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.*

*P34131*

Преподаватель

*Пинкевич В. Ю.*

Санкт-Петербург, 2023 г.

Цели лабораторной работы

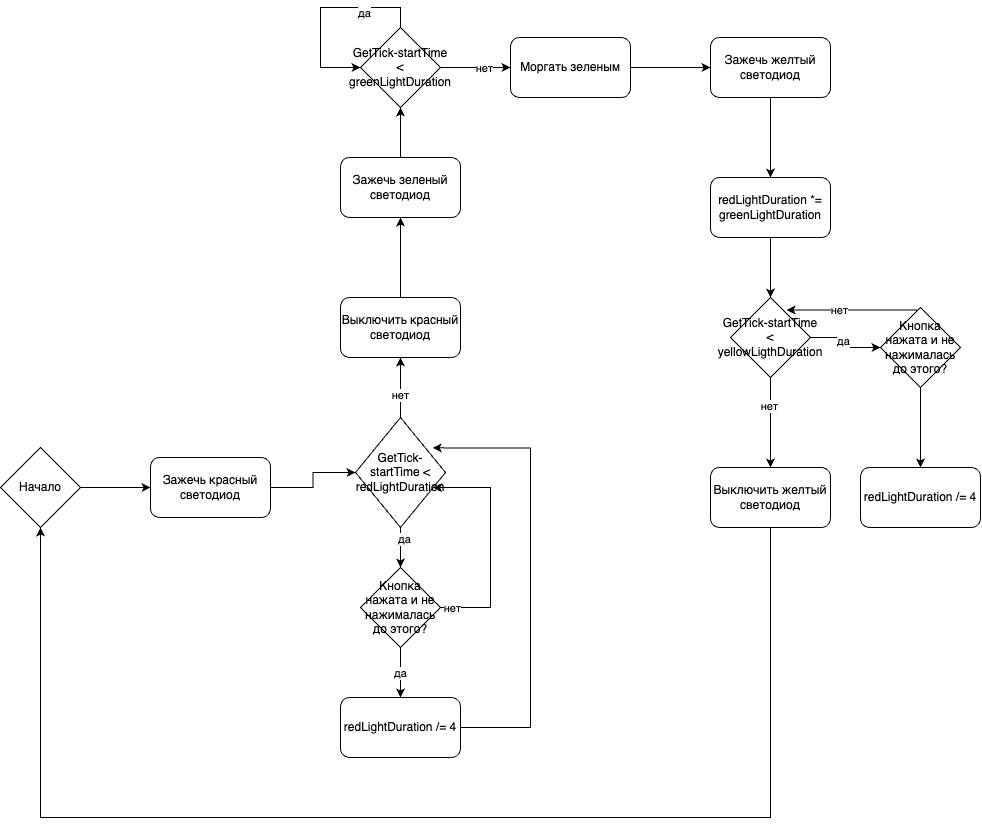
Получить базовые знания о принципах устройства стенда SDK-1.1M и программировании микроконтроллеров. Изучить устройство интерфейсов ввода-вывода общего назначения (GPIO) в микроконтроллерах и приемы использования данных интерфейсов.

Задание лабораторной работы

Разработать и реализовать драйверы управления светодиодными индикаторами и чтения состояния кнопки стенда SDK-1.1М (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 от дребезга. 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, зелѐный, зелѐный мигающий, жѐлтый, 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 зелѐный. После нажатия кнопки общий цикл работы светофора не нарушается, но период горения красного должен быть сокращен до ¼ своего обычного периода. Если кнопка нажата во время горения красного, когда он уже горит более ¼ 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 светодиодо:*

**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