

ДОСЛІДЖЕННЯ ЦИФРОВИХ ПРОТОКОЛІВ ПЕРЕДАВАННЯ ІНФОРМАЦІЇ

Мета: ознайомитись з функціональними можливостями цифрових багатоканальних логічних аналізаторів сигналів та дослідити особливості обміну інформації по шині I²C, 1-Wire, 4-х бітний паралельний інтерфейс LCD індикатора з контролером HD44780.

1 Короткі теоретичні відомості

1.1 Двохнаправлена послідовна шина I²C

У двохнаправленій шині I²C використовується лише 2 лінії: послідовна лінія синхронізації SCL (SCLock) та послідовна лінія даних SDA (SDAta). Сигнал на лінії SCL формується процесором для синхронізації даних периферійного пристрою. Обидва виводи (SDA та SCL) виконуються з відкритим колектором або відкритим стоком. Вони з'єднані з позитивним джерелом живлення через резистор навантаження за схемою «Монтажне І» і, таким чином, декілька пристроїв можуть одночасно використовувати лінії SCL та SDA.

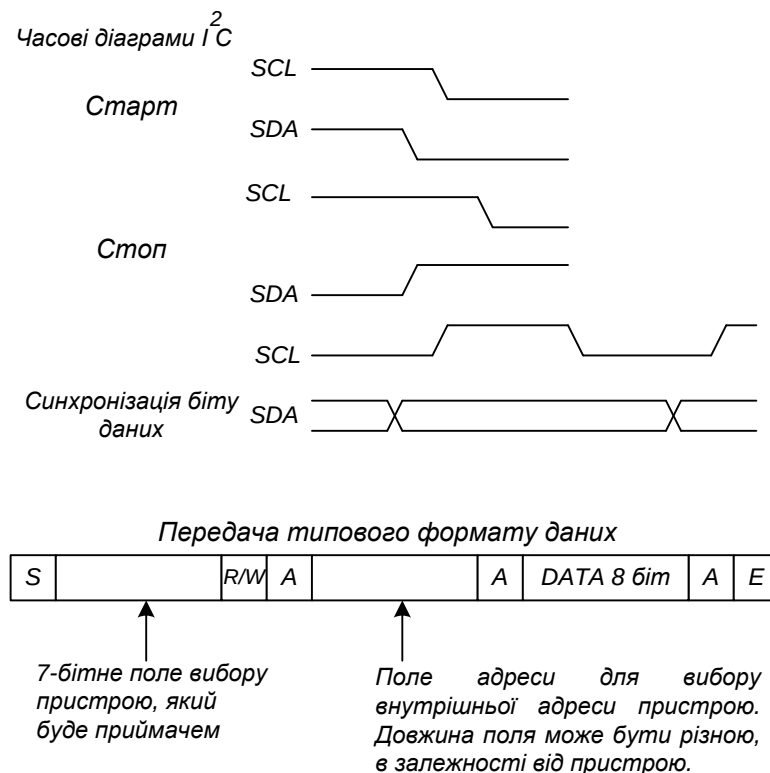


Рисунок 1.1 – Часові діаграми роботи шини I²C

Коли на лінії SCL високий рівень, то зміни рівня сигналу на лінії SDA з одного рівня в інший інтерпретуються як умови «СТАРТ» або «СТОП». Для передачі даних по лінії SDA сигнал на лінії SCL повинний бути встановлений у низький рівень.

Якщо SDA переходить у низький, тоді як на лінії SCL високий рівень, усі периферійні пристрої на шині сприйматимуть цю подію як старт-умову. Якщо SDA переходить у високий, коли на SCL високий рівень, то генерується Стоп-умова. На рисунку 1.1 показаний приклад типової передачі даних шиною I²C. Процесор керування генерує Старт-умову, а потім посилає одночасно всім периферійним пристроям адресу довжиною 7 біт, повідомляючи пристроям, який з них вибраний, та восьмий біт читання/записи (0 – запис, 1 – читання). 7-бітова адреса дозволяє підключати по I²C-шині 128 (2⁷) периферійних пристроїв за однієї умови: ємність шини не повинна перевищувати 400 пФ.

Під час передачі в першому байті восьмого біта читання/запису (R/W) процесор встановлює напрям передачі: програмує R/W на запис (передачу) даних від МП до периферійного пристрою, якщо цей біт дорівнює 0. Інакше (1) МП налаштовується на читання (прийом) даних від периферійного пристрою з адресою, вказаною в 7 бітах. Слід відмітити, що перші 7 біт передаються, починаючи із старшого біта і закінчуючи молодшим бітом, і восьмим бітом передається біт R/W. Після прийому кожного байта, включаючи адресний байт, вибраний периферійний пристрій (приймальна сторона) по лінії SDA посилає сигнал підтвердження на виконання функції прийому переведенням рівня на лінії SDA в низький, щоб показати, що він прийняв адресу та умову читання/запису.

Після прийому біта підтвердження, процесор керування встановлює адресу іншого периферійного пристрою, з яким він хоче встановити зв'язок. Довжина цього поля залежить від периферійного пристрою. Потім приймається біт підтвердження та передаються дані. Деякі периферійні пристрої дозволяють зчитувати або записувати декілька байтів за одну передачу. Процесор повторює послідовність команд «Дані/біт підтвердження» (data/acknowledge) до тих пір, поки всі байти не будуть передані. Периферійний пристрій збільшує свою внутрішню адресу після кожної передачі.

1.2 Шина 1-Wire

Шина 1-Wire є основою мереж MicroLAN і побудована за технологією Master/Slave. На шині повинний бути хоча б один ведучий пристрій (Master). Всі інші пристрої повинні бути веденими (Slave). Ведучий пристрій ініціює всі процеси передачі інформації в межах шини. Master може прочитати дані з будь-якого Slave-пристрою або записати їх туди. Передача інформації від одного Slave-пристрою до іншого безпосередньо неможлива. Для того, щоб Master міг звертатися до кожного з ведених пристроїв по шині, кожний ведений пристрій містить у собі індивідуальний код (ID-код).

Протокол 1-Wire містить у собі спеціальну команду пошуку, за допомогою якої ведучий пристрій (Master) може здійснювати автоматичний пошук ведених пристроїв. У процесі пошуку Master визначає ID-коди для всіх підключених до

мережі мікросхем. Пошук відбувається шляхом поступового відсіювання неіснуючих адрес. Тому для того, щоб знайти всі пристрої, що підключені до шини, потрібен досить значний час. Середня швидкість пошуку елементів у мережі MicroLAN становить близько 75 вузлів за секунду.

Обмін інформацією по шині 1-Wire відбувається за такими правилами.

1. Обмін завжди ведеться з ініціативи одного ведучого пристрою, що у більшості випадків є мікроконтролером.
2. Будь-який обмін інформацією починається з подачі імпульсу скидання ("Reset Pulse" або просто RESET) у лінію 1-Wire ведучим пристроєм.
3. Для інтерфейсу 1-Wire у загальному випадку передбачається "гаряче" підключення й відключення пристроїв.
4. Будь-який пристрій, підключений до 1-Wire, після одержання живлення видає в лінію DQ імпульс присутності, який називається "Presence pulse". Цей же імпульс пристрій завжди видає в лінію, якщо виявить сигнал RESET.
5. Поява в шині 1-Wire імпульсу PRESENCE після видачі RESET однозначно інформує про наявність хоча б одного підключеного пристрою.
6. Обмін інформації ведеться так званими тайм-слотами: один тайм-слот служить для обміну одним бітом інформації.
7. Дані передаються байтами, біт за бітом, починаючи з молодшого біта. Ймовірність переданих/прийнятих даних (перевірка відсутності спотворень) гарантується шляхом підрахунку циклічної контрольної суми.

На рисунку 1.2 показана діаграма сигналів RESET і PRESENCE, з яких завжди починається обмін даними.

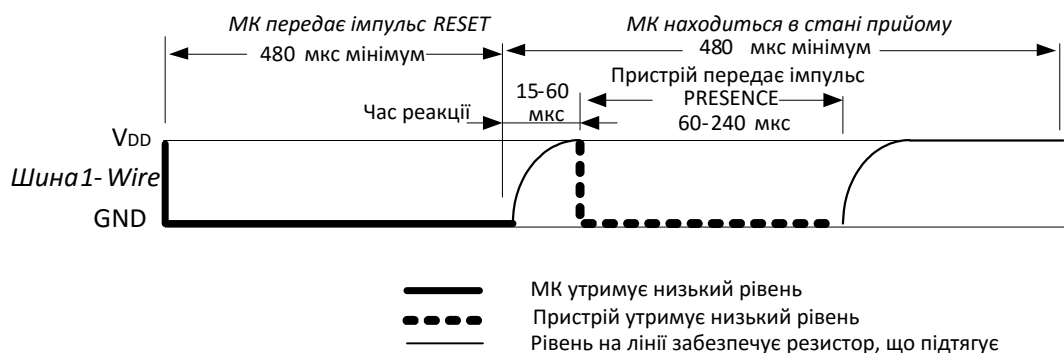


Рисунок 1.2 – Діаграма сигналів RESET і PRESENCE, з яких завжди починається обмін даними

Імпульс RESET формує МК, що переводить в низький логічний рівень шину 1-Wire і втримує її в цьому стані мінімум 480 мікросекунд. Далі МК повинний "відпустити" шину. Через якийсь час, що залежить від ємності лінії й опору резистора, що підтягує, у лінії встановиться високий логічний рівень.

Протокол 1-Wire обмежує цей час «релаксації» діапазоном від 15 до 60 мікросекунд, що і є визначальним для вибору резистора, що підтягує.

Виявивши імпульс RESET, ведений пристрій (Slave) приводить свої внутрішні вузли у вихідний стан і формує відповідний імпульс PRESENCE не пізніше 60 мкс після завершення імпульсу RESET. Для цього пристрій переводить у низький рівень лінію та втримує її в цьому стані від 60 до 240 мікросекунд. Після цього пристрій так само «відпускає» шину.

Але після завершення імпульсу PRESENCE пристрою дається ще якийсь час для завершення внутрішніх процедур ініціалізації, таким чином, МК повинен приступити до будь-якого обміну з пристроєм не раніше, ніж через 480 мкс після завершення імпульсу RESET.

Отже, процедура ініціалізації інтерфейсу, з якої починається будь-який обмін даними між пристроями, триває мінімум 960 мікросекунд, складається з передачі від МК сигналу RESET і прийому від пристрою сигналу PRESENCE. Якщо сигнал PRESENCE не виявлений – значить на шині 1-Wire немає готових до обміну пристроїв.

Обмін бітами інформації здійснюється певними тайм-слотами. Тайм-слот – це певна, досить жорстко лімітована за часом послідовність зміни рівнів сигналу в лінії 1-Wire. Розрізняють 4 типи тайм-слотів: передача «1» від МК, передача «0» від МК, прийом «1» від Slave-пристрою, прийом «0» від Slave-пристрою.

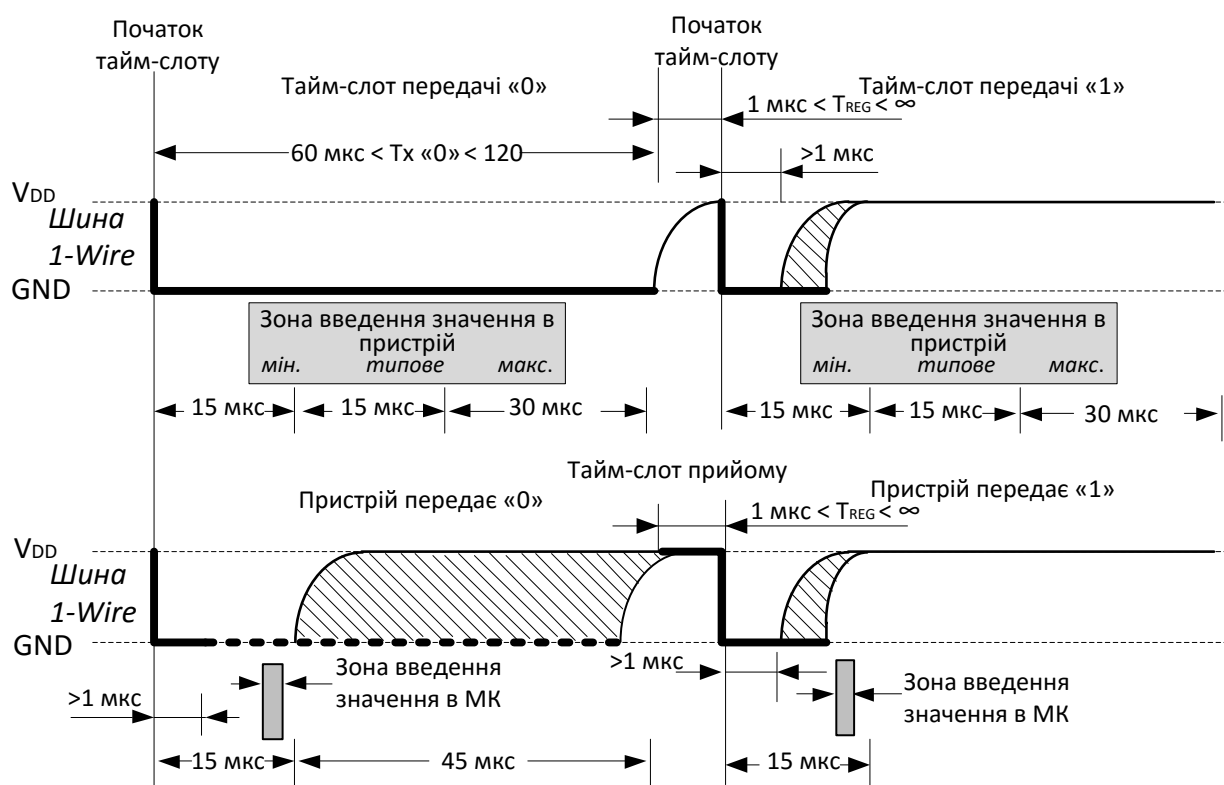


Рисунок 1.3 – Часові діаграми тайм-слотів на лінії 1-Wire

Будь-який тайм-слот завжди починає МК шляхом переведення шини 1-Wire у низький логічний рівень. Тривалість будь-якого тайм-слоту повинна перебувати в межах від 60 до 120 мікросекунд. Між окремими тайм-слотами завжди повинен передбачатися інтервал не менший 1 мікросекунди.

Тайм-слоти передачі відрізняються від тайм-слотів прийому поведінням МК: при передачі він тільки формує сигнали; при прийманні, крім того, ще й опитує рівень сигналу в лінії 1-Wire. Рисунок 1.3 демонструє часові діаграми тайм-слотів всіх 4-х типів: угорі показані тайм-слоти передачі від МК, унизу – прийому від Slave-пристрою.

Тайм-слот передачі «0» полягає просто в утриманні шини 1-Wire у низькому рівні протягом всієї тривалості тайм-слоту. Передача «1» здійснюється шляхом «відпускання» шини 1-Wire з боку МК не раніше 1 мікросекунди після початку тайм-слоту, але не пізніше 15 мкс. Ведений пристрій опитує рівень у шині 1-Wire протягом часового інтервалу, умовно показаного у вигляді сірого прямокутника, тобто починаючи з 15-ї мкс від початку тайм-слоту й закінчуючи 60-ю мкс від початку. Типовий момент уведення рівня в пристрій – біля 30-ї мкс від початку тайм-слоту.

Заштрихована область – це область «наростання» рівня в шині 1-Wire, що залежить від ємності лінії й опору резистора, що підтягує. Тайм-слоти прийому інформації відрізняються тим, що МК формує тільки початок тайм-слоту (абсолютно так само, як при передачі «1»), а потім керування рівнем шини 1-Wire бере на себе пристрій, а МК здійснює введення цього рівня так само в певній зоні часових інтервалів. Заштрихована область – область невизначеності, тому для введення мікроконтролеру залишається навіть не проміжок, а скоріше конкретний момент, коли він повинен увести рівень сигналу з лінії. Цей момент часу – 14-а або 15-а мкс від початку тайм-слоту.

Таким чином МК починає тайм-слот з видачі в шину 1-Wire «0» протягом 1 мкс. Наступний рівень залежить від типу тайм-слоту: для прийому й передачі «1» рівень повинен стати високим, а для передачі «0» – залишатися низьким аж до кінця тайм-слоту, тобто не менш 60 і не більше 120 мкс. Якщо МК приймає дані, то опитування рівня в шині він повинен зробити на проміжку від 13-ї до 15-ї мкс тайм-слоту. МК повинен забезпечити інтервал між окремими тайм-слотами не менше 1 мкс (краще – більше, максимальне значення не обмежене).

Кожний пристрій 1-Wire має унікальний ідентифікаційний 64-бітовий номер, який програмується на етапі виробництва мікросхеми.

Мережний рівень. На цьому рівні відбувається адресація елементів на однопроводовій шині. Перебуваючи на цьому рівні, всі Slave пристрої очікують одну з команд адресації. Така команда повинна вибрати один або кілька пристроїв, з якими Master буде працювати на транспортному рівні. Всі команди мережного рівня наведені в таблиці 1.1

Таблиця 1.1 – Команди мережного рівня

Команда	Значення байта	Опис
SEARCH ROM	0×00	Пошук адреси – використовується при універсальному алгоритмі визначення кількості й адрес підключених пристроїв
READ ROM	0×33	Читання адреси пристрою – використовується для визначення адреси єдиного пристрою на шині
MATCH ROM	0×55	Вибір адреси – використовується для звертання до конкретної адреси пристрою з багатьох підключених
SKI ROM	0×CC	Ігнорувати адресу – використовується для звертання до єдиного пристрою на шині, при цьому адреса пристрою ігнорується (можна звертатися до невідомого пристрою)

Після того, як МК видасть команду READ ROM, від пристрою надійде 8 байтів його власної унікальної адреси – МК повинен їх прийняти. Будь-яка процедура обміну даними з пристроєм повинна бути завершена повністю або перервана посиленням сигналу RESET.

Якщо відправлено команду MATCH ROM, то після неї МК повинен передати так само й 8 байтів конкретної адреси пристрою, з яким буде здійснюватися наступний обмін даними. За цією командою кожний пристрій порівнює передану адресу зі своєю власною. Всі пристрої, адреси яких не збіглися, припиняють аналіз і видачу сигналів у лінії 1-Wire, а пристрій, що пізнав адресу, продовжує роботу. Тепер всі дані, що передані МК, будуть потрапляти лише до цього пристрою. Які саме дані треба послати в пристрій або одержати від нього після його адресації, залежить від конкретного пристрою.

Якщо пристрій один на шині, то можна прискорити процес взаємодії з ним за допомогою команди SKI ROM. Отримавши цю команду, пристрій відразу вважає, що адреса збіглася, хоча ніякої адреси для цієї команди не треба. Деякі процедури не вимагають прийому від пристрою ніяких даних, у цьому випадку команду SKI ROM можна використовувати для передачі деякої інформації відразу всім пристроям.

Транспортний рівень. Всі мікросхеми мережі MicroLAN переходять на цей рівень після будь-якої команди мережного рівня. Набір команд транспортного рівня для різних мікросхем дещо відрізняється. Приклади команд транспортного рівня для різних типів мікросхем наведені в таблиці 1.2. Основні команди транспортного рівня – це команди «Запис пам'яті» та «Читання

пам'яті». Як приклад команди транспортного рівня розглянемо команду «Читання пам'яті».

Таблиця 1.2 – Приклади команд транспортного рівня

Мікросхема	Код	Команда
DS1994 (4096-бітова перезаписовна енергонезалежна пам'ять)	0FH	Запис в блокнотну пам'ять
	CAAH	Читання блокнотної пам'яті
	55H	Копіювання блокнотної пам'яті
	0F0H	Читання основної пам'яті
DS2417 Годинник реального часу	66H	Читання часу
	99H	Запис часу
DS2450 4-канальний АЦП	0AAH	Читання з пам'яті результатів перетворення напруги
	55H	Запис до пам'яті команд керування
	3CH	Запуск процесу перетворення напруги в код
DS18S20/DS18B20/DS1822 Електронні датчики температури	44H	Запуск процесу конвертування температури в код
	4EH	Запис у блокнотну пам'ять (меж регулювання температури)
	0BEH	Читання блокнотної пам'яті
	48H	Копіювання блокнотної пам'яті в спеціальні осередки внутрішньої EEPROM
	0B8H	Читання інформації з EEPROM у блокнотну пам'ять
	0B4H	Читання інформації про режим живлення

На рисунку 1.4 наведена часова діаграма виконання команди «Читання пам'яті». Для адресації мікросхеми в наведеному прикладі використана команда мережного рівня – «Збіг ПЗП». Як видно з рисунка, виконання команди починається з імпульсу скидання. Далі всі ведені пристрої виробляють сигнал присутності на лінії. Витримавши паузу Master-пристрій видає на лінію команду «Збіг ПЗП». Команда складається з восьми бітів коду операції й 64-бітового ID-коду. Одержавши цю команду, всі ведені мікросхеми переходять на транспортний рівень. Але тільки одна з них залишиться активною. Інші перейдуть у пасивний режим очікування.

Далі Master-пристрій передає на шину команду «Читання пам'яті» (8 бітів). Одержавши цю команду, обрана мікросхема переходить у режим видачі даних.

Master-пристрій читає ці дані з веденої мікросхеми. Обсяг даних, що зчитуються, залежить від типу мікросхеми. Найчастіше він дорівнює 8 байтам (64 біти) і обов'язково містить контрольну суму в останніх восьми бітах.

Після того, як всі дані прийняті, виконання команди можна вважати закінченим. Для того, щоб виконати наступну команду, потрібно все починати з початку. Тобто, з імпульсу початкового скидання.



Рисунок 1.4 – Часова діаграма виконання команди «Читання пам'яті»

Структура індивідуального коду, записаного в спеціальний ПЗП будь-якої мікросхеми, що підтримує інтерфейс 1-Wire, умовно зображена на рисунку 1.5. Код складається з трьох основних елементів:

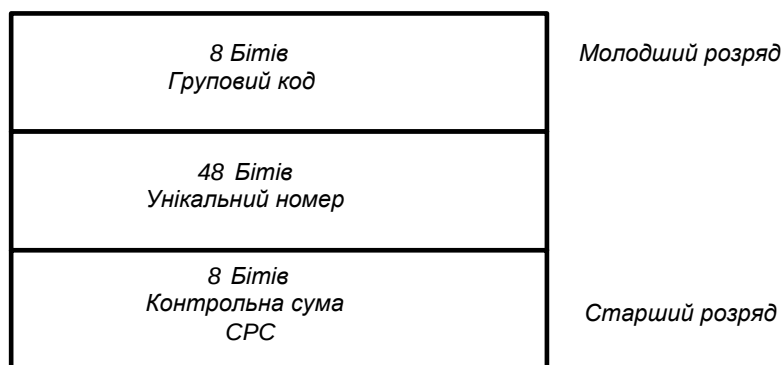


Рисунок 1.5 – Структура індивідуального коду мікросхеми 1 -Wire

Перші 8 бітів, починаючи з молодшого розряду, – це груповий код мікросхеми. Груповий код однозначно визначає її вид. Приклади групового коду для різних видів мікросхем наведені в таблиці 1.3. Наступні 48 бітів – це унікальний серійний номер мікросхеми. Останні 8 бітів – це контрольна сума (CRC). Контрольна сума або CRC – це байт, значення якого передається найостаннішим і обчислюється за спеціальним алгоритмом на основі значення усіх семи попередніх байтів. Алгоритм підрахунку такий, що якщо усі байти, що передаються-приймаються, без спотворень (а спотворення цілком можливі, якщо згадати характер апаратної реалізації інтерфейсу), то прийнятий байт контрольної суми обов'язково збіжиться з розрахованим в МК (чи пристрої) значенням. Тобто при реалізації програмного алгоритму обміну інформацією необхідно при передачі і прийомі байтів підраховувати їх контрольну суму за певним алгоритмом, а потім або передати отримане значення, або порівняти

розрахункове значення з отриманим значенням CRC. Тільки при збігу обох CRC МК або пристрій вважають прийняті дані достовірними. Інакше продовження обміну неможливе.

Очевидно, що алгоритм підрахунку CRC має бути однаковим як для МК, так і для будь-якого пристрою. Він «стандартизований» та описаний в документації.

Таблиця 1.3 – Групові коди деяких видів мікросхем

Вид мікросхеми	Опис	Груповий код
DS18S20	Електронний термометр	10H
DS18B20	Електронний термометр	28H
DS18B22	Електронний термометр	22H
DS1992	Незалежна пам'ять	08H
DS1993	Незалежна пам'ять	0EH
DS1994	Незалежна пам'ять	04H
DS2401	Кремнієвий серійний номер	01H
DS2417	Таймер реального часу	27H
DS2450	4-канальний АЦП	20H
DS2890	Електронний змінний резистор	2CH

1.3 LCD індикатор з контролером типу HD44780

LCD індикатор з контролером типу HD44780 фірми Hitachi є промисловим стандартом де-факто на ринку цифро-буквених індикаторів.

Є два способи підключення LCD індикатора: 8 бітний та 4 бітний. У восьми бітному режимі (рис. 1.6) простіше передавати байти – не потрібно зсувати байт, але в чотирьох бітному режимі використовується на чотири ніжки контролера менше. Підключення індикатора до МК ATtiny2313 у чотирьох бітному режимі наведено на рисунку 1.7.

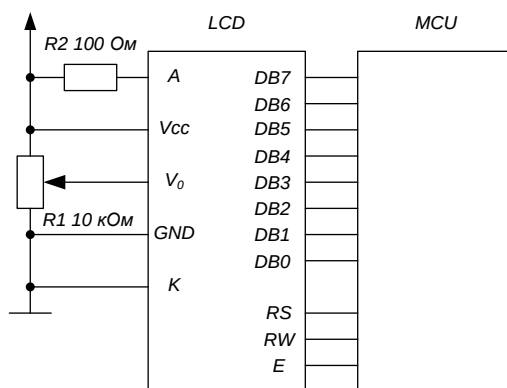


Рисунок 1.6 – Підключення LCD індикатора на базі HD44780 до мікроконтролера у восьми бітному режимі

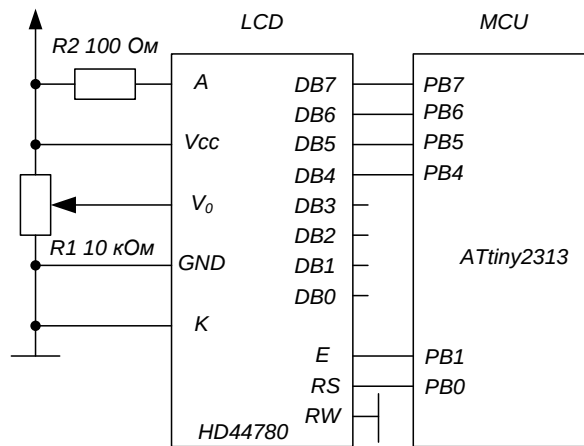


Рисунок 1.7 – Підключення LCD індикатора на базі HD44780 мікроконтролера у чотирьох бітному режимі

Виводи DB7...DB0 це шина даних/адреси.

E – стробувальний вхід. Імпульс напруги на цій лінії повідомляє дисплею, що потрібно забирати/віддавати дані з/на шину даних.

RW – визначає в якому напрямку рухаються дані. Якщо 1, то читання з дисплея, якщо 0 то запис у дисплей.

RS – визначає, що передається команда (RS=0) або дані (RS=1). Дані будуть записані на згадку по поточній адресі, а команда виконана контролером.

GND – мінус, він же загальний.

Vcc – плюс живлення, звичайно 5В.

V0 – вхід контрастності.

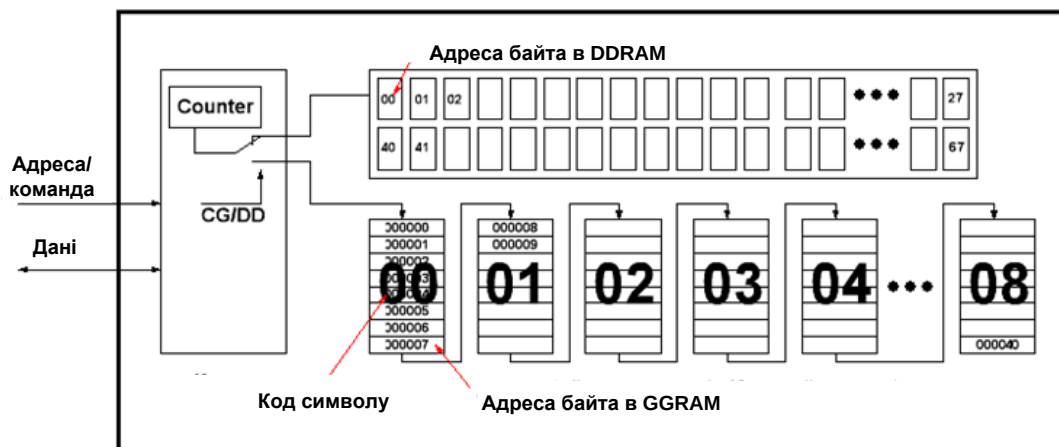


Рисунок 1.8 – Структура адресації контролера HD44780

Контролер HD44780 має свій блок керування, що обробляє команди й пам'ять. Вона ділиться на три види: DDRAM, CGROM, CGRAM (рис. 1.8).

DDRAM – пам'ять дисплея. Все що запишеться в DDRAM буде виведено на екран. Тобто, наприклад, записали код 0x31 – на екрані відобразиться символ «1», так як 0x31 це ASCII код цифри 1. Але є одна особливість – DDRAM пам'ять набагато більше ніж видима область екрана. Як правило,

DDRAM містить 80 осередків – 40 у першому рядку й 40 у другому, а на дисплеї може рухатися по цій лінійці (рис. 1.9). Наприклад, можна записати в DDRAM відразу п'ять пунктів меню, а потім просто показувати по одному пункті. Для переміщення дисплея є спец команда. Також є поняття курсору – це місце в яке буде записаний наступний символ, тобто поточне значення лічильника адреси. Курсор не обов'язково може бути на екрані, він може розташовуватися й за екраном або бути відключений зовсім

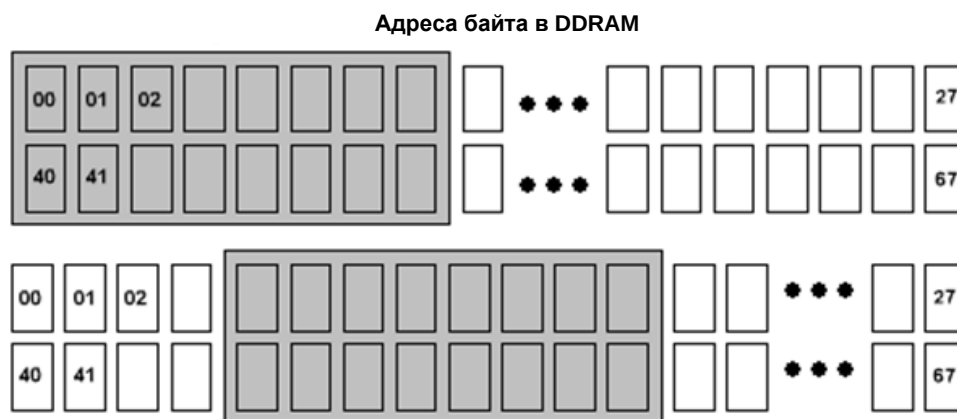


Рисунок 1.9 – Видима й прихована область екранної пам'яті

CGROM – таблиця символів. Коли записуємо в осередок DDRAM байт, то з таблиці береться символ і рисується на екрані. CGROM не можна змінити, тому важливо, щоб вона мала англійські або російські букви.

CGRAM – теж таблиця символів, але її можна міняти, створюючи свої символи. Адресується вона лінійно, тобто спочатку йде 8 байт одного символу, як наведено на рисунку 1.10.

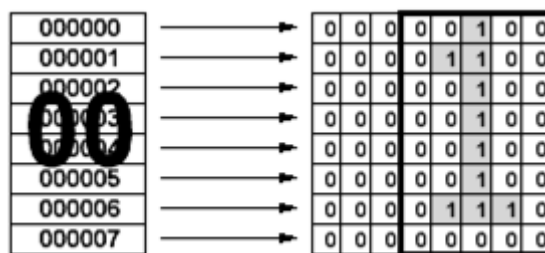


Рисунок 1.10 – Формування символу в комірці CGRAM

Усього в CGRAM може бути 8 символів, відповідно CGRAM має 64 байта пам'яті. Ці програмувальні символи мають коди від 0x00 до 0x07. Так що, записавши, наприклад, у перші 8 байт CGRAM (перший символ з кодом 00) будь який символ, і записавши в DDRAM нуль (код першого символу в CGRAM) побачимо на екрані цей символ.

Командою вибирається в яку саме пам'ять і починаючи з якої адреси будуть записуватись дані, а потім пересилаємо байти. Система команд контролера індикатора дуже проста. Про те, що передається команда контролеру дисплея повідомить RS=0. Сама команда складається зі старшого

біта, що визначає за що відповідає дана команда й біт параметрів, що вказують контролеру HD44780 як далі працювати (табл. 1.4).

Таблиця 1.4 – Таблиця команд HD44780

DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	Значення
0	0	0	0	0	0	0	1	Очищення екрана. Лічильник адреси на 0 позицію DDRAM
0	0	0	0	0	0	1	-	Адресація на DDRAM скидання зсуву. Лічильник адреси на 0
0	0	0	0	0	1	I/D	S	Настроювання зсуву екрана й курсору
0	0	0	0	1	D	C	B	Настроювання режиму відображення
0	0	0	1	S/C	R/L	-	-	Зсув курсору або екрана, залежно від біт
0	0	1	DL	N	F	-	-	Вибір числа ліній, ширини шини й розміру символу
0	1	AG	AG	AG	AG	AG	AG	Перемкнути адресацію на SGRAM і задати адресу BSGRAM
1	AD	AD	AD	AD	AD	AD	AD	Перемкнути адресацію на DDRAM і задати адресу в DDRAM

I/D – інкремент або декремент лічильника адреси. По замовчуванні встановлений 0 (декремент).

S – зсув екрана, якщо поставити 1, то з кожним новим символом буде пересуватись вікно екрана, поки не досягне кінця DDRAM.

D – включити дисплей. Якщо поставити 0 то зображення зникне, а в цей час можемо у відеопам'яті записати нову інформацію. Для того, щоб вона з'явилася в цю позицію треба записати 1.

C – включити курсор у вигляді прочерку.

B – зробити курсор у вигляді чорного квадрата.

S/C – зсув курсору або екрана. Якщо встановлено 0, то зсувається курсор. Якщо 1, то екран. По одному разу за команду.

R/L – визначає напрямок зсуву курсору й екрану (0 – ліворуч, 1 – праворуч).

D/L – біт, що визначає ширину шини даних (1 – 8 біт, 0 – 4 біти).

N – число рядків (0 – один рядок, 1 – два рядки).

F – розмір символу (0 – 5x8, 1 – 5x10).

AG – адреса в пам'яті CGRAM. AD – адреса в пам'яті DDRAM.

1.4 Спеціалізовані вимірювальні пристрої

Логічний пробник це прилад, що підключається до того ж джерела живлення, що й досліджуємо схема, має загострений щуп, що використовується для тестування різних точок схеми. Покази індикаторів логічного пробника зображені на рис 1.11. Логічний імпульсний генератор призначений для подачі імпульсу на вхід схеми, у той час як логічний пробник відслідковує логічний стан виходу. Звичайно логічний імпульсний генератор використовується разом з логічним пробником і живиться від схеми тестування.

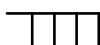
Стан логічного пробника	1	0	Імпульс
Обрив	○	○	○
Логічний 0	○	●	○
Логічна 1	●	○	○
Прямокутні імпульси >100кГц	●	●	◐
Прямокутні імпульси <100кГц	○	○	◐
	○	●	◐
	●	○	◐

Рисунок 1.11 – Покази індикаторів логічного пробника

У складних цифрових схемах, особливо в мікропроцесорних системах, необхідно знати зв'язок між багатьма різними сигналами. У цій ситуації форми напруги не так важливі, як логічні стани, що виникають у певні моменти часу. Логічний аналізатор - це прилад, що дозволяє одночасно збирати й зберігати інформацію про логічний стан по багатьом каналам (48 і більше). Ці виміри виконуються через певні інтервали часу, які задаються внутрішніми джерелами тактових імпульсів або зовнішнім джерелом, що перебуває в схемі дослідження. Логічний аналізатор може бути з'єднаний з персональним комп'ютером. Він може представляти інформацію у вигляді часових діаграм або таблиці станів.

2 Прилади та обладнання

2.1 Логічний аналізатор цифрових сигналів

Логічний аналізатор цифрових сигналів (Logic Analyzer) призначений для захоплення та передавання до ПК даних про логічні стани цифрових сигналів на кожному каналі приладу для їх подальшого представлення у вигляді часових діаграм з можливістю виміру основних параметрів та аналізу протоколів за допомогою програмного забезпечення «Saleae logic».

Робота з логічним аналізатором (рис. 2.1) можлива тільки після установки програмного забезпечення. Після підключення до вільного USB порта ПК загорається світлодіод PWR зеленим кольором. Logic Analyzer забезпечує частоту опитування 24 MHz на кожний канал, якщо на шині USB відсутні інші пристрої та ПК має достатню продуктивність. Якщо USB-контролер обслуговує інші пристрої, то частота опитування знижується до 16MHz або 12MHz.



Рисунок 2.1- Зовнішній вигляд 8 каналного USB логічного аналізатора

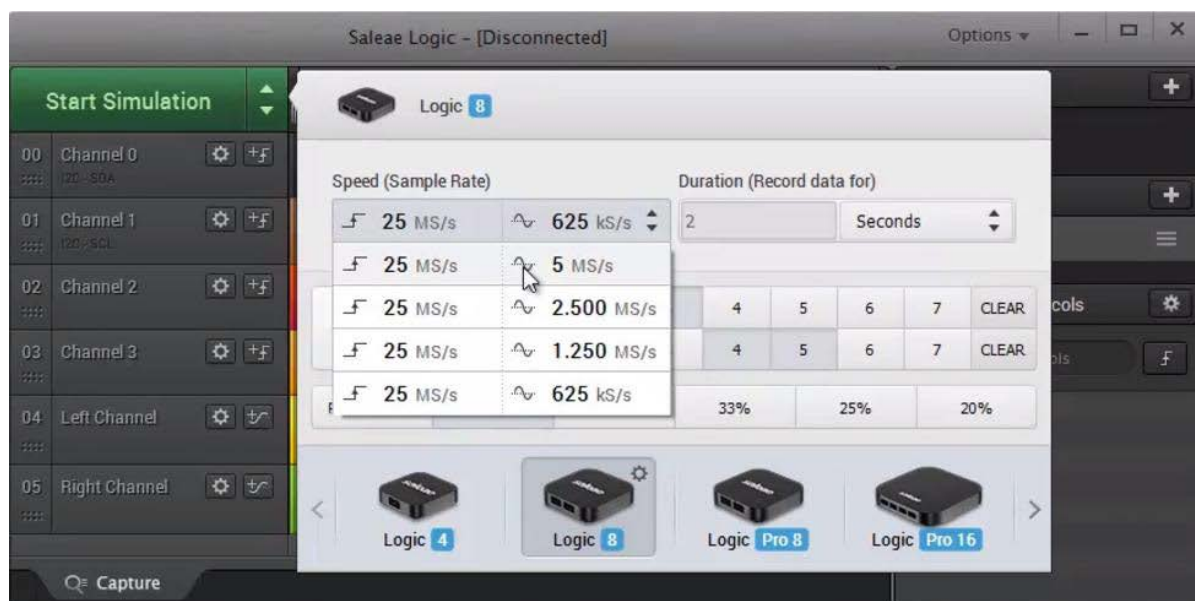


Рисунок 2.2 – Saleae Logic. Режим налаштувань

Після запуску програмного забезпечення «Saleae Logic» на екран виводиться основне вікно аналізатора, на якому присутнє поле діаграми з вісями даних для 8 каналів та мітками часу, кнопки вибору бажаного об'єму вибірки та частоти опитування каналів. Заголовок програми Disconnected / Connected змінюється коли логічний аналізатор підключений до ПК. На рисунку 2.2 показано режим налаштування програмного забезпечення, який викликається натисненням на правий кут кнопки «Start Simulation». У даному режимі необхідно вибрати «Logic8»; встановити частоту опитування (Speed) – 25MS/s для цифрових сигналів, 5MS/s для аналогових сигналів; час аналізу сигналів (Duration) – 2 с; вибрати канали для захоплення та режим захоплення за натисненням кнопки або за подію по тригеру.

Програмне забезпечення дозволяє у зручному виді переглядати записані дані, об'єм та частоту зняття яких задається програмно. Передбачений режим курсорних вимірювань, вимірювань частоти та тривалості сигналів. Є можливість експорту даних в зручних форматах для документування.

Для вимірювання часових параметрів передбачені такі режими:

- тривалість (Width): час між двома виділеними фронтами сигналу;
- період (Period): час, між трьома виділеними фронтами сигналу;
- частота (Frequency): величина, що обернена до періоду;
- шпаруватість (Duty Cycle): % часу, при якому сигнал був у стані «1» за певний період;
- байт (Byte): цифрове представлення усіх 8-ми каналів у заданий момент часу;
- T1: абсолютне значення часу, що задане маркером T1;
- T2: абсолютне значення часу, що задане маркером T2.
- $|T1 - T2|$: час, що пройшов між маркерами.

Приклад аналізу та вимірювання параметрів цифрового сигналу за допомогою логічного аналізатора наведений на рисунку 2.3.

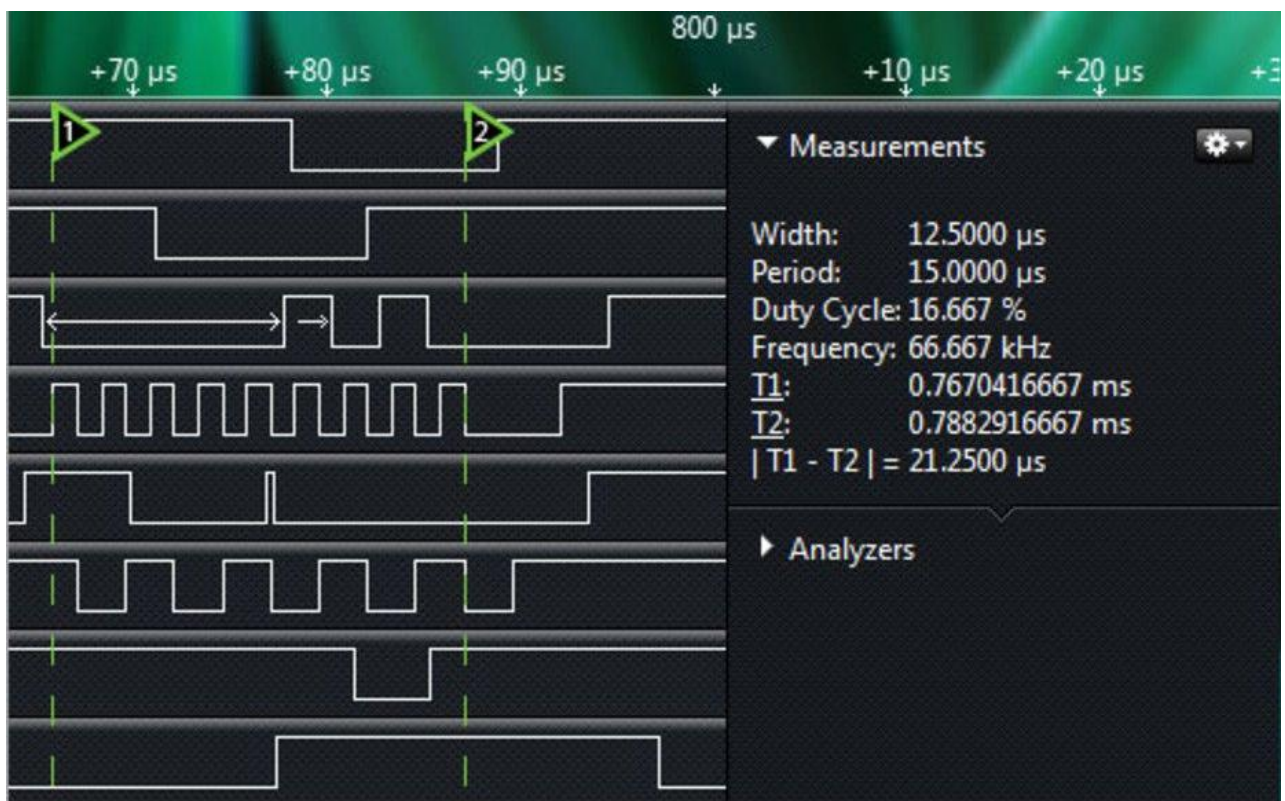


Рисунок 2.3 - Salae Logic. Вимірювання параметрів цифрових сигналів

Logic Analyzer супроводжується потужним програмним забезпеченням з можливістю аналізу багатьох популярних інтерфейсів обміну даними (UART, I2C, SPI, 1Wire, CAN та ін.). Для додавання аналізатора протоколу, необхідно вибрати його з меню на панелі аналізаторів (рис. 2.4) та вказати який канал за який сигнал відповідає (рис. 2.5). Після додавання/редагування аналізатора, він автоматично обробляє усі накопичені в каналі дані (рис. 2.6).

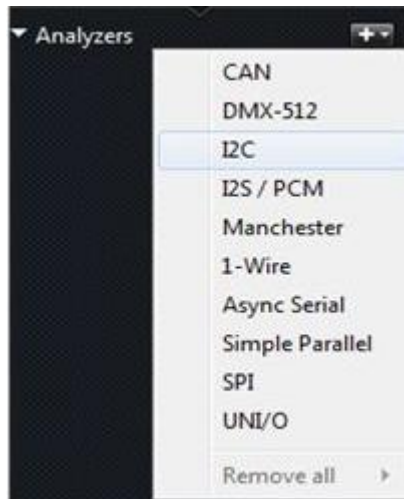


Рисунок 2.4 – Вибір аналізатора протоколів

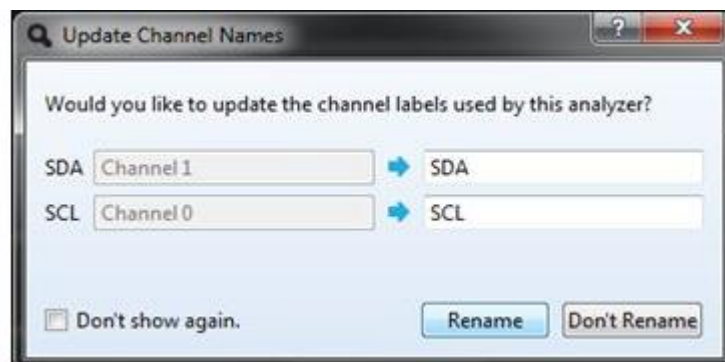


Рисунок 2.5 – Налаштування аналізатора протоколів I2C

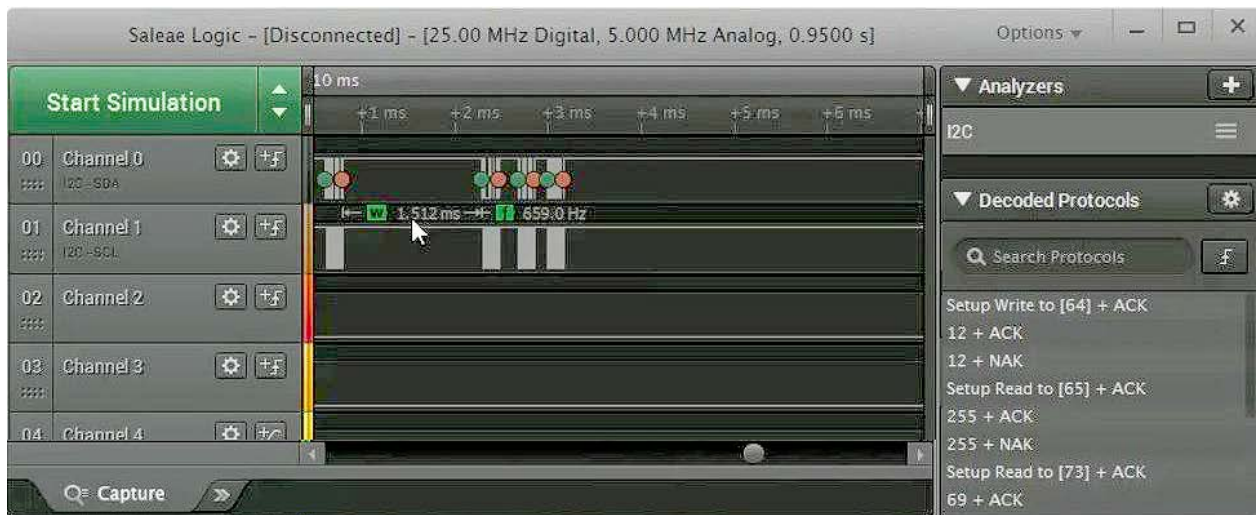


Рисунок 2.6 – Аналіз та декодування інформації шини I2C

2.2 Лабораторний макет

Лабораторний макет представляє собою годинник-календар з здавачем температури. Схема лабораторного пристрою наведена на рисунку 2.7. Схема складається з контролера керування ATtiny2313 (U1), LCD індикатора 16x2 на HD44780, годинника реального часу DS1307 (U3) з інтерфейсом I2C, давача температури DS18B20 з інтерфейсом 1-Wire. Для встановлення годин та

календаря необхідно натиснути кнопку «*». Натиснув «*» один раз, попадаємо в режим установки хвилин. Кнопками «+» та «-» установлюється значення хвилин. Натиснення «*» другий раз переводить до режиму установки годин; третій раз - установки дня тижня (не відображається), далі дата, місяць, рік. Наступне натиснення – вихід з режиму установки. В основному циклі програми відбувається опитування натиснення кнопок. За перериванням компаратора, орієнтовно, раз за секунду, мікроконтролер отримує значення температури, дані від годинника реального часу, обробляє ці дані та виводить їх на LCD індикатор.

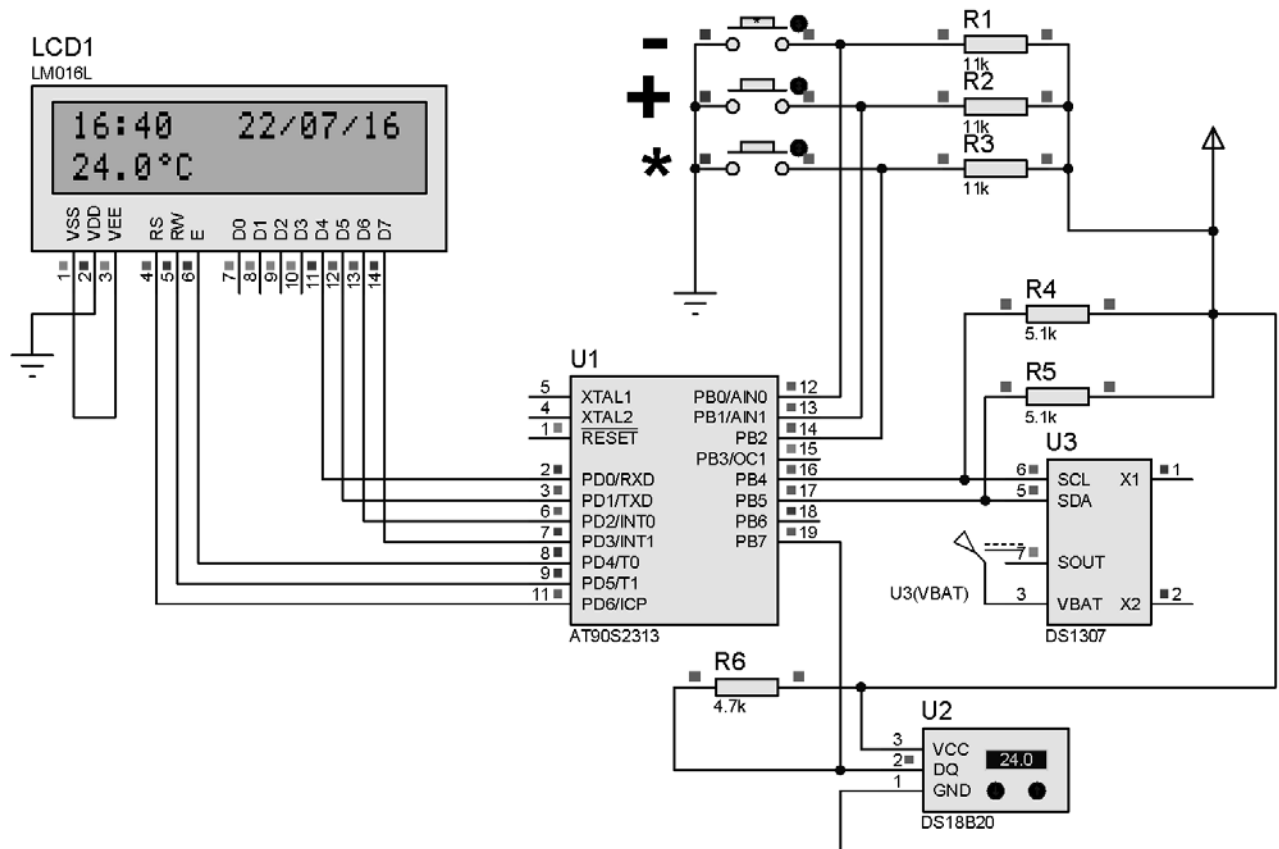


Рисунок 2.7 – Схема лабораторного макета

3 Хід роботи

1. Ознайомитись з конструкцією лабораторного макету (рис. 2.8) та логічного аналізатора (рис. 2.1).
2. Підключити логічний аналізатор до персонального комп'ютера. Підключити лабораторний макет до персонального комп'ютера. Встановити поточну дату та час на LCD індикаторі. З'єднати контакти SCL, SDA лабораторного макету з Logic Analyzer. Запустити програмне забезпечення «Saleae logic» та налаштувати його для роботи (рис. 2.2).
3. Підключити аналізатор протоколів I²C (рис. 2.3, рис. 2.4) та виконати захоплення сигналів логічним аналізатором. Провести аналіз даних на лінії SDA. Навести ретельний аналіз даних з поясненням особливостей обміну інформації між МК (U1) та DS1307 (U3) (рис. 2.7). Визначити

фізичну адресу DS1307 (U3), режим роботи (Read/Write) та дані, що передаються до МК. Виміряти параметри імпульсного сигналу на лінії SCL та задокументувати отриману інформацію.

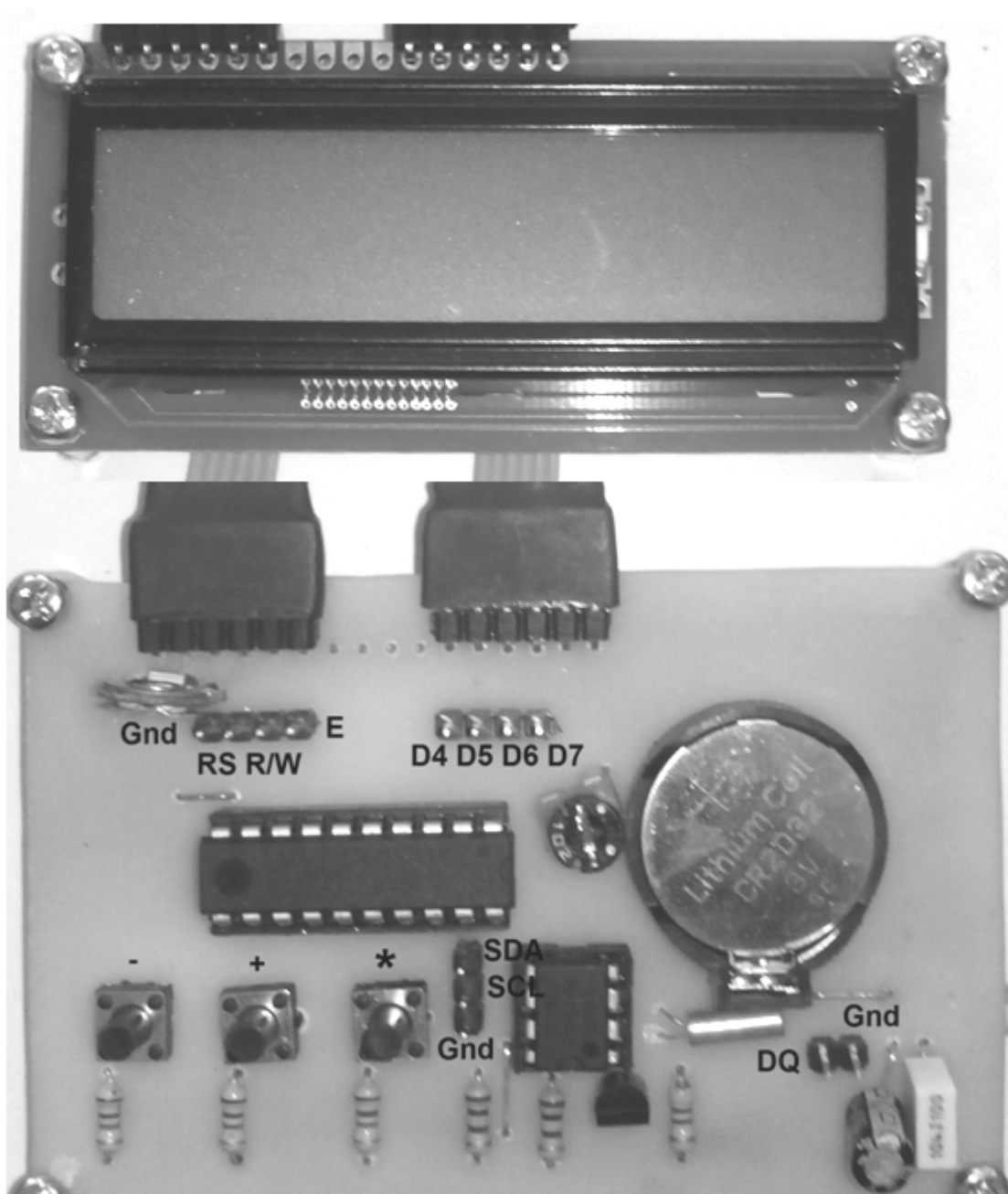


Рисунок 2.8 – Зовнішній вигляд лабораторного макета

4. З'єднати контакт DQ лабораторного макету з Logic Analyzer. Підключити аналізатор протоколів 1-Wire, налаштувати його та виконати захоплення сигналів логічним аналізатором. Провести аналіз обміну інформацією між МК (U1) та датчиком температури DS18B20 (U2) по шині 1-Wire. Визначити команди мережного та транспортного рівня, що передає МК (рис. 3.1). Визначити коди даних та CRC код. Задокументувати отриману інформацію. Показати на часовій діаграмі такі елементи: 1-Wire Reset, Presence Pulse, Overdrive commands and data, All Standard ROM commands, Data Bytes. Виміряти тривалість імпульсу 1-Wire Reset, паузи та імпульсу

присутності 1-Wire пристрою Presence Pulse.

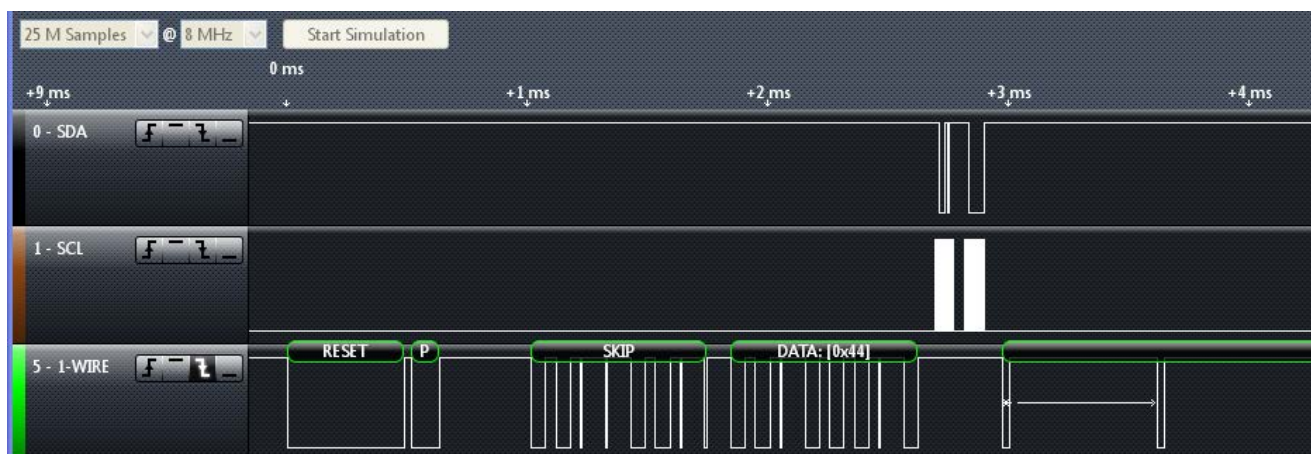


Рисунок 3.1 – Аналіз обміну інформації по шині 1-Wire

E Signal	4 - 'Channel 4'
RS Signal	6 - 'Channel 6'
RW Signal	5 - 'Channel 5'
DB0 Signal	None
DB1 Signal	None
DB2 Signal	None
DB3 Signal	None
DB4 Signal	3 - 'Channel 3'
DB5 Signal	2 - 'Channel 2'
DB6 Signal	1 - 'Channel 1'
DB7 Signal	0 - 'Channel 0'
Mark Timing Errors	☒
Enable Cycle Min (ns)	500
Enable Pulse Width Min (ns)	230
Address Setup Min (ns)	40
Address Hold Min (ns)	10
Data Write Setup Min (ns)	80
Data Write Hold Min (ns)	10
Data Read Delay Max (ns)	160
Data Read Hold Min (ns)	5
Ignore E Pulses When LCD Controller is Busy	☒
Busy Time for Clear/Home (us)	1520
Busy Time for Command/Data (us)	37
Start In 4-bit Mode (Init Missing)	☒
Do Not Generate Frames for Busy Checks	☐

Рисунок 3.2 – Налаштування аналізатора протоколів інтерфейсу HD44780

5 Підключити шупи логічного аналізатора до контактів LCD індикатора: DB7-CH0, DB6-CH1, DB5-CH2, DB4-CH3, E-CH4, R/W-CH5, RS-CH6. Підключити аналізатор протоколів HD44780 (рис. 3.2), налаштувати його та виконати захоплення сигналів логічним аналізатором. Провести аналіз обміну інформацією між МК (U1) та LCD індикатором (рис. 3.3). Навести коди команд та даних, що передаються мікроконтролером. Розшифрувати їх користуючись даними таблиці 1.4. Порівняти їх з інформацією, що відображається на індикаторі.

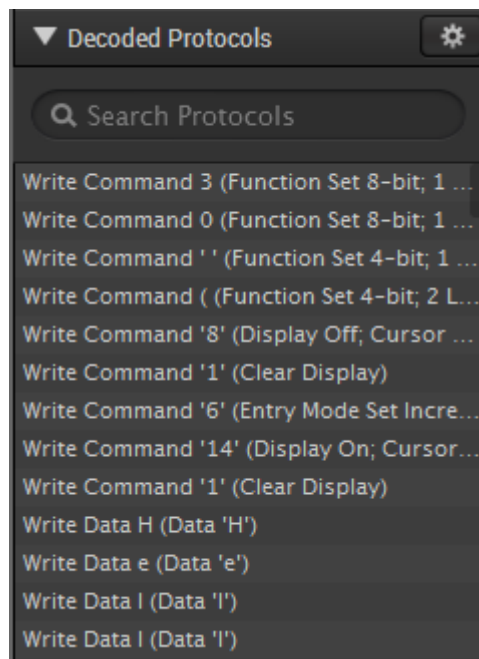


Рисунок 3.3 – Приклад декодування протоколу паралельного інтерфейсу індикатора з контролером HD44780

4 Контрольні питання

1. Особливості обміну інформацією по шині I²C.
2. Навести часові діаграми роботи шини I²C для передавання інформації від МК до EEPROM (adr – 01H, data – 09A5H).
3. Навести часові діаграми роботи шини I²C для передавання інформації від EEPROM до МК (adr – 01H, data – 1357H).
4. Особливості обміну інформацією по шині 1-Wire.
5. Як ведучий пристрій (Master) здійснює автоматичний пошук ведених пристроїв?
6. Пояснити призначення сигналів RESET і PRESENCE. Які особливості їх передавання по шині 1-Wire?
7. Наведіть тайм-слоти передавання інформації від МК до датчика температури DS18B20.
8. Наведіть тайм-слоти приймання інформації МК від датчика температури DS18B20.

9. Поясніть особливості застосування команд 33H, 55H, CCH. Наведіть практичні приклади застосування кожної команди.
10. Які команди транспортного рівня необхідно подати МК для зчитування інформації про значення температури в лабораторному макеті?
11. Наведіть алгоритм обміну інформацією по шині 1-Wire між МК (U1) та здавачем температури (U2).
12. Наведіть послідовність команд та даних для LCD індикатора з контролером HD44780 для виведення на нього значення температури (рис. 2.7)
13. Наведіть послідовність команд та даних для LCD індикатора з контролером HD44780 для виведення на нього значення поточного часу (рис. 2.7)
14. Наведіть послідовність команд та даних для LCD індикатора з контролером HD44780 для виведення на нього значення поточної дати (рис. 2.7).

Література

1. Цирульник С. М. Проектування мікропроцесорних систем/ С.М. Цирульник, Г.Л. Лисенко. – Вінниця: ВНТУ, 2012. – 191с. – ISBN 978-966-641-463-5.
2. Рюмик С. М. 1000 и одна микронтроллерная схема. Вып. I/ С. М. Рюмик. – М. : Додэка-XXI, 2010. –356 с.
3. Рюмик С. М. 1000 и одна микронтроллерная схема. Вып. II / Рюмик С. М. – М. : «Додека-XXI», 2011. – 400 с.
4. Радиокот. [Електроний ресурс]/ Часы, календарь, термометр. – Режим доступу: <http://radiokot.ru/circuit/digital/home/54/>, вільний. – Загл. з екрана. – Мова російська.
5. Офіційна сторінка компанії Saleae. [Електроний ресурс]. – Режим доступу: <http://www.saleae.com/logic/>, вільний. – Загл. з екрана. – Мова англійська.