УНИВЕРСИТЕТ ИТМО

Факультет программной инженерии и компьютерной техники

Направление подготовки 09.03.04 Программная инженерия

Дисциплина «Проектирование вычислительных систем»

**Лабораторная работа №1**

Вариант 11

Студент

*Митрофанов Е. Ю.*

*Любкин А. С.*

*P34101*

Преподаватель

*Пинкевич В. Ю.*

Санкт-Петербург, 2022 г.

Задание лабораторной работы

Разработать и реализовать драйверы управления светодиодными индикаторами и чтения состояния кнопки стенда SDK-1.1М (индикаторы и кнопка расположены на боковой панели стенда). 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 задержек на определенное время с использованием активного ожидания (функция HAL\_Delay() и собственные варианты аналогичной реализации), а также активного ожидания событий в циклах. 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: красный, зелёный, зелёный мигающий, жёлтый, снова красный и т. д. По умолчанию период горения красного в четыре раза больше периода горения зеленого. Если в данный момент не горит зеленый, то после нажатия кнопки светофор запоминает необходимость переключения на зелёный. Цикл работы светофора не нарушается, но период горения красного должен быть сокращен до ¼ своего обычного периода.

Исходный код

*Функция ожидания заданного временного интервала:*

**void** **wait**(uint32\_t duration)

{

uint32\_t startTime = HAL\_GetTick();

**while**((HAL\_GetTick() - startTime) < duration)

{

}

}

*Функция выключения всех световых индикаторов:*

**void** **turnAllOff**()

{

HAL\_GPIO\_WritePin(GPIOD, GPIO\_PIN\_13, *GPIO\_PIN\_RESET*);

HAL\_GPIO\_WritePin(GPIOD, GPIO\_PIN\_14, *GPIO\_PIN\_RESET*);

HAL\_GPIO\_WritePin(GPIOD, GPIO\_PIN\_15, *GPIO\_PIN\_RESET*);

}

*Функция включения заданного цвета:*

**void** **turnSpecificLightOn**(uint16\_t light)

{

turnAllOff();

HAL\_GPIO\_WritePin(GPIOD, light, *GPIO\_PIN\_SET*);

}

*Функция мигания заданным цветом определенное количество раз:*

**void** **blinkSpecificLight**(uint32\_t count, uint16\_t light, uint32\_t blinkDuration)

{

**for**(uint32\_t i = 0; i < count; i++){

turnAllOff();

wait(blinkDuration);

turnSpecificLightOn(light);

wait(blinkDuration);

}

}

*Основная программа:*

uint32\_t startTime = 0;

uint32\_t duration = 10000;

uint32\_t blinkDuration = 500;

uint32\_t durationForRed = 4 \* duration;

uint32\_t durationForYellow = 3000;

uint8\_t buttonFlag = 0;

/\* USER CODE BEGIN WHILE \*/

**while** (1)

{

startTime = HAL\_GetTick();

//Red on

turnSpecificLightOn(RED);

**while**((HAL\_GetTick() - startTime) < durationForRed)

{

**if**(HAL\_GPIO\_ReadPin(GPIOC, GPIO\_PIN\_15) == 0 && buttonFlag == 0) {

durationForRed = durationForRed / 4;

buttonFlag = 1;

}

}

//Red of, Green on

turnSpecificLightOn(GREEN);

wait(duration);

//Green blinks

blinkSpecificLight(3, GREEN, blinkDuration);

//Reset duration for Red

durationForRed = 4 \* duration;

buttonFlag = 0;

//Yellow on

turnSpecificLightOn(YELLOW);

startTime = HAL\_GetTick();

**while**((HAL\_GetTick() - startTime) < durationForYellow)

{

**if**(HAL\_GPIO\_ReadPin(GPIOC, GPIO\_PIN\_15) == 0 && buttonFlag == 0) {

durationForRed = durationForRed / 4;

buttonFlag = 1;

}

}

}

/\* USER CODE END 3 \*/

Вывод

Во время выполнения лабораторной работы мы получили базовые знания об устройстве стенда SDK 1.1M и изучили работу с интерфейсами ввода-вывода в микроконтроллерах, разработали собственную программу для управления световыми индикаторами в которой продемонстрировали полученные знания.