```
int8_t user_spi_write (uint8_t dev_id, uint8_t reg_addr, uint8_t *reg_data, uint16_t len);

int main(void) {
    uint8_t meas_stat = 0x20;
    uint32_t temp;
    init_spi_MCU();                             //initialize MCU and BME680 SPI communication
    init_spi_BME680();
    user_spi_write(0, 0x72, (void *)0, 1);      //turn off humidity reading
    user_spi_write(0, 0x74, (void *)0x20, 1);   //turn off pressure reading, temperature oversampling x1, sleep mode
    user_spi_write(0, 0x75, (void *)0x04, 1);   //IIR filter coefficient = 1, SPI 4 wire mode
    user_spi_write(0, 0x70, (void *)0x08, 1);   //turn off heater

    while (1) {
        user_delay_ms(1000);                        //delay 1s
        user_spi_write(0, 0x74, (void *)0x21, 1);           //enable forced mode
        while (meas_stat & 0x20) {                  //poll measuring status flag
            user_spi_read(0, 0x1D, &meas_stat, 1);
        }
        user_spi_read(0, 0x22, &temp, 1);           //read 3 temperature data registers
        temp <<= 8;
        user_spi_read(0, 0x23, &temp, 1);
        temp <<= 8;
        user_spi_read(0, 0x24, &temp, 1);
        temp >>= 4;
        raw_temperature = temp;                     //store in raw_temperature
    }
}

void init_spi_MCU (void) {
    REG_GCLK_PCHCTRL19 = 0x00000040;    /* SERCOM1 core clock not enabled by default */

    ARRAY_PORT_PINCFG0[16] |= 1;    /* allow pmux to set PA16 pin configuration */
    ARRAY_PORT_PINCFG0[17] |= 1;    /* allow pmux to set PA17 pin configuration */
    ARRAY_PORT_PINCFG0[18] |= 1;    /* allow pmux to set PA18 pin configuration */
    ARRAY_PORT_PINCFG0[19] |= 1;    /* allow pmux to set PA19 pin configuration */
    ARRAY_PORT_PMUX0[8] = 0x22;     /* PA16 = MOSI, PA17 = SCK */
    ARRAY_PORT_PMUX0[9] = 0x22;     /* PA18 = SS,   PA19 = MISO */
```