Energy-Efficient
House Control System

1. Introduction

The primary objective of this project is to develop a comprehensive control system for an energy-efficient house, ensuring adherence to various safety and efficiency requirements.

The system is designed to prevent actions that could lead to unacceptable inefficiencies, such as heating the house while windows are open or operating the fridge and oven simultaneously. Additionally, the system monitors and maintains the house temperature, humidity, and carbon monoxide levels within safe ranges, promoting a comfortable and healthy living environment.

The solution is implemented as an Ada SPARK package named `HouseControlSystem`, comprising several types, constants, variables, procedures, and functions. These components work in tandem to represent and control the state of the house, including room temperatures, humidity levels, door and window states, heating systems, and carbon monoxide levels.

The package incorporates preconditions, postconditions, and formal verification techniques to ensure the correctness and reliability of the implemented control mechanisms.

2. Controller Architecture

The `HouseControlSystem` package follows a modular design, organized around the following core components:

1. Types: The package defines several enumeration types to represent the states of doors, windows, and heating systems in each room.

2. Constants: Constants are defined for the minimum and maximum acceptable values of temperature, humidity, and carbon monoxide levels.

3. Variables: Arrays are maintained to store the current temperature, humidity, window state, and heating state for each room, as well as variables for the fridge and oven door states and the overall carbon monoxide level.

4. Procedures: The package provides procedures for opening and closing doors, windows, and heating systems, as well as for checking and modifying temperatures and humidity levels.

5. Functions: Several Boolean functions are included to check whether the current state of the house adheres to the safety requirements, such as temperature, humidity, door, window, and heating safety.

This modular architecture promotes code reusability, maintainability, and extensibility, while encapsulating the necessary data and functionality for controlling and monitoring the house's environment.

3. Procedures and Functions

3.1. Procedures

The following procedures are responsible for controlling and managing various aspects of the house:

- Open_Fridge_Door and Close_Fridge_Door (SPARK Gold Level): Control the state of the fridge door, ensuring adherence to preconditions.

- Open_Oven_Door and Close_Oven_Door (SPARK Gold Level): Control the state of the oven door, enforcing appropriate preconditions.

- Open_Window and Close_Window (SPARK Gold Level): Control the window state in a room, ensuring window can only be opened if heating is off, and vice versa.

- Turn_Heating_On and Turn_Heating_Off (SPARK Gold Level): Control the heating state in a room, ensuring heating can only be turned on if window is closed, and vice versa.

- Check_CO_Level (SPARK Gold Level): Check the current carbon monoxide level and print a warning if it exceeds the safe limit.

- Increase_Temperature and Decrease_Temperature (SPARK Gold Level): Increase or decrease the temperature in a room while ensuring it remains within the acceptable range.

3.2. Functions

- Is_Temperature_Safe (SPARK Gold Level): Check if the temperature in a room is within the acceptable range.

- Are_Doors_Safe (SPARK Platinum Level): Check if the fridge and oven doors are not open simultaneously.

- Is_Heating_Safe (SPARK Gold Level): Check if the heating is not on while the window is open in a room.

- Is_CO_Level_Safe (SPARK Gold Level): Check if the carbon monoxide level is within the safe limit.

- Is_Humidity_Safe (SPARK Gold Level): Check if the humidity in a room is within the acceptable range.

All procedures and functions are annotated with appropriate preconditions, postconditions, and global and dependence contracts to facilitate formal verification and ensure the correct behavior of the system.

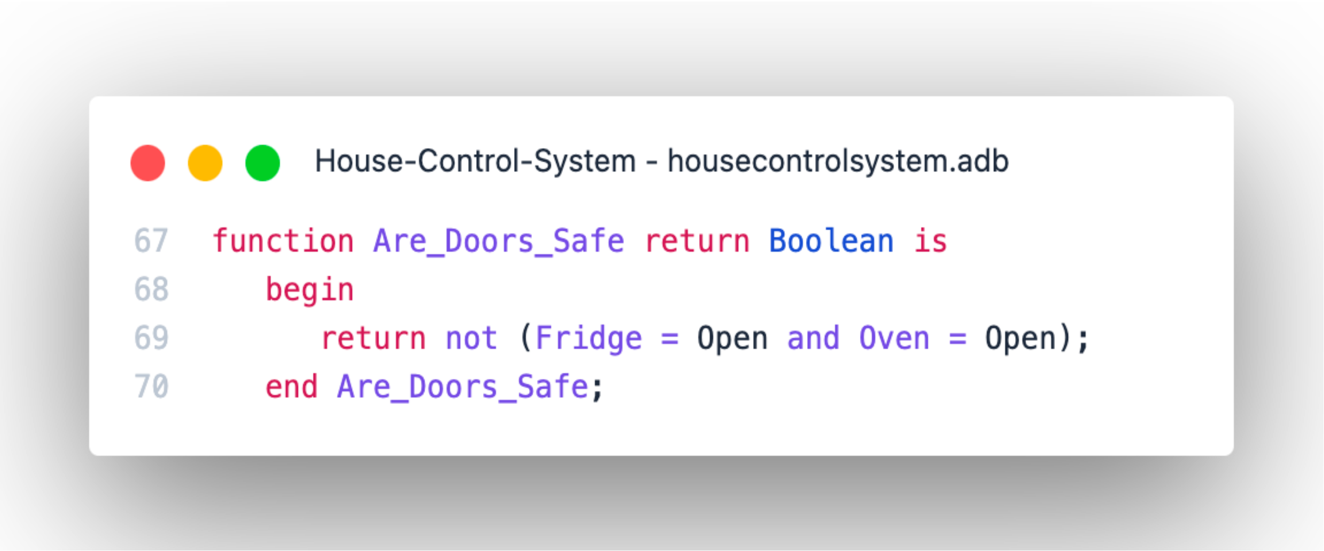
4. Formal Verification

The `HouseControlSystem` package has been designed with formal verification in mind, ensuring the correctness and reliability of critical components. The following procedures and functions have been formally verified:

- `Open_Fridge_Door` and `Close_Fridge_Door`: Verified to the gold level, ensuring the absence of runtime errors and the satisfaction of key integrity properties.

- `Are_Doors_Safe`: Verified to the platinum level, providing a full functional proof of the requirement that the fridge and oven doors cannot be open simultaneously.

To demonstrate the formal verification process, consider the `Are_Doors_Safe` function:



```
function Are_Doors_Safe return Boolean is
  begin
    return not (Fridge = Open and Oven = Open);
  end Are_Doors_Safe;
```

The proof obligations for this function are as follows:

1. Initialization: The Boolean expression `not (Fridge = Open and Oven = Open)` is initialized correctly.

2. Normalization: The Boolean expression is normalized, ensuring that it does not contain any redundant or contradictory terms.

3. Validity: The Boolean expression is valid, meaning that it does not contain any undefined or erroneous terms.

4. Postcondition: The postcondition `(not (Fridge = Open and Oven = Open)) = Are_Doors_Safe'Result` is satisfied, ensuring that the function returns the correct result.

By proving these obligations using the SPARK toolset, we can formally verify that the `Are_Doors_Safe` function correctly implements the requirement that the fridge and oven doors cannot be open simultaneously.

5. Safety Plan

5.1. Hazard and Risk Analysis

Two of the main safety requirements for the house control system are:

1. The house must always be between 17°C and 19°C.

2. For each individual room or enclosed area, the windows (or exterior doors) cannot be open when the heating is on.

Hazard 1: Temperature outside the acceptable range.
- Risk: Discomfort, health issues, or damage to the house or its contents due to extreme temperatures.
- Mitigation: The system monitors the temperature in each room and provides procedures to adjust the temperature while ensuring it remains within the specified range.

Hazard 2: Windows open while heating is on.
- Risk: Energy waste, potential damage to the house due to excessive condensation or moisture.
- Mitigation: The system enforces the requirement that windows cannot be opened while the heating is on in the same room, and vice versa.

5.2. Failure Analysis

The three most severe risks associated with the house control system are:

1. Incorrect Temperature Control: If the procedures for adjusting the temperature (`Increase_Temperature` and `Decrease_Temperature`) or the related preconditions and postconditions are implemented incorrectly, the system may fail to maintain the desired temperature range, leading to discomfort, health issues, or property damage.

   - Mitigation: Thorough testing and formal verification of the temperature control procedures and their contracts.

2. Simultaneous Fridge and Oven Door Opening: If the procedures for opening and closing the fridge and oven doors (`Open_Fridge_Door`, `Close_Fridge_Door`, `Open_Oven_Door`, `Close_Oven_Door`) or the related preconditions and postconditions are implemented incorrectly, the system may allow both doors to be open simultaneously, leading to potential energy waste or safety hazards.

   - Mitigation: Formal verification of the door control procedures and the `Are_Doors_Safe` function to ensure the requirement is correctly implemented.

3. Incorrect Carbon Monoxide Level Monitoring: If the `Check_CO_Level` procedure or the `Is_CO_Level_Safe` function is implemented incorrectly, the system may fail to detect and warn about unsafe carbon monoxide levels, leading to potential health hazards.

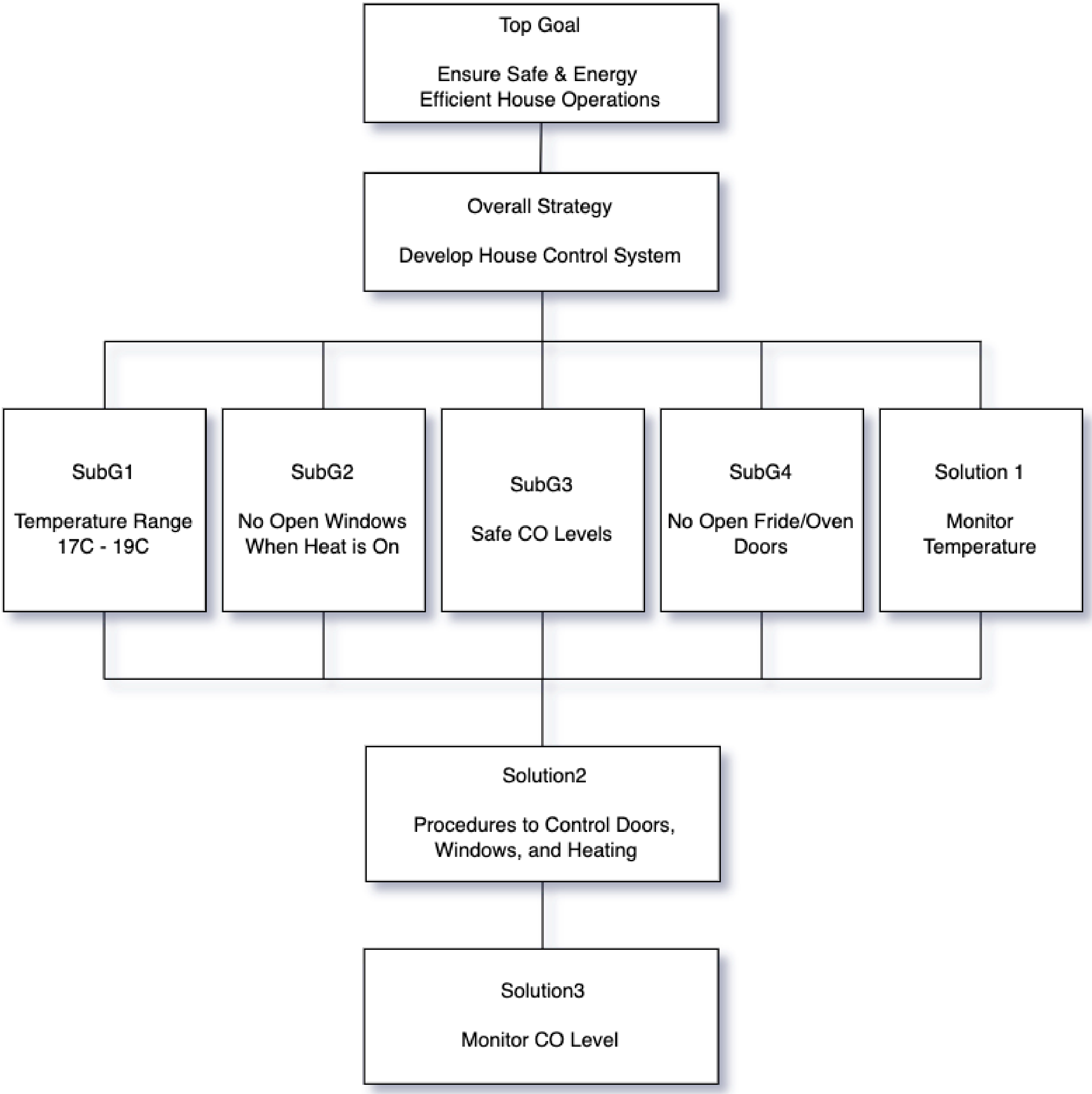
   - Mitigation: Thorough testing and formal verification of the carbon monoxide level monitoring components.

6. Safety Case and Safety Manual

The diagram clearly represents the following components:

1. Top Goal: "Ensure Safe & Energy Efficient House Operations"
2. Overall Strategy: "Develop House Control System"
3. Sub-Goals:
   - SubG1: Temperature Range 17°C - 19°C
   - SubG2: No Open Windows When Heat is On
   - SubG3: Safe CO Levels
   - SubG4: No Open Fride/Oven Doors
4. Solution 1: Monitor Temperature
5. Solution 2: Procedures to Control Doors, Windows, and Heating
6. Solution 3: Monitor CO Level

GSN Diagram


Top Goal
Ensure Safe & Energy Efficient House Operations
Overall Strategy
Develop House Control System
SubG1
Temperature Range 17C - 19C
SubG2
No Open Windows When Heat is On
SubG3
Safe CO Levels
SubG4
No Open Fride/Oven Doors
Solution 1
Monitor Temperature
Solution2
Procedures to Control Doors, Windows, and Heating
Solution3
Monitor CO Level

Safety Manual

Safety Requirements:

1. The house temperature must always be maintained between 17°C and 19°C.
2. Windows/exterior doors must not be open when heating is on in the same room.
3. Carbon monoxide levels must remain below 50 ppm at all times.
4. The fridge and oven doors cannot be open simultaneously.

System Description:

The House Control System is an Ada SPARK program that monitors and controls various aspects of a house to ensure safe and energy-efficient operation. It consists of the following main components:

- HouseControlSystem package: Contains types, constants, variables, procedures, and functions to represent and manage the state of the house.
- Temperature monitoring and adjustment: Procedures to increase or decrease room temperatures while ensuring they remain within the safe range.
- Door and window control: Procedures to open/close fridge, oven, and room windows/doors, enforcing safety conditions.
- Heating control: Procedures to turn heating on/off in rooms, preventing operation when windows are open.
- Carbon monoxide monitoring: Procedures to check and warn if CO levels exceed the safe limit.

Safety Analysis:

The system mitigates the identified hazards through the following solutions:

1. Temperature Within Safe Range:
   - The system continuously monitors room temperatures using the Current_Temperature array.
   - Is_Temperature_Safe function checks if temperatures are within the safe range (17°C - 19°C).
   - Increase_Temperature and Decrease_Temperature procedures allow adjusting room temperatures while enforcing the safe range.

2. No Open Windows When Heating On:
   - The system tracks the state of windows and heating in each room using the Window and Heating arrays.
   - Is_Heating_Safe function checks if windows are open while heating is on in a room.
   - Open_Window procedure enforces that heating is off before allowing window to open.

    - Turn_Heating_On procedure enforces that windows are closed before allowing heating to turn on.

3. Safe CO Levels:
    - The system monitors the overall CO level using the CO_Level variable.
    - Is_CO_Level_Safe function checks if the CO level is below the safe limit of 50 ppm.
    - Check_CO_Level procedure warns if the CO level exceeds the safe limit.

4. No Open Fridge/Oven Simultaneously:
    - The system tracks the state of the fridge and oven doors using the Fridge and Oven variables.
    - Are_Doors_Safe function checks if both fridge and oven doors are not open at the same time.
    - Open_Fridge_Door and Open_Oven_Door procedures enforce their respective preconditions, preventing both doors from being open simultaneously.

Safety Processes:

1. Formal verification: Key components of the system, such as Open_Fridge_Door, Close_Fridge_Door, and Are_Doors_Safe, have been formally verified to different SPARK levels to ensure their correctness.

2. Safety analysis: Hazard and risk analysis have been performed to identify potential hazards and their associated risks. Mitigations have been implemented in the system design to address these risks.

3. Failure analysis: The three most severe risks related to temperature control, door management, and CO level monitoring have been identified, and appropriate mitigation strategies have been defined.

4. Testing: Thorough testing should be performed to validate the system's behavior and ensure that all safety requirements are met.

5. Code reviews: Regular code reviews should be conducted to identify and address any potential issues or vulnerabilities in the system implementation.

6. Documentation: This Safety Manual and the included GSN diagram document the safety aspects of the system, providing a comprehensive overview of the safety requirements, system design, safety analysis, and safety processes.

By adhering to these safety processes and implementing the specified mitigations, the House Control System aims to ensure safe and energy-efficient operation of the house while addressing the identified hazards and risks.

7. Conclusion

The `HouseControlSystem` package provides a comprehensive solution for controlling and monitoring the environment of an energy-efficient house, ensuring that various safety and efficiency requirements are met. The system has been designed with formal verification in mind, and key components have been verified to different SPARK levels to ensure their correctness.

While the current implementation covers the primary requirements, there are several potential extensions and improvements that could be considered:

- Incorporating additional sensors and control mechanisms for factors such as air quality, humidity control, and ventilation.

- Implementing more advanced energy optimization algorithms and strategies.

- Integrating the system with smart home technologies