

Software Requirement Analysis for Public Transportation System

Project Team

Team1

Date

2014-12-12

201111347 김태호 / tae_ho_@naver.com
201111356 박준한 / junhan0531@gmail.com
201111360 손준익 / sji6227@naver.com
201111367 여승훈 / gnszz91@naver.com

Table of Contents

- 1 Introduction
 - 1.1 Purpose
 - 1.2 Scope
 - 1.3 Definition, acronyms, and abbreviations
 - 1.4 Reference
 - 1.5 OverDisplay
- 2 Overall Description
 - 2.1 Product Perspective
 - 2.2 Product functions
 - 2.3 User characteristics
 - 2.4 Constraints and Assumptions
- 3 Structured Analysis
 - 3.1 Terminal System
 - 3.1.1 System Context Diagram
 - 3.1.1.1 Basic System Context Diagram
 - 3.1.1.2 Event List
 - 3.1.1.3 The System Context Diagram
 - 3.1.2 Data Flow Diagram
 - 3.1.2.1 DFD level 0
 - 3.1.2.1.1 DFD

3.1.2.1.2 Process Specification

3.1.2.1.3 Data Dictionary

3.1.2.2 DFD level 1

3.1.2.2.1 DFD

3.1.2.2.2 Process Specification

3.1.2.2.3 Data Dictionary

3.1.2.3 DFD level 2

3.1.2.3.1 DFD

3.1.2.3.2 Process Specification

3.1.2.3.3 Data Dictionary

3.1.2.4 DFD level 3

3.1.2.4.1 DFD

3.1.2.4.2 Process Specification

3.1.2.4.3 Data Dictionary

3.1.2.4.4 State Transition Diagram Controller 2.1.2

3.1.2.5 DFD level 4

3.1.2.5.1 DFD

3.1.2.5.2 Process Specification

3.1.2.5.3 State Transition Diagram Controller 2.1.1.1

3.1.2.6 Overall DFD

3.2 Fundamental Management System

3.2.1 System Context Diagram

3.2.1.1 Basic System Context Diagram

3.2.1.2 Event List

3.2.1.3 The System Context Diagram

3.2.2 Data Flow Diagram

3.2.2.1 DFD level 0

3.2.2.1.1 DFD

3.2.2.1.2 Process Specification

3.2.2.1.3 Data Dictionary

3.2.2.2 DFD level 1

3.2.2.2.1 DFD

3.2.2.2.2 Process Specification

3.2.2.2.3 Data Dictionary

3.2.2.3 DFD level 2

3.2.2.3.1 DFD

3.2.2.3.2 Process Specification

3.2.2.3.3 Data Dictionary

3.2.2.3.4 State Transition Diagram Time Checker 1.3.1

3.2.2.4 Overall DFD

1 Introduction

1.1 Purpose

본 문서는 2014 년 건국대학교의 소프트웨어공학 개론 강의의 실습과제를 설명한다. 실습과제는 대중교통시스템(PTS : Public Transportation System)을 소프트웨어만을 이용한 가상의 시스템으로 구현하는 것을 의미한다.

1.2 Scope

현재 운영중인 PTS 는 <그림 1>(김형환 2010)과 같다. 본 프로젝트는 전체 PTS 중 지하철, 버스 및 정산 시스템만을 대상으로 구현하는 것으로 규모를 제한한다. 또한 버스 1 대와 지하철 2 호선 중 5 개 역(건대입구, 왕십리, 합정, 신림, 강남)만을 대상으로 한다.

모든 시스템은 SW 만으로 구현한다. HW 가 필요한 부분은 SW 모듈을 만들어 가상의 HW 를 구현한다.

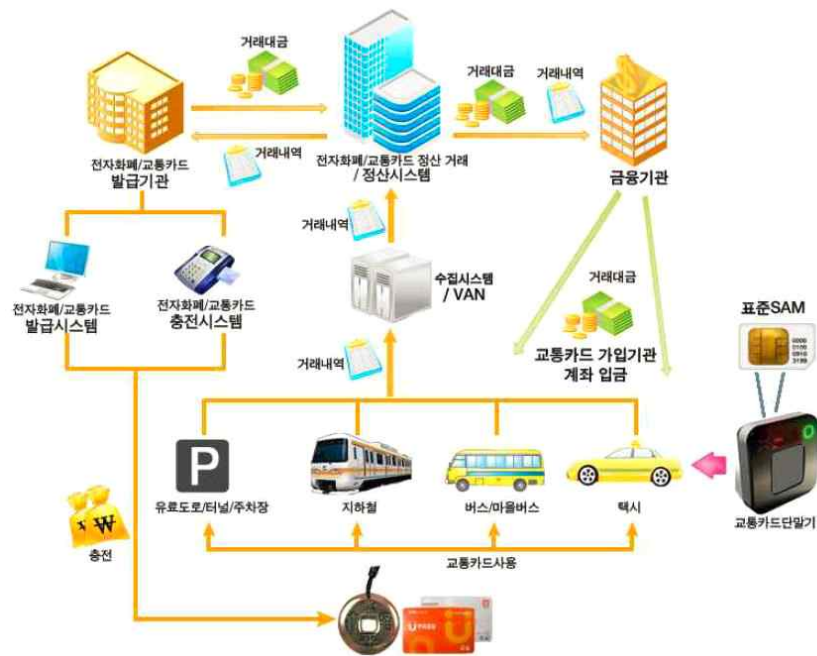


그림 1 서울의 교통카드 운영시스템

1.3 Definition, acronyms, and abbreviations

HW : hardware

PTS : Public Transportation System

SW : Software

태그 : 카드와 단말기가 통신할 수 있도록 하는 행위; 승·하차 시 요금 결제를 위한 행위

1.4 Reference

(김형환 2010) 김형환, 신동석 "교통카드 무인판매/충전기 통합 운영시스템 개발", 韓國컴퓨터情報學會論文誌 15(3), 99-109, 2010

(김경선, 2009) 김경선, "교통카드 시스템 사례 연구-수도권 교통카드 중심". 수도권 교통 본부, 2009

1.5 OverDisplay

2 장 개발 대상에 대한 설명; 3 장 세부 기능 명세

2 Overall Description

2.1 Product Perspective

SW 로 개발된 PTS 는 총 3 가지로 구성된다. 버스용 단말기와 지하철용 단말기, 정산 시스템이다.

2.2 Product functions

버스용 단말기는 버스에 부착돼, 탑승 태그와 하차 태그가 가능하다. 기본료를 지불하면 1 회 탑승이 가능하다.

지하철용 단말기는 역에 부착돼 탑승 태그와 하차 태그가 가능하다. 1 개역 이하를 이동하면 기본료가 부가되며 두 개역을 이동하면 추가 요금이 부가된다.

버스와 지하철 간에는 정해진 시간 내에 환승이 가능하다. 지하철에서 버스로 환승한 경우는 단위 시간 당 버스의 환승 요금이 추가로 부가된다. 버스에서 지하철로 환승한 경우는 한 역당 지하철의 환승 요금 추가로 부가된다. 하차 시 단말기에 태그를 하지 않으면 환승은 적용되지 않는다.

정산은 하루에 한 번 이뤄진다. 버스과 지하철의 기록을 분석하고 버스과 지하철에 각각 수익을 배분한다.

2.3 User characteristics

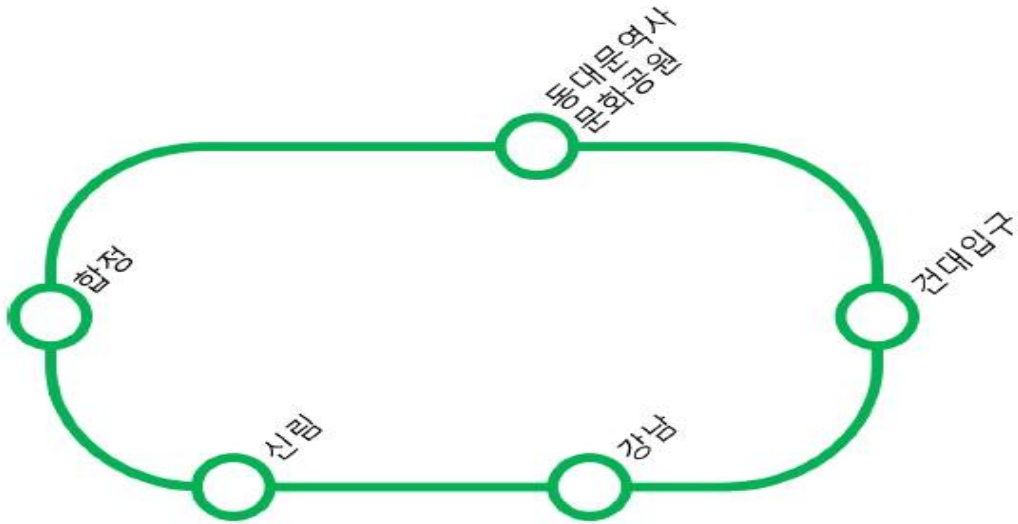
사용자는 대중교통 승차 시 카드를 태그한다.

사용자는 하차 시 카드를 태그할 수도 있고 아닐 수도 있다.

2.4 Constraints and Assumptions

버스 승차는 지역에 상관없이 할 수 있다.

지하철은 2 호선 역 중 5 개만 고려한다: 건대입구, 동대문역사문화공원, 합정, 신림, 강남.



하루는 3 분으로 가정한다.

다음 날 운행이 시작하기 전까지는 정산이 반드시 이루어 져야 한다.

정산 후 모든 프로그램의 정보는 초기화 된다. (사용자 카드 정보 제외)

환승이 가능한 시간은 15 초 이내이다.

버스 환승 시 추가요금의 기준인 단위 시간은 30 초이다.

버스와 지하철의 기본료는 1050 원이다.

교통카드는 지정된 텍스트 파일로 가능하고, 교통카드 태그 행위를 해당 파일을 입력 하는 것으로 가정한다.

잔액이 모자를 경우 태우지 않는다.

버스 환승 최고 부과금액인 700 원이 남아 있지 않으면 버스로 환승시키지 않는다.

지하철 환승 최고 부과금액인 600 원이 남아 있지 않으면 지하철로 환승시키지 않는다.

하루(3 분) 종료 시 탑승되어 있는 승객은 미정산으로 처리한다.

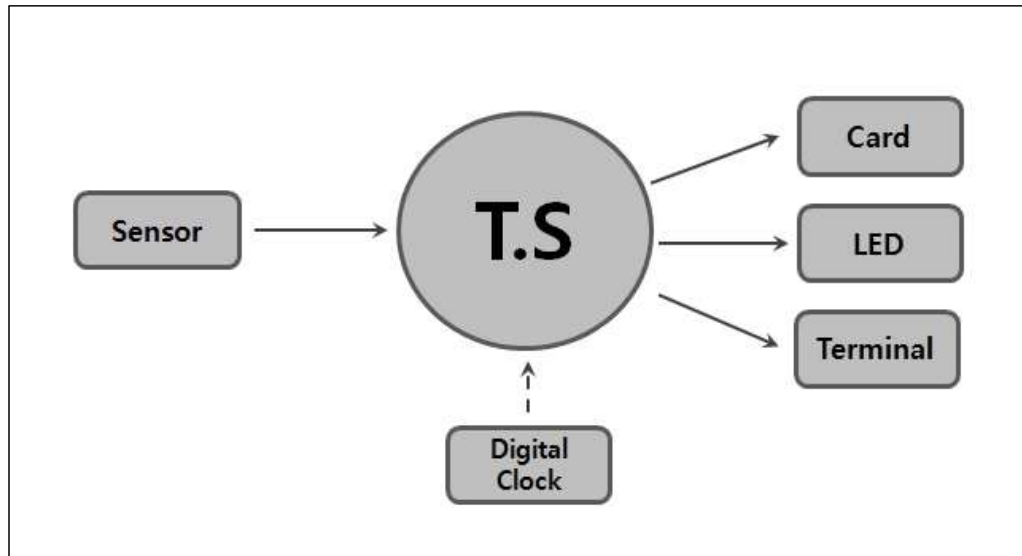
정산시 소수점 이하는 반올림한다.

3 Structured Analysis

3.1 Terminal System

3.1.1 System Context Diagram

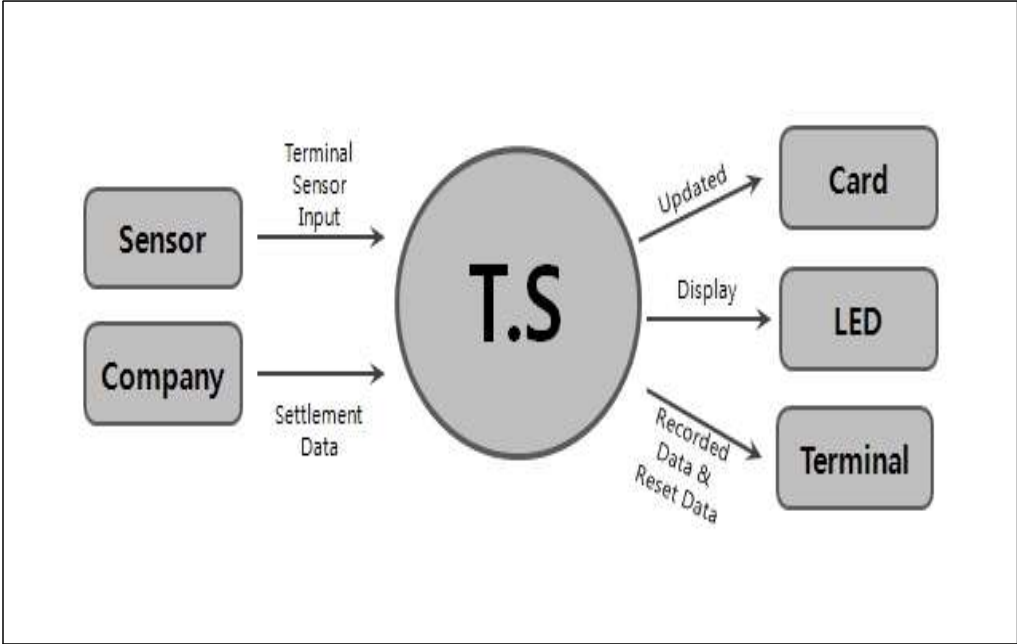
3.1.1.1 Basic System Context Diagram



3.1.1.2 Event List

Input / Output Event	Description
Terminal Sensor Input	단말기 센서에 태그 되었을 때 교통카드의 정보
Updated	단말기 시스템 안에서 계산된 결과를 가지고 갱신시킬 카드의 정보
Display	단말기 시스템 안에서 계산된 결과를 가지고 LED 화면에 출력할 정보 (Wait Mode / Correct Mode / Warning Mode)
Settlement Status	Company 에서 보내준 정산 완료 여부 정보
Recorded Data	터미널에 누적계산기록에 기록될 결제 정보
Reset Data	터미널 누적계산기록을 초기화시킬 여부를 결정할 정보

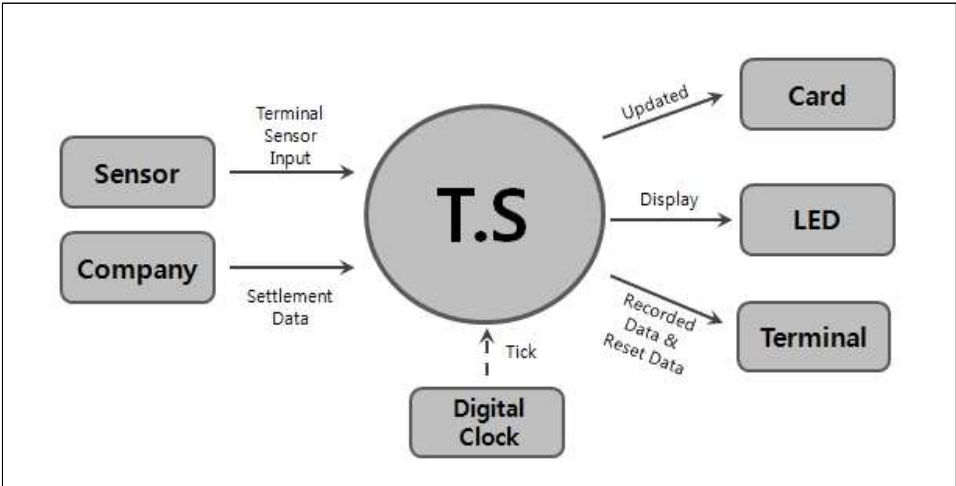
3.1.1.3 The System Context Diagram



3.1.2 Data Flow Diagram

3.1.2.1 EDFD level 0

3.1.2.1.1 DFD



3.1.2.1.2 Process Specification

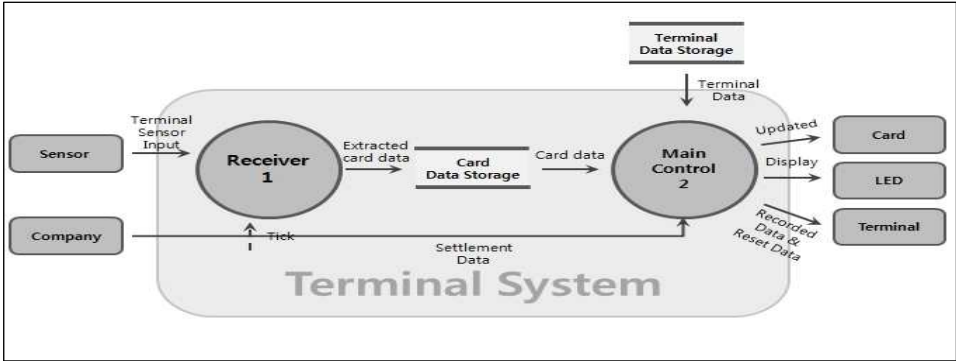
Reference No.	0
Name	Terminal System
Input	Terminal Sensor Input, Settlement Satus
Output	Updated, Display, Calculated Result, Reset Data
Process Description	Sensor 로부터 받은 카드 정보를 가지고 Terminal System 안에 있는 정보랑 비교한 후 Card 를 갱신, LED 화면에 출력하고 Terminal 누적결제기록을 기록 정보를 전송한다. Company 로부터 받은 Settlement Status 를 가지고 Terminal 을 Reset 할 데이터를 전송한다.

3.1.2.1.3 Data Dictionary

Input / Output Event	Description	Format / Type
Terminal Sensor Input	단말기 센서에 태그 되었을 때 교통카드의 정보	True / False, Interrupt
Updated	단말기 시스템 안에서 계산된 결과를 가지고 갱신시킬 카드의 정보	Updated Data
Display	단말기 시스템 안에서 계산된 결과를 가지고 LED 화면에 출력할 정보	Wait / Correct / Warning
Settlement Data	Company 에서 보내준 정산 완료 여부 정보	Settlement Data Periodic
Recorded Data	터미널에 누적계산기록에 기록될 결제 정보	Recorded Data
Reset Data	누적결제기록 파일을 초기화시킬 정보	Reset

3.1.2.2 DFD level 1

3.1.2.2.1 DFD



3.1.2.2.2 Process Specification

Reference No.	1
Name	Receiver
Input	Terminal Sensor Input
Output	Extracted Card Data
Process Description	Sensor 로부터 받아온 모든 카드 정보를 터미널 시스템 안에 필요한 카드 정보만을 Card Data Storage 로 전송한다.

Reference No.	2
Name	Main Control
Input	Card Data, Terminal Data, Settlement Data
Output	Updated, Display, Calculated Result, Reset Data
Process Description	<p>Card Data Storage 에서 Card Data 를 받고 Terminal Data Storage 에서 Terminal Data 를 입력 받아서 카드를 갱신시킬 정보를 Card 로 보내고 LED 화면에 출력할 정보를 LED 로 보내고 Terminal 에 누적결제기록 정보를 보낸다.</p> <p>Company 로부터 받은 Settlement Data 를 기반으로 Reset Data 를 Terminal 에 보낸다.</p>

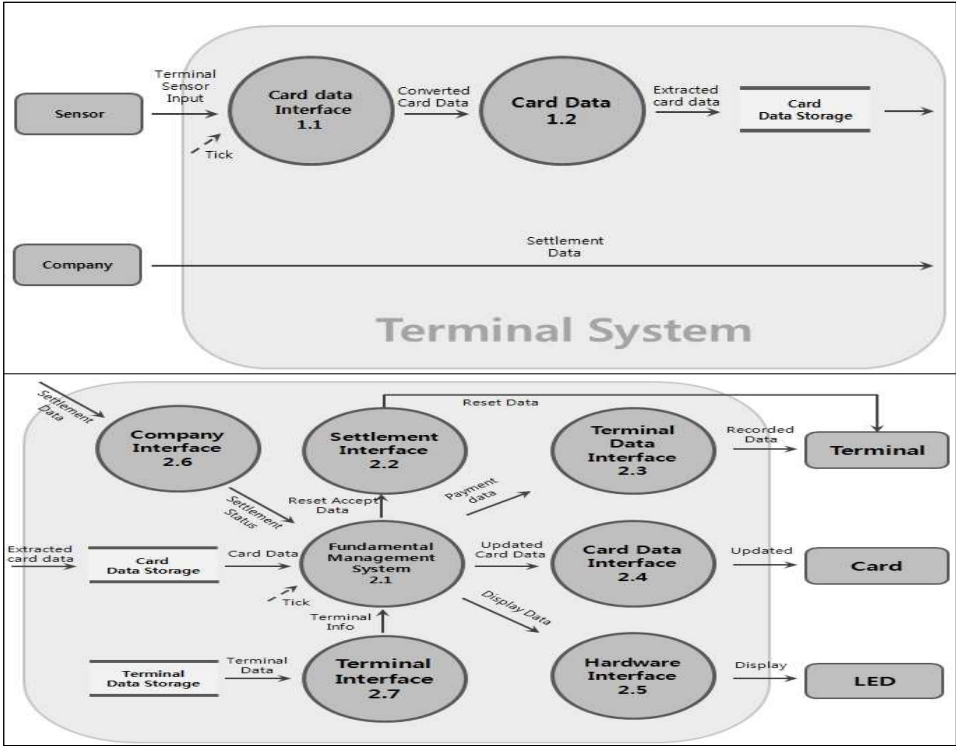
3.1.2.2.3 Data Dictionary

Input / Output Event	Description	Format / Type
Card Data Storage	단말기로부터 받은 카드 정보 저장소	/
Terminal Data Storage	단말기 시스템 내부의 있는 단말기 정보의 저장소	/
Extracted Card Data	Receiver 프로세서로부터 추출되어 나온 Card Data	Structure
	on_board(boolean) : 승하차여부 balance(int) : 잔액 terminal_ID(Character) : 단말기 고유 아이디 time1(Character) : 전태그된시간 time2(Character) : 전전태그된시간 c_time(Character) : 현재시간 transportation(Character) : 교통수단 transportation1(Character) : 전교통수단 transportation2(Character) : 전전교통수단 count(int) : 단말기 태그 횟수	
Converted Card Data	Card Data Storage 로부터 받아온 계산에 필요한 Card Data	Structure
	time(Character) : 태그된시간 transportation(Character) : 교통수단 on_board(boolean) : 승하차여부 balance(int) : 잔액 terminal_ID(Character) : 단말기 고유 아이디	
Terminal Data	Terminal Data Storage 로부터 받아온 계산에 필요한 Terminal Data	Structure
	time(Character) : 태그된시간 transportation(Character) : 교통수단 on_board(boolean) : 승하차여부 terminal_ID(Character) : 단말기 고유 아이디	

Input / Output Event	Description	Format / Type
Extracted Terminal Data	Receiver 프로세서로부터 추출되어 나온 Terminal Data	Structure
	transportation(Character) : 교통수단 terminal_ID(Character) : 단말기 고유 아이디 count(int) : 단말기 태그 횟수	

3.1.2.3 DFD level 2

3.1.2.3.1 DFD



3.1.2.3.2 Process Specification

Reference No.	1.1
Name	Card Data Interface
Input	Terminal Sensor Input, Tick
Output	Converted Card Data
Process Description	센서로부터 받은 아날로그신호를 디지털신호로 변환한다.

Reference No.	1.2
Name	Card Data
Input	Converted Card Data
Output	Extracted Card Data
Process Description	입력받은 데이터중 필요한 데이터만을 Card Data Storage 로 추출하여 전송한다.

Reference No.	2.1
Name	Fundamental Management System
Input	Card Data, Terminal Data, Settlement Status, Tick
Output	Reset Accept Data, Payment Data, Updated Card Data, Display Data
Process Description	Settlement Status 와 Terminal Data, Card Data 를 입력받아 관리 및 계산 후 Output Data 를 전송한다.

Reference No.	2.2
Name	Settlement Interface
Input	Reset Accept Data
Output	Reset Data
Process Description	Reset Accept Data 를 받아 Terminal 에 Reset Data File 을 보낸다.

Reference No.	2.3
Name	Terminal Data Interface
Input	Payment Data
Output	Recorded Data
Process Description	Payment Data 를 입력받아 Terminal 에 누적결제기록 File 에 저장할 Recorded Data 를 보낸다.

Reference No.	2.4
Name	Card Data Interface
Input	Updated Card
Output	Updated
Process Description	Updated Card Data 를 받아서 Card 정보를 Update 시키는 정보를 보낸다.

Reference No.	2.5
Name	Hardware Interface
Input	Display Data
Output	Display
Process Description	Display Data 를 받아서 LED 화면에 출력할 Display 정보를 보낸다.

Reference No.	2.6
Name	Company Interface
Input	Settlement Data
Output	Settlement Status
Process Description	Company Terminator 로부터 Settlement Data 를 받아서 F.M.S 프로세스로 Settlement Status 를 보낸다.

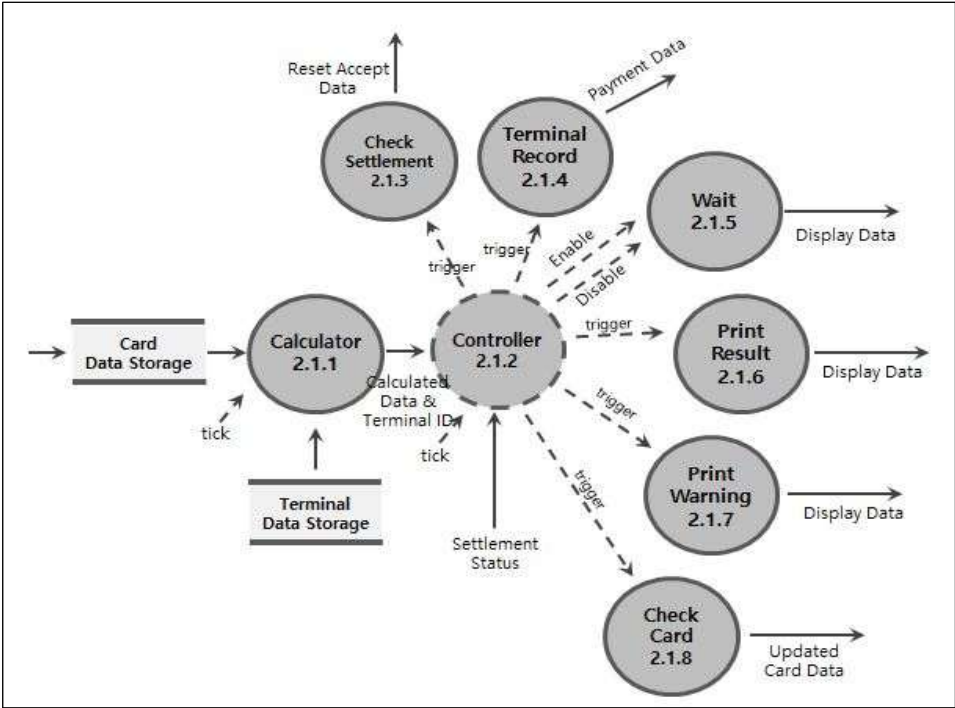
Reference No.	2.7
Name	Terminal Interface
Input	Terminal Data
Output	Terminal Info
Process Description	Terminal 시스템 안에 있는 Terminal Data Storage 로부터 Terminal Info 를 받아 F.M.S 프로세스로 Terminal Data 를 보낸다.

3.1.2.3.3 Data Dictionary

Input / Output Event	Description	Format / Type
Updated Card Data	Fundamental_Management_System (Process 2.1, 이하 F.M.S) 프로세스에서 처리되어 나온 카드 갱신 데이터	Structure
	time(Character) : 태그된시간 transportation(Character) : 교통수단 on_board(boolean) : 승하차여부 balance(int) : 잔액 Terminal_ID(Character) : 단말기 ID count(int) : 단말기 태그 횟수	
Display Data	Display(String) : 상태 결정 fee(int) : 금액 Calculated Balance(int) : balance-fee	Structure
Reset Accept Data	time(Character) : 다음날짜값 기록 reset(boolean) : 초기화 여부	Structure
Payment Data	F.M.S 프로세스에서 처리되어 나온 결제기록데이터	Structure
	time(Character) : 태그된시간 transportation(Character) : 교통수단 on_board(boolean) : 승하차여부 fee(int) : 금액 Terminal_ID(Character) : 단말기 ID count(int) : 단말기 태그 횟수	
Settlement Status	Company Interface 프로세스에서 처리되어 F.M.S 프로세스에 입력되는 정산여부 데이터	boolean
Terminal Info	Terminal Interface 프로세스에서 처리되어 F.M.S 프로세스에 입력되는 Terminal 정보	Structure
	Terminal_ID(Character) : 단말기 ID count(int) : 태그횟수	

3.1.2.4 DFD level 3

3.1.2.4.1 DFD



3.1.2.4.2 Process Specification

Reference No.	2.1.1
Name	Calculator
Input	Card Data, Terminal Data, Tick
Output	Calculated Data, Terminal ID
Process Description	Terminal Data 와 Card Data 를 입력 받아 계산된 정보와 환승 여부에 따라 Terminal ID 를 보내준다.

Reference No.	2.1.2
Name	Controller
Input	Calculated Data, Terminal ID, Settlement Status
Output	Enable, Disable, Trigger
Process Description	입력받은 Data 를 종합하여 조건에 맞는 Process 를 실행시킨다.

Reference No.	2.1.3
Name	Check Settlement
Input	Trigger
Output	Reset Accept Data
Process Description	현재시간 % 180 ==0 일 때 프로세스가 실행 되어 Reset Accept Data 를 Settlement Interface 로 전송한다.

Reference No.	2.1.4
Name	Terminal Record
Input	Trigger
Output	Payment Data
Process Description	Calculated Balance >=0 일 때 프로세스가 실행 되어 Terminal Data Interface 로 Payment Data 를 전송한다.

Reference No.	2.1.5
Name	Wait
Input	Enable, Disable
Output	Display Data
Process Description	Hardware Interface 에 Display Data(Wait)를 전송

Reference No.	2.1.6
Name	Print Result
Input	Calculated Data, trigger, tick
Output	Display Data
Process Description	Calculated Balance >=0 일 때 프로세스가 실행 되어 Hardware Interface 에 Display Data(Correct)를 전송

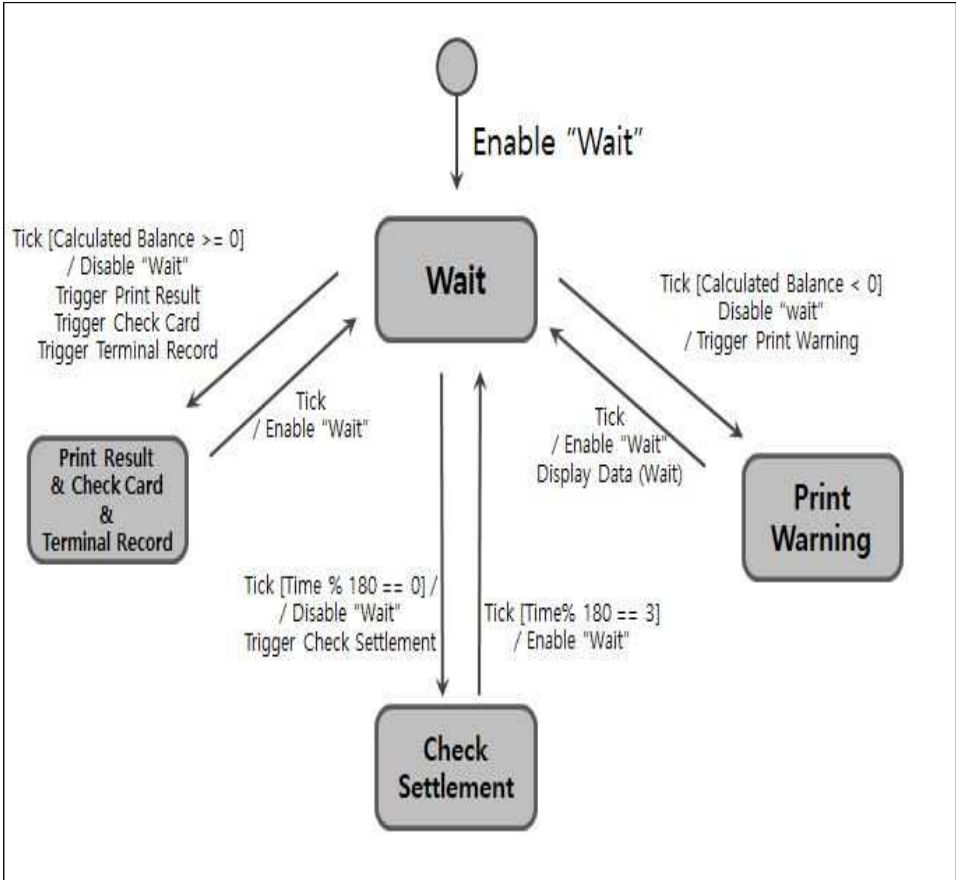
Reference No.	2.1.7
Name	Print Warning
Input	trigger
Output	Display Data
Process Description	Calculated Balance < 0 일 때 프로세스가 실행 되어 Hardware Interface 에 Display Data(Warning)를 전송한다.

Reference No.	2.1.8
Name	Check Card
Input	trigger
Output	Updated Card Data
Process Description	Calculated Balance >= 0 일 때 프로세스가 실행 되어 Card Data Interface 에 Updated Card Data 를 전송한다.

3.1.2.4.3 Data Dictionary

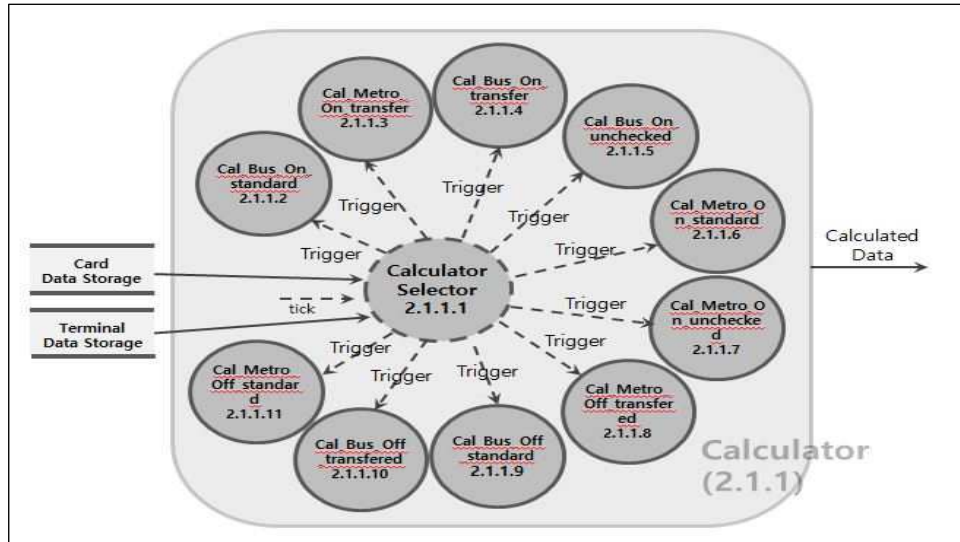
Input / Output Event	Description	Format / Type
Calculated Data	카드정보와 단말기 정보를 받아 Calculator 프로세스에서 계산한 요금	Structure
	time(Character) : 태그된시간 transportation(Character) : 교통수단 on_board(boolean) : 승하차여부 balance(int) : 잔액 fee(int) : 금액 Calculated Balance(int) : balance-fee Terminal_ID(char) : 단말기 ID count(int) : 단말기 태그 횟수	

3.1.2.4.4 State Transition Diagram (Controller 2.1.2)



3.1.2.5 DFD level 4

3.1.2.5.1 DFD



3.1.2.5.2 Process Specification

Reference No.	2.1.1.1
Name	Calculator Selector
Input	Card data, Terminal Data, tick
Output	Trigger
Process Description	Card data 와 Terminal Data 를 분석해서 작동할 Process 를 선택해서 실행시킨다.

Reference No.	2.1.1.2
Name	Cal_Bus_On_standard
Input	Trigger, Card Data, Terminal Data
Output	Calculated Data, Terminal ID
Process Description	<p>Tick[(ecd.count==0 && select == 0) date1==current_date && ecd.on_board==0 && ecd.transportation!="M" && (current_hour - _hour1) > 120] 위의 Trigger 가 조건을 충족할 경우, 즉 버스에 탑승 하는 경우중 기본요금을 계산하는 Process 로 Calculated Data 와 현재 Terminal ID 를 출력한다.</p>

Reference No.	2.1.1.3
Name	Cal_Bus_On_transfer
Input	Trigger , Card Data, Terminal Data
Output	Calculated Data, Terminal ID
Process Description	<p>Tick[date1==current_date && ecd.on_board==0 && ecd.transportation==M && (current_hour - _hour1) <= 120] 위의 Trigger 가 조건을 충족할 경우, 즉 버스에 탑승 하는 경우중 환승하는 경우를 계산하는 Proces 로 Calculated Data 와 저장되어있는 Terminal ID 를 출력한다.</p>

Reference No.	2.1.1.4
Name	Cal_Bus_On_unchecked
Input	Trigger , Card Data, Terminal Data
Output	Calculated Data, Terminal ID
Process Description	<p style="text-align: center;">Tick[date1==current_date && ecd.on_board==1 && Ecd.count>1 && Ecd.trasnpotation != etd.transportation (date!=current_date && ecd.on_board==1 && ecd.transportation == "M")]</p> <p>위의 Trigger 가 조건을 충족할 경우, 즉 버스에 탑승 하는 경우중 미정산 요금을 계산하는 Proce 로 Calculated Data 와 저장되어있는 Terminal ID 를 출력한다.</p>

Reference No.	2.1.1.5
Name	Cal_Metro_On_standard
Input	Trigger , Card Data, Terminal Data
Output	Calculated Data, Terminal ID
Process Description	<p style="text-align: center;">Tick[(ecd.count==0 && select == 1) date1==current_date && ecd.on_board==0 && ecd.transportation!="B" && (_current_hour - _hour1) > 120]</p> <p>위의 Trigger 가 조건을 충족할 경우, 즉 지하철에 탑승 하는 경우중 기본 요금을 계산하는 Proce 로 Calculated Data 와 현재 Terminal ID 를 출력한다.</p>

Reference No.	2.1.1.6
Name	Cal_Metro_On_transfer
Input	Trigger , Card Data, Terminal Data
Output	Calculated Data, Terminal ID
Process Description	Tick[date1==current_date && ecd.on_board==0 && ecd.transportation=="B" && (_current_hour - _hour1) <= 120] 위의 Trigger 가 조건을 충족할 경우, 즉 지하철에 탑승 하는 경우중 환승 요금을 계산하는 Proces 로 Calculated Data 와 저장되어있는 Terminal ID 를 출력한다.

Reference No.	2.1.1.7
Name	Cal_Metro_On_unchecked
Input	Trigger , Card Data, Terminal Data
Output	Calculated Data, Terminal ID
Process Description	<pre> Tick[date1==current_date && ecd.on_board==1 && Ecd.count>1 && Ecd.trasnpotation != etd.transportation&& (_hour1 - _hour2) <= 120 && ecd.transportation == M (date!=current_date && ecd.on_board==1 && ecd.transportation == "M")] </pre> <p>위의 Trigger 가 조건을 충족할 경우, 즉 지하철에 탑승 하는 경우중 미정산 요금을 계산하는 Proces 로 Calculated Data 와 저장되어있는 Terminal ID 를 출력한다.</p>

Reference No.	2.1.1.8
Name	Cal_Bus_Off_standard
Input	Trigger , Card Data, Terminal Data
Output	Calculated Data, Terminal ID
Process Description	<p style="text-align: center;">Tick[Date==current_Date && Ecd.on_board==1 && etd.transportation=="B"&& _hour1-_hour2>120]</p> <p>위의 Trigger 가 조건을 충족할 경우, 즉 버스에서 하차 하는 경우중 기본 요금을 계산하는 Proces 로 Calculated Data 와 현재 Terminal ID 를 출력한다.</p>

Reference No.	2.1.1.9
Name	Cal_Bus_Off_transferd
Input	Trigger , Card Data, Terminal Data
Output	Calculated Data, Terminal ID
Process Description	<p style="text-align: center;">Tick[Date==current_Date && Ecd.on_board==1 && etd.transportation=="B"&& _hour1-_hour2<=120&& Ecd.on_board_check!=1&& Date1==date2]</p> <p>위의 Trigger 가 조건을 충족할 경우, 즉 버스에서 하차 하는 경우중 환승 요금을 계산하는 Proces 로 Calculated Data 와 현재 Terminal ID 를 출력한다.</p>

Reference No.	2.1.1.10
Name	Cal_Metro_Off_standard
Input	Trigger , Card Data, Terminal Data
Output	Calculated Data, Terminal ID
Process Description	<pre> Tick[date1==current_date && Ecd.on_bard==1 && Etd.transportation=="M" && _hour1-_hour2 >120] </pre> <p>위의 Trigger 가 조건을 충족할 경우, 즉 지하철에서 하차 하는 경우중 기본 요금을 계산하는 Proces 로 Calculated Data 와 현재 Terminal ID 를 출력한다.</p>

Reference No.	2.1.1.11
Name	Cal_Metro_Off_transfer
Input	Trigger , Card Data, Terminal Data
Output	Calculated Data, Terminal ID
Process Description	<pre> Tick[date1==current_date && Ecd.on_bard==1 && Etd.transportation=="M" && _hour1-_hour2 <=120 && Ecd.on_board_check!=1 && Date1==date2] </pre> <p>위의 Trigger 가 조건을 충족할 경우, 즉 지하철에서 하차 하는 경우중 환승 요금을 계산하는 Proces 로 Calculated Data 와 현재 Terminal ID 를 출력한다.</p>

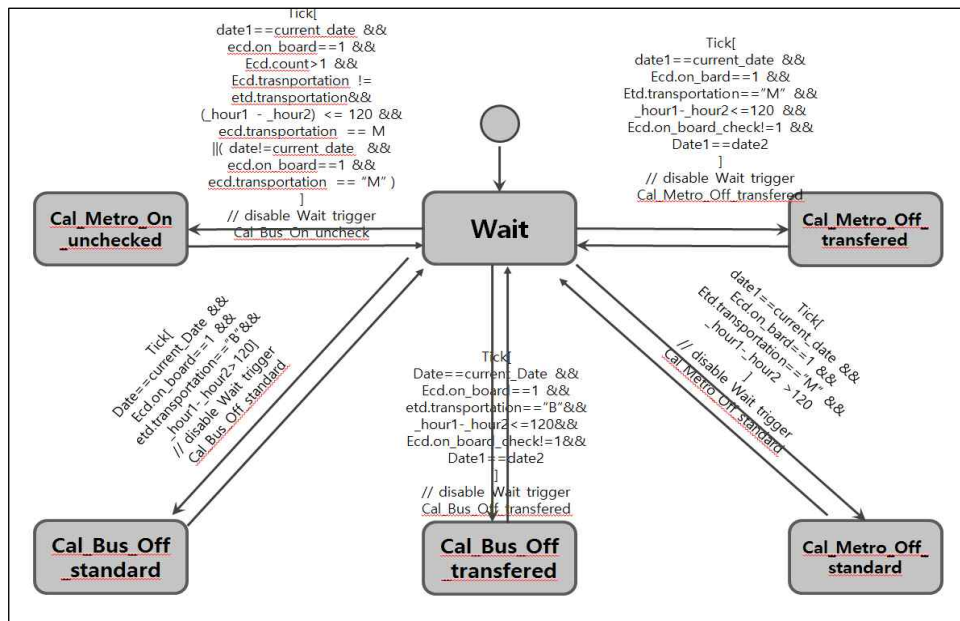
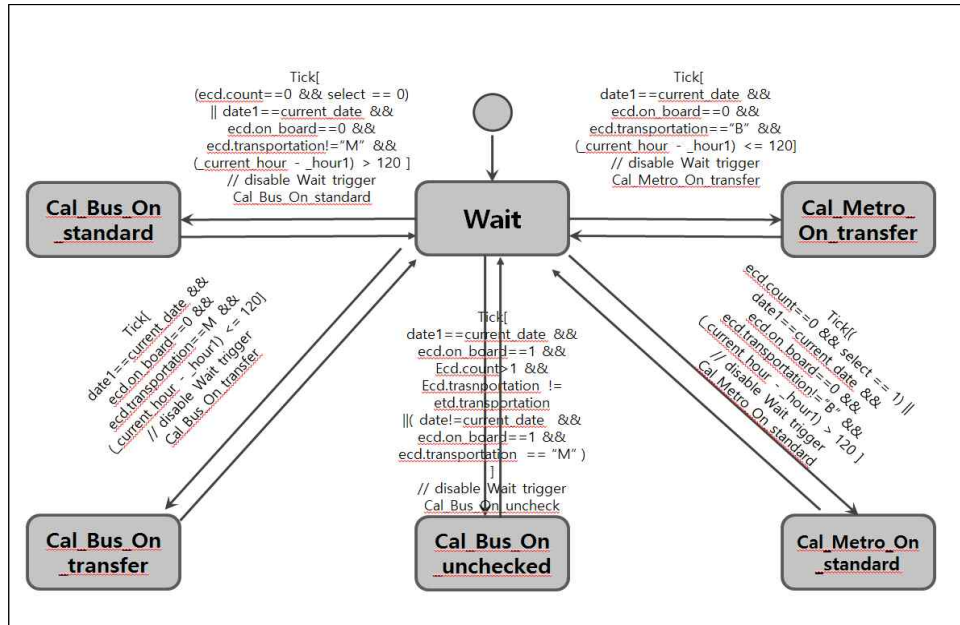
Reference No.	2.1.1.12
Name	Cal_Bus_On_Unchecked2
Input	Trigger , Card Data, Terminal Data
Output	Calculated Data, Terminal ID
Process Description	<p>Tick[Date1==current_date && Etd.transportation=="B"&& Ecd.count>1 && Ecd.transportation!=etd.transportation && _hour1-_hour2>120]</p> <p>위의 Trigger 가 조건을 충족할 경우, 즉 지하철에서 하차 하는 경우중 환승 요금을 계산하는 Proces 로 Calculated Data 와 현재 Terminal ID 를 출력한다.</p>

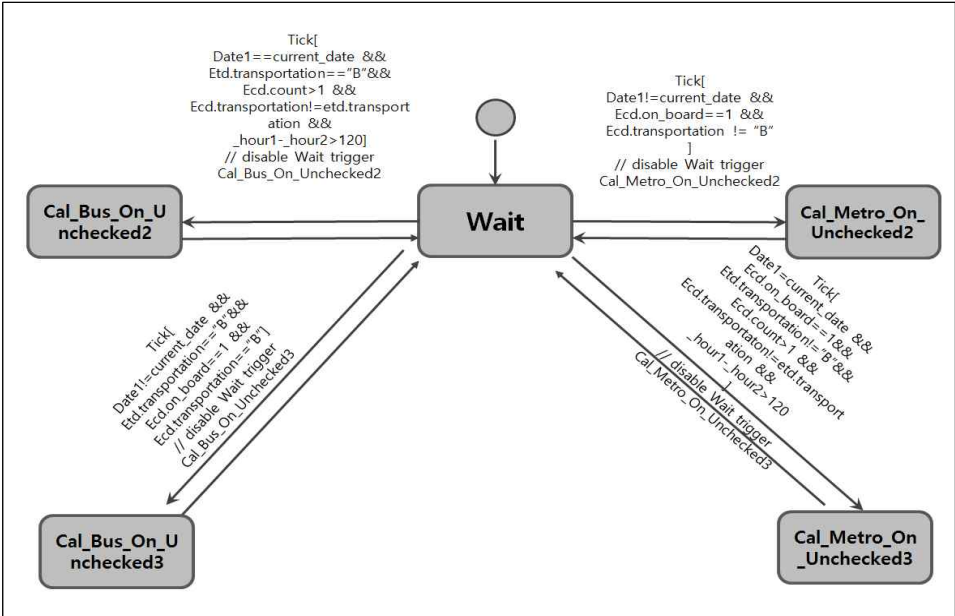
Reference No.	2.1.1.13
Name	Cal_Bus_On_Unchecked3
Input	Trigger , Card Data, Terminal Data
Output	Calculated Data, Terminal ID
Process Description	<p>Tick[Date1!=current_date && Etd.transportation=="B"&& Ecd.on_board==1 && Ecd.transportation=="B"]</p> <p>위의 Trigger 가 조건을 충족할 경우, 즉 지하철에서 하차 하는 경우중 환승 요금을 계산하는 Proces 로 Calculated Data 와 현재 Terminal ID 를 출력한다.</p>

Reference No.	2.1.1.14
Name	Cal_Metro_On_Unchecked2
Input	Trigger , Card Data, Terminal Data
Output	Calculated Data, Terminal ID
Process Description	<pre> Tick[Date1!=current_date && Ecd.on_board==1 && Ecd.transportation != "B"] </pre> <p>위의 Trigger 가 조건을 충족할 경우, 즉 지하철에서 하차 하는 경우중 환승 요금을 계산하는 Proces 로 Calculated Data 와 현재 Terminal ID 를 출력한다.</p>

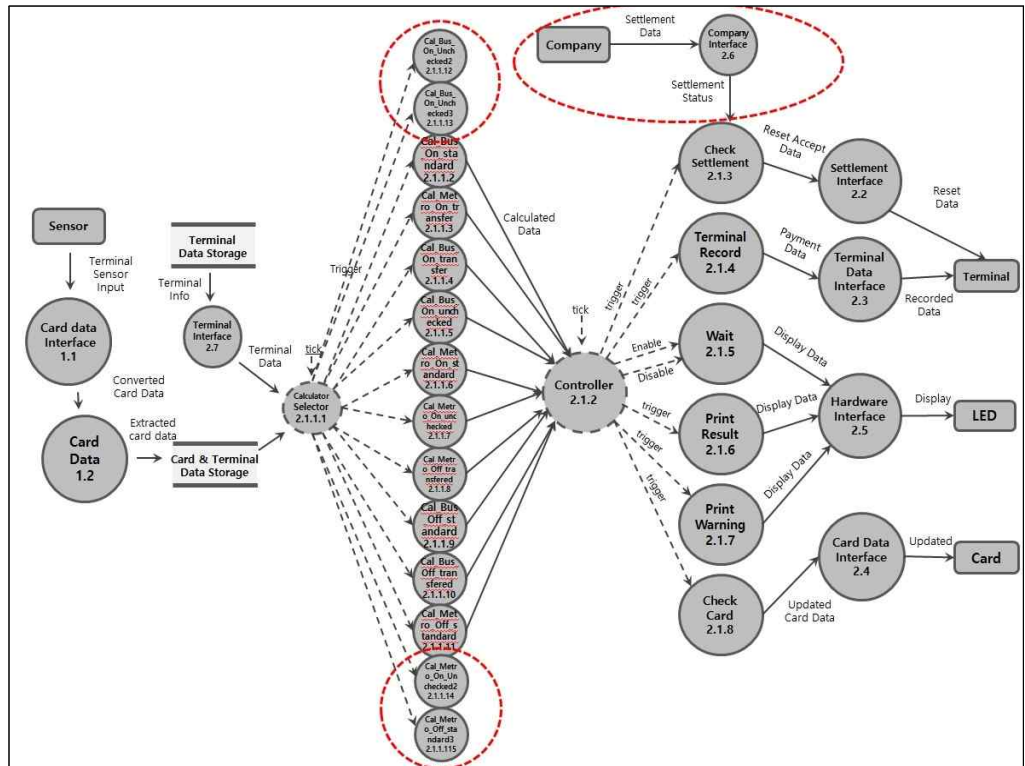
Reference No.	2.1.1.15
Name	Cal_Metro_On_Unchecked3
Input	Trigger , Card Data, Terminal Data
Output	Calculated Data, Terminal ID
Process Description	<pre> Tick[Date1=current_date && Ecd.on_board==1&& Etd.transportation!="B"&& Ecd.count>1 && Ecd.transportaton!=etd.transportation && _hour1-_hour2>120] </pre> <p>위의 Trigger 가 조건을 충족할 경우, 즉 지하철에서 하차 하는 경우중 환승 요금을 계산하는 Proces 로 Calculated Data 와 현재 Terminal ID 를 출력한다.</p>

3.1.2.5.3 State Transition Diagram (Calculator Selector 2.1.1.1)





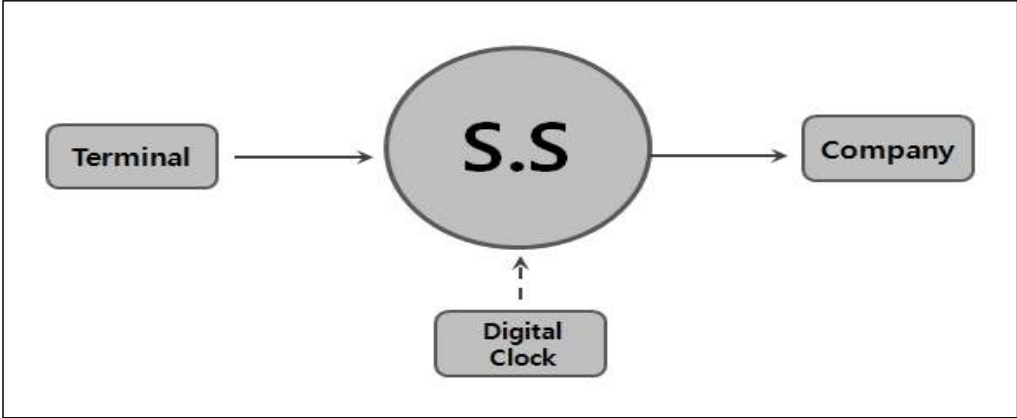
3.1.2.6 Overall DFD



3.2 Fundamental Management System

3.2.1 System Context Diagram

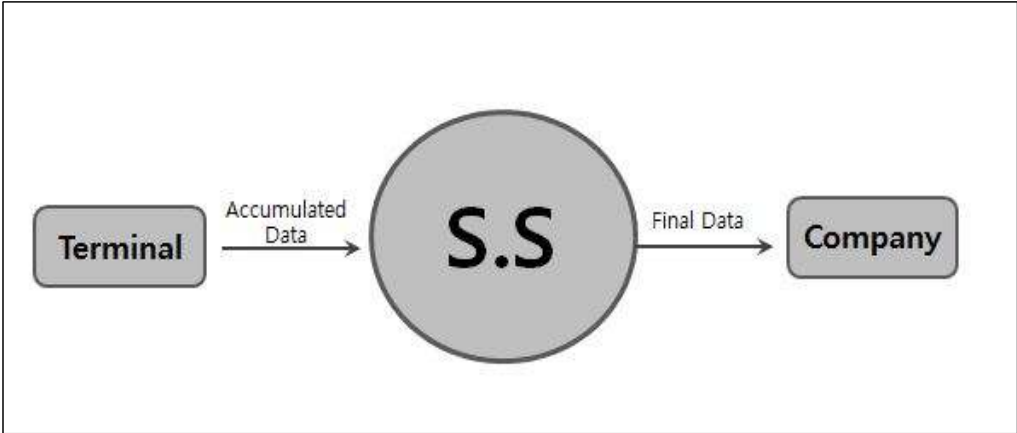
3.2.1.1 Basic System Context Diagram



3.2.1.2 Event List

Input / Output Event	Description
Accumulated Data	각각의 Terminal 에서 누적된 data
Final Bus Data	Settlement System 에서 정산된 Bus Company 지분 또는 Hold 를 결정하는 데이터
Final Metro Data	Settlement System 에서 정산된 Metro Company 지분 또는 Hold 를 결정하는 데이터

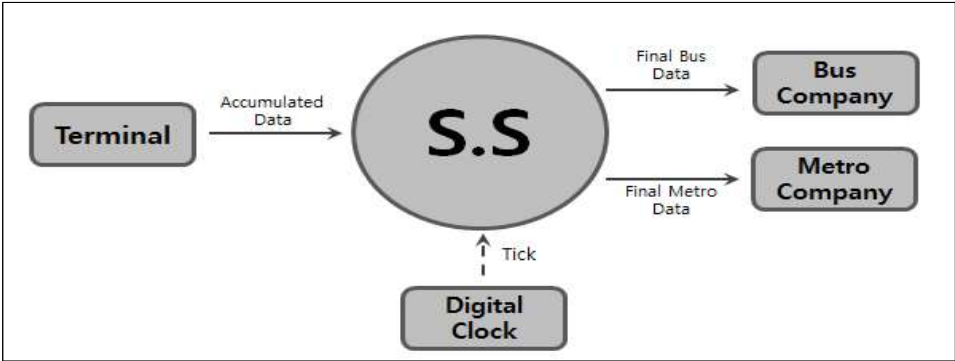
3.2.1.3 The System Context Diagram



3.2.2 Data Flow Diagram

3.2.2.1 DFD level 0

3.2.2.1.1 DFD



3.2.2.1.2 Process Specification

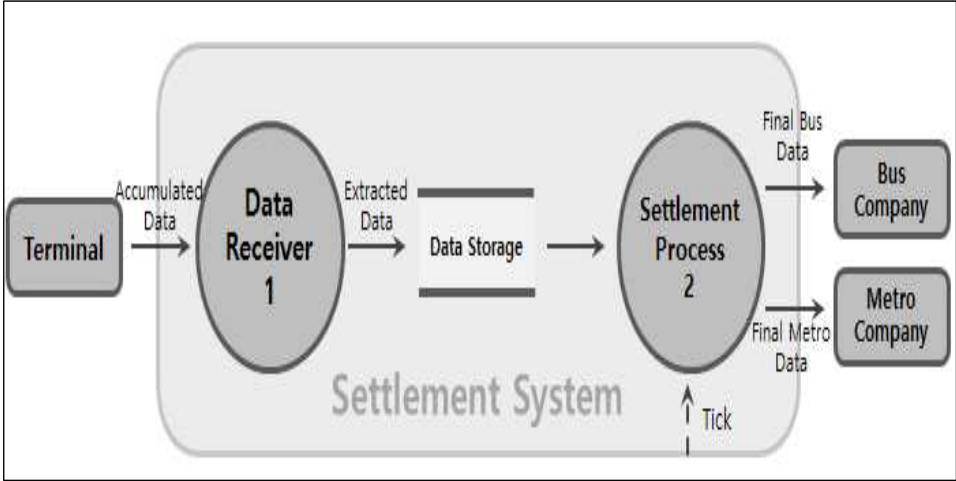
Reference No.	0
Name	Settlement System
Input	Accumulated Data
Output	Final Bus Data, Final Metro Data
Process Description	단말기로부터 받은 정보를 받아 정산 시 필요한 계산을 수행하여 버스 회사와 지하철 회사로 분배

3.2.2.1.3 Data Dictionary

Input / Output Event	Description	Format / Type
Accumulated Data	Terminal 에서 누적된 data	Periodic
Final Bus Data	Settlement System 에서 Bus Company 로 보내줄 정보를 저장한 File 또는 정산할 시간이 안 되어 있음을 알려주는 Signal Data	File / Hold
Final Metro Data	Settlement System 에서 Bus Company 로 보내줄 정보를 저장한 File 또는 정산할 시간이 안 되어 있음을 알려주는 Signal Data	File / Hold

3.2.2.2 DFD level 1

3.2.2.2.1 DFD



3.2.2.2.2 Process Specification

Reference No.	1
Name	Data Receiver
Input	Accumulated Data
Output	Extracted Data
Process Description	Terminal 에서 Accumulated Data 를 입력받아 정산에 필요한 요금에 관련된 Data 만 추출해 전달해준다.

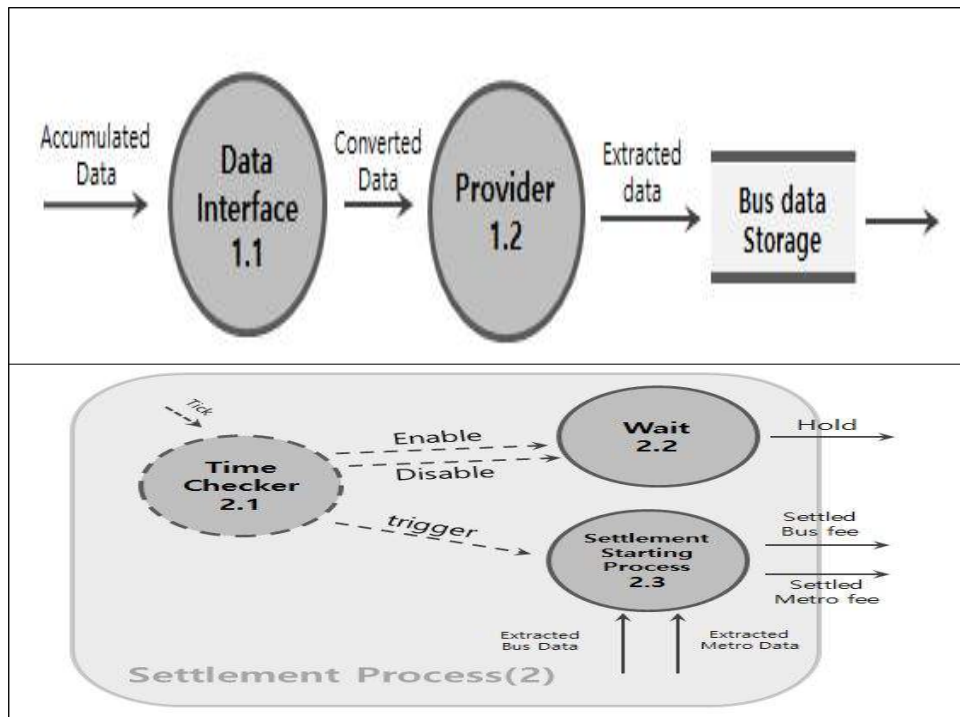
Reference No.	2
Name	Settlement Process
Input	Extracted Data, tick
Output	Final Bus Data, Final Metro Data
Process Description	Bus&Metro Data Storage 로부터 Extracted Data 만 받아서 각각 계산 후 Bus Company 에 Final Bus Data 를 Metro Company 에 Final Metro Data 를 전송 혹은 Hold Signal 을 전송한다.

3.2.2.2.3 Data Dictionary

Input / Output Event	Description	Format / Type
Extracted Data	Terminal Data 에서 Settlement Process 에서 요구하는 데이터만 추출한 누적데이터	Structure
	time_Bus(character) : Bus 태그 시간 time_Metro(character) : Metro 태그 시간 fee_Bus(int) : 버스 누적 결제 기록 fee_Metro(int) : 지하철 누적 결제 기록	

3.2.2.3 DFD level 2

3.2.2.3.1 DFD



3.2.2.3.2 Process Specification

Reference No.	1.1
Name	Data Interface
Input	Accumulated Data
Output	Converted Data
Process Description	Terminal 로부터 관련된 모든 누적 기록을 받아서 금액과 교통수단에 관한 Extracted data 만 넘겨준다.

Reference No.	1.2
Name	Provider
Input	Converted Data
Output	Extracted Data
Process Description	Data Interface 로부터 Extracted Data 를 받아서 Data Storage 로 전송한다.

Reference No.	2.1
Name	Time Checker
Input	tick
Output	Enable, Disable, trigger
Process Description	Wait Process 를 실행시키다 Time%180 ==0 이 조건이 만족하면 Wait Process 를 중지시키고 Settlement Starting Process 를 실행시킨 후 완료되면 Wait Process 를 실행시킨다.

Reference No.	2.2
Name	Wait
Input	Enable, Disable
Output	Hold
Process Description	Enable 상태 시 Hold 데이터를 Bus Company Interface 와 Metro Company Interface 로 전송한다.

Reference No.	2.3
Name	Settlement Starting Process
Input	Extracted Data, trigger
Output	Settled Bus Fee, Settled Metro Fee
Process Description	Time%180==0 조건이 만족되어 프로세스가 실행되면 Extracted Data 를 각각 이용해 Settled Bus Fee, Settled Metro Fee 를 각각 Bus Company Interface, Metro Company Interface 로 전송한다.

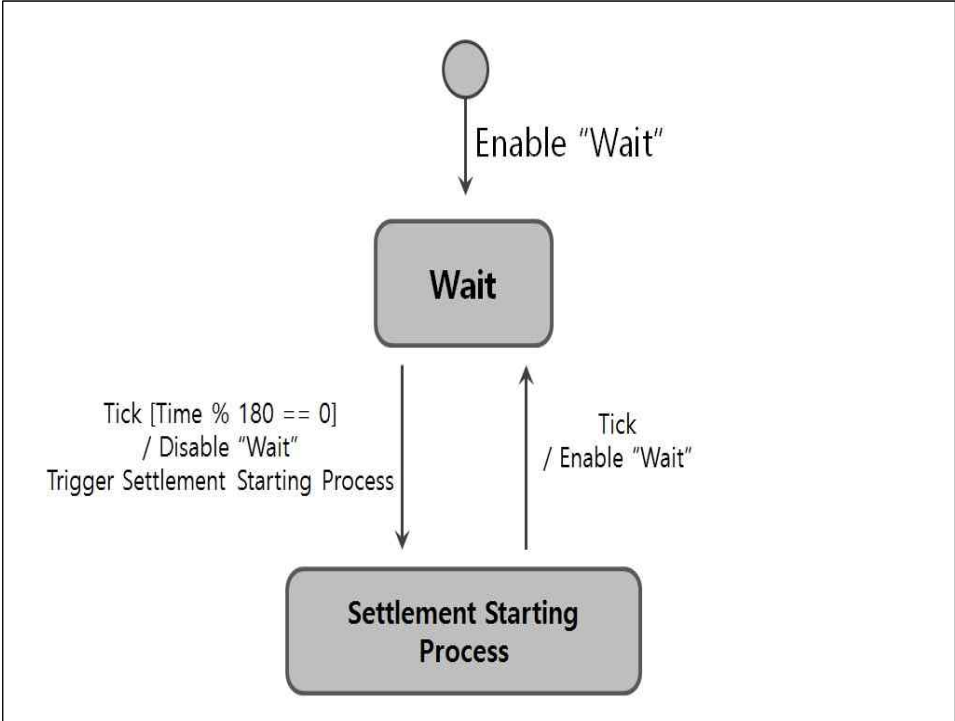
Reference No.	2.4
Name	Bus Company Interface
Input	Settled Bus Fee, Hold
Output	Final Bus Data
Process Description	Wait 과 Settlement Starting Process 에서 Output 된 정보를 받아 Bus Company 로 정산 혹은 Hold 될 데이터를 전송한다.

Reference No.	2.5
Name	Metro Company Interface
Input	Settled Metro Fee, Hold
Output	Final Metro Data
Process Description	Wait 과 Settlement Starting Process 에서 Output 된 정보를 받아 Metro Company 로 정산 혹은 Hold 될 데이터를 전송한다.

3.2.2.3.3 Data Dictionary

Converted_Data	Data Interface 에서 단말기에 누적된 데이터 중 필요한 데이터로 전환한 데이터	Structure
	time(character) : 태그 된 시간 transportation(character) : 교통수단 on_Board(boolean) : 승하차 여부 fee(int) 누적 결제 기록 Terminal_ID(character) : 단말기 고유 ID count(int) : 단말기 태그 횟수	

3.2.2.3.4 State Transition Diagram Time Checker 1.3.1



3.2.2.4 Overall DFD

