**Lab9. 인터럽트**

# **예제1.순차적으로 LED가 순차적으로 켜지다가 스위치를 누르는 동안 꺼지는 프로그램**

|  |  |  |
| --- | --- | --- |
| **구분** | **설명** | **코드** |
| pinMode(pin,Input\_PULLUP)  digitalPinToInterrrupt(pin)  attachInterrupt(interrupt,ISR,mode) | 버튼을 누를때 입력이 되는 핀을 설정합니다.  특정 핀을 Interrupt 핀으로 사용하고, 인터럽트 번호를 반환합니다.(2 또는 3)  특정 인터럽트의 입력 방식과 실행할 함수 , 트리거를. 설정합니다.  (interrupt 번호 , 함수, 트리거) | #define interruptPin 2  pinMode(interruptPin,INPUT\_PULLUP);  attachInterrupt  (digitalPinToInerrupt(interruptPin),ledoff  ,FALLING); |
|  |  |  |

# **예제 2. 1초 단위로 LED를 토글하는 프로그램**

|  |  |  |
| --- | --- | --- |
| **구분** | **설명** | **코드** |
| #include<MsTimer2.h>  MsTimer2::set(time,mode)  MsTimer2::start()  MsTimer2::stop()  toggle | Timer Interrupt를 사용하기 위한 라이브러리 호출  Timer Interrupt를 사용할때 필요한 timer와 그때 동작할 함수를 설정  Timer Interrupt를 start합니다.  Timer Interrupt를 stop 합니다.  Led토글 함수를 생성합니다. | #include<MsTimer2.h>  MsTimer2::set(1000,toggle);  MsTimer2::start();  MsTimer2::stop();  void toggle(){  LED = !LED;  } |

|  |
| --- |
| **고찰 및 실험 평가** |

**1. 순차적으로 LED가 켜지다가 스위치를 누르고 있으면 LED가 꺼지는 프로그램**

**소스 코드**

#define LED1 7

#define LED2 8

#define LED3 9

#define interruptPin 2

void setup() {

pinMode(LED1,OUTPUT);

pinMode(LED2,OUTPUT);

pinMode(LED3,OUTPUT);

pinMode(interruptPin,INPUT\_PULLUP);

attachInterrupt(digitalPinToInterrupt(interruptPin),ledoff,FALLING);

}

void loop() {

digitalWrite(LED3,LOW);

digitalWrite(LED1,HIGH);

delay(1000);

digitalWrite(LED1,LOW);

digitalWrite(LED2,HIGH);

delay(1000);

digitalWrite(LED2,LOW);

digitalWrite(LED3,HIGH);

delay(1000);

}

void ledoff(){

digitalWrite(LED1,LOW);

digitalWrite(LED2,LOW);

digitalWrite(LED3,LOW);

}

**회로 구성**

전자기기, 회로이(가) 표시된 사진

자동 생성된 설명

**2. 1초단위로 LED를 토글하는 프로그램**

**소스 코드**

#include <MsTimer2.h>

#define LED1 7

#define LED2 8

#define LED3 9

int LED = 0;

void setup() {

pinMode(LED1,OUTPUT);

pinMode(LED2,OUTPUT);

pinMode(LED3,OUTPUT);

MsTimer2::set(1000,toggle);

MsTimer2::start();

}

void loop() {

digitalWrite(LED1,LED);

digitalWrite(LED2,LED);

digitalWrite(LED3,LED);

}

void toggle(){

LED = !LED;

}

**회로 구성**

**전자기기, 회로이(가) 표시된 사진

자동 생성된 설명**

**3. 스탑워치 프로그램**

**소스 코드**

#include <MsTimer2.h>

#define int0 2

#define int1 3

int echo = 1; // echo default ON

char text; // usart serial input

double elapsedtime = 0; // stopwatch data

int SWstate = 0; // stopwatch default OFF

void setup() {

pinMode(int0, INPUT\_PULLUP);

attachInterrupt(digitalPinToInterrupt(int0), echotoggle, FALLING);

pinMode(int1, INPUT\_PULLUP);

attachInterrupt(digitalPinToInterrupt(int1), stopwatch, FALLING);

MsTimer2::set(100, timecount);

Serial.begin(9600);

while(!Serial);

}

void loop() {

if(Serial.available())

{

text = Serial.read();

if(echo)

Serial.write(text);

}

}

void echotoggle(){

if(!echo)

Serial.println("Echo ON");

else

Serial.println("Echo OFF");

echo = !echo;

}

void stopwatch(){

if(!SWstate)

{

MsTimer2::start();

Serial.println("Stopwatch Begin");

}

else

{

MsTimer2::stop();

Serial.print("Elapsed Time : ");

Serial.println(elapsedtime);

elapsedtime = 0;

}

SWstate = !SWstate;

}

void timecount(){

elapsedtime += 0.1;

}

**회로 구성**

**전자기기, 회로이(가) 표시된 사진

자동 생성된 설명**

**실험 고찰**

이번 실험에서는 인터럽트를 이용하여 순차적으로 켜지는 LED에 스위치를 누르고 있으면 꺼지는 프로그램과 타이머 인터럽트를 사용하여 1초 단위로 LED를 토글하는 프로그램, 그리고 이들을 모두 이용하여, 스탑워치를 만드는 프로그램을 실습하였다.

먼저 첫번째 실험에서는 인터럽트를 이용하여 LED가 1초간격으로 하나씩 꺼지는 루프문에서, 버튼이 입력되면, falling 상태에서 인터럽트가 된 후ledoff 함수가 동작하는 원리로 코드를 구현하였다. 두번째 실험에서는 타이머 인터럽트 라이브러리를 이용하여 타이머 인터럽트를 사용하였다. msTimer::set을 이용하여, 1초간격으로 토글을 하는 함수가 인터럽트 되게 하여, 1초 단위로 LED를 토글하도록 구성하였다. 세번째 실험에서는 이들을 모두 종합적으로 이용하여, 스탑워치를 만드는 프로그램을 작성하였다. 처음에 인터럽트를 무조건 사용해야한다는 생각에 루프문 안에서 대부분의 함수들을 인터럽트로 사용하여 구성하였지만 번번이 오류가 발생하였다. 인터럽트는 루프문에 영향을 미치기 때문에, 함수에서 특정한 기능만을 사용하여 구성을 하여야 한다는것을 알게 되었다. 이를 이용하여 위의 코드를 살펴 보면, 토글을 하는 함수, 시간을 측정하는 함수, 스탑워치의 기능을 동작하는 함수, 총 3가지로 구성하였고, 타임 카운터를 이용하여, 100ms 마다 초를 측정할 수 있게 시간을 측정하는 함수를 구성하였다. 그리고 스탑워치 함수에서 버튼 input이 들어왔을때, SWstate변수가 0일때 타이머 인터럽트가 동작하게 하였고, SWstate가 0이 아닐때 타이머 인터럽트가 멈추고, 시간을 호출하여 시리얼 모니터에 보여지게 코드를 만든 것을 확인 할 수 있었다.