AMERICAN UNIVERSITY OF BEIRUT FACULTY OF ENGINEERING AND ARCHITECTURE DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING		
PROF. M. SAGHIR	**EECE 321 MICROPROCESSOR SYSTEMS**	SPRING 2003-2004
QUIZ 2		
NAME:		APRIL 30, 2004
CLOSED BOOK (2 HRS) *NON-PROGRAMMABLE CALCULATORS ARE ALLOWED* *PROVIDE SOLUTION ON THE ANSWER BOOKLET* *THE QUESTION SHEET MUST BE RETURNED WITH THE ANSWER BOOKLET*		

READ ALL QUESTIONS ON ALL PAGES CAREFULLY.

(10%)

1. Consider the following fragment of C++ code:

```
if (x > y) {
  z = 0;
}
else {
  z = 255;
}
```

Assuming **x**, **y**, and **z** are 8-bit variables, show the sequence of PIC16F84 declarations and instructions needed to implement the above code fragment.

Since there are no compare or conditional branch instructions in the PIC16F84 instruction set, we have to use available instructions to achieve the same thing: subtract x from y (or y from x) and use a bit-test-and-skip (`BTFSC`; `BTFSS`) to test the appropriate flag (e.g. zero, carry) in the status register. Alternatively, you can test the value of the result (to check if it's zero) or its sign (to check if it's negative). A `GOTO` instruction can be used immediately after the bit-test-and-skip instruction to jump to the 'else' sequence.

(25%)

2. Consider the following PIC16F84 code fragment:

```
COUNT1    EQU     d'12'
COUNT2    EQU     d'13'
          :
          :
          ORG     0x0
DELAY     MOVLW   C
          MOVWF   COUNT1
          MOVLW   d'0'
          MOVWF   COUNT2
```

```
LOOP    INCFSZ    COUNT1
        GOTO      LOOP
        DECFSZ    COUNT2
        GOTO      LOOP
```

Assuming the PIC16F84 operates at 5 MHz, and ignoring the effects of the watchdog timer, what should the value of ***C*** be so that the above sequence of instructions executes in 63.4896 msec?

As given, the outer loop (the one controlled by `COUNT2`) will iterate 256 times (`DECFSZ` executes 256 times and `GOTO` executes 255 times), and the inner loop (the one controlled by COUNT1) will iterate 256-C times on the first iteration of the outer loop, and 256 times on subsequent iterations. All instructions execute in 4 clock cycles; branches (and skips) execute in 8 clock cycles. The clock period is 0.2 μsec. The actual execution time will therefore be greater than 63.4896 msec. Correct reasoning was treated as a correct answer.

(15 %)

3. Interrupts.

   a) What is a non-maskable interrupt? (2%)

      A non-maskable interrupt is one that cannot be disabled. It is always serviced and is typically assigned to the event with the highest priority (e.g. power failure).

   b) The MC68000 uses autovectored interrupts. What is the difference between vectored and autovectored interrupts? (3%) Briefly explain the steps taken to service an autovectored interrupt request. (10%)

      Please refer to Section 4.3.2 of the textbook.

(40 %)

4. Consider the following MC68000 instruction:

   ```
   ADD.W   $20000,(A3)+
   ```

   a) How many words does this instruction occupy in memory? (5%)

      Although this is not a valid MC68000 instruction (invalid addressing mode for result), one can reason that such an instruction would occupy three, 16-bit words in memory: one to encode the opcode, register specifier, and addressing modes, and two to store the long immediate operand.

   b) How many memory accesses are needed to fetch and execute this instruction? (5%)

      Three memory reads are needed to fetch the instruction, two memory reads are needed to fetch the source operands, and one memory write is needed to store the result: Total: six memory accesses.

c) Assuming the MC68000 is connected to main memory over a synchronous system bus, use a timing diagram to illustrate the activity on the bus (i.e. the changes on the address, data, and control lines) when fetching and executing the above instruction. (25%)

You may assume the following:

- The address bus is 24 bits wide and the data bus is 16 bits wide.
- The control bus includes the following signals: Read/Write and MFC (memory function complete – you can think of this as a slave ready signal).
- The processor is already the bus master.
- It takes a finite (but irrelevant) amount of time to decode and execute the instruction.

Since the bus is synchronous and a slave ready signal is used, we can conclude that a memory access will require multiple bus accesses (refer to Figure 4.25 in the textbook). Every memory read or write will require three bus cycles: during the first bus cycle, the master places the address and the command on the bus (in case of a write; it also places the data on the bus); during he second bus cycle, the signals are given an opportunity to propagate along the bus, and the slave is given time to decode the address after accounting for skew; finally, during the third bus cycle, the slave places the data on the bus (or reads the data from the bus in case of a memory write) and asserts the MFC signal.

d) How many bus cycles are needed to fetch and execute the above instruction? (5%)

Six memory accesses × three bus cycles / memory access = 18 bus cycles.

(3%)

5. What are the two DMA transfer modes? (1%) How are they different? (2%)

Cycle stealing and burst/block mode. Please refer to your notes for an explanation of how they are different.

(5%)

6. In the PCI bus protocol, what is the function of the **FRAME#** signal? (2%) How does it work? (3%)

The FRAME# signal is used to mark the duration of the bus transfer. It remains asserted until the bus cycle before last.

(2%)

7. The USB bus supports isochronous data transfers. What does this mean?

Isochronous data transfers are those that are synchronized to some clock. Examples include samples of streaming speech, music, or video.