

# REALTEK Interview Presentation

---

- **Interviewee :** 陳文遠
- **Position :** Audio系統設計工程師 (預聘\_110年度應屆畢業)
- **Date :** October 19, 2020

# INTRODUCTION

# ABOUT

---

## EDUCATION

- **National Tsing Hua University (GPA 4.08)**  
Institute of Communications Engineering (MS)  
Hsinchu, Taiwan  
Sep. 2019 – Present
- **Feng Chia University**  
Department of Communications Engineering (BS)  
Taichung, Taiwan  
Sep. 2014 – Jan. 2019

## ACADEMIC EXPERIENCE

1. **2018 National University Competition of Python**  
Certified
2. **2017 IMP 2017 Conference**  
Publish
3. **2017 Independent Study Competition**  
Honorable Mention Award

# SKILLS

---

WHAT I HAVE DONE	
Skills	Tools
Website and Server Develop	Node.js(Server), HTML, Javascript, CSS
Computer Vision	C++(or Python) with OpenCV
Network Programming	C language
Embedded System Development	C(or Python) with Raspberry PI
Communication Simulation	Matlab
Blockchain Technology	Go lang with Hyperledger
Linux (UNIX) Operation	Basic operation

PROJECT

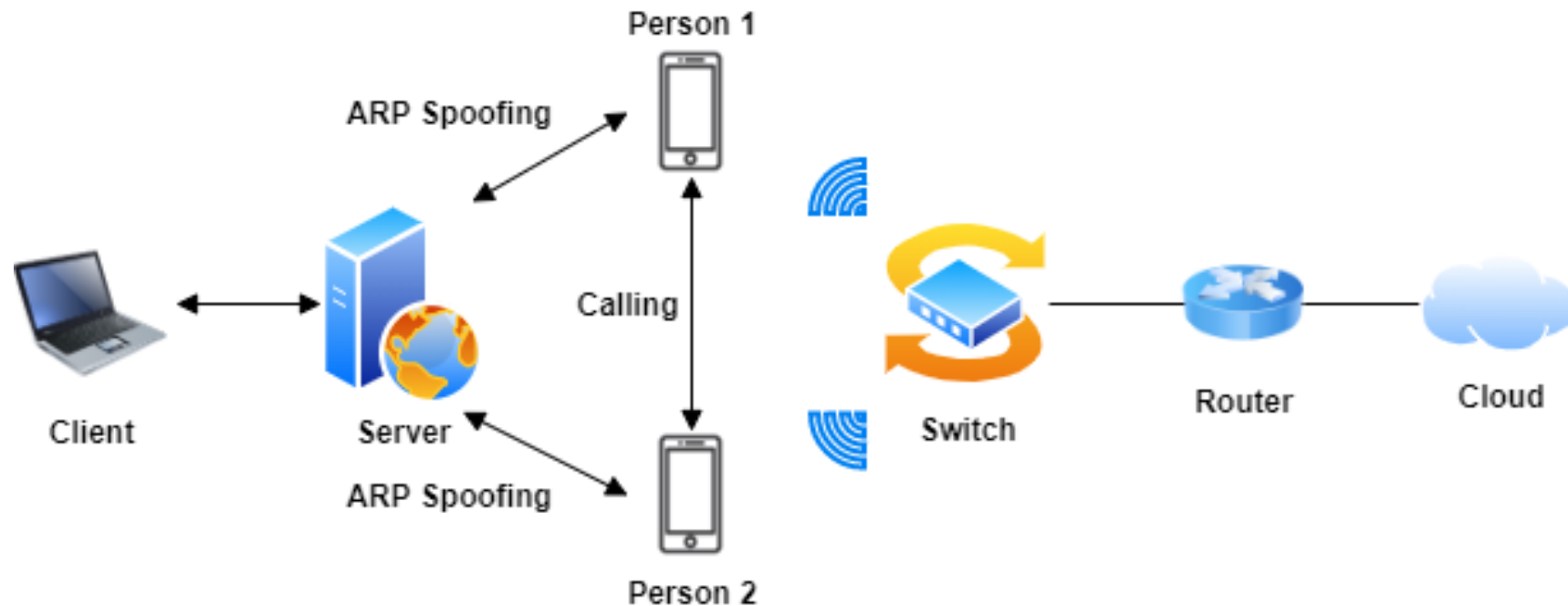
## **(1) VoIP Phone Call Monitoring Platform**

# (1) VoIP Phone Call Monitoring Platform

## PURPOSE

- ❑ Provide VoIP phone call monitoring and management service to user.

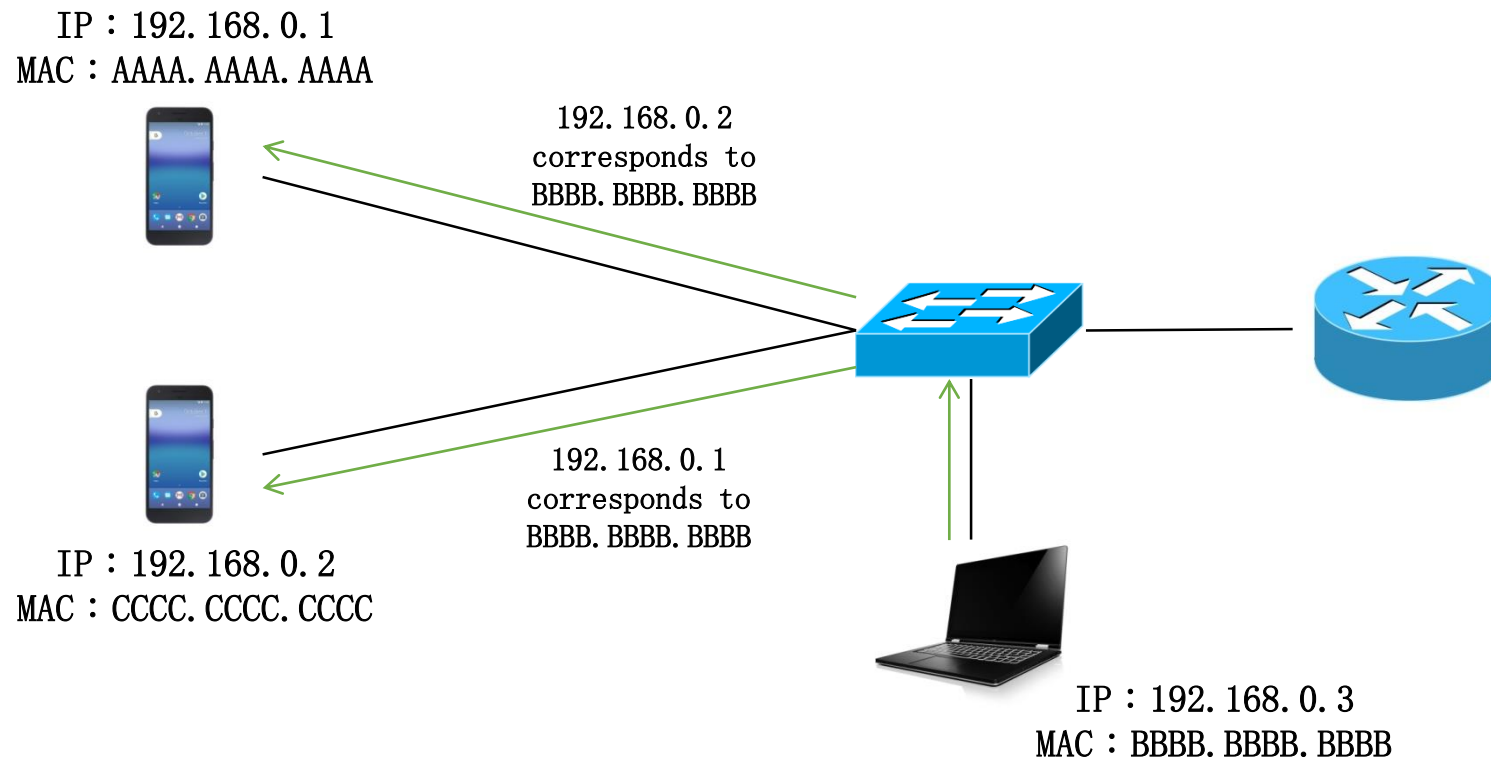
## ARCHITECTURE



# (1) VoIP Phone Call Monitoring Platform (Cont.)

## METHOD

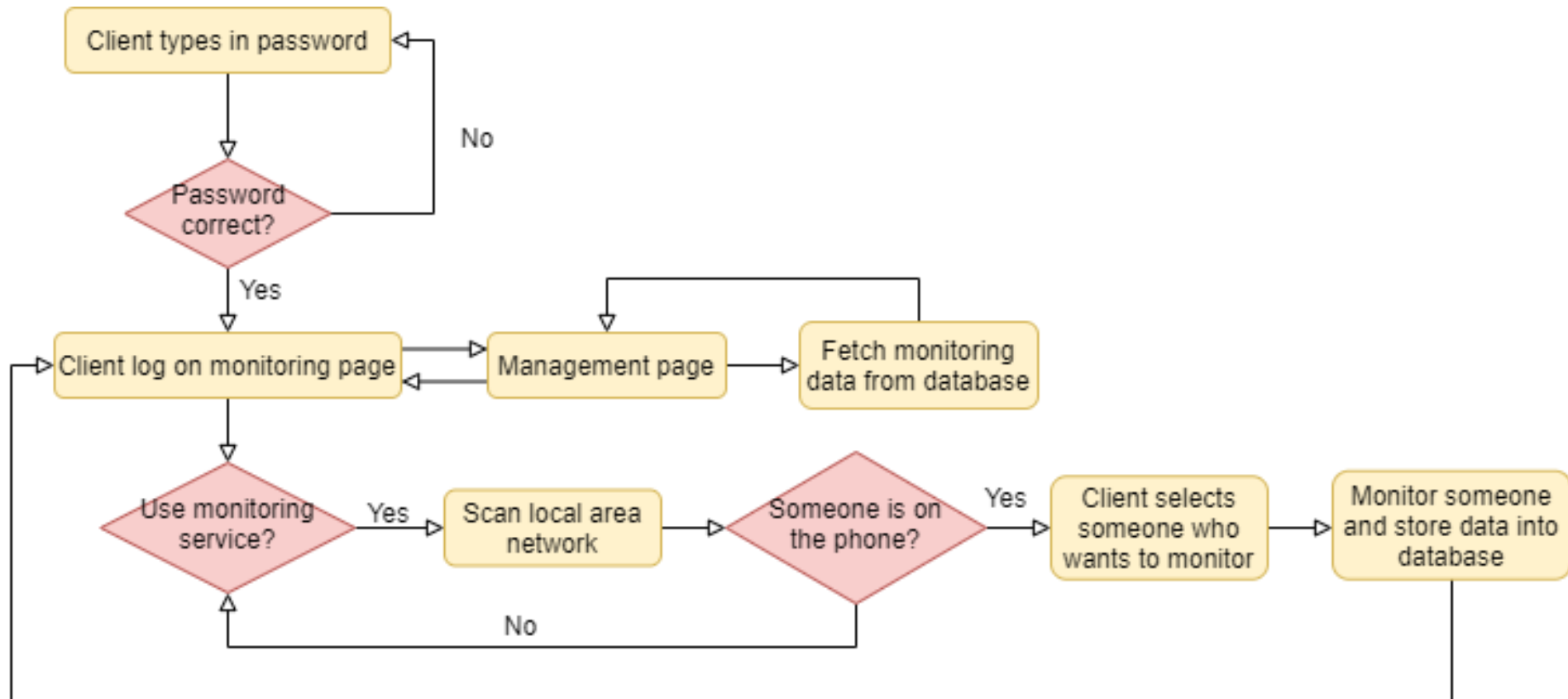
### □ ARP Spoofing





# (1) VoIP Phone Call Monitoring Platform (Cont.)

## FLOWCHART



# (1) VoIP Phone Call Monitoring Platform (Cont.)

## RESULTS

監控平台登入

使用者名稱

密碼

 認證

提示！請通過驗證以進入監控平台

Fig 1. Login page

網路電話監控平台

資料庫 登出 使用者D0349119

IP 192.168.43.1/24 PORT 5060 掃描網域

網域掃描結果

- ☒ 192.168.43.1:5060
- ☒ 192.168.43.236:5060

開始監聽! 停止監聽! 下載音訊

語音辨識結果

哈囉你好我正在做專題測試聽到請回答

辨識完畢 下載辨識文本

選擇語言

中文 (台灣)

Fig 2. Monitoring page

# (1) VoIP Phone Call Monitoring Platform (Cont.)

## RESULTS

資料庫平台

監控平台

登出

使用者D0349119

可刪除資訊

監控的年月日及時間

下載音訊檔及辨識文本

刪除

監控日期

下載檔案

✕

2017-08-24\_08:09:29

下載音訊

下載文本

刪除

監控日期

下載檔案

✕

2017-08-24\_08:20:08

下載音訊

下載文本

刪除

監控日期

下載檔案

✕

2017-08-24\_08:33:30

下載音訊

下載文本

刪除

監控日期

下載檔案

✕

2017-08-26\_07:49:08

下載音訊

下載文本

刪除

監控日期

下載檔案

✕

2017-09-10\_13:00:10

下載音訊

下載文本

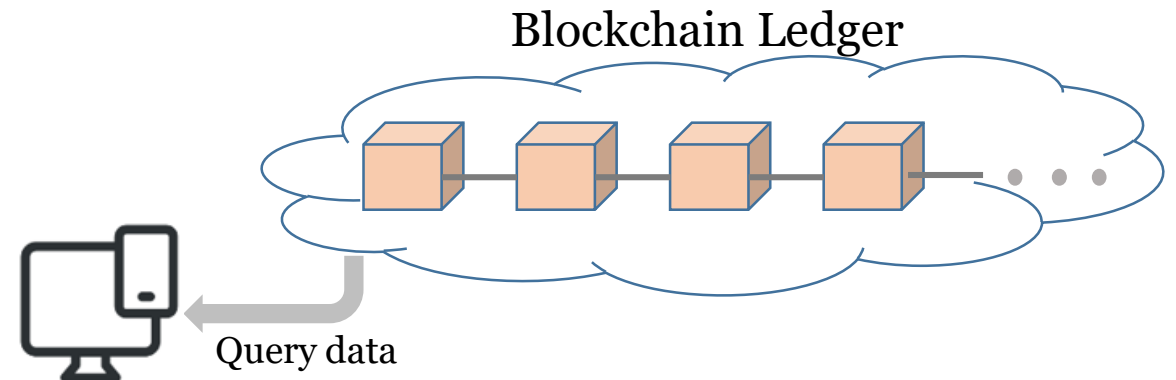
Fig 3. Management page

## **(2) Blockchain Traceability System with HyperQL**

## (2) Blockchain Traceability System with HyperQL

### What's Blockchain?

- ❑ Blockchain is a kind of decentralized database.
- ❑ **Advantage of blockchain technology :**  
Blockchain has immutable property that keep data secure.
- ❑ **Disadvantage of blockchain technology :**  
Query data from the blockchain ledger is complex and slow.

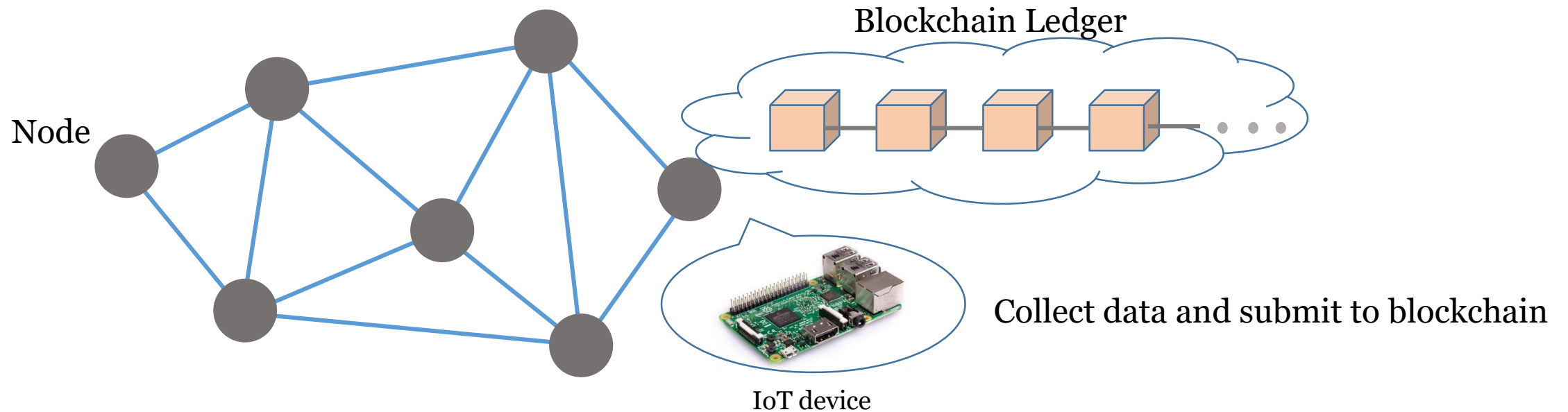


## (2) Blockchain Traceability System with HyperQL (Cont.)

### PURPOSE

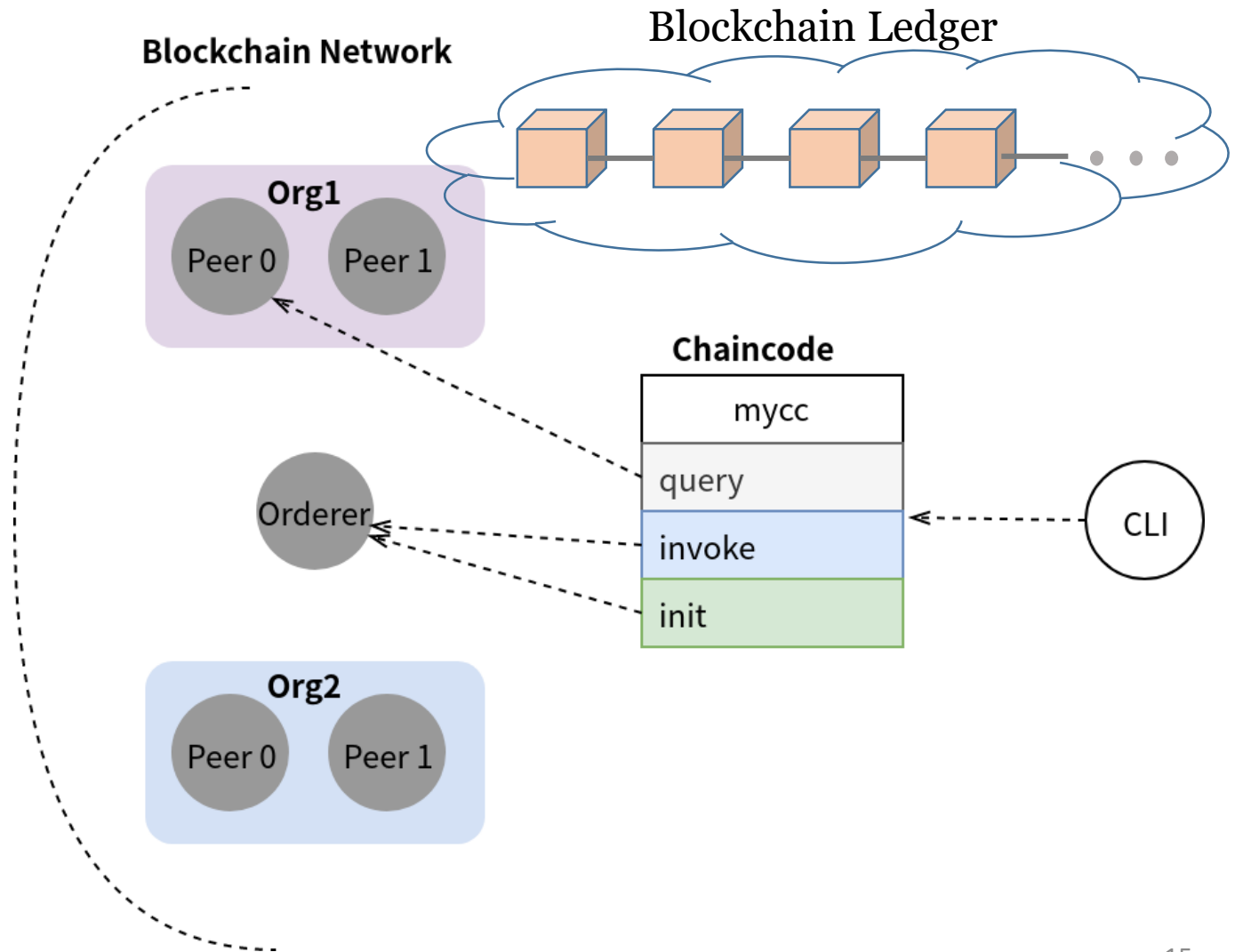
- ❑ Create a blockchain-based IoT system and propose an architecture, named “HyperQL”, to speed up the query speed.

### ARCHITECTURE



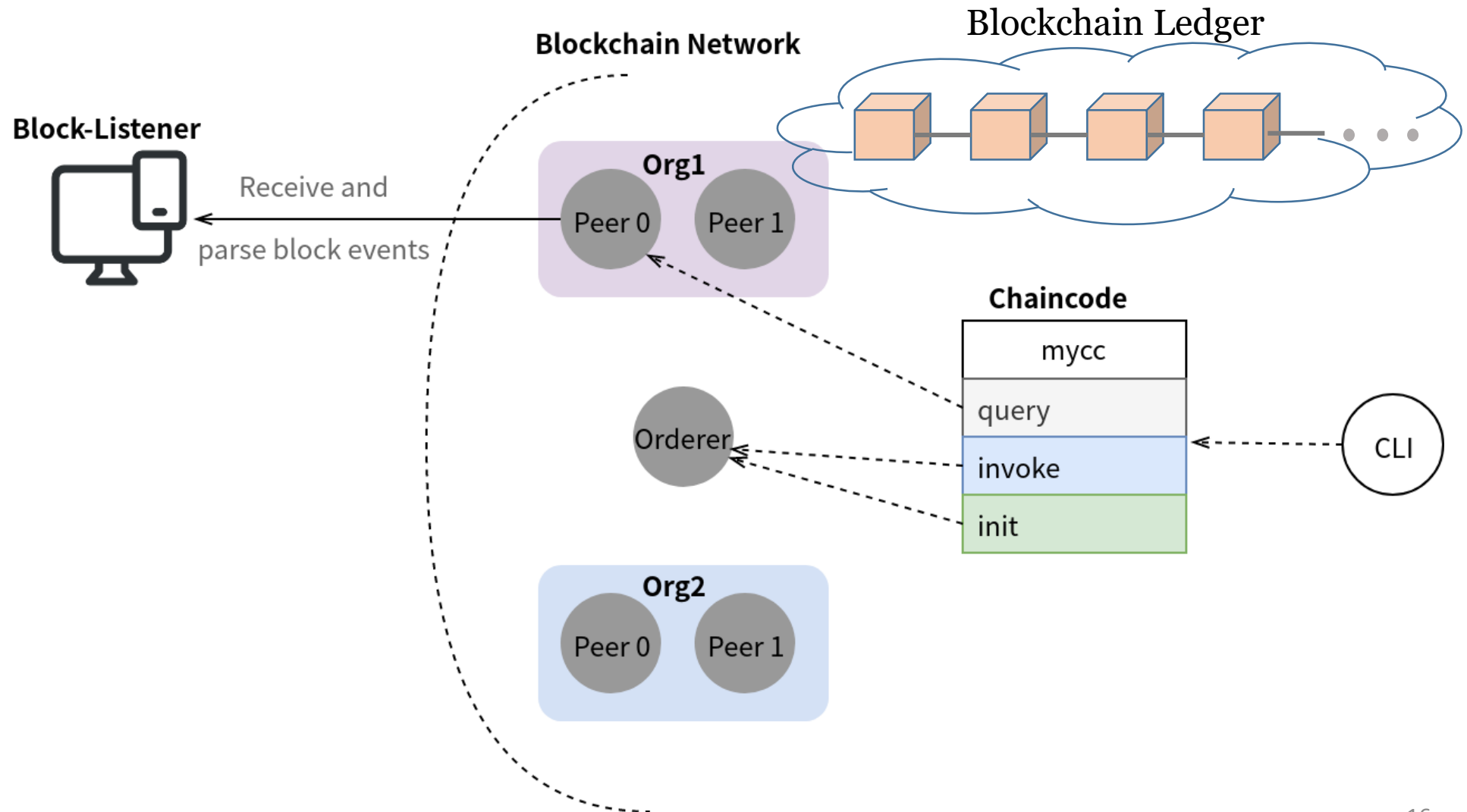
## (2) Blockchain Traceability System with HyperQL (Cont.)

### METHOD



## (2) Blockchain Traceability System with HyperQL (Cont.)

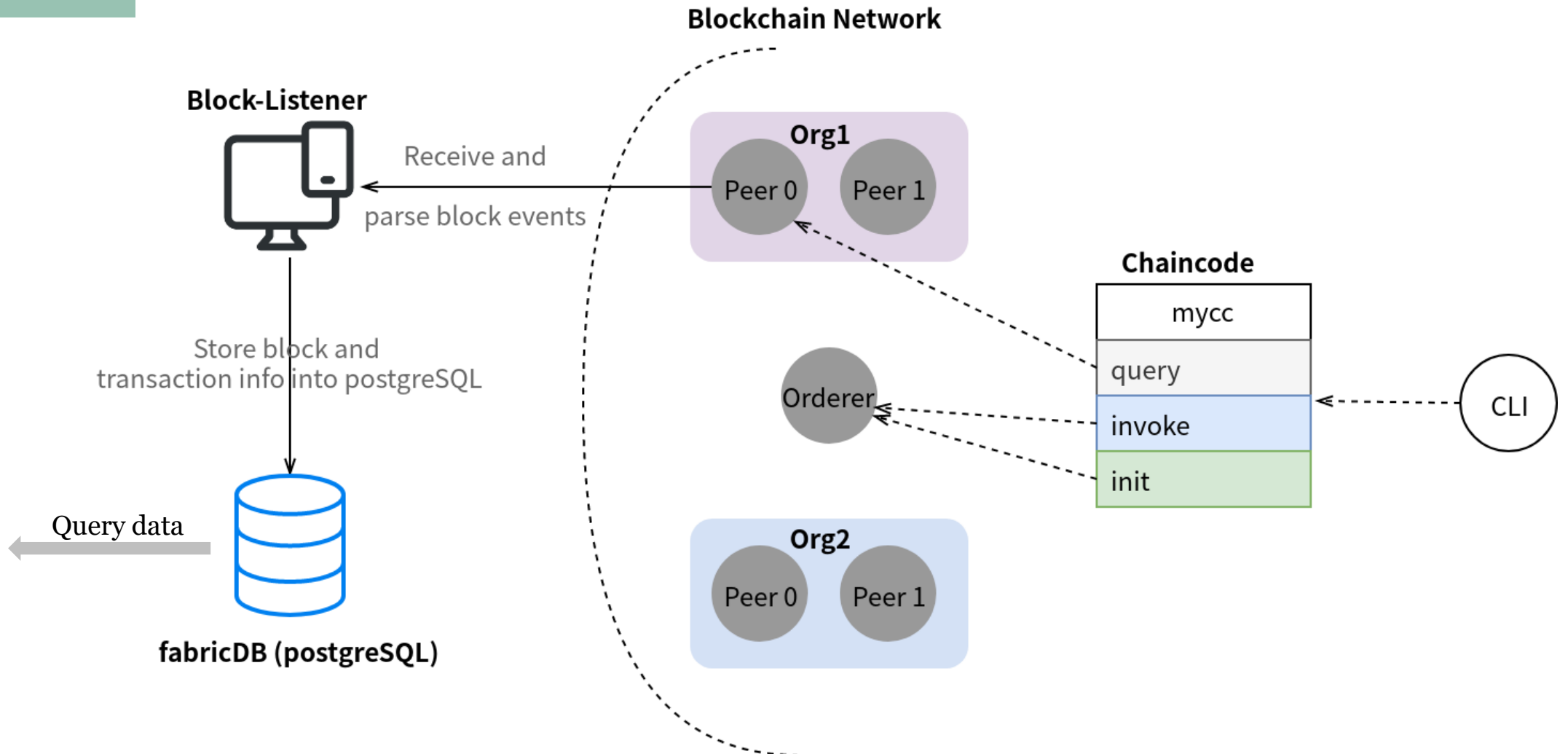
### METHOD





## (2) Blockchain Traceability System with HyperQL (Cont.)

### METHOD



## (2) Blockchain Traceability System with HyperQL (Cont.)

### ISSUES

#### ❑ Advantage :

Query data from the database is **nearly 50 times** faster than query from blockchain.

#### ❑ Disadvantage :

The data in the database is not immutable.

#### ❑ Future work :

Design an algorithm to efficiently synchronize the data in blockchain and database.  
With this, we can keep the data secure while having faster query speed.

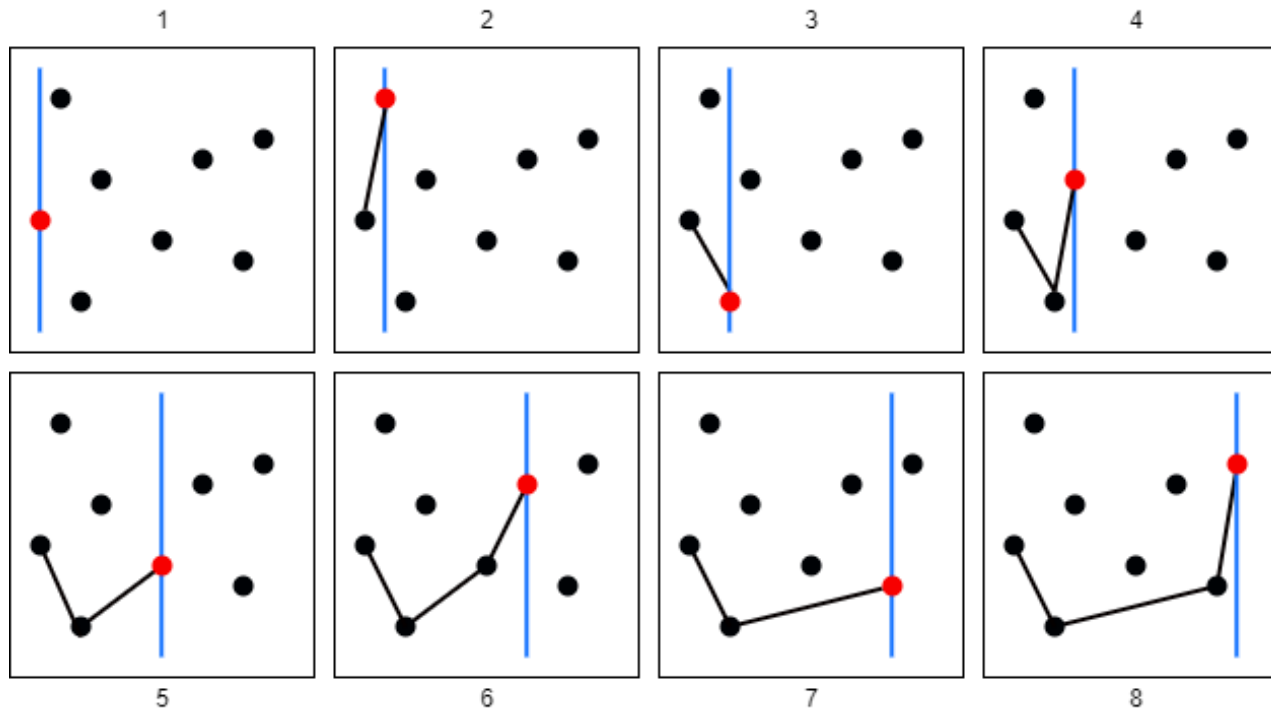
### **(3) Convex Hull Algorithm**

### (3) Convex Hull Algorithm

#### PURPOSE

- Can be applied to some fields such as image processing.

#### METHOD



My way of thinking :

1. Find the bottom-left point first.
2. Iterate the others from left to right to find the point that can form the smallest slope.
3. If the same slope occurs, take the longer one.
4. Do step 1 2 3 until reaching the far right
5. Then do the same thing from right to left.

### (3) Convex Hull Algorithm (Cont.)

#### ANALYZE

##### ❑ Time Complexity :

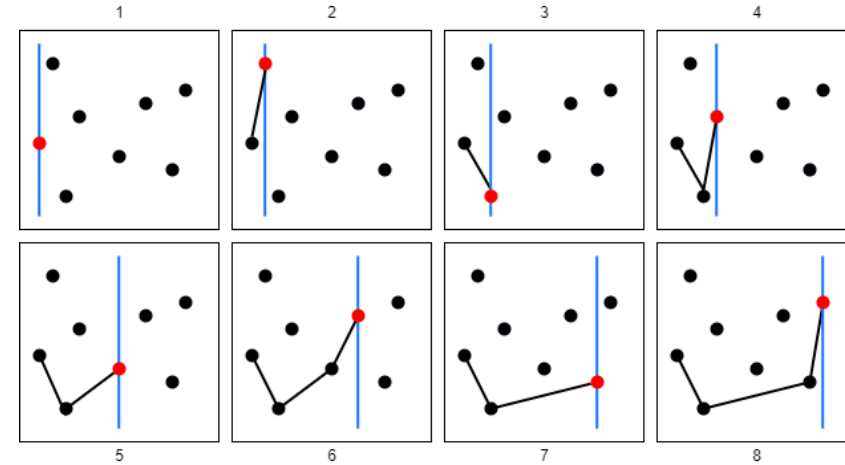
Since we need to use double loop to iterate all the points, the time complexity is  $O(n^2)$ .

##### ❑ Space Complexity :

We don't need any additional space to store all the points, so the space complexity is  $O(1)$

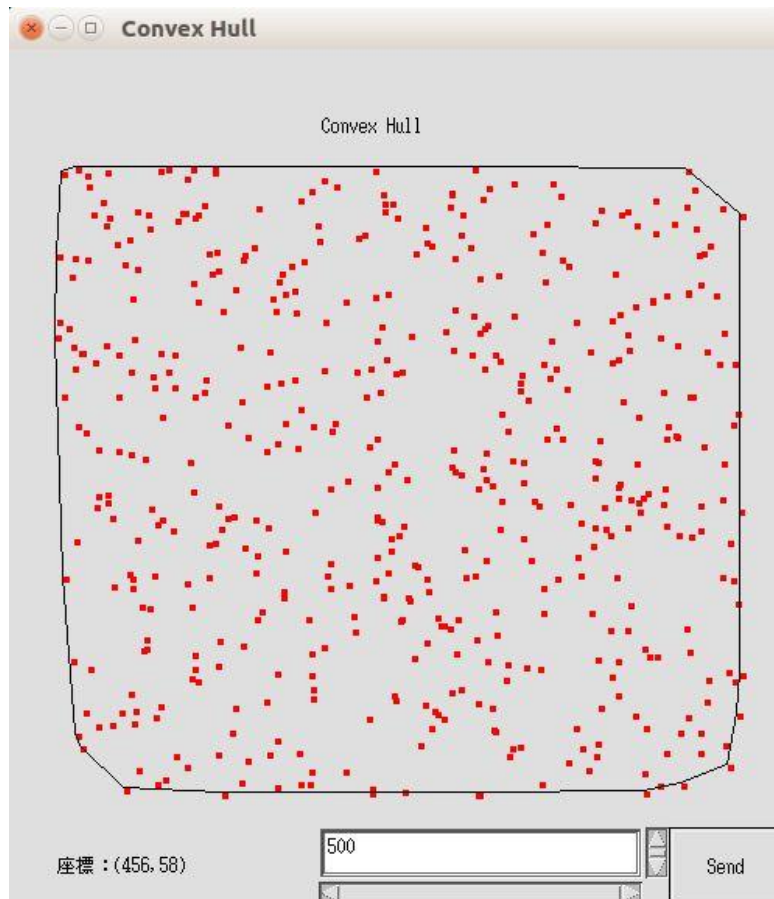
##### ❑ Discuss :

Although the time complexity is  $O(n^2)$ , but we don't have to sort all the points first.



### (3) Convex Hull Algorithm (Cont.)

#### RESULTS



```
yuan@yuan-VirtualBox: ~/Algorithm
at java.awt.EventQueue$3.run(EventQueue.java:709)
at java.awt.EventQueue$3.run(EventQueue.java:703)
at java.security.AccessController.doPrivileged(Nat
at java.security.ProtectionDomain$JavaSecurityAcce
rivilege(ProtectionDomain.java:80)
at java.security.ProtectionDomain$JavaSecurityAcce
rivilege(ProtectionDomain.java:90)
at java.awt.EventQueue$4.run(EventQueue.java:731)
at java.awt.EventQueue$4.run(EventQueue.java:729)
at java.security.AccessController.doPrivileged(Nat
at java.security.ProtectionDomain$JavaSecurityAcce
rivilege(ProtectionDomain.java:80)
at java.awt.EventQueue.dispatchEvent(EventQueue.java:
at java.awt.EventDispatchThread.pumpOneEventForFil
ad.java:201)
at java.awt.EventDispatchThread.pumpEventsForFilt
java:116)
at java.awt.EventDispatchThread.pumpEventsForHiera
ad.java:105)
at java.awt.EventDispatchThread.pumpEvents(EventDi
at java.awt.EventDispatchThread.pumpEvents(EventDi
at java.awt.EventDispatchThread.run(EventDispatchT
處理了500個點,耗費了 0.005000 秒
```

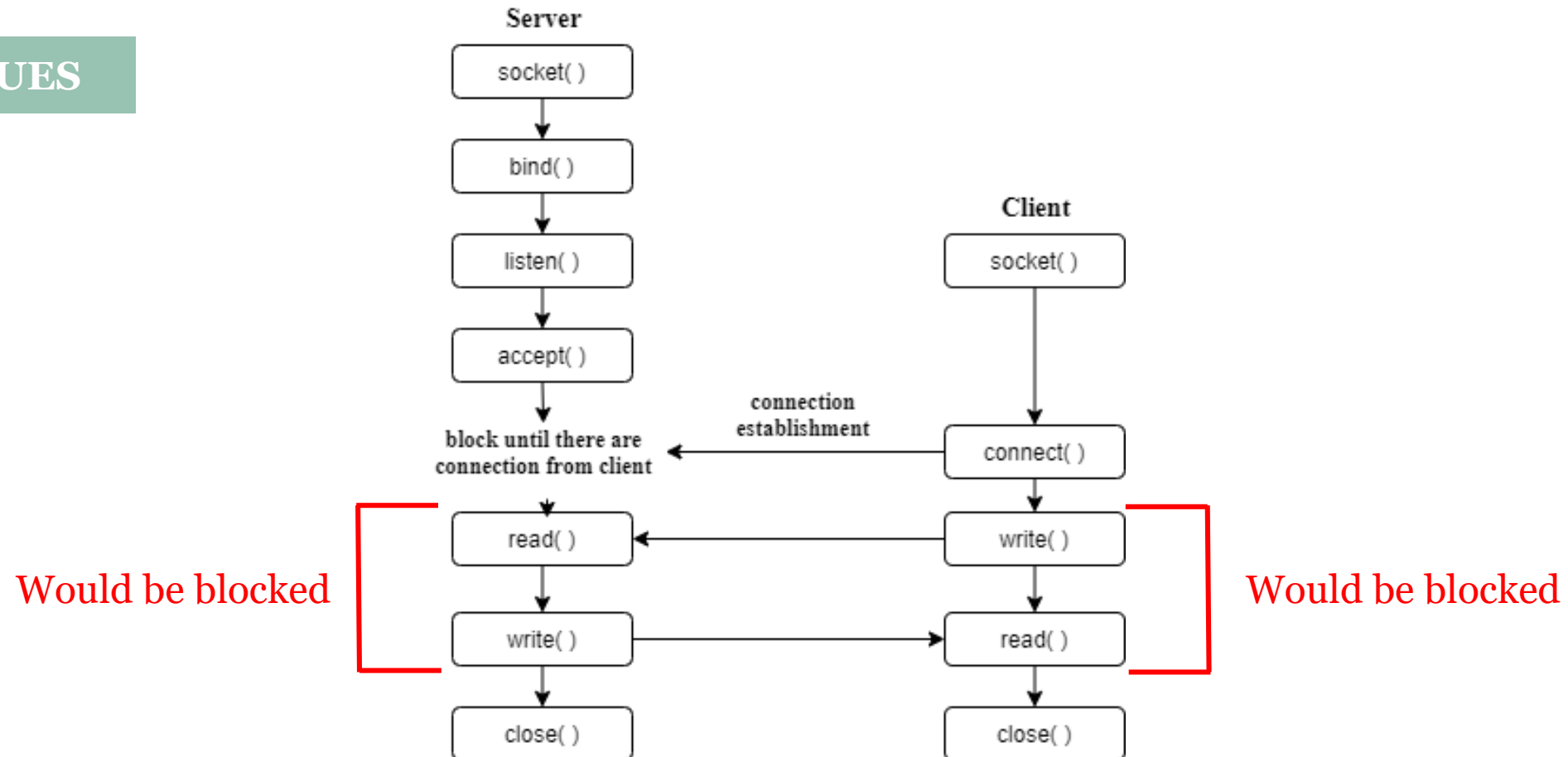
## **(4) Simple TCP Communication (IPv4)**

## (4) Simple TCP Communication (IPv4)

### PURPOSE

- ❑ Implement TCP message communication using `select( )` in C language.

### ISSUES





## (4) Simple TCP Communication (IPv4) (Cont.)

### METHOD

#### ❑ Server uses `select( )` function (Non-blocking):

`select( )` function privileges you to monitor multiple file descriptors at the same time.

#### ❑ Client uses multi-thread (Non-blocking)

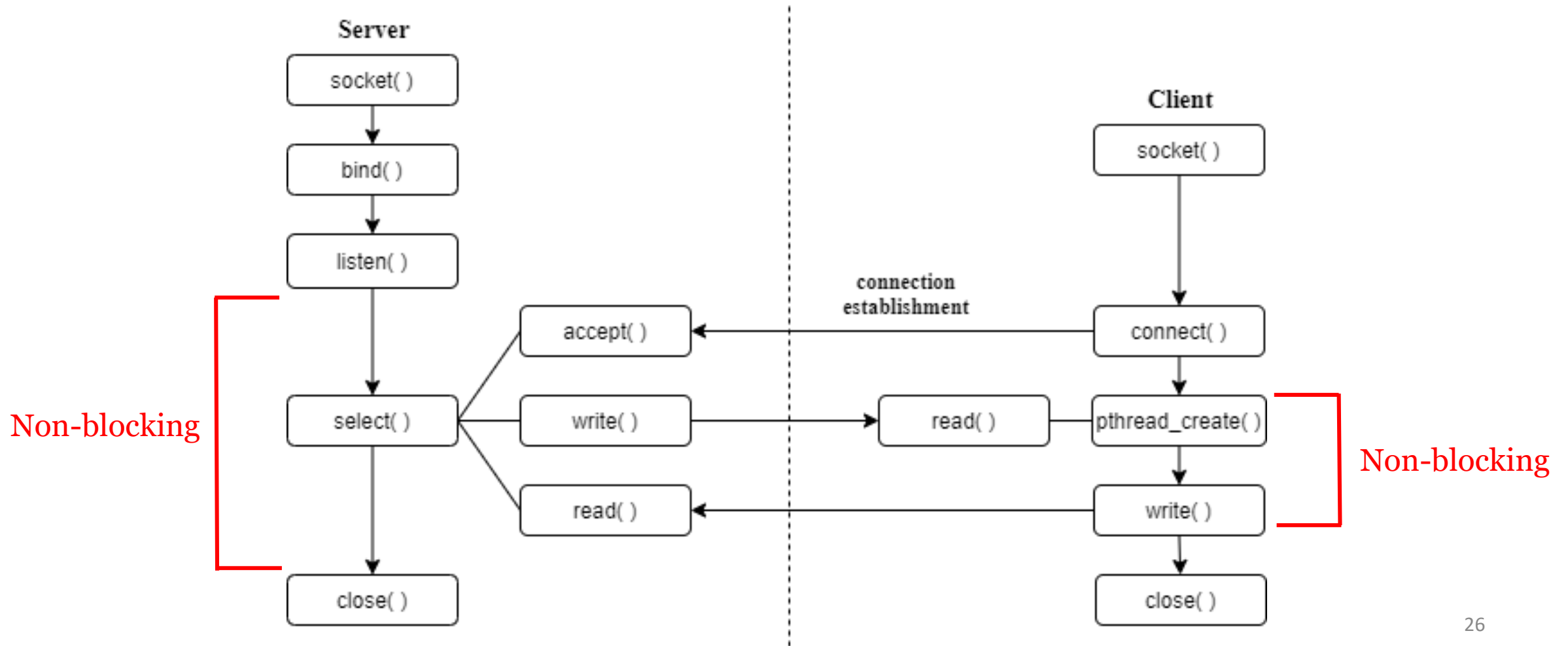
The thread is in charge of reading data which created by `pthread_create( )`. The main process is in charge of writing data.

#### ❑ Handle `SIGPIPE` signal

If a Client is disconnected, but the Server keeps sending data to the client, it will cause `SIGPIPE` signal produced.

## (4) Simple TCP Communication (IPv4) (Cont.)

### ARCHITECTURE



## (4) Simple TCP Communication (IPv4) (Cont.)

### RESULTS

#### □ Server :

```
chris@chris-X553MA: ~/G...etworkProgramming/hw4
File Edit Tabs Help
chris@chris-X553MA:~/Github/NetworkProgramming/hw4$ ./server
New connection from 127.0.0.1 on socket 4
New connection from 127.0.0.1 on socket 5
Hello, I'm Server
Hello, I'm Server
I'm client1, nice to meet you
nice to meet you, too. I'm client2
Time Out...
```

#### □ Client 1 :

```
chris@chris-X553MA: ~/...orkProgramming/hw4
File Edit Tabs Help
chris@chris-X553MA:~/Github/NetworkProgramming/hw4$ ./client
Hello, I'm Server
I'm client1, nice to meet you
nice to meet you, too. I'm client2
█
```

#### □ Client 2 :

```
chris@chris-X553MA: ~/...orkProgramming/hw4
File Edit Tabs Help
chris@chris-X553MA:~/Github/NetworkProgramming/hw4$ ./client
Hello, I'm Server
I'm client1, nice to meet you
nice to meet you, too. I'm client2
█
```

## **(5) Simple LED Driver on RPI**

## (5) Simple LED Driver on RPI

### PURPOSE

- ❑ Create a simple device driver on RPI.

### DRIVER

- ❑ In first step, we have to initial the kernel module.

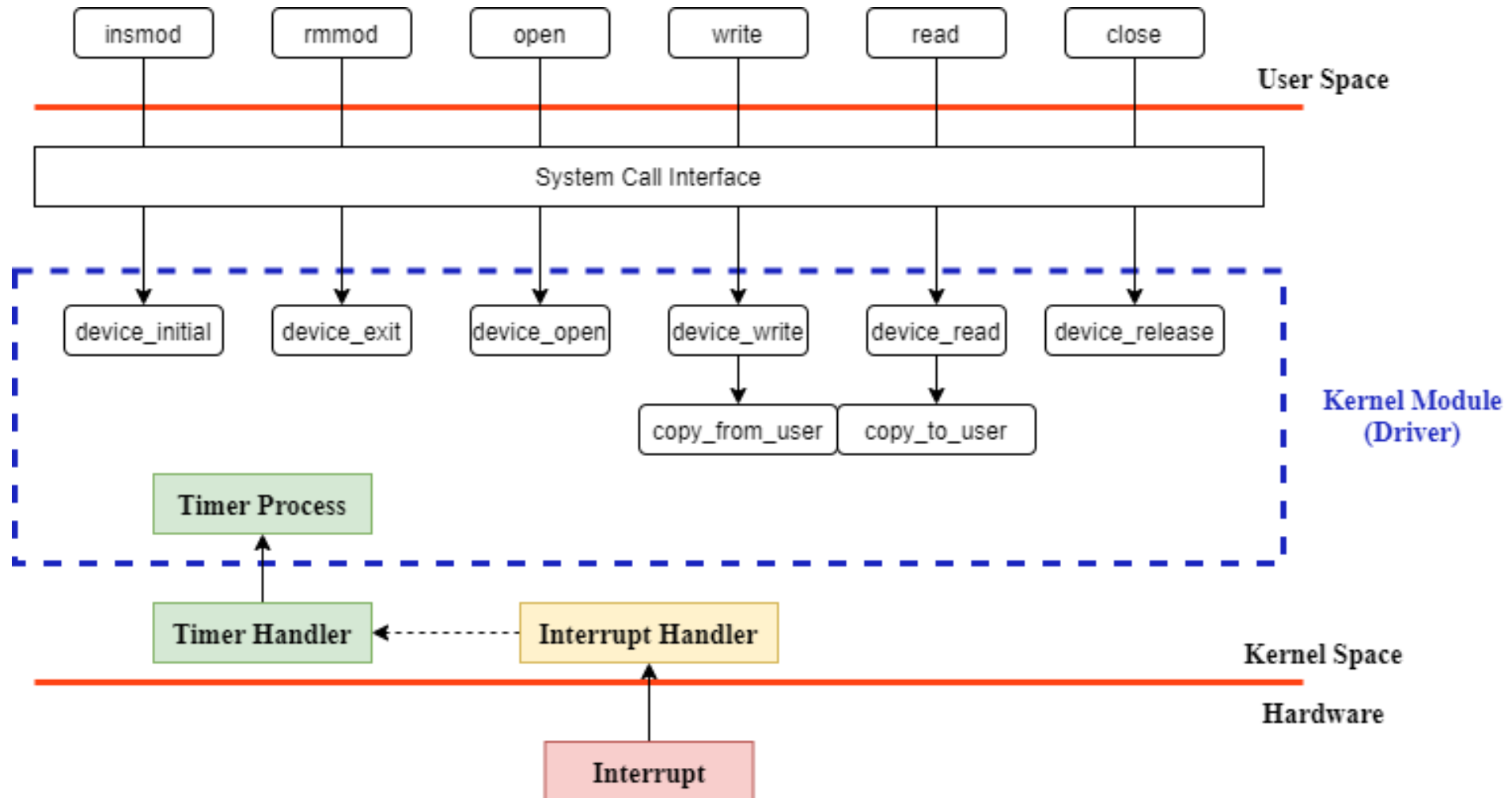
```
static int hello_init(void) {  
    misc_register(&misc);  
    printk(DEVICE_NAME" initialized\n");  
  
    return SUCCESS;  
}
```

```
static struct miscdevice misc = {  
    .minor = MISC_DYNAMIC_MINOR,  
    .name = DEVICE_NAME,  
    .fops = &fops,  
};
```

```
static struct file_operations fops = {  
    .owner = THIS_MODULE,  
    .read = device_read,  
    .write = device_write,  
    .open = device_open,  
    .release = device_release,  
};
```

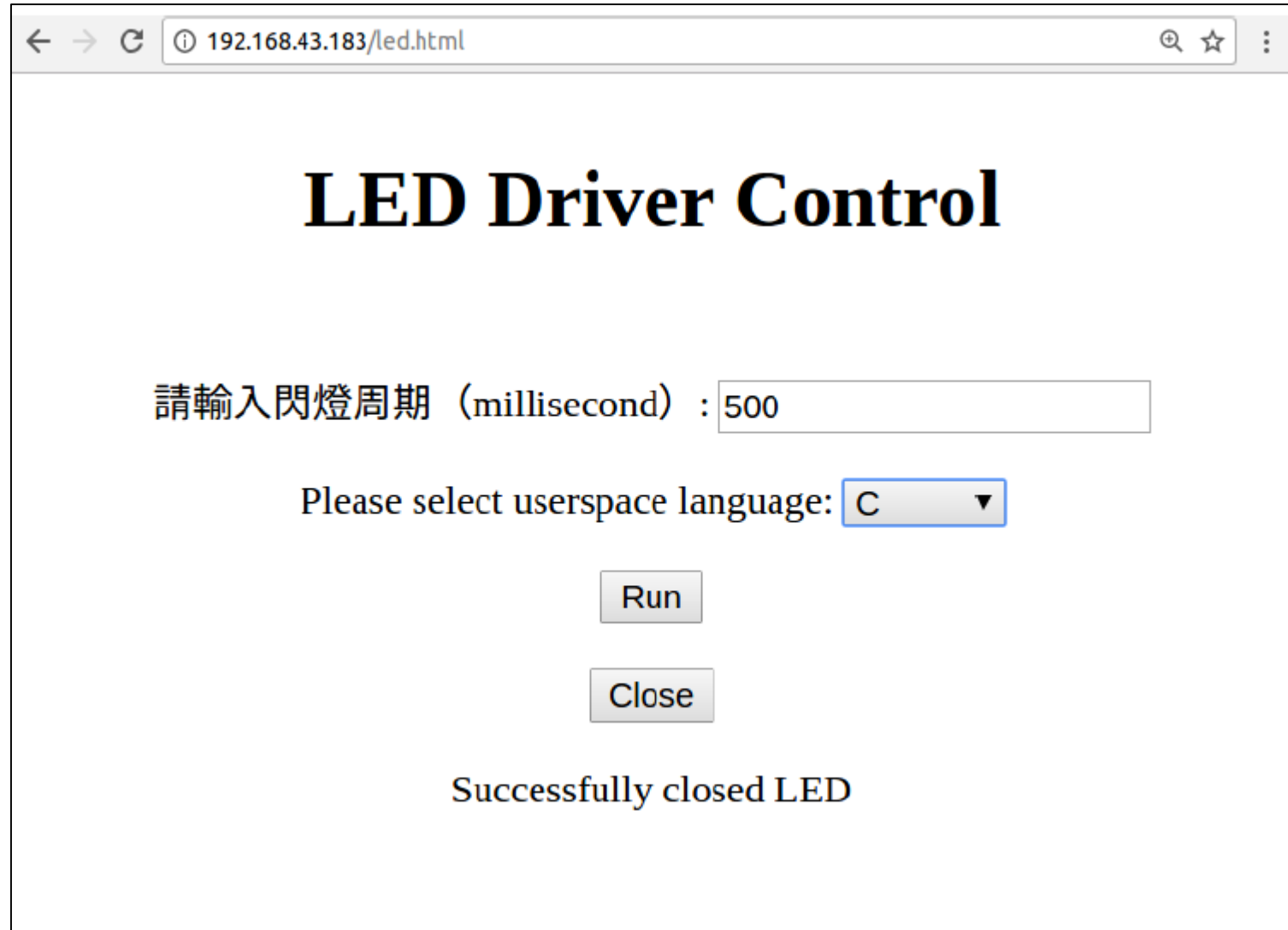
## (5) Simple LED Driver on RPI (Cont.)

### ARCHITECTURE



## (5) Simple LED Driver on RPI (Cont.)

### RESULTS



A screenshot of a web browser window displaying the 'LED Driver Control' interface. The browser's address bar shows the URL '192.168.43.183/led.html'. The page title is 'LED Driver Control'. The main content area contains a form with the following elements:

- A label '請輸入閃燈周期 (millisecond) :' followed by a text input field containing the value '500'.
- A label 'Please select userspace language:' followed by a dropdown menu with 'C' selected.
- A 'Run' button.
- A 'Close' button.
- A status message 'Successfully closed LED' at the bottom.

**Thank you for your time!**