# coffee vending machine

-NuSMV-

202172247 허윤아

DSLAB

# MODULE user

```
MODULE user
VAR
    choose_espresso : boolean;
    choose_black : boolean;
    choose_milk : boolean;
    choose_latte : boolean;
    choose_mocha : boolean;
    coin_in : {0, 100, 500, 1000};
    refund_in : boolean;
ASSIGN
    init(choose_espresso) := FALSE;
    init(choose_black) := FALSE;
    init(choose_milk) := FALSE;
    init(choose_latte) := FALSE;
    init(choose_mocha) := FALSE;
    init(refund_in) := FALSE;
    next(refund_in) :=
        case
            choose_espresso | choose_black | choose_milk | choose_latte | choose_mocha : FALSE;
            TRUE : {TRUE, FALSE};
        esac;
    init(coin_in) := 0;
```

# MODULE coin

```
MODULE coin
VAR
    current_coin : 0..5000;
    user : user;
ASSIGN
    init(current_coin) := 0;
    next(current_coin) :=
        case
            user.coin_in + current_coin <= 5000 & !user.refund_in & !user.choose_espresso & !user.choose_black & !user.choose_milk & !user.choose_latte & !user.choose_mocha :
            current_coin + user.coin_in;
            user.choose_espresso & !user.choose_black & !user.choose_milk & !user.choose_latte & !user.choose_mocha & current_coin >= 100 & !user.refund_in : current_coin - 100;
            !user.choose_espresso & user.choose_black & !user.choose_milk & !user.choose_latte & !user.choose_mocha & current_coin >= 200 & !user.refund_in : current_coin - 200;
            !user.choose_espresso & !user.choose_black & user.choose_milk & !user.choose_latte & !user.choose_mocha & current_coin >= 300 & !user.refund_in : current_coin - 300;
            !user.choose_espresso & !user.choose_black & !user.choose_milk & user.choose_latte & !user.choose_mocha & current_coin >= 400 & !user.refund_in : current_coin - 400;
            !user.choose_espresso & !user.choose_black & !user.choose_milk & !user.choose_latte & user.choose_mocha & current_coin >= 500 & !user.refund_in : current_coin - 500;
            user.refund_in : 0;
            TRUE : current_coin;
        esac;
```

# MODULE stock

```
MODULE stock
VAR
    current_espresso : 0..500;
    current_water : 0..1000;
    current_milk : 0..1000;
    current_choco : 0..500;
    user : user;
    manager : manager;
ASSIGN
    init(current_espresso) := 500;
    next(current_espresso) :=
        case
            user.choose_espresso & !user.choose_black & !user.choose_milk & !user.choose_latte & !user.choose_mocha & current_espresso >= 20 : current_espresso - 20;
            !user.choose_espresso & user.choose_black & !user.choose_milk & !user.choose_latte & !user.choose_mocha & current_espresso >= 20 : current_espresso - 20;
            !user.choose_espresso & !user.choose_black & !user.choose_milk & user.choose_latte & !user.choose_mocha & current_espresso >= 20 : current_espresso - 20;
            !user.choose_espresso & !user.choose_black & !user.choose_milk & !user.choose_latte & user.choose_mocha & current_espresso >= 20 : current_espresso - 20;
            manager.fill_espresso : 500;
            TRUE : current_espresso;
        esac;
    init(current_water) := 1000;
    next(current_water) :=
        case
            !user.choose_espresso & user.choose_black & !user.choose_milk & !user.choose_latte & !user.choose_mocha & current_water >= 80 : current_water - 80;
            manager.fill_water : 1000;
            TRUE : current_water;
        esac;
    init(current_milk) := 1000;
    next(current_milk) :=
        case
            !user.choose_espresso & !user.choose_black & !user.choose_milk & user.choose_latte & !user.choose_mocha & current_milk >= 80 : current_milk - 80;
            !user.choose_espresso & !user.choose_black & user.choose_milk & !user.choose_latte & !user.choose_mocha & current_milk >= 100 : current_milk - 100;
            !user.choose_espresso & !user.choose_black & !user.choose_milk & !user.choose_latte & user.choose_mocha & current_milk >= 70 : current_milk - 70;
            manager.fill_milk : 1000;
            TRUE : current_milk;
        esac;
    init(current_choco) := 500;
    next(current_choco) :=
        case
            !user.choose_espresso & !user.choose_black & !user.choose_milk & !user.choose_latte & user.choose_mocha & current_choco >= 10 : current_choco - 10;
            manager.fill_choco : 500;
            TRUE : current_choco;
        esac;
```

# MODULE manager

```
MODULE manager
VAR
    fill_espresso : boolean;
    fill_water : boolean;
    fill_milk : boolean;
    fill_choco : boolean;
    power : boolean;
ASSIGN
    init(fill_espresso) := FALSE;
    init(fill_water) := FALSE;
    init(fill_milk) := FALSE;
    init(fill_choco) := FALSE;
    init(power) := FALSE;
    next(power) :=
        case
            fill_espresso : FALSE;
            fill_water : FALSE;
            fill_milk : FALSE;
            fill_choco : FALSE;
            TRUE : TRUE;
        esac;
```

# MODULE alarm

```
MODULE alarm
VAR
    no_espresso : boolean;
    no_black : boolean;
    no_milk : boolean;
    no_latte : boolean;
    no_mocha : boolean;
    stock : stock;
ASSIGN
    init(no_espresso) := FALSE;
    next(no_espresso) :=
        case
            stock.current_espresso < 20 : TRUE;
            TRUE : FALSE;
        esac;
    init(no_black) := FALSE;
    next(no_black) :=
        case
            stock.current_espresso < 20 | stock.current_water < 80 : TRUE;
            TRUE : FALSE;
        esac;
    init(no_milk) := FALSE;
    next(no_milk) :=
        case
            stock.current_milk < 100 : TRUE;
            TRUE : FALSE;
        esac;
    init(no_latte) := FALSE;
    next(no_latte) :=
        case
            stock.current_espresso < 20 |stock. current_milk < 80 : TRUE;
            TRUE : FALSE;
        esac;
    init(no_mocha) := FALSE;
    next(no_mocha) :=
        case
            stock.current_espresso < 20 | stock.current_choco < 10 | stock.current_milk < 70 : TRUE;
            TRUE : FALSE;
        esac;
```

# MODULE main

```
MODULE main
VAR
    state : {off, idle, filling, espresso_out, black_out, milk_out, latte_out, mocha_out};
    user : user;
    coin : coin;
    stock : stock;
    manager : manager;
    alarm : alarm;
ASSIGN
    init(state) := idle;
    next(state) :=
        case
            manager.power : idle;
            state = idle & !manager.power : off;
            state = idle & user.choose_espresso & !user.choose_black & !user.choose_milk & !user.choose_latte & !user.choose_mocha & coin.current_coin >= 100 & !no_espresso :
            espresso_out;
            state = idle & !user.choose_espresso & user.choose_black & !user.choose_milk & !user.choose_latte & !user.choose_mocha & coin.current_coin >= 200 & !no_black :
            black_out;
            state = idle & !user.choose_espresso & !user.choose_black & user.choose_milk & !user.choose_latte & !user.choose_mocha & coin.current_coin >= 300 & !no_milk :
            milk_out;
            state = idle & !user.choose_espresso & !user.choose_black & !user.choose_milk & user.choose_latte & !user.choose_mocha & coin.current_coin >= 400 & !no_latte :
            latte_out;
            state = idle & !user.choose_espresso & !user.choose_black & !user.choose_milk & !user.choose_latte & user.choose_mocha & coin.current_coin >= 500 & !no_mocha :
            mocha_out;
            state = idle & manager.fill_espresso | manager.fill_water | manager.fill_milk | manager.fill_choco : filling;
            TRUE : idle;
        esac;
```

# CTL Properties

| Property | Description | Expected Output | Result | Module |
|---|---|---|---|---|
| SPEC AG(user.coin_in = 100 & current_coin = 100 & !user.refund_in & !user.choose_espresso & !user.choose_black & !user.choose_milk & !user.choose_latte & !user.choose_mocha -> AX(current_coin = 200)) | 돈을 집어넣으면 돈이 추가된다 | TRUE | TRUE | coin |
| SPEC AG(user.coin_in = 0 & current_coin = 1000 & !user.refund_in & !user.choose_espresso & !user.choose_black & !user.choose_milk & !user.choose_latte & !user.choose_mocha -> AX(current_coin = 1000)) | 돈을 넣지 않으면 자판기에 남은 돈이 추가되지 않는다 | TRUE | TRUE | |
| SPEC AG(user.coin_in = 500 & current_coin = 5000 & !user.refund_in & !user.choose_espresso & !user.choose_black & !user.choose_milk & !user.choose_latte & !user.choose_mocha -> AX(current_coin = 5000)) | 어떠한 버튼도  누르지 않는 상태에서 돈을 집어넣었을 때 5000원 이상이 되면 돈을 받지 않는다 | TRUE | TRUE | |
| SPEC AG(current_coin = 1000 & user.refund_in & !user.choose_espresso & !user.choose_black & !user.choose_milk & !user.choose_latte & !user.choose_mocha -> AX(current_coin = 0)) | 환불 버튼을 누르면 돈이 환불된다 | TRUE | TRUE | |
| SPEC AG(user.coin_in = 0 & current_coin = 800 & !user.refund_in & !user.choose_espresso & !user.choose_black & !user.choose_milk & !user.choose_latte & user.choose_mocha -> AX(current_coin = 300)) | 돈을 넣은 상태에서 음료를 고르면 음료의 가격만큼 남은 돈이 차감된다 | TRUE | TRUE | |
| SPEC AG EX TRUE | Deadlock freeness | TRUE | TRUE | main |
| SPEC AG(manager.power -> AX(state = idle)) | 전원을 켜면 전원이 켜지고 idle 상태에 진입한다 | TRUE | TRUE | |
| SPEC AG(!manager.power -> state != espresso_out & state != black_out & state != milk_out & state != latte_out & state != mocha_out) | 전원이 꺼져있으면 음료 버튼을 눌러도 음료가 제조되지 않는다 | TRUE | TRUE | |
| SPEC AG(user.choose_espresso & coin.current_coin < 100 & state = idle & manager.power -> AX(state = idle)) | 전원이 켜져있는 상태에서 유저가 음료 버튼을 누르더라도 돈이 충분하지 않으면 음료는 나오지 않는다 | TRUE | TRUE | |
| SPEC AG(user.choose_mocha & user.coin_in = 1000 & (stock.current_choco < 10 \| stock.current_espresso < 20 \| stock.current_milk < 70) & state = idle & manager.power -> AX(state = idle)) | 전원이 켜져있는 상태에서 재고가 충분하지 않으면 음료는 나오지 않는다 | TRUE | TRUE | |

# CTL Properties

| Property | Description | Expected Output | Result | Module |
|---|---|---|---|---|
| SPEC AG(user.choose_espresso & user.choose_latte & !user.refund_in & state = idle & manager.power -> AX(state = idle)) | 전원이 켜져있는 상태에서 유저가 두 개 이상의 음료 버튼을 동시에 누르면 음료가 나오지 않는다 | TRUE | TRUE | main |
| SPEC AG(user.coin_in = 0 & user.choose_black & !user.refund_in & manager.power -> AX(state = idle)) | 전원이 켜져있는 상태에서 돈을 넣지 않으면 음료가 나오지 않는다 | TRUE | TRUE | |
| SPEC AG(user.coin_in = 1000 & !user.refund_in & !user.choose_espresso & !user.choose_black & !user.choose_milk & user.choose_latte & !user.choose_mocha & stock.current_espresso = 100 & stock.current_milk = 100 & state = idle & manager.power -> AX(state = latte_out & stock.current_espresso = 80 & stock.current_milk = 20 & coin.current_coin = 600)) | 전원이 켜져있고 돈이 충분한 상태에서 음료를 선택하면 언젠가는 돈과 재료가 차감되고 음료가 나온다 | TRUE | FALSE | |
| SPEC AG(stock.current_espresso < 20 -> AX(no_espresso & no_black & no_latte & no_mocha)) | 에스프레소가 20보다 적은 경우 우유를 제외한 나머지 음료를 제조할 수 없다 | TRUE | TRUE | alarm |
| SPEC AG(stock.current_choco < 20 -> AX(no_mocha)) | 초코시럽이 20보다 적은 경우 다음 상태에서 모카가 나올 수 없다 | FALSE | FALSE | |
| SPEC EF(choose_black & refund_in) | 유저는 블랙커피 추출 버튼과 환불 버튼을 동시에 입력할 수 있다 | FALSE | TRUE | user |
| SPEC AG(user.choose_milk & user.choose_latte & current_milk = 100 & !manager.fill_milk -> AX(current_milk = 100)) | 사용자가 두 개 이상의 버튼을 동시에 누르면 음료가 제조되지 않는다(재고가 줄어들지 않는다) | TRUE | TRUE | stock |