

УДК 621.396

Е.Е. Малафеев, Е.Е. Малафеев

ОАО «АО Научно-исследовательский институт радиотехнических измерений», Харьков

ДИНАМИЧЕСКИЕ АЛГОРИТМИЧЕСКИЕ ПРЕОБРАЗОВАТЕЛИ

Исследованы виды настраиваемых алгоритмических преобразователей. Рассмотрены принципы построения динамических алгоритмических преобразователей и особенности их работы. Сформулирована задача синтеза динамических алгоритмических преобразователей и предложена процедура их построения.

Ключевые слова: алгоритмические преобразователи, динамические алгоритмические преобразователи, настраиваемые алгоритмические преобразователи.

Введение

Постановка проблемы. Одним из перспективных направлений повышения эффективности устройств обработки информации является унификация алгоритмических, программных и аппаратных средств, т.е. создание универсальных в заданном классе средств, реализующих при соответственном преобразовании заданное множество типовых решений [1].

Вопросы унификации и типизации алгоритмических средств имеют первостепенное значение при решении различных задач разработки автоматизированных систем обработки информации и управления, т.к. позволяют снизить общее их число, что упрощает организацию адаптивного комплекса алгоритмов [2]. В связи с этим возникает задача разработки универсальных алгоритмических преобразователей.

Анализ исследований и публикаций. Методам разработки универсальных алгоритмических преобразователей для логических схем посвящены работы В.Л. Артюхова, Г.Н. Копейкина [3], А.А. Шалыто [4], С.В. Шидловского [5] и др.

Построение и рациональное использование настраиваемых логических модулей из функциональных элементов рассмотрены в работе [3]. Найдены оценки сложности схемных реализаций. Предложенный подход распространен на цифровые интегральные схемы и релейно-контактные элементы, серийно выпускаемые промышленностью.

Методы аппаратной и программной реализации алгоритмов логического управления, заданных булевыми функциями разных классов, булевыми формулами и графами переходов рассмотрены в работе [4]. Предложены методы синтеза схем из априори известных произвольных элементов. Разработаны методы построения схем алгоритмов, обладающих различными свойствами.

Изложены основные положения SWITCH-технологии, предназначенной для алгоритмизации и программирования задач логического управления,

при использовании промышленных компьютеров и программируемых логических контроллеров.

В работе [5] рассматривается автоматное программирование – подход к разработке программных систем со сложным поведением, основанный на модели автоматизированного объекта управления (расширении конечного автомата). Автомат с памятью рассматривается в качестве многофункционального модуля, настраиваемого состояниями на реализацию в определенной последовательности различных ортогональных систем булевых формул, описывающих комбинационные схемы и зависящих от различных групп входных переменных.

Алгоритмы и способы создания управляющих систем, построенных на базе логических структур и обладающих высокой надежностью за счет изменения внутренней структуры, предложены в работе [6]. На основе многофункциональных логических модулей решены проблемы вычисления систем булевых формул определенных классов. Отражены вопросы автоматизированного синтеза и исследования функциональных свойств многофункциональных логических модулей и построенных на их основе изотропных и квазиизотропных сред с помощью разработанной логической системы имитационного моделирования Cell System.

Рассмотренные выше методы ориентированы на создание универсальных логических элементов, что ограничивает область их применения.

Целью статьи является разработка нового подхода к созданию настраиваемых алгоритмических преобразователей.

Изложение основного материала исследования

Одним из перспективных методов построения универсальных алгоритмических средств является создание настраиваемых алгоритмических преобразователей (НАП) [6]. В общем случае, НАП можно представить в виде структуры, приведенной на рис. 1. НАП состоит из формирователя настроек и реализатора базового алгоритма (РБА).

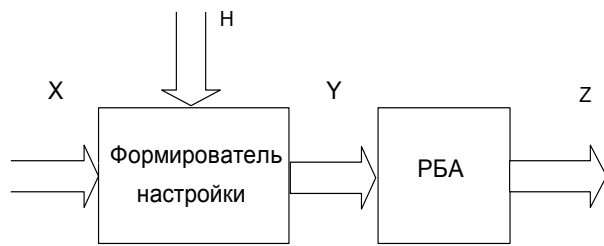


Рис. 1. Структура настраиваемого алгоритмического преобразователя

Формирователь настроек производит преобразование множества входных сигналов X в зависимости от настройки H в множество выходных сигналов Y , которые поступают на входы РБА.

Под настройкой понимают подстановку в качестве сигналов Y значений из множества $\{0, 1, X_1, \bar{X}_1, \dots, X_n, \bar{X}_n\}$. В результате настройки базовый алгоритм преобразуется в соответствующий частный (производный) алгоритм от меньшего числа переменных.

Разновидностью настраиваемых алгоритмических преобразователей являются динамические алгоритмические преобразователи – (ДАП) – программно-аппаратные средства, реализующие заданный алгоритм при различных преобразованиях операндов. Обобщенная структура динамического алгоритмического преобразователя приведена на рис. 2.

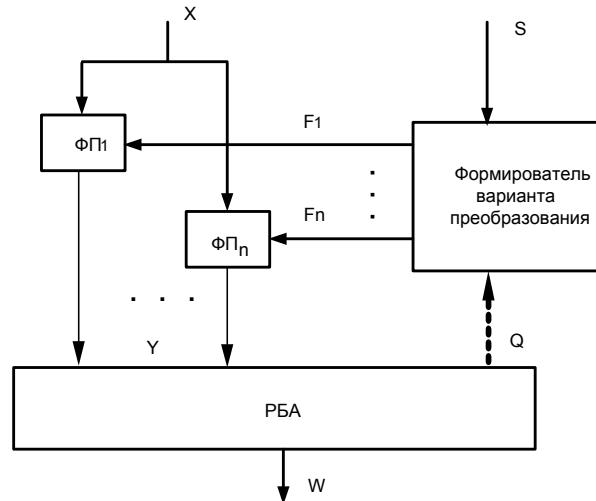


Рис. 2. Структура динамического алгоритмического преобразователя

Функциональные преобразователи на основании заданного варианта преобразования преобразуют множество входных сигналов (переменных) X в соответствующие логические функции Y , поступающие на РБА. Вариант преобразования формируется на основании управляющих сигналов S и (в ряде случаев) специальных сигналов Q , формируемых ДАП. Выбор структуры ДАП зависит от вида реализуемых алгоритмов. В результате преобразо-

ваний базовый алгоритм, реализуемый РБА, преобразуется в некоторый алгоритм от меньшего количества переменных.

Несмотря на общность подхода к реализации алгоритмов, у динамических алгоритмических преобразователей есть существенные отличия:

1. НАП реализует заданное множество производных алгоритмов, а ДАП – только один производный алгоритм.

2. Количество вариантов реализации производных алгоритмов в НАП – один (фиксированный) для каждого алгоритма. В ДАП один и тот же производный алгоритм должен реализовываться с помощью заданного числа различных настроек (т.н. количество модификаций алгоритма).

3. Выбор настроек при каждой реализации производного алгоритма в НАП всегда постоянный для каждого производного алгоритма, а в ДАП при каждой реализации производного алгоритма используется новая настройка.

4. В НАП выбор настройки зависит только от вида реализуемого алгоритма, а в ДАП – от множества факторов, которые предусмотрены или случайным образом.

В общем случае, задача синтеза ДАП может быть сформулирована следующим образом.

- Пусть
- A – алгоритм, реализуемый РБА;
 - n – количество условных переменных алгоритма A ;
 - $H = \{h_1, h_2, \dots, h_q\}$ – множество преобразований алгоритма A ;
 - $A(h_i)$ – производный алгоритм, получаемый из алгоритма A в результате преобразования h_i ;
 - C – заданный алгоритм;
 - k – количество условных переменных алгоритма C ;
 - z – заданное количество модификаций алгоритма C .

Тогда задача синтеза ДАП формулируется следующим образом: найти вид алгоритма A и подмножество преобразований $H^* = \{h_1^*, h_2^*, \dots, h_u^*\}$, $H^* \in H$, при которых $A(h_i^*) = C$; $i = 1, \dots, u$; $u \geq z$; $n \rightarrow \min$.

В основе процедуры построения базового алгоритма A лежит последовательное формирование его значений для соответствующих наборов значений условных переменных.

При этом будем полагать, что изначально алгоритм A не определен и в процессе синтеза ДАП определяется. Алгоритмы $A(h_i)$ и C будем называть эквивалентными (обозначение $A(h_i) = C$), если их значения на соответствующих наборах условных переменных совпадают, и частично эквивалентными (обозначение $A(h_i) \approx C$), если их значения на d наборах условных переменных ($1 \leq d < 2^k$) совпадают, а

на остальных $2^k - d$ наборах значения алгоритма $A(h_i)$ не определены.

Начальное значение количества переменных n получим из следующих соображений. Самым простым способом настройки является выделение отдельных «настроечных» входов в РБА и подача на них фиксированных значений из множества $\{0, 1\}$. Тогда общее количество входов РБА определяется следующим образом:

$$n = k + \lceil \log_2 z \rceil,$$

где $\lceil a \rceil$ означает ближайшее целое, не меньшее a .

Процедура синтеза ДАП состоит из следующих этапов:

1. Полагаем, что текущий номер преобразования $i = 0$, а текущее количество модификаций алгоритма $u = 0$.

2. Определяем номер следующего преобразования, $i = i + 1$.

3. Выбираем преобразование h_i из множества преобразований.

4. Определяем частный алгоритм $A(h_i)$.

5. Если $A(h_i) = C$, то переходим к п. 9.

6. Если $A(h_i) \approx C$, то переходим к п. 8.

7. Переходим к п. 2.

8. Доопределяем алгоритм A т.о., чтобы выполнялось условие $A(h_i) = C$.

9. $u = u + 1$.

10. Если $u < z$, то переходим к п. 2.

11. Определяем количество условных переменных алгоритма A :

$$n = \lceil \log_2 R \rceil,$$

где R – наибольший номер десятичного эквивалента двоичных наборов условных переменных в множестве преобразований $H^* = \{h_1^*, h_2^*, \dots, h_z^*\}$.

12. Конец.

Одним из возможных применений ДАП является обфускация (запутывание) алгоритмов. Поскольку входные операнды постоянно преобразуются, то каждый раз при вызове процедуры будут активизированы различные «ветви» алгоритма, что

затруднит анализ алгоритма и программы. Это свойство ДАП позволяет производить запутывание алгоритмов.

Выводы

Предложенный метод построения динамических алгоритмических преобразователей позволяет повысить сложность обратной трансляции и анализа программ, так как запутывание программы является многоуровневым и начинается на алгоритмическом уровне. Для реализации метода необходимо исследовать виды эффективных преобразований и разработать процедуры формирования функциональных преобразователей.

Список литературы

1. Артюхов В.Л. Повышение эффективности устройств обработки информации / В.Л. Артюхов, Г.Н. Копейкин, А.А. Шальто. – Л.: Энергоиздат, 1981. – 168 с.
2. Шальто А.А. Логическое управление. Методы аппаратной и программной реализации / А.А. Шальто. – СПб.: Наука, 2000. – 780 с.
3. Поликарпова Н.В. Автоматное программирование / Н.В. Поликарпова, А.А. Шальто. – СПб.: Питер, 2010. – 176 с.
4. Шидловский С.В. Автоматическое управление. Перестраиваемые структуры / С.В. Шидловский. – Томск: Томский государственный университет, 2006. – 288 с.
5. Малафеев Е.Е. Динамические алгоритмические преобразователи / Е.Е. Малафеев, Е.Е. Малафеев, А.Г. Нестерович // Матеріали міжнар. наук.-практ. конф. «Наукові дослідження – теорія та експеримент 2009». – Полтава: ІнтерГрафіка, 2009. – Т. 6. – С. 124-126.
6. Настройка алгоритмических преобразователей / И.В. Чумаченко, В.Г. Кучмиев, Н.В. Доценко, Е.Е. Малафеев, В.В. Косенко // Інформаційно-керуючі системи на залізничному транспорті. – 2003. – №2 (41). – С. 16-18.

Поступила в редколлегию 20.08.2010

Рецензент: д-р техн. наук, проф. В.М. Илюшко, Национальный аэрокосмический университет им. Н.Е. Жуковского «ХАИ», Харьков.

ДИНАМІЧНІ АЛГОРИТМІЧНІ ПЕРЕТВОРЮВАЧІ

Є.Є. Малафеев, Є.Є. Малафеев

Досліджені види алгоритмічних перетворювачів, що настроюються. Розглянуті принципи побудови динамічних алгоритмічних перетворювачів і особливості їх роботи. Сформульовано завдання синтезу динамічних алгоритмічних перетворювачів і запропонована процедура їх побудови.

Ключові слова: алгоритмічні перетворювачі, динамічні алгоритмічні перетворювачі, алгоритмічні перетворювачі, що настроюються.

DYNAMIC ALGORITHMIC TRANSFORMERS

Ye.Ye. Malafeev, Ye.Ye. Malafeev

The types of the influenced algorithmic transformers are investigational. Principles of construction of dynamic algorithmic transformers and feature of their work are considered. The task of synthesis of dynamic algorithmic transformers is formulated and procedure of their construction is offered.

Keywords: algorithmic transformers, dynamic algorithmic transformers, influenced algorithmic transformers.