

# ОРГАНИЗАЦИЯ ЗНАНИЙ В АВТОМАТИЗИРОВАННОЙ СИСТЕМЕ ДИАГНОСТИРОВАНИЯ И ИДЕНТИФИКАЦИИ ЭЛЕКТРОННЫХ УСТРОЙСТВ

В.И. Месюра, к.т.н.,

Винницкий Государственный Технический Университет,  
286027, Украина, г. Винница, пр. Космонавтов, д. 61, кв. 2.

## Abstract

Algorithms of synthesis of test actions must provide given trustworthiness and completeness of test relatively to considered class of unit under test (UUT) faults. Efficiency of such algorithms in the decisive degree is determined by the presence of adequate mathematical model of object. Basis for creating mathematical model is, as a rule, UUT functional diagram. However, in case of technical documentation lack, information about demanded UUT characteristics we can receive only in the UUT identification mode by building mathematical model of UUT on the results of input and output signals measurements [1,2].

The goal of the realization in-circuit identification mode is expanding of in-circuit test systems (ICT) functional possibilities:

- restoring UUT functional diagram in case of technical documentation lack or with existing changes of diagram which is not reflected in this documentation;
- automated test program generation is realized in the ICT selfprogramming mode (in this case studying UUT may be defective).

Present paper is devoted to consideration of training system of electronic devices in-circuit test and identification.

Keywords: digital component, identification, in-circuit testing, training.

Алгоритмы синтеза тестовых воздействий должны обеспечивать заданные полноту и достоверность тестирования относительно рассматриваемого класса дефектов электронного устройства (ЭУ). Эффективность таких алгоритмов в значительной мере определяется степенью адекватности используемой математической модели реальному ЭУ. Основой для создания модели служит, как правило, электрическая принципиальная схема устройства. Однако, в случае отсутствия комплекта технической документации, требуемая для создания модели ЭУ информация о характеристиках устройства может быть получена только в режиме его идентификации, на основании анализа входных и выходных воздействий.

Целью реализации режима покомпонентной идентификации (in-circuit identification) является расширение функциональных возможностей автоматизированных систем диагностирования ЭУ в части:

- восстановления электрической принципиальной схемы устройства в случае ее отсутствия, либо при наличии в схеме изменений, не отраженных в комплекте документации;
- автоматизированной генерации тестовых программ в режиме самопрограммирования системы диагностирования [1].

В докладе рассматриваются вопросы построения базы знаний автоматизированной системы покомпонентного диагностирования и идентификации (АСДИ) цифровых электронных устройств.

## Формальное описание объекта идентификации

Предлагаемый структурно-функциональный подход к идентификации цифровых ЭУ основывается на комплексном анализе структурных и функциональных характеристик цифрового компонента  $D_j$  и ОД в целом. Соответственно, математическая модель компонента  $D_j$ , представляется, в общем виде, как:

$$M_j = (F_j, V_j),$$



где  $F_j$  - реализуемая компонентом  $D_j$  функция, отображающая зависимости между уровнями сигналов присутствующих на множестве  $V_j$  его выводов;

$$V_j = X_j \cup Y_j,$$

где  $X_j$  - множество входов, а  $Y_j$  - множество выходов компонента.

Структурные характеристики компонента будем представлять вектором структурных признаков  $Z(D_j)$ , размерность которого совпадает с количеством выводов компонента, а значения координат определяются следующим образом:

$$z_{i,j} = \begin{cases} 1, & \text{if } v_{i,j} \in X_j; \\ 0, & \text{if } v_{i,j} \in Y_j. \end{cases}$$

Для снижения размерности поставленной задачи используем декомпозиционное представление функции цифрового компонента в виде:

$$F_j = \{f_{i,j} \mid i = \overline{1, |Y_j|}\},$$

где  $\{f_{i,j}\}$  - множество элементарных функций, реализуемых базовыми элементами, входящими в состав компонента.

Каждая элементарная функция  $f_{i,j}$  отображает зависимость состояния одного выхода  $U_{i,j} \in Y_j$  компонента  $D_j$  от состояния некоторых подмножеств его входов  $X_{i,j} \subset X_j$ , описанную в алгебраической форме [1].

Например, состояние выхода  $Q$  для компонента типа LN74LS74 будет описываться выражением:

$$Q_1 = (1 - S_1) + S_1 \times R_1 \times [Q_1^\alpha + C' \times (D_1 - Q_1^\alpha)], \quad (1)$$

где  $Q^\alpha$  - состояние выхода  $Q$  в предшествующем такте испытаний.

Систему уравнений вида (1), описывающих функциональные свойства компонента, будем называть функционально-алгебраической моделью (ФАМ) цифрового компонента [1].

Обозначим через  $v_{i,j}^t$  значения уровней сигналов, присутствующих в момент времени  $t$  на выводах компонента  $D_j$ . Так как  $v_{i,j}^t$  могут принимать значения только из области  $\{0, 1\}$ , состояние ЦК в каждый момент времени может быть описано числовой характеристикой  $n_j^t$ , которую будем называть кодом состояния компонента:

$$n_j^t = \sum_{i=1}^{|V_j|} 2^{i-1} \cdot v_{i,j}^t \quad (2)$$

Подставим в (2), на места выводов  $v_{i,j} \in Y_j$ , выражения вида (1), описывающие взаимосвязь между значениями уровней сигналов на входах и выходах компонента. Полученное в результате выражение назовем структурно-функциональной алгебраической моделью (СФАМ) ЦК.

Множество кодов состояний  $N_j = \{n_j^k, k = \overline{1, 1}\}$ , получаемое из СФАМ подстановкой все возможных комбинаций входных сигналов, будем называть описанием пространства состояний (ПС) компонента  $D_j$  в цифровой форме и обозначать  $N_j$  или  $N(D_j)$ . Как следует из определения, ПС представляет как структурные, так и функциональные свойства ЦК.

#### Представление знаний

В АСДИ используется эвристическая модель знаний. Знания о ЭУ и ЦК представляются продукциями, описывающими, в частности, эвристические правила разработки и конструирования печатных узлов, например, такие как:

- ЕСЛИ компонент имеет счетный вход, ТО это счетчик с вероятностью  $P=1$ ;



- ЕСЛИ выводы компонента связаны между собой И соединены с потенциальной шиной питания ТО это выходы с вероятностью  $P=0,8$ .

- ЕСЛИ два вывода компонента связаны между собой И не соединены ни с одним другим выводом ТО один из этих выводов является входом, а другой - выходом, с вероятностью  $P=1$ , и.т.д.

Знания о идентифицируемых компонентах и реализуемых ими функциях организованы в иерархию типов объектов. В основу иерархии положено множество ФАМ элементарных функций. Совокупность таких функций определяет функциональность ЦК. Абстрактный тип объекта TFuncEl (базовый функциональный элемент), включает описание реализуемой им элементарной функции, а также описание количества и типов входов и выходов базового элемента. Основными методами TFuncEl являются:

- формирование пространства состояний по ФАМ;
- определение состояний выходов элемента по заданному состоянию входов;
- вычисление входного набора, соответствующего заданному состоянию выходов элемента;
- проверка принадлежности некоторого состояния пространству состояний компонента заданного типа

Для описания отдельных компонентов, фрагментов печатных узлов и печатных узлов в целом используются типы объектов TChip, TCluster и TUnit соответственно. Эти типы объектов связаны с TFuncEl и друг с другом отношением агрегатирования.

### Принципы классификации объектов в АСДИ

Источником информации для построения описаний цифровых компонентов является как справочная литература, так и физические образцы компонентов, на которые отсутствует техническая документация. В общем случае, априорная информация о множестве всех объектов, подлежащих идентификации, отсутствует. Имеющаяся справочная информация может быть использована лишь для определения первоначально-го словаря признаков, который в дальнейшем пополняется в процессе функционирования АСДИ электронных устройств. При этом возникает задача выбора, в пределах выделенных на построение аппаратной части ресурсов, оптимального рабочего словаря признаков и решающих правил, обеспечивающих наиболее эффективное использование решений, принимаемых по результатам идентификации структуры объекта.

На стадии обучения АСДИ ей предъявляют исходную совокупность объектов  $\omega_1 \dots \omega_k$ , заданных наборами значений своих признаков

$R_i = \{r_j^n\}$ ,  $n = \overline{1, N}$ . При этом принадлежность объектов исходной выборки к какому-либо

классу не указывается. Вместо этого предлагается набор правил, в соответствии с которым на этапе самообучения АСДИ сама вырабатывает классификацию, которая, в общем случае, может отличаться от привычной для специалистов в данной предметной области. Тем не менее, принятый подход освобождает оператора от трудоемкой работы по классификации объектов и позволяет АСДИ создать наиболее удобные для ее функционирования классы.

Целью режима самообучения АСДИ является разбиение множества предъявляемых ей объектов на такие классы, к каждому из которых можно применить некий оптимизированный алгоритм идентификации.

С этой точки зрения, в соответствии с предлагаемым подходом, в словарь признаков АСДИ были включены:

- множество функционально-алгебраических моделей (ФАМ) базовых элементов [3], реализуемых в составе цифровых компонентов;
- векторы структурных признаков компонентов [1], представляемые в виде вектора входов  $V_j^I$  и вектора выходов  $V_j^O$  компонента  $D_j$ ;

- векторы входов и выходов  $V_{j,k}^I$ ,  $V_{j,k}^O$  для каждого из базовых элементов, входящих в состав  $D_j$ ;

- типы входов компонента: синхронный (С), установочный (R,S), счетный (+,-), информационный (D) и т.д.

Указанные признаки могут быть получены из СФАМ компонента и, наоборот, этих признаков достаточно для синтеза СФАМ компонента.

Так, вектор признаков компонента SN74LS00 будет включать следу-



ющие значения:

$$Z(\text{SN74LS00}) = \{(f_k), (1B1B, 04A4), (0003, 0004; 0018, 0020; 0300, 0080; 1800, 0400), (D1, D2; D4, D5; D9, D10; D12, D13)\},$$

где  $f_k = 1 - \prod_{i=1}^n D_i$ ,  $n = 2$ ,  $k = \overline{1, 4}$  - ФАМ базовых элементов компонента;

3B5B, 04A4 - шестнадцатиричное представление векторов входов и выходов компонента;

0003, 0004 - шестнадцатиричное представление векторов входов и выходов первого базового элемента;

D1...D13 - описание типов входов компонента.

Следует отметить, что выводы питания компонентов, относимые в ВСП к множеству входов компонента, в векторе входов не учитываются.

Вектор признаков компонента SN74LS74 примет следующий вид:

$$Z(\text{SN74LS74}) = \{(f_1, f_m), (1E3F, 01B0), (000F, 0030; 1E00, 0180), (R1, D2, C3, R4, S10, C11, D12, R13)\},$$

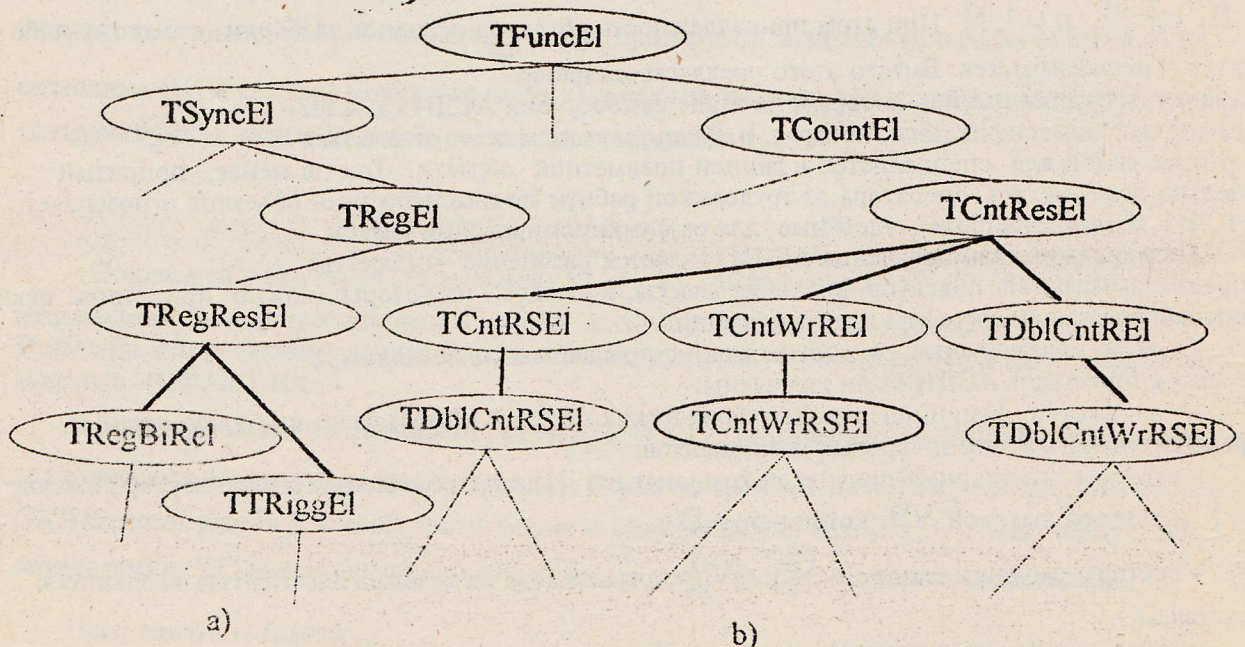
$$\text{где } f_1 = (1 - S) + S \times R \times (Q^\alpha + C' \times (D - Q^\alpha)),$$

$$f_m = (1 - R) + S \times R \times (\bar{Q}^\alpha + C' \times (D - \bar{Q}^\alpha)).$$

В процессе самообучения осуществляется некоторое множество классификаций, объединяющих объекты по разным признакам:

- функциональным (выполняемые компонентом функции: счетчик, сумматор, регистр, дешифратор, мультиплексор и т.д.);
- структурно-функциональным (наличие в составе компонентов однотипных базовых элементов);
- структурным (совпадение, возможно частичное, векторов входов базовых элементов компонента).

На рис.1 приведен пример иерархической классификации цифровых компонентов по функциональным признакам.



TSyncEl - включает в свой состав синхровход (вход C);  
TRegEl - включает входы D и C (D - информ. вход);



TTriggEl- включает входы D,C,R,S (R - вход уст. в 0,  
S - вход уст. в 1);

.....  
TCountEl- включает вход +1 (счетный вход);

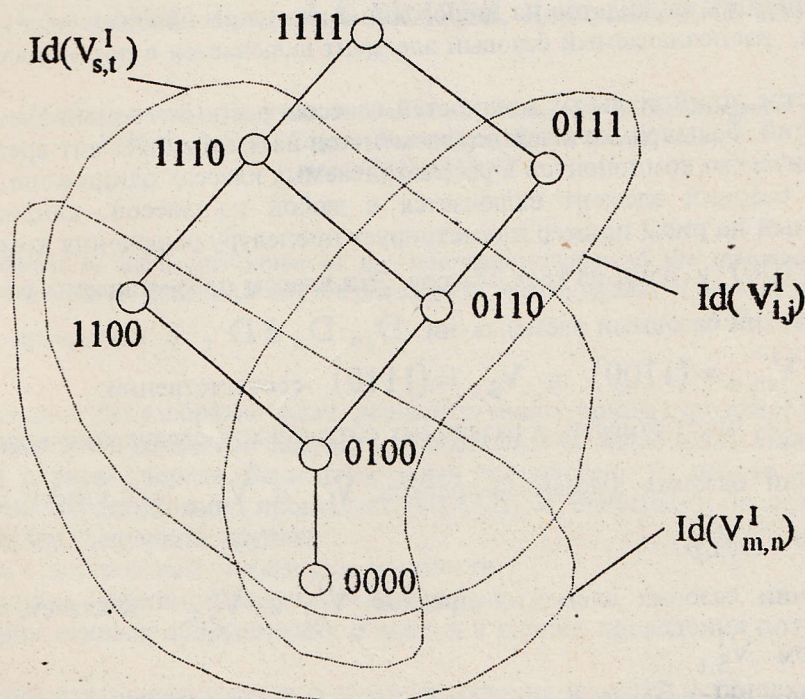
.....  
TDbtCntWrRSEl - включает входы (+1, -1, Wr -параллельной  
записи, R,S).

Рис.1. Фрагменты иерархии базовых функциональных элементов,  
построенных на базе а) синхровхода; б) счетного входа:

Основой для классификации здесь служат абстрактные объекты ви- да TFuncEl, располагающиеся в вершинах соответствующих поддеревьев иерархии функциональных элементов [3]. Так, к одному классу объек-тов могут быть отнесены компоненты имеющие синхровход, поскольку хотя бы по одному из базовых элементов таких компонентов имеют общего предка TSyncEl, расположенного наверху соответствующей иерар-хии.

По мере роста размера библиотеки компонентов может происходить разбиение одного класса на ряд новых классов, родительские элементы которых будут, при этом, расположены на более низких уровнях дерева иерархии по сравнению с родительской вершиной исходного класса.

Для выполнени класификации по структурным признакам удобно использовать представление множества векторов входов базовых функ-циональных элементов в виде решеток [4], а именно, единичных n-мер-ных кубов. Рассмотрим принцип классификации на собственной подрешетке единичного 4-мерного куба, показанной на рис.2.



$$V_{l,j}^I = 0111: \text{Id}(0111) = \{0111, 0110, 0100, 0000\};$$

$$V_{m,n}^I = 1100: \text{Id}(1100) = \{1100, 0100, 0000\};$$

$$V_{s,t}^I = 1110: \text{Id}(1110) = \{1110, 1100, 0110, 0100, 0000\}.$$

Рис.2. Классификация компонентов с использованием решеток:



В рассматриваемом примере компоненты  $D_k$ , один из базовых элементов которых описывается входным вектором  $V_{i,k}^I$ , будут отнесены к одному и тому же классу в случае, если они удовлетворяют следующим условиям:

$$V_{i,k}^I \subset \text{Id}(V_{l,j}^I); \quad (3)$$

$$V_{i,k}^O \cap V_{l,j}^I = \emptyset; \quad (4)$$

$$V_{i,k}^I \neq \emptyset; \quad (5)$$

где  $V_{l,j}^I$  - входной вектор родительского базового элемента класса.

Условие (3), в соответствии с терминологией объектно-ориентированного проектирования, определяет принцип полиморфизма [5]. При его выполнении точки приложения тестовых воздействий, обеспечивающие возможность идентификации базового элемента предка, обеспечат также возможность идентификации любых его дочерних элементов.

Выполнение условия (4) гарантирует исключение принудительного наведения сигналов на выходы дочерней базового элемента при подаче на его входы тестового воздействия, предназначенного для идентификации базового элемента - предка данного класса.

Условие (5) исключает возможность включения в класс базовых элементов, ни один из входов которых не совпадает со входами элемента-предка, расположенного в вершине соответствующей иерархии.

При возможности включения базового элемента более чем в один класс, классификация выполняется по следующему правилу:

- распознаваемый базовый элемент включается в класс генерируемый элементом-предком, расположенным на наиболее высоком уровне иерархии;

- при наличии двух кандидатов на порождение класса расположенных на одном и том же уровне иерархии, распознаваемый базовый элемент включается в потенциально более мощный класс;

- при равенстве потенциальных мощностей классов, распознаваемый элемент включается в класс, содержащий большее количество элементов на данный момент времени;

- если и количество компонентов в рассматриваемых классах одинаково, то распознаваемый базовый элемент включается в любой из классов - кандидатов.

Приведенный на рис.2 пример иллюстрирует процедуру включения компонента

$D_k \cap V_{i,k}^I = (0100)$  в один из трех классов. Эти классы представлены расположенными в вершинах их иерархий базовыми элементами  $D_j, D_n \in D_t$  с векторами входов  $V_{l,j}^I = (0111)$ ,  $V_{m,n}^I = (1100)$  и  $V_{S,t}^I = (1110)$  соответственно.

Компонент  $D_j$  будет отнесен, в различных ситуациях, к следующим классам:

- при наличии базовых элементов-предков  $V_{l,j}^I$  и  $V_{m,n}^I$  - к классу, представляемому базовым элементом  $V_{l,j}^I$ ;

- при наличии базовых элементов-предков  $V_{l,j}^I$  и  $V_{S,t}^I$  - к классу, представляемому базовым элементом  $V_{S,t}^I$ .

Рассмотренное представление информации в базе знаний АСДИ оказалось целесообразным, так как позволило существенно сократить затраты времени на реализацию процедуры идентификации. Программное обеспечение базы знаний реализовано на языке Borland Pascal 7.0 for Windows with Objects.

#### Список использованных источников

- [1] Baida N.P., Mesyura V.I. Method of Component Identification in Printed Circuits Units with Unknown Structure. Proceedings of the XIII IMEKO World Congress, Torino, 1994, pp. 2246-2251.



- [2] Байда Н.П., Месюра В.И., Роик А.М. Самообучающиеся анализаторы производственных дефектов РЭА. М.: Радио и связь, 1991. - 256с.
- [3] Дмитренко С.В., Ефименко А.Я., Месюра В.И. Иерархия объектов системы идентификации структуры электронных устройств// Приборостроение - 94 / Материалы научн.-техн. конф. с междунар. участием. - Симферополь, 1994. - с.34.
- [4] Биркгоф Г., Барти Т. Современная прикладная алгебра. - М.: Мир, 1976. - 400 с.
- [5] Буч Г. Объектно-ориентированное проектирование с примерами применения: Пер. с англ. - М.: Конкорд, 1992. - 519 с.

## Основные задачи системы информационной поддержки оператора судовых ядерных энергетических установок

Анатолий И. САБАДАШ, Валерий А. БЕЛЯЕВ, Константин Ю. КНЯЗЕВСКИЙ

*Государственная морская академия им. адмирала С.О. Макарова  
183026, Россия, Санкт-Петербург, Косая линия, 15а,  
тел. (812) 4213447*

**Abstract.** The problems of intellectualization of the information support of the operator's activities at the marine nuclear installations are considered and the principles of making systems of the kind for the ice-breakers are developed.

Учитывая большое разнообразие задач, решаемых операторами судовой ЯЗУ, при создании СИПО следует выделить основные задачи, составляющие базовую часть системы и подлежащие решению в первую очередь. Выделение задач произведено на основе анализа действий операторов при эксплуатационных происшествиях (ЭП) со значимыми последствиями. Задачи можно разделить на следующие группы:

- связанные с экстренным снижением мощности;
- срабатыванием аварийной защиты реактора;
- при наличии отказов оборудования и систем в период проведения потенциально опасных работ;
- при наличии отдельных отказов средств управления и частей установки.

Из рассмотрения исключены ситуации при наличии повторных отказов вследствие их малочисленности (не более 2% от общего числа анализируемых случаев). Задачи первых трех групп оператору приходится решать наиболее часто (21%, 40% и 37% соответственно). Атомные суда работают на акваториях и рейдах портов, обеспечивают проводку караванов в фарватерах северных рек, узкостях, тяжелых ледовых полях. Поэтому возникновение эксплуатационного происшествия, а также неблагоприятное его развитие приводит к значительному до