SystemePLC Studio

Программное обеспечение SystemePLC Studio

для интеллектуальных реле SystemePLC SR и ПЛК SystemePLC S172

Руководство пользователя

systeme.ru

Système electric

Содержание

1. Установка ПО SystemePLC Studio 5
1.1 Минимальные системные требования для SystemePLC Studio 5
1.2 Инструкция по установке SystemePLC Studio 5
1.3 Рекомендации по установке SystemePLC Studio 6
2. Библиотека функциональных блоков и функций 7
2.1 Преобразование базовых типов (Basic Type conversion) 7
2.2 Преобразование базовых типов в BCD (BCD conversion) 7
2.3 Преобразование типов времени (TIM conversion) 7
2.4 Числовые операции (Numerical) 7
2.5 Арифметические операции (Arithmetic) 9
2.6 Операции со временем (Time) 11
2.7 Битовые сдвиги (Bit Shift) 13
2.8 Побитовые операции (Bit wise) 15
2.9 Операции выбора (Selection) 17
2.10 Операции сравнения (Comparison) 19
2.11 Операции со строками (Character String) 20
2.12 Стандартные функциональные блоки (Standard Function Blocks) 23
2.12.1 SR (триггер с приоритетом Set) 23
2.12.2 RS (триггер с приоритетом Reset) 23
2.12.3 R_TRIG (Детектор фронта 0->1) 24
2.12.4 F_TRIG (Детектор фронта 1->0) 24
2.12.5 CTU (Счетчик импульсов с предустановкой) 25
2.12.6 CTU_LINT (Счетчик импульсов с предустановкой, 64-битный) 26
2.12.7 CTU_UDINT (Счетчик импульсов с предустановкой, 32-битный беззнаковый) 26
2.12.8 CTU_ULINT (Счетчик импульсов с предустановкой, 64-битный беззнаковый) 27
2.12.9 CTD (Счетчик обратного счета с предустановкой) 28
2.12.10 CTD_LINT (Счетчик обратного счета с предустановкой, 64-битный) 29
2.12.11 CTD_UDINT (Счетчик обратного счета с предустановкой, 32-битный беззнаковый) 30
2.12.12 CTD_ULINT (Счетчик обратного счета с предустановкой, 64-битный беззнаковый) 30
2.12.13 CTUD (Реверсивный счетчик) 31

2.12.14 CTUD_LINT (Реверсивный счетчик, 64-битный) ... 32
2.12.15 CTUD_UDINT (Реверсивный счетчик, 32-битный беззнаковый) ... 34
2.12.16 CTUD_ULINT (Реверсивный счетчик, 64-битный беззнаковый) ... 35
2.12.17 TP (Импульсный таймер) ... 36
2.12.18 TON (Таймер включения с задержкой) ... 37
2.12.19 TOF (Таймер выключения с задержкой) ... 38
2.13 Интеллектуальные функциональные блоки (Smart Function Blocks) ... 38
2.13.1 GET_TICKS ... 38
2.13.2 GET_RTC ... 39
2.13.3 SET_RTC ... 39
2.13.4 POPUP_PAGE ... 40
2.13.5 GEN_RAND ... 40
2.13.6 PIDAdvanced ... 40
2.13.7 PIDAutoTuning ... 47
2.13.8 advPID ... 50
2.14 Функции преобразования битов (Bit Conversion Functions) ... 54
2.15 Специальные функции (Special functions) ... 57
2.15.1 Hours_Counter (Счетчик моточасов) ... 57
2.15.2 Threshold_Trigger (Пороговый триггер) ... 59
2.15.3 UpDown_Counter (Реверсивный счетчик) ... 60
2.15.4 Analog_Amplifier (Аналоговый усилитель) ... 62
2.15.5 Analog_Comparator (Аналоговый компаратор) ... 63
2.15.6 Analog_DifferentialTrigger (Аналоговый дифференциальный триггер) ... 65
2.15.7 Analog_Filter (Аналоговый фильтр) ... 67
2.15.8 Analog_MUX (Аналоговый мультиплексор) ... 68
2.15.9 Analog_Ramp (Аналоговый рамповый генератор) ... 69
2.15.10 Analog_ThresholdTrigger (Аналоговый пороговый триггер) ... 70
2.15.11 Analog_Watchdog (Аналоговый сторожевой таймер) ... 72
2.15.12 AverageValue (Усреднение значения) ... 74
2.15.13 Mathematicinstruction (Математическая инструкция) ... 76
2.15.14 MaxMin (Максимум/минимум) ... 79
2.15.15 PWM (ШИМ-модулятор) ... 81

2.15.16 Latching_Relay (Бистабильное реле) 84
2.15.17 Pulse_Relay (Импульсное реле) 85
2.15.18 Shift_Register (Сдвиговый регистр) 86
2.15.19 Asynchronous_PulseGenerator (Асинхронный генератор импульсов) 88
2.15.20 EdgeTriggered_WipingRelay (Триггерное реле с задержкой) 89
2.15.21 MultipleFunctionSwitch (Многофункциональный переключатель) 91
2.15.22 Off_Delay (Таймер задержки выключения) 93
2.15.23 On_Delay (Таймер задержки включения) 94
2.15.24 OnOff_Delay (Таймер задержки включения/выключения) 95
2.15.25 Random_Generator (Случайный генератор) 96
2.15.26 Retentive_OnDelay (Таймер включения с памятью) 97
2.15.27 StairwayLightingSwitch (Лестничный выключатель) 98
2.15.28 Stopwatch (Секундомер) 99
2.15.29 Weekly_Timer (Недельный таймер) 101
2.15.30 Wiping_Relay (Импульсное реле) 103
2.15.31 Yearly_Timer (Годовой таймер) 104
3. Коды ошибок 109
3.1 Внутренние коды ошибок ПО SystemePLC Studio 109
3.1.1 Журнал аварийных событий ПЛК (Alarm History) с фиксированным числом записей (N) 112
3.1.2 Таблица кодов ошибок SystemePLC Studio 114
3.2 Коды ошибок Modbus 117
3.2.1 Описание кода ошибки Modbus 118

Введение

Программное обеспечение SystemePLC Studio представляет собой среду программирования для всех интеллектуальных реле SystemePLC SR и ПЛК SystemePLC S172. Оно используется для конфигурирования и разработки приложений, поставляется бесплатно и доступно для скачивания с облачного хранилища.
ПО и прошивки устройств периодически обновляются и будут доступны по ссылке.

Данный документ содержит описание библиотеки функциональных блоков и справочную информацию по основным функциям ПО SystemePLC Studio.

В дополнение к данному руководству пользователя обратите внимание на **обучение SystemePLC Studio «Быстрый Старт»** по ссылке**,** которое состоит из нескольких частей и призвано осуществить знакомство пользователя со средой программирования SystemePLC Studio.

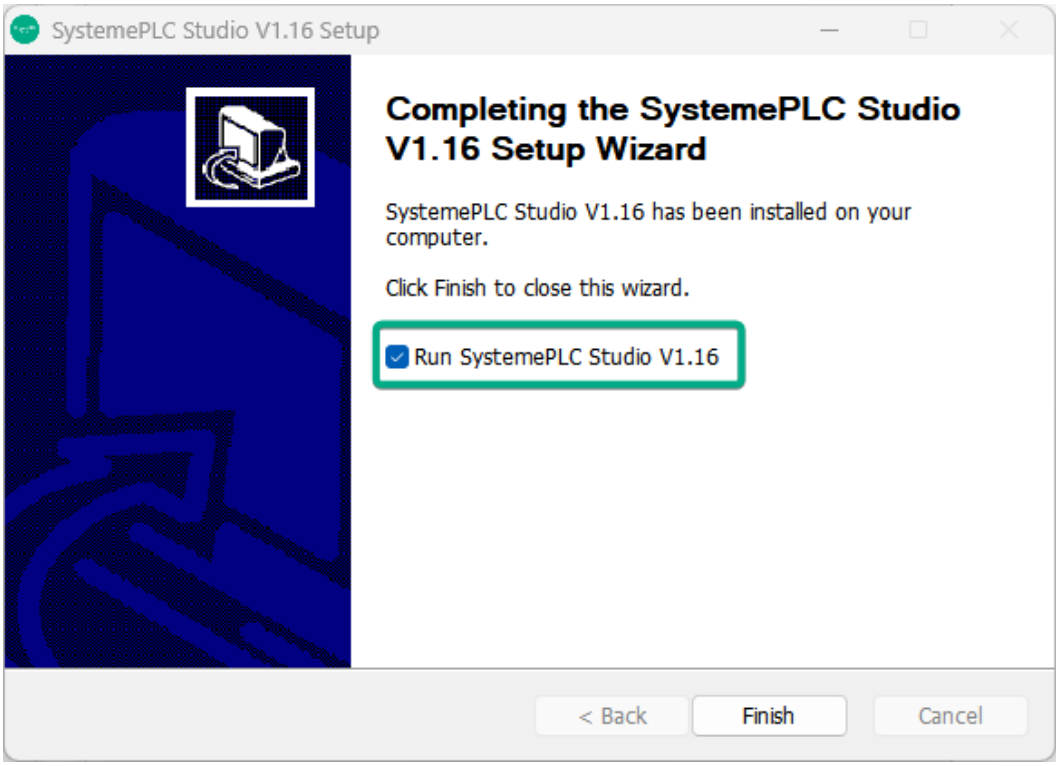
1. Установка ПО SystemePLC Studio

1.1 Минимальные системные требования для SystemePLC Studio

- Операционная система: Windows 10 64-bit / Windows 11 64-bit
- Процессор: процессор 64-bit совместимый с частотой 2 ГГц или выше (двухъядерный или многоядерный)
- Оперативная память: 4 ГБ или больше
- Место на жестком диске: 10 ГБ или больше
- Рекомендуется использовать разрешение дисплея: 1920×1080 (1080p) или выше

1.2 Инструкция по установке SystemePLC Studio

- Запустите исполняемый файл SystemePLC Studio V.xxx.exe
- Следуйте инструкциям установщика
- Оставьте включенной галочку «Run SystemePLC Studio» на завершающем экране установки ПО (для запуска инициализации ПО при первом запуске)



1.3 Рекомендации по установке SystemePLC Studio

- Перед установкой новой версии ПО SystemePLC Studio удалите старую версию.
- После установки новой версии ПО SystemePLC Studio, обновите прошивки ПЛК. Прошивки ПЛК предоставляются вместе с дистрибутивом ПО.

2. Библиотека функциональных блоков и функций

2.1 Преобразование базовых типов (Basic Type conversion)

В соответствии со стандартом IEC 61131-3, функции преобразования типов должны иметь форму **TO***, где *** — это тип входной переменной, а ***** — тип выходной переменной.

2.2 Преобразование базовых типов в BCD (BCD conversion)

№	Функция преобразования	Детали преобразования
1	BCD_TO_USINT	Преобразование кода BCD в значение USINT.
2	BCD_TO_UINT	Преобразование кода BCD в значение UINT.
3	BCD_TO_UDINT	Преобразование кода BCD в значение UDINT.
4	BCD_TO_ULINT	Преобразование кода BCD в значение ULINT.
5	USINT_TO_BCD	Преобразование значений USINT в формат кода BCD.
6	UINT_TO_BCD	Преобразование значений UINT в формат кода BCD.
7	UDINT_TO_BCD	Преобразование значений UDINT в формат кода BCD.
8	ULINT_TO_BCD	Преобразование значений ULINT в формат кода BCD.

2.3 Преобразование типов времени (TIM conversion)

1.3.1 DATE_AND_TIME_TO_TIME_OF_DAY

Дата и время во время дня.

Имя	Ввод/Вывод	Тип данных	Описание
IN	Input	DT	
OUT	Output	TOD	

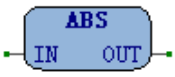
1.3.2 DATE_AND_TIME_TO_DATE

Дата и время в дату.

Имя	Ввод/Вывод	Тип данных	Описание
IN	Input	DT	
OUT	Output	DATE	

2.4 Числовые операции (Numerical)

1.4.1 ABS (Абсолютное значение)



Имя	Ввод/Вывод	Тип данных	Описание

IN	Input	ANY_NUM	абсолютное значение
OUT	Output	ANY_NUM	Например: ABS(X)=Y

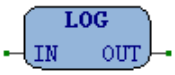
1.4.2 SQRT (Квадратный корень)

Имя	Ввод/Вывод	Тип данных	Описание
IN	Input	ANY_REAL	Квадратный корень
OUT	Output	ANY_REAL	Например: SQRT(X)=Y

1.4.3 LN (Натуральный логарифм)

Имя	Ввод/Вывод	Тип данных	Описание
IN	Input	ANY_REAL	Натуральный логарифм
OUT	Output	ANY_REAL	Например: LN(X)=Y

1.4.4 LOG (Логарифм по основанию 10)

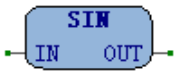


Имя	Ввод/Вывод	Тип данных	Описание
IN	Input	ANY_REAL	Логарифм по основанию 10
OUT	Output	ANY_REAL	Например: LOG(X)=Y

1.4.5 EXP (Экспонента)

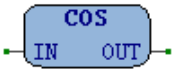
Имя	Ввод/Вывод	Тип данных	Описание
IN	Input	ANY_REAL	Натуральная экспонента
OUT	Output	ANY_REAL	Например: EXP(X)=Y

1.4.6 SIN (Синус)



Имя	Ввод/Вывод	Тип данных	Описание
IN	Input	ANY_REAL	Синус входа в радианах
OUT	Output	ANY_REAL	Например: SIN(X)=Y

1.4.7 COS (Косинус)



Имя	Ввод/Вывод	Тип данных	Описание
IN	Input	ANY_REAL	Косинус в радианах
OUT	Output	ANY_REAL	Например: COS(X)=Y

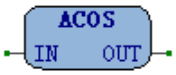
1.4.8 TAN (Тангенс)

Имя	Ввод/Вывод	Тип данных	Описание
IN	Input	ANY_REAL	Тангенс в радианах
OUT	Output	ANY_REAL	Например: TAN(X)=Y

1.4.9 ASIN (Арксинус)

Имя	Ввод/Вывод	Тип данных	Описание
IN	Input	ANY_REAL	Главный арксинус
OUT	Output	ANY_REAL	Например: ASIN(X)=Y

1.4.10 ACOS (Арккосинус)



Имя	Ввод/Вывод	Тип данных	Описание
IN	Input	ANY_REAL	Главный арккосинус
OUT	Output	ANY_REAL	Например: ACOS(X)=Y

1.4.11 ATAN (Арктангенс)

Имя	Ввод/Вывод	Тип данных	Описание
IN	Input	ANY_REAL	Главный арктангенс
OUT	Output	ANY_REAL	Например: ATAN(X)=Y

2.5 Арифметические операции (Arithmetic)

Инструкции ADD, MUL, SUB, DIV могут добавлять несколько входных контактов путем щелчка правой кнопкой мыши на функциональном блоке в программе и выбора "Increase pins".

1.5.1 ADD (Сложение)

Этот функциональный блок используется для сложения переменных или констант. Типы входных и выходных переменных должны быть одинаковыми.

Имя	Ввод/Вывод	Тип данных	Описание
IN1	Input	ANY_NUM	
IN2	Input	ANY_NUM	OUT = IN1 + IN2
OUT	Output	ANY_NUM	

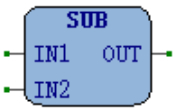
1.5.2 MUL (Умножение)

Этот функциональный блок используется для умножения переменных или констант. Типы входных и выходных данных должны быть одинаковыми.

Имя	Ввод/Вывод	Тип данных	Описание
IN1	Input	ANY_NUM	
IN2	Input	ANY_NUM	OUT = IN1 * IN2
OUT	Output	ANY_NUM	

1.5.3 SUB (Вычитание)

Этот функциональный блок используется для вычитания переменных или констант. Типы входных и выходных данных должны быть одинаковыми.



Имя	Ввод/Вывод	Тип данных	Описание
IN1	Input	ANY_NUM	OUT = IN1 - IN2
IN2	Input	ANY_NUM	
OUT	Output	ANY_NUM	

1.5.4 DIV (Деление)

Функциональные блоки используются для деления переменных или констант. Типы входных и выходных переменных должны быть одинаковыми.

Имя	Ввод/Вывод	Тип данных	Описание
IN1	Input	ANY_NUM	OUT = IN1 / IN2

IN2	Input	ANY_NUM	
OUT	Output	ANY_NUM	

1.5.5 MOD (Остаток от деления)

Этот функциональный блок используется для деления значения IN1 на значение IN2 для получения остатка. Типы данных входных и выходных значений должны быть одинаковыми.

Имя	Ввод/Вывод	Тип данных	Описание
IN1	Input	ANY_INT	OUT = IN1 Modulo IN2
IN2	Input	ANY_INT	
OUT	Output	ANY_INT	

1.5.6 EXPT (Возведение в степень)

Имя	Ввод/Вывод	Тип данных	Описание
IN1	Input	ANY_REAL	OUT = IN1^IN2
IN2	Input	ANY_REAL	
OUT	Output	ANY_REAL	

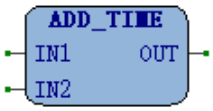
1.5.7 MOVE (Присвоение)

Этот функциональный блок используется для присвоения значения константы или переменной другой переменной.

Имя	Ввод/Вывод	Тип данных	Описание
IN	Input	ANY	OUT = IN
OUT	Output	ANY	

2.6 Операции со временем (Time)

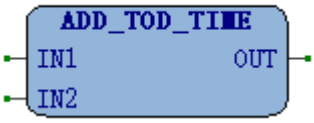
1.6.1 ADD_TIME (Сложение времени)



Имя	Ввод/Вывод	Тип данных	Описание
IN1	Input	TIME	
IN2	Input	TIME	

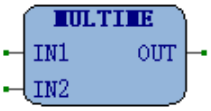
OUT	Output	TIME	

1.6.2 ADD_TOD_TIME (Сложение времени суток и времени)



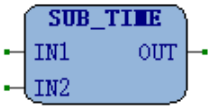
Имя	Ввод/Вывод	Тип данных	Описание
IN1	Input	TOD	
IN2	Input	TIME	
OUT	Output	TOD	

1.6.3 MUL_TIME (Умножение времени)



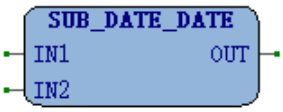
Имя	Ввод/Вывод	Тип данных	Описание
IN1	Input	TIME	
IN2	Input	ANY_NUM	
OUT	Output	TIME	

1.6.4 SUB_TIME (Вычитание времени)



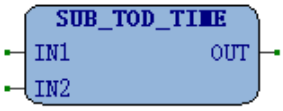
Имя	Ввод/Вывод	Тип данных	Описание
IN1	Input	TIME	
IN2	Input	TIME	
OUT	Output	TIME	

1.6.5 SUB_DATE_DATE (Вычитание дат)



Имя	Ввод/Вывод	Тип данных	Описание
IN1	Input	DATE	
IN2	Input	DATE	
OUT	Output	TIME	

1.6.6 SUB_TOD_TIME (Вычитание времени из времени суток)

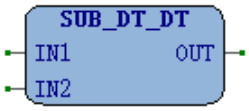


Имя	Ввод/Вывод	Тип данных	Описание
IN1	Input	TOD	
IN2	Input	TIME	
OUT	Output	TOD	

1.6.7 SUB_DT_TIME (Вычитание времени из даты-времени)

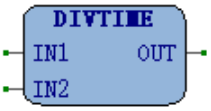
Имя	Ввод/Вывод	Тип данных	Описание
IN1	Input	DT	
IN2	Input	TIME	
OUT	Output	DT	

1.6.8 SUB_DT_DT (Вычитание дат-времен)



Имя	Ввод/Вывод	Тип данных	Описание
IN1	Input	DT	
IN2	Input	DT	
OUT	Output	TIME	

1.6.9 DIV_TIME (Деление времени)



Имя	Ввод/Вывод	Тип данных	Описание
IN1	Input	TIME	
IN2	Input	ANY_NUM	
OUT	Output	TIME	

2.7 Битовые сдвиги (Bit Shift)

Примеры:

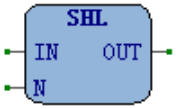
IN = 2#0001 1001 типа BYTE, N=3

SHL(IN,3)=2#1100 1000

SHR(IN,3)=2#0000 0011

ROL(IN,3)=2#1100 1000
ROR(IN,3)=2#0010 0011
ПРИМЕЧАНИЕ: IN типа Bool (один бит) не имеет смысла. Это ошибка, если значение входа N меньше нуля.

1.7.1 SHL (Сдвиг влево)



Имя	Ввод/Вывод	Тип данных	Описание
IN	Input	ANY_BIT	
N	Input	ANY_INT	Количество битов для сдвига
OUT	Output	ANY_BIT	OUT = IN сдвинутый влево на N битов, с заполнением нулями справа

1.7.2 SHR (Сдвиг вправо)

Имя	Ввод/Вывод	Тип данных	Описание
IN	Input	ANY_BIT	
N	Input	ANY_INT	Количество битов для сдвига
OUT	Output	ANY_BIT	OUT = IN сдвинутый вправо на N битов, с заполнением нулями слева

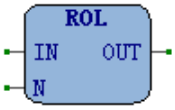
1.7.3 ROR (Циклический сдвиг вправо)

Этот функциональный блок используется для циклического сдвига операнда вправо по битам, при этом бит, выдвинутый справа, непосредственно добавляется к старшему биту слева.

Имя	Ввод/Вывод	Тип данных	Описание
IN	Input	ANY_NBIT	
N	Input	ANY_INT	Количество битов для сдвига
OUT	Output	ANY_NBIT	OUT = IN циклически сдвинутый вправо на N битов

1.7.4 ROL (Циклический сдвиг влево)

Этот функциональный блок используется для циклического сдвига IN влево по битам, при этом биты, выдвинутые слева, непосредственно добавляются к младшему биту справа.



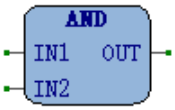
Имя	Ввод/Вывод	Тип данных	Описание
IN	Input	ANY_NBIT	
N	Input	ANY_INT	Количество битов для сдвига
OUT	Output	ANY_NBIT	OUT = IN циклически сдвинутый влево на N битов

2.8 Побитовые операции (Bit wise)

Инструкции AND, OR, XOR могут добавлять несколько входных контактов. Чтобы добавить их, щелкните правой кнопкой мыши на функциональном блоке в программе и выберите "Increase pins".

1.8.1 AND (Побитовое И)

Этот функциональный блок используется для операции переменных или констант, и типы входных и выходных переменных должны быть согласованы. Если все входы равны 1, то выход равен 1.



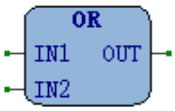
Имя	Ввод/Вывод	Тип данных	Описание
IN1	Input	ANY_BIT	
IN2	Input	ANY_BIT	
OUT	Output	ANY_BIT	

Таблица истинности AND:

IN1	IN2	OUT
0	0	0
0	1	0
1	0	0
1	1	1

1.8.2 OR (Побитовое ИЛИ)

Функциональные блоки используются для побитовой операции ИЛИ над переменными или константами, и типы входных и выходных данных должны быть согласованы. Когда хотя бы один вход равен 1, выход также равен 1. В противном случае выход равен 0.



Имя	Ввод/Вывод	Тип данных	Описание
IN1	Input	ANY_BIT	
IN2	Input	ANY_BIT	
OUT	Output	ANY_BIT	

Таблица истинности OR:

IN1	IN2	OUT
0	0	0
0	1	1
1	0	1
1	1	1

1.8.3 XOR (Исключающее ИЛИ)

Этот функциональный блок используется для операции XOR переменных или констант, и типы входных и выходных данных должны быть согласованы.
Когда только один вход равен 1, выходной бит также генерирует 1. Когда все входы равны 1 или 0, выход равен 0.

Имя	Ввод/Вывод	Тип данных	Описание
IN1	Input	ANY_BIT	
IN2	Input	ANY_BIT	
OUT	Output	ANY_BIT	

Таблица истинности XOR:

IN1	IN2	OUT
0	0	0
0	1	1
1	0	1
1	1	0

1.8.4 NOT (Побитовое НЕ)

Этот функциональный блок используется для двоичных операций НЕ над переменными или константами, где двоичные операции НЕ выполняются побитно, и типы входных и выходных данных должны быть согласованы.
Когда вход равен 0, выход равен 1. Когда вход равен 1, выход равен 0.

Имя	Ввод/Вывод	Тип данных	Описание
IN	Input	ANY_BIT	

OUT	Output	ANY_BIT	

Таблица истинности NOT:

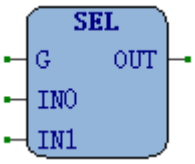
IN	OUT
0	1
1	0

2.9 Операции выбора (Selection)

Инструкции MAX, MIN и MUX могут добавлять несколько входных контактов путем щелчка правой кнопкой мыши на функциональном блоке в программе и выбора "Increase pins".

1.9.1 SEL (Выбор из двух)

Этот функциональный блок используется для выбора одной входной переменной из двух входных переменных в качестве выхода.



Имя	Ввод/Вывод	Тип данных	Описание
G	Input	BOOL	Выбор из 2 входов (управление)
IN0	Input	ANY	
IN1	Input	ANY	
OUT	Output	ANY	OUT = IN0 если G = FALSE; OUT = IN1 если G = TRUE

1.9.2 MAX (Максимум)

Этот функциональный блок используется для выбора максимального числа из значений двух или более входных переменных в качестве выхода.

Имя	Ввод/Вывод	Тип данных	Описание
IN1	Input	ANY	
IN2	Input	ANY	
OUT	Output	ANY	OUT = MAX(IN1, IN2)

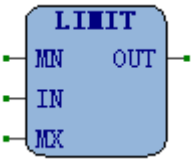
1.9.3 MIN (Минимум)

Этот функциональный блок используется для выбора минимального числа из значений двух или более входных переменных в качестве выхода.

Имя	Ввод/Вывод	Тип данных	Описание
IN1	Input	ANY	
IN2	Input	ANY	
OUT	Output	ANY	OUT = MIN(IN1, IN2)

1.9.4 LIMIT (Ограничение)

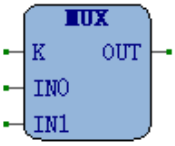
Этот функциональный блок используется для ограничения выхода значений входной переменной, то есть независимо от того, как изменяется входная переменная IN, выводятся только числовые значения между MN и MX.



Имя	Ввод/Вывод	Тип данных	Описание
MN	Input	ANY	Минимальное значение для ограничения
IN	Input	ANY	
MX	Input	ANY	Максимальное значение для ограничения
OUT	Output	ANY	OUT = MN если IN <= MN; OUT = MX если IN >= MX; OUT = IN в остальных случаях

1.9.5 MUX (Мультиплексор)

Этот функциональный блок используется для выбора переменной выхода из нескольких входных контактов.



Имя	Ввод/Вывод	Тип данных	Описание
K	Input	ANY_INT	Выбор из n входов
IN0	Input	ANY	
IN1	Input	ANY	
OUT	Output	ANY	OUT = MUX (K, IN0, IN1...,INn). Если K=0, OUT =IN0, Если K=n, OUT =INn

2.10 Операции сравнения (Comparison)

Инструкции GT, GE, EQ, LT, LE могут добавлять несколько входных контактов. Чтобы добавить их, щелкните правой кнопкой мыши на функциональном блоке в программе и выберите "Increase pins".

1.10.1 GT (Больше)

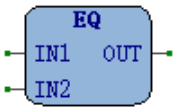
Это логический оператор. Когда первый операнд больше второго, возвращаемое значение равно TRUE.

Имя	Ввод/Вывод	Тип данных	Описание
IN1	Input	ANY	
IN2	Input	ANY	
OUT	Output	BOOL	Если IN1 > IN2>...> INn, OUT = TRUE, иначе OUT = FALSE.

1.10.2 GE (Больше или равно)

Имя	Ввод/Вывод	Тип данных	Описание
IN1	Input	ANY	
IN2	Input	ANY	
OUT	Output	BOOL	Если IN1 ≥ IN2 ≥...≥ INn, OUT = TRUE, иначе OUT = FALSE.

1.10.3 EQ (Равно)



Имя	Ввод/Вывод	Тип данных	Описание
IN1	Input	ANY	
IN2	Input	ANY	
OUT	Output	BOOL	Если IN1 = IN2=...= INn, OUT = TRUE, иначе OUT = FALSE.

1.10.4 LT (Меньше)

Имя	Ввод/Вывод	Тип данных	Описание
IN1	Input	ANY	

IN2	Input	ANY	
OUT	Output	BOOL	Если IN1 < IN2 <...< INn, OUT = TRUE, иначе OUT = FALSE.

1.10.5 LE (Меньше или равно)

Имя	Ввод/Вывод	Тип данных	Описание
IN1	Input	ANY	
IN2	Input	ANY	
OUT	Output	BOOL	Если IN1 ≤ IN2 ≤...≤ INn, OUT = TRUE, иначе OUT = FALSE.

1.10.6 NE (Не равно)

Имя	Ввод/Вывод	Тип данных	Описание
IN1	Input	ANY	
IN2	Input	ANY	
OUT	Output	BOOL	Если IN1 ≠ IN2 ≠...≠ INn, OUT = TRUE, иначе OUT = FALSE.

2.11 Операции со строками (Character String)

1.11.1 CONCAT (Конкатенация строк)

Объединяет две строки в одну новую выходную строку.

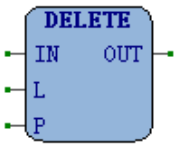
Имя	Ввод/Вывод	Тип данных	Описание
IN1	Input	STRING	Начало строки для объединения (например, "it")
IN2	Input	STRING	Конец строки для объединения (например, "em")
OUT	Output	STRING	Результирующая строка (например, "item")

1.11.2 CONCAT_DATE_TOD (Объединение даты и времени)

Имя	Ввод/Вывод	Тип данных	Описание
IN1	Input	DATE	
IN2	Input	TOD	
OUT	Output	DT	

1.11.3 DELETE (Удаление части строки)

Удаляет определенную непрерывную часть из строки, оставшиеся символы формируют выходную строку.



Имя	Ввод/Вывод	Тип данных	Описание
IN	Input	STRING	Исходная строка
L	Input	ANY_INT	Длина удаляемых символов
P	Input	ANY_INT	Позиция начала удаления, отсчет слева (включая эту позицию)
OUT	Output	STRING	Результирующая строка

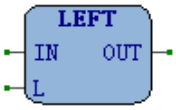
1.11.4 FIND (Поиск подстроки)

Находит начальную позицию определенной строки в строке.

Имя	Ввод/Вывод	Тип данных	Описание
IN1	Input	STRING	Исходная строка
IN2	Input	STRING	Искомая строка
OUT	Output	INT	Начальная позиция IN2 в IN1. Если IN2 не найдена в IN1, выводится '0'

1.11.5 LEFT (Левые символы)

Извлекает определенное количество символов с самого левого края строки направо, формируя новую выходную строку.



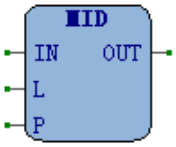
Имя	Ввод/Вывод	Тип данных	Описание
IN	Input	STRING	Входная строка (например, "character")
L	Input	ANY_INT	Количество извлекаемых символов слева (например, 5)
OUT	Output	STRING	Выходная строка (например, "chara")

1.11.6 LEN (Длина строки)

Имя	Ввод/Вывод	Тип данных	Описание
IN	Input	STRING	Входная строка (например, "character")
OUT	Output	INT	Длина строки (например, 9)

1.11.7 MID (Средние символы)

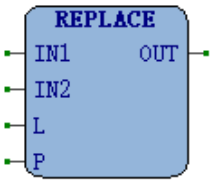
Извлекает определенное количество символов из середины строки для формирования новой выходной строки.



Имя	Ввод/Вывод	Тип данных	Описание
IN	Input	STRING	Входная строка (например, "character")
L	Input	ANY_INT	Количество извлекаемых символов (например, 5)
P	Input	ANY_INT	Позиция начала извлечения, отсчет слева (например, 3)
OUT	Output	STRING	Выходная строка (например, "racte")

1.11.8 REPLACE (Замена части строки)

Заменяет часть более длинной строки на другую строку, формируя новую выходную строку.



Имя	Ввод/Вывод	Тип данных	Описание
IN1	Input	STRING	Исходная строка (например, "character")
IN2	Input	STRING	Строка для замены (например, "AA")
L	Input	ANY_INT	Длина заменяемых символов (например, 3)
P	Input	ANY_INT	Позиция начала замены, отсчет слева (включая эту позицию) (например, 3)
OUT	Output	STRING	Результирующая строка (например, "chaAAter")

1.11.9 RIGHT (Правые символы)

Извлекает определенное количество символов с самого правого края строки налево, формируя новую выходную строку.

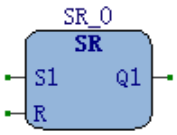
Имя	Ввод/Вывод	Тип данных	Описание
IN	Input	STRING	Входная строка (например, "character")

L	Input	ANY_INT	Количество извлекаемых символов (например, 5)
OUT	Output	STRING	Выходная строка (например, "acter")

2.12 Стандартные функциональные блоки (Standard Function Blocks)

2.12.1 SR (триггер с приоритетом Set)

Описание: Бистабильный триггер типа SR является элементом памяти (защелкой), в котором вход установки S1 имеет приоритет над входом сброса R.



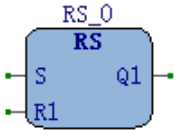
Имя	Ввод/Вывод	Тип данных	Описание
S1	Input	BOOL	Вход установки (Set). Активный уровень (TRUE) переводит выход Q1 в состояние 1 и поддерживает его.
R	Input	BOOL	Вход сброса (Reset). Активный уровень (TRUE) пытается сбросить выход Q1 в состояние 0.
Q1	Output	BOOL	Выходное состояние триггера.

Логика работы:

- **Приоритет Set:** Если S1 = TRUE, выход Q1 немедленно устанавливается в TRUE и удерживается в этом состоянии, **независимо** от состояния входа R.
- **Сброс:** Если S1 = FALSE и R = TRUE, выход Q1 сбрасывается в FALSE.
- **Удержание состояния:** Если оба входа S1 и R равны FALSE, выход Q1 сохраняет свое последнее состояние.

2.12.2 RS (триггер с приоритетом Reset)

Описание: Бистабильный триггер типа RS является элементом памяти (защелкой), в котором вход сброса R1 имеет приоритет над входом установки S.



Имя	Ввод/Вывод	Тип данных	Описание
S	Input	BOOL	Вход установки (Set). Активный уровень (TRUE) пытается установить выход Q1 в состояние 1.
R1	Input	BOOL	Вход сброса (Reset). Активный уровень (TRUE) переводит выход Q1 в состояние 0 и поддерживает его.
Q1	Output	BOOL	Выходное состояние триггера.

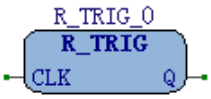
Логика работы:

- **Приоритет Reset:** Если R1 = TRUE, выход Q1 немедленно сбрасывается в FALSE и удерживается в этом состоянии, **независимо** от состояния входа S.

- **Установка:** Если R1 = FALSE и S = TRUE, выход Q1 устанавливается в TRUE.
- **Удержание состояния:** Если оба входа R1 и S равны FALSE, выход Q1 сохраняет свое последнее состояние.

2.12.3 R_TRIG (Детектор фронта 0->1)

Описание: Функциональный блок R_TRIG обнаруживает rising edge (положительный перепад, фронт 0→1) на входе CLK и генерирует на выходе Q одиночный импульс длительностью в один цикл выполнения ПЛК.



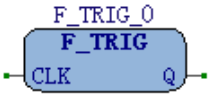
Имя	Ввод/Вывод	Тип данных	Описание
CLK	Input	BOOL	Входной сигнал, на котором детектируется rising edge (переход из состояния FALSE в состояние TRUE).
Q	Output	BOOL	Выходной импульс. Устанавливается в TRUE на один цикл выполнения ПЛК при обнаружении rising edge на входе CLK.

Логика работы:

- **Обнаружение фронта:** Выход Q устанавливается в TRUE только в том цикле выполнения ПЛК, когда вход CLK изменил свое состояние с FALSE на TRUE.
- **Длительность импульса:** Выход Q остается в состоянии TRUE ровно **один цикл** выполнения ПЛК, после чего автоматически сбрасывается в FALSE.
- **Статический сигнал:** Если вход CLK постоянно находится в состоянии FALSE или постоянно в состоянии TRUE, выход Q сохраняет состояние FALSE.

2.12.4 F_TRIG (Детектор фронта 1->0)

Описание: Функциональный блок F_TRIG обнаруживает falling edge (отрицательный перепад, спад 1→0) на входе CLK и генерирует на выходе Q одиночный импульс длительностью в один цикл выполнения ПЛК.



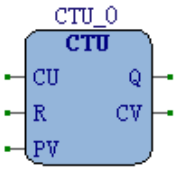
Имя	Ввод/Вывод	Тип данных	Описание
CLK	Input	BOOL	Входной сигнал, на котором детектируется falling edge (переход из состояния TRUE в состояние FALSE).
Q	Output	BOOL	Выходной импульс. Устанавливается в TRUE на один цикл выполнения ПЛК при обнаружении falling edge на входе CLK.

Логика работы:

- **Обнаружение спада:** Выход Q устанавливается в TRUE только в том цикле выполнения ПЛК, когда вход CLK изменил свое состояние с TRUE на FALSE.
- **Длительность импульса:** Выход Q остается в состоянии TRUE ровно **один цикл** выполнения ПЛК, после чего автоматически сбрасывается в FALSE.
- **Статический сигнал:** Если вход CLK постоянно находится в состоянии TRUE или постоянно в состоянии FALSE, выход Q сохраняет состояние FALSE.

2.12.5 CTU (Счетчик импульсов с предустановкой)

Описание: Счетчик CTU (Count Up) используется для подсчета положительных фронтов (переходов 0→1) на счетном входе CU. Выход Q сигнализирует о достижении или превышении счетчиком заданного значения PV.



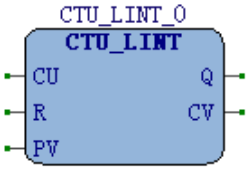
Имя	Ввод/Вывод	Тип данных	Описание
CU	Input	BOOL	Счетный вход (Count Up). Каждый положительный фронт (0→1) увеличивает текущее значение CV на 1.
R	Input	BOOL	Вход сброса (Reset). Активный уровень (TRUE) сбрасывает текущее значение CV в 0.
PV	Input	INT	Предустановленное значение (Preset Value). Задает значение, при достижении которого активируется выход Q.
Q	Output	BOOL	Флаг переполнения. Устанавливается в TRUE, когда CV >= PV. Сбрасывается, когда CV < PV.
CV	Output	INT	Текущее значение счетчика (Current Value).

Логика работы:

1. **Счет:** При каждом обнаружении положительного фронта (0→1) на входе CU текущее значение CV увеличивается на 1.
2. **Сброс:** Если вход R находится в состоянии TRUE, значение CV немедленно сбрасывается в 0. В этом состоянии счетчик не реагирует на фронты CU.
3. **Флаг переполнения (Q):**
   - Выход Q устанавливается в TRUE в том цикле, когда CV становится больше или равно PV.
   - Q сбрасывается в FALSE, когда CV становится меньше PV (например, после сброса).
4. **Переполнение:** При достижении максимального значения для типа INT счетчик останавливается.

2.12.6 CTU_LINT (Счетчик импульсов с предустановкой, 64-битный)

Описание: Счетчик CTU_LINT (Count Up, Long Integer) используется для подсчета положительных фронтов на счетном входе CU. Функционально идентичен блоку CTU, но использует 64-битный целочисленный тип данных (LINT) для значений PV и CV, что позволяет работать с очень большими числами.



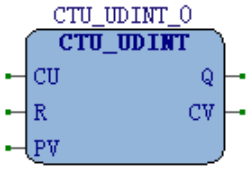
Имя	Ввод/Вывод	Тип данных	Описание
CU	Input	BOOL	Счетный вход (Count Up). Каждый положительный фронт (0→1) увеличивает текущее значение CV на 1.
R	Input	BOOL	Вход сброса (Reset). Активный уровень (TRUE) сбрасывает текущее значение CV в 0.
PV	Input	LINT	Предустановленное значение (Preset Value). Задает значение, при достижении которого активируется выход Q.
Q	Output	BOOL	Флаг переполнения. Устанавливается в TRUE, когда CV >= PV. Сбрасывается, когда CV < PV.
CV	Output	LINT	Текущее значение счетчика (Current Value).

Логика работы:

1. **Счет:** При каждом обнаружении положительного фронта (0→1) на входе CU текущее значение CV увеличивается на 1.
2. **Сброс:** Если вход R находится в состоянии TRUE, значение CV немедленно сбрасывается в 0. В этом состоянии счетчик не реагирует на фронты CU.
3. **Флаг переполнения (Q):**
   - Выход Q устанавливается в TRUE в том цикле, когда CV становится больше или равно PV.
   - Q сбрасывается в FALSE, когда CV становится меньше PV (например, после сброса).
4. **Диапазон значений:** Использование типа LINT позволяет работать с диапазоном значений от -9,223,372,036,854,775,808 до 9,223,372,036,854,775,807.

2.12.7 CTU_UDINT (Счетчик импульсов с предустановкой, 32-битный беззнаковый)

Описание: Счетчик CTU_UDINT (Count Up, Unsigned Double Integer) используется для подсчета положительных фронтов на счетном входе CU. Функционально идентичен блоку CTU, но использует 32-битный беззнаковый целочисленный тип данных (UDINT) для значений PV и CV, что позволяет работать с положительными числами в расширенном диапазоне.



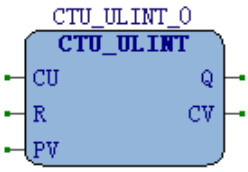
Имя	Ввод/Вывод	Тип данных	Описание
CU	Input	BOOL	Счетный вход (Count Up). Каждый положительный фронт (0→1) увеличивает текущее значение CV на 1.
R	Input	BOOL	Вход сброса (Reset). Активный уровень (TRUE) сбрасывает текущее значение CV в 0.
PV	Input	UDINT	Предустановленное значение (Preset Value). Задает значение, при достижении которого активируется выход Q.
Q	Output	BOOL	Флаг переполнения. Устанавливается в TRUE, когда CV >= PV. Сбрасывается, когда CV < PV.
CV	Output	UDINT	Текущее значение счетчика (Current Value).

Логика работы:

1. **Счет:** При каждом обнаружении положительного фронта (0→1) на входе CU текущее значение CV увеличивается на 1.
2. **Сброс:** Если вход R находится в состоянии TRUE, значение CV немедленно сбрасывается в 0. В этом состоянии счетчик не реагирует на фронты CU.
3. **Флаг переполнения (Q):**
   - Выход Q устанавливается в TRUE в том цикле, когда CV становится больше или равно PV.
   - Q сбрасывается в FALSE, когда CV становится меньше PV (например, после сброса).
4. **Диапазон значений:** Использование беззнакового типа UDINT позволяет работать с диапазоном значений от 0 до 4,294,967,295.

2.12.8 CTU_ULINT (Счетчик импульсов с предустановкой, 64-битный беззнаковый)

Описание: Счетчик CTU_ULINT (Count Up, Unsigned Long Integer) используется для подсчета положительных фронтов на счетном входе CU. Функционально идентичен блоку CTU, но использует 64-битный беззнаковый целочисленный тип данных (ULINT) для значений PV и CV, что позволяет работать с очень большими положительными числами.



Имя	Ввод/Вывод	Тип данных	Описание

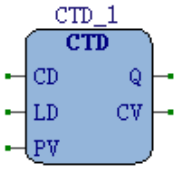
CU	Input	BOOL	Счетный вход (Count Up). Каждый положительный фронт (0→1) увеличивает текущее значение CV на 1.
R	Input	BOOL	Вход сброса (Reset). Активный уровень (TRUE) сбрасывает текущее значение CV в 0.
PV	Input	ULINT	Предустановленное значение (Preset Value). Задает значение, при достижении которого активируется выход Q.
Q	Output	BOOL	Флаг переполнения. Устанавливается в TRUE, когда CV >= PV. Сбрасывается, когда CV < PV.
CV	Output	ULINT	Текущее значение счетчика (Current Value).

Логика работы:

1. **Счет:** При каждом обнаружении положительного фронта (0→1) на входе CU текущее значение CV увеличивается на 1.
2. **Сброс:** Если вход R находится в состоянии TRUE, значение CV немедленно сбрасывается в 0. В этом состоянии счетчик не реагирует на фронты CU.
3. **Флаг переполнения (Q):**
   - Выход Q устанавливается в TRUE в том цикле, когда CV становится больше или равно PV.
   - Q сбрасывается в FALSE, когда CV становится меньше PV (например, после сброса).
4. **Диапазон значений:** Использование беззнакового типа ULINT позволяет работать с диапазоном значений от 0 до 18,446,744,073,709,551,615.

2.12.9 CTD (Счетчик обратного счета с предустановкой)

Описание: Счетчик CTD (Count Down) используется для обратного отсчета от предустановленного значения PV. Выход Q сигнализирует о достижении счетчиком нуля при декременте.



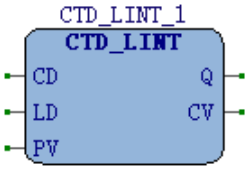
Имя	Ввод/Вывод	Тип данных	Описание
CD	Input	BOOL	Счетный вход (Count Down). Каждый положительный фронт (0→1) уменьшает текущее значение CV на 1.
LD	Input	BOOL	Вход загрузки (Load). Активный уровень (TRUE) устанавливает текущее значение CV равным PV.
PV	Input	INT	Предустановленное значение (Preset Value). Значение, в которое загружается счетчик при активации входа LD.
Q	Output	BOOL	Флаг нуля. Устанавливается в TRUE, когда CV <= 0.
CV	Output	INT	Текущее значение счетчика (Current Value).

Логика работы:

1. **Обратный счет:** При каждом обнаружении положительного фронта (0→1) на входе CD текущее значение CV уменьшается на 1.
2. **Загрузка:** Если вход LD находится в состоянии TRUE, значение CV устанавливается равным PV. В этом состоянии счетчик не реагирует на фронты CD.
3. **Флаг нуля (Q):**
   - Выход Q устанавливается в TRUE, когда CV становится меньше или равно 0.
   - Q сбрасывается в FALSE, когда CV больше 0 (например, после загрузки нового значения PV).
4. **Нижний предел:** При достижении нуля счетчик продолжает отсчет в отрицательную область (если позволяет тип данных).

2.12.10 CTD_LINT (Счетчик обратного счета с предустановкой, 64-битный)

Описание: Счетчик CTD_LINT (Count Down, Long Integer) используется для обратного отсчета от предустановленного значения PV. Функционально идентичен блоку CTD, но использует 64-битный целочисленный тип данных (LINT) для значений PV и CV, что позволяет работать с очень большими числами.



Имя	Ввод/Вывод	Тип данных	Описание
CD	Input	BOOL	Счетный вход (Count Down). Каждый положительный фронт (0→1) уменьшает текущее значение CV на 1.
LD	Input	BOOL	Вход загрузки (Load). Активный уровень (TRUE) устанавливает текущее значение CV равным PV.
PV	Input	LINT	Предустановленное значение (Preset Value). Значение, в которое загружается счетчик при активации входа LD.
Q	Output	BOOL	Флаг нуля. Устанавливается в TRUE, когда CV <= 0.
CV	Output	LINT	Текущее значение счетчика (Current Value).

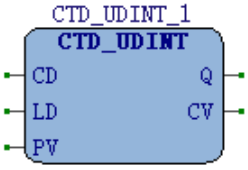
Логика работы:

1. **Обратный счет:** При каждом обнаружении положительного фронта (0→1) на входе CD текущее значение CV уменьшается на 1.
2. **Загрузка:** Если вход LD находится в состоянии TRUE, значение CV устанавливается равным PV. В этом состоянии счетчик не реагирует на фронты CD.
3. **Флаг нуля (Q):**
   - Выход Q устанавливается в TRUE, когда CV становится меньше или равно 0.
   - Q сбрасывается в FALSE, когда CV больше 0 (например, после загрузки нового значения PV).

4. **Диапазон значений:** Использование типа LINT позволяет работать с диапазоном значений от -9,223,372,036,854,775,808 до 9,223,372,036,854,775,807.

2.12.11 CTD_UDINT (Счетчик обратного счета с предустановкой, 32-битный беззнаковый)

Описание: Счетчик CTD_UDINT (Count Down, Unsigned Double Integer) используется для обратного отсчета от предустановленного значения PV. Функционально идентичен блоку CTD, но использует 32-битный беззнаковый целочисленный тип данных (UDINT) для значений PV и CV.



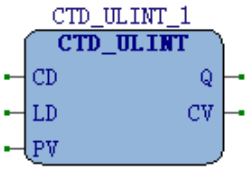
Имя	Ввод/Вывод	Тип данных	Описание
CD	Input	BOOL	Счетный вход (Count Down). Каждый положительный фронт (0→1) уменьшает текущее значение CV на 1.
LD	Input	BOOL	Вход загрузки (Load). Активный уровень (TRUE) устанавливает текущее значение CV равным PV.
PV	Input	UDINT	Предустановленное значение (Preset Value). Значение, в которое загружается счетчик при активации входа LD.
Q	Output	BOOL	Флаг нуля. Устанавливается в TRUE, когда CV = 0.
CV	Output	UDINT	Текущее значение счетчика (Current Value).

Логика работы:

1. **Обратный счет:** При каждом обнаружении положительного фронта (0→1) на входе CD текущее значение CV уменьшается на 1.
2. **Загрузка:** Если вход LD находится в состоянии TRUE, значение CV устанавливается равным PV. В этом состоянии счетчик не реагирует на фронты CD.
3. **Флаг нуля (Q):**
   - Выход Q устанавливается в TRUE, когда CV становится равным 0.
   - Q сбрасывается в FALSE, когда CV больше 0 (после загрузки нового значения PV).
4. **Особенность:** Поскольку тип данных UDINT беззнаковый, счетчик останавливается на значении 0 и не может уходить в отрицательный диапазон.

2.12.12 CTD_ULINT (Счетчик обратного счета с предустановкой, 64-битный беззнаковый)

Описание: Счетчик CTD_ULINT (Count Down, Unsigned Long Integer) используется для обратного отсчета от предустановленного значения PV. Функционально идентичен блоку CTD, но использует 64-битный беззнаковый целочисленный тип данных (ULINT) для значений PV и CV.



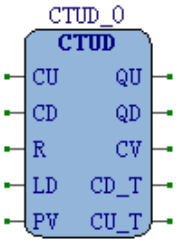
Имя	Ввод/Вывод	Тип данных	Описание
CD	Input	BOOL	Счетный вход (Count Down). Каждый положительный фронт (0→1) уменьшает текущее значение CV на 1.
LD	Input	BOOL	Вход загрузки (Load). Активный уровень (TRUE) устанавливает текущее значение CV равным PV.
PV	Input	ULINT	Предустановленное значение (Preset Value). Значение, в которое загружается счетчик при активации входа LD.
Q	Output	BOOL	Флаг нуля. Устанавливается в TRUE, когда CV = 0.
CV	Output	ULINT	Текущее значение счетчика (Current Value).

Логика работы:

1. **Обратный счет:** При каждом обнаружении положительного фронта (0→1) на входе CD текущее значение CV уменьшается на 1.
2. **Загрузка:** Если вход LD находится в состоянии TRUE, значение CV устанавливается равным PV. В этом состоянии счетчик не реагирует на фронты CD.
3. **Флаг нуля (Q):**
   - Выход Q устанавливается в TRUE, когда CV становится равным 0.
   - Q сбрасывается в FALSE, когда CV больше 0 (после загрузки нового значения PV).
4. **Особенности:**
   - **Беззнаковый тип:** Поскольку тип данных ULINT беззнаковый, счетчик останавливается на значении 0 и не может уходить в отрицательный диапазон.
   - **Диапазон значений:** Использование типа ULINT позволяет работать с диапазоном значений от 0 до 18,446,744,073,709,551,615.

2.12.13 CTUD (Реверсивный счетчик)

Описание: Реверсивный счетчик CTUD (Count Up/Down) имеет два счетных входа CU и CD и может выполнять как инкремент, так и декремент. Используется для подсчета в обоих направлениях с возможностью сброса и загрузки предустановленного значения.



Имя	Ввод/Вывод	Тип данных	Описание

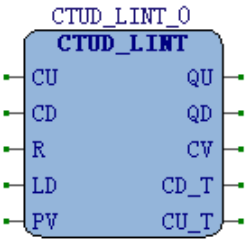
CU	Input	BOOL	Вход инкремента (Count Up). Положительный фронт (0→1) увеличивает CV на 1.
CD	Input	BOOL	Вход декремента (Count Down). Положительный фронт (0→1) уменьшает CV на 1.
R	Input	BOOL	Вход сброса (Reset). Активный уровень (TRUE) сбрасывает CV в 0.
LD	Input	BOOL	Вход загрузки (Load). Активный уровень (TRUE) устанавливает CV равным PV.
PV	Input	INT	Предустановленное значение (Preset Value).
QU	Output	BOOL	Флаг верхнего предела. Устанавливается в TRUE, когда CV >= PV.
QD	Output	BOOL	Флаг нижнего предела. Устанавливается в TRUE, когда CV <= 0.
CV	Output	INT	Текущее значение счетчика (Current Value).
CD_T	Output	R_TRIG	Выходной триггер для детектирования фронтов на входе CD.
CU_T	Output	R_TRIG	Выходной триггер для детектирования фронтов на входе CU.

Логика работы:

1. **Счет:**
   - Положительный фронт на CU увеличивает CV на 1
   - Положительный фронт на CD уменьшает CV на 1
2. **Управление:**
   - При R = TRUE значение CV сбрасывается в 0
   - При LD = TRUE значение CV устанавливается равным PV
3. **Флаги состояния:**
   - QU = TRUE, когда CV >= PV
   - QD = TRUE, когда CV <= 0
4. **Триггеры:** Выходы CD_T и CU_T позволяют детектировать фронты на соответствующих счетных входах для создания более сложной логики.

2.12.14 CTUD_LINT (Реверсивный счетчик, 64-битный)

Описание: Реверсивный счетчик CTUD_LINT (Count Up/Down, Long Integer) имеет два счетных входа CU и CD и может выполнять как инкремент, так и декремент. Функционально идентичен блоку CTUD, но использует 64-битный целочисленный тип данных (LINT) для значений PV и CV.



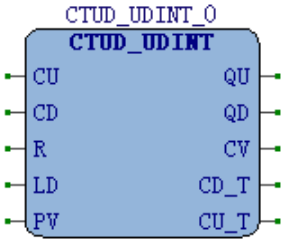
Имя	Ввод/Вывод	Тип данных	Описание
CU	Input	BOOL	Вход инкремента (Count Up). Положительный фронт (0→1) увеличивает CV на 1.
CD	Input	BOOL	Вход декремента (Count Down). Положительный фронт (0→1) уменьшает CV на 1.
R	Input	BOOL	Вход сброса (Reset). Активный уровень (TRUE) сбрасывает CV в 0.
LD	Input	BOOL	Вход загрузки (Load). Активный уровень (TRUE) устанавливает CV равным PV.
PV	Input	LINT	Предустановленное значение (Preset Value).
QU	Output	BOOL	Флаг верхнего предела. Устанавливается в TRUE, когда CV >= PV.
QD	Output	BOOL	Флаг нижнего предела. Устанавливается в TRUE, когда CV <= 0.
CV	Output	LINT	Текущее значение счетчика (Current Value).
CD_T	Output	R_TRIG	Выходной триггер для детектирования фронтов на входе CD.
CU_T	Output	R_TRIG	Выходной триггер для детектирования фронтов на входе CU.

Логика работы:

1. **Счет:**
   - Положительный фронт на CU увеличивает CV на 1
   - Положительный фронт на CD уменьшает CV на 1
2. **Управление:**
   - При R = TRUE значение CV сбрасывается в 0
   - При LD = TRUE значение CV устанавливается равным PV
3. **Флаги состояния:**
   - QU = TRUE, когда CV >= PV
   - QD = TRUE, когда CV <= 0
4. **Триггеры:** Выходы CD_T и CU_T позволяют детектировать фронты на соответствующих счетных входах.
5. **Диапазон значений:** Использование типа LINT позволяет работать с диапазоном значений от -9,223,372,036,854,775,808 до 9,223,372,036,854,775,807.

2.12.15 CTUD_UDINT (Реверсивный счетчик, 32-битный беззнаковый)

Описание: Реверсивный счетчик CTUD_UDINT (Count Up/Down, Unsigned Double Integer) имеет два счетных входа CU и CD и может выполнять как инкремент, так и декремент. Функционально идентичен блоку CTUD, но использует 32-битный беззнаковый целочисленный тип данных (UDINT) для значений PV и CV.



Имя	Ввод/Вывод	Тип данных	Описание
CU	Input	BOOL	Вход инкремента (Count Up). Положительный фронт (0→1) увеличивает CV на 1.
CD	Input	BOOL	Вход декремента (Count Down). Положительный фронт (0→1) уменьшает CV на 1.
R	Input	BOOL	Вход сброса (Reset). Активный уровень (TRUE) сбрасывает CV в 0.
LD	Input	BOOL	Вход загрузки (Load). Активный уровень (TRUE) устанавливает CV равным PV.
PV	Input	UDINT	Предустановленное значение (Preset Value).
QU	Output	BOOL	Флаг верхнего предела. Устанавливается в TRUE, когда CV >= PV.
QD	Output	BOOL	Флаг нижнего предела. Устанавливается в TRUE, когда CV = 0.
CV	Output	UDINT	Текущее значение счетчика (Current Value).
CD_T	Output	R_TRIG	Выходной триггер для детектирования фронтов на входе CD.
CU_T	Output	R_TRIG	Выходной триггер для детектирования фронтов на входе CU.

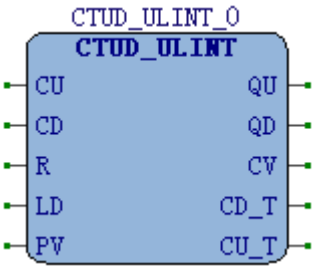
Логика работы:

1. **Счет:**
   - Положительный фронт на CU увеличивает CV на 1
   - Положительный фронт на CD уменьшает CV на 1
2. **Управление:**
   - При R = TRUE значение CV сбрасывается в 0
   - При LD = TRUE значение CV устанавливается равным PV
3. **Флаги состояния:**
   - QU = TRUE, когда CV >= PV
   - QD = TRUE, когда CV = 0 (не может быть меньше 0 из-за беззнакового типа)

4. **Триггеры:** Выходы CD_T и CU_T позволяют детектировать фронты на соответствующих счетных входах.
5. **Диапазон значений:** Использование типа UDINT позволяет работать с диапазоном значений от 0 до 4,294,967,295.

2.12.16 CTUD_ULINT (Реверсивный счетчик, 64-битный беззнаковый)

Описание: Реверсивный счетчик CTUD_ULINT (Count Up/Down, Unsigned Long Integer) имеет два счетных входа CU и CD и может выполнять как инкремент, так и декремент. Функционально идентичен блоку CTUD, но использует 64-битный беззнаковый целочисленный тип данных (ULINT) для значений PV и CV.



Имя	Ввод/Вывод	Тип данных	Описание
CU	Input	BOOL	Вход инкремента (Count Up). Положительный фронт (0→1) увеличивает CV на 1.
CD	Input	BOOL	Вход декремента (Count Down). Положительный фронт (0→1) уменьшает CV на 1.
R	Input	BOOL	Вход сброса (Reset). Активный уровень (TRUE) сбрасывает CV в 0.
LD	Input	BOOL	Вход загрузки (Load). Активный уровень (TRUE) устанавливает CV равным PV.
PV	Input	ULINT	Предустановленное значение (Preset Value).
QU	Output	BOOL	Флаг верхнего предела. Устанавливается в TRUE, когда CV >= PV.
QD	Output	BOOL	Флаг нижнего предела. Устанавливается в TRUE, когда CV = 0.
CV	Output	ULINT	Текущее значение счетчика (Current Value).
CD_T	Output	R_TRIG	Выходной триггер для детектирования фронтов на входе CD.
CU_T	Output	R_TRIG	Выходной триггер для детектирования фронтов на входе CU.

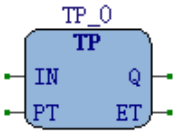
Логика работы:

1. **Счет:**
   - Положительный фронт на CU увеличивает CV на 1
   - Положительный фронт на CD уменьшает CV на 1 (до достижения 0)
2. **Управление:**

    - При R = TRUE значение CV сбрасывается в 0
    - При LD = TRUE значение CV устанавливается равным PV
3. **Флаги состояния:**
    - QU = TRUE, когда CV >= PV
    - QD = TRUE, когда CV = 0
4. **Особенности:**
    - **Беззнаковый тип:** Счетчик не может уходить в отрицательный диапазон, останавливается на 0
    - **Диапазон значений:** Использование типа ULINT позволяет работать с диапазоном от 0 до 18,446,744,073,709,551,615
    - **Триггеры:** Выходы CD_T и CU_T позволяют детектировать фронты на соответствующих счетных входах

2.12.17 TP (Импульсный таймер)

Описание: Импульсный таймер TP (Pulse Timer) генерирует на выходе Q импульс фиксированной длительности, заданной параметром PT, при обнаружении положительного фронта на входе IN.



Имя	Ввод/Вывод	Тип данных	Описание
IN	Input	BOOL	Запускающий вход. Положительный фронт (0→1) запускает генерацию импульса.
PT	Input	TIME	Длительность импульса (Pulse Time). Время, в течение которого выход Q остается в состоянии TRUE после запуска.
Q	Output	BOOL	Выход таймера. Активен (TRUE) во время генерации импульса.
ET	Output	TIME	Прошедшее время (Elapsed Time). Текущее время с момента запуска таймера.

Логика работы:

1. **Запуск:** При обнаружении положительного фронта (0→1) на входе IN:
    - Выход Q немедленно устанавливается в TRUE
    - Начинается отсчет времени ET
    - Таймер продолжает работу независимо от последующего состояния входа IN
2. **Генерация импульса:**
    - Выход Q остается в состоянии TRUE в течение всего времени PT
    - Когда ET достигает значения PT:
        - Выход Q сбрасывается в FALSE
        - Отсчет времени ET прекращается

3. **Повторный запуск:** Если новый положительный фронт на входе IN поступает, когда таймер уже активен:
    - Таймер перезапускается
    - Выход Q остается в состоянии TRUE
    - Отсчет времени ET начинается заново

Особенности:

- Длительность выходного импульса всегда точно равна значению PT
- Вход IN может возвращаться в FALSE в любой момент после запуска - это не влияет на работу таймера
- Таймер гарантирует минимальную длительность импульса PT при любых условиях запуска

2.12.18 TON (Таймер включения с задержкой)

Описание: Таймер TON (Timer ON Delay) используется для задержки установки выхода в состояние TRUE на фиксированный период после того, как вход становится активным.

Имя	Ввод/Вывод	Тип данных	Описание
IN	Input	BOOL	Вход разрешения таймера.
PT	Input	TIME	Уставка времени задержки (Preset Time).
Q	Output	BOOL	Выход таймера. Устанавливается в TRUE после истечения времени PT при условии, что вход IN остается активным.
ET	Output	TIME	Прошедшее время (Elapsed Time).

Логика работы:

1. **Сброс:** Когда вход IN = FALSE, таймер сбрасывается:
    - Q = FALSE
    - ET = 0
2. **Запуск:** При переходе IN из FALSE в TRUE начинается отсчет времени:
    - ET начинает увеличиваться от 0
3. **Срабатывание:** Когда ET >= PT:
    - Q устанавливается в TRUE
    - ET продолжает увеличиваться, пока IN = TRUE
4. **Прерывание:** Если IN возвращается в FALSE до того, как ET достигнет PT:
    - Таймер немедленно сбрасывается
    - Q остается в состоянии FALSE

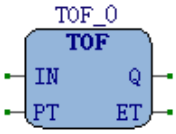
Особенности:

- Выход Q активируется только если вход IN непрерывно активен в течение всего времени PT
- Выход ET показывает текущее значение накопленного времени

- При достижении ET = PT таймер продолжает отсчет, но Q остается в состоянии TRUE

2.12.19 TOF (Таймер выключения с задержкой)

Описание: Таймер TOF (Turn-Off Delay) используется для задержки сброса выхода на фиксированный период после деактивации входного сигнала.



Имя	Ввод/Вывод	Тип данных	Описание
IN	Input	BOOL	Входной сигнал разрешения работы таймера.
PT	Input	TIME	Время задержки выключения (Preset Time).
Q	Output	BOOL	Выходное состояние таймера.
ET	Output	TIME	Прошедшее время (Elapsed Time).

Логика работы:

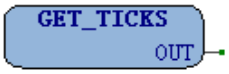
1. **Активное состояние:**
   - Когда IN = TRUE, выход Q немедленно устанавливается в TRUE
   - Время ET сбрасывается в 0
2. **Запуск задержки:**
   - При переходе IN из TRUE в FALSE начинается отсчет времени ET
   - Выход Q продолжает оставаться в состоянии TRUE
3. **Сброс выхода:**
   - Когда ET достигает значения PT, выход Q сбрасывается в FALSE
   - Таймер останавливается, ET сохраняет значение PT
4. **Прерывание задержки:**
   - Если во время отсчета ET вход IN снова активируется (TRUE), то:
   - Выход Q остается в состоянии TRUE
   - Время ET сбрасывается в 0
   - Отсчет прекращается

Применение: Используется для создания задержки выключения исполнительных механизмов, обеспечения времени на завершения процессов, управления временем останова оборудования.

2.13 Интеллектуальные функциональные блоки (Smart Function Blocks)

2.13.1 GET_TICKS

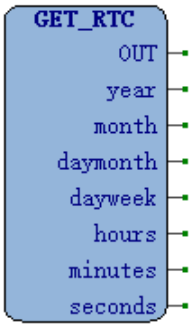
Возвращает количество миллисекунд, прошедших с момента включения ПЛК. Это число переполнится (обнулится) примерно через 50 дней.



Имя	Ввод/Вывод	Тип данных	Описание
OUT	Return	UDINT	Время в мс с момента включения

2.13.2 GET_RTC

Получает системное время. Функция возвращает 0 при успехе.



Имя	Ввод/Вывод	Тип данных	Описание
year	Output	UINT	Год [2000...2099]
month	Output	UINT	Месяц [1...12]
daymonth	Output	UINT	День месяца [1...31]
dayweek	Output	UINT	День недели [1...7]
hours	Output	UINT	Часы [0...23]
minutes	Output	UINT	Минуты [0...59]
seconds	Output	UINT	Секунды [0...59]
OUT	Return	DINT	Код возврата (0-успех)

2.13.3 SET_RTC

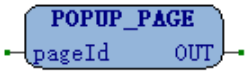
Устанавливает системное время. Функция возвращает 0 при успехе.

Имя	Ввод/Вывод	Тип данных	Описание
year	Input	UINT	Год [2000...2099]
month	Input	UINT	Месяц [1...12]
daymonth	Input	UINT	День месяца [1...31]
hours	Input	UINT	Часы [0...23]
minutes	Input	UINT	Минуты [0...59]
seconds	Input	UINT	Секунды [0...59]

OUT	Return	DINT	Код возврата (0-успех)

2.13.4 POPUP_PAGE

Отображает всплывающую страницу.



Имя	Ввод/Вывод	Тип данных	Описание
pageId	Input	UINT	ID отображаемой страницы
OUT	Return	DINT	Код возврата: 0-успех, 1-уже отображена, 2-несуществующий ID

2.13.5 GEN_RAND

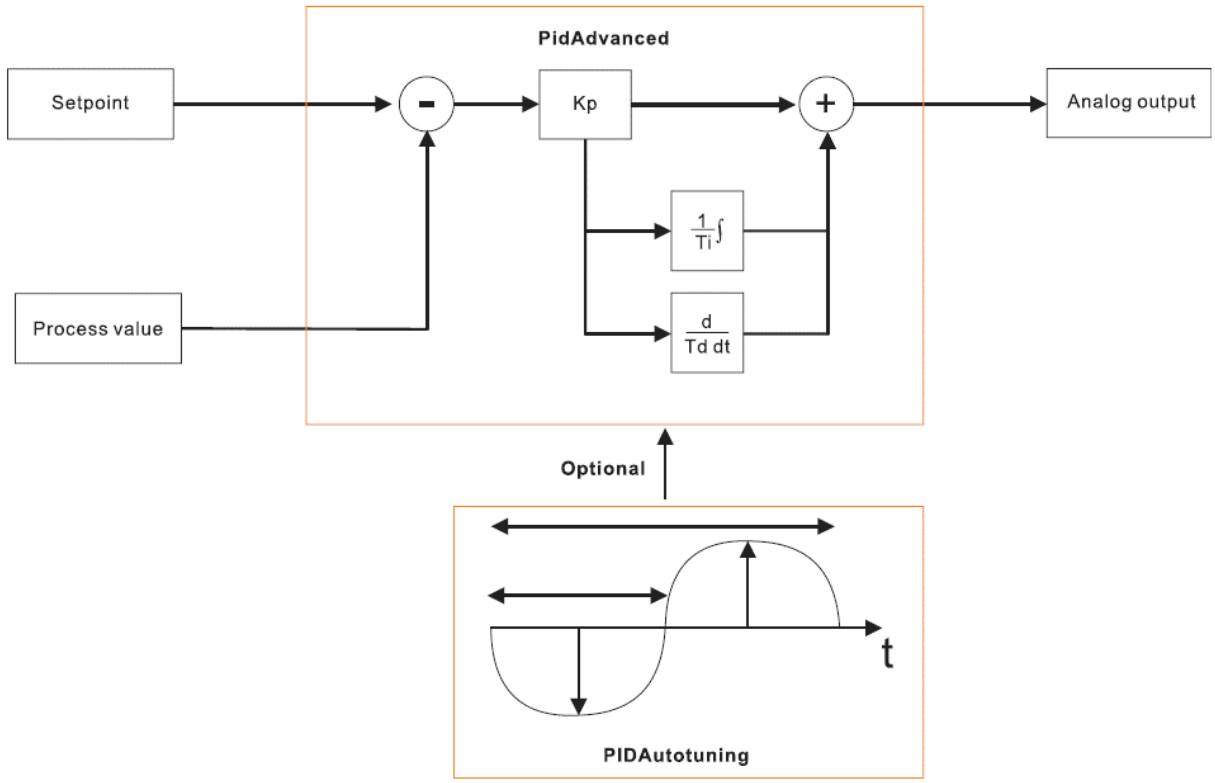
Генерирует 32-битное случайное число.

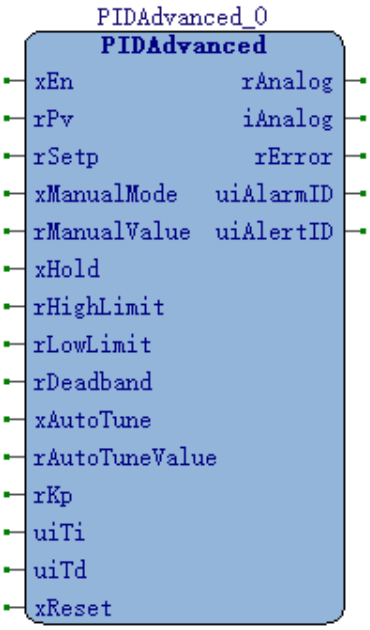
Имя	Ввод/Вывод	Тип данных	Описание
maxOut	Input	UDINT	Максимальное значение. 0 - без ограничения
OUT	Return	UDINT	Случайное число

2.13.6 PIDAdvanced

Описание: Функциональный блок PIDAdvanced предназначен для мониторинга и управления широким спектром технологических процессов. Он включает функции мертвой зоны (dead band), ручного режима и удержания (hold). Может использоваться в приложениях, требующих регулирования, таких как контроль температуры, давления и расхода. Блок реализует PID-алгоритм.

Примечание: Блок PIDAdvanced может быть расширен функцией автонастройки (autotuning).





Имя	Ввод/Вывод	Тип данных	Описание
xEn	Input	BOOL	**Разрешение работы блока.** TRUE: Включено
rPv	Input	REAL	**Текущее значение процесса (Process Value).** *Примечание: Значения rPv и rSetp должны иметь одинаковые единицы измерения.* Единицы: Пользовательские

rSetp	Input	REAL	**Заданное значение (уставка).** Единицы: Пользовательские
xManualMode	Input	BOOL	**Ручной режим.** TRUE: Включает ручной режим, выходы rAnalog и iAnalog принимают значение rManualValue. FALSE: Отключает ручной режим, расчет начинается с последнего введенного ручного значения rManualValue. По умолчанию: FALSE
rManualValue	Input	REAL	**Задание для ручного режима** для аналоговых выходов iAnalog и rAnalog. Единицы: %
xHold	Input	BOOL	**Удержание (Замораживание выхода).** TRUE: Замораживает выход PID и расчет интегральной составляющей. FALSE: Возобновляет работу PID. По умолчанию: FALSE *Примечание: При переходе в FALSE расчет начинается с последнего замороженного значения аналоговых выходов.*
rHighLimit	Input	REAL	**Верхний предел выхода PID.** По умолчанию: 100.0. Единицы: % *Пример: rHighLimit = 100.0 (%), rLowLimit = 0.0 (%): PID управляет выходом в диапазоне от 0.0 до 100.0 %.*
rLowLimit	Input	REAL	**Нижний предел выхода PID.** По умолчанию: 0.0. Единицы: %
rDeadband	Input	REAL	**Зона нечувствительности (мертвая зона)** для ошибки регулирования. Используется для стабилизации выхода PID и предотвращения "дребезга" при работе вблизи уставки. См пример ниже.
xAutoTune	Input	BOOL	**Автонастройка.** FALSE: Не активна. TRUE: Активна внешняя функция автонастройки. По умолчанию: FALSE
rAutoTuneValue	Input	REAL	**Значение для выхода rAnalog, определяемое внешней функцией автонастройки.** По умолчанию: 0.0. Единицы: %

rKp	Input	REAL	**Коэффициент усиления (пропорциональная составляющая).** Диапазон: -300.0 до 300.0. По умолчанию: 4.0
uiTi	Input	UINT	**Время интегрирования (интегральная составляющая).** Задается пользователем или определяется автонастройкой. По умолчанию: 0. Единицы: 0.1с
uiTd	Input	UINT	**Время дифференцирования (дифференциальная составляющая).** Задается пользователем. По умолчанию: 0. Единицы: 0.1с
xReset	Input	BOOL	**Сброс аварий.** FALSE: Не активен. TRUE: Сбрасывает аварии по фронту 0->1. *Примечание: Сбрасывает только сигнализацию аварии (Alarms). Для сброса аварии измените параметр, вызвавший ее, и подайте фронт на xReset. Предупреждения (Alerts) сбрасываются автоматически.*
rAnalog	Output	REAL	**Аналоговый выход.** Диапазон: от rLowLimit до rHighLimit (%)
iAnalog	Output	INT	**Целочисленный аналоговый выход.** Диапазон: 0 до 1000 (соответствует 0.0% до 100.0%)
rError	Output	REAL	**Ошибка регулирования (рассогласование)** между rSetp и rPv. Единицы: Пользовательские
uiAlarmID	Output	UINT	**Идентификатор аварии (Alarm).**
uiAlertID	Output	UINT	**Идентификатор предупреждения (Alert).**

Дополнительные пояснения:

Режимы работы:

- xManualMode = TRUE: Ручной режим. Выходы rAnalog и iAnalog равны rManualValue.
- xManualMode = FALSE: Автоматический режим. Расчет начинается с последнего значения rManualValue.

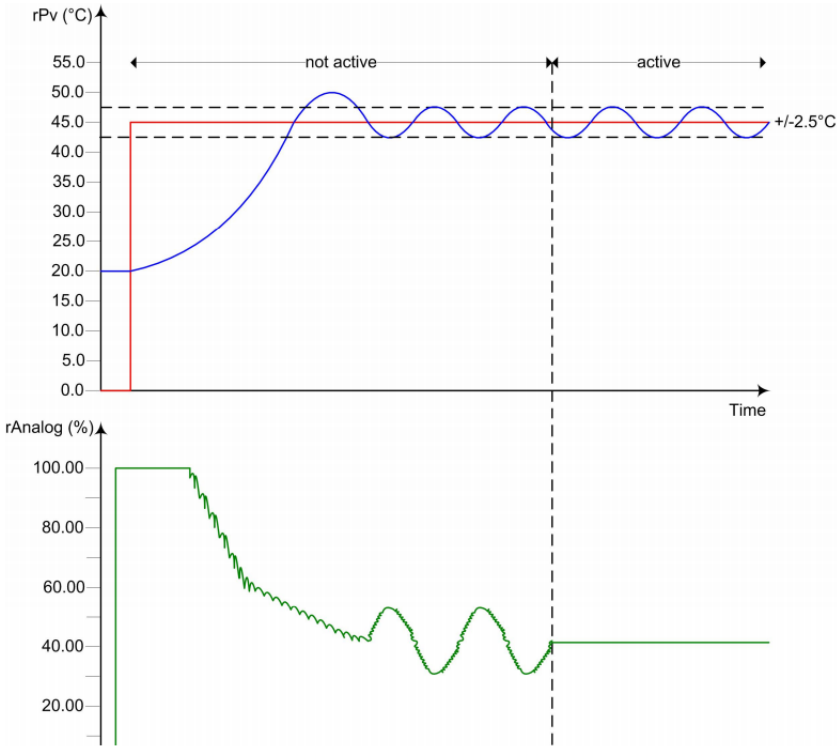
Пределы выхода:

- rHighLimit = 100.0, rLowLimit = 0.0: PID управляет выходом в диапазоне 0.0...100.0 %.

Мертвая зона (rDeadband): Значение определяет зону нечувствительности вокруг уставки, внутри которой выходной сигнал стабилизируется, чтобы избежать "дребезга".

Пример. Когда ошибка rSetp – rPv ≤ rDeadband, выходной сигнал стабилизируется.
rSetp = 45.0 °C (уставка)

rPv = 0...60.0 °C (диапазон датчика)
rHighLimit = 100.0 (%)
rLowLimit = 0.0 (%)
rDeadband = 2.5 °C (±2.5°C от уставки)
Зона нечувствительности от 42.5 °C до 47.5 °C.*



*При такой настройке PID-регулятор не будет реагировать на отклонения в пределах ±2.5°C от уставки 45.0°C, что обеспечивает плавную работу системы.

Принцип работы:

- Когда технологическая переменная (rPv) находится в пределах мертвой зоны (rSetp ± rDeadband), выходной сигнал блока (rAnalog/iAnalog) поддерживается постоянным
- Это предотвращает постоянные корректировки выхода при незначительных колебаниях процесса вокруг уставки
- Особенно полезно для уменьшения износа исполнительных механизмов (клапанов, нагревателей) в системах с шумом измерений

Коэффициент усиления (rKp):

- rKp > 0: Прямое управление (например, нагрев).
- Kp = 0: Выходы rAnalog и iAnalog не зависят от ошибки.
- Kp < 0: Обратное управление (например, охлаждение).

Время интегрирования (uiTi):

- uiTi = 1: Быстрое интегрирование, быстрое влияние на выход.

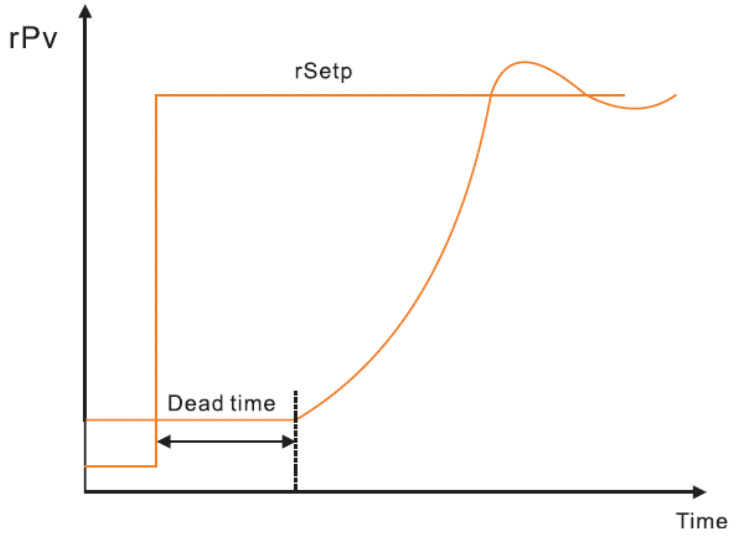
- uiTi = 10: В 10 раз медленнее, более плавное влияние на выход.
- uiTi = 0: Интегральная составляющая отключена.

Время дифференцирования (uiTd):

- uiTd = 1: Минимальное демпфирование, сильное влияние на выход.
- uiTd = 10: Демпфирование в 10 раз слабее, меньшее влияние на выход.
- uiTd = 0: Дифференциальная составляющая отключена.

Примечание: В системах с запаздыванием получения обратной связи (Dead Time) рекомендуется устанавливать uiTd = 0.

- Значение uiTd должно быть больше времени цикла задачи
- Если uiTd меньше времени цикла, его значение будет автоматически перезаписано значением времени цикла
- Это предотвращает неустойчивую работу системы из-за излишней чувствительности к шуму измерений



Идентификаторы аварий (uiAlarmID):

Бит	Причина аварии	Воздействие
0	Время цикла приложения превышает 2000 мс	Выходы rAnalog и iAnalog устанавливаются в 0.
1	Недопустимое значение параметров (rHighLimit, rLowLimit)	
2	Недопустимое значение параметра rDeadband	
3	Недопустимое значение параметров rKp, uiTi или uiTd	
4~15	Зарезервировано	

Идентификаторы предупреждений (uiAlertID):

Бит	Причина предупреждения	Воздействие
0	Недопустимое значение входа rManualValue	Выходы rAnalog и iAnalog устанавливаются в rHighLimit или rLowLimit

1	rLowLimit равен rHighLimit	Выходы rAnalog и iAnalog устанавливаются в rLowLimit
2	Активна автонастройка (xAutoTune = TRUE)	Выходы rAnalog и iAnalog устанавливаются в rAutoTuneValue
3	rKp установлен в 0.0	Выходы rAnalog и iAnalog устанавливаются в 0
4~15	Зарезервировано	

Диагностика и устранение неисправностей (Troubleshooting)

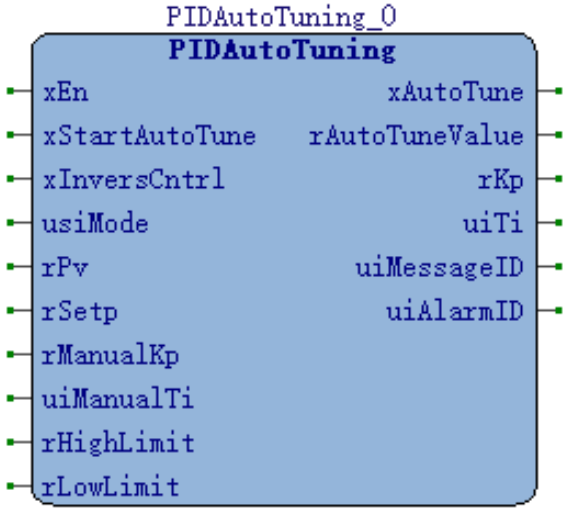
Авария/Предупреждение	**Проблема**	**Решение**
uiAlarmID.0	Время цикла приложения превышает 2000 мс.	Уменьшите объем и сложность приложения.
uiAlarmID.1	rLowLimit > rHighLimit, или rLowLimit > 100.0, или rLowLimit < 0.0, или rHighLimit > 100.0, или rHighLimit < 0.0.	Проверьте, что значения параметров находятся в своих допустимых диапазонах.
uiAlarmID.2	rDeadBand < 0.0, или rDeadBand > rHighLimit.	Проверьте, что значения параметров находятся в своих допустимых диапазонах.
uiAlarmID.3	rKp < -300.0, или rKp > +300.0, или uiTi < 0, или uiTi > 6000, или uiTd < 0, или uiTd > 6000.	Проверьте, что значения параметров находятся в своих допустимых диапазонах.
uiAlertID.0	rManualValue > rHighLimit, или rManualValue < rLowLimit.	Проверьте, что значения параметров находятся в своих допустимых диапазонах.
uiAlertID.1	rLowLimit равен rHighLimit.	Проверьте, что значения параметров находятся в своих допустимых диапазонах.
uiAlertID.2	Активна автонастройка (xAutoTune = TRUE).	Дождитесь завершения процесса автонастройки.
uiAlertID.3	rKp установлен в 0.0.	Проверьте, что значения параметров находятся в своих допустимых диапазонах.

2.13.7 PIDAutoTuning

Описание: Функциональный блок PIDAutoTuning измеряет динамическую отклика системы управления и автоматически рассчитывает параметры uiTi (время интегрирования в секундах) и rKp (коэффициент усиления) для этой системы. Эти параметры могут быть подключены к соответствующим входам PID-блока (например, PIDAdvanced).

Примечание 1: Этот блок должен использоваться совместно с PIDAdvanced. При включении автонастройки он вызывает колебания процесса вокруг заданного значения. После завершения трех колебаний автонастройка вычисляет набор PI-параметров.

Примечание 2: Во время процесса автонастройки система выводится на минимальный и максимальный рабочие пределы для измерения времени отклика процесса. Убедитесь, что минимальный и максимальный рабочие пределы заданы корректно.



Имя	Ввод/Вывод	Тип данных	Описание
xEn	Input	BOOL	**Разрешение работы блока.**
xStartAutoTune	Input	BOOL	**Запуск/Остановка автонастройки.** TRUE: Запускает автонастройку по фронту 0->1, FALSE: отключает автонастройку.
xInversCntrl	Input	BOOL	**Инверсия управления.** Например: TRUE: Охлаждение, FALSE: Нагрев.
usiMode	Input	USINT	**Режим определения параметров.** 0 = Ручные параметры 1 = Параметры ATR (медленно) 2 = Параметры ATR (средне) 3 = Параметры ATR (быстро) **Примечание:** Если автонастройка ранее не выполнялась, установите usiMode = 0 (ручные параметры).

rPv	Input	REAL	**Текущее значение процесса (Process Value).** Единицы: Пользовательские
rSetp	Input	REAL	**Заданное значение (уставка).** Единицы: Пользовательские *Примечание: rSetp — это рабочее значение уставки для PID-блока.*
rManualKp	Input	REAL	**Ручной коэффициент усиления.** Диапазон: -300.0 до 300.0 *Примечание: Активен, когда usiMode = 0.*
uiManualTi	Input	UINT	**Ручное время интегрирования.** Диапазон: 0 до 6000 (x0.1 с) *Примечание: Активно, когда usiMode = 0.*
rHighLimit	Input	REAL	**Верхний предел выхода PID.** Диапазон: 0.0 до 100.0 %
rLowLimit	Input	REAL	**Нижний предел выхода PID.** Диапазон: 0.0 до 100.0 %
xAutoTune	Output	BOOL	**Статус автонастройки.** TRUE: Процесс автонастройки активен. FALSE: Не активен. *Примечание: Переход xAutoTune из TRUE в FALSE (фронт 1->0) означает завершение автонастройки.*
rAutoTuneValue	Output	REAL	**Выходное значение автонастройки.** Диапазон: от rLowLimit до rHighLimit (%)
rKp	Output	REAL	**Ручной или рассчитанный коэффициент усиления.** Диапазон: -300.0 до 300.0
uiTi	Output	UINT	**Ручное или рассчитанное время интегрирования.** Диапазон: 0 до 6000 (x0.1 с)
uiMessageID	Output	UINT	**Идентификатор сообщения.**
uiAlarmID	Output	UINT	**Идентификатор аварии.**

Дополнительные пояснения:

- **rSetp, rPv:** Колебания сигнала могут привести к некорректному расчету параметров. rSetp и rPv должны быть в одинаковых единицах измерения.
- **xInversCntrl:**
  - Если для параметра xInversCntrl задано значение TRUE, то входное значение xInversCntrl инвертирует вычисление.
  - Если необходимо, чтобы при rAutoTuneValue = 100.0% значение rPv увеличивалось от меньшего к большему - установите xInversCntrl = FALSE (Прямое управление).

    - Если необходимо, чтобы при rAutoTuneValue = 100.0% значение rPv уменьшалось от большего к меньшему - установите xInversCntrl = TRUE (Обратное управление).
- **rLowLimit, rHighLimit:** Эти параметры определяют диапазон управления (rAutoTuneValue). PID-регулятор (например, PIDAdvanced) должен иметь те же значения. Максимальный диапазон: 0.0 до 100.0%.
    - *Примечание: В заведомо мощных (oversized) или слабых (undersized) системах может потребоваться ограничить диапазон выхода rAutoTuneValue (например, 0.0 до 50.0% для мощной системы).*
- **usiMode:**
    - 0 = manual: На выходы rKp и uiTi передаются ручные параметры.
    - 1 = slow: На выходы передаются рассчитанные PI-параметры (медленный отклик).
    - 2 = medium: На выходы передаются рассчитанные PI-параметры (средний отклик).
    - 3 = fast: На выходы передаются рассчитанные PI-параметры (быстрый отклик).

Идентификаторы сообщений (uiMessageID):

Бит	Причина сообщения	Значение
0	Настройка выполняется.	Процесс автонастройки в процессе.
1	Настройка завершена.	Автонастройка завершена успешно.
2	Настройка завершена. Система слишком мощная (oversized).	Автонастройка завершена, рассчитанные параметры указывают на избыточную мощность системы. Можно использовать как рассчитанные, так и ручные параметры. *Примечание: Диапазон выхода rAutoTuneValue можно ограничить с помощью rHighLimit.*
3	Настройка завершена. Система слишком слабая (undersized).	Автонастройка завершена, рассчитанные параметры указывают на недостаточную мощность системы. Можно использовать как рассчитанные, так и ручные параметры.
4~15	Зарезервировано	

Идентификаторы аварий (uiAlarmID):

Бит	Причина аварии	Воздействие
0	Недопустимое время цикла.	
1	Недопустимое значение параметров rHighLimit или rLowLimit.	rAutoTuneValue устанавливается в 0, автонастройка не запускается или прерывается.
2	Недопустимое значение параметров rManualKp или uiManualTi.	

3	Недопустимое значение параметра usiMode.	
4	Параметр xInversCntrl изменен во время выполнения автонастройки.	
5	PI-параметры не могут быть рассчитаны для данной системы.	rAutoTuneValue устанавливается в 0, автонастройка не запускается или прерывается. Ранее рассчитанные параметры (rKp, uiTi) удаляются.
6~15	Зарезервировано	

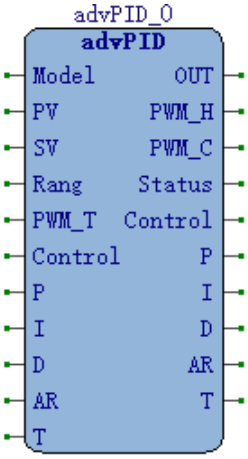
Диагностика и устранение неисправностей (Troubleshooting)

Авария	Проблема	Решение
uiAlarmID.0	Время цикла > 2000 мс.	Приложение слишком объемное. Уменьшите приложение и проверьте время цикла.
uiAlarmID.1	rHighLimit > rLowLimit, rHighLimit > 100.0%, rLowLimit = rHighLimit.	Проверьте, что значения параметров находятся в своих допустимых диапазонах.
uiAlarmID.2	rManualKp < -300.0, rManualKp > +300.0, uiManualTi < 0, uiManualTi > 6000.	Проверьте, что значения параметров находятся в своих допустимых диапазонах.
uiAlarmID.3	Недопустимое значение параметра usiMode.	Проверьте, что значения параметров находятся в своих допустимых диапазонах.
uiAlarmID.4	Изменения xInversCntrl во время автонастройки.	Не изменяйте значение, пока автонастройка выполняется.
uiAlarmID.5	Автонастройка выполнялась долго (>45 минут). Система медленная.	Остановите автонастройку. Установите PI-параметры вручную.
uiAlarmID.5	Внутренние переменные обнаружили некорректные значения. Сигнал текущего значения процесса колеблется.	Если параметры не работают, установите PI-параметры вручную (usiMode = 0).

2.13.8 advPID

Описание: Блок advPID реализует расширенные PID-алгоритмы для управления температурой, влажностью, расходом, давлением, положением, скоростью и другими параметрами. Включает функции автонастройки, онлайн-коррекции PID-параметров (алгоритм CT-TUNE) и предустановленных моделей PID. Пользователям не требуется сложное программирование — достаточно задать несколько простых параметров.

Примечание: При использовании этой библиотеки необходимо установить период цикла задачи для AdvPID равным 10 мс.



Имя	Вход / Выход	Тип данных	Описание
Model	Input	INT	**Выбор режима PID-алгоритма.** • **0**: Режим по умолчанию, стандартное PID-управление, для объектов с фиксированной моделью. • **1**: Стандартный PID-алгоритм с онлайн-коррекцией параметров. В этом режиме алгоритм CT-TUNE выполняет коррекцию PID-параметров в реальном времени на основе эффекта управления. Для объектов с часто меняющейся моделью. • **2**: Стандартный PID-алгоритм, режим управления движением. Базовое время устанавливается в 1 мс, дифференциальное действие отключается. По умолчанию работает с циклом 10 мс. Для контуров управления натяжением, положением, скоростью. • **3**: Режим инкрементального PI-управления. Используется инкрементальный алгоритм, дифференциальное действие отключено. Для исполнительных механизмов с обратной связью или объектов, где недопустимы резкие изменения выхода. • **4**: Режим инкрементального PID-управления. *Примечание: При переключении режимов соответствующие параметры контрольного слова будут изменены, а состояние работы сброшено. Сбросьте бит запуска контрольного слова во время работы.*
PV	Input	INT	**Текущее значение процесса (Process Variable).** Например, температура, давление, расход. Младший разряд

			представляет десятичную дробь (например, 4000 = 400.0). Диапазон: -32000...32000
SV	Input	INT	**Заданное значение (Set Value).** Ожидаемое значение для управляемого объекта. Младший разряд представляет десятичную дробь (например, 4000 = 400.0). Диапазон: -32000...32000
Rang	Input	INT	**Резерв.**
PWM_T	Input	INT	**Задание периода ШИМ-выхода.** Эффективный диапазон: 0.1...120.0 секунд. *Примечание: Для уменьшения износа выходных компонентов (например, механических реле) рекомендуется устанавливать значение не менее 5 секунд. Время PWM_T должно быть больше или равно параметру T.* Диапазон: 1...1200
OUT	Output	INT	**Выходное значение PID.** Диапазон определяется комбинацией битов 2 и 3 контрольного слова: • Режим нагрева: 0...32000 • Режим охлаждения: -32000...0 • Совместный режим: -32000...32000
PWM_H	Output	BOOL	**Выход ШИМ для нагрева (горячий выход).** 1 = ВКЛ, 0 = ВЫКЛ.
PWM_C	Output	BOOL	**Выход ШИМ для охлаждения (холодный выход).** 1 = ВКЛ, 0 = ВЫКЛ.
Status	Output	WORD	**Слово состояния текущего PID.** Состояния представлены соответствующими битами. Неперечисленные биты зарезервированы.
Control	InOut	WORD	**Слово управления PID.** Управляющая настройка алгоритма PID выбирает функции на основе 16 бит слова.
P	InOut	INT	**Зона пропорциональности (Proportional Band).** Младший разряд — десятичная дробь. Диапазон: 0...32000
I	InOut	INT	**Время интегрирования (Integral Time).** Младший разряд — десятичная дробь, макс. 3200.0 секунд. *Примечание: При значении 0 интегральная функция отключается.* Диапазон: 0...32000
D	InOut	INT	**Время дифференцирования (Differential Time).** Младший разряд — десятичная дробь, макс. 3200.0 секунд. *Примечание: При значении 0 дифференциальное действие отключается.* Диапазон: 0...32000
AR	InOut	INT	**Уровень производительности параметра автонастройки.** Диапазон: 0...2
T	InOut	INT	**Время цикла расчета PID.** Эффективный диапазон: 0.001...10.0 секунд.

			Примечание: Базовое значение определяется битом 13 контрольного слова. Диапазон: 1...1000

Слово состояния (Status Word) - Output

Текущее значение статуса текущего PID представлено соответствующими битами для указания статуса, а биты состояния, не указанные в списке, являются зарезервированными битами.

Бит состояния	**Описание**
Bit 0	1: PID работает; 0: PID остановлен
Bit 1	1: Выполняется автонастройка PID; 0: Автонастройка не выполняется
Bit 2	1: Выполняется адаптация PID; 0: Адаптация не выполняется
Bit 3	1: Выход ШИМ нагрева (PWM_H) ВКЛ; 0: ВЫКЛ
Bit 4	1: Выход ШИМ охлаждения (PWM_C) ВКЛ; 0: ВЫКЛ
Bit 14	1: Ошибка расчета PID (обычно из-за выхода PV за пределы); 0: Нормальное состояние
Bit 15	1: Сбой автонастройки PID; 0: Нормальное состояние

Контрольное слово (Control Word) - InOut

Управляющая настройка алгоритма PID выбирает функции на основе 16 бит слова.

Примечания:

1. *При запуске PID необходимо одновременно выбрать нагрев, охлаждение или совместное управление, иначе PID не запустится нормально.*
2. *Bit2 и Bit3 управляются совместно. Для совместного управления нагревом и охлаждением установите Bit2 и Bit3 в 1 одновременно. В этом случае диапазон выхода OUT составляет -32000...32000, и активны оба выхода PWM_H и PWM_C.*
3. *Неперечисленные биты зарезервированы.*

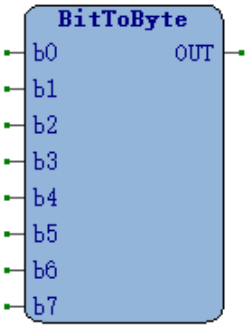
Бит управления	**Описание**
Bit 0	1: Запустить расчет PID; 0: Остановить PID, выход = 0
Bit 1	1: Запустить алгоритм автонастройки PID (автоматически сбрасывается после завершения); 0: Отключить автонастройку
Bit 2	1: Выход — одностороннее управление нагревом. ШИМ активен только на выходе нагрева (PWM_H). Диапазон OUT: 0...32000; 0: Не выбрано
Bit 3	1: Выход — одностороннее управление охлаждением. ШИМ активен только на выходе охлаждения (PWM_C). Диапазон OUT: -32000...0; 0: Не выбрано
Bit 4	1: Отключить функцию точек; 0: Не выбрано
Bit 5	1: Отключить дифференциальное действие; 0: Не выбрано
Bit 6	1: Активировать алгоритм онлайн-адаптации PID Expert (CT-TUNE); 0: Не выбрано

Bit 13	1: Базовое время цикла расчета PID = 1 мс; 0: Базовое время цикла расчета PID = 100 мс
Bit 14	1: Алгоритм онлайн-адаптации CT-TUNE инициирует расширение полосы пропускания; 0: Не выбрано
Bit 15	1: Сброс PID-параметров; 0: Нет

2.14 Функции преобразования битов (Bit Conversion Functions)

1.14.1 BitToByte

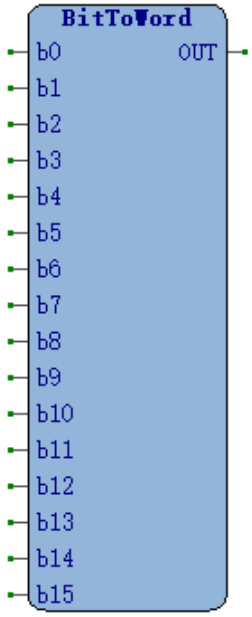
Формирует байт из 8 битов.



Имя	Ввод/Вывод	Тип данных	Описание
b0-b7	Input	BOOL	Биты 0-7
OUT	Return	BYTE	Результирующий байт

1.14.2 BitToWord

Формирует слово из 16 битов.

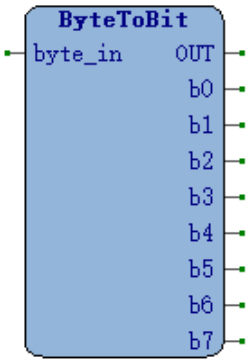


Имя	Ввод/Вывод	Тип данных	Описание

b0-b15	Input	BOOL	Биты 0-15
OUT	Return	WORD	Результирующее слово

1.14.3 ByteToBit

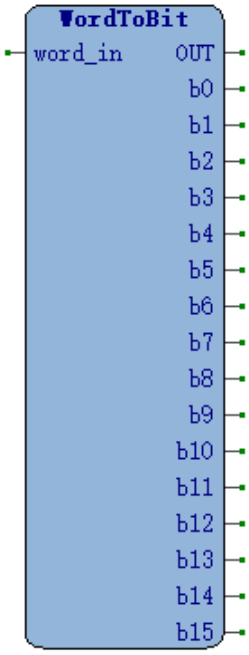
Разбивает байт на 8 битов.



Имя	Ввод/Вывод	Тип данных	Описание
byte_in	Input	BYTE	Входной байт
b0-b7	Output	BOOL	Биты 0-7
OUT	Return	BYTE	Входной байт

1.14.4 WordToBit

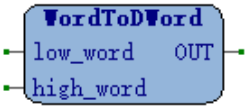
Разбивает слово на 16 битов.



Имя	Ввод/Вывод	Тип данных	Описание
word_in	Input	WORD	Входное слово
b0-b15	Output	BOOL	Биты 0-15
OUT	Return	WORD	Входное слово

1.14.5 WordToDWord

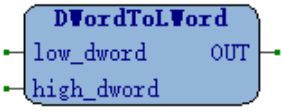
Объединяет два 16-битных слова в одно 32-битное двойное слово.



Имя	Ввод/Вывод	Тип данных	Описание
low_word	Input	WORD	Младшее слово
high_word	Input	WORD	Старшее слово
OUT	Return	DWORD	Результирующее двойное слово

1.14.6 DWordToLWord (Объединение двойных слов в длинное слово)

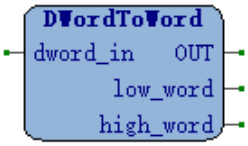
Объединяет два 32-битных двойных слова (DWORD) в одно 64-битное длинное слово (LWORD).



Имя	Ввод/Вывод	Тип данных	Описание
low_dword	Input	DWORD	Младшее двойное слово (биты 0-31)
high_dword	Input	DWORD	Старшее двойное слово (биты 32-63)
OUT	Return	LWORD	Результирующее длинное слово

1.14.7 DWordToWord (Разделение двойного слова на слова)

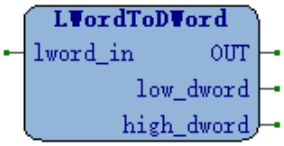
Разделяет 32-битное двойное слово (DWORD) на два 16-битных слова (WORD).



Имя	Ввод/Вывод	Тип данных	Описание
dword_in	Input	DWORD	Входное двойное слово
low_word	Output	WORD	Младшее слово (биты 0-15)
high_word	Output	WORD	Старшее слово (биты 16-31)
OUT	Return	DWORD	Входное двойное слово

1.14.8 LWordToDWord (Разделение длинного слова на двойные слова)

Разделяет 64-битное длинное слово (LWORD) на два 32-битных двойных слова (DWORD).



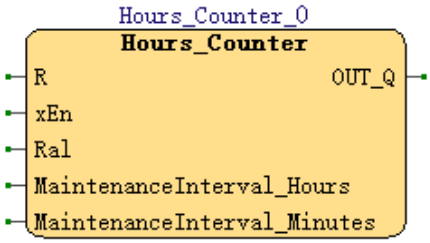
Имя	Ввод/Вывод	Тип данных	Описание
lword_in	Input	LWORD	Входное длинное слово

low_dword	Output	DWORD	Младшее двойное слово (биты 0-31)
high_dword	Output	DWORD	Старшее двойное слово (биты 32-63)
OUT	Return	LWORD	Входное длинное слово

2.15 Специальные функции (Special functions)

2.15.1 Hours_Counter (Счетчик моточасов)

Сконфигурированное время активируется сигналом на мониторинговом входе xEn. Выход OUT_Q устанавливается, когда это время истекло.

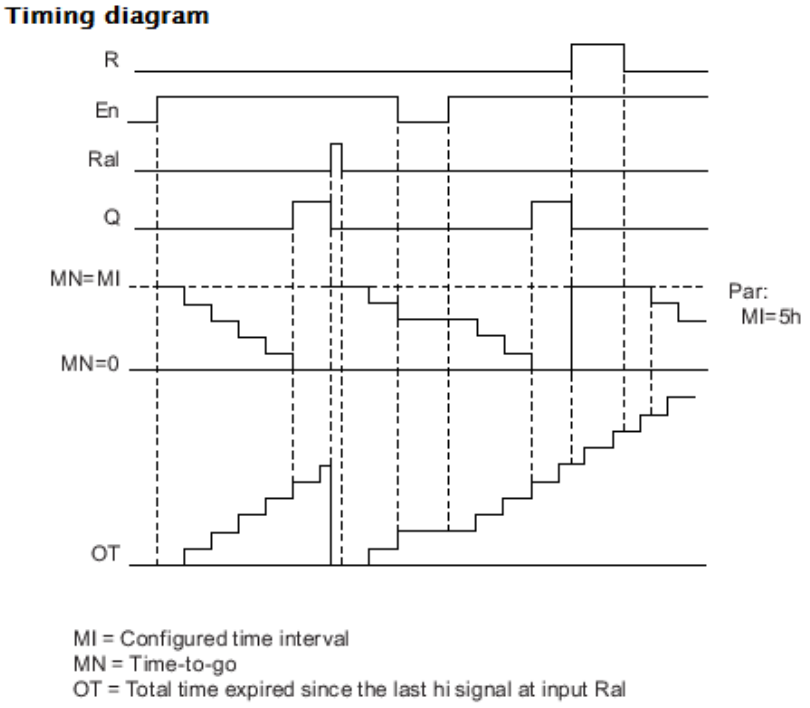


Имя	**Ввод/Вывод**	**Тип данных**	**Описание**
R	IN	BOOL	Положительный фронт (переход 0→1) на входе R сбрасывает выход Q и устанавливает настроенное значение MI в счетчике для длительности оставшегося времени (MN).
xEn	IN	BOOL	En — это мониторинговый вход. Устройство сканирует время включения (On Time) этого входа.
Ral	IN	BOOL	Положительный фронт на входе Ral (Reset all — Сбросить все) сбрасывает счетчик моточасов (OT) и выход, а также устанавливает значение оставшегося времени (MN) равным настроенному интервалу обслуживания (MI): Выход Q = 0; Измеренные рабочие часы OT = 0; Оставшееся время до обслуживания MN = MI.
MaintenanceInterval_Hours	IN	UINT	Интервал обслуживания, задаваемый в единицах часов и минут.

MaintenanceInterval_Minutes	IN	UINT	Интервал обслуживания, задаваемый в единицах часов и минут.
OUT_Q	OUT	BOOL	Выходной сигнал.

Счетчик моточасов Hours_Counter отслеживает вход En. Пока статус на этом входе равен 1, SystemePLC вычисляет истекшее время и оставшееся время MN. SystemePLC отображает эти времена, когда установлен в режим конфигурации. Выход OUT_Q устанавливается в 1, когда оставшееся время MN становится равным нулю.
Вы можете сбросить выход OUT_Q и счетчик оставшегося времени MN до указанного значения MI с помощью сигнала на входе R. Счетчик рабочих часов OT при этом не затрагивается.
Вы можете сбросить выход OUT_Q и счетчик оставшегося времени MN до указанного значения MI с помощью сигнала на входе Ral. Счетчик рабочих часов OT сбрасывается в 0.
В зависимости от конфигурации параметра Q, выход сбрасывается либо сигналом сброса на входе R или Ral ("Q → R"), либо когда сигнал сброса равен 1 или сигнал En равен 0 ("Q → R+En").

Временная диаграмма (Timing diagram)

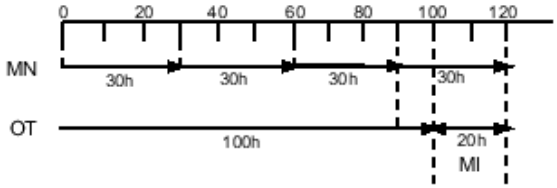


- Временная диаграмма, показывающая взаимосвязь между сигналами En, Ral, OUT_Q, значениями MN (оставшееся время) и MI (заданный интервал). На диаграмме видно, как MN уменьшается от MI до 0 при активном En, и как OUT_Q устанавливается в 1 при MN=0. Сигнал Ral сбрасывает MN к MI и OT к 0.

Предельное значение OT

Значение рабочих часов в OT сохраняется при сбросе счетчика моточасов сигналом на входе R. Счетчик моточасов OT будет сброшен в ноль при переходе из 0 в 1 на входе Ral. Счетчик моточасов OT продолжает счет, пока En = 1, независимо от статуса на входе сброса R. Предел счетчика OT составляет 99999 часов. Счетчик моточасов останавливается, когда достигает этого значения.
В режиме программирования вы можете установить начальное значение OT. MN рассчитывается по следующей формуле, если вход сброса R никогда не активировался: MN = MI - (OT % MI). Оператор % предоставляет остаток от целочисленного деления.

Пример:
MI = 30ч, OT = 100ч
MN = 30 - (100 % 30)
MN = 30 - 10
MN = 20ч

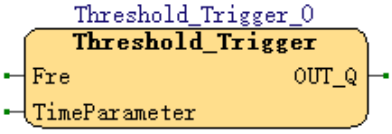


- Диаграмма, иллюстрирующая пример: шкала от 0 до 120 часов, на которой отмечены интервалы MI по 30 часов, текущее значение OT=100ч и рассчитанное оставшееся время MN=20ч до следующего обслуживания.

В режиме выполнения значение OT не может быть предустановлено. Если значение MI изменено, перерасчет для MN выполняться не будет. MN примет значение MI.

2.15.2 Threshold_Trigger (Пороговый триггер)

Выходы OUT_Q переключаются в зависимости от двух настраиваемых частот.

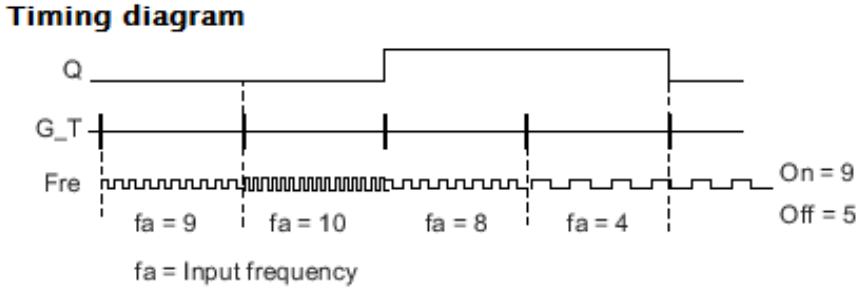


Имя	Ввод/Вывод	Тип данных	Описание
Fre	IN	BOOL	Функция подсчитывает переходы 0→1 на входе Fre. Переходы 1→0 не подсчитываются.
TimeParameter	IN	TIME	Временной интервал или время стробирования, в течение которого измеряются входные импульсы. • On: Порог включения. Диапазон значений: 0000 до 9999 • Off: Порог выключения. Диапазон значений:

			0000 до 9999 • G_T: Временной интервал или время стробирования, в течение которого измеряются входные импульсы. Диапазон значений: 00:00 с до 99:99 с
OUT_Q	OUT	BOOL	Q устанавливается и сбрасывается в соответствии с фактическим значением Cnt и установленными порогами.

Триггер Threshold_Trigger измеряет сигналы на входе Fre. Импульсы фиксируются в течение настраиваемого периода G_T. OUT_Q устанавливается или сбрасывается в соответствии с установленными порогами. См. следующее правило расчета.

Временная диаграмма (Timing diagram)



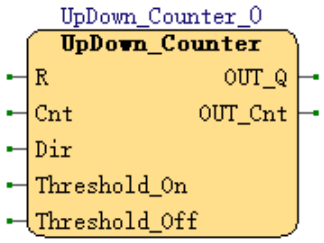
- Диаграмма, показывающая взаимосвязь между сигналами Fre, G_T и OUT_Q. На диаграмме отмечены моменты измерения частоты fa и состояния выхода Q в зависимости от сравнения fa с порогами On и Off.

Правило расчета (Calculation rule)

Если порог On ≥ порога Off, то:
OUT_Q = 1, если fa > On
OUT_Q = 0, если fa ≤ Off.
Если порог On < порога Off, то OUT_Q = 1, если:
On ≤ fa < Off.

2.15.3 UpDown_Counter (Реверсивный счетчик)

Входной импульс увеличивает или уменьшает внутреннее значение в зависимости от настройки параметров. Выход устанавливается или сбрасывается при достижении настроенного порога. Направление счета можно изменить сигналом на входе Dir.

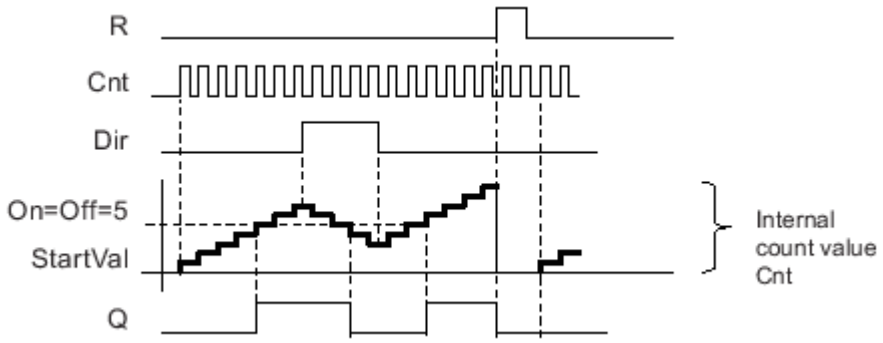


Имя	Ввод/Вывод	Тип данных	Описание
R	IN	BOOL	Сигнал на входе R (Reset) сбрасывает выход и внутреннее значение счетчика на начальное значение (StartVal).
Cnt	IN	BOOL	Эта функция подсчитывает переходы 0→1 на входе Cnt. Переходы 1→0 не подсчитываются.
Dir	IN	BOOL	Вход Dir (Direction) определяет направление счета: Dir = 0: Вверх, Dir = 1: Вниз
Threshold_On	IN	UDINT	Порог включения
Threshold_Off	IN	UDINT	Порог выключения
OUT_Q	OUT	BOOL	Q устанавливается и сбрасывается в соответствии с фактическим значением Cnt и установленными порогами.
OUT_Cnt	OUT	UDINT	Текущее значение счетчика.

Функция UpDown_Counter увеличивает (Dir = 0) или уменьшает (Dir = 1) внутренний счетчик на единицу с каждым положительным фронтом на входе Cnt.
Вы можете сбросить внутреннее значение счетчика на начальное значение с помощью сигнала на входе сброса R. Пока R=1, выход Q равен 0, и импульсы на входе Cnt не подсчитываются.
Выход OUT_Q устанавливается и сбрасывается в соответствии с фактическим значением OUT_Cnt и установленными порогами. См. следующие правила расчета.

Правило расчета (Calculation rule)
Если порог включения On ≥ порога выключения Off, то:
OUT_Q = 1, если OUT_Cnt ≥ On
OUT_Q = 0, если OUT_Cnt < Off.
Если порог включения On < порога выключения Off, то:
OUT_Q = 1, если On ≤ OUT_Cnt < Off.

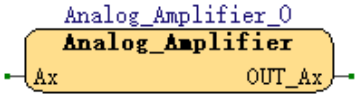
Временная диаграмма (Timing diagram)



- Диаграмма, показывающая работу счетчика. Пример: On=5, Off=5, StartVal=0. При Dir=0 и импульсах на Cnt значение OUT_Cnt увеличивается. OUT_Q устанавливается в 1, когда OUT_Cnt достигает 5. Сигнал R сбрасывает OUT_Cnt в StartVal и OUT_Q в 0.*

2.15.4 Analog_Amplifier (Аналоговый усилитель)

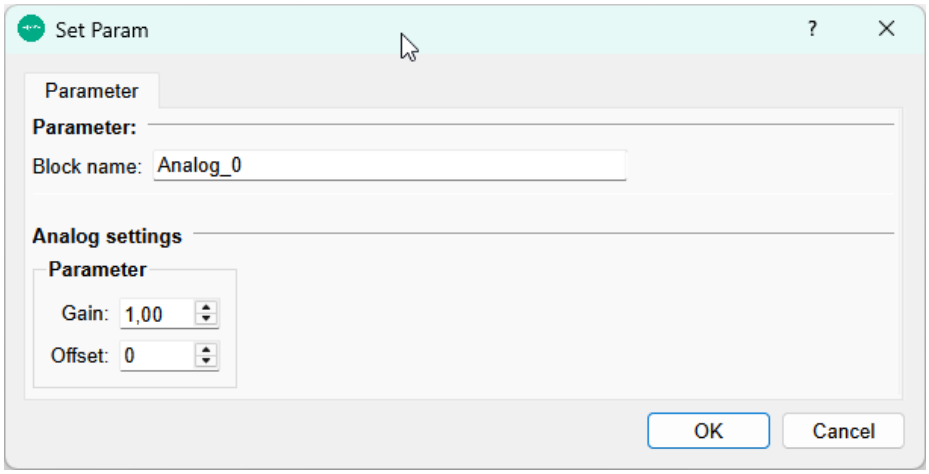
Этот функциональный блок усиливает аналоговое входное значение и возвращает его на аналоговом выходе.



Имя	Ввод/Вывод	Тип данных	Описание
Ax	IN	INT	Аналоговый сигнал
OUT_Ax	OUT	INT	Усиленный аналоговый сигнал. Диапазон: -32768 ~ +32767

Set Param (Установка параметров)

Дважды щелкните на блоке, чтобы открыть диалоговое окно настройки параметров.



Диалоговое окно содержит следующие поля:

- **Block name** (Имя блока): Analog_0
- **Gain** (Усиление): 1.00
- **Offset** (Смещение): 0

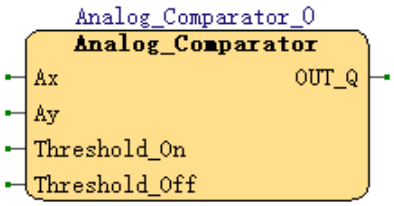
Gain (Коэффициент усиления)
Диапазон значений: -10.00 до 10.00
Offset (Смещение)
Диапазон значений: -10000 до 10000

Описание функции (Description of the function)
Функция Analog_Amplifier считывает значение аналогового сигнала на входе Ax.
Это значение умножается на параметр Gain (A). Параметр Offset (B) добавляется к произведению следующим образом:
(Ax * Gain) + Offset = Actual value Ax (Фактическое значение Ax).
Фактическое значение Ax является выходным значением OUT_Ax.

Аналоговый выход (Analog output)
Если вы подключаете эту специальную функцию к реальному аналоговому выходу, обратите внимание, что аналоговый выход может обрабатывать только значения от 0 до 1000. Для этого подключите дополнительный усилитель между аналоговым выходом специальной функции и реальным аналоговым выходом. С помощью этого усилителя вы стандартизируете выходной диапазон специальной функции до диапазона значений от 0 до 1000.

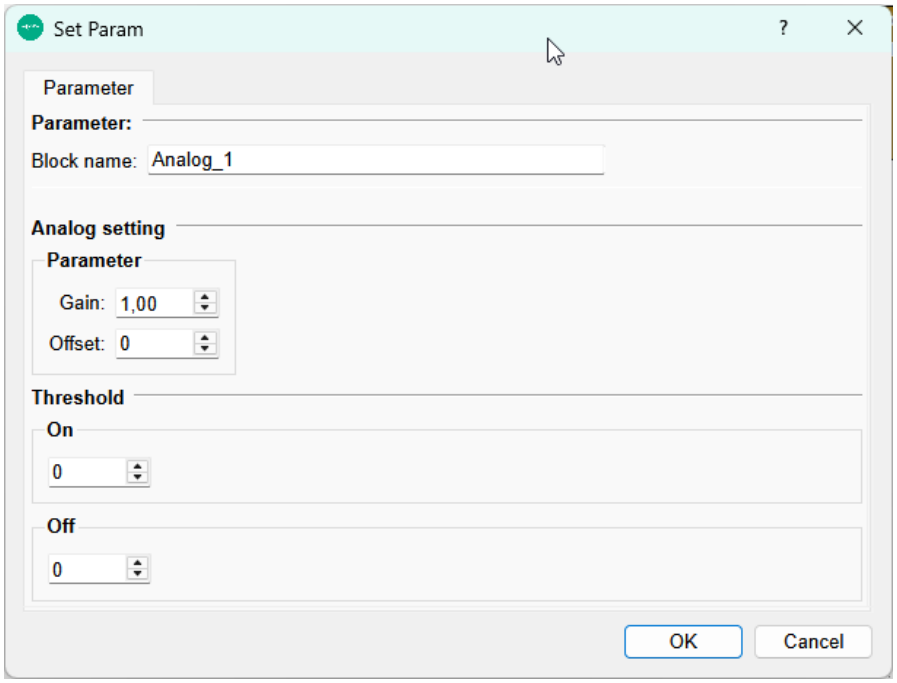
2.15.5 Analog_Comparator (Аналоговый компаратор)

Выход устанавливается и сбрасывается в зависимости от разности Ax-Ay и двух настраиваемых порогов.



Имя	Ввод/Вывод	Тип данных	Описание
Ax	IN	INT	Аналоговый сигнал
Ay	IN	INT	Аналоговый сигнал
Threshold_On	IN	INT	Порог включения
Threshold_Off	IN	INT	Порог выключения
OUT_Q	OUT	BOOL	Q устанавливается или сбрасывается в зависимости от установленных порогов.

Set Param (Установка параметров)
Дважды щелкните на блоке, чтобы открыть диалоговое окно настройки параметров.



Диалоговое окно содержит следующие поля:

- **Block name** (Имя блока): Analog_1
- **Gain** (Усиление): 1.00
- **Offset** (Смещение): 0
- **On** (Порог вкл.): 0
- **Off** (Порог выкл.): 0

Gain (Коэффициент усиления)

Диапазон значений: -10.00 до 10.00

Offset (Смещение)

Диапазон значений: -10000 до 10000

On: Порог включения (On threshold)

Диапазон значений: -20000 до 20000

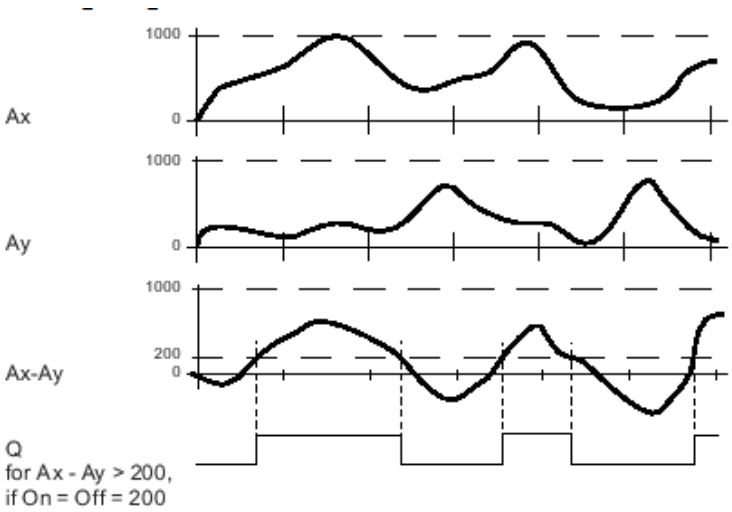
Off: Порог выключения (Off threshold)

Диапазон значений: -20000 до 20000

Parameter p (number of decimals)

Параметр p применяется только к значениям Ax, Ay, Delta, On и Off, отображаемым в тексте сообщения. Параметр p не применяется к сравнению значений On и Off. (Функция сравнения игнорирует десятичную точку.)

Временная диаграмма (Timing diagram)



- Диаграмма, показывающая взаимосвязь между сигналами Ax, Ay, разностью Ax-Ay и состоянием выхода OUT_Q. Пример: OUT_Q = 1, если Ax - Ay > 200, при On = Off = 200.*

Описание функции (Description of the function)

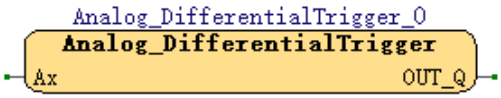
Функция Analog_Comparator считывает значение сигнала на аналоговом входе Ax.
Это значение умножается на значение параметра A (Gain). Параметр B (Offset) добавляется к произведению, следовательно:
(Ax * Gain) + Offset = Actual value Ax (Фактическое значение Ax).
(Ay * Gain) + Offset = Actual value Ay (Фактическое значение Ay).
Выход OUT_Q устанавливается или сбрасывается в зависимости от разности фактических значений Ax – Ay и установленных порогов. См. следующее правило расчета.

Правило расчета (Calculation rule)

Если порог On ≥ порога Off, то:
OUT_Q = 1, если (Actual value Ax - Actual value Ay) > On
OUT_Q = 0, если (Actual value Ax - Actual value Ay) ≤ Off.
Если порог On < порога Off, то OUT_Q = 1, если:
On ≤ (Actual value Ax - Actual value Ay) < Off.

2.15.6 Analog_DifferentialTrigger (Аналоговый дифференциальный триггер)

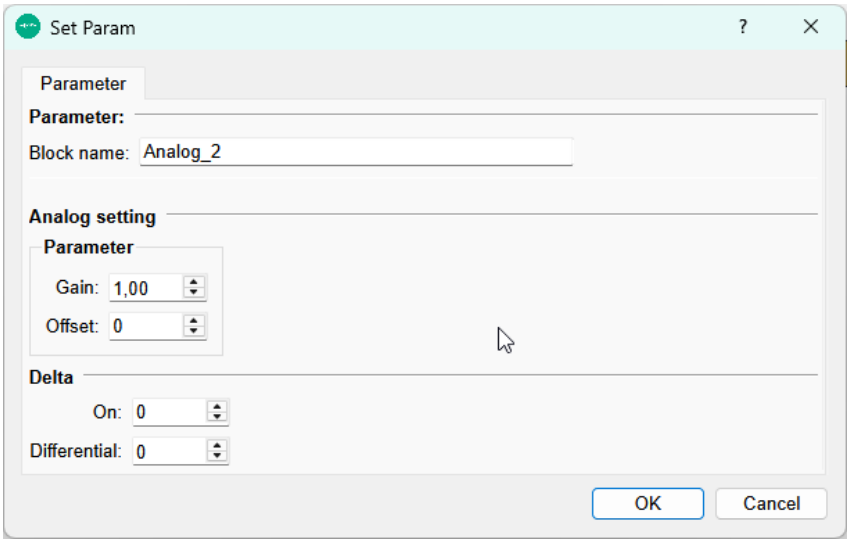
Выход устанавливается и сбрасывается в зависимости от настраиваемого порога и дифференциального значения.



Имя	Ввод/Вывод	Тип данных	Описание
Ax	IN	INT	Аналоговый сигнал
OUT_Q	OUT	BOOL	Q устанавливается или сбрасывается в зависимости от порога и разностных значений.

Set Param (Установка параметров)

Дважды щелкните на блоке, чтобы открыть диалоговое окно настройки параметров.



Диалоговое окно содержит следующие поля:

- **Block name** (Имя блока): Analog_2
- **Gain** (Усиление): 1.00
- **Offset** (Смещение): 0
- **On** (Порог): 0
- **Differential** (Дифференциальное значение): 0

Gain (Коэффициент усиления)

Диапазон значений: -10.00 до 10.00

Offset (Смещение)

Диапазон значений: -10000 до 10000

On: Порог включения/выключения (On/Off threshold)

Диапазон значений: -20000 до 20000

DifferentialΔt: Дифференциальное значение для расчета параметра выключения (Differential value for calculating the off parameter)

Диапазон значений: -20000 до 20000

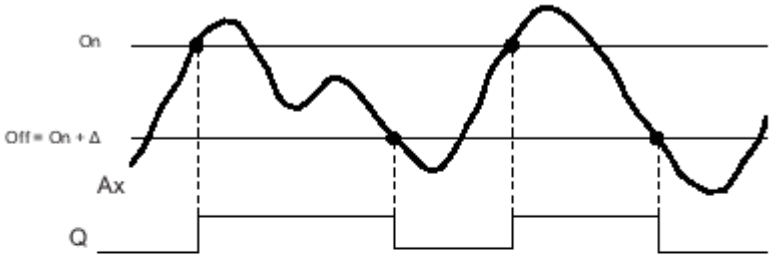
Временная диаграмма (Timing diagram) A: Функция с отрицательным дифференциальным значением Delta

- Диаграмма, показывающая работу с отрицательным Delta. Порог On ≥ порога Off. OUT_Q = 1, когда Ax > On, и OUT_Q = 0, когда Ax ≤ Off.*

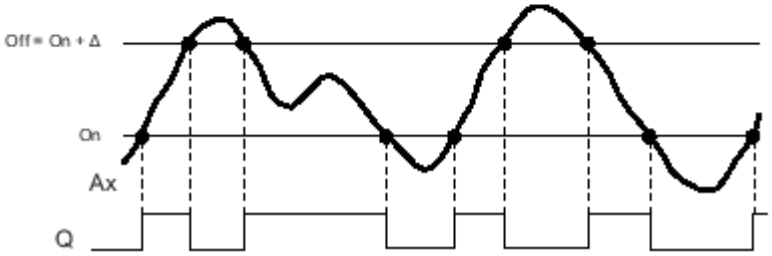
Временная диаграмма (Timing diagram) B: Функция с положительным дифференциальным значением Delta

- Диаграмма, показывающая работу с положительным Delta. Порог On < порога Off. OUT_Q = 1, когда On ≤ Ax < Off.*

Timing diagram A: Function with negative difference Delta



Timing diagram B: Function with positive difference Delta



Описание функции (Description of the function)

Функция Analog_DifferentialTrigger получает аналоговый сигнал на входе Ax.
Ax умножается на значение параметра A (Gain), и значение параметра B (Offset) добавляется к произведению, т.е.
(Ax * Gain) + Offset = Actual value of Ax (Фактическое значение Ax).
Выход OUT_Q устанавливается или сбрасывается в зависимости от установленного порога On и разностного значения Delta. Функция автоматически вычисляет параметр Off: Off = On + Delta, где Delta может быть положительным или отрицательным. См. правило расчета ниже.

Правило расчета (Calculation rule)

Когда вы устанавливаете отрицательное дифференциальное значение Delta, порог On ≥ порога Off, и:
OUT_Q = 1, если Actual value Ax > On
OUT_Q = 0, если Actual value Ax ≤ Off.
См. временную диаграмму A.
Когда вы устанавливаете положительное дифференциальное значение Delta, порог On < порога Off, и OUT_Q = 1, если:
On ≤ Actual value Ax < Off.
См. временную диаграмму B.

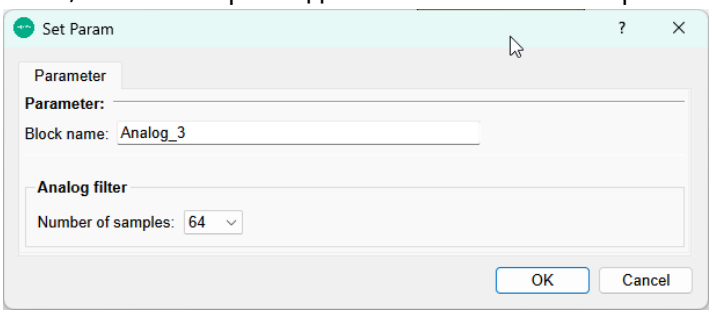
2.15.7 Analog_Filter (Аналоговый фильтр)

Функциональный блок Analog_Filter используется для сглаживания аналогового входного сигнала.

Имя	Ввод/Вывод	Тип данных	Описание
Ax	IN	INT	Аналоговый сигнал
OUT_AQ	OUT	INT	AQ выдает среднее значение аналогового входа Ax по текущему количеству выборок, и оно устанавливается или сбрасывается в зависимости от аналогового входа и количества выборок.

Set Param (Установка параметров)

Дважды щелкните на блоке, чтобы открыть диалоговое окно настройки параметров.

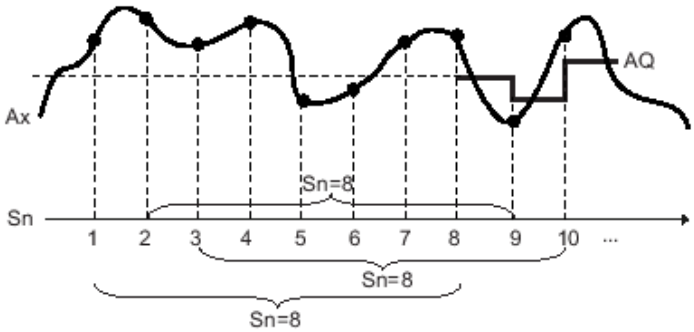


Диалоговое окно содержит следующие поля:

- **Block name** (Имя блока): Analog_3
- **Number of samples** (Количество выборок): 64

После того как вы установите параметр, аналоговый фильтр вычисляет среднее значение выборок и присваивает это значение OUT_AQ.

Временная диаграмма (Timing diagram)

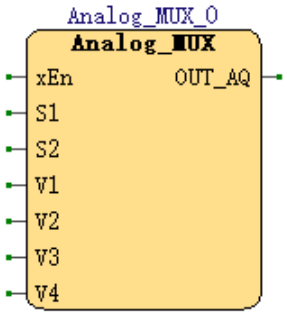


- *Диаграмма, показывающая входной сигнал Ax и выходной сигнал OUT_AQ после усреднения по 8 выборкам (Sn=8). Выходной сигнал сглажен и запаздывает относительно входного.*

Функция выдает среднее значение после выборки аналогового входного сигнала в соответствии с установленным количеством выборок. Этот SFB может уменьшить ошибку аналогового входного сигнала.

2.15.8 Analog_MUX (Аналоговый мультиплексор)

Когда разрешено, специальный функциональный блок Analog_MUX отображает одно из четырех предопределенных аналоговых значений в зависимости от условий на входах.



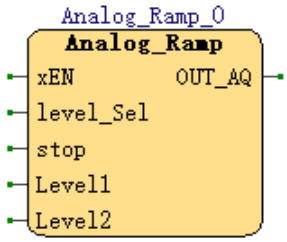
Имя	Ввод/Вывод	Тип данных	Описание
xEn	IN	BOOL	1 на входе En (Enable — Разрешение) переключает, в зависимости от S1 и S2, параметризованное аналоговое значение на выход AQ. 0 на входе EN переключает 0 на выход AQ.
S1	IN	BOOL	Селектор 1 для выбора аналогового значения.
S2	IN	BOOL	Селектор 2 для выбора аналогового значения.
V1	IN	INT	Аналоговое значение 1.
V2	IN	INT	Аналоговое значение 2.
V3	IN	INT	Аналоговое значение 3.
V4	IN	INT	Аналоговое значение 4.
OUT_AQ	OUT	INT	Выходное аналоговое значение.

Логика выбора (Selection Logic):

- S1 = 0 и S2 = 0: Выводится значение V1
- S1 = 0 и S2 = 1: Выводится значение V2
- S1 = 1 и S2 = 0: Выводится значение V3
- S1 = 1 и S2 = 1: Выводится значение V4

2.15.9 Analog_Ramp (Аналоговый рамповый генератор)

Инструкция Analog_Ramp позволяет изменять выход от текущего уровня до выбранного уровня с заданной скоростью.

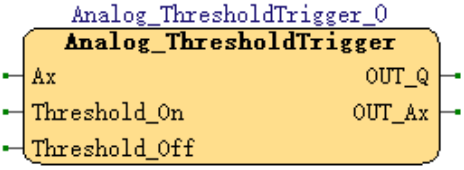


Имя	Ввод/Вывод	Тип данных	Описание
xEn	IN	BOOL	Изменение статуса с 0 на 1 на входе En (Enable — Разрешение) применяет старт/стоп уровень (Offset "B" + StSp) к выходу на 100 мс и запускает рамповое

			изменение до выбранного уровня. Изменение статуса с 1 на 0 немедленно устанавливает текущий уровень равным Offset "B", что делает выход AQ равным 0.
level_Sel	IN	BOOL	level_Sel = 0: Выбран шаг 1 (уровень 1). level_Sel = 1: Выбран шаг 2 (уровень 2). Изменение статуса Sel вызывает начало изменения текущего уровня до выбранного уровня с указанной скоростью.
stop	IN	BOOL	Изменение статуса с 0 на 1 на входе St (Decelerated Stop — Останов с замедлением) вызывает уменьшение текущего уровня с постоянной скоростью до достижения старт/стоп уровня (Offset "B" + StSp). Старт/стоп уровень поддерживается в течение 100 мс, а затем текущий уровень устанавливается равным Offset "B", что делает выход AQ равным 0.
Level1	IN	INT	Уровень 1.
Level2	IN	INT	Уровень 2.
OUT_AQ	OUT	INT	Выход AQ масштабируется с использованием формулы: (Current Level - Offset "B") / Gain "A".

2.15.10 Analog_ThresholdTrigger (Аналоговый пороговый триггер)

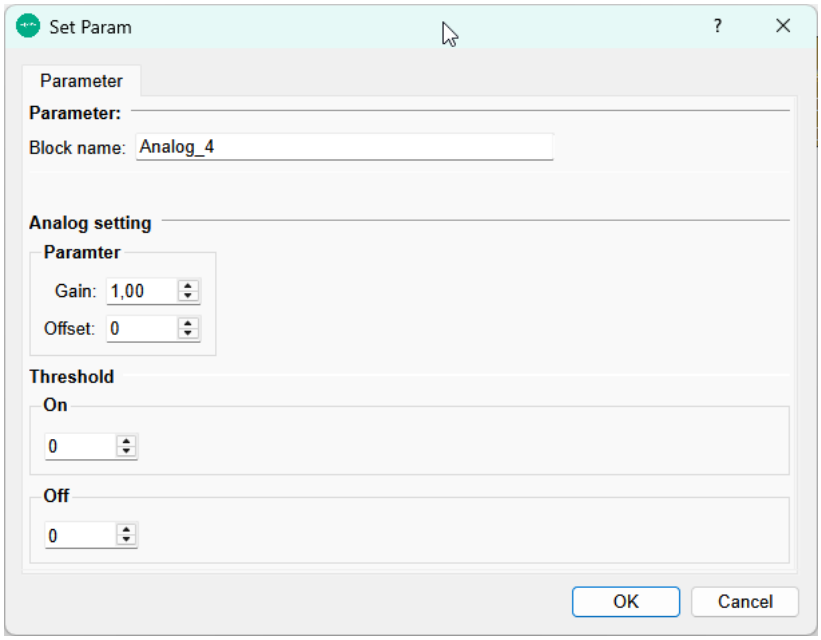
Выход устанавливается или сбрасывается в зависимости от двух настраиваемых порогов (гистерезис).



Имя	Ввод/Вывод	Тип данных	Описание
Ax	IN	INT	Аналоговый сигнал
Threshold_On	IN	INT	Порог включения. Диапазон значений: -20000 до 20000
Threshold_Off	IN	INT	Порог выключения. Диапазон значений: -20000 до 20000
OUT_Q	OUT	BOOL	Q устанавливается или сбрасывается в зависимости от установленных порогов.
OUT_Ax	OUT	INT	Обработанный аналоговый сигнал.

Set Param (Установка параметров)

Дважды щелкните на блоке, чтобы открыть диалоговое окно настройки параметров.



Диалоговое окно содержит следующие поля:

- **Block name** (Имя блока): Analog_5
- **Gain** (Усиление): 1.00
- **Offset** (Смещение): 0
- **On** (Порог вкл.): 0
- **Off** (Порог выкл.): 0

Gain (Коэффициент усиления)

Диапазон значений: -10.00 до 10.00

Offset (Смещение)

Диапазон значений: -10000 до 10000

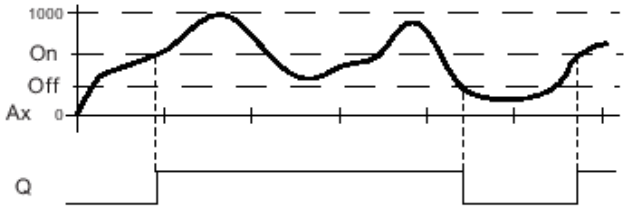
On: Порог включения (On threshold)

Диапазон значений: -20000 до 20000

Off: Порог выключения (Off threshold)

Диапазон значений: -20000 до 20000

Временная диаграмма (Timing diagram)



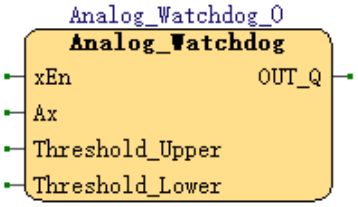
- Диаграмма, показывающая гистерезисную характеристику. Выход OUT_Q включается, когда входной сигнал Ax превышает порог On, и выключается, когда Ax опускается ниже порога Off.

Правило расчета (Calculation rule)

Если порог On ≥ порога Off, то:
OUT_Q = 1, если Actual value Ax > On
OUT_Q = 0, если Actual value Ax <= Off.
Если порог On < порога Off, то OUT_Q = 1, если:
On <= Actual value Ax < Off.

2.15.11 Analog_Watchdog (Аналоговый сторожевой таймер)

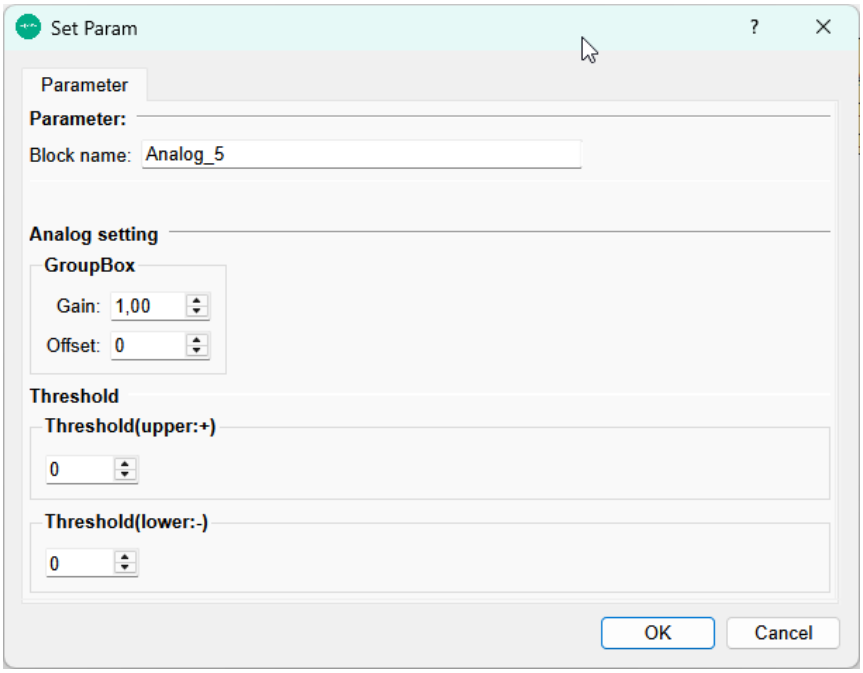
Этот специальный функциональный блок сохраняет переменную процесса аналогового входа в память и устанавливает выход, когда выходная переменная превышает или падает ниже этого сохраненного значения плюс настраиваемое смещение.



Имя	Ввод/Вывод	Тип данных	Описание
xEn	IN	BOOL	Положительный фронт (переход 0→1) на входе En сохраняет аналоговое значение на входе Ax ("Aen") в память и запускает мониторинг аналогового диапазона Aen + Delta.
Ax	IN	INT	Аналоговый входной сигнал.
Threshold_Upper	IN	INT	Разностное значение выше Aen: порог вкл/выкл.
Threshold_Lower	IN	INT	Разностное значение ниже Aen: порог вкл/выкл.
OUT_Q	OUT	BOOL	Q устанавливается/сбрасывается в зависимости от сохраненного аналогового значения и смещения.

Set Param (Установка параметров)

Дважды щелкните на блоке, чтобы открыть диалоговое окно настройки параметров.



Диалоговое окно содержит следующие поля:

- **Block name** (Имя блока): Analog_6
- **Gain** (Усиление): 1.00
- **Offset** (Смещение): 0
- **Threshold(upper:+)** (Порог (верхний:+)): 0
- **Threshold(lower:-)** (Порог (нижний:-)): 0

Gain (Коэффициент усиления)

Диапазон значений: -10.00 до 10.00

Offset (Смещение)

Диапазон значений: -10000 до 10000

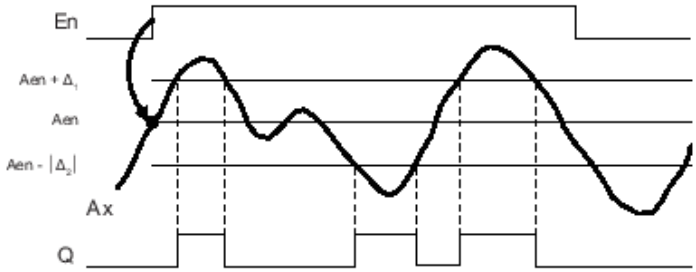
Threshold 1: Разностное значение выше Aen: порог вкл/выкл (Difference value above Aen: on/off threshold)

Диапазон значений: 0 до 20000

Threshold 2: Разностное значение ниже Aen: порог вкл/выкл (Difference value below Aen: on/off threshold)

Диапазон значений: 0 до 20000

Временная диаграмма (Timing diagram)



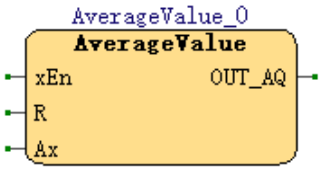
- Диаграмма, показывающая работу сторожевика. В момент фронта на En фиксируется значение Aen. OUT_Q устанавливается в 1, когда входной сигнал Ax выходит за пределы диапазона Aen + Threshold_1 / Aen - Threshold_2.*

Описание функции (Description of the function)

Переход 0→1 на входе En сохраняет значение сигнала на аналоговом входе Ax. Эта сохраненная переменная процесса обозначается как "Aen".
Как аналоговые фактические значения Ax, так и Aen умножаются на значение параметра A (Gain), и затем параметр B (Offset) добавляется к произведению, следующим образом:
(Ax * Gain) + Offset = Actual value Aen (когда вход En изменяется с 0 на 1), или
(Ax * Gain) + Offset = Actual value Ax.
Выход OUT_Q устанавливается, когда сигнал на входе En = 1 и если фактическое значение на входе Ax выходит за пределы диапазона Aen + Threshold_1 / Aen - Threshold_2.
Выход OUT_Q сбрасывается, когда фактическое значение на входе Ax находится в пределах диапазона Aen + Threshold_1 / Aen - Threshold_2, или когда сигнал на входе En изменяется на низкий уровень.

2.15.12 AverageValue (Усреднение значения)

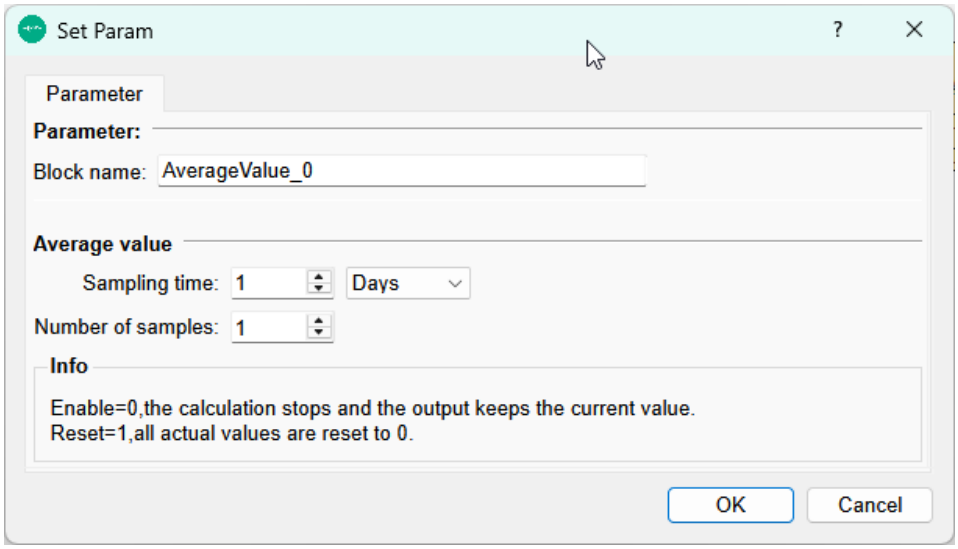
Функция AverageValue выполняет выборку аналогового входного сигнала в течение настроенного периода времени и выдает среднее значение на AQ.



Имя	Ввод/Вывод	Тип данных	Описание
xEn	IN	BOOL	Положительный фронт (переход 0→1) на входе En (Enable — Разрешение) устанавливает выход AQ равным среднему значению входа Ax после настроенного времени. Отрицательный фронт (переход 1→0) удерживает выход на последнем вычисленном значении.
R	IN	BOOL	Положительный фронт (переход 0→1) на входе R (Reset — Сброс) сбрасывает выход AQ в 0.
Ax	IN	INT	Аналоговый сигнал.
OUT_AQ	OUT	INT	AQ выдает среднее значение за указанное время выборки.

Set Param (Установка параметров)

Дважды щелкните на блоке, чтобы открыть диалоговое окно настройки параметров.



Диалоговое окно содержит следующие поля:

- **Block name** (Имя блока): AverageValue_0
- **Sampling time** (Время выборки): 1 Days (Дни)
- **Number of samples** (Количество выборок): 1
- **Info** (Информация): Enable=0,the calculation stops and the output keeps the current value. Reset=1,all actual values are reset to 0. (Enable=0, вычисления останавливаются и выход сохраняет текущее значение. Reset=1, все текущие значения сбрасываются в 0.)

St (Sampling time — Время выборки): Можно установить в Секундах, Днях, Часах или Минутах.

Диапазоны значений:

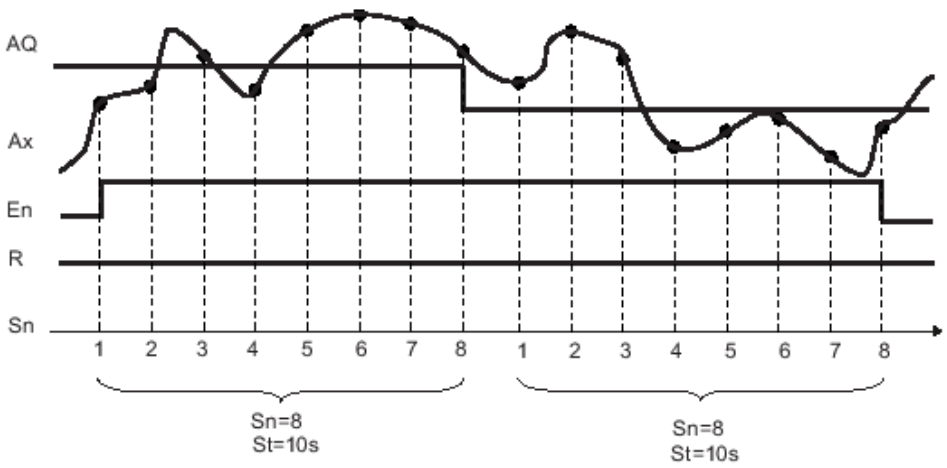
- Если St = Секунды: 1 до 59
- Если St = Дни: 1 до 365
- Если St = Часы: 1 до 23
- Если St = Минуты: 1 до 59

Sn (Number of samples — Количество выборок):

Диапазоны значений:

- Если St = Секунды: 1 до St*100
- Если St = Дни: 1 до 32767
- Если St = Часы: 1 до 32767
- Если St = Минуты и St <= 5 минут: 1 до St*6000
- Если St = Минуты и St >= 6 минут: 1 до 32767

Временная диаграмма (Timing diagram)



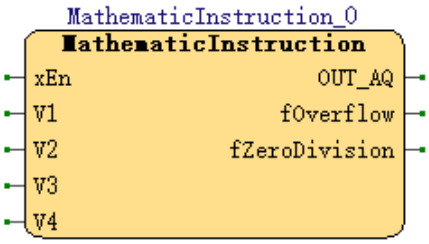
- Диаграмма, показывающая процесс усреднения. Пример: Sn=8, St=10s. Входной сигнал Ax изменяется. При активации En начинается вычисление среднего. Через время St (10c) и после Sn выборок (8) на выходе OUT_AQ устанавливается вычисленное среднее значение. Сигнал R сбрасывает выход в 0.

Описание функции (Description of the function)

Когда En = 1, функция AverageValue вычисляет среднее значение выборок в течение настроенного временного интервала. В конце времени выборки эта функция устанавливает выход AQ равным этому вычисленному среднему значению.
Когда En = 0, вычисление останавливается, и AQ сохраняет последнее вычисленное значение.
Когда R = 1, AQ сбрасывается в 0.

2.15.13 Mathematicinstruction (Математическая инструкция)

Функциональный блок Mathematicinstruction вычисляет значение AQ уравнения, сформированного из пользовательских операндов и операторов.

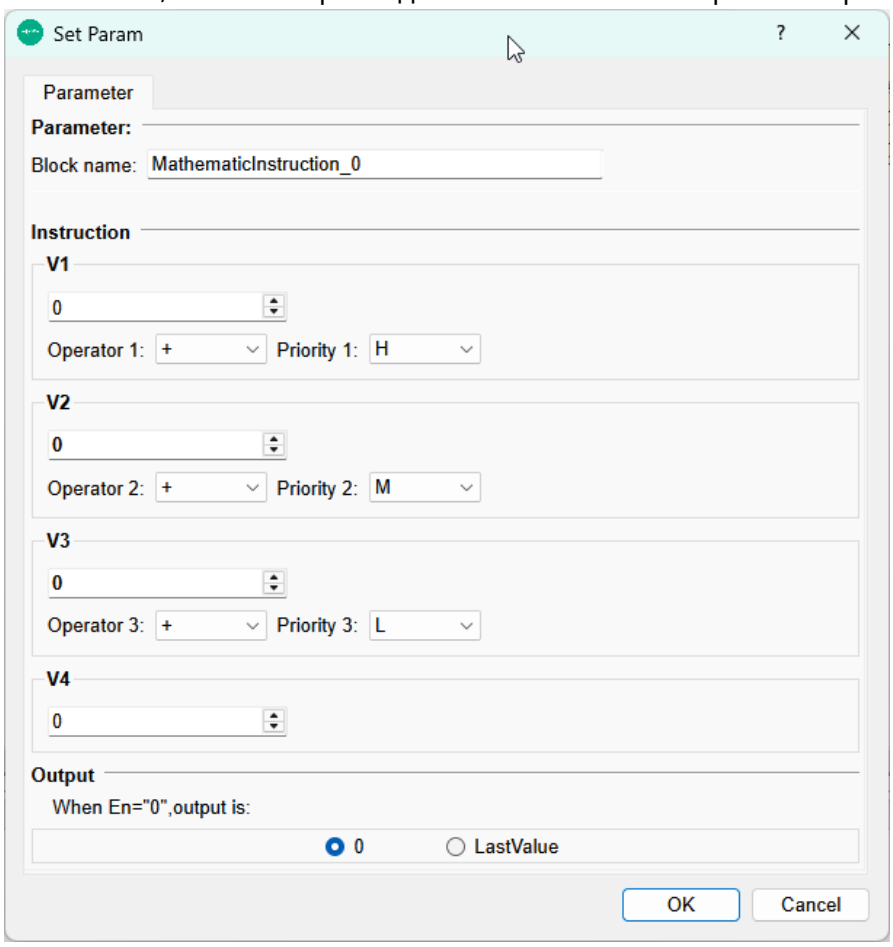


Имя	Ввод/Вывод	Тип данных	Описание
xEn	IN	BOOL	Положительный фронт на входе En разрешает работу функционального блока математической инструкции.
V1	IN	INT	Значение 1: Первый операнд.
V2	IN	INT	Значение 2: Второй операнд.
V3	IN	INT	Значение 3: Третий операнд.
V4	IN	INT	Значение 4: Четвертый операнд.

OUT_AQ	OUT	INT	Выход AQ является результатом уравнения, сформированного из значений операндов и операторов.
fOverflow	OUT	BOOL	Выход TRUE, если в текущем периоде происходит переполнение.
fZeroDivision	OUT	BOOL	Выход TRUE, когда в текущем периоде происходит деление на 0.

Set Param (Установка параметров)

Дважды щелкните на блоке, чтобы открыть диалоговое окно настройки параметров.



Диалоговое окно содержит следующие поля:

- **Block name** (Имя блока): MathematicInstruction_0
- **V1** (Значение 1): Поле ввода для первого операнда.
- **Operator 1** (Оператор 1): Выпадающий список для выбора первого оператора (+, -, *, /).
- **Priority 1** (Приоритет 1): Выпадающий список для выбора приоритета первой операции (High, Medium, Low).
- **V2** (Значение 2): Поле ввода для второго операнда.
- **Operator 2** (Оператор 2): Выпадающий список для выбора второго оператора.
- **Priority 2** (Приоритет 2): Выпадающий список для выбора приоритета второй операции.

- **V3** (Значение 3): Поле ввода для третьего операнда.
- **Operator 3** (Оператор 3): Выпадающий список для выбора третьего оператора.
- **Priority 3** (Приоритет 3): Выпадающий список для выбора приоритета третьей операции.
- **V4** (Значение 4): Поле ввода для четвертого операнда.
- **Output** (Выход): Настройка поведения выхода при En=0 (0 или LastValue).

Диапазон значений: -32768 до 32767

Описание функции (Description of the function)

Функция Mathematicinstruction объединяет четыре операнда и три оператора для формирования уравнения. Оператором может быть любой из четырех стандартных операторов: +, -, * или /. Для каждого оператора необходимо установить уникальный приоритет: Высокий ("H"), Средний ("M") или Низкий ("L"). Операция с высоким приоритетом выполняется первой, за ней следует операция со средним приоритетом, а затем операция с низким приоритетом. У вас должна быть ровно одна операция каждого приоритета. Значения операндов могут ссылаться на другую ранее определенную функцию для предоставления значения. Функция математической инструкции округляет результат до ближайшего целого значения.

Количество значений операндов фиксировано на четыре, а количество операторов фиксировано на три. Если вам нужно использовать меньше операндов, используйте конструкции типа "+ 0" или "* 1" для заполнения оставшихся параметров.

Вы также можете настроить поведение функции, когда параметр разрешения "En"=0. Функциональный блок может либо сохранять свое последнее значение, либо быть установленным в 0.

Примеры (Examples)

В таблицах ниже показаны параметры некоторых простых примеров блоков математических инструкций, а также результирующие уравнения и выходные значения:

V1	Operator1 (Priority 1)	V2	Operator2 (Priority 2)	V3	Operator3 (Priority 3)	V4
12	+ (M)	6	/ (H)	3	- (L)	1

Уравнение: (12 + (6 / 3)) - 1

Результат: 13

V1	Operator1 (Priority 1)	V2	Operator2 (Priority 2)	V3	Operator3 (Priority 3)	V4
2	+ (L)	3	* (M)	1	+ (H)	4

Уравнение: 2 + (3 * (1 + 4))

Результат: 17

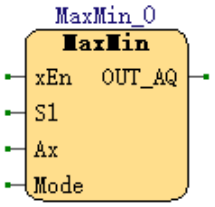
V1	Operator1 (Priority 1)	V2	Operator2 (Priority 2)	V3	Operator3 (Priority 3)	V4
100	- (H)	25	/ (L)	2	+ (M)	1

Уравнение: (100 - 25) / (2 + 1)

Результат: 25

2.15.14 MaxMin (Максимум/минимум)

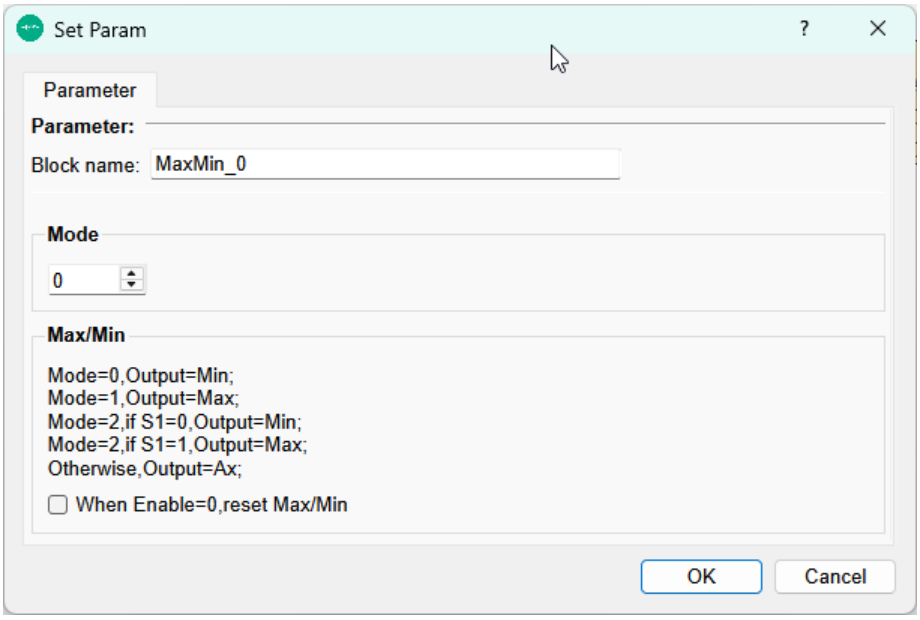
Функциональный блок MaxMin записывает максимальное или минимальное значение.



Имя	Ввод/Вывод	Тип данных	Описание
xEn	IN	BOOL	Функция входа En (Enable — Разрешение) зависит от настроек параметра Mode и выбора флажка "when En = 0, reset Max/Min".
S1	IN	BOOL	Этот вход активен, когда вы устанавливаете Mode = 2: Положительный переход (0→1) на входе S1 устанавливает выход AQ равным максимальному значению. Отрицательный переход (1→0) на входе S1 устанавливает выход AQ равным минимальному значению.
Ax	IN	INT	Аналоговый сигнал.
Mode	IN	USINT	Возможные настройки: 0, 1, 2, 3
OUT_AQ	OUT	INT	AQ выдает минимальное, максимальное или фактическое значение в зависимости от входов или сбрасывается в 0, если настроено так делать, когда функция отключена.

Set Param (Установка параметров)

Дважды щелкните на блоке, чтобы открыть диалоговое окно настройки параметров.



Диалоговое окно содержит следующие поля:

- **Block name** (Имя блока): MaxMin_0
- **Mode** (Режим): 0
- **Max/Min** (Макс/Мин): Описание режимов.
- **When Enable=0, reset Max/Min** (Когда Enable=0, сбросить Макс/Мин): Флажок.

Режимы работы:

- Mode = 0: AQ = Min
- Mode = 1: AQ = Max
- Mode = 2 и S1= 0 (низкий): AQ = Min
- Mode = 2 и S1= 1 (высокий): AQ = Max
- Mode = 3 или на значение блока есть ссылка: AQ = Ax

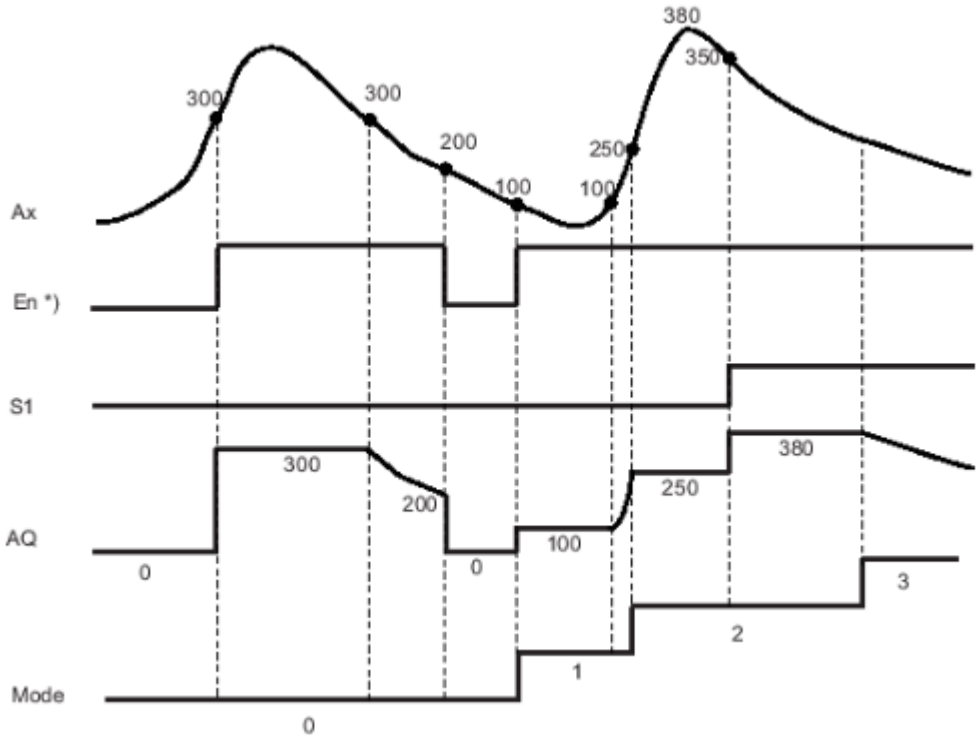
Если вы выбрали флажок "when Enable = 0, reset Max/Min":

- Enable = 0: Функция устанавливает значение AQ в 0.
- Enable = 1: Функция выдает значение на AQ в зависимости от настроек Mode и S1.

Если вы не выбрали флажок "when Enable = 0, reset Max/Min":

- Enable = 0: Функция удерживает значение AQ на текущем значении.
- Enable = 1: Функция выдает значение на AQ в зависимости от настроек Mode и S1.

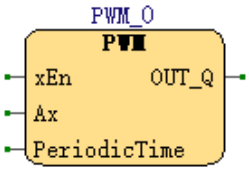
Временная диаграмма (Timing diagram)



- Диаграмма, показывающая работу блока. Пример: Mode=2. При S1=0 на выходе AQ отображается минимальное значение входного сигнала Ax. При S1=1 на выходе AQ отображается максимальное значение. Сигнал En разрешает или приостанавливает фиксацию значений. При сбросе (если флажок установлен) выход обнуляется.

2.15.15 PWM (ШИМ-модулятор)

Инструкция Pulse Width Modulator (PWM) модулирует аналоговое входное значение Ax в импульсный цифровой выходной сигнал. Ширина импульса пропорциональна аналоговому значению Ax.

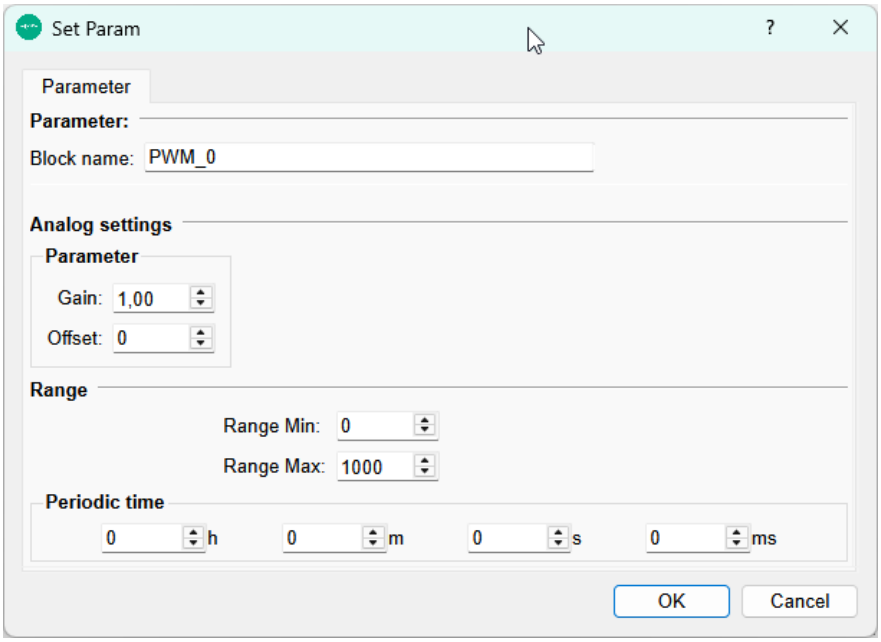


Имя	Ввод/Вывод	Тип данных	Описание
xEn	IN	BOOL	Положительный фронт (переход 0→1) на входе En разрешает работу функционального блока PWM.
Ax	IN	INT	Аналоговое входное значение.
PeriodicTime	IN	TIME	Период времени, в течение которого модулируется цифровой выход.
OUT_Q	OUT	BOOL	Q устанавливается или сбрасывается на часть каждого периода времени в соответствии с долей

			стандартизированного значения Ax к диапазону аналоговых значений.

Set Param (Установка параметров)

Дважды щелкните на блоке, чтобы открыть диалоговое окно настройки параметров.



Диалоговое окно содержит следующие поля:

- **Block name** (Имя блока): PWM_0
- **Gain** (Усиление): 1.00
- **Offset** (Смещение): 0
- **Range Min** (Мин. диапазона): 0
- **Range Max** (Макс. диапазона): 1000
- **Periodic time** (Период времени): Поля для ввода часов, минут, секунд, миллисекунд.

Gain (Коэффициент усиления)

Диапазон значений: -10.00 до 10.00

Offset (Смещение)

Диапазон значений: -10,000 до 10,000

PT: Periodic time (Период времени)

Период времени, в течение которого модулируется цифровой выход.

Описание функции (Description of the function)

Функция PWM считывает значение сигнала на аналоговом входе Ax.

Это значение умножается на значение параметра A (Gain). Параметр B (Offset) добавляется к произведению следующим образом:

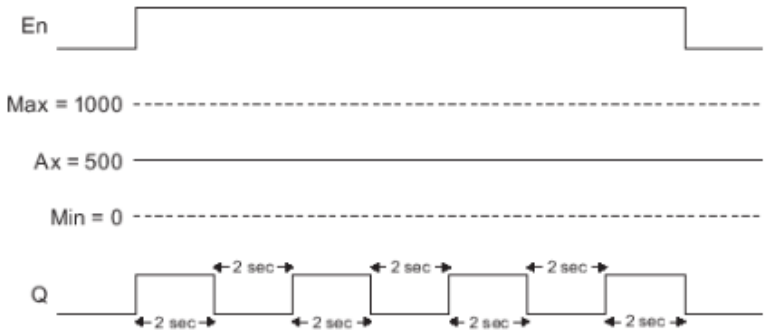
(Ax * Gain) + Offset = Actual value Ax (Фактическое значение Ax)

Функциональный блок вычисляет долю значения Ax относительно диапазона. Блок устанавливает цифровой выход Q в высокий уровень на такую же долю параметра PT

(времени периода) и устанавливает Q в низкий уровень на оставшуюся часть периода времени.

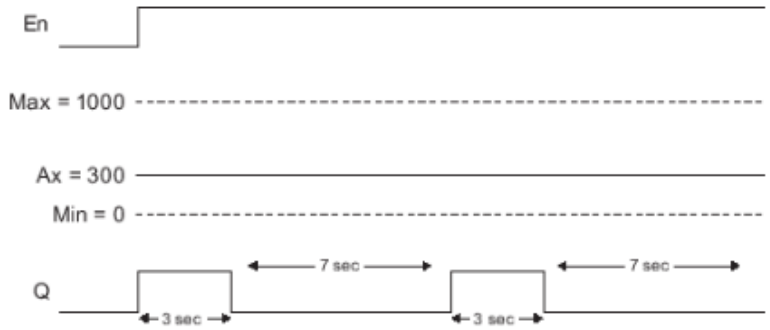
Примеры с временными диаграммами (Examples with Timing Diagrams)
Следующие примеры показывают, как инструкция PWM модулирует цифровой выходной сигнал из аналогового входного значения:

Пример 1
Аналоговое входное значение: 500 (диапазон 0 до 1000)
Период времени T: 4 секунды
Цифровой выход функции PWM составляет 2 секунды высокий уровень, 2 секунды низкий уровень, 2 секунды высокий уровень, 2 секунды низкий уровень и продолжается в этом режиме, пока параметр "En" = высокий уровень.



- Временная диаграмма для Примера 1, показывающая сигналы En, Ax, Q с соответствующими временными интервалами.

Пример 2
Аналоговое входное значение: 300 (диапазон 0 до 1000)
Период времени T: 10 секунд
Цифровой выход функции PWM составляет 3 секунды высокий уровень, 7 секунд низкий уровень, 3 секунды высокий уровень, 7 секунд низкий уровень и продолжается в этом режиме, пока параметр "En" = высокий уровень.



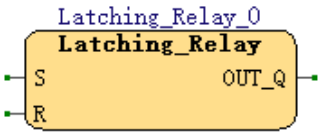
- Временная диаграмма для Примера 2, показывающая сигналы En, Ax, Q с соответствующими временными интервалами.

Правило расчета (Calculation rule)

Q = 1, в течение (Ax – Min) / (Max – Min) от периода времени PT
Q = 0, в течение PT – [(Ax – Min) / (Max – Min)] от периода времени PT.
Примечание: Ax в этом расчете относится к фактическому значению Ax, рассчитанному с использованием Gain и Offset. Min и Max относятся к минимальному и максимальному значениям, указанным для диапазона.

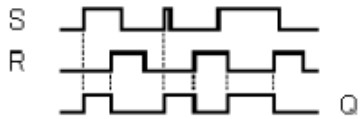
2.15.16 Latching_Relay (Бистабильное реле)

Сигнал на входе S устанавливает выход Q. Сигнал на входе R сбрасывает выход Q.



Имя	Ввод/Вывод	Тип данных	Описание
S	IN	BOOL	Установить выход Q сигналом на входе S (Set — Установка).
R	IN	BOOL	Сбросить выход Q сигналом на входе R (Reset — Сброс). Выход Q сбрасывается, если S и R установлены одновременно (сброс имеет приоритет над установкой).
OUT_Q	OUT	BOOL	Q устанавливается сигналом на входе S и остается установленным до тех пор, пока не будет сброшен сигналом на входе R.

Временная диаграмма (Timing diagram)



- Диаграмма, показывающая работу бистабильного реле. Сигнал S устанавливает OUT_Q в 1. Сигнал R сбрасывает OUT_Q в 0. Одновременный сигнал на S и R приводит к сбросу выхода (приоритет R).

Описание функции (Description of the function)

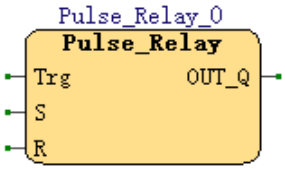
Бистабильное реле Latching_Relay представляет собой простую бистабильную логику памяти. Выходное значение зависит от состояний входов и предыдущего статуса на выходе.

Логическая таблица бистабильного реле (Logic table of the latching relay):

S	R	Q	Remark (Примечание)
0	0	X	Status unchanged (Состояние не изменяется)
0	1	0	Reset (Сброс)
1	0	1	Set (Установка)
1	1	0	Reset (Сброс - приоритет)

2.15.17 Pulse_Relay (Импульсное реле)

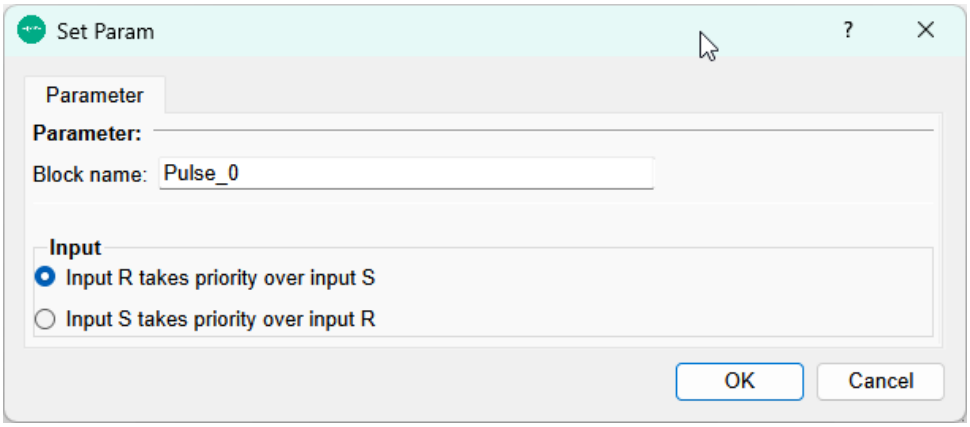
Выход OUT_Q устанавливается и сбрасывается коротким импульсом на входе.



Имя	Ввод/Вывод	Тип данных	Описание
Trg	IN	BOOL	Переключение выхода Q осуществляется импульсом на входе Trg (Trigger — Триггер).
S	IN	BOOL	Импульс на входе S (Set — Установка) переводит выход в логическую 1.
R	IN	BOOL	Импульс на входе R (Reset — Сброс) возвращает выход в логический 0.
OUT_Q	OUT	BOOL	Q включается импульсом на Trg и сбрасывается следующим импульсом на Trg, если оба входа S и R = 0.

Set Param (Установка параметров)

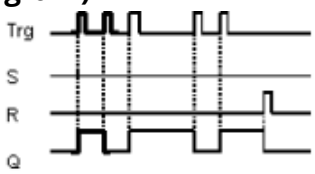
Дважды щелкните на блоке, чтобы открыть диалоговое окно настройки параметров.



Диалоговое окно содержит следующие параметры:

- **Input** (Входы):
  - Input R takes priority over input S (Вход R имеет приоритет над входом S)
  - Input S takes priority over input R (Вход S имеет приоритет над входом R)

Временная диаграмма (Timing diagram)



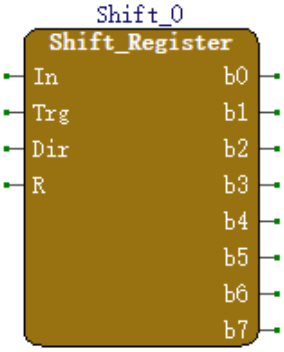
- Временная диаграмма, иллюстрирующая работу импульсного реле. Показаны сигналы Trg, S, R и соответствующий им выход OUT_Q. Демонстрируется переключение выхода по фронту Trg и приоритетная логика при активации входов S и R.

Описание функции (Description of the function)

Состояние выхода OUT_Q изменяется с каждым фронтом 0→1 на входе Trg при условии, что оба входа S и R = 0, то есть выход переключается (включается или выключается).
Вход Trg не влияет на функциональный блок, когда S = 1 или R = 1.
Импульс на входе S устанавливает импульсное реле, то есть выход переводится в логическую 1.
Импульс на входе R сбрасывает импульсное реле в исходное состояние, то есть выход переводится в логический 0.
В зависимости от конфигурации либо вход R имеет приоритет над входом S (сигнал на входе S не имеет эффекта, пока R = 1), либо вход S имеет приоритет над входом R (сигнал на входе R не имеет эффекта, пока S = 1).

2.15.18 Shift_Register (Сдвиговый регистр)

Функция сдвигового регистра Shift_Register считывает входное значение и сдвигает биты. Выходное значение соответствует настроенному биту сдвигового регистра. Направление сдвига можно изменить с помощью специального входа.

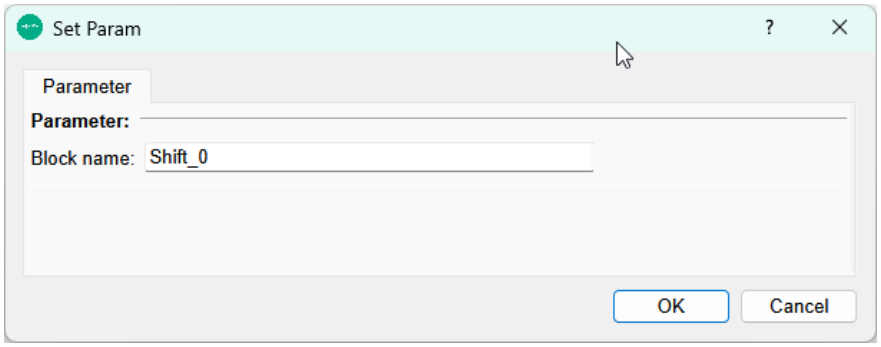


Имя	Ввод/Вывод	Тип данных	Описание
In	IN	BOOL	Функция при запуске считывает значение этого входа.
Trg	IN	BOOL	ФБ запускается положительным фронтом (переход 0→1) на входе Trg (Trigger — Триггер). Переход 1→0 не имеет значения.
Dir	IN	BOOL	Направление сдвига битов сдвигового регистра bo...b7 задается входом Dir: Dir=0 — сдвиг вверх; Dir=1 — сдвиг вниз.
R	IN	BOOL	Функциональный блок (ФБ) сбрасывается, когда вход R (Reset — Сброс) = 1. При сбросе ФБ все биты сдвигового регистра устанавливаются в 0.
b0	OUT	BOOL	Бит 0

b1	OUT	BOOL	Бит 1
…	…	…	…
b7	OUT	BOOL	Бит 7

Set Param (Установка параметров)

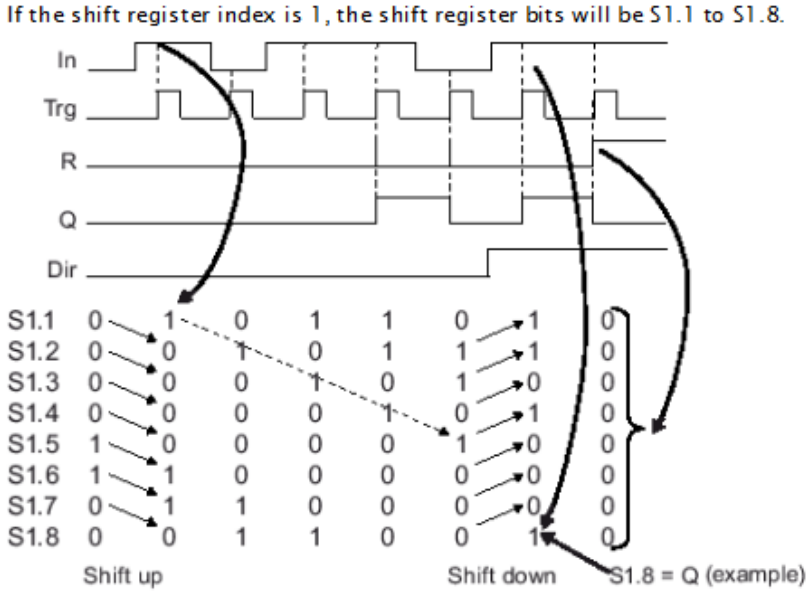
Дважды щелкните на блоке, чтобы открыть диалоговое окно настройки параметров.



Диалоговое окно содержит поле:

- **Shift register bit at the output connector** (Бит сдвигового регистра на выходном коннекторе): Выпадающий список для выбора бита (b0-b7), значение которого будет отображаться на основном выходе Q блока.

Временная диаграмма (Timing diagram)



- Диаграмма, иллюстрирующая работу сдвигового регистра. Показаны сигналы In, Trg, R, Dir и состояния битов регистра (например, $1.1 … $1.8) при сдвиге вверх (Shift up) и вниз (Shift down). Основной выход Q (пример для $1.8) отображает значение выбранного бита.

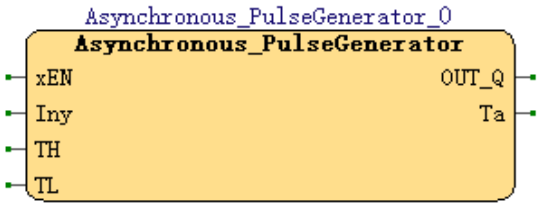
Описание функции (Description of the function)

Это значение записывается в биты сдвигового регистра Sx.1 до Sx.8 в зависимости от установленного направления сдвига:

- Dir = 0 (Сдвиг вверх): Sx.1 принимает значение входа In, предыдущее значение Sx.1 сдвигается в Sx.2, Sx.2 в Sx.3 ... Sx.7 в Sx.8.
- Dir = 1 (Сдвиг вниз): Sx.8 принимает значение входа In; предыдущее значение S1.8 сдвигается в Sx.7, Sx.7 в Sx.6 ... Sx.2 в Sx.1.
- Положительный фронт (переход 0→1) на входе R (Reset — Сброс) сбрасывает сдвиговый регистр. Все биты сдвигового регистра (Sx.1 до Sx.8) и выход Q устанавливаются в 0.
- Q выдает значение настроенного бита сдвигового регистра.

2.15.19 Asynchronous_PulseGenerator (Асинхронный генератор импульсов)

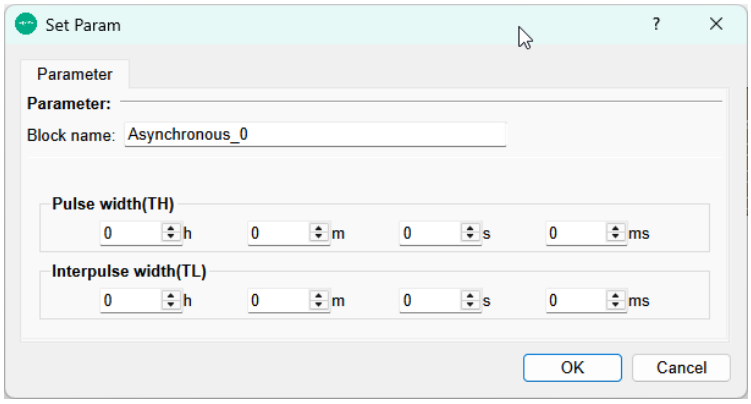
Форма импульса на выходе OUT_Q может быть изменена с помощью настраиваемого коэффициента заполнения (соотношения импульс/пауза).



Имя	Ввод/Вывод	Тип данных	Описание
xEN	IN	BOOL	Разрешение/запрет работы асинхронного генератора импульсов сигналом на входе En.
Inv	IN	BOOL	Вход Inv может использоваться для инверсии выходного сигнала активного асинхронного генератора импульсов.
TH	IN	TIME	Длительность импульса (Time High).
TL	IN	TIME	Длительность паузы (Time Low).
OUT_Q	OUT	BOOL	Q циклически переключается (включается и выключается) с временами импульса TH и паузы TL.
Ta	OUT	TIME	Текущее время таймера.

Set Param (Установка параметров)

Дважды щелкните на блоке, чтобы открыть диалоговое окно настройки параметров.

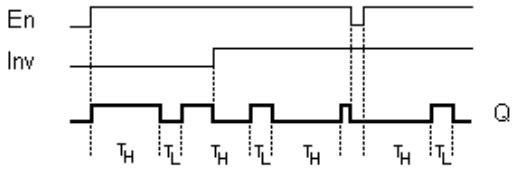


Диалоговое окно содержит поля для настройки:

- **Pulse width(TH)** (Длительность импульса (TH))
- **Interpulse width(TL)** (Длительность паузы (TL))

Вы можете установить соотношение импульс/пауза с помощью параметров TH (Time High — Время высокого уровня) и TL (Time Low — Время низкого уровня).
Вход INV может использоваться для инверсии выходного сигнала. Входной блок INV инвертирует выходной сигнал только в том случае, если блок разрешен с помощью EN.

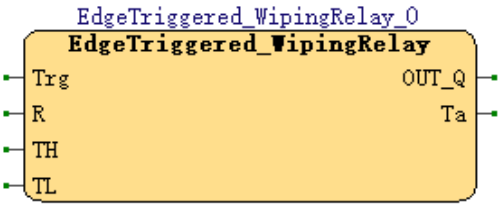
Временная диаграмма (Timing diagram)



- Диаграмма, показывающая циклическую работу генератора. Выход OUT_Q периодически переключается между состоянием 1 (длительность TH) и состоянием 0 (длительность TL). Сигнал Inv инвертирует выходной сигнал, если блок разрешен (EN=1).

2.15.20 EdgeTriggered_WipingRelay (Триггерное реле с задержкой)

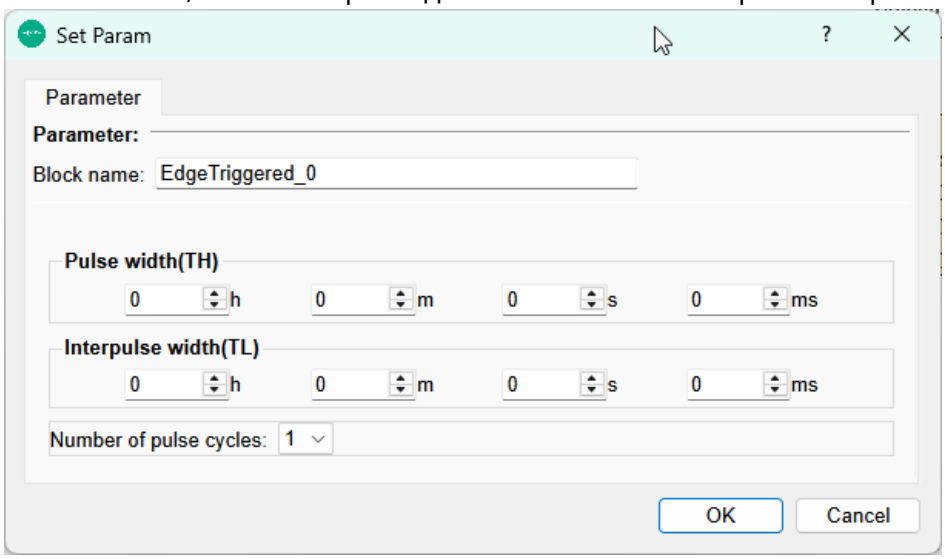
Входной импульс генерирует заданное количество выходных импульсов с определенным коэффициентом заполнения (с возможностью повторного запуска) после истечения настроенного времени задержки.



Имя	Ввод/Вывод	Тип данных	Описание
Trg	IN	BOOL	Временные параметры для Edge-triggered wiping relay запускаются импульсом на входе Trg (Trigger — Триггер).
R	IN	BOOL	Выход и текущее время Ta сбрасываются в 0 сигналом на входе R.
TH	IN	TIME	Длительность импульса (Pulse width).
TL	IN	TIME	Длительность паузы (Interpulse width).
OUT_Q	OUT	BOOL	Выход Q устанавливается, когда время TL истекло, и сбрасывается, когда истекает время TH.
Ta	OUT	TIME	Текущее время таймера.

Set Param (Установка параметров)

Дважды щелкните на блоке, чтобы открыть диалоговое окно настройки параметров.



Диалоговое окно содержит поля для настройки:

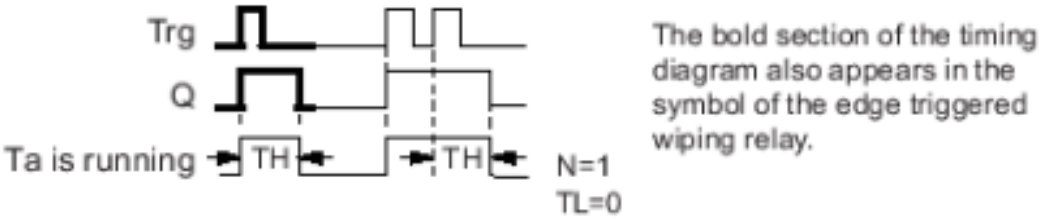
- **Pulse width(TH)** (Длительность импульса (TH))
- **Interpulse width(TL)** (Длительность паузы (TL))
- **Number of pulse cycles** (Количество импульсных циклов): 1

Описание функции (Description of the function)

Изменение на входе Trg на 1 запускает время TL (time low — время низкого уровня). После того как время TL истечет, SystemePLC устанавливает выход Q в 1 на длительность времени TH (time high — время высокого уровня).

Если SystemePLC повторно запускает вход Trg до истечения установленного времени (TL + TH), время Ta сбрасывается, и период импульс/пауза перезапускается.

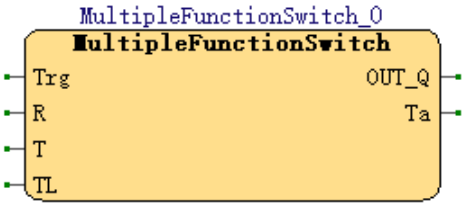
Временная диаграмма (Timing diagram)



- Диаграмма, показывающая работу триггерного реле. Импульс на Trg запускает время TL, после которого следует импульс длительностью TH на выходе OUT_Q. Показан случай с N=1 (один цикл) и TL=0.
- Жирный участок временной диаграммы также отображается в символе Edge triggered wiping relay.

2.15.21 MultipleFunctionSwitch (Многофункциональный переключатель)

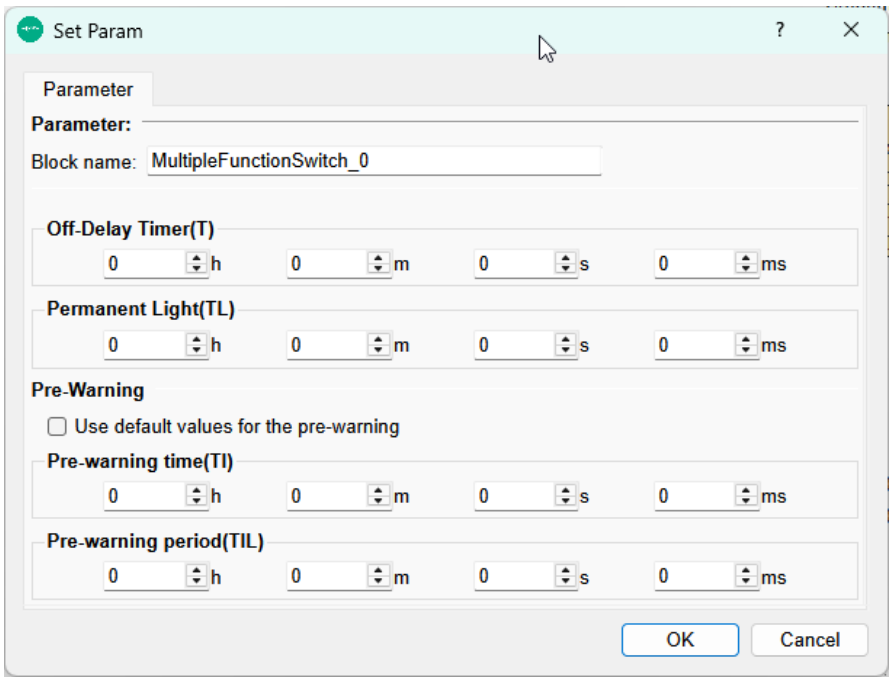
Переключатель с двумя различными функциями: Импульсный переключатель с задержкой выключения и Переключатель (постоянный свет).



Имя	Ввод/Вывод	Тип данных	Описание
Trg	IN	BOOL	Сигнал на входе Trg (Trigger — Триггер) устанавливает выход Q (постоянный свет) или сбрасывает Q с задержкой выключения. Когда активно, выход Q может быть сброшен сигналом на входе Trg.
R	IN	BOOL	Сигнал на входе R сбрасывает текущее время Ta и сбрасывает выход.
T	IN	TIME	Определяет время задержки выключения. Выход сбрасывается (переход 1→0), когда время T истекает.
TL	IN	TIME	Определяет период, в течение которого вход должен быть установлен, чтобы включить функцию постоянного света.
OUT_Q	OUT	BOOL	Выход Q устанавливается сигналом на входе Trg и снова сбрасывается после истечения настроенного времени и в зависимости от длительности импульса на входе Trg, или сбрасывается другим сигналом на входе Trg.
Ta	OUT	TIME	Текущее время таймера.

Set Param (Установка параметров)

Дважды щелкните на блоке, чтобы открыть диалоговое окно настройки параметров.



Диалоговое окно содержит поля для настройки:

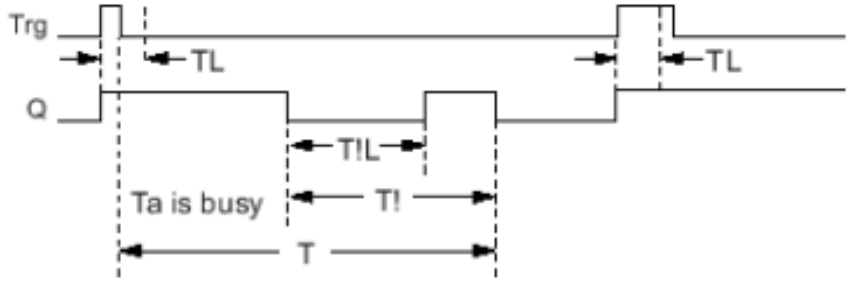
- **Off-Delay Timer(T)** (Таймер задержки выключения (T))
- **Permanent Light(TL)** (Постоянный свет (TL))
- **Pre-Warning** (Предупреждение): Опция использования значений по умолчанию
- **Pre-warning time(TI)** (Время предупреждения (TI))
- **Pre-warning period(TIL)** (Период предупреждения (TIL))
- **Retentivity** (Сохраняемость): Опция сохранения состояния в памяти

T: определяет время задержки выключения. Выход сбрасывается (переход 1→0), когда время T истекает.
TL: определяет период, в течение которого вход должен быть установлен, чтобы включить функцию постоянного света.
TI: определяет время задержки включения для предупреждения.
TIL: определяет длительность периода предупреждения.
Retentivity on = статус сохраняется в памяти.

Описание функции (Description of the function)

Выход OUT_Q устанавливается в 1 при переходе сигнала 0→1 на Trg.
Если выход Q = 0, и вход Trg установлен в высокий уровень как минимум на длительность TL, функция постоянного освещения активируется, и выход Q устанавливается соответствующим образом.
Время задержки выключения T запускается, когда статус на входе Trg изменяется на 0 до того, как время TL истекло.
Выход Q сбрасывается, когда Ta = T.
Вы можете выдать сигнал предупреждения перед истечением времени задержки выключения (T - TI), который сбрасывает Q на длительность времени предупреждения TIL.
Последующий сигнал на входе Trg всегда сбрасывает T и выход Q.

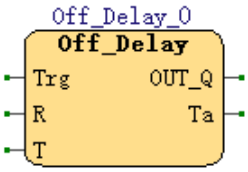
Временная диаграмма (Timing diagram)



- Диаграмма, показывающая работу переключателя. Короткий импульс на Trg активирует режим задержки выключения. Длительный импульс (больше TL) активирует режим постоянного света. Показаны времена T, TL, TI и соответствующие состояния выхода OUT_Q.

2.15.22 Off_Delay (Таймер задержки выключения)

Выход с задержкой выключения сбрасывается после истечения заданного времени.

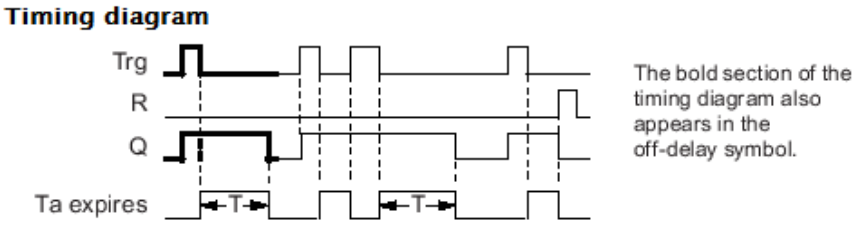


Имя	Ввод/Вывод	Тип данных	Описание
Trg	IN	BOOL	Запуск времени задержки выключения отрицательным фронтом (переход 1→0) на входе Trg (Trigger — Триггер).
R	IN	BOOL	Сброс времени задержки выключения и установка выхода в 0 через вход R (Reset — Сброс). Сброс имеет приоритет над Trg.
T	IN	TIME	Выход выключается по истечении времени задержки T (переход выходного сигнала 1→0).
OUT_Q	OUT	BOOL	Q включается на длительность времени T после триггера на входе Trg.
Ta	OUT	TIME	Текущее время таймера.

Описание функции (Description of the function)

Переход 0→1 на входе Trg устанавливает выход OUT_Q в 1 мгновенно.
Когда происходит переход 1→0 на входе Trg, текущее время T перезапускается, и выход остается установленным. Когда Ta достигает значения, заданного T (Ta=T), выход Q сбрасывается в 0 (задержка выключения).
Одиночный триггер на входе Trg перезапускает время Ta.
Время Ta и выходы могут быть сброшены до окончания времени Ta с помощью входа R (Reset — Сброс).

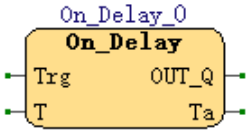
Временная диаграмма (Timing diagram)



- Диаграмма, показывающая работу таймера. При переходе 1→0 на Trg запускается время Ta. Выход OUT_Q остается активным до истечения времени T. Сигнал R сбрасывает таймер досрочно.
- Жирный участок диаграммы отображается в символе таймера задержки выключения.

2.15.23 On_Delay (Таймер задержки включения)

Выход включается только после истечения настроенного времени задержки.



Имя	Ввод/Вывод	Тип данных	Описание
Trg	IN	BOOL	Вход Trg (Trigger — Триггер) запускает время задержки включения.
T	IN	TIME	Представляет время задержки включения, после которого выход включается (переход выходного сигнала 0→1).
OUT_Q	OUT	BOOL	Q включается после истечения указанного времени T при условии, что Trg все еще установлен.
Ta	OUT	TIME	Текущее время таймера.

Описание функции (Description of the function)

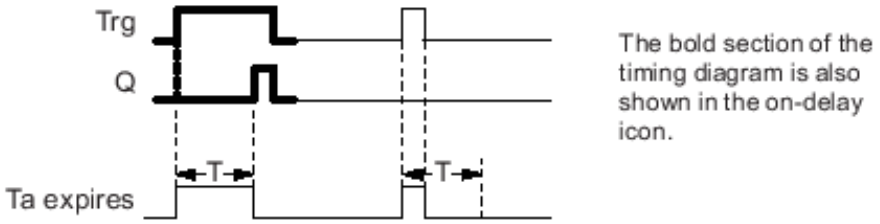
Переход 0→1 на входе Trg запускает время Ta.

Если статус на входе Trg остается равным 1 как минимум в течение настроенного времени T, выход устанавливается в 1 по истечении этого времени (выходной сигнал включения следует за входным сигналом с задержкой).

Время сбрасывается, если статус на входе Trg снова изменяется на 0 до истечения времени T.

Выход сбрасывается в 0, когда вход Trg равен 0.

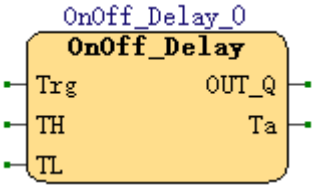
Временная диаграмма (Timing diagram)



- Диаграмма, показывающая работу таймера. При активации Trg начинается отсчет времени T. Если Trg активен все время T, то по его истечении OUT_Q устанавливается в 1. Если Trg сбрасывается до истечения T, таймер прерывается.
- Жирный участок диаграммы отображается в символе таймера задержки включения.

2.15.24 OnOff_Delay (Таймер задержки включения/выключения)

Функциональный блок OnOff_Delay может устанавливать выход после настроенного времени задержки включения, а затем снова сбрасывать его после истечения второго настроенного времени.



Имя	Ввод/Вывод	Тип данных	Описание
Trg	IN	BOOL	Запуск задержки включения положительным фронтом (переход 0→1) на входе Trg (Trigger — Триггер). Запуск задержки выключения отрицательным фронтом (переход 1→0).
TH	IN	TIME	Время задержки включения для выхода (переход выходного сигнала 0→1).
TL	IN	TIME	Время задержки выключения для выхода (переход выходного сигнала 1→0).
OUT_Q	OUT	BOOL	Q включается по истечении настроенного времени TH, если Trg все еще установлен. Выключается снова по истечении времени TL и если Trg не был установлен снова.
Ta	OUT	TIME	Текущее время таймера.

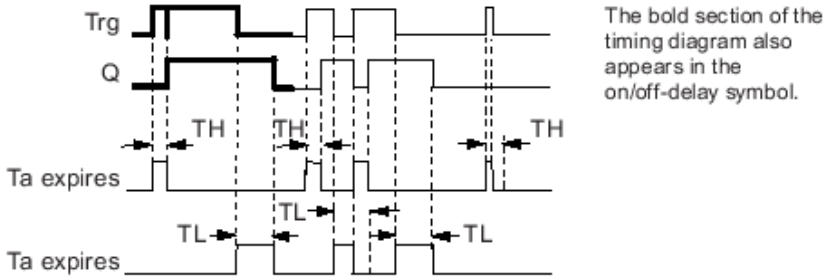
Описание функции (Description of the function)

Переход 0→1 на входе Trg запускает время TH.

Если состояние входа Trg равно 1 как минимум в течение настроенного времени TH, выход устанавливается в логическую 1 в конце этого времени (выход включается с задержкой относительно входного сигнала).

Если состояние входа Trg сбрасывается в 0 до окончания этого времени, время TH сбрасывается.
Переход 1→0 на выходе запускает время TL.
Если состояние входа Trg остается равным 0 как минимум в течение настроенной длительности времени TL, выход сбрасывается в 0 в конце этого времени (выход выключается с задержкой относительно входного сигнала).
Если состояние входа Trg возвращается в 1 до окончания времени TL, время TL сбрасывается.

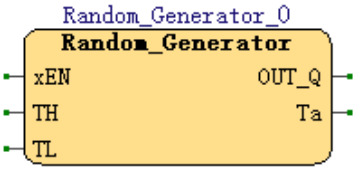
Временная диаграмма (Timing diagram)



- Диаграмма, показывающая работу таймера. При активации Trg начинается отсчет времени TH, после которого OUT_Q устанавливается в 1. При деактивации Trg начинается отсчет времени TL, после которого OUT_Q сбрасывается в 0. Прерывание Trg во время отсчета TH или TL сбрасывает соответствующий таймер.
- Жирный участок диаграммы отображается в символе таймер задержки включения/выключения.

2.15.25 Random_Generator (Случайный генератор)

Выход OUT_Q случайного генератора переключается в пределах настраиваемого времени.



Имя	Ввод/Вывод	Тип данных	Описание
xEN	IN	BOOL	Положительный фронт (переход 0→1) на входе разрешения En (Enable) запускает задержку включения для случайного генератора. Отрицательный фронт (переход 1→0) запускает задержку выключения для случайного генератора.
TH	IN	TIME	Время задержки включения определяется случайным образом и находится между 0 с и TH.
TL	IN	TIME	Время задержки выключения определяется случайным образом и находится между 0 с и TL.

OUT_Q	OUT	BOOL	Q устанавливается по истечении задержки включения, если En все еще установлен. Сбрасывается, когда время задержки выключения истекло и En не был установлен снова.
Ta	OUT	TIME	Текущее время таймера.

Описание функции (Description of the function)

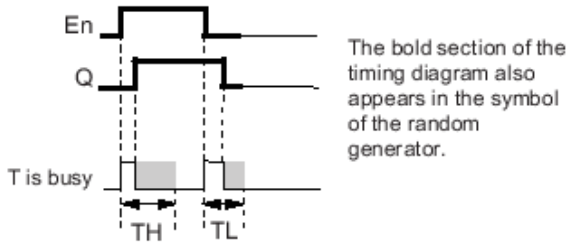
При переходе 0→1 на входе En устанавливается и запускается случайное время (время задержки включения) между 0 с и TH. Если статус на входе En равен 1 как минимум в течение времени задержки включения, выход устанавливается в 1 по истечении этого времени задержки включения.

Время сбрасывается, если статус на входе En сбрасывается в 0 до истечения времени задержки включения.

Когда вход En сброшен в 0, устанавливается и запускается случайное время (время задержки выключения) между 0 с и TL.

Если статус на входе En равен 0 как минимум в течение времени задержки выключения, выход Q сбрасывается в 0 по истечении времени задержки выключения.

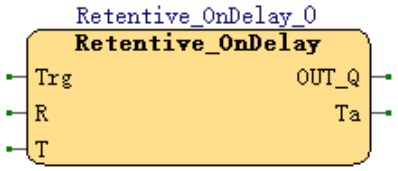
Временная диаграмма (Timing diagram)



- Диаграмма, показывающая работу генератора. Включение и выключение выхода OUT_Q происходит через случайные интервалы времени в пределах, заданных TH и TL.
- Жирный участок диаграммы отображается в символе случайного генератора.

2.15.26 Retentive_OnDelay (Таймер включения с памятью)

Одиночный импульс на входе запускает настраиваемое время. Установить выход по истечении этого времени.



Имя	Ввод/Вывод	Тип данных	Описание
Trg	IN	BOOL	Запуск времени задержки включения через вход Trg (Trigger — Триггер).

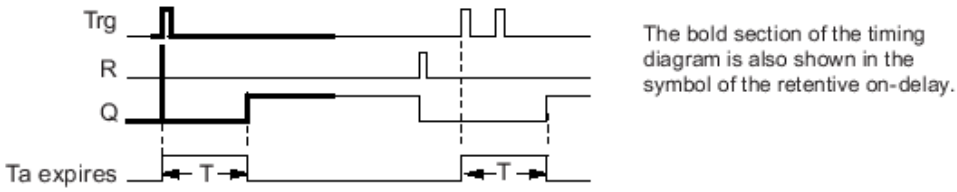
R	IN	BOOL	Сброс времени задержки включения и сброс выхода в 0 через вход R (Reset — Сброс). Сброс имеет приоритет над Trg.
T	IN	TIME	T — время задержки включения для выхода (переход выходного сигнала 0→1).
OUT_Q	OUT	BOOL	Q включается по истечении времени T.
Ta	OUT	TIME	Текущее время таймера.

Описание функции (Description of the function)

Сигнальный переход 0→1 на входе Trg запускает текущее время Ta. Когда Ta достигает времени T, выход Q устанавливается в 1.

Выходы и время Ta сбрасываются в 0 только в том случае, если сигнал на входе R равен 1. Если удержание не установлено, выход Q и время истечения сбрасываются после сбоя питания.

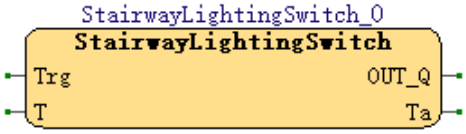
Временная диаграмма (Timing diagram)



- Диаграмма, показывающая работу таймера. Импульс на Trg запускает отсчет времени T. Выход OUT_Q устанавливается в 1 по истечении T, даже если Trg уже сброшен. Сброс происходит только по сигналу R.
- Жирный участок диаграммы отображается в символе таймера включения с памятью.

2.15.27 StairwayLightingSwitch (Лестничный выключатель)

Фронт входного импульса запускает настраиваемое время. Выход сбрасывается, когда это время истекло. Перед истечением этого времени может быть выдано предупреждение о выключении.



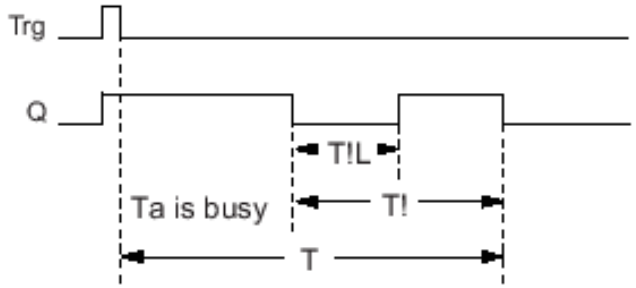
Имя	Ввод/Вывод	Тип данных	Описание
Trg	IN	BOOL	Запуск времени (задержки выключения) для лестничного выключателя осуществляется сигналом на входе Trg (Trigger — Триггер).
T	IN	TIME	Выход сбрасывается (переход 1→0), когда время задержки выключения T истекло.

OUT_Q	OUT	BOOL	Q сбрасывается после истечения времени T. Перед истечением этого времени может быть выдан предупреждающий сигнал.
Ta	OUT	TIME	Текущее время таймера.

Описание функции (Description of the function)

Выход OUT_Q устанавливается в 1 при переходе сигнала 0→1 на входе Trg. Переход 1→0 на входе Trg запускает текущее время, и выход OUT_Q остается установленным.
Выход OUT_Q сбрасывается в 0, когда Ta достигает времени T. До истечения времени задержки выключения (T - TI) можно выдать предупреждение, которое сбрасывает Q на длительность времени предупреждения о выключении TIL.
Ta перезапускается (опционально) при следующем переходе высокий/низкий уровень на входе Trg и если Ta истекает.

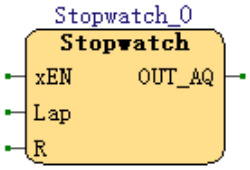
Временная диаграмма (Timing diagram)



- Диаграмма, показывающая работу лестничного выключателя. Импульс на Trg включает свет (OUT_Q=1). Начинается отсчет времени T. Перед окончанием T активируется предупреждение (TIL). По истечении T свет выключается. Новый импульс на Trg перезапускает таймер.

2.15.28 Stopwatch (Секундомер)

Секундомер Stopwatch фиксирует время, прошедшее с момента его активации.



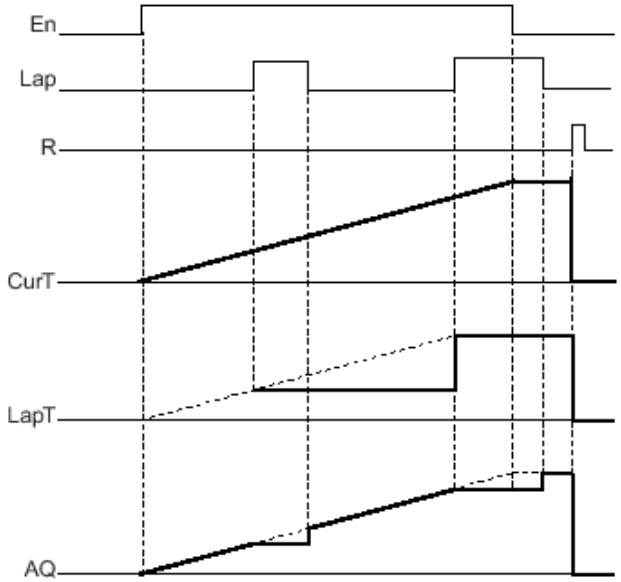
Имя	Ввод/Вывод	Тип данных	Описание
xEN	IN	BOOL	En (Enable — Разрешение) является мониторинговым входом. Устанавливает текущее прошедшее время в 0 и начинает отсчет времени, когда En переходит из 0 в 1. Когда En переходит из 1 в 0, прошедшее время замораживается.

Lap	IN	BOOL	Положительный фронт (переход 0→1) на входе Lap приостанавливает секундомер и устанавливает выход на время круга. Отрицательный фронт (переход 1→0) на входе Lap возобновляет секундомер и устанавливает выход на текущее прошедшее время.
R	IN	BOOL	Сигнал на входе R (Reset — Сброс) очищает текущее прошедшее время и время круга.
OUT_AQ	OUT	TIME	Выход AQ выдает значение текущего прошедшего времени при отрицательном фронте (переход 1→0) на входе Lap и выдает значение времени круга при положительном фронте (переход 0→1) на входе Lap. Положительный фронт (переход 0→1) сбрасывает значение на выходе AQ в 0.

Логика работы:

- Когда En = 1, текущее время увеличивается.
- Когда En = 0, текущий счет времени приостанавливается.
- Когда En = 1 и Lap = 0, выход AQ выдает значение текущего прошедшего времени.
- Когда En = 1 и Lap = 1, текущее время продолжает увеличиваться, но выход AQ выдает значение времени круга.
- Когда En = 0 и Lap = 1, выход AQ выдает значение времени круга.
- Когда En = 0 и Lap = 0, выход AQ выдает значение последнего текущего времени.
- Когда R = 1, и текущее время, и время круга сбрасываются.

Временная диаграмма (Timing diagram)

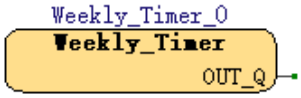


- Диаграмма, показывающая работу секундомера. Отсчет времени начинается при En=1. Сигнал Lap фиксирует промежуточное время без остановки основного таймера. Сигнал

R полностью сбрасывает измерения. Показаны взаимосвязи между сигналами En, Lap, R и выходными значениями CurT (текущее время), LapT (время круга) и AQ.

2.15.29 Weekly_Timer (Недельный таймер)

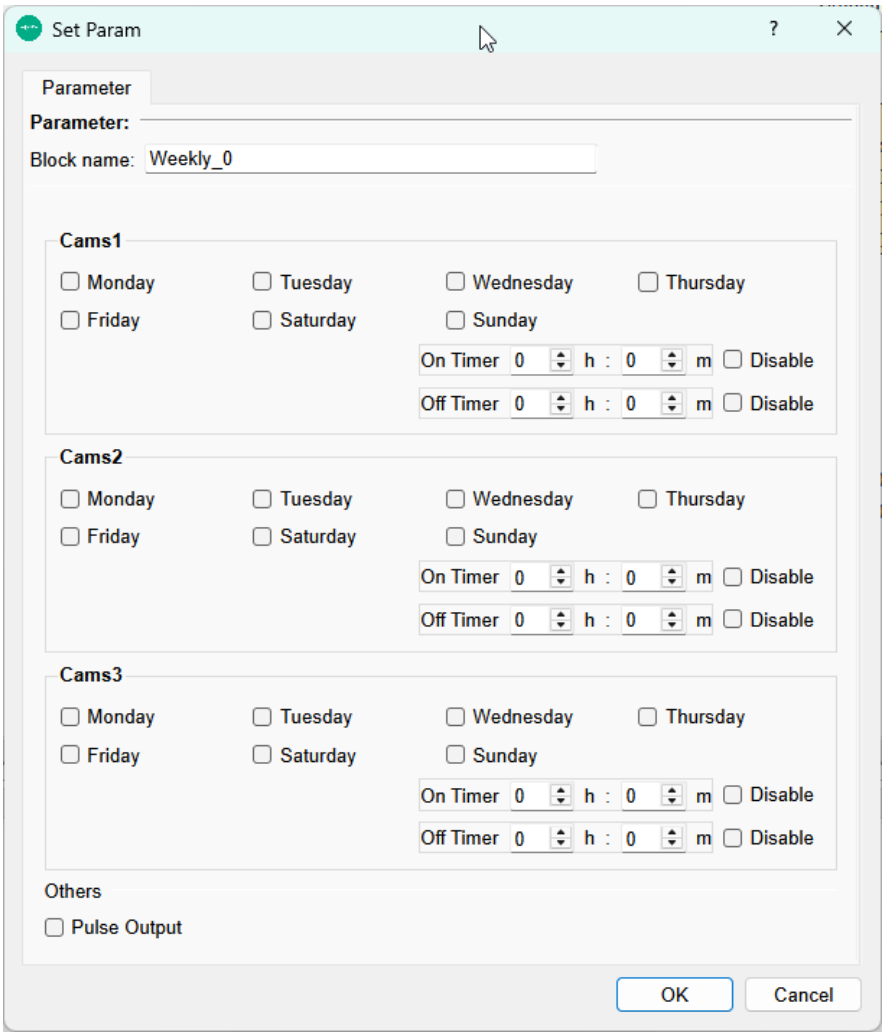
Выход управляется с помощью настраиваемой даты включения/выключения. Функция поддерживает любые комбинации дней недели.



Имя	Ввод/Вывод	Тип данных	Описание
OUT_Q	OUT	BOOL	Q устанавливается, когда настроенный кулачок активирован.

Set Param (Установка параметров)

Дважды щелкните на блоке, чтобы открыть диалоговое окно настройки параметров.

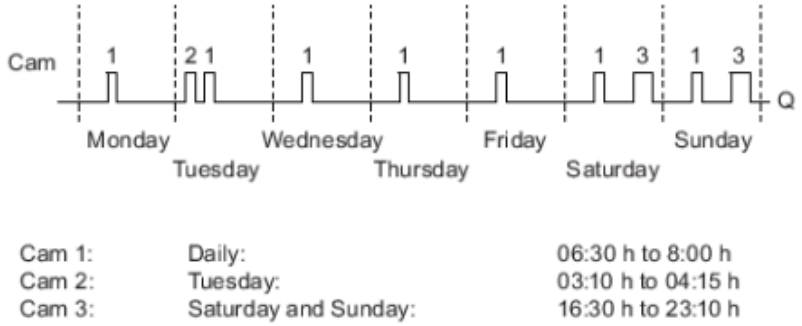


Диалоговое окно содержит настройки для трех кулачков (Cams1-3):

- **Дни недели:** Monday, Tuesday, Wednesday, Thursday, Friday, Saturday, Sunday
- **On Timer** (Таймер включения): время включения
- **Off Timer** (Таймер выключения): время выключения
- **Disable** (Отключить): опция отключения таймера
- **Pulse Output** (Импульсный выход): опция импульсного выхода

Временная диаграмма (Timing diagram)

Три практических примера.



Диаграмма, показывающая примеры расписаний:

- **Cam 1:** Ежедневно: с 06:30 до 8:00
- **Cam 2:** Вторник: с 03:10 до 04:15
- **Cam 3:** Суббота и Воскресенье: с 16:30 до 23:10

Описание функции (Description of the function)

Каждый недельный таймер Weekly_Timer оснащен тремя кулачками. Вы можете настроить временной гистерезис для каждого отдельного кулачка. На кулачках вы устанавливаете гистерезис включения и выключения. Недельный таймер устанавливает выход в определенное время, если он еще не установлен.

Недельный таймер сбрасывает выход в время выключения, если вы настроили время выключения, или в конце цикла, если вы указали импульсный выход. Конфликт возникает в недельном таймере, когда время включения и время выключения на другом кулачке совпадают. В этом случае кулачок 3 имеет приоритет над кулачком 2, а кулачок 2 имеет приоритет над кулачком 1.

Состояние переключения недельного таймера определяется статусом на кулачках No1, No2 и No3.

Время включения (On-times)

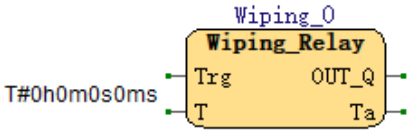
Время включения - это любое время между 00:00 и 23:59. Вы также можете настроить время включения как импульсный сигнал. Блок таймера будет активирован в указанное время на один цикл, а затем выход будет сброшен. Время выключения в этом случае отключено, так как не применимо.

Особенности настройки (Special characteristics to note when configuring)

Окно свойств блока предлагает вкладку для каждого из трех кулачков. Здесь вы можете установить день недели для каждого кулачка. Каждая вкладка дополнительно предлагает возможность определения времени включения и выключения для каждого кулачка в единицах часов и минут. Следовательно, самый короткий цикл переключения составляет одну минуту. Также на каждой вкладке у вас есть возможность указать импульсный выход для кулачка.
Вы можете отключать время включения и выключения индивидуально. Вы можете достичь циклов переключения, охватывающих более одного дня, например, установив время включения для кулачка 1 на понедельник 7:00 и время выключения кулачка 2 на среду 13:07, отключив время включения для кулачка 2.

2.15.30 Wiping_Relay (Импульсное реле)

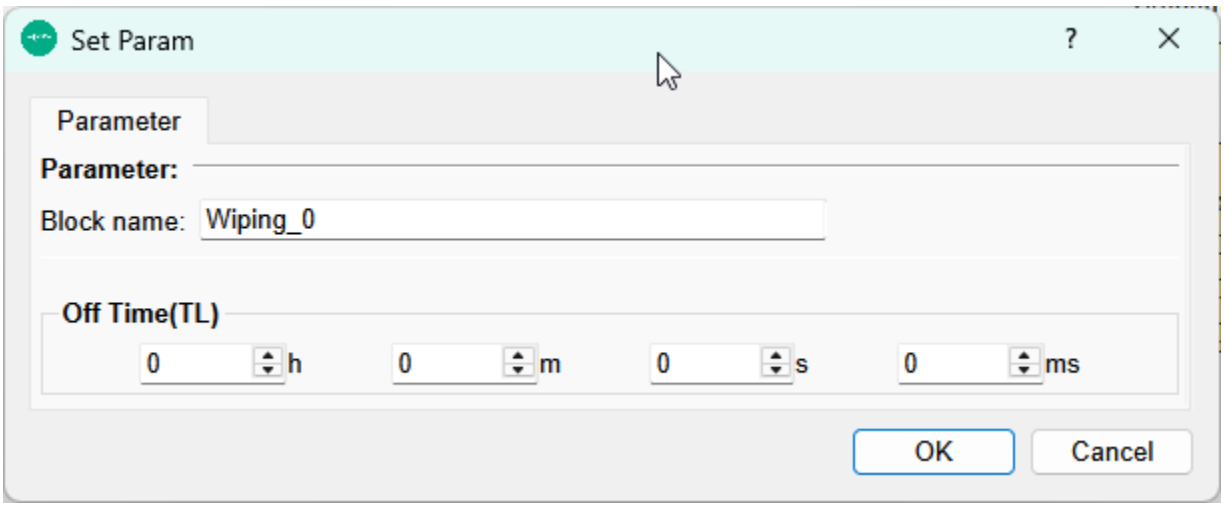
Входной сигнал генерирует выходной сигнал настраиваемой длительности.



Имя	Ввод/Вывод	Тип данных	Описание
Trg	IN	BOOL	Запуск времени для импульсного реле осуществляется сигналом на входе Trg (Trigger — Триггер). Положительный фронт (0→1) запускает время и устанавливает выход Q в 1.
T	IN	TIME	T - длительность выходного импульса. Время, в течение которого выход Q остается активным после запуска. После истечении T выход Q сбрасывается 1→0.
OUT_Q	OUT	BOOL	Импульс на Trg устанавливает выход Q=1. Выход остается активным до тех пор, пока время T не истечет и если Trg = 1 в течение этого времени. Переход 1→0 на Trg до истечения T также сбрасывает выход в 0.
Ta	OUT	TIME	Текущее значение времени. Отсчет времени с момента последнего запуска.

Set Param (Установка параметров)

Дважды щелкните на блоке, чтобы открыть диалоговое окно настройки параметров.



Диалоговое окно содержит поле:

- **Off Time(TL)** (Время выключения(TL))

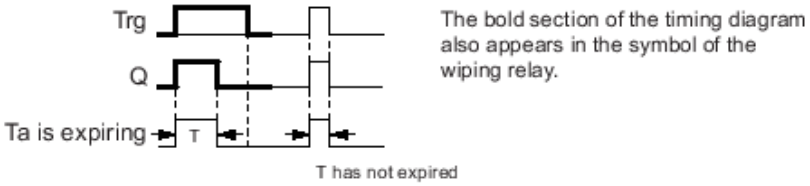
Описание функции (Description of the function)

Входной сигнал Trg = 1 устанавливает выход Q в 1. Сигнал также запускает время Ta, при этом выход остается установленным.

Когда Ta достигает значения, определенного в T (Ta=T), выход Q сбрасывается в состояние 0 (импульсный выход).

Если сигнал на входе Trg изменяется с 1 на 0 до истечения этого времени, выход немедленно сбрасывается с 1 на 0.

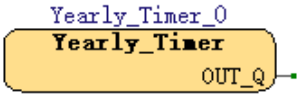
Временная диаграмма (Timing diagram)



- Диаграмма, показывающая работу импульсного реле. Импульс на Trg устанавливает OUT_Q в 1. Если Trg остается активным, выход сбрасывается после истечения времени T. Если Trg сбрасывается раньше, выход сбрасывается немедленно.

2.15.31 Yearly_Timer (Годовой таймер)

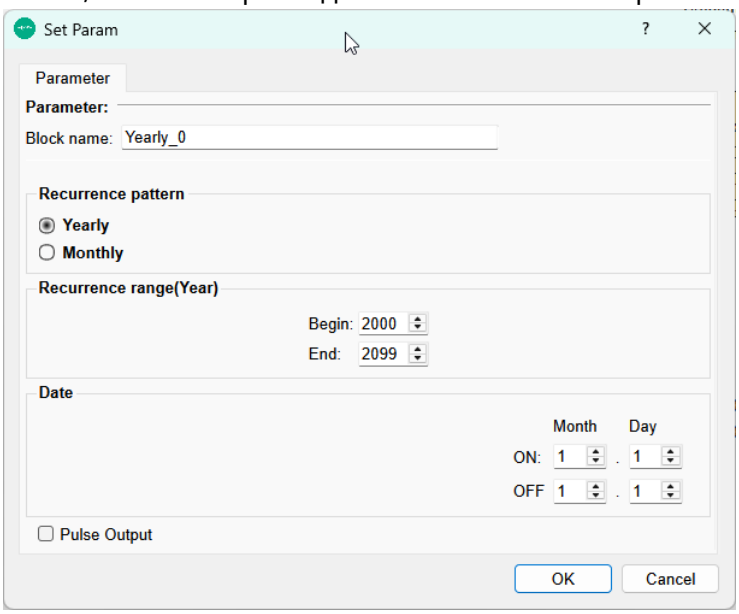
Выход управляется с помощью настраиваемой даты включения/выключения. Вы можете настроить таймер для активации по годичной, месячной основе или основе, определяемой пользователем. В любом режиме вы также можете настроить таймер на импульсный выход в течение определенного периода времени.



Имя	Ввод/Вывод	Тип данных	Описание
OUT_Q	OUT	BOOL	Выходной сигнал. Устанавливается в 1 при активации настроенного задания (cam), когда текущее время попадает в диапазон работы таймера.

Set Param (Установка параметров)

Дважды щелкните на блоке, чтобы открыть диалоговое окно настройки параметров.



Диалоговое окно содержит следующие параметры:

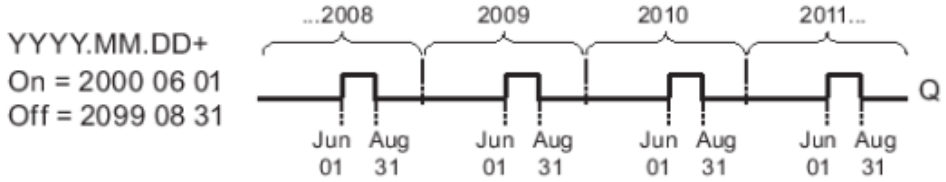
- **Recurrence pattern** (повторяющийся паттерн)
  - **Monthly** (Месячный)
  - **Yearly** (Годовой)
- **Recurrence range Year** (диапазон повторяемости Год)
  - **Begin** (начало): 2000
  - **End** (конец): 2099
- **Date (дата)**
  - **ON** (Время включения) - поля ввода:
    - **Month** (Месяц) [1-12]
    - **Day** (День) [1-31]
  - **OFF** (Время выключения) - поля ввода:
    - **Month** (Месяц) [1-12]
    - **Day** (День) [1-31]
- **Pulse Output** (Импульсный выход) - флажок

Описание функции (Description of the function)

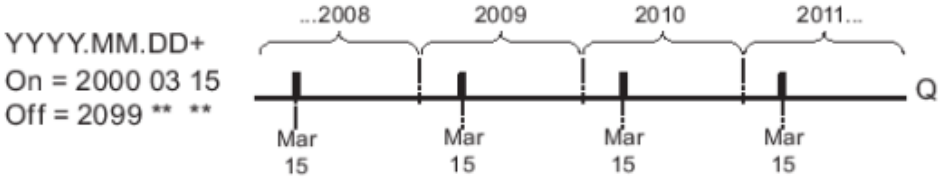
Годовой таймер Yearly_Timer устанавливает и сбрасывает выход в определенные даты включения и выключения. Установки и сбросы выполняются в 00:00. Если вашему

приложению требуется другое время, используйте недельный таймер вместе с годовым таймером в вашей схемной программе.
On Time (Время включения) определяет месяц и день, когда таймер устанавливается. Off Time (Время выключения) определяет месяц и день, когда выход снова сбрасывается. Для времени включения и выключения обратите внимание на порядок полей: первое поле определяет год, второе — месяц и третье — день.
Когда вы выбираете флажок Monthly (Месячный), выход таймера включается каждый месяц в указанный день времени начала и остается включенным до указанного дня времени выключения. On Year (Год начала) определяет начальный год, в котором таймер активируется. Off Year (Год окончания) определяет последний год, в котором таймер выключается. Максимальный год — 2099.
Если вы выбираете флажок Yearly (Годовой), выход таймера включается каждый год в указанные месяц и день времени начала и остается включенным до указанных месяца и дня времени выключения. On Year определяет начальный год, в котором таймер активируется. Off Year определяет последний год, в котором таймер выключается. Максимальный год — 2099.
Если вы выбираете флажок Pulse Output (Импульсный выход), выход таймера включается в указанное время включения на один цикл, а затем выход таймера сбрасывается. Вы можете выбрать импульсный режим таймера на месячной или годовой основе, или всего один раз.
Если вы не выбираете ни один из флажков Monthly, Yearly или Pulse, вы можете определить конкретный временной период с помощью On Time и Off Time. Он может охватывать любой выбранный вами временной период.
Для технологического действия, которое должно включаться и выключаться несколько раз, но в нерегулярное время в течение года, вы можете определить несколько годовых таймеров с выходами, соединенными через функциональный блок OR (ИЛИ).

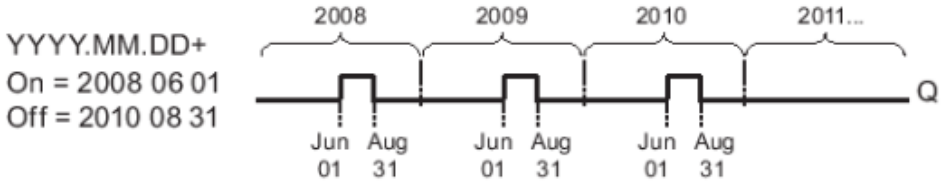
Временная диаграмма (Timing diagram)
Пример 1: Выбран годовой режим, On Time = 2000.06.01, Off Time = 2099.08.31. Каждый год 1 июня выход таймера включается и остается включенным до 31 августа.



Пример 2: Выбран годовой режим, выбран Pulse Output, On Time = 2000.03.15, Off Time = 2099**...** Каждый год 15 марта таймер включается на один цикл.

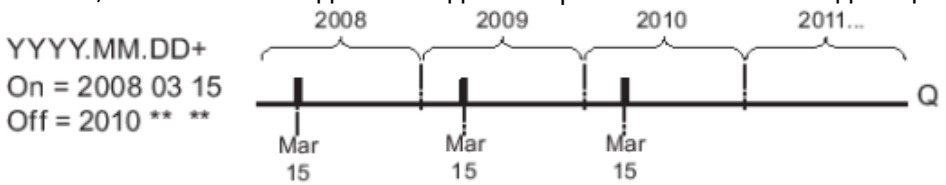


Пример 3: Выбран годовой режим, On Time = 2008.06.01, Off Time = 2010.08.31. 1 июня 2008, 2009 и 2010 годов выход таймера включается и остается включенным до 31 августа.



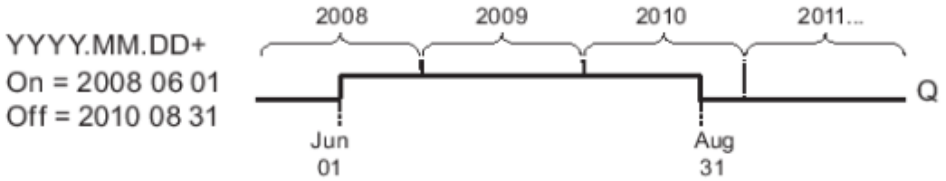
- Диаграмма для примера 3, показывающая периоды активности с июня по август 2008, 2009 и 2010 годов.

Пример 4: Выбран годовой режим, выбран Pulse Output, On Time = 2008.03.15, Off Time = 2010**..** 15 марта 2008, 2009 и 2010 годов выход таймера включается на один цикл.



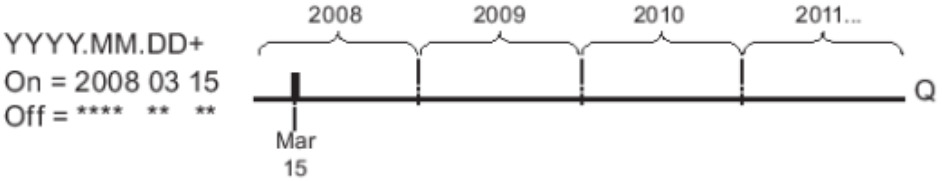
- Диаграмма для примера 4, показывающая импульсы в марте 2008, 2009 и 2010 годов.

Пример 5: Месячный режим не выбран, годовой режим не выбран, On Time = 2008.06.01, Off Time = 2010.08.31. 1 июня 2008 года выход таймера включается и остается включенным до 31 августа 2010 года.



- Диаграмма для примера 5, показывающая один продолжительный период активности с 2008 по 2010 год.

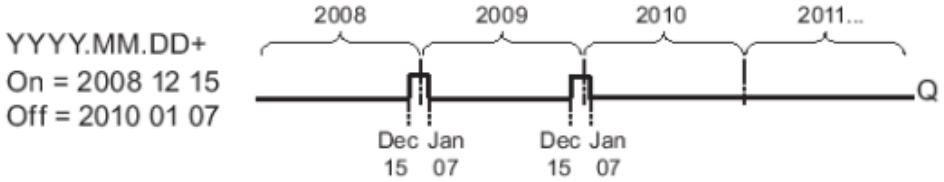
Пример 6: Месячный режим не выбран, годовой режим не выбран, выбран Pulse Output, On Time = 2008.03.15, Off Time = *..**. 15 марта 2008 года таймер включается на один цикл. Поскольку таймер не имеет месячного или годового действия, выход таймера импульсирует только один раз в указанное время включения.



- Диаграмма для примера 6, показывающая одиночный импульс в 2008 году.

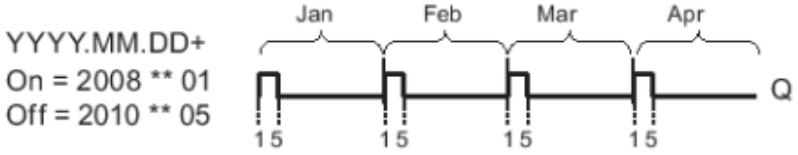
Пример 7: Выбран годовой режим, On Time = 2008.12.15, Off Time = 2010.01.07. 15 декабря 2008 и 2009 годов выход таймера включается и остается включенным до 7 января

следующего года. Когда выход таймера выключается 7 января 2010 года, он НЕ включается снова следующего 15 декабря.



- Диаграмма для примера 7, показывающая периоды активности с декабря по январь 2008/2009 и 2009/2010 годов.

Пример 8: Выбран месячный режим, On Time = 2008.**.01, Off Time = 2010.**.05. Начиная с 2008 года, в первый день каждого месяца выход таймера включается и выключается на пятый день месяца. Таймер продолжает работать в этом режиме до последнего месяца 2010 года.



- Диаграмма для примера 8, показывающая циклы активности с 1 по 5 число каждого месяца в период с 2008 по 2010 год.

3. Коды ошибок

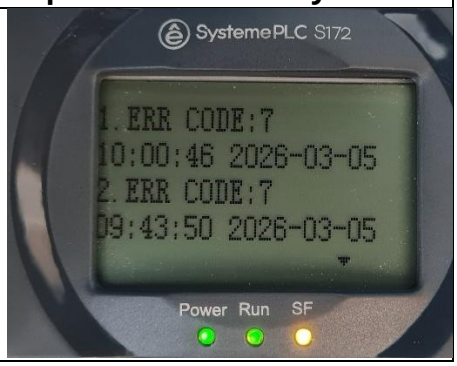
3.1 Внутренние коды ошибок ПО SystemePLC Studio

Принципы обработки ошибок в ПЛК

1. При возникновении ошибки, ПЛК формирует соответствующий ей код, а также присваивает метку времени возникновения ошибки.
2. Код ошибки можно просмотреть разными способами на экране ПЛК (при его наличии) или из ПО SystemePLC Studio.
3. Код ошибки и метку времени можно обрабатывать в программе пользователя.

Просмотр кода ошибки на дисплее ПЛК (только чтение)

1. На системном экране ПЛК в нижней строке (при наличии дисплея) отображается короткое описание возникшей ошибки или «No Faults» при ее отсутствии. Описание соответствует коду ошибки (см. Таблица кодов ошибок SystemePLC Studio).
2. На экране ПЛК «Alarm history» (при наличии дисплея) отображается журнал аварийных событий из 5 последних кодов ошибки с метками времени.

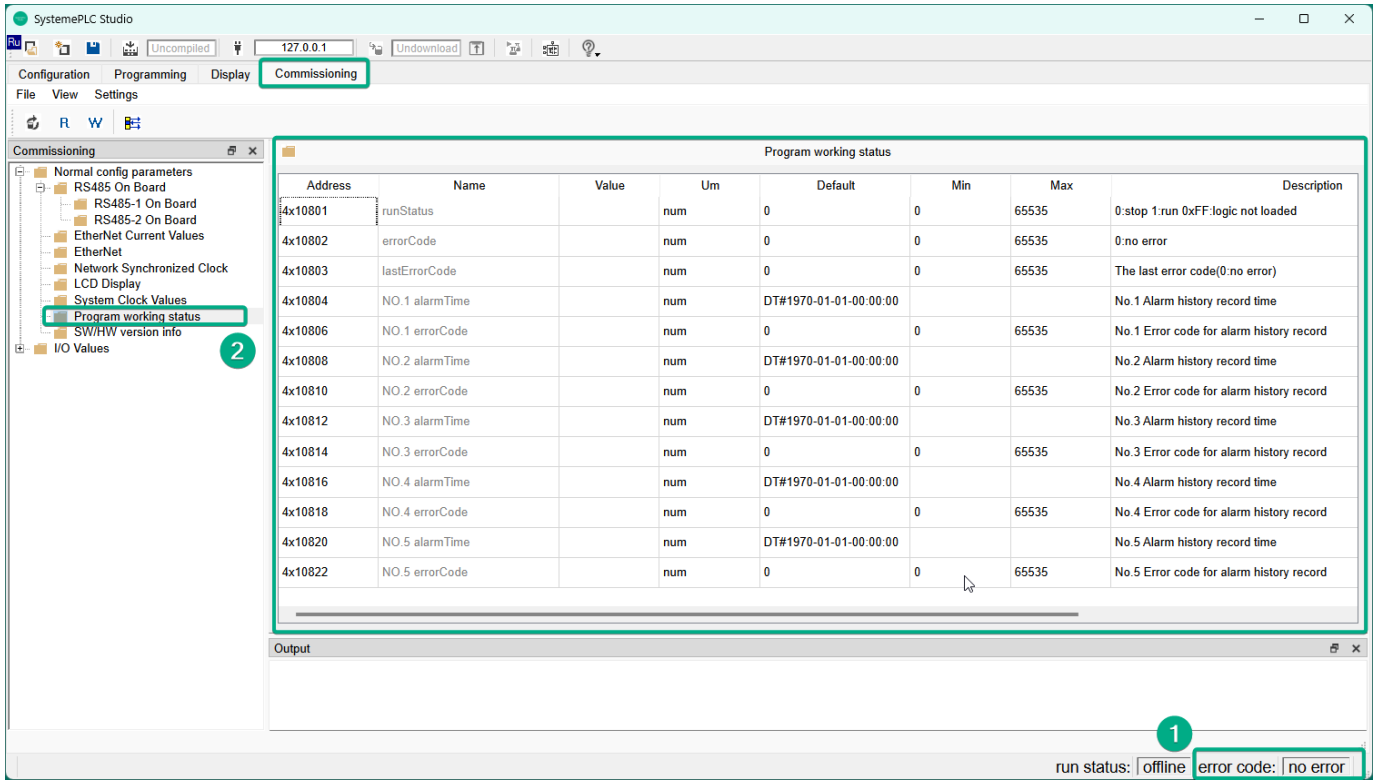
Системный экран ПЛК	Системное меню ПЛК	Экран «Alarm history»
		

В данном примере ПЛК (без подключения внешнего питания) подключен кабелем USB TypeC к ПК

- На системном экране ПЛК отображается «No Power»
- В журнале ошибок «Alarm history» два события с кодом ошибки (ERR CODE) «7» («Отсутствует питание контроллера») и метками времени
- Горит индикатор ошибки «SF»

Просмотр кода ошибки в ПО SystemePLC Studio (только чтение)

1. В **строке состояния (Status bar)** отображается код текущей ошибки (**error code**).
2. На вкладке **Commisioning > Program Working Status** отображается
   a. переменные errorCode, lastErrorCode.
   b. журнал аварийных событий ПЛК (Alarm History) состоящий из 5 последних кодов ошибки (NO1_errorCode, ..., NO5_errorCode) с метками времени (NO1_AlarmTime, ..., NO5_AlarmTime)

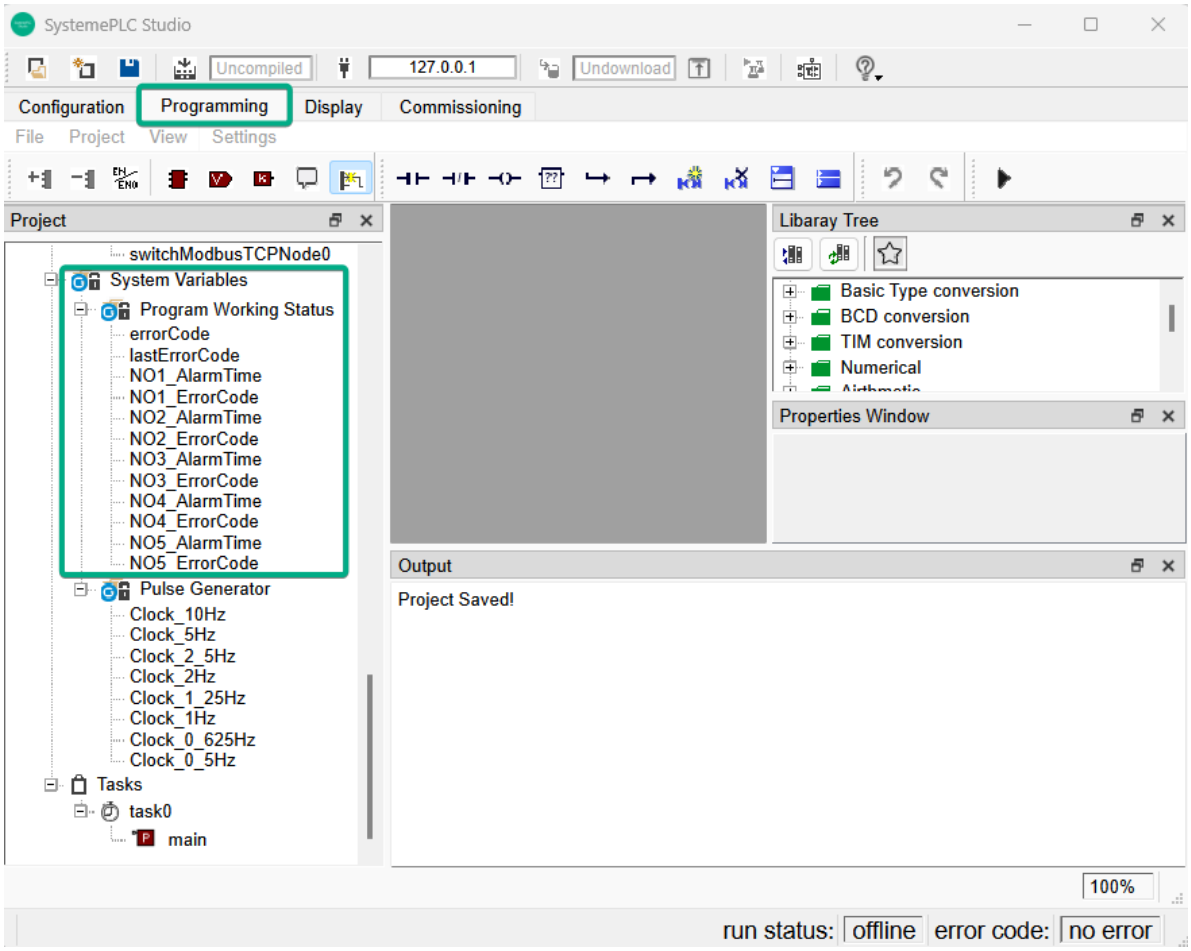


Описание переменных для обработки кодов ошибки

- **errorCode** – код текущей ошибки (тип UINT)
  - Пока ошибка есть, переменная errorCode содержит код текущей ошибки
  - Если ошибка устранена, переменная errorCode содержит 0
- **lastErrorCode** – код последней возникшей ошибки (тип UINT)
  - Сохраняет код последней ошибки даже после того, как ошибка была устранена
- **NO1_AlarmTime**, ..., **NO5_AlarmTime** – время возникновения ошибки №1,2,3,4,5 [тип UTC (Unix Time)]
- **NO1_errorCode**, ..., **NO5_errorCode** – код ошибки №1,2,3,4,5 (тип UINT)

Использование кода ошибки в программе пользователя в ПО SystemePLC Studio

Переменные для обработки ошибок, которые можно использовать в программе пользователя содержаться в дереве проекта на вкладке:
Programming > Systeme Variables > Program Working Status



Описание переменных, находящихся в Programming > Systeme Variables > Program Working Status полностью совпадает с переменными на вкладке Commisioning > Program Working Status.

3.1.1 Журнал аварийных событий ПЛК (Alarm History) с фиксированным числом записей (N)

Переменные журнала ошибок хранят пять (N=5) последних ошибок с меткой времени в соответствующих переменных

- NO1_AlarmTime – время возникновения ошибки №1
- NO1_errorCode – код ошибки №1
- …
- NO5_AlarmTime – время возникновения ошибки №5
- NO5_errorCode – код ошибки №5

Принцип работы

Журнал предназначен для регистрации аварийных событий с указанием времени их возникновения и кода ошибки, а также для последующего отображения в ПО SystemePLC Studio, анализа и использовании в программе контроллера, и передачи по протоколу Modbus TCP и Modbus RTU.

Журнал аварий представляет собой структуру переменных фиксированного размера, содержащую **N=5 записей**. Каждая запись соответствует одному аварийному событию.

Журнал работает по принципу **скользящего окна с логическим сдвигом**, где всегда хранятся последние N аварий, отсортированные по времени возникновения ошибки.

Структура одной записи журнала содержит следующие поля:

- **Код ошибки** (ErrorCode): целочисленное значение (UINT).
- **Время события** (Timestamp): дата и время возникновения ошибки.

В журнале хранится **время возникновения события**. Время фиксируется по **локальному времени контроллера**. Внутреннее представление времени в формате UTC (Unix Time) (32 бит), с последующим преобразованием в локальное время при отображении в ПО SystemePLC Studio.

При первом старте системы или после инициализации все N записей журнала считаются пустыми. Код ошибки во всех записях равен **0**. Время равно **0** (что соответствует 01.01.1970 00:00:00 UTC).
Пустая запись интерпретируется как отсутствие зарегистрированной аварии.

Правила добавления аварий

1. Одиночная авария.
   a. При возникновении одной аварии фиксируется код ошибки, фиксируется точное время её возникновения. Запись помещается в **первую строку журнала**. Все существующие записи сдвигаются на одну позицию вниз.
2. Последовательные аварии.

   a. При возникновении последующих аварий новая авария всегда записывается в первую строку журнала. Более ранние записи последовательно сдвигаются вниз. Самая старая запись удаляется при превышении размера N. Таким образом, верхняя строка — самая **поздняя** авария, а нижняя строка — самая **ранняя** из сохранённых.
3. Переполнение журнала.
   a. После заполнения всех N строк и при добавлении новой аварии самая ранняя (нижняя) запись безвозвратно удаляется. Размер журнала остаётся постоянным и равным N.
4. Одновременное возникновение нескольких аварий.
   a. Если в системе возникает несколько аварий они должны рассматриваться как **последовательность отдельных событий**. Для каждой аварии формируется отдельная запись.
   b. Записи сортируются по времени возникновения: более ранние — ниже, а более поздние — выше.
   c. Если время возникновения нескольких аварий **совпадает**, то допускается произвольный порядок размещения таких записей между собой.

3.1.2 Таблица кодов ошибок SystemePLC Studio

*Коду ошибки соответствует надпись на системном экране ПЛК (при наличии дисплея).

Код	Тип ошибки	Описание ENG	Описание RU	Отображение на дисплее ПЛК*
0	Нет ошибок	No alarm	Нет ошибок	No Faults
5	Ошибка совместимости	Firmware version incompatible with hardware	Версия прошивки несовместима с аппаратной частью	—
6	Аппаратная ошибка	Main controller analog input hardware fault	Неисправность аналогового входа контроллера	ADC Fault
7	Ошибка питания	Main controller missing primary power supply	Отсутствует питание контроллера	No Power
8	Ошибка сброса	Restart alarm triggered by watchdog timeout	Перезапуск из-за срабатывания сторожевого таймера (Watchdog)	—
18	Останов по команде	Manual operation stop alarm	Остановка оборудования из-за ручного вмешательства	Manual Stop
30	Ошибка программы	User program type error	Неверный тип пользовательской программы	—
31	Ошибка инициализации	User program entry initialization failure	Сбой инициализации точки входа пользовательской программы	—
32	Переполнение памяти	User program memory usage exceeds limit	Использование памяти пользовательской программой превышает лимит	—
33	Неверный адрес	User program address invalid	Недопустимый адрес в пользовательской программе	—
34	Ошибка retain-переменных	Retain variable usage exceeds limit	Размер retain-переменных превышает лимит	—
35	Индекс вне диапазона	Read/write variable type index out of range	Индекс типа индексируемой переменной выходит за пределы	—
36	Ошибка типа	Read/write variable type error	Ошибка типа переменной при чтении/записи	—
37	Ошибка FLASH	User program FLASH area error	Ошибка в FLASH-области пользовательской программы	—

38	Ошибка вычислений	Division-by-zero error in program	Деление на ноль в программе	ZeroDivision Err
110	Ошибка связи	Expansion module 1 communication fault	Ошибка связи с модулем расширения 1	Ext:1 disconnect
111	Ошибка связи	Expansion module 2 communication fault	Ошибка связи с модулем расширения 2	Ext:2 disconnect
112	Ошибка связи	Expansion module 3 communication fault	Ошибка связи с модулем расширения 3	Ext:3 disconnect
113	Ошибка связи	Expansion module 4 communication fault	Ошибка связи с модулем расширения 4	Ext:4 disconnect
114	Ошибка связи	Expansion module 5 communication fault	Ошибка связи с модулем расширения 5	Ext:5 disconnect
115	Ошибка связи	Expansion module 6 communication fault	Ошибка связи с модулем расширения 6	Ext:6 disconnect
116	Ошибка связи	Expansion module 7 communication fault	Ошибка связи с модулем расширения 7	Ext:7 disconnect
118	Модуль не найден	Configured expansion module not connected	Настроенный модуль расширения не подключён к шине	ExtM not found
126	Несоответствие конфигурации	Expansion module 1 config mismatch	Конфигурация модуля 1 не соответствует аппаратному обеспечению	Ext:1 config err
127	Несоответствие конфигурации	Expansion module 2 config mismatch	Конфигурация модуля 2 не соответствует аппаратному обеспечению	Ext:2 config err
128	Несоответствие конфигурации	Expansion module 3 config mismatch	Конфигурация модуля 3 не соответствует аппаратному обеспечению	Ext:3 config err
129	Несоответствие конфигурации	Expansion module 4 config mismatch	Конфигурация модуля 4 не соответствует аппаратному обеспечению	Ext:4 config err
130	Несоответствие конфигурации	Expansion module 5 config mismatch	Конфигурация модуля 5 не соответствует аппаратному обеспечению	Ext:5 config err
131	Несоответствие конфигурации	Expansion module 6 config mismatch	Конфигурация модуля 6 не соответствует аппаратному обеспечению	Ext:6 config err
132	Несоответствие конфигурации	Expansion module 7 config mismatch	Конфигурация модуля 7 не соответствует аппаратному обеспечению	Ext:7 config err

134	Ошибка питания	Expansion module 1 power abnormal	Отсутствует питания модуля расширения 1	Ext:1 no power
135	Ошибка питания	Expansion module 2 power abnormal	Отсутствует питания модуля расширения 2	Ext:2 no power
136	Ошибка питания	Expansion module 3 power abnormal	Отсутствует питания модуля расширения 3	Ext:3 no power
137	Ошибка питания	Expansion module 4 power abnormal	Отсутствует питания модуля расширения 4	Ext:4 no power
138	Ошибка питания	Expansion module 5 power abnormal	Отсутствует питания модуля расширения 5	Ext:5 no power
139	Ошибка питания	Expansion module 6 power abnormal	Отсутствует питания модуля расширения 6	Ext:6 no power
140	Ошибка питания	Expansion module 7 power abnormal	Отсутствует питания модуля расширения 7	Ext:7 no power
210	Выход за диапазон	Expansion module 1 AI out of range	Аналоговый вход (AI) модуля 1 вышел за пределы диапазона	Ext:1 AI OR
211	Выход за диапазон	Expansion module 2 AI out of range	Аналоговый вход (AI) модуля 2 вышел за пределы диапазона	Ext:2 AI OR
212	Выход за диапазон	Expansion module 3 AI out of range	Аналоговый вход (AI) модуля 3 вышел за пределы диапазона	Ext:3 AI OR
213	Выход за диапазон	Expansion module 4 AI out of range	Аналоговый вход (AI) модуля 4 вышел за пределы диапазона	Ext:4 AI OR
214	Выход за диапазон	Expansion module 5 AI out of range	Аналоговый вход (AI) модуля 5 вышел за пределы диапазона	Ext:5 AI OR
215	Выход за диапазон	Expansion module 6 AI out of range	Аналоговый вход (AI) модуля 6 вышел за пределы диапазона	Ext:6 AI OR
216	Выход за диапазон	Expansion module 7 AI out of range	Аналоговый вход (AI) модуля 7 вышел за пределы диапазона	Ext:7 AI OR
232	Индикация	User test warning indicator	Пользовательский тестовый светодиодный индикатор (предупреждение)	Test Led

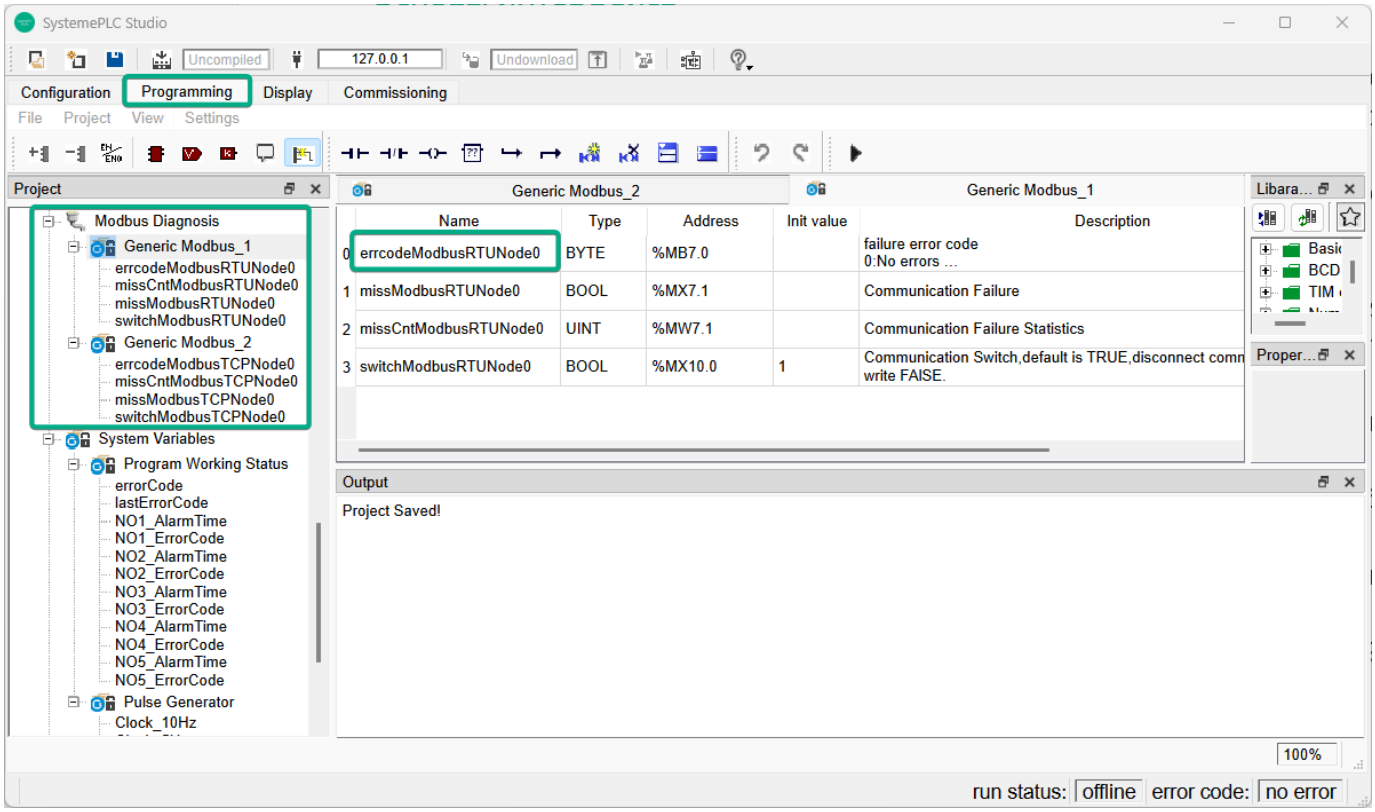
3.2 Коды ошибок Modbus

Переменные для обработки ошибок Modbus

Переменные для обработки ошибок Modbus находятся в дереве проекта во вкладке: Programming > Modbus Diagnostics > Generic Modbus_xxx

- errcodeModbusRTUNode0 – содержит код ошибки Modbus RTU
- errcodeModbusTCPNode0 – содержит код ошибки Modbus TCP

Примечание: раздел Programming > Modbus Diagnostics появится после создания мастер устройства (Generic Modbus) для сети RS-485 (Modbus RTU) или Ethernet (Modbus TCP) на вкладке Configuration.



Переменные errcodeModbusRTUNode0 и errcodeModbusTCPNode0 содержат код ошибки Modbus.

Важно! Значение кода ошибки в переменных errcodeModbusRTUNodeX, errcodeModbusTCPNodeX не будет автоматически сбрасываться в 0 при её отсутствии в последующих коммуникациях. Для контроля повторного возникновения той же ошибки необходимо в программе предусмотреть запись 0 в соответствующую переменную errcodeModbus....

3.2.1 Описание кода ошибки Modbus

Код ошибки Modbus может принимать следующие значения

Данные получены (16#00)

- 16#00: Нет ошибок

Данные не получены (16#01 – 16#02)

- 16#01: Сбой передачи данных
- 16#02: Таймаут приема

Данные получены с ошибкой (16#10 – 16#15)

- 16#10: Неверная CRC
- 16#11: Неверные данные
- 16#12: Неверный код исключения
- 16#14: Слишком много данных
- 16#15: Ответ не от запрошенного ведомого устройства

Получен код исключения 16#8x.

Формат кода ошибки **16#8x, где x (младший полубайт) – содержит код исключения Modbus.**

Байт **16#8x** разбит на две части по 4 бита - полубайт (ниббл). Если старший полубайт равен **16#8**, то младший полубайт **16#x** содержит соответствующий код исключения Modbus (см. документ Modbus Protocol Specification V1.1b3 стр. 47-49)

Например:

- 16#81: Соответствует коду исключения 01, что означает недопустимый код функции
- 16#82: Соответствует коду исключения 02, значение недопустимый адрес данных

Полный список **стандартных кодов исключений Modbus** см. в документе:
Modbus Protocol Specification V1.1b3 стр. 47-49 на официальном сайте
https://www.modbus.org/modbus-specifications

4. Защита пользовательского проекта

Следующие функции позволяют осуществить комплексную защиту пользовательского проекта.

Защита пользовательского проекта

- Защита проекта паролем (File > Security) позволяет защитить пользовательский проект от открытия не авторизованным лицом.
- Защита паролем пользовательской программы (POU), пользовательских функциональных блоков и функций (ПКМ > Crypt / Decrypt) позволяет защищать отдельные части проекта от просмотра и редактирования не авторизованным лицом.
- Шифрование устройства (Configuration > Device Password) позволяет установить пароль на выгрузку проекта из ПЛК (Upload Project).

Загрузка/ выгрузка исполняемого файла программы

- Функция загрузки в ПЛК / выгрузки из ПЛК конфигурационных и исполняемых файлов пользовательской программы (Settings > PLC > Download / Upload bin file) в формате *.bin позволяет не передавать в эксплуатацию сам проект, а отправить только исполняемый файл готовой программы. В этом случае редактируемый проект программы ПЛК остается только у его разработчика.

4.1 Защита проекта паролем

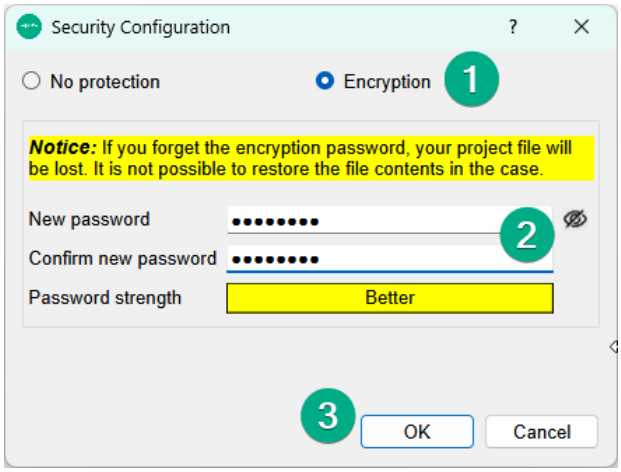
Важно! В случае если вы забыли пароль проект будет утерян! Восстановить его будет невозможно!

Защита проекта паролем (пароль на открытие проекта)

Откройте пункт Меню File > Security

1. В открывшемся окне выберите Encryption
   a. Encryption – включить защиту проекта паролем,
   b. No Protection – выключить защиту проекта паролем
2. Задайте пароль и подтвердите его
3. Нажмите OK
4. Сохраните проект File > Save Project
5. Загрузите проект в ПЛК (Download PLC code)

При установленном пароле после закрытия проекта (или выгрузке его из ПЛК) и его последующем открытии, программа запросит пароль.



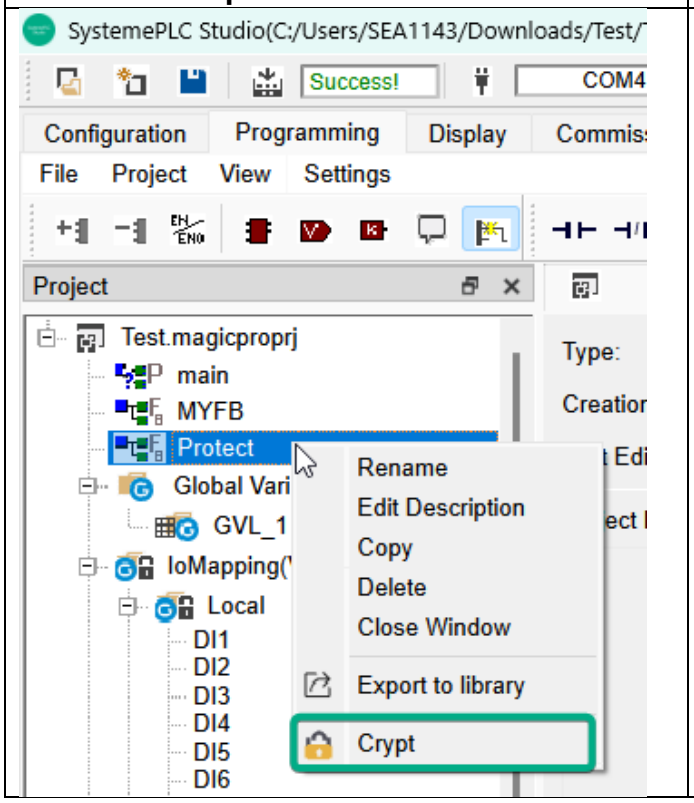
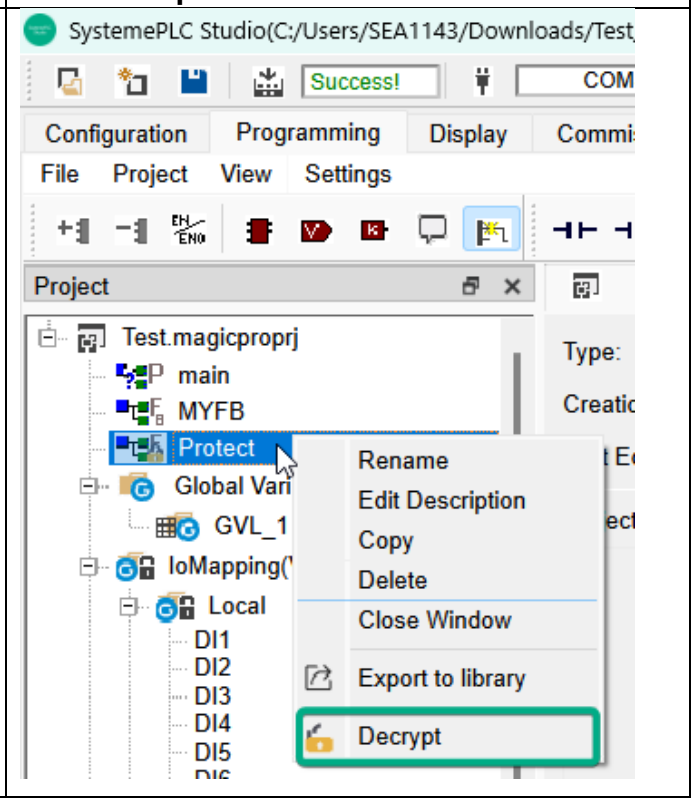
4.2 Защита паролем пользовательской программы (POU), пользовательских функциональных блоков и функций

Можно защитить паролем отдельные (или все сразу) части проекта, такие как: пользовательские программы (POU), пользовательские функциональные блоки и пользовательские функции.

Важно! В случае если вы забыли пароль пользовательская программа (POU), пользовательский блок или пользовательская функция будут утеряны! Восстановить их будет невозможно!

Установка / снятие пароля на отдельные части проекта

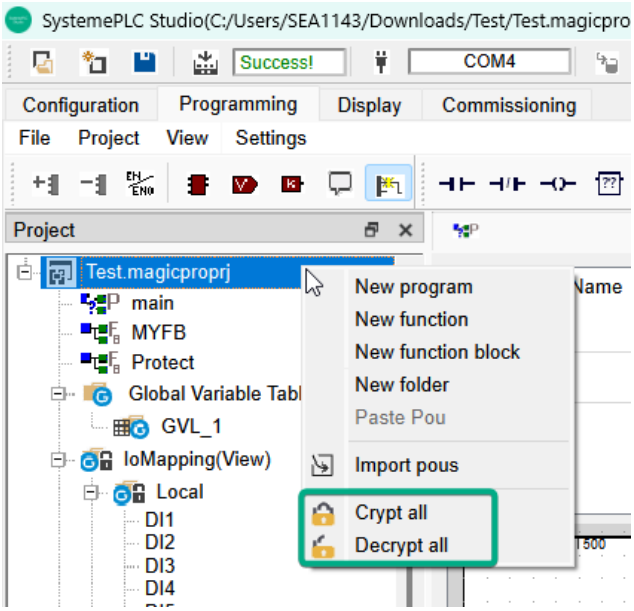
1. Нажмите ПКМ на пользовательской программе (POU), пользовательском функциональном блоке или пользовательской функции
2. В контекстном меню выберите Crypt – установить пароль
   a. Введите пароль
   b. После этого на защищаемом элементе отобразится значок замка. Пароль установлен.
3. В контекстном меню выберите Decrypt – снять пароль
   (1) Введите пароль
   (2) После этого на защищаемом элементе значок замка исчезнет. Пароль снят.

Установка пароля	Снятие пароля
	

Установка / снятие пароля на все части проекта одновременно

1. Нажмите ПКМ на имени проекта

2. В контекстном меню выберите Crypt all – установить пароль
   a. Введите пароль
   b. После этого на защищаемых элементах отобразится значок замка. Пароль установлен.
3. В контекстном меню выберите Decrypt all – снять пароль
   a. Введите пароль
   b. После этого на защищаемых элементах значок замка исчезнет. Пароль снят.

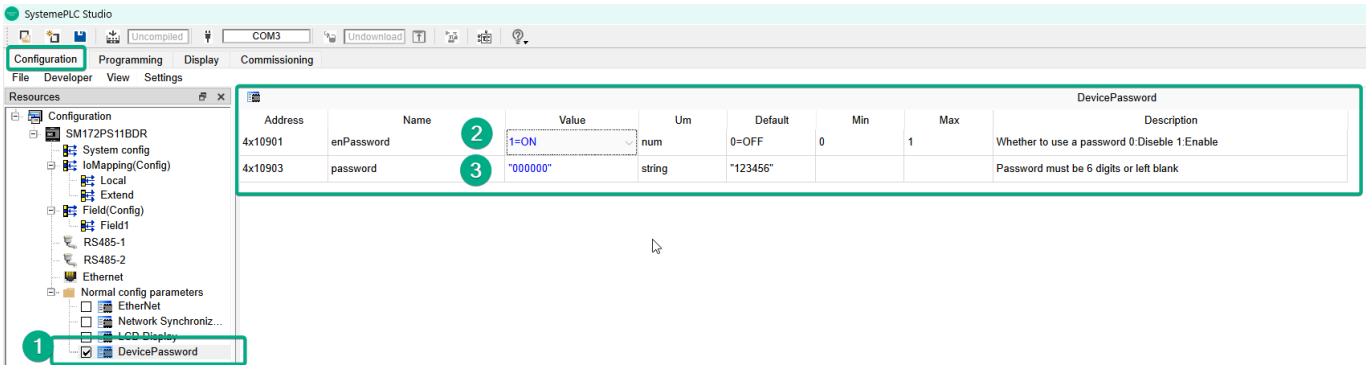


4.3 Шифрование устройства

Настройка шифрования устройства (пароль на выгрузку проекта)

1. Перейдите на вкладку меню Configuration
   a. Поставьте галочку напротив пункта Device Password
   b. Двойным щелчком ЛКМ откройте окно Device Password
2. Задайте значение переменной enPassword в поле Value
   a. 1 – использовать шифрование устройства
   b. 0 – не использовать шифрование устройства
3. Задайте значение переменной password в поле Value – задает пользовательский пароль
   a. Минимальная длина пароля 6 символов
   b. Пароль по умолчанию «123456»
4. Загрузите программу в ПЛК
5. Перезагрузите ПЛК по питанию.

После перезагрузки шифрование устройства будет активировано. При попытке выгрузить проект (Upload Project) программа запросит пароль.



4.4 Загрузка/выгрузка исполняемых файлов

Важно! Загрузка исполняемого файла (*.bin) с ПК или флешки в ПЛК **не позволяет** выгрузить проект для редактирования (из ПЛК на ПК).

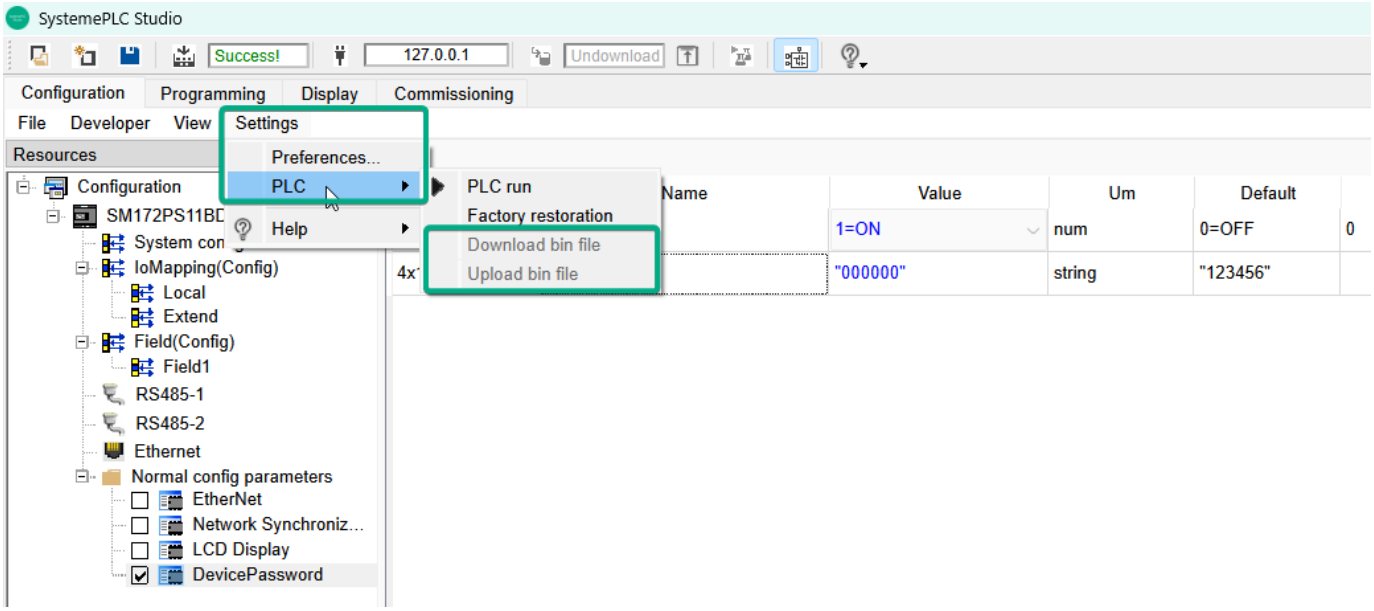
Для загрузки/выгрузки в/из ПЛК доступны следующие исполняемые файлы:

- Файл проекта <имя проекта>\Build**prog.bin** содержит проект целиком: программу и значение переменных по умолчанию.
- Файл проекта <имя проекта>\Build**config.bin** содержит только значения переменных по умолчанию.

4.4.1 Загрузка с ПК в ПЛК / выгрузка из ПЛК на ПК исполняемых файлов

Откройте пункт Меню File > Settings > PLC

- Download bin file – загрузить с ПК в ПЛК исполняемый файл (prog.bin)
  - Выберите путь к файлу <имя проекта>\Build\prog.bin
- Upload bin file – выгрузить из ПЛК на ПК исполняемый (prog.bin) и конфигурационный (config.bin) файлы
  - Выберите путь сохранения файлов prog.bin, config.bin



4.4.2 Загрузка с флешки в ПЛК исполняемых файлов*

*Доступно только для устройств с портом USB Type-A (без необходимости подключения к ПК).

Важно! максимальный объем карты памяти до 32 GB; формат: FAT32; размер сектора: 4KB.

Загрузка с флешки в ПЛК исполняемого файла (программа и значение переменных по умолчанию)

Исполняемый файл проекта <имя проекта>\Build**prog.bin** содержит программу и значение переменных по умолчанию.

Для загрузки его в ПЛК с флешки необходимо

1. Записать в корневую папку флешки файл проекта из <имя проекта>\Build**prog.bin**
1. Вставить флешку в порт ПЛК USB TypeA
2. С экрана ПЛК выбрать команду USB Flash > Load Logic. Нажать Yes.
   a. Не вынимать флешку и не отключать питание ПЛК пока на экране горит сообщение «Do not unplug PLC».
   b. Через некоторое время произойдет переключение на экран ПЛК по умолчанию (системный или пользовательский). После этого флешку можно извлечь.

Загрузка с флешки в ПЛК файла конфигурации (значения переменных по умолчанию)

Файл проекта <имя проекта>\Build**config.bin** содержит только значения переменных по умолчанию.

Для загрузки его в ПЛК с флешки необходимо

1. Записать в корневую папку флешки файл проекта из <имя проекта>\Build**config.bin**
2. Вставить флешку в порт ПЛК USB TypeA
3. С экрана ПЛК выбрать команду USB Flash > **Load Configuration**. Нажать Yes.

a. Не вынимать флешку и не отключать питание ПЛК пока на экране горит сообщение «Do not unplug PLC».
b. Через некоторое время произойдет переключение на экран ПЛК по умолчанию (системный или пользовательский). После этого флешку можно извлечь.

Социальные сети

systemeelectric_official

youtube.com/c/SystemeElectric

vk.com/Systemeelectric

Systeme Electric

www.systeme.ru

Наши бренды

Système electric

Dēkraft

Механотроника

Système soft