

## ЛЕКЦІЯ

### Мікропроцесорна система на базі інтерфейсу 1-Wire

#### *Загальна характеристика*

Однопровідний інтерфейс 1-Wire, розроблений в кінці 90-х років фірмою Dallas Semiconductor, регламентований розробниками для вживання в трьох основних сферах-застосуваннях:

- системи ідентифікації і контролю доступу (технологія iButton або Touch Memory);
- програмування вбудованої пам'яті інтегральних компонентів;
- системи автоматизації (технологія мереж MICROLAN).

Для прийому-передачі інформації використовується одна двонаправлена сигнальна лінія (другий дріт - земляний).

В системах ідентифікації і контролю доступу – (технології Touch Memory) обмін здійснюється в режимі напівдуплекса (або прийом, або передача). Взаємодія приладів по однопровідному інтерфейсу організована за принципом "веде-ведений" (master-slave). При цьому прочитуючий пристрій завжди веде, а один або декілька приладів Touch Memory - ведені. Взаємодія декількох приладів з прочитуючим пристроєм по одній двонаправленій лінії підтримується апаратними засобами Touch Memory.

Групу команд обміну з ПЗП складають чотири команди: читання ПЗП, пропуск, порівняння і пошук. Дві останні команди забезпечують взаємодію

по одній лінії декілька Touch Memory з прочитуючим пристроєм. Команда порівняння ініціює обмін з приладом, серійний номер якого вказаний. Команда пошуку дозволяє визначити серійний номер одного з приладів, підключених до двонапрямленої лінії.

Всі команди обміну мають фіксований розмір - один байт, дані представлені 8-розрядними цілими числами. Ведучій пристрій завжди ініціює обмін, посилаючи команди веденому пристрою.

Для забезпечення цілісності передаваної інформації протокол обміну на фізичному рівні строго регламентує часові параметри сигналів на лінії.

Програмування вбудованої пам'яті інтегральних компонентів забезпечує можливість легкої перебудови функцій напівпровідникових компонентів з малою кількістю зовнішніх виводів, вироблюваних фірмою Dallas Semiconductor

#### *Системи автоматизації на базі мереж MicroLAN.*

MICROLAN є інформаційною мережею, що використовує для здійснення цифрового зв'язку одну лінію даних і один зворотний (або земляний) дріт. (рис. 1.1) Таким чином, для реалізації середовища обміну цієї мережі можуть бути використані, як кабелі, що містять неекрановану виту пару тієї або іншої категорії, так і звичайний телефонний дріт. Подібні кабелі при їх прокладці не вимагають, як правило, наявності спеціального устаткування. Обмеження максимальної довжини однопровідної лінії, що реалізовується без спеціальних додаткових допоміжних пристроїв (повторювачів), регламентоване на рівні 300м.

Основою архітектури мереж MICROLAN, є топологія загальної шини, коли кожен з пристроїв підключений безпосередньо до єдиної магістралі, без яких-небудь каскадних з'єднань або розгалужень. При цьому як базова використовується структура мережі з одним ведучим або майстром і багатьма веденими, хоча існує ряд прийомів організації роботи подібних мереж в режимі мультимастера.

Протокол обміну по цьому інтерфейсу дуже простий і легко реалізується програмно практично на будь-яких МК.



Рис. 1. 1. Схема обміну по 1-wire інтерфейсу

На рис.1.1 показана спрощена схема апаратної реалізації інтерфейсу 1-Wire. Виведення DQ пристрою є входом КМОП-логічного елементу, який може бути зашунтований (замкнений на загальний дріт) польовим транзистором. Шина 1-Wire має бути підтягнута окремим резистором до напруги живлення пристроїв. Опір цього резистора 4.7 К, проте, це значення рекомендоване лише для досить коротких ліній. Якщо шина 1-Wire використовується для підключення віддалених на велику відстань пристроїв, то опір цього резистора слід зменшити. Мінімальний допустимий опір - близько 300 Ом, а максимальний - близько 20-30 кілоом.

Підключення шини 1-Wire до МК на рис. 1.56 показано умовно в двох варіантах: з використанням 2 окремих виводів МК (один як вихід, а інший як вхід), так і одного, що працює і на введення і на вивід.

#### *Обмін інформацією по шині 1-Wire*

1) Обмін завжди ведеться за ініціативою одного ведучого пристрою, який в більшості випадків є мікроконтролером (МК). Для інтерфейсу 1-Wire в

загальному випадку передбачається "гаряче" підключення і відключення пристроїв. Будь-який обмін інформацією починається з подачі імпульсу скидання ("Reset Pulse" або просто RESET) в лінію 1-Wire ведучим пристроєм.

2) Будь-який пристрій, підключений до 1-Wire після надання живлення видає в лінію DQ імпульс присутності, званий "Presence pulse" (далі я використовуватиму термін PRESENCE). Цей же імпульс пристрій завжди видає в лінію, якщо виявить сигнал RESET.

3). Поява в шині 1-Wire імпульсу PRESENCE після видачі RESET однозначно свідчить про наявність хоч би одного підключеного пристрою.

4). Обмін інформації ведеться так званими тайм-слотами: один тайм-слот служить для обміну одним бітом інформації.

5). Дані передаються побайтно, біт за бітом, починаючи з молодшого біта. Достовірність переданих/прийнятих даних (перевірка відсутності спотворень) гарантується шляхом підрахунку циклічної контрольної суми.

На рис.1.2. показана діаграма сигналів RESET і PRESENCE, з яких завжди починається будь-який обмін даними. Видача імпульсу RESET в процесі обміну служить для дострокового завершення процедури обміну інформацією.

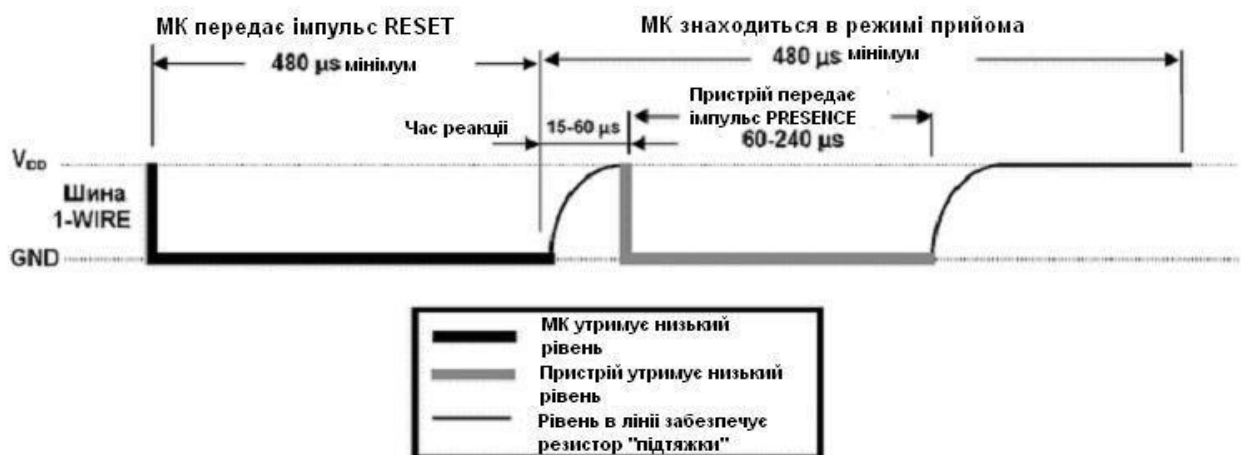


Рис. 1.2. Діаграма сигналів RESET і PRESENCE

Як бачимо, тривалість більшості часових інтервалів дуже приблизна і має лише обмеження лише по мінімуму (не менше вказаного). Умовні позначення ліній, показані на рис. 1.2, використовуватимуться і далі.

Імпульс RESET формує ведучий МК, переводячи в низький логічний рівень шину 1-Wire і утримуючи її в цьому стані мінімум 480 мікросекунд. Потім МК повинен "відпустити" шину. Через деякий час, залежний від ємності лінії і опору підтягуючого резистора, в лінії встановиться високий логічний рівень. Протокол 1-Wire обмежує цей час "релаксації" діапазоном від 15 до 60 мікросекунд, що і є визначальним для вибору підтягуючого резистора (ємність лінії міняти неможливо).

Після прийому імпульсу RESET, ведений пристрій приводить свої внутрішні вузли у вихідний стан і формує у відповідь імпульс PRESENCE, як впливає з рис. 1.2. - не пізніше 60 мікросекунд після завершення імпульсу RESET. Для цього пристрій переводить в низький рівень лінію DQ і утримує її в цьому стані від 60 до 240 мікросекунд. Конкретний час утримання залежить від багатьох параметрів, але завжди знаходиться у вказаному діапазоні. Після цього пристрій так само "відпускає" шину.

Після завершення імпульсу PRESENCE пристрою дається ще деякий час для завершення внутрішніх процедур ініціалізації, таким чином, МК повинен приступити до будь-якого обміну з пристроєм не раніше, чим через 480 мікросекунд після завершення імпульсу RESET.

Итак, процедура ініціалізації інтерфейсу, з якою 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, це означає що на шині 1-Wire немає готових до обміну пристроїв.

Розглянемо процедури обміну бітами інформації, які, здійснюються певними тайм-слотами. Тайм-слот - це жорстко лімітована за часом послідовність зміни рівнів сигналу в лінії 1-Wire. Далі будемо

використовувати термін МК, як синонім "провідного пристрою" і просто "пристрій", як синонім "ведений". Розрізняють 4 типи тайм-слотів: передача "1" від МК, передача "0" від МК, прийом "1" від пристрою і прийом "0" від пристрою.

Кожен тайм-слот завжди починає МК шляхом переведення шини 1-Wire в низький логічний рівень. Тривалість будь-якого тайм-слота повинна знаходитися в межах від 60 до 120 мікросекунд. Між окремими тайм-слотами завжди повинен передбачатися інтервал не менше 1 мікросекунди (конкретне значення визначається параметрами веденого пристрою).

Тайм-слоти передачі відрізняються від тайм-слотів прийому поведінкою МК: при передачі він лише формує сигнали, при прийомі, крім того, ще і опитує (тобто приймає) рівень сигналу в лінії 1-Wire. Рис.1. 3. демонструє часові діаграми тайм-слотів всіх 4-х типів: вгорі показані тайм-слоти передачі від МК, внизу - прийому від пристрою.

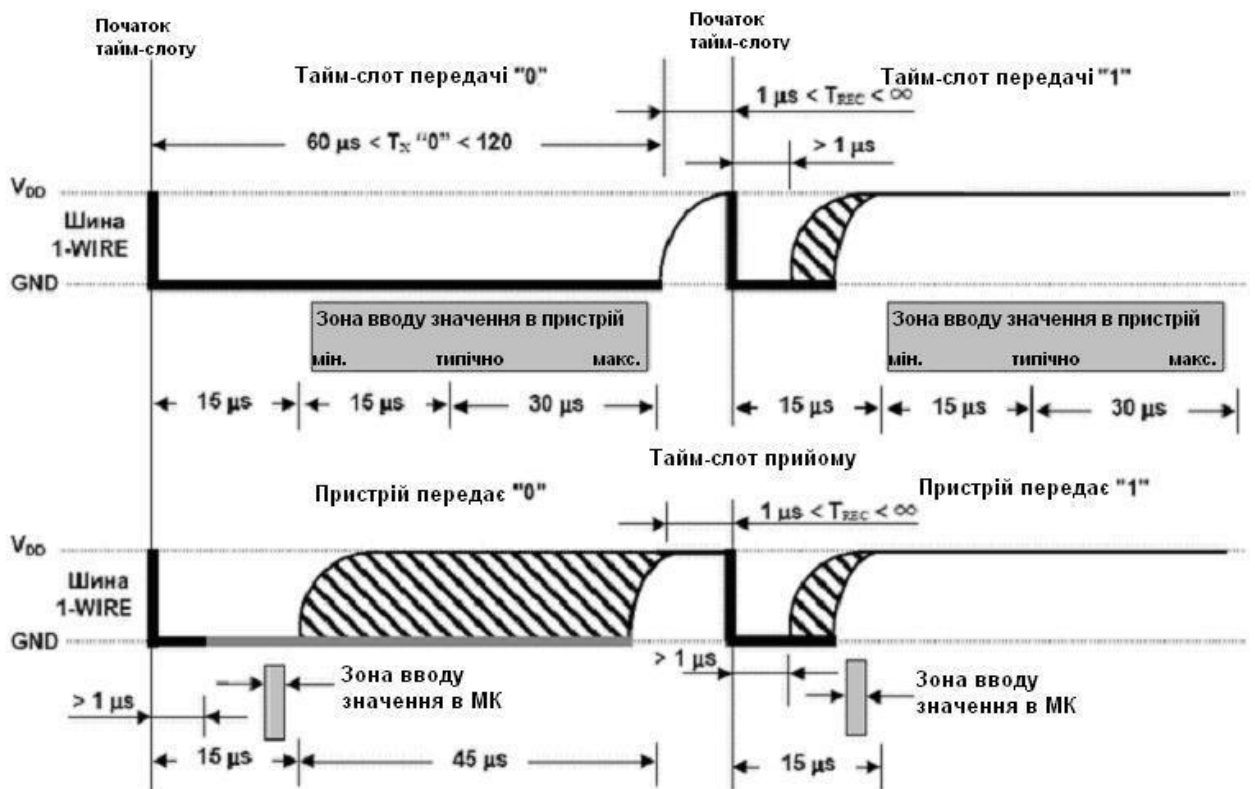


Рис.1.3. Тайм-слоти

Тайм-слот передачі "0" полягає просто в утриманні шини 1-Wire в низькому рівні протягом всієї тривалості тайм-слота. Передача "1" здійснюється шляхом "відпуску" шини 1-Wire з боку МК не раніше чим через 1 мікрорекунду після початку тайм-слота, але не пізніше ніж через 15 мікрорекунд. Ведений пристрій опитує рівень в шині 1-Wire протягом часового інтервалу, умовно показаного у вигляді сірого прямокутника, тобто починаючи з 15-ої мікрорекунди від початку тайм-слота і закінчуючи 60-ою мікрорекундою від початку. Типовий момент введення рівня в пристрій (тобто характерний для більшості пристроїв) - біля 30-ої мікрорекунди від початку тайм-слота.

Заштрихована область - це область "наростання" рівня в шині 1-Wire, яка залежить від ємкості лінії і опору підтягуючого резистора, вона приведена для довідки.

Тайм-слоти прийому інформації відрізняються тим, що МК формує лише початок тайм-слота (абсолютно так само, як при передачі "1"), а потім управління рівнем шини 1-Wire бере на себе пристрій, а МК здійснює введення цього рівня так само в певній зоні часових інтервалів. Зона ця, як видно з табл.1.1, досить мала. Контролер повинен ввести рівень сигналу з лінії на 13-15-ій мікрорекунді від початку тайм-слота.

Таким чином, МК починає тайм слот видачою в шину 1-Wire "0" протягом 1 мікрорекунди. Подальший рівень залежить від типу тайм слота: для прийому і передачі "1" рівень повинен стати високим, а для передачі "0" - залишатися низьким до кінця тайм-слота, тобто не менше 60 і не більше 120 мікрорекунд. Якщо МК приймає дані, то опит рівня в шині він повинен зробити на проміжку від 13-ої до 15-ій мікрорекунді тайм-слота. МК повинен забезпечити інтервал між окремими тайм-слотами не менше 1 мікрорекунди. Слід точно забезпечувати в шині 1-Wire необхідні часові інтервали, оскільки, наприклад, збільшення тривалості тайм-слота виведення "0" зверху рекомендованого значення може привести до помилкового сприйняття цього

тайм-слота, як сигналу RESET. Всі сигнали, які повинен формувати МК, слід формувати за принципом необхідного мінімуму тривалості (тобто трохи більше, чим вказана мінімальна тривалість), а від пристрою слід чекати сигналів за принципом найгіршого (тобто орієнтуватися на самі гірші варіанти часових параметрів сигналу).

Кожен пристрій 1-Wire має унікальний ідентифікаційний 64-бітовим номер, що програмується на етапі виробництва мікросхеми. Унікальний - це означає, що фірма-виробник гарантує, що не знайдеться двох мікросхем з однаковим ідентифікаційним номером (принаймні впродовж декількох десятиків років при існуючих темпах виробництва).

### *Протокол обміну*

При розгляді протоколу обміну вважаємо що на шині 1-Wire є більш ніж один пристрій. В цьому випадку перед МК встають 2 проблеми: визначення кількості наявних пристроїв і вибір (адресація) одного конкретного з них для обміну даними. Вирішення першої проблеми здійснюється двома шляхами: універсальним і гнучким, але вимагаючим досить складного алгоритму, що і простим, при якому відомі номери всіх використовуваних у вашій схемі 1-Wire-пристроїв. Номер деяких пристроїв нанесені на корпусі мікросхем (наприклад, для пристроїв iButton - всім відомих ключів-пігулок), а номери інших можна визначити за допомогою спеціальних програм або пристроїв.

Хай відомі номери всіх пристроїв 1-Wire на шині. Алгоритм наступний: МК посилає, імпульс RESET, і все наявні пристрої видають PRESENCE. Потім МК посилає в шину команду, яку приймають всі пристрої. Команд визначено декілька загальних для всіх типів 1-Wire-пристроїв, а так само можуть бути команди, унікальні для окремих типів. Серед загальних команд найбільш поширені наступні (див. табл. 1.1).



Таблиця 1.1 - Команди інтерфейсу 1-Wire

Команда	Значення байта	Опис
SEARCH ROM	0xF0	Пошук адрес - використовується при універсальному алгоритмі визначення кількості і адрес підключених пристроїв
READ ROM	0x33	Читання адреси пристрою - використовується для визначення адреси єдиного пристрою на шині
MATCH ROM	0x55	Вибір адреси - використовується для звернення до конкретної адреси пристрою з багатьох підключених
SKIP ROM	0xCC	Ігнорувати адресу - використовується для звернення до єдиного пристрою на шині, при цьому адреса пристрою ігнорується (можна звертатися до невідомого пристрою)

Перша команда використовується в складному універсальному алгоритмі, друга дозволяє визначити адресу пристроїв, перед їх установкою в готовий виріб, а дві останні напевно є основними.

Після того, як МК видасть команду READ ROM, від пристрою поступить 8 байт його власної унікальної адреси - МК повинен їх прийняти. Будь-яка процедура обміну даними з пристроєм має бути завершена повністю або перервана посилкою сигналу RESET.

Якщо відправлена команда MATCH ROM, то після неї МК повинен передати так само і 8 байт конкретної адреси пристрою, з яким здійснюватиметься подальший обмін даними. Це рівносильно виставлянню адреси на паралельній шині в мікропроцесорних пристроях. Приймавши цю команду, кожен пристрій порівнює передавану адресу зі своєю власною. Всі пристрої, адреса яких не збіглася, припиняють аналіз і видачу сигналів в лінії 1-Wire, а пристрій, що пізнав адресу, продовжує роботу. Тепер всі дані, передавані МК потраплятимуть лише до цього "адресованому" пристрою. То, які саме дані треба послати в пристрій або отримати від нього після його адресації, залежить від конкретного пристрою Наприклад, для згаданого термометра це

можуть бути команди запуску вимірювання або прочитування результату, для ключа-пігулки не визначені жодні інші команди, окрім основних, а для мікросхем АЦП додаткових команд може бути близько десятка.

Якщо пристрій один на шині то можна прискорити процес взаємодії з ним за допомогою команди SKIP ROM. Отримавши цю команду, пристрій відразу вважає адресу такою, що збіглася, хоча жодної адреси за цією командою не слідує. Деякі процедури не вимагають прийому від пристрою жодних даних, в цьому випадку команду SKIP ROM можна використовувати для передачі якоїсь інформації відразу всім пристроям. Це можна використовувати, наприклад, для одночасного запуску циклу вимірювання температури декількома датчиками-термостатами типа DS18S20.

Прийом і передача байтів завжди починається з молодшого біта. Порядок дотримання байтів при передачі і прийомі адреси пристрою так само ведеться від молодшого до старшого. Порядок передачі іншої інформації залежить від конкретного пристрою, тому слід звертатися до документації на вживані вами пристрої.

Унікальна 64-бітова номер-адреса пристроїв 1-Wire складається з 8 байт: одного байта ідентифікатора сімейства, шести байт (48 біт) власне унікальної адреси і одного байта контрольної суми всіх попередніх байтів.

Контрольна сума або CRC - це байт, значення якого передається найостаннішим і обчислюється по спеціальному алгоритму на основі значення всіх 7 попередніх байтів. Алгоритм підрахунку такий, що якщо всі байти передані-прийняті без спотворень, прийнятий байт контрольної суми обов'язково збіжиться з розрахованим в МК (або пристрої) значенням. Тобто при реалізації програмного алгоритму обміну інформацією необхідно при передачі і прийомі байтів підраховувати їх контрольну суму по строго певному алгоритму, а потім або передати набутого значення, або порівняти розрахункове значення з набутого значення CRC. Лише при збігу обох CRC МК або пристрій вважають прийняті дані достовірними. Інакше продовження

обміну неможливе.

Вочевидь, що алгоритм підрахунку CRC має бути однаковим як для МК, так і для будь-якого пристрою. Він "стандартизований" і описаний в документації.

"Understanding and Using Cyclic Redundancy Checks with Maxim integrated iButton™ Products" Приклад програмної реалізації CRC алгоритму на мові асемблера.

Ця підпрограма використовує один байт пам'яті CRC для зберігання результату. Перед першим викликом цей байт необхідно обнулити. У акумуляторі - черговий прийнятий або передаваний байт. Після того, як всі байти передані/прийняті у комірку пам'яті CRC вийде контрольна сума.

Підпрограма не псує жодних регістрів, окрім регістра стану.

Використання цієї (та і подальшої) підпрограми дуже просте: перед початком прийому або передачі треба обнулити комірку CRC, а потім кожен прийнятий або переданий байт помістити в акумулятор і викликати цю підпрограму. Після того, як прийняті всі 8 (звернете увагу - саме 8!) байтів унікальної адреси пристрою, необхідно перевірити вміст комірки CRC: ненульове її значення свідчить про наявність спотворення прийнятих даних. Якщо ж  $CRC=0$  - це означає, що дані прийняті без спотворень. Якщо ж МК вів передачу унікальної адреси пристрою, то вміст CRC має бути переданий 8-м байтом після попередніх семи.

Таким чином:

- будь-який обмін інформацією починається з передачі імпульсу RESET і прийому імпульса PRESENCE;
- якщо імпульсу PRESENCE не виявлено - на шині немає пристроїв;

- МК завжди ініціює обмін, починаючи кожен тайм-слот обміну бітом інформації;
- часові параметри кожного тайм-слота слід дотримувати з максимально можливою точністю;
- для вибору одного з багатьох пристроїв на шині 1-Wire МК повинен передати в шину команду MATCH ROM і потім 8 байт адреси пристрою, останній (8-й) байт цієї адреси - є контрольна сума попередніх семи;
- якщо пристрій на шині один - МК може взяти його адресу шляхом посланки команди READ ROM, після чого прийняти від пристрою 8 байтів адреси, останній з яких так само буде контрольною сумою перших семи;
- для роботи з єдиним пристроєм на шині можна відмовитися від вказівки його адреси, для цього МК повинен передати пристрою команду SKIP ROM, після чого можна починати звичайний обмін даними;
- будь-яка почата процедура обміну може тривати скільки завгодно довго за рахунок пауз між окремими тайм-слотами, але завжди має бути завершена повністю;
- перервати початий обмін можна у будь-який момент шляхом видачі імпульсу RESET в шину 1-Wire (але це може порушити нормальну роботу деяких пристроїв).