

# LAB 01

## Exercise

---

2023.03.08

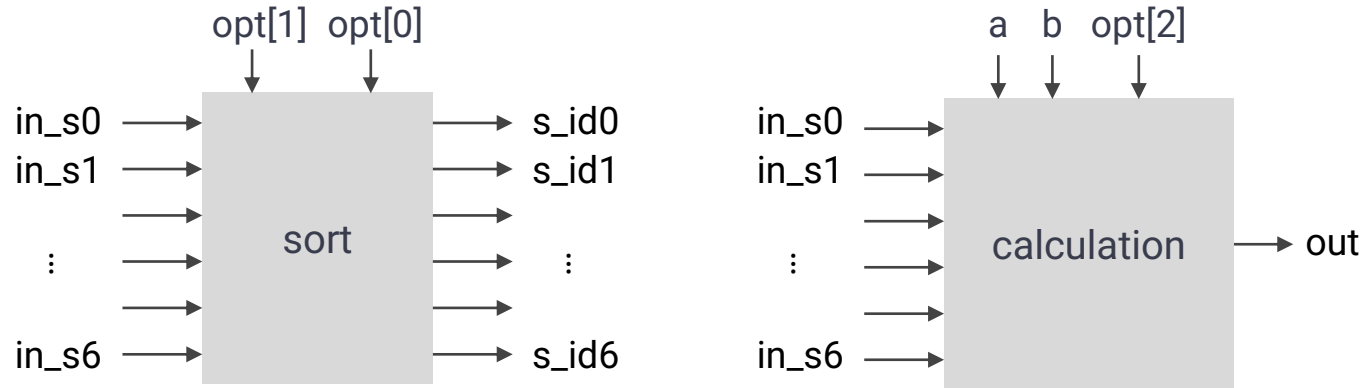


Presenter: Tzu-Yun, Yen

# Outline

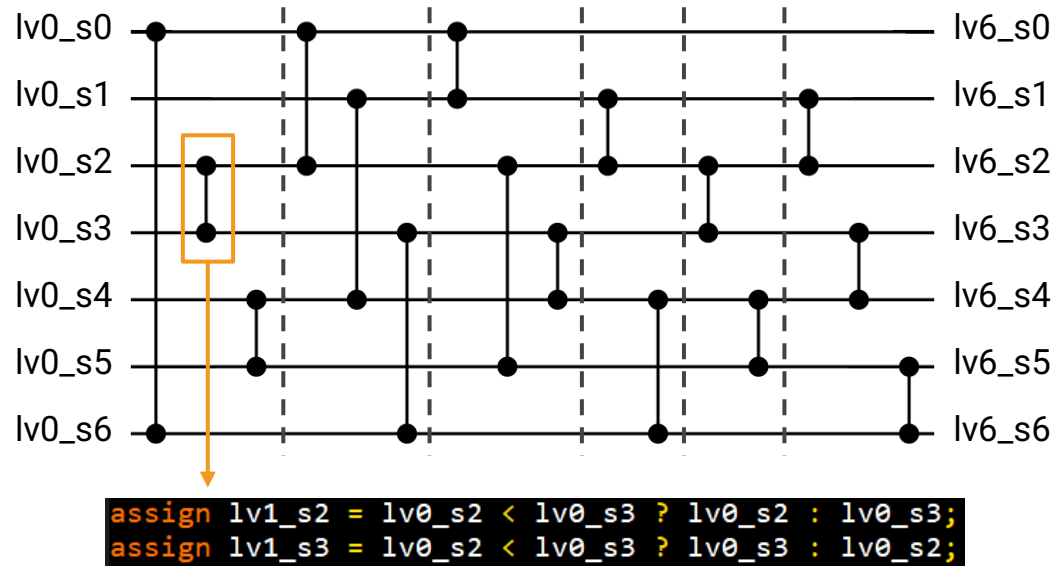
- Architecture Overview
- Sort
- Signed/Unsigned
- Ascending/Descending
- Calculation

# Architecture Overview



# Sort

- 7 elements → 16 comparators



# Signed/Unsigned

- `opt[0] = 1` : regarded as signed value
- `opt[0] = 0` : regarded as unsigned value

<u>signed</u>		<u>unsigned</u>
3 → 011		111 → 7
2 → 010		110 → 6
1 → 001		101 → 5
0 → 000		100 → 4
-1 → 111	→	011 → 3
-2 → 110		010 → 2
-3 → 101		001 → 1
-4 → 100		000 → 0

5-bit signed comparators

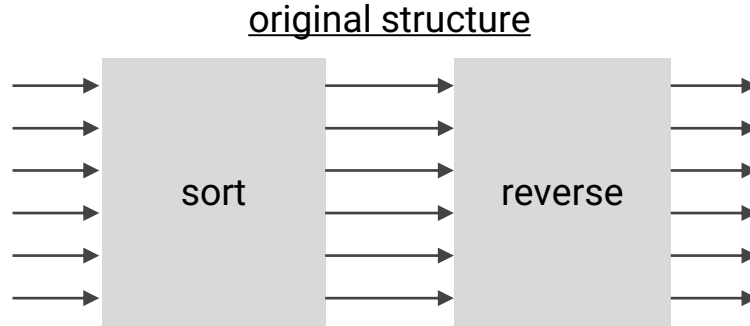


4-bit unsigned comparators

```
assign lv0_s0[6:3] = opt[0] ? {~in_lv0_s0[3], in_lv0_s0[2:0]} : in_lv0_s0;
```

# Ascending/Descending

- $\text{opt}[1] = 1$  : descending order
- $\text{opt}[1] = 0$  : ascending order



complex control signals are needed to handle  
the same score problem

# Ascending/Descending

- `opt[1] = 1` : descending order
- `opt[1] = 0` : ascending order

<u>descending</u>		<u>ascending</u>
7 → 111		000 → 0
6 → 110		001 → 1
5 → 101		010 → 2
4 → 100	➔	011 → 3
3 → 011		100 → 4
2 → 010		101 → 5
1 → 001		110 → 6
0 → 000		111 → 7

arrange scores in descending order  
= arrange **complement of scores** in  
ascending order

```
assign in_lv0_s0 = opt[1] ? ~in_s0 : in_s0;
```

# Merge Scores & IDs

## original

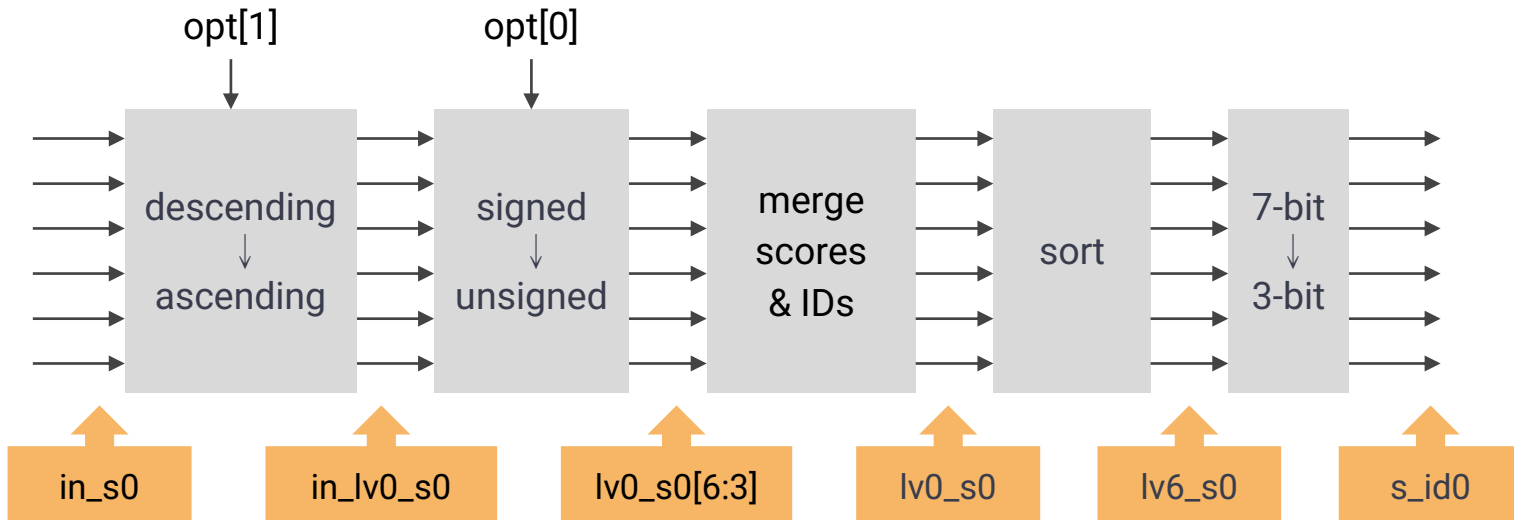
```
assign bigger_score = in_s0 > in_s1;  
assign bigger_id = bigger_score || (in_s0 == in_s1) && (in_id0 < in_id1);  
  
assign out_s0 = bigger_score ? in_s0 : in_s1;  
assign out_s1 = bigger_score ? in_s1 : in_s0;  
assign out_id0 = bigger_id ? in_id0 : in_id1;  
assign out_id1 = bigger_id ? in_id1 : in_id0;
```

## optimized

```
assign lv0_s0[2:0] = 3'd0; ← ID  
assign lv0_s0[6:3] = opt[0] ? {~in_lv0_s0[3], in_lv0_s0[2:0]} : in_lv0_s0; ← Score  
assign lv1_s0 = lv0_s0 < lv0_s6 ? lv0_s0 : lv0_s6;  
assign lv1_s6 = lv0_s0 < lv0_s6 ? lv0_s6 : lv0_s0;
```



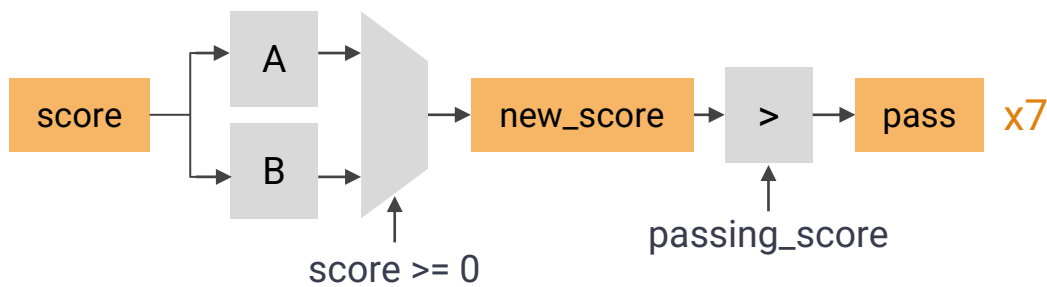
# Optimized Sort



# Calculation

- passing score =  $\mu - a$
- passing conditions (original)
  - A.  $\text{score} \geq 0 \rightarrow (\text{score} * (a + 1) + b) > \text{passing\_score}$
  - B.  $\text{score} < 0 \rightarrow (\text{score} / (a + 1) + b) > \text{passing\_score}$

7 comparators  
7 multipliers  
7 dividers  
14+1 adders



# Calculation

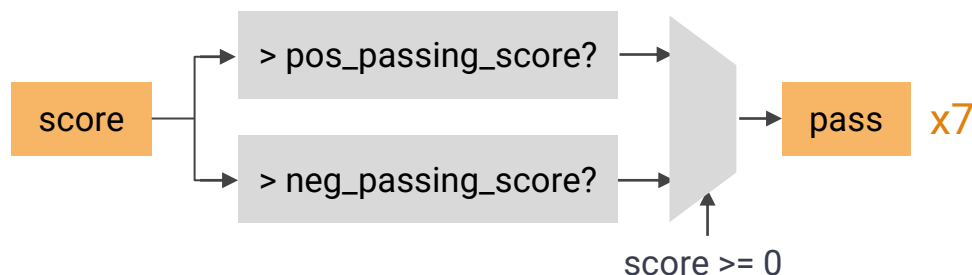
- passing score =  $\mu - a$
- passing conditions (optimized)
  - A.  $\text{score} \geq 0 \rightarrow \text{score} > (\text{passing\_score} - b + a) / (a + 1)$
  - B.  $\text{score} < 0 \rightarrow \text{score} > (\text{passing\_score} - b) * (a + 1) - a$

14 comparators

1 multipliers

1 dividers

4 adders



# Area Report

```
Number of ports:                60
Number of nets:                 772
Number of cells:               699
Number of combinational cells: 698
Number of sequential cells:    0
Number of macros/black boxes:  0
Number of buf/inv:            146
Number of references:          37

Combinational area:             13801.233805
Buf/Inv area:                  1456.963253
Noncombinational area:         0.000000
Macro/Black Box area:         0.000000
Net Interconnect area:         undefined (No wire load specified)

Total cell area:               13801.233805
Total area:                    undefined
```



# The End

---

Thanks for listening :)

