

MÔN: HỆ THỐNG NHÚNG
TÀI LIỆU HƯỚNG DẪN CHO DRIVER CẢM BIẾN TCS34725

GVHD: TS. Bùi Hà Đức

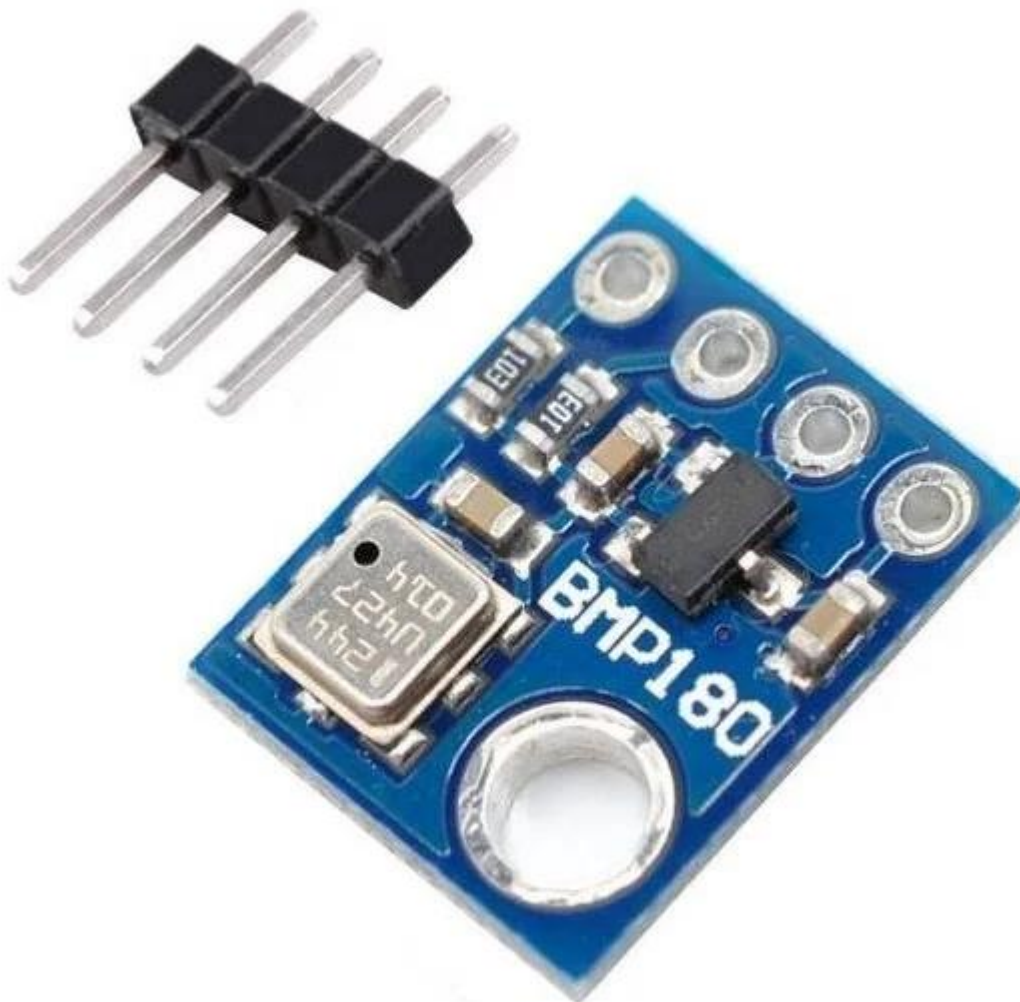
Tên: Nguyễn Trung Nhân - 211460333

Trần Trương Huy Hoàng - 21146389

Restricted data sheet

BMP180

Digital pressure sensor



BMP180

Digital pressure sensor

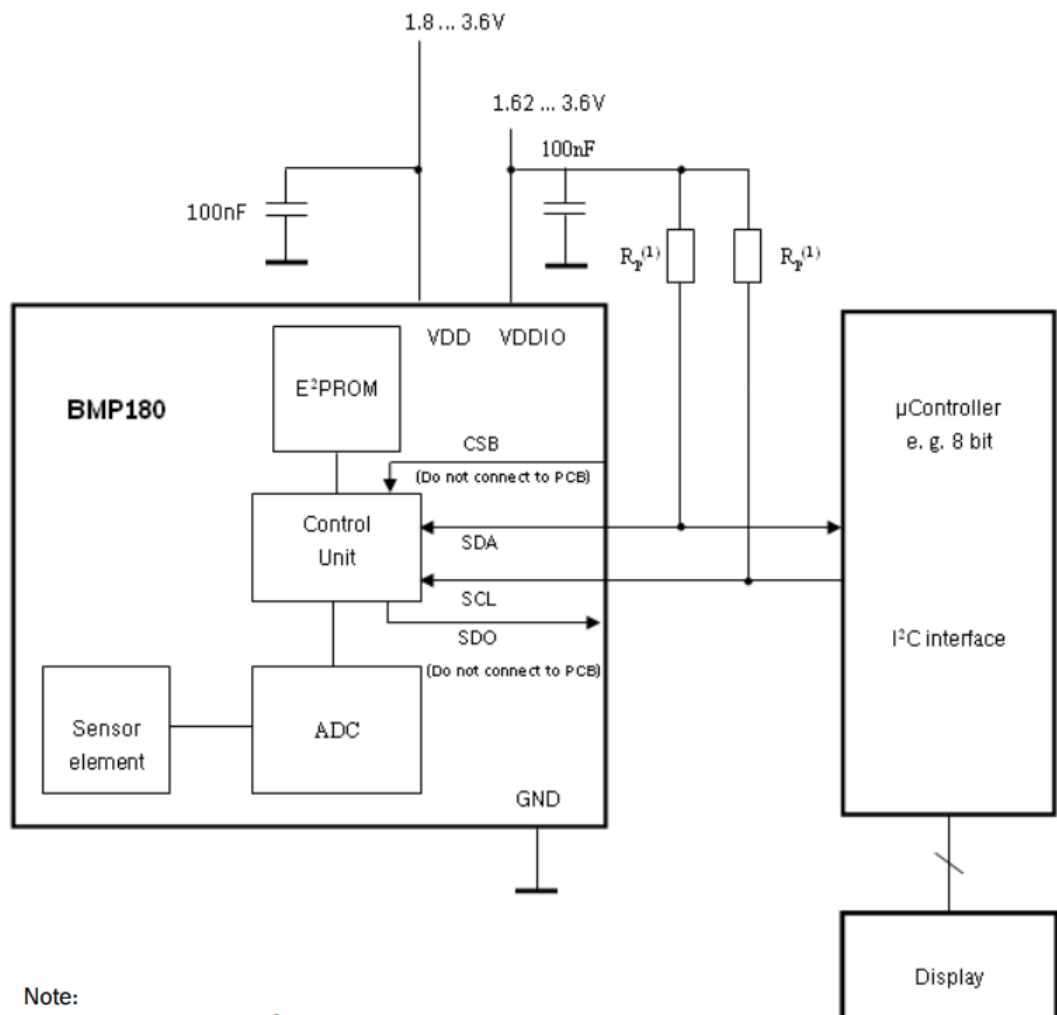
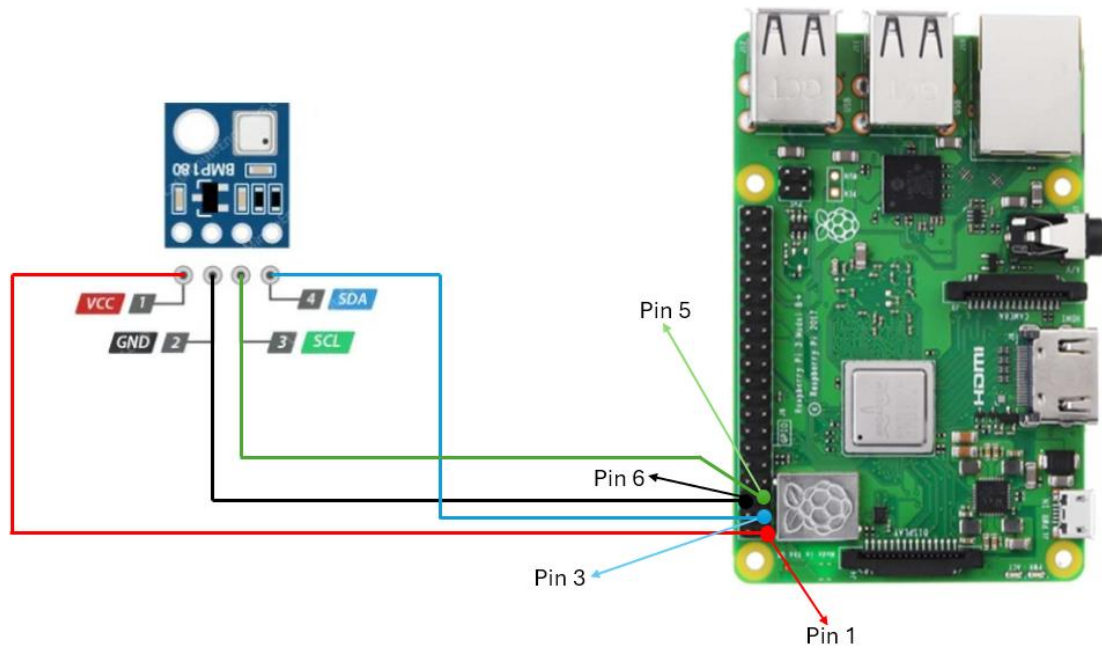
Đặc điểm nổi bật:

- Dải áp suất: 300 ... 1100hPa (+9000m ... -500m so với mực nước biển)
- Điện áp cung cấp: 1.8 ... 3.6V (VDD)
1.62V ... 3.6V (VDDIO)
- Vỏ đóng gói: LGA với nắp kim loại
- Kích thước nhỏ: 3.6mm x 3.8mm
- Siêu mỏng: 0.93mm chiều cao
- Tiêu thụ điện thấp: 5 μ A tại 1 mẫu / giây ở chế độ tiêu chuẩn
- Độ nhiễu thấp: 0.06hPa (0.5m) ở chế độ siêu tiết kiệm điện
0.02hPa (0.17m) ở chế độ độ phân giải cao
- Bao gồm đo nhiệt độ
- Giao tiếp I2C
- Hiệu chuẩn hoàn toàn
- Không chì, không halogen và tuân thủ RoHS
- MSL 1

Ứng dụng điển hình:

- Cải thiện định vị GPS (ước lượng quãng đường, phát hiện độ dốc, v.v.)
- Điều hướng trong nhà và ngoài trời
- Giải trí và thể thao
- Dự báo thời tiết
- Chỉ báo vận tốc theo phương thẳng đứng (tốc độ lên/xuống)

I. Sơ đồ kết nối:



Note:

(1) Pull-up resistors for I²C bus, $R_p = 2.2k\Omega \dots 10k\Omega$, typ. $4.7k\Omega$

II. Nguyên lí hoạt động:

- Cảm biến áp suất :
BMP180 sử dụng một phần tử cảm biến áp suất vi cơ điện (MEMS) để đo áp suất khí quyển.
Khi áp suất không khí thay đổi, màng cảm biến sẽ bị biến dạng → thay đổi điện trở nội tại.
- Đo tín hiệu analog:
Tín hiệu từ phần tử cảm biến là analog (dạng điện áp nhỏ).
Tín hiệu này được chuyển đổi thành tín hiệu số thông qua bộ ADC (Analog-to-Digital Converter) tích hợp sẵn.
- Đo nhiệt độ:
BMP180 có cảm biến nhiệt độ tích hợp để đo nhiệt độ môi trường.
Dữ liệu nhiệt độ này dùng để bù sai số cho phép đo áp suất (vì áp suất phụ thuộc nhiệt độ).
- Hiệu chuẩn nội bộ:
BMP180 chứa EEPROM với hệ số hiệu chuẩn riêng cho từng cảm biến (được nhà sản xuất lập trình).
Bộ vi điều khiển phải sử dụng các hệ số này để tính toán ra giá trị áp suất và nhiệt độ thực tế từ dữ liệu thô (raw).
- Giao tiếp I2C:
BMP180 giao tiếp với vi điều khiển qua chuẩn I2C, chỉ cần 2 dây: SDA và SCL.
Sau khi gửi lệnh đo, vi điều khiển sẽ đọc dữ liệu thô từ BMP180, sau đó áp dụng công thức bù để lấy giá trị chính xác.

III. Hướng dẫn sử dụng driver:

- Để giao tiếp giữa lớp user và kernel sử dụng hàm sau:

Ex: `ioctl(device, IOCTL_BMP180_GET_TEMP, &temp);`

`temp` có thể là giá trị dùng lưu data từ sensor hoặc dùng để cấu hình

Tương ứng với `IOCTL_BMP180_GET_TEMP` dưới đây là các IOCTL còn lại:

`IOCTL_BMP180_COEF`

`IOCTL_BMP180_GET_PRES`

`IOCTL_BMP180_GET_ALTI`

`IOCTL_BMP180_GET_OSS`

`IOCTL_BMP180_SET_OSS`

- Thanh ghi Measurement control:

Register Name	Register Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Reset
ctrl_meas	F4h	oss<1:0>		sco	measurement					00h

Ghi lệnh để bắt đầu phép đo nhiệt độ hoặc áp suất

Địa chỉ: 0xF4

Quyền truy cập: Read/Write

Giá trị reset: 0x00

bit 0-2: chọn phép đo

000: không đo

001: đo nhiệt độ

010: đo áp suất

bit 3-4: không sử dụng (reversed)

bit 5: start of conversion: khi đặt bit này thành 1, bắt đầu phép đo

bit 6-7: OverSampling Setting (Cấu hình lấy mẫu áp suất):

00 (0): Độ phân giải thấp, thời gian đo nhanh (4.5 ms)

01 (1): Trung bình (7.5 ms)

10 (2): Cao (13.5 ms)

11 (3): Rất cao (25.5 ms)

Ex:

`long pres = 0;`

`int oss = -1;`

`res = ioctl(device, IOCTL_BMP180_GET_PRES, &pres);`

`ioctl(device, IOCTL_BMP180_GET_OSS, &oss);`

- Thanh ghi out_msb, out_lsb, out_xlsb(Output MSB Register, Output LSB Register, Output XLSB Register):

Register Name	Register Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Reset
out_xlsb	F8h	adc_out_xlsb<7:3>					0	0	0	00h
out_lsb	F7h	adc_out_msb<7:0>								00h
out_msb	F6h	adc_out_lsb<7:0>								80h

Chứa kết quả đo lần lượt là:

out_msb: Dữ liệu đo – byte cao

out_lsb: Dữ liệu đo – byte giữa

out_xlsb: 5 bit thấp nhất của kết quả áp suất

Địa chỉ: 0xF6, 0xF7, 0xF8

Quyền truy cập: Read

Ex:

```
ioctl(device, IOCTL_BMP180_GET_TEMP, &temp);
```

Driver sẽ đọc 0xF6, 0xF7, 0xF8 để tạo ra giá trị temp

- Thanh ghi ID:

Register Name	Register Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Reset
id	D0h	id<7:0>								55h

Địa chỉ: 0xD0

Chứa mã định danh của thiết bị (mặc định 0x55)

Quyền truy cập: Read

Ex:

```
char buf[] = { REG_CHIP_ID }; // REG_CHIP_ID = 0xD0
```

```
i2c_master_send(client, buf, 1); // Gửi địa chỉ thanh ghi
```

```
buf[0] = 0;
```

```
i2c_master_recv(client, buf, 1); // Nhận 1 byte dữ liệu
```

```
if (buf[0] != CHIP_ID) // CHIP_ID = 0x55
```

```
return -ENODEV;
```

- Thanh ghi Calibxx (Calibration coefficients):

Table 5: Calibration coefficients

	BMP180 reg adr	
Parameter	MSB	LSB
AC1	0xAA	0xAB
AC2	0xAC	0xAD
AC3	0xAE	0xAF
AC4	0xB0	0xB1
AC5	0xB2	0xB3
AC6	0xB4	0xB5
B1	0xB6	0xB7
B2	0xB8	0xB9
MB	0xBA	0xBB
MC	0xBC	0xBD
MD	0xBE	0xBF

Địa chỉ 0xAA

Đọc các hệ số hiệu chuẩn AC1–AC6, B1, B2, MB, MC, MD (22 byte, từ 0xAA đến 0xBF).

Ex:

```
bmp180_read_bytes(REG_COEF,(char*)bmp180_coef, sizeof(struct
bmp180_coefficients));
```


HƯỚNG DẪN CÀI ĐẶT FILE DRIVER

Biên dịch

Để biên dịch tất cả các thành phần, sử dụng:

```
make all clean
```

Để chỉ biên dịch mô-đun nhân, tập tin overlay cây thiết bị hoặc ứng dụng kiểm tra, sử dụng một trong các lệnh sau:

```
make driver clean
```

```
make tree
```

```
make app
```

Để xóa tất cả các tập tin trừ các thành phần đã tạo và mã nguồn:

```
make clean
```

Để xóa tất cả các tập tin trừ mã nguồn:

```
make cleanall
```

Cài đặt

Để áp dụng tập tin overlay cây thiết bị (nếu bmp180 chưa có sẵn trong cây thiết bị), sử dụng:

```
sudo dtoverlay i2c_bmp180.dtbo
```

Để gỡ bỏ overlay cây thiết bị, khởi động lại hoặc sử dụng:

```
sudo dtoverlay -r i2c_bmp180
```

Để cài đặt mô-đun:

```
sudo insmod i2c_bmp180.ko
```

Để gỡ bỏ mô-đun:

```
sudo rmmod i2c_bmp180
```
