Topics

• Addressing
  – Tanenbaum 5.4

• Instruction types
  – Tanenbaum 5.5

Specifying Operand Locations

• Three operands
  – Destination = Source1 + Source2

• Two operands
  – Register2 = Register2 + Source1

• One operand (accumulator)
  – Accumulator = Accumulator + MemoryRef

• Zero operands
  – Stack architecture
Addressing Modes

• Immediate
  – Operand is specified within the instruction

• Direct
  – Give full memory address of operand

• Register
  – Specify the register where the operand is located

Addressing Modes

• Register indirect
  – Register contains a pointer for the memory address of the operand

• Indexed
  – Addressing memory by giving a register (explicit or implicit) plus a constant offset

• Based-indexed
  – Adds two registers plus an optional offset

• Stack
  – Memory referenced with a Last-In First-Out (LIFO) queue
Addressing Modes

<table>
<thead>
<tr>
<th>Addressing mode</th>
<th>Pentium II</th>
<th>UltraSPARC II</th>
<th>JVM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Immediate</td>
<td>×</td>
<td>×</td>
<td>×</td>
</tr>
<tr>
<td>Direct</td>
<td>×</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Register</td>
<td>×</td>
<td>×</td>
<td></td>
</tr>
<tr>
<td>Register indirect</td>
<td>×</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Indexed</td>
<td>×</td>
<td>×</td>
<td>×</td>
</tr>
<tr>
<td>Based-indexed</td>
<td></td>
<td>×</td>
<td></td>
</tr>
<tr>
<td>Stack</td>
<td></td>
<td></td>
<td>×</td>
</tr>
</tbody>
</table>

Figure 5-28. A comparison of addressing modes.

Instruction Types

- Data movement
- Dyadic (two operands)
- Monadic (one operand)
- Comparisons and conditional branches
- Procedure calls
- Loop control
- Input/output
IJVM Examples

- Data movement
  - Ex: ILOAD, ISTORE

- Dyadic (two operands)
  - Ex: IADD, ISUB, IMUL

- Monadic (one operand)
  - Ex: DUP, ISHR, ISHL

- Comparisons and conditional branches
  - Ex: IFEQ, IFLT

- Procedure calls
  - Ex: INVOKEVIRTUAL

Three Bears Theory of Computer Science

- The core of a modern computer
  - Deeply-pipelined
  - Three-register load/store RISC engine

- Some computer instructions are too big
  - Pentium II (IA-32 ISA)

- Some computer instructions are too small
  - picoJava II (JVM ISA)

- Some computer instructions are just right
  - UltraSPARC II (Version 9 SPARC)