FSM Model for coffee machine

***Coffee Machine*:**

The coffee machine in the example provides the user with a cup of coffee or a cup of hot water (for tea), given that the customer has inserted the required amount of money. The coffee machine can handle coins of the values 5 and 10, where 5 is the price for a cup of tea while a cup of coffee costs 10.

There are some points:

* If a coin with the value of 10 is inserted and the Coffee button is pressed, the customer receives a cup of coffee.
* If a coin with the value of 10 is inserted and the Tea button is pressed, the customer receives a cup of hot water plus change.
* If a coin with the value of 5 is inserted and the Coffee button is pressed, the money is returned.
* If a coin with the value of 5 is inserted and the Tea button is pressed, the customer receives a cup of hot water

***Model*:** A picture containing text, whiteboard

Description automatically generated

Possible interaction sequences:

1. Coin-10; tea; return-5; cup of tea; coin-x; return-coin; coin-5.
2. Coin-5; tea; cup of tea; coin-10; coffee; cup of Coffee.

Because of the rendezvous interactions, at the end of sequence (1), no coin can be entered (the Coffee Machine has no coin transition in its current state (which is *paid 5*). Similarly, no cup of Coffee interaction is possible in the initial state.

***LTSA Notation:***

A picture containing text, whiteboard

Description automatically generatedIn the LTSA notation, this state machine can be defined by the following text:

IDLE = (coin [5] -> PAIDFIVE | coin [10] -> PAIDTEN | coin Else -> return Change -> IDLE),  
PAIDFIVE = (tea -> MAKETEA | coffee -> return Change -> IDLE),  
PAIDTEN = (tea -> return Coin [5] -> MAKETEA | coffee -> MAKECOFFEE),  
MAKETEA = (cup of Tea -> IDLE),  
MAKECOFFEE = (cup of Coffee -> IDLE).

***IOA Model:***

***Diagram, schematic

Description automatically generated***

Timeline

Description automatically generated with low confidence

The states and transitions of this model correspond directly to the states and transition of the LTS model above.

Here a distinction is made between input and output interactions, indicated by, 0 and 1, respectively.

The main difference with LTS modeling is that interactions are not rendezvous. For instance, an input coin-5 would be possible in the state Paid 10, but the fact that the diagram contains no coin-5 transition from this state means that **the hypothesis is made about the behavior of the system environment** that the environment will not produce this input when the Coffee Machine is in the Paid 10 state. If this happens nevertheless, we say that this input is a **non-specified input.**

***FSM Model:***

A piece of paper with writing on it

Description automatically generated with medium confidence

This model is like the IOA model, but in the FSM formalism, a transition contains input and output interactions, therefore this model has less states and transitions. For instance, the tea transition from the Paid 10 state contains the tea input event, plus two output interactions, return 5, and cup of Tea. This transition also includes (implicitly) two states that were shown explicitly in the IOA diagram above.

***Internal Structure:***

In the architectural model (3), the component Controller interacts with the user (as shown in the above behavior models) and with the Hardware component which produces the drinks. The LTS model of the Hardware component is shown in the LTS notation below. Each command given to the Hardware is followed by a corresponding response rendezvous interaction.

HARDWARE = (fill Water -> water OK -> HARDWARE | heat Water -> warm -> HARDWARE | fill Coffee -> PREPARECOFFEE),  
PREPARECOFFEE = (coffee OK -> HARDWARE).

A picture containing diagram

Description automatically generated

Using the LTS modeling approach with rendezvous interactions, the functionality of the Hardware may be used by the Controller component when it is in the Make Coffee state of the LTS behavior model above (we assume that this diagram also models the Controller) - however, in the state Make Coffee, it interacts with the Hardware as shown by the diagram below. A similar diagram defines how tea is made in the Make Tea state of the Controller.

Bar chart

Description automatically generated with medium confidence

The following is a model of the Controller in LTSA notation. The first three lines of the Controller definition are the same as above.

IDLE = (coin[5] -> PAIDFIVE | coin[10] -> PAIDTEN | coin Else -> return Change -> IDLE),  
PAIDFIVE = (tea -> MAKETEA | coffee -> return Change -> IDLE),  
PAIDTEN = (tea -> return Coin[5] -> MAKETEA | coffee -> MAKECOFFEE),  
MAKETEA = (fill Water -> water -> heat Water -> warm -> cup of Tea -> IDLE),  
MAKECOFFEE = (fill water -> water -> fill Coffee -> coffee OK -> heat Water -> warm -> cup of Coffee -> IDLE).