The J Computer

Advay Mengle
MIT 6.111, Spring 2007
April 25, 2007
The J Computer

- Executes Java ME-like bytecode in hardware
- Constraints: (relatively) slow clock rate, not much memory
  - Use Connected, Limited Device Configuration v1.0
- Downloads user-specified class files from PC over RS-232 connection
- Interfaces with hardware devices to allow user interaction
System Architecture

Main General uP

Specialized uP for Static Class 1

Specialized uP for Static Class 2

Specialized uP for Static Class 3

Method Bus: control and data

SRAM

RS-232

More processors if FPGA size allows
System Architecture

Main
General uP

Specialized uP for Static Graphics Class

Specialized uP for Static System Class

VGA_Controller

Overlay_Manager

Method Bus: control and data

Mouse_Handler

Keybd_Handler

Generic PS2_Handler

N overlays

More processors if FPGA size allows

RGB

SRAM

RS-232

x,y

pos, color

ps2_frame

ps2_frame

scancode

x,y

mouse_clock

mouse_data

keybd_clock

keybd_data

mouse_clock

mouse_data

keybd_clock

keybd_data

ARGB, oue

x,y

x,y

x,y
Executing HLLs in Software vs. in Hardware

Software VM

Java Source

Compiled

javac

Java Bytecode

Verified

interpreted

JIT-compiled

java (VM)

Native MC

Executed

Native Processor

Java Source

Compiled

javac

Java Bytecode

Executed

μP

Each bytecode can be one or more native instructions
General Microprocessor: Simplified

**Bytecode Processing**
Variable length instructions split into 8-bit “cells”

*loading, parsing, flow control*

**SOP (Simple OPeration) Processing**
40-bit instructions w/ 8-bit op-code, and either 4 8-bit params or one 32-bit literal

*computation*
Most bytecodes can be directly translated into one or more constant SOPs (looked up in ROM).
Others may generate custom SOPs dynamically.
Goals & Challenges

- Final demo: Write MIT Pong in Java
- The most complex bytecodes: `instanceof`, `invoke_virtual`
  - Need to traverse type trees
  - Raise traps via SOP and write in Java, or
  - Implement via in-BRAM data structures
- Gate count
  - Can we fit 4 processors on a single FPGA?