# Structure Machine

USTHB, le 02/02/2022

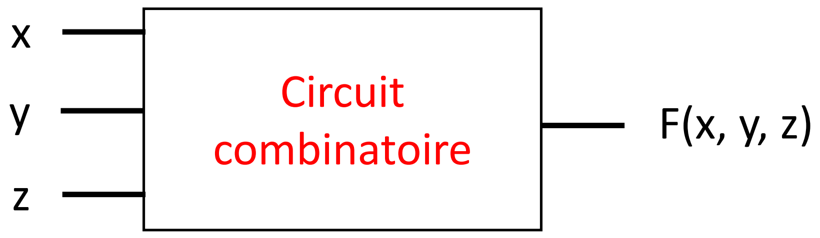
**1- Les circuits séquentiels**
2- **Les mémoires**
**3- Introduction sur les Machines pédagogiques (Structure et fonctionnement d’un ordinateur de base)**

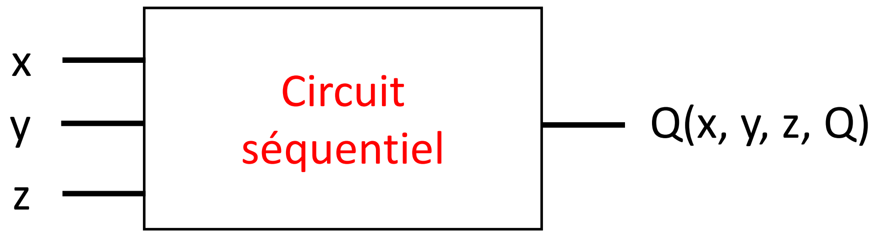
By L.ABADA

# Les circuits séquentiels

**Introduction**

Dans un circuit combinatoire, une sortie est uniquement fonction des entrées. Par contre, dans un circuit séquentiel, une sortie est une fonction des entrées mais aussi des sorties du circuit. Cela signifie qu’un circuit séquentiel garde la mémoire des états passés.



# Les circuits séquentiels

**Définitions**

Une bascule est un circuit logique capable de mémoriser une information binaire.

Cette information est représentée par la sortie Q de la bascule (état de la bascule).

$Q_{t+1} = f\ (E_i, Q_t)$

$Q_{t+1}$ $\quad Q^+$ $\quad Q$

Qt $\quad Q$ $\quad Q^-$

$E_i$ : sont les entrées de la bascule.

$Q_t$ : est l'état de la bascule à l'instant t (ancien état)

$Q_{t+1}$ est l'état de la bascule l'instant t+1 (nouvel état)

Un circuit séquentiel est une interconnexion de bascules.

# Les circuits séquentiels

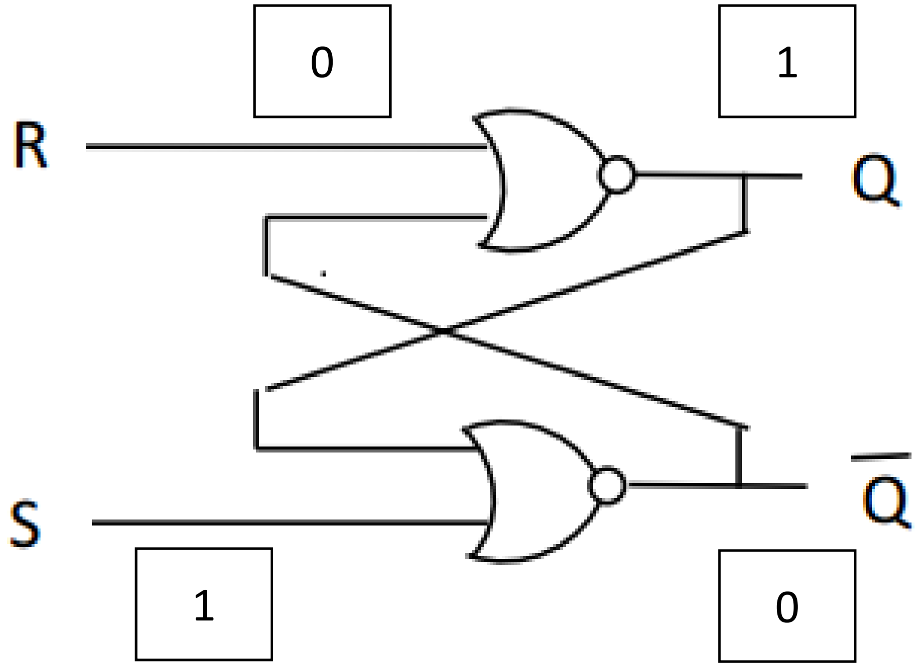
**Bascules usuelles**

Il existe quatre types de bascules :

1. **La bascule RS**
2. **La bascule JK**
3. **La bascule D**
4. **La bascule T**

# Les circuits séquentiels

## 1. Bascule RS

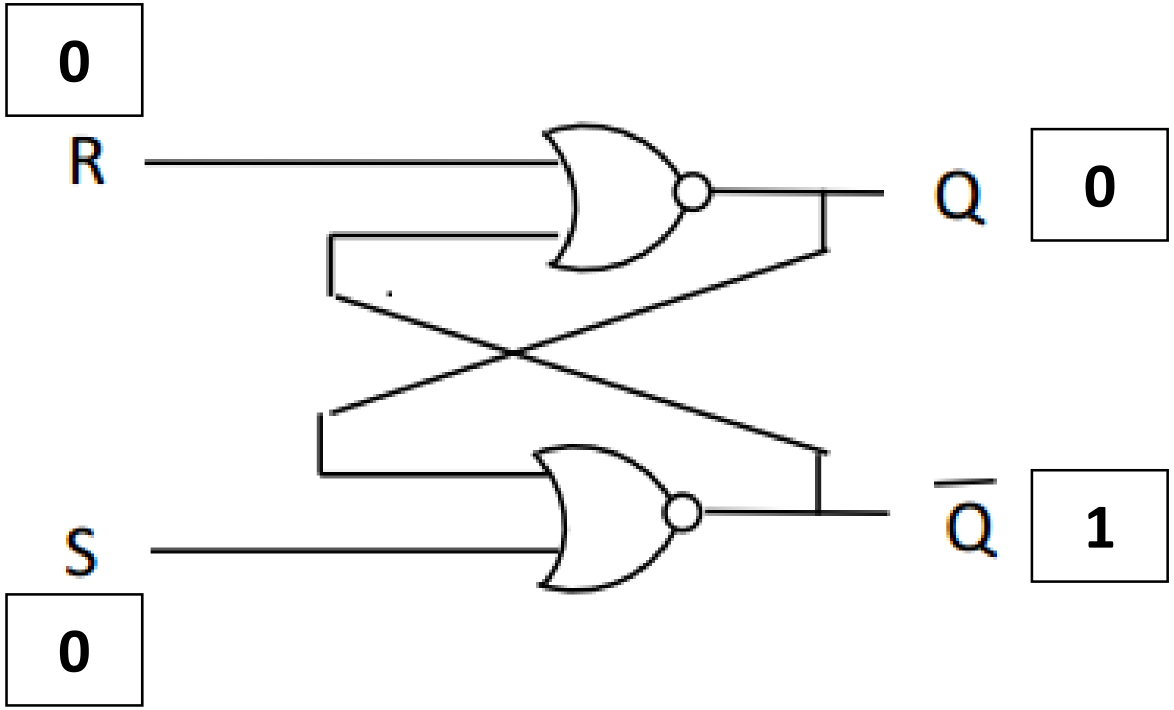



$$Q^+ = \overline{R + \overline{Q}}$$
$$\overline{Q} = \overline{S + Q}$$
$$Q^+ = \overline{R + \overline{S + Q}}$$
$$Q^+ = \overline{R}\ (S + Q)$$
$$Q^+ = \overline{R}\,S + \overline{R}\,Q$$

$$\overline{Q}^+ = \overline{S + Q}$$
$$Q = \overline{R + \overline{Q}}$$
$$\overline{Q}^+ = \overline{S + \overline{R + \overline{Q}}}$$
$$\overline{Q}^+ = \overline{S}\ (R + \overline{Q})$$
$$\overline{Q}^+ = \overline{S}\,R + \overline{S}\,\overline{Q}$$

# Les circuits séquentiels

## 1. Bascule RS




| $R$ | $S$ | $Q$ | $Q^+$ | $\overline{Q}^+$ |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 1 |
| 0 | 0 | 1 | 1 | 0 |
| 0 | 1 | 0 | 1 | 0 |
| 0 | 1 | 1 | 1 | 0 |
| 1 | 0 | 0 | 0 | 1 |
| 1 | 0 | 1 | 0 | 1 |
| 1 | 1 | 0 | 0 | 0 |
| 1 | 1 | 1 | 0 | 0 |

# Les circuits séquentiels

**La table caractéristique de la bascule RS est donc :**

| $R$ | $S$ | $Q^+$ |
|---|---|---|
| 0 | 0 | Q |
| 0 | 1 | 1 |
| 1 | 0 | 0 |
| 1 | 1 | X |

Mémorisation

Mise à 1 « Set »

Mise à 0 « Reset »

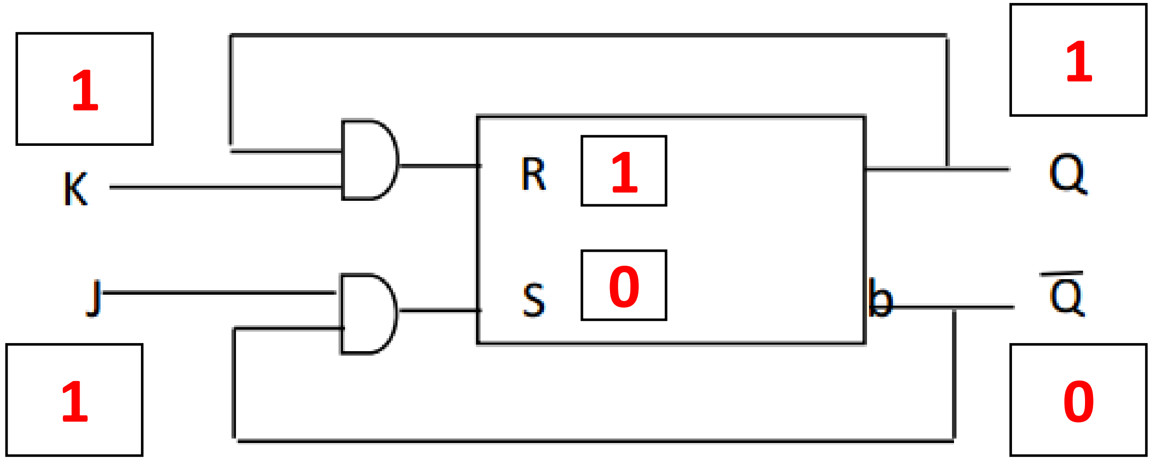
Indéterminé

| $R$ | $S$ | $Q$ | $Q^+$ | $\overline{Q}^+$ |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 1 |
| 0 | 0 | 1 | 1 | 0 |
| 0 | 1 | 0 | 1 | 0 |
| 0 | 1 | 1 | 1 | 0 |
| 1 | 0 | 0 | 0 | 1 |
| 1 | 0 | 1 | 0 | 1 |
| 1 | 1 | 0 | X | X |
| 1 | 1 | 1 | X | X |

Son équation de sortie est $Q^+ = S + \overline{R}Q$

# Les circuits séquentiels

## 1. Bascule JK



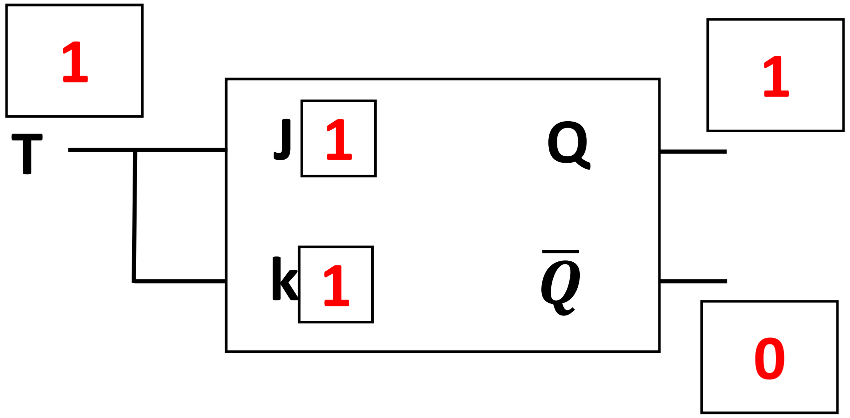

| $K$ | $J$ | $Q$ | $Q^+$ |
|---|---|---|---|
| 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 |
| 0 | 1 | 0 | 1 |
| 0 | 1 | 1 | 1 |
| 1 | 0 | 0 | 0 |
| 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 0 |

| $K$ | $J$ | $Q^+$ |
|---|---|---|
| 0 | 0 | Q |
| 0 | 1 | 1 |
| 1 | 0 | 0 |
| 1 | 1 | $\overline{Q}$ |

Son équation de sortie vaut Q+ = J $\overline{Q}$+ $\overline{K}$ Q

# Les circuits séquentiels

## 1. Bascule T



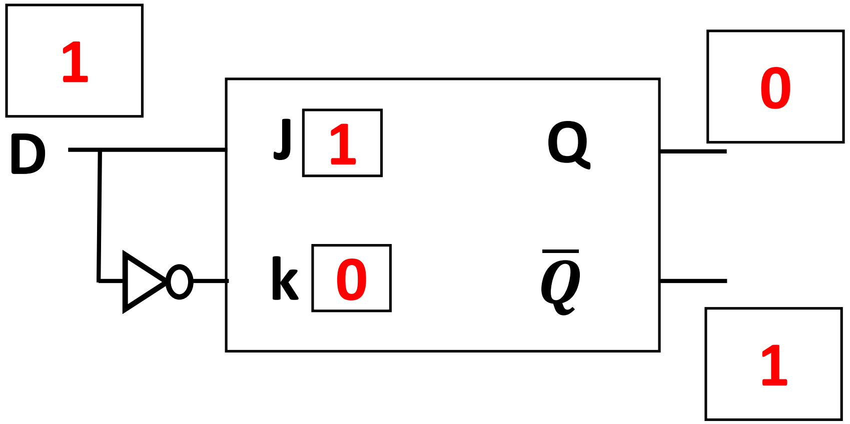

| $T$ | $Q$ | $Q^+$ |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

| $T$ | $Q^+$ |
|---|---|
| 0 | Q |
| 1 | $\overline{Q}$ |

Son équation de sortie vaut $Q^+ = T \oplus Q$

# Les circuits séquentiels

## 1. Bascule D




| $D$ | $Q$ | $Q^+$ |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 1 |
| 1 | 1 | 1 |

| $D$ | $Q^+$ |
|---|---|
| 0 | 0 |
| 1 | 1 |

Son équation de sortie vaut Q+ = D

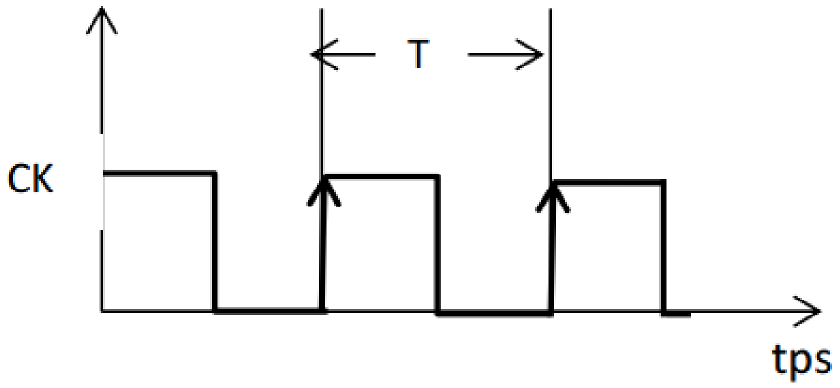
# Les circuits séquentiels

**Notion horloge**

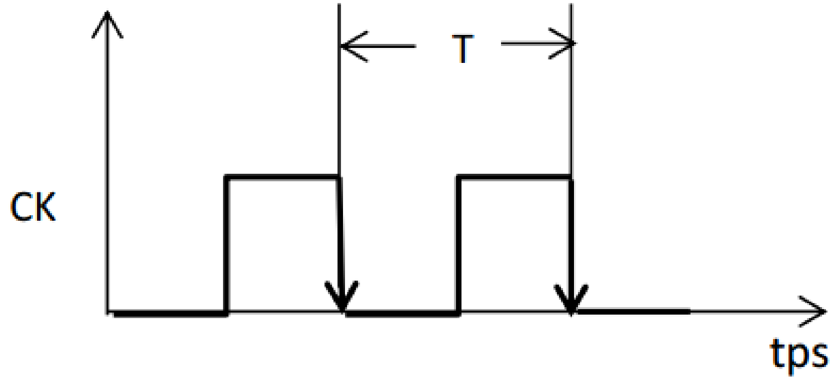
Une horloge est une variable logique qui passe successivement de 0 à 1 et de 1 à 0 d'une façon périodique.

Cette variable est utilisée souvent comme une entrée des circuits séquentiels, le circuit est dit alors synchrone.

L'horloge est notée généralement CK ( Clock) ou parfois H (Horloge)




Horloge active sur front montant




Horloge active sur front descendant

T est la période calculée généralement en secondes.

F = 1/T est la fréquence calculée en Hertz

# Les circuits séquentiels

**Bascules asynchrones.**

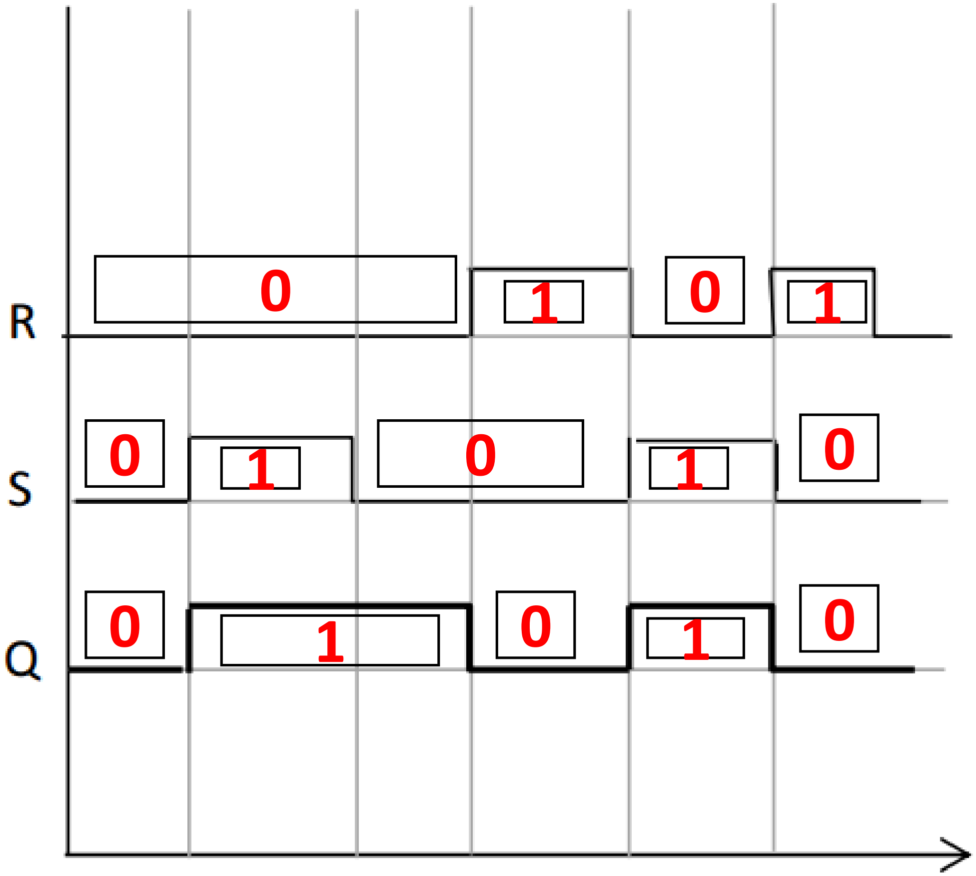
Les sorties des bascules asynchrones peuvent changer à tout moment dès qu’une ou plusieurs entrées changent.

**Bascules synchrone.**

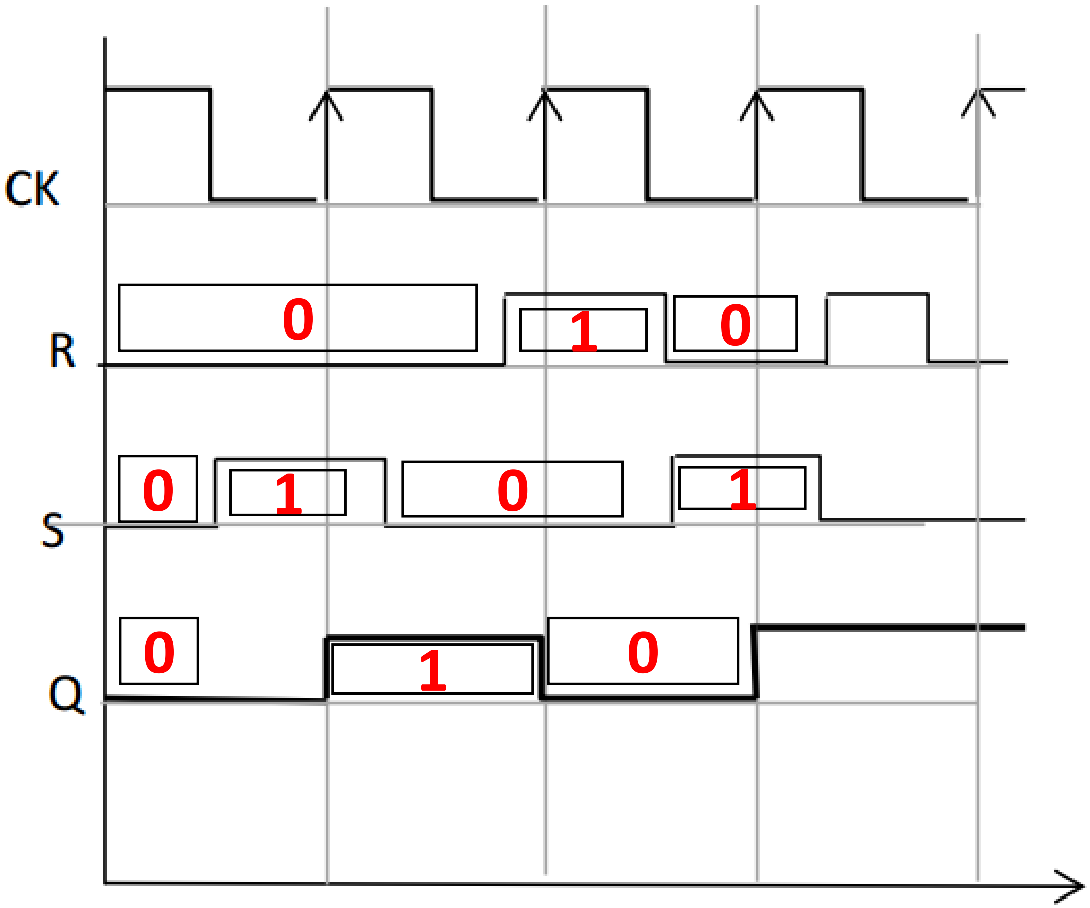
Le changement sur les sorties se produit après le changement de l’horloge. Les entrées servent à préparer le changement d’état, mais ne provoquent pas de changement des sorties. Tout changement d’état est synchronisé par l’horloge.

# Les circuits séquentiels

**Exemples de bascules synchrone et asynchrone**




Bascule RS asynchrone




Bascule RS synchrone

# Les circuits séquentiels

**Analyse d'un circuit séquentiel**

Pour analyser un circuit séquentiel, il faut :

a) Etablir les équations d'entrée de chaque bascule

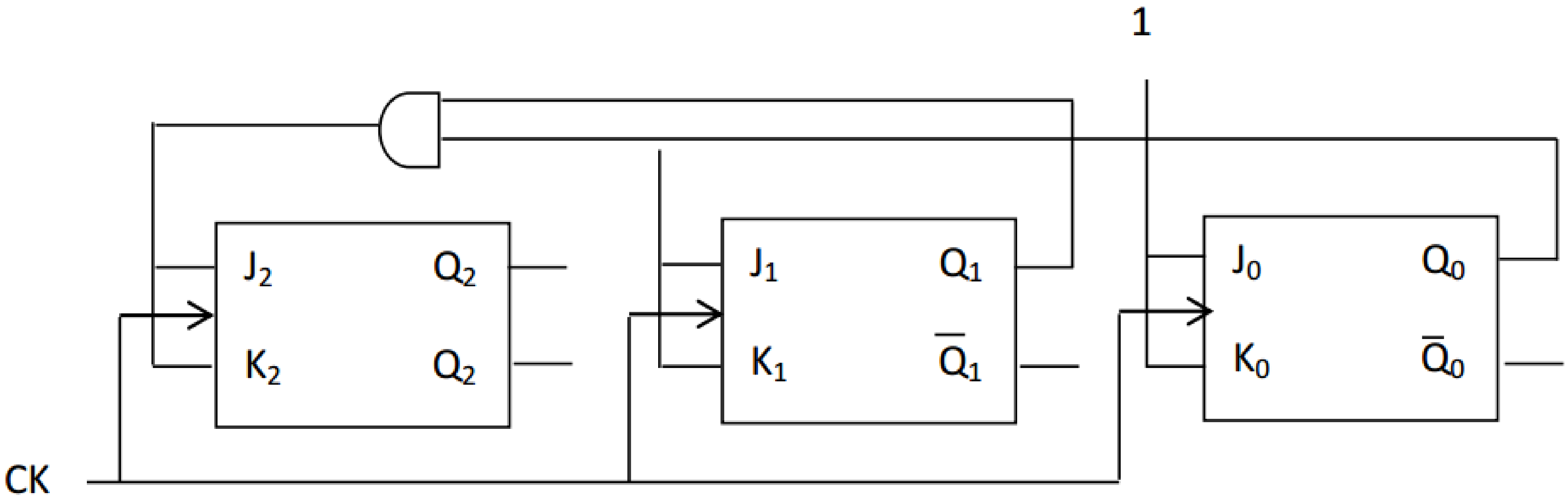
b) Réaliser la Table de Vérité du circuit (le principe est de retrouver les Qi+ à partir des valeurs des équations d'entrée).

c) En déduire le diagramme des états d'où le rôle du circuit.

Le diagramme des états peut être constitué d'une ou plusieurs séquences

# Les circuits séquentiels

**Exemple :**




$$J_2 = Q_0Q_1 \qquad J_1 = Q_0 \qquad J_0 = 1$$

$$K_2 = Q_0Q_1 \qquad K_1 = Q_0 \qquad K_0 = 1$$

# Les circuits séquentiels

**Exemple :**

$J_2 = Q_0Q_1$

$K_2 = Q_0Q_1$

$J_1 = Q_0$

$K_1 = Q_0$

$J_0 = 1$

$K_0 = 1$

| $K$ | $J$ | $Q^+$ |
|---|---|---|
| 0 | 0 | Q |
| 0 | 1 | 1 |
| 1 | 0 | 0 |
| 1 | 1 | $\overline{Q}$ |

**Table de Vérité**

| $Q_2$ $Q_1$ $Q_0$ | $J_2$ $K_2$ | $J_1$ $K_1$ | $J_0$ $K_0$ | $Q_2^+$ $Q_1^+$ $Q_0^+$ |
|---|---|---|---|---|
| 0 0 0 | 0 0 | 0 0 | 1 1 | 0 0 1 |
| 0 0 1 | 0 0 | 1 1 | 1 1 | 0 1 0 |
| 0 1 0 | 0 0 | 0 0 | 1 1 | 0 1 1 |
| 0 1 1 | 1 1 | 1 1 | 1 1 | 1 0 0 |
| 1 0 0 | 0 0 | 0 0 | 1 1 | 1 0 1 |
| 1 0 1 | 0 0 | 1 1 | 1 1 | 1 1 0 |
| 1 1 0 | 0 0 | 0 0 | 1 1 | 1 1 1 |
| 1 1 1 | 1 1 | 1 1 | 1 1 | 0 0 0 |

# Les circuits séquentiels

**Diagramme des états**

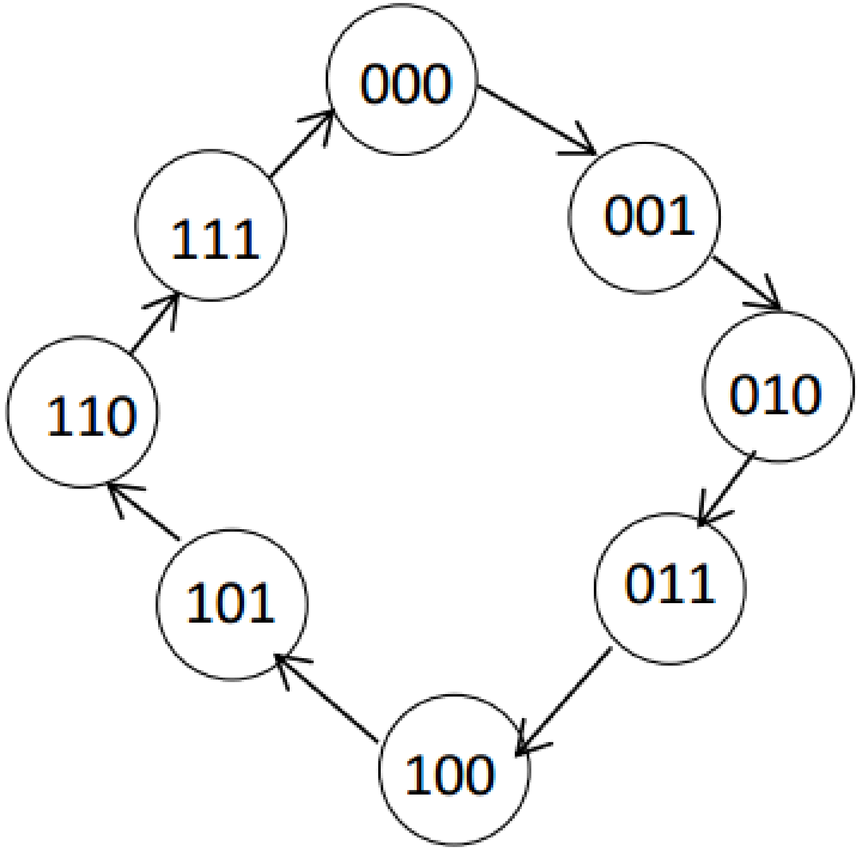


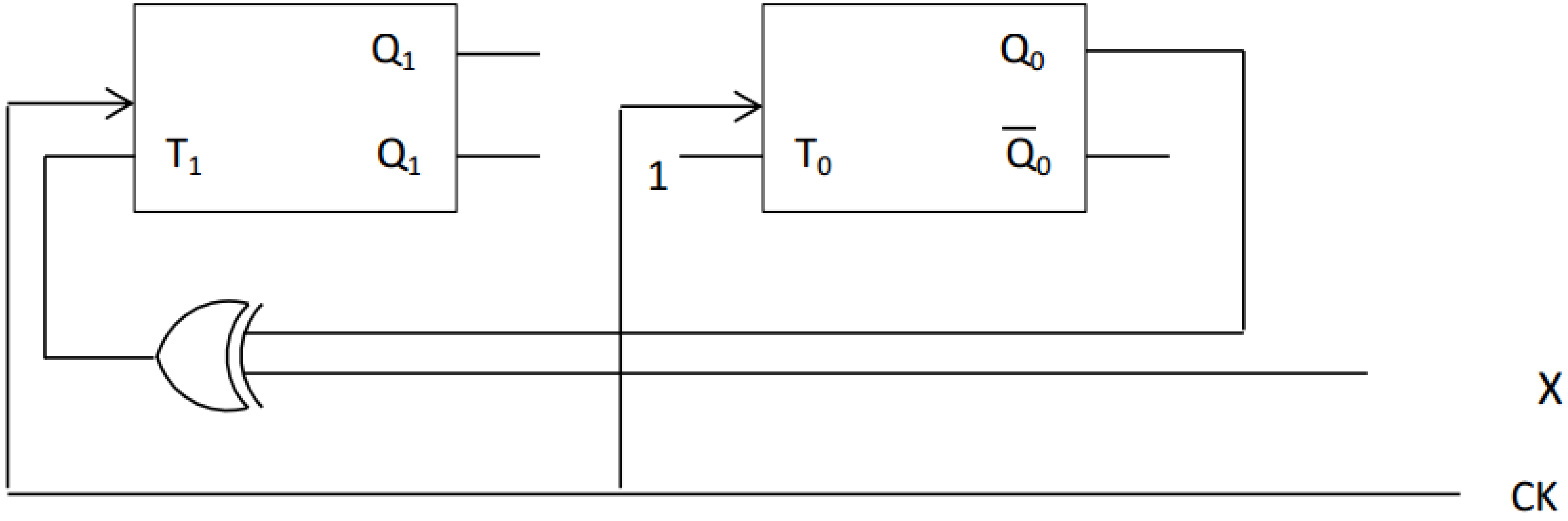

**Table de Vérité**

| $Q_2$ $Q_1$ $Q_0$ | $J_2$ $K_2$ | $J_1$ $K_1$ | $J_0$ $K_0$ | $Q_2^+$ $Q_1^+$ $Q_0^+$ |
|---|---|---|---|---|
| 0 0 0 | 0 0 | 0 0 | 1 1 | 0 0 1 |
| 0 0 1 | 0 0 | 1 1 | 1 1 | 0 1 0 |
| 0 1 0 | 0 0 | 0 0 | 1 1 | 0 1 1 |
| 0 1 1 | 1 1 | 1 1 | 1 1 | 1 0 0 |
| 1 0 0 | 0 0 | 0 0 | 1 1 | 1 0 1 |
| 1 0 1 | 0 0 | 1 1 | 1 1 | 1 1 0 |
| 1 1 0 | 0 0 | 0 0 | 1 1 | 1 1 1 |
| 1 1 1 | 1 1 | 1 1 | 1 1 | 0 0 0 |

Ce circuit représente **un compteur binaire modulo 8**, il compte de 0 à 7

# Les circuits séquentiels

**Exemple 2 :**




$\mathbf{T_1 = Q_0 \oplus x}$ $\mathbf{T_0 = 1}$

<u>Si x = 0</u>

$T_1 = Q_0$ $T_0 = 1$

<u>Si x = 1</u>

$T_1 = \overline{Q_0}$ $T_0 = 1$

# Les circuits séquentiels

**Exemple 2 :**

| | $X$ | $Q_1$ | $Q_0$ | $T_1$ | $T_0$ | $Q_1^+$ | $Q_0^+$ |
|---|---|---|---|---|---|---|---|
| X=0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 |
| | 0 | 0 | 1 | 1 | 1 | 1 | 0 |
| | 0 | 1 | 0 | 0 | 1 | 1 | 1 |
| | 0 | 1 | 1 | 1 | 1 | 0 | 0 |
| X=1 | 1 | 0 | 0 | 1 | 1 | 1 | 1 |
| | 1 | 0 | 1 | 0 | 1 | 0 | 0 |
| | 1 | 1 | 0 | 1 | 1 | 0 | 1 |
| | 1 | 1 | 1 | 0 | 1 | 1 | 0 |

| $T$ | $Q^+$ |
|---|---|
| 0 | $Q$ |
| 1 | $\overline{Q}$ |

$\mathbf{T_1 = Q_0 \oplus x}$ $\mathbf{T_0 = 1}$

<u>Si x = 0</u>

$T_1 = Q_0$ $T_0 = 1$

<u>Si x = 1</u>

$T_1 = \overline{Q_0}$ $T_0 = 1$

# Les circuits séquentiels

**Exemple 2 :**

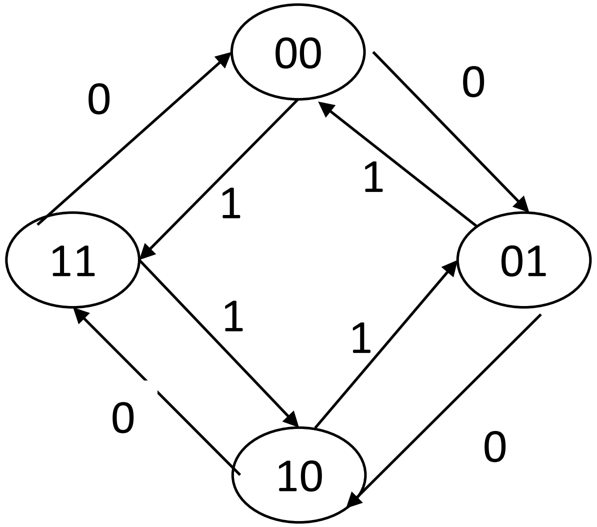
X = 0

00 → 01 → 10 → **11**

X = 1

00 → 11 → 10 → 01

| | X | $Q_1$ | $Q_0$ | $T_1$ | $T_0$ | $Q_1^+$ | $Q_0^+$ |
|---|---|---|---|---|---|---|---|
| X=0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 |
| | 0 | 0 | 1 | 1 | 1 | 1 | 0 |
| | 0 | 1 | 0 | 0 | 1 | 1 | 1 |
| | 0 | 1 | 1 | 1 | 1 | 0 | 0 |
| X=1 | 1 | 0 | 0 | 1 | 1 | 1 | 1 |
| | 1 | 0 | 1 | 0 | 1 | 0 | 0 |
| | 1 | 1 | 0 | 1 | 1 | 0 | 1 |
| | 1 | 1 | 1 | 0 | 1 | 1 | 0 |




Pour X = 0 ce circuit est **un compteur modulo 4** et pour X = 1 c'est **un décompteur modulo 4**

# Les circuits séquentiels

**Tables d’excitation des bascules**

Une table d’excitation (ou table de transition) consiste à retrouver les valeurs d’entrée d’une bascule selon la variation des états de sortie de cette bascule.

# Les circuits séquentiels

## 1. Bascule RS

| $R$ | $S$ | $Q$ | $Q^+$ | |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | R=0 S=0 |
| 0 | 0 | 1 | 1 | R=0 S=0 |
| 0 | 1 | 0 | 1 | R=0 S=1 |
| 0 | 1 | 1 | 1 | R=0 S=1 |
| 1 | 0 | 0 | 0 | R=1 S=0 |
| 1 | 0 | 1 | 0 | R=1 S=0 |
| 1 | 1 | 0 | X | |
| 1 | 1 | 1 | X | |

**Table d'excitation (R S)**

| $Q$ | $Q^+$ | $R$ | $S$ |
|---|---|---|---|
| 0 | 0 | X | 0 |
| 0 | 1 | 0 | 1 |
| 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | X |

# Les circuits séquentiels

## 1. Bascule JK

| $K$ | $J$ | $Q$ | $Q^+$ |
|---|---|---|---|
| 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 |
| 0 | 1 | 0 | 1 |
| 0 | 1 | 1 | 1 |
| 1 | 0 | 0 | 0 |
| 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 0 |

J = 0 K=0

J = 1 K=0

**Table d'excitation (J K)**

| $Q$ | $Q^+$ | $J$ | $K$ |
|---|---|---|---|
| 0 | 0 | 0 | X |
| 0 | 1 | 1 | X |
| 1 | 0 | X | 1 |
| 1 | 1 | X | 0 |

# Les circuits séquentiels

## 1. Bascule T

| $T$ | $Q$ | $Q^+$ |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

**Table d'excitation (T)**

| $Q$ | $Q^+$ | $T$ |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

# Les circuits séquentiels

## 1. Bascule D

| $D$ | $Q$ | $Q^+$ |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 1 |
| 1 | 1 | 1 |

**Table d'excitation (D)**

| $Q$ | $Q^+$ | $D$ |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

# Les circuits séquentiels

**Synthèse d'un circuit séquentiel**

La synthèse d'un circuit séquentiel consiste à retrouver les fonctions d'entrée de ce circuit à partir d'un diagramme d'états (séquence). Pour cela il faut :

a) Etablir le diagramme des états (ou séquence) et donner le nombre de bascules nécessaires.

b) Réaliser **la table de transition**

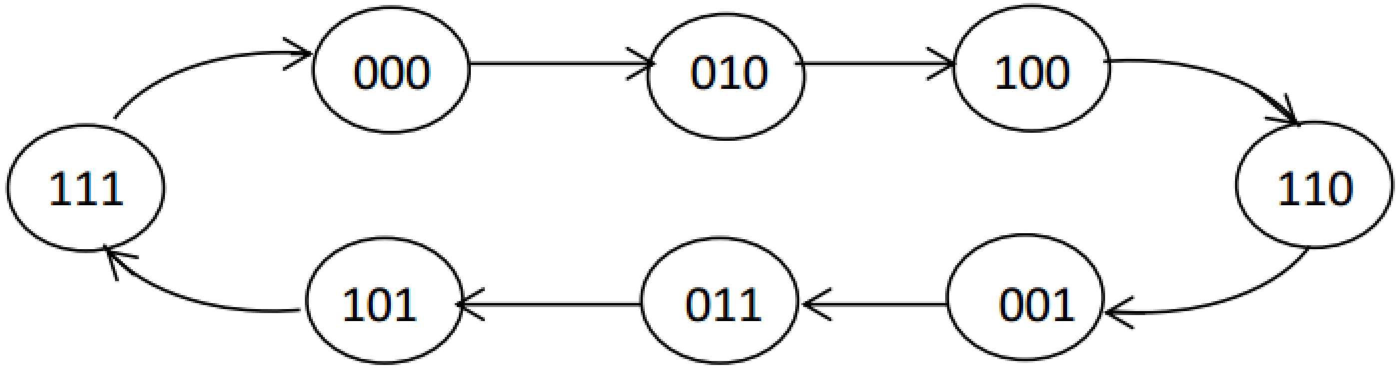
c) En déduire les équations d'entrée aux bascules

d) Réaliser le circuit correspondant

# Les circuits séquentiels

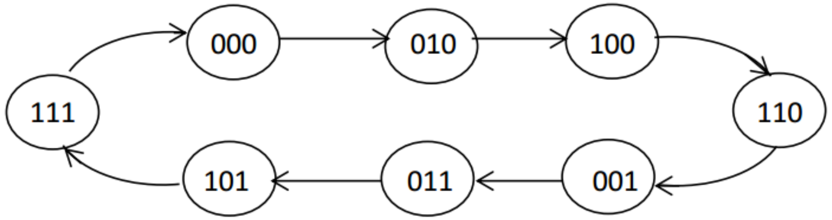
**Exemple :** $0 \rightarrow 2 \rightarrow 4 \rightarrow 6 \rightarrow 1 \rightarrow 3 \rightarrow 5 \rightarrow 7$

**Séquence**



# Les circuits séquentiels

**Table de transition**




| $Q$ | $Q^+$ | $J$ | $K$ |
|---|---|---|---|
| 0 | 0 | 0 | X |
| 0 | 1 | 1 | X |
| 1 | 0 | X | 1 |
| 1 | 1 | X | 0 |

| $Q_2$ $Q_1$ $Q_0$ | $Q_2^+$ $Q_1^+$ $Q_0^+$ | $J_2$ $K_2$ | $J_1$ $K_1$ | $J_0$ $K_0$ |
|---|---|---|---|---|
| 0 0 0 | 0 1 0 | 0 X | 1 X | 0 X |
| 0 0 1 | 0 1 1 | 0 X | 1 X | X 0 |
| 0 1 0 | 1 0 0 | 1 X | X 1 | 0 X |
| 0 1 1 | 1 0 1 | 1 X | X 1 | X 0 |
| 1 0 0 | 1 1 0 | X 0 | 1 X | 0 X |
| 1 0 1 | 1 1 1 | X 0 | 1 X | X 0 |
| 1 1 0 | 0 0 1 | X 1 | X 1 | 1 X |
| 1 1 1 | 0 0 0 | X 1 | X 1 | X 1 |

## Table de transition

| $Q_2$ $Q_1$ $Q_0$ | $Q_2^+$ $Q_1^+$ $Q_0^+$ | $J_2$ $K_2$ | $J_1$ $K_1$ | $J_0$ $K_0$ |
|---|---|---|---|---|
| 0 0 0 | 0 1 0 | 0 X | 1 X | 0 X |
| 0 0 1 | 0 1 1 | 0 X | 1 X | X 0 |
| 0 1 0 | 1 0 0 | 1 X | X 1 | 0 X |
| 0 1 1 | 1 0 1 | 1 X | X 1 | X 0 |
| 1 0 0 | 1 1 0 | X 0 | 1 X | 0 X |
| 1 0 1 | 1 1 1 | X 0 | 1 X | X 0 |
| 1 1 0 | 0 0 1 | X 1 | X 1 | 1 X |
| 1 1 1 | 0 0 0 | X 1 | X 1 | X 1 |

$J_2 = Q_1$ $\quad J_1 = 1$ $\quad J_0 = Q_2Q_1$

$K_2 = Q_1$ $\quad K_1 = 1$ $\quad K_0 = Q_2Q_1$

| $Q_2$ \ $Q_1Q_0$ | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 0 | 0 | 0 | 1 | 1 |
| 1 | X | X | X | X |

$J_2 = Q_1$

| $Q_2$ \ $Q_1Q_0$ | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 0 | X | X | X | X |
| 1 | 0 | 0 | 1 | 1 |

$K_2 = Q_1$

| $Q_2$ \ $Q_1Q_0$ | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 0 | 0 | X | 0 | X |
| 1 | 0 | X | X | 1 |

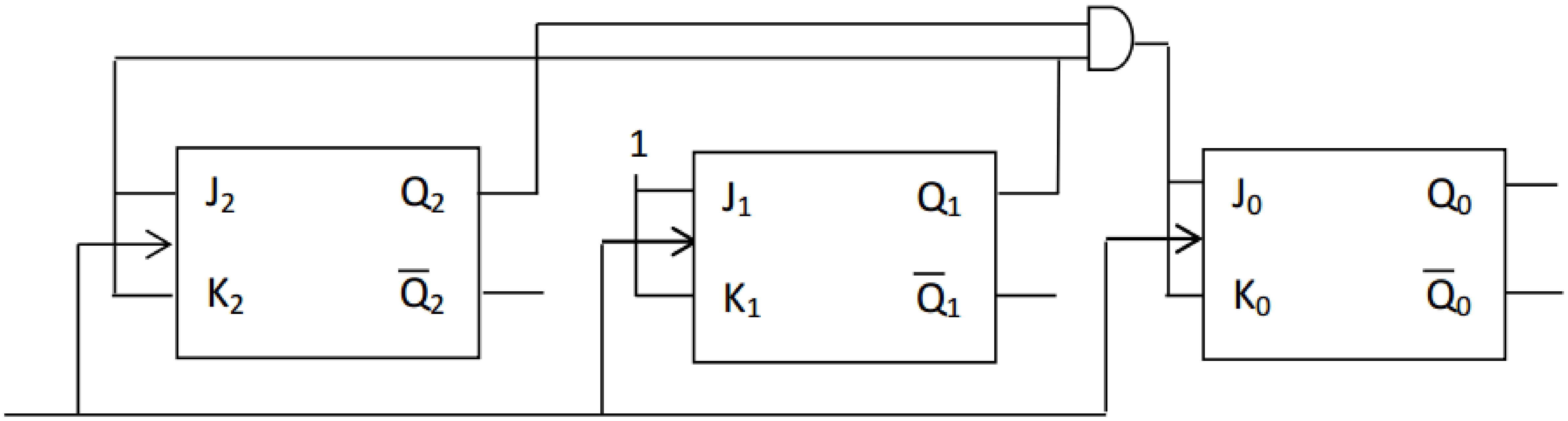
$J_0 = Q_2Q_1$

| $Q_2$ \ $Q_1Q_0$ | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 0 | X | 0 | 0 | X |
| 1 | X | 0 | 1 | X |

$K_0 = Q_2Q_1$

# Les circuits séquentiels

**Circuit :**



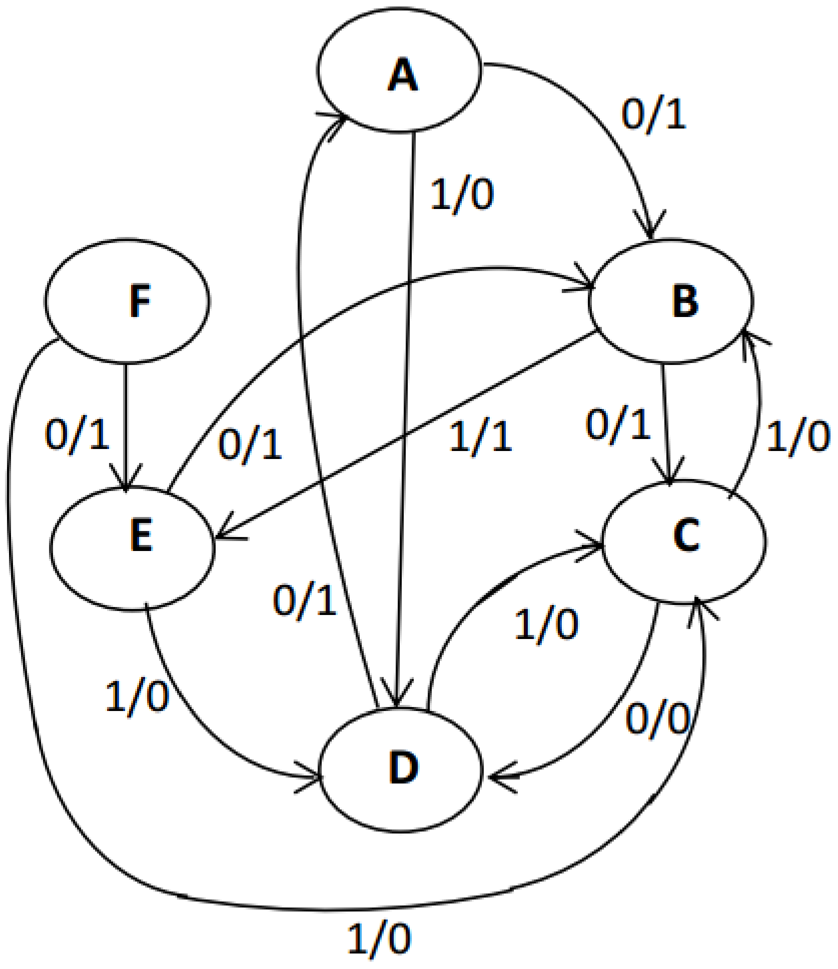
# Les circuits séquentiels

**Exemple 2 (Diagramme avec états équivalents)**

Dans un diagramme d'états, il arrive parfois que deux ou plusieurs états soient équivalents, dans ce cas il faut le simplifier. Deux états à l'instant (t) sont équivalents s'ils ont le même état suivant (t+1) avec les mêmes variables de contrôle (entrée et sortie)

Pour simplifier le diagramme, on le représente sous forme tabulaire

Diagramme des états initial



# Les circuits séquentie

**Exemple 2 (Diagramme avec états équivalents)**

Pour simplifier le diagramme, on le représente sous forme tabulaire

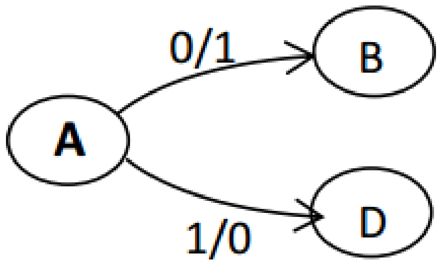
On voit que pour les mêmes variables d’entrée/sortie A et E ont les mêmes états suivants donc ils sont équivalents ; Il faut alors supprimer un des 2.

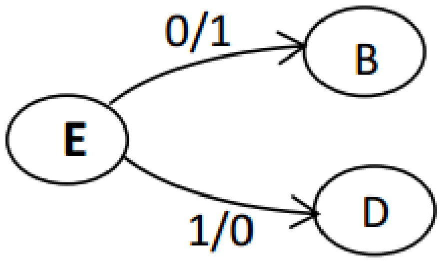
Dans cet exemple on supprime E et on remplace tous les E du tableau par A

Représentation sous forme tabulaire

| Etat (t) | X | Y | Etat (t+1) |
|---|---|---|---|
| A | 0 | 1 | B |
| A | 1 | 0 | D |
| B | 0 | 1 | C |
| B | 1 | 1 | ~~E~~ A |
| C | 0 | 0 | D |
| C | 1 | 0 | B |
| D | 0 | 1 | A |
| D | 1 | 0 | C |
| E | 0 | 1 | B |
| E | 1 | 0 | D |
| F | 0 | 1 | ~~E~~ A |
| F | 1 | 0 | C |

| Etat (t) | X | Y | Etat (t+1) |
|---|---|---|---|
| A | 0 | 1 | B |
| A | 1 | 0 | D |
| | | | |
| E | 0 | 1 | B |
| E | 1 | 0 | D |



# Les circuits séquentiels

**Exemple 2 (Diagramme avec états équivalents)**

On a également les états **D** et **F** qui sont équivalents, on supprime donc **F** et

on obtient un nouveau tableau et un nouveau diagramme

| Etat (t) | X | Y | Etat (t+1) |
|---|---|---|---|
| A | 0 | 1 | B |
| A | 1 | 0 | D |
| B | 0 | 1 | C |
| B | 1 | 1 | A |
| C | 0 | 0 | D |
| C | 1 | 0 | B |
| D | 0 | 1 | A |
| D | 1 | 0 | C |

Tableau simplifié

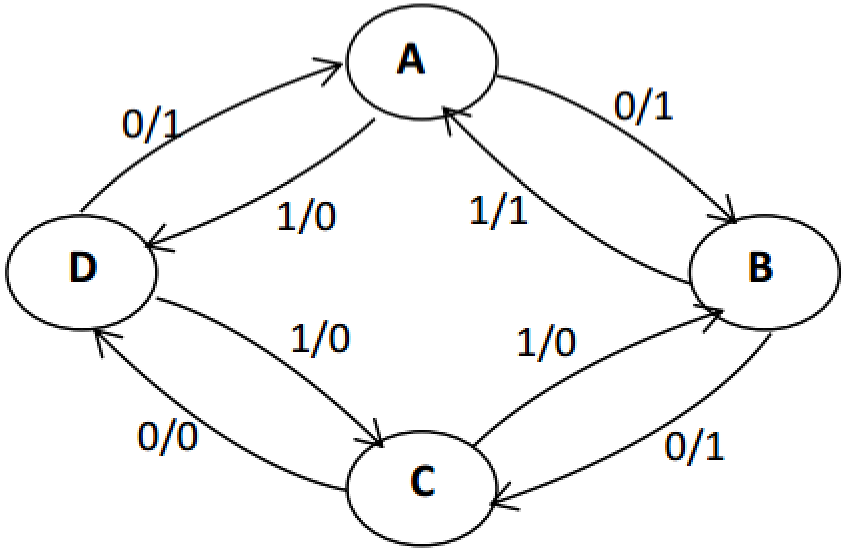



Diagramme des états simplifié

# Les circuits séquentiels

**Exemple 2 (Diagramme avec états équivalents)**

Pour réaliser le circuit il faut codifier les états. Généralement on utilise des codes binaires croissant selon l’ordre alphabétique.

A = 00 B = 01 C = 10 D = 11

Max (A,B,C,D) = D = 11

La plus grande valeur s’écrit sur 2 bits donc il faut 2 bascules pour réaliser ce circuit.

# Les circuits séquentiels

**Exemple 2 (Diagramme avec états équivalents)**

Table de transition :

| $X$ | $Q_1$ | $Q_0$ | $Q_1^+$ | $Q_0^+$ | $T_1$ | $T_0$ | $Y$ |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 |
| 0 | 0 | 1 | 1 | 0 | 1 | 1 | 1 |
| 0 | 1 | 0 | 1 | 1 | 0 | 1 | 0 |
| 0 | 1 | 1 | 0 | 0 | 1 | 1 | 1 |
| 1 | 0 | 0 | 1 | 1 | 1 | 1 | 0 |
| 1 | 0 | 1 | 0 | 0 | 0 | 1 | 1 |
| 1 | 1 | 0 | 0 | 1 | 1 | 1 | 0 |
| 1 | 1 | 1 | 1 | 0 | 0 | 1 | 0 |

| $Q_1Q_0$ \ $X$ | 0 0 | 0 1 | 1 1 | 1 0 |
|---|---|---|---|---|
| 0 | 0 | 1 | 1 | 0 |
| 1 | 1 | 0 | 0 | 1 |

$$T_1 = X \oplus Q_0$$

$$T_0 = 1$$

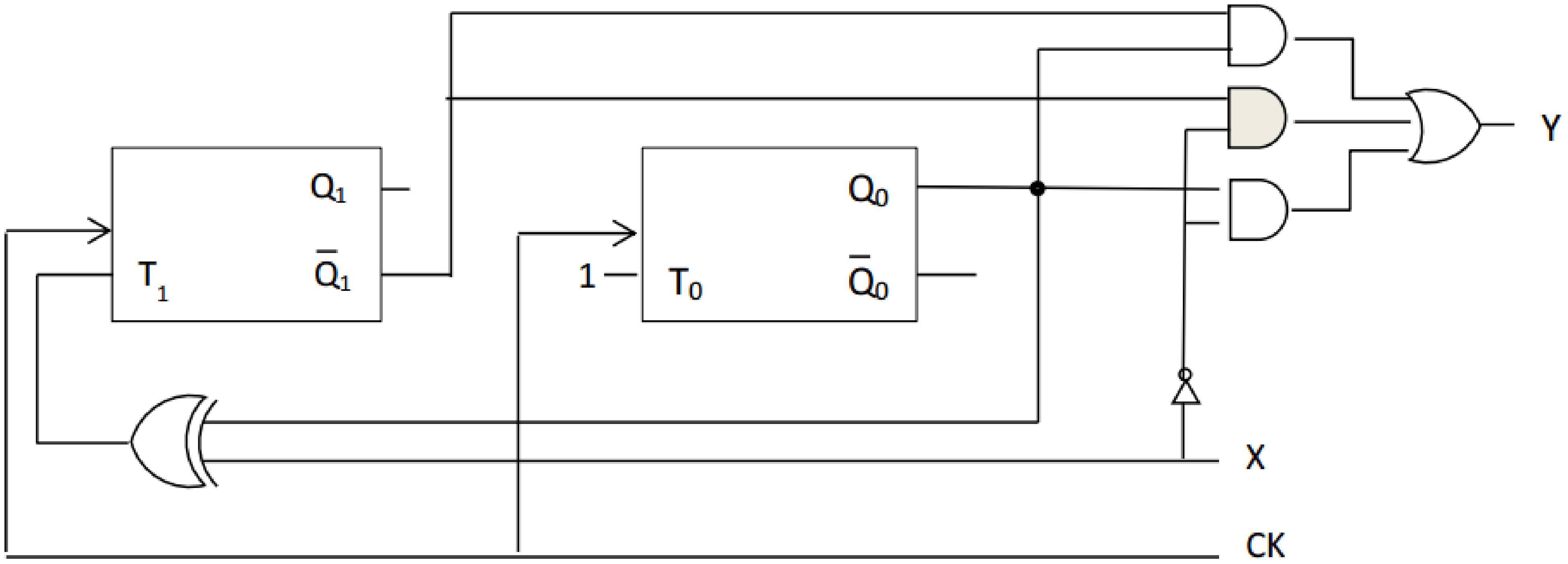
| $Q_1Q_0$ \ $X$ | 0 0 | 0 1 | 1 1 | 1 0 |
|---|---|---|---|---|
| 0 | 1 | 1 | 1 | 0 |
| 1 | 0 | 1 | 0 | 0 |

$$Y = \overline{X}\,\overline{Q_1} + \overline{X}\,Q_0 + \overline{Q_1}\,Q_0$$

# Les circuits séquentiels

**Exemple 2 (Diagramme avec états équivalents)**

<u>**Circuit correspondant :**</u>



# Chronogrammes

**Le chronogramme** est une représentation graphique de l'évolution de l’état d’un circuit dans le temps. Il utilise une horloge de période T.
Il existe deux types de circuits séquentiels, les circuits synchrones et les circuits asynchrones :

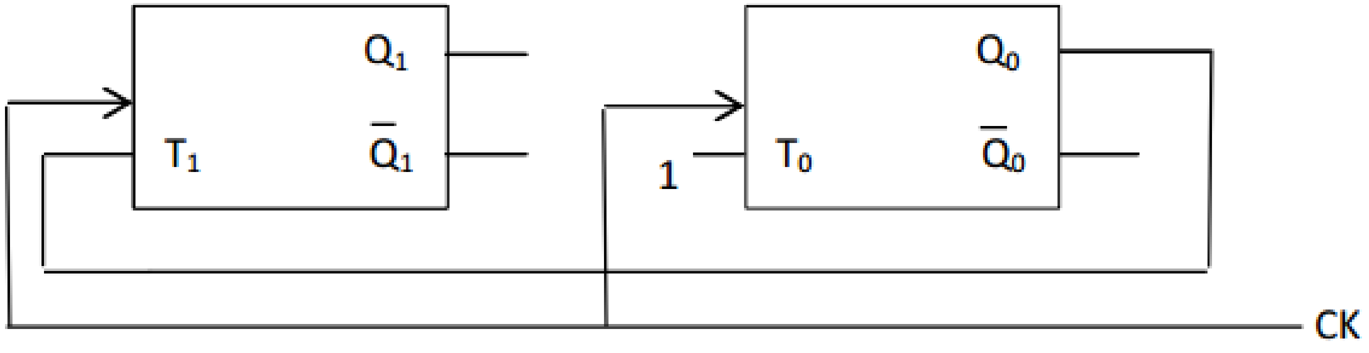
- **Dans un circuit synchrone**, toutes les bascules sont reliées à la même horloge.
- **Dans un circuit asynchrone**, les bascules n’ont pas la même horloge.

- Généralement on retrouve la séquence d’un circuit à l’aide d’une table de vérité
- mais on peut aussi retrouver directement celle – ci à l’aide d’un **chronogramme.**

# Chronogrammes

- Pour réaliser le chronogramme d'un circuit il faut :
  a) Donner l'état initial de chaque bascule à l'instant (t)
  b) En déduire les valeurs des entrées pour chaque bascule à l'instant (t)
  c) A partir de ces entrées, retrouver l'état de chaque bascule à l'instant (t+1)
  d) (t+1) devient (t) et on recommence jusqu'à ce qu'on retrouve l'état initial

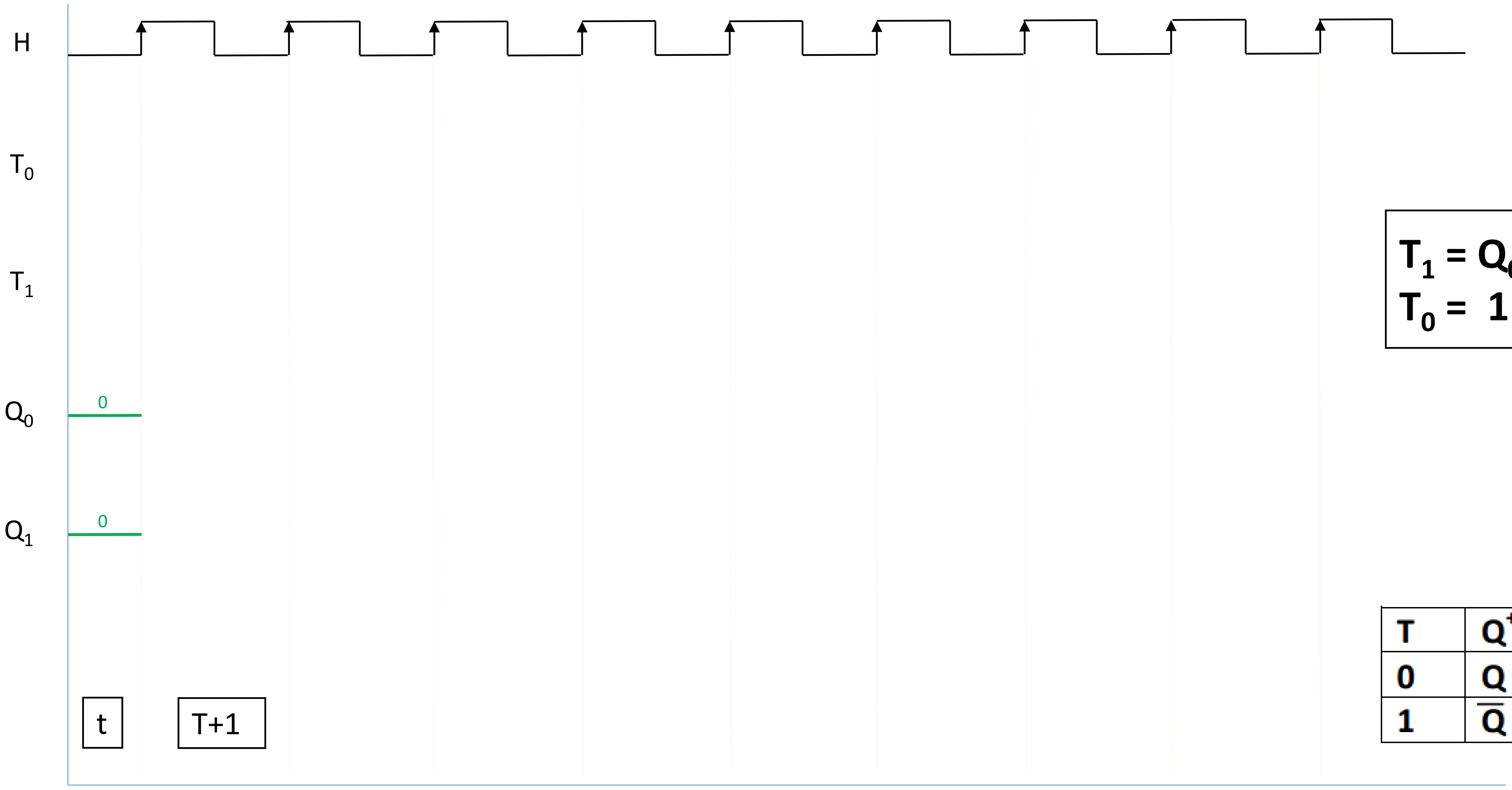
- **Exemple 1 :** **circuit synchrone**




$T0 = 1$

$T1 = Q_0$

# Exemple 1

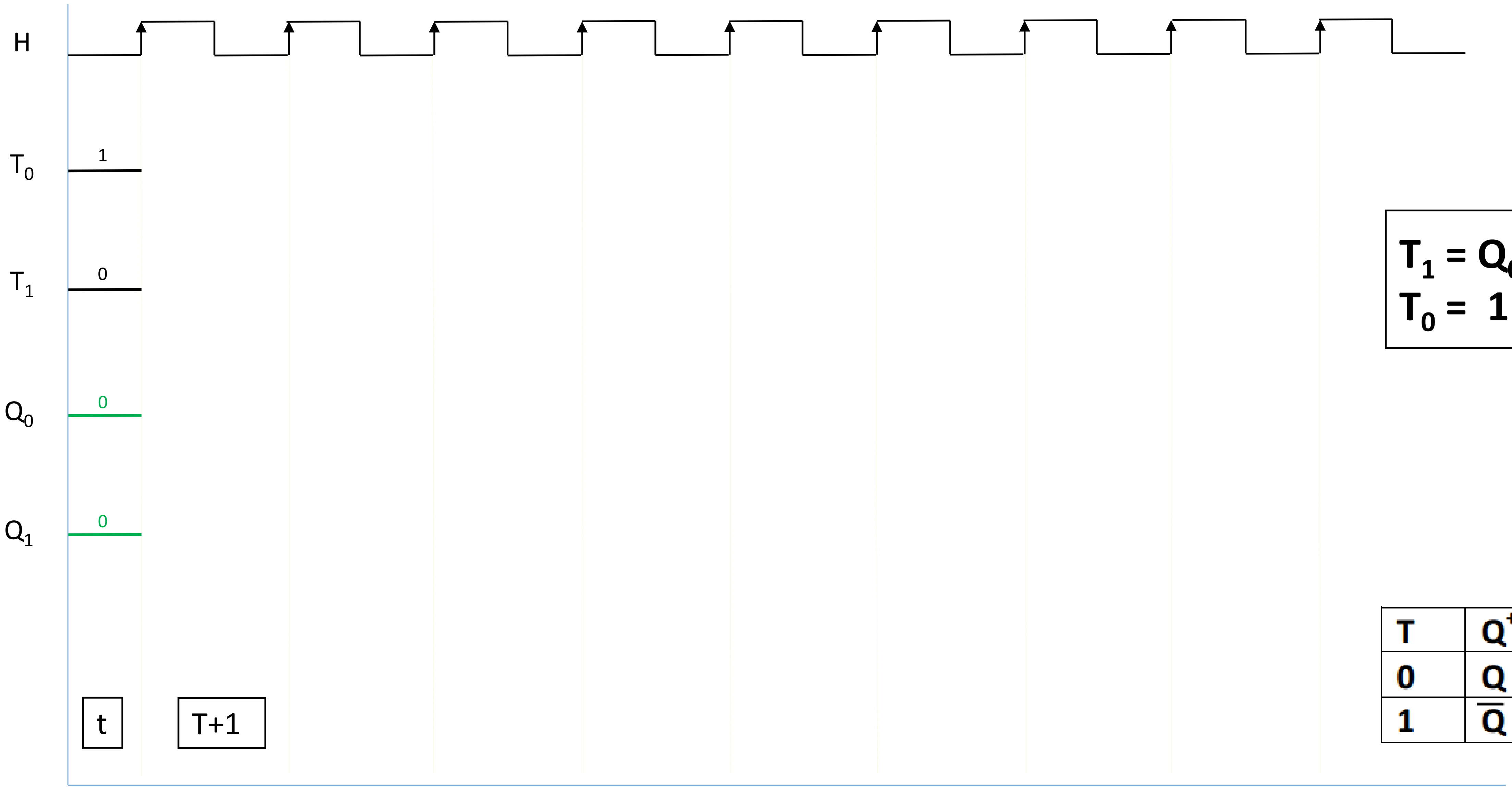



$T_1 = Q_0$

$T_0 = 1$

| T | $Q^+$ |
|---|---|
| 0 | $Q$ |
| 1 | $\overline{Q}$ |

# Exemple 1

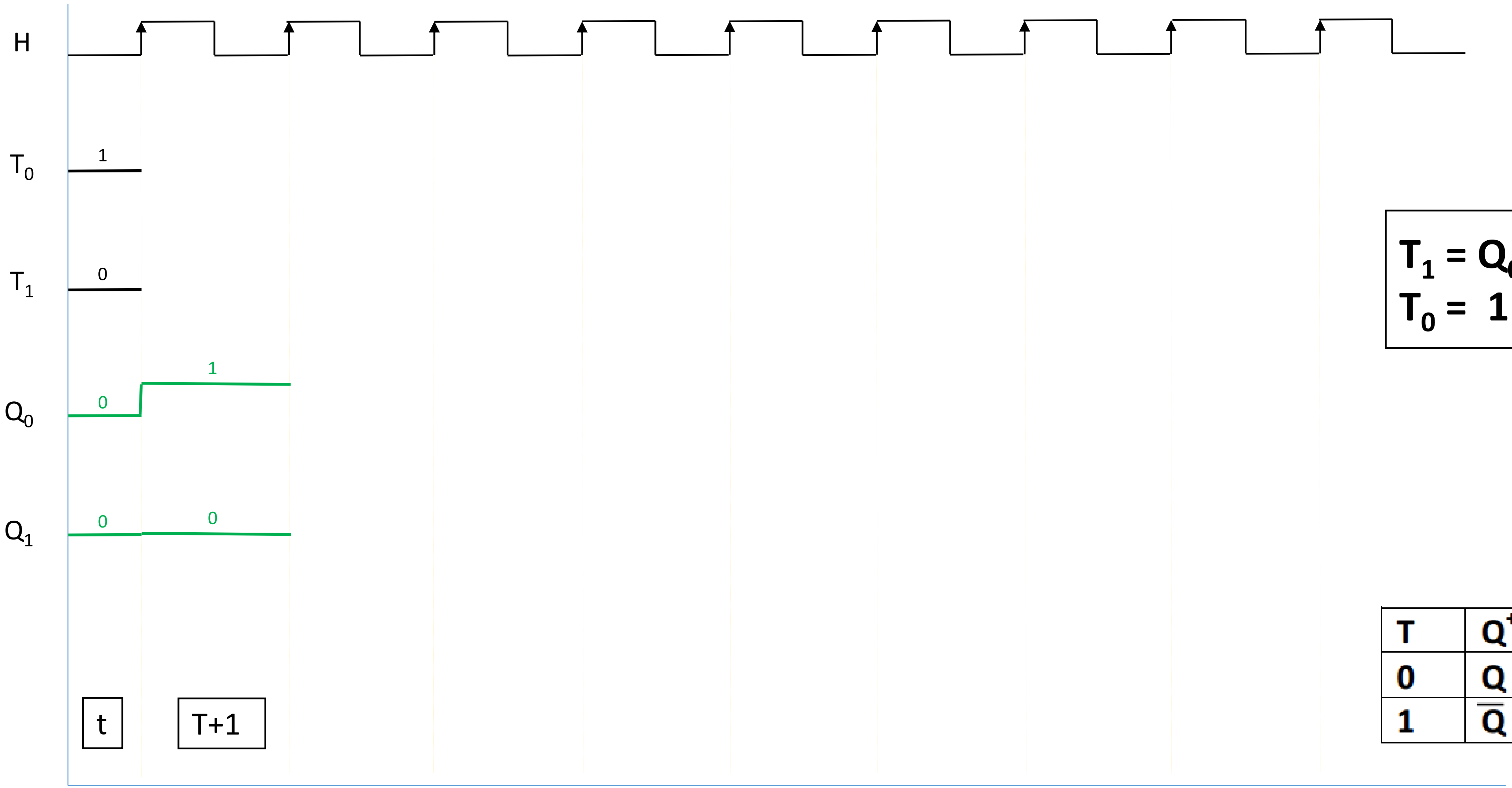



$T_1 = Q_0$

$T_0 = 1$

| T | $Q^+$ |
|---|---|
| 0 | $Q$ |
| 1 | $\overline{Q}$ |

# Exemple 1

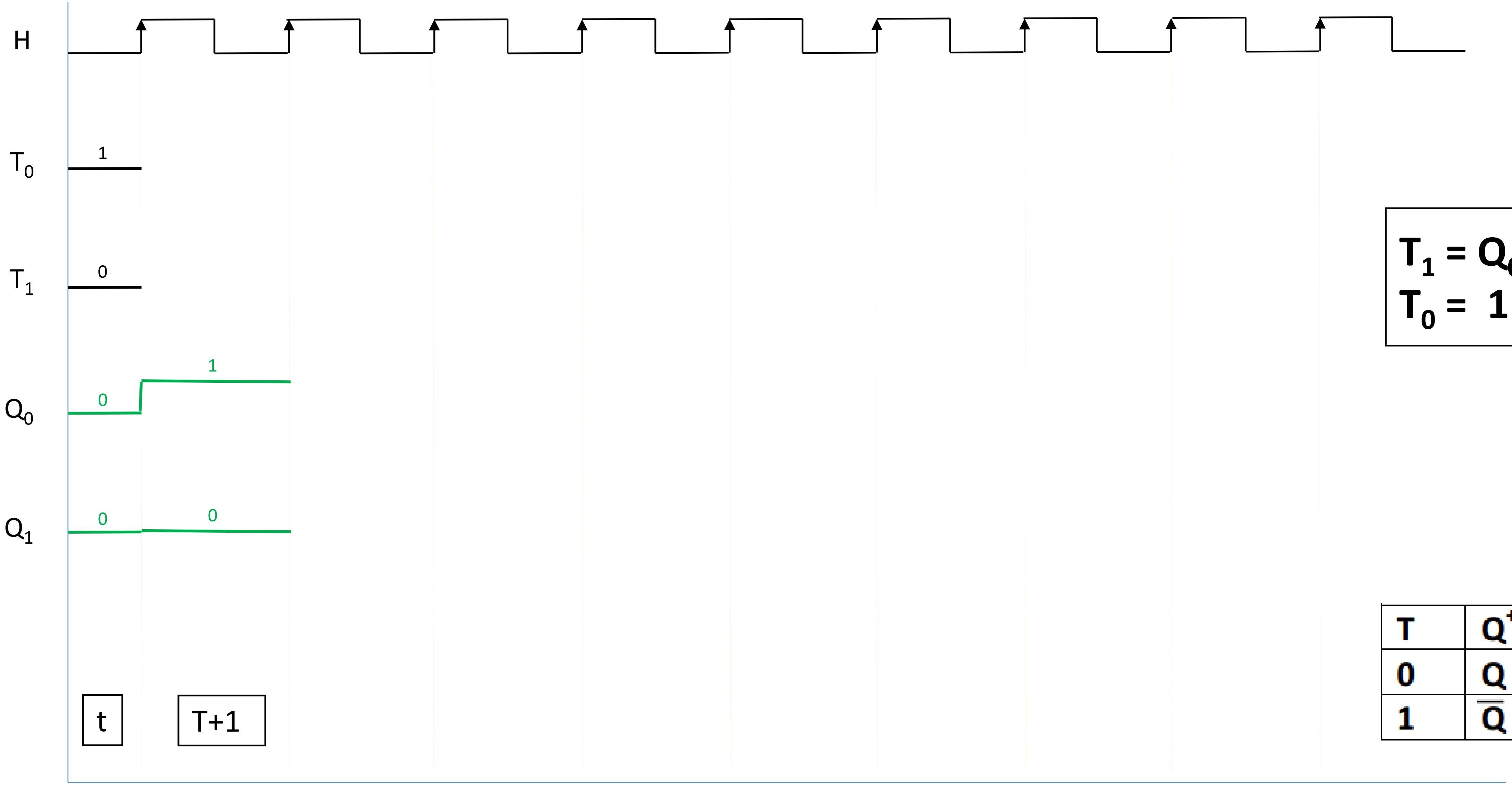



$T_1 = Q_0$

$T_0 = 1$

| T | $Q^+$ |
|---|---|
| 0 | $Q$ |
| 1 | $\overline{Q}$ |

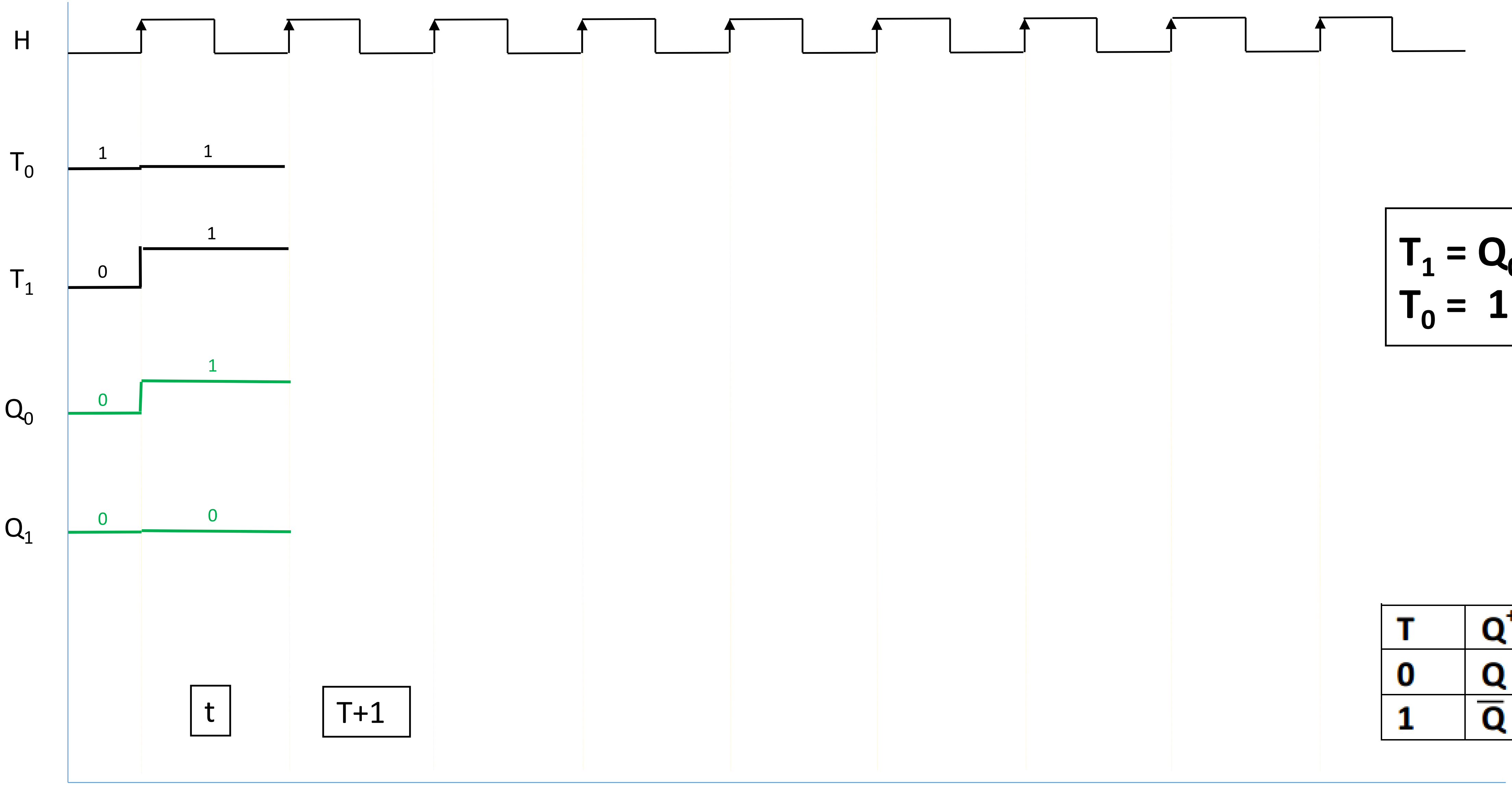
# Exemple 1




$$T_1 = Q_0$$
$$T_0 = 1$$

| T | $Q^+$ |
|---|---|
| 0 | $Q$ |
| 1 | $\overline{Q}$ |

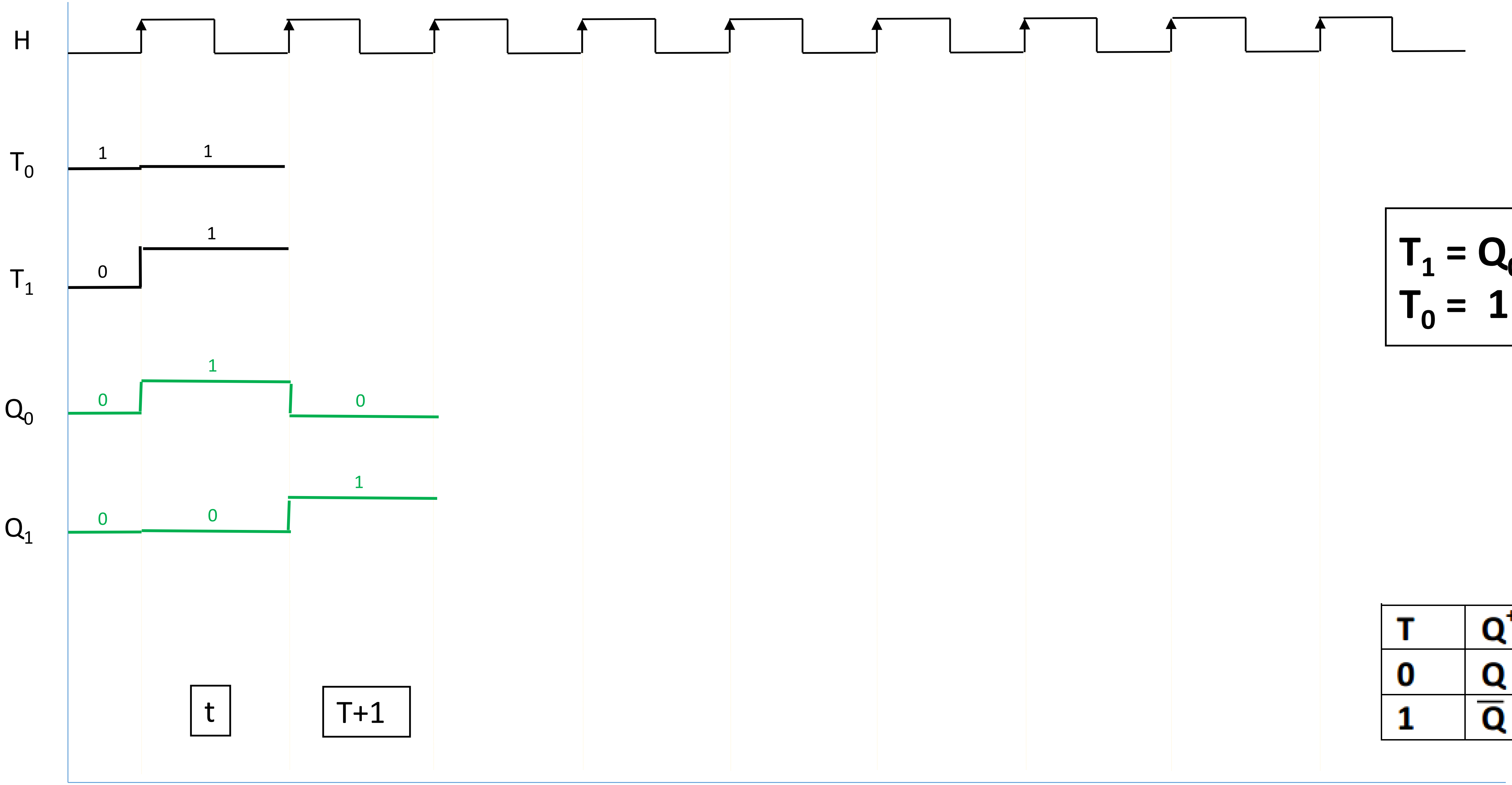
# Exemple 1




$T_1 = Q_0$
$T_0 = 1$

| T | $Q^+$ |
|---|---|
| 0 | Q |
| 1 | $\overline{Q}$ |

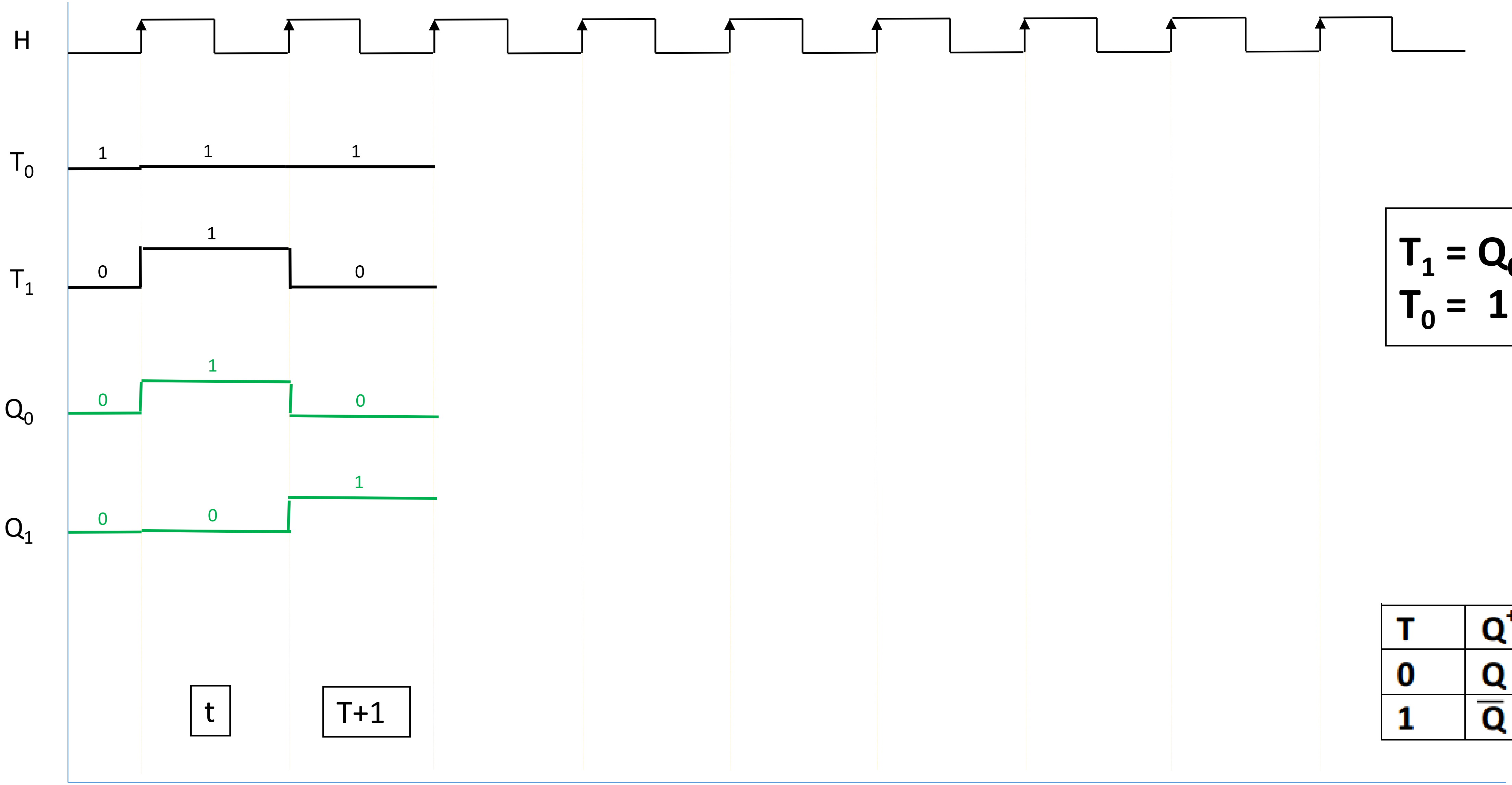
# Exemple 1




$T_1 = Q_0$
$T_0 = 1$

| T | $Q^+$ |
|---|---|
| 0 | $Q$ |
| 1 | $\overline{Q}$ |

# Exemple 1

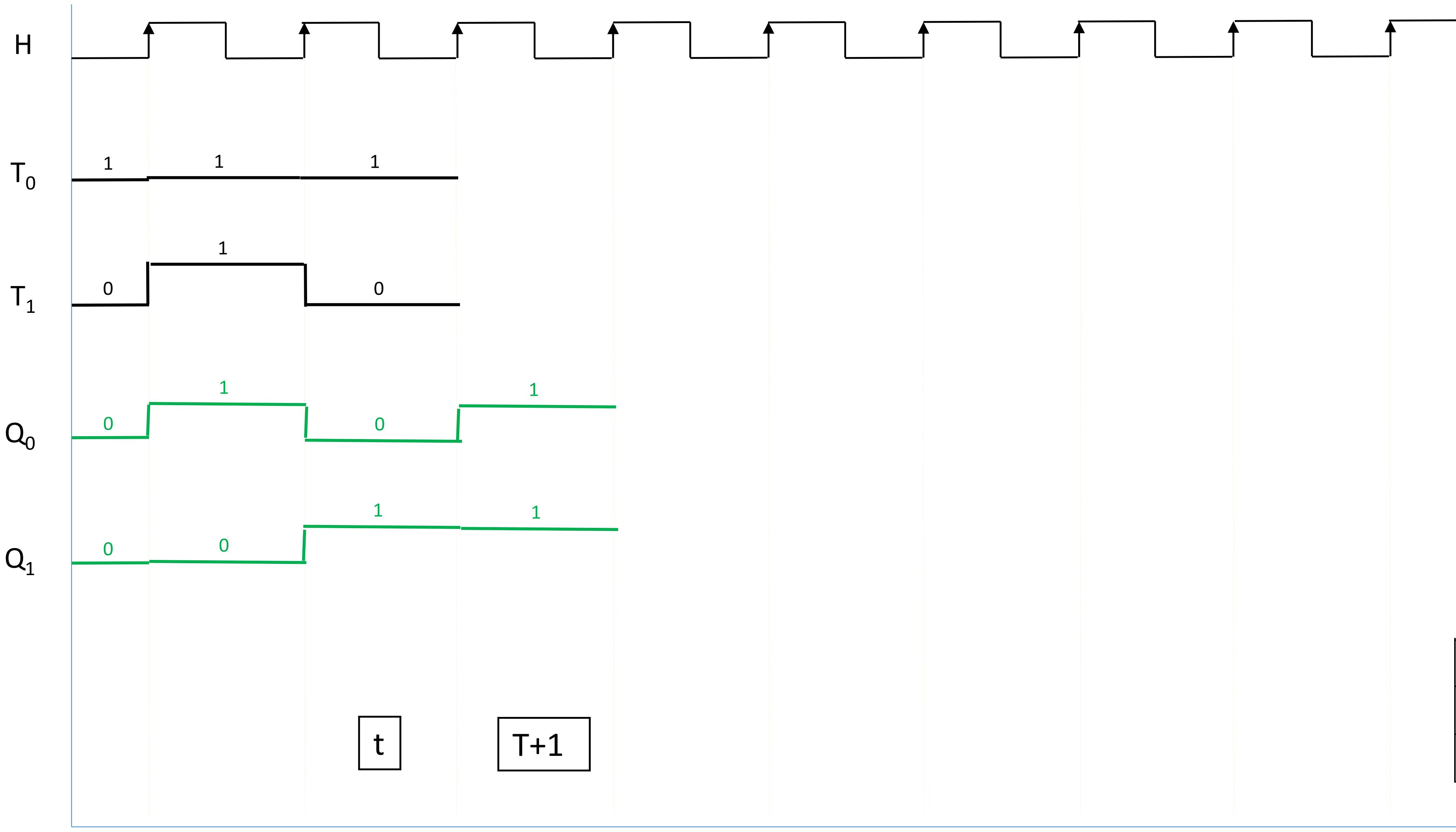



$T_1 = Q_0$

$T_0 = 1$

| T | $Q^+$ |
|---|---|
| 0 | $Q$ |
| 1 | $\overline{Q}$ |

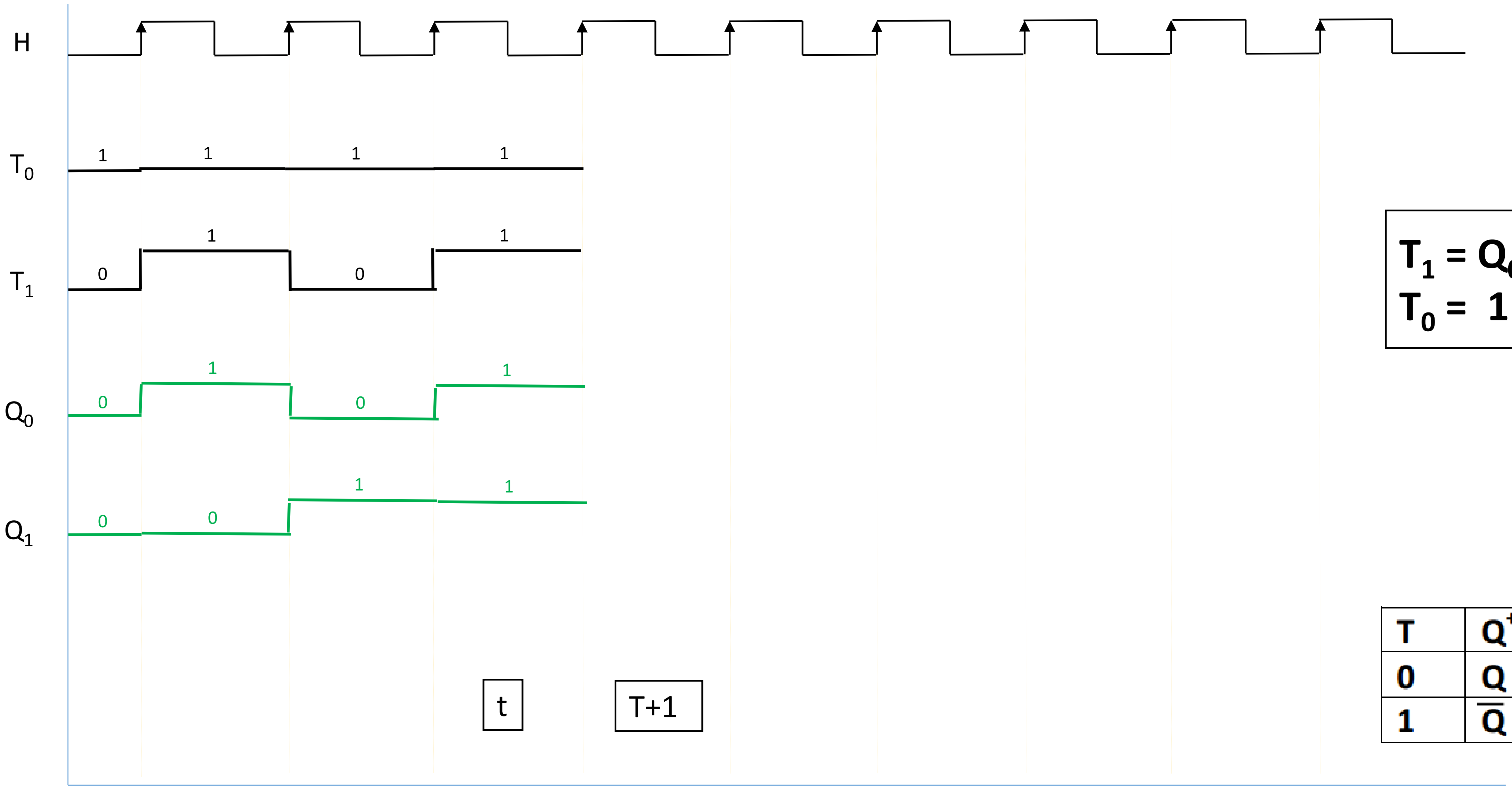
# Exemple 1




$$T_1 = Q_0$$
$$T_0 = 1$$

| T | $Q^+$ |
|---|---|
| 0 | $Q$ |
| 1 | $\overline{Q}$ |

# Exemple 1

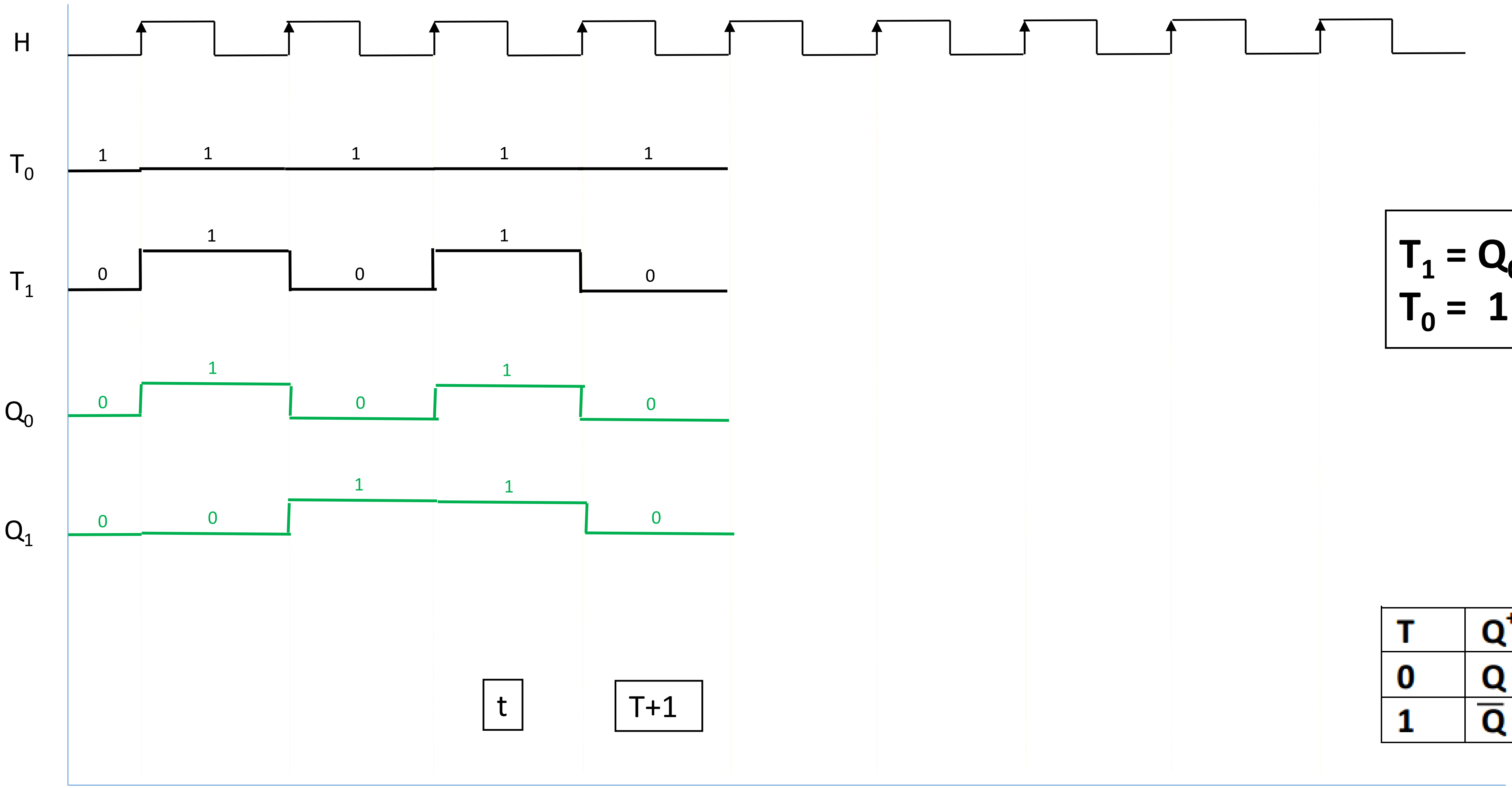



$T_1 = Q_0$

$T_0 = 1$

| T | $Q^+$ |
|---|---|
| 0 | $Q$ |
| 1 | $\overline{Q}$ |

# Exemple 1

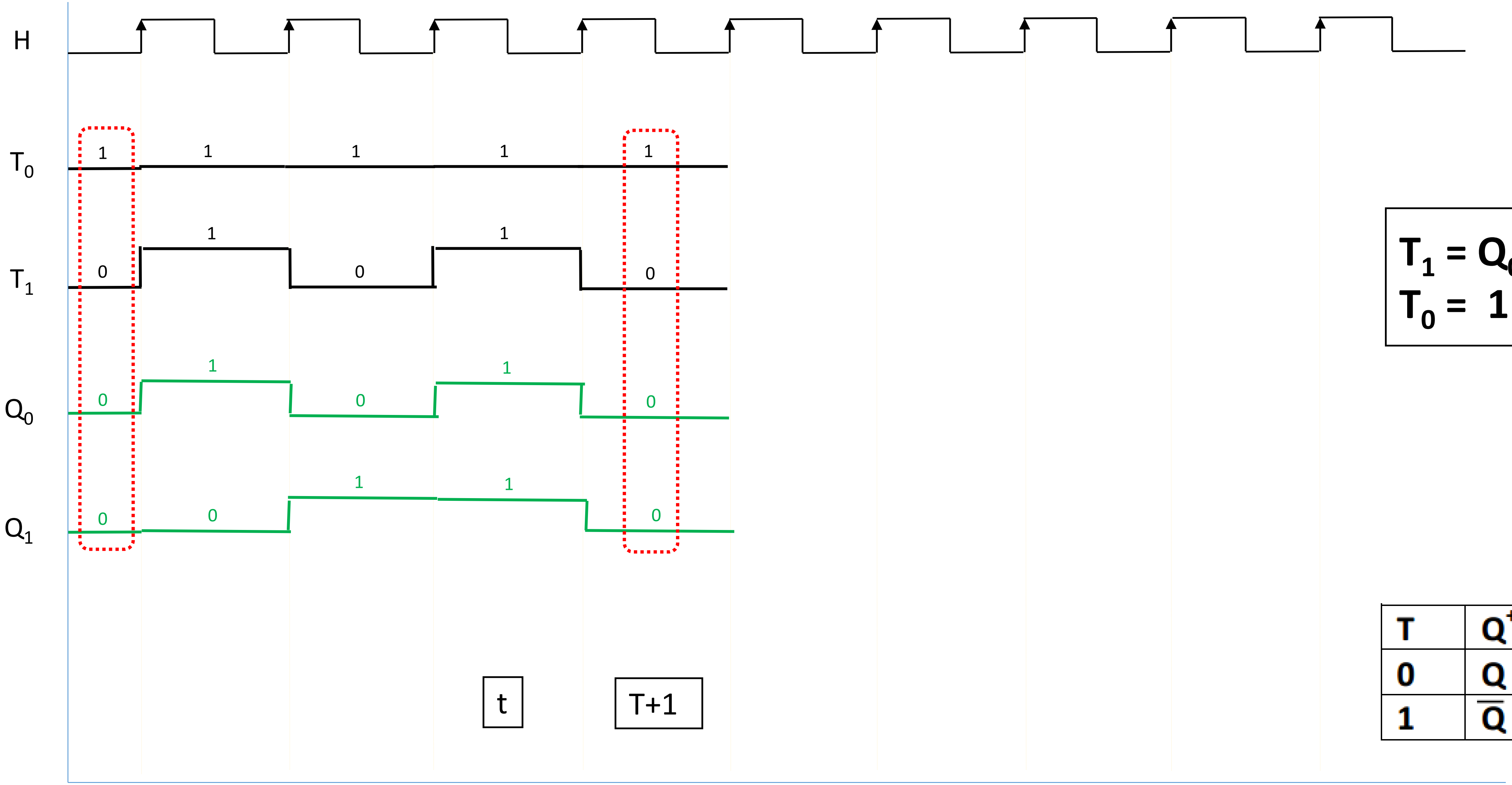



$T_1 = Q_0$

$T_0 = 1$

| $T$ | $Q^+$ |
|---|---|
| 0 | $Q$ |
| 1 | $\overline{Q}$ |

# Exemple 1

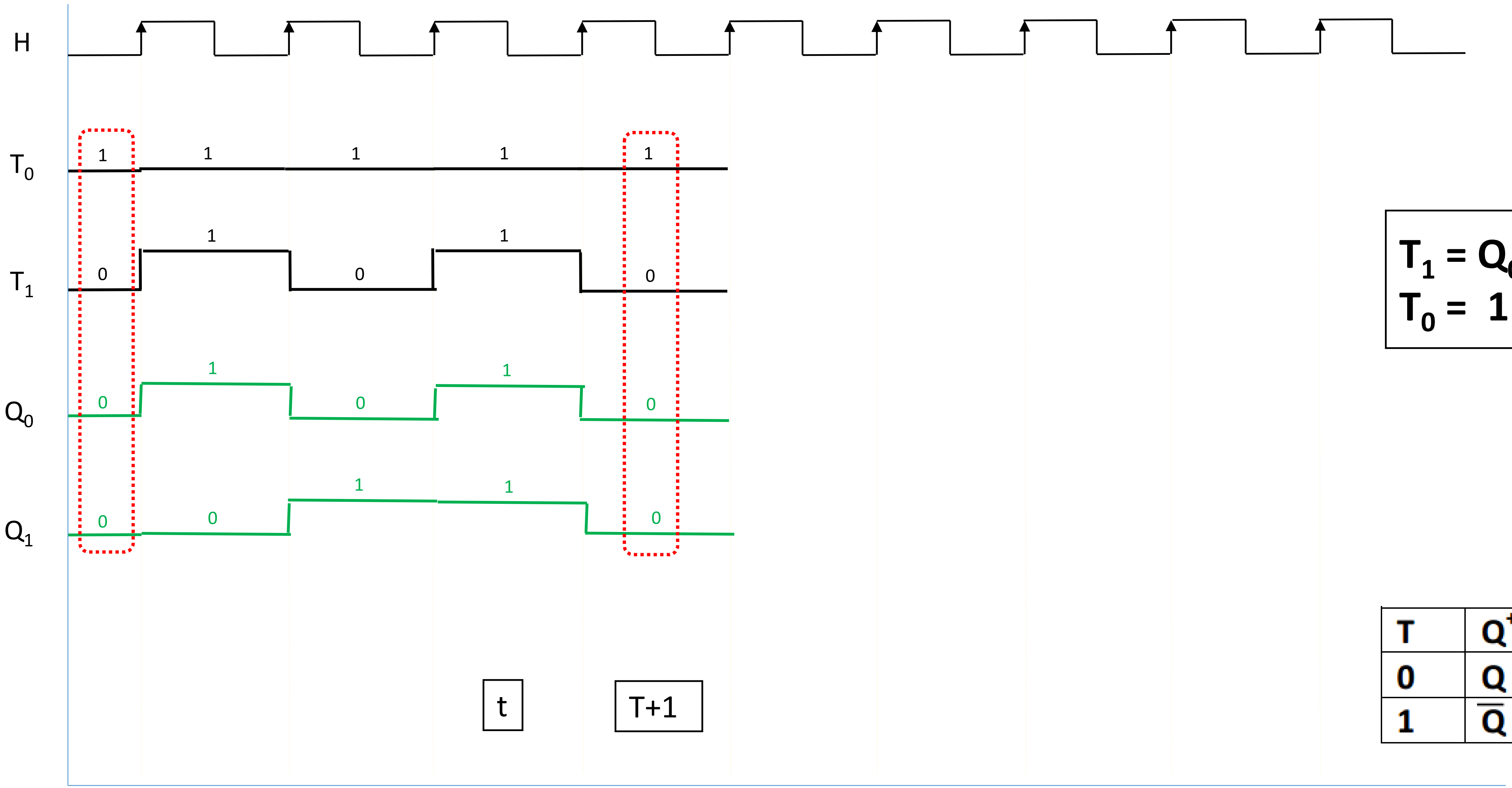



$T_1 = Q_0$

$T_0 = 1$

| T | $Q^+$ |
|---|---|
| 0 | $Q$ |
| 1 | $\overline{Q}$ |

# Exemple 1

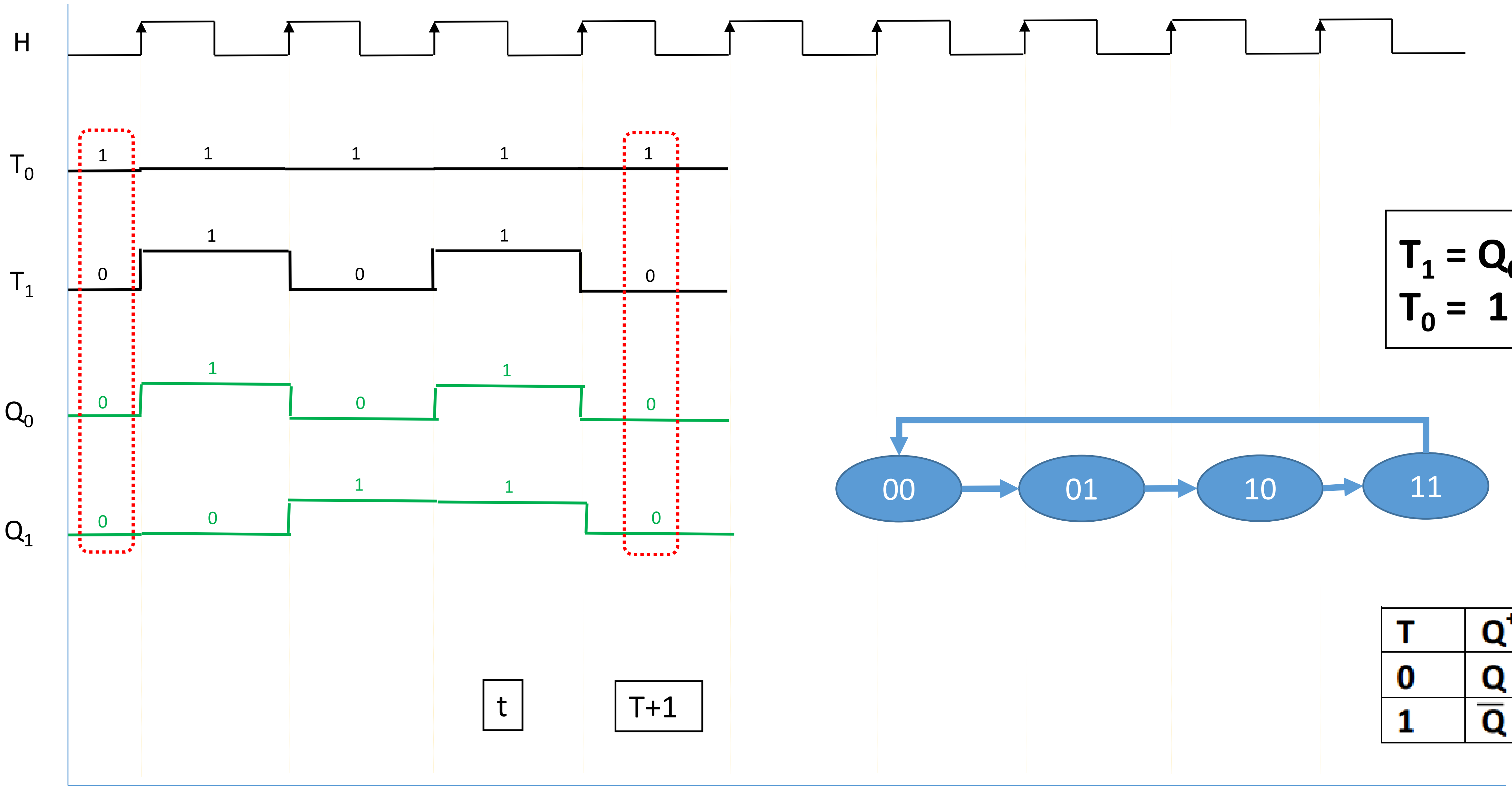



$T_1 = Q_0$

$T_0 = 1$

| T | $Q^+$ |
|---|---|
| 0 | $Q$ |
| 1 | $\overline{Q}$ |

# Exemple 1

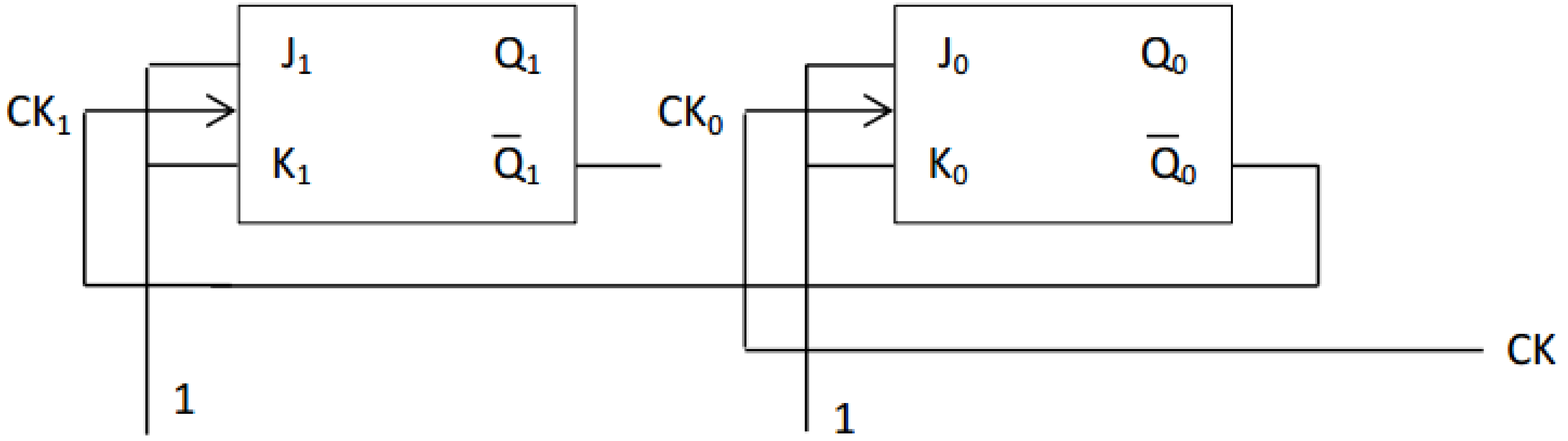



$$T_1 = Q_0$$
$$T_0 = 1$$

| T | $Q^+$ |
|---|---|
| 0 | $Q$ |
| 1 | $\overline{Q}$ |

# Chronogrammes

- **Exemple 2 :** **Circuit asynchrone**




J0 = 1

K0 = 1

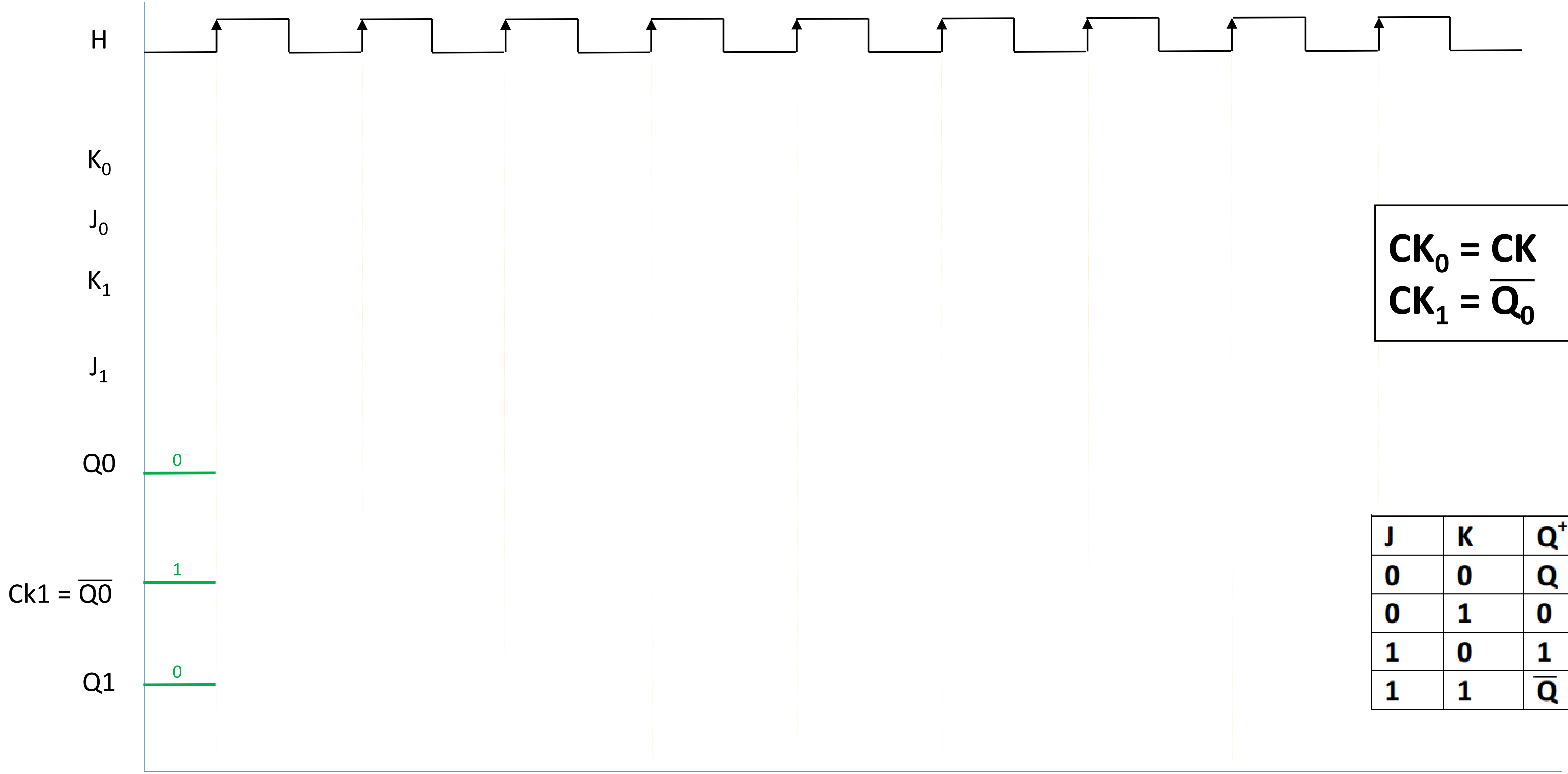
J1 = 1

J1 = 1

CK0 = CK

CK1 = $\overline{Q_0}$

# Exemple 2

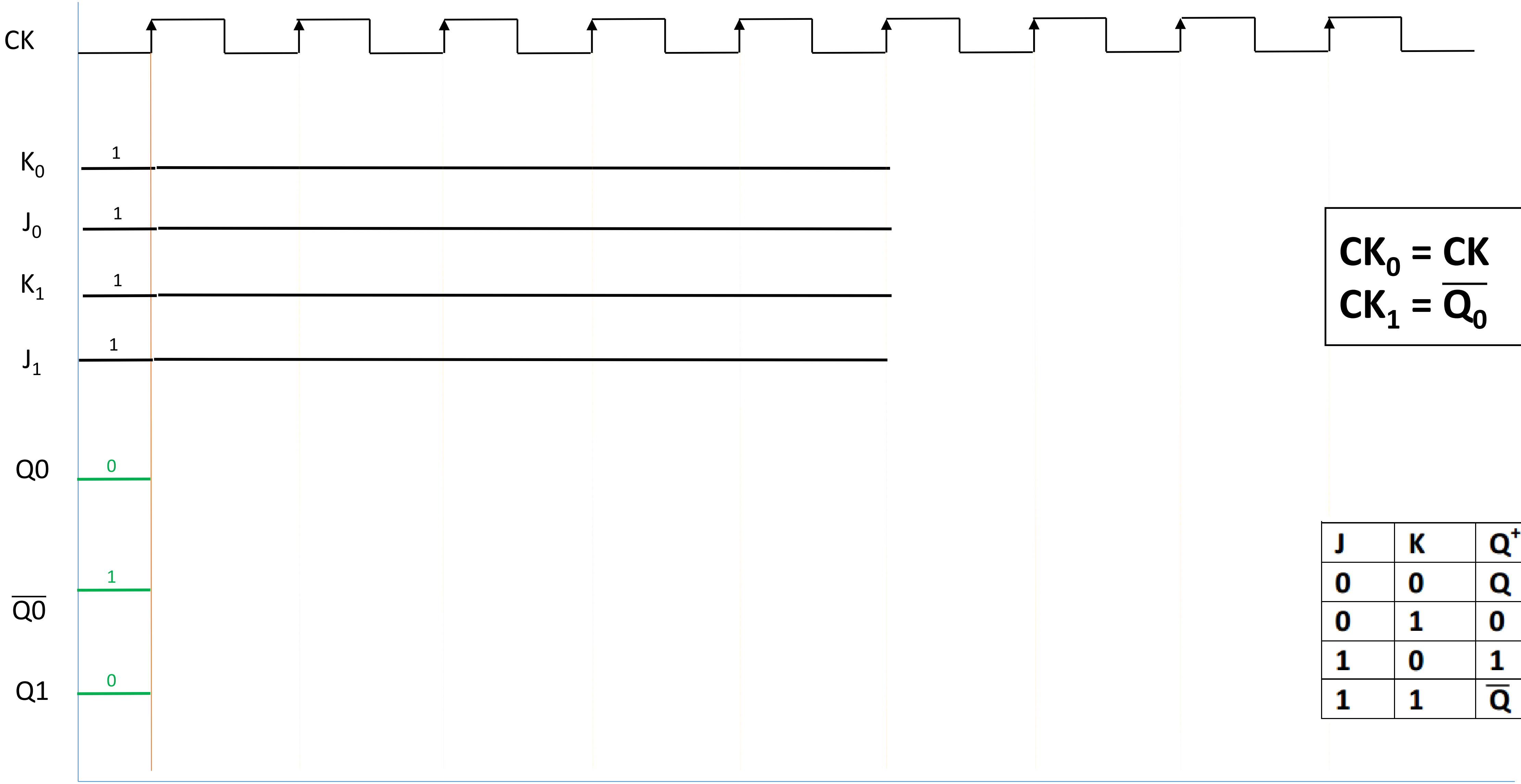



$CK_0 = CK$

$CK_1 = \overline{Q_0}$

| J | K | $Q^+$ |
|---|---|---|
| 0 | 0 | Q |
| 0 | 1 | 0 |
| 1 | 0 | 1 |
| 1 | 1 | $\overline{Q}$ |

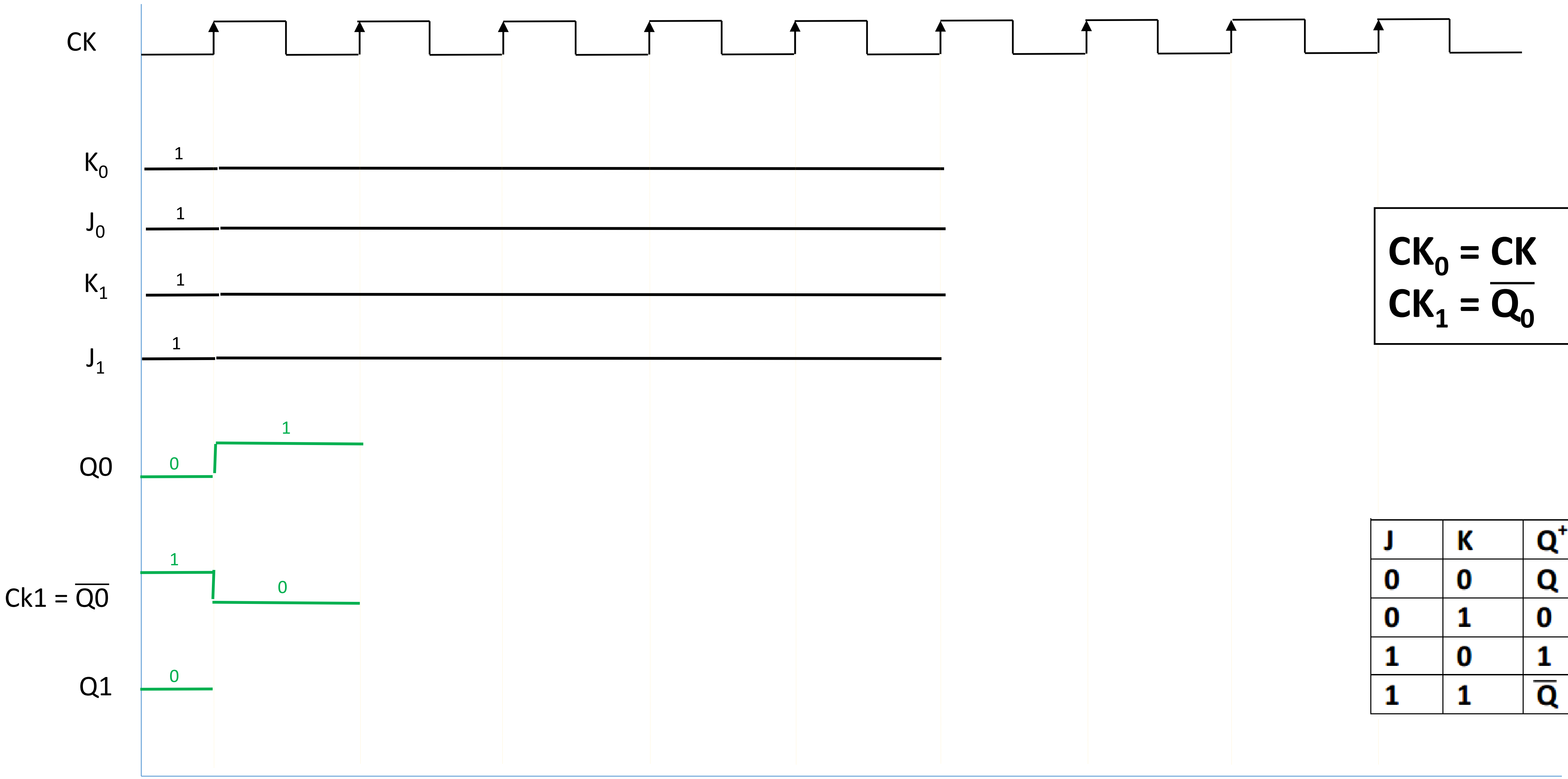
# Exemple 2




$CK_0 = CK$
$CK_1 = \overline{Q_0}$

| J | K | $Q^+$ |
|---|---|---|
| 0 | 0 | Q |
| 0 | 1 | 0 |
| 1 | 0 | 1 |
| 1 | 1 | $\overline{Q}$ |

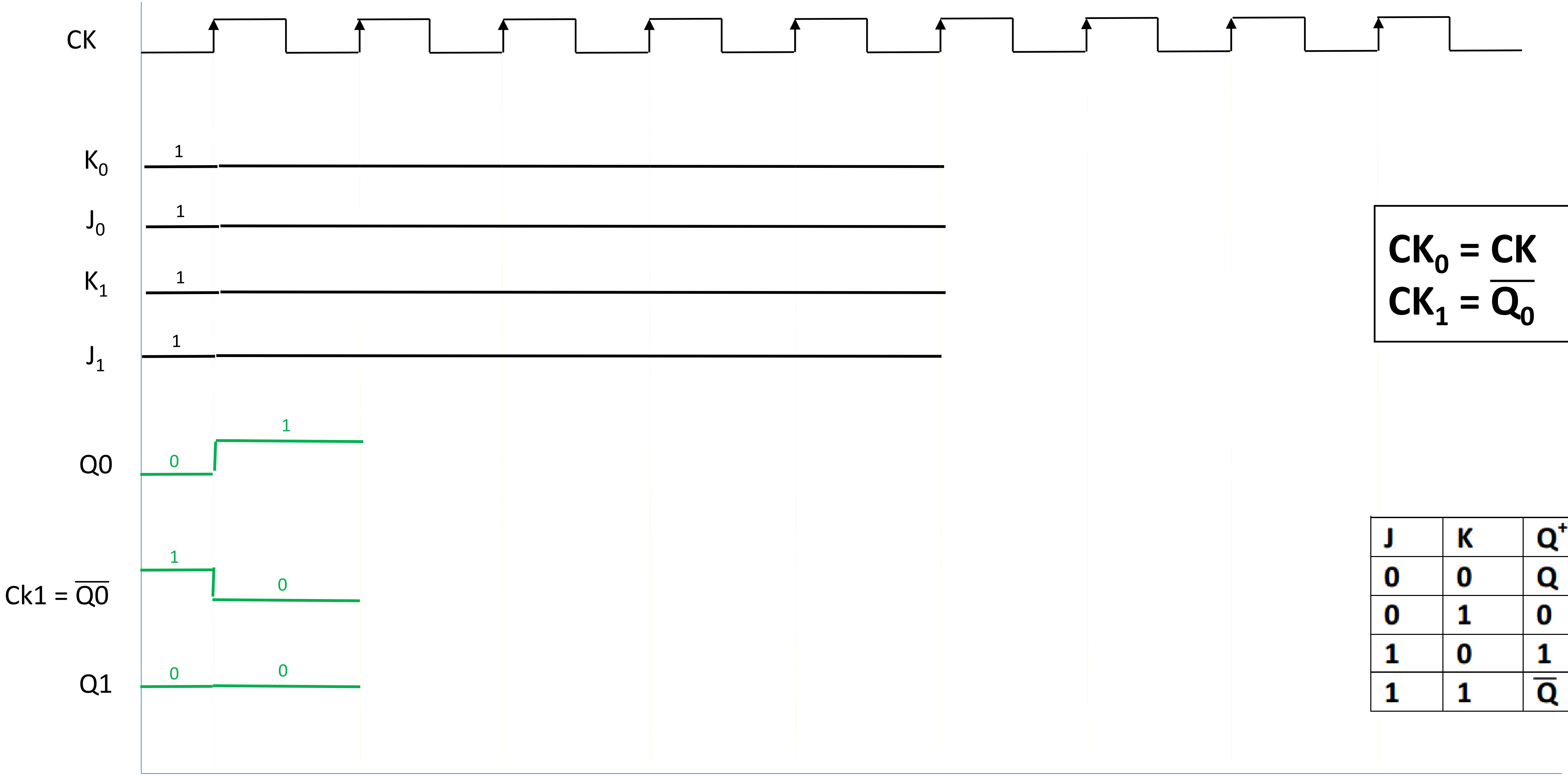
# Exemple 2




$$CK_0 = CK$$
$$CK_1 = \overline{Q_0}$$

| J | K | $Q^+$ |
|---|---|---|
| 0 | 0 | Q |
| 0 | 1 | 0 |
| 1 | 0 | 1 |
| 1 | 1 | $\overline{Q}$ |

# Exemple 2

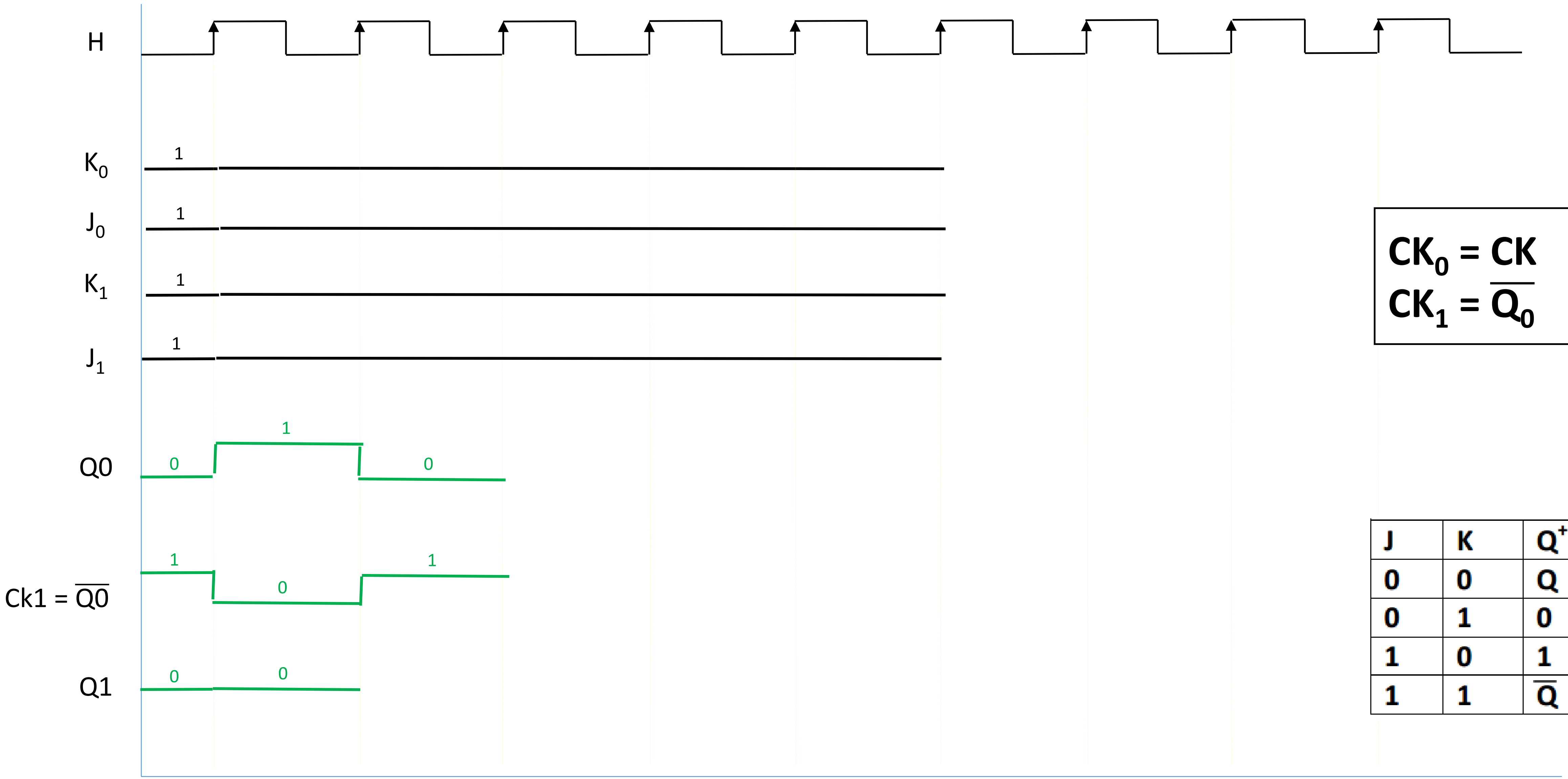



$CK_0 = CK$

$CK_1 = \overline{Q_0}$

| J | K | $Q^+$ |
|---|---|---|
| 0 | 0 | Q |
| 0 | 1 | 0 |
| 1 | 0 | 1 |
| 1 | 1 | $\overline{Q}$ |

# Exemple 2

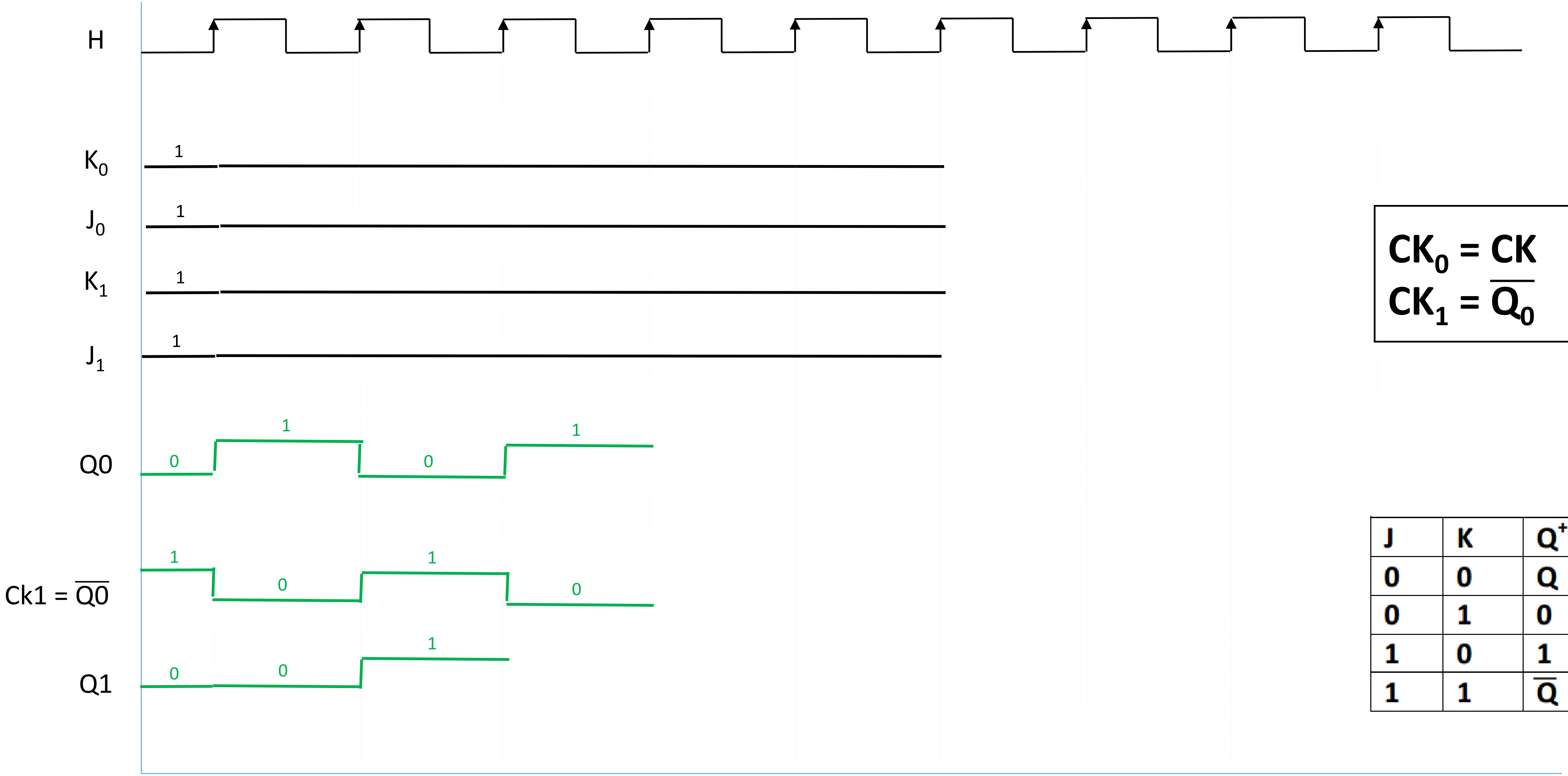



$CK_0 = CK$

$CK_1 = \overline{Q_0}$

| J | K | $Q^+$ |
|---|---|---|
| 0 | 0 | Q |
| 0 | 1 | 0 |
| 1 | 0 | 1 |
| 1 | 1 | $\overline{Q}$ |

# Exemple 2

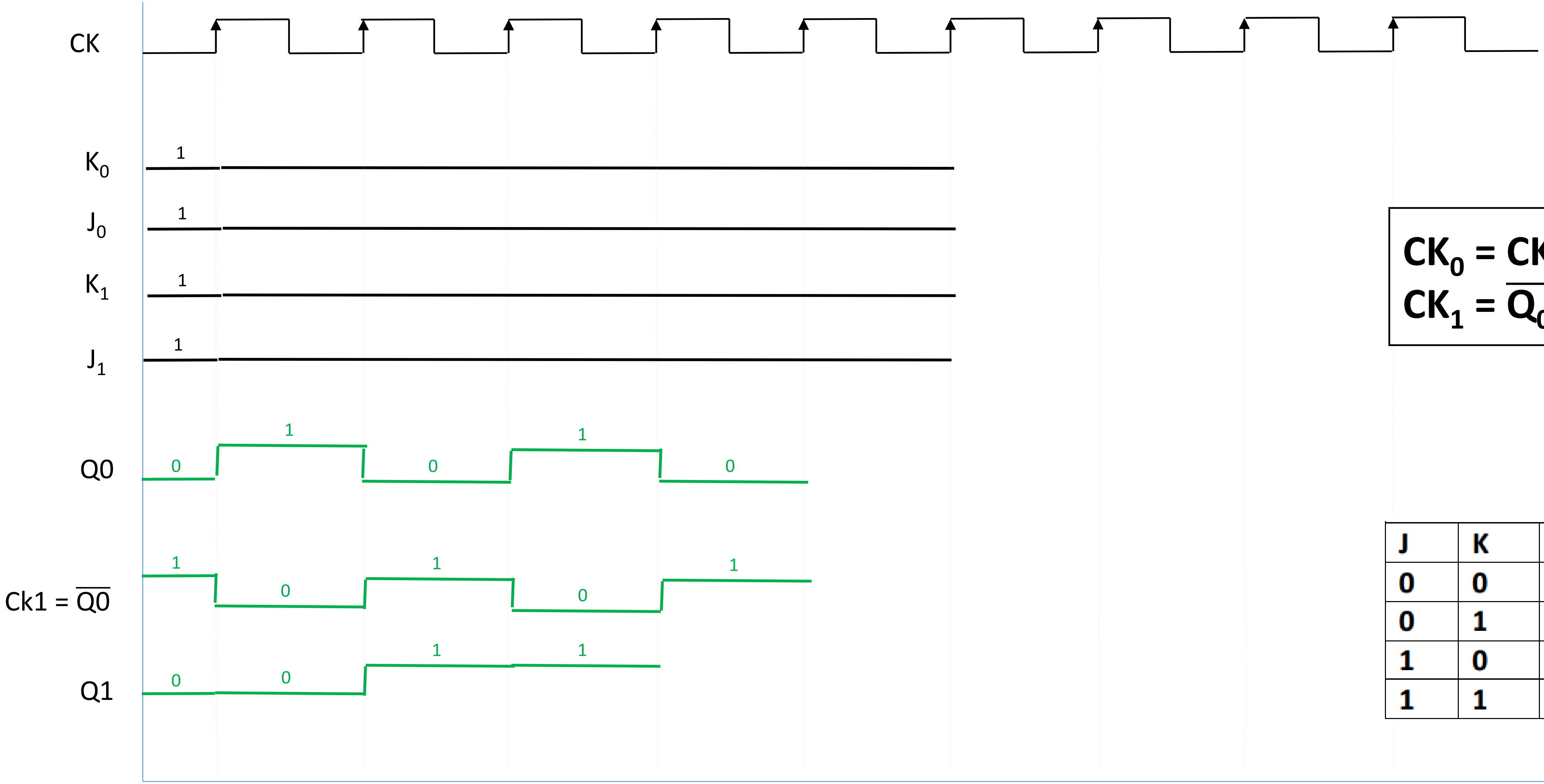



$CK_0 = CK$

$CK_1 = \overline{Q_0}$

| J | K | $Q^+$ |
|---|---|---|
| 0 | 0 | Q |
| 0 | 1 | 0 |
| 1 | 0 | 1 |
| 1 | 1 | $\overline{Q}$ |

# Exemple 2

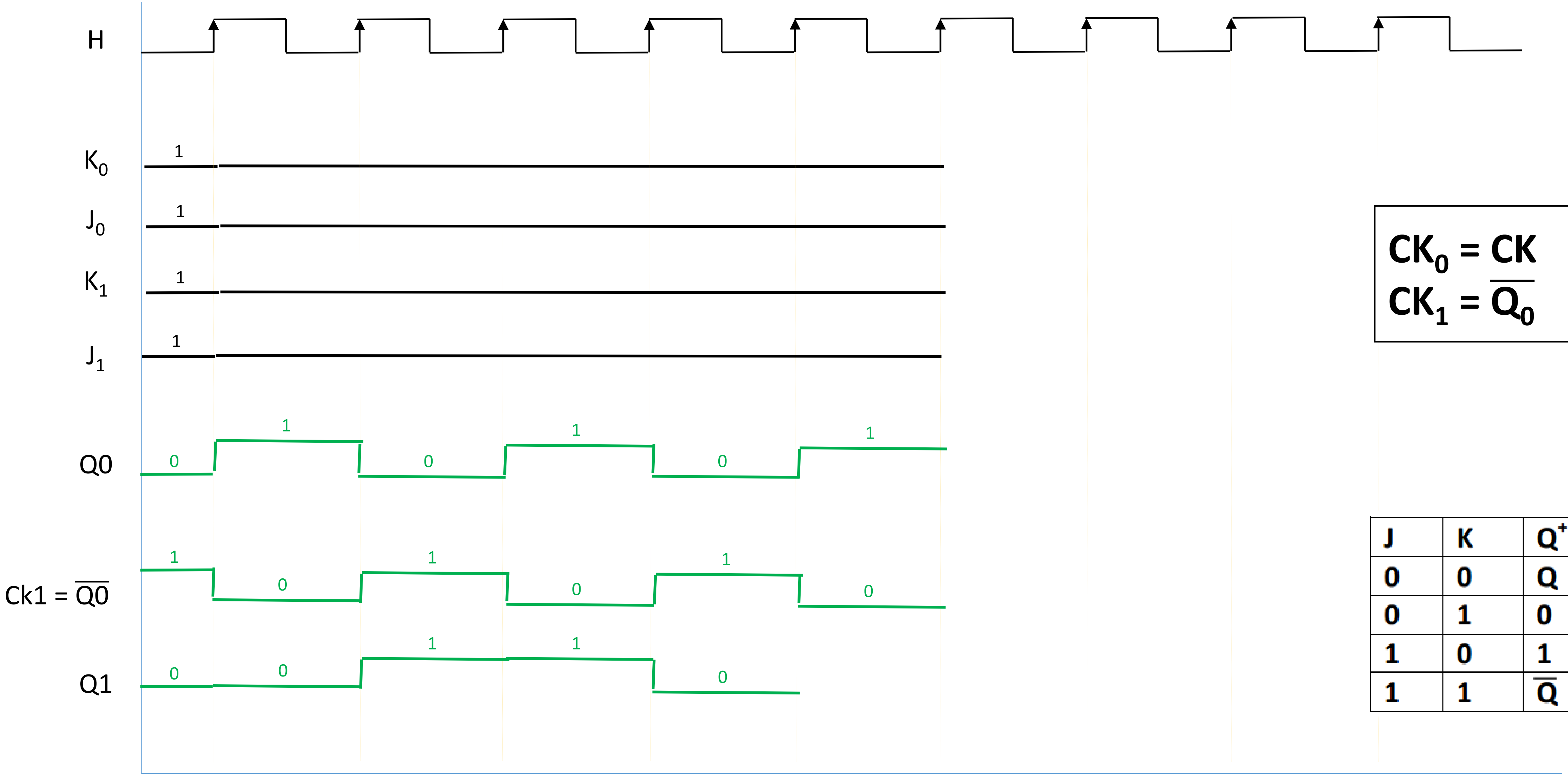



$CK_0 = CK$

$CK_1 = \overline{Q_0}$

| J | K | $Q^+$ |
|---|---|---|
| 0 | 0 | Q |
| 0 | 1 | 0 |
| 1 | 0 | 1 |
| 1 | 1 | $\overline{Q}$ |

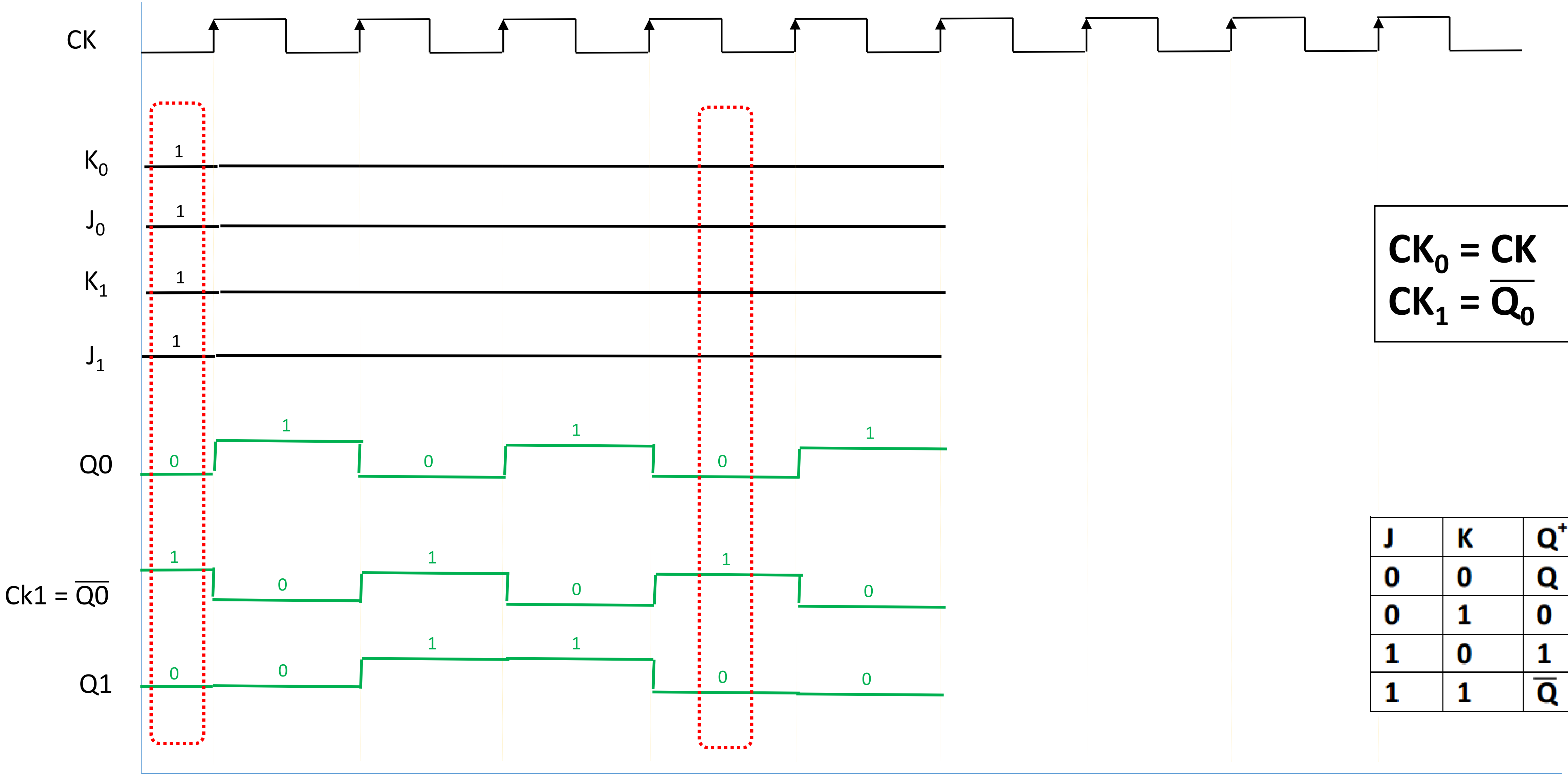
# Exemple 2




**$CK_0 = CK$**
**$CK_1 = \overline{Q_0}$**

| J | K | $Q^+$ |
|---|---|---|
| 0 | 0 | Q |
| 0 | 1 | 0 |
| 1 | 0 | 1 |
| 1 | 1 | $\overline{Q}$ |

# Exemple 2




$CK_0 = CK$

$CK_1 = \overline{Q_0}$

| J | K | $Q^+$ |
|---|---|---|
| 0 | 0 | Q |
| 0 | 1 | 0 |
| 1 | 0 | 1 |
| 1 | 1 | $\overline{Q}$ |

# Fonctions de forçage Clear et Preset

- **<u>Fonction Clear :</u>**

- Clear est une fonction logique qui permet de mettre à zéro une bascule à n'importe quel moment quelque soient son état, ses entrées ou son horloge.

- **Dans un circuit séquentiel** lorsque la fonction Clear est activée, toutes les bascules affichent instantanément zéro sans tenir compte des autres entrées.

- C'est la remise à zéro du circuit

# Fonctions de forçage Clear et Preset

- <u>**Fonction Preset**</u>

- Preset est une fonction logique qui permet de mettre à 1 une bascule à n'importe quel moment quelque soient son état, ses entrées ou son horloge.

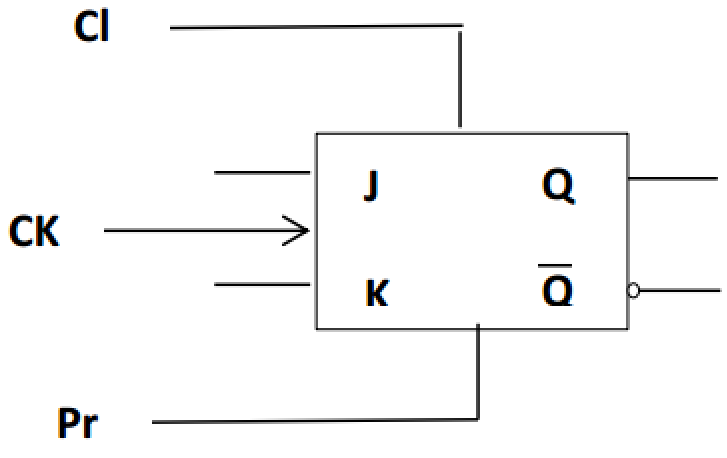
- **Dans un circuit séquentiel** lorsque la fonction Preset est activée, toutes les bascules affichent instantanément 1 sans tenir compte des autres entrées.

C'est la mise à 1 du circuit

# Fonctions de forçage Clear et Preset

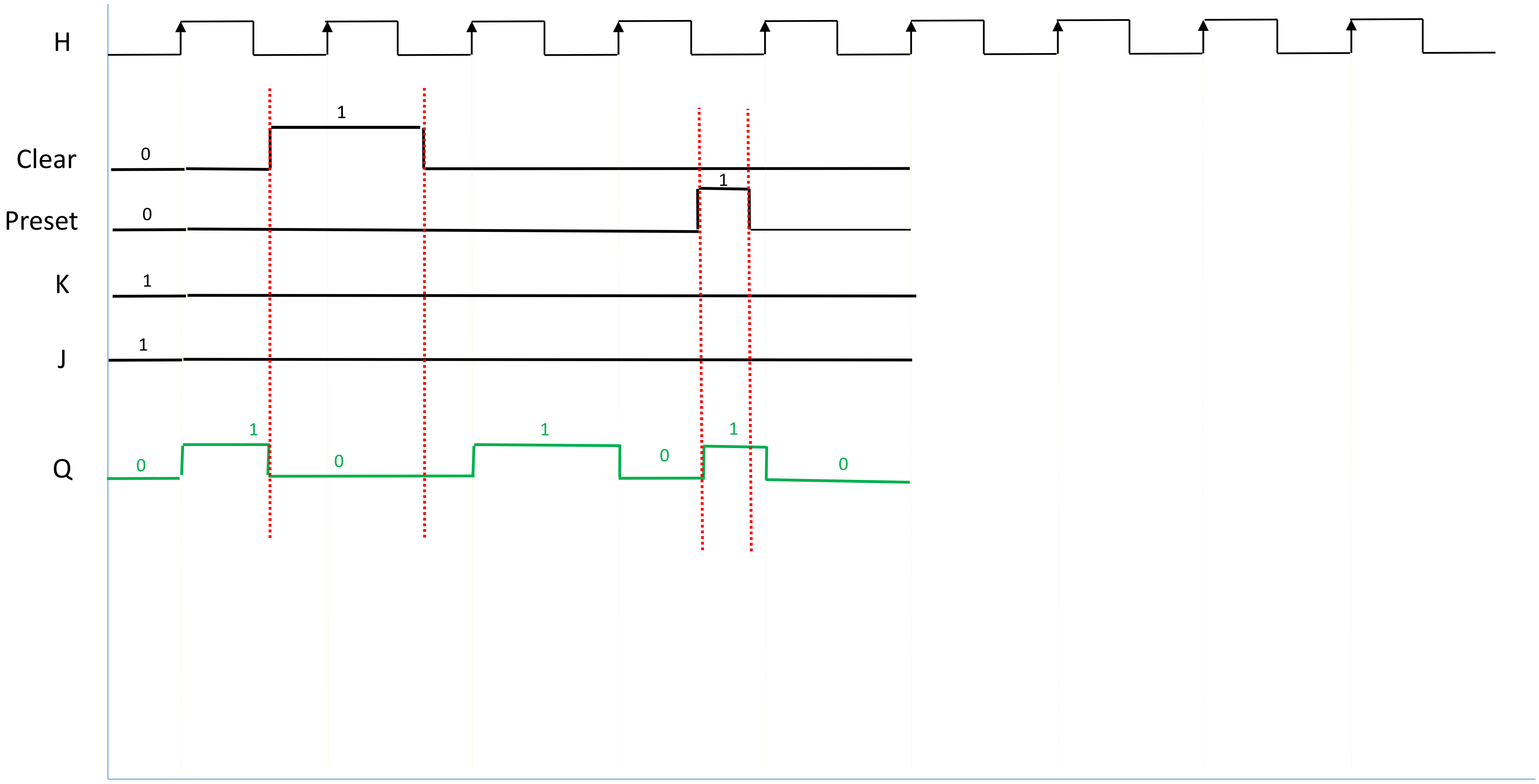
- **Exemple pour une bascule JK**

- Les entrées Clear et Preset forcent la bascule à se mettre à 0 ou à 1 sans tenir compte des valeurs des entrées J et K ni de l'horloge CK .




| Clear | Preset | $Q_{t+1}$ |
|---|---|---|
| 0 | 0 | Dépend de (J K) et CK |
| 0 | 1 | Mise à 1 ∀ (J K) et CK |
| 1 | 0 | Mise à 0 ∀ (J K) et CK |
| 1 | 1 | Interdit |

# Exemple 2



# Ch1. La première partie

## Circuits séquentiels

### Analyse

**Table de vérité :**

a) Etablir les équations d'entrée de chaque bascule

b) Réaliser la Table de Vérité du circuit (le principe est de retrouver les Qi+ à partir des valeurs des équations d'entrée).

c) En déduire le diagramme des états d'où le rôle du circuit.

Circuits synchrones

**Chronogramme :**

a) Donner l'état initial de chaque bascule à l'instant (t) b) En déduire les valeurs des entrées pour chaque bascule à l'instant (t) c) A partir de ces entrées, retrouver l'état de chaque bascule à l'instant (t+1) d) (t+1) devient (t) et on recommence jusqu'à ce qu'on retrouve l'état initial

Circuits synchrones et asynchrones

### Synthèse

**Table de transition :**

a) Etablir le diagramme des états (ou séquence) et donner le nombre de bascules nécessaires.

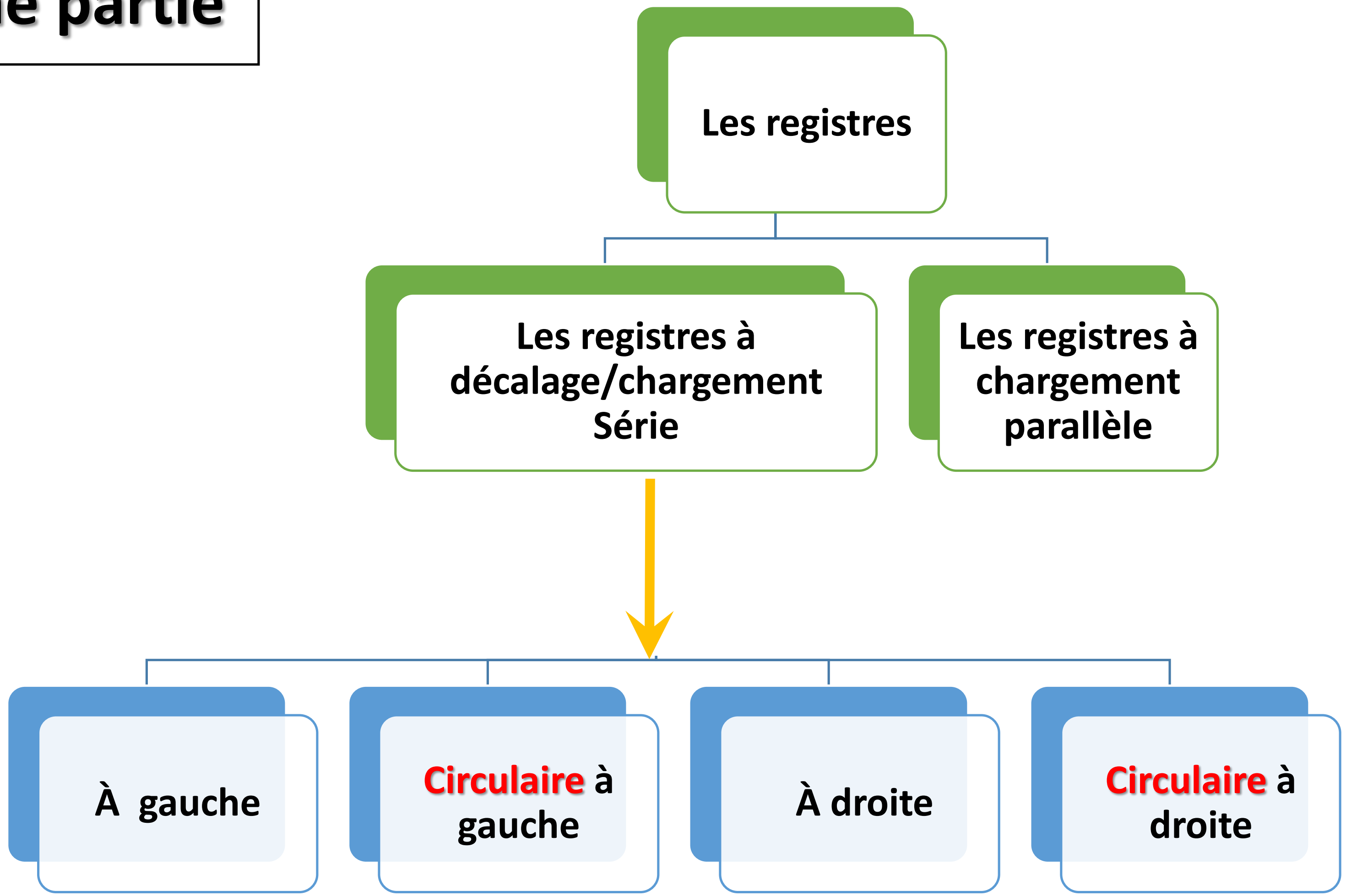
b) Réaliser la table de transition

c) En déduire les équations d'entrée aux bascules

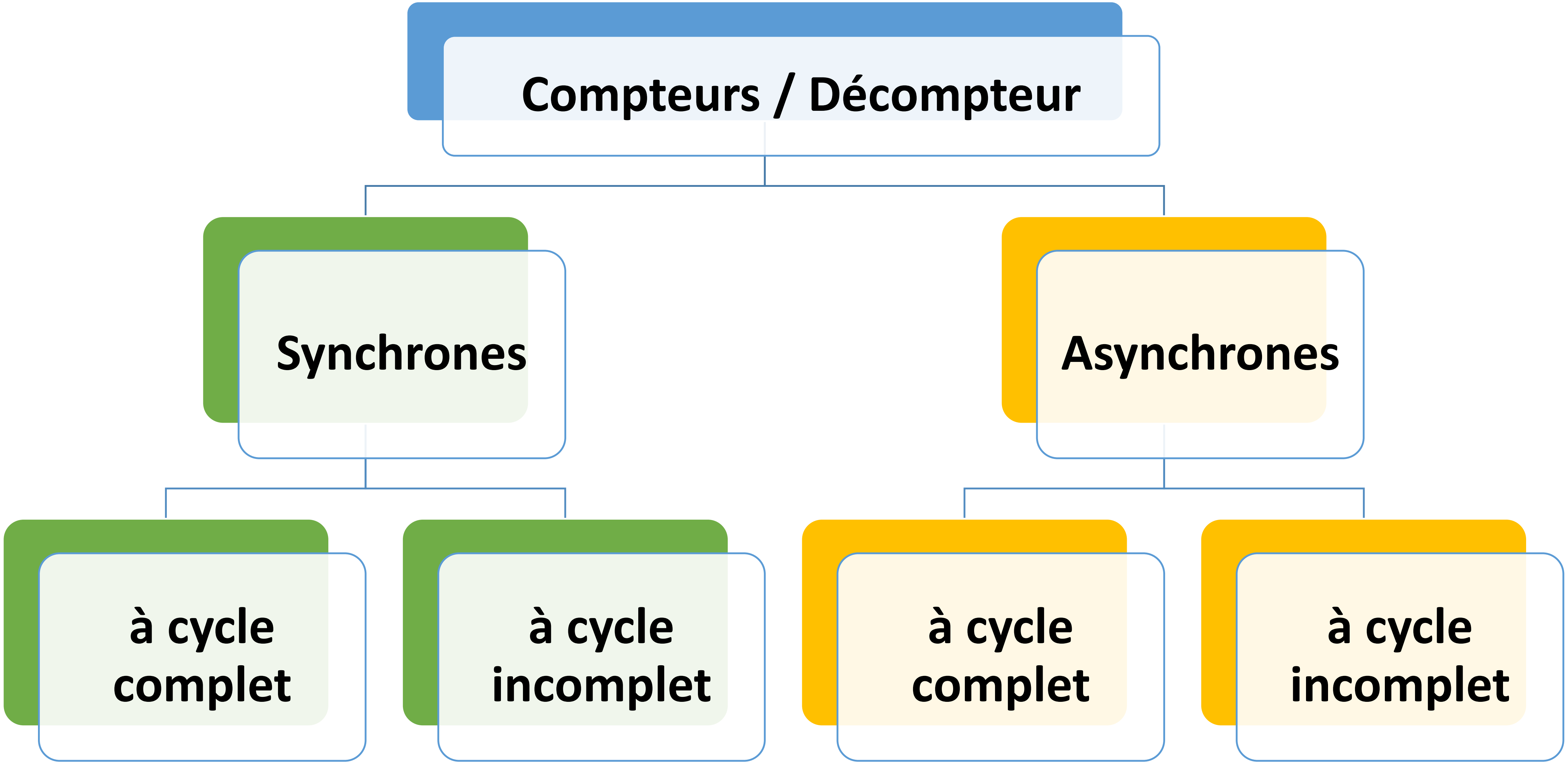
d) Réaliser le circuit correspondant

Circuits synchrones

# Ch1. La deuxième partie



# Ch1. La troisième partie



# Les registres

**Registres :** Un registre est un ensemble ordonné de **n bascules** qui permet de mémoriser une information sur **n** bits.

**Registres à chargement série (décalage) :** Les registres à chargement série ou décalage sont des registres à entrées série et sortie série

A gauche | gauche | A droite | droite | décalage/chargement Série | chargement parallèle

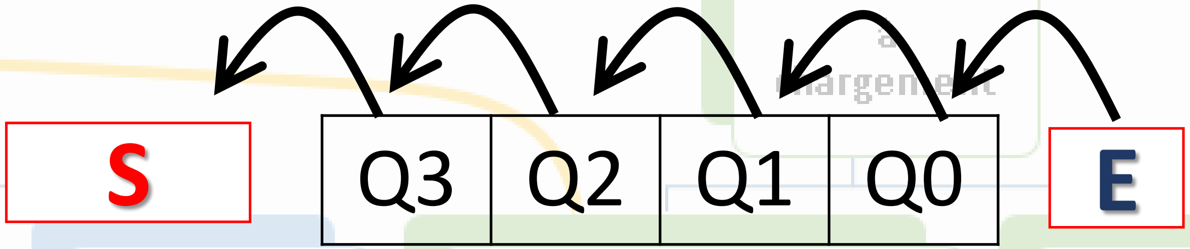
**bascules D (Décalage)**

| D | $Q^+$ |
|---|---|
| 0 | 0 |
| 1 | 1 |

# Les registres

## a) Registre à décalage à gauche (Registres à chargement série à gauche) :

| | | | |
|---|---|---|---|
| Q2 | Q1 | Q0 | E |




**L'ensemble du registre est décalé d'une position vers la gauche :**

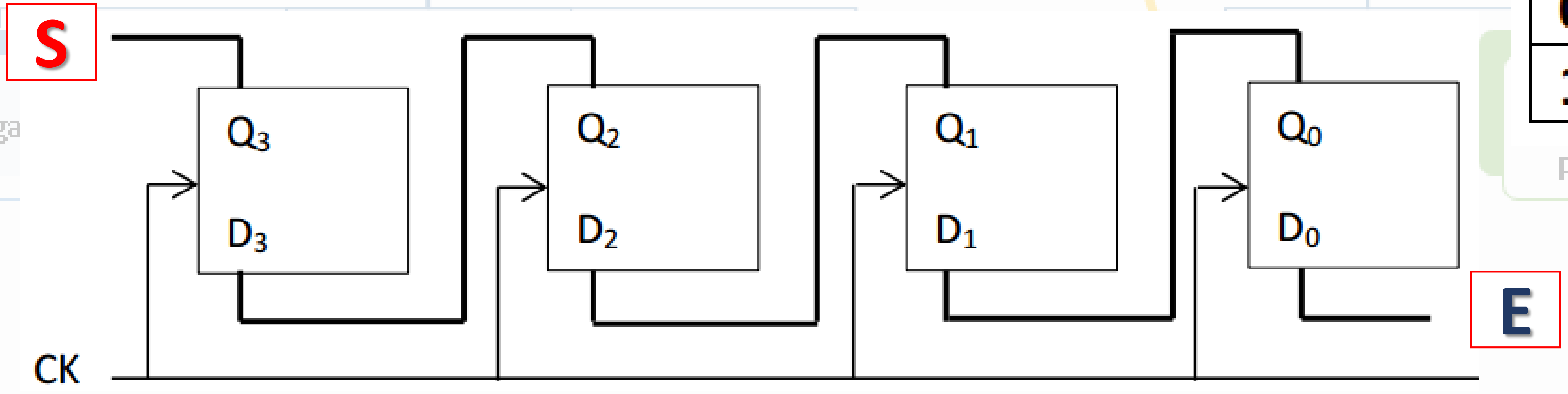
| | | | | |
|---|---|---|---|---|
| $S = Q3$ | $Q3^{+} = Q2$ | $Q2^{+} = Q1$ | $Q1^{+} = Q0$ | $Q0^{+} = E$ |

# Les registres

## a) Registre à décalage à gauche (Registres à chargement série à gauche) :

| $S = Q3$ | $D3=Q3^{+} = Q2$ | $D2=Q2^{+} = Q1$ | $D1=Q1^{+} = Q0$ | $D0=Q0^{+}= E=0$ |
|---|---|---|---|---|

| D | $Q^{+}$ |
|---|---|
| 0 | 0 |
| 1 | 1 |



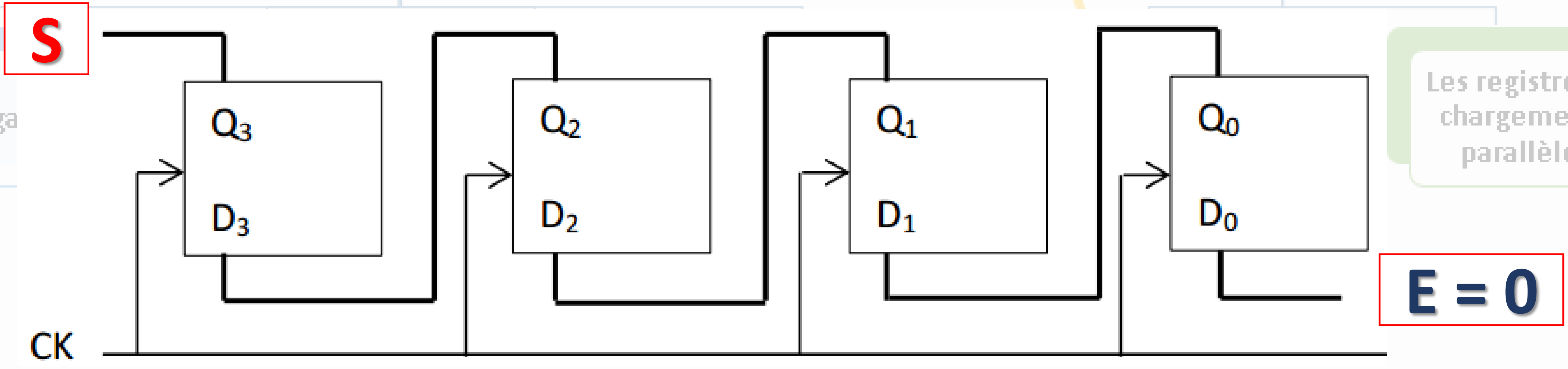

Plus généralement pour un registre de **n** bascules on a :

$D_0 = E$ , $D_i = Q_{i-1}$ et $S = Q_{n-1}$

# Les registres

## a) Registre à décalage à gauche (Registres à chargement série à gauche) :

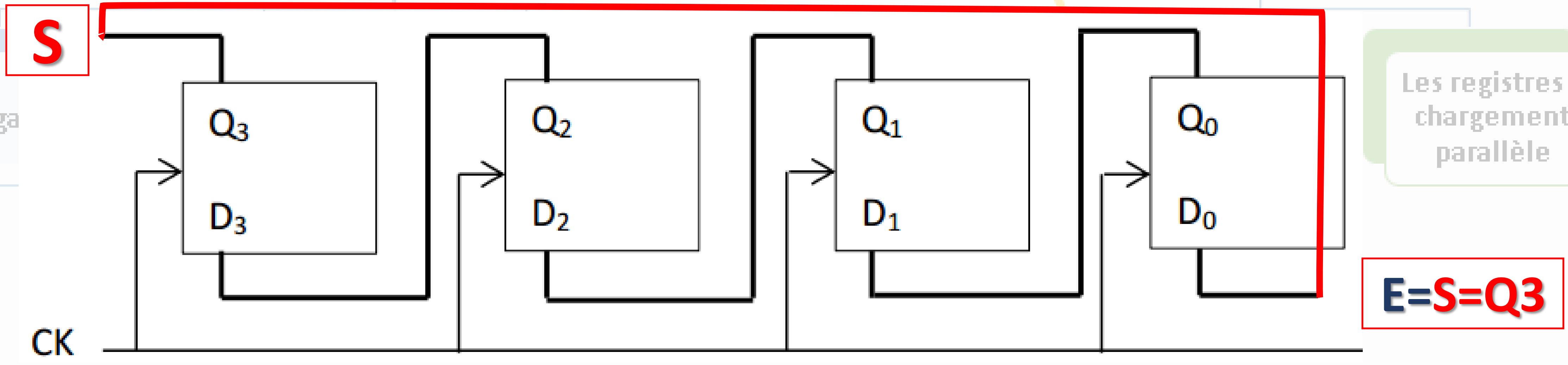
| | | | | |
|---|---|---|---|---|
| **S** = Q3 | $Q3^+ = Q2$ | $Q2^+ = Q1$ | $Q1^+ = Q0$ | $Q0^+ =$ **E=0** |



# Les registres

## a) Registre à décalage circulaire à gauche :

| | | | |
|---|---|---|---|
| $D3=Q3^{+}=Q2$ | $D2=Q2^{+}=Q1$ | $D1=Q1^{+}=Q0$ | $D0=Q0^{+}=Q3=S$ |




| Q3 | Q2 | Q1 | Q0 |
|---|---|---|---|
| 0 | 1 | 1 | 0 |

| Q3+ | Q2+ | Q1+ | Q0+ |
|---|---|---|---|
| 1 | 1 | 0 | 0 |

# Les registres

## a) Registre à décalage à droite (Registres à chargement série à droite) :

| | | | |
|---|---|---|---|
| Q3 | Q2 | Q1 | Q0 |

| | | | | | |
|---|---|---|---|---|---|
| **E** | Q3 | Q2 | Q1 | Q0 | **S** |

**L'ensemble du registre est décalé d'une position vers la gauche :**

| | | | | |
|---|---|---|---|---|
| $D3=Q3^{+}=E$ | $D2=Q2^{+} = Q3$ | $D1=Q1^{+} = Q2$ | $D0=Q0^{+} = Q1$ | $S = Q0$ |

| D | $Q^{+}$ |
|---|---|
| 0 | 0 |
| 1 | 1 |

Plus généralement pour un registre de **n** bascules on a :

$D_{n-1} = E$ ,     $D_{i-1} = Q_i$     et     $S = D_0$

# Les registres

**Les registres à chargement parallèle**

Le chargement d'une information donnée sur un registre consiste à donner à ce registre la valeur de cette information quelque soit son état précédent.

Le registre à chargement parallèle peut charger une information sur **n** bits en même temps.

Chaque bascule prend la valeur de l'information contenue dans le bit correspondant

# Les registres

**Les registres à chargement parallèle**

Si on veut charger par exemple l'information A B C D sur un registre de 4 bits, quelque soit l'état de ce registre à l'instant t, il affichera A B C D à l'instant t+1

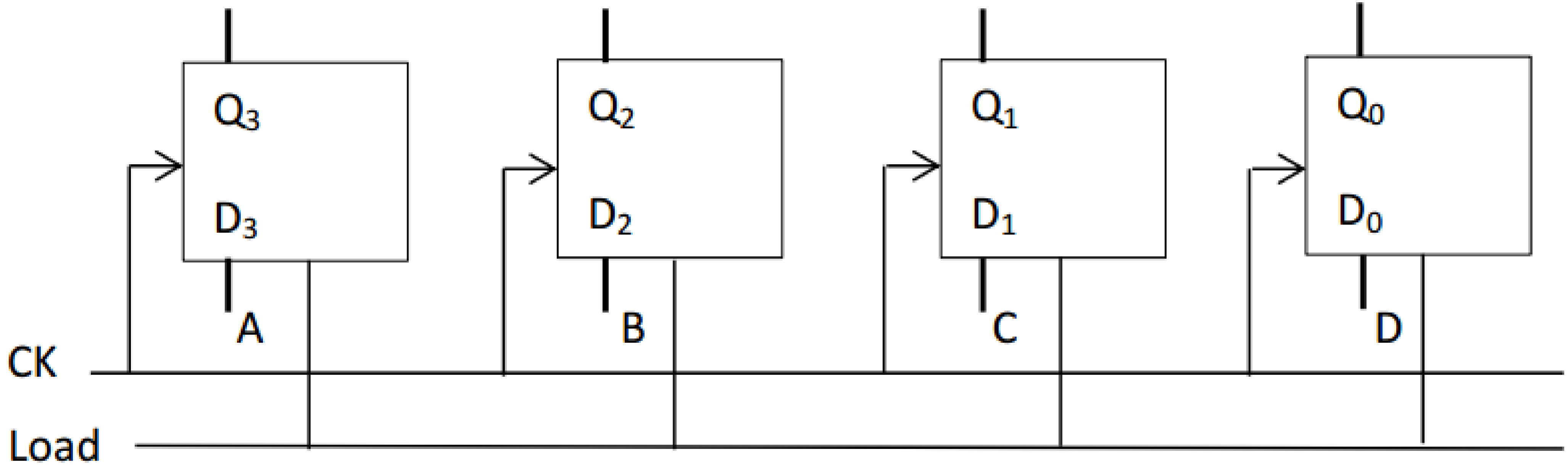
| A | B | C | D |
|---|---|---|---|
| ↓ | ↓ | ↓ | ↓ |
| Q3 | Q2 | Q1 | Q0 |

| | | | |
|---|---|---|---|
| A | B | C | D |

| | | | |
|---|---|---|---|
| $Q3^{+}=A$ | $Q2^{+} = B$ | $Q1^{+} = C$ | $Q0^{+} = D$ |

# Les registres

## Les registres à chargement parallèle




Si Load = 0  pas de changement

Si Load = 1 chargement de l'l'information

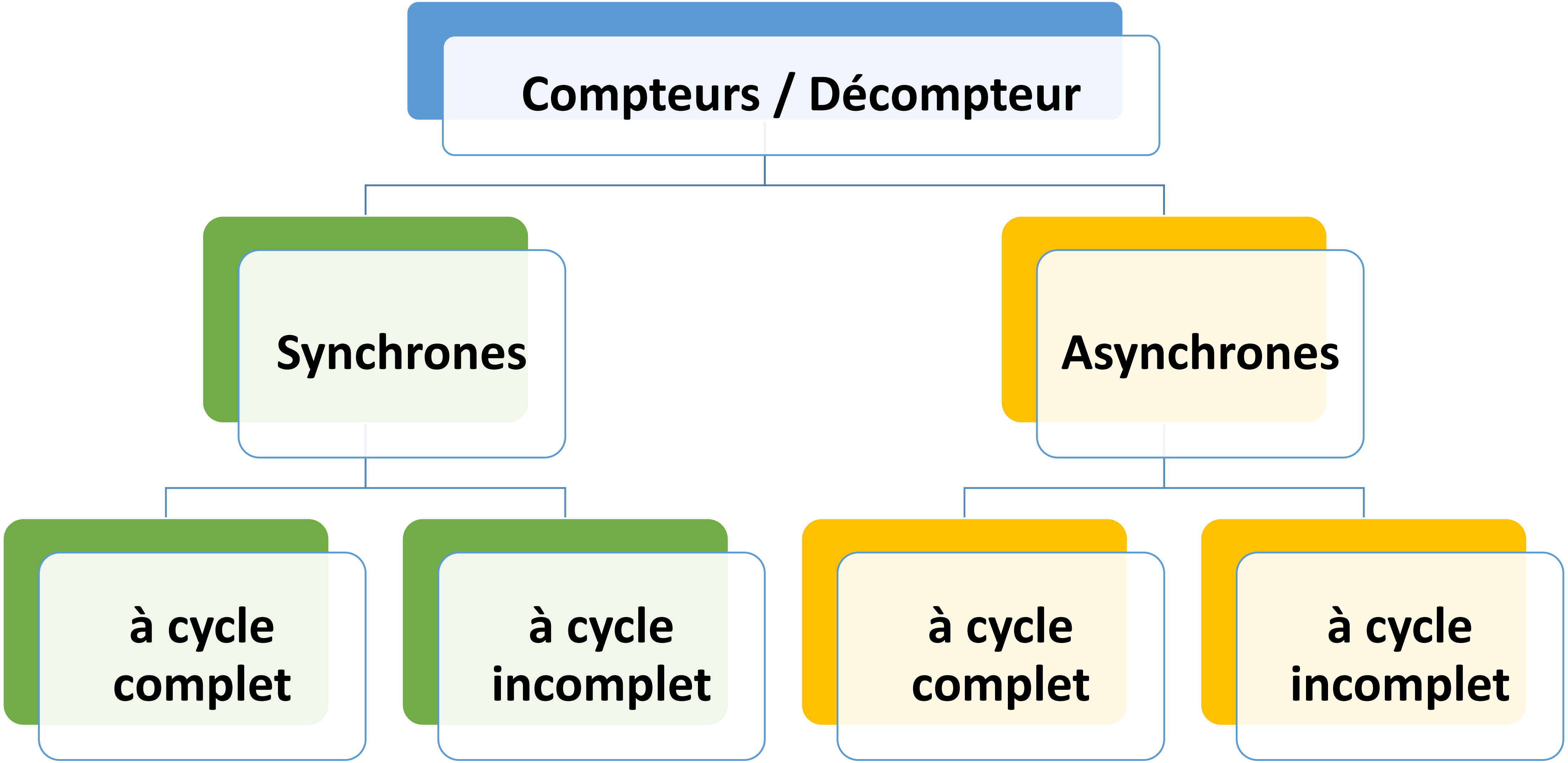
# Les registres

## Les registres à chargement parallèle

| 0 | 1 | 0 | 1 |
|---|---|---|---|
| Q3 | Q2 | Q1 | Q0 |

| | | | | |
|---|---|---|---|---|
| **T+1** | 0 | 1 | 0 | 1 |

| | | | |
|---|---|---|---|
| $Q3^{+}=0$ | $Q2^{+} = 1$ | $Q1^{+} = 0$ | $Q0^{+} = 1$ |


Compteurs / Décompteur
Synchrones
Asynchrones
à cycle complet
à cycle incomplet
à cycle complet
à cycle incomplet

# Les compteurs

Un compteur est un circuit séquentiel qui possède n états (0, 1, 2, ……n-1)
A chaque impulsion d'horloge il passe de l'état i à l'état i+1 et revient toujours à l'état initial.

Le modulo d'un compteur est le nombre d'états de celui-ci.
Un compteur modulo 8 par exemple est un compteur à 8 états (0, 1, 2, …7)

Il existe deux types de compteurs, les compteurs synchrones et les compteurs asynchrones.

# Les compteurs

**Les compteurs synchrones**

Un compteur synchrone est un compteur ou toutes les bascules possèdent la même horloge.

Le nombre de bascules d'un compteur modulo n est égal à **p** si $\mathbf{2^{p-1} < n \leq 2^p}$

P est la puissance de 2 qui est immédiatement supérieure à n

Par exemple le nombre de bascules d'un compteur modulo 6 est **3** car $\mathbf{2^2 < 6 <= 2^3}$

# Les compteurs

**Compteur synchrone à cycle complet**

Un compteur à cycle complet est un compteur modulo n tel que $\mathbf{n = 2^p}$

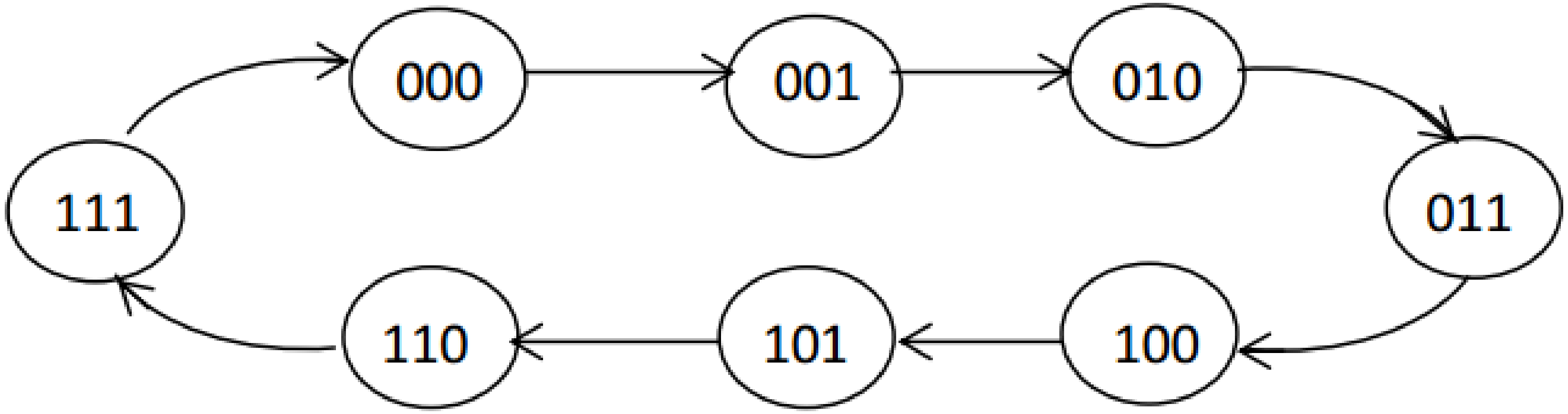
Par exemple le compteur modulo 8 et le compteur modulo 16 sont des compteurs à cycle complet. $8 = 2^3$

Exemple : Réalisation d'un compteur modulo 8 à l'aide de bascules JK

# Les compteurs

Exemple : Réalisation d’un compteur modulo 8 à l’aide de bascules JK

N=8 ➔ $8 = 2^3$



# Les compteurs

Table de transition

| $Q_2$ $Q_1$ $Q_0$ | $Q_2^+$ $Q_1^+$ $Q_0^+$ | $J_2$ $K_2$ | $J_1$ $K_1$ | $J_0$ $K_0$ |
|---|---|---|---|---|
| 0 0 0 | 0 0 1 | 0 X | 0 X | 1 X |
| 0 0 1 | 0 1 0 | 0 X | 1 X | X 1 |
| 0 1 0 | 0 1 1 | 0 X | X 0 | 1 X |
| 0 1 1 | 1 0 0 | 1 X | X 1 | X 1 |
| 1 0 0 | 1 0 1 | X 0 | 0 X | 1 X |
| 1 0 1 | 1 1 0 | X 0 | 1 X | X 1 |
| 1 1 0 | 1 1 1 | X 0 | X 0 | 1 X |
| 1 1 1 | 0 0 0 | X 1 | X 1 | X 1 |

Equations d'entrée :

$J_0 = 1$

$K_0 = 1$

$J_1 = Q_0$

$K_1 = Q_0$

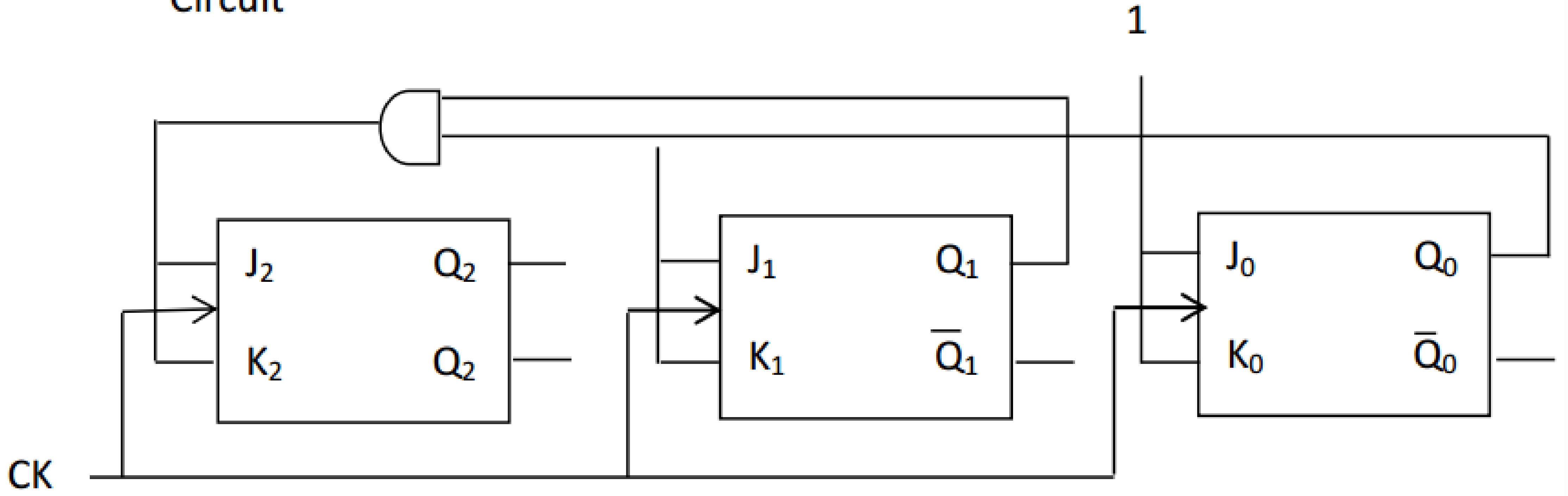
$J_2 = Q_0Q_1$

$K_2 = Q_0Q_1$

# Les compteurs

Compteurs / Décompteur

Circuit




complet

incomplet

complet

incomplet

# Les compteurs

Dans un cas plus général on aura :

$$J_0 = 1$$
$$K_0 = 1$$

$$J_j = Q_0Q_1.....Q_{j-1}$$
$$K_j = Q_0Q_1.....Q_{j-1}$$

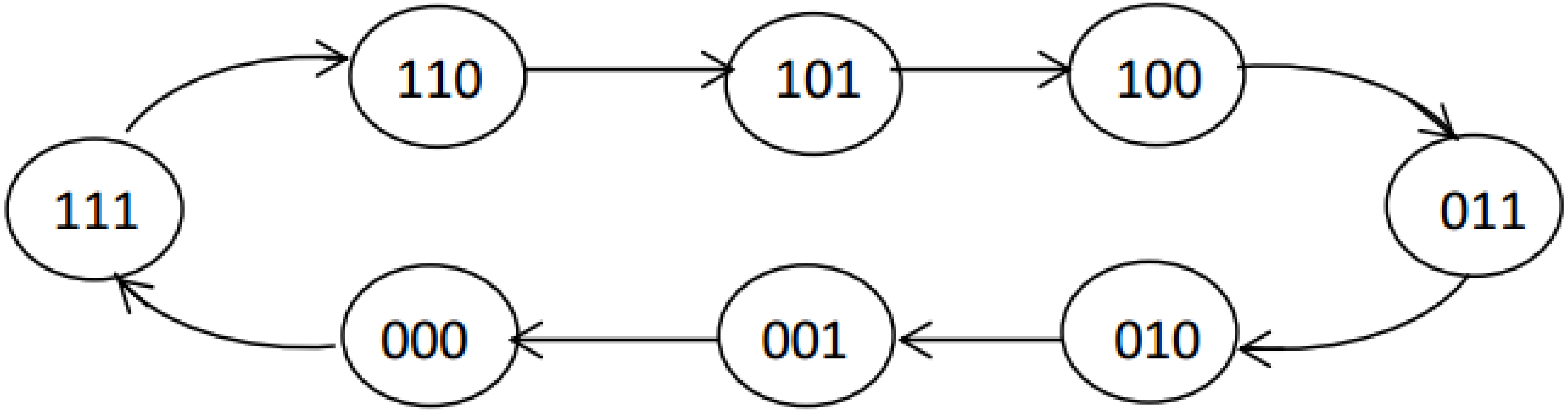
J0 =1 k0 =1
J1 =Q0 K1=Q0
J2 = K2 = Q0Q1
J3 = K3 = Q0Q1Q2
J4=K4 = Q0Q1Q2Q3

à cycle incomplet

**Décompteur synchrone à cycle complet**

Un décompteur à cycle complet est un décompteur modulo n tel que **n = $2^p$** ,
il passe de l'état **i** à l'état **i-1** et revient toujours à l'état **n-1**.

Exemple : Réalisation d'un décompteur modulo 8 à l'aide de bascules JK



# Les compteurs

Table de transition

| $Q_2$ $Q_1$ $Q_0$ | $Q_2^+$ $Q_1^+$ $Q_0^+$ | $J_2$ $K_2$ | $J_1$ $K_1$ | $J_0$ $K_0$ |
|---|---|---|---|---|
| 0 0 0 | 1 1 1 | 1 X | 1 X | 1 X |
| 0 0 1 | 0 0 0 | 0 X | 0 X | X 1 |
| 0 1 0 | 0 0 1 | 0 X | X 1 | 1 X |
| 0 1 1 | 0 1 0 | 0 X | X 0 | X 1 |
| 1 0 0 | 0 1 1 | X 1 | 1 X | 1 X |
| 1 0 1 | 1 0 0 | X 0 | 0 X | X 1 |
| 1 1 0 | 1 0 1 | X 0 | X 1 | 1 X |
| 1 1 1 | 1 1 0 | X 0 | X 0 | X 1 |

Equations d'entrée

$J_0 = 1$

$K_0 = 1$

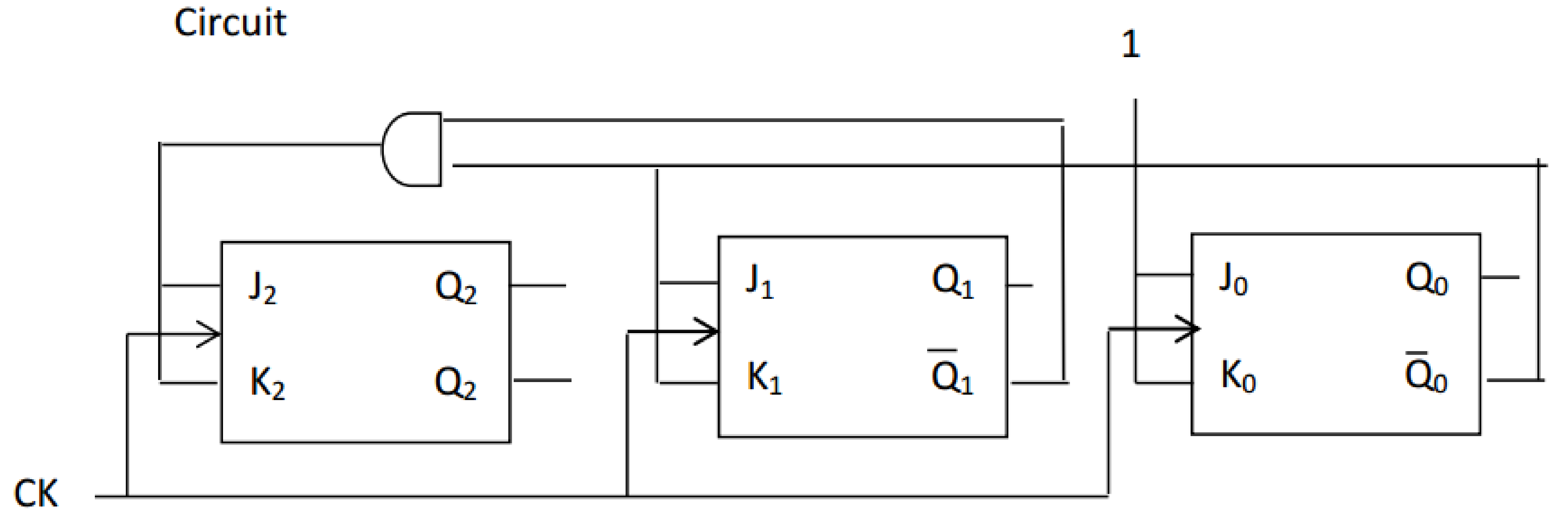
$J_1 = \overline{Q_0}$

$K_1 = \overline{Q_0}$

$J_2 = \overline{Q_0}\,\overline{Q_1}$

$K_2 = \overline{Q_0}\,\overline{Q_1}$

# Les compteurs




a cycle complet

a cycle incomplet

a cycle complet

a cycle incomplet

# Les compteurs

Dans un cas plus général on aura :

$$J_0 = 1$$
$$K_0 = 1$$

$$J_i = \overline{Q_0}\,\overline{Q_1}\ldots\ldots\overline{Q_{j-1}}$$
$$K_i = \overline{Q_0}\,\overline{Q_1}\ldots\ldots\overline{Q_{j-1}}$$

J0 =1 k0 =1
J1 =/Q0 K1=/Q0
J2K2 = /Q0/Q1
J3=K3 = /Q0/Q1/Q2

J5 = K5 = /Q0/Q1/Q2/Q3/Q4

à cycle incomplet

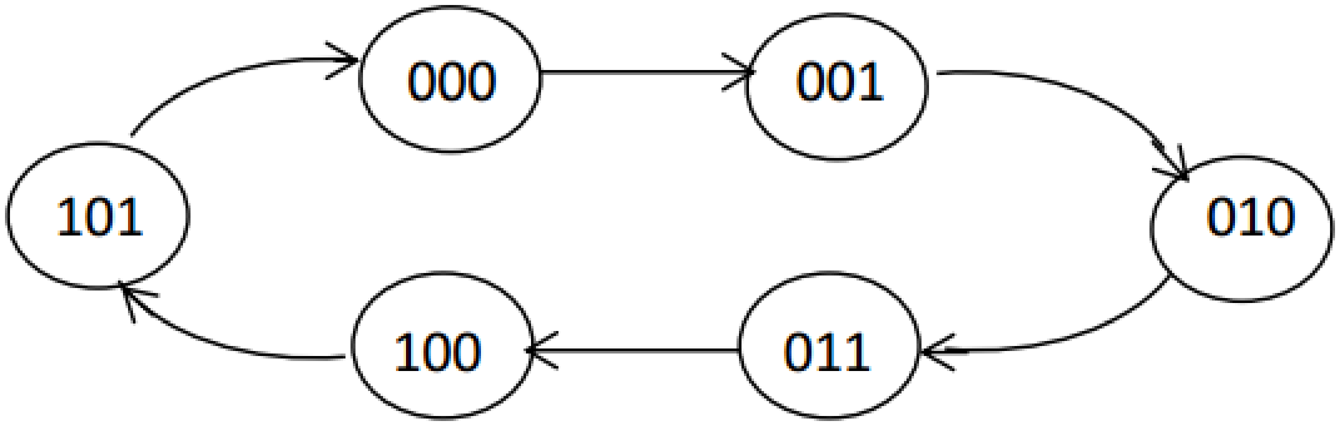
# Les compteurs

**Compteur synchrone à cycle incomplet**

Un compteur à cycle incomplet est un compteur à n états tel que $2^{p-1} < n < 2^p$ (n n'est pas une puissance de 2).

Le nombre de bascules de ce compteur est égal à **p**

Exemple : Le compteur modulo 6 : $4=2^2 < \mathbf{6} < 8=2^3$



# Les compteurs

Table de transition

| $Q_2\ Q_1\ Q_0$ | $Q_2^+\ Q_1^+\ Q_0^+$ | $J_2\ K_2$ | $J_1\ K_1$ | $J_0\ K_0$ |
|---|---|---|---|---|
| 0 0 0 | 0 0 1 | 0 X | 0 X | 1 X |
| 0 0 1 | 0 1 0 | 0 X | 1 X | X 1 |
| 0 1 0 | 0 1 1 | 0 X | X 0 | 1 X |
| 0 1 1 | 1 0 0 | 1 X | X 1 | X 1 |
| 1 0 0 | 1 0 1 | X 0 | 0 X | 1 X |
| 1 0 1 | 0 0 0 | X 1 | 0 X | X 1 |
| 1 1 0 | X X X | X X | X X | X X |
| 1 1 1 | X X X | X X | X X | X X |

Equations d'entrée :

**$J_0 = 1$**

**$K_0 = 1$**

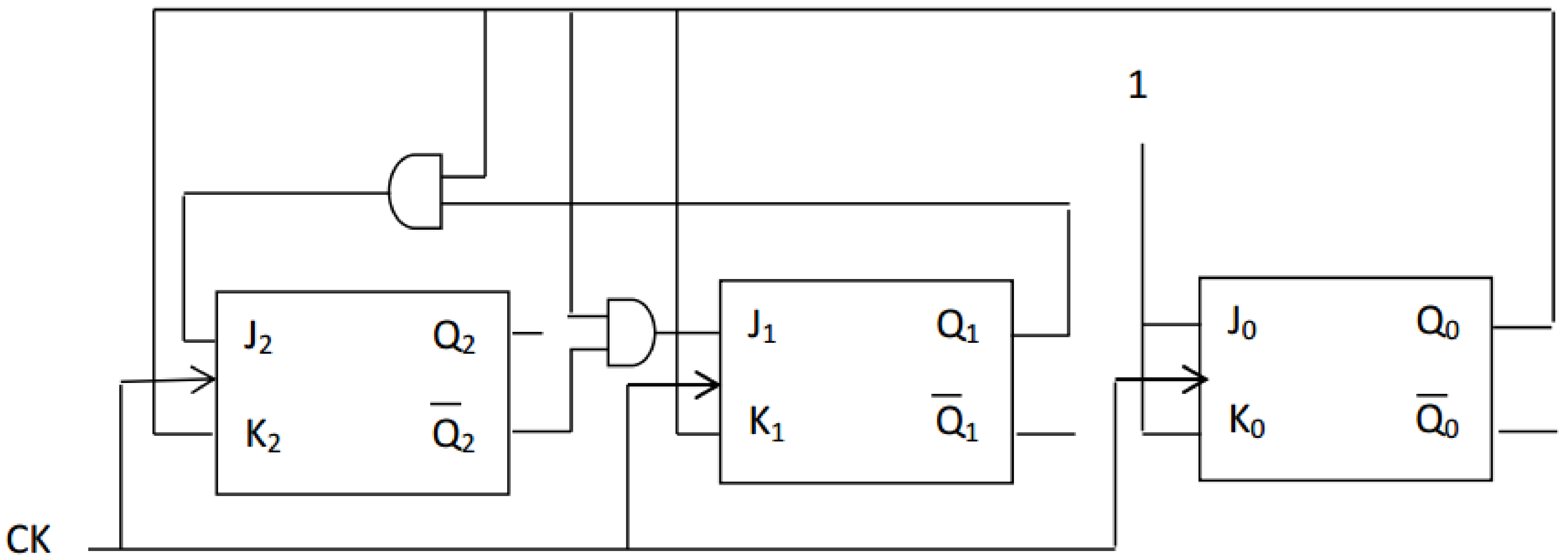
**$J_1 = \overline{Q_2}Q_0$**

**$K_1 = Q_0$**

**$J_2 = Q_1Q_0$**

**$K_2 = Q_0$**

# Les compteurs




à cycle complet

à cycle incomplet

à cycle complet

à cycle incomplet

# Les compteurs

## Les compteurs asynchrones

Un compteur asynchrone est un circuit séquentiel ou toutes les bascules n'ont pas la même horloge.
L'horloge de chaque bascule dépend de la sortie de la bascule qui la précède.

Les entrées des bascules sont conçues de telle sorte que leurs sorties soient inversées à chaque impulsion d'horloge ;
par exemple pour les bascules JK on prendra J = 1 et K = 1 ($Qt+1 = Qt$)

Pour réaliser le circuit d'un compteur asynchrone, on trace son chronnogramme puis on en déduit le circuit

**Compteur asynchrone à cycle complet**

Comme le compteur synchrone, le compteur asynchrone à cycle complet est un compteur modulo n tel que $n = 2^p$

Exemple : compteur asynchrone modulo 8 avec horloge à front montant.
Pour un compteur modulo 8 on a besoin de 3 bascules car $8 = 2^3$

Le compteur est asynchrone donc on a :
$J_2 = K_2 = 1$ $J_1 = K_1 = 1$ $J_0 = K_0 = 1$

000 -> 001 -> 010 -> 011 -> 100 -> 101->110 -111 ->000

# Les compteurs

**Réalisation du chronogramme**

1) Tracer les fonctions Q0 Q1 Q2 qui réalisent la séquence du compteur.

2) Déterminer les horloges de chaque fonction Qi

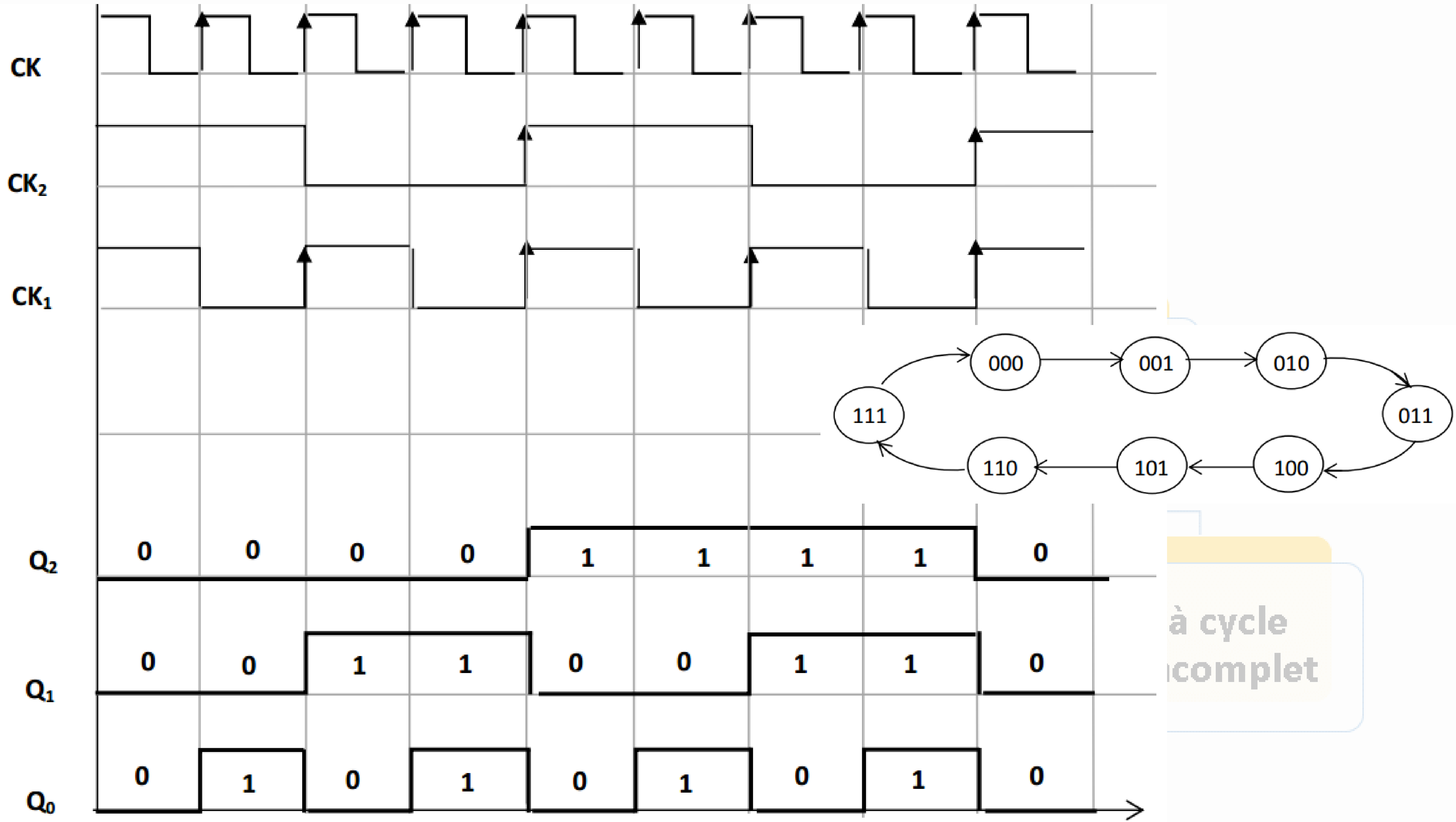
**-** Le changement d'état de Q0 se fait à chaque impulsion d'horloge donc CK0 = CK

**-** Le changement d'état de Q1 se fait toutes les 2 périodes donc on aura un front montant d'horloge toutes les 2 périodes ; cette horloge correspond à Q0 donc CK1 = Q0

**-** Le changement d'état de Q2 se fait toutes les 4 périodes donc on aura un front montant d'horloge toutes les 4 périodes ; cette horloge correspond à Q1 donc CK2 = Q1

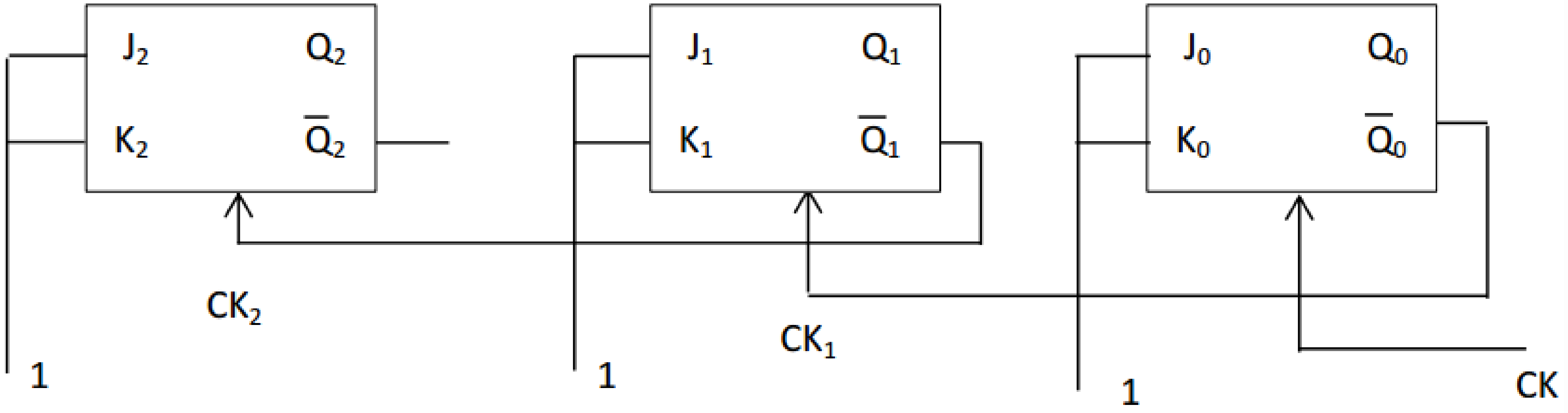
Pour un compteur asynchrone modulo 8 on aura donc :

**CK0 = CK** **CK1 = $\overline{Q0}$** **CK2 = $\overline{Q1}$**



CK
$CK_2$
$CK_1$
$Q_2$
0 0 0 0 1 1 1 1 0
$Q_1$
0 0 1 1 0 0 1 1 0
$Q_0$
0 1 0 1 0 1 0 1 0
000
001
010
011
100
101
110
111
à cycle
complet

## Circuit




Pour un compteur asynchrone de N bascules avec horloge à front montant on aura :

**$CK_0 = CK$ et $CK_i = \overline{Q_{i-1}}$**

# Les compteurs

**Décompteur asynchrone à cycle complet**

On utilise la même méthode que pour le compteur.
Pour un décompteur asynchrone de N bascules avec horloge à front montant

on aura :
**$CK_0 = CK$ et $CK_i = Q_i$**

à cycle complet

à cycle incomplet

à cycle complet

à cycle incomplet

# Les compteurs

**Compteur asynchrone à cycle incomplet**

Un compteur à cycle incomplet est un compteur à n états tel que $\mathbf{2^{p-1} < n < 2^{p}}$ (**n** n'est pas une puissance de 2).
Le nombre de bascules de ce compteur est égal à **p**
Le principe consiste à remettre le compteur à zéro dès qu'il arrive à n-1

Exemple : Le compteur modulo 6 (0→5) $4=2^2 < \mathbf{6} < 8=2^3$
Le compteur est doté d'une fonction Clear qui le remet à zéro dès qu'il arrive à 5

# Les compteurs

| $Q_2$ $Q_1$ $Q_0$ | $Q_2^+$ $Q_1^+$ $Q_0^+$ | Clear |
|---|---|---|
| 0 0 0 | 0 0 1 | 0 |
| 0 0 1 | 0 1 0 | 0 |
| 0 1 0 | 0 1 1 | 0 |
| 0 1 1 | 1 0 0 | 0 |
| 1 0 0 | 1 0 1 | 0 |
| **1 0 1** | **0 0 0** | **1** |
| 1 1 0 | X X X | 0 |
| 1 1 1 | X X X | 0 |

Le compteur se remet à zéro lorsqu'il arrive à 5.

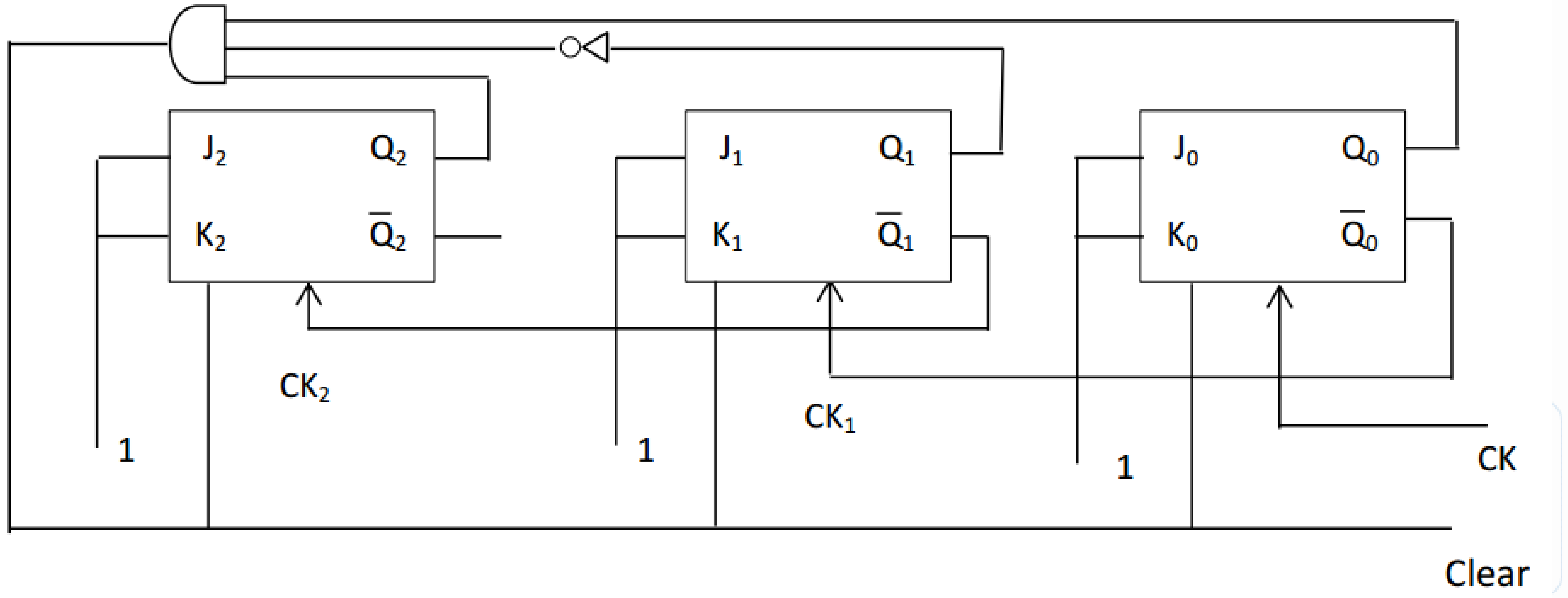
La fonction Clear est donc activée lorsque le compteur arrive à 5.

L'équation de la fonction Clear est donnée par la table de vérité :

**Clear = $Q2\,\overline{Q1}\,Q0$**

# Les compteurs

Circuit :



# Chapitre 2 : Les mémoires

## Définition

Une mémoire est un dispositif capable de stocker et conserver des informations de telle sorte qu’un utilisateur puisse y accéder à n’importe quel moment.

# Les mémoires

## Hiérarchie des mémoires

Les éléments mémoire d'un ordinateur se répartissent en plusieurs niveaux caractérisés par leur capacité et leur temps d'accès.

**La Mémoire Centrale :** C'est l'organe principal de rangement des informations utilisées par le CPU (Processeur).
Pour exécuter un programme il faut le charger en mémoire centrale (instructions et données).

**La Mémoire Cache :** C'est une mémoire de faible capacité utilisée comme mémoire tampon entre le CPU et la mémoire centrale. Elle permet au CPU de faire moins d'accès à la mémoire centrale et ainsi de gagner du temps.

# Les mémoires

## Hiérarchie des mémoires (suite)

**Les Mémoires Auxiliaires :** Les mémoires auxiliaire sont appelées aussi mémoires de masses, ce sont de mémoires périphériques de grande capacité et de coût relativement faible. Elles servent d'élément de stockage permanent.

**La Mémoire d'appui :** C'est une mémoire intermédiaire entre la mémoire centrale et les mémoires auxiliaires. Elle permet d'augmenter la vitesse d'échange des informations entre ces deux niveaux.

# Les mémoires

## Organisation de l’information

Les informations d’un ordinateur doivent s’adapter à un certain format dont les caractéristiques générales sont présentées dans ce qui suit :

**Le bit :** Le bit constitue l’unité de base de l’information. Dans une mémoire, le plus petit élément de stockage est appelé point mémoire, il mémorise un bit d’information

**L’octet (byte) :** L’unité de base, il correspond à un groupement de 8 bits.

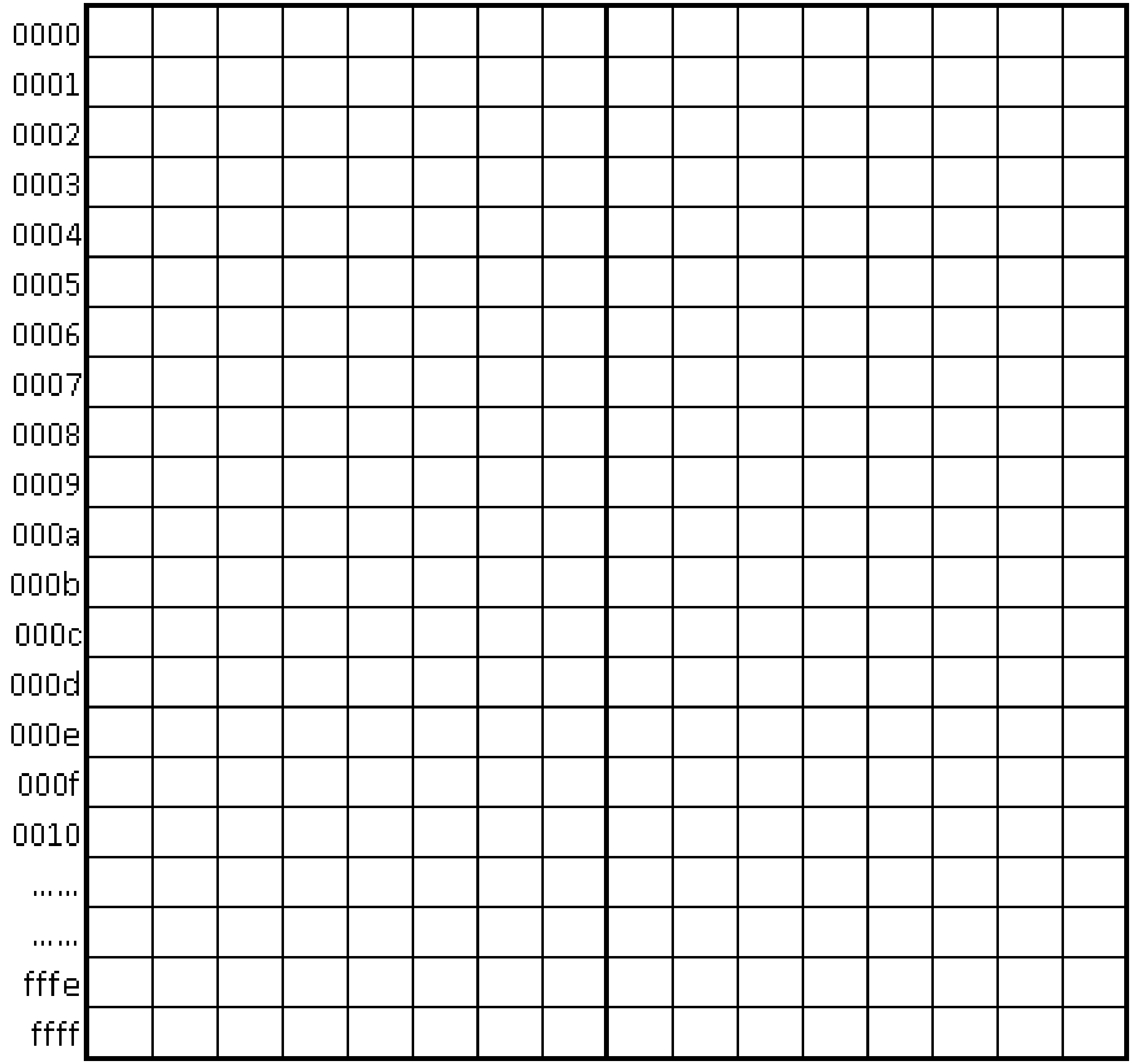
**Le caractère :** Le caractère est un groupement de 8 bits permettant le codage d’un caractère alpha numérique ou d’un caractère spécial selon la convention de codage ( en code ASCII un caractère est codé sur un octet)

# Les mémoires

## Organisation de l'information

**Le mot :** Le mot est un ensemble de bits constituant une unité d'information adressable en mémoire centrale. Il est constitué de 2, 4 ou 8 octet selon la configuration de l'ordinateur.

**L'adresse :** L'adresse est la valeur numérique désignant un élément de mémoire (mot mémoire)



# Les mémoires

## Caractéristiques des mémoires

**La capacité :** La capacité mémoire est la taille de la mémoire, elle correspond au nombre d’informations qu’elle peut contenir.

On peut exprimer cette valeur en fonction du nombre de bits, du nombre d’octets ou du nombre de mots.

Unités de mesure de la capacité d’une mémoire :

1 Ko = 1 Kilo Octet = $2^{10}$ octet

1 Mo = 1 Méga Octet = $2^{20}$ octet

1 Go = 1 Giga Octet = $2^{30}$ octet

1 To = 1 Tera Octet = $2^{40}$ octet

**Le temps d’accès:** Le temps d’accès d’une mémoire est le temps qui s’écoule entre le lancement d’une opération (lecture ou écriture) et le début de son accomplissement.

# Les mémoires

## Caractéristiques des mémoires (suite)

**Le débit :** Le débit est le volume d'informations échangé par unité de temps, il est exprimé en bits par seconde

**La volatilité :** La volatilité caractérise l'aptitude d'une mémoire à conserver ou non les données lorsqu'elle n'est plus alimentée électriquement.
Une mémoire volatile perd son contenu lorsqu'elle n'est plus alimentée par le courant électrique. La mémoire vive est volatile alors que les mémoires auxiliaires telles que le disque dur ne sont pas volatiles.

**L'accès séquentiel :** L'accès séquentiel est relativement long car pour accéder à une donnée on doit parcourir toutes les informations qui la précèdent

**L'accès direct :** Dans ce cas les informations ont une adresse propre grâce à laquelle on peut y accéder directement.

## La mémoire vive (RAM)

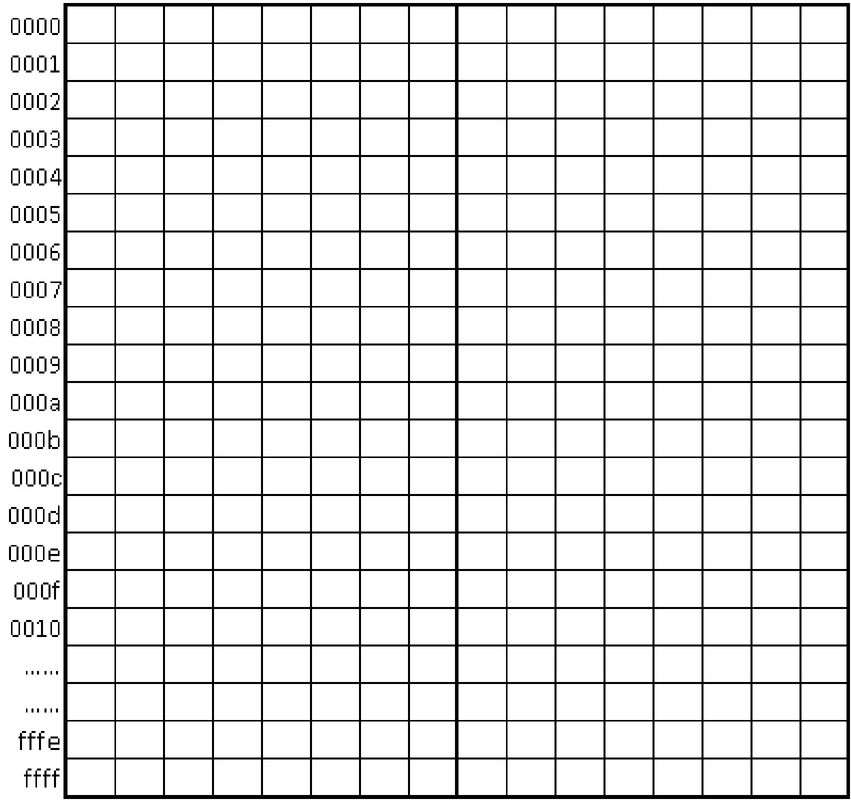
La mémoire vive appelée aussi RAM (Random Access Mémory) est une mémoire à accès direct.

C’est la mémoire dans laquelle l’ordinateur place les données lors de leur traitement (lecture ou écriture).

La RAM fonctionne seulement lorsque l’ordinateur est allumé.

La RAM est constituée de m mots de n bits.

L’expression qui représente cette RAM est : **RAM m x n**



## La mémoire vive (RAM)

A chaque RAM sont associés les éléments suivants :

Le **CS** (Chip Select) : c'est l'entrée qui permet d'activer ou de désactiver la RAM

Le **MAR** (Memory Address Register): c'est le registre d'adresse mémoire. Il contient une adresse qui nous permet d'accéder directement à un mot précis de la RAM grâce à un décodeur.

Le **MBR** (Mémory Buffer Register): c'est le registre mot ou registre de données. C'est un registre tampon par lequel passent toutes les données allant de la RAM vers le CPU est inversement.

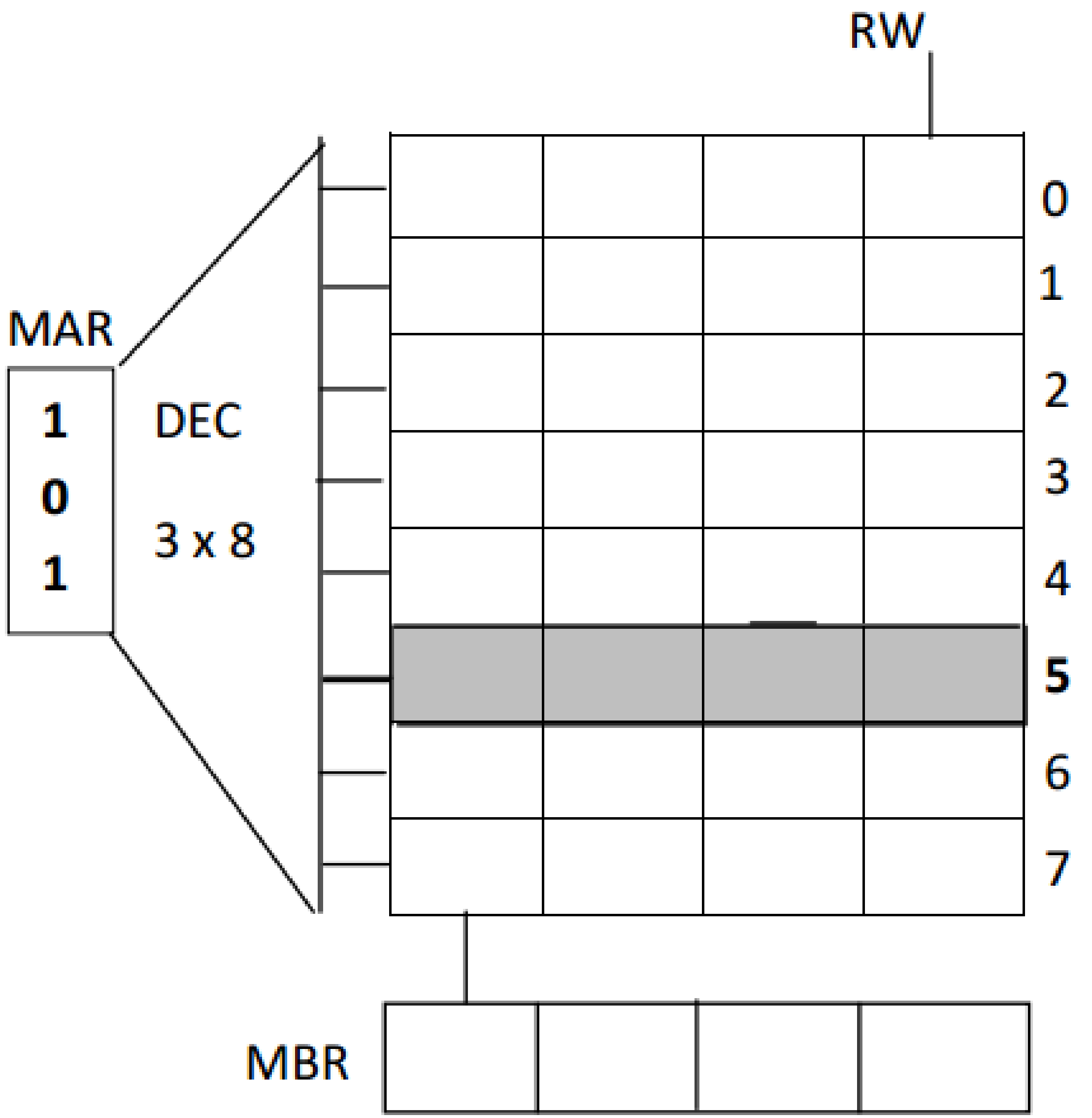
Le $\boldsymbol{R/\overline{W}}$ (Read/Write) : c'est la fonction de lecture écriture ( on utilisera dans ce cours RW= 1 pour la lecture et RW = 0 pour l'écriture)

# Les mémoires

Le MAR de cette RAM affiche la valeur (101) c'est donc l'adresse 5 de la RAM qui sera sélectionnée à l'aide du DEC 3x8.

Si la fonction RW = 1 il s'agit d'une **lecture** ; le mot d'adresse 5 sera transférée dans le MBR.

Si la fonction RW = 0 il s'agit d'une **écriture**; le mot qui se trouve dans le MBR sera transféré dans la mémoire à l'adresse 5



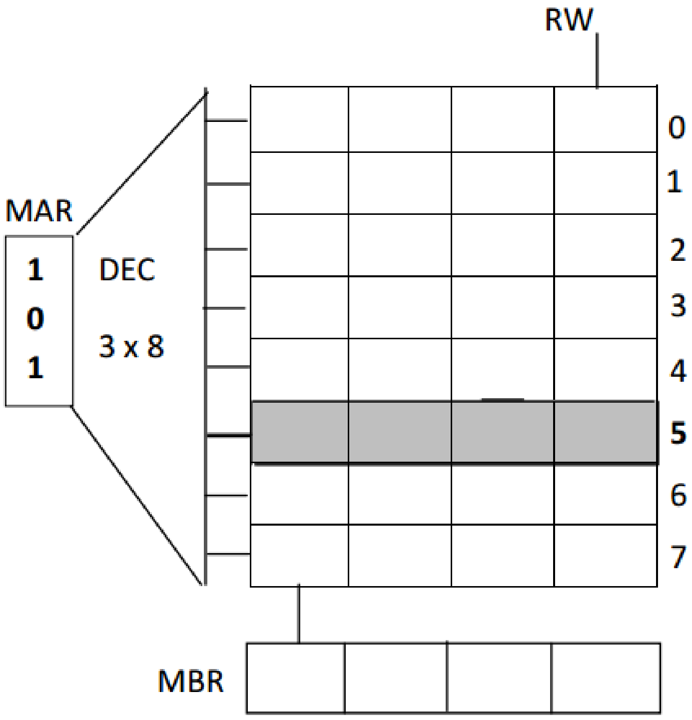
# Les mémoires

Le transfert d’informations entre la RAM et les registres se fait au moyen de lignes de bus ;

on a un bus d’adresse entre le MAR et la RAM et un bus de données entre le MBR et la RAM.

Le nombre total de bits de cette RAM est 8x4 c’est-à-dire 32 bits.

Chaque bit représente une cellule binaire CB et chaque cellule binaire possède 3 entrées et une sortie.



# Les mémoires

Définition d'un bus :

Un bus d'ordinateur est un moyen de transférer des informations depuis une partie de l'ordinateur vers une autre, il contient autant de lignes que les registres qui l'utilisent.

Le bus de données est bidirectionnel, il est composée de 2 lignes jumelées qui permettent de fonctionner dans les 2 sens (lecture et écriture)

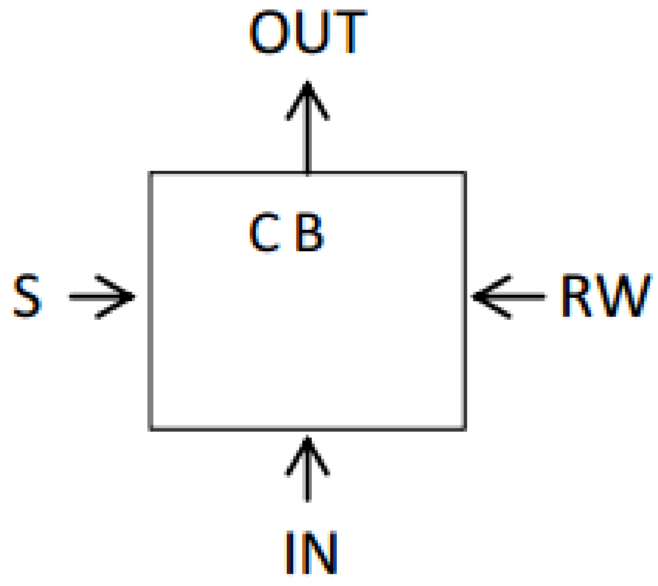
# Les mémoires

**S** représente la ligne du décodeur qui sélectionne la cellule binaire**.**

**RW** est la fonction de lecture /écriture

**IN** (Input) représente l'information à écrire dans la cellule binaire.

**OUT** (Output) représente l'information lire à partir de la cellule binaire.



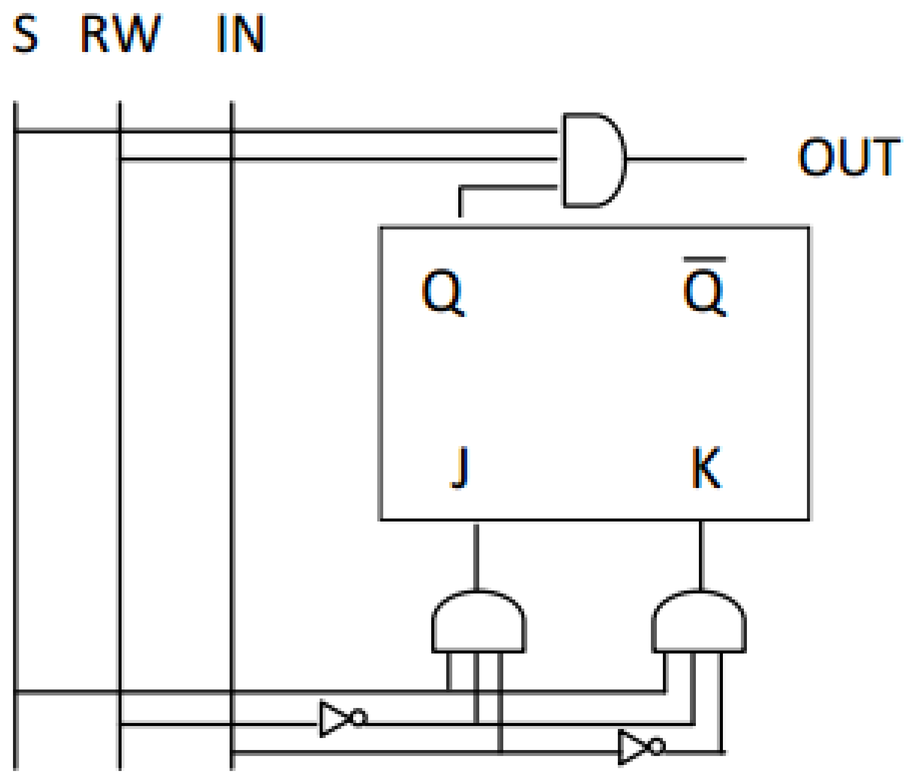
**Représentation d'une cellule binaire à l'aide d'une bascule JK :**

1) **S = 0** la cellule binaire n'est pas sélectionnée
Il n'y a pas de changement au niveau de la bascule JK
J = 0 et K = 0

2) **S = 1** La cellule binaire est sélectionnée (donc active)
Si RW = 0 on a une opération d'écriture .
Si RW = 1 on a une opération de lecture

**Table de Vérité de la CB :**

| S | RW | IN | J | K |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 1 | 0 | 0 |
| 1 | 0 | 0 | 0 | 1 |
| 1 | 0 | 1 | 1 | 0 |
| 1 | 1 | 0 | 0 | 0 |
| 1 | 1 | 1 | 0 | 0 |



# Extension de la mémoire

**Extension de la mémoire**

Pour réaliser des mémoires **de grande capacité** on peut utiliser des mémoires **de petites capacités** et les connecter entre elles.

Si on veut augmenter **le nombre de mots** on connecte les mémoires en parallèle en augmentant le bus d'adresse.

Si on veut augmenter **la longueur du mot** mémoire on connecte les mémoires en série en augmentant le bus de données.

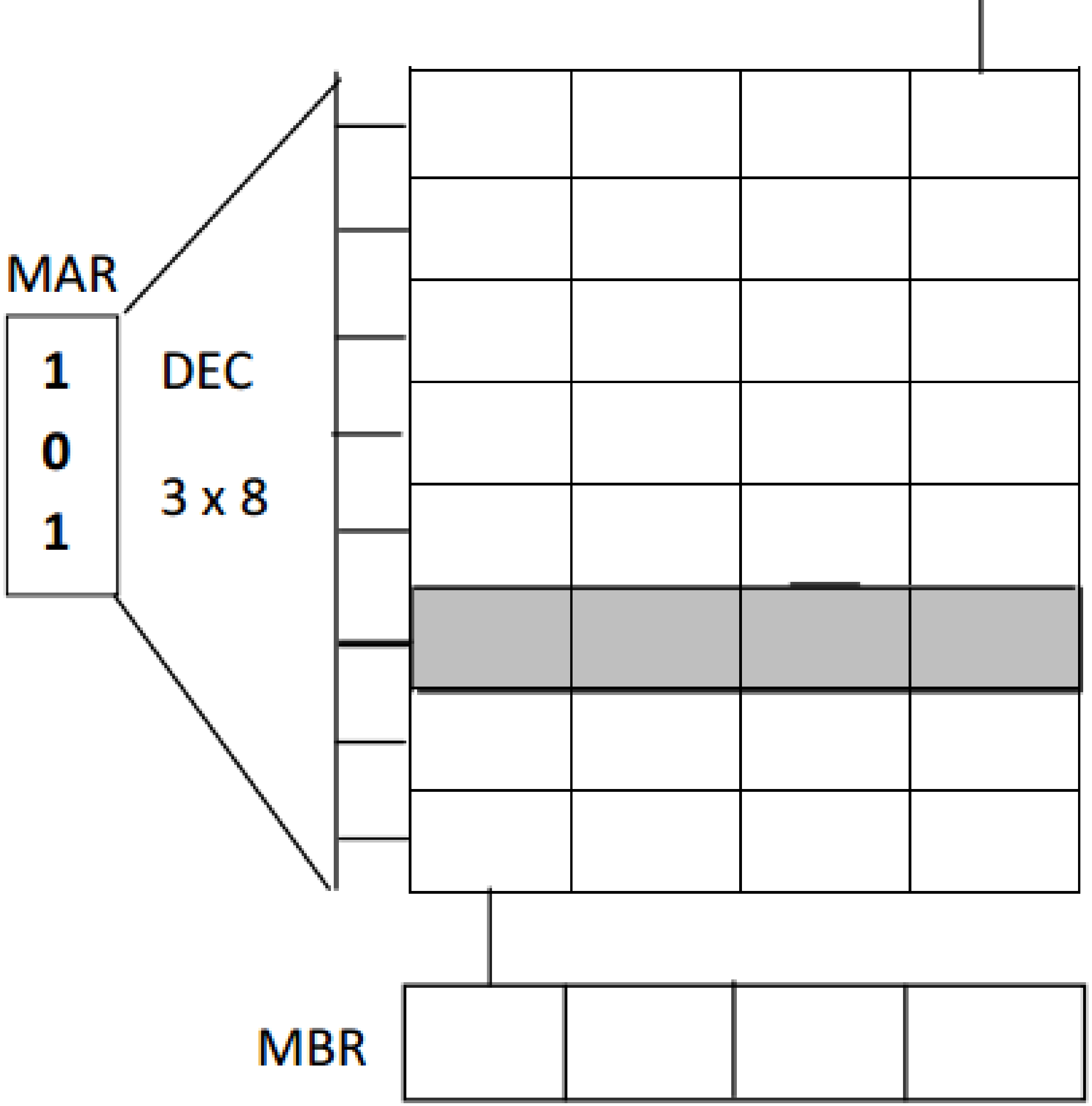
## Augmentation de l'espace d'adressage :

Si on veut augmenter l'espace d'adressage
en parallèle ; pour cela il faut qu'aies aien

Pour la réalisation **d'une RAM ($2^m$ x P)** à l'a
nombre de RAM ($2^n$ x P) nécessaires.

Il suffit de diviser la capacité de la RAM de
**($2^m$ x P) / ($2^n$ x P) = $2^{m-n}$**

On utilisera un registre d'adresse (MAR) de
**(m-n)** adresses complémentaires.

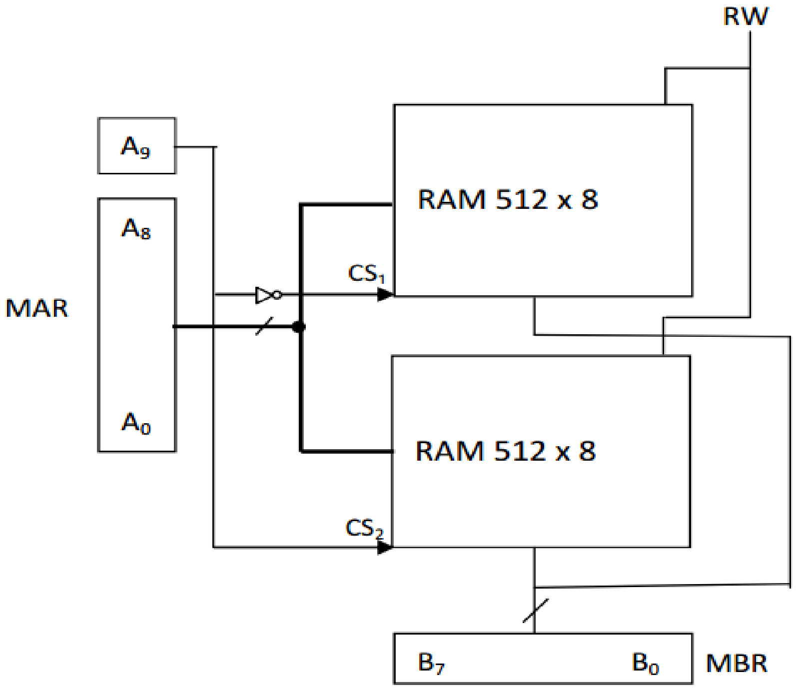


# Extension de la mémoire

Exemple 1 : **m = 10 n = 9 et p = 8**
On veut réaliser une **RAM $2^{10}$x 8** à partir de RAM **RAM $2^{9}$ x 8**

Le nombre de RAM nécessaires est :
**$2^{10}$ x8/ $2^{9}$ x8= 2 RAM**

On aura **un MAR** de **9 bits** et **(m-n)=1 bit** d'adresse complémentaire, **le MBR** contient **8 bits**.



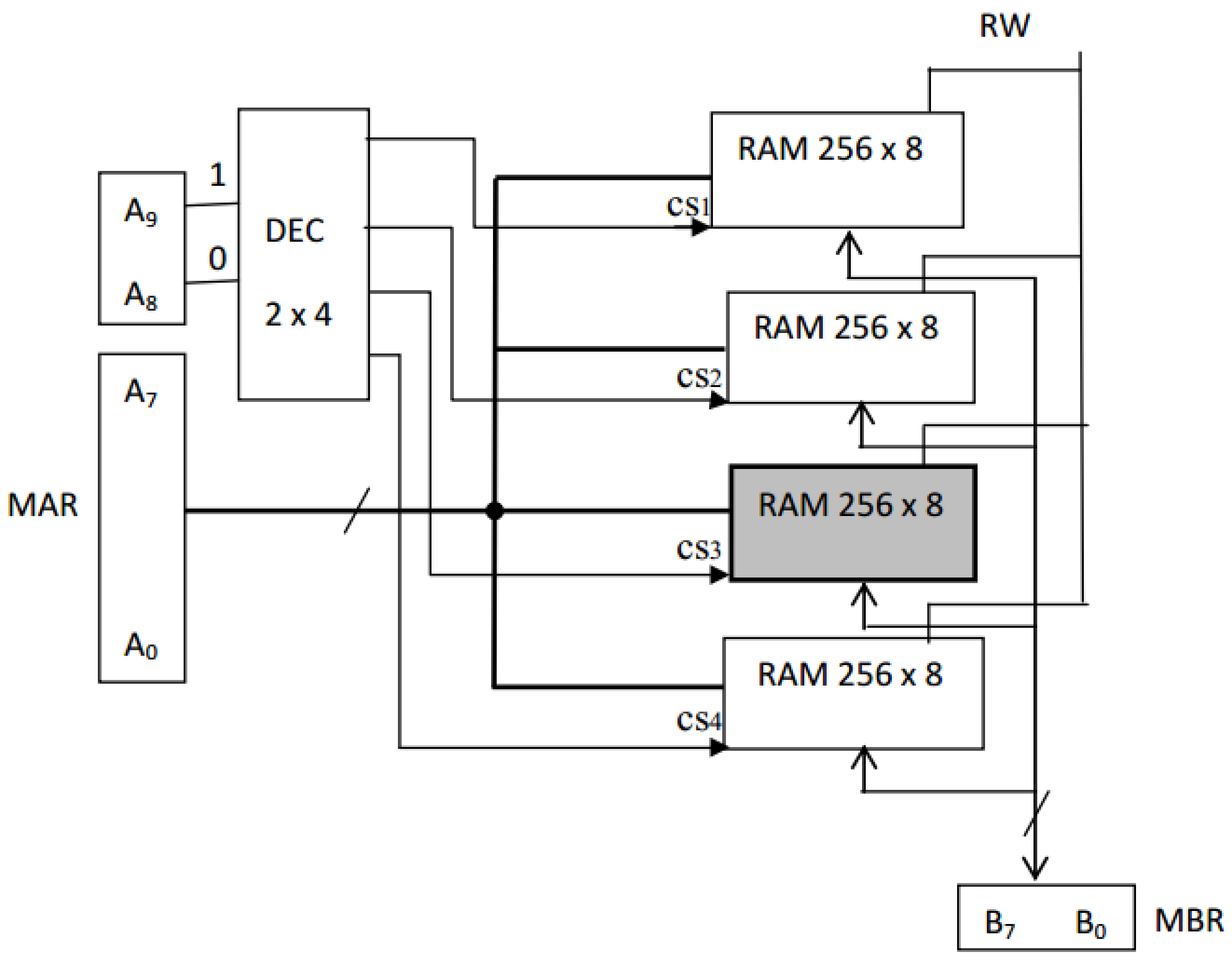
# Extension de la mémoire

Exemple 2 : **m = 10 n = 8 et p = 8**
On veut réaliser une **RAM 1K x 8** à partir de **RAM 256 x 8**

On veut réaliser une RAM de $2^{10}$ x 8

À l'aide des RAMs de $2^{8}$ x 8

RAMs utilisées = $2^{10}$ x 8/ $2^{8}$ x 8 = $2^{10-8}$ = 4
M-n bits comp = 2,



## Extension de la mémoire

**Augmentation de la longueur du mot :**

Si on veut augmenter la longueur d'un mot mémoire il faut connecter plusieurs **boitiers en série** ; pour cela il faut qu'ils aient le même nombre de mots.

Pour la réalisation d'une **RAM ($2^n$ x P)** à l'aide de **RAM ($2^n$ x Q)** il faut d'abord définir le nombre de RAM **($2^n$ x Q)** nécessaires. Il suffit de **diviser P par Q**

On utilisera **un seul registre d'adresse (MAR)** de **n bits**, **un registre de données** virtuel de **P bits** composé de plusieurs registres de **Q bits** (le nombre de ces registres est égal à **P/Q** ) et un même Chip Select pour les **2 RAM**

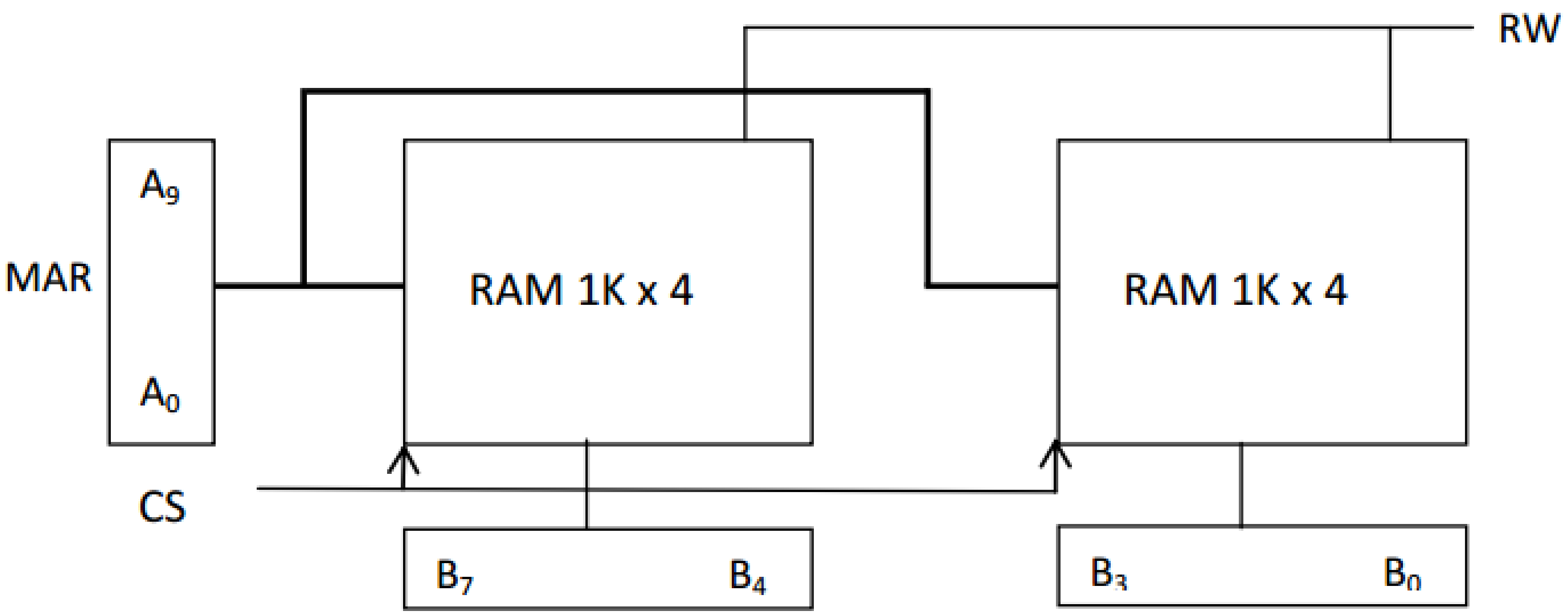
## Extension de la mémoire

Exemple : **n = 10 P = 8 et Q = 4**

On veut réaliser une **RAM $2^{10}$ x 8** à partir de **RAM $2^{10}$ x 4**

Le nombre de RAM nécessaires est : **8 / 4 = 2 RAM**

On aura un MAR de 10 bits et le MBR contient **2 x 4 = 8 bits**.



# Extension de la mémoire

**Augmentation de l'espace d'adressage et de la longueur du mot :**

Pour la réalisation d'une **RAM ($2^m$ x P)** à l'aide de **RAM ($2^n$ x Q)** il faut d'abord définir le nombre total de **RAM ($2^n$ x Q)** nécessaires. Pour cela on divise le nombre de bits de la RAM qu'on veut réaliser par le nombre des RAM proposées :

**(2m x P) / (2n x Q)** = **2m-n x P/Q**

On aura ainsi **$2^{m-n}$ RAM en parallèle** composées chacune de **(P/Q) RAM en série**.

Remarque : On divise les nombres de mots et les longueurs de mots séparément.

# Extension de la mémoire

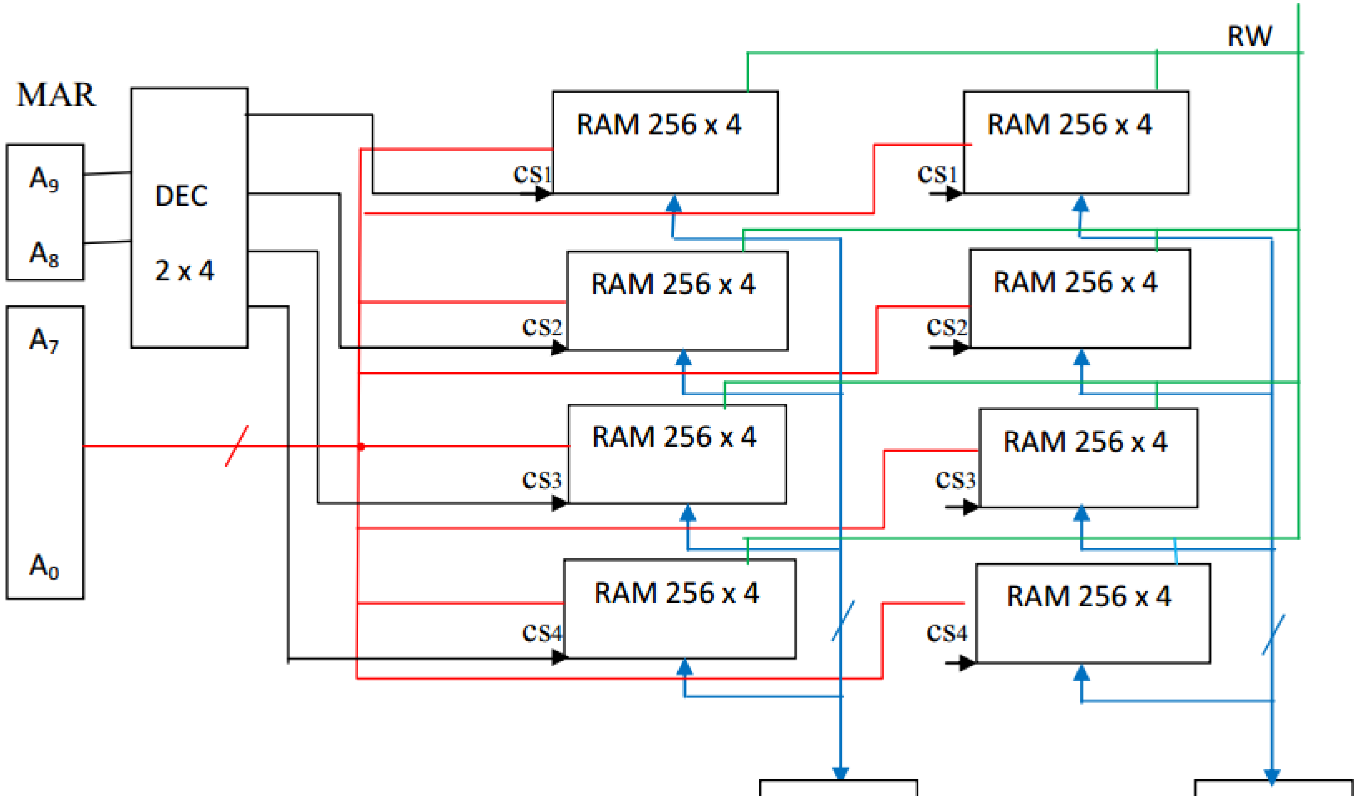
Exemple : **m = 10 P = 8 et n = 8 Q = 4**

On veut réaliser une **RAM $2^{10}$ x 8** à partir de **RAM $2^{8}$ x 4**

Le nombre de RAM nécessaires est :

**Parallèles = $2^{10}/2^{8}$ = 4,**

**Série = 8/4 = 2,**



# Les mémoires

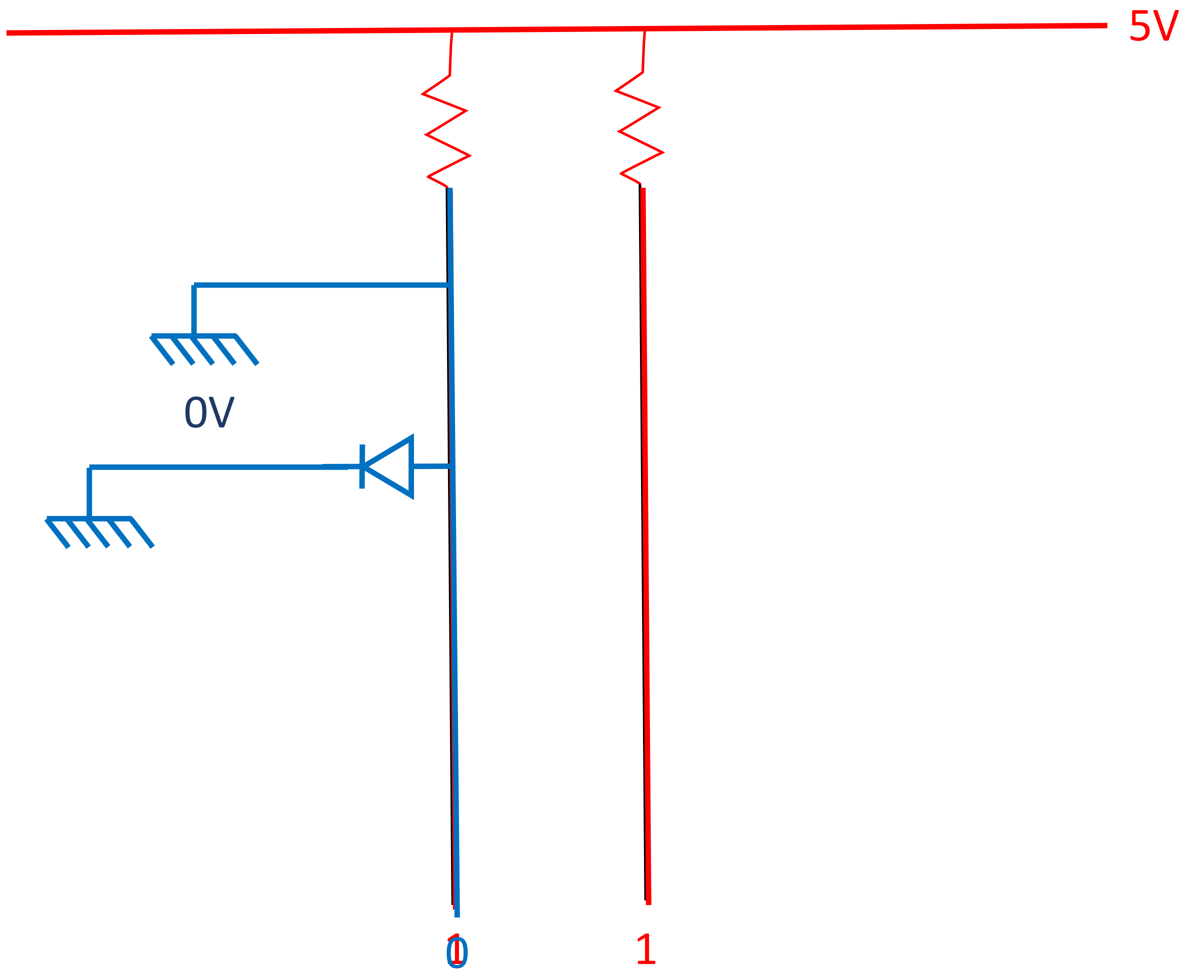
## Mémoires mortes ROM

La ROM (Read Only Memory) est une mémoire permanente non volatile et en lecture seule contrairement à la RAM.

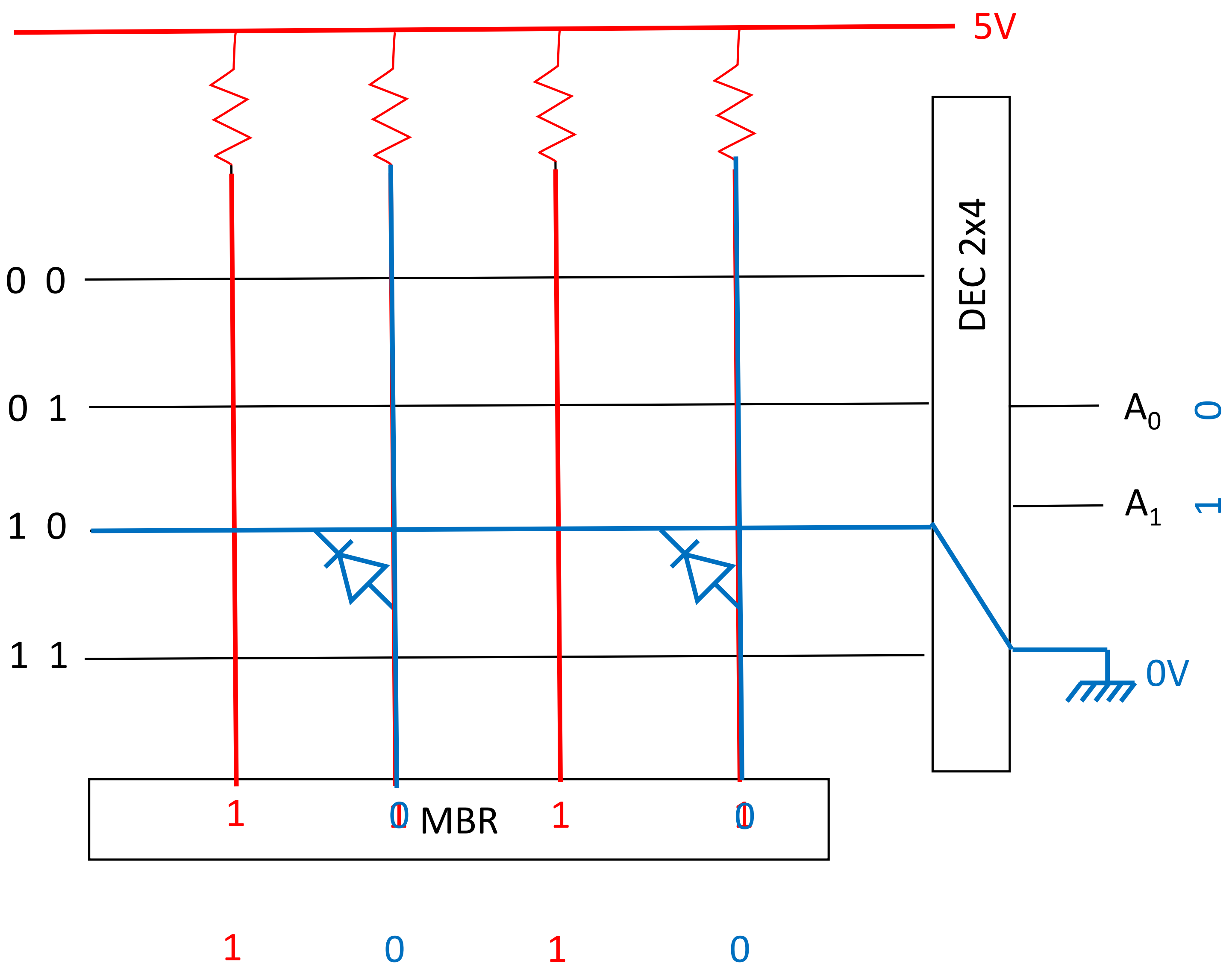
Ce type de mémoire est utilisé pour conserver des informations qui ne s'effacent jamais même quand il n'y a plus de mise sous tension. Elle est cependant plus lente que la RAM.

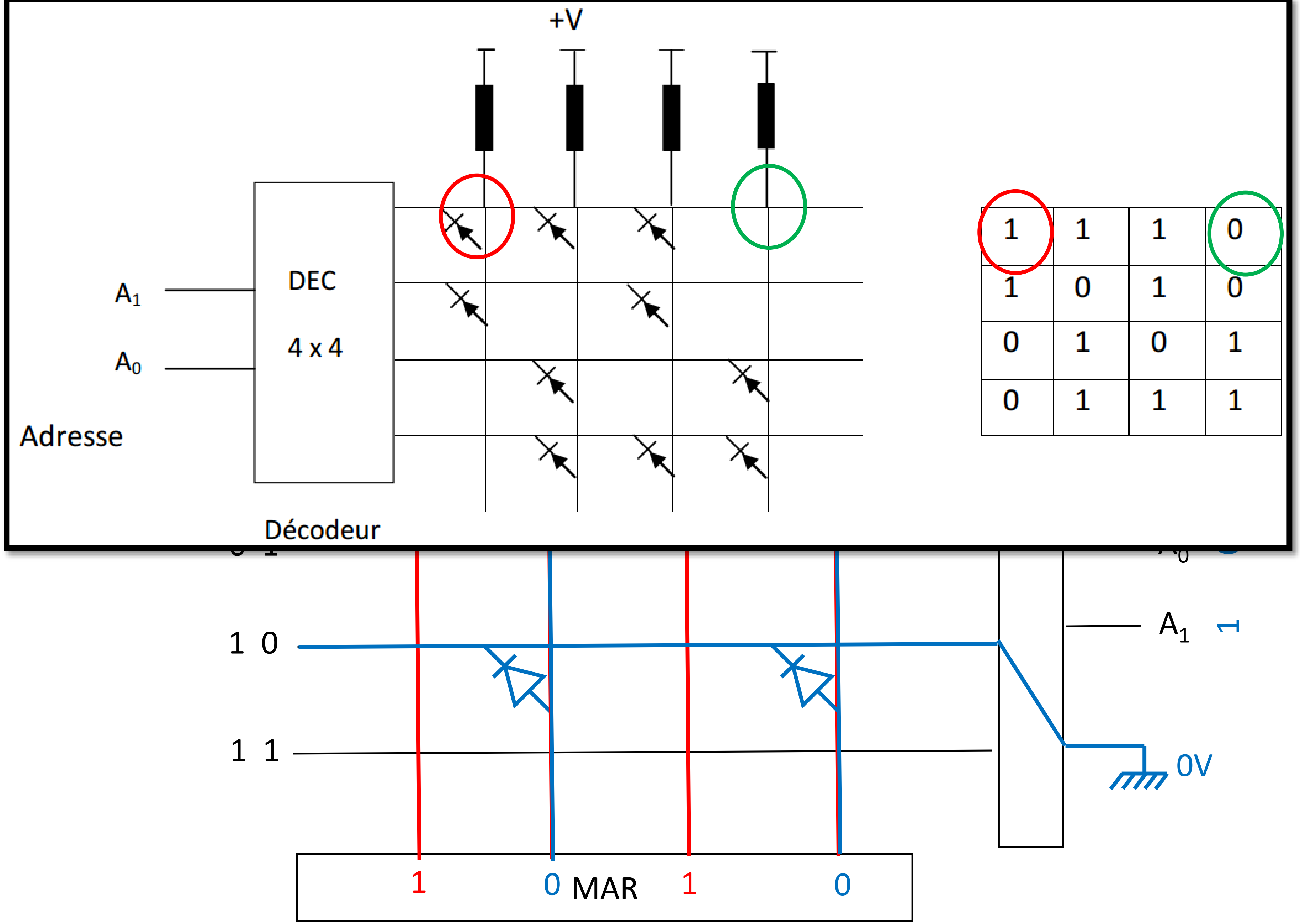
La ROM est réalisée à partir de diodes munies de petits fusibles. Lorsque le fusible est intact la diode laisse passer le courant, elle représente alors un 1, lorsqu'on claque le fusible, le courant ne passe plus, nous avons alors un 0

# ROM



# ROM




+V
DEC
4 x 4
A1
A0
Adresse
Décodeur
1 1 1 0
1 0 1 0
0 1 0 1
0 1 1 1
1 0
1 1
A1
1
0V
1
0
MAR
1
0

## Mémoires mortes ROM

La ROM permet notamment de conserver les données nécessaires au démarrage de l'ordinateur (le BIOS) ; elle permet également dans certains cas de réaliser certains circuits combinatoires (fonctions logiques).

Il existe plusieurs types de ROM :

**Les ROM :** leur contenu est défini lors de la fabrication, il est non modifiable.

**Les PROM (Programmable Read Only Memory) :** elles sont programmables une seule fois par l'utilisateur.

**Les EPROM (Erasable programmable Read Only Mémory) :** elles sont effaçables et. reprogrammables. La réinitialisation se fait en reconstituant les fusibles après exposition aux rayons UV.

## Mémoires mortes ROM

### Exemple de ROM :

On se limitera dans ce cours à la représentation de fonctions de circuits combinatoires.

Dans ce cas les bits d’adresse seront représentés par les variables d’entrées et les lignes de données par les fonctions de sortie.

On prendra comme exemple le circuit de l’additionneur complet à un bit.

| A | b | c | S | R |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 | 0 |
| 0 | 1 | 0 | 1 | 0 |
| 0 | 1 | 1 | 0 | 1 |
| 1 | 0 | 0 | 1 | 0 |
| 1 | 0 | 1 | 0 | 1 |
| 1 | 1 | 0 | 0 | 1 |
| 1 | 1 | 1 | 1 | 1 |

# Les mémoires

| A | b | c | S | R |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 | 0 |
| 0 | 1 | 0 | 1 | 0 |
| 0 | 1 | 1 | 0 | 1 |
| 1 | 0 | 0 | 1 | 0 |
| 1 | 0 | 1 | 0 | 1 |
| 1 | 1 | 0 | 0 | 1 |
| 1 | 1 | 1 | 1 | 1 |