

학사학위 논문

지도교수 윤 의 중

사각지대 사고율을 줄이기 위한 안전기술

Safety technique to reduce accident rates in
blind spots

2022

호 서 대 학 교 자 동 차 I C T 공 학 과

자 동 차 I C T 공 학 과

임 호 균

사각지대 사고율을 줄이기 위한 안전기술

Safety technique to reduce accident rates in
blind spots

이 논문을 학사학위 논문으로 제출함.

2022년 12월

호 서 대 학 교 자 동 차 I C T 공 학 과

자 동 차 I C T 공 학 과

임 호 균

심사위원 윤의중 인

2022년 12월

호 서 대 학 교 자 동 차 I C T 공 학 과

목 차

I. 서론	1
II. 본론	1
1. 대형차량 사각지대	1
2. 사용 부품	2
3. 회로도 설계	3
4. 하드웨어 구성	3
5. 소프트웨어의 프로그래밍	3
III. 실험 결과	4
1. 성능 테스트	4
IV. 결론 및 향후연구	4
참고문헌	4
<부록 1> 아두이노 소스코드	5

사각지대 사고율을 줄이기 위한 안전기술

Safety technique to reduce accident rates in blind spots.

임호균

호서대학교 자동차ICT공학과

(S.N: 20192412, Mobile: 010-2449-6793 E-mail: kshihk1007@naver.com)

지도교수: 윤의중 (호서대학교 자동차ICT 공학과)

요 약

현재 많은 자동차 안전 기술의 발전에도 불구하고 대형 차량 사각지대 사고가 빈번하게 발생하고 있다. 고비용으로 장착하지 못하거나 장착이 된 경우일 경우 휴먼에러로 인한 사고가 이어지고 있다. 본 논문은 사각지대 차량이 진입할 경우 외부와 내부에 경고 장치를 부착해 이중으로 경고하는 장치이다. 초음파센서를 이용해 거리 연산 후 네오픽셀, 피에조 부저, LCD를 출력해 경고할 것이다.

I. 서론

현재 자동차 안전 기술의 많은 발전에도 불구하고, 대형차량 사각지대 사고는 여전히 발생하고 있다. 고비용으로 장착하지 않거나 기본으로 장착이 된 차량이라도 휴먼에러로 인한 사고가 발생하고 있다.

본 논문은 사각지대 차량이 진입할 경우 운전자에게만 경고하는 것이 아닌 사각지대에 진입한 운전자에게 불빛 신호와 소리 신호를 이용해 경고하는 장치다. 또한 운전자에게는 LCD를 이용해 사각지대 안에 차량이 진입한 것을 경고한다. 즉 이중으로 경고하는 장치이므로 사고율을 줄일 수 있다 예상된다.

II. 본론

2.1 대형차량 사각지대



그림 01. 자동차 사각지대

대형 차량의 경우 일반 차량에 비해 2배 이상 크기가 크므로 사각지대 또한 많게는 일반 차량에 비해 3배가량 많은 것을 알 수 있다.

대형 차량의 전방 사각지대의 경우 5t 화물차 기준 2m정도의 사각지대가 존재하며, 좌측 사각지대는 운전석 기준 30° 정도의 사각지대가 존재하는 것을 알 수 있다. 우측 사각지대는 운전석 기준 40° 정도의 사각지대가 존재하는 것을 알 수 있다.

2.2 하드웨어 구성

가. 초음파센서

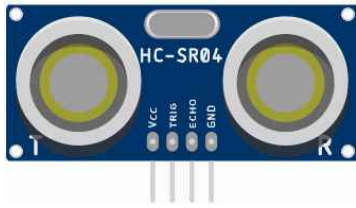


그림 02. HC-SR04 초음파센서

HC- SR04 초음파센서는 약 42kHz 소리의 주파수를 발사하여 돌아오는 값을 계산하여 거리를 구한다. 작동전압은 D C5V, 작동전류는 2.2mA이며 감지 범위는 2cm ~ 400cm까지 측정할 수 있다.

거리를 계산하는 방법은 초음파의 속도 340m/s이므로 초음파의 이동 거리는 속도 * 이동시간이며, 실제 물체까지의 거리는 1/2이므로 물체의 거리를 구할 수 있다.

$$\text{물체까지 거리} = \frac{341m/s \times \text{측정시간}}{2}$$

나. LCD 1602*2(I2C)



그림 03. LCD 1602*2(I2C)

LCD(Liquid Crystal Display)의 원리는 고체 표면에 액정이 특정 방향으로 정렬할 수도 있어 전기적 신호를 통해 원하는 방향으로 빛을 투과시키거나 차단시키는 원리이다.

I2C 통신은 데이터를 주고받기 위한 선 하나와 송수신 타이밍 동기화를 위한 클럭선(CSL) 하나로 이루어져 있으며, 하나의 마스터와 N개의 슬레이브로 이루어져 있다.

다. 네오픽셀(NeoPixel)

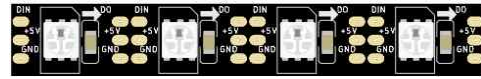


그림 04. 네오픽셀(NeoPixel)

네오픽셀(NeoPixel)이란 WS2811, WS 2812 등과 같이 칩이 내장된 LED를 의미합니다. LED 하나에서 원하는 색을 만들기 위해 R, G, B에 전류량을 제어해 원하는 색을 만든다. 작동전류는 R, G, B 각각 20mA를 사용하며 첫 번째 LED를 보호하기 위해 300~500Ω 사용한다.

연결 방법은 캐스케이드 방법(Cascade method)으로 연결해 준다.

라. 피에조 부저(Piezo Buzzer)



그림 05. 피에조 부저(Piezo Buzzer)

피에조 효과를 이용한 소자로 인가된 전압에 의해 진동시켜 음을 재생한다. 자기 손실이 적고 효율적으로 전기 에너지를 소리로 변환한다.

2.3 회로도 설계

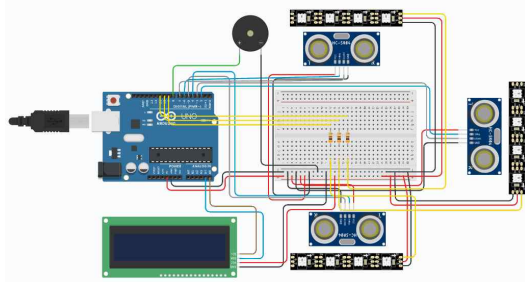


그림 06. 회로도

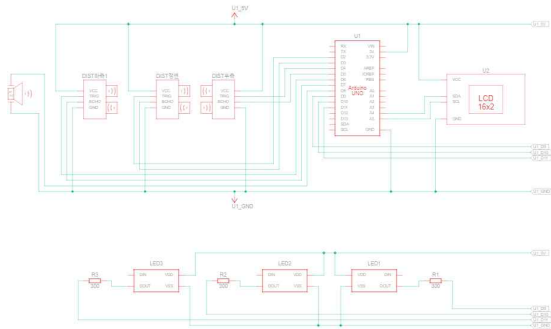


그림 07. 회로도

초음파센서의 연결은 Trig-(2, 4, 6), Echo-(3, 5, 7), Vcc-(5V), GND-(GND)로 연결한다.

LCD(Liquid Crystal Display)의 연결은 SDA-(A4), SCL-(A5), VCC-(5V), GND-(GND)로 연결 한다.

네오픽셀(NeoPixel)의 연결은 DO-(9, 10, 11), VCC(5V), GND-(GND)로 연결 한다.

피에조 부저(Piezo Buzzer)의 연결은 VCC-(8), GND-(GND)로 연결 한다.

2.4 하드웨어 구현

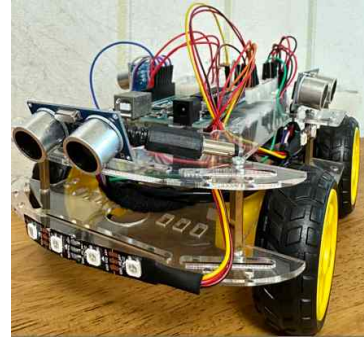


그림 08. 센서 장착

회로도를 바탕으로 상단에 LCD(Liquid Crystal Display)와 피에조 부저(Piezo Buzzer)를 설치했으며, 초음파센서 3개와 3개의 네오픽셀(NeoPixel)을 각각 정면, 좌측, 우측에 설치하였다.

2.5 소프트웨어의 프로그래밍

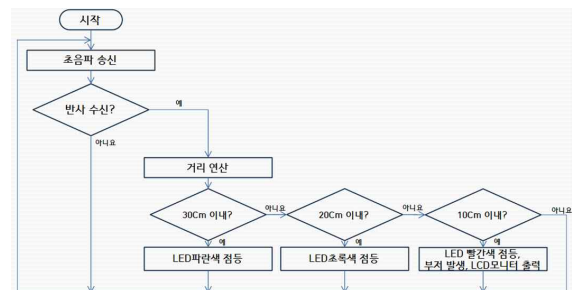


그림 09. 순서도

초음파센서에서 반사된 수신을 통해 거리연산을 하며, 연산을 통해 30cm 이내일 경우 파란색을 출력, 20cm 이내일 경우 초록색을 출력, 10cm 이내일 경우 빨간색을 출력하며 부저, LCD에 출력해 운전자에게 알린다.

III. 실험 결과

3.1 성능 테스트

거리	네오픽셀	LCD	부저
40cm	○	○	○
30cm	●	○	○
20cm	●	○	○
10cm	●	●	●

표 01. 차량감지에 대한 변화

표01과 같이 차량이 감지 될 경우 네오픽셀의 경우 40cm를 제외한 30, 20, 10cm에서 점등되는 것을 볼 수 있다. LCD의 경우 10cm에만 출력하여 운전자에게 사각지대 안에 차량이 있다는 신호를 알린다. 부저 또한 10cm에만 출력하여 사각지대 안에 차량이 있다는 것을 출력한다.

IV. 결론 및 향후 연구

현재 경고 장치는 운전자에게만 경고하는 장치로 사각지대에 진입한 운전자는 자신이 사각지대에 들어왔다는 사실을 모른다. 이중으로 경고하는 본 작품을 통해 현재 장치의 부족한 점을 보완하여 사고율을 줄일 수 있다 생각한다. 실제 차량에 적용 시 초음파의 거리와 각도의 한계만 넘어선다면 저렴한 비용으로 초음파센서를 이용해 감지할 수 있음을 확인하였다.

현재 사용 하고 있는 HC-SR04 초음파센서의 최대 감지 범위인 4m는 실제 차량의 사각지대를 감지하기에는 짧은 것을 확인할 수 있다. 실제 적용 한다면 또 다른 센서를 이용해 최적의 센서를 찾아봐야 한다.

참고문헌

- [1] 유동석. (2020). “블루투스를 이용한 자동차 사각지대 감지 장치 개발” (국내석사학위논문). 아주대학교 pp. 27-29
- [2] 강갑생. “트럭 ‘아이 못봤다’ 이유 있었다…참변 부른 '죽음의 사각지대'” 중앙일보. 2021. 3면 (<https://www.joongang.co.kr/article/25031993#home>)
- [3] 신나는 금요일의 블로그. “네오픽셀에 대해 자세히 알아봅시다.”(<https://blog.naver.com/yunc26/222920215659>), 2022.12.08
- [4] 코딩피플 “아두이노로 I2C LCD 다양하게 제어하기” (<https://www.youtube.com/watch?v=rHWgC55VhiQ&t=8s>). 2022.12.08.

<부록 1> 아두이노 소스코드

```
#include <Wire.h>

#include <hd44780.h>

#include <hd44780ioClass/hd44780_I2Cexp.
h>

#include <Adafruit_NeoPixel.h>

int NUMPIXELS1 = 4;

int PIXELPIN1 = 9;

int NUMPIXELS2 = 4;

int PIXELPIN2 = 10;

int NUMPIXELS3 = 4;

int PIXELPIN3 = 11;

int trigPin1 = 2;

int echoPin1 = 3;

int trigPin2 = 4;

int echoPin2 = 5;

int trigPin3 = 6;

int echoPin3 = 7;

int tonePin = 8;

int freq = 600;

int dur1 = 300, dur2 = 300;

long duration1, distance1;

long duration2, distance2;

long duration3, distance3;

long duration4, distance4;
```

```
Adafruit_NeoPixel pixels1(NUMPIXELS1,
PIXELPIN1, NEO_GRB + NEO_KHZ800);

Adafruit_NeoPixel pixels2(NUMPIXELS2,
PIXELPIN2, NEO_GRB + NEO_KHZ800);

Adafruit_NeoPixel pixels3(NUMPIXELS3,
PIXELPIN3, NEO_GRB + NEO_KHZ800);

hd44780_I2Cexp lcd;

void setup() {
    lcd.begin(16, 2);
    pixels1.begin();
    pixels2.begin();
    pixels3.begin();
    pinMode(trigPin1, OUTPUT);
    pinMode(echoPin1, INPUT);
    pinMode(trigPin2, OUTPUT);
    pinMode(echoPin2, INPUT);
    pinMode(trigPin3, OUTPUT);
    pinMode(echoPin3, INPUT);
    Serial.begin(9600);
}

void loop() {
    //-----1(정면)

    digitalWrite(trigPin1, LOW);
    delayMicroseconds(2);
    digitalWrite(trigPin1, HIGH);
    delayMicroseconds(10);
```

```

digitalWrite(trigPin1, LOW);
duration1 = pulseIn(echoPin1, HIGH);
distance1= duration1*0.034/2;

if (distance1 < 10) {
  for(int i=0; i<NUMPIXELS1; i++){
    pixels1.setPixelColor(i, 255, 0, 0);
  }
  pixels1.show();
  delay(100);

  tone(tonePin, freq, dur1);
  delay(dur2);
  noTone(tonePin);
  lcd.clear();
  lcd.setCursor(5,0);
  lcd.print("Front");
  lcd.setCursor(4,1);
  lcd.print("Warning!");
  delay(1);
}
else if (distance1 <20){
  for(int i=0; i<NUMPIXELS1; i++){
    pixels1.setPixelColor(i, 0, 255, 0);
  }
  pixels1.show();
  delay(100);

}
else if (distance1 < 30){

```

```

  for(int i=0; i<NUMPIXELS1; i++){
    pixels1.setPixelColor(i, 0, 0, 255);
  }
  pixels1.show();
  delay(100);
}
else{
  for(int i=0; i<NUMPIXELS1; i++){
    pixels1.setPixelColor(i, 0, 0, 0);
  }
  pixels1.show();
  delay(100);
  lcd.clear();
}

if (distance1 >= 500 || distance1 <= 0){
  Serial.println("Out of range");
}
else {
  Serial.print ( "Sensor1 : ");
  Serial.print ( distance1);
  Serial.println("cm");
}
delay(20);

```

<pre>//-----2(우측) digitalWrite(trigPin2, LOW); delayMicroseconds(2); digitalWrite(trigPin2, HIGH); delayMicroseconds(10); digitalWrite(trigPin2, LOW); duration2 = pulseIn(echoPin2, HIGH); distance2= duration2*0.034/2; if (distance2 < 10) { for(int ii=0; ii<NUMPIXELS2; ii+){ pixels2.setPixelColor(ii, 255, 0, 0); } pixels2.show(); delay(100); tone(tonePin, freq, dur1); delay(dur2); noTone(tonePin); lcd.clear(); lcd.setCursor(3,0); lcd.print("RightSide"); lcd.setCursor(4,1); lcd.print("Warning!"); delay(1); } else if (distance2 <20){ for(int ii=0; ii<NUMPIXELS2; ii+){ pixels2.setPixelColor(ii, 0, 255, 0); } </pre>	<pre> pixels2.show(); delay(100); } else if (distance2 < 30){ for(int ii=0; ii<NUMPIXELS2; ii+){ pixels2.setPixelColor(ii, 0, 0, 255); } pixels2.show(); delay(100); } else{ for(int ii=0; ii<NUMPIXELS1; ii+){ pixels2.setPixelColor(ii, 0, 0, 0); } pixels2.show(); delay(100); lcd.clear(); } if (distance2 >= 500 distance2 <= 0){ Serial.println("Out of range"); } else { Serial.print("Sensor2 : "); Serial.print(distance2); Serial.println("cm"); } delay(20); </pre>
--	---

<pre>//-----3(좌측) digitalWrite(trigPin3, LOW); delayMicroseconds(2); digitalWrite(trigPin3, HIGH); delayMicroseconds(10); digitalWrite(trigPin3, LOW); duration3 = pulseIn(echoPin3, HIGH); distance3= duration3*0.034/2; if (distance3 < 10) { for(int iii=0; iii<NUMPIXELS3; iii++){ pixels3.setPixelColor(iii, 255, 0, 0); } pixels3.show(); delay(100); tone(tonePin, freq, dur1); delay(dur2); noTone(tonePin); lcd.clear(); lcd.setCursor(4,0); lcd.print("LeftSide"); lcd.setCursor(4,1); lcd.print("Warning!"); delay(1); } else if (distance3 <20){ for(int iii=0; iii<NUMPIXELS3; iii++){ pixels3.setPixelColor(iii, 0, 255, 0); } pixels3.show();</pre>	<pre> delay(100); } else if (distance3 < 30){ for(int iii=0; iii<NUMPIXELS3; iii++){ pixels3.setPixelColor(iii, 0, 0, 255); } pixels3.show(); delay(100); } else{ for(int iii=0; iii<NUMPIXELS1; iii++){ pixels3.setPixelColor(iii, 0, 0, 0); } pixels3.show(); delay(100); lcd.clear(); } if (distance3 >= 500 distance3 <= 0){ Serial.println("Out of range"); } else { Serial.print("Sensor3 : "); Serial.print(distance3); Serial.println("cm"); } delay(20); }</pre>
---	--