# HW #5. Simple Buffer Insertion

(Due: 5/31/2017)

In this program, you need to insert buffers in front of nodes to reduce delay time.

## 1. Goal:

(1) Read a wire resistance and node capacitance sequences of a non-fork (one-line) circuit and a buffer library.
(2) Add buffers in front of nodes to make **delay < max allowed delay**.
(3) Your solution should **minimize the cost** and **cannot violate the max allowed delay.** (The cost and delay calculation are mentioned in Section 3.)
(4) If the costs of valid insertion combination are the same, please output the one with minimal delay.

## 2. Input and Output Format:

### 2.1 Input file: input.txt

testcase_num
max_allowed_delay //case1

node_num //Circuit Information
1 $\Omega_1$ $C_1$
2 $\Omega_2$ $C_2$
3 $\Omega_3$ $C_3$
…

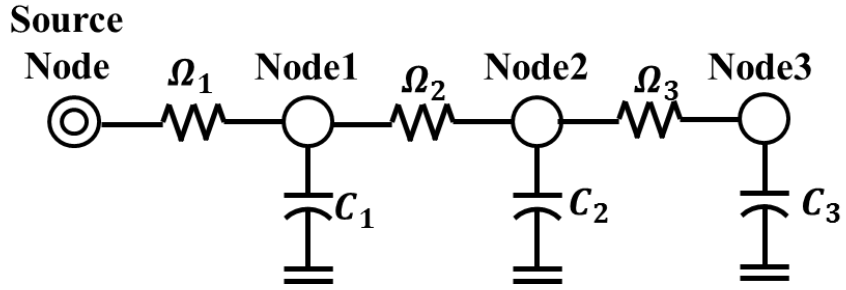
buf_num //Buffer Library
1 $\Omega_{buf1}$ $C_{buf1}$ $Cost_{buf1}$
2 $\Omega_{buf2}$ $C_{buf2}$ $Cost_{buf2}$

… //case 2……

(1) Circuit Information

First line is the number of nodes. Then, each line includes node_id, the resistance of wire i ($\Omega_i$) and the capacitance of node i ($C_i$).

The circuit consists of those nodes in order. (Source node -> node 1 -> node 2 -> node 3 …)




(2) Buffer Library

First line is the number of kinds of buffers in buffer library. Each line contains buffer_id, the resistance of buffer k ($\Omega_{buf\ k}$), the capacitance of buffer k ($C_{buf\ k}$) and the cost of buffer k ($Cost_{buf\ k}$).

You need to choose which kind of buffer to insert. Each kind of buffer can be chosen repeatedly.

**2.2 Output file: output.txt**

One case is showed in one line, and different cases are splited by newline.

(1) Solvable

You should list insertion position (node_id) and inserted buffer_id in one line, splited by space and sorted by node_id.

e.g. Insert buffer 2 on node 1 and buffer 1 on node 3

```
1 2 3 1                                  //node_id buf_id node_id buf_id …
```

(2) No Solution

If there is no valid combination of inserting buffers to make delay < max_allowed_delay, you should write “NO SOLUTION” in output.txt.

e.g.

```
NO SOLUTION
```

## 3. Relative Formula

### 3.1 Cost Calculation

$$cost = \sum_{buffer\ k\ you\ insert} Cost_{bufk}$$

e.g. if you insert buffer 1 on node 1 and buffer 2 on node 3, your cost is $Cost_{buf1} + Cost_{buf2}$.

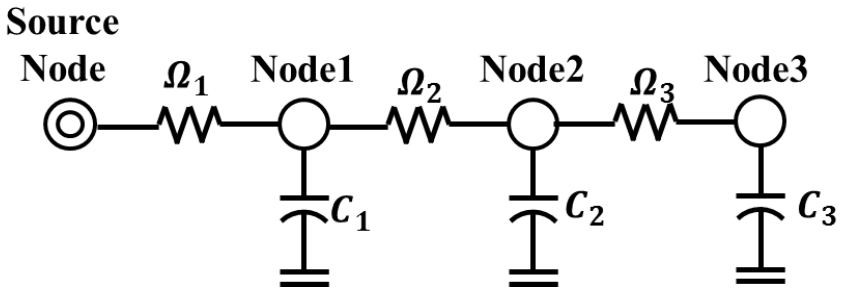
### 3.2 Delay Calculation

The delay from source node to last node is shown below:

$$delay = \sum_{node\ i\ \in\ circuit} \Omega_i * \ dscap_i$$

$dscap_i$ is the downstream capacitance which means accumulated capacitance form last node to node i.

(1) Without buffers



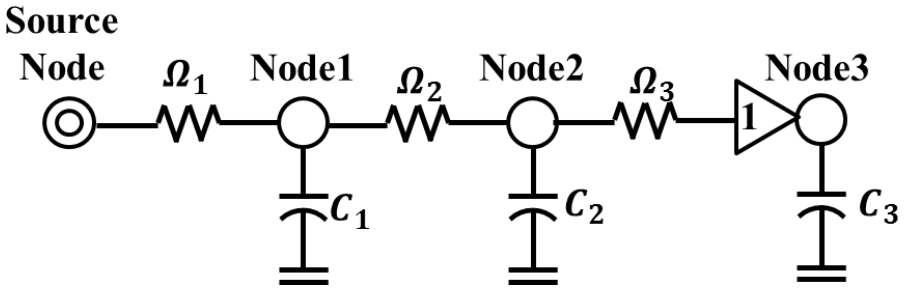

delay = $\Omega_1 * (C_1 + C_2 + C_3)$ +
$\Omega_2 * (C_2 + C_3)$ +
$\Omega_3 * C_3$

(2) Insert buffers

If you insert a buffer k on node i, **the $dscap_i$ will be replaced by the capacitance of inserted buffer ($C_{bufk}$)**. But the delay should plus $\Omega_{bufk} * original\ dscap_i$.

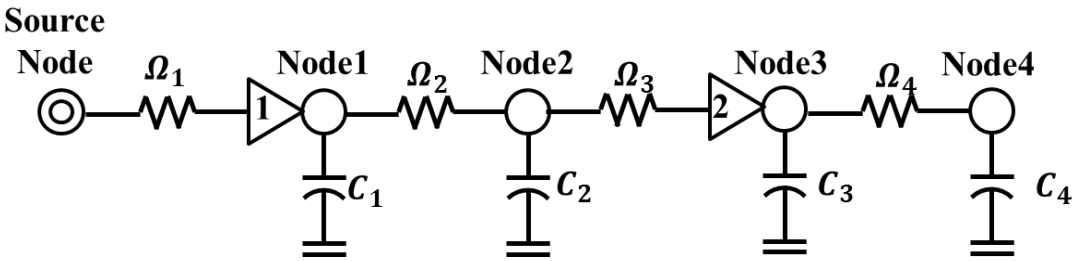
e.g.

a) Insert buffer 1 on node 3




$$\text{delay} = \Omega_1 * (C_1 + C_2 + C_{buf1}) + \Omega_2 * (C_2 + C_{buf1}) + \Omega_3 * C_{buf1} + \Omega_{buf1} * C_3$$

b) Insert buffer 1 on node 1 and buffer 2 on node 3




$$\text{delay} = \Omega_1 * C_{buf1} + \Omega_{buf1} * (C_1 + C_2 + C_{buf2}) + \Omega_2 * (C_2 + C_{buf2}) + \Omega_3 * C_{buf2} + \Omega_{buf2} * (C_3 + C_4) + \Omega_4 * C_4$$

### 3.3 Verify

We provide a "Verify" program to show delay and cost of your solution.

- Usage: ./Verify [input_filename] [output_filename]
- If it cannot be executed, use the following command:
  $ chmod +x Verify

## 4. Example

input.txt

```
2                               // 2 cases
2000                            // case 1

3
1 44 10.61
2 10 20
3 10 15

2
1 6.12 3.5 11.5
2 44 0.42 1.03

500                             // case 2

3
1 44 10.61
2 10 20
3 10 15

2
1 6.12 3.5 11.5
2 44 0.42 1.03
```

output.txt

```
1 1                             // case 1 result
NO SOLUTION                     // case 2 result

```

$ ./Verify input.txt output.txt

```
-----------------------------------------
[vrf] Case 1
Node 3
        Delay: 150
        Dscap: 15
        Cost: 0
Node 2
        Delay: 500
```

```
        Dscap: 35
        Cost: 0
Node 1
        Delay: 933.133
        Dscap: 3.5
        Cost: 11.5
======== solution info ========
Delay: 933.133
Slack: 1066.87                  //Slack = Max_Allowed_Delay - Delay
Cost: 11.5
----------------------------------------
----------------------------------------
[vrf] Case 2
NO SOLUTION
Node 3
        Delay: 150
        Dscap: 15
        Cost: 0
Node 2
        Delay: 500
        Dscap: 35
        Cost: 0
Node 1
        Delay: 2506.84
        Dscap: 45.61
        Cost: 0
======== solution info ========
Delay: 2506.84
Slack: -2006.84                 //Slack = Max_Allowed_Delay - Delay
Cost: 0
----------------------------------------
```

## 5. Discussion Board

If you have any problem about this homework, please post your question in our discussion board. You are encouraged to post the question on discussion board as others might have the similar question.