УДК 004.272.26

DOI: 10.21779/2542-0321-2023-38-4-25–31

С.В. Шалагин

Сложность вычисления нелинейных полиномиальных функций над полем Галуа в архитектуре ПЛИС/FPGA

Казанский национальный исследовательский технический университет им. А.Н. Туполева – КАИ; Россия, 420111, г. Казань, ул. Карла Маркса, 10; SShalagin@mail.ru

Аннотация. Получены оценки аппаратной сложности вычисления нелинейной полиномиальной функции, определённой над полем Галуа, в архитектуре ПЛИС класса FPGA. Оценки получены на основе однотипных IP-ядер двух типов, количество которых определено при использовании 2^k-арного дерева. IP-ядра первого типа определены по количеству листьев дерева, IP-ядра второго типа – по количеству остальных вершин данного дерева. Оценки сложности самих IP-ядер получены исходя из структуры реконфигурируемых однотипных логических элементов заданной ПЛИС/FPGA. Оценки сложности нелинейной полиномиальной функции вычислимы по количеству LUT-таблиц и блоков ввода-вывода для различных ПЛИС/FPGA – как для существующих, так и для перспективных.

Ключевые слова: нелинейная функция, поле Галуа, сложность, ПЛИС.

Введение

В настоящее время актуальны задачи высокоскоростной обработки массивов данных большого объема [1–6]. Одним из подходов к решению данного класса задач является их аппаратная реализация при использовании специализированных ЭВМ, архитектура которых отличается от фон-неймановской. В качестве аппаратной базы для решения таких задач применимы программируемые логические интегральные схемы класса FPGA (ПЛИС) [7–9]. Перспективным подходом к реализации высокоскоростной обработки и генерирования массивов данных на ПЛИС является применение систем нелинейных полиномиальных функций (НПФ) от заданного числа переменных, определённых над полем Галуа [10; 11]. Особенностью арифметики полей Галуа является то, что их элементы представимы двоичными векторами и операции над данными векторами допускают распределённые вычисления на уровне отдельных разрядов. Под распределёнными вычислениями будем понимать параллельные вычисления, выполняемые с сохранением промежуточных результатов. Архитектура ПЛИС включает в себя большое количество конфигурируемых логических блоков и программируемых межсоединений, что позволяет реализовать распределённые вычисления над двоичными векторами. Данное обстоятельство способствует применению ПЛИС для решения задач распределённого вычисления НПФ, определённых над полем Галуа.

В работе предложена модель вычисления НПФ над $GF(2^k)$ при использовании 2^k-арного дерева, где k – степень поля Галуа. Получены оценки сложности указанной модели на основе количества вершин и листьев данного дерева. Данные оценки позволяют сделать вывод о том, какие именно НПФ (системы НПФ) могут быть реализованы на заданной ПЛИС, как на существующей, так и на перспективной.

Основные понятия и определения

Нелинейную полиномиальную функцию от m переменных над полем Галуа вида $GF(2^k)$ обозначим как НПФ(m, 2^k) и представим в виде [12]:

$$f(x_1,...,x_m)=\sum_{i_1=0}^{2^k-1} f_{i_1}(x_2,...,x_m)\cdot x_1^{i_1}=f_0(x_2,...,x_m)+ \\ +f_1(x_2,...,x_m)\cdot x_1+...+f_{2^k-1}(x_2,...,x_m)\cdot x_1^{2^k-1}. \quad (1)$$

Коэффициенты $f_{i_1}(x_2,...,x_m)$ в формуле (1) представимы согласно формулам вида [10]:

$$f_{i_1...i_{d-1}}(x_d,...,x_m)=\sum_{i_d=0}^{2^k-1} f_{i_1...i_d}(x_{d+1},...,x_m)\cdot x_d^{i_d},\ d=\overline{2,m-1},$$

$$f_{i_1...i_{m-1}}(x_m)=\sum_{i_m=0}^{2^k-1} f_{i_1...i_m}\cdot x_m^{i_m},\ i_1,...,i_{m-1}=\overline{0,2^k-1}. \quad (2)$$

Согласно (2) НПФ(m, 2^k) представима при использовании 2^k коэффициентов, значения которых вычисляются на основе заданных НПФ($m-1$, 2^k). В свою очередь, каждая из 2^k НПФ($m-1$, 2^k) представима при использовании 2^k коэффициентов, значения которых получаются на основе заданных НПФ($m-2$, 2^k). В общем случае НПФ(m, 2^k) представима при использовании $(2^k)^d$ НПФ($m-d$, 2^k), $d=\overline{1,m-1}$, причём для $(2^k)^{m-1}$ НПФ(1, 2^k) значения $f_{i_1...i_m}$, $i_1,...,i_{m-1}=\overline{0,2^k-1}$ являются константами.

Процесс вычисления НПФ(m, 2^k) представим в виде 2^k-арного m-уровневого дерева T, на каждом уровне которого находится своя переменная. Определим для T количество вершин, в том числе вершин, являющихся листьями. Согласно [13] общее количество вершин T равно $\left(2^{km}-1\right)/\left(2^k-1\right)$, среди которых $2^{k(m-1)}$ – листья.

Формулы вида (1) и (2) описывают процесс вычисления НПФ(1, 2^k) на основе однотипных операций над элементами поля $GF\left(2^k\right)$. Указанные операции могут быть выполнены параллельно в архитектуре ПЛИС.

Оценки сложности НПФ(m, 2^k)

Введём в рассмотрение однотипные IP-ядра [14], которые позволяют вычислить функцию от (2^k+1) k-разрядных переменных над $GF\left(2^k\right)$ согласно (2). Данные IP-ядра разделим на две группы, которые обозначим как IP_1 и IP_2. Ядра из первой группы обозначим как ядра IP_1, а из второй – ядра IP_2. Ядра IP_1 реализуются как НПФ от одной переменной и 2^k констант (соответствуют листьям T). Ядра IP_2 реализованы согласно (2) как НПФ от (2^k+1) переменных. Ввиду того, что на 2^k входов ядер IP_1 подаются константы, они требуют для своей реализации существенно меньше вычислительных ресурсов, чем ядра IP_2. Ядра IP_1 описываются согласно (2) как $(2^k)^{m-1}$ НПФ(1, 2^k):

$$f_{i_1...i_{m-1}}(x_m)=\sum_{i_m=0}^{2^k-1} f_{i_1...i_m}\cdot x_m^{i_m},\ i_1,\ ...,\ i_{m-1}=\overline{0,2^k-1};$$

ядра IP_2 – как $\left(2^k\right)^{m-h}$ НПФ$\left(h,2^k\right)$, $h=\overline{2,m-1}$, вида:

$$f_{i_1...i_{d-1}}(x_d,...,x_m)=\sum_{i_d=0}^{2^k-1} f_{i_1...i_d}(x_{d+1},...,x_m)\cdot x_d^{i_d},\ d=\overline{2,m-1},$$

а также как одна НПФ(m, 2^k) согласно формуле (1) в виде

$$f\left(x_1,...,x_m\right)=\sum_{i_1=0}^{2^k-1} f_{i_1}\left(x_2,...,x_m\right)\cdot x_1^{i_1}\cdot$$

На основе вышеизложенного сформулирована

Лемма. Вычисление НПФ(m, 2^k) реализуемо при использовании однотипных IP-ядер, количество которых равно $\left(2^{km}-1\right)\Big/\left(2^k-1\right)$, из них принадлежащих к группе IP_1 – $2^{k(m-1)}$, и к группе IP_2 – $\left(2^{k(m-1)}-1\right)\Big/\left(2^k-1\right)\cdot$

Пусть ядра IP_1 и IP_2, выполняющие операции над $GF\left(2^k\right)$, реализуются в архитектуре ПЛИС, которая включает в себя LUT-таблицы и блоки ввода-вывода (БВВ). Для реализации ядер IP_1 требуется Q_1 LUT-таблиц, а для реализации ядер IP_2 – Q_2 LUT-таблиц. Количество входов для двоичных переменных НПФ(m, 2^k) равно $k\cdot m$, а выходов – k. Данные входы и выходы реализуются при использовании БВВ ПЛИС. Обозначим через $Q\left(m,2^k\right)$ и $I\left(m,2^k\right)$ количество LUT-таблиц и БВВ, требуемых для реализации НПФ(m, 2^k) на ПЛИС. При этом справедливы соотношения

$$Q\left(m,2^k\right)\le 2^{k(m-1)}\cdot Q_1+Q_2\cdot\left(2^{k(m-1)}-1\right)\Big/\left(2^k-1\right) \text{ и} \quad (3)$$

$$I\left(m,2^k\right)=k\left(m+1\right)\cdot \quad (4)$$

Имеет место следующая

Теорема. Оценка аппаратной сложности реализации НПФ(m, 2^k) в архитектуре ПЛИС по количеству LUT-таблиц удовлетворяет неравенству (3), а для БВВ – равенству (4).

Для реализации НПФ(m, 2^k) в одном корпусе ПЛИС требуется, чтобы оценки (3) и (4) не превышали заданных граничных значений [15]:

$$Q\left(m,2^k\right)\le k_Q\cdot Q_{\max},\quad I\left(m,2^k\right)\le k_I\cdot I_{\max}, \quad (5)$$

где $Q_{\max}$ и $I_{\max}$ – максимальное количество доступных пользователю LUT-таблиц и БВВ в одном корпусе заданной ПЛИС, а k_Q и k_I – допустимые коэффициенты использования ресурсов ПЛИС, установленные эмпирическим путём; обычно $k_Q, k_I\in\left[0{,}5,\,0{,}7\right]$ [16].

Из приведённой выше теоремы вытекает

Следствие. Если оценки сложности НПФ(m, 2^k) (3) и (4) удовлетворяют условиям вида (5) для заданной ПЛИС, то НПФ(m, 2^k) пригодна для реализации в одном корпусе данной ПЛИС.

Следствие обосновывает условия, согласно которым определена возможность реализации НПФ(m, 2^k) в одном корпусе ПЛИС с заданными характеристиками. Причём ПЛИС может быть не только существующей, но и перспективной, в настоящее время находящейся в процессе проектирования.

Анализ полученных результатов

Согласно введённой древовидной модели, вычисление НПФ(m, 2^k) производится параллельно. Сначала выполняется параллельное вычисление значений $\left(2^k\right)^{m-1}$ НПФ(1, 2^k) при использовании ядер IP_1, на входы каждого из которых подаётся переменная x_m и 2^k констант. Затем выполняется параллельное вычисление при использо-

вании ядер IP_2 и при заданном h вычислении $(2^k)^{m-h}$, НПФ(h, 2^k) на входы которых подаются $\left(2^k+1\right)$ переменных: x_{m-h+1} и 2^k значений, полученных на основе заданных по формуле (2) $\text{НПФ}\left((h-1), 2^k\right)$, $h=\overline{2,m}$. Архитектура ПЛИС позволяет реализовать данные вычисления в одном кристалле и параллельно при выполнении условий, заданных согласно (5).

Однотипные ядра IP_1 и IP_2 адекватно вписываются в архитектуру ПЛИС при заданных значениях степени поля Галуа $k = 2$ и 3, что будет показано далее. Рассмотрим примеры оценок сложности для частных случаев НПФ(m, 2^k) при заданном m. Варьирование величины k ограничено количеством входов LUT-таблиц для ПЛИС заданных семейств, которое должно быть больше или равно $2k$.

Оценки Q_1 и Q_2 для $\text{НПФ}\left(m, 2^2\right)$ будут вычислены по количеству LUT-таблиц от четырех переменных (для ПЛИС семейств Virtex-4, Quartus II и т. п.) и составляют 2 и 8, соответственно [4; 5]. Для $k=2$ формулы (3) и (4) имеют вид:

$$Q\left(m, 2^2\right) \le 2\cdot 4^{(m-1)} + 8\cdot\left(4^{(m-1)}-1\right)\Big/3 \text{ и} \tag{6}$$

$$I\left(m, 2^2\right) = 2\left(m+1\right) \tag{7}$$

Для $\text{НПФ}\left(m, 2^3\right)$ оценки $Q_1 = 3$ и $Q_2 = 24$, соответственно, и вычислены по количеству LUT-таблиц от шести переменных (для ПЛИС семейств Virtex-5, Virtex-6, Virtex-7 и т. п. [3]. Формулы (3) и (4) для $k=3$ имеют вид:

$$Q\left(m, 2^3\right) \le 3\cdot 8^{(m-1)} + 24\cdot\left(8^{(m-1)}-1\right)\Big/7 \text{ и} \tag{8}$$

$$I\left(m, 2^3\right) = 3\left(m+1\right). \tag{9}$$

Значения, вычисленные согласно (7) и (9), растут линейно при возрастании количества переменных m НПФ(m, 2^k) при заданном k, $k = 2, 3$.

Значения, полученные по формулам (6) и (8), растут экспоненциально при возрастании m. График роста $Q\left(m, 2^2\right)$ в зависимости от m приведён на рис. 1, а соответствующий график для $Q\left(m, 2^3\right)$ – на рис. 2. На рис. 1 и 2 по оси абсцисс отложены значения m, ось ординат имеет логарифмическую шкалу.

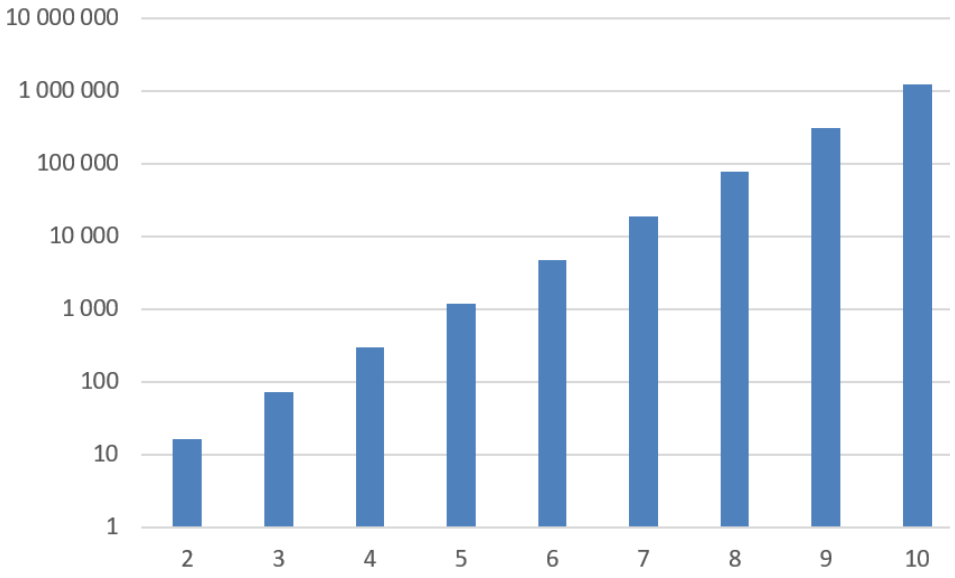

Рис. 1. Зависимость $Q\left(m, 2^2\right)$ от количества переменных m

Сопоставим между собой значения на рис. 1 для m = 3, 6 и 9, со значениями на рис. 2 для m = 2, 4 и 6, соответственно. Количество двоичных разрядов, поступающих на вход $\text{НПФ}\left(m, 2^k\right)$, k = 2, 3, при указанных значениях m равно 6, 12 и 18. С выходов $\text{НПФ}\left(m, 2^2\right)$ снимаются два, а с выходов $\text{НПФ}\left(m, 2^3\right)$ – три бинарных значения. Количество LUT-таблиц от шести переменных, применяемых для реализации $\text{НПФ}\left(m, 2^3\right)$, будет примерно в 1,45–1,5 раза меньше, чем LUT-таблиц от четырёх переменных, применяемых для реализации $\text{НПФ}\left(m, 2^2\right)$.

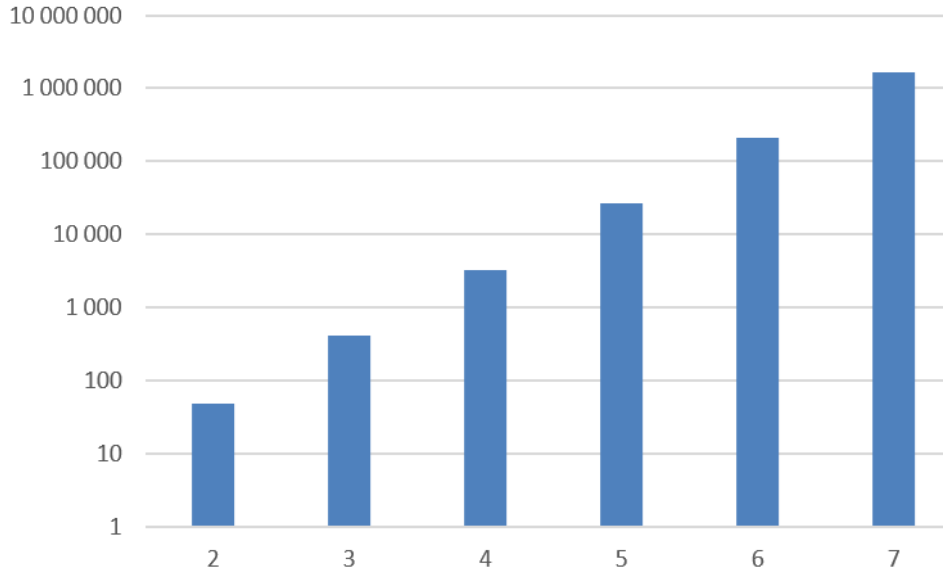

Рис. 2. Зависимость $Q\left(m, 2^3\right)$ от количества переменных m

При увеличении количества переменных количество задействованных LUT-таблиц будет превышать величину $Q_{\max}$, заданную согласно (5) для существующих ПЛИС.

Заключение

Рассмотрена задача оценки сложности вычисления нелинейной полиномиальной функции, определённой над полем Галуа, в архитектуре ПЛИС, в зависимости от количества переменных НПФ(m, 2^k), k = 2, 3. Размерность поля зависит от заданного количества переменных, подаваемых на входы LUT-таблиц внутри ПЛИС. Вычисление НПФ согласно древовидной модели реализуемо на основе однотипных IP-ядер, что делает данный процесс соответствующим архитектуре ПЛИС класса FPGA.

Приведённые вычисления показали, как растёт количество LUT-таблиц и блоков ввода-вывода ПЛИС в зависимости от количества переменных НПФ для поля Галуа степени k = 2, 3, над которым определена нелинейная полиномиальная функция.

Литература

1. *Кузнецов А.А.*, *Кузнецов П.А.*, *Кузьменко Э.Ю.* Правовые основы цифровой экономики в Российской Федерации // Актуальные проблемы государства и права. 2023. Т. 7, № 2 (26). – С. 190–197. – DOI 10.20310/2587-9340-2023-7-2-190–197.

2. Конструктивное моделирование процессов синтеза / *В.А. Райхлин*, *И.С. Вершинин*, *Р.К. Классен* [и др.]; Казанский национальный исследовательский технический университет им. А.Н. Туполева-КАИ. – Казань: Изд-во «Фэн» Академии наук Республики Татарстан, 2020. – 248 с.

3. *Макаров А.М.*, *Ермаков А.С.* Основы теории интегрального преобразования Меллина и разработка на его базе цифровой модели Эйлерова интеграла первого рода // Вестник Дагестанского государственного университета. Сер. 1: Естественные науки. 2021. Т. 36, вып. 3. – С. 79–88. – DOI 10.21779/2542-0321-2021-36-3-79–88.

4. *Магомедов А.М.*, *Лавренченко С.А.* Вычислительные средства C# для решения задачи перечисления разбиений прямоугольника // Вестник Дагестанского государственного университета. Сер. 1: Естественные науки. 2020. Т. 35, вып. 4. – С. 13–26. – DOI 10.21779/2542-0321-2020-35-4-13–26.

5. *Магомедов А.М.*, *Лавренченко С.А.*, *Ибрагимова З.И.* Алгоритм автоматизации создания тестов // Вестник Дагестанского государственного университета. Сер. 1: Естественные науки. 2019. Т. 34, вып. 2. – С. 63–71. – DOI 10.21779/2542-0321-2019-34-2-63–71.

6. *Зайченко П.А.* Анализ радиационных характеристик тепловыделяющей сборки исследовательского реактора по результатам моделирования облучения в активной зоне // Вестник Дагестанского государственного университета. Сер. 1: Естественные науки. 2020. Т. 35, вып. 3. – С. 12–16. – DOI 10.21779/2542-0321-2020-35-3-12–16.

7. FPGA Leadership across Multiple Process Nodes/ Advanced Micro Devices, Inc. 2023. – Режим доступа: https://www.xilinx.com/products/silicon-devices/fpga.html

8. ПЛИС Altera. ООО «Электроника-РА». 2014–2021. – Режим доступа: https://elra.ru/mikroskhemy/plis-cpld/plis-altera/

9. Новые российские ПЛИС // Современная электроника. Новости. 2019. – Режим доступа: https://www.soel.ru/novosti/2019/novye_rossiyskie_plis/ (дата обращения: 20.03.2023).

10. *Захаров В.М.*, *Шалагин С.В.*, *Эминов Б.Ф.* Автоматные марковские модели над конечным полем. – Казань: Спец. фонд управления целевым капиталом для развития Казанского национального исследовательского технического университета им. А.Н. Туполева. – КАИ, 2022. – 328 с.

11. *Shalagin S.V.* Computing a group of polynomials over a Galois field in FPGA architecture // Mathematics. 2021. Vol. 9, no. 24. – DOI 10.3390/math9243251.

12. *Лидл Р.*, *Нидеррайтер Г.* Конечные поля: в 2 т. – М.: Мир, 1988.

13. *Ахо А.*, *Хопкрофт Дж.*, *Ульман Дж.* Построение и анализ вычислительных алгоритмов / пер. с анг. А.О. Слисенко; под ред. Ю.В. Матиясевича. – М.: Мир, 1979. – 536 с.

14. *Зотов В.Ю.* Проектирование встраиваемых микропроцессорных систем на основе САПР фирмы Xilinx. – М.: Горячая линия – Телеком, 2006. – 522 с.

15. *Шалагин С.В.* Сложность вычисления нелинейных полиномиальных функций над полем $GF(2^2)$ на ПЛИС/FPGA // Поиск эффективных решений в процессе создания и реализации научных разработок в российской авиационной и ракетно-космической промышленности: материалы Межд. научно-практич. конф. (Казань, 5–8 августа 2014 г.). – Казань: Изд-во Казанского государственного технического университета, 2014. Т. II. – С. 661–664.

16. *Пономарев В.И.*, *Шабалин Л.А.* Проектирование реконфигурируемых устройств обработки цифровых потоков данных // Информационные технологии. 1996. № 5. – С. 24–28.

Поступила в редакцию 21 сентября 2023 г.
Принята 15 октября 2023 г.

UDC 004.272.26

DOI: 10.21779/2542-0321-2023-38-4-25–31

The Complexity of Computing Nonlinear Polynomial Functions Over the Galois Field in FPGA-Architecture

S.V. Shalagin

Kazan National Research Technical University named after A.N. Tupolev – KAI; Russia, 420111, Kazan, Karl Marks st., 10; SShalagin@mail.ru

Abstract. The estimates of the hardware complexity of computing a nonlinear polynomial function defined over a Galois field in FPGA-architecture are obtained. The estimates are based on the same model of IP-cores of two types, the number of which is determined using a 2k-ary tree. IP-cores of the first type are determined by the number of leaves of the tree, IP-cores of the second type are determined by the number of other vertices of this tree. The estimates of the complexity of the IP-cores themselves are based on the structure of reconfigurable similar logic elements of a given FPGA. The estimates are calculated by the number of LUT-tables and I/O blocks for various FPGAs, both for existing and prospective ones.

Keywords: nonlinear function, Galois field, complexity, FPGA.

Received 21 September, 2023
Accepted 15 October, 2023