

Следящий потоковый вычислительный преобразователь для интеллектуальных измерительных систем

Н. М. Сафьянников¹, О. И. Буренева²

Санкт-Петербургский государственный электротехнический
университет «ЛЭТИ» им. В.И. Ульянова (Ленина)

¹nmsafyannikov@etu.ru, ²oibureneva@etu.ru

Аннотация. Следящий потоковый вычислительный преобразователь по сравнению с известными устройствами позволяет повысить помехоустойчивость измерительной системы в информационном канале преобразования и передачи данных. Это достигается рекурсивной организацией вычислительного процесса при которой наряду с выполнением операции умножения реализуется оператор сглаживания. Дополнительным преимуществом является простота взаимодействия с источником и приемником информации и возможность использования предлагаемого устройства в составе измерительно-вычислительных систем с распределенной обработкой информации, а также высокое быстродействие при обработке малых приращений аргументов.

Отказоустойчивость; потоковые преобразования; вычислительный преобразователь; рекурсивные вычисления; измерительные системы; сглаживание результата

I. ВВЕДЕНИЕ

Большинство чувствительных элементов, используемых в измерительных системах, изначально формируют аналоговый сигнал, который подается непосредственно на выход датчика. Благодаря возможностям интегральных технологий, активно развиваемых в настоящее время, промежуточный преобразователь может быть внедрен в структуру датчика, для получения готового к использованию обработанного цифрового сигнала.

Обработка предполагает как преобразование формы представления информации, так и выполнение линеаризации сигнала датчика с использованием различных функций. При этом предпочтение отдается датчикам, обеспечивающим непрерывный процесс измерений и преобразований.

Интеллектуальные измерительные системы часто предполагают первичную обработку измерительной информации для принятия решений о результатах измерений с целью реализации управления. При этом в большинстве случаев требуются плавные изменения сигналов управления, например, при воздействии на исполнительные механизмы роботов, манипуляторов, бионических протезов и т.д. Для сглаживания могут использоваться различные

функции: логарифмическая, экспоненциальная, тригонометрические.

Таким образом, становится актуальной разработка следящих потоковых вычислительных преобразователей для интеллектуальных измерительных систем.

Следящие вычислительные преобразователи легко реализуются в классе бит-потоковых устройств, обрабатывающих данные, представленные потоками единичных бит [1, 2].

Это направление развивается и нами применительно к разработке отказоустойчивых бит-потоковых преобразователей [3, 4]. В основе созданных устройств лежат нетрадиционные структурно-функциональные решения, которые позволяют получить более эффективные результаты за счет оригинальных подходов к организации вычислений.

Разработанные принципы обработки бит-потоковой информации могут быть распространены на преобразователи, выполняющие вычисления с информацией, представленной в кодовой форме, или код-потоковые.

II. ОБОБЩЕННАЯ СТРУКТУРНАЯ СХЕМА ПРЕОБРАЗОВАТЕЛЯ

Работа следящего код-потокового вычислительного преобразователя может быть основана на тех же принципах организации вычислительного процесса, что и бит-потоковые функциональные преобразователи.

Бит-потоковый преобразователь (рис. 1, а) содержит блоки формирования сигналов цепи прямой и обратной связи $(D, \Theta)/f$ и интегратор, в качестве которого использован реверсивный счетчик СТ. На входы устройства поступают воздействия N, Θ, f , представленные соответственно в кодовой, и потоковой (широко-импульсной и частотной) форме. Блоки формирования сигналов прямой и обратной связи осуществляют операции над сигналами различной формы представления, формируя на выходе частотно-импульсный сигнал. Выход бит-потокового устройства может иметь как кодовое, так и бит-потоковое представление.

На основе аналогий математических функций, воспроизводимых отдельными решающими узлами или их сово-

Работа выполнена в рамках гос. задания Минобрнауки России
№8.2080.2017/4.6

купностями, в бит-поточковых устройствах можно перейти к структуре код-поточкового преобразователя.

Так, например, функции интегратора, выполняемые реверсивным счетчиком, могут быть реализованы совокупностью вычитателя и накапливающего сумматора. Они будут выполнять те же функции: суммирование, вычитание и интегрирование полученной разности с выдачей результата в виде двоичного кода. Для выполнения базовых математических операций могут использоваться комбинационные умножители, сумматоры, вычитатели и другие типовые элементы.

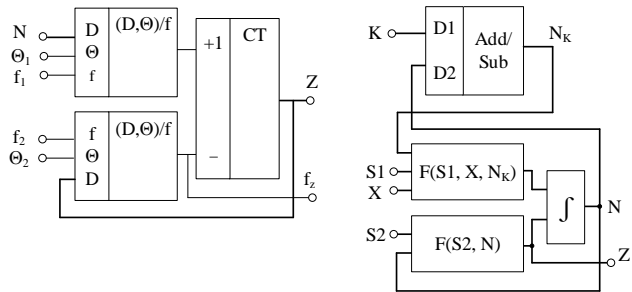


Рис. 1. Структурные схемы потоковых вычислительных преобразователей: а – бит-поточкового, б – код-поточкового,

Условием установления результата преобразования как в бит-поточковых устройствах, так и в код-поточковых преобразователях является равенство приращений, поступающих на интегратор.

Код-поточковый преобразователь (рис. 1, б) содержит сумматор-вычитатель Add/Sub, блоки формирования сигналов цепи прямой $F(S1, X, N_k)$ и обратной $F(S2, N)$ связи, интегратор \int . На входы устройства поступают воздействия масштаба K , наборы опорных величин $S1, S2$ и аргумент X , представленные в кодовой форме. Сумматор-вычитатель и блоки формирования сигналов прямой и обратной связи являются комбинационными элементами и осуществляют операции в кодовой форме. Элементы памяти используются только в интеграторе.

Отличительной особенностью бит-поточковых устройств является необходимость промежуточных преобразований кодов в потоковые формы представления. Код-поточковые устройства работают только с кодовыми представлениями данных. При этом благодаря общности структуры и подходам к обработке данных эти устройства можно отнести к одному классу – классу замкнутых итерационных структур с автоматической компенсацией, что обеспечивает повышенную отказоустойчивость вычислений.

III. РЕАЛИЗАЦИЯ ЛОГАРИФИЧЕСКОГО ПРЕОБРАЗОВАТЕЛЯ

В качестве примера реализации код-поточкового преобразователя рассмотрим устройство для вычисления логарифма, широко востребованное в различных вычислительных приложениях [5, 6].

В основу работы рассмотренного типа устройства положен принцип итерационного процесса формирования

реализуемой функции, с помощью дробно-рациональной аппроксимации на основе автоматической, компенсации задаваемых и вырабатываемых в процессе работы данных, приводящей к установлению результата при равенстве сигналов прямой и обратной связи.

Преобразователь (рис. 2) работает следующий образом. Пусть в начальный момент времени разряды накапливающего сумматора 10 обнулены. На входы устройства поступают: код аргумента X , коды опорных величин N_1, N_2, N_3, N_4 , соответствующие следующим значениям: 0,25062; 1,18889; 1; 1,7053.

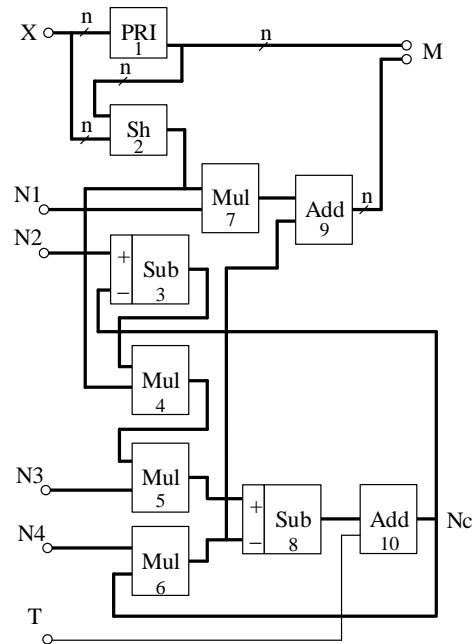


Рис. 2. Схема логарифмического преобразователя

Вычисление логарифма по основанию 2 основано на соотношениях:

$$\log_2 X = N - 1 + \log_2(t + 1),$$

где

$$t = \frac{X - 2^{N-1}}{2^{N-1}}, \quad 0 \leq t < 1,$$

N – номер старшего единичного разряда кода X .

Блок приоритетного шифратора PRI определяет номер N старшего единичного разряда входного кода X , формируя тем самым целую часть результирующего кода M . Результат приоритетной шифрации также поступает на комбинационный блок сдвига. Этот блок выполняет сдвиг входного кода X на $N-1$ разрядов вправо, формируя значение t . Комбинационный умножитель Mul 7 вычисляет значение $N_1 t$.

Сумматор Sub 3 формирует значения $N_2 - N_c$, а умножители Mul 4 и 5 – значения $(N_2 - N_c)t$ и $(N_2 - N_c)N_3 t$ соответственно. Блок цифровой обратной связи реализован

на базе умножителя Mul 6, который формирует код, соответствующий значению $N_c N_4$. Сумматор Sub 8 обеспечивает поступление на вход накапливающего сумматора 10 приращения после первой итерации, равного:

$$\Delta_1 = (N_2 - N_c)N_3t - N_c N_4 = N_2 N_3 t.$$

В результате второй итерации в сумматор передается второе приращение, которое будет просуммировано с его содержанием:

$$\Delta_2 = N_2 N_3 t (1 - N_3 t - N_4).$$

Последующие приращения описываются выражениями:

$$\Delta_3 = N_2 N_3 t (1 - N_3 t - N_4)^2,$$

$$\Delta_4 = N_2 N_3 t (1 - N_3 t - N_4)^3,$$

.....

$$\Delta_n = N_2 N_3 t (1 - N_3 t - N_4)^{n-1}.$$

Фиксация данных в накапливающем сумматоре осуществляется под управлением тактового сигнала T. Его частота определяется временем срабатывания комбинационных цепей, количество тактовых импульсов соответствует количеству просуммированных приращений, что определяет точность выполнения преобразования.

Условием окончания итерационного процесса суммирования приращений является равенство 0 приращения накапливающего сумматора, то есть

$$(N_2 - N_c)N_3t - N_c N_4 = 0.$$

Откуда код числа, сформированный накапливающим сумматором Add 10 при достижении установившегося режима, будет определяться как

$$N_c = \frac{N_2 N_3 t}{N_3 t + N_4}.$$

На второй вход сумматора Add 9 поступит код

$$N_c = \frac{N_2 N_3 N_4 t}{N_3 t + N_4},$$

а на первом входе, с учетом функции умножителя Mul 7, будет удерживаться код $N_1 t$.

Таким образом, на выходе сумматора Add 9 будет сформирована дробная часть результата преобразования, описываемая выражением

$$M = \frac{N_1 N_3 t^2 + (N_1 N_4 + N_2 N_3 N_4) t}{N_3 t + N_4}.$$

При заданных значениях $N_1 - N_4$ последнее выражение может быть преобразовано к виду

$$M = \frac{0,25062t^2 + 2,4548t}{t + 1,7053}.$$

Это выражение аппроксимирует функцию $\log_2(t+1)$ с методической погрешностью, не превышающей $1,38 \cdot 10^{-4}$.

Математическое моделирование работы преобразователя выполнялось с использованием табличного процессора Excel и состояло в вычислении приращений и их интегрировании. Процесс достижения результата вычисления функции $\log_2 X$ для значения $X=264$ показан в таблице 1.

ТАБЛИЦА 1 РЕЗУЛЬТАТ МОДЕЛИРОВАНИЯ ПРОЦЕССА ИТЕРАЦИОННОГО ВЫЧИСЛЕНИЯ ФУНКЦИИ ЛОГАРИФМА

Номер итерации	Приращение	N_c	M $\log_2(t+1)$	$\log_2 X$
1	0,0372	0,0000	0,0712	8,0712
2	-0,0274	0,0372	0,0245	8,0245
3	0,0202	0,0098	0,0589	8,0589
4	-0,0148	0,0299	0,0336	8,0336
5	0,0109	0,0151	0,0522	8,0522
6	-0,0081	0,0260	0,0385	8,0385
7	0,0059	0,0180	0,0486	8,0486
8	-0,0044	0,0239	0,0412	8,0412
9	0,0032	0,0195	0,0466	8,0466
10	-0,0024	0,0228	0,0426	8,0426
11	0,0017	0,0204	0,0456	8,0456
12	-0,0013	0,0221	0,0434	8,0434
13	0,0009	0,0208	0,0450	8,0450
14	-0,0007	0,0218	0,0438	8,0438
15	0,0005	0,0211	0,0447	8,0447
16	-0,0004	0,0216	0,0440	8,0440
17	0,0003	0,0212	0,0445	8,0445
18	-0,0002	0,0215	0,0442	8,0442
19	0,0002	0,0213	0,0444	8,0444
20	-0,0001	0,0215	0,0442	8,0442

Анализ полученных данных показывает, что устройство достигает установившегося режима на 18 итерации. При этом уже на первом такте получен приближенный результат, который уточняется на последующих тактах.

Благодаря наличию обратной связи появление одиночных сбоев не приводит к потере результата, а лишь замедляет его повторное достижение.

IV. РЕАЛИЗАЦИЯ ПРЕОБРАЗОВАТЕЛЯ

Предложенный преобразователь был описан на языке проектирования аппаратуры VerilogHDL. С целью упрощения процесса синтеза использован структурный подход, для реализации которого подготовлены следующие параметризованные модули: mult – комбинационный умножитель, add, sub, add_sub – комбинационные сумматор, вычитатель и сумматор/вычитатель. Накапливающий сумматор реализован парой сумматор/регистр.

Языковое описание использовано для синтеза устройства с использованием САПР Quartus II, полученный RTL-вид фрагмента устройства, обеспечивающего вычисление дробной части результата преобразования, приведен на рис. 3.

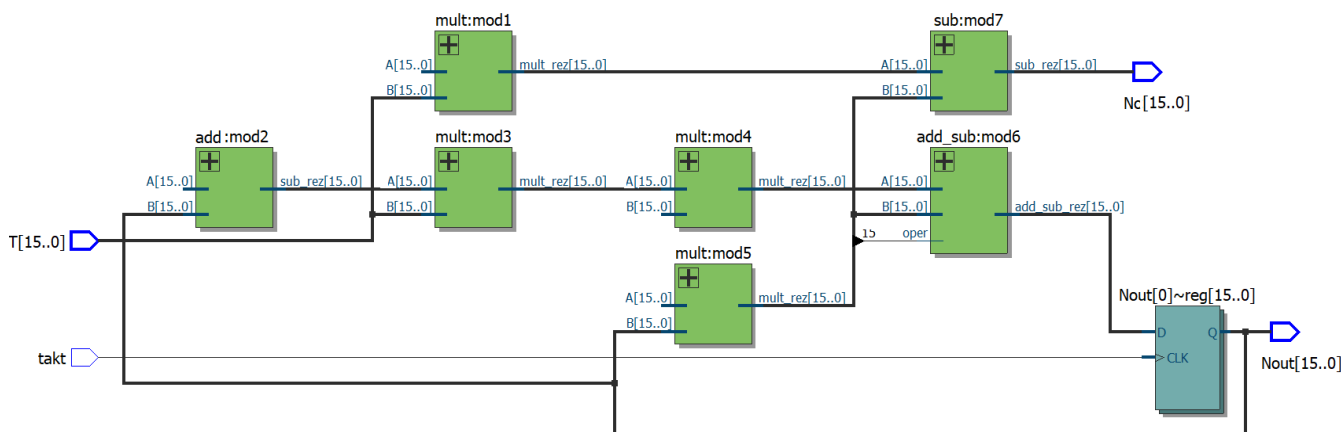


Рис. 3. RTL описание фрагмента логарифмического преобразователя

Коэффициенты аппроксимирующего выражения в рассматриваемом языковом описании устройства задаются с помощью параметров. Перенос коэффициентов на внешние входы позволит расширить функциональность преобразователя: оперативно меняя значения кодов можно изменять вид аппроксимирующего выражения и использовать преобразователь для вычисления других функций.

Моделирование устройства выполнялось с использованием системы QuestaSim. В ходе модельных экспериментов подтверждена работоспособность и метрологические характеристики преобразователя.

V. ЗАКЛЮЧЕНИЕ

Код-потокковые следящие вычислительные преобразователи представляют интерес при создании интеллектуальных измерительных систем. Они находят применение в структуре датчиков, оснащаемых встроенными промежуточными преобразователями, а также при формировании сложных функционально зависимых сигналов управления.

Учитывая потоковый характер обработки кодовой информации их применение эффективно в ситуациях, когда идет непрерывное отслеживание параметров в первичных преобразователях киберфизических систем.

Предлагаемые преобразователи обладают повышенной отказоустойчивостью, которая обеспечивается автоматическим восстановлением результата после сбоев.

Изменение технико-экономических характеристик элементной базы снимает ряд ограничений аппаратного характера и дает возможность по-новому оценивать

параметры проектируемых устройств и соответственно приводит к возможности обновления технологии информационной обработки. Реализация предлагаемых устройств, например, на программируемых логических микросхемах не представляет сложности. Возможно их изготовление и по технологиям заказных интегральных схем как в виде отдельных элементов, так и непосредственно в структуре датчиков.

СПИСОК ЛИТЕРАТУРЫ

- [1] Energy-Efficient Convolutional Neural Networks with Deterministic Bit-Stream Processing / S.R. Faraji, M. H. Najafi, B. Li, K. Bazargan, D.J. Lilja // The 2019 Design, Automation, and Test in Europe (DATE), Флоренция, Италия, 25–29 марта 2019 / 2019. С 1–6.
- [2] Kaining H., Warren J. G., Junchao W. Bit-Wise Iterative Decoding of Polar Codes using Stochastic Computing // IEEE Transactions on Signal Processing. 2019. Т. 67, вып 5. С. 1138–1151.
- [3] Assurance of Fault-Tolerance in Bit-Stream Computing Converters / A.I. Gulin, N.M. Safyannikov, O.I. Bureneva, A.Yu. Kaydanovich // Proceeding of 16th IEEE East-West Design & Test Symposium (EWDTS'2018), Казань, 14–17 сент. 2018 / Proceedings of IEEE, 2018. С. 418–421.
- [4] Stream tracking devices for soft measurements implementation / O.I. Bureneva // Proceedings of 2017 20th IEEE International Conference on Soft Computing and Measurements, SCM 2017, С-Петербург, 24–26 мая 2017 / Proceedings of IEEE, 2017. С. 614–616.
- [5] Suganth Paul, Nikhil Jayakumar, Sunil P. Khatri. A Fast Hardware Approach for Approximate, Efficient Logarithm Computations // IEEE transactions on very large scale integration (vlsi) systems. 2009. Т 17, вып. 2. С. 269–277.
- [6] J.-A. Pineiro. Algorithm and architecture for logarithm exponential and powering computation // IEEE Trans. Computers. 2004. Т 53, вып. 9. С. 1085–1096.