

Chapter-1 1.13 Exercises: 1.8, 1.14

1.8 The Pentium 4 Prescott processor, released in 2004, had a clock rate of 3.6 GHz and voltage of 1.25 V. Assume that, on average, it consumed 10 W of static power and 90 W of dynamic power.

The Core i5 Ivy Bridge, released in 2012, had a clock rate of 3.4 GHz and voltage of 0.9 V. Assume that, on average, it consumed 30 W of static power and 40 W of dynamic power.

1.8.1 For each processor find the average capacitive loads.

$$P_{\text{dynamic}} = \frac{1}{2} \times \text{CapacitiveLoad} \times \text{Voltage}^2 \times \text{FrequencySwitched}$$

$$DP = \frac{1}{2} \times C \times V^2 \times F$$

$$C = (2 \times DP) / (V^2 \times F)$$

For Pentium 4:

$$C = (2 \times 90) / (1.25^2 \times 3.6 \times 10^9) = 3.2 \times 10^{-8} \text{ F}$$

For Core i5:

$$C = (2 \times 40) / (0.9^2 \times 3.4 \times 10^9) = 2.905 \times 10^{-8} \text{ F}$$

1.8.2 Find the percentage of the total dissipated power comprised by static power and the ratio of static power to dynamic power for each technology.

For Pentium 4:

total dissipated power comprised by static power

$$10 / (10 + 90) \times 100\% = 10\%$$

ratio of static power to dynamic power

$$10 / 90 = 0.11$$

For Core i5:

total dissipated power comprised by static power

$$30 / (30 + 40) \times 100\% = 42.86\%$$

ratio of static power to dynamic power

$$30 / 40 = 0.75$$

1.8.3 If the total dissipated power is to be reduced by 10%, how much should the voltage be reduced to maintain the same leakage current? Note: power is defined as the product of voltage and current.

Total power = static power + dynamic power. Leakage current is due to static power.

$$Power_{static} = Current_{static} \times Voltage$$

$$Current_{static} = Power_{static} / Voltage$$

For Pentium 4, the total power is 100W, the leakage current is $I = 10W / 1.25V = 8A$

$$P_{new} = V_{new}I_{new} + 1/2 C_{new} V_{new}^2 f_{new} = 0.9 P_{old}$$

$$V_{new} \times 8A + 1/2 \times 3.2 \times 10^{-8} F \times V_{new}^2 \times 3.6 \times 10^9 Hz = 0.9 \times 100$$

$$8V_{new} + 57.6 V_{new}^2 = 90$$

$$V_{new} = 1.18 V$$

$$(1.18 - 1.25) / 1.18 \times 100\% = -5.9\% \quad \text{Which is a 5.9 \% reduction from 1.25 V}$$

For Core i5, the total power is 70W, the leakage current is $I = 30W / 0.9V = 33.33A$

$$33.33 \times V_{new} + 1/2 \times 2.905 \times 10^{-8} F \times V_{new}^2 \times 3.4 \times 10^9 Hz = 0.9 \times 70$$

$$33.33 V_{new} + 49.385 V_{new}^2 = 63$$

$$V_{new} = 0.82 V$$

$$(0.82 - 0.9) / 0.82 \times 100\% = -9.8\% \quad \text{Which is a 9.8 \% reduction from 0.9 V}$$

1.14 Assume a program requires the execution of 50×10^6 FP instructions, 110×10^6 INT instructions, 80×10^6 L/S instructions, and 16×10^6 branch instructions. The CPI for each type of instruction is 1, 1, 4, and 2, respectively. Assume that the processor has a 2 GHz clock rate.

FP 50×10^6 CPI-1

INT 110×10^6 CPI-1

L/S 80×10^6 CPI-4

Branch 16×10^6 CPI-2

Clock rate 2×10^9 Hz

1.14.1 By how much must we improve the CPI of FP instructions if we want the program to run two times faster?

$$Time_{old} = (50 \times 1 + 110 \times 1 + 80 \times 4 + 16 \times 2) \times 10^6 / 2 \times 10^9 = 256 \times 10^{-3}$$

$$\text{Time}_{\text{new}} = \text{Time}_{\text{old}} / 2 = 128 \times 10^{-3}$$

$$\text{CPI}_{\text{new}} = 128 \times 10^{-3} \times 2 \times 10^9 - (110 + 80 \times 4 + 16 \times 2) \times 10^6 = -4.12$$

It's impossible to improve CPI of FP instructions when we run the program two times faster because it would be negative

1.14.2 By how much must we improve the CPI of L/S instructions if we want the program to run two times faster?

$$\text{CPI}_{\text{new}} = [128 \times 10^{-3} \times 2 \times 10^9 - (50 + 110 + 16 \times 2) \times 10^6] / 80 = 0.8$$

$$0.8 / 4 = 0.2$$

Thus the new CPI of L/S instructions must be 0.8, the reduction is 80%

1.14.3 By how much is the execution time of the program improved if the CPI of INT and FP instructions is reduced by 40% and the CPI of L/S and Branch is reduced by 30%?

$$\text{FP } 50 \times 10^6 \quad \text{CPI}_{\text{new}}=0.6$$

$$\text{INT } 110 \times 10^6 \quad \text{CPI}_{\text{new}}=0.6$$

$$\text{L/S } 80 \times 10^6 \quad \text{CPI}_{\text{new}}=2.8$$

$$\text{Branch } 16 \times 10^6 \quad \text{CPI}_{\text{new}}=1.4$$

$$\text{Clock rate } 2 \times 10^9 \text{ Hz}$$

$$\text{Time}_{\text{new}} = (50 \times 0.6 + 110 \times 0.6 + 80 \times 2.8 + 16 \times 1.4) \times 10^6 / 2 \times 10^9 = 163.2 \times 10^{-3}$$

$$163.2/256 = 0.6375$$

Thus the execution time will improve 36.25%

Chapter-2 2.22 Exercises: 2.7, 2.9, 2.11, 2.16,

2.7 Show how the value 0xabcdef12 would be arranged in memory of a little-endian and a big-endian machine. Assume the data is stored starting at address 0.

Little-endian

Address	Byte
0	12
1	ef
2	cd
3	ab

Big-endian

Address	Byte
0	ab
1	cd
2	ef
3	12

2.9 Translate the following C code to MIPS. Assume that the variables f, g, h, i, and j are assigned to registers \$s0, \$s1, \$s2, \$s3, and \$s4, respectively. Assume that the base address of the arrays A and B are in registers \$s6 and \$s7, respectively. Assume that the elements of the arrays A and B are 4-byte words: $B[8] = A[i] + A[j]$;

```
sll $t0, $s3, 2

add $t0, $t0, $s6

lw $t0, 0($t0)

sll $t1, $s4, 2

add $t1, $t1, $s6

lw $t1, 0($t1)

add $t2, $t1, $t0

sw $t2, 32($s7)
```

2.11 For each MIPS instruction, show the value of the opcode (OP), source register (RS), and target register (RT) fields. For the I-type instructions, show the value of the immediate field, and for the R-type instructions, show the value of the destination register (RD) field.

```
addi $t0, $s6, 4      I-type    value of the immediate field: 4
OP: 8  RS: $s6 is 22  RT: $t0 is 8

add $t1, $s6, $0      R-type    value of the destination register field $t1 is 9
OP: 32 RS: $s6 is 22  RT: $t1 is 9

sw $t1, 0($t0)        I-type    value of the immediate field: 0
OP: 43 RS: $t0 is 8   RT: $t1 is 9

lw $t0, 0($t0)        I-type    value of the immediate field: 0
OP: 35 RS: $t0 is 8   RT: $t0 is 8

add $s0, $t1, $t0      R-type    value of the destination register field $s0 is 16
OP: 32 RS: $t1 is 9   RT: $s0 is 16
```

2.16 Provide the type, assembly language instruction, and binary representation of instruction described by the following MIPS fields: op=0, rs=3, rt=2, rd=3, shamt=0, funct=34

R-type

rs=3 \$v1 rt=2 \$v0 rd=3 \$v1 funct=34 sub

Assembly instruction is: sub \$v1 \$v1 \$v0

Binary representation of instruction:

op	rs	rt	rd	shamt	funct
000000	00011	00010	00011	00000	100010