

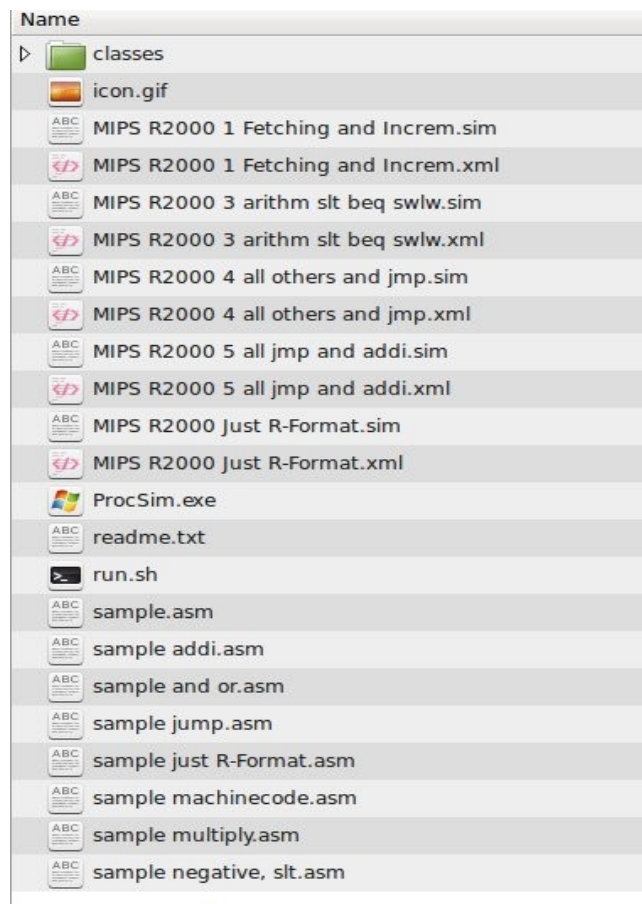
## ProcSim – οδηγίες χρήσης.

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 αποσυμπιέστε τα περιεχόμενα.

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

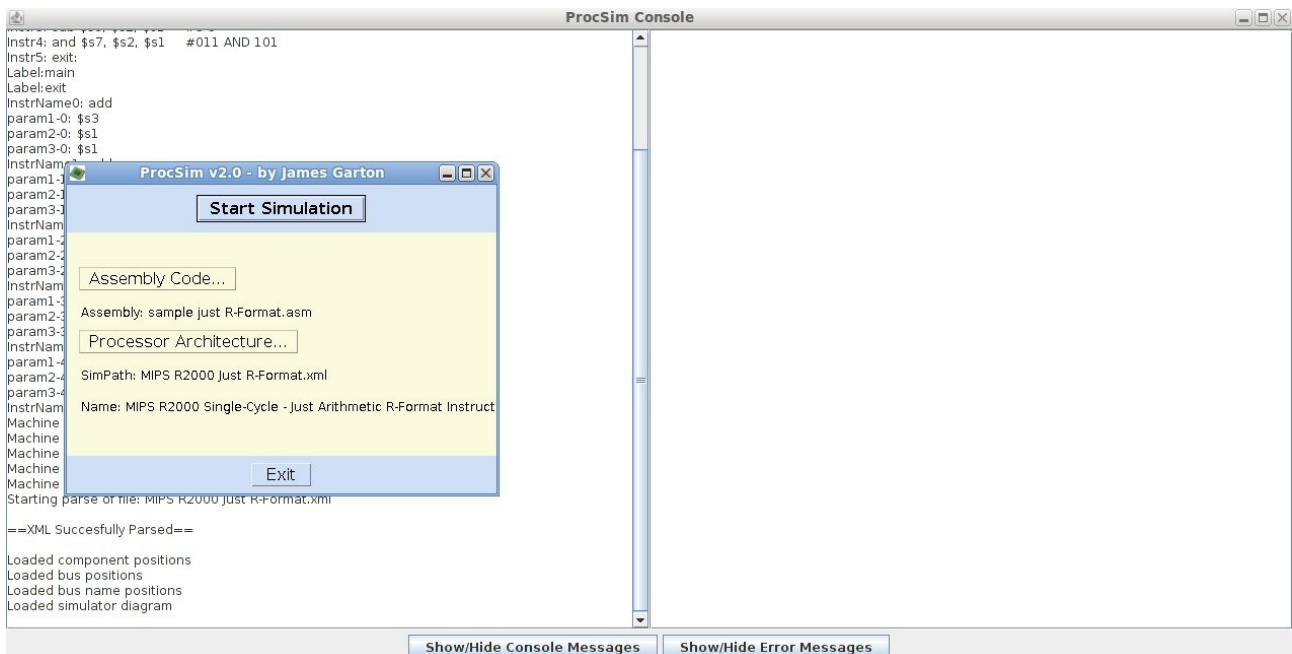
Ανοίξτε τον φάκελο ProcSim όπου θα δείτε τα εξής αρχεία:



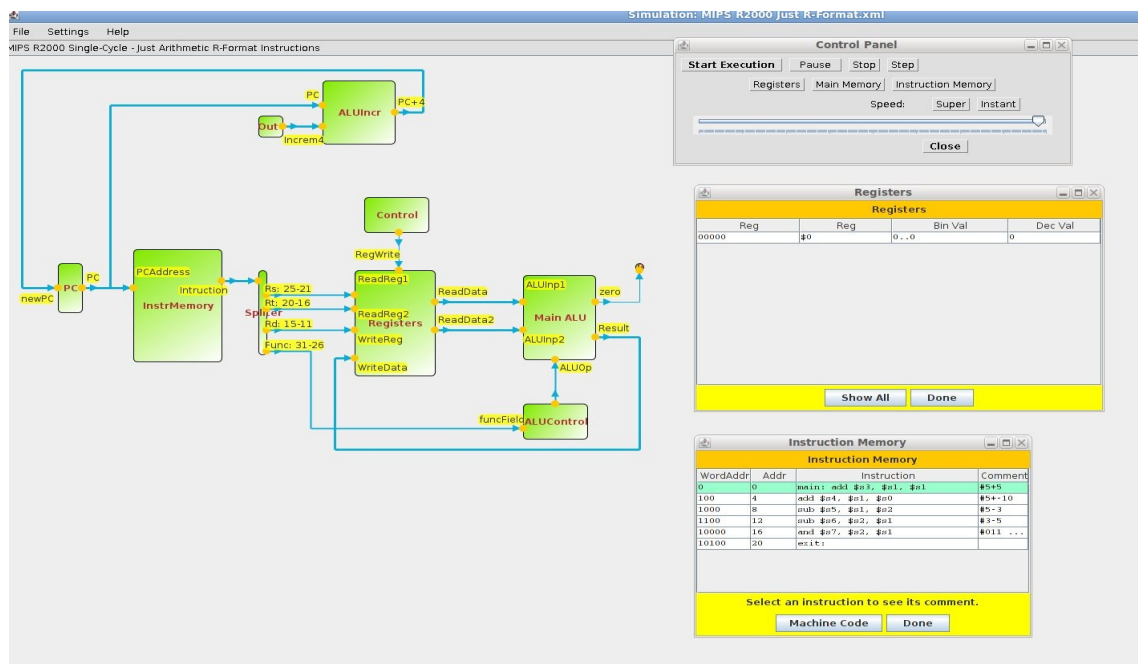
Σε ένα τερματικό, αφού μεταφερθείτε στον φάκελο αυτό, πληκτρολογήστε ./run.sh

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

Εναλλακτικά, μπορείτε να το τρέξετε σε Windows εκτελώντας το “ProcSim.exe”.  
Αφού το κάνετε αυτό θα δείτε να ανοίγουν τα παράθυρα:



Πατώντας “Start Simulation” θα δείτε να ανοίγουν μερικά άλλα παράθυρα.



Το κυρίως παράθυρο έχει το σχέδιο του datapath που προσομοιώνουμε.

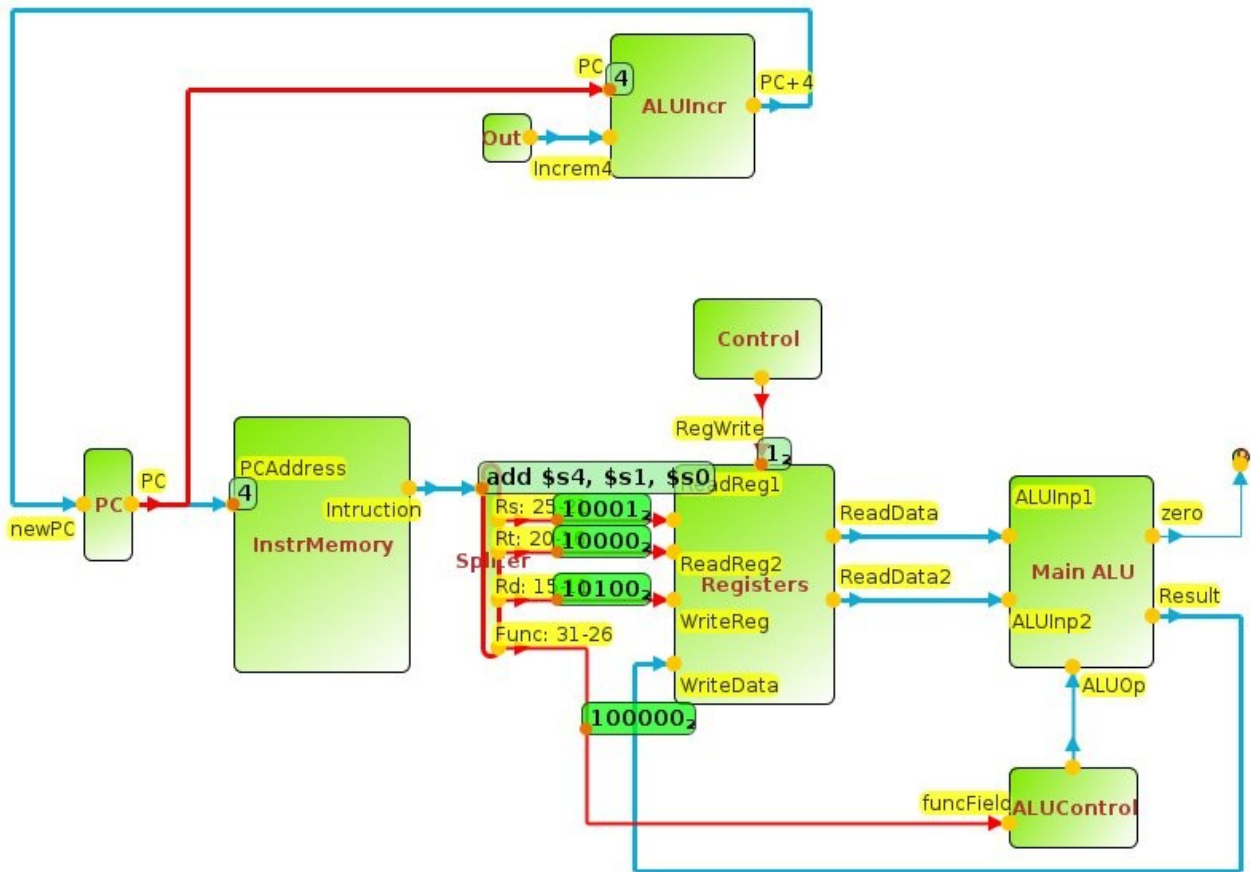
Τα άλλα τρία είναι από πάνω προς τα κάτω όπως εμφανίζονται τα:

**Control Panel:** Ορίζει την ταχύτητα της προσομοίωσης και ποια άλλα παράθυρα θα φαίνονται.

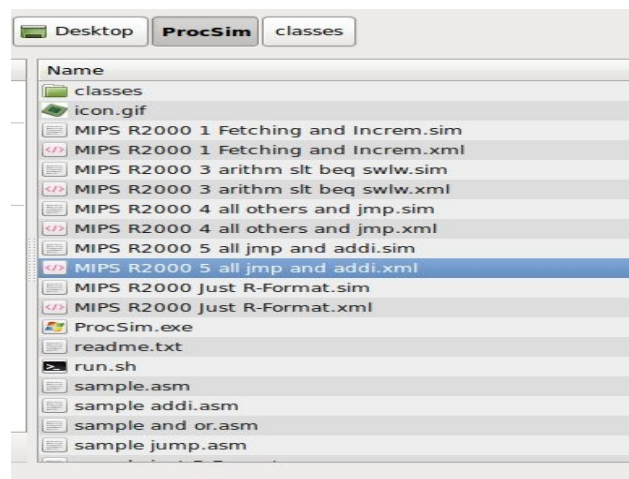
**Registers:** Δείχνει τα περιεχόμενα του register file.

**Instruction Memory:** Δείχνει τα περιεχόμενα της μνήμης εντολών.

Πατώντας “start execution” στο παράθυρο “Control Panel” ξεκινάει η προσομοίωση της εκτέλεσης και μπορούμε να δούμε τις τιμές πάνω στα καλώδια καθώς αυτές ‘ταξιδεύουν’ από το ένα μέρος του datapath στο άλλο.

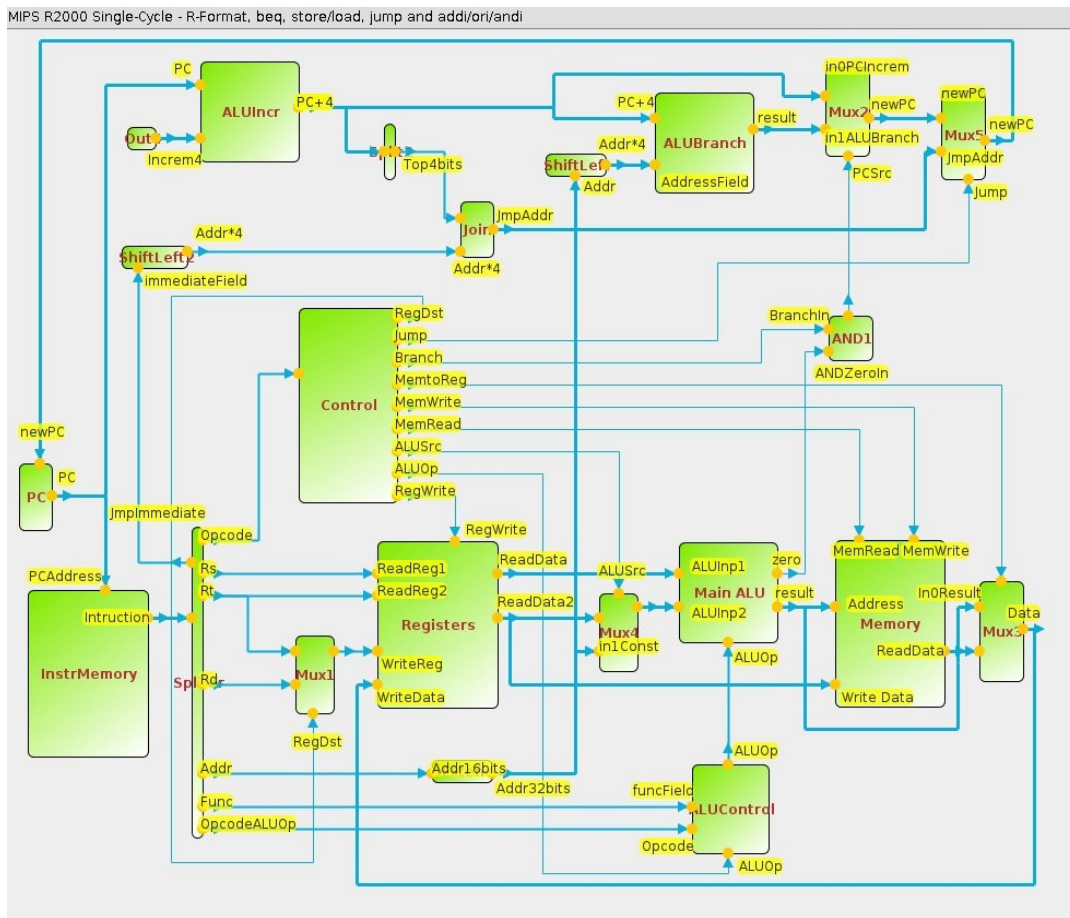


Διαλέγοντας File → Open Sim απο το μενού του κυρίως παραθύρου μπορούμε να φορτώσουμε διαφορετικά datapaths απο τα διαφορετικά αρχεία .xml που βρίσκονται μέσα στο directory του ProcSim.



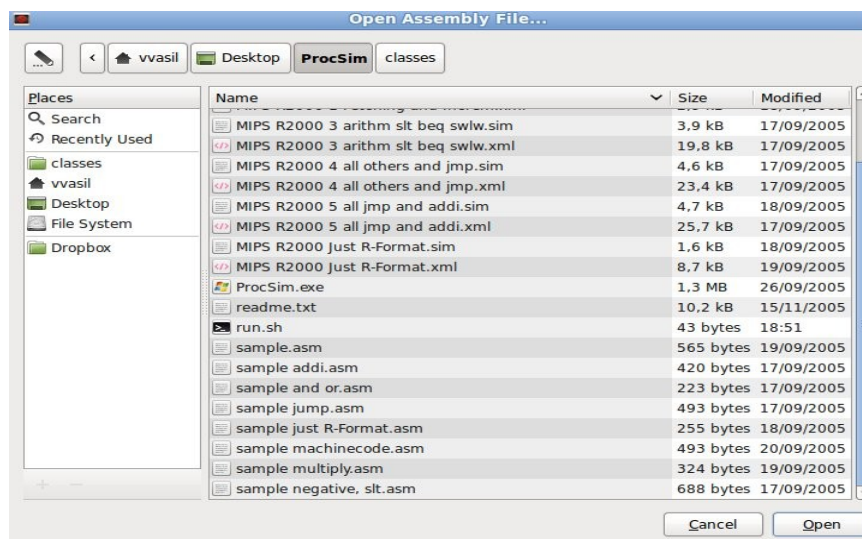
Τα Datarpaths όλα αφορούν τον ίδιο επεξεργαστή που μάθαμε στο μάθημα, απλά το κάθε ένα είναι σε διαφορετικό στάδιο υλοποίησης.  
 Π.χ: το **“MIPS R2000 1 Fetching and Increm.xml”** Δείχνει μόνο το κομμάτι του datapath για το fetch των εντολών και την αύξηση του Program Counter.

Αφου διαλέξουμε κάποιο απο αυτά (Στο παράδειγμα μας το “MIPS R2000 5 all jmp and addi.xml” που είναι και το πληρέστερο), βλέπουμε το νέο datarath να σχηματίζεται στο κυρίως παράθυρο:



Μπορούμε επίσης να διαλέξουμε διαφορετικά προγράμματα που θα τρέξουμε στην προσομοίωση μας επιλέγοντας file → Open Assembly απο το μενου του κυρίως παραθύρου και επιλεοντας κάποιο .asm αρχείο απο το directory του ProcSim.

Τα περιεχόμενα του νεου αρχείου .asm φαινονται στο παράθυρο “Instruction Memory”.

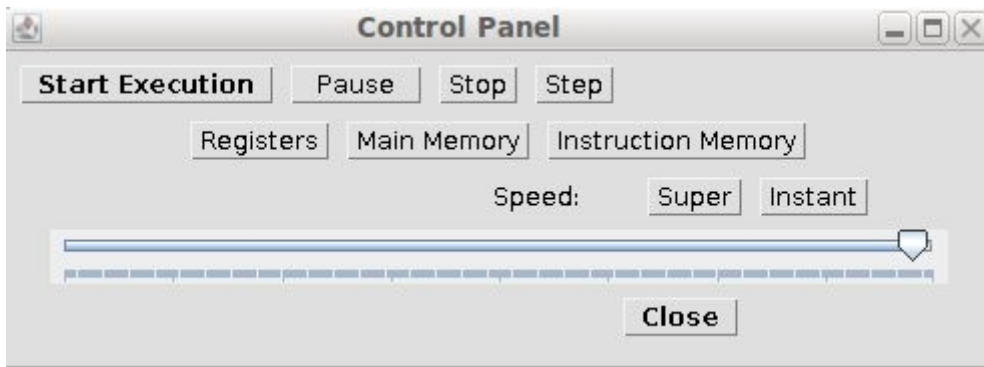


### Control Panel:

Μπορείτε να ελέγξετε την προσομοίωση από 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 (Δεδομένων και Instructions) και αρχείου καταχωρητών αντίστοιχα.



### Registers:

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

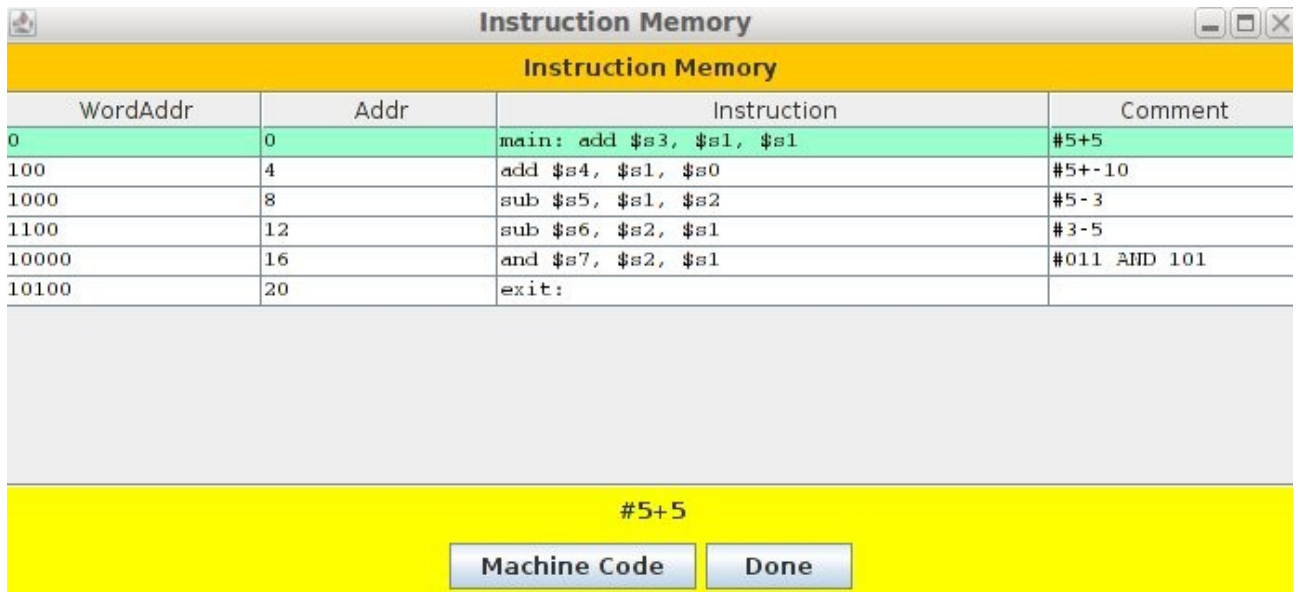
Πατήστε “Show all” για να δείτε τα περιεχόμενα όλων των καταχωρητών διαφορετικά φαίνονται μόνο οι καταχωρητές που έχουν χρησιμοποιηθεί από το πρόγραμμα μέχρι εκείνη την στιγμή.



Reg	Reg	Bin Val	Dec Val
00000	\$0	0..0	0
10000	\$s0	11111111111111111111...	-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”



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:	

#5+5

Machine Code Done

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.



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