

<https://znaniya.com/task/28405254>.

Предложите схему, которая на каждый тактовый импульс формирует очередной код из последовательности:

а) 000011, 000110, 001100, 011000, 110000, 100001, ...;

б) 000111, 001110, 011100, 111000, 110001, 100011, ...

Приведите временные диаграммы работы. Помогите пожалуйста, буду вам благодарна. Разрядов 6.

РЕШЕНИЕ

Я вчера так прикинул. Если на рассыпной логике, то три ИС придется задействовать. Логика работы похожа на кольцевой счётчик. Если организовать предварительную загрузку кода, то можно обойтись для прокручивания кодов на выходе одной схемой для обоих вариантов задания.

Поскольку разрядов 6, то задействовать сдвиговые регистры сдаётся мне неудобным. Они на 8 или 4 разряда. Возможно собрать некоторое подобие на отдельных триггерах. Есть мысль собрать такую схему (смотрите прилагаемый к ответу файл со схемой). Основа — D триггеры TM8 (74LS175) триггерах (K55TM8, K1533TM8, 74LS175, 74LS175). Элементы на DD1, DD2 на схеме.

Требуется предварительная загрузка кода, который будет прогоняться по счётчику. Для этого я предлагаю применить один мультиплексор 2 в 1, например K1533КП16 (ALS157) на схеме элемент DD3.

Управляющие сигналы подаются через разъём X1. На контакт 1, цепь *Reset* подается сигнал сброса. Подача лог 0 переводит все выходы (сигналы *D0 — D5*) в 0. Если не требуется управляемый сброс, можно задействовать RC цепочку для сброса только при включении питания. Фильтрующие конденсаторы не показаны. При этом устранены «висячие» входы.

Неиспользуемые входы цифровой ИМС должны быть подтянуты либо к 0, либо к 1.(в нашем случае либо к GND, либо к +5V).

Загружаемый код подаётся на контакт 3, цепь *Input*. Сигнал синхронизации подаётся на X1.2, цепь *SYN*. Перед загрузкой на контакт X1.2, цепь *Select*, подаётся лог 1. При этом на вывод 1 элемента DD3 будет передаваться сигнал с вывода 3. Затем последовательно на контакт X1.3, цепь *Input* передаётся требуемый код. Биты синхронизируются положительным фронтом сигнала *SYN*. По окончании загрузки на контакт X1.2, цепь *Select*, подаётся лог 0. При этом на вывод 1 элемента

DD3 будет передаваться сигнал с вывода 2, соединённым с выходом счётчика (DD2 вывод 7). Дальнейшее поступление импульсов *SYN* заставит код смещаться на одну позицию влево (от младших к старшим разрядам кода) с каждым фронтом *SYN*. Поскольку сигнал с выхода счётчика (DD2 вывод 7) через мультиплексор передаётся на вход, значение из самого старшего разряда будет передаваться в самый младший. Таким образом выходная комбинация (сигналы *D0 — D5*) будет изменяться циклически.

NB. В этом случае сигнал *SYN* не является тактовым для всей схемы, если рассматривать её вместе со схемой загрузки, которую мы тут не показали.

На выходах (цепи сигналов *D0 — D5*) дорисованы элементы световой индикации (светодиоды HL1 — HL6 с токоограничивающими резисторами R1 — R6).

Схема одна на оба варианта. Разный только код, который вначале загружается в счётчик.

Диаграммы приведены ниже на рисунке 1. Строим их из расчёта, что при загрузке передаётся все 6 бит кода. При этом старший бит передаётся первым. Штриховкой обозначены безразличные значения сигнала. Т. е. что бы там не было : 0 или 1 на состояние выходов это не влияет.

Замечу, что если б мне нужно было переключать гирлянды (эффект бегущий огонь или бегущая тень, а куда ещё такое применить), я бы просто использовал (и использовал, когда надо было) контроллер. Аппаратная часть сильно упрощается. Контроллер и буферные элементы. В нашем случае достаточно задействовать один вывод контроллера как вход, если есть внешний тактовый сигнал. В противном случае обходимся тактовым генератором микроконтроллера. Шесть выводов какого либо порта контроллера надо включить как выходы (сигналы *D0 — D5*). Вся логика работы определяется программой микроконтроллера. А количество эффектов ограничено лишь фантазией программиста, ну и скоростью работы МК.

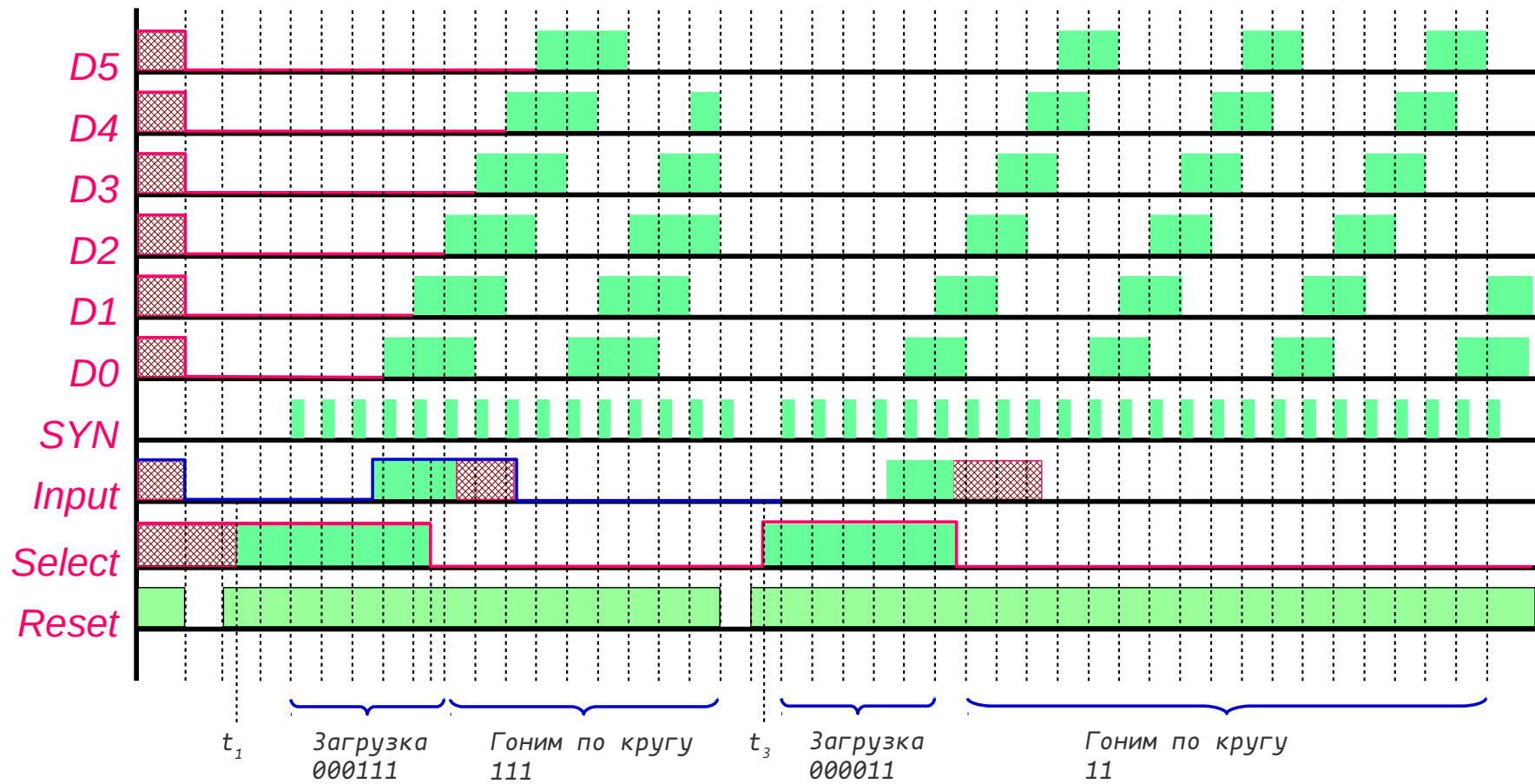


Рисунок 1: Временные диаграммы работы схемы. Тут представлена работа схемы по обоим вариантам. В моменты t_1, t_3 — включаем режим загрузки. Сначала прогоняем вариант б), затем, после сброса, вариант а).