

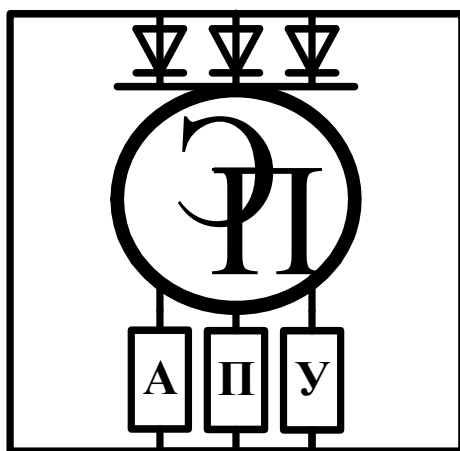
МЕЖГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
«БЕЛОРУССКО-РОССИЙСКИЙ УНИВЕРСИТЕТ»

Кафедра «Электропривод и автоматизация промышленных установок»

МИКРОПРОЦЕССОРНЫЕ СРЕДСТВА В АВТОМАТИЗИРОВАННОМ ЭЛЕКТРОПРИВОДЕ

*Методические рекомендации к лабораторным работам
для студентов специальности
1-53 01 05 «Автоматизированные электроприводы»
дневной и заочной форм обучения*

Часть 1



Могилев 2023

УДК 004.42
ББК 32.973.26
М75

Рекомендовано к изданию
учебно-методическим отделом
Белорусско-Российского университета

Одобрено кафедрой «Электропривод и автоматизация промышленных установок» «17» сентября 2022 г., протокол № 2

Составители: ст. преподаватель О. А. Капитонов;
ассистент Д. В. Шнип

Рецензент С. В. Болотов

Методические рекомендации к лабораторным работам предназначены для студентов специальности 1-53 01 05 «Автоматизированные электроприводы» очной и заочной форм обучения.

Учебное издание

МИКРОПРОЦЕССОРНЫЕ СРЕДСТВА
В АВТОМАТИЗИРОВАННОМ ЭЛЕКТРОПРИВОДЕ

Часть 1

Ответственный за выпуск	С. М. Фурманов
Корректор	А. А. Подошевка
Компьютерная верстка	М. М. Дударева

Подписано в печать 11.05.2023 . Формат 60×84/16. Бумага офсетная. Гарнитура Таймс.
Печать трафаретная. Усл. печ. л. 2,79. Уч.-изд. л. 2,94. Тираж 81 экз. Заказ № 558.

Издатель и полиграфическое исполнение:
Межгосударственное образовательное учреждение высшего образования
«Белорусско-Российский университет».
Свидетельство о государственной регистрации издателя,
изготовителя, распространителя печатных изданий
№ 1/156 от 07.03.2019.
Пр-т Мира, 43, 212022, г. Могилев.

© Белорусско-Российский
университет, 2023

Содержание

1 Методические указания к лабораторным работам	4
2 Лабораторная работа № 1. Изучение технических параметров, устройства и порядка работы с учебным оборудованием	4
3 Лабораторная работа № 2. Изучение утилиты для генерирования кода, инициализирующей микроконтроллеры STM32F4xx.....	9
4 Лабораторная работа № 3. Изучение организации ввода-вывода дискретных сигналов в микропроцессорных устройствах	11
5 Лабораторная работа № 4. Изучение организации параллельных портов в микропроцессорных устройствах	16
6 Лабораторная работа № 5. Изучение организации жидкокристаллической индикации в микропроцессорных устройствах	18
7 Лабораторная работа № 6. Изучение системы прерываний микропроцессорных устройств.....	22
8 Лабораторная работа № 7. Изучение организации таймеров в микропроцессорных устройствах	26
9 Лабораторная работа № 8. Изучение организации семисегментной индикации в микропроцессорных устройствах	28
10 Лабораторная работа № 9. Изучение организации клавиатуры в микропроцессорных устройствах	31
11 Лабораторная работа № 10. Изучение организации устройств ввода аналоговой информации в микропроцессорных устройствах.....	33
12 Лабораторная работа № 11. Изучение организации устройств вывода аналоговой информации в микропроцессорных устройствах.....	36
13 Лабораторная работа №12. Изучение организации последовательных портов в микропроцессорных устройствах	39
14 Лабораторная работа № 13. Разработка и отладка программ управления технологическими объектами в реальном времени	45
Список литературы	47

1 Методические указания к лабораторным работам

Перед началом лабораторного занятия необходимо изучить теоретические сведения к лабораторной работе.

По полученному заданию студент должен:

- произвести подробный анализ поставленной задачи – выделить список задач, необходимых для ее решения, определить настройки используемых периферийных узлов микроконтроллера, обеспечивающие решение задачи;
- составить блок-схему алгоритма её выполнения;
- в графической среде STM32CubeMX подготовить проект программы, обеспечивающий настройку аппаратной части микроконтроллера;
- в среде Keil μ Vision полученную программу привести в соответствие с разработанной блок-схемой;
- загрузить собранную программу в память микроконтроллера учебного стенда;
- произвести отладку программы;
- составить отчет по работе;
- ответить на контрольные вопросы.

Отчет по лабораторной работе должен содержать:

- цель работы;
- постановка задачи;
- схема электрическая принципиальная к лабораторной работе;
- блок-схема алгоритма пользовательской программы;
- текст пользовательской программы;
- выводы по работе.

2 Лабораторная работа № 1. Изучение технических параметров, устройства и порядка работы с учебным оборудованием

Цель работы: изучить состав, функциональную схему учебного стенда, ознакомиться со структурой и принципом работы микроконтроллера, изучить порядок работы платы.

2.1 Структура учебного стенда

Структурная схема учебного стенда приведена на рисунке 2.1.

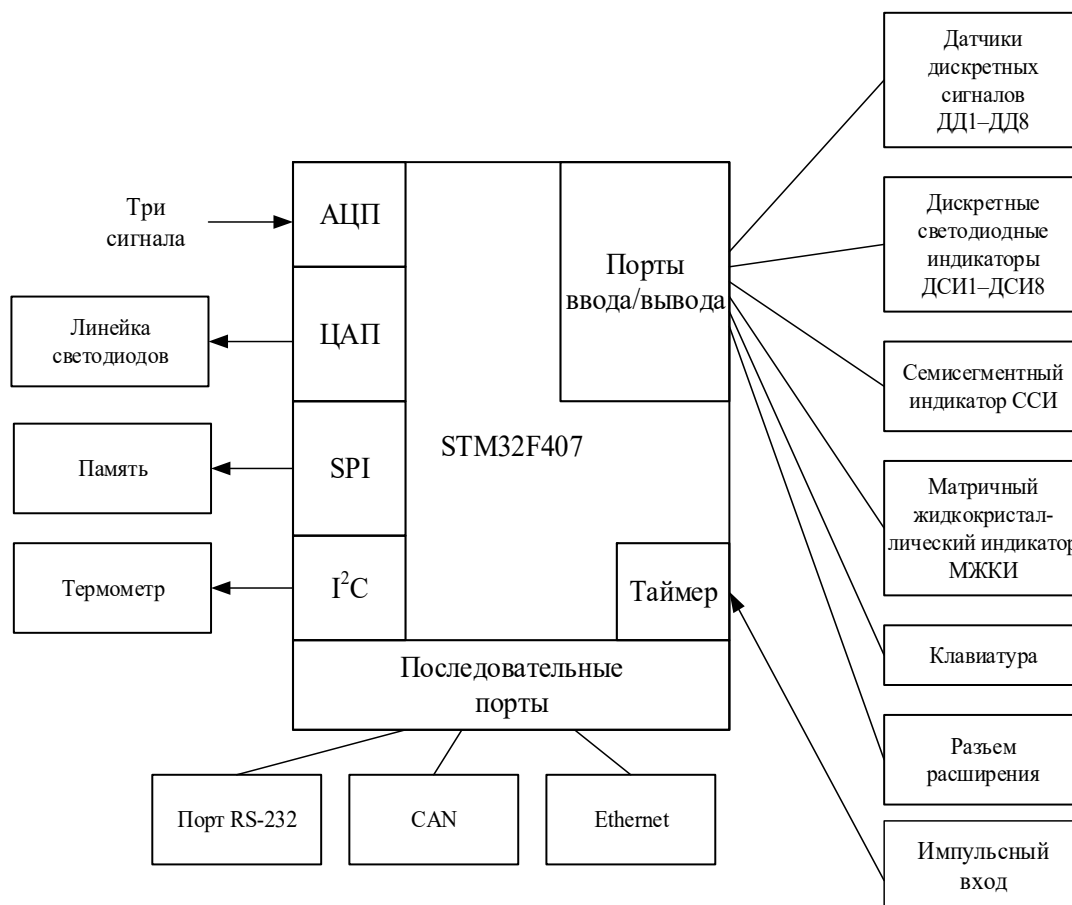


Рисунок 2.1 – Структурная схема учебного стенда

Стенд построен на базе микроконтроллера STM32F407ZET6 семейства ARM Cortex-M4. Контроллер обладает большим количеством встроенных периферийных устройств, поэтому все органы управления и индикации подключены непосредственно к выводам микроконтроллера.

Стенд позволяет исследовать ввод и вывод информации, представленной в различном виде. Так, для исследования вывода используются:

- дискретные светодиодные индикаторы;
- четырехсимвольный семисегментный светодиодный индикатор;
- устройство матричной жидкокристаллической индикации;
- светодиодная линейка.

Для исследования ввода используются:

- датчики дискретных сигналов;
- двенадцатикнопочная клавиатура;
- три аналоговых сигнала.

На плате организован ряд последовательных интерфейсов: I2C – используется для обмена информацией с датчиком температуры, SPI – используется для подключения к электрически стираемому ППЗУ, для связи с другими устройствами, например с персональным компьютером. Кроме того, стенд оснащен последовательными портами RS232, CAN и Ethernet.

Кроме того, предусмотрена возможность подключения внешних модулей расширения при помощи разъема.

Для питания платы используется внешний блок питания на 12 В.

2.2 Органы управления и индикации учебной платы *СУ-ARM-СМ4*

Необходимо обратить внимание на то, что учебный стенд выполнен в бескорпусном исполнении – на металлических стойках расположена основная печатная плата. Все органы расположены непосредственно на поверхности печатной платы. Внешний вид учебной платы приведен на рисунке 2.2, расположение элементов поясняется на рисунке 2.3.

На плате расположены:

- матричный жидкокристаллический индикатор МЖКИ (1), контрастностью которого можно управлять при помощи переменного резистора (1');
- четырехсимвольный семисегментный светодиодный индикатор ССИ (2);
- дискретный светодиодный индикатор (набор из восьми светодиодов ДСИ1–ДСИ8) (3) (нумерация светодиодов приведена на рисунке 2.4, а);
- линейка светодиодов – индикатор аналогового сигнала на выходе ЦАП–ЛСИ (4);
- датчики дискретных сигналов (набор из восьми переключателей ДД1–ДД8) (5) (нумерация переключателей приведена на рисунке 2.4, в);
- имитаторы аналогового сигнала на входах АЦП АД1–АД3 (6) (нумерация аналоговых сигналов приведена на рисунке 2.4, б);
- двенадцатикнопочная клавиатура (7);
- кнопка сброса (8);
- для регулирования параметров входного импульсного сигнала на плате установлены резисторы (9).

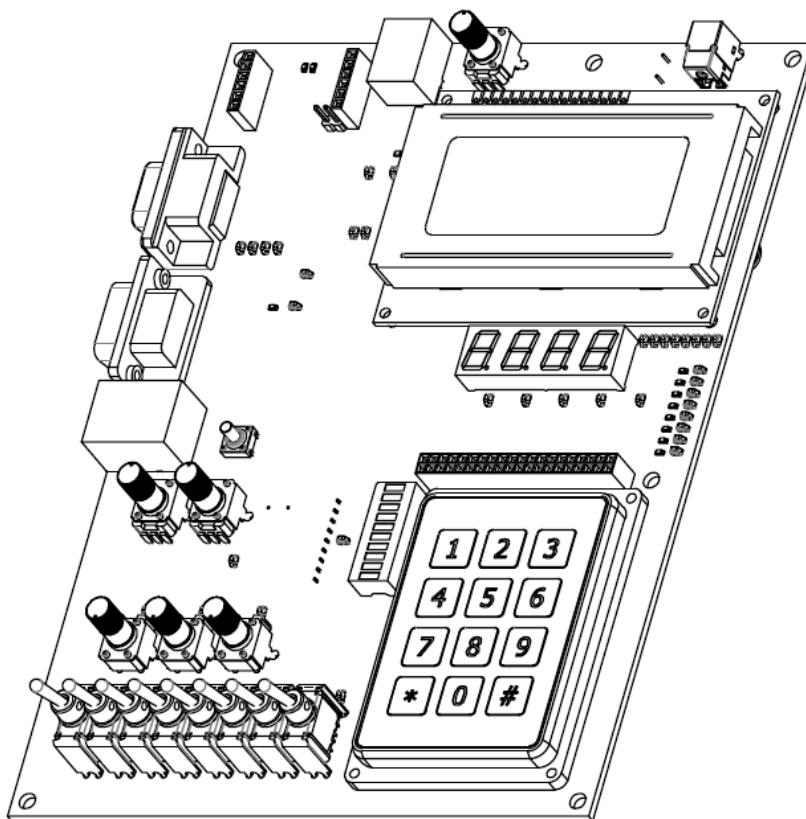


Рисунок 2.2 – Внешний вид платы стенда

Кроме того, на плате находятся разъемы:

- для подключения питания (10);
- отладки (11);
- Ethernet (12);
- разъем RS-232, предоставляющий возможность подключения стенда к персональному компьютеру (13);
- разъем CAN-интерфейса (14);
- разъем расширения для подключения дополнительных внешних устройств (15).

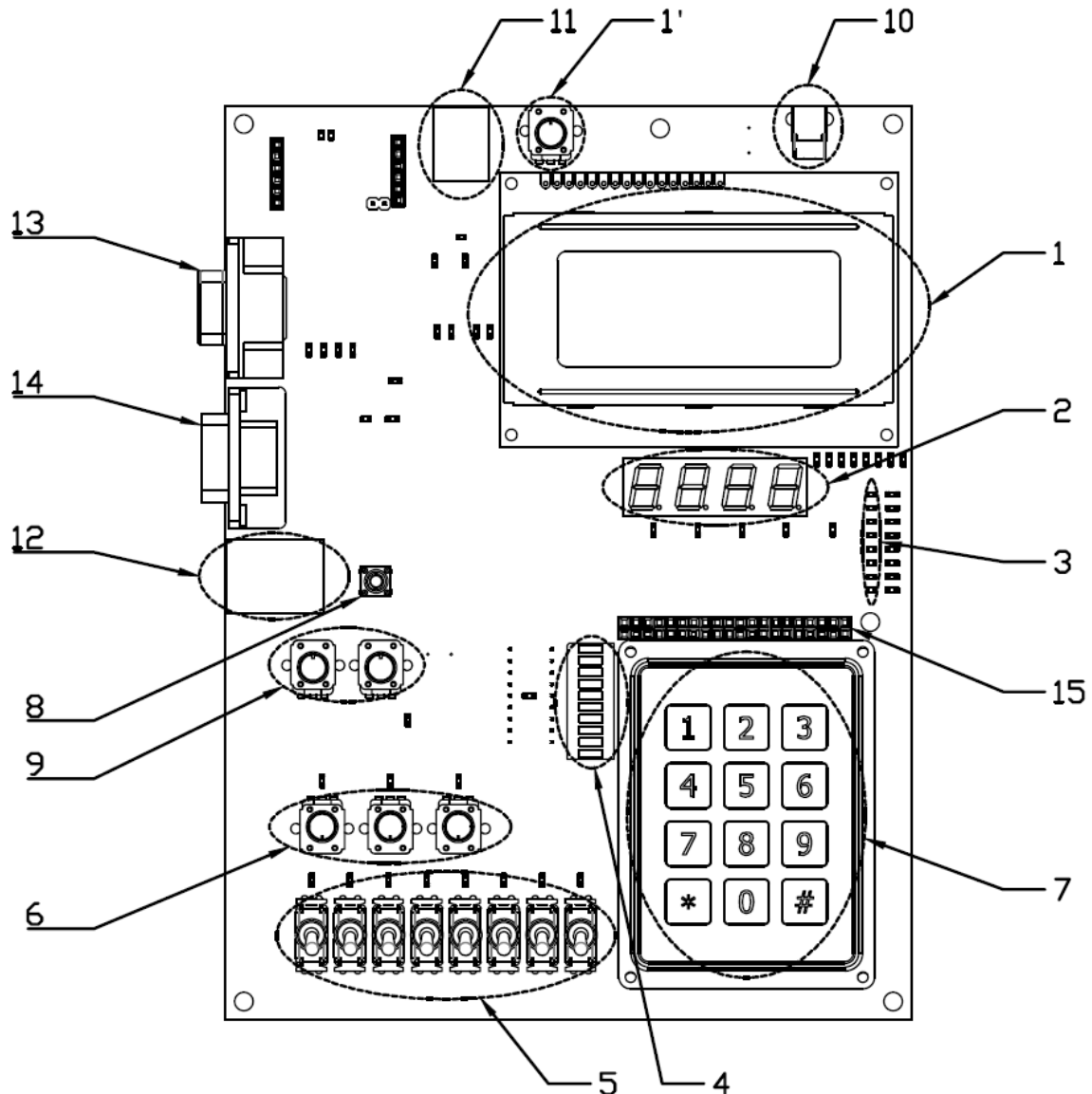
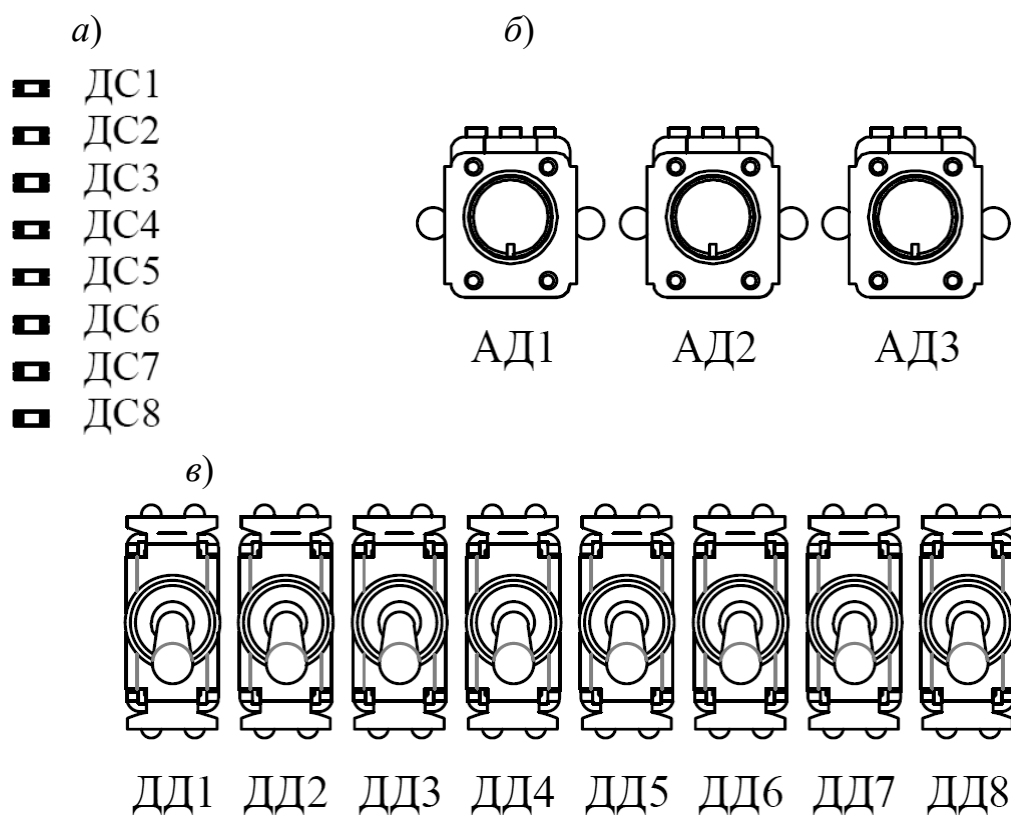


Рисунок 2.3 – Расположение органов на плате

Более подробные сведения о характеристиках применяемого микроконтроллера и возможностях приложения **STM32 ST-LINK Utility**, используемого для загрузки программ в память контроллера, можно найти в методических указаниях по лабораторной работе № 1, электронная версия которых имеется в классе ПЭВМ кафедры (лаб. 207/2) по следующему адресу:

D:\методические рекомендации\СТАНДАРТЫ РБ\Специальность 1-53 01 05 АЭП и АПУ\Микропроцессорные средства в автоматизированном электроприводе\МПС 01.pdf, а также на сервере филиала кафедры по адресу \\Study\Методические указания\МПСвАЭП\МПС 01.pdf.



а – светодиодов ДСИ; б – имитаторов аналоговых сигналов; в – датчиков ДД

Рисунок 2.4 – Расположение органов управления

2.3 Порядок выполнения работы

- 1 Подключить стенд к персональному компьютеру.
- 2 Подключить стенд с помощью адаптера к сети 220 В.
- 3 Загрузить утилиту STM32 ST-LINK Utility.
- 4 Загрузить программу диагностики стенда в утилиту. Расположение hex-файла программы уточнить у преподавателя.
- 5 Записать программу в память программ учебного стенда. Программа диагностики предоставляет пользователю меню, позволяющее протестировать большинство органов управления и индикации стенда: пункт АЦП – позволяет проверить работу аналогово-цифрового и цифроаналогового преобразователей, пункт ДД – дискретных датчиков ДД1–ДД8, пункт Терм – состояние электронного термометра, пункт Время – ССИ.
- 6 Пункт Счетный вход – измерить параметры входного импульсного сигнала. Управление меню осуществляется при помощи клавиатуры.
- 7 Используя клавиатуру, перейти к тестированию ДД. С помощью ДД1–ДД8

выставить двоичный код десятиразрядного числа в соответствии с заданием преподавателя (верхнее положение тумблера – логическая «1», нижнее – логический «0»). Состояние тумблеров проверить на МЖКИ.

8 Используя клавиатуру, перейти к тестированию АЦП/ЦАП. Установить АД1, АД2, АД3 поочередно в крайнее левое, среднее, крайнее правое положение. Результат преобразования аналогового сигнала в цифровой код наблюдать на МЖКИ.

Результат работы ЦАП можно наблюдать на ЛСИ. Выбор аналогового сигнала, управляющего ЦАП, можно выбрать из меню.

9 Работа ДСИ контролируется с помощью изменения состояния датчиков ДД1–ДД8, находясь в соответствующем меню.

10 Используя клавиатуру, перейти к проверке ССИ. В появившемся меню необходимо выбрать, какую величину следует изменять при помощи клавиатуры – часы, минуты, секунды – и установить их в значение соответствующее текущему времени. После ввода соответствующее значение отображается на ССИ: старшие два разряда – часы, младшие два разряда – минуты.

11 Пользуясь клавиатурой перейти к проверке параметров входного импульсного сигнала. Пользуясь соответствующими переменными резисторами, расположенными на плате, установить поочередно их в крайнее левое, среднее, крайнее правое положение. Значение периода импульсного сигнала наблюдать на МЖКИ.

12 По результатам выполненных операций сделать выводы.

Контрольные вопросы

1 Перечислите характерные черты архитектуры однокристальных микроконтроллеров.

2 Какие структурные элементы входят в состав микроконтроллеров Cortex-M4 на примере семейства STM32F4xx?

3 Каковы основные структурные элементы учебного стенда?

4 Опишите органы управления лабораторного стенда.

5 Каков порядок работы со стендом?

3 Лабораторная работа № 2. Изучение утилиты для генерирования кода, инициализирующего микроконтроллеры STM32F4xx

Цель работы: изучить набор периферийных устройств микроконтроллеров семейства STM32F4xx, а также способ их настройки с использованием STM32 CubeMX.

В ходе работы необходимо:

- изучить структурную схему микроконтроллера;
- изучить назначение выводов микроконтроллера в составе учебного стенда;
- составить шаблон программы для учебного стенда;

– составить отчет о выполнении работы.

3.1 Основные сведения о STM32 CubeMX

STM32CubeMX является частью STMicroelectronics STMCube™, положившей начало облегчения жизни разработчиков за счет сокращения усилий, времени и затрат на разработку. STM32Cube охватывает весь пакет STM32.

STM32Cube включает STM32CubeMX, графический инструмент для настройки программного обеспечения, который позволяет генерировать код инициализации C с использованием графической части.

Он также включает комплексную программную платформу, поставляемую на каждую серию (например, STM32CubeF4 для серии STM32F4). Эта платформа включает в себя STM32Cube HAL (встроенное программное обеспечение уровня абстракции STM32, обеспечивающее максимальную мобильность в портфеле STM32), STM32Cube LL (низкоуровневые API, быстрый, легкий, ориентированный на экспертный уровень), а также последовательный набор промежуточного программного обеспечения, таких как RTOS, USB, TCP / IP и графика. Все встроенные программные утилиты поставляются с полным набором примеров.

STM32CubeMX – это графический инструмент, который позволяет легко настраивать микроконтроллеры STM32 и генерировать соответствующий код инициализации C посредством пошагового процесса.

Ключевые особенности:

- интуитивно понятный выбор контроллеров;
- графическая настройка микроконтроллера;
- выводы с автоматическим решением проблем;
- последовательность питания с оценкой результатов потребления;
- функциональные режимы периферийных устройств и промежуточного программного обеспечения, и инициализация с динамической проверкой ограничения параметров;
- генерация кода C, включающая инициализацию микроконтроллера STM32, совместимую с компиляторами IAR™, Keil™ и GCC;
- доступно как автономное программное обеспечение, работающее под управлением Windows®, Linux® и macOS®, или через плагин Eclipse.

Более подробные сведения о структуре используемого в стенде микроконтроллера, назначении основных органов управления в среде STM32 CubeMX можно найти в методических указаниях по лабораторной работе № 2, электронная версия которых имеется в классе ПЭВМ кафедры (лаб. 207/2) по следующему адресу: D:\методические рекомендации\СТАНДАРТЫ РБ\Специальность 1-53 01 05 АЭП и АПУ\Микропроцессорные средства в автоматизированном электроприводе\МПС 02.pdf, а также на сервере филиала кафедры по адресу \\Study\Методические указания\МПСвАЭП\МПС 02.pdf.

3.2 Варианты индивидуальных заданий к лабораторной работе

Разработать шаблон проекта в STM32 CubeMX для учебного стенда, в котором произведена настройка тактового генератора, а также входы/выходы в соответствии с заданием. Вариант задания получить у преподавателя.

По полученному заданию студент должен:

- выполнить настройку микроконтроллера;
- на каждом этапе настройки выполнять фиксацию результатов;
- составить отчет по работе.

4 Лабораторная работа № 3. Изучение организации ввода-вывода дискретных сигналов в микропроцессорных устройствах

Цель работы: изучить структуру и особенности работы портов микроконтроллера, схему подключения входных и выходных дискретных сигналов к микроконтроллеру, особенности программирования ввода-вывода дискретных сигналов на языке программирования микроконтроллеров. Составить программу ввода, обработки по заданному алгоритму и вывода дискретных сигналов, записать в память программ микроконтроллера и выполнить.

Перед началом лабораторных занятий студентам необходимо изучить следующие вопросы.

1 Характерные черты архитектуры однокристальных микроконтроллеров, направленные на взаимодействие с объектами управления.

2 Альтернативные функции параллельных портов.

3 В каком состоянии находятся параллельные порты после формирования сигнала NRST?

4 Может ли порт одновременно являться источником операнда и приемником результата операции?

5 Как инвертировать отдельные биты портов?

4.1 Параллельные порты ввода/вывода информации микроконтроллера STM32F4xx

4.1.1 Общая информация.

Порты контроллеров данного семейства представляют собой 16-разрядные устройства ввода/вывода информации. Основное назначение – ввод/вывод дискретных сигналов в микроконтроллер, однако большинство линий портов можно использовать и для других целей. Каждый порт содержит управляемые регистры-защёлки входа и выхода, входной буфер и выходной драйвер. Структурная схема одной линии порта приведена на рисунке 4.1.

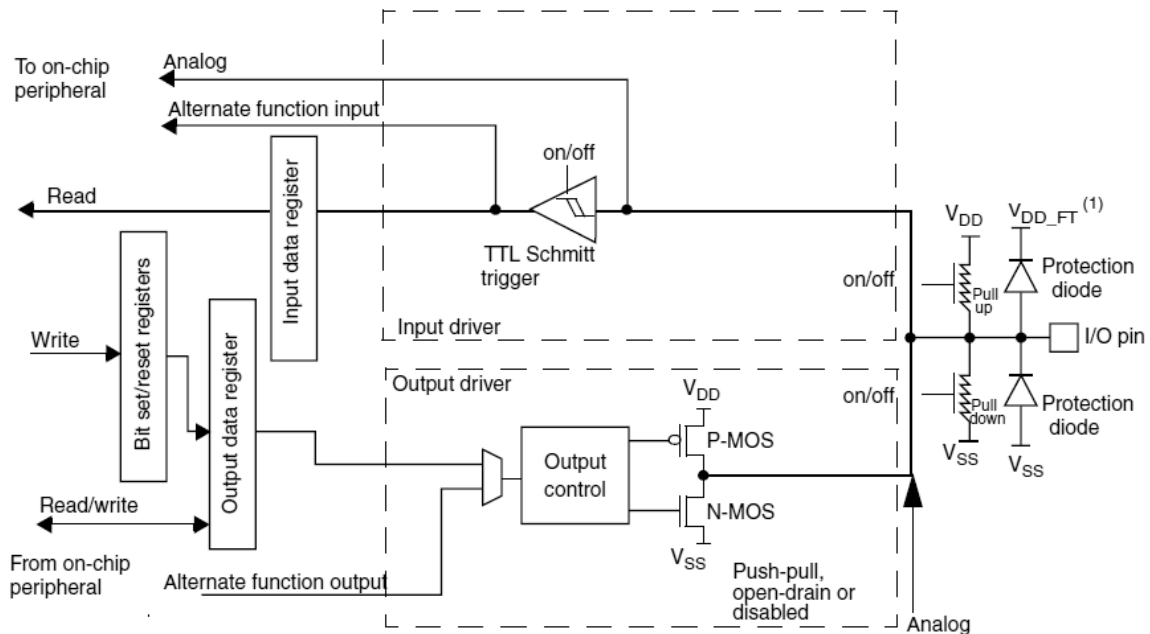


Рисунок 4.1 – Пример организации линии порта

Каждый вывод порта включает в свой состав входной и выходной драйверы (Input driver, Output driver), регистр входных данных (Input data register), а также регистры управления выходом порта: установки/сброса битов (Bit set/reset registers) и выходных данных (Output data register). Каждая линия порта имеет на выходе защитные диоды (Protection diode) и управляемые резисторы подтяжки. Как понятно из названия:

- регистр входных данных позволяет прочесть состояние порта;
- регистры управления выходами управляют состоянием порта;
- входной драйвер осуществляет ввод данных с линии порта;
- выходной драйвер изменяет состояния линии порта.

Входной драйвер состоит из управляемого триггера Шмитта.

Выходной драйвер состоит из устройства управления (Output control) и выходного каскада, представленного двумя полевыми транзисторами, которые могут находиться в трех состояниях.

По структурной схеме видно, что линии порта могут не только вводить/выводить дискретные сигналы, но и подключаются к внутренним периферийным устройствам микроконтроллера, придавая порту дополнительные функции (Analog Input, Alternate Function Input, Alternate Function Output).

Порты контроллера могут работать в разных режимах:

- плавающий (floating) вход (находящийся в высокоимпедансном состоянии);
- вход с подтяжкой вверх;
- вход с подтяжкой вниз;
- аналоговый вход (для каналов АЦП);
- выход с открытым коллектором (стоком, если быть точным);
- выход push-pull (тяги-толкай);
- альтернативные функции, т. е. работа от периферии.

Каждый порт содержит четыре 32-битных регистра конфигурации (GPIOx_MODER, GPIOx_OTYPER, GPIOx_OSPEEDR и GPIOx_PUPDR), два 32-битных регистра данных (GPIOx_IDR, GPIOx_ODR), 32-битный регистр установки/сброса (GPIOx_BSRR) и 32-битный регистр блокировки (GPIOx_LCKR).

Каждый вывод порта свободно конфигурируем, однако регистры доступны только как 32-битные слова (побайтный доступ или доступ по полслова не возможны). Регистр GPIOx_BSRR предоставляет атомарный доступ на чтение/модификацию к каждому выходу порта.

В таблице 4.1 приведены возможные варианты конфигурации выводов порта.

Таблица 4.1 – Конфигурация *i*-го вывода порта

MODER(i) [1:0]	OTYPER(i)	OSPEEDR(i) [B:A]		PUPDR(i) [1:0]		Режим конфигурации линии порта		
1	2	3		4		5		
01	0	SPEED [B:A]		0	0	GP-выход	PP	
	0			0	1	GP-выход	PP+PU	
	0			1	0	GP-выход	PP+PD	
	0			1	1	Зарезервировано		
	1			0	0	0	GP-выход	PP
	1			0	1	GP-выход	PP+PU	
	1			1	0	GP-выход	PP+PD	
	1			1	1	1	Зарезервировано (GP-выход OD)	
10	0	SPEED [B:A]		0	0	AF	PP	
	0			0	1	AF	PP+PU	
	0			1	0	AF	PP+PD	
	0			1	1	Зарезервировано		
	1			0	0	0	AF	OD
	1			0	1	AF	OD+PU	
	1			1	0	AF	OD+PD	
	1			1	1	1	Зарезервировано	
00	x	x	x	0	0	Вход	Плавающий	
	x	x	x	0	1	Вход	PU	
	x	x	x	1	0	Вход	PD	
	x	x	x	1	1	Зарезервировано (плавающий вход)		
11	x	x	x	0	0	Вход/выход	Аналоговый	
	x	x	x	0	1	Зарезервировано		
	x	x	x	1	0			
	x	x	x	1	1			

Примечание – GP – вывод порта общего назначения; PP – push-pull; PU – включена подтяжка вверх; PD – включена подтяжка вниз; OD – вывод порта с открытым стоком; AF – вывод альтернативной функции порта; x – не имеет значения

После сброса альтернативные функции портов деактивируются и все выходы портов настраиваются в режим плавающий вход.

Когда вывод настроен на вход регистр GPIOx_IDR в каждом цикле шины АНВ1 захватывает данные, находящиеся на входе порта.

Когда вывод настроен на выход, значение, записанное в регистр GPIOx_ODR, выдается на вывод порта. Выходной драйвер можно использовать в режиме push-pull или с открытым стоком (в этом случае, когда на выход выдается 0, активируется только N–MOS-транзистор драйвера).

Все выводы порта содержат слабые подтягивающие резисторы, которые могут быть активированы или нет в зависимости от значения, записанного в регистр GPIOx_PUPDR.

Кроме того, выводы портов подключены к встроенным периферийным модулям контроллера через мультиплексор, который позволяет подключить только одну из альтернативных периферийных функций (AF) к выводу порта. Поэтому не возникает конфликтов между периферийными модулями, которые подключены к одному и тому же выводу порта.

Каждый вывод порта содержит мультиплексор с 16 входами альтернативных функций (от AF0 до AF15), которые могут быть сконфигурированы при помощи регистров GPIOx_AFRL (для выводов с 0-го по 7-й) и GPIOx_AFRH (для выводов с 8-го по 15-й):

- после сброса все выводы подключаются к системной альтернативной функции 0 (AF0);
- альтернативные функции периферийных модулей сопоставлены с функциями от AF1 до AF13;
- событие Cortex®-M4 с FPU сопоставлено с AF15.

Для того чтобы получить более подробную информацию о соответствии и закреплении за конкретными выводами портов системных и периферийных альтернативных функций, см. таблицу «Alternate function mapping» в документации на микроконтроллер.

Более подробные сведения о режимах работы и способах настройки линий параллельных портов микроконтроллера можно найти в методических указаниях по лабораторной работе № 3, электронная версия которых имеется в классе ПЭВМ кафедры (лаб. 207/2) по следующему адресу: D:\методические рекомендации\СТАНДАРТЫ РБ\Специальность 1-53 01 05 АЭП и АПУ\Микропроцессорные средства в автоматизированном электроприводе\МПС 03.pdf, а также на сервере филиала кафедры по адресу \\Study\Методические указания\МПСвАЭП\МПС 03.pdf.

4.2 Ввод дискретных сигналов

Для ввода дискретной информации в микроконтроллер широко применяются различные переключатели, кнопки и клавиатуры. Пример простейшей схемы подключения источника дискретного сигнала в виде кнопки к порту микроконтроллера приведен на рисунке 4.2.

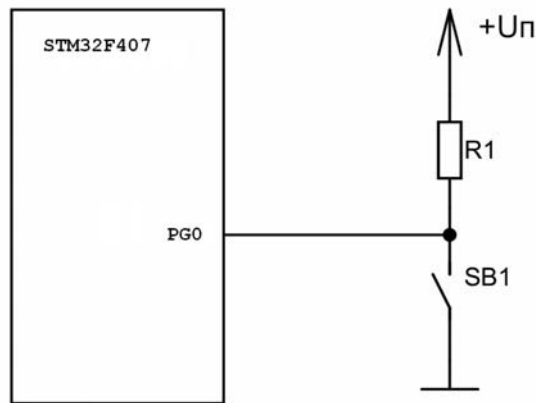


Рисунок 4.2 – Подключение источника дискретных сигналов к порту микроконтроллера

4.3 Вывод дискретных сигналов

Для управления исполнительным устройством, работающим по принципу включено/выключено, на соответствующей выходной линии порта необходимо сформировать статический сигнал 0 или 1, что реализуется командами вывода непосредственного операнда, содержащего в требуемом бите значение 0 или 1.

На рисунке 4.3 приведен пример простейшего исполнительного устройства в виде светодиода, подключенного к порту микроконтроллера.

В случае параллельного управления группой автономных исполнительных устройств, подключенных к выходному порту, формируется не двоичное управляющее воздействие, а управляющее слово, каждому из разрядов которого ставится в соответствие 1 или 0 в зависимости от того, какие исполнительные устройства должны быть включены, а какие выключены.

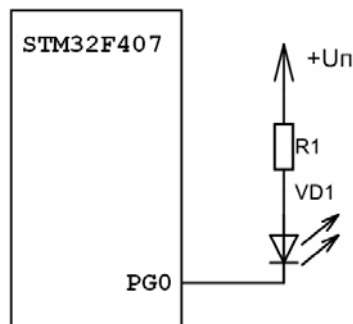


Рисунок 4.3 – Подключение исполнительного устройства к порту микроконтроллера

4.4 Варианты индивидуальных заданий к лабораторной работе

Разработать программу для учебного стенда, позволяющую выполнять следующие действия в соответствии с заданием. Вариант задания получить у преподавателя.

Более подробные сведения о способах ввода-вывода дискретных сигналов с примером программы можно найти в методических указаниях по лаборатор-

ной работе № 3, электронная версия которых имеется в классе ПЭВМ кафедры (лаб. 207/2) по следующему адресу: D:\методические рекомендации\СТАНДАРТЫ РБ\Специальность 1-53 01 05 АЭП и АПУ\Микропроцессорные средства в автоматизированном электроприводе\МПС 03.pdf, а также на сервере филиала кафедры по адресу \\Study\Методические указания\МПСвАЭП\МПС 03.pdf.

Контрольные вопросы

- 1 Каким образом организуется ожидание статического сигнала?
- 2 Каким образом организуется ожидание импульсного сигнала?
- 3 Каковы ограничения на длительность импульса? Как эти ограничения можно преодолеть?
- 4 В чем сущность явления дребезга контактов? Какие есть способы для преодоления этого явления?
- 5 Как реализуется способ устранения дребезга путем подсчета числа совпадающих значений сигнала?
- 6 Как реализуется способ устранения дребезга путем временной задержки?
- 7 Как организуется периодический опрос датчиков?
- 8 Как организуется опрос группы дискретных датчиков?
- 9 Как организуется вывод статических дискретных сигналов?
- 10 Как организуется вывод импульсных дискретных сигналов?

5 Лабораторная работа № 4. Изучение организации параллельных портов в микропроцессорных устройствах

Цель работы: изучить структуру и особенности работы портов микроконтроллера, различные схемы внешних портов ввода-вывода дискретных сигналов; составить программу ввода, обработки по заданному алгоритму и вывода дискретных сигналов, записать в память программ микроконтроллера и выполнить.

Перед началом лабораторных занятий студентам необходимо изучить следующие вопросы:

- параллельные порты микроконтроллеров семейства STM32F4xx;
- основные схемы триггеров;
- использование микросхем параллельных регистров;
- использование микросхем последовательных регистров.

5.1 Способы построения внешних портов ввода-вывода

Микроконтроллеры семейства STM32F4xx имеют в своем составе несколько параллельных портов ввода/вывода. Например, STM32F401CB оснащен четырьмя портами: GPIOA (16 разрядов), GPIOB (15 разрядов), GPIOC (три разряда), GPIOD (два разряда), суммарное количество линий ввода/вывода

дискретных сигналов составляет 36. А у контроллера STM32F407I девять портов (А, В, С, D, Е, F, H, I) с суммарным количеством линий ввода/вывода – 140. Хотя это достаточно много, но существуют приложения, в которых и такого количества портов ввода/вывода недостаточно. Тем более, все линии, кроме функций стандартного ввода/вывода, поддерживают дополнительные аппаратные функции.

Существуют различные схемы расширения портов ввода-вывода в основе которых лежат параллельные или сдвиговые регистры.

Для организации однонаправленного вывода можно использовать микросхемы регистров-защелок 1533ИР22, 1533ИР33, 580ИР82, 580ИР83 и др. Для организации двунаправленной передачи информации можно использовать микросхемы с тремя состояниями на выходах или микросхемы интегральных приемопередатчиков (555АП6, 580ВА86, 580ВА87 и др.).

Микросхемы сдвиговых регистров (74НС595, 74НС165) можно использовать в тех случаях, когда количество свободных линий микроконтроллера ограничено. Они позволяют использовать для сопряжения с контроллером последовательный интерфейс, в то время как передача информации в систему или из нее осуществляется в параллельном виде.

5.2 Варианты индивидуальных заданий к лабораторной работе

Разработать программу для учебного стенда, позволяющую выполнять определенные действия с дискретными светодиодными индикаторами ДСИ, в зависимости от состояния дискретных датчиков ДД. Обмен данными производить в параллельном коде: вводить с ДД, выдавать на ДСИ. Вариант задания получить у преподавателя.

Более подробные сведения о способах построения внешних портов ввода-вывода с примером программы, обеспечивающей ввод-вывод параллельных данных, можно найти в методических указаниях по лабораторной работе № 4, электронная версия которых имеется в классе ПЭВМ кафедры (лаб. 207/2) по следующему адресу: D:\методические рекомендации\СТАНДАРТЫ РБ\Специальность 1-53 01 05 АЭП и АПУ\Микропроцессорные средства в автоматизированном электроприводе\МПС 04.pdf, а также на сервере филиала кафедры по адресу \\Study\Методические указания\МПСвАЭП\МПС 04.pdf.

Контрольные вопросы

- 1 Назовите способы построения внешних портов ввода-вывода.
- 2 Использование микросхем параллельных регистров.
- 3 Использование микросхем интегральных приемопередатчиков.
- 4 Использование микросхемы сдвиговых регистров.

6 Лабораторная работа № 5. Изучение организации жидкокристаллической индикации в микропроцессорных устройствах

Цель работы: ознакомиться с организацией и техническими параметрами устройства матричной жидкокристаллической индикации; изучить схему сопряжения микроконтроллера с устройством матричной индикации; разработать и отладить программу вывода информации на устройство матричной жидкокристаллической индикации.

Перед началом лабораторных занятий студентам необходимо изучить следующие вопросы:

- принцип действия жидкокристаллических индикаторов (ЖКИ);
- основные параметры жидкокристаллических индикаторов.

6.1 Теоретические основы и принципы работы ЖКИ

Жидкокристаллические индикаторы управляют отражением и пропусканием света для создания изображений цифр, букв, символов и т. д. В отличие от светодиодов, жидкокристаллические индикаторы не излучают свет.

Термином жидкий кристалл обозначается мезофаза между твердым состоянием и изотропным жидким состоянием, при этом мезофаза сохраняет фундаментальные свойства, присущие двум состояниям материи. Жидкие кристаллы, с одной стороны, обладают текучестью как изотропная жидкость, с другой стороны, сохраняет определенный порядок в расположении молекул (как кристалл).

Таким образом, основу ЖКИ составляют жидкие кристаллы (ЖК), молекулы которых упорядочены послойно определенным образом между двумя стеклянными пластинами. В каждом слое сигарообразные молекулы ЖК выстраиваются в одном направлении, их оси становятся параллельны (рисунок 6.1).

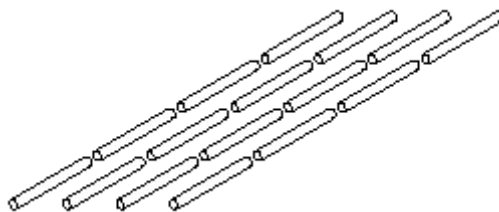


Рисунок 6.1 – Один слой молекул ЖК. Все молекулы параллельны друг другу

Стеклянные пластины имеют специальное покрытие, такое, что направленность молекул в двух крайних слоях перпендикулярна. Ориентация каждого слоя ЖК плавно изменяется от верхнего к нижнему слою, формируя спираль (рисунок 6.2).

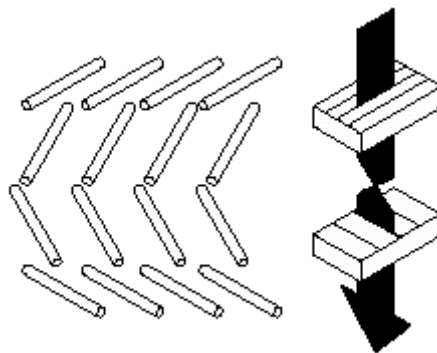


Рисунок 6.2 – Несколько слоев молекул ЖК, упорядоченные так, что поляризованный свет «скручивается», проходя через них. Молекулы в разных слоях выстраиваются по спирали

Эта спираль «скручивает» поляризацию света по мере его прохождения через дисплей. Под действием электрического поля молекулы ЖК переориентируются параллельно полю. Этот процесс называется твист-нематическим полевым эффектом. При такой ориентации поляризация света не скручивается при прохождении через слой ЖК (рисунок 6.3).

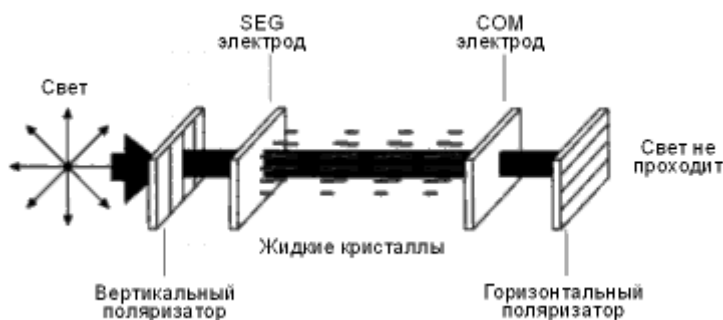


Рисунок 6.3 – «Включенное» состояние. Электрическое поле переориентирует ЖК молекулы так, что они не изменяют поляризацию света. Свет не проходит

Если передний поляризатор ориентирован перпендикулярно заднему, свет пройдет через включенный дисплей, но заблокируется задним поляризатором. В этом случае ЖКИ действует как заслонка свету. Отображение различных символов достигается избирательным травлением проводящей поверхности, предварительно созданной на стекле. Не вытравленные области становятся символами, а вытравленные – фоном дисплея.

Символы создаются из одного или нескольких сегментов. Каждый сегмент может быть адресован (запитан) индивидуально, чтобы создать отдельное электрическое поле. Таким образом, прохождение света управляется электрически, включая и отключая необходимые сегменты. В неактивной части дисплея направленность молекул остается спиральной, формируя фон. Запитанные сегменты составляют символы, контрастирующие с фоном.

В зависимости от ориентации поляризатора, ЖКИ может отображать позитивное или негативное изображение. В дисплее с позитивным изображением передний и задний поляризатор перпендикулярны друг другу, так что незапитанные сегменты и фон пропускают свет с измененной поляризацией, а запи-

танные препятствуют прохождению света. В результате – темные символы на светлом фоне.

В дисплее с негативным изображением поляризаторы параллельны, «в фазе» – препятствуют прохождению света с повернутой поляризацией, так что незапитанные символы и фон темные, а запитанные – светлые.

Рефлективный ЖКИ имеет отражатель (рефлектор) за задним поляризатором, который отражает свет, прошедший через незапитанные сегменты и фон. В негативных рефлективных дисплеях свет отражается через запитанные, «включенные» сегменты. Трансмиссивные дисплеи используют те же принципы, но фон или сегменты становятся ярче за счет использования задней подсветки.

6.2 Жидкокристаллический индикатор *Volymín BC1604A-GPLCW*

Жидкокристаллический модуль BC1604A представляет собой алфавитно-цифровой дисплей. Тип сегментов – матричный. Алфавитно-цифровые (или символьные индикаторы) в своем составе обязательно имеют знакогенератор, который при поступлении соответствующей команды формирует те или иные последовательности микрокоманд, обеспечивающие воспроизведение на экране дисплея выбранных стандартных символов (цифры, буквы и знаки). Кроме того, система команд модуля обеспечивает возможность программирования пользователем до восьми символов в дополнение к стандартным. В состав модуля входит контроллер KS0066 (или совместимый с ним, например, HD44780), который может потенциально управлять двумя строками по 40 символов в каждой (для модулей с четырьмя строками используется два однотипных контроллера).

Условное обозначение индикатора приведено на рисунке 6.4.

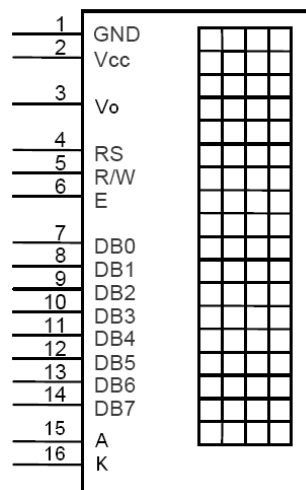


Рисунок 6.4 – Условное обозначение индикатора BC1604A

Назначение выводов индикатора BC1604A-GPLCW приведено в таблице 6.1.

Таблица 6.1 – Назначение выводов индикатора BC1604A-GPLCW

Обозначение вывода	Номер	Назначение
GND	1	Земля
Vcc	2	Питание
Vo	3	Регулировка контраста
RS	4	Выбор регистра («1» – регистр данных, «0» – регистр команд)
R/W	5	«1» – чтение данных, «0» – запись данных
E	6	Сигнал разрешения чтения/записи
DB0–DB7	7–14	0–7 – биты данных
A	15	+ источника питания светодиода задней подсветки
K	16	– источника питания светодиода задней подсветки

Основные параметры устройства:

- количество символов (16 символов, четыре линии);
- размер видимой области экрана (62,0 × 26,0 мм);
- размер символов (5 × 8 точек);
- встроенный контроллер (KS0066 или совместимый с ним);
- напряжение питания – 3 В;
- ток потребления – 1,2 мА.

Упрощенная структурная схема индикатора представлена на рисунке 6.5.

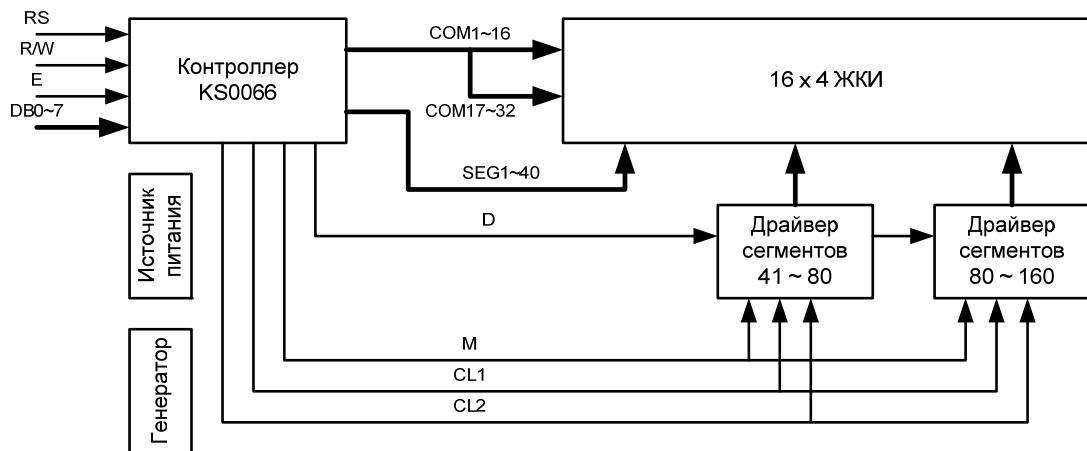


Рисунок 6.5 – Упрощенная структурная схема индикатора

6.3 Варианты индивидуальных заданий к лабораторной работе

Разработать программу для учебного стенда, позволяющую выводить поясняющие надписи и результат вычислений на матричный жидкокристаллический индикатор (МЖКИ). Вариант задания получить у преподавателя.

Более подробные сведения о параметрах и принципах функционирования различных типов индикаторов, а также об использовании жидкокристаллического модуля BC1604A с примером программы, можно найти в методических указаниях по лабораторной работе № 5, электронная версия которых имеется в классе ПЭВМ кафедры (лаб. 207/2) по следующему адресу: D:\методические

рекомендации\СТАНДАРТЫ РБ\Специальность 1-53 01 05 АЭП и АПУ\Микропроцессорные средства в автоматизированном электроприводе\МПС 05.pdf, а также на сервере филиала кафедры по адресу \\Study\Методические указания\МПСвАЭП\МПС 05.pdf.

Контрольные вопросы

- 1 Какова физическая сущность работы жидкокристаллических индикаторов?
- 2 Какие используются режимы отображения жидкокристаллических индикаторов?
- 3 В чем сущность работы рефлективных индикаторов?
- 4 В чем сущность работы трансмиссивных индикаторов?
- 5 В чем сущность работы трансфлективных индикаторов?
- 6 Как влияет температура на рабочие характеристики жидкокристаллических индикаторов?
- 7 Что такое изоконтрастная кривая жидкокристаллических индикаторов?
- 8 Какие типы сегментов используются в жидкокристаллических индикаторах?
- 9 Каково типичное время срабатывания жидкокристаллических индикаторов?
- 10 Какова внутренняя структура жидкокристаллического индикатора ВС1604А?

7 Лабораторная работа № 6. Изучение системы прерываний микропроцессорных устройств

Цель работы: ознакомиться с характеристиками системы прерываний процессора Cortex-M4, а также особенностями организации внешних прерываний микроконтроллеров семейства STM32F4xx; разработать и отладить программу, обеспечивающую реакцию микропроцессорной системы на внешние события.

Перед началом лабораторных занятий студентам необходимо изучить следующие вопросы:

- понятие прерывания и системы прерываний;
- основные характеристики системы прерываний.

7.1 Система прерываний микроконтроллеров на базе Cortex-M4

7.1.1 Общие сведения.

Архитектура исключений, реализованная в процессоре Cortex-M4, поддерживает несколько системных исключений и большое число внешних прерываний. Каждое исключение имеет свой порядковый номер, при этом исключения с номерами от 1-го до 15-го являются системными, а с номерами 16 и более соответствуют входам внешних прерываний. Большинство исключений имеют программируемый приоритет и лишь некоторые – фиксированный приоритет. Кристаллы с процессором Cortex-M4 могут иметь разное количество входов внешних прерываний (от 1-го до 240-го) и разное число уровней приоритета.

Это позволяет разработчикам микросхем конфигурировать исходный код процессора в соответствии со своими нуждами. В модели STM32F407 контроллер прерываний может обрабатывать 82 источника внешних прерываний. Общая таблица исключений контроллера приведена в таблице 7.1.

Таблица 7.1 – Исключения STM32F407

Номер исключения	Тип исключения	Приоритет	Тип приоритета	Описание
1	2	3	4	5
1	Reset	-3 (наивысший)	Фиксированный	Сброс
2	NMI	-2	Фиксированный	Немаскируемое прерывание (вход внешнего немаскируемого прерывания)
3	Hard Fault	-1	Фиксированный	Любой отказ, если соответствующий обработчик не разрешён
4	MemManage Fault	0	Программируемый	Отказ системы управления памятью; нарушение правил доступа, заданных модулем MPU, или обращение по некорректному адресу
5	Bus Fault	1	Программируемый	Отказ шины; происходит при получении интерфейсом шины АНВ сигнала ошибки от ведомого устройства на шине (также называется отказом предвыборки, если ошибка возникла при выборке команды, или отказом данных, если ошибка возникла при обращении к данным)
6	Usage Fault	2	Программируемый	Ошибка в программе или попытка обращения к сопроцессору (процессор Cortex-M4 не поддерживает сопроцессор)
7–10	Зарезервировано	–	–	–
11	SVCall	3	Программируемый	Вызов супервизора
12	Debug monitor	4	Программируемый	Исключение монитора отладки (точки останова, точки наблюдения или внешняя команда отладки)
13	Зарезервировано	–	–	–
14	PendSV	5	Программируемый	Запрос системной службы
15	SYSTICK	6	Программируемый	Системный таймер
16	IRQ0	7	Программируемый	Внешнее прерывание 0
...
97	IRQ81	88	Программируемый	Внешнее прерывание 81

Управление и обработка прерываний производится контроллером приоритетных векторных прерываний NVIC (Nested Vectored Interrupt Controller). Данный блок является частью ядра Cortex-M. При возникновении, некоторого события контроллер прерываний автоматически прерывает выполнение основной программы, и вызывает соответствующую функцию обработки прерываний. После выхода из функции обработчика прерываний программа продолжает выполнение с того места, где произошло прерывание.

Номер исключения, обрабатываемого в данный момент времени, содержится в регистре состояния прерывания IPSR процессора, а также в регистре управления и состояния прерывания ICSR контроллера NVIC (поле VECTACTIVE). При возникновении разрешенного исключения, которое не может быть обработано немедленно (например, в том случае, когда выполняется процедура обработки прерывания с более высоким приоритетом или установлен регистр маскирования прерываний), данное исключение будет отложено (это не касается ряда системных исключений). Откладывание исключения означает, что запрос на его обработку будет храниться в специальном регистре до тех пор, пока не появится возможность обработки данного исключения.

Источниками системных исключений являются периферийные устройства ядра процессора, поэтому они являются общими для всех контроллеров, построенных на базе ядер Cortex-M4. Источники внешних прерываний находятся вне ядра, и конкретный их набор зависит от периферии микроконтроллера. В технической документации на контроллер можно найти полный список доступных источников. Кроме того, в файле `stm32f4xx.h/stm32f4xx_hal.h` описан тип данных (перечисление) `IRQn_Type`, в котором указан полный список поддерживаемых прерываний каждой моделью микроконтроллеров STM32F4xx.

В процессоре Cortex-M4 очередность обработки исключений, а также сама возможность обработки конкретного исключения определяется приоритетом этого исключения. Исключения с более высоким приоритетом (меньшим значением уровня приоритета) могут прерывать обработку исключения, имеющего меньший приоритет (большее значение уровня приоритета), т. е. поддерживаются вложенные исключения/прерывания. Некоторые типы исключений (Reset, NMI и Hard Fault) имеют фиксированные значения приоритета, которые представлены отрицательными числами, отражая тот факт, что данные исключения обладают наивысшим приоритетом среди всех остальных. Прочие исключения имеют программируемые уровни приоритета.

Приоритет каждого источника делится на две части: приоритет группы и субприоритет. Разбиение регистров приоритета исключений, имеющих программируемый приоритет, определяется полем PRIGROUP регистра AIRCR контроллера NVIC.

Старшая часть регистра приоритета (левые биты) определяет приоритет группы, а младшая (правые биты) – субприоритет. Значение уровня приоритета группы определяет возможность возникновения прерывания, если в данный момент уже выполняется обработчик другого прерывания. Значение уровня субприоритета используется только в случае одновременного возникновения нескольких исключений с одинаковыми значениями уровня приоритета группы.

При этом исключение с более высоким субприоритетом (с меньшим значением уровня) будет обработано первым. Максимальное количество приоритетов – 16, т. е. для описания значения приоритета отводится 4 бита. При разбиении приоритетов на группы часть битов отводится под значение приоритета группы, оставшаяся – субприоритет. Поэтому возможна следующая настройка: восемь групп, два субприоритета или четыре группы, четыре субприоритета. Кроме того, необходимо отметить, что группа приоритета влияет только на очередность обработки прерываний. Если процессор выполняет обработчик прерывания, другое прерывание с тем же групповым приоритетом, что и обрабатываемое, не вытесняет обработчик, даже если имеет более высокое значение субприоритета.

При возникновении исключения процессору необходимо определить начальный адрес обработчика этого исключения. Данная информация хранится в таблице векторов, расположенной в памяти процессора. По умолчанию таблица векторов располагается, начиная с нулевого адреса, а адрес каждого вектора в указанной таблице равен порядковому номеру исключения, умноженному на 4.

Таблица векторов представляет собой массив 32-битных значений, расположенных в системной области памяти, каждое из которых является начальным адресом обработчика одного конкретного исключения.

7.1.2 Контроллер внешних прерываний STM32F407.

В состав STM32F407 входит контроллер внешних прерываний EXTI, обеспечивающий:

- независимое детектирование и маску на каждую линию прерывания/события;
- специальный бит состояния для каждой линии прерывания;
- формирование до 23 программных запроса прерываний/событий;
- обнаружение внешних сигналов с длительностью импульса меньше периода частоты шины APB2.

Контроллер внешних прерываний EXTI осуществляет проверку всех входов микроконтроллера на предмет наличия событий (до 140 линий) – образуя 16 линий прерываний (по количеству линий портов).

7.2 Варианты индивидуальных заданий к лабораторной работе

Разработать программу для учебного стенда, позволяющую, используя систему прерываний, обрабатывать события (включение/выключение) на ДД, выводить поясняющие надписи и результат вычислений на МЖКИ или ДСИ. Вариант задания получить у преподавателя.

Более подробные сведения о системе прерываний микроконтроллеров Cortex-M4, а также настройке прерываний в программе, можно найти в методических указаниях по лабораторной работе № 6, электронная версия которых имеется в классе ПЭВМ кафедры (лаб. 207/2) по следующему адресу: D:\методические рекомендации\СТАНДАРТЫ РБ\Специальность 1-53 01 05 АЭП и АПУ\Микропроцессорные средства в автоматизированном электропри-

воде\MПС 06.pdf, а также на сервере филиала кафедры по адресу \\Study\Методические указания\MПСВАЭП\MПС 06.pdf.

Контрольные вопросы

- 1 В чем назначение и каковы функции контроллера прерываний?
- 2 Какие существуют типы прерываний?
- 3 Как осуществляется настройка приоритетов прерываний?
- 4 Какими возможностями обладает контроллер внешних прерываний STM32F407?

8 Лабораторная работа № 7. Изучение организации таймеров в микропроцессорных устройствах

Цель работы: изучить особенности программной и аппаратной реализации временных функций, режимы работы и порядок программирования таймеров микроконтроллера, реализацию временных функций с помощью языка программирования С; составить программу заданной временной функции, перевести ее в машинные коды, записать в память программ микроконтроллера и выполнить.

Перед началом лабораторных занятий студентам необходимо изучить следующие вопросы:

- применение таймеров-счетчиков в микропроцессорных системах (МПС);
- система прерываний микроконтроллеров STM32F4xx.

8.1 Реализация функций времени

Микропроцессорные системы управления являются системами, работающими в режиме реального времени, подразумевающим формирование сигналов управления в определенные моменты времени, реакцию на появляющиеся события. Для реализации временных функций в микропроцессорных системах управления используют следующие способы: программные, аппаратные, программно-аппаратные.

Любая команда микропроцессором выполняется за определенное время. При программном способе разрабатывается программа, набор команд в которой подобран таким образом, чтобы программа выполнялась в течение заданного промежутка времени.

При аппаратном способе для формирования задержек времени используются аппаратные элементы (таймеры, счетчики) и система прерываний микропроцессора, позволяющие во время выполнения программы осуществлять выдержку времени, а по ее окончании выполнить необходимые действия в подпрограмме обработки прерываний.

Программно-аппаратный способ основывается на использовании аппаратных элементов для выдержки времени и программном контроле их работы. По-

сле окончания выдержки времени аппаратные элементы устанавливают аппаратные флаги, опрашивая которые процессор может судить об окончании процесса.

8.1.1 Программное формирование временной задержки.

Процедура реализации временной задержки использует метод программных циклов. При этом в некоторый рабочий регистр загружается число, которое затем в каждом проходе цикла уменьшается на 1. Так продолжается до тех пор, пока содержимое рабочего регистра не станет равным нулю, что интерпретируется программой как момент выхода из цикла. Время задержки при этом определяется числом, загруженным в рабочий регистр, и временем выполнения команд, образующих программный цикл.

Поскольку в программе может потребоваться многократно формировать временные задержки фрагмент программы, реализующей задержку, целесообразно оформить в виде подпрограммы. Например:

```

delay
  LDR R0,=X
loop
  SUBS R0,R0,#1
  BNE loop
  BX LR

```

Для получения требуемой временной задержки необходимо определить число X , загружаемое в рабочий регистр. Определение числа X выполняется на основе расчёта времени выполнения команд, образующих данную подпрограмму. При этом необходимо учитывать, что команды LDR и BX LR выполняются однократно, а число повторений команд SUBS и BNE равно числу X . Кроме того, обращение к подпрограмме временной задержки осуществляется по команде BL delay, время выполнения которой также необходимо учитывать при подсчете временной задержки. В описании системы команд микроконтроллера указывается, сколько машинных циклов (МЦ) исполняет каждая команда. Особенностью контроллеров ARM является наличие конвейера на три команды, который необходимо обновлять при выполнении команд перехода (B, BX, BNE и т. д.). На основании этих данных определяется суммарное число машинных циклов в подпрограмме: «BL delay» – 3 МЦ, LDR – 2 МЦ, SUBS – 1 МЦ, BNE – 2 МЦ, «BX LR» – 3 МЦ.

Таким образом, задержка времени, осуществляемая данной подпрограммой, может быть рассчитана следующим образом:

$$T = (3 + 2 + X \cdot (1 + 2) + 3) \cdot T_{\text{ц}} = (8 + X \cdot 3) \cdot T_{\text{ц}}, \quad (8.1)$$

где $T_{\text{ц}}$ – длительность машинного цикла процессора.

Длительность машинного цикла процессоров ARM зависит от частоты основной шины АНВ1. Для контроллеров STM32F407 максимальное значение

шины составляет 168 МГц, соответственно, длительность машинного цикла равна $1/168 \text{ МГц} \approx 5,95 \text{ нс}$.

Недостатком такого способа выдержек времени является загрузка процессора бесполезной работой – вместо выполнения действий по обработке или обмену информацией, он занят выдержкой времени.

Для устранения этого недостатка необходимо использовать аппаратные средства микропроцессорной системы.

8.2 Варианты индивидуальных заданий к лабораторной работе

Разработать программу для учебного стенда, позволяющую выполнять действия в соответствии с заданием. Вариант задания получить у преподавателя.

Более подробные сведения о возможностях и режимах работы таймеров STM32F407, а также настройке и использовании их в программе можно найти в методических указаниях по лабораторной работе № 7, электронная версия которых имеется в классе ПЭВМ кафедры (лаб. 207/2) по следующему адресу: D:\методические рекомендации\СТАНДАРТЫ РБ\Специальность 1-53 01 05 АЭП и АПУ\Микропроцессорные средства в автоматизированном электроприводе\МПС 07.pdf, а также на сервере филиала кафедры по адресу \\Study\Методические указания\МПС в АЭП\МПС 07.pdf.

Контрольные вопросы

- 1 Как программно формируются задержки малой длительности?
- 2 Как программно формируются задержки большой длительности?
- 3 Использование таймеров для получения задержки.
- 4 Как определяется частота импульсного сигнала?
- 5 Как определяется длительность импульса?
- 6 Как можно использовать таймеры/счетчики для определения длительности импульсов?
- 7 Как можно использовать таймеры/счетчики для определения частоты сигналов?

9 Лабораторная работа № 8. Изучение организации семисегментной индикации в микропроцессорных устройствах

Цель работы: ознакомиться с организацией семисегментных индикаторов; изучить схему сопряжения микроконтроллера с устройством динамической индикации; разработать и отладить программу вывода информации на индикатор.

Перед началом лабораторных занятий студентам необходимо изучить следующие вопросы:

- способы построения цифровой индикации;
- устройства динамической индикации;
- устройства статической индикации.

9.1 Устройства цифровой индикации

Цифровой индикатор – устройство, предназначенное для отображения значения числовой величины в цифровом виде. В микропроцессорных системах для вывода числовой информации наибольшим распространением пользуются светодиодные семисегментные индикаторы (рисунок 9.1). Это наиболее простая реализация индикатора, который может отображать арабские цифры. Для отображения букв используются более сложные многосегментные и матричные индикаторы.

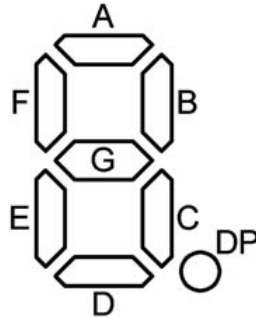


Рисунок 9.1 – Семисегментный индикатор

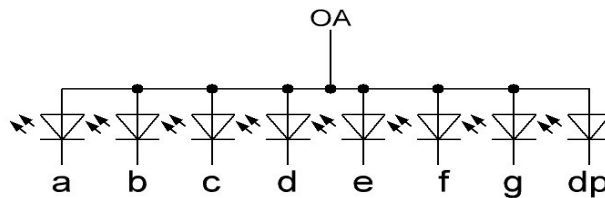
Сегменты индикатора расположены в виде восьмерки. Иногда добавляют еще один восьмой сегмент – десятичную запятую. Засвечивая группы сегментов, можно получить все цифры и некоторые символы.

Конструктивно индикаторы оформляются в виде светодиодных модулей с общим катодом или с общим анодом (рисунок 9.2).

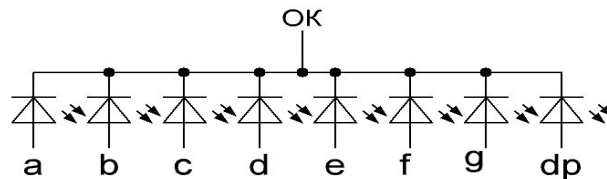
Для того чтобы такой индикатор отобразил число, необходимо на общий вывод подключить один из выводов источника питания и управлять включением соответствующих сегментов индикатора, чтобы сформировать на нем соответствующее изображение.

Однако такой индикатор позволяет отобразить только одну цифру. Для того чтобы сформировать многозначное число, необходимо в схему включить несколько таких индикаторов, и подключить их к устройству управления. При построении систем отображения информации различают два подхода: статическая и динамическая индикация.

а)



б)



а – с общим анодом; б – с общим катодом

Рисунок 9.2 – Электрические принципиальные схемы семисегментных индикаторов

Статическая индикация состоит в постоянной засветке каждого индикатора от одного источника информации. В этом случае каждый индикатор подключен к микропроцессору через отдельный дешифратор и регистр-защелку. Каждый регистр адресуется с помощью устройства дешифрации адреса. Такая схема предполагает значительные аппаратные затраты, т. к. на каждый индикатор необходим по крайней мере один регистр.

Сущность динамической индикации заключается в поочередном циклическом подключении микропроцессором каждого индикатора к общему источнику данных. При использовании такого способа индикации значительно сокращаются аппаратные затраты. Но при этом необходимо обеспечить достаточное время свечения, для того, чтобы не уменьшалась яркость свечения индикаторов. С другой стороны, необходимо обеспечить достаточно быстрое переключение индикаторов, чтобы не было заметно мерцание.

9.2 Варианты индивидуальных заданий к лабораторной работе

Разработать программу для учебного стенда, позволяющую выводить результат вычислений на светодиодный символьный индикатор (ССИ). Вариант задания получить у преподавателя.

Более подробные сведения об организации семисегментной индикации, используемой в стенде устройстве индикации, а также пример программирования контроллера для работы с таким индикатором, можно найти в методических указаниях по лабораторной работе № 8, электронная версия которых имеется в классе ПЭВМ кафедры (лаб. 207/2) по следующему адресу: D:\методические рекомендации\СТАНДАРТЫ РБ\Специальность 1-53 01 05 АЭП и АПУ\Микропроцессорные средства в автоматизированном электроприводе\МПС 08.pdf, а также на сервере филиала кафедры по адресу \\Study\Методические указания\МПСвАЭП\МПС 08.pdf.

Контрольные вопросы

- 1 Как организуется вывод цифровой информации в микроконтроллерных системах?
- 2 В чем принцип действия семисегментных светодиодных индикаторов?
- 3 Как организуется статическая индикация?
- 4 Как организуется динамическая индикация?
- 5 Особенности программной реализации динамической индикации.

10 Лабораторная работа № 9. Изучение организации клавиатуры в микропроцессорных устройствах

Цель работы: ознакомиться со схемами подключения клавиатуры к микроконтроллеру; изучить схему сопряжения микроконтроллера с клавиатурой; разработать и отладить программу обработки событий клавиатуры.

Перед началом лабораторных занятий студентам необходимо изучить следующие вопросы:

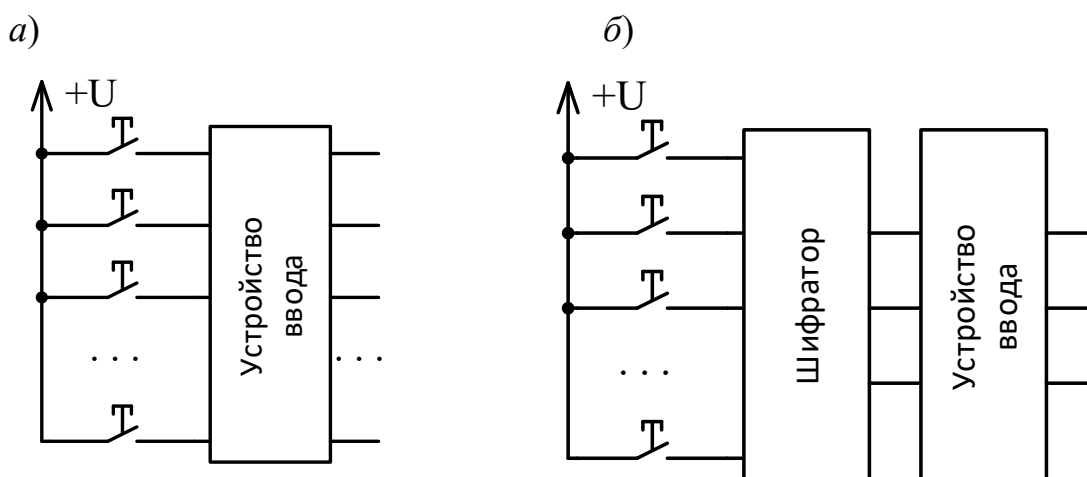
- назначение и варианты организации клавиатур в МПС;
- параллельные порты контроллеров STM32.

10.1 Общие сведения о клавиатурах

Клавиатуры предназначены для ввода информации в микропроцессорное устройство, т. е. для организации интерфейса с человеком. Они состоят из двух основных частей: набора клавиш (дискретных переключателей), соответствующих набору знаков алфавита и функций редактирования, и устройств кодирования для преобразования сигнала нажатой клавиши в параллельный код обмена информацией. При этом за каждым переключателем закрепляется определенный знак или функция. Особенностью клавиатур является достаточно большое количество таких переключателей (обычно более 10).

Для кодирования нажатой клавиши используется два способа адресации – линейный и двухкоординатный.

Самым простым способом организации клавиатур является линейный (рисунок 10.1, а). При таком построении клавиатуры, сигнал от каждой кнопки вводится в систему через устройства ввода информации (порт ввода/вывод) по отдельной линии.



а – с программной шифрацией; б – с аппаратной шифрацией

Рисунок 10.1 – Линейная организация клавиатур

За одно считывание в такой схеме микропроцессор определяет состояние всех переключателей. При этом определение кода нажатой клавиши необходимо выполнять программным способом.

Преимуществом такого метода является простота схемной реализации и обработки получаемой информации.

Недостаток такого способа заключается в том, что большое количество переключателей требует использования большого количества линий параллельных портов, поэтому применяется при относительно небольшом числе переключателей. Уменьшить количество сигналов, вводимых в систему, можно за счет включения в схему шифратора (см. рисунок 10.1, б), который в зависимости от того, какая из кнопок будет нажата (т. е. выработает активный уровень сигнала на входе шифратора), сформирует на своем выходе двоичный код нажатой клавиши. Однако при большом количестве кнопок построение шифратора значительно усложняет аппаратную часть микропроцессорной системы.

Одним из наиболее распространенных способов уменьшения требуемых линий для подключения клавиатур к параллельным портам является организация клавиатур по принципу матричного шифратора, в узлах которого размещены коммутационные элементы – клавиши (рисунок 10.2).

Из показанного на рисунке примера видно, что 16-клавишную клавиатуру можно подключить, используя только восемь линий параллельного порта. При этом часть линий, например, горизонтальные R1–R4 (ряды или строки) являются выходными, а вертикальные C1–C4 (колонки) – входными.

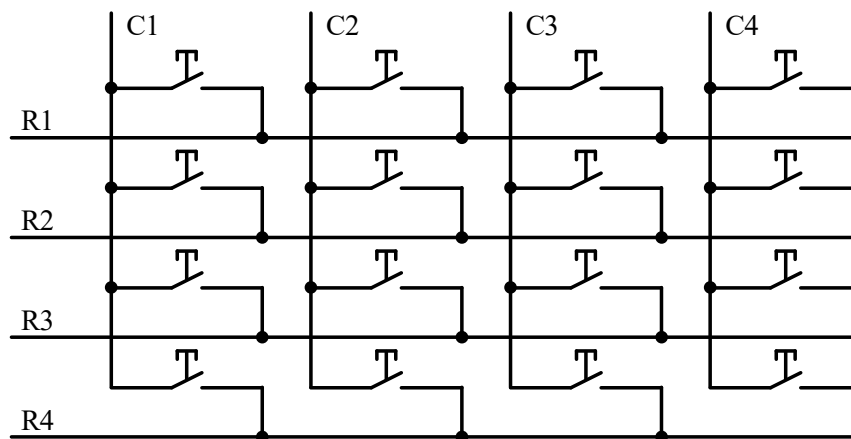


Рисунок 10.2 – Матричная организация клавиатур

Опрос клавиатуры заключается в постоянном сканировании клавиш с целью определения кода нажатой кнопки. Работа с ней производится следующим образом. На горизонтальных линиях (при вышеуказанном допущении о направлении передачи по линиям) формируется «бегущий» сигнал, например, логическая единица. На остальных линиях в этом случае сохраняется логический нуль. Для каждой фазы «бегущей» единицы производится считывание и анализ кода на линиях C1–C4. Если ни одна клавиша не нажата, то сигнал на линиях C1–C4 отсутствует. Если клавиша в сканируемом ряду нажата, то на соответствующей линии C1–C4 появляется поданный на линию Ri сигнал. Таким обра-

зом, зная какой ряд в данный момент сканируется и на какой из линий С1–С4 появился сигнал, можно определить, какая клавиша нажата в данный момент.

Код сканирования получают шифрацией унитарного кода опроса и номеров битов активных выходных линий.

10.2 Варианты индивидуальных заданий к лабораторной работе

Разработать программу для учебного стенда, позволяющую вводить данные с клавиатуры, выполнять определенные действия над данными и выводить поясняющие надписи и результат вычислений на матричный жидкокристаллический индикатор или на светодиодный семисегментный индикатор. Клавишу «#» использовать, как подтверждение ввода, а клавишу «*» – как сброс операции. Вариант задания получить у преподавателя.

Дополнительную информацию с примером программирования контроллера для работы с клавиатурой можно найти в методических указаниях по лабораторной работе № 9, электронная версия которых имеется в классе ПЭВМ кафедры (лаб. 207/2) по следующему адресу: D:\методические рекомендации\СТАНДАРТЫ РБ\Специальность 1-53 01 05 АЭП и АПУ\Микропроцессорные средства в автоматизированном электроприводе\МПС 09.pdf, а также на сервере филиала кафедры по адресу \\Study\Методические указания\МПСвАЭП\МПС 09.pdf.

Контрольные вопросы

- 1 Для чего используются клавиатуры?
- 2 По какому принципу чаще всего организуются клавиатуры?
- 3 Как работает матричный клавиатурный шифратор?

11 Лабораторная работа № 10. Изучение организации устройств ввода аналоговой информации в микропроцессорных устройствах

Цель работы: ознакомиться с техническими характеристиками и принципом действия АЦП, изучить схему ввода аналоговых сигналов в микроконтроллер; разработать и отладить программу для сбора аналоговой информации.

11.1 Общие сведения об аналогово-цифровых преобразователях

Аналогово-цифровой преобразователь (АЦП) – один из самых важных электронных компонентов в измерительном и тестовом оборудовании. АЦП преобразует напряжение (аналоговый сигнал) в код, над которым микропроцессор и программное обеспечение выполняют определенные действия. Даже если вы работаете только с цифровыми сигналами, скорее всего используете АЦП в составе осциллографа, чтобы узнать их аналоговые характеристики.

Существует несколько основных типов архитектуры АЦП, отличающихся

по сложности реализации, быстродействию, помехоустойчивости, хотя в пределах каждого типа существует также множество вариаций. Различные типы измерительного оборудования используют различные типы АЦП. Например, в цифровом осциллографе используется высокая частота дискретизации, но не требуется высокое разрешение. В цифровых мультиметрах нужно большее разрешение, но можно пожертвовать скоростью измерения.

Системы сбора данных общего назначения по скорости дискретизации и разрешающей способности обычно занимают место между осциллографами и цифровыми мультиметрами. В оборудовании такого типа используются АЦП последовательного приближения либо сигма-дельта АЦП. В системах, где основным критерием является быстродействие, применяют АЦП параллельного преобразования. Однако преобразователи этого типа сложны в реализации. Для n -разрядного АЦП необходимо $2n-1$ компараторов и параллельный делитель напряжения, который вырабатывает $2n-1$ уровней квантования. Кроме того, существуют также интегрирующие АЦП с высоким разрешением и помехоподавлением. Такие АЦП состоят из двух преобразователей. Измеряемое напряжение преобразуется в длительность импульса, а потом длительность импульса преобразуется в цифровой код.

11.2 Встроенный в микроконтроллер STM32F407 аналогово-цифровой преобразователь

В микроконтроллере STM32F407 встроены три 12-битных АЦП, каждый из которых может выполнять 2,4 млн выборок в секунду, а в режиме попарной работы достигать 7,2 млн выборок в секунду. Общая структура подсистемы ввода аналоговых сигналов в контроллере приведена на рисунке 11.1.

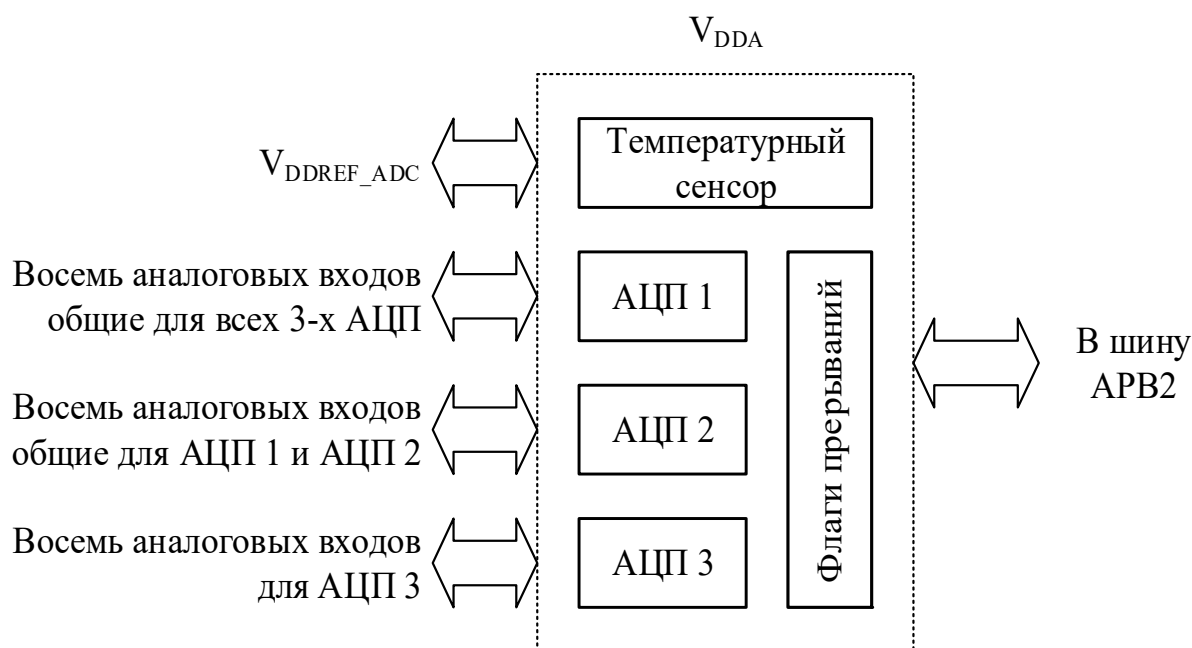


Рисунок 11.1 – Блок ввода аналоговых сигналов STM32F407

Как видно, блок содержит в своем составе три АЦП и температурный сенсор. В качестве источника питания используется напряжение VDDA. Ввод аналоговых сигналов осуществляется через 24 входа, которые распределены между АЦП следующим образом: восемь каналов являются общими для всех трех АЦП, восемь каналов являются общими для АЦП 1 и АЦП 2 и восемь каналов подключены к АЦП 3.

Каждый АЦП представляет собой мощный модуль, который имеет хорошие характеристики и интересные особенности:

- 19 каналов ввода (16 внешних, два внутренних и канал VBAT);
- разрешение 12 бит, однако может быть настроено 12, 10, 8 или 6 бит;
- большое количество режимов преобразования: однократное, непрерывное, прерываемое (для быстрого преобразования группы каналов), сканирование (автоматическое преобразование каналов от 0 до N), по триггеру, по таймеру;
- удобное выравнивание результата преобразования;
- наличие аналогового наблюдателя (Analog Watchdog), позволяющего приложению обнаруживать момент, когда входное напряжение выходит за заданные пользователем пределы;
- генерирование запроса прерывания в конце преобразования, в конце инжектированного преобразования, а также в случае наступления событий Analog Watchdog или Overrun (переполнение);
- Dual/Triple режимы, обеспечивающие синхронную работу нескольких АЦП (двух-трех);
- возможность автокалибровки;
- возможность генерирования запроса DMA во время преобразования обычных каналов;
- напряжение питания от 2,4 до 3,6 В при максимальной частоте преобразования, с возможностью питания от 1,8 В при низких частотах: $VDDA = 1,8 \dots 3,6 \text{ В}$;
- диапазон входных напряжений ($VREF_- \leq VIN \leq VREF_+$);
- в качестве источника опорного напряжения может использоваться внутренний источник $VREFINT = 1,18 \dots 1,24 \text{ В}$, или внешний источник $VDDREF_ADC = 1,8 \dots VDDA$;
- погрешность преобразования от двух до пяти младших значащих разрядов (МЗР);
- программируемое время преобразования от 0,3 до 16,4 мкс;
- ток потребления в режиме преобразования до 1,8 мА.

Контрольные вопросы

- 1 Какие схемы АЦП применяют в быстродействующих системах?
- 2 Какие схемы АЦП применяют при повышении помехоустойчивости?
- 3 Как работают АЦП, построенные по схеме сравнения?
- 4 Из каких структурных элементов состоит АЦП, встроенный в микроконтроллер STM32F407?
- 5 Каковы основные параметры АЦП, встроенного в микроконтроллер STM32F407?

6 В каких режимах может работать АЦП, встроенный в микроконтроллер STM32F407?

7 Каким образом можно изменить режим работы АЦП?

8 На какие группы делятся каналы АЦП и чем они отличаются?

12 Лабораторная работа № 11. Изучение организации устройств вывода аналоговой информации в микропроцессорных устройствах

Цель работы: ознакомиться с техническими характеристиками и принципом действия ЦАП; изучить схему вывода аналоговых сигналов в микропроцессорной системе; разработать и отладить программу для вывода аналоговой информации.

Перед началом лабораторных занятий студентам необходимо изучить следующие вопросы:

- основные типы и принцип действия ЦАП;
- основные технические характеристики ЦАП.

12.1 Общие сведения о ЦАП

Цифроаналоговый преобразователь (ЦАП) – это устройство для преобразования цифрового кода в аналоговый сигнал по величине, пропорциональной значению кода.

ЦАП применяются для связи цифровых управляющих систем с устройствами, которые управляются уровнем аналогового сигнала. Также, ЦАП является составной частью во многих структурах аналогово-цифровых устройств и преобразователей.

ЦАП характеризуется функцией преобразования. Она связывает изменение цифрового кода с изменением напряжения или тока. Функция преобразования ЦАП выражается следующим образом:

$$U_{\text{ВЫХ}} = \frac{U_{\text{max}}}{N_{\text{max}}} N_{\text{ВХ}}, \quad (12.1)$$

где $U_{\text{ВЫХ}}$ – значение выходного напряжения, соответствующее цифровому коду $N_{\text{ВХ}}$, подаваемому на входы ЦАП;

U_{max} – максимальное выходное напряжение, соответствующее подаче на входы максимального кода N_{max} .

Принцип работы большинства ЦАП – это суммирование долей аналоговых сигналов (веса разряда), в зависимости от входного кода.

ЦАП можно реализовать с помощью суммирования токов, суммирования напряжений и деления напряжений. В первом и втором случае в соответствии

со значениями разрядов входного кода, суммируются сигналы генераторов токов и источников ЭДС. Последний способ представляет собой управляемый кодом делитель напряжения. Два последних способа не нашли широкого распространения в связи с практическими трудностями их реализации.

Кроме того, ЦАП можно классифицировать и по другим признакам:

- по роду выходного сигнала (с токовым выходом, выходом по напряжению, с изменяющимся сопротивлением);

- полярности выходного напряжения (постоянному, переменному) и т. д.

Основные структуры, используемые в ЦАП интегрального исполнения:

- со взвешенными резисторами в цепях эмиттеров;

- со взвешенными резисторами в цепях нагрузки;

- с лестничной матрицей $R-2R$ в цепях эмиттеров транзисторов источников тока;

- с выходной лестничной матрицей $R-2R$.

Структура простейшего ЦАП с взвешенным суммированием токов приведена на рисунке 12.1.

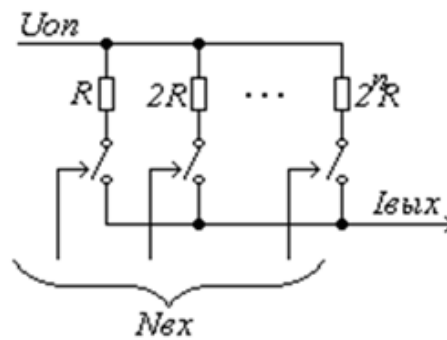


Рисунок 12.1 – Упрощенная схема ЦАП

12.2 Встроенный в микроконтроллер STM32F407 цифроаналоговый преобразователь

Для организации выдачи аналоговых сигналов в цифровых системах используются цифроаналоговые преобразователи. В состав микроконтроллера STM32F407 входит модуль ЦАП, представляющий собой 12-битный цифроаналоговый преобразователь с напряжением на выходе. ЦАП могут быть сконфигурированы в 8- или 12-битном режиме и могут использоваться вместе с контроллером прямого доступа в память (ПДП). В 12-битном режиме данные могут быть выровнены по левому или по правому краю регистра. Модуль ЦАП имеет два выходных канала, каждый из которых управляется своим преобразователем. В «сдвоенном» режиме преобразования могут быть выполнены независимо или одновременно, когда оба канала сгруппированы вместе для синхронного обновления выхода. В качестве опорного напряжения может использоваться сигнал на входе V_{REF+} (используемый совместно с АЦП).

Основные особенности модуля ЦАП:

- два преобразователя (у каждого один выходной канал);
- правое или левое выравнивание данных в 12-битном режиме;
- возможность синхронного обновления;
- генератор шума;
- генератор треугольного сигнала;
- режим «сдвоенных» каналов для независимых или одновременных преобразований;
- возможность ПДП для каждого канала;
- обнаружение ошибок опустошения ПДП;
- возможность запуска преобразования от внешнего сигнала;
- вход опорного напряжения V_{REF+} ;
- время установления сигнала на выходе не превышает 6 мкс;
- максимальная скорость преобразований может достигать 1 млн преобразований в секунду.

12.3 Варианты индивидуальных заданий к лабораторной работе

Разработать программу для учебного стенда, позволяющую вводить данные с дискретных датчиков (ДД1–ДД8), выполнять определенные действия над данными и выводить аналоговый сигнал с помощью цифроаналогового преобразователя. Вариант задания получить у преподавателя.

Более подробные сведения о возможностях ЦАП с примером его использования можно найти в методических указаниях по лабораторной работе № 11, электронная версия которых имеется в классе ПЭВМ кафедры (лаб. 207/2) по следующему адресу: D:\методические рекомендации\СТАНДАРТЫ РБ\Специальность 1-53 01 05 АЭП и АПУ\Микропроцессорные средства в автоматизированном электроприводе\МПС 11.pdf, а также на сервере филиала кафедры по адресу: \\Study\Методические указания\МПСвАЭП\МПС 11.pdf.

Контрольные вопросы

- 1 Какую функцию выполняют цифроаналоговые преобразователи?
- 2 Какие методы построения ЦАП применяют?
- 3 Как классифицируются ЦАП?
- 4 При помощи каких структур реализуются ЦАП?
- 5 Назовите основные особенности ЦАП микроконтроллера STM32F407.
- 6 Из каких структурных элементов состоит ЦАП микроконтроллера STM32F407?
- 7 Как осуществляется управление ЦАП микроконтроллера STM32F407?

13 Лабораторная работа № 12. Изучение организации последовательных портов в микропроцессорных устройствах

Цель работы: изучить структуру и особенности работы последовательного интерфейса в различных режимах, программу передачи и приема информации по последовательному каналу на языке программирования микроконтроллера; составить программу передачи или приема информации по последовательному каналу в заданном режиме, используя язык программирования микроконтроллера, перевести ее в машинные коды, записать в память программ микроконтроллера, соединив приемник и передатчик, проверить работу интерфейса.

Перед началом лабораторных занятий студентам необходимо изучить следующие вопросы:

- основные виды последовательных интерфейсов;
- основные характеристики последовательных интерфейсов.

13.1 Интерфейс RS-232-C

Интерфейс RS-232-C разработан ассоциацией электронной промышленности (Electronic Industries Association – EIA) как стандарт для соединения компьютеров и различных последовательных периферийных устройств. Широко известен как последовательный порт персональных компьютеров. Исторически имел широкое распространение в телекоммуникационном оборудовании. В настоящее время всё ещё используется для подключения всевозможного специального или устаревшего оборудования к компьютерам, однако в основном уже вытеснен интерфейсом USB.

13.1.1 Основные понятия и термины.

Последовательная передача данных означает, что данные передаются по единственной линии. При этом биты байта данных передаются по очереди с использованием одного провода. Для синхронизации группе битов данных обычно предшествует специальный стартовый бит, после группы битов следуют бит проверки на четность и один или два стоповых бита. Иногда бит проверки на четность может отсутствовать (рисунок 13.1).



Рисунок 13.1 – Формат передачи данных интерфейса RS-232

Из рисунка видно, что исходное состояние линии последовательной передачи данных – уровень логической единицы. Это состояние линии называют отмеченным – MARK. Когда начинается передача данных, уровень линии переходит в 0. Это состояние линии называют пустым – SPACE. Если линия находится в таком состоянии больше определенного времени, считается, что линия перешла в состояние разрыва связи – BREAK.

Стартовый бит START сигнализирует о начале передачи данных. Далее передаются биты данных, вначале младшие, затем старшие.

Если используется бит четности P, то передается и он. Бит четности имеет такое значение, чтобы в пакете битов общее количество единиц (или нулей) было четно или нечетно, в зависимости от установки регистров порта. Этот бит служит для обнаружения ошибок, которые могут возникнуть при передаче данных из-за помех на линии. Приемное устройство заново вычисляет четность данных и сравнивает результат с принятым битом четности. Если четность не совпала, то считается, что данные переданы с ошибкой.

В самом конце передаются один или два стоповых бита STOP, завершающих передачу байта. Затем до прихода следующего стартового бита линия снова переходит в состояние MARK.

Использование бита четности, стартовых и стоповых битов определяют формат передачи данных. Очевидно, что передатчик и приемник должны использовать один и тот же формат данных, иначе обмен будет невозможен.

Другая важная характеристика – скорость передачи данных. Она также должна быть одинаковой для передатчика и приемника.

Скорость передачи данных обычно измеряется в бодах (по фамилии французского изобретателя телеграфного аппарата Emile Baudot – Э. Бодо). Боды определяют количество передаваемых битов в секунду. При этом учитываются и старт/стопные биты, а также бит четности.

Иногда используется другой термин – биты в секунду (bps). Здесь имеется в виду эффективная скорость передачи данных, без учета служебных битов.

Очень важно, чтобы тактовые частоты приемника и передатчика были одинаковыми, допустимое расхождение – не более 10 %. Скорость передачи по RS-232C может выбираться из ряда: 110, 150, 300, 600, 1200, 2400, 4800, 9600, 19200, 38400, 57600 и 115200 бит/с.

Интерфейс RS-232-C определяет обмен между устройствами двух типов: DTE (Data Terminal Equipment – терминальное устройство) и DCE (Data Communication Equipment – устройство связи). В большинстве случаев, но не всегда, компьютер является терминальным устройством. Модемы, принтеры, графопостроители всегда являются устройствами связи.

В компьютерах типа IBM PC – внешний последовательный порт – оформлен в виде разъемов типа DB25 или DB9 (таблица 13.1).

Таблица 13.1 – Разводка разъема последовательной передачи данных DB25

Номер контакта DB25	Номер контакта DB9	Назначение контакта	Вход или выход компьютера
1	–	Защитное заземление (Frame Ground, FG)	–
2	3	Передаваемые данные (Transmitted Data, TD)	Выход
3	2	Принимаемые данные (Received Data, RD)	Вход
4	7	Запрос для передачи (Request to send, RTS)	Выход
5	8	Сброс для передачи (Clear to Send, CTS)	Вход
6	6	Готовность данных (Data Set Ready, DSR)	Вход
7	5	Сигнальное заземление (Signal Ground, SG)	–
8	1	Детектор принимаемого с линии сигнала (Data Carrier Detect, DCD)	Вход
9–19	–	Не используются	–
20	4	Готовность выходных данных (Data Terminal Ready, DTR)	Выход
21	–	Не используется	–
22	9	Индикатор вызова (Ring Indicator, RI)	Вход
23–25	–	Не используются	–

13.1.2 Сигналы интерфейса RS-232-C.

Входы TD и RD используются устройствами DTE и DCE по-разному. Устройство DTE использует вход TD для передачи данных, а вход RD для приема данных. И, наоборот, устройство DCE использует вход TD для приема, а вход RD для передачи данных.

Поэтому для соединения терминального устройства и устройства связи выводы их разъемов необходимо соединить напрямую. Соединение устройства DTE и DCE показано на рисунке 13.2.

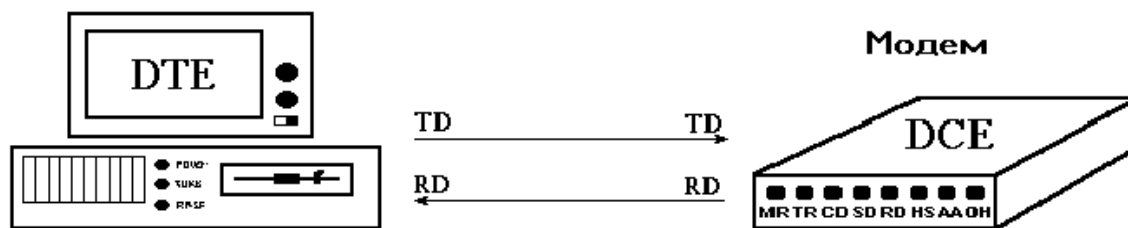


Рисунок 13.2 – Соединение устройства DTE и DCE

Рассмотрим процесс подтверждения связи между компьютером и модемом. В начале сеанса связи компьютер должен удостовериться, что модем может произвести вызов (находится в рабочем состоянии). Затем, после вызова абонента, модем должен сообщить компьютеру, что он произвел соединение с удаленной системой. Подробнее это происходит следующим образом.

Компьютер подает сигнал по линии DTR, чтобы показать модему, что он готов к проведению сеанса связи. В ответ модем подает сигнал по линии DSR. Когда модем произвел соединение с другим, удаленным модемом, он подает сигнал по линии DCD, чтобы сообщить об этом компьютеру.

Если напряжение на линии DTR падает, это сообщает модему, что компьютер не может далее продолжать сеанс связи, например, из-за того, что выключено питание компьютера. В этом случае модем прервет связь. Если напряжение на линии DCD падает, это сообщает компьютеру, что модем потерял связь и не может больше продолжать соединение. В обоих случаях эти сигналы дают ответ на наличие связи между модемом и компьютером.

Так, мы рассмотрели самый низкий уровень управлением связи – подтверждение связи. Существует более высокий уровень, который используется для управления скоростью обмена данными, но он также реализуется аппаратно. Практически управление скоростью обмена данными (управление потоком) необходимо, если производится передача больших объемов данных с высокой скоростью. Когда одна система пытается передать данные с большей скоростью, чем они могут быть обработаны принимающей системой, результатом может стать потеря части передаваемых данных. Чтобы предотвратить передачу большего числа данных, чем то, которое может быть обработано, используют управление связью, называемое «управление потоком» (flow-controll handshake).

Стандарт RS-232-C определяет возможность управления потоком только для полудуплексного соединения. Полудуплексным называется соединение, при котором в каждый момент времени данные могут передаваться только в одну сторону.

Однако фактически этот механизм используется и для дуплексных соединений, когда данные передаются по линии связи одновременно в двух направлениях.

13.1.3 Управление потоком.

В полудуплексных соединениях устройство DTE подает сигнал RTS, когда оно желает передать данные. DCE отвечает сигналом по линии CTS, когда оно готово, и DTE начинает передачу данных. До тех пор, пока оба сигнала RTS и CTS не примут активное состояние, только DCE может передавать данные.

При дуплексных соединениях сигналы RTS/CTS имеют противоположные значения по сравнению с теми, которые они имели для полудуплексных соединений.

Когда DTE может принять данные, он подает сигнал по линии RTS. Если при этом DCE готово для принятия данных, оно возвращает сигнал CTS. Если напряжение на линиях RTS или CTS падает, то это сообщает передающей системе, что получающая система не готова для приема данных.

13.1.4 Технические параметры интерфейса RS-232-C.

При передаче данных на большие расстояния без использования специальной аппаратуры из-за помех, наводимых электромагнитными полями, возможно возникновение ошибок. Вследствие этого накладываются ограничения на длину соединительного кабеля между устройствами DTR-DTR и DTR-DCE.

Официальное ограничение по длине для соединительного кабеля по стандарту RS-232-C составляет 15,24 м.

Логический ноль (SPACE) представляется положительным напряжением в диапазоне от +3 до +25 В, а логическая единица (MARK) – отрицательным

напряжением в диапазоне от -3 до -25 В. Промежуток от -3 до $+3$ В соответствует неопределенному значению (рисунок 13.3).



Рисунок 13.3 – Уровни сигналов

Если вы подключаете внешние устройства к разъему интерфейса RS-232-C (а также при соединении двух компьютеров нуль-модемом), предварительно выключите его и компьютер, а также снимите статический заряд (подсоединив заземление). В противном случае можно вывести из строя асинхронный адаптер. Земля компьютера и земля внешнего устройства должны быть соединены вместе.

13.2 Последовательный порт микроконтроллеров STM32F407

13.2.1 Общие сведения.

Наряду с портами ввода-вывода GPIO универсальный синхронно-асинхронный последовательный приёмопередатчик USART (Universal Synchronous Asynchronous Receiver Transmitter) является одним из самых востребованных портов для связи любого микроконтроллера с внешними устройствами. С помощью порта USART можно легко организовать связь микроконтроллера с компьютером, провести его программирование, а также связать микроконтроллеры между собой по интерфейсам RS-232, RS-485, RS-422 и т. д.

Преимуществами порта USART являются простота в использовании, а также, в отличие от интерфейса USB, гибкость в настройке и надёжность работы.

Микроконтроллеры STM32 имеют обычно несколько портов USART. Например, STM32F407 имеет четыре синхронно-асинхронных последовательных приёмопередатчиков USART (USART1, USART2, USART3 и USART6) и два асинхронных последовательных приёмопередатчика UART (UART4 и UART5).

Кроме обычного асинхронного режима работы с использованием сигналов RXD и TXD, порты USART STM32 поддерживают несколько расширенных режимов работы. В отличие от стандартных портов USART они могут:

- работать с однопроводной полудуплексной линией связи;
- поддерживать интерфейсы Smart-карт стандарта ISO7618-3, LIN (local interconnection network) и IrDA (infrared data association);
- связываться с внешними устройствами, оснащёнными SPI-совместимым интерфейсом, по трехпроводной линии.

К особенностям порта USART микроконтроллеров STM32 также относится возможность дробного деления тактовой частоты для формирования заданной скорости работы. Благодаря этому можно получить стандартные скорости связи порта при любой частоте тактового сигнала.

Кроме того, с помощью блока DMA (Direct Memory Access) для любого порта USART может быть организован прямой доступ к памяти как для приёма, так и для передачи данных.

Порты USART STM32 способны поддерживать скорость обмена до 4,5 Мбит/с. Формат слова USART может составлять 8 или 9 бит данных и 0,5; 1; 1,5 или 2 стоповых бит. Дробные значения стоповых бит применяются в режиме порта Smartcard. Первым передаётся и принимается младший бит данных.

Приемопередатчики оснащены выходом синхронизации передатчика для обеспечения синхронной передачи.

Некоторые сигналы USART STM32 можно программно переопределить на другие выводы микроконтроллера.

13.3 Варианты индивидуальных заданий к лабораторной работе

Разработать программу для учебного стенда, позволяющую вводить данные с персонального компьютера через последовательный интерфейс RS-232, выполнять определенные действия над данными и выводить поясняющие надписи и результат вычислений на персональный компьютер через последовательный интерфейс RS-232 и на МЖКИ или на ССИ. Для работы на персональном компьютере использовать любую терминальную программу. Вариант задания получить у преподавателя.

Более подробные сведения о возможностях модуля UART с примером его использования для передачи информации по RS-232 можно найти в методических указаниях по лабораторной работе № 12, электронная версия которых имеется в классе ПЭВМ кафедры (лаб. 207/2) по следующему адресу: D:\методические рекомендации\СТАНДАРТЫ РБ\Специальность 1-53 01 05 АЭП и АПУ\Микропроцессорные средства в автоматизированном электроприводе\МПС 12.pdf, а также на сервере филиала кафедры по адресу \\Study\Методические указания\МПС в АЭП\МПС 12.pdf.

Контрольные вопросы

- 1 Для чего предназначен интерфейс RS-232?
- 2 Что такое последовательная передача данных?
- 3 Формат передачи данных интерфейса RS-232.
- 4 Какие соглашения по скоростям передачи данных приняты в RS-232?
- 5 Что такое DTE- и DTR-устройства?
- 6 Опишите низкоуровневый интерфейс подтверждения связи.
- 7 Опишите процесс управления потоком.
- 8 Каковы технические параметры интерфейса RS-232?
- 9 Перечислите технические возможности последовательного порта в микроконтроллере STM32F407.
- 10 Перечислите технические возможности последовательного порта в микроконтроллере STM32F407.
- 11 Как настроить скорость передачи данных через последовательный порт?
- 12 Опишите процесс приема/передачи данных через последовательный порт в микроконтроллере STM32F407.

14 Лабораторная работа № 13. Разработка и отладка программ управления технологическими объектами в реальном времени

Цель работы: изучить связь между аналоговыми регуляторами и цифровыми фильтрами, формы представления цифровых фильтров, особенности их программной реализации, реализацию с помощью языка программирования; составить программу реализации заданного цифрового фильтра, используя язык программирования микроконтроллера, перевести ее в машинные коды, записать в память программ микроконтроллера и проверить работу, используя аналоговые входы и выходы.

Перед началом лабораторных занятий студентам необходимо изучить следующие вопросы:

- разностные уравнения;
- принцип действия АЦП и ЦАП.

14.1 Цифровые системы управления

Цифровые системы строятся на базе комплекса средств вычислительной техники, основными элементами которого являются: цифровой вычислительный модуль (ЦВМ), устройства ввода, устройства вывода.

Функции ЦВМ могут выполнять: ЭВМ (компьютеры), DSP-цифровые сигнальные процессоры, цифровые устройства на жесткой логике. Первые относятся к универсальным устройствам управления, вторые специализированы для

приложений, третьи разрабатываются для конкретных устройств (например, цифровой фильтр, имеющийся в каждом $\sigma - \Delta$ АЦП).

Устройствами ввода и вывода в случае состыковки с аналоговыми сигналами являются АЦП и ЦАП, а в случае состыковки с цифровыми сигналами – порты и интерфейсы.

В системах с ЦВМ устройства могут выполнять роли: регулятора, регулятора и устройства сравнения, корректирующего устройства или самого объекта.

Если ЦВМ универсальная (ЭВМ), то возможно построение многофункциональных САУ, когда одна ЦВМ обслуживает комплекс составляющих объект устройств. Например, в автомобиле: система навигации, система бортового электропитания, ABS, электронная подвеска, управление топливopодачей и т. д. В подобных случаях в состав системы ЦУ должны входить аналоговые или цифровые мультиплексоры и демультимплексоры.

Во всех случаях ЦВМ предоставляет легко доступные информационные потоки, позволяющие, кроме прямого управления, осуществлять функции: контроля, оптимизации, координации и организации всех процессов.

14.2 Варианты индивидуальных заданий к лабораторной работе

Разработать программу для учебного стенда, позволяющую вырабатывать управляющее воздействие в соответствии с заданными параметрами ПИД-регулятора. Входное воздействие считывается с одного из аналоговых датчиков АД1–АД3 через АЦП. Управляющее воздействие выводить на выход аналогового сигнала через ЦАП. Вывести числовые значения, соответствующие входному воздействию и управляющему воздействию на индикатор ССИ или МЖКИ. Предусмотреть возможность генерации тестового воздействия в соответствии с состоянием дискретных датчиков ДД1–ДД8. Вариант задания получить у преподавателя.

Пример выполнения работы можно найти в методических указаниях по лабораторной работе № 13, электронная версия которых имеется в классе ПЭВМ кафедры (лаб. 207/2) по следующему адресу: D:\методические рекомендации\СТАНДАРТЫ РБ\Специальность 1-53 01 05 АЭП и АПУ \Микропроцессорные средства в автоматизированном электроприводе\МПС 13.pdf, а также на сервере филиала кафедры по адресу \\Study\Методические указания\МПС в АЭП\МПС 13.pdf.

Контрольные вопросы

- 1 Где применяются цифровые системы управления?
- 2 Какую роль играют АЦП и ЦАП в цифровых системах управления?
- 3 Опишите структуру ПИД-регулятора.
- 4 Какие существуют схемы алгоритмов реализации цифровых фильтров?
- 5 Опишите преимущества и недостатки последовательной и параллельной схем построения алгоритмов цифровой фильтрации.

Список литературы

- 1 **Гуров, В. В.** Микропроцессорные системы: учебное пособие / В. В. Гуров. – Москва: ИНФРА-М, 2022. – 336 с.
- 2 **Жежера, Н. И.** Микропроцессорные системы автоматизации технологических процессов: учебное пособие / Н. И. Жежера. – 2-е изд. – Москва; Вологда: Инфра-Инженерия, 2020. – 240 с. : ил.
- 3 **Овсянников, Е. М.** Электрический привод: учебник / Е. М. Овсянников. – Москва: Форум, 2011. – 224 с.
- 4 **Хартов, В. Я.** Микропроцессорные системы: учебное пособие / В. Я. Хартов. – Москва: Академия, 2010. – 352 с.