

Купова А. В., Купов А. В., Соловьев В. А.
A. V. Kupova, A. V. Kupov, V. A. Solovyov

ПРОГРАММНАЯ РЕАЛИЗАЦИЯ НЕЧЁТКОГО РЕГУЛЯТОРА КОМПЕНСИРУЮЩЕГО УСТРОЙСТВА НА ЯЗЫКЕ ФУНКЦИОНАЛЬНЫХ БЛОКОВ

PROGRAM IMPLEMENTATION OF FUZZY COMPENSATING DEVICE REGULATOR IN THE LANGUAGE OF FUNCTIONAL BLOCKS

Купова Анастасия Викторовна – старший преподаватель кафедры «Теоретические основы электротехники» Санкт-Петербургского государственного электротехнического университета «ЛЭТИ» им. В. И. Ульянова (Ленина) (Россия, Санкт-Петербург). E-mail: avkupova@etu.ru.

Anastasia V. Kupova – Senior Lecturer, Theoretical Bases of Electrical Engineering Department, Saint Petersburg Electrotechnical University «LETI» (Russia, Saint Petersburg). E-mail: avkupova@etu.ru.

Купов Александр Викторович – кандидат технических наук, заместитель директора по аддитивным технологиям ООО «ПЕЕК» (Россия, Санкт-Петербург). E-mail: alek-erapu@mail.ru.

Alexander V. Kupov – Ph.D. in Technology, Deputy Director for Additive Technologies, PEEK LLC (Russia, Saint Petersburg). E-mail: alek-erapu@mail.ru.

Соловьев Вячеслав Алексеевич – доктор технических наук, профессор кафедры «Электропривод и автоматизация промышленных установок» Комсомольского-на-Амуре государственного университета (Россия, Комсомольск-на-Амуре). E-mail: erapu@knastu.ru.

Vyacheslav A. Solovyev – Doctor of Engineering, Professor, Electric Drive and Industrial Plant Automation Department, Komsomolsk-na-Amure State University (Russia, Komsomolsk-on-Amur). E-mail: erapu@knastu.ru.

Аннотация. Показан вариант реализации однокаскадного нечёткого регулятора для стабилизации напряжения в системе электроснабжения с компенсирующим устройством применительно к общепромышленным логическим контроллерам (ПЛК) на языке функциональных блоков по стандарту ГОСТ Р МЭК 61131-3-2016. Рассмотренный подход позволяет использовать универсальные ПЛК при отладке аппаратных макетов электротехнических систем, содержащих в контуре регулирования однокаскадные нечёткие регуляторы по Мамдани и Сугено.

Summary. The variant of implementation of a single-stage fuzzy regulator for voltage stabilization in the power supply system with a compensating device in relation to general industrial logic controllers (PLC) in the language of functional blocks according to GOST R IEC 61131-3-2016 standard is shown. The considered approach allows using universal PLCs when debugging hardware layouts of electrical systems containing in the control loop single-cascade fuzzy regulators by Mamdani and Sugeno.

Ключевые слова: нечёткая логика, компенсирующее устройство, стабилизация напряжения.

Key word: fuzzy logic, compensating device, voltage stabilization.

УДК 004.896

При проектировании регулятора для системы управления с быстроизменяющимися параметрами объекта бывает невозможно определить оптимальные коэффициенты для адаптивного регулятора, реализующего классические П-, ПИ- или ПИД-законы управления. Может наблюдаться как сильный разброс коэффициентов, исключающий возможность построения аппроксимирующей функции, так и определённая нечувствительность регулятора при некоторых состояниях объекта. В этом случае применение нечёткой логики (НЛ) в системе управления оправдано, т. к. позволяет справиться с перечисленными недостатками классических регуляторов.

Данный вывод подтверждают результаты исследований, выполненных на модели системы электроснабжения дуговой сталеплавильной печи, которая характеризуется как резкопеременная нагрузка [1; 2]. В данном случае нечёткая логика используется в системе управления статическим

тиристорным компенсатором для стабилизации уровня напряжения, что позволяет снизить колебательность измеряемых параметров и уровень статической ошибки, а также уменьшить время реакции системы управления.

Для широкого применения систем с нечёткой логикой требуется распространение аппаратных платформ с поддержкой операций нечёткой логики, таких как фаззификация, нечёткий вывод, дефаззификация и др. Несмотря на относительную простоту математического аппарата операций нечёткой логики [3; 4], далеко не всякий образец вычислительной платформы будет поддерживать полноценную работу с ней. Для этого потребуется как минимум формализация перечисленных операций на языке, понятном для исполнения данной ЭВМ.

Известно, что поддержку аппарата нечёткой логики на уровне промышленных логических контроллеров (ПЛК) предоставляют некоторые производители систем комплексной автоматизации, такие как Allen Bradley, Siemens и некоторые другие. Это не позволяет говорить о широкой доступности аппаратных платформ для реализации систем автоматизированного управления (САУ) с нечёткими регуляторами (НР). В связи с этим на этапе аппаратной отладки алгоритмов нечёткого регулирования приходится находить более доступные аппаратно-программные платформы и ПЛК, способные реализовать математический аппарат классического нечёткого регулятора, перечисленный выше. Одним из таких способов является отображение функций НР на одном из стандартных формальных языков, регламентированном по ГОСТ Р МЭК 61131-3-2016 [5]. Оптимальным был сочтён графический язык FBD как пригодный для визуального отображения проектируемого алгоритма. В полной мере аппарат языка FBD реализован отечественными разработчиками устройств для САУ «OWEN» в среде программирования Owen Logic. Однако, как говорилось выше, встроенная поддержка операций НЛ в среде Owen Logic отсутствует, что вынудило реализовать их вручную из имеющихся блоков математических и логических операций.

На рис. 1 показана структурная схема системы управления компенсирующим устройством для реализации на языке FBD. В соответствии с ГОСТ Р МЭК 61131-7-2017 [6] в канале регулирования явным образом должен присутствовать интегратор. Остальные блоки соответствуют модели НР, показанной в [1; 2].

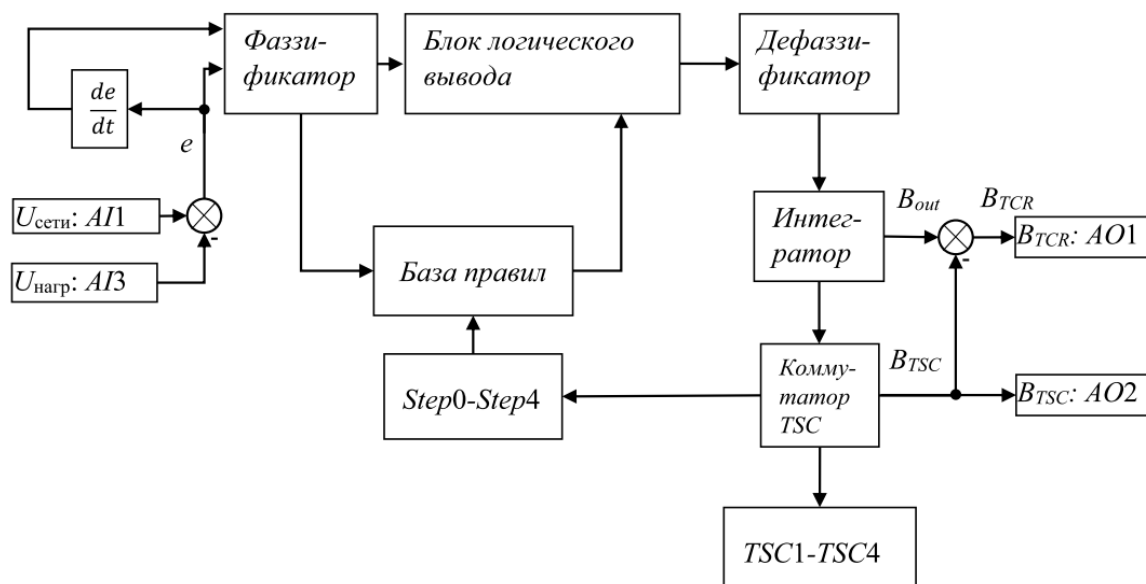


Рис. 1. Структурная схема системы управления компенсирующим устройством

На рис. 2 показаны функции принадлежности (ФП), используемые для фаззификации сигналов ошибки e и производной сигнала ошибки de/dt . В практической реализации входные переменные Step0 – Step4 представляют собой чёткие переменные, показывающие, какие фильтрокомпенсирующие цепи включены в настоящий момент. В НР эти переменные вводятся для исключения колебаний, которые появляются в результате частых переключений компенсирующих цепей.

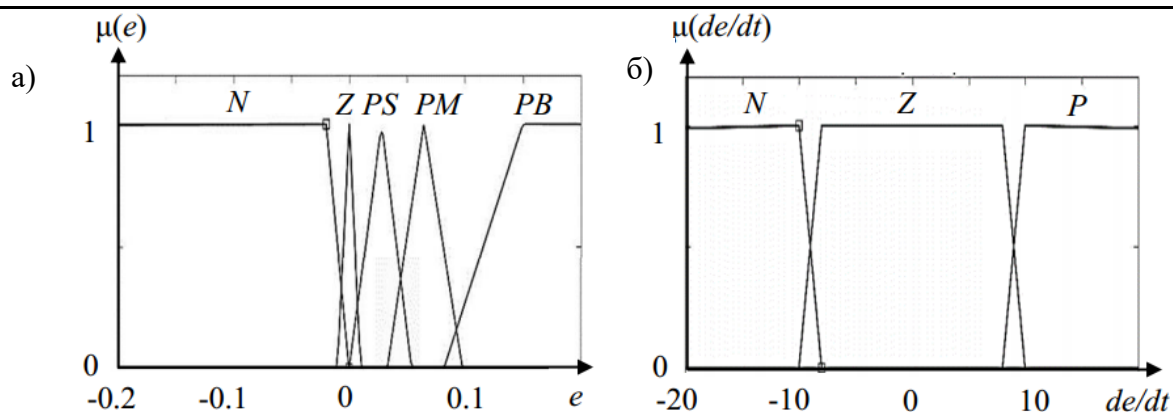


Рис. 2. Функции принадлежности сигналов ошибки e (а) и производная ошибки de/dt (б)

Для описания на языке функциональных блоков треугольные ФП были разделены на отрезки прямых $y(x) = kx$ и преобразованы к виду $Y = (X - X_{min}) / (X_{max} - X_{min})$. Каждый из отрезков представлен отдельной функциональной единицей – макросом. Макрос может включать в себя подмакросы для улучшения восприятия и упрощения отладки программы.

Элементы треугольных ФП, представленные линейными уравнениями, показаны на рис. 3.

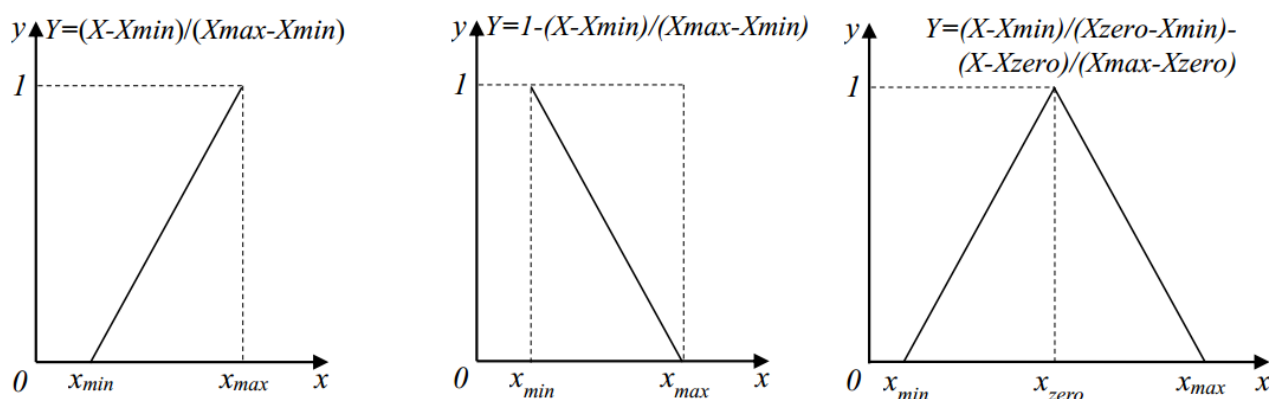


Рис. 3. Отображение уравнений трёхфазных ФП для фазификации на языке FBD

Если $k > 0$, то уравнение прямой имеет вид

$$Y = (X - X_{min}) / (X_{max} - X_{min}),$$

где X – текущее значение входной переменной x ; X_{min} – левая координата пересечения с осью абсцисс данной треугольной ФП; X_{max} – координата максимума данной ФП.

Если $k < 0$, то уравнение прямой преобразуется в выражение

$$Y = 1 - (X - X_{min}) / (X_{max} - X_{min}).$$

Следовательно, для оцифровки треугольной ФП можно использовать выражение

$$Y = (X - X_{min}) / (X_{zero} - X_{min}) - (X - X_{zero}) / (X_{max} - X_{zero}),$$

где X_{zero} – координата вершины треугольника ФП.

Пример формирования макроса для треугольной ФП показан на рис. 4. Блоки на приведённой схеме имеют следующие значения:

fSUB – определяет разность входных значений;

fDIV – вычисляет отношение значений на входе;

fGT – сравнивает входные значения. Если входные значения равны, то на выход подаётся логический 0, иначе логическая 1;

fSEL – условная операция сравнения. Если на первом входе логический 1, то на выход передаётся значение второго входа. Иначе на выход поступает значение третьего входа.

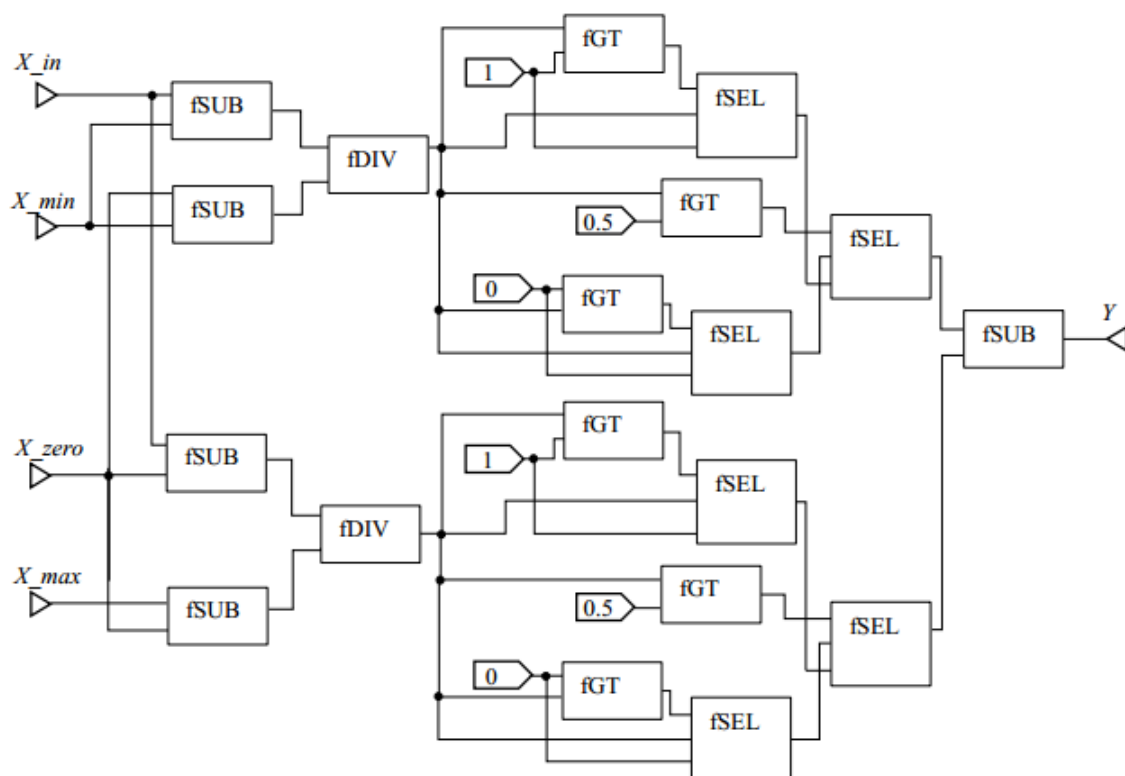


Рис. 4. Макрос для треугольных ФП

Для фаззификации по трапецеидальным ФП следует разделить их на два вида. Первый вид – трапеция, у которой одна из сторон уходит в бесконечность. К таким трапециям относятся ФП N , PB переменной e и ФП N и P переменной de/dt (см. рис. 2). В программе фаззификация выполняется в зависимости от попадания значения в диапазон существования наклонной линии: в пределах существования фаззификация выполняется по наклонной линии; вне пределов – по максимуму, равному 1. На рис. 5, 6 представлены примеры формирования макросов для них.

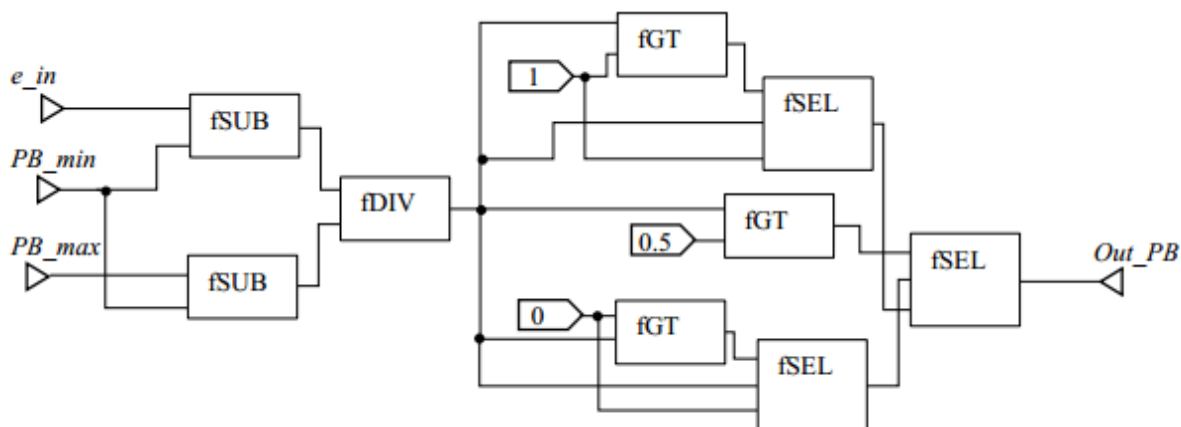


Рис. 5. Макрос, описывающий наклонную линию ФП PB

Второй вид трапеций – это трапеция вида ФП Z переменной de/dt (см. рис. 2). Её описание выполняется из выражений для треугольной ФП путём раздельного описания вершины треугольника двумя разнесёнными точками:

$$Y = (X - X_{min1}) / (X_{min2} - X_{min1}) - (X - X_{max1}) / (X_{max2} - X_{max1}).$$

Макрос, описывающий трапецидальную ФП, подобен показанному на рис. 4, но с использованием ещё одного входа. Таким образом, на первую пару блоков fSUB поступают сигналы X_{in} , X_{min2} , X_{min1} , на вторую пару – X_{in} , X_{max2} , X_{max1} .

Формирование базы правил нечёткого регулятора осуществляется на логических конструкциях вида «Если $TSC \& e \& de/dt$, то B » по табл. 1. B – выходная переменная, функции принадлежности которой показаны на рис. 7.

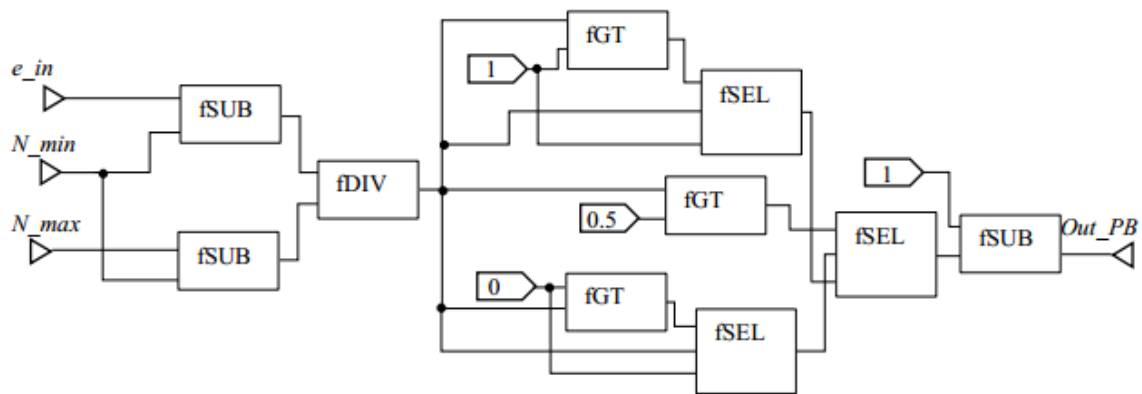


Рис. 6. Макрос, описывающий наклонную линию ФП N

Таблица 1

База правил нечёткого регулятора

de/dt	e	TSC				
		Step0	Step1	Step2	Step3	Step4
Z	N	C0	C0	C0	C0	C0
	Z	C0	C0	C0	C1	C1
	PS	C1	C1	C1	C3	C4
	PM	C1	C2	C3	C4	C4
	PB	C4	C4	C4	C4	C4
P	N	C0	C1	C2	C3	C4
	Z	C0	C1	C2	C3	C4
	PS	C1	C1	C2	C3	C4
	PM	C1	C2	C3	C4	C4
	PB	C2	C3	C4	C4	C4
N	N	C0	C0	C0	C1	C2
	Z	C0	C0	C1	C2	C3
	PS	C0	C0	C1	C2	C3
	PM	C0	C1	C2	C3	C4
	PB	C0	C1	C2	C3	C4

Процедура дефаззификации для вывода чёткого значения выходной величины B_{out} может быть реализована, в частности, по методу центра тяжести:

$$B_{out} = \frac{\int xz(x)dx}{\int z(x)dx}. \quad (1)$$

В данном случае в состав дефаззификатора будут входить одностипные макросы для описания числителя и знаменателя центроида как площади трапеции, показанной на рис. 8.

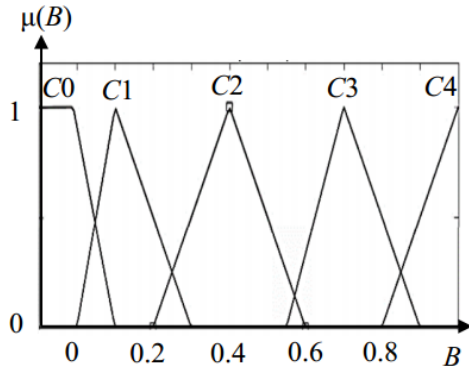


Рис. 7. Функции принадлежности выходной переменной B

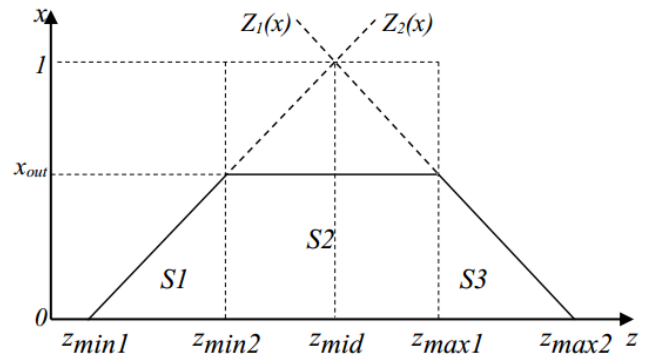


Рис. 8. Графическое представление формирования $z(x)$

Уравнения прямых:

$$Z_1(x) = x(Z_{mid} - Z_{min1}) + Z_{min1},$$

$$Z_2(x) = x(Z_{mid} - Z_{max2}) + Z_{max2}.$$

Координаты $Z_{min2}(x)$ и $Z_{max1}(x)$ определяются при $x = x_{out}$:

$$Z_{min2}(x) = Z_1(x_{out}),$$

$$Z_{max1}(x) = Z_2(x_{out}).$$

Площади результирующих многоугольников рассчитываются по выражениям

$$S_1 = \frac{x_{out}}{2} (Z_{min2}(x) - Z_{min1}) = \frac{x_{out}}{2} [x_{out} (Z_{mid} - Z_{min1}) + Z_{min1} - Z_{min1}] = \frac{x_{out}^2}{2} (Z_{mid} - Z_{min1});$$

$$S_2 = x_{out} (Z_{max1}(x) - Z_{min2}(x)) = x_{out} [(x_{out} (Z_{mid} - Z_{max2}) + Z_{max2}) - (x_{out} (Z_{mid} - Z_{min1}) + Z_{min1})];$$

$$S_3 = \frac{x_{out}}{2} (Z_{max2} - Z_{max1}(x)) = \frac{x_{out}}{2} [Z_{max2} - x_{out} (Z_{mid} - Z_{max2}) - Z_{max2}] = \frac{x_{out}^2}{2} (Z_{max2} - Z_{mid}).$$

Для представления на языке функциональных блоков интегрирование в выражении (1) может быть заменено суммой. На рис. 9 показан макрос, вычисляющий чёткое значение B_{out} .

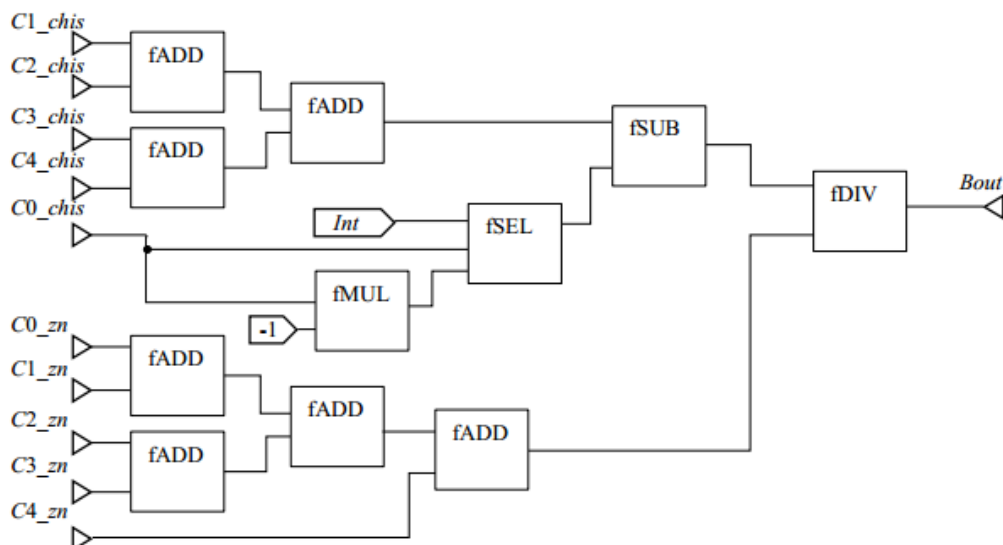


Рис. 9. Макрос, реализующий вычисление B_{out}

На рис. 9 приводятся функции суммирования значений входных сигналов fADD и перемножения fMUL. Переменная *Int* используется для исключения неопределённости, возникающей при вычислениях *Bout* в районе нулевых значений. Определение сигналов *C0_chis* – *C0_chis*, *C0_zn* – *C4_zn* выполняется в макросе, описание которого показано на рис. 10. Входные сигналы в макрос задаются по функциям принадлежности выходной переменной *B* и сигнала *x_out*, полученного по базе правил НР.

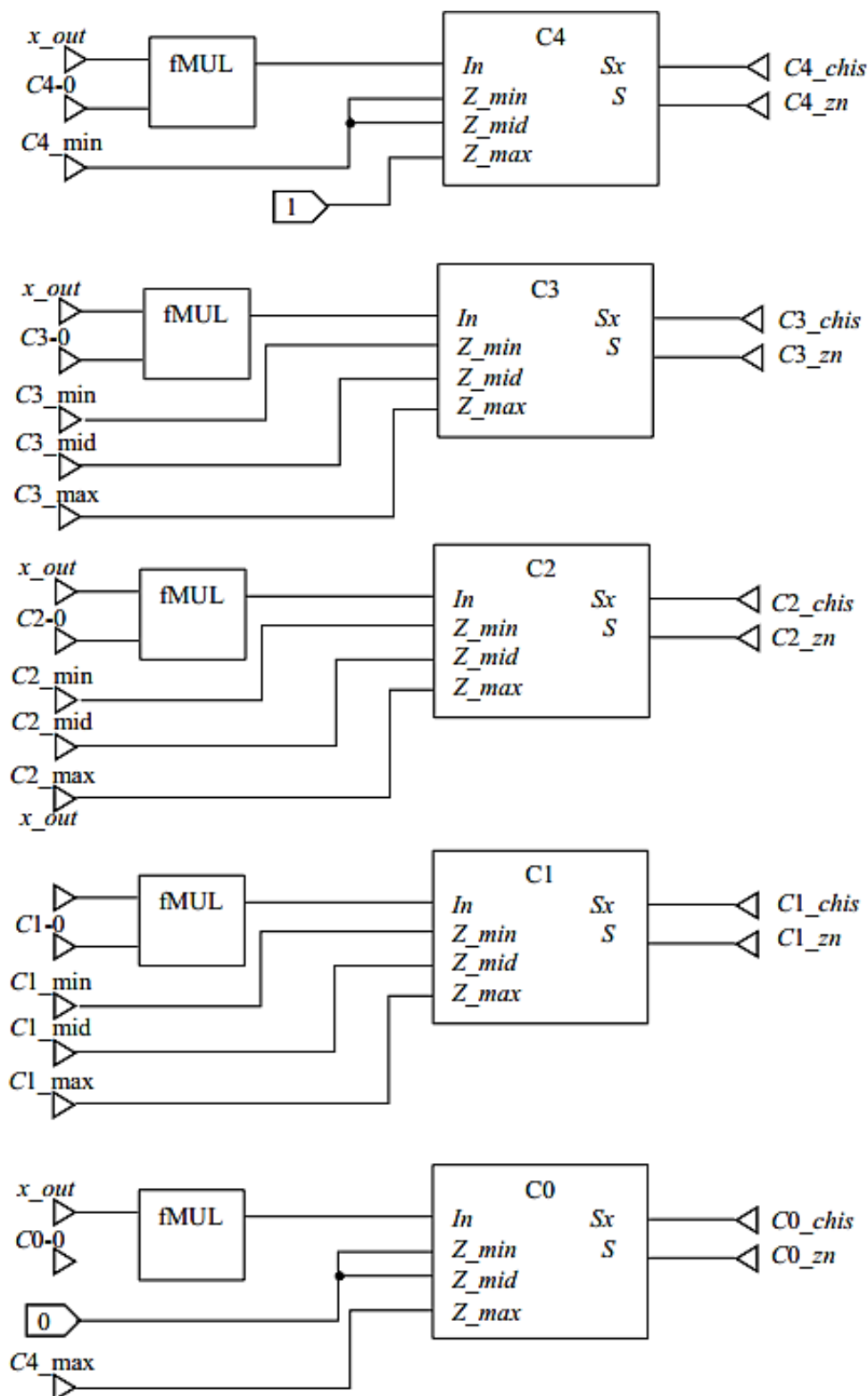


Рис. 10. Формирование сигналов *C0_chis* – *C0_chis*, *C0_zn* – *C4_zn*

На рис. 10 блоки C0 – C4 содержат подмакросы, определяющие значения $xz(x)$ и $z(x)$ для вычисления выходного сигнала по формуле (1). Их реализация показана на рис. 11.

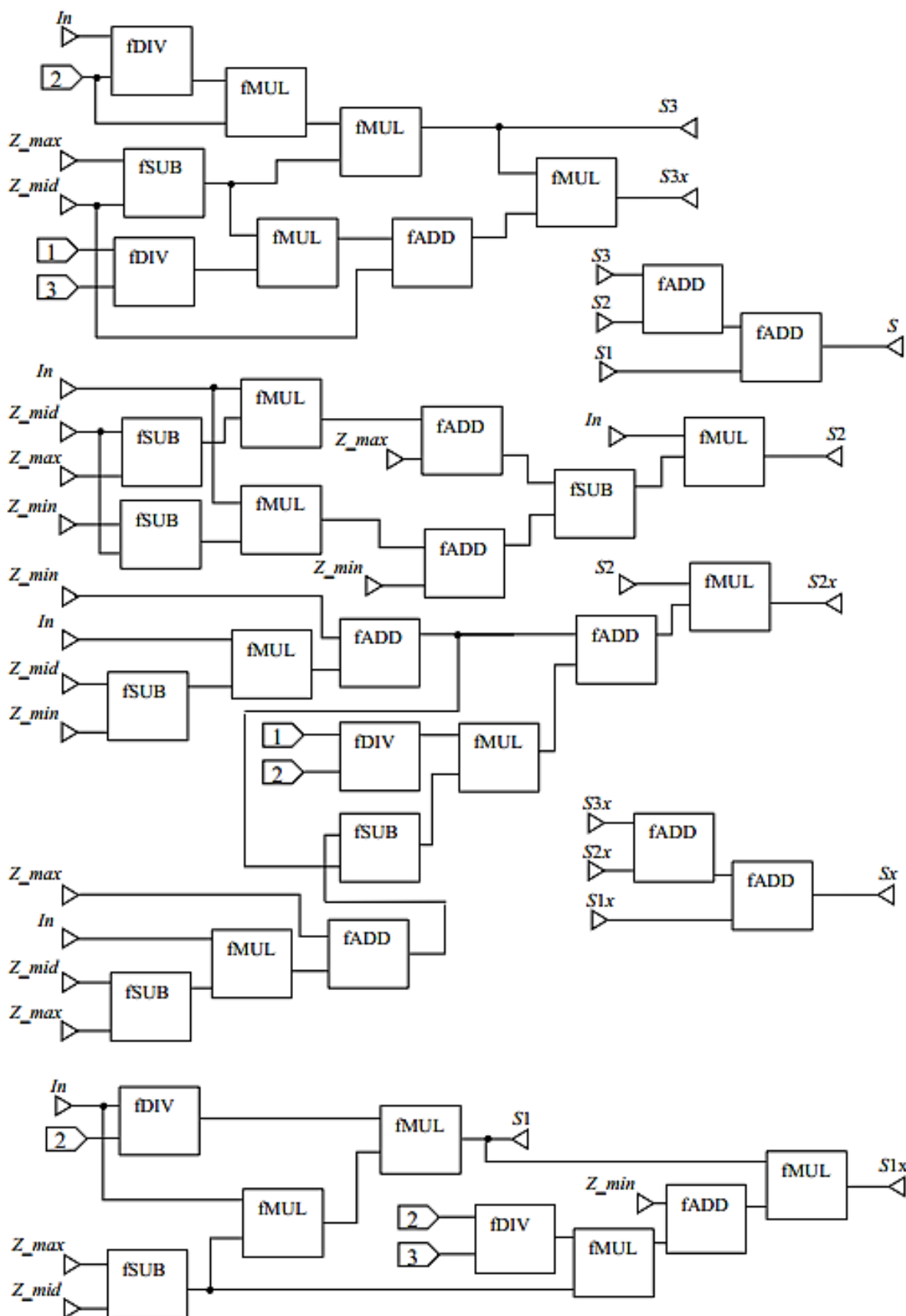


Рис. 11. Формирование значений $xz(x)$ и $z(x)$

Очевидно, что центроид площади, определённый рассматриваемой программой, не обеспечивает максимальное значение сигнала $B = 1$. При отладке было достигнуто максимальное значение $B = 0.933$. Поэтому было установлено единичное значение предела интегрирования $B_{out-I} = 1$, что обеспечило корректную работу коммутатора компенсатора TSC . Также в процессе отладки проявилась необходимость в коррекции определения выходного сигнала в районе нулевых значений и определения дополнительной переменной Int .

При выполнении натурных экспериментов на макете с программируемым реле ПР200 фирмы «OWEN» выявлено время выполнения одного цикла программного регулирования – менее 10 мс. Это обеспечивает регулируемость стационарных электромагнитных процессов в сетях промышленной частоты 50 Гц. В то же время было выявлено, что дискретизация аналоговых входов ПР 200 добавляет задержку в обработке сигналов порядка 33 мс. Однако практическая реализация нечёткого регулятора показала принципиальную возможность использования неспециализированных платформ для нечёткого управления компенсирующими устройствами с переменной нагрузкой.

ЛИТЕРАТУРА

1. Дерюжкова, Н. Е. Нечёткие алгоритмы управления статическим тиристорным компенсатором в системе электроснабжения с переменной нагрузкой / Н. Е. Дерюжкова, А. В. Купова, В. А. Соловьев // Учёные записки Комсомольского-на-Амуре государственного технического университета. Науки о природе и технике. – 2020. – № I-1 (41). – С. 51-58.
2. Стабилизация напряжения статическим тиристорным компенсатором с нечётким регулятором в системе с переменной нагрузкой / А. В. Купова, В. А. Соловьев, А. В. Светлаков, В. А. Челухин // Учёные записки Комсомольского-на-Амуре государственного технического университета. Науки о природе и технике. – 2021. – № VII (55). – С. 48-57.
3. Демидова, Г. Л. Регуляторы на основе нечёткой логики в системах управления техническими объектами / Г. Л. Демидова, Д. В. Лукичев. – СПб.: Университет ИТМО, 2017. – 81 с.
4. Нечёткое моделирование и управление в технических системах: учеб. пособие для вузов / Ю. И. Кудинов, Ф. Ф. Пашенко, И. Ю. Кудинов, А. Ф. Пашенко. – СПб.: Лань, 2020. – 208 с.
5. ГОСТ Р МЭК 61131-3-2016. Контроллеры программируемые. Часть 3. Языки программирования = Programmable controllers. Part 3. Programming languages: национальный стандарт Российской Федерации: издание официальное: утверждён и введён в действие Приказом Федерального агентства по техническому регулированию и метрологии от 13 мая 2016 г. № 313-ст: введён впервые: дата введения 2017-04-01 / подготовлен Негосударственным образовательным частным учреждением «Новая Инженерная Школа» (НОЧУ «НИШ»). – М.: Стандартиформ, 2016. – 227 с.
6. ГОСТ Р МЭК 61131-7-2017. Контроллеры программируемые. Часть 7. Программирование нечёткого управления = Programmable controllers. Part 7. Fuzzy control programming: национальный стандарт Российской Федерации: издание официальное: утверждён и введён в действие Приказом Федерального агентства по техническому регулированию и метрологии от 15 сентября 2017 г. № 1127-ст: введён впервые: дата введения 2018-09-01 / подготовлен Негосударственным образовательным частным учреждением «Новая Инженерная Школа» (НОЧУ «НИШ»). – М.: Стандартиформ, 2017. – 48 с.