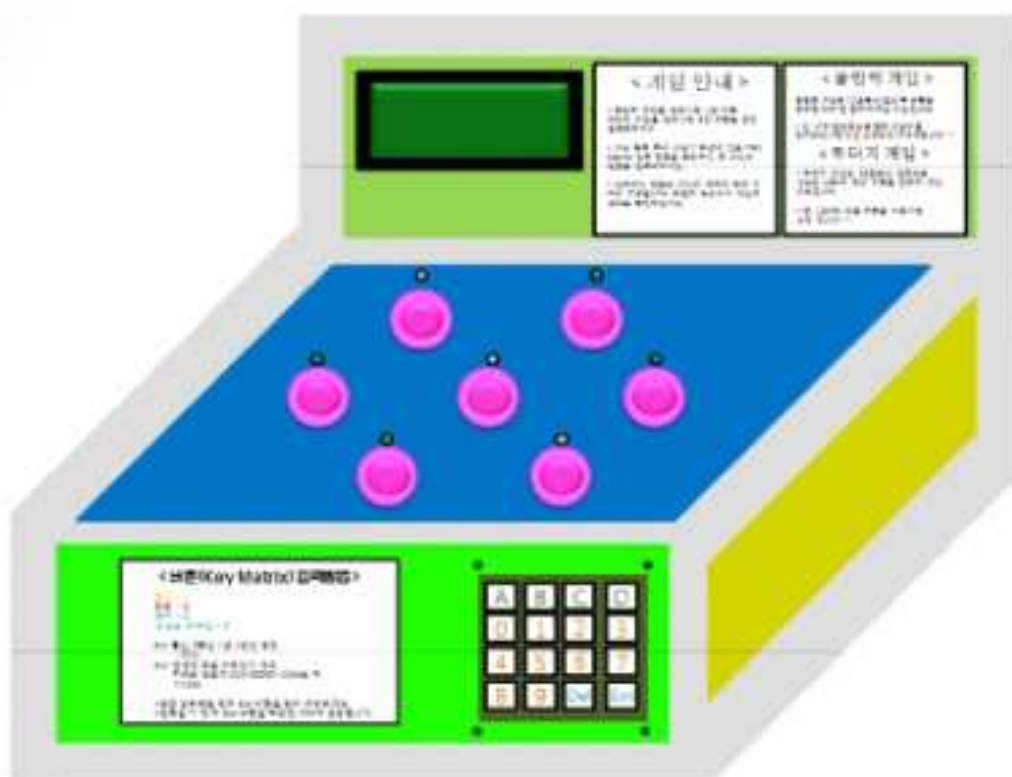


O.M.G(게임)



MDP명	정보통신기기설계
학과 학년	정보통신기기과 2학년
학생성명	선예은 박예빈 강수빈 황민혁 김경빈 김세인
지도교사	조아형 선생님, 최보람 선생님



인천전자마이스터고등학교

1. 개요

많은 학교생활 중 모두가 함께 즐길 수 있는 과제가 뭐가 있을까. 한 기계에서 여러 게임을 한꺼번에 모두 할 수 있을까. 개인이 게임기를 만들어 사용하는 사례를 많이 보아왔는데 우리가 그 게임기를 만들 수 있을까하여 종합게임기를 만들게 되었습니다. 그래서 보통 단일기계로만 있는 기(G)역력 게임, 올(O)림픽 게임, 두(D)더지잡기 게임을 합쳐보기로 하였습니다. 조작법을 간단하게 하여 누구나 쉽게 즐길 수 있도록 하였습니다. 기역력게임은 스위처에 차례로 불이 들어오게 하여 후에 사용자가 그 순서대로 다시 누르는 게임입니다. 올림픽 게임은 제한시간 안에 버튼을 얼마나 많이 눌렀느냐로 기록을 세우는 게임이며, 두더지잡기게임은 흔히 아는 게임의 방식에 두더지를 LED로 대신하여, 켜진 LED에 알맞은 스위치를 누르면 득점하는 게임입니다.

2. 조원 소개 및 역할 분담

이름	성별	특징	역할	비고
선예은	여	모든 면에 능동함	게임 소스 작성, 서버 제작, 하드웨어 총괄, 보고서 작성	조장
박예빈	여	회로 해석 및 구상에 능동함	회로 설계 및 제작	
강수빈	여	소프트웨어 설계를 잘함	게임 소스 작성	
황민혁	남	회로 해석과, 캐드활용을 잘함	하드웨어 설계 및 외형물 제작	
김경빈	남	구조물 설계 및 제작을 잘함	외형물 제작	
김세원	남	구조물 제작을 잘함	외형물 제작	

3. 제작 기간: 2018.3.~2018.11.

기간 내용	3월	4월	5월	6월	7월	8월	9월	10월	11월
논론, 아이디어, 구상									
아이디어 구상									
소프트웨어 개발									
소프트웨어 개발									
물류, 사출, 조립									
시험 개발									

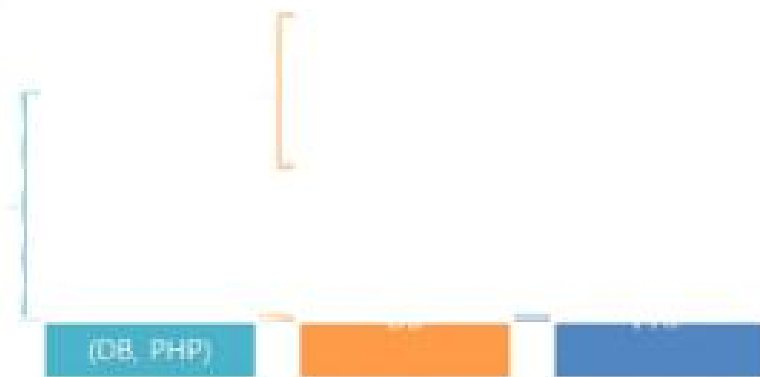
4. 작품 제작 계획하기

가. 아이디어 구상



나. 블록별 모형

1) 소프트웨어블록도



2) 하드웨어블록도



다. 동작 설명

1) 전원 스위치를 켜다.

2) 1번, 2번 스위치를 통해 게임을 선택한다.

올림픽게임) 1번 스위치를 눌러 올림픽 게임을 선택한 경우,

10초 동안 스위치를 최대한 많이 눌러 누른 개수만큼 득점하게 된다.

두더지게임) 2번 스위치를 눌러 두더지 잡기 게임을 선택한 경우,

15초 동안 7개의 LED가 랜덤적으로 1개씩 점등할 것이다.

점등 된 LED 바로 밑에 있는 스위치를 타이밍에 맞춰 누르면 1점씩 득점한다.

10초가 지나면 LED의 점등 속도는 더 빨라진다.

단, 잘못 누를 시 설점하게 된다.

3) 게임이 끝나면 LCD에 자신의 신상정보를 입력하도록 표시된다.

4) KeyMatrix를 이용하여 학과, 학번(전화번호 뒷자리) 순으로 입력한 후 Enter를 누른다.

5) 입력한 정보가 점수와 함께 서버에 전송된다.

5. 사용 부품 목록

부품명	모델명 및 사양	구매 사이트
ATmega128	ATmega128	
Raspberry Pi	Version 2	
LCD	Character LCD 16x2	
LED	2Ω	
switch	DC, 원형	http://elecloud.co.kr

6. 주요 부품의 특징

가. ATmega128

1) 특징

- 가) 64핀으로 구성
- 나) TQFP형 패키지의 저 전력 8비트 CMOS 마이크로컨트롤러
- 다) ISP 다운로드에서 PDI, PDO 사용
- 라) 데이터 메모리란 외부에 최대 64kByte 까지 확장 가능
- 마) mega103 호환모드를 가짐
- 바) 2개의 UART 회로
- 사) RTC 기능을 가짐
- 아) JTAG을 지원
- 자) 8비트 타이머/카운터 2개
- 차) 16비트 타이머/카운터 2개
- 카) 실시간 타이머 카운터
- 타) 아날로그 비교기
- 과) 10비트 A/D 변환기
- 하) 2개의 USART, 2개의 TWI 저력 인터페이스



나. 라즈베리파이

영국의 라즈베리파이 재단이 학교에서의 기초 컴퓨터 과학 교육을 프로젝트의 목적으로 개발한 신용카드 크기의 초소형/초저가 PC이다. 2006년에 개념이 형성되고 재단이 만들어져 2012년 처음 제품이 나온 이후 2013년 1월 초에 백만 대가 판매되었다. 라즈베리파이의 특징은 아두이노와 달리 키보드, 마우스, 모니터만 연결하면 PC가 될 수 있다. 즉 일반 데스크톱과 유사하다는 것이 강조되는 제품이다. Linux OS를 기반으로 하고 세부적인 설정을 제공함으로써 초보 프로그래머에 맞춤형 환경을 제공한다. 결국, 많은 사람들을 열정과 취미를 통해 입문하게 한 다음 전문 프로그래머로 불리는 계층을 제공함으로써 임베디드 OS기반의 S/W의 인력 풀과 현장을 줄이는데 이바지하는 또 다른 OSHW 프로젝트 결과물이다. 초기 제품은 브로드컴의 BCM2835 SoC를 기반으로 하는데, 이 칩은 ARM Core 11 700MHz 프로세서, VideoCore IV GPU, 512MB(모델B/B+용 A/A+는 256) SDRAM을 포함하여 아두이노에 비해 그래픽 기능이 포함된 것이 차이점이다. HDMI를 통해 모니터 연결이 가능하다. SD 메모리를 보조 메모리로 사용하며 통신 기능은 모델B/B+만 Ethernet을 가지고 있다. 무선 통신을 하려면 USB를 통한 확장 WiFi 모듈 등을 이용해야 한다. 아두이노와 같이 센서, 액추에이터 등을 연결해 다양한 기능을 구현할 수 있으나 아두이노와 달리 동영상 카메라를 어려움 없이 적용할 수 있어 2013년 5월에는 전용 카메라 모듈도 발매했다. 라즈베리 재단에서 공식적으로 지원하는 OS는 리눅스 데비안을 기반으로 하는 라즈비안(Raspbian)이다.



다. LCD



1) 소개

인가전압에 따른 액정 두께도의 변화를 이용하여 각종 장치에서 발생하는 여러 가지 전기적인 정보를 시각정보로 변화시켜 전달하는 전기소자이다. 자기발광성이 없어 후광이 필요하지만 소비전력이 적고 휴대용으로 편리해 널리 사용하는 평판 디스플레이이다. LCD의 핵심은 화면을 표현하는 소자인 액정(Liquid crystal: 液晶)이다. 수많은 액정을 규칙적으로 배열한 재질을 전면에 배치한 뒤, 그 뒤쪽에 위치한 백라이트(back light: 후광 조명)가 빛을 가하도록 한다. 각 액정 소자는 외부에서 가해진 전기 신호에 따라 내부적인 분자의 배열이 변화하며 각각 일정한 패턴의 방출성을 띠게 된다. 이에 따라 백라이트에서 전해진 빛은 각각의 액정을 통과하면서 각기 다른 패턴으로 굴절하며, 이 빛이 액정 패널 앞에 있는 컬러 필터의 편광 필터를 통과하면 굴절 패턴에 따라 각기 다른 색상과 밝기를 띤 하나의 화소(pixel: 화면을 구성하는 하나의 점)가 되므로 이들이 모여 전체 화면을 구성하게 된다. 물론, 위와 달리 백라이트 없이 외부의 빛에 의존하는 경우도 있으며, 후면 화면만 표시하는 경우도 있는 등, 제조사나 제품에 따라 세부 구조에 차이가 나기도 한다.

2) 특징

- 가) Character LCD 16x2
- 나) 5x8 dots includes cursor
- 다) Built-in controller (ST7066 or Equivalent)
- 라) +5V power supply (Also available for +3V)
- 마) Negative voltage optional for +3V power supply
- 바) 1/16 duty cycle
- 사) LED can be driven by PIN1, PIN2, PIN15, PIN16 or A and K
- 아) WH1602B2: total thickness 10.0mm

라. Switch



전로(電路)를 끊거나 잇는 기구. 보통 30A 이하의 개폐기를 말한다. 누름 버튼 스위치, 회전 스위치, 나이프 스위치(knife switch), 변환 스위치 등 종류가 매우 많다. 개폐기라고도 한다. 구조는 접점(接點)과 그것을 동작시키는 기구로 되어 있다. 손으로 동작시키는 간단한 것에서 전자기력(電磁力)에 의하여 작동되는 대형까지 매우 많은 방식과 구조가 있다. 전력용에서는 개폐에 수반되는 불꽃이나 열에 의한 접점의 소손(燒損)을 피하기 위하여 멀리 끊는 기구를 설치한 것이 있으며, 대형에는 소호장치(消弧裝置)를 붙이기도 한다. 신호회로에는 다접점(多接點)의 스위치가 사용되며 또 전기신호로 동작하는 계전기(릴레이)·진공관·반도체 등을 이용한 전자스위치는 논리회로로서 이용된다. 이와 같이 전압·전류·사용목적 등에 따라 많은 종류가 있다. 게임기에 사용된 스위치는 TACT 스위치라고 볼 수 있다.

바. LED



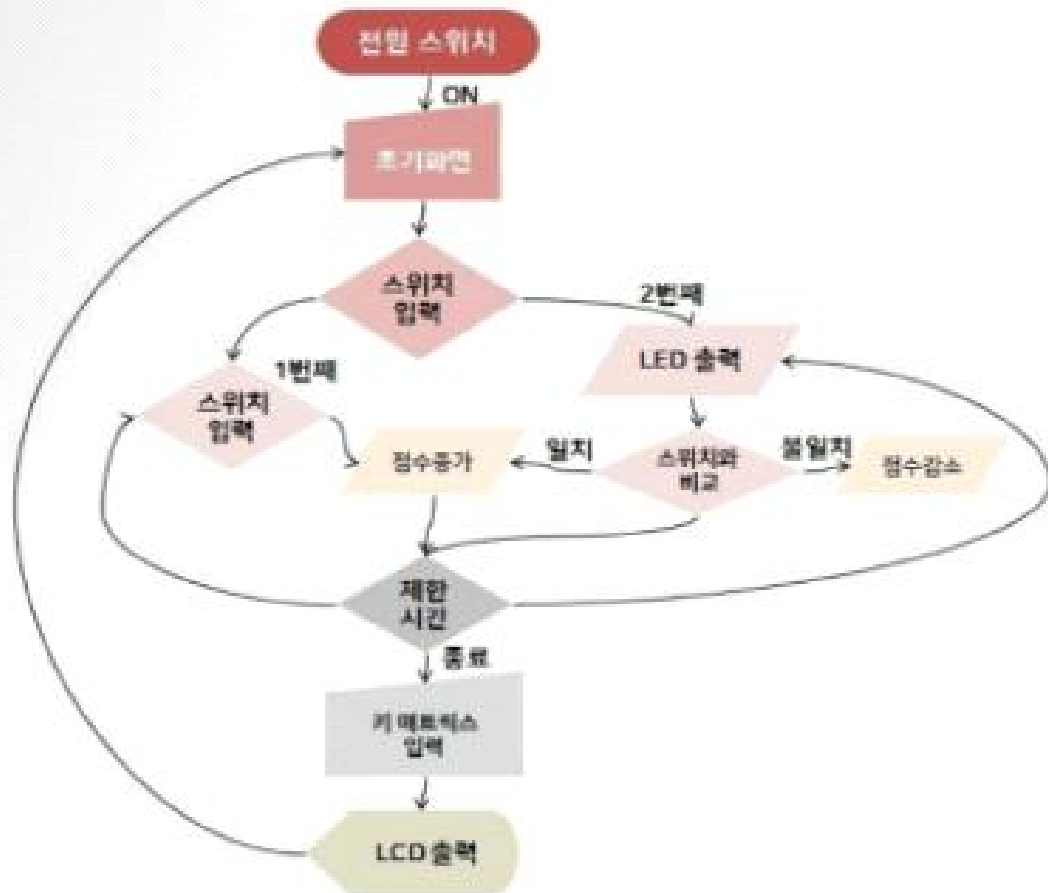
발광 다이오드, Ga(갈륨), P(인), As(비소)를 재료로 하여 만들어진 반도체, 다이오드의 특성을 가지고 있다. LED는 전자를 발생시키기 쉬운 전자공여물질(전자물질)과 구멍을 형성하는 전자수용물질(구멍물질)을 결합시켜 만든 소자이다. 여기에 잘 조절된 낮은 전류를 흘리면 전자와 구멍이 접합면에서 만나 합선해서 엑사이톤을 만든다. 이 엑사이톤은 곧 빛으로 변해서 밖으로 발하게 되어 붉은색, 녹색, 노란색으로 빛을 발한다. 전구에 비해 수명이 길고 응답 속도(전류가 흘러서 빛을 발하기까지의 시간)가 빠르고 다양한 모양으로 만들 수 있다는 데 이점이 있어, 피크 인디케이터 또는 숫자 표시에 가장 적합하다.

7. 회로 설계하기

가. 전체 회로도 및 회로설명

<p>[그림 1] Switch, LED 회로도</p>	<p>[그림 2] KeyMatrix 회로도</p>
<p>[그림 1]은 LCD모듈을 사용하기 위한 회로도. [그림 2]은 개인 정보 입력을 위한 KeyMatrix 회로도다.</p>	
<p>[그림 3] Switch, LED 회로도</p>	
<p>[그림 3]는 기존에 설계된 Active_low 회로에 우리가 필요한 스위치와 LED개수만큼 재구성한 회로도.</p>	

8. 소프트웨어 작성하기 가. 프로그램 흐름도



나. 서버 소스

Index 소스)

```
<!DOCTYPE HTML>
```

```
<html>
```

```
<head>
```

```
<title>O.M.G</title>
```

```
<meta charset="utf-8" />
```

```
<meta name="viewport" content="width=device-width, initial-scale=1" />
```

```
<!--[if lte IE 8]><script src="assets/js/ie/html5shiv.js"></script><![endif]-->
```

```
<link rel="stylesheet" href="assets/css/main.css" />
```

```
<!--[if lte IE 9]><link rel="stylesheet" href="assets/css/ie9.css" /><![endif]-->
```

```
<!--[if lte IE 8]><link rel="stylesheet" href="assets/css/ie8.css" /><![endif]-->
```

```
</head>
```

```
<body>
```

```
<!-- Sidebar -->
```

```
<section id="sidebar">
```

```
<div class="inner">
```

```
<nav>
```

```
<ul>
```

```
<li><a href="#intro">Welcome</a></li>
```

```
<li><a href="#one">Intro.</a></li>
```

```

        <li><a href="#two">Go Score</a></li>
    </ul>
</nav>
</div>
</section>

```

```

<!-- Wrapper -->
<div id="wrapper">

```

```

    <!-- Intro -->
    <section id="intro" class="wrapper style1 fullscreen fade-up">
        <div class="inner">
            <h1>O.M.G</h1>
            <p>정보통신기기 설계반<br/>
                MDP프로젝트 작품 - 게임기</p>
            <ul class="actions">
                <li><i><a href="#one" class="button scrolly">Next</a></i></li>
            </ul>
        </div>
    </section>

```

```

    <!-- One -->
    <section id="one" class="wrapper style1 fullscreen fade-up">
        <div class="inner">
            <h1>O.M.G</h1>
            <p>힘든 학교생활 중 모두가 원하고 즐길 수 있는 과제가 뭐가 있을까,<br />
                한 기계에서 여러 게임을 한꺼번에 모두 할 순 없을까,<br/>
                개인이 게임기를 만들어 사용하는 사례를 많이 보아왔는데 <br/>
                우리가 그 게임기를 만들 순 없을까 하여 종합게임기를 만들게 되었습니다.<br/>
                그래서 단일기계로만 있는 올림픽게임과 두더지게임을 골라 게임기를 만들었습니다.<br/></p>
            <ul class="actions">
                <li><i><a href="#two" class="button scrolly">Next</a></i></li>
            </ul>
        </div>
    </section>

```

```

    <!-- two -->
    <section id="two" class="wrapper style3 fade-up">
        <div class="inner">
            <h2>Go_Score</h2>
            <hr>
            <ul class="actions">
                <li><a href="olympic.php" class="button">OLYMPIC.</a></li>
            </ul>

```



```

        <ul class="actions">
            <li><a href="mole.php" class="button">MOLE.</a></li>
        </ul>
    </div>
</section>
</div>

<!-- Footer -->
<footer id="footer" class="wrapper style1-alt">
    <div class="inner">
        <ul class="menu">
            <li>Made by, 전예은 박예민 강수빈 김경빈 황민혁 김세현</li>
        </ul>
    </div>
</footer>

<!-- Scripts -->
<script src="assets/js/jquery.min.js"></script>
<script src="assets/js/jquery.scrollex.min.js"></script>
<script src="assets/js/jquery.scrolly.min.js"></script>
<script src="assets/js/skel.min.js"></script>
<script src="assets/js/util.js"></script>
<!--[if lte IE 8]><script src="assets/js/ie/respond.min.js"></script><![endif]-->
<script src="assets/js/main.js"></script>

</body>
</html>

```

올림픽게임 점수창 소스)

```

<?php
$link=mysqli_connect("localhost","jbts","jbts","score");
if (!$link)
{ echo "MySQL error : "; echo mysqli_connect_error(); exit(); }
mysqli_set_charset($link,"utf8");
$sql = "SELECT * FROM `mole` ";
$result = mysqli_query($link, $sql);
?>

<!DOCTYPE HTML>
<html>
<head>
    <title>두더지 게임 점수 확인</title>
    <meta charset="utf-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1, user-scalable=no" />
    <link rel="stylesheet" href="assets/css/main.css" />
    <noscript><link rel="stylesheet" href="assets/css/noscript.css" /></noscript>

```

```

</head>
<body class="is-preload">

<!-- Header -->
<header id="header">
  <a href="index.php" class="title">O.M.G</a>
  <nav>
    <ul>
      <li><a href="olympic.php">Olympic</a></li>
      <li><a href="mole.php">Mole</a></li>
    </ul>
  </nav>
</header>

<!-- Wrapper -->
<div id="wrapper">

<!-- Main -->
<section id="main" class="wrapper">
  <div class="inner">

<!-- Table -->
<section>
  <h2>두더지 게임</h2>
  <p>A:동선 B:회로 C:제어 D:일반인</p>
  <div class="table-wrapper">
    <table>
      <table>
        <tr> <td> 플레이어 정보</td>
        <td> 점수 </td>
      <?php
        while($row=mysqli_fetch_array($result)){ ?>
        <tr>
          <td> <?=$row[0]?></td>
          <td> <?=$row[1]?></td>
        </tr>
      <?php
        }
      ?>
    </tr></tr>
  </table>
</table>
</div>
</section>
</section>

```

```

<!-- Footer -->
<footer id="footer" class="wrapper alt">
  <div class="inner">
    <ul class="menu">
      <li>&copy; Made by. 신예은 박예빈 강수빈 김정빈 황민혁 김세인</li>
    </ul>
  </div>
</footer>

<!-- Scripts -->
<script src="assets/js/jquery.min.js"></script>
<script src="assets/js/jquery.scrollex.min.js"></script>
<script src="assets/js/jquery.scrolly.min.js"></script>
<script src="assets/js/browser.min.js"></script>
<script src="assets/js/breakpoints.min.js"></script>
<script src="assets/js/util.js"></script>
<script src="assets/js/main.js"></script>
</body>
</html>

```

두더지 잡기 게임 점수 표시 창)

```

<?php
$link=mysqli_connect("localhost","jbt","jbt","score");
if (!$link)
{ echo "MySQL error : "; echo mysqli_connect_error(); exit(); }
mysqli_set_charset($link,"utf8");
$sql = "SELECT * FROM `olympic` ORDER BY `score` DESC LIMIT 6";
$result = mysqli_query($link, $sql);
?>
<!DOCTYPE HTML>
<html>
  <head>
    <title>올림픽 게임 점수 확인</title>
    <meta charset="utf-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1, user-scalable=no" />
    <link rel="stylesheet" href="assets/css/main.css" />
    <noscript><link rel="stylesheet" href="assets/css/noscript.css" /></noscript>
  </head>
  <body class="is-preload">

    <!-- Header -->
    <header id="header">
      <a href="index.php" class="title">O.M.G</a>
      <nav>

```

```

<ul>
  <li><a href="olympic.php">Olympic</a></li>
  <li><a href="mole.php">Mole</a></li>
</ul>
</nav>
</header>

```

```

<!-- Wrapper -->
<div id="wrapper">

```

```

  <!-- Main -->
  <section id="main" class="wrapper">
    <div class="inner">

```

```

      <!-- Table -->
      <section>
        <h2>올림픽 게임</h2>
        <p>A:불신 B:회로 C:제어 D:일반인</p>
        <div class="table-wrapper">
          <table>
            <table>
              <tr> <td> 플레이어 정보</td>
              <td> 점수 </td>
              <?php while($row=mysqli_fetch_array($result)){ ?>
              <tr>
                <td> <?=$row[0]?></td>
                <td> <?=$row[1]?></td>
              </tr>
              <?php }?>
              </tr> </tr>
            </table>
          </table>
        </div>
      </section>
    </section>

```

```

  <!-- Footer -->
  <footer id="footer" class="wrapper alt">
    <div class="inner">
      <ul class="menu">
        <li>&copy; Made by. 선예은 박예민 강수민 김경민 황민혁 김세인</li>
      </ul>
    </div>
  </footer>

```

```

<!-- Scripts -->
<script src="assets/js/jquery.min.js"></script>
<script src="assets/js/jquery.scrollex.min.js"></script>
<script src="assets/js/jquery.scrolly.min.js"></script>
<script src="assets/js/browser.min.js"></script>
<script src="assets/js/breakpoints.min.js"></script>
<script src="assets/js/util.js"></script>
<script src="assets/js/main.js"></script>
</body>
</html>

```

게임 메인소스)

```

#include <mega128a.h>
#include <stdlib.h>
#include <stdio.h>
#include <delay.h>
#include <lcd.h>

int cnt, flag=0, timer=0, timerCnt=0, startFlag=0, limitTimer=40, startcnt=0, mode=0,
modeFlag=0, start=0, end=0, lcd_cnt=0, lcd2_cnt=0;
unsigned char keyin, swin;
int led=0, keyflag=0, gameflag=0, clearFlag=0;

unsigned char userinfo[6] = {'W0','W0','W0','W0','W0'}; //비밀번호 입력 변수
unsigned char i = 0, j = 0;
unsigned char sbuf[50];

void enter(void);

#define KEY_DDR DDRD //D포트를 키 매트릭스로 사용
#define KEY_PORT PORTD
#define KEY_PIN PIND

interrupt [TIM2_OVF] void timer2_ovf_isr(void)
{
    TCNT2=0x16;

    if(mode==1){ //올림픽 게임 모드 : 1
        if(cnt>=1){ //입력이 시작될 때
            timerCnt++;
            if(timerCnt>100){ //1초씩 증가할 때
                timer++;
                timerCnt=0;
            }
        }
        if(timer==10){ //10초일 때

```

```

        flag=2;
    }
    if(timer>10){ //10초가 지났을 때
        timerCnt=0;
        timer=0;
        end=1;
        keyflag=1;
        gameflag=1;
    }
}

else if(mode==2){ //두더지 게임 모드 : 2
    if(startFlag==1){ //시작
        timerCnt++;
        if(timerCnt%limitTimer==0){ //led의 값을 랜덤으로 읽어 보내줌(조절가능)
            switch (led%7) {
                case 0: PORTF=0xfe; break;
                case 1: PORTF=0xfd; break;
                case 2: PORTF=0xfb; break;
                case 3: PORTF=0xf7; break;
                case 4: PORTF=0xef; break;
                case 5: PORTF=0xdf; break;
                case 6: PORTF=0xbf; break;
            }
        }
    }
    if(timerCnt>100){ //1초일때
        timer++;
        timerCnt=0;

        if(timer==10) limitTimer=20; //10초가 되면 led출력이 점점 더 빨라짐
    }
    if(timer==15){ //15초일 때
        flag=2;
    }
    if(timer>15){ //15초가 지났을 때
        timerCnt=0;
        flag=2;
        timer=0;
        end=1;
        keyflag=1;
        gameflag=1;
        startFlag=0;
    }
}
}

```

```

    }
}

char keyscan(void) { // 키메트릭스를 스캔하는 함수
    unsigned char i, i_value, pin_mix;

    KEY_DDR = 0x00;
    KEY_PORT = 0xFF;

    for(i=4; i<8; i++){
        KEY_DDR |= (1 << i);
        KEY_PORT &= ~(1<< i);

        switch (i){
            case 4:          i_value = 3;          break;
            case 5:          i_value = 2;          break;
            case 6:          i_value = 1;          break;
            case 7:          i_value = 0;          break;
            default:         break;
        }

        pin_mix = KEY_PIN & 0x0F;

        if(pin_mix != 0x0F){
            switch (pin_mix){
                case 14:      return (i_value*4 + 0);      break;
                case 13:      return (i_value*4 + 1);      break;
                case 11:      return (i_value*4 + 2);      break;
                case 7:       return (i_value*4 + 3);      break;
                default:      break;
            }
        }
        KEY_DDR = 0x00;
        KEY_PORT = 0xFF;
    }
}

```

```

char keysort(void) { //스캔한 키메트릭스 변수를 정리해서 도어락 숫자패드 크기인 3 x 4 로 정리

    unsigned char key = keyscan();

    switch(key) {
        case 0: return 10; break;
        case 1: return 11; break;
        case 2: return 12; break;

```



```

        case 3: return 13; break;
        case 4: return 0; break;
        case 5: return 1; break;
        case 6: return 2; break;
        case 7: return 3; break;
        case 8: return 4; break;
        case 9: return 5; break;
        case 10: return 6; break;
        case 11: return 7; break;
        case 12: return 8; break;
        case 13: return 9; break;
        case 14: return 14; break;
        case 15: return 15; break;

        default: return 16; break;
    }
}

void keydisplay(void) (    // LCD에 상태를 띄우는 함수

    unsigned char keyscanvalue;

    keyscanvalue = keysort();    // 정리한 값을 키스캔 값에 넣음

    lcd_clear();                // LCD 출력
    lcd_gotoxy(0,0);
    lcd_puts("ENTER THE CODE");

    lcd_gotoxy(0,1);
    sprintf(sbuf,"%X%X%X%X%X",userinfo[0],userinfo[1],userinfo[2],userinfo[3],userinfo[4]);
    lcd_puts(sbuf);
    delay_ms(10);

    if(keyscanvalue == 14) {    // delete 버튼 누를 시

        for( i = 0; i < 5; i++)
        {
            userinfo[i] = 'W0';
        }
        i = 0;                // 초기화
    }

    else if(keyscanvalue == 15)
    { // enter 버튼 누를 시
        enter(); // enter 함수 호출
    }
}

```

```
}
```

```
else if(keyscanvalue < 16) { // 그 외 1~9까지의 숫자들을 누를 시
```

```
    userinfo[i] = keyscanvalue; // 입력 배열 변수에 집어넣음.  
    delay_ms(300);
```

```
    if(i < 6) // 5글자만 보내기 위해 6글자 미만까지  
        i++;
```

```
    lcd_clear(); // LCD 출력  
    lcd_gotoxy(0,0);  
    lcd_puts("ENTER THE CODE");
```

```
    lcd_gotoxy(0,1);
```

```
    sprintf(sbuf, "[%X%X%X%X%X]",userinfo[0],userinfo[1],userinfo[2],userinfo[3],userinfo[4]); //  
    사용자가 입력한 값 출력
```

```
    lcd_puts(sbuf);
```

```
}
```

```
}
```

```
void enter(void) { // 사용자가 입력한 값을 확인하는 함수
```

```
    lcd_clear();  
    lcd_puts("PLEASE WAIT...");  
    delay_ms(1000);
```

```
    //USART 함수 만들어야 함
```

```
    for(j = 0; j < 5; j++)  
    {  
        userinfo[j] = 'W0';  
    }
```

```
    i = 0; // 배열 초기화  
    lcd_clear();  
    mode=0;  
    keyflag=0;  
    gameflag=0;
```

```
}
```

```
void main(void)
```

```
{
```

```

// Timer/Counter 2 initialization
// Clock source: System Clock
// Clock value: 15.625 kHz
// Mode: Normal top=0xFF
// OC2 output: Disconnected
// Timer Period: 10.048 ms
ASSR=0<<AS0;
TCCR2=(0<<WGM00) | (0<<COM01) | (0<<COM00) | (0<<WGM01) | (1<<CS02) |
(0<<CS01) | (1<<CS00);
TCNT2=0x16;
OCR2=0x00;

// Timer(s)/Counter(s) Interrupt(s) initialization
TIMSK=(0<<OCIE2) | (1<<TOIE2) | (0<<TICIE1) | (0<<OCIE1A) | (0<<OCIE1B) |
(0<<TOIE1) | (0<<OCIE0) | (0<<TOIE0);
ETIMSK=(0<<TICIE3) | (0<<OCIE3A) | (0<<OCIE3B) | (0<<TOIE3) | (0<<OCIE3C) |
(0<<OCIE1C);

DDRA=0xff;

DDRB=0x00;
PORTB=0xff;
DDRF=0xff;
PORTF=0xff;
DDRD=0x0f;
PORTD=0x00;

#asm("sei")

led_init(16);

while (1)
{
    keyin=PINB;

    led+ +;
    if(led==7) led=0;

    if(keyflag==1) keydisplay();    // 비밀번호 입력을 위해 UI 출력

    if(end==1){    //게임실행이 한 번 끝났을 때 변수의 초기화 값 재설정
        start=0;
        cnt=0;
        end=0;
        led_cnt=0;
    }
}

```

```

    lcd2_cnt=0;
    clearFlag=0;
}

if(keyin!=0xff | flag==0){    //스위치의 입력이 들어왔을 경우 모드 선택
    modeFlag=1;
    if(start==1) modeFlag=0;
}

if(mode==0){ //modeFlag==1
    swin=PINB;
    switch (swin)
    {
        case 0xfe: mode=1; break;
        case 0xfd: mode=2; break;
    }
}

if(gameflag==0){    //올림픽 게임 실행
    if(mode==1){
        start=1;

        if(end==1) {
            lcd_clear();
        }

        lcd_cnt++;
        if(lcd_cnt==1)lcd_clear();

        lcd_gotoxy(0,0);
        lcd_printf("TIME : %2d", 10 - timer);

        lcd_gotoxy(0,1);
        lcd_printf("SCORE = %02d",cnt);

        delay_ms(1);

        if(timer==10){    //게임이 끝나면 led로 알려주기 위해 빠르게 표시
            delay_ms(25);
            switch (led%7)
            {
                case 0: PORTF=0xfe; break;
                case 1: PORTF=0xfd; break;
                case 2: PORTF=0xfb; break;
            }
        }
    }
}

```

```

        case 3: PORTF=0xf7; break;
        case 4: PORTF=0xef; break;
        case 5: PORTF=0xdf; break;
        case 6: PORTF=0xbf; break;
    }
}

if(flag!=2){          //게임이 끝나지 않았을 때
    if(keyin!=0xff & flag==0){
        flag=1;
        cnt++;
    }

    if(PINB==0xff){
        flag=0;
    }
}

} else if(mode==2){    //두더지게임 실행
    start=1;

    if(end==1) {
        lcd_clear();
    }

    lcd2_cnt++;
    if(lcd2_cnt==1) lcd_clear();

    lcd_gotoxy(0,0);
    lcd_printf("TIME : %2d", 15 - timer);

    lcd_gotoxy(0,1);
    lcd_printf("SCORE = %02d",cnt);

    delay_ms(1);

    led++;
    if(led==7) led=0;

    if(timer==15){      //게임이 끝나면 led로 알려주기 위해 빠르게 표시
        delay_ms(25);
        switch (led%7)
        {
            case 0: PORTF=0xfe; break;
            case 1: PORTF=0xfd; break;
            case 2: PORTF=0xfb; break;

```

```

        case 3: PORTF=0xf7; break;
        case 4: PORTF=0xef; break;
        case 5: PORTF=0xdf; break;
        case 6: PORTF=0xbf; break;
    }
}

if(flag!=2){          //게임이 끝나지 않았을 때
    if(keyin!=0xff & flag==0){
        startFlag=1;
        flag=1;
        startcnt++;

        if(keyin==PINF){    //두더지 위치의 스위치를 누르면 점수 증가
            cnt++;
            PORTF=0xff;
        }else if(startcnt==1){    //첫 시작 점수는 0으로 초기화
            cnt=0;
        }else cnt--;           //잘못 누를 경우 점수 감소
    }

    if(PINB==0xf0){
        flag=0;
    }
}

} else { //mode 0일때 : 초기화면
    PORTF=0xff;

    if(end==1) delay_ms(5000);
    flag=0;

    clearFlag++;
    if(clearFlag==1){
        lcd_clear();
    }
    lcd_gotoxy(0,0);
    lcd_printf("GAME SELECT");

    lcd_gotoxy(0,1);
    lcd_printf("1:OGAME 2:MGAME");
}
}

```