1.

a. `bruh`,
Initial:

| Tag | BHT | Target |
|---|---|---|
| 0x00 | 11 | 0x34 |
| 0x20 | 10 | 0x34 |
| 0x00 | 00 | 0x00 |
| 0x00 | 10 | 0x24 |

After `JAL x2, 36`: will always go to target so no change but PC = 0x24 or 36

| Tag | BHT | Target |
|---|---|---|
| 0x00 | 11 | 0x34 |
| 0x20 | 10 | 0x34 |
| 0x00 | 00 | 0x00 |
| 0x00 | 10 | 0x24 |

After `BEQ x0, x3, END`: will not be equal so no change, PC = 0x28

| Tag | BHT | Target |
|---|---|---|
| 0x00 | 11 | 0x34 |
| 0x20 | 10 | 0x34 |
| 0x00 | 00 | 0x00 |
| 0x00 | 10 | 0x24 |

After `JAL x0, -12`: unconditional jump back to `BEGIN` at 0x1c, save tag as PC (which is 0x28). Change BHT to 11

| Tag | BHT | Target |
|---|---|---|
| 0x00 | 11 | 0x34 |
| 0x20 | 10 | 0x34 |

| | | |
|---|---|---|
| 0x00 | 00 | 0x00 |
| 0x28 | 11 | 0x1c |

After `BEQ x0, x3, END`: we predict not taken when in reality it's taken since x3 is 0. Replace 00 with 10 for BHT 3rd row

| Tag | BHT | Target |
|---|---|---|
| 0x00 | 11 | 0x34 |
| 0x20 | 10 | 0x34 |
| 0x1c | 10 | 0x28 |
| 0x28 | 11 | 0x1c |

b.

| PC | 0x0c | 0x24 | 0x28 | 0x1c |
|---|---|---|---|---|
| Prediction | Taken | Not Taken | Taken | Not Taken |
| Actual Outcome | Taken | Not Taken | Taken | Taken |

c.

| | 1 | 2 | 3 | 4 | 5 | 6 |
|---|---|---|---|---|---|---|
| addi | f | d | e | m | w | |
| addi | | f | d | e | m | w |

2.

## C_2: Rename

```
0x100: lw x1, 4, x1
0x104: xor x3, x0, x1
0x108: sra x1, x3, x2
0x10c: sub x1, x1, x0
```

rename →

0x100: lw p6, 4, p1
0x104: xor p7, p0, p6

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
|---|---|---|---|---|---|---|---|---|---|---|
| lw | f | d | r | | | | | | | |
| xor | f | d | r | | | | | | | |
| sra | | f | d | | | | | | | |
| sub | | f | d | | | | | | | |

| use | OP | Dest Reg | Src Reg 1 | Src 1 Ready | Src Reg 2 | Src 2 Ready | IMM | FU | ROB# |
|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |

| R? | |
|---|---|
| 1 | 1 |
| 2 | 1 |
| 3 | 1 |
| 4 | 1 |
| 5 | 1 |
| 6 | 1 |
| 7 | 1 |
| 8 | 1 |
| 9 | 1 |
| 10 | 1 |

| V | DestReg | "OLD" DestReg | PC | Comp? |
|---|---|---|---|---|
| 0 | | | | |
| 0 | | | | |
| 0 | | | | |
| 0 | | | | |
| 0 | | | | |
| 0 | | | | |
| 0 | | | | |

| RAT | |
|---|---|
| A-reg | PMap |
| x1 | p6 |
| x2 | p2 |
| x3 | p7 |
| x4 | p4 |
| x5 | p5 |

| "Free" Pool |
|---|
| - |
| - |
| p8 |
| p9 |
| p10 |

| FU R? | |
|---|---|
| ALU-1 | 1 |
| ALU-2 | 1 |
| MEM | 1 |

## C_3: Dispatch and Rename

```
0x100: lw x1, 4, x1
0x104: xor x3, x0, x1
0x108: sra x1, x3, x2
0x10c: sub x1, x1, x0
```

rename →

0x100: lw p6, 4, p1
0x104: xor p7, p0, p6
0x108: sra p8, p7, p2
0x10c: sub p9, p8, x0

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
|---|---|---|---|---|---|---|---|---|---|---|
| lw | f | d | r | di | | | | | | |
| xor | f | d | r | di | | | | | | |
| sra | | f | d | r | | | | | | |
| sub | | f | d | r | | | | | | |

| use | OP | Dest Reg | Src Reg 1 | Src 1 Ready | Src Reg 2 | Src 2 Ready | IMM | FU | ROB# |
|---|---|---|---|---|---|---|---|---|---|
| 1 | lw | p6 | p1 | 1 | - | - | 4 | 2 | 0 |
| 1 | xor | p7 | p0 | 1 | p6 | 0 | - | 0 | 1 |
| | | | | | | | | | |
| | | | | | | | | | |

| R? | |
|---|---|
| 1 | 1 |
| 2 | 1 |
| 3 | 1 |
| 4 | 1 |
| 5 | 1 |
| 6 | 0 |
| 7 | 0 |
| 8 | 1 |
| 9 | 1 |
| 10 | 1 |

| V | DestReg | "OLD" DestReg | PC | Comp? |
|---|---|---|---|---|
| 1 | p6 | p1 | 0x100 | 0 |
| 1 | p7 | p3 | 0x104 | 0 |
| 0 | | | | |
| 0 | | | | |
| 0 | | | | |
| 0 | | | | |
| 0 | | | | |

| RAT | |
|---|---|
| A-reg | PMap |
| x1 | 9(8,1) |
| x2 | p2 |
| x3 | p7 |
| x4 | p4 |
| x5 | p5 |

| "Free" Pool |
|---|
| - |
| - |
| p8 |
| p9 |
| p10 |

| FU R? | |
|---|---|
| ALU-1 | 1 |
| ALU-2 | 1 |
| MEM | 1 |

## C_4: Issue , Dispatch

```
0x100: lw x1, 4, x1
0x104: xor x3, x0, x1
0x108: add x1, x3, x0
0x10c: sub x1, x2, x0

* LW takes two cycles.
```

rename →

0x100: lw p6, 4, p1
0x104: xor p7, p0, p6
0x108: add p8, p7, p0
0x10c: sub p9, p2, p0

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
|---|---|---|---|---|---|---|---|---|---|---|
| lw | F | D | R | Di | I | | | | | |
| xor | F | D | R | Di | Di | | | | | |
| add | | F | D | R | Di | | | | | |
| sub | | F | D | R | Di | | | | | |

| R? | |
|---|---|
| 1 | 1 |
| 2 | 1 |
| 3 | 1 |
| 4 | 1 |
| 5 | 1 |
| 6 | 0 |
| 7 | 0 |
| 8 | 0 |
| 9 | 0 |
| 10 | 1 |

| V | DestReg | "OLD" DestReg | PC | Comp? |
|---|---|---|---|---|
| 1 | p6 | p1 | 0x100 | 0 |
| 1 | p7 | p3 | 0x104 | 0 |
| 1 | p8 | p6 | 0x108 | 0 |
| 1 | p9 | p8 | 0x10c | 0 |
| 0 | | | | |
| 0 | | | | |
| 0 | | | | |

| use | OP | Dest Reg | Src Reg 1 | Src 1 Ready | Src Reg 2 | Src 2 Ready | IMM | FU | ROB# |
|---|---|---|---|---|---|---|---|---|---|
| 0 | | | | | | | | | |
| 1 | xor | p7 | p0 | 1 | p6 | 0 | - | 0 | 1 |
| 1 | add | p8 | p7 | 0 | p0 | 1 | - | 1 | 2 |
| 1 | sub | p9 | p2 | 1 | p0 | 1 | - | 0 | 3 |

| RAT | |
|---|---|
| A-reg | PMap |
| x1 | 9(8,6,1) |
| x2 | p2 |
| x3 | 7(3) |
| x4 | p4 |
| x5 | p5 |

| "Free" Pool |
|---|
| - |
| - |
| - |
| - |
| p10 |

| FU R? | |
|---|---|
| ALU-1 | 1 |
| ALU-2 | 1 |
| MEM | 0 |

## C_5: Finish?

```
0x100: lw x1, 4, x1
0x104: xor x3, x0, x1
0x108: sra x1, x3, x2
0x10c: sub x1, x1, x0
```

rename →

0x100: lw p6, 4, p1
0x104: xor p7, p0, p6
0x108: sra p8, p7, p2
0x10c: sub p9, p8, x0

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
|---|---|---|---|---|---|---|---|---|---|---|
| lw | f | d | r | di | | | | | | |
| xor | f | d | r | di | | | | | | |
| sra | | f | d | r | | | | | | |
| sub | | f | d | r | | | | | | |

| R? | |
|---|---|
| 1 | 1 |
| 2 | 1 |
| 3 | 1 |
| 4 | 1 |
| 5 | 1 |
| 6 | 0 |
| 7 | 0 |
| 8 | 1 |
| 9 | 1 |
| 10 | 1 |

| V | DestReg | "OLD" DestReg | PC | Comp? |
|---|---|---|---|---|
| 1 | p6 | p1 | 0x100 | 0 |
| 1 | p7 | p3 | 0x104 | 0 |
| 0 | | | | |
| 0 | | | | |
| 0 | | | | |
| 0 | | | | |
| 0 | | | | |

| use | OP | Dest Reg | Src Reg 1 | Src 1 Ready | Src Reg 2 | Src 2 Ready | IMM | FU | ROB# |
|---|---|---|---|---|---|---|---|---|---|
| 1 | lw | p6 | p1 | 1 | - | - | 4 | 2 | 0 |
| 1 | xor | p7 | p0 | 1 | p6 | 0 | - | 0 | 1 |
| | | | | | | | | | |
| | | | | | | | | | |

| RAT | |
|---|---|
| A-reg | PMap |
| x1 | 9(8,1) |
| x2 | p2 |
| x3 | p7 |
| x4 | p4 |
| x5 | p5 |

| "Free" Pool |
|---|
| - |
| - |
| p8 |
| p9 |
| p10 |

| FU R? | |
|---|---|
| ALU-1 | 1 |
| ALU-2 | 1 |
| MEM | 1 |