МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

ДОНЕЦЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ

# **МЕТОДИЧНІ ВКАЗІВКИ**

ДО ВИКОНАННЯ ЛАБОРАТОРНИХ РОБІТ З КУРСУ

“ПРОЕКТУВАННЯ ЗАСОБІВ ТА СИСТЕМ УПРАВЛІННЯ”

ЧАСТИНА 1. МЕРЕЖІ ДЛЯ САУ

Донецьк 2005

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

ДОНЕЦЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ

# МЕТОДИЧНІ ВКАЗІВКИ

ДО ВИКОНАННЯ ЛАБОРАТОРНИХ РОБІТ З КУРСУ

“ПРОЕКТУВАННЯ ЗАСОБІВ ТА СИСТЕМ УПРАВЛІННЯ”

ЧАСТИНА 1. МЕРЕЖІ ДЛЯ САУ

для студентів спеціальності

“Комп’ютеризовані системи управління і автоматика”

##### Затверджено

на засіданні кафедри

автоматики і телекомунікацій

##### Протокол №\_\_\_від\_\_\_\_\_\_\_\_

Донецьк 2005

УДК 681.5.017(076.5)

Методичні вказівки до виконання лабораторних робіт з курсу “Проектування засобів та систем управління” (Частина 1. Мережі для САУ) (для студентів спеціальності 7.091401).

Укл.: С. Ф. Суков, Є. О. Вінник. – Донецьк: ДонНТУ, 2005. – 36 с.

### Укладачі: С. Ф. Суков, доц.

Є. О. Вінник, ст. гр. АТ-00

Відп. за випуск: В. І. Бессараб, доц.

###### **Лабораторна робота №1**

# **Вивчення мережі MicroLAN**

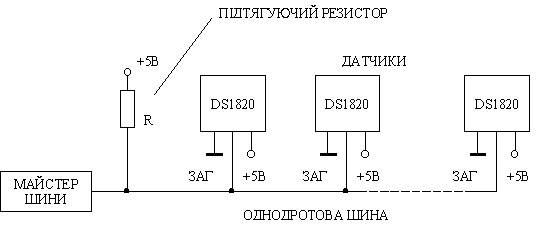
**Ціль роботи:** вивчити протокол передачі даних MicroLAN, ознайомитися з обладнанням для сполучення мережі MicroLAN з комп'ютером, навчитися працювати з окремим датчиком.

# **Теоретичні відомості**

**1. Загальні відомості про мережу MicroLAN**

Мікролокальна мережа (Miniature Local Area Network – MicroLAN) – це мережа, що використовує для цифрового обміну однодротову лінію зв'язку. Ця мережа була створена фірмою Dallas Semiconductor на основі однодротового інтерфейсу. Для роботи в цій мережі виготовляється цілий ряд пристроїв, таких як адресовані ключі, АЦП, термометри, годинники реального часу, цифрові потенціометри.

В основу роботи MicroLAN покладена організація зв'язку процесора з периферійними пристроями через однодротову шину. Однодротова шина являє собою систему, що складається з одного майстра шини (пристрій, керуючий роботою шини, яким може бути мікропроцесор або комп'ютер) і одного чи декількох помічників (пристрій, підключений до однодротової шини, яким може бути датчик, адресований ключ, модуль пам'яті і т.п.) (мал. 1).

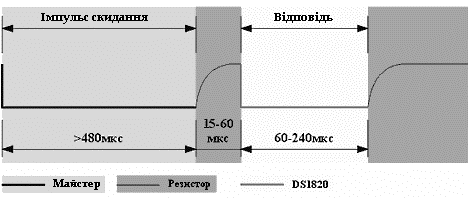


###### Малюнок 1 – Однодротова шина

У нашому випадку помічниками будуть виступати термодатчики. Інтелектуальні температурні датчики являють собою спеціалізовані мікропроцесори, що вимірюють температуру навколишнього середовища і перетворюють її значення в послідовний двійковий код. Кожен датчик індивідуально маркірований, тобто містить усередині себе ідентифікаційний номер, за яким керуючий комп'ютер може розпізнати кожен конкретний датчик у мережі і звернутися до нього. Двох датчиків з однаковими ідентифікаційними номерами не існує.

Однодротова шина оперує з ТТЛ-рівнями, тобто логічна одиниця представлена рівнем напруги близько 5 В, а логічний нуль – напругою поблизу 0 В. Основний стан шини – високий. Він забезпечується підтягуючим резистором R, що підключає шину до джерела живлення. Будь-яка передача інформації здійснюється шляхом закорочування шини майстром чи помічниками, тобто переведенням її в низький стан. Як тільки і майстер, і помічники відпускають шину, на ній установлюється вихідний високорівневий стан.

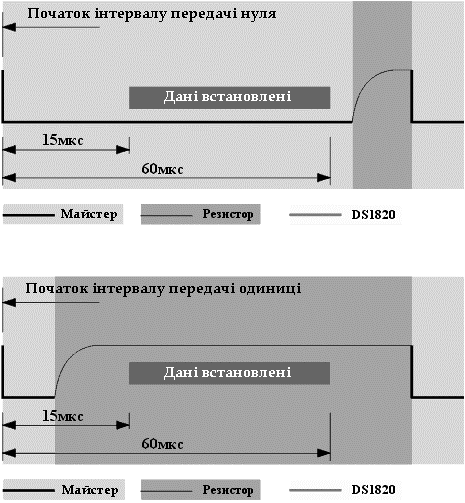
Ініціатором обміну по однодротовій шині завжди виступає майстер. Усі пересилання починаються з процесу ініціалізації. Ініціалізація створюється в наступній послідовності (мал. 2):



###### Малюнок 2 - Ініціалізація обміну по однодротовій шині

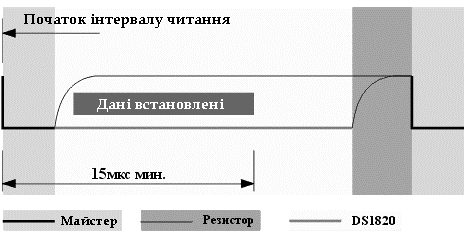
* Майстер посилає імпульс скидання - сигнал низького рівня тривалістю не менш ніж 480 мкс.
* За імпульсом скидання іде відповідь підлеглого пристрою - сигнал низького рівня тривалістю 60 - 240 мкс, що генерується через 15 - 60 мкс після завершення імпульсу скидання.

Відповідь підлеглого пристрою дає майстру зрозуміти, що на шині присутній датчик і він готовий до обміну. Після того, як майстер розпізнав відповідь, він може передати датчику одну з команд. Як правило команда складається з восьми біт. Передача ведеться шляхом формування майстром спеціальних часових інтервалів. Кожен часовий інтервал служить для передачі одного біта. Першим передається молодший біт. Інтервал починається імпульсом низького рівня, тривалість якого лежить у межах 1 - 15 мкс. У підлеглому пристрої запускається схема часової затримки, що визначає момент зчитування даних. Номінальне значення затримки дорівнює 30 мкс, однак, воно може коливатися у межах 15 - 60 мкс. За імпульсом низького рівня іде переданий біт. Він повинен утримуватися майстром на шині протягом 60 - 120 мкс від початку інтервалу. Часовий інтервал завершується переведенням шини у стан високого рівня на час не менш ніж 1 мкс. Потрібно відзначити, що обмеження на цей час згори не накладається. Аналогічним образом формуються часові інтервали для всіх переданих бітів (мал. 3):



Малюнок 3 - Передача біта по однодротовій шині

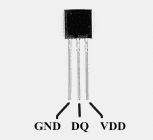
У залежності від посланої майстром команди, можливо, виникне необхідність читати послані помічником дані. При прийманні даних від підлеглого пристрою часові інтервали для бітів, що приймаються, теж формує майстер. Інтервал починається імпульсом низького рівня тривалістю 1 - 15 мкс. Потім майстер повинний звільнити шину, щоб дати можливість датчику вивести біт даних. По переходу з одиниці в нуль помічник виводить на шину біт даних і запускає схему часової затримки, що визначає, як довго біт даних буде присутній на шині. Цей час лежить у межах 15 - 60 мкс. Для того щоб дані на шині, яка завжди має деяку ємність, гарантовано встановилися, потрібен деякий час. Тому момент зчитування даних майстром повинний відстояти якнайдалі, але не більше ніж на 15 мкс від початку часового інтервалу (мал. 4):



Малюнок 4 - Читання біта по однодротовій шині

**2. Інтелектуальні температурні датчики.**

Розглянемо принцип дії і особливості роботи програмуємого термодатчика DS18B20.



###### Малюнок 5 – Зовнішній вигляд датчика температури DS18B20

VDD – зовнішнє живлення

DQ – уведення/вивід даних

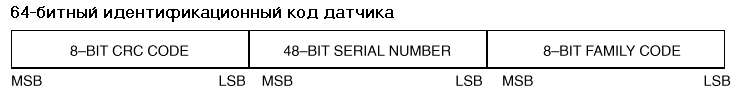
GND – земля.

*2.1. Структура ідентифікаційного коду датчика*

Пам'ять датчика DS18B20 складається з ПЗП, у якому міститься 64-бітовий ідентифікаційний код датчика, надоперативної пам'яті й енергонезалежної електрично стираємої пам'яті, у якій зберігаються значення граничних температур і регістр конфігурації.

Тому що всі помічники спілкуються по одному дроті, то для вказівки, якому помічнику призначається команда і якому помічнику необхідно в даний момент часу відповідати на запити майстра, необхідний індивідуальний ідентифікаційний код, що має кожен датчик фірми Dallas Semiconductor.

Структура цього коду наступна (мал. 6):

****

###### Малюнок 6 - Ідентифікаційний код датчика

Ідентифікаційний номер датчика складається з трьох складових:

* 8-бітний циклічний надлишковий код;
* 48-бітний серійний номер;
* 8-бітний сімейний код.

Розглянемо ці частини більш докладно.

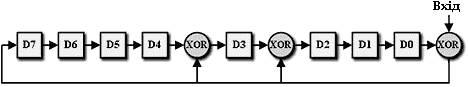
## *Призначення і застосування ЦНК*

ЦНК - це код перевірки на помилку або код контролю циклічної надлишковості (Cyclic Redundancy Code). У випадку якщо прийняті помилкові дані, що визначається CRC кодом, необхідно повторювати операцію читування/запису доти, поки отримана інформація не буде зафіксована як правильна.

В обчисленні байта ЦНК беруть участь перші 7 байт, чи 56 переданих біт. Для обчислення використовується наступний поліном:

CRC = X8+X5+X4+1

Після прийняття даних майстер повинний обчислити ЦНК і порівняти обчислене число з переданим CRC. Якщо ці дані збігаються, то це означає, що прийняття даних пройшло без помилок. Можна також обчислити контрольну суму для всіх 64 прийнятих біт, яка у цьому випадку повинна дорівнювати нулю. Блок-схема алгоритму обчислення контрольної суми показана на мал. 7. Алгоритм використовує операції здвигу і «виключаюче АБО». Квадратиками показані біти змінної, котра використовується для обчислення CRC. Перед обчисленням її необхідно обнулити, а потім на вхід алгоритму потрібно послідовно подати 56 прийнятих біт у тому порядку, у якому вони були прийняті. У результаті змінна буде містити значення CRC.



Малюнок 7 – Блок-схема алгоритму обчислення ЦНК

Такий же алгоритм обчислення ЦНК використовується й у випадку читання надоперативної пам'яті, тільки там ЦНК розрахований для 8-ми байтів даних.

# ***Структура 48-бітного індивідуального коду***

Даний код - це код серійного випуску датчика. Він привласнюється кожному датчику і виключає повторення. При наявності коду з 48 біт можна випустити 281474976710656 однотипних датчиків, виключаючи повторень у коді.

# ***"Сімейний" код і його призначення***

Сімейний код - це код продукції по функціональному призначенню. При 8-бітному значенні цього коду можливо випустити 256 датчиків різних по призначенню і виконуємих функціях. Для датчиків серії DS18B20 сімейний код дорівнює значенню 28h.

*2.2. Вимірювання температури*

По прочитаних з надопертивної пам'яті нульовому і першому байтам може бути визначена температура середовища, що оточує датчик. Старша тетрада цього двобайтного числа визначає знак температури. Якщо тетрада містить нулі, то знак “+”, якщо одиниці, то знак “-”. Якщо температура позитивна, то наступні три тетрады необхідно розглядати в такий спосіб. Дві старші з цих трьох тетрад представляють байт, що визначає цілу частину значення температури. Молодша тетрада відповідає за дробову частину значення температури: одиниця в нульовому біті – +0.0625 до значення температури; у першому біті – +0.125; у другому – +0.25; у третьому – +0.5. Якщо температура негативна, то двобайтне число необхідно інвертувати і додати до нього одиницю, а далі розглядати так як і у випадку позитивної температури, не забуваючи про негативний знак.

У табл. 1 приведена відповідність вихідних даних і виміряної температури для 12-бітної дискретності вимірювання. Якщо встановлена менша дискретність, то незначущі біти будуть містити нулі.

Таблиця 1. Зв'язок вихідних даних з температурою для DS18B20

|  |  |  |
| --- | --- | --- |
| Температура | Двійковий вихід | Шістнадцятковий вихід |
| +125ºC | 0000 0111 1101 0000 | 07D0h |
| +25.0625ºC | 0000 0001 1001 0001 | 0191h |
| +10.125ºC | 0000 0000 1010 0010 | 00A2h |
| +0.5ºC | 0000 0000 0000 1000 | 0008h |
| 0ºC | 0000 0000 0000 0000 | 0000h |
| -0.5ºC | 1111 1111 1111 1000 | FFF8h |
| -10.125ºC | 1111 1111 0101 1110 | FF5Eh |
| -25.0625ºC | 1111 1110 0110 1111 | FE6Fh |
| -55ºC | 1111 1100 1001 0000 | FC90h |

*2. 3. Система команд*

Будь-який обмін даними з датчиком повинний складатися з трьох етапів:

* ініціалізація;
* виконання однієї з команд функції ПЗП;
* виконання однієї з команд функції пам'яті.

Дана послідовність може бути перервана на будь-якому етапі імпульсом скидання.

Розглянемо деякі команди, що знадобляться нам для виконання даної роботи.

# ***Команди функції ПЗП***

Як тільки майстер шини визначив наявність приєднаних помічників, він може видати одну з команд функції ПЗП. Усі команди функції ПЗП мають довжину 8 біт. Молодший біт передається першим. Список цих команд наступний:

## Читання ПЗП [33h]

Команда дозволяє майстру шини прочитати 64-бітовий ПЗП помічника. Її можна використовувати, якщо до шини підключений тільки один помічник. Якщо до шини підключено більше помічників, то при використанні цієї команди може виникнути конфлікт даних, коли всі помічники відразу почнуть передачу. При цьому буде отриманий результат, еквівалентний монтажному множенню.

## Вибір ПЗП [55h]

Ця команда, якщо за нею іде 64-бітовий код, дозволяє майстру адресуватися до конкретного пристрою. Тільки той помічник, чий код відповідає посланої майстром 64-бітової послідовності, відгукнеться на наступну команду функції пам'яті. Всі інші помічники будуть чекати імпульсу скидання. Команда може використовуватися як при єдиному підключеному до шини помічнику, так і у випадку, коли їх багато.

## Ігнорування ПЗП [CCh]

Команда заощаджує час у випадку, якщо до шини підключений тільки один помічник. Вона дозволяє майстру звернутися до функцій пам'яті без попереднього генерування 64-бітового коду ПЗП. Якщо на шині присутні декілька помічників і після команди CCh іде команда читання, виникне конфлікт даних, тому що всі помічники будуть передавати одночасно. Буде отриманий результат, еквівалентний монтажному множенню.

# ***Команди функції пам'яті***

Після успішного виконання команди функції ПЗП майстер може видати одну з команд функції пам'яті.

## Перетворення температури [44h]

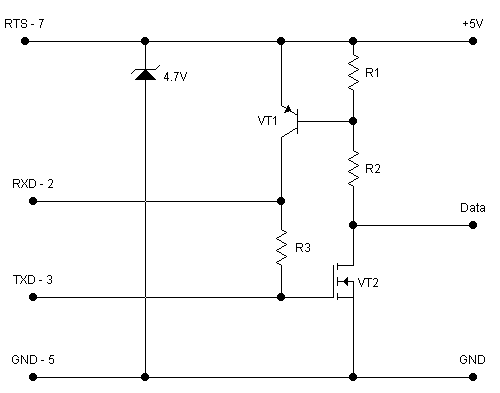
Команда починає перетворення температури. При цьому ніяких додаткових даних не потрібно. Буде зроблене перетворення температури в код і датчик перейде в режим очікування. Датчик видасть на шину “0” на період часу, поки він буде зайнятий перетворенням температури; коли процес перетворення завершиться, датчик поверне на шину “1”.

Читання надоперативної пам'яті [BEh]

Команда читає зміст надоперативної пам'яті. Читання починається з нульового байта і закінчується дев'ятим, що містить ЦНК. Якщо не потрібно читати всі байти, майстер може припинити читання в будь-який момент видачею імпульсу скидання.

**3. Сполучення персонального комп'ютера з мережею MicroLAN через COM порт**

Схема розроблялася для підключення до COM порту комп'ютера максимальної кількості датчиків температури DS18B20, що використовують інтерфейс MicroLAN. Число датчиків, що підключаються, необмежено, одночасний вимір температури можуть проводити до 10 датчиків, сумарна відстань мережі MicroLAN до 300м. При одночасній передачі інформації вхідний і вихідний потік даних накладається за принципом монтажне множення. Схема адаптера представлена на малюнку 8:



###### Малюнок 8 – Схема узгоджуючого адаптера

*Опис схеми:*

* стабілітрон обмежує максимальну напругу присутню в схемі, слаботочний стабілітрон на 4.7 В при струмі 10..15 мА забезпечує вихідну напругу 4.9..5.1 В, що є оптимальним рівнем для сполучення з ТТЛ мікросхемами;
* R1+R2, включені послідовно, підтягують лінію даних до рівня логічної одиниці, усі виходи повинні бути підключені за принципом “загальний колектор”;
* транзистор VT2 використовується для передачі сигналів від лінії TXD у лінію даних ТТЛ рівня;
* транзистор разом з резисторами R1-R3 перетворює сигнал із ТТЛ рівнями 0..5В у сигнал з рівнями +5..-12В необхідний для роботи COM порту. -12В є присутнім у пасивному стані на виході TXD, і при закритому VT1 на вхід RXD надходить -12В. Якщо на вході DATA рівень напруги менше 2.5В, то напруга транзистора база-емітер зростає до 0.7В, транзистор відкривається і на вхід RXD надходить сигнал відповідний логічній одиниці (більш ніж 2..3В).

**4. Опис програми для роботи з інтелектуальними температурними датчиками**

Інтерфейс користувача складається з трьох частин: головного меню, вікна введення команд і допоміжної панелі в правій частині, у якій відбивається допоміжна інформація.

***Головне меню***

Головне меню складається з чотирьох розділів: "Файл", "Порт", "Режим" і "Допомога".

*Розділ "ФАЙЛ":*

"Очистити робочу область" - очищає вікно введення команд і допоміжну панель;

"Вихід" - вихід із програми.

*Розділ "ПОРТ"* містить у собі перелік COM-портів, якими володіє комп'ютер і які не зайняті іншими пристроями. Порт, за допомогою якого здійснюється діалог з датчиком (датчиками), позначений прапорцем. Якщо жоден порт не обраний, то робота з датчиком (датчиками) неможлива. Обраний порт чи відсутність вибору при виході з програми зберігається в INI-файлі, і при наступному вході програма буде намагатися відкрити цей порт. При неможливості зробити це на екран буде видане відповідне повідомлення.

*Розділ "РЕЖИМ"* містить два пункти: "Загальний" і "Детальний", причому тільки один з них може бути обраний.

"Загальний" - користувач має можливість у більш наочній формі вести спілкування з датчиком (датчиками), після введення команди автоматично приймати дані й одержувати коментарі до прийнятих даних.

"Детальний" - усі проблеми прийняття-передачі даних лягають на плечі користувача.

Більш детально робота у вищевказаних режимах буде розглянута далі.

*Розділ "ДОПОМОГА"* дозволяє викликати довідку, а також подивитися схему використовуваного адаптера.

***Система команд***

Команди для реалізації діалогу з датчиком (датчиками) вводяться у вікні введення команд. При введенні команди не враховується регістр, а також не беруться до уваги пробіли. Розглядається тільки останній рядок. У залежності від режиму роботи система команд може мати два різних вигляди. Розберемо системи команд для кожного з режимів роботи, ілюструючи застосування команд прикладами.

### Режим "ЗАГАЛЬНИЙ"

##### Ініціалізація

RST або RESET - реалізує ініціалізуючу послідовність: подається імпульс скидання і якщо отриманий імпульс присутності, то видається повідомлення "Імпульс присутності отриманий", інакше буде повідомлення "Датчики відсутні на лінії".

**rst**

**Імпульс присутності отриманий**

##### Команди функції ПЗП

## *Читання ПЗП [33h]*

Якщо дані прочитані вірно, то ПЗП помічника відіб'ється на допоміжній панелі.

**33**

**88 h - ЦНК (дані прочитані вірно)**

###### **000000415E8F h - серійний номер**

**28 h - код сімейства**

## *Вибір ПЗП [55h]*

Код ПЗП задається в шістнадцятковому виді, починаючи з молодшого байта ПЗП.

**55 288F5E4100000088**

**Обрано датчик з кодом 88000000415E8F28**

## *Ігнорування ПЗП [CCh]*

**CC**

**Ігнорування ПЗП**

##### Команди функції пам'яті

*Читання надоперативної пам'яті [BEh]*

У результаті видається розписаний побайтно зміст НОП.

###### **BE**

**B0 h - ЦНК (дані прочитані вірно)**

###### 1006FF h - зарезервовані

**1F h - байт конфігурації**

**20 h = 32 °C - нижній поріг температур**

**30 h = 48 °C - верхній поріг температур**

###### **01EA h = 30,6250 °C - температура**

## *Перетворення температури [44h]*

**44**

**Перетворення температури закінчено**

Приклад визначення температури при одному датчику:

**rst**

**Імпульс присутності отриманий**

**CC**

**Ігнорування ПЗП**

**44**

###### **Перетворення температури закінчено**

**rst**

**Імпульс присутності отриманий**

**cc**

**Ігнорування ПЗП**

**BE**

**42 h - ЦНК (дані прочитані вірно)**

**1008FF h - зарезервовані**

**1F h - байт конфігурації**

**20 h = 32 °C - нижній поріг температур**

**30 h = 48 °C - верхній поріг температур**

**01C8 h = 28,5000 °C - температура**

### *Режим "ДЕТАЛЬНИЙ"*

Ініціалізація в цьому режимі аналогічна попередньому режиму.

Передача даних здійснюється в такий спосіб. Будь-яке введене шістнадцяткове число розбивається на байти і побайтно передається починаючи з молодшого біта. Відсутні до байта позиції заповнюються нулями. Якщо введено "1" або "0", то в лінію передається відповідний біт.

Прийняття даних здійснюється за допомогою двох команд "RBYTE" і "RBIT" - прочитати байт і біт відповідно. Якщо після букви "R" вставити будь-яке ціле позитивне число, то стільки байт або біт буде прочитано. Наприклад, R8BYTE - прочитати 8 байт. При цьому прочитані байти будуть відбиватися в шістнадцятковому виді, а прочитані біти - у двійковому.

Приклад визначення температури при одному датчику:

**rst**

**Імпульс присутності отриманий**

**CC**

**Кодова посилка здійснена**

**44**

**Кодова посилка здійснена**

**rst**

**Імпульс присутності отриманий**

**CC**

**Кодова посилка здійснена**

**BE**

**Кодова посилка здійснена**

**R9BYTE**

**C80130201FFF081042 h**

Таким чином, поточна температура – 01С8=28,5ºC.

1. Порядок виконання роботи
2. Увійти в загальний режим.
3. Визначити ПЗП датчика. Перевірити чи є він датчиком температури DS18B20.
4. По відомому ПЗП звернутися до датчика, задавши команду перетворення температури.
5. Ігноруючи ПЗП датчика, прочитати вміст надоперативної пам'яті, і по перших двох байтах визначити поточну температуру.
6. Увійти в детальний режим.
7. Проробити те ж, що і при загальному режимі.

**Зміст звіту**

1. Опис команд датчика.
2. Отримані результати.

###### **Лабораторна робота №2**

# **Дослідження інтелектуального датчика DS18B20**

**Ціль роботи:** ознайомитися з побудовою і принципами роботи інтелектуальних температурних датчиків DS18B20, дослідити статичну характеристику й оцінити похибку датчика.

# **Теоретичні відомості**

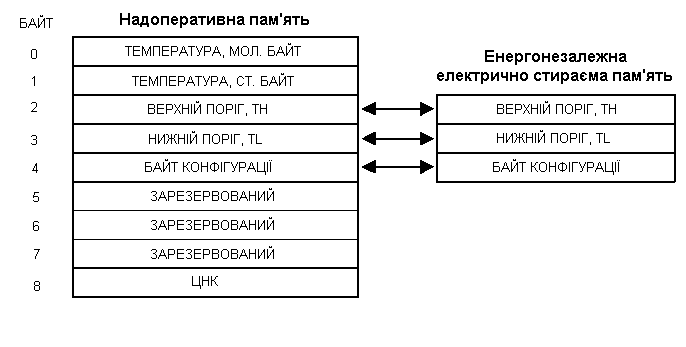
Деякі властивості і команди термодатчика DS18B20 були розглянуті в лабораторній роботі №1. Розглянемо інші характеристики датчика, а також команди роботи з пам'яттю.

**1. Структура пам'яті датчика DS18B20**

Пам'ять датчика DS18B20 складається з ПЗП, у якому міститься 64-бітовий ідентифікаційний код датчика, надоперативної пам'яті й енергонезалежної електрично стираємої пам'яті, у якій зберігаються значення граничних температур і регістр конфігурації.

Надоперативна пам'ять організована у вигляді восьми байтів пам'яті (мал. 1). Перші два байти містять інформацію про обмірювану температуру (молодший і старший байт відповідно). Третій і четвертий байти є тимчасовими копіями тригерів TH і TL, що відповідають верхньому і нижньому порогам температур. Ці байти очищаються при кожному вимиканні живлення. П'ятий байт є копією регістра конфігурації. Він теж очищається при вимиканні живлення. Останні три байти використовуються для внутрішніх потреб датчика і для користувача не мають значення.

У надоперативній пам'яті мається ще і дев'ятий байт. Він містить ЦНК попередніх восьми байтів.

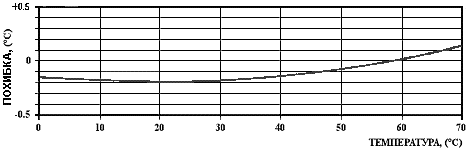


Малюнок 1 – Структура надоперативної і енергонезалежної

пам'яті датчика DS18B20

**2. Похибка вимірювання температурного датчика**

Термодатчик DS18B20 забезпечує вимірювання температури в діапазоні –55..+125°C, але датчик температури, як і всякий інший прилад, має деяку похибку. Гарантована точність складає ±0.5°C у діапазоні –10..+85°C і ±2°C у всьому діапазоні робочих температур. Типова крива похибки вимірювання температури наведена на мал. 2.



###### Малюнок 2 – Типова крива похибки термодатчика DS18B20

Незважаючи на обмежену абсолютну точність, мала дискретність представлення температури є дуже бажаною, тому що дуже часто на практиці вимагаються відносні виміри. Для цього датчик DS18B20 може бути запрограмований на визначену дискретність перетворення температури.

**3. Регістр конфігурації.**

Датчик DS18B20 може бути запрограмований на задану дискретність перетворення температури. Для цього його надоперативна пам'ять оснащена байтом регістра конфігурації, у який записується код необхідної температурної дискретності. Біти цього регістра організовані, як показано на мал. 3.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| TM | R1 | R0 | 1 | 1 | 1 | 1 | 1 |

###### Малюнок 3 – Байт конфігурації

Розряди з нульового по четвертий завжди містять одиницю і перезапису не підлягають. TM у старшому розряді означає “Test Mode Bit” – біт режиму тестування. Якщо він містить 0, то датчик сконфігурований на робочий режим, а якщо 1, то – на режим тестування. При виготовленні приладу цей біт встановлюється в 0, і змінювати його не треба. R0, R1 – біти, що встановлюють здатність дискретизациї. У табл. 1 наведена залежність здатності дискретизації термометра і часу перетворення температури в код від вмісту цих двох бітів. Прилад випускається сконфігурованим на максимальну здатність дискретизації (12-бітове перетворення), тобто в бітах R0 і R1 записані одиниці.

##### Таблиця 1. Установка бітів дискретизації

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| R1 | R0 | Дискретизація термометра, біти | Дискретизація термометра, ºC | Максимальний час перетво-рення, мс |
| 0 | 0 | 9 | 0.5 | 93.75 |
| 0 | 1 | 10 | 0.25 | 187.5 |
| 1 | 0 | 11 | 0.125 | 375 |
| 1 | 1 | 12 | 0.0625 | 750 |

**4. Аварійна сигналізація**

Після кожного перетворення температури отримане значення порівнюється зі значеннями граничних температур, записаними у тригерні регістри TL і TH енергонезалежної пам'яті. Оскільки ці регістри 8-бітові, то ігноруються чотири молодших біти, тобто порівняння виробляється з точністю 1ºC. Старший значущий біт регістрів TL і TH прямо відповідає знаковому біту 16-бітового регістра температури. Якщо результат вимірювання температури вище TH або нижче TL, усередині датчика встановлюється прапор тривоги. Цей прапор модифікується після кожного вимірювання температури. Поки цей прапор установлений, датчик буде відгукуватися на команду аварійного опитування. Це дозволяє включати паралельно багато датчиків, що здійснюють одночасне температурне перетворення. Якщо на якому-небудь датчику або датчиках температура вийшла за встановлені межі, вони можуть бути ідентифіковані без необхідності читання інших датчиків, де температура знаходиться в межах норми.

**5. Команди для роботи з пам'яттю і відгуку на аварійне опитування**

## Аварійне опитування [ECh]

Датчик буде відгукуватися на цю команду тільки в тому випадку, якщо при останньому температурному вимірюванні був зафіксований вихід за межі обраних порогів. Дана команда відноситься до команд функції ПЗП.

# *Команди функції пам'яті*

Запис у надоперативну пам'ять [4Eh]

По цій команді у надоперативну пам'ять записуються три наступних за нею байта в адресні позиції з 2-ї по 4-у. Ці три байти повинні бути записані всі разом, “пакетом”, до видачі найближчого імпульсу скидання.

## Копіювання надоперативної пам'яті [48h]

Команда копіює вміст надоперативної пам'яті в енергонезалежну пам'ять, зберігаючи граничні температури і регістр конфігурації. Датчик видасть на шину “0” на період часу, поки він буде зайнятий копіюванням; коли процес копіювання завершиться, датчик поверне на шину “1”.

Виклик вмісту енергонезалежної пам'яті [B8h]

Команда викликає значення граничних температур і регістра конфігурації, збережених в енергонезалежній пам'яті у надоперативну пам'ять. Датчик видасть на шину “0” на період часу, поки він буде зайнятий копіюванням; коли процес копіювання завершиться, датчик поверне на шину “1”.

Приклади використання даних команд у програмі для роботи з інтелектуальними температурними датчиками:

## Аварійне опитування [ECh]

**EC**

**88000000415E8F28 h**

Запис у надоперативну пам'ять [4Eh]

Перший байт – верхній поріг температур, другий – нижній поріг температур, третій – байт конфігурації.

###### **4E 30201F**

**У НОП записано 1F2030**

## Копіювання надоперативної пам'яті [48h]

**48**

###### **Вміст НОП скопійований в Е-пам’ять**

Виклик вмісту енергонезалежної пам'яті [B8h]

**B8**

**Вміст Е-пам’яті скопійований у НОП**

1. Порядок виконання роботи
2. Встановити пороги температур для аварійної сигналізації: нижній – 0 0С; верхній – 50 0С.
3. Встановити максимальну дискретність вимірювання температури.
4. Перевірити правильність попередніх установок.
5. Включити термостат.
6. Зняти статичну характеристику термодатчика з інтервалом у 5 хвилин, контролюючи показання по лабораторному термометру, зафіксувавши при цьому температуру спрацьовування аварійної сигналізації.
7. Побудувати статичну характеристику.
8. Побудувати залежність похибки вимірювання від температури.
9. Оцінити максимальну, абсолютну і відносну похибки вимірювання температури.

## Зміст звіту

1. Опис послідовності виконання роботи.
2. Статична характеристика термодатчика.
3. Висновки по точності вимірювання датчика і її залежності від вимірюваної температури.

###### **Лабораторна робота №3**

**Дослідження мережі MicroLAN із груповим підключенням датчиків температури**

**Ціль роботи:** ознайомитися з методикою визначення кодів температурних датчиків, підключених до шини.

# **Теоретичні відомості**

**1. Команда “Дослідження ПЗП”**

До складу системи команд датчиків температури DS18B20 входить команда функції ПЗП, що дозволяє визначити кількість і коди підключених датчиків.

## Дослідження ПЗП [F0h]

При першому включенні системи майстер шини може не знати, скільки пристроїв підключено до шини, а також їхні коди. Дана команда дозволяє майстру методом виключення ідентифікувати ПЗП-коди всіх підключених помічників.

Процес дослідження ПЗП являє собою повторення 3-крокового алгоритму: прочитати біт, прочитати доповнення до цього біта (протилежне значення), записати потрібне значення цього біта. Майстер шини застосовує цей 3-кроковий алгоритм до кожного біта 64-бітового ПЗП. Після проходження всіх 64 бітів майстер знає вміст ПЗП одного пристрою. Коди інших пристроїв виділяються повторними проходженнями.

Наступний приклад припускає, що до шини підключені 3 датчики. У ПЗП цих датчиків зашиті такі дані:

ПЗП1 011001…

ПЗП2 111011…

ПЗП3 010011…

Процес дослідження буде наступним:

1. Майстер шини починає процес ініціалізації видачею імпульсу скидання. Помічники відгукуються одночасною видачею імпульсів присутності.
2. Майстер видає в шину команду F0h.
3. Майстер читає один біт з шини. ПЗП1 і ПЗП3 розмістять на шині нулі, а ПЗП2 – одиницю. Результатом буде логічне множення, і майстер побачить нуль. Потім майстер ще раз читає біт. ПЗП1 і ПЗП3 розмістять одиниці, а ПЗП2 – нуль. Майстер знову побачить нуль. Дані, отримані після двох читань, інтерпретуються в такий спосіб:

00 На шині є пристрої, що конфліктують по першому біту.

01 Усі пристрої мають нульовий біт у першій позиції.

1. Усі пристрої мають одиничний біт у першій позиції.
2. До шини не підключений жодне пристрій.
3. Майстер записує нуль. Це виключає з подальшого аналізу ПЗП2, залишаючи приєднаними до шини ПЗП1 і ПЗП3.
4. Майстер проводить ще два читання й одержує “10”. Це означає, що всі пристрої, що залишилися підключеними до шини, мають у другому біті ПЗП одиниці.
5. Майстер записує одиницю, щоб ці пристрої залишилися підключеними до шини.
6. Майстер проводить два читання й одержує “00”. Це означає, що пристрої, що залишилися підключеними, в третьому біті ПЗП мають і “0”, і “1”.
7. Майстер записує нуль. Це виключає з подальшого розгляду ПЗП1 і залишає підключеним тільки ПЗП3.
8. Майстер читає інші біти ПЗП3 і, якщо потрібно, працює з цим датчиком. На цьому закінчується перший прохід, у результаті якого виявляється цілком ідентифікованим один датчик.
9. Майстер починає нову послідовність для дослідження ПЗП, повторюючи кроки з 1 по 7, ідентифікуючи в такий спосіб ПЗП1 і ПЗП3 і виключаючи із розгляду ПЗП2.
10. Майстер записує одиницю в третій біт. Це виключає з подальшого розгляду ПЗП3 і залишає підключеним тільки ПЗП1.
11. Майстер читає інші біти ПЗП1 і, якщо потрібно, працює з цим датчиком. На цьому закінчується другий прохід, у результаті якого виявляється цілком ідентифікованим ще один датчик.
12. Майстер починає нову послідовність для дослідження ПЗП, повторюючи кроки з 1 по 3.
13. Майстер записує одиницю. Це виключає з подальшого розгляду ПЗП1 і ПЗП3 і залишає підключеним тільки ПЗП2.
14. Майстер читає інші біти ПЗП2 і, якщо потрібно, працює з цим датчиком. На цьому закінчується третій прохід, у результаті якого виявляється цілком ідентифікованим ще один датчик.

Аналогічний алгоритм має команда “Аварійне опитування”.

**2. Приклади використання команди “Дослідження ПЗП”**

при роботі програми для роботи з інтелектуальними температурними датчиками

## У “Загальному” режимі у результаті виконання команди “Дослідження ПЗП” [F0h] будуть видані коди присутніх пристроїв, а також відображені на допоміжній панелі.

**F0**

**4E00000041BD5628 h**

**88000000415E8F28 h**

Фрагмент дослідження ПЗП датчиків у детальному режимі:

**rst**

**Імпульс присутності отриманий**

**F0**

**Кодова посилка здійснена**

**R2BIT**

**01**

**0**

**Посилка біта здійснена**

**R2BIT**

**10**

**1**

**Посилка біта здійснена**

**R2BIT**

**01**

**0**

**Посилка біта здійснена**

**R2BIT**

**00**

**0**

**Посилка біта здійснена**

**R2BIT**

**10**

**1**

**Посилка біта здійснена**

**.**

**.**

**.**

Звідси висновок – на лінії присутні як мінімум два датчики і конфліктують вони по третьому біті. У цьому фрагменті була визначена частина ПЗП одного з датчиків – 01001…

1. Порядок виконання роботи
2. Включити детальний режим.
3. Визначити кількість датчиків, підключених до шини і їхні коди.
4. Перевірити правильність дослідження, опитавши температуру з отриманих датчиків.
5. Включити загальний режим і перевірити правильність досліджень.

## Зміст звіту

1. Опис послідовності виконання роботи.
2. Встановлені коди датчиків.

###### **Лабораторна робота №4**

# **Дослідження топології мікролокальної мережі**

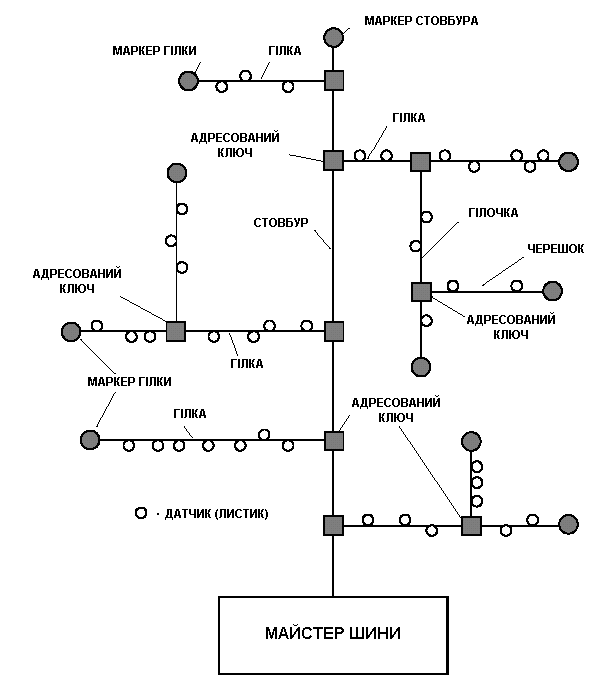
**Ціль роботи:** ознайомитися з побудовою і принципами роботи адресованих ключів DS2405, дослідити топологію мікролокальної мережі.

# **Теоретичні відомості**

**1. Структура мікролокальної мережі**

Мережа MicroLAN не обмежена заздалегідь визначеною структурою. У невеликих конфігураціях вона представляє із себе шинну структуру, з підключенням усіх приладів на одну загальну магістраль. При більш складній конфігурації структура мережі може видозмінюватися в деревоподібну, утримуючу “стовбур” (магістраль) і багато “гілок” (мал. 1). Магістраль безпосередньо підключена до майстра шини. На вершині “стовбура” (віддаленому його кінці) розміщується маркер стовбура. Уздовж “стовбура” розташовуються адресовані ключі (електронні комутатори), через які до “стовбура” підключаються додаткові лінії – “гілки”. Ці “гілки” на своїх віддалених кінцях також мають маркери. “Гілки”, у свою чергу, можуть мати відгалуження – “гілочки”. Від “гілочок” можуть відходити більш дрібні відростки, “черешки”. У будь-якому випадку кожне з розгалужень має на кінці свій маркер. Призначення маркерів – перевіряти, чи надійно здійснюється обмін інформацією у межах даного розгалуження. Застосування маркерів актуально у випадку великих мереж з безліччю пристроїв, що підключаються, і безліччю розгалужень. Пристрої підключаються до будь-якого рівня розгалуження.

Щоб обмінятися інформацією, наприклад, із пристроєм А, підключеним до якої-небудь “гілочки”, майстер шини повинний спочатку включити адресований ключ, що з'єднує “стовбур” з тією “гілкою”, від якої відходить потрібна “гілочка”, а потім уключити ще один ключ, що з'єднує дану “гілку” з потрібною “гілочкою”, на якій знаходиться необхідний “листик” – пристрій А. Таке програмно кероване виборче підключення окремих сегментів мережі дозволяє уникнути перевантаження кабелю, що виникає при одночасному включенні всіх наявних у мережі пристроїв, і, отже, дозволяє будувати досить протяжні і розгалужені мережі.



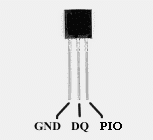
###### Малюнок 1 – Мережа MicroLAN деревоподібної структури

Розглянемо більш докладно невід'ємну складову деревоподібної структури мікролокальної мережі – адресований ключ – на прикладі адресованого ключа DS2405.

**2. Адресований ключ DS2405**

Адресовані ключі являють собою електронні комутатори, призначені для організації розгалужень в однодротовій мережі MicroLAN.

Зовнішній вигляд адресованого ключа представлений на мал. 2.



###### Малюнок 2 – Зовнішній вигляд адресованого ключа DS2405

PIO – порт уведення/виводу

DQ – уведення/вивід даних

GND – земля.

Електрична еквівалентна схема ключа надана на мал. 3. Адресований ключ DS2405 являє собою польовий транзистор n-типу з відкритим стоком, який можна відкривати і закривати шляхом посилки в нього 64-бітового двійкового числа, що збігає зі змістом його ПЗП. Сімейний код для адресованих ключів DS2405 – 05h.

Схема підключення адресованого ключа до однодротової шини мікролокальної мережі приведена на мал. 3.

Ключ не вимагає зовнішнього джерела живлення. Він бере енергію безпосередньо із шини даних, накопичуючи її на спеціальному конденсаторі пасивного живлення в моменти часу, коли на шині присутній високий логічний рівень, і витрачаючи її в ті проміжки часу, коли на шині присутній низький логічний рівень.

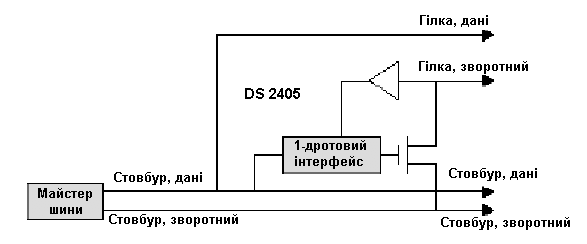
Для обміну даними використовується стандартний протокол однодротового інтерфейсу. Тому що адресований ключ DS2405 містить тільки 64-бітове ПЗП і не має іншої пам'яті, то для спілкування з ним досить команд функції ПЗП. Відповідно до протоколу MicroLAN, майстер шини може видати одну з п'яти команд функції ПЗП:

1. читання ПЗП [33h];
2. вибір ПЗП [55h];
3. дослідження ПЗП [F0h];
4. ігнорування ПЗП [CCh];
5. дослідження ПЗП тільки активних пристроїв [ECh].

Команди читання ПЗП, дослідження ПЗП й ігнорування ПЗП мають той же ефект, що і при роботі з датчиками температури. Команда вибору ПЗП робить переключення виводу PIO на протилежне, тобто якщо ключ був відкритий, то він закриється і навпаки. Початковий стан ключа при подачі живлення – закритий. Команда “Дослідження ПЗП тільки активних пристроїв” є аналогом команди “Аварійне опитування” для датчиків температури з тією лише різницею, що на цю команду відгукнуться тільки ключі, що знаходяться у відкритому стані.

**3. Підключення адресованого ключа до однодротової шини.**

Для зв'язку досить одного проводу (і землі).



Малюнок 3 – Схема підключення адресованого ключа до однодротової шини мікролокальної мережі

На мал. 3 показано, як адресований ключ робить підключення “гілки” до “стовбура” (точно так само відбувається комутація і будь-яких інших ділянок мережі). Прямий дрот (лінія даних) постійно підключений до всіх пристроїв мережі. Зворотний дрот знаходиться в провідному стані тільки для тих ділянок мережі (“гілок” і “гілочок”), що забезпечують шлях до потрібного в даний момент пристрою (“листику”).

1. Порядок виконання роботи
2. Встановити кількість присутніх на лінії датчиків і адресованих ключів, їхні ПЗП.
3. Встановити стан адресованих ключів.
4. Визначити ПЗП датчиків, що знаходяться в гілках, що з'єднуються з мережею за допомогою адресованих ключів.
5. Визначити положення кожного пристрою в структурі мікролокальної мережі, впливаючи на датчики (нагріваючи-прохолоджуючи їх).

## Зміст звіту

1. Опис послідовності виконання роботи.
2. Топологічне дерево мережі з точним положенням кожного пристрою.

###### Навчальне видання

# МЕТОДИЧНІ ВКАЗІВКИ

ДО ВИКОНАННЯ ЛАБОРАТОРНИХ РОБІТ З КУРСУ

“ПРОЕКТУВАННЯ ЗАСОБІВ ТА СИСТЕМ УПРАВЛІННЯ”

ЧАСТИНА 1. МЕРЕЖІ ДЛЯ САУ

для студентів спеціальності

“Комп’ютеризовані системи керування і автоматика”

33

32

5

31

6

30

7

29

8

28

9

27

10

26

11

25

12

24

13

23

14

22

15

21

16

20

17

19

18