

*КЛИМЕНКО І.А.,
САКАРА Н. А.,
ФЕДОРЧУК М.В.*

РЕАЛІЗАЦІЯ КОНТРОЛЕРА ПРІОРИТЕТНИХ ПЕРЕРИВАНЬ ДЛЯ ОБЧИСЛЮВАЛЬНОЇ СИСТЕМИ З ВІДКРИТОЮ АРХІТЕКТУРОЮ

Стаття присвячена рішення проблемі підвищення продуктивності обчислювальних систем реального часу за рахунок вдосконалення процедур обробки переривань на апаратному рівні. З метою гарантованого обслуговування переривань від великої кількості зовнішніх пристроїв за певний проміжок часу запропоновано використання розподіленого контролера пріоритетних переривань з різними типами дисциплін динамічної обробки запитів. За результатами експериментальних досліджень доведено, що застосування запропонованого підходу усуває недоліки систем з географічними абсолютними пріоритетами, та сприяє вирішенню проблеми підвищення продуктивності систем реального часу з відкритою архітектурою.

The paper is dedicated to solving the problem of increasing productivity real time computing systems by improving the procedures for handling interrupts in hardware. To guarantee interrupt service from a large number of external devices for a certain period of time requested for the new priority interrupt controller with various types of disciplines dynamic query processing. Experimental studies proved that the application of the proposed approach eliminates the disadvantages of geographical systems an absolute priority, and to address the problem of increasing the productivity of real-time systems with open architecture.

Вступ

Підвищення продуктивності обчислювальних систем (ОС) реального часу є важливою задачею в області застосування автоматичних систем управління. Особливістю таких систем є функціонування в режимі реального часу, що вимагає забезпечення гарантій виконання жорстких часових вимог. Складність виконуваних задач підвищують вимоги до сучасних ОС реального часу і обумовлюють розробку нових апаратних та програмних засобів для підвищення ефективності обробки та обміну даними.

Огляд літератури в даній області показав, що підвищення продуктивності паралельних обчислювальних систем реального часу досягається застосуванням методів і засобів зменшення витрат часу на ініціалізацію і синхронізацію процесів обміну даними. Так в роботі [1] запропоновані методи автоматичного динамічного розпаралелювання обчислень на рівні команд, програмних модулів і програм у системах з різною архітектурою, метою яких є підвищення реакції систем управління на зовнішні події та розширення області їх застосування. В роботі [2] були запропоновані формули апаратно-орієнтованих алгоритмів швидких перспективних перетворень, адаптовані для програмно-апаратної реалізації процесорів реального часу, що задовольняють критеріям точності та

швидкодії. Окрім того, однією з функціональних особливостей систем управління реального часу є необхідність обробки переривань від великої кількості зовнішніх пристроїв. Це потребує ефективної реалізації системи переривань. На програмному рівні звичай питання обробки переривань вирішуються за допомогою спеціалізованих операційних системи реального часу, які базуються на системі пріоритетів процесів і алгоритмів планування [3].

В даній роботі проблема підвищення продуктивності систем реального часу вирішується за рахунок модифікації архітектури системи переривань, що забезпечить гарантоване обслуговування переривань від великої кількості зовнішніх пристроїв на протязі фіксованого часу циклу управління.

Постановка задачі

Цілі дослідження: забезпечення гарантованого обслуговування переривань від кожного зовнішнього пристрою та уникнення тупикових ситуацій, характерних застосуванню географічного пріоритету, за рахунок чого підвищення продуктивності обчислювальних систем реального часу.

Реалізація розподіленого контролера пріоритетних переривань (КПП) з фіксованими та динамічними рівнями пріоритетів запитів

на ПЛІС, виконання моделювання в САПР Quartus II.

Особливості реалізації КПП з фіксованими та динамічними пріоритетами в ОС

З точки зору швидкодії обробка переривань на апаратному рівні є більш ефективною в ОС розглянутого класу [3]. Для обробки переривань на апаратному рівні використовуються спеціалізовані пристрої – централизовані та розподілені КПП. Перевагами розподілених контролерів переривань є невелика кількість ліній зв'язку у шині управління та простота нарощування зовнішніх пристроїв. Недоліками такої системи є велика частота звернень процесора до системної магістралі під час ініціалізації системи, а також використання фіксованих рівнів пріоритетів запитів, які не забезпечують гарантованого обслуговування заявок від зовнішніх пристроїв. Запити з низьким рівнем пріоритету, за великої інтенсивності запитів, можуть не виконуватися тривалий час, що може призвести до сповільнення обчислювального процесу, а іноді – до тупикової ситуації. Досліджена обчислювальна система, структурна схема якої зображена на рис. 1. Система складається з процесора P (Processor) і зовнішніх пристроїв PU (Peripheral Unit), пов'язаних між собою загальною шиною GB (Global Bus). У системі реалізована обробка зовнішніх векторних переривань за допомогою розподіленого контролера переривань, у склад кожного PU входить блок обробки переривань IB (Interrupt Block). Готовий IB до обміну даними видає сигнал запити переривання IR (Interrupt Request) на загальну лінію вимоги переривань ID (Interrupt Demand). Відповідний сигнал процесора підтвердження переривання IA (Interrupt Acknowledgement) поширюється послідовно через усі IB , що утворюють так названий пріоритетний «ланцюжок» (Daisy Chain), пріоритети обробки запитів переривань від зовнішніх пристроїв залежать від їх географічного розташування по відношенню до процесора. Таким чином найвищий пріоритет має зовнішній пристрій, що розташований найближче до процесора, а самий останній у ланцюзі – має найнижчий пріоритет. Така система не забезпечує гарантованого обслуговування заявок від зовнішніх пристроїв на визначеному відрізку часу. Заявки з низьким рівнем пріоритету за великої інтенсивності заявок можуть не виконуватися

тривалий час. Цього можна уникнути за допомогою забезпечення динамічних пріоритетів. В КПП з послідовним передаванням пріоритету в кожному такті максимальний пріоритет, тобто початок ланцюга, передається наступному у ланцюгу процесору. За цього максимальний час очікування обслуговування дорівнює $(n - 1)$ тактів, де n – кількість зовнішніх пристроїв. За такого способу реалізації динамічних пріоритетів максимальний пріоритет (початок ланцюга) отримує зовнішній пристрій, наступний за тим, що був обслугований у поточному такті.

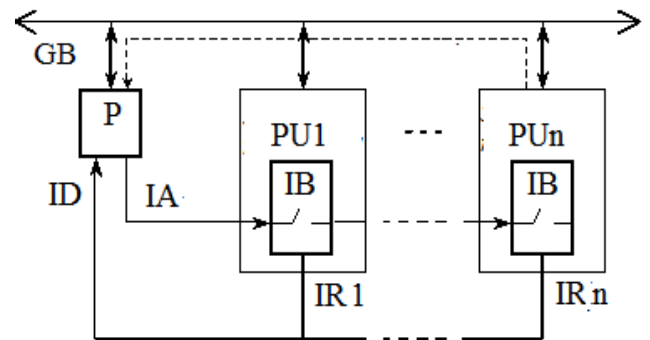


Рис. 1. Структурна схема обчислювальної системи з фіксованими пріоритетами

Обчислювальна система (рис. 2) містить процесор P , n зовнішніх пристроїв PU , загальну шину GB , до якої підключені процесор P та зовнішні пристрої PU . До складу кожного зовнішнього пристрою входить блок формування переривань IB , тригер D , елемент І (&) та елемент АБО (I) (апаратура зовнішніх пристроїв, яка не стосується реалізації переривань на рис. 2 умовно не показана). Вихід тригера D підключений до першого входу елемента І, вихід якого зв'язаний з першим входом елемента АБО, який своїм виходом підключений до входу блока переривань IB . Вихід $NEXT_TR$ кожного тригера D приєднаний до інформаційного входу DIN тригера наступного зовнішнього пристрою, при цьому тригер D останнього зовнішнього пристрою PU підключений до входу першого зовнішнього пристрою PU . Таким чином, тригери D об'єднані у кільце. Вихід $NEXT_PU$ кожного блоку переривань приєднаний до входу елемента АБО наступного зовнішнього пристрою, при цьому блок переривань останнього зовнішнього пристрою PU підключений до входу першого зовнішнього пристрою PU . Таким чином, блоки переривань IB через елементи АБО об'єднані у кільце. Виходи IR блоків переривань IB об'єднані у єдину лінію і підключені до входу вимоги переривань (ID) процесора P , вихід підтвердження переривання (IA) якого

підведений до керуючих входів тригерів D та других входів елементів I .

Обчислювальна система працює наступним чином. У кожний момент часу тільки в одному із тригерів D записана одиниця (під час ініціалізації – у тригері першого в пріоритетному ланцюзі PU). Всі інші тригери встановлені в нуль. Найвищий пріоритет має зовнішній пристрій PU , тригер пріоритету якого встановлений в одиницю. Готовий до обміну інформацією з процесором P будь-який зовнішній пристрій PU формує сигнал запиту переривання (IR). За наявності такого сигналу розривається пріоритетний ланцюжок між входом та виходом $NEXT_PU$ блоку переривань IB . Якщо є хоч один сигнал на виходах IR , формується загальний сигнал вимоги переривань на вході ID процесора. Після закінчення чергової команди процесор у відповідь на сигнал ID формує сигнал підтвердження переривання на виході IA . Цей сигнал потрапляє у пріоритетний ланцюжок, замкнутий у кільце. При цьому початок ланцюжка визначає логічний елемент I , який ви-

дкривається високим рівнем сигналу в тригері збереження пріоритету, як вже було зазначено в такому стані знаходиться тільки один зовнішній пристрій системи. Сигнал IA розповсюджується по ланцюжку до першого на його шляху блока переривань IB , який виставив сигнал запиту переривання IR . В цьому блоці IB формується сигнал на виході $NEXT_PU$. За зрізом сигналу $NEXT_PU$ формується сигнал дозволу видачі вектору переривання, який через загальну шину GB надходить в процесор P . Процесор переходить на виконання програми обслуговування переривання після чого знімає сигнал IA . Після зняття процесором сигналу IA і за фронтом сигналу IA одиниця з виходу $NEXT_TR$ зовнішнього пристрою PU з найвищим пріоритетом переписується у тригер D наступного PU . У цьому випадку пріоритети передаються послідовно від одного зовнішнього пристрою до іншого. Що гарантує обробку переривань від всіх зовнішніх пристроїв на визначеному проміжку часу.

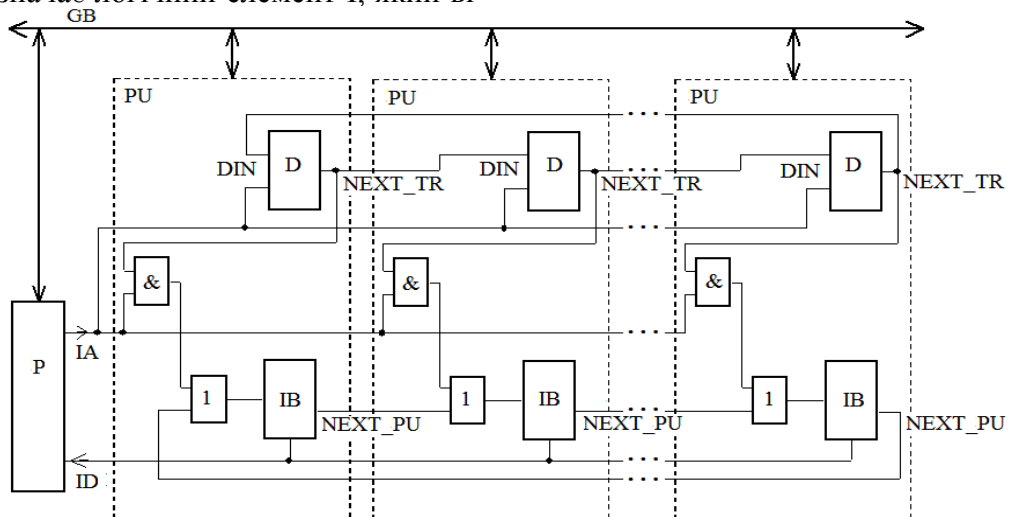


Рис. 2. Обчислювальна система з послідовним передаванням пріоритету

В обчислювальній системі з динамічним передаванням пріоритету (рис. 3) на відміну від обчислювальної системи з послідовним передаванням пріоритету, кожний блок переривань має третій вихід $NEXT_TR$, який приєднаний до інформаційного входу DIN тригера наступного зовнішнього пристрою, при цьому блок переривання останнього зовнішнього пристрою PU підключений до входу першого зовнішнього пристрою PU . Таким чином, блоки переривання IB через елементи АБО об'єднані у кільце. Виходи IR блоків переривання IB об'єднані у єдину лінію і підключені до входу вимоги переривання (ID) процесора P , вихід підтвердження пере-

ривання (IA) якого підведений до керуючих входів тригерів D та других входів елементів I .

Особливістю роботи обчислювальної системи з динамічним передаванням пріоритету є формування сигналу $NEXT_TR$ та вектору переривання на виході блоку IB . Одиниця з виходу $NEXT_TR$ переписується у тригер D наступного PU . Таким чином забезпечується перенос початку пріоритетного ланцюжка. У цьому випадку зовнішній пристрій, що обслуговується, одержує мінімальний пріоритет, а максимальний рівень пріоритету одержує наступний зовнішній пристрій.

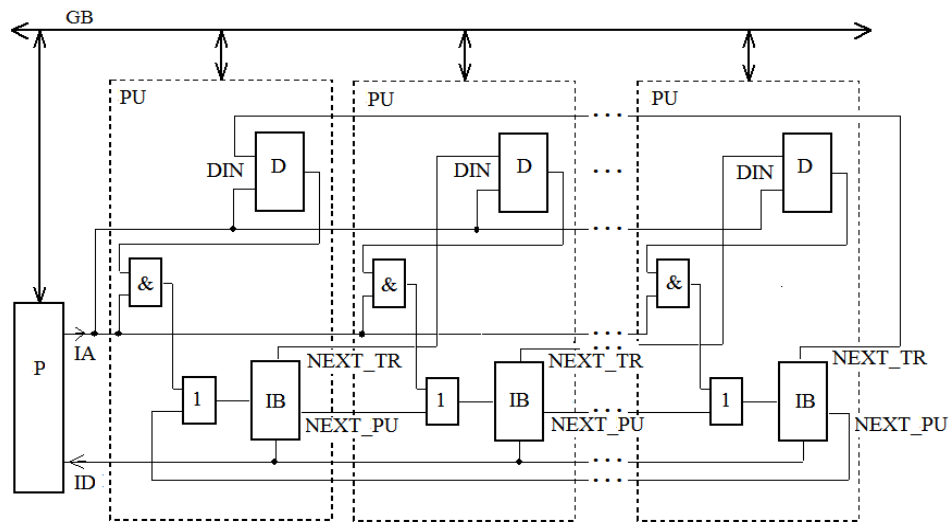


Рис. 3. Обчислювальна система з динамічним передаванням пріоритету

Опис досліджуваних ОС на мові VHDL

Для моделювання роботи обчислювальної системи створений спеціальний «test-bench» рівень коду, який генерує вхідні параметри для проекту. Test-bench рівень представлений у вигляді наступних компонент: *PU_N* – блок, що емулює роботу п'яти зовнішніх пристроїв, *generator* – генератор запитів від зовнішніх пристроїв, що випадковим чином видає сигнали вимоги переривань, *save_request* – компонент для збереження поточного рівня запитів від зовнішніх пристроїв, *proc* – процесор.

Блок розподіленого контролера переривань, що входить у склад кожного зовнішнього пристрою складається з наступних компонент: *d_triger* – тригер *D*, що зберігає максимальний пріоритет; *and1* – елемент І, який управляє початком ланцюжка залежно від стану тригера пріоритету, що надходить по лінії *next_tr*; *or* – елемент АБО, *and2* та *notand2* – блоки переривань відповідно до способу реалізації динамічних пріоритетів. Особливість функціонування системи з послідовним передаванням пріоритету: максимальний пріоритет передається послідовно з виходу тригера зберігання пріоритету поточного ЗП на вхід тригера зберігання пріоритету наступного у пріоритетному ланцюзі ЗП (лінія *next_tr*) і фіксується там за управління сигналу вимоги переривань (*signal_id*), незалежно від того який ЗП був обслуговуваний.

В системі з динамічним передаванням пріоритету в структурі компоненту *PU_N* максимальний пріоритет формується в блоці переривань зовнішнього пристрою, що був обслугований у поточному такту, та за лінією *next_tr* поступає на тригер зберігання

пріоритету наступного у ланцюзі зовнішнього пристрою.

Розподілені контролери переривань синтезовані на мові VHDL.

Результати моделювання

Виконано моделювання розподілених контролерів пріоритетних переривань з послідовним передаванням пріоритету та динамічним передаванням пріоритету в САПР *ModelSim*. В якості результату досліджень наведені часові діаграми роботи контролерів переривань.

Моделювання модулів КПП показало, що за наявності сигналу запити переривання *IR* зовнішній пристрій *PU* отримує сигнал підтвердження переривання *IA*. Якщо на тригері *D* встановлено нульове значення, зовнішній пристрій *PU* переходить у стан очікування за чергою.

Висновки

Застосування розподіленого КПП відповідає модульному принципу організації ОС і ефективно вирішує задачу масштабування обчислювальних систем з відкритою архітектурою, що до кількості зовнішніх пристроїв.

Застосування системи з динамічними пріоритетами гарантує обслуговування переривань від кожного зовнішнього пристрою на визначеному проміжку часу, що надає можливість уникнути тупикових ситуацій та простоїв і підвищити швидкодію системи.

Застосування контролерів з послідовним передаванням пріоритетів ефективно в обчислювальних системах з однорідним циклом функціонування. Тут кожний зовнішній пристрій на протязі $(n-1)$ тактів роботи системи гарантовано отримує обслуговування. В системах що вирішують різного роду задачі

управління в тому числі і траєкторні задачі, цикли управління характеризуються своєю неоднорідністю. Таким чином у визначений момент часу виконується опитування та обробка переривань від певної групи ЗП, що впливають на стратегію управління. Інші ж ЗП знаходяться у пасивному стані. В таких системах застосування КПП з послідовним передаванням пріоритетів буде обумовлювати затримку початку обслуговування переривань, що буде визначатись довжиною пріоритетного ланцюжка. Застосування КПП з динамічним передаванням пріоритету дозволяє підвищити ефективність обробки пе-

реривань в системах з неоднорідним циклом управління і великою кількістю зовнішніх пристроїв за рахунок „обминання” зовнішніх пристроїв, які не приймають участі на даному етапі управління.

Що до перспектив подальшого дослідження планується проведення наступного етапу розроблення розподілених контролерів переривань з динамічними пріоритетами для масштабованих обчислювальних систем у САПР *Quartus II* з урахуванням реальних параметрів мікросхеми, завантаження отриманого пристрою в ПЛІС та визначення реальних характеристик системи.

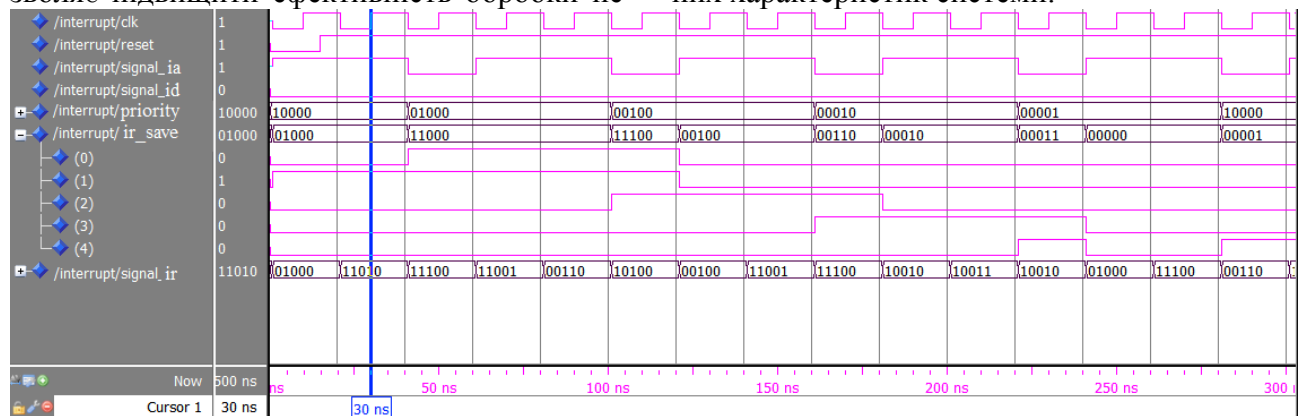


Рис. 4. Часова діаграма обчислювальної системи з послідовним передаванням пріоритету

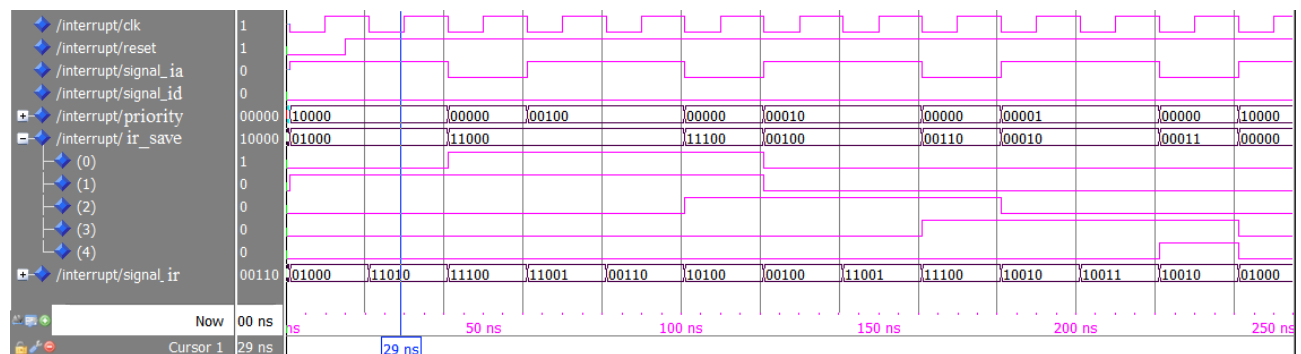


Рис. 5. Часова діаграма обчислювальної системи з динамічним передаванням пріоритету

Список посилань

1. Автореф. дис. д-ра техн. наук: 05.13.13 / В.І. Жабін; Нац. техн. ун-т України "Київ. політехн. ін-т". — К., 2006. — 36 с. — укр.
2. Автореф. дис. канд. техн. наук: 05.13.13 / М.В. Васильев; Харк. нац. ун-т радіоелектрон. — Х., 2003. — 19 с.: рис. — укр.
3. Жабін В.И. Архитектура вычислительных систем реального времени. — К.: ВЕК+, 2003. — 176 с.

Поступила в редакцию 7.12.2009