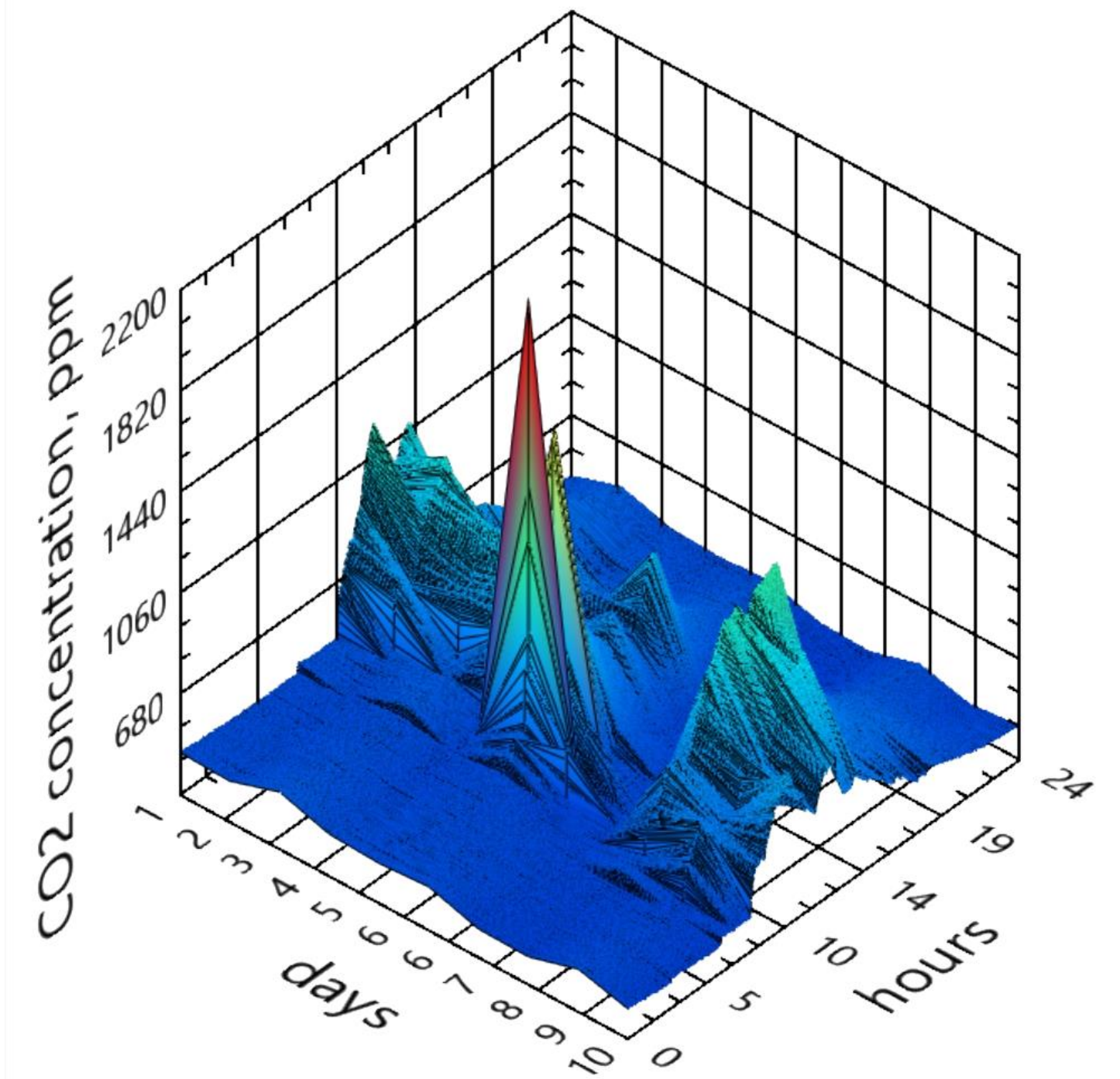
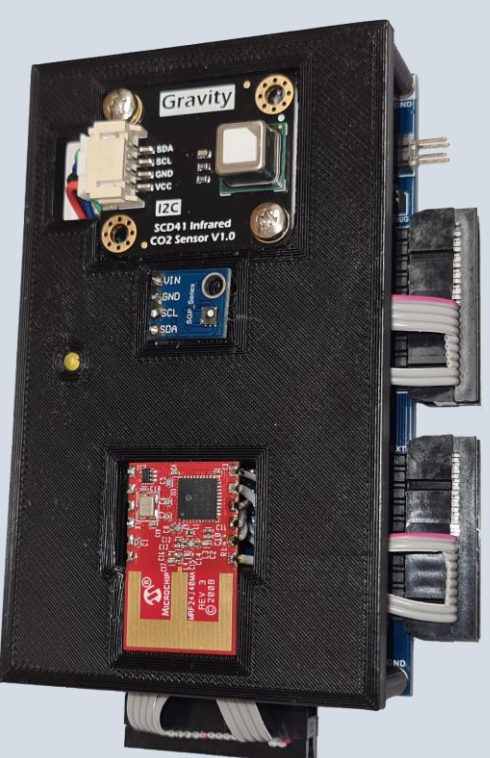


Zaprojektowanie i uruchomienie systemu bezprzewodowej sieci sensorycznej do określania jakości powietrza wewnętrznego w wielu pomieszczeniach

Współczesny człowiek średnio statystycznie spędza około 80% swojego życia wewnątrz pomieszczeń. Jakość powietrza wewnętrznego wpływa na nasze samopoczucie, zdrowie, możliwość koncentracji, wydajność pracy oraz efektywność odpoczynku. Skład powietrza wewnętrznego jest jednym z głównych czynników określających warunki mikroklimatu panującego w pomieszczeniach. Znajomość parametrów powietrza takich jak stężenie gazów i temperatura, pozwala na podejmowanie decyzji o zmianie nastaw systemu wentylacji lub klimatyzacji w celu poprawy jakości powietrza.

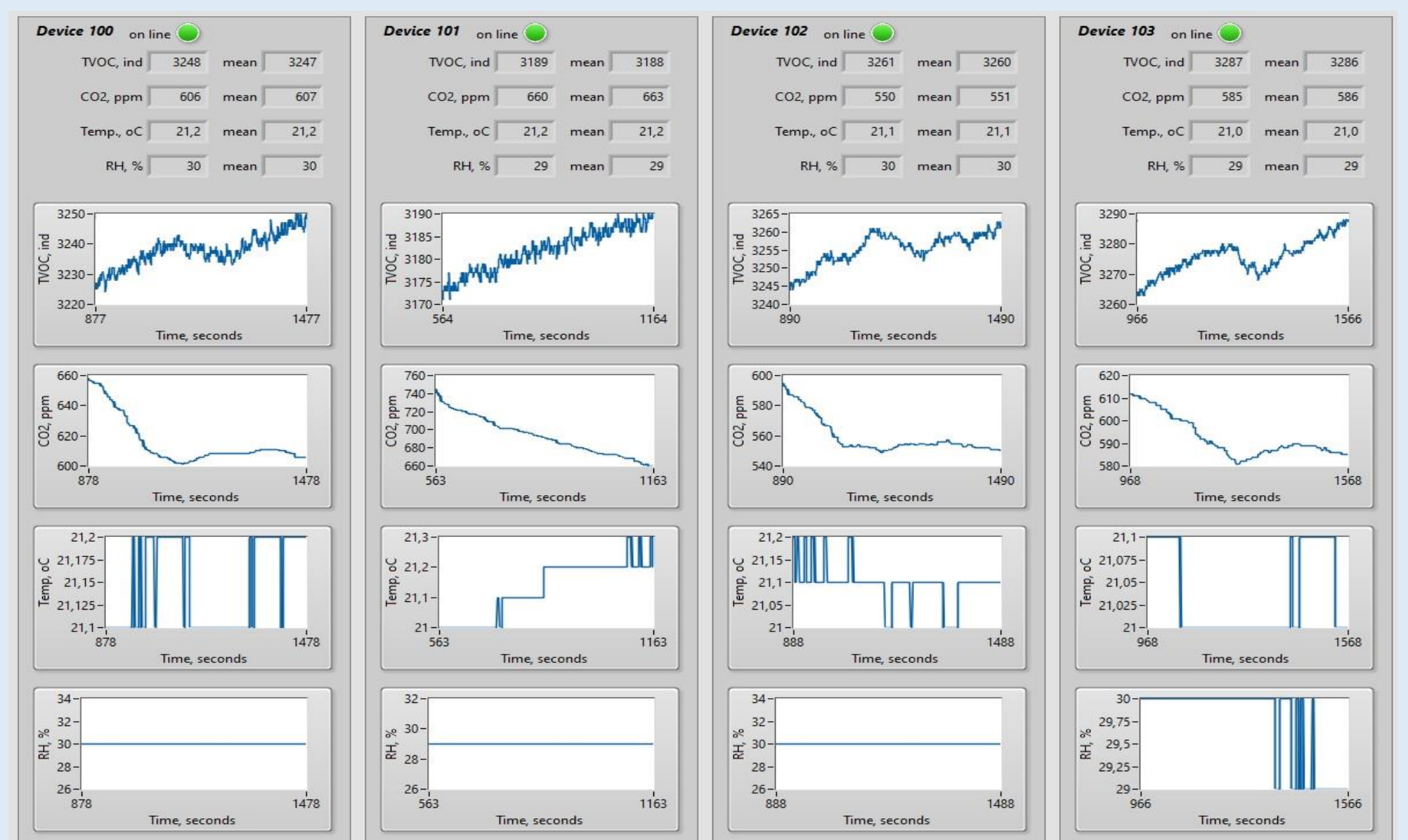


Dane pomiarowe z przetworników pomiarowych nadane do koordynatora drogą radiową przesyłane są do komputera poprzez łącze transmisji szeregową USB. Podczas pomiarów na komputerze jest uruchomiony program „Monitor” przechwytyjący dane z sieci sensorycznej i zapisujący je do pliku pomiarowego w formacie TDMS (Technical Data Management Streaming). Wybrany format zapisu danych do pliku pozwala na jego późniejsze wykorzystanie w wielu popularnych programach użytkowych, takich jak edytory baz danych, arkusze kalkulacyjne lub programy do analizy danych.



Moduł pomiarowy

W ramach projektu planowane było stworzenie bezprzewodowej sieci sensorycznej, której elementami pomiarowymi będą przetworniki opracowane w poprzednim projekcie PBL. Przetworniki pomiarowe rozmieszczone w różnych pomieszczeniach, umożliwiają pomiar wartości stężenia dwutlenku węgla CO₂, stężenia lotnych związków organicznych VOC, temperatury i wilgotności względnej RH powietrza. Zmierzone wartości są przesyłane drogą radiową w pasmie 2,4GHz w protokole ZigBee do modułu centralnego podłączonego do komputera.



Program „Monitor” wyświetlający dane pomiarowe z czterech przetworników

Oprogramowanie wewnętrzne mikrokontrolerów napisano w języku C w środowisku programowym Microchip Studio firmy Microchip.

```
void Fun_SPI_Initialization (void) {
    uint32_t
    LocVAR_UInt;

    // clock initialization
    GCLK->PCHCTRL[SERCOS_GCLK_ID_CORE].reg = GCLK_PCHCTRL_GEN_GCLK0_Val | (1 << GCLK_PCHCTRL_CHEN_Pos);
    GCLK->PCHCTRL[SERCOS_GCLK_ID_SLOW].reg = GCLK_PCHCTRL_GEN_GCLK3_Val | (1 << GCLK_PCHCTRL_CHEN_Pos);
    MCLK->APBCHASK.reg |= MCLK_APBCHASK_SERCOS;

    //SPI_0_init();
    if (!SERCOS->SPI_SYNCBUSY.bit.SMRST) {
        if (SERCOS->SPI_CTRLA.bit.ENABLE) {
            SERCOS->SPI_CTRLA.bit.ENABLE = false;
            while (SERCOS->SPI_SYNCBUSY.bit.ENABLE);
        }
        SERCOS->SPI_CTRLA.bit.SMRST = true;
    }
    while (SERCOS->SPI_SYNCBUSY.bit.SMRST);

    SERCOS->SPI_CTRLA.reg = (0 << SERCOM_SPI_CTRLA_DORD_Pos /* Data Order: disabled */
    | 0 << SERCOM_SPI_CTRLA_CPOL_Pos /* Clock Polarity: disabled */
    | 0 << SERCOM_SPI_CTRLA_CPHA_Pos /* Clock Phase: disabled */
    | 0 << SERCOM_SPI_CTRLA_FOUP_Pos /* Frame Format: 0 */
    | 0 << SERCOM_SPI_CTRLA_I200N_Pos /* Immediate Buffer Overflow Notification: disabled */
    | 0 << SERCOM_SPI_CTRLA_RUNSTDBY_Pos /* Run In Standby: disabled */
    | 3 << SERCOM_SPI_CTRLA_MODE_Pos); /* Operating Mode: 3 */

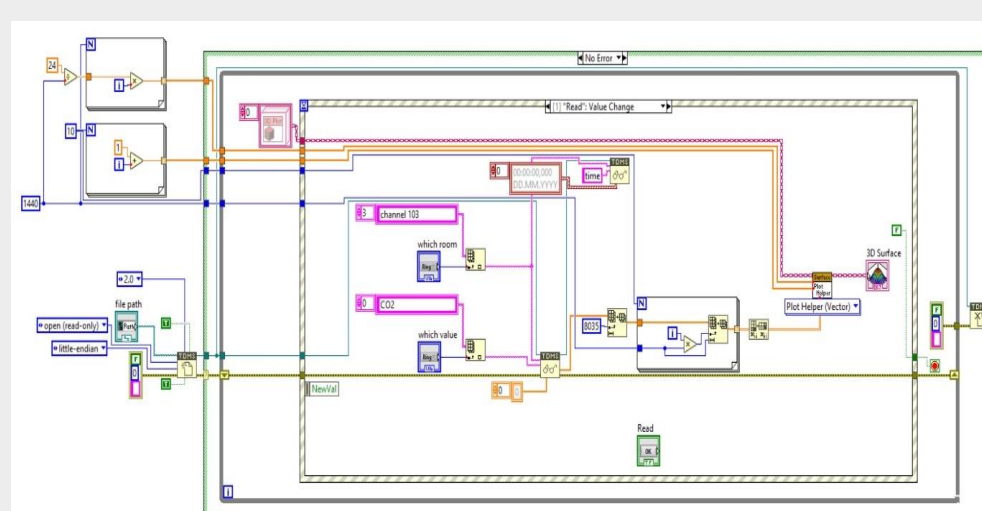
    //spi_sercomspi_write_CTRLA_DORD_bf(SERCOS, SERCOS_TXPO);
    LocVAR_UInt = SERCOS->SPI_CTRLA.reg;
    LocVAR_UInt &= ~SERCOS_SPI_CTRLA_DORD_Msk;
    LocVAR_UInt |= (0x01 << SERCOM_SPI_CTRLA_DORD_Pos);
    SERCOS->SPI_CTRLA.reg = LocVAR_UInt;
    while (SERCOS->SPI_SYNCBUSY.bit.SMRST);
    while (SERCOS->SPI_SYNCBUSY.bit.ENABLE);

    LocVAR_UInt = SERCOS->SPI_CTRLA.reg;
    LocVAR_UInt &= ~SERCOS_SPI_CTRLA_DIPO_Msk;
    LocVAR_UInt |= (0x00 << SERCOM_SPI_CTRLA_DIPO_Pos);
    SERCOS->SPI_CTRLA.reg = LocVAR_UInt;
    while (SERCOS->SPI_SYNCBUSY.bit.SMRST);
    while (SERCOS->SPI_SYNCBUSY.bit.ENABLE);

    SERCOS->SPI_CTRLB.reg = ((1 << SERCOM_SPI_CTRLB_RXEN_Pos /* Receiver Enable: enabled */
    | (0 << SERCOM_SPI_CTRLB_RXSEN_Pos) /* Master Slave Select Enabl: disabled */
    | (0 << SERCOM_SPI_CTRLB_AMODE_Pos) /* Address Mode: 0 */
    | (0 << SERCOM_SPI_CTRLB_SSDE_Pos) /* Slave Select Low Detect Enable: disabled
```

Fragment oprogramowania wewnętrznego

Programy komputerowe „Monitor” i „Viewer” stworzono w środowisku programowym LabView firmy National Instrument (Emerson).



Fragment oprogramowania „Viewer”

Elementami bezprzewodowej sieci sensorycznej są mikroprocesorowe przetworniki pomiarowe.



Mikrokontroler będący częścią przetwornika pomiarowego