```c
/*!
 * @brief This API is used to set the oversampling, filter and T,P,H, gas selection
 * settings in the sensor.
 */
int8_t bme680_set_sensor_settings(uint16_t desired_settings, struct bme680_dev *dev)
{
    int8_t rslt;
    uint8_t reg_addr;
    uint8_t data = 0;
    uint8_t count = 0;
    uint8_t reg_array[BME680_REG_BUFFER_LENGTH] = { 0 };
    uint8_t data_array[BME680_REG_BUFFER_LENGTH] = { 0 };
    uint8_t intended_power_mode = dev->power_mode; /* Save intended power mode */

    /* Check for null pointer in the device structure*/
    rslt = null_ptr_check(dev);
    if (rslt == BME680_OK) {
        if (desired_settings & BME680_GAS_MEAS_SEL)
            rslt = set_gas_config(dev);

        dev->power_mode = BME680_SLEEP_MODE;
        if (rslt == BME680_OK)
            rslt = bme680_set_sensor_mode(dev);

        /* Selecting the filter */
        if (desired_settings & BME680_FILTER_SEL) {
            rslt = boundary_check(&dev->tph_sett.filter, BME680_FILTER_SIZE_0, BME680_FILTER_SIZE_127, dev);
            reg_addr = BME680_CONF_ODR_FILT_ADDR;

            if (rslt == BME680_OK)
                rslt = bme680_get_regs(reg_addr, &data, 1, dev);

            if (desired_settings & BME680_FILTER_SEL)
                data = BME680_SET_BITS(data, BME680_FILTER, dev->tph_sett.filter);

            reg_array[count] = reg_addr; /* Append configuration */
```