CS 301 Assignment 4

Student ID: 200482797

Owen Monus

April 11th, 2024

1. (a) You are given an one-address machine with an accumulator and the following memory values:
   Word 20 contains 40.
   Word 30 contains 50.
   Word 40 contains 60.
   Word 50 contains 70

   What values do the following instructions load into the accumulator?

   i. LOAD IMMEDIATE 20
   This loads 20 into the accumulator
   ii. LOAD DIRECT 20
   This loads 40 into the accumulator
   iii. LOAD INDIRECT 20
   This loads 60 into the accumulator
   iv. LOAD IMMEDIATE 30
   This loads 30 into the accumulator
   v. LOAD DIRECT 30
   This loads 50 into the accumulator

   (b) An address field in an instruction contains a decimal value 14. Where is the corresponding operand located for...

   i. Immediate addressing?
   The operand is in the instruction
   ii. Direct addressing?
   The operand is in address 14
   iii. Indirect addressing?
   The operand is pointed to by the address contained in address 14
   iv. Register addressing?
   The operand is in register 14

v. Register indirect addressing?
The operand is pointed to by the address contained in register 14

2. Let the address stored in the program counter (PC) be designated by the symbol X1. The instruction stored in X1 has an address part (operand reference) X2. The operand needed to execute the instruction is stored in the memory word with address X3. An index register contains the value X4. What is the relationship between these various quantities if the addressing mode of the instruction is (a) direct; (b) indirect; (c) PC relative; (d) indexed?

   (a) X3 is equal to X2.
   (b) X3 is the value pointed to by the address stored in X2.
   (c) X3 is X1 + X2 + 1, since the PC is incremented before execution.
   (d) X3 is X2 + X4, the index register.

3. (a) A PC-relative mode branch instruction is 3 bytes long. The address of the instruction, in decimal, is 256028. Determine the branch target address if the signed displacement in the instruction is -31.

   PC = 256031, since instructions are 3 bytes long, the resultant address is: $256,000$.

   (b) A PC-relative mode branch instruction is stored in memory at address 620_{10}. The branch is made to location 530_{10}. The address field in the instruction is 10 bits long. What is the binary value in the instruction? Note that (x)10 indicates that x is a decimal number.

   The address field would contain the value 1110100101_2, or -91_{10}

4. (a) If the last operation performed on a computer with an 8-bit word was an addition in which the two operands were 00000010 and 00000011, what would be the value of the following flags?

   i. Overflow = 0
   ii. Sign = 0

   (b) If the last operation performed on a computer with an 8-bit word was a subtraction in which the two operands were 11110000 and 0010100 (i.e., 11110000 - 00010100 was performed), what would be the value of the following flags?

   i. Carry = 1
   ii. Zero = 0
   iii. Overflow = 0

   (c) A pipelined processor has a clock rate of 2.5 GHz and executes a program with 1.5 million instructions. The pipeline has five stages, and instructions are issued at a rate of one per clock cycle. Ignore penalties due to branch instructions and out-of-sequence executions. What is the speedup of this processor for this program compared to a nonpipelined processor? You can assume that (a) each instruction goes through all five stages of the pipeline; (b) the five stages are of equal duration; (c) all of the stages can be performed in parallel; and (d) there are no conditional branch instructions.

$\frac{1{,}500{,}000 \times 5}{1{,}500{,}000 + 5 - 1} \approx 4.999...$

5. Write a summary of what you have learned in this course in about 1000 words.

A digital system consists of two notions: architecture and organization. Architecture refers to the attributes of the system visible to a programmer. These include: instruction set, data types, memory addressing mode, and I/O mechanisms. Organization refers to the operational units and their interconnections. Organizational attributes involve control signals, interfaces, etc.

We describe a computer as a hierarchical system. Structure refers to the way in which the components are interrelated. We describe structure for a single processor computer and a multicore computer. Function refers to the operation of each individual component as part of the structure. There are four basic functions that a computer can perform: data processing, data storage, data movement, and control.

To improve speed, some techniques are in the following. Pipelining enables a processor to work simultaneously on multiple instructions by performing a different stage for each of the multiple instructions at the same time. Branch prediction: The processor predicts which instructions are likely to be processed next, and it pre-fetches the instructions so that the processor is kept busy. Superscalar execution: the ability to issue more than one instruction in every processor clock cycle. Data flow analysis: The processor analyzes which instructions are dependent on each other's results, to create an optimized schedule of instructions. Speculative execution: Using branch prediction and data flow analysis, some processors speculatively execute instructions ahead of their actual appearance in the program. This enables the processor to keep its execution engines as busy as possible.

The use of multiple processors on a same chip is called multicore. Multicore provides the potential to increase performance without increasing the clock rate. Cache memory can be shared among processors in multicore. Cache memory can also be placed in the same chip. When cache memory is larger, there can be multiple levels of cache. Within a processor, the increase in performance is roughly proportional to the square root of the increase in complexity.

The basic function of a computer is to execute programs, which is achieved by interconnecting CPU, memory, and I/O components. Interconnection can be bus interconnection or point-to-point interconnection. The processing required for a single instruction is called an instruction cycle. Two steps of an instruction cycle are the fetch cycle and the execute cycle. At the beginning of each instruction cycle, the processor fetches an instruction from memory. In a typical processor, a register called the program counter (PC) holds the address of the instruction to be fetched next. Unless told otherwise, the processor always increments the PC after each instruction fetch so that it will fetch the next instruction in sequence. The fetched instruction is loaded into a register in the processor known as the instruction register (IR). The instruction contains bits that specify the action the processor is to take. The processor interprets the instruction and performs the required action. In general, these actions fall into four categories. Processor-memory (load and store), Processor-I/O Data processing. Control.

Temporal locality refers to a program's tendency to access memory units in the near future that were recently accessed in the past, such as in iteration loops or when using constants and temporary variables. Spatial locality refers to a program's tendency to access memory units whose addresses are close to each other. This occurs when a unit of memory is accessed, like when processing arrays. There are 4 types of memory access: Sequential access involves accessing records in a linear order, typical of tape units. Direct access allows accessing individual blocks or records with unique addresses based on physical location, often associated with disk units. Random access enables selecting and accessing any location in memory directly through unique addresses, as seen in main memory and some cache memory. Associative access, a type of random access memory, retrieves words based on their content rather than their address. Each location has its addressing mechanism, with constant retrieval time. Cache memories may use associative access.

Greater memory capacity typically comes with slower access time, while higher access speeds usually result in greater cost per bit. Similarly, larger capacities tend to have lower cost per bit, while faster access times are often associated with higher costs per bit.

Instruction types are categorized as follows: data processing instructions encompass arithmetic and logic operations; data storage instructions involve moving data between registers and memory locations; data movement instructions handle input/output operations; and control instructions include branching operations.

Immediate addressing directly embeds the operand value within the instruction, saving a memory cycle but restricting the operand size to the address field. Direct addressing uses the address field to denote the location of the operand in memory. Indirect addressing employs a word in memory to hold the full-length address of the operand, expanding the address space but necessitating two memory references for operand retrieval. Register addressing replaces memory addresses with register references, facilitating faster execution but offering limited address space. Register indirect addressing, akin to indirect addressing, substitutes memory addresses with a register reference, sharing similar advantages and limitations. Displacement addressing merges direct and register indirect addressing, utilizing two address fields to calculate the effective address based on a register's contents.

There are three types of registers: user-facing, control, and status. User-visible registers enable assembly language programmer to minimize main memory references by optimizing use of registers. Control registers are used by the control unit to control the operation of the processor and by operating system to control the execution of programs. These registers are typically not visible to the user. The status register(s) contain condition codes plus other status information.

This is only **some** of the content covered in CS 301.