**Практическое задание №5**

по дисциплине:

«Основы технологии интернет вещей»

Выполнил студент  
Группы ПИ20-1В  
Зайцев Н.

Задача:

1. Напишите программу для микроконтроллера Arduino, реализующую имитацию двух светофоров с различными фазами работы и различной продолжительностью сигналов без использования функции delay().
2. Зафиксируйте размер скетча в байтах.
3. Визуализацию процессов реализуйте на эмуляторе Proteus.

Примечания:

Используйте в программе функцию millis() и библиотеку Thread.h.

1.[Проект](https://wokwi.com/projects/365099900329714689)

2.Исходный код

#include <Thread.h>

byte green\_pin\_1 = 5;

byte yellow\_pin\_1 = 6;

byte red\_pin\_1 = 7;

byte green\_pin\_2 = 8;

byte yellow\_pin\_2 = 9;

byte red\_pin\_2 = 10;

boolean led\_on = true;

boolean yellow\_on = LOW;

unsigned long loop\_time;

unsigned long current\_time;

Thread thread1 = Thread();

Thread thread2 = Thread();

void setup() {

pinMode(green\_pin\_1, OUTPUT);

pinMode(yellow\_pin\_1, OUTPUT);

pinMode(red\_pin\_1, OUTPUT);

pinMode(green\_pin\_2, OUTPUT);

pinMode(yellow\_pin\_2, OUTPUT);

pinMode(red\_pin\_2, OUTPUT);

thread1.onRun(traffic\_light\_1);

thread2.onRun(traffic\_light\_2);

}

void traffic\_light\_1(){

current\_time = millis();

if(led\_on == false){

if(current\_time >= (loop\_time) && current\_time < (loop\_time + 500)) {

digitalWrite(yellow\_pin\_1, HIGH);

}

if(current\_time >= (loop\_time + 500) && current\_time < (loop\_time + 1000)) {

digitalWrite(yellow\_pin\_1, LOW);

}

if(current\_time >= (loop\_time + 1000)) {

loop\_time = current\_time;

}

}

if(led\_on == true) {

if(current\_time >= loop\_time && current\_time < (loop\_time + 10000)) {

digitalWrite(green\_pin\_1, HIGH);

digitalWrite(yellow\_pin\_1, LOW);

digitalWrite(red\_pin\_1, LOW);

}

if(current\_time >= (loop\_time + 10000) && current\_time < (loop\_time + 10500)) {

digitalWrite(green\_pin\_1, LOW);

}

if(current\_time >= (loop\_time + 10500) && current\_time < (loop\_time + 11000)) {

digitalWrite(green\_pin\_1, HIGH);

}

if(current\_time >= (loop\_time + 11000) && current\_time < (loop\_time + 11500)) {

digitalWrite(green\_pin\_1, LOW);

}

if(current\_time >= (loop\_time + 11500) && current\_time < (loop\_time + 12000)) {

digitalWrite(green\_pin\_1, HIGH);

}

if(current\_time >= (loop\_time + 12000) && current\_time < (loop\_time + 12500)) {

digitalWrite(green\_pin\_1, LOW);

}

if(current\_time >= (loop\_time + 12500) && current\_time < (loop\_time + 13000)) {

digitalWrite(green\_pin\_1, HIGH);

}

if(current\_time >= (loop\_time + 13000) && current\_time < (loop\_time + 13500)) {

digitalWrite(green\_pin\_1, LOW);

}

if(current\_time >= (loop\_time + 13500) && current\_time < (loop\_time + 1400)) {

digitalWrite(green\_pin\_1, HIGH);

}

if(current\_time >= (loop\_time + 14000) && current\_time < (loop\_time + 14300)) {

digitalWrite(green\_pin\_1, LOW);

}

if(current\_time >= (loop\_time + 14300) && current\_time < (loop\_time + 17500)) {

digitalWrite(yellow\_pin\_1, HIGH);

}

if(current\_time >= (loop\_time + 17500) && current\_time < (loop\_time + 27500)) {

digitalWrite(yellow\_pin\_1, LOW);

digitalWrite(red\_pin\_1, HIGH);

}

if(current\_time >= (loop\_time + 44200) && current\_time < (loop\_time + 47200)) {

digitalWrite(yellow\_pin\_1, HIGH);

}

if(current\_time >= (loop\_time + 47200)) {

digitalWrite(green\_pin\_1, HIGH);

digitalWrite(yellow\_pin\_1, LOW);

digitalWrite(red\_pin\_1, LOW);

loop\_time = current\_time;

}

}

}

void traffic\_light\_2(){

current\_time = millis();

if(led\_on == false){

if(current\_time >= (loop\_time) && current\_time < (loop\_time + 500)) {

digitalWrite(yellow\_pin\_2, HIGH);

}

if(current\_time >= (loop\_time + 500) && current\_time < (loop\_time + 1000)) {

digitalWrite(yellow\_pin\_2, LOW);

}

if(current\_time >= (loop\_time + 1000)) {

loop\_time = current\_time;

}

}

if(led\_on == true) {

if(current\_time >= loop\_time && current\_time < (loop\_time + 10000)) {

digitalWrite(green\_pin\_2, LOW);

digitalWrite(yellow\_pin\_2, LOW);

digitalWrite(red\_pin\_2, HIGH);

}

if(current\_time >= (loop\_time + 20500) && current\_time < (loop\_time + 23500)) {

digitalWrite(yellow\_pin\_2, HIGH);

}

if(current\_time >= (loop\_time + 23500) && current\_time < (loop\_time + 33500)) {

digitalWrite(green\_pin\_2, HIGH);

digitalWrite(yellow\_pin\_2, LOW);

digitalWrite(red\_pin\_2, LOW);

}

if(current\_time >= (loop\_time + 33500) && current\_time < (loop\_time + 34000)) {

digitalWrite(green\_pin\_2, LOW);

}

if(current\_time >= (loop\_time + 34000) && current\_time < (loop\_time + 34500)) {

digitalWrite(green\_pin\_2, HIGH);

}

if(current\_time >= (loop\_time + 34500) && current\_time < (loop\_time + 35000)) {

digitalWrite(green\_pin\_2, LOW);

}

if(current\_time >= (loop\_time + 35000) && current\_time < (loop\_time + 35500)) {

digitalWrite(green\_pin\_2, HIGH);

}

if(current\_time >= (loop\_time + 35500) && current\_time < (loop\_time + 36000)) {

digitalWrite(green\_pin\_2, LOW);

}

if(current\_time >= (loop\_time + 36000) && current\_time < (loop\_time + 36500)) {

digitalWrite(green\_pin\_2, HIGH);

}

if(current\_time >= (loop\_time + 36500) && current\_time < (loop\_time + 37000)) {

digitalWrite(green\_pin\_2, LOW);

}

if(current\_time >= (loop\_time + 37000) && current\_time < (loop\_time + 37500)) {

digitalWrite(green\_pin\_2, HIGH);

}

if(current\_time >= (loop\_time + 37500) && current\_time < (loop\_time + 38000)) {

digitalWrite(green\_pin\_2, LOW);

}

if(current\_time >= (loop\_time + 38000) && current\_time < (loop\_time + 41200)) {

digitalWrite(yellow\_pin\_2, HIGH);

}

if(current\_time >= (loop\_time + 41200) && current\_time < (loop\_time + 47200)) {

digitalWrite(yellow\_pin\_2, LOW);

digitalWrite(red\_pin\_2, HIGH);

}

if(current\_time >= (loop\_time + 47200)) {

loop\_time = current\_time;

}

}

}

void loop() {

thread1.run();

thread2.run();

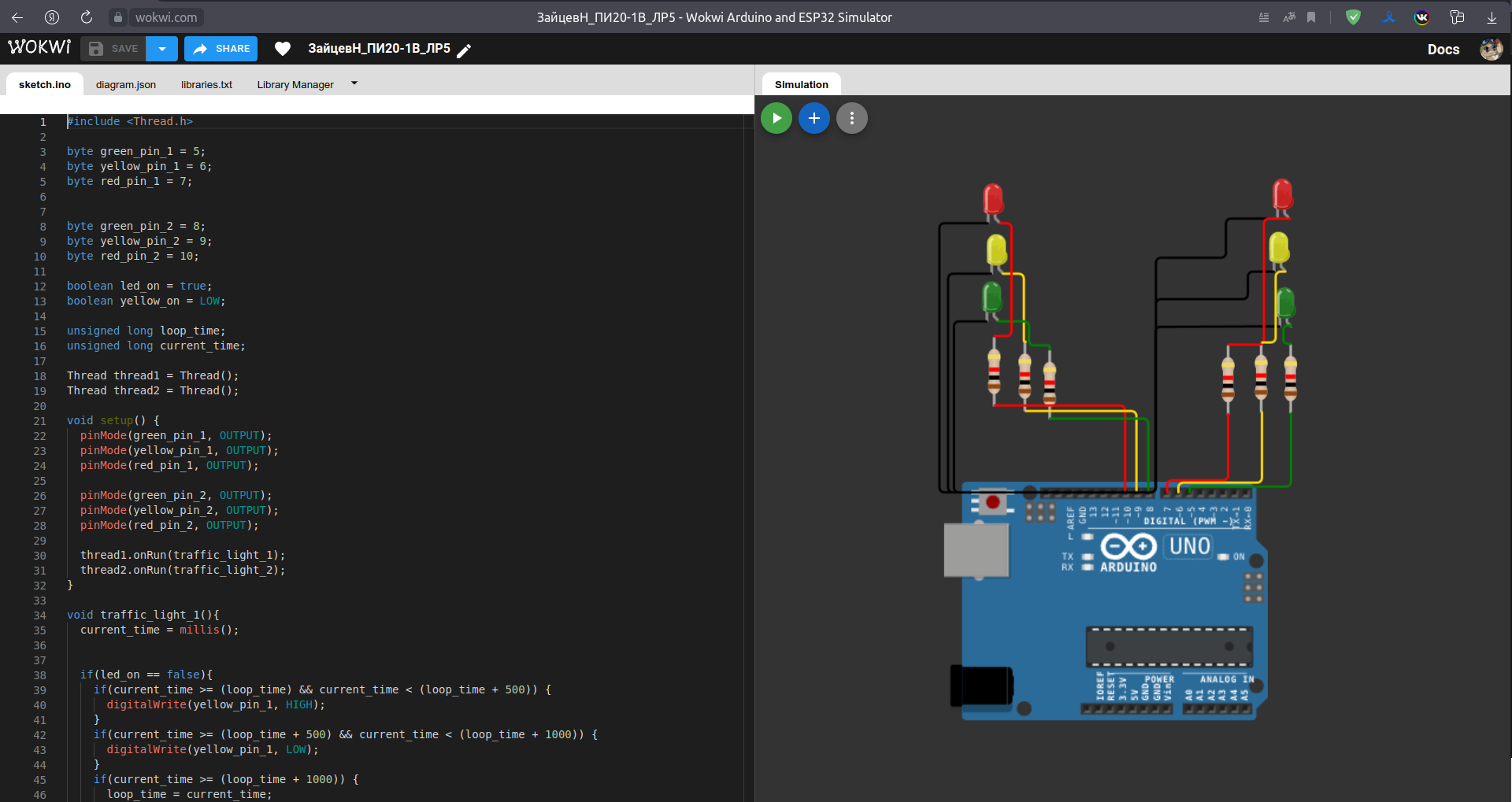
}

Объем скетч-файла в кБ

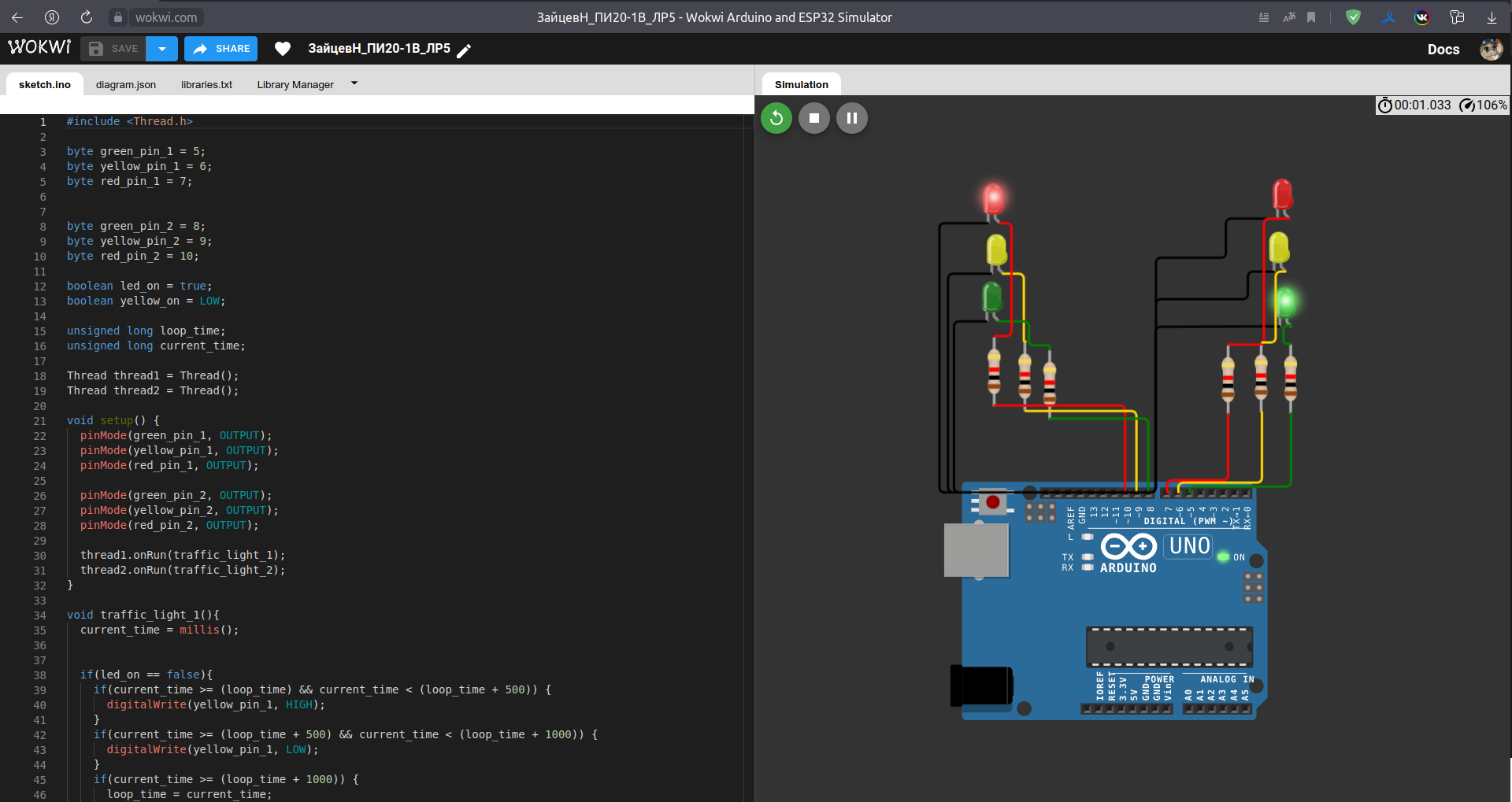
Скетч использует 4312 байт (13%) памяти устройства. Всего доступно 32256 байт.

Глобальные переменные используют 127 байт (6%) динамической памяти, оставляя 1921 байт для локальных переменных. Максимум: 2048 байт.

Исходное состояние



Активное состояние



Выводы: мы изучили основные принципы работы с библиотекой ArduinoThreads.