

HƯỚNG DẪN SỬ DỤNG MODULE CẢM BIẾN BME280

I. Tổng quan về module cảm biến

Module BME280 là module cảm biến đo nhiệt độ, độ ẩm và áp suất có thể được ứng dụng rộng rãi: điều khiển hệ thống sưởi, điều hòa, thông gió; dự báo thời tiết...

Module BME280 có 4 chân bao gồm 2 chân nguồn (VCC, GND) và 2 chân giao tiếp theo chuẩn I2C.

Module sử dụng nguồn cấp 3,3V. Dải đo của cảm biến BME280:

- Nhiệt độ: -40...+85 °C
- Độ ẩm: 0...100 %
- Áp suất: 300...1100 hPa (1 hPa = 100 Pa)

Module cảm biến có 3 chế độ hoạt động: sleep mode, forced mode, normal mode.

- Sleep mode: không hoạt động. Tuy nhiên, có thể truy cập tất cả các thanh ghi. Tiêu thụ năng lượng thấp nhất. Sau khi cảm biến được cấp nguồn, tự động chuyển sang chế độ này.
- Forced mode: Chỉ thực hiện một phép đo, lưu trữ dữ liệu và chuyển sang sleep mode. Muốn tiếp tục đo ở chế độ này phải “đánh thức” cảm biến.
- Normal mode: đo liên tục.

II. Các hàm trong thư viện

bme280_open(): Gọi hàm này để mở driver. Nếu có lỗi xảy ra, sẽ có thông báo “Failed to open the device”.

bme280_config(int fd, int osrsT, int osrsP, int osrsH, int Mode, int Tsb, int IIR): hàm này dùng để cài đặt chế độ hoạt động cho cảm biến.

- fd: tên của đối tượng.
- osrsT, osrsP, osrsH: chế độ lấy mẫu cho nhiệt độ, áp suất, độ ẩm (bao gồm: OFF, 1, 2, 4, 8, 16). Tổng quát code cần nhập: OSRS_x (x: các chế độ được liệt kê). Ví dụ: OSRS_OFF, OSRS_1,...
- Mode: chế độ hoạt động của cảm biến (bao gồm: SLEEP, FORCED, NORMAL). Tổng quát code cần nhập: MODE_x (x: các chế độ được liệt kê). Ví dụ: MODE_NORMAL
- Tsb: thời gian nghỉ trong normal mode (standby time, tính theo ms) (bao gồm: 0.5; 62.5; 125; 250; 500; 1000; 10; 20). Tổng quát code cần nhập: T_SB_x (x: các chế độ được liệt kê). Ví dụ: T_SB_0p5, T_SB_62p5, T_SB_125,...
- IIR: hệ số lọc (bao gồm: OFF, 2, 4, 8, 16). Tổng quát code cần nhập: IIR_x (x: các chế độ được liệt kê). Ví dụ: IIR_OFF, IIR_2,...

bme280_close(int fd): Gọi hàm này để đóng driver. Nếu có lỗi xảy ra, sẽ có thông báo “Failed to close the device”.

bme280_wakeup(int fd): Hàm này dùng để “đánh thức” cảm biến khi dùng Forced mode.
bme280_status(int fd): Hàm này dùng để đọc trạng thái data và trạng thái chuyển đổi data.
bme280_reset(int fd): Hàm này dùng để soft reset cảm biến.
bme280_read_temp(int fd): Hàm này dùng để đọc giá trị nhiệt độ.
bme280_read_hum(int fd): Hàm này dùng để đọc giá trị độ ẩm.
bme280_read_press(int fd): Hàm này dùng để đọc giá trị áp suất.

III. Hướng dẫn sử dụng thư viện và driver

❖ Include thư viện

```
#include "lib_bme280.h"
```

❖ Khởi tạo 1 đối tượng có tên tùy ý (ví dụ lấy tên “fd”), sau đó gọi hàm *bme280_open()*

```
int fd = bme280_open();
```

❖ Gọi hàm *bme280_config* và truyền vào các đối số để thiết lập chế độ hoạt động.

```
bme280_config(fd, OSRS_16, OSRS_2, OSRS_1, MODE_NORMAL, T_SB_1000, IIR_16);
```

(Ví dụ trên truyền vào đối số thứ nhất đối tượng là fd; chế độ lấy mẫu cho nhiệt độ, áp suất, độ ẩm lần lượt là 16, 2, 1; chế độ hoạt động là normal; standby time là 1000 ms; chế độ lọc là 16).

❖ Trong vòng lặp while, gọi hàm tương ứng để đọc nhiệt độ, độ ẩm, áp suất sau đó delay chờ đến lần đọc tiếp theo. Nếu dùng chế độ Forced mode thì sau khi config và đọc về giá trị cảm biến, gọi hàm *bme280_wakeup(int fd)*.

```
while (1) {
    float temp = bme280_read_temp(fd);
    float hum = bme280_read_hum(fd);
    float press = bme280_read_press(fd);
    printf("Temperature: %.2f °C\n", temp);
    printf("Humidity: %.2f %%\n", hum);
    printf("Pressure: %.2f Pa\n", press);
    sleep(1); // Delay 1 second
    // Neu dung Forecd mode thi goi ham wake up o day
    //bme280_wakeup(fd);
}
```