

Unit Testing Plan for Digital Watch System

- Test Plan
- Test Design Specification
- Test Cases Specification

Project Team

이동아

Date

2012-10-23

Team Information

이동아: dalee.dslab@gmail.com

Table of Contents

1	Introduction	4
1.1	Objectives.....	4
1.2	Background	4
1.3	Scope.....	4
1.4	Project plan	4
1.5	Configuration management plan.....	4
1.6	References.....	4
2	Test items	4
3	Approach.....	5
4	Item pass/fail criteria	6
5	Unit test design specification.....	6
5.1	Test design specification identifier	6
5.2	Features to be tested	6
5.3	Approach refinements.....	6
5.4	Test identification	6
5.5	Feature pass/fail criteria	6
6	Unit test case specification.....	6
6.1	Test case specification identifier	6
6.2	Test items	6
6.3	Input specifications.....	6
6.4	Output specifications.....	7

7	Testing tasks	7
8	Environmental needs	7
9	Unit Test deliverables	7
10	Schedules	7

1 Introduction

1.1 Objectives

본 문서는 2012년 2학기 SE 수업의 T2가 개발한 Digital Watch System (DWS)을 Unit Testing하기 위한 계획문서이다. T2가 정의한 Unit을 Testing을 수행하기 위하여 Testing Pass/Fail Criteria를 정의하고 이를 수행하기 위한 Test design & test cases를 제작한다.

1.2 Background

2012년 2학기 SE수업에서 개발하는 모든 DWS는 SASD기법을 이용하여 개발된다. 기능 별로 나눠진 Unit은 SRA, SDS 문서에 모두 정의되어 있다.

1.3 Scope

SRA 및 SDS 문서에 정의된 Unit을 Testing한다.

1.4 Project plan

1.5 Configuration management plan

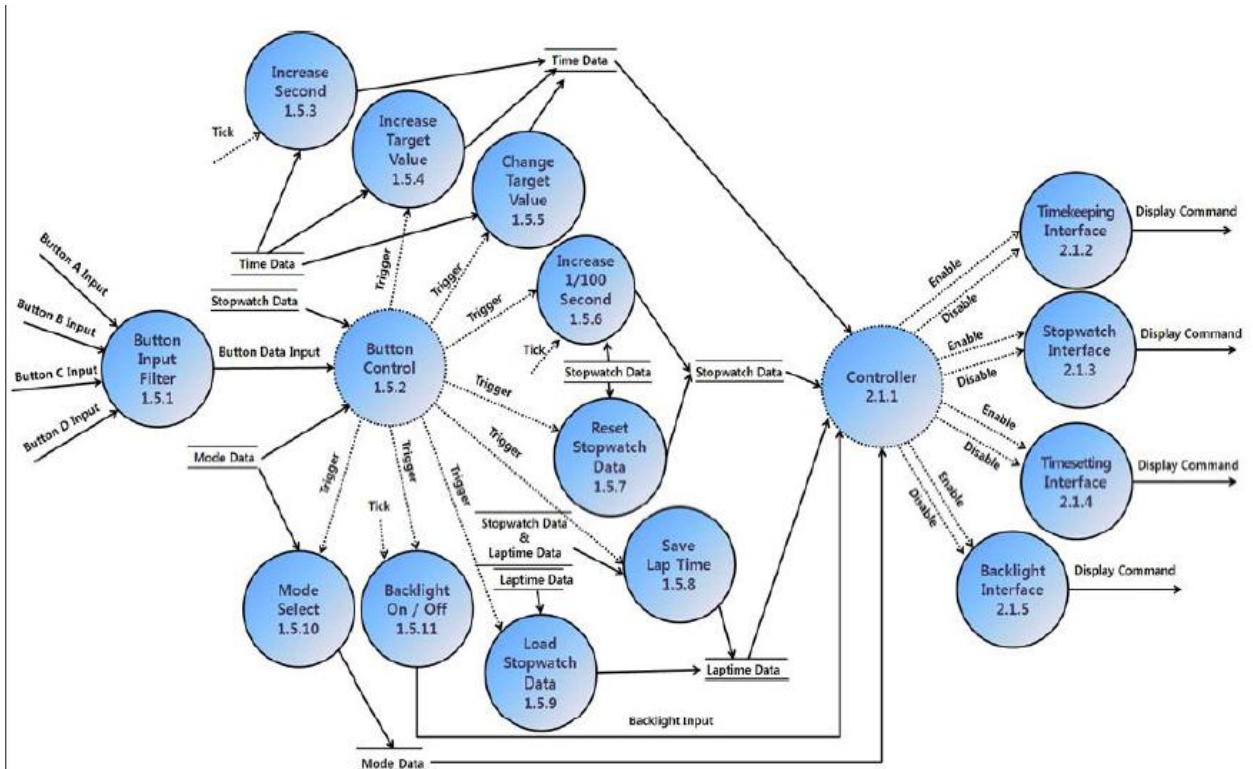
1.6 References

T2-2012.DWS.SRA-2.1

T2-2012.DWS.SDS-1.1

2 Test items

T2가 SASD기법을 이용하여 개발한 DWS을 Testing한다. SA와 SD에서 분류한 각 process/module 별로 Testing을 수행한다. <오류! 참조 원본을 찾을 수 없습니다.>은 SA를 이용하여 요구사항을 분석한 결과 중 전체 DFD를 나타낸 그림이며, <오류! 참조 원본을 찾을 수 없습니다.>은 SD의 Basic Structural Chart를 나타낸 그림이다. 각 그림을 참조하여 Unit을 지정하고, 지정한 Unit을 SRA에 명세 된 내용과 같은 동작을 하는지 확인한다.



Team 2

20

Figure 1 Overall DFD

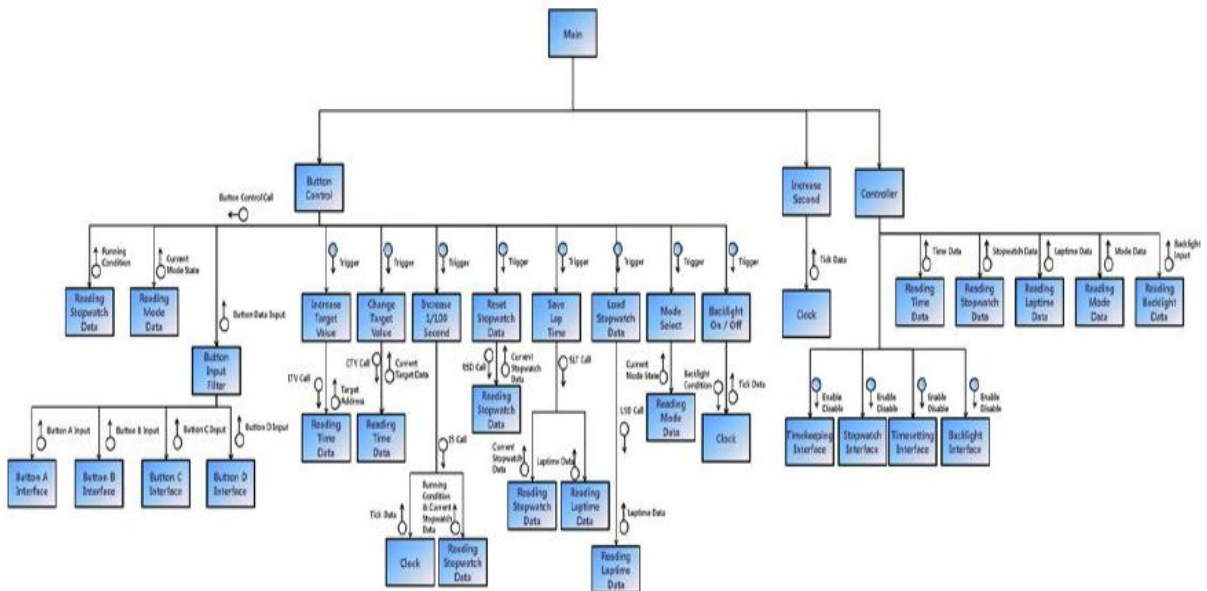


Figure 2 Structural Chart

Brute Force, SRA의 가장 하위 Level의 Process부터 Testing을 시작하여 상위 Level의 Process 까지 Testing한다.

4 Item pass/fail criteria

모든 Units은 SRA에 명시된 Process Specification과 동일한 결과를 출력해야 한다. 각 Unit별 Pass/Fail Criteria는 <>를 참조한다.

5 Unit test design specification

5.1 Test design specification identifier

DWS.UTC.Number

5.2 Features to be tested

Process 1-2 (DWS.UTC.100 – 200)

Process 1.1-1.5 (DWS.UTC.110 – 150)

Process 2.1-2.2 (DWS.UTC.210 – 220)

Process 1.5.1-1.5.11 (DWS.UTC.151 – 15B)

Process 2.1.1-2.1.5 (DWS.UTC.211 - 215)

5.3 Approach refinements

각 Process Specification에 명시된 내용을 기반으로 Test Design 및 Test Cases를 생성해 낸다.

5.4 Test identification

DWS.UTC.Number

5.5 Feature pass/fail criteria

6 Unit test case specification

6.1 Test case specification identifier

6.2 Test items

6.3 Input specifications

- 6.4 Output specifications
- 7 Testing tasks
- 8 Environmental needs
- 9 Unit Test deliverables
- 10 Schedules

Table 1 Test Design Specification

Identifier	Feature	Pass/Fail
DWS.UTC.100	Process Spec 없음	
DWS.UTC.200	Process Spec 없음	
DWS.UTC.110	A 입력을 받으면 True 출력, 아니면 False	
DWS.UTC.120	B 입력을 받으면 True 출력, 아니면 False	
DWS.UTC.130	C 입력을 받으면 True 출력, 아니면 False	
DWS.UTC.140	D 입력을 받으면 True 출력, 아니면 False	
DWS.UTC.150	Process Spec 없음	
DWS.UTC.210	Process Spec 없음	
DWS.UTC.220	Process Spec 없음	
DWS.UTC.151	입력된 버튼 A, B, C, D에 따라 출력해 준다. - 정확한 값 정의가 되어있지 않음	
DWS.UTC.152	State Transition Diagram 확인 어려움	
DWS.UTC.153	Tick이 올 때마다 Time Data의 초에 +1 Time Data의 초가 60이 되면 0으로 초기화	
DWS.UTC.154	Trigger 되면 Time Data의 Target을 1 증가 시킴 Target이 지날 수 있는 값의 범위는?	
DWS.UTC.155	Trigger될 때마다 Time Data의 Target 정보를 1씩 증가시킴.	
DWS.UTC.156	Trigger될 때마다 & 10ms 마다 Stopwatch Data.동작여부 에 따라 동작 중 - Stopwatch Data.동작여부 중단 동작 중 아님 - Stopwatch Data.동작여부 를 동작으로 수정	

	Stopwatch data.측정시간 값을 증가시킴	
DWS.UTC.157	Trigger되면 Stopwatch Data.측정시간 을 0으로 초기화함.	
DWS.UTC.158	Trigger 되면 Stopwatch data.측정시간 을 읽어와 Laptime Data에 저장. Laptime Data.Display우선순위 값을 변경 (값이 어떻게 변경?)	
DWS.UTC.159	Trigger 되면 Laptime Data.Display 우선순위를 낮게 변경	
DWS.UTC.15A	Trigger되면 Mode Data를 변경 (Mode Data를 어떻게 변경..?)	
DWS.UTC.15B	Trigger되면 2초간 Backlight Data를 (어떤 데이터 값을?) 출력함	
DWS.UTC.211	State Transition Diagram 확인 어려움	
DWS.UTC.212	Enable 되면 주어진(?) Data들을 Timekeeping의 형태(?)로 출력	
DWS.UTC.213	Enable 되면 주어진(?) Data들을 Stopwatch의 형태(?)로 출력	
DWS.UTC.214	Enable 되면 주어진(?) Data들을 Time setting의 형태(?)로 출력	
DWS.UTC.215	Process Specification 상의 불분명함 존재.	

Table 2 Test Case Identification

Identifier	Feature	Output
DWS.UTC.100	Process Spec 없음	
DWS.UTC.200	Process Spec 없음	
DWS.UTC.110	A 입력을 받으면 True 출력, 아니면 False	
DWS.UTC.120	B 입력을 받으면 True 출력, 아니면 False	
DWS.UTC.130	C 입력을 받으면 True 출력, 아니면 False	
DWS.UTC.140	D 입력을 받으면 True 출력, 아니면 False	
DWS.UTC.150	Process Spec 없음	
DWS.UTC.210	Process Spec 없음	
DWS.UTC.220	Process Spec 없음	
DWS.UTC.151	입력된 버튼 A, B, C, D에 따라 출력해 준다. - 정확한 값 정의가 되어있지 않음	int buttonFilter()
	Scanf로 받는 것을 강제 입력으로 수정하여 진행하려 했으나 내부 변수 파악까지 해야 하는 관계로 보류	
DWS.UTC.152	State Transition Diagram 확인 어려움	
DWS.UTC.153	Tick이 올 때마다 Time Data의 초에 +1 Time Data의 초가 60이 되면 0으로 초기화	void increaseSecond(DataP data)
DWS.UTC.153.00	DataPyear == 2012 DataPmonth == 1 DataP.date == 1 DataPhour == 0 DataPminute == 0	DataPyear == 2012 DataPmonth == 1 DataP.date == 1 DataPhour == 0 DataPminute == 0

	DataP.second == 0	DataP.second == 1
DWS.UTC.153.01	DataP.year == 2012 DataP.month == 1 DataP.date == 1 DataP.hour == 0 DataP.minute == 0 DataP.second == 59	DataP.year == 2012 DataP.month == 1 DataP.date == 1 DataP.hour == 0 DataP.minute == 1 DataP.second == 0
DWS.UTC.153.02	DataP.year == 2012 DataP.month == 1 DataP.date == 1 DataP.hour == 0 DataP.minute == 59 DataP.second == 59	DataP.year == 2012 DataP.month == 1 DataP.date == 1 DataP.hour == 1 DataP.minute == 0 DataP.second == 0
DWS.UTC.153.03	DataP.year == 2012 DataP.month == 1 DataP.date == 1 DataP.hour == 23 DataP.minute == 59 DataP.second == 59	DataP.year == 2012 DataP.month == 1 DataP.date == 2 DataP.hour == 0 DataP.minute == 0 DataP.second == 0
DWS.UTC.153.04	DataP.year == 2012 DataP.month == 1 DataP.date == 1 DataP.hour == 23 DataP.minute == 59 DataP.second == 59	DataP.year == 2012 DataP.month == 1 DataP.date == 2 DataP.hour == 0 DataP.minute == 0 DataP.second == 0

DWS.UTC.153.05	DataP.year == 2012 DataP.month == 1 DataP.date == 31 DataP.hour == 23 DataP.minute == 59 DataP.second == 59	DataP.year == 2012 DataP.month == 2 DataP.date == 1 DataP.hour == 0 DataP.minute == 0 DataP.second == 0
DWS.UTC.153.06	DataP.year == 2012 DataP.month == 2 DataP.date == 28 DataP.hour == 23 DataP.minute == 59 DataP.second == 59	DataP.year == 2012 DataP.month == 2 DataP.date == 29 DataP.hour == 0 DataP.minute == 0 DataP.second == 0
DWS.UTC.153.07	DataP.year == 2013 DataP.month == 2 DataP.date == 28 DataP.hour == 23 DataP.minute == 59 DataP.second == 59	DataP.year == 2012 DataP.month == 3 DataP.date == 1 DataP.hour == 0 DataP.minute == 0 DataP.second == 0
DWS.UTC.153.08	DataP.year == 2012 DataP.month == 3 DataP.date == 31 DataP.hour == 23 DataP.minute == 59 DataP.second == 59	DataP.year == 2012 DataP.month == 4 DataP.date == 1 DataP.hour == 0 DataP.minute == 0 DataP.second == 0
DWS.UTC.153.09	DataP.year == 2012	DataP.year == 2012

	DataP.month == 4 DataP.date == 30 DataP.hour == 23 DataP.minute == 59 DataP.second == 59	DataP.month == 5 DataP.date == 1 DataP.hour == 0 DataP.minute == 0 DataP.second == 0
DWS.UTC.153.10	DataP.year == 2012 DataP.month == 5 DataP.date == 31 DataP.hour == 23 DataP.minute == 59 DataP.second == 59	DataP.year == 2012 DataP.month == 6 DataP.date == 1 DataP.hour == 0 DataP.minute == 0 DataP.second == 0
DWS.UTC.153.11	DataP.year == 2012 DataP.month == 6 DataP.date == 30 DataP.hour == 23 DataP.minute == 59 DataP.second == 59	DataP.year == 2012 DataP.month == 7 DataP.date == 1 DataP.hour == 0 DataP.minute == 0 DataP.second == 0
DWS.UTC.153.12	DataP.year == 2012 DataP.month == 7 DataP.date == 31 DataP.hour == 23 DataP.minute == 59 DataP.second == 59	DataP.year == 2012 DataP.month == 8 DataP.date == 1 DataP.hour == 0 DataP.minute == 0 DataP.second == 0
DWS.UTC.153.13	DataP.year == 2012 DataP.month == 8	DataP.year == 2012 DataP.month == 9

	DataP. date == 31 DataP.hour == 23 DataP.minute == 59 DataP.second == 59	DataP. date == 1 DataP.hour == 0 DataP.minute == 0 DataP.second == 0
DWS.UTC.153.14	DataP.year == 2012 DataP.month == 9 DataP. date == 30 DataP.hour == 23 DataP.minute == 59 DataP.second == 59	DataP.year == 2012 DataP.month == 10 DataP. date == 1 DataP.hour == 0 DataP.minute == 0 DataP.second == 0
DWS.UTC.153.15	DataP.year == 2012 DataP.month == 10 DataP. date == 31 DataP.hour == 23 DataP.minute == 59 DataP.second == 59	DataP.year == 2012 DataP.month == 11 DataP. date == 1 DataP.hour == 0 DataP.minute == 0 DataP.second == 0
DWS.UTC.153.16	DataP.year == 2012 DataP.month == 11 DataP. date == 30 DataP.hour == 23 DataP.minute == 59 DataP.second == 59	DataP.year == 2012 DataP.month == 12 DataP. date == 1 DataP.hour == 0 DataP.minute == 0 DataP.second == 0
DWS.UTC.153.17	DataP.year == 2012 DataP.month == 12 DataP. date == 31	DataP.year == 2013 DataP.month == 11 DataP. date == 1

	DataP.hour == 23 DataP.minute == 59 DataP.second == 59	DataP.hour == 0 DataP.minute == 0 DataP.second == 0
DWS.UTC.154	Trigger 되면 Time Data의 Target을 1 증가 시킴	void changeTargetValue(DataP data)
DWS.UTC.154.00	DataP.target	DataP.Target ++
DWS.UTC.155	Trigger될 때마다 Time Data의 Target 정보를 1씩 증가시킴.	
DWS.UTC.155.00	Trigger DataP.target = ??? DataP.Time	(Target으로 지정된)DataP.Time ++
DWS.UTC.156	Trigger될 때마다 & 10ms 마다 Stopwatch Data.동작여부 에 따라 동작 중 – Stopwatch Data.동작여부 중단 동작 중 아님 – Stopwatch Data.동작여부 를 동작으로 수정 Stopwatch data.측정시간 값을 증가시킴	void increase10mSecond(DataP data)
DWS.UTC.156.00	Trigger DataP. runningCondition == 1 가동	DataP. runningCondition == 0 중단
DWS.UTC.156.01	Trigger DataP. runningCondition == 0 중단	DataP. runningCondition == 1 가동 DataP. stopwatchData += 10ms
DWS.UTC.157	Trigger되면 Stopwatch Data.측정시간 을 0으로 초기화함.	
DWS.UTC.157.00	Trigger DataP	Data.lapTime == 0 data.stopwatchdata == 0
DWS.UTC.158	Trigger 되면 Stopwatch data.측정시간 을 읽어와 Laptime Data에 저장. Laptime Data.Display우선순위 값을 변경 (값이 어떻게 변경?)	void saveLaptime(DataP data)
DWS.UTC.158.00	Trigger DataP	Data.lapTime == data.stopwatchdata Data.lapdisplay ==1
DWS.UTC.159	Trigger 되면 Laptime Data.Display 우선순위를 낮게 변경	void loadStopwatchData(DataP data)

DWS.UTC.159.00	Spec만으로 I/O 결과를 유추할 수 없음. Code를 보고 생성 Trigger DataP	DataP->lapDisplay = 0
DWS.UTC.15A	Trigger되면 Mode Data를 변경 (Mode Data를 어떻게 변경..?) 구현 내용과 spec이 다름.	void modeSelect(DataP data, int button)
DWS.UTC.15B	Trigger되면 2초간 Backlight Data를 (어떤 데이터 값을?) 출력함	void backlightOn(DataP data)
DWS.UTC.15B.00	Tick을 자체적으로 만들어서 사용함.	
DWS.UTC.211	State Transition Diagram 확인 어려움	
DWS.UTC.212	Enable 되면 주어진(?) Data들을 Timekeeping의 형태(?)로 출력 Interface 미구현	
DWS.UTC.213	Enable 되면 주어진(?) Data들을 Stopwatch의 형태(?)로 출력 Interface 미구현	
DWS.UTC.214	Enable 되면 주어진(?) Data들을 Time setting의 형태(?)로 출력 Interface 미구현	
DWS.UTC.215	Process Specification 상의 불분명함 존재. Interface 미구현	