

## Лабораторная работа №4

### Цель работы:

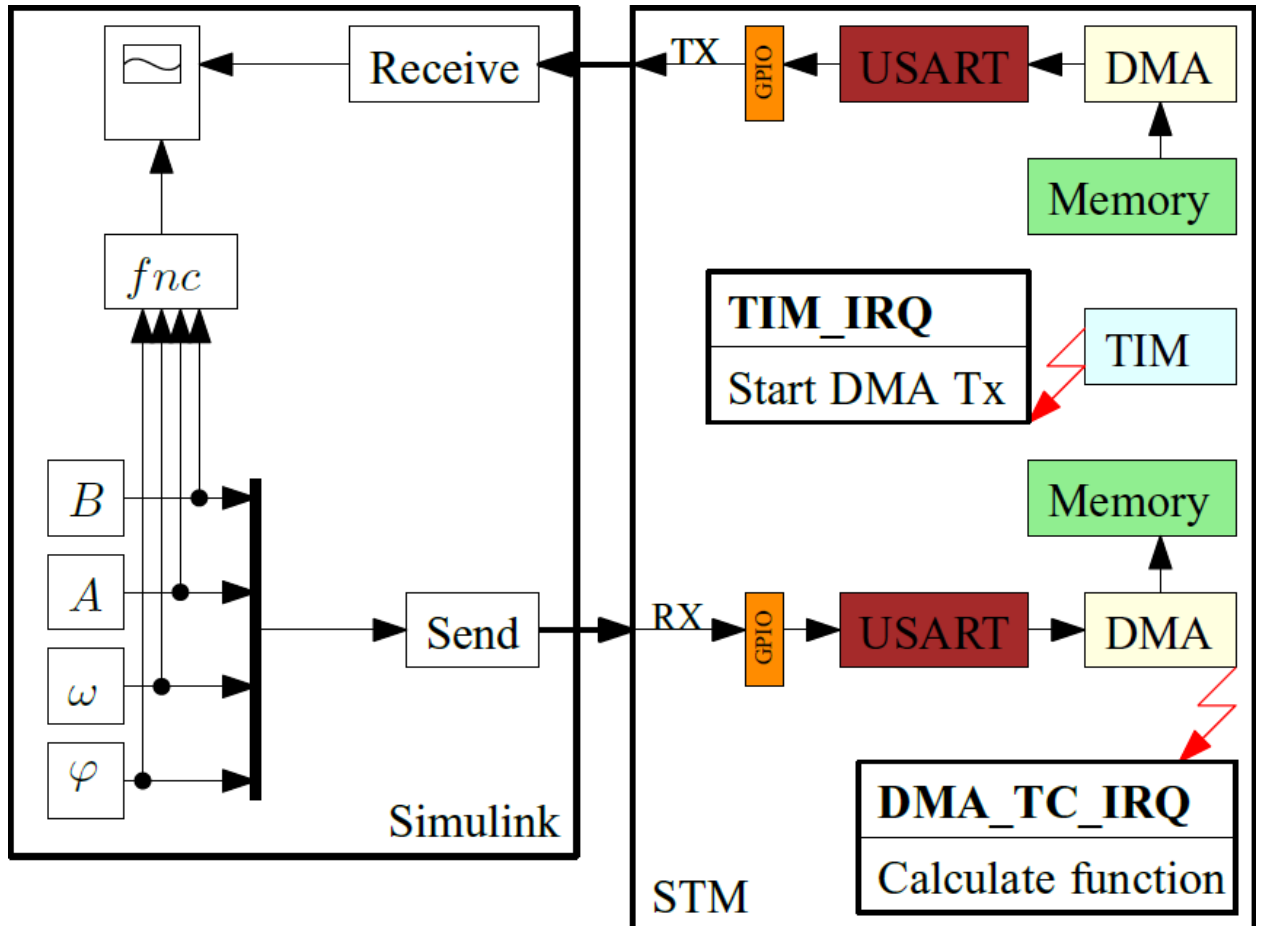
Получение навыков работы с таймерами и DMA, отправка пакетов данных в Matlab/Simulink.

### Программное обеспечение:

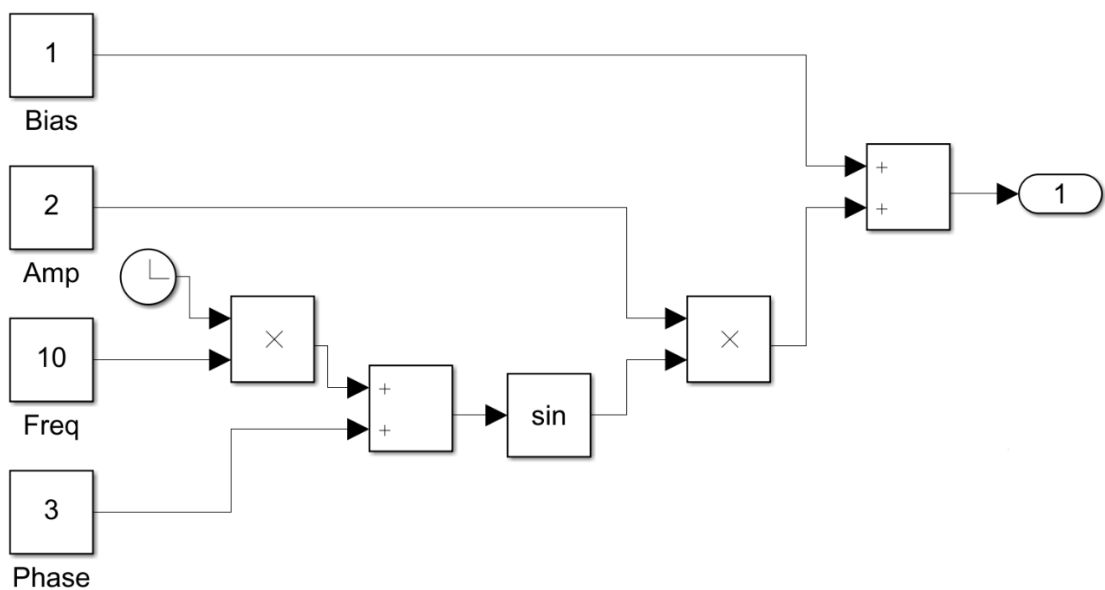
STM32CubeIDE, Matlab/Simulink.

### Общие сведения:

Схема разрабатываемой системы:

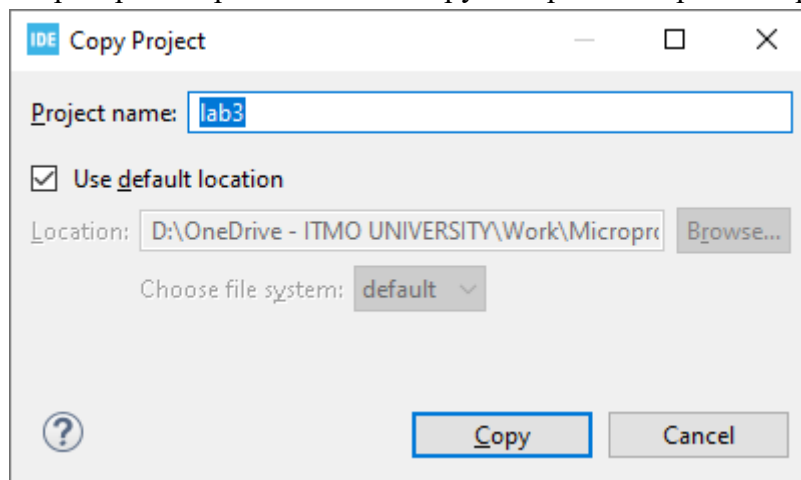


Для формирования сигнала в Matlab можно собрать такую схему:



### Порядок выполнения работы.

1. Запустите STM32CubeIDE, в качестве workspace выберите папку с предыдущими лабораторными работами. Скопируйте проект второй лабораторной работы.



2. Запустите Matlab, Simulink. Создайте новый проект. Для обеспечения приема и передачи данных от микроконтроллера по последовательному интерфейсу, добавьте на схему блоки:

- Serial Configuration;
- Serial Receive;
- Serial Send.

3. В блоке «Serial Configuration» настройте baudrate. Во всех блоках укажите COM порт, к которому подключен микроконтроллер.

4. Откройте Параметры системы(Model Settings) в Simulink, во вкладке «Solver» установите фиксированный шаг с размером 0.01.

5. Используя информацию из лекции, документацию на микроконтроллер и схему стенда, выполнить задание в соответствии с вариантом.

### Задание:

1. Настроить таймер на прерывание с частотой 100 Гц.
2. Настроить USART на прием и передачу данных при помощи DMA.
3. В прерывании по таймеру необходимо отправлять пакет данных в Matlab.
4. В прерывании от DMA по приему данных, необходимо обработать полученный пакет.
5. Написать программу, которая выполняет следующие функции:
  - Matlab отправляет в микроконтроллер параметры периодической функции;
  - микроконтроллер принимает данные, рассчитывает значение функции и отправляет результат вычислений обратно в Matlab;
  - Matlab принимает данные от микроконтроллера и строит график в score, график от микроконтроллера и сформированный самим matlab должны совпадать.

### Содержание отчета:

- Титульный лист.

- Цель работы.
- Листинг разработанной программы.
- Вывод.

**Варианты:**

№ варианта	$B$	$A$	$\omega$	$\varphi$
1	0	2	2	0
2	1	3	2	0
3	2	2	3	90
4	3	4	3	90
5	4	2	4	180
6	5	5	4	180
7	6	2	5	270
8	7	6	5	270
9	8	2	6	100
10	9	1	6	100