EE2004: Microcomputer Systems

Test 1 Review

Topics

1. Introduction to Computer Systems
   (a) Buses
   - Address Bus – unidirectional
     - How many memory locations can be addressed by an N-bit address bus?
   - Data Bus – bidirectional
     - What is the largest unsigned number that can be carried into a CPU by an N-bit data bus?
   - Control Bus
   - Describe the interactions between these three types of buses in the read and write operations.

   (b) Three different components:
   - Central processing unit (CPU): Made up of three components:
     - Arithmetic-logic unit (ALU)
     - Registers
     - Control unit: controls instruction processing
   - Memory unit
   - Input/Output (I/O) unit
     - Define peripherals and ports.

   (c) Terminology
   - How many bits are there in a nibble? byte?
   - Write down the exact values represented by kilo, mega, giga.

   (d) Harvard vs. Von Neumann Architecture
   - Von Neumann
     - Data and program memories are combined
     - Fetching and execution cannot be done in the same instruction cycle
   - Harvard
     - Data and program memories are separated.
     - Each has its own address and data buses
     - Fetching and execution can be done in the same instruction cycle.

(e) Describe the difference between microcontroller and microprocessor
- Microprocessor is just the CPU. Peripherals are implemented separately and are required to connect to CPU by long wires.
- In a microcontroller, CPU and all peripherals are implemented in one chip
- Harvard architecture is too expensive to be implemented in microprocessor. CPU and peripherals are required to be connected by long wires (costly). The wire required for Harvard is doubled. → Von Neumann is preferred
- We can afford two sets of buses required for Harvard in microcontroller because CPU and peripherals are packed together.

2. PIC18 Microcontroller
   (a) Architecture
   (b) Access bank vs. BSR
   (c) Understand the documentation in data sheet
   (d) How the status register changes in an addition operation
   (e) Describe the difference between addwf and addwfc.

3. I/O Programming

   (a) Code assembly language to use the ports for input or output. Need to know how to set the TRISX register appropriately, where X = {A, B, C, D, E} depending on the port being used.

   (b) Code I/O bit manipulation programs for the PIC

   (c) Check the state of an I/O port (i.e., PORTX register, where X = {A, B, C, D, E}) and make branching decision based on it

   (d) Interface with 1-digit/4-digit 7-segment LED and keypad matrix

4. PIC18 Assembly Language Programming

4.1

(a) Terminology
- Machine vs. assembly language
- 2 types of assembly language statements: instructions and directives
- 4 elements of an assembly language statement:
  - label
  - mnemonics
  - operands
  - comments

(b) Know functions of common directives: org, set, equ, cblock
(c) Fetching and execution in PIC18
- Understand each step of my animation

4.2

(a) Know instructions used for subtracting unsigned number
(b) Know microcontroller does not subtract; it adds a negative number; negative number is represented by the 2's complement format
(c) Be able to determine all status flags in an addition operation
- OV: would not occur when +ve added by a –ve number
- If +ve added by a +ve results in an –ve number *or* –ve added by a –ve results in a +ve number, then OV must have occurred.

(d) BCD addition
- Under two conditions, you would get an incorrect BCD addition result
- Use daw to adjust

(e) Compare instruction: Make branching decision based on the value in a file register

4.3
(a) Looping
- Initialization
- Statements to repeat
- monitor number of iterations/repetitions

(b) Conditional jumps
- make branching decision based on status flag
- encode destination by relative address

- relative address is 8-bit, ranging from -128 to 127

(c) goto
- encode destination by absolute address
- last bit of address not encoded; only most significant 20 bits are encoded.

(d) bra
- encode destination by relative address
- relative address is 11-bit, ranging from -1024 to 1023

(e) Know how absolute/relative addresses are encoded in machine code.

(f) Branch instruction timing
- Conditional jumping: 2 instruction cycles if jump, 1 if not jump
- Unconditional jumping always takes 2 instruction cycles.
- Conditional skipping: 1 instruction cycle if not jump, 2 or 3 if jump.

(g) Application of nested loop in generating time delay
- Calculate the total time delay generated by the loop with different values of [DELAY_H] and [DELAY_L]. See Week 6 tutorial.

4.4

(a) Know difference between branching and subroutine calling
- After calling a subroutine, program counter needs to get back to the main program.
- Return address must be stored in hardware stack before executing the subroutine

(b) List the events that occur when calling and returning from a subroutine

(c) Know how the instruction call encodes absolute address of destination and the instruction rcall encodes relative address of destination.

(d) Know how to determine the contents of the hardware stack, TOS and STKPTR at all points of a program, and in particular immediately after the execution of a call/rcall or return instruction.
- Be prepared to answer a question similar to the tutorial question.