

아두이노 프로그래밍

# 6차 과제 문제 풀이

14장 인터럽트 고찰문제 11~13번

2020년 1학기

서경대학교 김진헌

# 11번

1

pwm\_based\_Periodic\_interrupt.ino

11. analogWrite() 함수는 390Hz의 PWM 펄스를 발생시키는 것으로 알려져 있다. 1) 이를 이용해 390Hz의 periodic interrupt를 발생시키는 방안을 기술하라. 2) 이 주기적 인터럽트를 이용하면 시계를 만들 수 있다. PWM 펄스를 이용하여 스톱워치 프로그램을 작성하고 실험한 결과를 제시하시오.

Arduino uno의 경우 analogWrite() 함수는 단자 5번은 490Hz, 단자 6번은 980Hz의 인터럽트를 발생한다.

<https://www.arduino.cc/reference/en/language/functions/analog-io/analogwrite/>

// 본 실험에서는 6번 단자로 980Hz의 펄스를 발생하게 만든다.

```
#define PWM_PIN 6  
analogWrite(PWM_PIN, 128); // duty rate 50%, 980Hz
```

// 6번 단자를 jumper 선을 이용하여 2번에 연결하여 980x2 Hz의 인터럽트를 발생하게 만든다.

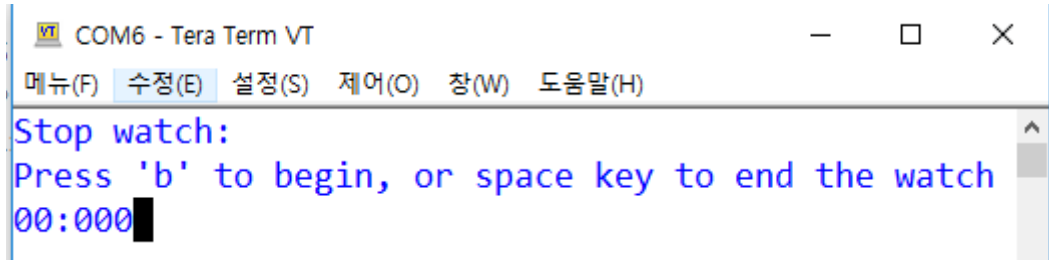
```
#define INT_PIN 2  
#define Mode CHANGE  
attachInterrupt(digitalPinToInterrupt(INT_PIN), ISR, mode)
```



# 11번

2

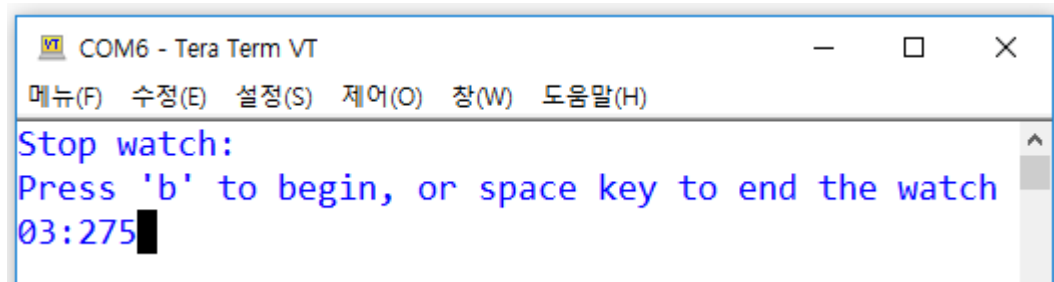
pwm\_based\_Periodic\_interrupt.ino



VT COM6 - Tera Term VT

메뉴(F) 수정(E) 설정(S) 제어(O) 창(W) 도움말(H)

Stop watch:  
Press 'b' to begin, or space key to end the watch  
00:000



VT COM6 - Tera Term VT

메뉴(F) 수정(E) 설정(S) 제어(O) 창(W) 도움말(H)

Stop watch:  
Press 'b' to begin, or space key to end the watch  
03:275

# 11번

```
94 void increment() {  
95     if (holding == false)    // 멈추어야 하는가?  
96         count++;  
97 }
```

3

pwm\_based\_Periodic\_interrupt.ino

```
18 #define PWM_PIN 6    // 6번과 2번을 jumper cable로 서로 연결한다.  
19 #define INT_PIN 2    // PWM_PIN과 INT_PIN 단자는 서로 연결되어 있다.  
20 #define mode CHANGE  
21 #define MS_500 98  
22 #define SEC_1 980*2  
23  
24 const byte ledPin = 13;    // 1초에 한 번씩 내장 LED를 점멸시킬 예정  
25 volatile unsigned int count = 0;  
26 volatile int sec=0, ms=0, old_ms=1;  
27 volatile bool holding = true;    // 시간 출력 중지  
28  
29 void setup() {  
30     analogWrite(PWM_PIN, 128);    // duty rate 50%, 980Hz의 pulse를 PWM 단자로 출력한다.  
31     attachInterrupt(digitalPinToInterrupt(INT_PIN), increment, mode);  
32     //Serial.begin(115200);    // max speed  
33     Serial.begin(9600);  
34     //pinMode(INT_PIN, INPUT_PULLUP);    // 꼭 필요한 조치는 아님.  
35     pinMode(ledPin, OUTPUT);  
36     Serial.println("Stop watch:\nPress 'b' to begin, or space key to end the watch");  
37     //Serial.print("00");    // 현재 초 보여주며 대기...  
38 }
```

# 11번

4

pwm\_based\_Periodic\_interrupt.ino

```
40 void loop() {
41   unsigned long t1, t2;
42   int tmp1, tmp2;
43   char ch;
44   if(Serial.available() > 0) {
45     ch=Serial.read();
46     switch(ch) {
47       case 'b': // begin, or start
48         sec = 0;
49         ms = 0;
50         count = 0;
51         holding = false; // 시간 갱신을 계속하라.
52         break;
53       case ' ': // end, or stop
54         holding = true; // 시간 출력을 멈춘다.
55         count = 0;
56         break;
57       default:
58         break;
59     }
60 }
```

```
61 #define MS_25 49 // 25ms
62 // #define MS_100 98*2 // 100ms
63 if(holding == false) {
64   if (count >= MS_25) {
65     ms += 25;
66     count = 0;
67   }
68   if ( ms >= 1000) {
69     sec += 1;
70     ms = 0;
71   }
72 }
73
74 if (ms != old_ms) {
75   Serial.write('\r');
76   Serial.write(0x30+sec/10);
77   Serial.write(0x30+sec%10);
78   Serial.write(':');
79   Serial.write(0x30+ms/100);
80   Serial.write(0x30+(ms/10)%10);
81   Serial.write(0x30+(ms%10));
82   old_ms = ms;
83 }
84 }
```

# 12번

5

12. 클럭 소스(MsTimer2)를 이용하여 1,000Hz 인터럽트를 발생시킨다. 이를 이용하여 지정한 주파수의 음정을 생성하는 myTone(pin, frequency)과 myNoTone() 함수를 제작한다. 이 함수를 기반으로 직렬 통신(시리얼 모니터)으로 입력받은 주파수를 2초간 출력하는 프로그램을 설계하시오. 생성할 수 없는 주파수를 입력하면 범위 밖이라고 거절하여야 하고, 입력한 주파수를 시리얼 모니터에 알려주어야 한다. 지정하지 않은 여타의 조건에 대해서는 개인의 판단과 창의력에 따라 자신이 설정한 조건으로 수행한다.

7차 레포트 Problem 1으로 설명으로 같음합니다.

MsTimer2: 1ms 기반의 인터럽트 발생 => tone() 생성 주파수 범위의 한계(Max. 500Hz)  
TimerOne 라이브러리 기반의 tome() 함수 제작: 1us 기반의 인터럽트 발생 (Max. 500KHz)  
⇒ Source 배포      TimerOne\_base\_Tone.ino

# 13번

6

13. 20KHz의 클럭 소스를 이용하여 인터럽트를 발생시키고자 한다. 이를 기반으로 tone() 함수를 설계하고 한 옥타브의 음정을 연주하고자 한다. 위 과정에 대한 타당성 및 실행 방안을 다음 관점에서 기술하시오.

1. 20KHz의 클럭 소스를 타이머로부터 생성하는 방법의 가능성 타진

1. 아래의 timerOne 라이브러리는 1us 단위의 인터럽트를 발생시킬 수 있으므로 최대 500,000Hz의 음정을 만들어 낼 수 있다.

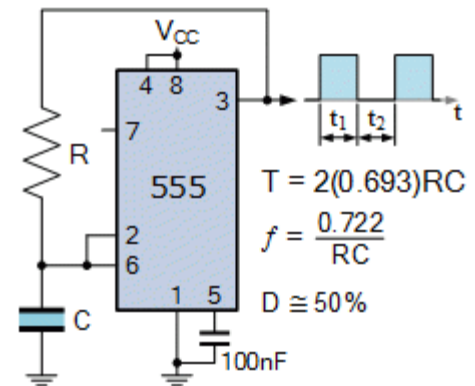
참조 링크: <https://github.com/PaulStoffregen/TimerOne>

2. 20KHz의 주파수를 가진 외부 OSC 소자 혹은 H/W 회로를 이용하여 클럭소스를 확보하는 방법



oscillator

555 timer



3. 1혹은 2의 방법으로 클럭소스가 확보되었다고 하고 자체 tone() 함수를 제작하시오. 7차 레포트 Problem 1 설명으로 같음.

TimerOne 라이브러리 기반의 tome() 함수 제작 => Source 배포

TimerOne\_base\_Tone.ino