

УДК 681.5

І.О. ФУРМАН, С.Я. БОВЧАЛЮК

*Харківський національний технічний університет сільського господарства ім. Петра Василенка, Україна*

## HDL-МОДЕЛЬ ПРОГРАМОВАНОГО ЛОГІЧНОГО КЕРУЮЧОГО АВТОМАТА ПАРАЛЕЛЬНОЇ ДІЇ

Представлено структуру і HDL-модель керуючого автомата паралельної дії. Наведено результати моделювання роботи окремих функціональних блоків і усього автомата в цілому

**програмований логічний контролер, паралельна архітектура, математична модель, надійність**

### Аналіз стану питання

Більшість сучасних систем промислової автоматизації будується на основі високонадійних і легко компонуємих програмованих логічних контролерів (ПЛК) і індустріальних комп'ютерів. Основним недоліком більшості відомих ПЛК є використовуємих у них послідовний принцип обробки інформації, що значно знижує ефективність виконання задач керування високопродуктивними агрегатами і установками, вузли і механізми яких повинні функціонувати паралельно синхронно або асинхронно, залежно або незалежно один від одного [1].

У роботі [2] запропоновано математичну модель і архітектуру логічного керуючого автомата паралельної дії, на базі якої було реалізовано та впроваджено у виробництво функціонально завершені програмовані логічні контролери паралельної дії (ПЛК ПД). Їх аналіз, та досвід практичного використання показав, що маючи безумовні переваги перед ПЛК послідовної дії, вони мають і деякі недоліки, які були враховані у роботі [3], в якій також було вказано, що суттєве підвищення якості функціонування керуючих пристроїв і систем може бути досягнуте використанням у якості елементної бази регулярних структур – програмованих логічних інтегральних схем (ПЛІС).

**Постановка задачі дослідження.** Метою статті є

надання опису моделі програмованого логічного керуючого автомата паралельної дії (ПЛКАП) мовою описання апаратури (Hardware Description Language) – HDL-моделі.

### Основні матеріали дослідження

На рис. 1 показано архітектуру ПЛКАП, математична модель і детальний опис роботи якого наведено у [3]. На базі вказаної структури розроблено HDL-модель (рис. 2), для чого були використані спеціальні інструментальні засоби розробки і моделювання, що входять до складу системи проектування MAX+PLUSII.

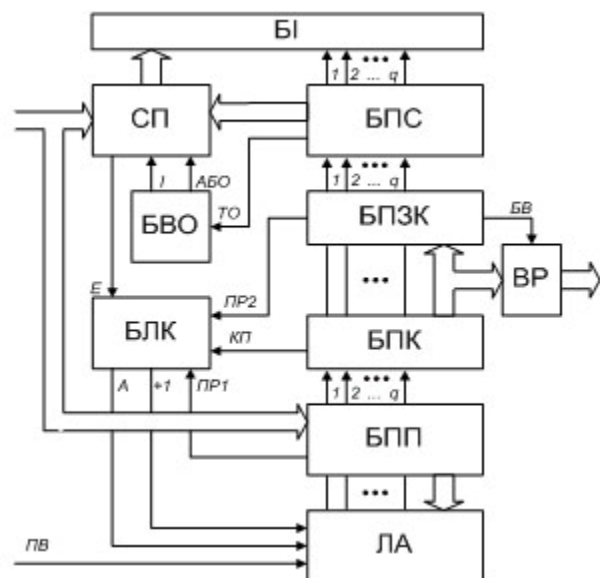


Рис. 1. Архітектура ПЛКАП

Розглянемо складові функціональні блоки і компоненти блоків із наведенням призначення, виконуваних функцій і результатів моделювання, як окремих блоків, так і усього керуючого автомата в цілому.

**Функціональний блок *bps:\_bps*** – блок пам'яті станів, забезпечує зберігання комбінації станів детермінованих входів, яка є єдиною "дозволеною" для включення відповідної кінцевої комбінації механізмів. Містить  $q$ -розрядний вхід адреси  $adr[q..1]$ ,  $k$ -розрядний вихід двобітних даних  $vec\_Ai[1..k][1..2]$  і двійковий вихід  $ot$  (тип операції) на якому формується сигнал логічного «0» або «1» у залежності від типу логічної операції, що має виконуватись схемою порівняння.

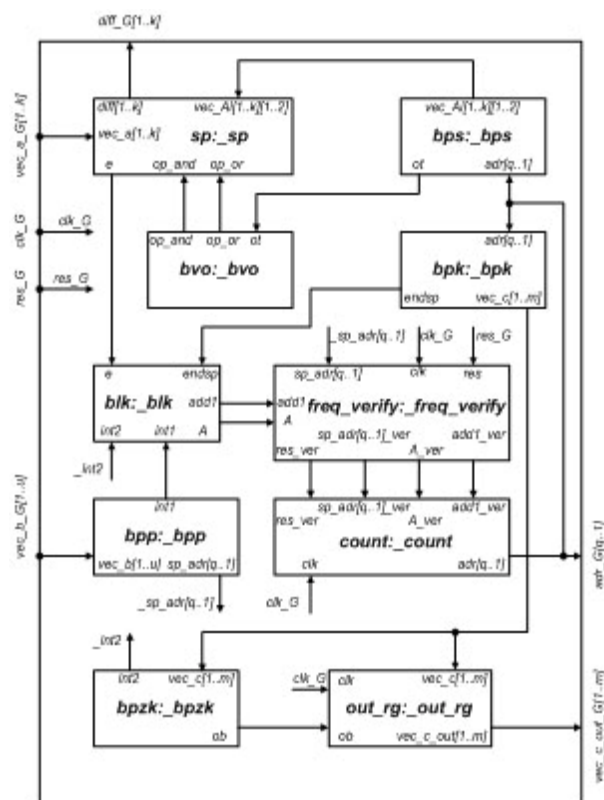


Рис. 2. HDL-модель керуючого автомата

**Функціональний блок *bpk:\_bpk*** – блок пам'яті команд, забезпечує зберігання команд керування технологічним обладнанням. Містить  $q$ -розрядний вхід адреси  $adr[q..1]$ ,  $m$ -розрядний вихід даних  $vec\_c[1..m]$  і двійковий вихід  $endsp$  (кінець підпрограми), що приймає значення логічної «1», при наявності у векторі даних ознаки кінця підпрограми.

явності у векторі даних ознаки кінця підпрограми.

**Функціональний блок *bpb:\_bpb*** – блок пам'яті заборонених комбінацій, забезпечує зберігання заборонених комбінацій команд керування технологічним обладнанням. Містить  $m$ -розрядний вхід даних і виходи  $int2$  (аварійне переривання 2) і  $ob$  (блокування виходу), на яких формуються сигнали керування суміжними блоками при наявності на вході  $vec\_c[1..m]$  забороненої комбінації команд керування.

**Функціональний блок *sp:\_sp*** – забезпечує логічне порівняння сигналів на  $k$ -розрядних входах даних  $vec\_a[1..k]$  і  $vec\_Ai[1..k][1..2]$ , при цьому у залежності від сигналів на входах  $op\_and$  і  $op\_or$  (операція «І», операція «АБО») сигнал логічної «1» на виході  $e$  (еквівалентність) формується або при порозрядному співпадінні сигналів векторів  $vec\_a[1..k]$  і  $vec\_Ai[1..k][1..2]$ , або при співпадінні сигналів хоча б у одному із розрядів вхідних векторів. Сигнали на  $k$ -розрядному виході  $diff[1..k]$  формуються як результат вказаної операції. Результат моделювання роботи блоку показано на рис. 3.

Функціональний блок *sp:\_sp* – забезпечує логічне порівняння сигналів на  $k$ -розрядних входах даних  $vec\_a[1..k]$  і  $vec\_Ai[1..k][1..2]$ , при цьому у залежності від сигналів на входах  $op\_and$  і  $op\_or$  (операція «І», операція «АБО») сигнал логічної «1» на виході  $e$  (еквівалентність) формується або при порозрядному співпадінні сигналів векторів  $vec\_a[1..k]$  і  $vec\_Ai[1..k][1..2]$ , або при співпадінні сигналів хоча б у одному із розрядів вхідних векторів. Сигнали на  $k$ -розрядному виході  $diff[1..k]$  формуються як результат вказаної операції. Результат моделювання роботи блоку показано на рис. 3.

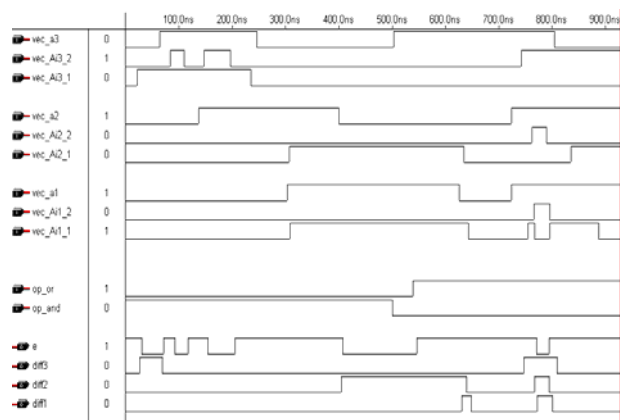


Рис. 3. Результат моделювання роботи блоку *sp:\_sb*

**Функціональний блок *bvo:\_bvo*** – забезпечує формування команд вибору операції для блоку логічного порівняння, містить двійковий вхід *ot* (тип операції) і двійкові виходи *op\_and* (операція «І») і *op\_or* (операція «АБО»).

**Функціональний блок *count:\_count*** – лічильник адреси, забезпечує адресацію блоків пам'яті станів і команд. Містить *q*-розрядні вхід *sp\_adr\_ver[q..1]* (адреса переходу) і вихід *adr[q..1]*, вхід синхронізації *clk*, входи керування *res\_ver* (команда скиду), *A\_ver* (команда паралельного завантаження) і *add1\_ver* (команда додавання одиниці). Результат моделювання роботи блоку показано на рис. 4.

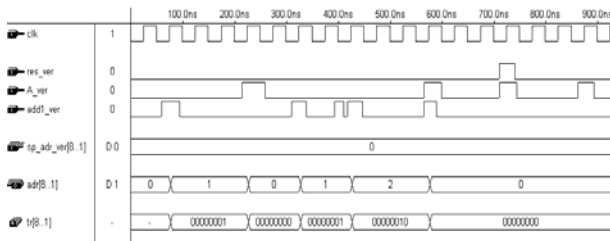


Рис. 4. Результат моделювання роботи блоку *count:\_count*

**Функціональний блок *blk:\_blk*** – блок логічного керування, забезпечує керування лічильником адреси у залежності від сигналів, що сформовані суміжними функціональними блоками логічного автомата. Містить логічні входи *e* (еквівалентність), *endsp* (кінець підпрограми), *int1*, *int2* (аварійне переривання 1 і 2), а також виходи керування лічильником – *A* (паралельне завантаження адреси) і *add1* (додавання одиниці). Результат моделювання роботи блоку показано на рис. 5.

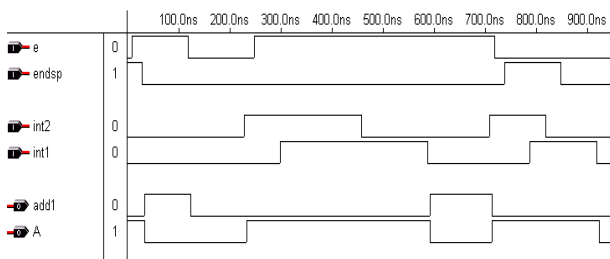


Рис. 5. Результат моделювання роботи блоку *blk:\_blk*

**Функціональний блок *out\_rg:\_out\_rg*** – вихідний регістр, призначений для фіксації і передачі

команд керування від блоку пам'яті команд на виконавчі механізми технологічного агрегату, а також блокування такої передачі у випадку наявності забороненої комбінації цих команд. Містить *m*-розрядний вхід *vec\_c[1..m]* і вихід *vec\_c\_out[1..m]* даних, вхід синхронізації *clk* і вхід блокування виходу *ob*.

**Функціональний блок *freq\_verify:\_freq\_verify*** – забезпечує 8-ми кратну перевірку достовірності сигналів блоку логічного керування, сигналу скиду логічного контролера і адреси підпрограм. Структурна схема блоку показана на рис. 6, а результат моделювання роботи – на рис. 7.

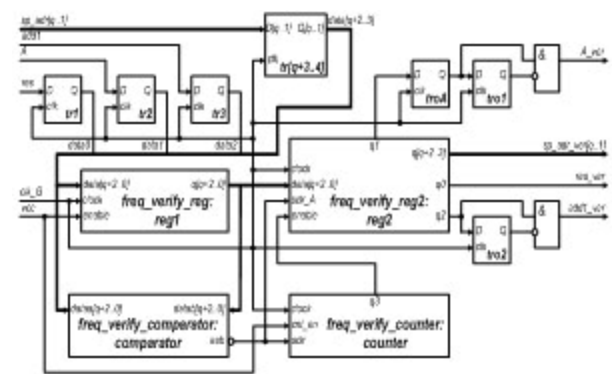


Рис. 6. Структурна схема функціонального блоку *freq\_verify:\_freq\_verify*

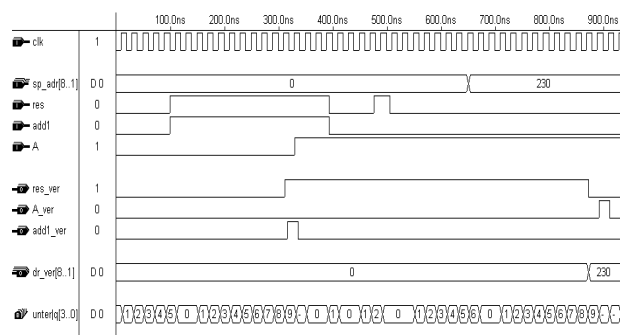


Рис. 7. Результат моделювання роботи блоку *freq\_verify:\_freq\_verify*

Основними складовими компонентами блоку *freq\_verify:\_freq\_verify* є такі.

**Компонент *freq\_verify\_reg:reg1*** –  $(q+3)$ -и розрядний паралельний регістр, який містить входи *clock* (сигнал синхронізації), *enable* (вхід дозволу запису інформації), а також інформаційні входи *data[q+2..0]* і виходи *q[q+2..0]*. При цьому три мо-

лодші розряди регістру використовуються для зберігання сигналів  $res$ ,  $A$ ,  $add1$  (скиду, адреси переходу, додавання одиниці), а  $q$  розрядів для зберігання адреси переходу  $sp\_adr[q..1]$ .

Компонент *freq\_verify\_reg2:reg2* –  $(q+3)$ -и розрядний паралельний регістр, призначений для фіксації і тимчасового зберігання достовірної комбінації сигналів  $res$ ,  $A$ ,  $add1$  і  $sp\_adr[q..1]$ . Конструктивно аналогічний компоненту *freq\_verify\_reg:reg1*, на відміну від якого містить додатковий вхід *sclr\_A* – вхід асинхронного скиду 1-го розряду регістра (розряду команди  $A\_ver$ ). На старших  $q$  розрядах формуються достовірні сигнали  $sp\_adr\_ver[q..1]$ , а з молодшого розряду  $q0$  знімається достовірна команда  $res\_ver$  – скиду контролера.

Компонент *freq\_verify\_comparator:comparator* – схема порівняння, забезпечує фіксацію зміни станів вхідних сигналів. Містить інформаційні входи  $dataa[q+2..0]$ ,  $datab[q+2..0]$  і один інверсний вихід  $aeb$ , на якому формується сигнал логічної одиниці при нееквівалентності сигналів  $dataa[q+2..0]$  і  $datab[q+2..0]$ .

Компонент *freq\_verify\_counter:counter* – 4-х розрядний лічильник, який містить входи *clock* (сигнал синхронізації), *cnt\_en* (вхід дозволу роботи), *sclr* (вхід синхронного скиду лічильника), а також вихід  $q3$ , на якому формується сигнал логічної «1» у випадку, якщо протягом певного інтервалу часу вхідні сигнали  $data[q+2..0]$  були незмінними.

Компонент *tr (tr1-tr[q+3])* – синхронні D-тригери призначені для синхронізації подання сигналів  $res$ ,  $A$ ,  $add1$  і  $sp\_adr [q..1]$  із глобальним сигналом  $clk\_G$ . Містить входи даних  $D$  і синхронізації  $clk$ , а також вихід  $Q$ .

Компонент *tro (tro1, tro2)* – синхронні D-тригери, які сумісно з логічними елементами «І» формують достовірні команди  $ver\_A$  і  $ver\_add1$  довжиною в один період глобального сигналу синхронізації  $clk\_G$ . Побудова компонентів *tro* аналогічна компонентам *tr*.

Компонент *troA* – синхронний D-тригер, конструктивно аналогічний компоненту *tr* і призначений

для затримки видачі команди  $A\_ver$  на один період глобального сигналу синхронізації  $clk\_G$ .

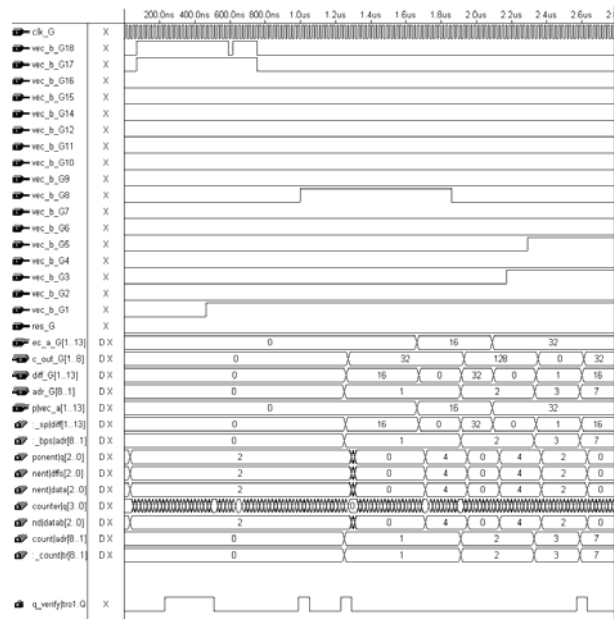


Рис. 8. Результат моделювання роботи ПЛКАП

## Висновки

Розроблена HDL-модель є основою для практичної побудови функціонально закінчених ПЛК паралельної дії нового покоління – паралельних ПЛІС-контролерів.

## Література

1. Фурман І.О. Програмовані логічні контролери: Підручник для ВНЗ / М-во освіти і науки України. – К.: МОНУ, 2003. – 214 с.
2. Фурман И.А. Научно-технические основы создания и промышленного применения параллельных логических контроллеров на программируемых БИС с матричной структурой: Дис. ...докт. техн. наук: 05.13.05. –К., 1989. –197 с.
3. Фурман И.А., Бовчалюк С.Я. Совершенствование математической модели и архитектуры логических управляющих автоматов параллельного действия // ИКСЗТ. – 2006. – №3(59). – С. 72-76.

Надійшла до редакції 6.02.2007

**Рецензент:** д-р техн. наук, проф. В.А. Краснобаев, Харківський національний технічний університет сільського господарства ім. Петра Василенка, Харків.