CMP SCI 2700 – Homework 2

UMSL

Complete Chapter 11 End of Chapter **Problems**

- 11.1
- 11.3

Complete Chapter 3 End of Chapter **Problems**

- 3.3
- 3.7

Complete Chapter 4 End of Chapter **Problems**

- 4.2
- 4.3
- 4.4
- 4.8

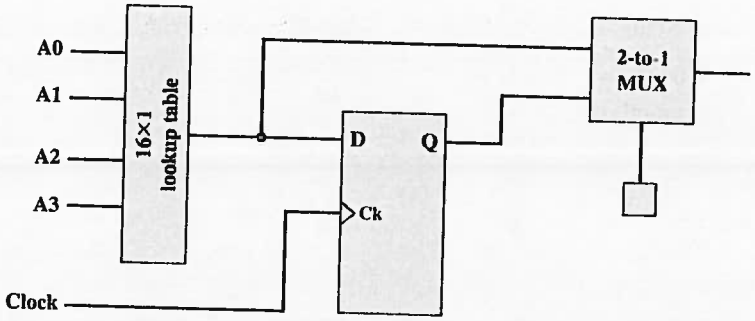

Figure 11.34 A Simple FPGA Logic Block

11.6 KEY TERMS AND PROBLEMS

Key Terms

adder	OR gate	register
AND gate	parallel register	excitation table
assert	combinational circuit	field-programmable gate array (FPGA)
Boolean algebra	complex PLD (CPLD)	flip-flop
clocked S–R flip-flop	counter	ripple counter
D flip-flop	decoder	sequential circuit
gates	product of sums (POS)	shift register
graphical symbol	programmable array logic (PAL)	simple PLD (SPLD)
J–K flip-flop	programmable logic array (PLA)	sum of products (SOP)
Karnaugh map	programmable logic device (PLD)	synchronous counter
logic block	Quine–McCluskey method	S–R Latch
lookup table	read-only memory (ROM)	truth table
multiplexer		XOR gate
NAND gate		
NOR		

Problems

11.1 Construct a truth table for the following Boolean expressions:

a. $ABC + \overline{A}\,\overline{B}\,\overline{C}$

b. $ABC + A\,\overline{B}\,\overline{C} + \overline{A}\,\overline{B}\,\overline{C}$

c. $A(B\overline{C} + \overline{B}C)$

d. $(A + B)(A + C)(\overline{A} + \overline{B})$

11.2 Simplify the following expressions according to the commutative law:

a. $A \cdot \overline{B} + \overline{B} \cdot A + C \cdot D \cdot E + \overline{C} \cdot D \cdot E + E \cdot \overline{C} \cdot D$

b. $A \cdot B + A \cdot C + B \cdot A$

c. $(L \cdot M \cdot N)(A \cdot B)(C \cdot D \cdot E)(M \cdot N \cdot L)$

d. $F \cdot (K + R) + S \cdot V + W \cdot \overline{X} + V \cdot S + \overline{X} \cdot W + (R + K) \cdot F$

11.3 Apply DeMorgan's theorem to the following equations:
- **a.** $F = \overline{V + A + L}$
- **b.** $F = \overline{A} + \overline{B} + \overline{C} + \overline{D}$

11.4 Simplify the following expressions:
- **a.** $A = S \cdot T + V \cdot W + R \cdot S \cdot T$
- **b.** $A = T \cdot U \cdot V + X \cdot Y + Y$
- **c.** $A = F \cdot (E + F + G)$
- **d.** $A = (P \cdot Q + R + S \cdot T)T \cdot S$
- **e.** $A = \overline{\overline{D} \cdot \overline{D} \cdot E}$
- **f.** $A = Y \cdot (W + X + \overline{\overline{Y} + \overline{Z}}) \cdot Z$
- **g.** $A = (B \cdot E + C + F) \cdot C$

11.5 Construct the operation XOR from the basic Boolean operations AND, OR, and NOT.

11.6 Given a NOR gate and NOT gates, draw a logic diagram that will perform the three-input AND function.

11.7 Write the Boolean expression for a four-input **NAND gate**.

11.8 A combinational circuit is used to control a seven-segment display of decimal digits, as shown in Figure 11.35. The circuit has four inputs, which provide the four-bit code used in packed decimal representation ($0_{10} = 0000, \ldots, 9_{10} = 1001$). The seven outputs define which segments will be activated to display a given decimal digit. Note that some combinations of inputs and outputs are not needed.
- **a.** Develop a truth table for this circuit.
- **b.** Express the truth table in SOP form.
- **c.** Express the truth table in POS form.
- **d.** Provide a simplified expression.

11.9 Design an 8-to-1 multiplexer.

11.10 Add an additional line to Figure 11.15 so that it functions as a demultiplexer.

11.11 The Gray code is a binary code for integers. It differs from the ordinary binary representation in that there is just a single bit change between the representations of any two numbers. This is useful for applications such as counters or analog-to-digital converters where a sequence of numbers is generated. Because only one bit changes at a time, there is never any ambiguity due to slight timing differences. The first eight elements of the code are

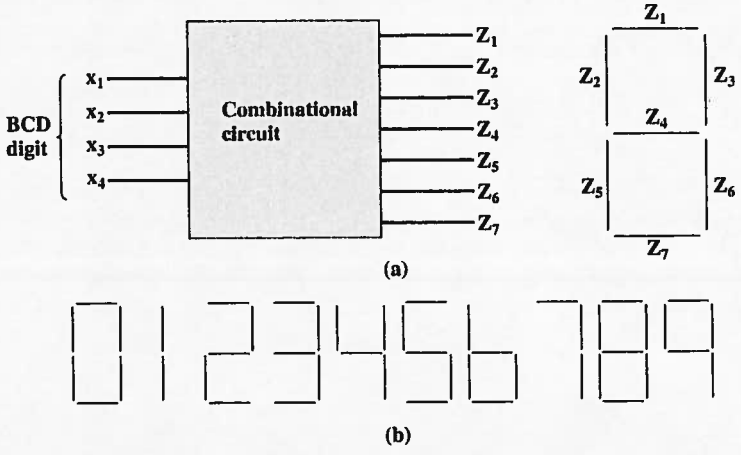

Figure 11.35 Seven-Segment LED Display Example

3.3 Consider a hypothetical 32-bit microprocessor having 32-bit instructions composed of two fields: the first byte contains the opcode and the remainder the immediate operand or an operand address.

a. What is the maximum directly addressable memory capacity (in bytes)?

b. Discuss the impact on the system speed if the microprocessor bus has:

1. 32-bit local address bus and a 16-bit local data bus, or

2. 16-bit local address bus and a 16-bit local data bus.

c. How many bits are needed for the program counter and the instruction register?

3.4 Consider a hypothetical microprocessor generating a 16-bit address (for example, assume that the program counter and the address registers are 16 bits wide) and having a 16-bit data bus.

a. What is the maximum memory address space that the processor can access directly if it is connected to a "16-bit memory"?

b. What is the maximum memory address space that the processor can access directly if it is connected to an "8-bit memory"?

c. What architectural features will allow this microprocessor to access a separate "I/O space"?

d. If an input and an output instruction can specify an 8-bit I/O port number, how many 8-bit I/O ports can the microprocessor support? How many 16-bit I/O ports? Explain.

3.5 Consider a 32-bit microprocessor, with a 16-bit external data bus, driven by an 8-MHz input clock. Assume that this microprocessor has a bus cycle whose minimum duration equals four input clock cycles. What is the maximum data transfer rate across the bus that this microprocessor can sustain, in bytes/sec? To increase its performance, would it be better to make its external data bus 32 bits or to double the external clock frequency supplied to the microprocessor? State any other assumptions you make, and explain. *Hint:* Determine the number of bytes that can be transferred per bus cycle.

3.6 Consider a computer system that contains an I/O module controlling a simple keyboard/printer teletype. The following registers are contained in the processor and connected directly to the system bus:

INPR: Input Register, 8 bits
OUTR: Output Register, 8 bits
FGI: Input Flag, 1 bit
FGO: Output Flag, 1 bit
IEN: Interrupt Enable, 1 bit

Keystroke input from the teletype and printer output to the teletype are controlled by the I/O module. The teletype is able to encode an alphanumeric symbol to an 8-bit word and decode an 8-bit word into an alphanumeric symbol.

a. Describe how the processor, using the first four registers listed in this problem, can achieve I/O with the teletype.

b. Describe how the function can be performed more efficiently by also employing IEN.

3.7 Consider two microprocessors having 8- and 16-bit-wide external data buses, respectively. The two processors are identical otherwise and their bus cycles take just as long.

a. Suppose all instructions and operands are two bytes long. By what factor do the maximum data transfer rates differ?

b. Repeat assuming that half of the operands and instructions are one byte long.

3.8 Figure 3.26 indicates a distributed arbitration scheme that can be used with an obsolete bus scheme known as Multibus I. Agents are daisy-chained physically in priority order. The left-most agent in the diagram receives a constant *bus priority in* (BPRN) signal indicating that no higher-priority agent desires the bus. If the agent does not require the bus, it asserts its *bus priority out* (BPRO) line. At the beginning of a clock

direct mapping	logical cache	spatial locality
high-performance computing (HPC)	memory hierarchy	split cache
hit	miss	tag
hit ratio	multilevel cache	temporal locality
instruction cache	physical address	unified cache
L1 cache	physical cache	virtual address
L2 cache	random access	virtual cache
L3 cache	replacement algorithm	write back
line	secondary memory	write through
locality	sequential access	
	set-associative mapping	

Review Questions

4.1 What are the differences among sequential access, direct access, and random access?

4.2 What is the general relationship among access time, memory cost, and capacity?

4.3 How does the principle of locality relate to the use of multiple memory levels?

4.4 What are the differences among direct mapping, associative mapping, and set-associative mapping?

4.5 For a direct-mapped cache, a main memory address is viewed as consisting of three fields. List and define the three fields.

4.6 For an associative cache, a main memory address is viewed as consisting of two fields. List and define the two fields.

4.7 For a set-associative cache, a main memory address is viewed as consisting of three fields. List and define the three fields.

4.8 What is the distinction between spatial locality and temporal locality?

4.9 In general, what are the strategies for exploiting spatial locality and temporal locality?

Problems

4.1 A set-associative cache consists of 64 lines, or slots, divided into four-line sets. Main memory contains 4K blocks of 128 words each. Show the format of main memory addresses.

4.2 A two-way set-associative cache has lines of 16 bytes and a total size of 8 kB. The 64-MB main memory is byte addressable. Show the format of main memory addresses.

4.3 For the hexadecimal main memory addresses 111111, 666666, BBBBBB, show the following information, in hexadecimal format:

- **a.** Tag, Line, and Word values for a direct-mapped cache, using the format of Figure 4.10
- **b.** Tag and Word values for an associative cache, using the format of Figure 4.12
- **c.** Tag, Set, and Word values for a two-way set-associative cache, using the format of Figure 4.15

4.4 List the following values:

- **a.** For the direct cache example of Figure 4.10: address length, number of addressable units, block size, number of blocks in main memory, number of lines in cache, size of tag
- **b.** For the associative cache example of Figure 4.12: address length, number of addressable units, block size, number of blocks in main memory, number of lines in cache, size of tag

c. For the two-way set-associative cache example of Figure 4.15: address length, number of addressable units, block size, number of blocks in main memory, number of lines in set, number of sets, number of lines in cache, size of tag

4.5 Consider a 32-bit microprocessor that has an on-chip 16-kB four-way set-associative cache. Assume that the cache has a line size of four 32-bit words. Draw a block diagram of this cache showing its organization and how the different address fields are used to determine a cache hit/miss. Where in the cache is the word from memory location ABCDE8F8 mapped?

4.6 Given the following specifications for an external cache memory: four-way set associative; line size of two 16-bit words; able to accommodate a total of 4K 32-bit words from main memory; used with a 16-bit processor that issues 24-bit addresses. Design the cache structure with all pertinent information and show how it interprets the processor's addresses.

4.7 The Intel 80486 has an on-chip, unified cache. It contains 8 kB and has a four-way set-associative organization and a block length of four 32-bit words. The cache is organized into 128 sets. There is a single "line valid bit" and three bits, B0, B1, and B2 (the "LRU" bits), per line. On a cache miss, the 80486 reads a 16-byte line from main memory in a bus memory read burst. Draw a simplified diagram of the cache and show how the different fields of the address are interpreted.

4.8 Consider a machine with a byte addressable main memory of 2^{16} bytes and block size of 8 bytes. Assume that a direct mapped cache consisting of 32 lines is used with this machine.

a. How is a 16-bit memory address divided into tag, line number, and byte number?

b. Into what line would bytes with each of the following addresses be stored?

0001	0001	0001	1011
1100	0011	0011	0100
1101	0000	0001	1101
1010	1010	1010	1010

c. Suppose the byte with address 0001 1010 0001 1010 is stored in the cache. What are the addresses of the other bytes stored along with it?

d. How many total bytes of memory can be stored in the cache?

e. Why is the tag also stored in the cache?

4.9 For its on-chip cache, the Intel 80486 uses a replacement algorithm referred to as **pseudo least recently used**. Associated with each of the 128 sets of four lines (labeled L0, L1, L2, L3) are three bits B0, B1, and B2. The replacement algorithm works as follows: When a line must be replaced, the cache will first determine whether the most recent use was from L0 and L1 or L2 and L3. Then the cache will determine which of the pair of blocks was least recently used and mark it for replacement. Figure 4.19 illustrates the logic.

a. Specify how the bits B0, B1, and B2 are set and then describe in words how they are used in the replacement algorithm depicted in Figure 4.19.

b. Show that the 80486 algorithm approximates a true LRU algorithm. *Hint:* Consider the case in which the most recent order of usage is L0, L2, L3, L1.

c. Demonstrate that a true LRU algorithm would require 6 bits per set.

4.10 A set-associative cache has a block size of four 16-bit words and a set size of 2. The cache can accommodate a total of 4096 words. The main memory size that is cacheable is 64K × 32 bits. Design the cache structure and show how the processor's addresses are interpreted.

4.11 Consider a memory system that uses a 32-bit address to address at the byte level, plus a cache that uses a 64-byte line size.

a. Assume a direct mapped cache with a tag field in the address of 20 bits. Show the address format and determine the following parameters: number of addressable units, number of blocks in main memory, number of lines in cache, size of tag.