

## 2017 지능형모형차 경진대회 보고서

학 교	강원대학교
팀 명	수동주행차
유 형	본 경기
팀 장	이효근 (전자통신공학과)
팀 원	홍명지 (전자통신공학과) 김성민 (전자통신공학과) 김태영 (전자통신공학과) 최승주 (전자통신공학과)

# 1. 개요

## 1.1 설계 배경

한양대 지능형 모형차 경진대회에서 사용하는 자율주행은 현대 사회에서 각광받고 발전해나가는 Smart Car 의 핵심기술이다. 자율주행은 운전자의 간섭 없이 자동차가 많은 센서들을 이용해 도로상황을 인지하고 스스로 목적지까지 안전하고 정확하게 도착하는 것을 말하는데 이 기술은 스마트화 되어가는 시대에 맞게 운전자의 편의도 보장되지만 교통사고 비중에서 가장 큰 부분인 운전자 부주의에 대한 사고를 줄일 수 있다.

이 자율주행의 핵심기술은 내장형 제어시스템(Embedded Control System)을 이용해 주행의 모든 것을 자동화 하는 것이다. 이 기술은 학부생활을 하면서 배우는 전자, 제어, 통신 등 하드웨어와 소프트웨어의 모든 것을 필요로 한다. 전공관련 기술인 만큼 이 분야에 관심을 가지게 되었고 자율주행 모형차를 제작하면서 지금까지 배워온 기술을 사용하고 성장할 수 있는 기회가 될 거라고 생각했다.

## 1.2 설계 목표

일반적인 속도 구간 주행에서 차선을 이탈하지 않기 위해 라인카메라 2 개를 사용한다. 각각 왼쪽 오른쪽 라인을 검출하여 벗어나지 않고 차체가 중앙에 위치하며 안정적으로 주행할 수 있도록 해준다.

미션 구간인 School Zone 을 인식하는 것은 카메라 센서와 수·발광센서를 이용하여 10cm 의 굵은 선을 인지하여 판단한다. 또한 미션 구간 내에 있는 장애물 회피 구간에서는 적외선 감지 센서를 이용하여 물체를 인지하고 주행차선을 변경하여 회피할 수 있도록 하였다. 마찬가지로 적외선 감지 센서를 이용하여 물체를 인식하였을 때 충돌하지 않도록 자율비상제동(AEB, Autonomous Emergency Braking)을 수행할 수 있도록 하였다. 회피와 충돌방지의 판단은 School Zone 인지 아닌지 여부에 따라 판단할 수 있게 하였다.

마지막으로 이 기술에서 가장 중요한 부분이 아닐 수 없는 속도제어는 Encoder 를 통해 현재 값을 인식하고 PID 제어를 통해 원하는 속도(목표치)에 빠르고 정확하게 도달할 수 있도록 제어하였다.

## 2. 설계 내용

### 2.1 하드웨어 구성

#### 2.1.1 하드웨어 개략도

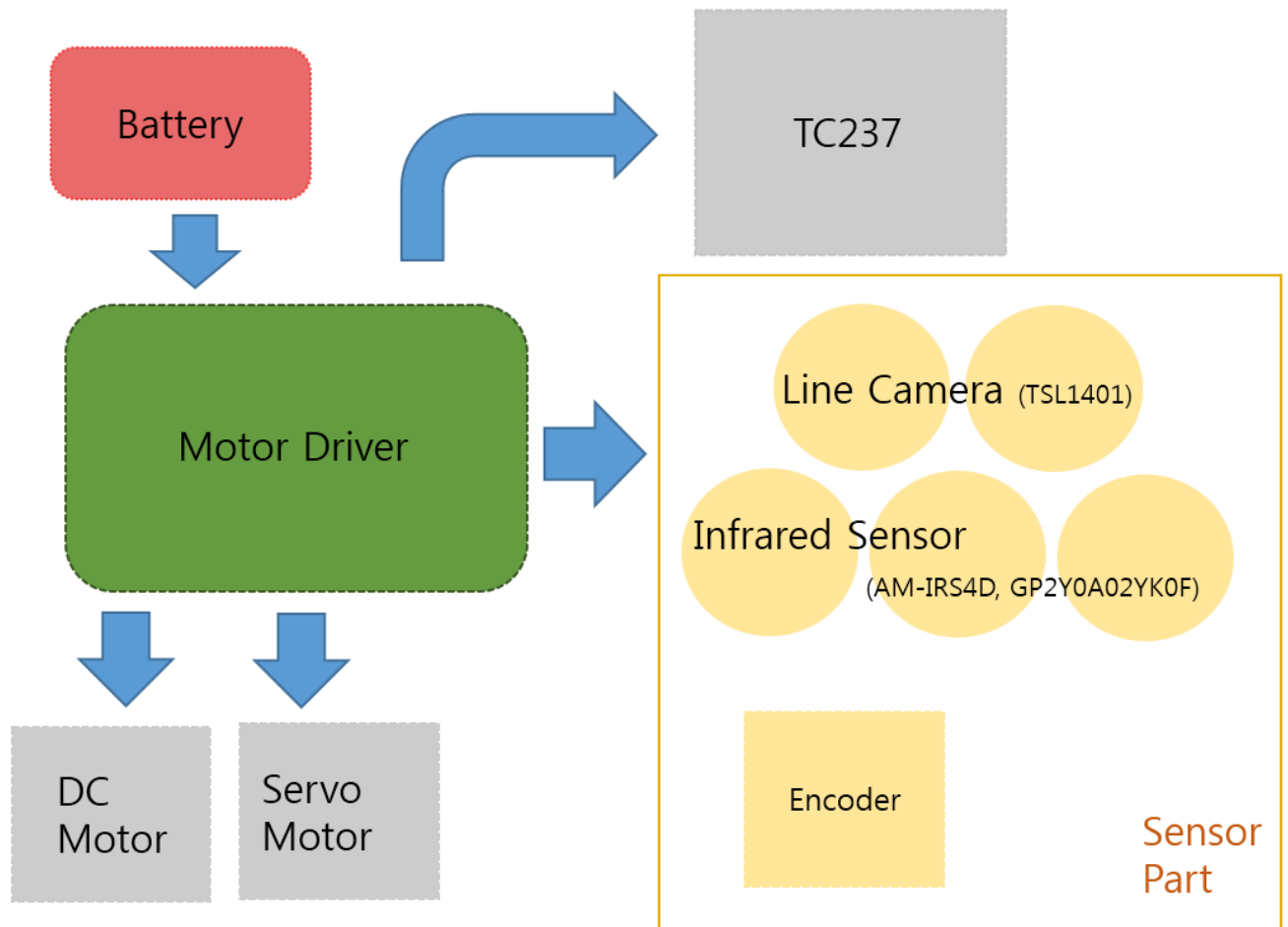
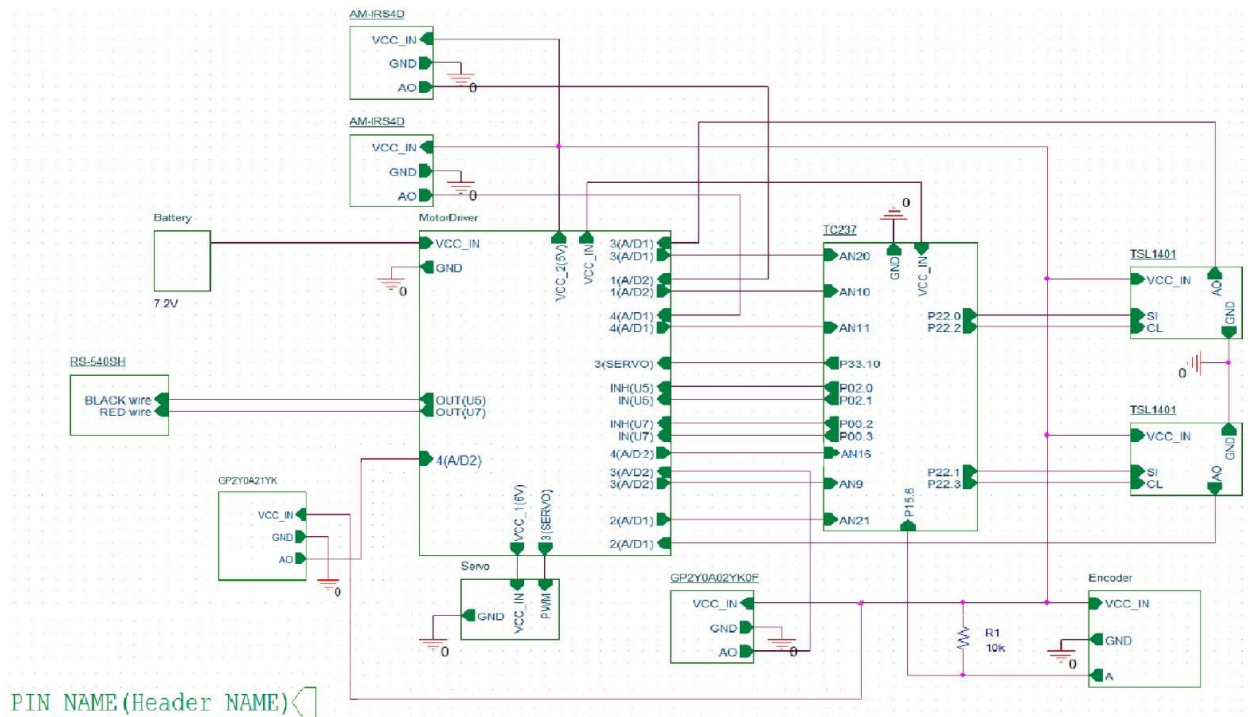


Figure 1.

7.2V 의 Battery 를 모터드라이버에 연결해주고 MCU 는 모터드라이버에 연결된 7.2V 를 나머지 센서들은 모터드라이버에 내장되어있는 regulator 를 이용해서 나오는 5V, 6V 로 구동시켜주었다.

### 2.1.2 회로도





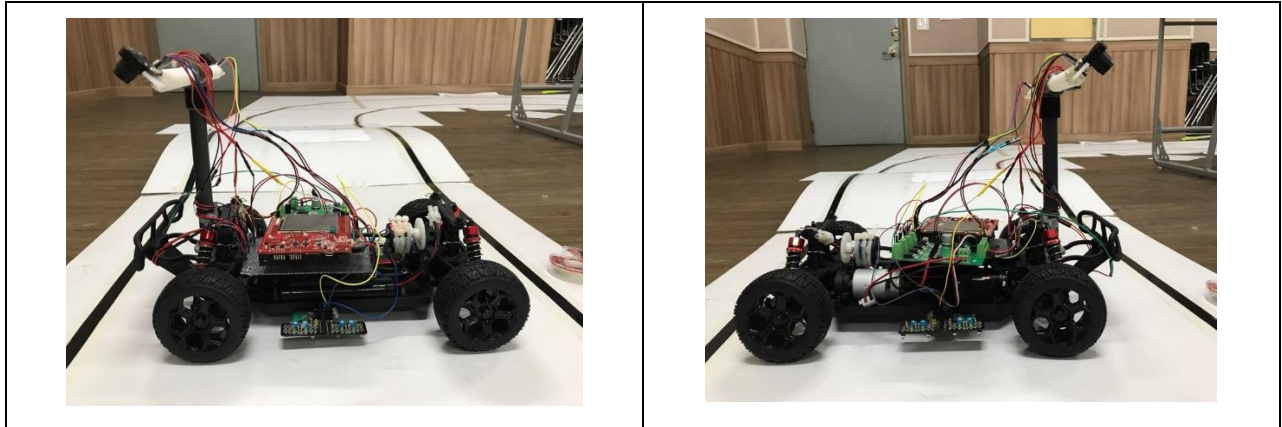
**Figure 2.**

회로도에는 MCU와 센서, 구동, 조향부가 신호와 전력을 어디서 어떻게 주고 받는지 알 수 있다.

똑같은 센서는 알아보기 쉽도록 핀과 port 를 붙여서 사용하였다.

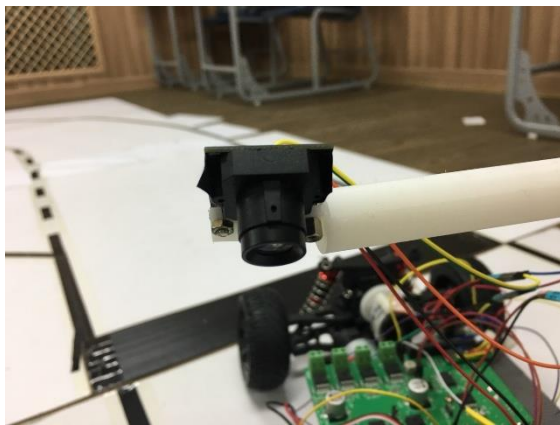
### 2.1.3 전체 외관

전면	후면
	
좌 측면	우 측면



## 2.1.4 센서부

### 2.1.4.1 Line Camera

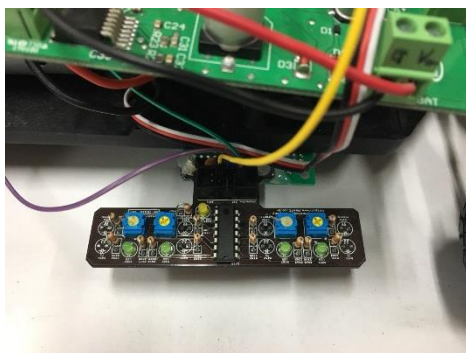


Line Camera 는 128 개의 배열로 128 x 1 Pixel 이다. 이 센서는 충 · 방전의 원리로 각 픽셀마다 순서대로 1~4095 의 아날로그 값을 A0 핀을 통해 출력해준다. 이 신호를 MCU 에서 AN 핀을 통해 Input 시켜준다.

이 센서는 밝은 픽셀일수록 높은 값을 출력시킨다. 이를 이용하여 어두운 픽셀과 밝은 픽셀을 이용해 라인을 인지할 수 있다.

Figure 3.

### 2.1.4.2 수 · 발광센서



수 · 발광센서는 Line Camera 와 마찬가지로 아날로그 값을 출력해준다. 가변저항을 통해 민감도를 조절할 수 있고 검은색을 읽으면 0~20 정도의 값을, 흰색을 읽으면 2000~3500 정도의 값을 Output 한다.

이 신호를 MCU 에서 AN 핀을 통해 Input 시켜준다.

Figure 4.

### 2.1.4.3 적외선 거리센서

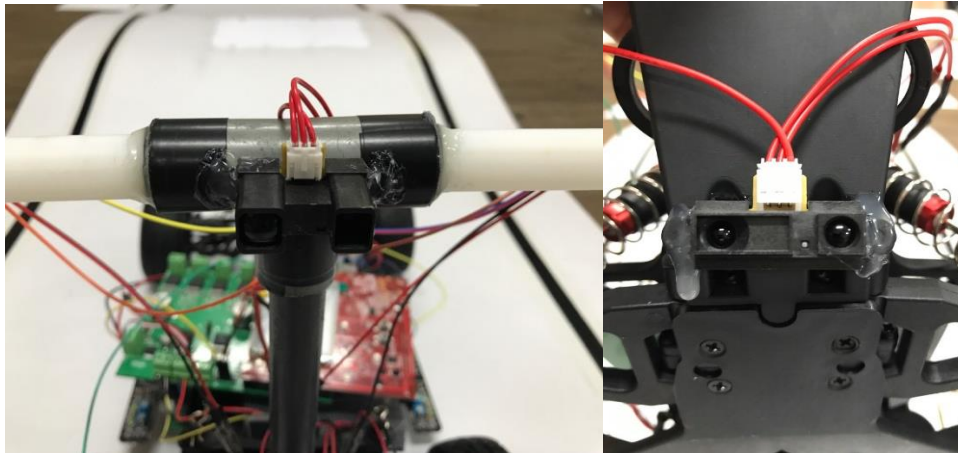


Figure 5.

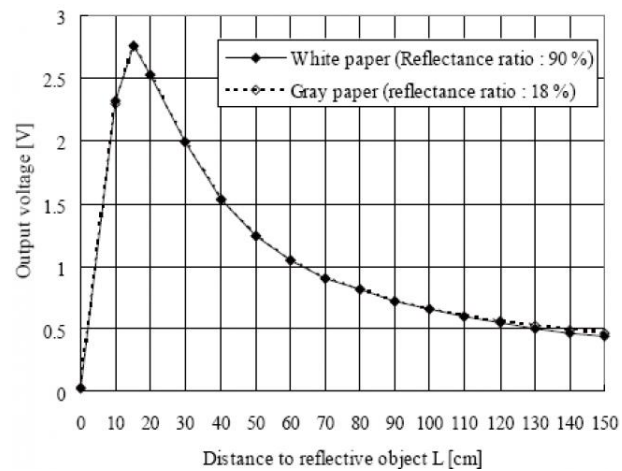


Figure 6.

적외선 거리센서 또한 아날로그 값을 출력해준다. 20~150cm 를 인식가능하며 거리에 따라 다른 아날로그 값을 Output 한다. 거리에 따른 값은 **Figure 6.**을 따라서 출력한다.

이 신호를 MCU 에서 AN 핀을 통해 Input 시켜준다.

상단에 달린 것은 장애물을 인식하고 하단에 달린 것은 언덕을 인지하게 만들었다.

### 2.1.5 구동부



### 2.1.5.1 DC-Motor

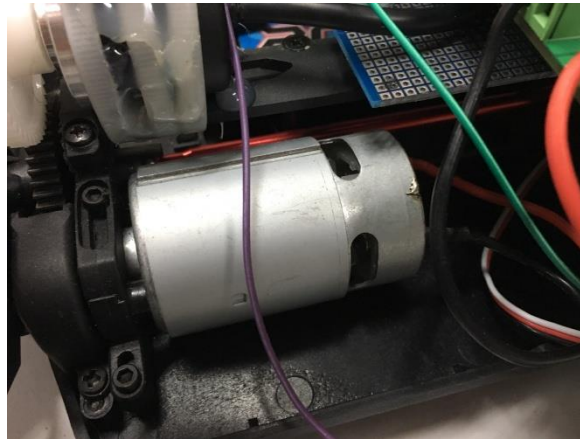


Figure 7.

DC-Motor 는 차체에 달려있는 제품을 사용하였고 MCU 에서 보내는 PWM 을 모터드라이버에서 받아 조절한다.

### 2.1.5.2 Encoder

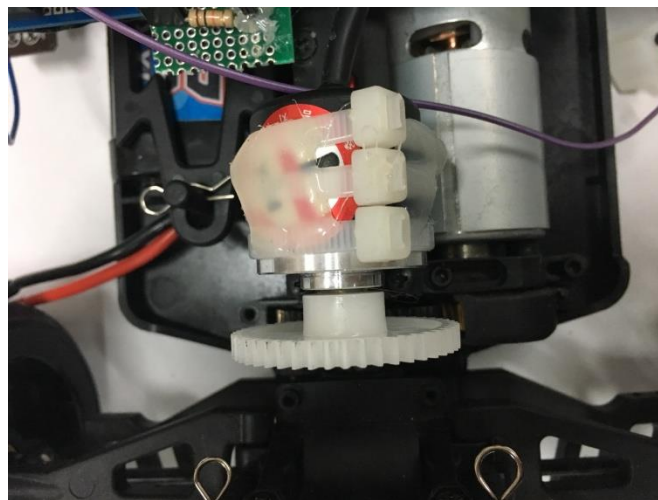


Figure 8.

Encoder 는 차체의 속도를 구하기 위해 사용하는데 DC-Motor 의 기어와 맞물려서 Encoder 가 도는 회전량에 따라 나오는 Pulse 수를 이용하여 사용한다. 이 Encoder 는 1 바퀴당 200Pulse 를 Output 시켜준다. 해당 Encoder 를 사용하기 위해서는 Pull up 저항을 사용해야 했다.

### 2.1.6 조향부

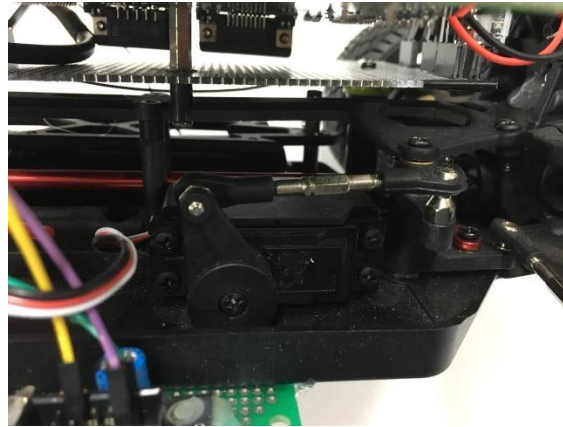


Figure 9.

Servo-Motor 는 조향을 위해 사용되는데 모터드라이버 내에 regulator 를 이용하여 6V 를 주고 MCU 에서 나오는 PWM 을 이용해 회전각을 조절한다.

## 2.2 소프트웨어 구성

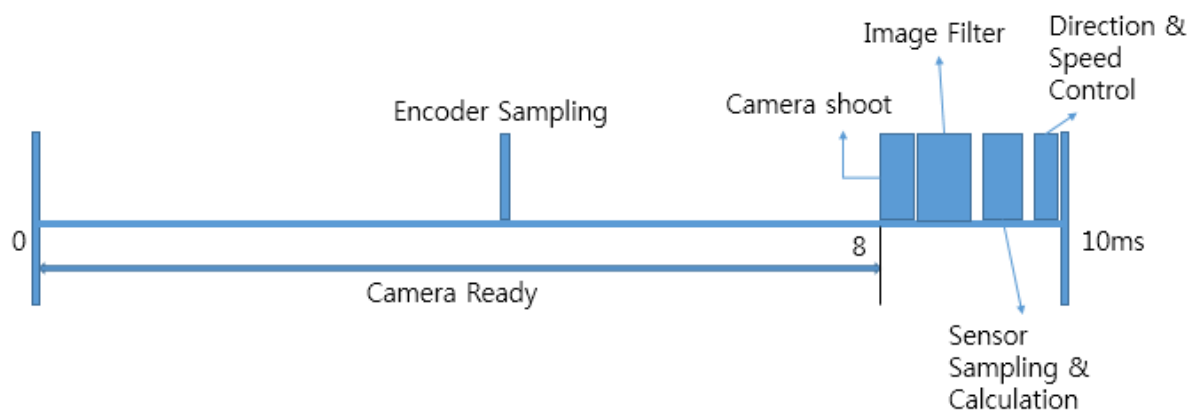


Figure 10.

카메라 센서는 한번 촬영 후 바로 촬영하면 충전할 시간이 부족해서 값이 제대로 나오지 않기 때문에 위 그림에서 Camera Ready 와 같은 적당한 준비 시간이 필요했다. 또한 순서도 중요하기 때문에 Sensor 들이 값을 받아오고 이후 filtering 이나 calculation 을 통해 제어 할 수치를 도출하고 마지막에 이 수치들로 차량의 움직임을 제어했다.

### 2.2.1 Camera signal processing & Lane detecting



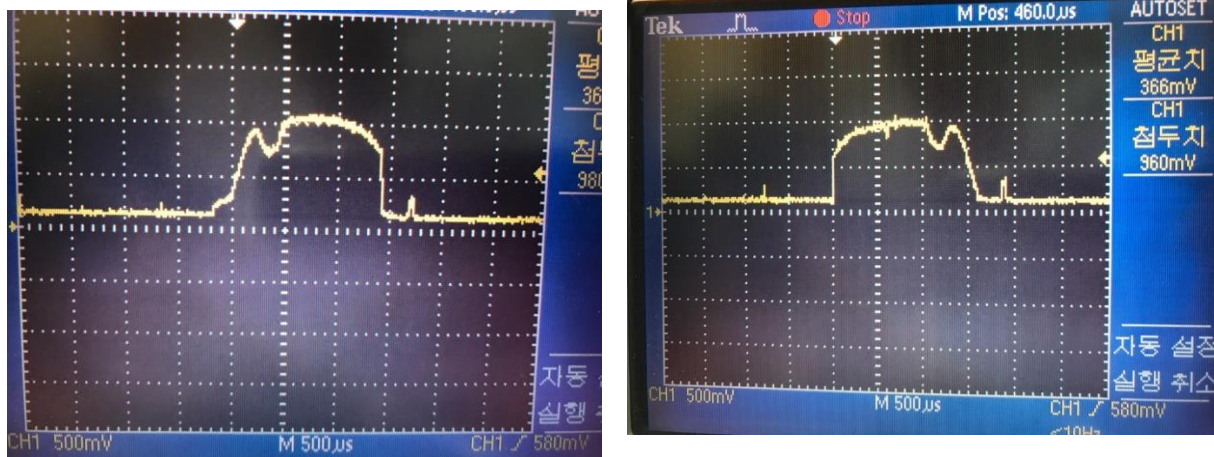


Figure 11.

Figure 11. 은 일반 직진 라인에서 중앙에 차가 있을 때 왼쪽, 오른쪽 카메라의 원본 값이다.

보면 1V정도가 최대값임을 알 수 있는데 이는 조명에 따라 약 800mV~1600mV까지 값이 차이가 상당히 크다. 이를 보완하기 위해 카메라의 최대 최소값이 바뀌어도 보정할 수 있는 정규화(normalization), 잡음을 줄여주는 중간 값 필터(median filter) 검은색 라인을 읽을 수 있게 해주는 Gaussian filter 를 순서대로 사용한다.

또한 Line Camera 센서는 양쪽 끝 값이 미분 값이 나와 값의 차이가 크기 때문에 1~15, 99~128 사이의 값은 배제하여 값을 보다 좋은 값으로 확인할 수 있도록 하였다.

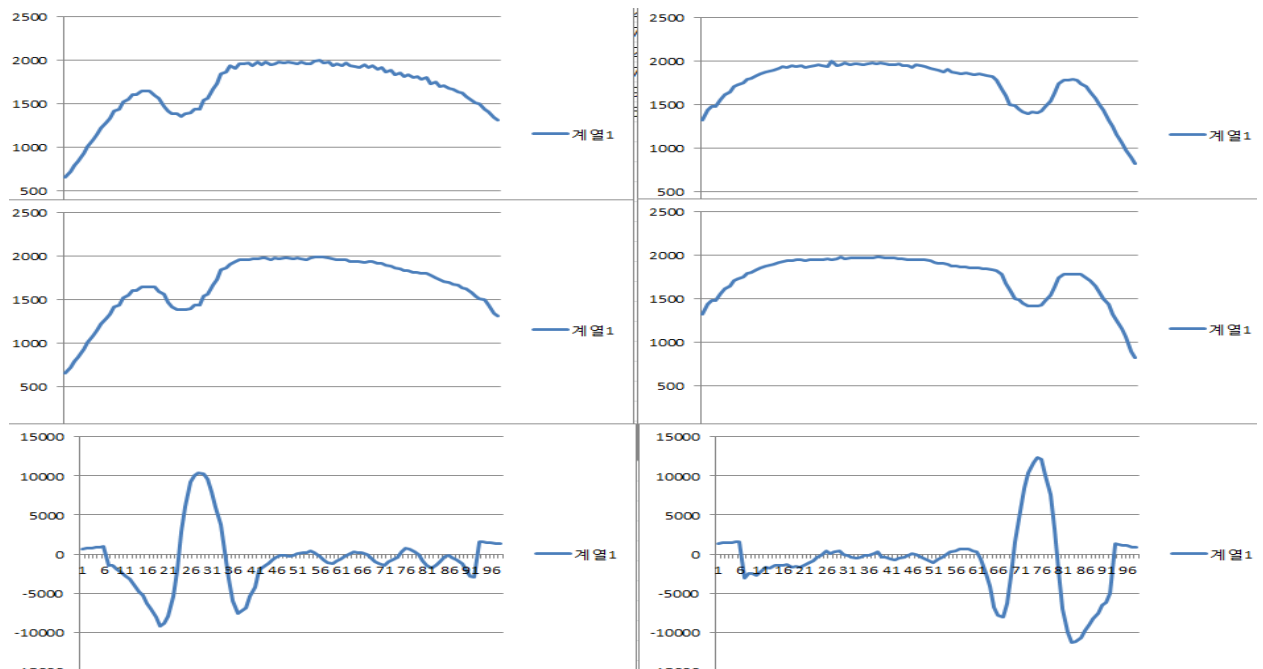


Figure 12.

Figure 12. 은 위부터 순서대로 normalization -> median filter -> Gaussian filter 한 값이다.

Normalization은 2000 정도로 정규화 됨을 알 수 있고 median filter 은 정규화 된 값보다 부드러워진 걸 볼 수 있다. 또한 Gaussian filter 는 라인부분 값이 확실히 튀어 보기 좋게 변한 것을 알 수 있다. Gaussian 을 거친 값에서 여러 번 시험하여 가장 좋은 threshold 값을 정해 잡음은 읽지 않도록 지정하였다. Threshold 값은 정규화를 거쳤기 때문에 카메라에서 받아온 신호들의 수치가 어느 정도 일정해서 변경할 필요 없이 어느 상황에서도 사용가능 해지는 장점이 있다.

### 2.2.3 AEB (Autonomous Emergency Braking)

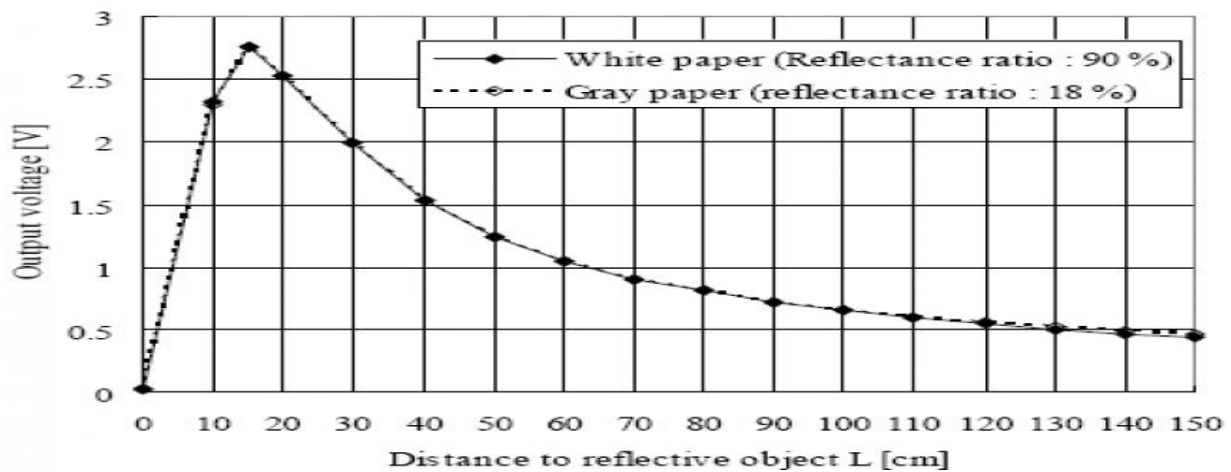


Figure 13.

자율비상제동(AEB)은 적외선 거리센서로 일정 거리가 되면 Figure13. 과 같은 값을 통해 제어를 시켜준다. 차가 물체를 인식했을 때 계속 받아오는 거리센서의 값을 보고 정지시켜준다. 20~150cm 를 볼 수 있지만 잡음을 줄이기 위해 몇 번의 값을 Sampling 하고 그 중간 값을 이용해서 사용한다. 또한 아무것도 읽지 않을 때와 혼란을 줄이기 위해 최대값을 사용하지 않고 약 20~110cm 를 사용하였다.

## 2.2.4 School Zone

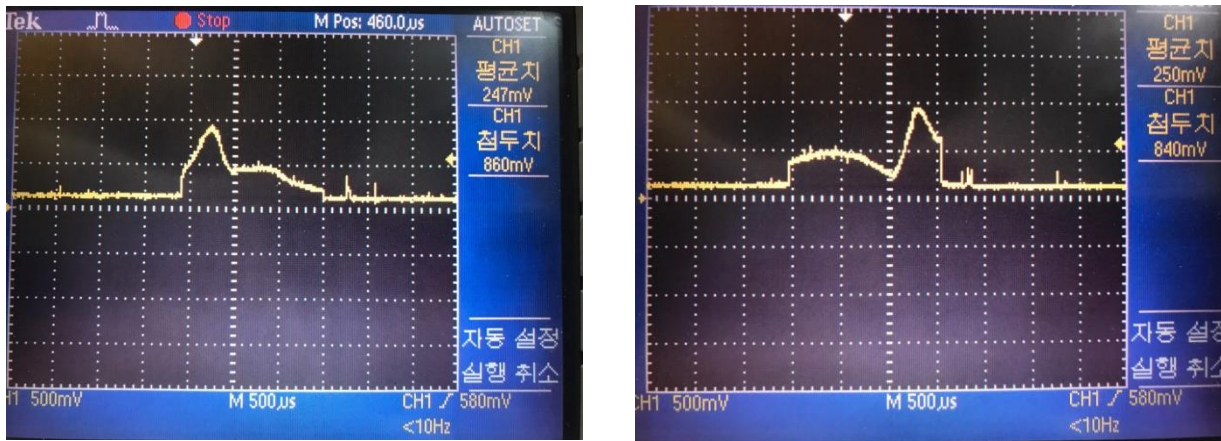


Figure 14.

School Zone 판단을 위한 10cm 의 굵은 선을 카메라가 보게 되면 Figure 14. 과 같이 98 개의 픽셀 중 라인의 가까운 라인들이 값이 크게 감소하는 것을 알 수 있다. 이를 이용하여 내부 30 개 정도의 픽셀을 왼쪽, 오른쪽 카메라의 값을 모두 합해서 평균을 내준 뒤에 일정 값 이하 일 때만 사용하도록 하였다. 또한 시험을 하면서 구간구간 어두운 부분이나 재질이 다른 부분을 읽으면 School Zone 으로 인식하는 경우가 있기 때문에 수 · 발광센서와 함께 판단하여 정확도를 높였다.

## 3. 주요 장치 이론 및 적용 방법

### 3.1 PID 제어

#### 3.1.1 정의

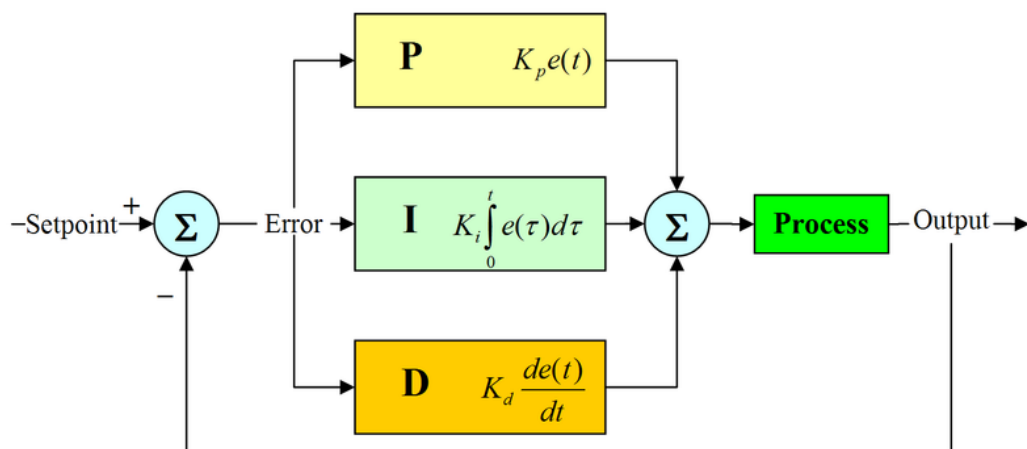


Figure 15.

비례·적분·미분 제어기(PID 제어)는 피드백(feedback) 제어기의 형태를 가지고 있으며, 제어하고자 하는 대상의 출력 값을 측정하여 이를 원하고자 하는 목표 값을 과 비교하여 오차 값(error)을 계산하고 이를 이용하여 제어하는 구조로 되어있다.

- P: 현재 값에서 오차 값의 크기에 비례해 제어한다.
- I: 비례제어만으로 생기는 잔류편차를 줄여 오차를 없애는 작용을 한다.
- D: 출력 값의 급격한 변화에 제동을 걸어 오버 샷(overshoot)를 줄이고 안정성을 향상시킨다.

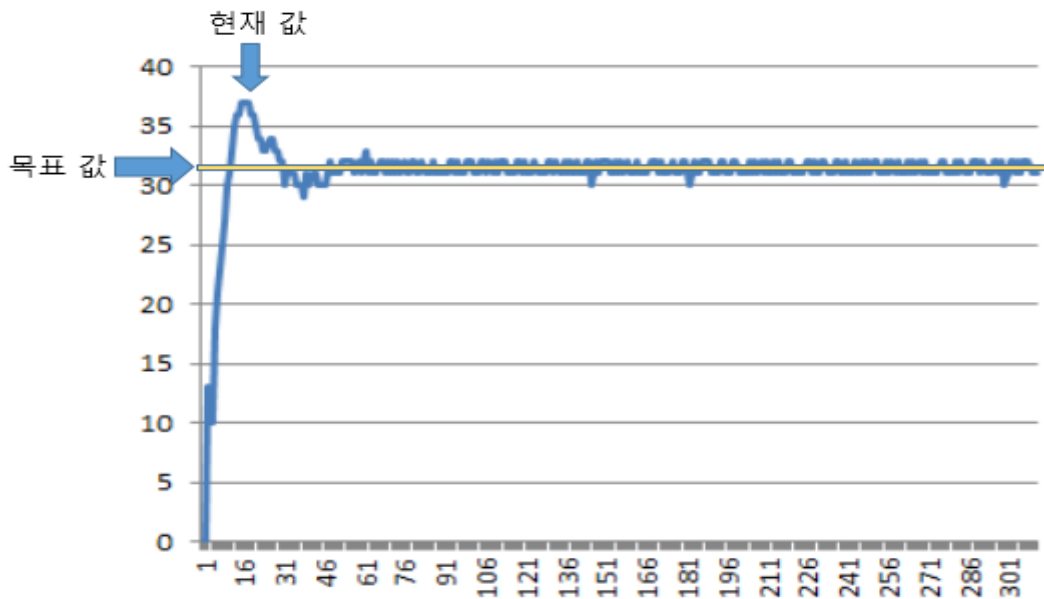


Figure 16.

Figure 16. 은 PID 를 대회 차량에 실제 적용 후 Uart 통신으로 Serial 값을 받아 목표 값과 현재 값을 그래프로 나타낸 것이다.

이론상으로 알고 있는 이상적인 그래프와는 일치하지 않지만 비슷한 그림을 그리며 목표 값에 수렴하는 것을 알 수 있다.

## 3.2 Serial 값과 LCD 를 이용한 Data 분석

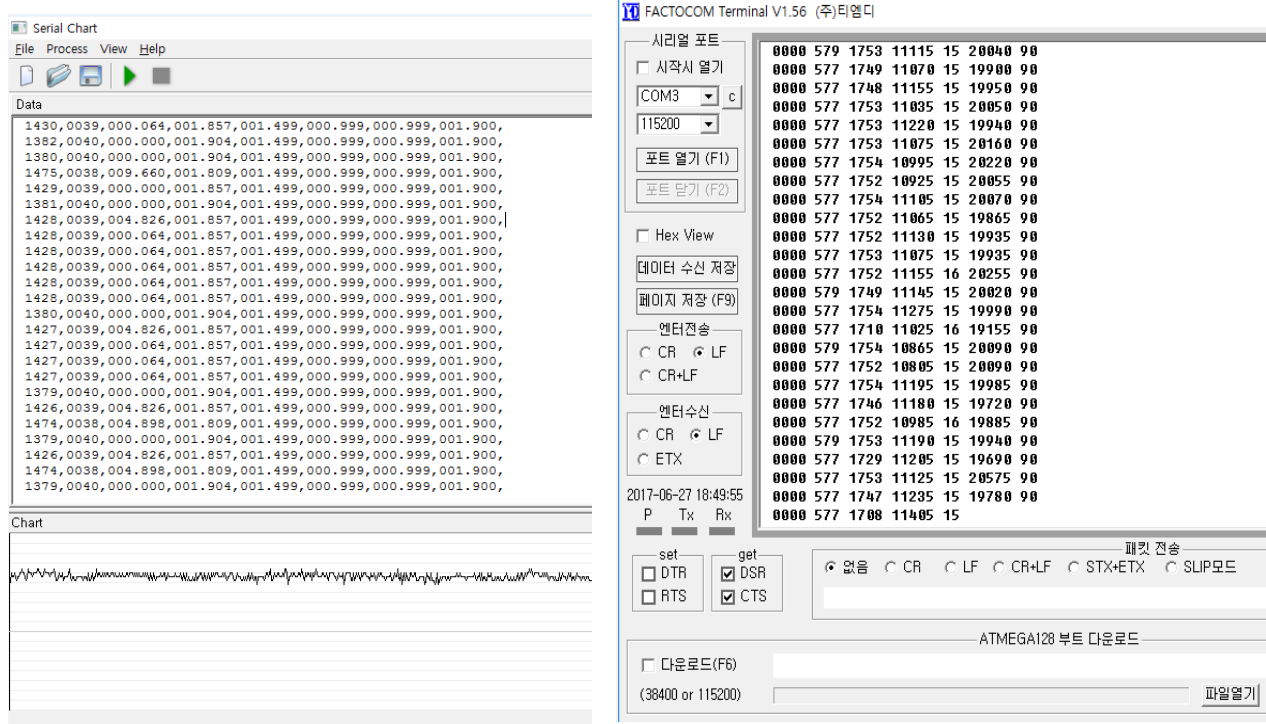


Figure 17.

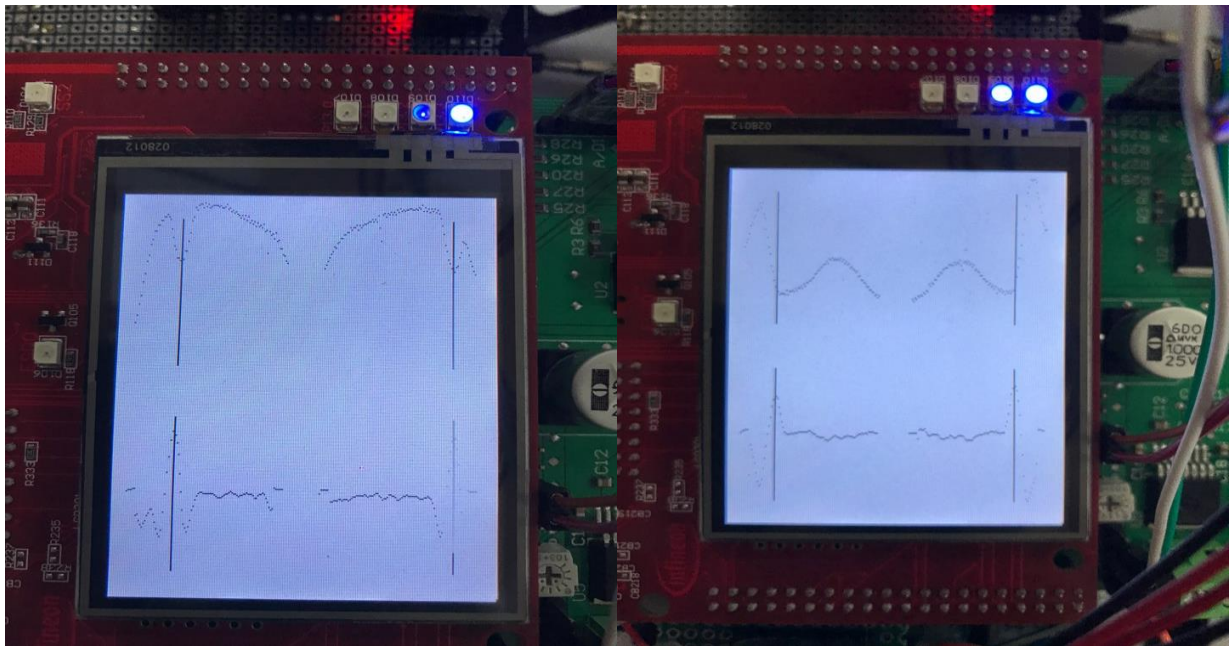


Figure 18.

Serial 값과 LCD 를 이용해서 제어 값을 정하는 것은 실제로 매우 큰 효과를 가져왔다. Uart 를 이용해서 키보드를 이용해 수치를 조정하고 필요 값을 받아서 봄으로서 실시간으로 확인가능하고 실제 반응도 즉각 적으로 보면서 제어가 가능했다. 시간과 능률 모두 잡을 수 있는 효과적인 제어 방법이었다. 또한 터미널 프로그램에 따라 수치로 그래프를 그리면서 보는 것도 가능해서 더 큰 효과를 볼



수 있다. LCD 는 DC 모터를 돌리지 않고 직접 라인을 읽으면서 실시간으로 확인이 가능해서 매우 뛰어난 효과를 보았다. 현재 읽는 pixel 과 카메라의 원본 측정 값, filtering 된 값을 LCD 로 즉시 확인 하며 문제점을 발견, 보완하는 작업이 매우 수월하였다.

Figure 18. 은 카메라센서의 pixel 과 filtering 된 신호와 라인으로 인식하는 pixel 에 줄을 그어 보기 편하게 해놓은 사진이다. 이와 같이 그래프, 직선 등 원하는 그래프를 자유롭게 그려가며 실시간으로 원하는 위치에 보고 사람이 직접 문제점을 분석하고 보완 가능하게 하는 큰 효과가 있었다.

## 4.1 결론

현재까지 배워온 이론부터 대회준비를 하며 들어서 배워서 사용한 이론까지 학업적인 내용뿐이 아닌 준비를 하면서 다양한 경험을 했던 것 같다. 학기 중에는 학부 공부를 병행하며 시간과의 싸움도 해야 했고 시험기간이 겹치면 더한 스트레스도 받아가며 한편으로는 대회 준비를 하며 데이터시트를 읽어서 센서와 보드를 이용하여 구동시키는 것부터 밤낮없이 시리얼 값으로 테스트하고 알고리즘을 구상하고 실제로 테스트를 하면서 알고 있던 이론을 실제로 사용하는 것도 구상하고 새로운 이론들을 접하고 실행하면서 약 4 개월 간의 기간 동안 많은 성장을 한 것 같다. 처음에는 차를 굴리는 것에 급급했다면 완성도가 높아 지면서 주행의 정확도가 높아짐에 대해 좀더 중점을 둘 수 있게 되었다. 또한 구동하고 시험용 트랙을 제작하여 시험을 거치면서 많은 변수들이 발생하고 수정하며 경기장에서 발생할 수 있는 많은 변수들을 예측하여 그 상황에 대해 대비하도록 노력했다.

## 4.2 배운 점

사실 학부에서 이론으로만 배우던 것들을 실제로 실험하는 것이 쉽지 않고 어느 정도 한계가 있다. 하지만 이 대회를 준비하며 실제로 이론들을 적용시키고 코딩과 테스트를 해나가면서 이론과 책에서 나온 그래프들을 우리가 직접 구동하며 확인을 함으로써 더 깊이 이해하고 모르던 것들을 확인하며 전공분야의 능력을 대회준비하기 이전보다 확실히 크게 키운 것 같았다. 대회 준비의 가장 큰 장점은 실습에서 예제를 따라 하는 것이 아니라 실제로 구현 하면서 월등히 성장할 수 있다는 점인 것 같다.

구체적으로 PID 제어나 여러 가지 median, gaussian 과 같은 영상필터는 학교에서 배우고 발표하며 이론으로는 알고 있었지만 실제로 코딩하고 사용할 일은 없었다. 위와 같은 점이 큰 장점 인 것 같다.

하드웨어적으로는 특히 회로관련 수업을 들으면서 배운 건 많았지만 실제로 쓴 것이 적었다. 회로도 를 보는 것부터 그것을 이해하고 직접 활용 하는 것까지 해볼 수 있었고 대회를 준비하면서 센서들의 신호가 오고 가는 것부터 신호를 처리하는 것을 실제로 제작하며 많은 실습을 한 것 같았다.

소프트웨어적으로는 MCU 에 대해 전반적으로 이해를 할 수 있었다. C 언어, 자바에서는 일반적인 순



수 프로그래밍을 코딩 했지만 경험할 일이 많지 않았던 MCU 레지스터를 직접 사용한다는 것 또한 순수 소프트웨어가 아닌 하드웨어와 결합한 펌웨어(firmware)의 많은 부분을 경험할 수 있었다.

이 대회가 끝나가는 이 시점에서 정말 많은 것을 배우고 좋은 경험을 한 것 같고 우리 스스로 한층 더 성장 한 것 같다.

## Appendix – 부품 목록

제조사	부품명	수량	사용목적
TAOS	TSL1401	2	Line 검출 및 school zone 인식
Autonics	E30S4-200-3-N-5	1	모형차 속도 검출
Sharp	GP2Y0A02YK0F	1	장애물 감지
Sharp	GP2Y0A21YK	1	언덕 감지
NewTC	AM-IRS4D	2	school zone 인식
	RS-540H-7520	1	차량구동
	Servo-motor	1	조향제어
VTX KOREA	VTXBCC	1	7.2V, 2000mAh 전원공급