1. PROJECT

ROOT_OBJECTS
--|examples_AADL\CoffeeMachine|--,
--|examples_AADL/Common\Behavior_Properties|--,
--|examples_AADL/Common\Data_Model|--,
--|examples_AADL/Common\Base_Types|--,
--|examples_AADL/Common\Math|--,
--|examples_AADL/Common\Gui|--

END

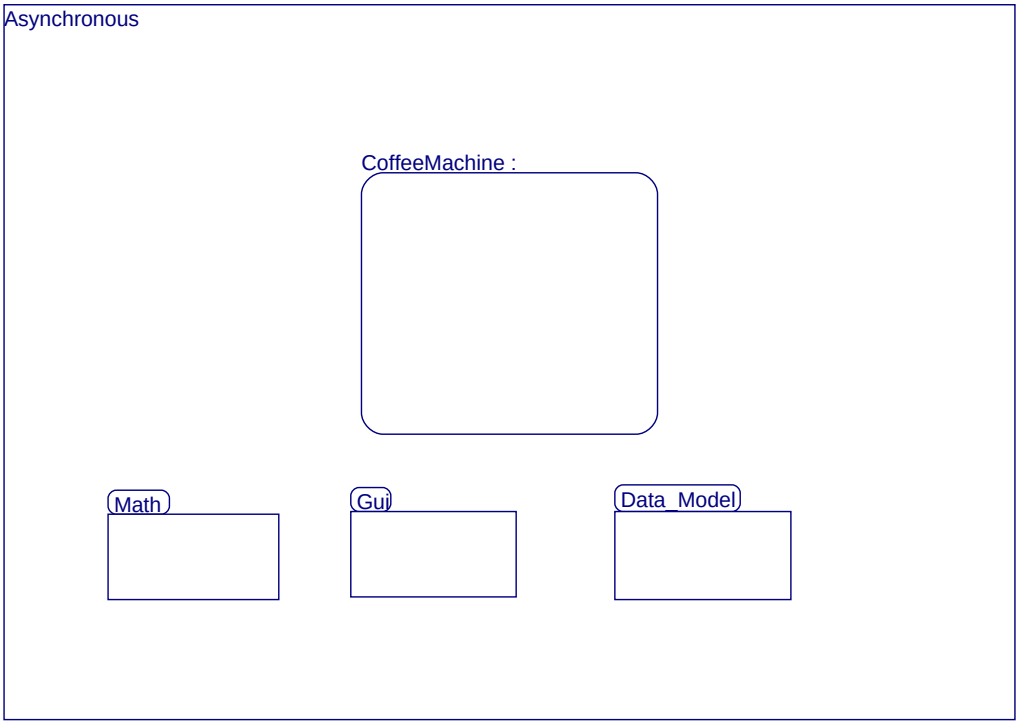
1.1. Project Description

ASYNCHRONOUS CONTROL SYSTEM

This project shows an event driven "coffee machine" automata.
The user can control the coffe machine thanks to its
automatically generated user interface.

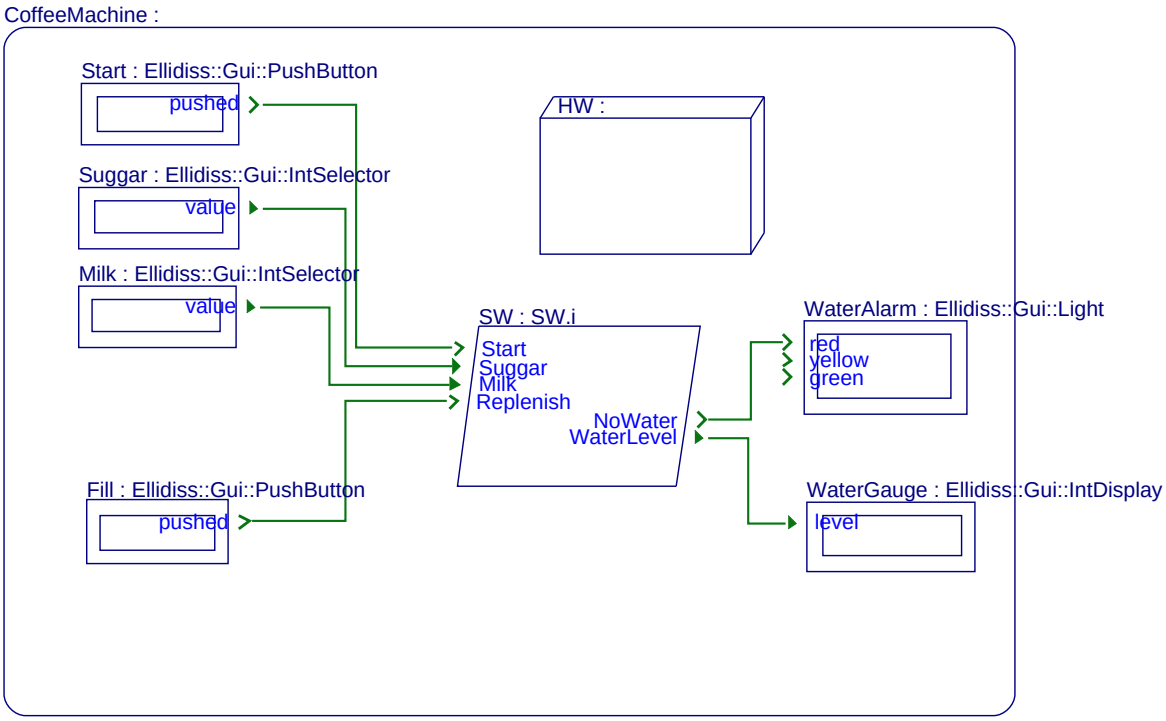
The AADL Behavior Annex is used to implement the behavior
of each thread. It shows examples of use of if..end if and for
statements

1.2. AADL Diagram



2. SYSTEM CoffeeMachine IS

2.1. AADL Diagram



2.2. IMPLEMENTATION

2.2.1. BEHAVIOR

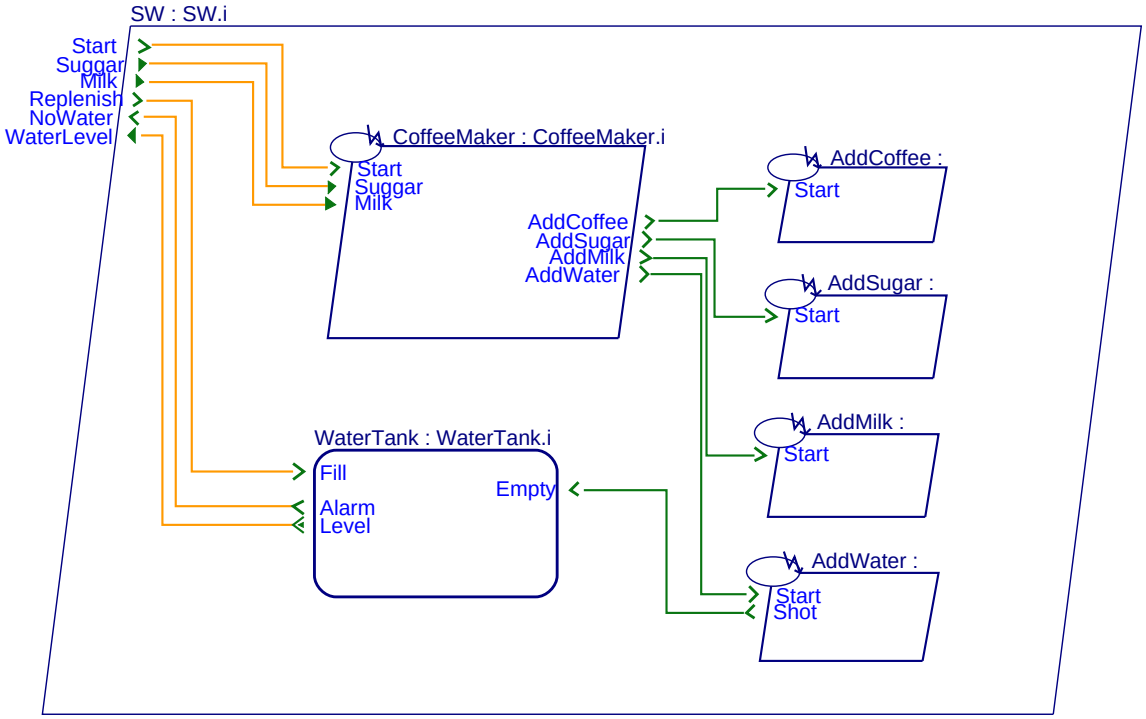
2.2.1.1. TRANSITIONS

3. PROCESS SW IS

3.1. Statement of the Problem (text)

The main SW application is represented by an AADL process that reads and sends data from/to the coffee machine dashbo
(buttons, selectors and displays).

3.2. AADL Diagram



3.3. IMPLEMENTATION

3.3.1. BEHAVIOR

3.3.1.1. TRANSITIONS

4. THREAD CoffeeMaker IS

4.1. Statement of the Problem (text)

The real time software architecture can be split into two parts:

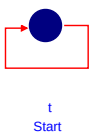
- the coffee maker and its four satellite threads to add the various coffee ingredients.

- the water tank manager thread group

4.2. IMPLEMENTATION

4.2.1. BEHAVIOR

4.2.1.1. State Transition Diagram



4.2.1.2. TRANSITIONS

4.2.1.2.1. t

4.2.1.2.2. Transition Description

add a coffee dose

if selected, add a suggar dose

if selected, add a milk dose

add a water dose

4.2.1.2.3. Transition Actions (aadl)

```
AddCoffee!;
if (Suggar = 1) AddSugar! end if;
if (Milk = 1) AddMilk! end if;
AddWater!
```

5. THREAD AddCoffee IS

5.1. IMPLEMENTATION

5.1.1. BEHAVIOR

5.1.1.1. TRANSITIONS

6. THREAD AddSugar IS

6.1. IMPLEMENTATION

6.1.1. BEHAVIOR

6.1.1.1. TRANSITIONS

7. THREAD AddMilk IS

7.1. IMPLEMENTATION

7.1.1. BEHAVIOR

7.1.1.1. TRANSITIONS

8. THREAD AddWater IS

8.1. IMPLEMENTATION

8.1.1. BEHAVIOR

8.1.1.1. State Transition Diagram

8.1.1.2. TRANSITIONS

8.1.1.2.1. t

8.1.1.2.2. Transition Description

a unit quantity of water is added three times

8.1.1.2.3. Transition Actions (aadl)

```
for (i : int in 0..3)
```

```
{ Shot! }
```

9. THREAD GROUP WaterTank IS

9.1. Statement of the Problem (text)

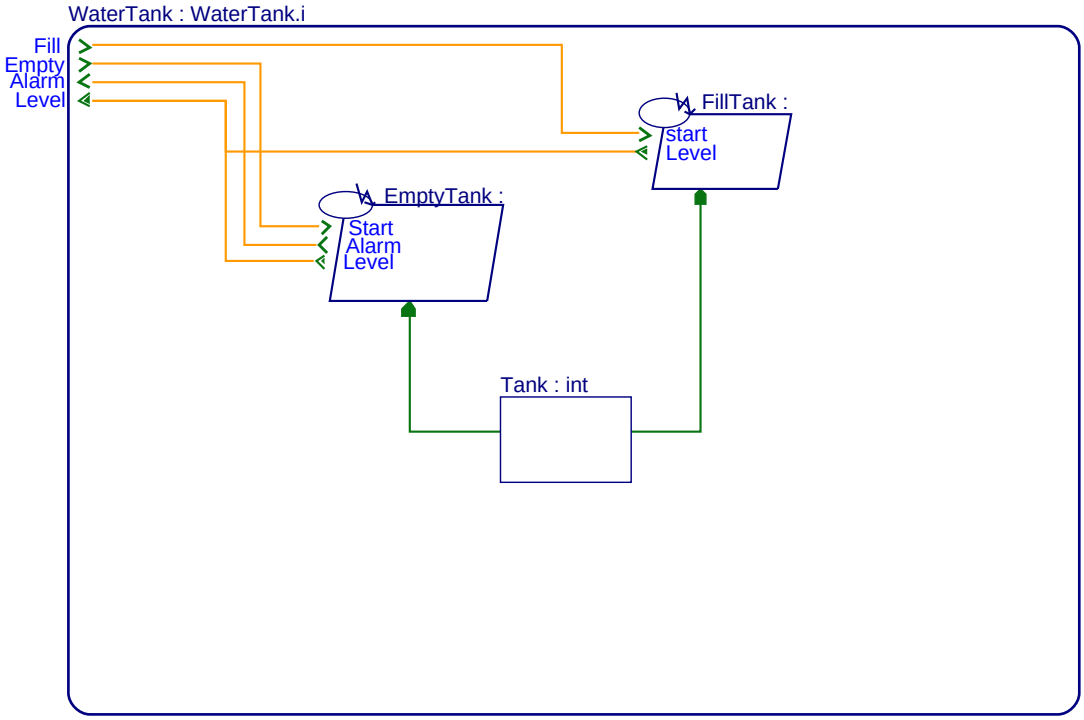
The water tank manager is composed of two threads and a shared data component:

- FillTank: used to resplenish the tank.

- EmptyTank: used by the AddWater thread while making a coffee.

- Tank: represents the current level of water in the tank.

9.2. AADL Diagram



9.3. IMPLEMENTATION

9.3.1. BEHAVIOR

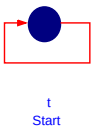
9.3.1.1. TRANSITIONS

10. THREAD EmptyTank IS

10.1. IMPLEMENTATION

10.1.1. BEHAVIOR

10.1.1.1. State Transition Diagram



10.1.1.2. TRANSITIONS

10.1.1.2.1. t

10.1.1.2.2. Transition Description

lock the Tank resource

decrease the water level

if the Tank is empty, raise an alarm

update the water level indicator

release the Tank resource

10.1.1.2.3. Transition Actions (aadl)

```
Tank!<;
  Tank := Tank - 1;
  if (Tank <= 0) Alarm!; Tank := 0 end if;
  Level := Tank;
Tank!>
```

11. DATA Tank IS

11.1. IMPLEMENTATION

11.1.1. BEHAVIOR

11.1.1.1. TRANSITIONS

12. THREAD FillTank IS

12.1. IMPLEMENTATION

12.1.1. BEHAVIOR

12.1.1.1. State Transition Diagram

12.1.1.2. TRANSITIONS

12.1.1.2.1. t

12.1.1.2.2. Transition Description

lock the Tank resource

increase the water level

update the water level indicator

release the Tank resource

12.1.1.2.3. Transition Actions (aadl)

```
Tank !<;
  Tank := 10;
Tank!>;
Level := 10
```