

# ***“Архітектура обчислювальних систем та комп’ютерна схемотехніка”.***

## ***Розділ 1. Теоретичні основи функціонування цифрових обчислювальних машин.***

---

### ***Тема лекції: Головні елементи та вузли ЕОМ.***

#### ***План лекції***

- 1.Регістри***
- 2.Лічильники***

## **Рекомендована література до теми**

1. Злобін Г.Г, Рикалюк Р.Є. Архітектура та апаратне забезпечення ПЕОМ: Навч.посіб. –К., 2006, 2012.
2. Воробйова О. М., Іванченко В. Д. Основи схемотехніки: У двох частинах: [Навчальний посібник](#). — [Одеса](#): [ОНАЗ ім. О. С. Попова](#). — 2004, Ч. 2. — 172с.: іл.
3. Шеннон К. Работы по теории информации и кибернетике / Пер. с англ.— М., 1963.
4. Хэмминг Р.В. Теория кодирования и теория информации / Пер. с англ. — М., 1983.
5. Биркгоф Г., Барти Т. Современная прикладная алгебра / Пер.с англ. —М.; 1976.
6. Ланцов А.Л., Зворыкин Л.Н., Осипов И.Ф. Цифровые устройства на комплементарных МПД интегральных микросхемах. —М., 1983.
7. Горбунов В.Л., Панфилов Д.И., Преснухин Д.Л. Справочное пособие по микропроцессорам и микроЭВМ. —М., 1988.
8. Каган Б.М. Электронные вычислительные машины и системы: Учеб.пособие для вузов.—М., 1991.

# *Терміни і позначення:*

- **Регістром (від англ. register – журнал запису)** називається операційний вузол комп'ютера, що служить для запам'ятовування та перетворення слів і забезпечує в загальному випадку виконання наступних мікрооперацій:
- встановлення реєстра в нуль (скидання, погашення);
- прийом слова з іншого реєстра, суматора, лічильника і т. д.
- передача слова на інший реєстр, суматор, лічильник і т. д.
- перетворення кодів збережуваних слів в інверсні коди;
- зсув слова вліво або вправо на потрібне число розрядів;
- перетворення послідовного коду в паралельний і навпаки;
- порозрядні операції кон'юнкції, диз'юнкції й додавання за mod 2.

Схеми конкретних реєстрів в конкретних випадках можуть реалізувати лиш деякі з перерахованих мікрооперацій.

# Головні елементи та вузли ЕОМ

- **Регістри** будують на основі тригерних схем. Кількість тригерів визначає розрядність слів, які записують чи зберігають у регістрі.

Регістри є:

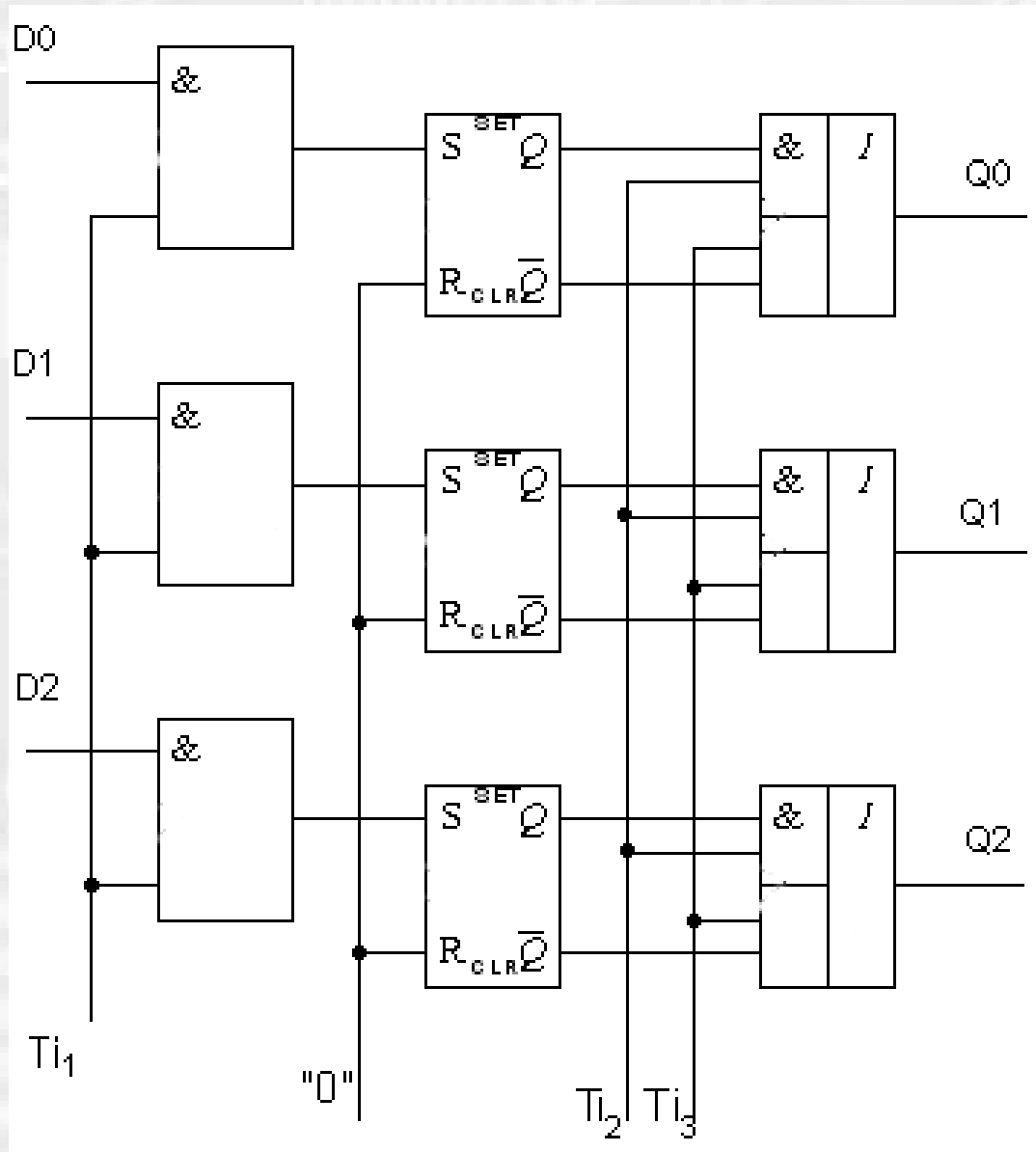
- *послідовні і паралельні,*
- *одно- і двотактні,*
- *зсувні*
- *перетворювальні.*

Розглянемо кілька прикладів схем регістрів. Побудуємо регістр, наприклад, трирозрядний паралельний. Очевидно, що для цього треба мінімум три тригери, які будемо використовувати для запису кожного розряду.

Використаємо звичайні *RS*-тригери.

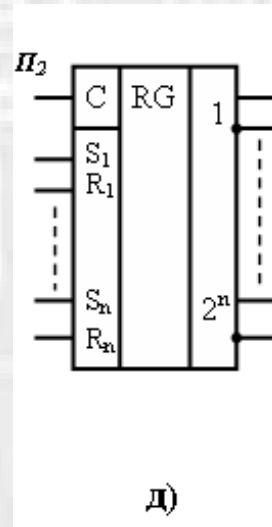
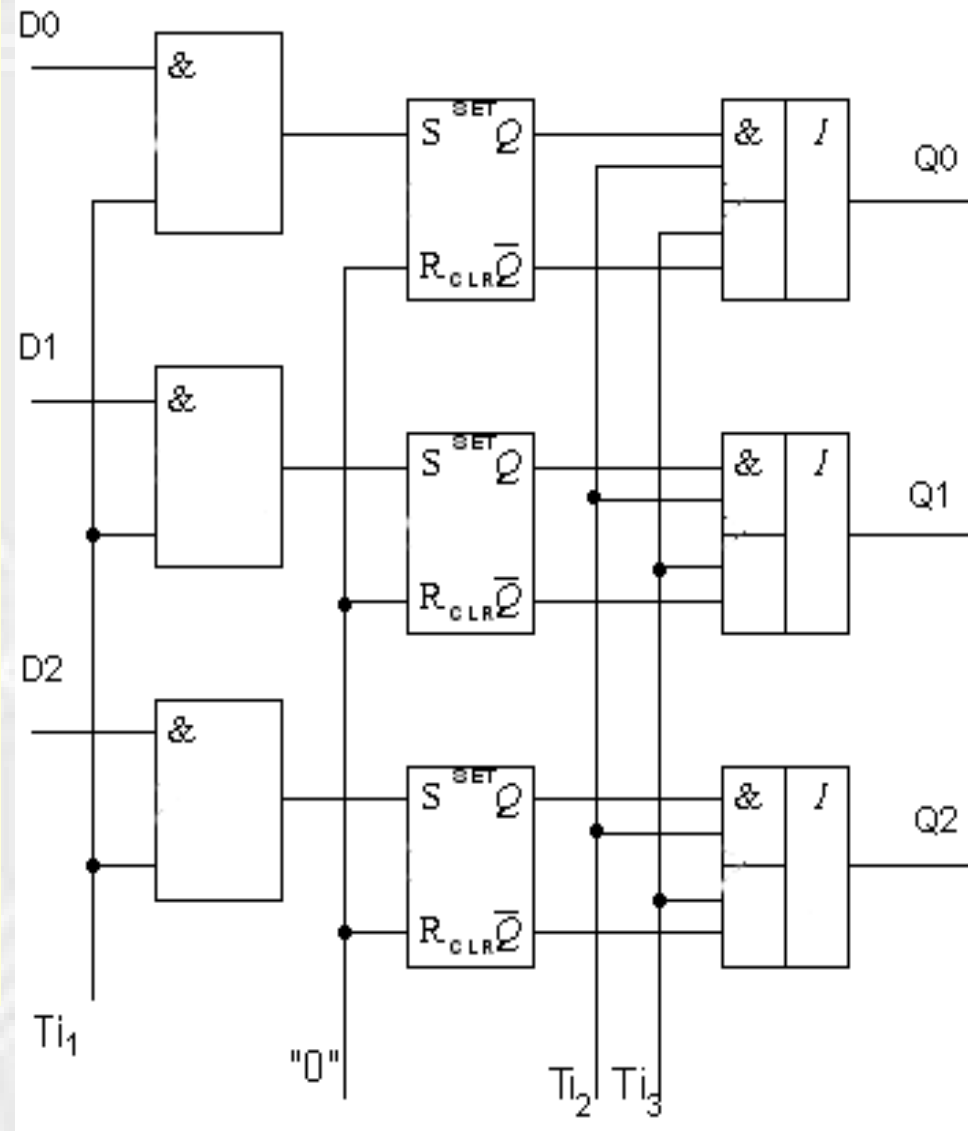
## Головні елементи та вузли EOM

Трохи добудуємо схему ліворуч і праворуч. Ліворуч організуємо синхронний запис інформації, а праворуч – формування прямого та оберненого коду. Перед записом інформації всі тригери сигналом керування  $R$  поставило в нуль  $\Rightarrow (Q=0)$ . Запис у тригери виконується за тактовим імпульсом  $Ti_1$ . Власне ці два сигнали ( $R$  і  $Ti_1$ ) визначають тип регістра, який називають двотактовим паралельної дії (див.рис.).



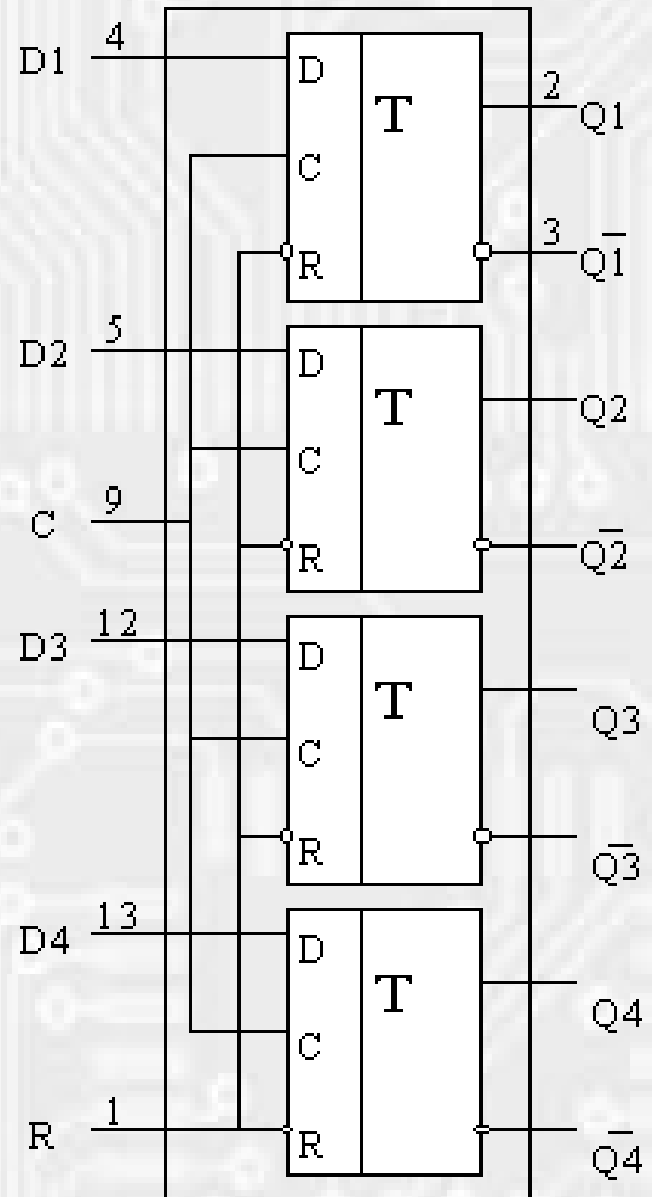
## Головні елементи та вузли EOM

- Код регістра видають за допомогою другого і третього тактових імпульсів:  $Ti_2$  - прямий код,  $Ti_3$  - обернений код.
- Якщо на вхід можна подавати парафазний код (тобто вхідне значення подають у прямому й оберненому коді), то відпадає потреба у такті установлення в "0", тобто отримуємо однотактний паралельний регістр.



## Головні елементи та вузли ЕОМ

- **Регістри зберігання** будують також з одноступеневих D- тригерів. З цією метою можна також застосовувати і JK-тригери, але їх можливості більші, ніж потрібно для регістру зберігання. Варіант чотирирозрядного регістра зберігання наведено на рис. Тут зображено ІМС K155TM8, яка містить 4 D-тригери з об'єднаними входами установки нуля та синхронізації. Числа в тригери регістра записуються по входах D при сигналі дозволу  $C=1$ . Після зміни сигналу на вході C на 0 тригери переходять в режим зберігання. В цей час на входи D можна подати наступне число, яке при  $C=1$  запишеться в регістр. Для обнулення регістра до входу R підводять від'ємний імпульс на час, який потрібний для переведення всіх тригерів в 0. Цей час вказують у довідниках. Поки регістр виконує свої функції, на вході R підтримується напруга високого рівня.



## Головні елементи та вузли ЕОМ

- Регістр є дуже зручним пристроєм для "зсування" інформації праворуч чи ліворуч або перетворення послідовного коду у паралельний (рис.).
- Зсув можна робити як праворуч, так і ліворуч. Такі реєстри називають *реверсивними*.

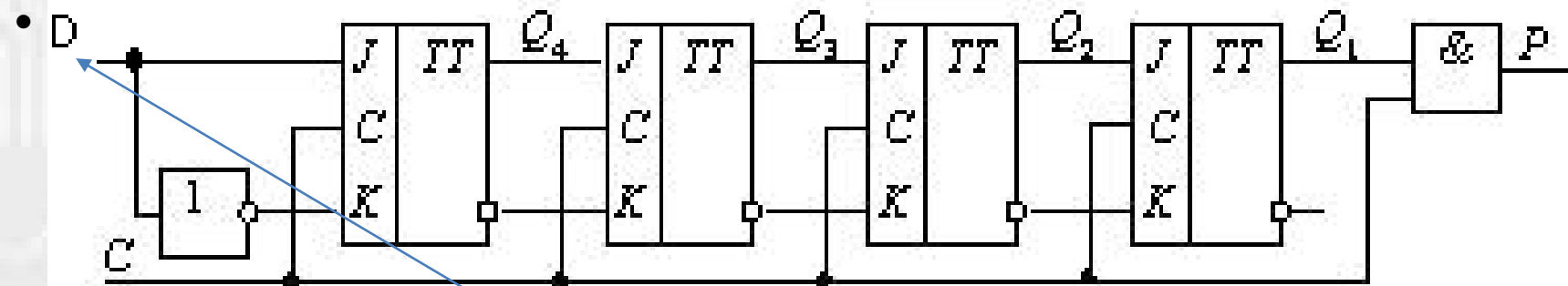
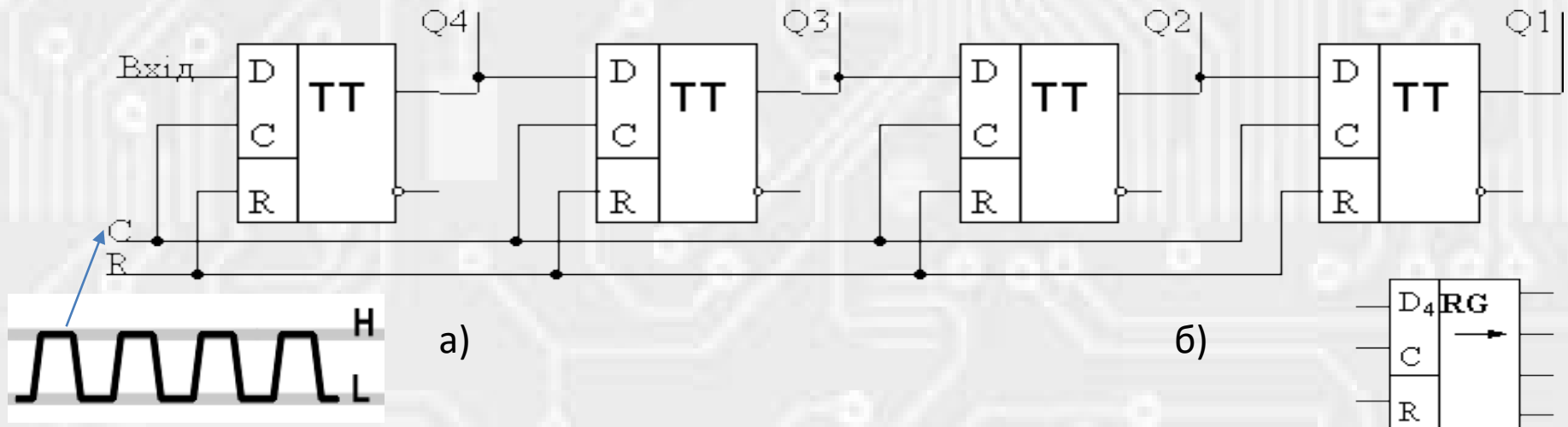


Схема чотирирозрядного реєстра для зсування вправо, яка забезпечує перетворення кодів, побудована на JK-тригерах. Старший розряд реєстра за допомогою інвертора на K-вході працює в режимі D-тригера. На вхід K старшого розряду тригера Q4 по лінії D надходить послідовний код, наприклад, **1101** (код передається в напрямку від старших розрядів коду до молодших). Значення розрядів слова поступає одночасно із синхроімпульсами, які забезпечують як приймання коду в старший розряд, так і одночасний зсув вмісту реєстра вправо. Після приходу чотирьох синхроімпульсів на виходах реєстра Q4 - Q1 встановиться код 1101. В такий спосіб здійснюється перетворення послідовного коду в паралельний, яке часто називають послідовним введенням слова в реєстр.



# Головні елементи та вузли ЕОМ

- Регістр для "зсування" інформації праворуч чи ліворуч (рис.).

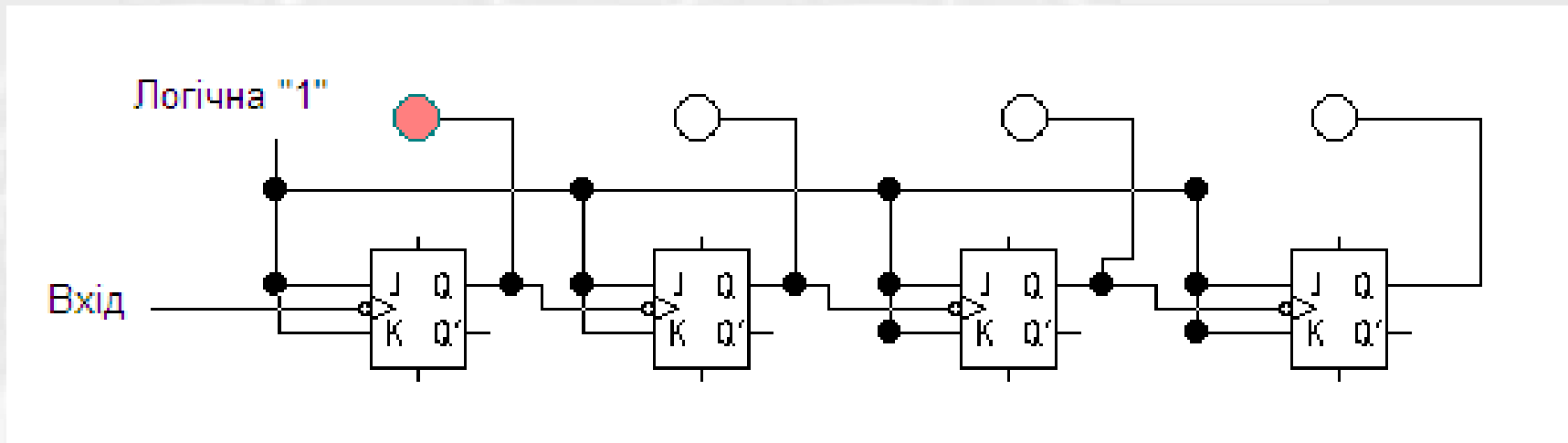


Нехай у регістрі є код **1011**. Стан тригерів після кожного імпульсу зсуву  $t_i$  буде змінюватися відповідно до таблиці:

Імпульс	$t_1$	$t_2$	$t_3$	$t_4$
$T_4$	1	0	0	0
$T_3$	0	1	0	0
$T_2$	1	0	1	0
$T_1$	1	1	0	1

## Головні елементи та вузли ЕОМ

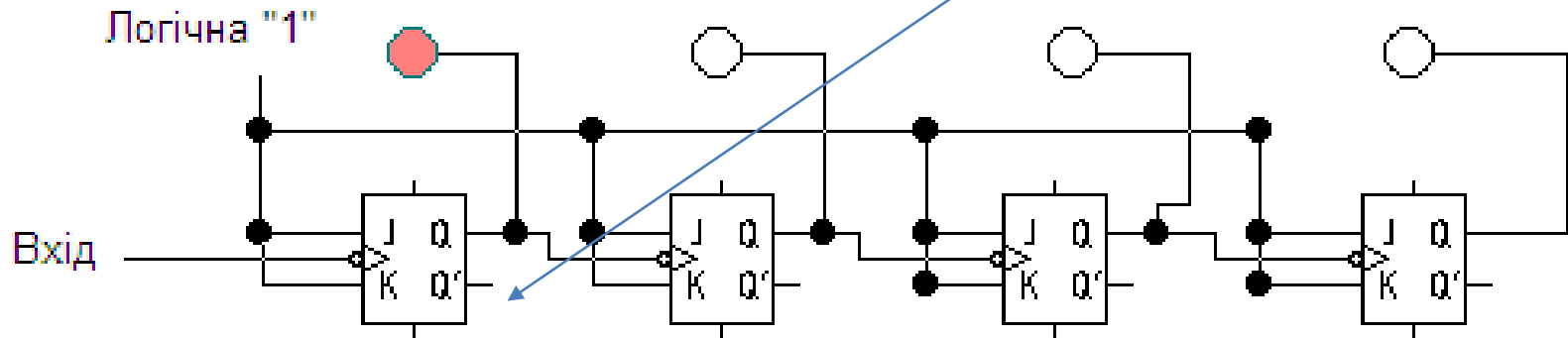
- Лічильник** – пристрій, призначений для підрахунку кількості імпульсів. Лічильники бувають підсумовувальні, віднімальні та реверсивні. Реверсиний лічильник залежно від перекомутації може бути підсумовувальним або віднімальним. Будують лічильники на основі тригерів, використовують у пристроях керування та в арифметичних пристроях для рахунку номерів команд, кількості циклів програми, кількості тактів у разі множення і ділення, а також як суматори. Приклад 4-розрядного підсумовувального двійкового лічильника, зібраного на синхронних JK-тригерах, які працюють у режимі Т-тригерів, показано на рис.



Такий лічильник порахує до 16 (1111) і знову стане в "0". Максимальна кількість імпульсів, яку може підрахувати двійковий лічильник, що складається з  $n$  розрядів, дорівнює  $2^n - 1$ .

## Головні елементи та вузли EOM

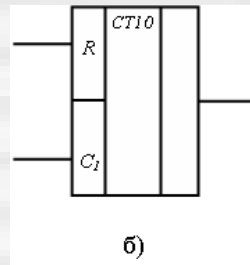
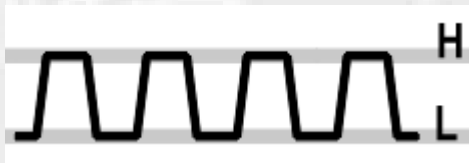
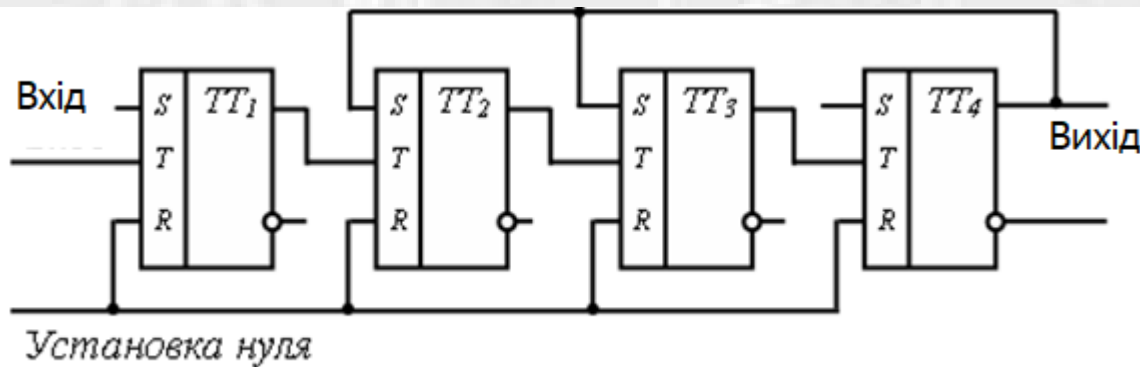
- Віднімальний лічильник отримаємо тоді, коли сигнал знімати з інверсного виходу тригера.
- Лічильники, які однаково можна використовувати як для додавання, так і для віднімання імпульсів, називають *реверсивними*.



Максимальна кількість імпульсів, яку може підрахувати двійковий лічильник, що складається з  $n$  розрядів, дорівнює  $2^n - 1$ .

# Головні елементи та вузли ЕОМ

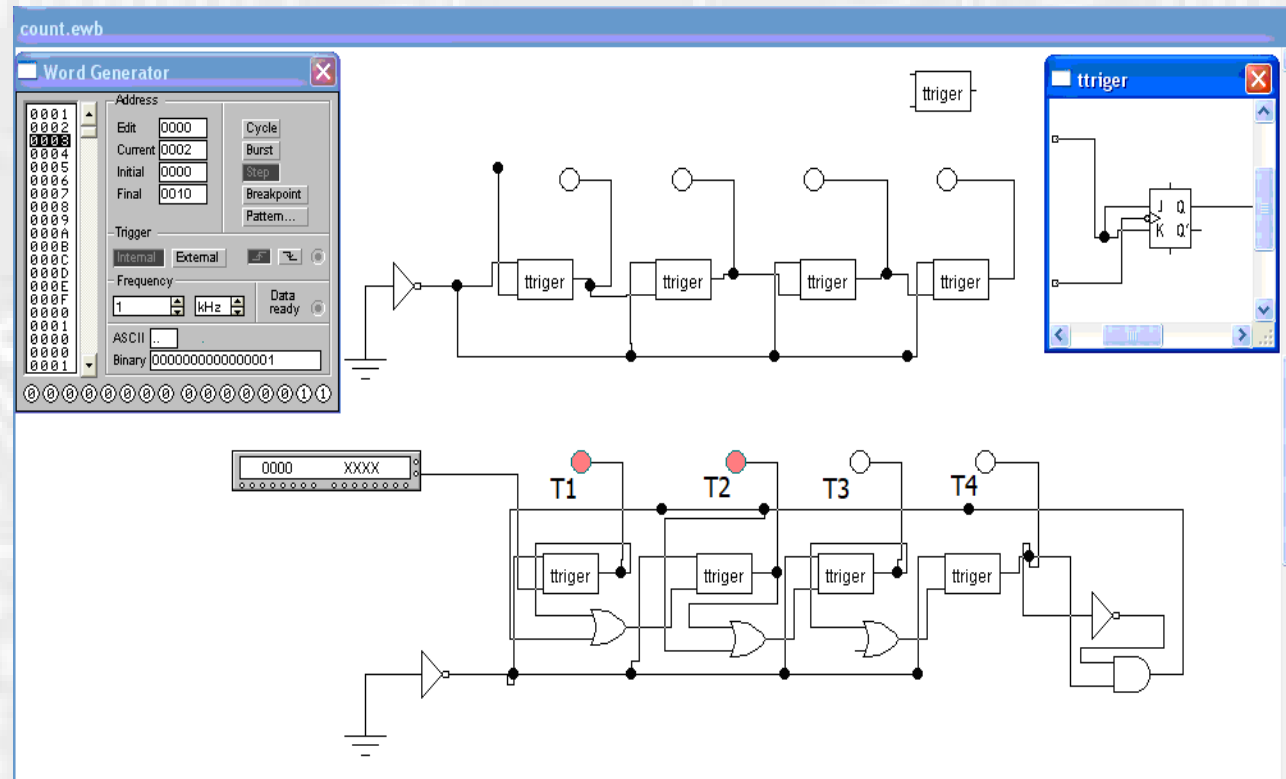
- Іноді потрібно отримати значення  $K \neq 2^n$ . Тоді будують спеціальні перерахункові схеми, де від певних розрядів уводять обернений зв'язок. Перерахункові схеми з  $K=10$  називають декадними лічильниками, які застосовують для побудови **десяткових лічильників** (рис.). На схемі показано, що вихідний сигнал тригера  $T_4$ , потрапляє на входи  $T_2$  і  $T_3$ . Завдяки цьому після надходження на вхід лічильника восьмого імпульсу на виході тригера  **$T_4$**  з'являється сигнал „1”, який переводить тригери  **$T_3$ ,  $T_2$** , із стану „0” в стан „1”



Номер вхідного імпульсу	Стан $T_4$	три $T_3$	ге $T_2$	рів $T_1$
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
<b>8</b>	<b>1</b>	<b>0(1)</b>	<b>0(1)</b>	<b>0</b>
9	1	1	1	1
10	0	0	0	0

# Головні елементи та вузли ЕОМ

Вихідний сигнал тригера  $T_4$ , потрапляє на входи  $T_2$  і  $T_3$ . Завдяки цьому після надходження на вхід лічильника восьмого імпульсу на виході тригера  **$T_4$**  з'являється сигнал „1”, який переводить тригери  **$T_3$** ,  **$T_2$** , із стану „0” в стан „1”



Номер вхідного імпульсу	Стан $T_4$	три $T_3$	ге $T_2$	рів $T_1$
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
<b>8</b>	<b>1</b>	<b>0(1)</b>	<b>0(1)</b>	<b>0</b>
9	1	1	1	1
10	0	0	0	0

# ***Запитання***

