**대중교통 비접촉 생체인식 및 자동 발열감지 시스템 결과보고서**

**이름:** 전채린

|  |
| --- |
| **1. 개발목표와 특징** |

*(각 팀에서 개발하고자 하는 제안시스템의 특징을 반영한 개발목표를 기술)*

|  |
| --- |
| 버스나 지하철과 같은 대중교통에서 비접촉으로 생체를 인식하고 온도를 측정하는 자동 발열 감지 시스템을 아두이노로 개발하려고 한다. 초음파센서 모듈을 이용해 사람을 인식하고, 적외선 온도센서를 이용해 비접촉으로 온도를 측정하여 발열을 검사한다. 그리고 시리얼 포트와 I2C LCD 디스플레이를 이용해 화면에 측정 온도와 정상체온 여부를 보여주고, LED를 이용해 측정 온도에 따라 저체온은 노란색, 정상체온은 녹색, 고체온은 빨간색으로 나타낸다. 만약 측정 온도가 고체온이면 수동부저를 이용해 삐 소리를 출력하고 발열 출입자로 인식해 출입을 거부하는 시스템이다. 또한, 출입자의 출입을 실시간으로 기록하여 출입 인원을 알 수 있게 구현하려고 한다. |

|  |
| --- |
| **2. 개발필요성** |

*(왜 본 시스템 개발이 필요한지 기술적 환경 및 수요 배경을 중심으로 기술)*

|  |
| --- |
| 코로나가 심각한 지금 버스나 지하철과 같은 대중교통 이용 시 비접촉으로 자동 발열 감지와 생체를 인식하는 시스템을 아두이노로 개발하려고 한다. 대규모 상업시설, 학교, 교육기관, 건설 현장, 병원, 사무실, 공항 등 다양한 곳에선 이미 비접촉 자동발열 감지 시스템을 이용하고 있다. 그러나 버스와 지하철과 같은 대규모의 사람들이 이용하는 대중교통에서는 아직 비접촉으로 생체를 인식하고 온도를 체크하는 자동 발열 감지 시스템이 설치되지 않았다. 대중교통을 이용하는 대규모의 사람들을 빠르고 정확하게 인식하고 온도를 측정하는 것이 어렵기 때문이다. 코로나가 심화되는 상황에서 많은 사람이 이용하는 대중교통은 코로나와 같은 바이러스에 감염되기 쉽다. 많은 사람이 바이러스의 공포에 불안해하는 시기에 발열 체크와 같은 예방은 꼭 필요하다. 그러나 사람이 일일이 대중교통을 이용하는 많은 사람의 체온을 직접 측정하는 것은 거의 불가능하다고 볼 수 있다. 그렇기에 빠르고 정확하게 사람을 인식하고 비접촉으로 온도를 측정할 수 있는 시스템을 아두이노로 구현하려고 한다. |

|  |
| --- |
| **3. H/W 설계** |

*H/W를 어떻게 구현할 지를 설계, 사용 센서, 부품 들을 나열.*

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 아두이노 우노 보드, 브레드보드, 초음파센서 모듈(HC-SR04), 적외선 온도센서, I2C LCD 디스플레이, LED(노랑, 초록, 빨강), 수동부저, 1kΩ저항, 케이블, USB포트  다음과 같이 센서를 아두이노에 연결한다.     |  |  | | --- | --- | | **HC-SR04** | **아두이노** | | **VCC** | 5V | | **Trig** | D13 | | **Echo** | D12 | | **GND** | GND |  |  |  | | --- | --- | | **MLX90614E** | **아두이노** | | **SCL** | SCL | | **SDA** | SDA | | **VCC**  **GND** | 5V  GND |      |  |  | | --- | --- | | **I2C LCD** | **아두이노** | | **SCL** | SCL | | **SDA** | SDA | | **VCC** | 5V | | **GND** | GND |      |  |  | | --- | --- | | **LED** | **아두이노** | | **노랑** | D3 | | **녹색** | D4 | | **빨강** | D5 |      |  |  | | --- | --- | | **Buzzer** | **아두이노** | | **+** | D8 | | **-** | GND |       [완성된 H/W]    (1) 정상체온 일 때    (2) 저체온 일 때    (3) 고체온 일 때 |
|  |

|  |
| --- |
| **4. S/W 설계** |

*S/W를 어떻게 구현할 지를 설계*

|  |
| --- |
| 초음파센서 모듈(HC-SR04)을 벽에서 일정 거리 떨어진 곳에 고정하면 초음파센서는 거의 일정한 거릿값을 출력하게 된다. 만약 센서와 벽간의 통로를 사람이 지나가면 거리 값이 작게 측정되는데 이는 사람이 지나갔다는 것을 의미한다. 그리고 지나간 횟수만큼 count를 증가 시켜 출입 인원수를 시리얼포트와 LCD 화면에 출력한다.  문제는 사람이 지나가는 속도와 패턴이 일정하지 않을 수 있기에 한 번에 한 사람씩 지나가고, 사람과 사람이 지나갈 때의 시차는 1초가 넘고, 센서 앞에 사람이 1초 이상 머무르지 않는다고 가정하였다.  센서와 벽 사이의 거리인 CM 값을 100이라고 가정하고, 사람이 지나갈 때의 값(센서와 사람 사이의 거리)을 50으로 잡았다. 오차범위를 고려해 값이 60보다 작거나 같으면 사람이 지나간 것으로 간주했다.  카운트한 후에 중복해서 카운트되는 것을 막기 위해 delay(1000) 함수를 사용하여 1초간 시스템을 정지시킨다.  적외선 온도센서를 이용해 체온을 측정하여 발열을 검사한다.  스케치 코드에 비접촉식 온도측정센서 라이브러리인 Adafruit\_MLX90614.h를 포함하여 온도 값 strTemp를 측정한다. 오참범위를 고려해 strTemp 값이 34.9~37.6도이면 정상체온, 34.9도보다 낮으면 저체온, 37.6도보다 높으면 고체온으로 정의한다.  측정 온도에 따라 저체온이면 D3에 HIGH 신호를 주어 노랑 LED에 불이 들어오고, 정상체온이면 D4에 HIGH 신호를 주어 녹색 LED에 불이 들어오고, 고체온이면 D5에 HIGH 신호를 주어 빨강 LED에 불이 들어온다.  만약 측정 온도가 고체온이면 수동부저로 삐 소리를 출력하고 발열 출입자로 인식해 출입을 거부한다.  그리고 측정온도와 정상체온 여부를 시리얼포트와 LCD 화면에 출력한다.  [스케치코드]  // IOT시스템 미니프로젝트 스케치 컴퓨터소프트웨어공학과 2-YA 20202296 전채린  #include <Wire.h> // i2c 통신을 위한 라이브러리  #include <Adafruit\_MLX90614.h> // 비접촉식 온도측정센서 라이브러리  #include <LiquidCrystal\_I2C.h> // lcd 1602 i2c 용 라이브러리  #define trigPin 13  #define echoPin 12  #define maximumRange 200 // 최대 측정 거리  #define minimumRange 0 // 최소 측정 거리  int ledy = 3; // 노랑 LED  int ledg = 4; // 녹색 LED  int ledr = 5; // 빨강 LED  int buzzer = 8; // 부저  int count = 0; // 출입 인원수  LiquidCrystal\_I2C lcd(0x27, 16, 2);  Adafruit\_MLX90614 mlx = Adafruit\_MLX90614();  void setup()  {  pinMode(trigPin, OUTPUT);  pinMode(echoPin, INPUT);  pinMode(ledy, OUTPUT);  pinMode(ledg, OUTPUT);  pinMode(ledr, OUTPUT);  pinMode(buzzer, OUTPUT);  Serial.begin(9600);  lcd.begin(); // lcd 시작  lcd.backlight(); // 백라이트 켜기  lcd.noCursor(); // 커서를 없앰  lcd.noBlink(); // 커서를 깜박이지 않음  mlx.begin(); // mlx 모듈을 읽기 시작  }  long microsecondsToCentimeters(long microseconds)  {  // The speed of sound is 340 m/s or 29 microseconds per centimeter.  // The ping travels out and back, so to find the distance of the  // object we take half of the distance travelled.    return microseconds / 29 / 2;  }  void loop()  {  long duration, cm;  digitalWrite(trigPin, LOW);  delayMicroseconds(2);  digitalWrite(trigPin, HIGH);  delayMicroseconds(10);  digitalWrite(trigPin, LOW);  // echoPin이 HIGH를 유지한 시간을 저장  duration = pulseIn(echoPin, HIGH);  // HIGH였을 때 시간(초음파가 보냈다가 다시 들어온 시간)을 가지고 거리를 계산  cm = microsecondsToCentimeters(duration);  // 측정 범위를 벗어났을 때  if (cm >= maximumRange || cm <= minimumRange)  {  Serial.println("out of range");  lcd.clear();  lcd.print("out of range");  }    // 측정 범위일 때  else  {  Serial.print(cm, DEC);  Serial.println("cm");    // 사람이 지나갈 때  if(cm <= 60)  {  count = count + 1;  Serial.print("count : ");  Serial.println(count);  lcd.clear();  lcd.print("count : ");  lcd.print(count);    double strTemp; // 온도 측정  strTemp = mlx.readObjectTempC();    // 정상체온일 때  if(strTemp >= 34.9 && strTemp <= 37.6) {  Serial.print("Temperature : ");  Serial.println(strTemp);  Serial.println("normal temperature");  Serial.println("초록 LED ON");  lcd.clear();  lcd.print("Temperature");  lcd.setCursor(4,1);  lcd.print(strTemp);  delay(500);  lcd.clear();  lcd.print("normal temperature");  digitalWrite(ledg, HIGH); // 초록 LED ON  delay(500);  digitalWrite(ledg, LOW); // 초록 LED OFF  }  // 저체온일 때  else if(strTemp < 34.9) {  Serial.print("Temperature : ");  Serial.println(strTemp);  Serial.println("low temperature");  Serial.println("노랑 LED ON");  lcd.clear();  lcd.print("Temperature");  lcd.setCursor(4,1);  lcd.print(strTemp);  delay(500);  lcd.clear();  lcd.print("low temperature");  digitalWrite(ledy, HIGH); // 노랑 LED ON  delay(500);  digitalWrite(ledy, LOW); // 노랑 LED OFF  }  // 고체온일 때  else if(strTemp > 37.6) {  Serial.print("Temperature : ");  Serial.println(strTemp);  Serial.println("high temperature");  Serial.println("빨강 LED ON");  Serial.println("buzzer ON");  lcd.clear();  lcd.print("Temperature");  lcd.setCursor(4,1);  lcd.print(strTemp);  delay(500);  lcd.clear();  lcd.print("high temperature");  digitalWrite(ledr, HIGH); // 빨강 LED ON  digitalWrite(buzzer, HIGH); // Buzzer ON  delay(500);  digitalWrite(ledr, LOW); // 빨강 LED OFF  digitalWrite(buzzer, LOW); // Buzzer OFF  }  }  delay(1000); // 1초 간격으로 측정  }  } |
|  |

|  |
| --- |
| **5. 구현 소감** |

*개발중 있었던 일과 느낌*

|  |
| --- |
| 스케치 코드로 구현한 내용을 직접 아두이노에 연결해 실행할 때 오류는 없는데 실행이 잘 안 돼서 어려움을 겪었다. 또한, 온습도센서는 공기 중의 온도와 습도를 측정하는 온도로 체온 측정에 적합하지 않고 빠르게 측정할 수 없어 적외선 온도센서를 구매해 프로그램을 구현했다. 적외선 온도센서는 -70~380도까지 측정할 수 있고 오차범위가 +-0.02인데 구매한 적외선 온도센서 부품이 불량이어서 온도가 1037로 측정되어 당황스러웠다. 그래서 구매한 곳에 문의드렸더니 새로운 제품을 보내주신다고 하셨는데 발송이 늦어져서 초조하고 불안했었다. 그래도 완성해 실행되는 것을 보니 뿌듯했고 아두이노로 다양한 경험을 해볼 수 있어서 의미 있었다. |

|  |
| --- |
| **5. 향후 보완 내용** |

|  |
| --- |
| 비접촉 생체인식 및 자동 발열감지 시스템을 학교에서 학생을 인식하고 정보를 저장할 수 있도록 확대해보고 싶다. 휴대전화/태블릿을 통해 학생 사진을 등록하고, 출입구에 설치된 페이스 출력 체크로 안면 인식하여 출입시간, 발열 체크 이상 유무 등 데이터를 전송하고 이를 데이터베이스에 저장한다. 발열자 발생 및 감지 시 관리자에게 문자를 전송하고, 데이터베이스에 저장된 출입기록 및 발열 체크 기록을 조회할 수 있도록 보완해보고 싶다. |

|  |
| --- |
| **5. 참고 자료** |

*계획 시 참고한 자료, 사이트 등*

|  |
| --- |
| 참고 자료 : 아두이노 완전정복, IOT시스템 강의 자료 |