

УДК 004.27:519.6:681.5

И. Ф. Чебурахин, д-р техн. наук, проф., cybernetics@mati.ru,

МАТИ—РГТУ имени К. Э. Циолковского, Москва,

С. В. Шалагин, д-р техн. наук, доц., sshalagin@mail.ru,

Казанский национальный исследовательский технический университет им. А. Н. Туполева — КАИ

Конвейерное вычисление булевых функций на основе однотипных IP-ядер в архитектуре ПЛИС/Virtex

Исследована и решена задача реализации произвольных булевых функций на основе структурно-функциональной параллельной (логической) декомпозиции в архитектуре ПЛИС семейства Virtex. Предложен метод синтеза устройств для вычисления булевых функций при использовании однотипных IP-ядер. За счет применения конвейерной обработки данных с сохранением промежуточных результатов достигается высокое быстродействие указанных устройств на ПЛИС/Virtex.

Ключевые слова: булевы функции, синтез формул и схем, декомпозиция, сложность, минимизация, функциональные уравнения

Введение

В настоящее время актуальна задача синтеза цифровых устройств на многопроцессорных вычислительных системах с программируемой архитектурой (МВС ПА), элементами которой являются программируемые логические интегральные схемы (ПЛИС) класса FPGA [1, 2]. Перспективным подходом к решению данной задачи является синтез указанных устройств на основе однотипных IP-ядер (англ. Intellectual Property) [3, 4], определенных в архитектуре FPGA, причем на одном корпусе ПЛИС/FPGA реализуемо не более чем одно IP-ядро. В работе [5] показано развитие указанного подхода на решение задачи синтеза генераторов дискретных стохастических процессов класса марковских и их функций, а также устройств цифровой обработки сигналов, реализующих дискретные ортогональные преобразования, в том числе — дискретные преобразования Фурье, Хартли и цифровую фильтрацию с конечной импульсной характеристикой, над двоичными числами разрядности n , $n \leq 8$ [6].

Согласно работам [5, 7] данный класс генераторов и устройств реализуем при использовании однотипных IP-ядер, выполняющих произвольную булеву функцию f от заданного числа переменных w — функцию $f^{(w)}$.

От аппаратной сложности и быстродействия цифровых схем зависят основные характеристики устройств, описываемых данными схемами [8—12]. В работах [13—15] исследованы вопросы реализации и минимизации $f^{(w)}$ в классе формул и схем из функциональных элементов в различных базисах.

В данной работе рассматривается задача реализации устройств для вычисления $f^{(d)}$, определенных

в архитектуре ПЛИС/FPGA семейства Virtex [16—18]. ПЛИС класса FPGA включает в свой состав однотипные конфигурируемые логические блоки (англ. Look-Up Tables, LUT), реализующие произвольную $f^{(d)}$. Указанные LUT, а также блоки ввода-вывода и межсоединения (МС) примем в качестве базиса ПЛИС [19], применимого для реализации $f^{(w)}$ при условии, что $w > d$. В работах [5, 20] исследованы теоретические оценки сложности вычисления дискретной детерминированной нелинейной функции при использовании системы $f^{(w)}$, однако вопросы оценки сложности и степени соответствия архитектуре ПЛИС/FPGA для устройств, реализующих $f^{(w)}$, исследованы не достаточно.

Среди современных ПЛИС, выпускаемых Xilinx Inc., ПЛИС семейства Virtex [17, 18] в большей степени, чем другие семейства, ориентированы на реализацию логических функций, в частности, $f^{(w)}$. ПЛИС семейства Virtex-4 [17] применяются как элементы МВС ПА "Медведь", изготавливаемых в НИИ МВС (г. Таганрог) [1, 2]. В данной работе рассмотрены и сопоставлены возможности по реализации $f^{(w)}$ при использовании структурно-функциональной параллельной (логической) декомпозиции (СФПД) [13—15] на ПЛИС семейств Virtex-4 [17] и Virtex-7 с применением специализированной САПР ISE 13.4 (Xilinx Corp.), далее — САПР [18]. Получены оценки временной сложности и реальных затрат конфигурируемых ресурсов ПЛИС/Virtex для однотипных IP-ядер, позволяющих реализовать устройство для вычисления $f^{(w)}$, $w > d$. Определена степень соответствия IP-ядер, участвующих в вычислении $f^{(w)}$ на основе декомпозиции СФПД, архитектуре ПЛИС/Virtex.

Теоретический анализ реализации вычисления $f^{(w)}$ на ПЛИС/FPGA

Вычисление произвольной функции $f^{(w)}$ реализуемо в два этапа: на первом этапе вычислимы $2^{(w-d)}$ значений $f^{(d)}$, на втором — мультиплексирование $2^{(w-d)}$ значений на один выход при использовании $(w-d)$ адресных входов [15]. В случае сохранения промежуточных результатов на выходах оценка временной сложности вычисления значения $f^{(w)}$ составляет

$$T(f^{(w)}) = \max(T(f^{(d)}), T(mx2^{(w-d)} \rightarrow 1)) + T_D, \quad (1)$$

где $T(f^{(d)})$, $T(mx2^{(w-d)} \rightarrow 1)$ и T_D — время задержки функционирования IP-ядер, реализующих вычисление $f^{(d)}$ и мультиплексор " $2^{(w-d)} \rightarrow 1$ ", а также время задержки запоминающего устройства — D -триггера, соответственно.

Внутри ПЛИС семейства Virtex проводится вычисление $f^{(d)}$ при использовании однотипных генераторов функций от d переменных — $LUT(d)$, $d \in \{4, 5, 6\}$, а сохранение промежуточных результатов — при использовании однотипных D -триггеров. $LUT(d)$ применяемы и для реализации мультиплексоров. В соответствии с известной методикой [5, 7] возможность реализации на одной ПЛИС вычислительного устройства, приближенного к оптимальному по времени задержки функционирования, обеспечивается при условии, что будут задействованы не более 50 % всех конфигурируемых ресурсов — $LUT(d)$, D -триггеров и блоков ввода-вывода (БВВ, англ. сокращение IOB). Логические ресурсы ПЛИС семейства Virtex организованы в "слайсы" (от англ. "slices" — слои), включающие в себя вышеуказанные конфигурируемые логические ресурсы — мультиплексоры, $LUT(d)$ и D -триггеры [16–18]. Для элементов в (1) имеют место следующие нижние оценки временной сложности:

$$T(f^{(d)}) \geq t_{LUT(d)},$$

где $t_{LUT(d)}$ — время задержки $LUT(d)$; для мультиплексора " $2^{(w-d)} \rightarrow 1$ ":
если $d \in \{3, 4, 5\}$, то $T(mx2^{(w-d)} \rightarrow 1) \geq (w-d)t_{LUT(d)}$,
если $d \in \{6, 7, 8, 9, 10\}$ и $w \bmod 2 = 0$, то $T(mx2^{(w-d)} \rightarrow 1) \geq 0,5(w-d)t_{LUT(d)}$.

Достаточно существенный вклад (до 70 %) в указанные оценки вносят МС ПЛИС/FPGA [19]. На ПЛИС семейства Virtex существует возможность применения локальных линий МС (внутри одного слоя — "слайса") при реализации мультиплексоров " $2^{(w-d)} \rightarrow 1$ ", $w > d$ [16–18], что серьезно снижает вклад МС в общее время задержки их функционирования. Данные обстоятельства позволяют выполнить мультиплексирование " $2^{(w-d)} \rightarrow 1$ " при использовании системы мультиплексоров " $2^z \rightarrow 1$ " с сохранением промежуточных результатов при следующих условиях: $(w-d) \bmod z = 0$, $w > d$. На выбор характеристики z влияет условие, связанное

с быстродействием IP-ядер, реализующих мультиплексор на ПЛИС заданного семейства Virtex: $T(f^{(d)}) \cong T(mx2^z \rightarrow 1)$. Тогда конвейерное вычисление $f^{(w)}$ проводится в $(w-d-z+1)$ этапов, а оценка (1) имеет вид:

$$T_3 = \max(T_{\min}(LUT(d)), T_{\min}(mx2^z \rightarrow 1)) + T_D + T_{IOB} + T_{MC}, \quad (2)$$

где $T_{\min}(LUT(d)) = t_{LUT(d)}$, $T_{\min}(mx2^z \rightarrow 1) = l \cdot t_{LUT(d)}$ — нижние оценки временной сложности вычисления $f^{(d)}$ и мультиплексирования " $2^z \rightarrow 1$ ", T_D — время задержки D -триггера, T_{IOB} и T_{MC} — оценки времени задержки IOB и МС внутри ПЛИС. Рост значения выражения (2) происходит за счет величины T_{MC} , тогда как величины $t_{LUT(d)}$, T_{IOB} и T_{MC} имеют постоянные значения для заданного типа ПЛИС семейства Virtex.

Рассмотрим возможность вычисления $f^{(w)}$ на ПЛИС различных семейств серии Virtex. Для ПЛИС семейства Virtex-4 целесообразно, чтобы $d = 4$, так как внутри микросхем данного семейства присутствуют до 178 тыс. $LUT(4)$ и столько же D -триггеров. Кроме того, для хранения значений на входе и выходе микросхемы применимо до 960 D -триггеров — по одному внутри каждого из IOB указанной ПЛИС. Для реализации мультиплексора " $2^{(w-d)} \rightarrow 1$ " внутри ПЛИС/Virtex-4 требуется задействовать $2^{(w-4)} - 1$ мультиплексора " $2 \rightarrow 1$ ", причем каждый из мультиплексоров " $2 \rightarrow 1$ " реализуем на $LUT(4)$. Для вычисления $f^{(w)}$, $w > 4$, на ПЛИС/Virtex-4 требуется $(2^{(w-3)} - 1) LUT(4)$, из которых $2^{(w-4)}$ — для реализации множества $f^{(4)}$, $(2^{(w-4)} + 1)$ для реализации D -триггеров и $(w+1)$ для реализации IOB. В формуле (2) для ПЛИС/Virtex-4 $l = w - d$.

Для ПЛИС семейства Virtex-7 примем $d = 6$: внутри микросхемы xc7vx980t-2flg1930 присутствуют до 612 тыс. $LUT(6)$, до 1,22 млн D -триггеров и до 900 IOB. Для реализации мультиплексора " $2^{(w-d)} \rightarrow 1$ " на ПЛИС семейства Virtex-7 требуется $(2^{(w-6)} - 1)/3$ мультиплексоров " $4 \rightarrow 1$ ", каждый из которых требует для своей реализации по одному $LUT(6)$. Вычисление $f^{(w)}$, $w > 6$, на ПЛИС Virtex-7 реализуемо при использовании $(2^{(w-4)} - 1)/3 LUT(6)$, из которых $2^{(w-6)} - 1$ для реализации множества $f^{(6)}$, $1 + (2^{(w-6)} - 1)/3$ для реализации D -триггеров и $(w+1)$ для реализации IOB. В формуле (2) для ПЛИС/Virtex-7 $l = 0,5(w-d)$.

ПЛИС класса FPGA включает в свой состав $LUT(d)$, $d \leq 10$, число которых, требуемых для реализации мультиплексора " $2^{(w-d)} \rightarrow 1$ ", находится согласно формуле

$$Q(mx2^{(w-d)} \rightarrow 1) = \begin{cases} d \in \{3, 4, 5\}: 2^{(w-d)} - 1; \\ d \in \{6, 7, 8, 9, 10\}: (2^{(w-d)} - 1)/3. \end{cases} \quad (3)$$

В результате справедливо следующее утверждение.

Утверждение 1. Вычисление $f^{(w)}$ на ПЛИС, включающей в свой состав $LUT(d)$, выполняется при использовании $2^{(w-6)} + Q(mx2^{(w-d)} \rightarrow 1)$ $LUT(d)$, $2^{(w-6)} + 1 + Q(mx2^{(w-d)} \rightarrow 1)$ D -триггеров и $(w+1)$ IOB, где $Q(mx2^{(w-d)} \rightarrow 1)$ получено согласно (3).

Метод вычисления $f^{(w)}$ на ПЛИС/Virtex на основе IP-ядер

Определим метод вычисления $f^{(w)}$ на основе структурно-функциональной параллельной (логической) декомпозиции на ПЛИС Virtex при использовании конвейерной схемы, приведенной на рис. 1. Внутри первой ступени конвейера реализуемо вычисление $f^{(d)}$, а внутри второй и последующих ступеней — мультиплексор " $2^z \rightarrow 1$ ", $(w-d) \bmod z = 0$, $w > d$. Имеет место следующее утверждение.

Утверждение 2. Общее число ступеней конвейерной схемы для вычисления $f^{(w)}$ на основе структурно-функциональной параллельной (логической) декомпозиции при использовании однотипных IP-ядер, реализующих $f^{(d)}$ и мультиплексоры " $2^z \rightarrow 1$ ", равно $(w-d-z+1)$.

Ступени конвейерной схемы реализуемы при использовании однотипных IP-ядер. Первая ступень реализуема при использовании $2^{(w-d)}$ IP-ядер, осуществляющих вычисление заданной $f^{(d)}$. Вторая и последующая ступени реализуемы при использовании $2^{(w-d-z)}$ IP-ядер, реализующих мультиплексор " $2^z \rightarrow 1$ ". Обозначим указанные IP-ядра как IP-1 и IP(z), где параметр z определяет разрядность мультиплексора " $2^z \rightarrow 1$ ". Выбор z проводится в целях минимизации значения в выражении (2).

Для ПЛИС семейств Virtex-4 и Virtex-7 IP-1 позволяет вычислить $f^{(4)}$ и $f^{(6)}$, соответственно. Это связано с тем, что для реализации данного ядра требуется только один однотипный элемент заданной ПЛИС — $LUT(4)$ и $LUT(6)$, соответственно.

IP(z), реализуемые на ПЛИС семейств Virtex-4 и Virtex-7, позволяют выполнить мультиплексирование " 2^z в 1". Для ПЛИС каждого семейства z выбираем таким образом, чтобы минимизировать значение (2).

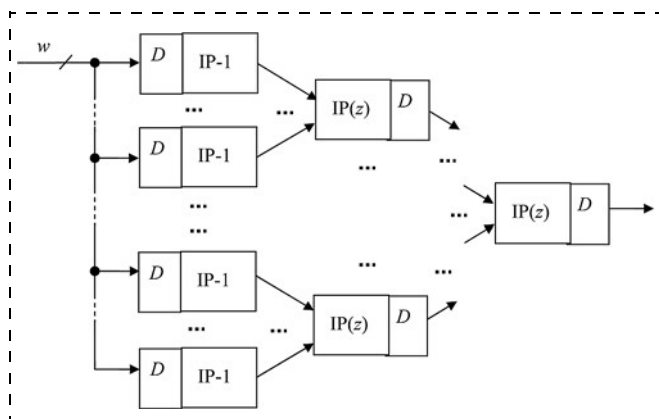


Рис. 1. Конвейерная схема вычисления $f^{(w)}$

Что касается ограничений на аппаратную сложность при реализации $f^{(w)}$ на основе IP-1 и IP(z), то они заданы на основе следующего выражения:

$$\min \frac{q_{(LUT(d))}}{Q_{(LUT(d))}}, \frac{q_{(D)}}{Q_{(D)}}, \frac{q_{(Sl)}}{Q_{(Sl)}}, \frac{q_{(IOB)}}{Q_{(IOB)}} \leq 0,5, \quad (4)$$

где $q_{(LUT(d))}$, $q_{(D)}$, $q_{(Sl)}$ и $q_{(IOB)}$ — общее число $LUT(d)$, D -триггеров, слайсов и IOB, задействованных для реализации устройства, позволяющего вычислить $f^{(w)}$; $Q_{(LUT(d))}$, $Q_{(D)}$, $Q_{(Sl)}$ и $Q_{(IOB)}$ — общее число указанных логических ресурсов, доступных для реконfigurирования внутри заданной ПЛИС семейства Virtex.

На основе утверждений 1 и 2 предложен метод вычисления $f^{(w)}$ при заданных параметрах d и z , включающий три этапа.

Этап 1. Определение числа IP-1 и IP(z), требуемых для вычисления $f^{(w)} = 2^{(w-d)}$ и $2^{(w-d-z)}$, соответственно.

Этап 2. Определение затрат реконfigurируемых ресурсов с учетом ограничения (4) и времени задержки функционирования аппаратных модулей, реализующих IP-1 и IP(z), на ПЛИС семейства Virtex с применением САПР.

Этап 3. Оценка времени задержки функционирования устройства, реализующего вычисление $f^{(w)}$, согласно формуле

$$T_{f^{(w)}} = \max(T_{IP-1}, T_{IP(z)}), \quad (5)$$

где T_{IP-1} и $T_{IP(z)}$ — оценки временной сложности для IP-1 и IP(z), соответственно.

Замечание 1. Предложенный метод позволяет определить максимальное число переменных функции $f^{(w)}$, реализуемой внутри заданной ПЛИС семейства Virtex [5].

Оценки сложности для IP-ядер, реализуемых на ПЛИС/Virtex

Конвейерное вычисление $f^{(w)}$ на основе структурно-функциональной параллельной (логической) декомпозиции проводится при использовании $2^{(w-d)}$ IP-1 для вычисления $f^{(d)}$, а также $2^{(w-d-z)}$ IP(z), каждое из которых реализует мультиплексор " 2^z в 1".

Применение САПР позволяет определить число задействованных реконfigurируемых элементов и быстродействие аппаратных модулей, реализующих IP-1 и IP(z), $z = 4, 6$, на ПЛИС Virtex-4 и Virtex-7. Указанные параметры приведены в табл. 1 и 2 соответственно. При использовании САПР для каждого IP-ядра определены: число $LUT(d)$, $d = 4, 6$, БВВ (IOB), D -триггеров, в том числе применяемых в БВВ (D -триггеров IOB), число задействованных слайсов (slices) [3, 4, 17, 18], общее время задержки функционирования (T_3), время задержки за счет МС (T_{MC}), а также за счет LUT и БВВ ($T_L = T_3 - T_{MC}$). Согласно [19] определен вклад K_{MC} в общее время

Оценки сложности для IP-ядер, реализуемых на Virtex-4

IP-ядро	LUT(4)	IOB	D-триггеры	D-триггеры IOB	Slices	T_3 , нс	T_{MC} , нс	T_L , нс	K_{MC} , %
IP-1	1	6	1	4	1	4,90	1,06	3,84	21,6
IP(5)	16	39	33	0	24	4,60	0,76	3,84	16,7
IP(6)	33	72	65	0	49	5,61	2,67	2,94	47,6
Total	178 176	960	178 176	960	89 088	xc4vlx200-10ff1513			

Таблица 2

Оценки сложности для IP-ядер, реализуемых на Virtex-7

IP-ядро	LUT(6)	IOB	D-триггеры	D-триггеры IOB	Slices	T_3 , нс	T_{MC} , нс	T_L , нс	K_{MC} , %
IP-1	1	8	1	6	1	3,63	0,97	2,66	26,7
IP(4)	53	22	17	0	20	3,68	1,02	2,66	27,7
IP(5)	106	39	33	0	51	3,77	1,11	2,66	29,4
Total	612 000	900	1 224 000	900	153 000	xc7vx980t-2flg1930			

задержки функционирования $T_{MC} : K_{MC} = T_{MC}/T_3$, выраженный в процентах. В последних строках табл. 1 и 2 приведены характеристики определенных микросхем ПЛИС семейства Virtex, на которых проводилась реализация устройства для вычисления $f^{(w)}$.

На рис. 2 показан фрагмент ПЛИС xc7vx980t-2flg1930, на которой реализовано ядро IP(4).

Решение задачи о размещении при выполнении ограничения (3) и минимизации значения (2) дает следующие результаты. Реализация $f^{(w)}$ на ПЛИС/Virtex-4 возможна для $w \leq 18$. Реализация $f^{(w)}$ на ПЛИС/Virtex-7 допустима для $w \leq 22$. Верхняя граница для значения w в обоих случаях определена числом конфигурируемых ресурсов ПЛИС, доступных пользователю.

Замечание 2. Реализация $f^{(w)}$ на ПЛИС/Virtex-4 для $w \leq 4$ выполнима на одном LUT(4), а реализация на ПЛИС/Virtex-7 при $w \leq 6$ — на одном LUT(6).

Подставляя в формулу (5) данные по оценкам временной сложности из табл. 1, получим, что минимальное значение $T_{f^{(w)}} = 4,90$ нс, если для реализации устройства для вычисления $f^{(w)}$ применены IP(5), и $T_{f^{(w)}} = 5,61$ нс, если $f^{(w)}$ вычислена при ис-

пользовании IP(6). Когда значение (5) определено на основе данных из табл. 2, оценка $T_{f^{(w)}}$ составляет 3,68 нс и 3,77 нс, если для вычисления $f^{(w)}$ применены IP(4) и IP(5), соответственно.

В результате внутри одного корпуса ПЛИС Virtex-4 (на примере ПЛИС xc4vlx200-10ff1513) реализуемо устройство для вычисления $f^{(w)}$, $w \leq 18$, максимальная частота функционирования которого определена в соответствии с формулой (4) в диапазоне 178...204 МГц. Внутри одного корпуса ПЛИС Virtex-7 (на примере ПЛИС xc7vx980t-2flg1930) реализуемо устройство для вычисления $f^{(w)}$, $w \leq 22$, максимальная частота функционирования которого определена в диапазоне 265...272 МГц в соответствии с формулой (5).

Заключение

Итогом проведенных исследований является метод расчета временной и аппаратной сложности вычисления булевой функции от w переменных на ПЛИС/Virtex при использовании структурно-функциональной параллельной (логической) декомпозиции [13—15]. Определены ограничения на число переменных w при реализации $f^{(w)}$ на указанных ПЛИС. Применение предложенной методики позволяет определить, что для ПЛИС семейства Virtex-7 оценки времени задержки функционирования устройства, реализующего $f^{(w)}$, $w \leq 22$, будет примерно в 1,3...1,5 раза меньше, чем для устройства, реализующего $f^{(w)}$, $w \leq 18$, на ПЛИС семейства Virtex-4. Повышение быстродействия достигается за счет снижения для ПЛИС семейства Virtex-7 времен задержки функционирования LUT(d) при увеличении значения d с четырех (для ПЛИС Virtex-4) до шести. Следует отметить, что максимальная частота функционирования устройства, реализующего $f^{(w)}$, ограничена сверху не только характеристиками конфигурируемых элементов ПЛИС/FPGA, но и линиями связи вне корпуса ПЛИС.

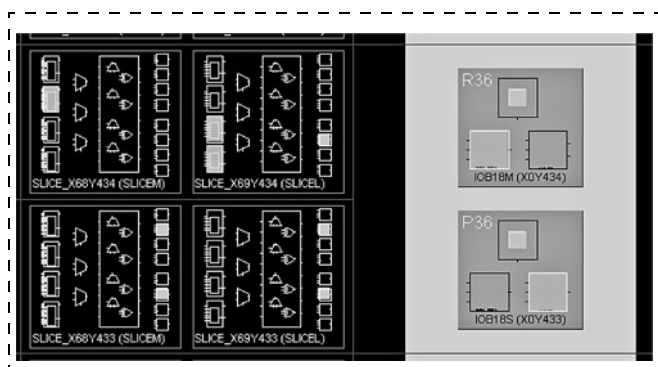


Рис. 2. Фрагмент топологии ПЛИС Virtex-7 для реализации IP(4)

Список литературы

1. **Каляев И. А., Левин И. И.** Реконфигурируемые мультимиконвейерные вычислительные системы для решения потоковых задач обработки информации и управления // Параллельные вычисления и задачи управления (РАСО'10): пленарные докл. 5-й Междунар. конф. 26—28 окт. 2010. М.: ИПУ РАН, 2010. С. 23—37.
2. **Реконфигурируемые** мультимиконвейерные вычислительные структуры / И. А. Каляев, И. И. Левин, Е. А. Семерников и др. 2-е изд. Ростов н/Д: Изд-во ЮНЦ РАН, 2009. 344 с.
3. **Зотов В. Ю.** Проектирование встраиваемых микропроцессорных систем на основе САПР фирмы Xilinx. М.: Горячая линия — Телеком, 2006. 522 с.
4. **Зотов В. Ю.** Средства проектирования встраиваемых микропроцессорных систем, реализуемых на основе ПЛИС фирмы Xilinx // Современная электроника. 2006. № 7. С. 60—67.
5. **Шалагин С. В.** Реализация устройств вычислительной техники на многопроцессорных системах с программируемой архитектурой // Вестник МарГТУ. 2011. № 1 (11). С. 38—46.
6. **Шалагин С. В.** Моделирование дискретных марковских процессов на распределенной вычислительной системе с программируемой архитектурой // Динамика нелинейных дискретных электротехнических и электронных систем: материалы 10-й Всерос. науч.-техн. конф. 6—8 июня 2013. Чебоксары: Изд-во Чуваш. ун-та, 2013. С. 60—61.
7. **Захаров В. М., Шалагин С. В.** Вычисление нелинейных полиномиальных функций на многопроцессорной системе с программируемой архитектурой // Информационные технологии. 2012. № 5. С. 6—11.
8. **Журавлев Ю. И.** Теоретико-множественные методы в алгебре логики // Проблемы кибернетики. 1962. № 8 (1). С. 25—45.
9. **Кудрявцев В. Б., Андреев А. Е.** О сложности алгоритмов // Фундаментальная и прикладная математика. 2009. № 15 (3). С. 135—181.

10. **Лупанов О. Б.** О сложности реализации функций алгебры логики формулами // Проблемы кибернетики. 1960. Вып. 3. С. 61—80.
11. **Яблонский С. В.** Об алгоритмических трудностях синтеза минимальных контактных схем // Проблемы кибернетики. 1959. № 2. С. 75—121.
12. **Поспелов Д. А.** Логические методы анализа и синтеза схем. Изд. 3-е, перераб. и доп. М.: Энергия, 1974. 368 с.
13. **Чебурахин И. Ф.** Математические модели для минимизации и автоматизации синтеза дискретных управляющих систем // Мехатроника, автоматизация, управление. 2012. № 4. С. 5—13.
14. **Чебурахин И. Ф.** О логическом управлении и обработке информации в дискретных технических системах на основе функциональных уравнений // Мехатроника, автоматизация, управление. 2013. № 11. С. 9—16.
15. **Shannon Claude E.** The Synthesis of Two-Terminal Switching Circuits // Bell System Technical Journal. 1949. N. 28. P. 59—98.
16. **Хоскин Р.** Как выбрать подходящую ПЛИС // Электронные компоненты. 2008. № 1. С. 38—42.
17. **Virtex-4 Family Overview** / Xilinx Inc. Cop. 2011. URL: http://www.xilinx.com/support/documentation/data_sheets/ds112.pdf.
18. **Virtex-7 FPGAs Data Sheet: DC and Switching Characteristics. DS183 (v1.0) March 1, 2011** / Xilinx Inc. Cop. 2011. URL: http://www.xilinx.com/support/documentation/data_sheets/ds183_Virtex_7_Data_Sheet.pdf.
19. **Шалагин С. В.** Экспериментальное исследование методики синтеза комбинационных схем на программируемых микросхемах класса FPGA // Микроэлектроника. 2004. Т. 33, № 1. С. 56—67.
20. **Шалагин С. В.** О представлении нелинейных полиномов над конечным полем распределенной вычислительной системой // Нелинейный мир. 2009. № 5. С. 376—379.

I. F. Cheburakhin, Professor, cybernetics@mati.ru,

MATI — Russian State Technological University, Moscow,

S. V. Shalagin, Associate professor, sshalagin@mail.ru,

Kazan National Research Technical University named after A. N. Tupolev — KAI (KNRTU KAI)

Conveying Calculation of Boolean Functions on the Basis of the Same Type IP Cores in Architecture FPGA/Virtex

The task of implementing arbitrary Boolean functions based on the Shannon's decomposition in the FPGA architecture Virtex-family are investigated and solved. The method of synthesis devices for computing Boolean functions using the same type of IP-cores is offered. Throught the use of pipelining in the data saving intermediate results nigh performance of these devices on the FPGA/Virtex is achieved.

Keywords: Boolean function, syntheses formula and circuit, decomposition, difficulty, minimization, functional of the equation

References

1. **Kalyaev I. A., Levin I. I.** Реконфигурируемые мультимиконвейерные вычислительные системы для решения потоковых задач обработки информации и управления. *Parallel'nye vychisleniya i zadachi upravleniya (PACO'10): plenarnye dokl. 5-y Mezhdunar. konf.* 26—28 okt. 2010. M.: IPU RAN, 2010. P. 23—37.
2. **Реконфигурируемые** мультимиконвейерные вычислительные структуры. I. A. Kalyaev, I. I. Levin, E. A. Semernikov i dr. 2-e izd. Rostov n/D: Izd-vo YuNTs RAN, 2009. 344 p.
3. **Zotov V. Yu.** Proektirovaniye vstraivaemykh mikroprotssessornykh sistem na osnove SAPR firmy Xilinx. M.: Goryachaya liniya — Telekom, 2006. 522 p.

4. **Zotov V. Yu.** Sredstva proektirovaniya vstraivaemykh mikroprotssessornykh sistem, realizuemykh na osnove PLIS firmy Xilinx. *Sovremennaya elektronika*. 2006. N. 7. P. 60—67.
5. **Shalagin S. V.** Realizatsiya ustroystv vychislitel'noy tekhniki na mnogoprotssessornykh sistemakh s programmiruемой arkhitekturoy. *Vestnik MarGTU*. 2011. N. 1 (11). P. 38—46.
6. **Shalagin S. V.** Modelirovaniye diskretnykh markovskikh protsessov na raspredeleyennoy vychislitel'noy sisteme s programmiruемой arkhitekturoy. *Dinamika nelineynykh diskretnykh elektrotekhnicheskikh i elektronnykh sistem: materialy 10-y Vseros. nauch.-tekhn. konf. 6—8 iyunya 2013*. Cheboksary: Izd-vo Chuvash. un-ta, 2013. P. 60—61.
7. **Zakharov V. M., Shalagin S. V.** Vychisleniye nelineynykh polinomial'nykh funktsiy na mnogoprotssessornoy sisteme s programmiruемой arkhitekturoy. *Informatsionnye tekhnologii*. 2012. N. 5. P. 6—11.

8. Zhuravlev Yu. I. Teoretiko-mnozhestvennye metody v algebre logiki. *Problemy kibernetiki*. 1962. N. 8 (1). P. 25–45.
9. Kudryavtsev V. B., Andreev A. E. O slozhnosti algoritmov. *Fundamental'naya i prikladnaya matematika*. 2009. N. 15 (3). P. 135–181.
10. Lupanov O. B. O slozhnosti realizatsii funktsiy algebry logiki formulami. *Problemy kibernetiki*. 1960. Iss. 3. P. 61–80.
11. Yablonskiy S. V. Ob algoritmicheskikh trudnostyakh sinteza minimal'nykh kontaktnykh skhem. *Problemy kibernetiki*. 1959. N. 2. P. 75–121.
12. Pospelov D. A. Logicheskie metody analiza i sinteza skhem. Izd. 3-e, pererab. i dop. M.: Energiya, 1974. 368 p.
13. Cheburakhin I. F. Matematicheskie modeli dlya minimizatsii i avtomatizatsii sinteza diskretnykh upravlyayushchikh system. *Mekhatronika, avtomatizatsiya, upravlenie*. 2012. N. 4. P. 5–13.
14. Cheburakhin I. F. O logicheskom upravlenii i obpabotke informatsii v diskretnykh tekhnicheskikh sistemakh na osnove funktsional'nykh uravneniy. *Mekhatronika, avtomatizatsiya, upravlenie*. 2013. N. 11 P. 9–16.
15. Shannon Claude E. The Synthesis of Two-Terminal Switching Circuits. *Bell System Technical Journal*. 1949. N. 28. P. 59–98.
16. Khoskin R. Kak vybrat' podkhodyashchuyu PLIS. *Elektronnye komponenty*. 2008. N. 1. P. 38–42.
17. Virtex-4 Family Overview. Xilinx Inc. Cop. 2011. URL: http://www.xilinx.com/support/documentation/data_sheets/ds112.pdf
18. Virtex-7 FPGAs Data Sheet: DC and Switching Characteristics. DS183 (v1.0) March 1, 2011. Xilinx Inc. Cop. 2011. URL: http://www.xilinx.com/support/documentation/data_sheets/ds183_Virtex_7_Data_Sheet.pdf.
19. Shalagin S. V. Eksperimental'noe issledovanie metodiki sinteza kombinatsionnykh skhem na programmirovemykh mikroskhemakh klassa FPGA. *Mikroelektronika*. 2004. V. 33, N. 1. P. 56–67.
20. Shalagin S. V. O predstavlenii nelineynykh polinomov nad konechnym polem raspredelennoy vychislitel'noy sistemoy. *Nelineyny mir*. 2009. N. 5. P. 376–379.

УДК 621.311.23:629.12

Ю. Н. Хижняков, д.р. техн. наук, проф., luda@at.pstu.ru,
А. А. Южаков, д.р. техн. наук, проф., зав. кафедрой, uz@at.pstu.ru,
 Пермский национальный исследовательский политехнический университет

Робастное управление недетерминированным объектом с применением многоуровневой обратной связи

Рассматривается робастное управление недетерминированным объектом с применением адаптивного нечеткого регулятора, адаптивной нечеткой обратной связи, реализованной на базе многоуровневого релейного элемента, управляемого текущим значением частотного показателя колебательности. Разработан адаптивный нечеткий регулятор и приведен алгоритм управления коэффициентом обратной связи в системе регулирования.

Ключевые слова: робастное управление, недетерминированный объект, адаптивный нечеткий регулятор, нечеткая обратная связь, частотный показатель колебательности, адаптивный нейрон, многоуровневый релейный элемент

Введение

Современные проблемы автоматизации сложных нелинейных (недетерминированных) объектов связаны с отсутствием их математического описания. Примерами таких объектов могут быть источники электроэнергии, работающие параллельно, авиационный двигатель, объекты с внутренним противоречием управления и т. д. Как правило, недетерминированные объекты разного класса требуют разного рода управления: группового, многорежимного, нечеткого, нечетко-нейронного и других видов управления. Так, например, групповое управление требуется для управления параллельной работой источников электроэнергии (синхронных генераторов и т. д.). Для управления ими разработаны разные методы: статических характеристик, мнимостатических характеристик, квазистатических характеристик и т. д. [1, 2].

Недетерминированный объект, например, авиационный двигатель, имеющий один вход (дозирующее устройство) и несколько выходов (частоты вращения свободной турбины, ротора компрессора высокого давления, вентилятора турбореактивного двигателя, температура газа в камере сгорания, давление воздуха за компрессором высокого давления

и т. д. при работе на Земле, взлете и крейсерской скорости), требует многорежимного управления [3].

Нечеткое управление недетерминированными объектами с нечетким математическим описанием снимает неопределенность в системах реального времени с помощью алгоритмов Заде, Мамдани, Ларсена, Тсукамото, Сугено-Такаги. Недостатком нечеткого управления является отсутствие адаптивности, которое устраняется методами стохастической аппроксимации: Уидроу-Хохфа, последовательного обучения и т. д. [4]. Дальнейшее развитие нечеткого управления с применением нейронных сетей вылилось в нейронечеткое управление (например, на основе нейросети *Anfis* и ее модификации с применением нечетких алгоритмов Сугено-Такаги, Ванга—Менделя и *TSK*) [4, 17].

Разработка управления недетерминированными объектами в системах реального времени связана с проблемой устойчивости, снижающей область практического применения. Однако с начала 1980-х годов и до настоящего времени разрабатываются методы анализа, где в разной степени используют динамические модели регулируемого объекта и/или регулятора. В работе [5] рассмотрены автоматные и реляционные нечеткие динамические