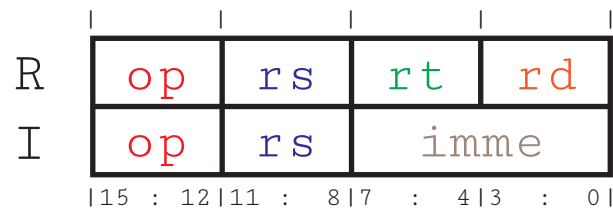


MIA: A 16-bit RISC ISA

<https://github.com/MariaElysse/archel>

MIA Instruction Format

| Insn | Type | Opcode | Operation |
|------|------|--------|--|
| NOP | ? | 0000 | NOP |
| ADD | R | 0001 | $R[\text{rd}] \leftarrow R[\text{rs}] + R[\text{rt}]$ |
| ADDI | I | 0010 | $R[\text{rs}] \leftarrow R[\text{rs}] + \text{sextimme}$ |
| SUB | R | 0011 | $R[\text{rd}] \leftarrow R[\text{rs}] - R[\text{rt}]$ |
| AND | R | 0100 | $R[\text{rd}] \leftarrow R[\text{rs}] \& R[\text{rt}]$ |
| OR | R | 0101 | $R[\text{rd}] \leftarrow R[\text{rs}] R[\text{rt}]$ |
| SLT | R | 0110 | $R[\text{rd}] \leftarrow R[\text{rs}] < R[\text{rt}]$ |
| LW | R | 1000 | $R[\text{rd}] \leftarrow M[\text{rt}]$ |
| SW | R | 1001 | $M[\text{rt}] \leftarrow R[\text{rs}]$ |
| SWI | I | 1010 | $M[\text{imme}] \leftarrow R[\text{rs}]$ |
| BEZI | I | 1100 | $R[\text{rs}] == 0 ? PC \leftarrow \text{imme}$ |
| BNZI | I | 1101 | $R[\text{rs}] != 0 ? PC \leftarrow \text{imme}$ |
| BEZR | R | 1110 | $R[\text{rs}] == 0 ? PC \leftarrow R[\text{rt}]$ |
| BNZR | R | 1111 | $R[\text{rs}] != 0 ? PC \leftarrow R[\text{rt}]$ |



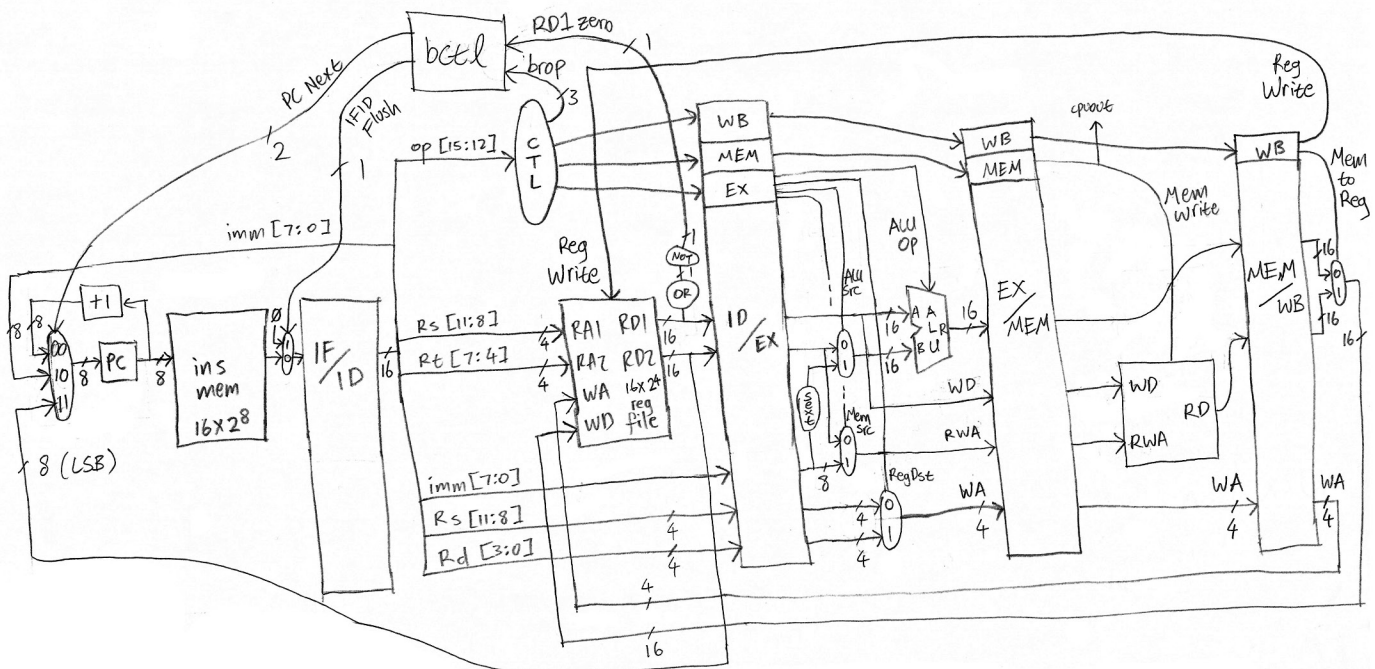
MIA Register Conventions

| Reg | Usage |
|------|-----------------|
| \$0 | Constant zero |
| \$1 | Function result |
| \$2 | Argument 1 |
| \$3 | Argument 2 |
| \$4 | Argument 3 |
| \$5 | Argument 4 |
| \$6 | Temporary |
| \$7 | Temporary |
| \$8 | Temporary |
| \$9 | Temporary |
| \$10 | Temporary |
| \$11 | Temporary |
| \$12 | Temporary |
| \$13 | Temporary |
| \$14 | Stack pointer |
| \$15 | Return address |

ARCHEL Features

Harvard architecture (separate insn & data mems).
5-stage pipeline, for extra efficiency.
Signed arithmetic 16-bit multi-operation ALU.
Fully working stack and main memory.
No compiler, no assembler, no offset addressing,
to build character.

ARCHEL: A MIA Implementation



Quicksort in MIA

Memory Trace

Stack / Call Trace

```

0: addi $14 i+255
1: addi $6 i+51
2: addi $7 i+29
3: addi $8 i+78
4: addi $9 i+87
5: addi $10 i+35
6: addi $11 i+24
7: addi $12 i+43
8: addi $13 i+41
9: swi $6 i+0
10: swi $7 i+1
11: swi $8 i+2
12: swi $9 i+3
13: swi $10 i+4
14: swi $11 i+5
15: swi $12 i+6
16: swi $13 i+7
17: addi $2 i+0
18: addi $3 i+3
19: addi $15 i+152
20: nop
21: nop
22: nop
23: nop
24: quicksort:
25: sw $15 $14 --
26: addi $14 i-1
27: slt $2 $3 $6
28: nop
29: nop
30: bezi $6 :qs-end
31: add $0 $0 $15
32: nop
33: nop
34: addi $15 :qs0
35: bezi $0 :partition
36: qs0:
37: sw $3 $14 --
38: nop
39: nop
40: addi $14 i-1
41: nop
42: nop
43: sw $1 $14 --
44: nop
45: nop
46: addi $14 i-1
47: add $1 $0 $3
48: nop
49: nop
50: addi $3 i-1
51: add $0 $0 $15
52: nop
53: nop
54: addi $15 :qs1
55: bezi $0 :quicksort
56: qs1:
57: addi $14 i+1
58: nop
59: nop
60: lw -- $14 $2
61: nop
62: nop
63: addi $2 i+1
64: addi $14 i+1
65: nop
66: nop
67: lw -- $14 $3
68: add $0 $0 $15
69: nop
70: nop
71: addi $15 :qs2
72: bez $0 :quicksort
73: qs2:
74: qs-end:
75: addi $14 i+1
76: nop
77: nop
78: lw -- $14 $15
79: nop
80: nop
81: nop
82: bezr $0 $15 --
83: nop
84: nop
85: nop
86: nop
87: partition:
88: add $0 $0 $6
89: nop
90: nop
91: lw -- $2 $6
92: add $0 $2 $7
93: add $0 $3 $8
94: nop
95: nop
96: addi $7 i+1
97: add $0 $0 $9
98: notdone:
99: leftmarking:
100: slt $8 $7 $11
101: lw -- $7 $10
102: nop
103: nop
104: slt $6 $10 $12
105: nop
106: nop
107: or $11 $12 $11
108: nop
109: nop
110: bnzi $11 :rightmarking
111: addi $7 i+1
112: bezi $0 :leftmarking
113: rightmarking:
114: slt $8 $7 $11
115: lw -- $8 $10
116: nop
117: nop
118: slt $10 $6 $12
119: nop
120: nop
121: or $11 $12 $11
122: nop
123: nop
124: bnzi $11 :checkdone
125: addi $8 i-1
126: bezi $0 :rightmarking
127: checkdone:
128: slt $8 $7 $9
129: nop
130: nop
131: bnzi $9 :done
132: lw -- $7 $10
133: lw -- $8 $11
134: nop
135: nop
136: sw $10 $8 --
137: sw $11 $7 --
138: bezi $0 :notdone
139: done:
140: lw -- $2 $10
141: lw -- $8 $11
142: nop
143: nop
144: sw $10 $8 --
145: sw $11 $2 --
146: add $0 $8 $1
147: bezr $0 $15 --
148: nop
149: nop
150: nop
151: nop
152: alldone:
153: add $0 $0 $6
154: add $0 $0 $7
155: add $0 $0 $8
156: add $0 $0 $9
157: add $0 $0 $10
158: add $0 $0 $11
159: add $0 $0 $12
160: add $0 $0 $13
161: addi $6 i+0
162: addi $7 i+1
163: addi $8 i+2
164: addi $9 i+3
165: addi $10 i+4
166: addi $11 i+5
167: addi $12 i+6
168: addi $13 i+7
169: lw -- $6 $6
170: lw -- $7 $7
171: lw -- $8 $8
172: lw -- $9 $9
173: lw -- $10 $10
174: lw -- $11 $11
175: lw -- $12 $12
176: lw -- $13 $13
177: halt:
178: bezi $0 :halt

```

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | | |
|------------------|----|-------|----|----|----|----|----|---|--------|--------------|
| 51 | 29 | 78 | 87 | 35 | 24 | 43 | 41 | | + [1] | ret<alldone> |
| p | l | | | | | | r | | + [2] | last=7 |
| | | -> | | | | | <- | | + [3] | pivot=5 |
| 51 | 29 | 41 | 87 | 35 | 24 | 43 | 78 | | 0 -> 1 | (left) |
| p | l | | | | | | r | | + [4] | ret<0:qs1> |
| | | -> | | | | | <- | | + [5] | last=4 |
| 51 | 29 | 41 | 43 | 35 | 24 | 87 | 78 | | + [6] | pivot=0 |
| p | l | | | | | | r | | 1 -> 2 | (left) |
| | | -> | | | | | <- | | + [7] | ret<1:qs1> |
| 51 | 29 | 41 | 43 | 35 | 24 | 87 | 78 | | - [7] | ret<1:qs1> |
| p | l | | | | | | r | | 2 -> 1 | (left) |
| | | -> | | | | | <- | | - [6] | pivot=0 |
| 24 | 29 | 41 | 43 | 35 | 51 | 87 | 78 | | - [5] | last=4 |
| LLLLLLLLLLLLLLLL | PP | RRRRR | | | | | | | 1 -> 2 | (right) |
| 24 | 29 | 41 | 43 | 35 | | | | | + [5] | ret<1:qs2> |
| pr l | | | | | | | | | + [6] | last=6 |
| >< | | | | | | | | | + [7] | pivot=1 |
| 24 | 29 | 41 | 43 | 35 | | | | | 2 -> 3 | (left) |
| PP RRRRRRRRRRRR | | | | | | | | | + [8] | ret<2:qs1> |
| | | | | | | | | | - [8] | ret<2:qs1> |
| 29 | 41 | 43 | 35 | | | | | | 3 -> 2 | (left) |
| pr l | | | | | | | | | - [7] | pivot=1 |
| >< | | | | | | | | | - [6] | last=4 |
| 29 | 41 | 43 | 35 | | | | | | 2 -> 3 | (right) |
| PP RRRRRRRRRR | | | | | | | | | + [6] | ret<2:qs2> |
| | | | | | | | | | + [7] | last=4 |
| 41 | 43 | 35 | | | | | | | + [8] | pivot=3 |
| p l r | | | | | | | | | 3 -> 4 | (left) |
| -> <- | | | | | | | | | + [9] | ret<3:qs1> |
| 41 | 35 | 43 | | | | | | | - [9] | ret<3:qs1> |
| p r l | | | | | | | | | 4 -> 3 | (left) |
| -> <- | | | | | | | | | - [8] | pivot=3 |
| 35 | 41 | 43 | | | | | | | - [7] | last=4 |
| LL PP RR | | | | | | | | | 3 -> 4 | (right) |
| | | | | | | | | | + [7] | ret<3:qs2> |
| | | | | | | | | | - [7] | ret<3:qs2> |
| | | | | | | | | | - [6] | ret<2:qs2> |
| | | | | | | | | | 3 -> 2 | (right) |
| | | | | | | | | | - [5] | ret<1:qs2> |
| | | | | | | | | | 2 -> 1 | (right) |
| | | | | | | | | | - [4] | ret<0:qs1> |
| | | | | | | | | | 1 -> 0 | (left) |
| | | | | | | | | | - [3] | pivot=5 |
| | | | | | | | | | - [2] | last=7 |
| | | | | | | | | | 0 -> 1 | (right) |
| | | | | | | | | | + [2] | ret<0:qs2> |
| | | | | | | | | | + [3] | last=7 |
| | | | | | | | | | + [4] | pivot=7 |
| 24 | 29 | 35 | 41 | 43 | 51 | 78 | 87 | | 1 -> 2 | (left) |
| | | | | | | | | | + [5] | ret<1:qs1> |
| | | | | | | | | | - [5] | ret<1:qs1> |
| | | | | | | | | | 2 -> 1 | (left) |
| | | | | | | | | | - [4] | pivot=7 |
| | | | | | | | | | - [3] | last=7 |
| | | | | | | | | | 1 -> 2 | (right) |
| | | | | | | | | | + [3] | ret<1:qs2> |
| | | | | | | | | | - [3] | ret<1:qs2> |
| | | | | | | | | | 2 -> 1 | (right) |
| | | | | | | | | | - [2] | ret<0:qs2> |
| | | | | | | | | | 1 -> 0 | (right) |
| | | | | | | | | | - [1] | ret<alldone> |