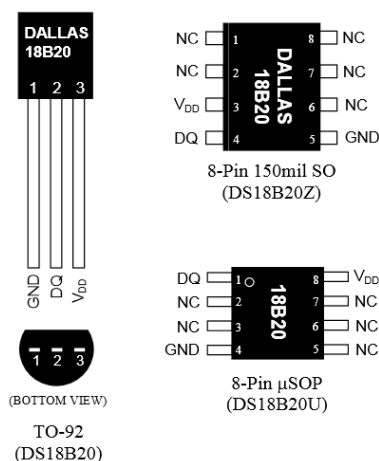


Рис. 3.15. Зовнішній вигляд мікросхеми термодатчика DS18B20 у варіантах виконання TO-92, SOIC та μ SOP

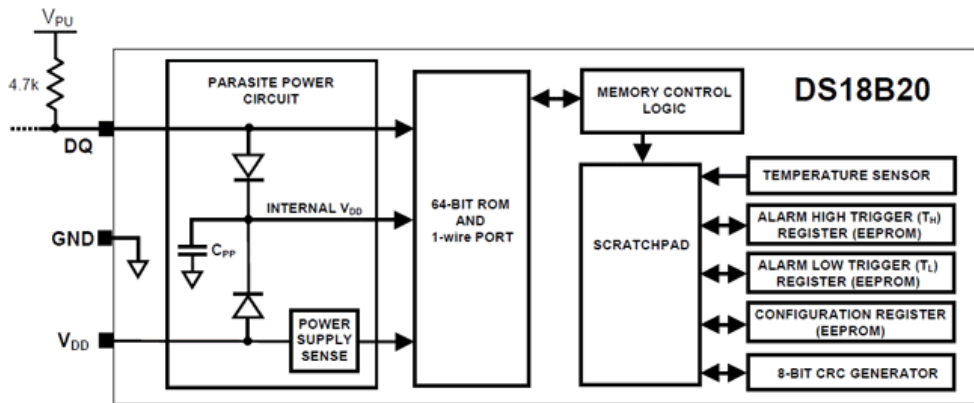


Рис. 3.16. Внутрішня структурна архітектура мікросхеми термодатчика DS18B20

На представленій блок-схемі відображено ключові функціональні вузли датчика, що забезпечують прецизійне вимірювання температури. Основними компонентами є:

- 64-бітне ПЗП (ROM): зберігає унікальний серійний номер пристрою для ідентифікації на шині 1-Wire.
- Датчик температури: аналоговий вузол, що перетворює теплову енергію в цифровий код.
- Логіка керування 1-Wire: забезпечує обмін даними та живлення пристрою в паразитному режимі.
- Пам'ять Scratchpad (Блокнот): містить 2-байтний регістр температури, а також енергонезалежні регістри тривоги (T_H та T_L) і регістр конфігурації роздільної здатності (9 – 12 біт).

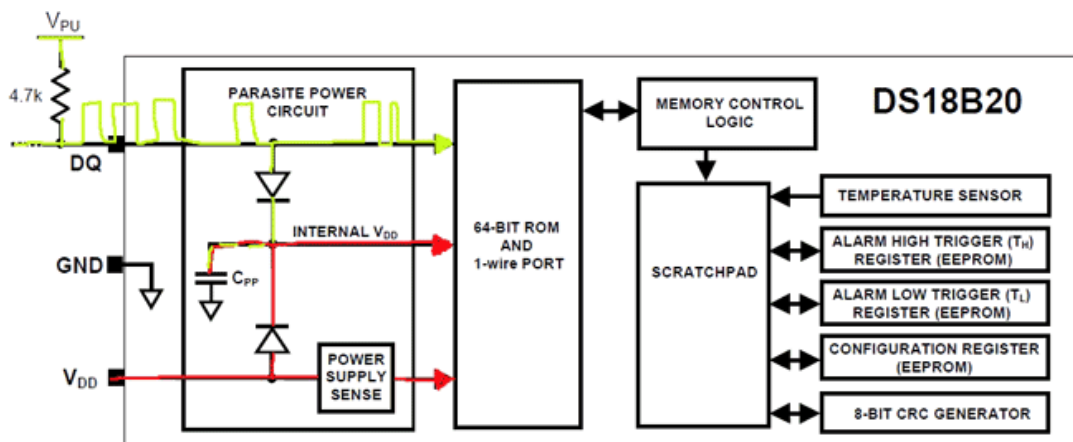


Рис. 3.17. Схема електрична принципова підсистеми живлення мікросхеми термодатчика DS18B20

Архітектура датчика дозволяє реалізувати двопровідну схему підключення з використанням технології паразитного живлення. Механізм енергозабезпечення (позначений на схемі зеленим кольором) базується на використанні енергії шини даних DQ. У моменти, коли на лінії встановлено високий логічний рівень, струм через підтягуючий резистор та внутрішній діодний каскад забезпечує заряджання накопичувального конденсатора C_{PP} . Одночасно з цим здійснюється живлення внутрішніх вузлів датчика через лінію INTERNAL Vdd. У фазі передачі даних, коли на шині DQ формується низький логічний рівень, стабільне функціонування пристрою підтримується завдяки накопиченому заряду конденсатора.

Використання паразитного живлення накладає певні обмеження на часові діаграми обміну даними. Тривале утримання шини DQ у стані низького логічного рівня призводить до розряду буферного конденсатора та припинення енергозабезпечення внутрішньої лінії INTERNAL Vdd, що деактивує пристрій. Тому в режимі очікування на лінії обов'язково має підтримуватися високий потенціал. Особливу увагу слід приділити енергоємним операціям: перетворенню температури (Convert T) та запису даних у EEPROM. У ці моменти струм споживання зростає до 1,5 мА. Стандартний підтягуючий резистор через падіння напруги не здатний забезпечити такий струм, що призводить до збоїв. Для вирішення цієї проблеми застосовується схема “сильної підтяжки” (Strong Pullup), яка шунтує резистор низькоомним каналом (наприклад, через польовий транзистор) на час виконання критичних операцій.

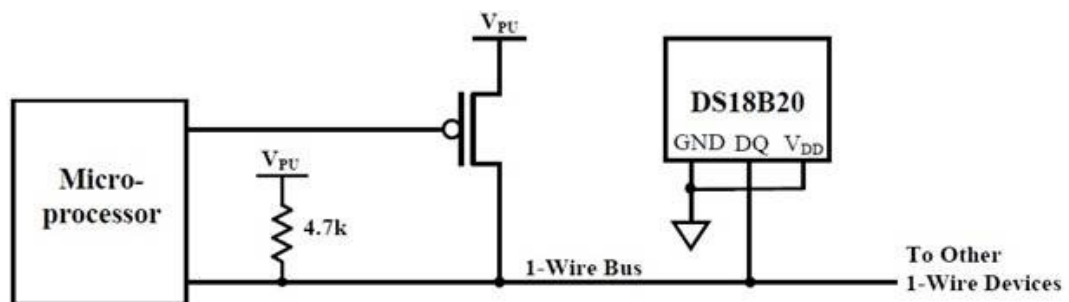


Рис. 3.18. Схема активної підтяжки шини 1-Wire для забезпечення режиму паразитного живлення

На Рис.3.18 представлена схема вузла форсованого живлення, що складається зі стандартного резистора підтяжки R_{pu} (4,7k) та шунтуючого польового транзистора (MOSFET). Транзистор підключається паралельно резистору і керується окремим виводом мікроконтролера. Під час виконання енергоємних операцій (перетворення температури), транзистор відкривається, створюючи канал з низьким опором між лінією DQ та шиною живлення VDD. Це дозволяє подати струм до 1,5 мА без падіння напруги, що критично для стабільної роботи датчика в двопровідному режимі.

Реалізація режиму енергозабезпечення через шину даних вимагає чіткої синхронізації керуючих сигналів. Безпосередньо після ініціалізації команд оцифрування температури чи перенесення даних до EEPROM, необхідно забезпечити форсоване живлення лінії DQ. Максимально допустима затримка ввімкнення MOSFET-транзистора становить 10 мкс. Під час активної фази підтяжки будь-яка активність на шині заборонена, оскільки це призведе до просідання напруги та збою в роботі сенсора. Варто зауважити, що при активному (зовнішньому) живленні схема значно спрощується, а використання потужної підтяжки стає недоцільним, оскільки струм споживання покривається прямим підключенням до джерела живлення.

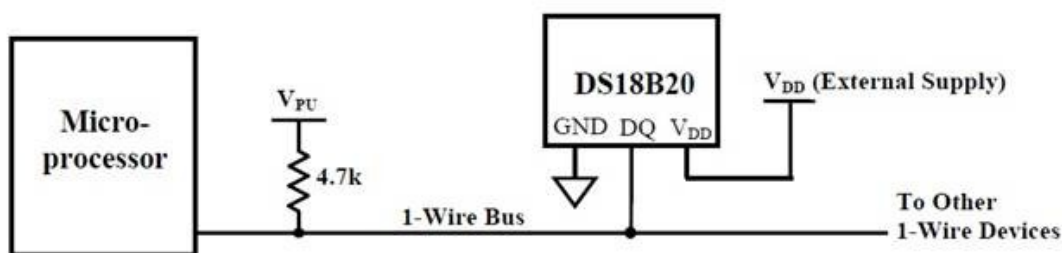


Рис. 3.19. Схема підключення мікросхеми DS18B20 із використанням зовнішнього живлення

При виборі конфігурації живлення термосенсора необхідно враховувати планований температурний діапазон експлуатації. Використання режиму паразитного живлення не є доцільним для систем, що працюють у зоні температур понад 100°C. Це обумовлено експоненціальним зростанням струмів витоку в напівпровідниковій структурі датчика при сильному

нагріванні, що може призвести до критичного просідання напруги на внутрішньому накопичувальному конденсаторі та виникнення значних похибок вимірювання. Для забезпечення достовірності даних у високотемпературних середовищах обов'язковим є застосування схеми із зовнішнім джерелом живлення.

Протокол 1-Wire дозволяє ведучому пристрою (МК) динамічно визначати конфігурацію живлення підключених вузлів. Процедура перевірки включає послідовну передачу команд Skip ROM [CCh] та Read Power Supply [B4h]. Реакція датчика на цей запит дозволяє МК обрати стратегію керування шиною:

- Низький рівень на DQ: датчик працює в паразитному режимі, вимагаючи ввімкнення потужної підтяжки при оцифруванні.
- Високий рівень на DQ: активовано стандартне живлення, що дозволяє проводити вимірювання без додаткових апаратних маніпуляцій з лінією даних. Такий підхід забезпечує універсальність програмного забезпечення та надійність збору даних у мережах із різними типами підключення.

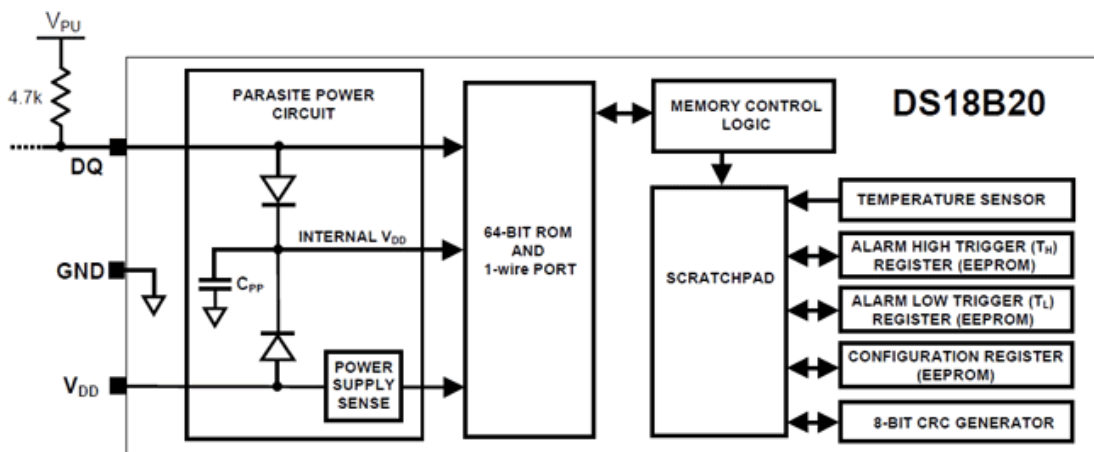


Рис. 3.20. Внутрішня архітектура та функціональна організація вузлів мікросхеми DS18B20

Функціональний блок “64-BIT ROM AND 1-Wire PORT” інтегрує в собі засоби апаратної ідентифікації та комунікації. Він містить унікальний 64-розрядний серійний номер, записаний у нелетку пам'ять (ROM) на етапі виробництва, що забезпечує індивідуальну адресацію пристрою. Також у

цьому вузлі реалізовано фізичний та логічний рівні інтерфейсу 1-Wire, який відповідає за двосторонній обмін даними з центральною системою керування за допомогою лише однієї сигнальної лінії.

Керування потоками даних всередині датчика здійснюється модулем “Memory Control Logic”. Ця підсистема забезпечує прозорий обмін інформацією між портом інтерфейсу 1-Wire та оперативною пам’яттю типу Scratchpad. Остання виконує роль буфера, через який реалізується доступ до ключових ресурсів пристрою: результатів вимірювання температури, конфігураційного регістру, а також регістрів налаштування температурних порогів (T_H та T_L). Крім того, логіка керування пам’яттю взаємодіє з генератором 8-бітного циклічного надлишкового коду (CRC), що гарантує цілісність даних та захист системи від помилок при передачі.

При поданні напруги живлення датчик автоматично ініціалізується з роздільною здатністю 12 біт за замовчуванням і переходить у стан очікування з мінімальним енергоспоживанням (Idle mode). Для запуску циклу вимірювання температури ведучий пристрій (мікроконтролер) має надіслати команду Convert T [44h]. Після завершення процесу аналого-цифрового перетворення отриманий результат фіксується в оперативній пам’яті (Scratchpad) у форматі 16-бітного слова (двох байтів). Одразу після оновлення даних у регістрах пристрій автоматично повертається до енергозберігаючого режиму для оптимізації загального споживання системи.

Після ініціації вимірювання командою Convert T [44h] мікроконтролер має можливість здійснювати моніторинг стану готовності датчика шляхом зчитування часових слотів. Якщо пристрій повертає логічний “0”, це свідчить про те, що цикл аналого-цифрового перетворення ще триває. Отримання логічної “1” підтверджує завершення формування цифрового коду температури. Проте варто зауважити, що даний метод детермінації стану готовності є неприпустимим для конфігурацій із паразитним живленням. Це зумовлено необхідністю утримання шини DQ у стані високого рівня (через

MOSFET-ключ) для забезпечення енергопотреб датчика протягом усього інтервалу T_{conv} .

Процес вимірювання температури в DS18B20 базується на використанні інтегрованого аналого-цифрового перетворювача (АЦП). Внутрішній термочутливий елемент генерує аналоговий сигнал, який квантується АЦП і перетворюється у цифровий еквівалент. Отримане значення реєструється в спеціалізованому двобайтному регістрі температури, що є частиною оперативної пам'яті (Scratchpad). Дані про температуру представлені у вигляді числа з фіксованою комою у форматі доповняльного коду, що дозволяє коректно відображати як позитивні, так і негативні значення.

LS байт	біт7	біт6	біт5	біт4	біт3	біт2	біт1	біт0
	2^3	2^2	2^1	2^0	2^{-1}	2^{-2}	2^{-3}	2^{-4}
MS байт	біт15	біт14	біт13	біт12	біт11	біт10	біт9	біт8
	S	S	S	S	S	2^6	2^5	2^4

Формат представлення температури в DS18B20 базується на використанні 16-бітного поля даних. Старші п'ять розрядів (біти 15 – 11) відведені під прапорець знаку (S). Значення $S=0$ вказує на позитивну температуру, тоді як $S=1$ сигналізує про від'ємне значення, представлене у форматі додаткового коду (доповнення до двох). При встановленні максимальної роздільної здатності (12 біт) всі розряди з 0 по 11 містять валідні дані. У разі зниження точності квантування, молодші біти стають надлишковими: при 11-бітному перетворенні не враховується біт 0, при 10-бітному – біти 0 та 1, а при 9-бітному – біти 0, 1 та 2 відповідно.

Функціонал термостата в датчику DS18B20 базується на використанні двох 8-бітних енергонезалежних регістрів: T_H (верхній поріг) та T_L (нижній поріг). У разі, якщо поточний результат вимірювання виходить за межі діапазону $[T_L, T_H]$, внутрішня логіка датчика встановлює прапорець тривоги (Alarm Flag). Ведучий пристрій (мікроконтролер) може ідентифікувати такі датчики в багатоточковій мережі за допомогою спеціалізованої команди Alarm Search [ECh]. Важливою особливістю є те, що стан прапорця тривоги

автоматично оновлюється після завершення кожного циклу аналого-цифрового перетворення температури.

Варто зауважити, що апаратна логіка порівняння порогових значень використовує лише розряди з 11 по 4 регістру температури. Це означає, що функція термостата оперує виключно цілочисельними значеннями, ігноруючи дробову частину. Регістри T_H та T_L фізично реалізовані на базі енергонезалежної пам'яті EEPROM, що забезпечує збереження встановлених порогів після відключення живлення. За своєю структурою вони ідентичні старшому байту регістру температури: кожен регістр є 8-бітним, де старший розряд (біти 15 – 11 у 16-бітній сітці, або 7-й біт у 8-бітній) виконує роль знакового прапорця S , аналогічно до формату представлення температурних даних.

біт7	біт6	біт5	біт4	біт3	біт2	біт1	біт0
S	2^6	2^5	2^4	2^3	2^2	2^1	2^0

Як було зазначено раніше, унікальний 64-розрядний код є ключовим елементом для адресації пристроїв у багатоточкових (multi-drop) мережах 1-Wire. Даний ідентифікатор записаний у нелетку пам'ять (ROM) кожної мікросхеми на етапі виробництва, що виключає наявність двох пристроїв із однаковими адресами на одній шині. Структура цього коду має чітку ієрархію розрядів:

- 8-бітний код сімейства (Family Code): Визначає тип пристрою (для DS18B20 цей код завжди дорівнює 28h).
- 48-бітний серійний номер: Унікальна комбінація бітів, що ідентифікує конкретний екземпляр датчика.
- 8-бітна контрольна сума (CRC): Обчислюється на основі попередніх 56 біт і призначена для верифікації цілісності зчитаного коду.

8-бітний CRC		48-бітний серійний номер		8-бітний код сімейства	
MSB	LSB	MSB	LSB	MSB	LSB

Структура 64-бітного ідентифікаційного коду ROM має чіткий розподіл за функціональним призначенням розрядів. Молодший байт (біти 0–7) відведений під код сімейства пристроїв, який для моделі DS18B20 має сталі

значення 28h. Наступний масив із 48 біт (біти 8–55) формує унікальний серійний номер, що забезпечує індивідуальну адресацію кожного сенсора. Завершальний старший байт (біти 56–63) містить значення циклічного надлишкового коду (CRC). Ця контрольна сума обчислюється на основі попередніх 56 біт і використовується ведучим пристроєм для підтвердження достовірності зчитаного серійного номера.

Організація пам'яті в DS18B20. Регістрова структура DS18B20 базується на поєднанні оперативної пам'яті блокуного типу (Scratchpad) та енергонезалежної пам'яті EEPROM (Рис.3.21). Останні призначені для стабільного зберігання конфігураційних параметрів та порогових значень термостата (T_H та T_L). При відключенні живлення дані байтів 2, 3 та 4 автоматично зберігаються в EEPROM, а при наступному ввімкненні системи вони автоматично перезавантажуються в Scratchpad, забезпечуючи неперервність налаштувань. Байти 0 та 1 зарезервовані для результатів оцифрування температури, тоді як байти 5, 6 та 7 є службовими і недоступні для користувацького редагування. Завершальний 8-й байт містить апаратно генерований CRC-код, що верифікує цілісність попередніх семи байтів. Слід підкреслити універсальність регістрів T_H та T_L : за умови відсутності потреби у функції термостата, їх можна використовувати як енергонезалежну пам'ять загального призначення для зберігання довільних ідентифікаторів чи параметрів системи.

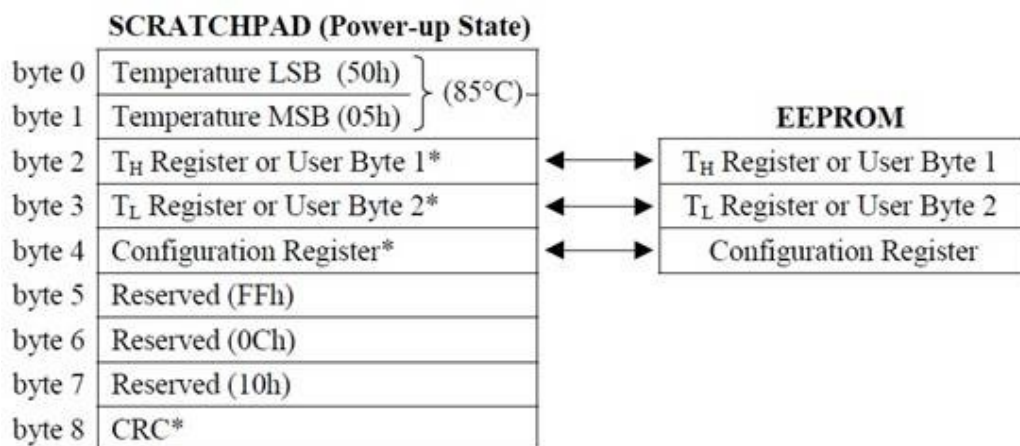


Рис. 3.21. Структура регістрів оперативної пам'яті (Scratchpad) датчика DS18B20

Запис конфігураційних параметрів у регістри T_H , T_L та Configuration Register (байти 2, 3 та 4 відповідно) здійснюється за допомогою команди Write Scratchpad [4Eh]. Передача даних ініціюється з молодшого значущого біта (LSB) другого байта. Для забезпечення надійності функціонування системи передбачено процедуру верифікації: записані значення зчитуються командою Read Scratchpad [BEh]. При цьому мікроконтролер приймає повний пакет даних, починаючи з молодшого біта нульового байта, що дозволяє порівняти фактично записані параметри з еталонними.

Для фіксації порогових значень термостата (T_H , T_L) та параметрів конфігураційного регістру в енергонезалежній пам'яті EEPROM, ведучий пристрій має ініціювати команду Copy Scratchpad [48h]. Ця операція забезпечує перенесення даних із оперативних регістрів у нелетку пам'ять, що гарантує їх збереження при повному знеструмленні датчика. При наступній подачі живлення на мікросхему реалізується автоматичний цикл ініціалізації: вміст відповідних комірок EEPROM трансліюється назад у регістри Scratchpad, відновлюючи робочі параметри системи без участі мікроконтролера.

Архітектура датчика передбачає можливість примусового відновлення параметрів із енергонезалежної пам'яті в оперативну в будь-який момент функціонування системи. Це актуально у випадках, коли під час виконання певних ітерацій вимірювання було тимчасово змінено роздільну здатність або пороги тривоги, і виникла потреба повернути пристрій до вихідних (заздалегідь збережених) налаштувань. Для реалізації цієї процедури ведучий пристрій має ініціювати команду Recall E² [B8h], яка копіює дані з EEPROM у відповідні регістри Scratchpad, забезпечуючи швидку реініціалізацію робочого стану без перезавантаження живлення. Протягом дії команди відновлення даних мікроконтролер може перевіряти статус готовності датчика: логічний "0" означає, що процес копіювання ще триває, а логічна "1" – що дані з EEPROM повністю перенесені у робочі регістри. Це дозволяє системі

динамічно підлаштовуватися під час виконання внутрішніх операцій датчика без використання фіксованих пауз.

Конфігурування роздільної здатності перетворення датчика DS18B20. У внутрішньому регістрі конфігурації пристрою за вибір точності вимірювань відповідають два розряди: R0 та R1.

bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
0	R1	R0	1	1	1	1	1

За замовчуванням біти R0 та R1 встановлені в стан логічної “1”, що відповідає максимальній 12-бітній роздільній здатності. Користувач має можливість змінювати ці параметри для оптимізації швидкодії системи (Табл.3.9). Слід враховувати зворотну залежність: підвищення роздільної здатності збільшує тривалість циклу вимірювання. Зокрема, для 12-бітного режиму максимальний час перетворення становить 750 мс, тоді як для 9-бітного він скорочується в 8 разів.

Таблиця 3.9. Залежність роздільної здатності та часу перетворення датчика DS18B20

R1	R0	Роздільна здатність	Дискретність	Макс. час перетворення	
0	0	9 біт	0,5°C	93,75 мс	(tconv/8)
0	1	10 біт	0,25°C	187,5 мс	(tconv/4)
1	0	11 біт	0,125°C	375 мс	(tconv/2)
1	1	12 біт	0,0625°C	750 мс	(tconv)

Фізична реалізація інтерфейсу 1-Wire для датчика DS18B20. Як було зазначено, обмін даними між датчиком та центральним мікроконтролером здійснюється через послідовну шину 1-Wire. Для забезпечення коректної роботи цього протоколу, керуюча система повинна підтримувати вихідний каскад типу “відкритий стік” або переводити лінію в стан високого опору (Hi-Z). Така конфігурація дозволяє декільком пристроям на шині спільно використовувати одну лінію зв’язку, уникаючи конфліктів рівнів напруги.

На Рис.3.22 зображено архітектуру вхідних/вихідних каскадів датчика, що базується на використанні транзистора з відкритим стоком. Така конфігурація дозволяє реалізувати двоспрямований обмін даними по одній сигнальній лінії. Внутрішній польовий транзистор забезпечує активне

підтягування шини до потенціалу “землі” (GND) при передачі логічного нуля, тоді як стан логічної одиниці формується зовнішнім резистором R_{PU} .

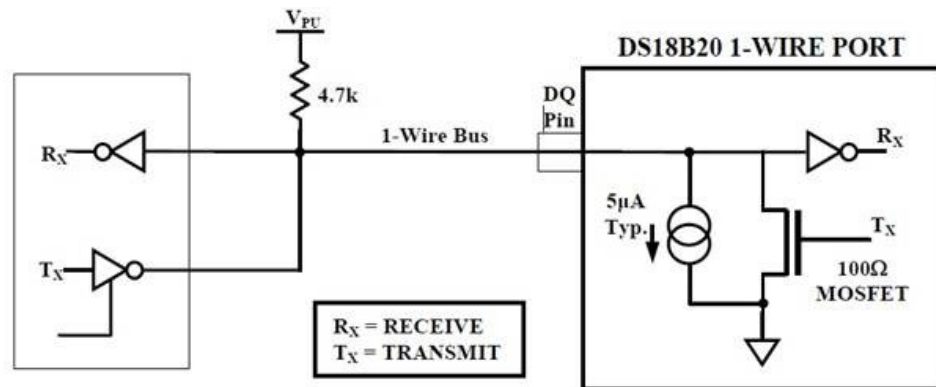


Рис. 3.22. Спрощена схема внутрішнього інтерфейсу 1-Wire датчика DS18B20

Алгоритм обміну даними за протоколом 1-Wire. Взаємодія керуючої системи з термодатчиком базується на трирівневій структурі транзакцій. Будь-яке звернення до пристрою має суворо дотримуватися такої послідовності:

1. Ініціалізація: Формування сигналу скидання (Reset) ведучим та отримання імпульсу присутності (Presence) від веденого.
2. Команда ROM: Ідентифікація конкретного датчика на шині (наприклад, звернення за унікальним 64-бітним серійним номером).
3. Функціональна команда: Виконання безпосередніх операцій (запуск вимірювання температури, читання пам'яті тощо).

Порушення цього циклу призводить до переходу датчика в режим очікування, за винятком спеціалізованих команд пошуку (Search ROM), після яких алгоритм автоматично повертається до першого кроку. На Рис.3.23 відображено початковий етап будь-якої транзакції – цикл “Скидання – Присутність” (Reset/Presence).

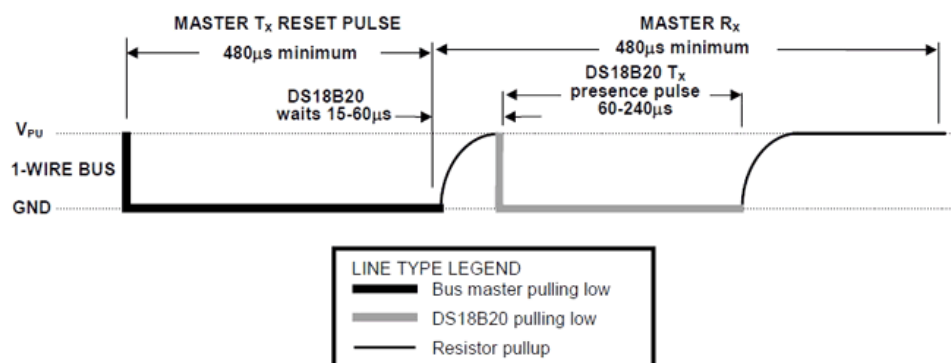


Рис. 3.23. Часова діаграма процедури ініціалізації шини 1-Wire

Алгоритм ініціалізації починається з формування ведучим пристроєм сигналу скидання: лінія DQ утримується в низькому стані щонайменше 480 мкс. Після завершення цього інтервалу шина звільняється, і підтягувальний резистор повертає її до високого логічного рівня. Датчик, зафіксувавши наростаючий фронт, витримує паузу в межах 15–60 мкс, після чого генерує імпульс присутності, примусово знижуючи потенціал лінії на період від 60 до 240 мкс. Після відпускання шини датчиком вона знову стабілізується у високому стані. Загальний час циклу від моменту детектування скидання становить не менше 480 мкс.

Процедура зчитування або передачі окремого біта інформації базується на часовому сумуванні станів шини. Master-пристрій ініціює тайм-слот, протягом якого здійснюється детектування логічного рівня на лінії: наявність високого потенціалу інтерпретується як логічна одиниця, а низького – як нуль. Проте фізична реалізація цього процесу суворо регламентована часовими діаграмами, оскільки зчитування має відбуватися у чітко визначений момент всередині кожного часового інтервалу.

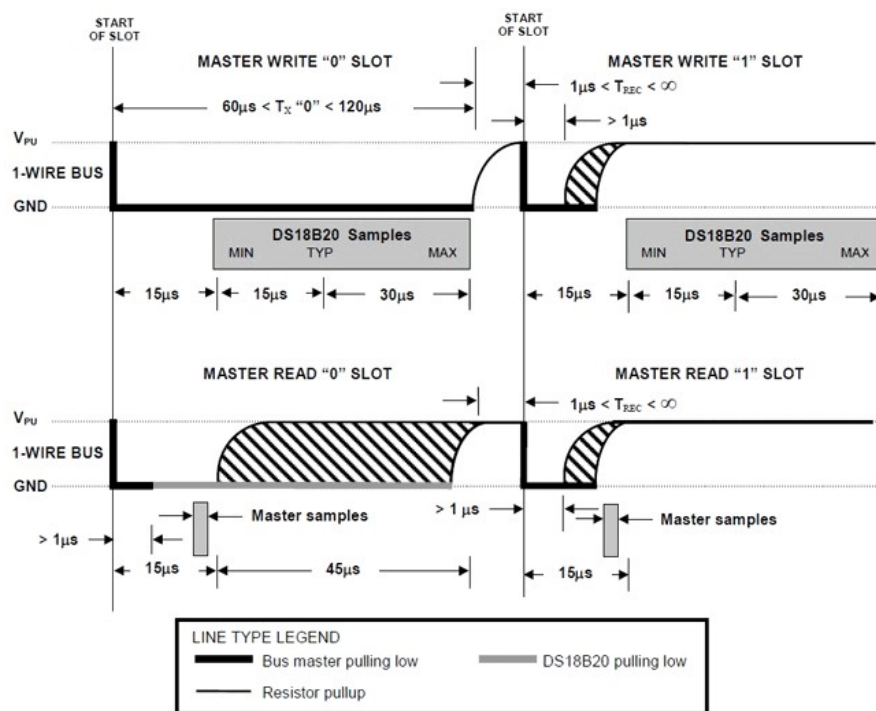


Рис. 3.24. Часові діаграми сигналів при обміні даними між МК та сенсором DS18B20

Фізична реалізація часових слотів запису. Кожна операція передачі біта ініціюється ведучим пристроєм (мікроконтролером) шляхом формування спадаючого фронту на лінії DQ. Повна тривалість часового слота становить від 60 до 120 мкс.

- Запис логічного “0”: ведучий активно утримує низький рівень протягом усього інтервалу слота.
- Запис логічної “1”: контролер генерує короткий імпульс низького рівня (мінімум 1 мкс), після чого переводить порт у стан високого опору, дозволяючи резистору підтяжки повернути лінію до високого стану.

Між послідовними слотами обов’язково передбачається інтервал відновлення шини (t_{REC}) тривалістю не менше 1 мкс.

Процес зчитування даних ініціюється ведучим пристроєм шляхом короткочасного притягування лінії DQ до низького рівня (мінімум 1 мкс). Критичне вікно опитування стану лінії становить 15 мкс від моменту спадаючого фронту. Якщо протягом цього інтервалу датчик утримує низький рівень, контролер фіксує логічний “0”. Якщо ж після звільнення лінії вона повертається до високого стану, передається логічна “1”. При записі даних датчик використовує аналогічне вікно детектування: стан лінії в проміжку 15 – 60 мкс визначає прийняте значення біта.

Після завершення процедури ініціалізації настає етап передачі ROM-команд (Табл.3.10). Ці інструкції призначені для ідентифікації та вибору конкретного пристрою на шині (наприклад, через процедуру пошуку адреси). Кожна така команда має фіксовану довжину 8 біт. Тільки після успішного виконання відповідної ROM-команди стає можливою відправка функціональних команд, що безпосередньо керують роботою обраного датчика.

Таблиця 3.10. ROM-команди інтерфейсу 1-Wire датчика DS18B20

Назва команди	Код (HEX)	Опис та призначення
Search ROM	F0h	Алгоритм ідентифікації унікальних 64-бітних кодів усіх пристроїв на шині.
Read ROM	33h	Зчитування 64-бітного коду датчика (використовується лише при наявності одного пристрою на лінії).

Match ROM	55h	Звернення до конкретного датчика за його унікальним серійним номером.
Skip ROM	CCh	Одночасне звернення до всіх пристроїв на шині (пропуск етапу ідентифікації).
Alarm Search	ECh	Пошук датчиків, у яких зафіксовано перевищення встановлених температурних порогів (Trigger Alarm).

Функціональні команди (Function Commands) призначені для ініціації конкретних робочих процесів сенсора. До них належать операції запуску температурного перетворення, читання або копіювання вмісту внутрішньої пам'яті (EEPROM) та інші сервісні процедури. Набір складається з шести унікальних восьмибітних інструкцій, кожна з яких визначає певну дію пристрою після успішної адресації через ROM-команди, Табл.3.11.

Таблиця 3.11. Перелік та логіка виконання функціональних команд DS18B20

Команда	HEX	Послідуючі дії в циклі обміну
Convert T	44h	МК опитує стан шини (0 – триває перетворення, 1 – завершено).
Write Scratchpad	4Eh	МК передає 3 байти даних у датчик.
Read Scratchpad	BEh	МК генерує слоти читання для прийому до 9 байтів від датчика.
Copy Scratchpad	48h	Затримка або утримання високого рівня на шині (при паразитному живленні).
Recall E2	B8h	Датчик передає статус завершення операції копіювання.
Read Power Supply	B4h	Датчик видає один біт інформації про стан лінії живлення.

Особливості застосування інтерфейсу 1-Wire у системах моніторингу.
Однопровідний інтерфейс 1-Wire, розроблений корпорацією Dallas Semiconductor, є де-факто стандартом для побудови розподілених сенсорних мереж з низькою швидкістю передачі даних. Історично технологія була орієнтована на чотири ключові вектори: персональну ідентифікацію (пристрої iButton у корпусах MicroCAN), конфігурування вбудованої пам'яті інтегральних схем, захист інтелектуальної власності та створення розгалужених систем автоматизації. У даному проекті використання 1-Wire обумовлено можливістю підключення великої кількості датчиків DS18B20 до одного порту мікроконтролера, що мінімізує апаратні витрати.

Основними перевагами інтерфейсу 1-Wire є: спрощена ідентифікація пристроїв за рахунок індивідуальних адрес; програмна та апаратна простота реалізації протоколу; використання двопровідної фізичної лінії

(сигнал+земля); підтримка режиму паразитного живлення; висока адаптивність до змін топології мережі; можливість роботи на значних відстанях; оптимальна вартість впровадження.

Інтегральний перетворювач абсолютного тиску MPX4115. Сенсор серії MPX4115 виробництва Freescale Semiconductor (нині NXP) являє собою високоточний кремнієвий сенсор, виконаний за монолітною технологією. Архітектура пристрою на одному кристалі об'єднує чутливий елемент, схеми температурної компенсації та операційні підсилювачі для нормалізації вихідного сигналу. Сенсор спеціально оптимізований для вимірювання абсолютного атмосферного тиску в діапазоні 15...115 кПа, що робить його ефективним рішенням для застосування у цифрових барометрах, висотомірах та метеорологічних станціях.

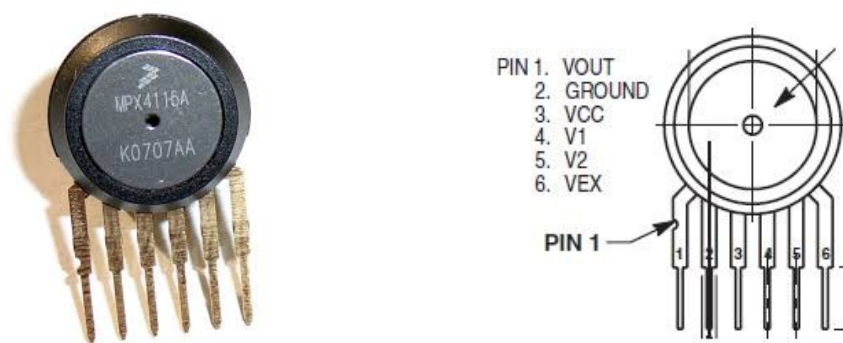


Рис. 3.25. Зовнішній вигляд та розпіновка сенсора атмосферного тиску MPX4115 (Freescale Semiconductor)

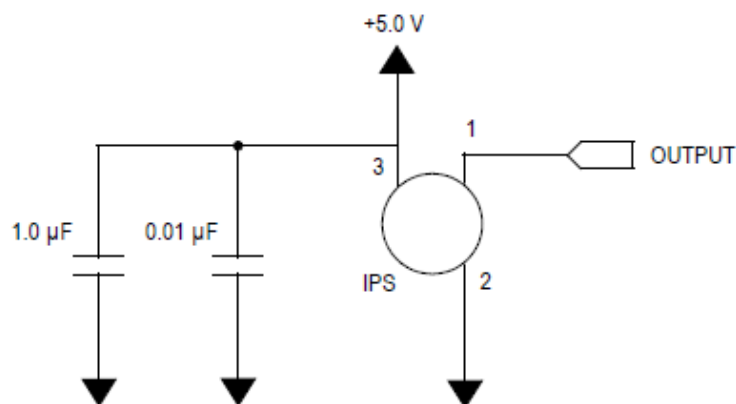


Рис. 3.26. Електрична схема включення інтегрального сенсора тиску MPX4115

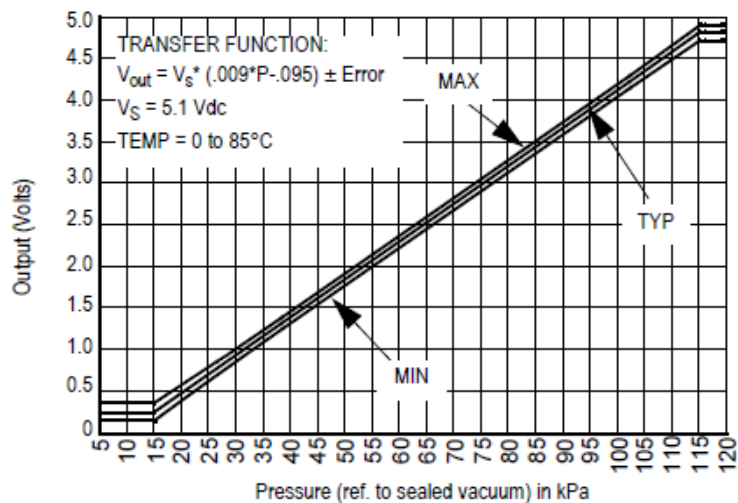


Рис. 3.27. Передавальна характеристика датчика тиску MPX4115: залежність вихідної напруги від величини абсолютного тиску

У Табл.3.12 наведено основні технічні параметри сенсора MPX4115.

Таблиця 3.12 Основні технічні параметри перетворювача тиску MPX4115

Параметр	Значення
Діапазон вихідної напруги	0,2 ... 4,8 В
Чутливість (Sensitivity)	46 мВ/кПа
Похибка (0...85°C)	≤1,5 %
Час відгуку (Response time)	1 мс
Струм споживання (тип./макс.)	7/10 мА
Напруга живлення (U_{CC})	4,85 ... 5,35 В
Робоча температура	-40 ... +125 °C

Оскільки датчик аналоговий, формула перерахунку (transfer function) має вигляд:

$$V_{out} = V_s \times (0,009 \times P - 0,095) \pm \text{Error} \quad (3.1)$$

Де V_s – напруга живлення (5 В), P – тиск у кПа.

Вихідний сигнал датчика подається на вхід 10-бітного АЦП мікроконтролера ATmega32, що забезпечує достатню дискретність вимірювання. Типова чутливість датчика складає 45.9 мВ/кПа, що дозволяє фіксувати навіть незначні зміни висоти (близько 1 метра).

Застосування символічних РК-дисплеїв для відображення даних.
 Використання алфавітно-цифрових РК-модулів є економічно виправданим рішенням, яке дозволяє скоротити час на розробку та мінімізувати апаратні

витрати. Такі пристрої поєднують у собі здатність до виводу великих обсягів текстової інформації з наднизьким енергоспоживанням. Наявність інтегрованого підсвічування гарантує чітке зчитування даних у сутінках або за повної відсутності світла. Завдяки розширеному температурному діапазону (від -20°C до $+70^{\circ}\text{C}$) ці модулі є стабільними у роботі в складі бортових систем, переносних приладів та обладнання, що експлуатується в польових умовах.

Архітектура контролера HD44780 дозволяє обслуговувати до 80 знаків, розподілених на два рядки (по 40 символів у кожному). У випадках, коли модуль має конфігурацію 4x40, управління здійснюється парою ідентичних чипів. Стандартною для даного контролера є матриця символів розміром 5x7 пікселів. Хоча розробниками закладена підтримка формату 5x10, сучасні РК-модулі з такою роздільною здатністю символів на ринку майже не представлені.

На сучасному ринку представлений широкий спектр стандартних конфігурацій РК-модулів, серед яких найбільш вживаними є формати 8x2, 16x1, 16x2, 16x4, 20x1, 20x2, 20x4, 24x2, 40x2 та 40x4 (кількість символів у рядку). Також існують специфічні варіанти, такі як 8x1, 12x2 або 32x2. Важливо зазначити, що архітектура контролера не встановлює жорстких лімітів на кількість знакомісць — технічно модуль може підтримувати будь-яку комбінацію обсягом від 1 до 80 символів, залежно від потреб проектного пристрою.

Конструктивно РК-модуль LM016L об'єднує в собі рідкокристалічну панель та спеціалізований контролер керування. Останній є функціональним аналогом поширених чипів HD44780 (Hitachi) та KS0066 (Samsung), що спрощує процес програмної розробки. Дана модель обладнана LED-підсвіткою та розрахована на вивід двох рядків тексту по 16 знаків у кожному. Формування символів відбувається за допомогою матриці 5x8 точок, при цьому для покращення читабельності між сусідніми знакомісцями передбачено технологічний інтервал шириною в один піксель.

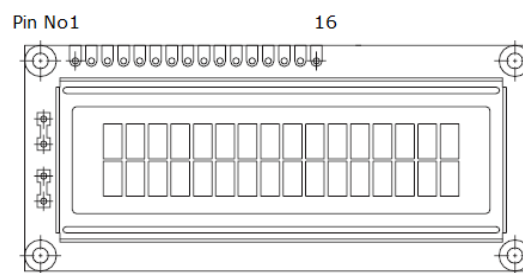


Рис. 3.28. Зовнішній вигляд символьного РК-індикатора формату 16x2
(модель WH1602B-YGK-CTK)

Для виводу інформації модуль використовує відповідність між кодом символу та його розташуванням у внутрішній пам'яті. Система керування включає блоки пам'яті для стандартних символів і знакогенератор для індивідуальних налаштувань користувача. Всі операції з формування сигналів для РК-матриці керує внутрішня логічна схема модуля.

Вибраний рідкокристалічний модуль характеризується високою функціональністю. Зокрема, архітектура контролера дозволяє працювати з двома кодовими сторінками (кирилиця/латиниця) та підтримує два типи інтерфейсів передачі даних (4 та 8 біт). Важливою перевагою є можливість повноцінного зворотного зв'язку: розробник може не лише записувати дані в пам'ять дисплея, а й зчитувати їх або перевіряти статус готовності пристрою. Для розширення графічних можливостей передбачено створення 8 власних символів. Візуальний комфорт забезпечується вибором типу курсора та регулюванням оптичних параметрів панелі (контрастності та інтенсивності підсвічування).

Програмування і управління РК-індикатором. Перш ніж переходити до алгоритмів керування РК-індикатором, доцільно проаналізувати архітектуру контролера HD44780. Розуміння внутрішньої будови чипа є ключовим для освоєння принципів функціонування модулів на його базі. Варто зауважити, що такі вузли, як знакогенератор, формувач курсора, драйвери та регістри зсуву, не потребують прямого втручання з боку керуючої програми. Їхня роль полягає в автоматизації процесу регенерації та візуалізації зображення на фізичній панелі.

Програмне управління контролером здійснюється через звернення до регістрів команд (IR) та даних (DR). Вибір між ними контролюється лінією RS: значення “0” адресує запити до командного регістра, а “1” – до регістра даних. Введені через DR дані записуються у відповідні сектори пам'яті (DDRAM/CGRAM) за адресою, яку в реальному часі формує внутрішній лічильник AC. Це забезпечує послідовне заповнення дисплея або оновлення бази користувацьких символів.

Дані з регістра IR обробляються блоком виконання команд, тоді як коди для відображення зберігаються у 80-байтовій відеопам'яті. Логічно ця пам'ять завжди розділена на два сегменти по 40 байтів, незалежно від фактичної кількості рядків на екрані пристрою. Це означає, що адресація в модулях форматів 20x4 чи 80x1 все одно підпорядковується правилу “2 рядки по 40 символів”. Контролер працює в режимі динамічної індикації, що гарантує постійну актуалізацію зображення на дисплеї без участі зовнішнього процесора.

Фізично РК-індикатор організований як матриця сегментів, конфігурація якої залежить від режиму роботи контролера. Для одного рядка символів (5x7 або 5x10 точок) виділяється відповідно 8 або 11 ліній, тоді як для дворядкового виводу (5x7) матриця налічує 16 ліній. При максимальному заповненні (40 знаків у рядку) кожна лінія об'єднує 200 сегментів, що забезпечує формування чіткого графічного або текстового зображення.

Базових 40 сегментних виходів контролера HD44780 вистачає лише для підтримки 8-символьних індикаторів. Тому в модулях великої інформаційної ємності основний чип доповнюється зовнішніми драйверами типу HD44100, кожен з яких обслуговує додаткові 40 сегментів. Керування роботою всієї структури та її окремих модулів здійснюється через внутрішні прапорці станів, які ініціалізуються керуючою програмою.

Організація взаємодії з РК-модулем. Робота з РК-модулями в середовищі розробки базується на використанні функцій, що містяться у файлі lcd.h. Цей набір інструментів адаптований для дисплеїв під керуванням чипів HD44780

та їх аналогів. Процес інтеграції бібліотеки вимагає попередньої конфігурації: користувач має вказати конкретний порт МК, задіяний для обміну даними. Програмне забезпечення підтримує більшість стандартних типорозмірів РКІ, зокрема популярні формати 16x2, 20x4 та інші варіації (від 8 до 80 знакомісць загалом).

3.2. Проектування цифрового пристрою для планування тривалих чергувань та оцінки стану особового складу в екстремальних умовах

Проектований цифровий пристрій призначений для автоматизації планування чергувань та моніторингу умов довкілля під час виконання завдань особовим складом поліції в екстремальних ситуаціях. Пристрій інтегрує функції хронометража (годинник, будильник) та метеорологічного контролю (термометр, барометр). На дисплей виводиться комплексний масив даних: поточна дата, точний час, атмосферний тиск та температура, а також статус активності будильника (On/Off). Система передбачає наявність меню для конфігурації годинника реального часу та параметрів звукової сигналізації. Структурна схема пристрою (Рис. 3.29) базується на синергії апаратних та програмних засобів.



Рис. 3.29. Блок-схема архітектури багатоцільового цифрового пристрою для координації графіків служби та моніторингу середовища перебування особового складу

Апаратна частина реалізована на базі мікроконтролера AVR ATmega32, який виконує роль центрального процесора. До його периферії підключено

датчики вимірювання фізичних величин, модуль RTC (Real Time Clock) та алфавітно-цифровий РК-дисплей формату 16x2.

На Рис. 3.30 представлено схему електричну принципову апаратної частини багатоцільового цифрового пристрою, розроблену за допомогою засобів САПР Proteus VSM. Згідно зі схемою, центральний процесор ATmega32 взаємодіє з датчиками за допомогою різних типів інтерфейсів. Зокрема, для зчитування температури використано шину 1-Wire (пін PC2), а для синхронізації з годинником реального часу DS1307 — шину I2C (піни PC0, PC1). Вимірювання атмосферного тиску здійснюється шляхом обробки аналогового сигналу з датчика MPX4115 через вбудований аналого-цифровий перетворювач (канал PA3). Візуалізація даних забезпечується РК-індикатором, підключеним до порту А. Керування режимами роботи та налаштування параметрів реалізовано через блок кнопок, що комутуються з молодшими розрядами порту В.

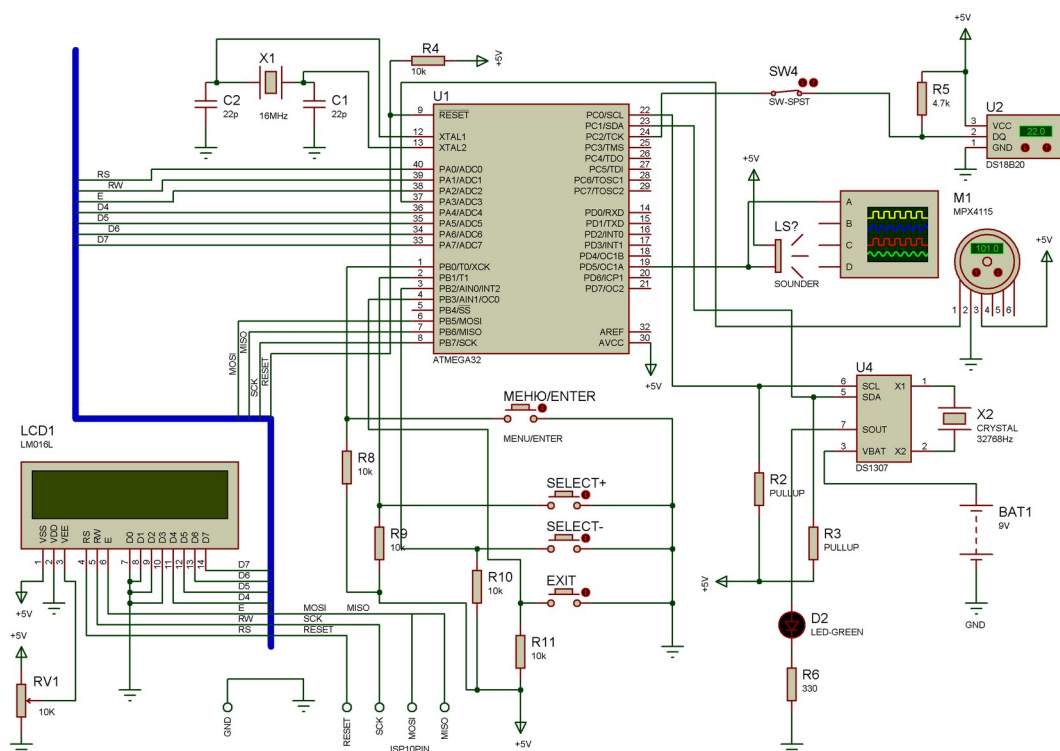


Рис. 3.30. Схема електрична принципова цифрового пристрою, що інтегрує функції метеостанції та системи часової координації чергувань, в середовищі Proteus VSM

РОЗДІЛ 4

ПРОГРАМНО-АЛГОРИТМІЧНЕ ЗАБЕЗПЕЧЕННЯ ЦИФРОВОГО ПРИБОРУ ДЛЯ ПЛАНУВАННЯ ТРИВАЛИХ ЧЕРГУВАНЬ ТА ОЦІНКИ СТАНУ ОСОБОВОГО СКЛАДУ В ЕКСТРЕМАЛЬНИХ УМОВАХ

4.1. Алгоритм роботи цифрового пристрою для планування тривалих чергувань та оцінки особового складу

На Рис.4.1 зображено блок-схему алгоритму роботи цифрового пристрою багатоцільового пристрою для планування тривалих чергувань та оцінки стану особового складу в екстремальних умовах (з функціями годинника-будильника, термометра і барометра) [19].

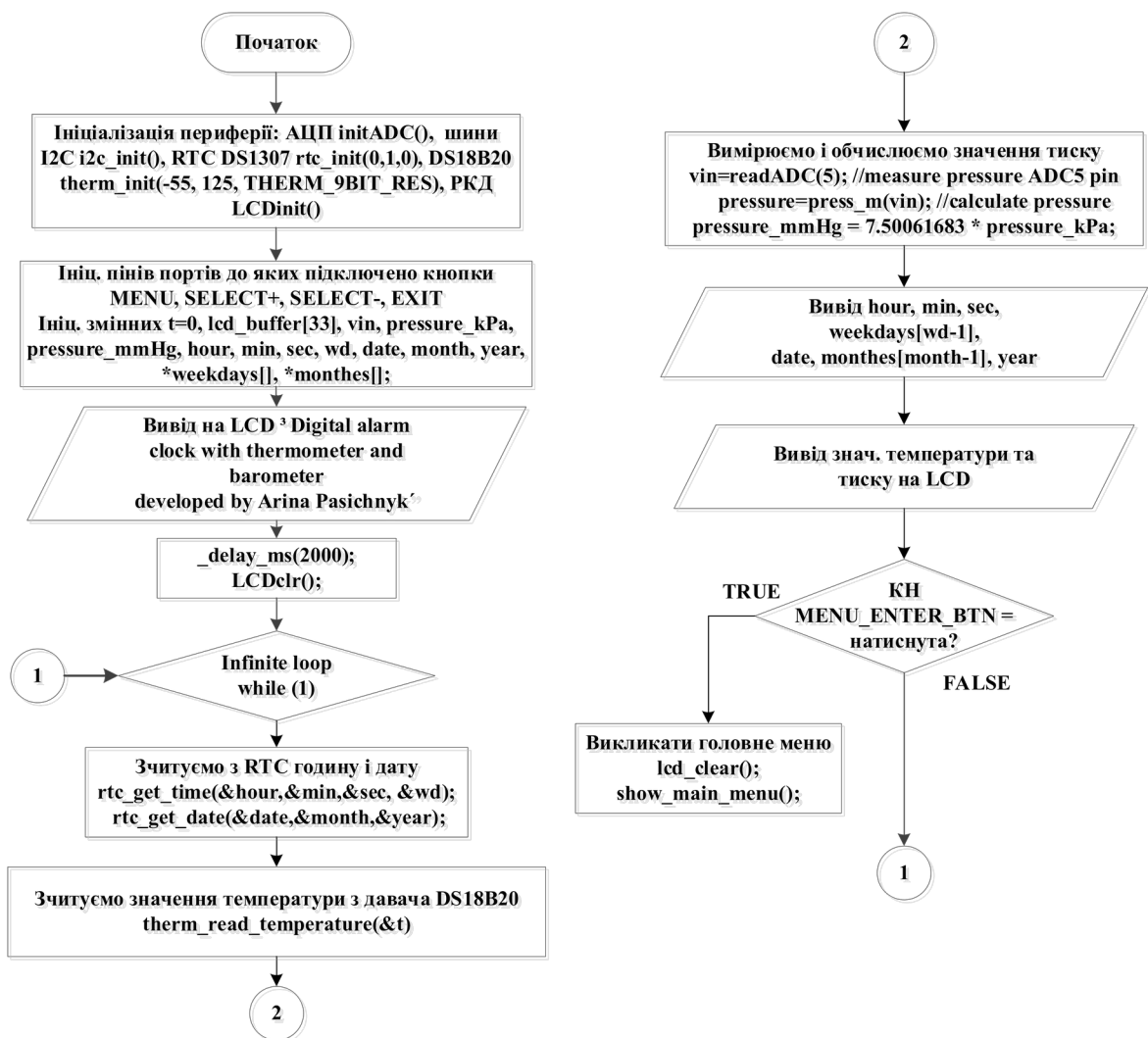


Рис.4.1. Алгоритм роботи багатоцільового цифрового пристрою для планування тривалих чергувань та оцінки стану особового складу в екстремальних умовах



Рис.4.2. Алгоритм функціонування вбудованого меню цифрового пристрою

Процес функціонування пристрою стартує з активації мікроконтролера ATmega32, що викликає виконання програми, яка зберігається в його флеш-пам'яті [20-24]. Алгоритм запуску передбачає два послідовні кроки: спочатку ініціалізуються службові змінні (включаючи числові типи `t`, `vin`, `pressure_kPa`, `pressure_mmHg` та текстові масиви `weekdays[]` і `monthes[]`), а потім здійснюється конфігурація периферійних модулів мікроконтролера та підключеного сенсорного обладнання.

Керування пристроєм реалізовано за допомогою чотирьох тактових кнопок: MENU (МЕНЮ), SELECT+ (ВИБРАТИ ВПЕРЕД), SELECT- (ВИБРАТИ НАЗАД) та EXIT (ВИХІД), які комутуються з лініями 0–3 порту В мікроконтролера. Для їх опитування відповідні виводи конфігуруються як цифрові входи із активацією внутрішніх підтягувальних (pull-up) резисторів до шини живлення: `BTN_DDR &= ~( (1 << MENU_ENTER_BTN) | (1 << SELECT_PLUS_BTN) | (1 << SELECT_MINUS_BTN) | (1 << EXIT_BTN) );` `BTN_PORT |= (1 << MENU_ENTER_BTN) | (1 << SELECT_PLUS_BTN) | (1 << SELECT_MINUS_BTN) | (1 << EXIT_BTN);`.

Конфігурування апаратних модулів передбачає ініціалізацію мікросхеми розкладу реального часу за допомогою функції `rtc_init(0, 1, 0)`. Паралельно здійснюється налаштування цифрового термосенсора DS18B20 через виклик `therm_init(-55, 125, THERM_9BIT_RES)`. Ця підпрограма визначає граничні межі вимірювання температурного діапазону, а також встановлює розрядність аналого-цифрового перетворення (АЦП) датчика. Для гнучкого керування точністю оцифрування у програмному забезпеченні заздалегідь визначено макроси (директиви `#define`), що відповідають за вибір апаратної роздільної здатності сенсора від 9 до 12 біт:

```
#define THERM_9BIT_RES 0 // Точність перетворення: 9 біт (дискретність 0.5°C)
#define THERM_10BIT_RES 1 // Точність перетворення: 10 біт (дискретність 0.25°C)
#define THERM_11BIT_RES 2 // Точність перетворення: 11 біт (дискретність 0.125°C)
#define THERM_12BIT_RES 3 // Точність перетворення: 12 біт (дискретність 0.0625°C)
```

Налаштування алфавітно-цифрового рідкокристалічного дисплея (РКД) забезпечується викликом підпрограми `LCDinit()`. Після успішного завершення конфігурації екрана на нього виводиться вітальне інформаційне повідомлення за допомогою функції `LCDstring("Multifunction Device")`, яке залишається активним протягом двох секунд. У подальшому мікроконтролер переходить до виконання основного робочого циклу. Цей етап передбачає постійне зчитування поточних календарних даних, часових міток, а також показників температури й атмосферного тиску з відповідних

первинних перетворювачів. Отримані результати підлягають математичній обробці, паралельно з чим здійснюється безперервний моніторинг стану тактової кнопки MENU_ENTER_BTN.

У межах основного циклу програми здійснюється аналого-цифрове перетворення сигналу, що надходить від сенсора тиску. Цю операцію виконує підпрограма `vin = readADC(3)`, яка опитує третій канал інтегрованого АЦП (вивід PA3/ADC3). Отримане цифрове значення `vin` передається у функцію `pressure = press_m(vin)` для розрахунку абсолютного тиску в кілопаскалях. Для подальшого переведення обчисленої величини у міліметри ртутного стовпчика використовується лінійне співвідношення:

$$pressure_mmHg = 7,500617 \cdot pressure_kPa$$

Одночасно з цим для реєстрації температурних показників викликається функція `therm_read_temperature(&t)`, яка здійснює читання даних із термосенсора та записує фінальний результат у дійсну змінну `t`.

Отримання актуальних календарних та часових міток із зовнішньої мікросхеми годинника реального часу DS1307 реалізовано за допомогою двох спеціалізованих підпрограм: `rtc_get_time(&hour, &min, &sec, &wd);` `rtc_get_date(&date, &month, &year);`. Перша функція забезпечує зчитування поточного часу й дня тижня, записуючи ці параметри за посиланням у відповідні змінні `hour` (години), `min` (хвилини), `sec` (секунди) та `wd` (день тижня). Друга функція відповідає за реєстрацію поточної дати, зберігаючи числові значення дня, місяця та року в змінних `date`, `month` і `year` відповідно.

Форматування даних для відображення на екрані здійснюється за допомогою стандартної функції `sprintf()`. Вона компонує текстовий рядок на основі поточних значень часу та дати й записує його в масив `lcd_buffer`. Порозрядний поділ числових величин на десятки (ділення на 10) та одиниці (залишок від ділення `%10`) дозволяє отримати фіксований двозначний формат для кожного параметра: `sprintf(lcd_buffer, "%u%u:%u%u %u%u.%u%u.%u%u", hour/10, hour%10, min/10, min%10, date/10, date%10, month/10, month%10,`

year/10, year%10). Візуалізація цього буфера на рідкокристалічному дисплеї реалізується за допомогою двох послідовних команд: функція `LCDGotoXY(0,0)` встановлює курсор на початкову позицію (перше знакомісце першого рядка), після чого підпрограма `LCDstring(lcd_buffer)` виводить сформований масив символів на екран.

Для представлення телеметричних показників навколишнього середовища у текстовому форматі використовується функція `sprintf()`, яка записує відформатований рядок у масив `lcd_buffer`. Специфікатор шаблону реалізує відображення дійсної змінної температури `t` із точністю до одного знака після коми та обов'язковим виведенням знака «плюс» чи «мінус» (`%+.1f`). Шістнадцятковий код `0xdf` відповідає за генерацію символу градуса, після якого виводяться приведені до цілого типу дані атмосферного тиску в міліметрах ртутного стовпчика: `sprintf(lcd_buffer, "%+.1f%cC %ummHg", t, 0xdf, (unsigned int)pressure_mmHg)`. Виведення сформованого інформаційного пакета у другий рядок рідкокристалічного індикатора забезпечується встановленням курсора на початкову позицію нижнього рядка за допомогою функції `LCDGotoXY(0,1)` та подальшим викликанням макросу чи підпрограми `LCDstring(lcd_buffer)`.

Моніторинг стану органів керування та обробка сигналів від тактових клавіш здійснюється за допомогою функцій `get_key_status` та `get_prev_key_status`. Наведена у лістингу логічна конструкція `if` реалізує алгоритм виявлення моменту безпосереднього натискання кнопки `MENU_ENTER_BTN` (фіксація фронту сигналу):

```
if (get_key_status(MENU_ENTER_BTN)) //enter
{
    if (!get_prev_key_status(MENU_ENTER_BTN))
    {
        lcd_clear();
        show_main_menu();
    }
}
```

Якщо поточний стан кнопки зафіксовано як активний, а її попередній стан (збережений у буфері під час минулого опитування) був пасивним, система розпізнає це як новий клік. У результаті виконується повне очищення вмісту рідкокристалічного екрана через виклик `lcd_clear()`, після чого підпрограма `show_main_menu()` ініціює візуалізацію головного вікна користувацького меню.

4.2. Розроблення програмного модуля інтеграції датчика температури DS18B20

Для забезпечення інформаційного обміну з напівпровідниковим термосенсором DS18B20 було створено програмний модуль (бібліотеку) `therm_ds18b20`. Структурно її програмний інтерфейс (API) розділено на кілька функціональних груп: допоміжні низькорівневі процедури, алгоритми верифікації даних та високорівневі функції керування.

Службові функції внутрішнього інтерфейсу:

- `void therm_delay(uint16_t delay)` – забезпечує формування точних часових інтервалів (затримок), необхідних для дотримання специфікації протоколу 1-Wire;
- `uint8_t therm_reset(void)` – генерує імпульс скидання для синхронізації та відновлення зв'язку між обчислювальним ядром МК і термодатчиком;
- `void therm_write_bit(uint8_t bit)` – реалізує побітовий запис (передачу) логічного нуля або одиниці в однопровідну шину DQ;
- `uint8_t therm_read_bit(void)` – здійснює посимвольне (побітове) зчитування поточного стану інформаційної лінії DQ;
- `uint8_t therm_read_byte(void)` — забезпечує послідовний прийом повної 8-бітової одиниці даних (байта) з боку сенсора;
- `void therm_write_byte(uint8_t byte)` — виконує надсилання байта конфігурації або команди по шині DQ.

Контроль цілісності інформації:

- `uint8_t therm_crc8(unsigned char *data, unsigned char num_bytes)` — розраховує 8-бітну циклічну контрольну суму (CRC8) для підтвердження вірогідності прийнятого пакета даних.

Високорівневі функції користувача:

- `uint8_t therm_init(int8_t temp_low, int8_t temp_high, uint8_t resolution)` — конфігурує початкові параметри датчика, встановлюючи порогові значення термостата та апаратну розрядність АЦП;
- `uint8_t therm_read_temperature(float *temp)` — ініціює цикл вимірювання, зчитує вихідний код із внутрішньої пам'яті сенсора, конвертує його та передає за посиланням у дійсну змінну обчисленого значення температури.

4.3. Розроблення програмного модуля обробки даних з сенсора тиску MPX4115

Для програмного спряження аналогового барометричного сенсора MPX4115 із МК ATmega32 розроблено драйвер `mpx4115`. Його архітектура містить функції конфігурування інтегрованого аналого-цифрового перетворювача (АЦП), алгоритми покрокового зчитування напруги та математичні процедури лінійної апроксимації даних. Програмний інтерфейс драйвера складається з таких базових компонентів:

- `void initADC(void)` – здійснює первинне налаштування модулів АЦП, встановлюючи опорну напругу від виводу живлення `AVCC` та задаючи коефіцієнт ділення частоти;
- `unsigned int readADC(unsigned char ch)` – ініціює цикл аналого-цифрового перетворення для обраного аналогового каналу із апаратним очікуванням завершення процесу;
- `unsigned int press_m(unsigned int vin)` – масштабує отриманий двійковий код у напругу, після чого розраховує абсолютне значення атмосферного тиску.

4.4. Розроблення програмних засобів взаємодії з годинником реального часу DS1307

Програмне забезпечення для конфігурування та збору даних з апаратного модуля реального часу базується на розробленій бібліотеці ds1307. Функціонал цього драйвера розділено на ініціалізаційні процедури, а також блоки читання та запису хронометричних і календарних параметрів.

Функція первинного налаштування (ініціалізації):

- `void rtc_init(unsigned char PeriodSelect, unsigned char SQWe, unsigned char OUTlevel)` — забезпечує початкове конфігурування мікросхеми. Параметр `PeriodSelect` визначає частоту вихідного прямокутного сигналу (меандра) на універсальному виводі `SQW/OUT`: 0 – генерація сигналу з частотою 1 Гц; 1 – формування частоти 4096 Гц; 2 – формування частоти 8192 Гц; 3 – формування частоти 32768 Гц. Активація виходу частоти контролюється параметром `SQWe` (при встановленні в 1). Якщо генерацію вимкнено (`SQWe = 0`), параметр `OUTlevel` визначає постійний статичний рівень напруги (логічний нуль або одиницю) на цьому виводі.

Інтерфейс роботи з часовими мітками:

- `void rtc_get_time(unsigned char *hour, unsigned char *min, unsigned char *sec, unsigned char *wd)` – виконує читання поточних значень годин, хвилин, секунд та дня тижня з регістрів мікросхеми;
- `void rtc_set_time(unsigned char hour, unsigned char min, unsigned char sec, unsigned char wd)` – здійснює прямий запис нових значень часу та дня тижня у пам'ять RTC-модуля.

Інтерфейс роботи з календарними даними:

- `void rtc_get_date(unsigned char *date, unsigned char *month, unsigned char *year)` – зчитує поточні параметри числа, місяця та року у відповідні змінні;

- `void rtc_set_date(unsigned char date, unsigned char month, unsigned char year)` – оновлює календарну інформацію (день, місяць, рік) всередині мікросхеми.

4.5. Розроблення драйвера низького рівня для підключення алфавітно-цифрового РКД-модуля

Візуалізація текстової та цифрової інформації на алфавітно-цифровому рідкокристалічному індикаторі формату 16×2 (два рядки по 16 знакомісць у кожному) реалізується за допомогою програмних засобів розробленої бібліотеки `lcd`. Створений прикладний інтерфейс (API) містить у собі повний набір функцій для низькорівневого керування дисплеєм:

Функції керування станом та ініціалізації:

- `void LCDinit(void)` – виконує первинне апаратне налаштування РКД, задає розрядність інтерфейсу обміну даними, активує роботу знакогенератора та вмикає курсор;
- `void LCDclr(void)` – здійснює повне очищення візуального поля екрана шляхом запису символів пробілу в усі комірки пам'яті DDRAM та повертає курсор у початкову позицію.

Функції позиціонування та виведення даних:

- `void LCDGotoXY(unsigned char x, unsigned char y)` – забезпечує переміщення віртуального курсора виведення у задану точку координатної сітки дисплея, де параметр `x` визначає номер стовпця (від 0 до 15), а `y` – номер рядка (0 або 1);
- `void LCDsendChar(char chr)` – відповідає за посимвольний запис та відображення одного ASCII-символу у поточному знакомісці екрана;
- `void LCDstring(char* str)` – реалізує послідовне виведення повного текстового масиву (рядка), переданого за посиланням за допомогою покажчика на буфер пам'яті.

4.6. Програмна реалізація алгоритму роботи цифрового пристрою для планування тривалих чергувань та оцінки особового складу

Основою програмного забезпечення розробленого цифрового пристрою є концепція наскрізного циклічного опитування периферійних вузлів та інтерактивної взаємодії з користувачем через багаторівневе меню. Архітектура вбудованого ПЗ розділена на дві ключові взаємопов'язані частини: головну функцію системи `main()`, що виконує диспетчеризацію основних завдань вимірювання, та функцію `main_menu()`, яка відповідає за конфігурування внутрішніх констант і режимів роботи. Для забезпечення інформативності інтерфейсу та розширення графічних можливостей символної матриці розроблено бітовий масив `termometru`. Він містить призначений для користувача графічний символ (іконку термометра), який завантажується у пам'ять знакогенератора дисплея (CGRAM).

Лістинг програмної реалізації головного алгоритму роботи цифрового пристрою для планування тривалих чергувань та оцінки особового складу в екстремальних умовах наведено у додатку.

4.7. Моделювання цифрового пристрою для планування тривалих чергувань та оцінки стану особового складу в екстремальних умовах в Proteus ISIS

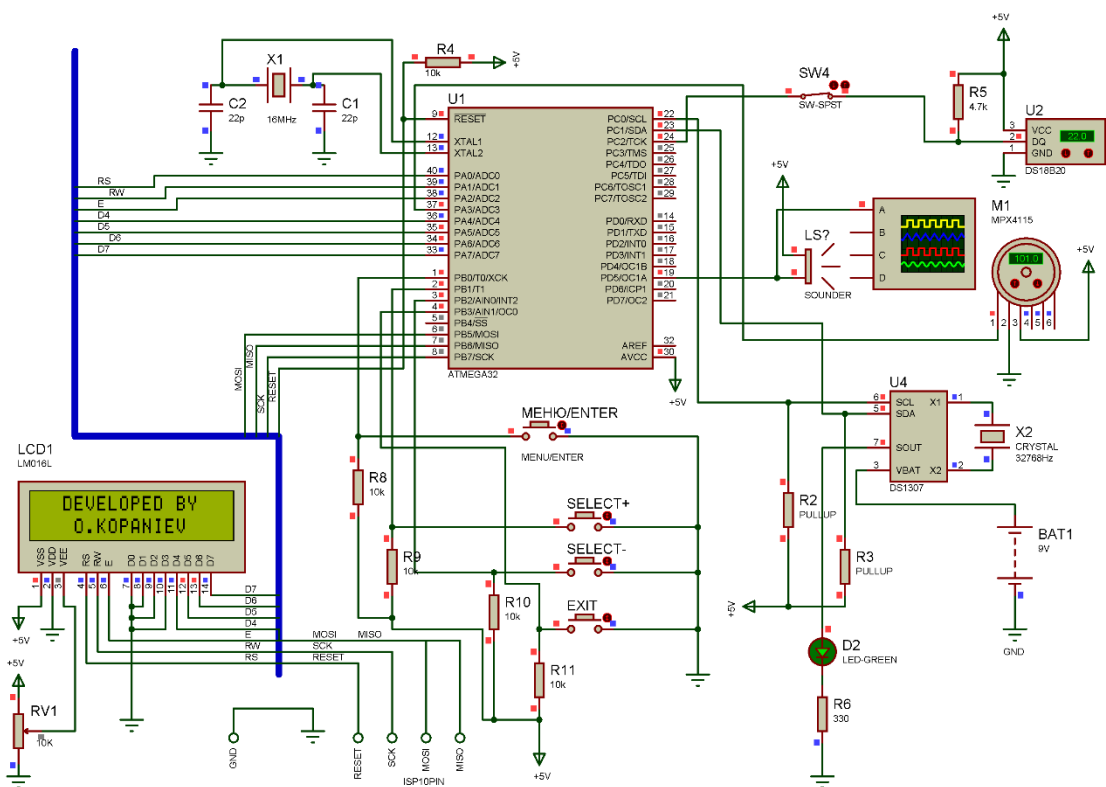
Результатом компіляції розробленого програмного забезпечення для мікроконтролерів архітектури AVR є файл із розширенням `.hex` (або `.cof/.elf` для налагодження), який згодом завантажується в енергонезалежну пам'ять обчислювального ядра. Верифікація багатофункціонального цифрового комплексу, призначеного для оптимізації графіків тривалих чергувань та оцінки психофізіологічного стану особового складу підрозділів поліції під час виконання завдань в екстремальних умовах, здійснювалася в середовищі програмного моделювання Proteus ISIS (VSM).

Пакет ПЗ Proteus VSM (Virtual System Modeling) є високоефективним інструментом для наскрізного проектування, симуляції та налагодження

складних мікропроцесорних систем. Функціональні можливості середовища дозволяють одночасно моделювати роботу декількох обчислювальних ядер, зокрема різних архітектурних родин, у межах єдиного віртуального пристрою.

Середовище комп'ютерного моделювання Proteus VSM є повнофункціональною системою автоматизованого проектування електронних компонентів та схемотехнічних рішень. Ключовою особливістю Proteus VSM є можливість високоточного інтерактивного моделювання цифрових пристроїв із вбудованим програмним керуванням, зокрема на базі сучасних мікроконтролерів.

Результати комп'ютерного моделювання та верифікації алгоритмів функціонування розробленого цифрового пристрою представлені на наведених нижче графічних матеріалах



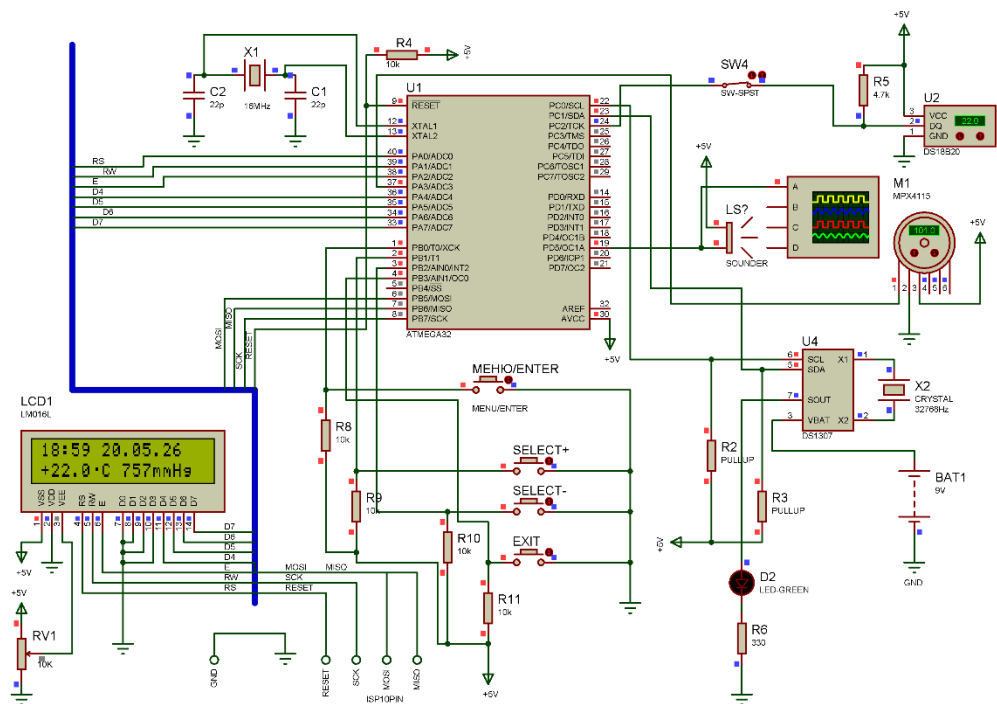
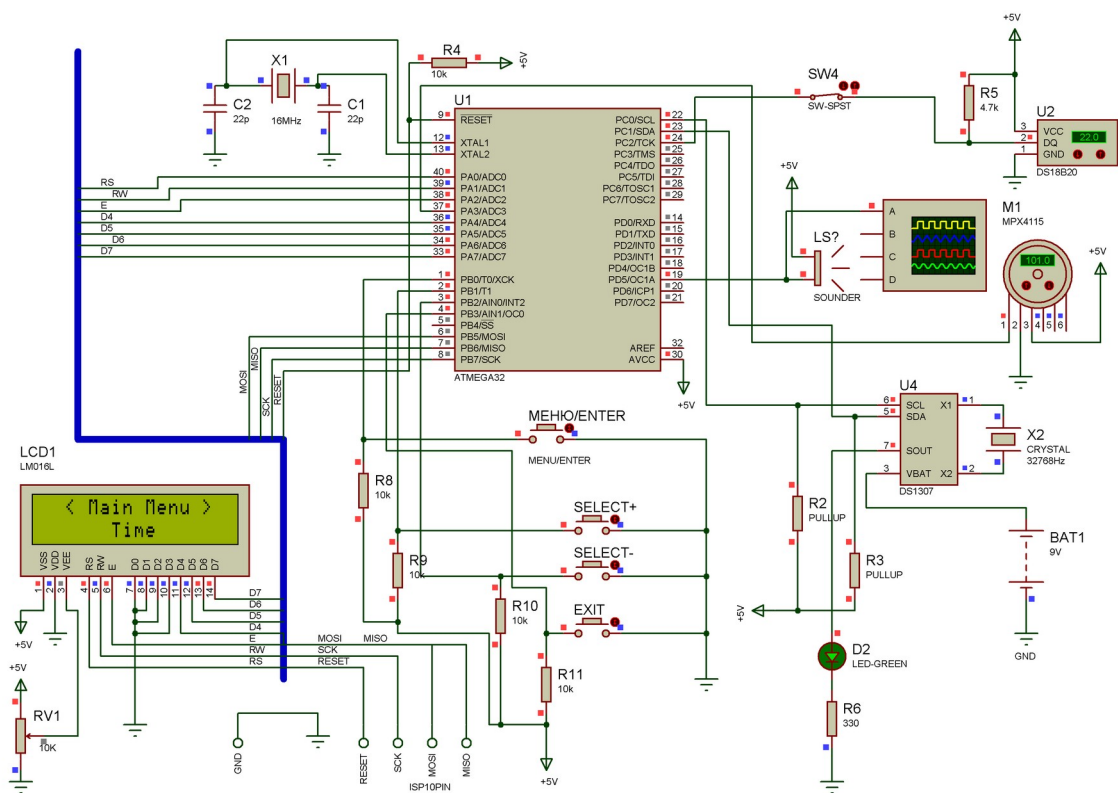


Рис.4.3. Моделювання роботи цифрового пристрою для планування тривалих чергувань та оцінки стану особового складу в екстремальних умовах в емуляторі Proteus ISIS



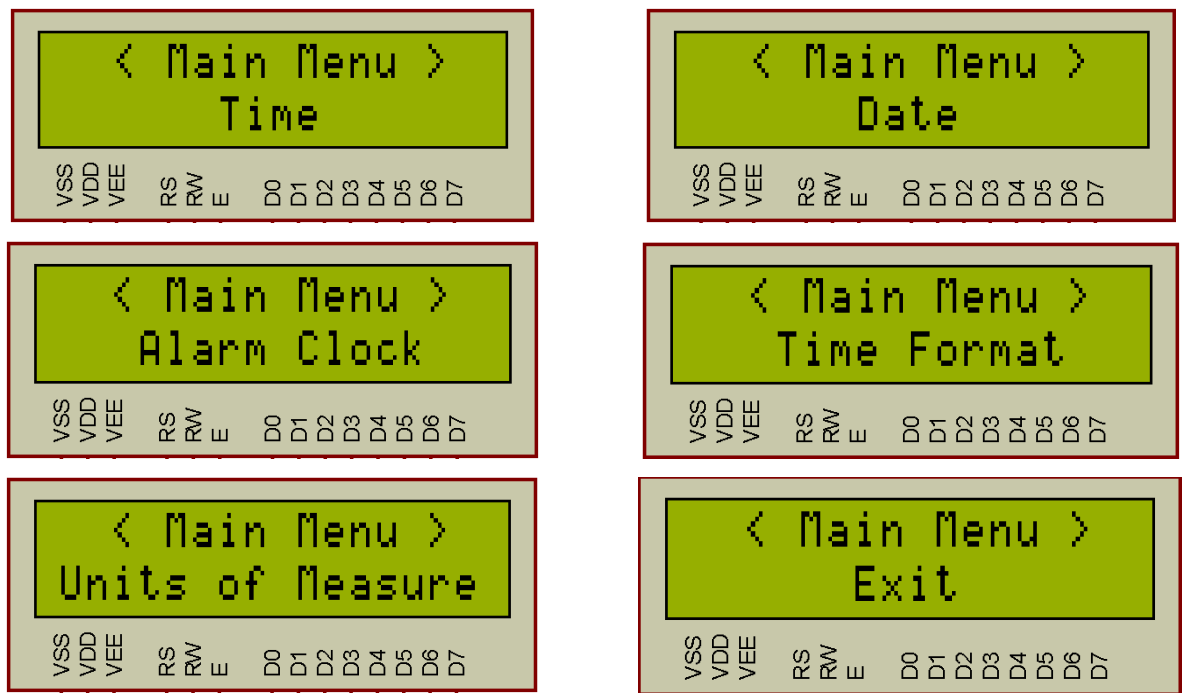


Рис. 4.4. Моделювання роботи цифрового пристрою. Головне меню; Time (час), Date (Дата), Alarm Clock (будильник), Time Format (формат часу), Unit of Measure (одиниці вимірювання), Exit (вихід)

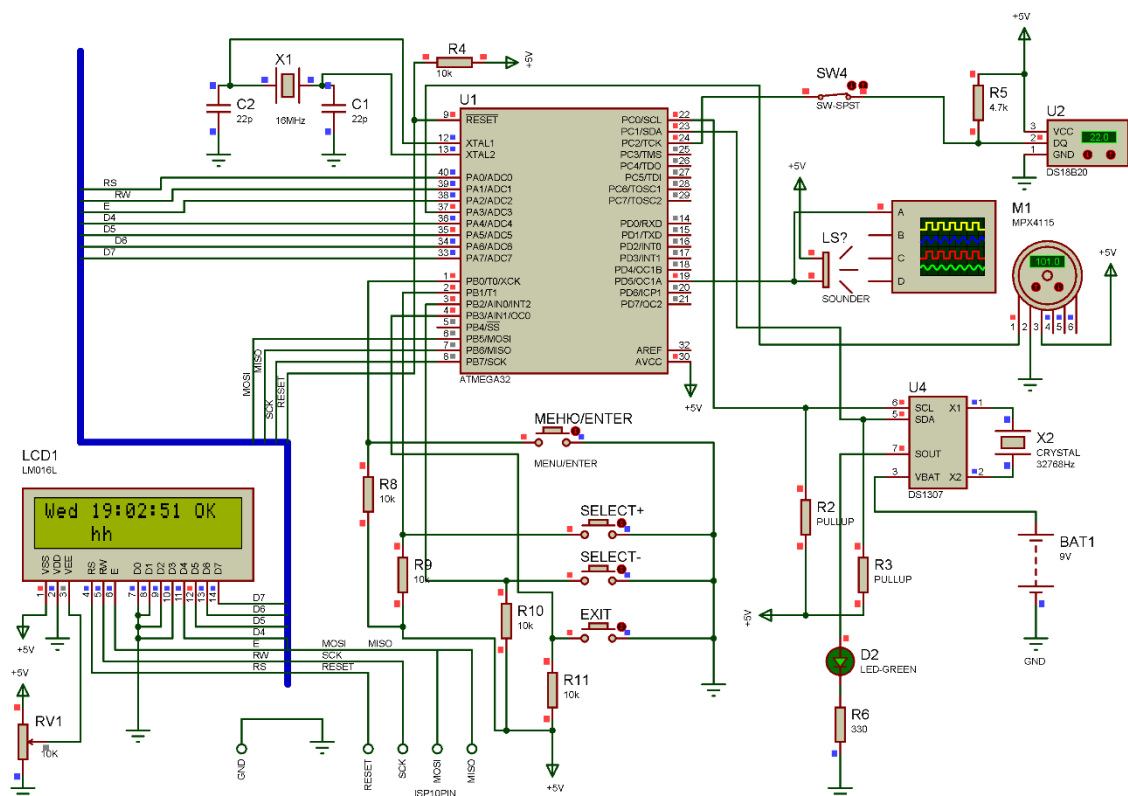


Рис. 4.5. Моделювання роботи цифрового пристрою. Встановлення часу

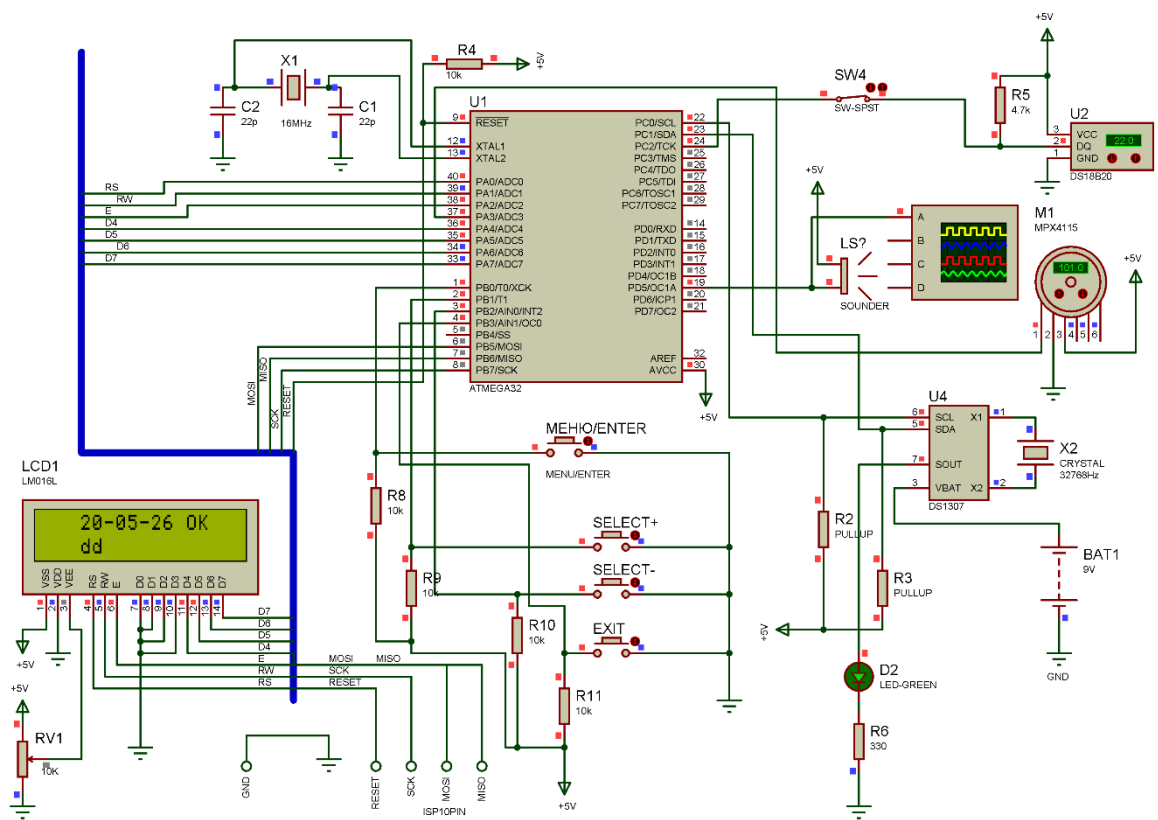


Рис. 4.6. Моделювання роботи цифрового пристрою для планування тривалих чергувань та оцінки стану особового складу в екстремальних умовах. Встановлення дати

ВИСНОВКИ

У рамках кваліфікаційної роботи спроектовано та реалізовано багатофункціональний цифровий пристрій на базі мікроконтролера архітектури AVR. Пристрій розроблено для оптимізації планування чергувань та контролю умов перебування особового складу поліції під час виконання завдань в екстремальних середовищах. Функціонал системи поєднує можливості метеостанції (вимірювання температури та атмосферного тиску за допомогою сенсорів DS18B20 і MPX4115) та органайзера. Координація часових параметрів здійснюється через модуль реального часу DS1307. Візуалізація даних, включаючи поточну дату, час та показники довкілля, реалізована на символьному РК-індикаторі формату 16x2. Передбачено інтерактивний інтерфейс користувача для конфігурації часових констант, параметрів будильника та режимів відображення.

Для побудови багатофункціонального цифрового пристрою застосовано сучасну елементну базу з високим рівнем інтеграції. Архітектурне ядро системи базується на МК сімейства AVR, що забезпечує необхідну обчислювальну потужність. Часову синхронізацію та ведення календаря покладено на модуль RTC DS1307. Вимірювальний тракт сформовано на основі цифрового термометра DS18B20 та аналогового датчика абсолютного тиску MPX4115. Візуалізація результатів вимірювань та службових повідомлень здійснюється за допомогою символьного РК-дисплея формату 16x2, побудованого на базі контролера HD44780.

Виконано повний цикл проектування цифрового пристрою: від створення електричної принципової схеми до розробки ПЗ. Прошивка МК AVR реалізована мовою C у середовищі WinAVR. Програмна архітектура складається з окремих функціональних модулів, що забезпечують взаємодію з термосенсором, барометром MPX4115, годинником RTC DS1307 та РК-дисплеєм. Верифікацію розроблених алгоритмів проведено шляхом симуляції в оболонці Proteus ISIS. Отримані результати підтвердили адекватність програмної реалізації та стабільність роботи всіх складових системи.

СПИСОК ВИКОРИСТАНИХ ЛІТЕРАТУРНИХ ДЖЕРЕЛ

1. ATmega32 Microcontroller. Data Sheet [Електронний ресурс] / Microchip Technology Inc. – 2024. – Режим доступу: <https://ww1.microchip.com/downloads/en/DeviceDoc/doc2503.pdf>.
2. AVR Instruction Set Manual. Datasheet [Електронний ресурс] / Microchip Technology. – 2021. – 166 p.
3. AN2519: AVR Microcontroller Hardware Design Considerations [Електронний ресурс] / Microchip Technology. – 2017. – 27 p.
4. DS1307 Serial I2C Real-Time Clock. Datasheet [Електронний ресурс] / Maxim Integrated / Dallas Semiconductor. – Режим доступу: <https://cdn.sparkfun.com/assets/5/3/1/1/3/DS1307.pdf>.
5. DS18B20 Programmable Resolution 1-Wire Digital Thermometer [Електронний ресурс] / Dallas Semiconductor. – Режим доступу: https://teslabs.com/openplayer/docs/docs/other/ds18b20_pre1.pdf.
6. MPX4115 Series Integrated Silicon Pressure Sensor. Data Sheet [Електронний ресурс] / NXP Semiconductors. – Режим доступу: <https://www.nxp.com/docs/en/data-sheet/MPX4115.pdf>.
7. Padrue J. C Programming for microcontrollers: Featuring ATMEL's AVR Butterfly and the free WinAVR Compiler. – Smiley Micros Publisher, 2005. – 304 p.
8. Barrett S. F., Pack D., Thornton M. Atmel AVR Microcontroller Primer: Programming and Interfacing. – Morgan & Claypool Publishers, 2007. – 180 p.
9. Mazidi M. A., Naimi S., Naimi S. The AVR Microcontroller and Embedded Systems: Using Assembly and C. – Pearson, 2015. – 752 p.
10. Williams E. Make: AVR Programming. Learning to Write Software for Hardware. – Maker Media, Inc, 2014. – 472 p.
11. White E. Making Embedded Systems: Design Patterns for Great Software. – 2nd ed. – O'Reilly Media, 2024. – 425 p.
12. Boxall J. AVR Workshop: A Hands-On Introduction with 60 Projects. – No Starch Press, 2022. – 368 p.

13. Bartlett J. Electronics for Beginners: A Practical Introduction to Schematics, Circuits, and Microcontrollers. – Apress, 2020. – 537 p.
14. Barnett R. H., Cox S. A., O’Cull L. Embedded C Programming and the Atmel AVR. – 2nd ed. – Cengage Learning, 2006. – 560 p.
15. Proteus VSM. Schematic Capture and PCB Design Software [Электронный ресурс] / Labcenter Electronics Ltd. – Режим доступа: <https://www.labcenter.com/>.
16. WinAVR Toolchain for AVR Microcontrollers [Электронный ресурс] / SourceForge. – Режим доступа: <https://winavr.sourceforge.net/>.
17. Date and time representation by country [Электронный ресурс] / Wikipedia. – Режим доступа: https://en.wikipedia.org/wiki/Date_and_time_representation_by_country.
18. Interfacing RTC DS1307 with ATmega16/32 [Электронный ресурс] / Electronic Wings. – Режим доступа: <https://www.electronicwings.com/avr-atmega/real-time-clock-rtc-ds1307-interfacing-with-atmega16-32>.
19. AVR Library for HD44780 based LCD [Электронный ресурс] / Davide Gironi Software Blog. – Режим доступа: <https://davidegironi.blogspot.com/2013/06/an-avr-atmega-library-for-hd44780-based.html>.

ДОДАТКИ

Додаток 1

Лістинг головного програмного модуля цифрового пристрою для планування тривалих чергувань та оцінки стану особового складу в екстремальних умовах

```
/******  
Project : Digital device for planning long shifts and assessing the condition of personnel  
(Цифровий пристрій для планування тривалих чергувань та оцінки стану особового  
складу)  
Date   : 2026  
Author : (C) Oleksandra Kopaniev  
*****/  
#include <avr/io.h>  
#include <util/delay.h>  
#include <stdio.h>  
#include "lcd.h"  
#include "mpx4115.h"  
#include "ds1307.h"  
#include "therm_ds18b20.h"  
  
/* Глобальні змінні для зберігання хронометричних даних */  
unsigned char hour, min, sec, wd; unsigned char date, month, year;  
char lcd_buffer[32];  
  
/* Масив власного символу термометра для CGRAM дисплея */  
const uint8_t termometru[8] = {  
    0b00100, // *  
    0b01010, // * *  
    0b01010, // * *  
    0b01110, // ***  
    0b01110, // ***  
    0b11111, // *****  
    0b11111, // *****  
    0b01110 // ***  
};  
  
/* Модуль обробки та візуалізації головного меню користувача */  
void main_menu(void) {  
    char *menu_items[] = {"   Time   ", "   Date   ", " Alarm Clock ",  
        " Time Format ", "Units of Measure", "   Exit   " };  
    char menu_title[] = "< Main Menu >"; unsigned char selected = 0;  
    LCDclr();  
    while(1) {  
        PREV_BTN_PIN = BTN_PIN;  
        // Виведення заголовка та активного пункту меню  
        LCDGotoXY(2, 0); LCDstring(menu_title); LCDGotoXY(0, 1);  
        LCDstring(menu_items[selected]);  
        // Обробка навігації «Вперед / Плюс»  
        if(get_key_status(SELECT_PLUS_BTN)) {  
            if(!get_prev_key_status(SELECT_PLUS_BTN)) {  
                if(selected == 5) selected = 0;  

```

```

        else selected++;    }    }
// Обробка навігації «Назад / Мінус»
if(get_key_status(SELECT_MINUS_BTN)) {
    if(!get_prev_key_status(SELECT_MINUS_BTN)) {
        if(selected == 0) selected = 5;
        else selected--;    }    }
// Вибір активного пункту меню
if(get_key_status(MENU_ENTER_BTN)) {
    if(!get_prev_key_status(MENU_ENTER_BTN)) {
        switch(selected) {
            case 0: set_time();    break;
            case 1: set_date();    break;
            case 2: /* Функціонал будильника */ break;
            case 3: /* Формат часу */    break;
            case 4: set_units_format(); break;
            case 5: LCDclr();    return; // Вихід з меню
            default: break;    }
        LCDclr();    }    }
// Екстрений вихід з меню за допомогою виділеної клавіші
if(get_key_status(EXIT_BTN)) {
    if(!get_prev_key_status(EXIT_BTN)){PREV_BTN_PIN = BTN_PIN; LCDclr();
return;    }    } }}

/* Головна керуюча функція пристрою */
int main(void){
    float t = 0; unsigned int vin, pressure_kPa, pressure_mmHg;
    // Налаштування таймера T1 для генерації звукових сповіщень (ШИМ)
    DDRB |= (1 << PORTB1); TCCR1A = 0xA1; OCR1A = 128; TCCR1B = 0x05;
    /* Конфігурування напрямку ліній порту для роботи з кнопками */
    BTN_DDR  &= ~(1 << MENU_ENTER_BTN) & ~(1 << SELECT_PLUS_BTN) & ~(1 <<
SELECT_MINUS_BTN) & ~(1 << EXIT_BTN);
    BTN_PORT |= (1 << MENU_ENTER_BTN) | (1 << SELECT_PLUS_BTN) | (1 << SELECT_MINUS_BTN) |
(1 << EXIT_BTN);
    // Апаратна ініціалізація периферійних модулів МК та шин обміну
    initADC();i2c_init();  rtc_init(0, 1, 0); therm_init(-55, 125, THERM_9BIT_RES);
    /* Налаштування параметрів рідкокристалічного дисплея */
    LCDinit(); LCDcursorOFF();
    // Виведення стартового інформаційного вікна розробника
    LCDGotoXY(3,0); LCDstring("PLANNING &"); LCDGotoXY(3,1); LCDstring("ASSESSING");
    _delay_ms(500); LCDclr(); LCDGotoXY(3,0); LCDstring("CONDITION");
    LCDGotoXY(1,1); LCDstring("OF PERSONNEL"); _delay_ms(500);
    LCDclr(); LCDGotoXY(2,0); LCDstring("DEVELOPED BY"); LCDGotoXY(3,1);
    LCDstring("O.KOPANIEV"); _delay_ms(2000); _delay_ms(2000); LCDclr();
    /* Головний нескінченний робочий цикл системи */
    while(1) {
        // Збір аналогових та цифрових телеметричних даних
        vin = readADC(3); pressure_kPa = press_m(vin);
        // Математичний перерахунок кПа в міліметри ртутного стовпчика
        pressure_mmHg = 7.50061683 * pressure_kPa;
        // Зчитування показників з датчиків температури та часу
        therm_read_temperature(&t); rtc_get_time(&hour, &min, &sec, &wd);
        rtc_get_date(&date, &month, &year);
        // Формування та виведення верхнього інформаційного рядка (Час та Дата)

```

```
    sprintf(lcd_buffer, "%02u:%02u %02u.%02u.%02u", hour, min, date, month, year);
    LCDGotoXY(0, 0); LCDstring(lcd_buffer);
    // Формування та виведення нижнього інформаційного рядка (Метеодані)
    sprintf(lcd_buffer, "%+.1f%cC %u mmHg ", t, 0xdf, (unsigned int)pressure_mmHg);
    LCDGotoXY(0, 1); LCDstring(lcd_buffer);
    // Обробка переривання основного циклу для входу в налаштування
    if (get_key_status(MENU_ENTER_BTN)) {
        if (!get_prev_key_status(MENU_ENTER_BTN)) {
            LCDclr(); main_menu(); LCDclr();
        }
    }
    // Фіксація стану ліній для програмного антидребезгу
    PREV_BTN_PIN = BTN_PIN;
} return 0;}
```