



ΥΠΟΥΡΓΕΙΟ ΕΘΝΙΚΗΣ ΠΑΙΔΕΙΑΣ ΚΑΙ ΘΡΗΣΚΕΥΜΑΤΩΝ  
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ ΕΠΕΑΕΚ



ΕΥΡΩΠΑΪΚΗ ΕΝΩΣΗ  
ΣΥΧΡΗΜΑΤΟΔΟΤΗΣΗ  
ΕΥΡΩΠΑΪΚΟ ΚΟΙΝΩΝΙΚΟ ΤΑΜΕΙΟ



Η ΠΑΙΔΕΙΑ ΣΤΗΝ ΚΟΡΥΦΗ  
Επιχειρησιακό Πρόγραμμα  
Εκπαίδευσης και Αρχικής  
Επαγγελματικής Κατάρτισης



Αθήνα 2004. Αθήνα 2008.

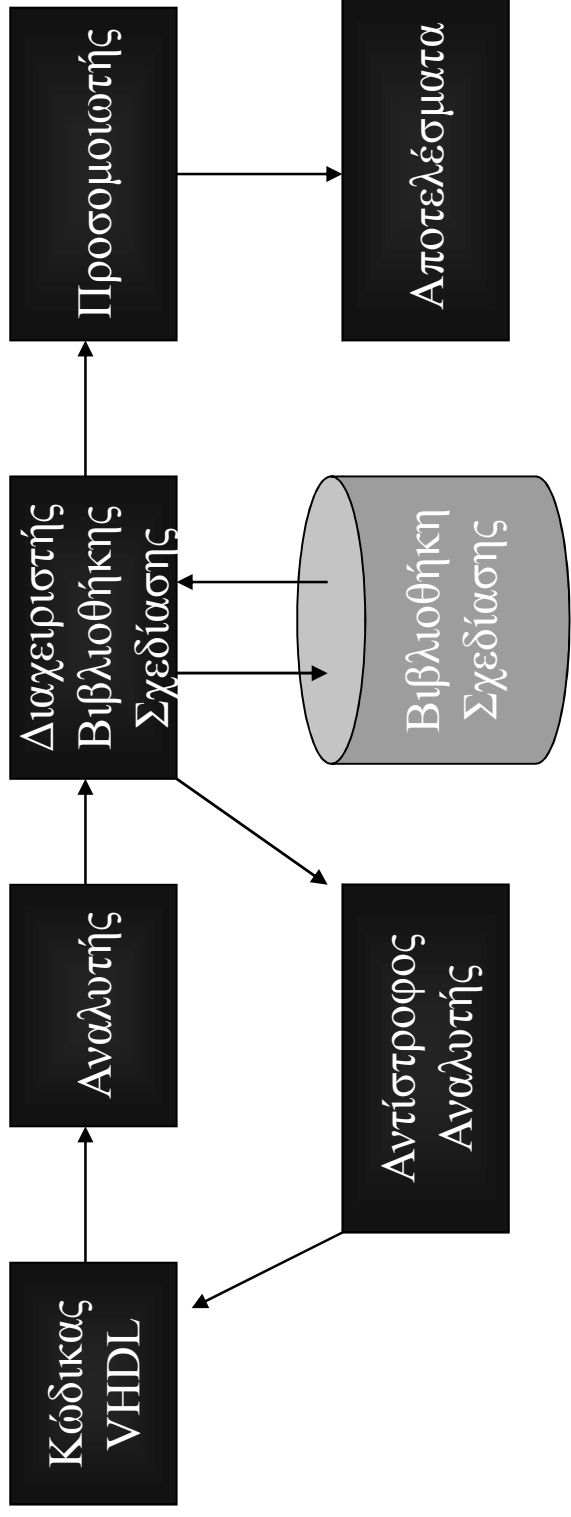
# Προσομοίωση Συστημάτων με VHDL

*(Peter Ashenden, The Students Guide to VHDL)*



# Περιβάλλον Σχεδίασης

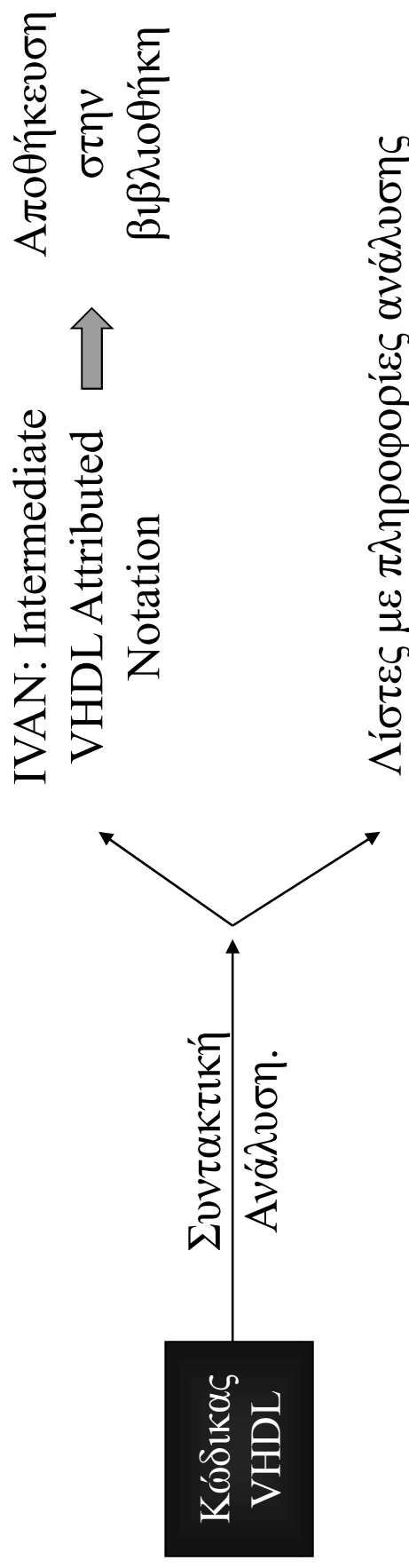
---



# Αναλυτής / Αντίστροφος Αναλυτής

---

Επεξεργάζεται τα VHDL αρχεία.



Ο αντίστροφος αναλυτής μετατρέπει οποιαδήποτε μονάδα της βιβλιοθήκης από την μορφή IVAN σε ισοδύναμη VHDL.

# Βιβλιοθήκη / Διαχειριστής Βιβλιοθήκης

---

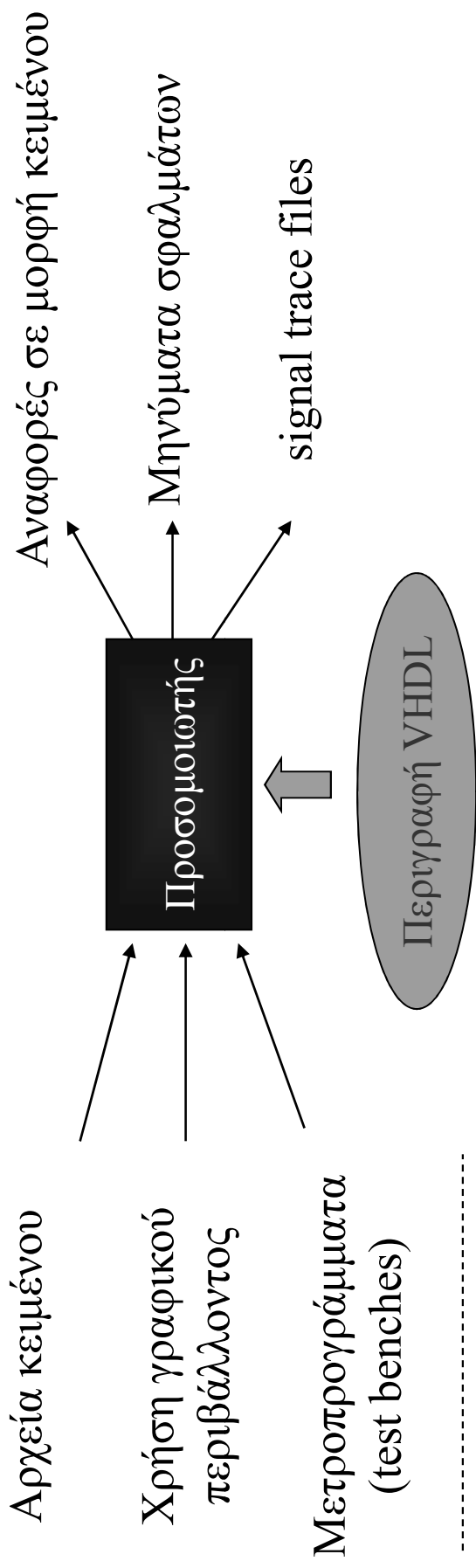
- Η βιβλιοθήκη σχεδίασης είναι η πηγή δεδομένων σχεδίασης
- Η προεπιλεγμένη βιβλιοθήκη είναι η "work"
- Οι μονάδες σχεδίασης της βιβλιοθήκης είναι ελεύθερες από σφάλματα.
- Κάθε βιβλιοθήκη μπορεί να χρησιμοποιηθεί από διάφορες σχεδιάσεις.
- Ο διαχειριστής χρησιμοποιείται από τον αναλυτή για την χρήση της βιβλιοθήκης.
- Παράγεται τελικά ένα σύνολο από διασυνδεδεμένα στοιχεία.



# Προσομοιωτής

---

Μοντελοποιεί την αντίδραση ενός συστήματος σε μία ή περισσότερες διεγέρσεις της εισόδου του.



Διεγέρσεις

---

# Προσομοίωση

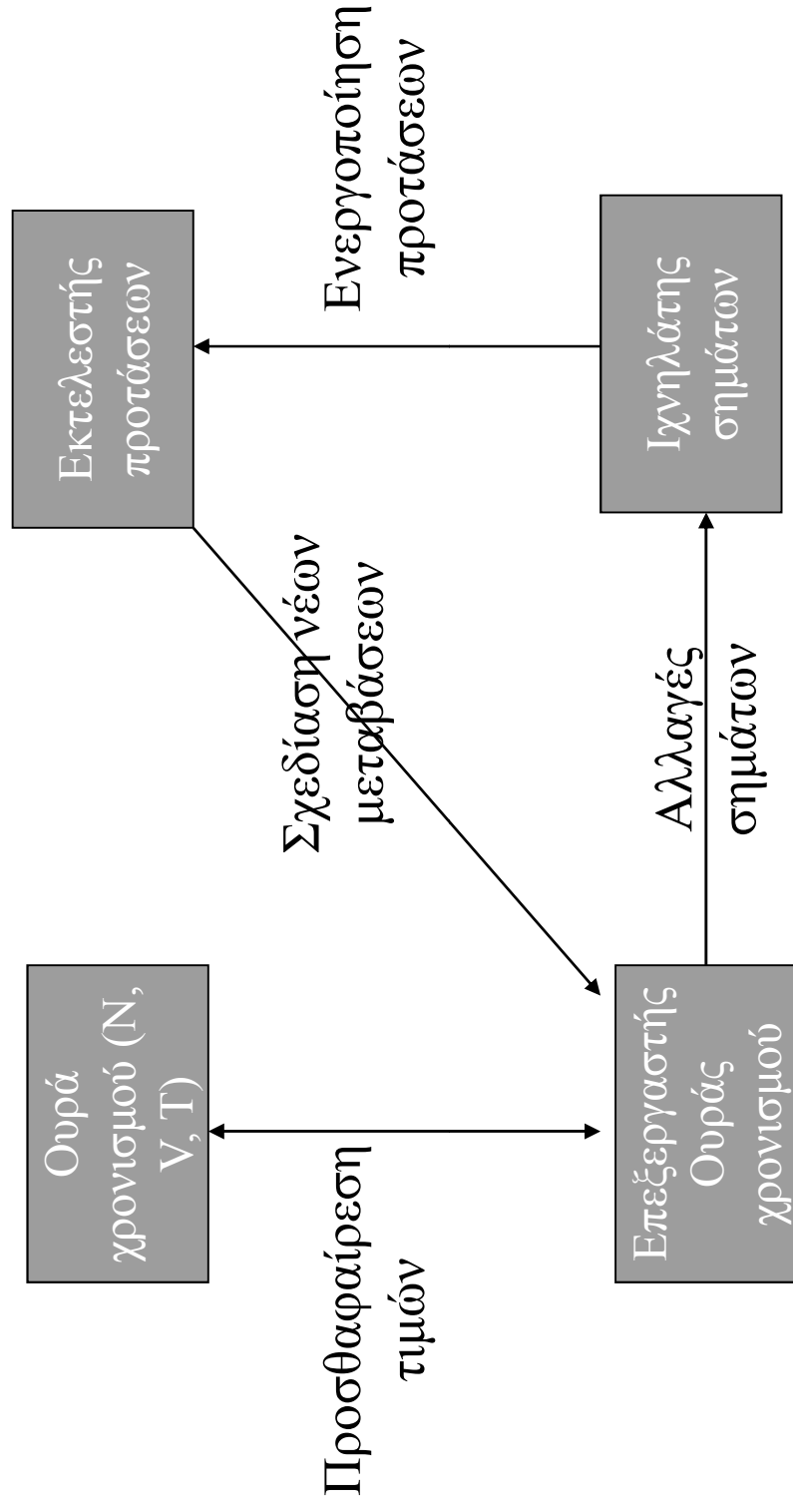
---

- Ο κύκλος εκτέλεσης αποτελείται από την φάση αρχικοποίησης ακολουθούμενη από έναν επαναλαμβανόμενο κύκλο εξομοίωσης.
- Η φάση αρχικοποίησης γίνεται στον χρόνο 0 και όλοι οι οδηγοί σημάτων παίρνουν αρχική τιμή.
- Στην συνέχεια κάθε process ή συντρέχουσα πρόταση ξεκινάει και εκτελεί τις ακολουθιακές εντολές της και συνήθως προγραμματίζει αναθέσεις τιμών σε σήματα.
- Όλες οι processes διακόπτουν την λειτουργία τους κάποια στιγμή και περιμένουν να ενεργοποιηθούν από κάποια σήματα που ανήκουν στην λίστα ευαισθησίας τους.
- Εκτελούνται οι πρώτες αναθέσεις σημάτων (νωρίτερα στον χρόνο) και ενεργοποιούνται οι processes που είναι ευαίσθητες.



# Πυρήνας Προσομοίωση

---



---

# Μετροπρόγραμμα - Test Bench

---

- Το Test Bench είναι ένας VHDL κώδικας ο οποίος τροφοδοτεί με εισόδους τον VHDL κώδικα του σχεδιασμού μας, και καταγράφει τις εξόδους πιθανώς συγκρίνοντας τις με τις αναμενόμενες.
- Το Test Bench έχει συνήθως τα ακόλουθα χαρακτηριστικά:
  1. Είναι μία ξεχωριστή οντότητα με αρχιτεκτονική.
  2. Δεν έχει θύρες εισόδου και σταθερές generic (self-contained).
  3. Δηλώνει σαν component τον σχεδιασμό και τον δεσμεύει.
  4. Δηλώνει όλα τα σήματα εισόδου/εξόδου και άλλα εσωτερικά.
  5. Χρησιμοποιεί οποιαδήποτε δομή για παραγωγή κυμματομορφών (δεν συντίθεται).
  6. Παράγει μηνύματα ελέγχου σε περιπτώσεις λαθών.



# Decoder Test Bench

---

```
library IEEE;
use IEEE.std_logic_1164.all;
use IEEE.std_logic_arith.all;
use IEEE.std_logic_unsigned.all;

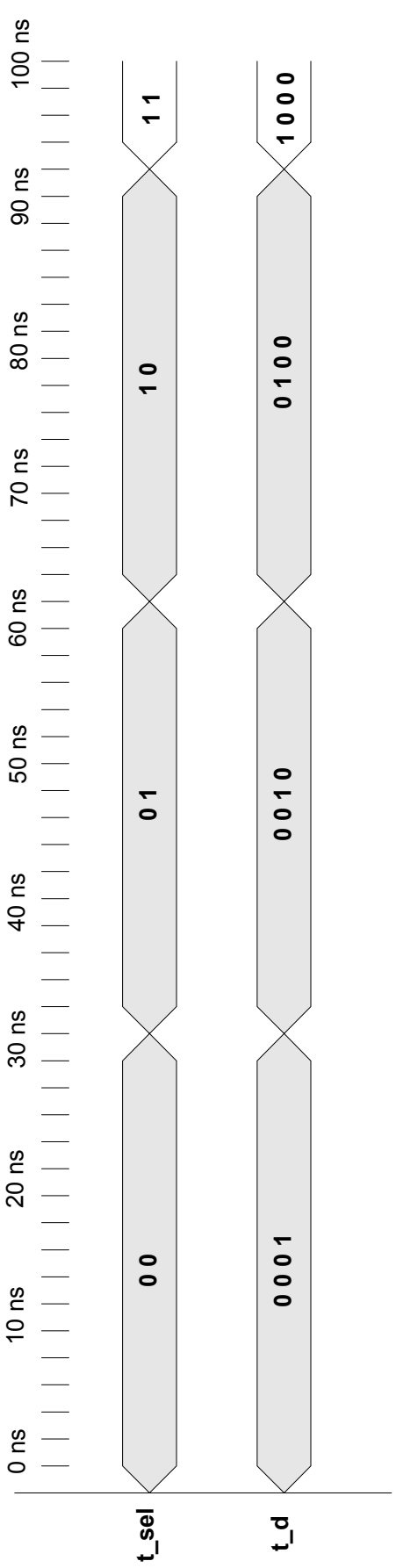
entity test_dec2_4 is
end test_dec2_4; } -----> Κενή δήλωση οντότητας

architecture test_bench of test_dec2_4 is
component dec2_4
port (sel: in std_logic_vector(1 downto 0);
      d : out std_logic_vector( 3 downto 0));
end component; } Δήλωση component
                 } αποκωδικοποιητή 2 σε 4

signal t_sel: std_logic_vector(1 downto 0);
signal t_d: std_logic_vector( 3 downto 0); } Δήλωση σημάτων
begin
  u0: dec2_4 port map(t_sel=>sel, t_d=>d);
  t_sel <= "00", "01" after 30 ns, "10" after 60 ns, "11" after 90 ns;
end test_bench;
```

# Αποτελέσματα προσομοίωσης

---



# Multiplexer Test Bench

---

```
library IEEE;
use IEEE.std_logic_1164.all;
use IEEE.std_logic_arith.all;
use IEEE.std_logic_unsigned.all;

entity TBSELECTOR is
end TBSELECTOR; } → Κενή δήλωση οντότητας

architecture BEH of TBSELECTOR is
component SELECTOR
port (A:in std_logic_vector(15 downto 0);
SEL : in std_logic_vector( 3 downto 0);
Y : out std_logic);
end component; } Δήλωση component

signal A: std_logic_vector(15 downto 0);
signal SEL: std_logic_vector( 3 downto 0);
signal exp_Y, Y : std_logic;
constant PERIOD : time := 50 ns;
constant STROBE : time := 45 ns; } Δήλωση σημάτων
```



# Selector Test Bench

---

```
begin
p0 : process
begin
for j in 0 to 15 loop
SEL <= conv_std_logic_vector(j,4);
exp_Y<='0';
A <= (A'range=>'1');
A( conv_integer(SEL) ) <= '0';
wait for PERIOD;
exp_Y<='1';
A <= (A'range=>'0');
A(conv_integer(SEL) <= '1';
wait for PERIOD;
end loop;
wait;
end process;
```

Η αναμενόμενη σωστή τιμή.

Αρχικοποίηση εισόδου: Όλες  
οι γραμμές σε μία τιμή και η  
επιλεγόμενη στην αντίστροφη

---

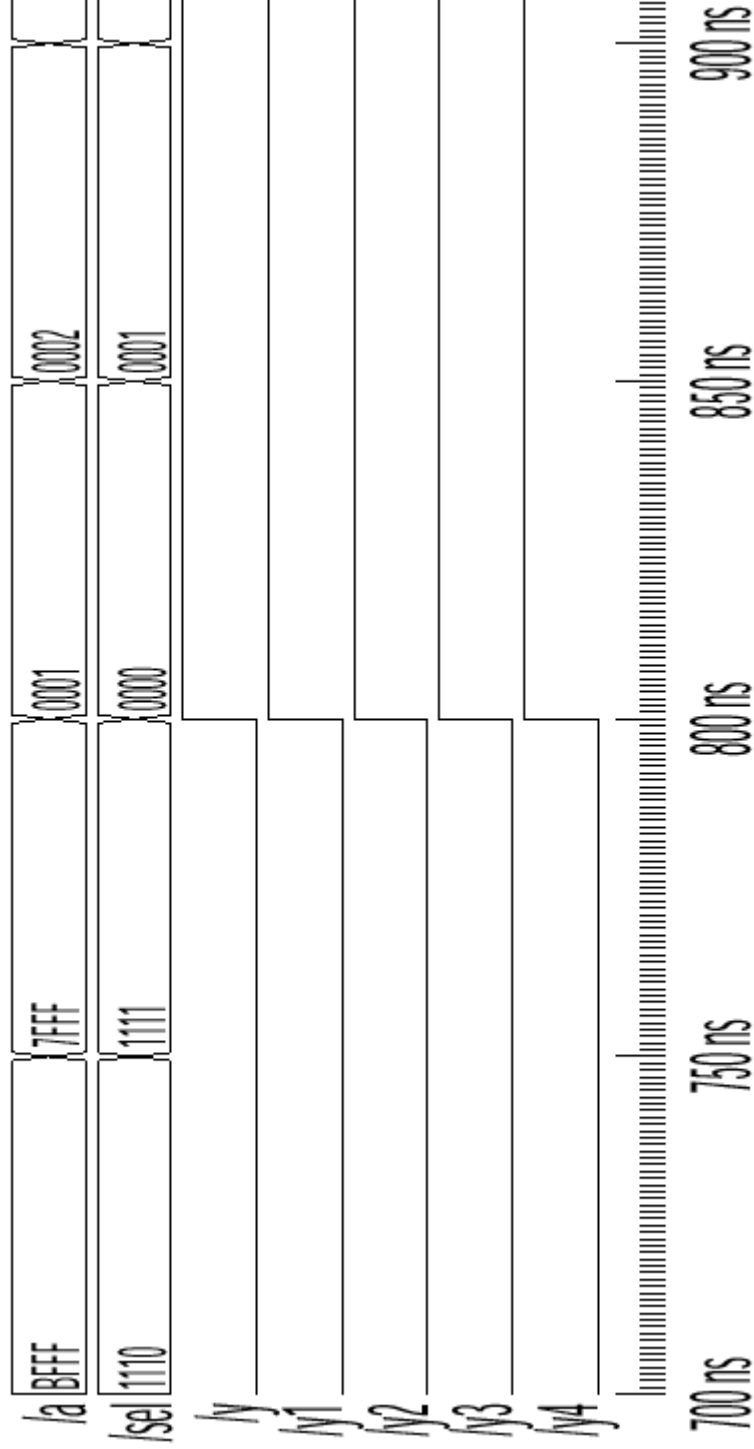
# Selector Test Bench

---

```
check : process
variable err_cnt : integer := 0;
begin
wait for STROBE;
for j in 0 to 31 loop
assert FALSE report "comparing ..." severity NOTE;
if (Y /= exp_Y) then
assert FALSE report "not compared" severity WARNING;
err_cnt := err_cnt + 1;
end if;
wait for PERIOD;
end loop;
assert (err_cnt = 0) report "test failed" severity ERROR;
assert (err_cnt /= 0) report "test passed" severity NOTE;
wait;
end process;
sel : SELECTOR port map ( A => A, SEL => SEL, Y => Y);
end BEH;
```

*Αυτόματος Έλεγχος  
Αποτελεσμάτων*

# Εξομοίωση



Ο έλεγχος ενός κυκλώματος μπορεί να γίνει και οπτικά με τις κυματομορφές αλλά αυτό είναι ιδιαίτερα περίπλοκο στα μεγάλα κυκλώματα