

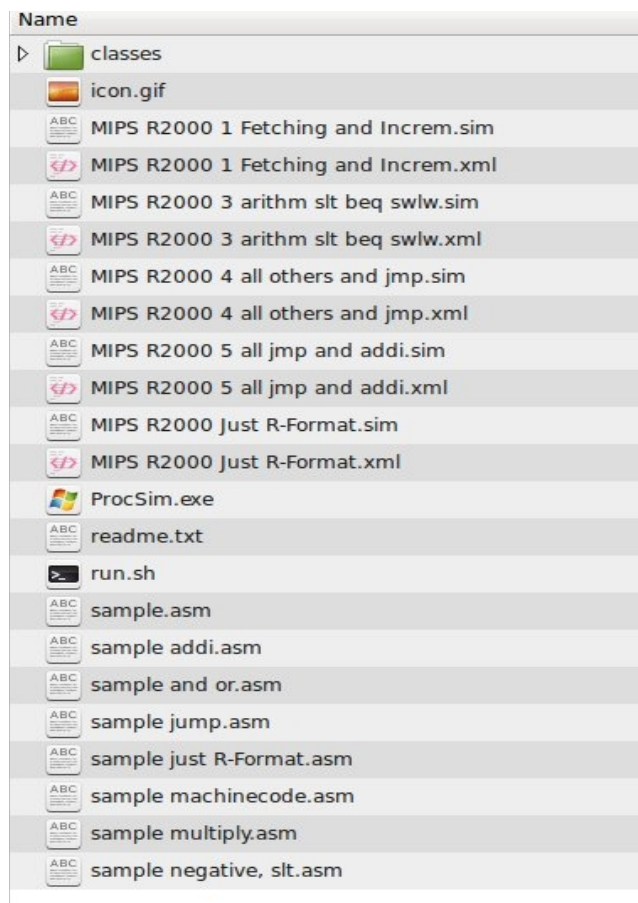
ProcSim – οδηγίες χρήσης.

Το ProcSim είναι ένα εργαλείο προσομοίωσης – Οπτικοποίησης του Datapath του απλού επεξεργαστή που περιγράφεται στο μάθημα (και στο βιβλίο). Το εργαλείο αυτό εκτελεί προγράμματα γραμμένα σε γλώσσα μηχανής όπως την έχετε διδαχθεί στο μάθημα και δείχνει τις τιμές όπως αυτές “ταξιδεύουν” μέσα στον επεξεργαστή. Επίσης παρέχει τα περιεχόμενα του αρχείου καταχωρητών (register file), της μνήμης εντολών και της μνήμης δεδομένων.

Το ProcSim μπορείτε να το βρείτε στο διαδίκτυο (<http://www.jamesgart.com/procsim/>) στην περιοχή του μαθήματος υπάρχει μια έκδοση με μια μικρή διόρθωση για linux, αντιγράψτε το αρχείο ProcSim.tar.gz στην περιοχή σας και αποσυμπίστε τα περιεχόμενα.

```
cp ~hy225/ProcSim/ProcSim.tar.gz .  
tar -xvf ProcSim.tar.gz
```

Ανοίξτε τον φάκελο ProcSim όπου θα δείτε τα εξής αρχεία:



Σε ένα τερματικό, αφού μεταφερθείτε στον φάκελο αυτό, πληκτρολογήστε ./run.sh

```
~ $ cd Desktop/  
~/Desktop $ cd ProcSim/  
~/Desktop/ProcSim $ ./run.sh
```

The screenshot displays the ProcSim v2.0 application window. The main window has a title bar "ProcSim v2.0 - by James Garton". On the left, there is a vertical list of components: Instr4, Instr5, Label, Label, InstrName0, param1-0, param2-0, param3-0, InstrName1, param1-1, param2-1, param3-1, InstrName2, param1-2, param2-2, param3-2, InstrName3, param1-3, param2-3, param3-3, InstrName4, param1-4, param2-4, param3-4, InstrName5, Machine, Machine, Machine, Machine, Machine. The main area of the window is divided into two panes. The top pane, titled "ProcSim Console", contains the following text: Instr4: and \$s7, \$s2, \$s1 #011 AND 101
Instr5: exit:
Label:main
Label:exit
InstrName0: add
param1-0: \$s3
param2-0: \$s1
param3-0: \$s1
InstrName1: ...
param1-1: ...
param2-1: ...
param3-1: ...
InstrName2: ...
param1-2: ...
param2-2: ...
param3-2: ...
InstrName3: ...
param1-3: ...
param2-3: ...
param3-3: ...
InstrName4: ...
param1-4: ...
param2-4: ...
param3-4: ...
InstrName5: ...
Machine
Machine
Machine
Machine
Machine
Starting parse or file: MIPS R2000 Just R-Format.xml
==XML Successfully Parsed==
Loaded component positions
Loaded bus positions
Loaded bus name positions
Loaded simulator diagram
The bottom pane is empty. The bottom of the window has two buttons: "Show/Hide Console Messages" and "Show/Hide Error Messages". A "Start Simulation" dialog box is open in the center of the screen. The dialog box has a title bar "ProcSim v2.0 - by James Garton". It contains a "Start Simulation" button at the top. Below it is a text area labeled "Assembly Code..." containing the text "Assembly: sample just R-Format.asm". Below the text area is a button labeled "Processor Architecture...". Below that is a text area labeled "SimPath: MIPS R2000 Just R-Format.xml". Below the text area is a button labeled "Exit".

The screenshot displays the MIPS R2000 Single-Cycle simulator interface. The main window shows the datapath, which includes components like the PC, InstrMemory, Register File, ALU, and ALUControl. The PC is initially set to 00000. The InstrMemory outputs the instruction at address 0, which is 0x00000000 (nop). The Register File shows registers \$0 through \$31, with \$0 containing 00000. The ALU is performing a no-operation (nop). The Control Panel shows the execution status as 'Start Execution'. The Instruction Memory window shows the instruction at address 0, which is 0x00000000 (nop).

File **Settings** **Help**

MIPS R2000 Single-Cycle - Just Arithmetic R-Format Instructions

Control Panel

Start Execution | Pause | Stop | Step

Registers | Main Memory | Instruction Memory

Speed: Super | Instant

Close

Registers

Reg	Bin Val	Dec Val
\$0	0..0	0

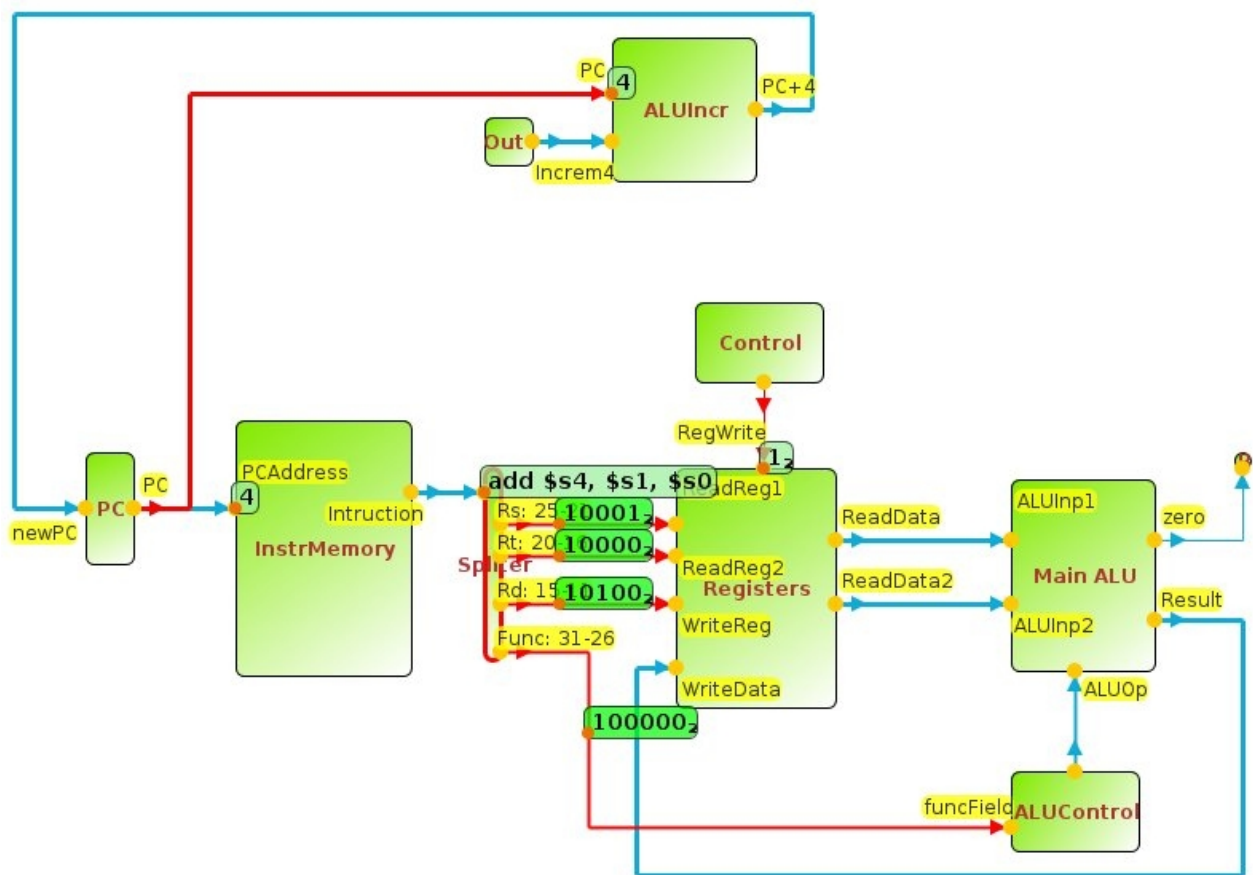
Show All Done

Instruction Memory

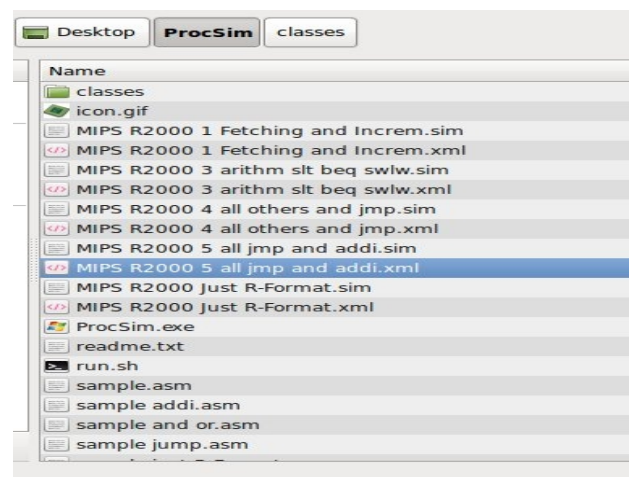
WordAddr	Addr	Instruction	Comment
0	0	main: add \$s3, \$s1, \$s1	#s+5
100	4	add \$s4, \$s1, \$s0	#s+10
1000	8	sub \$s5, \$s1, \$s2	#s-3
1100	12	sub \$s6, \$s2, \$s1	#s+5
10000	16	and \$s7, \$s2, \$s1	#011 ...
10100	20	exit:	

Select an instruction to see its comment.

Machine Code Done



Διαλέγοντας File → Open Sim απο το μενού του κυρίως παραθύρου μπορούμε να φορτώσουμε διαφορετικά datapaths απο τα διαφορετικά αρχεία .xml που βρίσκονται μέσα στο directory του ProcSim.



Τα Datapaths όλα αφορούν τον ίδιο επεξεργαστή που μάθαμε στο μάθημα, απλά το κάθε ένα είναι σε διαφορετικό στάδιο υλοποίησης.

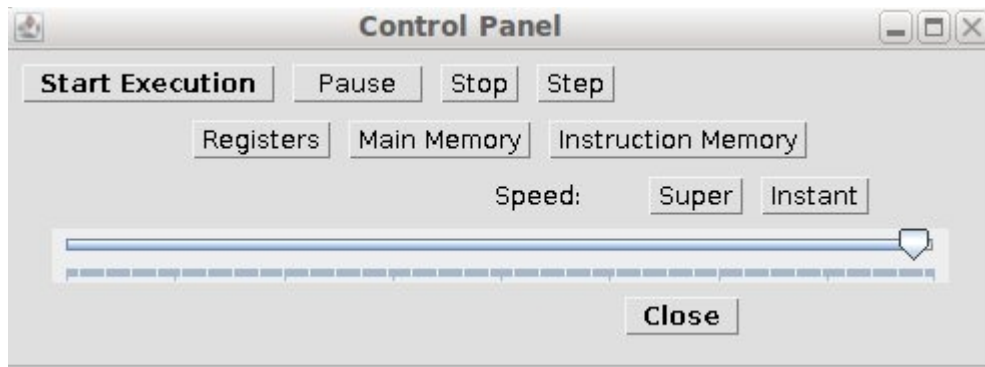
Π.χ: το **“MIPS R2000 1 Fetching and Increm.xml”** Δείχνει μόνο το κομμάτι του datapath για το fetch των εντολών και την άυξηση του Program Counter.

Control Panel:

Μπορείτε να ελέγξετε την προσομοίωση απο το παράθυρο “Control Panel”.

Πατώντας **Start Execution** ξεκινάει η προσομοίωση, μπορείτε να την σταματήσετε με τα πλήκτρα **pause** και **stop** και να ελέγξετε την ταχύτητα μέσω της μπάρας στο κάτω μέρος καθώς και μέσω των πλήκτρων **super** και **instant**. Πειραματιστείτε ελεύθερα.

Τα πλήκτρα “**Registers**”, “**Main Memory**” και “**Instruction Memory**” εμφανίζουν ή κρύβουν τα αντίστοιχα παράθυρα που δείχνουν τα περιεχόμενα των Μνημών (Δεδομενων και Instructions) και αρχείου καταχωρητών αντίστοιχα.



Registers:

Το παράθυρο “registers” δείχνει τα περιεχόμενα του Register File.

Πατήστε “Show all” για να δείτε τα περιεχόμενα όλων των καταχωρητών διαφορετικά φαίνονται μόνο οι καταχωρητές που έχουν χρησιμοποιηθεί απο το πρόγραμμα μέχρι εκείνη την στιγμή.



Registers			
Reg	Reg	Bin Val	Dec Val
00000	\$0	0..0	0
10000	\$s0	1111111111111111...	-10
10001	\$s1	0..101	5
10010	\$s2	0..11	3
10011	\$s3	0..1010	10

Instruction Memory:

Στο παράθυρο αυτό φαίνονται τα περιεχόμενα της μνήμης εντολών, είτε σε μορφή κώδικα assembly είτε σε δυαδική μορφή πατώντας το πλήκτρο “Machine code”

Instruction Memory			
WordAddr	Addr	Instruction	Comment
0	0	main: add \$s3, \$s1, \$s1	#5+5
100	4	add \$s4, \$s1, \$s0	#5+ -10
1000	8	sub \$s5, \$s1, \$s2	#5-3
1100	12	sub \$s6, \$s2, \$s1	#3-5
10000	16	and \$s7, \$s2, \$s1	#011 AND 101
10100	20	exit:	

Αντίστοιχα, υπάρχει και το παράθυρο της κύριας μνήμης (Μνήμης Δεδομένων) με την δυνατότητα να δείχνει όλες τις θέσεις μνήμης ή μόνο αυτές που έχουν χρησιμοποιηθεί.

Main Memory

ByteAddr	WordAddr	ByteVal	WordVal
0	0	0	0
1		0	
2		0	
3		0	
4	1	0	0
5		0	
6		0	
7		0	
8	2	0	0
9		0	
10		0	
11		0	
12	3	0	0
13		0	

Hide Empty Done

Οδηγίες εγκατάστασης για Windows

Αφού κατεβάσετε το procsim από την περιοχή του μαθήματος ή απο το site :

<http://jamesgart.com/procsim/>

όπου θα επιλέξετε να κατεβάσετε την έκδοση που δείχνει η παρακάτω εικόνα

Download (1.03MB) ProcessorSim Windows - Just for Windows.

Download (2.11MB) ProcessorSim - should work on most OSs (read the readme.txt).

You need the new Java 1.5 (v5) runtimes. You can download it [here](#).

Minimum Requirements:

Processor: 1GHz

Memory: 64MB

Resolution Monitor: 1024*768

Windows: 2000/XP (Not tested on other systems - but should work on Linux and most other OSs as it is Java based)

Θα αποσυμπιέσετε το αρχείο που κατεβάσατε θα ανοίξετε τον φάκελο και θα τρέξετε το procsim πατώντας στο εικονίδιο της παρακάτω εικόνας.

MIPS R2000 Just R-Format	9/19/2005 10:26 PM	XML File	9 KB
ProcSim	9/26/2005 2:20 PM	Application	1,248 KB
readme	11/15/2005 3:17 PM	TXT File	10 KB

Προσοχή!!! Όταν κατεβάσετε τα αρχεία από την περιοχή του μαθήματος και το ProcSim θα πρέπει να βρίσκονται σε φακέλους όπου το path τους δε θα περιέχει ελληνικούς χαρακτήρες για παράδειγμα το path **"C:\users\Γιώργος\ProcSim"** θα προκαλέσει προβλήματα.

Εάν βεβαιωθήκατε ότι το path δεν περιέχει ελληνικούς χαρακτήρες και πάλι δεν τρέχει και σας βγάζει διάφορα error πρέπει να κατεβάσουμε την Java SE Runtime Enviroment 8.

Πηγαίνετε στον παρακάτω σύνδεσμο :

<http://www.oracle.com/technetwork/java/javase/downloads/jre8-downloads-2133155.html>

και κατεβάστε την **32-bit** έκδοση της JR8 για windows όπως φαίνεται στην εικόνα :

Java SE Runtime Environment 8u161		
You must accept the Oracle Binary Code License Agreement for Java SE to download this software.		
☐ Accept License Agreement ☒ Decline License Agreement		
Product / File Description	File Size	Download
Linux x86	63.4 MB	jre-8u161-linux-i586.rpm
Linux x86	79.29 MB	jre-8u161-linux-i586.tar.gz
Linux x64	60.4 MB	jre-8u161-linux-x64.rpm
Linux x64	76.35 MB	jre-8u161-linux-x64.tar.gz
macOS	74.17 MB	jre-8u161-macosx-x64.dmg
macOS	65.86 MB	jre-8u161-macosx-x64.tar.gz
Solaris SPARC 64-bit	52.24 MB	jre-8u161-solaris-sparcv9.tar.gz
Solaris x64	50 MB	jre-8u161-solaris-x64.tar.gz
Windows x86 Online	1.78 MB	jre-8u161-windows-i586-iftw.exe
Windows x86 Offline	61.35 MB	jre-8u161-windows-i586.exe
Windows x86	64.56 MB	jre-8u161-windows-i586.tar.gz
Windows x64 Offline	68.02 MB	jre-8u161-windows-x64.exe
Windows x64	68.58 MB	jre-8u161-windows-x64.tar.gz

Αφού κατέβει το αρχείο **jre-8uXXX-windows-i586** ανοίξτε το και ολοκληρώστε την εγκατάσταση της JR8. Μετά την εγκατάσταση το ProcSim θα πρέπει να λειτουργεί κανονικά.