

## **Приложение свойств обобщённых логических функций к синтезу многофункциональных триггерных устройств**

*Белгородский государственный технологический университет, Россия  
Национальный аэрокосмический университет им. Н.Е. Жуковского «ХАИ»*

**Постановка проблемы.** Общеизвестно, что одним из путей улучшения основных характеристик цифровых устройств автоматики, вычислительной техники и систем управления является использование программируемых логических интегральных схем (ПЛИС) [1]. Однако здесь возникает ряд проблем, обусловленных ограниченностью отечественного рынка ПЛИС, стоимостью импортной элементной базы и пакетов систем автоматизированного проектирования (САПР).

Кроме того, доступные пакеты САПР могут по каким-то критериям не удовлетворять разработчиков аппаратуры, а несанкционированной корректировке, как правило, не поддаются.

Решение возникающих при этом проблем под силу крупным предприятиям и экономически оправдано только при серийном производстве. Кроме того, использование ПЛИС не всегда целесообразно при изготовлении экспериментальных образцов, макетов в учебных и научно-исследовательских лабораториях, а также устройств мелкосерийного и штучного производства нетиповых блоков и узлов цифровых устройств автоматики и систем управления.

Поэтому актуальны вопросы разработки новых аналитических методов логического проектирования и совершенствования существующих, поскольку такие методы могут использоваться самостоятельно, а также служить основой разработки новых пакетов САПР.

подавляющее большинство известных аналитических методов логического проектирования основано на традиционном представлении значений функций в области определения в виде логического нуля и единицы.

Наряду со многими достоинствами такого представления оно имеет и некоторые недостатки, обусловленные большим числом точек области определения, что влечёт за собой усложнение многих задач анализа и синтеза, связанное с решением объёмных комбинаторных задач.

Как было показано в [2], представление традиционных логических функций в форме обобщённых логических функций (ОЛФ) позволяет упростить процедуру минимизации, а разработанные программные средства сжатия области определения [3] дают возможность автоматизировать процесс представления этих функций в форме обобщённых с последующей их минимизацией в сжатой области.

При синтезе многофункциональных триггерных устройств, имеющих большое число управляющих сигналов, которые обеспечивают настройку на заданные режимы, возникают трудности, обусловленные большой размерностью таблицы функционирования. Более того, сложилось мнение [4] о принципиальной невозможности решения подобных задач формальным способом, основанным на табличном представлении функций при приемлемых размерах таблицы.

**Анализ исследований и публикаций.** Обычно в таких случаях используют метод декомпозиции [4,5], разбивая задачу на систему более простых подзадач. Поскольку задача декомпозиции не формализована, то результат её решения в подобных случаях в значительной степени зависит от опыта проектировщика.

**Цель статьи** – совершенствование метода синтеза многофункциональных триггерных устройств, основанного на представлении и преобразовании функций алгебры логики в обобщённой форме и обеспечивающего сокращение времени разработки и снижение её стоимости.

**Метод решения.** Исходной позицией формального синтеза многофункциональных триггеров являются таблицы функционирования, число наборов которых равно  $2^{n+1}$ , где  $n$  – число управляющих сигналов (переменных), определяющих режимы настройки триггера.

При решении поставленной задачи будем ориентироваться на представление и минимизацию функций в сжатых картах с соседним кодированием [6].

Для иллюстрации приложения свойств обобщённых логических функций к синтезу многофункциональных триггеров рассмотрим два примера.

В качестве первого примера, основанного на представлении функций в форме ОЛФ с зависимыми параметрами, выполним синтез синхронного универсального триггера, имеющего пять режимов: хранения, синхронной установки в нулевое состояние, синхронной установки в единичное состояние, мультиплексной загрузки информации по заданному адресу и счёта. Условное изображение триггера показано на рис. 1.

Триггер имеет три адресных входа ( $A_2A_1A_0$ ), семь информационных входов ( $D_7-D_1$ ), вход разрешения режима счёта ( $T$ ), вход синхронной установки в единичное состояние ( $S$ ), вход приоритетной синхронной установки в нулевое состояние ( $R$ ) и вход подачи импульсов синхронизации ( $C$ ).

Режимы работы триггера:

- режим хранения ( $Q^{n+1} = Q^n$ ), если  $R = S = A_2 = A_1 = A_0 = T = 0$ ;
- режим синхронной приоритетной установки в нулевое состояние ( $Q^{n+1} = 0$ ), если  $R = 1$  (независимо от остальных сигналов);
- режим синхронной установки в единичное состояние ( $Q^{n+1} = 1$ ), если  $R = 0, S = 1$  (независимо от остальных сигналов);
- режим разрешения счёта ( $Q^{n+1} = \bar{Q}^n$ ), если  $R = S = A_2 = A_1 = A_0 = 0, T = 1$ ;
- режим мультиплексной загрузки с любого из семи информационных входов  $D_i$  ( $Q^{n+1} = D_i$ ), номер которого задаётся значением адресного слова  $A_2A_1A_0$ , отличным от нуля, при  $R = S = 0$ .

В соответствии с общим алгоритмом проектирования цифровых автоматов с памятью [5] представляем проектируемый триггер в форме однобитового автомата, состоящего из двух частей: памяти (собственно триггера) и комбинационной части (схемы управления), как показано на рис. 2.

Проектирование сводится к классической задаче синтеза однобитового автомата с памятью, выполненного на типовом  $D$ -триггере, при определении его функции возбуждения  $D'$ :

$$D' = f(R, S, T, A_2, A_1, A_0, D_7, D_6, D_5, D_4, D_3, D_2, D_1, Q^n).$$

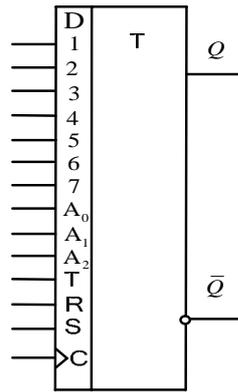


Рис. 1. Условное обозначение триггера

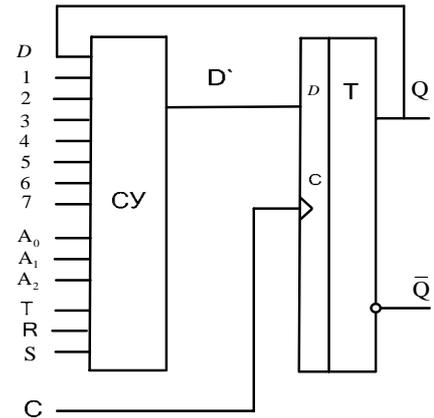


Рис. 2. Схема однобитового автомата

Попытка решить эту задачу, представляя функцию в таблице истинности, заведомо обречена на неудачу, поскольку таблица будет содержать  $2^{14}$  наборов.

Предлагается формализованное решение данной задачи с помощью представления функции  $D'$  в форме ОЛФ с трактовкой переменных  $R, S, T$  как первичных, а остальных – как вторичных. В качестве независимых параметров выбираем  $D_i, Q^n, \bar{Q}^n$ , а в качестве зависимых – минтермы, образуемые литералами адресных переменных ( $m_i = \tilde{A}_2 \tilde{A}_1 \tilde{A}_0$ ).

Как было отмечено выше, при  $R = S = 0$  режим работы зависит от значений адресных переменных (минтермов, определяемых этими переменными): нулевые значения определяют режим хранения  $Q^{n+1} = Q^n$  (при  $T = 0$ ) и счёта  $Q^{n+1} = \bar{Q}^n$  (при  $T = 1$ ), а ненулевые – режим мультиплексной записи информации  $Q^{n+1} = D_i$ . Следовательно, функцию  $D'$  в режиме хранения можем представить в виде произведения  $Q^n m_0$ , в режиме счёта –  $\bar{Q}^n m_0$ , в мультиплексном режиме –  $D_i m_i (i \neq 0)$ .

При  $R = 0, S = 1 - D' = 1$ , при  $R = 1 - D' = 0$ . Компактное представление режимов приведено в табл. 1.

Таблица 1

Таблица режимов настройки триггера

$R$	$S$	$T$	$D'$
0	0	0	$Qm_0 \vee D_1m_1 \vee D_2m_2 \vee D_3m_3 \vee D_4m_4 \vee D_5m_5 \vee D_6m_6 \vee D_7m_7$
0	0	1	$\bar{Q}m_0 \vee D_1m_1 \vee D_2m_2 \vee D_3m_3 \vee D_4m_4 \vee D_5m_5 \vee D_6m_6 \vee D_7m_7$
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	0

Представленные в таблице значения функции  $D'$  записываем в соответствующие клетки восьмизлементной карты с соседним кодированием по переменным  $R, S, T$ , как показано на рис. 3.

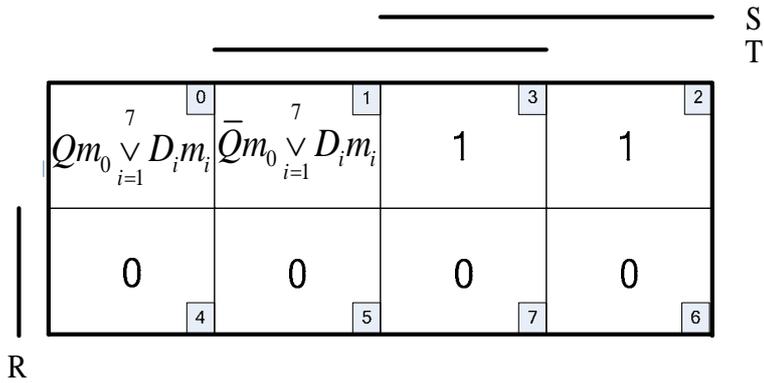


Рис. 3. Карта Карно функции возбуждения триггера

Опуская комментарий к процедуре выделения правильных конфигураций, записываем порядковые номера конфигураций, номера клеток, образующих эти конфигурации, и соответствующие им простые импликанты в виде произведения минтерма (логической суммы минтермов) на координату конфигурации [6,7]:

$$1 - \langle 0,2 \rangle: \bar{R}\bar{T}Qm_0; \quad 2 - \langle 1,3 \rangle: \bar{R}T\bar{Q}m_0; \quad 3 - \langle 0,1,2,3 \rangle: \bar{R}D_i m_i; \quad 4 - \langle 2,3 \rangle: \bar{R}S.$$

Соединяя полученные простые импликанты знаком логической суммы, записываем минимальную ДНФ функции  $D'$ , а также одну из скобочных форм:

$$D' = \bar{R}\bar{T}Qm_0 \vee \bar{R}T\bar{Q}m_0 \vee D_i m_i \bar{R} \vee \bar{R}S = \bar{R}[(Q \oplus T)m_0 \vee D_i m_i \vee S].$$

При реализации схемы на ПЛИС предпочтение следует отдать первой форме представления функции, а при реализации на микросхемах малого уровня интеграции – второй.

В качестве второго примера рассмотрим синтез многофункционального тестируемого триггера.

Вопросы тестирования особенно актуальны при построении автоматов на ПЛИС, поскольку доступ к отдельным элементам схемы ограничен.

Идея построения тестируемого цифрового автомата состоит в том, что его элементы памяти (D-триггеры) соединены в последовательную цепочку, образуя сдвигающий регистр, что даёт возможность подачи тестовой последовательности и её считывания [8].

Для построения таких автоматов требуется D-триггер, имеющий дополнительный информационный вход  $D_1$ , как показано на рис. 4.

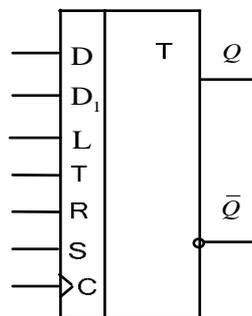


Рис. 4. Условное обозначение тестируемого триггера

Режимы работы триггера:

- режим хранения ( $Q^{n+1} = Q^n$ ), если  $R = S = L = T = 0$ ;

- режим синхронной установки в нулевое состояние ( $Q^{n+1} = 0$ ), если  $R = 1, S = 0$  (независимо от остальных сигналов);
- режим синхронной установки в единичное состояние ( $Q^{n+1} = 1$ ), если  $R = 0, S = 1$  (независимо от остальных сигналов);
- режим загрузки с альтернативного информационного входа  $D_1$  (режим тестирования  $Q^{n+1} = D_1$ ) при  $R = S = 1$  (независимо от остальных сигналов);
- режим загрузки с основного информационного входа  $D$  ( $Q^{n+1} = D$ ) при  $R = S = 0, L = 1$  (значение сигнала  $T$  несущественно);
- режим разрешения счёта ( $Q^{n+1} = \bar{Q}^n$ ), если  $R = S = L = 0, T = 1$ .

Представляем функцию  $Q^{n+1}$  в форме ОЛФ с независимыми параметрами, принимая за независимые переменные сигналы  $R, S, L$ , а остальные ( $Q^n, T, D, D_1$ ) – за параметры (таблица сократится до восьми строк, как показано в табл. 2).

**Таблица 2**  
Таблица режимов настройки триггера

$S$	$R$	$L$	$D' = Q^{n+1}$
0	0	0	$T \oplus Q^n$
0	0	1	$D$
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	$D_1$
1	1	1	$D_1$

Представленные в таблице значения  $D' = Q^{n+1}$  записываем в клетки восьми-элементной карты с соседним кодированием по переменным  $R, S, L$  (рис. 5).

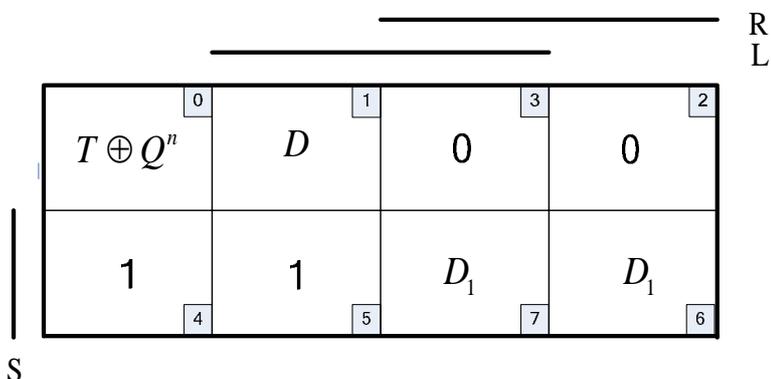


Рис. 5. Карта Карно функции возбуждения триггера

Опуская комментарий к процедуре выделения правильных конфигураций, записываем порядковые номера конфигураций, номера клеток, образующих эти конфигурации, и соответствующие им простые импликанты в виде произведения координаты этой конфигурации на значение функции в ней:

$$1- \langle 0,4 \rangle: \bar{R}\bar{L}(T \oplus Q^n); \quad 2- \langle 1,5 \rangle: \bar{R}LD; \quad 3- \langle 4,5,6,7 \rangle: SD_1; \quad 4- \langle 4,5 \rangle: S\bar{R}.$$

Соединяя полученные простые импликанты знаком логической суммы, записываем минимальную ДНФ функции  $D'$ , а также одну из скобочных форм:

$$D' = \bar{R}\bar{L}\bar{T}Q^n \vee \bar{R}\bar{L}\bar{T}\bar{Q}^n \vee \bar{R}LD \vee \bar{R}S \vee SD_1 = \bar{R}[\bar{L}(Q \oplus T) \vee LD \vee S] \vee SD_1.$$

При реализации схемы на ПЛИС предпочтение следует отдать первой форме представления функции, а при реализации на микросхемах малого уровня интеграции – второй.

## Выводы

Предложен метод синтеза многофункциональных триггеров, основанный на разбиении множества всех переменных, определяющих режимы работы триггера, на два подмножества, одно из которых трактуем как первичные переменные, определяющие координаты точек новой области определения на данном разбиении, а второе – как вторичные, определяющие значения функции в точках новой области определения, с последующим представлением и минимизацией функций в сжатых картах с соседним кодированием. Предложенный метод позволил сократить число точек области определения функций возбуждения триггера, что упростило процедуру синтеза в целом.

## Список литературы

1. Соловьёв В.В. Проектирование цифровых систем на основе программируемых логических интегральных схем. – М.: Горячая линия – Телеком, 2001. – 636 с.
2. Рубанов В.Г., Коробкова Е.Н. Разработка алгоритма сжатия области определения логических функций // Труды современного гуманитарного университета. Белгородский филиал. – Белгород: БФ СГУ. – 2000. – Вып. 18. – С. 105 -112.
3. Коробкова Е.Н., Шпак А.С. К программной реализации алгоритма сжатия области определения логических функций // Системы обработки информации. – Х.: ХВУ. – 2006. – Вып. 2(51). – С. 51-61.
4. Потёмкин И.С. Функциональные узлы цифровой автоматики. – М.: Энергоатомиздат, 1988. – 320 с.
5. Закревский А.Д. Алгоритмы синтеза дискретных автоматов. – М.: Наука, 1971. – 512 с.
6. Коробкова Е.Н. Графоаналитический метод минимизации полностью определённых логических функций в сжатых картах // Системы обработки информации. – Х.: НАНУ, ПАНМ, ХВУ. – 2002. – Вып. 6(22). – С. 288 - 298.
7. Рубанов В.Г., Коробкова Е.Н. Графоаналитический метод нахождения минимальных дизъюнктивных нормальных форм обобщённых логических функций // Системы обработки информации. – Х.: НАНУ, ПАНМ, ХВУ. – 2002. – Вып. 3(19). – С. 46-53.
8. Уэйкерли Дж.Ф. Проектирование цифровых устройств. В 2-х т. – М.: Постмаркет, 2002. – Т.1 – 544 с. Т.2 – 528 с.