

# ПРОЕКТИРОВАНИЕ НА ПЛИС УСТРОЙСТВ ЛОГИЧЕСКОГО УПРАВЛЕНИЯ ДЛЯ СИСТЕМ ЦИФРОВОЙ ОБРАБОТКИ СИГНАЛОВ

Соловьев В.В., Климович А.

БГУИР, Минск (Республика Беларусь), e-mail: [valsol@mail.ru](mailto:valsol@mail.ru),  
Белостокский политехнический университет (Польша), [aklim@ii.pb.bialystok.pl](mailto:aklim@ii.pb.bialystok.pl)

## 1. Введение

Одним из важнейших вопросов при проектировании систем цифровой обработки сигналов (ЦОС) является разработка устройств логического управления (УЛУ). В настоящее время программируемые логические интегральные схемы (ПЛИС) широко используются в качестве элементной базы для построения систем ЦОС. Поэтому актуальной является задача проектирования на ПЛИС УЛУ для систем ЦОС. К сигналам управления также относятся сигналы синхронизации, которые в значительной мере влияют на производительность системы ЦОС.

В общем случае структуру системы ЦОС можно представить в виде совокупности подсистем  $E_1, \dots, E_F$ , общего УЛУ и генератора синхросигналов СГ. В данной работе рассматривается подход к проектированию УЛУ сложной системы ЦОС, состоящей из отдельных подсистем, который включает синтез на ПЛИС отдельных УЛУ для каждой подсистемы ЦОС, синтез общего УЛУ, а также синтез системного генератора синхросигналов.

## 2. Синтез УЛУ отдельных подсистем

Пусть алгоритм логического управления каждой подсистемы  $E_f$  описывается на языке граф-схем алгоритмов (ГСА)  $G_f, f = \overline{1, F}$ . Проектирование некоторого УЛУ подсистемы  $E_f$  сводится к синтезу микропрограммного автомата (МПА)  $S_f$ , реализующего ГСА  $G_f, f = \overline{1, F}$ . Для синтеза некоторого МПА  $S_f, f = \overline{1, F}$ , согласно [1] все операторные вершины отмечаются метками  $a_1, \dots, a_m$ , которые рассматриваются в качестве внутренних состояний МПА.

Традиционно в операторных вершинах ГСА записываются выходные переменные МПА, единичные значения которых вызывают выполнение одноименных микроопераций, при этом по умолчанию принимается, что значения остальных выходных переменных равны нулю. В то же время анализ взаимодействия операционного и управляющего устройств позволяет для каждой операторной вершины ГСА указать множество микроопераций, которые в данной операторной вершине могут принимать как единичные, так и нулевые значения. Из анализа взаимодействия операционного и управляющего устройства в каждой операторной вершине ГСА, помеченной меткой  $a_m, m = \overline{1, M}$ , записывается два множества микроопераций:  $Y^1(a_m)$ , принимающие единичные значения; и  $Y(a_m)$ , которые могут принимать произвольные значения. В качестве математической модели МПА  $S_f, f = \overline{1, F}$ , используется автомат класса С [2], синтез которого сводится к решению задачи кодирования внутренних состояний.

Для кодирования внутренних состояний МПА  $S_f, f = \overline{1, F}$ , строится булева матрица  $B$ , столбцы которой поставлены в соответствие состояниям  $a_1, \dots, a_m$ , а строки - возможным наборам значений двоичных переменных  $d_1, \dots, d_R$ , значения которых могут выступать в качестве кодов внутренних состояний МПА. На пересечении строки  $i$  и столбца  $j$  матрицы  $B$  ставится единица, если двоичный набор, соответствующий строке  $i$ , может являться кодом состояния  $a_j$ , т.е. двоичный набор содержит единицы на месте выходных переменных множества  $Y^1(a_j)$ , любые значения на месте безразличных выходных переменных множества  $Y(a_j)$  и нули в остальных позициях. Необходимо каждому столбцу матрицы  $B$  поставить в соответствие некоторую строку, имеющую единицу в этом столбце, причем различным столбцам должны соответствовать различные строки. Последняя задача сводится к разновидности задачи о назначениях и для ее решения может быть предложен следующий алгоритм.

1. Полагается  $R:=0$ .

2. Строится матрица  $B$ .

3. Если число строк матрицы  $B$  меньше числа столбцов, то полагается  $R:=R+1$ , вводится дополнительная переменная  $d_R$  и осуществляется переход к пункту 2; иначе выполняется переход к пункту 4.

4. В матрице  $B$  выбирается строка  $i$  и столбец  $j$ , на пересечении которых находится единица и которые содержат наименьшее суммарное число единиц. Если подобных пар строк и столбцов может быть выбрано несколько, выбирается пара, двоичный набор для которой содержит минимальное число единиц.

5. Двоичный набор, соответствующий строке  $i$ , назначается в качестве кода состоянию  $a_j$ . Из дальнейшего рассмотрения в матрице  $B$  исключаются строка  $i$  и столбец  $j$ , а также, возможно, появившиеся нулевые строки.

Пункты 4 и 5 алгоритма повторяются до тех пор, пока каждому столбцу матрицы  $B$  не будет поставлена в соответствие некоторая строка, либо пока из матрицы  $B$  не будут исключены все строки.

6. Если все состояния МПА закодировать не удалось (т.е. не каждому столбцу матрицы  $B$  поставлена в соответствие некоторая строка), то полагается  $R:=R+1$ , вводится дополнительная переменная  $d_R$  и осуществляется переход к пункту 2; иначе выполняется переход к пункту 7.

7. Конец.

### 3. Синтез общего УЛУ

Общее УЛУ всей системы ЦОС представляет собой МПА  $S_0$ , который управляет совокупностью слабо взаимодействующих параллельных процессов. Функционирование всей системы начинается с работы МПА  $S_0$ , при этом остальные МПА находятся в состоянии ожидания, вырабатывая сигналы готовности множества  $H = \{h_1, \dots, h_F\}$ . Для запуска одного или нескольких МПА нижнего уровня МПА  $S_0$  формирует иницирующие сигналы множества  $Z = \{z_1, \dots, z_F\}$ . Во время активности некоторого автомата  $S_f$ ,  $f = \overline{1}, \overline{F}$ , сигнал готовности  $h_f$  сбрасывается и вновь устанавливается при возврате  $S_f$  в исходное состояние ожидания. Функционирование автомата  $S_0$  представляется обычной ГСА, в которой сигналы множества  $H$  выступают в качестве логических условий, а сигналы множества  $Z$  - в качестве микроопераций. Одновременный запуск нескольких автоматов нижнего уровня описывается с помощью операторной вершины, в которую записываются соответствующие сигналы множества  $Z$ . Ожидание окончания функционирования нескольких автоматов описывается с помощью условной вершины, в которой записывается логическая функция, представляющая собой элементарную конъюнкцию соответствующих переменных множества  $H$ . Для синтеза МПА  $S_0$  может быть применен метод [3].

### 4. Синтез системного генератора

Пусть  $C_f$  - длительность такта синхросигнала МПА  $S_f$  (подсистемы  $E_f$ ),  $f = \overline{1}, \overline{F}$ , выраженная в числе тактов опорного синхросигнала CLK. Для каждого МПА  $S_f$  системный генератор формирует сигнал  $g_f$  с периодом  $C_f$ , который используется в качестве сигнала синхронизации памяти МПА  $S_f$ ,  $f = \overline{1}, \overline{F}$ . Работу системного генератора можно представить в виде циклического автомата Мура с числом состояний  $\max(C_f)$ ,  $f = \overline{1}, \overline{F}$ . Сигнал  $g_f$  вырабатывается в состоянии автомата  $a_i$ , если  $C_f = i$ . Для синтеза системного генератора может быть использован рассмотренный выше алгоритм при условии, что все множества  $Y(a_m)$  являются пустыми множествами.

В случае, когда длительности  $C_1, \dots, C_F$  тактов синхросигналов выражаются в единицах астрономического времени, необходимо применить метод работы [5] для выражения каждого  $C_f$ ,  $f = \overline{1}, \overline{F}$ , в числе тактов опорного синхросигнала CLK, а также для определения длительности такта опорного синхросигнала.

### 5. Заключение

Отметим, что все предлагаемые методы синтеза МПА и системного генератора ориентированы на архитектуру ПЛИС. В качестве элементов памяти МПА  $S_1, \dots, S_F$  используются триггеры выходных макроячеек ПЛИС. Каждый отдельный МПА  $S_f$ ,  $f = \overline{1}, \overline{F}$ , преимущественно реализуется на одном функциональном блоке ПЛИС. Для связи между МПА  $S_1, \dots, S_F$  между собой, а также с общим УЛУ используется матрица межсоединений ПЛИС [4].

Работа выполнена при финансовой поддержке гранта W/II/4/99 Белостокского политехнического университета (Польша).

### Литература

1. Баранов С.И. Синтез микропрограммных автоматов. - Ленинград: Энергия, 1979. - 232 с.
2. Соловьев В.В. Синтез микропрограммных автоматов на программируемых матрицах логики // Весці Акадэміі Навук Беларусі. Сер. физ.-техн. наук. -1994. -№ 1. - с. 68-72.
3. Соловьев В.В. Реализация на программируемых матрицах логики параллельных алгоритмов логического управления // Управляющие системы и машины. - 1995. -№ 6. - с.24-31.
4. Соловьев В.В., Васильев А.Г. Программируемые логические интегральные схемы и их применение. - Минск: Беларуская навука, 1998. - 270 с.
5. Соловьев В.В. Проектирование цифровых систем на основе программируемых логических интегральных схем. - Москва: Горячая линия - Телеком, 2001. - 636 с.