(1) (20 points)

The W&W Technology has invented an advanced 64M-bit embedded DRAM core (the WW6416), which has a 16-bit data buffer (i.e. this is a 4M x 16-bit DRAM core). The WW6416 has an amazing 4K x 16K memory-cell array, and the following control inputs: *RAS, CAS, WE,* and *OE*.

(a) Specify the sizes (in number of bits) of the data buffer, row address buffer, and column address buffer of the WW6416.

(b) Use the WW6416 as the basic component and design an 8M x 64-bit RAM. Draw its block diagram.

(2) (10 points)

Define and briefly explain the following terms: (a) cache memory, (b) flash memory, (c) bus protocol, (d) clock skew, (e) latency.

(3) (20 points)

A full adder has three inputs: the two input number bits x_i, y_i, and a carry in from previous stage c_i, and two outputs: the sum bit s_i and carry out c_{i+1}. Write down the logic expression of s_i and c_{i+1}. Design a 4-bit carry-lookahead adder using the full adder and appropriate logic gates for generation and propagation signals.

(a) Using the 4-bit carry-lookahead adder as building blocks, design a 16-bit adder with carry lookahead technique applied between the 4-bit adder blocks.

(b) Assume that the propagation delay of each logic gate (AND, OR, NOT) is τ, what is the total propagation delay of the 16-bit adder in (b)?

(4) (15 points)

If you have two 4-bit adders and some basic logic gates like AND, OR, XOR, etc., design an eight-bit binary add-subtract network with overflow signal indicating an overflow condition.

(5) (15 points)

A byte-addressable computer has a small data cache capable of holding eight 32-bit words. Each cache block consists of one 32-bit word. When a given program is executed, the processor reads data from the following sequence of hex addresses:

200, 204, 208, 20C, 2F4, 2F0, 200, 204, 218, 21C, 24C, 2F4

This pattern is repeated four times. Show the contents of the cache at the end of each pass through this loop if a direct-mapped cache is used. Assume that the cache is initially empty.

(6) (20 points)

Compare 0- and 1-address machines by writing programs to compute

$$X = (A + B * C) / (D - E * F)$$

for each of the two machines. The instructions available for use are as follows:

0 Address	1 Address
PUSH M	LOAD M
POP M	STORE M
ADD	ADD M
SUB	SUB M
MUL	MUL M
DIV	DIV M

M is a 16-bit memory address. The 0-address machine uses a stack and the 1-address machine uses an accumulator. Assuming 8-bit opcodes and instruction lengths that are multiples of 4 bits, how many bits does each machine need to compute X?