**Software Design Specification**

**for Autonomous Object Tracking Robot**

**Author**

이종혁

**Date**

2022-12-02

**Team Information**

이종혁 개인 프로젝트

**Table of Contents**

1. Introduction
   1. Purpose
   2. Scope
   3. Definition, acronyms, and abbreviations
   4. Reference
   5. OverDisplay

2 Overall Description

2.1 Product Perspective

2.2 Product functions

2.3 User characteristics

2.4 Constraints and Assumptions

2.4.1 Definition of Tracking Target

2.4.2 Definition of Obstacle

2.4.3 Driving Condition

3 Structured Analysis

3.1 Autonomous Object Tracking Robot 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

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 Data Dictionary

3.1.2.6 Overall DFD

1. Introduction
   1. Purpose

본 문서는 자율주행 물체 추적 로봇 프로젝트의 요구사항 Analysis를 설명한다. 해당 문서의 작성 목적은 자율 주행 로봇 시스템을 구성하는 기능, 컨트롤러, 인터페이스 간의 Data Flow를 설계하는 데에 있다.

* 1. Scope

자율주행 물체 추적 로봇 시스템의 구조는 크게 5가지로 구성된다. 물체 추적, 라인 추적, 위험 예방, 우선 순위 판단, 모터 제어의 기능이 그것이며, 본 문서는 해당 기능들로 구성된 자율주행 물체추적 시스템을 설명한다. 해당 기능들은 카메라, 초음파, 적외선 센서 등 HW 모듈에서 수신한 데이터를 바탕으로 구현된 SW를 통해 구현된다.

* 1. Definition, acronyms, and abbreviations

HW : Hardware

SW : Software

ARS : Autonomous Robot System

FOV : Field of view (Camera, Ultrasonic Sensor)

Camera : Pixy2 Camera

* 1. Reference

[1] Atmega328 Datasheet : https://www.microchip.com/en-us/product/ATmega328

[2] Pixy2 Specification : https://docs.pixycam.com/wiki/doku.php?id=wiki:v2:overview

[3] HC-SR04 Specification : https://www.seeedstudio.com/blog/2019/11/04/hc-sr04-features-arduino-raspberrypi-guide/

[4] L298 Motor Driver Datasheet :

https://www.tech.dmu.ac.uk/~mgongora/Resources/L298N.pdf

[5 SG90 Servo Motor Datasheet :

https://datasheetspdf.com/pdf/791970/TowerPro/SG90/1

[6] TCRT5000 Line Tracking Sensor Datasheet:

https://datasheetspdf.com/pdf/377371/VishayTelefunken/TCRT5000/1

[7] IR1838 Sensor :

<https://datasheet4u.com/datasheet-pdf/ETC/IR1838/pdf.php?id=706293>

* 1. OverDisplay

2장 개발 대상에 대한 설명; 3장 세부 기능 명세

1. Overall Description
   1. Product Perspective

시스템을 구성하는 기능은 5가지로 구성된다. 물체 추적, 라인 추적, 위험 예방, 우선순위 계산, 모터 제어 기능이 그것이며, 우선 순위 계산 기능은 물체 추적, 라인추적, 위험 예방의 3가지 기능 중 최우선순위를 부여할 기능을 결정하는 역할을 수행한다. 마지막으로 모터 제어 기능은 앞선 3개의 기능에서 산출한 속력, 방향 값 중 최우선순위를 할당 받은 기능의 산출 값을 모터에 적용한다.

* 1. Production functions

물체 추적 기능은 미리 선정한 추적 대상에 대하여, 추적 대상과의 거리를 기반으로 설정한 속력 값을 계산해내는 기능이다. 또한 FOV 상의 추적 대상과 이루는 각도를 알아내어 방향 전환을 위한 값을 계산해낼 수 있다. 해당 기능은 초음파 센서와 카메라 모듈의 센싱 데이터에 의존한다.

라인 추적 기능은 두 개의 라인 사이를 벗어나지 않도록 하며 주행하는 기능이다. 라인 추적 기능은 FOV 상 라인의 유무를 판단하여 방향 값을 계산해내며, 이는 적외선 센서와 카메라 센서의 센싱 데이터에 의존한다.

위험 예방 기능은 로봇의 주행 중 마주하게 되는 장애물 등의 요소를 회피하도록 하는 기능이다. 이 기능은 카메라 모듈과 초음파 센서의 파싱 데이터에 의존한다.

우선순위 계산 기능은 위 3개 기능이 산출한 데이터를 종합하여, 현재 로봇의 상태를 파악하고 기능간 최우선순위 기능을 선별하는 기능을 한다. 가장 우선순위가 높은 기능이 산출한 모터의 속력, 방향 값이 로봇 제어 기능에 적용된다.

모터 제어 기능은 우선순위 계산 기능에서 계산해낸 최우선순위 기능이 생산해낸 속력, 방향 값을 모터에 적용하는 직접 기능이다. 해당 과정 중 방향 값 입력에 따라 모터의 전, 후진, 그리고 속력 값을 보정하여 원하는 방향으로의 주행을 가능케 한다.

* 1. User characteristics

사용자는 일회용 AA 알칼라인 건전지 혹은 18650 배터리 쌍을 이용하여 하드웨어 구동에 필요한 전원을 공급한다.

Arduino System에 전원을 공급하면 소프트웨어가 자동으로 구동된다.

사용자는 IR 원격 컨트롤러를 통해 로봇 구동에 대한 제어를 수행할 수 있다.

* 1. Constraints and Assumption
     1. Definition of Tracking Target

물체 추적 대상은 지름 4cm 가량의 흰색 구체로 한다. 물체의 인식 근거는 사전에 등록된 색체가 되며, 주변 환경에 유사한 색체를 가진 물체가 있다면 주행은 제한된다.

라인의 식별은 TCRT4000 센서의 적외선 신호를 이용하여 판단한다. 또한 카메라 모듈을 통해 이를 보조하는데, 주변 조도가 어두울 경우 라인 추적 기능이 제한된다.

추적할 대상의 인식율은 90퍼센트(TBD) 이상이어야 한다.

* + 1. Definition of Obstacle

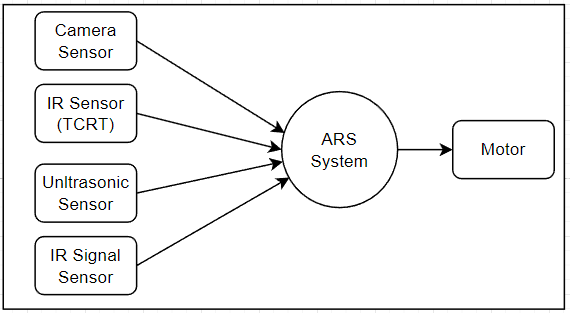
장애물의 대상은 카메라가 인식하는 y 좌표 값이 20(TBD) 이상의 추적 대상이 아닌 물체로 한다. 장애물은 물체 추적 기능을 수행할 때에 설치되며, 따라서 장애물의 색체는 추적 대상의 색체와 유사해서는 안된다.

* + 1. Driving Condition

주행 환경은 주행에 장애가 되지 않도록 평평한 지면이어야 한다.

로봇이 주행하는 환경은 조도 500Lux(TBD) 이상의 환경이 갖춰져야 한다.

1. Structured Analysis
   1. Autonomous Object Tracking Robot System
      1. System Context Diagram
         1. Basic System Context Diagram



* + - 1. Event List

|  |  |
| --- | --- |
| Input/Output Event | Description |
| Camera Sensor Input | Pixy Camera로부터 전송되는 물체의 position, width, height 등의 정보 |
| IR Sensor  Input(TCRT) | TCRT 센서로부터 인식한 라인 존재의 유무에 대한 정보 |
| IR Signal  Input | IR 리시버에서 인식하는 정보. 원격 리모트 컨트롤러의 신호를 입력 받아 전달하며, 로봇 시스템의 On/Off 기능을 담당한다. |
| Ultrasonic  Sensor Input | 초음파 센서로부터 입력되는 거리 정보 |

* + - 1. The System Context Diagram



* + 1. Data Flow Diagram
       1. DFD level 0
          1. DFD



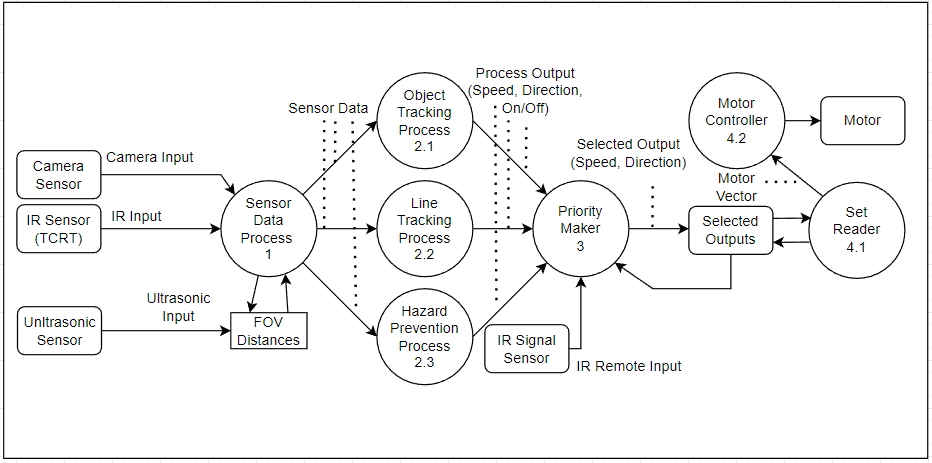
* + - * 1. Process Specification

|  |  |
| --- | --- |
| Reference No. | 0 |
| Name | ARS System |
| Input | Camera Sensor Input, IR Input, Ultrasonic Input |
| Output | Motor Data |
| Process Description | 각종 센서로부터 입력 받은 Input 데이터를 이용하여 이를 파싱하고, 파싱된 데이터를 알고리즘에 적용하여 물체 추적, 라인 추적, 위험 예방 중 기능 간 우선 순위를 선정한다.  우선 순위가 가장 높은 기능에서 연산한 데이터를 모터 제어에 적용하게 된다. |

* + - * 1. Data Dictionary

|  |  |  |
| --- | --- | --- |
| Input/Output Event | Description | Format/Type |
| Camera Input | 카메라 센서가 입력 받는 물체의 좌표, 너비, 높이, 색체 코드 정보.  추적 대상과 장애물 대상 2개의 최대 3개 대상을 멤버 변수로 갖는 구조체를 생성한다. | Structure Array |
| uint16\_t signature :  전방 물체의 색체 코드  uint16\_t pos\_x :  전방 물체의 x좌표  uint16\_t pos\_y :  전방 물체의 y좌표  uint16\_t width:  전방 물체의 너비  uint16\_t height:  전방 물체의 높이 |
| IR Input(TCRT) | IR 라인 트레이싱 센서를 통해 입력 받는 라인의 유무 정보 | Structure |
| int line\_left :  좌측 라인의 유무 정보.  0일 경우 미인식, 1일 경우 인식  int line\_middle :  중앙 라인의 유무 정보.  0일 경우 미인식, 1일 경우 인식  int line\_right :  우측 라인의 유무 정보.  0일 경우 미인식, 1일 경우 인식 |
| IR Remote Input | 원격 IR 컨트롤러로부터 전달된 적외선 신호로, 센서를 통해 입력 받는 로봇 제어 정보 | Unsigned Integer |
| uint16\_t system\_on\_off :  0일 경우 시스템 구동 중단,  1일 경우 시스템 구동 |
| Ultrasonic Input | 초음파 센서로부터 입력 받는 FOV 상 물체와의 거리 정보 | Float |
| Float distance :  Fov상 초음파 센서가 인지한 물체까지의 거리 |
| Motor Data | 초음파 센서를 통한 거리의 누적 값 및 최고 우선 순위 기능이 연산한 모터의 방향 값 및 속력 값 | Structure |
| Float speed :  ARS의 Process를 거쳐 보정된 모터에 적용될 속력 값  Float direction :  ARS의 Process를 거쳐 보정된 모터에 적용될 방향 값 |

* + - 1. DFD level 1
         1. DFD



* + - * 1. Process Specification

|  |  |
| --- | --- |
| Reference No. | 1 |
| Name | Sensor Data Process |
| Input | Camera Input, IR Input, Ultrasonic Input |
| Output | Sensor Data |
| Process Description | 각종 Sensor로부터 데이터를 받아들인다.  해당 데이터들을 종합하여 하나의 구조체 형태로 만들어내고(Sensor Data), 이 구조체를 우선순위 계산 기능에 넘겨주는 역할을 한다. |

|  |  |
| --- | --- |
| Reference No. | 2.1 |
| Name | Object Tracking Process |
| Input | Sensor Data |
| Output | Process Output (Speeds, Directions, On/Off flags) |
| Process Description | Sensor Data를 입력 받아 물체 추적 알고리즘을 수행한다.  전방의 물체가 추적 대상이 맞는지 확인하며,  평균 필터 알고리즘을 적용하여 거리 값을 보정한다.  보정된 전방 물체와의 거리에 따라 모터 방향 값을 다르게 적용하여 추적 대상이 가까워 질수록 방향 전환이 커지도록 하고, 속력이 낮아지도록 기능한다. |

|  |  |
| --- | --- |
| Reference No. | 2.2 |
| Name | Line Tracking Process |
| Input | Sensor Data |
| Output | Process Output (Speeds, Directions, On/Off flags) |
| Process Description | FOV Data를 입력 받아 라인 추적 알고리즘을 수행한다.  만약 현재 전방에 물체 추적 대상이 있다면, 해당 기능의 On/Off 값은 0이 된다.  그렇지 않고, FOV 데이터 상 입력된 라인의 값이 유효할 경우 On/Off 값을 1로 저장하고 아래의 기능을 수행한다.  중앙 TCRT 센서로부터 전달 받은 값을 가장 먼저 읽어, 만약 전달 받은 값이 1인 경우 모터의 방향 값을 0으로 설정한다.  좌측 TCRT 센서로부터 전달 받은 값이 1인 경우, 모터를 좌측으로 30도(TBD) 회전하도록 공유 방향 값에 더해준다.  좌측 TCRT 센서로부터 전달 받은 값이 1인 경우, 모터를 우측으로 30도(TBD) 회전하도록 공유 방향 값에 더해준다. |

|  |  |
| --- | --- |
| Reference No. | 2.3 |
| Name | Hazard Prevention Process |
| Input | Sensor Data |
| Output | Process Output (Speeds, Directions, On/Off flags) |
| Process Description | Sensor Data를 입력 받아 위험 예방 알고리즘을 수행한다.  전방 물체가 장애물인지 추적 대상인지 파악하며,  전방 추적 물체 혹은 유효 장애물과의 거리에 따라 적용할 방향 및 속력 값을 결정한다.  이때 거리 값은 평균 필터를 적용한다. |

|  |  |
| --- | --- |
| Reference No. | 3 |
| Name | Priority Maker |
| Input | Process Output(Speed, Direction, Output) |
| Output | Selected Output(Speed, Direction) |
| Process Description | 기능별 Enable 여부 정보를 가진 On/Off, 속력 값과 방향 값의 집합을 전달받아, 또한 이전에 사용한 모터 출력 값 혹은 현재 물체와의 거리 데이터를 비교하여 어떤 기능에 최우선순위를 부여할지를 결정한다.  가장 높은 우선순위를 할당 받은 기능의 속력, 방향 값을 포함한 구조체를 생성하고, 이를 전역변수로 관리한다. |

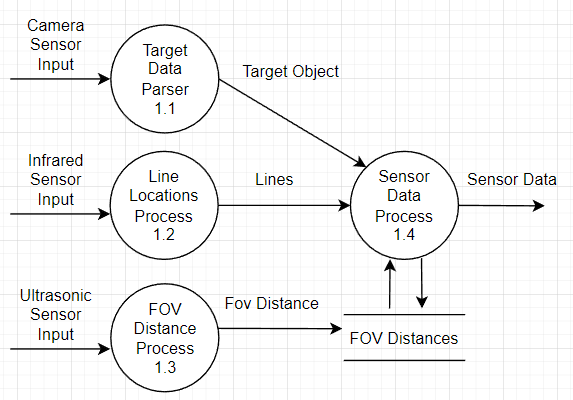
|  |  |
| --- | --- |
| Reference No. | 4.1 |
| Name | Set Reader |
| Input | Selected Output(Speed, Direction, Function Code) |
| Output | Motor Data |
| Process Description | 최우선 순위를 할당 받은 기능이 산출한 속력, 방향 값을 읽어낸다.  만약 해당 데이터에 이상치가 있다고 판단될 경우, 속력 및 방향 값을 보정하여 Motor Controller로 데이터 쌍을 전달한다. |

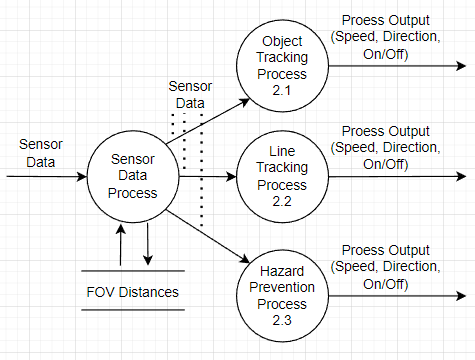
|  |  |
| --- | --- |
| Reference No. | 4.2 |
| Name | Motor Controller |
| Input | Motor Vector |
| Output | - |
| Process Description | 최우선 순위를 할당 받은 기능의 보정된 속력, 방향 값을 전달받는다.  전달받은 속력과 방향 값을 모터가 실제로 이용할 수 있는 형태로 바꾸는 역할을 수행하며, 또 바꾼 형태의 값을 실제로 적용하는 역할을 수행한다. |

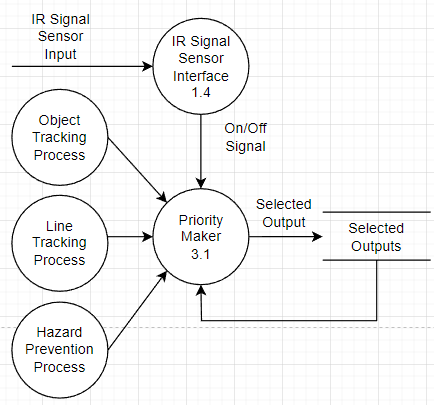
* + - * 1. Data Dictionary

|  |  |  |
| --- | --- | --- |
| Input/Output Event | Description | Format/Type |
| Camera Input | 카메라 센서가 입력 받는 물체의 좌표, 너비, 높이, 색체 코드 정보.  추적 대상과 장애물 대상 2개의 최대 3개 대상을 멤버 변수로 갖는 배열을 입력 받는다.  해당 데이터의 각 요소는 하단 정보와 같으며, 이들을 Parsing 처리한 유효한 추적 대상 구조체를 Target Object로 설정한다. | Structure |
| pos\_x(uint16) : 카메라 모듈이 인식한 물체의 x좌표  pos\_y(uint16) : 카메라 모듈이 인식한 물체의 y좌표  width(uint16) : 카메라 모듈이 인식한 물체의 너비  height(uint16) : 카메라 모듈이 인식한 물체의 높이  angle(uint16) : 카메라 모듈이 인식한 물체와의 각도  signature(uint16) : 카메라 모듈이 인식한 물체의 색체 시그니쳐 넘버 |
| IR Input(TCRT) | IR 라인 트레이싱 센서를 통해 입력 받는 라인과의 거리 정보 | Structure |
| int line\_left :  좌측 라인의 유무 정보.  0일 경우 미인식, 1일 경우 인식  int line\_middle :  중앙 라인의 유무 정보.  0일 경우 미인식, 1일 경우 인식  int line\_right :  우측 라인의 유무 정보.  0일 경우 미인식, 1일 경우 인식 |
| Ultrasonic Input | 초음파 센서로부터 입력 받는 FOV 상 물체와의 거리 정보 | Float |
| Float distance :  Fov상 초음파 센서가 인지한 물체까지의 거리 |
| IR Remote Input | 원격 IR 컨트롤러로부터 전달된 적외선 신호로, 센서를 통해 입력 받는 로봇 제어 정보 | Unsigned Integer |
| uint16\_t system\_on\_off :  0일 경우 시스템 구동 중단,  1일 경우 시스템 구동 |
| Sensor Data | Camera, IR, Ultrasonic 센서로부터 입력받은 값 | Structure |
| Lines lines\_info:  IR Input(TCRT)의 데이터와 동일  TargetObject target\_object\_info:  Camera Input의 데이터와 동일  float\* fov\_distances: |
| Process Output | 각 기능이 연산한 속력과 방향 값, 그리고 기능의 수행 여부(On/Off)가 담긴 구조체 | Structure |
| speed(float) : 우선순위가 가장 높은 기능이 연산한 속력  direction(float) : 우선순위가 가장 높은 기능이 연산한 방향  on\_or\_off(int) : 해당 기능을 수행할지에 대한 여부가 담긴 변수. 수행 할 수 있는 경우 1, 그렇지 않을 경우 0으로 전달. |
| Selected Output | 우선 순위가 가장 높은 기능이 계산한 속력과 방향 값의 구조체로, 전역 변수인 Selected Outputs에 누적된다. | Selected Output |
| speed(float) : 우선순위가 가장 높은 기능이 연산한 속력  direction(float) : 우선순위가 가장 높은 기능이 연산한 방향 |
| Motor Data | 가장 높은 우선 순위를 가진 기능이 산출한 속력, 방향 값으로, 구조체로부터 Speed & Direction 값을 추출해 적용한다. | Structure |

* + - 1. DFD level 2
         1. DFD







* + - * 1. Process Specification

|  |  |
| --- | --- |
| Reference No. | 1.1 |
| Name | Target Data Parser |
| Input | Target Object |
| Output | - |
| Process Description | 카메라 센서로부터 입력 받은 데이터를 Target Object의 정보로 변환한다.  Target Object의 정보를 구조화시켜 Position, width, height, signature number 정보를 Sensor Data의 멤버로 제공한다. |

|  |  |
| --- | --- |
| Reference No. | 1.2 |
| Name | Line Locations Process |
| Input | Infrared Sensor Input |
| Output | Lines |
| Process Description | Infrared Sensor로부터 입력 받은 데이터를 구조화시켜 Line 정보로 변환한다.  Line은 최대 3개까지 인식 가능하다. |

|  |  |
| --- | --- |
| Reference No. | 1.3 |
| Name | FOV Distance Process |
| Input | Ultrasonic Sensor Input |
| Output | FOV Distance |
| Process Description | 초음파 센서로부터 FOV 상 물체의 거리 정보를 입력 받고, 해당 정보를 10개의 거리 정보를 담을 수 있는 Circular Queue에 옮겨 담는다. |

|  |  |
| --- | --- |
| Reference No. | 1.4 |
| Name | Sensor Data Process |
| Input | Target Object, Lines, FOV Distances, On/Off Signal |
| Output | Sensor Data |
| Process Description | 타겟 정보, 라인 정보, FOV 상 거리 정보를 종합하여, 포괄적인 데이터를 가진 구조체를 만들어낸다. |

|  |  |
| --- | --- |
| Reference No. | 2.1 |
| Name | Object Tracking Process |
| Input | Sensor Data |
| Output | Process Output |
| Process Description | Sensor Data를 입력 받아 물체 추적 알고리즘을 수행한다.  물체 추적을 수행하기 위해 물체 방향으로의 바퀴 회전 전환과 추적에 적절한 속력을 만들어내는 역할을 수행한다.  센서 데이터의 x좌표로부터 로봇 전방으로부터 물체까지의 각도를 구해내며, FOV Distances 배열(원형 큐)의 원소를 이용해 거리 값을 구해낸다.  해당 각도와 거리 값을 파라미터로 하여 내부 알고리즘을 수행해, 적절한 속력과 방향 값을 산출한다. 해당 산출 방식은 DFD level 3에서 기술한다. |

|  |  |
| --- | --- |
| Reference No. | 2.2 |
| Name | Line Tracking Process |
| Input | Sensor Data |
| Output | Process Output |
| Process Description | FOV Data를 입력 받아 라인 추적 알고리즘을 수행한다.  만약 현재 전방에 물체 추적 대상이 있다면, 해당 기능의 On/Off 값은 0이 된다.  그렇지 않고, FOV 데이터 상 입력된 라인의 값이 유효할 경우 On/Off 값을 1로 저장하고 아래의 기능을 수행한다.  중앙 TCRT 센서로부터 전달 받은 값을 가장 먼저 읽어, 만약 전달 받은 값이 1인 경우 모터의 방향 값을 0으로 설정한다.  좌측 TCRT 센서로부터 전달 받은 값이 1인 경우, 모터를 좌측으로 30도(TBD) 회전하도록 공유 방향 값에 더해준다.  좌측 TCRT 센서로부터 전달 받은 값이 1인 경우, 모터를 우측으로 30도(TBD) 회전하도록 공유 방향 값에 더해준다. |

|  |  |
| --- | --- |
| Reference No. | 2.3 |
| Name | Hazard Prevention Process |
| Input | Sensor Data |
| Output | Process Output |
| Process Description | FOV Data를 입력 받아 위험 예방 알고리즘을 수행한다.  카메라 모듈로부터 전방 물체가 장애물인지, 추적 대상인지 식별하며 전방 대상에 따라 회피 혹은 감속 등의 알고리즘을 수행한다.  거리 이상치 등을 보정하기 위하여 새로 10개의 거리 값을 받아들여 이를 이용해 평균 필터를 적용하며, 이렇게 필터링 처리된 거리 값을 이용해 내부에 적용된 알고리즘으로 속력 값을 계산한다.  또한 전방 물체가 fov 기준 어느 위치에 있는지에 따라, 바퀴의 회전을 달리하도록 방향 값을 계산한다.  위의 과정을 통해 계산된 속력, 방향 값을 구조체를 생성한다. |

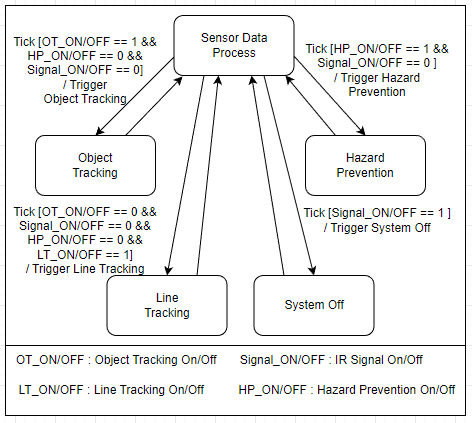
|  |  |
| --- | --- |
| Reference No. | 3.1 |
| Name | Priority Maker |
| Input | Process Outputs, IR Signal Sensor |
| Output | Selected Output |
| Process Description | 원격 컨트롤러부터의 수신된 IR Signal Sensor의 수신 값, 물체 추적, 라인 추적, 위험 예방 기능이 산출한 구조체의 On/Off 값을 입력 받아 실제 모터에 적용될 속력과 방향 값을 결정한다.  모든 기능의 On/Off 값이 On일 경우, 아래와 같은 우선순위를 부여 받는다. 그러나 각 기능이 Off 인 경우 “>”기호를 따라 다음 순서의 기능의 수행 가능 여부를 판단하게 된다. .  원격 컨트롤러 신호 > 위험 예방 > 물체 추적 > 라인 추적 |

|  |  |
| --- | --- |
| Reference No. | 3.2 |
| Name | IR Signal Sensor Interface |
| Input | IR Signal Sensor Input |
| Output | On/Off Signal |
| Process Description | 원격 IR 컨트롤러로부터 입력 받은 신호를 Sensor Data 구조체의 멤버로 저장한다. |

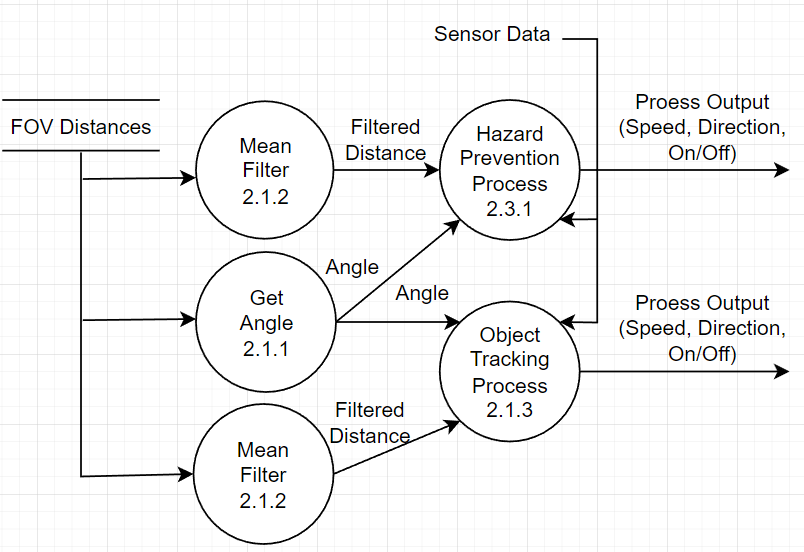
* + - * 1. Data Dictionary

|  |  |  |
| --- | --- | --- |
| Input/Output Event | Description | Format/Type |
| Camera Sensor  Input | 카메라 센서가 입력 받는 물체의 정보가 포함된 18 바이트 Serial Data. 물체의 좌표, 너비, 높이, 색체 코드 정보를 가지고 있으며, DFD Level 1의 Camera Input과 동일하다. | Byte Array |
| Infrared Sensor  Input | 원격 IR 컨트롤러와 IR 라인 트레이싱 센서를 통해 입력 받는 라인과의 거리 정보.  각각의 센서는 1개의 이진형 입력을 받아들이며, 총 3개의 센서의 출력 값으로 이루어진 Lines 구조체를 생성하게 된다. | Unsigned Integer |
| Ultrasonic Sensor  Input | 초음파 센서로부터 입력 받는 FOV 상 물체와의 거리 정보  DFD Level 1의 Ultrasonic Input과 동일하다. | Float |
| Target Object | 카메라 센서 입력 값을 파싱하여 구분한 정제된 값 | Structure |
| pos\_x(uint16) : 카메라 모듈이 인식한 물체의 x좌표  pos\_y(uint16) : 카메라 모듈이 인식한 물체의 y좌표  width(uint16) : 카메라 모듈이 인식한 물체의 너비  height(uint16) : 카메라 모듈이 인식한 물체의 높이  signature(uint16) : 카메라 모듈이 인식한 물체의 색체 시그니쳐 넘버 |
| Lines | 3개의 IR 센서로부터 입력 받은 라인과의 거리 정보를 합친 구조체형 데이터 | Structure |
| line\_left(short) : 좌측 센서가 받아들인 라인 존재 여부 데이터  line\_center(short) : 중앙 센서가 받아들인 라인 존재 여부 데이터  line\_right(short) : 우측 센서가 받아들인 라인 존재 여부 데이터 |
| FOV Distances | 초음파 센서로부터 입력 받은 거리 값의 리스트. 10개의 거리 정보가 누적되어 선형 큐의 형태로 갱신된다. | Float array |
| distance(float) : 장애물과의 거리로, 단위는 cm |
| Sensor Data | Target Object, Lines, Obstacle 데이터가 종합적으로 묶인 구조체 | Structure |
| Process Output | 각 기능이 연산한 속력과 방향 값, 그리고 기능의 수행 여부(On/Off)가 담긴 구조체 | Structure |
| speed(float) : 우선순위가 가장 높은 기능이 연산한 속력  direction(float) : 우선순위가 가장 높은 기능이 연산한 방향  on\_or\_off(int) : 해당 기능을 수행할지에 대한 여부가 담긴 변수. 수행 할 수 있는 경우 1, 그렇지 않을 경우 0으로 전달. |
| IR Signal Sensor  Input | 원격 컨트롤러로부터 수신된 로봇 시스템의 On/Off 정보로, High와 Low의 이진형 값. | Const int |
| On/Off Signal | IR Signal Sensor Input을 1과 0의 데이터로 Convert한 것으로, High는 1, Low는 0에 대응한다. | int |
| on\_or\_off(int) : 로봇 시스템을 구동할지에 대한 여부가 담긴 변수. 수행 할 수 있는 경우 1, 그렇지 않을 경우 0으로 전달. |
| Selected Output | 우선 순위가 가장 높은 기능이 계산한 속력과 방향 값, 그리고 우선 순위 코드의 세트 | Structure |
| speed(float) : 우선순위가 가장 높은 기능이 연산한 속력  direction(float) : 우선순위가 가장 높은 기능이 연산한 방향  priority\_code(int) : 우선순위가 가장 높은 기능의 고유 코드 |
| Selected Outputs | 이전 Cycle들에 의해 선정된 모터와 속력 값의 Stack.  Selected Output의 배열로, 최대 5개의 원소를 유지한다. | Structure Array |

* + - * 1. State Transition Diagram



* + - 1. DFD level 3
         1. DFD



* + - * 1. Process Specification

|  |  |
| --- | --- |
| Reference No. | 2.1.1 |
| Name | Get Angle |
| Input | SensorData.pos\_x |
| Output | Angle |
| Process Description | 센서 데이터의 x좌표를 이용하여 전방 추적 물체가 차량 전방을 기준으로 좌, 우측 중 어느 방향에 있는지 파악한다.  하드웨어 스펙 상 x좌표의 Range가 0~315이며, 카메라 fov 범위는 0~60도이므로, SensorData.pos\_x \* 60/315의 계산으로 추적 대상과의 각도를 구해낸다.  위 과정에서 구해낸 각도 a가 30 이상이라면, a-30을 반환한다.  위 과정에서 구해낸 각도 a가 30 미만이라면, (30-a)\*(-1)을 반환한다. |

|  |  |
| --- | --- |
| Reference No. | 2.1.2 |
| Name | Mean Filter |
| Input | FOV Distances |
| Output | Filtered Distance |
| Process Description | FOV Distances 컨테이너의 모든 원소의 값을 더하여 산술 평균을 반환한다. |

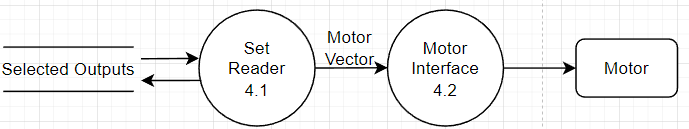
|  |  |
| --- | --- |
| Reference No. | 2.1.3 |
| Name | Object Tracking Process |
| Input | Angle, Filtered Distance, Sensor Data |
| Output | Process Output |
| Process Description | Sensor Data의 색체 코드를 통해 전방 대상이 추적 대상인지 확인하고, on\_or\_off 값을 1 또는 0으로 설정한다.(1은 기능 수행 가능, 0은 기능 수행 불가로 가정한다.)  입력 받은 각도에 30.0을 나누어 0~1 사이의 float 값을 만들어내는데, 해당 값은 방향 전환을 위한 속력 비로 이용된다.  또한 평균 필터로 계산해낸 거리 값을 이용하여 물체 추적 수행시 모터에 입력할 속력 값을 산출한다. |

|  |  |
| --- | --- |
| Reference No. | 2.3.1 |
| Name | Hazard Prevention Process |
| Input | Angle, Filtered Distance, Sensor Data |
| Output | Process Output |
| Process Description | Sensor Data의 색체 코드를 통해 전방 대상과의 거리 값을 파악하고, on\_or\_off 값을 1 또는 0으로 설정한다.(1은 기능 수행 가능, 0은 기능 수행 불가로 가정한다.)  입력 받은 각도에 30.0을 나누어 0~1 사이의 float 값을 만들어내는데, 해당 값은 방향 전환을 위한 속력 비로 이용된다.  또한 평균 필터로 계산해낸 거리 값을 이용하여, distance의 1/3에 해당하는 값을 계산해 모터에 입력할 속력 값을 산출한다. |

* + - * 1. Data Dictionary

|  |  |  |
| --- | --- | --- |
| Input/Output Event | Description | Format/Type |
| Filtered Distance | Mean Filter를 거쳐 생성된, FOV Distances 컨테이너 원소들의 평균 거리 값 | float |
| Filtered Distance(float) |
| Angle | 전방 물체의 좌표로부터 계산된 각도 값. | Byte |
| Angle(int) : 로봇 전방 기준 좌측의 경우는 음수, 우측의 경우는 양수 값을 가진다. 값의 범위는 -30~30에 제한된다. |
| Process Output | 각 기능이 연산한 속력과 방향 값, 그리고 기능의 수행 여부(On/Off)가 담긴 구조체 | Structure |
| speed(float) : 특정 기능을 수행하여 계산된 속력 값  direction(float) : 특정 기능을 수행하여 계산된 방향 값  on\_or\_off(int) : 해당 기능을 수행할지에 대한 여부가 담긴 변수. 수행 할 수 있는 경우 1, 그렇지 않을 경우 0으로 전달. |

* + - 1. DFD level 4
         1. DFD



* + - * 1. Process Specification

|  |  |
| --- | --- |
| Reference No. | 4.1 |
| Name | Set Reader |
| Input | Selected Data, Selected Output |
| Output | Motor Vector, Selected Output |
| Process Description | Selected Data의 Set을 읽어와, 이상치가 없는 지 확인한다.  모터의 속력은 60cm/s를 넘어서면 안되며, 모터의 방향 각도는 60도를 넘을 수 없다.  이상치가 확인될 경우, 모터의 속력과 방향을 0으로 설정한다.  이상치가 없음이 확인될 경우, 모터 속력 및 방향 값이 그대로 Motor Vector 구조체의 멤버 변수로 저장되어 이를 반환한다. |

|  |  |
| --- | --- |
| Reference No. | 4.2 |
| Name | Motor Interface |
| Input | Motor Vector |
| Output | - |
| Process Description | Motor Vector를 입력 받아, 해당 구조체의 멤버인 속력과 방향을 사용할 수 있는 형태로 변환한다.  방향 값의 경우, -30~30 사이의 값으로 전달되는데 -10~10 사이의 값의 경우는 직진, -10 미만의 값의 경우 좌회전, 10 이상의 값은 우회전으로 가정한다.  이후 DC 모터의 방향, 속력 값을 실제로 적용한다. 모터에 적용될 속력 값(Pulse Width) 에 방향 값의 절대값 / 30을 곱해주는데, 만약 방향 값이 음수일 경우는 좌측 모터에, 방향 값이 양수일 경우는 우측 모터에 대해 연산을 적용한다. |

* + - * 1. Data Dictionary

|  |  |  |
| --- | --- | --- |
| Input/Output Event | Description | Format/Type |
| Motor Vector | DC 모터에 최종적으로 적용할 벡터 값을 가진 구조체 | Structure |
| speed(float array) : DC모터에 적용할 속력 값  direction(float array) : 우선순위가 가장 높은 기능이 연산한 방향 |

* + - 1. Overall DFD

