|  |  |
| --- | --- |
| Gerb-BMSTU_01 | **Министерство науки и высшего образования Российской Федерации**  **Федеральное государственное автономное образовательное учреждение**  **высшего образования**  **«Московский государственный технический университет**  **имени Н.Э. Баумана**  **(национальный исследовательский университет)»**  **(МГТУ им. Н.Э. Баумана)** |

ФАКУЛЬТЕТ **ИНФОРМАТИКА И СИСТЕМЫ УПРАВЛЕНИЯ**

КАФЕДРА **КОМПЬЮТЕРНЫЕ СИСТЕМЫ И СЕТИ (ИУ6)**

НАПРАВЛЕНИЕ ПОДГОТОВКИ **09.03.01 Информатика и вычислительная техника**

**ОТЧЕТ**

**по лабораторной работе № \_6\_**

**Название:** Программирование микроконтроллеров STM32

**Дисциплина:** Микропроцессорные системы

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Студент | ИУ6-63Б |  |  | В.К. Залыгин |
|  | (Группа) |  | (Подпись, дата) | (И.О. Фамилия) |
|  |  |  |  |  |
| Преподаватель |  |  |  | Е.Ю. Гаврилова |
|  |  |  | (Подпись, дата) | (И.О. Фамилия) |

Москва, 2025

**Цель работы**:

* знакомство с микроконтроллером семейства STM32 и приемами его программирования;
* получение навыков отладки программ в среде STM32CubeIDE;
* получение опыта прототипирования устройств с использованием макетной платы.

**Вариант 16**

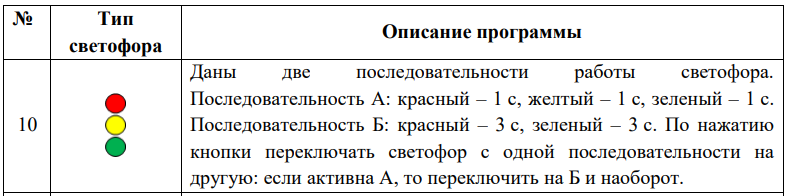


Рисунок 1 – Индивидуальное задание по варианту

Граф состояний перехода светофора по индивидуальному варианту

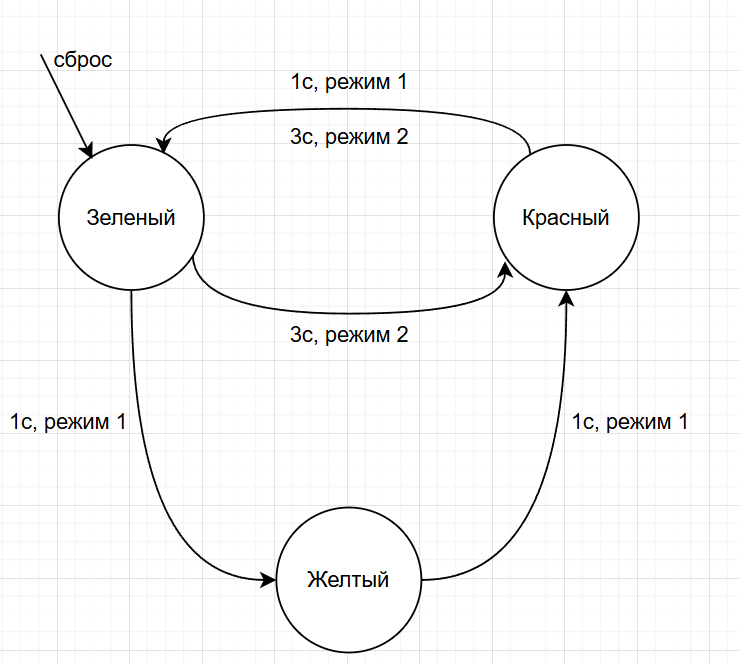


Рисунок 2 – Граф состояний перехода светофора

Добавим новую структуру данных TrafficLightStep, которая будет в себе содержать состояние светофора и некоторые переменные.

Листинг 1 **-** Новая структура

|  |
| --- |
| typedef enum  {    TL\_GREEN,    TL\_YELLOW,    TL\_RED  } TrafficLightState; //list of possible traffic light states  #define TrafficLight\_GPIO\_Port GPIOA  #define LED\_ON 0U  #define LED\_OFF 1U  #define TrafficLight\_PlanSize1 3U // количество состояний  #define TrafficLight\_PlanSize2 2U // количество состояний  #define TrafficLight\_State\_Duration1 6U // длительность состояния  #define TrafficLight\_State\_Duration2 6U\*3 // длительность состояния |

Подкорректируем работу со светодиодами в нашем варианте, оставив только нужные состояния.

Листинг 2 — Измененная функция TrafficLight\_SetState

|  |
| --- |
| void TrafficLight\_SetState(TrafficLightState state)  {    switch (state)    {    case TL\_GREEN:      HAL\_GPIO\_WritePin(TrafficLight\_GPIO\_Port, TL\_RED\_Pin|TL\_YELLOW\_Pin, LED\_OFF);      HAL\_GPIO\_WritePin(TrafficLight\_GPIO\_Port, TL\_GREEN\_Pin, LED\_ON);    break;    case TL\_YELLOW:      HAL\_GPIO\_WritePin(TrafficLight\_GPIO\_Port, TL\_RED\_Pin|TL\_GREEN\_Pin, LED\_OFF);      HAL\_GPIO\_WritePin(TrafficLight\_GPIO\_Port, TL\_YELLOW\_Pin, LED\_ON);    break;    case TL\_RED:      HAL\_GPIO\_WritePin(TrafficLight\_GPIO\_Port, TL\_GREEN\_Pin|TL\_YELLOW\_Pin, LED\_OFF);      HAL\_GPIO\_WritePin(TrafficLight\_GPIO\_Port, TL\_RED\_Pin, LED\_ON);    break;    default:    break;    }  } |

Изменим функционал обработчика прерываний таймера под нашу структуру, сравнивая теперь количество случившихся прерываний с установленной длительностью текущего состояния.

Листинг 3 — Измененная функция Timer\_Overflowhandler

|  |
| --- |
| void Timer\_OverflowHandler(void)  {    HAL\_GPIO\_TogglePin(BOARD\_LED\_GPIO\_Port, BOARD\_LED\_Pin); //toggle LED    TimerOverflowCount++;    if (state == 0) {      if (TimerOverflowCount == TrafficLight\_State\_Duration1) {        TimerOverflowCount = 0;        TrafficLight\_PlanIndex++;        if (TrafficLight\_PlanIndex == TrafficLight\_PlanSize1) {          TrafficLight\_PlanIndex = 0;        }        TrafficLight\_SetState(TrafficLight\_Plan[TrafficLight\_PlanIndex]);      }    } else {      if (TimerOverflowCount == TrafficLight\_State\_Duration2) {        TimerOverflowCount = 0;        TrafficLight\_PlanIndex++;        if (TrafficLight\_PlanIndex == TrafficLight\_PlanSize2) {          TrafficLight\_PlanIndex = 0;        }        TrafficLight\_SetState(TrafficLight\_Plan[TrafficLight\_PlanIndex]);      }    }  } |

Также по заданию мы должны переключаться в новое состояние, не дожидаясь таймера после нажатия на кнопку. Реализуем переход в следующее состояние в обработчике нажатия этой кнопки.

Листинг 4 — Измененная функция HAL\_GPIO\_EXTI\_Callback

|  |
| --- |
| void HAL\_GPIO\_EXTI\_Callback(uint16\_t GPIO\_Pin)  {    if (GPIO\_Pin == EXT\_BUTTON\_Pin) //check if external button triggered    {      if (state == 0) {        state = 1;      } else {        state = 0;      }    }  } |

На рисунке 3 представлен макет светофора с кнопкой.

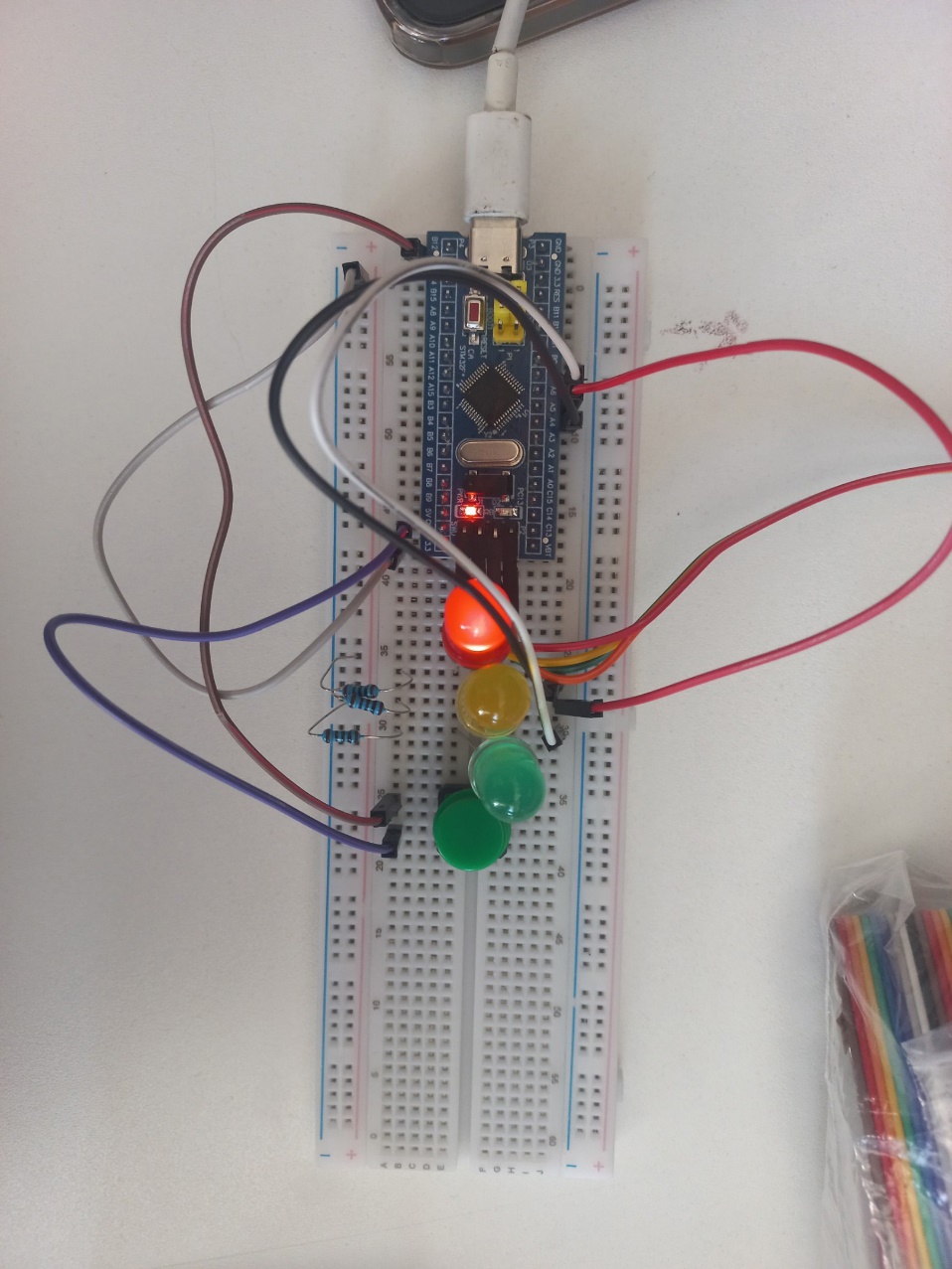


Рисунок 3 — Макет

**Вывод**

В ходе работы было проведено знакомство с микроконтроллером семейства STM32, изучены основные приёмы его программирования в среде STM32CubeIDE, а также получены навыки отладки и прототипирования устройств на макетной плате.