

Auto Drive Robot

Final Presentation

T8

200611451 강용진

200611462 김지용

200611471 박지성

200611481 송준현

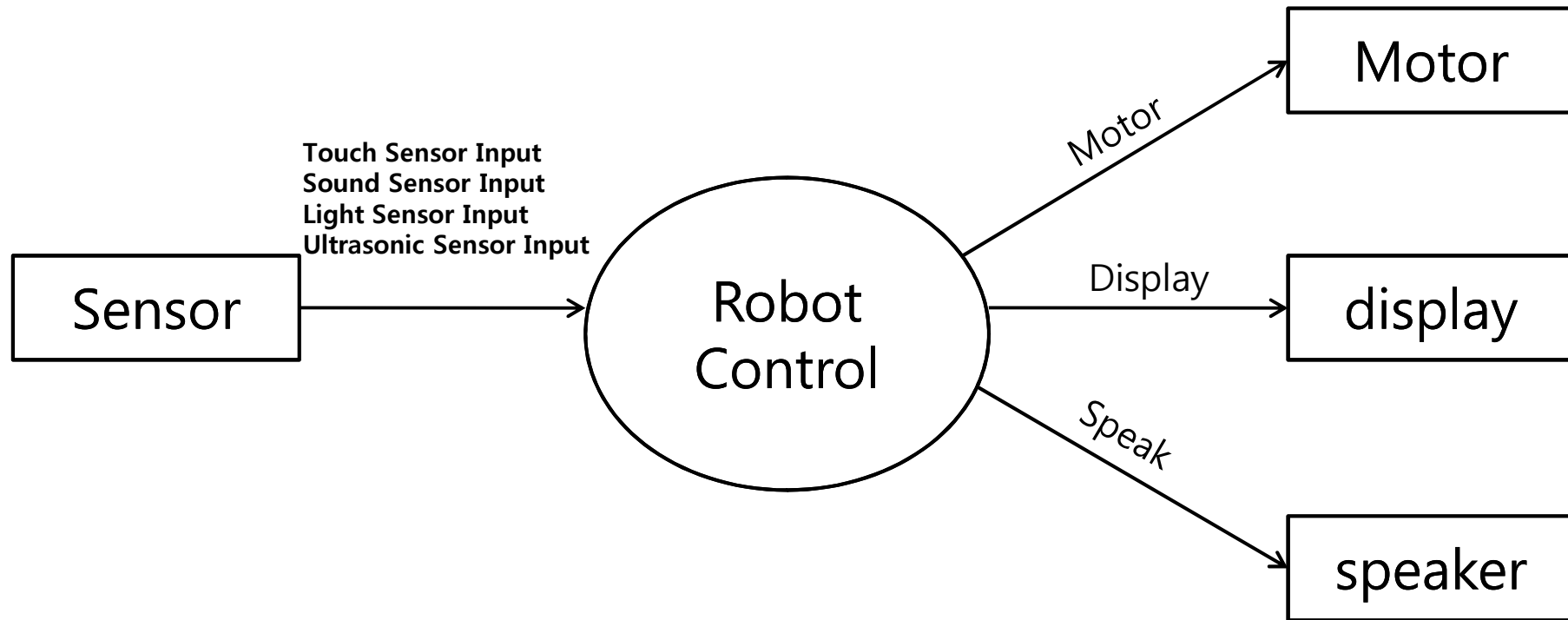
작성일 : 2009. 12. 14

Modified Article

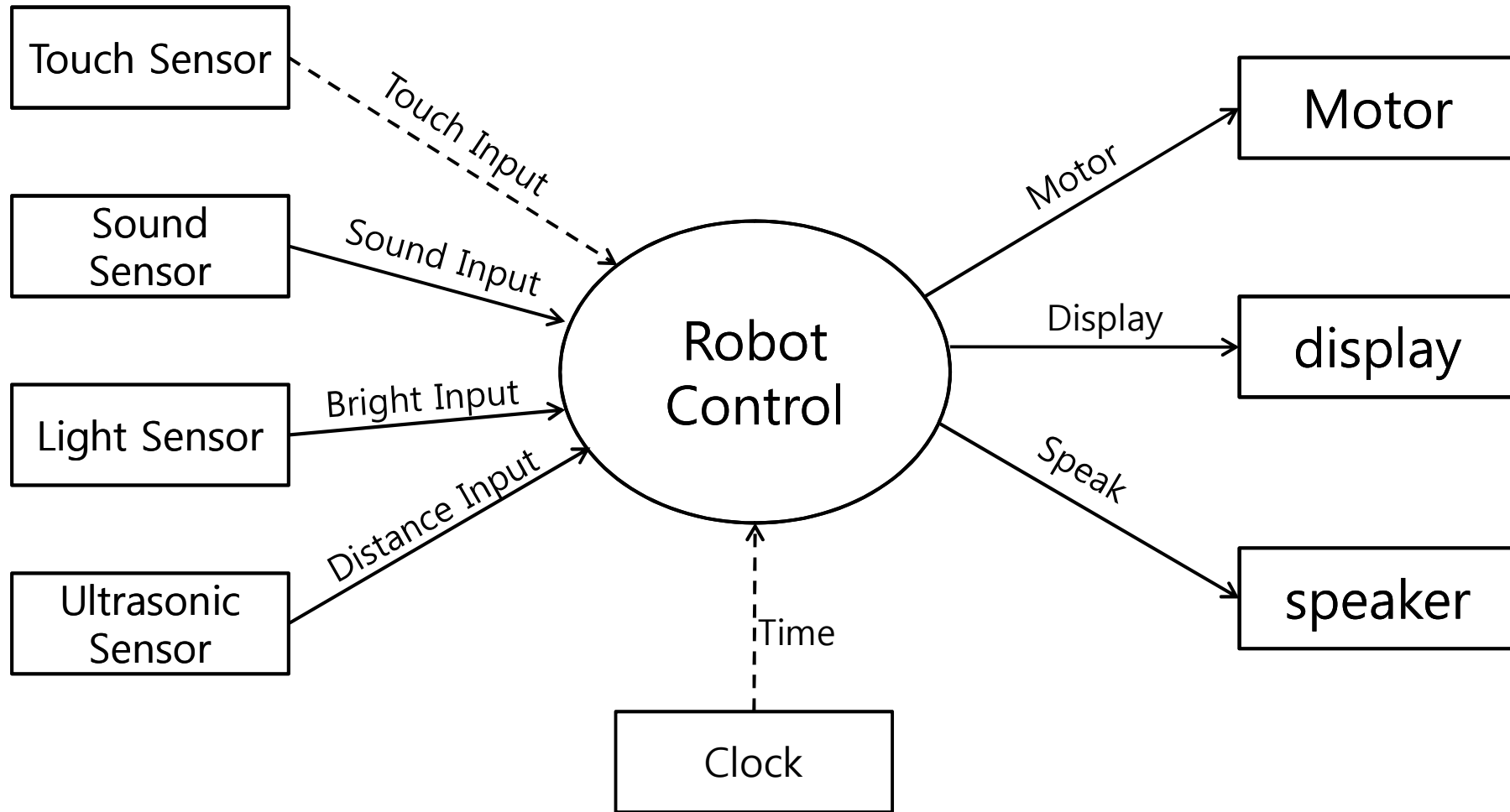
- No modify

Structured Analysis

System Context Diagram



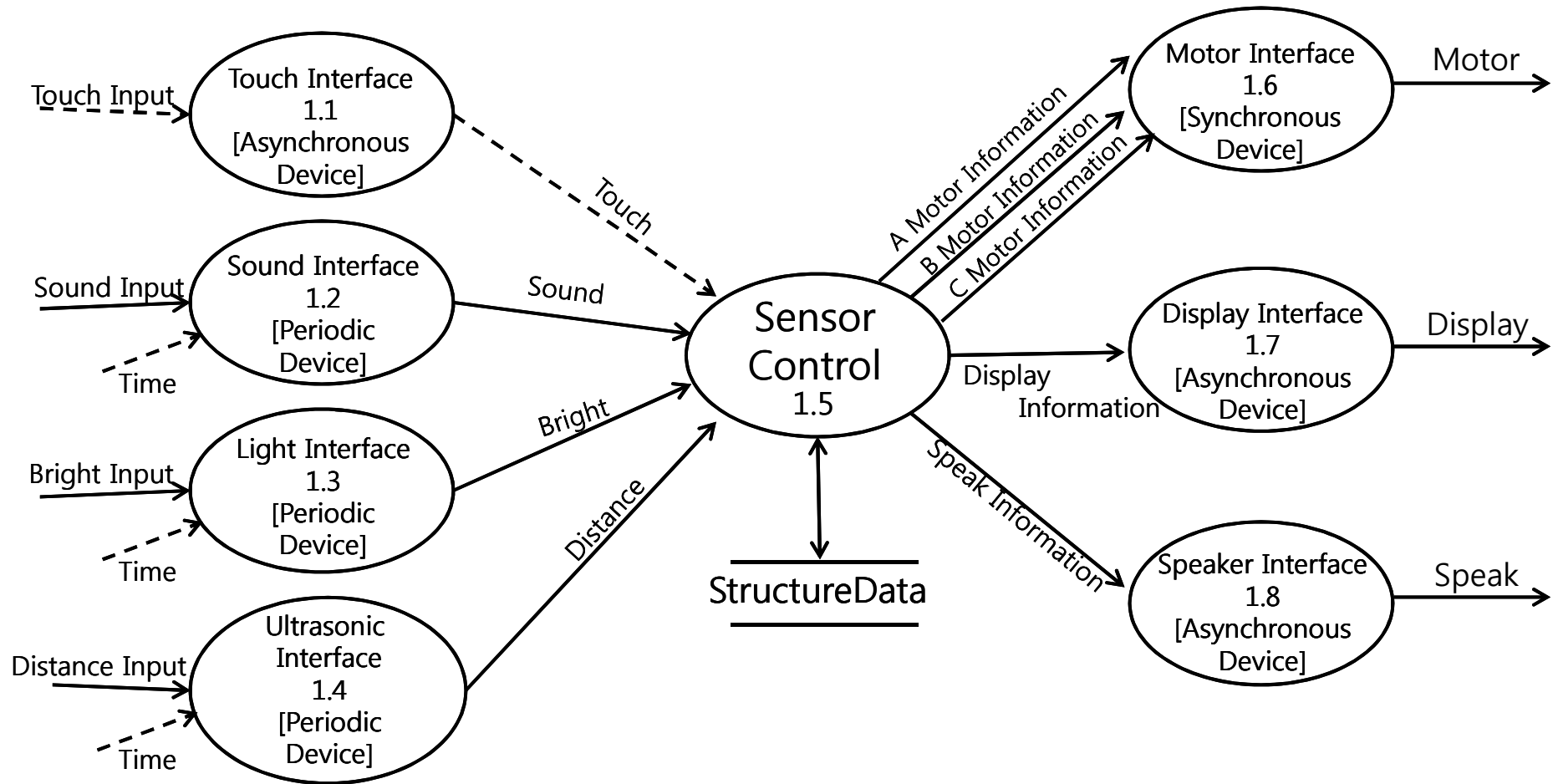
DFD Level 0



DFD Level 0 – Information Table

Input/ Output Event		Description	Format / Type
Input	Touch Input	터치 센서를 사용자가 눌렀음을 나타냄.	True/False Interrupt
	Sound Input	10ms의 주기로 Sound Sensor의 소리 정보를 나타냄.	Int/ Periodic
	Bright Input	10ms의 주기로 Light Sensor의 밝기 정보를 나타냄.	Int/ Periodic
	Distance Input	10ms의 주기로 Ultrasonic Sensor를 통해 들어오는 사물과의 거리정보를 나타냄.	Int/ Periodic
	Time	타이머 Interrupt로서 10ms의 주기로 입력됨.	trigger/ Interrupt
Output	Motor	로봇의 움직임을 제어할 모터의 제어정보를 나타냄. 속도와 방향정보를 가진다.	A/B/C
	Speak	Speaker를 통해 외부에 출력할 소리 정보를 나타냄.	Good Morning Night/Thank you
	Display	Display를 통해 외부에 출력할 글씨 정보를 나타낸다.	Working/ Sleeping

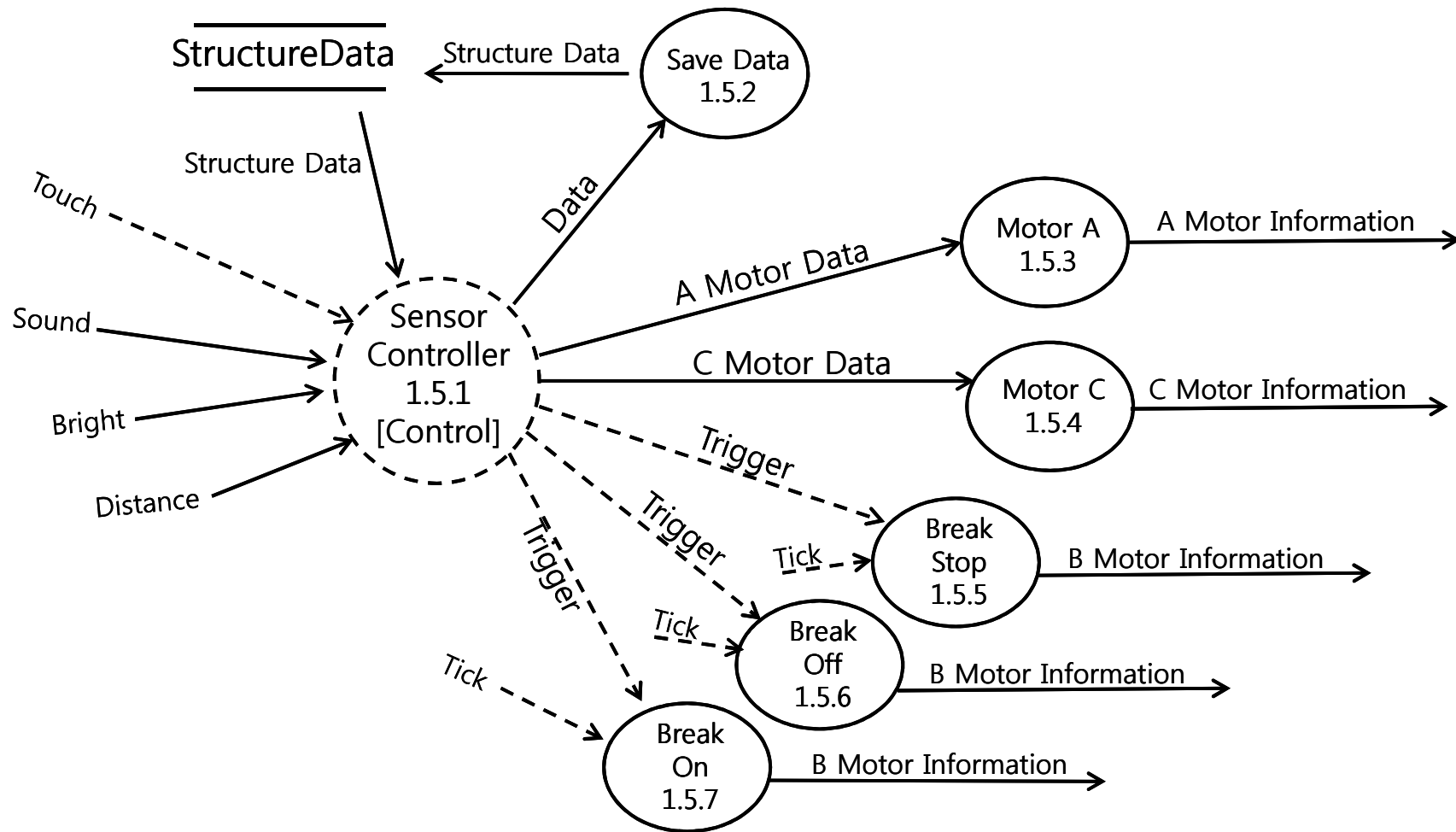
DFD Level 1



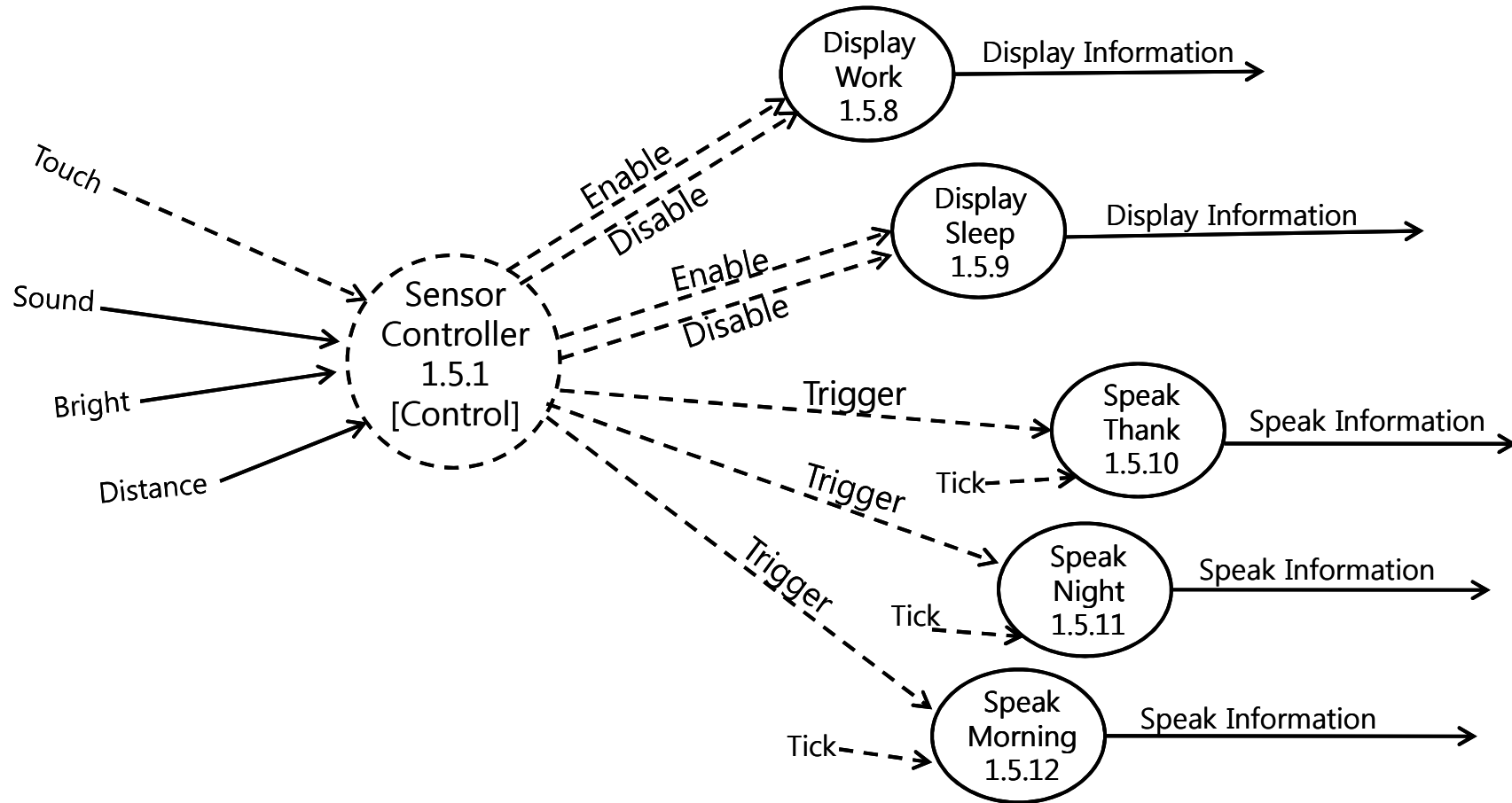
DFD Level 1 – Structure Data

이름	타입	설명
Speed	int	현재의 속도를 저장하여 터치 입력이 들어왔을 때, 다음 속도의 값을 Motor에 전달할 수 있도록 한다. (0~100이하의 모터의 최대 속도에 대한 비율 값을 가진다. Touch가 들어올 때마다 10씩 증가하며, 100이 되면 다시 10으로 값을 돌린다.)
Direction Count	int	현재까지 Turn한 횟수를 저장하여, 장애물을 감지했을 때, 오른쪽, 왼쪽, 180도 턴의 동작 중 하나를 결정할 수 있게 한다.
Touch Count	int	Touch 센서의 버튼이 눌릴 때 마다 count를 증가하여, 홀수 / 짝수를 속도의 증가 / 감소와 매칭을 시킨다.
Dark	int	Light값을 기준으로 40이하일 때의 로봇의 상태 (직진상태에서의 breaking 진입여부와 stop진입 여부)를 판단할 수 있도록 한다. (0일 때, 1이 들어오면 : forward -> breaking 1일 때, 1이 들어오면 : breaking -> stop 1일 때, 0이 들어오면 : breaking -> forward 0일 때, 0이 들어오면 : forward -> forward)
Touch Check	bool	터치의 입력여부를 저장하여 이전 값과 현재 값을 비교하여 속도를 계속 증가시킬 것인지, 감소시킬 것인지를 결정할 수 있게 한다.

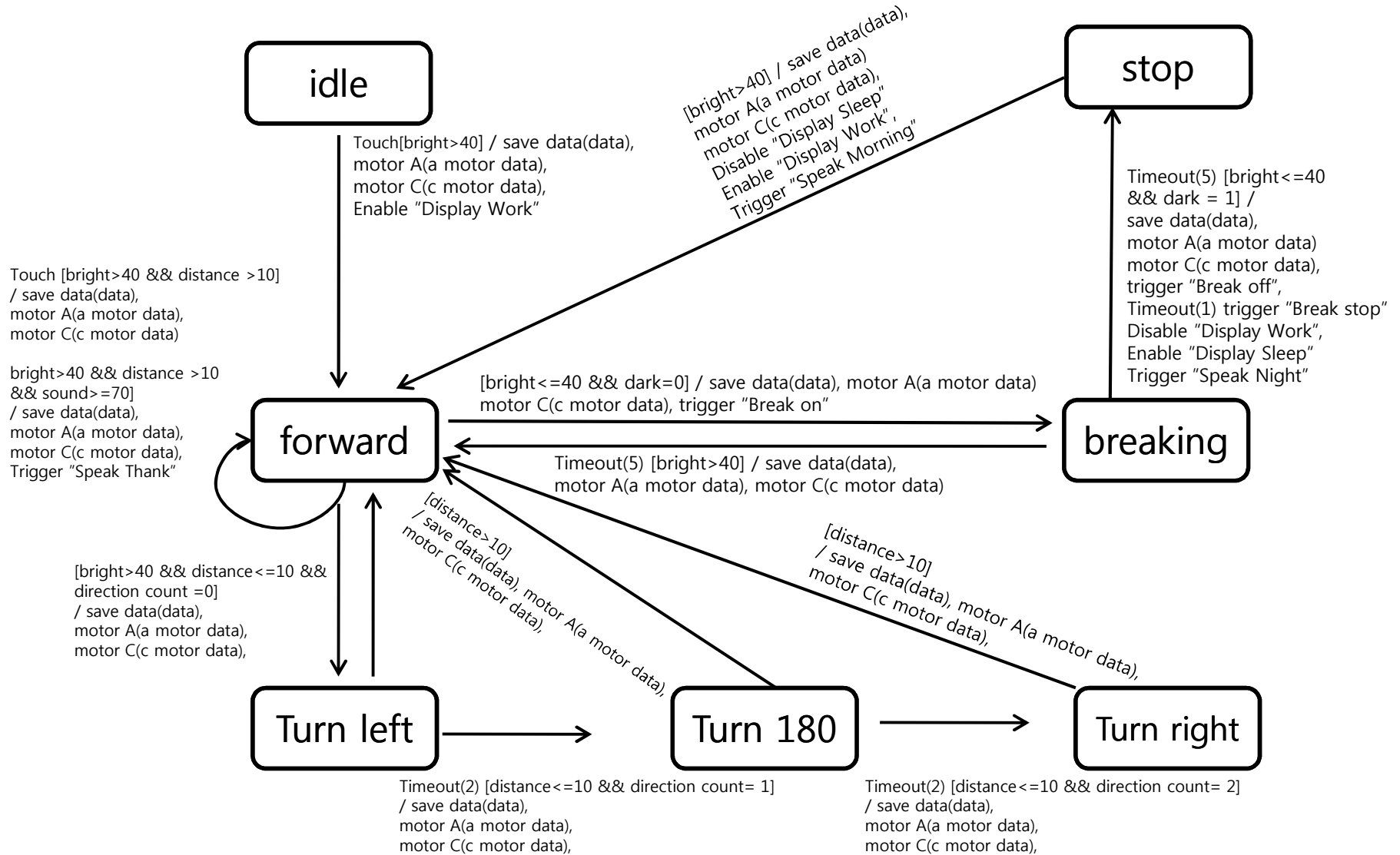
DFD Level 2 – Sensor Control 1.5



DFD Level 2 – Sensor Control 1.5 (계속)



State Machine – Sensor Controller 1.5.1



Process Specification

Reference No.	1.1
Stereotype	Asynchronous Function
Name	Touch Interface
Input	Touch Input
Output	(bool) Touch
Process Description	Touch Sensor에서 입력을 받으면 T/F로 구분하여 Controller에 보낸다. 눌렀을 때가 True이다.

Reference No.	1.2
Stereotype	Periodic Function
Name	Sound Interface
Input	Sound Input, Tick
Output	(int) Sound
Process Description	Sound Sensor에서 Sound Input을 받으면 정수로 변환하여 Controller에 보낸다. 범위는 0~100이다.

Process Specification (계속)

Reference No.	1.3
Stereotype	Periodic Function
Name	Light Interface
Input	Bright Input, Tick
Output	(int) Bright
Process Description	Light Sensor에서 Bright Input을 받으면 정수로 변환하여 Controller로 보낸다. 범위는 1~100이며 Bright는 Light Sensor에서 감지한 밝기 정보 데이터이다.

Reference No.	1.4
Stereotype	Periodic Function
Name	Ultrasonic Interface
Input	Distance Input, Tick
Output	(int) Distance
Process Description	Ultrasonic Sensor에서 Distance Input을 받아 장애물과의 거리를 측정한 뒤 Controller에게 보낸다

Process Specification (계속)

Reference No.	1.5.1
Stereotype	Control
Name	Controller
Input	(bool)Touch, (int)Sound, (int)Bright, (int) Distance, Structure Data
Output	A,C Motor Data(bool, int), Speed Data, Trigger
Process Description	입력센서와 Structure Data 로부터 들어온 정보들을 바탕으로 행동을 결정하고 행동을 수행하기 위한 정보(데이터)들을 다음 단계의 프로세스에 전달한다.
Reference No.	1.5.2
Stereotype	Synchronous Function
Name	Save Data
Input	(bool) Touch, (int) Sound, (int) Bright, (int) Distance, (bool) Touch_check
Output	Structure Data(bool, int, int, int, bool)
Process Description	Controller에서 들어온 Structure 데이터를 Structure에 저장하여 다음 단계의 행동을 결정하는데 사용하도록 한다.

Process Specification (계속)

Reference No.	1.5.3
Stereotype	Synchronous Function
Name	Motor A
Input	A Motor Data(bool, int)
Output	A Motor Information(int, bool, int)
Process Description	Controller로부터 받은 A Motor Data를 Motor Interface에 보낸다. A Motor의 진행방향과 속도가 포함된다.

Reference No.	1.5.4
Stereotype	Synchronous Function
Name	Motor C
Input	C Motor Data(bool, int)
Output	C Motor Information(int, bool, int)
Process Description	Controller로부터 받은 C Motor Data를 Motor Interface에 보낸다. C Motor의 진행방향과 속도가 포함된다.

Process Specification (계속)

Reference No.	1.5.5
Stereotype	Synchronous Function
Name	Brake Stop
Input	Trigger, Tick
Output	B Motor Information(int, bool, int)
Process Description	Controller로부터 Trigger가 들어오면 Motor Interface에 B Motor Information을 보낸다. B Motor의 속도가 0이 되도록 한다.

Reference No.	1.5.6
Stereotype	Synchronous Function
Name	Brake Off
Input	Trigger, Tick
Output	B Motor Information(int, bool, int)
Process Description	Controller로부터 Trigger가 들어오면 Motor Interface에 B Motor Information을 보낸다. B Motor의 속도가 20, 방향은 뒤쪽으로 회전하도록 한다.

Process Specification (계속)

Reference No.	1.5.7
Stereotype	Synchronous Function
Name	Brake On
Input	Trigger, Tick
Output	B Motor Information(int, bool, int)
Process Description	Controller로부터 Trigger가 들어오면 Motor Interface에 B Motor Information을 보낸다. B Motor의 속도가 100, 방향은 앞으로 회전하도록 한다.

Reference No.	1.5.8
Stereotype	Synchronous Function
Name	Display Work
Input	Enable / Disable
Output	Display Information(string)
Process Description	Controller로부터 Enable/Disable을 받아 Display Interface에 Display Information을 보낸다.

Process Specification (계속)

Reference No.	1.5.9
Stereotype	Synchronous Function
Name	Display Sleep
Input	Enable / Disable
Output	Display Information(string)
Process Description	Controller로부터 Enable/Disable을 받아 Display Interface에 Display Information을 보낸다.

Reference No.	1.5.10
Stereotype	Synchronous Function
Name	Speak Thank
Input	Trigger, Tick
Output	Speak Information(string)
Process Description	Controller로부터 Trigger를 받아 Speaker Interface에 Speak Information을 보낸다.

Process Specification (계속)

Reference No.	1.5.11
Stereotype	Synchronous Function
Name	Speak Night
Input	Trigger, Tick
Output	Speak Information(string)
Process Description	Controller로부터 Trigger를 받아 Speaker Interface에 Speak Information을 보낸다.

Reference No.	1.5.12
Stereotype	Synchronous Function
Name	Speak Morning
Input	Trigger, Tick
Output	Speak Information(string)
Process Description	Controller로부터 Trigger를 받아 Speaker Interface에 Speak Information을 보낸다.

Process Specification (계속)

Reference No.	1.6
Stereotype	Synchronous Function
Name	Motor Interface
Input	A,B,C Motor Information(int, bool, int)
Output	Motor
Process Description	각 Motor Information을 받아서 정보에 해당하는 Motor 가 동작하게 하는 프로세스이다.
Reference No.	1.7
Stereotype	Asynchronous Function
Name	Display Interface
Input	Display Information(string)
Output	Display
Process Description	Display Information을 받아 그에 맞는 직접적인 정보를 Display Device에 전달하여 화면이 출력되게 한다.
Reference No.	1.8
Stereotype	Asynchronous Function
Name	Speaker Interface
Input	Speak Information(string)
Output	Speak
Process Description	Speak Information을 받아 그에 맞는 직접적인 정보를 Speaker Device에 전달하여 소리를 출력하게 한다.

Data Dictionary

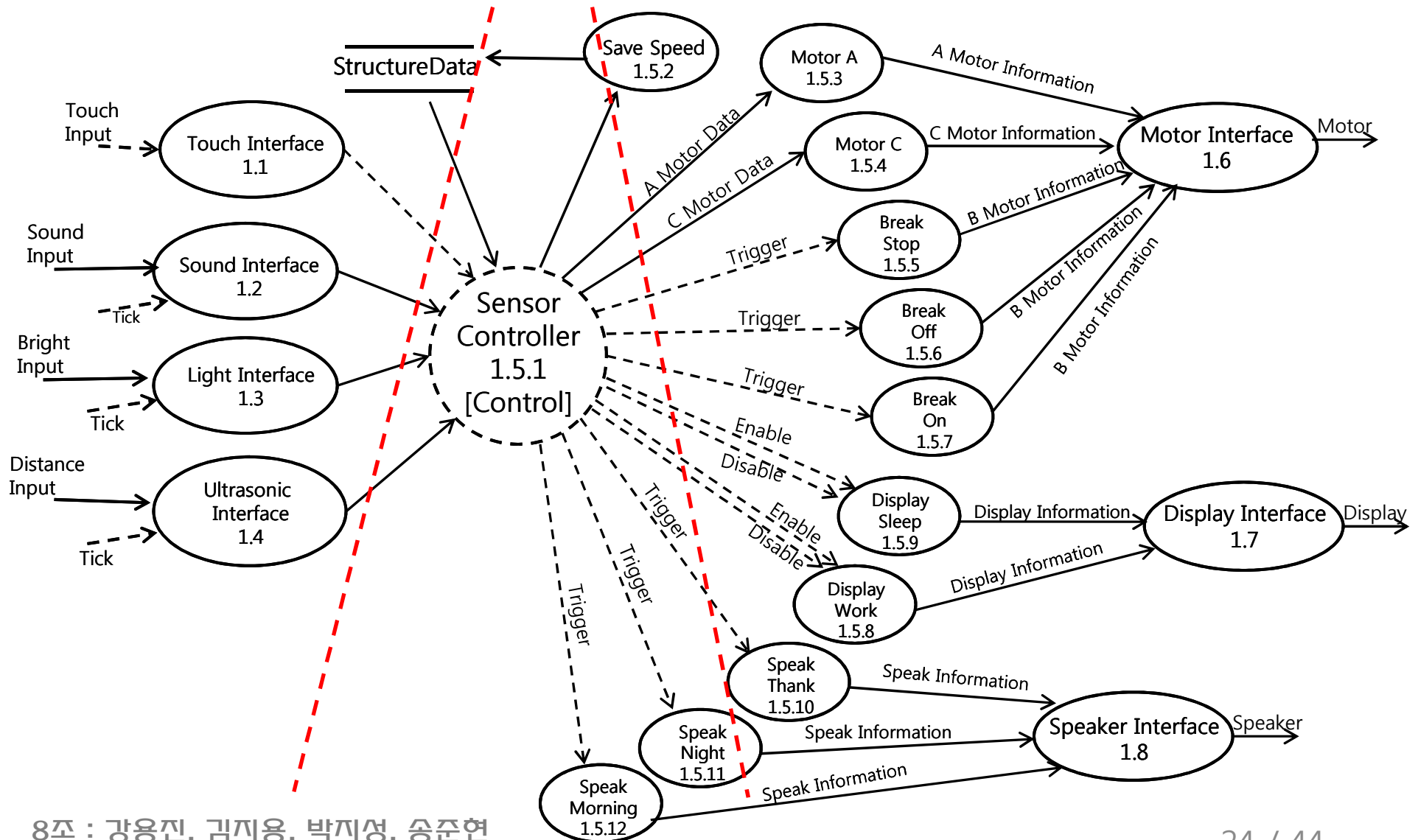
데이터 명	설명
Touch	Touch Sensor로부터 Touch Input이 들어왔음을 나타내는 이벤트이다. T/F로 구성되며 True가 Touch Input이 들어왔을 때이다.
Sound	Sound Sensor로부터 들어오는 Sound Input의 크기
Bright	Light Sensor로부터 들어오는 Bright Input의 크기
Distance	Ultrasonic Sensor를 통해 들어오는 Distance Input 크기
data	Controller에 의해 결정된 Structure Data의 새로운 값들
Structure Data	Touch Count, Dark, Direction Count, Speed, touch_check 다섯가지의 값들을 저장하여 Controller가 로봇의 다음 행동을 결정하는데 쓸 수 있도록 한다.
A Motor Data	A 모터에 대한 방향과 속도를 저장하고 있다.
C Motor Data	C 모터에 대한 방향과 속도를 저장하고 있다.

Data Dictionary (계속)

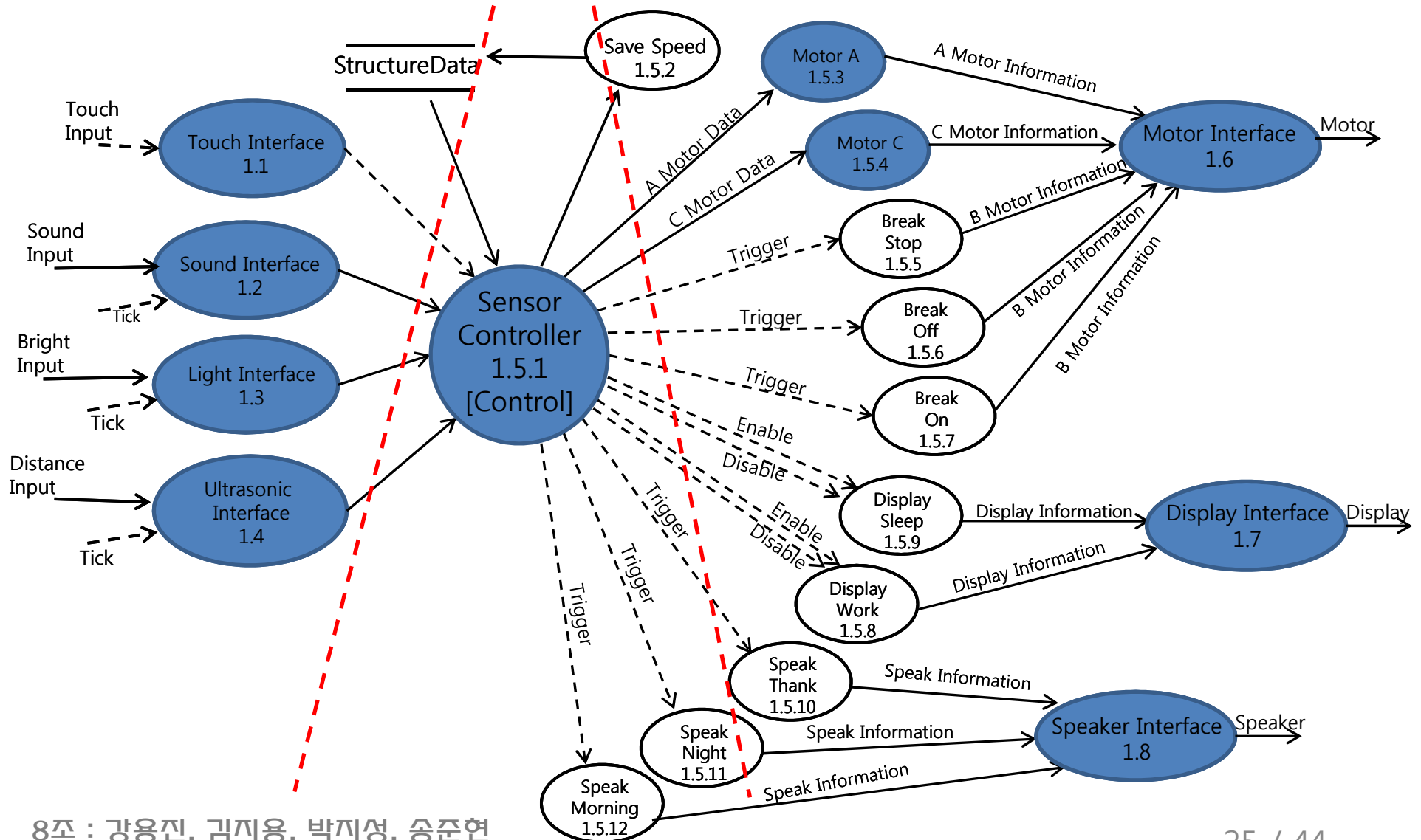
데이터 명	설명
A Motor Information	A모터에 대한 속도와 방향을 포함하여 A모터가 저장된 값에 따라 동작하도록 하는 데이터이다.
B Motor Information	B모터에 대한 속도와 방향을 포함하여 B모터가 저장된 값에 따라 동작하도록 하는 데이터이다.
C Motor Information	C모터에 대한 속도와 방향을 포함하여 C모터가 저장된 값에 따라 동작하도록 하는 데이터이다.
Display Information	로봇의 Display 화면에 실제로 출력할 문자열 데이터를 나타낸다. 소스 코드상에서 해당 문자열을 직접 표현하도록 사용한다.
Speak Information	Speaker가 실제로 낼 소리 데이터를 나타낸다. NXT에 내장되어 있는 소리 DB를 이용하여 소스 코드상에서 사용한다.

Structured Design

Structured Charts – Transform Analysis



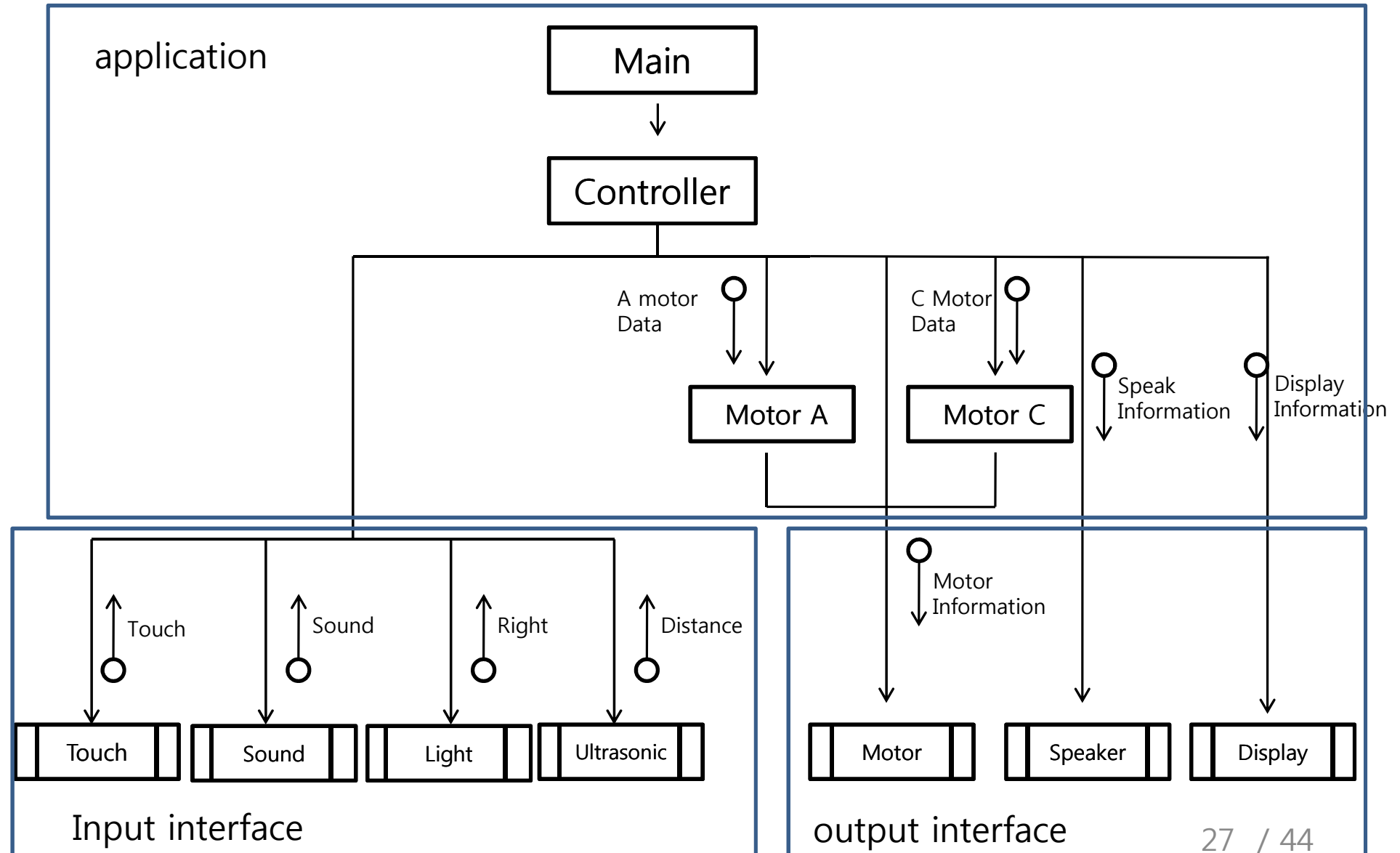
Structured Charts –Task 설정



Task Identification

Task	프로세스	선별기준	모듈
Task 1	Touch interface 1.1	Asynchronous	Touch
Task 2	Sound interface 1.2	Periodic	Sound
Task 3	Light interface 1.3	Periodic	Bright
Task 4	Ultrasonic interface 1.4	Periodic	Ultrasonic
Task 5	Controller 1.5.1 Save data 1.5.2 Break stop 1.5.5 Break off 1.5.6 Break on 1.5.7 Working 1.5.8 Sleeping 1.5.9 Thank you 1.5.10 Night 1.5.11 Good morning 1.5.11	Control 제어응집도 제어응집도 제어응집도 제어응집도 제어응집도 제어응집도 제어응집도 제어응집도 제어응집도	Controller
Task 6	Motor A 1.5.3	Synchronous	Motor A
Task 7	Motor C 1.5.4	Synchronous	Motor C
Task 8	Motor interface 1.6	synchronous	Motor
Task 9	Display interface 1.7	Asynchronous	Display
Task 10	Speaker interface 1.8	Asynchronous	Speaker

Module Structure



Module Definition

모듈 ID	RC_AM01	모듈 명	Main
모듈 개요	Controller 를 실행시킴		
인터페이스	없음 - 프로그램 구동 시 실행		

모듈 ID	RC_AM02	모듈 명	Controller
모듈 개요	Input 값들을 가지고 Motor, Speaker, Display를 제어함		
인터페이스	Start_controller() - 메인 함수에서 실행		

모듈 ID	RC_AM03	모듈 명	Touch
모듈 개요	Touch Sensor로부터 입력을 받아 처리함.		
인터페이스	Get_touch() - touch sensor의 입력을 받아 (bool)touch로 반환하는 함수		

모듈 ID	RC_AM03	모듈 명	Sound
모듈 개요	Sound Sensor로부터 입력을 받아 처리함.		
인터페이스	Get_sound() - Sound Sensor의 입력을 받아 (int)sound로 반환하는 함수		

Module Definition (계속)

모듈 ID	RC_AM05	모듈 명	Light
모듈 개요	Light Sensor로부터 입력을 받아 처리함.		
인터페이스	Get_Light() - Light sensor의 입력을 받아 (int)Light로 반환하는 함수		
모듈 ID	RC_AM06	모듈 명	Ultrasonic
모듈 개요	Ultrasonic Sensor로부터 입력을 받아 처리함.		
인터페이스	Get_distance() - Ultrasonic Sensor의 입력을 받아 (int)distance로 반환하는 함수		
모듈 ID	RC_AM07	모듈 명	Motor A
모듈 개요	현재 저장된 속도에 맞게 모터 A를 설정함		
인터페이스	Set_motor_a(int bool, int speed) - A 모터의 스피드를 받아 저장된 형태로 모터를 동작시키는 함수		
모듈 ID	RC_AM08	모듈 명	Motor B
모듈 개요	현재 저장된 속도에 맞게 모터 C를 설정함		
인터페이스	Set_motor_c(int bool, int speed) - C 모터의 스피드를 받아 저장된 형태로 모터를 동작시키는 함수		

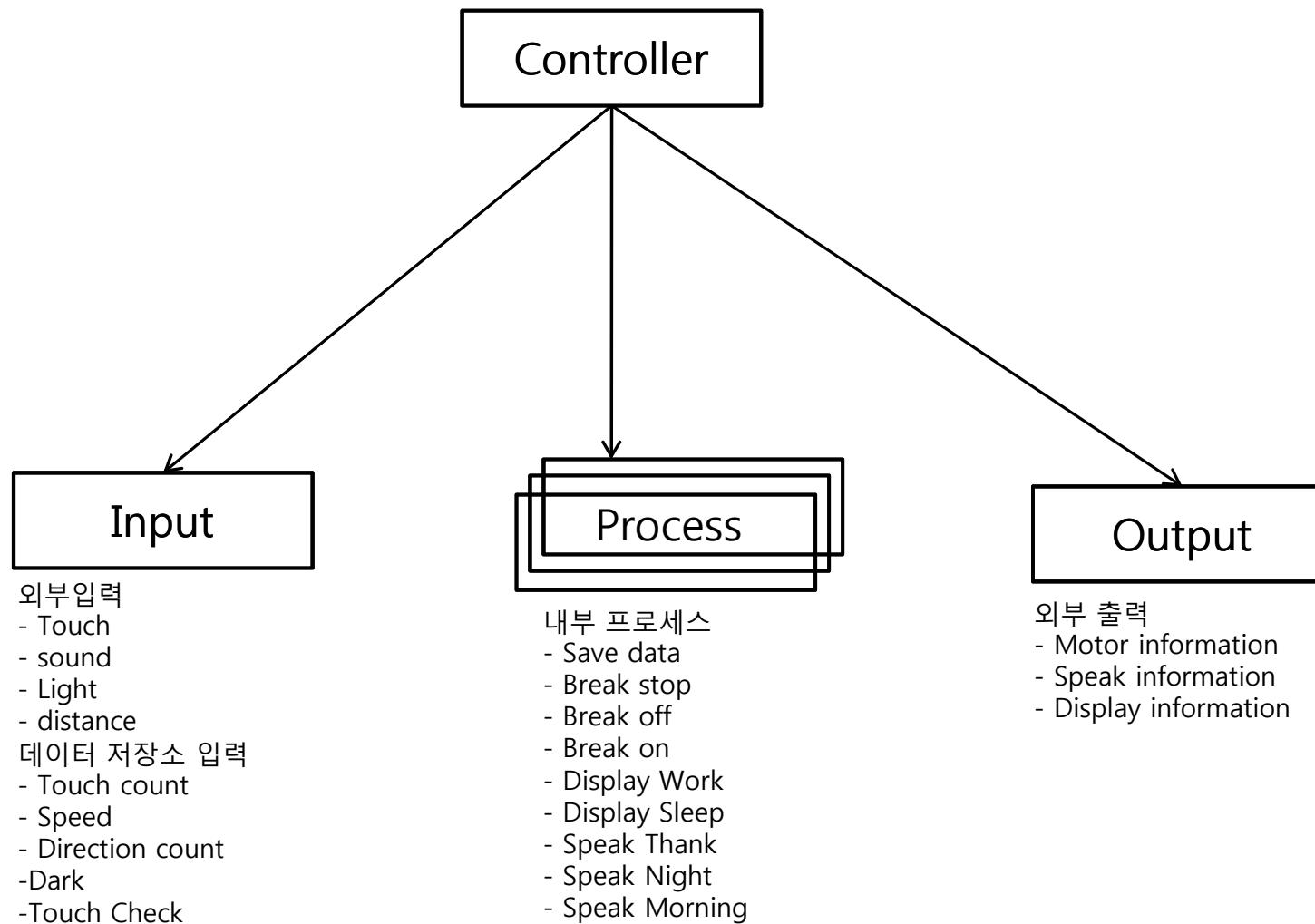
Module Definition (계속)

모듈 ID	RC_AM09	모듈 명	Motor
모듈 개요	현재 저장된 속도와 방향으로 해당 모터를 회전		
인터페이스	Set_motor(int choose, bool direction, int speed) 모터 이름과, 방향(앞과 뒤), 스피드를 인자로 받아 모터를 동작시키는 함수		

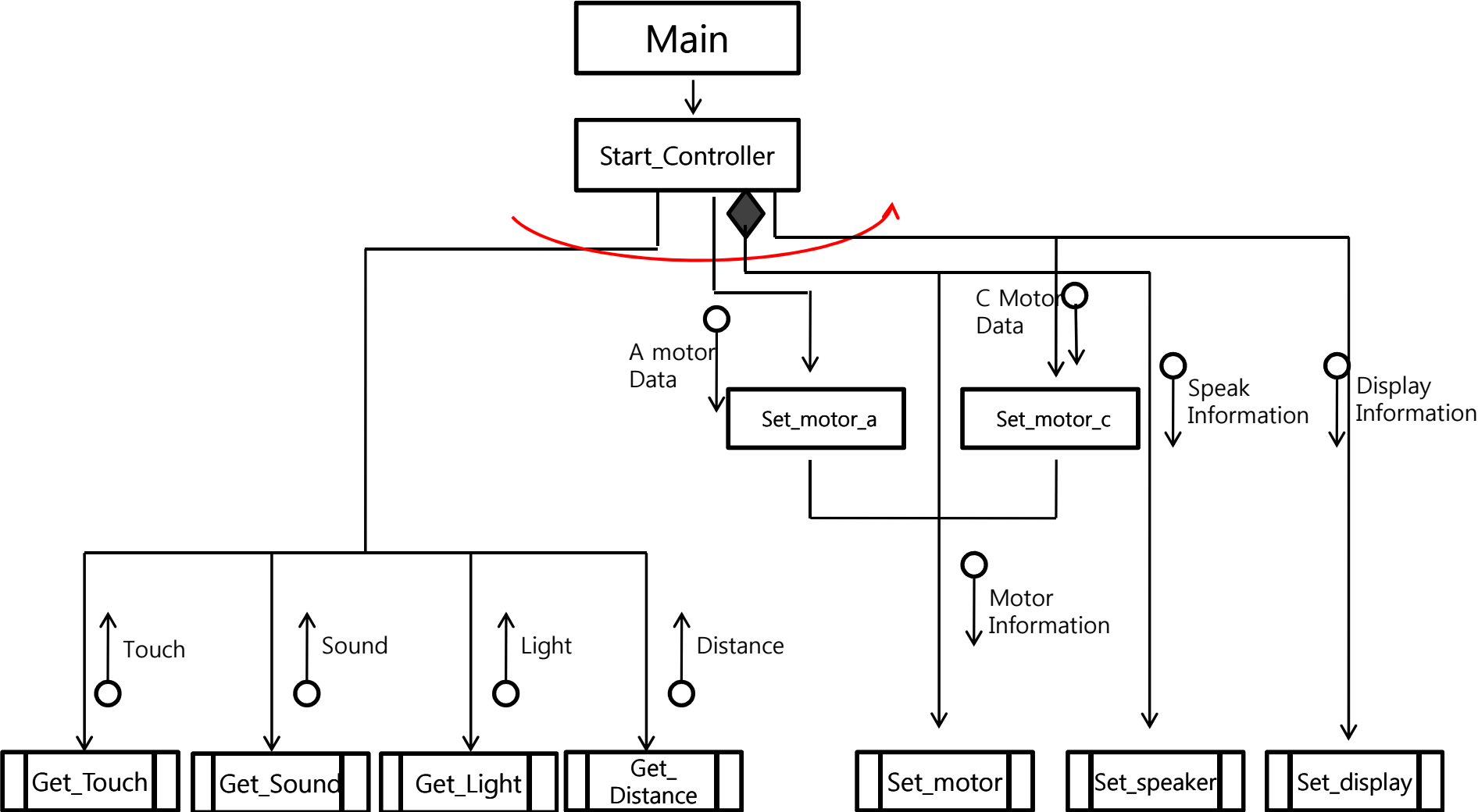
모듈 ID	RC_AM010	모듈 명	Speaker
모듈 개요	소프트웨어에서 처리된 출력 값을 이용해 외부 스피커를 동작할 수 있도록 처리함		
인터페이스	set_speaker(string sp_info) - 출력할 소리 정보를 받아서 speaker를 동작하게 하는 함수		

모듈 ID	RC_AM11	모듈 명	Display
모듈 개요	소프트웨어에서 처리된 출력 값을 이용해 외부 디스플레이를 동작할 수 있도록 처리함		
인터페이스	Set_display(string dis_info) - 화면에 출력할 정보를 받아 display를 동작하게 하는 함수		

Structured Charts – Transform Analysis



Structured Charts

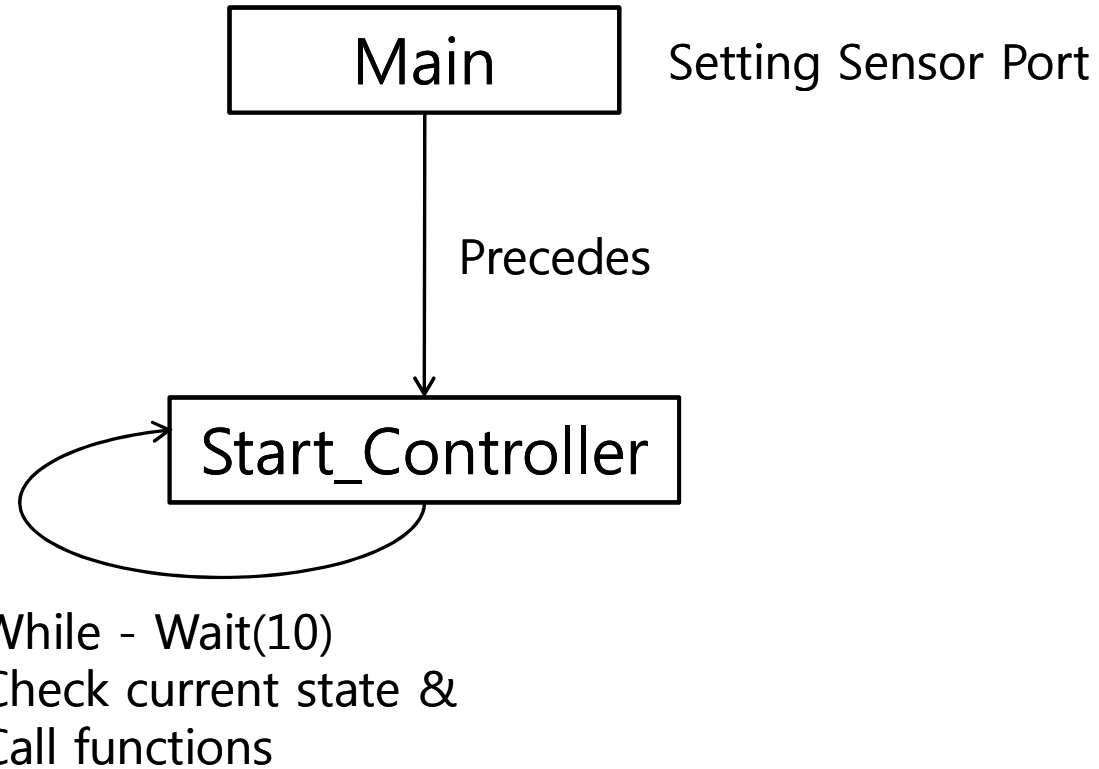


Data Definition

데이터 명	설명	타입
Touch	Touch 센서의 입력 정보를 가진 변수	bool
Sound	Sound 센서의 입력 정보를 가진 변수	Int (0~100)
Light	Light 센서의 입력 정보를 가진 변수	Int (0~100)
Distance	Ultrasonic 센서의 입력 정보를 가진 변수	Int(0~255)
A Motor data	A 모터를 움직일 데이터 변수	Structure (bool direction, int speed) (방향, 세기)
C Motor data	C 모터를 움직일 데이터 변수	Structure (bool direction, int speed) (방향, 세기)
Motor information	Motor Interface에게 보내는 정보 구조	Structure(int choose, bool direction, int speed) (모터 종류, 방향, 세기)
Speak information	Speaker Interface에게 보내는 string 정보	String
Display information	Display Interface에게 보내는 string 정보	String

Implement

Program Flow



Functions

<Input>

- bool Get_Touch()
- int Get_Sound()
- int Get_Light()
- int Get_Distance()

<Output>

- void Set_Motor(int Choose, bool Direction, int Speed)
- void Set_Motor_A(int Bool, int Speed)
- void Set_Motor_C(int Bool, int Speed)
- void Set_Speaker(string Sp_Info)
- void Set_Display(string Dis_Info)

<Control>

- task Start_Controller()

<Main>

- Task main()

Functions – Main

- SetSensorTouch(IN_1);
- SetSensorSound(IN_2);
- SetSensorLight(IN_3);
- SetSensorType(IN_3,IN_TYPE_LIGHT_INACTIVE);
- SetSensorLowspeed(IN_4);

- Precedes(Start_Controller);

Functions - Input

- `bool Get_Touch()`
- `int Get_Sound()`
- `int Get_Light()`
- `int Get_Distance()`

- Each function returns its sensor input
ex) `return SENSOR_2, return SensorUS(IN_4)`

Functions - Output

- Set_Motor_A(int Bool, int Speed)
Call Set_Motor function
- Set_Motor_C(int Bool, int Speed)
Call Set_Motor function
- Set_Motor(int Choose, bool Direction, int Speed)
Switch(Choose)
 - case 0 : if(Direction)
 - OnFwd(OUT_A,Speed)
 - else
 - OnRev(OUT_A,Speed)
 - break
 - case 1 :
 - ⋮

Functions – Output(계속)

- Set_Speaker(string Sp_Info)
 switch(StrLen(Sp_Info))
 case 7 : PlayFileEx("Goodmorning.rso",3,FALSE)
 break

 ⋮
- Set_Display(string Dis_Info)
 ClearScreen()
 TextOut(5,LCD_LINE2,Dis_Info)

Functions - Control

- task Start_Controller()

Make local variable

while(true)

Idle state until touch is pushed

while(true)

Update sensor variable

if – Check 'Forward' state [T/F → F]

if – Change state from 'Stop' or 'Brake' to 'Forward' by Dark [S/B → F]

if – Increase / decrease speed by touch conditions [F → F]

if – Change direction by Direction_Count [F/T → T]

Turn left / Turn 180 / Trun right

if – Check noise [F → F]

if – Operate brake by Dark condition [F → B / B → S]

Wait(10)

Sort of state

F : Forward

S : Stop

B : Break

(T : Turn)

TL : Turn Left

TR : Turn Right

T180 : Turn 180

Special Feature

- 속도가 100 이상이 될 경우 10 으로 낮춘다.

```
while(Speed > 10)
{
    Speed - - ;
    Set_Motor_A(TRUE,Speed);
    Set_Motor_C(TRUE,Speed);
    if(Get_Sound()>=70)
    {   Sound = Get_Sound();   }
    Wait(5);
}
Wait(40);
```

- 모터 속도를 100 에서 10 으로 바로 바꾸면 모터에 무리가 올 수 있다.
 - 속도를 1씩 낮춘다.
 - 속도 100 에서 10까지 1씩 낮추면 While 문에서 90회 반복된다.
 - $90 * 5 + 40 = 490$ (기본 주기 10초)

- 터치를 누르면 0.5초마다 속도가 10씩 증/감 한다.

```
repeat(49)
{
    if(Get_Sound()>=70)
    {   Sound = Get_Sound();   }
    Wait(10);
}
```

- 0.5초 사이에 소음 발생시 처리할 수 없다.
 - Wait(500)이 아닌 Repeat을 활용한다.
 - $49 * 10 = 490$ (기본 주기 10초)

Happenings

- 코딩 보다는 NXC라는 언어의 문법을 몰라서 사용법을 익히는것이 더 어려웠다.
- TRUE / FALSE 를 대문자로 써야 한다.
- 속도를 줄일 때 줄여진 속도 값과 True를 보내야 하는데 False값과 줄여진 값을 사용해서 처음에 마인드스톰이 뒤로 가는 현상이 나타났다.
- Stop상태로 진입 후 계속 어두운 경우 Stop상태로 재진입 하여 밝아지기 전까지 계속 'Night' 소리가 났다.
- 컨트롤러가 하나이기 때문에 한 모터에 동시에 2개의 커맨드가 나가지 않는다. → Mutex 사용하지 않음

Afternote

- SA 단계에서 우리가 흔히 쓰던 변수(저장매개)들을 쓸수 없다는 가정하에 분석을 하다보니 우리가 쓰던 변수들의 힘이 얼마나 강한지 알수 있게 되었다.
- SA 단계에서 실선, 점선, Trigger, etc.. 정확한 개념을 잡지 못해서 초반에 매우 어려웠다.
- 리얼 시스템 프로그램을 구현 하는거라 시스템이 돌아가는 속도(클럭)에 처음으로 고려해보면서 구현을 하다보니 이론이 어떻게 적용되는지 알수있게 되어 재미있었다.
- 모르는 부분이 많아서 교수님을 자주 찾아갔는데, 이른 아침, 점심시간, 늦은 저녁 등 귀찮게 해드린거 같아 죄송하고, 어느 때 찾아가도 친절하게 설명을 잘 해주셔서 감사드린다.

Q & A