

Этап 4 проекта B06

Задание:

1. Скачайте с репозитория исходные коды программ [encoder](#), [keypad](#) и [step_motor](#), скопируйте их в соответствующие поддиректории папки `./stage_04`.
2. Самостоятельно скомпилируйте программы [encoder](#), [keypad](#) и [step_motor](#) из исходных файлов.
3. Создайте makefile для каждой из программ, переименовав имена исполняемых файлов на [encoder_ФИО](#) и [step_motor_ФИО](#). Обязательные команды: `all`, `clean`.
4. Модернизируйте исходные коды программ следующим образом - [keypad_ФИО](#) - добавить к сообщением о введенном приращении время ввода, а также добавить аргумент при вызове - указатель на именованный канал [keypad_data](#); [step_motor_ФИО](#) - должна работать без остановки и принимать вводимые значения углов поворота от именованных каналов [encoder_data](#) и [keypad_data](#). Обе программы должны обмениваться с именованными каналами с помощью системных вызовов `read()/write()`. Скомпилируйте и отладьте работу программ. Время можно определить с помощью функции [clock_gettime\(\)](#).
5. Модернизируйте `bash`-скрипт, запускающий эти программы, чтобы передавалось корректное число аргументов для программ [encoder_ФИО](#) и [step_motor_ФИО](#).
6. Скопируйте на ПК исходные коды, `make` файлы и `bash`-скрипт из [stage_04](#). Добавьте новое `readme.pdf` с инструкцией по сборке и запуску программ и сделайте коммит на сервер.
7. Продемонстрируйте преподавателю работу программ и `bash`-скрипта, а также созданный репозиторий.
8. Подготовьте ответы на вопросы этапа 4 проекта.

Список вопросов:

1. Что значат флаги компиляции программ [encoder](#) и [step_motor](#)?
2. Что делают команды `all` и `clean` в Ваших `make` файлах? Какие еще команды можно реализовать в `Make` файлах?
3. С помощью какой библиотеки определяется время в модернизированных программах?
4. Важен ли порядок передачи аргументов в программы при запуске и если да, то можно ли реализовать произвольную передачу аргументов?
5. Какого функционала еще не хватает Вашему комплекту ПО для выполнения требований курсового проекта?
6. Что такое объектные файлы и можно ли их удалить для успешного запуска программ [encoder_ФИО](#) и [step_motor_ФИО](#)?
7. Можно ли переписать `make` файл таким образом, чтобы он собирал сразу две программы [encoder_ФИО](#) и [step_motor_ФИО](#)?
8. Чем отличается Ваша инструкция к этой лабораторной работы от предыдущей?
9. Докажите, что Ваши исходные коды программ удовлетворяют [требованиям оформления исходных кодов](#) на нескольких правилах.
10. Какой следующий этап доработки Вашего проекта?

Порядок выполнения и сдачи [курсового проекта](#):

1. [Этап проекта №1](#)
2. [Этап проекта №2](#)
3. [Этап проекта №3](#)

4. [Этап проекта №4](#)
5. [Этап проекта №5](#)
6. [Этап проекта №6](#)
7. [Этап проекта №7](#)
8. [Этап проекта №8](#)