Міністерство освіти і науки України

**Прикарпатський національний університет**

**імені В.Стефаника**

Факультет математики та інформатики

Кафедра інформаційних технологій

Програмування вбудованих систем

Лабораторна робота № 8

Тема: «**Робота з IR-пультами та приймачами**»

Варіант : **21**

Виконав: Рижкін О. К.

Група ІПЗ-23

Дата: 7 грудня 2023р.

Викладач: Лазарович І.М.

Івано-Франківськ – 2023р.

**Мета:** навчитись опрацьовувати імпульсні сигнали на низькому рівні без

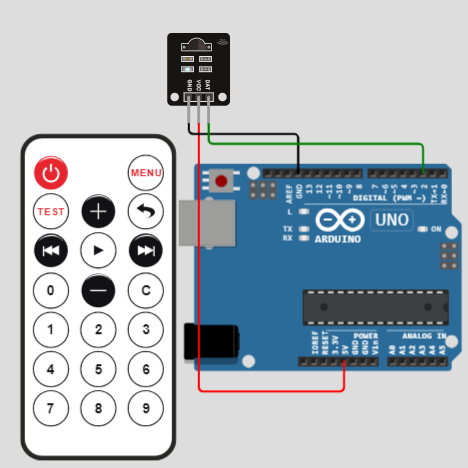
спеціалізованих бібліотек, розпізнавати команди з IR-пульта.

**Завдання**

Зібрати схему згідно рисунка 8.9, використовуючи симулятор Wokwi.

Розробити програму, яка «впізнає» тільки ті кнопки, які потрібно, щоб задати місяць і дату народження студента-виконавця роботи, натиснення інших кнопок ігнорувати. При натисненні потрібної кнопки виводити в монітор порту відповідну цифру і двійковий код кодового пакету. При написанні програми НЕ дозволяється застосовувати бібліотеки для роботи з IR-пультами/приймачами. Обробку сигналу від приймача виконати з використанням зовнішнього переривання.

**Схема**

****

**Посилання на проект: https://wokwi.com/projects/383579853152390145**

**Програмний код**

boolean nec\_ok = 0;

byte  i, nec\_state = 0, command, inv\_command;

unsigned int address;

unsigned long nec\_code;

byte codes[4] = {22, 12, 28, 90};

byte values[4] = {0, 1, 5, 6};

char buffer[33];

void setup() {

**Serial**.begin(9600);

  TCCR1A = 0;

  TCCR1B = 0;

  TCNT1  = 0;

  TIMSK1 = 1;

  attachInterrupt(0, remote\_read, CHANGE);

}

void remote\_read() {

  unsigned int timer\_value;

  if (nec\_state != 0) {

    timer\_value = TCNT1;

    TCNT1 = 0;

  }

  switch (nec\_state) {

    case 0 :

      TCNT1  = 0;

      TCCR1B = 2;

      nec\_state = 1;

      i = 0;

      return;

    case 1 :

      if ((timer\_value > 19000) || (timer\_value < 17000)) {

        nec\_state = 0;

        TCCR1B = 0;

      }

      else

        nec\_state = 2;

      return;

    case 2 :

      if ((timer\_value > 10000) || (timer\_value < 8000)) {

        nec\_state = 0;

        TCCR1B = 0;

      }

      else

        nec\_state = 3;

      return;

    case 3 :

      if ((timer\_value > 1400) || (timer\_value < 800)) {

        TCCR1B = 0;

        nec\_state = 0;

      }

      else

        nec\_state = 4;

      return;

    case 4 :

      if ((timer\_value > 3600) || (timer\_value < 800)) {

        TCCR1B = 0;

        nec\_state = 0;

        return;

      }

      if ( timer\_value > 2000)

        bitSet(nec\_code, (31 - i));

      else

        bitClear(nec\_code, (31 - i));

      i++;

      if (i > 31) {

        nec\_ok = 1;

        detachInterrupt(0);

        return;

      }

      nec\_state = 3;

  }

}

ISR(TIMER1\_OVF\_vect) {

  nec\_state = 0;

  TCCR1B = 0;

}

void loop() {

if (nec\_ok) {

    nec\_ok = 0;

    nec\_state = 0;

    TCCR1B = 0;

    command = nec\_code >> 8;

    for (int i = 0; i < 4; i++){

        if (codes[i] == command ) {

**Serial**.println(values[i]);

**Serial**.println( ltoa(nec\_code, buffer, 2));

**Serial**.println();

      }

    }

    attachInterrupt(0, remote\_read, CHANGE);

  }

}