PROJECT PHASE – 2

Members

- Chakradhar Siyyadri
- Sri Vibhav J
- Aravind Kumar P

Overview

This phase of the project aims to display pipelined stages of each instructions, stalls implementation and instructions per cycle.

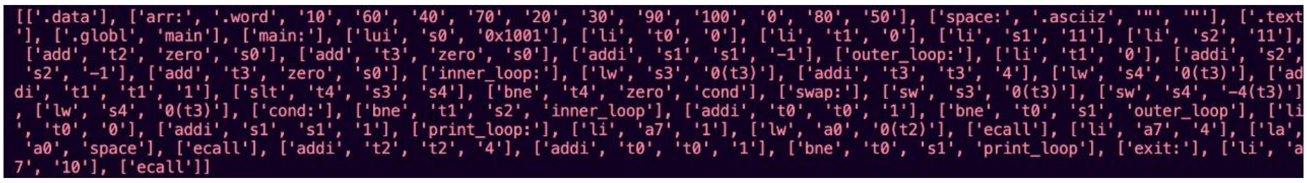
It is a fact that pipelined stages does not require the information in the data segment of the assembly code. To neglect such unnecessary information, another python file is created which skims the code off and gives lines including only the instructions.

The code is written to the part of simulation without data forwarding. Part with data forwarding is yet to be done. We have an idea on how it must be done and is under progress.

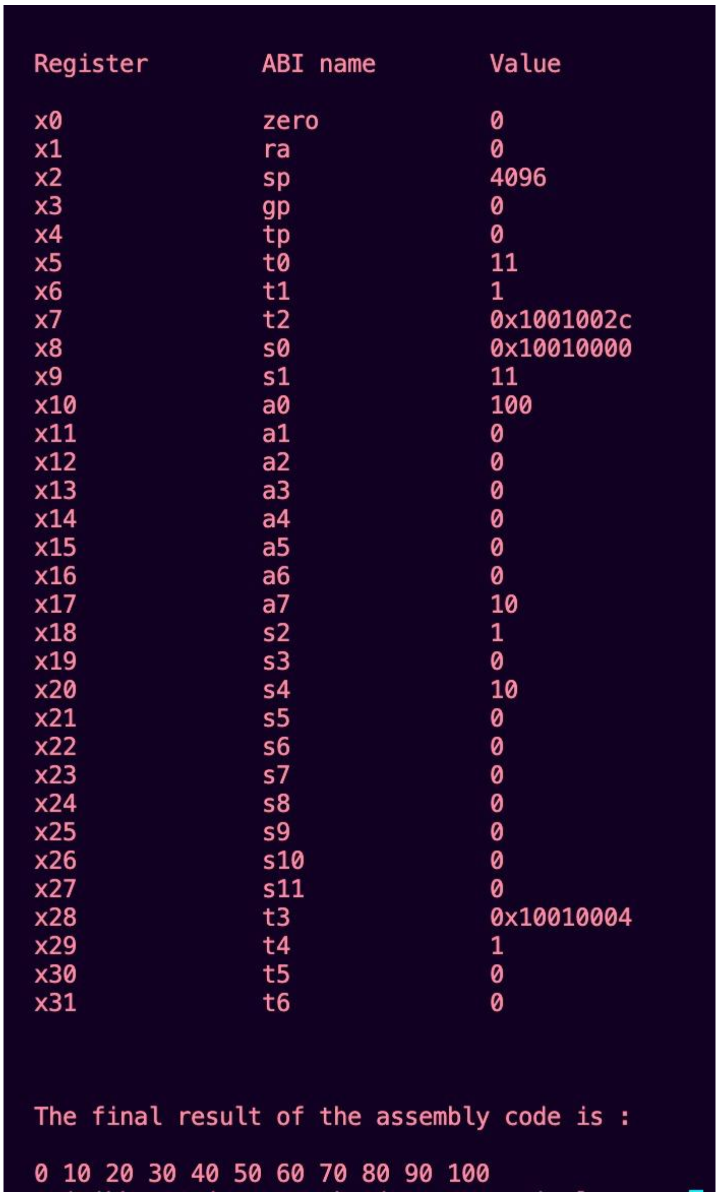
Logic

The main logic behind the program is to parse through each line, and add instruction number and clock cycle for respective stages of IF, ID/RF, EXE, MEM, WB. As a part of this process the cycle number for possible stalls are also marked. Whenever the cycle with stall is encountered the stage is incremented to the cycle immediate to the last stall.

Screenshots



```
[['.data'], ['arr:', '.word', '10', '60', '40', '70', '20', '30', '90', '100', '0', '80', '50'], ['space:', '.asciiz', '"', '"'], ['.text
'], ['.globl', 'main'], ['main:'], ['lui', 's0', '0x1001'], ['li', 't0', '0'], ['li', 't1', '0'], ['li', 's1', '11'], ['li', 's2', '11'],
 ['add', 't2', 'zero', 's0'], ['add', 't3', 'zero', 's0'], ['addi', 's1', 's1', '-1'], ['outer_loop:'], ['li', 't1', '0'], ['addi', 's2',
 's2', '-1'], ['add', 't3', 'zero', 's0'], ['inner_loop:'], ['lw', 's3', '0(t3)'], ['addi', 't3', 't3', '4'], ['lw', 's4', '0(t3)'], ['ad
di', 't1', 't1', '1'], ['slt', 't4', 's3', 's4'], ['bne', 't4', 'zero', 'cond'], ['swap:'], ['sw', 's3', '0(t3)'], ['sw', 's4', '-4(t3)']
, ['lw', 's4', '0(t3)'], ['cond:'], ['bne', 't1', 's2', 'inner_loop'], ['addi', 't0', 't0', '1'], ['bne', 't0', 's1', 'outer_loop'], ['li
', 't0', '0'], ['addi', 's1', 's1', '1'], ['print_loop:'], ['li', 'a7', '1'], ['lw', 'a0', '0(t2)'], ['ecall'], ['li', 'a7', '4'], ['la',
 'a0', 'space'], ['ecall'], ['addi', 't2', 't2', '4'], ['addi', 't0', 't0', '1'], ['bne', 't0', 's1', 'print_loop'], ['exit:'], ['li', 'a
7', '10'], ['ecall']]
```



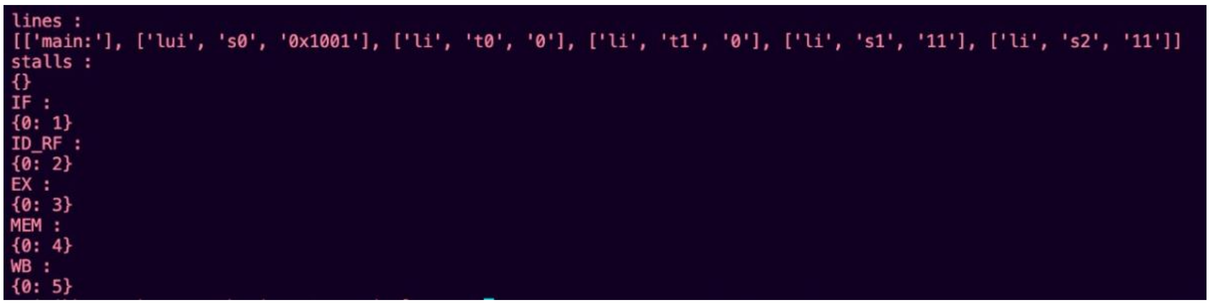
```
Register        ABI name        Value

x0              zero            0
x1              ra              0
x2              sp              4096
x3              gp              0
x4              tp              0
x5              t0              11
x6              t1              1
x7              t2              0x1001002c
x8              s0              0x10010000
x9              s1              11
x10             a0              100
x11             a1              0
x12             a2              0
x13             a3              0
x14             a4              0
x15             a5              0
x16             a6              0
x17             a7              10
x18             s2              1
x19             s3              0
x20             s4              10
x21             s5              0
x22             s6              0
x23             s7              0
x24             s8              0
x25             s9              0
x26             s10             0
x27             s11             0
x28             t3              0x10010004
x29             t4              1
x30             t5              0
x31             t6              0



The final result of the assembly code is :

0 10 20 30 40 50 60 70 80 90 100
```



```
lines :
[['main:'], ['lui', 's0', '0x1001'], ['li', 't0', '0'], ['li', 't1', '0'], ['li', 's1', '11'], ['li', 's2', '11']]
stalls :
{}
IF :
{0: 1}
ID_RF :
{0: 2}
EX :
{0: 3}
MEM :
{0: 4}
WB :
{0: 5}
```

RoadBlocks

- A clear understanding of stalls and pipelining had to be improved.
- Our team chose Python for which we need more practice and exposure to get on with.
- We fell behind the project deadlines because of tighter schedules of travelling to campus, mid-semester and Tirutsava.

Developments to be done

- A few bugs are to be fixed. Stalls and pipeling must be more detailed.
- A GUI could be developed to enhance easy understanding.
- Data forwarding must be implemented.