자동주차,RC카기능

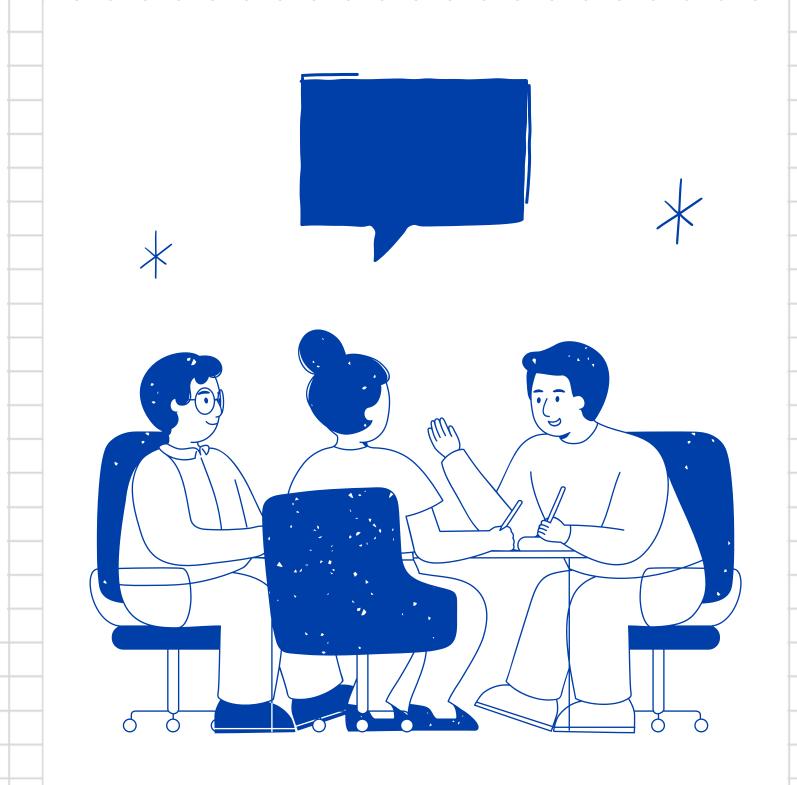
통합및프로젝트

세 얼간이 김준혁 장성준 황승혁











목차

1.

<u>프로젝트 개요</u>

2.

<u>프로젝트 구성</u>

3.

<u>프로젝트 수행 절차 및 방법</u>

4.

<u>프로젝트 수행 결과</u>

5.

데모 영상

6.

<u>제차 평가 의견</u>





1.프로젝트개요

1.개발 목표

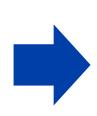
2.개발 컨셉

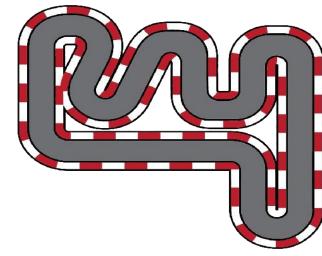




- 1. RC카를 조종해 정해진 경로 주행.
- 2. 장애물이 나타나면 AEB 기능 작동.
- 3. 후방 주차 시 경고 및 브레이크.
- 4 특정 버튼을 누르면 자동주차기능 작동.

추가 목표: 능동 회피 기능 구현.















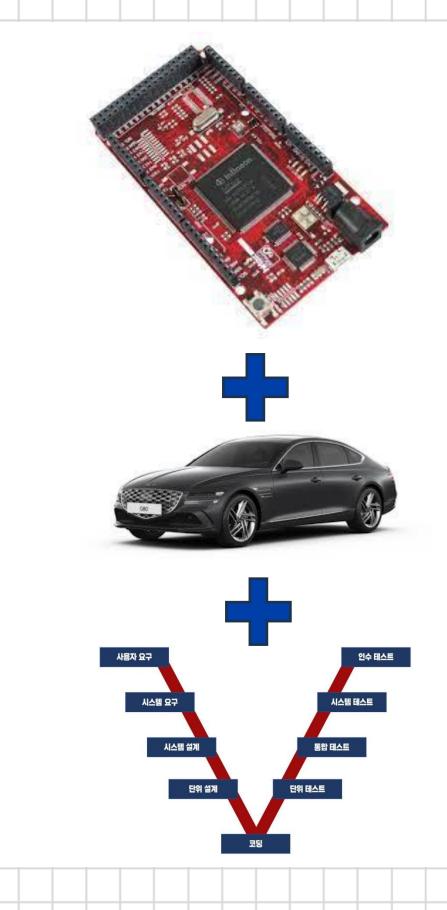


개발컨셉

1.TC275 사용한 개발.

2.실제 차량에 탑재되는 기능들을 RC카에 구현.

3. V 모델에 입각한 프로젝트 진행.





2.프로젝트 구성

1.팀 구성 및 역할

2.프로젝트 일정



팀구성및역할



황승혁

- -프로젝트 매니저 -프로그램 설계
- -시스템 개발



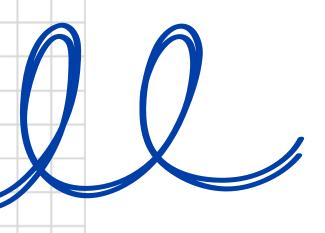
김준혁

- -프로젝트 검증
- -유닛 개발
- -문서 및 유닛 테스트



장성준

- -프로젝트 검증
- -유닛 개발
- -H/W 제작 및 통합 테스트





프로젝트 일정

구분	세부사항				
		26	27	28	29
요구사항 분석	요구사항 분석				
	요구사항 명세화				
시스템/아키텍 처 설계	요구사항에 따른 시 스템 아키텍처 설계				
	구조 설계				
	회로 연결도 설계				
	HW 사양서 설계				
S/W 설계 및 구	SW 설계				
현	SW 구현				
	SW 통합				
테스트	단위 테스트				
	통합 테스트				
	시스템 테스트				
최종 산출물	발표자료 작성				
	완료보고서 작성				
	시연동영상 제작				



3.프로젝트 수행 절차 및 방법

1.고객 요구사항	서(S/W,H/W)	2.용어통일	
3.시스템 요구/	사항 명세서	4.의뢰 시나리오	
5.구조 설계 &	State Flow	6.H/W Spec	
7.Pun Num	ber 설정	8.H/W 구성도	



고객 요구사항서(S/W)

중요도/긴급도를 나눈 기준

상: 기능이 작동이 안되면 목숨을 잃거나 재산에 큰 피해를 줄 수 있는 경도가 높은 경우 & 기초 주행 기능

중 : 주행기능 외 작동 되지 않으면 요구사항을 만족시킬 수 없는 기능

하: 기능이 작동 되지 않아도 시연에 문제가 없거나 목숨에

지장이 없는 기능

_	Req.id_SW	Summary	중요도/긴급도	요청자/일자
	1	키보드 혹은 스마트폰 등을 통해 RC카를 제어하며, R 키를 눌러 차량의 시동을 걸어야 한다.	상/상	김진호/23.02.26
	2	정해진 코스를 주행해야 한다.	중/중	김진호/23.02.26
	3	긴급 제동 구간을 통과하는 도중에는 임의의 장애물을 도중에 포함하여 긴급 제동 기능 확인 해야 한다.	상/상	김진호/23.02.26
_	4	AEB 작동 시 제동거리를 생각해야 한다.	상/상	김진호/23.02.26
	5	정차 지점에서 후진하여 주차 지점에 주차해야 하며, 후방 주차 시, 거리를 알려주는 각각 다른 부 저음을 줘야 한다.	중/중	김진호/23.02.26
	6	후방 주차 시 장애물에 닫기 전에 자동으로 멈춰야 한다.	중/상	김진호/23.02.26
	7	특정 버튼을 누르면, 주차자리를 탐색한 후 자동주차가 되어야 한다.	중/하	김진호/23.02.26
	8	정차 지점에는 전진 주차해야 한다. (RC카 정지)	하/하	김진호/23.02.26



추가고객 요구사항서(S/W)

Req.id_SW	Summary	중요도/긴급도	요청자/일자
9	예상치 못한 장애물이 좌/우 측에서 나타나면 회피기동을 실시한다.	하/상	김진호/23.02.26
10	좌/우 측에 나타나는 장애물은 좌/우측 초음파 센서를 통해 감지한다.	하/상	김진호/23.02.26



고객 요구사항서(H/W)

Req.id_HW	Summary	중요도/긴급도	요청자/일자
1	전방 AED 감지는 레이저 센서를 사용할 것.		김진호/23.02.26
2	후방 주차 센서는 초음파 센서를 사용할 것 .		김진호/23.02.26
3	TC_275를 사용할 것.	상/상	김진호/23.02.26
4	Motor Sheld 를 사용해서 Motor 를 제어할 것.		김진호/23.02.26
5	Easy Module 을 사용하여 부저를 사용할 것.		김진호/23.02.26
6	회피 기동 시 좌/우측에 초음파 센서를 장착해서 사용할 것.	하/하	김진호/23.02.26



용어통일

1.

Speed 1 : 오른쪽 모터의 출력Speed 2 : 왼쪽 모터의 출력

2.

duty 1 : 오른쪽 모터의 출력 과 방향duty 2 : 왼쪽 모터의 출력 과 방향

3.

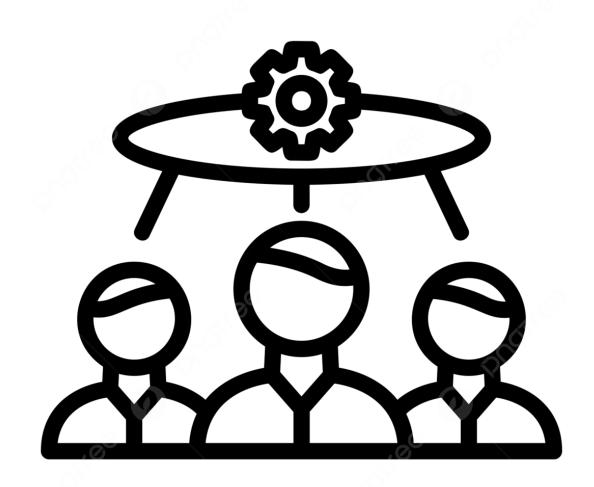
• drive : 시동 flag

• auto park : 자동주차 공간 찾는 flag

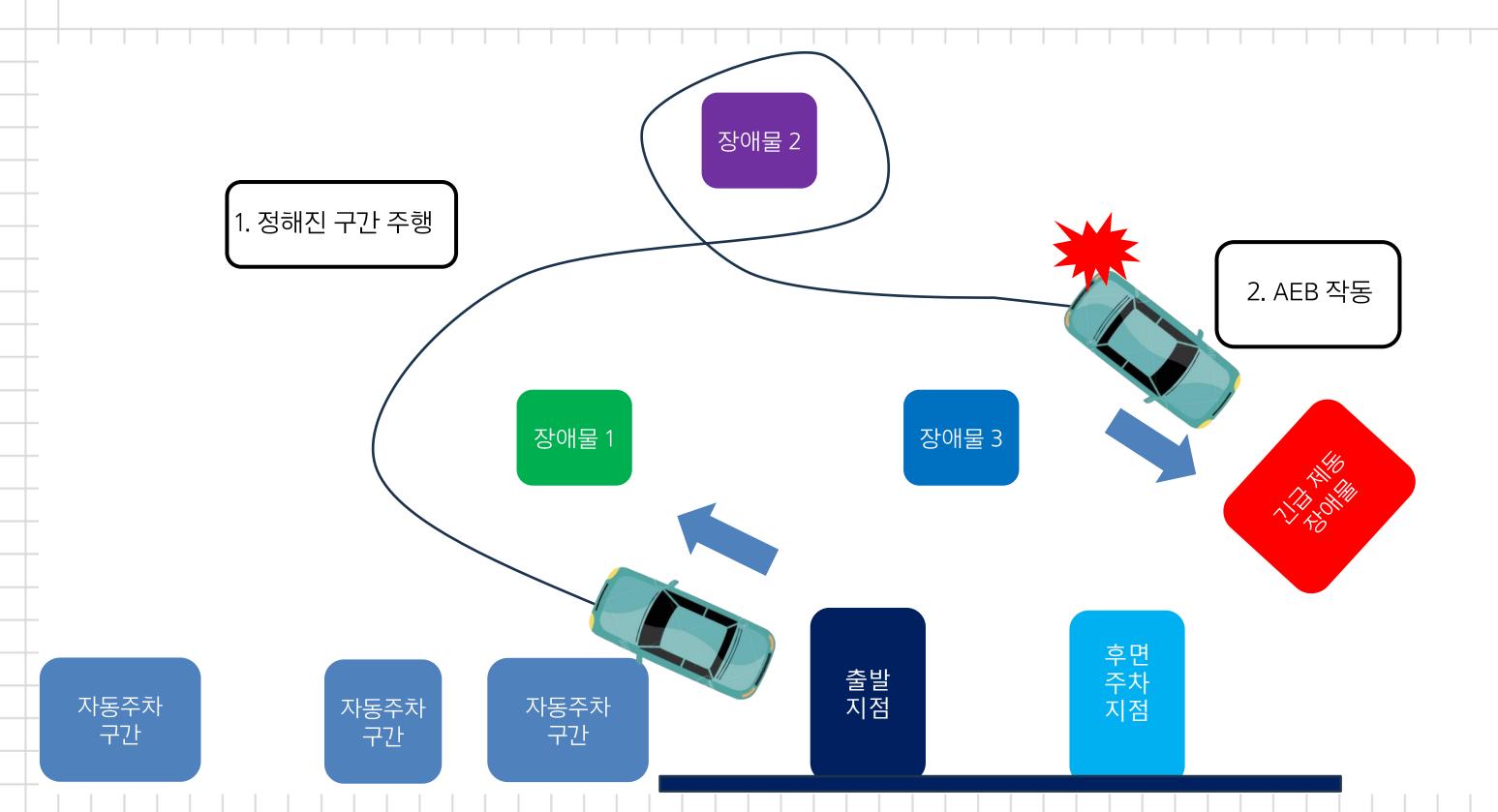
• do_parking : 자동주차 flag

• avoid flag : 회피기동 flag

• Buzzer flag : 부저 on/off flag









장애물 2

장애물 1

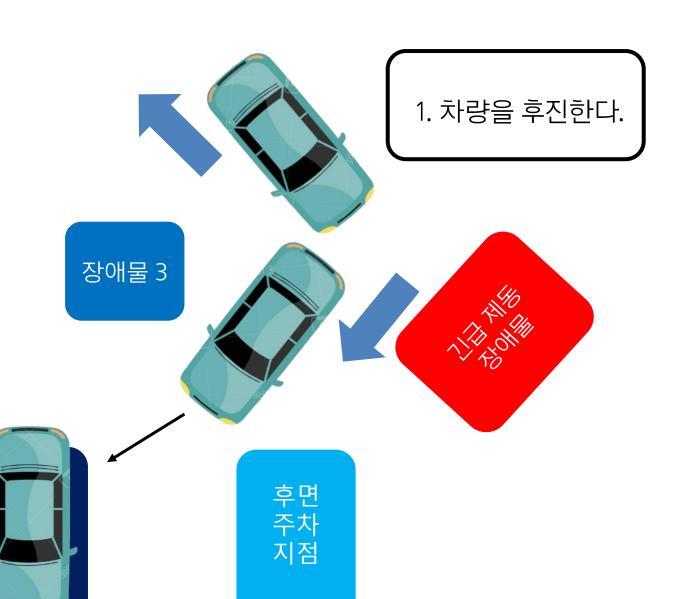
2. 차량을 출발지점 으로 전면 주차한다.

자동주차 구간

자동주차

구간

자동주차 구간





장애물 2

장애물 1

7 가능을 후진주차 구간에 주차한다.

지동주차 구간

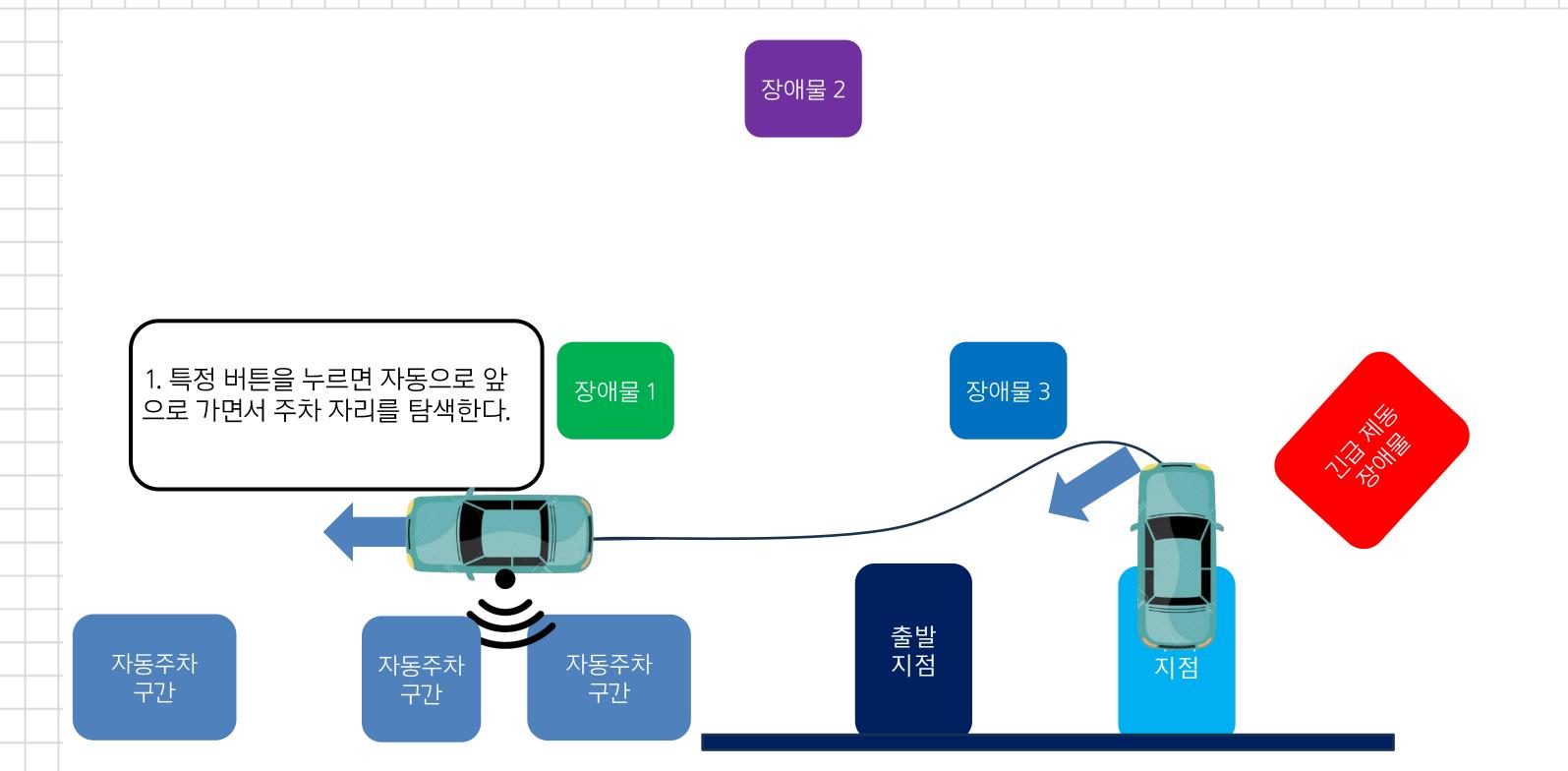
후방 주차 시, 초음파를 통해 거리를 측정하며, 거리를 알려주는 각각 다른 부저를 울린다.

자동주차 구간

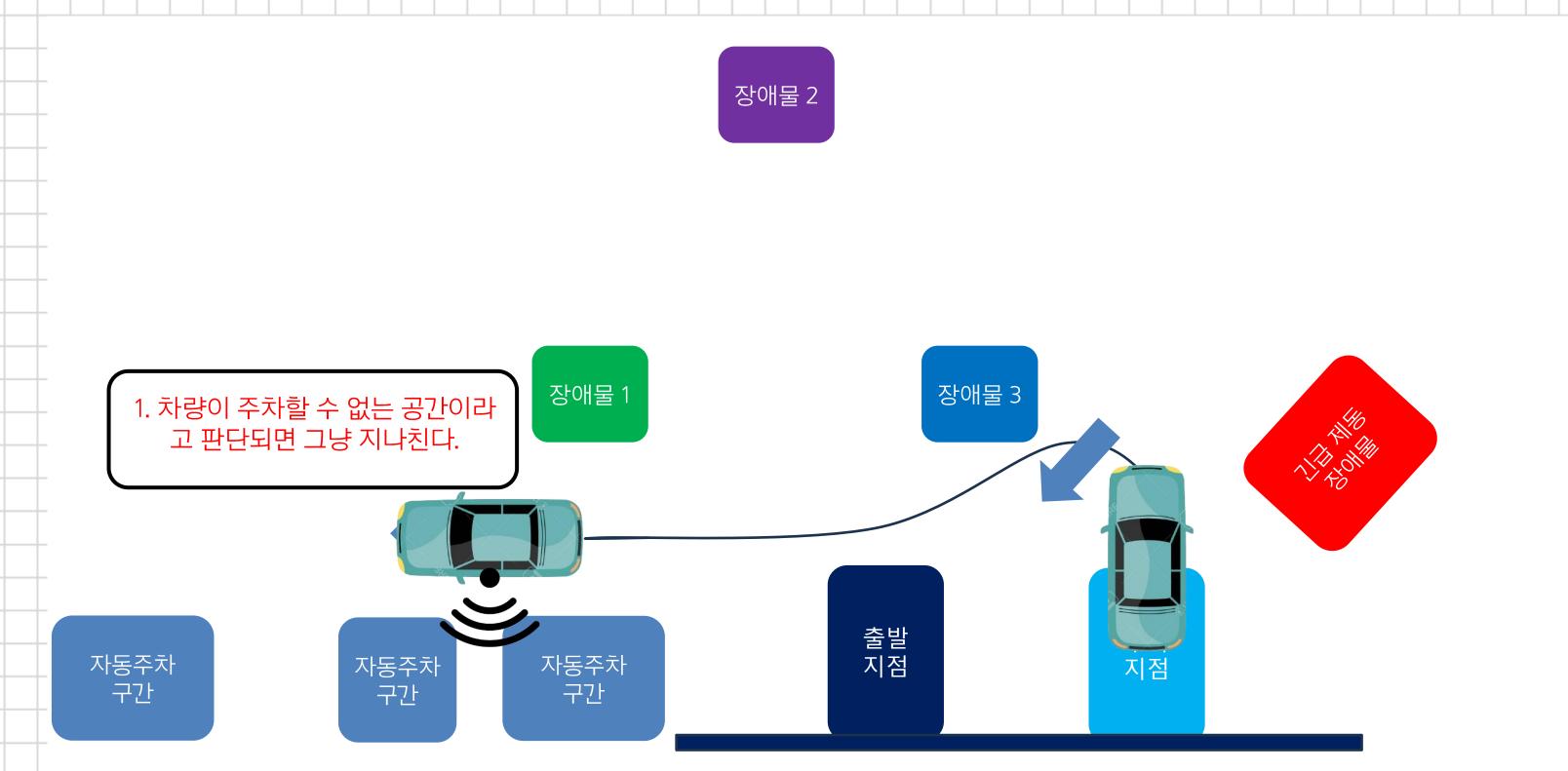
자동주차 구간













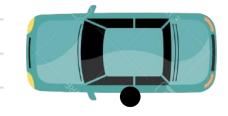
장애물 2

1. 차량이 주차할 수 있는 공간이라고 판단되면 Stop

장애물 1

장애물 3







탐지

자동주차 구간 자동주차 구간 출발 지점 후면 주차 지점



장애물 2

2. 자동 후진주차를 시작한다.

장애물 3



자동주차 구간

자동주차 구간

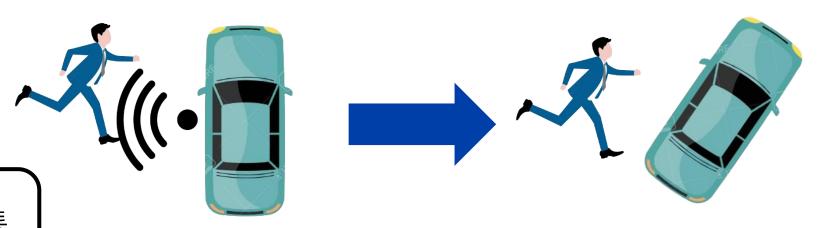
자동주차 구간

장애물 1

출발 지점 후면 주차 지점

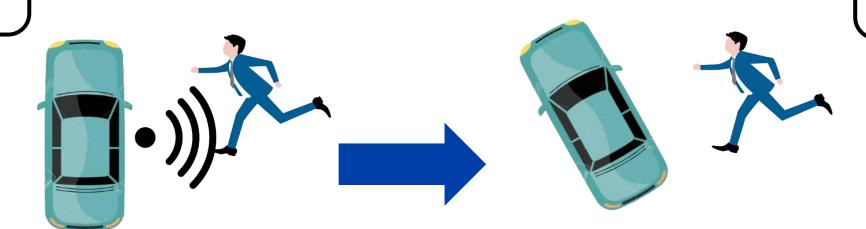


추가시나리오



1. 차량의 좌우에 초음파 센서를 통 해 예상치 못한 장애물 탐지

2. 차량을 좌/우로 회피기동





시스템 요구사항 명세서

Details .id	추적성 (Req.id)	분류	summary	검토자/일자	비고
1	Req.id_SW1 Req.id_SW2 Req.id_HW3 Req.id_HW4	하	키보드의 R 을 눌러 시동을 건 후, RC 카는 키보드를 통해 정해진 코스를 주행 한다.	황승혁/24.02.27	구현 가능
2	Req.id_SW8 Req.id_HW4	주차	정차 지점에는 W,A,S,D,Space bar 를 사용하여 전방주차를 해야한다. (RC카 정지)	황승혁/24.02.27	구현 가능
3	Req.id_SW5 Req.id_HW5 Req.id_HW2		정차 지점에서 후진하여 주차 지점에 주차해야 하며, 후방 주차 시, 초음파를 통해 거리를 측정하며, 거 리를 알려주는 각각 다른 부저를 울린다. 15cm 초과 33cm 이하 시 100cycle 로 울린다. 33cm 초과 45cm 이하 시 300cycle 로 울린다. 그 외에는 0cycle 로 울린다.	장성준/24.02.27	구현 가능
4	Req.id_SW6 Req.id_HW5		후방 주차 시 장애물에 닫기 전 15cm 전에 자동으로 멈춰야 한다.	황승혁/24.02.27	구현 가능
5	Req.id_SW7		키보드 버튼 'P'를 누르면, 차량이 들어갈 수 있는 공간을 초음파로 통해 Count를 하며, Count 가 멈추게 되면, 공간을 지나쳐 왔다고 판단, Count/2 만큼의 뒤로 거리로 이동한 후, 오른쪽 모터를 역방향으로 -5 식 17번, 왼쪽 모터를 정방향으로 +5 씩 17번을 곱해서 주차를 한다.	장성준/24.02.27	구현 가능
6	Req.id_SW3 Req.id_SW4 Req.id_HW1	제동	전방 장애물은 레이저 센서를 통해 거리를 측정하며, 긴급 제동 구간을 통과하는 도중에는 임의의 장애물을 도중에 포함하여 긴급 제동 거리는 차량의 속도를 고려해 27cm 에서 작동 되도록 한다.	김준혁/24.02.27	구현 가능

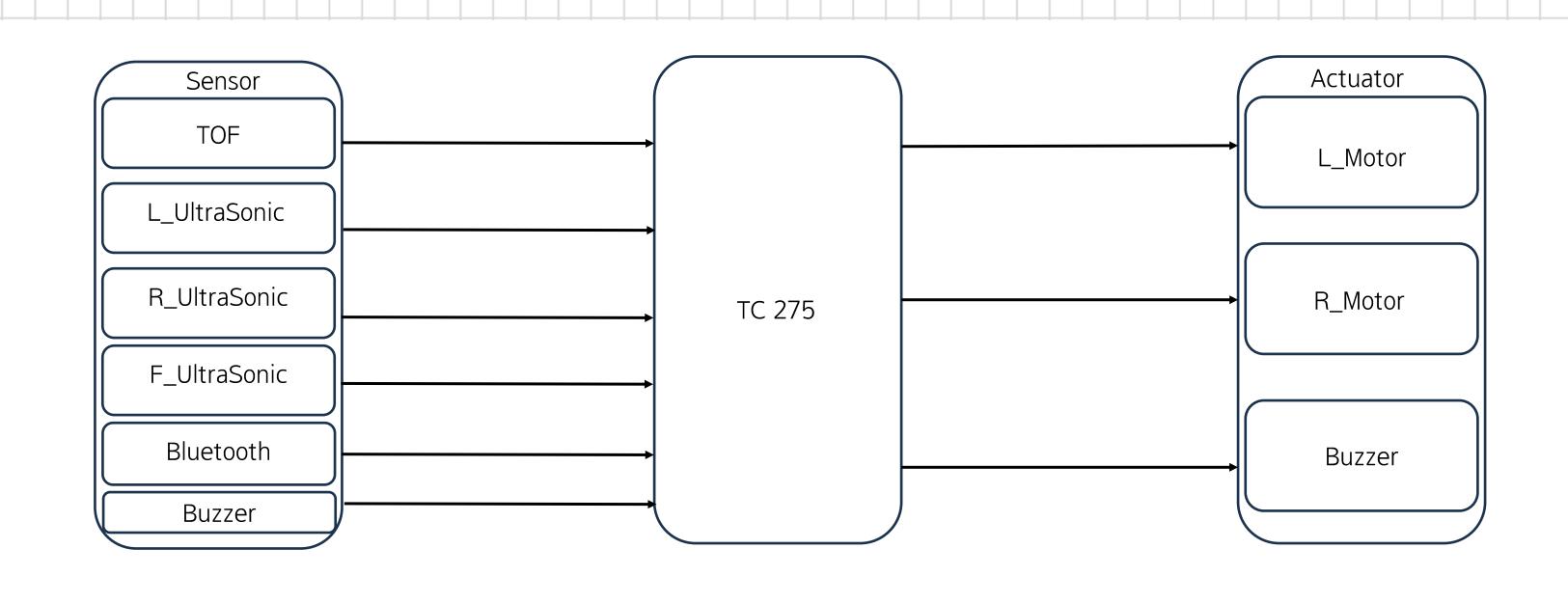


추가기능시스템 요구사항 명세서

Details .id	추적성 (Req.id)	분류	summary	검토자/일자	비고
7	Req.id_SW9 Req.id_SW10	회피	좌/우측에 초음파 센서에 장애물이 15cm 미만에 포착이 된다면, 우측에 장애물 탐지 시, 우측모터 출력을 80, 왼쪽 모터의 출력을 30으로 하며 좌측탐지 시 반대로 출력을 한다.	황승혁/24.02.27	구현 가능
8	Req.id_HW6		초음파 센서를 좌/우측에 장착한다.	황승혁/24.02.27	구현 가능



H/W 아케텍처



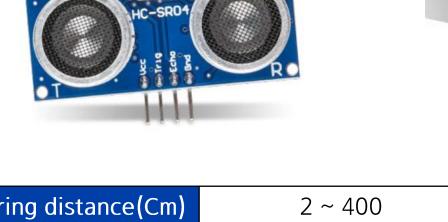


H/W 설계

Measuring distance(Cm)	2 ~ 400
Resolution(Cm)	0.3
Recommended operating voltage(V)	5~8









Core	TriCore TM 1.5 CPU
	Core
	32bit-Processor
	Max.freq 2000MHZ
Memory	CAN, UART, I/O
Communication Interface(Cm)	CAN,LIN ,Ethernet



Measuring distance(m)	0.01~5.00
Output method	CAN , UART , I/O
Precision(Cm)	±1.5

Measuring distance(Cm)	2 ~ 400
Resolution(Cm)	0.3
Effectual angle(°)	<15°



Laser 제원 검증

					_				
1	input	output	결과	비고	28	48	49	Р	
2	a self-ruled street(cm)	value of sensing(cm)	P/F	±1.5cm 의 오차는 인정한다.	29	49	50	Р	
3	30	31	P		30	50	51	Р	
4	31	32	Р		31	51	52	P	
5	32	33	Р		32	52	53	Р	
6	33	34	Р		33	53	54	Р	
7	34	35	Р		34	54	55	Р	
8	35	36	Р		35	55	56	Р	
9	36	37	Р		36	56	57	Р	
10	37	38	Р		37	57	58	P	
11	38	39	Р		38	58	59	Р	
12	38.5	39.5	Р		39	59	60	Р	
13	39	40	Р		40	60	61	Р	
14	39.5	40.5	Р		41	61	62	P	
15	40	41	Р		42	62	63	Р	
16	40.5	41.5	Р	저바 자에무 타지 시 자에무이	43	63	64	Р	
17	41	42	Р	전방 장애물 탐지 시 장애물의	44	64	65	P	
18	41.5	42.5	Р	거리에 맞게 제동거리를		65	66	P	
19	42	43	Р	조절하기 위하여 40 전후로 부	46	66	67	P	
20	42.5	43.5	Р	터는 0.5cm 증가하여 측정	47	67	68	P	
21	43	44	Р		48	68	69	P	
22	43.5	44.5	Р		49	69	70	P	
23		45	Р		50	70	71	P	
24	44.5	45.5	Р		50			•	
25	45	46	Р						
26	46	47	Р						
27		48	Р						



Ultrasonic Sensor 제원 검증

																			<u> </u>
	input		output		결과	비고													
	value(cm)	L_UltraSonic(cm)	R_Ultrasonic(cm)	B_Ultrasonic(cm)	P/F	±1.5cm 의 오차는 인정한다.	24	23	23.1	23.1	22.9	Р	43	42	41.9	41.9	41.9	Р	
	2	2.1	2.1	2	Р		25	24	24	24	24	Р	44	43	43.1	42.9	42.9	P	
	3	3.1	3.1	2.9	Р		26	25	25.09	24.9	24.917	Р	45	44	44	44	44	Р	
	4	4	4	4	Р		27	26	25.97	26.1	25.97	P	46	45	45	45.1	44.9	P	
	5	5	4.9	4.9	Р								47	46	45.97	45.9	45.9	Р	
	6	5.9	6.1	5.9	Р		28	27	27.1	27	26.91	Р	48	47	47.1	46.9	46.9	Р	
	7	7.1	7	6.91	Р		29	28	27.9	28.1	27.9	Р	49	48	47.9	48	47.9	Р	
		7.9	8.1	7.9	 Р		30	29	29.1	29.1	28.95	P	50	49	49.1	48.9	48.9	Р	
1		9.1	9.1	8.9	Р		31	30	29.9	29.9	29.93	Р							
1	1 10	9.9	9.9	9.9	Р		32	31	30.9	30.9	30.9	Р							
1	2 11	11.1	10.9	10.9	Р		33	32	31.9	31.9	32.1	Р							
1	3 12	12.1	11.9	12.1	Р		34	33	33.1	32.9	32.9	Р							
1	4 1 3	13.1	12.9	12.9	Р		35	34	34	34	34	Р							
1	5 14	14	14	14	Р		36	35	35	35.1	35.1	Р							
1	5 15	15.1	15.1	15.1	Р		37	36	35.97	35.9	36	P							
1	7 16	15.9	15.9	16	Р											_			
1	3 17	17.1	16.9	17	Р		38	37	37.1	36.9	37	Р							
1	9 18	18.1	18	17.9	Р		39	38	37.9	38	37.9	Р							
2	19	19	18.9	18.9	Р		40	39	39.1	38.9	38.9	Р							

40.1

40.9

40.9

41.9

40.1

41.9

20.1

20.1

21.9

20

20.1

20.1

20.1

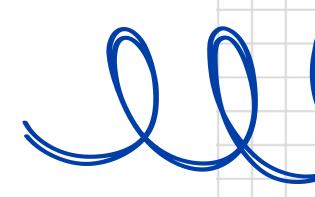
21.9

P



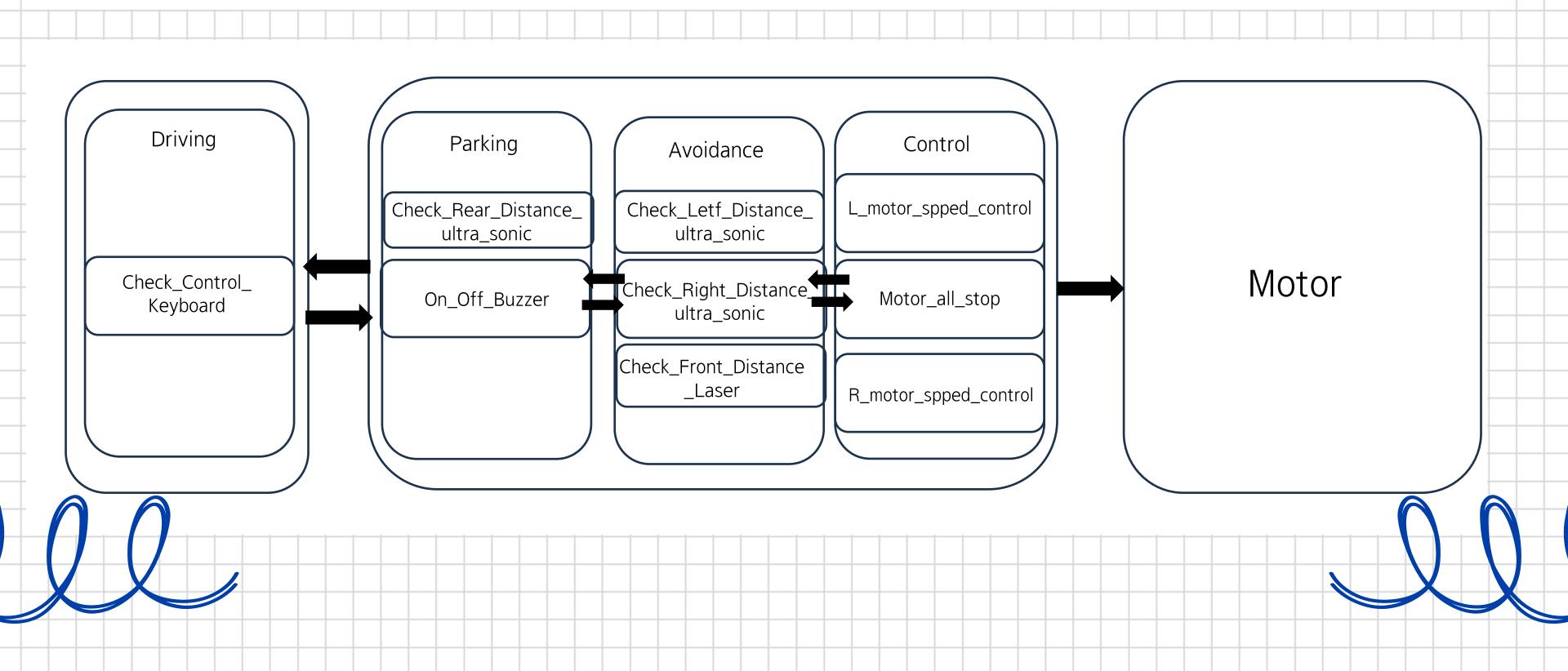
H/W 결선도

전방 레이저	Rx: 14(P15.0), Tx: 15(p15.1)
후방 초음파	Trigger : 25, echo : 27
우측 초음파	Trigger: 31, echo: 33
작측 초음파	Trigger : 37, echo : 39
부저	P02.3
블루투스 rx	Rx : 22, Tx : 23
PWM A _L	3
PWM B_R	11
Motor Dir A	12
Motor Dir B	13



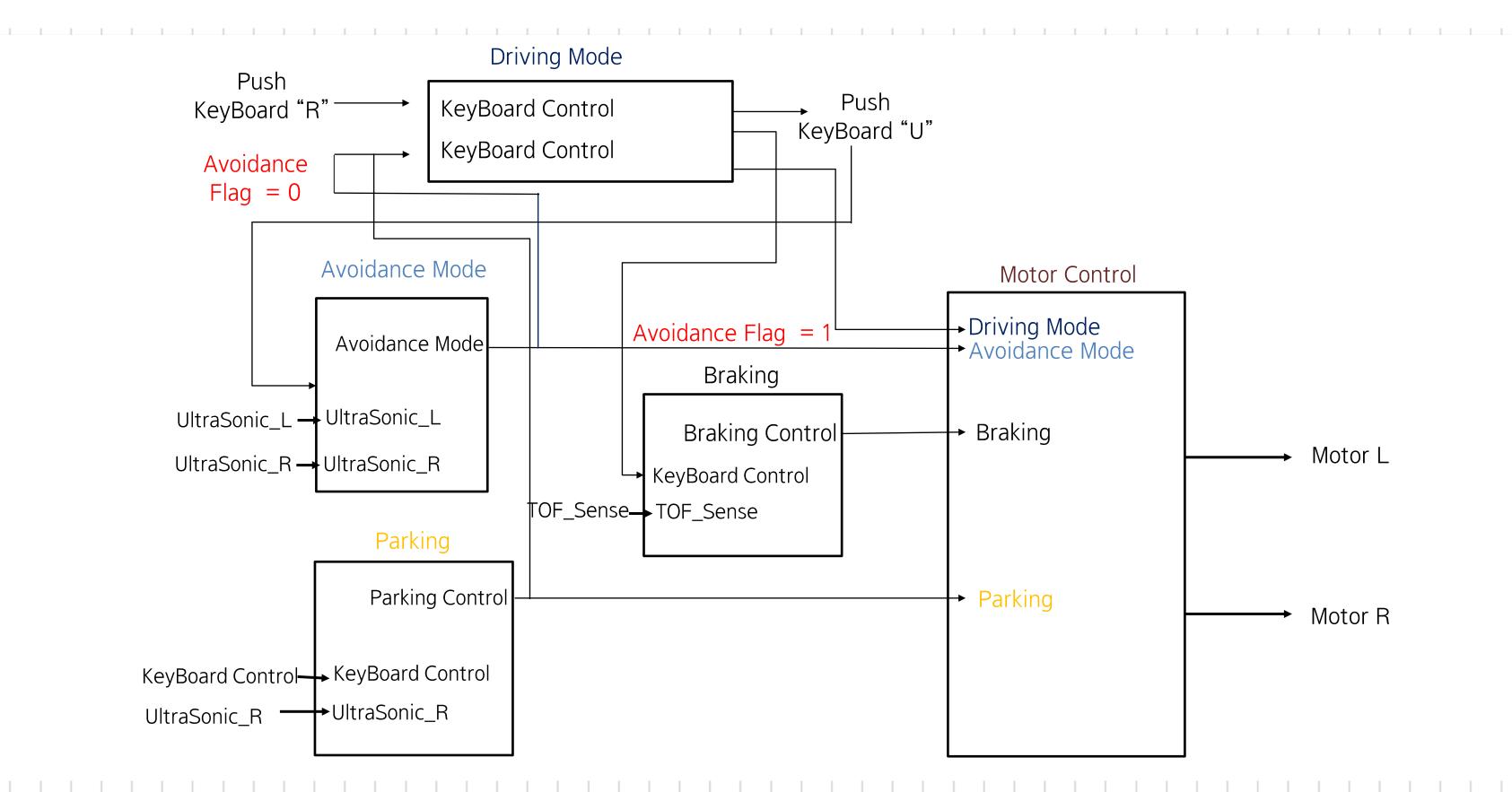


S/W 아케텍처





구조 설계





4. 프로젝트 구현

1.구현 별 Code & 순서도

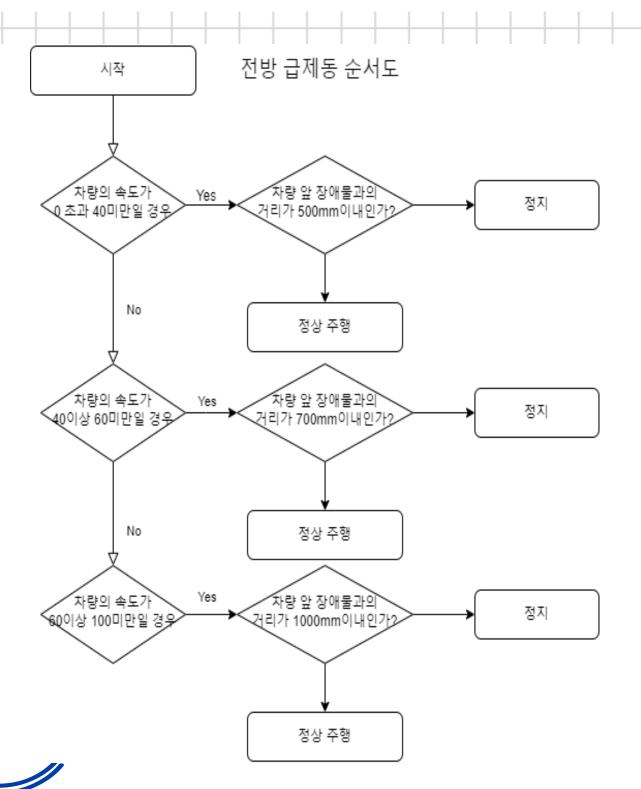
2. 각 모듈 별 구현

3. Unit Test Case & Test

4. Integrated & System Test case & Test



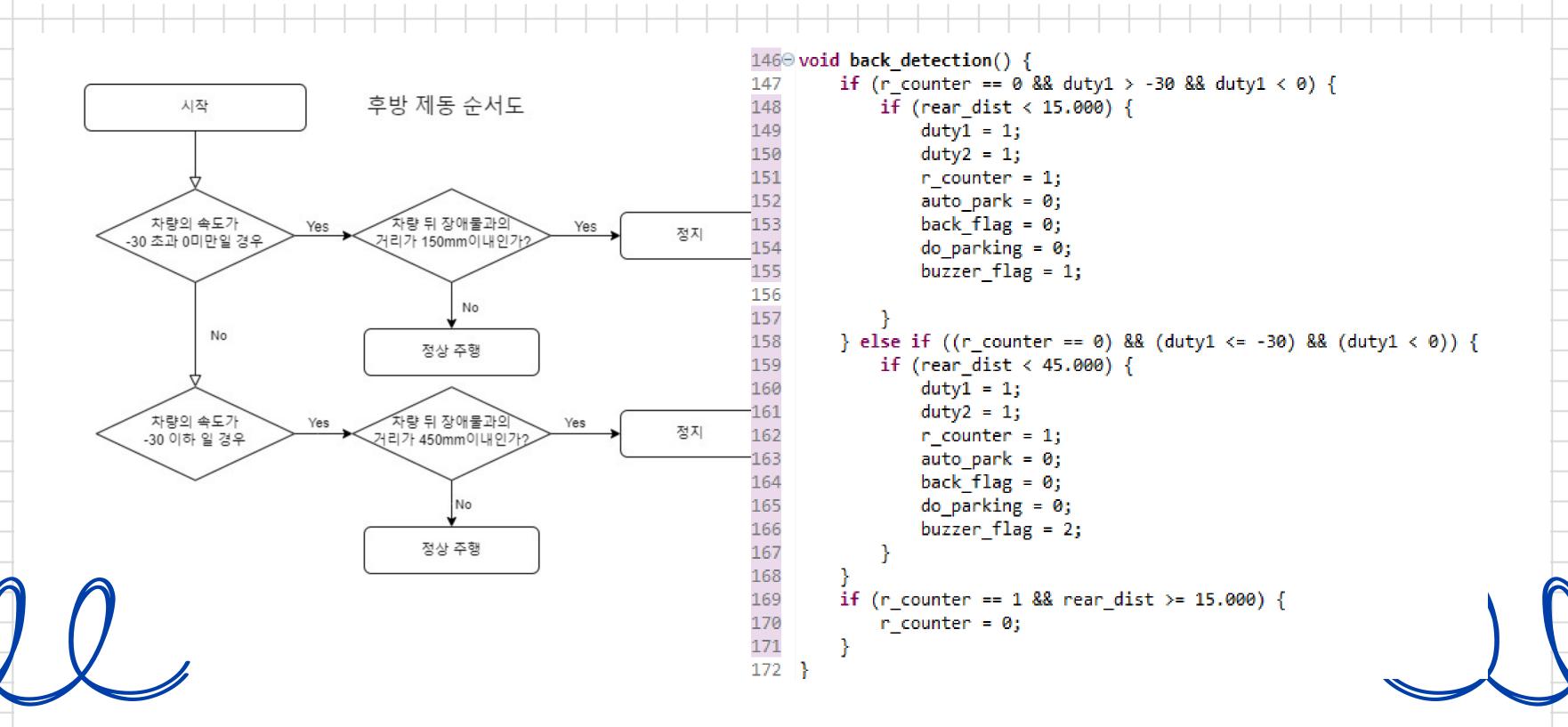
구현 front detection



```
174⊖ void front detection() {
        if (duty1 < 40 && duty1 > 0) {
            if (f counter == 0 && front_dist < 500 && front_dist != -2) //8cm 미만
                duty1 = 1;
                duty2 = 1;
                f counter = 1;
        } else if (duty1 < 60 && duty1 > 0) {
            if (f_counter == 0 && front_dist < 700 && front_dist != -2) //8cm 미만
                duty1 = 1;
                duty2 = 1;
                f_counter = 1;
        } else if (duty1 > 60 && duty1 > 0) {
            if (f_counter == 0 && front_dist < 1000 && front_dist != -2) //8cm 미만
                duty1 = 1;
                duty2 = 1;
                f counter = 1;
195
        if (f counter == 1 && front_dist >= 500) {
            f counter = 0;
200 }
```



구현 back detection

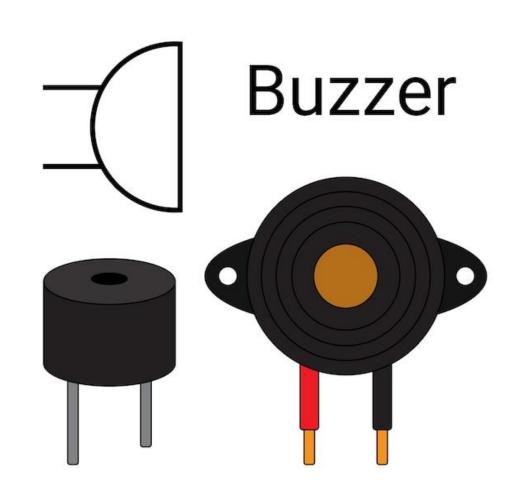


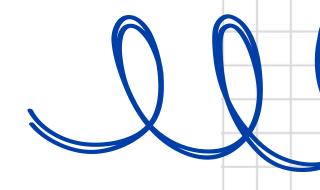


구현_buzzer setting

```
202⊖ void buzzer_setting() {
203
         if (drive == 1) {
204
              if (buzzer_flag > 0) {
205
206
                  setBeepCycle(1);
207
              } else {
                  if (duty1 > -30) {
208
209
                      if (rear_dist <= 33.00) {</pre>
210
                           setBeepCycle(100);
211
                      } else if (rear_dist <= 45.00) {
212
                           setBeepCycle(300);
213
                      } else {
214
                           setBeepCycle(0);
215
216
217
                  if (duty1 < -30) {
218
                      if (rear_dist <= 90.00) {</pre>
219
                           setBeepCycle(100);
220
                      } else if (rear_dist <= 135.00) {</pre>
221
222
                           setBeepCycle(300);
                      } else {
223
224
                           setBeepCycle(0);
225
226
227
228
```

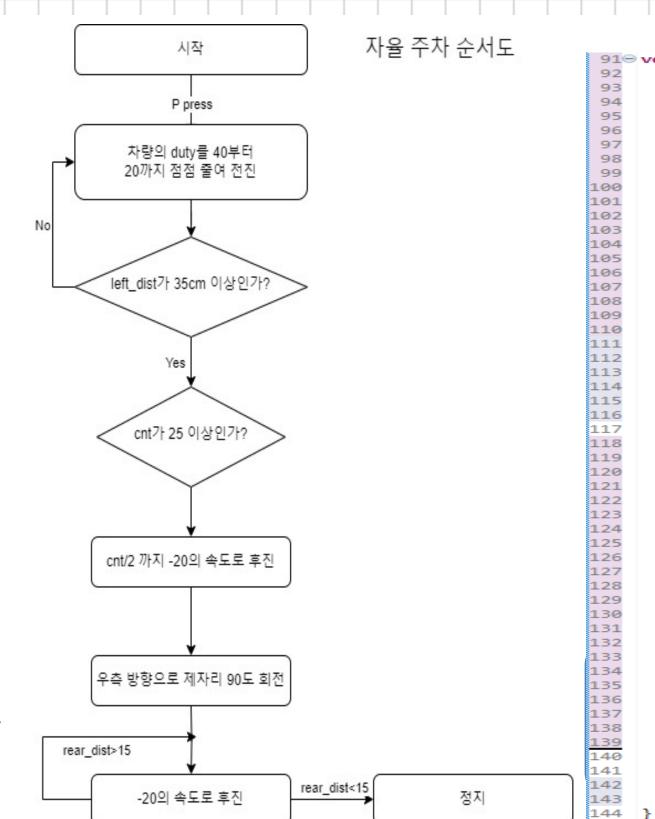
229







구현_Auto Parking



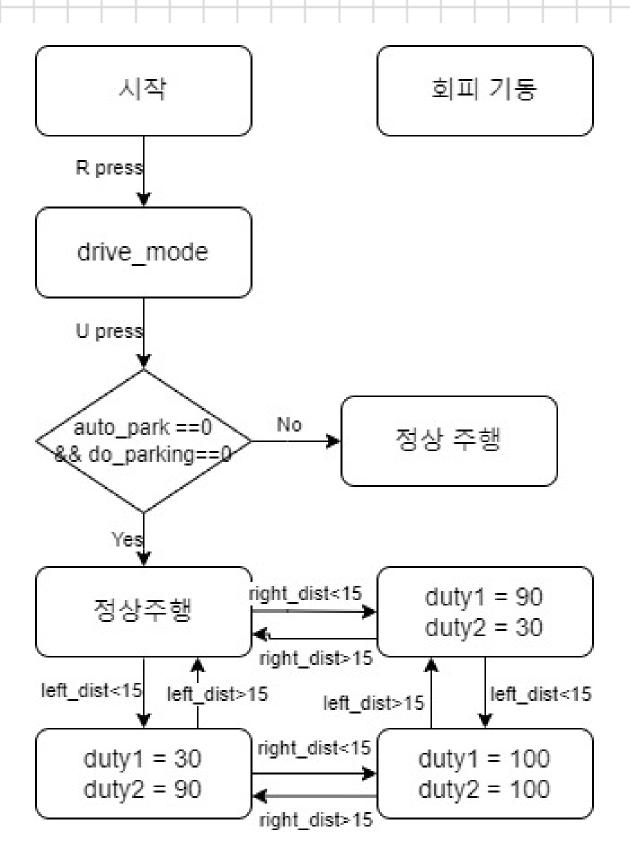
```
91⊖ void auto_parking() {
         if (auto_park == 1) { //자동주차 on
 93
             if (duty1 != 20) {
 94
                 duty1 -= 4;
                 duty2 -= 4;
 95
 96
 97
             if (left dist > 35.000) {
 98
                 cnt++;
 99
             } else if (left_dist < 35.000) {</pre>
100
                 if (cnt < 25) {
                      cnt = 0;
                 } else {
                      do_parking = 1;
                      auto_park = 0;
                      cnt2 = cnt;
                      duty1 = -36;
107
                      duty2 = -36;
108
109
110
111
         //주차가능시 수행
112
         if (do_parking == 1) {
113
             if (cnt > cnt2 / 2) {
114
                 if (duty1 < -20) {
115
                      duty1 += 2;
116
                      duty2 += 2;
117
118
                 cnt--;
119
             } else if (cnt == cnt2 / 2) {
120
                 cnt--;
121
                 duty1 = 0;
122
                 duty2 = 0;
123
             } else if (cnt > cnt2 / 4 && back flag == 0) {
                 duty1 = 45;
                 duty2 = 45;
126
                 movChA_PWM(duty1, 1);
127
                 movChB_PWM(duty2, 1);
128
                 delay ms(180);
                 back_flag = 1;
129
130
                 decrease_flag = 1;
131
             } else if (cnt > cnt2 / 4 && back_flag == 1) {
132
                 if (decrease_flag == 1) {
133
                      duty1 = -34;
134
                      duty2 = -34;
135
                      decrease_flag = 0;
136
                 } else {
137
                      if (duty1 < -20) {
138
                          duty1 += 2;
139
140
                          duty2 += 2;
141
                 }
142
```

동주차 구간



구현_avoid operation

266 }



```
231⊖ void avoid operation() {
         if (avoid flag == 1) {
232
233
             if ((right_dist < 15.000 || left_dist < 15.000) && auto_park == 0
                     && do parking == 0 && drive == 1) {
234
235
                 if (left_dist < 15.000 && right_dist < 15.000) {
236
                     speed1 = 100;
237
                     speed2 = 100;
238
                 } else if (right dist < 15.000) {
239
                     speed1 = 90;
240
                     speed2 = 30;
                     right emer cnt++;
241
242
                 } else if (left dist < 15.000) {
243
                     speed1 = 30;
244
                     speed2 = 90;
245
                     left emer cnt++;
246
247
                 emergency = 1;
248
249
250
             // 회피기동 복구
251
             if ((right dist >= 15.000 && left dist >= 15.000) && emergency == 1
252
                     && drive == 1) {
253
                 if (right emer cnt != 0) {
254
                     right emer cnt--;
255
                     speed1 = 30;
256
                     speed2 = 90;
257
                 } else if (left emer cnt != 0) {
258
                     left emer cnt--;
259
                     speed1 = 90;
260
                     speed2 = 30;
261
                 } else if (left emer cnt == 0 && right emer cnt == 0) {
262
263
                     emergency = 0;
264
265
```



Test Case_작성기준

기준

• 고객 요구사항에 중요도/긴급도가 최우선 기준으로 작성 및 테스트 진행.





Motor Unit Test Case & Test

1		input		output	결과	비고 26	28	28	28	28	P	
2	duty1(R)	duty2(L)	Speed1(R)	Speed2(L)	P/F	27		29	29	29	P	
3	5	5	5	5 Speed2(L)	P	28		30	30	30	P	
4			_	_	P	29		31	31	31	Р	
	6	6	6	6		30		32	32	32	Р	
5	7	7	7	7	P	31		33	33	33	Р	
6	8	8	8	8	P	32	34	34	34	34	Р	
7	9	9	9	9	P	33	35	35	35	35	Р	
8	10	10	10	10	Р	34		36	36	36	Р	
9	11	11	11	11	P	35	37	37	37	37	Р	최적의 모터 속도 구간
10	12	12	12	12	P	35 36	38	38	38	38	Р	
11	13	13	13	13	P	37	39	39	39	39	Р	
12	14	14	14	14	Р	38	40	40	40	40	Р	
13	15	15	15	15	Р	39	41	41	41	41	P	
14	16	16	16	16	Р	40	42	42	42	42	Р	
15	17	17	17	17	Р	41	43	43	43	43	Р	
16	18	18	18	18	Р	42	44	44	44	44	P	
17	19	19	19	19	P	43	45	45	45	45	P	
18	20	20	20	20	Р	44	46	46	46	46	Р	
19	21	21	21	21	P	45	47	47	47	47	P	
20						46	48	48	48	48	P	
	22	22	22	22	P	47		49	49	49	P	
21	23	23	23	23	P	48		50	50	50	P	
22	24	24	24	24	P	49	51	51	51	51	P	
23	25	25	25	25	P	50		52	52	52	P	
24	26	26	26	26	Р	51		53	53	53	P	
25	27	27	27	27	P	52		54	54	54	P	
26	28	28	28	28	P	53	55	55	55	55	P	
27	29	29	29	29	Р							



Keyboard + Motor Integrated Test Case & Test

— 1 <mark>i_i</mark>	d	trace	input	state		outpu			putty_outp		결과	19	17 req_1	W	86	86	90	90	90	90	1 front	P	37	35 req_1	S	34	34	38	38	38	38	1 back	P
2	-	-	keyboard	speed1 spe	eed2 sp	peed1 sp	eed2 spe	ed1 s	speed2 dri	ve direct	-	20	18 req_1	W	90	90	94	94	94	94	1 front	P	38	36 req_1	S	38	38	42	42	42	42	1 back	P
- 3		1 req_1	r	0	0	0	0	0	0	1 front	P	21	19 req_1	W	94	94	98	98	98	98	1 front	P	39	37 req_1	S	42	42	46	46	46	46	1 back	P
4		2 req_1	w(init)	0	0	30	30	30	30	1 front	P	22	20 req_1	W	98	98	100	100	100	100	1 front	P	40	38 req_1		46	46	50	50	50	50	1 back	Р
5	,	3 req_1	W	30	30	34	34	34	34	1 front	P	23	21 req_1	a	30	30	60	15	60	15	1 front	P	41	39 req_1		50	50	54	54	54	54	1 back	
- 6		4 req_1	W	34	34	38	38	38	38	1 front	P	24	22 req_1	a	34	34	68	17	60	15	1 front	P	42	40 req_1		30	30	60	15	60	15	1 back	
— 7		5 req_1	W	38	38	42	42	42	42	1 front	P	25	23 req_1	a	38	38	76	19	60	15	1 front	P	12	41 req_1		34	34	68	17	60	15	1 back	
- 8		6 req_1	W	4 2	42	46	46	46	46	1 front	P	26	24 req_1	a	42	42	84	21	60	15	1 front	P	43			38	38	76	19	60	15		
9		7 req_1	W	46	46	50	50	50	50	1 front	P	27	25 req_1	a	46	46	92	23	60	15	1 front	P	44	42 req_1								1 back	
10		8 req_1	W	50	50	54	54	54	54	1 front	P	28	26 req_1	a	50	50	100	25	60	15	1 front	P	45	43 req_1		42	42	84	21	60	15	1 back	
- 11		9 req_1	W	54	54	58	58	58	58	1 front	P	29	27 req_1	d	30	30	15	60	15	60	1 front	P	46	44 req_1		46	46	92	23	60	15	1 back	
12	1	0 req_1	W	58	58	62	62	62	62	1 front	P	30	28 req_1	d	34	34	17	68	17	68	1 front	P	47	45 req_1		50	50	100	25	60	15	1 back	
13		1 req_1	W	62	62	66	66	66	66	1 front	Р	31	29 req_1	d	38	38	19	76	19	76	1 front	Р	48	46 req_1	d	30	30	15	60	15	60	1 back	Р
14			W	66	66	70	70	70	70	1 front	Р	32	30 req_1	d	42	42	21	84	21	84	1 front	Р	49	47 req_1	d	34	34	17	68	17	68	1 back	P
15		3 req_1	W	70	70	74	74	74	74	1 front	Р	33	31 req_1	d	46	46	23	92	23	92	1 front	Р	50	48 req_1	d	38	38	19	76	19	76	1 back	P
16		4 req_1	W	74	74	78	78	78	78	1 front	Р	34	32 req_1	d	50	50	25	100	25	100	1 front	Р	51	49 req_1	d	42	42	21	84	21	84	1 back	P
17		5 req_1	W	78	78	82	82	82	82	1 front		35	33 req_1	s(init)	0	0	30	30	30	30	1 back	Р	52	50 req_1	d	46	46	23	92	23	92	1 back	P
18		6 req 1_		82	82	86	86	86	86	1 front		36	34 req_1		30	30	34	34	34	34	1 back	Р	53	51 req_1	d	50	50	25	100	25	100	1 back	P
										.,											'			,,_									



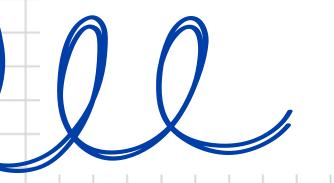
Motor + R_Ultrasonic + Buzzer Integrated Test Case & Test

1	i_id	trace	in	put	sta	ate	out	tput			결과
2	-	-	keyboard	ultrasonic	speed1	speed2	speed1	speed2	direct	buzzer	
3	1	req_5	r	15	0	0	0	0	front	off	Р
4	2	req_5	s	15	30	30	0	0	back	on	Р
5	3	req_5	s	15	34	34	0	0	back	on	Р
6	4	req_5	s	45	38	38	0	0	back	on	P
7	5	req_5	s	45	42	42	0	0	back	on	P
8	6	req_5	s	45	46	46	0	0	back	on	P
9	7	req_5	S	45	50	50	0	0	back	on	Р
10	8	req_5	a	15	30	30	0	0	back	on	Р
11	9	req_5	a	15	34	34	0	0	back	on	Р
12	10	req_5	a	45	38	38	0	0	back	on	Р
13	11	req_5	a	45	42	42	0	0	back	on	Р
14	12	req_5	a	45	46	46	0	0	back	on	Р
15	13	req_5	a	45	50	50	0	0	back	on	Р
16	14	req_5	d	15	30	30	0	0	back	on	Р
17	15	req_5	d	15	34	34	0	0	back	on	Р
18	16	req_5	d	45	38	38	0	0	back	on	Р
19	17	req_5	d	45	42	42	0	0	back	on	Р
20	18	req_5	d	45	46	46	0	0	back	on	Р
21	19	req_5	d	45	50	50	0	0	back	on	Р



Motor + Laser senser Integrated Test Case & Test

1	i_id	trace		put		state			tput	결과
2	_	-	keyboard	lazer	speed1	speed2		speed1	speed2	-
3	1	req_3	r	500	0	0	front	0	0	P
4	2	req_3	w	500	30	30	front	0	0	P
5	3	req_3	w	500	34	34	front	0	0	P
6	4	req_3	w	500	38	38	front	0	0	P
7	5	req_3	w	700	42	42	front	0	0	P
8	6	req_3	w	700	46	46	front	0	0	P
9	7	req_3	w	700	50	50	front	0	0	P
10	8	req_3	w	700	54	54	front	0	0	P
11	9	req_3	w	700	58	58	front	0	0	P
12	10	req_3	w	1000	62		front	0	0	P
13	11	req_3	w	1000	66		front	0	0	Р
14	12	req_3	w	1000	70		front	0	0	Р
15	13	req_3	w	1000	74		front	0		P
16	14	req_3	w	1000	78	78	front	0	0	P
17	15	req_3	w	1000	82	82	front	0	0	P
18	16	req_3	w	1000	86	86	front	0	0	P
19	17	req_3	w	1000	90	90	front	0	0	P
20	18	req_3	w	1000	94	94	front	0	0	P
21	19	req_3	w	1000	98	98	front	0	0	P
22	20	req_3	a	1000	30	30	front	0	0	P
23		req_3	а	1000	34	34	front	0	0	P
24	22	req_3	а	1000	38	38	front	0	0	P
25	23	req_3	а	1000	42	42	front	0	0	P
26	24	req_3	а	1000	46	46	front	0	0	P
27	25	req_3	а	1000	50	50	front	0		P
28	26	req_3	d	1000	30	30	front	0	0	P
29	27	req_3	d	1000	34	34	front	0	0	Р
30	28	req_3	d	1000	38	38	front	0	0	Р
31	29	req_3	d	1000	42	42	front	0	0	P
32	30	req_3	d	1000	46	46	front	0	0	P
33	31	req_3	d	1000	50	50	front	0	0	P





Motor + Automatic parking Integrated Case & Test

4				거기
1	i_id	trace	summary	결과
2	1	req_7	전진하면서 왼쪽 초음파센서로 주차공간을 센싱한다.	P
3	2	req_7	주차공간 센싱이 완료된 후 부족하다고 판단하면 다음으로 이동한다.	P
4	3	req_7	주차공간 센싱이 완료된 후 충분하다고 판단하면 정지한다.	P
5	4	req_7	정지한 후 센싱한 거리의 반을 후진한다.	Р
6	5	req_7	후진이 완료된 후 차량을 오른쪽으로 90도 회전한다.	Р
7	6	req_7	회전이 완료 후 후진하여 주차를 시작한다.	Р
8	7	req_7	천천히 후진하며 후방 초음파 센서의 값을 센싱한다.	Р
9	8	req_7	후방주차 시 후방 초음파의 값이 일정값 이하이면 정지한다.	Р
			후방 초음파 센서의 값에 따라 부저를 다르게 울린다.	
		7	1. 45cm이하일 경우 부저는 300ms 주기로 울린다.	Ь
	9	req_7	2. 33cm이하일 경우 부저는 100ms 주기로 울린다.	Р
10			3. 15cm이하일 경우 부저는 1ms 주기로 울린다.	
11	10	req_7	자동주차가 완료된 후 키보드를 통해 다시 RC카를 제어한다.	Р



Motor + Avoidance Integrated Test Case & Test

i_id	trace		input		stat	te	outpu	t		putty_out	put	결과	54	52 req_9,10	w	14	14	74	74	78	78	100	100	1 front	Р
-		keyboard	r_ultrasensor l_ultras	sensor sp								-	55	53 req_9,10		14	14	78	78	82	82	100	100	1 front	Р
	1 reg 9,10	r	14	0	0	0	0	0	0	0	1 front	Р	56	54 req_9,10		14	14	82	82	86	86	100	100	1 front	Р
	2 req_9,10		14	0	0	0	30	30	90	30	1 front	P	57	55 req_9,10	w	14	14	86	86	90	90	100	100	1 front	Р
	3 req_9,10		14	0	30	30	34	34	90	30	1 front	P	58	56 req_9,10	w	14	14	90	90	94	94	100	100	1 front	Р
	4 req_9,10		14	0	34	34	38	38	90	30	1 front	Р	59	57 req_9,10	w	14	14	94	94	98	98	100	100	1 front	Р
	5 req_9,10		14	0	38	38	42	42	90	30	1 front	P	60	58 req_9,10	w	14	14	98	98	100	100	100	100	1 front	P
	6 req_9,10		14	0	42	42	46	46	90	30	1 front	P P	61	59 req_9,10	a	14	1 0	30	30	60	15	90	30	1 front	P
	7 req_9,10 8 req_9,10		14 14	0	46 50	46 50	50 54	50 54	90	30	1 front 1 front	P	62	60 req_9,10	a	14	1 0	34	34	68	17	90	30	1 front	P
	9 req_9,10		14	0	54	54	58	58	90 90	30 30	1 front	P	63	61 req_9,10	a	14	1 0	38	38	76	19	90	30	1 front	Р
	10 req_9,10		14	0	58	58	62	62	90	30	1 front	P	64	62 req_9,10	а	14	1 0	42	42	84	21	90	30	1 front	P
	11 req_9,10		14	0	62	62	66	66	90	30	1 front	P	65	63 req_9,10	a	14	1 0	46	46	92	23	90	30	1 front	
	12 req_9,10		14	0	66	66	70	70	90	30	1 front	P	66	64 req_9,10	а	14	1 0	50	50	100	25	90	30	1 front	P
	13 req_9,10		14	0	70	70	74	74	90	30	1 front	P	67	65 req_9,10		(14	30	30	60	15	30	90	1 front	P
	14 req_9,10	w	14	0	74	74	78	78	90	30	1 front	Р	68	66 req_9,10		(14	34	34	68	17	30	90	1 front	Р
	15 req_9,10	w	14	0	78	78	82	82	90	30	1 front	P	69	67 req_9,10			14	38	38	76	19	30	90	1 front	Р
	16 req_9,10	w	14	0	82	82	86	86	90	30	1 front	P	70	68 req_9,10		(42	42	84	21	30	90	1 front	Р
	17 req_9,10		14	0	86	86	90	90	90	30	1 front	P	71	69 req_9,10			14	46	46	92	23	30	90	1 front	Р
	18 req_9,10		14	0	90	90	94	94	90	30	1 front	Р	72	70 req_9,10		(50	50	100	25	30	90	1 front	Р
	19 req_9,10		14	0	94	94	98	98	90	30	1 front	P	73	71 req_9,10		14		30	30	60	15	100	100	1 front	Р
	20 req_9,10		14	0	98	98	100	100	90	30	1 front	P	74	72 req_9,10		14		34	34	68	17	100	100	1 front	Р
	21 req_9,10		0	14 14	30	30	30 34	30 34	30 30	90 90	1 front 1 front	P	75	73 req_9,10		14		38	38	76	19	100	100	1 front	P
	22 req_9,10 23 req_9,10		0	14	34	34	38	38	30	90	1 front	P	76	74 req_9,10		14		42	42	84	21	100	100	1 front	P
	24 req_9,10		0	14	38	38	42	42	30	90	1 front	P	77	75 req_9,10		14		46	46	92	23	100	100	1 front	P
	25 req_9,10		0	14	42	42	46	46	30	90	1 front	P	78	76 req_9,10		14		50	50	100	25	100	100	1 front	Р
	26 req_9,10		0	14	46	46	50	50	30	90	1 front	P	79	77 req_9,10		14			30	15	60	90	30	1 front	P P
	27 req_9,10		0	14	50	50	54	54	30	90	1 front	Р	80	78 req_9,10		14			34	17	68	90	30	1 front	
	28 req_9,10		О	14	54	54	58	58	30	90	1 front	P	81	79 req_9,10	d	14	1 0	38	38	19	76	90	30	1 front	Р
	29 req_9,10		О	14	58	58	62	62	30	90	1 front	Р	82	80 req_9,10	d	14	1 0	42	42	21	84	90	30	1 front	P
	30 req_9,10		0	14 14	62	62 66	66	66	30	90	1 front	P	0.2						46				20		D
	31 req_9,10 32 req_9,10		0	14	66 70	70	70 74	70 74	30 30	90 90	1 front 1 front	P	83	81 req_9,10		14	1 0	40	46	23	92	90	30	1 front	
	33 req_9,10		0	14	74	74	78	78	30	90	1 front	P	84	82 req_9,10	d	14	1 0	50	50	25	100	90	30	1 front	P
	34 req_9,10		0	14	78	78	82	82	30	90	1 front	Р	85	83 req_9,10	Ч		14	30	30	15	60	30	90	1 front	Р
	35 req_9,10	w	0	14	82	82	86	86	30	90	1 front	Р		-											
	36 req_9,10		0	14	86	86	90	90	30	90	1 front	P	86	84 req_9,10	d	() 14	34	34	17	68	30	90	1 front	Р
	37 req_9,10		0	14	90	90	94	94	30	90	1 front	P	87	85 req_9,10	d		14	38	38	19	76	30	90	1 front	Р
	38 req_9,10 39 req_9,10		0	14 14	94 98	94 98	98 100	98 100	30 30	90 90	1 front 1 front	P		-									00		Р
	40 req_9,10		14	14	0	0	30	30	100	100	1 front	P	88	86 req_9,10) 14	42	42	21	84	30	90	1 front	Ρ
	41 req_9,10		14	14	30	30	34	34	100	100	1 front	P	89	87 req_9,10	d		14	46	46	23	92	30	90	1 front	Р
	42 req_9,10		14	14	34	34	38	38	100	100	1 front	P	90	88 req_9,10	Ч	(14	50	50	25	100	30	90	1 front	Р
	43 req_9,10		14	14	38	38	42	42	100	100	1 front	Р													
	44 req_9,10		14	14	42	42	46	46	100	100	1 front	P	91	89 req_9,10	d	14	14	30	30	15	60	100	100	1 front	Р
	45 req_9,10		14	14	46	46	50	50	100	100	1 front	P	92	90 req_9,10	d	14	14	34	34	17	68	100	100	1 front	Р
	46 req_9,10 47 req_9,10		14 14	14 14	50 54	50 54	54 58	54 58	100 100	100 100	1 front 1 front	P													
	48 req_9,10		14	14	58	58	62	62	100	100	1 front	P	93	91 req_9,10		14	14	38	38	19	76	100	100		
	49 req_9,10		14	14	62	62	66	66	100	100	1 front	P	94	92 req_9,10	d	14	14	42	42	21	84	100	100	1 front	Р
	50 req_9,10		14	14	66	66	70	70	100	100	1 front	Р				14			46			100	100	1 front	
	51 req_9,10		14	14	70	70	74	74	100	100	1 front	Р	95	93 req_9,10						23	92				
													96	94 req_9,10	d	14	14	50	50	25	100	100	100	1 front	Р



Kyungshin

System Test Case & Test

Details.id		Category	summary	P/F							
_ 1	Req.id_SW1 Req.id_SW2 Req.id_HW3 Req.id_HW4	주행	키보드의 R 을 눌러 시동을 건 후, RC카는 키보드를 통해 정해진 코스를 주행한다.	Р	6	Req.id_SW3 Req.id_SW4	제동	전방 장애물은 레이저 센서를 통해 거리를 측정하며, 긴급 제동 구간을 통과하는 도중에는 임의의 장애물을 도중에 포함하여 긴급 제동 거리는	P		
_ 2	Req.id_SW8 Req.id_HW4		정차 지점에는 W,A,S,D,Space bar 를 사용하여 전방주차를 해야한다. (RC카 정지)	Р		Req.id_HW1		차량의 속도를 고려해 27cm 에서 작동 되도록 한다.			
3	Req.id_SW5 Req.id_HW5 Req.id_HW2		정차 지점에서 후진하여 주차 지점에 주차해야 하며, 후방 주차 시, 초음파를 통해 거리를 측정하며, 거리를 알려주는 각각 다른 부저를 울린다. 15cm 초과 33cm 이하 시 100cycle 로 울린다. 33cm 초과 45cm 이하 시 300cycle 로 울린다. 그 외에는 0cycle 로 울린다.	Р	7	Req.id_SW9 Req.id_SW10	회피	좌/우측에 초음파 센서에 장애물이 15cm 미만에 포착이 된다면, 우측에 장애물 탐지 시, 우측모터 출력을 80, 왼쪽 모터의 출력을 30으로 하며 좌측탐지 시 반대로 출력을 한다.	P		
		주차				Req.id_HW6		초음파 센서를 좌/우측에 장착한다.			
– 4	Req.id_SW6 Req.id_HW5		후방 주차 시 장애물에 닫기 전 15cm 전에 자동으로 멈춰야 한다.	Р	8	neq.id_rivvo			Р		
			키보드 버튼 'P'를 누르면, 차량이 들어갈 수 있는 공간을 초음파로 통해 Count를 하며,								

Count 가 멈추게 되면, 공간을 지나쳐 왔다고 판단, Count/2 만큼의 뒤로 거리로 이동한 후,

오른쪽 모터를 역방향으로 -5 식 17번, 왼쪽 모터를 정방향으로 +5 씩 17번을 곱해서 주차를 한다.

Req.id_SW7



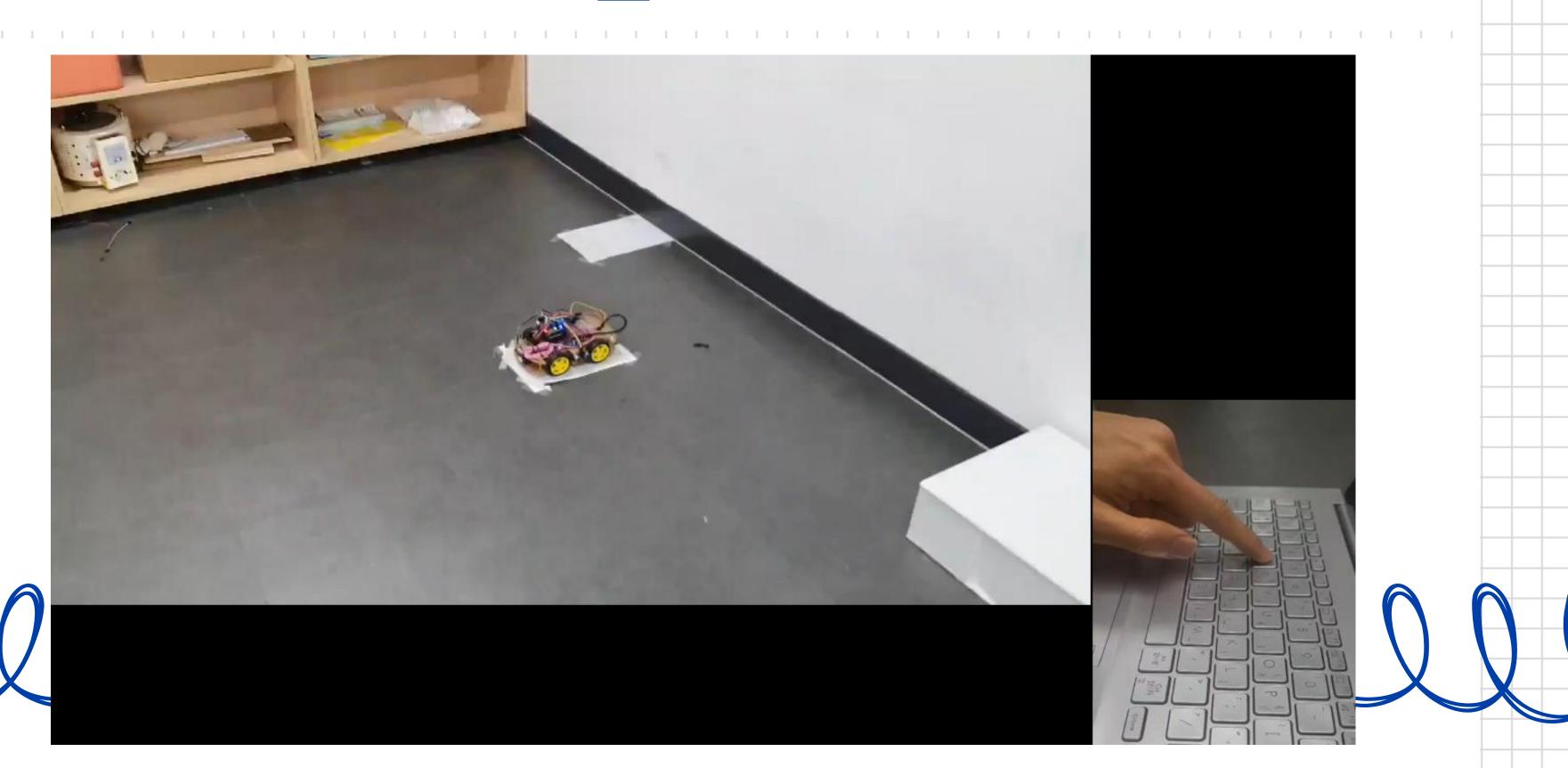
5.데모영상

1.데모 영상

2.추가 요구사항 데모 영상

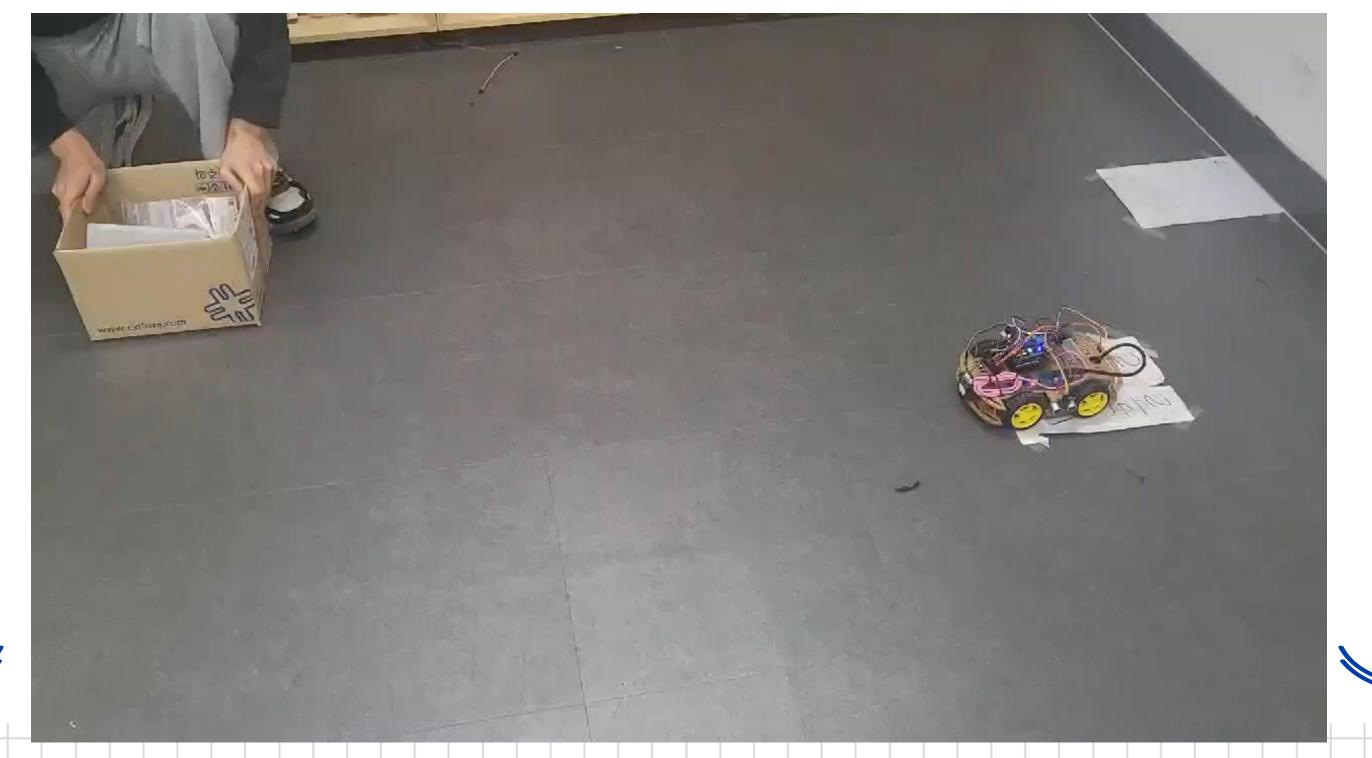


데모영상_기본요구사항





데모 영상_추가 요구사항(회피기동)







1.자체 의견 평가



자체 의견 평가

• 팀원들끼리 역할 분담이 잘 이루어지고 소통이 잘 되어서, 짧은 시간안에 프로젝트를 성공적으로 끝낼 수 있었다.

• 총 4번의 프로젝트를 통해 점점 더 V 모델에 익숙해 짐을 느꼈다. 주행 연습 공간이 협소하여서
 연습주행을 많이 못해본 것이 아쉽다.