

ЛАБОРАТОРНА РОБОТА № 8

Переривання таймера

Мета роботи: ознайомитися з принципами роботи переривань. Отримати практичні навички по організації переривань та структури програми. Використовуючи наробки, реалізувати годинник та таймер.

Виконання роботи:

Завдання 1: Ввести приклад, що представлено у лістингу 8.1. розібратися з роботою програми.

Посилання на проєкт: <https://www.tinkercad.com/things/cBLA0giYRGT-lab-8-task-1>

Код програми:

```
volatile short c_250ms = 0;
volatile bool flag_250ms = false;
volatile short c_400ms = 0;
volatile bool flag_400ms = false;
short led1 = 10;
short led2 = 11;
short led3 = 12;
short led4 = 13;
ISR(TIMER1_COMPA_vect) // обробочик прерывания. Вызывается каждую
миллисекунду
{
    c_250ms++;
    if (c_250ms == 250)
    { // Выполняем каждые 250 мс
        flag_250ms = true;
        c_250ms = 0;
    }
    c_400ms++;
    if (c_400ms == 400)
    { // Выполняем каждые 400 мс
        flag_400ms = true;
        c_400ms = 0;
    }
}
void initTimer()
{ // инициализация Timer1
```

	cli(); // отключить глобальные прерывания								
	TCNT1 = 0;								ДУ «Житомирська політехніка».22.121.21.000 – Лр1
Змн.	Арк.	№ докум.	Підпис	Дата					
	TCNT1A = 0;								
Розроб.	Петровський Н.В.								
Перевір.	Петросян Р.В.								
Керівник									
Н. контр.									
Зав. каф.									
Звіт з лабораторної роботи						Літ.	Арк.	Аркуші	
							1	22	
						ФІКТ Гр. ІПЗ-20-4			

```

TCCR1B = 0;
TCCR1B |= (1 << WGM12); // включить CTC режим
TCCR1B |= (0b001 << CS10); // Установить биты на коэффициент деления

OCR1A = 15999; // установка регистра совпадения
TIMSK1 |= (1 << OCIE1A); // включить прерывание по совпадению таймера

sei(); // включить глобальные прерывания
}
void setup()
{
// put your setup code here, to run once:
pinMode(led1, OUTPUT);
pinMode(led2, OUTPUT);
pinMode(led3, OUTPUT);
pinMode(led4, OUTPUT);
digitalWrite(led1, HIGH);
digitalWrite(led2, HIGH);
digitalWrite(led3, HIGH);
digitalWrite(led4, HIGH);

initTimer();
}
void loop()
{
if (flag_250ms)
{ // Выполняем каждые 250 мс
flag_250ms = false;

digitalWrite(led4, LOW);
}
if (flag_400ms)
{ // Выполняем каждые 400 мс
flag_400ms = false;

digitalWrite(led4, HIGH);
}
}

```

		Петровський Н.І.			ДУ «Житомирська політехніка».22.121.21.000 – Лр1	Арк.
		Петросян Р.В.				
Змн.	Арк.	№ докум.	Підпис	Дата		2

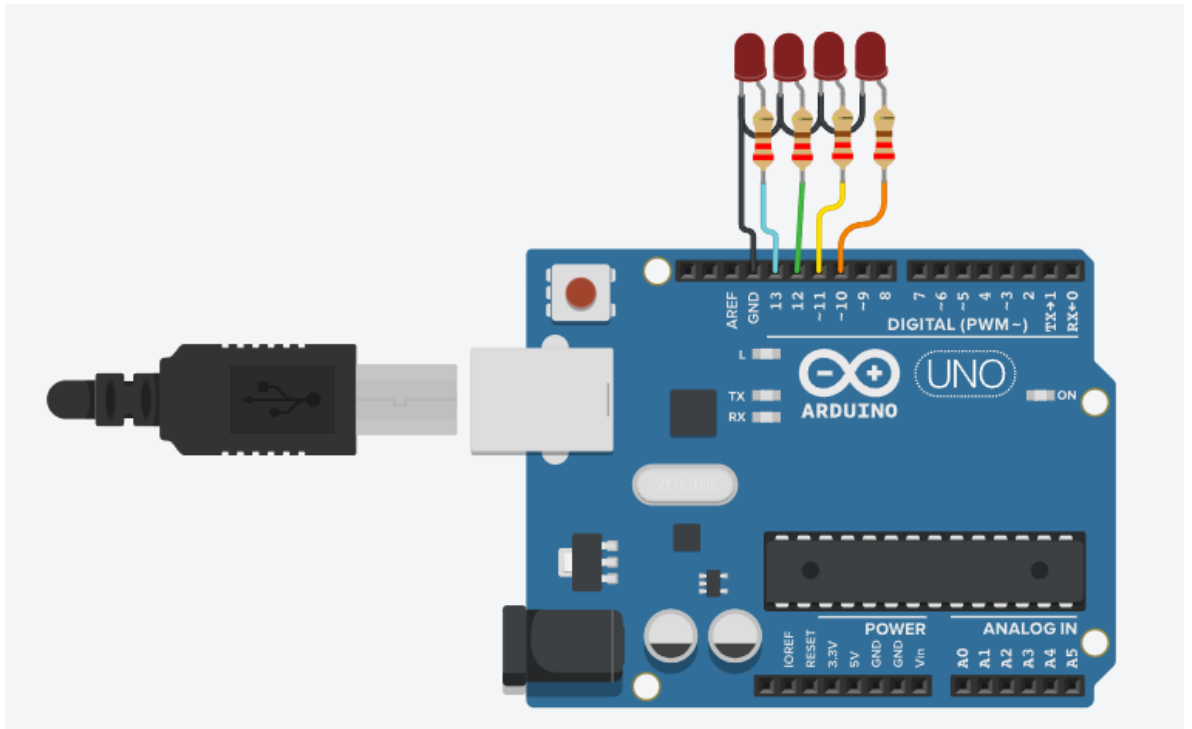


Рис. 1.1 Схема

Завдання 2: Реалізувати додаток з наступними функціями у відповідності з варіантом:

Варіант	Завдання
1,4,7,10,13 +15	<p>Реалізувати годинник з наступними вимогами:</p> <ul style="list-style-type: none"> – відображати на семисегментному індикаторі години та хвилини; – формувати сигнал оповіщення кожну годину; – керування годинником реалізувати за допомогою 3х кнопок: 1 – редагування годин у режимі редагування; 2 – редагування хвилин у режимі редагування; 3 – зміна режиму (хід годинників/редагування значень годинників).

Посилання на проєкт: <https://www.tinkercad.com/things/97iaKJr4kU1-lab-8-task-2>

Код програми:

```
#define LATCH_PIN A4 // Пін для ST_CP (latch) зсувного регістру
#define CLOCK_PIN A5 // Пін для SH_CP (clock) зсувного регістру
#define DATA_PIN A3 // Пін для DS (data) зсувного регістру
#define BUTT_MODE 2 // Пін зміни режиму / збереження
#define BUTT_E_MIN 3 // Пін зміни хвилин
#define BUTT_E_HR 4 // Пін зміни годин
#define PZ_PIN 8 // Пін пезо
```

```
const unsigned char segdisp[12] = {
    0b00111111,
    0b00000110,
```

```

0b01011011,
0b01001111,
0b01100110,
0b01101101,
0b01111101,
0b00100111,
0b01111111,
0b01101111,
0b01000000,
0b00000000,
};

int NumDigitsInNumber(int number)
{
    int digitCount = 0;
    while (number > 0)
    {
        number /= 10;
        digitCount++;
    }
    return digitCount;
}

void ReverseArray(int arr[], int size)
{
    int start = 0;
    int end = size - 1;

    while (start < end)
    {
        // Обмін значень між початковим та кінцевим індексами
        int temp = arr[start];
        arr[start] = arr[end];
        arr[end] = temp;

        // Пересуваємо початковий та кінцевий індекси
        start++;
        end--;
    }
}

void SplitInteger(int number, int *digits, int &digitsSize)
{

```

		Петровський Н.І.			ДУ «Житомирська політехніка».22.121.21.000 – Лр1	Арк.
		Петросян Р.В.				
Змн.	Арк.	№ докум.	Підпис	Дата		4

```

// Визначаємо кількість цифр у числі
int digitCount = NumDigitsInNumber(number);

// Якщо число дорівнює 0
if (number == 0)
{
    digits[0] = 0;
    digitsSize = 1;
    return;
}

// Ініціалізуємо масив та його розмір
int *tempArray = new int[digitCount];
digitsSize = digitCount;

// Розбиваємо число на цифри та зберігаємо у масиві
for (int i = digitCount - 1; i >= 0; i--)
{
    tempArray[i] = number % 10;
    number /= 10;
}

// Копіюємо дані у вихідний масив
for (int i = 0; i < digitCount; i++)
{
    digits[i] = tempArray[i];
}

// Звільняємо виділену пам'ять
delete[] tempArray;
}

void LcdOutSegment(int number, int sizeLCD)
{
    // Перевірка на від'ємність числа
    bool isNegative = (number < 0);

    // Перетворення в додатне число
    number = abs(number);

    // Отримуємо масив цифр розбитого цілого числа та його розмір
    int digitsCount = NumDigitsInNumber(number);

    int digitsSize;

```

		Петровський Н.І.			ДУ «Житомирська політехніка».22.121.21.000 – Лр1	Арк.
		Петросян Р.В.				5
Змн.	Арк.	№ докум.	Підпис	Дата		

```

int digits[digitsCount];

SplitInteger(number, digits, digitsSize);
ReverseArray(digits, digitsSize);

// Формуємо масив цифр та символів для дисплею
int segmentDigits[sizeLCD];

for (int i = 0; i < sizeLCD; i++)
{
    if (i > digitsSize - 1)
    {
        if (isNegative)
        {
            segmentDigits[i] = segdisp[10];
            isNegative = false;
        }
        else
        {
            segmentDigits[i] = segdisp[0];
        }
    }
    else
    {
        segmentDigits[i] = segdisp[digits[i]];
    }
}

// Відображаємо на дисплеї
digitalWrite(LATCH_PIN, LOW);
for (int i = 0; i < sizeLCD; i++)
{
    shiftOut(DATA_PIN, CLOCK_PIN, MSBFIRST, segmentDigits[i]);
}

digitalWrite(LATCH_PIN, HIGH);
}

bool editHours = false;
bool editMinutes = false;

bool modeButtonState = false;
bool hoursButtonState = false;
bool minutesButtonState = false;

```

		Петровський Н.І.			ДУ «Житомирська політехніка».22.121.21.000 – Лр1	Арк.
		Петросян Р.В.				6
Змн.	Арк.	№ докум.	Підпис	Дата		

```

bool modeButtonPressed = false;
bool hoursButtonPressed = false;
bool minutesButtonPressed = false;

int hours = 13;
int minutes = 59;

volatile bool modeEnabled = false;
// секунда
volatile short c_250ms = 0;
volatile bool flag_250ms = false;

// хвилина
volatile unsigned int c_60000ms = 0;
volatile bool flag_60000ms = false;

// година
volatile long c_3600000ms = minutes * 60000;
volatile bool flag_3600000ms = false;

// обробник переривання, викликається кожну мілісекунду
ISR(TIMER1_COMPA_vect)
{
    // виконуємо кожних 250 мс для обробки кнопок
    c_250ms++;
    if (c_250ms == 250)
    {
        flag_250ms = true;
        c_250ms = 0;
    }
    // якщо невимкнене редагування
    if (!modeEnabled)
    {
        c_60000ms++;
        // виконуємо кожну хвилину
        if (c_60000ms == 60000)
        {
            flag_60000ms = true;
            c_60000ms = 0;
        }
        c_3600000ms++;
        // виконуємо кожну годину
        if (c_3600000ms == 3600000)

```

		Петровський Н.І.			ДУ «Житомирська політехніка».22.121.21.000 – Лр1	Арк.
		Петросян Р.В.				7
Змн.	Арк.	№ докум.	Підпис	Дата		

```

    {
        flag_3600000ms = true;
        c_3600000ms = 0;
    }
}
else if (modeEnabled && c_60000ms != 0)
{
    // хвилина та година анулюються в режимі редагування
    flag_60000ms = false;
    c_60000ms = 0;
    flag_3600000ms = false;
    c_3600000ms = minutes * 60000;
}
}

void initTimer()
{
    cli(); // вимкнути глобальні переривання
    TCNT1 = 0;
    TCCR1A = 0; // встановити регістр керування А в 0
    TCCR1B = 0;
    TCCR1B |= (1 << WGM12); // ввімкнути СТС режим
    TCCR1B |= (0b001 << CS10); // Встановити біти на коефіцієнт ділення

    OCR1A = 15999; // встановлення регістра співпадіння
    TIMSK1 |= (1 << OCIE1A); // ввімкнення переривання по співпадінню
таймера

    sei(); // ввімкнення глобальних переривання
}

void setHours(int h)
{
    if (h < 0)
    {
        hours = 23;
    }
    else if (h > 23)
    {
        hours = 0;
    }
    else
    {
        hours = h;
    }
}

```

		Петровський Н.І.			ДУ «Житомирська політехніка».22.121.21.000 – Лр1	Арк.
		Петросян Р.В.				8
Змн.	Арк.	№ докум.	Підпис	Дата		


```

    }
}

void setMinutes(int m)
{
    if (m < 0)
    {
        minutes = 59;
    }
    else if (m > 59)
    {
        minutes = 0;
    }
    else
    {
        minutes = m;
    }
}

void displayMode()
{
    if (modeEnabled)
    {
        Serial.println("Edit enabled!");
    }
    else
    {
        Serial.println("Clock movement!");
    }
}

void hourSignal()
{
    Serial.print("Time: ");
    Serial.print(hours);
    Serial.println(":00");
    tone(PZ_PIN, 800, 500);
}

void time()
{
    // оновлення годин та хвилин
    if (flag_3600000ms)
    {

```

		Петровський Н.І.			ДУ «Житомирська політехніка».22.121.21.000 – Лр1	Арк.
		Петросян Р.В.				9
Змн.	Арк.	№ докум.	Підпис	Дата		

```

flag_3600000ms = false;
setHours(hours + 1);
hourSignal();
displayClock();
}
if (flag_60000ms)
{
    flag_60000ms = false;
    setMinutes(minutes + 1);
    displayClock();
}
if (flag_250ms)
{
    flag_250ms = false;
    // Зчитування значень з кнопок з інвертуванням
    modeButtonState = !digitalRead(BUTT_MODE);

    // Обробка кнопки режиму редагування / ходу годинника
    if (modeButtonState && !modeButtonPressed)
    {
        modeButtonPressed = true;
        if (editHours)
        {
            editHours = false;
        }
        else if (editMinutes)
        {
            editMinutes = false;
        }
        else
        {
            modeEnabled = !modeEnabled;
            displayMode();
        }
    }
    else if (!modeButtonState && modeButtonPressed)
    {
        modeButtonPressed = false;
    }
    if (modeEnabled)
    {
        hoursButtonState = !digitalRead(BUTT_E_HR);
        minutesButtonState = !digitalRead(BUTT_E_MIN);
        // обробка зміни хвилин та годин

```

		Петровський Н.І.			ДУ «Житомирська політехніка».22.121.21.000 – Лр1	Арк.
		Петросян Р.В.				10
Змн.	Арк.	№ докум.	Підпис	Дата		

```

        if (hoursButtonState && !hoursButtonPressed && !editHours &&
!editMinutes)

```

```

    {
        hoursButtonPressed = true;
        editHours = true;
    }

```

```

    else if (!hoursButtonState && hoursButtonPressed)

```

```

    {
        hoursButtonPressed = false;
    }

```

```

        if (minutesButtonState && !minutesButtonPressed && !editMinutes &&
!editHours)

```

```

    {
        minutesButtonPressed = true;
        editMinutes = true;
    }

```

```

    else if (!minutesButtonState && minutesButtonPressed)

```

```

    {
        minutesButtonPressed = false;
    }

```

```

// Обробка кнопок редагування на збільшення та зменшення

```

```

if (hoursButtonState && !hoursButtonPressed)

```

```

{
    hoursButtonPressed = true;
    if (editHours)
    {
        setHours(hours - 1);
    }
    else if (editMinutes)
    {
        setMinutes(minutes - 1);
        c_3600000ms = minutes * 60000;
    }
    displayClock();
}

```

```

else if (!hoursButtonState && hoursButtonPressed)

```

```

{
    hoursButtonPressed = false;
}

```

```

if (minutesButtonState && !minutesButtonPressed)

```

```

{

```

		Петровський Н.І.			ДУ «Житомирська політехніка».22.121.21.000 – Лр1	Арк.
		Петросян Р.В.				11
Змн.	Арк.	№ докум.	Підпис	Дата		

```

minutesButtonPressed = true;
if (editHours)
{
    setHours(hours + 1);
}
else if (editMinutes)
{
    setMinutes(minutes + 1);
    c_3600000ms = minutes * 60000;
}
displayClock();
}
else if (!minutesButtonState && minutesButtonPressed)
{
    minutesButtonPressed = false;
}
}
}
}

```

// відображення на семисегментних індикаторах

```
void displayClock()
```

```

{
    LcdOutSegment(hours * 100 + minutes, 4);
}

```

```
void setup()
```

```

{
    Serial.begin(9600);
    // семисегментні індикатори
    pinMode(LATCH_PIN, OUTPUT);
    pinMode(CLOCK_PIN, OUTPUT);
    pinMode(DATA_PIN, OUTPUT);

    // кнопки
    pinMode(BUTT_MODE, INPUT_PULLUP);
    pinMode(BUTT_E_HR, INPUT_PULLUP);
    pinMode(BUTT_E_MIN, INPUT_PULLUP);

```

```
// п'єзо
```

```
pinMode(PZ_PIN, OUTPUT);
```

```
// відображення початкового часу
```

```
displayClock();
```

		Петровський Н.І.			ДУ «Житомирська політехніка».22.121.21.000 – Лр1	Арк.
		Петросян Р.В.				12
Змн.	Арк.	№ докум.	Підпис	Дата		

```

// запуск таймеру (переривань)
initTimer();
}

void loop()
{
  time();
}

```

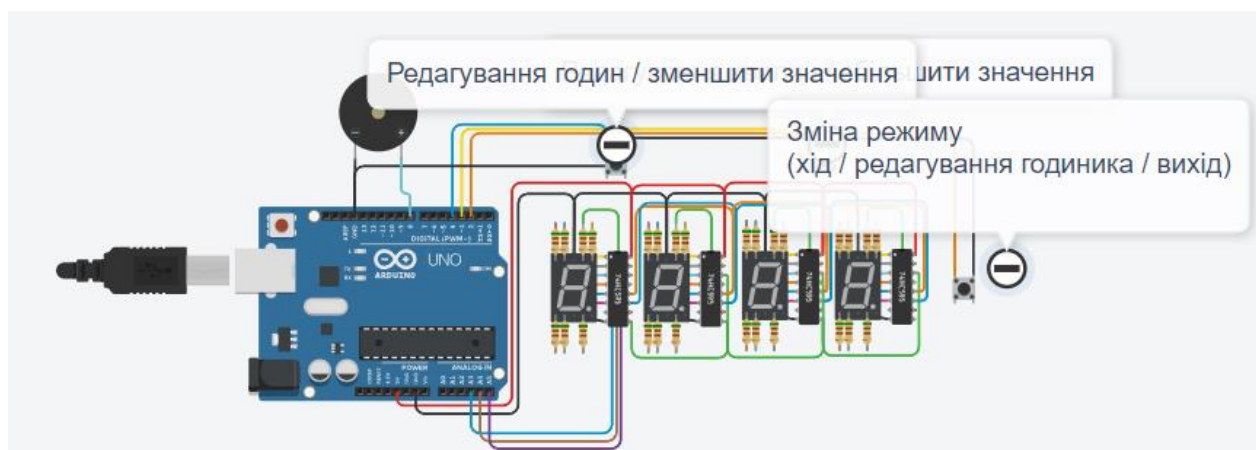


Рис. 2.1 Схема

Висновок: В результаті виконання лабораторної роботи було ознайомлено з принципами роботи переривань, отримано практичні навички по організації переривань та структури програми, використано набірки, реалізовано годинник та таймер.

		Петровський Н.І.			ДУ «Житомирська політехніка».22.121.21.000 – Лр1	Арк.
		Петросян Р.В.				13
Змн.	Арк.	№ докум.	Підпис	Дата		