УДК 681.325:519.713

В.И. ХАХАНОВ, АБУ ЗАНУНЕХ И.М. ХАЛИЛЬ, А.А. ЕГОРОВ

БУЛЕВА ФОРМА ГРАФОВЫХ МОДЕЛЕЙ ДЛЯ ОПИСАНИЯ ЦИФРОВЫХ СИСТЕМ И СЕТЕЙ

Предлагается дальнейшее развитие булевой формы описания графовых структур [1,2] для описания модей цифровых систем и сетей в целях их последующего анализа на предмет генерации тестов и моделирования неисправного поведения. Указываются свойства, которыми обладает такая форма.

1. Автоматный подход к описанию графа

Графовая структура является наиболее удобной формой представления модели системы или сети для пользователя. Естественным представляется сделать такую форму технологичной и для машины. Это означает разработать модель, которая обладала бы следующими свойствами:

- компактность представления информации о графе;
- привязка к распространенному математическому аппарату;
- наличие эффективных методов анализа графовых структур.

Учитывая, что вершины графа и переменные в булевой алгебре связаны между собой системой отношений, воспользуемся аппаратом последней для описания графовых структур.

Предлагается аналитическая запись описания графа на основе его автоматного представления в форме булевых уравнений (Boolean Equation Form) – булева форма графа (БФГ).

Граф $G=<H,E>$ определен на множестве вершин $H=\{H_1,...,H_i,...,H_k\}, k>0$, связанных дугами $E=\{E_1,...,E_r,...,E_n\}, E_r=(H_i,H_j)\in H\times H$ [3].

В автоматном представлении [4] определение графа есть частный случай, задаваемый выражением

$$M_{FSM}=(X,Y,Z,Z_0,f,g);$$
$$Z(t)=f[X(t),Z(t-1)];$$
$$Y(t)=f[X(t),Z(t-1)],$$

где X,Y,Z,Z_0,f,g – параметры задания конечного автомата: входной алфавит, множество выходных состояний, множество отмеченных состояний, начальное состояние, функции переходов и выходов соответственно.

С учетом следующих условий, используемых для задания графа, характеристические уравнения конечного автомата приводятся к виду:

$$(X;Y=\varnothing;Z=H,Z_0=\varnothing,f=E,g=\varnothing)\Rightarrow$$
$$\Rightarrow\{H(t)=E[X,H(t-1)];Y(t)=\varnothing\}\Rightarrow$$
$$\Rightarrow G=(H,E,X),H=E(X,H),$$

где H,E,X – множество вершин, дуг, идентификаторов дуг. Функция E задает маркированный символом X_S переход из вершины-истока H_I графовой структуры в вершину-сток H_S: $H_S=E(X_S,H_I)$.

Выполнив маркирование самой функции E, переменную X можно исключить как избыточную. В этом случае получается следующая автоматная форма описания модели графовой структуры:

$$G=(H,E),H=E(H),$$

здесь каждый переход обозначен выражением $H_S=E_S(H_I)$.

Далее из общего вида описания автомата осуществляется переход к тривиальному, задаваемому булевыми уравнениями.

2. Булева форма графа

Определение 1. Ориентированный граф $G=<H,E>$ есть множество вершин $H=\{H_1,H_2,...,H_i,...,H_k\}$, которые связаны между собой отношениями, идентифицируемыми совокупностью дуг $H=\{E_1,E_2,...,E_r,...,E_n\}$, где каждая дуга задается конкатенацией двух соединенных между собой вершин и обозначается в виде $E_r=H_i\wedge H_j$ или $H_i\&H_j$, или $H_i\cdot H_j$, или H_iH_j.

Определение 2. Ориентированная дуга $H_i\&H_j$, связывающая две вершины в графе, где H_i – исток, H_j – сток, называется импликативным отношением.

Определение 3. Если α,β – множества вершин или дуг, принадлежащих графу G, то $\alpha\vee\beta$ есть дизъюнкция или объединение упомянутых множеств.

Определение 4. Символ вершины $H_i\in H$ есть формула БФГ.

Символ 1 есть обозначение псевдовершины и является также формулой.

Если α,β – формулы БФГ, то $\alpha\cdot\beta$ и $\alpha\vee\beta$ также являются формулами.

Формулы используют открывающую и закрывающую скобки.

Упорядоченная совокупность вершин, соединенная знаками импликации, называется термом БФГ.

Булевой формой представления графа являются выражения, построенные в соответствии с указанным выше определением.

Для БФГ выполняются следующие аксиомы абстрактной решетки.

Аксиома 1. Коммутативность:

$$\alpha \vee \beta = \beta \vee \alpha\ . \tag{1}$$

Аксиома 2. Ассоциативность:

$$\alpha \vee (\beta \vee \delta) = (\alpha \vee \beta) \vee \delta;$$
$$\alpha \wedge (\beta \wedge \delta) = (\alpha \wedge \beta) \wedge \delta. \tag{2}$$

Аксиома 3. Дистрибутивность:

$$\alpha \wedge (\beta \vee \delta) = (\alpha \wedge \beta) \vee (\alpha \wedge \delta);$$
$$\alpha \vee (\beta \wedge \delta) = (\alpha \vee \beta) \wedge (\alpha \vee \delta). \tag{3}$$

Аксиома 4. Идемпотентность:

$$\alpha \vee \alpha = \alpha\ . \tag{4}$$

Аксиома 5. Аксиома об единичном элементе:

$$\alpha \vee 1 = \alpha;$$
$$\alpha \wedge 1 = \alpha. \tag{5}$$

Для целей структурного анализа и синтеза графов полезными являются следующие тождества.

Тождество 1. Правило поглощения (минимизации) фрагмента графа. Если α, β есть выражения БФГ, описывающие два фрагмента графа, то действительными являются следующие уравнения:

$$\alpha\beta \vee \beta = \alpha\beta;$$
$$\alpha\beta \vee \alpha = \alpha\beta. \tag{6}$$

На основании аксиомы (3) для первого равенства имеем, что

$$\alpha\beta \vee \beta = (\alpha \vee 1)\beta,$$

а после применения (5) получаем конечный и требуемый результат

$$(\alpha \vee 1)\beta = \alpha\beta.$$

Аналогично доказывается и справедливость второго равенства.

Следствие 1. Если α, β – вершины графа, то дуга $\alpha\beta$ поглощает любую вершину, входящую в данную дугу.

Тождество 2. Правило конкатенации (подстановки). Если α, β, δ есть выражения БФГ, то действительно следующее уравнение:

$$\alpha\beta \vee \beta\delta = \alpha\beta\delta. \tag{7}$$

Пусть α, β, δ есть вершины графа G. Тогда выражение $\alpha\beta \vee \beta\delta$ означает, что существуют направленные дуги, соединяющие вершины α, β и β, δ. Но поскольку вершина β является для первой дуги стоком, а для второй – истоком, то существует единственное решение, когда через вершину проходит путь $\alpha\beta\delta$.

Тождество 3. Правило разложения (декомпозиции). Если α, β, δ есть выражения БФГ, являющиеся компонентами или буквами терма, то его можно разложить на два по любой букве (переменной), входящей в исходное выражение:

$$\alpha\beta\delta = \alpha \vee \alpha\beta\delta;\ \alpha\beta\delta = \alpha\beta \vee \beta\delta;\ \alpha\beta\delta = \delta \vee \alpha\beta\delta. \tag{8}$$

В самом деле, первое и последнее равенства истинны согласно правилу поглощения (6). Второе верно вследствие применения правила конкатенации (7).

Теорема 1. *Если* α, β *есть произвольные формулы БФГ, то истинным является выражение* $\alpha\beta\alpha = \beta\alpha\beta$.

Доказательство. На основании тождества (7) имеем $\alpha\beta\alpha = \alpha\beta \vee \beta\alpha$, а с учетом (1) получаем $\alpha\beta \vee \beta\alpha = \beta\alpha \vee \alpha\beta$. После этого, используя (7), получаем требуемое выражение $\beta\alpha \vee \alpha\beta = \beta\alpha\beta$.

Доказанная теорема определяет необходимые и достаточные условия существования контура в графе, что представляется интересным для решения задач анализа цифровых схем, систем и сетей в целях разрыва глобальных обратных связей и последующего ацикливания графовых структур [5].

Следствие 2. Если в терме (пути) существует вершина α, входящая в терм дважды, то все вершины, находящиеся между α и α совместно с вершиной α образуют контур или замкнутый путь.

Определение 5. Терм, в котором существуют только две одинаковые вершины α и α и не существует других вершин, называется единичным. Терм, состоящий из вершин, образующих контур, называется контурным.

Определение 6. Длина контура, задаваемого контурным термом, определяется числом входящих в него букв без одной:

$$L(g_i) = \mathrm{card}(g_i) - 1, \tag{9}$$

где $g_i \in G = \{g_1, g_2, \dots, g_i, \dots, g_n\}$ – контурный терм, входящий а аналитическую запись графовой структуры.

Определение 7. Критерий смежности вершин. Неориентированный граф можно представить в виде расширения ориентированного. Основное условие при этом – если две вершины α, β являются смежными, то для ориентированной структуры данное обстоятельство предполагает наличие двух дуг $\alpha\beta \vee \beta\alpha$.

Лемма 1. Каждая пара смежных дуг неориентированного графа представлена в БФГ контуром, длины 2.

В самом деле, если вершины α, β смежные, то существует путь (ориентированная дуга) $\alpha\beta$ и наоборот $\beta\alpha$. В этом случае две дуги можно записать в виде одного терма $\alpha\beta \vee \beta\alpha = \beta\alpha \vee \alpha\beta$, что является, по определению 7, критерием смежности для неориентированного графа.

Следствие 3. Для алгебраической формы представления неориентированного графа выполняется аксиома коммутативности относительно операции конкатенации

$$\alpha \wedge \beta = \beta \wedge \alpha . \qquad (10)$$

Следствие 4. БФГ есть универсальный математический аппарат для аналитического (компактного) представления графов, включающих фрагменты с ориентированными, неориентированными дугами или без них.

3. Решение задачи тестировния на БФГ

Целесообразность использования булевой формы описания графа иллюстрируется на рассмотрении задачи построения тестов для проверки механизма хранения и передачи данных в микропроцессоре. Модель для проверки механизма хранения и передачи данных представлена совокупностью регистров и связей между ними. Это определяет граф регистровых передач (ГРП), который имеет регистры-вершины R={R_0, ..., R_{m-1}) и две дополнительные вершины IN, OUT, отображающие внешнюю среду, J^A={I_1, ..., I_r} – множество команд пересылок и ветвлений в МП [6]. Проектирование ГРП основывается на следующих правилах: 1. Если существует хотя бы одна команда из J^A, при которой информация передается из R_i в R_j, то в ГРП строится дуга, направленная от R_i к R_j. Она помечается списком команд, которые активизируют данную дугу. 2. Если существует хотя бы одна команда, активизирующая передачу информации из внешней среды в R_i, строится дуга (IN,R_i). 3. Если существует хотя бы одна команда, при которой информация передается из R_i во внешнюю среду, то строится дуга (R_i,OUT). 4. Если передача активизируется командой условного перехода, дуга дополнительно отмечается условием перехода в скобках.

Процедура построения минимизированного теста путем составления неизбыточной совокупности путей ГРП, покрывающей все дуги, представляет собой задачу покрытия: для каждого пути выбирается кратчайшая последовательность команд, активизирующая этот путь; при этом каждому ребру пути ставится в соответствие дизъюнкция команд, помечающих это ребро; составляется некоммутативная конъюнкция дизъюнкций всех ребер пути от IN до OUT; выполняется переход от конъюнктивной нормальной формы (КНФ) к дизъюнктивной нормальной форме (ДНФ), раскрывая скобки с сохранением порядка

конъюнкций применяя, где это возможно, операцию поглощения; выбирается терм полученного выражения, состоящий из минимального числа команд. Для понимания основных шагов процедуры рассмотрим проектирование теста проверки передачи данных на примере ГРП, представленном рисунке.

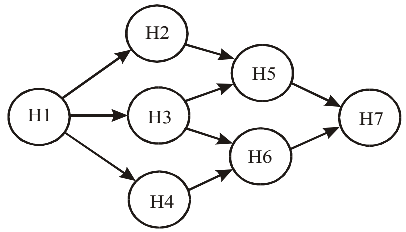

Рис. Граф модели микропроцессора

Для такого графа минимальные покрытие в виде ДНФ имеет следующий вид:

$$G = H_1H_2H_5H_7 \vee H_1H_3H_5H_7 \vee H_1H_3H_6H_7 \vee H_1H_4H_6H_7 .$$

Каждой дуге графа ставится в сответствие набор команд, активизирующих переход $H_i \to H_j$:

$$H_1H_2 = 1 \vee 2 \vee 3; H_1H_3 = 5 \vee 6; H_1H_4 = 7 \vee 8 \vee A;$$
$$H_2H_5 = 2 \vee 3 \vee 4; H_3H_5 = 7; H_3H_6 = 8; H_4H_6 = B;$$
$$H_5H_7 = 3; H_6H_7 = 8 \vee B; \ .$$

Перемножив конъюнкции микроопераций модели процессора, формирующие четыре пути, получим дизъюнктивные формы записи участков графовой структуры:

F1=123 $\vee$ 223 $\vee$ 323 $\vee$ 133 $\vee$ 233 $\vee$ 333 $\vee$ 143 $\vee$ 243 $\vee$ 343=3;
F2=573 $\vee$ 673;
F3=588 $\vee$ 688 $\vee$ 58B $\vee$ 68B=58 $\vee$ 68;
F4=7B8 $\vee$ 8B8 $\vee$ AB8 $\vee$ 7BB $\vee$ 8BB $\vee$ ABB=7B $\vee$ 8B $\vee$ AB.

Минимальные термы, активизирующие все участки выбранных путей, представляют собой искомое решение, состоящее из восьми команд: $F = 3 \vee 573 \vee 58 \vee 7B$, выполнение которых с операндами и адресами теста переноса [6] обеспечит проверку механизма хранения и передачи данных предложенного графа МП.

Таким образом, представление ориентированного графа в виде булевых функций позволяет достаточно эффективно решать задачи технической диагностики традиционными, достаточно развитыми методами логического анализа и синтеза булевых уравнений.

Список литературы: 1. *Воробьев В.Н.* Об алгебраическом способе описания графов. Изв. АН СССР. 1984. Вып. 4. С. 78-82. 2. *Хаханов В.И.* Элементы анализа моделей дискретных устройств/Деп. в УкрНИИНТИ Харьк. ин-том радиоэлектрон. №219Ук-85Деп. Х.:ХИРЭ, 1985. 18с. 3. *Горбатов В.А.* Основы дискретной математики. М.: Высш. шк., 1986. 311 с. 4. *Глушков В.М.* Кибернетика. Вопросы теории и практики. М.: Наука, 1986. 488 с. 2. *Бондаренко М.Ф., Кривуля Г.Ф., Рябцев В.Г., Фрадков С.А., Хаханов В.И.* Проектирование и диагностика компьютерных систем и сетей. К.: НМЦ ВО. 2000. 306 с. 6. *Чипулис В.П., Шаршунов С.Г.* Построение тестов микропроцессоров. 2. Проверка хранения и передачи данных // Автоматика и телемеханика. 1986. №1. С.139–150.

Поступила в редколлегию 03.12.2000

Хаханов Владимир Иванович, д-р техн. наук, профессор кафедры автоматизации проектирования вычислительной техники ХТУРЭ. Научные интересы: техническая диагностика вычислительных систем. Увлечения: баскетбол, футбол, горные лыжи. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. 40-93-26. E-mail: Hahanov@kture.kharkov.ua

Абу Занунех Халиль И.М., аспирант кафедры автоматизации проектирования вычислительной техники ХТУРЭ. Научные интересы: техническая диагностика вычислительных устройств и сетей. Хобби: шахматы, футбол, теннис. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. 40-93-26.

Егоров Александр Андреевич, студент факультета компьютерной инженерии и управления ХТУРЭ. Научные интересы: техническая диагностика и проектирование вычислительных устройств и систем. Хобби: иностранные языки. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. 40-93-26.

УДК 004.056

И.Ш.НЕВЛЮДОВ, О.В.ТУЧИН, АЛЬ МОХАДМЕХ ЗАФЕР

РАЗРАБОТКА АЛГОРИТМА РАБОТЫ КОММУТАТОРА ДАТЧИКОВ ПЕРВИЧНОЙ ИНФОРМАЦИИ СИСТЕМЫ УПРАВЛЕНИЯ БЕЗОПАСНОСТЬЮ ОБЪЕКТА

Рассматриваются вопросы разработки алгоритмов функционирования коммутаторов систем безопасности объекта на базе микропроцессорной техники с использованием критериев: количественный и качественный состав датчиков первичной информации; способ считывания информации с датчиков; способ обработки информации; разрядность шины данных, связывающей коммутатор с ЭВМ более высокого уровня; режимы работы коммутатора (ручной, автоматический, режим обучения).

Существование современных автоматизированных систем управления производством неразрывно связано с организацией всех видов деятельности, отображающих производственную структуру субъекта хозяйствования. Обеспечение безопасности предприятия является одной из важнейших задач функционирования современного автоматизированного производства.

Существующие автоматизированные системы безопасности рассчитаны в основном на автономное функционирование [1]. Их интеграция в единую автоматизированную систему управления производством, как правило, затруднена. В связи с этим разработка автоматических устройств охранной сигнализации, способных интегрироваться в единую систему управления, является актуальной задачей.

Управление процессом охраны любого объекта фактически состоит в том, что необходимо обеспечить контроль безопасности и в случае возникновения проблемных (нештатных) ситуаций принять решение об их устранении и обеспечить выполнение данного решения.

Наша задача – разработка алгоритма функционирования коммутатора системы безопасности объекта, выполненного на базе микропроцессорной техники. При этом необходимо обеспечить:

а) оптимальный алгоритм опроса датчиков первичной информации;

б) возможность интеграции в автоматизированную систему управления производством;

в) возможность функционирования в автономном режиме.

Целью данной работы является разработка алгоритма функционирования программируемого микроконтроллера, на основании которого построен коммутатор охранной системы, в рамках комплекса работ по разработке программно-аппаратного комплекса управления безопасностью объекта, представляющего собой систему подготовки принятия решения об устранении возникших нештатных (конфликтных) ситуаций [2,3].

Разработку алгоритма управления системой безопасности осуществляем с использованием модульного принципа. Именно такая организация позволит разработать общую методику формирования информационной модели и принятия решения в экспертной системе, что предусматривается в процессе разработки [2].

Первым этапом разработки программного продукта является [4]:

а) разработка алгоритма функционирования программируемого микроконтроллера;

б) разработка алгоритма функционирования компьютера, осуществляющего подготовку информации для главного компьютера (условимся называть его компьютером второго уровня);

в) разработка алгоритма функционирования главного компьютера;

г) написание программного кода.

В работу микроконтроллера в общем случае входят такие задачи:

а) опрос датчиков первичной информации;

б) анализ информации с датчиков;

в) подключение видеокамер;

г) передача информации о состоянии системы компьютеру более высокого уровня.