

# READ DATA FROM HS300X SENSOR

Nguyễn Đức Duy

102200374

Nguyễn Phước Đại Toàn 102200035

Nguyễn Quốc Tĩnh 102200393

# NỘI DUNG

01

**Giới thiệu về cảm biến H3001**

02

**Cấu hình và xây dựng chương trình**

03

**Demo kết quả đọc cảm biến**

04

**Bài tập**

# Giới thiệu về H3001

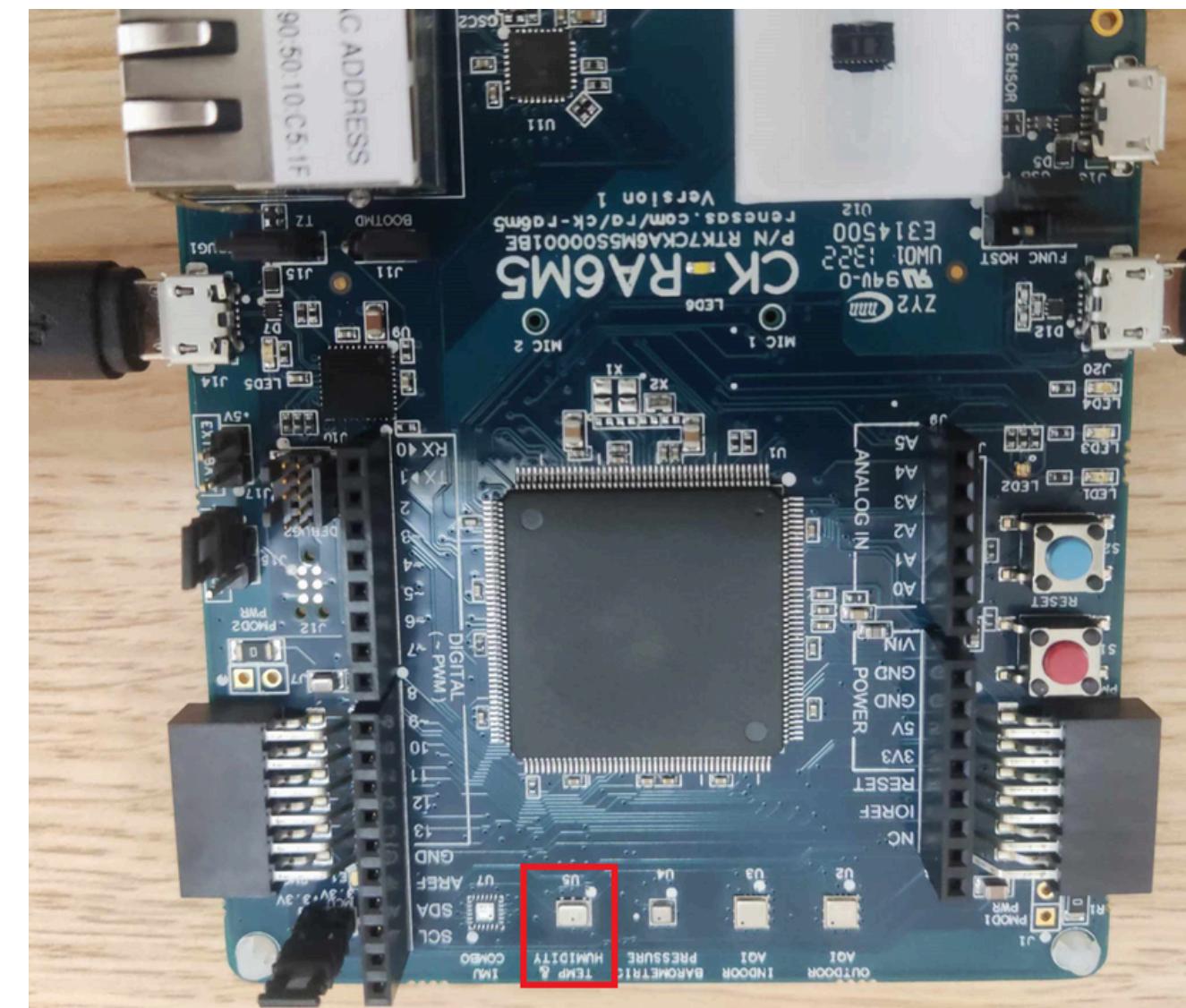
1.

IOT

# 1. Giới thiệu về HS3001

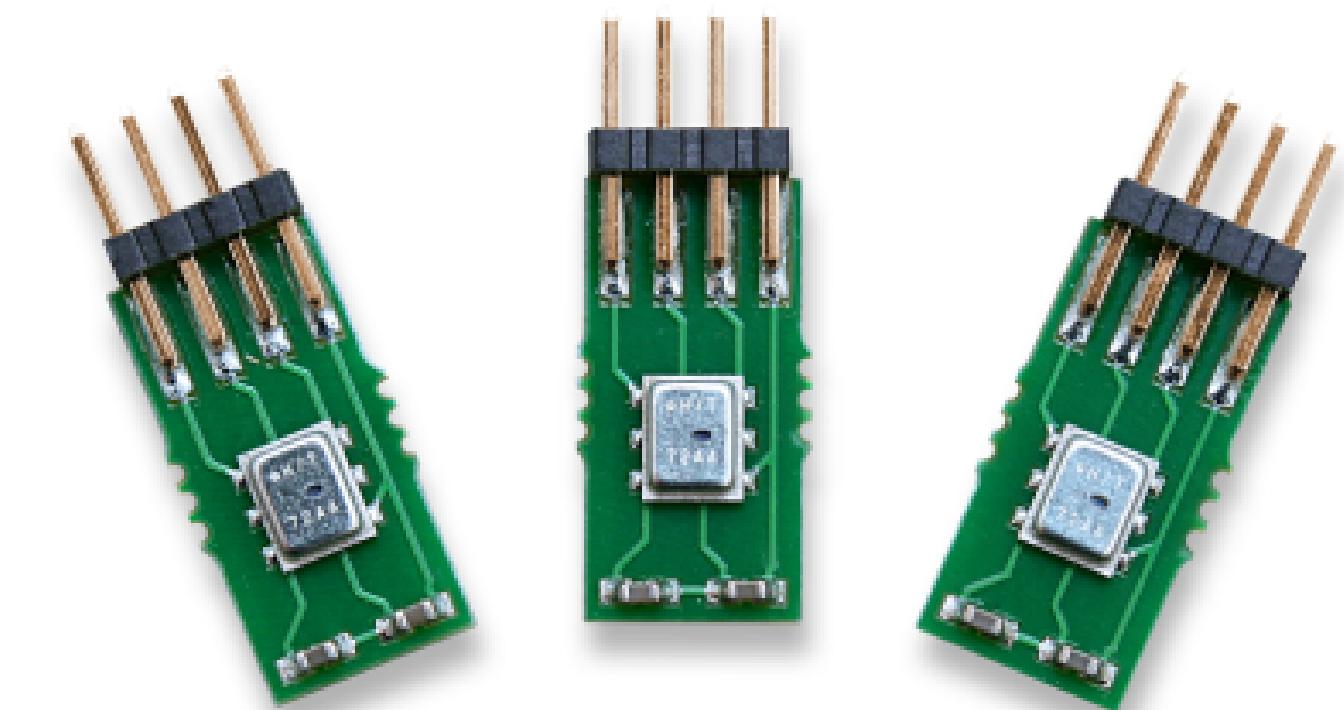
HS3001 là cảm biến độ ẩm và nhiệt độ, có đặc điểm:

- Độ chính xác cao
  - Thời gian phản hồi đo nhanh
  - Ổn định
  - Kích thước gói nhỏ
- => Lý tưởng cho nhiều ứng dụng từ môi trường di động đến môi trường khắc nghiệt.



# 1. Giới thiệu về HS3001

- RH accuracy:  $\pm 1.5\%$  RH
- T accuracy:  $\pm 0.2^\circ\text{C}$  ( $-10^\circ\text{C}$  to  $+80^\circ\text{C}$ )
- Fast RH response time (typical 6 seconds)



2.

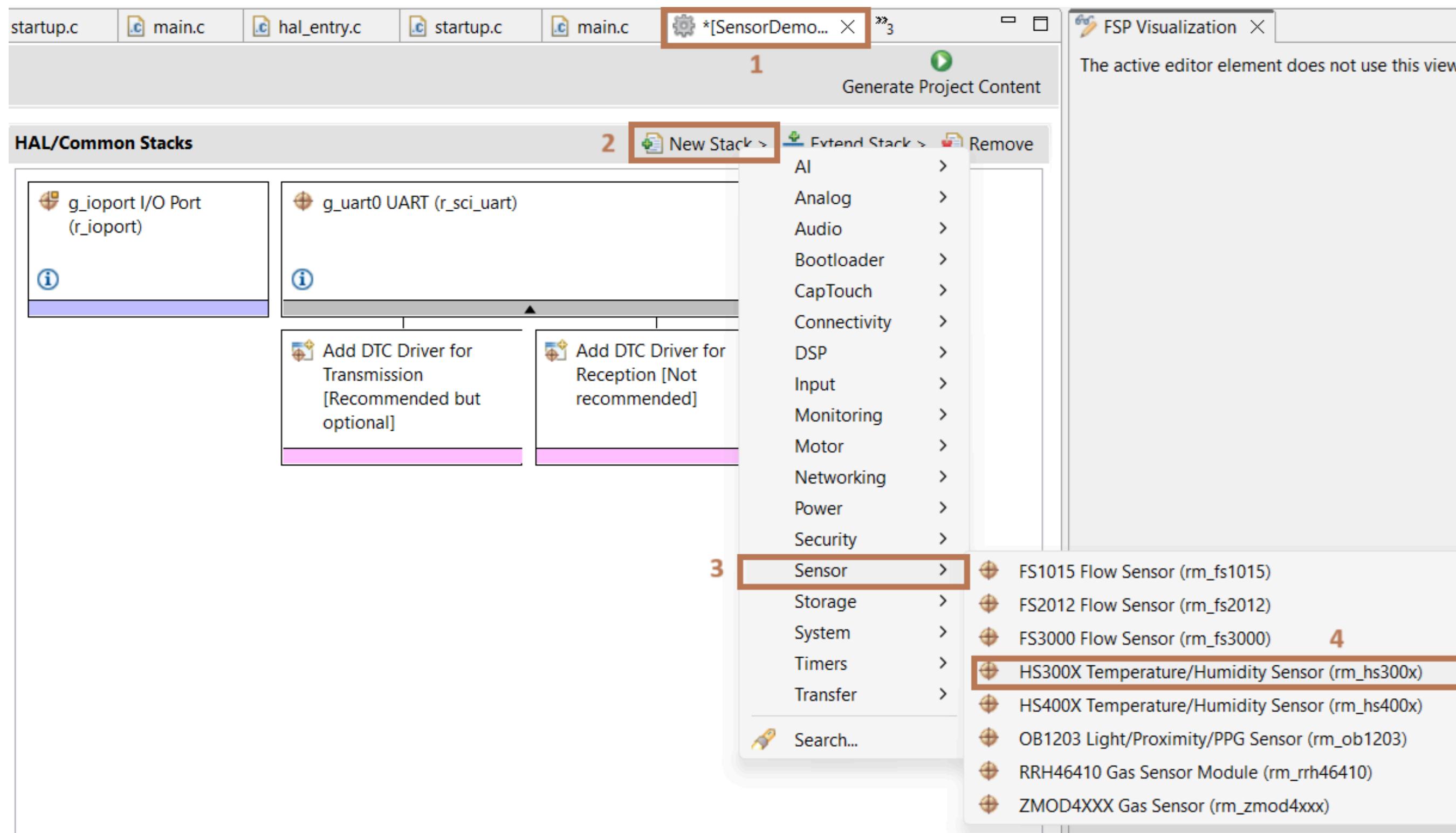
## Cấu hình và xây dựng chương trình

IOT

## 2.1. Cấu hình Stack trong Configuration file

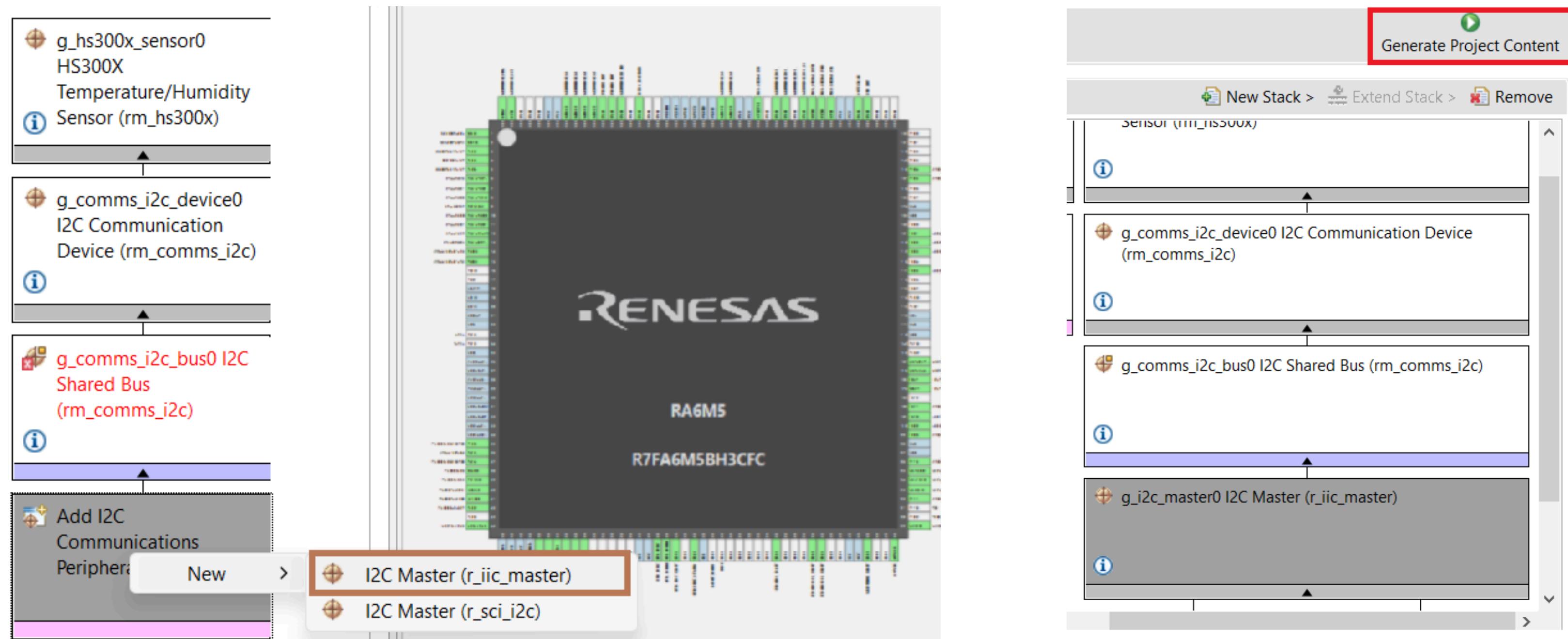
Cấu hình tương tự lab1 - lab2

Chọn configuration.xml => New Stack => Sensor => HS300X Sensor



## 2.1. Cấu hình Stack trong Configuration file

Add I2C Communications Peripheral => New => r\_iic\_master => Generate Project



## 2.2. Xây dựng chương trình

### Định nghĩa hàm callback

hal\_entry.c

C++

```
void hs300x_callback(rm_hs300x_callback_args_t * p_args);
static volatile bool hs300x_complete = false;
```

```
void hs300x_callback(rm_hs300x_callback_args_t * p_args){
    switch (p_args->event) {
        case RM_HS300X_EVENT_SUCCESS:
        {
            hs300x_complete = true;
            break;
        }
        default:
        {
        }
    }
}
```

## 2.2. Xây dựng chương trình

Khởi tạo và thiết lập các thành phần cần thiết để giao tiếp với cảm biến

hal\_entry.c

C++

```
fsp_err_t err = FSP_SUCCESS; // biến kiểm tra mã lỗi
rm_hs300x_raw_data_t hs300x_raw_data; // dữ liệu thô trước khi tính toán
rm_hs300x_data_t hs300x_data; // dữ liệu sau khi tính toán
uint8_t calculated_flag = 0;

/* Mở giao tiếp UART */
R_SCI_UART_Open(&g_uart0_ctrl, &g_uart0_cfg);

/* Thiết lập driver I2C cho cảm biến HS300X */
rm_comms_i2c_bus_extended_cfg_t * p_extend_hs300x =
(rm_comms_i2c_bus_extended_cfg_t *)g_hs300x_sensor0_cfg.p_instance->p_cfg->p_extend;
i2c_master_instance_t * p_driver_instance_hs300x = (i2c_master_instance_t *)
p_extend_hs300x->p_driver_instance;

/* Mở driver I2C */
p_driver_instance_hs300x->p_api->open(p_driver_instance_hs300x->p_ctrl,
p_driver_instance_hs300x->p_cfg);

/* Khởi tạo cảm biến */
RM_HS300X_Open(&g_hs300x_sensor0_ctrl, &g_hs300x_sensor0_cfg);
```

## 2.2. Xây dựng chương trình

Đọc dữ liệu từ cảm biến HS300X, tính toán giá trị độ ẩm và nhiệt độ

hal\_entry.c

```
while(1){  
    /* Di chuyển con trỏ về vị trí góc trên bên trái của terminal */  
    sprintf(write_buffer, "\x1b[H");  
    console_write(write_buffer);  
  
    hs300x_complete = false;  
    /* Bắt đầu đo lường giá trị cảm biến */  
    RM_HS300X_MeasurementStart(&g_hs300x_sensor0_ctrl);  
    while (false == hs300x_complete)  
    {  
        /* Chờ hàm callback */  
    }  
}
```

hal\_entry.c

```
do  
{  
    hs300x_complete = false;  
    /* Read ADC Data from HS300X */  
    RM_HS300X_Read(&g_hs300x_sensor0_ctrl, &hs300x_raw_data); // đọc giá trị  
    while (false == hs300x_complete)  
    {  
        /* Chờ hàm callback */  
    }  
    /* Tính toán giá trị nhiệt độ và độ ẩm từ giá trị thô */  
    err = RM_HS300X_DataCalculate(&g_hs300x_sensor0_ctrl, &hs300x_raw_data,  
&hs300x_data);  
    if (FSP_SUCCESS == err)  
    {  
        calculated_flag = 1;  
    }  
    else if (FSP_ERR_SENSOR_INVALID_DATA == err)  
    {  
        calculated_flag = 0;  
    }  
} while (0 == calculated_flag);
```

## 2.2. Xây dựng chương trình

Hiển thị kết quả lên màn hình terminal

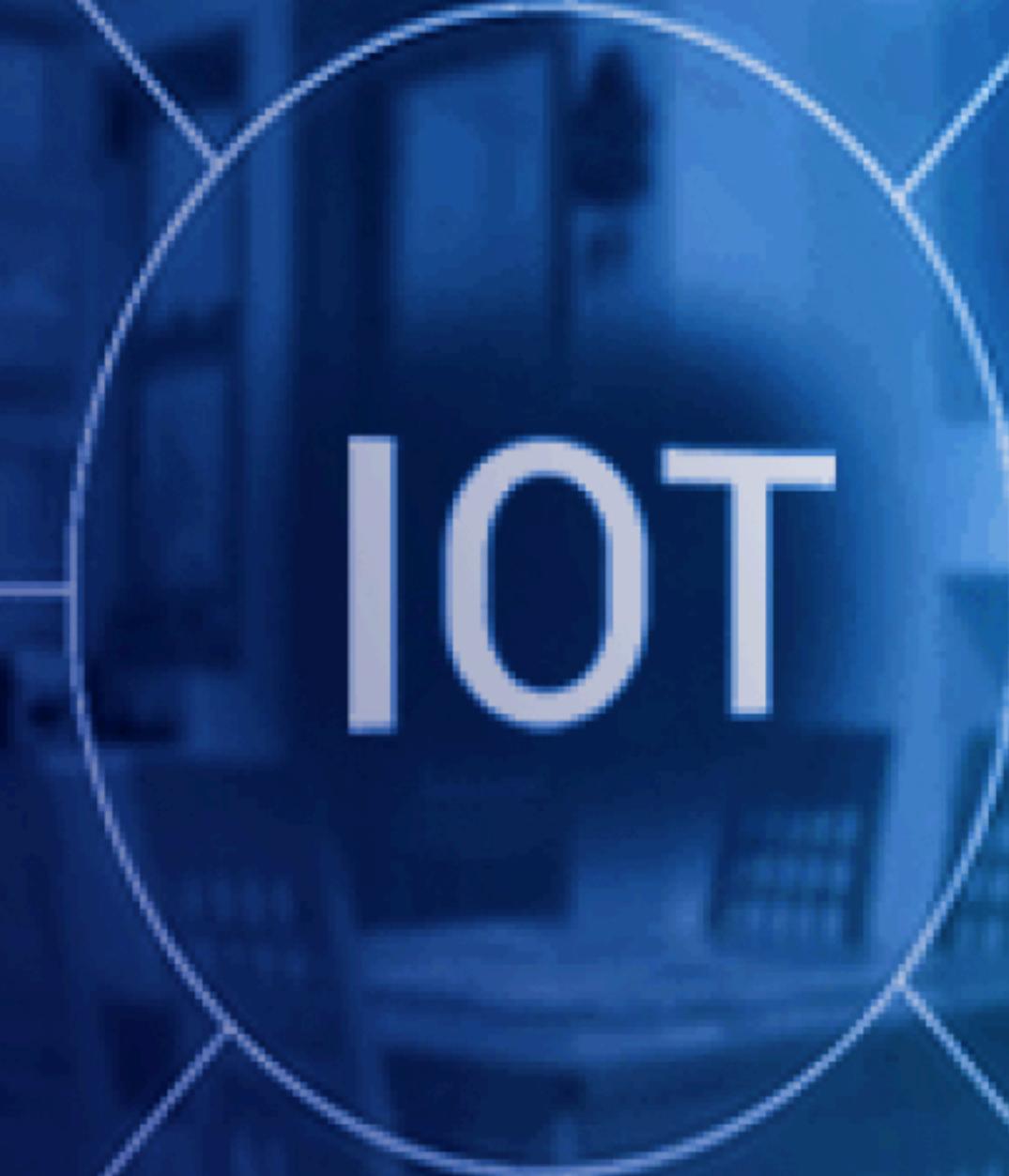
hal\_entry.c

Objective-C

```
    sprintf(write_buffer, "Humidity value: %d.%d \n\r", hs300x_data.humidity.integer_part,
hs300x_data.humidity.decimal_part);
    console_write(write_buffer);
    sprintf(write_buffer, "Temperature value: %d.%d
\n\r", hs300x_data.temperature.integer_part, hs300x_data.temperature.decimal_part);
    console_write(write_buffer);
    R_BSP_SoftwareDelay(delay, bsp_delay_units);
};
```

3 .

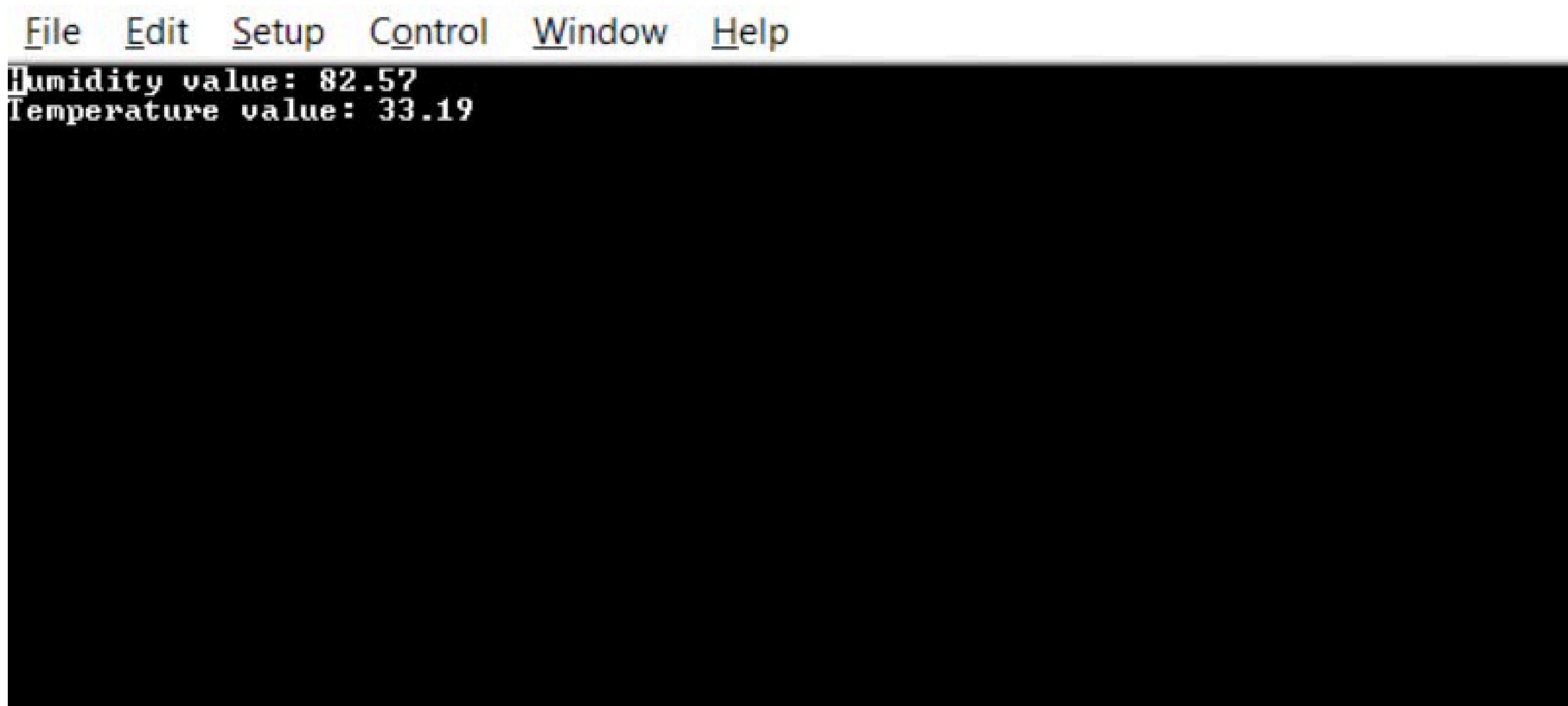
## Demo kết quả



IOT



### 3. Demo kết quả



4.

## Bài tập

IOT

## 4. Bài tập

Thay đổi led khi độ ẩm vượt quá ngưỡng  
65 (Led xanh -> đỏ)