

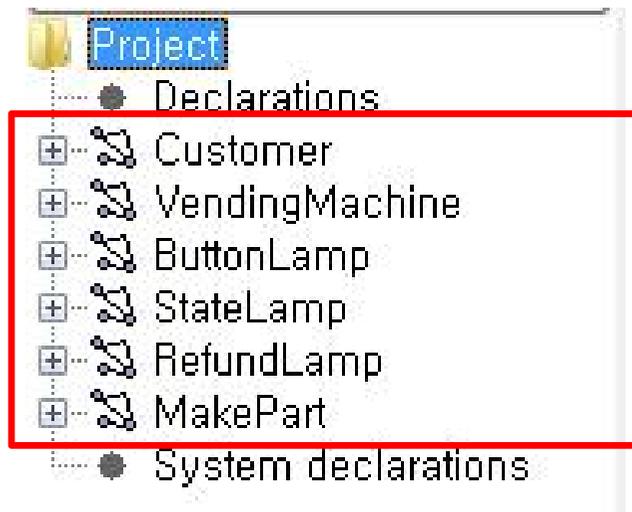
UPPAAL을 이용한 커피자판기 설계

Team 2(이근수, 김준영)

Editor를 이용한 설계(1)

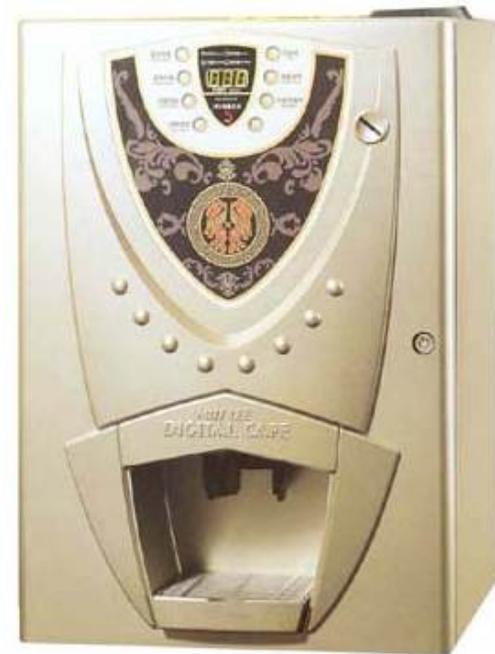
▶ 구성

- ▶ 입력을 발생시키는 “Customer”
- ▶ 입력에 의해 행동을 수행하는 “VendingMachine”
- ▶ 버튼의 Lamp의 On, Off를 확인하는 “XXXLamp”
- ▶ 음료를 만드는 과정을 수행하는 “MakePart”



Editor를 이용한 설계(2)

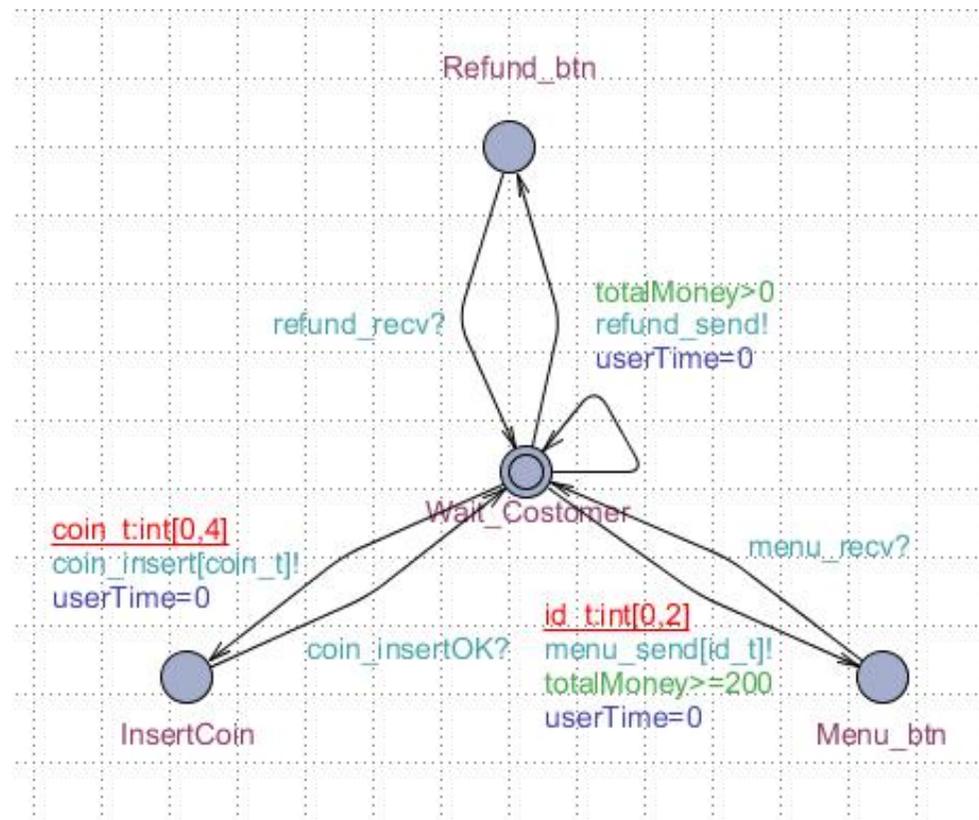
- ▶ 가상의 자판기 특징
 - ▶ “10, 50, 100, 500, 1000원”권 투입 가능
 - ▶ 10,000원을 초과하여 투입 불가
 - ▶ 5초 이상 입력이 없으면 잔돈 반환
 - ▶ 3가지 종류의 음료 제공
 - ▶ 커피(200원)
 - ▶ 홍차(200원)
 - ▶ 녹차(200원)
 - ▶ 반환 버튼 제공
 - ▶ 모든 음료는 버튼을 누르는 시간부터 20초 안에 제공이 됨
 - ▶ 반환 버튼을 누르면 3초 이내에 돈이 반환 되어야 함



Editor를 이용한 설계(3)

▶ Customer의 Location 정의

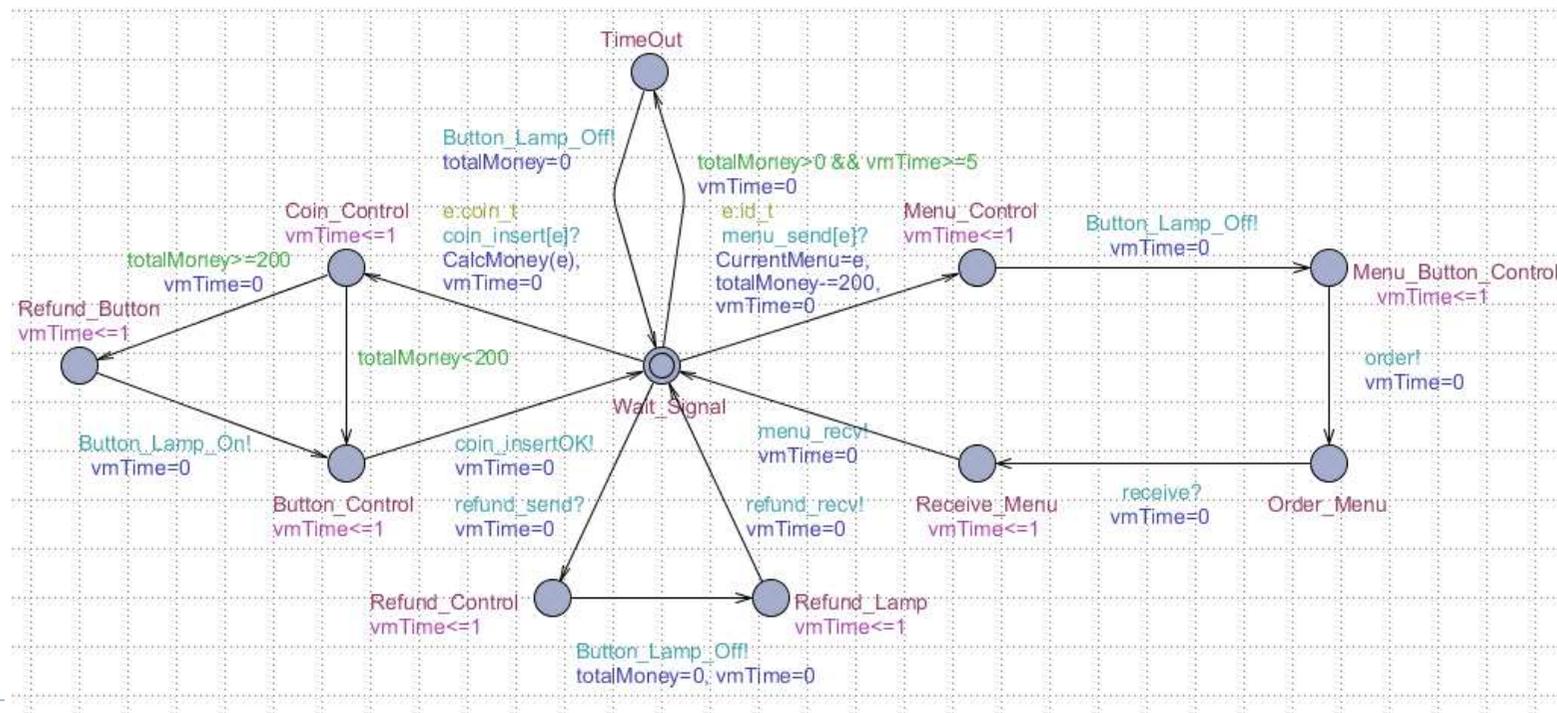
- ▶ 자판기 사용자가 없는 “WaitingCustomer”
- ▶ 메뉴를 선택하는 “Menu_btn”
- ▶ 돈을 투입하는 “InsertCoin”
- ▶ 반환 버튼을 누르는 “Refund_btn”



Editor를 이용한 설계(3)

▶ VendingMachine의 Location 정의

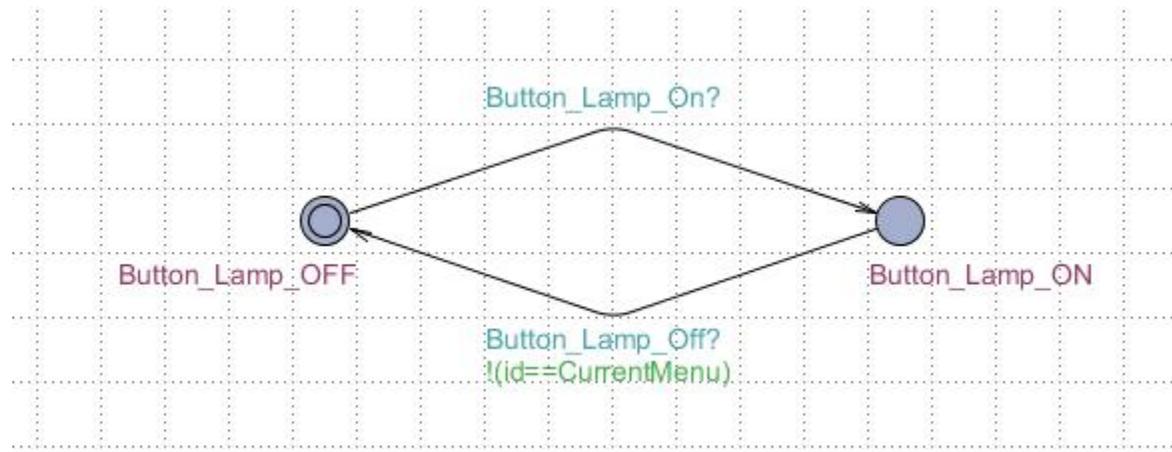
- ▶ Customer에서 입력을 대기하는 “Wait_Signal”
- ▶ 돈이 투입 되었을 경우 “Coin_Control”
- ▶ 음료가 선택 되었을 경우 “Menu_Control”



Editor를 이용한 설계(4)

▶ ButtonLamp

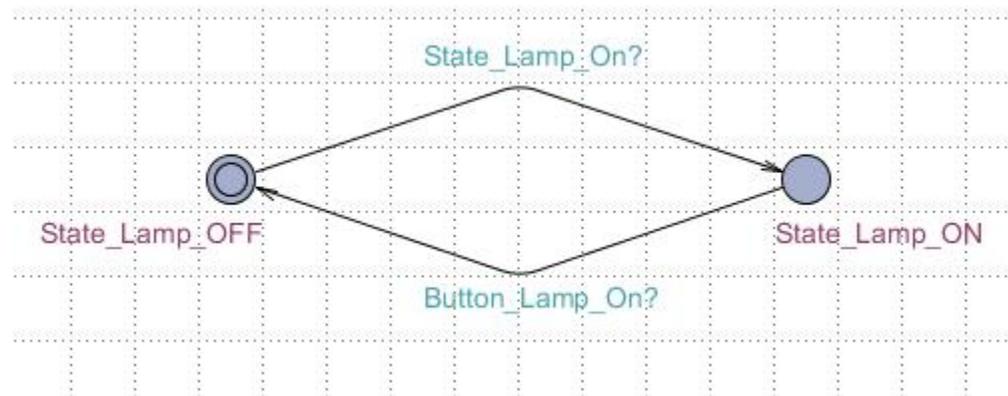
- ▶ Parameter로 id_t 인자를 전달
- ▶ 3가지 음료에 대하여 Lamp를 서로 구분
- ▶ 동전 투입 후 선택을 할 수 있는 음료에 대하여 Lamp ON



Editor를 이용한 설계(5)

- ▶ StateLamp

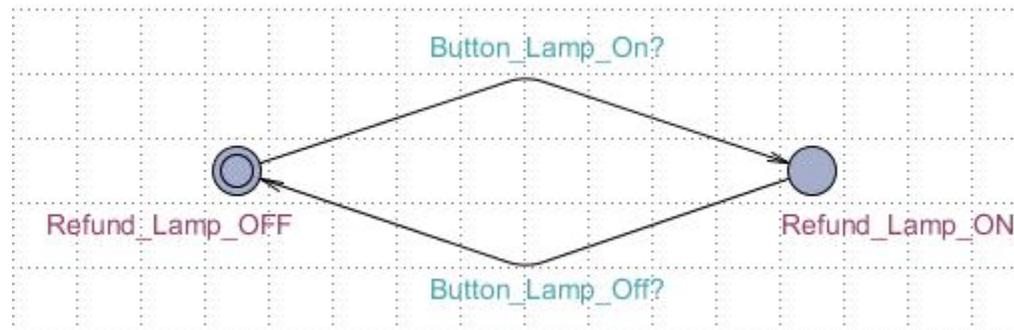
- ▶ 선택된 음료가 나오는 동안 Lamp ON



Editor를 이용한 설계(6)

▶ RefundLamp

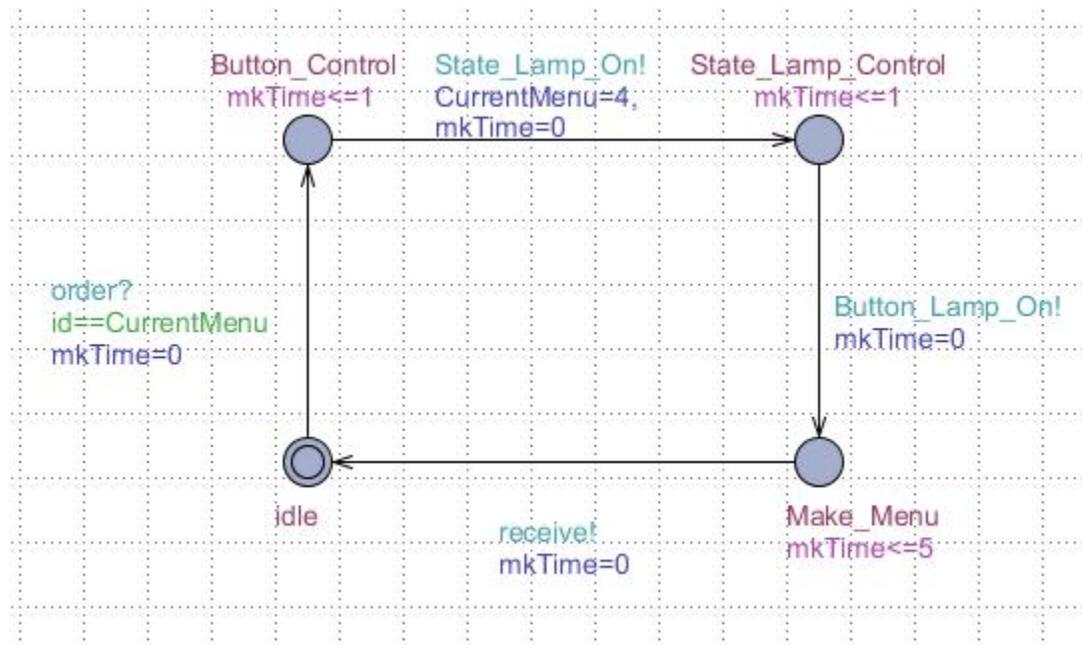
- ▶ 동전이 투입이 되면 Lamp ON
- ▶ 반환 버튼을 누르면 Lamp OFF



Editor를 이용한 설계(7)

▶ MakePart

- ▶ Parameter로 id_t 인자를 전달
- ▶ 3가지 음료에 대하여 만드는 동작을 분류

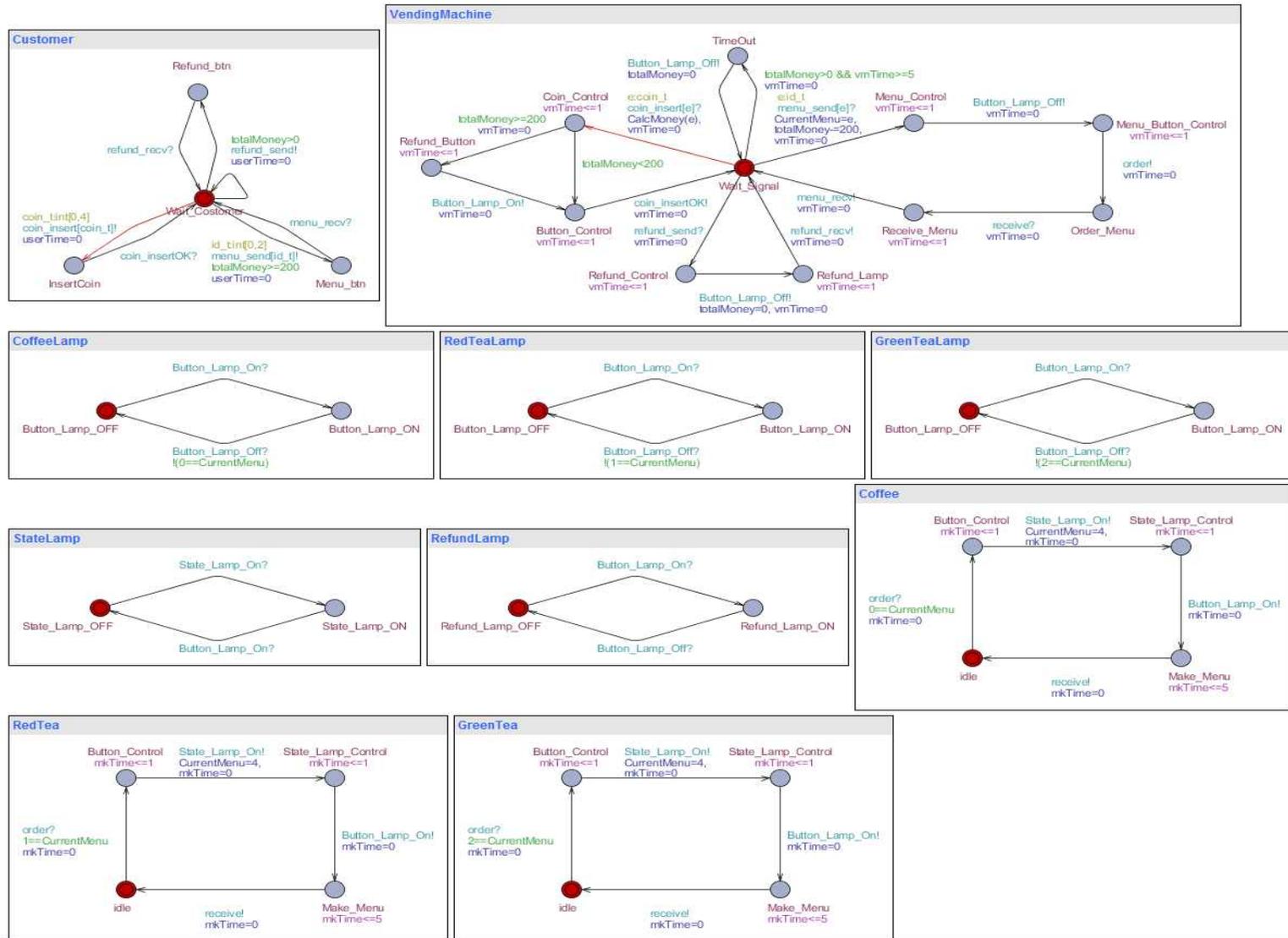


Simulator를 이용한 확인(1)

The image shows a simulator interface with two main panels. The left panel, titled "Enabled Transitions", contains a list box with one entry: "receive: Coffee --> VendingMachine". Below this list are "Next" and "Reset" buttons. The right panel, titled "Simulation Trace", displays a list of simulation events. The events are as follows:

```
(Wait_Costomer, Wait_Signal, Button_Lamp_OFF, Button_Lamp_OFF, Button_Lamp_OFF, State_Lamp_OFF, Refund_Lamp_OFF, idle, idle, idle)
coin_insert[coin_t]: Customer --> VendingMachine
(InsertCoin, Coin_Control, Button_Lamp_OFF, Button_Lamp_OFF, Button_Lamp_OFF, State_Lamp_OFF, Refund_Lamp_OFF, idle, idle, idle)
VendingMachine
(InsertCoin, Button_Control, Button_Lamp_OFF, Button_Lamp_OFF, Button_Lamp_OFF, State_Lamp_OFF, Refund_Lamp_OFF, idle, idle, idle)
coin_insertOK: VendingMachine --> Customer
(Wait_Costomer, Wait_Signal, Button_Lamp_OFF, Button_Lamp_OFF, Button_Lamp_OFF, State_Lamp_OFF, Refund_Lamp_OFF, idle, idle, idle)
coin_insert[coin_t]: Customer --> VendingMachine
(InsertCoin, Coin_Control, Button_Lamp_OFF, Button_Lamp_OFF, Button_Lamp_OFF, State_Lamp_OFF, Refund_Lamp_OFF, idle, idle, idle)
VendingMachine
(InsertCoin, Refund_Button, Button_Lamp_OFF, Button_Lamp_OFF, Button_Lamp_OFF, State_Lamp_OFF, Refund_Lamp_OFF, idle, idle, idle)
Button_Lamp_On: VendingMachine --> CoffeeLamp, RedTeaLamp, GreenTeaLamp, RefundLamp
(InsertCoin, Button_Control, Button_Lamp_ON, Button_Lamp_ON, Button_Lamp_ON, State_Lamp_OFF, Refund_Lamp_ON, idle, idle, idle)
coin_insertOK: VendingMachine --> Customer
(Wait_Costomer, Wait_Signal, Button_Lamp_ON, Button_Lamp_ON, Button_Lamp_ON, State_Lamp_OFF, Refund_Lamp_ON, idle, idle, idle)
menu_send[id_t]: Customer --> VendingMachine
(Menu_btn, Menu_Control, Button_Lamp_ON, Button_Lamp_ON, Button_Lamp_ON, State_Lamp_OFF, Refund_Lamp_ON, idle, idle, idle)
Button_Lamp_Off: VendingMachine --> RedTeaLamp, GreenTeaLamp, RefundLamp
(Menu_btn, Menu_Button_Control, Button_Lamp_ON, Button_Lamp_OFF, Button_Lamp_OFF, State_Lamp_OFF, Refund_Lamp_OFF, idle, idle, idle)
order: VendingMachine --> Coffee
(Menu_btn, Order_Menu, Button_Lamp_ON, Button_Lamp_OFF, Button_Lamp_OFF, State_Lamp_OFF, Refund_Lamp_OFF, Button_Control, idle, State_Lamp_On: Coffee --> StateLamp
(Menu_btn, Order_Menu, Button_Lamp_ON, Button_Lamp_OFF, Button_Lamp_OFF, State_Lamp_ON, Refund_Lamp_OFF, State_Lamp_Control, Button_Lamp_On: Coffee --> RedTeaLamp, GreenTeaLamp, StateLamp, RefundLamp
(Menu_btn, Order_Menu, Button_Lamp_ON, Button_Lamp_ON, Button_Lamp_ON, State_Lamp_OFF, Refund_Lamp_ON, Make_Menu, idle, idle)
```

Simulator를 이용한 확인(2)



Verification 명세

- 1) 모든 state에서 deadlock이 존재하지 않는다.
→ $A \square$ not deadline
- 2) CVM에서 유지하는 돈의 범위는 항상 0 ~ 9999 만족한다.
→ $A \square$ not (totalMoney < 0 and totalMoney > 10000)
- 3) InsertCoin과 Menu_btn은 항상 동시에 할 수 가 없다.
→ $A \square$ not (Customer.InsertCoin and Customer.Menu_btn)
- 4) 커피, 홍차, 녹차를 만드는 작업은 항상 동시에 이루어 질 수 없다.
→ $A \square$ not (Coffee.Make_Menu and RedTea.Make_Menu and GreenTea.Make_Menu)
- 5) 사용자가 Refund_btn을 눌렀을 경우에 환불이 이루어져야 한다.
→ Customer.Refund_btn - - > VendingMachine.Refund_Lamp



Verification 명세(2)

- 6) 사용자가 Menu_btn을 눌렀을 경우 음료가 만들어져야 한다.
→ $E \leftrightarrow \text{Customer.Menu_btn and (Coffee.Make_Menu or RedTea.Make_Menu or GreenTea.Make_Menu)}$
- 7) CVM은 입력이 없을 경우 Timeout 상태에 진입 할 수 있어야 한다.
→ $E \leftrightarrow \text{VendingMachine.TimeOut}$
- 8) 사용자가 메뉴 버튼을 누르면 20초 안에 음료가 제공이 되어야 한다.
→ $A \square \text{Customer.Menu_btn and VendingMachine.Receive_Menu imply Customer.userTime} \leq 20$
- 9) 사용자가 동전을 넣었을 경우 다음 입력 대기까지 걸리는 시간은 5초 이내여야 한다.
→ $A \square \text{Customer.InsertCoin and VendingMachine.Button_Menu imply Customer.userTime} \leq 5$
- 10) 사용자가 반환 버튼을 눌렀을 때 3초 안에 금액을 반환 해야 한다.
→ $A \square \text{Customer.Refund_btn and VendingMachine.Refund_Lamp imply Customer.userTime} \leq 3$



Verification 결과

Status

Established direct connection to local server,
(Academic) UPPAAL version 4.0.10 (rev. 4417), September 2009 -- server,
Disconnected,
Established direct connection to local server,
(Academic) UPPAAL version 4.0.10 (rev. 4417), September 2009 -- server,

- 1) A[] not deadlock
Property is satisfied,
- 2) A[] not (totalMoney < 0 and totalMoney > 10000)
Property is satisfied,
- 3) A[] not (Customer.InsertCoin and Customer.Menu_btn)
Property is satisfied,
- 4) A[] not (Coffee.Make_Menu and RedTea.Make_Menu and GreenTea.Make_Menu)
Property is satisfied,
- 5) Customer.Refund_btn --> VendingMachine.Refund_Lamp
Property is satisfied,
- 6) E<> Customer.Menu_btn and (GreenTea.Make_Menu or RedTea.Make_Menu or GreenTea.Make_Menu)
Property is satisfied,
- 7) E<> VendingMachine.TimeOut
Property is satisfied,
- 8) A[] VendingMachine.Receive_Menu imply Customer.userTime <=20
Property is satisfied,
- 9) A[] VendingMachine.Button_Control imply Customer.userTime <=5
Property is satisfied,
- 10) A[] VendingMachine.Refund_Lamp imply Customer.userTime <=3
Property is satisfied,