Review from Last Lecture

- Course is now Great Ideas in Computer Architecture
  - Layers of Representation/Interpretation
  - Moore's Law
  - Principle of Locality/Memory Hierarchy
  - Parallelism
  - Performance Improvement
- Computer commands called instructions
- Computer vocabulary is called instruction set
  - MIPS is example instruction set in this class
  - Rigid format: 1 operation, 2 source operands, 1 destination operand
- 1st Design Principle: Simplicity favors regularity

Operands of Computer Hardware

- High-Level Programming languages: could have millions of variables
- Instruction sets have fixed, smaller number
- Called registers
  - The brick of computer hardware
  - Used to construct computer hardware
  - Visible to programmer
- MIPS Instruction Set has 32 registers

Why Just 32 Registers?

- Design Principle 2: Smaller is faster.
- Hardware would likely be slower with 64, 128, or 256 registers
- 32 is enough for compiler to translate typical C and Java programs are not run out of registers very often
  - ARM instruction set has 16 registers; may be faster, but compiler may run out of registers too often

Names of MIPS Registers

- For registers that hold programmer variables: $s0, s1, s2, ...
- For registers that hold temporary variables: $t0, t1, t2, ...
- Suppose variables $f, g, h, i,$ and $j$ are assigned to the registers $s0, s1, s2, s3,$ and $s4,$ respectively. What is MIPS for $f = (g + h) - (i + j);$
Size of Registers

- A bit is an atom of Computer Hardware: contains either 0 or 1
  - As we shall see, the real “alphabet” of computer hardware is 0, 1 so we will eventually express MIPS instructions as combinations of 0s and 1s
- MIPS registers are 32 bits wide
- MIPS calls this quantity a word
  - Some computers use 16-bit wide words, like Intel 80x86

What about Data Structures versus Simple Variables?

- In addition to registers, a computer also has memory that holds millions / billions of words
- Memory is a single dimension array, starting at 0
- To access memory need an address (index)
  - But MIPS instructions only operate on registers!
  - Solution: instructions that just transfer words (data) between memory and registers
- Called data transfer instructions

Transfer from Memory to Register?

- MIPS instruction is Load Word, abbreviated lw
- Assume that A is an array of 100 words, variables g and h map to registers $s1$ and $s2$, the starting address, or base address, of the array A is in $s3$.
- int A[100];
- g = h + A[8];
- Turns into
  - lw $t0, 8(s3)$ # Temporary reg $t0$ gets A[8]
  - add $s1, s2, t0$ # $g = h + A[8]$

Transfer from Register to Memory?

- MIPS instruction is Store Word, abbreviated sw
- Assume that A is an array of 100 words, variables g and h map to registers $s1$ and $s2$, the starting address, or base address, of the array A is in $s3$.
  - int A[100];
  - g = h + A[8];
  - Turns into
    - lw $t0, 32(s3)$ # Temporary reg $t0$ gets A[8]
    - add $s1, s2, t0$ # $g = h + A[8]$
    - add $s0, t0, t0$ # $t0 = h + A[8]$
It seems like he can no longer be focusing on his family. His wife, Brenda, complains, forgets things like dinner plans, and he has trouble symulating he gets from his electronic gadgets. He of data. Even after he unplugs, he craves the connection to the outside world. He continues to struggle with the effects of the deluge after apologizing to his suitor, Mr. Campbell, for overlooking it. The computer code he was writing.

The resulting distractions can have deadly consequences, as when physicians-eating drivers and train engineers crash wrecks. And for millions of people like Mr. Campbell, these urges can affect jobs and costs on creativity and deep thought, interrupting work and family life. While many people say multitasking makes them more productive, research shows otherwise. Many multitaskers actually have more trouble focusing and switching out information, scientists say, and they experience more stress. And scientists are discovering that even after the multid+r, brain’s functioning is still off balance. In other words, this is also your brain on computers."

Scientists say juggling e-mail, phone calls and other incoming information can change how people think and behave. They say our ability to focus is being undermined by bursts of information. These play to a primitive urge to respond to immediate opportunities and threats. The stimulation provides excitement – a dopamine squirt – that researchers say can be addictive. In its absence, people feel restless. The resulting distractions can have deadly consequences, as when physicians-eating drivers and train engineers crash wrecks. And for millions of people like Mr. Campbell, these urges can affect jobs and costs on creativity and deep thought, interrupting work and family life. While many people say multitasking makes them more productive, research shows otherwise. Many multitaskers actually have more trouble focusing and switching out information, scientists say, and they experience more stress. And scientists are discovering that even after the multid+r, brain’s functioning is still off balance. In other words, this is also your brain on computers.

The Secret to Getting Good Grades

- Grad student said he figured finally it out
  – (Mike Dahlin, now Professor at UT Texas)

- What is the secret?

- Do assigned reading night before, so that get more value from lecture
Constants as Operands

- Many programs use constant numbers
  - E.g., Increment for a loop, index in a data structure
  - >50% of operands are constants in MIPS programs
- So far, would have to be in memory
- To add the constant 4 to register $s3
  \text{lw } \text{t0}, \text{AddrConstant4($s1)} # \text{t0} = \text{constant 4}
  \text{add } \text{s3}, \text{t0, t10} # \text{s3} = \text{t3} + \text{t0} (\text{t0} == 4)
  \text{assuming that } \text{s1} + \text{AddrConstant4 is the memory address of the constant 4}

How translate Java to MIPS?

- A program called a compiler translates from language programmers write (Java) into language computer understands (MIPS) before the program is run on the hardware
  - Act of translating called compiling
- Brings in concepts of
  - Compile-time: what compiler does to program before it is run
  - Run-time: what program does during execution

Guess the other MIPS instructions

- Java operator &: $s1 & s2$ and put result in $s0$?
- Java operator |: $s1 | s2$ and put result in $s0$?
- Java operator <<: $s1 << s2$ and put result in $s0$?
- Java operator >>: $s1 >> s2$ and put result in $s0$?

MIPS Logical Instructions

- Useful to operate on fields of bits within a word
  - e.g., characters within a word (8 bits)
- Operations to pack /unpack bits into words
  - Called logical operations

<table>
<thead>
<tr>
<th>Logical operations</th>
<th>C operators operators</th>
<th>Java</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bit-by-bit AND</td>
<td>&amp;</td>
<td>&amp;</td>
</tr>
<tr>
<td>Bit-by-bit OR</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Shift left</td>
<td>&lt;&lt;</td>
<td>=</td>
</tr>
<tr>
<td>Shift right</td>
<td>&gt;&gt;</td>
<td>srl</td>
</tr>
</tbody>
</table>

Fall 2010 - Lecture #2
Constants as Operands

- Immediate version of many instructions
- And immediate \textit{andi}
- Or immediate \textit{ori}
- Shift immediate \textit{sll, srl}
  - Assumption that normally want to define a constant shift amount
  - Variable shifts using \textit{slv, srlv}
- Why no subtract immediate?

Zero as a Special Constant

- For reasons we shall see, 0 is such a popular constant that MIPS reserves an register to always have the value 0
- $\textit{szero}$

Communicating with People

- Computers were invented to crunch numbers, but soon needed text to interact with people
- Needed representation of English alphabet
- How many symbols do you need?
- How many bits for that many symbols?

What about Non-English Alphabet Languages?

- Unicode (\textit{U}) tried to capture all languages, include long gone ones; below are example alphabets!
**Summary**

- Operands are limited in instruction sets: registers
  - 32 registers in MIPS
- 2nd Design Principle: Smaller is faster
  - Each 32 bits wide ("word")
  - They are 100-500 times faster than word in memory
- Constants also operands
  - 3rd Design Principle: Make the common case fast.
- Variation of instructions: addi, multi, divi,
- Computers do logical operations as well as arithmetic
  - and, or, shift left (sll), shift right (srl), andi, ori,
- Characters: 8-bit ASCII and 16-bit Unicode
- From Java to MIPS instructions called compiling
  - Compile-time (before program run) vs. run-time