# EGC442 Class Notes 2/3/2023

**Baback Izadi**
Division of Engineering Programs
bai@engr.newpaltz.edu

```
/* This is a Verilog description for an 8 x 8
register file */
module regfile8x8
  (input write,
   input [2:0] wrAddr,
   input [7:0] wrData,
   input [2:0] rdAddrA,
   output reg [7:0] rdDataA,
   input [2:0] rdAddrB,
   output reg [7:0] rdDataB);
    reg [7:0]  register [0:7];
    always @(*) begin
      case (rdAddrA)
     0: rdDataA = register[0];
     1: rdDataA = register[1];
     2: rdDataA = register[2];
     3: rdDataA = register[3];
     5: rdDataA = register[5];
     6: rdDataA = register[6];
     7: rdDataA = register[7];
     default: rdDataA = 8'hXX;
      endcase
   end
always @(*) begin
      case (rdAddrB)
     0: rdDataA = register[0];
```

```
     1: rdDataA = register[1];
     2: rdDataA = register[2];
     3: rdDataA = register[3];
     5: rdDataA = register[5];
     6: rdDataA = register[6];
     7: rdDataA = register[7];
     default: rdDataA = 8'hXX;
       endcase
   end
   always @(posedge write) begin
     case (wrAddr)
       0: register[0] <= wrData;
       1: register[1] <= wrData;
       2: register[2] <= wrData;
       3: register[3] <= wrData;
       4: register[4] <= wrData;
       5: register[5] <= wrData;
       6: register[6] <= wrData;
       7: register[7] <= wrData;
    endcase // case (wrAddr)
   end // always @ (posedge write)
endmodule
```

# Instruction Count and CPI

$$\text{Clock Cycles} = \text{Instruction Count} \times \text{Cycles per Instruction}$$

$$\text{CPU Time} = \text{Instruction Count} \times \text{CPI} \times \text{Clock Cycle Time}$$

$$= \frac{\text{Instruction Count} \times \text{CPI}}{\text{Clock Rate}}$$

- Instruction Count for a program
  - Determined by program, ISA and compiler
- Average cycles per instruction
  - Determined by CPU hardware
  - If different instructions have different CPI
    - Average CPI affected by instruction mix

5. Suppose we have two implementations of the same instruction set architecture. Computer A has a clock cycle time of 250 ps and a CPI of 2.0 for some program, and computer B has a clock cycle time of 500 ps and a CPI of 1.2 for the same program.

a. How does one know that each computer executes the same number of instructions for the program?

All computers use the same number of instructions for a given program.

Both computers use the same instruction set architecture.

Both computers use the same implementation.

b. Which computer has a faster clock?

Computer A

Computer B

c. Which computer requires fewer clock cycles to execute a single instruction?

Computer A

Computer B

d. If Computer A executes 1000 instructions for the program, what is the program's CPU time on Computer A?

1000 instr * 2.0 cycle/instr * 250 ps/cycle = 500,000 ps.

1000 instr * 1.2 cycle/instr * 500 ps/cycle = 600,000 ps.

e. If Computer A executes 1000 instructions for the program, how many instructions does Computer B execute for the program?

1000

1000 * 1.2 = 1200

1000 * 2.0 = 2000

f. For a particular program, Computers A and B execute 2000 instructions. A's CPU time is 2000 * 2.0 * 250 = 1,000,000 ps. B's is 2000 * 1.2 * 500 = 1,200,000 ps. How much faster is Computer A than B?

1.2

200,000

1,200,000 / 1,000,000 = 1.2

h. Computer A is better than Computer B.

Yes

Unclear

3. Computer C's performance is 4 times as fast as the performance of computer B, which runs a given application in 28 seconds. How long will computer C take to run that application?

$\frac{28}{4} = 7$ Sec.

$t_C = \frac{t_B}{4}$ → 28

| | CPI for each instruction class | | |
|---|---|---|---|
| | A | B | C |
| CPI | 1 | 2 | 3 |

| Code sequence | Instruction counts for each instruction class | | |
|---|---|---|---|
| | A | B | C |
| 1 | 2 | 1 | 2 |
| 2 | 4 | 1 | 1 |

a. Instruction class __C__ requires the largest number of cycles per instruction.

$CC_{code1} = 2\times1 + 1\times2 + 2\times3 = 10$

$CC_{code2} = 4\times1 + 1\times2 + 1\times3 = 9$

$CPI_1 = 10/5 = 2$

b. Code sequence 2 executes __6__ instructions.

$CPI_2 = 9/6 = 1.5$

c. Code sequence 2 requires __9__ CPU clock cycles.

Code1 → 5 inst

Code2 → 6 ins

d. Assume a new code sequence 3 contains the following instruction counts for each instruction class. What is code sequence 3's CPU clock cycles?

| Code sequence | Instruction counts for each instruction class | | |
|---|---|---|---|
| | A | B | C |
| 3 | 10 X 1 | 4 X 2 | 6 X 3 |

→ 36

e) # of inst = 10 + 4 + 6 = 20

$CPI = \frac{36}{20} = 1.8$

8. For a given number of instructions, assume CPI is increased by 20%, and clock cycle time is decreased by 10%. The program execution time decreases.

$T_{exec} = \# \text{of inst} * CPI * T_{cyc}$

$\uparrow f = \frac{1}{T \downarrow}$

$\uparrow 20\%$, $10\% \downarrow$

$T_{exec\ new} = \# \text{of inst} * CPI * 1.2 * T_{cyc} * 0.9$

$= 1.08\ \# \text{of inst} * CPI * T_{cyc}$

Increases

slower system

4. Our favorite program runs in 10 seconds on computer A, which has a 2 GHz clock. We are trying to help a computer designer build a computer, B, which will run this program in 6 seconds. The designer has determined that a substantial increase in the clock rate is possible, but this increase will affect the rest of the CPU design, causing computer B to requi 1.2 times as many clock cycles as computer A for this program.

SAME ISA

a. What is the CPU clock cycles for computer A?

b. Computer B's performance is improved by reducing the ______.

$f_B = 4GHz$, 250 ps

$f_A = 2 \times 10^9 \rightarrow T_A = \frac{1}{2} \times 10^{-9}$

$T_{exec}B = Instr * CPI * T_B$

$\frac{10}{6} = \frac{1}{1.2} * \frac{.5}{T}$

$f = \frac{12}{6} \cdot \frac{1}{.5} = 4GHz$

$T_{exec}A = 10 sec = instr \times CPI_1 * .5 ns$

$T_{exec}B = 6 sec = instr \times 1.2 CPI_1 * T_B$

$\rightarrow T_B = \frac{.5 ns \times 6}{10 \times 1.2}$

$\frac{\text{Performance of B}}{\text{Performance of A}} = \frac{\text{Execution of A}}{\text{Execution of B}} = \frac{10}{6} = 1.6$

$f_B = \frac{10 \times 1.2 \times 10^9}{6 \times .5}$

$f_B = 4GHz$

7. Assume CPI and clock cycle time remain constant. Reducing the instruction count will reduce the program's execution time.

$t_{exec}$ = # of instr * CPI * clock period

yes

2 A clock rate of 1 GHz corresponds to a period of 1 nanosecond, which is 1x109 seconds.

$1 \times 10^{9}$

$T = \frac{1}{1 \times 10^{9}} = 1 \times 10^{-9}$