**임베디드 기초실습**

**텀프로젝트 결과보고서**

* **Micromouse -**

학 번 : 14051012

이 름 : 도진우

제출날짜 : 2020-07-07

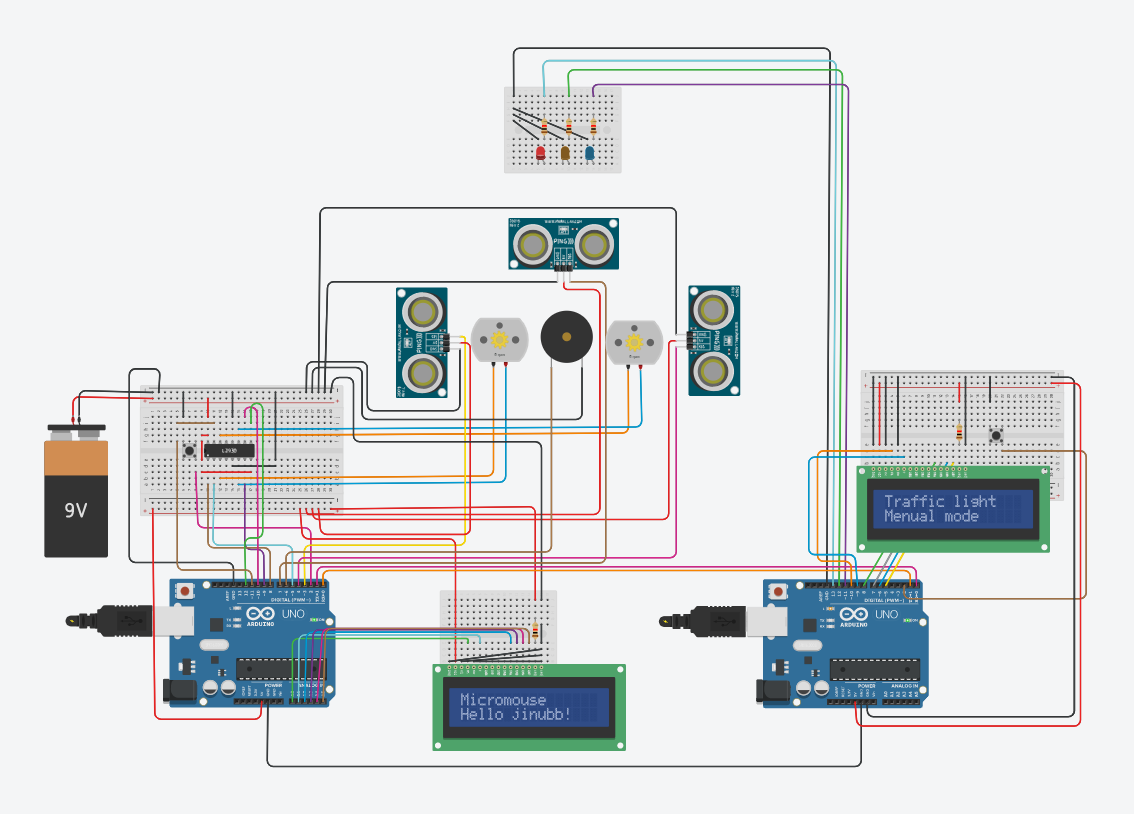


**목차**

1. *개요*
2. *Micromouse 소개*
3. *기능 설명*
4. *느낀점*
5. **개요**
   1. 회로 URL

<https://www.tinkercad.com/things/flvzzj8evwN-term-project-micromouse/editel>

* 1. 회로 사진



* 1. 사용부품 목록 및 핀 넘버링
     1. Uno 1 (Micromouse)

|  |  |  |  |
| --- | --- | --- | --- |
| 번호 | 사용부품 | 핀넘버링 | 용도 |
| 1 | 버튼스위치 | 2 | 마이크로마우스 정지/직진 모드 변경 |
| 2 | 초음파센서(좌측) | 3 | 좌측 벽 감지 |
| 3 | 초음파센서(우측) | 4 | 우측 벽 감지 |
| 4 | 초음파센서(전방) | 7 | 전방 벽 감지 |
| 5 | 피에조 부저 | 6 | 경고음 출력 |
| 6 | L293D 드라이버 | 8 | 좌측모터 Enable |
| 5 | 좌측모터 입력1 |
| 9 | 좌측모터 입력2 |
| 12 | 우측모터 Enable |
| 10 | 우측모터 입력1 |
| 11 | 우측모터 입력2 |
| 7 | DC 모터 | x | 좌측 모터 |
| x | 우측 모터 |
| 8 | LCD | A0~A5 | LCD 스크린 출력 |

* + 1. Uno 2 (Traffic light)

|  |  |  |  |
| --- | --- | --- | --- |
| 번호 | 사용부품 | 핀넘버링 | 용도 |
| 1 | 버튼스위치 | 2 | 신호등 수동/자동 모드 변경 |
| 2 | LCD | 5~10 | LCD 스크린 출력 |
| 3 | LED 다이오드(파란색) | 11 | 파란불(직진신호) |
| 4 | LED 다이오드(주황색) | 12 | 주황불 |
| 5 | LED 다이오드(빨간색) | 13 | 빨간불(정지신호) |

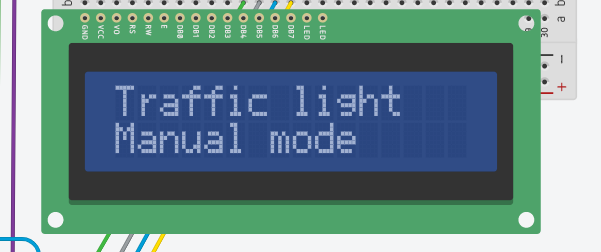
1. **Micromouse 소개**
   1. 자율주행 자동차 구현

* 전방과 좌/우측에 센서를 달아 장애물을 감지하고 회피하는 동작
* 신호등의 신호를 수신하여 자동차 출발/정지 제어
* 사용자에게 자동차의 상태를 LCD로 출력

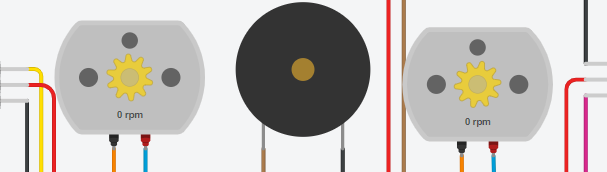
1. **기능 설명**
   1. LCD
      1. Micromouse LCD : 자동차의 상태를 LCD로 출력



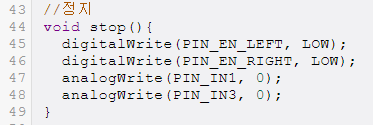
* 시작 메시지(Default) : jinubb
* 직진 : Go straight!
* 좌/우측 벽 근접 : Left/Right speedup
* 좌/우측 회전 : Turn Left/Right
* 행동불능 : Can’t go
* 정지 : Stop!
  + 1. Traffic light LCD : 신호등의 상태를 LCD로 출력



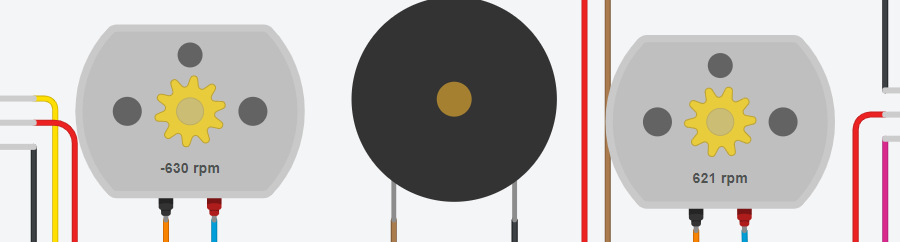
* 수동 모드(Default) : Manual mode // 버튼으로 자동차 제어
* 자동 모드 : Auto mode // 신호등에 따라 자동차 제어
  1. 정지(Default)



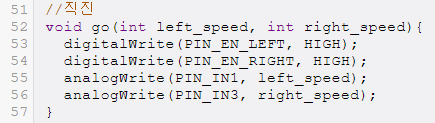
* 정지시 모터 사진



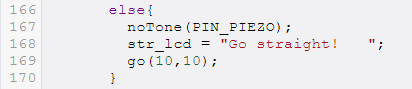
* + 1. 좌/우측 모터 드라이버 Enable핀에 LOW출력
    2. 좌/우측 모터 드라이버 입력1,3 핀에 pwm 0 출력
    3. 입력 2,4핀은 LOW로 setup되어있음.
    4. Enable에 LOW출력과 H브릿지에서 LOW-LOW 출력으로 모터 정지
  1. 직진



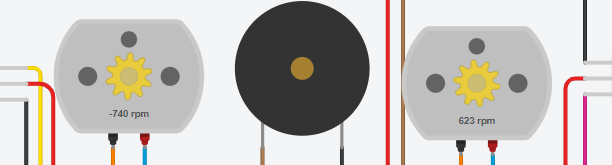
* 직진시 모터 사진



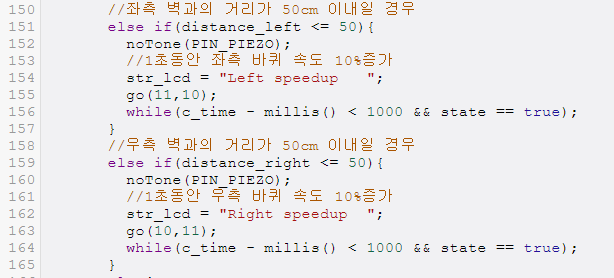
* 좌/우측 모터의 스피드를 pwm값으로 출력(0~255)



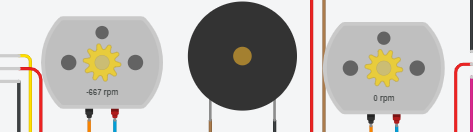
* 전방, 좌측, 우측
* 기본 직진 스피드값은 10으로 지정(600대rpm)
  1. 좌/우측 벽 근접(좌/우측 벽에 근접할 경우)



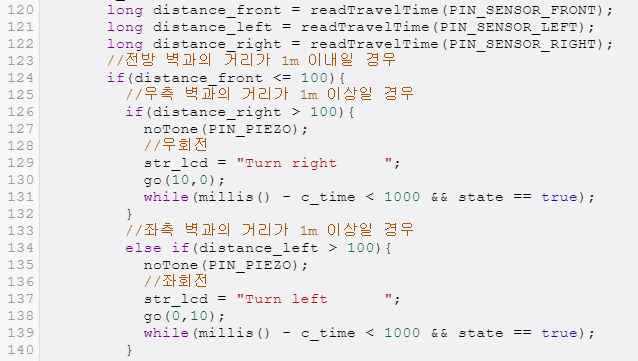
* 좌측 벽 근접시 모터 사진



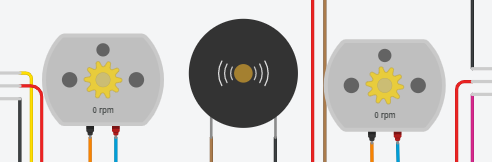
* 직진 모드 중 좌/우측 벽과의 거리가 50cm 이내일 경우 작동
* 1초동안 좌/우측 모터 스피드 10%증가(go함수의 pwm값 10->11 로 증가)
* millis()를 이용하여 1초를 기다리는동안 버튼입력(수동모드) 혹은 신호등 빨간불(자동모드) 시 빠른 반응속도로 제어
  1. 좌/우 회전



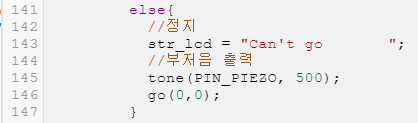
* 우회전시 모터 사진



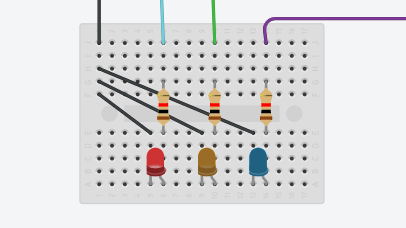
* 전방 벽과의 거리가 1m 이내이면서
* 좌/우측 벽과의 거리가 1m 이상일 경우에 좌/우 회전 동작
* 좌/우회전시 1초동안 좌/우측 모터 스피드 0으로 출력
* millis()를 이용하여 1초를 기다리는동안 버튼입력(수동모드) 혹은 신호등 빨간불(자동모드) 시 빠른 반응속도로 제어
  1. 행동불능

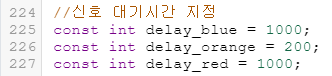


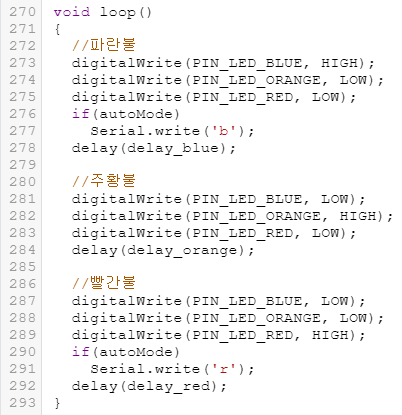
* 행동불능시 모터 및 피에조 사진



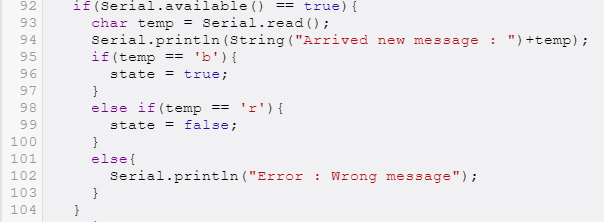
* 전방, 좌측, 우측 벽과의 거리가 모두 1m이내인 경우 행동불능
* 행동 불능 시 모터가 정지하고 피에조에서 부저음 출력
  1. 신호등





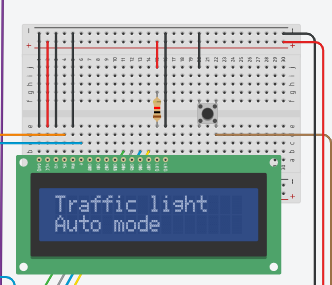


* 파란불->주황불->빨간불 순서로 지정된 시간만큼 해당 LED출력
* 모드 선택 변수(autoMode)가 false일 경우 수동모드
* 수동모드는 Micromouse로 신호등 정보를 보내지 않음
* 모드 선택 변수(autoMode)가 true일 경우 자동모드
* 자동모드는 Micromouse로 신호등 정보를 보내서 제어
* 하드웨어 시리얼 이용(0,1번 핀, 9600 Baud rate)



* Micromouse(Uno 1)의 신호등 정보 수신 코드
* 하드웨어 시리얼 이용(0,1번 핀, 9600 Baud rate)
* 빨간불일경우 정지, 파란불일경우 출발

* + 1. 자동/수동 모드 변경 스위치

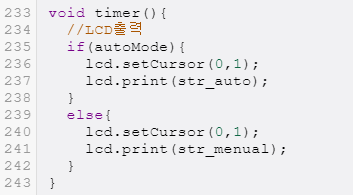






* 인터럽트 방식을 이용하여 변수값 변경





* Timer1 라이브러리를 이용하여 0.3초마다 변수값이 변동되었는지 확인하여 LCD에 출력

1. **느낀점**
   1. 아이디어 구상
      1. 초기 아이디어는 기본 기능에 추가적으로 적외선 리모컨으로 동작 제어, LCD로 상태 출력을 추가 기능으로 구상했다.
      2. 회로 설계 단계에서 신호등에서 자동차로 신호를 보내서 자율 주행 차량에 더 가깝게 해보면 어떨까 생각이 들어서 신호등을 추가했다.

* 1. 문제점
     1. Uno의 적은 pin 개수
* Uno는 다른 보드에 비해서 적은 pin개수를 갖는다. Uno1(Micromouse)의 digital pin이 부족해서 LCD를 연결하지 못하다가 analog pin을 이용해서 HIGH를 출력하는 방법으로 사용했다.
  + 1. 적외선 리모컨 오류
* 메인 코드에 적외선 리모컨 코드를 추가해서 돌렸는데 컴파일 에러가 나왔다. 라이브러리쪽 문제라서 확인은 불가능했고 같은 코드를 새로운 회로를 만들어서 돌려보니 잘 돌아갔다. 틴커캐드에서 다른 센서와 충돌하는 것 같아서 적외선 리모컨을 제거했다.
  1. 개선할 점
     1. 프로그램 성능
* 여러 가지 센서를 잡다하게 넣다 보니 프로그램이 너무 느리게 돌아간다. 데스크탑에서는 테스트 할 정도로 돌아가지만 성능이 떨어지는 노트북에서는 더욱 느리게 돌아갔다. 틴커캐드의 문제인지 아두이노의 한계인지 구별할 순 없었으나 millis()함수를 너무 남용한 문제점으로 예상된다. Uno2(Traffic light)에서는 성능 향상을 위해 버튼에 반응하여 LCD에 메시지를 띄울 때 millis()를 이용하지 않고 0.3초마다 동작하는 Timer1을 이용했지만 큰 차이는 없었다. 여러가지 동작을 하기 위해 delay()함수를 이용하지 않고 millis()함수를 이용했지만 다른 방법으로 문제점을 해결해야 할 것으로 생각된다.
  1. 프로젝트를 마치며
* 수업을 듣기 전에는 프로그램 코딩에는 자신이 있었지만 보드와 센서를 연결해서 회로를 만드는 부분에는 자신이 없었다. 이번 학기에 온라인으로 수업을 진행하면서 틴커캐드를 이용하여 편하게 회로를 구성하다 보니 회로에 대한 두려움이 사라졌다.
* 프로젝트 진행 순서를 아이디어 구상 -> 회로 설계 -> 코딩 순서로대로 진행하였는데 회로 설계 단계에서 여러 센서를 미리 설계하고 코딩에 들어갔다. 다른 프로젝트에서도 회로 설계와 코딩을 나누는 방법으로 협업을 하면 더 효과적으로 프로젝트를 진행할 수 있을 것 같다.