



УКРАЇНА

(19) UA (11) 91244 (13) C2  
(51) МПК (2009)  
G06F 7/00  
G06F 9/06  
G06F 15/76  
H04L 12/24

МІНІСТЕРСТВО ОСВІТИ  
І НАУКИ УКРАЇНИ

ДЕРЖАВНИЙ ДЕПАРТАМЕНТ  
ІНТЕЛЕКТУАЛЬНОЇ  
ВЛАСНОСТІ

## ОПИС ДО ПАТЕНТУ НА ВИНАХІД

(54) МЕРЕЖЕВИЙ МОДУЛЬ ОБРОБКИ ДАНИХ З ДИСТАНЦІЙНОЮ РЕКОНФІГУРАЦІЄЮ

1

2

(21) a200806336

(22) 13.05.2008

(24) 12.07.2010

(46) 12.07.2010, Бюл.№ 13, 2010 р.

(72) МАЙКІВ ІГОР МИРОСЛАВОВИЧ, КОЧАН РО-  
МАН ВОЛОДИМИРОВИЧ, ТУРЧЕНКО ІРИНА ВА-  
СИЛІВНА, КОЧАН ВОЛОДИМИР ВОЛОДИМИРО-  
ВИЧ

(73) МАЙКІВ ІГОР МИРОСЛАВОВИЧ, КОЧАН РО-  
МАН ВОЛОДИМИРОВИЧ, ТУРЧЕНКО ІРИНА ВА-  
СИЛІВНА, КОЧАН ВОЛОДИМИР ВОЛОДИМИРО-  
ВИЧ

(56) UA 25609; 30.10.1998

US 6417691 B1; 09.07.2002

WO 9414123; 23.06.1994

US 6282627 B1; 28.08.2001

EP 1351154 A2; 08.10.2003

US 20060206690 A1; 14.09.2006

US 20030097187 A1; 22.05.2003

US 20040098569 A1; 20.05.2004

US 20050055537 A1; 10.03.2005

(57) Мережевий модуль обробки даних з дистан-  
ційною реконфігурацією програмного забезпечен-

ня, в склад якого входять адаптери інтерфейсів зв'язку з сервером розподіленої системи та пристроями нижчого ієрархічного рівня, а також обчислювальний елемент, що складається з мікроконтролера і під'єднаних до нього регістра адреси та оперативного запам'ятовуючого пристрою, вхід запису якого підключений безпосередньо до відповідного виходу мікроконтролера, а вхід читання - до виходу елемента I, входи якого підключені до виходів читання даних і читання команд мікроконтролера, який **відрізняється** тим, що в склад модуля введено другий обчислювальний елемент, аналогічний першому, і окремий інтерфейсний мікроконтролер, до входів і виходів якого підключено входи і виходи згаданих адаптерів інтерфейсів, входи і виходи послідовних інтерфейсів обох обчислювальних елементів, входи скидання і керування режимом завантаження або виконання робочої програми обох обчислювальних елементів, причому програма завантаження робочої програми розміщена у внутрішній пам'яті програм мікроконтролерів обох обчислювальних елементів.

Запропонований пристрій відносяться до засобів обчислювальної техніки, зокрема комп'ютерних вимірювальних та керуючих систем і мереж. Він забезпечує можливість і високу надійність дистанційної заміни програмного забезпечення модулів мережі (реконфігурації) без припинення їх роботи.

Сучасні вимірювальні та керуючі системи найчастіше будуються на базі мережевих технологій з використанням ієрархічного принципу розподілу функціональних обов'язків компонентів [1, 2]. Такі системи називають розподіленими (дистрибутивними). Вони традиційно складаються з:

1. Сервера мережі, який підтримує функціонування мережі в цілому, зокрема, взаємодіє з оператором мережі та її технічними засобами - периферійними модулями нижчих ієрархічних рівнів і віддаленими користувачами даних різних ієрархічних рівнів, якщо такі є;

2. Периферійних вимірювальних та керуючих модулів, які безпосередньо взаємодіють з давачами та виконавчими пристроями системи. Загальною тенденцією є передача все більшого об'єму обробки даних цим периферійним модулям, виходячи з принципу - дані повинні оброблятися як найближче до місць їх виникнення та використання. Це дозволяє суттєво підвищити гнучкість системи, її живучість, а, в ряді випадків, знизити вартість, особливо при побудові периферійних модулів на дешевих мікроконтролерах. При цьому необхідно враховувати обмежені обчислювальні ресурси мікроконтролерів.

(13) C2

(11) 91244

(19) UA

Існує багато різних інформаційних технологій, а також технологічних процесів та об'єктів керування, які вимагають частої зміни алгоритмів функціонування, зокрема вимірювання та управління. Для розподілених комп'ютерних систем і мереж це означає потребу заміни (реконфігурації) програмного забезпечення їх периферійних модулів [3...8]. Найчастішим способом є перепрограмування постійного запам'ятовуючого пристрою (ПЗП), де зберігається програма роботи модуля [9], але вона вимагає відключення модуля (і, найчастіше, об'єкта, з яким модуль співпрацює), вилучення ПЗП, його перепрограмування з допомогою спеціального програматора, встановлення в модуль, вмикання та виведення на робочий режим. Таке перепрограмування трудомістке і використовується як крайній захід.

Можливе розміщення в постійній пам'яті (ПЗП або на дисковому накопичуванні) всього набору можливих варіантів попередньо відлагоджених прикладних програм (ПП) та керуючої системної програми - операційної системи (ОС), яка реалізує алгоритми їх вибору [10]. При цьому реконфігурація ПП може здійснюватись оператором шляхом натисканням кнопок на панелі модуля [5], або дистанційно, з допомогою "інженерної станції" [6], роль якої може виконувати сервер мережі. При цьому в оперативний запам'ятовуючий пристрій (ОЗП) модуля можуть завантажуватись нові значення констант для ПП. Недоліками цього способу є значні надлишкові апаратні затрати:

1. Збільшений об'єм постійної пам'яті через необхідність зберігати увесь набір ПП і ОС;
2. Збільшений об'єм ОЗП через потребу використовувати його частину для роботи ОС;
3. Необхідність значного збільшення продуктивності процесора через використання ОС частини його ресурсів.

Крім того, розширення бібліотеки варіантів ПП в процесі експлуатації системи є складним процесом - він відповідає процесу перепрограмування ПЗП, який використовує попередній аналог. Тому цей спосіб доцільно використовувати тільки тоді, коли всі алгоритми і режими роботи модуля відомі до його програмування, їх кількість мала, а процесор має надлишкову продуктивність.

Відома також дистанційна реконфігурація розміщених в ОЗП ПП робочих станцій комп'ютерної мережі під час її безперервної роботи [11] з допомогою спеціальної програми - системного завантажувача, розміщеного в ПЗП робочої станції (контролера), яка взаємодіє з сервером мережі, при умові відповідної архітектури робочої станції (типу фон Неймана) із суміщеним адресним простором пам'яті програм і даних. Такі мережі з бездисковими робочими станціями реалізовані на персональних комп'ютерах типу IBM PC (наприклад, Novell Netware з адаптером Ethernet), контролерах з використанням процесорів Z80 та ін. Цей спосіб принципово знижує затрати на зберігання бібліотеки ПП та дозволяє використовувати програми, які з'явилися під час експлуатації мережі за рахунок її адаптації і самонавчання. Але таке рішення не є оптимальним за затратами на реалізацію контролера. Крім того, згаданий спосіб вимагає мо-

дифікації адресів у завантаженій програмі. Це, поперше значно ускладнює відлагодження ПП на емуляторі, а, по-друге, служить джерелом збоїв через некоректно модифіковані адреси (особливо в програмах, скомпільованих компіляторами різних фірм). Останнє вимагає спеціального відслідковування коректності модифікації адресів і строгої заборони використання абсолютних адресів. Тому при використанні в системах, в яких відлагодження в складі системи є небажаним, описаний спосіб не забезпечує надійної роботи мережі.

Прототипом запропонованого пристрою є периферійний мережевий модуль [12] на базі мікроконтролера (однокристалної мікро-ЕОМ), яка має архітектуру фон-Неймана (або таку версію Гарвардської архітектури, яка може бути приведена до архітектури фон-Неймана) та може працювати як з внутрішньою так і зовнішньою пам'яттю програм. Зокрема, можна використати найбільш широко вживані мікроконтролери серії i51 (MCS51) або аналогічні [13]. В традиційну структуру модуля на базі мікроконтролера (мікроконтролер, до якого приєднано, через регістр адресу, ОЗП і системний дешифратор) введено додатковий тригер, вихід якого приєднано до входу формувача імпульсу скидання мікроконтролера і до його контакту, що вказує місце розміщення програми, яка має виконуватись (внутрішня чи зовнішня пам'ять). До другого входу формувача імпульсу скидання мікроконтролера підключено давач напруги живлення схеми. Входи додаткового тригера підключено до виходів давача напруги живлення схеми і системного дешифратора. Якщо використано мікроконтролери серії i51 необхідно привести їх до Гарвардської архітектури. Для цього їх виходи RD (читання ОЗП) і PSEN (читання пам'яті програм) підключають до входів елемента І, вихід якого підключають до входу RD ОЗП. Після вмикання периферійного модуля давач напруги живлення діє на входи тригера і формувача імпульсу скидання мікроконтролера, скидаючи тригер і мікроконтролер в початковий стан. При цьому тригер вказує мікроконтролеру на необхідність виконання програми, розміщеної у внутрішній пам'яті (ПЗП). Виконуючи цю програму, мікроконтролер посилає серверу мережі запит на завантаження ПП. Сервер передає виконуваний код ПП модулю, мікроконтролер якого, під керуванням програми, розміщеної в ПЗП, записує цей код у зовнішній ОЗП. Після кінця запису виконуваного коду ПП мікроконтролер звертається до адреси, яка системним дешифратором розшифровується як команда перекидання додаткового тригера. Останній вказує мікроконтролеру на необхідність виконання програми, розміщеної у зовнішньому ОЗП і, через формувач імпульсу скидання мікроконтролера, скидає його. Мікроконтролер починає виконувати завантажену ПП. Перевагами прототипу є можливість використання ПП всіх ресурсів мікроконтролера та виконання ПП "як є" (відсутність будь яких змін в ній, внесених при завантаженні). Першим недоліком прототипу є непрацездатність модуля під час завантаження виконуваної ПП. Цей недолік не є значним при завантаженні ПП після ввімкнення живлення, однак, якщо заміну ПП проводять в

процесі експлуатації системи керування, можливі аварійні стани за рахунок втрати керування об'єктом. Це обмежує область використання прототипу системами, що керують об'єктами, які можуть на час заміни ПП бути переведеними в безпечний режим. Другим недоліком є низька стійкість проти помилок в завантаженій ПП, внесених як завадами в канал зв'язку, так і при написанні ПП. Якщо під дією помилок ПП зависла, або в частині ПП, яка забезпечує дистанційне перепрограмування, є помилка, заміна ПП стає неможливою без скидання мікроконтролера шляхом вимкнення і повторно-го ввімкнення модуля. Крім того, якщо канал зв'язку під час перепрограмування буде пошкоджено, модуль залишиться непрацездатним і для відновлення його працездатності необхідно знову провести скидання мікроконтролера шляхом вимкнення і повторного ввімкнення модуля.

Метою винаходу є підвищення надійності роботи периферійного модуля за рахунок забезпечення, по-перше, можливості гарантованого дистанційного перепрограмування і, по-друге, функціонування при пошкодженні каналу зв'язку під час перепрограмування.

Суть пропонованого винаходу полягає в тому, що в мережевий модуль обробки даних з дистанційною реконфігурацією програмного забезпечення, в склад якого входять адаптери інтерфейсів зв'язку з сервером розподіленої системи та пристроями нижчого ієрархічного рівня, а також обчислювальний елемент, що складається з мікроконтролера і під'єднаних до нього регістра адресу та оперативного запам'ятовуючого пристрою, вхід запису якого підключений безпосередньо до відповідного виходу мікроконтролера, а вхід читання - до виходу елемента І, входи якого підключені до виходів читання даних і читання команд мікроконтролера, введено окремий інтерфейсний мікроконтролер і другий обчислювальний елемент, аналогічний першому. При цьому входи скидання та входи, що вказують місце розміщення програми, яка має виконуватися (внутрішня чи зовнішня пам'ять), а також входи та виходи послідовних інтерфейсів обох обчислювальних елементів під'єднані до відповідних виводів інтерфейсного мікроконтролера, до якого також під'єднані входи і виходи адаптерів інтерфейсів. Центральним елементом модуля є інтерфейсний контролер, який керує обчислювальними елементами і забезпечує зв'язок з іншими ієрархічними рівнями системи (мережі). Поточну обробку даних виконує один з двох обчислювальних елементів, інший зазвичай знаходиться в стані скидання.

Після включення живлення інтерфейсний контролер посилає серверу запит на завантаження ПП. При цьому він переводить, використовуючи свої виходи, підключені до виходів скидання та входів, що вказують місце розміщення програми, яка має виконуватися, один з обчислювальних елементів (наприклад, перший) в стан скидання, а інший (наприклад, другий) - в стан виконання програми, розміщеної у внутрішньому ПЗП. Сервер передає виконуваний код ПП модулю. Його приймає інтерфейсний мікроконтролер і передає другому обчислювальному елементу, мікроконтролер

якого виконує програму, розміщену в ПЗП. Згідно цієї програми, останній записує цей код у зовнішній ОЗП. Після кінця запису коду ПП інтерфейсний мікроконтролер переводить мікроконтролер другого обчислювального елемента в стан виконання програми, розміщеної в ОЗП, і скидає його. Другий обчислювальний елемент починає поточну обробку даних, перший продовжує знаходитися в стані скидання. При необхідності заміни виконуваної ПП сервер посилає відповідну команду, яка розшифровується інтерфейсним мікроконтролером. Він переводить перший обчислювальний елемент, який досі був у стані скидання, у стан виконання програми, розміщеної в ПЗП. Надалі інтерфейсний мікроконтролер взаємодіє як з першим (працюючим) обчислювальним елементом (який продовжує обмін даними з сервером і модулями нижчих ієрархічних рівнів та обробку поточних даних), так і з введеним в дію другим обчислювальним елементом, якому передає код нової ПП, що поступає з сервера. Захист від можливих колізій виконується відомими способами [14]. Після закінчення процесу завантаження програми та її перевірки (наприклад, шляхом підрахунку контрольної суми), інтерфейсний мікроконтролер переводить другий обчислювальний елемент в стан скидання, а перший обчислювальний елемент переводить в стан виконання програми, розміщеної в ОЗП і скидає його. Після цього поточну обробку даних і обмін даними з сервером і модулями нижчих ієрархічних рівнів починає перший обчислювальний елемент, другий знаходиться в стані скидання. Як видно, під час заміни ПП (завантаження нової ПП) периферійний модуль (його другий обчислювальний елемент) продовжує виконання старої ПП. Розпізнавання команди сервера на заміну виконуваної ПП здійснює інтерфейсний мікроконтролер, який має стабе, відлагоджене і протестоване програмне забезпечення, робота якого ніяк не залежить від помилок в завантаженій ПП, зависань обчислювальних елементів та ін. Таким чином, мета винаходу - підвищення надійності перепрограмування периферійного модуля і його функціонування під час перепрограмування - досягається за рахунок пропонованої структури і розподілу функціональних обов'язків складових частин. Вилучення будь-якої складової частини, введеної в склад прототипу, або перерозподіл їх функціональних обов'язків не дозволяє досягнути мету винаходу.

Суть винаходу пояснюють зображена на Фіг.1 структурна схема пропонованого мережевого модуля обробки даних і зображені на Фіг.2 схеми передачі керування між обчислювальними елементами та інтерфейсним мікроконтролером.

Мережевий модуль обробки даних, структурна схема якого зображена на Фіг.1, складається з двох обчислювальних елементів OE1 і OE2, інтерфейсного мікроконтролера IFC та адаптерів інтерфейсів верхнього і нижнього рівнів мережі IF1 і IF2. Кожен обчислювальний елемент складається з мікроконтролера MCU, регістра адресу RGA, оперативного запам'ятовуючого пристрою RAM та логічного елемента І. Модуль, представлений на Фіг.1, для прикладу, побудований на базі мікроконтролерів 89C52, які зараз найпопулярніші. В цих

мікроконтролерах порт P0, при наявності логічної одиниці на виході ALE, виконує роль шини молодших адресів, а при наявності логічного нуля - шини даних. Тому порт P0 підключений до входу регістра RGA, куди записуються молодші адреси при наявності логічної одиниці на виході ALE, і до входів-виходів даних оперативної пам'яті RAM. Порт P2 виконує роль шини старших адресів, він безпосередньо підключений до відповідних входів оперативної пам'яті RAM. Сигнал запису WR даних в оперативну пам'ять RAM також безпосередньо підключений до відповідного її входу. А сигнали читання команд PSEN і даних RD об'єднані з допомогою логічного елемента І, чим досягається приведення Гарвардської архітектури мікроконтролерів 89C52 в архітектуру фон-Неймана. Інтерфейсний мікроконтролер IFC теж, для прикладу, виконаний на базі мікроконтролера 89C52. Він керує обчислювальними елементами OE1 і OE2 через їх вхід скидання Reset і вхід EA, що вказує місце розміщення програми, яка має виконуватися (внутрішня чи зовнішня пам'ять). Для прикладу, прийнято, що всі послідовні інтерфейси в модулі та в системі є одного типу - RS232. Однак мікроконтролери 89C52 мають тільки один вбудований апаратний інтерфейс цього типу. Його найдоцільніше використати як інтерфейс мережі верхнього рівня, виконавши підключений до неї адаптер (наприклад, IF1) відповідним чином [15]. Решту інтерфейсів доцільно реалізувати програмно.

Після включення живлення IFC через IF1 посилає серверу запит на завантаження ПП і подає на входи Reset і EA OE1 та OE2 логічні одиниці (скидає MCU обох обчислювальних елементів). Далі на вхід Reset OE2 подається логічний нуль і його MCU починає виконання програми, записаної у його внутрішній ПЗП. Сервер передає виконуваний код ПП, який приймає IFC і передає MCU OE2, який, згідно програми, розміщеної в його ПЗП, записує цей код у RAM OE2. Після передачі всього коду ПП MCU OE2 перевіряє правильність записаних кодів (наприклад, шляхом підрахунку контрольної суми ПП і порівняння її з переданою з сервера) і видає IFC або сигнал підтвердження правильності ПП, або сигнал збою. При отриманні сигналу збою IFC повторно скидає MCU OE2 і посилає серверу запит на повторну передачу коду ПП. При отриманні сигналу підтвердження IFC подає на вхід EA OE2 логічний нуль і повторно скидає MCU OE2. Після скидання MCU OE2 починає виконання завантаженої ПП, розміщеної в RAM OE2. При цьому ПП може повністю використовувати всі ресурси MCU OE2 і ніяка модифікація адресів непотрібна. OE1 залишається у стані скидання.

Для заміни ПП сервер посилає команду, яку розшифровує IFC. IFC подає на вхід Reset MCU OE1 логічний нуль, тобто закінчує стан скидання. MCU OE1 виконує програму, записану у його внутрішній ПЗП (яка аналогічна програмі OE2). Запис коду ПП у RAM OE1 відбувається аналогічно, як і у RAM OE2. Але IFC при цьому як передає від сервера до OE1 код ПП, так і проводить обмін даними між MCU OE2 (що продовжує виконувати стару ПП) та сервером і модулями нижчого ієрархічного

рівня. Після завантаження ПП у RAM OE1 та підтвердження її контрольної суми IFC подає логічний нуль на вхід Reset MCU OE2 (скидає MCU OE2) і на вхід EA MCU OE1 (режим виконання ПП, завантаженої в RAM OE1). На вхід Reset MCU OE1 подається імпульс логічної одиниці (скидання) і OE1 починає обробку поточних даних.

Пропонована структура модуля є ієрархічною, зі зміною підпорядкування в процесі роботи. Підпорядкування в ній пояснює Фіг.2. Під час завантаження ПП після ввімкнення живлення IFC щодо OE1 і OE2 є Master, OE1 і OE2 щодо IFC Slave (див. Фіг.2а). Під час роботи IFC є Master щодо OE1 і щодо режиму роботи OE2, а щодо виконання ПП, то OE2 є Master, а IFC є Slave (див. Фіг.2б). Під час заміни ПП IFC залишається Master щодо OE1 і щодо режиму роботи OE2, і Slave щодо виконання ПП OE2 (див. Фіг.2б). Після завантаження ПП і скидання OE1, IFC стає Master щодо OE2 (утримує OE2 в стані скидання) і щодо режиму роботи OE1, і Slave щодо виконання ПП OE1 (див. Фіг.2в). Таким чином, як видно з Фіг.2, IFC завжди є Master щодо режиму роботи OE1 і OE2, тому може в будь-який момент, незалежно від стану OE1, OE2, ПП, провести перепрограмування модуля, тобто заміну його ПП.

Використання запропонованого технічного рішення дозволяє, при незначних затратах, реалізувати мережевий модуль обробки даних з дистанційною реконфігурацією (заміною) програми його роботи в процесі експлуатації, без необхідності зупинки обробки даних, вимикання, використання програматорів. При цьому забезпечена висока надійність роботи модуля за рахунок гарантованого дистанційного перепрограмування в довільний момент і продовження функціонування по старій програмі при пошкодженні каналу зв'язку під час перепрограмування. Це дає змогу реалізувати гнучкі мережі вимірювально-керуючих модулів для застосувань в наступних системах:

1. Універсальні системи, де запропонований мережевий модуль взаємодіє з різноманітним існуючим і розробленим в процесі експлуатації обладнанням. Наприклад, системи, які поступово, в процесі експлуатації на протязі довгого часу, наріщуються і модернізуються. Перевага запропонованого модуля в таких системах полягає в тому, що він дає можливість простої заміни і нарощення програмного забезпечення;

2. Розподілені вимірювальні та керуючі системи, в яких часто змінюється периферійне обладнання. Наприклад, розподілені системи, які на різних етапах експлуатації повинні виконувати різні задачі. Зокрема, розподілені системи екологічного моніторингу, обладнання яких залежить від виду оцінюваного забруднення. Перевага запропонованого модуля в таких системах полягає в тому, що (i) запропонований модуль різко збільшує вагу незмінного апаратного обладнання; (ii) можливість простої заміни алгоритмів обробки даних полегшує заміну обладнання "на ходу";

3. Адаптивні вимірювальні та керуючі системи з динамічною заміною алгоритмів обробки даних без переривання роботи. Наприклад, системи, в яких відлагодження і оптимізація роботи одних

каналів відбувається в процесі експлуатації інших каналів. Перевага пропонованого модуля в таких системах полягає в тому, що він створює можливість простої заміни програмного забезпечення без вимкнення системи. При відлагодженні програмне забезпечення нових каналів може просто тимчасово додаватися до програмного забезпечення старих каналів. Після відлагодження програмне забезпечення може бути об'єднане і оптимізоване;

4. Адаптивні вимірювальні та керуючі системи, в яких алгоритми обробки даних не можуть бути визначені наперед. Зокрема це стосується складних багатоканальних систем регулювання з взаємним впливом каналів. Перевага пропонованого модуля в таких системах полягає в тому, що він створює можливість простої заміни всього програмного забезпечення в довільний момент часу, тому можливе почергове випробування великої кількості цілком різних алгоритмів або їх версій без необхідності зупинки системи для заміни пам'яті програм або надлишкового ускладнення системи;

5. Вимірювальні та керуючі системи, які використовують алгоритми обробки даних змінної структури. Наприклад, системи, які використовують штучні нейронні мережі або генетичні алгоритми. Для розміщення всіх вживаних версій алгоритмів (архітектур нейронних мереж або набору початкових алгоритмів для схрещення) для традиційного модуля необхідна пам'ять програм великого об'єму, значно надлишкового для його поточної роботи. Перевага пропонованого модуля в таких системах полягає в тому, що в кожен момент часу в пам'яті розміщена тільки та програма, яка реалізує поточну версію алгоритму обробки. Це зменшує необхідний об'єм пам'яті програм і дозволяє використовувати прості мікроконтролери в складі обчислювальних елементів;

6. Вимірювальні та керуючі системи, базовані на мережі Internet. Такі системи часто мусять бути системами реального часу, а Internet є комунікаційною мережею з негарантованим доступом і часом доступу. Тому на нижньому рівні таких систем часто пропонують використовувати відносно дорогі одноплатні комп'ютери, які зберігають в постійній пам'яті програм повний набір потрібного програмного забезпечення. Ці комп'ютери можуть підтримувати функціонування системи і при тимчасовій відсутності зв'язку через Internet. Перевага пропонованого модуля в таких системах полягає в тому, що він повинен мати в пам'яті програм тільки поточний алгоритм роботи. Якщо перерви в зв'язку через Internet перевищують необхідний реальний час зміни алгоритму роботи, то тільки тоді необхідно включати в поточне програмне забезпечення пропонованого модуля додаткові програми. Але включаються тільки ті програми, які можуть бути потрібні в найближчому майбутньому. Тому пропонований модуль може бути значно простішим від згаданих комп'ютерів;

7. Інтелектуальні вимірювальні та керуючі системи та мережі. Такі системи та мережі: (i) вимагають виконання всіх особливостей, перелічених в попередніх пунктах; (ii) вони повинні передбачати самонавчання в процесі експлуатації. При використанні

станні модулів традиційної структури ці процеси повинні виконуватися сервером - комп'ютером верхнього рівня. Тоді необхідний гарантований доступ до нього в реальному часі. Це вимагає значних запасів обчислювальних ресурсів сервера і швидкодіючого доступу до нього. Перевага пропонованого модуля в таких системах полягає в тому, що процеси адаптації і самонавчання можуть виконуватися в пропонованому модулі під контролем сервера мережі. Тоді вимога доступу до сервера стосується не до реального часу поточної обробки даних самонавчання, а до реального часу вдосконалення алгоритмів самонавчання. Це значно послаблює вимоги до комунікації між модулем і сервером. Створюється можливість використання як сервера потужних комп'ютерів, доступ до яких здійснюється по мережі типу Internet.

Таким чином, запропонований мережевий модуль обробки даних з дистанційною реконфігурацією має досить широку перспективу застосування.

Джерела інформації:

1. К. Кругляк. Промышленные сети: цели и средства. "Современные технологии автоматизации", №4, 2002, с.6...17. ([www.cta.ru/Rubrics/239890\\_239961.htm](http://www.cta.ru/Rubrics/239890_239961.htm)).

2. К. Кругляк. Локальные сети Ethernet в АСУ ТП: быстрее, дальше, надежнее. "Современные технологии автоматизации", №4, 2002, с.6...17. ([www.cta.ru/Rubrics/239890\\_239961.htm](http://www.cta.ru/Rubrics/239890_239961.htm)).

3. Аникин В.С. Система сбора и обработки информации "РГРТА-ЭП". Тез. докл. IX науч.-техн. конф. Датчики и преобразователи информации систем измерения, контроля и управления "Датчик-97". Под ред. проф. Азарова. - М.: МГИЭМ, 1997. с.318-319.

4. Цветков Э.И. Процессорные измерительные средства Л.Энергоатомиздат, 1989. - 224с.

5. Гудков Ю.И. Базовые модули обработки информации для портативных контрольно-измерительных устройств. Тез. докл. IX науч.-техн. конф. Датчики и преобразователи информации систем измерения, контроля и управления "Датчик-97". Под ред. проф. Азарова. - М.: МГИЭМ, 1997. с.318.

6. Сташин В.В. и др. Проектирование цифровых устройств на однокристалльных микроконтроллерах. - М.: Энергоатомиздат, 1990.

7. Корнеева А.И. Программно-технические комплексы отечественного производства // Приборы и системы управления. 1997, N 9. - с.9-12.

8. Шальман Л.М. Автоматизация систем теплоснабжения // Приборы и системы управления. 1997, N 11. - с.7-9.

9. Боборикин А.В. и др. Однокристалльные микро-ЭВМ. М.: МИКАП, 1994.

10. А.Образцов, С.Образцов. Операционная система реального времени для микроконтроллеров X51. "Электронные компоненты", №8, 2003, с.92...95.

([http://www.elcp.ru/index.php?state=izd&i\\_izd=elcomp&i\\_num=2003\\_08&i\\_art=21](http://www.elcp.ru/index.php?state=izd&i_izd=elcomp&i_num=2003_08&i_art=21)).

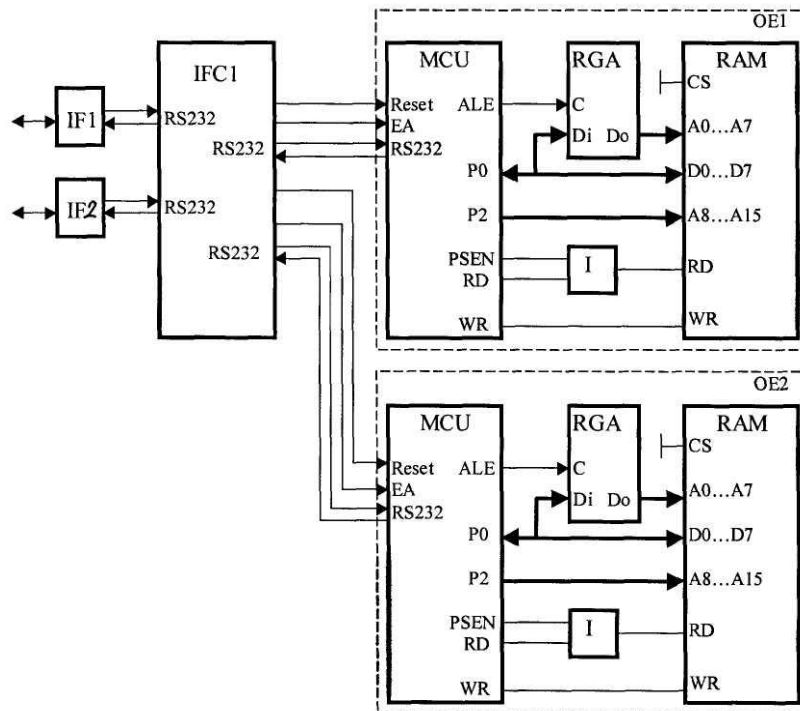
11. Гук М. Локальные сети Novell. - СПб: Питер, 1996. - 288с.

12. Васильків Н.М., Кочан В.В., Саченко А.О., Тимчишин В.О. Контролер з дистанційною реконфігурацією для адаптивної вимірально-керуючої мережі // Обчислювальна техніка. Вісник ДУ "Львівська політехніка". - 1998. - №287. - С.13-19.

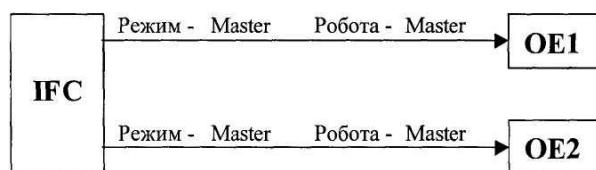
13. www.atmel.com.

14. Компьютерные сети. 4-е изд. / Э. Танненбаум. - СПб.: Питер, 2003. - 992с.

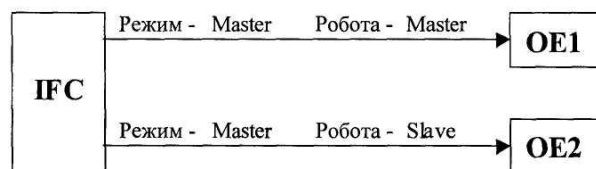
15. Патент 25609А України, МКІ G06F15/00. Двопровідна локальна обчислювальна мережа, повторювач сигналу та інвертор для використання в ній / В.В. Кочан, В.О. Тимчишин (Україна); Заявл. 30.10.97 №97105295; Видано 30.10.98.



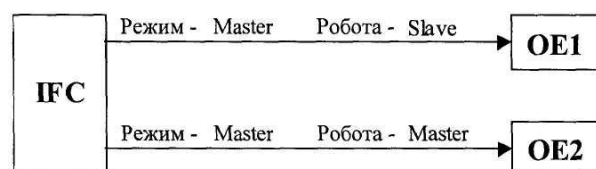
Фіг. 1. Структурна схема пропонованого мережевого модуля обробки даних з дистанційною реконфігурацією.



а) Стан інтерфейсного мікроконтролера щодо обчислювальних елементів після ввімкнення живлення.



б) Стан інтерфейсного мікроконтролера щодо обчислювальних елементів під час нормальної обробки даних і перепрограмування.



в) Стан інтерфейсного мікроконтролера щодо обчислювальних елементів під час нормальної обробки даних після перепрограмування.

Фіг. 2. Передача керування між інтерфейсним мікроконтролером та обчислювальними елементами