Послідовнісна логіка, тригери, регістри та лічильники

Мета лекції

Сформувати цілісне розуміння того, як цифрова система зберігає стан, синхронізує оновлення даних і будує регістри та лічильники з однобітних комірок пам’яті.

1. Комбінаційна і послідовнісна логіка

У комбінаційній схемі історія входів не має значення. У послідовнісній схемі однаковий поточний вхід може дати різний вихід, якщо попередній стан відрізняється.

Наприклад, сигнал D=0 безпосередньо не визначає Q регістра, доки не настане активний фронт Clock. До цього моменту регістр утримує раніше записаний біт.

Стан (state) — мінімальний набір збережених бітів, необхідний для прогнозування подальшої поведінки пристрою. Один тригер зберігає один біт стану, а n тригерів можуть подати до 2^n різних станів.

Поточний стан позначають Q(t), наступний стан — Q(t+1) або Q^+. Перехід від Q(t) до Q(t+1) викликає тактова подія або зміна керувального рівня залежно від типу елемента пам’яті.

Елементарна пам’ять виникає завдяки зворотному зв’язку: частина виходу повертається на вхід логічної структури. Після зникнення короткої команди Set або Reset схема може зберігати сформований стан.

Зворотний зв’язок робить поведінку часовою. Для правильної роботи потрібно враховувати затримки елементів і не подавати комбінації, які змушують взаємопов’язані гілки одночасно встановлювати суперечливі значення.

Властивість	Комбінаційна логіка	Послідовнісна логіка
Модель	Y=F(X)	Q^+=F(Q,X), Y=G(Q,X)
Пам’ять	Відсутня	Зберігається у тригерах
Приклади	Дешифратор, суматор	Регістр, лічильник, автомат

2. Тактування і часові параметри

2.1 Тактовий сигнал Clock

Тактовий сигнал (clock, CLK) задає моменти, у які синхронна система дозволяє зміну стану. Період T вимірюють між однаковими сусідніми фронтами, а частоту визначають як f=1/T.

Для частоти 50 MHz період становить 20 ns. Це весь часовий бюджет одного такту: до наступного активного фронту мають завершитися затримка тригера, робота комбінаційної логіки та час підготовки наступного тригера.

2.2 Рівень і фронт

Засувка чутлива до рівня (level-sensitive): поки сигнал дозволу активний, зміни входу можуть проходити на вихід. Тригер чутливий до фронту (edge-triggered): він вибірково фіксує значення лише під час переходу Clock.

Висхідний фронт (rising edge) — перехід 0→1, спадний фронт (falling edge) — перехід 1→0. Трикутник біля входу CLK позначає фронтове керування, а кружок разом із трикутником — спрацювання за спадним фронтом.

2.3 Час підготовки і час утримання

Час підготовки (setup time, tsetup) задає мінімальний інтервал, протягом якого D має бути стабільним до активного фронту. Час утримання (hold time, thold) задає мінімальний інтервал стабільності після фронту.

Якщо фронт настає в момент 100 ns, tsetup=2 ns, а thold=1 ns, вхід D не повинен змінюватися в інтервалі від 98 ns до 101 ns. Цей інтервал називають вікном захоплення.

2.4 Метастабільність

Порушення setup time або hold time може перевести тригер у метастабільний стан (metastability): вихід певний час не встановлюється на надійний 0 або 1. Тривалість такого стану неможливо точно передбачити.

Метастабільність не усувають логічним рівнянням. Для асинхронних однобітних сигналів застосовують синхронізатор із двох послідовних D-тригерів. Перший може стати метастабільним, а другий отримує додатковий такт для стабілізації.

Приклад. Для Clock=25 MHz період T=40 ns. Якщо шлях між двома тригерами потребує 31 ns, а tsetup приймального тригера дорівнює 4 ns, загальна вимога 35 ns виконується із запасом 5 ns.

3. Засувки

3.1 SR-засувка

SR-засувка (Set–Reset latch, засувка встановлення–скидання) має входи S і R. Команда S=1 встановлює Q=1, команда R=1 скидає Q=0, а комбінація S=R=0 зберігає попередній стан.

Для реалізації на двох перехресно з’єднаних NOR елементах комбінація S=R=1 заборонена. Вона примусово робить Q і $\overline{Q}$ однаковими, а після одночасного повернення входів у нуль остаточний стан може залежати від затримок.

3.2 Таблиця SR-засувки

Рядок зберігання не можна трактувати як невизначений вихід: Q^+ точно дорівнює попередньому Q. Позначення Q означає залежність від історії, а не довільне значення.

Заборонена комбінація потрібна для розуміння обмежень SR-структури. Практичні схеми або не допускають її, або застосовують D-, JK- чи T-тригер, у яких поведінка для всіх входів визначена.

3.3 SR-засувка на NAND

Засувка на NAND має активні нульові входи $\overline{S}$ і $\overline{R}$. Нуль на $\overline{S}$ встановлює Q=1, нуль на $\overline{R}$ скидає Q=0, а $\overline{S}=\overline{R}=1$ утримує стан.

Риска над назвою сигналу означає активність низьким рівнем (active low). Тому таблиця NAND-варіанта не копіює таблицю NOR-варіанта: логічні команди інвертовані.

3.4 D-засувка

D-засувка (Data latch, засувка даних) усуває заборонену комбінацію SR-засувки. З одного входу D формуються взаємно протилежні команди S і R, а сигнал Enable визначає, чи дозволено оновлення.

Коли Enable=1, засувка прозора і Q повторює D після затримки поширення. Коли Enable=0, вхідні команди неактивні, тому Q зберігає останнє значення. Рівняння: $Q^{+}=E\cdot D \vee \overline{E}\cdot Q$.

3.5 Прозорість D-засувки

Під час активного рівня Enable короткі зміни D можуть пройти на Q. Це називають прозорістю (transparency). Вона корисна в спеціальних двофазних схемах, але ускладнює аналіз, якщо кілька засувок одночасно відкриті.

Якщо D змінюється 0→1→0, поки E=1, вихід повторить обидва переходи. Тригер з фронтовим керуванням за той самий час зафіксував би лише значення в момент активного фронту.

S	R	Q^{+}	Тлумачення
0	0	Q	Пам’ять
0	1	0	Reset
1	0	1	Set
1	1	—	Заборонено

Приклад D-засувки. Нехай Q=0. Поки E=0, зміни D не впливають на Q. Після E=1 вихід повторює D. Якщо перед закриттям E повернути D у 1, засувка збереже 1 після E=0.

4. Тригери

4.1 Latch і Flip-Flop

Засувка (latch) реагує протягом усього активного рівня керування. Тригер (flip-flop) фіксує дані в короткий момент активного фронту. Це різні часові моделі, хоча обидва вузли зберігають один біт.

У синхронних системах фронтові тригери спрощують часовий аналіз: стан змінюється лише біля фронтів. Засувки можуть давати виграш у швидкодії, але потребують ретельного проєктування фаз і часових запасів.

4.2 D-тригер

D-тригер (D flip-flop, тригер даних) копіює D у Q під час активного фронту Clock і зберігає це значення до наступного фронту. Характеристичне рівняння має простий вигляд $Q^+=D$.

D-тригер є базовою коміркою регістра. Якщо чотири D-тригери отримують спільний Clock, вони одночасно фіксують чотирибітне слово D3D2D1D0.

4.3 Часова діаграма D-тригера

На часовій діаграмі значення D аналізують лише в позначених активних фронтах. Зміни між фронтами не впливають на Q, якщо не порушено setup time і hold time.

Приклад: перед трьома послідовними висхідними фронтами D дорівнює 1, 0 і 1. Після затримки clock-to-Q вихід Q набуде послідовності 1, 0, 1 та утримуватиме кожне значення протягом такту.

4.4 SR-тригер

SR-тригер переносить функції Set і Reset у фронтову модель. Входи аналізуються лише в активний фронт, після якого Q оновлюється.

Комбінація S=R=0 означає зберігання, S=1,R=0 — встановлення, S=0,R=1 — скидання. Для класичного SR-тригера S=R=1 залишається забороненою або невизначеною комбінацією.

4.5 JK-тригер

JK-тригер (JK flip-flop) розв’язує проблему S=R=1. Входи J і K відповідають функціям встановлення і скидання, але комбінація J=K=1 викликає перемикання Q у протилежний стан.

Характеристичне рівняння: $Q^+=J\cdot\overline{Q}\vee\overline{K}\cdot Q$. Воно дає зберігання для 00, скидання для 01, встановлення для 10 і перемикання для 11.

4.6 Таблиця JK-тригера

Для визначення Q^+ недостатньо знати лише J і K у режимах 00 та 11: потрібний поточний Q. При 00 наступний стан дорівнює Q, при 11 — $\overline{Q}$.

JK-тригер з J=K=1 працює як T-тригер з T=1. Тому він може бути основою лічильника, хоча в сучасних синхронних схемах часто використовують D-тригери з відповідною комбінаційною логікою.

4.7 T-тригер

T-тригер (Toggle flip-flop, перемикальний тригер) має один інформаційний вхід T. При T=0 він зберігає Q, при T=1 інвертує Q в кожному активному фронті.

Характеристичне рівняння $Q^+=Q\oplus T$. T-тригер можна отримати з D-тригера, подавши на D сигнал $Q\oplus T$, або з JK-тригера, з’єднавши J=K=T.

4.8 Перемикання T-тригера

Якщо T постійно дорівнює 1, Q змінює стан один раз за період Clock: 0,1,0,1... Для повного періоду Q потрібні два такти, тому частота Q дорівнює fCLK/2.

Якщо T=0 протягом окремого фронту, перемикання пропускається. Ця властивість безпосередньо використовується як Clock Enable у синхронних лічильниках.

4.9 Порівняння тригерів

D-тригер безпосередньо записує потрібний наступний стан. SR-тригер зручно описує команди встановлення і скидання. JK-тригер має повністю визначену поведінку, а T-тригер компактно реалізує перемикання.

Вибір типу тригера залежить від способу формування наступного стану. Одну й ту саму функцію можна реалізувати різними тригерами, якщо правильно синтезувати їхні вхідні рівняння.

J	K	Q	Q^+
0	0	0	0
0	0	1	1
0	1	0 або 1	0
1	0	0 або 1	1
1	1	0	1
1	1	1	0

Покроковий приклад T-тригера. Початково Q=0. Для послідовності T у п’яти фронтах 1,1,0,1,1 наступні стани дорівнюють 1,0,0,1,0. На третьому фронті T=0, тому стан зберігається.

5. Reset та Enable

5.1 Reset: синхронний і асинхронний

Скидання (Reset) переводить стан у відоме початкове значення. Асинхронний Reset діє незалежно від Clock, а синхронний Reset аналізується лише під час активного фронту.

Асинхронне скидання швидко задає стан, але його зняття біля фронту може порушити часові вимоги. Поширена практика: асинхронно активувати Reset і синхронно знімати його через спеціальний ланцюг.

5.2 Сигнал Enable

Дозвіл (Enable, EN) визначає, чи має стан оновитися в поточному такті. Для D-тригера з EN можна записати $D_{next}=EN\cdot D_{data} \vee EN\cdot Q$. При EN=0 на вхід фактично повертається поточний Q.

Clock Enable безпечніший за довільне логічне блокування тактового сигналу. Гейтування Clock звичайними вентилями може створити короткі імпульси і неконтрольовані фронти.

6. Регістри

6.1 Регістр

Регістр (register) — група тригерів зі спільними сигналами Clock і керування, яка зберігає багаторозрядне слово. Чотирибітний регістр містить щонайменше чотири комірки пам'яті.

Паралельний вхід означає, що всі біти подаються одночасно, а паралельний вихід дозволяє одночасно прочитати все слово. Реальний регістр також може мати Enable, Reset і вибір режиму.

6.2 Чотирибітний паралельний регістр

Чотири D-тригери отримують D3…D0 і спільний Clock. На активному фронті кожний тригер фіксує свій біт, тому слово оновлюється як єдине ціле.

Якщо D=1011 безпосередньо перед фронтом, а часові вимоги виконані, після затримки clock-to-Q виходи Q3…Q0 стають 1011. Зміна D після фронту не змінить Q до наступної події запису.

6.3 Приклад завантаження регістра

Нехай початково Q=0101. У першому такті EN=0, D=1110, тому регістр утримує 0101. У другому такті EN=1, D=1110, тому записує 1110. У третьому такті активується Reset, тому Q переходить у 0000.

Послідовність демонструє пріоритет керування. Якщо специфікація задає Reset вище за Enable, одночасні Reset=1 та EN=1 повинні дати 0000, а не D.

6.4 Регістр зсуву

Регістр зсуву (shift register) переміщує біти між сусідніми тригерами в кожному такті. Залежно від напрямку використовують зсув праворуч або ліворуч, а звільнене місце заповнює послідовний вхід.

Для правого зсуву чотирибітного Q3Q2Q1Q0 можна задати $Q3^{+}$=Sin, $Q2^{+}$=Q3, $Q1^{+}$=Q2, $Q0^{+}$=Q1. Усі присвоєння відбуваються одночасно на одному фронті.

6.5 Приклад зсуву 1011

Початковий стан 1011, правий зсув, Sin=0. Після першого фронту стан 0101, після другого — 0010, після третього — 0001, після четвертого — 0000.

Під час кожного такту крайній молодший біт залишає регістр через послідовний вихід Sout. Для початкового слова 1011 послідовність вихідних бітів під час правого зсуву: 1,1,0,1.

6.6 Типи регістрів зсуву

SISO (Serial In Serial Out, послідовний вхід і послідовний вихід) передає бітовий потік із затримкою. SIPO (Serial In Parallel Out) перетворює послідовні дані на слово.

PISO (Parallel In Serial Out) виконує зворотне перетворення, а PIPO (Parallel In Parallel Out) є звичайним паралельним регістром. Скорочення описують спосіб введення і виведення даних.

Такт	Reset	EN	D	Q
0	0	—	—	0101

Такт	Reset	EN	D	Q
1	0	0	1110	0101
2	0	1	1110	1110
3	1	1	0011	0000

Тип	Розшифрування	Призначення
SISO	Serial In Serial Out	Затримка послідовного потоку
SIPO	Serial In Parallel Out	Послідовно-паралельне перетворення
PISO	Parallel In Serial Out	Паралельно-послідовне перетворення
PIPO	Parallel In Parallel Out	Збереження слова

7. Лічильники

7.1 Лічильник і модуль лічби

Лічильник (counter) змінює стан за визначеною числовою послідовністю. Двійковий n-бітний лічильник природно має модуль 2^n, тобто проходить усі стани від 0 до 2^n-1.

Для модуля N потрібна мінімальна розрядність $n=\lceil \log_2 N \rceil$. Десятковий лічильник MOD-10 потребує 4 біти, але використовує лише стани 0000…1001; інші шість станів мають бути оброблені схемою повернення.

7.2 Асинхронний лічильник

В асинхронному лічильнику (asynchronous counter, ripple counter) зовнішній Clock надходить лише на перший тригер. Кожний наступний тригер тактується виходом попереднього.

Структура проста, але зміни проходять каскадом. Під час переходу 0111→1000 проміжно можуть з’явитися 0110, 0100 і 0000. Тому дешифратор виходів може короткочасно побачити хибний код.

7.3 Затримки асинхронного лічильника

Якщо затримка одного тригера $t_p=8$ ns, зміна четвертого біта може завершитися приблизно через $4\cdot 8=32$ ns після зовнішнього фронту. Це обмежує максимальну частоту і створює перехідні коди.

Назва ripple означає, що фронт ніби хвилею проходить через розряди. Такий лічильник підходить для простого ділення частоти, але не для схем, де всі біти мають змінитися одночасно.

7.4 Синхронний двійковий лічильник

У синхронному лічильнику (synchronous counter) всі тригери отримують спільний Clock. Комбінаційна логіка заздалегідь визначає, які розряди мають перемкнутися на наступному фронті.

Для лічби вгору T0=1, T1=Q0, T2=Q1Q0, T3=Q2Q1Q0. Молодший біт перемикається завжди, а кожний старший — лише коли всі молодші біти дорівнюють одиниці.

7.5 Таблиця трирозрядного лічильника

Трирозрядний лічильник MOD-8 проходить послідовність 000,001,010,011,100,101,110,111 і повертається в 000. Таблиця переходів однозначно задає його функцію.

Приклад: якщо поточний стан 110, наступний стан 111; після ще одного фронту виникає 000. Повернення від максимального коду до нуля називають переповненням лічильника.

7.6 Лічба вгору і вниз

Реверсивний лічильник (up/down counter) використовує керувальний сигнал напрямку. Для лічби вгору старший розряд перемикається, коли всі молодші дорівнюють 1; для лічби вниз — коли всі молодші дорівнюють 0.

Сигнал Enable дозволяє або забороняє зміну стану, а Reset задає початковий код. Якщо EN=0, лічильник зберігає поточне значення незалежно від напрямку.

Поточний стан	Наступний стан
000	001
001	010
010	011
011	100
100	101
101	110
110	111
111	000

Приклад MOD-10. Після стану 1001_2=9 лічильник має перейти в 0000. Стани 1010…1111 не належать основній послідовності. Надійна схема визначає для них перехід у допустимий стан, наприклад 0000.

8. Дільники частоти

Дільник частоти (clock divider) формує повільніший періодичний сигнал. Один T-тригер з T=1 ділить частоту на 2, каскад n таких ступенів — на 2^n.

Для перетворення 8 MHz на 1 MHz потрібне ділення на $8=2^3$, тобто три ступені. Виходи послідовно матимуть 4 MHz, 2 MHz і 1 MHz. Для довільного коефіцієнта застосовують лічильник і порівняння з граничним значенням.

Приклад. Для отримання 125 kHz із 8 MHz потрібно ділення на $64=2^6$, отже достатньо шести послідовних ступенів T-тригерів або шести молодших розрядів двійкового лічильника.

9. Правила надійного проєктування

Clock визначає момент оновлення, Enable — дозвіл на оновлення, Reset — початковий стан. Змішування цих ролей породжує помилки: дані не повинні використовуватися як такт, а звичайний вентиль не повинен без перевірки формувати Clock.

У синхронному проєкті один домен тактування має спільне джерело Clock. Переходи між різними доменами потребують синхронізаторів або спеціальних схем передавання.

- Один тактовий домен використовує узгоджений Clock.
- Асинхронні входи проходять синхронізацію.
- Reset і Enable мають явно визначений пріоритет.
- Вхідні дані стабільні в межах setup time і hold time.
- Невикористані стани лічильника мають визначений шлях повернення.

10. Терміни і скорочення

Термін або скорочення	Повна назва	Зміст
CLK	Clock	Тактовий сигнал
FF	Flip-Flop	Фронтовий тригер
SR	Set–Reset	Встановлення–скидання
D	Data	Вхід даних
JK	JK flip-flop	Тригер без забороненої комбінації
T	Toggle	Перемикальний тригер
EN	Enable	Дозвіл оновлення
SISO	Serial In Serial Out	Послідовний вхід і вихід
SIPO	Serial In Parallel Out	Послідовний вхід, паралельний вихід

Термін або скорочення	Повна назва	Зміст
PISO	Parallel In Serial Out	Паралельний вхід, послідовний вихід
PIPO	Parallel In Parallel Out	Паралельний вхід і вихід
MOD-N	Modulus N	Лічильник із N станами

11. Питання для самоперевірки

- Чому однакові входи можуть дати різні виходи послідовнісної схеми?
- Чим рівнева чутливість засувки відрізняється від фронтової чутливості тригера?
- Що відбувається при порушенні setup time або hold time?
- Як працюють режими SR-, D-, JK- і T-тригерів?
- Як Enable забезпечує збереження стану?
- Чому асинхронний лічильник має перехідні коди?
- Як визначити кількість бітів для MOD-N?
- Як отримати ділення частоти на 8?