

# **Auto Driving Robot**

## **Structured Analysis & Structured Design**

**200611482 송창근**  
**200611489 오정환**  
**200611477 석종수**

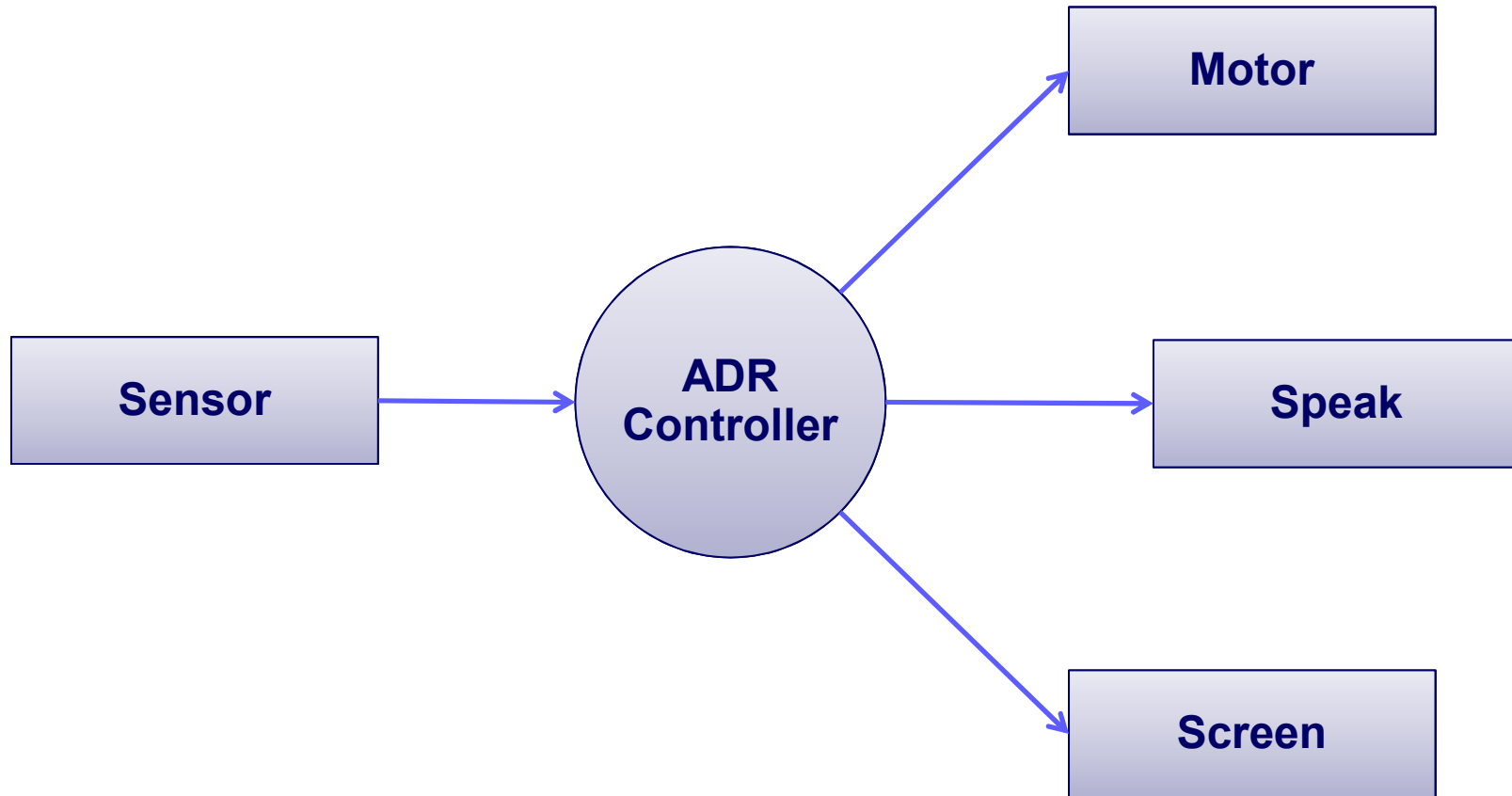
# Structured Analysis

# Statement of Purpose

## Auto Driving Robot ( ADR )

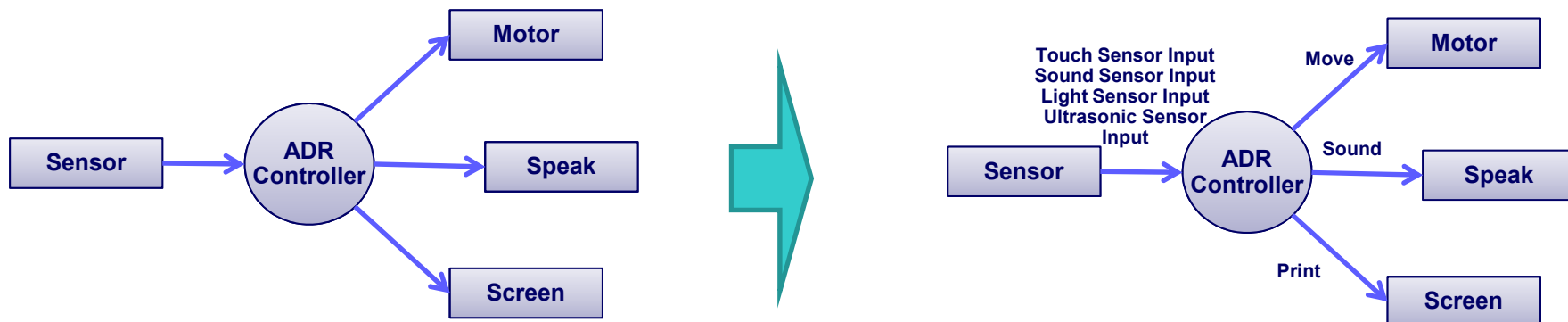
- By default, the robot moves forward.
- By default, the output is “working” on the screen While robot is working.
- Through the front of the sensor the robot is recognize obstacles and recognize obstacles, the robot can change the direction.
- After switching the direction the robot is moves forward.
- Through the TouchSensor, Adjust the speed of the robot.
- The sensor at the top of the robot senses the light, if the light in the dark Braking devices are operationg.
- According to the sound of clap, The robot make sound “thank you”

# System Context Diagram

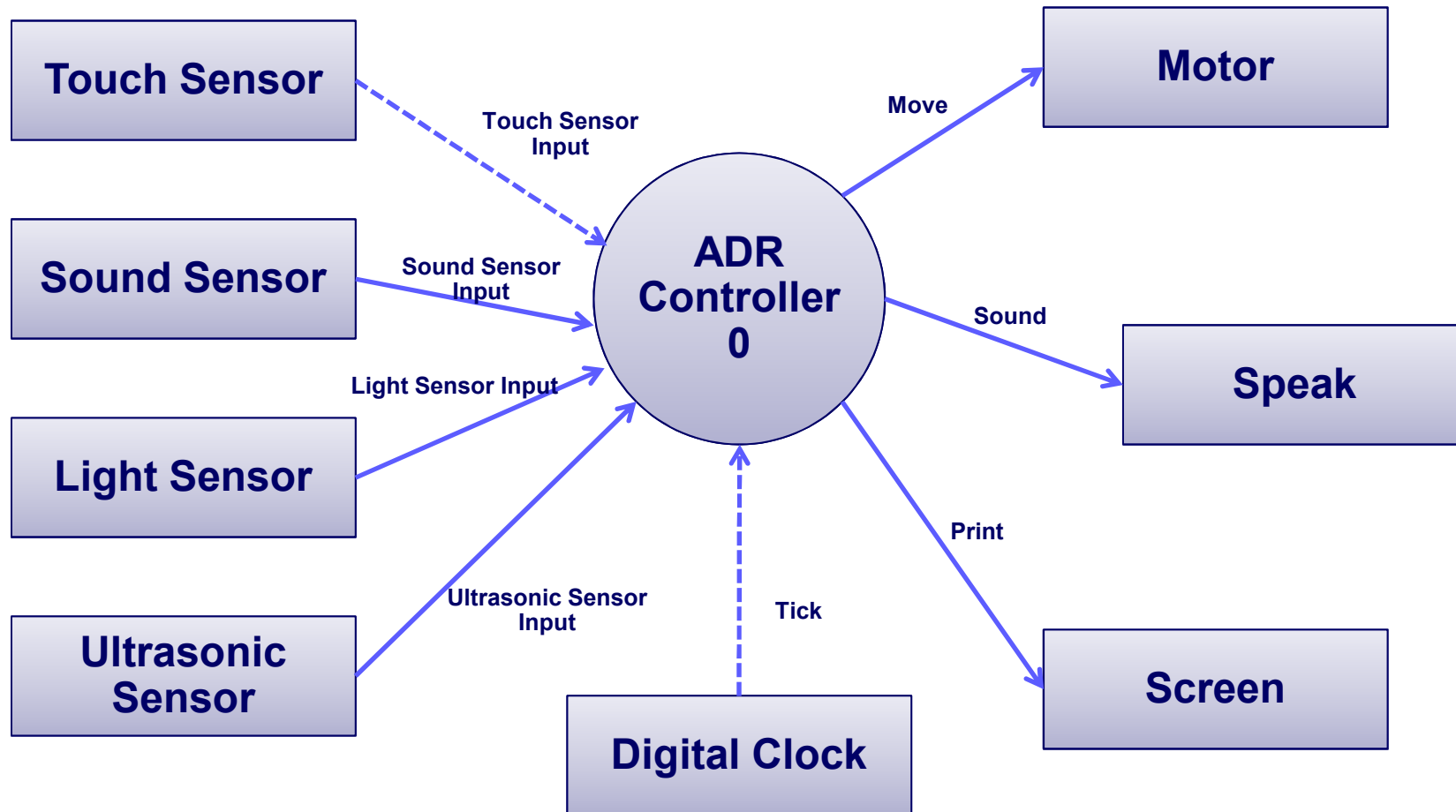


# Event List

Input/ Output Event	Description
Touch Sensor Input	Represent information whether user press the touch sensor button or not.
Sound Sensor Input	Represent information of Sound that comes through Sound Sensor.
Light Sensor Input	Represent information of the brightness of the light coming through the Light Sensor.
Ultrasonic Sensor Input	Represent information of the distance between obstacle that comes through Ultrasonic Sensor.
Move	Information about the movement of the motor.
Sound	The information is represented about Sound output
Print	The information displayed on the screen



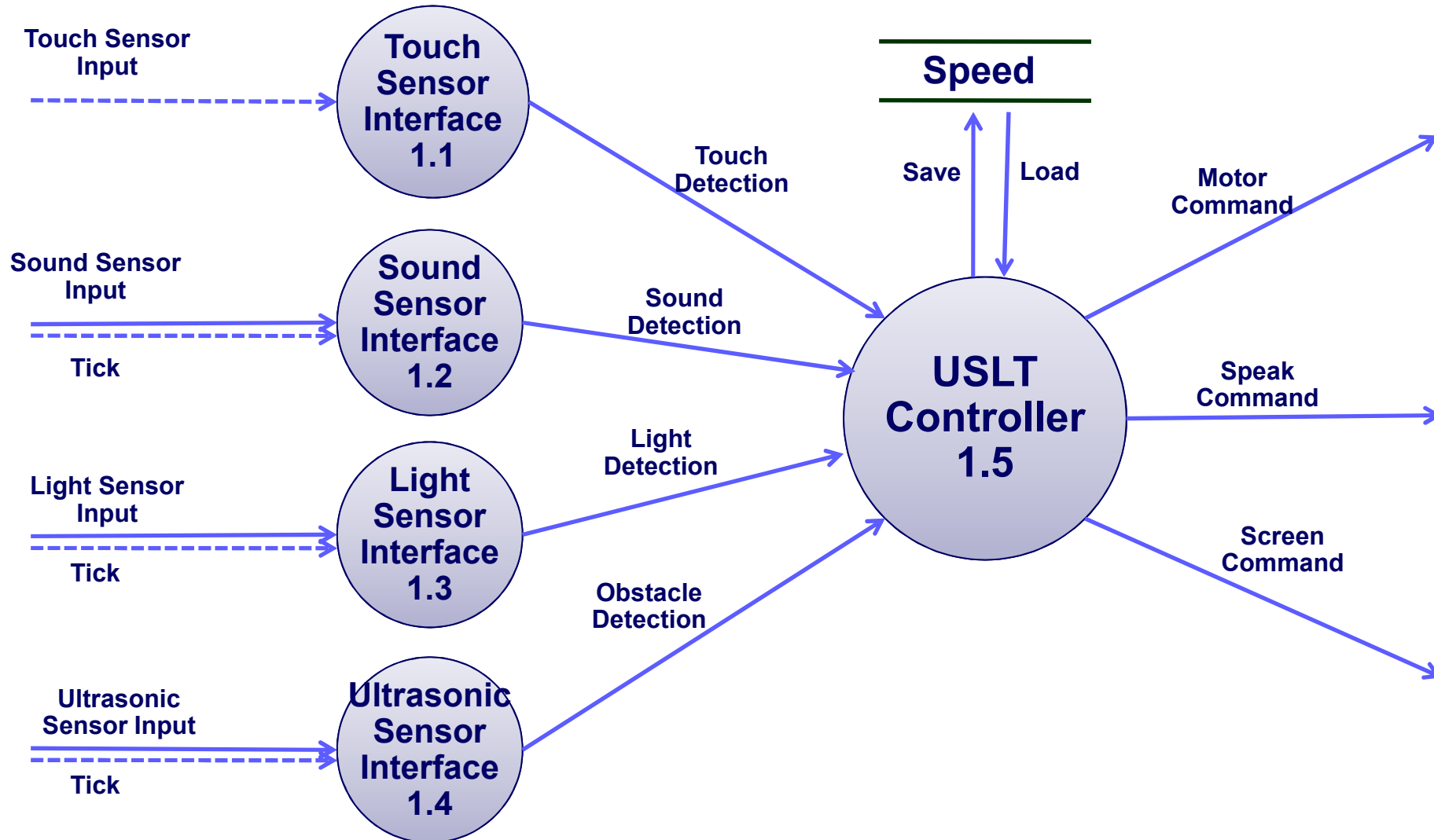
# DFD Level 0



# DFD Level 0

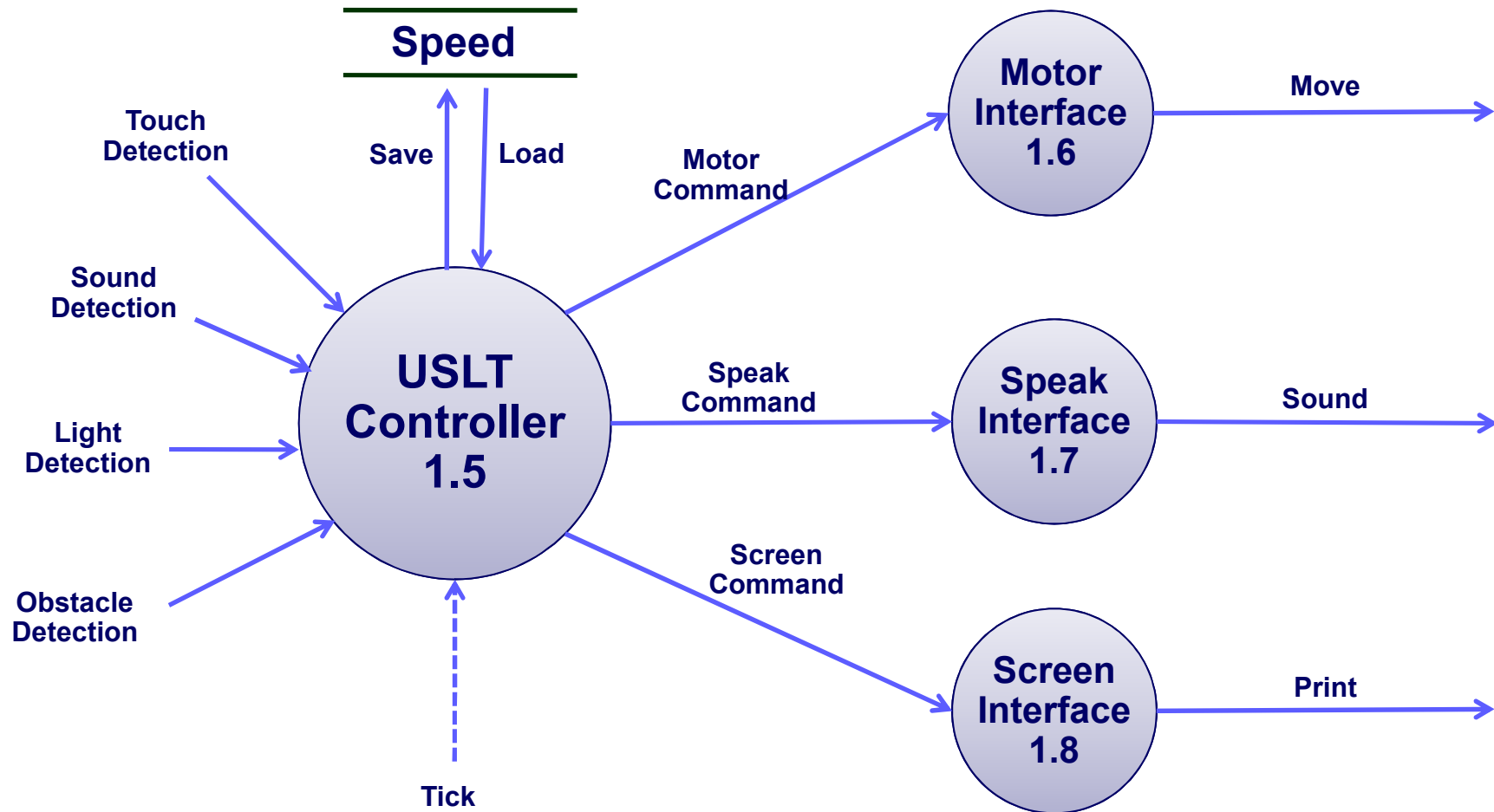
Input/ Output Event	Description	Format / Type
Touch Sensor Input	Represent information whether user press the touch sensor button or not.	True, False / Asynchronous
Sound Sensor Input (S.I)	Represent information of Sound that comes through Sound Sensor. Has an integer value between 0 to 100.	Integer / Periodic
Light Sensor Input (L.I)	Represent information of the brightness of the light coming through the Light Sensor. Has an integer value between 0 to 100.	Integer / Periodic
Ultrasonic Sensor Input (U.I)	Represent information of the distance between obstacle that comes through Ultrasonic Sensor. Has an integer value between 0 to 255.	Integer / Periodic
Move	Information about the movement of the motor.	Go Straight, Turn Left, Turn Right, Turn Back, Stop, Break
Sound	The information is represented about Sound output	Good morning, Night, Thank You
Print	The information displayed on the screen	Working, Sleeping

# DFD Level 1





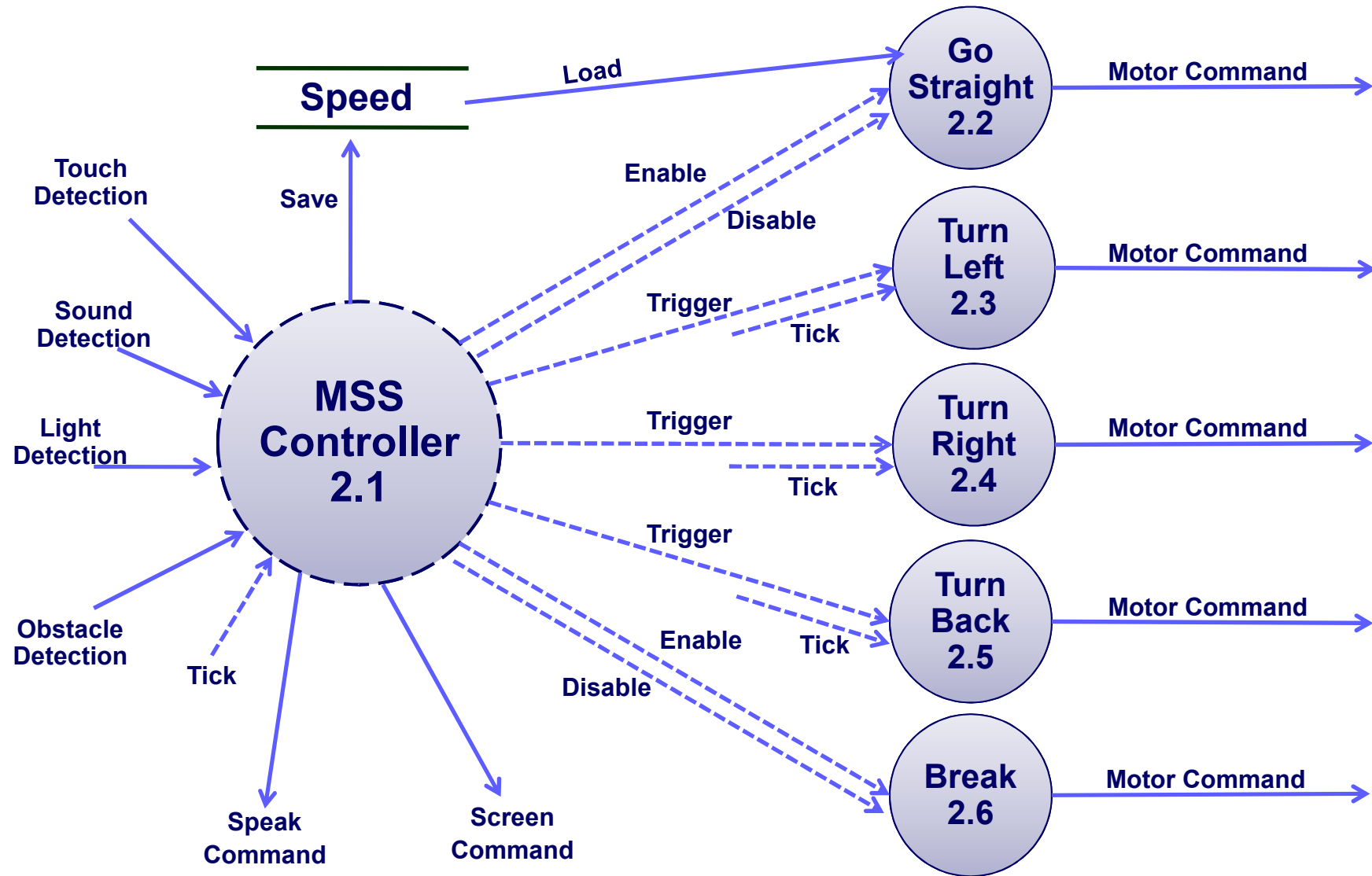
# DFD Level 1



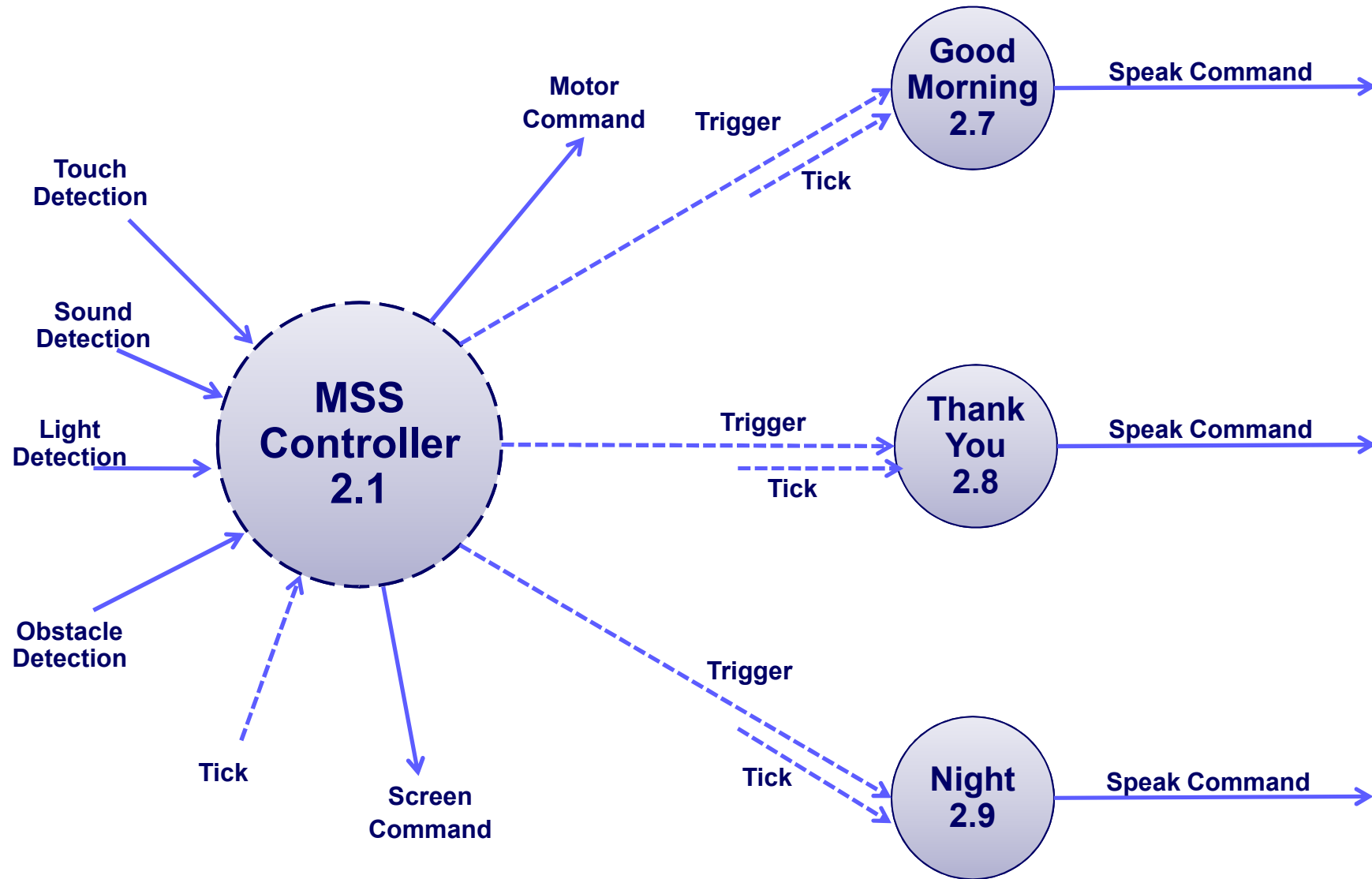
# DFD Level 1

Input/ Output Event	Description
Touch Detection	The information that whether Touch Input is come from Touch Sensor or not. Consist of True and False. If it is true it means that Touch Input is come in
Light Detection	The information of the size of the Light Input that come from Light Sensor.
Sound Detection	The information of the size of Sound Input that come from Light Sensor.
Obstacle Detection	The information of the size of Distance Input that come from Ultrasonic Sensor.
Motor Command	The information about movement that USLT Controller send to Motor Interface.
Speak Command	The information about Sound Output that USLT Controller send to Motor Interface.
Screen Command	The information about Screen Output that USLT Controller send to Motor Interface.
Speed	Speed before Stop is saved or loaded. Has an integer value between 0 to 100.
Save	Speed before Stop is saved .
Load	Speed before Stop is loaded.

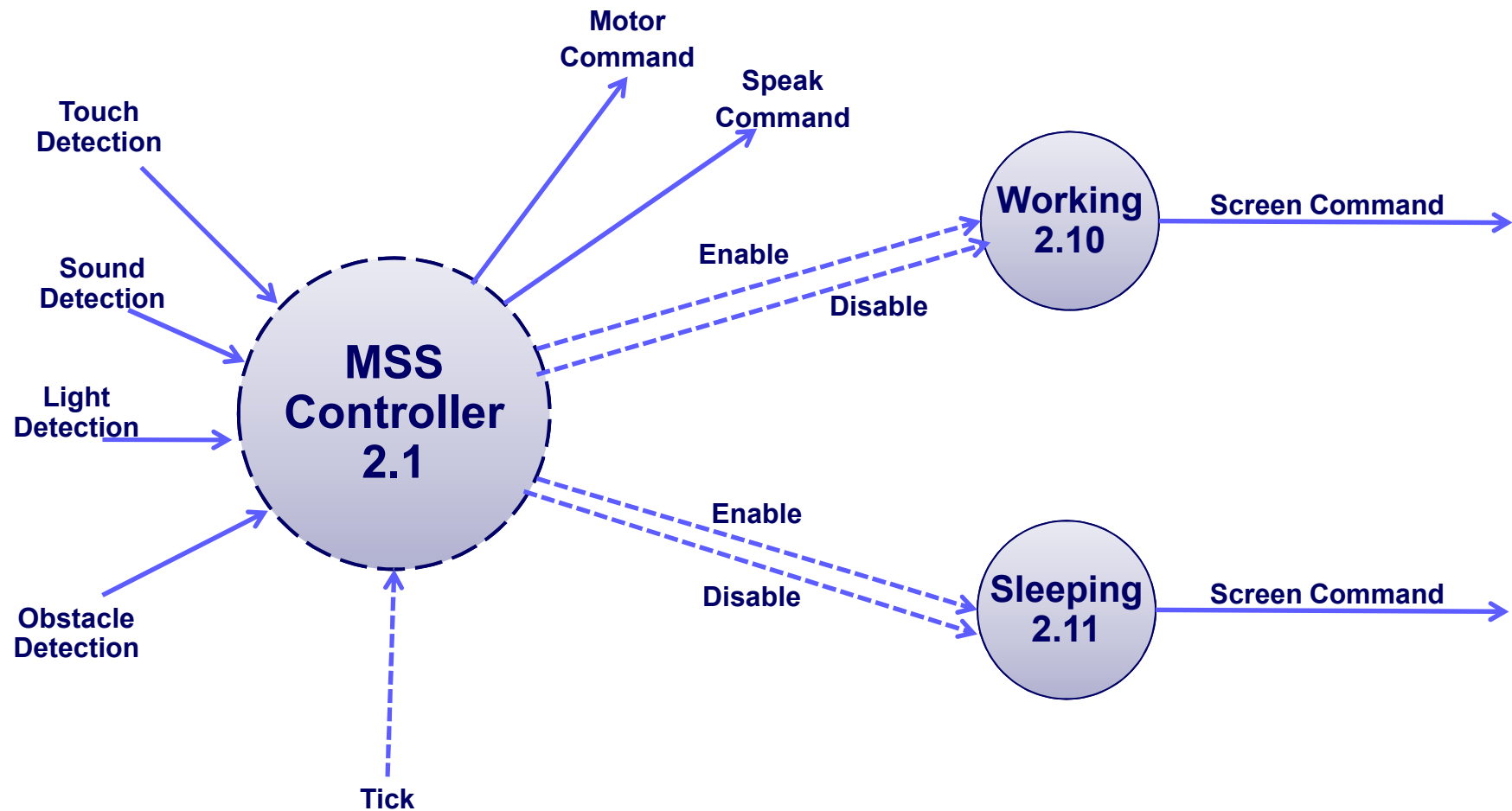
# DFD Level 2



# DFD Level 2

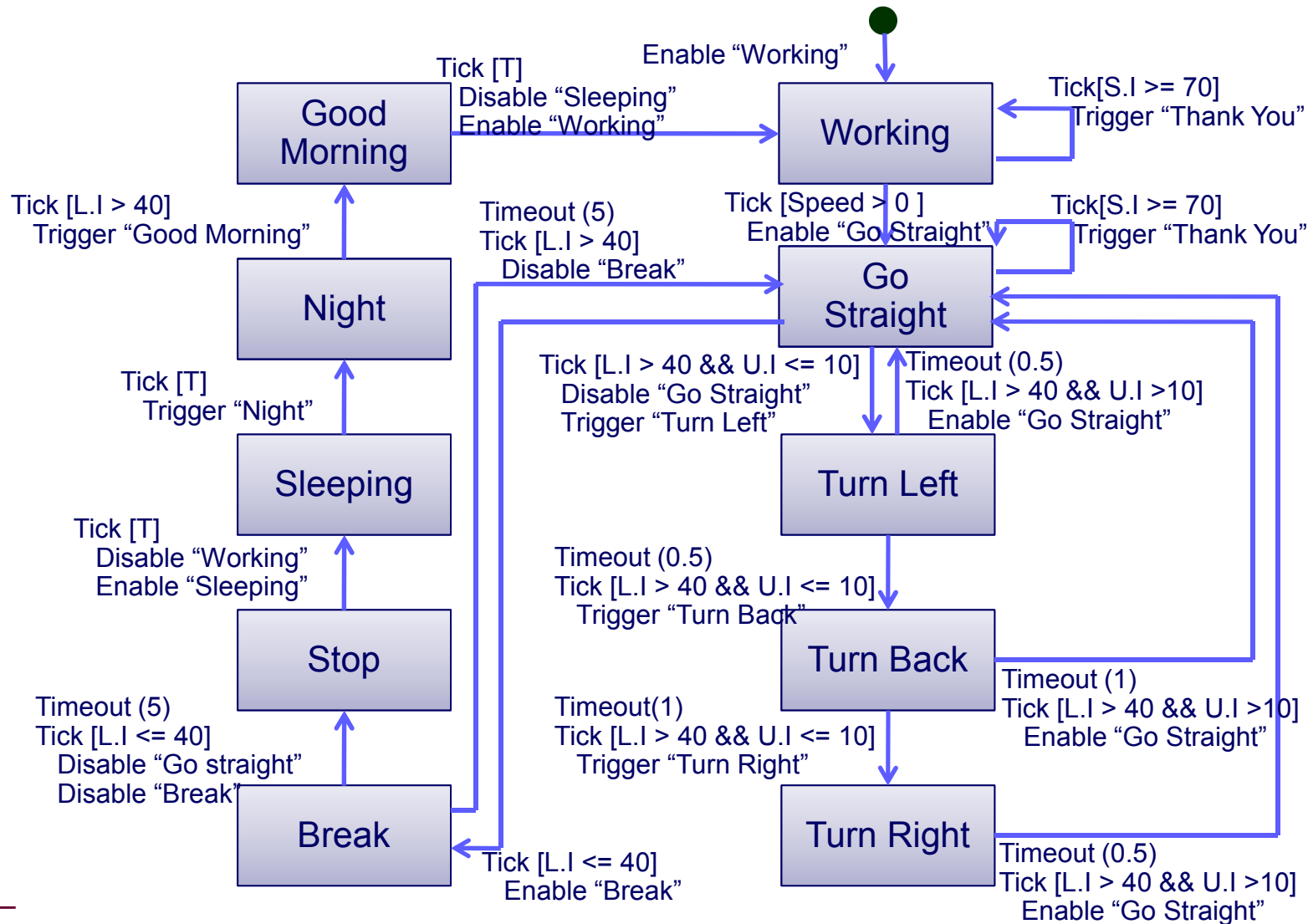


# DFD Level 2



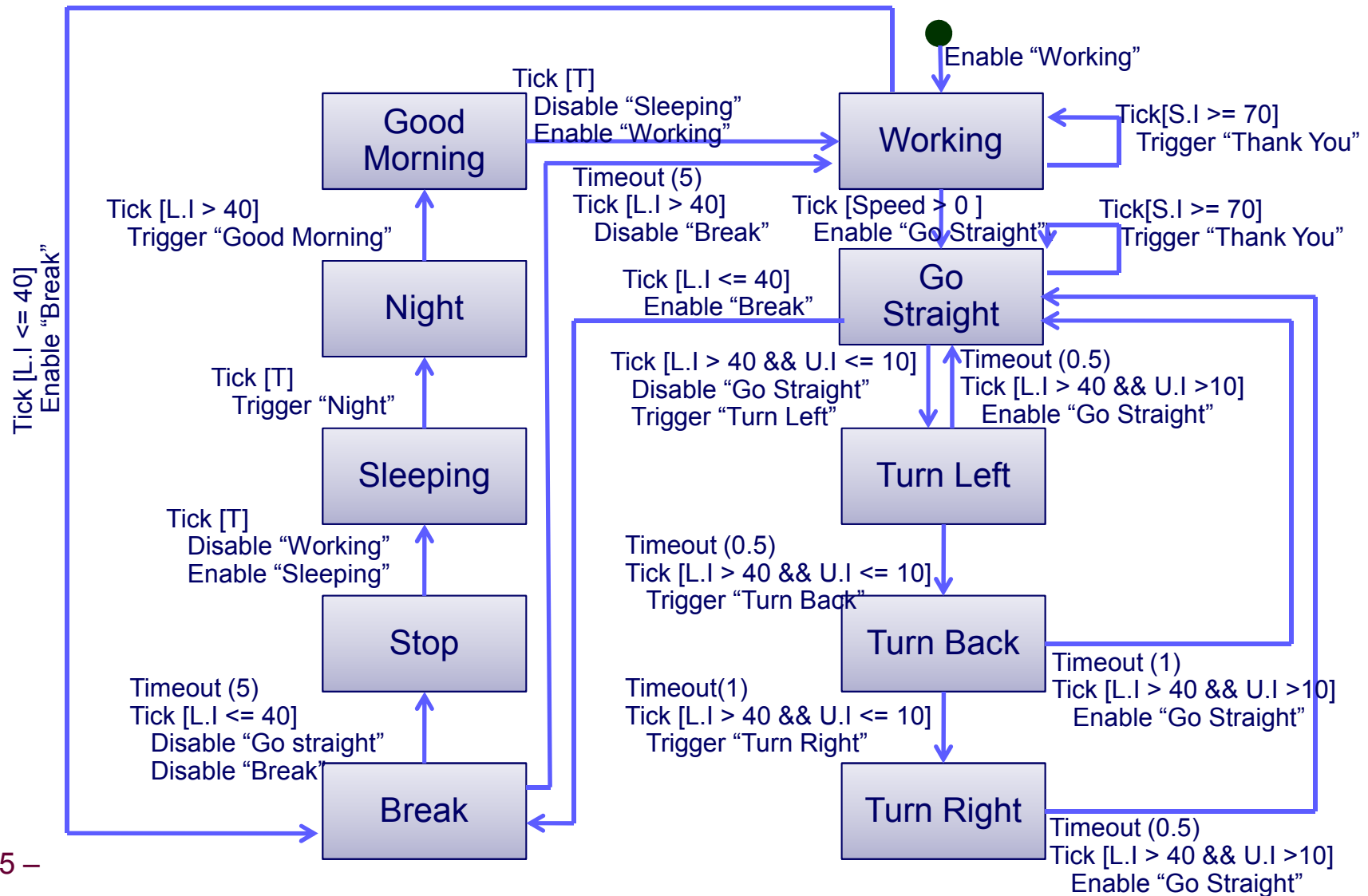
# DFD Level 3 – Before

## State Transition Diagram for 2.1

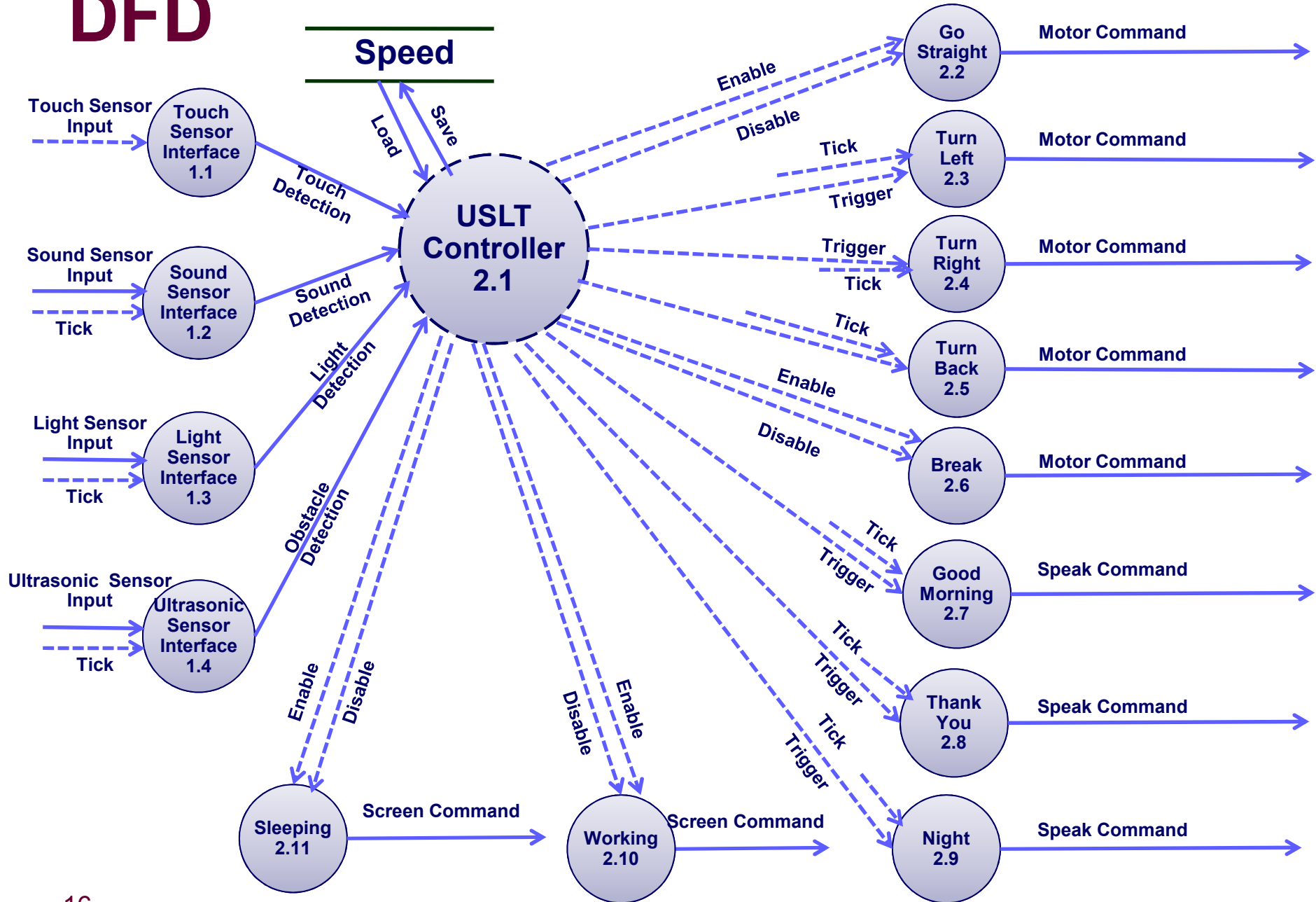


# DFD Level 3 – After

## State Transition Diagram for 2.1



# DFD





# Process Specification

Reference No.	1.1
Name	Touch Sensor Interface
Input	Touch Sensor Input
Output	Touch Detection(bool)
Process Description	Analog Type Touch Sensor Input comes in the form of Digital True / False as a change in the Output Touch Detection allows you to send.

Reference No.	1.2
Name	Sound Sensor Interface
Input	Sound Sensor Input, Tick
Output	Sound Detection(integer)
Process Description	Analog Sound Sensor Input of the type that comes in Digital format changed to Integer between 0-100 and Sound Detection Output sent to give.

# Process Specification

Reference No.	1.3
Name	Light Sensor Interface
Input	Light Sensor Input, Tick
Output	Light Detection(integer)
Process Description	Analog Light Sensor Input of the type that comes in Digital format changed to Integer between 0-100 Light Detection to send the Output to give.

Reference No.	1.4
Name	Ultrasonic Sensor Interface
Input	Ultrasonic Sensor Input, Tick
Output	Obstacle Detection(integer)
Process Description	Analog Ultrasonic Sensor Input of the type that comes in Digital format changed to Integer between 0-100 and send Output gives the Obstacle Detection.

# Process Specification

Reference No.	1.5
Name	USLT Controller
Input	Touch Detection(bool), Sound Detection(integer) Light Detection(integer), Obstacle Detection(integer) Speed(Data structure / Load)
Output	Motor Command, Speak Command, Screen Command Speed(Data structure / Save)
Process Description	Touch · Sound · Light · Obstacle Detection for the Input Command to export each of the received and speed save/load.

Reference No.	1.6
Name	Motor Interface
Input	Motor Command
Output	Move
Process Description	Input into a Motor Command received Motor allows you to adjust the movement.

# Process Specification

Reference No.	1.7
Name	Speak Interface
Input	Speak Command
Output	Sound
Process Description	Receive a sound command and allows the sound output.

Reference No.	1.8
Name	Screen Interface
Input	Screen Command
Output	Print
Process Description	Take on the Input Screen Command and screen out adjust.

# Process Specification

Reference No.	2.1
Name	MSS Controller
Input	Touch Detection(bool), Sound Detection(integer) Light Detection(integer), Obstacle Detection(integer), Tick
Output	Trigger, Enable, Disable
Process Description	Input from the current state of the action is determined, Output to perform actions with the event generates.

Reference No.	2.2
Name	Go Straight
Input	Enable, Disable
Output	Motor Command
Process Description	Enable event occurs is done, sending Motor Command data so Motor A, C can work.

# Process Specification

Reference No.	2.3
Name	Turn Left
Input	Trigger, Tick
Output	Motor Command
Process Description	Trigger event occurs is done, sending Motor Command data so 50% of the speed of 0.5 seconds Motor A allow work forward, 100% of the speed of 0.5 second Motor C allow work back.

Reference No.	2.4
Name	Turn Right
Input	Trigger, Tick
Output	Motor Command
Process Description	Trigger event occurs is done, sending Motor Command data so 100% of the speed of 0.5 seconds Motor A allow work back, 50% of the speed of 0.5 second Motor C allow work forward.

# Process Specification

Reference No.	2.5
Name	Turn Back
Input	Trigger, Tick
Output	Motor Command
Process Description	Trigger event occurs is done, sending Motor Command data so 50% of the speed of 1 seconds Motor A allow work forward, 50% of the speed of 1 second Motor C allow work back.

Reference No.	2.6
Name	Break
Input	Enable, Disable
Output	Motor Command
Process Description	Enable event occurs is done, sending Motor Command data so Motor B can work.

# Process Specification

Reference No.	2.7
Name	Good Morning
Input	Trigger, Tick
Output	Speak Command
Process Description	Trigger event occurs is done, sending Speak Command data and “Good Morning” sound out.

Reference No.	2.8
Name	Thank You
Input	Trigger, Tick
Output	Speak Command
Process Description	Trigger event occurs is done, sending Speak Command data and “Thank You” sound out.



# Process Specification

Reference No.	2.9
Name	Night
Input	Trigger, Tick
Output	Speak Command
Process Description	Trigger event occurs is done, sending Speak Command data and "Night" sound out.

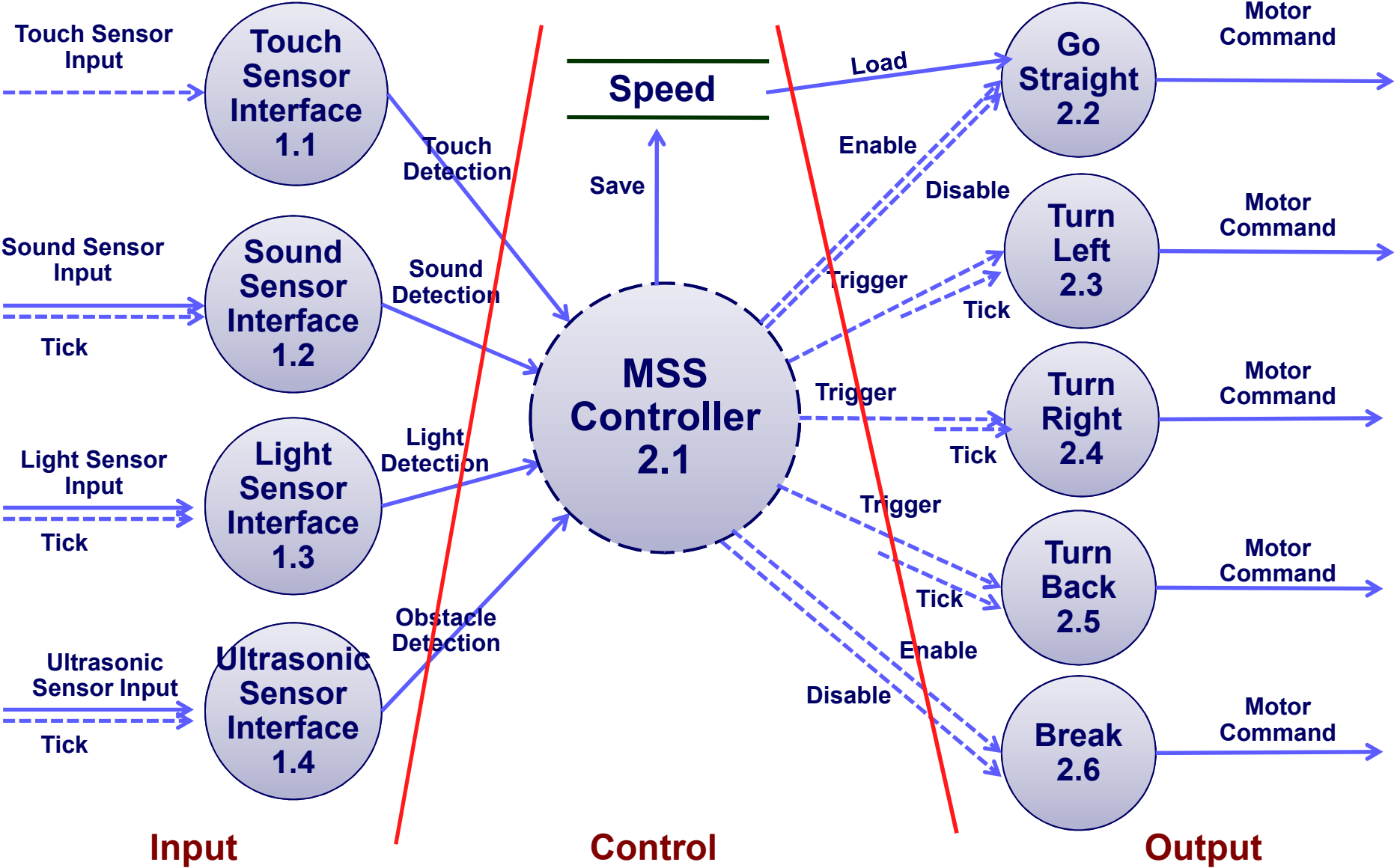
Reference No.	2.10
Name	Working
Input	Enable, Disable
Output	Screen Command
Process Description	Enable event occurs is done, sending Screen Command data and "Working" allows you to print text on the screen.

# Process Specification

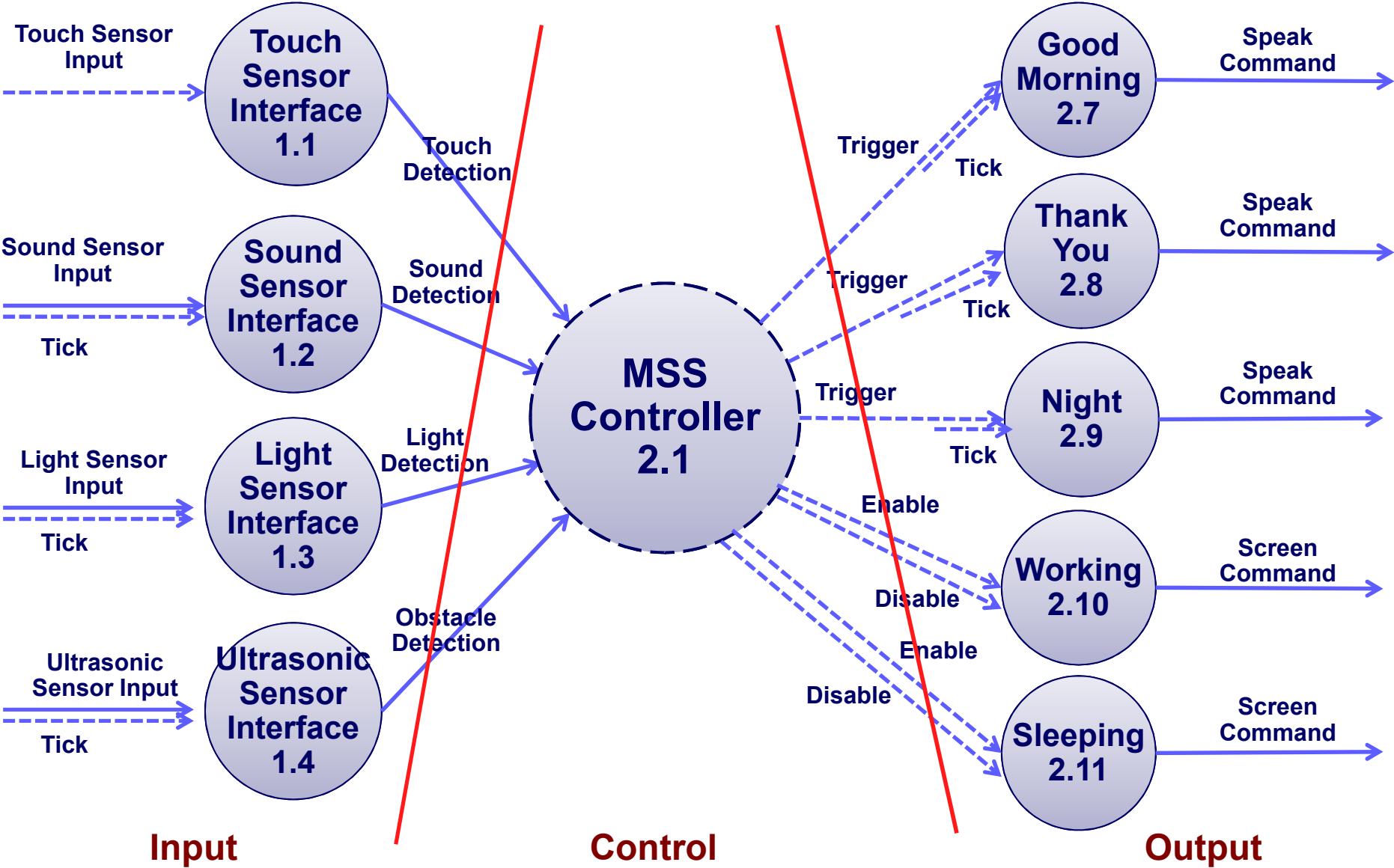
Reference No.	2.11
Name	Sleeping
Input	Enable, Disable
Output	Screen Command
Process Description	Enable event occurs is done, sending Screen Command data and "Sleeping" allows you to print text on the screen

# Structured Design

# Transform Analysis



# Transform Analysis



# Module Definition

Process	Selection criteria	Module
Touch Sensor Interface 1.1	Asynchronous	Touch
Sound Sensor Interface 1.2	Periodic	Sound
Light Sensor Interface 1.3	Periodic	Light
Ultrasonic Sensor Interface 1.4	Periodic	Ultrasonic
MSS Controller 2.1	Control	Control
Go Straight 2.2	Asynchronous	Go_Straight
Turn Left 2.3	Asynchronous	Turn_Left
Turn Right 2.4	Asynchronous	Turn_Right
Turn Back 2.5	Asynchronous	Turn_Back
Break 2.6	Asynchronous	Break

# Module Definition

Process	Selection criteria	Module
Good Morning 2.7	Asynchronous	Good_Morning
Thank You 2.8	Asynchronous	Thank_You
Night 2.9	Asynchronous	Night
Working 2.10	Asynchronous	Working
Sleeping 2.11	Asynchronous	Sleeping

# Module Definition

Module ID	ADR – IM01	Module Name	Touch
Module Outline	Touch Sensor detects the input sent to an bool value gives reversed for control.		
Interface	bool i_touch() Function that returns bool value for Touch Sensor		

Module ID	ADR – IM02	Module Name	Sound
Module Outline	Sound Sensor detects the input sent to an integer value gives reversed for control.		
Interface	int i_sound() Function that returns integer value for Sound Sensor		

Module ID	ADR – IM03	Module Name	Light
Module Outline	Light Sensor detects the input sent to an integer value gives reversed for control.		
Interface	int i_light() Function that returns integer value for Light Sensor		



# Module Definition

Module ID	ADR – IM04	Module Name	Ultrasonic
Module Outline	Ultrasonic Sensor detects the input sent to an Integer value gives reversed for control.		
Interface	int i_ultrasonic() Function that returns integer value for Ultrasonic Sensor		
Module ID	ADR – CM	Module Name	Control
Module Outline	<p>The data for each sensor has to store the value of each variable. Speed will be saved as the independent value. Data from each sensor value processing operations, and a decent feature allows you to operate.</p> <p>bool d_touch – Touch Input Value  int d_sound – Sound Input Value  int d_light – Light Input Value  int d_ultrasonic – Ultrasonic Input Value  int d_speed – Speed Value (Speed to save.)  bool count – When “count” is the “True“, the speed is increased by 10. When “count” the “False“, the speed is decreased by 10. If The “count” value does not change when the next sensing, The value of 0.5 second increments continuously added or reduced.</p>		
- 3 Interface	None – Programs running on startup		

# Module Definition

Module ID	ADR – OM01	Module Name	Go_Straight
Module Outline	Take command from control and Motor B, C that allows you to move.		
Interface	<pre>int o_go(d_speed)</pre> Function returns a value Interger d_speed and Motor factors to take into d_speed values, behavior		
Module ID	ADR – OM02	Module Name	Turn_Left
Module Outline	Take command from control and Motor A at a rate of 50% of the next 0.5 seconds, Motor C speed of 0.5 seconds behind a 100% working.		
Interface	<pre>void o_left()</pre> Function to the operation for turn left		
Module ID	ADR – OM03	Module Name	Turn_Right
Module Outline	Take command from control and Motor C at a rate of 50% of the next 0.5 seconds, Motor A speed of 0.5 seconds behind a 100% working.		
Interface	<pre>void o_right()</pre> Function to the operation for turn right		

# Module Definition

Module ID	ADR – OM04	Module Name	Turn_Back
Module Outline	Take command from control and at a rate of 50% of the Motor A Forward 1 seconds, 50% of Motor C speed is 1 seconds to back to work.		
Interface	void o_back() Function to the operation for turn 180°		

Module ID	ADR – OM05	Module Name	Break
Module Outline	Take command from control and Motor B will have to move.		
Interface	void o_break() Function to the operation for “Break”		

Module ID	ADR – OM11	Module Name	Good_Morning
Module Outline	Take command from control and the function outputs the sound “Good Morning”		
Interface	void o_morning() Function outputs the sound “Good Morning”		

# Module Definition

Module ID	ADR – OM12	Module Name	Thank_You
Module Outline	Take command from control and the function outputs the sound “Thank you”		
Interface	void o_thanks() Function outputs the sound “Thank you”		

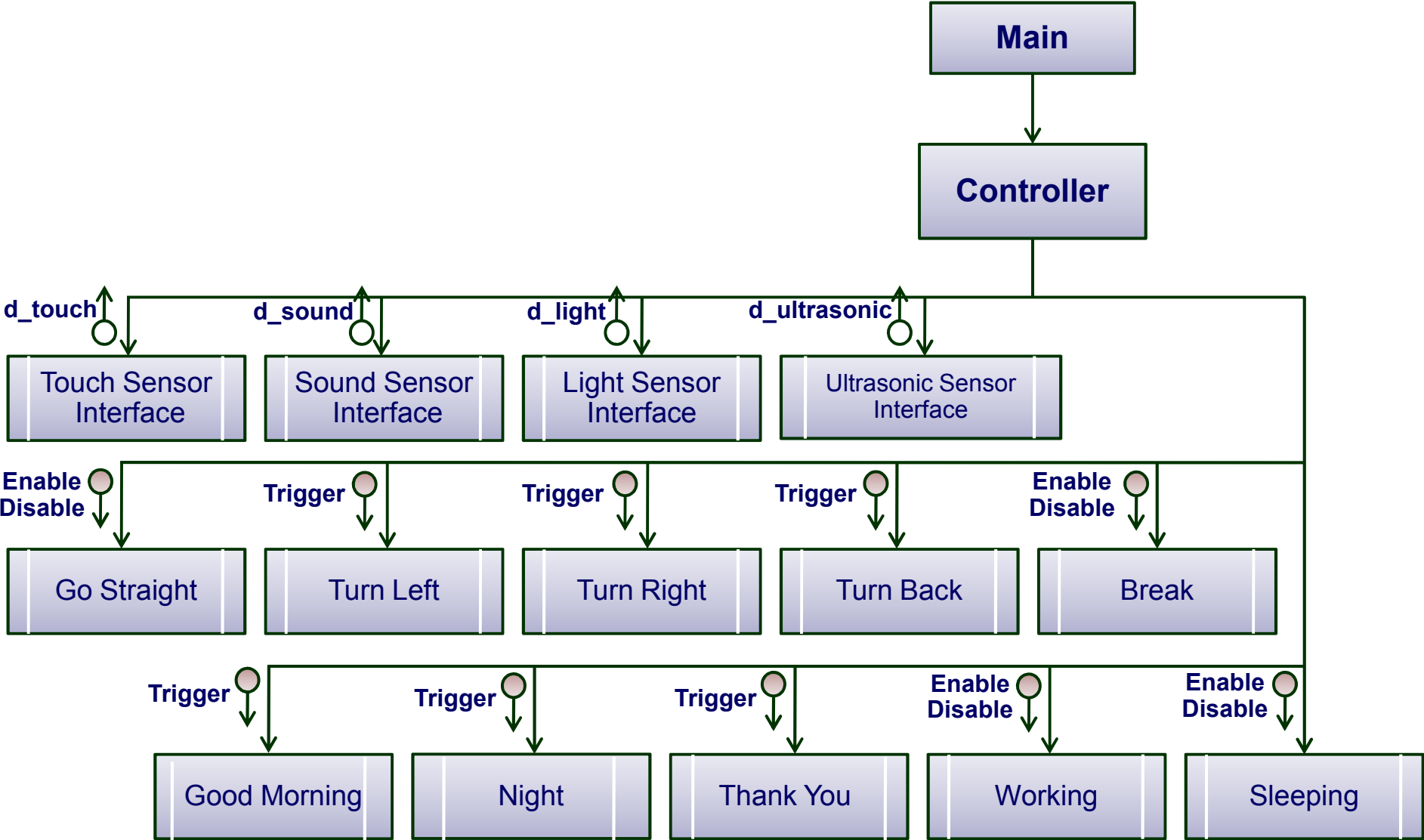
Module ID	ADR – OM13	Module Name	Night
Module Outline	Take command from control and the function outputs the sound “Night”		
Interface	void o_night() Function outputs the sound “Night”		

Module ID	ADR – OM21	Module Name	Working
Module Outline	Take command from control and display “Working” on the screen.		
Interface	void o_working() Function to display on the screen the “Working”		

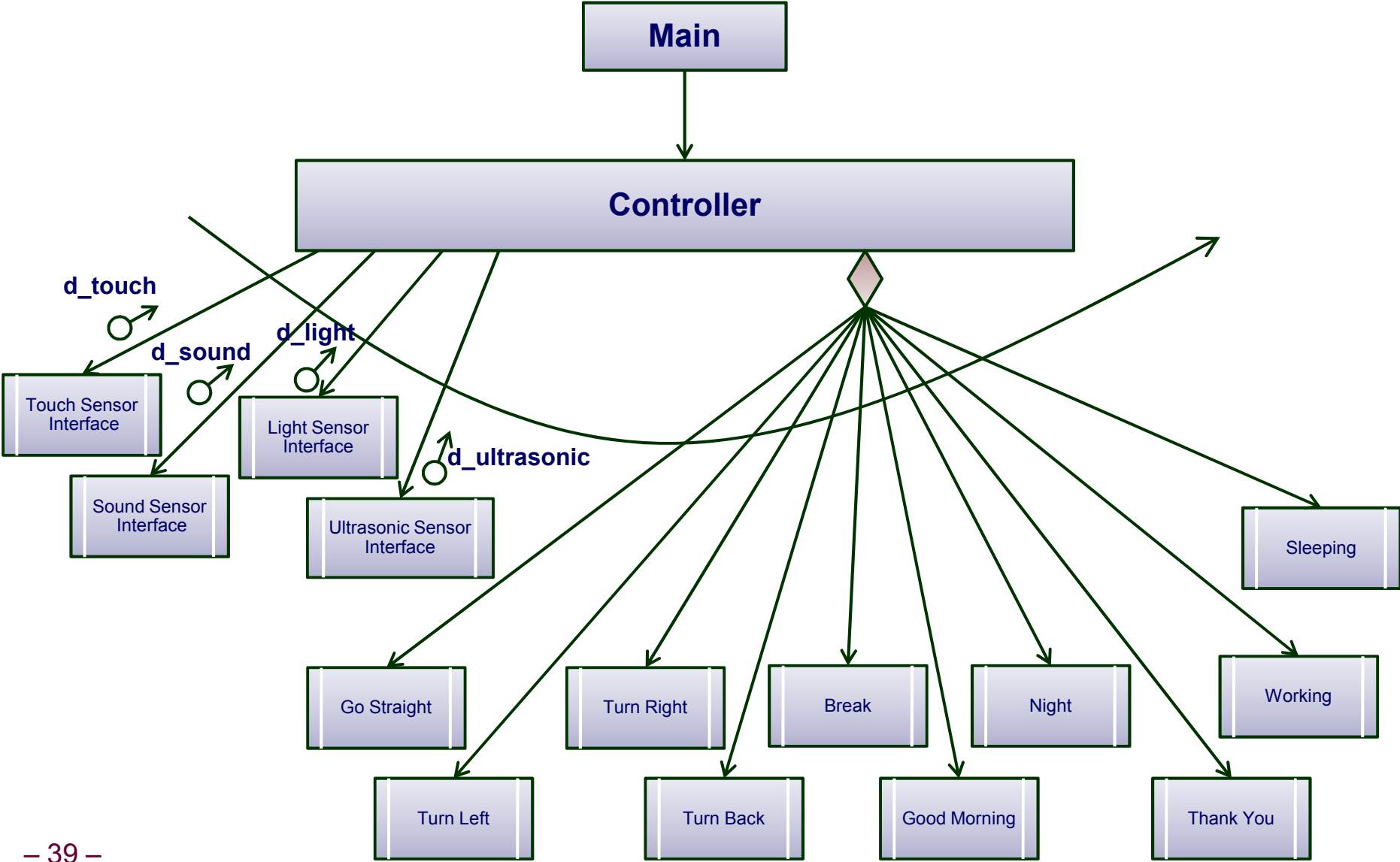
# Module Definition

Module ID	ADR – OM22	Module Name	Sleeping
Module Outline	Take command from control and display "Sleeping" on the screen.		
Interface	void o_sleeping() Function to display on the screen the "Sleeping"		

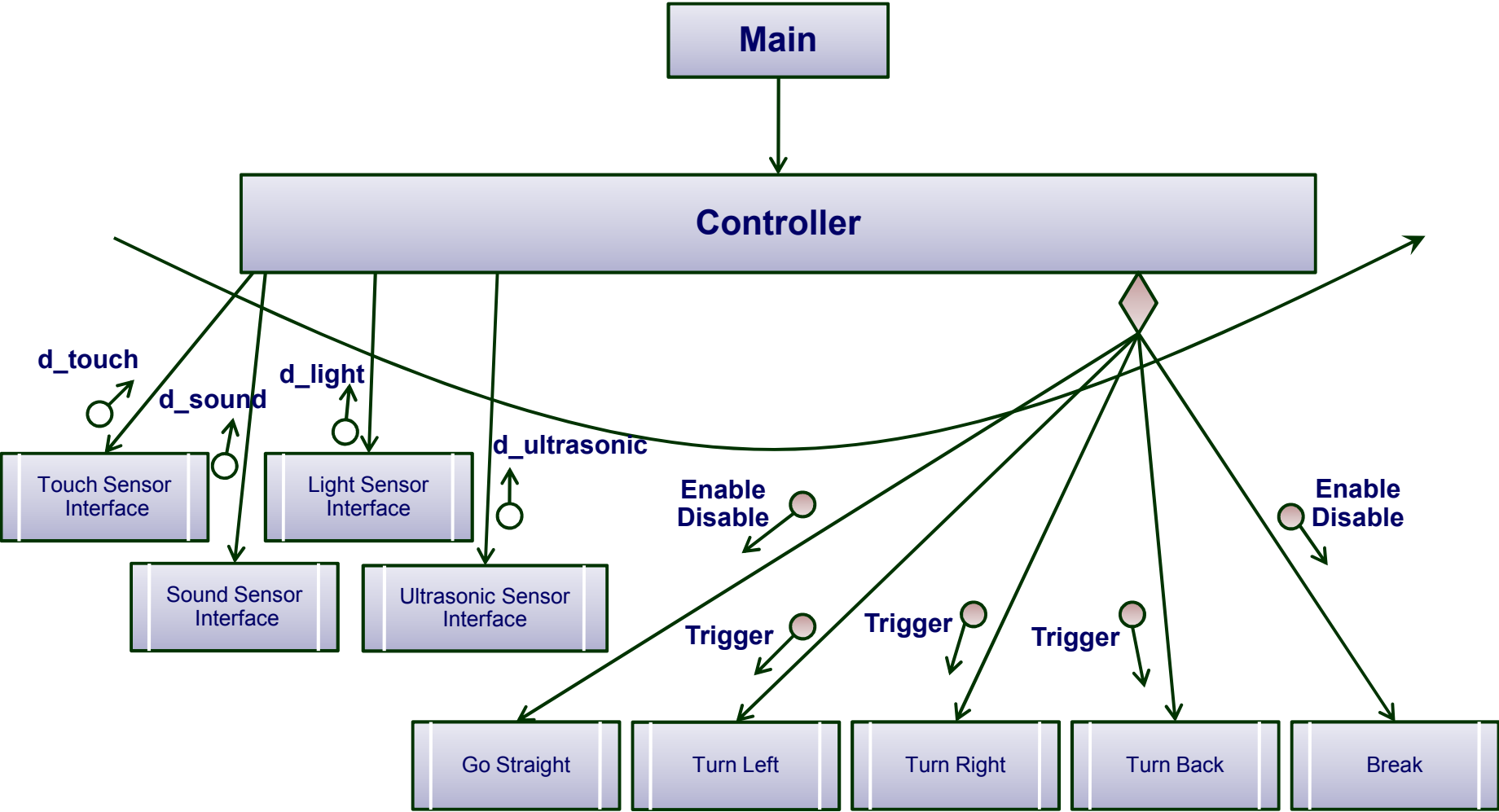
# Structured Charts – MMS (basic)



# Structured Charts – MMS (Advanced)



# Structured Charts – MMS (Advanced)





# Structured Charts – MMS (Advanced)

