



# 소프트웨어공학개론

## Electronic Door Lock System Unit Testing Plan

Team #3: 5bps

김태현 201211341

송병우 201011338

은다함 201013388

장호성 200611515

# Objectives

- 1) EDLS의 unit test를 수행하기 위해 필요한 활동 및 자원을 정의한다.
- 2) EDLS의 unit test를 수행하기 위한 test approach 및 techniques를 정의한다.
- 3) EDLS의 unit test를 수행하기 위한 환경적인 요구사항 및 test 도구들을 정의한다.

# Background

- Unit test는 시스템을 구성하는 최소 단위 모듈들을 대상으로 하는 test이며, 시스템의 성능을 좌우하는 feature들이 요구사항을 만족하는지 확인할 수 있는 기본적인 test approach이다.
- 이 프로젝트는 이번 수업의 SASD기법을 이용하여 개발된다. EDLS을 구성하는 unit은 SRA, SDS에서 정의한 것을 가져온 것이다.

# Scope

- 이 계획 문서는 EDLS의 unit test를 수행하기 위한 모든 것을 포함한다.
- EDLS의 unit test를 수행하기 위한 자원과 절차, test approach와 techniques과 필요로 하는 환경 및 도구 등을 정의 한다.
- EDLS의 unit test는 시스템을 구성하는 최소 단위의 모듈들을 대상으로 하며, 구현된 모듈이 요구사항을 만족하는지 test한다.

# Configuration management plan

- EDLS의 program source code 및 unit test를 위한 test code는 CTIP(Continuous Testing & Integration Platform)환경에서 이루어지며, program source code/test code 의 변경 및 수정사항은 지속적으로 통합되고 test된다.

## (1) program source code의 변경

- program source code의 변경 및 수정 사항이 발생 시, 이를 통합하고 수동적으로 unit test를 수행 한다.

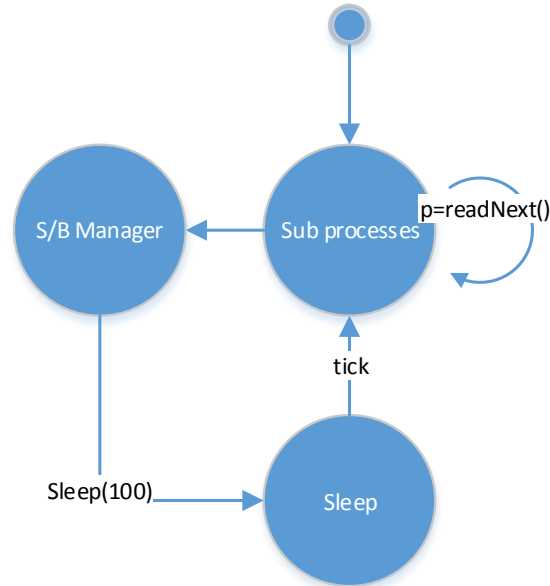
## (2) 일정주기

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





# Test items (cont.)



Reference No.	2.1
Name	Process controller
Input	Tick, Process Queue Control
Output	-
Process Description	매 Tick마다 프로세스 큐에 등록된 프로세스가 있으면 순차적으로 실행한다. 큐가 비어있어도 버튼/센서 매니저는 항상 동작한다.



# 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

# Features to be tested (cont.)

**Table 1. 테스트할 Process (DFD) 리스트**

Item
1 비밀번호 설정
2.1 Process controller
2.2 Process Queue controller
2.3 S/B Manager
2.3.1 Input controller
2.3.2 Read input
2.3.3 Add process
2.4 Sub Processes
2.4.1 Number input process
2.4.2 Authorize process
2.4.3 Door lock control process
2.4.4 Timer process
2.4.5 T.P-close door
2.4.6 T.P-Turn backlight off

# 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 테스트 하지 않을 Process(DFD)리스트>의 Process name 참조

# Features not to be tested (cont.)

**Table 2. 테스트하지 않을 Process (DFD) 리스트**

ID	Name	Description
2.4.3.2	Lock	문을 닫는다.
2.4.3.3	Unlock	문을 연다.
2.4.4.2	Count	타이머를 1 증가시킨다.
2.4.5.2	Close door	문을 닫는다.
2.4.6.2	Turn backlight off	백라이트를 끈다.
2.4.6.3	Alarm	시간초과 경고음을 낸다.

# Approach

- EDLS의 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) Brute force testing:

각 모듈의 요구사항을 만족하는지를 확인할 수 있는 test case를 작성한다. 그 이외의 예외 상황에 대해서는 test하지 않는다.

# Item pass/fail criteria

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

# Test identification

## Table 3. Test Design Identification

identifier	feature	Valid/ <i>Invalid</i> value
EDLS.UTC_000_000	1 비밀번호 설정	비밀번호 입력 상태에서 사용자 입력을 받는다.
EDLS.UTC_000_001	1 비밀번호 설정	S/B Setting 상태에서 센서 버튼 플래그를 설정한다.
EDLS.UTC_000_002	1 비밀번호 설정	<b>비밀번호가 설정되지 않는다.</b>
EDLS.UTC_001_000	2.1 EDLS Control	비밀번호 설정을 입력 받아 무한 루프 트리거를 실행한다.
EDLS.UTC_002_000	2.1 Process controller	PQManager에서 함수 주소를 입력 받아 trigger 값을 반환한다.
EDLS.UTC_003_000	2.2.1 Process queue controller	PQ를 입력 받아 명령을 수행한다.
EDLS.UTC_004_000	2.2.5 Read Next Process	current의 다음 PQ를 current로 저장한다.
EDLS.UTC_004_001	2.2.5 Read Next Process	current의 다음이 없으면 종료한다.
EDLS.UTC_005_000	2.2.3 Add process	함수주소와 인자를 입력 받아 PQ에 구조체를 추가한다.
EDLS.UTC_006_000	2.2 Sort	Trigger 되어 PQ를 우선순위 순서대로 정렬한다.
EDLS.UTC_007_000	2.2 Remove	입력 받은 함수주소의 PQ를 지운다.
EDLS.UTC_008_000	2.3.1 Input controller	S/B Flag를 입력 받아 하위Trigger한 결과를 반환한다.
EDLS.UTC_009_000	2.3.2 Read input	S/B Flag중 1인 index의 S/B값을 읽는다.

# Test identification (cont.)

EDLS.UTC_010_000	2.3.3 Add process	S/B index와 값을 입력 받아 PQ를 등록한다.
EDLS.UTC_011_000	2.4.1.1 Number Input Controller	명령번호로 F를 받으면 버튼입력 관련 S/B를 켜다.
EDLS.UTC_011_001	2.4.1.1 Number Input Controller	명령번호로 한 자리 숫자를 받으면 2.4.1.2를 trigger한다.
EDLS.UTC_012_000	2.4.1.2 Read input	0~9를 입력 받아 input password를 쓴다.
EDLS.UTC_013_000	2.4.1.3 Authorization process	Password가 일치하면 문 여는 프로세스를 등록한다.
EDLS.UTC_013_001	2.4.1.3 Authorization process	Password가 불일치하면 재입력을 준비한다.
EDLS.UTC_014_000	2.4.1.4 Turn backlight/timer on	Trigger되어 timer 프로세스를 PQ에 추가한다.
EDLS.UTC_015_000	2.4.1.5 Timer reset	Trigger되어 timer count를 0으로 설정한다.
EDLS.UTC_016_000	2.4.1.6 Self terminate	Trigger되어 PQ에서 숫자입력 프로세스를 삭제한다.
EDLS.UTC_017_000	2.4.1.7 Add authorization process	인증 프로세스 주소를 PQManager로 넘겨준다.
EDLS.UTC_018_000	2.4.2.1 Authorization controller	input password, set password를 입력 받는다.
EDLS.UTC_019_000	2.4.2.3 Disable user input	Trigger 되어 사용자 입력 S/B flag를 끈다.
EDLS.UTC_020_000	2.4.2.4 Initialize1	Trigger 되어 사용자 입력 S/B flag를 켜다.
EDLS.UTC_020_001	2.4.2.4 Initialize1	Trigger 되어 PQ에 타이머 프로세스를 삭제한다.
EDLS.UTC_021_000	2.4.2.5 Self terminate	PQ에서 Authorize 프로세스를 삭제한다.
EDLS.UTC_022_000	2.4.2.1 Add door control process	door lock control process 를 PQ에 등록한다.
EDLS.UTC_023_000	2.4.4.2 Count	Time / T.P addr를 입력 받아 타이머 값을 증가시킨다.
EDLS.UTC_024_000	2.4.4.3 Terminal	타이머 값을 0으로 초기화, 타이머 프로세스 삭제, PQ에 종료작업 등록
EDLS.UTC_025_000	2.4.5.3 Self Terminate	T.P close 프로세스를 PQ서 삭제한다.
EDLS.UTC_026_000	2.4.6.4 Self Terminate	문 종료 완료 시 T.P TBO 프로세스를 PQ에서 삭제한다.



# Feature pass/fail criteria

- EDLS의 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.

# Test case specification identifier

## Table 4. Test Case Identification

Test case identifier	Input specification	Output specification
EDLS.UTC_000_000	input!=0xFFFF	pass>>12!=0xFFFF
EDLS.UTC_000_001	SbFlags==0	SbFlags==(1<<DOS)  (1<<DB) (1<<KEY)
EDLS.UTC_000_002	input==0xFFFF	pass>>12==0xFFFF
EDLS.UTC_001_000	Pass>>12!=0xFFFF	Trigger loop
EDLS.UTC_002_000	fnAddr==PQManager .current	Func(fnAddr)
EDLS.UTC_003_000	PQ!=null	Trigger
EDLS.UTC_004_000	current.next!=null	current.next
EDLS.UTC_004_001	current.next==null	null
EDLS.UTC_005_000	fnAddr!=0/arg1/arg2	length>0 && last.addr==fnAddr
EDLS.UTC_006_000	Trigger in	first.next!=null && (getPriority(first.addr) > getPriority(first.next .addr))
EDLS.UTC_007_000	fnAddr!=0	find(fnAddr)<0
EDLS.UTC_008_000	SbFlags	Trigger
EDLS.UTC_009_000	SbFlags / index	SbFlags>>index&1
EDLS.UTC_010_000	Index / isOn	PQManager.length>0

# Test case specification identifier (cont.)

EDLS.UTC_011_000	Comm==0xF	((SBFlags>>NB)&1==1) && ((SBFlags>>KS)&1==1)
EDLS.UTC_011_001	Comm>=1 && Comm<=9	Trigger
EDLS.UTC_012_000	In>=0&&In<=9 / i<4	Pass>>(i-3)&1==In
EDLS.UTC_013_000	pass&0xFFFF== (pass>>12&0xFFFF)	PQManager.find(DoorControl)>=0
EDLS.UTC_013_001	pass&0xFFFF!= (pass>>12&0xFFFF)	PQManager.find(DoorControl)<0 && pass&0xFFFF==0xFFFF
EDLS.UTC_014_000	Trigger in	PQManager.find(Timer)>=0
EDLS.UTC_015_000	Trigger in	timer_cnt==0
EDLS.UTC_016_000	Trigger in	PQManager.find(NumIn)<0
EDLS.UTC_017_000	Trigger in	PQManager.find(Auth)>=0
EDLS.UTC_018_000	Trigger in	(Pass>>12&0xFFFF !=0xFFFF)&&(Pass& 0xFFFF!=0xFFFF)
EDLS.UTC_019_000	Trigger in	(SBFlags>>NB&1==0)&& (SBFlags>>KS&1==0)
EDLS.UTC_020_000	Trigger in	(SBFlags>>NB&1==1)&& (SBFlags>>KS&1==1)
EDLS.UTC_020_001	fn==Timer	PQManager.find(fn)<0
EDLS.UTC_021_000	fn==Auth	PQManager.find(fn)<0
EDLS.UTC_022_000	fn==DoorControl	PQManager.find(fn)>=0
EDLS.UTC_023_000	Time>0&&TPAddr!=0	timer_cnt>0
EDLS.UTC_024_000	TPAddr!=0	timer_cnt==0 && PQManager.find(Timer)<0 && PQManager.find(TPAddr)>0
EDLS.UTC_025_000	Trigger in	PQManager.find(TPClose)<0
EDLS.UTC_026_000	closed==1	PQManager.find(TPTBO)<0

# Testing tasks & Schedule

**Table 5. Testing tasks & Schedule**

Task	Predecessor tasks	Special skills
(1)Unit Test Plan 작성	EDLS-2013-SRS 작성 EDLS-2013-SRA 작성 EDLS-2013-SDS 작성 EDLS 구현	
(2)Test design specification	Task 1	EDLS에 대한 이해
(3)Test case specification	Task 2	EDLS에 대한 이해
(4) Test execution	Task 3	Test code 작성 Test tools 에 대한 이해
(5) Test result report	Task 4	
(6) 개발팀에 report 전달	Task 5	

# Environmental needs

- EDLS의 unit test를 위한 환경적 요구사항은 다음과 같다.

Hardware & Platform, Eclipse IDE (Integrated Development Environment)

- gcc compiler / linker

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