

Partiel 1 – Corrigé

Architecture des ordinateurs

Durée : 1 h 30

Exercice 1 (2 points)

1. Convertissez, **en détaillant chaque étape**, le nombre 163,5625 dans le format flottant IEEE 754 **simple précision**. Vous exprimerez le résultat final sous forme binaire, **en précisant chacun des champs**.

- $S = 0$
- $0,5625 \times 2 = 1,125$
 $0,125 \times 2 = 0,25$
 $0,25 \times 2 = 0,5$
 $0,5 \times 2 = 1$
 $|163,5625| = 163,5625 = 1010\ 0011,1001_2$
- $163,5625 = (1,01000111001)_2 \cdot 2^7$
 $M = 010001110010\dots0_2$ et $e = 7$
- $E = e + \text{biais} = 7 + 127 = 6 + 128$
 $E = 1000\ 0110_2$
- **163,5625 \rightarrow 0 10000110 010001110010000000000000**

2. Donnez, **en détaillant au maximum**, la représentation associée au nombre ci-après. Ce nombre est codé au format flottant IEEE 754 **double précision** : 7FF0 0000 0000 0000₁₆

- = 0111 1111 1111 0000.....0₂
- $S = 0$
 $E = 111\ 1111\ 1111_2$
 $M = 0$
 - $+\infty$

Exercice 2 (2 points)

Soit les quatre figures ci-dessous :

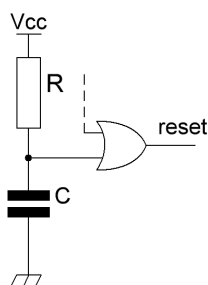


Figure 1

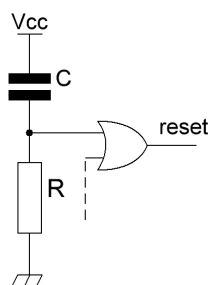


Figure 2

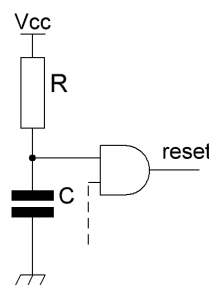


Figure 3

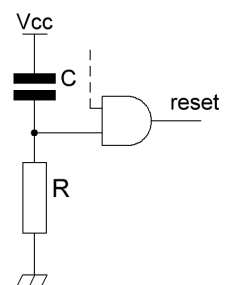


Figure 4

1. Lors d'une mise sous tension, on souhaite obtenir un **état haut** sur le *reset* pendant un court laps de temps. Choisissez la bonne figure parmi les quatre.

Il s'agit de la figure 2. À l'allumage, la tension aux bornes du condensateur est nulle, ce qui entraîne une tension V_{cc} aux bornes de la résistance et donc à l'entrée de la porte OU. Ce 1 est recopié sur la sortie. Une fois le condensateur chargé, la tension à ses bornes est V_{cc} et la tension aux bornes de la résistance est nulle : la sortie de la porte OU prend le niveau logique de sa seconde entrée.

2. Lors d'une mise sous tension, on souhaite obtenir un **état bas** sur le *reset* pendant un court laps de temps. Choisissez la bonne figure parmi les quatre.

Il s'agit de la figure 3. À l'allumage, la tension aux bornes du condensateur est nulle, ce qui entraîne un niveau logique 0 en entrée de la porte ET. Ce 0 est recopié sur la sortie. Puis le condensateur se charge jusqu'à atteindre V_{cc} . Le niveau logique considéré par l'entrée de la porte est donc 1 : la sortie de la porte ET prend le niveau logique de sa seconde entrée.

Exercice 3 (5 points)

Soit les deux figures ci-dessous :

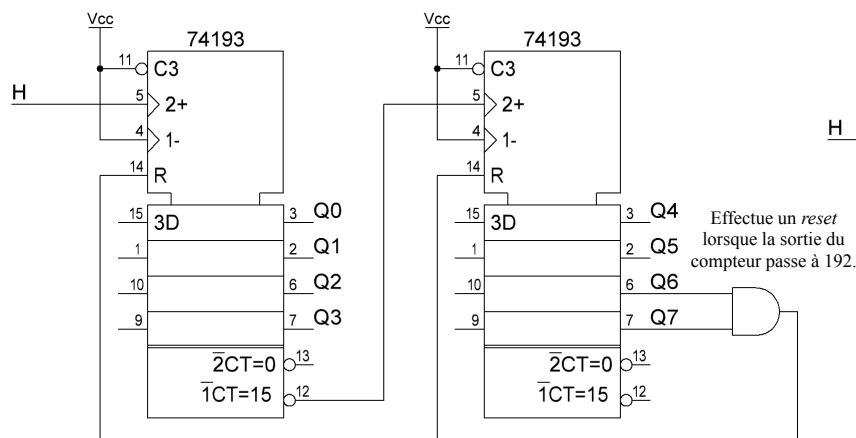


Figure 1

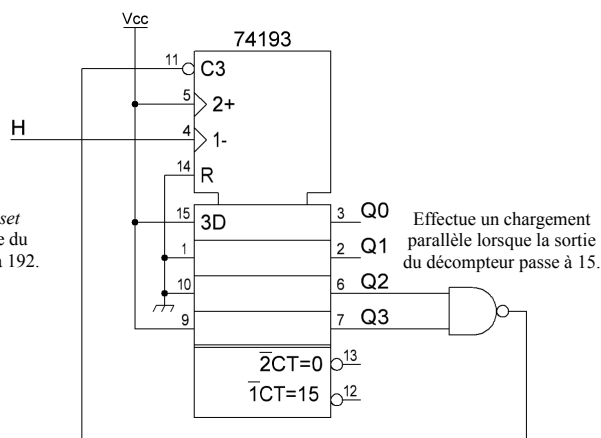


Figure 2

- La [documentation technique](#) du 74193 est fournie en annexes.
- Les entrées **H** sont des entrées d'horloge.
- Le montage de la figure 1 possède 8 sorties : de **Q0** à **Q7** (**Q0** étant le bit de poids faible).
- Le montage de la figure 2 possède 4 sorties : de **Q0** à **Q3** (**Q0** étant le bit de poids faible).
- Toutes les questions de cet exercice sont indépendantes.

1. Que réalise le montage de la figure 1 ?

Ce montage est un **compteur modulo 192**. Les deux 74193 sont câblés en mode compteur. La porte ET sert à détecter la valeur 192 et à la remplacer par la valeur 0. La valeur 192 est détectée au moment du passage à 1 des sorties **Q6** et **Q7**. La valeur 0 est forcée à l'aide d'un *reset*.

2. Que réalise le montage de la figure 2 ?

Ce montage est un **décompteur modulo 10**. Le 74193 est câblé en mode décompteur. La porte NON-ET sert à détecter la valeur 15 et à la remplacer par la valeur 9. La valeur 15 est détectée au moment du passage à 1 des sorties **Q2** et **Q3**. La valeur 9 est forcée à l'aide d'un chargement parallèle.

3. Câblez les bascules présentes sur le [document réponse](#) afin de réaliser un **décompteur asynchrone modulo 13**. Les bascules sont synchronisées sur fronts montants. Elles possèdent des entrées *set* et *reset* actives à l'état bas. Vous disposez de toutes les portes logiques nécessaires.

Il faut **détecter la valeur 15** (**Q1**, **Q2** et **Q3** suffisent) et **forcer la valeur 12** (**Q0 = Q1 = 0**, **Q2 = Q3 = 1**).

Exercice 4 (5 points)

On désire réaliser une ROM2 de 16 Kib avec un bus de donnée de 8 bits, à l'aide de plusieurs ROM1 de 4 Kib ayant un bus de donnée de 4 bits.

1. Donnez le nombre de fils du bus d'adresse de la ROM1.

$4 \text{ Kib} = (4 \text{ Ki} / 4) \text{ mots de 4 bits} = 1 \text{ Ki mots de 4 bits.}$

$1 \text{ Ki mots} = 2^{10} \text{ mots.}$

La ROM1 possède 10 fils d'adresse.

2. Donnez le nombre de fils du bus d'adresse de la ROM2.

$16 \text{ Kib} = (16 \text{ Ki} / 8) \text{ mots de 8 bits} = 2 \text{ Ki mots de 8 bits.}$

$2 \text{ Ki mots} = 2^{11} \text{ mots.}$

La ROM2 possède 11 fils d'adresse.

3. Combien de mémoires doit-on assembler en série ?

On passe d'une profondeur de 1 Ki mots à une profondeur de 2 Ki mots.

Il faut assembler 2 mémoires en série ($2 \text{ Ki} / 1 \text{ Ki} = 2$).

4. Combien de mémoires doit-on assembler en parallèle ?

On passe d'un bus de donnée de 4 bits à un bus de donnée de 8 bits.

Il faut assembler 2 mémoires en parallèle ($8 / 4 = 2$).

5. Combien de bits d'adresse vont servir à déterminer le CS de chaque ROM ?

C'est le bit d'adresse supplémentaire de la ROM2 (par rapport à la ROM1) qui va servir à déterminer le CS de chaque ROM ($11 - 10 = 1$).

6. Dessinez le schéma de câblage sur le [document réponse](#) (vous détaillerez le nombre de fils pour chaque bus et vous numéroterez les mémoires ROM1).
7. Quelles sont les mémoires ROM1 actives lors de l'accès en lecture à l'adresse $4B5_{16}$?

C'est le bit **A10** qui sélectionne les mémoires. Il faut donc déterminer sa valeur :

$4B5_{16} = 100\ 1011\ 0101_2 \rightarrow \mathbf{A10} = 1 \rightarrow \text{Les ROM1 numéros 3 et 4 sont actives (cf. schéma).}$

Exercice 5 (3 points)

On dispose d'une mémoire morte (ROM) possédant 12 fils d'adresse, d'une mémoire vive (RAM) possédant 10 fils d'adresse et de deux périphériques (P1 et P2) possédant respectivement 8 et 5 fils d'adresse. On désire les rendre accessibles à un microprocesseur via les bus d'adresse (16 fils), de donnée (8 fils) et de commande (dont le signal *Address Strobe*). Les mémoires et les périphériques possèdent un bus de donnée de 8 fils. La ROM sera située dans les adresses les plus faibles, viendront ensuite la RAM, P1 et P2. Pour tout l'exercice, c'est le mode zone qui sera utilisé avec le moins de zones possible.

1. Donnez les bits d'adresse qui serviront au décodage.

Pour 4 composants, les deux bits de poids fort suffisent : **A15** et **A14**.

Ces deux bits permettent de diviser l'espace mémoire en quatre zones. Chaque zone est associée à un composant.

Zone	A15	A14	Composant
0	0	0	ROM
1	0	1	RAM
2	1	0	P1
3	1	1	P2

← La ROM doit être située dans les adresses les plus faibles.

2. Donnez la fonction de décodage (équations du CS de chaque composant).

$$\mathbf{CS_{ROM}} = \mathbf{AS} \cdot \overline{\mathbf{A15}} \cdot \overline{\mathbf{A14}}$$

$$\mathbf{CS_{RAM}} = \mathbf{AS} \cdot \overline{\mathbf{A15}} \cdot \mathbf{A14}$$

$$\mathbf{CS_{P1}} = \mathbf{AS} \cdot \mathbf{A15} \cdot \overline{\mathbf{A14}}$$

$$\mathbf{CS_{P2}} = \mathbf{AS} \cdot \mathbf{A15} \cdot \mathbf{A14}$$

Avec **AS** = *Address Strobe* (validation de l'adresse par le microprocesseur).

3. Donnez la représentation de l'espace mémoire avec toutes les adresses remarquables.

ROM basse : $0000\ 0000\ 0000\ 0000_2 = 0000_{16}$

ROM haute : $0000\ 1111\ 1111\ 1111_2 = 0FFF_{16}$

RAM basse : $0100\ 0000\ 0000\ 0000_2 = 4000_{16}$

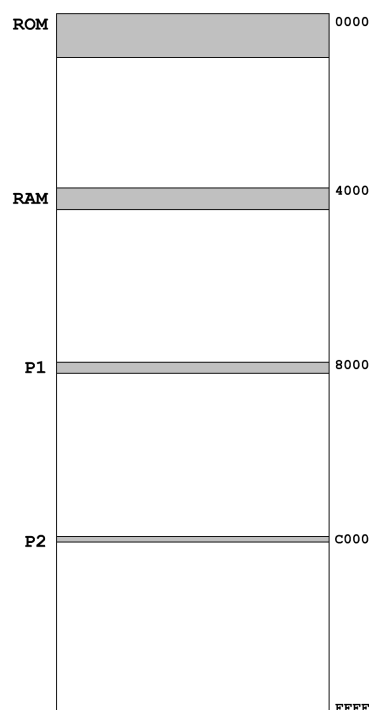
RAM haute : $0100\ 0011\ 1111\ 1111_2 = 43FF_{16}$

P1 basse : $1000\ 0000\ 0000\ 0000_2 = 8000_{16}$

P1 haute : $1000\ 0000\ 1111\ 1111_2 = 80FF_{16}$

P2 basse : $1100\ 0000\ 0000\ 0000_2 = C000_{16}$

P2 haute : $1100\ 0000\ 0001\ 1111_2 = C01F_{16}$



4. Est-il possible de réaliser un décodage de type linéaire ?

Le décodage de type linéaire associe un composant à un fil d'adresse du microprocesseur. Le plus grand composant connecté au microprocesseur est la ROM avec douze fils d'adresse. Sur les seize fils d'adresse du microprocesseur, quatre sont donc disponibles pour le décodage. Cela permet de connecter au moins quatre composants au microprocesseur. Cela est suffisant pour connecter les deux mémoires et les deux périphériques. Par conséquent, **il est possible de réaliser un décodage de type linéaire.**

Exercice 6 (3 points)

Architecture externe d'une mémoire :

1. Donner, en précisant leurs rôles, les quatre signaux principaux contenus dans le bus de commande (ou de contrôle). Lesquels sont des entrées ? Lesquels sont des sorties ?

Le bus de commande contient les signaux suivants :

- **R/W** : Lecture/Écriture. C'est une **entrée**.
- **AS** (*Address Strobe*) : Validation du bus d'adresse. C'est une **entrée**.
- **CS** (*Chip Select*) ou **CE** (*Chip Enable*) : Activation de la mémoire utilisée. C'est une **entrée**.
- **ACK** (*Acknowledge*) : Validation du bus de donnée. C'est une **sortie**.

Le microprocesseur 68000 :

2. Le 68000 est un microprocesseur 16 bits. Qu'est-ce que cela signifie ?

Cela signifie que **son bus de donnée est sur 16 bits**.

3. Quelle est la particularité des mémoires utilisées par le 68000 ?

Elles doivent être **adressables par octet**.

Presetable synchronous 4-bit binary up/down counter

74HC/HCT193

FEATURES

- Synchronous reversible 4-bit binary counting
- Asynchronous parallel load
- Asynchronous reset
- Expandable without external logic
- Output capability: standard
- I_{CC} category: MSI

GENERAL DESCRIPTION

The 74HC/HCT193 are high-speed Si-gate CMOS devices and are pin compatible with low power Schottky TTL (LSTTL). They are specified in compliance with JEDEC standard no. 7A.

The 74HC/HCT193 are 4-bit synchronous binary up/down counters. Separate up/down clocks, CP_U and CP_D respectively, simplify operation. The outputs change state synchronously with the LOW-to-HIGH transition of either clock input. If the CP_U clock is pulsed while CP_D is held HIGH, the device will count up. If the CP_D clock is pulsed while CP_U is held HIGH, the device will count down. Only one clock input can be held HIGH at any time, or erroneous operation will result. The device can be cleared at any time by the asynchronous master reset input (MR); it may also be loaded in parallel by activating the asynchronous parallel load input (\overline{PL}).

The "193" contains four master-slave JK flip-flops with the necessary steering logic to provide the asynchronous reset, load, and synchronous count up and count down functions.

Each flip-flop contains JK feedback from slave to master, such that a LOW-to-HIGH transition on the CP_D input will decrease the count by one, while a similar transition on the CP_U input will advance the count by one.

One clock should be held HIGH while counting with the other, otherwise the circuit will either count by two's or not at all, depending on the state of the first flip-flop, which cannot toggle as long as either clock input is LOW. Applications requiring reversible operation must make the reversing decision while the activating clock is HIGH to avoid erroneous counts.

The terminal count up ($\overline{TC_U}$) and terminal count down ($\overline{TC_D}$) outputs are normally HIGH. When the circuit has reached the maximum count state of 15, the next HIGH-to-LOW transition of CP_U will cause $\overline{TC_U}$ to go LOW.

$\overline{TC_U}$ will stay LOW until CP_U goes HIGH again, duplicating the count up clock.

Likewise, the $\overline{TC_D}$ output will go LOW when the circuit is in the zero state and the CP_D goes LOW. The terminal count outputs can be used as the clock input signals to the next higher order circuit in a multistage counter, since they duplicate the clock waveforms. Multistage counters will not be fully synchronous, since there is a slight delay time difference added for each stage that is added.

The counter may be preset by the asynchronous parallel load capability of the circuit. Information present on the parallel data inputs (D_0 to D_3) is loaded into the counter and appears on the outputs (Q_0 to Q_3) regardless of the conditions of the clock inputs when the parallel load (\overline{PL}) input is LOW. A HIGH level on the master reset (MR) input will disable the parallel load gates, override both clock inputs and set all outputs (Q_0 to Q_3) LOW. If one of the clock inputs is LOW during and after a reset or load operation, the next LOW-to-HIGH transition of that clock will be interpreted as a legitimate signal and will be counted.

Presettable synchronous 4-bit binary up/down counter

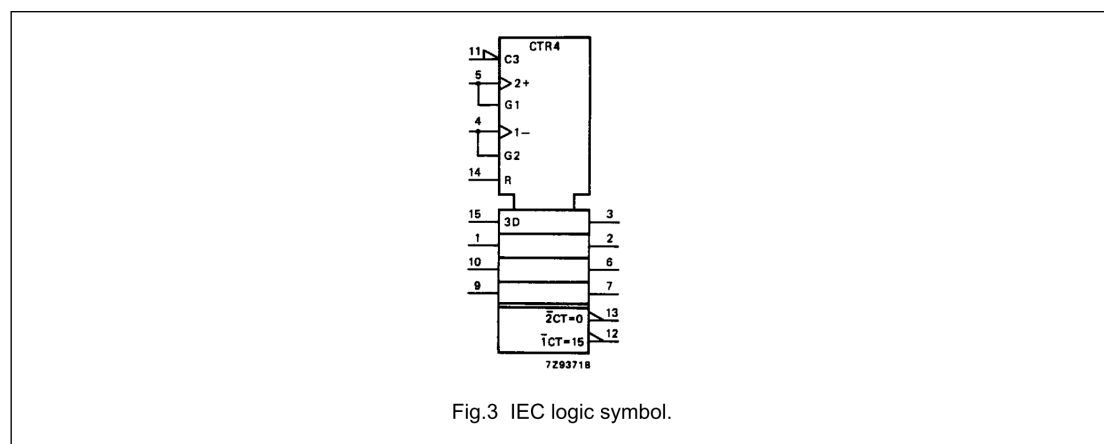
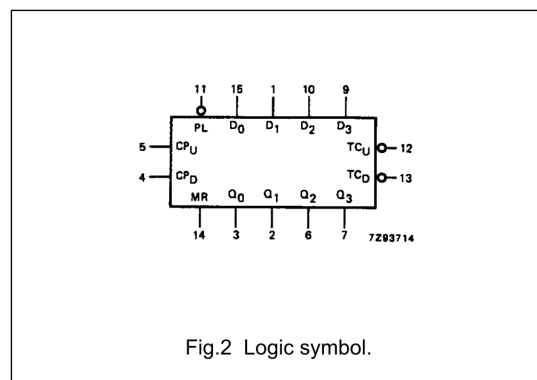
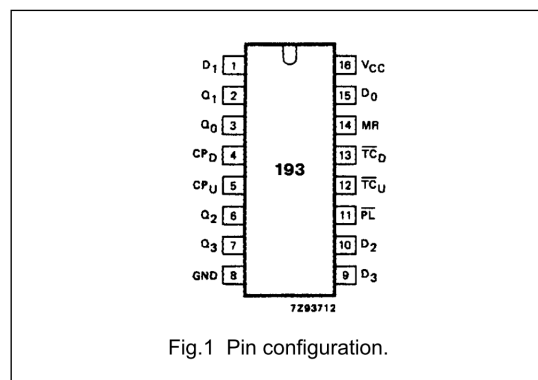
74HC/HCT193

PIN DESCRIPTION

PIN NO.	SYMBOL	NAME AND FUNCTION
3, 2, 6, 7	Q_0 to Q_3	flip-flop outputs
4	CP_D	count down clock input ⁽¹⁾
5	CP_U	count up clock input ⁽¹⁾
8	GND	ground (0 V)
11	\overline{PL}	asynchronous parallel load input (active LOW)
12	$\overline{TC_U}$	terminal count up (carry) output (active LOW)
13	$\overline{TC_D}$	terminal count down (borrow) output (active LOW)
14	MR	asynchronous master reset input (active HIGH)
15, 1, 10, 9	D_0 to D_3	data inputs
16	V_{CC}	positive supply voltage

Note

1. LOW-to-HIGH, edge triggered



Presettable synchronous 4-bit binary up/down counter

74HC/HCT193

FUNCTION TABLE

OPERATING MODE	INPUTS								OUTPUTS					
	MR	$\overline{\text{PL}}$	CP_U	CP_D	D_0	D_1	D_2	D_3	Q_0	Q_1	Q_2	Q_3	$\overline{\text{TC}}_\text{U}$	$\overline{\text{TC}}_\text{D}$
reset (clear)	H	X	X	L	X	X	X	X	L	L	L	L	H	L
	H	X	X	H	X	X	X	X	L	L	L	L	H	H
parallel load	L	L	X	L	L	L	L	L	L	L	L	L	H	L
	L	L	X	H	L	L	L	L	L	L	L	L	H	H
	L	L	L	X	H	H	H	H	H	H	H	H	L	H
	L	L	H	X	H	H	H	H	H	H	H	H	H	H
count up	L	H	\uparrow	H	X	X	X	X	count up				$\text{H}^{(2)}$	H
count down	L	H	H	\uparrow	X	X	X	X	count down				H	$\text{H}^{(3)}$

Notes

- H = HIGH voltage level
L = LOW voltage level
X = don't care
 \uparrow = LOW-to-HIGH clock transition
- $\overline{\text{TC}}_\text{U}$ = CP_U at terminal count up (HHHH)
- $\overline{\text{TC}}_\text{D}$ = CP_D at terminal count down (LLLL)

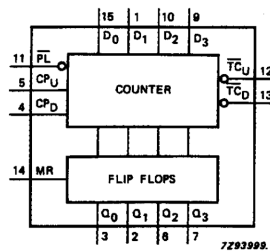
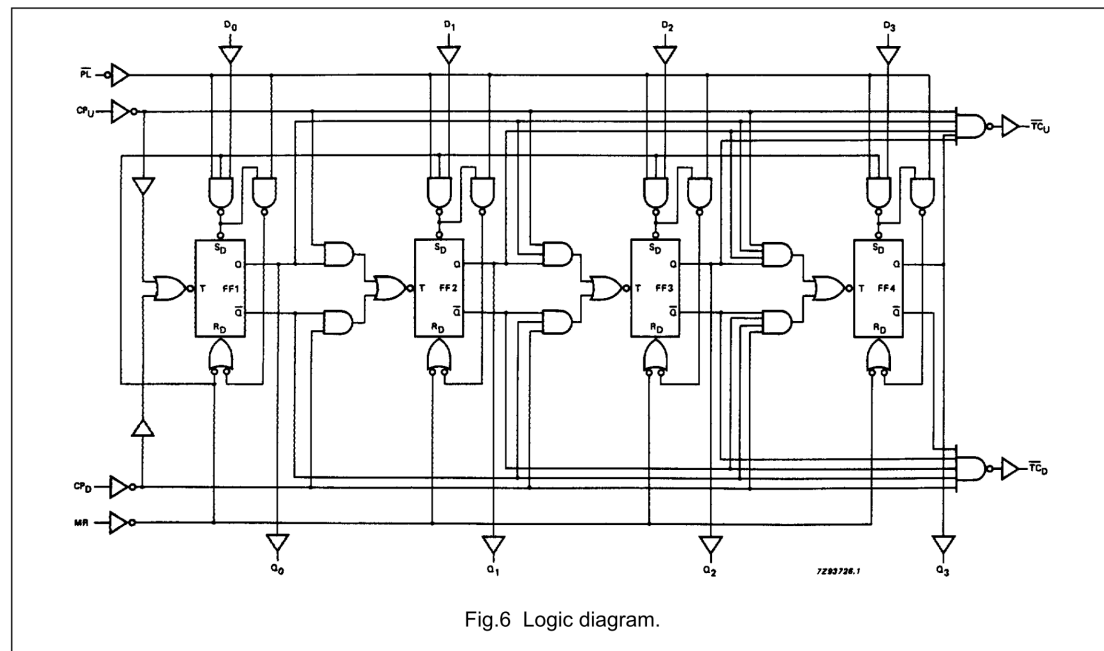
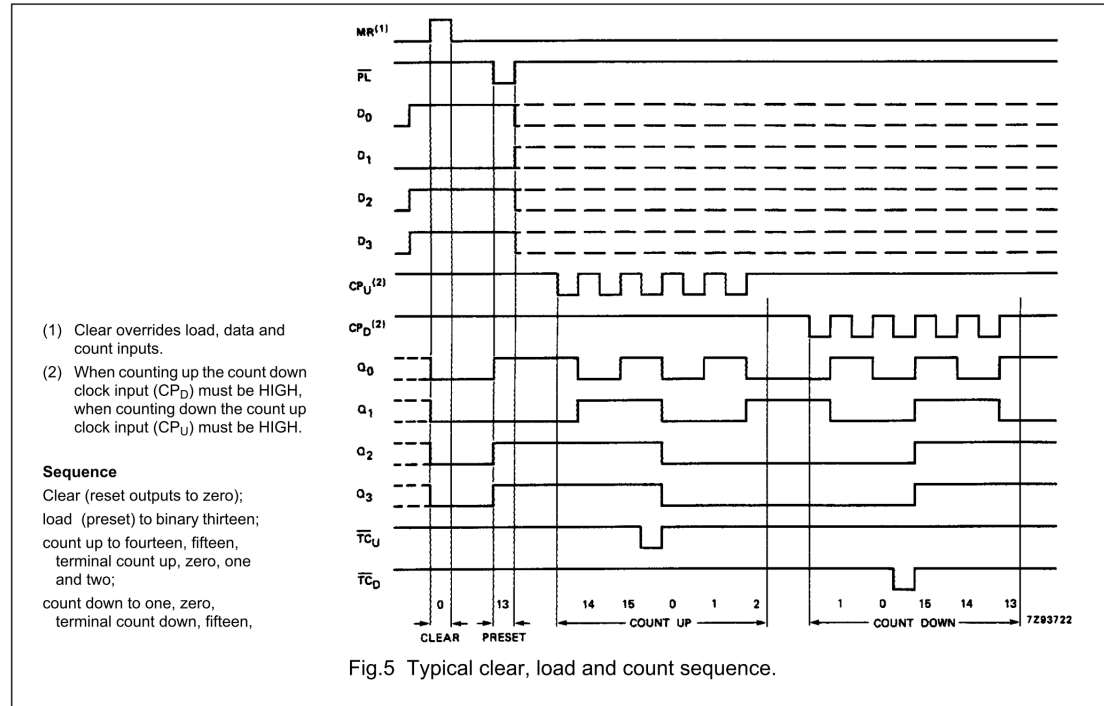


Fig.4 Functional diagram.

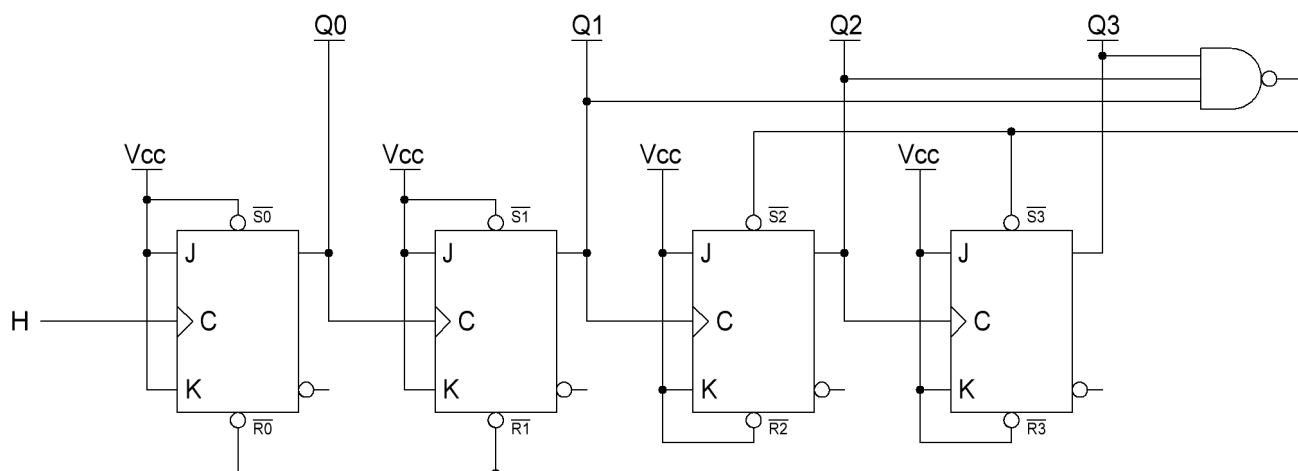
Presettable synchronous 4-bit binary up/down counter

74HC/HCT193



DOCUMENT RÉPONSE À RENDRE AVEC LA COPIE

Exercice 3



Exercice 4

