

---

# 아두이노 시뮬레이션 프로그래밍

---

v1.0

김영준 저

공학박사, 목원대학교 겸임교수  
前 Microsoft 수석연구원

헬로앱스

<http://www.helloapps.co.kr>

## 04 틱톡 스위치 만들기

### 학습 목표

- 버튼의 눌림 상태를 감지할 수 있는 기능을 구현할 수 있다.
- 버튼이 눌리거나 눌림이 해제될 때 반짝 반짝 LED가 점멸하는 기능을 구현할 수 있다.

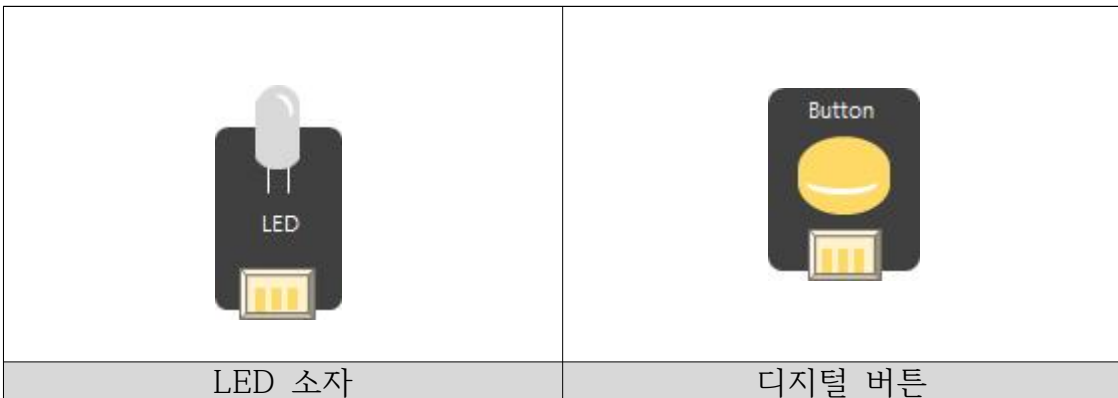
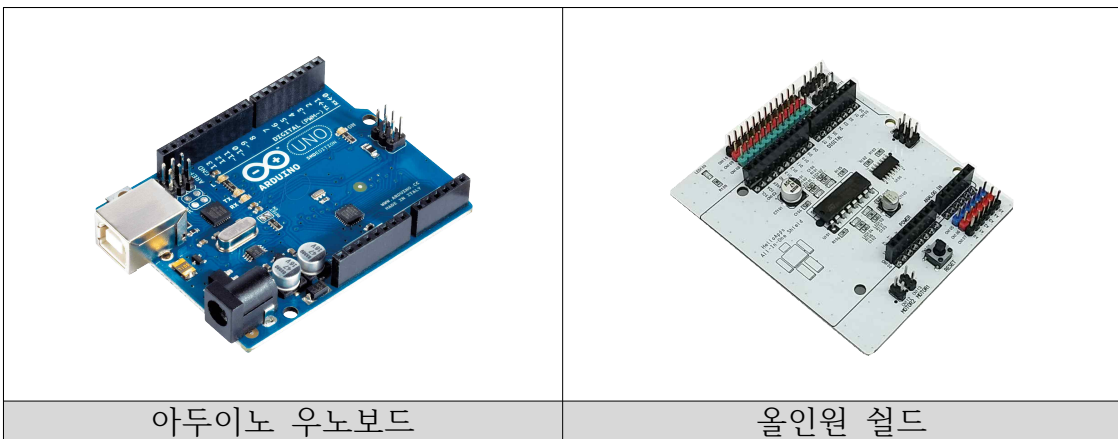
### 실습 개요

- 전역변수를 이용하여 버튼이 눌러져 있는 지 감지하는 기능을 구현한다.
- 버튼이 눌러지거나 올라오는 순간을 감지하는 기능을 구현한다.
- LED를 이용하여 버튼이 눌리거나 올라올 때 다른 LED가 켜지도록 한다.

## 4.1 준비하기

### 준비물

- 아래 준비물은 실제 아두이노 키트가 있는 경우를 예로 든 것이며, 아래 하드웨어 구성과 동일한 내용을 시뮬레이션으로 진행하게 된다.
- 아두이노 보드, 올인원 쉴드, 디지털 버튼 2개, LED 2개



시뮬레이션 상에서의 연결 정보

- 시뮬레이션 상에서는 디지털 LED 소자가 각각 디지털 11번, 12번, 13번에 연결되어 있으며, 버튼은 디지털 4번 ~ 8번에 연결되어 있다.

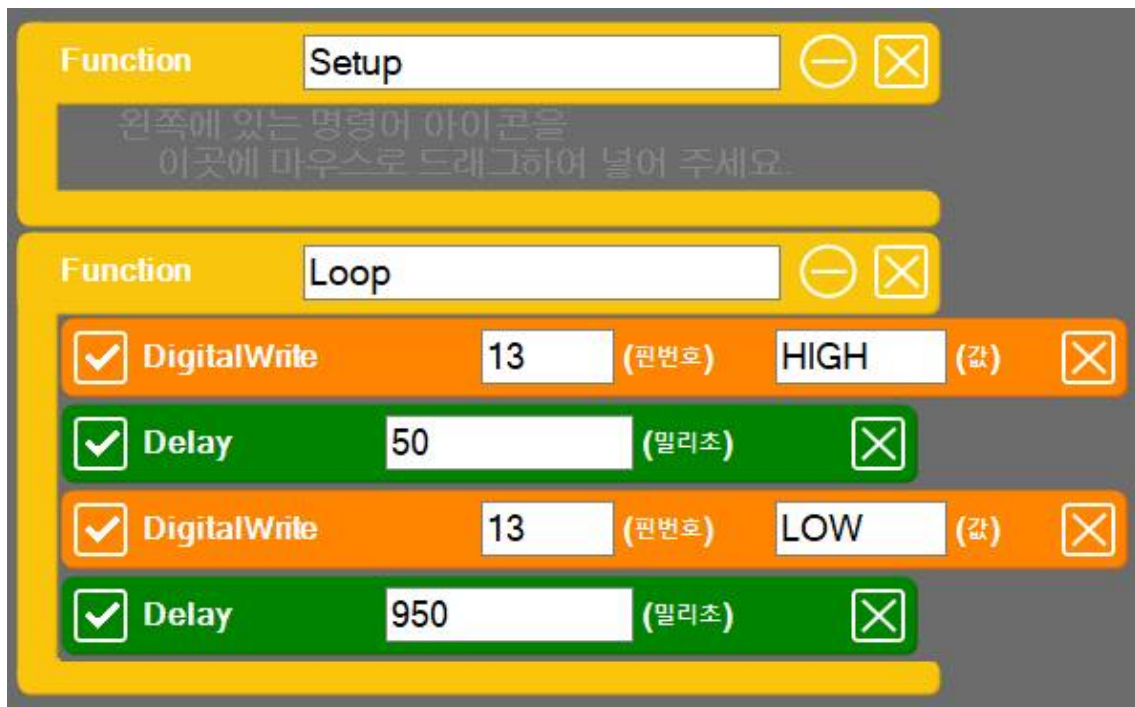


- 디지털 핀에 연결된 부품
  - 디지털 2번: 서보 모터
  - 디지털 4번 ~ 8번: 버튼 센서
  - 디지털 9번: 스피커
  - 디지털 11번: 초록색 LED
  - 디지털 12번: 노란색 LED
  - 디지털 13번: 빨간색 LED

## 4.2 반짝 반짝 LED 점멸 효과 만들기

### LED 짧게 점멸시키기

- 13번에 연결된 LED를 1초에 한 번씩 아주 짧게 점멸되는 기능을 구현해 본다.
- 1초 동안 중에서 50밀리초는 LED가 13번 켜져 있고, 나머지 950밀리초 동안에는 LED가 꺼져 있도록 구현해 본다.



SPL 스크립트	스케치 코드
<pre> void setup() { }  void loop() {     DigitalWrite(13, HIGH)     Delay(50)     DigitalWrite(13, LOW)     Delay(950) }                 </pre>	<pre> void setup() {     pinMode(13, OUTPUT); }  void loop() {     digitalWrite(13, HIGH);     delay(50);     digitalWrite(13, LOW);     delay(950); }                 </pre>

- 실행하여 결과를 확인해 본다.

반짝 반짝 두 번 점멸하기

- 이번에는 1초 간격으로 빠르게 반짝 반짝 두 번 점멸하는 기능을 구현해 본다. 앞에서 구현한 한 번 반짝이는 기능을 2번 반짝이는 기능으로 수정해 보는 것이다.
- 50밀리초 주기로 2번 빠르게 점멸하고 나머지 850 밀리초 동안 꺼져 있도록 기능을 수행한다.



SPL 스크립트	스케치 코드
<pre> void setup() { }  void loop() {     DigitalWrite(13, HIGH)     Delay(50)     DigitalWrite(13, LOW)     Delay(50)     DigitalWrite(13, HIGH)     Delay(50)     DigitalWrite(13, LOW)     Delay(850) }                 </pre>	<pre> void setup() {     pinMode(13, OUTPUT); }  void loop() {     digitalWrite(13, HIGH);     delay(50);     digitalWrite(13, LOW);     delay(50);     digitalWrite(13, HIGH);     delay(50);     digitalWrite(13, LOW);     delay(850); }                 </pre>

- 실행하여 결과를 확인해 본다.



### 4.3 경광등 장치 만들기

3개의 LED를 반짝이게 하기

- 3개의 LED를 연속적으로 반짝이게 하도록 해본다.
- 1초 시간 동안에 50밀리초 간격으로 각각 13번, 12번, 11번 LED를 순서대로 점멸시킨다.



SPL 스크립트	스케치 코드
<pre> void setup() { }  void loop() {     DigitalWrite(13, HIGH)     Delay(50)     DigitalWrite(13, LOW)     Delay(50)     DigitalWrite(12, HIGH)     Delay(50)     DigitalWrite(12, LOW)     Delay(50)     DigitalWrite(11, HIGH)     Delay(50)     DigitalWrite(11, LOW)     Delay(750) }         </pre>	<pre> void setup() {     pinMode(13, OUTPUT);     pinMode(12, OUTPUT);     pinMode(11, OUTPUT); }  void loop() {     digitalWrite(13, HIGH);     delay(50);     digitalWrite(13, LOW);     delay(50);     digitalWrite(12, HIGH);     delay(50);     digitalWrite(12, LOW);     delay(50);     digitalWrite(11, HIGH);     delay(50);     digitalWrite(11, LOW);     delay(750); }         </pre>

- 실행하여 결과를 확인해 본다.

경광등 장치 만들기

- 다음의 과정을 1초 간격으로 반복해 본다.
  - 13번 LED를 빠르게 2번 점멸
  - 11번 LED를 빠르게 2번 점멸

The image shows a Scratch-style code editor with a 'Loop' block. The code consists of the following sequence of blocks:

- Block 1:** DigitalWrite (13, HIGH) - 50ms Delay
- Block 2:** DigitalWrite (13, LOW) - 50ms Delay
- Block 3:** DigitalWrite (13, HIGH) - 50ms Delay
- Block 4:** DigitalWrite (13, LOW) - 50ms Delay
- Block 5:** DigitalWrite (11, HIGH) - 50ms Delay
- Block 6:** DigitalWrite (11, LOW) - 50ms Delay
- Block 7:** DigitalWrite (11, HIGH) - 50ms Delay
- Block 8:** DigitalWrite (11, LOW) - 650ms Delay

SPL 스크립트	스케치 코드
<pre> void setup() { }  void loop() {     DigitalWrite(13, HIGH)     Delay(50)     DigitalWrite(13, LOW)     Delay(50)     DigitalWrite(13, HIGH)     Delay(50)     DigitalWrite(13, LOW)     Delay(50)     DigitalWrite(11, HIGH)     Delay(50)     DigitalWrite(11, LOW)     Delay(50)     DigitalWrite(11, HIGH)     Delay(50)     DigitalWrite(11, LOW)     Delay(650) }                 </pre>	<pre> void setup() {     pinMode(13, OUTPUT);     pinMode(11, OUTPUT); }  void loop() {     digitalWrite(13, HIGH);     delay(50);     digitalWrite(13, LOW);     delay(50);     digitalWrite(13, HIGH);     delay(50);     digitalWrite(13, LOW);     delay(50);     digitalWrite(11, HIGH);     delay(50);     digitalWrite(11, LOW);     delay(50);     digitalWrite(11, HIGH);     delay(50);     digitalWrite(11, LOW);     delay(650); }                 </pre>

- 실행하여 결과를 확인해 본다.

## 나만의 경광등 장치 만들기

---

- 11번 LED, 12번 LED, 13번 LED 3개를 모두 사용하여 나만의 창의적인 경광등 장치를 만들어 본다.

#### 4.4 버튼의 눌림 상태를 전역변수로 저장하기

pressed 변수 선언

- 버튼이 현재 눌러져 있는 지 아니면 그렇지 않은지 상태를 저장하고 있는 전역변수를 다음과 같이 추가해 보자.

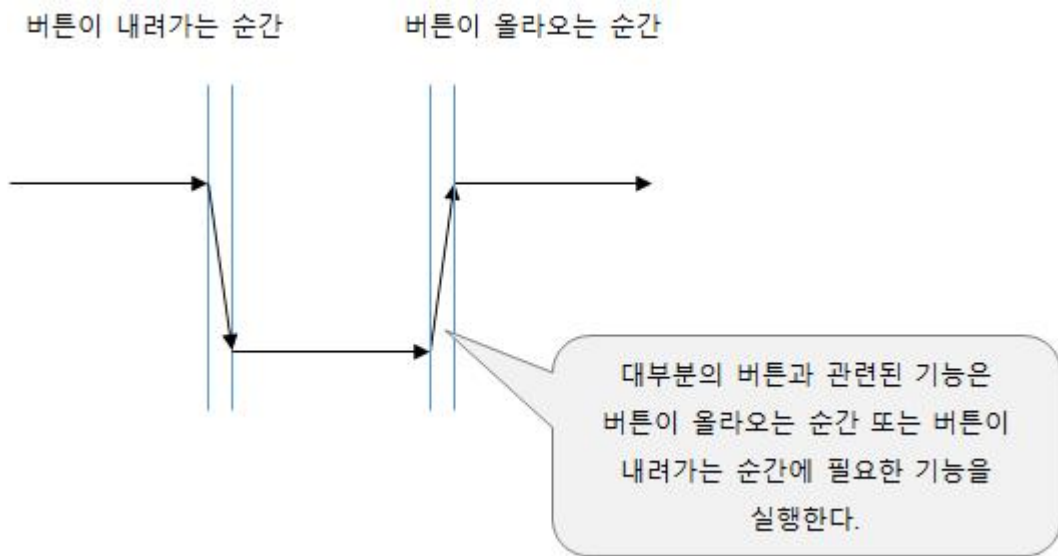


SPL 스크립트	스케치 코드
<pre> pressed = false  void setup() { }  void loop() {   d4 = DigitalRead(4)   if ( d4 == HIGH )   {     pressed = true   }   else   {     pressed = false   }    PrintLine(pressed)   Delay(100) } </pre>	<pre> boolean pressed = false;  void setup() {   pinMode(4, INPUT);   Serial.begin(115200); }  void loop() {   int d4 = digitalRead(4);   if ( d4 == HIGH )   {     pressed = true;   }   else   {     pressed = false;   }    Serial.println(pressed);   delay(100); } </pre>

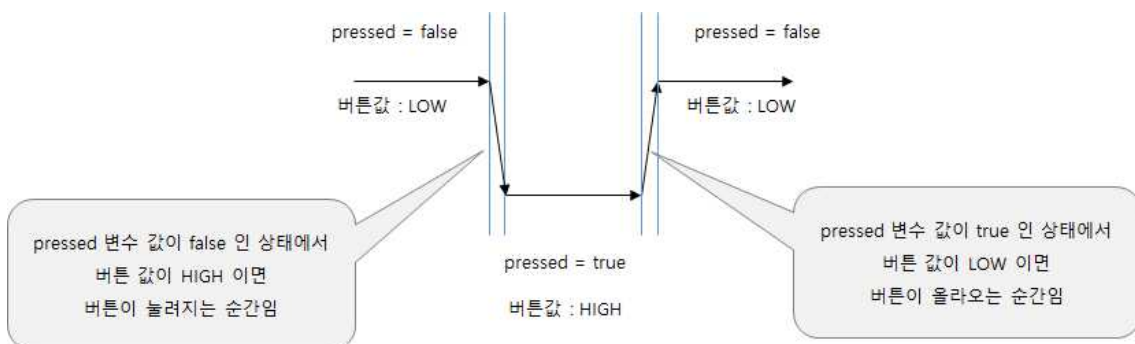
## 4.5 버튼이 눌러지는 순간과 올라오는 순간을 감지하기

### 버튼의 상태 변화 감지

- 버튼이 눌러지는 순간에 한번 “Down” 이라는 메시지를 출력하고, 버튼이 올라오는 순간에 한번 “Up” 이라는 메시지를 출력하려고 한다.



- 버튼의 상태가 변하는 순간을 감지하기 위해서는 버튼의 상태를 저장하고 있는 전역변수의 값을 비교해야 한다. 이전 활동에서 사용한 pressed와 같은 역할의 전역 변수 값을 비교하여 버튼이 눌러 지거나 올라오는 순간을 감지할 수 있다.





버튼의 상태 변화 감지 코드

- 버튼이 눌러지는 순간과 올라오는 순간을 감지하여 메시지를 표시하는 코드는 다음과 같다.

```

    Expression: pressed = false
    Function: Loop
    d4 == DigitalRead 4
    if d4 == HIGH
    if pressed == false
    PrintLine "Down"
    Expression: pressed = true
    else
    if pressed == true
    PrintLine "Up"
    Expression: pressed = false
    Delay 100 (밀리초)
    
```

SPL 스크립트	스케치 코드
<pre> pressed = false void setup() { }  void loop() {   d4 = DigitalRead(4)   if ( d4 == HIGH )   {     if (pressed == false)       PrintLine("Down")      pressed = true   }   else   {     if (pressed == true)       PrintLine("Up")      pressed = false   }    Delay(100) } </pre>	<pre> boolean pressed = false; void setup() {   pinMode(4, INPUT);   Serial.begin(115200); }  void loop() {   int d4 = digitalRead(4);   if ( d4 == HIGH )   {     if (pressed == false)       Serial.println("Down")      pressed = true;   }   else   {     if (pressed == true)       Serial.println("Up");      pressed = false;   }    delay(100); } </pre>

## 4.6 반짝 반짝 스위치 구현하기

반짝 반짝 기능 추가하기

- 이전 활동의 코드들을 활용하여 버튼이 눌리는 순간 50밀리초 간격으로 13번 LED가 한번 점멸하고, 버튼이 올라오는 순간 11번 LED가 50밀리초 간격으로 한번 점멸하도록 기능을 추가해 본다.

The image shows a Scratch-style code editor with the following blocks:

- Expression:** `pressed = false`
- Function:** `Loop`
- DigitalRead:** `d4 = DigitalRead 4` (핀번호)
- if:** `d4 == HIGH`
  - if:** `pressed == false`
    - DigitalWrite:** `13` (핀번호) `HIGH` (값)
    - Delay:** `50` (밀리초)
    - DigitalWrite:** `13` (핀번호) `LOW` (값)
  - Expression:** `pressed = true`
- else:**
  - if:** `pressed == true`
    - DigitalWrite:** `11` (핀번호) `HIGH` (값)
    - Delay:** `50` (밀리초)
    - DigitalWrite:** `11` (핀번호) `LOW` (값)
  - Expression:** `pressed = false`

- Delay:** `100` (밀리초)

SPL 스크립트	스케치 코드
<pre> pressed = false  void setup() { }  void loop() {   d4 = DigitalRead(4)   if ( d4 == HIGH )   {     if (pressed == false)     {       DigitalWrite(13, HIGH)       Delay(50)       DigitalWrite(13, LOW)     }      pressed = true   }   else   {     if (pressed == true)     {       DigitalWrite(11, HIGH)       Delay(50)       DigitalWrite(11, LOW)     }      pressed = false   }    Delay(100) } </pre>	<pre> boolean pressed = false; void setup() {   pinMode(4, INPUT);   pinMode(11, OUTPUT);   pinMode(13, OUTPUT);   Serial.begin(115200); }  void loop() {   int d4 = digitalRead(2);   if ( d4 == HIGH )   {     if (pressed == false)     {       digitalWrite(13, HIGH);       delay(50);       digitalWrite(13, LOW);     }      pressed = true;   }   else   {     if (pressed == true)     {       digitalWrite(11, HIGH);       delay(50);       digitalWrite(11, LOW);     }      pressed = false;   }    delay(100); } </pre>