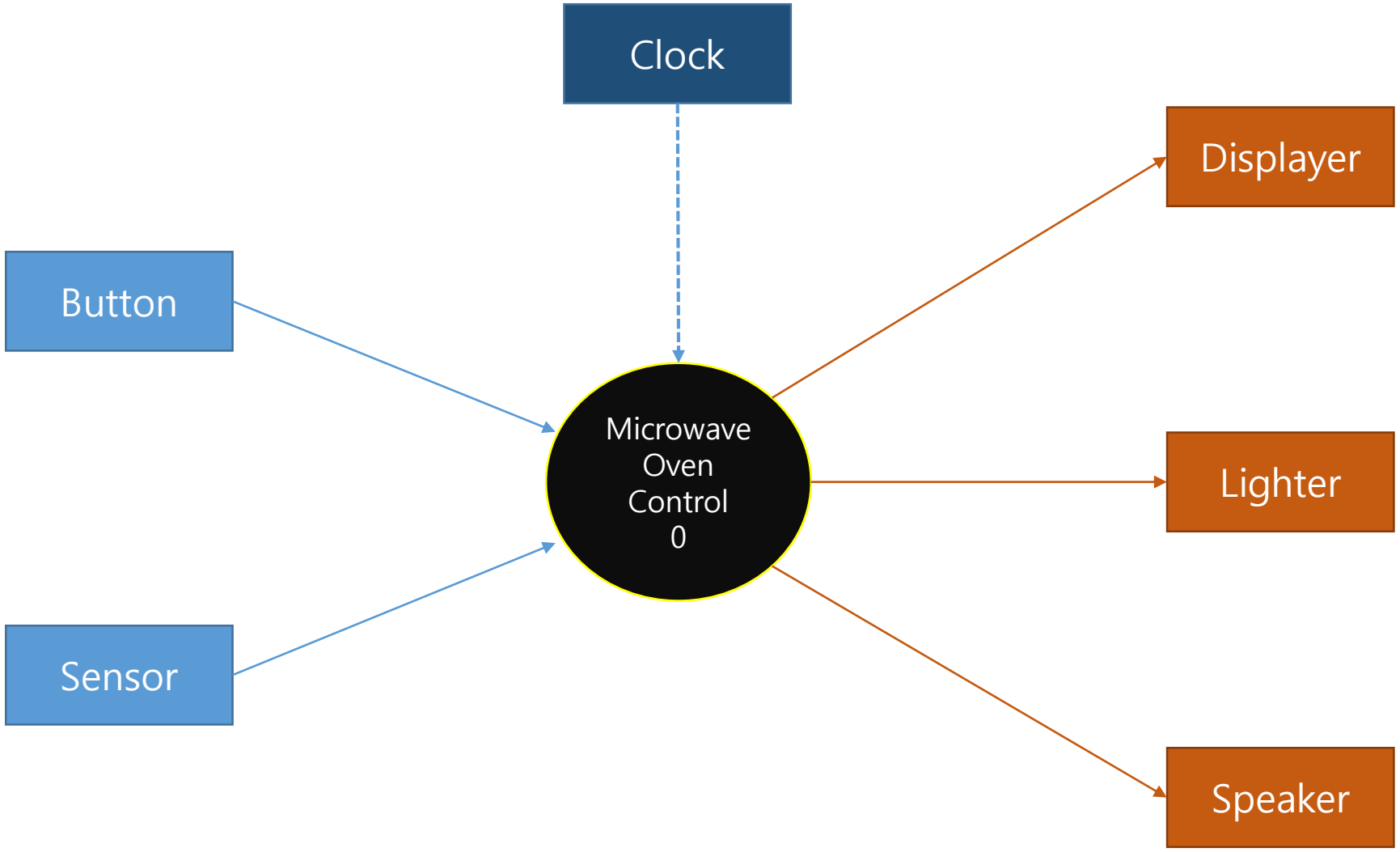
Microwave Oven System

T3

이경수,한득환,김대희,신민용

Structured Analysis

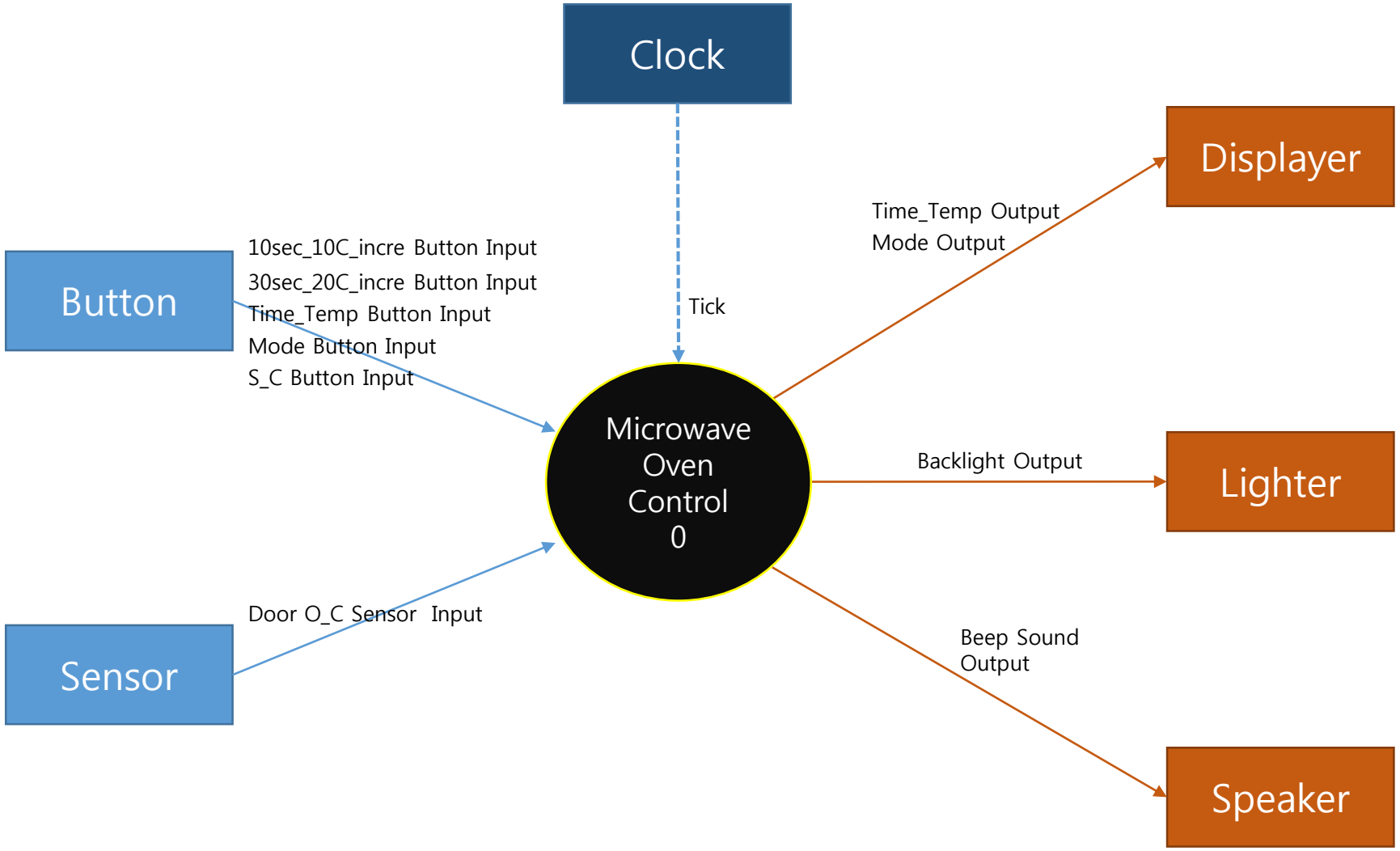
System Context Diagram



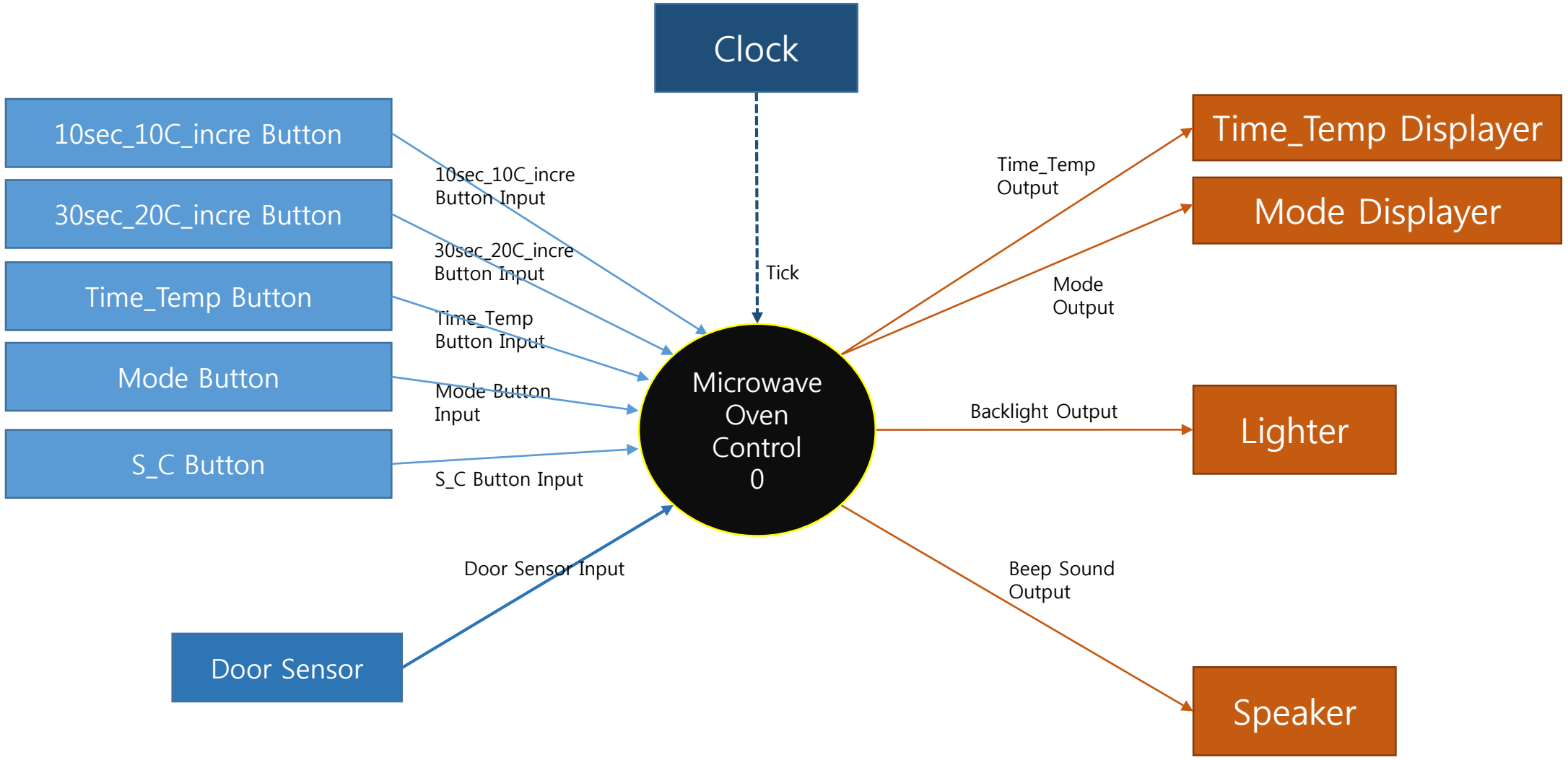
Event List

Event	Description
10sec_10C_incre Button Input	한번 누를 때 마다 시간/온도가 10sec/10C 씩 증가한다.
30sec_20C_incre Button Input	한번 누를 때 마다 시간/온도가 30sec/20C씩 증가한다.
Time_Temp Button Input	초기에는 Time으로 설정, 누를 때 마다 Time/Temp가 번갈아 변경된다.
Mode Button Input	미리 설정된 조리방법에 따라 정해진 시간으로 설정된다.
S_C Button Input	조리를 시작/취소 한다.
Door O_C Sensor Input	문을 열고 닫는다.
Temp Sensor Input	현재의 온도를 감지한다.
Tick	시간을 측정 해준다. Tick 하나당 1초로 가정한다.
Time_Temp Output	현재 시간 / 온도 를 보여준다.
Mode Output	현재 모드를 보여준다.
Backlight Output	Lighter을 통해 빛을 방출한다.
Beep Sound Output	Speaker를 통해 비프음을 낸다.

System Context Diagram



DFD Level 0


Clock
10sec_10C_incre Button
30sec_20C_incre Button
Time_Temp Button
Mode Button
S_C Button
10sec_10C_incre Button Input
30sec_20C_incre Button Input
Time_Temp Button Input
Mode Button Input
S_C Button Input
Tick
Microwave Oven Control 0
Door Sensor Input
Door Sensor
Time_Temp Output
Mode Output
Backlight Output
Beep Sound Output
Time_Temp Displayer
Mode Displayer
Lighter
Speaker

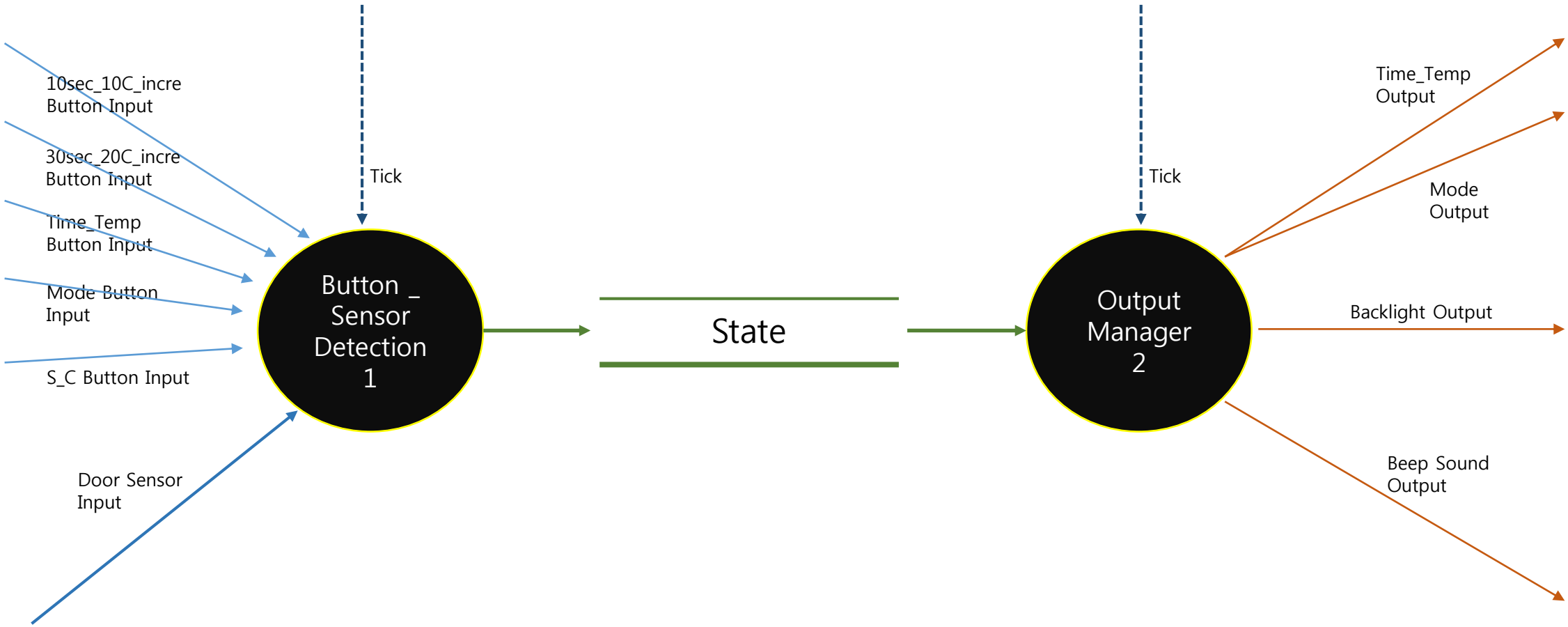
DFD Level 0 - Data Dictionary (1/2)

Event	Description	Format	Type
10sec_10C_incre Button Input	한번 누를 때 마다 시간/온도가 10sec/10C 씩 증가한다. Default는 False이고, 버튼의 입력이 들어 올 때 만 True가 된다.	False / True (Bool)	Interrupt
30sec_20C_incre Button Input	한번 누를 때 마다 시간/온도가 30sec/20C씩 증가한다. Default는 False이고, 버튼의 입력이 들어 올 때 만 True가 된다.	False / True (Bool)	Interrupt
Time_Temp Button Input	초기에는 Time으로 설정되어있고, 누를 때 마다 Temp -> Time 으로 번갈아 가며 변경된다. Default는 False이고, 버튼의 입력이 들어 올 때만 True가 된다.	False / True (Bool)	Interrupt
Mode Button Input	미리 설정된 조리방법에 따라 정해진 시간으로 설정된다. 처음에는 Manual로 되어있다. 버튼을 누를 때마다 "떡 ->죽 -> 밥 -> 국 -> 피자 -> Manual -> 떡 ..." 으로 진행된다.	Manual(0) / Dduck(1) / Juk(2) / Bob(3) / Kuk / Jjigae(4) / Pizza(5) (Integer)	Interrupt
S_C Button Input	조리를 시작/취소 한다. 조리 전이면 "시작" 하고, 조리 중이면 "취소 " 한다. Default는 False이고, 버튼의 입력이 들어 올 때 만 True가 되어 상황에 맞게 동작한다.	False / True (Bool)	Interrupt
Door O_C Sensor Input	문을 열고 닫는다. 조리 중에 문을 열면 조리가 멈추고 Backlight가 켜진다. Default는 False이고, 버튼의 입력이 들어 올 때 만 True가 된다.	False / True (Bool)	Interrupt

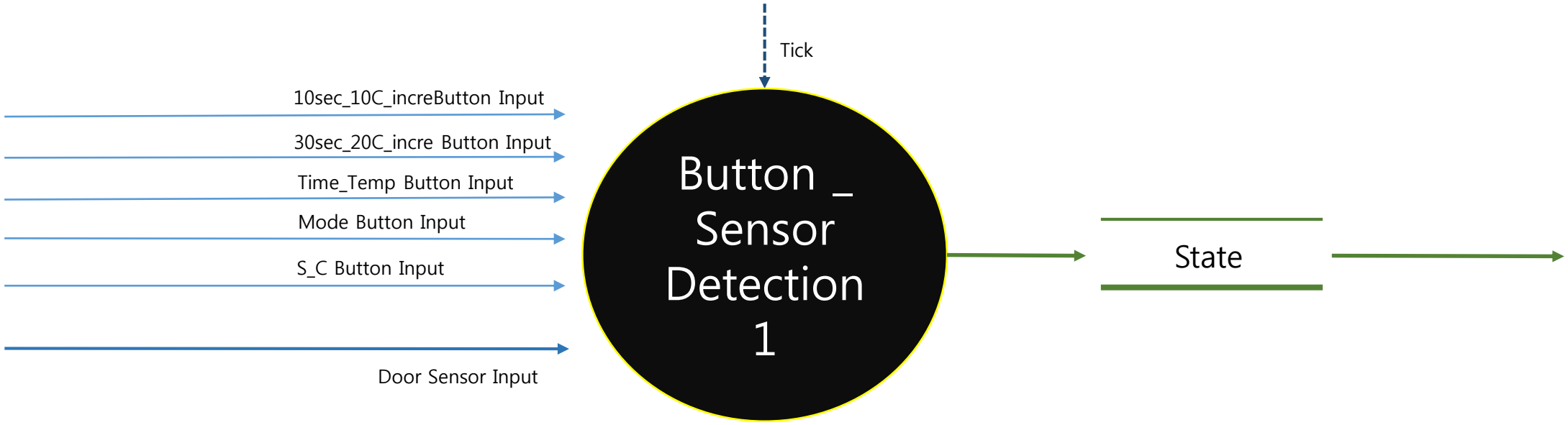
DFD Level 0 - Data Dictionary (2/2)

Event	Description	Format	Type
Tick	시간을 측정 해준다. Tick 하나당 1초로 가정한다. Default는 False이고 1초마다 Tick 이 True가 되어 들어간다.	False / True (Bool)	Temporal
Time_Temp Output	현재 시간 / 온도 를 보여준다.	00 : 00 / 00 C (Integer / Char)	Periodic
Mode Output	현재 모드를 보여준다. "00 : manual", "01 : 떡", "02 : 죽", "03 : 밥", "04 : 국_찌개", "05 : 피자" 이렇게 보여준다.	00 : ModeName (Integer / Char)	Periodic
Backlight Output	Lighter를 통해 온도와 시간, 모드를 노란색으로 표시해준다. Default는 False이고 문이 열렸을 때나 조리 중에만 켜진다.	False / True (Bool)	Periodic
Beep Sound Output	Speaker를 통해 비프음을 낸다. Default는 False이고 조리가 끝났을때 만 3초간 비프음을 낸다.	False / True (Bool)	Periodic

DFD Level 1

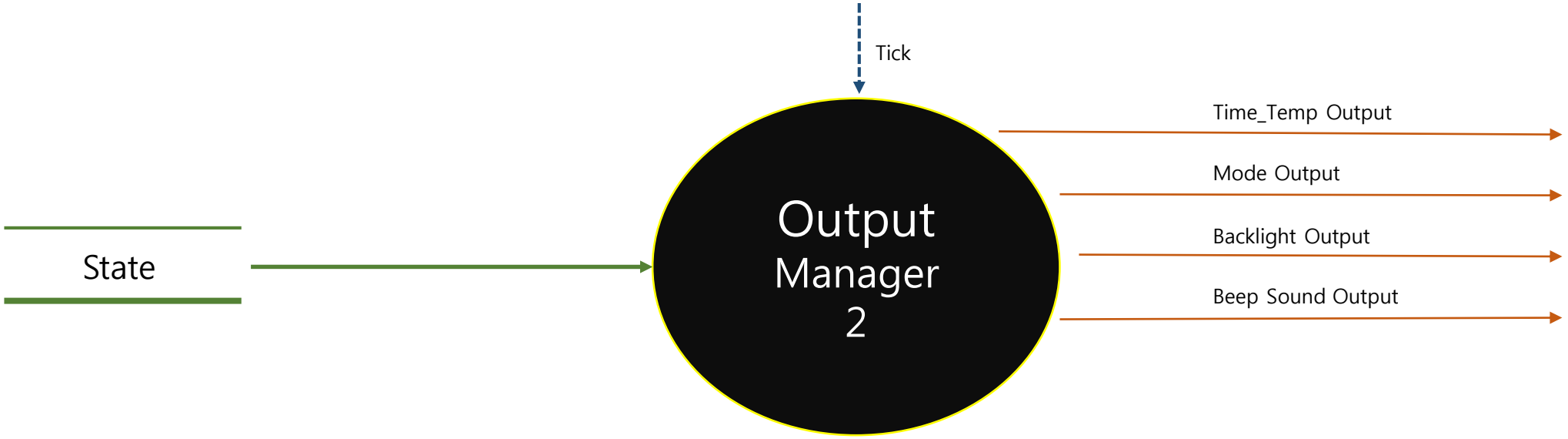

10sec_10C_incre
Button Input
30sec_20C_incre
Button Input
Time_Temp
Button Input
Mode Button
Input
S_C Button Input
Door Sensor
Input
Tick
Button _
Sensor
Detection
1
State
Tick
Output
Manager
2
Time_Temp
Output
Mode
Output
Backlight Output
Beep Sound
Output

DFD Level 1 – Process Specification



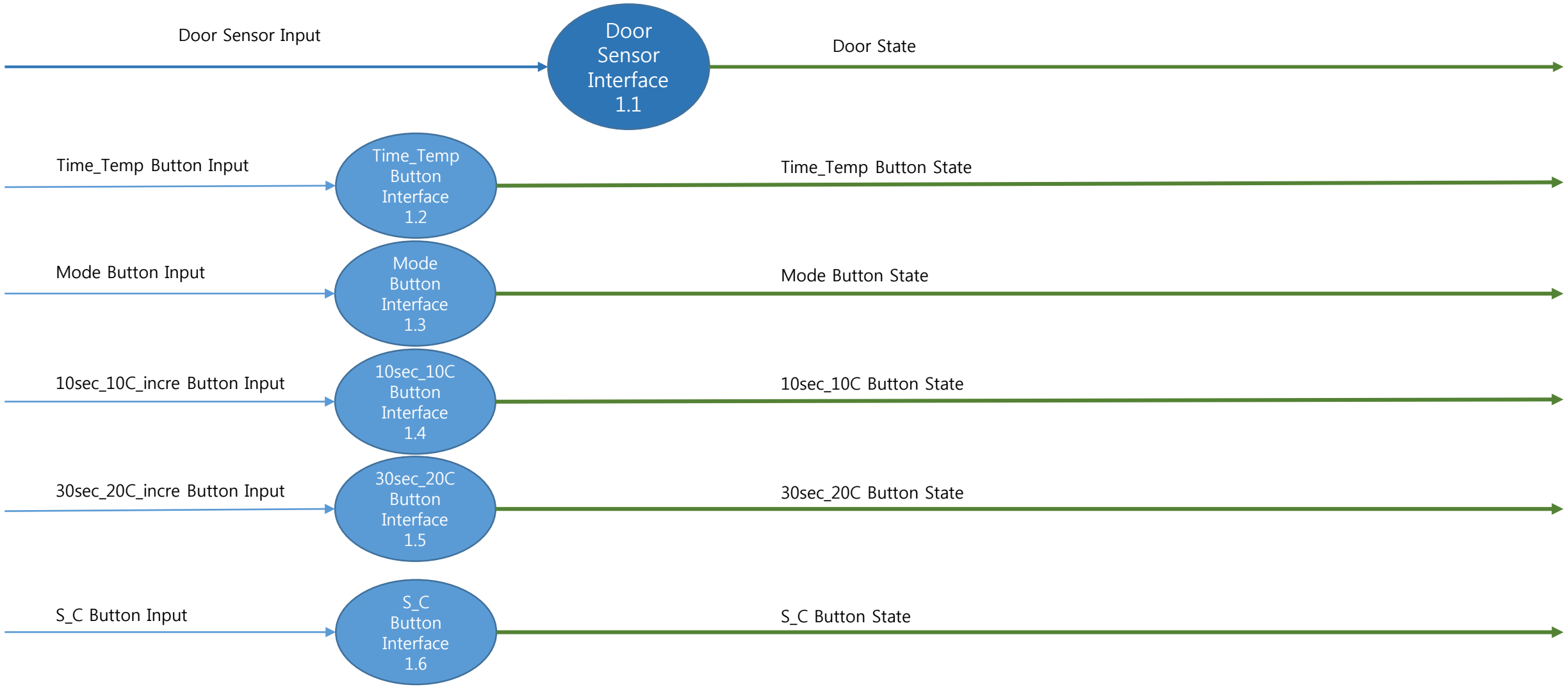
Reference No.	1
Name	Button _ Sensor Detection
Input	10sec_10C_incre Button Input, 30sec_20C_incre Button Input, Time_Temp Button Input, Mode Button Input, S_C Button Input, Door Sensor Input, Tick
Output	Input States / Door State
Process Description	여러 인풋들의 스테이트들을 모아 다음 프로세스에 전달한다.

DFD Level 1 – Process Specification



Reference No.	2
Name	Output Manager
Input	Input States / Door State, Tick
Output	Time_Temp Output, Mode Output, Backlight Output, Beep Sound Output
Process Description	여러 인풋들의 스테이트들 가지고 출력을 해준다.

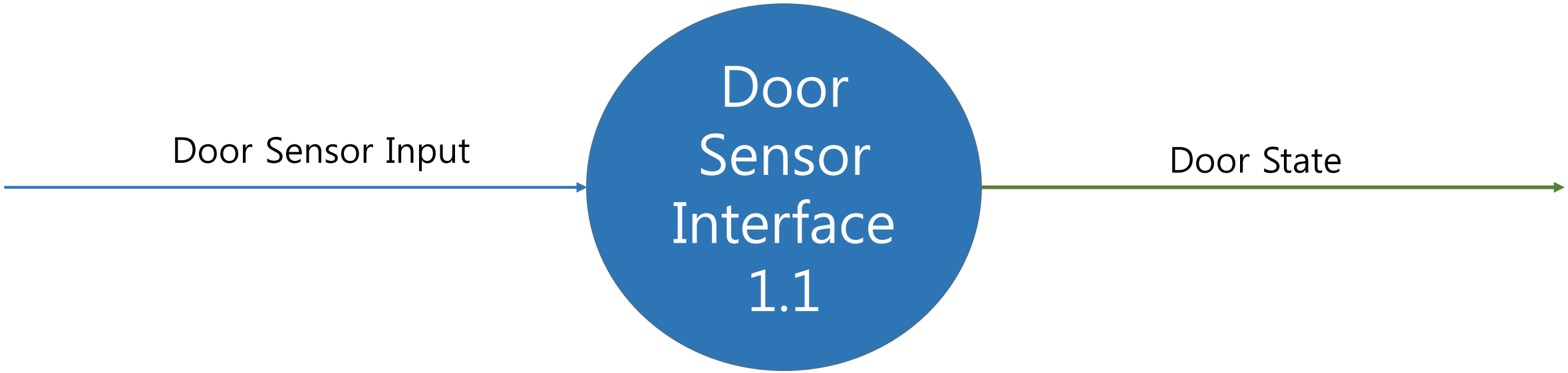
DFD Level 2 (1/2)



DFD Level 2 - Data Dictionary (1/2)

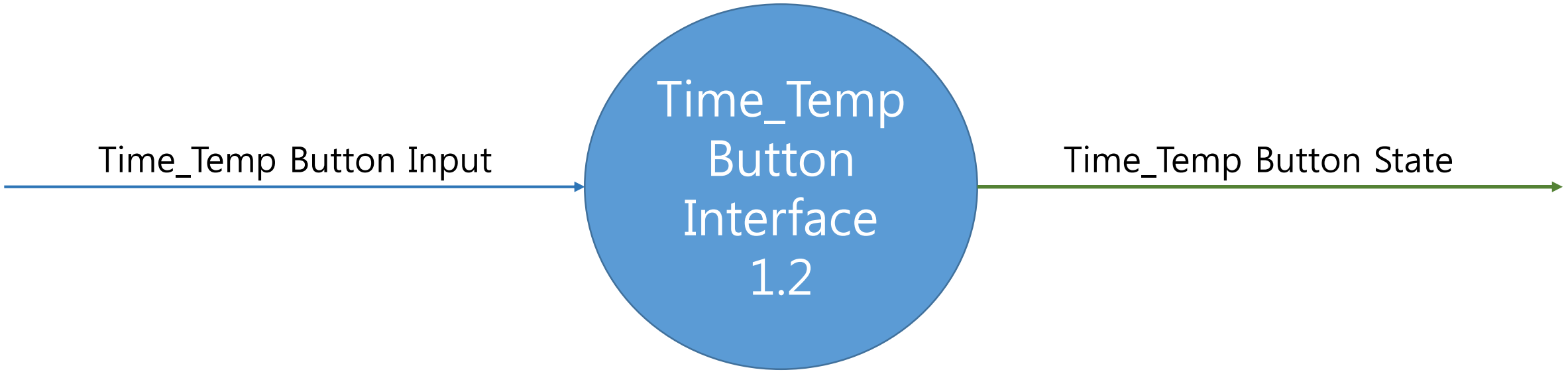
Event	Description	Format	Type
10sec_10C Button State	10sec_10C_incre Button의 입력이 들어왔는지 안 왔는지에 대한 상태를 나타낸다. (후에 1010B는 10sec_10C Button State 가 True면 1, False면 0 이다.)	False / True (Bool)	Interrupt
30sec_20C Button State	30sec_20C_incre Button의 입력이 들어왔는지 안 왔는지에 대한 상태를 나타낸다. (후에 3020B는 30sec_20C Button State 가 True면 1, False면 0 이다.)	False / True (Bool)	Interrupt
Time_Temp Button State	Time_Temp Button의 입력이 들어왔는지 안 왔는지에 대한 상태를 나타낸다. (후에 TT는 Time_Temp Button State가 True면 1, False면 0 이다.)	False / True (Bool)	Interrupt
Mode Button State	Mode Button의 입력이 들어왔는지 안 왔는지에 대한 상태를 나타낸다. (후에 M는 Mode Button State가 True면 1, False면 0 이다.)	False / True (Bool)	Interrupt
S_C Button State	S_C Button의 입력이 들어왔는지 안 왔는지에 대한 상태를 나타낸다. (후에 S는 S_C Button State가 True면 1, False면 0 이다.)	False / True (Bool)	Interrupt
Door State	문이 열려있는지 닫혀있는지에 대한 상태를 나타낸다. (후에 D는 Door State가 True면 1, False면 0 이다.)	False / True (Bool)	Interrupt

DFD Level 2 (1/2) – Process Specification



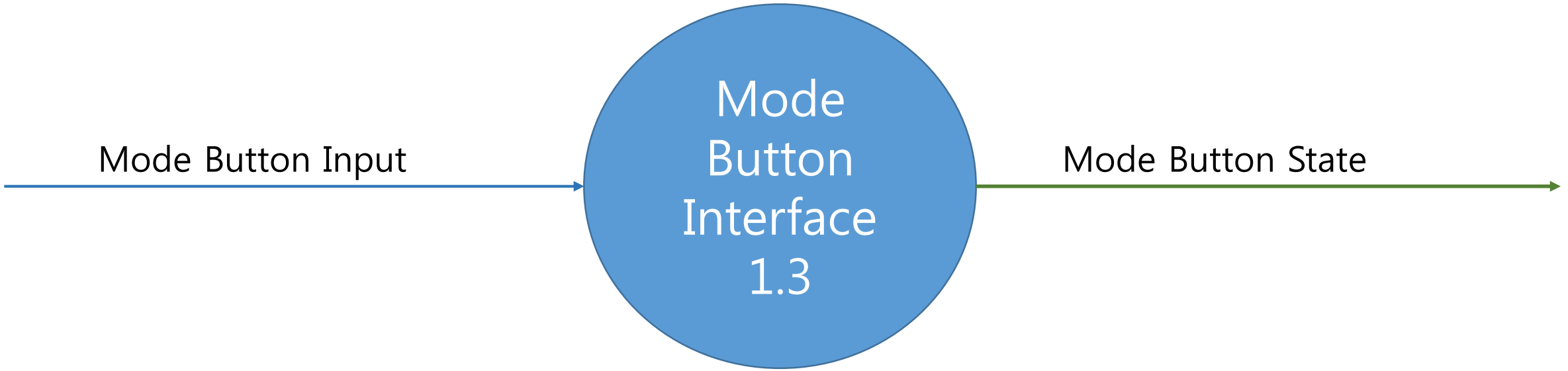
Reference No.	1.1
Name	Door Sensor Interface
Input	Door Sensor Input
Output	Door State (Bool)
Process Description	Door Sensor Input이 눌렸는지 안 눌렸는지 판단하여 Door State 를 True/False 로 내보낸다.

DFD Level 2 (1/2) – Process Specification



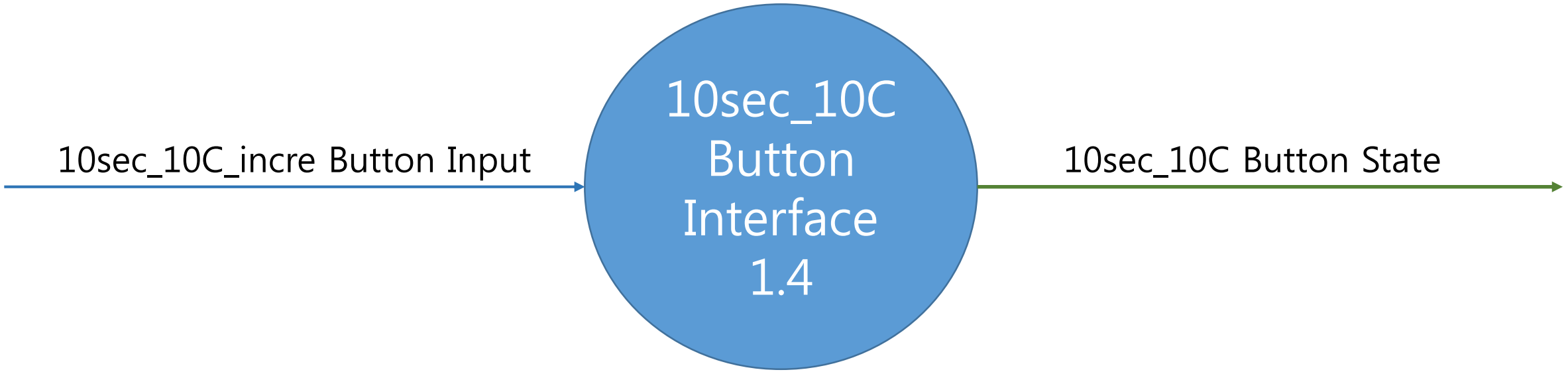
Reference No.	1.2
Name	Time_Temp Button Interface
Input	Time_Temp Button Input
Output	Time_Temp Button State (Bool)
Process Description	Time_Temp Button Input이 눌렸는지 안 눌렸는지 판단하여 Time_Temp State 를 True/False 로 내보낸다.

DFD Level 2 (1/2) – Process Specification



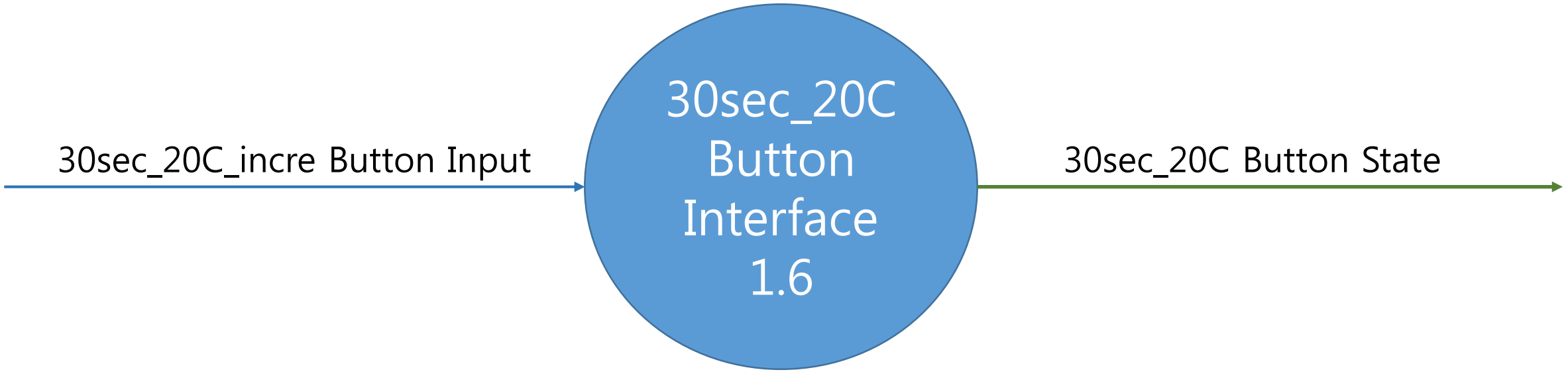
Reference No.	1.3
Name	Mode Button Interface
Input	Mode Button Input
Output	Mode Button State (Bool)
Process Description	Mode Button Input이 눌렸는지 안 눌렸는지 판단하여 Mode Button State 를 True/False 로 내보낸다.

DFD Level 2 (1/2) – Process Specification



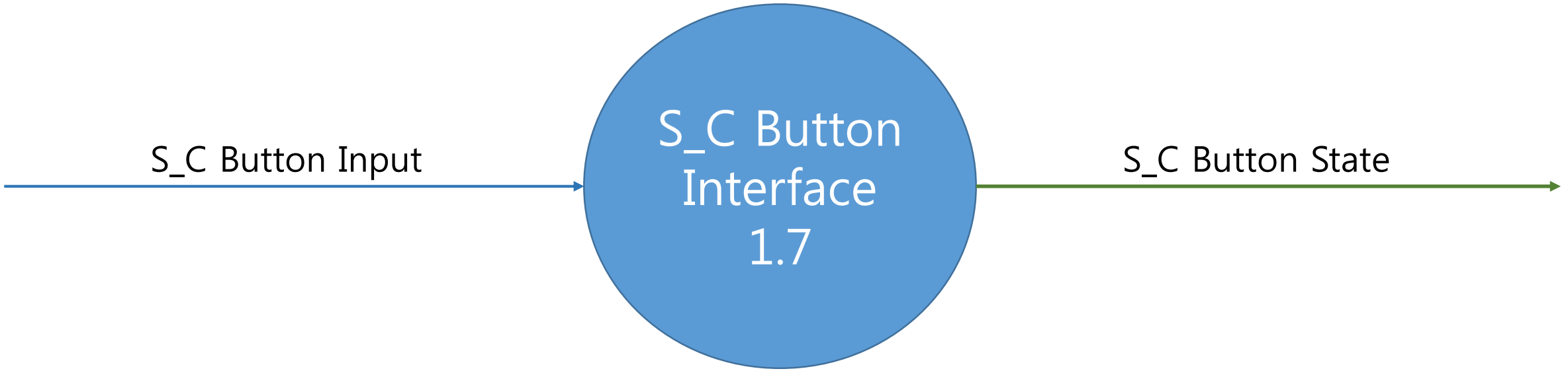
Reference No.	1.4
Name	10sec_10C Button Interface
Input	10sec_10C_incre Button Input
Output	10sec_10C Button State (Bool)
Process Description	10sec_10C_incre Button Input이 눌렸는지 안 눌렸는지 판단하여 10sec_10C Button State 를 True/False 로 내보낸다.

DFD Level 2 (1/2) – Process Specification



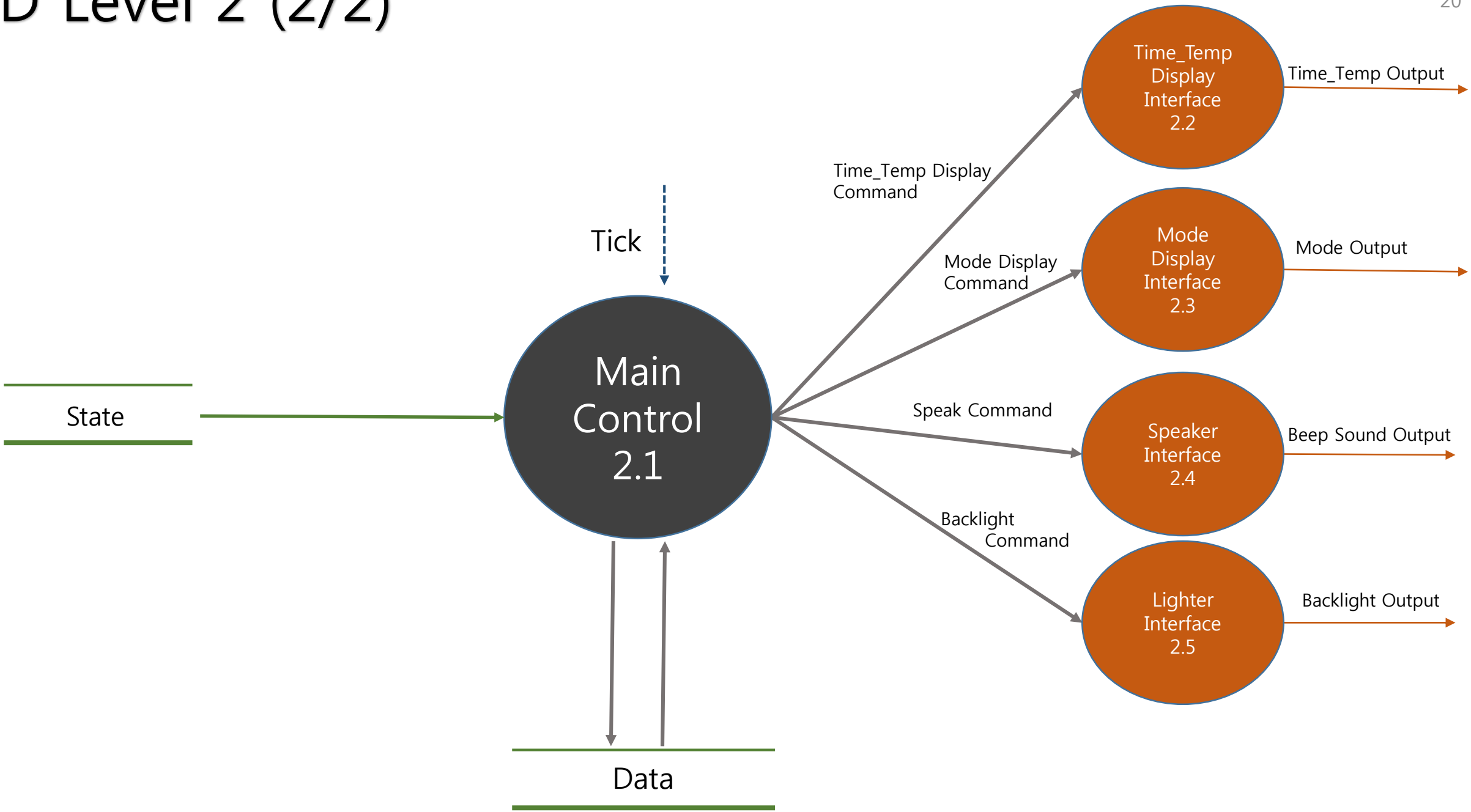
Reference No.	1.6
Name	30sec_20C Button Interface
Input	30sec_20C_incre Button Input
Output	30sec_20C Button State (Bool)
Process Description	30sec_20C_incre Button Input이 눌렸는지 안 눌렸는지 판단하여 30sec_20C Button State 를 True/False 로 내보낸다.

DFD Level 2 (1/2) – Process Specification



Reference No.	1.7
Name	S_C Button Interface
Input	S_C Button Input
Output	S_C Button State (Bool)
Process Description	S_C Button Input이 눌렸는지 안 눌렸는지 판단하여 S_C Button State 를 True/False 로 내보낸다.

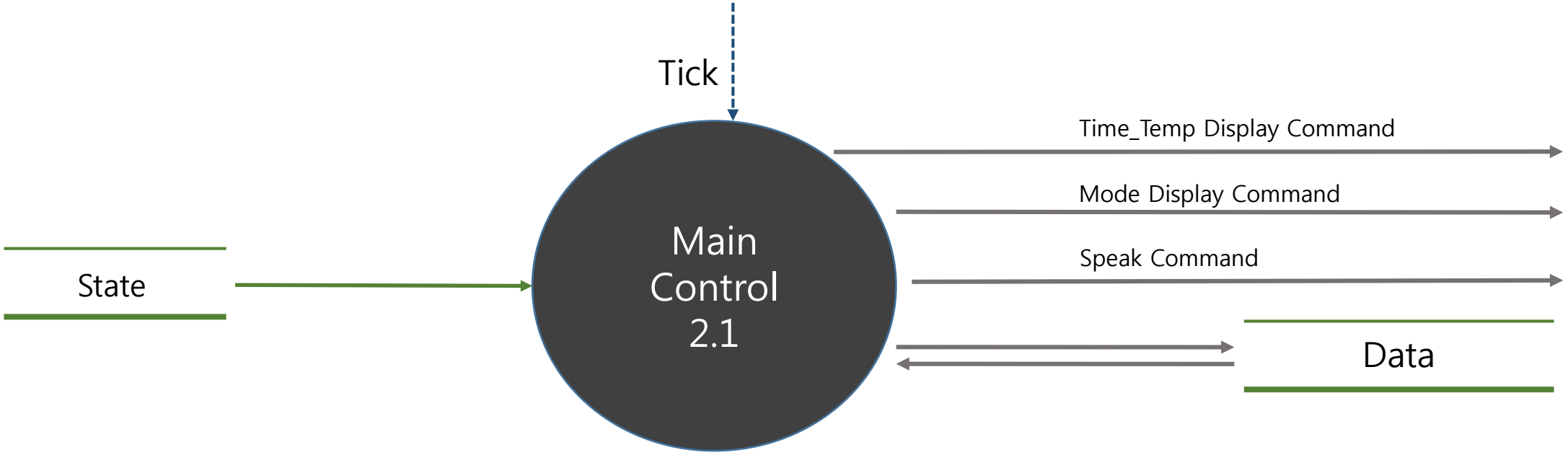
DFD Level 2 (2/2)



DFD Level 2 - Data Dictionary (2/2)

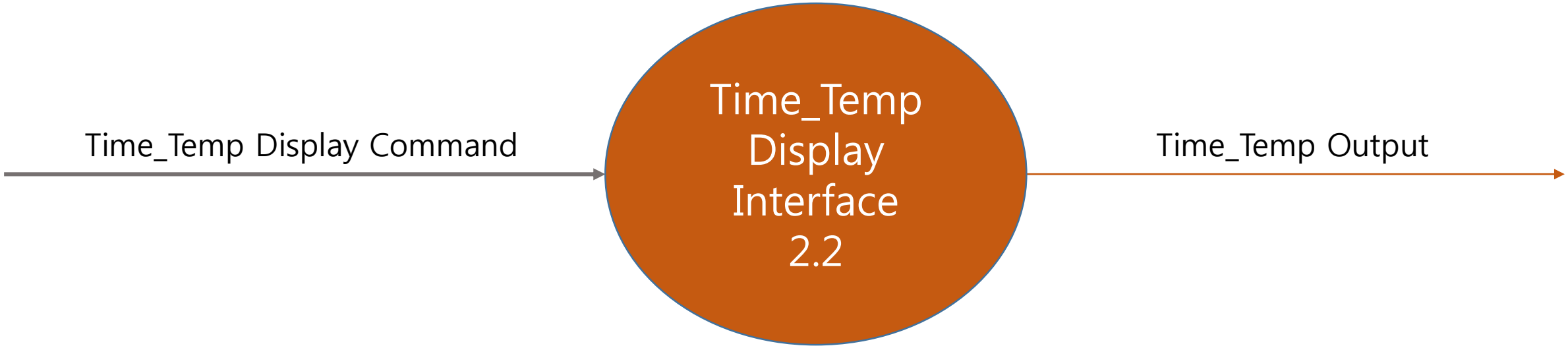
Event	Description	Format	Type
Time_Temp Display Command	Time_Temp Display Interface 를 통해 현재 시간과 온도를 출력하도록 명령 한다.	False / True (Bool)	Interrupt
Mode Display Command	Mode Display Interface 를 통해 현재 모드를 출력하도록 명령 한다.	False / True (Bool)	Interrupt
Speak Command	Speaker Interface 를 소리를 방출 하도록 명령 한다.	False / True (Bool)	Interrupt
Backlight Command	Lighter Interface를 통해 빛을 방출 하도록 명령 한다.	00 (Integer)	Periodic
Data	입력에 따른 시간을 stime 에 저장하고, 입력에 따른 온도를 stemp 에 저장한다. 그리고 현재 시간을 ctime, 현재 온도를 ctemp에 저장한다. 그리고 cmode에는 현재 모드가 저장된다.	00 "AB" (Integer, Char)	Periodic

DFD Level 2 (2/2) – Process Specification



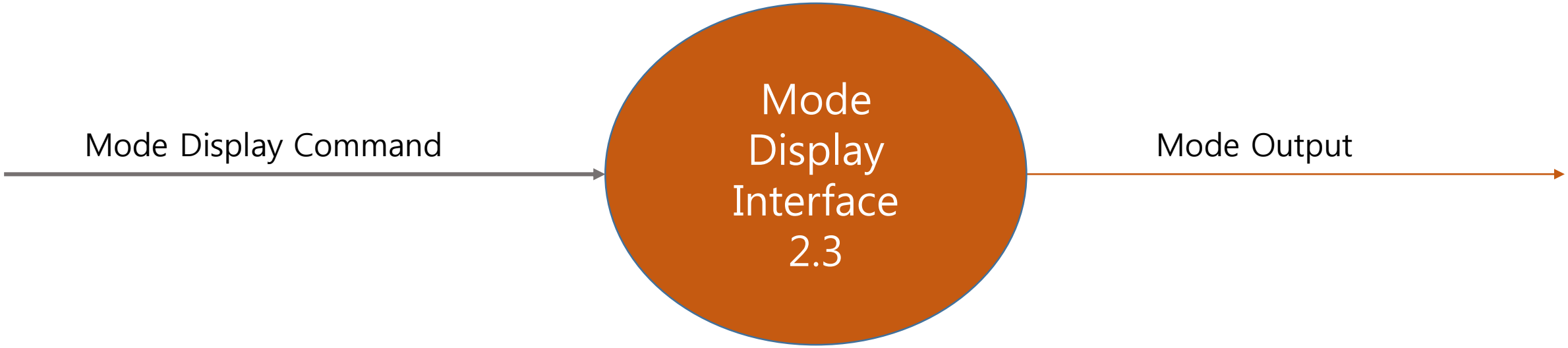
Reference No.	2.1
Name	Main Control
Input	State
Output	Time_Temp Display Command, Mode Display Command, Speak Command, Data
Process Description	버튼들의 인풋 스테이트를 받아와서 Data에 저장하고 끌어와 출력을 제어 한다.

DFD Level 2 (2/2) – Process Specification



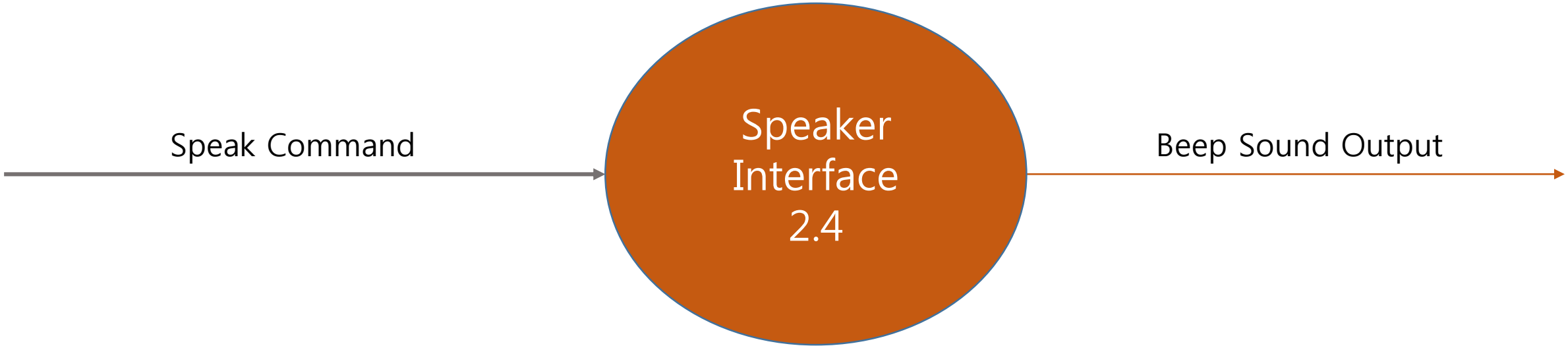
Reference No.	2.2
Name	Time_Temp Display Interface
Input	Time_Temp Display Command (Bool)
Output	Time_Temp Output
Process Description	컨트롤에서 받아온 Time_Temp Display Command 가 True면 시간 또는 온도를 출력해 준다.

DFD Level 2 (2/2) – Process Specification



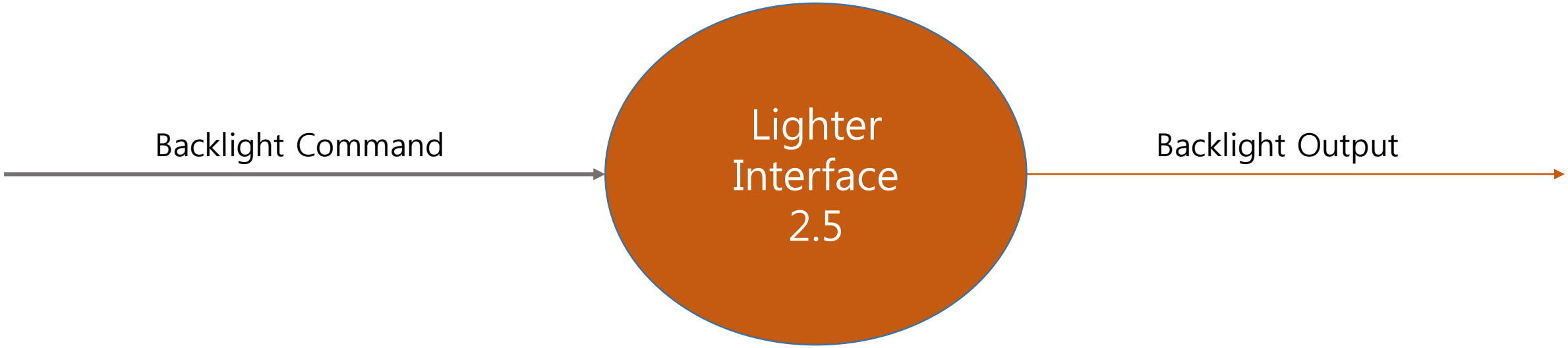
Reference No.	2.3
Name	Mode Display Interface
Input	Mode Display Command (Bool)
Output	Mode Output
Process Description	컨트롤에서 받아온 Mode Display Command 가 True면 모드를 출력해 준다.

DFD Level 2 (2/2) – Process Specification



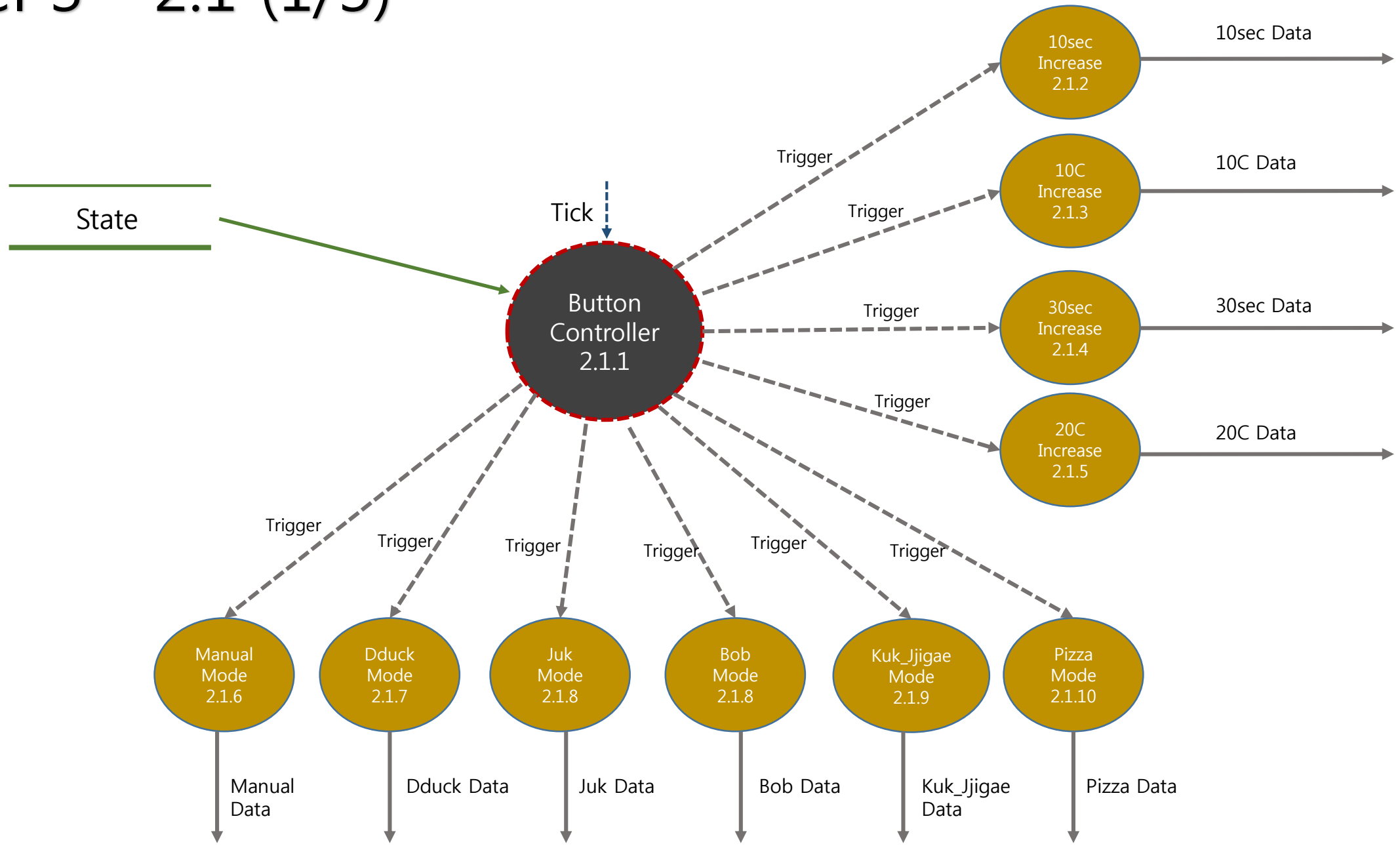
Reference No.	2.4
Name	Speaker Interface
Input	Speak Command (Bool)
Output	Beep Sound Output
Process Description	컨트롤에서 받아온 Speak Command 가 True면 비프음을 3초간 출력해 준다.

DFD Level 2 (2/2) – Process Specification



Reference No.	2.5
Name	Lighter Interface
Input	Backlight Command (Bool)
Output	Backlight Output
Process Description	컨트롤에서 받아온 Backlight Command 가 True면 빛을 방출해 준다.

DFD Level 3 - 2.1 (1/3)



DFD Level 3- 2.1 - Data Dictionary

Event	Description	Format	Type
10sec Data	10초 라는 시간 데이터를 Data에 있는 stime에 누적 시킨다.	00 (Integer)	Interrupt
10C Data	10도 라는 온도 데이터를 Data에 있는 stemp에 누적 시킨다.	00 (Integer)	Interrupt
30sec Data	30초 라는 시간 데이터를 Data에 있는 stime에 누적 시킨다.	00 (Integer)	Interrupt
20C Data	20도 라는 온도 데이터를 Data에 있는 stemp에 누적 시킨다.	00 (Integer)	Interrupt
Manual Data	Manual Mode 라는 문자 데이터를 Data에 있는 cmode에 변경 시킨다.	"AB" (Char)	Interrupt
Dduck Mode	Dduck Mode 라는 시간 데이터(1분)와 문자 데이터("Dduck")를 Data에 있는 stime과 cmode에 변경 시킨다.	00 / "AB" (Integer / Char)	Interrupt
Juk Mode	Juk Mode 라는 시간 데이터(1분 30초)와 문자 데이터("Juk")를 Data에 있는 stime과 cmode에 변경 시킨다.	00 / "AB" (Integer / Char)	Interrupt
Bob Mode	Bob Mode 라는 시간 데이터(1분)와 문자 데이터("Bob")를 Data에 있는 stime과 cmode에 변경 시킨다.	00 / "AB" (Integer / Char)	Interrupt
Kuk_Jjigae Mode	Kuk_Jjigar Mode 라는 시간 데이터(5분)와 문자 데이터("Kuk_Jjigae")를 Data에 있는 stime과 cmode에 변경 시킨다.	00 / "AB" (Integer / Char)	Interrupt
Pizza Mode	Pizza Mode 라는 시간 데이터(2분)와 문자 데이터("Pizza")를 Data에 있는 stime과 cmode에 변경 시킨다.	00 / "AB" (Integer / Char)	Interrupt

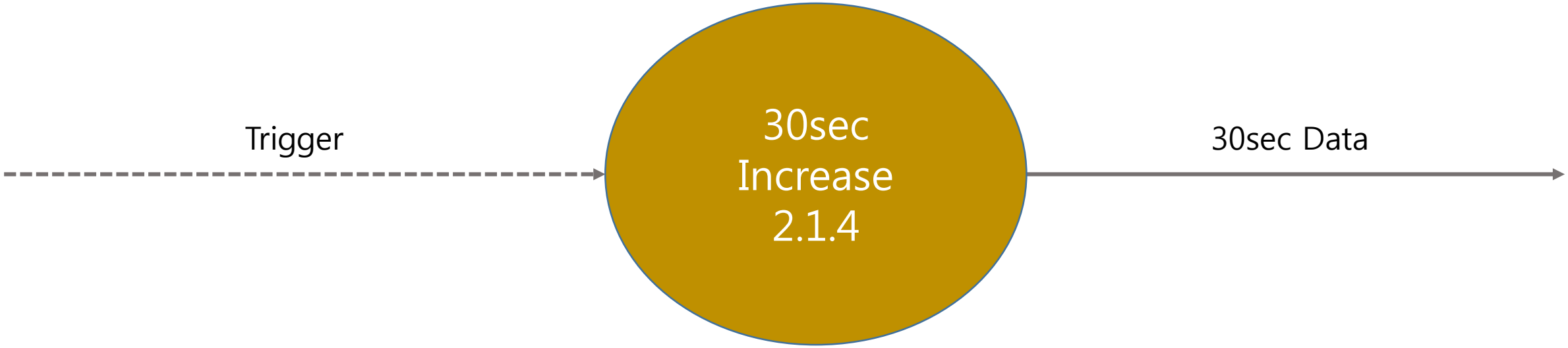
DFD Level 3 – 2.1 Process Specification

Reference No.	2.1.2
Name	10sec Increase
Input	Trigger
Output	10sec Data (Integer)
Process Description	10sec Increase 에 Trigger가 들어 오면 10초 라는 시간 데이터를 Data에 있는 stime에 누적 시킨다.

DFD Level 3 – 2.1 Process Specification

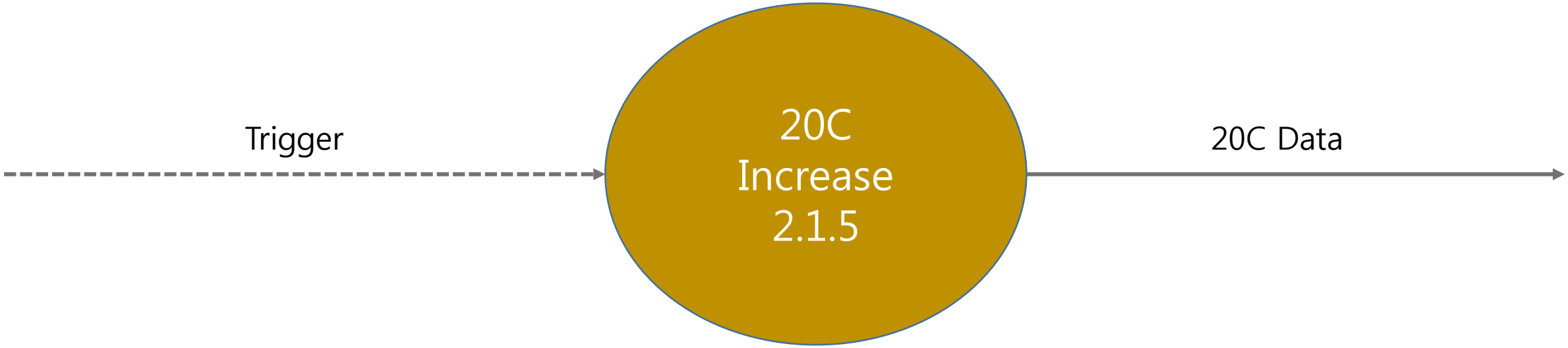
Reference No.	2.1.3
Name	10C Increase
Input	Trigger
Output	10C Data (Integer)
Process Description	10C Increase 에 Trigger가 들어 오면 10도 라는 온도 데이터를 Data에 있는 stemp에 누적 시킨다.

DFD Level 3 – 2.1 Process Specification



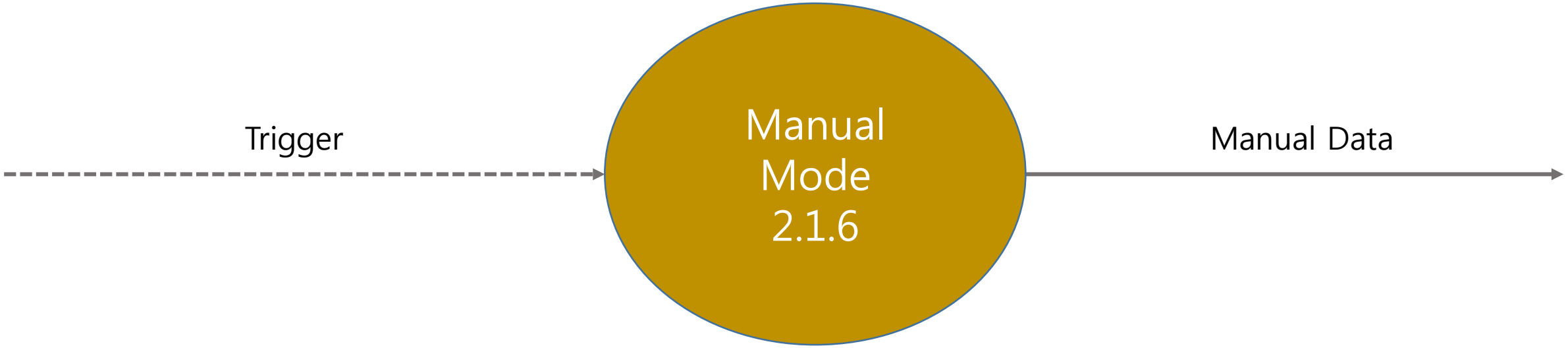
Reference No.	2.1.4
Name	30sec Increase
Input	Trigger
Output	30sec Data (Integer)
Process Description	30sec Increase 에 Trigger가 들어 오면 30초 라는 시간 데이터를 Data에 있는 stime에 누적 시킨다.

DFD Level 3 – 2.1 Process Specification



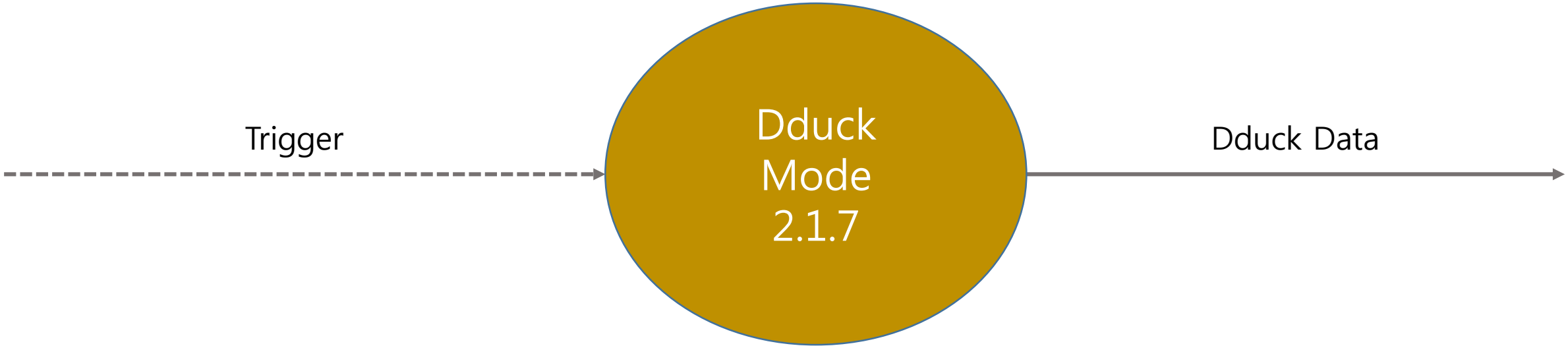
Reference No.	2.1.5
Name	20C Increase
Input	Trigger
Output	20C Data (Integer)
Process Description	20C Increase 에 Trigger가 들어 오면 20도 라는 온도 데이터를 Data에 있는 stemp에 누적 시킨다.

DFD Level 3 – 2.1 Process Specification



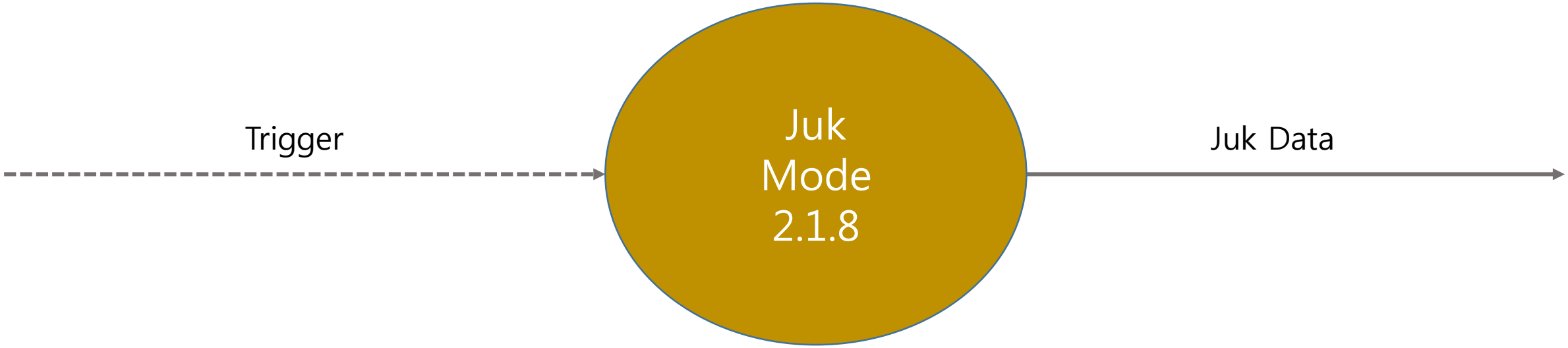
Reference No.	2.1.6
Name	Manual Mode
Input	Trigger
Output	Manual Data (Char)
Process Description	Manual Mode에 Trigger가 들어 오면 Manual Mode 라는 문자 데이터를 Data에 있는 cmode에 변경 시킨다.

DFD Level 3 – 2.1 Process Specification



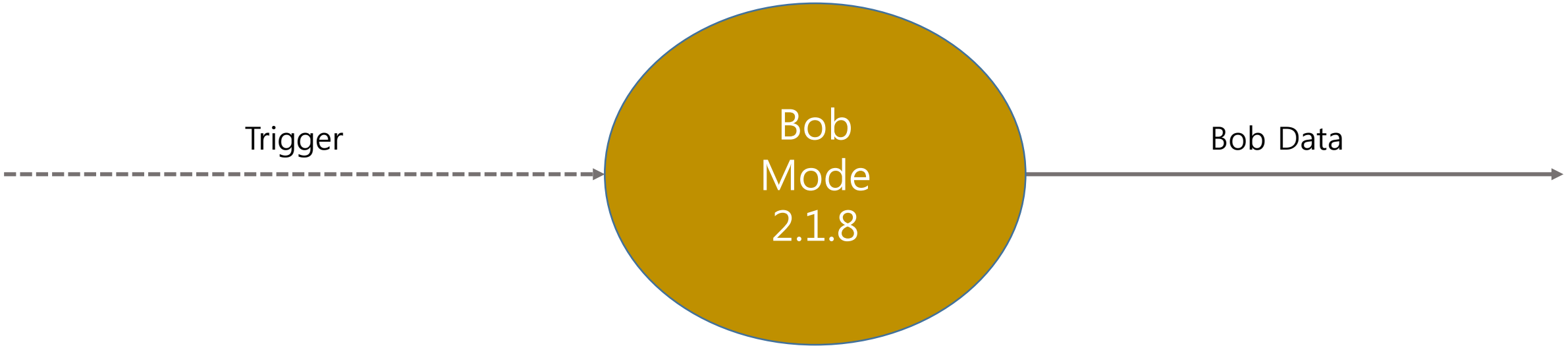
Reference No.	2.1.7
Name	Dduck Mode
Input	Trigger
Output	Dduck Data
Process Description	Dduck Mode에 Trigger가 들어 오면 Dduck Mode 라는 시간 데이터와 문자 데이터를 Data에 있는 stime과 cmode에 변경 시킨다.

DFD Level 3 – 2.1 Process Specification



Reference No.	2.1.8
Name	Juk Mode
Input	Trigger
Output	Juk Data
Process Description	Juk Mode에 Trigger가 들어 오면 Juk Mode 라는 시간 데이터와 문자 데이터를 Data에 있는 stime과 cmode에 변경 시킨다.

DFD Level 3 – 2.1 Process Specification



Reference No.	2.1.8
Name	Bob Mode
Input	Trigger
Output	Bob Data
Process Description	Bob Mode에 Trigger가 들어 오면 Bob Mode 라는 시간 데이터와 문자 데이터를 Data에 있는 stime과 cmode에 변경 시킨다.

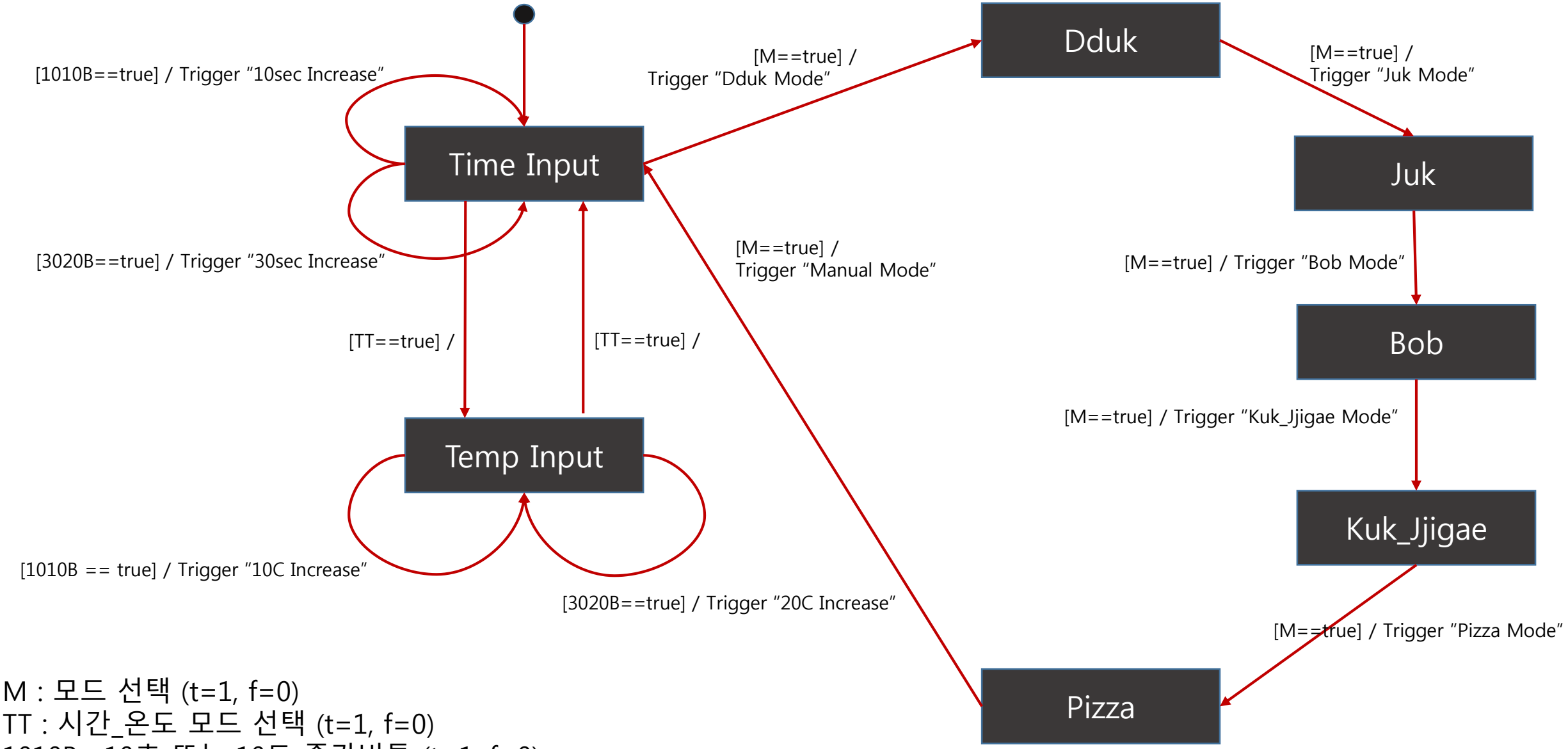
DFD Level 3 – 2.1 Process Specification

Reference No.	2.1.9
Name	Kuk_Jjigae Mode
Input	Trigger
Output	Kuk_Jjigae Data
Process Description	Kuk_Jjigae Mode에 Trigger가 들어 오면 Kuk_Jjigae Mode 라는 시간 데이터와 문자 데이터를 Data에 있는 stime과 cmode에 변경 시킨다.

DFD Level 3 – 2.1 Process Specification

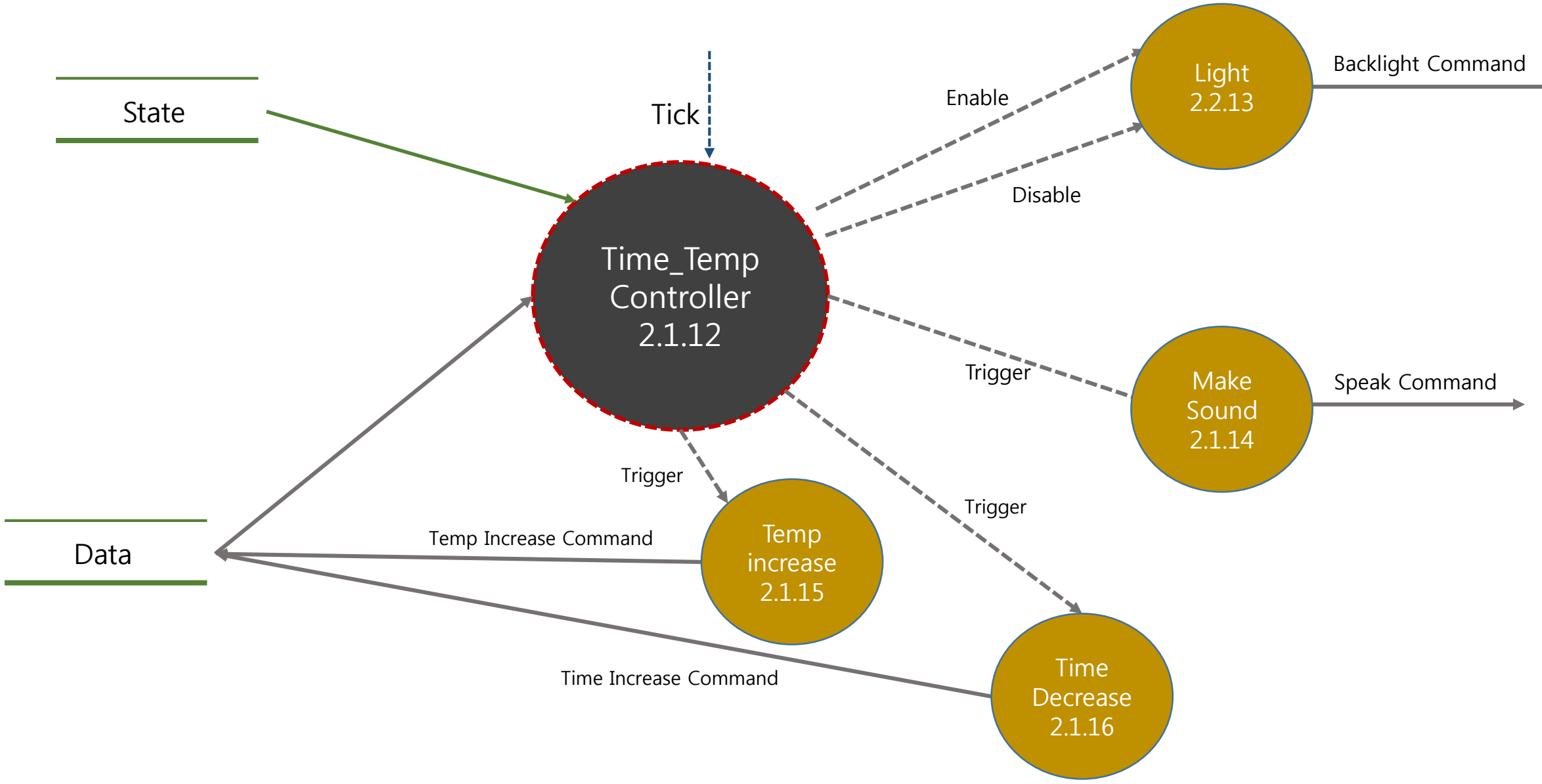
Reference No.	2.1.10
Name	Pizza Mode
Input	Trigger
Output	Pizza Data
Process Description	Pizza Mode에 Trigger가 들어 오면 Pizza Mode 라는 시간 데이터와 문자 데이터를 Data에 있는 stime과 cmode에 변경 시킨다.

DFD Level 3 - 2.1 (1/3) [State Transition Diagram for Button Controller 2.1.1]

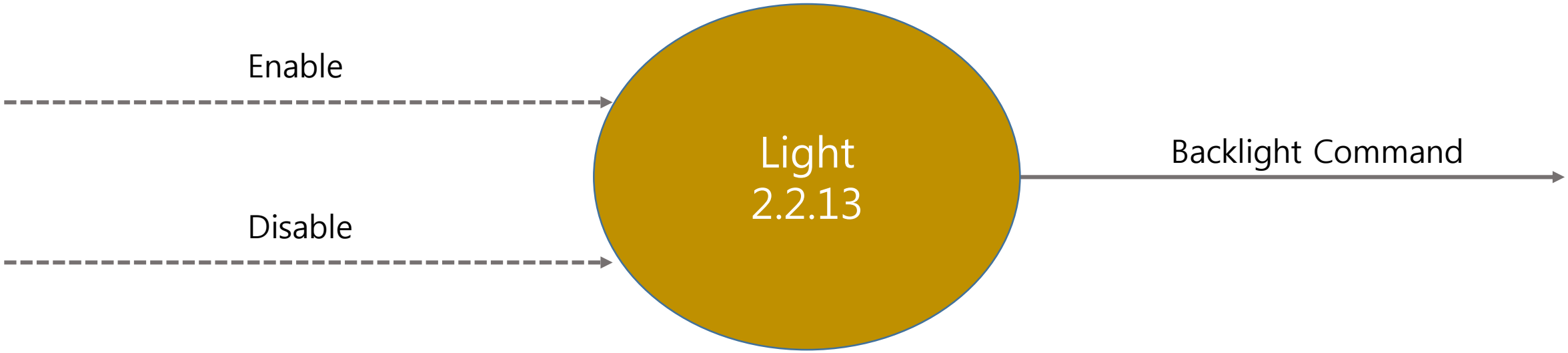


M : 모드 선택 (t=1, f=0)
TT : 시간_온도 모드 선택 (t=1, f=0)
1010B : 10초 또는 10도 증가버튼 (t=1, f=0)
3020B : 30초 또는 20도 증가버튼 (t=1, f=0)

DFD Level 3 - 2.1 (2/3)



DFD Level 3 – 2.1 Process Specification

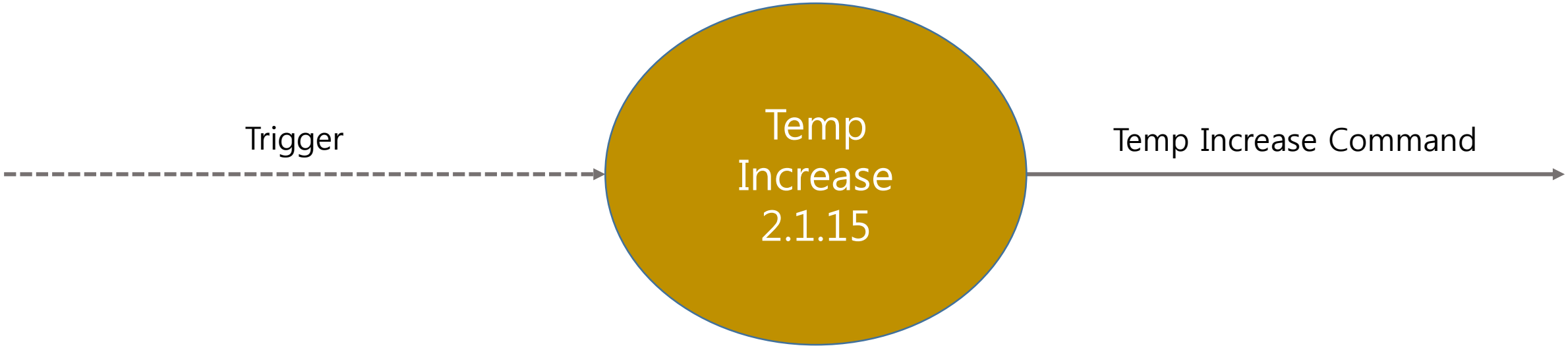


Reference No.	2.1.13
Name	Light
Input	Enable, Disable
Output	Backlight Command (Bool)
Process Description	Light에 Enable이 들어오면 Lighter Interface에게 Backlight를 켜라는 Backlight Command를 보낸다. Disable이 들어오면 끄라는 Backlight Command를 보낸다.

DFD Level 3 – 2.1 Process Specification

Reference No.	2.1.14
Name	Make Sound
Input	Trigger
Output	Speak Command (Bool)
Process Description	Make Sound에 Trigger가 들어오면 Speaker Interface에게 Speak하라는 Speak Command를 보낸다. Disable이 들어오면 끄라는 Speak Command를 보낸다.

DFD Level 3 – 2.1 Process Specification

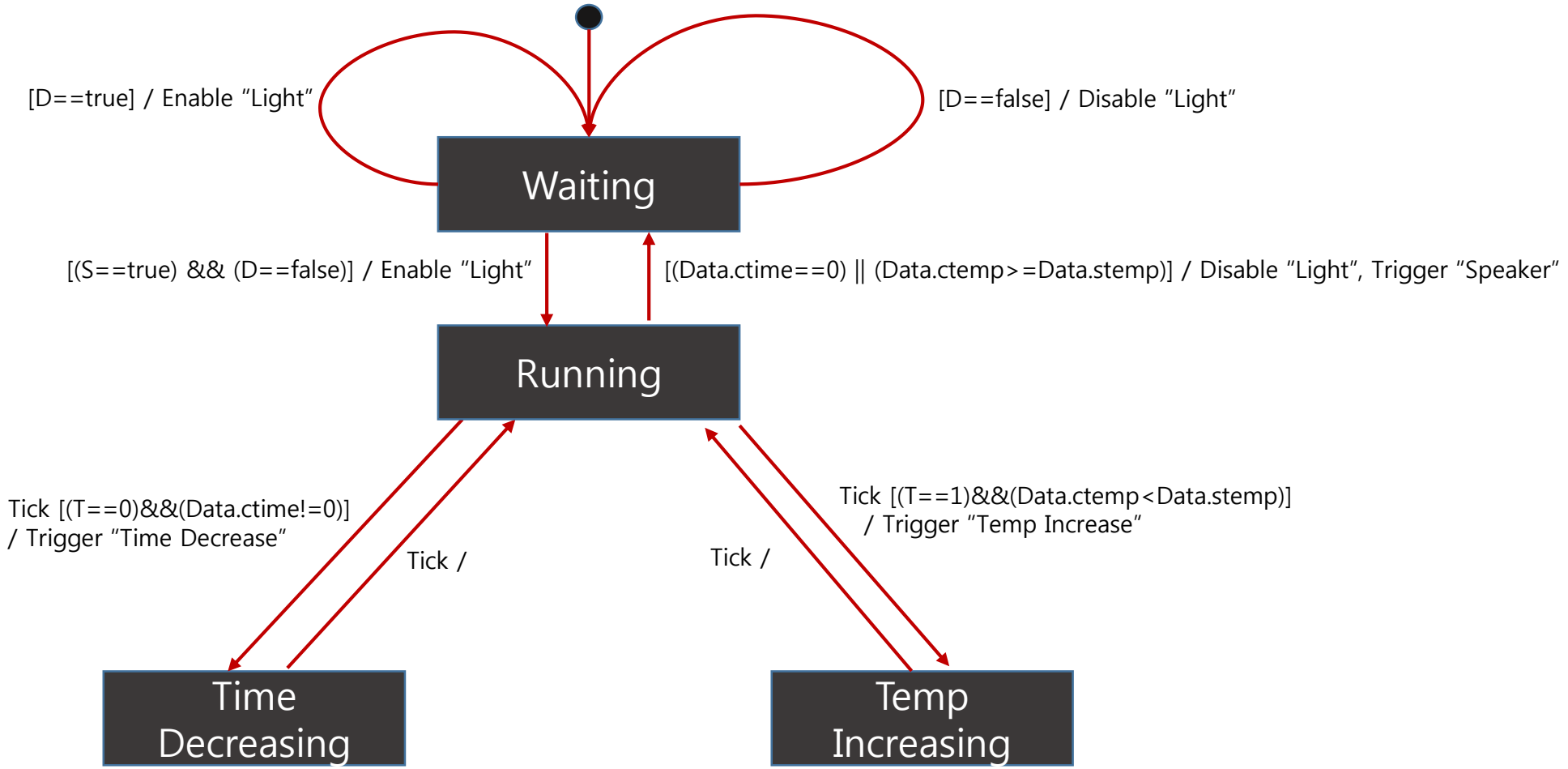


Reference No.	2.1.15
Name	Temp Increase
Input	Trigger
Output	Temp Increase Command
Process Description	Temp Increase에 Trigger가 들어오면 Data에 ctemp를 증가시킨다.

DFD Level 3 – 2.1 Process Specification

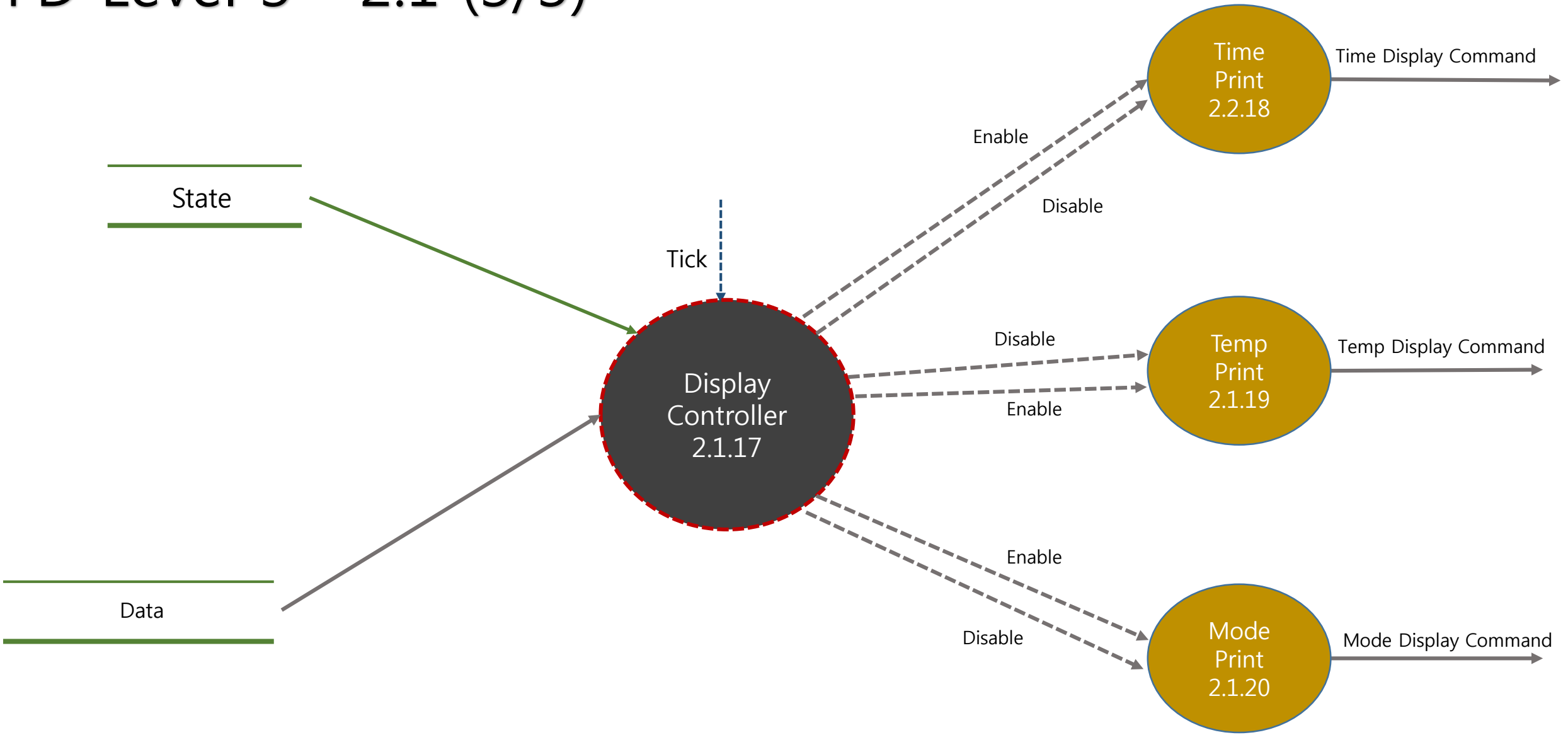
Reference No.	2.1.16
Name	Time Decrease
Input	Trigger
Output	Time Increase Command
Process Description	Time Increase에 Trigger가 들어오면 Data에 ctime을 증가시킨다.

DFD Level 3 - 2.1 (2/3) [State Transition Diagram for Time_Temp Controller 2.1.11]

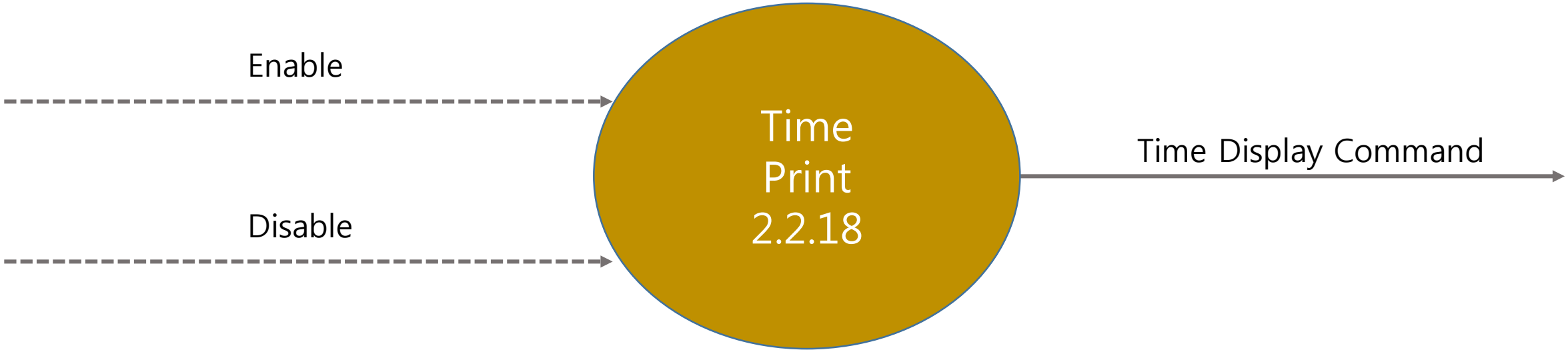


S : 조리 시작_취소 (t = 1, f = 0), D : 문상태 (t = 1, f = 0), T : Time / Temp (0 = 시간, 1 = 온도)
Data.ctime : Data에 저장되어있는 현재 시간, Data.ctemp : Data에 저장되어있는 현재 온도
Data.stime : Data에 저장되어있는 설정된 시간, Data.stemp : Data에 저장되어있는 설정된 온도

DFD Level 3 - 2.1 (3/3)

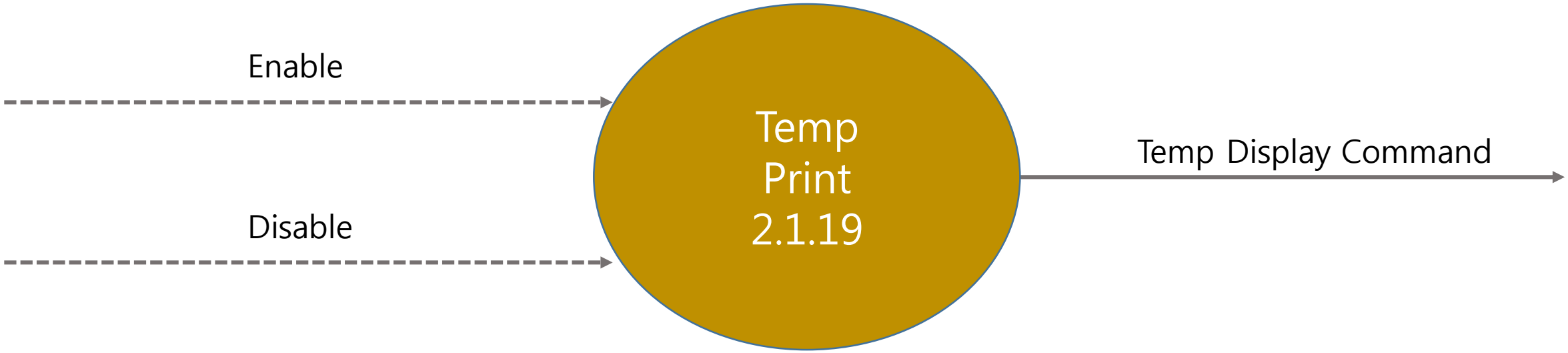


DFD Level 3 – 2.1 Process Specification



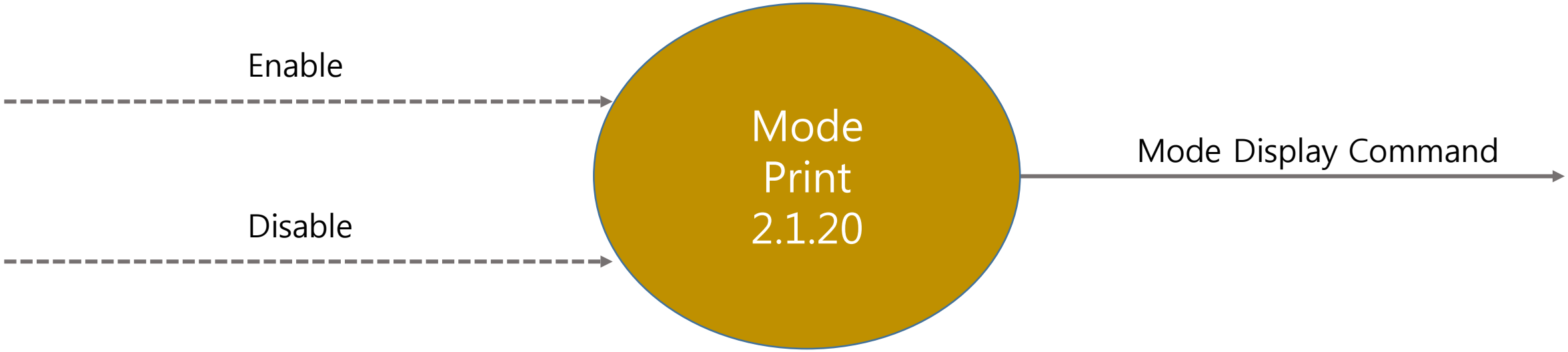
Reference No.	2.1.18
Name	Time Print
Input	Enable, Disable
Output	Time Display Command (Bool)
Process Description	Time Print에 Enable이 들어오면 Time_Temp Display Interface에게 Time을 Display하라는 Time Display Command를 보낸다. Disable이 들어오면 끄라는 Time Display Command를 보낸다.

DFD Level 3 – 2.1 Process Specification



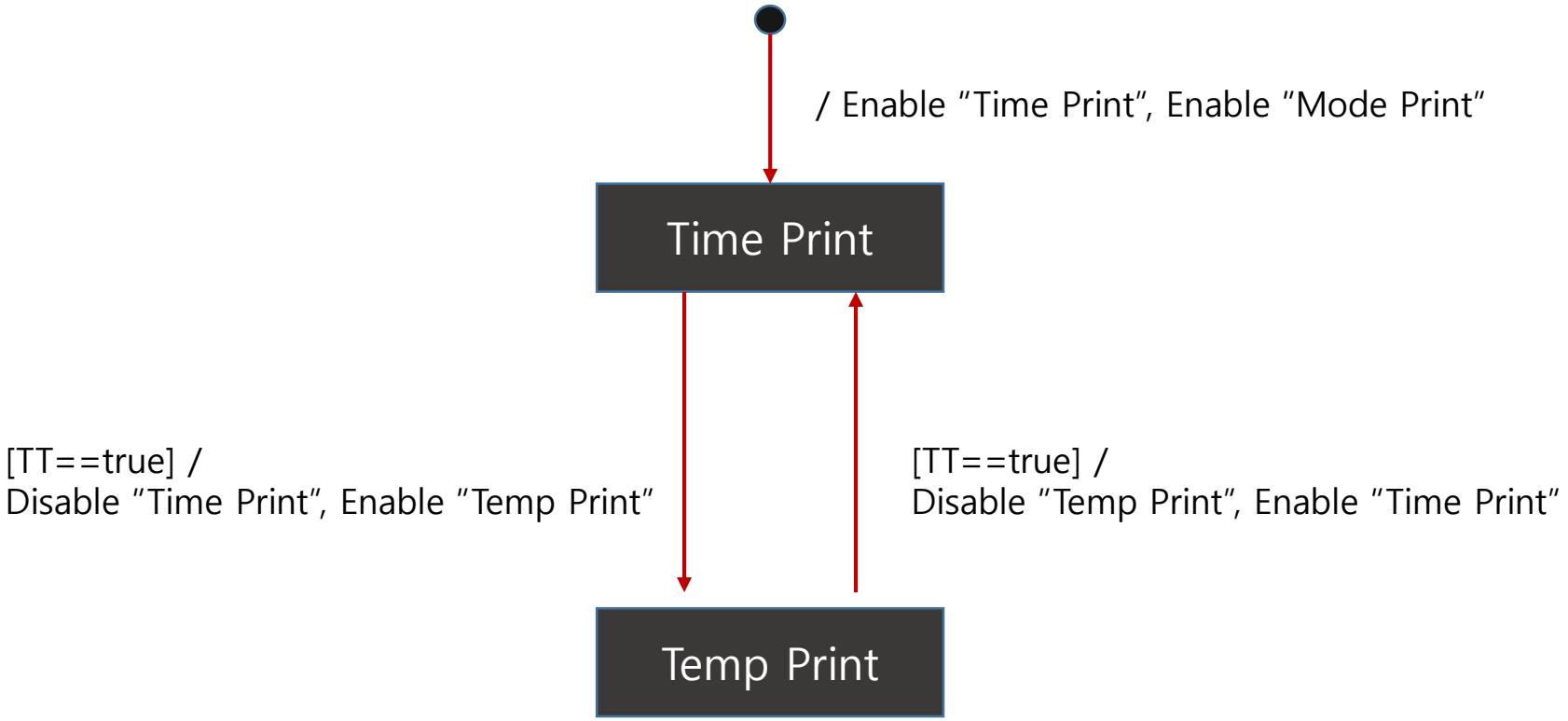
Reference No.	2.1.19
Name	Temp Print
Input	Enable, Disable
Output	Time Display Command (Bool)
Process Description	Temp Print에 Enable이 들어오면 Time_Temp Display Interface에게 Temp을 Display하라는 Temp Display Command를 보낸다. Disable이 들어오면 끄라는 Temp Display Command를 보낸다.

DFD Level 3 – 2.1 Process Specification



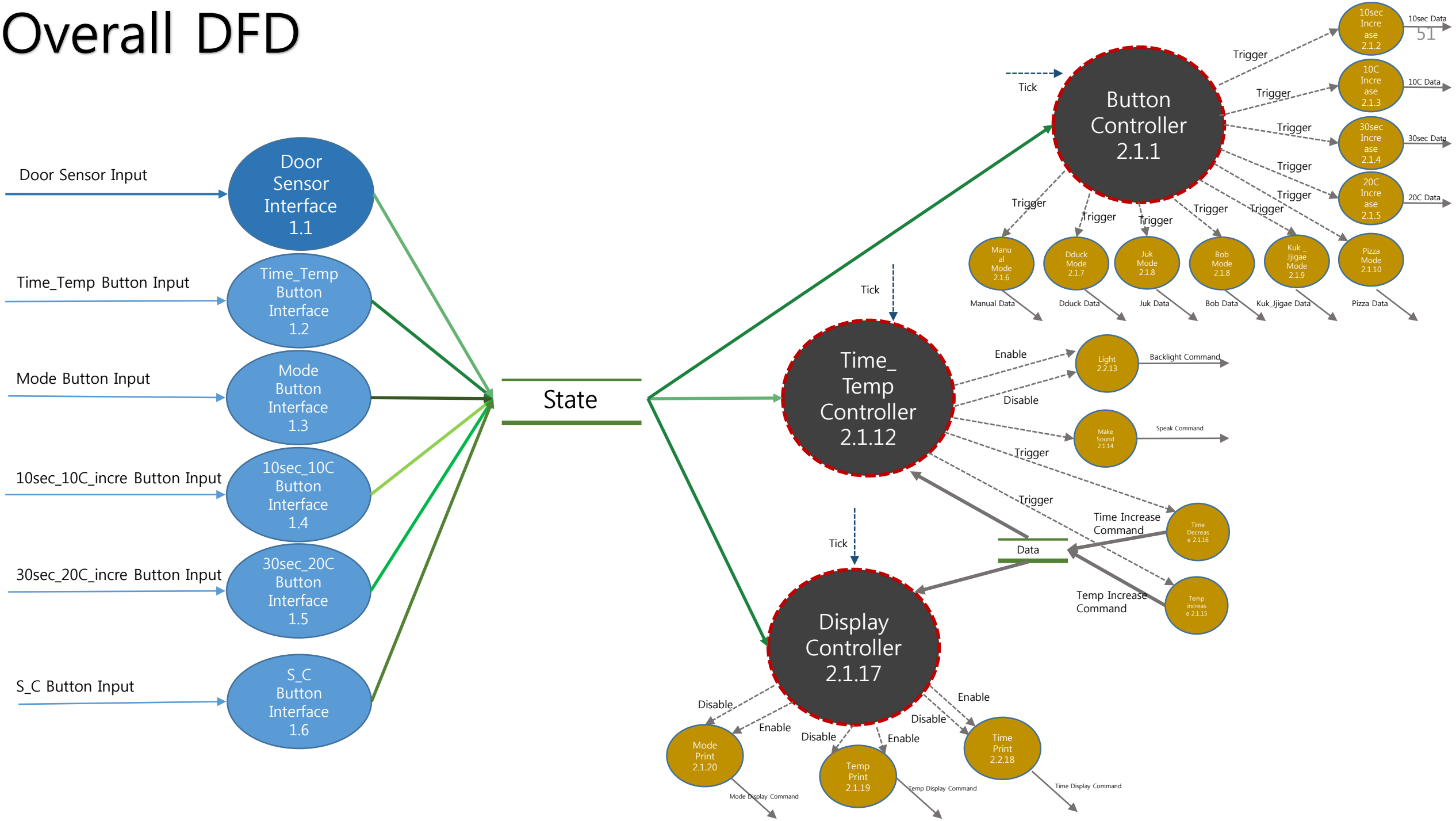
Reference No.	2.1.20
Name	Mode Print
Input	Enable, Disable
Output	Mode Display Command (Bool)
Process Description	Time Print에 Enable이 들어오면 Mode Display Interface에게 Mode를 Display하라는 Mode Display Command를 보낸다. Disable이 들어오면 끄라는 Mode Display Command를 보낸다.

DFD Level 3 - 2.1 (3/3) [State Transition Diagram for Display Control 2.1.15]



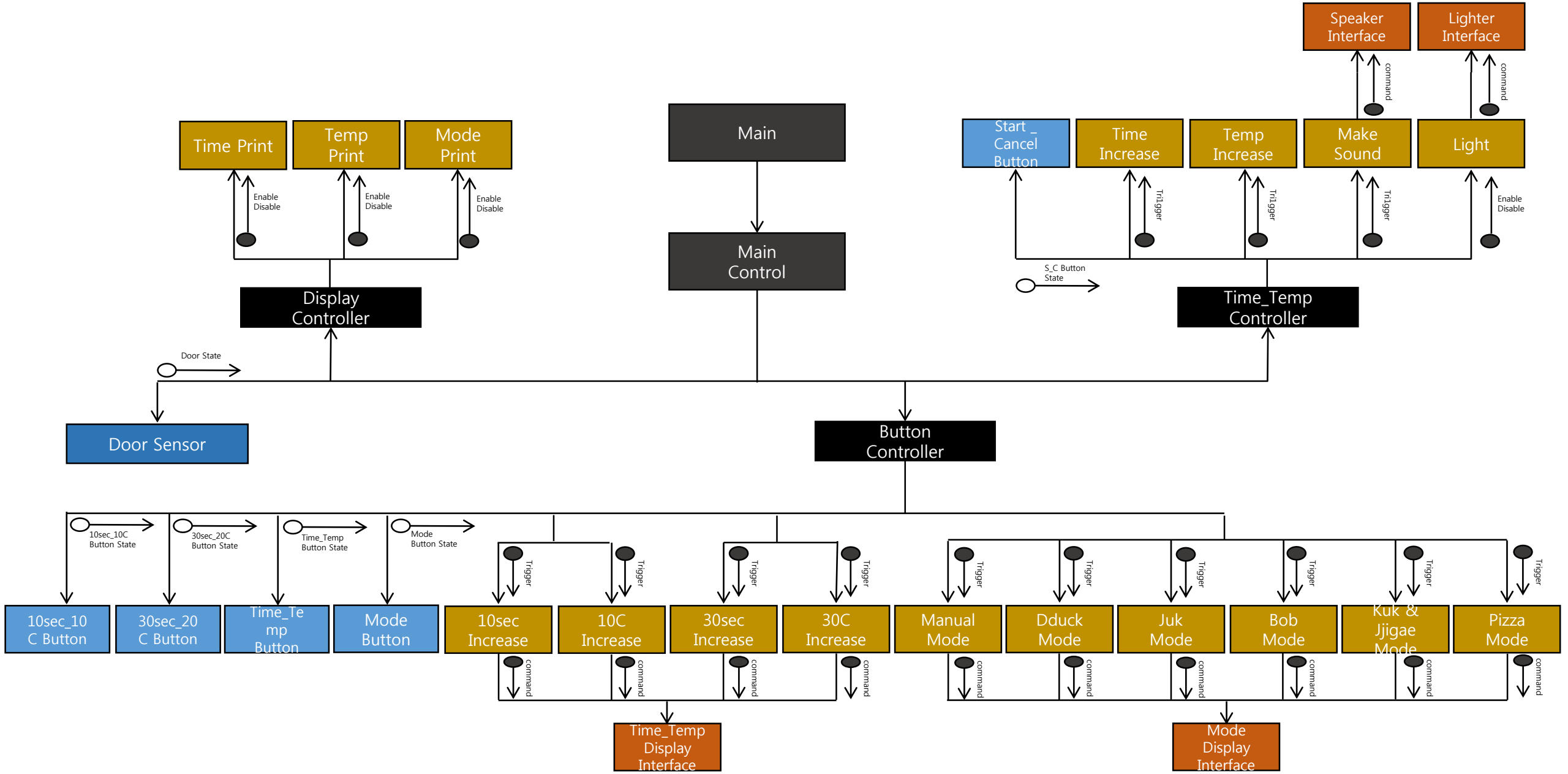
TT : 시간온도모드 (t = 1, f = 0)

Overall DFD

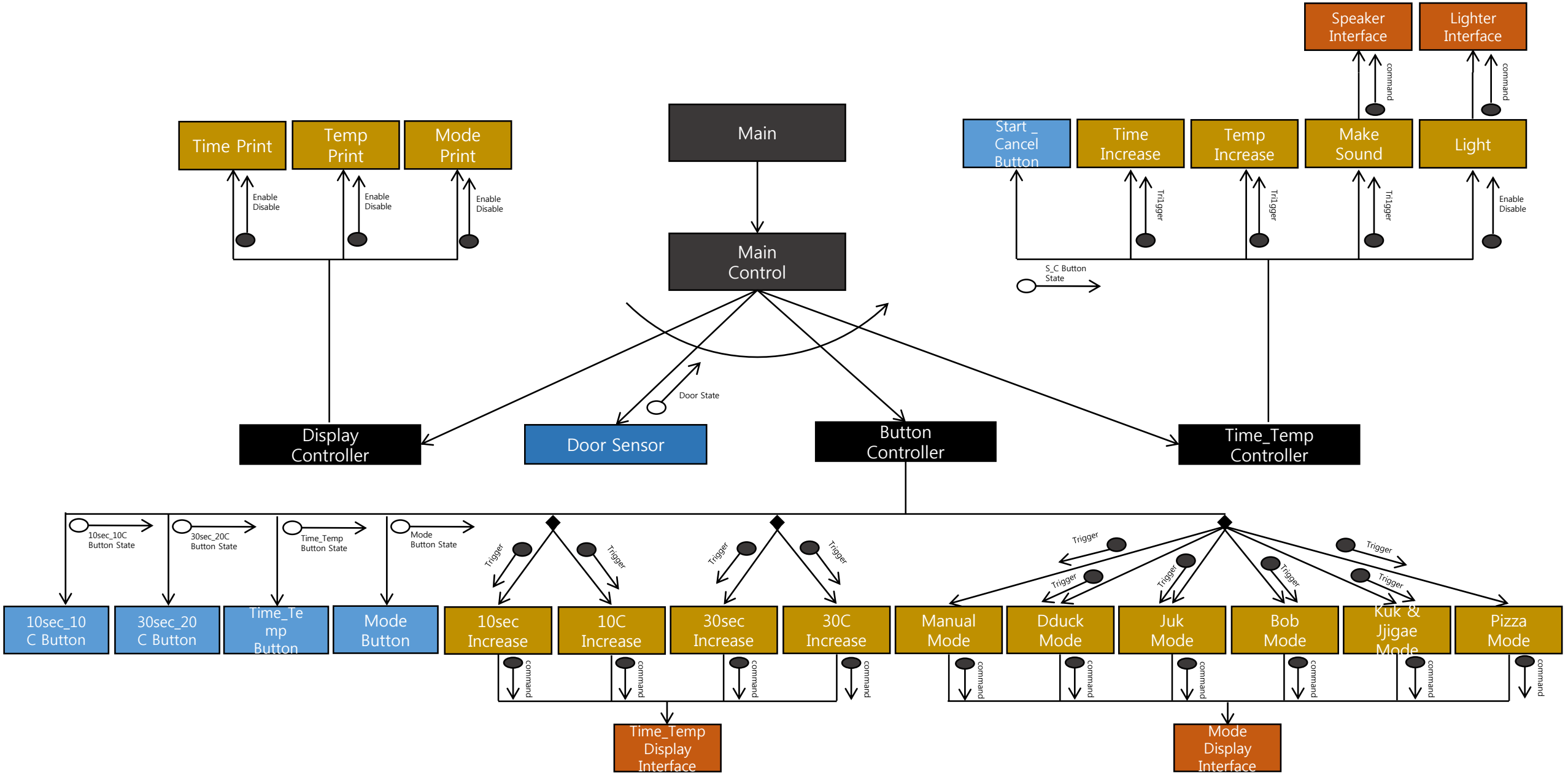

Door Sensor Input
Door Sensor Interface 1.1
Time_Temp Button Input
Time_Temp Button Interface 1.2
Mode Button Input
Mode Button Interface 1.3
10sec_10C_incre Button Input
10sec_10C Button Interface 1.4
30sec_20C_incre Button Input
30sec_20C Button Interface 1.5
S_C Button Input
S_C Button Interface 1.6
State
Tick
Button Controller 2.1.1
Trigger
10sec Increase 2.1.2
10sec Data
10C Increase 2.1.3
10C Data
30sec Increase 2.1.4
30sec Data
20C Increase 2.1.5
20C Data
Manual Mode 2.1.6
Manual Data
Dduck Mode 2.1.7
Dduck Data
Juk Mode 2.1.8
Juk Data
Bob Mode 2.1.8
Bob Data
Kuk _ Jjigae Mode 2.1.9
Kuk_Jjigae Data
Pizza Mode 2.1.10
Pizza Data
Time_ Temp Controller 2.1.12
Enable
Disable
Light 2.2.13
Backlight Command
Make Sound 2.1.14
Speak Command
Time Increase Command
Time Decrease 2.1.16
Temp Increase Command
Temp increase 2.1.15
Data
Display Controller 2.1.17
Mode Print 2.1.20
Mode Display Command
Temp Print 2.1.19
Temp Display Command
Time Print 2.2.18
Time Display Command

Structured Design

Structured Charts – Microwave Oven System (Basic)

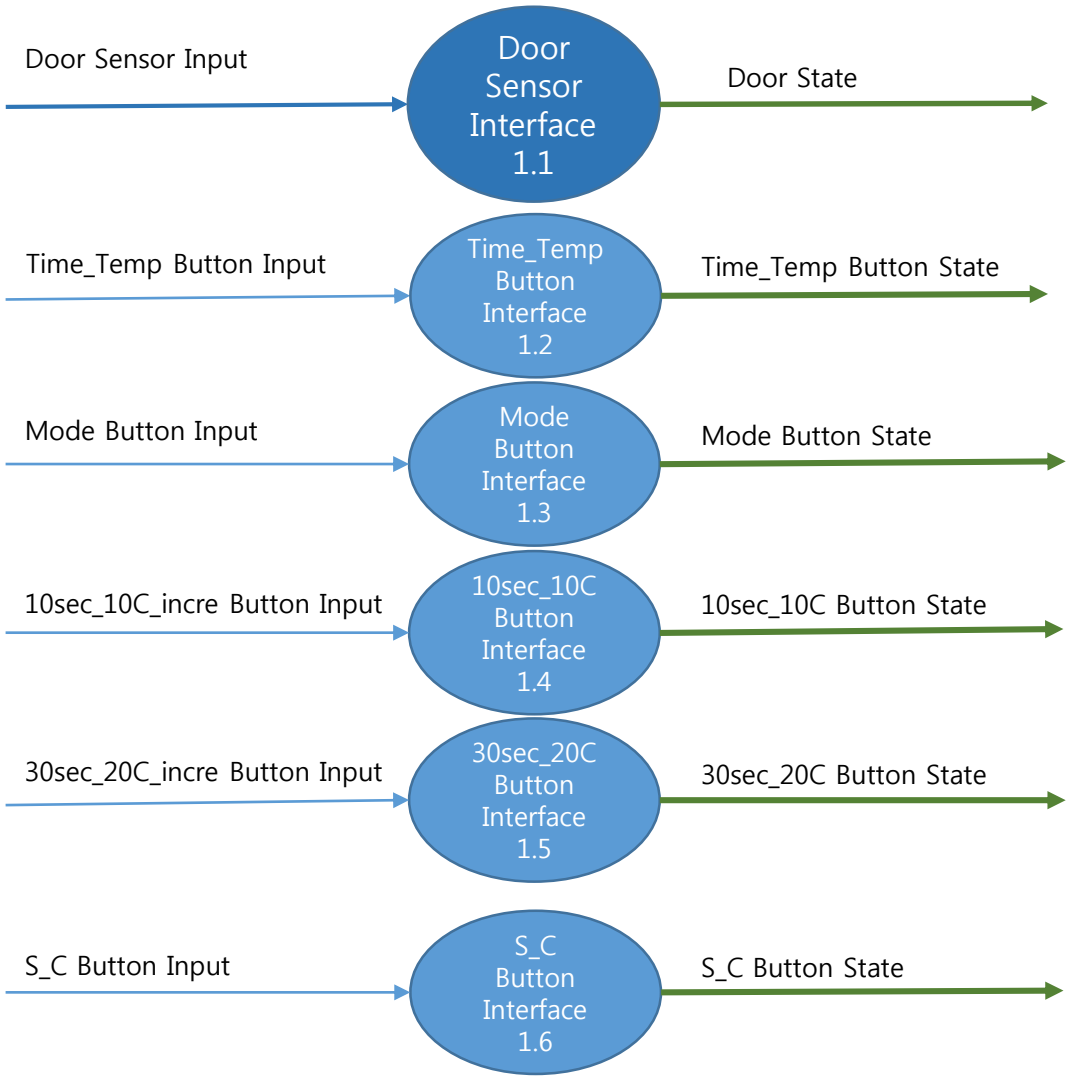


Structured Charts – Microwave Oven System (Advanced)



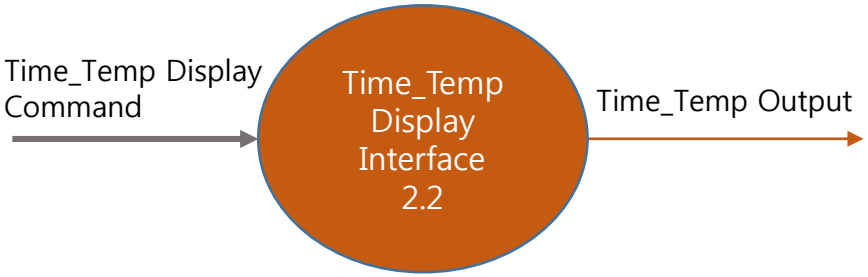
Process & Code

Prime Process – Microwave Oven System



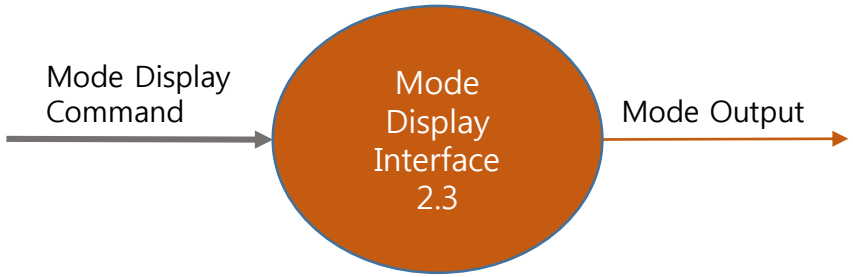

```
void DoorSenSorInterface (Bool DoorSensorInput){
s.DoorState = DoorSensorInput;
}

void Time_TempInterface (Bool Time_TempInput){
s.Time_TempButtonState = Time_TempInput;
}

void ModeButtonInterface (Bool ModeButtonInput){
s.ModeButtonState = ModeButtonInput;
}

void _10sec_10CButtonInterface (Bool
_10sec_10C_increButtonInput){
s._10sec_10CButtonState = _10sec_10C_increButtonInput;
}

void _30sec_20CButtonInterface (Bool
_30sec_20C_increButtonInput){
s._30sec_20CButtonState = _30sec_20C_increButtonInput;
}

void S_CButtonInterface (Bool S_CButtonInput){
s.S_CButtonState = S_CButtonInput;
}
```

Prime Process – Microwave Oven System



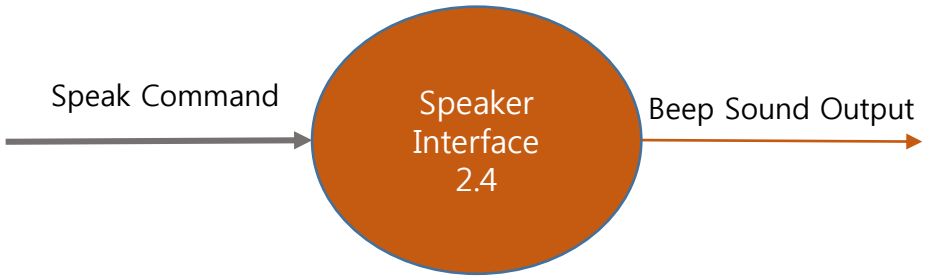

```
void Time_TempDisplayInterface(Bool TimeDisplayCommand,
Bool TempDisplayCommand) {
system("clear");
if ( (T == TimeDisplayCommand) && (F ==
TempDisplayCommand)) {
if ( d.ctime == -1 ) {
printf("%d:%02d₩t", (d.stime)/60, (d.stime)%60);
}
else {
printf("%d:%02d₩t", (d.ctime)/60, (d.ctime)%60);
}
}
else {
if ( d.ctemp == -1 ) {
printf("%dC₩t", d.stemp);
}
else {
printf("%dC₩t", d.ctemp);
}
}
}
```

Prime Process – Microwave Oven System

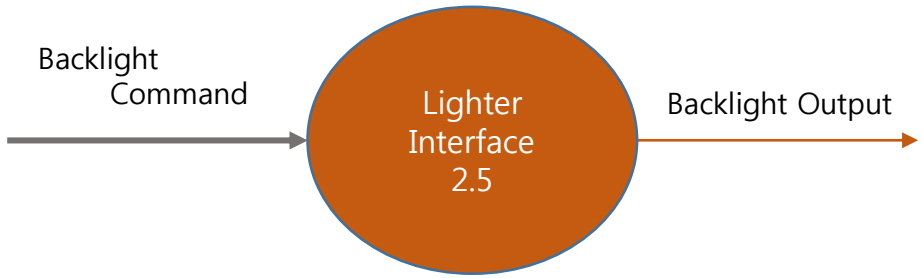



```
void ModeDisplayInterface(Bool ModeDisplayCommand) {

if (ModeDisplayCommand == T) {
printf("Mode : ");

switch (d.i_cmode) {
case 0:
printf("Manual₩n");
break;
case 1:
printf("Dduck₩n");
break;
case 2:
printf("Juk₩n");
break;
case 3:
printf("Bob₩n");
break;
case 4:
printf("Kuk_Jjigae₩n");
break;
case 5:
printf("Pizza₩n");
break;
default :
printf("Error, in switch of ModeDisplayInterface.₩n");
}
}
printf("₩n");
}
```

Prime Process – Microwave Oven System

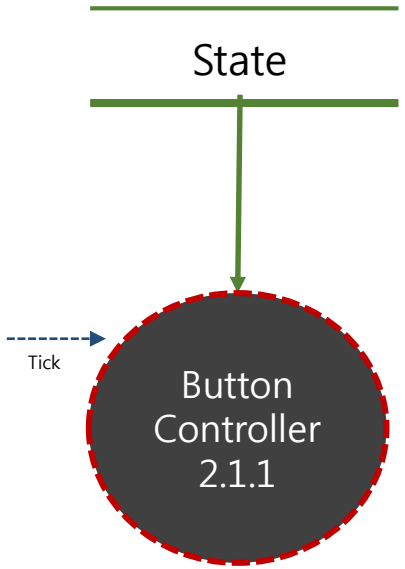



```
void LighterInterface(Bool BacklightCommand) {
if ( (T == BacklightCommand) ) {
printf("%c[1;33m",27);
}
else {
printf("%c[0m", 27);
}
}
```



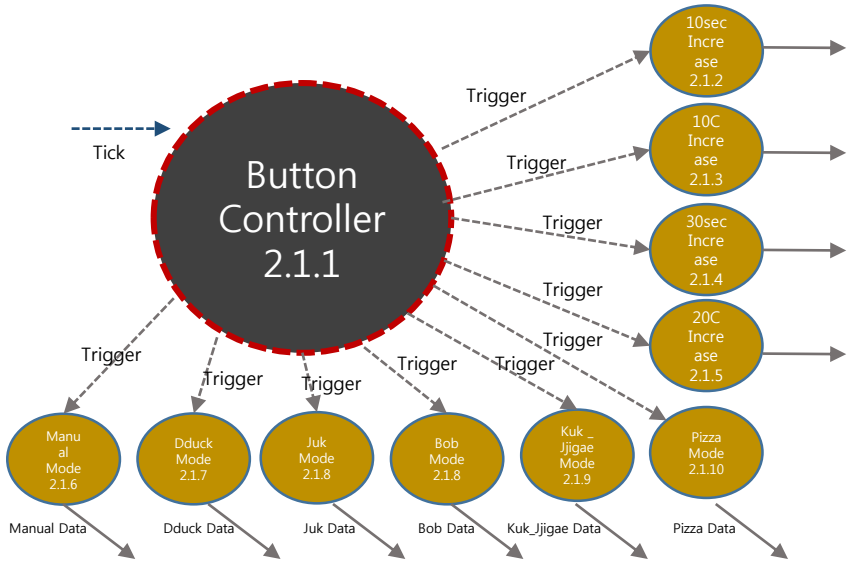

```
void SpeakerInterface (Bool SpeakCommand) {
int i;

if (T == SpeakCommand) {
printf("₩a");
sleep(1);
printf("₩a");
sleep(1);
printf("₩a");
sleep(1);
SC=1;
}
}
```

Process & Code (Button Controller)




```
void buttoncontroller (struct State *s){

static int state = 1;// odd : Time, even : Temp
static int mode = 0;// Mode
Bool trigger = F;

if(s->Time_TempButtonState == T){
state++;
ABLE++; // sum
s->Time_TempButtonState = F;
if ( (state % 2) != 0 ) {
d.stemp = 20;
}
else {
d.stime = 0;
}
}
///
if( ( s->ModeButtonState == T ) && ( (state % 2) != 0 ) ){
mode++;
mode %= 6;
if ( mode == 0 ) {
d.stime = 0;
}
s->ModeButtonState = F;
}
///
if (mode == 0) {
trigger = T;
ManualMode(trigger);

if( ( s->_10sec_10CButtonState == T ) && ( (state %
2) != 0 ) ) {
s->_10sec_10CButtonState = F;
trigger = T;
_10secIncrease(trigger);
}
if( ( s->_30sec_20CButtonState == T ) && ( (state %
2) != 0 ) ){
s->_30sec_20CButtonState = F;
trigger = T;
_30secIncrease(trigger);
}
////
if( ( s->_10sec_10CButtonState == T ) && ( (state % 2)
== 0 ) ) {
s->_10sec_10CButtonState = F;
trigger = T;
_10CIncrease(trigger);
}
if( ( s->_30sec_20CButtonState == T ) && ( (state % 2)
== 0 ) ){
s->_30sec_20CButtonState = F;
trigger = T;
_20CIncrease(trigger);
}
}
else {
switch(mode){
case 1: // Dduck Mode
trigger = T;
DduckMode(trigger);
break;
case 2: // Juk Mode
trigger = T;
JukMode(trigger);
break;
case 3: // Bob Mode
trigger = T;
BobMode(trigger);
break;
case 4: // Kuk_Jjigae Mode
trigger = T;
Kuk_JjigaeMode(trigger);
break;
case 5: // Pizza Mode
trigger = T;
PizzaMode(trigger);
break;
default :
printf("Error, in switch of
burroncontroller.₩n");
break;
} // switch
}
}
```

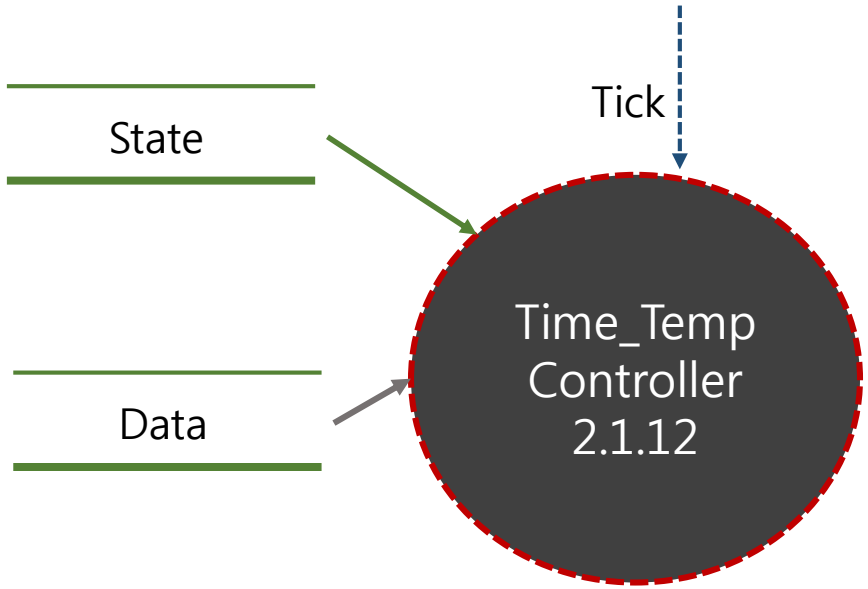
Process & Code




```
void _10secIncrease (Bool trigger) {
if(T == trigger) {
d.stime += 10;
d.stime %= 600;
}
}
void _30secIncrease (Bool trigger) {
if(T == trigger) {
d.stime += 30;
d.stime %= 600;
}
}
void _10CIncrease (Bool trigger) {
if(T == trigger) {
d.stemp += 10;
if(d.stemp > 90) {
d.stemp = (d.stemp % 70);
}
}
}
void _20CIncrease (Bool trigger) {
if(T == trigger)
d.stemp += 20;
if(d.stemp > 90) {
d.stemp = (d.stemp % 70);
}
}
```

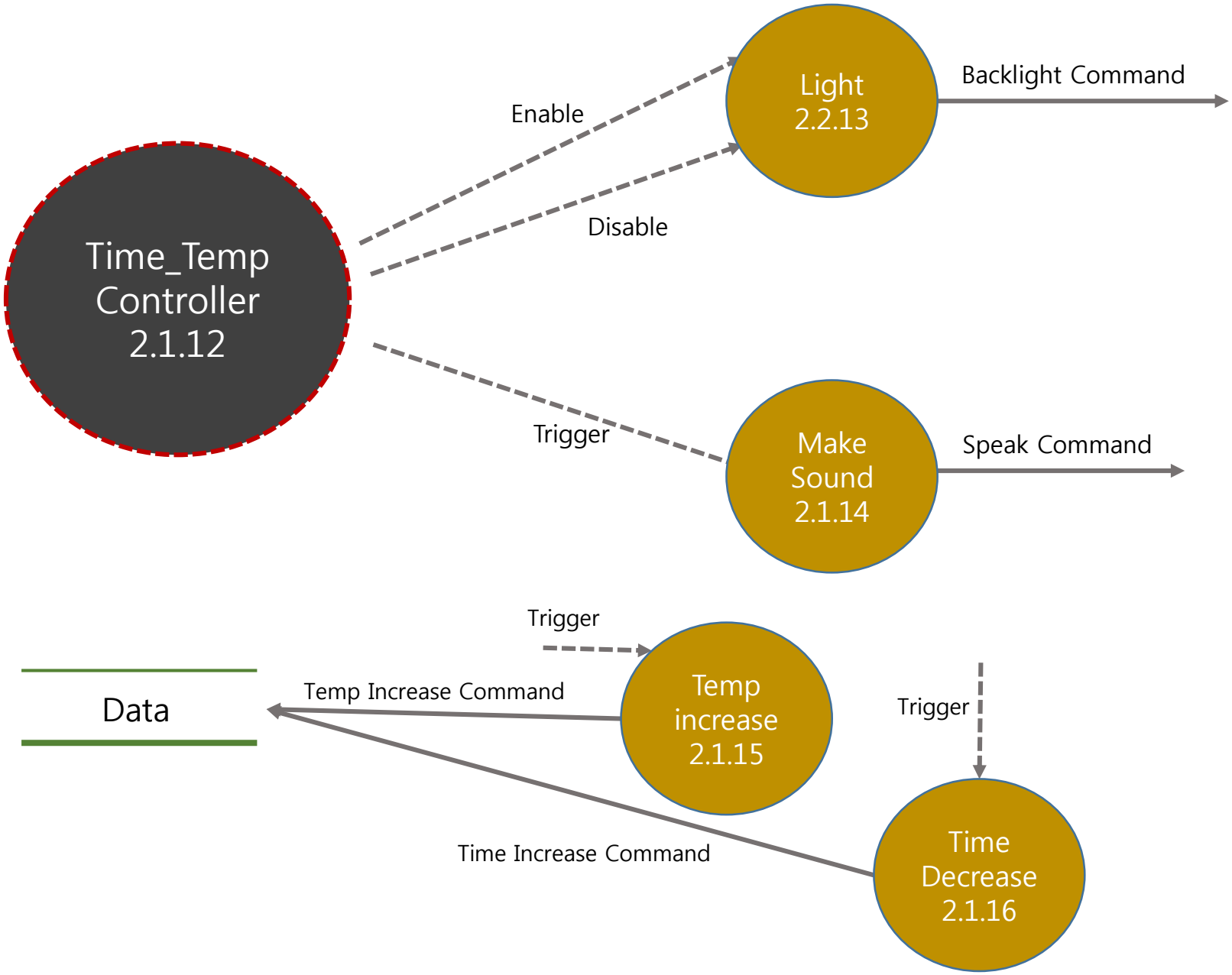
```
void ManualMode (Bool trigger) {
if(T == trigger) {
d.i_cmode = 0;
d.c_cmode = "Manual";
}
}
void DduckMode (Bool trigger) {
if(T == trigger) {
d.stime = 60;
d.i_cmode = 1;
d.c_cmode = "Dduck";
}
}
void JukMode (Bool trigger) {
if(T == trigger) {
d.stime = 90;
d.i_cmode = 2;
d.c_cmode = "Juk";
}
}
void BobMode (Bool trigger) {
if(T == trigger) {
d.stime = 120;
d.i_cmode = 3;
d.c_cmode = "Bob";
}
}
void Kuk_JjigaeMode (Bool trigger) {
if(T == trigger) {
d.stime = 300;
d.i_cmode = 4;
d.c_cmode = "Kuk_Jjigae";
}
}
void PizzaMode (Bool trigger) {
if(T == trigger) {
d.stime = 120;
d.i_cmode = 5;
d.c_cmode = "Pizza";
}
}
```

Process & Code (Time_Temp Controller)



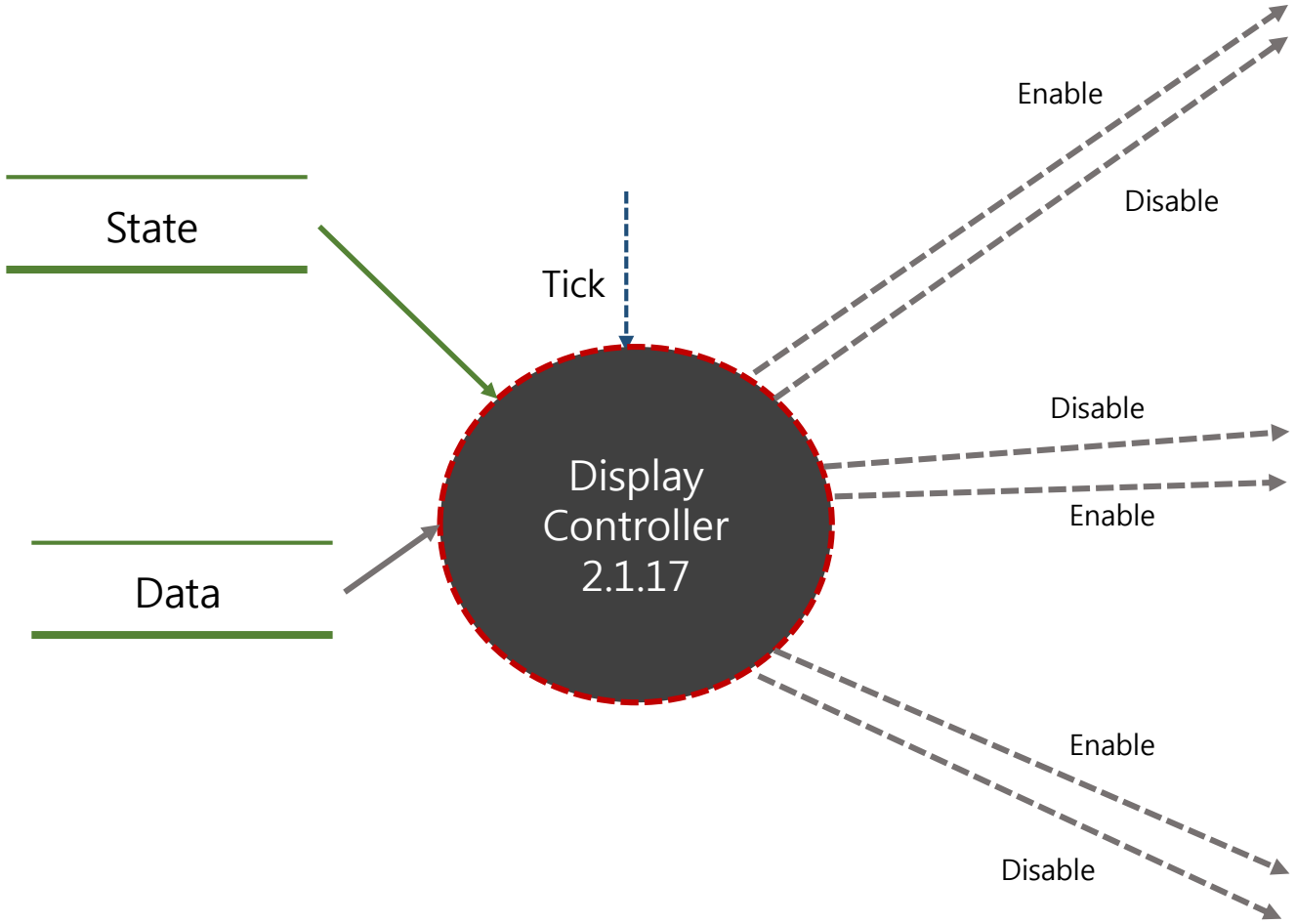

```
void Time_TempController(struct Data *d, struct State *s){

Bool trigger = F;
Bool enable = F;
Bool disable = F;

if ( s->DoorState == T ) {
s->DoorState = F;
DOOR++;
if ( (DOOR % 2) == 0 ) {
enable = T;
}
else {
disable = T;
}
Light(enable, disable);
}
else {
if ( (s->S_CButtonState == T) &&
((DOOR%2) != 0) ) {
if( (ABLE % 2) != 0 ){ // When ModeState is
Time.
if (d->ctime == 0 ) {
trigger = T;
disable = T;
Light(enable, disable);
Make_Sound(trigger);
}
else {
trigger = T;
enable = T;
Light(enable, disable);
TimeDecreasing(trigger);
}
}
else if( (ABLE % 2) == 0 ){// When
ModeState is Temp.
if (d->ctemp >= d->stemp ) {
trigger = T;
disable = T;
Light(enable, disable);
Make_Sound(trigger);
}
else {
trigger = T;
enable = T;
Light(enable, disable);
TempIncreasing(trigger);
}
}
} // if
else {
s->S_CButtonState = F;
}
} // else
}
```

Process & Code




```
void TimeDecreasing (Bool trigger){
DisplayController(&d, &s);
sleep(1);
d.ctime = d.ctime - 1;
}

void TempIncreasing (Bool trigger){
DisplayController(&d, &s);
sleep(3);
d.ctemp = d.ctemp + 10;
}

void Make_Sound (Bool trigger){
Bool SpeakCommand = F;
if (T == trigger) {
SpeakCommand = T;
SpeakerInterface(SpeakCommand);
}
else {
SpeakCommand = F;
SpeakerInterface(SpeakCommand);
}
}
void Light(Bool enable, Bool disable) {
Bool BacklightCommand = F;
if ( (T == enable) && (F == disable) ) {
BacklightCommand = T;
LighterInterface(BacklightCommand);
}
else {
LighterInterface(BacklightCommand);
}
}
```

Process & Code (Display Controller)




```
void DisplayController(struct Data *d, struct State *s) {
        Bool enable = F;
        Bool disable = F;

        if ( (ABLE % 2) != 0 ) {
                enable = T;
                TimePrint(enable, disable);
                ModePrint(enable, disable);
        }
        else {
                enable = T;
                TempPrint(enable, disable);
                ModePrint(enable, disable);
        }
}
```

Process & Code

State

```
typedef struct State {
        Bool DoorState;
        Bool Time_TempButtonState;
        Bool ModeButtonState;
        Bool _10sec_10CButtonState;
        Bool _30sec_20CButtonState;
        Bool S_CButtonState;
};
```

Process & Code

Data

```
typedef struct Data {
        int stime;
        int stemp;
        int ctime;
        int ctemp;
        //
        int i_cmode;
        char *c_cmode;
};
```

Thank you☺