

Розділ 8 CPU 21xDP

Короткий огляд Цей розділ описує застосування CPU 21xDP і його використання в мережах PROFIBUS. Ви отримаєте докладну інформацію по конфігуруванню PROFIBUS-DP слейвів. Закінчує розділ приклад реалізації мережі між CP 21xDP та CPU 21xDPM.

Зміст	Тема	Сторінка
	Розділ 8 CPU 21xDP	8-1
	Базові положення	8-2
	Конфігурування CPU 21xDP	8-7
	Параметри DP-слейва	8-12
	Діагностичні функції.....	8-15
	Внутрішні статусні повідомлення CPU	8-18
	Інструкція по створенню мережі PROFIBUS.....	8-20
	Введення в експлуатацію	8-23
	Приклад.....	8-25

Базові положення

Загальна інформація

PROFIBUS являється відкритим шинним стандартом, який широко застосовується в будівництві, виробництві та автоматизації технологічних процесів. PROFIBUS визначає технічні і функціональні властивості послідовної мережі, яка використовується для створення розподіленої системи електронних засобів автоматизації на нижньому (датчик/виконавчий механізм) та середньому рівні виробництва (процеси).

PROFIBUS складається з різних сумісних версій. В даному розділі докладно буде описуватись версія мережі PROFIBUS-DP.

PROFIBUS-DP

PROFIBUS-DP особливо розповсюджений у виробничих системах автоматизації. Специфікація DP має велику швидкодію, підтримує технологію Plug&Play і є економічно ефективною альтернативою каналам паралельної передачі даних між PLC та децентралізованою/розподіленою периферією.

PROFIBUS-DP розроблено для високошвидкісного обміну даних на рівні датчик-виконавчий механізм.

Обмін даними виконується циклічно. Під час одного шинного циклу майстер читає вхідні дані з підключених слейвів і записує нові значення на їх виходи.

Майстер і слейви

Шина PROFIBUS складається з активних станцій (майстрів) і пасивних станцій (слейвів).

Майстер

Майстер контролює потік даних на шині. На одній PROFIBUS-шині може бути декілька майстрів. Така топологія називається мульти-майстерною. На базі інтелектуальних пристроїв, підключених до шини, мережевий протокол встановлює логічну петлю по якій передається маркер.

Майстер, такий як CPU 21xDPM, може розсилати повідомлення по шині, лише якщо він володіє маркером (Token). У протоколі PROFIBUS майстри також називаються активними станціями.

Слейв

Зазвичай ці пристрої використовуються для підключення периферійного обладнання, такого як датчики, виконавчі механізми і перетворювачі. VIPA PROFIBUS є модульною слейвовою системою, яка передає дані між периферією System 200V і майстер-станціями.

Відповідно до стандарту PROFIBUS ці пристрої не мають доступу до шини. Вони можуть лише підтверджувати отримання повідомлення або передавати повідомлення до майстра на його прохання. Слейви також називаються пасивними станціями.

Комунікація

Для доступу до шини мережевий комунікаційний протокол може реалізувати дві процедури:

**Майстер з
Майстром**

Зв'язок з майстром в літературі також згадується як процедура естафетної передачі. Маркер, що передається між станціями, керує доступом до мережі. Маркер - це специфічне повідомлення, яке циркулює по шині.

Коли майстер володіє маркером, він має право доступу до шини і може обмінюватись даними з усіма іншими активними і пасивними станціями. Час утримання маркера визначається при конфігуруванні системи. Коли час утримання маркера закінчується, він передається до наступного майстра, який набуває право доступу до шини.

**Процедура
Майстер-
Слейв**

Обмін даними між майстром і слейвом відбувається в чітко визначених циклах. При конфігуруванні системи визначаються усі можливі зв'язки пасивних і активних станцій. Ви повинні вказати які DP-слейви будуть приймати участь в циклічному обміні даними програми, а які ні.

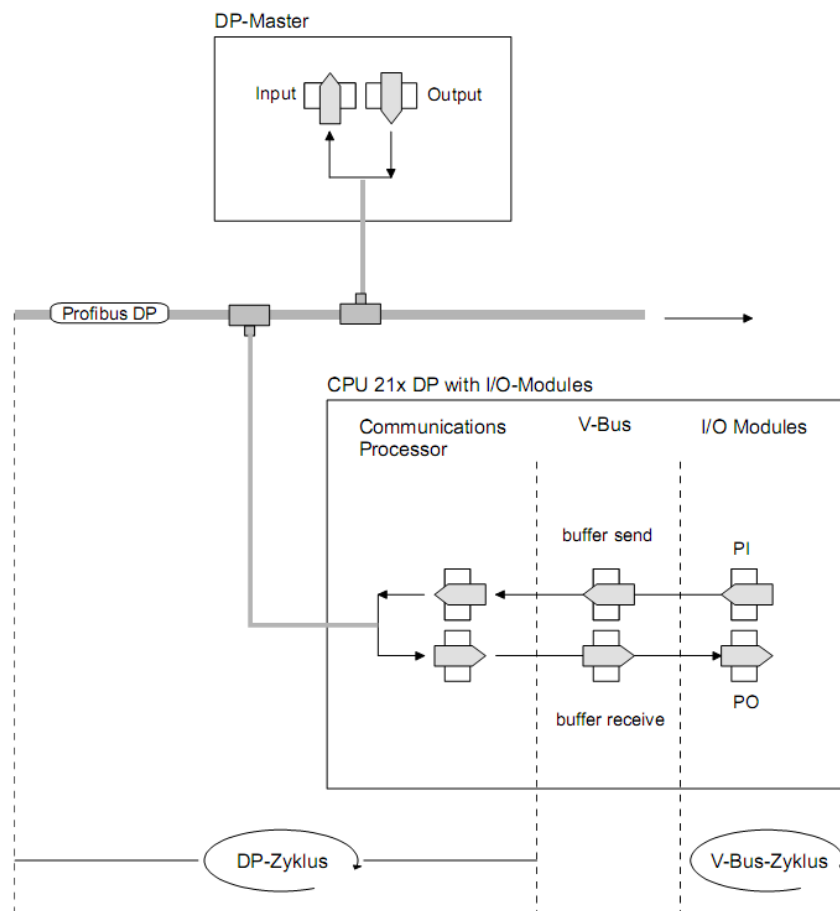
Організація обміну даними в системі майстер-слейв складається з параметризації, конфігурації та фази передачі даних. Перед тим, як DP-слейв переходить до передачі даних, майстер перевіряє під час виконання фази параметризації і конфігурації чи задані параметри збігаються з реально наявними апаратними засобами. Під час цього процесу перевіряється тип пристрою, формат і величина даних, кількість входів і виходів. Це забезпечує уникнення помилок в конфігурації.

Майстер самостійно керує мережевим обміном даних. Також Ви маєте можливість надіслати нові дані конфігурації розподільвача шини.

Якщо активовано стан DE „Data Exchange“, майстер передає нові дані в слейв, а назад отримує останні значення з входів.

Принцип передачі даних

Обмін даними між DP-майстром і DP-слейвом виконується циклічно з використанням буферу прийому-передачі.



PII: Образ процесу входів

PIQ: Образ процесу виходів

Цикл V-шини

В циклі V-шини (V-шина = внутрішня шина) усі вхідні дані з сигнальних модулів збираються в PI, а всі вихідні дані з PO передаються на вихідні модулі. Після завершення обміну даних PI передається в буфер відправки (буфер передачі), а вміст вхідного буфера (буфер прийому) передається в PO.

DP-цикл

В PROFIBUS-циклі майстер здійснює обмін даними з усіма своїми слейвами. При цьому області пам'яті, закріплені за PROFIBUS, переписуються і зчитуються.

Після цього DP-майстер передає дані вхідної області в буфер прийому комунікаційного процесора, а дані буфера передачі передаються в вихідну область PROFIBUS.

Обмін даними по шині між DP-майстром і слейвами здійснюється циклічно і не залежить від циклу V-шини.

Цикл V-шини / DP-цикл	<p>Для забезпечення одночасної передачі даних час циклу V-шини повинен завжди бути таким же або меншим, ніж час циклу DP.</p> <p>В GSD-файлі встановлено параметр min_slave_interval = 3ms.</p> <p>Таким чином, гарантується, що PROFIBUS-дані на V-шині оновлюються не швидше 3 мс, а обмін даними зі слейвами виконується швидше.</p>
Послідовність даних	<p>Якщо дані мають один логічний зміст, вони передаються безперервно. Дані, що передаються разом: старший і молодший байт аналогового значення (тип слово). Для доступу до регістрів необхідно вказувати контрольний та статусний байт з відповідним вордовським параметром. Цілісність даних при взаємодії між периферійними пристроями та контролером гарантується тільки для 1Byte. Тобто, біти одного байта викликаються та передаються разом. По-байтної послідовності достатньо для обробки цифрових сигналів.</p> <p>Якщо довжина даних перевищує один байт, наприклад, для аналогових значень, цілісність даних має бути розширена. PROFIBUS гарантує узгодженість необхідної довжини даних. Переконайтесь, що Ви використовуєте правильний метод читання послідовності даних PROFIBUS-майстра PLC.</p> <p>За додатковою інформацією, будь ласка, зверніться до технічної документації на PROFIBUS-майстер чи інтерфейсний модуль.</p>
Обмеження	<p>Якщо майстер переходить в помилку, цей стан не розпізнається процесором автоматично. В програмі завжди необхідно створювати контрольний байт, який буде вказувати на працездатність майстра і, тим самим, забезпечувати присутність в системі достовірних даних.</p> <p>В кінці цього розділу наведено приклад і пояснення використання контрольного байту.</p>
Обмеження	<p>Існує великий набір діагностичних функцій, які допомагають у діагностуванні та швидкій локалізації помилок в системі PROFIBUS-DP. Діагностичні дані передаються по шині усіма її учасниками і збираються у майстрі.</p>

**Середовище
передачі даних**

В якості середовища передачі PROFIBUS використовує два ізольовані скручені провідники, і базується на інтерфейсі RS485 чи дуплексному оптичному хвилеводі (OWG). Швидкість передачі даних для обох методів макс. 12Mbaud.

**Електрична
система на
інтерфейсі
RS485**

Інтерфейс RS485 заснований на передачі напруги різної величини. Мережа може мати як лінійну так і деревовидну структуру. Розподільувач VIPA PROFIBUS має 9-контактний роз'єм, через який контролер підключається в мережу PROFIBUS в якості слейва. Мережева структура на RS485 дозволяє легко додавати нові станції. Подальше розширення системи жодним чином не впливає на вже наявних учасників мережі. Система автоматично розпізнає вихід з ладу пристроїв мережі або додавання нових.

**Оптичні
системи на
волокні (FO)**

Оптична система хвилеводів використовує монохромні світлові імпульси. Оптичний хвилевід повністю незалежить від впливу напруги чи інших електричних перешкод. Така система має лінійну структуру. Кожен модуль повинен мати два зв'язки: один вхід і один вихід. На останньому сегменті мережі встановлення термінатора не потрібне. Оскільки система має лінійну структуру, підключення і відключення станцій має вплив на мережу.

Адресація

Кожен учасник мережі PROFIBUS повинен ідентифікувати себе певною адресою. В межах шини ця адреса повинна бути унікальною і може приймати значення від 0 до 125.

В CPU 21xDPM адреса задається через інструмент програмування.

GSD-файл

Для конфігурування підключення слейва, в інструмент програмування необхідно завантажити всю інформацію про модулі VIPA у формі електронного паспорта (GSD-файл).

Структура і зміст цього документа відповідає вимогам PROFIBUS User Organization (PNO) і може бути надана цією організацією.

Послідовність дій по завантаженню відповідного GSD-файлу знаходиться в технічній документації на інструмент програмування.

Конфігурування CPU 21хDP

Вступ

На відміну від VIPA PROFIBUS слейва IM 253DP, з'єднувач PROFIBUS в CPU 21хDP являється "інтелектуальним розподільувачем". Він здійснює обробку даних, які знаходяться у вхідній\вихідній області процесора. Ця область і область збереження статусу і діагностичних даних резервується за допомогою властивостей CPU 21хDP. Для вхідних і вихідних даних використовуються різні області пам'яті, керування якими здійснюється через програму PLC.

Завдяки налаштуванням системи, адреси областей, які зарезервовані за розподільувачем, не відображаються в апаратному конфігураторі Siemens. Для напряду підключених модулів System 200V, адреси яких також включені в діапазон доступних адрес, при автоматичному розподілі може виникнути їх перекриття.



Примітка!

Для конфігурування CPU та PROFIBUS-DP майстра, необхідне досконале знання SIMATIC Manager і апаратного конфігуратора від Siemens!

Конфігурування в SIMATIC Manager від Siemens

Розподіл адрес та параметризація підключених модулів відбувається в Siemens SIMATIC Manager через створення віртуальної PROFIBUS системи.

Після включення GSD-файлу в SIMATIC Manager користувачу стає доступна повна функціональність модулів VIPA.

Кроки по проектуванню

Для сумісності System 200V з Siemens SIMATIC, необхідно виконати наступні кроки:

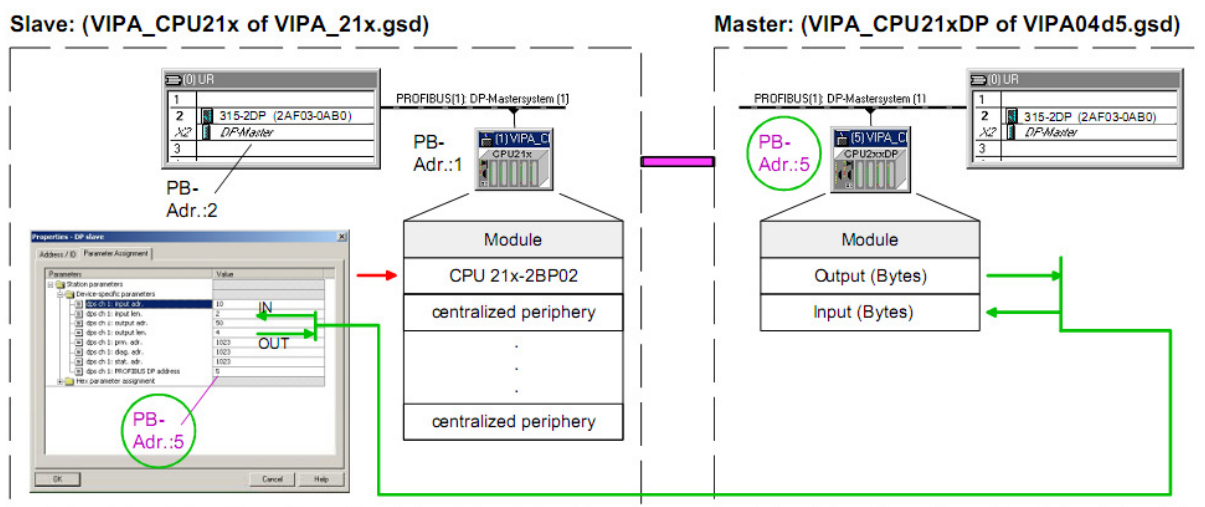
- Створіть проект CPU 315-2DP з системою DP-майстра (адреса 2)
- Додайте PROFIBUS **VIPA_CPU21x** слейв з VIPA_21x.gsd з адресою 1
- Вставте **CPU 21хDP** на перший слот слейв-системи
- Налаштуйте PROFIBUS-параметри CPU 21хDP
- Додайте в систему підключені периферійні модулі
- Передайте проект в CPU 21хDP через MPI.

Кроки по конфігуруванню майстра

- Створіть проект CPU з DP майстром з адресою 2
- Додайте PROFIBUS-слейва "VIPA_CPU2ххDP" з VIPA04D5.gsd
- Починаючи з 1-го слоту встановіть вхідні і вихідні діапазони PROFIBUS. Будь ласка, майте на увазі, що налаштування довжин даних в слейві повинні відповідати Byte-налаштуванням в майстрі!

Зв'язок між майстром і слейвом

На наступному малюнку показаний етап розробки проекту майстер-слейвової системи:



Розробка проекту для CPU 21xDP

CPU 21xDP конфігурується через Siemens SIMATIC Manager.

Передумови

Перед створенням проекту для System 200V чи System 300 повинні бути виконані наступні умови:

- Встановлено Siemens SIMATIC Manager
- В апаратний конфігуратор необхідно включити відповідні GSD-файли
- Необхідно реалізувати з'єднання між PG та CPU.

Установка апаратного конфігуратора Siemens

Апаратний конфігуратор являється частиною Siemens SIMATIC Manager, і використовується для створення проектів. Усі модулі, які можуть бути використані в проекті, перераховані в каталозі устаткування.

Для створення проекту PROFIBUS-DP слейва від VIPA необхідно включити відповідні модулі в каталог устаткування SIMATIC за допомогою GSD-файлу.

Вставка GSD-файлу

Перед виконанням процедури вставки необхідно розархівувати архів, який містить відповідний GSD-файл.

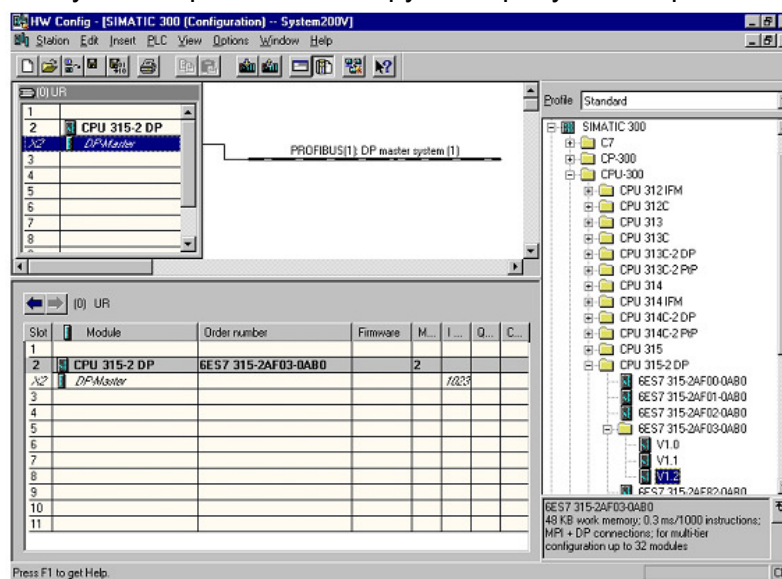
- Запустіть апаратний конфігуратор Siemens. Для включення нових GSD-файлів в системі не повинно бути відкрито жодного проекту
- В головному меню **Options** > *Install new GSD-file* відкрийте діалогове вікно вставки
- Виберіть потрібний GSD-файл і через [Open] завантажте його.

Тепер в розділі *PROFIBUS-DP* > *Additional Field devices* > *I/O* > *VIPA* каталогу устаткування Ви зможете знайти необхідні модулі VIPA.

Створення віртуальної PROFIBUS-системи

- Створіть новий проект System 300V і додайте профільну рейку з каталогу обладнання
- В каталозі CPU з PROFIBUS майстром виберіть: Simatic300 > CPU-300 > CPU315-2DP > **6ES7 315-2AF03-0AB0**
- Вставте CPU 315-2DP (**6ES7 315-2AF03-0AB0 V1.2**)
- Введіть для майстра адресу PROFIBUS = 2
- В *Object properties* натисніть DP і виберіть режим "DP-майстер". Підтвердіть вибір через OK
- Правою кнопкою миші натисніть на кнопку "DP" і в контекстному меню виберіть пункт "Add master system". Створіть нову підмережу PROFIBUS.

Наступне зображення ілюструє створену майстер-систему:



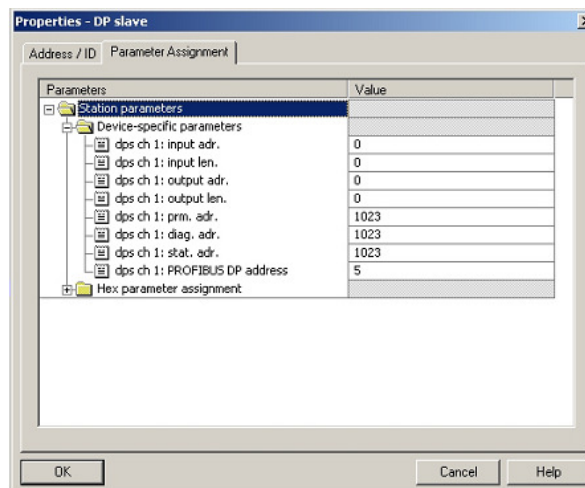
Конфігурування секції CPU

Щоб мати сумісність з Siemens SIMATIC, як уже згадувалося вище, в проект необхідно включити CPU в явному вигляді.

- Додайте System "VIPA_CPU21x" в підмережу PROFIBUS. Модуль Ви знайдете в каталозі устаткування *PROFIBUS DP > Additional field devices > IO > VIPA_System_200V*. Призначте слейву адресу PROFIBUS = 1
- В конфігураторі помістіть CPU 21x-2BM02 від VIPA у перший слот. Він знаходиться в каталозі устаткування в директорії *VIPA-CPU21x*
- Змінити параметри зарезервованих областей даних PROFIBUS Ви можете у вікні налаштувань CPU. Більш детальна інформація знаходиться в "Додавання секції PROFIBUS"
- Додайте модулі System 200V в порядку установки
- Якщо існує DP-майстер, його також необхідно розмістити у відповідне місце
- Збережіть Ваш проект і передайте його через MPI в CPU 21xDP.

Додавання PROFIBUS-секції

В секції PROFIBUS створюється образ його області даних в просторі адрес CPU 21xDP. Призначення областей відбувається у властивостях CPU 21xDP. Подвійним натисненням на CPU 21xDP викликається діалогове вікно параметризації областей даних PROFIBUS-слейва. Подробиці можна знайти в розділі "Параметри DP-слейва".

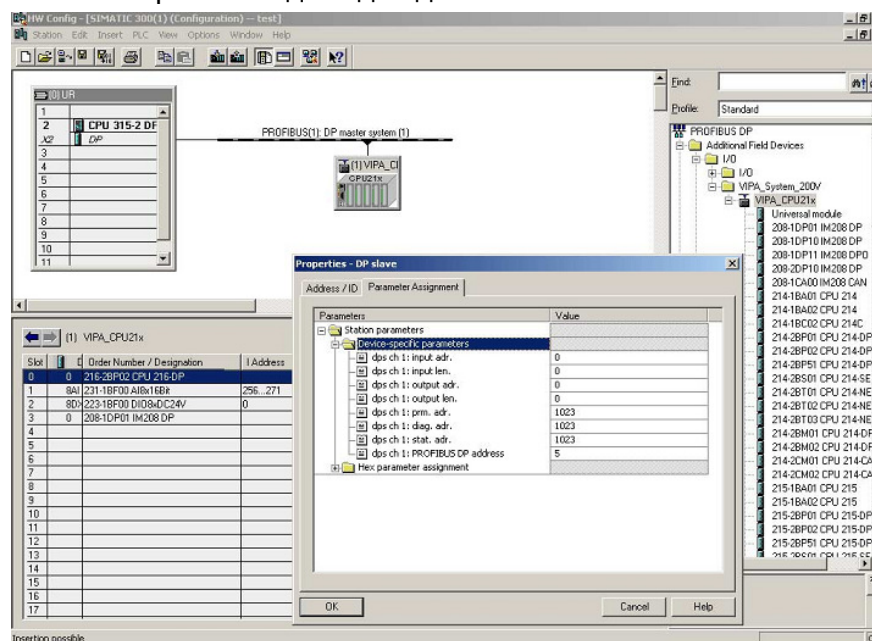


Увага!

Будь ласка, майте на увазі, що довжини областей даних в слейві і майстрі повинні бути ідентичними.

Через обмеження, що накладаються системою, області пам'яті CPU, які використовуються для секції PROFIBUS, можуть відображатись тільки у вікні конфігурування процесора.

Нижче зображено відповідне діалогове вікно:

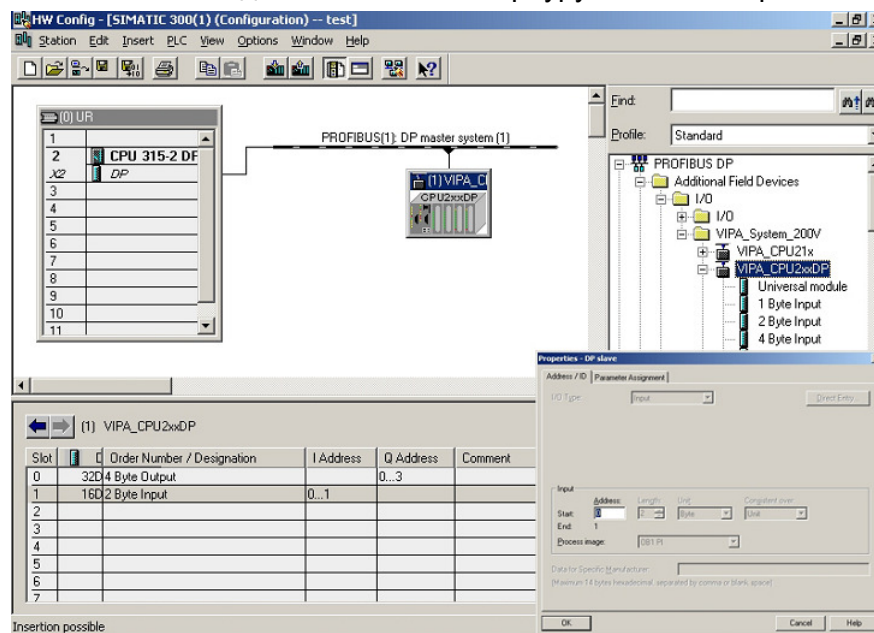


Підключення до DP-майстра

Щоб включити CPU 21xDP в майстер-систему в конфігуратор необхідно завантажити GSD-файл `vipa04d5.gsd`.

- Запустіть програму конфігурації і зконфігуруйте PROFIBUS-DP майстра, який буде ведучим CPU 21xDP
- Додайте DP-слейв станцію типу "CPU2xxDP". Її можна знайти в каталозі устаткування *PROFIBUS-DP > Additional field devices > I/O > VIPA > VIPA_CPU2xxDP*
- Призначте необхідну PROFIBUS адресу DP-слейву
- Для організації PROFIBUS-зв'язку, створіть такий же I/O діапазон, який Ви задали для слейва у розділі "modules". Будь ласка, зверніть увагу, що вихідна область слейва відноситься до вхідної області майстра, і навпаки
- Збережіть Ваш проект і передайте його в CPU майстра.

Нижче показане діалогове вікно конфігурування майстра:



Примітка!

Якщо Ваша DP майстер-система є модулями System 200V від VIPA, Ви можете параметризувати підключені модулі шляхом включення "DP200V" слейв-системи.

Для того щоб процесор VIPA розпізнав проект як центральну систему, слейву необхідно призначити PROFIBUS-адресу = 1!

Будь ласка, будьте уважні при використанні IM 208 PROFIBUS-DP майстра версії V 3.0 або вище. Версія його прошивки повинна співпадати з версією CPU. Номер версії можна знайти на етикетці на боковій стороні модуля.

Параметри DP-слейва

Короткий огляд

Секція PROFIBUS записує свої області даних в області пам'яті CPU 21xDP.

Вхідні або вихідні області повинні опитуватися в програмі PLC.

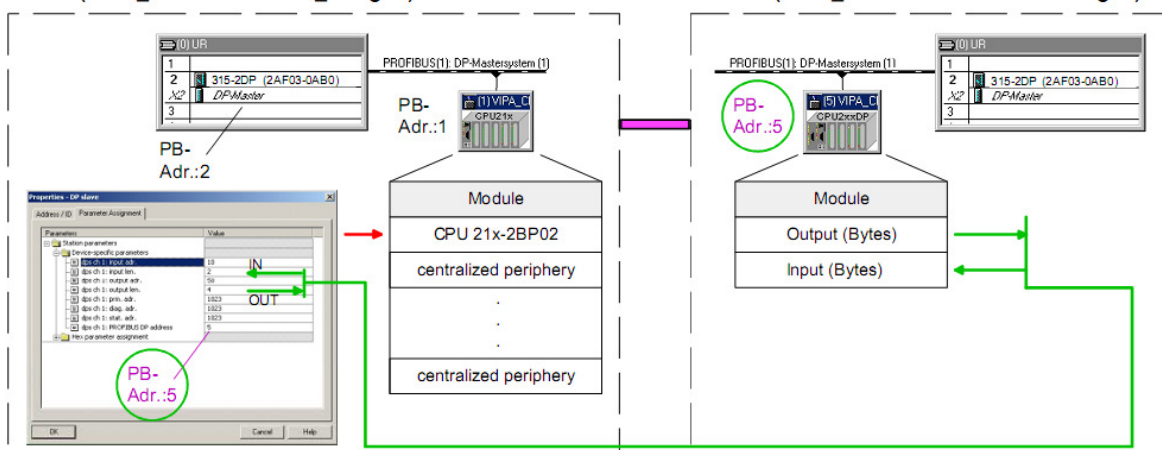


Увага!

Довжини даних у вхідній і вихідній області в слейві повинні відповідати байтовим значенням в конфігурації майстра. В противному випадку PROFIBUS-комунікація не буде можлива, а майстер просигналізує про збій слейва.

Slave: (VIPA_CPU21x of VIPA_21x.gsd)

Master: (VIPA_CPU21xDP of VIPA04d5.gsd)



Використання областей CPU

Як тільки Ви встановите значення довжини в 0, мережеві дані не будуть займати пам'ять в CPU.

Встановивши 255 (обмеження по пам'яті) в параметрах PRN, DIAG і STAT, Ви також можете звільнити зайняту в CPU пам'ять.



Примітка!

До версії прошивки процесора V 2.2.0 CPU 21x система PROFIBUS-DP підтримувала діапазон адрес від 0 до 255.

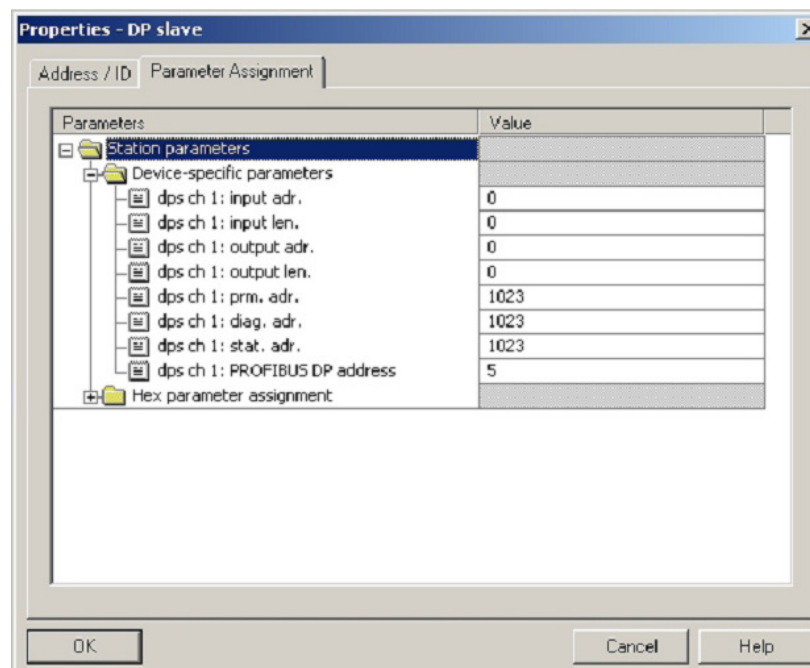
Починаючи з версії прошивки V 3.0 став підтримуватися діапазон адрес від 0 до 1023.

Версія прошивки зазначена на етикетці на боковій стороні модуля.

Значення 1023 використовується для відключення PRN, DIAG і STAT.

Опис параметризаційних даних

Діалогове вікно налаштування PROFIBUS-слейва викликається подвійним кліком на піктограмі CPU 21xDP в конфігураторі обладнання Siemens:



input adr, len

Адреси, звідки надходять дані по PROFIBUS. Повинні бути збережені в CPU разом з їхніми довжинами.

Довжина, яка має значення 0 не займає місце пам'яті в області вводу CPU.

output adr, len

Адреси, в яких зберігаються дані які відправляються через PROFIBUS. Тут також визначаються довжини даних.

Довжина, яка має значення 0 не займає місце пам'яті в області виводу CPU.

prm. adr.

Параметричні дані являються випискою з параметричної телеграми.

Параметрична телеграма створюється під час конфігурування майстра і відсилається до слейва, коли:

- CPU 21xDP знаходиться в стані запуску;
- зв'язок між CPU 21xDP і майстер було перервано.

Параметрична телеграма містить специфічні дані PROFIBUS (параметри шини) і дані користувача, а також вхідні і вихідні байти CPU 21xDP.

Користувацькі дані (Byte 7...31), розміщені в області пам'яті CPU, заносяться в *prm*.

За допомогою цього параметру можна перевірити чи отримав слейв налаштування від майстра.

prm. adr.

Діагностичні функції PROFIBUS-DP дозволяють швидко виявити помилки в мережі та локалізувати їх. Діагностичні повідомлення передаються через шину і збираються в майстрі.

CPU 21xDP пересилає діагностичні дані за запитом майстра або у випадку появи помилки. Діагностичні дані складаються з:

- Звичайні діагностичні дані (Byte 0... 5)
- Діагностичні дані пов'язані з пристроєм (Byte 6...10)
- **Діагностичні дані користувача (Byte 11...15)**

Через *diag* Ви визначаєте початкову адресу, з якої 6 Byte діагностичних даних користувача зберігаються в CPU.

Ви можете ініціювати і впливати на діагностику через контрольований доступ до цієї області.

**Примітка!**

Детальнішу інформацію про структуру та можливості керування через діагностичні повідомлення можна знайти в розділі "Діагностичні функції".

stat. adr.

Останній стан PROFIBUS-зв'язку можна зчитати з двох-байтової області стану в адресній області периферії CPU, починаючи з адреси статусу.

**Примітка!**

Детальнішу інформацію про структуру статусного повідомлення можна знайти в розділі "Внутрішні статусні повідомлення CPU".

**PROFIBUS DP
адреси**

За допомогою цього параметру Ви можете призначити PROFIBUS адресу в PROFIBUS системі.

**Звільнення
зон CPU**

Як тільки Ви виставите довжину в 0, відповідні дані не будуть займати пам'яті в CPU.

Ви також можете звільнити простір пам'яті CPU шляхом присвоєння параметрам *prn*, *diag* і *stat* максимальних адрес (255 чи 1023 для CPU версії прошивки > 2.2.0).

Діагностичні функції

Короткий огляд

PROFIBUS-DP оснащений великим набором діагностичних функцій які можуть бути використані для швидкого та ефективного виявлення проблем. Діагностичні повідомлення передаються по шині і збираються в майстрі.

CPU 21xDP передає діагностичні дані на вимогу майстра або при появі помилки. Оскільки частина діагностичних даних (Byte 11...15) знаходиться в периферійних адресах CPU, Ви можете ініціювати діагностику і змінити діагностичні дані. Діагностичні дані складаються з:

- стандартних діагностичних даних (Byte 0...5),
- діагностичних даних обладнання (Byte 6...15).

Структура

Структура діагностичних даних виглядає наступним чином:

Стандартні діагностичні дані

Byte 0	Статус станції 1
Byte 1	Статус станції 2
Byte 2	Статус станції 3
Byte 3	Адреса майстра
Byte 4	Ідентифікаційний номер (молодший)
Byte 5	Ідентифікаційний номер (старший)

Діагностичні дані обладнання

Byte 6	Довжина та код діагностики
Byte 7	Діагностичні повідомлення обладнання зарезервовано
Byte 8 ... Byte 10	Діагностичні дані користувача, які розподілені по периферійним адресам CPU і можуть модифікуватися та пересилатися майстру.
Byte 11 ... Byte 15	

**Стандартні
діагностичні
дані**

Докладну інформацію про структуру стандартних діагностичних даних можна отримати з літератури на PROFIBUS.

Стандартні діагностичні дані слейвів мають наступну структуру:

Byte	Bit 7 ... Bit 0
0	Bit 0: зафіксовано в 0 Bit 1: слейв не готовий до передачі даних Bit 2: конфігураційні дані не ідентичні Bit 3: слейв отримав зовнішні діагностичні дані Bit 4: слейв не підтримує цієї функції Bit 5: зафіксовано в 0 Bit 6: невірна параметризація Bit 7: зафіксовано в 0
1	Bit 0: слейв потребує нової параметризації Bit 1: статистична діагностика Bit 2: зафіксовано в 1 Bit 3: активація контролю відповіді Bit 4: отримана команда заморожки Bit 5: отримана команда синхронізації Bit 6: зарезервовано Bit 7: зафіксовано в 0
2	Bit 0...Bit 6: зарезервовано Bit 7: переповнення діагностичних даних
3	Адресація майстра після параметризації FFh: слейв без параметризації
4	Ідентифікаційний номер старшого байту
5	Ідентифікаційний номер молодшого байту

Діагностичні дані обладнання

Діагностичні дані обладнання містять докладну інформацію про слейвів і периферійні модулі. Довжина діагностичних даних обладнання фіксована, і складає 10Byte.

Byte	Bit 7 ... Bit 0
6	Bit 0...5: довжина діагностичних даних обладнання 001010: довжина 10Byte (фіксована) Bit 6...7: коди діагностики 00: код 00 (фіксовано)
7	Bit 0...Bit 7: діагностичні повідомлення 12h: Помилка: параметри довжини даних 13h: Помилка: конфігураційні дані довжини даних 14h: Помилка: конфігураційний запис 15h: Помилка: VPC3 обчислення буферу 16h: Помилка: втрата конфігураційних даних 17h: Помилка: відмінність параметризації DP та конфігурації 40h: користувачка діагностика коректна
8...10	зарезервовано
11...15	Діагностичні дані користувача, які зберігаються в діагностичному статусному байті в фоновому режимі образу процесу CPU. Ці дані можуть бути перезаписані і переслані до майстра.

Виконання діагностики

При діагностичному запиті, зміст статусних даних обладнання з Byte 11...15 передається в образ процесу CPU і статусний байт стає активним.

Місце розташування цього діагностичного блоку з довжиною 6Byte визначається через параметри CPU.

Ініціація діагностики відбувається за допомогою зміни статусу 0 → 1 в діагностичному байті статусу. При цьому усі діагностичні повідомлення передаються майстру. **Статус 0000 0011 ігнорується!**

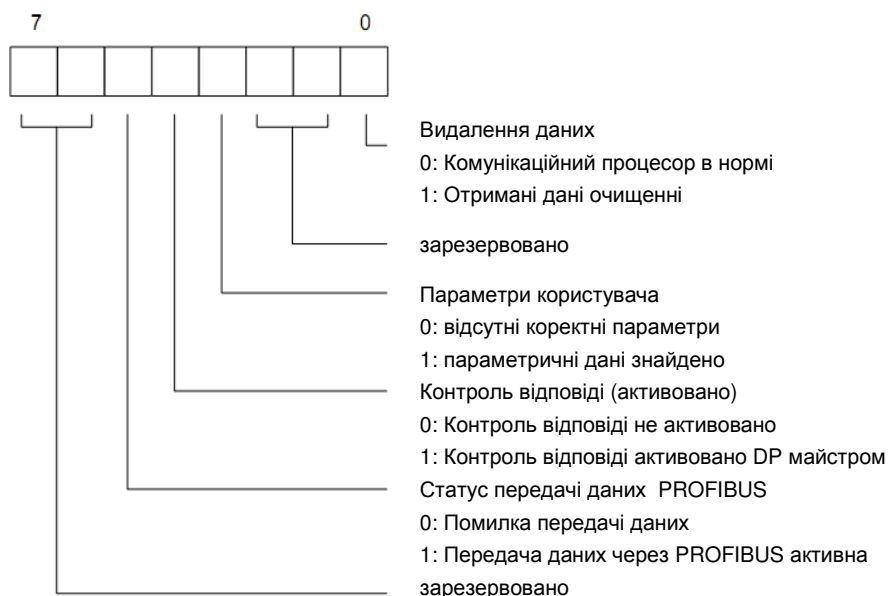
Діагностичний блок CPU має наступну структуру:

Byte	Bit 7 ... Bit 0
0	статусний байт діагностики: Bit 0: діагностичні дані користувача 0: не дійсні діагностичні дані 1: дійсні діагностичні дані (початок діагностики) Bit 1: видалення діагностики 0: помилка видалення діагностики 1: успішне видалення діагностики Bit 2...Bit 7: зарезервовано
1...5	Bit 0...Bit 7: діагностичні дані користувача, аналогічні до Byte 11...15 діагностичних даних обладнання

Внутрішні статусні повідомлення CPU

Поточний стан комунікаційної процедури PROFIBUS приймає значення від статусних повідомлень, які перерозподіляються до периферійних адрес CPU. Статусні повідомлення складаються з 2Byte і мають наступну структуру:

Статусний байт Byte 0



Статусний байт Byte 1



Параметри

Видалення даних	Буфери прийому\передачі очищуються при виникненні помилки.
зарезервовано	Ці біти зарезервовані для подальшого використання
Параметри користувача	Вказує на коректність параметричних даних. Параметричні дані задаються через конфігураційний інструмент майстра.
Контроль відповіді (активовано)	Вказує на стан активації контролю відповіді в PROFIBUS майстрі. Слейв припиняє зв'язок, коли контрольний час вийшов.
Статус передачі даних PROFIBUS	Індикація стану зв'язку з майстром. При невірній конфігурація чи некоректних параметрах зв'язок припиняється і активується біт помилки.
Параметризація	Відображує стан конфігураційних даних. Аналізується довжина параметризаційних даних конфігурації і кількість зайнятих байт. Обмін даними відбудеться, якщо вони коректні, і якщо займають не більше 31Byte.
Конфігурація	Індикація стану даних конфігурації, які були отримані від PROFIBUS майстра. Конфігурація визначається за допомогою інструменту конфігурування майстра.
Контроль відповіді (Watchdog)	Вказує на стан активації контролю відповіді в PROFIBUS майстрі. Якщо активовано контроль відповіді і перевищено відведений слейву час на відповідь, в цей біт заноситься помилка.
Апаратний контроль	Цей біт встановлюється, якщо PROFIBUS контролер процесора 21xDP несправний. У цьому випадку Вам необхідно звернутися в службу технічної підтримки VIPA.
Дані DP	Цей біт активує будь-яка помилка передачі в PROFIBUS.

Інструкція по створенню мережі PROFIBUS

Загальні відомості про PROFIBUS

- Мережа PROFIBUS DP може мати лише лінійну структуру.
- PROFIBUS DP складається з мінімум одного сегменту, який містить один майстер і один слейв.
- Майстером завжди виступає контролер з CPU.
- PROFIBUS підтримує макс. до 126 учасників.
- На сегмент допускається до макс. до 32 учасників.
- Макс. довжина сегмента залежить від швидкості передачі даних:

9.6 ... 187.5kbaud	→ 1000m
500kbaud	→ 400m
1.5Mbaud	→ 200m
3 ... 12Mbaud	→ 100m
- Можна організувати макс. 10 сегментів. Сегменти підключаються через ретранслятори. Кожен ретранслятор враховується як один учасник.
- З обох кінців шина має бути оснащена термінаторами.
- Обмін даними між усіма учасниками відбувається з однією швидкістю. Слейви автоматично налаштовуються на визначену майстром швидкість передачі даних.



Примітка!

При використанні оптичних пристроїв, для уникнення пошкодження очей і зменшення розсіювання променя вільний комунікаційний роз'єм на останній станції необхідно закрити заглушкою.

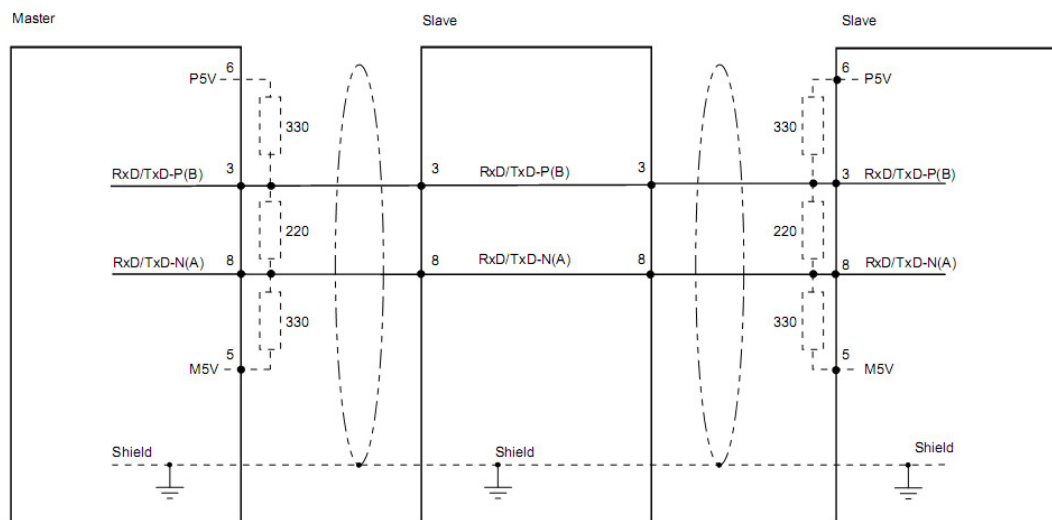
Середовище передачі даних

В якості середовища передачі PROFIBUS використовує ізольовану виту пару і базується на інтерфейсі RS485. Інтерфейс RS485 працює по технології контролю диференціальної напруги. Це забезпечує меншу чутливість до перешкод, ніж сигнали по струму чи фіксованій напрузі. Ви можете сконфігурувати мережу як за лінійною, так і за деревовидною структурою. Дopuskaється максимум до 32 учасників на кожен сегмент. В сегменті учасники з'єднуються за лінійною структурою. Сегменти підключаються через спеціальні конектори - ретранслятори. Максимальна довжина сегмента залежить від швидкості передачі даних. PROFIBUS DP використовує швидкість передачі даних в діапазоні між 9.6kbaud і 12Mbaud, слейви підстроюються автоматично. Усі учасники обмінюються даними з однією швидкістю.

Мережева структура, побудована на RS485, дозволяє здійснювати легке підключення\відключення станцій, а також покроковий запуск системи. Подальше розширення системи не впливає на вже встановлені станції. Система автоматично діагностує вихід з ладу учасника мережі чи появу нового.

Мережеве з'єднання

На наступному малюнку показане підключення термінальних резисторів на стартовій і кінцевій станції.



Примітка!

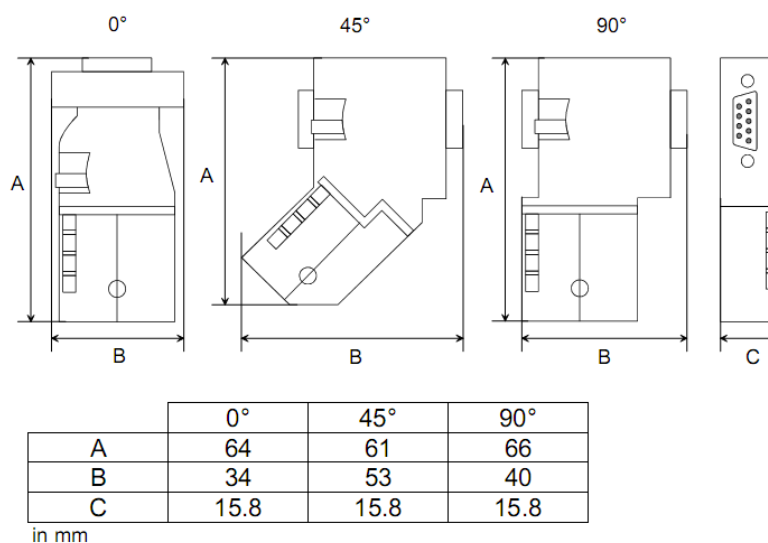
PROFIBUS кабель повинен закінчуватися термінальними резисторами на номінал, рівний його хвильовому опору. Будь ласка, переконайтеся, у дотриманні цих вимог.

Мережеві конектори EasyConn



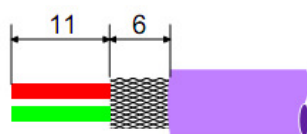
У PROFIBUS всі учасники підключаються паралельно. З цією метою кабель повинен бути прохідним.

Підключення учасників мережі відбувається за допомогою мережевих конекторів "EasyConn" VIPA 972-0DP10. Ці пристрої мають вбудований термінальний перемикач та світлодіодну LED-індикацію.



**Примітка!**

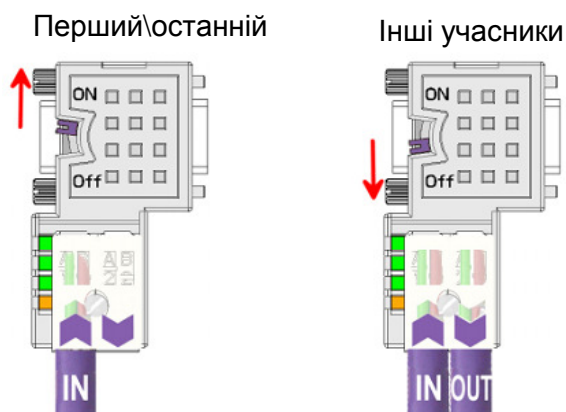
Для організації мережі PROFIBUS на базі конекторів EasyConn використовуйте стандартний кабель PROFIBUS тип A (EN50170). Починаючи з версії 5 також можна використовувати гнучкий мережевий кабель Lapp Kabel (код для замовлення: 2170222, 2170822, 2170322). Також можна замовити спеціальний інструмент для зняття ізоляції "EasyStrip" VIPA 905-6AA00.



Dimensions in mm

**Термінація
"EasyConn"**

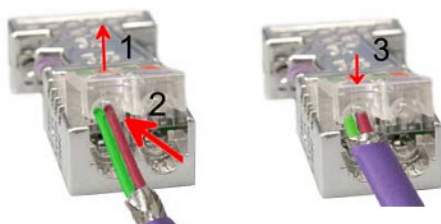
Конектор "EasyConn" обладнано перемикачем, за допомогою якого підключається термінальний резистор.

Підключення**Увага!**

Термінальний резистор має значення тільки якщо конектор підключений до учасника мережі і він підключений до живлення.

Примітка!

Повна інструкція по підключенню термінальних резисторів постачається разом з конектором.

Монтаж

- Послабьте гвинт
- Підніміть контакту кришку
- Вставте обидва дроти в канали (слідкуйте за дотриманням полярності)
- Будьте уважні, щоб не замкнути лінії даних і екран
- Закрийте кришку контактів
- Затягніть гвинт (макс. момент затягування 4Nm).

Зауваження!

Зелений провідник необхідно підключати до контакту A, червоний – до B.

Введення в експлуатацію

Короткий огляд

- Встановіть CPU 21xDP
- Зконфігуруйте CPU 21xDP в майстер-системі
- Зконфігуруйте периферійні модулі I/O, що підключені до внутрішньої шини
- Підключіть CPU 21xDP до мережі PROFIBUS
- Увімкніть живлення
- Завантажте проект в CPU.

Установка

Підключіть до CPU 21xDP усі необхідні периферійні модулі. Контролюйте, щоб максимальний споживаний струм не перевищив потужність джерела живлення.



Примітка!

Для уникнення проблем зі зв'язком через віддзеркалювання сигналу, мережевий кабель необхідно закінчувати термінальним резистором.

Конфігурування в майстер-системі

Зконфігуруйте CPU 21xDP в майстер-системі. З цією метою можна використовувати програмний пакет VIPA WinNCS. Для конфігурування відповідних модулів System 200V PROFIBUS слід необхідно завантажити їх GSD-файли.

Конфігурування CPU 21xDP та периферійних I/O

Периферійні модулі System 200V, зв'язані з CPU 21xDP через внутрішню шину, автоматично отримують адреси області пам'яті CPU. В будь-який час через апаратний конфігуратор від Siemens Ви можете змінити розподіл адрес.

Живлення системи

CPU 21xDP має вбудоване джерело живлення, яке необхідно підключити до DC 24V.

Джерело живлення забезпечує відповідною напругою не тільки процесор, а й внутрішню шину, через яку живляться периферійні модулі. Майте на увазі, що вбудований блок живлення забезпечує внутрішню шину макс. 3A.

PROFIBUS і внутрішня шина гальванічно ізольовані один від одного.

Завантаження проекту

Завантаження апаратної конфігурації в CPU здійснюється через порт MPI.

- Підключіть PG чи ПК через MPI до CPU.

Якщо Ваш програматор не має порту MPI, Ви можете реалізувати послідовне з'єднання типу точка-точка, на базі VIPA Green Cable.

Green Cable VIPA 950-0KB00 використовується тільки в VIPA CPU з інтерфейсом MP²I.

- Зконфігуруйте MPI інтерфейс вашого ПК.
- Через меню **PLC** > *Load to module* інструменту програмування, завантажте проект в CPU.
- Для додаткової безпеки зкопіюйте проект на MMC. Для цього вставте MMC в контролер і через меню **PLC** > *Copy RAM to ROM* передайте програму на MMC.

Під час процедури запису MC-LED індикатор на CPU буде блимати. По завершенню цієї операції індикатор погасне.

**Увага!**

Докладна інформація стосовно Green Cable і MP²I знаходиться в розділі "Конфігурування CPU 21xDP".

Етап ініціації

Після підключення живлення мережевий з'єднувач PROFIBUS виконує процедуру самотестування. В цей час виконуються деякі діагностичні операції, перевірка внутрішньої шини та PROFIBUS комунікації.

Якщо тест завершено успішно, зчитуються параметри CPU і перевіряються параметри PROFIBUS слейва.

Після завершення діагностики, статус шинного розподільювача переводиться в "READY".

Якщо шинний з'єднувач виявляє комунікаційні помилки у внутрішній шині, його статус переходить в STOP і після паузи в 2 секунди повторно ініціює процедуру діагностування. При успішному завершенні тесту, починає блимати RD-LED індикатор.

Якщо встановлюється зв'язок DE-LED індикатор починає світитись.

Приклад

Завдання

Наведений приклад демонструє організацію зв'язку між майстром CPU 214DPM і слейвом CPU 214DP.

Апаратні лічильники процесорного модуля обмінюються даними через PROFIBUS і відображаються на вихідних каналах модуля відповідного партнера.

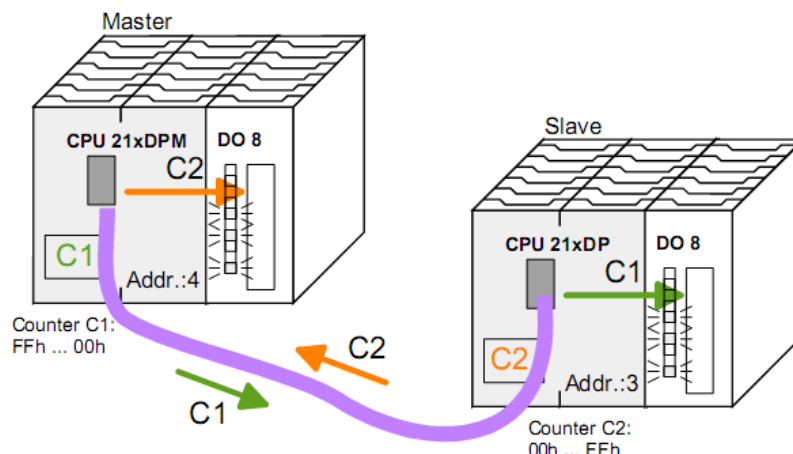
Детальний опис завдання

CPU 214DPM рахує від FFh до 00h і циклічно передає це значення у вихідну область PROFIBUS майстра. Після цього майстер має відправляти це значення до слейва CPU 214DP.

Отримане значення повинне зберігатися в зоні периферійних входів процесора і через внутрішню шину контролюватися на виході модуля (адреса 0).

З іншого боку, CPU 214DP рахує від 00h до FFh і зберігає це значення в вихідній області CPU слейва. Після цього воно передається до майстра через PROFIBUS.

Це значення повинне контролюватися на виході модуля (адреса 0) CPU 214DPM.



Конфігураційні дані

CPU 21xDPM

Лічильник: MB 0 (FFh...00h)

Адреса PROFIBUS: 4

Вхідна область: адреса 10 довжина: 2 Byte

Вихідна область: адреса 20 довжина: 2 Byte

CPU 21xDP

Лічильник: MB 0 (00h...FFh)

Адреса PROFIBUS: 3

Вхідна область: адреса 30 довжина: 2 Byte

Вихідна область: адреса 40 довжина: 2 Byte

Параметричні дані: адреса 50 довжина: 24 Byte (фіксовано)

Діагностичні дані: адреса 60 довжина: 6 Byte (фіксовано)

Статусні дані: адреса 100 довжина: 2 Byte (фіксовано)

Конфігурування CPU 21xDPM

Для сумісності System 200V з Siemens SIMATIC Manager необхідно виконати наступні кроки:

- Запустіть апаратний конфігуратор Siemens
- Встановіть GSD-файл VIPA_21x.gsd
- Зконфігуруйте CPU 315-2DP з DP майстер-системою (адреса 4)
- Додайте PROFIBUS слейва "VIPA_CPU21x" з адресою 1
- У 1-й слот слейв-системи розмістіть **CPU 214-2BM02**
- Розмістіть модуль виводу 222-1BF00.

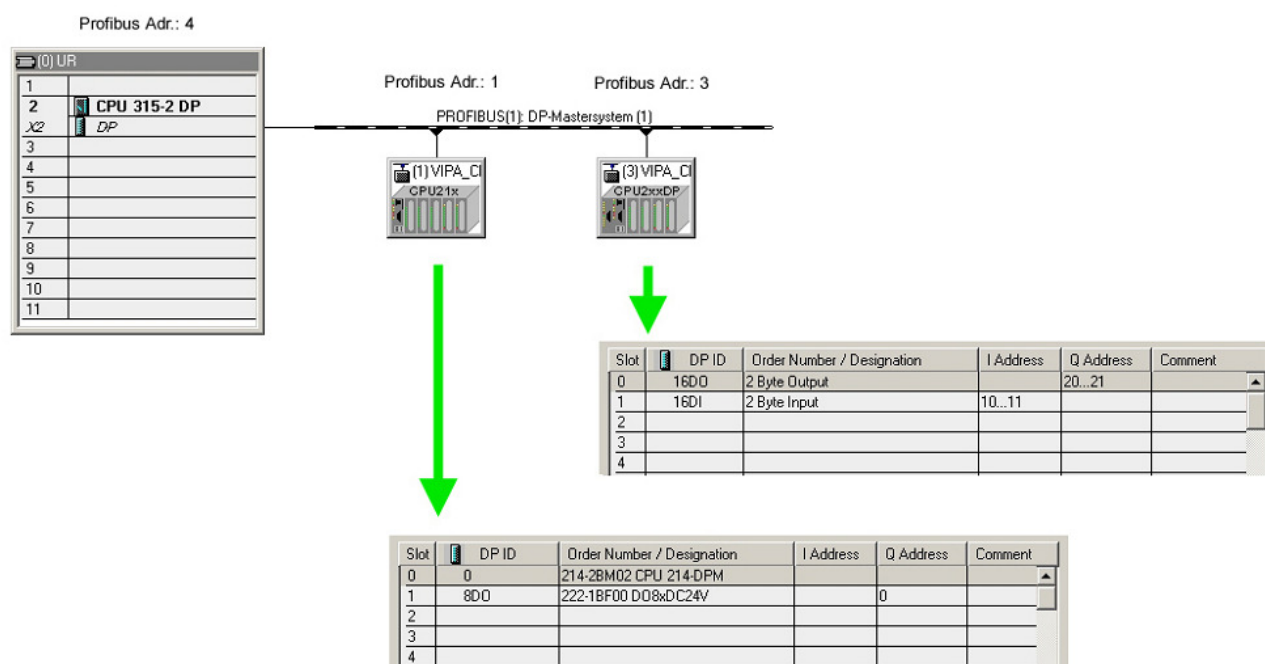
Для зв'язку з CPU 21xDP після додавання в конфігуратор GSD-файлу (VIPA04d5.gsd) необхідно виконати наступні кроки:

- Додайте PROFIBUS слейва "VIPA_CPU2xxDP" (адреса 3). DP слейв знаходиться в каталозі обладнання:

PROFIBUS-DP > Additional field devices > I/O > VIPA_System_200V > VIPA_CPU2xxDP

- Призначте область пам'яті CPU до вхідної і вихідної PROFIBUS-DP майстер-секції у вигляді байт блоків. Для цього необхідно на 1-му слоті розмістити елемент "2-х байтовий вихід" і вибрати вихідну адресу 20. В наступному слоті необхідно розмістити елемент "2-х байтовий вхід" і вибрати вхідну адресу 10

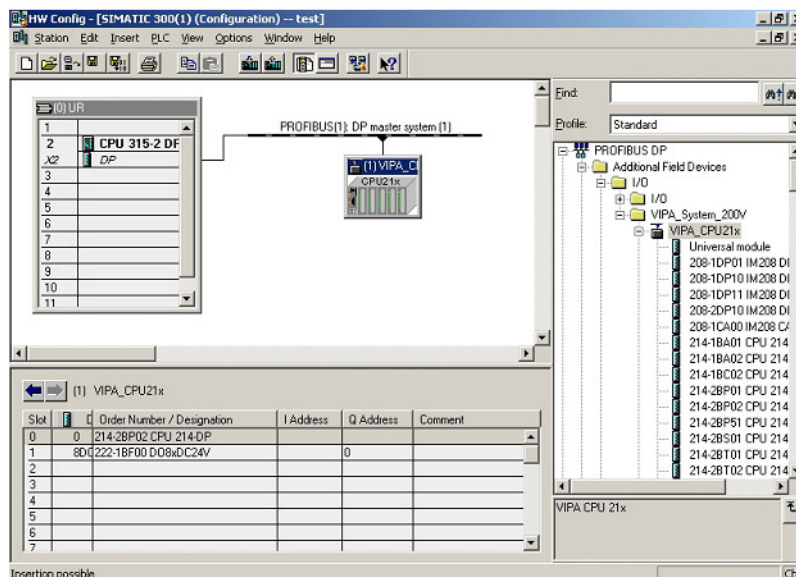
- Збережіть проект.



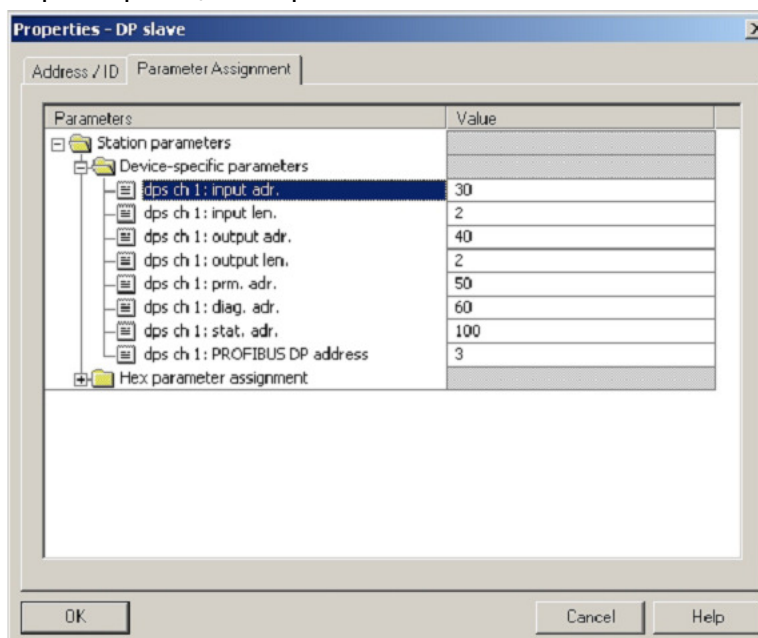
Конфігурування CPU 21xDP

Для сумісності System 200V з Siemens SIMATIC Manager необхідно виконати наступні кроки:

- Запустіть апаратний конфігуратор Siemens
- Зконфігуруйте CPU 315-2DP з DP майстер-системою (адреса 2)
- Додайте PROFIBUS слейва "VIPA_CPU21x" з адресою 1
- У 1-й слот слейв-системи розмістіть **CPU 214-2BM02**
- В наступний слот розмістіть модуль виводу 222-1BF00.



- У вікні параметризації виберіть CPU 214-2BP02:



- Збережіть проект.

**Прикладна
програма в CPU
21xDPM**

Прикладна програма в CPU 214DPM повинна виконувати два завдання, які будуть оброблятися в двох OB:

- перевірка зв'язку за допомогою контрольного байта.

Завантаження вхідного байту від PROFIBUS і передача його значення до модуля виходів.

OB 1 (циклічний виклик)

```

L      B#16#FF
T      AB  20          Control byte for Slave-CPU

L      B#16#FE          Load control value 0xFE
L      EB  10          Was the control byte transferred
<>I                    correctly from the slave CPU?
BEB                    No -> End

-----
Data exchange via PROFIBUS

L      EB  11          Load input byte 11 (output data
T      AB  0          of the CPU214DP) and
                      transfer to output byte 0
BE
```

- зчитування значення лічильника з MB 0, його зменшення, збереження в MB0 і передача до CPU 214DP через PROFIBUS.

OB 35 (таймер-OB)

```

L      MB  0          Counter from 0xFF to 0x00
L      1
-I
T      MB  0

T      AB  21          Transfer into output byte 21
                      (input data of CPU214DP)
BE
```

На цьому програмування CPU 214DPM завершено. Між обома станціями повинний встановитися PROFIBUS-зв'язок.

Через MPI за допомогою PLC-функцій передайте проект в CPU 214DPM.

**Прикладна
програма в CPU
21xDP**

Як показано вище, прикладна програма в CPU повинна виконувати два завдання, які будуть оброблятися в двох OB:

- завантаження вхідного байту від PROFIBUS і передача його значення до модуля виходів.

OB 1 (циклічний виклик)

```

L   PEW 100          Load status data and save in
T   MW 100           flag word

UN   M 100.5         Commissioning by the DP master
BEB                                     occurred? No -> End

U   M 101.4         Valid receive data?
BEB                                     No -> End
L   B#16#FF         Load control value and compare
L   PEB 30           to control byte
<>I                    (1st input byte)
BEB                 Received data does not contain
                        valid values

L   B#16#FE         Control byte for master-CPU
T   PAB 40

-----
Data exchange via PROFIBUS

L   PEB 31          Load peripheral byte 31 (input
T   AB 0            data from PROFIBUS slave) and
                        transfer into output byte 0

BE

```

- зчитування значення лічильника з MB 0, його збільшення, збереження в MB0 і передача до CPU 21x через PROFIBUS.

OB 35 (таймер-OB)

```

L   MB 0            Counter from 0x00 to 0xFF
L   1
+I
T   MB 0

T   PAB 41          Transfer counter value into
                        peripheral byte 41 (output data
                        of the PROFIBUS slave)

BE

```