

Пересинхронизатор.

Это домашнее задание на 3 (три) очка. В этой работе Вам необходимо написать свой модуль пересинхронизации между двумя клоковыми доменами. В действительности, в этой работе не будет асинхронных доменов, так что необходимости в использовании пересинхронизаторов в реальной задаче в этом случае нет. Но т.к. это не реальный проект, а образовательный, все же предлагаю его сделать.

1. Создайте два делителя частоты `clk0` и `clk1` с разной выходной частотой. Можете либо создать два блока, либо один с использованием параметра деления частоты.
2. Создайте блок ROM M9K памяти с одним входом и следующими параметрами:
 - (a) Установите размер одного слова 16 бит, размер памяти 16 слов.
 - (b) Создайте файл начальной инициализации ROM памяти (расширение `.hex`). Пока пропишите там любые произвольные значения. Удобно пользоваться редактором Vim, после открытия пропишите опцию `:%!xxd` для перехода в режим hex. Либо можно создать файл с таким расширением прямо в квартусе.
3. Запишите в файл инициализации ROM памяти код Грея. В каждом слове из 16 бит 4 числа по 4 бита (т.е. 4 hex числа). При этом каждое hex число является либо 0 либо 1. Т.е. в файле должны быть строки типа 0000, 0001, 0011, 0010, ..., при этом каждый ноль либо единица занимают четыре бита.
4. Создайте блок пересинхронизации. Необходимо создать блок описанный в статье <https://pdf4pro.com/view/clock-domain-crossing-cdc-design-amp-verification-2b742.html> (она же на вики в литературе по 7 лекции) на странице 27. Входы данных при этом должны быть размера 16 бит.
5. Создайте блок вывода четырех hex чисел на четыре семисегментных индикатора. Скорее всего Вам потребуется написать еще один делитель частоты.
6. Подключением блоков и написанием небольших вспомогательных блоков реализуйте следующий функционал: часть схемы, работающая по `clk0` читает значение из памяти и через блок пересинхронизатора отправляет прочитанное значение в часть схемы, работающую по `clk1`. Далее, часть схемы работающей от `clk0` ждет, пока к ней не придет ask из блока пересинхронизации от клокового домена `clk1`. После этого загружает следующее значение и далее все повторяет. Не забудьте, что необходимо правильно управлять сигналами пересинхронизатора в `clk0`. При этом часть схемы, работающая от `clk1`, выводит полученные значения на на семисегментный индикатор. Позаботьтесь, чтобы период `clk1` был таким, чтобы значения на семисегментном индикаторе хорошо воспринимались глазами.
7. Загрузите прошивку на ПЛИС. Проверьте, что при разных соотношениях клоков `clk0` и `clk1` (`clk0` больше или меньше `clk1`) значения не теряются и успешно передаются.
8. Код проекта выложите на гитхаб.