

```
static void init_BME680(void);
void user_delay_ms (uint32_t period);
static uint8_t spi_transfer (uint8_t data);
int8_t user_spi_read (uint8_t dev_id, uint8_t reg_addr, uint8_t *reg_data, uint16_t len);
int8_t user_spi_write (uint8_t dev_id, uint8_t reg_addr, uint8_t *reg_data, uint16_t len);

int main(void) {
    int KeyPress=0x04, Last_KeyPress, count=0;
    uint16_t set_required_settings, meas_period;
    //initialize SERCOM4 for RS232 communication and SERCOM1 for the LCD and BME680
    UART4_init();
    init_lcd_dog();
    init_spi_BME680();
    init_BME680();

    struct bme680_dev gas_sensor;           //create instance of bme680_dev named gas_sensor
    gas_sensor.dev_id = 0;                  //fill in various parameters for gas_sensor
    gas_sensor.intf = BME680_SPI_INTF;
    gas_sensor.read = user_spi_read;
    gas_sensor.write = user_spi_write;
    gas_sensor.delay_ms = user_delay_ms;
    gas_sensor.amb_temp = 25;
    int8_t rslt = BME680_OK;
    rslt = bme680_init(&gas_sensor);

    //set the temperature, pressure and humidity oversampling. set IIR filter size
    gas_sensor.tph_sett.os_hum = BME680_OS_2X;
    gas_sensor.tph_sett.os_pres = BME680_OS_4X;
    gas_sensor.tph_sett.os_temp = BME680_OS_8X;
    gas_sensor.tph_sett.filter = BME680_FILTER_SIZE_3;

    //enable gas measurements and configure gas heat plate temperature and heating duration
    gas_sensor.gas_sett.run_gas = BME680_ENABLE_GAS_MEAS;
    gas_sensor.gas_sett.heatr_temp = 320;
    gas_sensor.gas_sett.heatr_dur = 150;

    //put BME680 into forced mode
```