

Software Requirement Specification for Public Transportation System

Project Team

Team2

Date

2014-11-20

Team Information

201111341 김성민

201111391 진청현

201311259 권오승

201311303 이정은

Table of Contents

1 개요.....	4
1.1 목적.....	4
1.2 범위.....	4
1.3 용어정리.....	5
1.4 참고문헌.....	5
1.5 Overview.....	5
2 개발 대상 설명.....	5
2.1 개발 대상.....	5
2.2 기 능.....	5
2.3 사용자 특징.....	6
2.4 제약 및 가정 사항.....	6
3 Structured Analysis.....	8
3.1 System Context Diagram.....	8
3.1.1 Basic System Context Diagram.....	8
3.1.2 Event List.....	9
3.2 Data Flow Diagram.....	9
3.2.1 DFD level 0.....	9
3.2.1.1 DFD.....	9
3.2.1.2 Process Specification.....	9
3.2.1.2.1 Process 0.....	9
3.2.1.3 Data Dictionary.....	10
3.2.2 DFD level1.....	10
3.2.2.1 DFD.....	10
3.2.2.2 Process Specification.....	10
3.2.2.2.1 Process1.....	10
3.2.2.2.2 Process2.....	11
3.2.2.3 Data Dictionary.....	11
3.2.3 DFD level2.....	11
3.2.3.1 DFD.....	11
3.2.3.2 Process Specification.....	11
3.2.3.2.1 Process2.1.....	11
3.2.3.2.2 Process 2.2.....	12
3.2.3.3 Data Dictionary.....	12
3.2.4 DFD level 3.....	12
3.2.4.1 DFD for Calculator2.1	12

3.2.4.2 DFD for Calculated Data Control 2.2.....13

3.2.4.3 Process Specification.....13

3.2.4.3.1 Process 2.1.1.....13

3.2.4.3.2 Process 2.1.2.....14

3.2.4.3.3 Process 2.1.3.....14

3.2.4.3.4 Process 2.1.4.....14

3.2.4.3.5 Process 2.1.5.....14

3.2.4.3.6 Process 2.1.6.....15

3.2.4.3.7 Process 2.1.7.....15

3.2.4.3.9 Process 2.2.1.....15

3.2.4.3.10 Process 2.2.2.....15

3.2.4.3.11 Process 2.2.3.....16

3.2.4.3.12 Process 2.2.4.....16

3.2.4.3.13 Process 2.2.5.....16

3.2.4.3.14 Process 2.2.6.....16

3.2.4.3.15 Process 2.2.7.....16

3.2.4.4 State Transition Diagram for In/Out Check Controller2.1.1.....17

3.2.4.5 State Transition Diagram for Calculated Data Controller2.2.1.....17

3.3 System Context Diagram - Account System.....18

3.3.1 Basic System Context Diagram.....18

3.3.2 Event List.....18

3.3.3 DFD level 0.....18

3.3.3.1 DFD.....18

3.3.3.2 Process Specificataion.....19

3.3.3.2.1 Process 0.....19

3.3.3.2.2 Data dictionary.....19

3.3.4 DFD level 1.....19

3.3.4.1 DFD.....19

3.3.4.2 Process Specificataion.....20

3.3.4.2.1 Process 1.....20

3.3.4.2.2 Process 2.....20

3.3.5 DFD level 2.....20

3.3.5.1 DFD.....20

3.3.5.2 Process Specification.....21

3.3.5.2.1 Process 2.1.....21

3.3.5.2.1 Process 2.2.....21

3.3.5.2.1 Process 2.3.....21

3.3.5.2.1 Process 2.4.....21

3.3.5.2.1 Process 2.5.....21

4 Overall.....22

1 개요

1.1 목적

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

1.2 범위

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

모든 시스템은 SW만으로 구현한다. HW가 필요한 부분은 SW모듈을 만들어 가상의 HW를 구현한다.



그림 1 서울의 교통카드 운영시스템Ver. DS-2014SE-PTS-SRS-1.0

1.3 용어 정리

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 참고 문헌

(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”, 수도권교통본부, 2009

1.5 Overview

2장 개발 대상에 대한 설명: 3장 세부 기능 명세

2 개발 대상 설명

2.1 개발 대상

SW로 개발된 PTS는 총 3가지로 구성된다. 버스용 단말기와 지하철용 단말기, 전산 시스템 이다.

2.2 기능

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

지하철용 단말기는 역에 부착돼 탑승 태그와 하차 태그가 가능하다. 1개역 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.

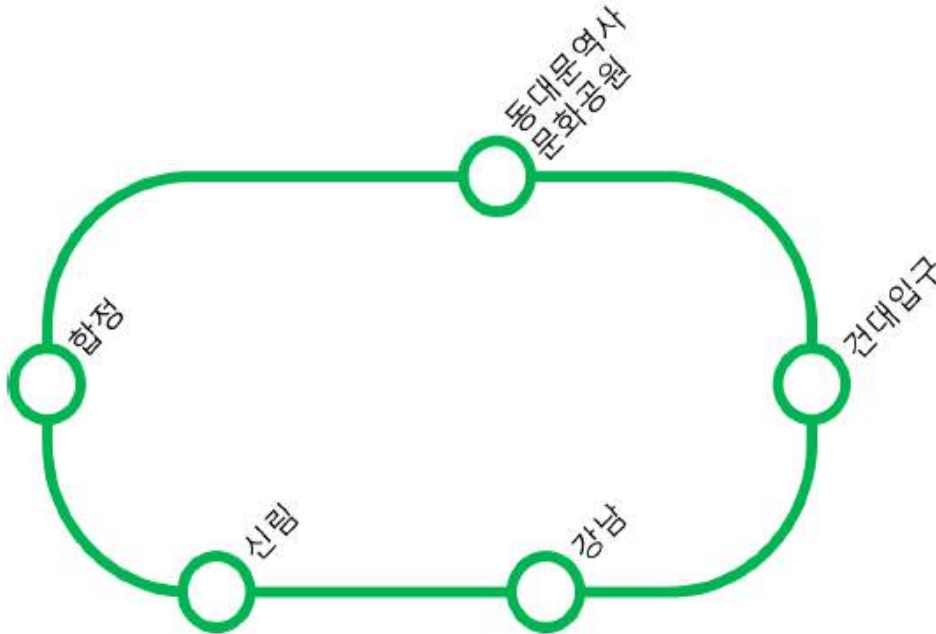
정산은 하루에 한 번 이뤄진다. 버스과 지하철의 기록을 분석하고 버스와 지하철에 각각 수익을 배분한다.

2.3 사용자 특징

사용자는 대중교통 승차 시 카드를 태그한다.
사용자는 하차 시 카드를 태그 할 수도 있고 아닐 수도 있다.

2.4 제약 및 가정 사항

버스 승차는 지역에 상관 없이 할 수 있다.
지하철은 2호선 역 중 5개만 고려한다: 건대입구, 동대문역사문화공원, 합정, 신림, 강남.



하루는 3분으로 가정한다.
다음 날 운행이 시작하기 전 까지는 정산이 반드시 이루어 져야 한다.
정산 후 모든 프로그램의 정보는 초기화 된다. (사용자 카드 정보 제외)
환승이 가능한 시간은 15초 이내다.
버스 환승 시 추가요금의 기준인 단위는 30초이다.
버스와 지하철의 기본료는 1050원이다.
교통카드는 저장된 텍스트 파일로 가정하고, 교통카드 태그 행위를 해당 파일을 입력 하는 것으로 가정한다.
잔액이 모자를 경우 태우지 않는다.
버스 환승 최고 부담금액인 700원이 남아 있지 않으면 버스로 환승시키지 않는다.
지하철 환승 최고 부담금액인 600원이 남아 있지 않으면 지하철로 환승시키지 않는다.
하루(3분) 종료 시 탑승되어 있는 승객은 미정산으로 처리한다. (미정산 금액: 표1,2 참조) 정산시 소수점 이하는 반올림 한다.(표 5참조)

표 2 지하철 요금 규정

기본 요금	1050원
거리 비례 추가 요금	2 정거장 이상시 200원 추가
환승 요금	1 정거정당 300원 추가 (최고 600원을 넘지 않음)
미정산 요금	일반 하차 시 단말기를 태그하지 않았을 경우 200원을 다음 승차 시 부과 한다.
	버스에서 지하철 환승 후 하차 시 단말기를 태그하지 않았을 경우 환승 최대 요금인 600원을 부과한다.

표 1 버스 요금 규정

기본 요금	1050원
환승 요금	30초당 100원 추가 (최고 700원을 넘지 않는다.)
미정산 요금	지하철에서 버스 환승 후 하차 시 단말기를 태그하지 않았을 경우 환승 최고 요금인 700원을 다음 승차 시 부과 한다.

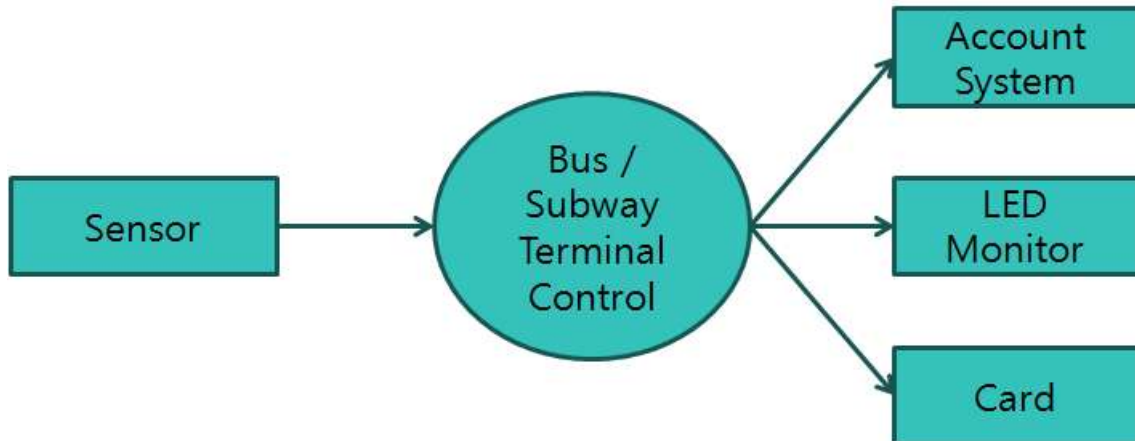
표 5 정산 방법

<p>공식: 버스→지하철 환승 버스 = 총금액 * 버스 요금 / (버스 요금 + 총금액) 지하철 = 총금액 * 총금액 / (버스 요금 + 총금액) 공식: 지하철→버스 환승 버스 = 총금액 * 총금액 / (지하철 요금 + 총금액) 지하철 = 총금액 * 지하철 요금 / (지하철 요금 + 총금액)</p>
<p>Ex) 버스 금액 : 1050 지하철 환승 금액 : 600 정산 금액 버스 : $1650 * 1050 / (1050 + 1650) = 642$ (반올림) 지하철 : $1650 * 1650 / (1050 + 1650) = 1008$ (반올림)</p>
<p>미 정산 금액의 경우 버스를 타고 지하철로 환승 후 단말기를 태그하지 않고 내렸을 경우 총금액을 지하철 환승 최고 가격인 600으로 하여 계산한다. 지하철을 타고 버스로 환승 후 단말기를 태그하지 않고 내렸을 경우 총금액을 버스 환승 최고 가격인 700으로 하여 계산한다. 추후 사용자가 버스나 지하철을 다시 탑승할 경우 사용자 카드에 해당 미정산 금액을 차감하고, 초기 규정 요금을 부과한다.</p>

3. Structured Analysis

3.1 System Context Diagram

3.1.1 Basic System Context Diagram



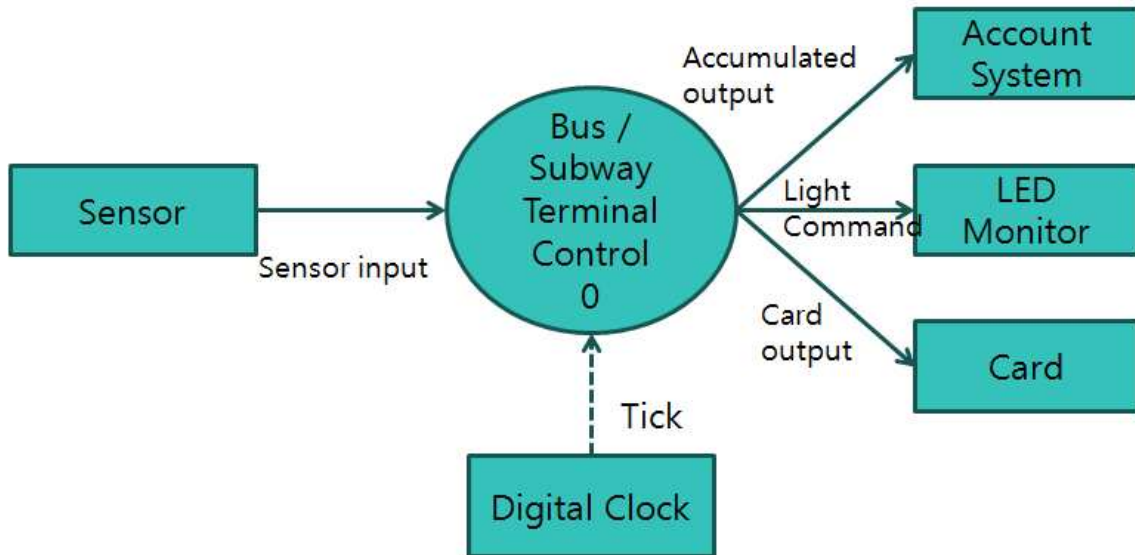
3.1.2 Event List

Input/Output Event	Description
Sensor Input	카드가 태그 되었을 때, 카드의 정보를 컨트롤러로 전달해준다.
Accumulated Output	계산된 카드정보를 정산 시스템으로 전달해준다.
Light	LED Monitor로 어떤 화면을 보여줄 것 인가에 대한 명령을 전달해준다.
Card Output	탑승이 되고나서 잔액 등 바뀐 카드 정보를 다시 카드에 넣어준다.

3.2 Data Flow Diagram

3.2.1 DFD level 0

3.2.1.1 DFD



3.2.1.2 Process Specification

3.2.1.2.1 Process 0

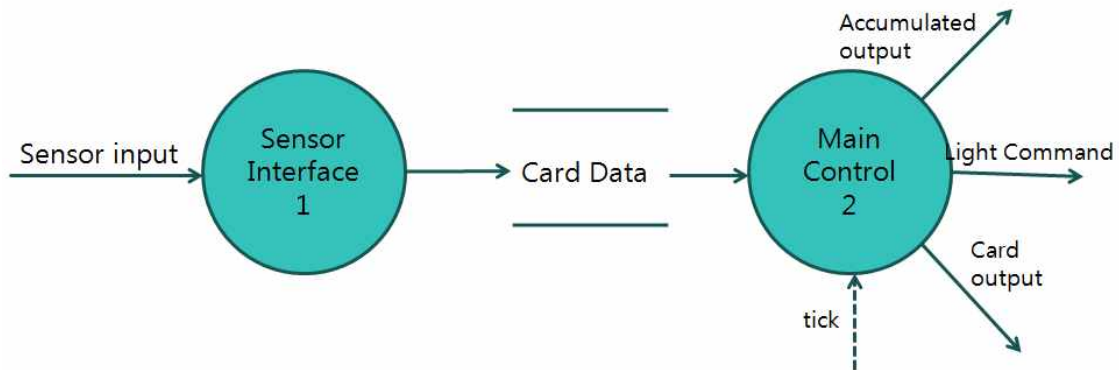
Reference No.	0
Name	Bus / Subway Terminal Control
Input	Sensor Input , Tick
Output	Accumulated Output, Light, Card Output
Process Description	단말기 시스템에서 제일 중요한 부분이고 카드정보와 시간정보를 받아서 탑승여부와 정산, LED Monitor화면 출력과 접촉된 카드의 정보를 다시 갱신 해준다.

3.2.1.3 Data Dictionary

Input/Output Event	Description	Format/Type
Sensor Input	카드가 태그 되었을 때, 카드의 정보를 컨트롤러로 전달해준다.	Ture/False,Interrupt
Accumulated Output	계산된 카드정보를 정산 시스템으로 전달해준다.	Stirng, double
Light	LED Monitor로 어떤 화면을 보여줄 것인가에 대한 명령을 전달해준다.	True/False,Interrupt
Card Output	탑승이 되고나서 잔액 등 바뀐 카드 정보를 다시 카드에 넣어준다.	String, double

3.2.2 DFD Level 1

3.2.2.1 DFD



3.2.2.2 Process Specification

3.2.2.2.1 Process 1

Reference No.	1
Name	Sensor Interface
Input	Sensor Input
Output	Card Data , Sensor input signal
Process Description	카드가 태그 되었을 때 카드정보를 메인 컨트롤로 전달 해준다.

3.2.2.2.2 Process 2

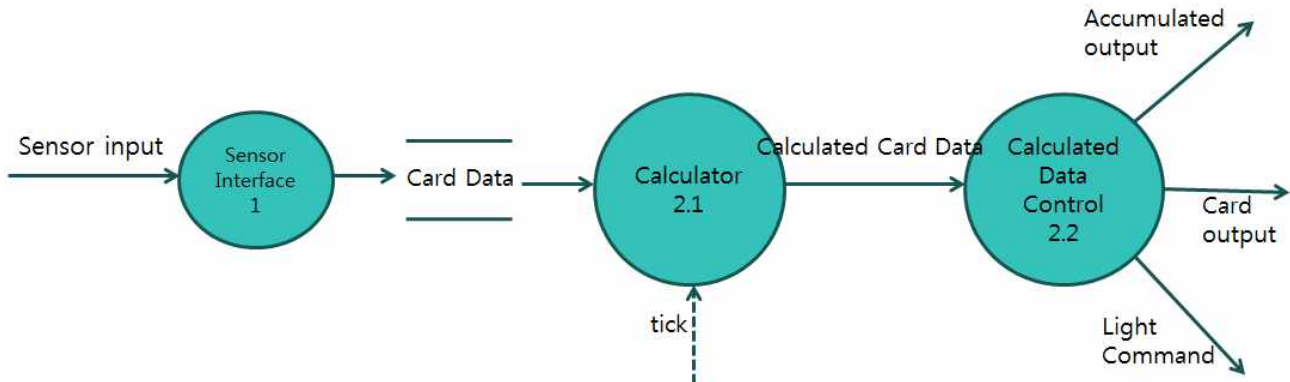
Reference No.	2
Name	Main Control
Input	Card Data, tick
Output	Accumulated Output, Light, Card Output
Process Description	모든 과정에 대한 정보를 갖고 있는 컨트롤러이고, 데이터 저장 소로부터 데이터를 받아서 정산 시스템에는 정산을 하는 정보, LED Monitor에는 어떤 화면을 출력 할지에 대한 명령, 카드에는 새로 갱신 될 정보를 보내준다.

3.2.2.3 Data Dictionary

Data Name	Description
Card Data	카드데이터는 시간, 단말기, In/Out, 잔액에 대한 정보를 갖고 있다.

3.2.3 DFD level 2

3.2.3.1 DFD



3.2.3.2 Process Specification

3.2.3.2.1 Process 2.1

Reference No.	2.1
Name	Calculator
Input	Card Data, Tick
Output	Calculated Card Data, Time data
Process Description	카드 정보와 시간을 이용해서 요금과 탑승여부를 결정해준다. 탑승이 허가 된다면 새로운 정보를 카드에 넣어준다.

3.2.3.2.2 Process 2.2

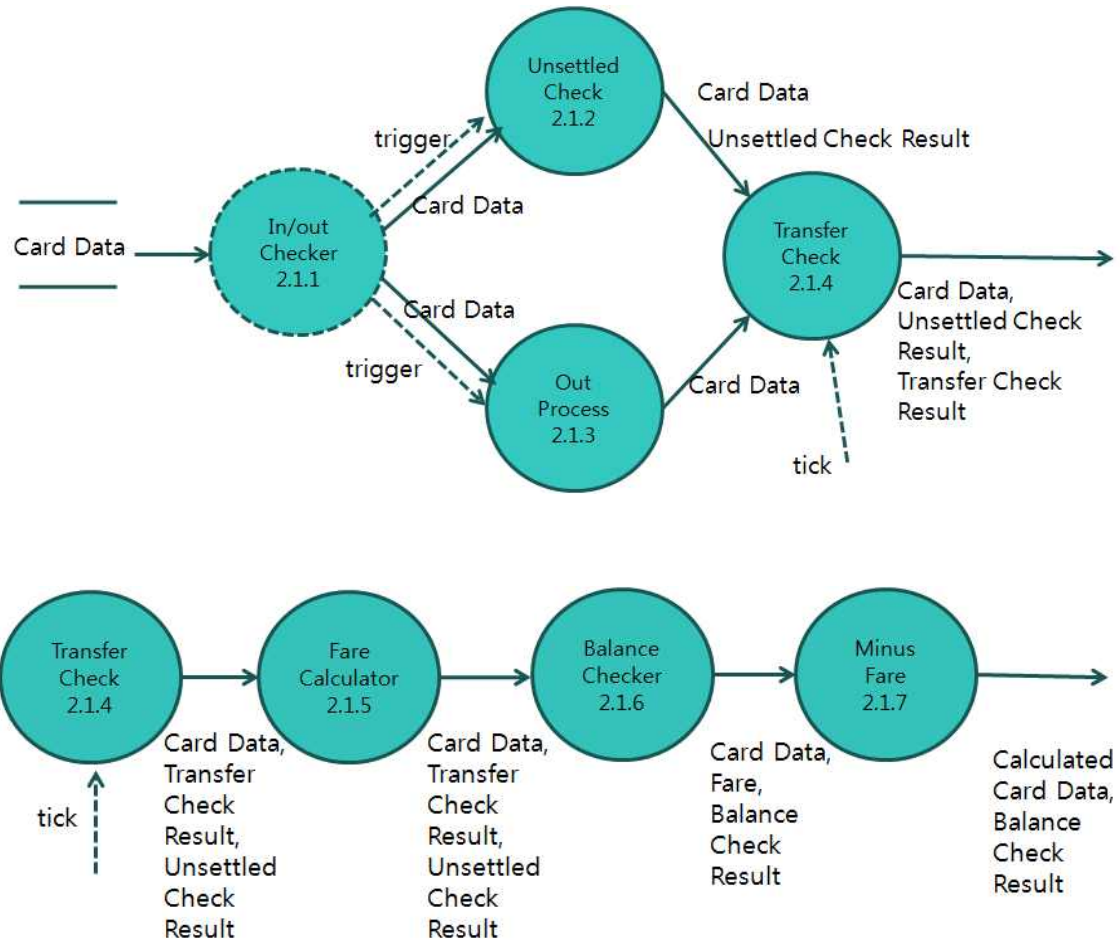
Reference No.	2.2
Name	Calculated Data Control
Input	Calculated Card Data, Time Data, Light command
Output	Accumulated Output, Card Output
Process Description	누적 된 정보를 정산시스템으로 보내주고 새로 계산된 카드정보를 카드로 보내준다. 또 LED Monitor 화면에 정보를 출력하도록 신호를 보내준다.

3.2.2.3 Data Dictionary

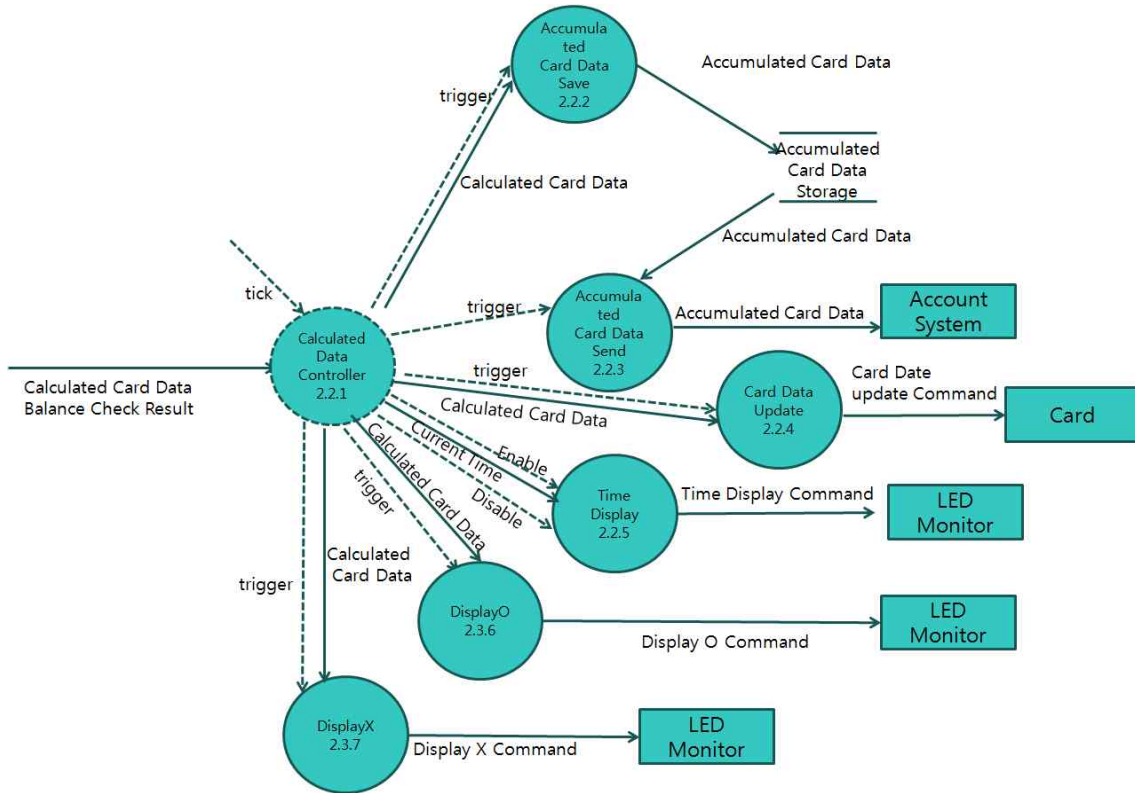
Data Name	Description
Calculated Card Data	원래의 시간, 단말기, In/Out, 잔액에 대한 카드정보가 새로 계산된 정보를 의미한다.

3.2.4 DFD level 3

3.2.4.1 DFD for Calculator2.1



3.2.4.2 DFD for Calculated Data Control 2.2



3.2.4.3 Process Specification

3.2.4.3.1 Process 2.1.1

Reference No.	2.1.1
Name	In/Out Check Controller
Input	Card Data
Output	Card Data, Trigger
Process Description	Card Data를 받아서 내가 지금 버스나 지하철에 IN하고 있는지 OUT을 하고 있는지를 판단해서 만약 IN을 하고 있다면 Unsettled Check쪽으로 보내주고 OUT을 하고 있다면 OUT Process쪽으로 보내준다.

3.2.4.3.2 Process 2.1.2

Reference No.	2.1.2
Name	Unsettled Check
Input	Card Data, Trigger
Output	Card Data, Unsettled Check Result
Process Description	In일 상태의 Calculated Card Data를 받아서 탑승자가 미정산 요금을 갖고 있는지 판단해서 데이터를 수정해서 다음으로 데이터를 넘겨준다.

3.2.4.3.3 Process 2.1.3

Reference No.	2.1.3
Name	Out Process
Input	Card Data, Trigger
Output	Card Data
Process Description	Out인 상태의 Card Data를 받아서 Transfer Check로 Card Data를 넘겨준다.

3.2.4.3.4 Process 2.1.4

Reference No.	2.1.4
Name	Transfer Check
Input	Card Data, Time Data, Unsettled Check Result
Output	Card Data, Time Data, Unsettled Check Result, Transfer Check Result
Process Description	In 일 때는 Calculated Card Data에 있는 Time Data를 이용해서 카드를 찍고 내렸을 때 들어있는 시간과 현재 시간을 비교해서 환승이 가능한지 불가능한지를 판단해준다. Out 일 때는 지금 내리는 것 전에 환승을 했는지 안했는지를 판단해서 추가요금이 어떻게 결정될지를 정해준다.

3.2.4.3.5 Process 2.1.5

Reference No.	2.1.5
Name	Fare Calculator
Input	Card Data, Time Data, Unsettled Check Result, Transfer Check Result
Output	Card Data, Time Data, Transfer Check Result, Unsettled Check Result, Fare
Process Description	여태까지 변경 돼서온 Calculated Card Data와 Time Data를 이용해서 버스/지하철 탑승요금이나 하차 시 추가요금을 계산해준다.

3.2.4.3.6 Process 2.1.6

Reference No.	2.1.6
Name	Balance Check
Input	Card Data, Time Data, Transfer Check Result, Unsettled Check Result, Fare
Output	Card Data, Time Data, Balance Check Result
Process Description	Calculated Card Data속에 들어있는 요금 정보와 카드의 잔액정보를 비교해서 잔액이 많으면 True값 적으면 False 값을 넘겨준다.

3.2.4.3.7 Process 2.1.7

Reference No.	2.1.7
Name	Minus Fare
Input	Card Data, Time Data, Balance Check Result
Output	Calculated Card Data, Time Data, Balance Check Result
Process Description	Calculated Card Data 속에 True정보가 들어있으면 잔액에서 요금을 빼주고 False 값이 들어있으면 잔액을 바꾸지 않는다.

3.2.4.3.8 Process 2.2.1

Reference No.	2.2.1
Name	Calculated Data Controller
Input	Calculated Card Data, Time Data, Balance Check Result
Output	Calculated Card Data, Time Data, Trigger, Enable, Disable
Process Description	계산된 카드 정보와 시간정보 그리고 잔액확인결과를 확인하여 누적정보를 저장하거나 누적정보를 보내고 LED Monitor 화면에 카드정보를 출력하고 카드의 정보를 업데이트 한다.

3.2.4.3.9 Process 2.2.2

Reference No.	2.2.2
Name	Calculated Card Data Save
Input	Calculated Card Data, Trigger
Output	Calculated Card Data
Process Description	잔액확인결과가 true일 때 계산된 카드 정보를 누적하여 accumulated card data storage에 저장한다.

3.2.4.3.10 Process 2.2.3

Reference No.	2.2.3
Name	Accumulated Card Data Send
Input	Calculated Card Data, Trigger
Output	Accumulated Card Data
Process Description	현재 시간이 정산시간이 되면 잔액확인여부와 상관없이 지금까지 저장했던 accumulated card data storage의 데이터를 가지고 정산기로 accumulated data를 보낸다.

3.2.4.3.11 Process 2.2.4

Reference No.	2.2.4
Name	Card Data Update
Input	Calculated Card Data, Trigger
Output	Card Data Update Command
Process Description	잔액확인여부가 true일 때 계산된 카드 정보를 받아와 카드의 정보를 업데이트 하라는 명령을 보낸다.

3.2.4.3.12 Process 2.2.5

Reference No.	2.2.5
Name	Time Display
Input	Time Data, Enable, Disable
Output	Light Command
Process Description	현재 시간 정보를 받아서 LED Monitor에 시간을 계속 출력한다.

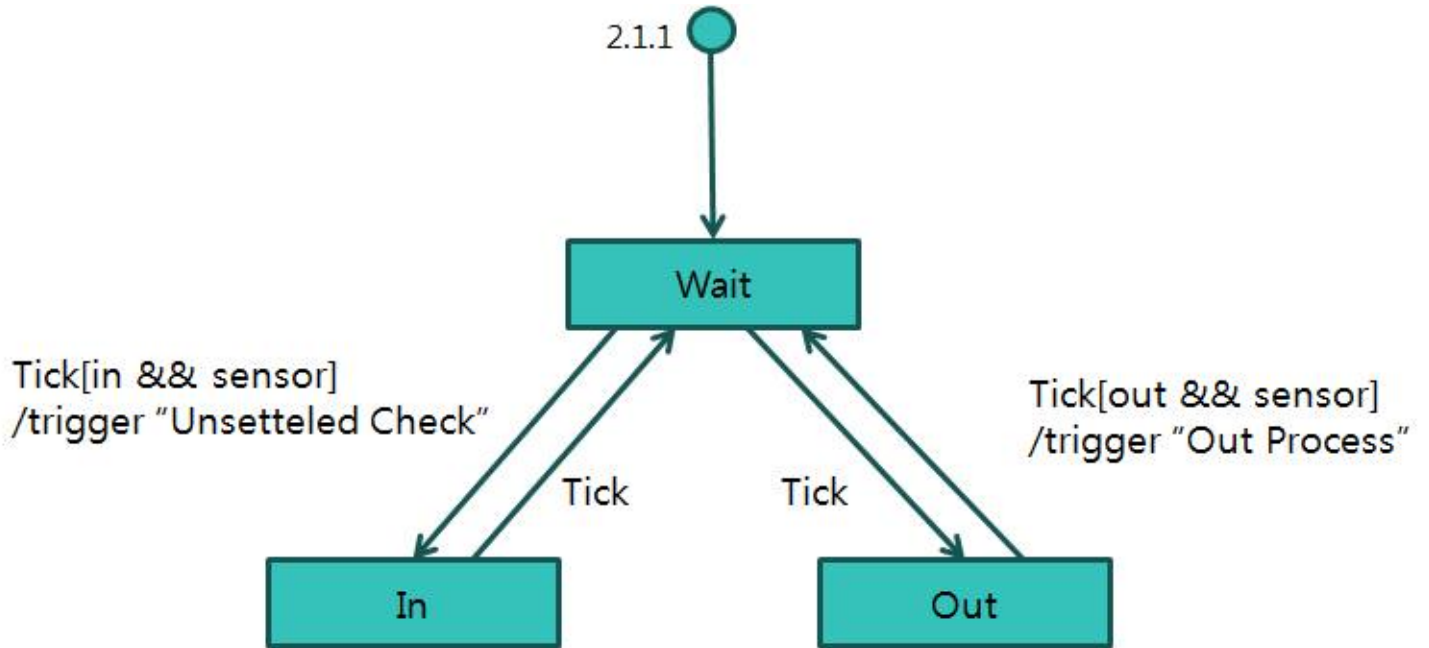
3.2.4.3.13 Process 2.2.6

Reference No.	2.2.6
Name	Display O
Input	Calculated Card Data, Trigger
Output	Display O Command
Process Description	잔액확인여부가 true일 때 계산된 카드 정보를 가지고 LED Monitor에 카드 정보를 출력할 수 있도록 신호를 보낸다.

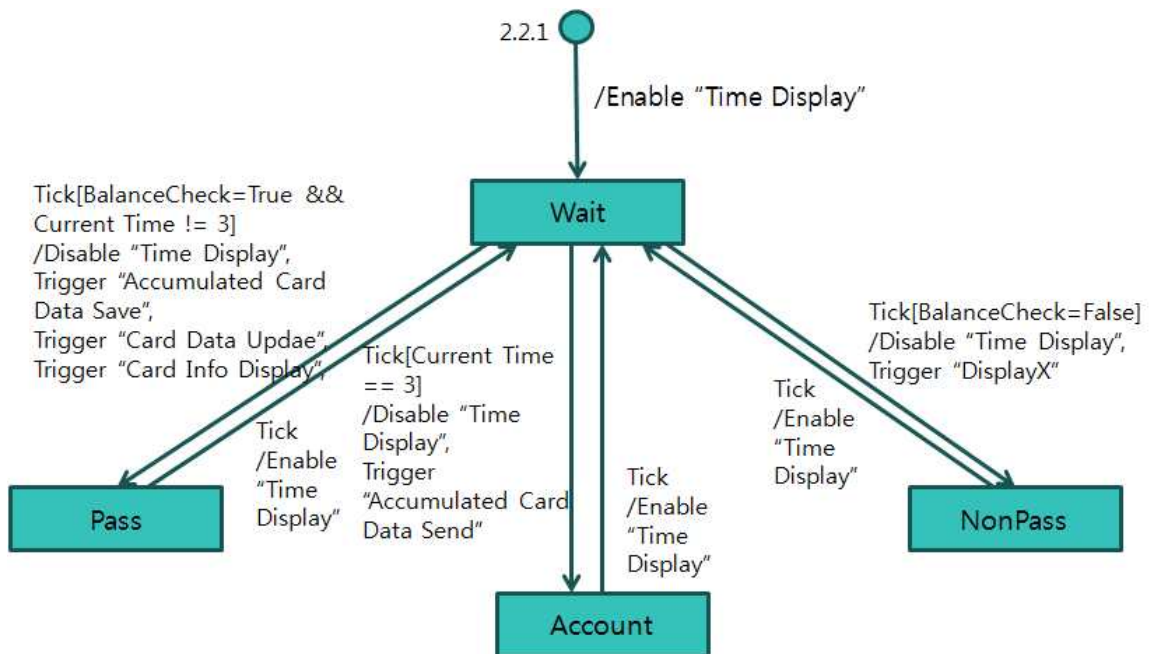
3.2.4.3.14 Process 2.2.7

Reference No.	2.2.7
Name	DisplayX
Input	Trigger, Calculated Card Data
Output	Display X Command
Process Description	잔액확인여부가 false일 때 탑승이 가능하지 않으므로 LED Monitor에 X표가 뜨도록 신호를 보낸다.

3.2.4.4 State Transition Diagram for In/Out Check Controller2.1.1

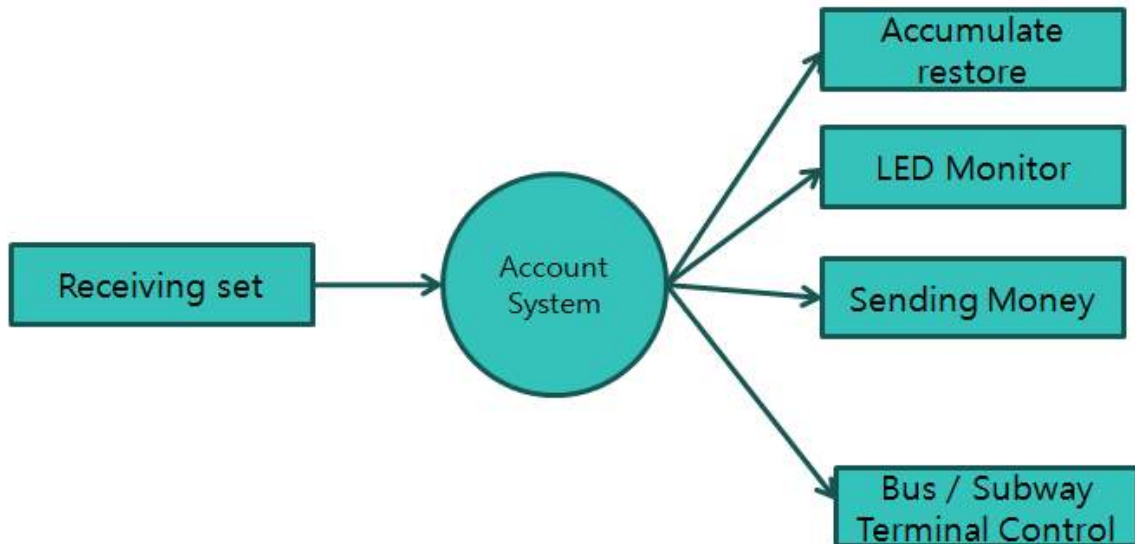


3.2.4.5 State Transition Diagram for Calculated Data Controller2.2.1



3.3 System Context Diagram - Account System

3.3.1 Basic System Context Diagram

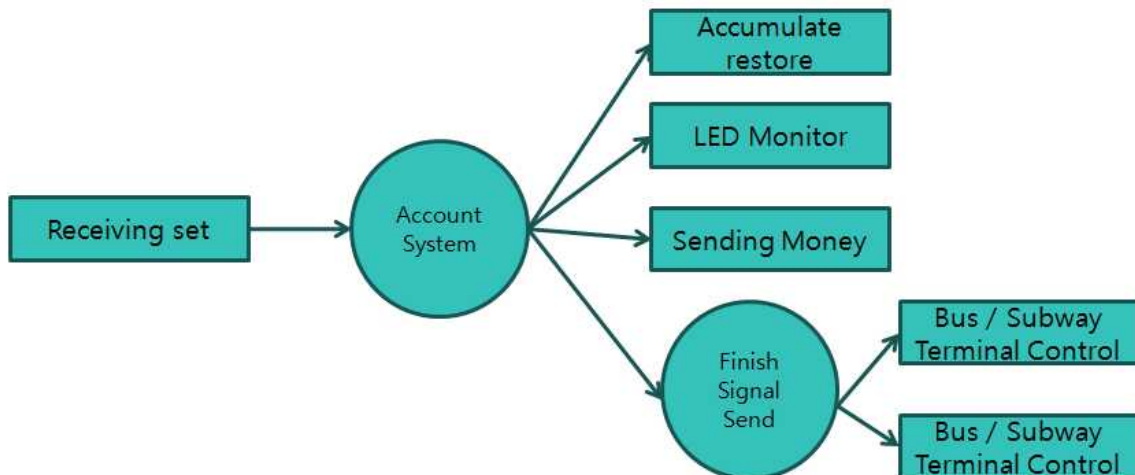


3.3.2 Event List

Input/Output Event	Description
Accumulated Input	하루 동안 단말기에 누적된 정보
Account Trigger	정산을 시작하라는 신호를 받는다.
Account Data	Accumulated Data를 이용해 정산한 정보이다.
LED Trigger	정산 결과와 정산이 완료됐다는 것을 보여주게 하는 신호이다.
Finish Signal	정산이 완료되었으니 버스와 지하철 단말기에 다시 운영을 시작하라는 정보를 보낸다.

3.3.3 DFD level 0

3.3.3.1 DFD



3.3.3.2 Process Specification

3.3.3.2.1 Process 0

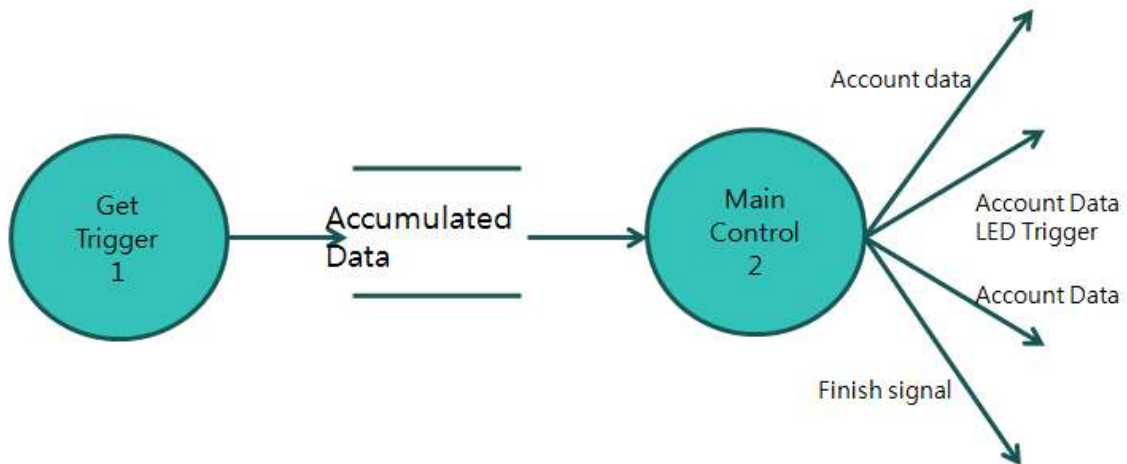
Reference No.	0
Name	Account System
Input	Accumulated Data , Account Trigger, Account Date
Output	Account Data, Finish Signal, LED Trigger
Process Description	메인 정산기 시스템이다. 정산을 시작하라는 신호를 받아서 여태 까지 누적된 정보를 갖고 지하철과 버스 단말기에 정산 중에는 정지 신호를 보내주고 정산이 완료되면 다시 정산을 하라는 신호를 보내주고 LED Monitor에 정산 결과를 출력해준다. 그리고 날짜별로 정산 정보가 어떻게 됐는지 저장을 해주고 각 버스 지하철 회사에 요금을 분배해준다.

3.3.3.2.2 Data Dictionary

Input/Output Event	Description	Format/Type
Accumulated Data	카드가 태그되면 정산에 쓰일 데이터가 정산기 시스템으로 들어온다.	Stirng, double
Account Data	누적된 정보를 갖고 Account System을 갖고 정산을 완료한 Data이다.	Stirng, double
Finish Signal	정산이 완료된 후 각 단말기에 정산이 정상적으로 끝났으니 다시 작동을 시작 하 게하는 정보를 보내준다.	True/False, Interrupt
LED Trigger	LED Monitor에 정산이 완료 된 정보를 출력하도록 Command를 보내준다.	True/False, Interrupt

3.3.4 DFD level 1

3.3.4.1 DFD



3.3.4.2 Process Specification

3.3.4.2.1 Process 1

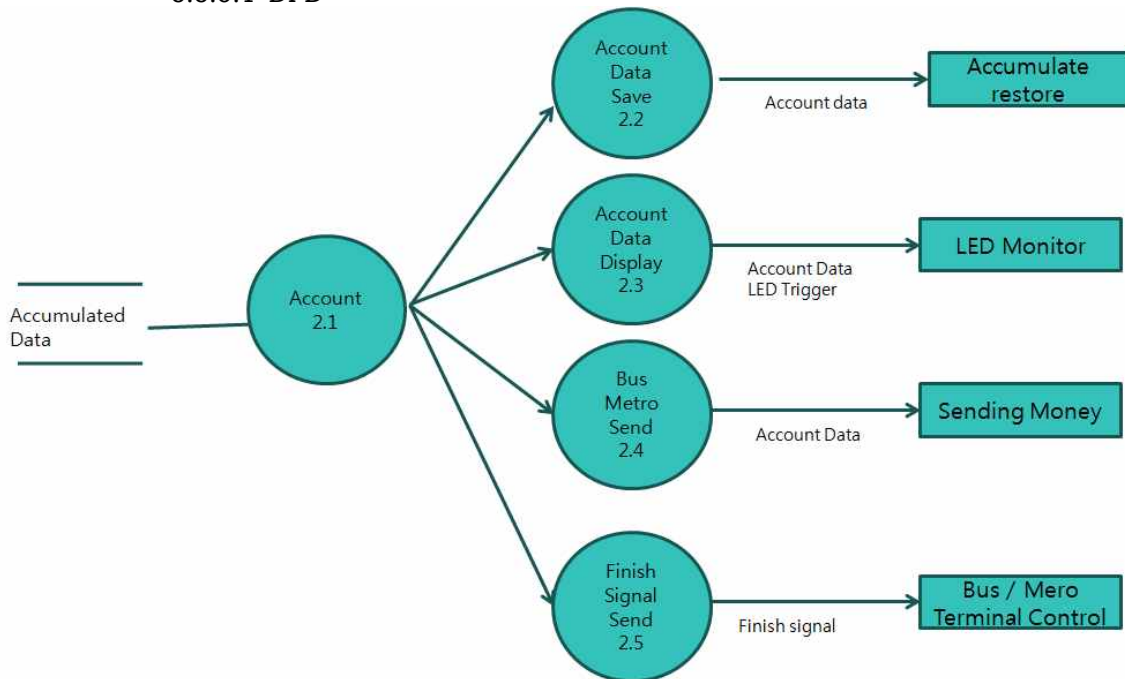
Reference No.	1
Name	Get Trigger
Input	Account Trigger , Accumulated Data
Output	Accumulated Data
Process Description	정산을 시작하라는 Trigger가 들어오면 Main Controll이 Accumulated Data를 받으라는 신호를 준다.

3.3.4.2.2 Process 2

Reference No.	2
Name	Main Control
Input	Accumulated Data
Output	Account Data, LED Trigger, Finish Signal
Process Description	Accumulated Data를 받아서 그 누적 정보를 갖고 정산을 시작한다 정산을 해서 그 정보를 Account Data로 바꿔주고 정산된 금액을 각 버스, 지하철 회사에 돈을 보내준다. 그리고 LED Monitor에 정산이 완료됐다는 신호와 각 단말기에 다시 운영을 시작하라는 Final Signal을 보내준다.

3.3.5 DFD level 2

3.3.5.1 DFD



3.3.5.2 Process Specification

3.3.5.2.1 Process 2.1

Reference No.	2.1
Name	Account
Input	Accumulated Data
Output	Account Data
Process Description	Accumulated Data를 받아서 계산식에 맞게 정보를 계산해서 정산을해서 Account Data로 만들어준다.

3.3.4.5.2 Process 2.2

Reference No.	2.2
Name	Account Data Save
Input	Account Data
Output	Account Data, Account Data Save Trigger
Process Description	Account Data를 받아서 그 정보를 저장하게 하는 Trigger를 보내줘서 그 정보를 날짜별로 구별해서 누적해서 저장하게 해준다.

3.3.4.5.3 Process 2.3

Reference No.	2.3
Name	Account Data Display
Input	Account Data
Output	Account Data, LED Trigger
Process Description	Account Data를 받아서 그것을 LED Monitor에 Trigger를 줘서 LED Monitor에 정산 결과를 출력하게 해준다.

3.3.4.5.4 Process 2.4

Reference No.	2.4
Name	Bus / Metro Send
Input	Account Data
Output	Account Data
Process Description	2.1에서 정산이 완료된 Account Data를 받아서 거기서 계산된 Bus의 금액과 Metro의 금액을 각각의 회사로 지급해준다.

3.3.4.5.5 Process 2.5

Reference No.	2.5
Name	Finish Signal Send
Input	Account Success Signal
Output	Finish Signal
Process Description	2.1에서 정산이 완료가 되면 그 신호를 받아서 Bus / Metro 단 말기로 정산이 완료되었다는 신호를 전해줘서 운행이 다시 시작하게 해준다.

4 Overall

