

ПРОГРАММНОЕ МОДЕЛИРОВАНИЕ КОМБИНАЦИОННЫХ СХЕМ

Луговой В.М., Постников А.И.

Научный руководитель – доцент Постников А.И.

Сибирский федеральный университет

В настоящее время существует огромное разнообразие электронных вычислительных машин, большинство из которых состоят – цифровые вычислительные машины. При проектировании цифровых схем для выбора оптимального варианта требуется оценивать аппаратурные затраты.

Для их определения в качестве абстрактной единицы аппаратуры целесообразно использовать логический элемент Буля, Шеффера, Пирса. Уменьшение числа элементов значительно упрощает схему, ведёт к снижению стоимости устройства. Другая характеристика цифровой схемы – время её работы (время, необходимое для прохождения сигналом самого длинного пути, и завершение переходных процессов в схеме). Как известно, эти два критерия, аппаратурные затраты и время работы, взаимоисключающие. Так, увеличив число элементов, можно добиться некоторого параллелизма выполнения логических операций и уменьшения времени распространения переходных процессов в схеме, проиграв при этом в энергопотреблении и простоте реализации. Таким образом, разработчики всегда вынуждены идти на компромисс и выбирать приемлемый вариант.

Цифровые схемы, построенные на элементах Буля, Шеффера, Пирса, представляют собой большую совокупность разнообразных элементов и соединений между ними. Каждый элемент обладает свойственными ему задержками распространения сигнала, что затрудняет определение времени распространения сигнала. Кроме того, время получения сигнала на выходе комбинационной схемы зависит от значений входных сигналов, поданных в схему, а также от предыдущего состояния всех элементов, которое, в свою очередь, определяется предыдущим набором входных сигналов. Действительно, если часть элементов схемы при поступлении новых данных не изменит своего состояния (сигнал на выходе элемента не изменится), то элементы, ожидающие сигнал из этой части, могут сработать быстрее. Следовательно, иногда имеет смысл знание времени срабатывания схемы при смене одной определённой комбинации входных сигналов на другую. А это можно сделать, лишь перебрав все входные комбинации данных, которые могут быть поданы на схему с учетом предыдущего состояния, определяемого предыдущей комбинацией входных сигналов.

Однако, каждый из элементов Буля, Шеффера, Пирса имеет своё время срабатывания, причём периоды переключения элемента из нулевого состояния в единичное и обратно, как правило, различны. Поэтому на этапе проектирования сложной функциональной комбинационной схемы для оценки времени её срабатывания целесообразно воспользоваться абстрактной единицей времени срабатывания (переключения) элемента. Будем считать, что элементы, из которых состоит цифровая схема, а это элементы Буля, Шеффера, Пирса, срабатывают за время τ .

Понятно, что анализировать сложную схему, начерченную на листе бумаги практически невозможно по причине большого количества элементов, связей между ними и пар (текущей и предыдущей) комбинаций входных сигналов. Решением этой проблемы является моделирование работы совокупности всех элементов с учётом связей между ними, при этом, по возможности, стоит учесть как можно больше особенностей работы логических элементов.

Авторами разработана программа, обобщённый алгоритм работы которой представлен на рис 1. Эта программа позволяет смоделировать работу логических цифровых схем с учётом того, что некоторые элементы срабатывают одновременно, параллельно, а другие – последовательно после переключения предыдущих. В программе рассмотрен перечень всех возможных входных комбинаций при различных предыдущих состояниях логических элементов и приведена итоговая таблица с результатами.

Результатами работы программы являются:

- выходные данные, сформированные схемой при подаче определённой комбинации входных сигналов;
- время работы схемы после завершения переходных процессов и ее перехода в установившийся режим, выраженное в абстрактных единицах времени работы τ ;
- возможность контроля правильности работы схемы.

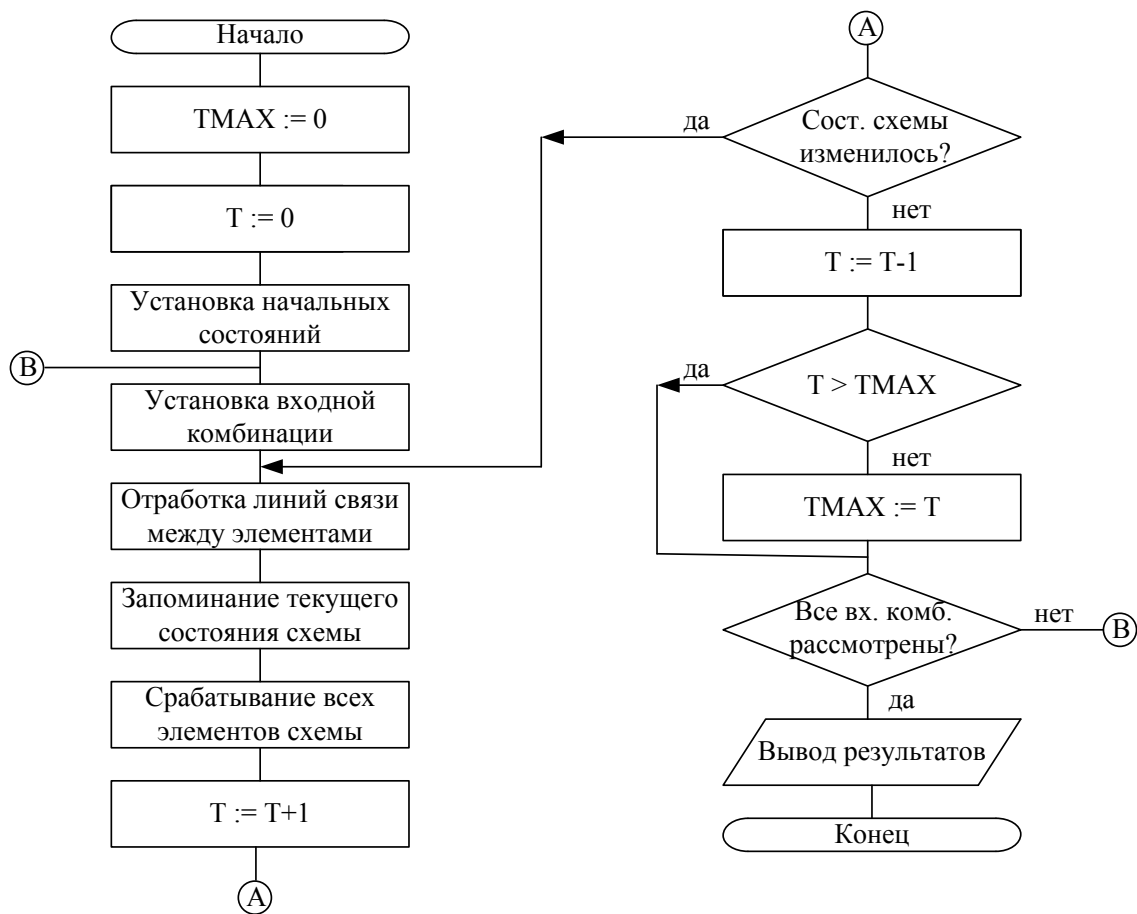


Рис.1. Обобщённый алгоритм программы моделирования комбинационных схем

$TMAX$ и T – максимальное время работы схемы, требуемое для завершения переходных процессов, а также время при обработке текущих входных данных. Каждый элемент представляется в виде метода, в котором задается задержка его срабатывания, а также логическая функция, которую он выполняет. Существует два основных контейнера памяти: первый наполнен логическими элементами, второй – проводниками, передающими информацию с выходов одних элементов на входы других. При первом запуске схемы опционально можно установить все элементы в неопределенное состояние. После чего подается входной сигнал и запускается цикл работы схемы. Вся работа считается дискретной. Она разделена на такты длительностью равной абстрактному

времени срабатывания элемента τ . Для каждого элемента количество тактов, необходимых для переключения, различно. После каждого такта все выходные сигналы копируются в массив, что нужно для определения состояния, когда все элементы отработают и перейдут в установившийся режим. Если при сравнении текущего состояния всех выходных сигналов на всех элементах и предыдущего, записанного в массиве, будут обнаружены различия, то это означает, что в данном такте работы было изменение выходного сигнала хотя бы одного из элементов, составляющих схему, т.е. срабатывание этого элемента.

В программе все логические элементы схемы срабатывают по очереди в рамках одного такта (псевдопараллельно), результатом этого является появление на выходах логических значений, которые потом по линиям связи, описанным в программе, переносятся на входы других элементов. Таким образом, проводник в модели содержит номера элементов и номера входов и выходов данного соответственно.

Программа реализована на языке *C++* под операционную систему *Windows*, с применением библиотек *STL* и *MFC*.

Проверка правильности работы программы моделирования, реализующей представленный на рис.1 алгоритм, была проведена на достаточно простых схемах, имеющих одновременно параллельно срабатывающие элементы. В качестве одной из схем, на которых производилась проверка корректности работы моделирующей программы, использовалась функциональная схема полного одноразрядного сумматора (рис. 2, б), условное графическое обозначение (УГО) которого представлено на рис. 2, а.

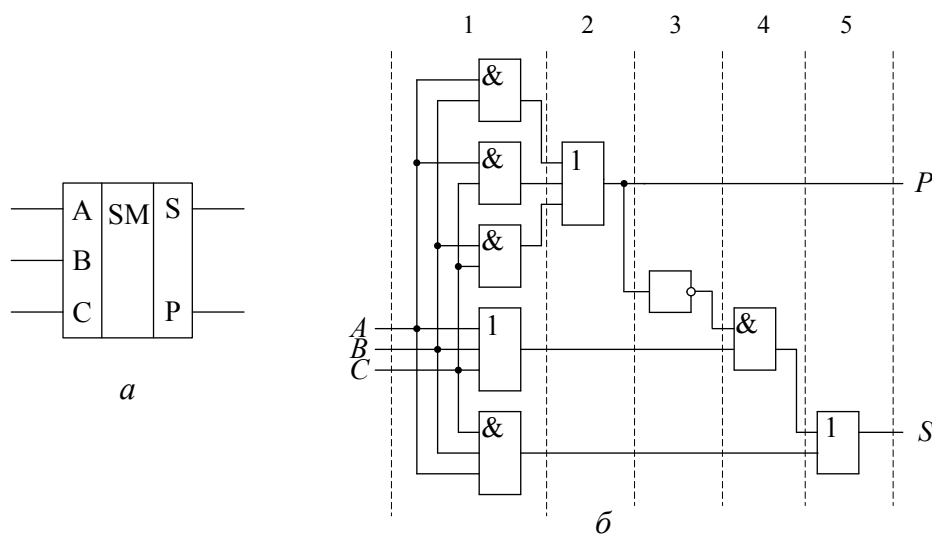


Рис.2. УГО полного одноразрядного сумматора (а) и его функциональная схема (б)

Данная функциональная схема имеет пятиярусную структуру. Элементы, расположенные в одном ярусе, срабатывают одновременно, и выходные сигналы из этого яруса являются входными сигналами для элементов, расположенных в следующих ярусах. Таким образом, для вышеприведённой схемы можно визуально определить, через какое время, рассчитываемое в абстрактных тактах τ , появятся выходные сигналы суммы S и переноса P в данном разряде в следующий разряд.

Разработанная программная модель показала, что время появления сигнала переноса составляет 2τ , а время появления сигнала суммы – 5τ . Аппаратурные затраты на схему составляют 9 элементов.

С помощью данной программы выполнено моделирование схем многоразрядных параллельных сумматоров с последовательным переносом, работающих в дополни-

тельных двоично-десятичных кодах Д1 (код 8421) и Д4 (код 8421+3). При представлении двоично-десятичных чисел каждая десятичная цифра представляется тетрадой двоичных символов.

Авторами были разработаны модули сумматоров одной тетрады двоично-десятичных чисел, представленных в дополнительных кодах Д1 и Д4. В качестве метода суммирования двоично-десятичных чисел, представленных в коде Д4, был выбран метод с четырьмя поправками [1]. Последовательное соединение таких модулей позволяет проектировать сумматоры любой разрядности, для чего они соединяются последовательно. Перенос внутри каждого модуля также распространяется последовательно как при суммировании тетрад входных чисел, так и при прибавлении соответствующих поправок. Работа этих модулей была проверена с помощью разработанной программной модели, результат работы которой позволил сделать выводы о правильности их работы. Также были получены количественные характеристики моделируемых схем, суммирующих одноразрядные двоично-десятичные числа со знаком, которые сведены в табл. 1.

Таблица 1

Код	Аппаратурные затраты	Время работы
Д1	80	19τ
Д4	98	24τ

Таким образом, моделирование показало, что при сложении одноразрядных десятичных чисел, представленных в двоично-десятичном коде как по аппаратуре, так и по времени срабатывания схемы, преимущество имеет код Д1.

В программе, в зависимости от разрядности слагаемых десятичных чисел, схема может динамически компоноваться, что позволяет суммировать числа разной разрядности за разное количество тактов. Построение и моделирование многоразрядных сумматоров показало, что с увеличением разрядности слагаемых, выигрыш в быстродействии будут иметь схемы, работающие в коде Д4. Это происходит вследствие того, что в сумматорах кода Д1 перенос в следующую тетраду формируется после прибавления поправки в данной тетраде, т.е. последовательно, с суммированием входных операндов, а в сумматоре кода Д4 перенос в следующую тетраду формируется сразу после сложения исходных двоично-десятичных цифр данной тетрады, а поправки ко всем тетрадам результата прибавляются одновременно после получения знака суммы.

В дальнейшем авторы планируют создание графической оболочки, что позволит пользователям, не знакомым с программированием, составлять и тестировать различные схемы. Планируется добавление элементов памяти (триггеров, регистров), мультиплексоров. Сейчас реализована возможность создания своих модулей на базе основных логических функций, введенных в программу, и использование этих модулей при моделировании сложных схем на их основе.

Разработанная программная модель комбинационных схем может быть использована в учебном процессе в дисциплинах "Схемотехника ЭВМ", "Теория цифровых автоматов" и т. п.