

**ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH  
TRƯỜNG ĐẠI HỌC BÁCH KHOA  
KHOA KHOA HỌC VÀ KỸ THUẬT MÁY TÍNH**



**PHÁT TRIỂN ỨNG DỤNG INTERNET OF THINGS (CO3038)**

---

**LAB 3 : RECEIVE COMMANDS FROM DASHBOARD  
AND WRITING DATA TO SENSORS' VIA UART**

---

**SV thực hiện:**

Nguyễn Trọng Tín

2012215



## Contents

<b>1</b>	<b>INTRODUCTION</b>	<b>2</b>
<b>2</b>	<b>SYSTEM ARCHITECTURE</b>	<b>2</b>
<b>3</b>	<b>IMPLEMENTATION</b>	<b>2</b>
<b>4</b>	<b>REFERENCES</b>	<b>6</b>

## List of Figures

1	Enter Caption . . . . .	2
2	Configuration commands in GATEWAY . . . . .	4
3	Enter Caption . . . . .	5
4	Sending and Handshaking protocol . . . . .	5

## 1 INTRODUCTION

This LAB will proceed the progress of LAB2. From this we will focus on receive command from platform, preliminary process it, then forward this command to sensors.

## 2 SYSTEM ARCHITECTURE

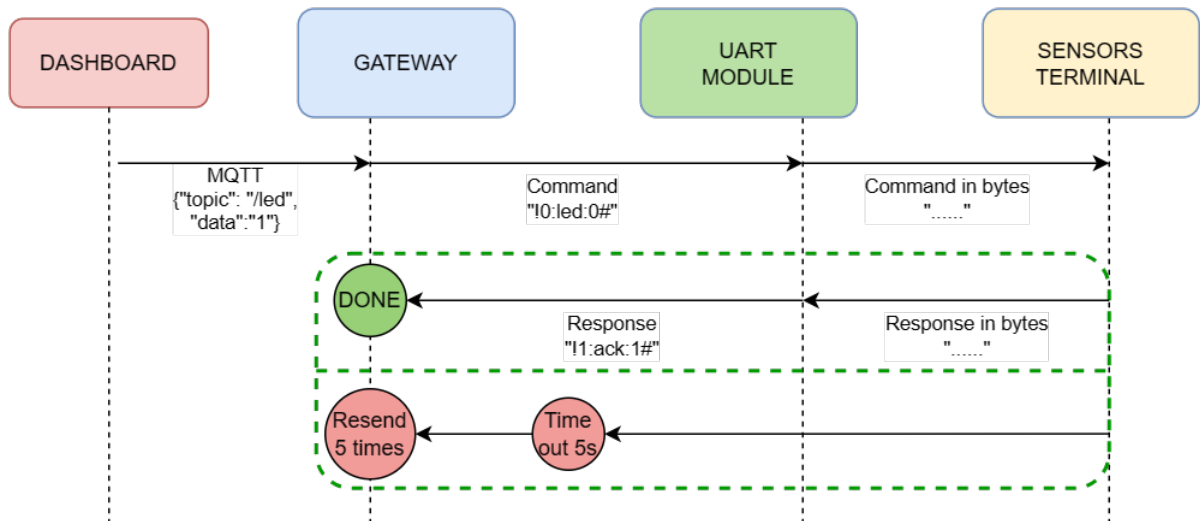


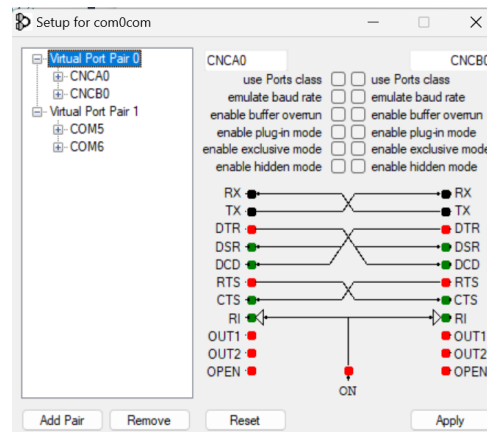
Figure 1: Enter Caption

This architecture describe a handshaking protocol between gateway and sensors. Gateway receives data commands from platform before processing this raw data. After that, Gateway will send specific command to sensors before it is waiting for ACK response for 5 seconds. If gateway does not receive any response in this time, it will resend data again (maximum trial is 5 times).

## 3 IMPLEMENTATION

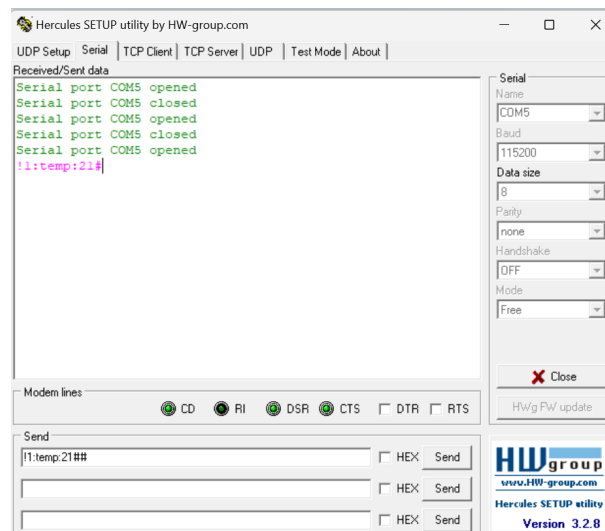
### 1. Create 2 virtual port using com0com

In the first step, we would open **com0com** to create 2 virtual ports. There are **COM5** and **COM6** in my case.



## 2. Assign 1 port to Hercules

Secondly, open Hercules Terminal, choose Serial Tab and Connect to one of two virtual ports. I connected to COM5.



## 3. Configuration of Gateway and Platform

There are 3 command-feeds in Platform including "water-heater", "led", "air-conditioner". Therefore we need define 3 commands to send to sensors "command\_number:feed\_name:value".

- "water-heater" commands: "1:heater:1"
- "led" commands: "1:led:50"
- "air-conditioner": "1:temp:38"

```
def EventCallBack ( client , feed_id , payload ):  
  
    if (feed_id == "led"):  
        if payload == "1":  
            if UART_driver.UART_WriteSerial("led", payload ):  
                print("LED ON !!!")  
        else:  
            if UART_driver.UART_WriteSerial( "led", "0" ):  
                print("LED OFF #")  
  
        elif (feed_id == "water-heater"):  
            if payload == "1":  
                if UART_driver.UART_WriteSerial( "heater", payload ):  
                    print("WATER HEATER ON !!!")  
            else:  
                if UART_driver.UART_WriteSerial( "heater", "0" ):  
                    print("WATER HEATER OFF ###")  
  
        elif (feed_id == "air-conditioner"):  
            global mySensor  
            if UART_driver.UART_WriteSerial("temp", payload ):  
                print (f"Setting Air Conditioner at {payload} °C")  
                mySensor.temp = float(payload) + 0.5  
  
        else:  
            print ("UNKNOWN COMMAND !!!")
```

Figure 2: Configuration commands in GATEWAY

#### 4. Implement sending data to Sensors and Handshake protocol

GATEWAY sends commands via UART to sensors, then it will be waiting for responses "1:ack:1" from sensors to confirm the sending accomplishment.

If there is no response in 3 seconds, the Gateway will resend data with 5 maximum re-trials.

After timeout and out of sending trials, Gateway recognizes the fail in sensors and notifies to Platform.

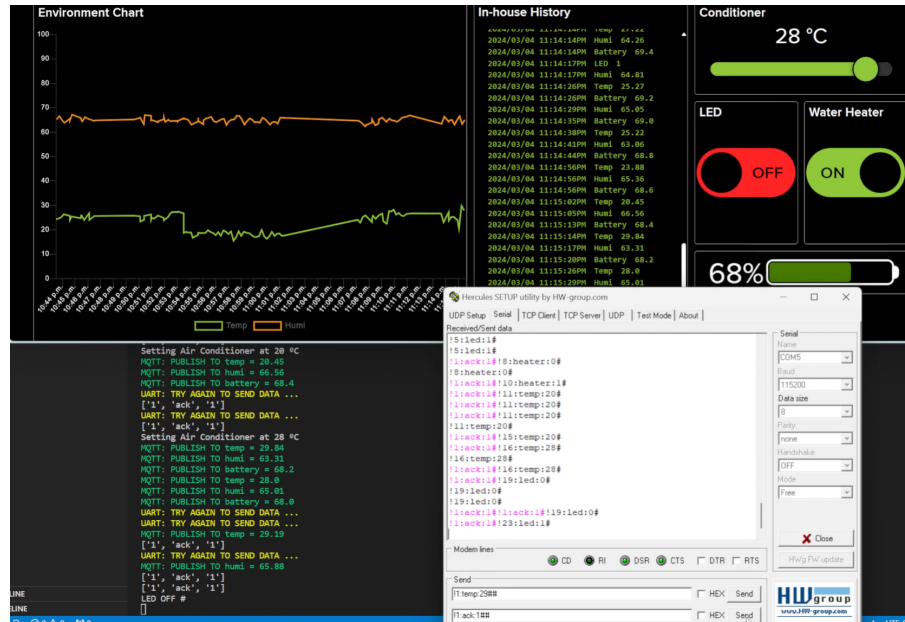


Figure 3: Enter Caption

```

write_count = 0
def UART_WriteSerial(topic:str, payload:str):
    global write_count
    global handshake
    data = "!" + str(write_count) + ":" + topic + ":" + payload + "#\n\r\n"

    # 5 trial sending
    trial = 5
    while trial:

        handshake = 0 #reset handshake before sending data
        ser.write( data.encode("utf-8") )

        write_count += 1

        # wait successful response from Sensor in 3 seconds
        waiting_ack = 3
        while waiting_ack:
            if handshake:
                break
            else:
                waiting_ack -= 1
                time.sleep(1)

        if waiting_ack > 0: #Successfully sending
            break
        else: #Timeout sending -> next trial, send again
            print_warning("UART: TRY AGAIN TO SEND DATA ... ")
            trial -=1

    if trial <= 0:
        print_error("UART: FAILED TO SEND DATA !!!")
        return 0

    else:
        return 1

```

Figure 4: Sending and Handshaking protocol



## 4 REFERENCES

Video Demo upload here: <https://youtu.be/0RejBR28rXw>

Please check my github for more information about source code and building problems

[https://github.com/JackWrion/IoT\\_LAB\\_2024/tree/main/Lab\\_03](https://github.com/JackWrion/IoT_LAB_2024/tree/main/Lab_03)