```
    /* Selecting humidity oversampling for the sensor */
    if (desired_settings & BME680_OSH_SEL) {
        rslt = boundary_check(&dev->tph_sett.os_hum, BME680_OS_NONE, BME680_OS_16X, dev);
        reg_addr = BME680_CONF_OS_H_ADDR;

        if (rslt == BME680_OK)
            rslt = bme680_get_regs(reg_addr, &data, 1, dev);
        data = BME680_SET_BITS_POS_0(data, BME680_OSH, dev->tph_sett.os_hum);

        reg_array[count] = reg_addr; /* Append configuration */
        data_array[count] = data;
        count++;
    }

    /* Selecting the runGas and NB conversion settings for the sensor */
    if (desired_settings & (BME680_RUN_GAS_SEL | BME680_NBCONV_SEL)) {
        rslt = boundary_check(&dev->gas_sett.run_gas, BME680_RUN_GAS_DISABLE,
            BME680_RUN_GAS_ENABLE, dev);
        if (rslt == BME680_OK) {
            /* Validate boundary conditions */
            rslt = boundary_check(&dev->gas_sett.nb_conv, BME680_NBCONV_MIN,
                BME680_NBCONV_MAX, dev);
        }

        reg_addr = BME680_CONF_ODR_RUN_GAS_NBC_ADDR;

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

        if (desired_settings & BME680_RUN_GAS_SEL)
            data = BME680_SET_BITS(data, BME680_RUN_GAS, dev->gas_sett.run_gas);

        if (desired_settings & BME680_NBCONV_SEL)
            data = BME680_SET_BITS_POS_0(data, BME680_NBCONV, dev->gas_sett.nb_conv);

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