

При виявленні перевантаження бажана також оцінка її величини і причини появи. Контроль завантаження буферних нагромаджувачів організується дуже просто шляхом перевірки записуваних адрес, підлягаючих обробці повідомлень і корисний не тільки при прийомі, але й при видачі повідомлень. Це дозволяє, зокрема, виявляти помилки в тестах, що ведуть до порушення нормального темпу формування тестів для деяких ОД.

Застосовуючи алгоритмічні методи захисту, можна істотно знизити шкідливий вплив перевантажень ресурсів системи і адаптацію алгоритмів на рівень припустимого завантаження. Зокрема така адаптація істотно знижує вплив помилок у структурі побудови алгоритмів, що ведуть до невеликих короткочасних перевантажень.

**Висновки.** Особливості забезпечення стійкості діагностичного процесу, оперативний захист від переключень інформації й обчислювального процесу може використовуватися як засіб виявлення помилок які складно виявляються, що особливо необхідно на завершальних етапах діагностування та у процесі експлуатації системи діагностування. Головна задача оперативного захисту від різних переключень складається в забезпеченні безперервності процесу керування з припустимими помилками у вихідних повідомленнях ОД. В системі діагностування використовуються наступні міри для забезпечення стійкості діагностичного процесу: відновлення інформації та збереження стійкості процесів діагностування, ігнорування виявленого переключення внаслідок його слабого впливу на весь процес діагностування і на вихідні результати, повторення функціонального алгоритму чи тесту при тих же вихідних даних, виключення тесту з обробки внаслідок його переключеності чи труднощів відновлення діагностичного процесу, короткочасне припинення розв'язання задач даного функціонального алгоритму до відновлення вихідних даних, перебудова режиму роботи алгоритму для зниження впливу перевантаження в зв'язку з втратою інформації про хід діагностичного процесу.

### Література

1. Хмельницький Ю.В. Метод адаптивного діагностування комп'ютерних мереж Вісник ТУП, № 3 / Хмельницький Ю.В. – Хмельницький : ХНУ, 2003. – С. 43– 48.
2. ДСТУ 2389-94. Технічне діагностування та контроль технічного стану. Терміни та визначення. – К. : Держстандарт України, 1994. – 24 с.
3. Хмельницький Ю.В. Дослідження та аналіз несправностей локальних обчислювальних мереж Вісник ТУП, № 2 / Хмельницький Ю.В. – Хмельницький : ХНУ, 2003. – С. 152– 155.

Надійшла 25.9.2012 р.  
Рецензент: д.т.н. Мясіщев О.А.

УДК 681.518

**В.П. НЕЗДОРОВІН, К.Л. ГОРЯЩЕНКО**

Хмельницький національний університет

**Є.Г. МАХРОВА**

Буковинський державний медичний університет

## РЕАЛІЗАЦІЯ ПРОТОКОЛУ MODBUS В СЕРЕДОВИЩІ CODESYS 2.3

*Розглянуто застосування сучасних програмних середовищ розробки програмного забезпечення для програмованих логічних контролерів на прикладі CoDeSys.*

*Application of modern software development environments software for programmable logic controllers on the example of CoDeSys.*

Ключові слова: програмований логічний контролер, SCADA.

Задачі автоматизації технологічного процесу на підприємстві можуть бути розв'язані шляхом використання засобів комп'ютерної та мікропроцесорної техніки. Основу реалізації автоматизації сучасного промислового підприємства складають промислової комп'ютери – ПЛК (програмовані логічні контролери), задача яких полягає у виконанні програми користувача із високим ступенем автономності. Найперший в світі програмований контролер Modular Digital Controller (Modicon) 084, що мав пам'ять всього 4 кбайт, був виготовлений у 1968 р. ПЛК розроблені для заміни релейних схем керування, які виготовлені із застосуванням дискретних елементів: реле, таймери, лічильники, елементи жорсткої логіки.

Принцип роботи ПЛК полягає у зборі та обробці даних згідно прикладної програми користувача. Наслідком виконання програми є формування вихідної послідовності сигналів на виконавчі пристрої.

Найбільш типові галузі застосування програмованих (інтелектуальних) контролерів:

- Автоматизація невеликих агрегатів для виробництва, збирання, обробки і упаковки;
- Автоматизація сільськогосподарських сфер (системи іригації, теплиці);
- Автоматизація шлагбаумів, відкатних воріт, систем контролю доступу;
- Автоматизація компресорів та систем кондиціонування повітря;

- Управління освітленням відповідно до різних заданих алгоритмів;
- Регулювання температури і вентиляції в житлових приміщеннях і на підприємствах;
- Управління водопостачанням будинку, фонтанами, акваріумами, насосними станціями;
- Управління транспортерами і змішувачами;
- Управління апаратурою на рухомій техніці, верстатами, виробничими лініями;
- Забезпечення сигналізації та оповіщення.

Ринок засобів і систем автоматизації насичений промисловими контролерами самих різних виробників, серед яких такі відомі фірми, як ABB, Allen Bradley, Honeywell, Omron, Moore Products, PEP Modular Computers і багатьох інших.

Побудова системи збору та обробки інформації можлива лише із застосуванням двох типових підходів. Перший шлях – це створення контролера, що має у своєму складі всі необхідні порти вводу та виведення. Другий шлях – використання універсального контролера, який реалізує можливість обміну даними за допомогою сторонніх блоків вводу-виведення, які підключені до нього.

Використання спеціалізованого контролера доречно лише за умови розміщення джерел інформації та виконавчих вузлів безпосередньо біля контролера або на достатній відстані від нього, при якій не має місця втрат інформації при передачі від джерела до ПЛК та назад. На промисловому підприємстві зазвичай немає можливості розмістити всі вузли на прийнятній відстані. Часто система збору інформації мусить працювати з окремими вузлами системи збору даних, що розміщені на відстанях порядку 20-1000 м від контролера.

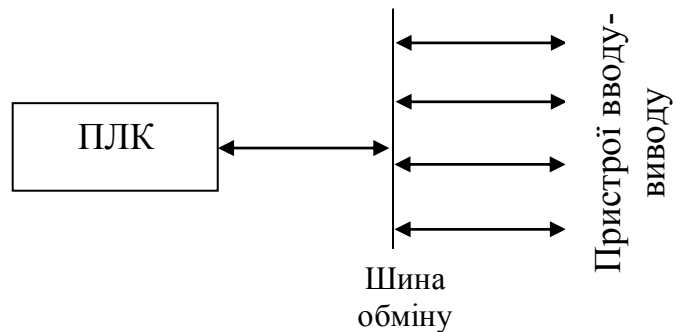


Рис. 1. Застосування універсального контролера

Збір інформації з системах збору та обробки даних виконується відповідно до обраного режиму, наприклад, циклічного опитування в системі Master-Slave або оновлення даних на складальному контролері шляхом передачі повідомлення по події при мульти-мастерній системі. Таким чином, до системи в цілому ставляться такі вимоги:

- гнучкість топології - легкість зміни як самої структури мережі, так і додавання в неї нових сегментів і інтелектуальних пристроїв;
- прийнятно висока швидкість обміну інформацією, що забезпечить своєчасну реакцію пристроїв системи на запити і керуючі команди від пульта управління;
- пропускну здатність каналу лінії зв'язку, що дозволить зменшити можливість виникнення колізій і забезпечить передумови для подальшого розширення функцій системи в цілому;
- адресація і (або) маршрутизація в мережі - запити зверху надходять тільки певному сегменту, в якому знаходиться запитувана пристрій, що зменшить інформаційну завантаження мережі;
- висока завадозахищеність передачі інформації - система повинна гарантувати доставку даних протягом певного інтервалу часу.

Для реалізації таких сукупностей вимог від ПЛК вимагається використання апаратних та програмних елементів реалізації певного протоколу взаємодії. Оскільки в вимогах є вимога топології мережі та її гнучкість, то це обумовлює або створення в процесорному комплексі власного протоколу обміну даними та реалізації або використання універсального протоколу.

Створення власного протоколу обміну даними є достатньо складною задачею, яку може собі дозволити лише відносно крупна фірма-виробник, яка має людський ресурс програмістів, інженерів-конструкторів, зв'язок з виробництвом та розгалужену мережу представництв. Лише за цих умов, розроблений власний протокол та його реалізація можуть бути витребувані кінцевими споживачами.

Використання універсального протоколу є більш прийнятним варіантом. Будь-який виробник ПЛК або окремих модулів вводу-виводу мусить лише використати один з відкритих або ліцензованих протоколів і забезпечити повну або часткову в межах допустимого реалізацію. Тоді споживач зможе використати модуль вводу-виводу та замінити його на модуль іншого виробника.

Одним зі зручних протоколів є протокол Modbus. Протокол Modbus фірми MODICON є необхідною частиною роботи системи. Він визначає наявність модулів як Master (MS, Майстер або Головний) і Slave (SL, Підлеглий). Протокол управляє циклом запиту і відповіді, який відбувається між пристроями MS і SL. Хоча його і зараховують до прикладних протоколів, тим не менш таким способом однозначно його визначити не можна. Річ у тому, що протокол Modbus крім власне передачі даних, ще і виконує такі завдання: контроль цілісності повідомлення (транспортний рівень), виконує задачі комутації (мережевий рівень), визначає початок і кінець передачі повідомлення (канальний рівень). Така невизначеність протоколу Modbus пояснюється тим, що він призначений для комунікації простих пристроїв, в яких немає достатнього об'єму пам'яті та обчислювальної потужності.

Протокол Modbus передбачає на загальній шині лише одного MS-пристрою і до 247 SL-пристроїв. Хоча протокол і підтримує до 247 SL, деякі прилади обмежують число SL, що підключаються до загальної

шини. Кожному SL-пристрою присвоєно унікальну адресу пристрою в діапазоні від 1 до 247. Тільки MS-пристрій може ініціювати передачу. Передачі бувають або типу запит / відповідь (адресується тільки один SL), або ширококомвні / без відповіді (адресуються все SL). Передача містить один кадр запиту і один кадр відповіді, або один кадр ширококомвного запиту.

Деякі характеристики протоколу Modbus фіксовані. До них відносяться формат кадру, послідовність кадрів, обробка помилок комунікації і виняткових ситуацій, і виконання функцій. Інші характеристики вибираються користувачем. До них відносяться тип засоби зв'язку, швидкість обміну, перевірка на парність, число степових біт, і режим передачі (ASCII або RTU). Параметри, обрані користувачем, встановлюються (апаратно або програмно) на кожній станції. Ці параметри не можуть бути змінені під час роботи системи.

При передачі по лініях даних, повідомлення поміщаються в «конверт». «Конверт» залишає пристрій через «порт» і «пересилається» по лініях до пристрою. Протокол Modbus описує «конверт» у формі кадрів повідомлень. Інформація в повідомленні представляє адресу потрібного одержувача, що одержувач повинен зробити, дані, необхідні для виконання цього, і механізм контролю достовірності.

Коли повідомлення досягає інтерфейсу SL, воно потрапляє в пристрій, що адресується через схожий «порт». Адресується пристрій розкриває конверт, читає повідомлення, і, якщо не виникло помилок, виконує необхідну задачу. Потім воно поміщає в конверт відповідь повідомлення і посилає його «відправникові». Інформація у відповідному повідомленні являє собою адресу адресується пристрою, виконану задачу, дані, отримані в результаті виконання завдання, і механізм контролю достовірності. Якщо повідомлення було ширококомвним (повідомлення для всіх SL), на що вказує адресу 0, то у відповідь повідомлення не передається.

У більшості випадках, MS посилає наступне повідомлення іншому SL або після прийому коректного відповідного повідомлення, або після проходження певного користувачем інтервалу часу, якщо Вам відповідь не був отримано. Всі повідомлення можуть розглядатися як запити, що генерують відповідні повідомлення від SL. Широкомвні повідомлення можуть розглядатися як запити, які не потребують відповідних повідомлень від SL.

Для розробки прикладного програмного забезпечення ПЛК перед розробником виникає задача реалізації алгоритму виконання поставленої задачі. Одночасно з цим мають бути розв'язані і супутні задачі забезпечення взаємодії базової програми з процедурами обміну даних з зовнішнім середовищем.

Для ПЕОМ функції обміну розміщуються на рівні операційної системи. А тому, взаємодія з ними виконується через виклики стандартних процедур та функцій. Можливості налаштування процедур обміну також у більшості випадків залишаються поза можливостями доступу. Наприклад, взаємодія із жорстким диском, така як запис, зчитування, послідовність зчитування та буферизація – визначаються безпосередньо розробником операційної системи.

Для ПЛК ситуація кардинально різниться. ПЛК має реалізовувати максимальну взаємодію з зовнішнім середовищем із можливістю встановлення потрібних параметрів доступу. Розглянемо взаємодію програми користувача в середовищі CoDeSys 2.3 з реалізацією протоколу Modbus. На рис. 2 показано конфігураційне вікно обраного ПЛК. В даному прикладі показано ПЛК фірми ОВЕН типу ПЛК-110-60.

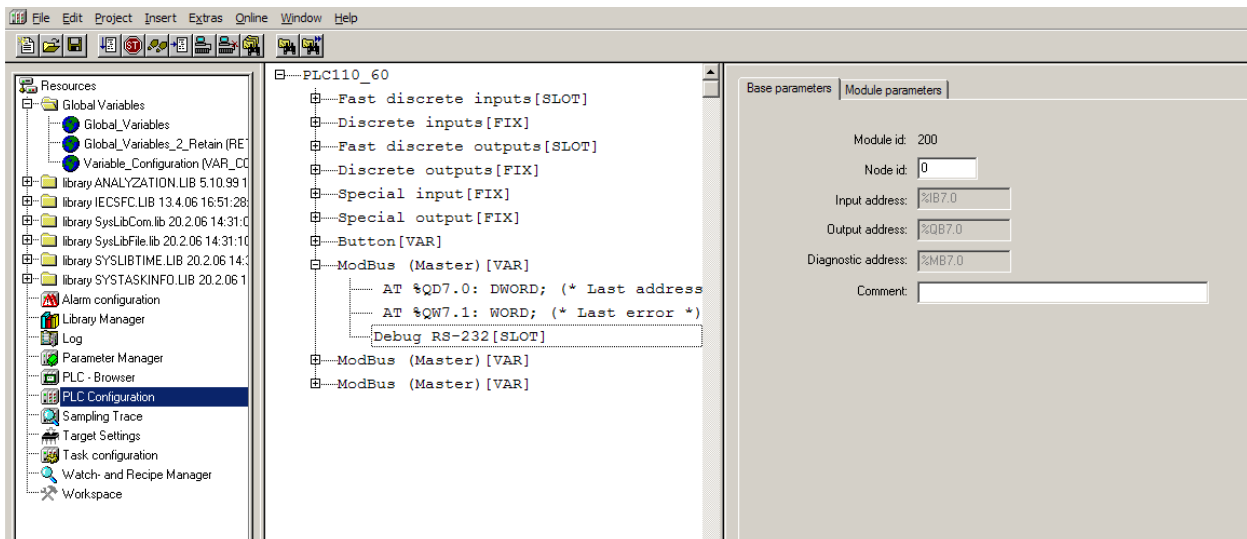


Рис. 2. Встановлення конфігурації ПЛК та включення протоколу Modbus

В середовищі CoDeSys 2.3 за реалізацію протоколу Modbus відповідає вже розроблена бібліотека, а тому залишається встановити потрібні параметри. Для цього потрібно виконати декілька послідовних дій.

По перше, це обрати потрібний порт взаємодії. Для ПЛК-110-60 доступні два порти RS-485, а також два порти RS-232. Для реалізації мережі пристроїв обираємо RS-485.

Наступним кроком необхідно підключити так званий "Universal Modbus device" – віртуальний елемент взаємодії, який дозволяє описати такі параметри як швидкість обміну, адреса підлеглого пристрою

та режим обміну між ним та ПЛК (рис. 3).

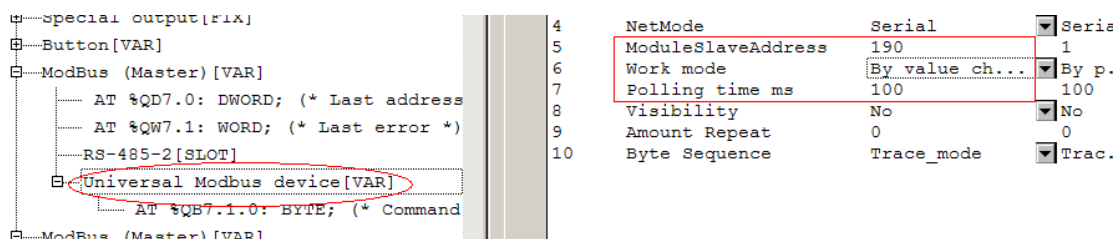


Рис. 3. Параметри Universal Modbus device

В розділі "Параметри" параметр ModuleSlaveAddress визначається адреса підлеглого пристрою (в прикладі – 190). Наступним важливим параметром є режим роботи – Work mode. Пропонується:

1) By poll time – за часом опитування. Всі операції обміну затримуються в часі, поки не пройде певний час, значення якого визначається в змінній Polling time. В цьому режимі обмін виконується циклічно, навіть якщо жодні данні не були змінені. Зазвичай цей режим краще використовувати при необхідності виконання зчитування з підлеглого пристрою.

2) By value change – при зміні значення. В цьому випадку, взаємодія із підлеглим пристроєм буде виконана лише за умови зміни значення вихідної змінної. Якщо такої зміни не буде, то і циклу виводу також не буде.

3) Both – комбінація попередніх двох режимів.

4) By command. В цьому випадку, Universal Modbus device може бути ввімкнено або вимкнено у довільний момент часу шляхом запису відповідної команди у керуючий регістр. Адреса цього регістра автоматично генерується. В прикладі на рис. 3 це байт за адресою %QB7.0.1. Найкраще використання цього режиму для реалізації періодичної зупинки каналу та відповідного звільнення ресурсів системи для виконання інших задач обміну даними.

Важливий елемент модуля – послідовність видачі байт. Визначається змінною Byte Sequence, яка приймає значення Trace\_mode та Native. Порядок визначається пристроєм згідно документації на модуль. Причому слід зауважити, що модулі одного виробника можуть оперувати різними режимами роботи. Так модулі вводу-виводу MB110-224.4ТД, МУ110-16Р-32 фірми OVEN вимагають порядку типу Trace\_mode. В той же час як панель СП270 тієї ж фірми – порядок типу Native.

Наступний крок це додавання змінних, значення яких будуть зчитуватись або записуватись в модулі – рис.4, 5.

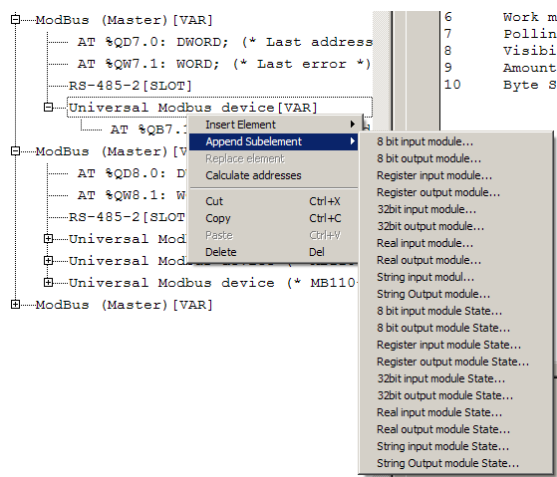


Рис. 4. Додавання змінних

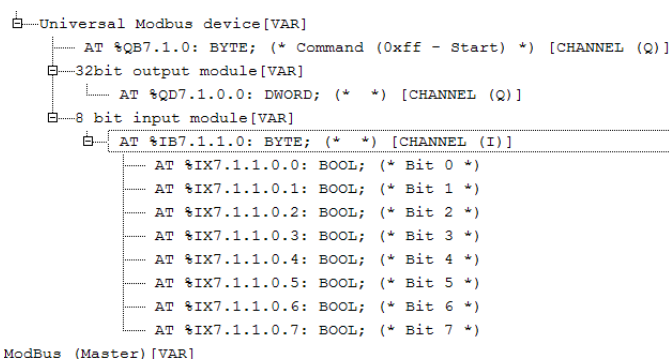


Рис. 5. Змінні типу 32 та 8-біт

Взаємодія зі змінними відбувається також без складних маніпуляцій. У ресурсах проекту в розділі "Global variables" необхідно описати змінні та прив'язати їх до модуля Modbus. Однією з особливостей слід відзначити можливість оперувати бітами 8-бітної змінної (змінна адресою %IB7.1.1.0 на прикладі):

```

VAR_GLOBAL
IntVar      AT %QD7.1.0.0      : INT;
FlagByte    AT %IB7.1.1.0     : BYTE;
FlagBit1    AT %IB7.1.1.0.3   : BOOL;      (* 4-й біт *)
END_VAR
    
```

Відповідна взаємодія з цими змінними відбувається з тіла програми як з простими внутрішніми змінними:

```

FUNCTION MODBUSIO : BOOL
IF (FlagBit1) OR (FlagByte = 33)
    
```

```

THEN IntVar := 450;
ELSE IntVar := 124;
END_IF;

```

### Висновки

1. Середовище CoDeSys надає дуже зручний механізм розробки прикладної програми взаємодії з протоколом Modbus, повністю забираючи на себе всі необхідні процедури та функції з реалізації обміну, встановлення порядку обміну даними із зовнішнім обладнанням.

2. Прикладне програмне забезпечення в середовищі програмування вимагає від розробника лише адекватного опису змінних та їх типу.

### Література

1. Руководство пользователя по программированию ПЛК в CoDeSys 2.3// 3S - Smart Software Solutions GmbH, Редакция RU 2.4, для CoDeSys V2.3.6.x. – 2006. – 453 с.

2. Минаев И. Г. Программируемые логические контроллеры в автоматизированных системах управления / И. Г. Минаев, В. М. Шарапов, В. В. Самойленко. – 2-е изд., перераб. и доп. – Ставрополь: АГРУС, 2010. – 128 с.

Надійшла 14.7.2012 р.

Рецензент: д.т.н. проф. Троцишин І.В.

УДК 621.391

О.В. КАЛЬВАТИНСЬКИЙ

Центр прийому і обробки спеціальної інформації та контролю навігаційного поля

А.А. ТАРАНЧУК, О.О. ПОСДИНЧУК

Хмельницький національний університет

## ЕФЕКТИВНІСТЬ ВИКОРИСТАННЯ ЗАВАДОСТІЙКОГО КАСКАДНОГО КОДУВАННЯ В ЗАДАЧАХ ДИСТАНЦІЙНОГО ЗОНДУВАННЯ ЗЕМЛІ

*Розроблена імітаційна модель каналу зв'язку «супутник – приймальна станція» на базі пакета математичного моделювання MATLAB/SIMULINK, підтверджена ефективність використання каскадних кодових конструкцій, зокрема, спільного використання згорткових кодів та кодів Ріда – Соломона в системах дистанційного зондування Землі.*

*The simulation model of communication link "the satellite - receiving station" of the base a packet mathematical simulation of MATLAB/SIMULINK is developed. Efficiency of use of the cascade code constructions, on the basis of sharing convolution codes with Reed – Solomon codes in system of remote sensing of Earth is shown.*

Ключові слова: канал зв'язку, завадостійке кодування, згортковий код, код Ріда – Соломона, енергетичний виграв.

**Постановка проблеми.** Подальший розвиток космічної галузі в області вирішення задач дистанційного зондування Землі (ДЗЗ) характеризується кластерним запуском космічних апаратів (КА) (запуск до шести КА малої маси одним ракетноносієм). Прикладом таких запусків є запуск з космодрому Байконур одним російським ракетноносієм «Дніпро» п'яти КА RapidEye (Німеччина) вагою 150 кг (рис. 1), а також запуск в одному кластері: білоруського комічного апарата БКА (вага 400 кг), російського "Канопус - В" вагою 450 кг, малого космічного апарату для фундаментальних космічних досліджень (МКА-ФКД) вагою до 100 кг, німецького TET-1, канадського ADS-1В одним російським ракетноносієм "Союз/Фрегат" [1].



Рис. 1. Супутники RapidEye на космодромі Байконур

Загальна тенденція по зменшенню маси КА призводить до погіршення енергозабезпечення космічних апаратів і відповідно до зменшення потужності їх передавальних пристроїв. При цьому, за специфікацією CCSDS 131.0-B-1 для систем космічного зв'язку значення вірогідності похибки на біт (англ. Bit Error Rate – BER) не повинне перевищувати рівень  $10^{-6}$  при енергетичному відношенні сигнал/шум