### Week 01:

1. What is the main difference between computer architecture and computer organization?

Computer architecture: what a computer does

Computer organization: how a computer does

1. What is microarchitecture?

Microarchitecture: organization

1. What does the term CPU stand for?

CPU: Central Process Unit

1. What does the term SRAM stand for?

SRAM: Static RAM

1. What does the term DRAM stand for?

DRAM: Dynamic RAM

6. What is the primary difference between computer registers and main computer memory?

Computer registers are inside of CPU while computer memory is outside of CPU, so registers processes have lower latency.

7. What is the width of a clock pulse if the clock frequency = 100KHz?

1/100KHz = 10^-5s

8. What is the width of a clock pulse if the clock frequency = 5KHz?

1/5KHz = 2\*10^-4s

9. Can a clever system programmer differentiate between the encoded data object and encoded program instruction from their binary bit patterns?

No. No one could tell data object from encoded program instructions from their binary bit patterns.

10. Define and differentiate between constants, variables, and pointers.

Constants: values which would not change in the program execution

Variables: things that store values and the value could be changed in the execution of the program.

Pointer: a variable whose value is an address

11. In register transfer language (RTL) notation, what does [X] ← [Y] + Z mean?

[X] ← [Y]+Z: the value stores in position x is assigned as the value stores in position y plus value z

12. In register transfer language (RTL) notation, what does [X] ← Y + Z mean?

[X] ← Y+Z: the value stores in position X is assigned as the sum of Y and Z

13. In register transfer language (RTL) notation, what does [X] = Y + Z mean?

[X] = Y+Z: the value stores in position X is initialized as the sum of Y and Z

14. Which is faster, cache memory or registers?

Registers are faster than Cache memory

15. Which is faster, cache memory or dynamic memory?

Cache memory is faster than Dynamic memory

16. Can a computer system have more than one bus?

Yes. A computer could have more than one bus.

17. Define bus width.

Bus width: the number of parallel data paths

18. Define bus bandwidth.

Bus bandwidth: a measure of the rate at which information can be transported across the bus, which is expressed in either bytes per second or bits per second.

19. Define bus latency.

Bus latency: the waiting period between a data transfer request and the actual data transmission completed.

### Week 02:

**Part 1:**

1. What is the minimum number of addressing bits that needed to access 3 K different values?

12

1. What is the minimum number of addressing bits that needed to access 3 Mega different values?

22

1. What is the minimum number of addressing bits that needed to access 5 Giga different values?

33

1. Pick a decimal number.

114

Convert this number to binary.

1110010

Convert back the generated binary number to a decimal number.

114

You need to get the same value that you started with.

1. Pick a decimal number.

514

Convert this number to hexadecimal.

202

Convert back the generated hexadecimal number to a decimal number.

514

You need to get the same value that you started with.

1. Pick a decimal number.

1919

Convert this number to octal.

3577

Convert back the generated octal number to a decimal number.

1919

You need to get the same value that you started with.

1. Pick a hexadecimal number.

810

Convert this number to octal.

4020

Convert back the generated octal number to a hexadecimal number.

810

You need to get the same value that you started with.

1. Pick an octal number.

114514

Convert this number to hexadecimal.

994C

Convert back the generated hexadecimal number to an octal number.

114514

You need to get the same value that you started with.

**Part 2:**

1. Pick a decimal number between -128 and 127.

114

Convert this number into the 8-bit sign-and-magnitude form.

01010010

Convert the number back from the 8-bit sign-and-magnitude to decimal to verify.

114

1. Pick a decimal number between -128 and 127.

51

Convert this number into the excess-127 form.

-76

Convert the number back from the 8-bit excess-127 to decimal to verify.

51

1. Pick two decimal numbers between -128 and 127.

-41

-91

Convert these two numbers into 8-bit 2’s complement form.

11010111

10100111

Add these two Converted numbers.

01111110

Check if an overflow occurred or not.

Yes

Convert the result back from the 2’s complement system to decimal to verify.

126

4. Pick two decimal numbers between -128 and 127.

Convert these two numbers into 8-bit 2’s complement form.

-98

-10

Subtract the 2nd Converted number from the first Converted number.

Check if an overflow occurred or not.

Convert the result back from the 2’s complement system to decimal to verify.

5. Pick a decimal number between -16 and 15.

Convert this number into 2’s complement form.

Perform arithmetic shift-left to the Converted number three times.

Convert the number back from the 8-bit 2’s complement to decimal to verify.

6. Pick a decimal number between 16 and 127.

Convert this number into 2’s complement form.

Perform arithmetic shift-left to the Converted number three times.

Convert the number back from the 8-bit 2’s complement to decimal to verify.

7. Pick a decimal number between -128 and -17.

Convert this number into 2’s complement form.

Perform arithmetic shift-left to the Converted number three times.

Convert the number back from the 8-bit 2’s complement to decimal to verify.

8. Pick a decimal number between -128 and 127.

Convert this number into 2’s complement form.

Perform arithmetic shift-right to the Converted number three times.

Convert the number back from the 8-bit 2’s complement to decimal to verify.

### Week 03:

1. Pick an unsigned decimal number that has at least three digits after the decimal point.

114.514

Convert this number into binary form.

1110010.1000

Round this converted number to two binary digits after the binary point using

1. round towards zero.

1110010.10

1. round towards + infinity

1110011.11

1. round to the nearest.

1110010.10

Convert the rounded numbers back to decimal.

114.5

Calculate the error in each case.

Compare the results in these three cases.

1. Pick an unsigned decimal number with at least three digits after the decimal point and three digits before the decimal point.

Convert this number into binary form.

Normalize this converted number.

Convert the result back to decimal to verify.

1. In a single-precision IEEE-754 floating-point format, what does it means to have the biased exponent = 0?

The true exponent is -127

1. In a single-precision IEEE-754 floating-point format, what does it means to have the biased exponent =255?

The number is underflow

1. In a single-precision IEEE-754 floating-point format, when is the excess-127 code used?

When the number is not underflow.

1. In a single-precision IEEE-754 floating-point format, when is the excess-126 code used?

When the number is underflow.

1. In a single-precision IEEE-754 floating-point format, how many zeros can be represented?

2

1. In a single-precision, IEEE-754 floating-point format, how many unique +NaN numbers can be represented?

1

1. In a single-precision, IEEE-754 floating-point format, how many unique -NaN numbers can be represented?

1

1. In a single-precision, IEEE-754 floating-point format, how many unique NaN numbers can be

2

represented?

1. In a single-precision IEEE-754 floating-point format, what is the largest normalized absolute number that can be represented?

3.403 \* 1038

12. In a single-precision IEEE-754 floating-point format, what is the smallest normalized absolute number that can be represented?

-3.403 \* 1038

13. Pick a decimal number.

114514

Convert this number into the single-precision IEEE-754 Floating-point format.

Convert the number back to decimal to verify.

### Week 04:

1. In a 5‐input AND gate, when will the output be one?

IFF all inputs are 1.

1. In a 5‐ input OR gate, when will the output be one?

If there is at least one input is 1.

1. In a 5‐ input XOR gate, when will the output be one?

When there are odd numbers of input is/are 1.

1. In a 5‐ input NAND gate, when will the output be one?

When there is at least one input is 0.

1. In a 5‐ input NOR gate, when will the output be one?

When all inputs are 0.

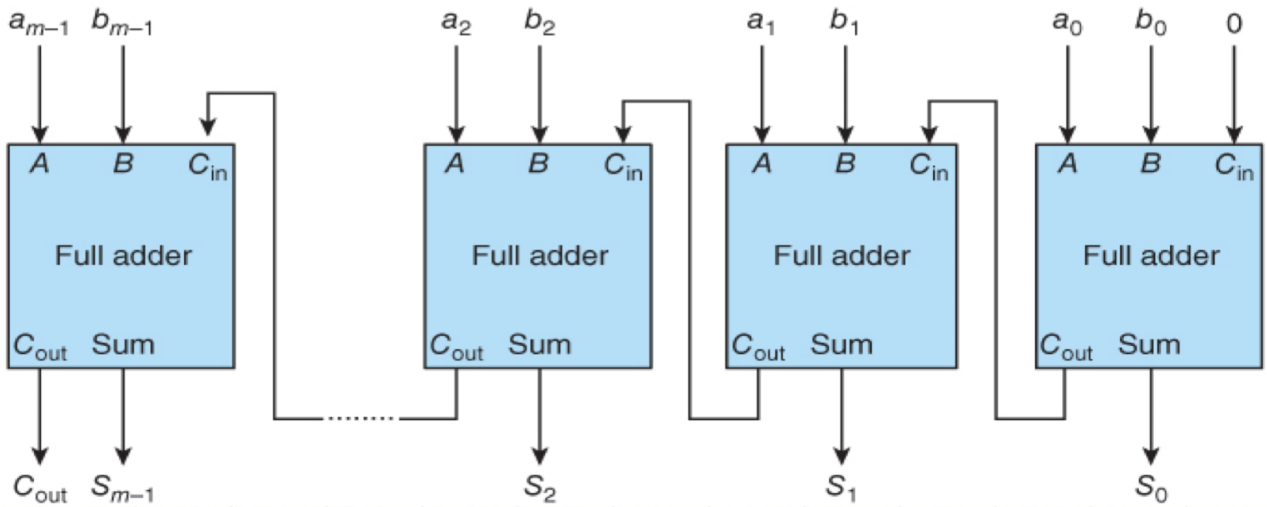
1. What is the Boolean expression that describes the sum in a single‐bit full adder?

SUM = (A XOR B) XOR Carryin = (A ⊕ B) ⊕ Carryin

1. What is the Boolean expression that describes the carry in a single‐bit full adder?

CARRY-OUT = A AND B OR Carryin(A XOR B) = A·B + Carryin(A ⊕ B)

1. Draw the block diagram of an N‐bit ripple‐through adder/subtractor.



1. What is the definition of the decoder circuit?

A decoder is combinational logic circuit that converts binary information from the n-bits coded input to a maximum of 2^n unique outputs.

1. What is the definition of the Multiplexer circuit?

A multiplexer is combinational logic circuit that has up to 2^n binary input lines and n select lines, where the n select-lines are used to forward one of the input values to the output line.

### Week 05:

### Week 06:

### Week 07:

### Week 08:

### Week 09:

### Week 0A:

### Week 0B:

### Week 0C:

### Week 0D: