

Особенности FPGA реализации умножителя кватернионов на базе схемы 4D CORDIC

Петровский Н.А.

Кафедра электронных вычислительных средств
Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь
e-mail: nick@petrovsky.eu

Аннотация—Представлен модифицированный 4D CORDIC алгоритм для умножения кватернионов с прореженными итерациями, которые представляют собой умножение на вырожденные кватернионы состоящие только из одной ненулевой мнимой части. По сравнению с известным CORDIC алгоритмом умножения кватернионов, предложенным Hisiao и Delosme, данный подход приводит к менее затратной схеме, которая состоит из мультиплексора и четырёх двухвходовых сумматоров-сдвигателей вместо 4-входовых сумматоров в прототипе. Модернизированный 4D CORDIC алгоритм имеет лучшую сходимость и меньший диапазон масштабирующего коэффициента. В докладе исследуется особенности FPGA реализации умножителя кватернионов.

Ключевые слова: умножитель кватернионов, CORDIC алгоритм, FPGA реализация.

I. ВВЕДЕНИЕ

В настоящее время кватернионы рассматриваются как новая парадигма в цифровой обработке сигналов – непосредственная обработка в многомерной области. Благодаря этому, многомерный сигнал может представляться как одиночная переменная, что обеспечивает обобщение и гибкость в моделировании трёх- и четырёхмерных сигналов. Кватернионы находят широкое применение в таких прикладных областях, как 3D графика, навигация, робототехника, обработка изображений, банки фильтров, многополосные вейвлет преобразования и т.д [1-3]. Не так много работ выполнено по поиску эффективных аппаратных решений операции умножения кватернионов [4-6].

II. УМНОЖЕНИЕ КВАТЕРНИОНОВ

Кватернион – гиперкомплексное число размерности 4, которое описывается выражением:

$$q = q_1 + q_2i + q_3j + q_4k, \quad q_1, q_2, q_3, q_4 \in \mathbf{R}, \quad (1)$$

где соответствующие мнимые единицы связаны следующим образом:

$$\begin{aligned} i^2 = j^2 = k^2 = ijk = -1, \\ ij = -ji = k, \quad jk = -kj = i, \quad ki = -ik = j. \end{aligned} \quad (2)$$

Известно [3], что операция умножения кватернионов является некоммутативной. Поэтому для выполнения умножения формируются специальные матрицы левого и правого операндов:

$$qx \Leftrightarrow \underbrace{\begin{bmatrix} q_1 & -q_2 & -q_3 & -q_4 \\ q_2 & q_1 & -q_4 & q_3 \\ q_3 & q_4 & q_1 & -q_2 \\ q_4 & -q_3 & q_2 & q_1 \end{bmatrix}}_{\mathbf{M}^+(q)} \underbrace{\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix}}_x, \quad (3)$$

но

$$xq \Leftrightarrow \underbrace{\begin{bmatrix} q_1 & -q_2 & -q_3 & -q_4 \\ q_2 & q_1 & q_4 & -q_3 \\ q_3 & -q_4 & q_1 & q_2 \\ q_4 & q_3 & -q_2 & q_1 \end{bmatrix}}_{\mathbf{M}^-(q)} \underbrace{\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix}}_x. \quad (4)$$

Непосредственное вычисление произведения кватернионов по (3) и (4) требует 12 сложений и 16 умножений действительных чисел.

III. 4D CORDIC АЛГОРИТМ УМНОЖЕНИЯ КВАТЕРНИОНОВ

A. Постановка задачи

Положим, кватернион множитель – константа. Вариант умножения кватернионов на основе алгоритма 4D CORDIC предложен в работе [4], где используется следующая аппроксимация матрицы умножения кватернионов:

$$\begin{aligned} \mathbf{M}^\pm \approx \prod_{n=0}^{N-1} \frac{1}{\left| \zeta \left(\tau^{(n)}, \sigma_1^{(n)}, \sigma_2^{(n)}, \sigma_3^{(n)} \right) \right|} \times \\ \times \mathbf{M}^\pm \left(\zeta \left(\tau^{(n)}, \sigma_1^{(n)}, \sigma_2^{(n)}, \sigma_3^{(n)} \right) \right). \end{aligned} \quad (5)$$

Умножение на гиперкомплексное число представляет n -ую итерацию 4D CORDIC алгоритма:

$$\begin{aligned} \zeta \left(\tau^{(n)}, \sigma_1^{(n)}, \sigma_2^{(n)}, \sigma_3^{(n)} \right) = \\ 1 + \sigma_1^{(n)} 2^{\tau^{(n)}} i + \sigma_2^{(n)} 2^{\tau^{(n)}} j + \sigma_3^{(n)} 2^{\tau^{(n)}} k. \end{aligned} \quad (6)$$

При этом, $\tau^{(n)}$ – величина сдвига, $\sigma_{1...3}^{(n)}$ – знак, n – номер итерации. Учитывая (5), возможна следующая факторизация матрицы умножения:

$$M^{\pm} \left(\zeta \left(\tau^{(n)}, \sigma_1^{(n)}, \sigma_2^{(n)}, \sigma_3^{(n)} \right) \right) = \begin{bmatrix} 1 & -\sigma_1^{(n)} 2^{\tau^{(n)}} & -\sigma_2^{(n)} 2^{\tau^{(n)}} & -\sigma_3^{(n)} 2^{\tau^{(n)}} \\ \sigma_1^{(n)} 2^{\tau^{(n)}} & 1 & \mp \sigma_3^{(n)} 2^{\tau^{(n)}} & \pm \sigma_2^{(n)} 2^{\tau^{(n)}} \\ \sigma_2^{(n)} 2^{\tau^{(n)}} & \pm \sigma_3^{(n)} 2^{\tau^{(n)}} & 1 & \mp \sigma_1^{(n)} 2^{\tau^{(n)}} \\ \sigma_3^{(n)} 2^{\tau^{(n)}} & \mp \sigma_2^{(n)} 2^{\tau^{(n)}} & \pm \sigma_1^{(n)} 2^{\tau^{(n)}} & 1 \end{bmatrix}. \quad (7)$$

Операция масштабирования может выполняться отдельно от итерации микрооборотов, и аппроксимируется последовательностью сдвигов и сложений:

$$S_{tot} = \prod_{n=0}^{N-1} \frac{1}{\sqrt{1 + 3 \cdot 2^{-2\tau^{(n)}}}} \approx \prod_{n=0}^{N-1} 1 - 2^{-s^{(m)}}. \quad (8)$$

В. FPGA реализация умножителя кватернионов

Известная схема [4] (рис.2.а) содержит четыре 4-входовых сумматора, существенно увеличивающих аппаратные затраты, к тому же данный сумматор не используется полностью для операции масштабирования. Предложенная схема (рис.2.б), согласно (5-8) состоит из 4-х 2-входовых сумматоров и схемы мультиплексирования (рис. 1). Комбинацию управляющих сигналов $(\sigma^{(n)}, \tau^{(n)}, \beta_1, \beta_2, s^{(n)})$ для входных данных n -ной итерации можно представить в виде микроинструкций управляющей микропрограммы.

Данная реализация алгоритма и его HDL описание имеет ряд преимуществ: позволяет задавать произвольное число итераций для заданной точности, а так же увеличивать количество исполнительных CORDIC модулей за счёт конвейеризации, разделяя микропрограмму на кратное число частей. Аппаратные затраты приводятся в таблице 1.

Табл. 1. ИСПОЛЬЗОВАННЫЕ РЕСУРСЫ FPGA VIRTEX-4 XC4V5X35

Тип оборудования	Количество	Использовано
Flip Flops	201	1%
4 LUT	554	1%

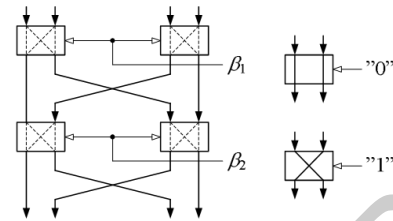


Рис. 1. Схема мультиплексирования «Interconnection network»

С. Заключение

Представленный умножитель кватернионов существенно уменьшает затраты оборудования по сравнению с прототипом [4], а также позволяет организовывать конвейерные схемы вычислений.

- [1] T. Bulow and G.Sommer, "Hypercomplex signals — a novel extension of the analytic signal to the multidimensional case," IEEE Trans. Signal Process., vol.49, no.11, pp.2844–2852, Nov.2001.
- [2] W. L. Chan, H. Choi, and R. G. Baraniuk, "Directional hypercomplex wavelets for multidimensional signal analysis and processing," in Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP), vol. 3, Montreal, Canada, 17–21 May 2004, pp.III–996–III–999.
- [3] M. Parfeniuk and A. Petrovsky, "Inherently lossless structures for eight- and six-channel linear-phase paraunitary filterbanks based on quaternion multipliers," Signal Process., vol.90, pp.1755–1767, 2010.
- [4] S. F. Hsiao and J. M. Delosme, "Parallel singular value decomposition of complex matrices using multidimensional CORDIC algorithms," IEEE Trans. Signal Process., vol.44, no.3, pp.685–697, Mar.1996.
- [5] S. -F. Hsiao, C. -Y. Lau, and J. -M. Delosme, "Redundant constant-factor implementation of multi-dimensional CORDIC and its application to complex SVD," J. VLSI Signal Process., vol.25, no.2, pp.155–166, Jun.2000.
- [6] N.Petrovsky, M. Parfeniuk, "The CORDIC-inside-Lifting Architecture for Constant-Coefficient Hardware Quaternion Multipliers", ICSES, p.6, Sep 2012.

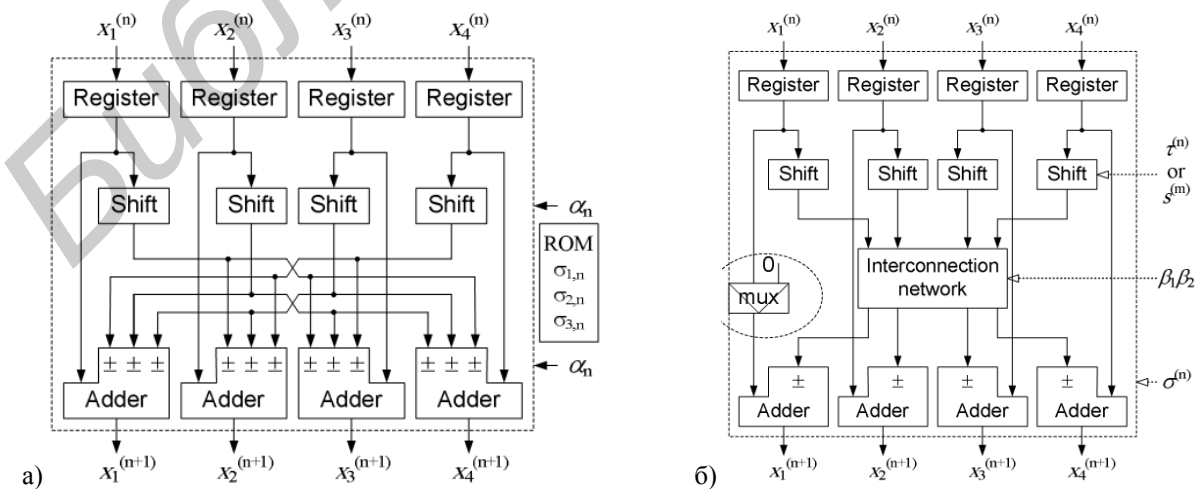


Рис. 2. а) функциональная схема 4D CORDIC; б) функциональная схема оптимизированного 4D CORDIC модуля