

Unit Testing Plan for Digital Watch System

- Test Plan
- Test Case Specification

Project Team

이동아

Date

2012-11-07

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.....	6
4	Item pass/fail criteria	6
5	Unit test case specification.....	6
5.1	Test case specification identifier	6
5.2	Features to be tested	6
5.3	Feature not to be tested.....	6
5.4	Approach refinements.....	7
5.5	Test identification	7
5.6	Feature pass/fail criteria	7
5.7	Input specifications.....	7
5.8	Output specifications.....	7
6	Testing tasks	7
7	Environmental needs.....	7

8	Unit Test deliverables	8
9	Schedules	8

1 Introduction

1.1 Objectives

본 문서는 2012년 2학기 SE 수업의 T1이 개발한 두 번째 버전의 Digital Watch System (DWS)을 Unit Testing하기 위한 계획문서이다. T1이 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 나뉜 Unit은 SRA, SDS 문서에 모두 정의되어 있다.

1.3 Scope

SRA 및 SDS 문서에 정의된 Unit을 Testing한다.

1.4 Project plan

Unit Testing Plan을 작성한다.

Unit Testing을 수행한다.

Unit Testing Report를 작성한다.

1.5 Configuration management plan

1.6 References

T1-2012.DWS.SRA-2.3

T1-2012.DWS.SDS-3.0

2 Test items

T1이 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.

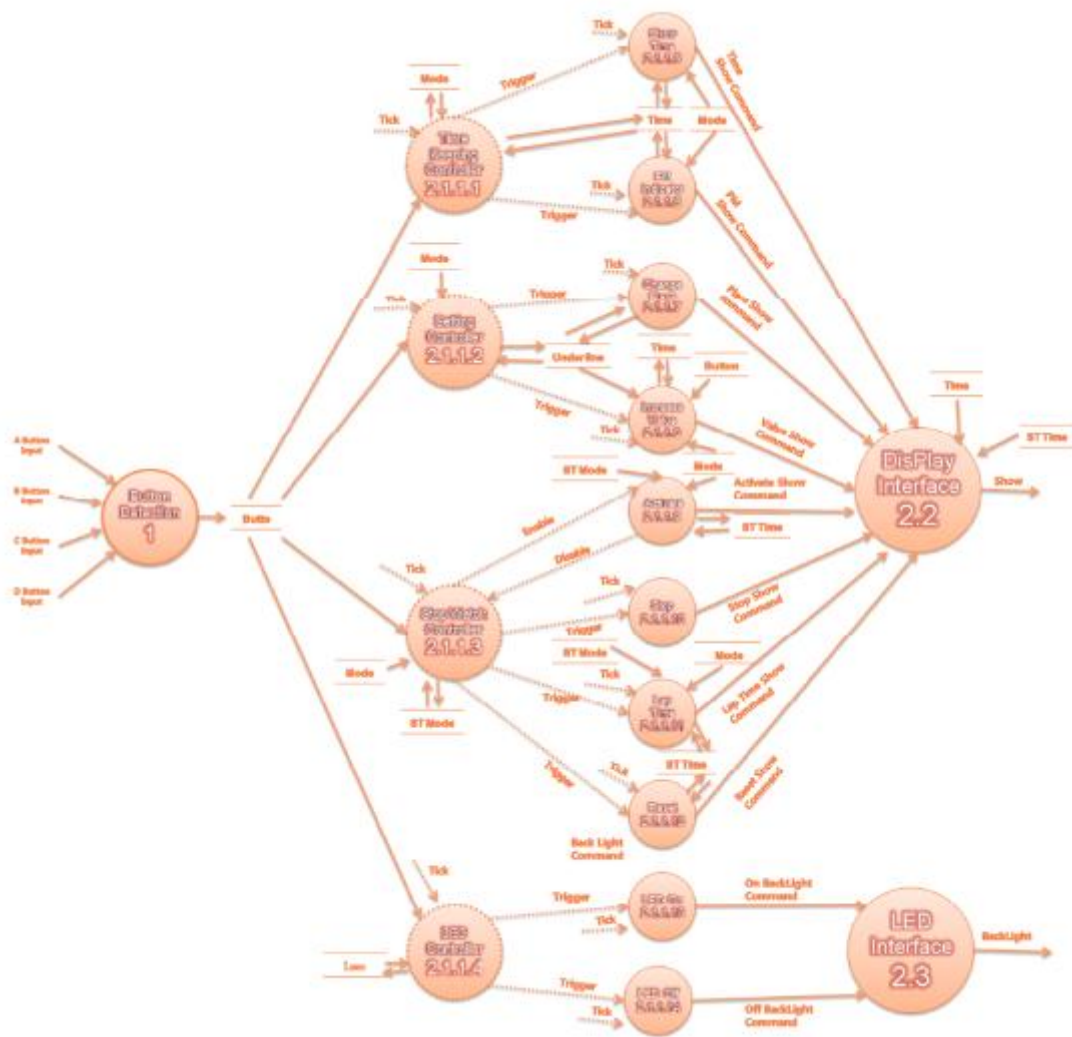


Figure 1 Overall DFD

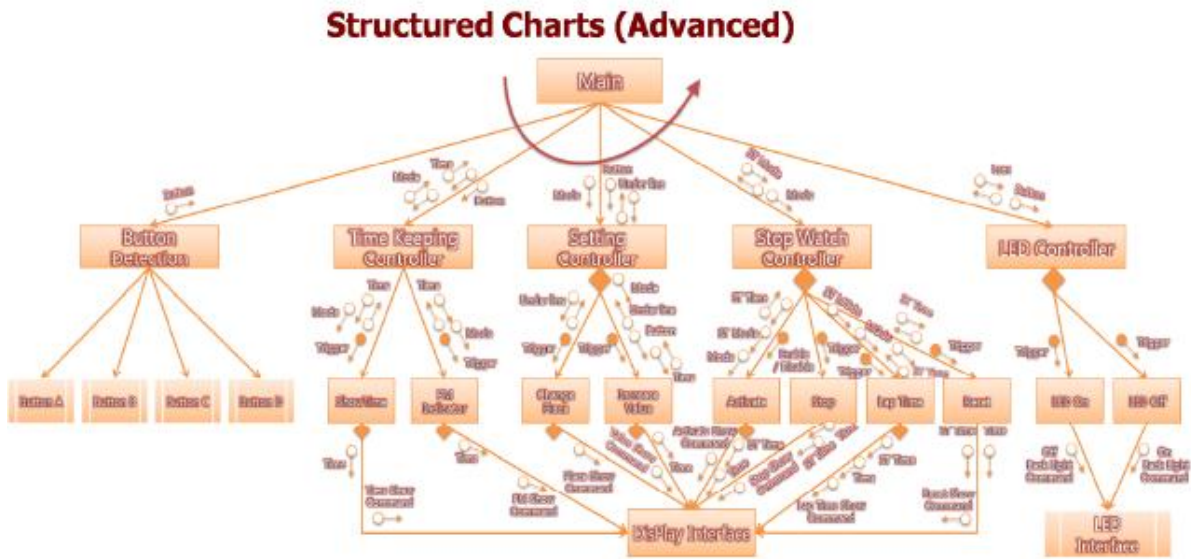


Figure 2 Structural Chart

3 Approach

Brute Force, SRA의 가장 하위 Level의 Process를 대상으로 Testing을 수행한다.

4 Item pass/fail criteria

모든 Units은 SRA에 명시된 Process Specification과 동일한 결과를 출력해야 한다. 각 Unit별 Pass/Fail Criteria는 <Table 2 Test Case Identification>를 참조한다.

5 Unit test case specification

5.1 Test case specification identifier

DWS.UTC.Number

5.2 Features to be tested

최하위 Modules 및 Processes

Process 1

Process 2.1.1.1-2.1.1.14

Process 2.2-2.3

5.3 Feature not to be tested

최상위 및 중간 Modules 및 Processes

5.4 Approach refinements

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

5.5 Test identification

<Table 2 Test Case Identification> 참조

5.6 Feature pass/fail criteria

<Table 2 Test Case Identification> 참조

5.7 Input specifications

<Table 2 Test Case Identification> 참조

5.8 Output specifications

<Table 2 Test Case Identification> 참조

6 Testing tasks

Table 1 Testing tasks & Schedule

Task	Predecessor tasks	Special skills	Effort	Finish date
(1) Unit Test Plan 작성	SRA 작성 SDS 작성 DWS 구현		3	
(2) Test design specification	Task 1	DWS 에 대한 이해	5	
(3) Test case specification	Task 2	DWS 에 대한 이해	5	
(4) Test execution	Task 3	Test code 작성 Test tools 에 대한 이해	4	
(5) Test result report	Task 4		1	
(6) 개발팀에 report 전달	Task 5		1	

7 Environmental needs

Eclipse IDE for C/C++ developers

Visual SVN server

Hudson CI

8 Unit Test deliverables

Unit Testing Report

9 Schedules

<Table 1 Testing tasks & Schedule> 참조

Table 2 Test Case Identification

Identifier	Feature	Pass/Fail, Comments
DWS.UTC.1000	A,B,C,D 버튼을 입력 받아서 Button에 저장	데이터가 올바르게 저장되는가?
DWS.UTC.2111	Tick마다 mil_sec 1 증가, Show Time, PM Indicator trigger시킴 100Tick마다 1초 증가 Mode와 Button에 따라서 다음 실행될 Mode로 변경	다음 실행될 모드에 대한 정의 없음
DWS.UTC.2112	Mode가 Setting이 되면 Under line = 1, trigger change place Button == c; trigger 'change place' Button == b; trigger 'increase value'	Description과 STD의 내용이 다름. STD 내용 기반으로 진행
DWS.UTC.2113	STD 기반으로 Testing 수행	
DWS.UTC.2114	STD 기반으로 Testing 수행	
DWS.UTC.2115	Time으로부터 요일 구함 윤년 계산 초, 분 == 60; 초, 분 = 0; 다음 단위증가 Mode == TimeKeeping; Time Show command를 내보낸다.	요일 확인 유년에 따른 2월의 일자 계산 확인 초, 분 계산 확인 출력 확인
DWS.UTC.2116	Time의 오전, 오후를 확인해 PM에 저장 Mode == TimeKeeping Setting; PM Show command 출력	PM Mode와 해당시간을 조합하여 PM에 저장? 오전, 오후에 따른 PM 값 확인 출력 확인
DWS.UTC.2117	Trigger; Under Line++ Under Line == 6일 때 trigger; Under Line = 1 Under Line의 값에 따라 Place Show Command 출력	값 변화 확인 조건에 따른 값 변화 확인 출력 확인
DWS.UTC.2118	Button=B일 때 Trigger; Under line의 값에 따라 Time++;	값 증가 확인

	Time++ 시 최대값 검사	최대 값 검사 확인
DWS.UTC.2119	1/100단위로 시간 증가 ST Mode == 0; ST Time = 0 ST Mode == 1 5; ST Time += 1/100; 초의 최대 값 검사 StopWatch Mode & !LapTime; Activate Show Command 출력	시간 증가 확인 시간 초기화 확인 시간의 최대 값 확인 출력 확인
DWS.UTC.211A	Trigger; Stop Show Command 출력	출력 확인
DWS.UTC.211B	Trigger; Lap_시간 = ST Time 만약 Mode == StopWatch Mode; LapTime Show Command 출력	시간 저장 확인 조건에 따른 출력 확인
DWS.UTC.211C	Trigger; ST Time = 0; Reset Show Command 출력	초기화 및 출력 확인
DWS.UTC.211D	Trigger; On Back Light Command (노란색 글씨 출력)	출력 확인
DWS.UTC.211E	Trigger; Off Back Light Command (검정색 글씨 출력)	Description이 이해하기 어려움 출력 확인
DWS.UTC.2200	Show command를 입력 받아 종류에 따라서 Time Data 또는 ST Time Data를 Show로 출력	어떤 종류일 때 무엇이 나가는지 확인 불가
DWS.UTC.2300	Back Light Command를 입력 받아 Light로 출력	데이터 처리에 대한 정보 없음

Table 3 Test Data Identification

Identifier	Input	Pass/Fail Criteria	비고
DWS.UTC.1000.00	System 함수 호출로 인해 테스트 불가	PASS	void button_detection()
DWS.UTC.2111.00	호출	Time.mil_sec == 1, Trigger "Show Time"; 모드 변경 정보를 알 수 없음 (FAIL)	void timekeeping()
DWS.UTC.2111.01	100회 호출	Time.mil_sec == 0 & Time.sec == 1; Trigger "PM Indicator" & "Show Time"; 모드 변경정보를 알 수 없음 (FAIL)	void timekeeping()
DWS.UTC.2112.00	State == ?? (초기 호출)	변화 없음	void setting()
DWS.UTC.2112.01	Mode == 1 && Button == 'A'	Trigger "Change Place"	
DWS.UTC.2112.02	State == ChagePlace & Button == 'B'	Trigger "Increase Value"	
DWS.UTC.2112.03	State == ChagePlace & Button == 'C'	Trigger "Change Place"	
DWS.UTC.2113.00	State ==?? && ST Mode != 0 && ST Mode != 1 && ST Mode != 5, (초기 호출)	변화 없음	void stopwatch()
DWS.UTC.2113.01	State ==? && ST Mode == 0	Enable "Activate"	
DWS.UTC.2113.02	State ==? && ST Mode == 1	Enable "Activate"	
DWS.UTC.2113.03	State ==? && ST Mode == 5	Enable "Activate"	
DWS.UTC.2113.04	State == Activate && Button == 'B' && ST Mode == 1	Trigger "Stop", Disable "Activate", State == Stop	
DWS.UTC.2113.05	State == Activate && ST Mode == 0	Trigger "Initialization", State == Activate	
DWS.UTC.2113.06	State == Activate && Button == 'A' && ST Mode == 1	Trigger "LapTime", State == Activate	

DWS.UTC.2113.07	State == LapTime && Button == 'A' && ST Mode == 5	Trigger "Laptime", State == LapTime	
DWS.UTC.2113.08	State == Activate && Button == 'B'	State == Activate	
DWS.UTC.2113.09	State == Stop && Button == 'B'	Enable "Activate", State == Activate	
DWS.UTC.2113.10	State == Stop && Button == 'A'	Trigger "Reset", State == Reset	
DWS.UTC.2113.11	State == Reset && Button == 'B'	Enable "Activate", State == Activate	
DWS.UTC.2114.00	State == ?? & 호출	State == LED Off, Trigger "LED Off"	<code>void LED()</code>
DWS.UTC.2114.01	State == LED Off && Button == 'D'	Trigger "LED On", State == LED On	
DWS.UTC.2114.02	State == LED On && Lsec == 200	Trigger "LED Off", State == LED Off	
DWS.UTC.2115.00	Time == 2012.11.07	Wday == 수요일	<code>void showtime()</code>
DWS.UTC.2115.01	Time == 2022.01.04	Wday == 화요일	
DWS.UTC.2115.02	Time == 2012.02	2월의 마지막 일수 == 29	
DWS.UTC.2115.03	Time == 2013.02	2월의 마지막 일수 == 28	
DWS.UTC.2115.04	Time == 10:29:60	Time == 10:30:00	
DWS.UTC.2115.05	Time == 10:59:60	Time == 11:00:00	
DWS.UTC.2115.06	Mode == TimeKeeping	Time Show Command 출력	
DWS.UTC.2116.00	Time < 12:00:00	PM == 0	<code>void pm_indicator()</code>
DWS.UTC.2116.01	Time >= 12:00:00	PM == 1	
DWS.UTC.2116.02	Mode == TimeKeeping	PM Show Command 출력	
DWS.UTC.2116.03	Mode == Setting	PM Show Command 출력	
DWS.UTC.2117.00	Under Line == 1	Under Line == 2, Place Show Command 출력	<code>void change_place()</code>
DWS.UTC.2117.01	Under Line == 6	Under Line == 1, Place Show Command 출력	
DWS.UTC.2118.00	Button == 'B', Under Line == 1, Time == 2020.03.05 10:20:30	Time == 2020.03.05 10:20:31	<code>void increase_value()</code>

DWS.UTC.2118.01	Button == 'B', Under Line == 2, Time == 2020.03.05 10:20:30	Time == 2020.03.05 11:20:30	
DWS.UTC.2118.02	Button == 'B', Under Line == 3, Time == 2020.03.05 10:20:30	Time == 2020.03.05 10:21:30	
DWS.UTC.2118.03	Button == 'B', Under Line == 4, Time == 2020.03.05 10:20:30	Time == 2021.03.05 10:20:30	
DWS.UTC.2118.04	Button == 'B', Under Line == 5, Time == 2020.03.05 10:20:30	Time == 2020.04.05 10:20:30	
DWS.UTC.2118.05	Button == 'B', Under Line == 6, Time == 2020.03.05 10:20:30	Time == 2020.03.06 10:20:30	
DWS.UTC.2119.00	ST Time == 00:00:00 && ST Mode != 0	ST Time == 00:00:01	<code>void activate()</code>
DWS.UTC.2119.01	ST Time == 00:00:60 && ST Mode != 0	ST Time == 00:00:61	
DWS.UTC.2119.02	ST Mode == 0	ST Time == 00:00:00	
DWS.UTC.2119.03	ST Time == 00:00:99 && ST Mode != 0	ST Time == 00:01:00	
DWS.UTC.2119.04	ST Time == 00:59:99 && ST Mode != 0	ST Time == 01:00:00 && ST Mode != 0	
DWS.UTC.2119.05	StopWatch Mode && !LapTime	PASS	
DWS.UTC.211A.00	호출	PASS	<code>void stop()</code>
DWS.UTC.211B.00	ST Time == 12:34:56	Lap_시간 = 12:34:56	<code>void laptime()</code>
DWS.UTC.211B.01	Mode == StopWatch Mode	PASS	
DWS.UTC.211C.00	호출	ST Time == 00:00:00 Reset Show Command 출력	<code>void reset()</code>
DWS.UTC.211D.00	호출	PASS	<code>void LED_on()</code>
DWS.UTC.211E.00	호출	PASS	<code>void LED_off()</code>
DWS.UTC.2200.00	Show Command	PASS	<code>void Display_interface(int)</code>

			command)
DWS.UTC.2300.00	Back Light Command	PASS	시스템 함수