МИНОБРНАУКИ РОССИИ

Федеральное государственное автономное образовательное учреждения высшего образования

«ЮЖНЫЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт компьютерных технологий и информационной безопасности

Кафедра математического обеспечения и применения ЭВМ

  

**ЛАБОРАТОРНАЯ РАБОТА № 1**

по дисциплине

**«Операционные** **системы и системное программное обеспечение»**

на тему:

**«Знакомство эмулятором renode, IDE STM32CubeIDE и микроконтроллерами семейства STM32»**

Выполнили:

Студенты группы

КТбо3-1

|  |  |  |  |
| --- | --- | --- | --- |
|  | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | | Яшенков А. В. |
|  | *подпись* | |
|  |  |

|  |  |  |  |
| --- | --- | --- | --- |
|  | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | | Заветная В.Т. |
|  | *подпись* | |
|  |  |

Проверил:

ассистенткафедры

МОП ЭВМ

|  |  |  |
| --- | --- | --- |
|  | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | Жиглатый А.А. |
|  | *подпись* |  |

Оценка

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

«\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_ 2021 г.

Таганрог 2021

СОДЕРЖАНИЕ

1. Цель работы
2. [Техническое задание](#_bookmark1)
   1. Базовая информация по платформе STM32F4DISCOVERY с кратким описанием основного микроконтроллера и задействованной периферии
   2. Информация об основных командах rednode, краткое описание конфигурационных файлов эмулятора для платформы STM32F4DISCOVERY
   3. Описание базовых исходных файлов, генерируемых для проектов STM32 Project — startup\_stm32f407vgtx.s, stm32f4xx\_it.c, system\_stm32f4xx.c, stm32f4xx\_hal\_msp.c, STM32F407VGTX\_FLASH.ld
   4. Описание функций инициализации SystemClock\_Config и MX\_GPIO\_Init
3. Выполнение программы
   1. в окне эмулятора
   2. с помощью отладчика
4. Контрольные вопросы
5. Вывод
6. Цель работы

Знакомство с основами работы по созданию приложений для микроконтроллеров семейства STM32 в среде STM32CubeIDE.

1. Теоретический материал
   1. Базовая информация по платформе STM32F4DISCOVERY с кратким описанием основного микроконтроллера и задействованной периферии.

Плата STM32F4Discovery предназначена для ознакомления с возможностями 32-битного МК на основе ARM-архитектуры, а также для реализации собственных устройств и приложений с использованием аппаратного обеспечения платы.

Плата STM32F4 Discovery оснащена: - микроконтроллером STM32F407VGT6 с ядром Cortex-M4F тактовой частотой 168 МГц, 1 Мб Flash-памяти, 192 кб RAM в корпусе LQFP100; - отладчиком ST-Link/V2 для отладки и программирования МК; - питанием платы через USB или от внешнего источника питания 5 В; - датчиком движения ST MEMS LIS302DL и выходами цифрового акселерометра по трем осям; - датчиком звука ST MEMS MP45DT02; - звуковым ЦАП CS43L22; - восемью светодиодами: LD1 (красный/зеленый) для USB-подключения, LD2 (красный) для питания 3.3 В, четыре пользовательские светодиода: LD3 (оранжевый), LD4 (зеленый), LD5 (красный), LD6 (синий) и два светодиода для USB On-The-Go – LD7 (зеленый) и LD8 (красный)$ - двумя кнопками (для программирования пользователем и для перезапуска).

Таким образом, отладочная плата оснащена большим количеством периферии, что позволяет сразу же реализовывать на ней примеры различной сложности.

Процессоры ARM содержат до 18 регистров: 16 регистров данных и 2 регистра процессов. Все   
регистры содержат 32 бита и именуются от R0 до R15. Регистры R13, R14, R15 используются для выполнения   
определенных специфических задач:   
R13 используется в качестве указателя стека;   
R14 используется как связывающий регистр;   
R15 играет роль счетчика.

* 1. Информация об основных командах rednode, краткое описание конфигурационных файлов эмулятора для платформы STM32F4DISCOVERY.

Команда **help** предоставляет список доступных команд с кратким описанием. Он предоставляет подробную информацию при использовании с другой встроенной командой в качестве аргумента, например, **help logLevel**.

При вводе любой команды с неправильными или неполными аргументами также выводится строка справки.

Для команд с аргументами файла знак **@** представляет путь к файлу - для удобства Renode обеспечивает автозаполнение также для имен файлов.

Наиболее распространенные команды (например, **start** или **quit**) предоставляют короткие, обычно однобуквенные, псевдонимы (s и q соответственно). Интерфейс командной строки предоставляет историю команд (стрелки вверх / вниз) с интерактивным поиском (Control-r), чтобы легко повторно выполнить предыдущие команды. Также доступна вставка с помощью Control-Shift-v, а также через контекстное меню при щелчке правой кнопкой мыши. Чтобы удалить текущую команду и вернуться к чистой подсказке, используйте Control-c.

Чтобы загрузить скрипт (который в Renode обычно использует расширение «.resc»), используйте: включить @ / путь / к / script.resc Если в приведенной выше команде вы используете start (или просто s) вместо include, эмуляция начнется сразу после загрузки скрипта.

* 1. Описание базовых исходных файлов, генерируемых для проектов STM32 Project — startup\_stm32f407vgtx.s, stm32f4xx\_it.c, system\_stm32f4xx.c, stm32f4xx\_hal\_msp.c, STM32F407VGTX\_FLASH.ld.

startup\_stm32f407vgtx.s – assembly-файл, который описывает основные секции памяти, основные прерывания, которые будут использоваться нашим устройством во время работы.

STM32F407VGTX\_FLASH.ld – файл, в котором описывается отображение программных секций памяти на физические сегменты.

stm32f4xx\_it.c – файл в котором описываются обработчики прерываний и исключений.

system\_stm32f4xx.c - этот файл предоставляет две функции и одну глобальную переменную для вызова из пользовательского приложения.

stm32f4xx\_hal\_msp.c - Этот файл содержит инструкции для кодов инициализации и деинициализации MSP.

* 1. Описание функций инициализации SystemClock\_Config и MX\_GPIO\_Init.

SystemClock\_Config – функция, в которой происходит настройка выходного напряжения основного внутреннего регулятора, инициализация генераторов RCC в соответствии с параметрами, указанными в структуре RCC\_OscInitTypeDef, инициализация тактовых частот шин CPU, AHB и APB.

MX\_GPIO\_Init – функция, в которой включение часов портов GPIO, настройка выходного уровня вывода GPIO и конфигурация GPIO-пинов.

1. Выполнение программы
   1. Работоспособность приложения: а) в окне эмулятора (изменения состояния GPIO ножек соответствующих светодиодам на плате) — с помощью команды sysbus.gpioPortD GetGPIOs;

Изображение выглядит как текст

Автоматически созданное описание

Изображение выглядит как текст

Автоматически созданное описание

Изображение выглядит как текст

Автоматически созданное описание

Изображение выглядит как текст

Автоматически созданное описание

* 1. Работоспособность приложения: б) с помощью отладчика из STM32CubeIDE — в режиме пошаговой отладке с помощью точек останова показать, что код выполняется корректно (бесконечно зацикливается в функции main, изменяя состояние светодиодов).

Изображение выглядит как текст, снимок экрана, компьютер, внутренний

Автоматически созданное описание

Изображение выглядит как текст, снимок экрана, внутренний, рабочий стол

Автоматически созданное описание

Изображение выглядит как текст, снимок экрана, внутренний, рабочий стол

Автоматически созданное описание

Изображение выглядит как текст, снимок экрана, внутренний, рабочий стол

Автоматически созданное описание

1. Контрольные вопросы
2. Какой GPIO-порт и ножки используются на плате STM32F4DISCOVERY для реализации подключения светодиодов? Пользовательской кнопки?

PD15-PD12 – ножки

GPIOD – Порт для подключения светодиодов

Порт PA0 – пользовательская кнопк

1. Для чего может применяться эмулятор renode?

Renode используется для эмулирования платы для отладки приложения, когда нет физической платы

1. С помощью какой команды эмулятора renode можно посмотреть подключенную к системной шине sysbus периферию?

С помощью команды peripherals.

1. Для чего необходимо окошко Pinout view конфигуратора проектов STM32 Project?

Для настройки функционала пинов

1. Для чего используются функции HAL\_GPIO\_TogglePin/LL\_GPIO\_TogglePin?

Для инвертирования состояния пина

1. В чем отличия драйвера HAL от драйвера LL?

HAL-драйверы ориентированы не на периферийные устройства микроконтроллера, а на функциональные возможности этих устройств.

Драйверы LL обеспечивают элементарные операции с регистрами периферии микроконтроллера. Их функции в точности повторяют возможные действия с регистрами. По сути, это те же операции с именами библиотеки CMSIS, только используется не прямой доступ к регистрам, а функции. И данные для конфигурации задаются не в числовой форме, а в виде символьных констант

1. Что настраивается на вкладке Clock Configuration конфигуратора проектов STM32 Project?

Настройка тактирования перифери

1. Что за код находится в файле startup\_stm32f407vgtx.s? За что отвечает секция isr\_vector?

В файле startup\_stm32f407vgtx.s находится assembly-код, который описывает основные секции памяти, основные прерывания, которые будут использоваться нашим устройством во время работы.

isr\_vector - минимальная таблица векторов

1. Что за код находится в файле stm32f4xx\_it.c?

Обработчики прерываний