

# Метастабильность

## Синхронные и асинхронные цифровые устройства

Современные цифровые устройства обычно являются синхронными. Это значит, что все изменения сигналов фиксируются по фронту или срезу сигнала тактирования. При этом основным элементом в синхронном цифровом устройстве является D-триггер, синхронный, с динамическим управлением (D-flip-flop).

Условие синхронности задает требование к разрешенным моментам изменения сигналов на входах триггеров: в синхронном устройстве эти изменения не могут происходить очень близко к фронту (или срезу) сигнала тактирования.

Тем не менее, обойтись без асинхронных изменений сигналов на входах триггеров практически невозможно. Асинхронные сигналы – это такие сигналы, которые могут меняться в произвольный момент времени, в том числе и близко к фронту<sup>1</sup> (или срезу) сигнала тактирования. Источником асинхронных сигналов могут быть различные переключатели (кнопки, клавиатуры и т.п.), подключенные к FPGA, какие-то внешние шины. Также асинхронные сигналы могут возникать и внутри FPGA, когда осуществляется передача сигнала из модуля, тактируемого одной тактовой частотой, в модуль, который тактируется другой тактовой частотой<sup>23</sup>.

Во всех этих случаях новый сигнал на вход триггера может прийти в любой момент времени, в том числе и с нарушением времен установки или удержания триггера.

## Метастабильность

Будем считать далее, что данные на входе триггера меняются с частотой меньшей частоты сигнала тактирования.

Для надежной работы триггера новый сигнал на его входе не должен меняться в течение времени установки и удержания. В этом случае через некоторое время  $T_{clk2out}$  (clock-to-output delay) после фронта на выходе триггера появится соответствующий выходной сигнал. Если же новый сигнал появится с нарушением времени установки или удержания, то произойдет одно из следующих событий:

- триггер все-таки захватит новый сигнал;
- триггер не успеет захватить новый сигнал, а выход триггера не изменится (сохранится старое состояние) (т.к. данные на входе меняются медленнее, чем сигнал тактирования, то новый сигнал будет захвачен на следующем фронте);
- триггер перейдет в метастабильное<sup>4</sup> состояние.

Однозначно предугадать, какое из событий произойдет нельзя.

Метастабильное состояние – это такое состояние неустойчивого равновесия последовательностного элемента (чаще всего триггера), в котором он не может разрешить состояние входного сигнала. Как следствие, на выходе появляется неразрешенное состояние на неопределенном интервале времени. Вследствие того, что новый сигнал пришел слишком близко по времени с фронтом сигнала тактирования, триггер не успевает корректно этот сигнал захватить.

---

<sup>1</sup> Далее не ограничивая общности будем считать, что триггер захватывает данные по фронту сигнала тактирования.

<sup>2</sup> В этом случае, говорят, что осуществляется переход между тактовыми доменами.

<sup>3</sup> Например, в FPGA реализованы интерфейс к внешней DDR памяти, который работает на одной тактовой частоте, и вычислительный модуль обработки данных из этой памяти, который работает на другой тактовой частоте.

<sup>4</sup> Вообще говоря, “метастабильное состояние” – это жаргонизм. Корректнее говорить “нестабильное состояние”. Тем не менее, общепринятый термин – “метастабильное состояние”.

Триггер начинает захват сигнала (например, заряд конденсатора), при этом начинает соответствующим образом меняться и выходной сигнал триггера, но в какой-то момент приходит фронт и триггер отключается от входа. Как следствие, переключение между логическими уровнями не завершилось, а на выходе остался сигнал промежуточного по напряжению уровня: между нулем и напряжением питания (например, для CMOS33 1,65 В).

Вследствие флуктуационных шумов внутри триггера он с течением времени выйдет из метастабильного состояния, но однозначно сказать, что будет на выходе нельзя: будет либо новое состояние, либо старое. При этом время появления выходного сигнала с корректным логическим уровнем (а не половина  $V_{dd}$ ) будет больше, чем  $T_{clk2out}$ .

Вероятность того, что триггер перейдет в метастабильное состояние и время выхода из него определяются условиями работы устройства (температура) и технологическим процессом (скорость переключения транзистора, напряжение питания и др.). В большинстве случаев триггер достаточно быстро вернется в разрешенное состояние.

Захват данных триггером можно иллюстрировать падением мяча на холм (рис. 1). Подошвы холма соответствуют разрешенным стабильным состояниям триггера (старому и новому). При асинхронном входном сигнале мяч будет падать либо прямо на подошвы холма, либо на его склоны, либо точно на вершину. В первом случае при выполнении требования по времени установки и удержания мяч упадет на правую (новую) подошву холма, данные будут захвачены корректно, а если сигнал придет после времени удержания, то мяч упадет на старую подошву холма, состояние триггера не изменится. Если же требования по времени установки и удержания нарушены, то мяч упадет на склоны холма или даже на его вершину. Время перехода триггера в стабильное состояние увеличится.

Вследствие неидеальностей мяч постепенно все-таки скатится с вершины в одно из устойчивых состояний, но предугадать его нельзя. Стоит также отметить, что мяч будет скатываться с ускорением: сначала он будет двигаться очень медленно, а под конец – быстро.

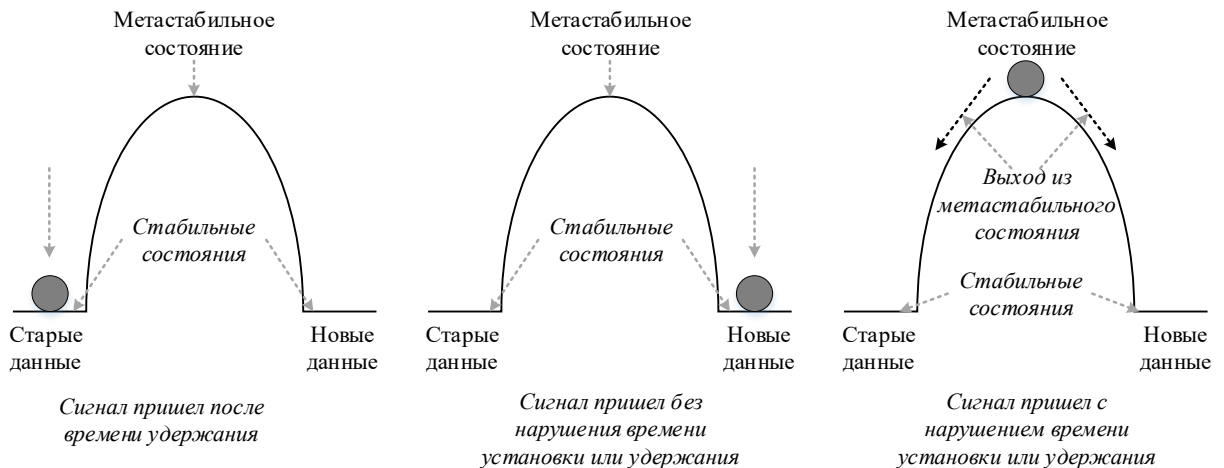


Рис. 1. Иллюстрация перехода триггера в стабильное или метастабильное состояние

На рис. 2 показано поведение сигнала на выходе триггера при переходе в метастабильное состояние и выходе из него.

Входной сигнал переходит из логического нуля в логическую единицу с нарушением времени установки. Выходной сигнал начинает меняться в соответствии со входным сигналом, но это изменение прекращается примерно посередине между уровнями напряжения логического нуля и единицы. Через некоторое время триггер переходит в одно из стабильных состояний. Причем время этого перехода больше  $T_{clk2out}$ .

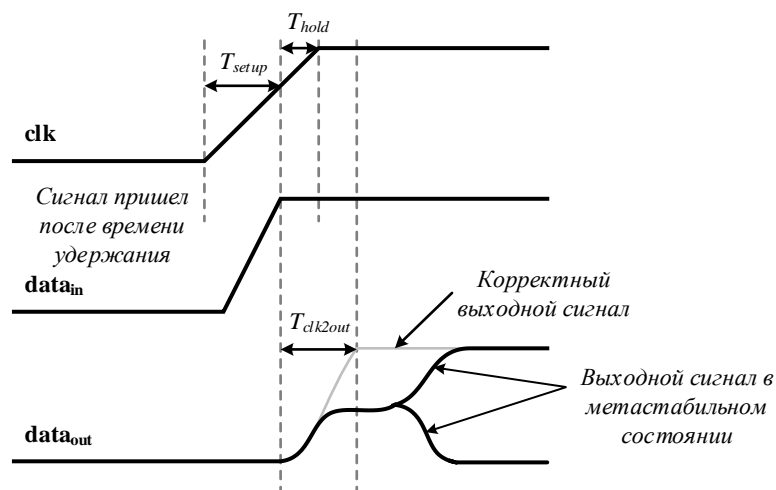


Рис. 2. Пример возникновения метастабильного состояния и выхода из него

Проблема метастабильного состояния проявляется, когда оно длится достаточно долго. В этом случае на входе триггера, который подключен к триггеру, находящемуся в метастабильном состоянии, будет некий промежуточный уровень напряжения. Если метастабильный триггер выдает сигнал на несколько триггеров, то одинаковые триггеры захватят такой промежуточный уровень напряжения по-разному: некоторые – как логический ноль, некоторые – как логическую единицу. Как следствие, например, конечный автомат перейдет в такое состояние, в которое при нормальной работе перейти не может. Произойдет отказ в работе устройства.

## Среднее время безотказной работы

Важнейшая вещь, которую необходимо понимать и с которой нужно смириться, – это то, что устранить или полностью победить метастабильность невозможно, если идет захват триггером асинхронного сигнала. Единственное, что можно сделать, – это уменьшить вероятность отказа до исчезающе малой величины или, что то же самое, увеличить обратную величину: среднее время безотказной работы (MTBF, mean time between failures).

Среднее время безотказной работы рассчитывается по следующей формуле:

$$MTBF = \frac{e^{t_{MET}/C_2}}{C_1 F_{clk} F_{DATA}}, \quad (1)$$

где  $C_1$  и  $C_2$  – константы, определяющиеся технологическим процессом и условиями работы;  $F_{clk}$  – частота тактирования захватывающего триггера;  $F_{DATA}$  – частота изменения данных на входе захватывающего триггера;  $t_{MET}$  – время, доступное для выхода из метастабильного состояния (помимо  $T_{clk2out}$ ).

Для одиночного триггера на современных тактовых частотах и скоростях передачи данных среднее время безотказной работы измеряется... всего лишь десятками миллисекунд.

Из (1) можно сделать следующие выводы:

- увеличение тактовой частоты уменьшает MTBF;
- увеличение частоты изменения данных уменьшает MTBF;
- основное влияние на MTBF оказывает константа  $C_2$  и время  $t_{MET}$ .

Таким образом, основным методом увеличения времени безотказной работы является увеличение времени  $t_{MET}$ , которое отводится триггеру для выхода из метастабильного состояния.

Если в FPGA присутствует несколько точек, в которых выполняется захват асинхронных сигналов, то их вероятности отказа складываются:

$$\frac{1}{MTBF_{device}} = \sum_{i=1}^N \frac{1}{MTBF_i} \quad (2)$$

где  $N$  – количество точек, в которых выполняется захват асинхронных сигналов.

Важно также помнить, что, если устройство с неким MTBF выпускается сотнями миллиардов штук, то общий MTBF также считается по формуле (2) (из-за вероятностной природы попадания в метастабильное состояние). Он будет значительно меньше MTBF одного устройства.

## Методы увеличения MTBF, виды синхронизаторов

Для увеличения времени безотказной работы применяются специальные устройства – синхронизаторы.

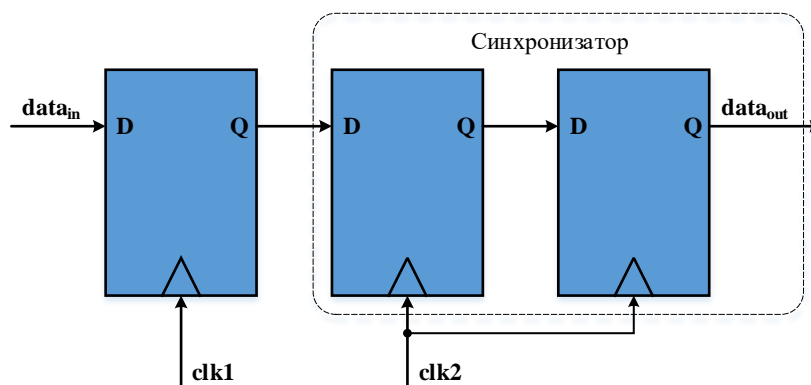


Рис. 3. Синхронизатор из цепочки триггеров

Простейшим синхронизатором (рис. 3) является цепочка триггеров, расположенных максимально близко друг к другу. Выход каждого триггера в цепочке подключен ко входу последующего. В этом случае время, которое отводится первому в цепочке триггеру для выхода из метастабильного состояния складывается из числа периодов на 1 меньше числа триггеров в цепочке (с точностью до суммарного времени распространения сигнала между триггерами). Между тем, синхронизатор на рис. 3 не гарантирует надежного захвата асинхронных многоразрядных данных. В этом случае ставится по синхронизирующей цепочке на каждый разряд, малая вероятность попадания в метастабильное состояние обеспечивается, но вот одновременное защелкивание многоразрядных данных – нет. Это значит, что на шине данных в определенный момент времени на выходе банка синхронизаторов в некоторых разрядах будут защелкнуты старые значения, а в некоторых – новые.

Для захвата асинхронных многоразрядных сигналов можно применять синхронизаторы с запросом-ответом, когда через синхронизатор на рис. 3 пропускаются только сигналы запроса-разрешения на захват данных.

В общем же случае для захвата асинхронных многоразрядных сигналов, а также для перехода от высокой тактовой частоты к меньшей, применяются асинхронные FIFO на основе двухпортовой памяти RAM (рис. 4).

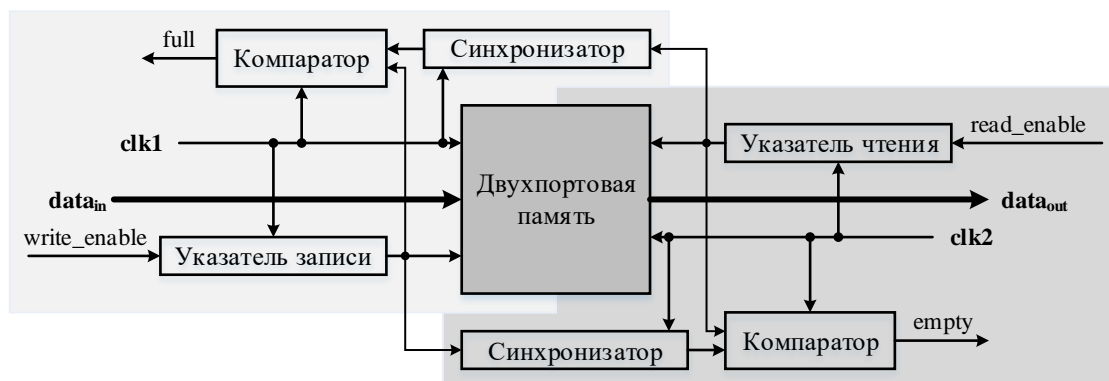


Рис. 4. Структурная схема асинхронного FIFO на основе двухпортовой памяти

В синхронизирующий FIFO запись данных осуществляется в домене одной тактовой частоты, а чтение – в домене другой тактовой частоты. В синхронизирующем FIFO указатель записи указывает на первую после данных пустую ячейку в памяти, т.е. указывает на место, куда можно записывать новый входной сигнал. Указатель чтения, в свою очередь, показывает из какой ячейки памяти выполнять очередное считывание. Память циклически замкнута. Запись осуществляется по сигналу разрешения записи `write_enable`, чтение – по сигналу разрешения чтения `read_enable`. Компараторы анализируют значения указателей записи и чтения и сообщают о фактах заполненного или пустого FIFO на выходах `full` и `empty` соответственно.

В современных FPGA асинхронные FIFO поддерживаются на аппаратном уровне.