

# Следящие и адаптивные алгоритмы регулирования в цифровых преобразователях угла для СКВТ

Виктор САФРОНОВ,  
к. т. н.  
vik.saf@yandex.ru

**Современная электронная элементная база открывает новое направление — создание цифровых преобразователей угла (ЦПУ) с использованием в них принципов следящего и адаптивного цифрового регулирования, позволяющих увеличить точность измерения углового положения выходного вала СКВТ. В данной статье приведены примеры цифровых следящих и адаптивных алгоритмов, в том числе с переменной структурой.**

Компактность программ, содержащих предложенные алгоритмы, синтезированные с помощью E-операторного метода [2], позволяет зашивать их во флэш-память микроконтроллеров с RISC-архитектурой, например AVR Atmega, PIC или ARM Cortex-M3, и функционировать им в масштабе реального времени.

## Введение

Тема разработки датчиков угла и способов преобразования с высокой точностью аналоговых сигналов синусно-косинусных трансформаторов (СКВТ) с 30-х годов прошлого столетия находится в центре внимания всех разработчиков следящих систем, но до сих пор так и не исчерпана. Более того, современная электронная элементная база открывает новые возможности — создание следящих цифровых преобразователей угла (СЦПУ) с использованием в них принципов цифрового следящего и адаптивного регулирования.

Еще в середине 1980-х годов принцип следящего регулирования использовала фирма Analog Devices (США), выпустив на рынок

популярную серию микросхем AD2S, предназначенных для первичной обработки сигналов СКВТ [3]. Задача была решена чисто аппаратным способом, а монопольные цены на эти микросхемы до сих пор баснословные.

В конце 2015 года фирма «Миландр» (г. Зеленоград, Россия) планирует начать выпуск опытных микросхем, аналогичных продукции фирмы Analog Devices. Цены пока не установлены.

Авторы книги [1] в разделе «Следящие ЦПУ как замкнутая система автоматического регулирования» еще в 1987 году предложили применять известный в ТАР принцип следящего регулирования для конструирования ЦПУ.

В данной статье будут рассмотрены следующие вопросы:

- на основе структурных данных примера ЦПУ в [1] синтезирован алгоритм СЦПУ, структурно существенно отличающийся от цифро-аналоговых методов в следящих преобразователях-микросхемах типа AD2S фирмы Analog Devices [3] и от алгоритмов, примененных в продукции компании «Миландр»;
- синтезированные адаптивные алгоритмы, в том числе с переменной структурой, позволяют целенаправленно влиять на длительность переходного процесса и уменьшать погрешность измерения углового положения вала СКВТ.

Все указанные алгоритмы синтезированы с помощью E-операторного метода, изложенного в [2]. Программы, использующие эти компактные алгоритмы, позволяют зашивать их во флэш-память микроконтроллеров с RISC-архитектурой, например семейств AVR (Atmega-328), PIC16 или ARM (Cortex-M3), и затем функционировать в масштабе реального времени.

## Синтез алгоритма СЦПУ

Принцип следящего алгоритма работы микросхем серии AD2S подробно описан в [3], при этом при повороте вала СКВТ на угол  $\Omega$  демодулированное значение ошибки имеет вид:

$$\Delta U(\Omega, \Psi) = U_0 \times \sin(\Omega - \Psi). \quad (1)$$

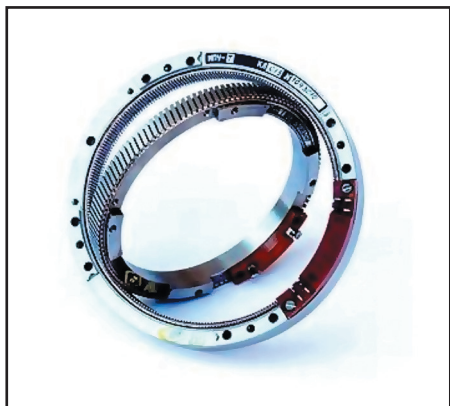
Учитывая, что следящая система по определению реализует сходящийся вычислительный процесс, то есть  $\lim_{t \rightarrow \infty} \Delta U(\Omega, \Psi) = 0$ , можно использовать свойство так называемого «первого замечательного предела»:  $\lim_{\Psi \rightarrow \Omega} \sin(\Omega - \Psi) = \Omega - \Psi$  и получить более простое выражение для вычисления ошибки в линейном виде:

$$\Delta U(\Omega, \Psi) = U_0 \times (\Omega - \Psi). \quad (2)$$

Однако очевидно, что выражение (2) справедливо лишь в узком диапазоне значений ошибки  $(\Omega - \Psi)$ , иначе следует пользоваться формулой (1), а это потребует больших временных и программных ресурсов, сложных аппаратных средств. Фирма Analog Devices решила эту задачу аппаратными средствами, введенными в состав семейства микросхем AD2S, и в результате до сих пор рыночные цены на эти микросхемы монопольно высоки.

Далее в статье мы используем тот же известный способ получения демодулированного сигнала (2), но будем решать эту задачу программно-аппаратными средствами иначе, для чего синтезируем альтернативный алгоритм СЦПУ, применив E-операторный метод, описанный в [2]. Кроме того:

- воспользуемся передаточными функциями из примера, приведенного в [1];



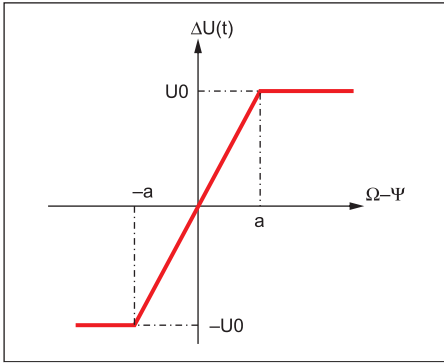


Рис. 1. Функция ошибки, формируемая нелинейным звеном

- предусмотрим для удобства пользователя формирование цифрового выходного сигнала и по углу, и по скорости изменения угла поворота вала СЦПУ (в отличие от аналогового выхода скорости в микросхемах семейства ADS2);
- запишем условие формирования ошибки в виде системы (3), принимая во внимание сходимость процесса слежения.

На рис. 1 показан график функции ошибки, а на рис. 2 — структура алгоритма СЦПУ.

В соответствии с примером в [1] имеем (физические смыслы констант, входящих в формулы, объяснены в том же источнике):

$$W_1(p) = \frac{k \times (1 + T_1 \times p)}{(1 + T_2 \times p) \times (1 + T_3 \times p)}, \quad (4)$$

$$W_2(p) = 1/p. \quad (5)$$

Теперь запишем структурные связи по рис. 2:

$$\begin{cases} U_1(p) = F_{вх}(p) - F_{вых}(p) \\ U_2(p) \leftarrow \overset{W_0}{\leftarrow} U_1(p) \\ V(p) = U_2(p) W_1(p) \\ F_{вых} = V(p) \times W_2(p) \end{cases} \quad (6)$$

Первому уравнению из системы (6) соответствует алгебраическое уравнение:

$$U_1(t) = f_{вх}(t) - f_{вых}(t). \quad (7)$$

Второму уравнению из системы (6) соответствуют условия (3), реализуемые программно в микроконтроллере.

Третье уравнение из системы (6) формирует выходной сигнал скорости изменения угла положения выходного вала СЦПУ. С учетом (4) ему соответствует:

$$V(p) = U_2(p) \times k \times \frac{(1 + T_1 \times p)}{(1 + T_2 \times p) \times (1 + T_3 \times p)}. \quad (8)$$

Четвертому уравнению из системы (6) с учетом (5) соответствует:

$$F_{вых}(p) = V(p) \times 1/p. \quad (9)$$

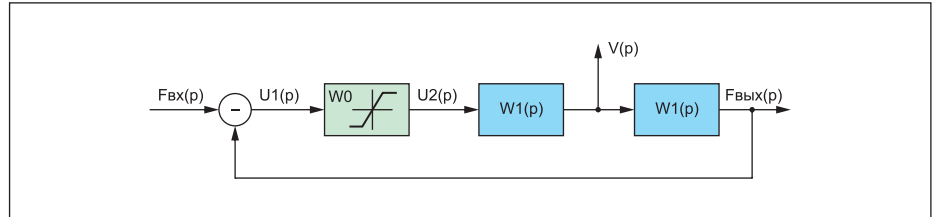


Рис. 2. Структура алгоритма СЦПУ с нелинейным звеном формирования ошибки:  
 $F_{вх}(p)$  — входное (демодулированное) воздействие, пропорциональное углу поворота вала;  
 $U_1(p), U_2(p)$  — промежуточные сигналы в структуре СЦПУ;  
 $F_{вых}(p)$  — выходное значение угла поворота вала (выходная реакция СЦПУ);  
 $V(p)$  — выход скорости изменения угла поворота вала

$$\begin{cases} \text{if } |\Omega - \Psi| \leq \alpha, \text{ then } \Delta U(t, \Omega, \Psi) = 1/\alpha \times U_0 \times (\Omega - \Psi) \\ \text{else } |\Delta U(t, \Omega, \Psi)| = U_0 \end{cases} \quad (3)$$

Уравнения (7) и (8), (9) в операторном виде вместе с условием (3) описывают функционирование алгоритмов СЦПУ. Перейдем теперь к синтезу самих алгоритмов, соответствующих этим формулам, применяя метод алгебраического  $E$ -операторного преобразования дифференциальных уравнений.

Этот эффективный метод преобразования дифференциальных и операторных уравнений к виду алгебраических разностных уравнений описан в [2]. Напомним, что  $E$ -оператор позволяет преобразовать как линейные дифференциальные уравнения, так и их операторные изображения по Лапласу в разностные алгебраические уравнения. Для этого достаточно в передаточной функции заменить производные  $(y)^k$  или операторы  $(p)^k$ , имеющие  $k$ -порядок, на соответствующие выражения вида  $(1-E)^k/\Delta t^k$  и затем вы-

разить явно нужную выходную переменную. При этом характерно, что полученное выражение представит собой линейную комбинацию выходной переменной со своими значениями, сдвинутыми во времени.

Итак, применяя  $E$ -оператор к (8), получим совсем простое алгебраическое уравнение (10):

$$v(t) = K_6 \times (U_2(t) \times E \times K_1 + U_2(t) \times K_2 - v(t) \times E^2 \times K_3 + v(t) \times E \times K_4). \quad (10)$$

Выражения для коэффициентов указаны ниже (рис. 3).

При желании числовые коэффициенты в уравнении можно почленно перемножить на  $K_6$ , но мы этого делать не станем, чтобы не потерять «физический» смысл коэффициентов.

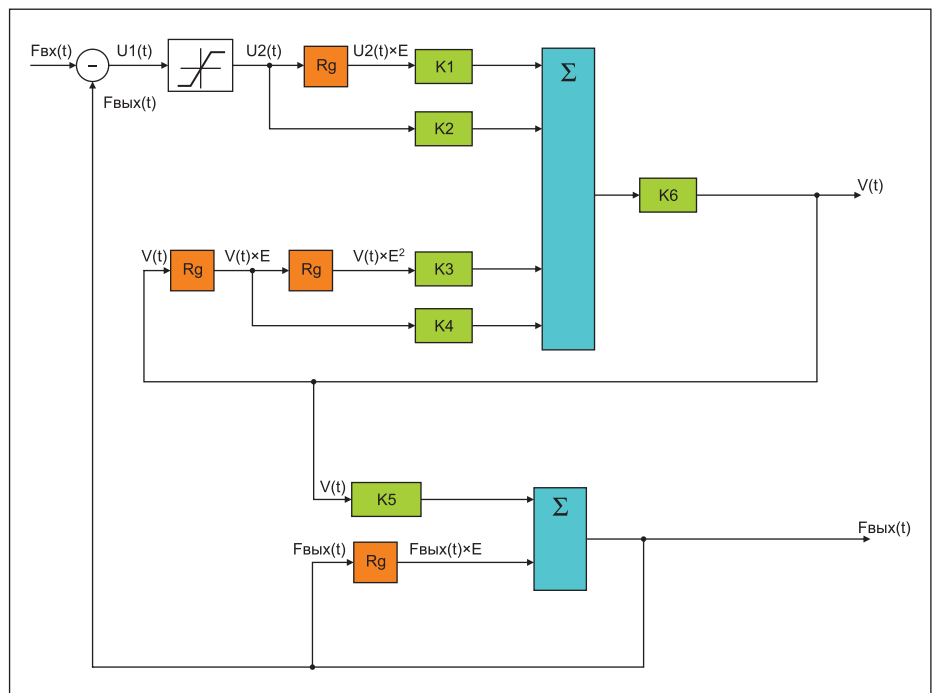


Рис. 3. Последовательный разностный алгоритм СЦПУ с нелинейным звеном формирования ошибки:  
 $\Sigma$  — сумматор; Rg — регистр

Конечно, здесь все коэффициенты — это просто числа, константы, вычисляемые и загружаемые во флэш-память микропроцессора в момент его программирования.

Теперь преобразуем (9) с помощью  $E$ -оператора:

$$f_{\text{вых}}(t) = v(t) \times \Delta t + f_{\text{вых}}(t) \times E. \quad (11)$$

И по полученным уравнениям (7, 10, 11) и условию (3) строим последовательный разностный алгоритм СЦПУ (рис. 3).

$K^*(1+T_1/dt)$  — числовой коэффициент, где:

$$K_1 = -k \times T_1 / \Delta t;$$

$$K_2 = k \times (1 + T_1 / \Delta t);$$

$$K_3 = T_2 \times T_3 / \Delta t^2;$$

$$K_4 = (T_2 + T_3 / \Delta t) + 2 \times (T_2 \times T_3 / \Delta t^2);$$

$$K_5 = \Delta t;$$

$$K_6 = \frac{1}{1 + \frac{T_2 + T_3}{\Delta t} + \frac{T_2 \times T_3}{\Delta \times t^2}}.$$

Напомним, что все коэффициенты — числовые константы, вычисляемые в процессе программирования до момента прошивки во флэш-память микропроцессора.

### Построение адаптивных алгоритмов СЦПУ второго и третьего рода

Для управления длительностями переходных процессов с целью уменьшения их влияния на точность измерений при резких возмущениях входного воздействия, то есть направления вращения и величины углового положения вала СКВТ, желательно иметь средство изменять структуру алгоритма СЦПУ для адаптивного сужения или расширения частотной полосы в ЛАХ от нуля до максимальной частоты среза.

В книге [1] описан пример подобной адаптации с помощью изменения структуры ЦПУ. Такой способ в литературе принято называть способом адаптации третьего рода.

В данной статье на основе указанного примера будет продемонстрировано два других способа (из трех возможных) адаптации структуры алгоритма СЦПУ:

- адаптация второго рода за счет варьирования значением полюса ( $T_3$ ) передаточной функции СЦПУ;
- адаптация третьего рода за счет изменения структуры передаточной функции

$$v(t) = \frac{U_2(t) \times E \times K_1 + U_2(t) \times K_2 - v(t) \times E^2 \times \lambda \times K_3 + v(t) \times E \times \left( \frac{T_2 + \lambda \times T_3}{\Delta t} + \frac{2 \times T_2 \times \lambda \times T_3}{\Delta t^2} \right)}{1 + \frac{T_2 + \lambda \times T_3}{\Delta t} + \frac{T_2 \times \lambda \times T_3}{\Delta t^2}}. \quad (12)$$

$$v(t) = \frac{U_2(t) \times E \times K_1 + U_2(t) \times K_2 + v(t) \times E \times (T_2 / \Delta t)}{1 + (T_2 / \Delta t)}. \quad (13)$$

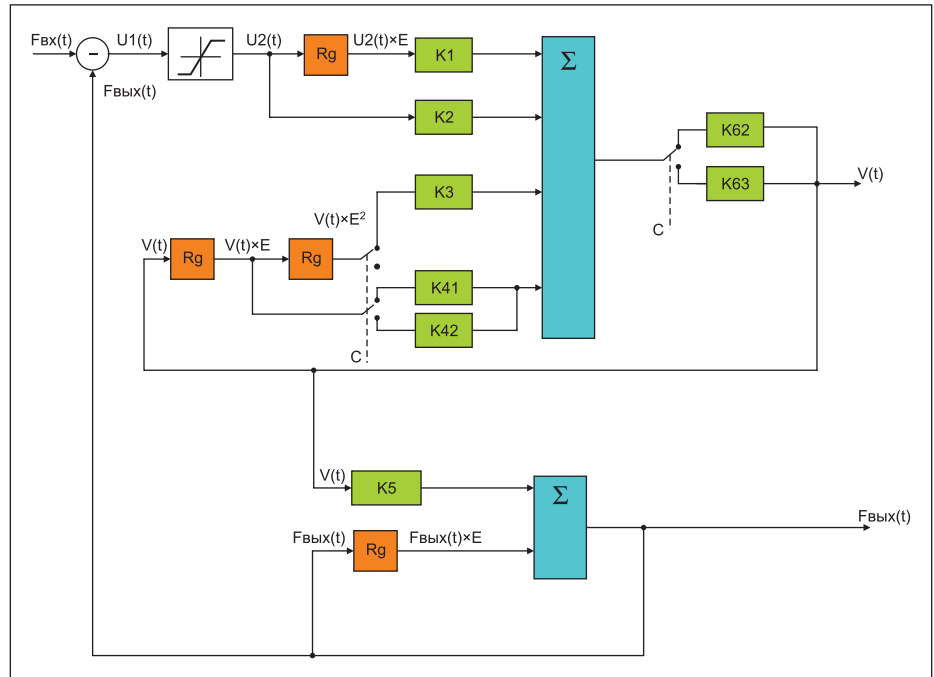


Рис. 4. Адаптивный алгоритм СЦПУ (третьего рода)

при помощи альтернативного переключения полюса ( $T_3$ ).

Выбор критериев адаптации и определения моментов переключения в процессах адаптации представляет собой самостоятельную задачу, выходящую за рамки данной статьи, и потому не рассматривается.

Адаптация второго рода в данном случае осуществляется очень просто — варьированием значения полюса ( $T_3$ ) в передаточной функции СЦПУ.

Особенно просто это выполнить в варианте программной реализации (10): достаточно заменить константу  $T_3$  на переменную  $\lambda \times T_3$ , где  $0 < \lambda \leq 1$  (важно, что  $\lambda \neq 0$ ), и получим (12).

Уравнения (7) и (11) не изменяются, поэтому структура (рис. 2) сохраняет свой вид и перерисовывать его еще раз нет необходимости, но лишь с заменой в соответствующих коэффициентах («зеленых квадратиках») константы  $T_3$  на переменную  $\lambda \times T_3$ .

Адаптация третьего рода осуществляется скачкообразным переключением в уравнении (12) значения  $\lambda > 0$  на значение  $\lambda = 0$ . При этом уравнение для  $v(t)$  при  $T_3 > 0$  будет иметь такой же вид, как (12), а при  $T_3 = 0$  станет (13).

Все остальные уравнения не изменяются, а структура алгоритма СЦПУ приобре-

тает соответствующий вид (рис. 4), где  $K_1, K_2, K_3, K_{41} = K_4, K_5, K_{61} = K_6$  — такие же, как на рис. 3.

$$K_{42} = T_2 / \Delta t; \\ K_{62} = 1 / (1 + (T_2 / \Delta t)).$$

Положения синхронных переключателей показаны условно. Программно выполнить эти переключения не составляет большой сложности с помощью условного оператора case или if, входящих в обработчик прерывания по событию.

Замечания к программной реализации:

- для реализации формул и условия (3) синтезированных алгоритмов на практике можно использовать недорогой микроконтроллер, например из семейств PIC, AVR или Cortex-M3;
- в регистрах хранятся глобальные переменные типа Real (действительное число), а в двух регистрах, соединенных последовательно, называемых «стек», сохраняется переменная типа Array of Real («массив» из двух действительных чисел, адресуемых сдвиговым указателем).

### Выводы

1. Синтез альтернативного алгоритма СЦПУ (по сравнению с продукцией фирмы Analog Devices) выполнен на примере ЦПУ, приведенном в отечественной литературе [1].
2. Синтезированные адаптивные алгоритмы СЦПУ второго и третьего рода позволяют целенаправленно влиять на переходные процессы, сужая или расширяя частотную полосу ЛАХ от нуля до максимальной частоты среза.

3. Представленные алгоритмы СЦПУ настолько компактны, что позволяют программно-аппаратными средствами на основе современных недорогих микроконтроллеров, например PIC, AVR или Cortex-M3, создавать эффективные вычислительные процессы для точного измерения угла поворота вала СКВТ.
4. Синтез алгоритмов проведен с применением математического  $E$ -операторного метода, изложенного в [2], позволяющего получить компактные алгоритмические структуры для построения по ним реальных вычислительных программ. ■

### Литература

1. Домрачев В. Г., Матвеевский В. Р., Смирнов Ю. С. Схемотехника цифровых преобразователей перемещений. Справочное пособие. М.: Энергоатомиздат, 1987.
2. Сафронов В. В. Синтез разностных алгоритмов управления цифровыми следящими электроприводами мобильных роботов  $E$ -операторным методом. Вестник ЮУрГУ. Серия «Компьютерные технологии, управление, радиоэлектроника». 2015. Т. 15 № 2.
3. Analog Device: 12-bit RDC with reference oscillator. Datasheet. [www.analog.com](http://www.analog.com)