

객체지향개발방법론

# 2nd Iteration - T3

201212519 컴퓨터공학과 김선우  
201510624 컴퓨터공학과 김용현  
201614157 컴퓨터공학과 김도연  
201614158 컴퓨터공학과 장다혜

SQA의 요청에 따른 수정사항

# Refine 1000 Planning

## 1003 : Define Requirement

1. 구분자사이의 연관성 모호 ←
  - a. R 1.2.1 Change mode와 R 1.2.2 Stop ringing 사이의 연관성이 보이지 않음 -> R 1.2와 R 1.3로 수정 요망 ←



R 1.1	Set time
R 2.1	Change mode
R 2.2	Stop ringing
R 3.1	Start stopwatch

## 1004 : Record Terms in Glossary

1. 비문 사용 ←
  - a. “타이머 설정한 타이머 시간이 알람이 울린다” -> “타이머에서 설정한 타이머 시간이 완료되면 알람이 울린다.” 로 변경 요망 ←



수정본 전달로 해결

## 1006. Define Business Use Case

1. Use Case Specification들 간의 Level이 맞지 않는 문제 ←
  - a. 유독 Set time의 Specification만 자세하게 설명되어 있음 Planning 단계이므로 더 간결하게 표현하기를 요망 ←



Use case	1. Set time
Actors	User
Description	- 한국, 미국, 영국, 중국, 러시아, 싱가포르 중 한 나라를 설정한다 - year를 설정한다 - month를 설정한다 - day를 설정한다 - hour를 설정한다 - minute를 설정한다 - second를 설정한다.

# Refine 1000 Planning

---

## 1003 : Requirement와의 동기화 문제

- a. R.3.1 Select alarm num, R.3.2 Remove alarm num :  
1006 단계에서 number로 용어를 임의 변경 -> 통일 요망
- b. R.5.1 Change world : 1006 단계에서 Change country로 임의  
변경 -> 통일 요망
- c. R.6.1 Turn on brightness control : 1006 단계에서 Control ->  
대소문자 통일 요망
- d. R.6.2 Turn off brightness control : 1006 단계에서 Control ->  
대소문자 통일 요망
- e. R.6.4 Change brightness : 1006 단계에서 change Brightness -  
대소문자 통일 요망
- f. R.6.3 Control brightness : 1006 단계에서 사라짐 -> 작성 요망
- g. 넘버링에서 12, 13번이 생략되고 14번으로 넘어감 -> 통일 요망



- 대소문자 통일 문제 전체 모든 문  
서에 걸쳐 수정
- 용어 일관성 문제 전체적으로 해  
결

# Refine 1000 Planning

---

- iv. 1009 : Develop System Test Case←
  - 1. 1006 : Define Business Use Case 과의 동기화 문제←
    - a. 2번 Show current time test의 Use case :Show current time 은 1006단계에서 Define 되지 않음←
    - b. 3번 Show current date test의 Use case :Show current date 은 1006단계에서 Define 되지 않음←
    - c. 4번 Stop ringing은 R 1.2.2로 명시 되어 있으나 R 1.3.2로 표기←



- 용어 일관성 문제 전체적으로 해결
- 2,3번 Use case의 경우, 삭제 되었기 때문에 1000, 2030, 2040 문서 전체에 걸쳐 수정 완료

# Refine 2030 Analysis

## 2031 : Define Essential Use Cases

1. Use Case Specification 내에서의 표기 Level이 맞지 않는 문제  
a. 대부분의 Use Case의 Specification에서 함수형 표기와 글 표기를 섞어서 사용 -> 하나로 통일하기를 요망
2. 모호한 표현  
a. "1. Set time"의 "(A): A 버튼을 눌러 Set Section로 넘어간다." 에서 Set Section에 대한 정의가 없어 판단이 모호함  
b. "4. Start stopwatch"의 "(S) stopwatchTime 값이 초당 0.01 단위로 증가한다" 에서 0.01 단위의 정의가 모호함  
c. "14. Start timer"의 "E.1 각 영역의 값이 0이 될 경우 감소를 시키지 않는다." 에서 각 영역이 뜻하는 바가 모호함  
d. "18. Change world"의 Pre-requisites "Set country 상태가 선행되어야 한다."에서 Set country에 대한 정의가 없어 해석이 모호함



1. 문장형 표기로 통일

1. Section에 관한 명세를 모든 Use case에 반영

1. Set country 상태에 대한 내용 수정 완료

# Refine 2030 Analysis

## 2031 : Define Essential Use Cases

1. Use Case Specification 내에서의 표기 Level이 맞지 않는 문제  
a. 대부분의 Use Case의 Specification에서 함수형 표기와 글 표기를 섞어서 사용 -> 하나로 통일하기를 요망
2. 모호한 표현  
a. "1. Set time"의 "(A): A 버튼을 눌러 Set Section로 넘어간다." 에서 Set Section에 대한 정의가 없어 판단이 모호함  
b. "4. Start stopwatch"의 "(S) stopwatchTime 값이 초당 0.01 단위로 증가한다" 에서 0.01 단위의 정의가 모호함  
c. "14. Start timer"의 "E.1 각 영역의 값이 0이 될 경우 감소를 시키지 않는다." 에서 각 영역이 뜻하는 바가 모호함  
d. "18. Change world"의 Pre-requisites "Set country 상태가 선행되어야 한다."에서 Set country에 대한 정의가 없어 해석이 모호함



1. 문장형 형식으로 통일 (변수명까지만 define)
1. Section에 관한 명세를 모든 Use case에 반영
1. Set country 상태에 대한 내용 수정 완료

# Refine 2030 Analysis

## 2035 : Define System Sequence Diagram

### 1. Use case Specification과의 불일치 문제

- a. "1. Set time"의 Specification에서 표기한 input/output의 갯수와 실제로 작성한 Diagram의 input/output 갯수가 다름 -> 일치시키기를 권고
- b. "1. Set time"의 Planning 단계의 Specification에서는 world에 대한 setting권한이 Change world에게 있음 -> Sequence Diagram에서 world를 변경하는 부분 삭제 요망
- c. "3. Stop ringing"의 "(S) Beep 이 실행된 지 3초가 지나지 않았을 때, 사용자의 요청이 들어오면 Beep을 멈춘다"에 대한 표현이 Sequence Diagram에서 누락됨 -> 추가하는 것을 권고
- d. "10. Set alarm time"의 "7. (A) C버튼을 사용하여 설정을 종료한다" 및 "8. (S) 설정한 시간을 화면에 출력한다"에 대한 표현이 Sequence Diagram에서 누락됨 -> 추가하는 것을 권고
- e. "19. Change SW"의 Sequence Diagram 에서 "3 : selectSW"에 대한 Specification 누락 -> 추가하는 것을 권고



- a. Use case와 Sequence의 input / output 개수는 다를수 있으므로 수정하지 않았음
- a. Timekeeping 모드 내의 Set time에서의 나라 변경과 Worldtime 모드에서의 나라 변경 use case를 혼동한 것으로 생각됨 -> 수정 x
- a. System 내부에서 동작하는 부분이여서 추가하지않음
- a. use case를 잘못써놓은 부분이여서 use case를 변경함(e도 동일)

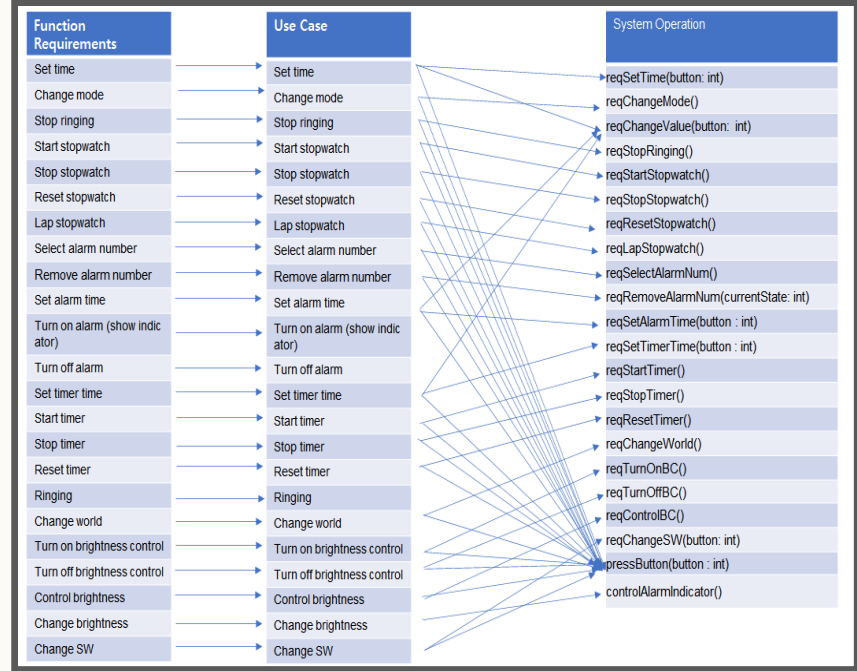


# Refine 2030 Analysis

## 2039 : Traceability Analysis

- vi. 2039: Traceability Analysis
1. 정의되지 않은 Use Case
    - a. 앞선 specification에서 명시되지 않은 Use Case
      - > 앞의 Use Case들을 다시 명세하기를 권고

1. use case에 명시된 것으로 Traceability Analysis를 변경함.



# Refine 2040 Design

## 2041 : Design Real Use Cases

### 1. 모호한 표현

- a. "1. Set time"의 "4.3.1 (A): 사용자가 버튼을 누른다," 에서 무슨 버튼을 누르는 지 명시 요청
- b. "10. Set alarm time"의 Exceptional courses of events에서 영역의 의미가 모호함



- 버튼 및 영역 명시 완료

### 5. 희망사항

- a. "1. Set time"의 "E4. hour값이 24H 일 때, B 버튼을 눌러 증가시키면 1H 이 된다. 1H 일 때, D 버튼을 눌러 감소시키면 24H 이 된다." 에서 24H->1H 보다는 23H->0H가 사용자 편의를 위해 더 좋아보임 ("13. Set timer time"에서는 "E3. hour의 최대값(23)에서 B 버튼을 누르면 변수값을 0으로 변경한다."로 이와 같이 구현되어 있다.)



- SQA의 희망사항대로 변경 완료

# Refine 2040 Design

---

## 2043 : Define Interaction Diagram

### ii. 2043 : Define Interaction Diagrams

#### 1. 전체적인 문제

- a. 보고서 상에서 Sequence Diagram의 Use case가 무엇인지 제대로 명시되어 있지 않아 Use case 별로 구분하여 문제점을 파악하기가 쉽지 않음 -> 수정 요망



- 각 Sequence에 Use Case 명시

# Refine 2040 Design

---

2. Use case Specification 과의 불일치 문제
  - a. "1."(Set time 으로 추정)의 "4.3.1 (A) : 사용자가 버튼을 누른다. 4.3.2 (S) : changeValue(currentState :int , button : int)를 호출한다."가 Sequence Diagram 에서 임의로 변경되어 적용됨 -> 수정 요망
  - b. "4."(Change mode 로 추정)의 "2-1. (S) changeMode(currentState)를 호출한다."에 대한 내용이 Sequence Diagram 에서 누락됨 -> 수정 요망
  - c. "4."(Change mode 로 추정)의 Sequence Diagram 에서 "6 : display ()"에 대한 내용이 Spec 에서 누락됨 -> 수정 요망
  - d. "Remove alarm num"의 "1. (A) User 가 D 버튼을 누른다."에 대한 내용이 Sequence Diagram 에서 "1 : pressButtonB"로 잘못 표기됨 -> 수정 요망
  - e. "Set alarm time"의 Sequence Diagram 에서 모든 "display"에 대한 내용이 Spec 에서 누락됨 -> 수정 요망
  - f. "Turn on alarm", "Turn off alarm"의 "2. (S) checkAlarmArray()를 호출한다."에 대한 내용이 Sequence Diagram 에서 조건문으로 임의 변경됨 -> Spec 수정 요망
  - g. "Start timer", "Stop timer", "Reset timer"의 Sequence Diagram 에서 "5 : display(timerTime)"이 Spec 에는 "5. (S) display 를 호출한다."로 표기 -> 통일 요망

- a. changeValues내 button 인자 추가
- b. changeMode에 관한 operation 삭제로 Use Case 수정
- c. display 추가
- d. pressButtonD로 수정
- e. display 추가
- f. 수정
- g. 통일 수정

# Refine 2040 Design

---

## 3. Sequence Diagram 문제

- a. "5."(Stop ringing으로 추정)에서 "6 : reqStopRinging"은 필요 없어 보임
- b. "Start stopwatch" Sequence Diagram의 "1 : pressButton"에서 B버튼을 누른다는 내용이 생략됨 -> 수정 요망
- c. "Stop stopwatch" Sequence Diagram의 "1 : pressButton"에서 B버튼을 누른다는 내용이 생략됨 -> 수정 요망
- d. "Set timer time"의 Sequence Diagram이 "Set alarm time"으로 잘못 표기됨
- e. "Change world"의 Sequence Diagram이 "Change country"라고 잘못 표기됨

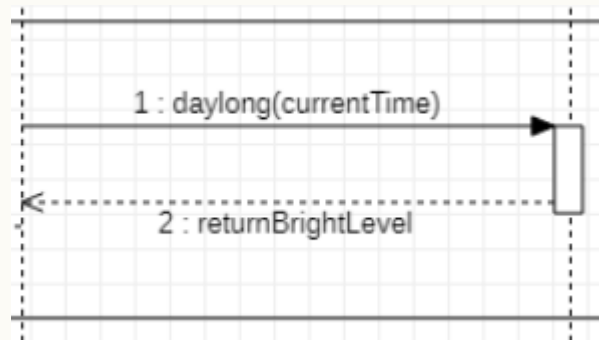
- a. 버튼을 눌러서 알람을 눌러야하므로 필요하다고 생각되어 수정 x
  - b. Sequence diagram에 버튼 추가 수정
  - c. Sequence diagram에 버튼 추가 수정
- Change world로 수정 완료

# Refine 2040 Design

- i. "19."(Turn on brightness control로 추정)에서 "(S) switchBC()를 호출하여 brightness 에 true 를 저장한다."가 Sequence Diagram에서 누락되어 있음 -> 수정 요망
- j. "19."(Turn on brightness control로 추정)의 Sequence Diagram에서 daylong의 parameter가 누락되어 있음 -> 수정 요망
- k. "20."(Turn off brightness control로 추정)에서 "4. (S) switchBC()를 호출하여 brightness 에 false 를 저장한다. 5. (S) initializeBC()를 호출하여 brightLevel을 '0' 으로 조정한다."가 Sequence Diagram에서 누락되어 있음 -> 수정 요망

- l. "21."(Control brightness로 추정)의 Sequence Diagram에서 모든 display에 임의로 변수가 추가됨 -> Spec과 통일 요망
- m. "Change brightness"의 Sequence Diagram에서 "1 : daylong"의 parameter가 누락됨

- switchBC(), daylong(), initialize() 함수는 checkBC()에 의해 BRC 클래스 내부에서 호출되는 함수이므로 시퀀스에 표시될 필요가 없음 -> 수정 X



# Refine 2040 Design

---

## 3. Sequence Diagram 문제

- "5."(Stop ringing으로 추정)에서 "6 : reqStopRinging"은 필요 없어 보임
- "Start stopwatch" Sequence Diagram의 "1 : pressButton"에서 B버튼을 누른다는 내용이 생략됨 -> 수정 요망
- "Stop stopwatch" Sequence Diagram의 "1 : pressButton"에서 B버튼을 누른다는 내용이 생략됨 -> 수정 요망
- "Set timer time"의 Sequence Diagram이 "Set alarm time"으로 잘못 표기됨
- "Change world"의 Sequence Diagram이 "Change country"라고 잘못 표기됨

- 버튼을 눌러 ring을 멈추는 spec에 따라 필요한 내용이므로 수정  
X
- Sequence diagram 수정 완료
- Sequence diagram 수정 완료
- Sequence diagram 수정 완료
- 수정 완료

# Unit Test



# Unit Test (Ringing)

✓ Test Results	568 ms
✓ RingTest	568 ms
✓ checkAlarm()	65 ms
✓ checkTimer()	1 ms
✓ ringing()	502 ms

```
@Test
void ringing() {
    // ringing() 제대로 작동하는지
    Ring ringTh = new Ring();
    Thread thr = new Thread(ringTh);
    thr.start();
    try {
        Thread.sleep( millis: 500);
    } catch (InterruptedException e) {
        e.printStackTrace();
    }
    assertTrue(ringTh.isRinging());
    thr.interrupt();
}
```

```
@Test
void checkAlarm() {
    // 알람이 경시에 울리는지
    AlarmMode alm = new AlarmMode();
    TimerMode trm = new TimerMode();
    TimekeepingMode tkm = new TimekeepingMode();
    Ring ring = new Ring(alm, trm);
    Ring ringTh = new Ring();
    tkm.setValue( currentState: 5, value: 0);
    tkm.setValue( currentState: 6, value: 0);
    alm.getAlarm( state: 0).setHour(0);
    alm.getAlarm( state: 0).setMinute(0);
    ring.checkAlarm(alm, tkm.getCurrentTime());
    assertTrue(ring.isOnOff());
}
```

```
@Test
void checkTimer() {
    AlarmMode alm = new AlarmMode();
    TimerMode trm = new TimerMode();
    trm.getTimerTime().setM_second(2);
    Ring ring = new Ring(alm, trm);
    assertTrue(ring.checkTimer(trm));
    trm.getTimerTime().setM_second(0);
    assertFalse(ring.checkTimer(trm));
}
```

# Code Revision

# Code Revision

R 2.1↔	T stopwatch.3↔	<u>스탑워치의 초 영역이 1/100초 단위로 측정되는지</u> 검증합니다.↔	R↔	<u>D버튼 연타시 1/100 스케일을 벗어나</u> <u>비정상작동</u> 합니다.↔
--------	----------------	---	----	--



M\_sec의 영역이 100이상 나오지 않게 수정

R 2.2↔	T stopwatch.6↔	<u>B버튼 이외의 입력으로 스타워치가 일시정지 되지</u> 않음을 검증합니다.↔	R↔	<u>D버튼을 연타할 경우 초 영역이</u> <u>비정상작동</u> 합니다.↔
--------	----------------	---	----	---



Start Stopwatch이후에 D버튼 작동을 수정

R 2.4↔	T stopwatch.8↔	<u>D버튼을 눌러 현재 lap time을</u> 보여줌을 검증합니다.↔	R↔	<u>2호 이상 lap time을</u> 체크하면 <u>분 초</u> 영역이 깨집니다.↔
R 2.4↔	T stopwatch.9↔	<u>B버튼을 누를 때 다시 현재 스타워치의 경과</u> 시간을 표시함을 검증합니다.↔	R↔	<u>lap time을 2회 이상</u> 체크하면 현재 스타워치의 시간을 표현하지 못합니다. 더불어 <u>분 초</u> 영역이 <u>비정상작동</u> 합니다.↔



stopwatch의 기능중 LapTime에 대한 전반적인 기능 수정

# Code Revision

---

R 4.3↩	T.timer.5↩	타이머 정지 상태에서 <u>C버튼으로 모드 변환</u> 시 타이머가 유지됨을 검증합니다.↩	F↩	<u>timer</u> 동작 후 다른 버튼이 동작하지 않음.↩
R 4.4↩	T.timer.6↩	<u>D버튼을 이용해 타이머의 시간을 리셋함</u> 을 검증합니다.↩	P↩	↩
R 4.2↩	T.timer.7↩	동작중인 타이머가 모드 변화에도 동작함을 검증합니다.↩	F↩	<u>timer</u> 동작 후 다른 버튼이 동작하지 않음.↩
R 4.2↩	T.timer.8↩	<u>timer모드가 죽지않고</u> 동작함을 검증합니다.↩	F↩	<u>timer</u> 동작 후 다른 버튼이 동작하지 않음.↩



Timer에 대한 1차 iter에서 부족한 부분 수정

# Code Revision

---

R 4.3	T.timer.5	타이머 정지 상태에서 <u>C버튼</u> 으로 모드 변환 시 타이머가 유지됨을 검증합니다.	R	timer 동작 후 다른 버튼이 동작하지 않음.
R 4.4	T.timer.6	<u>D버튼</u> 을 이용해 타이머의 시간을 <u>리셋함</u> 을 검증합니다.	P	↔
R 4.2	T.timer.7	동작중인 타이머가 모드 변화에도 동작함을 검증합니다.	R	timer 동작 후 다른 버튼이 동작하지 않음.
R 4.2	T.timer.8	<u>timer모드가 죽지않고</u> 동작함을 검증합니다.	R	timer 동작 후 다른 버튼이 동작하지 않음.



Timer에 대한 1차 iter에서 부족한 부분 수정

# Code Revision

R 1.3.2↵	T.buzzer.1↵	buzzer가 울릴때 어떠한 버튼으로도 buzzer를 중단시킬 수 있음을 검증합니다.↵	F↵	buzzer 미구현↵
R 1.3.2↵	T.buzzer.2↵	buzzer의 작동이 시작하는 시점에서 버튼 기능에 우선함을 검증합니다.↵	F↵	buzzer 미구현↵
R 1.3.2↵	T.buzzer.3↵	buzzer작동 후 3초 뒤 자동 중지됨을 검증합니다.↵	F↵	buzzer 미구현↵
R 4.5↵	T.buzzer.4↵	설정된 알람시간에 buzzer가 정확히 작동함을 검증합니다.↵	F↵	buzzer 미구현↵
R 4.5↵	T.buzzer.5↵	지정된 시간만큼 buzzer가 작동하는지	F↵	buzzer 미구현↵



Buzzer에 대한 미구현 사항들을 2차 Iter에서 추가 및 수정

# Code Revision

---

20	<u>timer</u>	<u>change num</u>	<u>changed</u>	<u>n/a</u>	<2.3.9.0 >	<u>F</u>
<i>AABAABBDAAABAAB</i> -> 2초씩 감소						
21	<u>timer</u>	<u>reset</u>	<u>set</u>	<u>n/a</u>	<2.4.10. 0.>	<u>P</u>
22	<u>timer</u>	<u>start timer</u>	<u>timer working</u>	<u>n/a</u>	<2.7.11. 0.>	<u>F</u>
<i>AABAABBDAAABAABBDAAABAAB</i> -> reset횟수를 늘릴수록 한번에 많은 값이 감소						



Timer의 리셋 횟수에 따른 감소량 변화 수정

# Traceability Matrix



Essential UseCase	S-Link	SID	Operation in Sequence Diagram	M-Link	MID	Method	Class	Unit Test	Class	System Test	Class
Set time	S1, S3, S21	S1	reqSetTime(button: int)	M1-8, M11	M1	setDisplay() : void	GUI	enterSetSection()		Check Button Test	
Change mode	S2, S21	S2	reqChangeMode()	M1-8	M2	setDisplay2() : void		displayMinute()		Set Time_Year Test	
Stop ringing	S4, S21	S3	reqChangeValue(button: int)	M1-8, M14, M24, M32	M3	setDisplay3() : void		changeValue()	TimekeepingMode	Set Time_Month Test	
Start stopwatch	S5, S21	S4	reqStopRinging()	M1-8, M25	M4	setDisplay4() : void		displayDay()		Set Time_Day Test	
Stop stopwatch	S6, S21	S5	reqStartStopwatch()	M1-8, M16	M5	setDisplay5() : void		previousWorld()		Set Time_Hour Test	
Reset stopwatch	S7, S21	S6	reqStopStopwatch()	M1-8, M17	M6	setDisplay6() : void		nextWorld()		Set Time_Minute Test	
Lap stopwatch	S8, S21	S7	reqResetStopwatch()	M1-8, M18	M7	setDisplay7() : void		displayMonth()		Set Time_World Test	
Select alarm number	S9, S21	S8	reqLapStopwatch()	M1-8, M19	M8	setDisplay8() : void		displayHour()		display_Year Test	
Remove alarm number	S10, S21	S9	reqSelectAlarmNum()	M1-8	M9	watchBLC(int i): void		displayYear()		display_Month Test	
Set alarm time	S3, S11, S21	S10	reqRemoveAlarmNum(currentState: in	M1-8, M21	M10	ringing () : void		startStopwatch()		display_Day Test	
Turn on alarm		S11	reqSetAlarmTime(button : int)	M1-8, M22	M11	enterSetSection(currentState: int) : int	TimekeepingMode	stopStopwatch()	StopwatchMode	display_Hour Test	
Turn off alarm		S12	reqSetTimerTime(button : int)	M1-8, M28	M12	nextWorld(): void		resetStopwatch()		display_Minute Test	
Set timer time	S3, S12, S21	S13	reqStartTimer()	M1-8, M29	M13	previousWorld() void		lapTime()		change_Mode Test	DWS controller
Start timer	S13, S21	S14	reqStopTimer()	M1-8, M30	M14	changeValue(currentState: int, button: int) : void		changeValue()	AlarmMode	Check Button Test	
Stop timer	S14, S21	S15	reqResetTimer()	M1-8, M31	M15	changeMode(currentState: int) : void		removeAlarmNum()		stop Ringing Test 1	AlarmMode
Reset timer	S15, S21	S16	reqChangeWorld()	M1-8, M33	M16	startStopwatch() : StopwatchTime	StopwatchMode	isAlarmIndicator()		stop Ringing Test 2	
Ringing		S17	reqTurnOnBC()	M1-8, M9, M35, M39	M17	stopStopwatch() : StopwatchTime		getAlarmTime()		Ringing Test	
Change world	S16, S21	S18	reqTurnOffBC()	M1-8, M9, M35, M39	M18	resetStopwatch() : StopwatchTime		resetTimer()	TimerMode	start Stopwatch Test 1	
Turn on brightness control	S17, S21	S19	reqControlBC()	M1-8, M9, M38	M19	lapTime() : StopwatchTime		changeValue()		start Stopwatch Test 2	
Turn off brightness control	S18, S21	S20	reqChangeSW(button: int)	M1-8, M41	M20	getAlarmTime() : Time	AlarmMode	startTimer()		start Stopwatch Test 3	
Control brightness	S19, S21	S21	pressButton(button : int)	M1-8	M21	removeAlarmNum() : void		stopTimer()		start Stopwatch Test 4	
Change brightness	S22	S22	controlAlarmIndicator()	M1-8, M23	M22	enterSetSection(currentState : int) : int		changeIndex()	WorldtimeMode	stop Stopwatch Test 1	
Change SW	S20, S21				M23	checkAlarmArray() : boolean		changeBCLevel()		stop Stopwatch Test 2	StopwatchMode
					M24	changeValue(currentState: int) : void		getBrightLevel()		stop Stopwatch Test 3	
					M25	check3Second() : boolean	Ring	initializeBC()	BrightcontrolMode	reset Stopwatch Test	
					M26	checkAlarm() : boolean		switchBC()		lap Stopwatch Test 1	
					M27	checkTimer() : boolean		checkBC()		lap Stopwatch Test 2	
					M28	enterSetSection(currentState : int) : int	TimerMode	dayLong()		select Alarm Num Test	
					M29	startTimer() : Time		selectSW()	SWMode	Remove Alarm Num Test	
					M30	stopTimer() : Time		changeSW()		Set alarm Time Test 1	AlarmMode
					M31	resetTimer() : Time				Set alarm Time Test 2	
					M32	changeValue(currentState: int) : void				Turn on Alarm Test	
					M33	changeIndex() : int	WorldtimeMode			Turn off Alarm Test	
					M34	changeWorldTime() : void				Set Timer Time Test	
					M35	checkBC() : boolean	BrightcontrolMode			Start Timer Test 1	
					M36	switchBC() : void				Start Timer Test 2	
					M37	initializeBC() : void				Start Timer Test 3	TimerMode
					M38	changeBCLevel() : int				Stop Timer Test 1	
					M39	daylong(currentTime: Time) : int				Stop Timer Test 2	
					M40	changeSW() : int	SWMode			Reset Timer Test	
					M41	enterChangeSW() : void				Change World Test	WorldtimeMode
					M42	selectSW() : int				Turn on Bright Control Test	
										Turn off Bright Control Test	BrightcontrolMode
										Control Brightness Test	
										Change Brightness Test	
										Change SW Test	SWMode

Method	Class	Unit Test	Class	System Test	Class
setDisplay1() : void	GUI	enterSetSection()	TimekeepingMode	Check Button Test	TimekeepingMode
setDisplay2() : void		displayMinute()		Set Time_Year Test	
setDisplay3() : void		changeValue()		Set Time_Month Test	
setDisplay4() : void		displayDay()		Set Time_Day Test	
setDisplay5() : void		previousWorld()		Set Time_Hour Test	
setDisplay6() : void		nextWorld()		Set Time_Minute Test	
setDisplay7() : void		displayMonth()		Set Time_World Test	
setDisplay8() : void		displayHour()		display_Year Test	
watchBLC(int i): void		displayYear()		display_Month Test	
ringing () : void		startStopwatch()		display_Day Test	
enterSetSection(currentState: int) : int	TimekeepingMode	stopStopwatch()	display_Hour Test		
nextWorld(): void		resetStopwatch()	display_Minute Test		
previousWorld() void		lapTime()	change_Mode Test	DWS controller	
changeValue(currentState: int, button: int) : void		changeValue()	Check Button Test	AlarmMode	
changeMode(currentState: int) : void		removeAlarmNum()	stop Ringing Test 1		
startStopwatch() : StopwatchTime	StopwatchMode	isAlarmIndicator()	stop Ringing Test 2		
stopStopwatch() : StopwatchTime		getAlarmTime()	Ringing Test		
resetStopwatch() : StopwatchTime		resetTimer()	start Stopwatch Test 1	StopwatchMode	
lapTime() : StopwatchTime		changeValue()	start Stopwatch Test 2		
getAlarmTime() : Time	AlarmMode	startTimer()	start Stopwatch Test 3		
removeAlarmNum() : void		stopTimer()	start Stopwatch Test 4		
enterSetSection(currentState : int) : int		changeIndex()	stop Stopwatch Test 1		
checkAlarmArray() : boolean		changeBCLevel()	stop Stopwatch Test 2		
changeValue(currentState: int) : void		getBrightLevel()	stop Stopwatch Test 3		
check3Second() : boolean	Ring	initializeBC()	reset Stopwatch Test		
checkAlarm() : boolean		switchBC()	lap Stopwatch Test 1		
checkTimer() : boolean		checkBC()	lap Stopwatch Test 2		
enterSetSection(currentState : int) : int	TimerMode	dayLong()	select Alarm Num Test	AlarmMode	
startTimer() : Time		selectSW()	Remove Alarm Num Test		
stopTimer() : Time		changeSW()	Set alarm Time Test 1		
resetTimer() : Time			Set alarm Time Test 2		
changeValue(currentState: int) : void			Turn on Alarm Test	TimerMode	
changeIndex() : int	WorldtimeMode		Turn off Alarm Test		
changeWorldTime() : void			Set Timer Time Test		
checkBC() : boolean	BrightcontrolMode		Start Timer Test 1		
switchBC() : void			Start Timer Test 2		
initializeBC() : void			Start Timer Test 3		
changeBCLevel() : int			Stop Timer Test 1		
daylong(currentTime: Time) : int			Stop Timer Test 2		
changeSW() : int	SWMode		Reset Timer Test		
enterChangeSW() : void			Change World Test		WorldtimeMode
selectSW() : int			Turn on Bright Control Test	BrightcontrolMode	
			Turn off Bright Control Test		
			Control Brightness Test		
			Change Brightness Test	SWMode	
			Change SW Test		