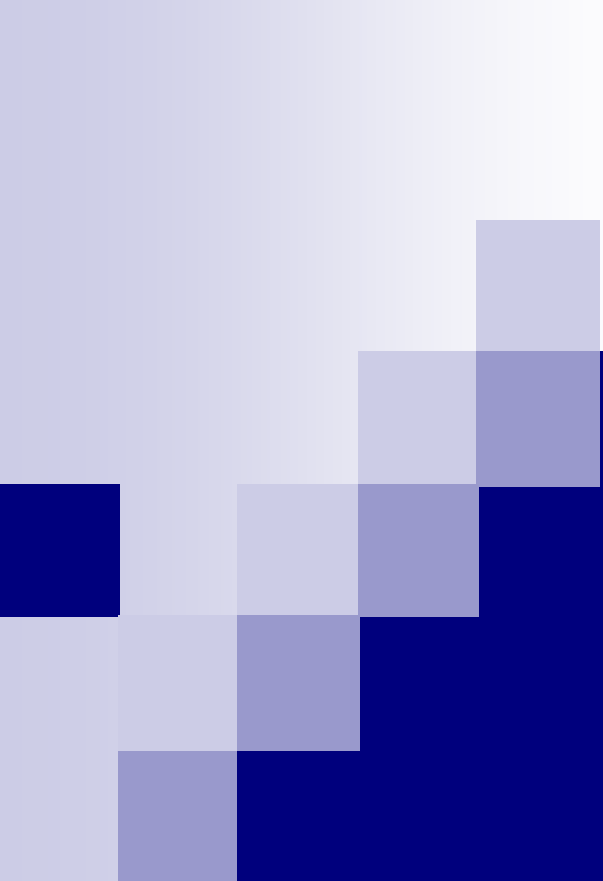




A Simple SPIM Tutorial

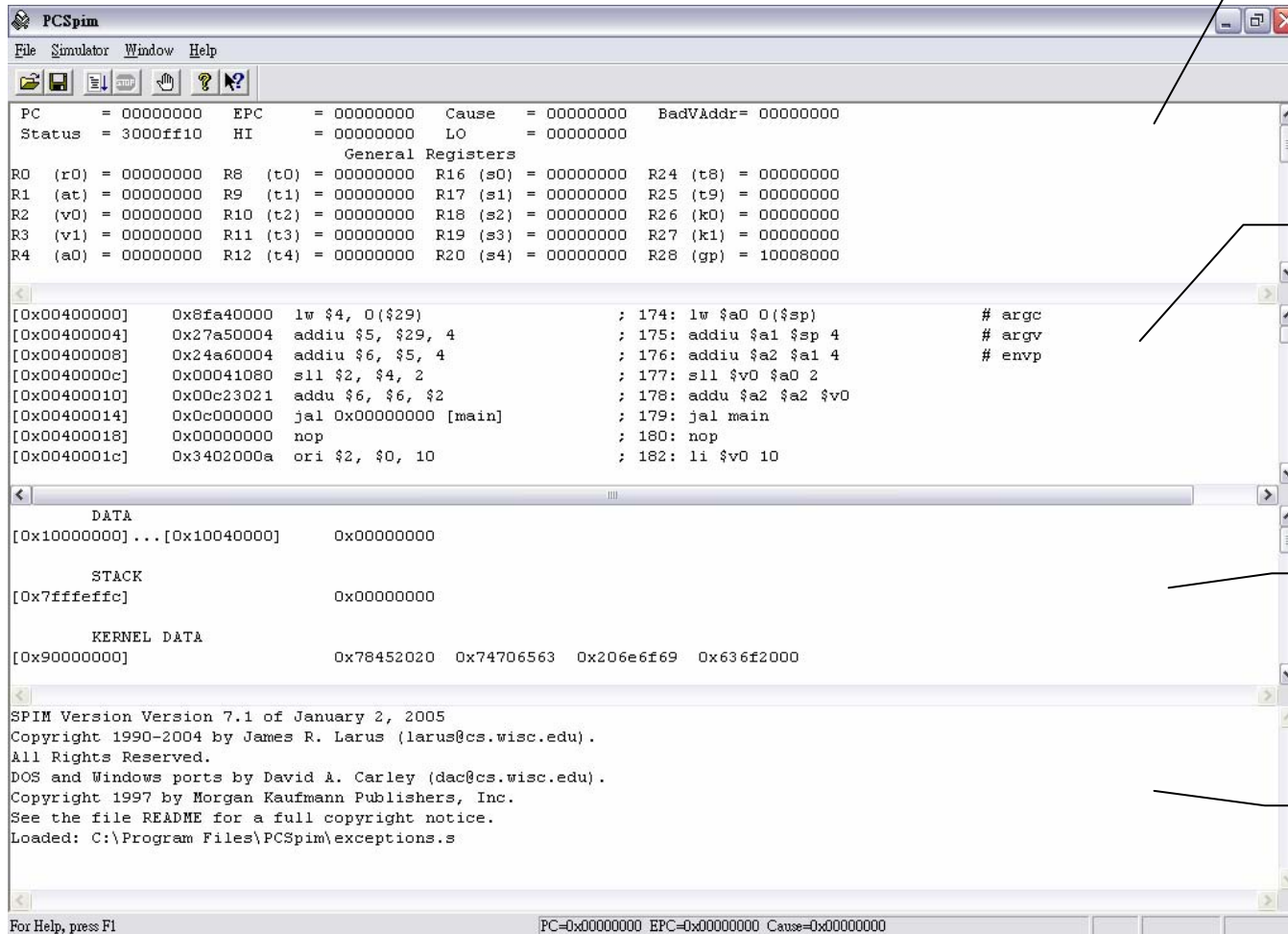
Lin-Chieh Shangkuan

r93066@csie.ntu.edu.tw

Introduction

- SPIM is a MIPS32 simulator that reads and executes assembly language program written for SPIM.
- SPIM does not execute binary program.
- The homepage of SPIM:
<http://www.cs.wisc.edu/~larus/spim.html>

Screenshot



Register
window

Text (code)
window

Data
window

Message
window

How to execute the program?

1. Prepare your assembly source code. (*.s file)
Assembling is NOT needed.
2. Load the source file in SPIM.
3. You can execute it by following 2 ways:
 - a) Add 'main' label in the first line of your source.
 - b) Start the program from specified address. (by noticing the text window to see where is your code loaded.)
4. Just run it! (go through / step by step)
5. Notice the information in register window and data window.

Example – string length

- To compute the length of a string, and put the result in the register \$t0.

- Source:

```
.data
str:
    .asciiz "Computer Architecture"
    .text
    .globl main
main:
    la $t1 str          # load the address of str
Loop:
    lb  $t3 0($t1)      # load a byte
    addi $t0 $t0 1      # increase the counter by 1
    addi $t1 $t1 1      # increase the index by 1
    bne $t3 $0 Loop     # if not '\0', back to Loop
    addi $t0 $t0 -1     # subtract the last calculating..
```

Homework (2005/03/01)

- To write a MIPS assembly program to calculate the sum of 10 integers.
- Hint:
 - how to store the 10 numbers in the data section?
 - `.word` syntax, e.g.,
`.word 1, 2, 3, 4, 5, 6, 7, 8, 9, 10`
Each word has 32-bit wide, and you can load them by using `lw` (load word) instruction.
- Due: 2005/03/09