FH
CAMPUS
WIEN
UNIVERSITY OF APPLIED SCIENCES

Design document

Group Nr. 3

Timur Ali Basnakajev
Tobias König
Jessica Marban
Patrick Neumann

Contents

1 Project description

1.1 Problem desription

When the CO2 level in a room reaches a certain level, a window should automatically open and a fresh air fan should switch on. After a certain lower CO2 level is reached and a certain run-on time has elapsed, the window is closed again and the fan is switched off again.

1.2 Hardware

1. MH-Z19 CO2 Sensor
2. Arduino with bluetooth module
3. Raspberry
4. Electric motor

1.3 Hardware

1. Analog input for measuring the CO2 contentr
2. Digital output for controlling a window opener
3. Digital output to control a fan

1.4 Schema

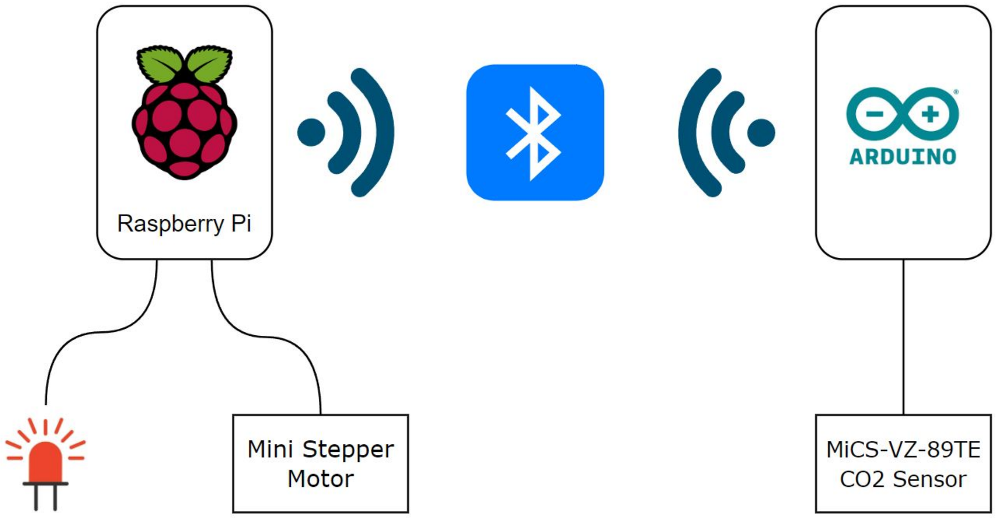

Figure 1: system description

2 Software architecture and design

2.1 Software modules

Safety related modules

1. Modul x:
   Description:
   Functions:
   Data:
   Requirements see: 1.1., ...

Security related modules

Modules with no influence on Safety and Security

2.2 Libraries

Description of used function with parameters.

2.3 Interrupts

Definition of priorities.

2.4 Pinout

3 Program flowchart

Hier bitte Sequenzdiagramme, bzw. Programmablaufdiagramme

4 Hazard identification

4.1 Identified hazards and countermeasures

1. Hazard 1:

   Co2 sensor delivers incorrect values 2.

2. Hazard 2:

   The window is stuck and does not open or close 3.

3. Hazard 3:

   Arduino fails 4.

4.2 Identified hazards without countermeasures

1. Hazard 1:

   The Co2 sensor fails and does not provide data

2. Hazard 2:

   The power supply fails

3. Hazard 3:

   The engine fails

5 Threat identification

5.1 Identified threats and countermeasures

1. **Threat 1:** Someone could manipulate the window this way, intentionally or unintentionally that it no longer closes after opening. This could result in a break-in consequences.

   Mitigation: The device carries out a self-test when commissioning (open 1 x and close 1 x)

2. **Threat 2**: The window couldn't open because someone put something in front of the window. This means there is no ventilation. The system no longer works.

   Mitigation: Refer to mitigation of Threat 1:

3. **Threat 3**: DoS: Someone could carry out a DoS attack on the Rasperry PI. This will prevents emails from being sent.

   Mitigation: Installation of a firewall. This means that the Rasperry PI cannot be reached from outside.

5.2 Identified threats without countermeasures

1. **Threat 1:** Someone could try to deliberately trigger the CO 2 sensor. This will open the window opened and a break-in could be carried out through the window.

2. **Threat 2:** Someone could install a jammer and thus disrupt the connection between Rasperry PI and Arduino and thus, for example, prevent the window from closing to carry out a break-in.

3. **Threat 3:** Someone could install a jammer and prevent the safety circuit from tripping. This means the system no longer works.

6 Requirements

6.1 Safety related requirements

1. Requirement:
   At program start all safety related functions must be tested.

   1.1. Requirement:
   If an error occurs during program start, this must be indicated by an LED (flashing at 1 second intervals)

   1.2. Requirement:
   Communication between Raspberry and Arduino must be established.

   1.3. Requirement:
   When the system starts, the window must be successfully opened and closed once.

2. Requirement:
   The values of the CO2 sensor must be continuously checked for their validity 1:

   2.1. Requirement:
   If the sensor value is outside the typical range, the value must be marked as invalid.

   2.2. Requirement:
   If the sensor values are invalid an email must be sent.

3. Requirement:
   It must be checked whether the window has been opened or closed sufficiently 2:

   3.1. Requirement:
   If the measured position of the stepper motor deviates by 5 mm when opening, the motor must stop immediately and go into error state (obstacle was detected).

   3.2. Requirement:
   If the measured position of the stepper motor deviates by 5 mm when closing, the motor must stop immediately and the window must open again (obstacle has been detected).

3.3. Requirement:
If an obstacle is detected during opening or closing, the system must report an error (email and LED).

4. Requirement:
It must be checked whether the Arduino is working properly 1:

4.1. Requirement:
The communication between Raspberry and Arduino must be checked cyclically (10 second interval).

4.2. Requirement:
If the communication between the Raspberry and the Arduino is interrupted for more than 30 seconds, the system must go into error state (Email and LED).

6.2 Security related requirements

1. Requirement:
The communication between the Raspberry and the Arduino must be encrypted.

2. Requirement:
The Raspberry PI must be hardened against cybersecurity attacks.

6.3 Requirements with no influence on Safety and Security

1. Requirement:
The system should be installed in an inaccessible location.