

# IA et Verilog

## TP M2 2023

---

06/04

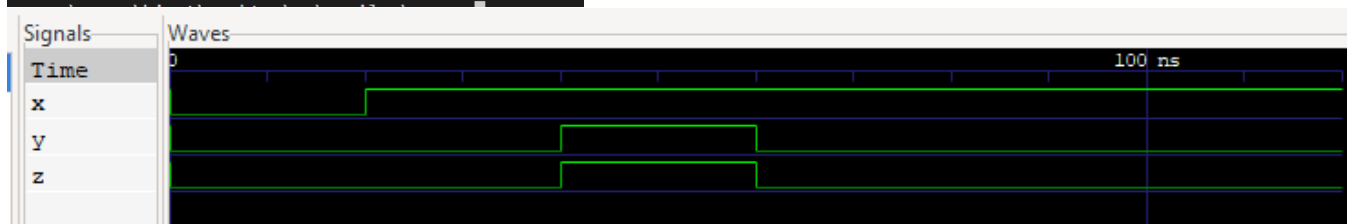
---

Antoine Binet  
Pour Efrei Paris



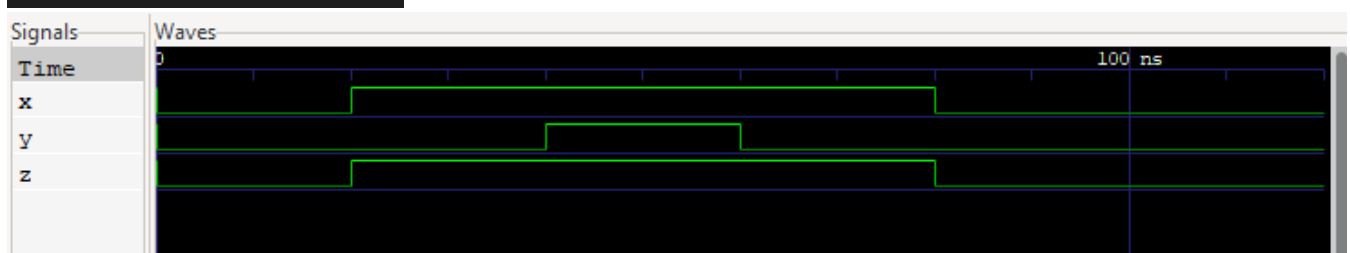
## Exercice 1

```
PS C:\Users\binet\Desktop\IA\verilog\Done> iverilog -  
PS C:\Users\binet\Desktop\IA\verilog\Done> vvp test2  
VCD info: dumpfile test.vcd opened for output.  
t= 0 x=0,y=0,z=0  
  
t= 20 x=1,y=0,z=0  
  
t= 40 x=1,y=1,z=1  
  
t= 60 x=1,y=0,z=0
```



## Exercice 2 - Porte logique Ou

```
VCD info: dumpfile tes  
t= 0 x=0,y=0,z=0  
  
t= 20 x=1,y=0,z=1  
  
t= 40 x=1,y=1,z=1  
  
t= 60 x=1,y=0,z=1  
  
t= 80 x=0,y=0,z=0
```



## Exercice Reg vs wire

*En faisant des recherches si nécessaire, expliquer la différence entre reg versus wire. Dans quels cas est-il plus pertinent d'utiliser l'un que l'autre ?*

En Verilog, reg représente une variable qui peut être écrite ou lue à partir d'une seule source, tandis que wire représente une connexion logique qui ne peut être lue que.

### Exercice - assign vs always

En faisant des recherches si nécessaire, expliquer la différence entre assign vs XXX. Dans quels cas est-il plus pertinent d'utiliser l'un que l'autre ?

En Verilog, assign est utilisé pour assigner une valeur constante ou calculée à un signal combinatoire, tandis que XXX (par exemple, always, initial, generate) est utilisé pour décrire le comportement d'un signal synchrone ou asynchrone.

### Exercice - Porte logique ET à trois entrées

```
VCD info: dumpfile test.vcd
t= 0 x=0,y=0,z=0,s=0

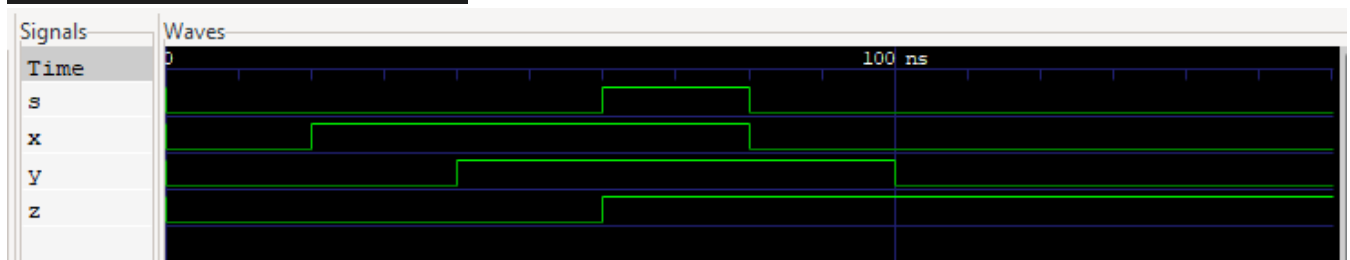
t= 20 x=1,y=0,z=0,s=0

t= 40 x=1,y=1,z=0,s=0

t= 60 x=1,y=1,z=1,s=1

t= 80 x=0,y=1,z=1,s=0

t=100 x=0,y=0,z=1,s=0
```



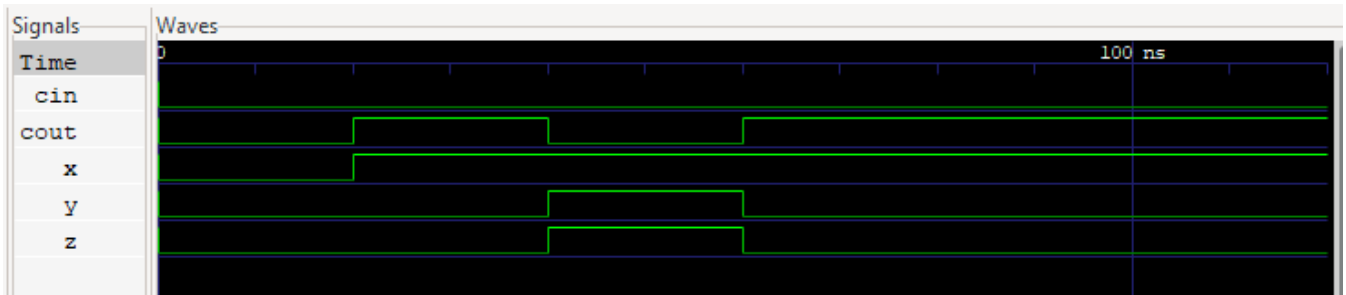
### Exercice - Additionneur 1 bit

```
VCD info: dumpfile test.vcd opened for output.
t= 0 cin=0,cout=0,x=0,y=0,z=0

t= 20 cin=0,cout=1,x=1,y=0,z=0

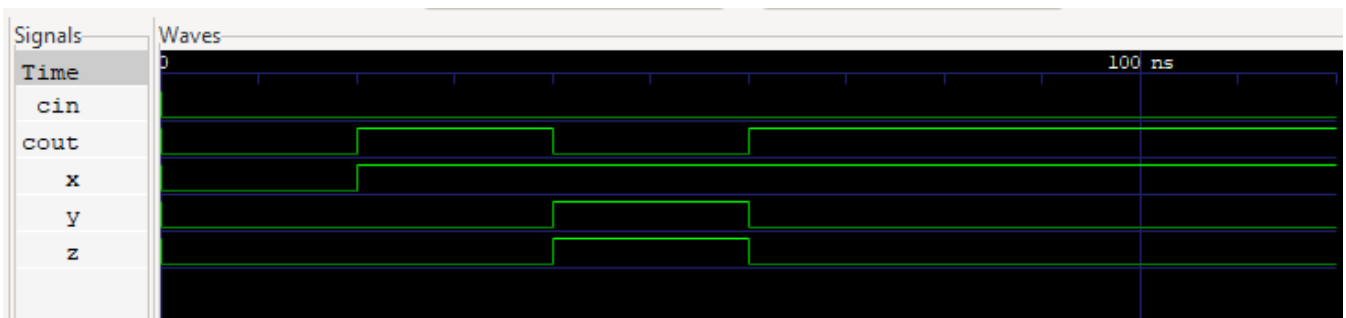
t= 40 cin=0,cout=0,x=1,y=1,z=1

t= 60 cin=0,cout=1,x=1,y=0,z=0
```



## Additionneur 8bit

```
PS C:\Users\binet\Desktop\IA\verilog\Done> vvp test2
VCD info: dumpfile test_adder_8bit.vcd opened for output.
a = 0, b = 0, result = 0
a = 5, b = 6, result = 11
a = 7, b = 8, result = 15
a = 170, b = 1, result = 171
a = 255, b = 255, result = 254
test_add_8bit.v:44: $finish called at 50000 (1ps)
```



## Multiplieur 8 bit

```
VCD info: dumpfile test.vcd opened for output.
a= 5, b= 10, result= 50
a= 3, b= 8, result= 24
a= 15, b= 7, result= 105
a= 0, b=255, result= 0
a=100, b= 0, result= 0
test_mult_8bit.v:57: $finish called at 500000 (1ps)
```

