

**Современная наука и образование:
новые подходы и актуальные исследования**

Мендель Виктор Васильевич,

кандидат физ.-мат. наук,

доцент кафедры математики и информационных технологий,

ФГБОУ ВО «Тихоокеанский государственный университет»;

Безденежных Никита Владимирович,

студент,

ФГБОУ ВО «Тихоокеанский государственный университет»,

г. Хабаровск

**СИНТЕЗ КАНОНИЧЕСКИХ УРАВНЕНИЙ КОНЕЧНОГО АВТОМАТА,
ЗАДАННОГО ТАБЛИЦАМИ ФУНКЦИЙ ПЕРЕХОДОВ И ВЫВОДА**

Аннотация. Представлен алгоритм построения системы булевых функций, являющихся каноническими уравнениями произвольного конечного автомата Мили. Изначально предполагается, что конечный автомат задан функциями вывода и внутренних переходов, представленными с помощью таблиц на произвольных конечных алфавитах. Для получения канонических уравнений используется приложение, созданное авторами для пакета компьютерной математики SCILAB.

Ключевые слова: конечный автомат, канонические уравнения, SCILAB.

В этой статье описан процесс построения системы канонических уравнений (булевых функций) для конечного автомата с использованием приложения, написанного для пакета Scilab 6.1.0 (официальный сайт продукта: <https://scilab.org>, с основами работы можно познакомиться в [1]), являющегося свободно распространяемым программным продуктом. Канонические уравнения необходимы для построения функциональной схемы и реального микропроцессора, управляющего работой автомата.

Постановка задачи. Процесс создания реального автоматического устройства состоит из нескольких этапов (смотри, например, [1] или [2]). На первом этапе определяются варианты входных сигналов и реакций. На этой основе

Современная наука и образование: новые подходы и актуальные исследования

формируются первичные входной и выходной алфавиты. На втором этапе описывается поведение автомата в конкретных ситуациях. На основе этого описания на третьем этапе определяются внутренние состояния автомата и строятся таблицы функций, описывающих его поведение. В этих таблицах описывается, как за такт работы автомата по текущей комбинации «входной сигнал-внутреннее состояние» генерируется реакция – выходной сигнал и новое внутреннее состояние.

Следующие несколько этапов предназначены для описания по имеющимся неформальным данным микропроцессорного устройства (схемы из функциональных элементов) с использованием стандартного бинарного кода. Для этого на четвертом этапе буквы входного, выходного алфавитов и алфавита внутренних состояний кодируются стандартным равномерным бинарным кодом. На пятом этапе строится неполная сводная таблица булевых функций. Количество переменных в этой таблице равно сумме длин бинарных кодов входного алфавита и алфавита внутренних состояний, а количество функций – сумме длин бинарных кодов алфавита внутренних состояний и выходного алфавита. На шестом этапе происходит дополнение сводной таблицы строками так, чтобы в ней отображались все возможные наборы входных значений. Результатом являются таблицы значений булевых функций, входящих в набор канонических уравнений конечного автомата. Имея канонический набор можно дальше проектировать функциональную схему и реальный микропроцессор, управляющий работой автомата.

Почему SCILAB? Выбор этого пакета объясняется несколькими причинами. Во-первых, это современный, высокопроизводительный и, что немаловажно, бесплатный программный продукт. Во-вторых, в нем реализована возможность обработки массивов или их фрагментов без использования операторов циклов. В-третьих, SCILAB имеет богатый набор операторов и функций, позволяющих быстро преобразовывать типы данных, выполнять логические операции

Современная наука и образование: новые подходы и актуальные исследования

и многое другое. Ниже мы рассмотрим код приложения прокомментируем некоторые из используемых операторов.

Приложение для SCILAB (комментарии даны курсивом)

Заголовок приложения

//ГЕНЕРАТОР КАНОНИЧЕСКИХ (ИЗ БУЛЕВЫХ ФУНКЦИЙ) УРАВНЕНИЙ КОНЕЧНОГО АВТОМАТА

clc //очистка консоли

Информация о размерностях изначально заданных алфавитов

//1. Определение исходных данных

//1.1 размерность входного, внутреннего и выходного алфавитов

n1=4; n2=5; n3=9

//1.2 длина бинарного кода для кодирования этих алфавитов

Длина равномерного бинарного кода – минимальное количество бит, которые нужны для двоичной кодировки алфавита, вычисляется как округленный с избытком логарифм по основанию 2 от количества букв алфавита (мощности)

k1=ceil(log2(n1)); k2=ceil(log2(n2)); k3=ceil(log2(n3))

//1.3 короткая запись значений функции переходов и функции вывода

длины строк равны n1*n2

Предполагается, что аргументы в таблице заданы следующим образом: для первой буквы входного алфавита последовательно подставляются все буквы алфавита внутренних состояний, потом это повторяется для второй буквы входного алфавита и так далее. Для упрощения ввода значения аргументов опущены, приводятся только значения функций.

mu=[8 0 4 4 5 8 5 1 7 3 0 6 4 1 2 0 3 5 1 3]//функция вывода

lmd=[4 0 3 2 1 1 0 2 2 4 1 0 1 3 0 1 0 3 1 2]//функция внутренних переходов

//2 построение полной таблицы функций перехода и вывода в стандартных алфавитах

Современная наука и образование: новые подходы и актуальные исследования

Здесь автоматически восстанавливаются столбцы аргументов таблицы, определяющей функции автомата.

```
//2.1 генерация всех вариантов аргументов в 1 и 2 столбцах
// первые аргументы - буквы входного алфавита
ishod=zeros(n1*n2,4)
ishod(:,1)=modulo((0:n1*n2-1),n1)
//вторые аргументы - буквы алфавита внутренних состояний
for i=1:n2
ishod(1+(i-1)*n1:i*n1,2)=ones(n1,1)*(i-1)
end
//2.2 копирование значений функций в 3 и 4 столбцы
ishod(:,3)=lmd
ishod(:,4)=mu
```

//3 построение расширенной таблицы значений функций перехода и вывода для расширенных алфавитов, количество букв в которых кратно степени двойки

Начинается построение сводной таблицы канонических уравнений – булевых функций, для этого буквы исходных алфавитов заменяются на бинарный код.

```
//3.1 генерация значений аргументов в 1 и 2 столбцах
//первые аргументы
svod=zeros(2^(k1+k2),4)
svod(:,1)=modulo((0:2^(k1+k2)-1),2^k1)
//вторые аргументы
for i=1:2^k2
svod((1+(i-1)*2^k1):i*2^k1,2)=ones(2^k1,1)*(i-1)
end
//3.2 копирование значений функций из массива ishod
```

**Современная наука и образование:
новые подходы и актуальные исследования**

```
 //(только для заданных диапазонов из 2.2)
```

```
 for i=1:n2
```

```
  j1=(i-1)*(2^k1)+1
```

```
  j2=j1+n1-1
```

```
  jj1=(i-1)*n1+1
```

```
  jj2=jj1+n1-1
```

```
  svod(j1:j2,[3 4])=ishod(jj1:jj2,[3 4])
```

```
 end
```

```
 //3.3 сортировка в порядке возрастания первого аргумента
```

```
 //(необходима для корректного задания переменных до конвертации в би-  
нарный код)
```

```
 svod=gsort(svod, "lr", "i")
```

```
 //3.4 построение главной части системы булевых функций
```

```
 //реализующих бинарный код функции переходов
```

```
 LMD=(dec2bin(svod(:,3),k2))
```

```
 LMD=asciimat(LMD)-48
```

```
 //3.5 построение главной части системы булевых функций
```

```
 //реализующих бинарный код функции вывода
```

```
 MU=(dec2bin(svod(:,4),k3))
```

```
 MU=asciimat(MU)-48
```

```
 //4 Вывод на экран результатов вычислений
```

```
 disp('ГЛАВНАЯ ЧАСТЬ БИНАРНЫХ ФУНКЦИЙ ПЕРЕХОДА')
```

```
 disp(LMD')
```

```
 disp('ГЛАВНАЯ ЧАСТЬ БИНАРНЫХ ФУНКЦИЙ ВЫВОДА')
```

```
 disp(MU')
```

Результаты работы приложения

ГЛАВНАЯ ЧАСТЬ БИНАРНЫХ ФУНКЦИЙ ПЕРЕХОДА

```
 Q1=[1.0.0.0.0.0.0.0.0.1.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.]
```

**Современная наука и образование:
новые подходы и актуальные исследования**

$Q_2=[0.0.1.0.0.0.0.0.0.0.0.1.1.0.0.0.1.0.0.0.0.0.0.0.1.1.0.0.1.0.0.0.]$

$Q_3=[0.1.0.1.0.0.0.0.0.1.0.1.1.0.0.0.1.0.1.0.1.0.0.0.0.0.0.1.0.0.0.0.]$

ГЛАВНАЯ ЧАСТЬ БИНАРНЫХ ФУНКЦИЙ ВЫВОДА

$V_1=[1.0.0.0.0.0.0.0.0.1.0.]$

$V_2=[0.1.1.1.0.0.0.0.0.0.0.0.0.1.0.0.0.1.1.0.0.0.0.0.0.1.0.1.0.0.0.0.]$

$V_3=[0.0.1.0.1.0.0.0.0.0.1.0.0.0.0.0.0.0.0.0.1.0.0.0.0.0.0.1.0.1.0.0.0.]$

$V_4=[0.1.1.0.1.0.0.0.0.0.1.1.1.0.0.0.0.1.0.0.1.0.0.0.0.1.0.0.1.0.0.0.]$

Заключение. Как видно из примера, с помощью небольшого по объему приложения удалось автоматически сгенерировать семь булевых функций, зависящих от пяти аргументов, каждая из которых принимает 32 значения. Эти функции и образуют систему канонических уравнений автомата. Небольшие мощности исходных алфавитов выбраны только для того, чтобы компактно продемонстрировать результаты работы приложения. В реальной ситуации могут быть использованы алфавиты с существенно большим количеством букв. В этом случае получение результата ограничено только размером памяти, установленной на используемом для вычислений компьютере.

СПИСОК ЛИТЕРАТУРЫ

- 1. Scilab: Решение инженерных и математических задач / Е.Р. Алексеев, О.В. Чеснокова, Е.А. Рудченко. – М.: ALT Linux; БИНОМ. Лаборатория знаний, 2008. – 260 с.*
- 2. Алешин С.В., Кудрявцев В.Б., Подколзин А.С. Введение в теорию автоматов/ С.В. Алешин, В.Б. Кудрявцев, А.С. Подколзин. – М.: Наука, 1985. – 320 с.*
- 3. Бувич В.А., Каландарашвили Н.Г., Таль А.А. Об описании конечного автомата с помощью конечного множества вход-выходных слов / В.А. Бувич, Н.Г. Каландарашвили, А.А. Таль // Автоматика и телемеханика. 1970. – № 1. – С. 112-122*