

УНИВЕРСИТЕТ ИТМО

Факультет программной инженерии и компьютерной техники

Направление подготовки 09.03.04 Программная инженерия

Дисциплина «Проектирование вычислительных систем»

Лабораторная работа №1

Вариант 6

Студент

Белогаев Д. В.

Кузнецов М. А.

Р34131

Преподаватель

Пинкевич В. Ю.

Санкт-Петербург, 2023 г.

Цели лабораторной работы

Получить базовые знания о принципах устройства стенда SDK-1.1M и программировании микроконтроллеров. Изучить устройство интерфейсов ввода-вывода общего назначения (GPIO) в микроконтроллерах и приемы использования данных интерфейсов.

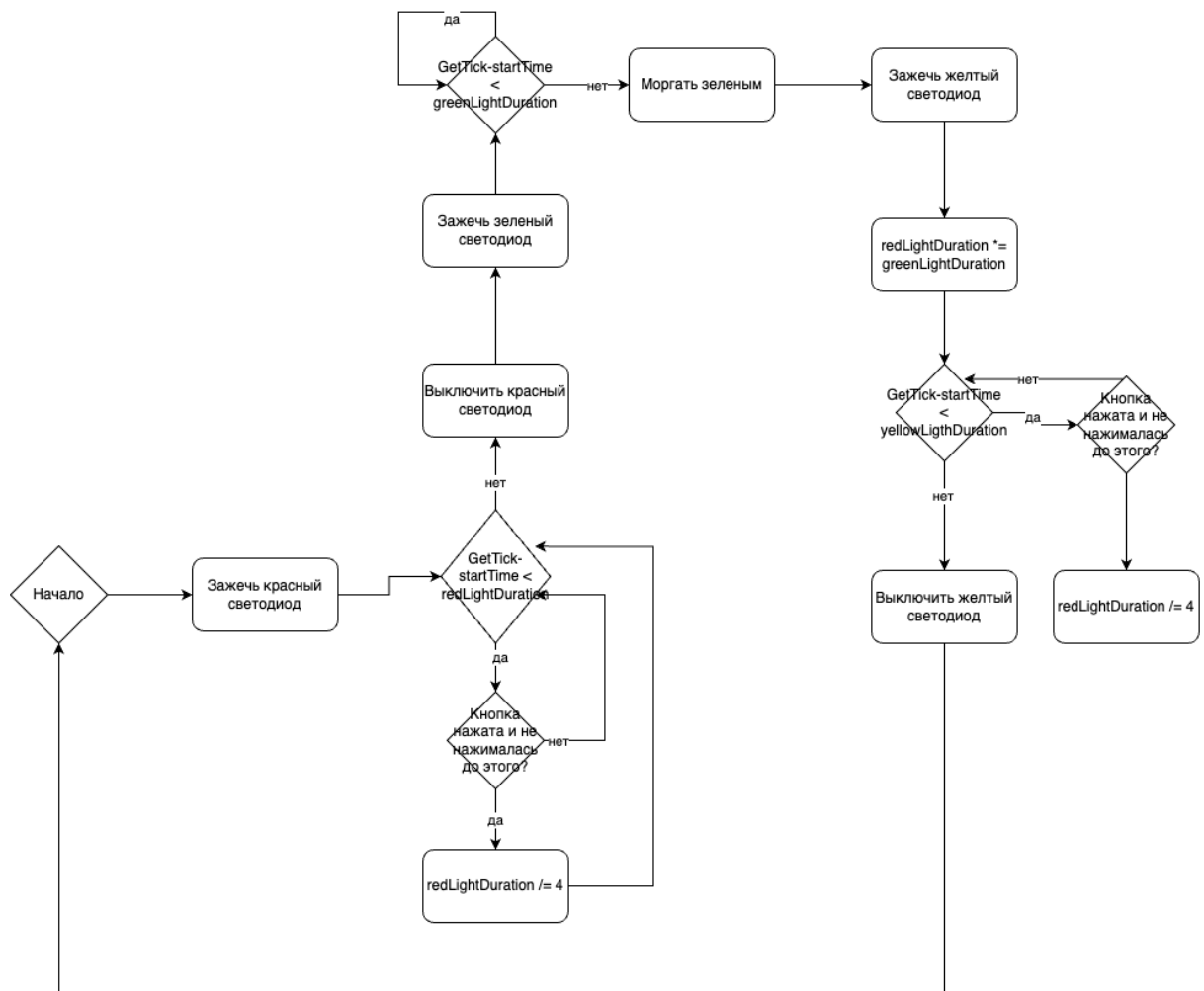
Задание лабораторной работы

Разработать и реализовать драйверы управления светодиодными индикаторами и чтения состояния кнопки стенда SDK-1.1M (расположены на боковой панели стенда). Контакты подключения кнопки и светодиодов должны быть настроены в режиме GPIO. Функции и другие компоненты драйверов должны быть универсальными, т.е. пригодными для использования в любом из вариантов задания и не должны содержать прикладной логики программы. Функции драйверов должны быть неблокирующими, то есть не должны содержать ожиданий события (например, нажатия кнопки). Также, в драйверах не должно быть пауз с активным ожиданием функция `HAL_Delay()` и собственные варианты аналогичной реализации. Обработка нажатия кнопки в программе должна включать программную защиту отдребезга. Написать программу с использованием разработанных драйверов в соответствии с вариантом задания

Вариант задания

Сымитировать работу светофора пешеходного перехода. Светофор циклически переключает цвета в следующем порядке (порядок условный, соответствие реальному светофору не соблюдается): красный, зелёный, зелёный мигающий, жёлтый, снова красный и т.д. По умолчанию период горения красного в четыре раза больше периода горения зеленого. Если во время горения зеленого мигающего, желтого или красного нажимается кнопка, светофор запоминает необходимость скорейшего переключения на зелёный. После нажатия кнопки общий цикл работы светофора не нарушается, но период горения красного должен быть сокращен до $\frac{1}{4}$ своего обычного периода. Если кнопка нажата во время горения красного,

когда он уже горит более $\frac{1}{4}$ периода, то сразу происходит переключение на зеленый.



Исходный код

Объявления необходимых переменных:

```

uint16_t GREEN_LIGHT = GPIO_PIN_13;
uint16_t YELLOW_LIGHT = GPIO_PIN_14;
uint16_t RED_LIGHT = GPIO_PIN_15;
uint16_t BUTTON = GPIO_PIN_15;

```

Функция ожидания заданного временного интервала:

```

void wait(uint32_t duration)
{
    uint32_t begin = HAL_GetTick();
    while((HAL_GetTick() - begin) < duration){}
}

```

Функция выключения определенного светодиода:

```
void turnSpecificLightOff(uint16_t light_type)
{
    HAL_GPIO_WritePin(GPIOD, light_type, GPIO_PIN_RESET);
}
```

Функция выключения всех светодиодов:

```
void shutdownAll()
{
    turnSpecificLightOff(GREEN_LIGHT);
    turnSpecificLightOff(YELLOW_LIGHT);
    turnSpecificLightOff(RED_LIGHT);
}
```

Функция включения определенного светодиода:

```
void turnSpecificLightOn(uint16_t light_type)
{
    HAL_GPIO_WritePin(GPIOD, light_type, GPIO_PIN_SET);
}
```

Функция мигания заданным светодиодом:

```
void blinkLight(uint32_t count, uint16_t light_type, uint32_t duration)
{
    for(uint32_t i = 0; i < count; i++)
    {
        wait(duration);
        turnSpecificLightOn(light_type);
        wait(duration);
        turnSpecificLightOff(light_type);
    }
}
```

Основная программа:

```
int main(void)
{
    /* USER CODE BEGIN 1 */
    /* USER CODE END 1 */
}
```

```

/* MCU Configuration-----*/
/* Reset of all peripherals, Initializes the Flash interface and the Systick. */
HAL_Init();
/* USER CODE BEGIN Init */
/* USER CODE END Init */
/* Configure the system clock */
SystemClock_Config();
/* USER CODE BEGIN SysInit */
/* USER CODE END SysInit */
/* Initialize all configured peripherals */
MX_GPIO_Init();
/* USER CODE BEGIN 2 */
uint32_t startTime = 0;
uint32_t greenLightDuration = 2000;
uint32_t blinkDuration = 500;
uint32_t redLightDuration = 4 * greenLightDuration;
uint32_t yellowLightDuration = 3000;
uint8_t buttonFlag = 0;
/* USER CODE END 2 */
/* Infinite loop */
/* USER CODE BEGIN WHILE */
while (1)
{
    startTime = HAL_GetTick();
    turnSpecificLightOn(RED_LIGHT);
    while((HAL_GetTick() - startTime) < redLightDuration)
    {
        if(HAL_GPIO_ReadPin(GPIOC, BUTTON) == 0 && buttonFlag == 0) {
            redLightDuration = redLightDuration / 4;
            buttonFlag = 1;
        }
    }
    turnSpecificLightOff(RED_LIGHT);
    turnSpecificLightOn(GREEN_LIGHT);
    wait(greenLightDuration);
    blinkLight(3, GREEN_LIGHT, blinkDuration);
    turnSpecificLightOn(YELLOW_LIGHT);
    startTime = HAL_GetTick();
    redLightDuration = 4 * greenLightDuration;
    buttonFlag = 0;
    while((HAL_GetTick() - startTime) < yellowLightDuration)
    {
        if(HAL_GPIO_ReadPin(GPIOC, BUTTON) == 0 && buttonFlag == 0) {
            redLightDuration = redLightDuration / 4;

```

```
        buttonFlag = 1;
    }
}
turnSpecificLightOff(YELLOW_LIGHT);
/* USER CODE END WHILE */
/* USER CODE BEGIN 3 */
}
/* USER CODE END 3 */
}
```

Вывод

Во время выполнения лабораторной работы мы:

- получили базовые знания об устройстве стенда SDK 1.1M
- изучили работу с интерфейсами ввода-вывода
- разработали программу для управления светодиодами с применением полученных знаний