Justin Arias

Homework 1

1.3 [2] <§1.3> Describe the steps that transform a program written in a high-level language such as C into a representation that is directly executed by a computer processor.

To transform a high-level language program like C into something a computer processor can execute:

1. Compilation: The high-level code is translated by a compiler into assembly language.
2. Assembly: An assembler converts the assembly language into machine code (binary), which the computer understands.
3. Execution: The machine code is loaded into memory and directly executed by the processor.

1.4 [2] <§1.4> Assume a color display using 8 bits for each of the primary colors (red, green, blue) per pixel and a frame size of 1280 X 1024.

a. What is the minimum size in bytes of the frame buffer to store a frame?

Each pixel in a color display uses 8 bits for each of the three primary colors (red, green, and blue), for a total of 24 bits per pixel.

- Frame resolution: 1280 × 1024 = 1,310,720 pixels
- Bits per pixel: 24 bits
- Total bits for the frame: 1,310,720 pixels × 24 bits/pixel = 31,457,280 bits

Now, convert this to bytes:

- Total bytes: 31,457,280 bits ÷ 8 bits/byte = 3,932,160 bytes

Thus, the minimum size of the frame buffer is 3,932,160 bytes.

b. How long would it take, at a minimum, for the frame to be sent over a 100 Mbit/s network?

- Network speed: 100 Mbit/s (megabits per second)
- Total size of frame: 31,457,280 bits

Now, calculate the time it would take to transmit this frame:

- Time = (Total bits) ÷ (Network speed)
  = 31,457,280 bits ÷ 100,000,000 bits/second
  = 0.3146 seconds

Thus, it would take approximately 0.3146 seconds to send the frame over a 100 Mbit/s network.

1.5 [4] <§1.6> Consider three different processors P1, P2, and P3 executing the same instruction set. P1 has a 3 GHz clock rate and a CPI of 1.5. P2 has a 2.5 GHz clock rate and a CPI of 1.0. P3 has a 4.0 GHz clock rate and has a CPI of 2.2.

a. Which processor has the highest performance expressed in instructions per second?

Formula:

Instructions per second = $\frac{Clock\ Rate}{CPI}$

We will calculate the instructions per second for each processor.

P1: $\frac{3\ GHz}{1.5} = \frac{3\ x\ 10^9}{1.5} = 2\ x\ 10^9\ instructions\ per\ second$

P2: $\frac{2.5\ GHz}{1.0} = 2.5\ x\ 10^9\ instructions\ per\ second$

P3: $\frac{4.0\ GHz}{2.2} = \frac{4\ x\ 10^9}{2.2} = 1.82\ x\ 10^9\ instructions\ per\ second$

Conclusion:

P2 has the highest performance with 2.5 × 10^9 instructions per second.

b. If the processors each execute a program in 10 seconds, find the number of cycles and the number of instructions.

To find the number of cycles and instructions, we can use the following formulas:

- Number of cycles:

  Number of cycles = Clock rate (in Hz) x Execution time (in seconds)

- Number of instructions:

  Number of instructions = $\frac{Number\ of\ Cycles}{CPI}$

For P1:

- Number of Cycles:

$$3\ x\ \ 10^9 \frac{cycles}{seconds}\ x\ 10\ seconds = 30\ x\ 10^9\ cycles$$

- Number of instructions:

$$\frac{30\ x\ 10^9\ cycles}{1.5\ CPI} = 20\ x\ 10^9\ instructions$$

For P2:

- Number of Cycles:

$$2.5\ x\ \ 10^9 \frac{cycles}{seconds}\ x\ 10\ seconds = 25\ x\ 10^9\ cycles$$

- Number of instructions:

$$\frac{25\ x\ 10^9\ cycles}{1.0\ CPI} = 25\ x\ 10^9\ instructions$$

For P3:

- Number of Cycles:

$$4\ x\ \ 10^9 \frac{cycles}{seconds}\ x\ 10\ seconds = 40\ x\ 10^9\ cycles$$

- Number of instructions:

$$\frac{40\ x\ 10^9\ cycles}{2.2\ CPI} = 18.18\ x\ 10^9\ instructions$$

c. We are trying to reduce the execution time by 30% but this leads to an increase of 20% in the CPI. What clock rate should we have to get this time reduction?

Let:

- Initial execution time = T
- New execution time = 0.7T (since we want a 30% reduction)
- Let the original CPI be CPIold and the new CPI be CPInew=1.2 × CPIold
- The goal is to find the new clock rate f_{new}.

Formula:

$$T_{new=} \frac{CPI_{new}\ x\ Instruction\ count}{f_{new}}$$

We know the old execution time:

$$T\ =\ \frac{CPI_{old}\ x\ Instruction\ count}{f_{old}}$$

Using the fact that T_{new} = 0.7T, we can equate the expressions:

$$0.7\ \times\ \frac{CPI_{old}\ x\ Instruction\ count}{f_{old}} = \frac{1.2\ x\ CPI_{old}\ x\ Instruction\ count}{f_{new}}$$

The instruction counts and CPI_{old} cancel out, simplifying to:

$$0.7\ \times\ \frac{1}{f_{old}} = \frac{1.2}{f_{new}}$$

Now solve for f_{new}:

$$f_{new} = \frac{1.2}{0.7} \times f_{old} = \frac{12}{7} \times f_{old} \approx 1.71 \; x \; f_{old}$$

Thus, the new clock rate should be approximately 1.71 times the original clock rate to achieve a 30% reduction in execution time with a 20% increase in CPI.

1.6 [5] Consider the table given next, which tracks several performance indicators for Intel desktop processors since 2010.
The "Tech" column shows the minimum feature size of each processor's fabrication process. Assume that the die size has remained relatively constant and the number of transistors that comprise each processor scales at $(1/t)^2$, where t = the minimum feature size.
For each performance indicator, calculate the average rate of improvement from 2010 to 2019, as well as the number of years required to double each at that corresponding rate.

Desktop processor	Year	Tech	Max. clock speed (GHz)	Integer IPC/core	Cores	Max. DRAM Bandwidth (GB/s)	SP floating point (Gflop/s)	L3 cache (MiB)
Westmere i7-620	2010	32	3.33	4	2	17.1	107	4
Ivy Bridge i7-3770K	2013	22	3.90	6	4	25.6	250	8
Broadwell i7-6700K	2015	14	4.20	8	4	34.1	269	8
Kaby Lake i7-7700K	2017	14	4.50	8	4	38.4	288	8
Coffee Lake i7-9700K	2019	14	4.90	8	8	42.7	627	12
Imp./year		__%	__%	__%	__%	__%	__%	__%
Doubles every		__years	__years	__years	__years	__years	__years	__years

Calculate the percentage improvement per year: For each performance metric, you calculate the percentage improvement per year using the formula:

$$Improvement\ per\ year = \left(\frac{New\ value}{Old\ value}\right)^{\frac{1}{Number\ of\ years}} - 1$$

Where:

- New value is the value from the later year.
- Old value is the value from the earlier year.
- Number of years is the time difference between the two measurements.

Determine the doubling time: The number of years it takes for a metric to double can be calculated using the formula:

$$Doubling\ time\ (\ in\ years) = \frac{\log(2)}{\log(1 + Improvement\ per\ year)}$$

Example Calculations

We will use the table values for 2010 (Westmere) and 2019 (Coffee Lake) for each performance metric.

Max. Clock Speed:

- Old value (2010) = 3.33 GHz
- New value (2019) = 4.90 GHz
- Number of years = 2019 - 2010 = 9 years

Improvement per year:

$$(\frac{4.90}{3.33})^{\frac{1}{9}} - 1 = (1.471)^{0.1111} - 1 \approx 0.0435 = 4.35\%$$

Doubling time:

$$\frac{\log(2)}{\log(1 + 0.0435)} = \frac{0.3010}{0.0183} \approx 16.44\ years$$

Integer IPC/Core:

- Old value (2010) = 4
- New value (2019) = 8
- Number of years = 9

Improvement per year:

$$(\frac{8}{4})^{\frac{1}{9}} - 1 = (2)^{0.1111} - 1 \approx 0.0808 = 8.08\%$$

Doubling time:

$$\frac{\log(2)}{\log(1 + 0.0808)} = \frac{0.3010}{0.0336} \approx 8.96\ years$$

Cores:

- Old value (2010) = 2
- New value (2019) = 8
- Number of years = 9

Improvement per year:

$$(\frac{8}{2})^{\frac{1}{9}} - 1 = (4)^{0.1111} - 1 \approx 0.1544 = 15.44\%$$

Doubling time:

$$\frac{\log(2)}{\log(1 + 0.1544)} = \frac{0.3010}{0.0651} \approx 4.62\ years$$

Max. DRAM Bandwidth**:**

- Old value (2010) = 17.1 GB/s
- New value (2019) = 42.7 GB/s
- Number of years = 9

Improvement per year:

$$(\frac{42.7}{17.1})^{\frac{1}{9}} - 1 = (2.5)^{0.1111} - 1 \approx 0.1069 = 10.69\%$$

Doubling time:

$$\frac{\log(2)}{\log(1 + 0.1069)} = \frac{0.3010}{0.0422} \approx 7.13\ years$$

SP Floating Point (Gflop/s):

- Old value (2010) = 107 Gflop/s
- New value (2019) = 627 Gflop/s
- Number of years = 9

Improvement per year:

$$(\frac{627}{107})^{\frac{1}{9}} - 1 = (5.859)^{0.1111} - 1 \approx 0.1965 = 19.65\%$$

Doubling time:

$$\frac{\log(2)}{\log(1 + 0.1965)} = \frac{0.3010}{0.0789} \approx 3.82\ years$$

L3 Cache:

- Old value (2010) = 4 MiB
- New value (2019) = 12 MiB
- Number of years = 9

Improvement per year:

$$(\frac{12}{4})^{\frac{1}{9}} - 1 = (3)^{0.1111} - 1 \approx 0.1292 = 12.92\%$$

Doubling time:

$$\frac{\log(2)}{\log(1 + 0.1292)} = \frac{0.3010}{0.0546} \approx 5.51\ years$$

Final Results:

Metric	Improvement per year	Doubling time (years)
Max. Clock Speed	4.35%	16.44
Integer IPC/Core	8.08%	8.96
Cores	15.44%	4.62
Max. DRAM Bandwidth	10.69%	7.13
SP Floating Point	19.65%	3.82
L3 Cache	12.92%	5.51

These values represent the average improvement rate and the number of years it would take for each metric to double at that rate.

1.7 [20] <§1.6> Consider two different implementations of the same instruction set architecture. The instructions can be divided into four classes according to their CPI (class A, B, C, and D). P1 with a clock rate of 2.5 GHz and CPIs of 1, 2, 3, and 3, and P2 with a clock rate of 3 GHz and CPIs of 2, 2, 2, and 2.

Given a program with a dynamic instruction count of 1.0E6 instructions divided into classes as follows: 10% class A, 20% class B, 50% class C, and 20% class D, which is faster: P1 or P2?

a. **What is the global CPI for each implementation?**

To find the global CPI for each implementation, we use the formula:

$$\text{Global CPI} = (\%\text{Class A}) \times \text{CPI}_A + (\%\text{Class B}) \times \text{CPI}_B + (\%\text{Class C}) \times \text{CPI}_C + (\%\text{Class D}) \times \text{CPI}_D$$

For P1:

$$\text{Global CPI for P1} = (0.10 \times 1) + (0.20 \times 2) + (0.50 \times 3) + (0.20 \times 3)$$

$$= 0.10 + 0.40 + 1.50 + 0.60 = 2.60$$

For P2:

$$\text{Global CPI for P2} = (0.10 \times 2) + (0.20 \times 2) + (0.50 \times 2) + (0.20 \times 2)$$

$$= 0.20 + 0.40 + 1.00 + 0.40 = 2.00$$

b. Find the clock cycles required in both cases.

The number of clock cycles is calculated using the formula:

$$\text{Clock Cycles} = \text{Instruction count} \times \text{Global CPI}$$

- Instruction count = 1.0×10^6 (1,000,000 instructions)

For P1:

$$\text{Clock Cycles for P1} = 1.0 \times 10^6 \times 2.60 = 2.60 \times 10^6 \text{ clock cycles}$$

For P2:

$$\text{Clock Cycles for P2} = 1.0 \times 10^6 \times 2.00 = 2.00 \times 10^6 \text{ clock cycles}$$

Conclusion:

P1 has a global CPI of 2.60 and requires 2.60 million clock cycles.

P2 has a global CPI of 2.00 and requires 2.00 million clock cycles.

Which is faster?

To determine which processor is faster, we use the execution time formula:

$$\text{Execution time} = \frac{\text{Clock Cycles}}{\text{Clock Rate}}$$

For P1:

$$\text{Execution time for P1} = \frac{2.60 \times 10^6}{2.5 \times 10^9} = 1.04 \times 10^{-3} \text{ seconds} = 1.04 \text{ milliseconds}$$

For P2:

$$\text{Execution time for P2} = \frac{2.00 \times 10^6}{3.0 \times 10^9} = 0.67 \times 10^{-3} \text{ seconds} = 0.67 \text{ milliseconds}$$

Final Answer:

- **P2** is faster, as it completes the program in **0.67 milliseconds**, while **P1** takes **1.04 milliseconds**.

1.11 Assume a 15 cm diameter wafer has a cost of 12, contains 84 dies, and has 0.020 defects/cm2. Assume a 20 cm diameter wafer has a cost of 15, contains 100 dies, and has 0.031 defects/cm2.

1.11.1 [10] <§1.5> Find the yield for both wafers.

The formula for **yield (Y)** is:

$$Y = \frac{1}{(1 + \text{Defects per cm}^2 \times \text{Die Area})^N}$$

Where:

- Defects per cm² is given.
- Die Area is estimated using the wafer area and the number of dies.
- N is typically between 2 and 3. For this question, we will assume N=3, which is often used in these types of calculations.

For the 15 cm wafer:

- Wafer diameter: 15 cm
- Wafer area: $\pi \times (7.5)^2 = 176.71\ cm^2$
- Number of dies: 84
- Defects per cm²: 0.020
- Die area: $\frac{\text{Wafer Area}}{\text{Number of Dies}} = \frac{176.71}{84} = 2.10 \text{ cm}^2$

Now calculate the yield for the 15 cm wafer:

$$Y = \frac{1}{(1 + 0.020 \times 2.10)^3} = \frac{1}{(1 + 0.042)^3} = \frac{1}{1.042^3} = \frac{1}{1.13} \approx 0.884$$

For the 20 cm wafer:

- Wafer diameter: 20 cm
- Wafer area: $\pi \times (10)^2 = 314.16\ cm^2$
- Number of dies: 100
- Defects per cm²: 0.031
- Die area: $\frac{\text{Wafer Area}}{\text{Number of Dies}} = \frac{314.16}{100} = 3.14\ \text{cm}^2$

Now calculate the yield for the 20 cm wafer:

$$Y = \frac{1}{(1 + 0.031 \times 3.14)^3} = \frac{1}{(1 + 0.097)^3} = \frac{1}{1.097^3} = \frac{1}{1.295} \approx 0.772$$

1.11.2 [5] <§1.5> Find the cost per die for both wafers.

The formula for **cost per die** is:

$$\text{Cost per die} = \frac{\text{Cost of wafer}}{\text{Number of good dies}} = \frac{\text{Cost of wafer}}{\text{Number of dies} \times \text{Yield}}$$

For the 15 cm wafer:

- Cost of wafer: 12
- Number of dies: 84
- Yield: 0.884

$$\text{Cost per die for 15 cm wafer} = \frac{12}{84 \times 0.884} = \frac{12}{74.256} \approx 0.162 \text{ units}$$

For the 20 cm wafer:

- Cost of wafer: 15
- Number of dies: 100
- Yield: 0.772

$$\text{Cost per die for 20 cm wafer} = \frac{15}{100 \times 0.772} = \frac{15}{77.2} \approx 0.194 \text{ units}$$

1.11.3 [5] <§1.5> If the number of dies per wafer is increased by 10% and the defects per area unit increases by 15%, find the die area and yield.

For the 15 cm wafer:

- Increase in number of dies: $84 \times 1.10 = 92.4 \approx 92$ dies
- Increase in defects per cm^2: $0.020 \times 1.15 = 0.023$ defects/ cm^2
- Wafer area: 176.71 cm^2 (from previous calculation)

Now calculate the new die area:

$$\text{Die area} = \frac{\text{Wafer Area}}{\text{Number of Dies}} = \frac{176.71}{92} = 1.92 \text{ cm}^2$$

Now calculate the new yield:

$$Y = \frac{1}{(1 + 0.023 \times 1.92)^3} = \frac{1}{(1 + 0.0442)^3} = \frac{1}{1.0442^3} = \frac{1}{1.139} \approx 0.878$$

1.11.4 [5] <§1.5> Assume a fabrication process improves the yield from 0.92 to 0.95. Find the defects per area unit for each version of the technology given a die area of 200 mm2.

The formula for **yield** based on defects and die area is:

$$Y = \frac{1}{(1 + \text{Defects per cm}^2 \times \text{Die Area})^3}$$

Rearranging the formula to solve for defects per cm^2:

$$\text{Defects per cm}^2 = \frac{(1 - Y^{1/3})}{\text{Die Area}}$$

For the original yield (0.92):

- Die Area: 200 mm^2 = 2 cm^2

$$\text{Defects per cm}^2 = \frac{1 - 0.92^{1/3}}{2} = \frac{1 - 0.965}{2} = \frac{0.035}{2} = 0.0175 \text{ defects/cm}^2$$

For the improved yield (0.95):

Die Area: 200 mm^2 = 2 cm^2

$$\text{Defects per cm}^2 = \frac{1 - 0.95^{1/3}}{2} = \frac{1 - 0.983}{2} = \frac{0.017}{2} = 0.0085 \text{ defects/cm}^2$$

1.12 The results of the SPEC CPU2006 bzip2 benchmark running on an AMD Barcelona has an instruction count of 2.389E12, an execution time of 750 s, and a reference time of 9650 s.

1.12.1 [5] <§§1.6, 1.9> Find the CPI if the clock cycle time is 0.333 ns.

We use the formula for CPU execution time:

$$\text{Execution time} = \frac{\text{Instruction count} \times \text{CPI}}{\text{Clock rate}}$$

Rearranging the formula to solve for CPI:

$$\text{CPI} = \frac{\text{Execution time} \times \text{Clock rate}}{\text{Instruction count}}$$

Given:

- Instruction count = 2.389×10^{12}
- Execution time = 750 seconds
- Clock cycle time = 0.333 ns = 0.333×10^{-9} s

Clock rate $f = \frac{1}{\text{Clock cycle time}} = \frac{1}{0.333 \times 10^{-9}} = 3.003 \times 10^{9}$ Hz

Now, plug in the values:

$$\text{CPI} = \frac{750 \times 3.003 \times 10^{9}}{2.389 \times 10^{12}} = \frac{2.25225 \times 10^{12}}{2.389 \times 10^{12}} \approx 0.942$$

So, **CPI ≈ 0.942**.

1.12.2 [5] <§1.9> Find the SPECratio.

The SPECratio is given by:

$$\text{SPECratio} = \frac{\text{Reference time}}{\text{Execution time}}$$

Given:

- Reference time = 9650 seconds
- Execution time = 750 seconds

$$\text{SPECratio} = \frac{9650}{750} \approx 12.87$$

So, **SPECratio ≈ 12.87**.

1.12.3 [5] <§§1.6, 1.9> Find the increase in CPU time if the number of instructions of the benchmark is increased by 10% without affecting the CPI.

The CPU execution time is directly proportional to the number of instructions if CPI and clock rate remain constant.

New instruction count:

$$\text{New instruction count} = 2.389 \times 10^{12} \times 1.10 = 2.6279 \times 10^{12}$$

Now, calculate the new execution time:

$$\text{New execution time} = \frac{\text{New instruction count} \times \text{CPI}}{\text{Clock rate}} = \frac{2.6279 \times 10^{12} \times 0.942}{3.003 \times 10^{9}}$$

$$\text{New execution time} = \frac{2.4753 \times 10^{12}}{3.003 \times 10^{9}} = 824.42 \text{ seconds}$$

So, the new execution time is **824.42 seconds**, which means the CPU time has **increased by approximately 74.42 seconds.**

1.12.4 [5] <§§1.6, 1.9> Find the increase in CPU time if the number of instructions of the benchmark is increased by 10% and the CPI is increased by 5%.

New instruction count:

$$\text{New instruction count} = 2.389 \times 10^{12} \times 1.10 = 2.6279 \times 10^{12}$$

New CPI:

$$\text{New CPI} = 0.942 \times 1.05 = 0.9891$$

Now, calculate the new execution time:

$$\text{New execution time} = \frac{2.6279 \times 10^{12} \times 0.9891}{3.003 \times 10^{9}}$$

$$\text{New execution time} = \frac{2.597 \times 10^{12}}{3.003 \times 10^{9}} = 864.75 \text{ seconds}$$

So, the new execution time is **864.75 seconds**, which means the CPU time has **increased by 114.75 seconds.**

1.12.5 [5] <§§1.6, 1.9> Find the change in the SPECratio for this change.

The new SPECratio is calculated as:

$$\text{New SPECratio} = \frac{\text{Reference time}}{\text{New execution time}} = \frac{9650}{864.75} \approx 11.16$$

So, the new SPECratio is **11.16**. The SPECratio has **decreased** from 12.87 to 11.16 due to the increase in instructions and CPI.

1.12.6 [10] <§1.6> Suppose that we are developing a new version of the AMD Barcelona processor with a 4 GHz clock rate. We have added some additional instructions to the instruction set in such a way that the number of instructions has been reduced by 15%. The execution time is reduced to 700 s and the new SPECratio is 13.7. Find the new CPI.

Given:

- New clock rate = 4 GHz = 4×10^9 Hz
- Reduced instruction count = $2.389 \times 10^{12} \times 0.85 = 2.03065 \times 10^{12}$
- Execution time = 700 seconds
- SPECratio = 13.7

Using the formula for CPI:

$$\text{CPI} = \frac{\text{Execution time} \times \text{Clock rate}}{\text{Instruction count}} = \frac{700 \times 4 \times 10^9}{2.03065 \times 10^{12}}$$

$$\text{CPI} = \frac{2.8 \times 10^{12}}{2.03065 \times 10^{12}} \approx 1.379$$

So, the new CPI is **1.379**.

1.12.7 [10] <§1.6> This CPI value is larger than obtained in 1.11.1 as the clock rate was increased from 3 GHz to 4 GHz. Determine whether the increase in the CPI is similar to that of the clock rate. If they are dissimilar, why?

The CPI in 1.12.1 was 0.942 for a 3 GHz clock, and now it is 1.379 for a 4 GHz clock. The CPI increased by:

$$\text{CPI increase} = \frac{1.379}{0.942} \approx 1.46$$

Since the clock rate increased from 3 GHz to 4 GHz:

$$\text{Clock rate increase} = \frac{4}{3} \approx 1.33$$

The increase in CPI (1.46) is slightly larger than the increase in clock rate (1.33). This could be due to the fact that the additional instructions added in the newer processor design may have more complex execution, resulting in a higher CPI.

1.12.8 [5] <§1.6> By how much has the CPU time been reduced?

Original execution time = 750 seconds

New execution time = 700 seconds

$$\text{Reduction in CPU time} = 750 - 700 = 50 \text{ seconds}$$

So, the CPU time has been reduced by **50 seconds.**

1.12.9 [10] <§1.6> For a second benchmark, libquantum, assume an execution time of 960 ns, CPI of 1.61, and clock rate of 3 GHz. If the execution time is reduced by an additional 10% without affecting to the CPI and with a clock rate of 4 GHz, determine the number of instructions.

Given:

- Execution time = 960 ns = 960×10^{-9} seconds
- CPI = 1.61
- Clock rate = 3 GHz = 3×10^{9} Hz

We can use the formula:

$$\text{Instruction count} = \frac{\text{Execution time} \times \text{Clock rate}}{\text{CPI}} = \frac{960 \times 10^{-9} \times 3 \times 10^{9}}{1.61}$$

$$\text{Instruction count} = \frac{2.88 \times 10^{3}}{1.61} \approx 1788 \text{ instructions}$$

So, the number of instructions is **1788**.

1.12.10 [10] <§1.6> Determine the clock rate required to give a further 10% reduction in CPU time while maintaining the number of instructions and with the CPI unchanged.

A 10% reduction in CPU time means:

$$\text{New execution time} = 960 \times 0.90 = 864 \text{ ns}$$

CPI and instruction count remain the same. Using the formula for clock rate:

$$\text{Clock rate} = \frac{\text{Instruction count} \times \text{CPI}}{\text{New execution time}}$$

$$\text{Clock rate} = \frac{1788 \times 1.61}{864 \times 10^{-9}} \approx 3.33 \times 10^{9} \text{ Hz} = 3.33 \text{ GHz}$$

So, the required clock rate is **3.33 GHz**.

1.12.11 [10] <§1.6> Determine the clock rate if the CPI is reduced by 15% and the CPU time by 20% while the number of instructions is unchanged.

Given:

- CPI is reduced by 15%: $\text{New CPI} = 1.61 \times 0.85 = 1.3685$
- CPU time is reduced by 20%: $\text{New CPU time} = 960 \times 0.80 = 768 \text{ ns}$

Now, calculate the new clock rate:

$$\text{Clock rate} = \frac{1788 \times 1.3685}{768 \times 10^{-9}} \approx 3.$$