

```

        rslt = set_mem_page(reg_addr1, dev);
    }

    if (rslt == BME680_OK) {
        rslt = bme680_get_regs(reg_addr1, &reg_data, 1, dev);
        if (rslt == BME680_OK) {
            dev->gas_sett.heatr_temp = reg_data;
            rslt = bme680_get_regs(reg_addr2, &reg_data, 1, dev);
            if (rslt == BME680_OK) {
                /* Heating duration register value */
                dev->gas_sett.heatr_dur = reg_data;
            }
        }
    }
}

return rslt;
}

#ifdef BME680_FLOAT_POINT_COMPENSATION

/*!
 * @brief This internal API is used to calculate the temperature value.
 */
static int16_t calc_temperature(uint32_t temp_adc, struct bme680_dev *dev)
{
    int64_t var1;
    int64_t var2;
    int64_t var3;
    int16_t calc_temp;

    var1 = ((int32_t) temp_adc >> 3) - ((int32_t) dev->calib.par_t1 << 1);
    var2 = (var1 * (int32_t) dev->calib.par_t2) >> 11;
    var3 = ((var1 >> 1) * (var1 >> 1)) >> 12;
    var3 = ((var3) * ((int32_t) dev->calib.par_t3 << 4)) >> 14;
    dev->calib.t_fine = (int32_t) (var2 + var3);
    calc_temp = (int16_t) (((dev->calib.t_fine * 5) + 128) >> 8);
}

```