

Разработка и исследование моделей блоков цифровых систем на основе их представления в виде семейства стационарных динамических систем

А.Д. Иванников

Институт проблем проектирования в микроэлектронике РАН (ИППМ РАН),

adi@ippm.ru

Аннотация — При отладке проектов цифровых систем методом моделирования важной задачей является выбор набора отладочных тестов, то есть входных воздействий, которые подаются на компьютерную модель проектируемой системы с целью проверки правильности ее функционирования. Формирование полного в каком-то смысле набора отладочных тестов возможно тем или иным способом, если известно множество допустимых входных воздействий на проектируемую систему. Формирование описания такого множества возможно, если известно описание множества допустимых входных воздействий на блоки проектируемой цифровой системы. В статье осуществляется исследование моделей блоков цифровых систем прежде всего с точки зрения описания множества допустимых входных воздействий. В качестве модели цифровых блоков используется множество стационарных динамических систем с непрерывным временем и дискретными значениями логических сигналов. Поскольку в ряде случаев обмен сигналами цифровых блоков с другими блоками системы и внешним миром инициируется самими блоками, то в качестве отладочных тестов рассматриваются входные взаимодействия, включающие изменения как входных сигналов блока, так и выходных сигналов управления обменом. Предлагается графовое представление допустимых входных взаимодействий цифровых блоков и системы в целом для каждой выполняемой функции.

Ключевые слова — логико-временной анализ цифровых систем, отладка методом моделирования, структура множества входных взаимодействий, входные взаимодействия цифровых блоков, графовое представление множества входных взаимодействий

I. ВВЕДЕНИЕ

При проектировании цифровых систем для отладки проектов широко используется метод моделирования. На компьютерную модель цифровой системы подаются некоторые входные воздействия, а реакция модели проектируемой системы проверяется на соответствие техническому заданию [1-4].

При этом важной задачей является выбор конечного числа конечных по времени тестовых входных взаимодействий (тестовых примеров). С ростом сложности проектируемых цифровых систем и, соответственно, ростом сложности и длительности

тестирования их проектов все более актуальной становится задача выбора минимального полного в определенном смысле набора тестов, правильное выполнение которого позволяет убедиться в отсутствии ошибок проектирования [5-7].

Для того, чтобы составить набор входных тестовых примеров необходимо иметь описание множества допустимых входных воздействий как на разрабатываемую цифровую систему, так и на ее блоки. Целью настоящего исследования является разработка моделей блоков цифровых систем прежде всего с точки зрения описания множества допустимых входных воздействий.

II. ОПИСАНИЕ ИСПОЛЪЗУЕМОЙ МОДЕЛИ

При отладке проектов цифровых систем методом моделирования необходимо выбрать уровень или уровни моделей цифровой системы и ее блоков. Обычно осуществляется декомпозиция задачи отладки проекта [8] прежде всего по типу выявляемых ошибок. Так, для верификации временных диаграмм обмена информацией между блоками используются модели цифровых элементов с многозначным представлением электрических сигналов [9]. Для проверки правильности логического функционирования используются модели с булевым представлением сигналов на входах и выходах [6, 10]. Используются также различные высокоуровневые модели [11-14].

При проектировании сложных цифровых систем разработчик должен обеспечить, прежде всего, требуемое внешнее поведение цифровой системы, то есть требуемое взаимодействие системы с внешней средой. При этом существенным является как последовательность выходных сигналов цифровой системы, так и моменты времени появления и изменений этих сигналов, причем временные ограничения обычно задаются интервалами значений. В связи с этим математической моделью внешнего поведения проектируемой цифровой системы или ее блока может служить семейство стационарных динамических систем [1, 15].

Взаимодействие цифровой системы с объектом управления и внешним миром вообще осуществляется

через внешние линии и шины – наборы линий, по которым передается однородная информация, например, адреса или данные. Причем в цифровых системах управления широко используются двунаправленные шины и линии, имеющие также состояние с высоким выходным сопротивлением (отключенное состояние). Будем рассматривать логическую модель сигналов на шинах и линиях цифровых систем, то есть считать, что значения сигналов представляются как 0 или 1 на линиях и как число из диапазона $0 - 2^n - 1$ на шинах системы. Цифровые сигналы внешних шин и линий назовем терминальными переменными – множество \mathbf{P} . Переменная $p \in \mathbf{P}$ всегда имеет одно из значений конечного множества \mathbf{Z}_p , элементы которого определяют как целочисленное значение сигнала, так и направленность работы шины или линии.

Событием по переменной p называется изменение переменной p со значения $z_1 \in \mathbf{Z}_p$ на значение $z_2 \in \mathbf{Z}_p$ в момент времени t . Обозначим такое событие χ_{p,z_1,z_2}^t . Взаимодействие цифровой системы с внешней средой, включая управляемый объект, есть последовательность переключений сигналов на терминальных шинах и линиях, то есть последовательность событий. Для каждой проектируемой системы имеется множество Ψ допустимых взаимодействий с внешней средой, каждое из которых есть отображение $\psi : [0, t) \rightarrow \mathbf{Q}$, $t \in \mathbf{T}$, $\mathbf{Q} = \prod_{p \in \mathbf{P}} \mathbf{Z}_p$.

В цифровых системах для каждого конечного временного интервала количество событий по терминальным переменным, то есть количество изменений их значений, конечно. В связи с этим любое взаимодействие ψ может быть представлено в виде вектора $(z_{p_1}^H, \dots, z_{p_k}^H)$ начальных значений переменных p_1, \dots, p_k (k – мощность множества \mathbf{P}) в момент времени $t = 0$ и последовательности событий по переменным множества \mathbf{P} с конечным числом событий за любой конечный интервал времени:

$$\psi = (z_{p_1}^H, \dots, z_{p_k}^H), \chi_{p_{i_1}, z_{j_1}, z_{j_2}}^{t_1}, \chi_{p_{i_2}, z_{j_3}, z_{j_4}}^{t_2}, \chi_{p_{i_3}, z_{j_5}, z_{j_6}}^{t_3}, \dots \quad (1)$$

где $t_1 \leq t_2 \leq t_3 \leq \dots$ – упорядоченная последовательность времен событий;

$p_{i_1}, p_{i_2}, p_{i_3}, \dots$ – переменные, принадлежащие множеству \mathbf{P} ;

$z_{j_1}, z_{j_3}, z_{j_5}, \dots$ – значения переменных непосредственно перед событием;

$z_{j_2}, z_{j_4}, z_{j_6}, \dots$ – значения переменных непосредственно после события.

Если в последовательности (1) выделить только события, являющиеся изменениями входных сигналов, то такую последовательность можно назвать входным воздействием. Однако часто моменты подачи входных сигналов на цифровую систему определяются готовностью системы принять эти сигналы, на что указывают определенные выходные сигналы системы. Выполнение какой-либо операции, например,

считывания данных цифровой системой, может инициироваться не сигналами внешней среды, а самой системой. В связи с этим использование в качестве аргументов функционирования цифровой системы входных воздействий не всегда удобно.

Выделим из последовательности событий (1) взаимодействия ψ – последовательность входных событий и выходных событий управления обменом, которые по заданному протоколу обмена обуславливают моменты времени входных событий. Назовем эту последовательность входным взаимодействием:

$$\mu = (z_{p_1}^H, \dots, z_{p_{n+q}}^H), \chi_{p_{i_1}, z_{j_1}, z_{j_2}}^{t_1}, \chi_{p_{i_2}, z_{j_3}, z_{j_4}}^{t_2}, \chi_{p_{i_3}, z_{j_5}, z_{j_6}}^{t_3}, \dots \quad (2)$$

где $\chi_{p_{i_1}, z_{j_1}, z_{j_2}}^{t_1}, \chi_{p_{i_2}, z_{j_3}, z_{j_4}}^{t_2}, \chi_{p_{i_3}, z_{j_5}, z_{j_6}}^{t_3}, \dots$ – входные события и выходные события управления обменом;

$t_1 \leq t_2 \leq t_3 \leq \dots$ – упорядоченная последовательность времен событий входного взаимодействия.

В рассматриваемой модели в качестве аргументов функционирования цифровых систем используются входные взаимодействия, что дает возможность рассматривать режимы работы, инициируемые как внешними входными сигналами, так и самими цифровыми системами [1, 13]. Все вышесказанное относится не только к цифровым системам в целом, но и к цифровым блокам, из которых цифровые системы состоят.

III. УЧЕТ ВРЕМЕННЫХ ОГРАНИЧЕНИЙ В МОДЕЛИ ДОПУСТИМЫХ ВЗАИМОДЕЙСТВИЙ

Каждый блок цифровой системы в процессе функционирования выполняет ту или иную последовательность функций (операций) из конечного алфавита функций \mathbf{K} . Выполнение каждой функции вызывается одним из входных взаимодействий определенного класса, причем каждое входное взаимодействие этого класса содержит конечное число событий.

Обозначим через f конечную последовательность функций, а через \mathbf{F} в общем случае счетное множество конечных последовательностей f . Каждая последовательность функций f , начинающаяся с момента времени $t = 0$ (например, включения питания), задается по крайней мере одним входным взаимодействием $\mu^f \in \mathbf{M}$. Этот факт следует из того, что \mathbf{M} содержит все допустимые входные взаимодействия для любой допустимой последовательности функций цифрового блока.

В большинстве случаев одни и те же функции могут выполняться с различными наборами данных, что обуславливает задание различными μ одной и той же последовательности функций f . В связи с тем, что для различных экземпляров блока цифровой системы задержки выходных событий управления обменом относительно входных событий различаются в определенных пределах, а также в связи с

допустимостью варьирования моментов времени входных событий относительно друг друга и относительно выходных событий управления обменом, множество \mathbf{M} содержит континуальное подмножество $\mathbf{M}^f \subset \mathbf{M}$ входных взаимодействий, каждое из которых вызывает выполнение цифровым блоком конечной последовательности функций f . Множество входных взаимодействий может быть представлено в виде:

$$\mathbf{M} = \bigcup_{f \in \mathbf{F}} \mathbf{M}^f; \mathbf{M}^{f'} \cap \mathbf{M}^{f''} = \emptyset \text{ при } f' \neq f''. \quad (3)$$

Входное взаимодействие $\mu \in \mathbf{M}^f$ содержит конечное множество событий

$$\{\chi_{p_{i_1}, z_{j_1}, z_{j_2}}^{t_1}, \dots, \chi_{p_{i_n}, z_{j_{2n-1}}, z_{j_{2n}}}^{t_n}\}, \quad (4)$$

где n – количество событий в μ .

Множество \mathbf{M}^f содержит также входные взаимодействия, времена событий в которых различаются в определенных пределах. Ограничения на эти различия могут быть заданы в виде:

$$t_{\min}^{l,m} \leq t_m - t_l \leq t_{\max}^{l,m}, \\ (l,m) \in \mathbf{C}, \mathbf{C} \subset \{1,2,\dots,n\} \times \{1,2,\dots,n\},$$

где $t_{\min}^{l,m}$, $t_{\max}^{l,m}$ – минимально и максимально допустимые промежутки времени между l -м и m -м событиями;

\mathbf{C} – конечное множество пар событий из (2), для которых заданы временные ограничения.

Выделим в \mathbf{C} все пары (l,m) , для которых t_m есть время выходного события управления обменом, в множество $\mathbf{C}_{\text{вых}}$, а все пары (l,m) , для которых t_m есть время входного события, в множество $\mathbf{C}_{\text{вх}}$. Тогда ограничения на моменты времени выходных событий обмена есть

$$t_{\min}^{l,m} \leq t_m - t_l \leq t_{\max}^{l,m}, (l,m) \in \mathbf{C}_{\text{вых}},$$

а ограничения на моменты времени входных событий

$$t_{\min}^{l,m} \leq t_m - t_l \leq t_{\max}^{l,m}, (l,m) \in \mathbf{C}_{\text{вх}}. \quad (5)$$

Рассмотрим пространство $\mathbf{G} = \text{Пс}_{\text{вых}}\{t_m - t_l | t_m - t_l \geq 0\}$. Каждая точка $g \in \mathbf{G}$ определяет конкретные значения задержек выходных событий управления обменом. В пространстве \mathbf{G} выделим область $\mathbf{G}_f \in \mathbf{G}$, для всех точек которой выполняются ограничения (5). Область \mathbf{G}_f определяет допустимые задержки выходных событий управления обменом. При этом $\mathbf{G}_f \neq \emptyset$.

Для любой точки $g \in \mathbf{G}_f$ в связи с допустимостью таких задержек выходных событий управления обменом существует непустое множество входных взаимодействий \mathbf{M}_g^f , обеспечивающих выполнение цифровым блоком последовательности функций f .

$$\mathbf{M}^f = \bigcup_{g \in \mathbf{G}_f} \mathbf{M}_g^f, \mathbf{M}_g^f \neq \emptyset,$$

где \mathbf{M}_g^f – множество входных взаимодействий, обеспечивающих выполнение конечной последовательности функций f при фиксированных задержках выходных событий управления обменом, определяемых $g \in \mathbf{G}_f$.

Таким образом, множество допустимых входных взаимодействий представимо в виде:

$$\mathbf{M} = \bigcup_{f \in \mathbf{F}} \mathbf{M}^f; \mathbf{M}^{f'} \cap \mathbf{M}^{f''} = \emptyset \text{ при } f' \neq f''; \\ \mathbf{M}^f = \bigcup_{g \in \mathbf{G}_f} \mathbf{M}_g^f; \mathbf{G}_f \neq \emptyset; \mathbf{M}_g^f \neq \emptyset \text{ при } f \in \mathbf{F}. \quad (6)$$

Таким образом, мы определяем структуру множества допустимых входных взаимодействий, то есть структуру возможных аргументов функционирования цифровых блоков.

IV. ЗАДАНИЕ МНОЖЕСТВА ВХОДНЫХ ВЗАИМОДЕЙСТВИЙ ДЛЯ КАЖДОЙ ФУНКЦИИ

Рассмотрим способ задания множества \mathbf{M}^k входных взаимодействий, обуславливающих выполнение цифровым блоком функции k .

Учитывая (2) и (4), каждое μ , $\mu \in \mathbf{M}^k$ может быть задано в виде:

$$((z_{p_1}^h, \dots, z_{p_m}^h), \{(t_1, p_{i_1}, z'_1, z''_1), (t_2, p_{i_2}, z'_2, z''_2), \dots, \\ (t_n, p_{i_n}, z'_n, z''_n)\}), \quad (7)$$

где $z_{p_1}^h, \dots, z_{p_m}^h$ – начальные значения переменных;

$t_i, p_{i_i}, z'_i, z''_i$ – четверка, описывающая i -ое событие.

Исходя из этого, попробуем задать все множество \mathbf{M}^k как:

$$(\tilde{z}_{p_1}^h, \dots, \tilde{z}_{p_m}^h), \{(\theta_1, p_{i_1}, \tilde{z}'_1, \tilde{z}''_1), (\theta_2, p_{i_2}, \tilde{z}'_2, \tilde{z}''_2), \dots, \\ (\theta_n, p_{i_n}, \tilde{z}'_n, \tilde{z}''_n)\}, \quad (8) \\ t_{\min}^{l,q} \leq \theta_q - \theta_l \leq t_{\max}^{l,q}, (l,q) \in \mathbf{C},$$

где $\theta_1, \theta_2, \dots, \theta_n$ – времена событий;

$\tilde{z}_{p_i}^h, i=1,2,\dots,m$ – подмножество возможных начальных значений переменных множества $\mathbf{P}' \cup \mathbf{P}^0$;

m – мощность множества $\mathbf{P}' \cup \mathbf{P}^0$;

$p_{i_1}, p_{i_2}, \dots, p_{i_n}$ – принадлежат множеству $\mathbf{P}' \cup \mathbf{P}^0$;

$\tilde{z}'_1, \tilde{z}'_2, \dots, \tilde{z}'_n$ – подмножества возможных значений переменных $p_{i_1}, p_{i_2}, \dots, p_{i_n}$ перед событием;

$\tilde{z}''_1, \tilde{z}''_2, \dots, \tilde{z}''_n$ – подмножества возможных значений переменных $p_{i_1}, p_{i_2}, \dots, p_{i_n}$ после события;

\mathbf{C} – множество пар событий, для промежутков между которыми заданы временные ограничения.

Входное взаимодействие μ , заданное в виде (7), принадлежит \mathbf{M}^k , определенному в виде (8), если:

$$a) z_{p_1}^h \in \tilde{z}_{p_1}^h, \dots, z_{p_m}^h \in \tilde{z}_{p_m}^h;$$

б) существует изоморфизм между множеством событий (7) и множеством четверок в (8), такой, что у

соответствующих событий совпадают имена переменных: $z'_1 \in \mathbf{Z}'_1, \dots, z'_n \in \mathbf{Z}'_n, z''_1 \in \mathbf{Z}''_1, \dots, z''_n \in \mathbf{Z}''_n$, а времена событий t_1, \dots, t_n в (9) удовлетворяют ограничениям из (8).

Рассматриваемое представление M^k и введение подмножеств $\mathbf{Z}^p_i, \mathbf{Z}', \mathbf{Z}''$ позволяют представить множество входных взаимодействий с различными наборами данных.

Пусть задано множество терминальных переменных \mathbf{P} , каждая переменная с множеством значений \mathbf{Z}_p . Для каждой переменной p рассмотрим некоторый алфавит значений $\mathbf{Z}_p = \mathbf{Z}_p \cup \{z_{ij} | (z_i \in \mathbf{Z}_p) \& (z_j \in \mathbf{Z}_p) \& (z_i \neq z_j)\}$, где z_{ij} – переход от z_i к z_j . В алфавите \mathbf{Z}_p выделим непустые подмножества $\mathbf{Z}^p_i \subset \mathbf{Z}_p$, такие, что $\mathbf{Z}_p = \cup_l \mathbf{Z}^p_i$. Например, если сигнал на шине может принимать значения 0-255, \$ (\$ – состояние с высоким выходным сопротивлением), то возможно использование следующих подмножеств:

$\mathbf{Z}_1 : \mathbf{Z}_1 = \{\$ \}, \mathbf{Z}_2 = \{0, 1, \dots, 255\}, \mathbf{Z}_3 = \mathbf{Z}_2 \cup \{z_{ij} | (z_i \in \mathbf{Z}_2 \cup \mathbf{Z}_1) \& (z_j \in \mathbf{Z}_2 \cup \mathbf{Z}_1)\}, \mathbf{Z}_4 = \mathbf{Z}_1 \cup \mathbf{Z}_2 \cup \mathbf{Z}_3$.

Здесь \mathbf{Z}_2 – стабильное значение данных, \mathbf{Z}_4 – нестабильное значение.

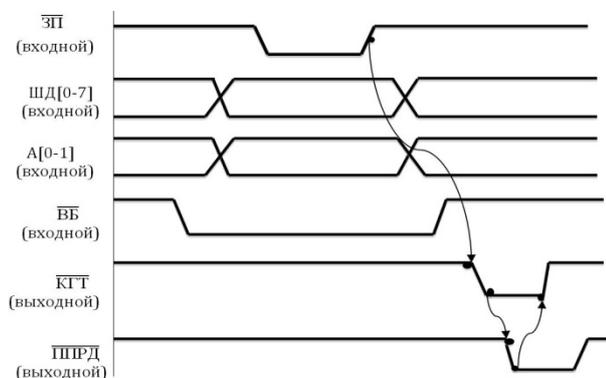
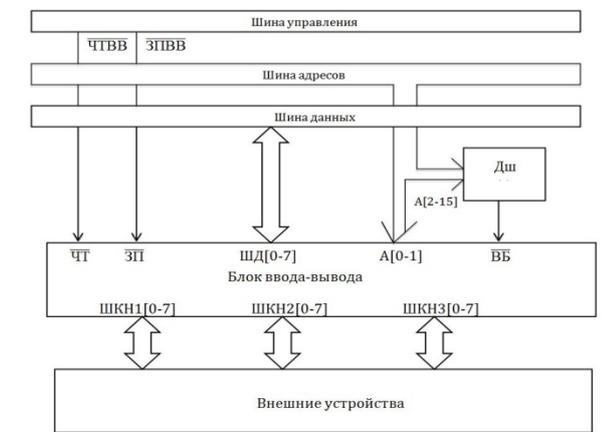


Рис. 1: а) блок параллельного интерфейса ввода-вывода; б) временные диаграммы его работы

Подмножества $\mathbf{Z}^p_l, l=1,2,\dots$ образуют конечный алфавит \mathbf{Z}_p . Каждое множество M^k может быть представлено как конечное множество событий в алфавитах \mathbf{Z}_p . Так как на множестве событий множества M^k определен частичный порядок событий по времени, то естественным представлением M^k является ориентированный граф $G^k(\mathbf{V}^k, \mathbf{E}^k)$, где каждая вершина из \mathbf{V}^k соответствует переходу одной из переменных из одного значения \mathbf{Z}^p_l в другое. Каждую вершину $v, v \in \mathbf{V}^k$ пометим обозначением переменной p и множествами $\mathbf{Z}^p_l, \mathbf{Z}^p_{l'}$, если вершине v соответствует переход переменной p из значения \mathbf{Z}^p_l в значение $\mathbf{Z}^p_{l'}$.

На множестве вершин \mathbf{V}^k_p , соответствующих изменениям значений одной и той же переменной p , задано отношение частичного порядка во времени, что определяет множество ребер \mathbf{E}^k . Каждое ребро пометим двумя числами t_{\min}, t_{\max} , причем $0 \leq t_{\min} \leq t_{\max} \leq \infty$.

В качестве примера рассмотрим некоторый цифровой блок параллельного интерфейса ввода-вывода, представленный на рис. 1а, временные диаграммы работы которого приведены на рис. 1б. При функционировании блока используются сигналы: ШД[0-7] – шина данных; А[0-1], А[2-15] – шины адреса соответствующей разрядности; СБР, ЧТ, ЗП – сигналы сброса, чтения и записи; ВБ, КГТ, ППРД – сигналы выбора блока, канал готов, подтверждения передачи; ШКН1, ШКН2, ШКН3 – шины каналов 1, 2 и 3 с указанием разрядности.

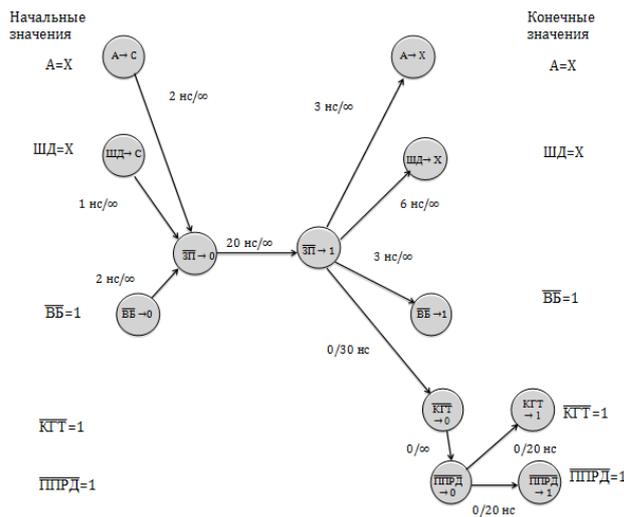


Рис. 2. Задание множества входных взаимодействий для функции записи информации в блок параллельного ввода-вывода

На рис. 2 представлено графовое задание множества M^k входных взаимодействий, соответствующих записи информации в рассматриваемый цифровой блок. Для этого случая:

$\mathbf{P} = \{A, \text{ШД}, \text{ВБ}, \text{ЗП}, \text{КГТ}, \text{ППРД}\}$;

$Z_A = \{0, 1, 2, 3, \$\}$; $Z_{ШД} = \{0, 1, \dots, 255, \$\}$; $Z_{ВВ} = \{0, 1\}$;
 $Z_{ЗП} = \{0, 1\}$; $Z_{КГТ} = \{0, 1\}$; $Z_{ППРД} = \{0, 1\}$;

$\hat{Z}_A = \{C, HC, VI, X\}$, $C = \{0, 1, 2, 3\}$; $HC = CU$
 $\{\chi_{i,k} | z_i \neq z_k, z_i \in C, z_k \in C\}$;
 $VI = \{\$, X = HC \cup VI \cup \{\chi_{i,\$} | z_i \in C\} \cup \{\chi_{\$,i} | z_i \in C\}$;

$\hat{Z}_{ШД} = \{C, HC, VI, X\}$, $C = \{0, 1, \dots, 255\}$;
 $HC = CU \{\chi_{i,k} | z_i \neq z_k, z_i \in C, z_k \in C\}$;
 $VI = \{\$, X = HC \cup VI \cup \{\chi_{i,\$} | z_i \in C\} \cup \{\chi_{\$,i} | z_i \in C\}$;

$\hat{Z}_{ВВ} = \{0, 1, X\}$, $X = \{0, 1, \beta_{0,1}, \beta_{1,0}\}$;

$\hat{Z}_{ЗП} = \{0, 1, X\}$, $X = \{0, 1, \beta_{0,1}, \beta_{1,0}\}$;

$\hat{Z}_{КГТ} = \{0, 1, X\}$, $X = \{0, 1, \beta_{0,1}, \beta_{1,0}\}$;

$\hat{Z}_{ППРД} = \{0, 1, X\}$, $X = \{0, 1, \beta_{0,1}, \beta_{1,0}\}$.

На рис. 2 вершины графа помечены обозначением переменной и новым ее значением. Если заданы начальные значения переменных в алфавите \hat{Z}_p , этого достаточно. При наличии графа $G^k(V^k, E^k)$ указанного вида для каждого μ , заданного в виде (7), можно определить, принадлежит ли μ множеству M^k или нет.

Граф $G^k(V^k, E^k)$ более наглядно задает множество M^k , чем временная диаграмма (рис. 1б), которая обычно используется для представления режимов работы цифровых блоков.

V. ЗАКЛЮЧЕНИЕ

Граф $G^k(V^k, E^k)$ определяет множество входных взаимодействий для выполнения функции k цифровым блоком. Используя тот же алгоритм, можно построить аналогичный граф для каждой функции цифровой системы в целом.

Этот граф вместе с выражениями (3) и (6) определяет структуру множества допустимых входных взаимодействий как для каждого цифрового блока в отдельности, так и для цифровой системы в целом. Структура множества допустимых взаимодействий служит исходными данными для выбора набора тестов для отладки проектов цифровых систем методом моделирования.

ПОДДЕРЖКА

Работа выполнена при поддержке гранта РФФИ № 17-07-00683.

ЛИТЕРАТУРА

[1] Иванников А.Д., Стемповский А.Л. Формализация задачи отладки проектов цифровых систем // Информационные технологии. 2014. № 9. С. 3-10.
 [2] Lin, Yi-Li; Su, Alvin W.Y. Functional Verification for SoC Software/Hardware Co-Design: From Virtual Platform to Physical Platform // 2011 IEEE International SOC Conference (SOCC), pp. 201-206.
 [3] Matsuda, A.; Ishihara, T. Developing an Integrated Verification and Debug Methodology // Design, Automation

& Test in Europe Conference & Exhibition (DATE), 2011, pp. 1-2.
 [4] Shi, Jin; Liu, Weichao; Jiang, Ming; et al. Software Hardware Co-Simulation and Co-Verification in Safety Critical System Design // 2013 IEEE International Conference on Intelligent Rail Transportation (ICIRT), pp. 71-74.
 [5] Иванников А.Д. Формирование отладочного набора тестов для проверки функций цифровых систем управления объектами // Мехатроника, автоматизация, управление. 2017. Т. 18. № 12. С. 795-801.
 [6] Кашеев Н.И., Пономарев Д.М., Подъяблонский Ф.М. Построение тестов цифровых схем с использованием обобщенной модели неисправностей и непрерывного подхода к моделированию // Вестник Нижегородского университета им. Н.И.Лобачевского. 2011. №3 (2). С. 72-77.
 [7] Cruz, A.M., Fernandez, R.B., Lozano, H.M., Ramirez Salinas, M.A., Vila Vargas, L.A. Automated Functional Test Generation for Digital Systems Through a Compact Binary Differential Evolution Algorithm // Journal of Electronic Testing-Theory and Applications. 2015. V. 31. № 4. P. 361-380.
 [8] Иванников А.Д. Анализ методов декомпозиции задачи отладки проектов цифровых систем // Информационные технологии. 2016. Т7 22. № 10. С. 758-763.
 [9] Стемповский А.Л., Гаврилов С.В., Глебов А.Л. Методы логического и логико-временного анализа цифровых КМОП СБИС. М.: Наука, 2007. 220 с.
 [10] Jasnetski, A., Oyeniran, S. A., Tsertoy, A. High-Level Modeling and Testing of Multiple Control Faults in Digital Systems // IEEE 19th International Symposium on Design and Diagnostics of Electronic Circuits & Systems (DDECS). 2016. Paper # 7482445.
 [11] Березкин А.В., Федотов А.А., Филиппов А.С. Тестирование цифровых систем, заданных высокоуровневыми спецификациями // Научно-технические ведомости Санкт-Петербургского государственного политехнического университета. Информатика. Телекоммуникации. Управление. 2011. Т. 6-1. № 138. С. 62-70.
 [12] Jain, S., Govani, P., Poddar, K.B., Lal, A.K., Parmar, R.M. Functional verification of DSP based on-board VLSI design // International Conference on VLSI Systems, Architectures, Technology and Applications (VLSI-SATA). 2016. P. 1-4.
 [13] Гаврилов С.В., Иванова Г.А., Стемповский А.Л. Теоретико-графовая модель сложно-функциональных блоков для КМОП технологий с трехмерной структурой транзистора // Известия ЮФУ. Технические науки. 2014. № 7 (156). С. 58-68.
 [14] Jasnetski A., Oyeniran S.A., Tsertoy A. High Level Modeling and Testing Of Multiple Control Faults in Digital Systems // IEEE 19th International Symposium on Design and Diagnostics of Electronic Circuits & Systems (DDECS). 2016. Paper # 7482445.
 [15] Иванников А.Д., Стемповский А.Л. Математическая модель отладки проектов сложных цифровых систем и микросистем на основе представления последних в виде семейства стационарных динамических систем // Проблемы разработки перспективных микро- и нанoeлектронных систем (МЭС). 2014. Часть II. С. 123-128.

Research and Development of Digital System Block Models Based on their Description as a Stationary Dynamical System Family

A.D. Ivannikov

Institute for Design Problems in Microelectronics of Russian Academy of Sciences (IPPM RAS),
adi@ippm.ru

Abstract — While digital system design debugging by computer simulation the important task is to generate debugging test set, e.g. set of input signals which are applied to a designing system computer model for checking the correctness of its functioning. The generation of complete in some sense debugging test set is possible by some way if the permissible input action set for the system is known. Description forming of such a set is possible if permissible input interaction set for digital system blocks are known. Digital system block model investigation is carried out, first of all, from the point of a set of permissible input interactions. The family of stationary dynamic systems with continuous time and logical signal discrete values are used as models for digital system blocks. In some cases signal exchange between blocks and with outer world is initiated by a block itself. That is why input interactions including input signals and output exchange driving signals are considered as debugging tests. For the description of permissible input interactions of digital system blocks and the system as a whole graph representation is proposed for each fulfilled function.

Keywords — digital system logical and timing analysis, debugging by simulation, input interaction set structure, digital block input interactions, graph representation for input interaction set.

REFERENCES

- [1] Ivannikov A.D., Stempkovsky A.L. Formalizatsiya zadachi otladki proektov cifrovih system (Formal Model of Digital System Design Debugging Task). *Informacionnie Technologii*, 2014, no. 9, pp. 3-10 (in Russian).
- [2] Lin, Yi-Li; Su, Alvin W.Y. Functional Verification for SoC Software/Hardware Co-Design: From Virtual Platform to Physical Platform. 2011 IEEE International SOC Conference (SOCC), pp. 201-206.
- [3] Matsuda, A.; Ishihara, T. Developing an Integrated Verification and Debug Methodology. *Design, Automation & Test in Europe Conference & Exhibition (DATE)*, 2011, pp. 1-2.
- [4] Shi, Jin; Liu, Weichao; Jiang, Ming; et al. Software Hardware Co-Simulation and Co-Verification in Safety Critical System Design. 2013 IEEE International Conference on Intelligent Rail Transportation (ICIRT), pp. 71-74.
- [5] Ivannikov A.D. Formirovanie otladochnogo nabora testov dlya proverki funktsiy cifrovih system upravleniya obektami (Debugging Input Set Generation for Testing of Control Digital Systems Functions). *Mekhatronika, Avtomatizatsiya, Upravlenie*, 2017, vol. 18, no.12, pp. 795-801.
- [6] Kasheev N.I., Ponomarev D.M., Podyablonsky F.M. Postroenie testov cifrovih chem. C ispolzovaniem obobshennoi modeli neispravnosti i neprerivnogo podhoda k modelirovaniu (Digital Circuits Test Generation Based on Generalized Malfunction Model and Continuous Simulation Approach). *Vestnik Nijegorodskogo Universiteta*, 2011, no. 3(2), pp. 72-77 (in Russian).
- [7] Cruz, A.M., Fernandez, R.B., Lozano, H.M., Ramirez Salinas, M.A., Vila Vargas, L.A. Automated Functional Test Generation for Digital Systems Through a Compact Binary Differential Evolution Algorithm // *Journal of Electronic Testing-Theory and Applications*. 2015. V. 31. № 4. P. 361-380.
- [8] Ivannikov A.D. Analiz metodov dekompozitsii zadachi otladki proektov cifrovih system (Decomposition Methods Analysis for Digital System Design Debugging). *Informacionnie Technologii*, 2016, vol. 22, no. 10, pp. 758-763.
- [9] Stempkovsky F.L., Gavrilov S.V., Glebov A.L. Metodi logicheskogo i logiko-vremennogo analiza cifrovih CMOS VLSI (Logical and Logical-Timing Analysis for Digital CMOS VLSI). Moscow, "Nauka", 2007, 220 p. (in Russian).
- [10] Jasnetski, A., Oyeniran, S. A., Tsertoy, A. High-Level Modeling and Testing of Multiple Control Faults in Digital Systems // *IEEE 19th International Symposium on Design and Diagnostics of Electronic Circuits & Systems (DDECS)*. 2016. Paper # 7482445.
- [11] Berezkin A.V., Fedotov A.A., Filippov A.S. Testirovanie cifrovih system, zadannih visokourovnevimi specifikatsiyami (Testing of Digital Systems, Defined by High Level Specifications). *Nauchno-Tekhnicheskie Vedomosti Sankt-Peterburgskogo Gosudarstvennogo Politehnicheskogo Universiteta. Informatika. Telekommunikatsii. Upravlenie*, 2011, vol. 6-1, no. 138, pp.62-70.
- [12] Jain, S., Govani, P., Poddar, K.B., Lal, A.K., Parmar, R.M. Functional verification of DSP based on-board VLSI design. *International Conference on VLSI Systems, Architectures, Technology and Applications (VLSI-SATA)*. 2016. P. 1-4.
- [13] Gavriliv S.V., Ivanova G.A., Stempkovsky A.L. Teoretiko-grafovaya model slojno-funktsionalnih blokov dlya CMOS tehnologii s trehmernoy strukturoy transistor (Theoretical Graph Model of Complex Functional Blocks for CMOS Technology with Three Dimension Transistor Structure). *Izvestiya UFU. Tehnicheskie Nauki*, 2014, no. 7 (156), pp.58-68 (in Russian).
- [14] Jasnetski A., Oyeniran S.A., Tsertoy A. High Level Modeling and Testing Of Multiple Control Faults in Digital Systems // *IEEE 19th International Symposium on Design and Diagnostics of Electronic Circuits & Systems (DDECS)*. 2016. Paper # 7482445.
- [15] Ivannikov A.D., Stempkovsky A.L. Matematicheskaya model otladki proektov slojnih cifrovih system i mikrosystem na osnove predstavleniya poslednih v vide semeistva stacionarnih dinamicheskikh system (Design Debugging Mathematical Model for Complex Digital and Microsystems on the Basis of Their Representation as a Family of Stationary Dynamic Systems). *Problemi Rasrabotki Perspektivnih Mikro- i Nanjelectronnih System (MES)*, 2014, no. II, pp. 123-128.