**Лабораторная работа № 4: Настройка UART интерфейса в микроконтроллере STM32.**

**Цель:** научиться настраивать UART интерфейс в микроконтроллере STM32 для передачи данных через COM порт компьютера.

**Оборудование и программное обеспечение:** плата STM32F4 Discovery, среда разработки

CooCox CoIDE 1.7.

**Теоретический материал**

UART интерфейс является последовательным интерфейсом, который используется для передачи и приема данных в виде последовательных битов между устройствами. В качестве сигналов передачи данных используются две линии: TX (Transmit) и RX (Receive). TX используется для передачи данных с одного устройства на другое, а RX - для приема данных. Обычно для работы с UART микроконтроллерSTM32 используются специальные периферийные модули USART, которые позволяют передавать данные с приемопередатчиком.

Универсальный синхронный/асинхронный приёмопередатчик (УАПП, англ. Universal Asynchronous Receiver-Transmitter, UART) — узел вычислительных устройств, предназначенный для организации связи с другими цифровыми устройствами. Преобразует передаваемые данные в последовательный вид так, чтобы было возможно передать их по цифровой линии другому аналогичному устройству. Метод преобразования хорошо стандартизован и широко применялся в компьютерной технике.

Представляет собой логическую схему, с одной стороны подключённую к шине вычислительного устройства, а с другой имеющую два или более выводов для внешнего соединения.

Пример кода для инициализации UART интерфейса в микроконтроллере STM32 на языке программирования C:

#include "stm32f4xx.h"

// Функция инициализации UART

void UART\_Init(void)

{

// Инициализация портов для работы с подключенным UART устройством

GPIO\_InitTypeDef GPIO\_InitStruct;

RCC\_AHB1PeriphClockCmd(RCC\_AHB1Periph\_GPIOB, ENABLE);

GPIO\_InitStruct.GPIO\_Pin = GPIO\_Pin\_6 | GPIO\_Pin\_7;

GPIO\_InitStruct.GPIO\_Mode = GPIO\_Mode\_AF;

GPIO\_InitStruct.GPIO\_OType = GPIO\_OType\_PP;

GPIO\_InitStruct.GPIO\_PuPd = GPIO\_PuPd\_NOPULL;

GPIO\_InitStruct.GPIO\_Speed = GPIO\_Speed\_50MHz;

GPIO\_Init(GPIOB, &GPIO\_InitStruct);

// Настройка альтернативной функции на пинах GPIOB6 и GPIOB7

GPIO\_PinAFConfig(GPIOB, GPIO\_PinSource6, GPIO\_AF\_USART1);

GPIO\_PinAFConfig(GPIOB, GPIO\_PinSource7, GPIO\_AF\_USART1);

// Инициализация модуля USART1

USART\_InitTypeDef USART\_InitStruct;

RCC\_APB2PeriphClockCmd(RCC\_APB2Periph\_USART1, ENABLE);

USART\_InitStruct.USART\_BaudRate = 9600;

USART\_InitStruct.USART\_WordLength = USART\_WordLength\_8b;

USART\_InitStruct.USART\_StopBits = USART\_StopBits\_1;

USART\_InitStruct.USART\_Parity = USART\_Parity\_No;

USART\_InitStruct.USART\_HardwareFlowControl = USART\_HardwareFlowControl\_None;

USART\_InitStruct.USART\_Mode = USART\_Mode\_Rx | USART\_Mode\_Tx;

USART\_Init(USART1, &USART\_InitStruct);

// Включение прерываний при приеме и передаче данных

USART\_ITConfig(USART1, USART\_IT\_RXNE, ENABLE);

USART\_ITConfig(USART1, USART\_IT\_TXE, ENABLE);

// Включение модуля USART1

USART\_Cmd(USART1, ENABLE);

}

Этот код выполнен для микроконтроллера STM32F4, однако в общем случае для других моделей микроконтроллеров STM32 может использоваться подобный подход к настройке UART интерфейса. В данном примере инициализируются порты для работы с подключенным UART устройством (GPIOB6 и GPIOB7), настраивается их альтернативная функция для работы с модулем USART1, задается скорость передачи данных (9600 бит/с), формат передачи данных (8 бит, 1 стоп-бит, без проверки четности) и включаются прерывания для передачи и приема данных. После инициализации модуль USART1 включается для начала работы с UART интерфейсом.

**Ход работы:**

1. Подключите микроконтроллер STM32 к компьютеру через COM порт.

2. Напишите программу, которая будет настраивать UART интерфейс на работу с заданной скоростью передачи данных через COM порт.

3. Напишите программу, которая будет отправлять строку данных через UART интерфейс на компьютер.

4. Включите программу и убедитесь, что передаваемая строка данных успешно отображается на экране компьютера.

**Индивидуальные задания**

1. Настройте UART интерфейс для работы на скорости 115200 бит/с.

2. Используя прерывания, реализуйте прием и передачу данных через UART интерфейс.

3. Разработайте программу, которая будет передавать и принимать символы через UART интерфейс, и подключите ее к ПК через USB-UART конвертер.

4. Настройте программу для передачи и приема двоичных данных (например, сигналы с датчиков), используя формат данных 8-бит, без проверки четности и 1 стоп-бит.

5. Разработайте программу для считывания данных с UART интерфейса и отправки их на LCD.