

```
}

/*!
 * @brief This API writes the given data to the register address
 * of the sensor.
 */
int8_t bme680_set_regs(const uint8_t *reg_addr, const uint8_t *reg_data, uint8_t len, struct bme680_dev *dev)
{
    int8_t rslt;
    /* Length of the temporary buffer is 2*(length of register)*/
    uint8_t tmp_buff[BME680_TMP_BUFFER_LENGTH] = { 0 };
    uint16_t index;

    /* Check for null pointer in the device structure*/
    rslt = null_ptr_check(dev);
    if (rslt == BME680_OK) {
        if ((len > 0) && (len < BME680_TMP_BUFFER_LENGTH / 2)) {
            /* Interleave the 2 arrays */
            for (index = 0; index < len; index++) {
                if (dev->intf == BME680_SPI_INTF) {
                    /* Set the memory page */
                    rslt = set_mem_page(reg_addr[index], dev);
                    tmp_buff[(2 * index)] = reg_addr[index] & BME680_SPI_WR_MSK;
                } else {
                    tmp_buff[(2 * index)] = reg_addr[index];
                }
                tmp_buff[(2 * index) + 1] = reg_data[index];
            }
            /* Write the interleaved array */
            if (rslt == BME680_OK) {
                dev->com_rslt = dev->write(dev->dev_id, tmp_buff[0], &tmp_buff[1], (2 * len) - 1);
                if (dev->com_rslt != 0)
                    rslt = BME680_E_COM_FAIL;
            }
        } else {
            rslt = BME680_E_INVALID_LENGTH;
        }
    }
}
```