CISC
COMPLEX INSTRUCTION SET COMPUTER

PERFORMANCE?
DESIGN TIME?
DESIGN ERRORS?
SINGLE CHIP?
PROGRAM SIZE?

RISC
REDUCED INSTRUCTION SET COMPUTER
CISC
COMPLEX INSTRUCTION SET COMPUTER

PERFORMANCE?
DESIGN TIME?
DESIGN ERRORS?
SINGLE CHIP?
PROGRAM SIZE?

RISC
REDUCED INSTRUCTION SET COMPUTER
1970's Design Principles

(1) *Semiconductor Memory Growth + Microprogramming*  
\[ \Rightarrow \text{"Costs little for richer instruction sets"} \]

(2) "Move software to "firmware" (micocode)"  
\[ \Rightarrow \text{"Faster & more reliable systems"} \]

(3) "Smaller programs are faster programs"  
\[ \Rightarrow \text{"Reduce code size"} \]

(4) "Registers are old fashioned" (hard for compilers)  
\[ \Rightarrow \text{"Memory-to-memory, stacks"} \]

"One’s eyebrows should rise whenever a future architecture is developed with a register oriented instruction set."  
-Glenford J. Myers  
1978
RISC Design Principles

(1) **Keep functions simple unless you have a very good reason not to.**

10% increase in cycle time

\[
\Rightarrow > 10\% \text{ fewer cycles?}
\]

(2) **Microinstructions are same speed as simple instructions.**

(3) **Microcode is not magic.**

(4) **Simple decoding and pipelined execution >> program size**

(5) **Use compiler technology to simplify instrs.**
292R
15 GRADS
EXPLORE
SIMPLE
ARCH.

248
5 GRADS
FORM
RISCI

292X
22 GRADS
MEAD
CONWAY
DESIGN
COMPONENTS

19 GRADS
REDESIGN
COMPONENT

292X
9 GRADS
INTEGRATE
PARTS

JUNE 22
"TAPEOUT"

NEW 20
"FIRST SILICON"

GOOD
SILICON?

SPRING  SUMMER  FALL  WINTER  SPRING  SUMMER
1980    1981
<table>
<thead>
<tr>
<th>NAME NO.</th>
<th>SPONSOR</th>
<th>CREW</th>
<th>ENGINE</th>
<th>VAX POWER</th>
</tr>
</thead>
<tbody>
<tr>
<td>VAXII 780</td>
<td>DEC</td>
<td>PRO:~30x3yr</td>
<td>'78 Bipolar</td>
<td>1</td>
</tr>
<tr>
<td>MC 68000</td>
<td>MOTOROLA</td>
<td>PRO:~20x3yr</td>
<td>'79 NMOS</td>
<td>y3</td>
</tr>
<tr>
<td>8000</td>
<td>Zilog</td>
<td>PRO:~15x3yr</td>
<td>'77 NMOS</td>
<td>y5</td>
</tr>
<tr>
<td>i 8086</td>
<td>INTEL</td>
<td>PRO:~20x2yr</td>
<td>'78 NMOS</td>
<td>y6</td>
</tr>
<tr>
<td>RISC I</td>
<td>U.C. BERK</td>
<td>AM:~10x1yr</td>
<td>'76 NMOS</td>
<td>?</td>
</tr>
</tbody>
</table>

If bet one month salary March '80, what fraction of a VAX would be the goal?
Ave Speed for 11C programs

VAX-11/780

PDP-11/70

BBN C/70

68000

Z8002

RISC I
NOT FINISHED UNTIL WRITTEN O.S. FOR RISC

- "OLD DAYS":
  NEW INSTRUCTION SET
  \Rightarrow NEW O.S.
- MY FAVORITE O.S. MODES
- UNIX ON 68000 😊

WHAT FUNCTIONS IN 68000 NOT IN RISCI?
Time for Berkeley to build microcomputer

Industry

4-7 years

$30,000,000.00

100 man years

Experience

U.C.B.

Berkeley Computer Aided Design

Reduced Instruction Set

2 Berkeley Student Years

Beginner's Luck
INSTRUCTION SET RATIONALE

FORMATS

GOAL WAS KEEPING SAME SIZE

<table>
<thead>
<tr>
<th>OP</th>
<th>DEST</th>
<th>SOURCE1</th>
<th>SOURCE2</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>5</td>
<td>5</td>
<td>14</td>
</tr>
</tbody>
</table>

\[ R_D \leftarrow R_{S1} \quad \text{OP} \{ R_{S2} / \text{CONSTANT} \} \]

ADD ADDC SHLA SHL AND XOR
SUB SUBC SHRA SHR OR

LOAD(BYTE, WORD, LONG) (SIGNED/UNSIGNED)
STORE(BYTE, WORD, LONG)

\[ R_D \leftarrow M \cdot R_{S1} + \{ R_{S2} / \text{CONSTANT} \} \]

CALL RETURN BRANCH

<table>
<thead>
<tr>
<th>OP</th>
<th>DEST</th>
<th>ADDRESS/CONSTANT</th>
</tr>
</thead>
</table>
| 8  | 5    | 19

CALL, RETURN, LOAD IMMEDIATE, BRANCH
PC \leftarrow PC \pm \text{ADDRESS}

NOTE: NO ATTEMPT AT REDUCING CODE SIZE
BUT COULD BE DONE MAKING FETCH MORE COMPLEX
### 11 Small programs

<table>
<thead>
<tr>
<th>Name</th>
<th>VAX</th>
<th>11/70</th>
<th>11/70 rel</th>
<th>RISC</th>
<th>RISC rel</th>
</tr>
</thead>
<tbody>
<tr>
<td>acker</td>
<td>120</td>
<td>130</td>
<td>1.08</td>
<td>208</td>
<td>1.73</td>
</tr>
<tr>
<td>brelse</td>
<td>172</td>
<td>140</td>
<td>0.81</td>
<td>252</td>
<td>1.47</td>
</tr>
<tr>
<td>fun</td>
<td>32</td>
<td>44</td>
<td>1.38</td>
<td>48</td>
<td>1.50</td>
</tr>
<tr>
<td>qsort</td>
<td>436</td>
<td>462</td>
<td>1.06</td>
<td>644</td>
<td>1.48</td>
</tr>
<tr>
<td>stats</td>
<td>284</td>
<td>316</td>
<td>1.11</td>
<td>416</td>
<td>1.46</td>
</tr>
<tr>
<td>sym</td>
<td>204</td>
<td>220</td>
<td>1.08</td>
<td>332</td>
<td>1.63</td>
</tr>
<tr>
<td>towers</td>
<td>100</td>
<td>124</td>
<td>1.24</td>
<td>132</td>
<td>1.32</td>
</tr>
<tr>
<td>spell</td>
<td>2996</td>
<td>3106</td>
<td>1.04</td>
<td>4376</td>
<td>1.46</td>
</tr>
<tr>
<td>sort</td>
<td>4996</td>
<td>4582</td>
<td>0.92</td>
<td>7396</td>
<td>1.48</td>
</tr>
<tr>
<td>finger</td>
<td>6544</td>
<td>6490</td>
<td>0.99</td>
<td>10352</td>
<td>1.58</td>
</tr>
<tr>
<td>puzzle</td>
<td>1668</td>
<td>2004</td>
<td>1.13</td>
<td>2468</td>
<td>1.48</td>
</tr>
<tr>
<td>Ave</td>
<td>1596</td>
<td>1602</td>
<td>1.1</td>
<td>2420</td>
<td>1.5</td>
</tr>
</tbody>
</table>

**Size = Speed?**
IMPLEMENTATION

3-phase/cycle RISC:

CPU organization

GOLD

Manolis H.G. Katevenis
3 SEP 80

REG. FILE
SH
DATA SH
ALU
IMM OFFS
PC
INC
INC
DATA IN
33 pins

CONTROL SIGNALS

Instruction decoder (g13)

Busses 32-bits (actually mixed with logic)