

TD 2 Architecture des ordinateurs

Exemple

Entrée : E

Sortie : S

Etat : Q

Circuit 1

E	Q_{i-1}	$D (=E \text{ xor } Q_{i-1})$	Q_i	S
0	0	0	0	0
0	1	1	1	1
1	0	1	1	1
1	1	0	0	0

→ Bascule T implémentée avec une bascule D !

Circuit 2

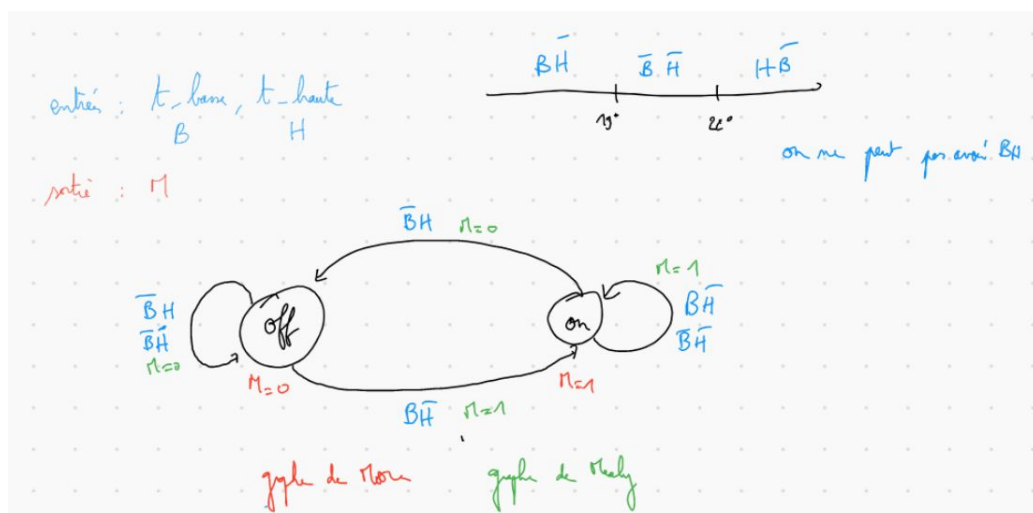
E	Q_{i-1}	$T (= E \text{ xor } Q_{i-1})$	Q_i	S
0	0	0	0	0
0	1	1	0	0
1	0	1	1	1
1	1	0	1	1

→ Bascule D implémentée avec une bascule T !

Exercice 1

B) Table de transition:

État départ	B	H	État arrivé
off	0	0	off
	0	1	off
	1	0	on
on	0	0	on
	0	1	off
	1	0	on



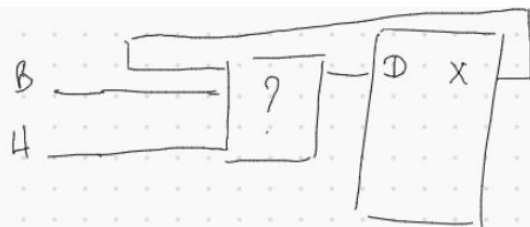
c) Ici, le graphe contient 2 états : une seule bascule suffit donc à décrire les états

	\bar{X} bascule				assignation: état
off	0				$M = X !$
on	1				
		table transition			
X_{i-1}	B	H	D	T	X_i
0	0	0	0	0	0
	0	1	0	0	0
	1	0	1	1	1
1	0	0	1	0	1
	0	1	0	1	0
	1	0	1	0	1

module chaudiere (B, H, ^{reset}rst, ^{clock}clk : M)

$X := D$ on clk, reset when rst
 $D = \bar{X} \wedge B \wedge H + X \wedge \bar{H}$
 $M = X$
 end module

affectation
 séquentielle
 "au top d'horloge"



$$D = \bar{X}B\bar{H} + X\bar{B}\bar{H} + X\bar{B}H$$

$$= \bar{X}B\bar{H} + X\bar{H}(\bar{B} + B)$$

$$T = \bar{X}B\bar{H} + X\bar{B}H$$

$$X := \bar{X} \wedge T + X \wedge \bar{T} \text{ on clk, reset when rst}$$

Exercise 2

```
module count4(rst, clk, en : s[3..0])
  t[0] = en
  t[1] = s[0]*en
  t[2] = s[0]*s[1]*en
  t[3] = s[0]*s[1]*s[2]*en

  s[3..0] := s[3..0]*t[3..0]+/s[3..0]*t[3..0] on clk, reset when rst
end module
```

Exercise 3

```
module count4Z(rst, clk, en, sclr : s[3..0])

  s[3..0] := /t[3..0]*s[3..0] + t[3..0]/s[3..0] on clk reset when rst

  t[0] = en*/sclr + s[0]*en*sclr
  t[1] = s[0]*en*/sclr + s[1]*en*sclr
  t[2] = s[0]*s[1]*en*/sclr + s[2]*en*sclr
  t[3] = s[0]*s[1]*s[2]*en*/sclr + s[3]*en*sclr

end module
```