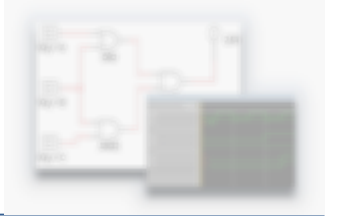


# ECE119 – Ψηφιακή Σχεδίαση

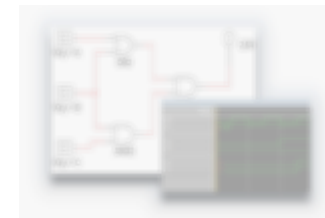
Διδάσκοντες Εργαστηρίου: Δ. Καραμπερόπουλος  
Δ. Γαρυφάλλου

## ➤ Lab 4: Icarus Verilog & GTKWave

# Icarus Verilog & GTKWave



# Παράδειγμα Verilog



## ➤ Μεταγλώττιση (**compile**) κώδικα Verilog

```
module main;  
  initial  
  begin  
    $display("Hello, world");  
  $finish ;  
  end  
endmodule
```

```
>iverilog -o hello hello.v
```

Εκτελέσιμο  
αρχείο που θα  
δημιουργηθεί

Αρχείο κώδικα Verilog  
το οποίο θέλουμε να  
μεταγλωττίσουμε

```
Command Prompt  
C:\Users\fevgas>cd c:\ECE119\lab02  
c:\ECE119\lab02>dir  
Volume in drive C has no label.  
Volume Serial Number is 5E1A-73E8  
  
Directory of c:\ECE119\lab02  
  
10/26/2021  01:47 PM  <DIR>      .  
10/26/2021  01:47 PM  <DIR>      ..  
10/26/2021  01:47 PM                107 hello.v  
                1 File(s)          107 bytes  
                2 Dir(s)  360,924,274,688 bytes free  
  
c:\ECE119\lab02>iverilog -o hello hello.v  
  
c:\ECE119\lab02>dir  
Volume in drive C has no label.  
Volume Serial Number is 5E1A-73E8  
  
Directory of c:\ECE119\lab02  
  
10/26/2021  01:55 PM  <DIR>      .  
10/26/2021  01:55 PM  <DIR>      ..  
10/26/2021  01:55 PM                775 hello  
10/26/2021  01:47 PM                107 hello.v  
                2 File(s)           882 bytes  
                2 Dir(s)  360,924,200,960 bytes free  
  
c:\ECE119\lab02>
```

## Παράδειγμα Verilog

- Η εκτέλεση του μεταγλωττισμένου κώδικα Verilog γίνεται με την εντολή **vvp** (run time engine)

```
>vvp hello
```

Εκτελέσιμο αρχείο που δημιουργήθηκε στην μεταγλώττιση

```
Command Prompt
Microsoft Windows [Version 10.0.19042.1288]
(c) Microsoft Corporation. All rights reserved.

C:\Users\fevgas>cd c:\ECE119\lab02

c:\ECE119\lab02>dir
Volume in drive C has no label.
Volume Serial Number is 5E1A-73E8

Directory of c:\ECE119\lab02

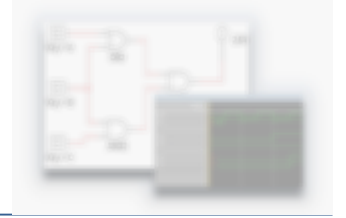
10/26/2021  03:38 PM    <DIR>          .
10/26/2021  03:38 PM    <DIR>          ..
10/26/2021  03:37 PM                90 hello.v
                1 File(s)      90 bytes
                2 Dir(s)  360,628,363,264 bytes free

c:\ECE119\lab02>iverilog -o hello hello.v

c:\ECE119\lab02>vvp hello
Hello, World

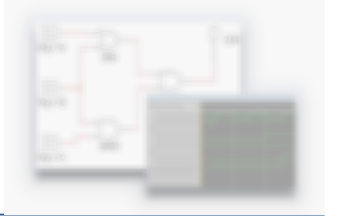
c:\ECE119\lab02>_
```

## Προσομοίωση κυκλώματος



- Αφού περιγράψουμε ένα κύκλωμα σε κάποια γλώσσα HDL (Verilog), θα πρέπει να **προσομοιώσουμε** την λειτουργία του.
- Για να θέσουμε σε λειτουργία το κύκλωμα που έχουμε περιγράψει χρησιμοποιούμε προγράμματα γραμμένα και αυτά σε HDL (Verilog), που ονομάζονται **μονάδες δοκιμής (test benches)**.

# Παράδειγμα testbench



## ➤ Παράδειγμα **test bench** στον half-adder

```
module half_adder (output S, C, input x, y);  
    xor (S, x, y);  
    and (C, x, y);  
endmodule
```

```
`timescale 1ns/10ps
```

```
module t_half_adder;
```

```
    wire t_S, t_C;  
    reg t_x, t_y;  
    half_adder HA (t_S,t_C,t_x,t_y);  
    initial  
        begin  
            t_x = 1'b0; t_y = 1'b0;  
            #100 t_x = 1'b0; t_y = 1'b1;  
            #100 t_x = 1'b1; t_y = 1'b0;  
            #100 t_x = 1'b1; t_y = 1'b1;
```

```
        end  
    initial  
        begin  
            $monitor("time = ", $time, " x = %b Y = %b S=%b C=%b",t_x,t_y,t_S,t_C);  
            #400 $finish;
```

```
        end  
    endmodule
```

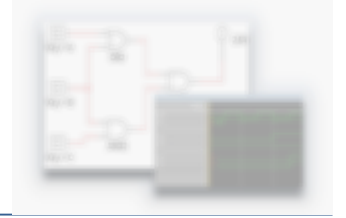
C:\Windows\system32\cmd.exe

```
c:\myVerilog>iverilog -o half_adder half_adder.v
```

```
c:\myVerilog>vvp half_adder
```

```
time = 0 X = 0 Y = 0 S=0 C=0  
time = 100 X = 0 Y = 1 S=1 C=0  
time = 200 X = 1 Y = 0 S=1 C=0  
time = 300 X = 1 Y = 1 S=0 C=1
```

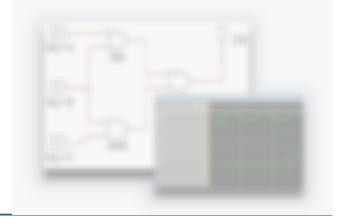
```
c:\myVerilog>
```



## Προσομοίωση κυκλώματος - Κυματομορφή

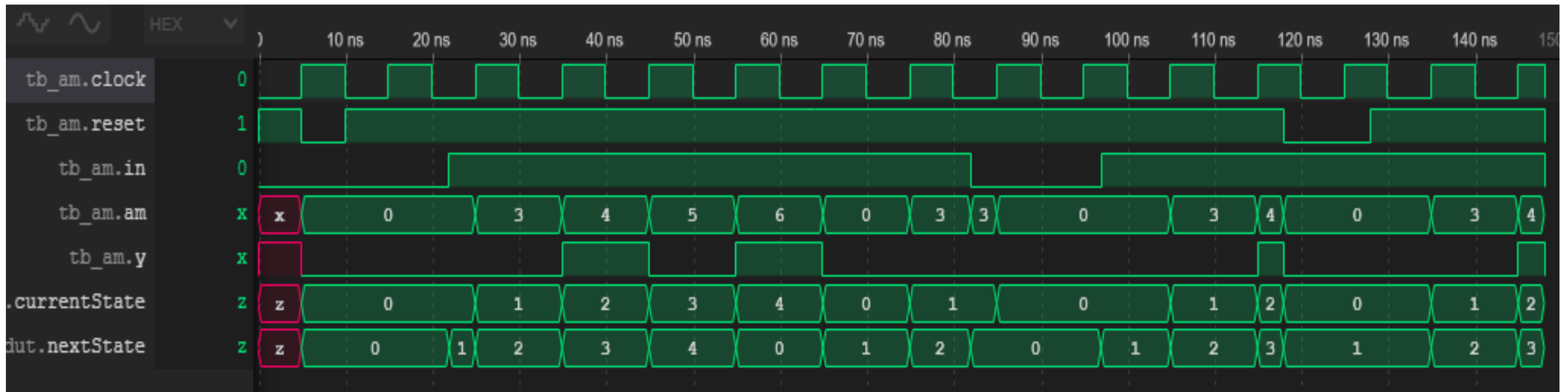
- Γραφική μέθοδος αναπαράστασης των τιμών των σημάτων κατά την διάρκεια μιας προσομοίωσης.





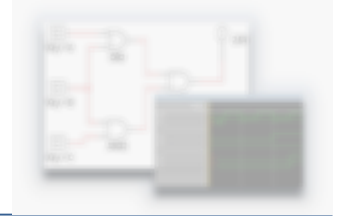
## Προσομοίωση κυκλώματος - Κυματομορφή

- Γραφική μέθοδος αναπαράστασης των τιμών των σημάτων κατά την διάρκεια μιας προσομοίωσης.



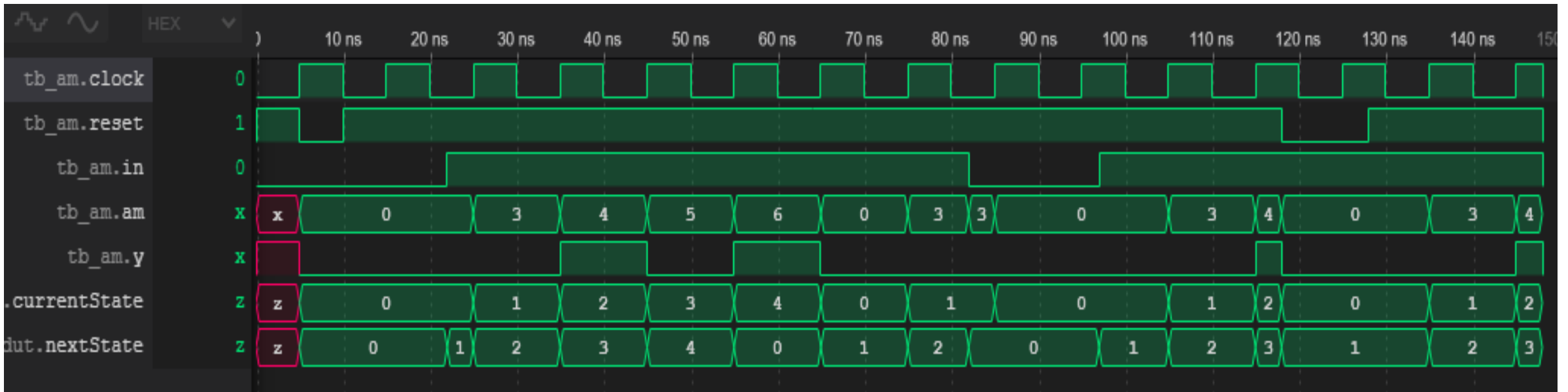


# Προσομοίωση κυκλώματος - Κυματομορφή



- Γραφική μέθοδος αναπαράστασης των τιμών των σημάτων κατά την διάρκεια μιας προσομοίωσης.

Θα το υλοποιήσετε στο ... Lab10!!!



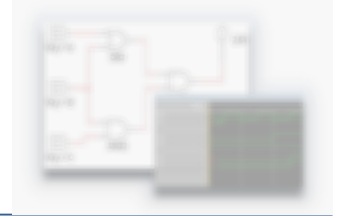
## Το περιβάλλον εκτέλεσης - vnr



- Παράγει και τα αρχεία που είναι απαραίτητα για να δούμε με την μορφή **κυματομορφών (waveforms)** τις τιμές που λαμβάνει κάθε καλώδιο κατά την διάρκεια της προσομοίωσης.
- Οι κυματομορφές αποθηκεύονται σε αρχεία **VCD (Value Change Dump)**
  - ASCII αρχεία καταγραφής (dumpfiles).
  - Κατάληξη .vcd
  - Περιέχουν:
    - ορισμούς μεταβλητών
    - τις αλλαγές που συμβαίνουν στις τιμές των μεταβλητών κατά την διάρκεια της προσομοίωσης.

```
$date Tue Dec 01
$send
$version Icarus Verilog
$send
$timescale 10ps
$send
$scope module t_half_adder $send
$var wire 1 ! t_S $send
$var wire 1 " t_C $send
$var reg 1 # t_x $send
$var reg 1 $ t_y $send
$scope module HA $send
$var wire 1 " C $send
$var wire 1 ! S $send
$var wire 1 # x $send
$var wire 1 $ y $send
$upscope $send
$upscope $send
$senddefinitions $send
#0
$dumpvars
0$
0#
0"
0!
$send
#10000
1!
1$
#20000
0$
1#
#30000
0!
1
```

Αρχείο vcd



## \$dumpfile & \$dumpvars

Όνομα αρχείου καταγραφής που θα δημιουργηθεί

➤ `$dumpfile ("dumpfile.vcd");`

➤ `$dumpvars (<levels> , <module> );`

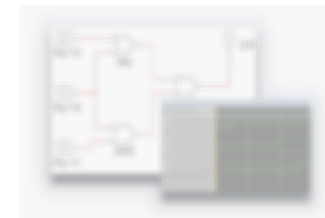
Όνομα module του οποίου θέλουμε να καταγράψουμε τις μεταβλητές του

- **Level 0** καταγράφονται οι μεταβλητές του **<module>** και όλων των στιγμιοτύπων (instances) υπομονάδων (modules) που το **<module>** δημιουργεί.
- **Level 1** καταγράφονται οι μεταβλητές του **<module>**

### Παράδειγμα:

- `$dumpfile ("lab03.vcd");`
- `$dumpvars(0, t_POS_3_3_Gates);`

# Παράδειγμα testbench - \$dumpfile και \$dumpvars



```
module POS_3_3_Gates (out, A, B, C);
    output out;
    input A, B, C;
    wire w1, w2, w3, w4, w5;
    ...
endmodule
```

```
module t_POS_3_3_Gates;
    reg A, B, C;
    wire out;
```

```
    POS_3_3_Gates dut(out, A, B, C);
    initial begin
```

```
        $dumpfile("lab03.vcd");
        $dumpvars(0,t_POS_3_3_Gates);
```

```
        A = 1'b0; B = 1'b0; C = 1'b0;
        #10 A = 1'b0; B = 1'b0; C = 1'b1;
        #10 A = 1'b0; B = 1'b1; C = 1'b0;
        #10 A = 1'b0; B = 1'b1; C = 1'b1;
        #10 A = 1'b1; B = 1'b0; C = 1'b0;
        #10 A = 1'b1; B = 1'b0; C = 1'b1;
        #10 A = 1'b1; B = 1'b1; C = 1'b0;
        #10 A = 1'b1; B = 1'b1; C = 1'b1;
```

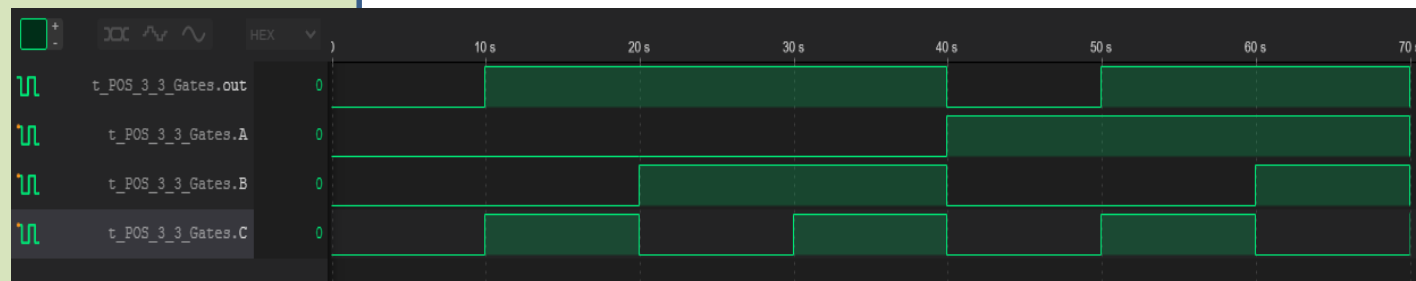
```
    end
    initial $monitor ("input= %b %b %b out= %b", A, B, C, out);
endmodule
```

Μέσα σε  
initial block

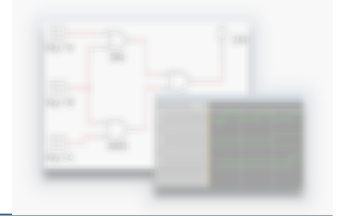
output

```
input= 0 0 0 Out= 0
input= 0 0 1 Out= 1
input= 0 1 0 Out= 1
input= 0 1 1 Out= 1
input= 1 0 0 Out= 0
input= 1 0 1 Out= 1
input= 1 1 0 Out= 1
input= 1 1 1 Out= 1
```

lab03.vcd



# Παράδειγμα testbench - \$dumpfile και \$dumpvars



```
module half_adder (output S, C, input x, y);  
    xor (S, x, y);  
    and (C, x, y);  
endmodule
```

```
`timescale 1ns/10ps
```

```
module t_half_adder;  
    wire t_S, t_C;  
    reg t_x, t_y;  
    half_adder HA (t_S,t_C,t_x,t_y);  
    initial  
    begin
```

```
        $dumpfile("half_adder.vcd");  
        $dumpvars(0,t_half_adder);
```

```
        t_x = 1'b0; t_y = 1'b0;  
        #100 t_x = 1'b0; t_y = 1'b1;  
        #100 t_x = 1'b1; t_y = 1'b0;  
        #100 t_x = 1'b1; t_y = 1'b1;
```

```
    end  
    initial  
    begin  
        $monitor("time = ", $time, " x = %b Y = %b S=%b C=%b",t_x,t_y,t_S,t_C);  
        #400 $finish;
```

```
    end  
endmodule
```

Γραμμή εντολών

```
C:\myVerilog>vvp half_adder
```

```
VCD info: dumpfile half_adder.vcd opened for output.
```

```
time =          0 X = 0 Y = 0 S=0 C=0  
time =         100 X = 0 Y = 1 S=1 C=0  
time =         200 X = 1 Y = 0 S=1 C=0  
time =         300 X = 1 Y = 1 S=0 C=1
```

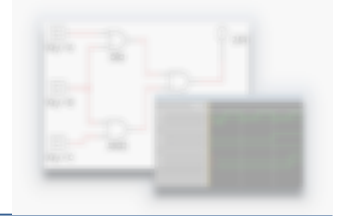
```
C:\myVerilog>dir half*
```

```
Volume in drive C has no label.  
Volume Serial Number is 3C90-3BF0
```

```
Directory of C:\myVerilog
```

```
01/12/2020  10:15  μμ           2.691 half_adder  
01/12/2020  10:13  μμ             614 half_adder.v  
01/12/2020  10:21  μμ           477 half_adder.vcd  
            3 File(s)           3.782 bytes  
            0 Dir(s)  35.694.866.432 bytes free
```

# GTKWave - Λογισμικό για την απεικόνιση κυματομορφών

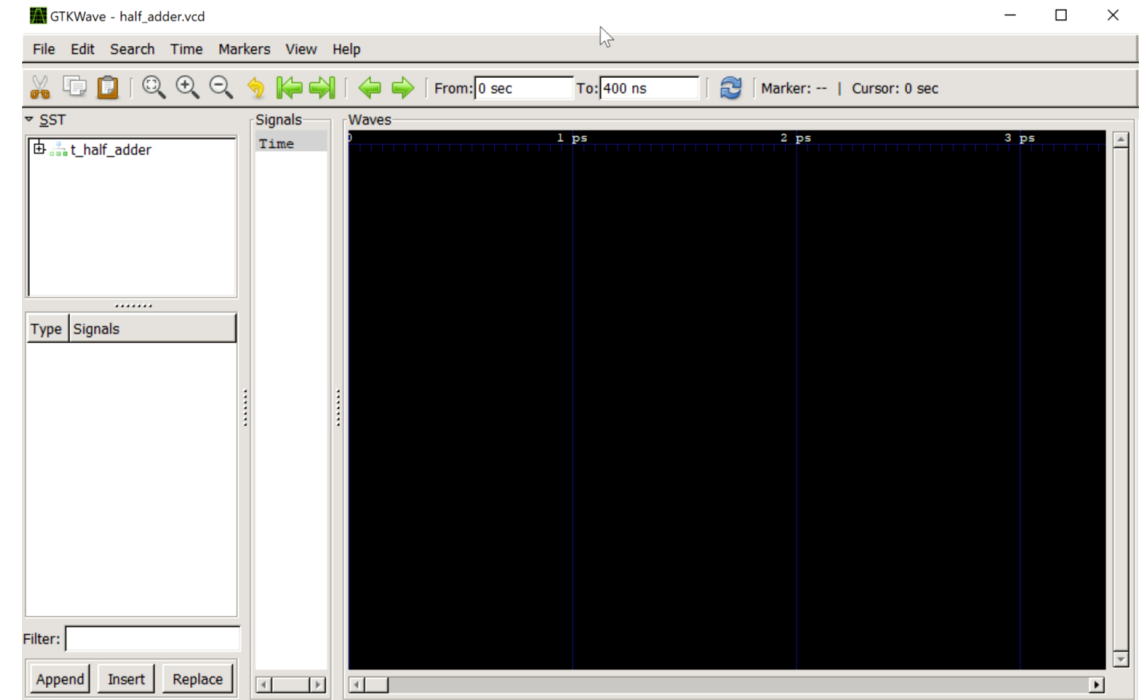


- **GTKWave** λογισμικό ανοικτού κώδικα για την απεικόνιση κυματομορφών.
- Εγκαθίσταται μαζί με τον Icarus.

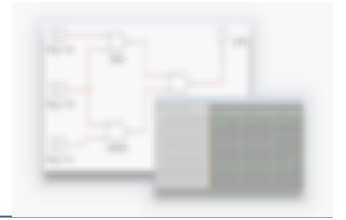
```
>gtkwave half_adder.vcd
```

Γραμμή εντολών

```
C:\myVerilog>gtkwave half_adder.vcd
```



# GTKWave - Λογισμικό για την απεικόνιση κυματομορφών



1.

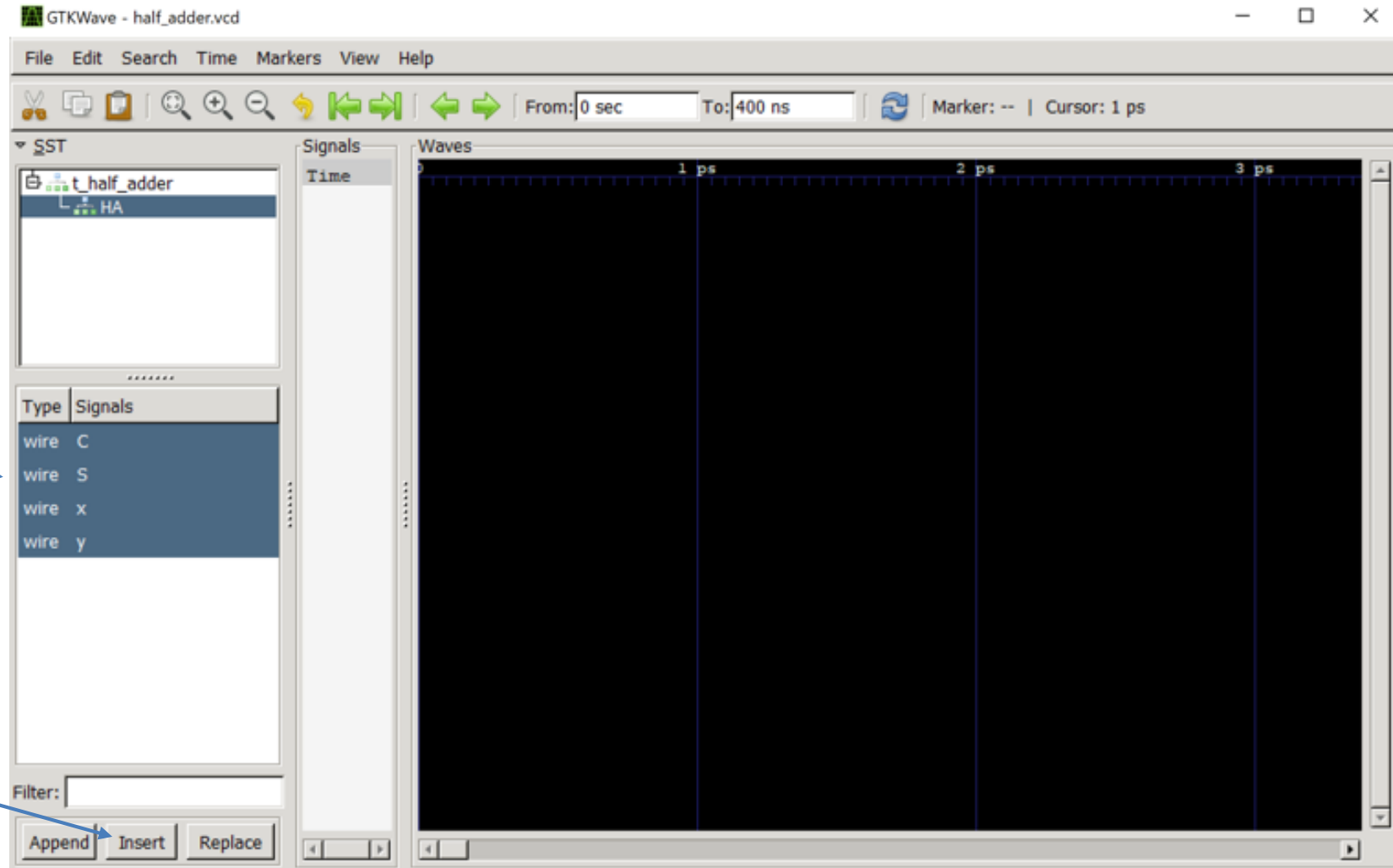
Ανοίγουμε την  
ιεραρχία

2.

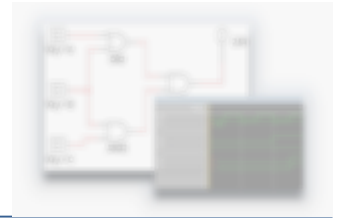
Επιλέγουμε  
σήματα

3.

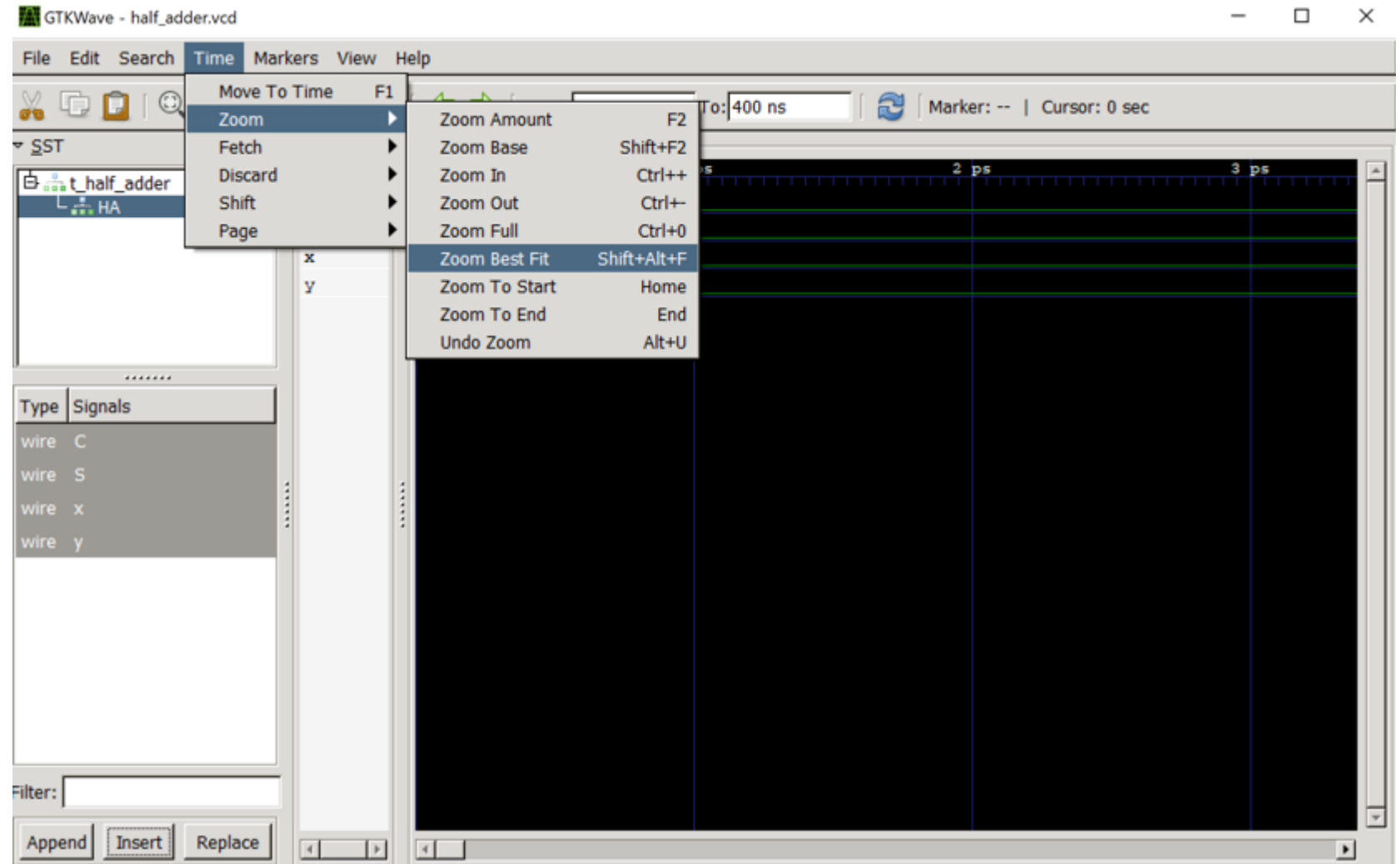
Πατάμε Insert



# GTKWave - Λογισμικό για την απεικόνιση κυματομορφών

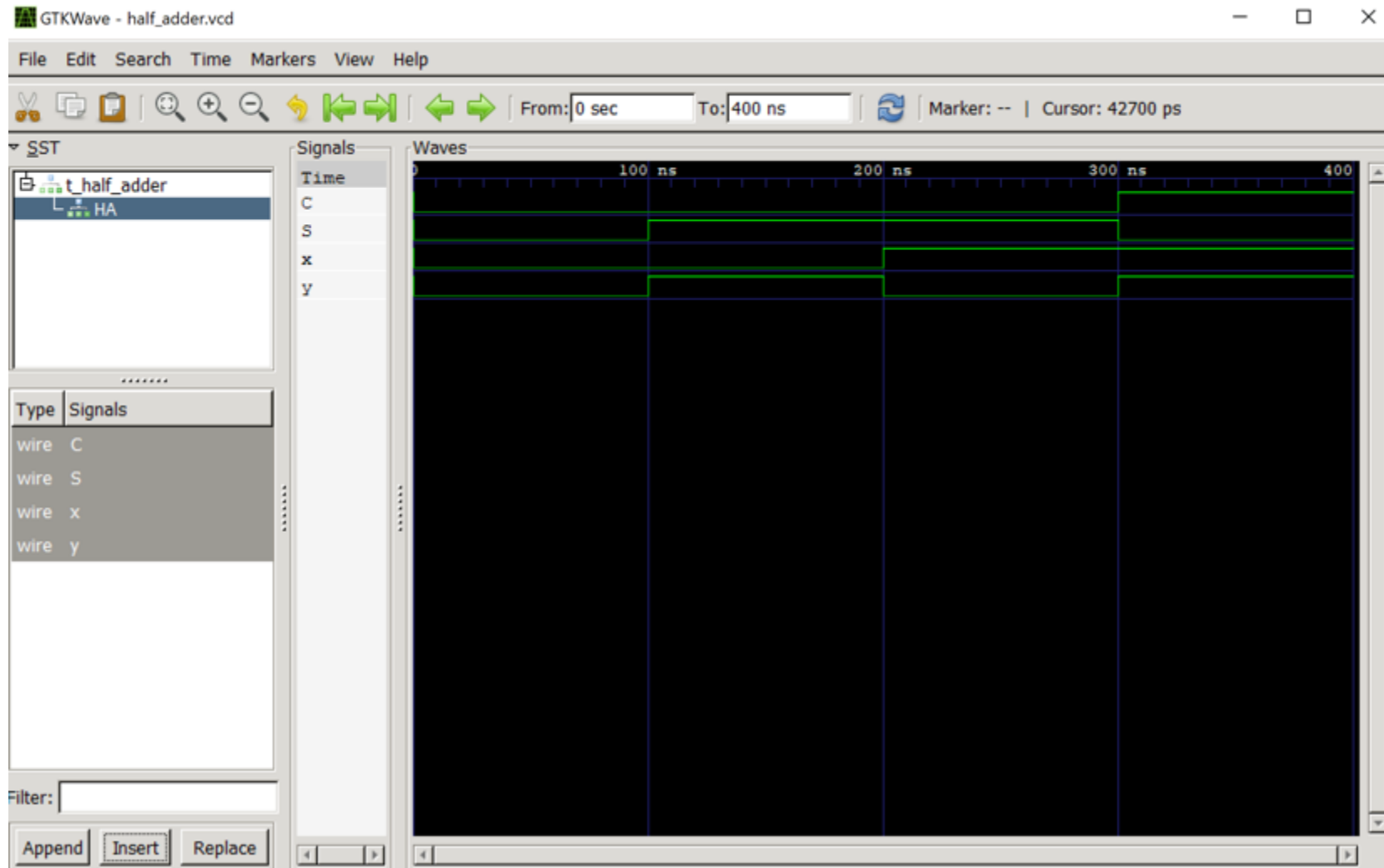
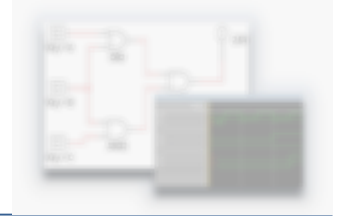


- Για να δούμε το σημείο της προσομοίωσης που μας ενδιαφέρει χρησιμοποιούμε την λειτουργία **Zoom**





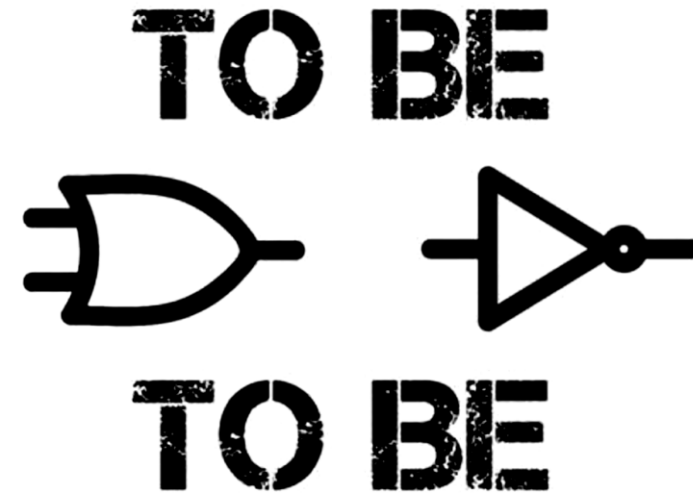
# GTKWave - Λογισμικό για την απεικόνιση κυματομορφών



# Ευχαριστώ για την προσοχή σας!



➤ Ερωτήσεις / Απορίες ;



Επικοινωνία: [ece119.uth@gmail.com](mailto:ece119.uth@gmail.com)