ECE 551 Intermediate Report

Gregory Duma
Maxwell Eggers
Matthew Weimer

4/4/2014

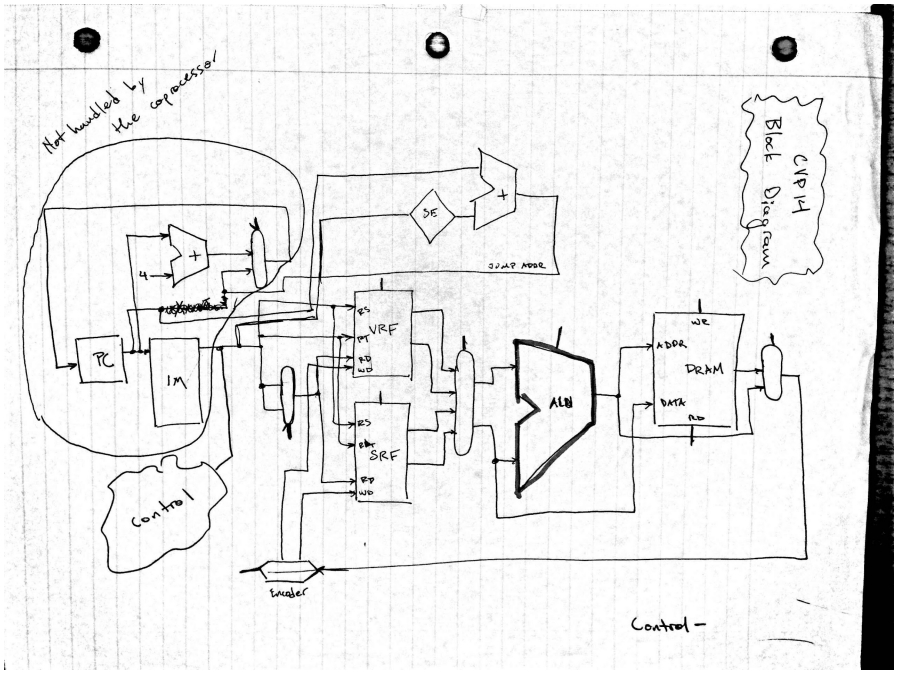
Plan for completing the design and the optimizing the design:

- Refactor large modules (ie. float_add) into smaller ones for more efficient compilation/synthesis
- Vector addition
    - add overflow/underflow and rounding
    - make multi-cycle to decrease area (by a huge amount)
- Assess timing paths report on our synthesized modules to minimize worst path delay

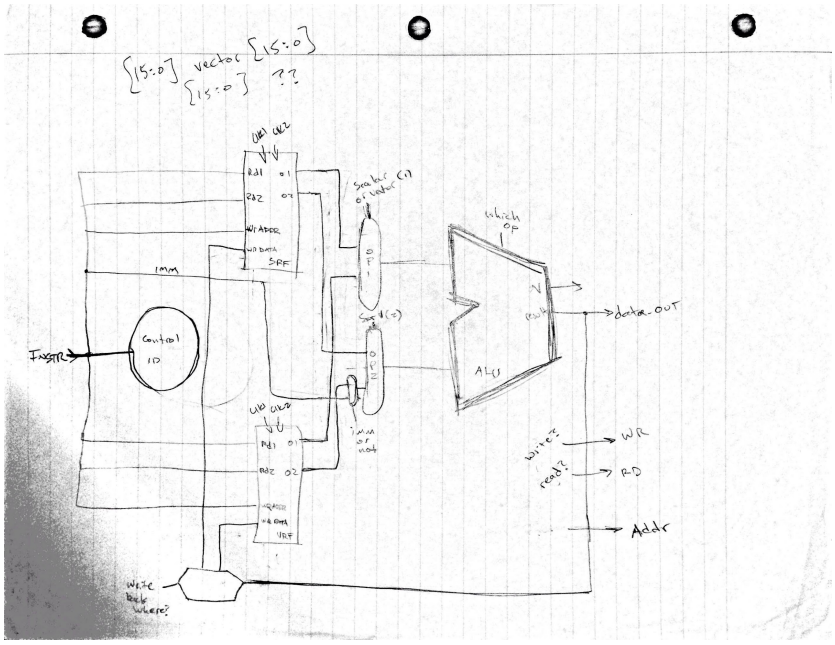
Team Contribution:

We began by meeting up and drawing out a block diagram. Matthew provided most of the background for our initial candidate architecture from his previous knowledge gained in his 552 course. From here the group quickly worked together to complete the scalar and vector register files as we believed that these files would be the simplest place to begin. Following the completion of the register files we split up and began working separately. Maxwell began working on the VADD function as we believed that this would be the most complicated aspect of the intermediate milestone. Matthew worked with the memory component of the coprocessor and obtained a deeper understanding of how the memory interacted with the coprocessor and how instructions would be pulled into the coprocessor. Matthew and Gregory began writing the control block of code in the CVP14 module, only to realize that with our original architecture we had no sufficient method of tracking the timings of the coprocessor. Following Max's completion of the VADD function Matthew and Max collaborated to reorganize our project so that all data could be passed through the separate modules and that the timings of the clocks would be respected in this new design. Additional modules were then written by Max and Matt to facilitate these new timing requirements, and allow for the testing of our design via the test.asm file. Greg then began to synthesize the design and provide area reports for the intermediate milestone report.

Our original block diagram. Utilized multiple muxes and extraneous modules that we later omitted. Notable features are the VRF, SRF ALU and DRAM.



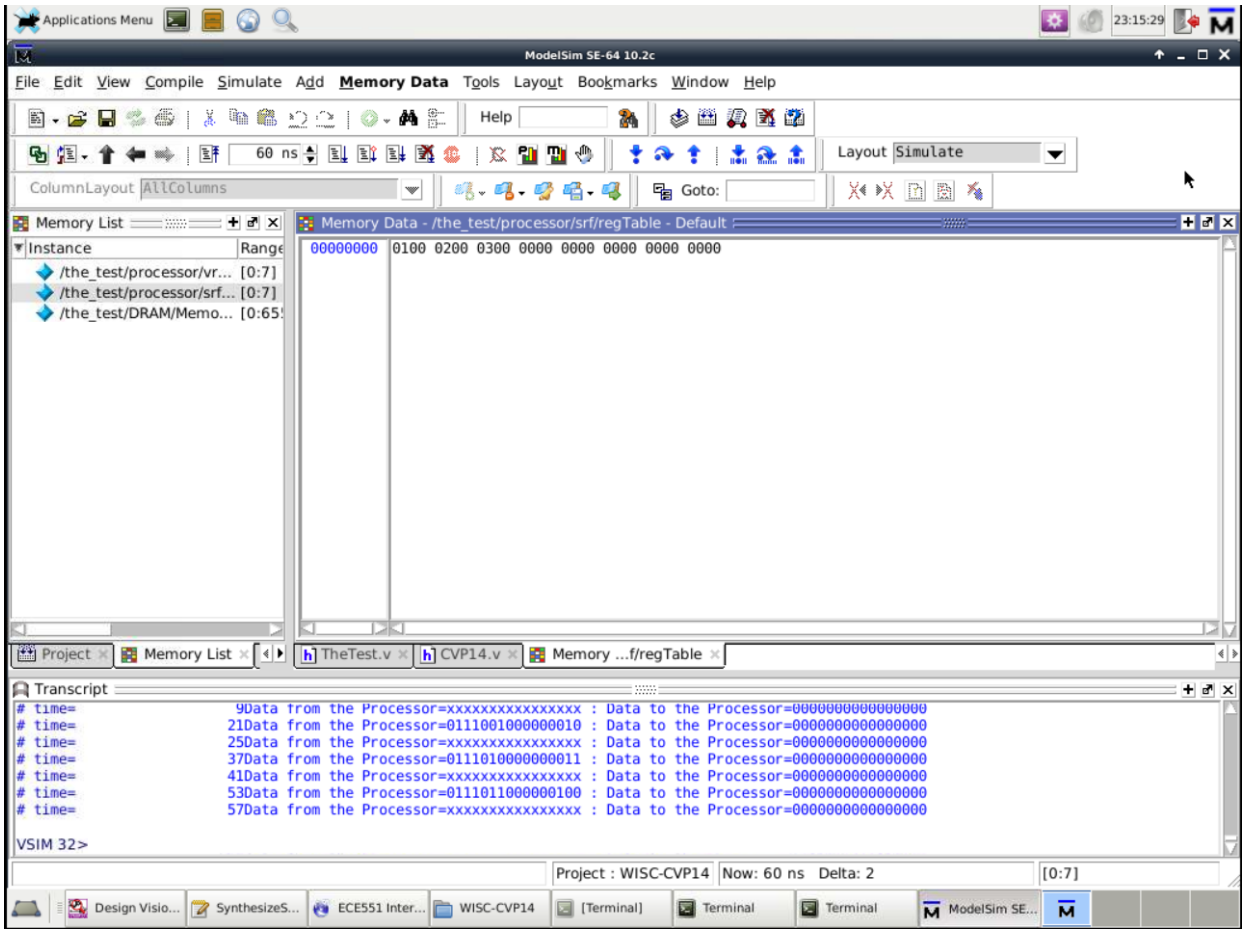
Our revised block diagram. Includes the notable features from the first diagram along with a control block, a more clearly defined operand passing structure, and clearly defined inputs and outputs from the CVP14 module.



Below is the data returned from our processor throughout the test given to us. The data on the left is the data passed through the processor, it includes all instructions and the timings of when they are received. This output shows how long each individual command is taking, with the scalar commands taking a far shorter amount of time than the following vector commands. By looking at this in combination with our timing reports we should be able to determine a significantly more efficient code with respect to time.

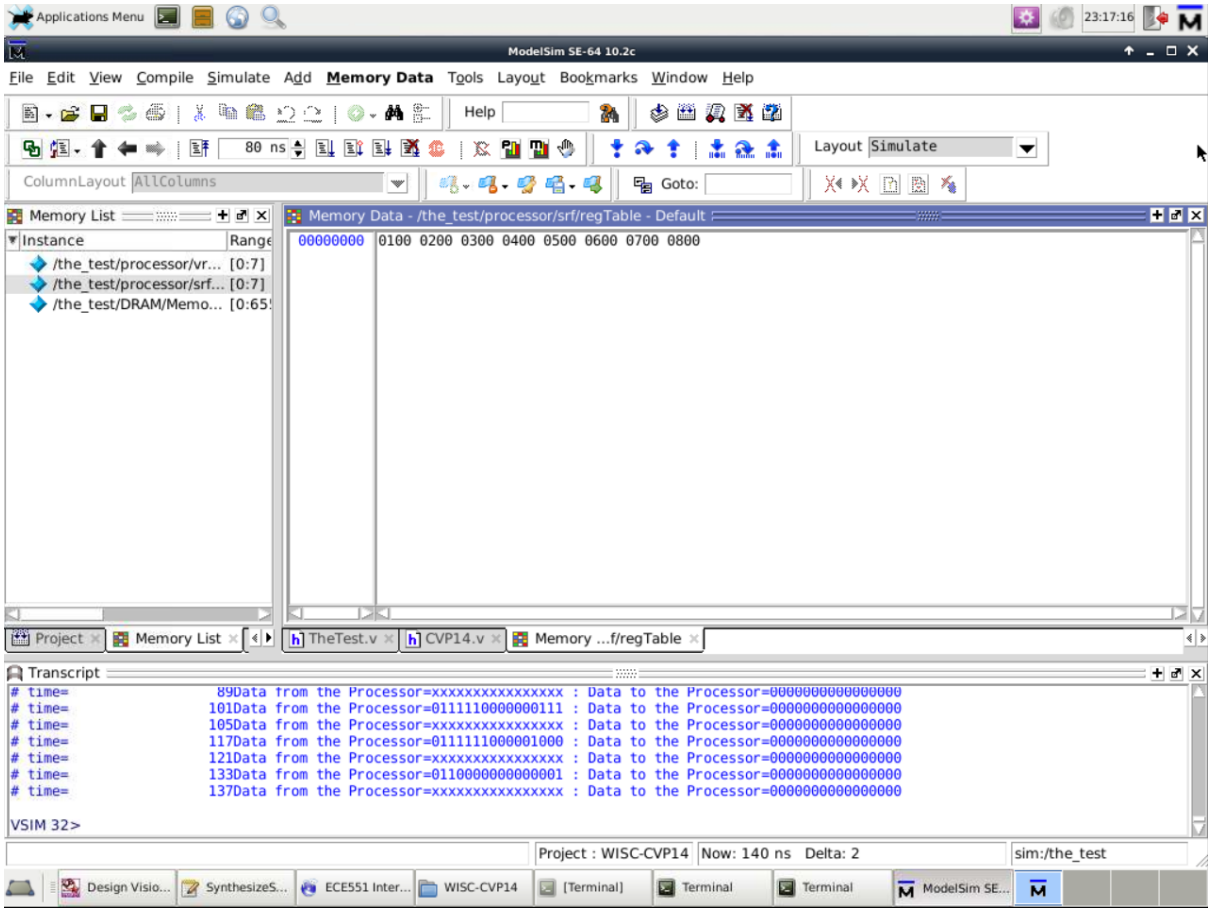
The initial SLH commands

time= 2Data from the Processor=xxxxxxxxxxxxxxxx : Data to the Processor=0000000000000000
time= 5Data from the Processor=0111000000000001 : Data to the Processor=0000000000000000
time= 9Data from the Processor=xxxxxxxxxxxxxxxx : Data to the Processor=0000000000000000
time= 21Data from the Processor=0111000000000001 : Data to the Processor=0000000000000000
time= 25Data from the Processor=xxxxxxxxxxxxxxxx : Data to the Processor=0000000000000000
time= 37Data from the Processor=0111001000000010 : Data to the Processor=0000000000000000
time= 41Data from the Processor=xxxxxxxxxxxxxxxx : Data to the Processor=0000000000000000
time= 53Data from the Processor=0111010000000011 : Data to the Processor=0000000000000000
time= 57Data from the Processor=xxxxxxxxxxxxxxxx : Data to the Processor=0000000000000000



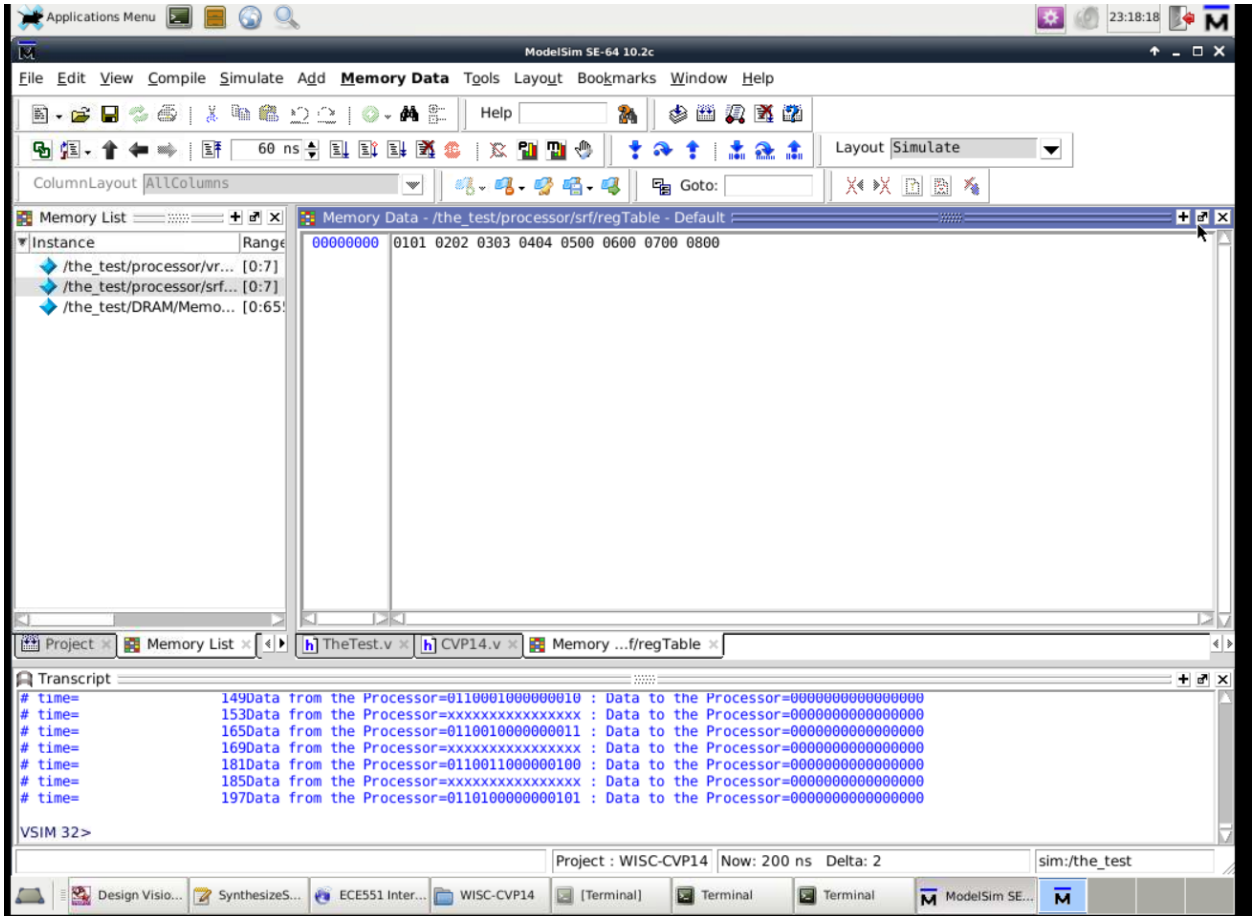
time= 69Data from the Processor=0111011000000100 : Data to the Processor=0000000000000000
time= 73Data from the Processor=xxxxxxxxxxxxxxxx : Data to the Processor=0000000000000000
time= 85Data from the Processor=0111100000000101 : Data to the Processor=0000000000000000
time= 89Data from the Processor=xxxxxxxxxxxxxxxx : Data to the Processor=0000000000000000
time= 101Data from the Processor=0111101000000110 : Data to the Processor=0000000000000000
time= 105Data from the Processor=xxxxxxxxxxxxxxxx : Data to the Processor=0000000000000000

time= 117Data from the Processor=0111110000000111 : Data to the Processor=0000000000000000
time= 121Data from the Processor=xxxxxxxxxxxxxxx : Data to the Processor=0000000000000000
time= 133Data from the Processor=0111111000001000 : Data to the Processor=0000000000000000

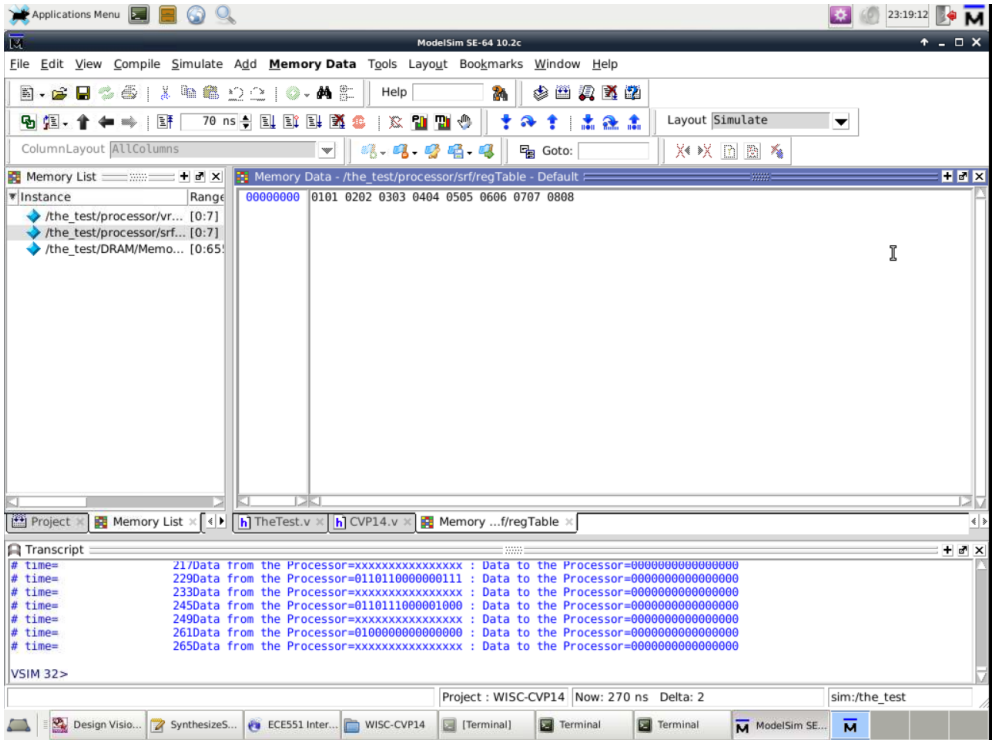


The SLL commands

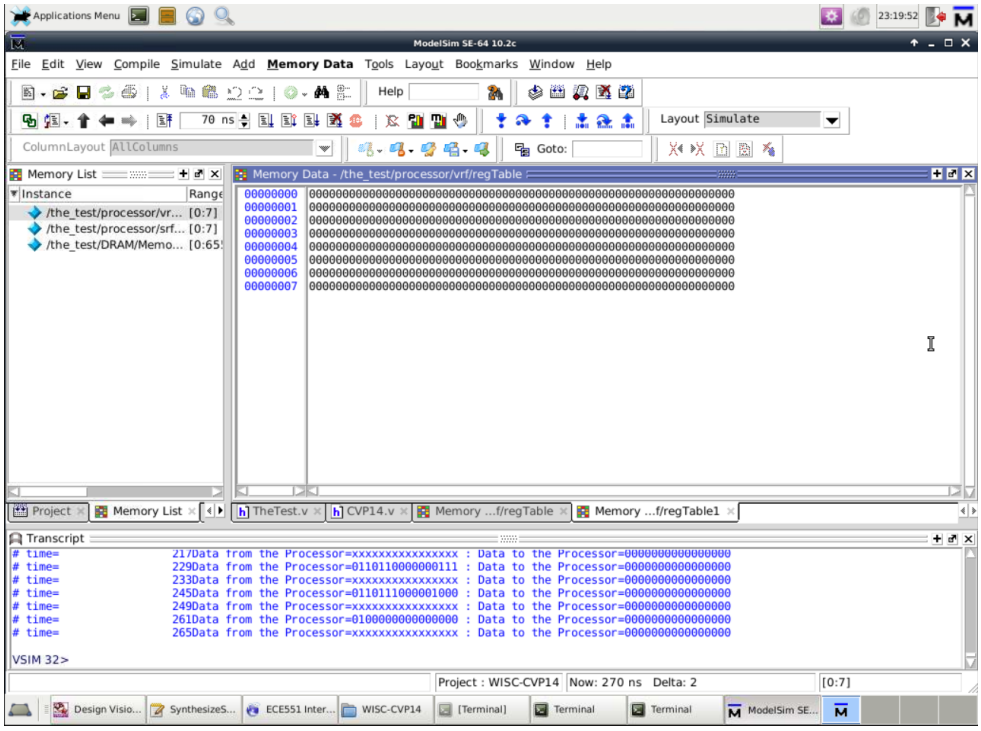
time= 137Data from the Processor=xxxxxxxxxxxxxxx : Data to the Processor=0000000000000000
time= 149Data from the Processor=0110000000000001 : Data to the Processor=0000000000000000
time= 153Data from the Processor=xxxxxxxxxxxxxxx : Data to the Processor=0000000000000000
time= 165Data from the Processor=0110001000000010 : Data to the Processor=0000000000000000
time= 169Data from the Processor=xxxxxxxxxxxxxxx : Data to the Processor=0000000000000000
time= 181Data from the Processor=0110010000000011 : Data to the Processor=0000000000000000
time= 185Data from the Processor=xxxxxxxxxxxxxxx : Data to the Processor=0000000000000000
time= 197Data from the Processor=0110011000000100 : Data to the Processor=0000000000000000



time= 201Data from the Processor=xxxxxxxxxxxxxxxx : Data to the Processor=0000000000000000
time= 213Data from the Processor=0110100000000101 : Data to the Processor=0000000000000000
time= 217Data from the Processor=xxxxxxxxxxxxxxxx : Data to the Processor=0000000000000000
time= 229Data from the Processor=0110101000000110 : Data to the Processor=0000000000000000
time= 233Data from the Processor=xxxxxxxxxxxxxxxx : Data to the Processor=0000000000000000
time= 245Data from the Processor=0110110000000111 : Data to the Processor=0000000000000000
time= 249Data from the Processor=xxxxxxxxxxxxxxxx : Data to the Processor=0000000000000000
time= 261Data from the Processor=0110111000001000 : Data to the Processor=0000000000000000
time= 265Data from the Processor=xxxxxxxxxxxxxxxx : Data to the Processor=0000000000000000

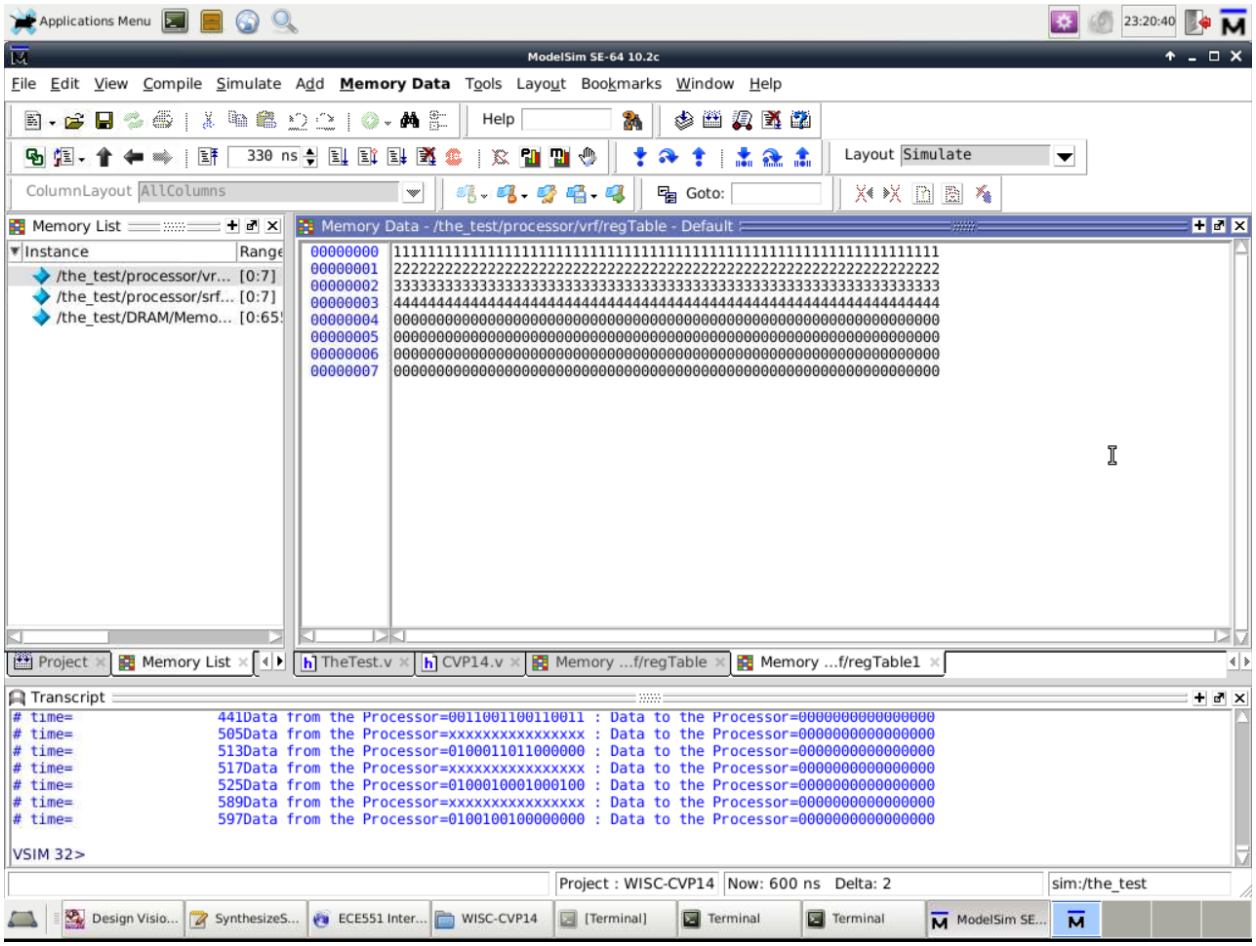


The vector load



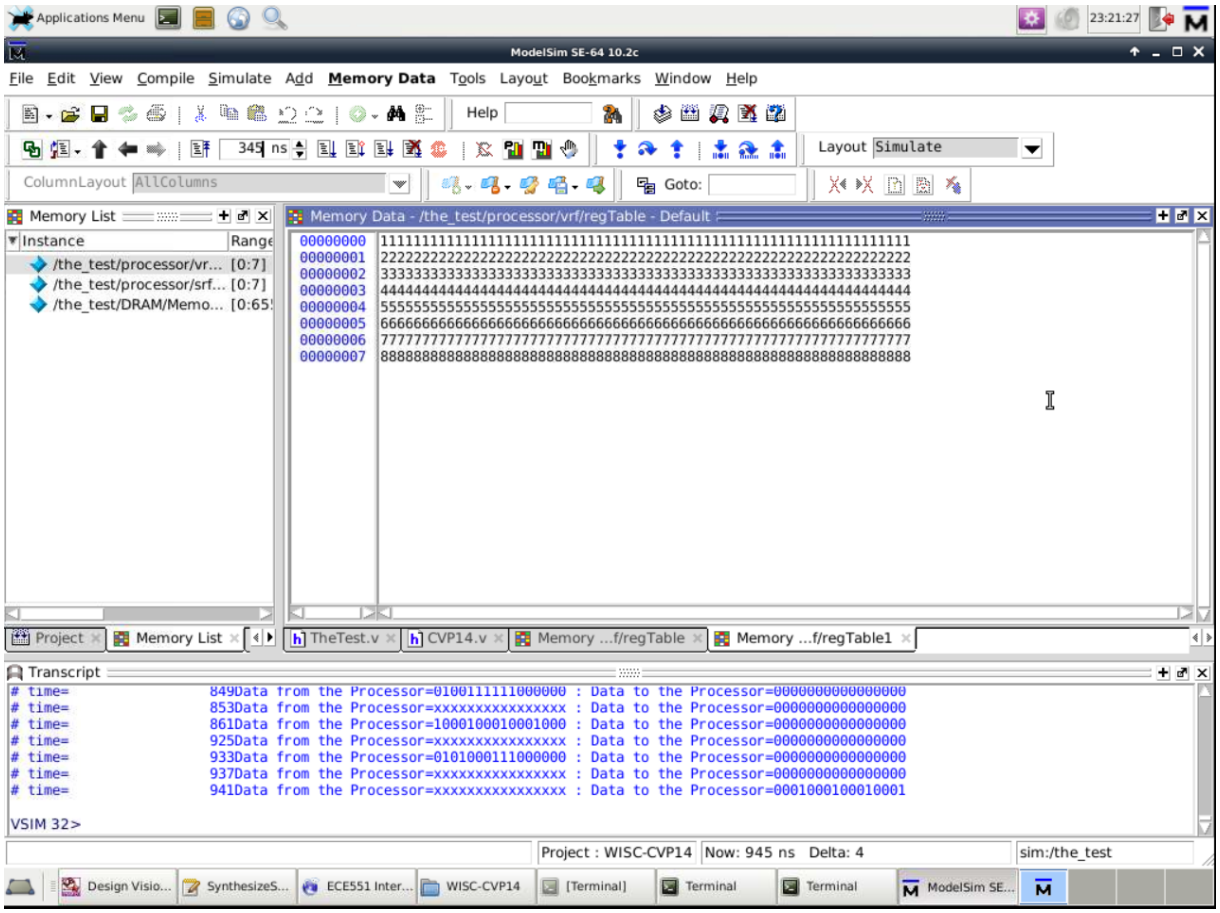
time= 277Data from the Processor=0100000000000000 : Data to the Processor=0000000000000000
time= 281Data from the Processor=xxxxxxxxxxxxxxxx : Data to the Processor=0000000000000000
time= 289Data from the Processor=0001000100010001 : Data to the Processor=0000000000000000
time= 353Data from the Processor=xxxxxxxxxxxxxxxx : Data to the Processor=0000000000000000
time= 361Data from the Processor=0100001001000000 : Data to the Processor=0000000000000000
time= 365Data from the Processor=xxxxxxxxxxxxxxxx : Data to the Processor=0000000000000000

time= 373Data from the Processor=0010001000100010 : Data to the Processor=0000000000000000
time= 437Data from the Processor=xxxxxxxxxxxxxxxx : Data to the Processor=0000000000000000
time= 445Data from the Processor=0100010010000000 : Data to the Processor=0000000000000000
time= 449Data from the Processor=xxxxxxxxxxxxxxxx : Data to the Processor=0000000000000000
time= 457Data from the Processor=0011001100110011 : Data to the Processor=0000000000000000
time= 521Data from the Processor=xxxxxxxxxxxxxxxx : Data to the Processor=0000000000000000
time= 529Data from the Processor=0100011011000000 : Data to the Processor=0000000000000000
time= 533Data from the Processor=xxxxxxxxxxxxxxxx : Data to the Processor=0000000000000000
time= 541Data from the Processor=0100010001000100 : Data to the Processor=0000000000000000



time= 605Data from the Processor=xxxxxxxxxxxxxxxx : Data to the Processor=0000000000000000
time= 613Data from the Processor=0100100100000000 : Data to the Processor=0000000000000000
time= 617Data from the Processor=xxxxxxxxxxxxxxxx : Data to the Processor=0000000000000000
time= 625Data from the Processor=0101010101010101 : Data to the Processor=0000000000000000
time= 689Data from the Processor=xxxxxxxxxxxxxxxx : Data to the Processor=0000000000000000
time= 697Data from the Processor=0100101101000000 : Data to the Processor=0000000000000000
time= 701Data from the Processor=xxxxxxxxxxxxxxxx : Data to the Processor=0000000000000000
time= 709Data from the Processor=0110011001100110 : Data to the Processor=0000000000000000
time= 773Data from the Processor=xxxxxxxxxxxxxxxx : Data to the Processor=0000000000000000
time= 781Data from the Processor=0100110110000000 : Data to the Processor=0000000000000000
time= 785Data from the Processor=xxxxxxxxxxxxxxxx : Data to the Processor=0000000000000000
time= 793Data from the Processor=0111011101110111 : Data to the Processor=0000000000000000
time= 857Data from the Processor=xxxxxxxxxxxxxxxx : Data to the Processor=0000000000000000
time= 865Data from the Processor=0100111111000000 : Data to the Processor=0000000000000000
time= 869Data from the Processor=xxxxxxxxxxxxxxxx : Data to the Processor=0000000000000000
time= 877Data from the Processor=1000100010001000 : Data to the Processor=0000000000000000

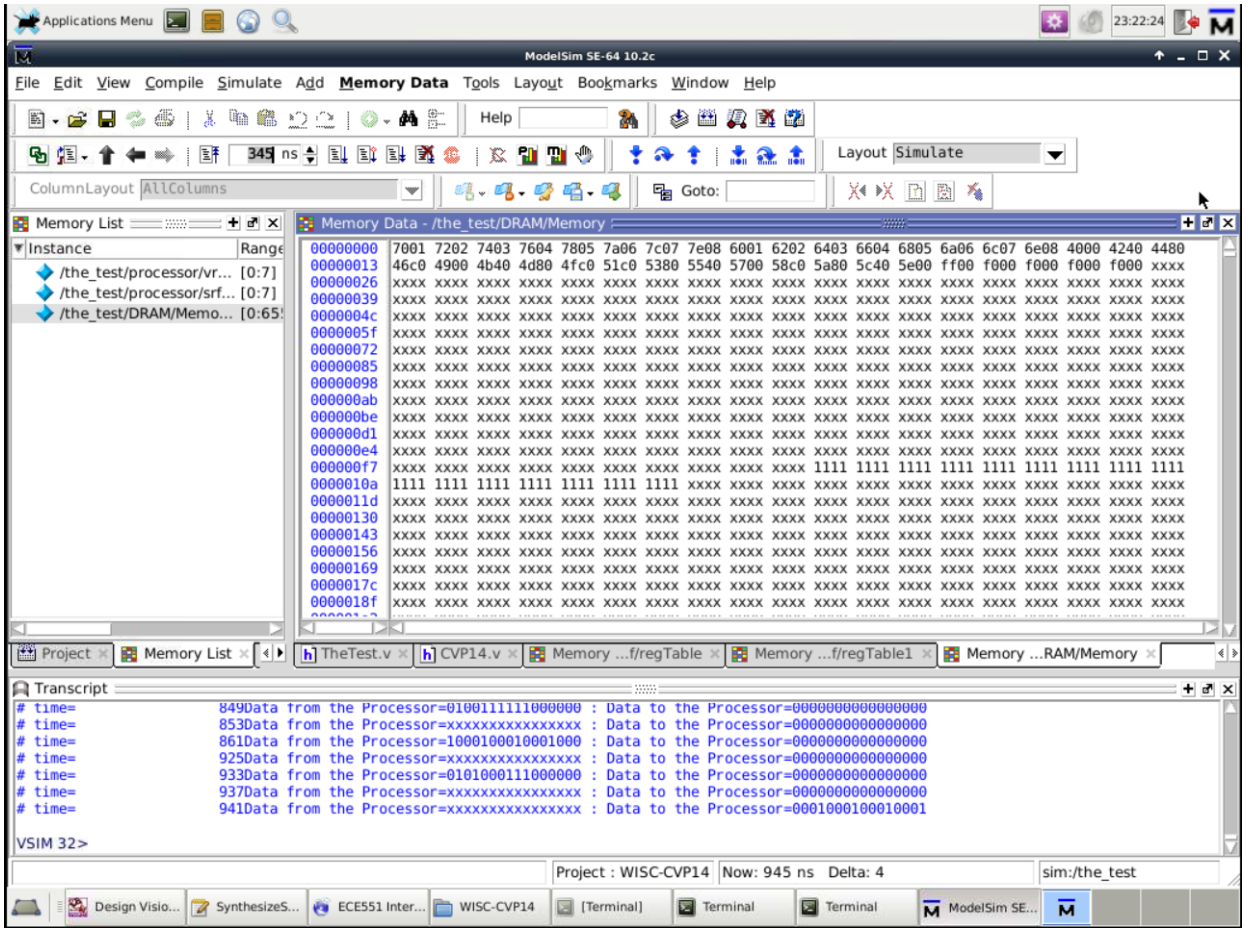
time= 941Data from the Processor=xxxxxxxxxxxxxxx : Data to the Processor=0000000000000000



The Vector Store commands

time= 949Data from the Processor=0101000111000000 : Data to the Processor=0000000000000000
time= 953Data from the Processor=xxxxxxxxxxxxxxx : Data to the Processor=0000000000000000
time= 957Data from the Processor=xxxxxxxxxxxxxxx : Data to the Processor=0001000100010001
time= 1021Data from the Processor=xxxxxxxxxxxxxxx : Data to the Processor=0000000000000000
time= 1025Data from the Processor=0101001110000000 : Data to the Processor=0000000000000000
time= 1029Data from the Processor=xxxxxxxxxxxxxxx : Data to the Processor=0000000000000000
time= 1033Data from the Processor=xxxxxxxxxxxxxxx : Data to the Processor=0010001000100010
time= 1097Data from the Processor=xxxxxxxxxxxxxxx : Data to the Processor=0000000000000000
time= 1101Data from the Processor=0101010101000000 : Data to the Processor=0000000000000000
time= 1105Data from the Processor=xxxxxxxxxxxxxxx : Data to the Processor=0000000000000000
time= 1109Data from the Processor=xxxxxxxxxxxxxxx : Data to the Processor=0011001100110011
time= 1173Data from the Processor=xxxxxxxxxxxxxxx : Data to the Processor=0000000000000000
time= 1177Data from the Processor=0101011100000000 : Data to the Processor=0000000000000000
time= 1181Data from the Processor=xxxxxxxxxxxxxxx : Data to the Processor=0000000000000000
time= 1185Data from the Processor=xxxxxxxxxxxxxxx : Data to the Processor=0100010001000100
time= 1249Data from the Processor=xxxxxxxxxxxxxxx : Data to the Processor=0000000000000000
time= 1253Data from the Processor=0101100011000000 : Data to the Processor=0000000000000000
time= 1257Data from the Processor=xxxxxxxxxxxxxxx : Data to the Processor=0000000000000000
time= 1261Data from the Processor=xxxxxxxxxxxxxxx : Data to the Processor=0101010101010101
time= 1325Data from the Processor=xxxxxxxxxxxxxxx : Data to the Processor=0000000000000000
time= 1329Data from the Processor=0101101010000000 : Data to the Processor=0000000000000000
time= 1333Data from the Processor=xxxxxxxxxxxxxxx : Data to the Processor=0000000000000000
time= 1337Data from the Processor=xxxxxxxxxxxxxxx : Data to the Processor=0110011001100110
time= 1401Data from the Processor=xxxxxxxxxxxxxxx : Data to the Processor=0000000000000000

time= 1405Data from the Processor=0101110001000000 : Data to the Processor=0000000000000000
time= 1409Data from the Processor=xxxxxxxxxxxxxxxx : Data to the Processor=0000000000000000
time= 1413Data from the Processor=xxxxxxxxxxxxxxxx : Data to the Processor=0111011101110111
time= 1477Data from the Processor=xxxxxxxxxxxxxxxx : Data to the Processor=0000000000000000
time= 1481Data from the Processor=0101111000000000 : Data to the Processor=0000000000000000
time= 1485Data from the Processor=xxxxxxxxxxxxxxxx : Data to the Processor=0000000000000000
time= 1489Data from the Processor=xxxxxxxxxxxxxxxx : Data to the Processor=1000100010001000
time= 1553Data from the Processor=xxxxxxxxxxxxxxxx : Data to the Processor=0000000000000000
time= 1557Data from the Processor=1111111100000000 : Data to the Processor=0000000000000000
time= 1561Data from the Processor=xxxxxxxxxxxxxxxx : Data to the Processor=0000000000000000



The finishing NOOP commands

time= 1573Data from the Processor=1111000000000000 : Data to the Processor=0000000000000000
time= 1577Data from the Processor=xxxxxxxxxxxxxxxx : Data to the Processor=0000000000000000
time= 1589Data from the Processor=1111000000000000 : Data to the Processor=0000000000000000
time= 1593Data from the Processor=xxxxxxxxxxxxxxxx : Data to the Processor=0000000000000000
time= 1605Data from the Processor=1111000000000000 : Data to the Processor=0000000000000000
time= 1609Data from the Processor=xxxxxxxxxxxxxxxx : Data to the Processor=0000000000000000
time= 1621Data from the Processor=1111000000000000 : Data to the Processor=0000000000000000