

고급 소프트웨어 공학

UPPAAL

음료자판기 **3**차 발표



200874157 이무열
200973352 장혜천
200974157 정성문

ANALYSIS

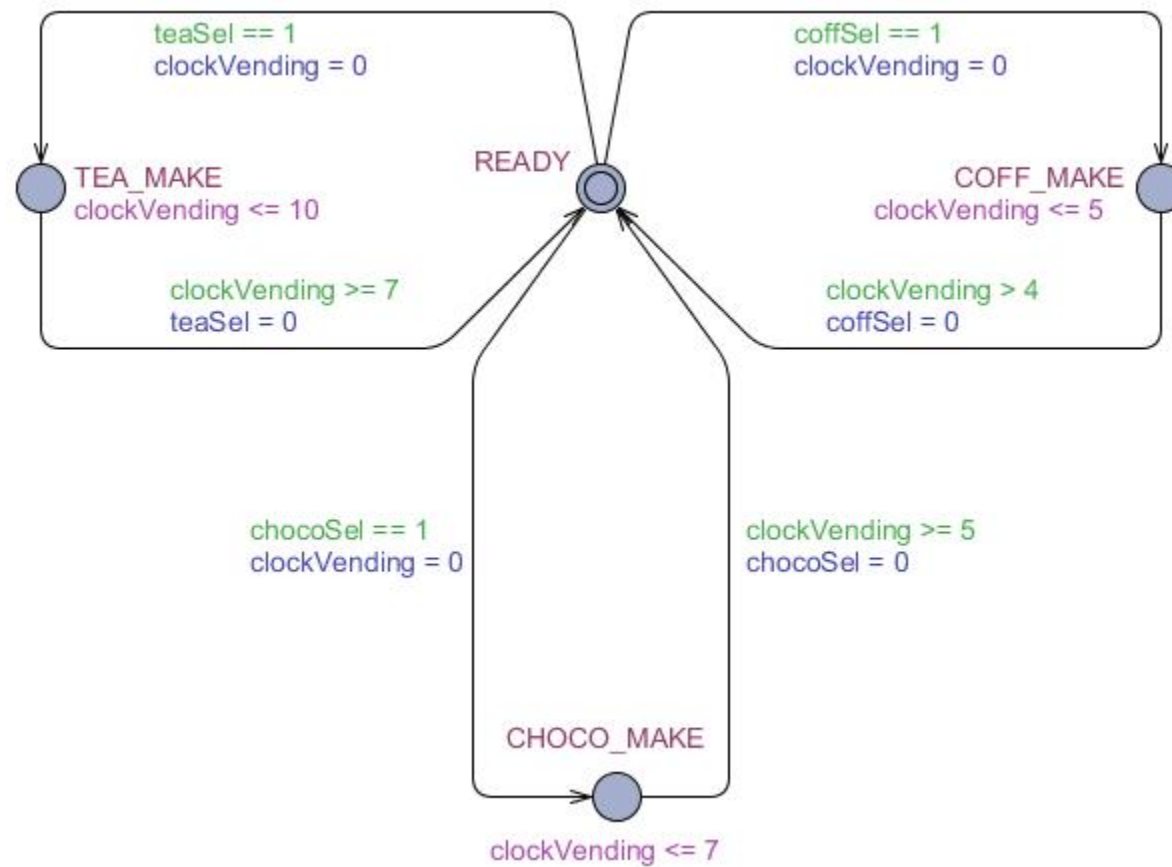
- 조작 버튼
 - 커피 버튼
 - 핫초코 버튼
 - 국산차 버튼
 - 반환 버튼
- 표시
 - 커피 선택 가능 & 선택 중 표시등
 - 핫초코 선택 가능 & 선택 중 표시등
 - 음료 제조 표시등



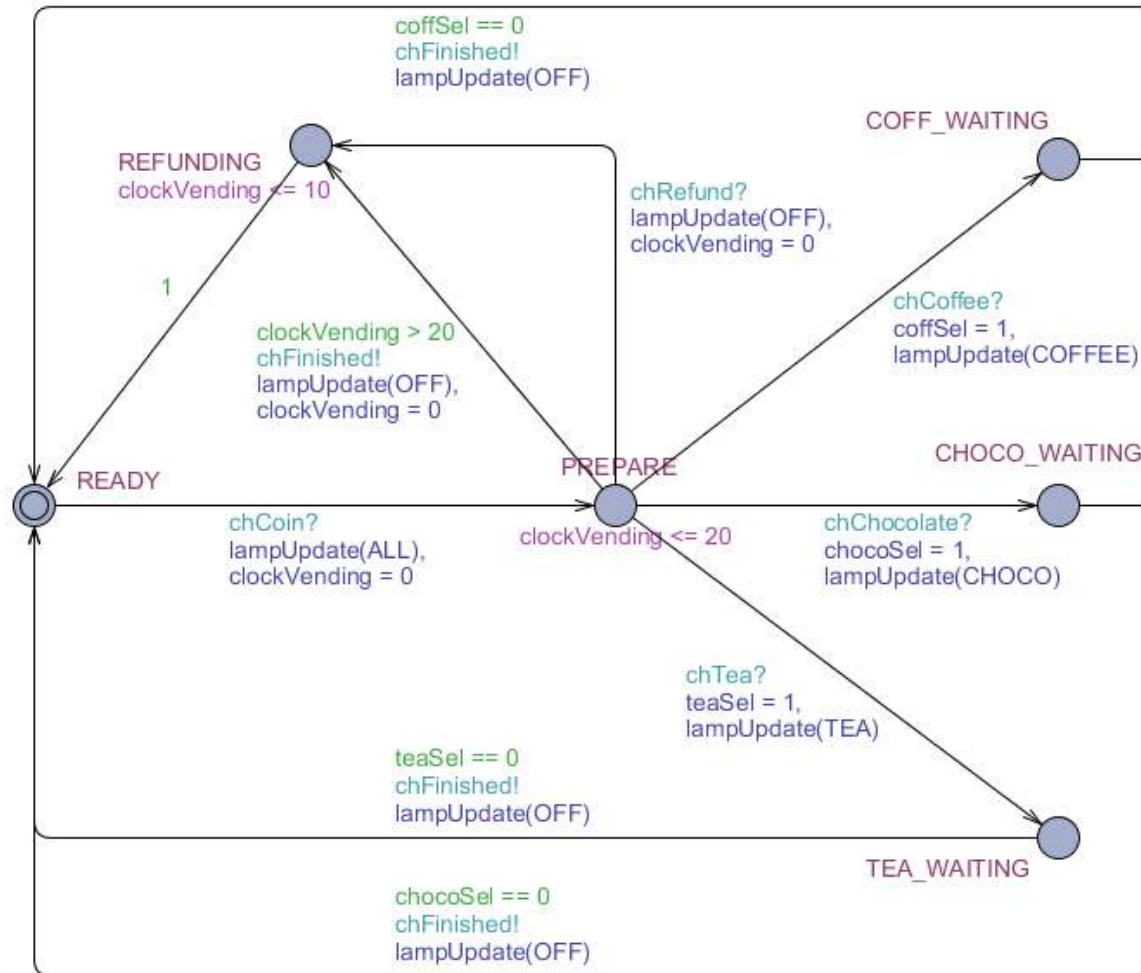
TEMPLATE

- 역할별 상태 부분
 - 자판기 음료 제작 기능 (VENDING PART)
 - 사용자 행동(USER ACTION PART)
- 제어 담당 부분
 - 모든 데이터 처리 및 제어(CONTROL PART)
- 표시 부분
 - 상태 확인 기능 (LAMPs PART)

VENDING

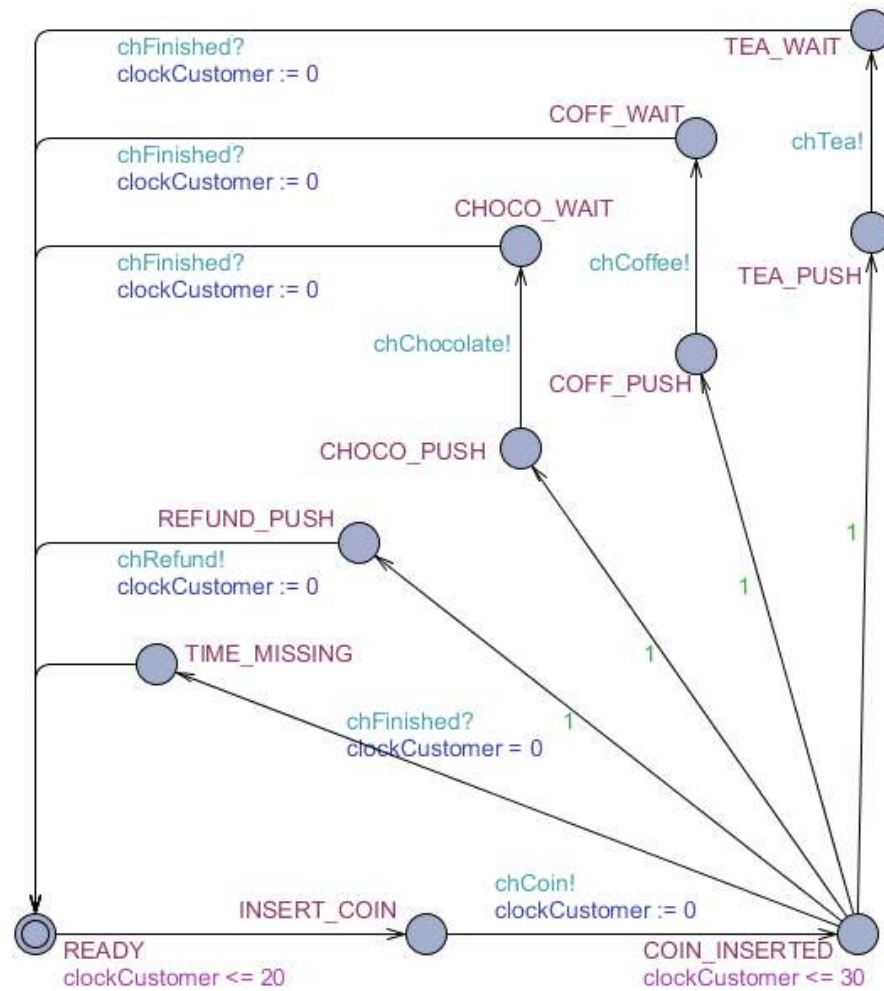


CONTROL

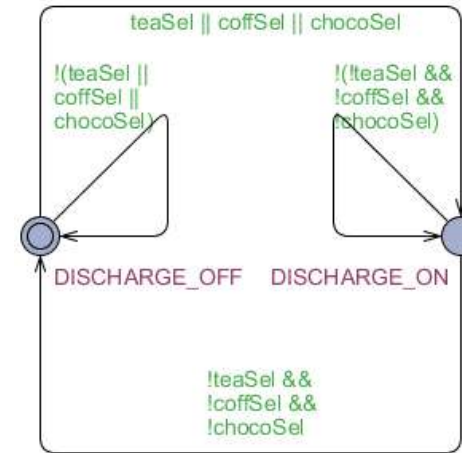
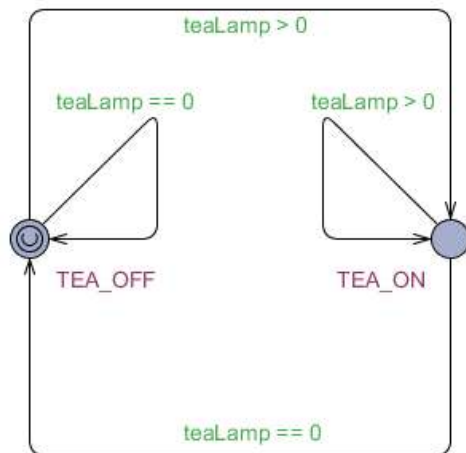
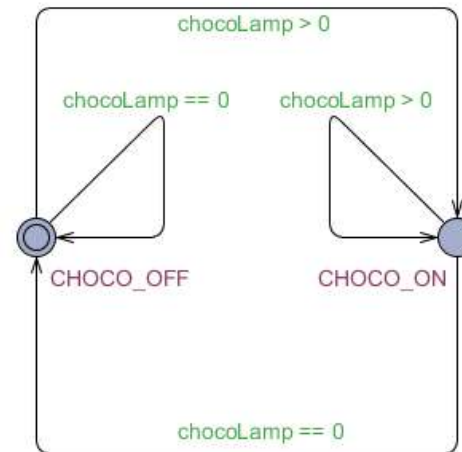
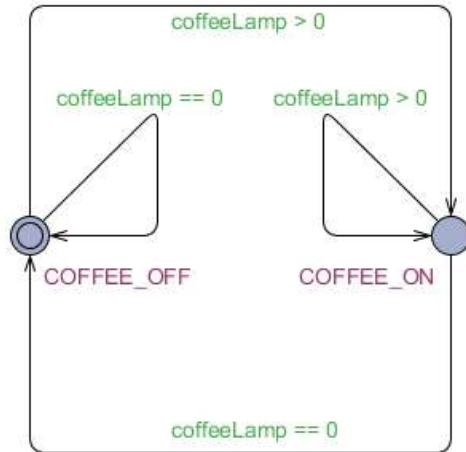




USER



LAMPS



VERIFICATION

- 유저가 반환버튼을 누르면 10초 내에 반환이 이루어진다.
 - $(\text{user.REFUND_PUSH} \ \&\& \ \text{control.PREPARE}) \ \rightarrow \ \text{control.READY}$
 $\text{imply clockVending} \leq 10$
- 음료를 준비할 수 없는 상태에서 컨트롤은 준비상태일 수 없다.
 - $A[] \ !(!\text{vending.READY} \ \&\& \ \text{control.READY})$
- 동전을 넣은 상태에서 20초 이내에 사용자는 다시 초기 상태로 돌아 올 수 있다.
 - $\text{control.PREPARE} \ \rightarrow \ \text{user.READY} \ \text{imply} \ \text{clockVending} \leq 20$
- 유저가 커피자판기를 조작하지 않을 때는 커피자판기는 대기 상태나 반환 이외의 동작을 하지 않는다.
 - $A[] \ !(!(\text{control.READY} \ || \ \text{control.REFUNDING}) \ \&\& \ \text{user.READY})$

VERIFICATION

- 커피가 나오기까지 커피 버튼등은 항상 켜져 있다.
 - $E \langle \rangle \text{control.COFF_WAITING} \ \&\& \ \text{cl.COFFEE_ON}$
- 커피와 핫 초코와 국산 차는 한번에 나올 수 없다.
 - $A[] \text{teaSel} + \text{coffSel} + \text{chocoSel} \leq 1$
- 커피 버튼을 누르면 20초 이내에 커피가 나온다.
 - $\text{user.COFF_PUSH} \ \rightarrow \ \text{vending.COFF_MAKE} \ \text{imply} \ \text{clockVending} \leq 20$
- 항상 컨트롤러와 벤더와 유저는 같은 초기시작을 갖는다.
 - $A \langle \rangle (\text{vending.READY} \ \&\& \ \text{control.READY} \ \&\& \ \text{user.READY})$
- 음료를 만드는 도중에 유저는 다른 조작을 하지 못한다.
 - $A[] \text{!}(!\text{vending.READY} \ \&\& \ \text{user.READY})$
- 데드락에 걸리지 않는다.
 - $A[] \text{not deadlock}$

VERIFICATION

```
(user.REFUND_PUSH && control.PREPARE) --> control.READY imply clockVending <= 10
A[] !(vending.READY && control.READY)
control.PREPARE --> user.READY imply clockVending <= 20
A[] !(control.READY || control.REFUNDING) && user.READY
E<> control.COFF_WAITING && cl.COFFEE_ON
A[] teaSel + coffSel + chocoSel <= 1
user.COFF_PUSH --> vending.COFF_MAKE imply clockVending <= 20
A<> (vending.READY && user.READY && control.READY)
A[] !(vending.READY && user.READY)
A[] not deadlock
```

