HW5 (pre-exam optional) – chapter 8

8.1, 8.8, 8.9 (next page)

1. *The following memories are specified by the number of words times the number of bits per word. How many address lines and input–output data lines are needed in each case? (a) 48K × 8, (b) 512K × 32, (c) 64M × 64, and (d) 2G × 1.

2. Word number $(835)_{10}$ in the memory shown in Figure 2 contains the binary equivalent of $(15{,}103)_{10}$. List the 10-bit address and the 16-bit memory contents of the word.

3. *A 64K × 16 RAM chip uses coincident decoding by splitting the internal decoder into row select and column select. (a) Assuming that the RAM cell array is square, what is the size of each decoder, and how many AND gates are required for decoding an address? (b) Determine the row and column selection lines that are enabled when the input address is the binary equivalent of $(32000)_{10}$.

4. Assume that the largest decoder that can be used in an $m \times 1$ RAM chip has 14 address inputs and that coincident decoding is employed. In order to construct RAM chips that contain more one-bit words than m, multiple RAM cell arrays, each with decoders and read/write circuits, are included in the chip.
   (a) With the decoder restrictions given, how many RAM cell arrays are required to construct a 2G × 1 RAM chip?
   (b) Show the decoder required to select from among the different RAM arrays in the chip and its connections to address bits and cell array select (CS) bits.

5. A DRAM has 15 address pins and its row address is 1 bit longer than its column address. How many addresses, total, does the DRAM have?

6. A 1 Gb DRAM uses 4-bit data and has equal-length row and column addresses. How many address pins does the DRAM have?

7. A DRAM has a refresh interval of 128 ms and has 4096 rows. What is the interval between refreshes for distributed refresh? What is the total time required out of the 128 ms for a refresh of the entire DRAM? What is the minimum number of address pins on the DRAM?

8. ***(a)** How many 128K × 16 RAM chips are needed to provide a memory capacity of 2 MB?
   (b) How many address lines are required to access 2 MB? How many of these lines are connected to the address inputs of all chips?
   (c) How many lines must be decoded to produce the chip select inputs? Specify the size of the decoder.

9. Using the 64K × 8 RAM chip in Figure 9 plus a decoder, construct the block diagram for a 512K × 16 RAM.

10. Explain how SDRAM takes advantage of the two-dimensional storage array to provide a high data access rate.

11. Explain how a DDRAM achieves a data rate that is a factor of two higher than a comparable SDRAM.

HW5 (pre-exam optional) – chapter 9

9.5, 9.8, 9.9, 9.10, 9.11, 9.15, 9.17

1. A datapath similar to the one in Figure 1 has 64 registers. How many selection lines are needed for each set of multiplexers and for the decoder?

2. *Given an 8-bit ALU with outputs F_7 through F_0 and available carries C_8 and C_7, show the logic circuit for generating the signals for the four status bits N (sign), Z (zero), V (overflow), and C (carry).

3. *Design an arithmetic circuit with two selection variables S_1 and S_0 and two n-bit data inputs A and B. The circuit generates the following eight arithmetic operations in conjunction with carry C_{in}:

S_1	S_0	$C_{in} = 0$	$C_{in} = 1$
0	0	$F = A + B$ (add)	$F = A + \overline{B} + 1$ (subtract A – B)
0	1	$F = \overline{A} + B$	$F = \overline{A} + B + 1$ (subtract B – A)
1	0	$F = A - 1$ (decrement)	$F = A + 1$ (increment)
1	1	$F = \overline{A}$ (1s complement)	$F = \overline{A} + 1$ (2s complement)

Draw the logic diagram for the two least significant bits of the arithmetic circuit.

4. 1 *Design a 4-bit arithmetic circuit, with two selection variables S_1 and S_0, that generates the arithmetic operations in the following table. Draw the logic diagram for a typical single-bit stage and the LSB stage.

$S_1 S_0$	$C_{in} = 0$	$C_{in} = 1$
0 0	$F = A + B$ (add)	$F = A + B + 1$
0 1	$F = A$ (transfer)	$F = A + 1$ (increment)
1 0	$F = \overline{B}$ (complement)	$F = \overline{B} + 1$ (negate)
1 1	$F = A + \overline{B}$	$F = A + \overline{B} + 1$ (subtract)

5. Inputs X_i and Y_i of each full adder in an arithmetic circuit have digital logic specified by the Boolean functions

$$X_i = A_i \qquad Y_i = \overline{B}_i S + B_i \overline{C}_{\text{in}}$$

where S is a selection variable, C_{in} is the input carry, and A_i and B_i are input data for stage i.

(a) Draw the logic diagram for the 4-bit circuit, using full adders and multiplexers.

(b) Determine the arithmetic operation performed for each of the four combinations of S and C_{in}: 00, 01, 10, and 11.

6. *Design one bit of a digital circuit that performs the four logic operations of exclusive-OR, exclusive-NOR, NOR, and NAND on register operands A and B with the result to be loaded into register A. Use two selection variables.

(a) Using a Karnaugh map, design minimum logic for one typical stage, and show the logic diagram.

(b) Repeat (a), trying different assignments of the selection codes to the four operations to see whether the logic for the stage can be simplified further.

7. +Design an ALU that performs the following operations:

$A + B$	sr A
$A + \overline{B} + 1$	$A \vee B$
$\overline{B}$	sl A
$\overline{B} + 1$	$A \wedge B$

Give the result of your design as the logic diagram for a single stage of the ALU. Your design should have one carry line to the left and one carry line to the right between stages and three selection bits. If you have access to logic optimization software, apply it to the design to obtain reduced logic.

8. *Find the output Y of the 4-bit barrel shifter in Figure 9 for each of the following bit patterns applied to S_1, S_0, D_3, D_2, D_1, and D_0:

(a) 110101 **(b)** 101011

(c) 011010 **(d)** 001101

9. Specify the 16-bit control word that must be applied to the datapath of Figure 11 to implement each of the following microoperations:

(a) $R5 \leftarrow 0$ **(b)** $R4 \leftarrow$ sl $R5$

(c) $R7 \leftarrow$ Data in **(d)** $R3 \leftarrow$ sr $R3$

(e) $R1 \leftarrow R3 - \text{Constant in}$ **(f)** $R1 \leftarrow R1 + 1$

(g) $R2 \leftarrow R1 \oplus R3$ **(h)** $R4 \leftarrow R3 + R5$

10. *Given the following 16-bit control words for the datapath of Figure 11, determine (a) the microoperation that is executed and (b) the change in the contents of the register for each control word (assume that the registers are 8-bit registers and that, before the execution of a control word, they contain the value of their number (e.g., register $R5$ contains 05 in hexadecimal)). Assume that Constant has value 6 and Data in has value 1B, both in hexadecimal.

(a) 101 100 101 0 1000 0 1 **(b)** 110 010 100 0 0101 0 1

(c) 101 110 000 0 1100 0 1 **(d)** 101 000 000 0 0000 0 1

(e) 100 100 000 1 1101 0 1 **(f)** 011 000 000 0 0000 1 1

11. Given the sequence of 16-bit control words below for the datapath in Figure 11 and the initial ASCII character codes in 8-bit registers, simulate the datapath to determine the alphanumeric characters in the registers after the execution of the sequence. The result is a scrambled word: what is it?

011 011 001 0 0010 0 1	$R0$	00000000
100 100 001 0 1001 0 1	$R1$	00100000
101 101 001 0 1010 0 1	$R2$	01000100
001 001 000 0 1011 0 1	$R3$	01000111
001 001 000 0 0001 0 1	$R4$	01010100
110 110 001 0 0101 0 1	$R5$	01001100
111 111 001 0 0101 0 1	$R6$	01000001
001 111 000 0 0000 0 1	$R7$	01001001

12. A computer has a 32-bit instruction word broken into fields as follows: opcode, six bits; two register fields, five bits each; and one immediate operand/register field, 16 bits.

(a) What is the maximum number of operations that can be specified?

(b) How many registers can be addressed?

(c) What is the range of unsigned immediate operands that can be provided?

(d) What is the range of signed immediate operands that can be provided, assuming that bit 15 is the sign bit?

13. *A digital computer has a memory unit with a 32-bit instruction and a register file with 64 registers. The instruction set consists of 130 different operations. There is only one type of instruction format, with an opcode

part, a register file address, and an immediate operand part. Each instruction is stored in one word of memory.

(a) How many bits are needed for the opcode part of the instruction?

(b) How many bits are left for the immediate part of the instruction?

(c) If the immediate operand is used as an unsigned address to memory, what is the maximum number of words that can be addressed in memory?

(d) What are the largest and the smallest algebraic values of signed 2s complement binary numbers that can be accommodated as an immediate operand?

14. A digital computer has 32-bit instructions. There are a number of different instruction formats, and the number of bits in each format used for opcodes varies depending on the bits needed for other fields. If the first bit of the opcode is 0, then there are three opcode bits. If the first bit of the opcode is 1 and the second bit of the opcode is 0, then there are six opcode bits. If the first bit of the opcode is 1 and the second bit of the opcode is 1, then there are nine opcode bits. How many distinct opcodes are available for this computer?

15. The single-cycle computer in Figure 15 executes the five instructions described by the register transfers in the table that follows.

(a) Complete the following table, giving the binary instruction decoder outputs from Figure 16 during execution of each of the instructions:

Instruction—Register Transfer	DA	AA	BA	MB	FS	MD	RW	MW	PL	JB
$R[0] = R[7] \oplus R[3]$										
$R[1] \leftarrow M[R[4]]$										
$R[2] \leftarrow R[5] + 2$										
$R[3] \leftarrow \text{sl } R[6]$										
if (R([4] = 0) $PC \leftarrow PC +$ se PC else $PC \leftarrow PC + 1$										

(b) Complete the following table, giving the instruction in binary for the single-cycle computer that executes the register transfer (if any field is not used, give it the value 0):

Instruction—Register Transfer	Opcode	DR	SA	SB or Operand
$R[0] = \text{sr } R[7]$				
$R[1] \leftarrow M[R[6]]$				
$R[2] \leftarrow R[5] + 4$				
$R[3] \leftarrow R[4] \oplus R[3]$				
$R[4] \leftarrow R[2] - R[1]$				

16. Using the information in the truth table in Table 10, verify that the design for the single-bit outputs in the decoder in Figure 16 is correct.

17. Manually simulate the single-cycle computer in Figure 15 for the following sequence of instructions, assuming that each register initially contains contents equal to its index (i.e., *R*0 contains 0, *R*1 contains 1, and so on):

```
ADD R0, R1, R2
SUB R3, R4, R5
SUB R6, R7, R0
ADD R0, R0, R3
SUB R0, R0, R6
ST R7, R0
LD R7, R6
ADI R0, R6, 2
ADI R3, R6, 3
```

Give (a) the binary value of the instruction on the current line of the results and (b) the contents of any register changed by the instruction, or the location and contents of any memory location changed by the instruction on the next line of the results. The results are positioned in this fashion because the new values do not appear in a register or memory, due to the execution of an instruction, until after a positive clock edge has occurred.

18. Give an instruction for the single-cycle computer that resets register *R*4 to 0 and updates the *Z* and *N* status bits based on the value 0 transferred to *R*4. [*Hint*: Try the exclusive-OR.] By examining the detailed ALU logic, determine the values of the *V* and *C* status bits.

19. List the control logic state table entries for the multiple-cycle computer (see Tables 12, 13 and 15) that implement the following register transfer statements. Assume that in all cases the present state is EX0. If an opcode is needed, use a symbolic name based on the problem part—e.g., for part (a), opcode_a.

(a) $R3 \leftarrow R7 - R2, \rightarrow EX1$. Assume DR = 3, SA = 7, SB = 2.

(b) $R8 \leftarrow \text{sr } R8, \rightarrow INF$. Assume DR = 5, SB = 5.

(c) if $(Z = 0)$ then $(PC \rightarrow PC + \text{se } AD, \rightarrow INF)$ else $(PC \rightarrow PC + 1, \rightarrow INF)$.

(d) $R6 \leftarrow R6, C \leftarrow 0, \rightarrow INF$. Assume DR = SA = 6.

20. **(a)** Manually simulate the SRM (shift right multiple) instruction in the multiple-cycle computer for operand 0101100111000111 for OP = 5.

(b) Repeat part (a) for the SLM (shift left multiple) instruction.

21. +In the SRM and SLM instructions, both the operand R[SA] and the shift amount field OP are checked to see if either is 0 before the shifts begin.

(a) Redraw the state machine diagram for these operations with these checks removed.

(b) Use the original diagram and the new diagram to compare the number of clock cycles required for values of OP equal to 0 through 7. Assume that the probability of each OP value for 1 through 6 is 1/8, for 0 is 1/4, and for 7 is 0. Assume that the likelihood of a 0 operand is 1/8. Perform calculations to determine the best implementation (with checks or without checks) based on the given probability information and comparative number of clock cycles for the two implementations. Provide a convincing argument for your selected answer.

22. A new instruction is to be defined for the multiple-cycle computer with opcode 0010001. The instruction implements the register transfer

$$R[\text{DR}] \leftarrow R[\text{SB}] + M[R[\text{SA}]]$$

Find the state machine diagram for implementing the instruction, assuming that 0010001 is the opcode. Form the part of the control state table that implements this instruction.

23. Repeat Problem 22 for the two instructions: Add and check OV (AOV), described by the register transfer

$$R[\text{DR}] \leftarrow R[\text{SA}] + R[\text{SB}], \quad V\text{: } R8 \leftarrow 1, \quad \overline{V}\text{: } R8 \leftarrow 0$$

and branch on overflow (BRV), described by the register transfer

$$R8 \leftarrow R8, \quad V\text{: } PC \leftarrow PC + \text{se AD}, \quad \overline{V}\text{: } PC \leftarrow PC + 1$$

The opcode for AOV is 1000101 and for BRV is 1000110. Note that register $R8$ is used as a "status" register that stores the overflow result V for the previous operation. All of the values N, Z, C and V could be stored in $R8$ to give a complete status on the prior arithmetic or logic operation.

24. A new instruction is to be defined for the multiple-cycle computer. The instruction compares two unsigned integers stored in register R[SA] and R[SB]. If the integers are equal, then bit 0 of R[DR] is set to 1. If R[SA] is greater than R[SB], then bit 1 of R[DR] is set to 1. Otherwise, bits 0

and 1 are both 0. All other bits of R[DR] have value 0. Find the state machine diagram for implementing the instruction, assuming that 0010001 is the opcode. Form the part of the control state table that implements this instruction.

25. A new instruction, SMR (Store Multiple Registers), with symbolic opcode name SMR, is to be implemented for the multiple-cycle computer. The instruction stores the contents of eight registers in eight consecutive memory locations. Register R[SA] specifies the address in memory M to which the first register R[SB] is to be stored. The registers to be stored are R[SB], R[(SB + 1)modulo 8], ..., R[(SB + 7)modulo 8] in Memory M addresses R[SA], R[SA] + 1, ..., R[SA] + 7. Design this instruction presenting your final results in the form shown in Table 15.

26. A new instruction LMR (Load Multiple Registers), with symbolic opcode name LMR, is to be implemented for the multiple-cycle computer. The instruction is to retrieve the register contents stored by use of SMR in Problem 25 from memory M and place it in the eight registers. Assume that R[SA] and SB have same values as for SMR for such a retrieval. Design this instruction presenting your final results in the form shown in Table 15.