

Microcontroller Lab 03 & 04

Development and deployment of a firmware driver for I2C and SPI on STM32 microcontroller to communicate with bosch temperature and pressure using sensor BMP280

Submitted by:

Kazi Shadman Sakib

FH-97

3rd Year

Computer Science and Engineering, University of Dhaka

Course Code: CSE-3116

Date of Submission: 3rd July, 2022

I2C

The I2C protocol stands for inter-integrated circuit. Basically, it's a bus interface connection protocol that is incorporated into devices for serial communication. It utilizes two bidirectional open-drain lines for communication known as SDA (Serial Data) and SCL (Serial Clock), and is also known as a two-wired interface.

I2C Operation Technique

The four operating modes for the interface are master transmitter, master receiver, slave transmitter and slave receiver.

A microcontroller is in charge of writing the initial configurations to the BMP280 sensor before receiving the raw measurement values in order to read temperature and pressure values. Since the microcontroller should be in charge of starting communication, it must function in master mode (by sending a start condition to the sensor).

I2C Pin Configuration

<u>Sensor Pin</u>	<u>Microcontroller Pin</u>
VCC	3.3V
GND	GND
SCL	PB8
SDA	PB7
CSB	3.3V
SDO	GND

I2C Observation

To effectively set up an I2C connection, we had to overcome a number of difficulties.

First of all, the humidity parameter was returning erratic results. Later, it was discovered that the sensor was actually a BMP280 and not a BME280 by examining the chip ID of the sensor. Humidity detection is not a feature offered by BMP280 sensors.

Additionally, the initial configuration of the sensor was challenging since improper addressing made the program get stuck in an endless loop, which stopped the sensor from transmitting acknowledgements.

By carefully and repeatedly consulting the sensor's datasheet, the problem was solved since addressing issues were corrected.

SPI

A synchronous serial communication interface specification is called the Serial Peripheral Interface (SPI). It is a four-wire bus, consisting of a serial clock, master input/slave output, master output/slave input and a chip selector pin.

SPI Operation Technique

The interface is set up as a master, therefore it gives the slave device the serial communication clock (in this case, the BMP280 sensor).

SPI Pin Configuration

<u>Sensor Pin</u>	<u>Microcontroller Pin</u>
VCC	3.3V
GND	GND
SCL	PC7
SDA	PC3
CSB	PA5
SDO	PC2

SPI Observation

The conflict during humidity measurement occurred here as well.

During the coding stage, another problem slowed down the progress. The pin configuration provided in the assignment description (PC1, PC2 and PC7 for MOSI (SDA), MISO (SDO) and SCK (SCL) respectively) seems to be a little flawed, as the same code that malfunctioned in the given configuration worked correctly upon the change of any one of these pins. After referring to the datasheet of STM32F446xx, we replaced PC1 with PC3.