

Demo & Summary for Coffee Machine

Machine

Team Presentation #ver 1.0

Team [T3]

201211335 김윤식

201311281 송종원

201411317 조민규

201511282 이재승

2016/12/05 월

INDEX

➤ Coffee Machine System

➤ Reference

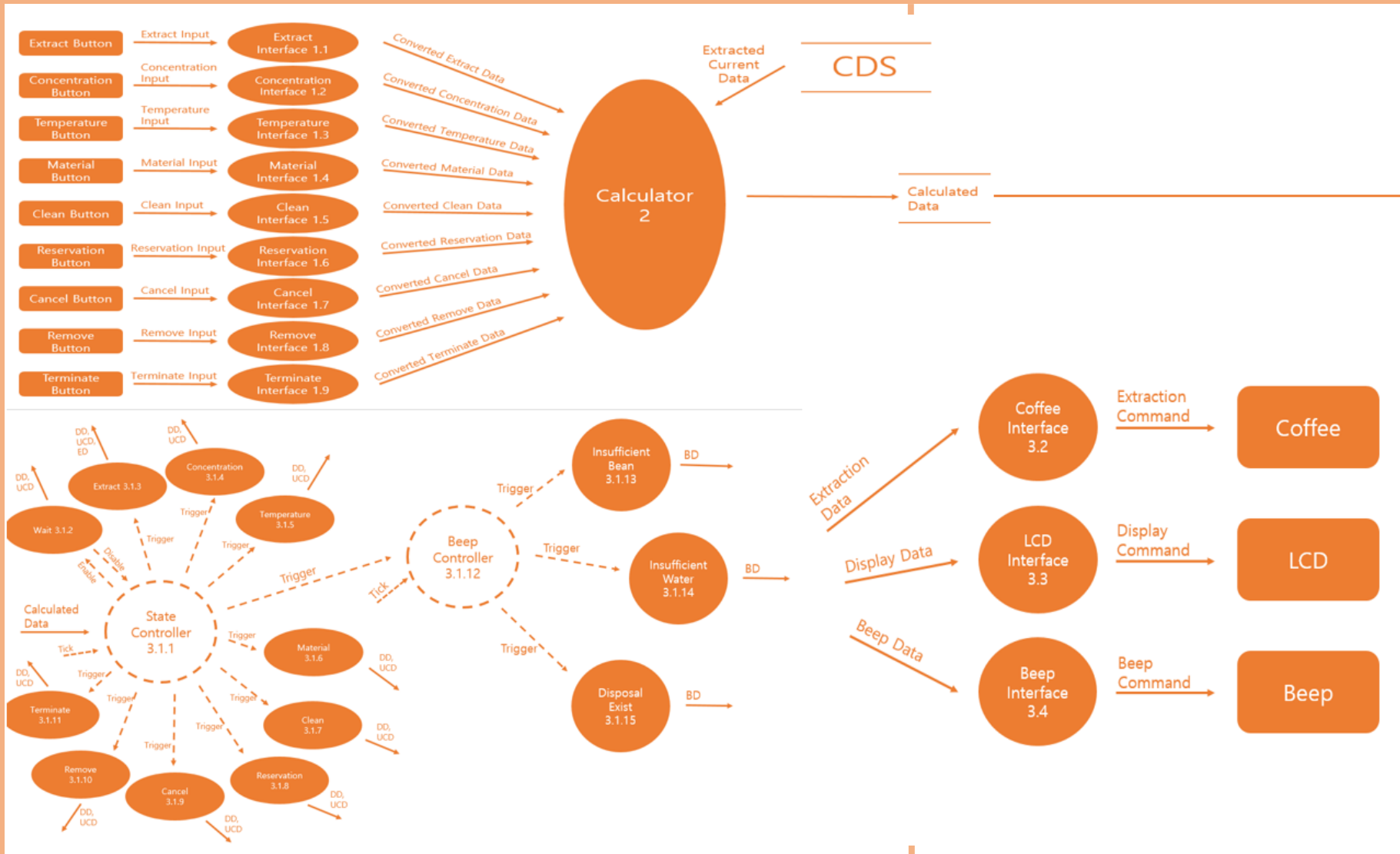
➤ Function

➤ Demo

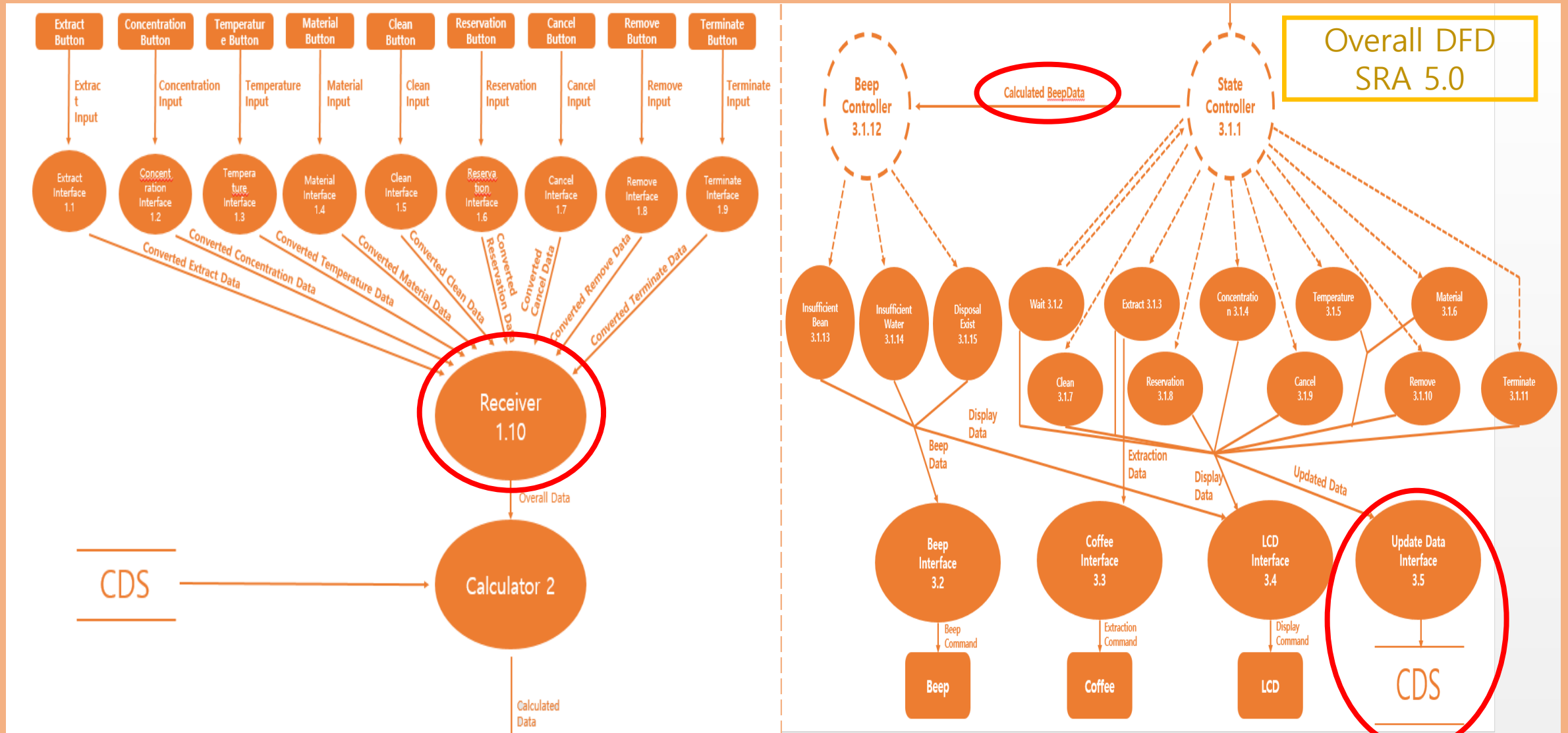
➤ Review

01 Coffee Machine System

Overall DFD
SRA 1.0

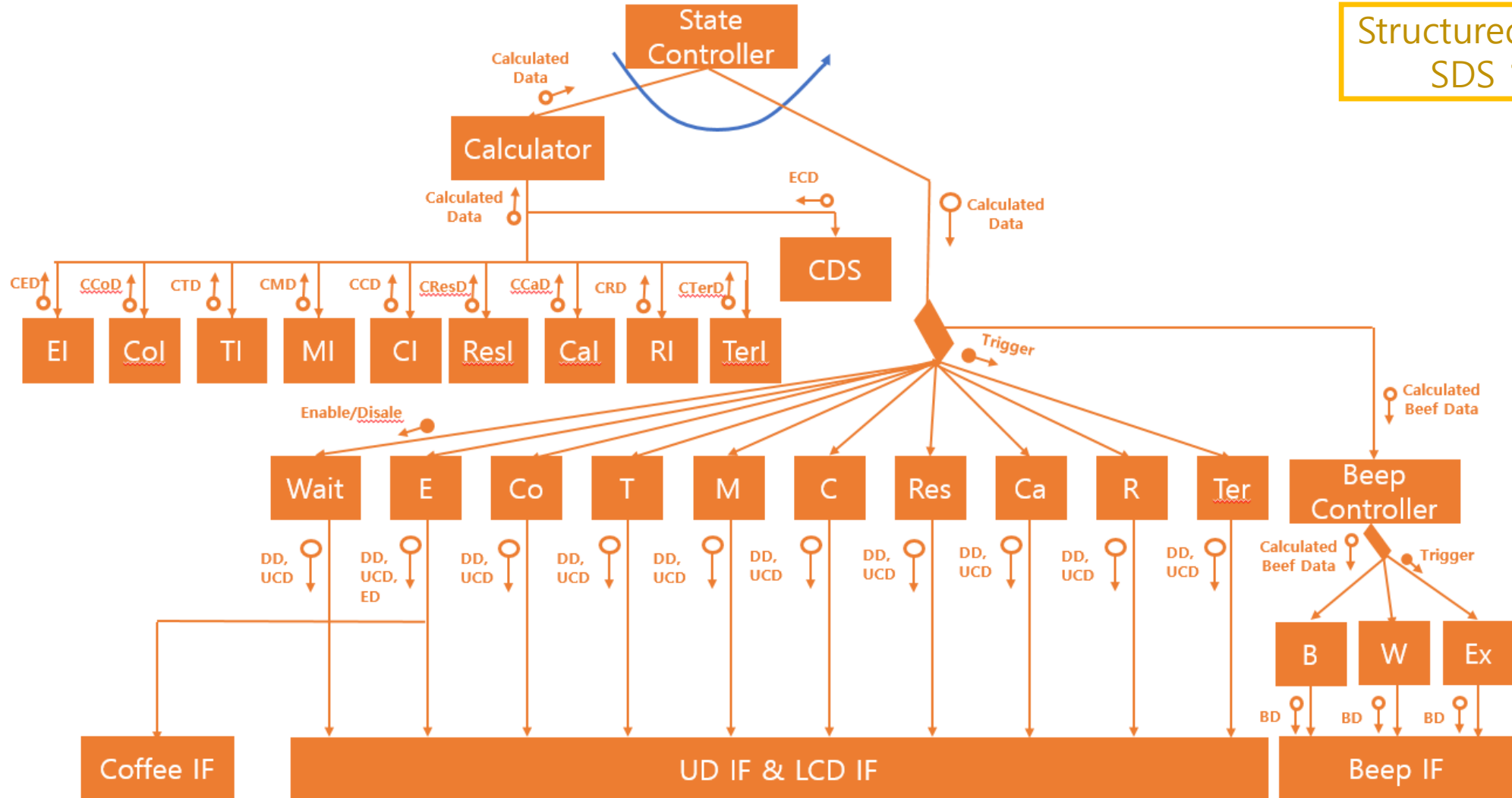


01 Coffee Machine System



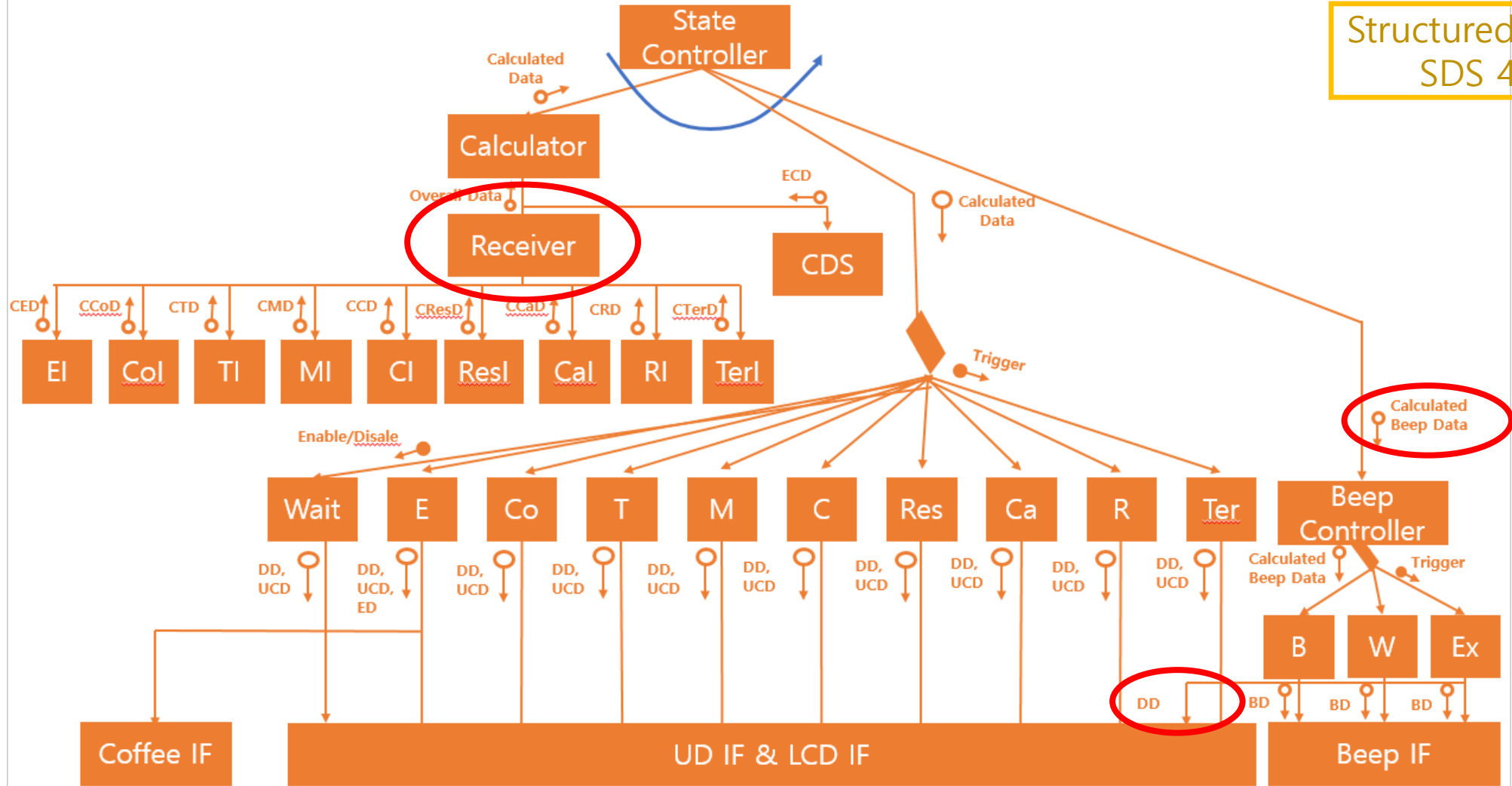
01 Coffee Machine System

Structured Chart
SDS 1.0



01 Coffee Machine System

Structured Chart
SDS 4.0



02 Reference

[2016SE_B][T3]SRA ver 5.0

Process Specification

DFD

State Transition Diagram

[2016SE_B][T3]SDA ver 4.0

Process Specification

Structure Chart

03 Function

Receiver

```
overall_data* receiver(BOOLEAN CED, int CCD, BOOLEAN CTD, material_data* CMD, BOOLEAN CLD, reservation_data* CRD, BOOLEAN CAD, BOOLEAN COD, BOOLEAN CND) {  
  
    overall_data *overallData = (overall_data *)malloc(sizeof(overall_data));  
    initOverallData(overallData);  
  
    if (!CED==FALSE) {  
        overallData->convertedExtractData = CED;  
        overallData->select = 1;  
    }  
    else {  
        .....생략  
    }  
    fflush(stdin);  
    return overallData;  
}
```


03 Function

Calculator

```
data_storage* calculator(overall_data *overallData, data_storage *currentData) {  
  
    data_storage *calculatedData = (data_storage *)malloc(sizeof(data_storage));  
    initData(calculatedData);  
    int over_select = overallData->select;  
    BOOLEAN cur_wait = currentData->wait;  
  
    if (over_select == -1 && (cur_wait)) {  
        calculatedData->wait = currentData->wait;  
    }  
    else {  
        .....나머지 생략  
    }  
  
    calculatedData->mode = overallData->select;  
    fflush(stdin);  
    return calculatedData;  
}
```

03 Function

State Controller

```
void wait(data_storage *calculatedData, data_storage *currentData);  
void extract(data_storage *calculatedData, data_storage *currentData);  
void concentration(data_storage *calculatedData, data_storage *currentData);  
void temperature(data_storage *calculatedData, data_storage *currentData);  
void material(data_storage *calculatedData, data_storage *currentData);  
void clean(data_storage *calculatedData, data_storage *currentData);  
void reservation(data_storage *calculatedData, data_storage *currentData);  
void cancel(data_storage *calculatedData, data_storage *currentData);  
void remove_(data_storage *calculatedData, data_storage *currentData);  
void terminate(data_storage *calculatedData, data_storage *currentData);  
beep_data* stateController(data_storage *calculatedData, data_storage *currentData);
```

03 Function

Beep Controller

```
void insufficientBean(beep_data* calculatedBeepData);  
void insufficientWater(beep_data* calculatedBeepData);  
void disposalExist(beep_data* calculatedBeepData);  
void beepInterface(int beepData);  
void beepController(beep_data* calculatedBeepData);
```

DEMO

04 Review

한 학기 동안 수고 많으셨습니다.
소프트웨어 공학개론이라는 수업을 통해,
배운 것을 다 안다고 할 수는 없지만,
많은 것을 배운 것 같습니다.

처음 배우는 내용들이 많아서인지, 모르는 부분을
물어본다고 조교님께 계속 메일 보내서 귀찮으셨을 텐데,
친절하게 알려주셔서 감사합니다.
이제 학기가 끝나가는데, 모두 마지막까지 수고하세요~

201311281 송종원

소공수업을 듣기 전에는 소프트웨어에도 무작정 개발하지
않고 사전에 계획하고, 준비하는 단계가 있을 거라 생각은
해봤지만 이렇게 체계적이고 세세한 부분인지 몰랐다.
구현뿐만 아니라 설계와 테스트의 중요성을 깨닫게 되었다.
질문이나 메일을 드리면 항상 정성스럽게 답해주시고
피드백 주시는 조교님과 팀원들에게도 감사하다고
전하고 싶다.

201211335 김윤식

이번 수업을 통해 직접 소프트웨어를 설계해보고 개발하고
테스팅하면서 소프트웨어의 설계가 얼마나 중요한지, 그리
고 설계에 따른 구현은 얼마나 힘든지 직접적으로 체험해본
것 같다. 컴퓨터공학과에 어느 정도 적응한 후 처음 제대로
체감해본 설계 수업이라 실수도 많고, 부족한 점도 많았지만
질문들 및 시행착오들을 통해서 많이 배우고 그로 인해 한
단계 더 성장했다고 생각한다. 앞으로도 소프트웨어에 대한
많은 것들을 공부하고 나만의 개발방법론도 생각해보면 좋
을 것 같다.

2014111317 조민규

그 동안 여러 프로젝트를 해봤지만 이렇게 체계적으로
SASD를 이용하여 해본 건 처음이었기 때문에 많이 힘들기
도 했지만 많은 걸 배우게 된 수업이었습니다.

201511282 이재승

Q&A

THANK YOU