

Software Verification

[T1]

201411276 서수빈

201411291 이금락

201411292 이민환

201411313 장진서

CONTENTS

- Specification Review
- Brute Force Test
- Category Partitioning Test
- Pairwise Combination Test

Specification Review

Specification Review

SMA [T1]

오타자

17	임의 방향 생성 시험	- 현재 시각을 See.d 하여 완전히 Random한 1~60 사이의 난수가 생성되는지 Test - 생성된 난수에 해당하는 LCD가 출력되는지 Test	17. Generate Random Direction	R6.1
----	-------------	--	-------------------------------	------

Use Case: Generate Rand. Direct. 1. 사용자가 Random Direction Gecerator 기능을 선택한 상태로 Start 버튼을 눌러 임의의 방향 출력을 요청한다.	Name	1: reqSetTime()
	Responsibilities	사용자가 시간 설정 기능을 요청한다.
	Type	Syetem
	Name	2: setTimePart()
	Responsibilities	사용자가 12/24h 포맷, 분, 시, 요일 등의 부분 설정을 한다.
	Type	Syetem
	Name	3: reqStoreTime()
	Responsibilities	사용자가 설정의 최종 저장을 요청한다.
	Type	Syetem

-> 스펙 설명에서 빈번한 오타가 발생한다

불명확하거나 구체적이지 않은 표현 사용

우리가 늘 타고 다니는 손목시계에 범용성이 높은 난수 생성 기능과 360° 랜덤 방위 출력 기능을 탑재하여 선택이 필요할 때마다 상황에 맞게 단순한 조작만

'시침'은 12분마다 시계방향으로 한 칸씩 이동하며, '남향'은 12시와 '시침'의 중간 지점에 표시된다. 만약 사이의 점 개수가 짝수이면 중간 지점의 양 쪽에 두 개의 점을 표시한다) 시침이 6시를 가리킨다면 어느 쪽을 가리키는지 여부 -> 구체화.

(A) : Actor, (S) : System

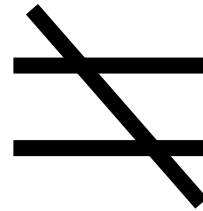
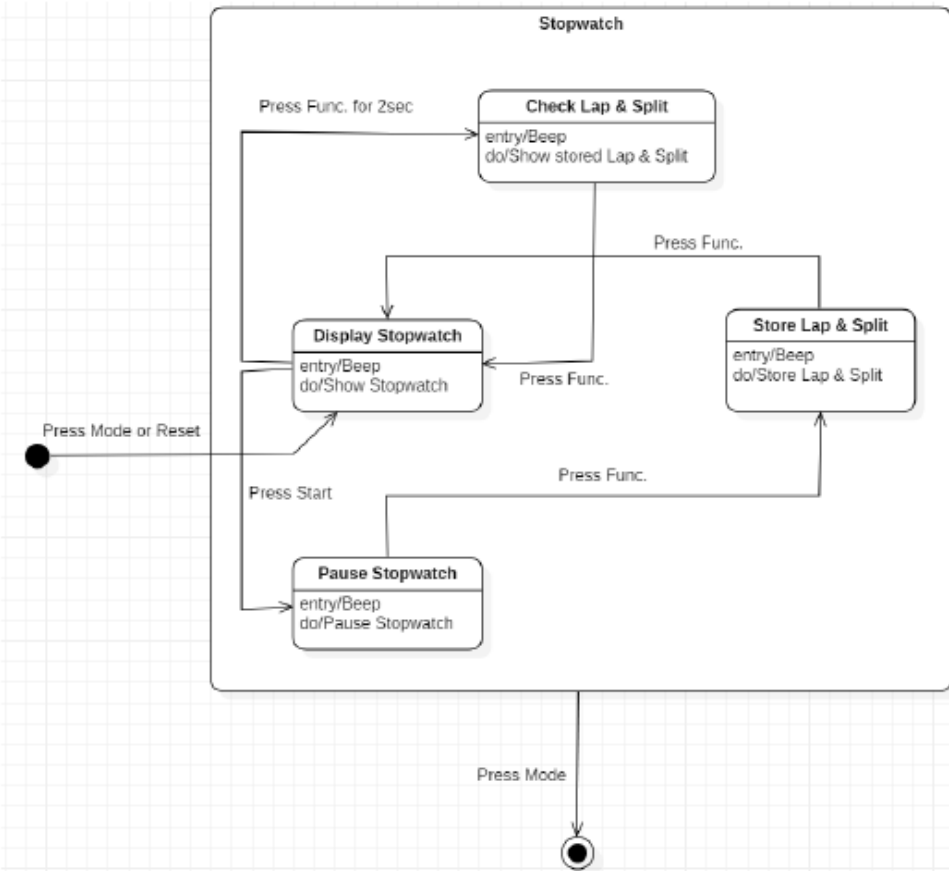
1. (A) : Mode 버튼을 제외한 임의의 버튼을 누른다. -> 예외처리를 추가해야 한다.
2. (S) : Buzzer를 즉시 멈추고 타이머를 초기화 한다.

2. Assign Class Name into Concepts & Draw a Conceptual Class Diagram



Use Case	Set Alarm
Actor	User
Purpose	Alarm을 설정한다.
	<u>알람 데이터가 있을때, 기존 알람의 세부설정이 아닌 새로운 알람을 추가시 어떤 버튼을 눌러야 하는지 설명이 없다</u>
Pre-Requisites	Select Function에 Alarm가 ON 상태여야 한다.
Typical Courses Of Events	<p>(A) : Actor, (S) : System</p> <ol style="list-style-type: none"> 1. (A) : Mode 버튼을 눌러 Alarm모드에 진입한다. 2. (S) : 저장된 알람 리스트의 첫 번째 데이터를 표시한다. 3. (A) : Start(다음)버튼과 Reset(이전)버튼을 이용해서 리스트를 확인한다. (일(날짜)을 표시하는 부분에 알람의 번호를 시간(24h 포맷)과 함께 표시한다. 또한 데두리의 LCD 중 12시부터 7개의 점들이 순서대로 월화수목금토일에 대한 알람 수행 여부를 나타내며, 6시의 점이 알람의 ON/OFF 여부를 나타낸다.) 4. (S) : 입력에 맞는 데이터를 화면에 표시한다. 5. (A) : Funct 버튼을 2초간 누른다. 6. (S) : 현재 표시된 알람의 세부를 설정하는 화면을 표시한다. 설정 중인 부분을 깜빡인다. 7. (A) : 알람 ON/OFF, 분(1의자리), 분(10의자리), 시(1의자리), 시(10의자리), 요일(월화수목금토일) 순서대로 각각을 설정하게 되며 Start 버튼이 +(ON), Reset 버튼이 -(OFF)의 기능을 하고 Funct 버튼을 눌러 다음 설정으로 넘어갈 수 있다. 8. (S) : User의 입력에 맞는 설정을 제공한다. 9. (A) : Funct 버튼을 2초간 눌러 설정을 저장하거나, Reset 버튼을 2초간 눌러 알람을 리스트에서 삭제한다. 10. (S) 입력에 맞게 알람을 저장 또는 삭제한다.
Alternative Courses Of Events	N/A
Exceptional Courses Of Events	E2. 저장된 데이터가 없는 경우 바로 세부설정 모드를 실행한다. E10. 알람 리스트는 Alarm 기능을 선택하지 않아도 삭제되지 않는다.

다이아그램 - 실제 구현과 다른 플로우 형태를 가짐



11. Set Alarm

User

사용자가 버튼을 조작해 알람을 설정한다.

요일과 시간을 최대 10개 까지 메모리에 저장할 수 있으며 알람, 삭제, 추가가 가능하고 알람 기능을 껐다 켜도 메모리는 초기화되지 않는다.

Alarm 기능을 선택하면 현재 저장된 Start 버튼이 +(다음), Reset 버튼이 -하는 부분에 알람의 번호를 시간(24)의 LCD 중 12시부터 7개의 점들이 램 수행 여부를 나타내며, 6시의 점이 원하는 알람을 선택한 상태에서 Func 세부설정 기능으로 넘어간다.

세부설정 기능이 켜진 상태에서 사용자는 알람 ON/OFF, 분(1의자리), 분(10의자리), 시(1의자리), 시(10의자리), 요일(월화수목금토일) 순서대로 각각을 설정하게 되며 현재 설정 중인 부분은 점멸한다.

설정 시 Start 버튼이 +(ON), Reset 버튼을 눌러 다음 설정으로 넘어갈 수 있고 설정 중 언제든지 Func 버튼을 2초로 돌아가며 Reset 버튼을 2초간 누르면 Alarm 기능 진입 시 알람 리스면으로 넘어간다.

7. Store Lap & Split

User

사용자가 버튼을 누르면 메모리에 Lap, Split 시간을 저장한다.

데이터는 최대 30개 까지 저장할 수 있다.

스톱워치가 시작된 상태에서 사용자가 Func 버튼을 누르면 Lap Time과 Split Time을 메모리에 순차적으로 저장한다.

8. Pause Stopwatch

User

스톱워치가 시작된 상태에서 사용자가 Start 버튼을 누르면 스톱워치를 일시 정지하고 다시 Start 버튼을 누르면 멈춘 시간부터 재개한다.

9. Check Lap & Split

User

사용자가 버튼을 조작해 Lap, Split 시간을 확인한다.

인디케이터를 통해 몇 번째 데이터인지 알 수 있으며 Lap과 Split Time 각각을 순차적으로 확인할 수 있다.

스톱워치를 일시정지한 상태에서 Func 버튼을 2초간 누르면 메모리에 저장된 Lap Time과 Split Time을 확인할 수 있다.

일(날짜)을 표시하는 부분에 현재 알람중인 Lap & Split의 번호를 표시하며 Start 버튼이 +(다음), Reset 버튼이 -(이전)의 기능을 하고

1번 Lap -> 1번 Split -> 2번 Lap -> 2번 Split -> ...의 순서로 저장된 시간을 표시한다. 마지막 데이터 이후에는 다시 1번부터 반복된다.

언제든지 Func 버튼을 누르면 스톱워치 화면으로 돌아간다.

일관성 문제 – 기존에 없던 내용이 추가되어 있음

Association Category	Associations
	Select Function – Time Keeping Select Function – Timer
	Select Function – Random Number Generator
	<div style="border: 2px solid red; padding: 5px; display: inline-block;"> Stopwatch 1 has 1 Time </div>
A refer to B	Time Keeping – South Finder
A has B	Time Keeping – Time South Finder – Time Stopwatch – Lap & Split Lap & Split – Time Timer – Time Alarm – Alarm Properties Alarm Properties – Time Random Number Generator – Time Random Direction Generator – Time

4번의 "Stopwatch has Time"이 3번에서는 언급되고 있지 않음

4. Alarm

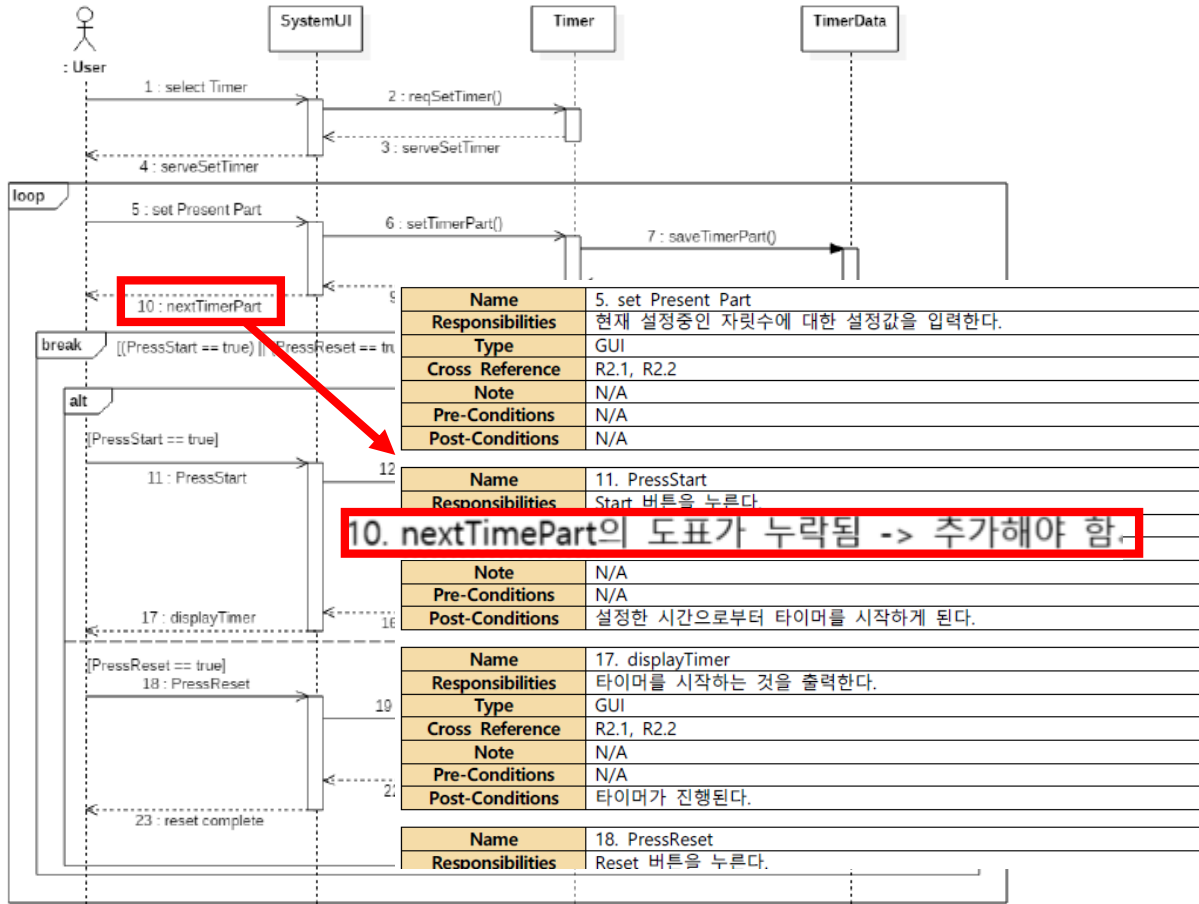


- ▶ 알람 리스트의 3번째 알람 정보를 열람 중인 모습
24h 포맷을 사용하며, 월, 화, 목요일 21시 30분으로 설정되어 있고
(12시 방향부터 7개의 LCD가 일~토요일을 의미)

알람이 ON 되어있는 상태(6시 방향의 LCD가 ON/OFF를 의미)

→ 기존에 없던 새로운 기능이 생김 : 업데이트가 필요함.

자료의 누락 및 중복



Name	11. displayLapSplit
Responsibilities	LapSplitList의 정보를 순서에 맞게 출력한다.
Type	GUI
Cross Reference	R3.4
Note	N/A
Pre-Conditions	Lap, Split 데이터를 열람중인 상태여야 한다.
Post-Conditions	N/A

Name	7. PressStart
Responsibilities	Start 버튼을 누른다.
Type	GUI
Cross Reference	R3.4
Note	N/A
Pre-Conditions	Lap, Split 데이터를 열람중인 상태여야 한다.
Post-Conditions	N/A

Name	N/A
Pre-Conditions	스톱워치가 일시정지 상태여야 한다.
Post-Conditions	N/A

Name	6. displayLapSplit
Responsibilities	LapSplitList의 정보를 순서에 맞게 출력한다.
Type	GUI
Cross Reference	R3.4
Note	N/A
Pre-Conditions	스톱워치가 일시정지 상태여야 한다.
Post-Conditions	N/A

Name	7. PressStart
Responsibilities	Start 버튼을 누른다.
Type	GUI
Cross Reference	R3.4
Note	N/A
Pre-Conditions	Lap, Split 데이터를 열람중인 상태여야 한다.
Post-Conditions	N/A



Specification Review

Specification Review

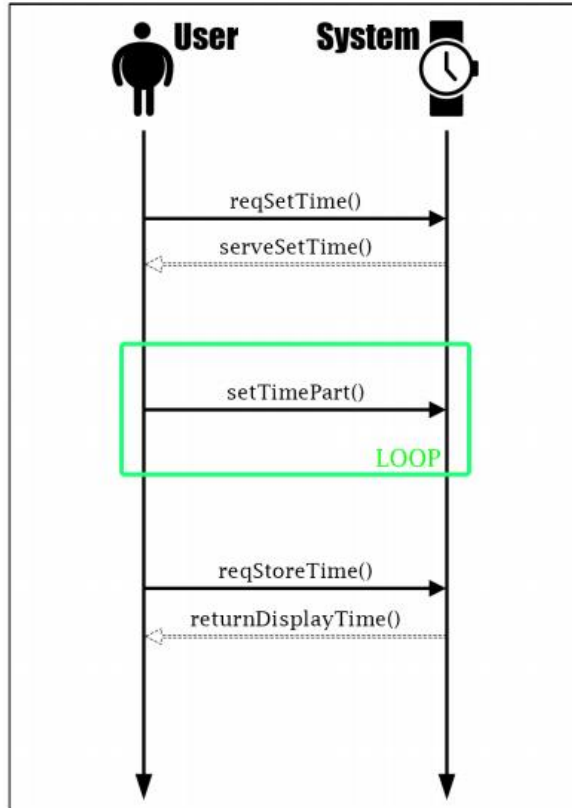
SMA [T2]

Specification Review

Team 1

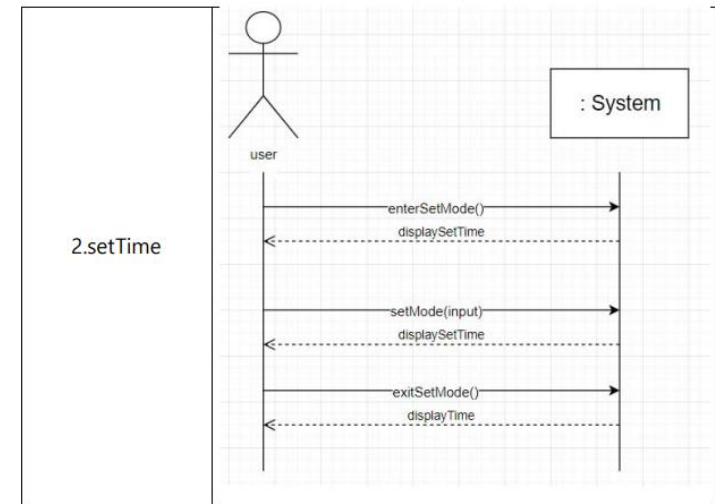
Use Case: Set Time

1. 사용자가 Time Keeping 기능에서 Funct 버튼을 2초간 눌러 시간 설정 기능을 요청한다.
2. 시스템이 시간 설정 기능을 제공한다.
3. 12/24h 포맷, 분, 시, 일, 월, 요일에 대한 부분별 설정을 반복한다.
4. 사용자는 설정 도중 언제든지 Funct 버튼을 2초간 눌러 설정의 저장을 요청한다.
5. 시스템은 새로운 설정을 저장하고 시간을 띄우는 화면으로 전환시킨다.



Team 2

Activity 2035. Define System Sequence Diagrams

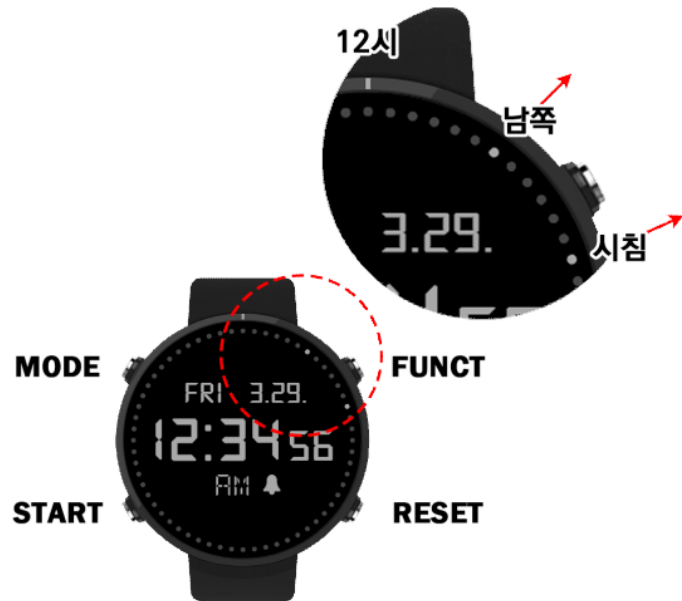


Specification Review

Team 1

2042. Design Reports, UI, and Storyboards

0. Buttons and Indicators Overview



- ▶ Mode, Funct, Start, Reset 4개의 버튼
- 태그키와 5개의 LCD/BDC, South, Find, On/Off Indicator, E)

Team 2

Activity 2042. Define Reports, UI, and Storyboards

2. 시계 UI



2.1 스톱워치 모드 실행



Specification Review

Team 1

Use Case	Set Scope
Actor	User
Purpose	난수 생성 범위를 설정한다.
Overview	User가 난수 생성에 쓰일 범위를 설정한다.
Type	Evident
Cross Reference	Functional Requirements : R.5.1 R.5.2
Pre-Requisites	Select Function에 Random Number Generator가 ON 상태여야 한다.
Typical Courses Of Events	(A) : Actor, (S) : System 1. (A) : Random Number Generator 모드에 진입한다. 2. (S) : 범위 설정 기능을 제공한다. 3. (A) : Start(+)버튼과 Reset(-)버튼을 이용해서 범위를 설정한다. 구간의 설정이 완료되면 Funct로 다음 자릿수 설정으로 넘어간다. ->시간 표시부의 6자리로 1부터 999999까지 범위를 정할 수 있다. 4. (S) : 설정 중인 구간의 숫자를 깜빡인다.
Alternative Courses Of Events	N/A
Exceptional Courses Of Events	N/A

Team 2

Use Case	26.ringRR
Actor	System
Purpose	롤렛게임의 벨을 올린다
Overview	롤렛게임의 확률값에 걸리면 벨을 올린다
Type	Hidden
Cross Reference	Functions : R 6.1, R 6.3, R 8.1 Use cases : showRR, playRR, changeMode
Pre-Requisites	playRR가 실행중이어야 한다.
UI Widgets	Window-1
Typical Courses of Events	(A) : Actor, (S) : System 1. (S) 버저를 올린다.

불명확하거나 구체적이지 않은 표현 사용

11.setAlarm	User	- 사용자가 확인 버튼을 2 초간 눌러 알람시간을 설정모드를 실행하여 알람시간을 설정할 수 있다. <u>setAlarm</u> -> 기존에 저장되어 있는 알람을 수정하는 것인지, 아님 아예 새로운 알람을 만드는 것인지 불명확.
--------------------	-------------	--

Use Case	3.showSW
Actor	System
Purpose	User 에게 스톱워치 현재상태를 보여준다.
Overview	User 에게 스톱워치 현재상태를 보여준다.
Type	Hidden
Cross Reference	Functions : R 2.2, R 2.3, R 2.4, R 2.5, R 8.1 Use cases : startSW, saveRecords, stopSW, resetSW, changeMode
Pre-Requisites	N/A Pre-Requisites에 스톱워치모드가 실행중이어야 한다.
Typical Courses of Events	(A) : Actor, (S) : System 1. (S) User 에게 스톱워치 현재상태를 보여준다.

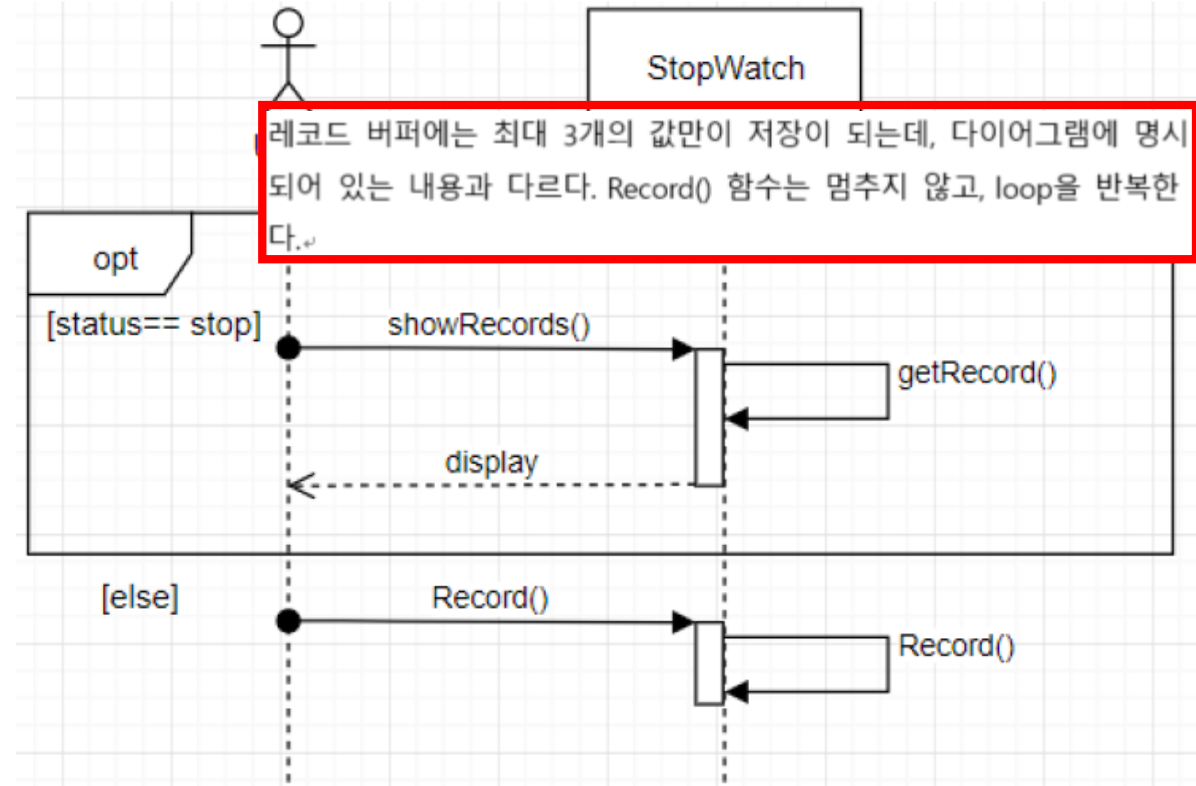
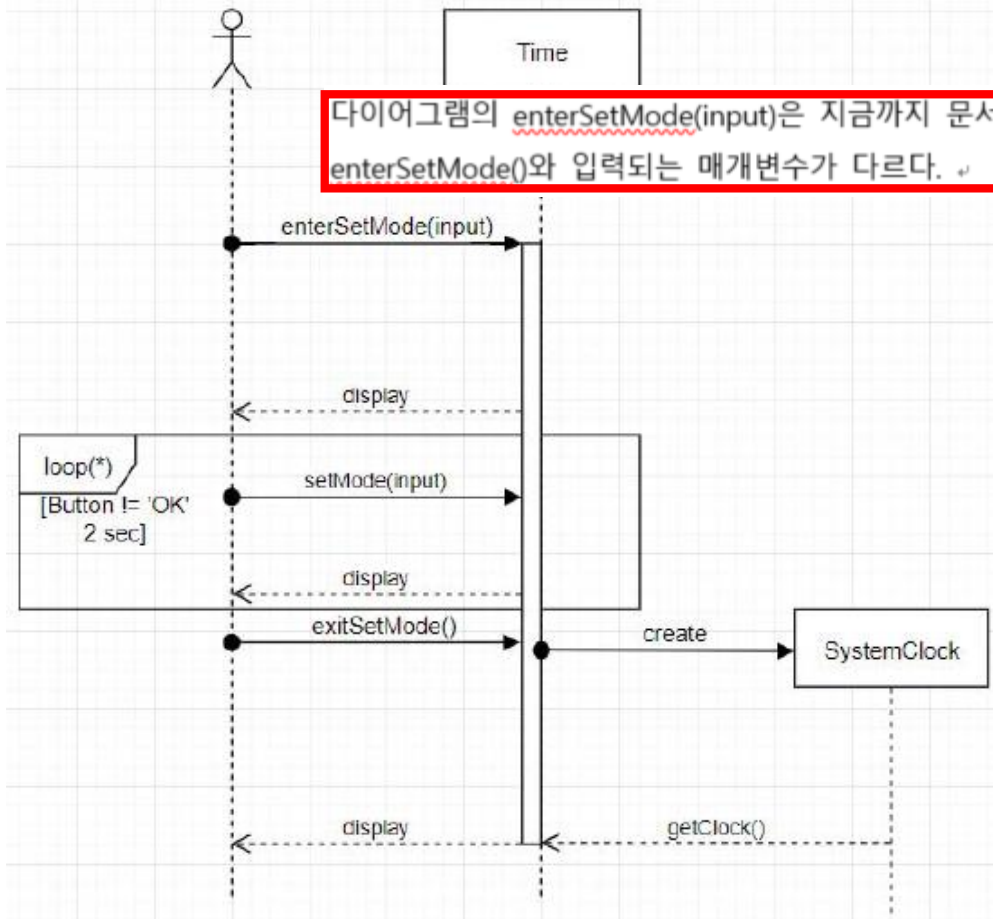
Use Case	19.ringTimer
Actor	System
Pre-Requisites	ringTimer -> 벨이 몇 초 동안 울리는지, 계속 울린다면 끄는 방법은 무엇인지 구체화 필요.
Type	Hidden
Cross Reference	Functions : R 4.1, R 4.3 Use cases : showTimer, startTimer
Pre-Requisites	타이머가 startTimer 가 실행된 상황이어야 한다.
Typical Courses of Events	(A) : Actor, (S) : System 1. (S) 작동중인 타이머가 0 초가 되면 버저를 울린다.
Alternative Courses of Events	N/A
Exceptional Courses of Events	N/A

불명확하거나 구체적이지 않은 표현 사용

Use Case	26.ringRR
	<u>ringRR</u> -> 벨이 몇 초 동안 울리는지, 계속 울린다면 끄는 방법은 무엇인지 구체화 필요.
Overview	플렛게임의 확률값에 걸리면 벨을 울린다
Type	Hidden
Cross Reference	Functions : R 6.1, R 6.3 Use cases : showRR, playRR
Pre-Requisites	playRR 가 실행중이어야 한다.
Typical Courses of Events	(A) : Actor, (S) : System 1. (S) 버저를 울린다.
Alternative Courses of Events	N/A
Exceptional Courses of Events	N/A

Use Case	28.selectMenu
Use Case	29.checkMenu
Use Case	30.changeMode
	<u>selectMenu</u> -> <u>selectMenu</u> 를 실행하는 버튼은 무엇인지 기술 필요. 만약, <u>showMenu</u> 상태에서 <u>selectMenu</u> 를 하는 것이라면 두 기능 간 화면 출력 내용이 다르므로 <u>통일해야함</u> .
	<u>checkMenu</u> -> 활성화 된 메뉴가 4개일 때 무조건 종료가 된다면 많은 문제점이 있으므로 수정 필요. 그런 기능이 아니라면, 조금 더 자세하게 기술 바람.
	<u>changeMenu</u> -> 각 모드마다 버튼들이 의미하는게 다르므로, 절대적인 버튼 명칭으로 기술 요망.

다이어그램 – 구현한 기능과 다른 형태를



일관성 문제 – 기존에 없던 내용이 추가되어 있거나, 기존의 내용과 다름

Name	changeTime(Button : Integer)	
Req Name	showRecords()	
Type Req Name	changeAlarm(Button : Integer)	
Client Type Req Name	changeTimer(time: Integer)	
Req Client Type Req Name	changeDice(Button: Integer)	
Req Name Client Type Req Name	changeRR(Button: Integer)	
Ext Req Name	Responsibilities	롤렛의 확률을 설정한다
Obj Ext Notes	Type	System
Pr Obj Excep	Cross	System functions:R6.2
Pr Obj Output References	Use Case: 24	
Pr Pre-conditions	설정모드이다	
Pr Post-conditions	None	

changeTime -> setTime 이란 이름이 갑자기 변경.↵

showRecords -> saveRecords에서 stopSW일 때 기능이라고 기술했지만 갑자기 나누어 버림.↵

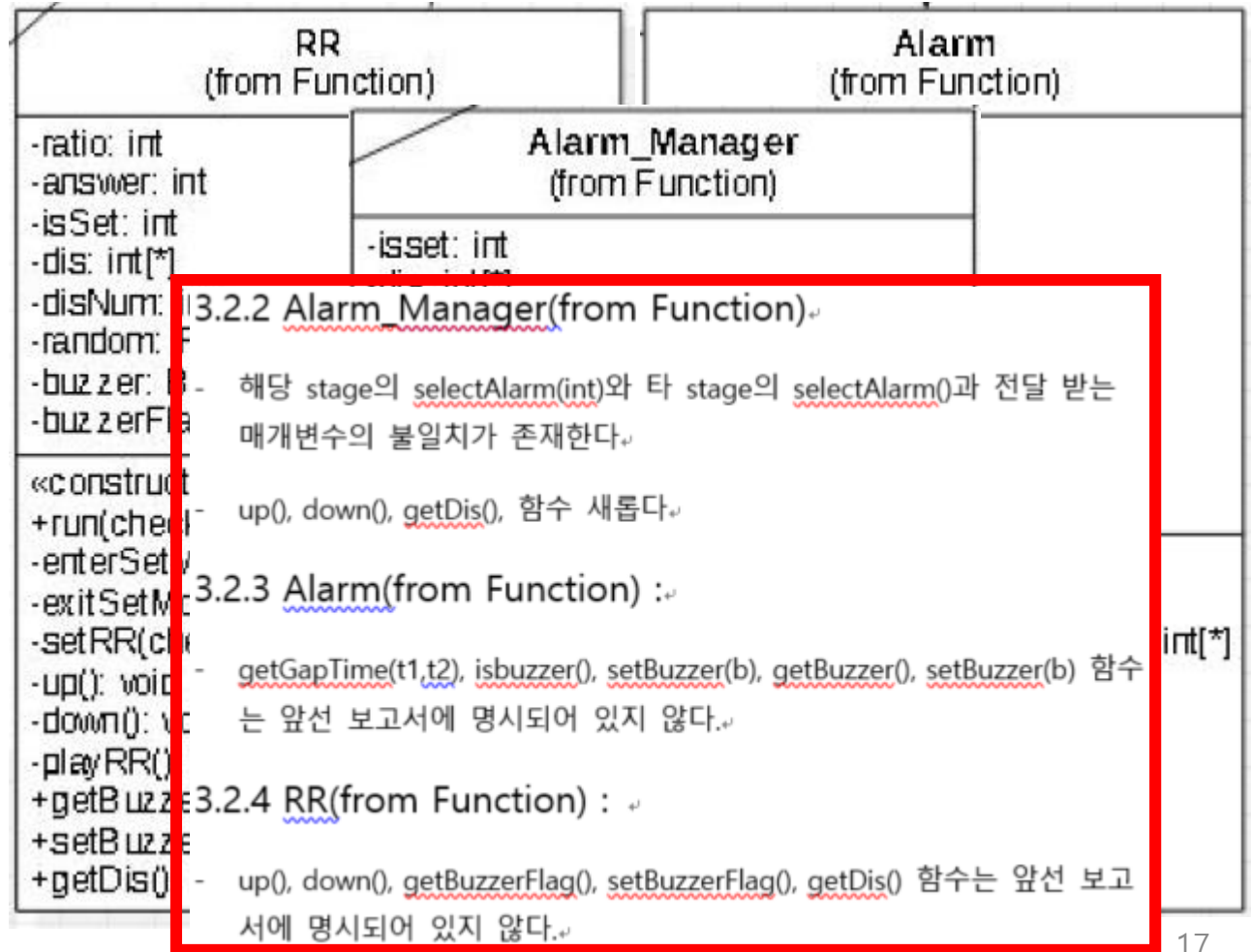
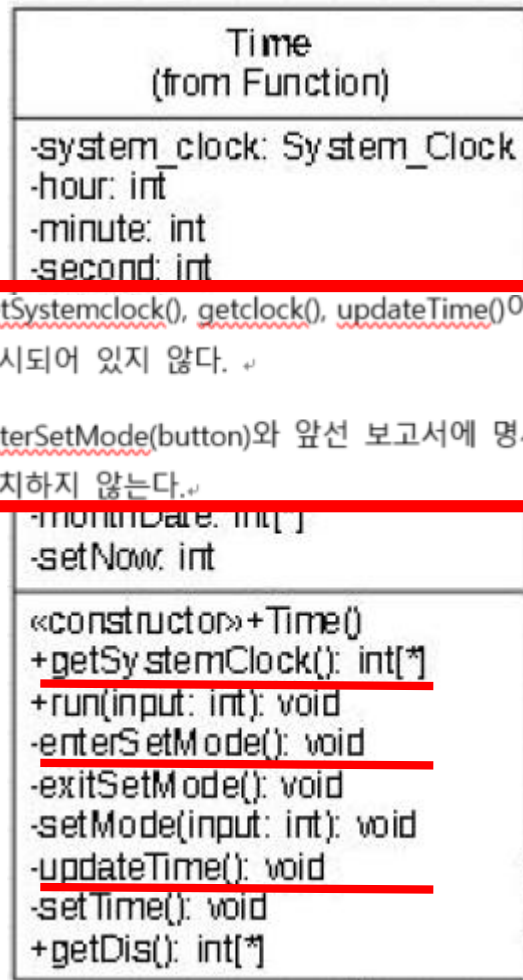
changeAlarm -> setAlarm 이란 이름이 갑자기 변경.↵

changeTimer -> setTimer 이란 이름이 갑자기 변경.↵

changeDice -> setDice 이란 이름이 갑자기 변경.↵

changeRR -> setRR 이란 이름이 갑자기 변경.↵

일관성 문제 – 기존에 없던 내용이 추가되어 있거나, 기존의 내용과 다름



일관성 문제 – 기존에 없던 내용이 추가되어 있거나, 기존의 내용과 다름

<p>Select_Mode (from Manage)</p> <p>-mode: int -now: int -count: int -display: int</p> <p>3.2.5 <u>Select_Mode</u>(from Manage) : - SelectMode()라는 함수는 앞선 보고서에 명시되어 있지 않다.</p> <p>3.2.6 <u>Stop_Watch</u>(from Function) : - updateSW(), pauseSW(), resetSW() 함수는 앞선 보고서에 명시되어 있지 않다. - showRecord()와 showRecords()의 철자가 앞선 보고서에 명시되어 있는 것과 동일하지 않다.</p> <p>«CONSTRUCTOR»+Select_Mode() +run(): void +setMode(mode: int): void +selectMode(mode: int): void +nextMode(): void +previousMode(): void +check(): void +getDis(): int[*]</p>	<p>Stop_Watch (from Function)</p> <p>-sc: System_Clock</p> <p>-Record(): void -getRecord(): int[*] -updateSW(): void +run(check: int): void -showRecord(): void -startSW(): void -pauseSW(): void -resetSW(): void +getDis(): int[*]</p>	<p>Timer (from Function)</p> <p>-setHour: int -setMinute: int -setSec: int -hour: int -minute: int -sec: int -sc: System_Clock2 -t: Thread -isSet: int -isRun: int -setNow: int -dis: int[*] -buzzerFlag: int -buzzer: Buzzer</p> <p>«CONSTRUCTOR»+Timer() +run(input: int): void -enterSetMode(): void -exitSetMode(): void -setMode(input: int): void +getBuzzerFlag(): int +setBuzzerFlag(): void +setBuzzerFlag(Buzzer): void +setBuzzerFlag(flag: int): void +updateTimer(): void +startTimer(): void +pauseTimer(): void +resetTimer(): void +getDis(): int[*]</p>	<p>Random_Number_Generator (from System_Support)</p> <p>-modeNum: int -random: Random</p> <p>«CONSTRUCTOR»+Random_Number_Generator() +setModeNum(mNum: int): void +getRandom(): int</p>
		<p>3.2.7 <u>Timer</u>(from Function) : - getBuzzer(), setBuzzer(), setBuzzer(flag), setBuzzerFlat(flag), updateTimer(), pauseTimer(), getDis() 함수는 앞선 보고서에 명시되어 있지 않다.</p> <p>3.2.8 <u>Random_Number_Generator</u>(from System_Support): - setModeNum(mNum) 함수는 앞선 보고서에 명시되어 있지 않다. - getRandom() 함수는 앞선 보고서에 명시되어 있는 함수의 매개변수와 일치하지 않는다.</p>	

자료의 오류 및 깨짐, 열의 불일치

operation in SSD	operation in interaction diagram	Method	class
1: enterSetMode - 1,18,19,28,37,42	enterSetMode()-1 :1	enterSetMode() : void :1	Time
2: exitSetMode - 3,21,22,30,39,44	changeTime(Button)-2 :2	changeTime(Button : Integer) : void :2	
3: changeTime-2	exitSetMode()-3 :3	exitSetMode() : void :3	
4: startSW- 4,5	startSW()-4 :4	startSW() : void :4	StopWatch
5: saveRecords- 8,9	createClock()-5 :44	showRecords() : void :5	
6: showRecords- 6,7	showRecords()-6 :5	nextRecord() : void :6	
7: stopSW- 10	nextRecord()-7 :6	saveRecords() : void :7	
8: resetSW-	saveRecords()-8	saveTimeToRecord() :	

operation in SSD	operation in interaction diagram	Method	class	Unit Test
1: enterSetMode - 1,18,19,28,37,42	enterSetMode()-1 :1	enterSetMode() : void :1	Time ;1	getInform ; 7
2: exitSetMode - 3,21,22,30,39,44	changeTime(Button)-2 :2	changeTime(Button : Integer) : void :2		activate ;7
3: changeTime- 2	exitSetMode()-3 :3	exitSetMode() : void :3		setDice ;5
4: startSW- 4,5	startSW()-4 :4	startSW() : void :4	StopWatch ;2	testchange DiceNumber ;5
5: saveRecords- 8,9	createClock()-5 :44	showRecords() : void :5		rollDice ;5
6: showRecords- 6,7	showRecords()-6 :5	nextRecord() : void :6		getDis ;5
7: stopSW- 10	nextRecord()-7 :6	saveRecords() : void :7		getBuzzerFlag ;6
8: resetSW-	saveRecords()-8	saveTimeToRecord() :		

Brute Force Test

Brute Force Test

2037. Refine System Test Case

Test #	Test 항목	Description	Use Case	Sys. Func.
1	시간 설정 시험	- User에게 시간을 입력 받는 기능 Test - 입력 후 저장되었는지 Test	1. Set Time	R1.1
2	시간 출력 시험	- 저장되어 있는 시간이 LCD에 출력되는지 Test - Southfinder가 정확한 방향을 가리키고 있는지 Test	2. Display Time	R1.2
3	타이머 설정 시험	- User에게 Timer 설정을 입력 받는 기능 Test - 입력 후 저장되었는지 Test	3. Set Timer	R2.1
4	타이머 버저 시험	- Timer가 0이되면 5초동안 버저를 울리는지 Test	4. Beep Timer Buzzer	R2.2
5	타이머 버저 중지 시험	- Timer의 버저가 울릴 때 User가 버튼을 누르면 버저가 정지하는지 Test - Timer가 초기화 되는지 Test	5. Stop Timer Buzzer	R2.3
6	스톱워치 시험	- User에게 Stopwatch 설정을 입력 받는 기능 Test - 입력 후 성공적으로 저장되었는지 Test	6. Start Stopwatch	R3.1
7	랩 스플릿 시험	- User가 버튼을 누를 때 마다 Lap, Split이 저장되는지 Test - 최대 30개까지 저장이 되는지 Test - 30개가 넘어가면 저장이 안되는지 Test	7. Store Lap & Split	R3.2
8	스톱워치 중지 시험	- User에게 스톱워치 일시정지를 입력받으면 정지하는지 Test - User에게 스톱워치 재개를 요청받으면 재개하는지 Test - User에게 버튼을 입력받아 저장되	8. Pause Stopwatch	R3.3



1. Brute Force Test

Ref.	UseCase	detail	결과
R1.1	Set Time	User에게 12h, 12시 4분 입력 입력 후 저장	↕
R1.2	Display Time	저장되어 있는 시간이 LCD에 출력 Southfinder에 1시, 12시 사이를 가리키는 지	↕
R2.1	Set Timer	1시 2분 22초 대입	↕
R2.2	Beep Timer Buzzer	Timer가 0이되면 5초동안 버저를 울리는 지	↕
R2.3	Stop timer Buzzer	Timer의 버저가 울릴 때 User가 mode버튼을 눌렀을 때 버저가 정지하는 지 Timer가 0으로 초기화 되는지	↕
R3.1	Stop Stopwatch	Stopwatch에서 start버튼 누르기 입력 후 성공적으로 저장 되었는지	↕
R3.2	Store Lap & Split	버튼을 누를 때 마다 Lap, Split이 저장되는지	↕

Activity 2038. Refine System Test Case

Test Num	Test 항목	Description	Use case	System Function
1	시간 표시 실험	시간이 1 초씩 증가하는지 TEST	1. showTime	R 1.1
2	시간 설정 실험	시간을 설정할 수 있는지 TEST	2. setTime	R 1.2
3	스톱워치 표시 실험	스톱워치의 상태가 표시되는지 TEST	3. showSW	R 2.1
4	스톱워치 시작 실험	스톱워치가 시작하는지 TEST	4. startSW	R 2.2
5	시간 저장 실험	스톱워치의 현재 시간이 저장되었는지 TEST	5. saveRecords	R 2.3



1. Brute Force Test

Ref.	UseCase	detail	결과
R1.1	ShowTime	1분 경과 시간을 제대로 출력하는가. 1시간 경과 시간을 제대로 출력하는가.	
R1.2	Set time	1시간 14분 22초 시각을 설정할 수 있는가. 4분 2초 시각을 설정할 수 있는가. 4초 시각을 설정할 수 있는가.	
R2.1	Stopwatch	스탑워치가 제대로 시작 하는가. 1시간 경과 스톱워치가 현재 시각을 제대로 저장하는 가.	
R2.2	save records	현재 시간이 저장되었는 지.	

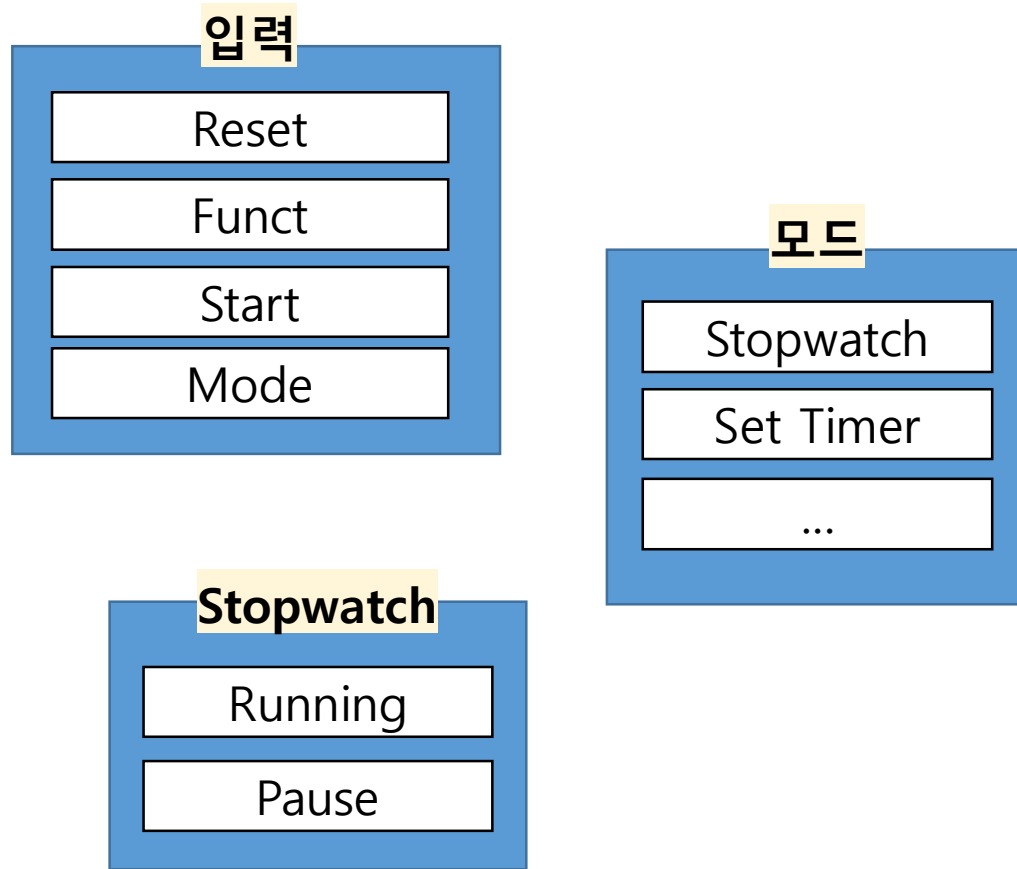
Category Partitioning Test

Category Partitioning Test

Testable Features↵

category↵	values↵
모드↵	time keeping↵
	set time↵
	set timer↵
	stopwatch↵
	set alarm↵
	set scope↵
	random direction generator↵
	select function↵
	alarm ring↵
	buzzer ring↵
입력↵	reset↵
	funct↵
	start↵
	mode↵

Category Partitioning Test

Representative Values[↵]

category [↵]	values [↵]	number [↵]
모드 [↵]	time keeping [↵]	1001 [↵]
	set time [↵]	1002 [↵]
	set timer [↵]	1003 [↵]
	stopwatch [↵]	1004 [↵]
	set alarm [↵]	1005 [↵]
	set scope [↵]	1006 [↵]
	random direction generator [↵]	1007 [↵]
	select function [↵]	1008 [↵]
	alarm ring [↵]	1009 [↵]
	buzzer ring [↵]	1010 [↵]
stopwatch 상태 [↵]	pause [↵]	1021 [↵]
	running [↵]	1022 [↵]
입력 [↵]	reset [↵]	51 [↵]
	funct [↵]	52 [↵]
	start [↵]	53 [↵]
	mode [↵]	54 [↵]
누르는시간 [↵]	2초 < [↵]	2001 [↵]
	2초 >= [↵]	2002 [↵]
포맷 입력 [↵]	12h [↵]	2011 [↵]
	24h [↵]	2012 [↵]
분 입력 [↵]	0 [↵]	2021 [↵]
	1~59 [↵]	2022 [↵]
	60 >= [↵]	2023 [↵]

lap & split 저장	0	2061
	1~9	2062
	10 >=	2063
난수범위	0	2071
	1~999999	2072
	999999 >	2073
기능선택	3개 <	2081
	3개 >=	2082

Total test cases

139,968

Error Constraints 적용

category	values
난수범위	999999 >=
total test case	139,968 -> 93,313

Total test cases

139,968 -> **93,313**

Single Constraints 적용

category	values
포맷입력	12h
	24h
분 입력	0
	60 >=
시 입력	0
	60 >=
초 입력	0
	60 >=
lap & split 저장	0
	10 >=
<u>난수범위</u>	0
total test case	93,313 -> 300

Total test cases

93,313 -> **300**

2.1.5 Property Constraints 적용

category	values	property
모드	time keeping	[property time keeping]
	set time	[property set time]
	set timer	[property set timer]
	stopwatch	[property stopwatch]
	set alarm	[property set alarm]
	set scope	[property set scope]
	random direction generator	[property random direction generator]
	select function	[property select function]
	alarm ring	[property alarm ringing]
	buzzer ring	[property buzzer ringing]
stopwatch 상태	pause	[if stopwatch] [property pause]
	running	[if stopwatch] [property running]
입력	reset	[property reset]
	func	[property func]
	start	[property start]
	mode	[property mode]
<u>누르는시간</u>	2초 <	[if (timekeeping alarm function timer) && (reset <u>func</u> start mode)]

Total test cases

300 -> **95**

Category Partitioning Test

SMA [T1, T2]

Team 1

All combination

139,968

Error Constraints

93,313

Single Constraints

300

Property Constraints

95

Team 2

All combination

124,416

Error Constraints

62,210

Single Constraints

4,617

Property Constraints

73

Pairwise Test

Available Tools

1. CATS (Constrained Array Test System) *)	[Sherwood] Bell Labs.	
2. OATS (Orthogonal Array Test System) *)	[Phadke] ATT	
3. AETG	Telecordia	Web-based, commercial
4. IPO (PairTest) *)	[Tai/Lei]	
5. TConfig	[Williams]	Java-applet
6. TCG (Test Case Generator) *)	NASA	
7. AllPairs	Satisfice	Perl script, free, GPL
8. Pro-Test	SigmaZone	GUI, commercial
9. CTS (Combinatorial Test Services)	IBM	Free for non-commercial use
10. Jenny	[Jenkins]	Command-line, free, public-domain
11. ReduceArray2	STSC, U.S. Air Force	Spreadsheet-based, free
12. TestCover	Testcover.com	Web-based, commercial
13. DDA *)	[Colburn/Cohen/Turban]	
14. Test Vector Generator		GUI, free
15. OA1	k sharp technology	
16. TESTONA	Assystem Germany	GUI, free for non-commercial use
17. AllPairs	[McDowell]	Command-line, free
18. Intelligent Test Case Handler (replaces CTS)	IBM	Free for non-commercial use
19. CaseMaker	Díaz & Hilterscheid	GUI, commercial
20. PICT	Microsoft Corp.	Command-line, open source at http://github.com/microsoft/pict
21. rdExpert	Phadke Associates, Inc.	

모드:time keeping,set time,set timer,stopwatch,set alarm,set scope,random direction generator,function,alarm ring,buzzer ring
 입력:reset,funct,start,mode
 누르는시간 2초:미만,이상
 포맷입력:12h,24h
 분 입력:0,1~59,60이상
 시 입력:0,1~59,60이상
 초 입력:0,1~59,60이상
 lap&split:0,1~9,10이상
 난수범위:0,1~999999,999999이상
 기능선택:미만,이상

```

IF NOT [모드] = "function" THEN [기능선택] = "미만";
IF NOT [모드] = "set scope" THEN [난수범위] = "0";
IF NOT [모드] = "stopwatch" THEN [lap&split] = "0";
IF NOT ([모드] = "set time" OR [모드] = "set timer") THEN [분 입력] = "0";
IF NOT ([모드] = "set time" OR [모드] = "set timer") THEN [시 입력] = "0";
IF NOT ([모드] = "set time" OR [모드] = "set timer") THEN [초 입력] = "0";
IF NOT [모드] = "set time" THEN [포맷 입력] = "12h";
  
```

```
C:\Program Files (x86)\PICT>pict.exe pict_test.txt > pict_result.txt
```

모드	입력	누르는시간	2초	포맷입력	분	입력	시	입력	초	입력	lap&split	난수범위	기능선택
set time	reset	미만	24h	0	0	60이상	0	0	0	0	미만		
set timer	start	이상	12h	60이상	1~59	0	0	0	0	0	미만		
buzzer ring	start	이상	24h	0	0	0	0	0	0	0	0	미만	
set alarm mode	reset	미만	12h	0	0	0	0	0	0	0	미만		
time keeping	reset	이상	12h	0	0	0	0	0	0	0	0	미만	
set time mode	start	이상	12h	1~59	60이상	1~59	0	0	0	0	미만		
stopwatch	mode	미만	24h	0	0	0	0	10이상	0	0	0	미만	
set time	start	미만	24h	1~59	1~59	60이상	0	0	0	0	미만		
set timer mode	start	미만	24h	60이상	60이상	60이상	0	0	0	0	미만		
set alarm funct	start	이상	24h	0	0	0	0	0	0	0	미만		
set scope	reset	미만	24h	0	0	0	0	0	0	0	999999이상	미만	
alarm ring	mode	이상	12h	0	0	0	0	0	0	0	0	미만	
stopwatch	funct	미만	12h	0	0	0	0	1~9	0	0	0	미만	
set timer	reset	미만	24h	60이상	0	1~59	0	0	0	0	미만		
stopwatch	reset	이상	12h	0	0	0	0	10이상	0	0	0	미만	
set time	funct	이상	12h	60이상	60이상	60이상	0	0	0	0	미만		
buzzer ring	reset	미만	12h	0	0	0	0	0	0	0	0	미만	
time keeping	funct	미만	24h	0	0	0	0	0	0	0	0	미만	
time keeping	mode	미만	12h	0	0	0	0	0	0	0	0	미만	
alarm ring	start	미만	24h	0	0	0	0	0	0	0	0	미만	
set scope	mode	이상	12h	0	0	0	0	0	0	0	1~999999	미만	
set scope	start	이상	12h	0	0	0	0	0	0	0	999999이상	미만	
set scope	start	미만	24h	0	0	0	0	0	0	0	0	미만	

Total test cases
64

Team 1

Total test cases

64

Team 2

Total test cases

59

Thank you