
Unit Testing Plan

이동아

SW 테스팅 수행

목표

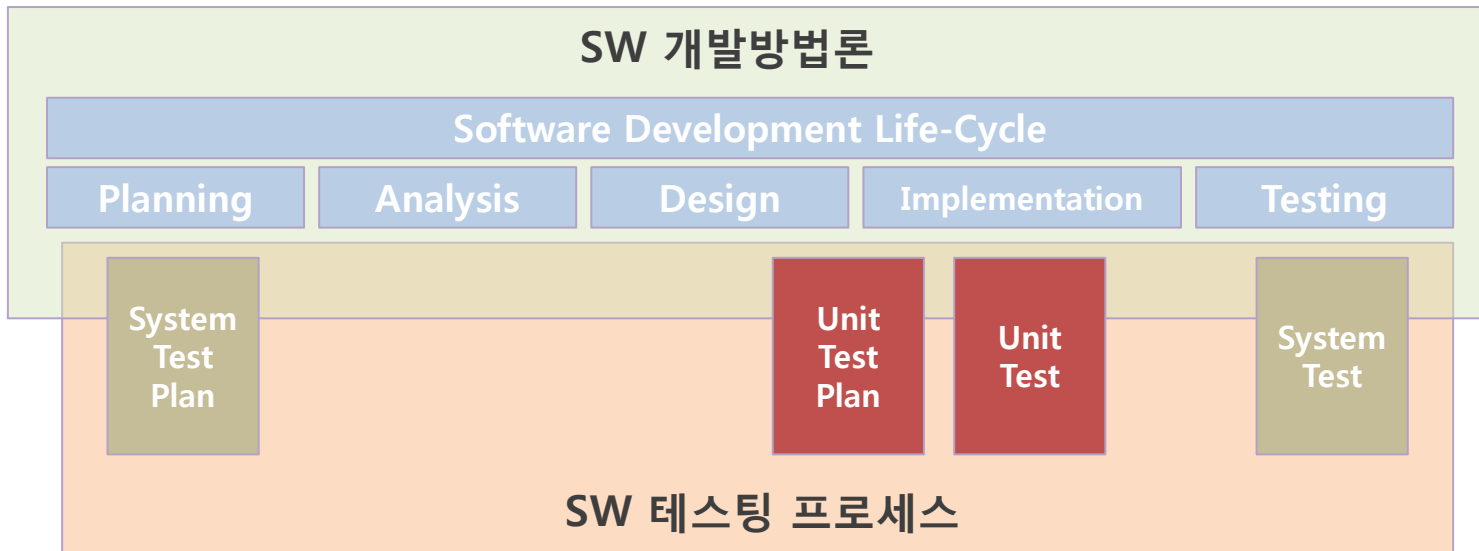
- SW 테스팅을 직접 수행하고 그 결과를 효과적으로 정리한다.
- 블랙박스 테스팅의 특징을 이해하고 효과적으로 적용한다.
- SW 테스팅에 필요한 자료들을 파악한다.

실습 내용

- DWS SW 테스트 수행
 - 시스템 테스트
 - 단위 테스트
- SW 테스팅 방법
 - 스펙 기반 기능 테스트 (블랙박스 테스트)
 - 기본적인 테스트 케이스 개발이론 적용

SW 테스트 & SW Development Life Cycle

- 'SW 테스트'은 'SW개발방법론' 및 'SDLC*'와 밀접하게 연계된다.
- "SW 테스트 프로세스"로서 개별적으로 고려 가능



*SDLC: Software Development Life Cycle

UTP (Unit Test Plan)

목표	• DWS UTP를 완성한다.
실습 내용	• DS.2012.DWS.UTP-0.7 을 ver. 1.1 로 완성한다. • SRS/SRA/SDS/C code를 참고 하여 Unit을 대상으로 블랙박스 테스트 케이스를 개발한다.



UTP Template



UTP 1.0

UTP

- 단위 테스트 계획서 (UTP: Unit Test Plan)
 - 단위 테스트를 위한 계획서
 - 테스트할 모듈의 개발자가 직접 작성
 - 일반적으로 생략됨
 - 단위 테스트의 범위 설정

- 단위 테스트 케이스 (입력값 + 출력값) 개발
 - Black/Gray-Box Testing 기법 사용 時,
 - SRS/SRA/SDS 내용을 기반으로
 - C Code (Interface 및 내부 로직 일부 등)를 참고해서 개발
 - White-Box Testing 기법 사용 時,
 - C Code로부터 CFG를 생성하고, Structural Coverage Criteria를 적용
 - Statement, Branch, Condition, MC/DC 커버리지
 - 자동으로 테스트 케이스를 생성하는 상용도구 多數
 - 출력값은 수동으로 만들어야 하는 경우가 대부분

UTP for DWS

Ver. DS.2012.DWS.UTP-1.1

Unit Testing Plan for Digital Watch System

- Test Plan
- Test Design Specification
- Test Cases Specification

Project Team
Dependable Software Laboratory

Date
2012-08-17

Team Information

Dong-Ah Lee	dlee.dslab@gmail.com
Sanghyun Yoon	shyoon.dslab@gmail.com
Jaeyeon Jo	entertm77@gmail.com
Jong-Hoon Lee	jhlee.dslab@gmail.com
Sun Hwi Lee	bigfarm@hanmail.net
Eui-Sub Kim	atang34@naver.com

[텍스트 입력]

Dependable Software Laboratory

1

Table of Contents

1. Introduction
 - 1.1 Objectives
 - 1.2 Background
 - 1.3 Scope
 - 1.4 Project plan
 - 1.5 Configuration management plan
 - 1.6 References
2. Test items
3. Features to be tested
4. Features not to be tested
5. Approach
6. Item pass/fail criteria
7. Unit test design specification
 - 7.1 Test design specification identifier
 - 7.2 Features to be tested
 - 7.3 Approach refinements
 - 7.4 Test identification
 - 7.5 Feature pass/fail criteria
8. Unit test case specification
 - 8.1 Test case specification identifier
 - 8.2 Test items
 - 8.3 Input specifications
 - 8.4 Output specifications
9. Testing tasks
10. Environmental needs
11. Unit test deliverables
12. Schedules

1 Introduction

1.1 Objectives

본 문서는 digital watch system의 unit test를 수행하기 위한 계획 문서이며 다음과 같은 목적을 갖는다.

- (1) Digital watch system의 unit test를 수행하기 위해 필요한 활동 및 자원을 정의한다.
- (2) Digital watch system의 unit test를 수행하기 위한 test approach 및 techniques을 정의한다.
- (3) Digital watch system의 unit test를 수행하기 위한 환경적인 요구사항 및 test 도구들을 정의한다.

1.2 Background

Digital watch system은 현재 시각을 알려주는 기본 기능 외에도 알람, 스톱워치 등의 다양한 기능을 가지고 있는 시계이다. 이 시스템은 여러 개의 입력을 가지고 있으며 실행 시간, 반응 시간 등도 시스템의 성능을 좌우하는 중요한 feature들이다. Unit test는 시스템을 구성하는 최소 단위 모듈들을 대상으로 하는 test이며, 시스템의 성능을 좌우하는 feature들이 요구사항을 만족하는지를 확인할 수 있는 기본적인 test approach이다.

1.3 Scope

이 계획 문서는 digital watch system의 unit test를 수행하기 위한 모든 것을 포함한다. Digital watch system의 unit test를 수행하기 위한 자원과 절차, test approach와 technique과 필요로 하는 환경 및 도구 등을 정의한다. Digital watch system의 unit test는 시스템을 구성하는 최소 단위의 모듈들을 대상으로 하며, 구현된 모듈이 요구사항을 만족하는지를 test한다.

1.4 Project plan

1.5 Configuration management plan

Digital watch system의 program source code 및 unit test를 위한 test code는 CTIP (Continuous Testing & Integration Platform) 환경에서 이루어지며, program source code/test code의 변경 및 수정 사항은 지속적으로 통합되고 test된다.

(1) Program source code의 변경

Program source code에 변경 및 수정 발생 시, 이를 통합하고 수동적으로 unit test를 수행한다.

(2) 일정 주기

CI server에 의해 관리되는 program source code는 일정 주기를 가지고 자동적으로 build 및 unit test를 수행한다.

1.6 References

Test plan은 다음 문서들을 기반으로 작성되었으며, unit test 수행을 위한 자원으로서 사용된다.

DS.2012.DWS.SRS-2.0

DS.2012.DWS.SRA-1.1

DS.2012.DWS.SDS-1.1

IEEE 829-1998 Standard for Software Test Documentation

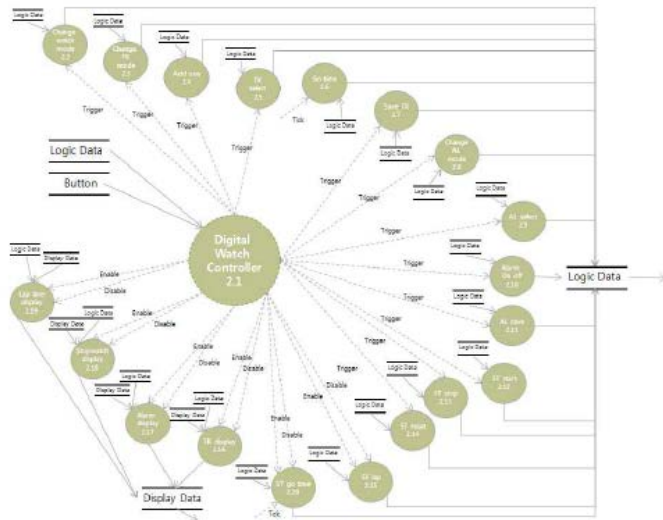
IEEE 1012-2012 Standard for System and Software Verification and Validation

2 Test items

Digital watch system을 구성하는 최소 단위의 모듈들이 unit test의 대상이 된다. 각 모듈의 요구사항을 만족하는지를 test하며, test item은 다음 자료들로부터 작성되었다.

(1) Functionality of modules - DS.2012.DWS.SRA-1.1: Process specification

아래 그림은 그 일부를 참조한 것임.



Reference No.	2.1
Name	Digital Watch Controller
Input	Button, Logic Data
Output	Enable, Disable, Trigger
Process Description	Button, Logic Data 를 받아서 상태를 통해 해당 프로세스에게 명령을 전달한다.

(2) Module interface - DS.2012.DWS.SRA-2.0: Process specification, Structure chart

3 Features to be tested

- (1) Processes in SRA: 각 프로세스가 가지고 있는 요구사항을 만족하는지를 test한다
- (2) Modules in SDS: 각 모듈이 가지고 있는 데이터 인터페이스를 test한다. <Table 1 테스트할 Process (DFD) 리스트>의 Process name 참조

Table 1 테스트할 Process (DFD) 리스트

ID	Name	Description
2.1	Digital Watch Controller	Button, Logic Data 를 받아서 상태를 통해 해당 프로세스에게 명령을 전달한다.
2.2	Change watch mode	2.1 Digital Watch Controller 가 모드를 변경할 때 호출하는 프로세스. Logic Data 내에 존재하는 display mode 를 변경한다.
2.3	Change TK mode	Trigger 명령을 받으면 TK modifying 을 전달받아 0 또는 1 로 변환하는 프로세스. Timekeeping mode 에서 Current time 을 수정 중인지 아닌지를 나타낸다.
2.4	Add one	시간을 수정할 때 1씩 증가시키는 프로세스
2.5	TK select	Timekeeping mode 에서 시간을 수정할 때, 연, 월, 일, 시, 분, 초 2.6 중 어느 것을 수정하고 있는지 표시하는 프로세스
2.6	Go time	Tick 을 입력 받아 현재 시간을 진행시키는 프로세스.
2.7	Save TK	수정된 현재시간을 저장하는 프로세스.
2.8	Change AL mode	Alarm mode 일 때, 수정 중인지 아닌지를 표현해주는 프로세스.
2.9	AI select	알람 시간을 수정할 때, 어느 부분을 수정 중인지 표현해주는 프로세스
2.10	Alarm On Off	알람 설정 여부를 표시할 때 2.1 Digital Watch Controller 가

		호출하는 프로세스.
2.11	AL save	수정한 알람 시간을 저장할 때 2.1 Digital Watch Controller 가 호출하는 프로세스.
2.12	ST start	Stopwatch mode 에서 현재 상태를 저장하는 프로세스. stOn 을 1로 변환.
2.13	ST stop	Stopwatch mode 에서 현재 상태를 저장하는 프로세스. stOn 을 0으로 변환.
2.14	ST reset	Elapsed time 을 00:00:00 으로 초기화하는 프로세스.
2.15	ST lap	Stopwatch mode 에서 현재 상태를 저장하는 프로세스. stOn 을 2로 변환.
2.20	ST go time	Stopwatch mode 에서 Tick 에 따라 Elapsed time 을 진행시키는 프로세스.
2.21	Alarm Controller	알람을 울려야 되는 상황에서 Enable, 아니라면 Disable 을 명령하는 프로세스.

4 Features not to be tested

- (1) Processes in SRA: 외부 장치 드라이버, 단순 데이터 전달 프로세스 등은 test에서 제외한다.
 (2) Modules in SDS: <Table 2 테스트하지 않을 Process (DFD) 리스트>의 Process name 참조

Table 2 테스트하지 않을 Process (DFD) 리스트

ID	Name	Description
1.1	Button a interface	버튼 'A' 입력을 받아서 Button 에 저장한다.
1.2	Button b interface	버튼 'B' 입력을 받아서 Button 에 저장한다.
1.3	Button c interface	버튼 'C' 입력을 받아서 Button 에 저장한다.
1.4	Button d interface	버튼 'D' 입력을 받아서 Button 에 저장한다.
2.16	TK display	Timekeeping mode 에서 출력을 위해 필요한 데이터를 전달하는 프로세스.
2.17	Alarm display	Alarm mode 에서 출력을 위해 필요한 데이터를 전달하는 프로세스.
2.18	Stopwatch display	Stopwatch mode 에서 출력을 위해 필요한 데이터를 전달하는 프로세스.
2.19	Lap time display	Stopwatch mode 에서 Lap time 출력을 위해 필요한 데이터를 전달하는 프로세스.
2.22	Light	Button 입력에 따라 Backlight signal 을 전달하는 프로세스.
2.23	Alarm indicator	Alarm on 을 통해 알람이 켜졌는지에 대한 여부를 조사하는 프로세스.
2.24	TK bar	현재 시간을 수정할 때 사용자에게 어느 부분을 수정 중인지 표시하기 위한 데이터를 전달하는 프로세스.
2.25	Alarm bar	알람 시간을 수정할 때 사용자에게 어느 부분을 수정

[텍스트 입력]

Dependable Software Laboratory

8

		중인지 표시하기 위한 데이터 전달 프로세스.
2.26	PM indicator	현재시간이 오후임을 알려주는 데이터를 전달하는 프로세스.
2.27	Sound out	2.21 Alarm Controller 의 입력을 통해 Sound signal 을 전달한다.
3.1	Display interface	화면출력과 관련된 모든 데이터를 전달받아 화면출력을 해주는 데이터를 전달해주는 프로세스.
3.2	Sound device interface	Sound signal 을 받아서 sound 를 전달해주는 프로세스.
3.3	Backlight interface	Backlight signal 을 전달받아 LED 가 켜고 꺼지는 정보를 전달해주는 프로세스.

5 Approach

Digital watch system의 program source code 및 unit test를 위한 test code는 CTIP (Continuous Testing & Integration Platform) 환경에서 이루어지며, program source code/test code 의 변경 및 수정 사항은 지속적으로 통합되고 test된다.

- (1) Brute force testing: 각 모듈의 요구사항을 만족하는지를 확인할 수 있는 test case를 작성한다. 그 이외의 예외상황에 대해서는 test하지 않는다.

6 Item pass/fail criteria

Functional test pass/fail criteria: 각 모듈은 요구사항을 모두 만족하여야 한다.

7 Unit test design specification

7.1 Test design specification identifier

DWS.UTD.00.00

7.2 Features to be tested

7.2.1 Processes in SRA

<Table 1 테스트할 Process (DFD) 리스트> 참조

7.3 Approach refinements

7.3.1 Brute force testing

DWS의 각 모듈이 요구사항을 만족하는지를 확인하기 위하여, 요구사항에 정의된 내용에 기반하여 test case를 작성한다. 그 이외의 예외 상황에 대해서는 test case를 작성하지 않는다.

[텍스트 입력]

Dependable Software Laboratory

9

7.4 Test identification

Table 3 Test Design Identification

Identifier	Feature (Process ID in DFD)	Valid/ Invalid value
DWS_UTC_000_000	2.1 Digital Watch Controller	Time keeping 상태에서 [a]입력이 들어온다.
DWS_UTC_000_001	2.1 Digital Watch Controller	Time keeping 상태에서 [b]입력이 들어온다.
DWS_UTC_000_002	2.1 Digital Watch Controller	Time keeping 상태에서 [c]입력이 들어온다.
DWS_UTC_000_003	2.1 Digital Watch Controller	Time keeping 상태에서 [d]입력이 들어온다.
DWS_UTC_000_004	2.1 Digital Watch Controller	Change TK mode 상태에서 [a]입력이 들어온다.
DWS_UTC_000_005	2.1 Digital Watch Controller	Change TK mode 상태에서 [b]입력이 들어온다.
DWS_UTC_000_006	2.1 Digital Watch Controller	Change TK mode 상태에서 [c]입력이 들어온다.
DWS_UTC_000_007	2.1 Digital Watch Controller	Change TK mode 상태에서 [d]입력이 들어온다.
DWS_UTC_000_008	2.1 Digital Watch Controller	Increase TK value 상태에서 [a]입력이 들어온다.
DWS_UTC_000_009	2.1 Digital Watch Controller	Increase TK value 상태에서 [b]입력이 들어온다.
DWS_UTC_000_010	2.1 Digital Watch Controller	Increase TK value 상태에서 [c]입력이 들어온다.
DWS_UTC_000_011	2.1 Digital Watch Controller	Increase TK value 상태에서 [d]입력이 들어온다.
DWS_UTC_000_012	2.1 Digital Watch Controller	Stopwatch on 상태에서 [a]입력이 들어온다.
DWS_UTC_000_013	2.1 Digital Watch Controller	Stopwatch on 상태에서 [b]입력이 들어온다.
DWS_UTC_000_014	2.1 Digital Watch Controller	Stopwatch on 상태에서 [c]입력이 들어온다.
DWS_UTC_000_015	2.1 Digital Watch Controller	Stopwatch on 상태에서 [d]입력이 들어온다.
DWS_UTC_000_016	2.1 Digital Watch Controller	Stopwatch off 상태에서 [a]입력이 들어온다.
DWS_UTC_000_017	2.1 Digital Watch Controller	Stopwatch off 상태에서 [b]입력이 들어온다.
DWS_UTC_000_018	2.1 Digital Watch Controller	Stopwatch off 상태에서 [c]입력이 들어온다.
DWS_UTC_000_019	2.1 Digital Watch Controller	Stopwatch off 상태에서 [d]입력이 들어온다.
DWS_UTC_000_020	2.1 Digital Watch Controller	Lap time 상태에서 [a]입력이 들어온다.
DWS_UTC_000_021	2.1 Digital Watch Controller	Lap time 상태에서 [b]입력이 들어온다.
DWS_UTC_000_022	2.1 Digital Watch Controller	Lap time 상태에서 [c]입력이 들어온다.
DWS_UTC_000_023	2.1 Digital Watch Controller	Lap time 상태에서 [d]입력이 들어온다.
DWS_UTC_000_024	2.1 Digital Watch Controller	Alarm 상태/stOn==0에서 [a]입력이 들어온다.
DWS_UTC_000_025	2.1 Digital Watch Controller	Alarm 상태/stOn==0에서 [b]입력이 들어온다.
DWS_UTC_000_026	2.1 Digital Watch Controller	Alarm 상태/stOn==0에서 [c]입력이 들어온다.
DWS_UTC_000_027	2.1 Digital Watch Controller	Alarm 상태/stOn==0에서 [d]입력이 들어온다.
DWS_UTC_000_028	2.1 Digital Watch Controller	Alarm 상태/stOn==1에서 [a]입력이 들어온다.
DWS_UTC_000_029	2.1 Digital Watch Controller	Alarm 상태/stOn==1에서 [b]입력이 들어온다.
DWS_UTC_000_030	2.1 Digital Watch Controller	Alarm 상태/stOn==1에서 [c]입력이 들어온다.
DWS_UTC_000_031	2.1 Digital Watch Controller	Alarm 상태/stOn==1에서 [d]입력이 들어온다.
DWS_UTC_000_032	2.1 Digital Watch Controller	Alarm 상태/stOn==2에서 [a]입력이 들어온다.
DWS_UTC_000_033	2.1 Digital Watch Controller	Alarm 상태/stOn==2에서 [b]입력이 들어온다.
DWS_UTC_000_034	2.1 Digital Watch Controller	Alarm 상태/stOn==2에서 [c]입력이 들어온다.
DWS_UTC_000_035	2.1 Digital Watch Controller	Alarm 상태/stOn==2에서 [d]입력이 들어온다.
DWS_UTC_000_036	2.1 Digital Watch Controller	Change Alarm mode상태에서 [a]입력이 들어온다.
DWS_UTC_000_037	2.1 Digital Watch Controller	Change Alarm mode상태에서 [b]입력이 들어온다.

[텍스트 입력]

Dependable Software Laboratory

10

DWS_UTC_000_038	2.1 Digital Watch Controller	Change Alarm mode상태에서 [c]입력이 들어온다.
DWS_UTC_000_039	2.1 Digital Watch Controller	Change Alarm mode상태에서 [d]입력이 들어온다.
DWS_UTC_000_040	2.1 Digital Watch Controller	Increase Alarm value상태에서 [a]입력이 들어온다.
DWS_UTC_000_041	2.1 Digital Watch Controller	Increase Alarm value상태에서 [b]입력이 들어온다.
DWS_UTC_000_042	2.1 Digital Watch Controller	Increase Alarm value상태에서 [c]입력이 들어온다.
DWS_UTC_000_043	2.1 Digital Watch Controller	Increase Alarm value상태에서 [d]입력이 들어온다.
DWS_UTC_001_000	2.2 Change watch mode	Trigger입력이 들어오면, Display mode 값을 변경한다.
DWS_UTC_002_000	2.3 Change TK mode	Trigger 입력이 들어오면, TK modifying 값을 변경한다.
DWS_UTC_003_000	2.4 Add one	Trigger 입력이 들어오면, 현재 수정 중인 current time / set time 의 값을 변경한다.
DWS_UTC_004_000	2.5 TK select	Trigger 입력이 들어오면, TK set mode 를 변경한다.
DWS_UTC_005_000	2.6 Go time	Tick 입력이 들어오면, current time 을 증가시킨다.
DWS_UTC_006_000	2.7 Save TK	Trigger 입력이 들어오면, 수정된 current time 을 저장한다.
DWS_UTC_007_000	2.8 Change AL mode	Trigger 입력이 들어오면, AL modifying 값을 변경한다.
DWS_UTC_008_000	2.9 AL select	Trigger 입력이 들어오면, Alarm set mode 값을 변경한다.
DWS_UTC_009_000	2.10 Alarm On Off	Trigger 입력이 들어오면, Alarm on 값을 변경한다.
DWS_UTC_010_000	2.11 AL save	Trigger 입력이 들어오면, 수정된 set time 값을 저장한다.
DWS_UTC_011_000	2.12 ST start	Trigger 입력이 들어오면, stOn 값을 1로 변경한다.
DWS_UTC_012_000	2.13 ST stop	Trigger 입력이 들어오면, stOn 값을 0으로 변경한다.
DWS_UTC_013_000	2.14 ST reset	Trigger 입력이 들어오면, Elapsed time 을 00:00:00으로 초기화한다.
DWS_UTC_014_000	2.15 ST lap	Trigger 입력이 들어오면, stOn 값을 2로 변경한다.
DWS_UTC_015_000	2.20 ST go time	Enable입력과 Tick입력이 들어오면, Elapsed time 값을 증가시킨다.
DWS_UTC_015_001	2.20 ST go time	Disable입력과 Tick입력이 들어오면, Elapsed time 값 변경 없음.
DWS_UTC_016_000	2.21 Alarm Controller	Idle 상태에서, Alarm on == true && Current time == set time
DWS_UTC_016_001	2.21 Alarm Controller	Alarm rings 상태에서,

[텍스트 입력]

Dependable Software Laboratory

11

		(Current time - set time == 20) (Button != 0 && Button < 5)
--	--	--

7.5 Feature pass/fail criteria

DWS의 각 모듈(프로세스)은 SRA에 정의되어 있는 요구사항 (입력 / 출력 및 동작) 을 모두 만족해야 한다. 각 모듈(프로세스)의 입력 / 출력 및 동작은 SRA의 process description 항목을 참조한다.

8 Unit test case specification

8.1 Test case specification identifier

Table 4 Test Case Identification

Test case identifier	Input specification	Output specification
DWS_UTC_000_000	controlState==Time keeping / a	ControlState==Change TK mode
DWS_UTC_000_001	State==Time keeping / b	State==Time keeping
DWS_UTC_000_002	State==Time keeping / c	State==Alarm
DWS_UTC_000_003	State==Time keeping / d	State==Time keeping
DWS_UTC_000_004	State==Change TK mode / a	State==Time keeping
DWS_UTC_000_005	State==Change TK mode / b	State==Increase TK value
DWS_UTC_000_006	State==Change TK mode / c	State==Change TK mode
DWS_UTC_000_007	State==Change TK mode / d	State== Change TK mode
DWS_UTC_000_008	State==Increase TK value / a	State==Time keeping
DWS_UTC_000_009	State==Increase TK value / b	State==Increase TK value
DWS_UTC_000_010	State==Increase TK value / c	State==Change TK mode

실습!!!

DWS_UTC_000_021	State==Lap time / b	State==Lap time
DWS_UTC_000_022	State==Lap time / c	State==Time keeping
DWS_UTC_000_023	State==Lap time / d	State==Lap time
DWS_UTC_000_024	State==Alarm / stOn==0 / a	State==Change Alarm mode
DWS_UTC_000_025	State==Alarm / stOn==0 / b	State==Alarm
DWS_UTC_000_026	State==Alarm / stOn==0 / c	State==Stopwatch off

DWS_UTC_000_027	State==Alarm / stOn==0 / d	State==Alarm
DWS_UTC_000_028	State==Alarm / stOn==1 / a	State==Change Alarm mode
DWS_UTC_000_029	State==Alarm / stOn==1 / b	State==Alarm
DWS_UTC_000_030	State==Alarm / stOn==1 / c	State==Stopwatch on
DWS_UTC_000_031	State==Alarm / stOn==1 / d	State==Alarm
DWS_UTC_000_032	State==Alarm / stOn==2 / a	State==Change Alarm mode
DWS_UTC_000_033	State==Alarm / stOn==2 / b	State==Alarm
DWS_UTC_000_034	State==Alarm / stOn==2 / c	State==Lap time
DWS_UTC_000_035	State==Alarm / stOn==2 / d	State==Alarm
DWS_UTC_000_036	State==Change Alarm mode / a	State==Alarm
DWS_UTC_000_037	State==Change Alarm mode / b	State==Increase Alarm value
DWS_UTC_000_038	State==Change Alarm mode / c	State==Change Alarm mode
DWS_UTC_000_039	State==Change Alarm mode / d	State==Change Alarm mode

실습!!!

DWS_UTC_013_000	Trigger in / e_time.tm_sec==30	e_time.tm_sec==0
DWS_UTC_014_000	Enable, Disable / stOn==30	stOn==2
DWS_UTC_015_000	Enable, tick	
DWS_UTC_015_001	Disable, tick	
DWS_UTC_016_000	State==Idle, Alarm on==true, current time==set time / alarm_state==0	Enable / alarm_state==1
DWS_UTC_016_001	State==alarm rings, (Current time - set time == 20) (Button != 0 && Button < 5) / alarm_state == 1	Disable / alarm_state==0

8.2 Test items

<Table 3 Test Design Identification> 참조

8.3 Input specifications

<Table 4 Test Case Identification> 참조

8.4 Output specifications

<Table 4 Test Case Identification> 참조

9 Testing tasks

Table 5 Testing tasks & Schedule

Task	Predecessor tasks	Special skills	Effort	Finish date
(1) Unit Test Plan 작성	DS-DWS-2012-SRS 작성 DS-DWS-2012-SRA 작성 DS-DWS-2012-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	

10 Environmental needs

Digital watch system의 unit test를 위한 환경적 요구사항은 다음과 같다.

(1) Hardware & Platform, Eclipse IDE (Integrated Development Environment)

gcc compiler/linker

(2) CTIP (Continuous Testing & Integration Platform) Environment

CI server

SVN repository server

CI/SVN server에 접근 가능한 단말 PC

Test tools

CUnit unit test framework for C

gCov code coverage measurement tool

11 Unit Test deliverables

[텍스트 입력]

Dependable Software Laboratory

14

11.1 Unit test plan

11.2 Unit test design specification

11.3 Unit test case specification

11.4 Unit test summary report

12 Schedules

<Table 5 Testing tasks & Schedule> 참조

[텍스트 입력]

Dependable Software Laboratory

15

실습시작!!!

정리 및 확인

1

단위 테스트의 수행 주체는 누구인가?

2

어떤 기법을 사용하여 Unit Test Case를 개발하였는가?

3

SASD에서 단위 테스트를 위한 정보는 어디서·어떻게 구하는가?

4

단위 테스트 수행(Execution)은 어떻게 진행되는가?

5

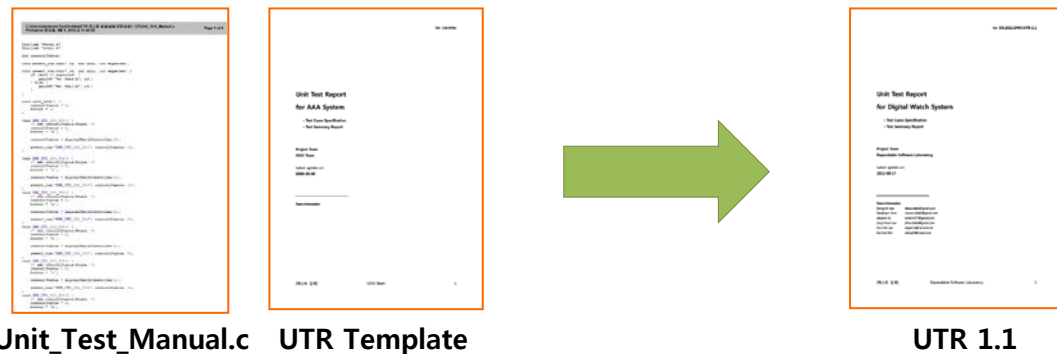
화이트 박스 기법을 이용할 경우, Unit Test Case는 어떻게 개발하는가?

실습 7.

UT (UNIT TEST)

실습 7 – UT (Unit Test)

목표	<ul style="list-style-type: none">• DWS UTP에 따라 단위 테스트를 수행하고 DWS UTR을 완성한다.• 단위 테스트를 위해 필요한 준비사항들을 이해한다.
실습 내용	<ul style="list-style-type: none">• UTR template 확인• DS.2012.DWS.UTR-0.7 을 ver. 1.1 으로 완성한다.• DWS code를 대상으로 단위 테스트를 수행한다.• DWS UT에 필요한 환경/Driver/Stub을 개발한다.

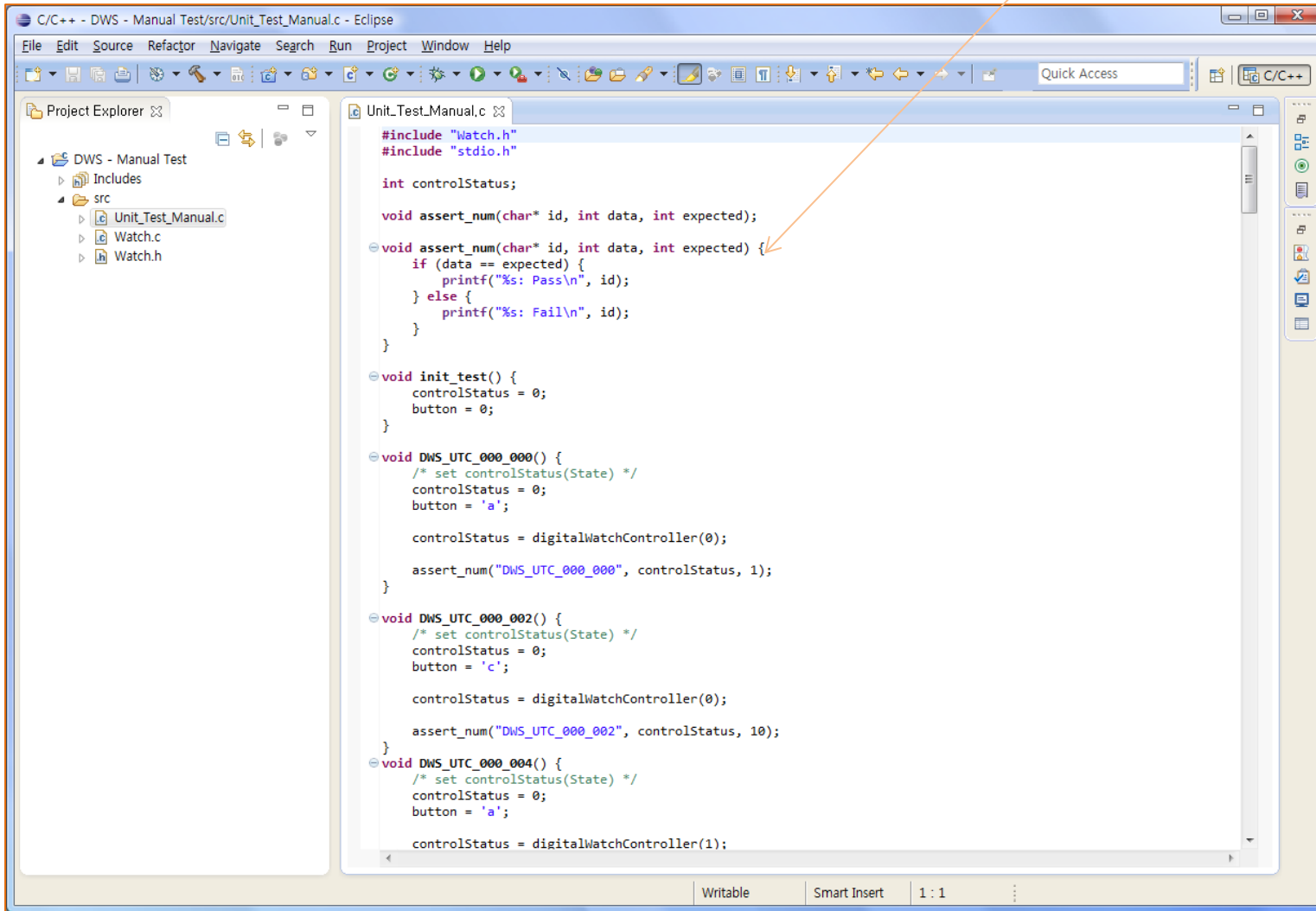


Unit Test

- 기본단위(component , module , function , method)에 대한 테스트
 - 개발자가 단위 모듈 개발 중에 실시하는 테스트
 - CUnit , JUnit 등의 테스트 자동화 지원도구 사용 가능
- 테스트 케이스 개발
 - Black/Gray-Box 테스트 : 문서 및 코드 일부를 참고하여 개발
 - White-Box 테스트 : 코드 구조를 분석하는 테스트 케이스 자동생성 도구 사용
- 테스트 실행
 - 테스트 지원 코드 (Driver 와 Stub) 구현 필요
 - 테스트 대상 코드에 대한 일부 수정 필요

테스트 지원 코드 작성

Testing을 위한 assertion code



```
C/C++ - DWS - Manual Test/src/Unit_Test_Manual.c - Eclipse
File Edit Source Refactor Navigate Search Run Project Window Help
Project Explorer
DWS - Manual Test
  Includes
  src
    Unit_Test_Manual.c
    Watch.c
    Watch.h
Unit_Test_Manual.c
#include "watch.h"
#include "stdio.h"

int controlStatus;

void assert_num(char* id, int data, int expected);
void assert_num(char* id, int data, int expected) {
    if (data == expected) {
        printf("%s: Pass\n", id);
    } else {
        printf("%s: Fail\n", id);
    }
}

void init_test() {
    controlStatus = 0;
    button = 0;
}

void DWS_UTC_000_000() {
    /* set controlStatus(State) */
    controlStatus = 0;
    button = 'a';

    controlStatus = digitalWatchController(0);

    assert_num("DWS_UTC_000_000", controlStatus, 1);
}

void DWS_UTC_000_002() {
    /* set controlStatus(State) */
    controlStatus = 0;
    button = 'c';

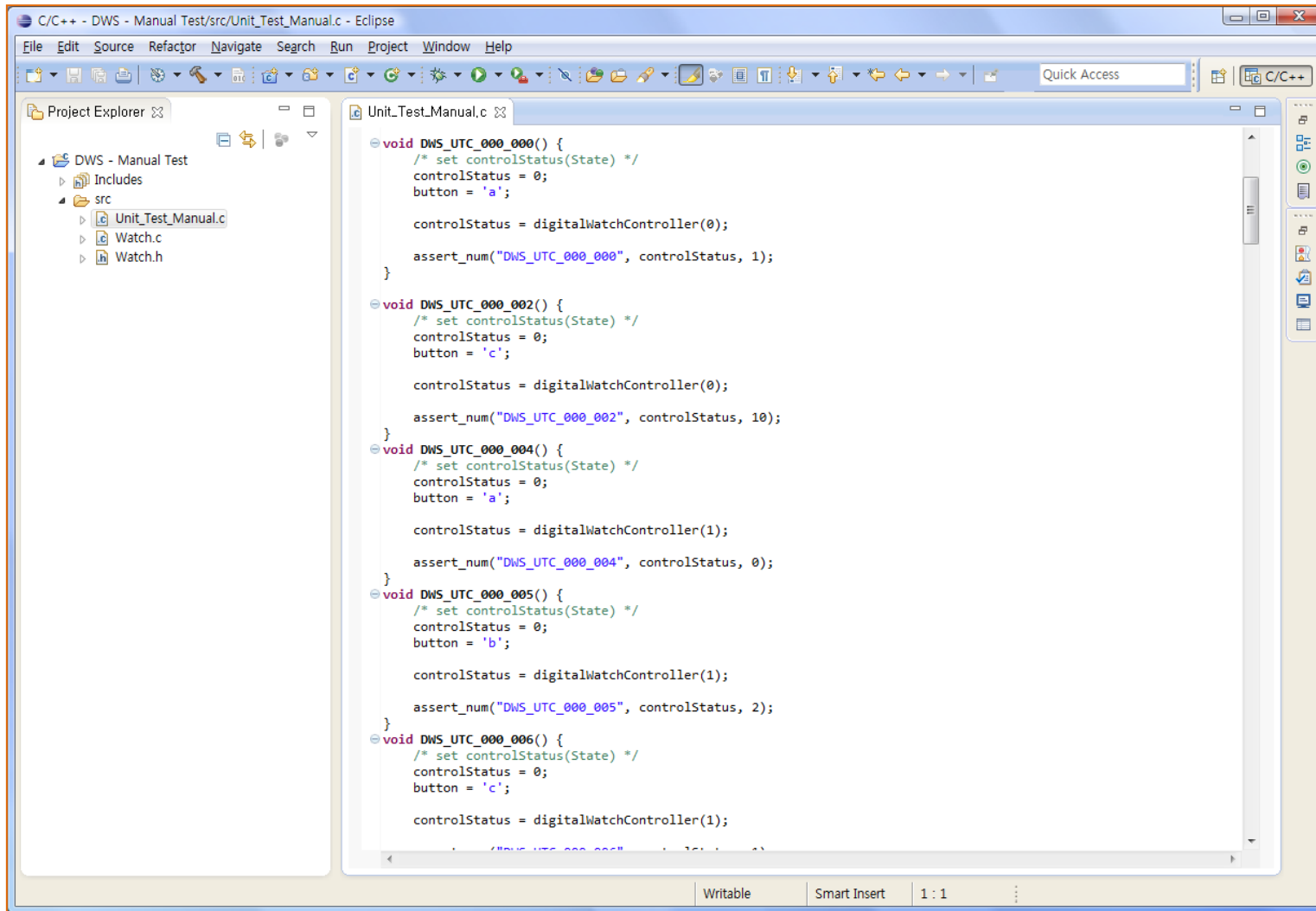
    controlStatus = digitalWatchController(0);

    assert_num("DWS_UTC_000_002", controlStatus, 10);
}

void DWS_UTC_000_004() {
    /* set controlStatus(State) */
    controlStatus = 0;
    button = 'a';

    controlStatus = digitalWatchController(1);
}
Writable Smart Insert 1 : 1
```

테스트 케이스 작성



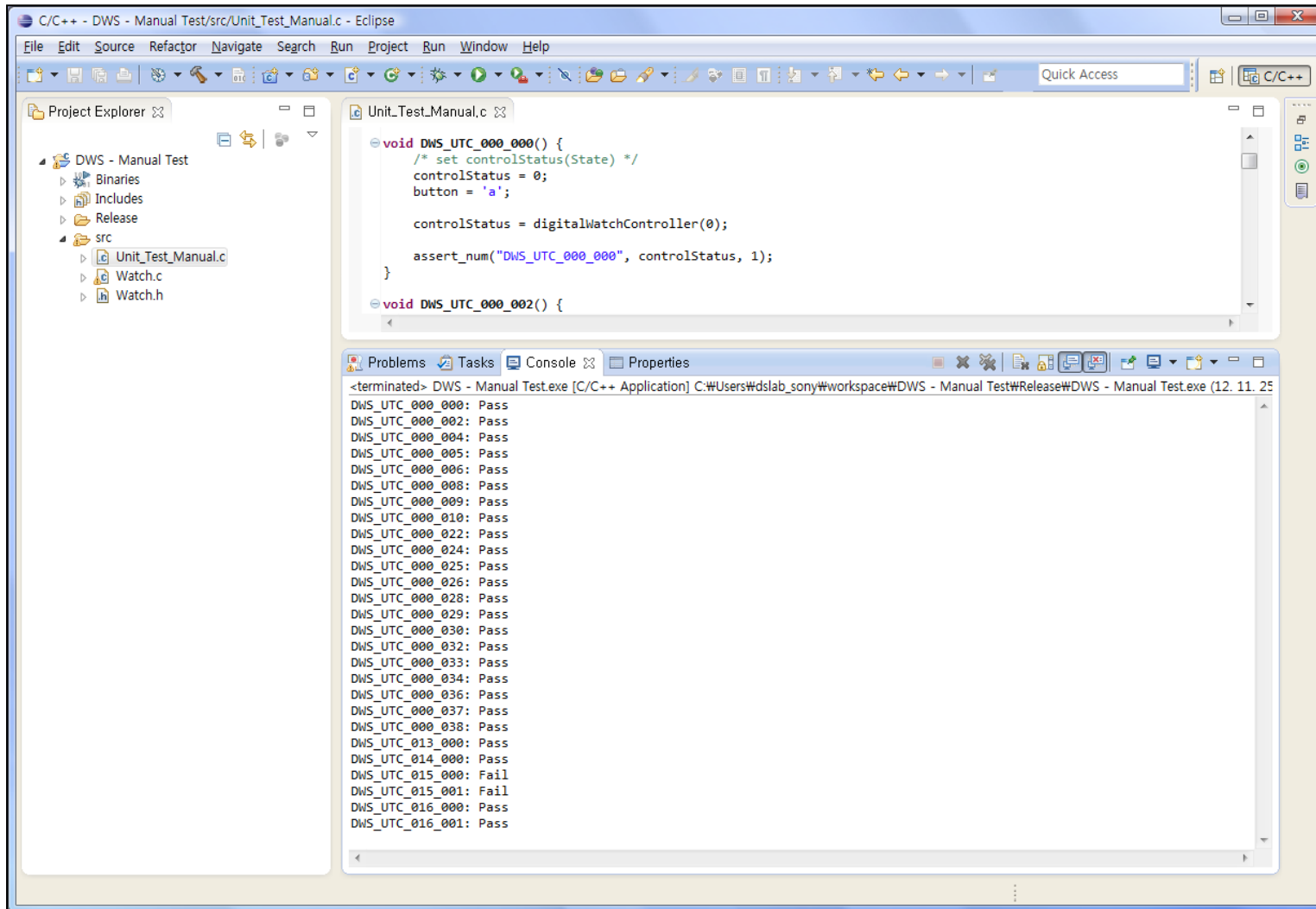
The screenshot shows the Eclipse IDE interface with the following details:

- Title Bar:** C/C++ - DWS - Manual Test/src/Unit_Test_Manual.c - Eclipse
- Menu Bar:** File, Edit, Source, Refactor, Navigate, Search, Run, Project, Window, Help
- Project Explorer (Left):** Shows a project named "DWS - Manual Test" with subfolders "Includes", "src", and "Unit_Test_Manual.c".
- Editor (Center):** Displays the source code for "Unit_Test_Manual.c". The code contains several test functions:

```
void DWS_UTC_000_000() {  
    /* set controlStatus(State) */  
    controlStatus = 0;  
    button = 'a';  
  
    controlStatus = digitalWatchController(0);  
  
    assert_num("DWS_UTC_000_000", controlStatus, 1);  
}  
  
void DWS_UTC_000_002() {  
    /* set controlStatus(State) */  
    controlStatus = 0;  
    button = 'c';  
  
    controlStatus = digitalWatchController(0);  
  
    assert_num("DWS_UTC_000_002", controlStatus, 10);  
}  
  
void DWS_UTC_000_004() {  
    /* set controlStatus(State) */  
    controlStatus = 0;  
    button = 'a';  
  
    controlStatus = digitalWatchController(1);  
  
    assert_num("DWS_UTC_000_004", controlStatus, 0);  
}  
  
void DWS_UTC_000_005() {  
    /* set controlStatus(State) */  
    controlStatus = 0;  
    button = 'b';  
  
    controlStatus = digitalWatchController(1);  
  
    assert_num("DWS_UTC_000_005", controlStatus, 2);  
}  
  
void DWS_UTC_000_006() {  
    /* set controlStatus(State) */  
    controlStatus = 0;  
    button = 'c';  
  
    controlStatus = digitalWatchController(1);  
  
    assert_num("DWS_UTC_000_006", controlStatus, 1);  
}
```
- Status Bar (Bottom):** Shows "Writable", "Smart Insert", and "1 : 1".

```
void DWS.UTC_000_042( ) {  
  
    /* set controlStatus(State) */  
    controlStatus = 0;  
    button = 'c';  
  
    controlStatus = digitalWatchController(12);  
  
    assert_num("DWS.UTC_000_042", controlStatus, 11);  
}
```

테스트 실행 결과



The screenshot shows the Eclipse IDE interface. The top menu bar includes File, Edit, Source, Refactor, Navigate, Search, Run, Project, Run, Window, and Help. The Project Explorer on the left shows a project named 'DWS - Manual Test' with subfolders for Binaries, Includes, Release, and src. The src folder contains Unit_Test_Manual.c, Watch.c, and Watch.h. The main editor window displays the code for Unit_Test_Manual.c, which includes two test functions: DWS_UTC_000_000() and DWS_UTC_000_002(). The console window at the bottom shows the output of the test execution, listing the results for each test case.

```
void DWS_UTC_000_000() {  
    /* set controlStatus(State) */  
    controlStatus = 0;  
    button = 'a';  
  
    controlStatus = digitalWatchController(0);  
  
    assert_num("DWS_UTC_000_000", controlStatus, 1);  
}  
  
void DWS_UTC_000_002() {
```

```
<terminated> DWS - Manual Test.exe [C/C++ Application] C:\Users#dslab_sony#workspace#DWS - Manual Test#Release#DWS - Manual Test.exe (12. 11. 25  
DWS_UTC_000_000: Pass  
DWS_UTC_000_002: Pass  
DWS_UTC_000_004: Pass  
DWS_UTC_000_005: Pass  
DWS_UTC_000_006: Pass  
DWS_UTC_000_006: Pass  
DWS_UTC_000_008: Pass  
DWS_UTC_000_009: Pass  
DWS_UTC_000_010: Pass  
DWS_UTC_000_022: Pass  
DWS_UTC_000_024: Pass  
DWS_UTC_000_025: Pass  
DWS_UTC_000_026: Pass  
DWS_UTC_000_028: Pass  
DWS_UTC_000_029: Pass  
DWS_UTC_000_030: Pass  
DWS_UTC_000_032: Pass  
DWS_UTC_000_033: Pass  
DWS_UTC_000_034: Pass  
DWS_UTC_000_036: Pass  
DWS_UTC_000_037: Pass  
DWS_UTC_000_038: Pass  
DWS_UTC_013_000: Pass  
DWS_UTC_014_000: Pass  
DWS_UTC_015_000: Fail  
DWS_UTC_015_001: Fail  
DWS_UTC_016_000: Pass  
DWS_UTC_016_001: Pass
```

테스트 대상 코드 수정

```
#include <stdio.h>

void example_func1(int data1, int data2)
{
    int data3, data4;

    data3 = data1 * data2 + 3;
    data4 = data3 * 3;

    printf("data3 is %d\n", data3);
    printf("data4 is %d\n", data4);
}
```

```
#include <stdio.h>

int data3;

int example_func1(int data1, int data2)
{
    int data4;

    data3 = data1 * data2 + 3;
    data4 = data3 * 3;

    printf("data3 is %d\n", data3);
    printf("data4 is %d\n", data4);

    return data4;
}
```

확인하고자 하는 변수 data3, data4를 각각 전역 변수 / 반환 값으로 수정하여 함수 외부에서 확인 가능하게 수정

```

void mainwatch() {

    int controlStatus = 0,
        alarm_state = 0;
    int temp = 0;
    init();

    struct timeval tv;
    struct alarm_data al;
    struct back_data back;
    int old_time = 0;
    new_time = 0;

    ...

}

```

Watch.c



```

#include "Watch.h"
#include "stdio.h"

int controlStatus;

void init_test() {
    controlStatus = 0;
    button = 0;
}

...

int main() {
    init_test();

    DWS.UTC_000_000();
    DWS.UTC_000_002();

    ...

}

```

Unit_Test_Manual.c

UTR (Unit Test Report)

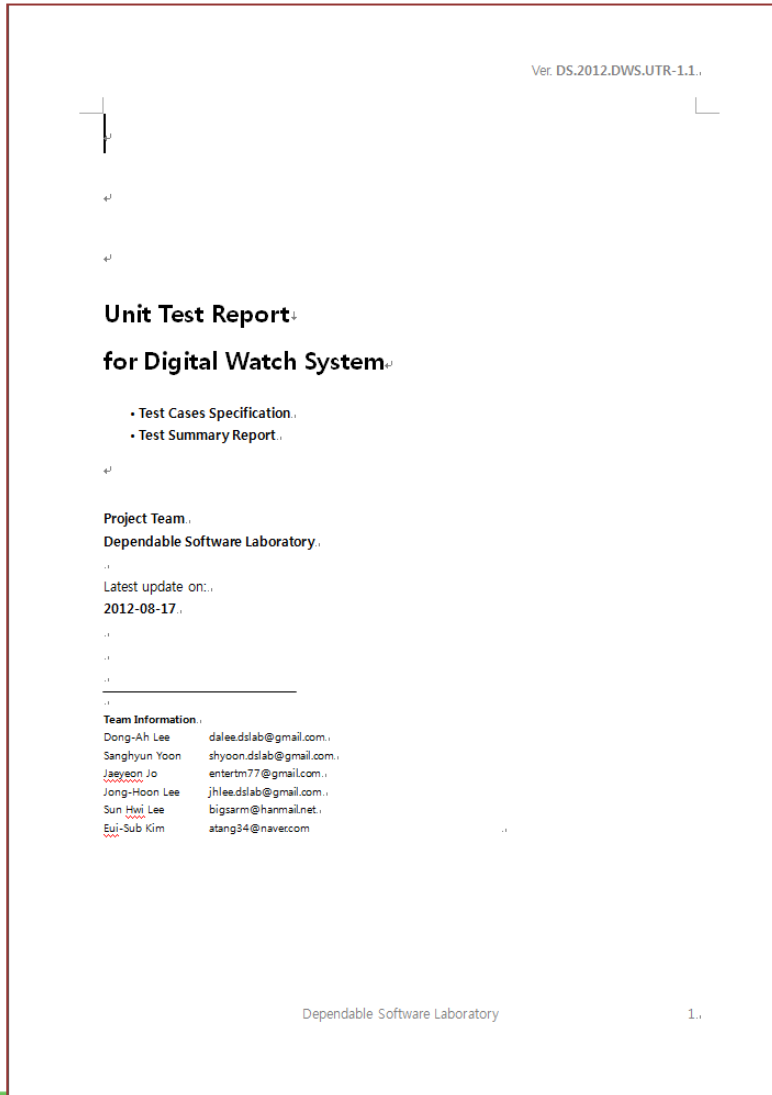


Table of Contents

1. Introduction
 - 1.1 Objectives
 - 1.2 References
2. Unit test case specification
 - 2.1 Test case specification identifier
 - 2.2 Test items
 - 2.3 Input specifications
 - 2.4 Output specifications
3. Environmental needs
4. Unit test summary report
 - 4.1 Test summary report identifier
 - 4.2 Evaluation

1 Introduction

1.1 Objectives

본 문서는 digital watch system의 unit test를 수행한 결과에 대한 report 문서이다. Test 요소들에 대한 test case와 test수행 결과에 대한 내용을 담고 있다.

1.2 References

DS.2012.DWS.SRS-2.0

DS.2012.DWS.SRA-1.1

DS.2012.DWS.SDS-1.1

DS.2012.DWS.UTP-1.1

2 Unit test case specification

2.1 Test case specification identifier

Table 1 Test Design Identification

Identifier	Feature (Process ID in DFD)	Valid/ Invalid value
DWS_UTC_000_000	2.1 Digital Watch Controller	Time keeping 상태에서 [a]입력이 들어온다.
DWS_UTC_000_001	2.1 Digital Watch Controller	Time keeping 상태에서 [b]입력이 들어온다.
DWS_UTC_000_002	2.1 Digital Watch Controller	Time keeping 상태에서 [c]입력이 들어온다.
DWS_UTC_000_003	2.1 Digital Watch Controller	Time keeping 상태에서 [d]입력이 들어온다.
DWS_UTC_000_004	2.1 Digital Watch Controller	Change TK mode 상태에서 [a]입력이 들어온다.
DWS_UTC_000_005	2.1 Digital Watch Controller	Change TK mode 상태에서 [b]입력이 들어온다.
DWS_UTC_000_006	2.1 Digital Watch Controller	Change TK mode 상태에서 [c]입력이 들어온다.
DWS_UTC_000_007	2.1 Digital Watch Controller	Change TK mode 상태에서 [d]입력이 들어온다.
DWS_UTC_000_008	2.1 Digital Watch Controller	Increase TK value 상태에서 [a]입력이 들어온다.
DWS_UTC_000_009	2.1 Digital Watch Controller	Increase TK value 상태에서 [b]입력이 들어온다.
DWS_UTC_000_010	2.1 Digital Watch Controller	Increase TK value 상태에서 [c]입력이 들어온다.
DWS_UTC_000_011	2.1 Digital Watch Controller	Increase TK value 상태에서 [d]입력이 들어온다.
DWS_UTC_000_012	2.1 Digital Watch Controller	Stopwatch on 상태에서 [a]입력이 들어온다.
DWS_UTC_000_013	2.1 Digital Watch Controller	Stopwatch on 상태에서 [b]입력이 들어온다.
DWS_UTC_000_014	2.1 Digital Watch Controller	Stopwatch on 상태에서 [c]입력이 들어온다.
DWS_UTC_000_015	2.1 Digital Watch Controller	Stopwatch on 상태에서 [d]입력이 들어온다.
DWS_UTC_000_016	2.1 Digital Watch Controller	Stopwatch off 상태에서 [a]입력이 들어온다.
DWS_UTC_000_017	2.1 Digital Watch Controller	Stopwatch off 상태에서 [b]입력이 들어온다.
DWS_UTC_000_018	2.1 Digital Watch Controller	Stopwatch off 상태에서 [c]입력이 들어온다.
DWS_UTC_000_019	2.1 Digital Watch Controller	Stopwatch off 상태에서 [d]입력이 들어온다.

DWS_UTC_000_020	2.1 Digital Watch Controller	Lap time 상태에서 [a]입력이 들어온다.
DWS_UTC_000_021	2.1 Digital Watch Controller	Lap time 상태에서 [b]입력이 들어온다.
DWS_UTC_000_022	2.1 Digital Watch Controller	Lap time 상태에서 [c]입력이 들어온다.
DWS_UTC_000_023	2.1 Digital Watch Controller	Lap time 상태에서 [d]입력이 들어온다.
DWS_UTC_000_024	2.1 Digital Watch Controller	Alarm 상태/stOn==0에서 [a]입력이 들어온다.
DWS_UTC_000_025	2.1 Digital Watch Controller	Alarm 상태/stOn==0에서 [b]입력이 들어온다.
DWS_UTC_000_026	2.1 Digital Watch Controller	Alarm 상태/stOn==0에서 [c]입력이 들어온다.
DWS_UTC_000_027	2.1 Digital Watch Controller	Alarm 상태/stOn==0에서 [d]입력이 들어온다.
DWS_UTC_000_028	2.1 Digital Watch Controller	Alarm 상태/stOn==1에서 [a]입력이 들어온다.
DWS_UTC_000_029	2.1 Digital Watch Controller	Alarm 상태/stOn==1에서 [b]입력이 들어온다.
DWS_UTC_000_030	2.1 Digital Watch Controller	Alarm 상태/stOn==1에서 [c]입력이 들어온다.
DWS_UTC_000_031	2.1 Digital Watch Controller	Alarm 상태/stOn==1에서 [d]입력이 들어온다.
DWS_UTC_000_032	2.1 Digital Watch Controller	Alarm 상태/stOn==2에서 [a]입력이 들어온다.
DWS_UTC_000_033	2.1 Digital Watch Controller	Alarm 상태/stOn==2에서 [b]입력이 들어온다.
DWS_UTC_000_034	2.1 Digital Watch Controller	Alarm 상태/stOn==2에서 [c]입력이 들어온다.
DWS_UTC_000_035	2.1 Digital Watch Controller	Alarm 상태/stOn==2에서 [d]입력이 들어온다.
DWS_UTC_000_036	2.1 Digital Watch Controller	Change Alarm mode상태에서 [a]입력이 들어온다.
DWS_UTC_000_037	2.1 Digital Watch Controller	Change Alarm mode상태에서 [b]입력이 들어온다.
DWS_UTC_000_038	2.1 Digital Watch Controller	Change Alarm mode상태에서 [c]입력이 들어온다.
DWS_UTC_000_039	2.1 Digital Watch Controller	Change Alarm mode상태에서 [d]입력이 들어온다.
DWS_UTC_000_040	2.1 Digital Watch Controller	Increase Alarm value상태에서 [a]입력이 들어온다.
DWS_UTC_000_041	2.1 Digital Watch Controller	Increase Alarm value상태에서 [b]입력이 들어온다.
DWS_UTC_000_042	2.1 Digital Watch Controller	Increase Alarm value상태에서 [c]입력이 들어온다.
DWS_UTC_000_043	2.1 Digital Watch Controller	Increase Alarm value상태에서 [d]입력이 들어온다.
DWS_UTC_001_000	2.2 Change watch mode	Trigger입력이 들어온다.
DWS_UTC_002_000	2.3 Change TK mode	Trigger 입력이 들어온다.
DWS_UTC_003_000	2.4 Add one	Trigger 입력이 들어온다.
DWS_UTC_004_000	2.5 TK select	Trigger 입력이 들어온다.
DWS_UTC_005_000	2.6 Go time	Tick 입력이 들어온다.
DWS_UTC_006_000	2.7 Save TK	Trigger 입력이 들어온다.
DWS_UTC_007_000	2.8 Change AL mode	Trigger 입력이 들어온다.
DWS_UTC_008_000	2.9 AL select	Trigger 입력이 들어온다.
DWS_UTC_009_000	2.10 Alarm On Off	Trigger 입력이 들어온다.
DWS_UTC_010_000	2.11 AL save	Trigger 입력이 들어온다.
DWS_UTC_011_000	2.12 ST start	Trigger 입력이 들어온다.
DWS_UTC_012_000	2.13 ST stop	Trigger 입력이 들어온다.
DWS_UTC_013_000	2.14 ST reset	Trigger 입력이 들어온다.
DWS_UTC_014_000	2.15 ST lap	Trigger 입력이 들어온다.

DWS_UTC_015_000	2.20 ST go time	Enable입력과 Tick입력이 들어온다.
DWS_UTC_015_001	2.20 ST go time	Disable입력과 Tick입력이 들어온다.
DWS_UTC_016_000	2.21 Alarm Controller	Idle 상태에서, Alarm on == true && Current time == set time
DWS_UTC_016_001	2.21 Alarm Controller	Alarm rings 상태에서, (Current time - set time == 20) (Button != 0 && Button < 5)

2.2 Test items

Table 2 Test Case Identification

Test case identifier	Input specification	Output specification
DWS_UTC_000_000	controlState==Time keeping / a	ControlState==Change TK mode
DWS_UTC_000_001	State==Time keeping / b	State==Time keeping
DWS_UTC_000_002	State==Time keeping / c	State==Alarm
DWS_UTC_000_003	State==Time keeping / d	State==Time keeping
DWS_UTC_000_004	State==Change TK mode / a	State==Time keeping
DWS_UTC_000_005	State==Change TK mode / b	State==Increase TK value
DWS_UTC_000_006	State==Change TK mode / c	State==Change TK mode
DWS_UTC_000_007	State==Change TK mode / d	State== Change TK mode
DWS_UTC_000_008	State==Increase TK value / a	State==Time keeping
DWS_UTC_000_009	State==Increase TK value / b	State==Increase TK value
DWS_UTC_000_010	State==Increase TK value / c	State==Change TK mode

실습!!!

DWS_UTC_000_021	State==Lap time / b	State==Lap time
DWS_UTC_000_022	State==Lap time / c	State==Time keeping
DWS_UTC_000_023	State==Lap time / d	State==Lap time
DWS_UTC_000_024	State==Alarm / stOn==0 / a	State==Change Alarm mode
DWS_UTC_000_025	State==Alarm / stOn==0 / b	State==Alarm
DWS_UTC_000_026	State==Alarm / stOn==0 / c	State==Stopwatch off
DWS_UTC_000_027	State==Alarm / stOn==0 / d	State==Alarm
DWS_UTC_000_028	State==Alarm / stOn==1 / a	State==Change Alarm mode
DWS_UTC_000_029	State==Alarm / stOn==1 / b	State==Alarm
DWS_UTC_000_030	State==Alarm / stOn==1 / c	State==Stopwatch on

DWS_UTC_000_031	State==Alarm / stOn==1 / d	State==Alarm
DWS_UTC_000_032	State==Alarm / stOn==2 / a	State==Change Alarm mode
DWS_UTC_000_033	State==Alarm / stOn==2 / b	State==Alarm
DWS_UTC_000_034	State==Alarm / stOn==2 / c	State==Lap time
DWS_UTC_000_035	State==Alarm / stOn==2 / d	State==Alarm
DWS_UTC_000_036	State==Change Alarm mode / a	State==Alarm
DWS_UTC_000_037	State==Change Alarm mode / b	State==Increase Alarm value
DWS_UTC_000_038	State==Change Alarm mode / c	State==Change Alarm mode
DWS_UTC_000_039	State==Change Alarm mode / d	State==Change Alarm mode

실습!!!

DWS_UTC_013_000	Trigger in / e_time.tm_sec==30	e_time.tm_sec==0
DWS_UTC_014_000	Enable, Disable / stOn==30	stOn==2
DWS_UTC_015_000	Enable, tick	
DWS_UTC_015_001	Disable, tick	
DWS_UTC_016_000	State==Idle, Alarm on==true, current time==set time / alarm_state==0	Enable / alarm_state==1
DWS_UTC_016_001	State==alarm rings, (Current time - set time == 20) (Button != 0 && Button < 5) / alarm_state == 1	Disable / alarm_state==0

2.3 Input specifications

<Table 1 Test Design Identification> 참조.

2.4 Output specifications

<Table 1 Test Design Identification> 참조.

3 Environmental needs

DS.2012.DWS.UTP-1.1, Environmental needs 항목 참조.

4 Unit test summary report

4.1 Test summary report identifier

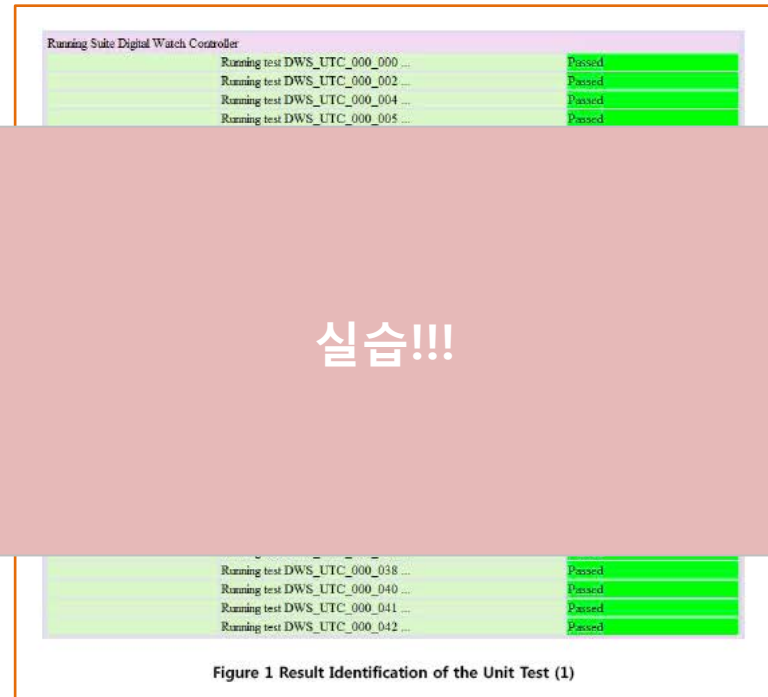
<Figure 1 Result Identification of the Unit Test (1)>

<Figure 2 Result Identification of the Unit Test (2)>

<Figure 3 Result Identification of the Unit Test (3)> 참조.

4.2 Evaluation

두 항목(DWS_UTC_015_000, DWS_UTC_015_001)은 Process [2.20 ST go time]을 테스트하기 위한 테스트 케이스이다. 해당 process는 tick을 받아서 시간을 올려주는 역할을 하며 DWS에서 중요한 역할을 담당한다. 하지만 해당 프로세스의 동작이 너무 단순하여 테스트가 불가능했다. 따라서 강제로 false입력을 넣어 결과를 출력하였다.



Running Suite Change watch mode	Running test DWS_UTC_001_000 ...	Passed
Running Suite Change TK mode	Running test DWS_UTC_002_000 ...	Passed
Running Suite Add one	Running test DWS_UTC_003_000 ...	Passed
Running Suite TK select	Running test DWS_UTC_004_000 ...	Passed
Running Suite Go time	Running test DWS_UTC_005_000 ...	Passed
Running Suite Save TK	Running test DWS_UTC_006_000 ...	Passed



Condition	1 == 0	Running test DWS_UTC_015_001 ...	Failed		
File Name	C:\Users\Junbeom\Yes\hmdsco\jobs\DWS\workspace\src\Test1.c	Line Number	408		
Condition	1 == 0	Running Suite Alarm Controller			
		Running test DWS_UTC_016_000 ...	Passed		
		Running test DWS_UTC_016_001 ...	Passed		
Cumulative Summary for Run					
Type	Total	Run	Succeeded	Failed	Inactive
Suites	17	17	- NA -	0	0
Test Cases	49	49	47	2	0
Assertions	49	49	47	2	n/a

Figure 3 Result Identification of the Unit Test (3)