ИСПОЛЬЗОВАНИЕ ПАМЯТИ ОБЪЕКТОВ УПРАВЛЕНИЯ В ЦЕЛЯХ МИНИМИЗАЦИИ БЛОКА ПАМЯТИ УПРАВЛЯЮЩИХ УСТРОЙСТВ

Иванов Н.Н., Шевченко Б.С.

Использование памяти объекта управления /ОУ/ при создании асинхронного дискретного управляющего устройства /АДУУ/, предназначенного для управления этим объектом, привлекает проектировщиков в связи с возможностью минимизации количества внутренних элементов памяти в АДУУ, таких как задержки, триггеры, регистры и др. При этом замкнутая система ОУ-АДУУ /рисунок 1/ может рассматриваться как автономный автомат, функционирование которого происходит в соответствии с предусмотренным проектировщиков режимом работы ОУ.

Использование элементов ОУ, обладающих памятью, в блоке памяти АДУУ описано в работах [1, 2] . Однако, язык секвенций, использованный там для описания АДУУ и тесно связанный с ее структурой, предопределил ограничения, налагаемые на ОУ и сводящиеся к тому, что количества входов (n) и выходов (m) ОУ связаны соотношением $m = \frac{n}{2}$, причем каждый выход x_i является выходом /возможно с отрицанием/ элемента памяти ОУ /исполнительного механизма/,на выходы которого попадаются взводящий и сбрасывающий сигналы z_{iS} и z_{iR} , являющиеся выходными для АДУУ.

В работе [3] рассматривается структура АДУУ, предполагающая использование в качестве внутренней памяти АДУУ элементов так называемой двойной памяти /рис. 2/. В дальнейшем будем считать, что в реальном ОУ все выходы ОУ могут быть разбиты на группы $\hat{x}_i = \{x_{i1}, x_{i2}, \ldots, x_{iK}\}$, в каждой из которых в установившемся состоянии ОУ имеется строго одна единица, в каждом такте работы системы ОУ-АДУУ существует по крайней мере одна группа выходов, равных нулю в переходном процессе. Поскольку в схему совпадения этой структуры /элементы "И"/ наряду с выходами блока памяти поступают также векторы, содержащие представителей из каждой группы $\hat{x}_i$, то появляется возможность переключать элементы внутренней памяти АДУУ в переходном процессе, когда некоторая группа выходов ОУ равна нулю. Предполагается, что переключение элементов памяти происходит со скоростью, существенно превышающей скорость переключения самого быстродействующего выхода ОУ. Такая организация АДУУ полностью исключает влияние состязаний элементов двойной памяти на его работоспособность и допускает вследствие этого произвольное, в том числе и логарифмическое кодирование состояний АДУУ.

Вместе с тем даже логарифмическое кодирование внутренней памяти АДУУ, в силу имеющихся возможностей использования памяти ОУ определяет избыточность такой структурной реализации АДУУ.

В настоящей работе рассматривается методика минимизации структуры, описанной в работе [3] , направленная как на сокращение количества элементов внутренней памяти /в сравнении с логарифмическим кодированием/, так и на уменьшение длин конъюнкций , соответствующих переходам /тактам/ в системе ОУ-АДУУ и опирающаяся на использование триггерных свойств ОУ.

Предварительно укажем способ описания поведения системы ОУ-АДУУ, используемый при составлении технического задания на проектирование АДУУ и основанный на составлении специального графа. Отдельным тактам работы системы взаимно-однозначно соответствуют ребра графа, нагруженные записью следующего вида:

$$\phi \ / \ Z - X, \qquad (1)$$

где ϕ - условие перехода, представляющее собой конъюнкцию $\tilde{y}_i , \ldots \tilde{y}_{is}$ внешних входов, Z - список команд на ОУ, X - реакция ОУ на данные команды - список значений изменяющихся на данном переходе выходов ОУ. Вершины графа соответствуют установившимся состояниям ОУ, причем если в одной вершине сходится несколько путей, состоящих из совокупности связанных ребер, то необходимо соблюдение равенства векторов x , составленных из входов ОУ, равных единице и получающихся в результате отработки этих путей. Таким образом вершины графа оказываются закодированными векторами x , причем может оказаться, что одним и тем же кодовым вектором x наделены несколько вершин.

Будем называть граф, составленный с соблюдением перечисленных правил графом-алгоритмом. Очевидно, что граф-алгоритм несет в себе всю информацию о поведении ОУ, т.е. о множестве допустимых в нем вход-выходных последовательностей $\{Z^* x^*\}$.

Отметим, что граф-алгоритм может быть истолкован как граф автомата Мура, ребра которого нагружены буквами входного алфавита, каковыми являются векторы Z , а состояния /вершины/ отождествляются с выходами x_ℓ /ℓ - номер вершины/.

Техническая реализация графа-алгоритма по структуре с двойной памятью может проводиться отождествлением ℓ -ой вершины с конъюнкцией B_ℓ всех переменных от ОУ, принимающих в данной вершине единичное значение /по одной переменной от каждой группы $\hat{x}_i$ /, и кода вершины, полученного от блока памяти:

$$B_\ell = x_{1j} \cdot \ldots \cdot x_{ht} \cdot \tilde{P}_1 \cdot \ldots \cdot \tilde{P}_q \quad , \qquad (2)$$

где h - число группы $\hat{x}_j$ /$j = 1, 2, \ldots h$/.

Каждое r -ое ребро графа-алгоритма отождествляется с конъюнкцией внешних входов $\tilde{y}_{i1} \cdot \ldots \cdot \tilde{y}_{is}$ и переменной вершины B_ℓ , из которой исходит данное ребро:

$$\Pi_r = B_\ell \cdot \tilde{y}_{i1} \cdot \ldots \cdot \tilde{y}_{is} \quad . \qquad (3)$$

Если на данном ребре условие перехода пусто /переход безусловный/, то $\Pi_r = B_\ell$. Сигналы, соответствующие ребрам, поступают на входы матрицы "ИЛИ".

Выходы матрицы "ИЛИ" поступают на блок памяти и выходной блок, содержащий при необходимости выходные триггеры, которые мы в дальнейшем будем относить к ОУ.

Работа блока внутренней памяти состоит в следующем. При появлении единичного сигнала на взводящем входе некоторого элемента двойной памяти взводится элемент памяти первого уровня, а при исчезновении этого сигнала единица с этого элемента переписывается на элемент уровня, являющийся выходным. Аналогично, при поступлении единичного сигнала на сбрасывающий вход элемента первого уровня на нем записывается 0, переписываемый на элемент второго уровня при исчезновении этого сигнала. Отметим, что исчезновение сигналов на входах блока памяти соответствует началу переходного процесса в ОУ, т.е. отработке команд, инициированных некоторым ребром графа-алгоритма.

Пусть теперь нам предъявлен некоторый граф-алгоритм, по которому составлены конъюнкции B_ℓ , $\ell = 1, 2, \ldots, p$, для всех его p вершин. В каждой конъюнкции выделим ту часть переменных ОУ, которая не изменяется на переходе из вершины, которой данная конъюнкция соответствует. Обозначим ее C_ℓ . Отметим, что если из данной вершины ведет несколько ребер, то соответственно их количеству мы получим конъюкции $C_{\ell 1}, \ldots, C_{\ell t}$.

Рассмотрим теперь произвольную пару вершин графа-алгоритма, которым соответствуют конъюнкции B_i и B_j , причем j-я вершина не является последователем i-ой по K-му переходу. По конъюнкции B_i построим C_{i_K} , соответствующую этому переходу из i-ой вершины. Если существует хотя бы одна группа $\hat{x}_j$, для которой в C_{i_K} и B_j единичное значение принимают различные переменные ОУ, то будем говорить, что j-я вершина изолирована на K-ом переходе из i-ой вершины. В противном случае j-я вершина бедет считаться неизолированной на K-ом переходе из i-ой вершины.

Условие изолированности всех вершин на каком-либо переходе /кроме тех, которые являются началом и концом его/ является достаточным условием отсутствия критических состязаний по переменным ОУ на этом переходе, т.е. на этом переходе вне зависимости от порядка переключения переменных ОУ не сможет стать равной еденице ни одна конъюнкция B_j , соответствующая вершине, в которую не ведет данный переход или из которой он исходит.

Вершину, которая изолирована по всем переходам /кроме тех, которые исходят из нее самой и ведут в нее/ будем называть изолированной. Очевидно, что, если все вершины графа-алгоритма изолированы, то отпадает необходимость во введении внутренней памяти АДУУ, хотя в некоторых случаях, исходя из требований надежности,может применяться кодирование вершин графа-алгоритма внутренней памятью АДУУ.

Пусть теперь в графе-алгоритме существуют вершины, которые не изолированы по некоторым переходам. Поставим задачу отыскания такого кодирования вершин графа-алгоритма внутренней памятью АДУУ, при котором число элементов памяти не превосходило бы $| \log_2 p |$.

Пусть нам удалось найти разбиение множества вершин графа-алгоритма на подмножества, обладающее следующими свойствами:

1. Каждая вершина входит ровно в одно подмножество.
2. Во всяком подмножестве, содержащем более одной вершины, каждая вершина изолирована на переходах, исходящих из других вершин этого подмножества /кроме тех, которые ведут в эту вершину/.
3. Во всяком подмножестве, содержащем более одной вершины, каждая вершина изолирована на переходах, связывающих остальные подмножества с рассматриваемым /если таковые существуют/.

Разбиение, обладающее указанными свойствами будем называть изолированным .

Всем состояниям, входящим в одно подмножество изолированного разбиения, присвоим одинаковые коды $\tilde{P}_1 \ldots \tilde{P}_{q'}$, внутренней памяти / q' - число элементов памяти/, однако, различным подмножествам будут соответствовать различные коды.

Т е о р е м а . При указанном способе кодирования вершин графа-алгоритма критические состязания отсутствуют.

Доказательство. Рассмотрим сначала произвольный переход, на котором переключение памяти не происходит /т.е. переход, соединяющий вершины одного и того же подмножества изолированного разбиения/. Пусть ℓ-я вершина не изолирована на данном переходе. Очевидно, что эта вершина не входит в подмножество, содержащее начальную и конечную вершины данного перехода по свойствам изолированного разбиения. Но тогда во время этого перехода $B_\ell = 0$, т.к. конъюнкция $\tilde{P}_1 \ldots \tilde{P}_{q'}$, соответствующая данной вершине, равна нулю.

Рассмотрим теперь переход, на котором переключаются элементы внутренней памяти АДУУ.

Если существует вершина с номером ℓ, не изолированная на данном переходе, то эта вершина не может находиться в подмножествах, содержащих начальную и конечную вершины данного перехода.

По этой причине $B_\ell = 0$ во время этого перехода, т.к. $\tilde{P}_1 \ldots \tilde{P}_{q'}$, соответствующая данной вершине, равна 0 как в начале так и в конце перехода, а состязания элементов памяти не являются критическими по причинам, указанным выше при описании структуры АДУУ. Теорема доказана.

Если применить логарифмическое кодирование подмножеств изолированного разбиения, то очевидно, что $q' \leq q = |\log_2 p|$. До сих пор мы предполагали, что в каждую конъюнкцию B_ℓ входит строго одна переменная ОУ из каждой группы $\hat{x}_j$.

Рассмотрим теперь, в какой мере могут быть сокращены длины конъюнкций, образуемых переменными ОУ. Для сокращенных конъюнкций введем обозначение B_ℓ'. Выделим прежде всего ту часть конъюнкции переменных ОУ, которая с необходимостью должна быть сохранена при любом возможном сокращении. Рассмотрим некоторую вершину с номером ℓ со всеми входящими ребрами. Для индикации окончания перехода в эту вершину и начала любого из следующих переходов, необходимо в B_ℓ' ввести конъюнцию тех выходных переменных ОУ, каждая из которых взвешена хотя бы на одном входящем ребре, т.е. переменных, входящих в $x_\ell' = \bigcup_{i=1}^{\alpha} X_i$, где α - количество входящих ребер.

Введем следующие обозначения. Пусть из некоторого вектора x_ℓ, соответствующего ℓ-ой вершине графа-алгоритма, выделено подмножество $x_\ell'' \leq x_\ell$. Множество индексов групп $\hat{x}_j$, в которые входят переменные из x_ℓ'', обозначим $N(x_\ell'') \leq \{1,2,..,h\}$. Соответственно множество индексов групп $\hat{x}_j$, в которые входят переменные ОУ, образующие X_i - реакцию ОУ на команды i-го перехода, обозначим $N(X_i)$, $i=1,...,\beta$, где β - количество исходящих из ℓ-ой вершины ребер.

Для того, чтобы на любом из следующих переходов и по окончании его B_ℓ' обратилась бы в ноль, необходимо в B_ℓ' включить конъюнкцию переменных ОУ, входящих в некоторое x_ℓ'' такое, что $N(x_\ell'') \cap N(X_i) \neq \phi$ для любого $i = 1,.., \beta$.

Заметим, что если первое подмножество x_ℓ' определяется однозначно, то нахождение минимального x_ℓ'' связано с некоторым перебором.

Полагая $\tilde{x}_\ell = x_\ell' \cup x_\ell''$, образуем из всех переменных входящих в $\tilde{x}_\ell$, конъюнкцию D_ℓ, которая войдет в B_ℓ' как основа.

Перейдем теперь к построению B_ℓ', имея в виду получение минимальных или близких к минимальным длин этих конъюнкций. рассмотрим произвольную вершину с номером ℓ, входящую в некоторое подмножество изолированного разбиения. Могут встретиться следующие ситуации:

а/ вершина изолирована на всех переходах;
б/ вершина не изолирована на некоторых переходах, на которых переключение внутренней памяти не происходит;
в/ вершина не изолирована на некоторых переходах, сопровождаемых переключением внутренней памяти.

В случае а/ при составлении B'_ℓ к D_ℓ следует добавить такие переменные из x_ℓ, которые позволят получить $B'_\ell = 0$ 1/ на всех переходах из других вершин этого же подмножества, 2/ на всех переходах, ведущих в данное подмножество, 3/ на всех переходах, сопровождаемых переключением более одного элемента памяти, и в том случае, если при некотором порядке переключения элементов памяти возможно получение кодовой комбинации, соответствующей рассматриваемой вершине. Нетрудно убедиться в том, что на всех остальных переходах $B'_\ell = 0$. Добавление переменных к D_ℓ может не понадобиться, если для каждого из всех перечисленных в пп. 1,2,3 переходов, найдется хотя бы одна группа $\hat{x}_j$, для которой в C_{iK} и D_ℓ единичное значение принимают различные переменные /C_{iK} - неизменяемая на переходе часть переменных ОУ/.

Рассмотрение ситуации б/ не приводит к необходимости учета других групп переходов, кроме перечисленных в пп. 1,2,3. Действительно, если рассматриваемая вершина не изолирована на некотором переходе, не сопровождаемом переключением внутренней памяти, то по свойствам изолированного разбиения этот переход соединяет пару вершин из другого ее подмножества. Но тогда $B'_\ell = 0$, т.к. на этом переходе равна нулю конъюнкция, соответствующая кодовой комбинации, получаемой от блока внутренней памяти.

В случае в/ необходимо рассмотреть кроме переходов, перечисленных в пп. 1,2,3, также 4/ переходы, на которых ℓ-я вершина не изолирована и происходит переключение более одного элемента внутренней памяти, причем при некотором порядке переключения возможно получение кодовой комбинации, соответствующей ℓ-ой вершине. В этом случае состязания элементов памяти становятся некритическими, если в B'_ℓ добавить возможно отсутствующую в D_ℓ конъюнкцию переменных из некоторого множества $x'''_\ell < x_\ell$, получаемого следующим образом.

Для каждого перехода, описанного в п. 4, построим пересечение $N_{K_i} = N(x'') \cap N(X_{K_i})$, где x''_K соответствует начальной вершине перехода, X_{K_i} - взвешенное на ребре множество переменных ОУ. Из x_ℓ выберем подмножество x'''_ℓ, такое что $N(x'''_\ell) = \cup N_{K_i}$, где объединение построено по всем переходам, описанным в п. 4. Тем самым будет гарантировано равенство $B'_\ell = 0$ в момент переключения внутренней памяти АДУУ.

Алгоритмически процессы построения изолированного разбиения и сокращенных конъюнкций B'_ℓ вполне доступны для реализации в виде программ для ЭВМ.

Алгоритм построения изолированного разбиения здесь не описывается. Заметим, что при его практическом построении целесообразно стремиться к тому, чтобы получающееся в результате работы алгоритма изолированное разбиение имело бы близкое к минимальному количеству переходов между подмножествами, что в конечном итоге может привести к уменьшению объема матрицы "ИЛИ". Вмешательство проектировщика в процессе автоматического синтеза АДУУ может потребоваться только на этапе кодирования подмножеств изолированного разбиения. При этом он должен стремиться к тому, чтобы максимально приблизиться к однопеременному кодированию, что сократит /исключит при строго однопеременном кодировании/ вычисления, связанные с осуществлением переходов, описанных в п.4.

Л И Т Е Р А Т У Р А

1. Захаров В.Н.: Секвенциальное описание управляющих автоматов.
Известия АН СССР, Техническая кибернетика, 1972, № 2.

2. Захаров В.Н.: Автоматы с распределенной памятью.
"Энергия", 1975.

3. Таль А.А., Руднев В.В., Чернышев В.И. и др.: Асинхронное управляющее устройство.
Авт. свид. СССР № 546885 по заявке от 20.07.73. Бюлл. изобр. № 6, 1977.