

# 소프트웨어공학개론

## System Testing & Unit Testing

T4

김태훈 윤성일 이동현 이대엽

# 1. System Testing

# 1. Test Design Identification

Identifier	Feature	Valid / Invalid value
MWS.STC.000	Time/Temp/Mode 선택	시간, 온도, 모드 선택
MWS.STC.001	Time/Temp/Mode 설정	시간, 온도 증가 및 모드 설정
MWS.STC.002	실행	설정대로 동작 하는지 확인(Time/Temp/Mode 각각 실행)
MWS.STC.003	실행 중 시간, 온도 변화	실행 중 시간, 온도 변화 확인
MWS.STC.004	종료 및 초기화	강제종료/정상종료 시 초기화 상태 확인
MWS.STC.005	Display	현재 시간, 온도 화면 출력 확인
MWS.STC.006	Back Light	문 열림, 작동 시 실행 여부 확인(글자 색 변환)
MWS.STC.007	Beep Sound	완료 후 3초 실행 여부 확인
MWS.STC.008	Time 모드 정상종료 후 재시작	동작 완료 후 재시작 시 오류 확인
MWS.STC.009	Temp 모드 정상종료 후 재시작	동작 완료 후 재시작 시 오류 확인
MWS.STC.010	Mode 상태 정상종료 후 재시작	동작 완료 후 재시작 시 오류 확인
MWS.STC.011	Time 모드 강제종료 후 재시작	동작 완료 후 재시작 시 오류 확인
MWS.STC.012	Temp 모드 강제종료 후 재시작	동작 완료 후 재시작 시 오류 확인
MWS.STC.013	Mode 상태 강제종료 후 재시작	동작 완료 후 재시작 시 오류 확인

# 2. Test Case Identification (000-002)

Test Case Identifier	Input Specification	Output Specification
MWS.STC.000.000	프로그램 실행	Time 모드로 초기화
MWS.STC.000.001	c; c;	Time/Temp 모드 'Time→Temp→Time' 순서로 변경
MWS.STC.000.002	(Temp 모드에서) d	변화 없음
MWS.STC.000.003	d; d;	모드 선택 '떡→죽→밥→국/찌개→피자→Manual→떡' 순서로 변경
MWS.STC.000.004	d; d;	각 모드 별 설정 시간 변경
MWS.STC.000.005	(Manual이 아닌 상태에서) a	변화 없음
MWS.STC.000.006	(Manual이 아닌 상태에서) b	변화 없음
MWS.STC.000.007	(Manual이 아닌 상태에서) c	변화 없음
MWS.STC.000.008	(Manual인 상태에서) c; c;	'Temp→Time→Temp' 순서로 변환
MWS.STC.001.000	(Time 모드에서) a; a;	10sec씩 증가
MWS.STC.001.001	(Time 모드에서) a; a;	10분이 넘어갈 경우 초기화
MWS.STC.001.002	(Time 모드에서) b; b;	30sec씩 증가
MWS.STC.001.003	(Time 모드에서) b; b;	10분이 넘어갈 경우 초기화
MWS.STC.001.004	(Temp 모드에서) a; a;	10℃씩 증가, 초기 온도 20℃
MWS.STC.001.005	(Temp 모드에서) a; a;	90℃가 넘어갈 경우 초기화
MWS.STC.001.006	(Temp 모드에서) b; b;	20℃씩 증가, 초기 온도 20℃
MWS.STC.001.007	(Temp 모드에서) b; b;	90℃가 넘어갈 경우 초기화
MWS.STC.002.000	(문이 열려있는 때) e	변화 없음
MWS.STC.002.001	(Time 모드에서) e	설정대로 정상 실행
MWS.STC.002.002	(Temp 모드에서) e	설정대로 정상 실행
MWS.STC.002.003	(Mode 상태에서) e	설정대로 정상 실행

# 2. Test Case Identification (003-008)

Test Case Identifier	Input Specification	Output Specification
MWS.STC.003.000	(Time 모드에서) 실행 중 버튼 조작이 없을 경우	시간, 온도 일정하게 변화
MWS.STC.003.001	(Temp 모드에서) 실행 중 버튼 조작이 없을 경우	시간, 온도 일정하게 변화
MWS.STC.003.002	(Mode 상태에서) 실행 중 버튼 조작이 없을 경우	시간, 온도 일정하게 변화
MWS.STC.003.003	(Time 모드에서) 실행 중 강제종료 버튼을 제외한 버튼 입력	변화 없음
MWS.STC.003.004	(Temp 모드에서) 실행 중 강제종료 버튼을 제외한 버튼 입력	변화 없음
MWS.STC.003.005	(Mode 상태에서) 실행 중 강제종료 버튼을 제외한 버튼 입력	변화 없음
MWS.STC.004.000	(Time 모드에서) 정상 종료	초기화
MWS.STC.004.001	(Temp 모드에서) 정상 종료	초기화
MWS.STC.004.002	(Mode 상태에서) 정상 종료	초기화
MWS.STC.004.003	(Time 모드에서) 강제 종료 e	초기화
MWS.STC.004.004	(Temp 모드에서) 강제 종료 e	초기화
MWS.STC.004.005	(Mode 상태에서) 강제 종료 e	초기화
MWS.STC.005.000	버튼 조작	과정이 실시간으로 화면에 출력
MWS.STC.005.001	실행 중 e	진행상황 출력
MWS.STC.006.000	문 열림/문 닫힘 f; f;	Back Light On/Off
MWS.STC.006.001	실행 중	Back Light On
MWS.STC.006.002	강제종료/정상종료	Back Light Off
MWS.STC.007.000	정상종료	Beep Sound 출력
MWS.STC.007.001	강제종료	Beep Sound 미출력
MWS.STC.008.000	초기화 후 Time/Temp/Mode 모드 변경 c; d;	버튼 입력 반영
MWS.STC.008.001	Time/Temp/Mode 재설정	버튼 입력 반영
MWS.STC.008.002	실행 e	설정대로 정상 실행
MWS.STC.008.003	정상종료	초기화
MWS.STC.008.004	강제종료 e	초기화

# 2. Test Case Identification (009-013)

Test Case Identifier	Input Specification	Output Specification
MWS.STC.009.000	초기화 후 Time/Temp/Mode 모드 변경 c; d;	버튼 입력 반영
MWS.STC.009.001	Time/Temp/Mode 재설정	버튼 입력 반영
MWS.STC.009.002	실행 e	설정대로 정상 실행
MWS.STC.009.003	정상종료	초기화
MWS.STC.009.004	강제종료 e	초기화
<b>MWS.STC.010.000</b>	초기화 후 Time/Temp/Mode 모드 변경 c; d;	버튼 입력 반영
<b>MWS.STC.010.001</b>	Time/Temp/Mode 재설정	버튼 입력 반영
<b>MWS.STC.010.002</b>	실행 e	설정대로 정상 실행
<b>MWS.STC.010.003</b>	정상종료	초기화
<b>MWS.STC.010.004</b>	강제종료 e	초기화
MWS.STC.011.000	초기화 후 Time/Temp/Mode 모드 변경 c; d;	버튼 입력 반영
MWS.STC.011.001	Time/Temp/Mode 재설정	버튼 입력 반영
MWS.STC.011.002	실행 e	설정대로 정상 실행
MWS.STC.011.003	정상종료	초기화
MWS.STC.011.004	강제종료 e	초기화
<b>MWS.STC.012.000</b>	초기화 후 Time/Temp/Mode 모드 변경 c; d;	버튼 입력 반영
<b>MWS.STC.012.001</b>	Time/Temp/Mode 재설정	버튼 입력 반영
<b>MWS.STC.012.002</b>	실행 e	설정대로 정상 실행
<b>MWS.STC.012.003</b>	정상종료	초기화
<b>MWS.STC.012.004</b>	강제종료 e	초기화
MWS.STC.013.000	초기화 후 Time/Temp/Mode 모드 변경 c; d;	버튼 입력 반영
MWS.STC.013.001	Time/Temp/Mode 재설정	버튼 입력 반영
MWS.STC.013.002	실행 e	설정대로 정상 실행
MWS.STC.013.003	정상종료	초기화
MWS.STC.013.004	강제종료 e	초기화

# 3. Result Identification of the System Test

Identifier	PASS / FAIL	Identifier	PASS / FAIL
MWS.STC.000.000	PASS	MWS.STC.002.001	PASS
MWS.STC.000.001	PASS	MWS.STC.002.002	PASS
MWS.STC.000.002	PASS	MWS.STC.002.003	PASS
MWS.STC.000.003	PASS	MWS.STC.003.000	PASS
MWS.STC.000.004	PASS	MWS.STC.003.001	PASS
MWS.STC.000.005	FAIL	MWS.STC.003.002	PASS
MWS.STC.000.006	FAIL	MWS.STC.003.003	FAIL
MWS.STC.000.007	FAIL	MWS.STC.003.004	FAIL
MWS.STC.000.008	PASS	MWS.STC.003.005	FAIL
MWS.STC.001.000	PASS	MWS.STC.004.000	PASS
MWS.STC.001.001	PASS	MWS.STC.004.001	PASS
MWS.STC.001.002	PASS	MWS.STC.004.002	PASS
MWS.STC.001.003	PASS	MWS.STC.004.003	PASS
MWS.STC.001.004	PASS	MWS.STC.004.004	PASS
MWS.STC.001.005	PASS	MWS.STC.004.005	PASS
MWS.STC.001.006	PASS	MWS.STC.005.000	PASS
MWS.STC.001.007	PASS	MWS.STC.005.001	PASS
MWS.STC.002.000	PASS	MWS.STC.006.000	PASS <sup>7</sup>

# 3. Result Identification of the System Test

Identifier	PASS / FAIL	Identifier	PASS / FAIL
MWS.STC.006.001	PASS	MWS.STC.010.003	PASS
MWS.STC.006.002	PASS	MWS.STC.010.004	PASS
MWS.STC.007.000	FAIL	MWS.STC.011.000	PASS
MWS.STC.007.001	PASS	MWS.STC.011.001	PASS
MWS.STC.008.000	PASS	MWS.STC.011.002	PASS
MWS.STC.008.001	PASS	MWS.STC.011.003	PASS
MWS.STC.008.002	PASS	MWS.STC.011.004	PASS
MWS.STC.008.003	PASS	MWS.STC.012.000	PASS
MWS.STC.008.004	PASS	MWS.STC.012.001	PASS
MWS.STC.009.000	PASS	MWS.STC.012.002	FAIL
MWS.STC.009.001	PASS	MWS.STC.012.003	PASS
MWS.STC.009.002	PASS	MWS.STC.012.004	PASS
MWS.STC.009.003	PASS	MWS.STC.013.000	PASS
MWS.STC.009.004	PASS	MWS.STC.013.001	PASS
MWS.STC.010.000	PASS	MWS.STC.013.002	PASS
MWS.STC.010.001	PASS	MWS.STC.013.003	PASS
MWS.STC.010.002	PASS	MWS.STC.013.004	PASS <sup>8</sup>



## 4. Comment

1. Manual이 아닌 떡, 죽, 밥 등의 모드 상태에서 a 혹은 b 버튼을 누르면 Time이 상승하고, c 버튼을 누르면 Mode가 변경됩니다. 변화가 없도록 수정해야 합니다.
2. Time/Temp/Mode 상태로 실행했을 때, 실행 도중 c버튼을 누르면 Temp 모드로 변화하며 강제종료 됩니다. c 버튼에 대한 반응이 없도록 수정해야 합니다.
3. 정상종료 시에 Beep Sound가 출력되지 않습니다. 출력되도록 수정해야 합니다.
4. Temp 모드로 실행 후 강제종료 했을 때, 예를 들어 온도가 30°C를 가리킬 때 강제종료를 했다고 가정하고, 다시 모드 세팅 후 재시작을 하게 되면 정상 초기 온도인 20°C가 아닌 강제종료 당시 온도인 30°C가 초기 온도가 됩니다. 20°C로 초기화 되도록 수정해야 합니다.
5. 4번의 상태에서 두 번째 실행 중 c 버튼을 누르면 온도가 무한대로 오릅니다. c 버튼에 대한 반응이 없도록 수정해야 합니다.

# 2. Unit Testing

# 1-1. 테스트 할 process (DFD) 리스트

ID	Name	Description
1.1	A Button Interface	A버튼 값 변경
1.2	B Button Interface	B버튼 값 변경
1.3	C Button Interface	C버튼 값 변경
1.4	D Button Interface	D버튼 값 변경
1.5	E Button Interface	E버튼 값 변경
1.6	F Sensor Interface	F센서 값 변경
1.7	Button Input	Interface에 신호 전달
2.1.1	Time Controller	Time Management에 trigger 전달.
2.1.2	Temp Controller	Temp Management에 trigger 전달.
2.1.3	Mode Controller	Mode Management에 trigger 전달.
2.1.4	Menu Controller	Menu Management에 trigger 전달.
2.1.5	Door Controller	Door Management에 trigger 전달.
2.1.6	S_C Controller	S_C Management에 Enable/Disable 전달.
2.1.7	Light Controller	Light Management에 Enable/Disable 전달.
2.1.8	Beep Controller	Beep Management에 Enable/Disable 전달.
2.1.9	Time Management	Enable->Time Management 실행 / Disable->다른 trigger 작동
2.1.10	Time Control A	시간 값 변경(+10s)
2.1.11	Time Control B	시간 값 변경(+30s)
2.1.12	Temp Management	Enable->Temp Management 실행 / Disable->다른 trigger 작동
2.1.13	Temp Control A	온도 값 변경(+10C)
2.1.14	Temp Control B	온도 값 변경(+20C)
2.1.15	Mode Management	시간/온도 선택지 저장
2.1.16	Menu Management	메뉴 저장
2.1.17	Door Management	Enable조건이면 MW 가동, Disable조건이면 MW 가동 중지
2.1.18	S_C Management	Enable조건이면 MW 가동, Disable조건이면 MW 가동 중지
2.1.19	Light Management	Enable/Disable에 따라 Light_Off에 Light Command를 보낸다.
2.1.20	Beep Management	Enable/Disable에 따라 Sound_Off에 Speaker Command를 보낸다.

## 1-2. 테스트 하지 않을 process (DFD) 리스트

ID	Name	Description
2.1.6	S_C Controller	조건문이 있으나 예외 없이 모두 S_C Management을 호출하기 때문에 테스트 의미 없음
2.1.8	Beep Controller	조건문이 있으나 예외 없이 모두 Beep Management을 호출하기 때문에 테스트 의미 없음

## 2. Testing Feature

Identifier	Feature (process ID in DFD)		Valid/ Invaild value
MW.UTC.00.00	1.1	A Button Interface	Button Input 으로부터 신호를 받으면 A_BUTTON_INPUT을 False 에서 TRUE로 바꾼다.
MW.UTC.01.00	1.2	B Button Interface	Button Input 으로부터 신호를 받으면 B_BUTTON_INPUT을 False 에서 TRUE로 바꾼다.
MW.UTC.02.00	1.3	C Button Interface	Button Input 으로부터 신호를 받으면 C_BUTTON_INPUT을 False 에서 TRUE로 바꾼다.
MW.UTC.03.00	1.4	D Button Interface	Button Input 으로부터 신호를 받으면 D_BUTTON_INPUT을 False 에서 TRUE로 바꾼다.
MW.UTC.04.00	1.5	E Button Interface	Button Input 으로부터 신호를 받으면 E_BUTTON_INPUT을 False 에서 TRUE로 바꾼다.
MW.UTC.05.00	1.6	F Sensor Interface	Button Input 으로부터 신호를 받으면 F_SENSOR_INPUT을 False 에서 TRUE로 바꾼다.
MW.UTC.06.00	1.7	Button Input	A버튼 입력이 감지되면 A_Button_Interface에 신호를 전달한다.
MW.UTC.06.01	1.7	Button Input	B버튼 입력이 감지되면 B_Button_Interface에 신호를 전달한다.
MW.UTC.06.02	1.7	Button Input	C버튼 입력이 감지되면 C_Button_Interface에 신호를 전달한다.
MW.UTC.06.03	1.7	Button Input	D버튼 입력이 감지되면 D_Button_Interface에 신호를 전달한다.
MW.UTC.06.04	1.7	Button Input	E버튼 입력이 감지되면 E_Button_Interface에 신호를 전달한다.
MW.UTC.06.05	1.7	Button Input	F센서 입력이 감지되면 F_Sensor_Interface에 신호를 전달한다.
MW.UTC.07.00	2.1.1	Time Controller	tick에 따라 시간을 제어해준다. Trigger를 Time Management에 보내준다.
MW.UTC.08.00	2.1.9	Time Management	Enable일 경우 Time Management 실행
MW.UTC.08.01	2.1.9	Time Management	Disable일 경우(*bsi).time_d.time--
MW.UTC.09.00	2.1.10	Time Control A	(*bsi).time_d.time<=590인 상태에서 시간을 10s 증가시킨다.
MW.UTC.09.01	2.1.10	Time Control A	(*bsi).time_d.time=600인 상태에서 시간을 10s로 바꾼다.
MW.UTC.09.02	2.1.10	Time Control A	(*bsi).time_d.time>590인 상태에서 시간을 600s로 바꾼다.
MW.UTC.10.00	2.1.11	Time Control B	(*bsi).time_d.time<=570인 상태에서 시간을 30s 증가시킨다.
MW.UTC.10.01	2.1.11	Time Control B	(*bsi).time_d.time=600인 상태에서 시간을 30s로 바꾼다.
MW.UTC.10.02	2.1.11	Time Control B	(*bsi).time_d.time>570인 상태에서 시간을 600s로 바꾼다.

## 2. Testing Feature

Identifier	Feature(Process ID in DFD)	Valid/ Invalid value
MW.UTC.11.00	2.1.2 Temp Controller	mode = 0인 상태에서 tick이 들어온다.
MW.UTC.11.01	2.1.2 Temp Controller	mode =1인 상태에서 tick이 들어온다.
MW.UTC.12.00	2.1.12 Temp Management	start_cancel=0, A_BUTTON_INPUT=TRUE인 상태에서 tick과 Trigger가 들어온다.
MW.UTC.12.01	2.1.12 Temp Management	start_cancel=0, B_BUTTON_INPUT=TRUE인 상태에서 tick과 Trigger가 들어온다.
MW.UTC.12.02	2.1.12 Temp Management	start_cancel=1인 상태에서 tick과 Trigger가 들어온다.
MW.UTC.13.00	2.1.13 Temp Control A	temp<=80인 상태에서 tick과 Trigger가 들어온다.
MW.UTC.13.01	2.1.13 Temp Control A	80<temp<90인 상태에서 tick과 Trigger가 들어온다.
MW.UTC.13.02	2.1.13 Temp Control A	temp=90인 상태에서 tick과 Trigger가 들어온다.
MW.UTC.14.00	2.1.13 Temp Control B	temp<=70인 상태에서 tick과 Trigger가 들어온다.
MW.UTC.14.01	2.1.13 Temp Control B	70<temp<90인 상태에서 tick과 Trigger가 들어온다.
MW.UTC.14.02	2.1.13 Temp Control B	Temp=90인 상태에서 tick과 Trigger가 들어온다.
MW.UTC.15.00	2.1.3 Mode Controller	C_bUTTON_INPUT==TURE인 상태에서 tick이 들어온다.
MW.UTC.15.01	2.1.3 Mode Controller	C_bUTTON_INPUT==FALSE인 상태에서 tick이 들어온다.
MW.UTC.16.00	2.1.15 Mode Management	mode=0인 상태에서 tick과 Trigger가 들어온다.
MW.UTC.16.01	2.1.15 Mode Management	mode=1인 상태에서 tick과 Trigger가 들어온다.
MW.UTC.17.00	2.1.18 S_C Management	start_cancel=0, time != 0 인 상태에서 Trigger가 들어온다.
MW.UTC.17.01	2.1.18 S_C Management	start_cancel=0, temp != 20 인 상태에서 Trigger가 들어온다.
MW.UTC.17.02	2.1.18 S_C Management	start_cancel=1, time=0, mode=0 인 상태에서 Trigger가 들어온다.
MW.UTC.17.03	2.1.18 S_C Management	start_cancel=1, temp=b_temp, mode=1 인 상태에서 Trigger가 들어온다.

## 2. Testing Feature

Identifier	Feature(Process ID in DFD)	Valid/ Invailld value
MW.UTC.18.00	2.1.4 Menu Controller	Tick에 따라 메뉴를 제어해준다. Trigger를 Menu management로 보내준다.
MW.UTC.18.01	2.1.4 Menu Controller	mode =1인 상태에서 tick이 들어온다.
MW.UTC.19.00	2.1.16 Menu Management	Menu Controller로부터 Trigger를 받아 조건에 맞춰 변경된 메뉴를 저장하고 변경된 메뉴를 출력하기 위해 Display Command에 정보를 보내준다.
MW.UTC.19.01	2.1.16 Menu Management	"
MW.UTC.19.02	2.1.16 Menu Management	"
MW.UTC.19.03	2.1.16 Menu Management	"
MW.UTC.19.04	2.1.16 Menu Management	"
MW.UTC.19.05	2.1.16 Menu Management	"
MW.UTC.19.06	2.1.16 Menu Management	"
MW.UTC.19.07	2.1.16 Menu Management	"
MW.UTC.19.08	2.1.16 Menu Management	"
MW.UTC.19.09	2.1.16 Menu Management	"
MW.UTC.19.10	2.1.16 Menu Management	"
MW.UTC.20.00	2.1.19 Light Management	Enable, disable에 따라 light_off에 light command를 보냄
MW.UTC.20.01	2.1.19 Light Management	"
MW.UTC.20.02	2.1.19 Light Management	"

## 2. Testing Feature

Identifier	Feature(Process ID in DFD)	Valid/ Invaield value
MW.UTC.20.03	2.1.19 Light Management	Enable, disable에 따라 light_off에 light command를 보낸다
MW.UTC.20.04	2.1.19 Light Management	"
MW.UTC.20.05	2.1.19 Light Management	"
MW.UTC.20.06	2.1.19 Light Management	"
MW.UTC.20.07	2.1.19 Light Management	"
MW.UTC.21.00	2.1.7 Light Controller	Tick에 따라 조명을 제어해준다. Disable, enable를 light management로 보내준다
MW.UTC.21.01	2.1.7 Light Controller	"
MW.UTC.21.02	2.1.7 Light Controller	"
MW.UTC.21.03	2.1.7 Light Controller	"
MW.UTC.21.04	2.1.7 Light Controller	"
MW.UTC.21.05	2.1.7 Light Controller	"
MW.UTC.21.06	2.1.7 Light Controller	"
MW.UTC.21.07	2.1.7 Light Controller	"



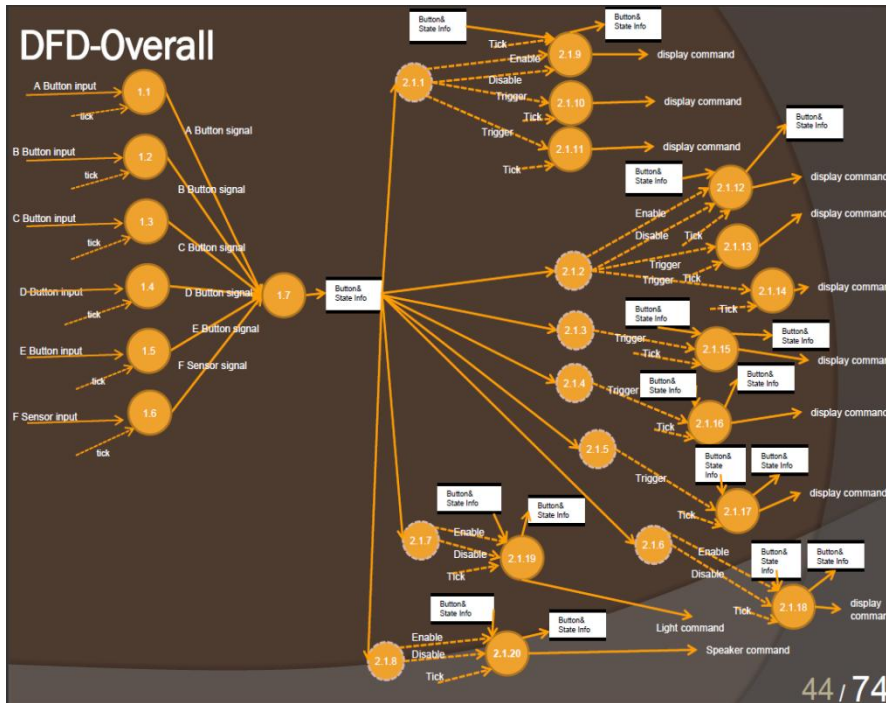
## 2. Testing Feature

Identifier	Feature(Process ID in DFD)	Valid/ Invaild value
MW.UTC.22.00	2.1.19 Light Interface	입력받은 커맨드를 바탕으로 결과값을 화면에 출력해준다.
MW.UTC.22.01	2.1.19 Light Interface	''
MW.UTC.22.02	2.1.19 Light Interface	''
MW.UTC.22.03	2.1.19 Light Interface	''
MW.UTC.23.00	2.1.7 Beep Management	Enable, disable에 따라 sound_off에 speaker command를 보낸다.
MW.UTC.23.01	2.1.7 Beep Management	''
MW.UTC.23.02	2.1.7 Beep Management	''
MW.UTC.23.03	2.1.7 Beep Management	''
MW.UTC.23.04	2.1.7 Beep Management	''
MW.UTC.23.05	2.1.7 Beep Management	''
MW.UTC.23.06	2.1.7 Beep Management	''
MW.UTC.23.07	2.1.7 Beep Management	''
MW.UTC.24.00	2.1.7 Beep Interface	입력받은 커맨드를 바탕으로 결과값을 beep로 출력해준다.
MW.UTC.24.01	2.1.7 Beep Interface	''
MW.UTC.24.02	2.1.7 Beep Interface	''
MW.UTC.24.04	2.1.7 Beep Interface	''

## 2. Testing Feature

Identifier	Feature(Process ID in DFD)	Valid/ Invaild value
MW.UTC.25.00	2.1.5 Door Controller	Tick에 따라 문 개폐를 제어해준다. Trigger를 Door management로 보내준다.
MW.UTC.25.01	2.1.5 Door Controller	"
MW.UTC.26.00	2.1.19 Door Management	enable상태의 조건이 맞으면 전자레인지를 가동한다. disable상태의 조건이 맞으면 전자레인지 가동을 중지한다.
MW.UTC.26.01	2.1.19 Door Management	

### 3. Testing Scenario



- DFD에서 데이터 흐름에 따라 테스트 순서를 정했습니다.
- 버튼입력부 -> Time Controller -> Temp Controller -> Mode Controller -> Menu Controller -> Door Controller -> S\_C Controller -> Light Controller -> Beep Controller 순으로 테스트를 진행했습니다.

# 4. Test Case

Test case identifier	Input specification	Output specification
MW.UTC.00.00	(*bsi).bd.A_BUTTON_INPUT, char c	A_BUTTON_INPUT=TRUE
MW.UTC.01.00	(*bsi).bd.B_BUTTON_INPUT, char c	B_BUTTON_INPUT=TRUE
MW.UTC.02.00	(*bsi).bd.C_BUTTON_INPUT, char c	C_BUTTON_INPUT=TRUE
MW.UTC.03.00	(*bsi).bd.D_BUTTON_INPUT, char c	D_BUTTON_INPUT=TRUE
MW.UTC.04.00	(*bsi).bd.E_BUTTON_INPUT, char c	E_BUTTON_INPUT=TRUE
MW.UTC.05.00	(*bsi).bd.F_SENSOR_INPUT, char c	F_SENSOR_INPUT=TRUE
MW.UTC.06.00	Button_State_Info *bsi	A_Button_Interface 호출
MW.UTC.06.01	Button_State_Info *bsi	B_Button_Interface 호출
MW.UTC.06.02	Button_State_Info *bsi	C_Button_Interface 호출
MW.UTC.06.03	Button_State_Info *bsi	D_Button_Interface 호출
MW.UTC.06.04	Button_State_Info *bsi	E_Button_Interface 호출
MW.UTC.06.05	Button_State_Info *bsi	F_Sensor_Interface 호출
MW.UTC.07.00	(*bsi).mode_d.mode=0	Time Management 호출
MW.UTC.08.00	(*bsi).sbc.start_cancel=0	Time Control A/B 호출
MW.UTC.08.01	(*bsi).sbc.start_cancel=1	(*bsi).time_d.time--
MW.UTC.09.00	(*bsi).time_d.time <= 590	(*bsi).time_d.time+10
MW.UTC.09.01	(*bsi).time_d.time=600	(*bsi).time_d.time=10
MW.UTC.09.02	(*bsi).time_d.time > 590	(*bsi).time_d.time=600
MW.UTC.10.00	(*bsi).time_d.time <= 570	(*bsi).time_d.time+30
MW.UTC.10.01	(*bsi).time_d.time=600	(*bsi).time_d.time=30
MW.UTC.10.02	(*bsi).time_d.time > 570	(*bsi).time_d.time=600

# 4. Test Case

Test case identifier	Input specification	Output specification
MW.UTC.11.000	Tick / (*bsi).mode_d.mode=0	(*bsi).temp_d.temp
MW.UTC.11.001	Tick / (*bsi).mode_d.mode=1	(*bsi).temp_d.temp Trigger/ Temp_Managemet
MW.UTC.12.000	Tick,Trigger in/(*bsi).sbc.start_cancel=0, (*bsi).bd.A_BUTTON_INPUT=TRUE	Trigger/ Temp_Control_A
MW.UTC.12.001	Tick,Trigger in/(*bsi).sbc.start_cancel=0, (*bsi).bd.B_BUTTON_INPUT=TRUE	Trigger/ Temp_Control_B
MW.UTC.12.002	Tick,Trigger in/(*bsi).sbc.start_cancel=1	(*bsi).temp_d.b_temp++
MW.UTC.13.000	Tick,Trigger in/ (*bsi).temp_d.temp	(*bsi).temp_d.temp+=10
MW.UTC.13.001	Tick,Trigger in/ (*bsi).temp_d.temp	(*bsi).temp_d.temp=90
MW.UTC.13.002	Tick,Trigger in/ (*bsi).temp_d.temp	(*bsi).temp_d.temp=20
MW.UTC.14.000	Tick,Trigger in/ (*bsi).temp_d.temp	(*bsi).temp_d.temp+=20
MW.UTC.14.001	Tick,Trigger in/ (*bsi).temp_d.temp	(*bsi).temp_d.temp=90
MW.UTC.14.002	Tick,Trigger in/ (*bsi).temp_d.temp	(*bsi).temp_d.temp=20
MW.UTC.15.000	Tick/ (*bsi).bd.C_BUTTON_INPUT=TRUE	Trigger/ Mode_Management (*bsi).mode_d.mode
MW.UTC.15.001	Tick/ (*bsi).bd.C_BUTTON_INPUT=FALSE	(*bsi).mode_d.mode
MW.UTC.16.000	Tick,Trigger in/ (*bsi).temp_d.temp, (*bsi).mode_d.mode=0,	(*bsi).mode_d.mode=1, (*bsi).time_d.time=0
MW.UTC.16.001	Tick,Trigger in/ (*bsi).temp_d.temp, (*bsi).mode_d.mode=1,	(*bsi).mode_d.mode=0, (*bsi).temp_d.temp=20
MW.UTC.17.000	Tick/ (*bsi).sbc.start_cancel=0, (*bsi).time_d.time != 0	(*bsi).sbc.start_cancel=1, (*bsi).time_d.TimeFLAG = 1, (*bsi).beep_d.beep_flag = 1
MW.UTC.17.001	Tick/(*bsi).sbc.start_cancel=0, (*bsi).temp_d.temp != 20	(*bsi).sbc.start_cancel=0, (*bsi).temp_d.TempFLAG = 1, (*bsi).beep_d.beep_flag = 1
MW.UTC.17.002	Tick/(*bsi).sbc.start_cancel=1, (*bsi).time_d.time=0, (*bsi).mode_d.mode = 0	(*bsi).sbc.start_cancel=0, (*bsi).time_d.TimeFLAG = 0, (*bsi).menu_d.menu=0
MW.UTC.17.003	Tick/(*bsi).sbc.start_cancel=1, (*bsi).temp_d.b_temp, (*bsi).mode_d.mode = 1	(*bsi).temp_d.TempFLAG = 0, (*bsi).sbc.start_cancel=0, (*bsi).temp_d.temp = 20 , (*bsi).temp_d.b_temp = 20

# 4. Test Case

Test case identifier	Input specification	Output specification
MW.UTC.18.00	bsi.bd.D_bUTTON_INPUT=TRUE	Menu Manager 호출
MW.UTC.18.01	bsi.bd.D_bUTTON_INPUT=FALSE	Menu Manager 호출 거절됨
MW.UTC.19.00	mode, menu, time = (0,0,0)	0,1,60
MW.UTC.19.01	mode, menu, time = (0,1,0)	0,2,90
MW.UTC.19.02	mode, menu, time = (0,2,0)	0,3,120
MW.UTC.19.03	mode, menu, time = (0,3,0)	0,4,300
MW.UTC.19.04	mode, menu, time = (0,4,0)	0,5,120
MW.UTC.19.05	mode, menu, time = (0,5,0)	0,0,0
MW.UTC.19.06	mode, menu, time = (1,1,0)	1,1,0
MW.UTC.19.07	mode, menu, time = (1,2,0)	1,2,0
MW.UTC.19.08	mode, menu, time = (1,3,0)	1,3,0
MW.UTC.19.09	mode, menu, time = (1,4,0)	1,4,0
MW.UTC.19.10	mode, menu, time = (1,5,0)	1,5,0
MW.UTC.20.00	start,doorst,lightst = ( 0, 0, 0)	0,0,0
MW.UTC.20.01	start,doorst,lightst = ( 0, 0, 1)	0,0,0
MW.UTC.20.02	start,doorst,lightst = ( 0, 1, 0)	0,1,1
MW.UTC.20.03	start,doorst,lightst = ( 0, 1, 1)	제외
MW.UTC.20.04	start,doorst,lightst = ( 1, 0, 0)	1,0,1
MW.UTC.20.05	start,doorst,lightst = ( 1, 0, 1)	1,0,0
MW.UTC.20.06	start,doorst,lightst = ( 1, 1, 0)	제외
MW.UTC.20.07	start,doorst,lightst = ( 1, 1, 1)	제외

# 4. Test Case

Test case identifier	Input specification	Output specification
MW.UTC.21.00	start,doorst,lightst = ( 0, 0, 0)	호출됨
MW.UTC.21.01	start,doorst,lightst = ( 0, 0, 1)	호출됨
MW.UTC.21.02	start,doorst,lightst = ( 0, 1, 0)	호출됨
MW.UTC.21.03	start,doorst,lightst = ( 0, 1, 1)	제외
MW.UTC.21.04	start,doorst,lightst = ( 1, 0, 0)	호출됨
MW.UTC.21.05	start,doorst,lightst = ( 1, 0, 1)	호출됨
MW.UTC.21.06	start,doorst,lightst = ( 1, 1, 0)	제외
MW.UTC.21.07	start,doorst,lightst = ( 1, 1, 1)	제외
MW.UTC.22.00	start,doorst= ( 0, 0)	검정
MW.UTC.22.01	start,doorst= ( 0, 1)	노랑
MW.UTC.22.02	start,doorst= ( 1, 0)	제외
MW.UTC.22.03	start,doorst= ( 1, 1)	노랑
MW.UTC.23.00	(*bsi).bd.E_BUTTON_INPUT, char c	E_BUTTON_INPUT=TRUE
MW.UTC.23.01	(*bsi).bd.F_SENSOR_INPUT, char c	F_SENSOR_INPUT=TRUE
MW.UTC.23.02	start,b_f,time= (0,0,0)	b_f = 0
MW.UTC.23.03	start,b_f,time= (0,0,1)	b_f = 0
MW.UTC.23.04	start,b_f,time= (0,1,0)	b_f = 1
MW.UTC.23.05	start,b_f,time= (0,1,1)	b_f = 0
MW.UTC.23.06	start,b_f,time= (1,0,0)	b_f = 0
MW.UTC.23.07	start,b_f,time= (1,0,1)	b_f = 0
MW.UTC.24.00	b_s,time = (0,0)	no beep
MW.UTC.24.01	b_s,time = (0,1)	no beep
MW.UTC.24.02	b_s,time = (1,0)	beep
MW.UTC.24.03	b_s,time = (1,1)	no beep
MW.UTC.25.00	bd.F_SEMSOR_INPUT==TRUE	door 호출됨
MW.UTC.25.01	bd.F_SEMSOR_INPUT==FALSE	door 호출 거절됨
MW.UTC.26.00	d_s = TRUE	d_s = FALSE
MW.UTC.26.01	d_s = FALSE	d_s = TRUE

# 5. Test Result

Identifier	PASS/FAIL
MW.UTC.00.00	PASS
MW.UTC.01.00	PASS
MW.UTC.02.00	PASS
MW.UTC.03.00	PASS
MW.UTC.04.00	PASS
MW.UTC.05.00	PASS
MW.UTC.06.00	PASS
MW.UTC.06.01	PASS
MW.UTC.06.02	PASS
MW.UTC.06.03	PASS
MW.UTC.06.04	PASS
MW.UTC.06.05	PASS
MW.UTC.07.00	PASS
MW.UTC.08.00	PASS
MW.UTC.08.01	PASS
MW.UTC.09.00	PASS
MW.UTC.09.01	PASS
MW.UTC.09.02	PASS
MW.UTC.10.00	PASS
MW.UTC.10.01	PASS
MW.UTC.10.02	PASS



# 5. Test Result

Identifier	PASS/FAIL
MW.UTC.11.00	PASS
MW.UTC.11.01	PASS
MW.UTC.12.00	PASS
MW.UTC.12.01	PASS
MW.UTC.12.02	PASS
MW.UTC.13.00	PASS
MW.UTC.13.01	PASS
MW.UTC.13.02	PASS
MW.UTC.14.00	PASS
MW.UTC.14.01	PASS
MW.UTC.14.02	PASS
MW.UTC.15.00	PASS
MW.UTC.15.01	PASS
MW.UTC.16.00	PASS
MW.UTC.16.01	PASS
MW.UTC.17.00	PASS
MW.UTC.17.01	PASS
MW.UTC.17.02	PASS
MW.UTC.17.03	PASS

# 5. Test Result

Identifier	PASS/FAIL
MW.UTC.18.00	PASS
MW.UTC.18.01	PASS
MW.UTC.19.00	PASS
MW.UTC.19.01	PASS
MW.UTC.19.02	PASS
MW.UTC.19.03	PASS
MW.UTC.19.04	PASS
MW.UTC.19.05	PASS
MW.UTC.19.06	PASS
MW.UTC.19.07	PASS
MW.UTC.19.08	PASS
MW.UTC.19.09	PASS
MW.UTC.19.10	PASS
MW.UTC.20.00	PASS
MW.UTC.20.01	PASS
MW.UTC.20.02	PASS
MW.UTC.20.03	PASS
MW.UTC.20.04	PASS
MW.UTC.20.05	PASS
MW.UTC.20.06	PASS
MW.UTC.20.07	PASS

# 5. Test Result

Identifier	PASS/FAIL
MW.UTC.21.00	PASS
MW.UTC.21.01	PASS
MW.UTC.21.02	PASS
MW.UTC.21.03	PASS
MW.UTC.21.04	PASS
MW.UTC.21.05	PASS
MW.UTC.21.06	PASS
MW.UTC.21.07	PASS
MW.UTC.22.00	PASS
MW.UTC.22.01	PASS
MW.UTC.22.02	PASS
MW.UTC.22.03	PASS
MW.UTC.23.00	PASS
MW.UTC.23.01	PASS
MW.UTC.23.02	PASS
MW.UTC.23.03	PASS
MW.UTC.23.04	PASS
MW.UTC.23.05	PASS
MW.UTC.23.06	PASS
MW.UTC.23.07	PASS
MW.UTC.24.00	PASS

# 5. Test Result

Identifier	PASS/FAIL
MW.UTC.24.01	PASS
MW.UTC.24.02	PASS
MW.UTC.24.03	PASS
MW.UTC.25.00	PASS
MW.UTC.25.01	PASS
MW.UTC.26.00	PASS
MW.UTC.26.01	PASS

## 6. Comment

- DFD와 소스코드가 내용이나 구성이 다를 경우는 소스코드를 우선으로 보고 테스트를 수행했습니다.
- DFD에서는 ID 1.1~1.6이 처리를 한 후 signal을 1.7에 보내 button&state info에 값을 저장한다고 되어있는데 소스코드를 살펴보면 1.7에서 1.1~1.6이 받아야 할 입력들을 모두 받은 후 1.1~1.6함수들을 형식적으로만 호출 합니다. DFD와 같게 구현하려면 1.7에서 쓰인 GetAsyncKeyState함수를 1.1~1.6에 각각 옮겨주고 1.1~1.6에서 1.7을 호출하게 해야 합니다.
- Process Specification 2.1.9에 쓰인 input값 enable/disable이 소스코드 상에서는 존재하지 않습니다. Enable/Disable이 어떤 상태를 의미하는 지 설명이 없어서 중지상태를 Enable, 동작상태를 Disable로 가정하고 테스트를 수행했습니다.