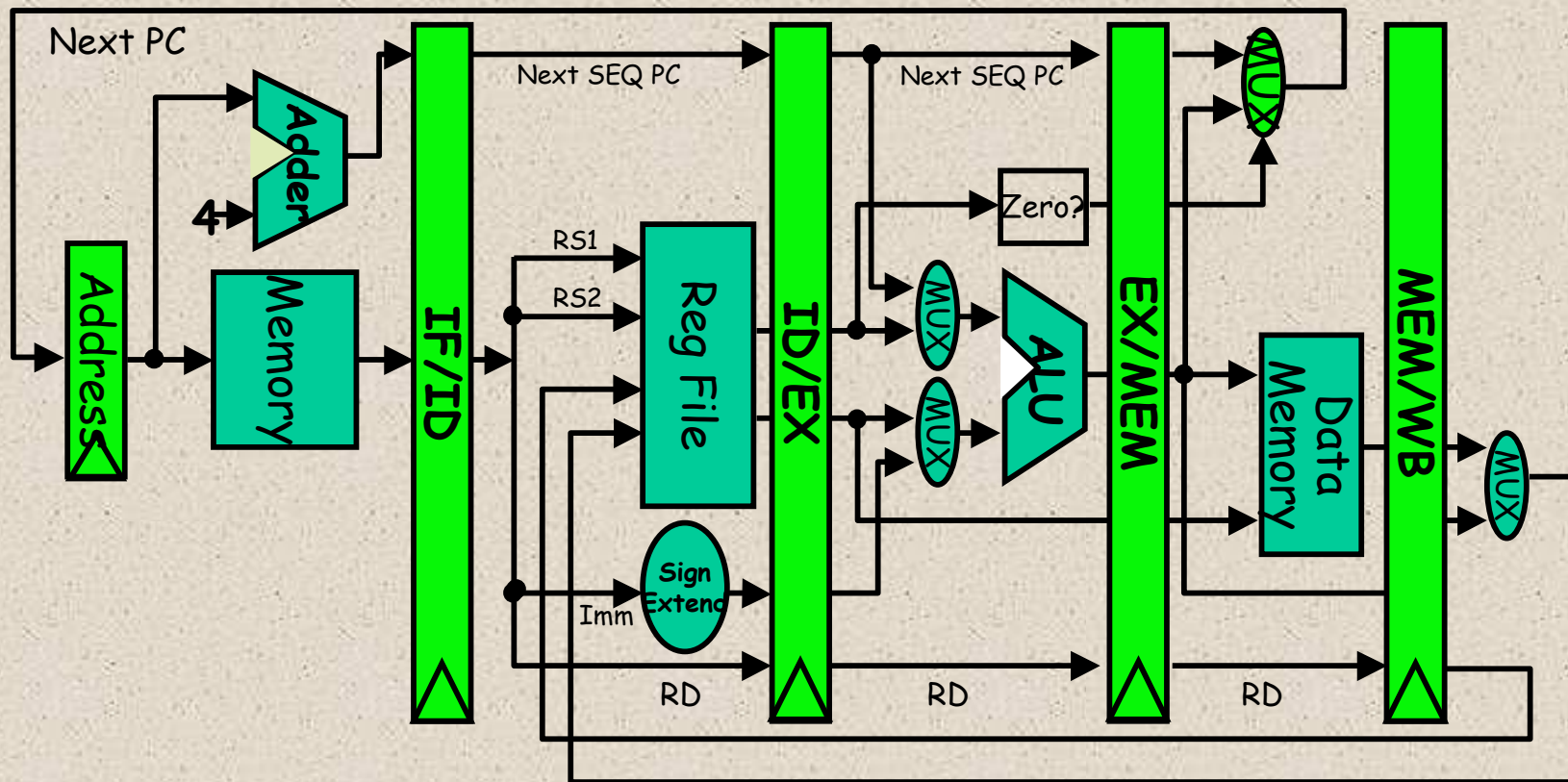
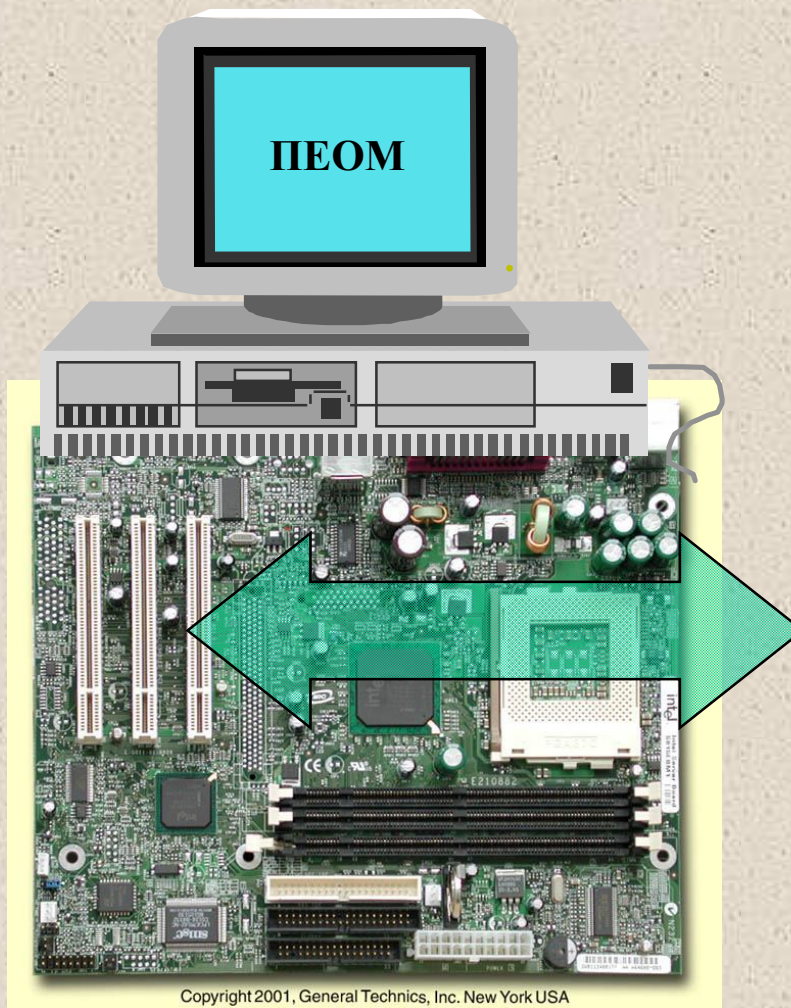


Періодичні алгоритми і конвейейерні обчислювачі

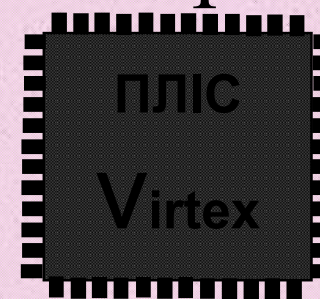


Прискорення обчислень за допомогою обробки даних на ПЛІС



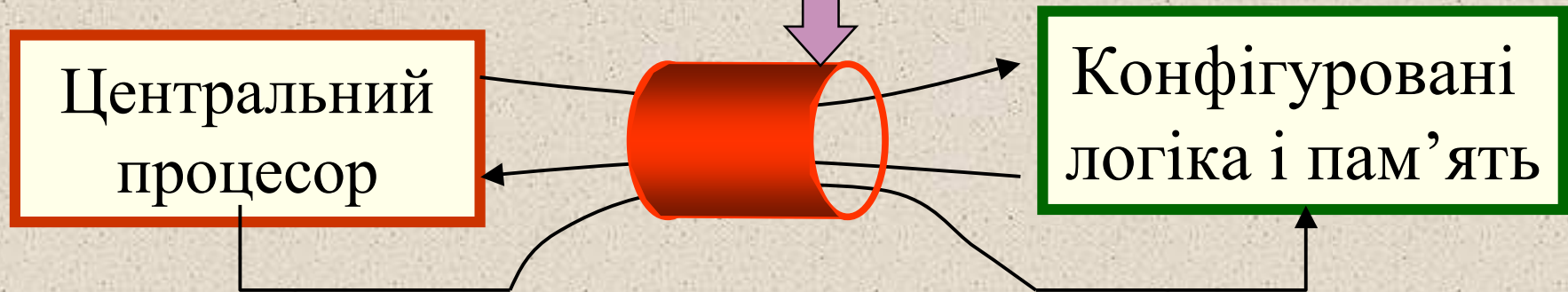
PCI interface

Плата апаратного
прискорювача



Прискорення обчислень за допомогою обробки даних на ПЛІС

Вузьке місце конфігурованих комп'ютерів
на ПЛІС



Час обчислень повинен переважати час обміну даних

Алгоритми, що реалізуються ефективно:

-мають складність $O(N^2)$ і більше, де N – характерна розмірність завдання. Наприклад:

-*Цифрова обробка сигналів*

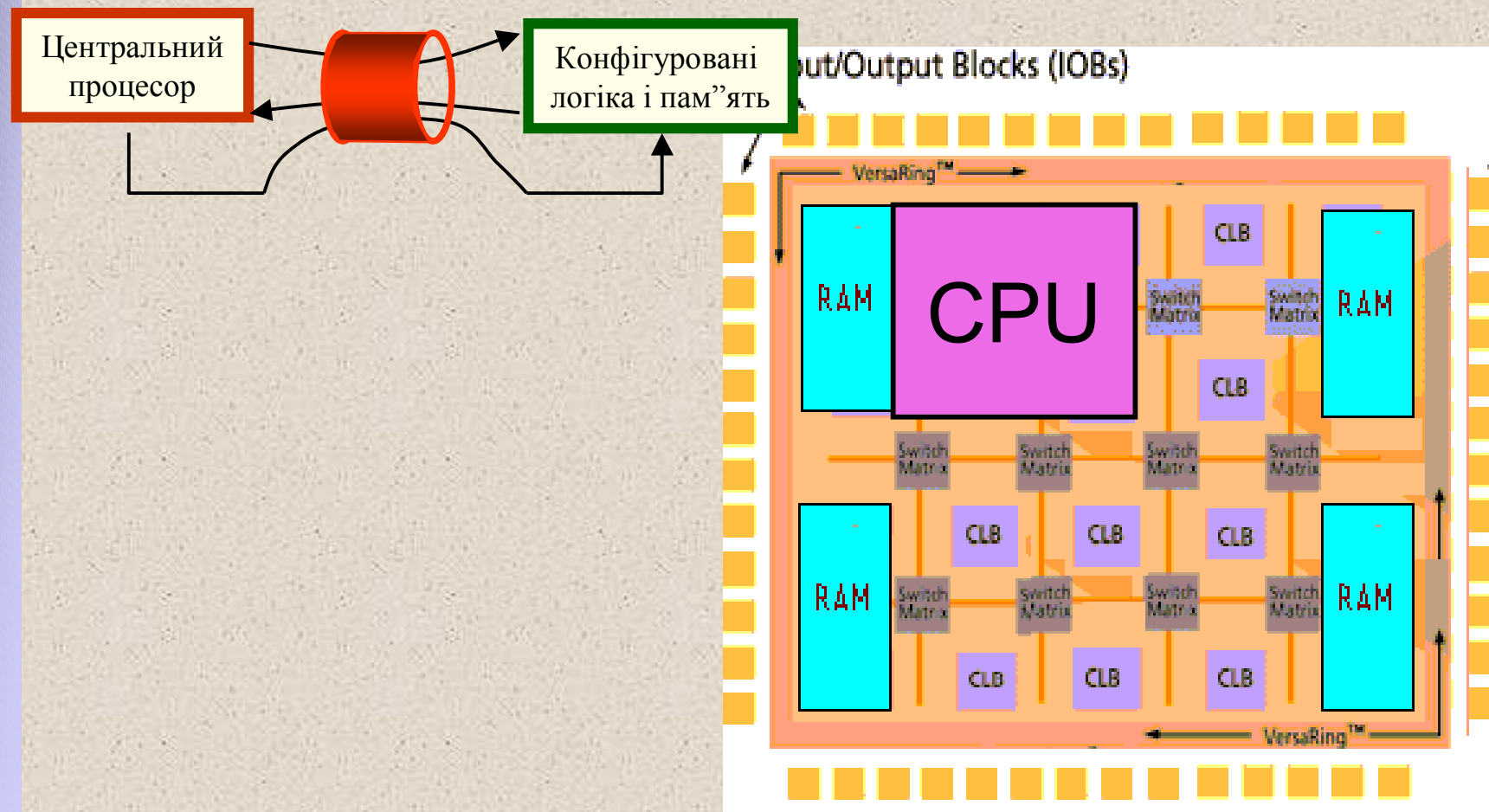
-*Компресія зображень*

-*Шифрування даних*

-*Сортування і пошук*

Прискорення обчислень за допомогою обробки даних на ПЛІС

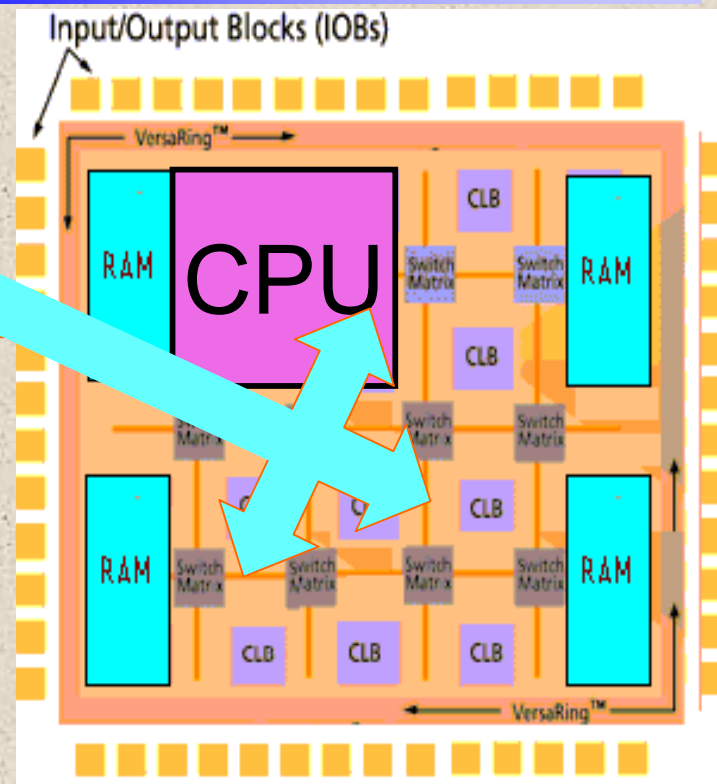
Перспективна архітектура : ПЛІС + Мікропроцесорне ядро



Прискорення обчислень за допомогою обробки даних на ПЛІС

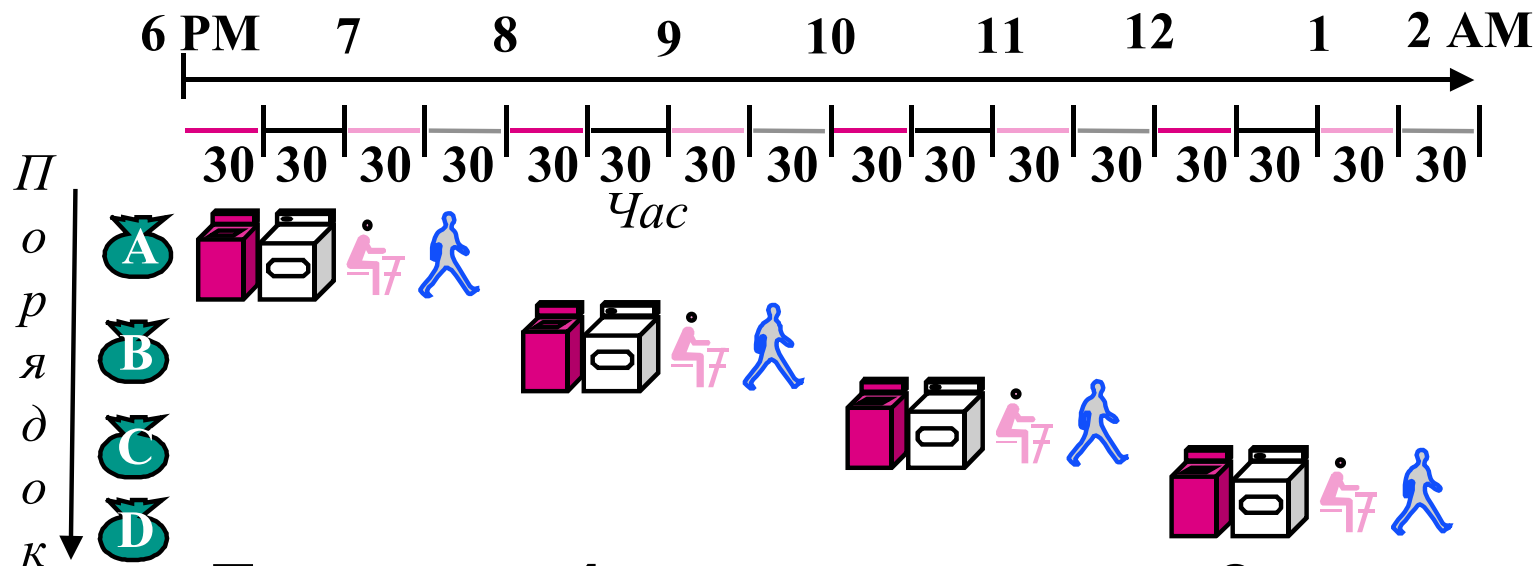
АЛГОРИТМ

Питання:
як відобразити
алгоритм
в обмежені
апаратні ресурси ПЛІС,
щоб одержати
максимальну швидкодію
(пропускну спроможність)?



Послідовна обробка даних

Послідовна пральня



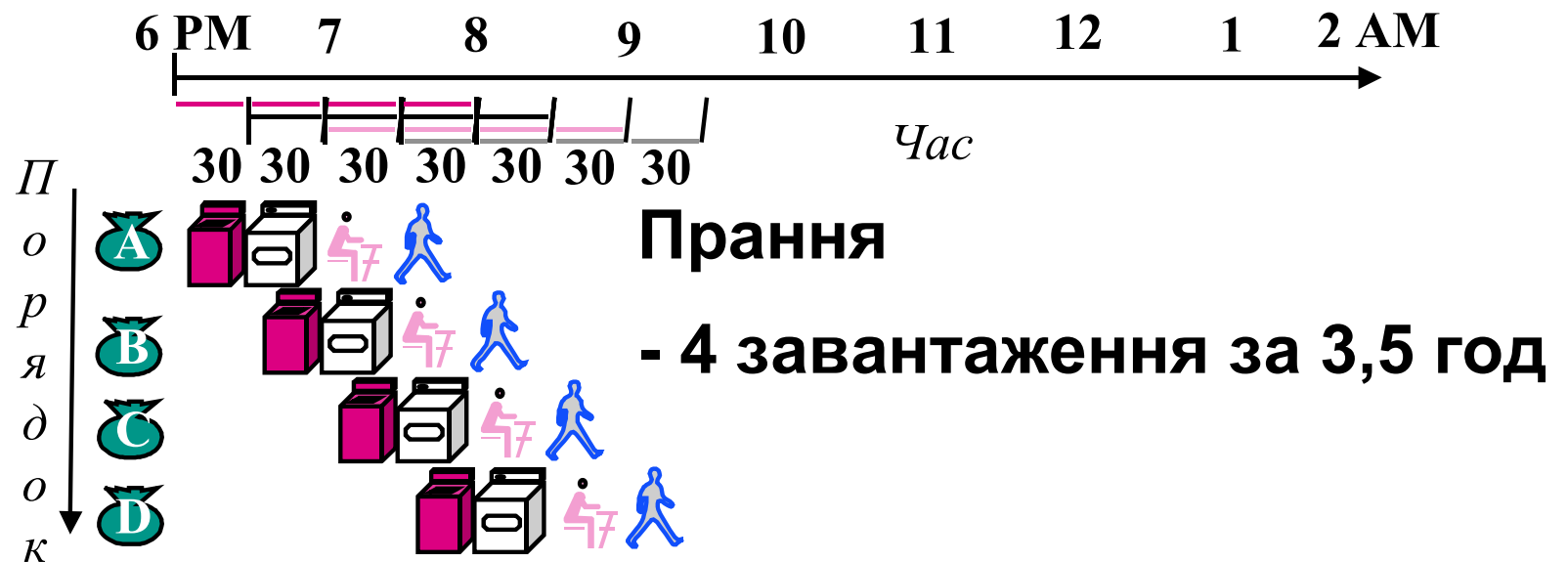
Прання - 4 завантаження за 8 годин

Виконання

Доки все обладнання не звільниться
від попереднього завдання,
доти не можна почати наступне завдання

Паралельна конвейерна обробка даних

Конвейерна пральня



Вхідні дані – однорідні і їх багато.

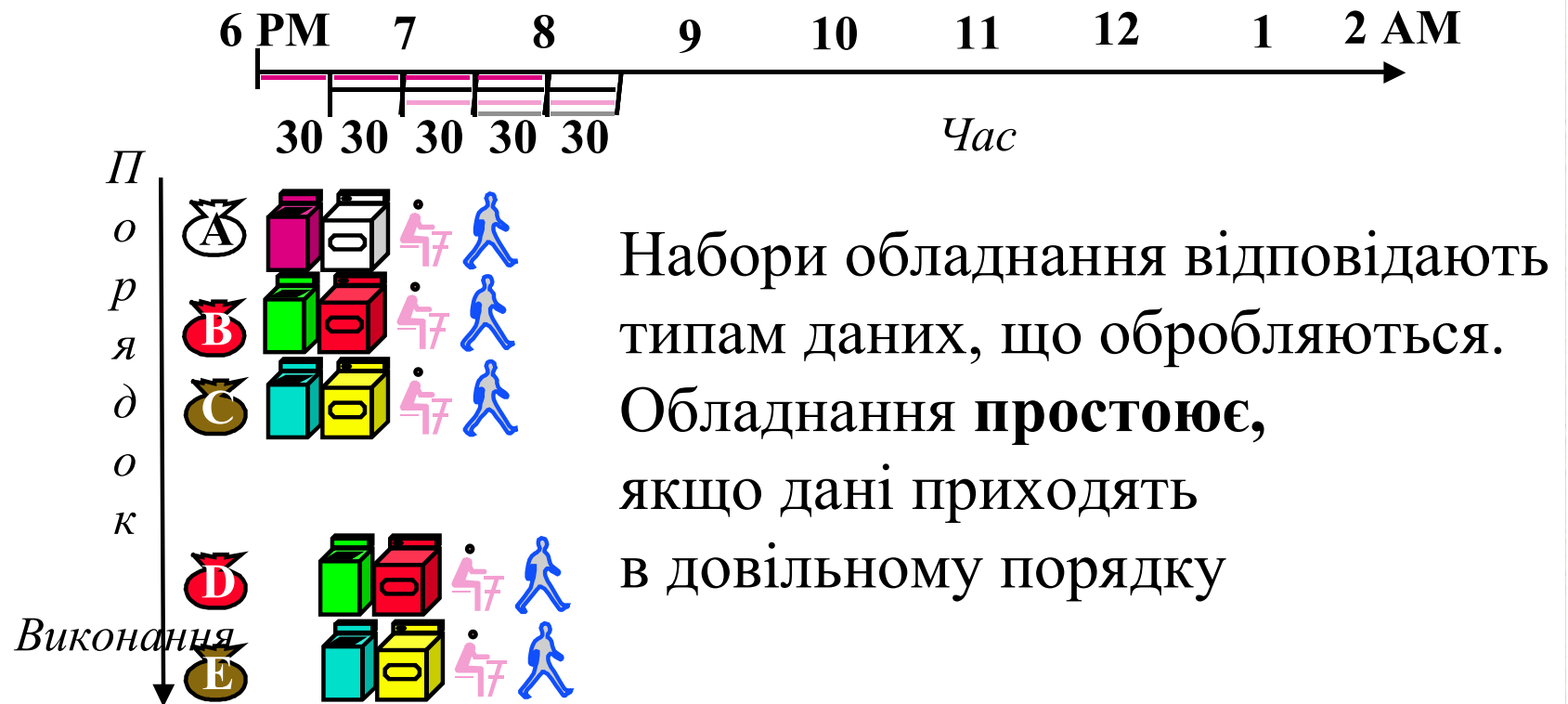
Виконання Кожна операція має однакову тривалість.

Обладнання ступенів різнорідне.

і пристосоване для відповідної операції.

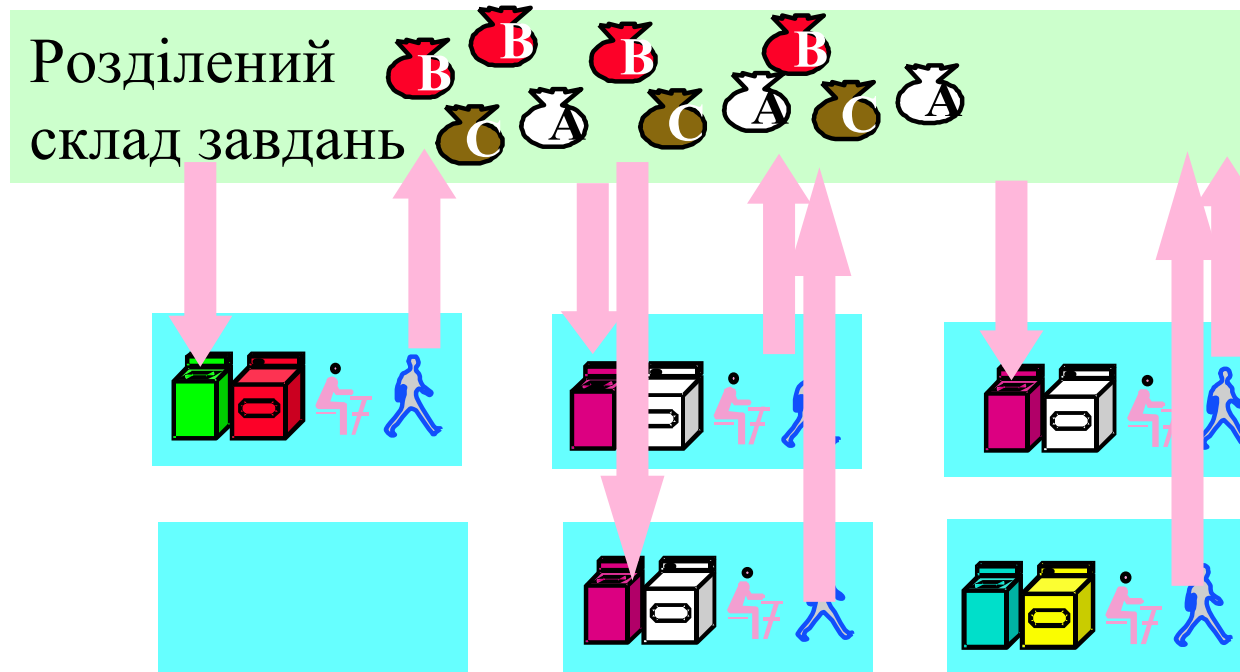
Паралельна суперскалярна обробка даних

Суперскалярна пральня



Повністю паралельна обробка даних

Мультипроцесорна пральня

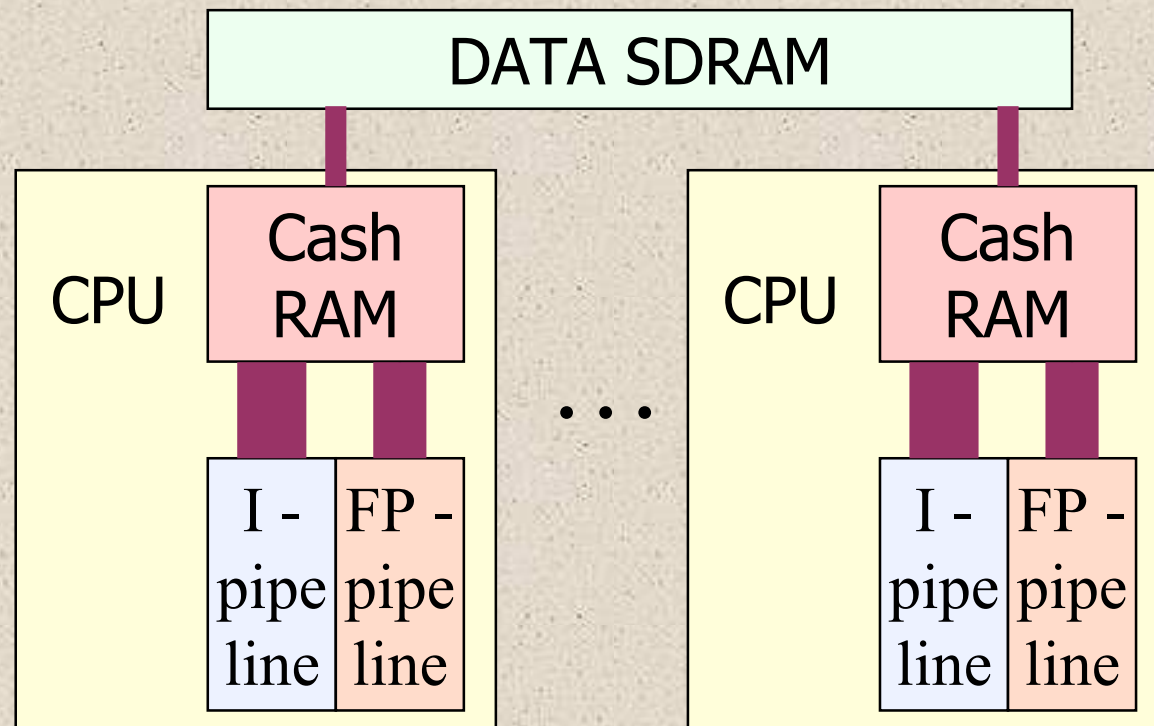


Всі **процесори** повинні бути **універсальними**.

Завдань повинно бути **багато**, щоб завантажити процесори.

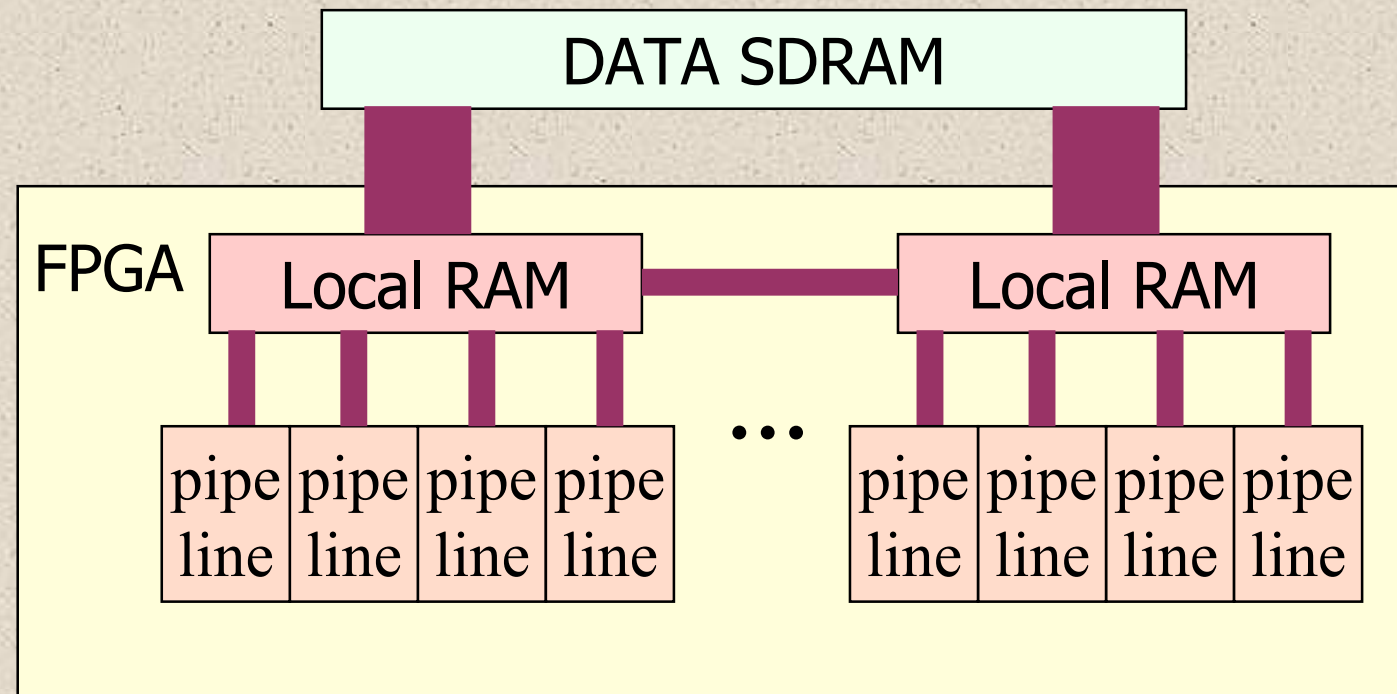
Завдання повинні бути **незалежними і довготривалими**, щоб не перевантажити канали обміну і не було конфліктів.

Система на базі суперскалярних мікропроцесорів



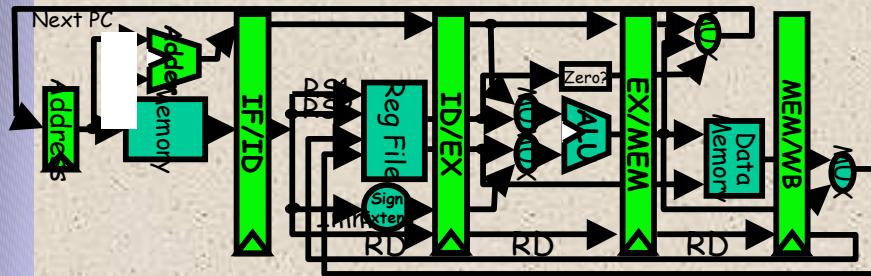
Продуктивність обмежена пропускною здатністю шин,
зернистістю алгоритму
і малою ефективністю завантаження конвейерів CPU

Система на базі ПЛІС



- ▲ Пропускна здатність шин на порядок більше, ніж у суперскалярних процесорів.
- ▲ Заповнення конвейерів - оптимальне.
- ▲ операції конвейера - адаптовані до алгоритму

Конвейерна обробка даних

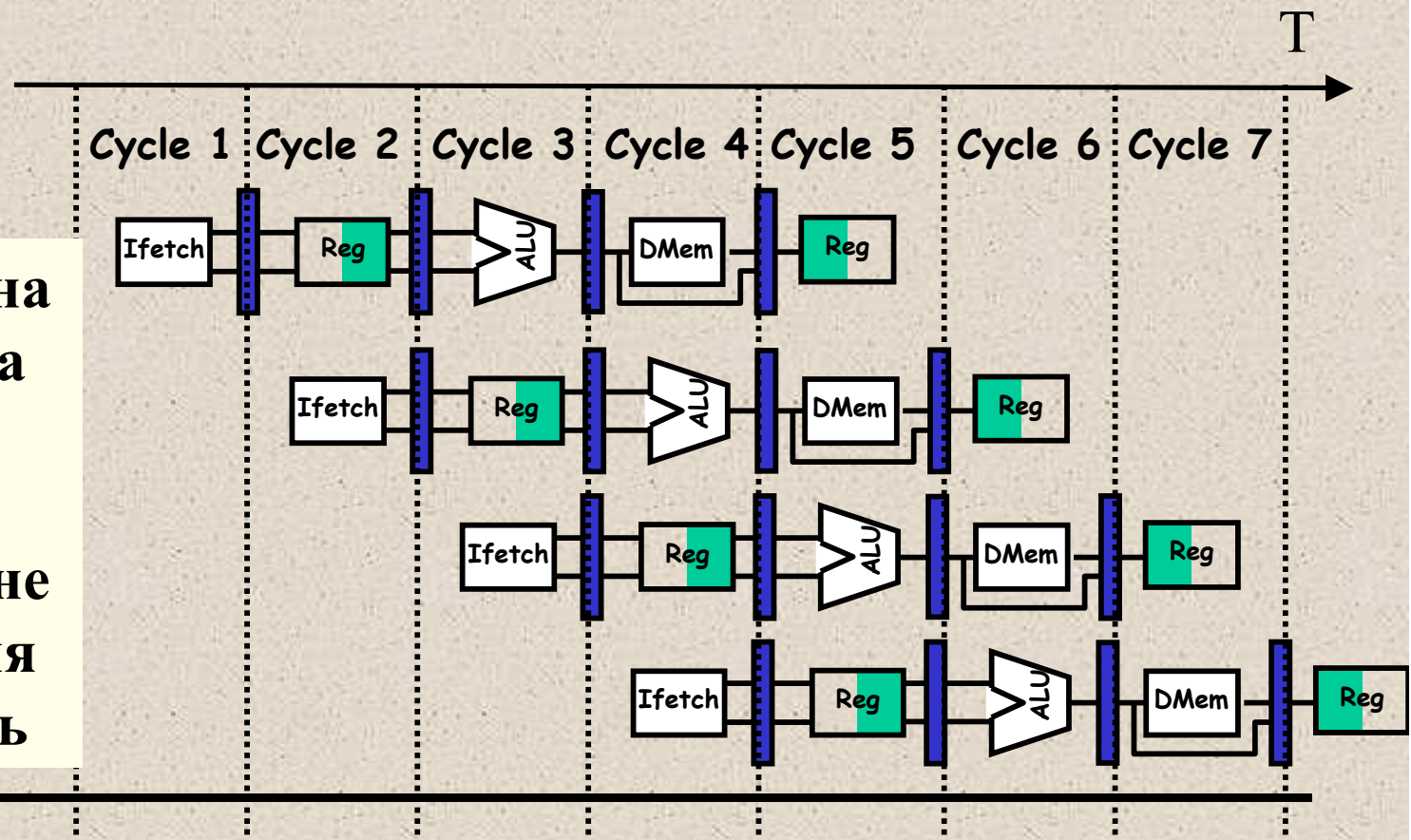


Мінімальні апаратні витрати.
Мінімальний тактовий період,
що дорівнює максимальній
затримці ступеня конвейера

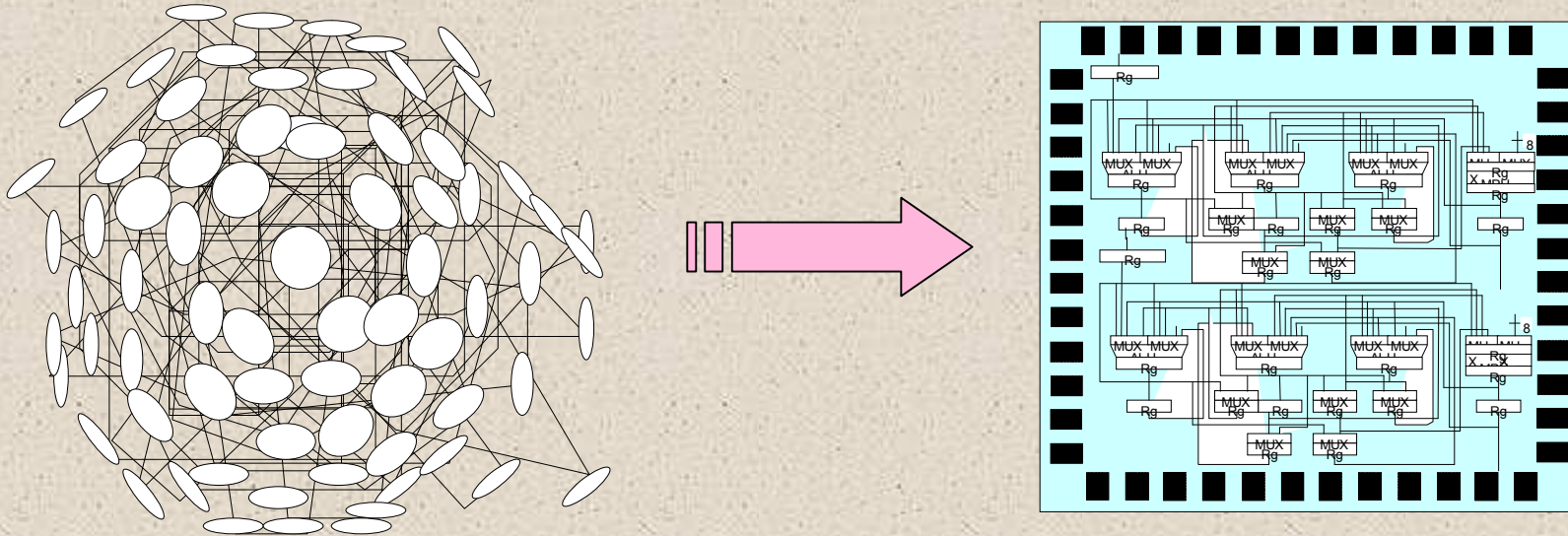
Конвейерна
структура

i

Конвейерне
виконання
обчислень



Відображення алгоритму

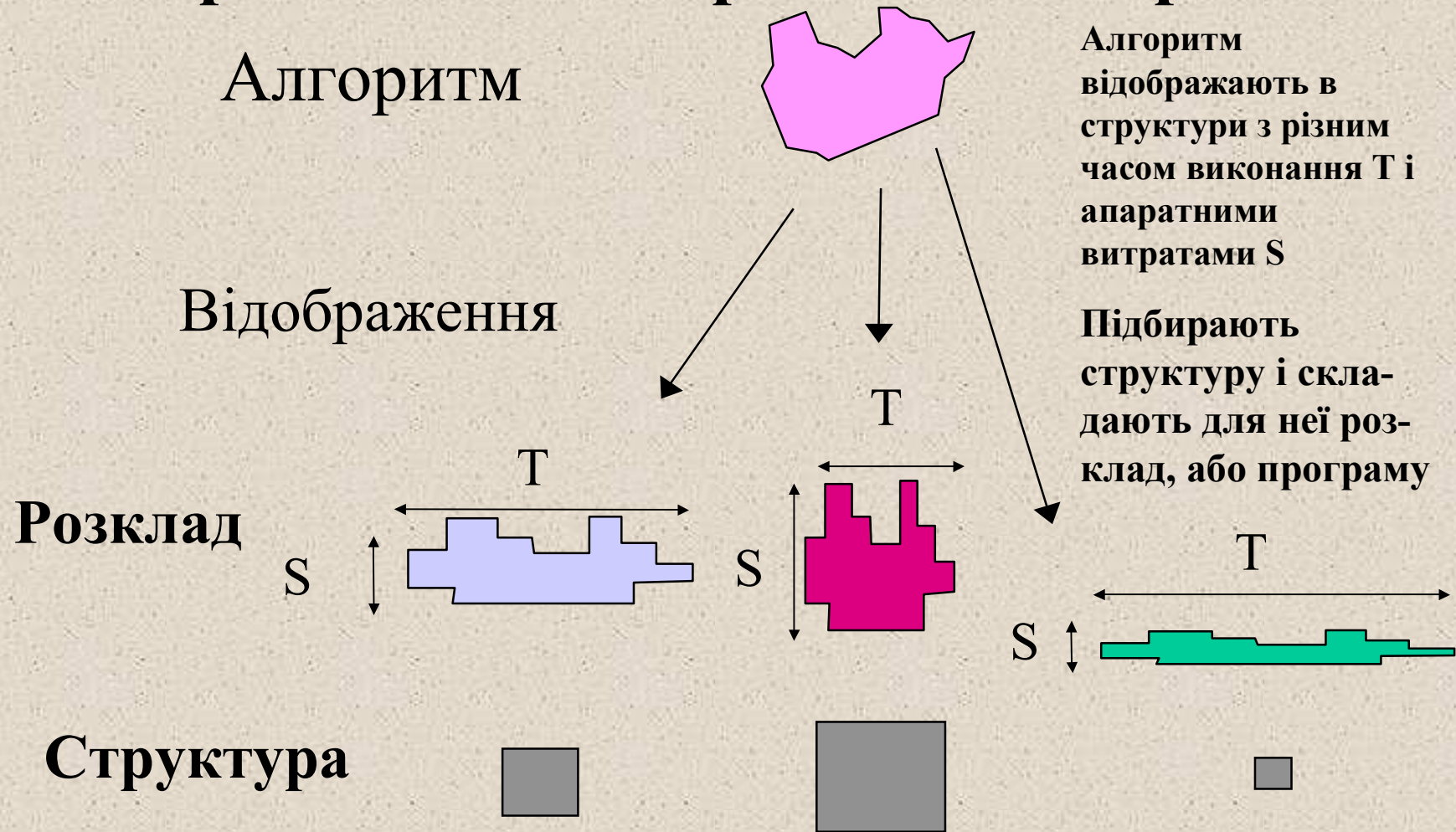


Як заставити апаратні ресурси виконувати оператори алгоритму

- **в потрібному порядку**
- **з вчасною передачею відповідних даних**
- **з мінімальними простоями обладнання і максимальною швидкістю ?**

Відображення алгоритму

Традиційне відображення алгоритмів

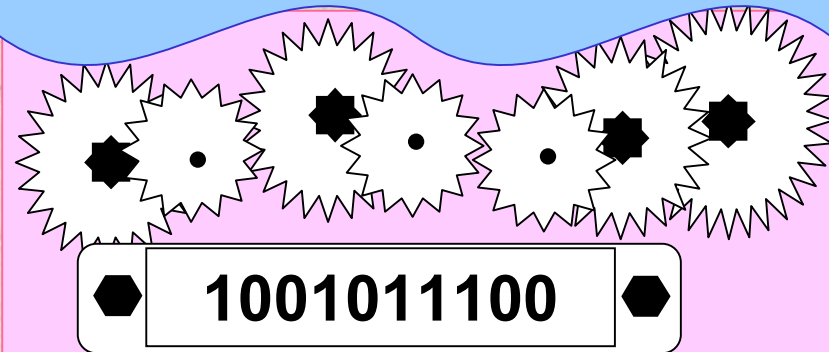


Періодичні алгоритми

Обчислювальний
алгоритм

=

Обчислювальні
процеси



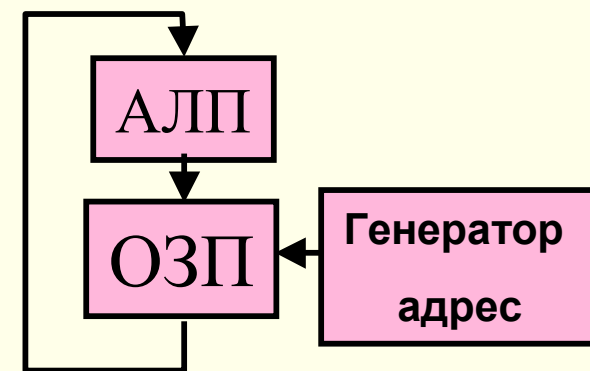
Модель обчислювача

Періодичні алгоритми

Алгоритм, що заданий циклом

```
for  $i$  in 1 to  $U$  loop  --  $i$  - індекс; 1, $U$  - границі циклу  
  st1:  $a := X(i) + X(i-1)$ ; --  $X(i)$  - вхідна змінна  
  st2:  $b := Y(i-1) + Y(i-2)$ ; --  $Y(i)$  - вихідна змінна, яка також  
  st3:  $Y(i) = a + b$ ;    -- виражає міжітераційну залежність  
end loop;
```

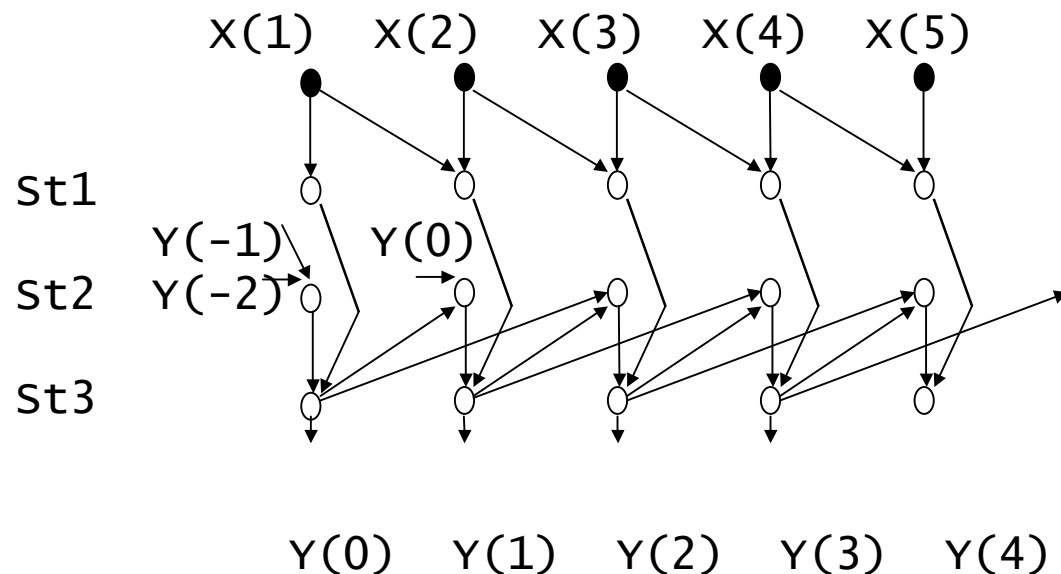
Програмний цикл передбачає зберігання змінних з індексами в **ОЗП** у вигляді масивів і їхню послідовну обробку в **одному АЛП**



Періодичні алгоритми

Граф періодичного алгоритму

```
for i in 1 to U loop  
  st1: a := X(i) + X(i-1);  
  st2: b := Y(i-1)+Y(i-2);  
  st3: Y(i) = a+b;  
end loop;
```



Кожна вершина – оператор (st1,st2,st3).

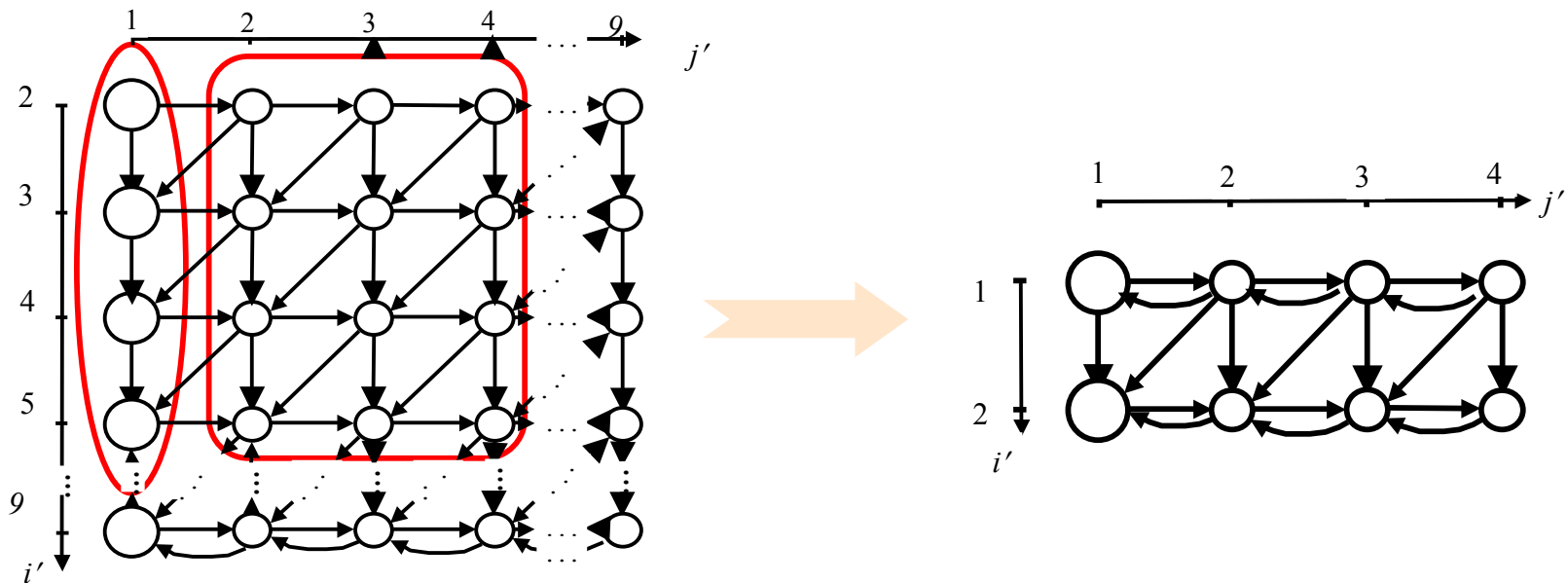
Дані передаються по дугам негайно.

Оператор спрацьовує зразу, як тільки на вході є дані.

Вхідні дані задані 1 раз.

Періодичні алгоритми

Відображення графа періодичного алгоритму



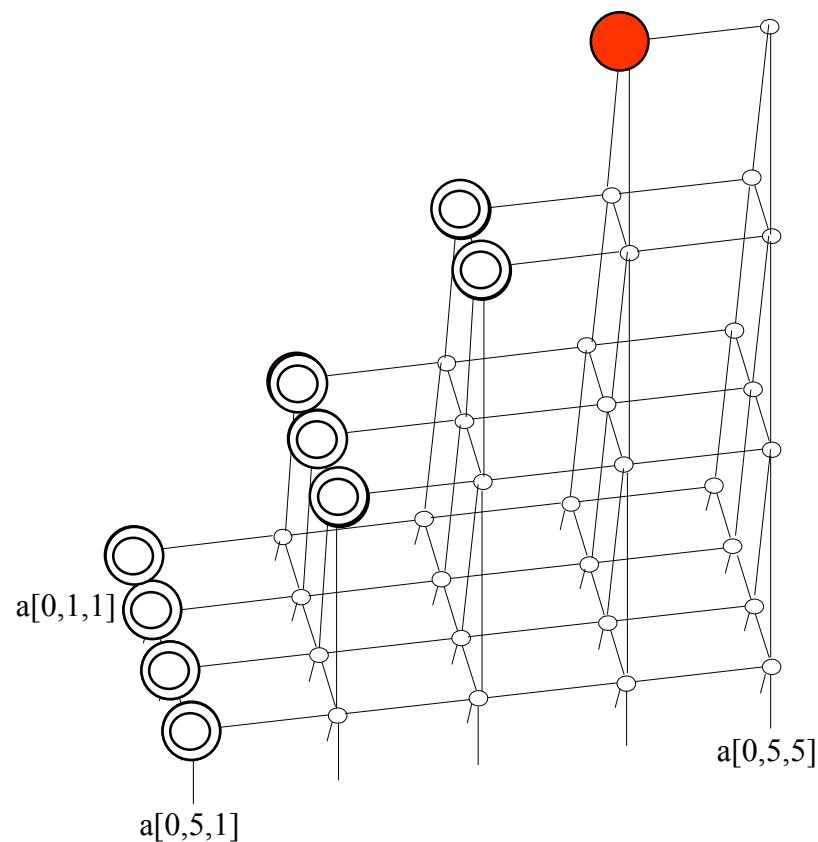
Одержати граф структури обчислювача можна **зклеюванням** вершин графа алгоритму.

В графі структури вершина ПЕ виконує зклеєні оператори, а дуги пересилають проміжні дані.

Періодичні алгоритми

Граф періодичного алгоритму

Граф QR-декомпозиції
Методом Гівенса



Періодичні алгоритми

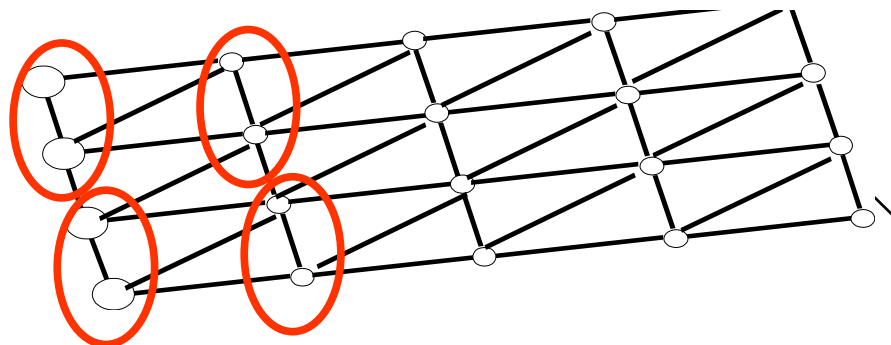
Відображення графу періодичного алгоритму

Algorithm graph

Twiddle factor calculator
○

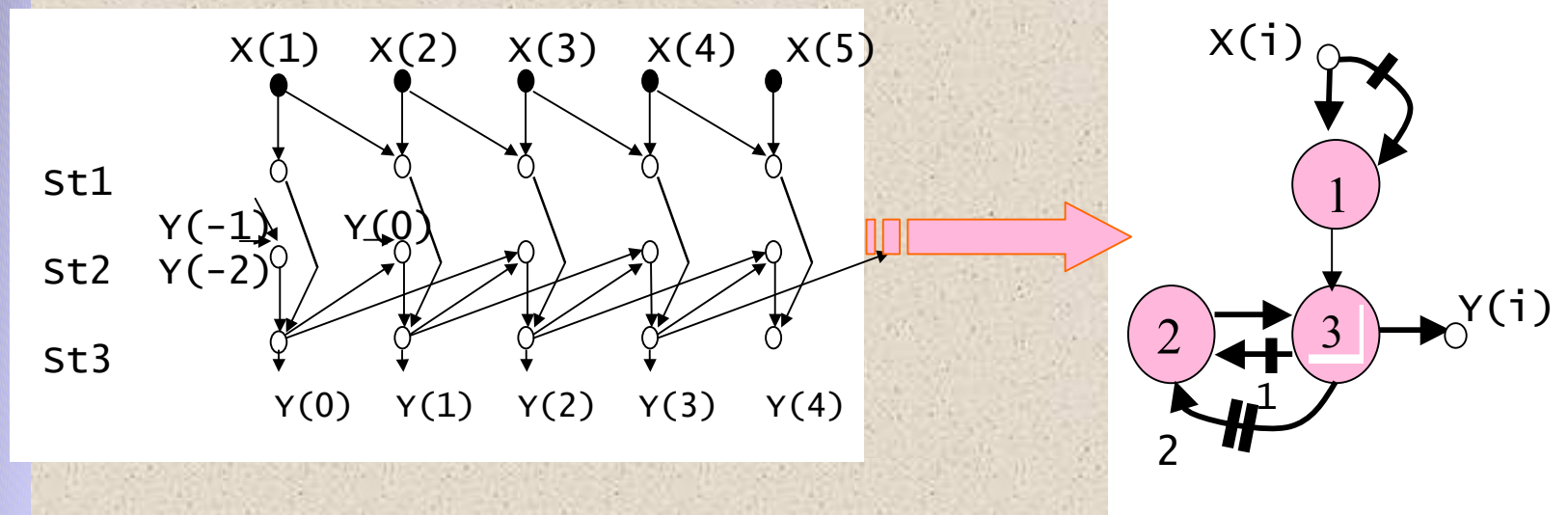
○ Rotator node

Node
gluing



Періодичні алгоритми

Граф синхронних потоків даних (редукований граф алгоритму, SDF)

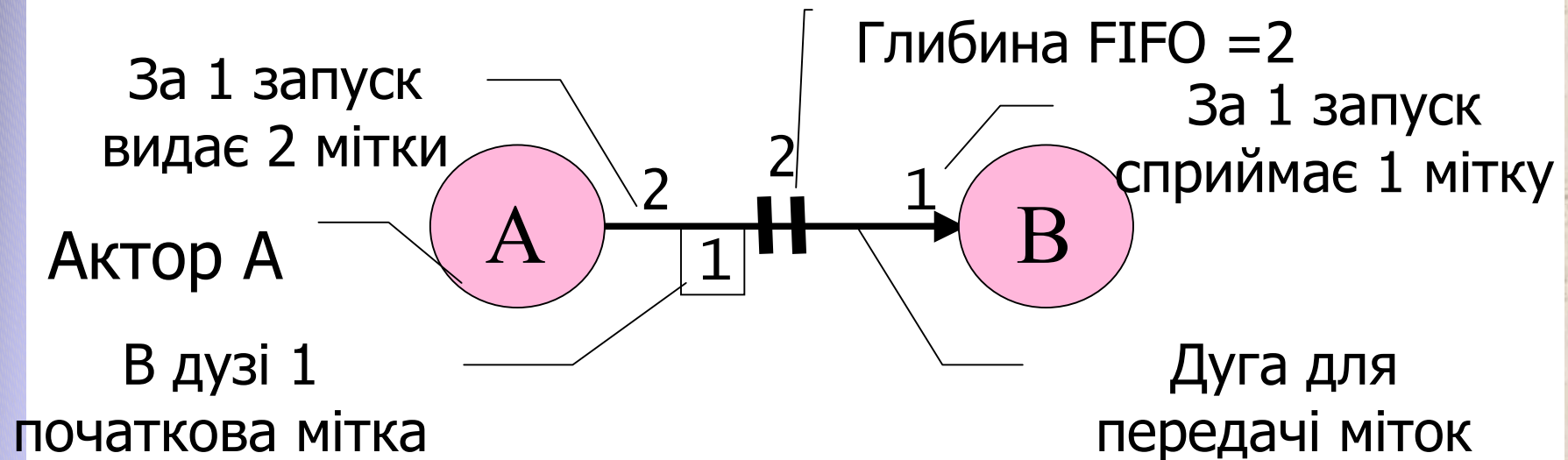


Вся інформація про граф періодичного алгоритму, крім граничних обчислень, вміщується в його періоді. даних

Періодичні алгоритми

Граф синхронних потоків даних (SDF):

Неоднорідний (multirate)



Можливий розклад виконання алгоритму:

Варіант 1: В А В В А В В ...

Варіант 2: А В В А В В А В В ...

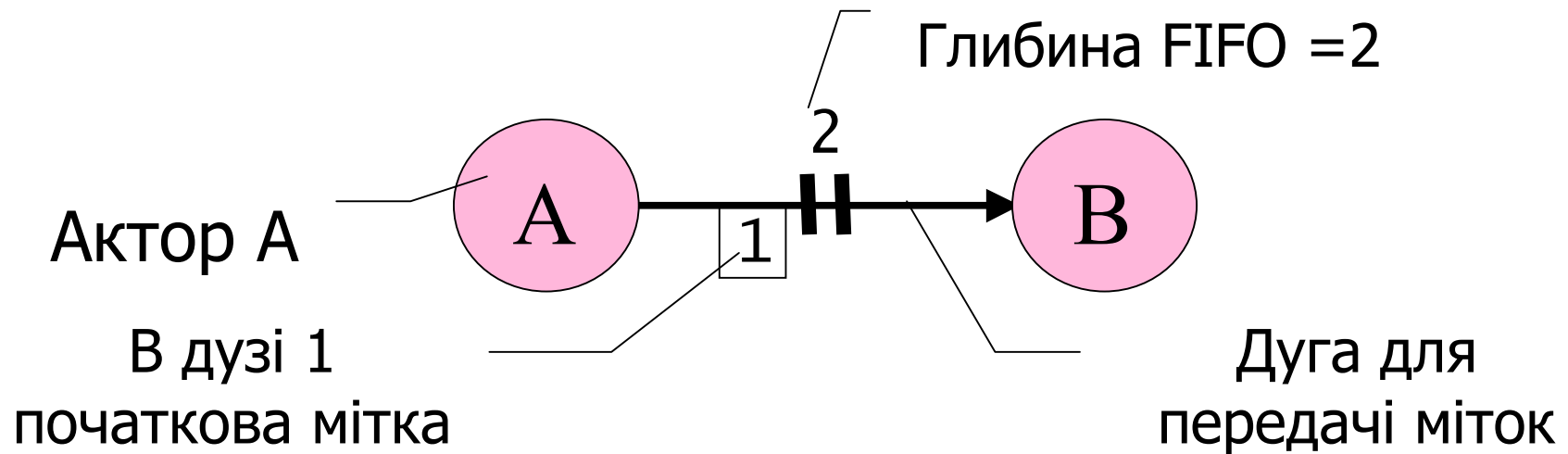
Період алгоритму

Періодичні алгоритми

Граф синхронних потоків даних (SDF):

однорідний (homogeneous),

Редукований граф алгоритму (РГА)

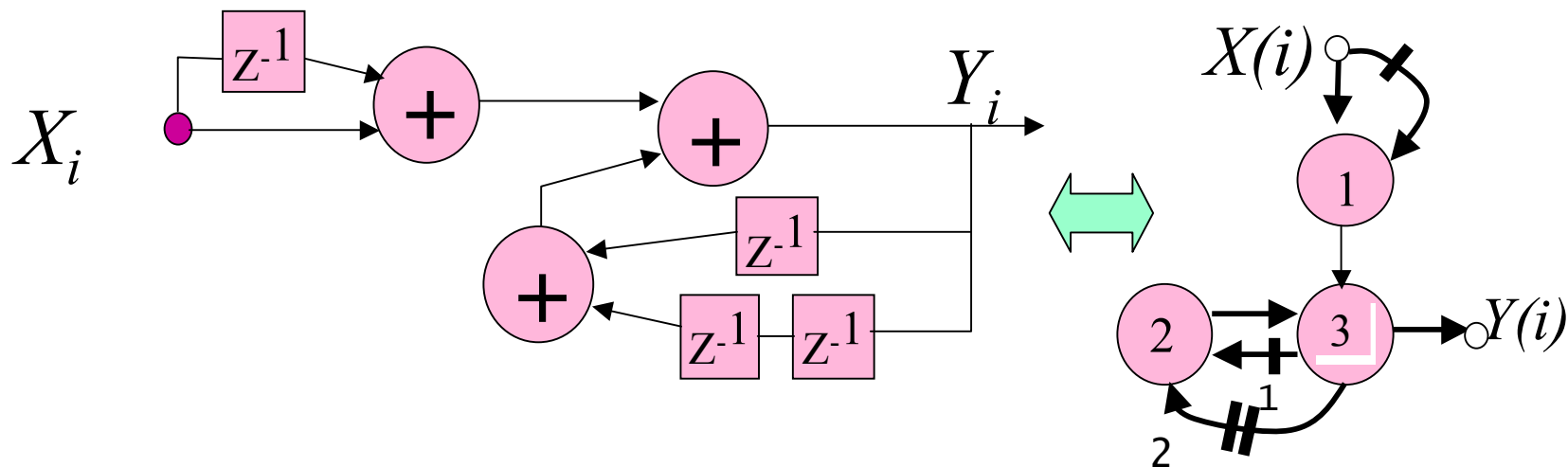


За 1 запуск кожен актор видає по 1 мітці

Періодичні алгоритми

Сигнальний граф

Передаточна функція $H(Z) = (1+Z^{-1})/(1-Z^{-1}-Z^{-2})$,
Відповідає різницевому рівнянню $Y_i = X_i + X_{i-1} + Y_{i-1} + Y_{i-2}$,



В сигнальному графі оператору Z^{-k}
відповідає затримка на k ітерацій.

Періодичні алгоритми

Класифікація графів потоків даних

