Structured Analysis
Robot Vacuum Cleaner

Team Presentation #1

TEAM [T2]
김영현 200811415
정용겸 200811455
조성우 200811457
허준행 200811465

KU KONKUK UNIVERSITY

Contents

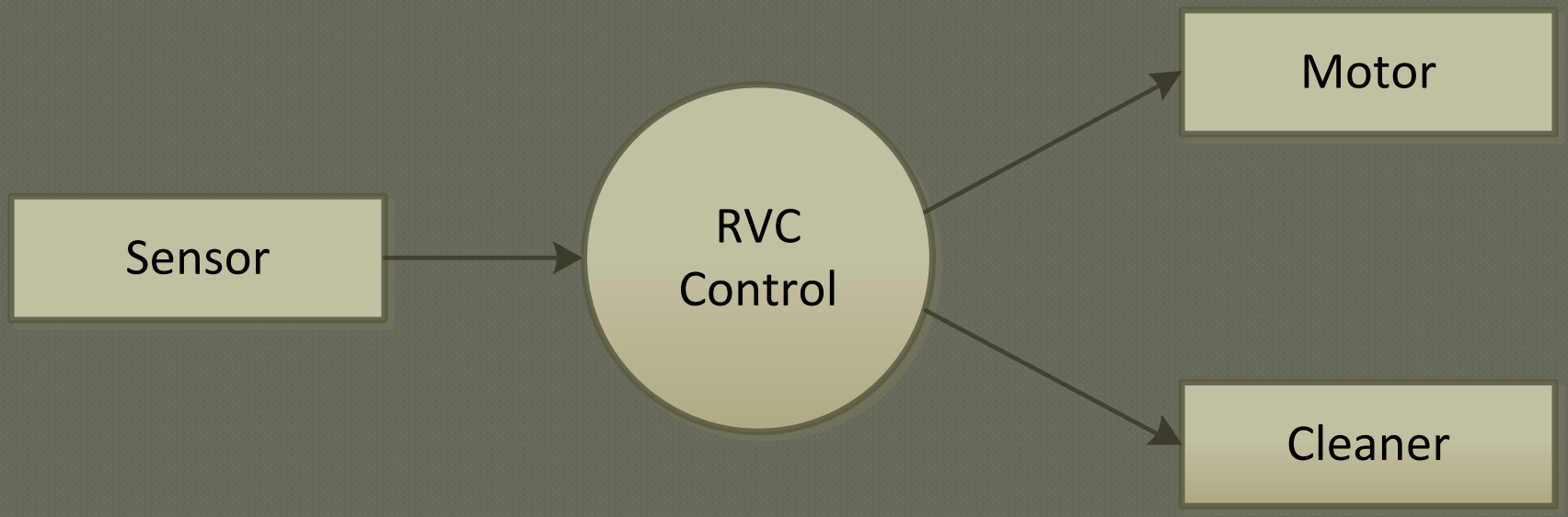
- Environmental Model
  - Statement of Purpose
  - System Context Diagram
  - Event List
- Behavioral Model
  - DFD (Data Flow Diagram)
  - State Machine
  - DD (Data Dictionary)
  - Process Specification

Statement of Purpose

- An RVC automatically cleans and mops household surface.
- It goes straight forward while cleaning.
- If its sensors found an obstacle, it stops cleaning, turns aside, and goes forward with cleaning.
- If it detects dust, power up the cleaning for a while
- We do not consider the detail design and implementation on HW controls.
- We only focus on the automatic cleaning function.

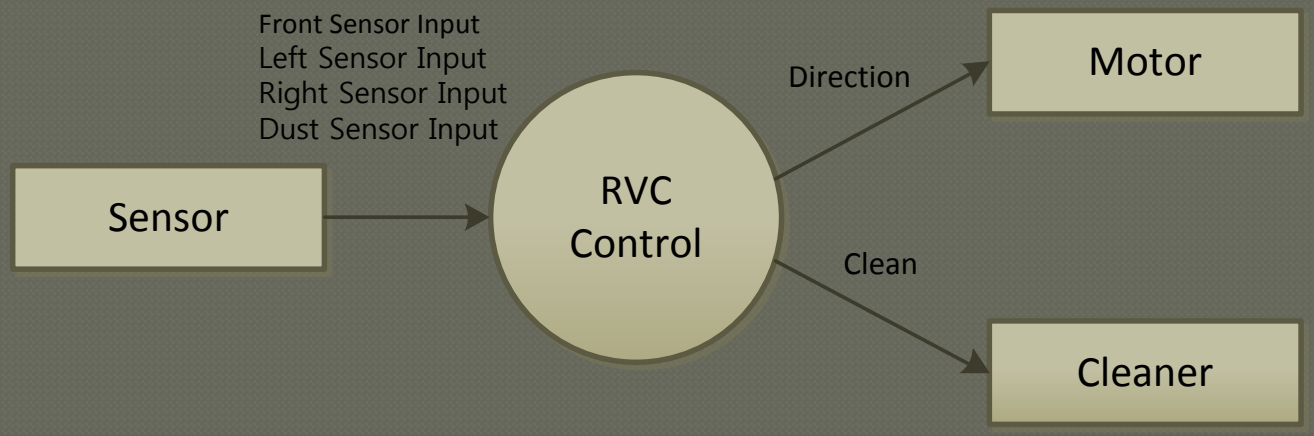
KU KONKUK UNIVERSITY

System Context Diagram



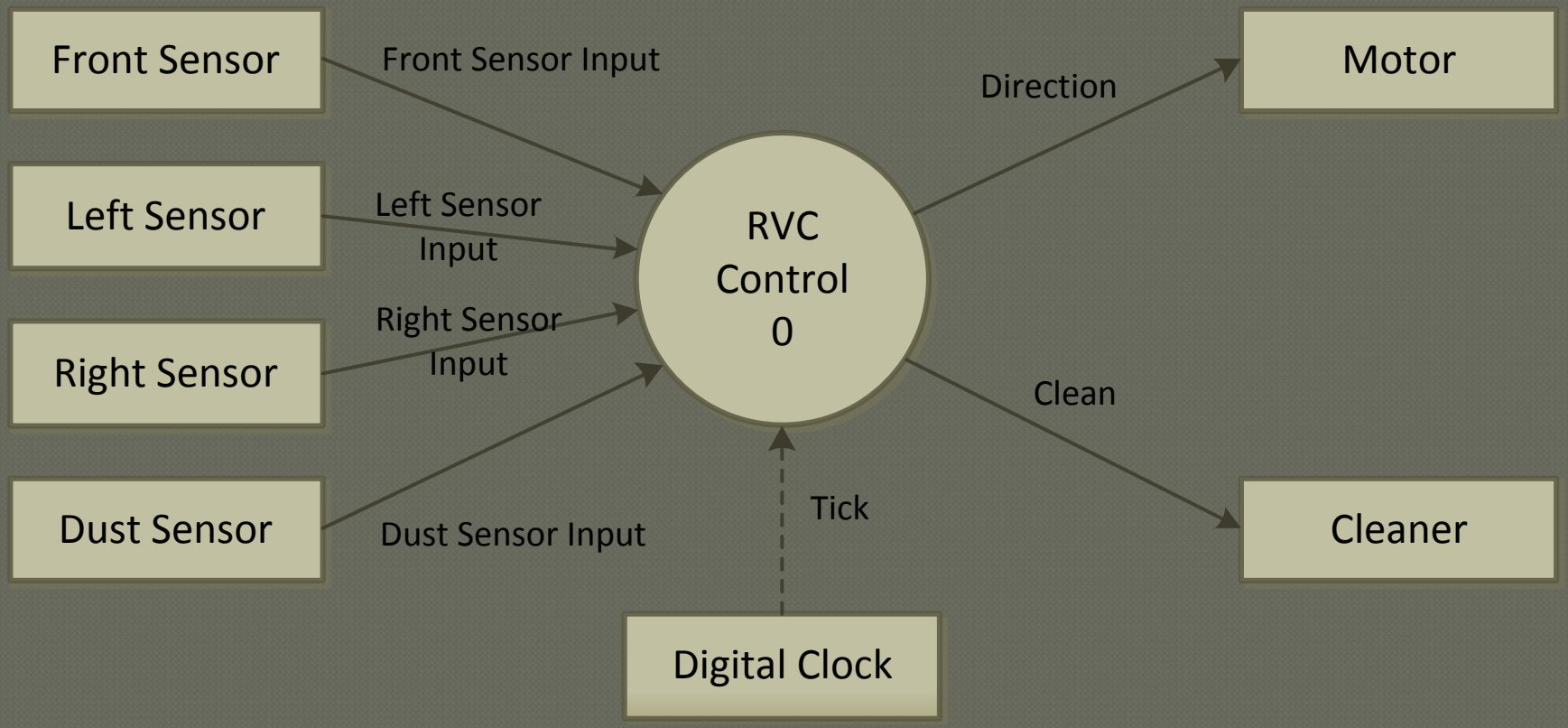
System Context Diagram
– Event List –

Input/Output Event	Description
Front Sensor Input	Detects obstacles in front of the RVC
Left Sensor Input	Detects obstacles in the left side of the RVC periodically
Right Sensor Input	Detects obstacles in the right side of the RVC periodically
Dust Sensor Input	Detects dust on the floor periodically
Direction	Direction commands to the motor (go forward / turn left with an angle / turn right with an angle)
Clean	Turn off / Turn on / Power-Up



KU KONKUK UNIVERSITY

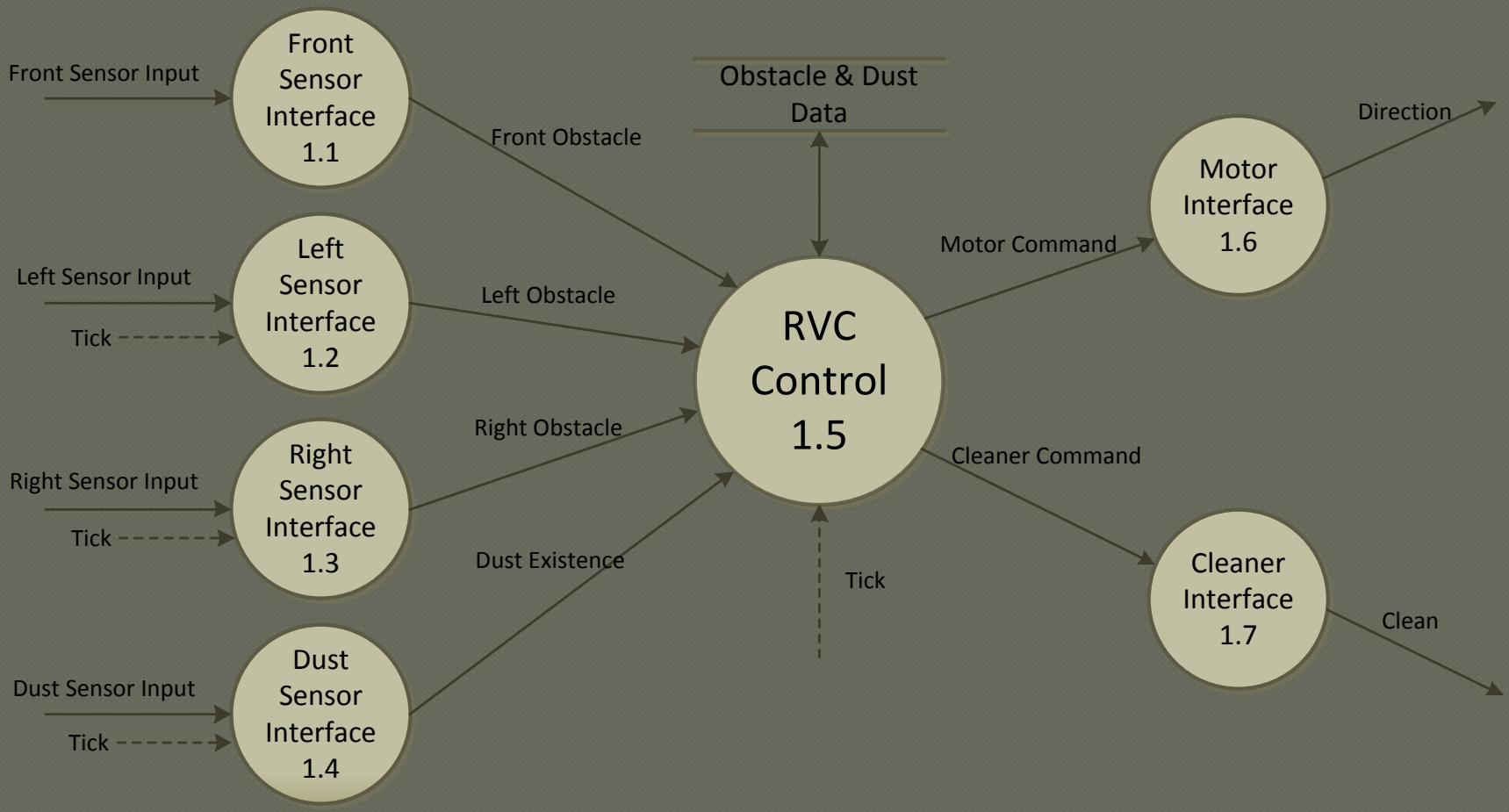
DFD level 0


Front Sensor
Left Sensor
Right Sensor
Dust Sensor
Front Sensor Input
Left Sensor Input
Right Sensor Input
Dust Sensor Input
RVC Control 0
Direction
Clean
Tick
Motor
Cleaner
Digital Clock

DFD level 0 Data Dictionary

Input/Output Event	Description	Format / Type
Front Sensor Input	Detects obstacles in front of the RVC	True / False , Interrupt
Left Sensor Input	Detects obstacles in the left side of the RVC periodically	True / False , Periodic
Right Sensor Input	Detects obstacles in the right side of the RVC periodically	True / False , Periodic
Dust Sensor Input	Detects dust on the floor periodically	True / False , Periodic
Direction	Direction commands to the motor (go forward / turn left with an angle / turn right with an angle)	Forward / Left / Right / Stop
Clean	Turn on / Turn off / Power-Up	On / Off / Up

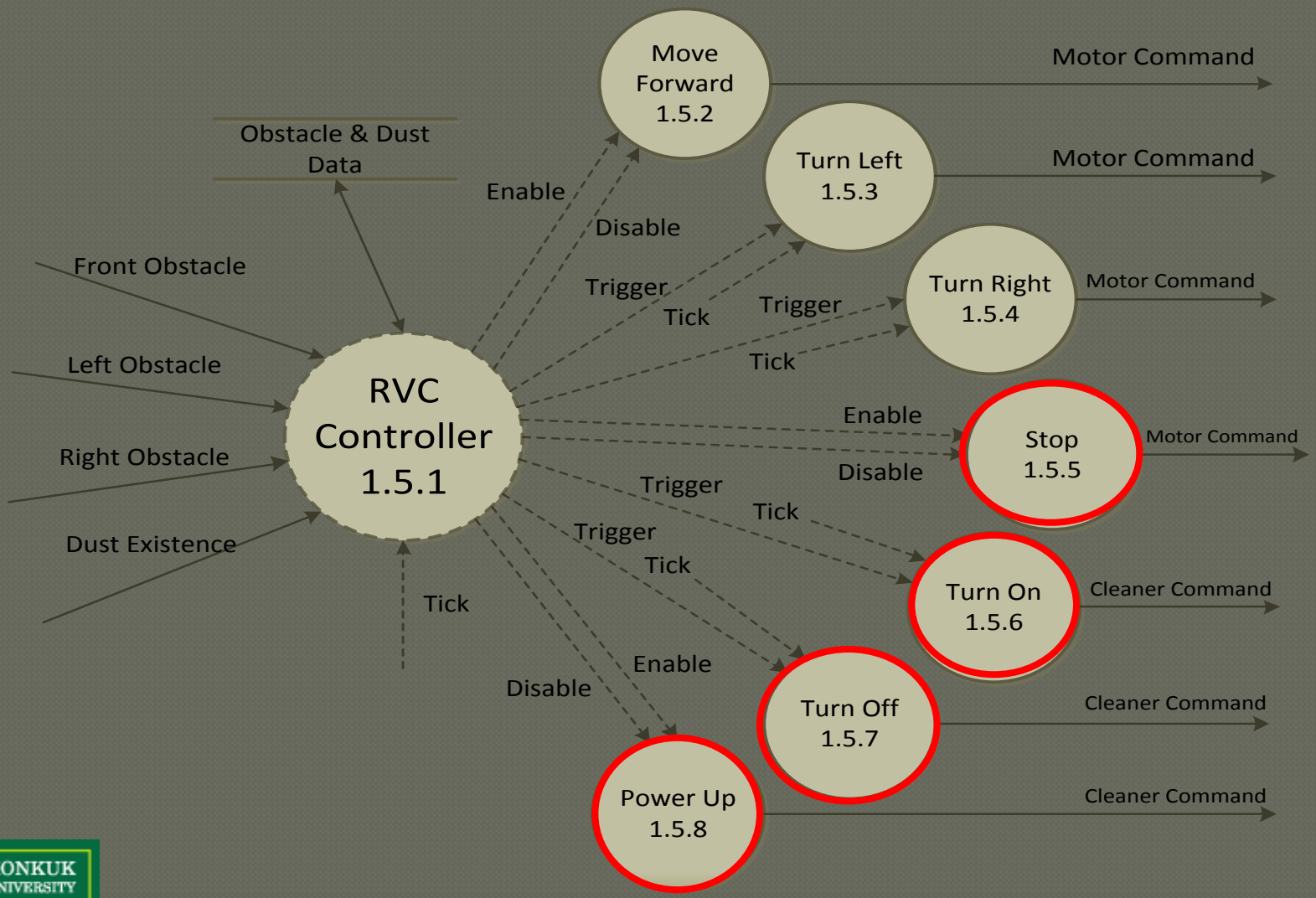
DFD level 1


Front Sensor Input
Front Sensor Interface 1.1
Front Obstacle
Obstacle & Dust Data
Direction
Motor Interface 1.6
Motor Command
Left Sensor Input
Tick
Left Sensor Interface 1.2
Left Obstacle
RVC Control 1.5
Right Obstacle
Right Sensor Input
Tick
Right Sensor Interface 1.3
Cleaner Command
Dust Existence
Tick
Cleaner Interface 1.7
Clean
Dust Sensor Input
Tick
Dust Sensor Interface 1.4

DFD level 1
Obstacle & Dust Data

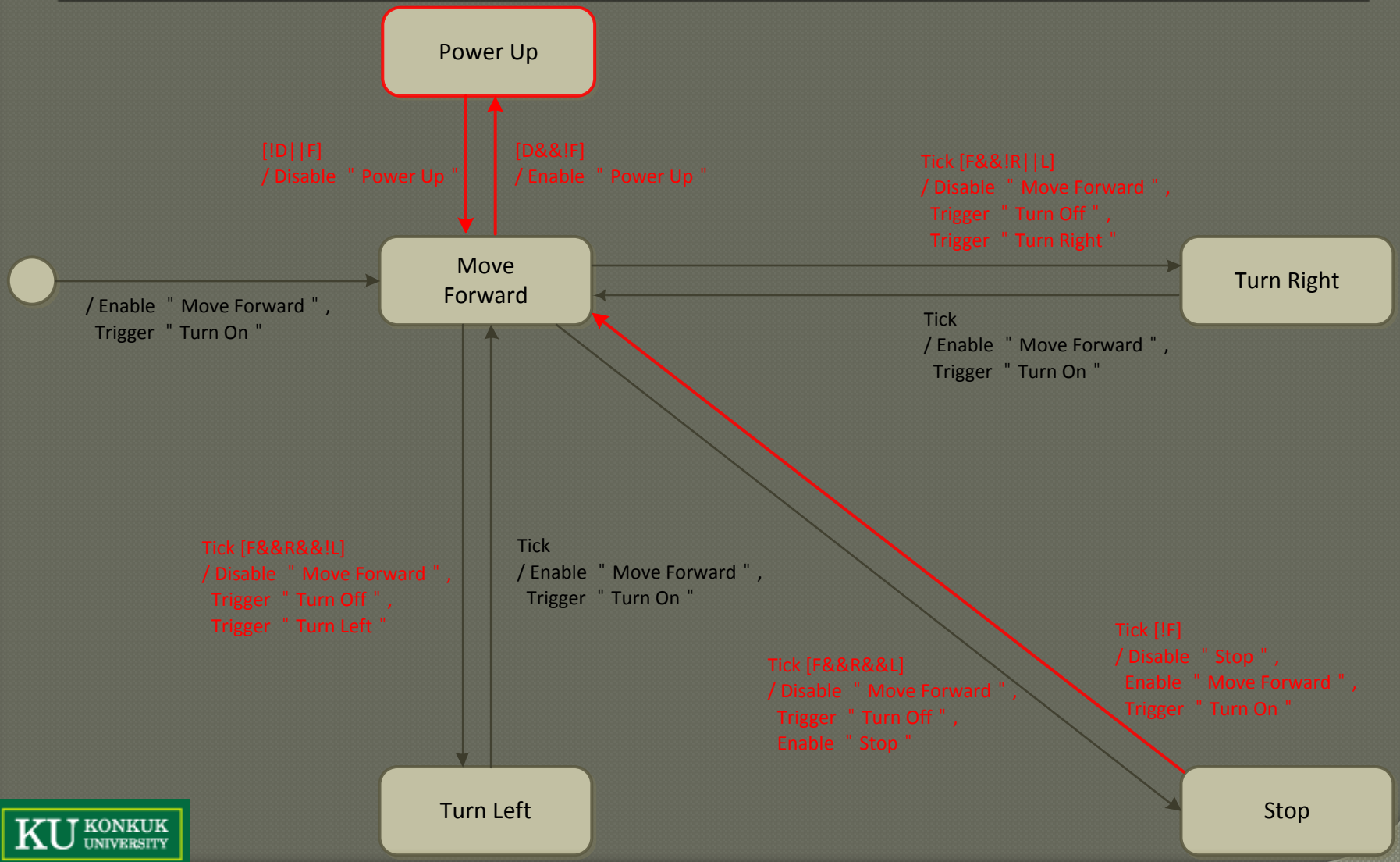
Data name	Description	Type
Front Obstacle	Whether Obstacle exists at front or not	Boolean
Left Obstacle	Whether Obstacle exists at left side or not	Boolean
Right Obstacle	Whether Obstacle exists at right side or not	Boolean
Dust Existence	Whether Dust exists or not	Boolean

DFD level 2



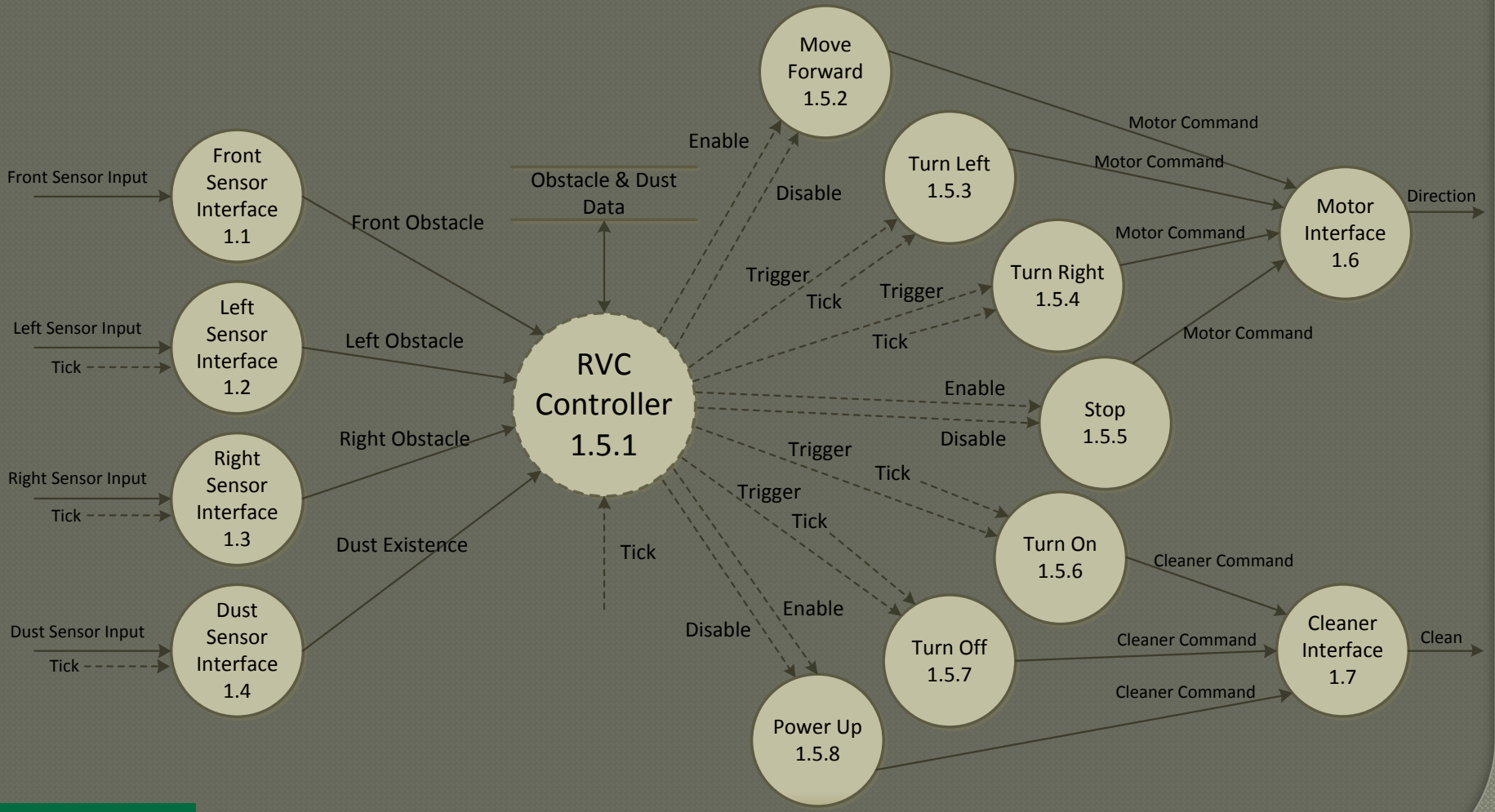
DFD level 3

State Machine for RVC Controller 1.5.1



KU KONKUK UNIVERSITY

DFD – overall



KU KONKUK UNIVERSITY

Data Dictionary

Data name	Description
Front Obstacle	Using information detected by **Front Sensor**, It saves data checking Whether Obstacle exists at front or not.
Left Obstacle	Using information detected by **Left Sensor**, It saves data checking Whether Obstacle exists at left side or not.
Right Obstacle	Using information detected by **Right Sensor**, It saves data checking Whether Obstacle exists at right side or not.
Dust Existence	Using information detected by **Dust Sensor**, It saves data checking Whether Dust exists or not.
Obstacle & Dust Data	It saves each boolean value which is converted from Obstacle & Dust information that sensor interface sent. (i.e. boolean[4] = { L, R, F, D })
Motor Command	It is data being sent to **Moter Interface** after it is processed by corresponding Motor process based on **Obstacle & Dust data**. It saves command about movement.
Cleaner Command	It is data being sent to **Cleaner Interface** after it is processed by corresponding Cleaner process based on **Obstacle & Dust data**. It saves command about cleaning the dust.

Process Specification

Reference No.	1.1
Name	Front Sensor Interface
Input	Front Sensor Input
Output	Front Obstacle
Process Description	Reads a analog value of the **Front sensor**, converts it into a digital value such as True/False, and assigns it into output variable **Front Obstacle** sent to Controller and then.

Reference No.	1.2
Name	Left Sensor Interface
Input	Left Sensor Input, Tick
Output	Left Obstacle
Process Description	Reads a analog value of the **Left sensor** periodically, converts it into a digital value such as True/False, and assigns it into output variable **Left Obstacle** sent to Controller and then.

KU KONKUK UNIVERSITY

Process Specification

Reference No.	1.3
Name	Right Sensor Interface
Input	Right Sensor Input, Tick
Output	Right Obstacle
Process Description	Reads a analog value of the **Right sensor** periodically, converts it into a digital value such as True/False, and assigns it into output variable **Right Obstacle** sent to Controller and then.

Reference No.	1.4
Name	Dust Sensor Interface
Input	Dust Sensor Input, Tick
Output	Dust Existence
Process Description	Reads a analog value of the **Dust sensor** periodically, converts it into a digital value such as True/False, and assigns it into output variable **Dust Existence** sent to Controller and then.

Process Specification

Reference No.	1.5.1
Name	RVC Controller
Input	Front/Left/Right Obstacle, Dust Existence, Obstacle & Dust Data, Tick
Output	Trigger, Enable, Disable
Process Description	It is main controller that determines RVC's state based on inputs (Front/Left/Right Obstacle and Dust Existence data) and then makes RVC command correct action by sending outputs to corresponding next process.

Reference No.	1.5.2
Name	Move Forward
Input	Enable, Disable
Output	Motor Command
Process Description	After receiving Enable/Disable value from RVC Controller, It sends **Moter Command** to **Moter Interface** in order to move RVC forward or stop moving.

Process Specification

Reference No.	1.5.3
Name	Turn Left
Input	Trigger, Tick
Output	Motor Command
Process Description	Being implemented by **RVC Controller**'s trigger, It sends **Motor Command** to **Motor Interface** in order that RVC can turn left with an angle.

Reference No.	1.5.4
Name	Turn Right
Input	Trigger, Tick
Output	Motor Command
Process Description	Being implemented by **RVC Controller**'s trigger, It sends **Motor Command** to **Motor Interface** in order that RVC can turn right with an angle.

Process Specification

Reference No.	1.5.5
Name	Stop
Input	Enable, Disable
Output	Motor Command
Process Description	After receiving Enable/Disable value from **RVC Controller**, It sends **Moter Command** to **Moter Interface** in order to enable/disable RVC to set Stop state. This process implements function that makes RVC turn right until there are no obstacles in front of RVC.

Reference No.	1.5.6
Name	Turn On
Input	Trigger, Tick
Output	Cleaner Command
Process Description	Being implemented by **RVC Controller**'s trigger, It sends **Cleaner Command** to **Cleaner Interface** in order that RVC can clean the dust by turning on **Cleaner**.

Process Specification

Reference No.	1.5.7
Name	Turn Off
Input	Trigger, Tick
Output	Cleaner Command
Process Description	Being implemented by **RVC Controller**'s trigger, It sends **Cleaner Command** to **Cleaner Interface** in order that RVC can stop cleaning by turning off **Cleaner**.

Reference No.	1.5.8
Name	Power Up
Input	Enable, Disable
Output	Cleaner Command
Process Description	After receiving Enable/Disable value from **RVC Controller**, It sends **Cleaner Command** to **Cleaner Interface** in order to power up cleaning.

Process Specification

Reference No.	1.6
Name	Motor Interface
Input	Motor Command
Output	Direction (Motor Action)
Process Description	It enables RVC to move or stop by sending **Direction** data which is converted from corresponding **Motor Command** to analog value to **Motor**.

Reference No.	1.7
Name	Cleaner Interface
Input	Cleaner Command
Output	Clean (Cleaner Action)
Process Description	It enables RVC to start or stop cleaning by sending **Clean** data converted from corresponding **Cleaner Command** to analog value to **Cleaner**.

Q & A

KU KONKUK UNIVERSITY