1.2

a. Performance via Pipelining
b. Dependability via Redundancy
c. Performance via Prediction
d. Make the common case fast
e. Hierarchy of memories
f. Performance via Parallelism
g. Design for Moore's Law
h. Use Abstraction to Simplify Design

1.3 The program is compiled into an assembly language program, which is then assembled into a machine language program.

1.4a. $1024 \times 1280 = 1310720$ pixels

$1310720 \times 3 = 3932160$ bytes / frame

b. $3932160 \text{ bytes} \times (8 \text{ bit} / \text{byte}) / 100E6 \text{ bits/second}$

$= 0.31$ seconds.

1.5 P_1: 3GHz P_2: 2.5GHz P_3: 4GHz

CPI: 1.5 GPI: 1.0 GPI: 2.2

a. 时间 = CPI × 指令数 × 周期

= CPI × 指令数 / 频率

频率 × IS / CPI = 每秒指令

P_1: 3GHz / 1.5 = 2G $= 2\times10^9$

P_2: 2.5GHz / 1.0 = 2.5G $= 2.5\times10^9$

P_3: 4GHz / 2.2 = 1.81G $= 1.8\times10^9$

P_2 处理器性能最高.

b. 周期 = 时间 × 频率

cycles(P_1) $= 10\times3\times10^9 = 30\times10^9 = 3\times10^{10}$

cycles(P_2) $= 10\times2.5\times10^9 = 25\times10^9 = 2.5\times10^{10}$

cycles(P_3) $= 10\times4\times10^9 = 40\times10^9 = 4\times10^{10}$

指令数 = 周期数 / CPI

instructions (P_1) $= 3\times10^{10}/1.5 = 2\times10^{10}$

instructions (P_2) $= 2.5\times10^{10}/1.0 = 2.5\times10^{10}$

instructions (P_3) $= 4\times10^{10}/2.2 = 1.8\times10^{10}$.

C. $t = I \times CPI / f$

$0.7t = I \times 1.2CPI / f'$

$\frac{1}{0.7} = \frac{f'}{1.2f}$

$f' = \frac{1.2}{0.7} f = 1.71 f$

$f(P_1) = 1.71 f_1 = 5.14GHz$

$f(P_2) = 1.71 f_2 = 4.28GHz$

$f(P_3) = 1.71 f_3 = 6.75GHz$.

1.6

CPI / f

		A	B	C	D
$f_{P_1} = 2.5GHz$	CPI	1	2	3	3
$f_{P_2} = 3GHz$	CPI	2	2	2	2

$I = 1 \times 10^6$

a. $\overline{CPI_1} = (1\times10^6\times10\%\times1 + 1\times10^6\times20\%\times2 + 1\times10^6\times50\%\times3 + 1\times10^6\times20\%\times3)/(1\times10^6)$

$= 2.6$

$\overline{CPI_2} = (1\times10^6\times10\%\times2 + 1\times10^6\times20\%\times2 + 1\times10^6\times50\%\times2 + 1\times10^6\times20\%\times2)/(1\times10^6)$

$= 2$

6. 时间 = 指令 × 平均指令数 × 周期

周期 = 1/频率

周期数 = 指令 × CPI

$cycles\ P_1 = 1\times10^6 \times 2.6 = 2.6\times10^6$

$cycles\ P_2 = 1\times10^6 \times 2.0 = 2.0\times10^6$

1.12 P_1: 4GHz P_2: 3GHz

$CPI_1 = 0.9$ $CPI_2 = 0.75$

5×10^9 条指令 1×10^9 条指令

1.12.1 $T(P_1) = 5\times10^9 \times 0.9 / (4\times10^9) = 1.125s$

$T(P_2) = 1\times10^9 \times 0.75 / (3\times10^9) = 0.25s$

$\because T(P_1) > T(P_2)$

$\therefore$ Performance (P_1) < Performance (P_2)

而 Clock Rate (P_1) > Clock Rate (P_2), 与题设矛盾.

1.12.2

$T(P_1) = 1\times10^9 \times 0.9 / (4\times10^9) = 0.225s$

$T(P_2) = N \times 0.75 / (3\times10^9) \Rightarrow N = 9\times10^8$

1.12.3

$MIPS = Clock\ Rate \times 10^{-6} / CPI$

$MIPS(P_1) = 4\times10^9 \times 10^{-6} / 0.9 = 4.44\times10^3$

$MIPS(P_2) = 3\times10^9 \times 10^{-6} / 0.75 = 4\times10^3$

MIPS(P_1) > MIPS(P_2),

performance(P_1) < performance(P_2).

1.12.4

MFLOPS = 浮点操作的数目 / (执行时间 $\times 10^6$)

MFLOPS(P_1) = $0.4 \times 5 \times 10^9 \times 10^{-6} / 1.125 = 1.78 \times 10^3$

MFLOPS(P_2) = $0.4 \times 1 \times 10^9 \times 10^{-6} / 0.25 = 1.60 \times 10^3$

MFLOPS(P_1) > MFLOPS(P_2)

performance(P_1) < performance(P_2).

1.13

程序：250S { 70S：浮点
85S：L/S　　55S 整数
40S：分支

1.13.1　$70 \times 20\% = 14S$

总 = 250 − 14 = 236S

Reduction：5.6%

1.13.2　$250 \times 0.8 = 200S$

$T_{fp} + T_{L/S} + T_{branch} = 165S$

$T_{int} = 35S$

Reduction of INT：58.8%

1.13.3

40/250 = 16% at most

NO.