

## I2C Role Play: Writing and Reading

I2C is just a formalized system of electronic communication that mirrors good communication in everyday life. The purpose of this role play is to emphasize how this is so—and to help you gain a deeper understanding of what bits are passed back and forth during I2C communication. Here, we focus on a specific device (Bosch BNO055 sensor fusion orientation sensor). Similar concepts apply more generally to other devices, but do be aware, the specific register addressing and data bits are device-specific. The only way to get them is to very carefully peruse the datasheet.

### Learning to WRITE

Example of an I<sup>2</sup>C write access to the BNO055 (i2c address in this case: 0101000b = 0x28):

Start	Slave address								RW	ACKS	dummy	Register address (0x00 .. 0x7F)								ACKS	Data								ACKS	Stop		
S	0	1	0	1	0	0	0	0	0	A	x	x	x	x	x	x	x	x	x	A	x	x	x	x	x	x	x	x	x	x	A	P

Figure 6: I<sup>2</sup>C write

Figure 1. . Example I2C write with Bosch BNO055 sensor. See page 93 of the data sheet (section 4.6)

*All peripherals sleeping, head on desk (or similar).*

Controller: Rise and shine, peripherals. All of you. We have some work to do measuring orientation, temperature, pH. It's gonna be fun! **[start condition]**

First, I need to **write** some device settings to the Bosch BNO025 sensor. Anyone available at I2C address **0x29**? Please acknowledge, if so. If it's not you, feel free to go back to bed. **[Addressing the peripheral]**

**Bosch sensor : That's me; 0x29 present and ready! [Acknowledge]**

Controller: Great, I need to configure the accelerometer range settings to +/- 4G; Bandwidth = 500 Hz; Operation Mode = normal. Reg Addr: ACC\_Config = 0x08 (section 4.4); Reg Value = 00011001 (Table 3.8). Before we get there, we gotta get on the same page.

So, I need you to flip to page 1 of your register map. Set your PAGE\_ID (register address 0x07) to 0x01. **[write register address]**. Give me a shout once you've the memory register address, cool?

**Bosch sensor: Bingo, got it! [Acknowledge]**

Controller: Excellent. Now I'm sending you 0x01 which you need to write into your PAGE\_ID memory register **[write data]**.

**Bosch sensor: Loud and clear, I'm on page 01, boss! [Acknowledge]**

Controller: That's great. We're finished for now, take a quick rest if you want. But let's talk again soon! **[stop condition]**

*Bosch sensor goes back to sleep.*

Controller: Rise and shine, peripherals. All of you. We have some work to do measuring orientation, temperature, pH. It's gonna be fun! **[start condition]**

Bosch Sensor, you still there? Sorry to bug you. It's me again. Now that we're on the same page, I need to configure some device settings. Are you available at I2C address 0x29? Please acknowledge, if so. If you're not 0x29, you can go back to bed. **[Addressing the peripheral]**

**Bosch sensor : Still here; 0x29 present and ready! [Acknowledge]**

Controller: Hey there. Sorry I woke you; coffee and croissants are on me next weekend. Anyway, we gotta get moving (or more like measure how some object is moving!).

We're now actually gonna configure your accelerometer settings now for: +/- 4G; Bandwidth; = 500 Hz; Operation Mode = normal. Reg Addr: ACC\_Config = 0x08 (section 4.4); Reg Value = 00011001 (Table 3.8).

First, here is register address I need to write: 0x08 = 00001000b. Sending these bits now. Please acknowledge. **[write register address]**

**Bosch sensor: Roger 10-4, good buddy! [acknowledge]**

Controller: Ole Bosch sensor fusion, pal. I knew I could count on you! The data I need to write to this register is: 00011001b. Sending these bits your way at 1/3 the speed of light. **[write data]**

**Bosch sensor: Loud and clear, all bits present and accounted for. All system are go! [acknowledge]**

Controller: Smooth as butter. Be well and let's talk again soon! I'm sure you'll have lots of data to send my way! **[stop condition]**

## Learning to READ

Now it is your turn. Carefully review the Bosch BNO055 datasheet. Develop a script that emphasizes the back and forth communication between the controller and the peripheral. Use the register maps to fill in the specific details of what bits are actually being sent, specifically to read the acceleration in the x direction.

Example of an I<sup>2</sup>C read access to the BNO055:



Figure 7: I<sup>2</sup>C multiple read

Figure 2. Example I2C read with Bosch BNO055 sensor. See page 93 of the data sheet (section 4.6)