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:

<table>
<thead>
<tr>
<th>Instruction</th>
<th>Timing</th>
</tr>
</thead>
<tbody>
<tr>
<td>add reg, mem</td>
<td>9 clock cycles</td>
</tr>
<tr>
<td>add reg,immed</td>
<td>3 clock cycles</td>
</tr>
<tr>
<td>loop label</td>
<td>7 clock cycles</td>
</tr>
</tbody>
</table>

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

```assembly
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:

<table>
<thead>
<tr>
<th>Instruction</th>
<th>Timing</th>
</tr>
</thead>
<tbody>
<tr>
<td>set reg,immed</td>
<td>1 clock cycle</td>
</tr>
<tr>
<td>load reg,mem</td>
<td>2 clock cycles</td>
</tr>
<tr>
<td>add reg,reg</td>
<td>2 clock cycles</td>
</tr>
<tr>
<td>add reg,immed</td>
<td>1 clock cycle</td>
</tr>
<tr>
<td>loop label</td>
<td>3 clock cycles</td>
</tr>
</tbody>
</table>
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 to next element,
     ; 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 4 seconds 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.