

NYCU Computer Organization HW 8

1. In this exercise we compare the performance of 1-issue and 2-issue processors, taking into account program transformations that can be made to optimize for 2-issue execution. Problems in this exercise refer to the following loop (written in C):

```
for (i = 0; i != j; i += 2)
    b[i] = a[i] - a[i + 1];
```

A compiler doing little or no optimization might produce the following RISC-V assembly code:

```

        addi    x12, x0, 0
        jal     ENT
TOP:    slli    x5, x12, 3
        add     x6, x10, x5
        lw      x7, 0(x6)
        lw      x29, 8(x6)
        sub     x30, x7, x29
        add     x31, x11, x5
        sw      x30, 0(x31)
        addi    x12, x12, 2
ENT:    bne     x12, x13, TOP
```

The code above uses the following registers

i	j	a	b	Temporary Values
x12	x13	x10	x11	x5 – x7, x29 – x31

Assume the two-issue, statically scheduled processor for this exercise has the following properties:

- a. One instruction must be a memory operation; the other must be an arithmetic/logic instruction or a branch
 - b. The processor has all possible forwarding paths between stages (including paths to the ID stage for branch resolution)
 - c. The processor has perfect branch prediction
 - d. Two instructions may not issue together in a packet if one depends on the other
 - e. If a stall is necessary, both instructions in the issue packet must stall.
- (1) Draw a pipeline diagram showing how RISC-V code given above executes on the two-issue processor. Assume that the loop exits after two iterations.
 - (2) What is the speedup of going from a one-issue to a two-issue processor? (Assume the loop runs thousands of iterations)