CS 271 Computer Architecture and Assembly Language

Self-Check for Lecture#19

Solutions are posted

1. Given a CISC machine with a 2 GHz clock (i.e., the clock ticks 2 billion times per second). This particular computer uses MASM-like instructions with the following timings:

add	*reg, mem*	9 clock cycles (i.e., the ADD microprogram has 9 instructions)
add	*reg,immed*	3 clock cycles
loop	*label*	7 clock cycles

Here's a short code fragment to sum the elements of a numeric array:

```
     mov  eax,0                ;initialize sum
     mov  ecx,MAX_SIZE         ;initialize loop counter
     mov  esi,OFFSET list      ;initialize array pointer
more:
     add  eax,[esi]            ;add current list element
     add  esi,4                ;move array pointer to next element
     loop more                 ;auto-decrement ecx, jump to more,
                               ;    if ecx ≠ 0
```

Assume unlimited array size. After initialization, how many array elements could be processed in 1 ms. (1 ms. = 1/1000 sec).

2. Given a RISC machine with a 2 GHz clock (i.e., the clock ticks 2 billion times per second). This particular computer uses an instruction cache, a data cache, an operand fetch unit, and an operand store unit. The instruction set includes simple instructions with the following timings:

set	*reg, immed*	1 clock cycle
load	*reg,mem*	2 clock cycles
add	*reg,reg*	2 clock cycles
add	*reg,immed*	1 clock cycle
loop	*label*	3 clock cycles

Here's a short code fragment to sum the elements of a numeric array:

```
     set  r1,0                ;initialize sum
     set  r2,MAX_SIZE         ;initialize loop counter
     set  r3,@list            ;initialize array pointer
more:
     load r4,[r3]             ;fetch current list element
     add  r1,r4               ;add current list element
     add  r3,4                ;move array pointer to next element
     loop more                ;auto-decrement r2, jump to more,
                              ;    if r2 ≠ 0
```

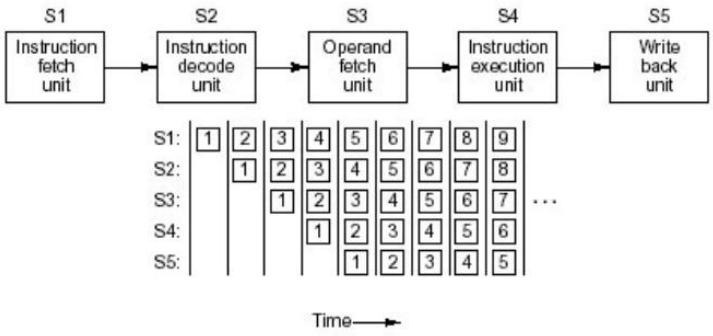
Assume unlimited array size. After initialization, how many array elements could be processed in 1 ms. (1 ms. = 1/1000 sec).

3. Given a five-stage pipeline as illustrated at the right: Suppose that each stage requires 3 nanoseconds (ns) to complete its task.
   a. How long will it take to complete 100 instructions with pipelining?

      _____ ns

   b. How long will it take to complete 100 instructions without pipelining?

      _____ ns



4. An algorithm takes 4seconds to execute on a single 2.4G processor. 30% of the algorithm is sequential. Assuming zero latency and perfect parallelism in the remaining code, how long should the algorithm take on a parallel machine with 8 2.4G processors?

_____ sec

5. Cite and explain two major reasons that software parallelism has not kept pace with developments in hardware parallelism.