



Πανεπιστήμιο Κρήτης, Τμήμα Επιστήμης Υπολογιστών
HY220 Εργαστήριο Ψηφιακών Κυκλωμάτων
Χειμερινό Εξάμηνο 2014-2015

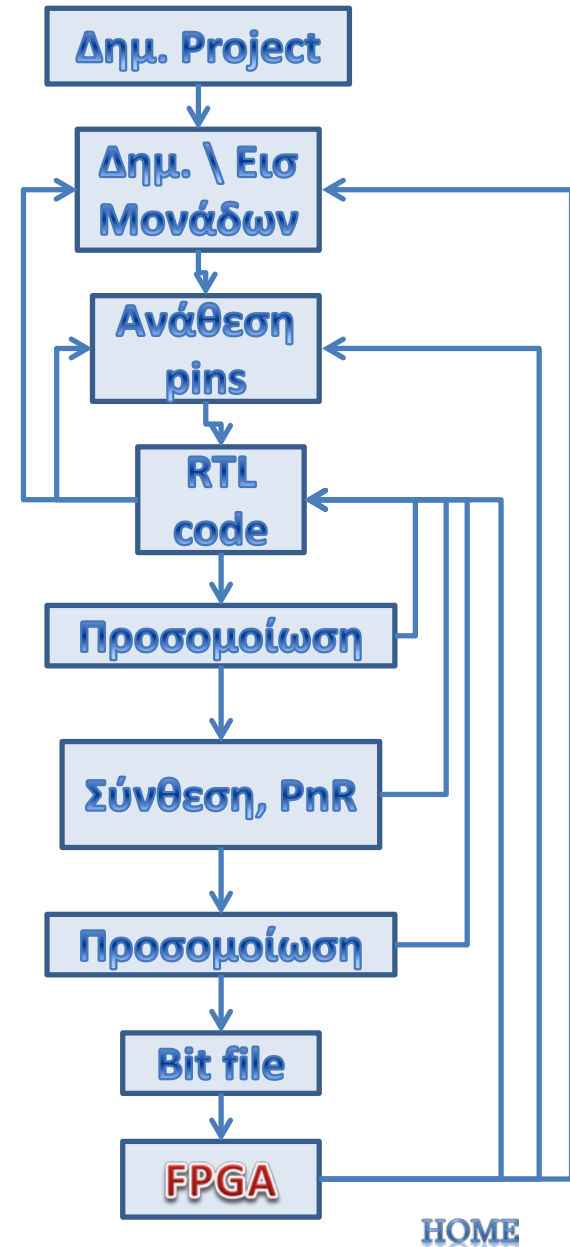
Εισαγωγή στη σχεδιαστική ροή της Xilinx

ISE Design Suite 14.7



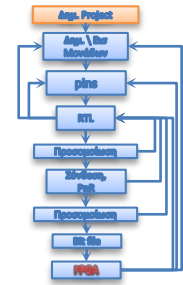
Στόχοι..

1. [Δημιουργία project](#)
2. [Δημ./Εισ. μονάδων](#)
3. [Μαρ σημάτων με FPGA pins](#)
4. [RTL](#)
5. [Post-RTL simulation](#)
6. [Synthesis, Place & Route \(PnR\)](#)
7. [Post-Synthesis, Post-PnR sim.](#)
8. [Δημιουργία bit file](#)

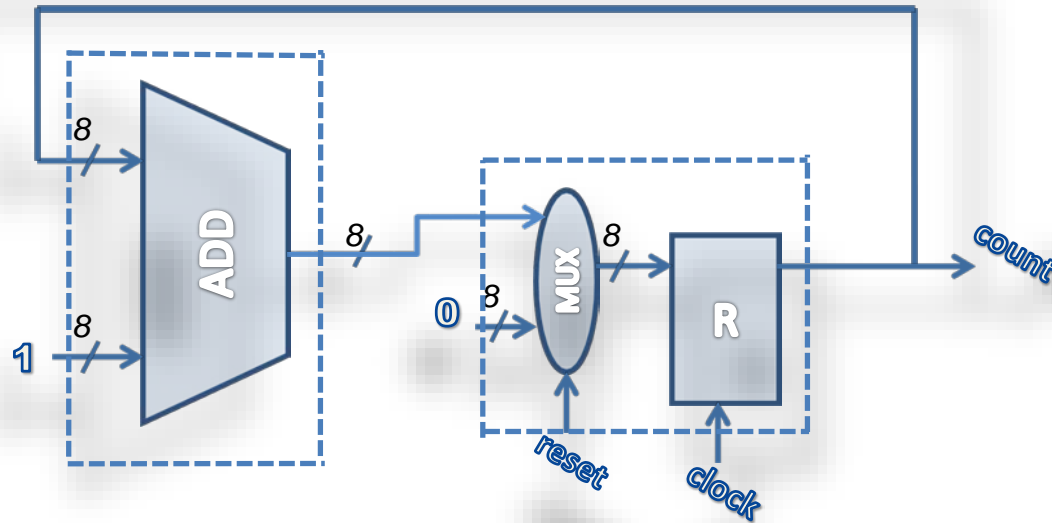




Δημιουργία project

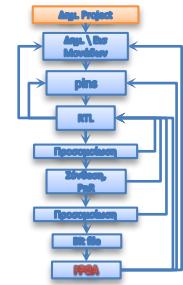


Π.χ. Ένας απλός 8 bit μετρητής με σύγχρονο reset:





Δημιουργία project



1. Ανοίγουμε το “ISE Design Suite 14.x” της Xilinx
2. Κλικ “New Project..” ή “File->New Project..”
3. Ορίζουμε όνομα project και path directory όπου θα αποθηκευτούν τα αρχεία του project
4. Μετά Next..

Welcome to the ISE® Design Suite

Project commands

Open Project... Project Browser...
New Project... Open Example...

Create project

Enter a name, locations, and comment for the project

Name: front0
Location: F:\LABS\fronthsthrio2012\front0
Working Directory: F:\LABS\fronthsthrio2012\front0
Description:

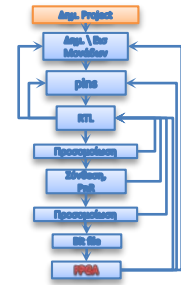
Select the type of top-level source for the project

Top-level source type:
HDL

More Info Next > Cancel



Δημιουργία project



A. Ορίζουμε:

1. Τύπο FPGA (οικογένεια, συσκευή και package)
2. Simulator, εδώ **ISim** για RTL και post-PnR προσομοίωση.
3. Πατάμε Next..

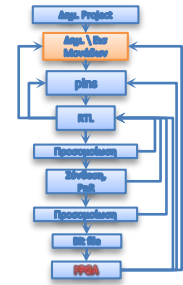
B. Στο επ. παράθυρο εμφανίζονται περιληπτικά οι επιλογές που κάναμε, αν είναι όλα σωστά κλικ “Finish”.

Property Name	Value
Top-Level Source Type	HDL
Evaluation Development Board	None Specified
Product Category	All
Family	Spartan3E
Device	XC3S100E
Package	CP132
Speed	-4
Synthesis Tool	XST (VHDL/Verilog)
Simulator	ISim (VHDL/Verilog)
Preferred Language	Verilog
Property Specification in Project File	Store non-default values only

OK Cancel Help

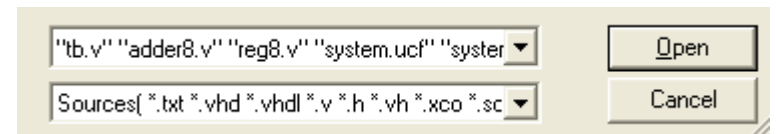
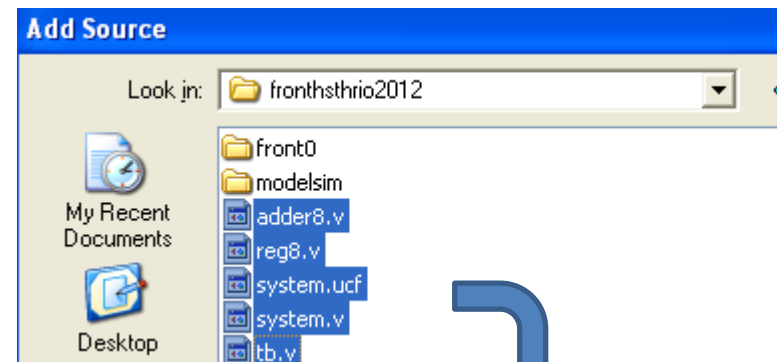
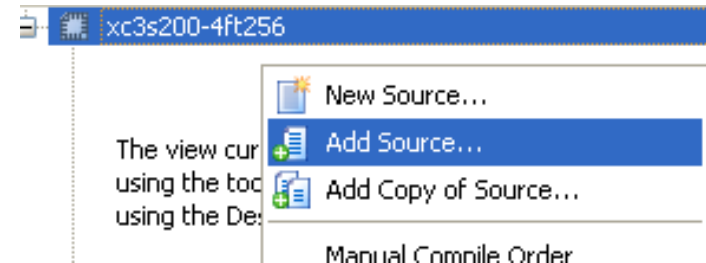


Δημ. \ Εισ. μονάδων



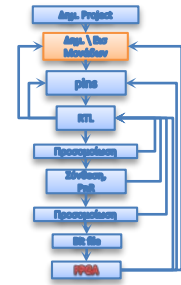
Όταν έχουμε έτοιμα τα .v αρχεία κάνουμε “Add Copy of Source...” αλλιώς “New Source...”

1. Αθροιστής (adder8.v),
2. 8bit register (reg8.v),
3. Τη μονάδα που εμφανίζει τις 2 προηγούμενες και τις διασύνδει (system.v),
4. το πλαίσιο ελέγχου ή testbench (tb.v),
5. τέλος το .ucf αρχείο που αντιστοιχεί τα σήματα εισόδου/εξόδου του κυκλώματος με τα pins της FPGA.
 - Το .ucf αρχείο μπορούμε είτε να το βάλουμε εξαρχής στο σχέδιο μας ή να το δημιουργήσουμε αργότερα μέσω του PlanAhead.






Δημ.\Εισ. μονάδων



1. Στο παράθυρο που εμφανίζεται,
2. διαλέγουμε το Association:
 - Για το πλαίσιο ελέγχου (tb.v) να είναι “simulation”
 - Για το αρχείο .ucf να είναι “Implementation”
 - Για τα υπόλοιπα, “all”
3. Τέλος, OK και συνεχίζουμε

	File Name	Association	Library
1	✓ tb.v	Simulation	work
2	✓ adder8.v	All	work
3	✓ reg8.v	All	work
4	✓ system.ucf	Implementation	work
5	✓ system.v	All	work



Adding files to project:  5 of 5 files (0 errors)

OK Cancel Help



Μαρ σημάτων με FPGA pins



A. Αν θέλουμε να δημιουργήσουμε το .ucf αρχείο κάνουμε τα εξής:

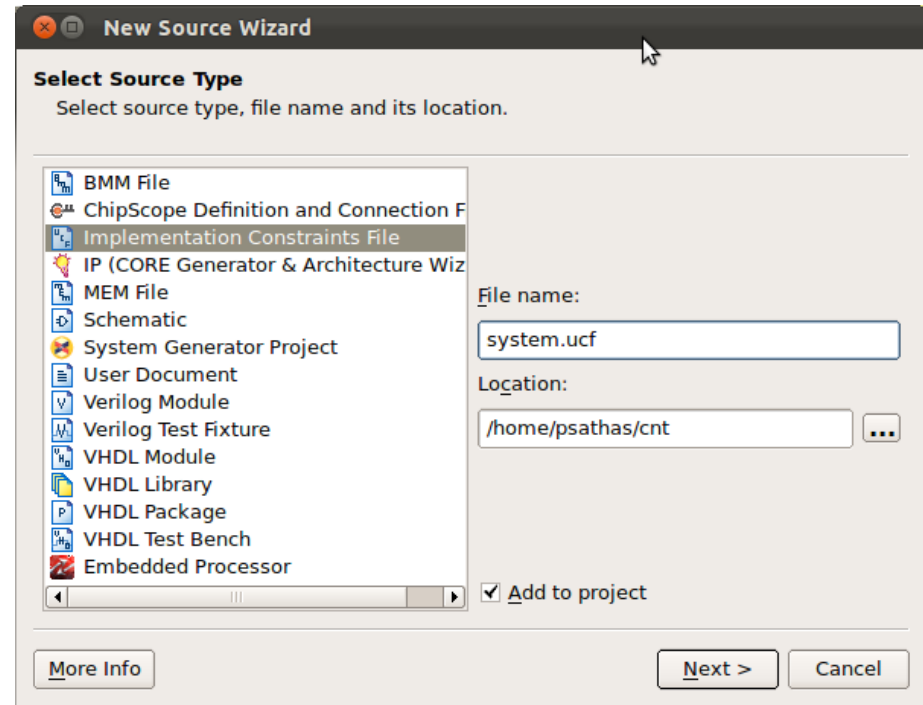
1. Δεξί κλικ στο top module (σε implementation view)
2. New source..
3. Επιλέγουμε “Implementation Constraints File”
4. Next..

B. Μέσα στο .ucf αρχείο κάνουμε μαρ τα I/O της FPGA, με τα I/O σήματα του κώδικα μας.

A. Κοιτάξετε τα mappings στο system.ucf.

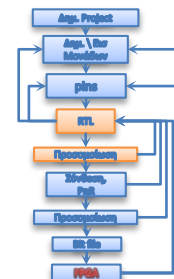
B. Η πλήρη λίστα με τα PINs της fpga βρίσκεται:

http://www.digilentinc.com/Data/Products/BASYS2/Basys2_100_250General.zip

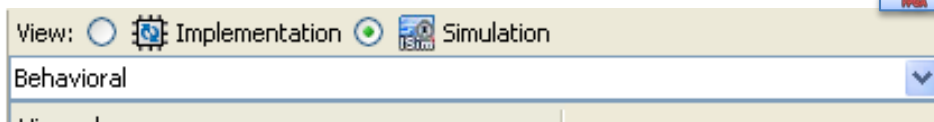




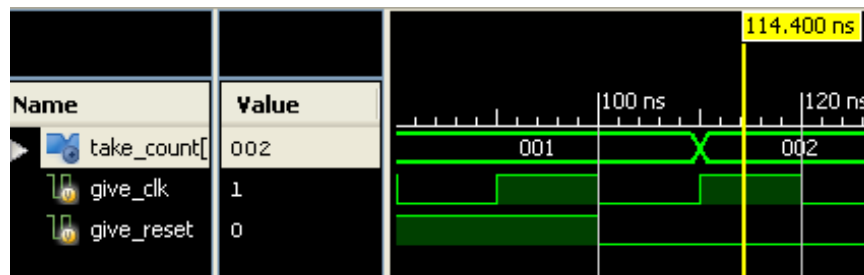
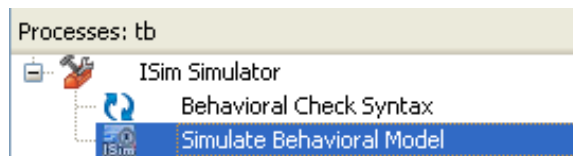
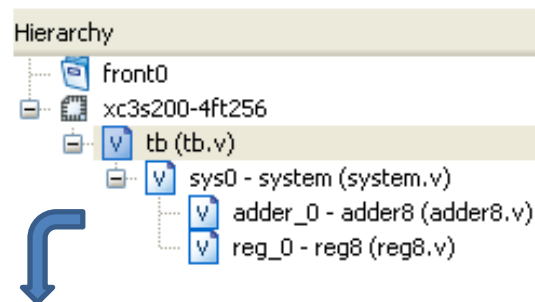
Post-RTL simulation



Έχοντας διαλέξει **ISim** :

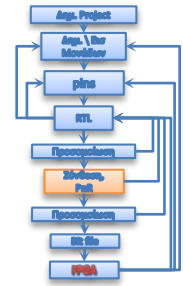


1. Αλλάζουμε το view σε Simulation, Behavioral.
2. Στο Hierarchy κάνουμε highlight το πλαίσιο ελέγχου (tb)
3. Στο Processes 2πλο-κλικ το Simulate Behavioral Model.





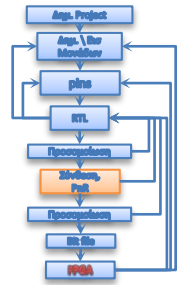
Σύνθεση, Τοποθέτηση και διασύνδεση



- A. Για να δημιουργηθεί το netlist, δηλ το αρχείο .v που περιέχει εμφανίσεις και διασυνδέσεις των δομικών modules του κυκλώματος (LUTs, ffs κ.α) (που επιλέχτηκαν κατά τη PnR διαδικασία) χρειαζόμαστε:
1. Τη βιβλιοθήκη (.v αρχεία) που περιγράφεται η λογική για κάθε module.
 2. Τις καθυστερήσεις για τη κάθε μονάδα και της κάθε διασύνδεσης.
- B. Όστε, όταν πάμε να κάνουμε προσομοίωση να έχουμε τα αρχεία:
1. netlist (`<project_dir>\netgen\par\system_timesim.v`)
 2. Τη βιβλιοθήκη (`<xilinx_dir>\ISE_DS\ISE\verilog\src\simprims*`)
 3. Τις καθυστερήσεις (`<proj_dir>\netgen\par\system_timesim.sdf`)
 - Παρατηρήστε ότι το .sdf αρχείο διαβάζεται μέσα από την εντολή `$sdf_annotate("<your_path\file.sdf">");` και βρίσκεται μέσα στο netlist γενικά πάντως μπαίνει μέσα σε πλαίσιο ελέγχου για να διατηρείται το netlist σε 'καθαρή' συνθέσιμη μορφή..
 4. Το Πλαίσιο ελέγχου (διατηρούμε το ίδιο πλαίσιο ελέγχου με αυτό που είχαμε κατά το RTL επίπεδο, το tb.v)



Σύνθεση, Τοποθέτηση και διασύνδεση

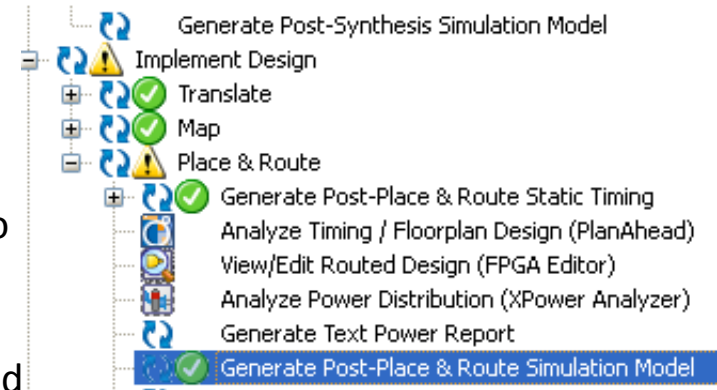
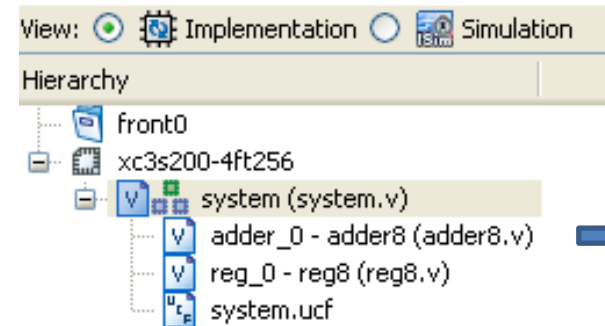


A. Οπότε, για το PnR χρειαζόμαστε να:

1. Επανεέλθουμε στο implementation view,
2. Κάνουμε highlight το top module (system)
3. Διπλό κλικ το Generate Post-P&E Sim. Model

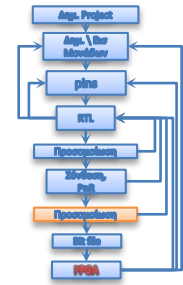
B. Και, αφού τελειώσει αυτή η διαδικασία μπορούμε έπειτα να:

1. Δούμε τη κάτοψη του FPGA με τις διασυνδέσεις των LUT μεταξύ τους,
2. Δούμε το slack της κάθε κρίσιμης διαδρομής ($slack = T_{clk} - D_{critical}$)
3. Κάνουμε προσομοίωση του κυκλώματος μετά το PnR όπου θα φαίνονται οι καθυστερήσεις των σημάτων λόγω των καλωδίων και της καθυστέρησης της κάθε πύλης που έγινε mapped στο υλικό (configured LUT) .



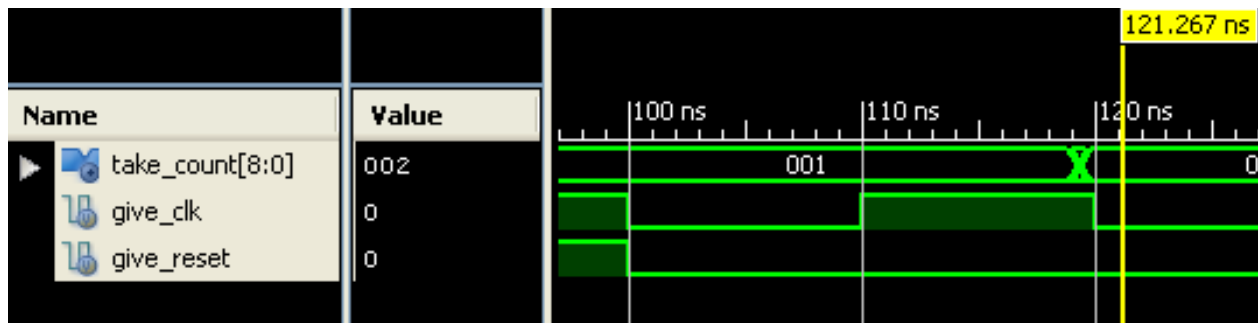
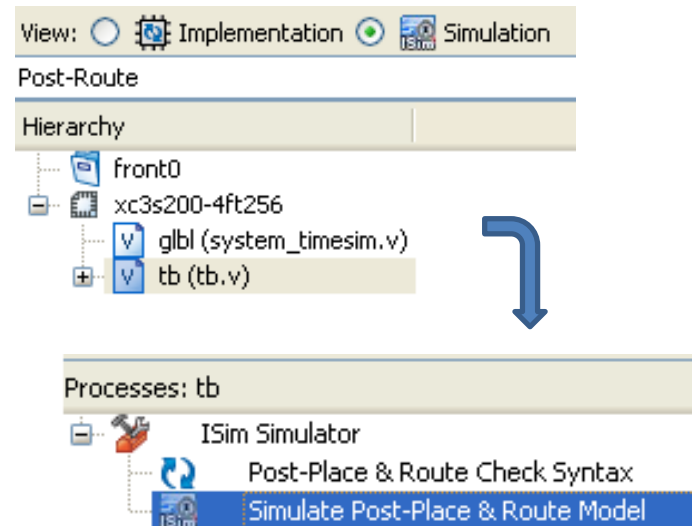


post-PnR simulation



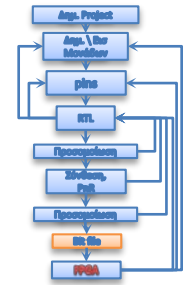
Isim:

1. Αλλάζουμε το view σε simulation, Post-Route
2. Highlight το πλαίσιο ελέγχου
3. 2πλό κλικ το Simulate Behavioral Model





Δημιουργία bit file



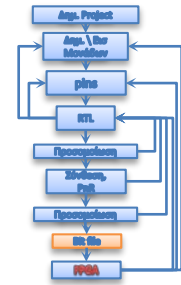
Αφού πλέον πιστεύουμε ότι το κύκλωμα μας θα δουλεύει στη συσκευή μπορούμε να παράγουμε το bit file το οποίο θα γίνει download στη πλακέτα ως εξής:

- Αλλαγή σε implementation view
- Highlight to το top module (εδώ system)
- 2πλό κλικ στο Generate Programming File
- Το bit file πλέον βρίσκεται στο project directory μας ως 'system.bit'
- Μετά, 2πλο κλικ στο Configure Target Device, και OK στο παράθυρο που βγαίνει.

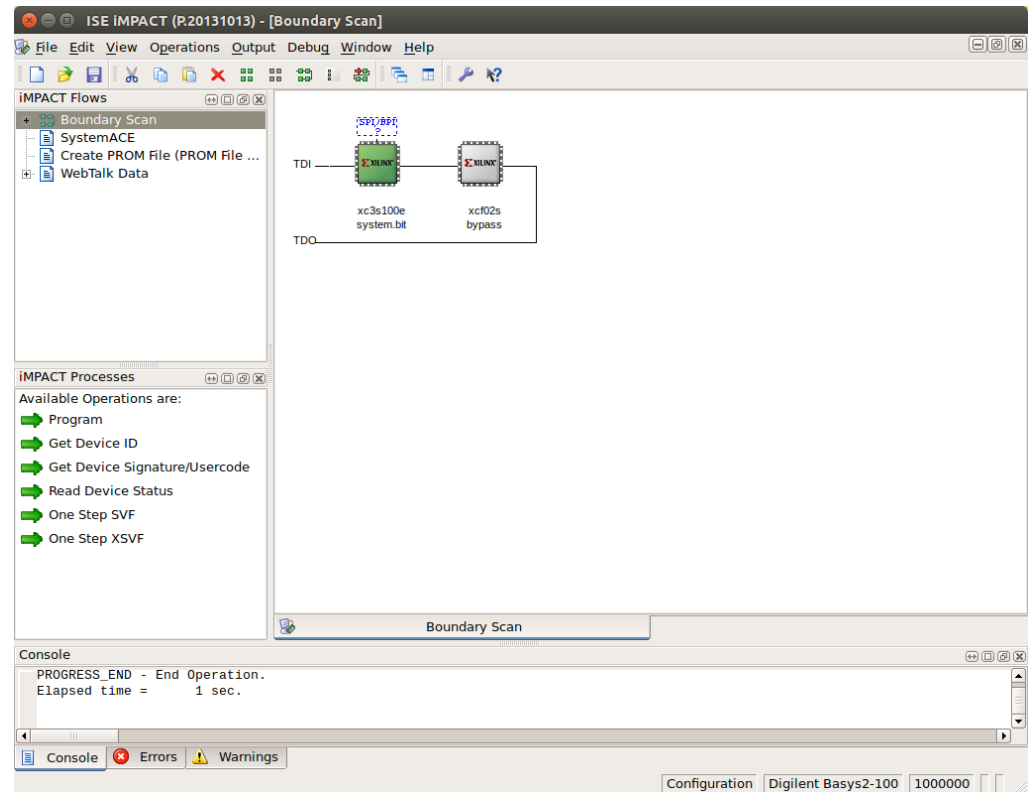
The screenshot shows the Xilinx IDE interface. At the top, the 'View' menu is set to 'Implementation'. Below it, the project hierarchy is displayed, with the 'system' module highlighted. A blue arrow points from the 'system' module to the 'Processes: system' window. In this window, the 'Generate Programming File' process is highlighted, indicating it is the current step in the workflow.



Δημιουργία bit file

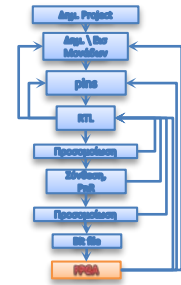


1. Διπλό κλικ στο Boundary Scan
2. Δεξί κλικ στο άσπρο πλαίσιο δεξιά
3. Initialize chain..
4. Yes
5. highlight to .bit του επόμενου παραθύρου και open
6. No στο παράθυρο «attach PROM»
7. Πατάμε bypass στο επόμενο παράθυρο μετά OK
8. Δεξί κλικ στο “πράσινο» program





FPGA



Τέλος ελέγχουμε αν λειτουργεί το κύκλωμα όπως τα αρχικά specifications.

Αν δεν συμφωνεί στα αρχικά specifications θα πρέπει να γυρίσουμε πίσω σε οποιοδήποτε από τα προηγούμενα στάδια ακολουθώντας ξανά τη ροή.