<**타이머/카운터**>

1. **목적**

- ATMEGA128을 이용해서 실생활에 활용 가능한 프로그램을 만들어보자

1. **코딩**

* **타이머/카운터를 이용해서 비밀번호를 해체하는 프로그램이다**
* **구동과정**

1. **1초가 지날 때마다 LED의 불이 왼쪽에서부터 오른쪽으로 이동하도록한다.**
2. **스위치를 누르면은 FND에 값이 입력된다.**
3. **각각의 스위치마다 입력되는 값을 다르게 설정한다.**

**=>도어락에서 숫자버튼을 눌려서 입력하는 것을 구현**

1. **FND에 값을 입력하는 과정에서 8초가 지나게 될 경우(LED의 불이 한번씩 들어와서 마지막 LED에 불이 들어올 때) 입력이 실패하게 되고 FND에 입력된 모든 값들이 초기화되고 모든 LED에 불들이 깜빡 거리기 시작합니다.**

**=>도어락에서 비밀번호를 해체할 때 시간이 지나면은 해체가 안되고 비밀번호가 초기화되는 것을 모습을 구현**

1. **FND에 입력되는 값이 설정해둔 값이 아닌 다른 값이 입력 될 경우 입력이 실패하게 되어 FND에 입력된 모든 값들이 초기화되고 모든 LED에 불들이 깜빡 거리기 시작합니다.**

**=>도어락에서 비밀번호 입력이 잘못되었을 다시 비밀번호를 입력하는 것을 보여준다.**

1. **FND에 입력되는 값이 설정해둔 값과 일치하는 경우에는 모든 LED의 불들이 켜지도록 한다**

**=> 도어락에 설정된 비밀번호가 입력되었을 때 잠금이 해체되는 것을 보여준다.**

**<소스코드>**

**#define F\_CPU 7372800UL //avr크리스털 주파수**

**#include <avr/io.h>**

**#include <avr/interrupt.h>**

**#include <util/delay.h>**

**static unsigned char pattern[8]={0xfe,0xfd,0xfb,0xf7,0xef,0xdf,0xbf,0x7f};**

**int i=0,count = 0, count2 = 0, flag = 0, tmp=0, correct = 1;**

**int input[4] = {0}; //비밀번호 저장할 공간**

**// LED가 1초마다 이동한다.**

**ISR(TIMER0\_COMP\_vect) //비교일치 인터럽트 서비스 루틴**

**{**

**if(flag==1)**

**{**

**static int index=0;**

**OCR0 = 71; // 1/7372800\*1024분주\*(1+72)=0.01 10msec**

**count++; // count 값 증가**

**if (count==100) // 1초 경과**

**{**

**// LED 왼쪽으로 쉬프트**

**count=0;**

**PORTA = pattern[index++];**

**if(index==8)index=0;**

**count2++;**

**}**

**// 8초가 되었을 때 입력된 값이 초기화되고 LED가 깜빡거림**

**if (count2 == 8) // 총 8초가 되었을 때**

**{**

**for(int j=0;j<4;j++)**

**{**

**PORTA=0xff; //led 깜빡임**

**\_delay\_ms(200);**

**PORTA=0x00;**

**\_delay\_ms(200);**

**}**

**PORTA=0xff; //led 꺼짐**

**i=0,count=0,index=0, count2=0, flag = 0, tmp=0, correct = 1; //모든것 초기화**

**}**

**}**

**}**

**int main(void)**

**{**

**DDRA=0xff; //PORTA 출력(led)**

**DDRD = 0xf0; //PORTD 상위비트 출력(트렌지스터)**

**DDRC=0xff; //fnd 출력**

**PORTA=0xff; //led끔**

**TCCR0 = 0x0f; // CTC 모드, 분주비 1024, TOP값 OCRn**

**OCR0 = 71; // 1/7372800\*1024분주\*(1+72)=0.01 10msec**

**TIMSK = 0x02; // 출력 비교일치 인터럽트 허용**

**sei(); //전역 인터럽트 허용**

**int x=0;**

**int password[4] = {0x01,0x02,0x04,0x08}; //비밀번호 설정**

**//스위치를 눌려서 FND에 값을 입력**

**while (1)**

**{**

**for(i=0;i<4;i++)**

**{**

**while(!(~PIND&0x0f)); // 스위치 누름을 기다림**

**input[i] = (~PIND&0x0f); // 스위치 입력값을 input에 넣음**

**PORTD=~0x10; //PD4에 fnd트렌지스터 연결했기때문에**

**if(input[i]==0x01)PORTC=~0x06; //fnd 1~4**

**else if(input[i]==0x02)PORTC=~0x5b;**

**else if(input[i]==0x04)PORTC=~0x4f;**

**else if(input[i]==0x08)PORTC=~0x66;**

**\_delay\_ms(500);**

**if(input[i]!=0) flag = 1; // 스위치가 입력 되었을 시 flag 1로**

**// 스위치를 누르지 않을 경우**

**tmp++;**

**if(tmp==4) //스위치 4번 누르면**

**{**

**for(x=0;x<4;x++)**

**{**

**correct = correct && (input[x]==password[x]); // 비밀번호 4자리 하나하나 비교**

**}**

**if (correct==1) // 비밀번호가 모두 일치할 경우**

**{**

**PORTA = 0x00; //led 다 켜짐**

**flag = 0; //모든 동작 멈춤**

**}**

**else**

**{**

**count2=8; //시간 지났을 때랑 같은 효과 내기위해**

**}**

**}**

**}**

**}**

**}**