УДК 519.713.1

ПРОЕКТИРОВАНИЕ ДЕТЕРМИНИРОВАННОГО КОНЕЧНОГО АВТОМАТА

Долгушев Ю.В., Кулецкий С.В., Чигарев В.А.

Белорусский национальный технический университет, Минск

Введение. Не все окружающие нас преобразователи информации выполняют функциональное отображение информации. Результат преобразования вход/выход зачастую зависит не только от того, какая информация в данный момент появилась на входе, но и от того, что происходило раньше, от предыстории преобразования. Огромное множество примеров тому дают биологические системы: например, один и тот же вход - извинение после того, как вам наступили на ногу в переполненном трамвае - вызовет у вас одну реакцию в первый раз и совсем другую в пятый раз, если вам постоянно наступают на ногу и извиняются. Таким образом, существуют более сложные, не функциональные преобразователи информации, реакция их зависит не только от входа в данный момент, но и от входной истории. Такие преобразователи называются автоматами.

Рассмотрим исследуемую задачу*:* Опишем поведение родителя, отправившего сына в школу. Сын приносит двойки и пятерки. Отец не хочет хвататься за ремень каждый раз, как только сын получает очередную двойку, и выбирает более тонкую тактику воспитания. Задавать автомат удобно графом, в котором вершины соответствуют состояниям, а ребро из состояния S в состояние q, помеченное x/y, проводиться тогда, когда автомат состояния S под воздействием входного сигнала x переходит в состояние q с выходной реакцией y. Граф автомата, моделирующего умное поведение родителя, представлен на рис.1.

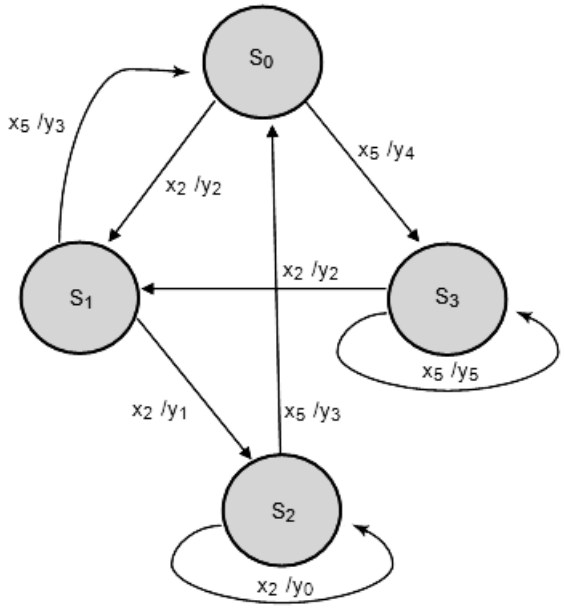

Рис. 1. Граф автомата, описывающий поведение «умного» отца

Здесь конечный автомат изображен графом с вершинами, соответствующими состояниям. Ребра графа соответствуют переходам между состояниями. Ребро помечено входным сигналом, под воздействием которого этот переход происходит, и выходным сигналом, который вырабатывает автомат при получении этого входного сигнала в те-

кущем состоянии. Автомат, моделирующий родителя, имеет четыре состояния $\{S_0, ... ,S_3\}$, и два входных сигнала - оценки, полученные сыном в школе: $\{x_2, x_5\}$. Начиная с начального состояния S_0 (оно помечено входной стрелкой), автомат под воздействием входных сигналов переходит из одного состояния в другое и выдает выходные сигналы - реакции на входы. Выходы автомата $\{y_0, ... , y_5\}$ будем интерпретировать как действия родителя так:

- y_0 – брать ремень;
- y_1 – ругать сына;
- y_2 – успокаивать сына;
- y_3 – надеяться;
- y_4 – радоваться;
- y_5 – ликовать.

Рассмотрим два вида реализации КА: программную и аппаратную.

Программную реализацию можно выполнить на любом языке высокого уровня разными способами. На рис.2 представлена блок-схема программы, реализующей поведение автомата. Нетрудно увидеть, что топология блок-схемы программы повторяет топологию графа переходов конечного автомата. С каждым состоянием связана операция NEXT, выполняющая функцию ожидания очередного события прихода нового входного сигнала и чтение его в некоторый стандартный буфер x, а также последующий анализ того, какой это сигнал. В зависимости от того, какой сигнал пришел на вход, выполняется та или иная функция y_0-y_5 и происходит переход к новому состоянию.

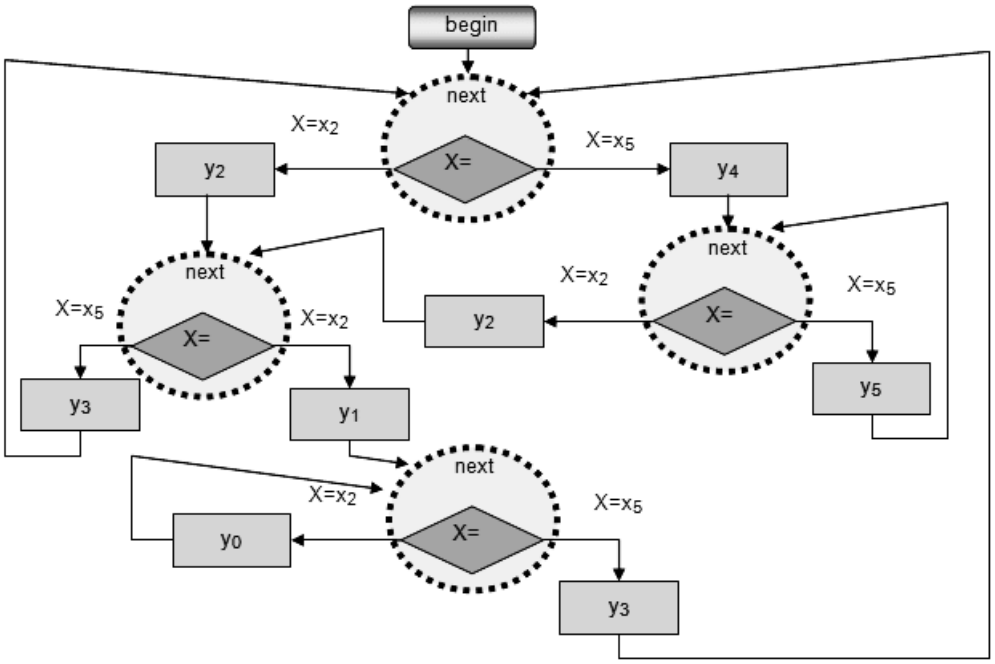

Рис. 2 Схема программы, реализующей поведение конечного автомат.

Необходимо так же подобрать структуру данных, которая содержит в себе информацию о конечном автомате, хранить информацию о различных переходах из одного состояние в другое. Один из вариантов – использовать «хэш-таблицу» с полями «ключ – значение» (key-value). «Ключ» в свою очередь содержит возможные входные параметры, сигналы. Когда «значение» содержит себе необходимую информацию о выходных параметрах, такие как «событие», значение нового состояния графа.

В качестве входных параметров используется текущее состояние конечного автомата, и отметка, которую принес сын. А в качестве выходных – реакция отца, и значение следующего конечного автомата, см. Таблицу 1.

Таблица 1 – Перечисление значений и полей хэш-таблицы

Key	Value
0-2	Успокаивание - 1
0-5	Эйфория - 3
1-2	Словесное наказание - 2
1-5	Надежда - 0
2-2	Наказание ремнем - 2
2-5	Надежда - 0
3-2	Успокаивание - 1
3-5	Эйфория - 3

Аппаратная реализация требует построения устройств памяти для запоминания текущего состояния автомата. Обычно на практике используют двоичные элементы памяти (триггеры), запоминающие значение только одного двоичного разряда. Функциональный блок автомата реализуется как конечный функциональный преобразователь. Таким образом, общий подход к аппаратной реализации конечного автомата такой:

- входные и выходные сигналы и внутренние состояния автомата кодируются двоичными кодами;
- по таблицам переходов и выходов составляются кодированные таблицы переходов и выходов - фактически, табличное задание отображения F рис.1;
- по кодированным таблицам переходов и выходов проводится минимизация двоичных функций и они реализуются в заданном базисе;
- решаются схемотехнические вопросы синхронизации - привязки моментов выдачи выходного сигнала и изменения состояния внутренней памяти к моментам поступления входных сигналов на вход автомата.

Примером схемы для решения данной задачи будет рис. 3.

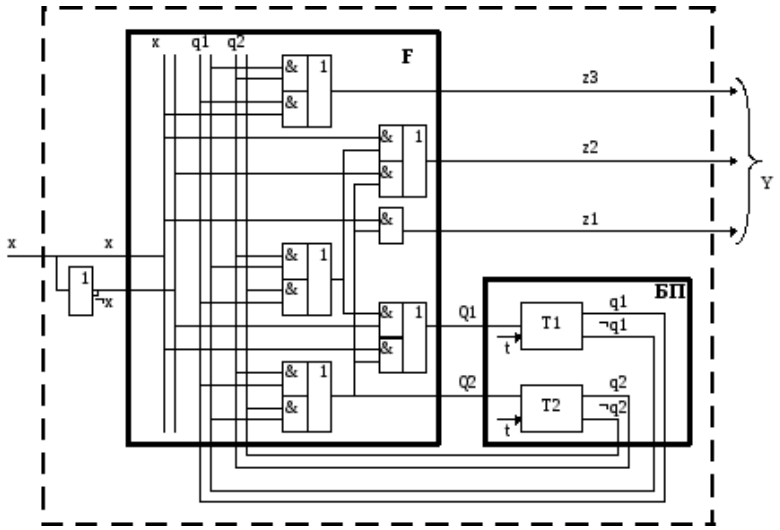

Рис. 3. Функциональная схема. Выделены функциональный блок и блок памяти автомата

Один двоичный разряд *x* кодирует два входных сигнала, пары двоичных разрядов q1, q2; Q1, Q2 кодируют соответственно текущее и следующее состояния, разряды z1, z2, z3 кодируют выходной сигнал. Блоки T1 и T2 - триггеры, которые запоминают двоичный сигнал до прихода следующего. Вход t в триггере - синхронизационный вход, разрешающий переключение триггера. Сигнал на этом входе появляется в момент наступления события получения автоматом очередного входного сигнала от окружения.

ЛИТЕРАТУРА

1. *Поликарпова Н.И., Шалыто А.А. Автоматное программирование. - СПб.:Питер, 2009.-176с.*
2. *Трахтенброт Б.А., Бардзинь Я.М. Конечные автоматы. Поведение и синтез. - М.: Наука, 1970.*
3. *Минский М. Вычисления и автоматы. - М.: Мир, 1971.*
4. *Шалыто А. А., Антипов В. В. Алгоритмизация и программирование задач логического управления. - СПб.: Моринтех, 1996.*