

고급소프트웨어공학

202271094 최미영

MODULE coin

```
MODULE coin(vm_state, available_coin, current_button)
VAR
    state : {insert_coin, refund_coin, change_coin, ready_menu};
    current_coin : 0..5000;

ASSIGN

    init(state) := insert_coin;
    next(state) := case
        vm_state = power_off : refund_coin;

        vm_state = power_on & state = insert_coin & available_coin < 1000 : insert_coin;
        vm_state = power_on & state = insert_coin & available_coin >= 1000 & current_button = refund : refund_coin;
        vm_state = power_on & state = insert_coin & available_coin >= 1000 & (current_button = coffee | current_button = ice_coffee
        | current_button = latte) : ready_menu;

        vm_state = power_on & state = ready_menu & available_coin < 1000 : change_coin;
        vm_state = power_on & state = ready_menu & current_button = refund : refund_coin;

        TRUE : state;

    esac;

    init(current_coin) := 0;
    next(current_coin) := case
        vm_state = power_on & state = ready_menu & (current_button = coffee | current_button = ice_coffee | current_button = latte)
        : current_coin - 1000;
        TRUE : 0;
    esac;
```

MODULE material

```
MODULE material(vm_state, current_button)
VAR
    state : {insert_material, fill_coffee, fill_ice_coffee, fill_latte, ready_material};

    current_coffee : 0..100;
    current_ice_coffee : 0..100;
    current_latte : 0..100;

ASSIGN
    init(state) := insert_material;
    next(state) := case
        vm_state = power_on & state = insert_material & current_coffee = 0 : fill_coffee;
        vm_state = power_on & state = insert_material & current_ice_coffee = 0 : fill_ice_coffee;
        vm_state = power_on & state = insert_material & current_latte = 0 : fill_latte;

        vm_state = power_on & state = insert_material & current_coffee > 0 & current_ice_coffee > 0 & current_latte > 0 :
            ready_material;

        TRUE : state;
    esac;

    init(current_coffee) := 0;
    next(current_coffee) := case
        vm_state = power_on & state = ready_material & current_button = coffee : current_coffee - 1;
        TRUE : 0;
    esac;

    init(current_ice_coffee) := 0;
    next(current_ice_coffee) := case
        vm_state = power_on & state = ready_material & current_button = ice_coffee : current_ice_coffee - 1;
        TRUE : 0;
    esac;

    init(current_latte) := 0;
    next(current_latte) := case
        vm_state = power_on & state = ready_material & current_button = latte : current_latte - 1;
        TRUE : 0;
    esac;
```

MODULE main

```
MODULE main
```

```
VAR
```

```
    state: {power_off, power_on, ready_product, get_product};
```

```
    coin : coin;
```

```
    material : material;
```

```
    available_coin : {0, 100, 500, 1000, 5000};
```

```
    current_button : {coffee, ice_coffee, latte, refund};
```

```
    check_coin : process coin(state, available_coin, current_button);
```

```
    check_material : process material(state, current_button);
```

```
ASSIGN
```

```
    init(state) := power_off;
```

```
    next(state) := case
```

```
        state = power_off : power_on;
```

```
        state = power_on & coin.current_coin > 0 & (material.current_coffee > 0 | material.current_ice_coffee > 0 |  
        material.current_latte > 0) : ready_product;
```

```
        state = power_on & state = ready_product : get_product;
```

```
        TRUE : state;
```

```
    esac;
```

CTL Properties

Property Spec	Description
SPEC AG EX TRUE	전원이 꺼져 있을때 - Deadlock
SPEC AG (vm_state = power_on & state = insert_coin & available_coin < 1000) -> AX(state = Insert_coin)	전원이 켜져 있을 때 / 금액이 1000원 미만일 때 -> Insert_coin
SPEC AG (vm_state = power_on & state = insert_coin & available_coin >= 1000 & current_button = refund) -> AX(state = refund_coin)	전원이 켜져 있을 때 / 금액이 1000원 이상일 때 / 환불 버튼 누를 때 -> refund_coin
SPEC AG (vm_state = power_on & state = insert_coin & available_coin >= 1000 & (current_button = coffee current_button = ice_coffee current_button = latte)) -> AX(state=ready_menu)	전원이 켜져 있을 때 / 금액이 1000원 이상일 때 / 메뉴 선택 가능 -> ready_menu
SPEC AG (vm_state = power_on & state = ready_menu & available_coin <= 1000) -> AX(state = change_coin)	전원이 켜져 있을 때 / 금액이 1000원 이하일 때 / 메뉴 선택 가능 -> change_coin
SPEC AG(vm_state = power_on & state = ready_menu & current_button = refund) ->AX(state = refund_coin)	전원이 켜져 있을 때 / 메뉴 선택 가능 / 환불버튼 누를 때 -> refund_coin

CTL Properties

Property Spec	Description
SPEC AG (vm_state = power_on & state = insert_material & current_coffee = 0) -> AX(state = fill_coffee)	전원이 켜져 있을 때 / 현재 커피 0 일때 - fill_coffee
SPEC AG (vm_state = power_on & state = insert_material & current_ice_coffee = 0) -> AX(state = fill_ice_coffee)	전원이 켜져 있을 때 / 현재 아이스 커피가 0 일때 - fill_ice_coffee
SPEC AG (vm_state = power_on & state = insert_material & current_latte = 0) -> AX(state = fill_latte)	전원이 켜져 있을 때 / 현재 라떼가 0 일때 - fill_latte

CTL Properties

Property Spec	Description
SPEC AG (vm_state = power_on & state = insert_material & current_coffee = 0) -> AX(state = fill_coffee)	전원이 켜져 있을 때 / 현재 커피 0 일때 - fill_coffee
SPEC AG (vm_state = power_on & state = insert_material & current_ice_coffee = 0) -> AX(state = fill_ice_coffee)	전원이 켜져 있을 때 / 현재 아이스 커피가 0 일때 - fill_ice_coffee
SPEC AG (vm_state = power_on & state = insert_material & current_latte = 0) -> AX(state = fill_latte)	전원이 켜져 있을 때 / 현재 라떼가 0 일때 - fill_latte
SPEC AG(state = power_on & coin.current_coin > 0 & (material.current_coffee > 0 material.current_ice_coffee > 0 material.current_latte > 0)) ->AX(state = ready_product)	전원이 켜져 있을 때 / 돈이 있고 / 재고가 있을 때 - ready_product
SPEC AG(state = power_on & state = ready_product) -> AX(state = get_product)	전원이 켜져 있을 때 / 돈이 있고 / 재고가 있을 때 / 상품 준비가 되었을 때 - get_product

Future progress – 상품입출고 시스템

제목 : 상품입출고 시스템

상품 등록 – 등록 날짜, 상품명, 상품수량, 상품정보, 작성자

상품 출고 – 출고일, 출고담당자

거래처 등록 – 등록 날짜, 거래처명, 거래시작일, 거래종료일, 거래처 담당자

거래처 상품 – 상품명

담당자 등록 – 담당자 명, 입사일, 퇴사일