

Содержание главы 7

7	<u>Таймеры</u>	3
7.1	<u>Программирование таймера</u>	4
7.1.1	<u>Общее представление о таймере</u>	4
7.1.2	<u>Запуск таймера</u>	6
7.1.3	<u>Определение длительности времени</u>	6
7.1.4	<u>Сброс таймера</u>	8
7.1.5	<u>Опрос таймера</u>	8
7.1.6	<u>Последовательность операций с таймерами</u>	9
7.1.7	<u>Блочный элемент таймера в цепи (LAD)</u>	10
7.1.8	<u>Блочный элемент таймера в логической схеме (FBD)</u>	10
7.2	<u>Импульсный таймер</u>	11
7.3	<u>Расширенный импульсный таймер</u>	13
7.4	<u>Таймер задержки включения</u>	15
7.5	<u>Таймер задержки включения с запоминанием</u>	17
7.6	<u>Таймер задержки выключения</u>	19
7.7	<u>IEC-таймеры</u>	21
7.7.1	<u>Импульсный таймер SFB 3 TP</u>	22
7.7.2	<u>Таймер задержки включения SFB 4 TON</u>	22
7.7.3	<u>Таймер задержки выключения SFB 5 TOF</u>	23

7 Таймеры

Таймеры (Timers) позволяют программно реализовать последовательности синхронизации, такие как интервалы ожидания и наблюдения, измерение интервалов или генерирование импульсов.

Существуют следующие типы таймеров:

- Импульсные таймеры (Pulse timers);
- Расширенные импульсные таймеры (Extended pulse timers);
- Таймеры задержки включения (On-delay timers);
- Таймеры задержки включения с запоминанием (Retentive on-delay timers);
- Таймеры задержки выключения (Off-delay timers).

Вы можете запрограммировать таймер полностью как блочный элемент или с помощью отдельных программных элементов. При запуске таймера вы определяете требуемый тип таймера, и как долго он должен работать; также вы можете сбросить таймер. Таймер контролируется путем запроса его состояния («Timer running», «Таймер работает») или текущего значения таймера, которое вы можете получить из таймера в двоичном или двоично-десятичном (BCD) коде.

Блочный элемент таймера

(в примере: импульсный таймер)



Наименование	Тип данных	Описание
S	BOOL	Вход запуска
TV	S5TIME	Спецификация длительности времени
R	BOOL	Вход сброса
BI	WORD	Текущее значение времени в двоичном представлении
BCD	WORD	Текущее значение времени в BCD-представлении
Q	BOOL	Состояние таймера

Рисунок 7.1 Таймер в виде блочного элемента (timer box)

7.1 Программирование таймера

7.1.1 Общее представление о таймере

Вы можете выполнять следующие операции над таймером:

- Запустить таймер, задав значение времени;
- Обнулить (сбросить) таймер;
- Проверить (двоичное) состояние таймера;
- Прочитать (цифровое) значение таймера в бинарном виде;
- Прочитать (цифровое) значение таймера в BCD-виде.

Блочный элемент для таймера содержит все эти операции таймера в форме функциональных входов и функциональных выходов (рисунок 7.1). Над блочным элементом расположен абсолютный или символический адрес таймера. В самом блочном элементе в качестве заголовка указан режим таймера (S_PULSE означает «Start pulse timer» или «Запуск импульсного таймера»). Назначения для входов S и TV обязательны, в то время как для других – нет.

Отдельные программные элементы в LAD

Также можно запрограммировать таймер с использованием отдельных программных элементов (рисунок 7.2). В этом случае запуск таймера осуществляется посредством катушки. Режим таймера указан в катушке (SP = start pulse timer или запуск импульсного таймера), а под катушкой расположено значение в формате S5TIME, определяющее длительность (duration). Чтобы обнулить таймер, используйте катушку сброса. NO- и NC-контакты можно применить для проверки состояния таймера. Наконец, вы можете сохранить текущее значение времени в двоичной форме в операнде размером в слово с помощью блочного элемента MOVE.

Отдельные программные элементы в FBD

Вы также можете запрограммировать таймер с помощью отдельных программных элементов (рисунок 7.3). В данном случае таймер стартует посредством простого блочного элемента, содержащего режим таймера (SP = start pulse timer или запуск импульсного таймера). Под элементом находится значение в формате S5TIME, определяющее длительность. Чтобы обнулить таймер, используйте блочный элемент сброса. Сканировать состояние таймера можно непосредственно или в инвертированном виде с использованием любого бинарного входа. Наконец, вы можете сохранить текущее значение времени в двоичной форме в операнде размером в слово с помощью блочного элемента MOVE.

Запуск таймера с определенным значением времени (катушка запуска с рабочими характеристиками)	
Сброс таймера (катушка сброса)	
Проверка состояния таймера (NO-контакт, NC-контакт)	
Чтение значения времени в двоичной форме (блочный элемент MOVE)	

Рисунок 7.2 Отдельные элементы таймера (LAD)

Для создания программ с помощью пошагового программирования элементы-таймеры вы можете найти в каталоге программных элементов (Program Element Catalog), вызываемого командой меню View → Catalog (Вид → Каталог) [Ctrl + K] или Insert → Program Elements (Вставка → Программные элементы), в разделе «Timers» («Таймеры»).

Запуск таймера с определенным значением времени (блочный элемент запуска с рабочими характеристиками)	
Сброс таймера (блочный элемент сброса)	
Проверка состояния таймера (прямой или инвертированный двоичный вход)	
Чтение значения времени в двоичной форме (блочный элемент MOVE)	

Рисунок 7.3 Отдельные элементы таймера (FBD)

7.1.2 Запуск таймера

Таймер стартует, когда результат логической операции (RLO) меняется на входе запуска (start input) или перед катушкой / блочным элементом запуска. Такая смена сигнала всегда требуется, чтобы запустить таймер. В случае таймера задержки включения RLO должен поменяться с «1» на «0»; все другие таймеры стартуют при смене RLO с «0» на «1».

Вы можете запустить в одном из пяти различных режимов (рисунок 7.4). Однако, нет смысла использовать любой данный таймер более, чем в одном режиме.

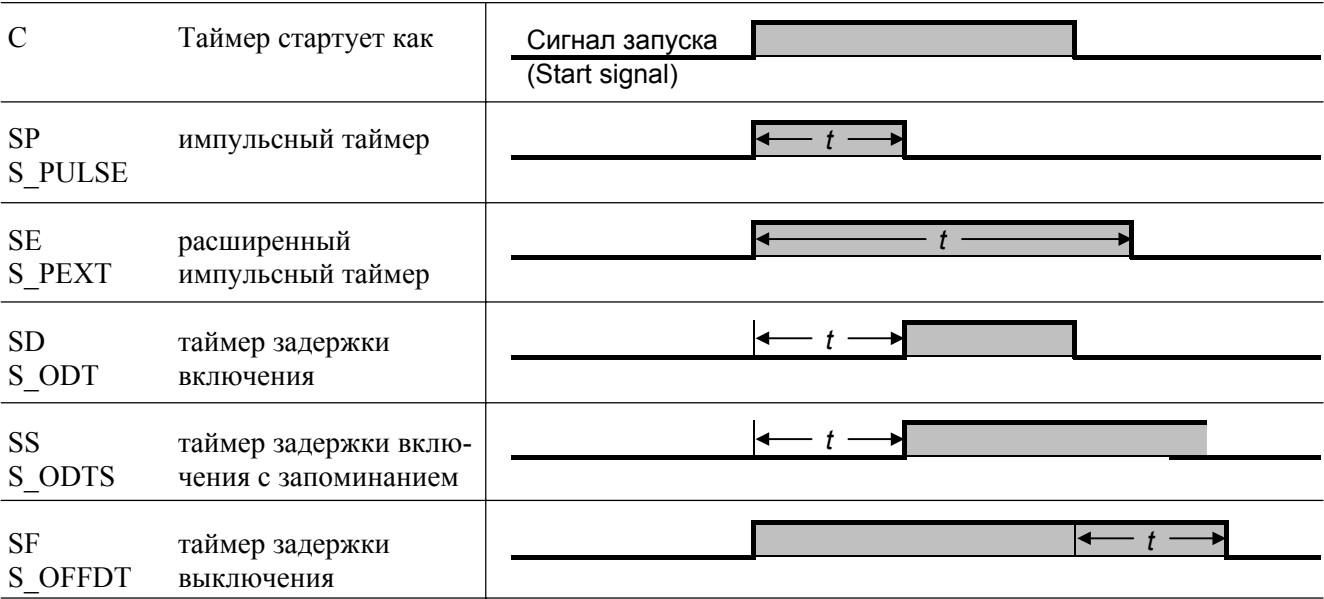


Рисунок 7.4 Рабочие характеристики таймера

7.1.3 Определение длительности времени

Таймер в качестве длительности принимает значение, указанное под катушкой / блочным элементом, или значение на входе TV. Вы можете задать длительность как константу, как операнд размером в слово или как переменную типа S5TIME.

Определение длительности как константы

S5TIME#10s	Длительность 10 с
S5T#1m10ms	Длительность 1 м + 10 мс

Длительность задается в часах, минутах, секундах и миллисекундах. Область значений простирается от S5TIME#10s до S5TIME#2h46min30s (что соответствует 9990 с). Промежуточные значения округляются до 10 мс. Для определения константы вы можете использовать S5TIME# или S5T#.

Определение длительности как операнда или переменной

MW 20	Операнд размером в слово, содержащий длительность
"Time1"	Переменная типа S5TIME

Значение в 16-битном операнде должно соответствовать типу данных S5TIME (см. ниже «Структура значения длительности времени»).

Структура значения длительности времени

Структурно длительность состоит из значения времени (time value) и временной базы (time base): длительность = значение времени × временная база. Длительность представляет собой интервал времени, в течение которого таймер активен («таймер работает»). Значение времени – это число циклов, которые обрабатывает таймер. Временная база определяет интервал, за который CPU должен изменить значение времени (рисунок 7.5).



Рисунок 7.5 Описание битов в значении длительности

Также можно задать длительность времени непосредственно в 16-битном операнде. Чем меньше временная база, тем точнее фактическая длительность. Например, если вы хотите реализовать длительность, равную одной секунде, то вы можете указать одну из трех спецификаций:

Длительность = 2001_{hex} Временная база 1 с

Длительность = 1010_{hex} Временная база 100 мс

Длительность = 0100_{hex} Временная база 10 мс

В данном случае последнее предпочтительнее.

При запуске таймера CPU принимает запрограммированное значение времени. Операционная система обновляет таймеры в фиксированные интервалы независимо от пользовательской программы, то есть она уменьшает значение времени всех актив-

ных таймеров согласно временной базе. Когда таймер достигает нуля, он «заканчивается». Затем CPU устанавливает состояние таймера (сигнальное состояние «0» или «1» в зависимости от режима, или «типа», таймера) и приостанавливает всю дальнейшую деятельность до рестарта таймера. Если вы определили нулевую длительность (0 мс или W#16#0000) при запуске таймера, то таймер остается активным, пока CPU не обработает его и не обнаружит, что время истекло.

Таймеры обновляются асинхронно по отношению к сканированию программы. В результате возможна ситуация, при которой состояние таймера в начале цикла отличается от состояния в конце цикла. Если вы используете таймеры только в одной точке программы и в предлагаемом порядке (см. ниже), то асинхронное обновление предотвратит появление ошибок.

7.1.4 Сброс таймера

LAD: таймер сбрасывается, когда электрический ток течет на входе сброса (reset input) или в катушке сброса (reset coil) (когда «RLO» равен «1»). Пока таймер остается сброшенным, сканирование с помощью NO-контакта вернет «0», с помощью NC-контакта – «1».

FBD: таймер обнуляется, когда на входе сброса присутствует «1». Пока таймер остается сброшенным, прямое сканирование состояния таймера возвратит «0», а инвертированное сканирование вернет «1».

Сброс таймера устанавливает данный таймер и временную базу в нуль. Вход R блочного элемента таймера должен быть назначен (то есть на этом входе должно присутствовать фактическое значение).

7.1.5 Опрос таймера

Проверка состояния таймера (LAD)

Состояние таймера можно получить на выходе Q блочного элемента таймера. Узнать состояние таймера вы можете также с помощью NO-контакта (соответствует выходу Q) или с помощью NC-контакта. Результаты считывания с помощью NO-контакта или выхода Q различаются в зависимости от типа таймера (см. описание типов таймера ниже). Как и в случае с входами, например, проверка с помощью NC-контакта генерирует в точности противоположный результат считывания по сравнению с результатом, полученным с помощью NO-контакта. Выход Q не должен использоваться в блочном элементе таймера.

Проверка состояния таймера (FBD)

Состояние таймера доступно на выходе Q блочного элемента таймера. Проверить состояние таймера вы также можете с использованием двоичного выхода функции (соответствует выходу Q). Результаты считывания таймера зависят от типа исполь-

зуемого таймера (см. описание типов таймера ниже). Выход Q может не использоваться в блочном элементе таймера.

Считывание значения времени (time value)

Выходы BI и BCD предоставляют значение времени таймера в двоичном (BI) и двоично-десятичном (BCD) виде. Это значение является текущим в момент считывания (если таймер активен, то значение времени отсчитывается от установленного значения в сторону уменьшения до нуля). Значение сохраняется в заданном операнде (передается как и при использовании блочного элемента MOVE). В блочном элементе таймера эти выходы можно не назначать.

Прямое считывание значения времени

Значение времени доступно в двоично-десятичном коде и может быть получено из таймера в этой форме. В таком случае временная база теряется и заменяется значением «0». Значение соответствует положительному числу в формате INT (Целое). Обратите внимание, что считанное значение времени не является длительностью! Вы также можете запрограммировать непосредственное чтение значения времени с помощью блочного элемента MOVE.

Закодированное считывание значения времени

Также вы можете вывести бинарное значение времени из таймера в «закодированной» форме. Здесь и значение времени, и временная база могут быть получены в двоично-десятичном виде. BCD-значение структурировано также, как для спецификации значения времени (см. выше).

7.1.6 Последовательность операций с таймерами

При программировании таймера не нужно использовать все доступные для этого операции. Вы должны применять только те операции, которые требуются для выполнения конкретной функции. Обычно к этим операциям относятся операции для запуска таймера и для проверки состояния таймера.

Чтобы таймер работал, как описано в этой главе, рекомендуется соблюдать следующий порядок при программировании с использованием отдельных программных элементов:

- Запуск таймера (Start);
- Сброс таймера (Reset);
- Считывание значения времени или длительности;
- Проверка состояния таймера.

При программировании ненужные элементы пропустите. Если вы соблюдаете порядок, показанный выше, и таймер запущен и сброшен «одновременно», то таймер запустится и немедленно обнулится. Последующая проверка таймера не обнаружит тот факт, что таймер был запущен.

7.1.7 Блочный элемент таймера в цепи (LAD)

Контакты перед входом запуска (start input) и входом сброса (reset input) вы можете соединить последовательно и параллельно, как и после выхода Q.

Сам блочный элемент таймера может быть расположен после Т-ветви и в ветви, которая непосредственно подключена к левой питающей шине (несущей). В этой ветви также могут быть контакты перед входом запуска, и это не должна быть самой верхней ветвью.

Примеры представления и группировки таймеров вы можете найти в библиотеке «LAD_Book» на прилагаемой к книге дискете, FB 107 программы «Basic Functions» («Основные функции»).

7.1.8 Блочный элемент таймера в логической схеме (FBD)

Перед входами запуска и сброса, как и после выхода Q, можно запрограммировать бинарные функции и функции для работы с памятью.

Блочный элемент таймера и отдельные элементы для запуска и сброса таймера могут также быть запрограммированы после Т-ветви.

Примеры представления и группировки таймеров вы можете найти в библиотеке «FBD_Book» на дискете, поставляемой с книгой, FB 107 программы «Basic Functions» («Основные функции»).

7.2 Импульсный таймер

Запуск импульсного таймера

Диаграмма на рисунке 7.6 поясняет характеристики таймера, когда он запускается как импульсный таймер, и когда он сбрасывается. Описание имеет силу, если при программировании с использованием отдельных элементов вы соблюдаете порядок, показанный в параграфе 7.1.6 «Последовательность операций с таймерами».

- ① Когда сигнальное состояние на входе запуска (start input) таймера меняется с «0» на «1» (положительный фронт), таймер запускается. Он работает на протяжении запрограммированной длительности, пока сигнальное состояние на входе запуска равно «1». Во время работы таймера выход Q выдает сигнальное состояние «1».

Имея стартовое значение в качестве начальной точки, значение времени отсчитывается в обратном порядке до нуля согласно временной базе.

- ② Если сигнальное состояние на входе запуска таймера меняется на «0» до истечения времени, то таймер останавливается. Выход Q становится равным «0». Значение времени показывает, сколько времени работал бы таймер еще, не будь он преждевременно прерван.

Сброс импульсного таймера

Сброс импульсного таймера имеет статический эффект и получает приоритет перед запуском таймера (рисунок 7.6).

- ③ Сигнальное состояние «1» на входе сброса (reset input) активного таймера обнуляет этот таймер. Выход Q в этом случае равен «0». Значению времени и временной базе также присваиваются нули. Если сигнальное состояние входа сброса переходит из «1» в «0», в то время как сигнальное состояние при входе установки еще равно «1», то таймер остается неизменным.
- ④ Сигнальное состояние «1» на входе сброса неактивного таймера никакого воздействия не оказывает.
- ⑤ Если сигнальное состояние на входе запуска переходит из «0» в «1» (положительный фронт), пока сигнал сброса еще присутствует, таймер стартует, но немедленно сбрасывается (на диаграмме показано линией). Если проверка состояния таймера была запрограммирована после сброса, то кратковременный запуск таймера не воздействует на проверку.

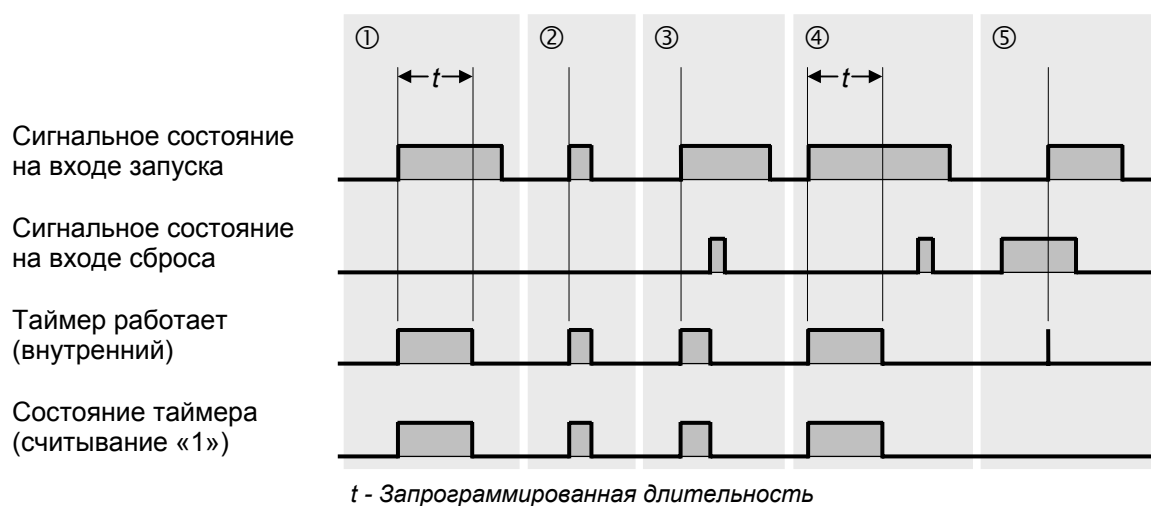


Рисунок 7.6 Рабочие характеристики при запуске и сбросе импульсного таймера

7.3 Расширенный импульсный таймер

Запуск расширенного импульсного таймера

Диаграмма на рисунке 7.7 описывает рабочие характеристики таймера после его запуска, и когда он сброшен. Описание применимо, если при программировании с использованием отдельных элементов вы соблюдаете порядок, показанный в параграфе 7.1.6 «Последовательность операций с таймерами».

- ❶❷ Когда сигнальное состояние на входе запуска таймера меняется с «0» на «1» (положительный фронт), таймер запускается. Он работает в течение запрограммированной длительности, даже когда сигнальное состояние на входе запуска переходит обратно в «0». Считывание с ожиданием сигнального состояния «1» (состояние таймера) возвращает результат проверки «1», пока таймер работает.

Имея стартовое значение в качестве начальной точки, значение времени отсчитывается в порядке уменьшения до нуля согласно временной базе.

- ❸ Если сигнальное состояние на входе запуска меняется с «0» на «1» (положительный фронт) во время работы таймера, он перезапускается с запрограммированным значением времени (таймер «повторно запускается»). Таймер может быть перезапущен любое количество раз, не доходя до завершения заданного времени.

Сброс расширенного импульсного таймера

Сброс расширенного импульсного таймера имеет статический эффект и получает приоритет перед запуском таймера (рисунок 7.7).

- ❹ Сигнальное состояние «1» на входе запуска работающего таймера сбрасывает его. Считывание с ожиданием сигнального состояния «1» (состояние таймера) возвращает результат проверки «0» для обнуления таймера. Значение времени и временная база также обнуляются.
- ❺ «1» на входе сброса неактивного таймера влияния не оказывает.
- ❻�� Если сигнальное состояние на входе запуска переходит из «0» в «1» (положительный фронт) при наличии сигнала сброса, то таймер запускается, но немедленно сбрасывается (на диаграмме показано линией). Если проверка состояния таймера запрограммирована после сброса, то кратковременный запуск таймера не повлияет на проверку таймера.

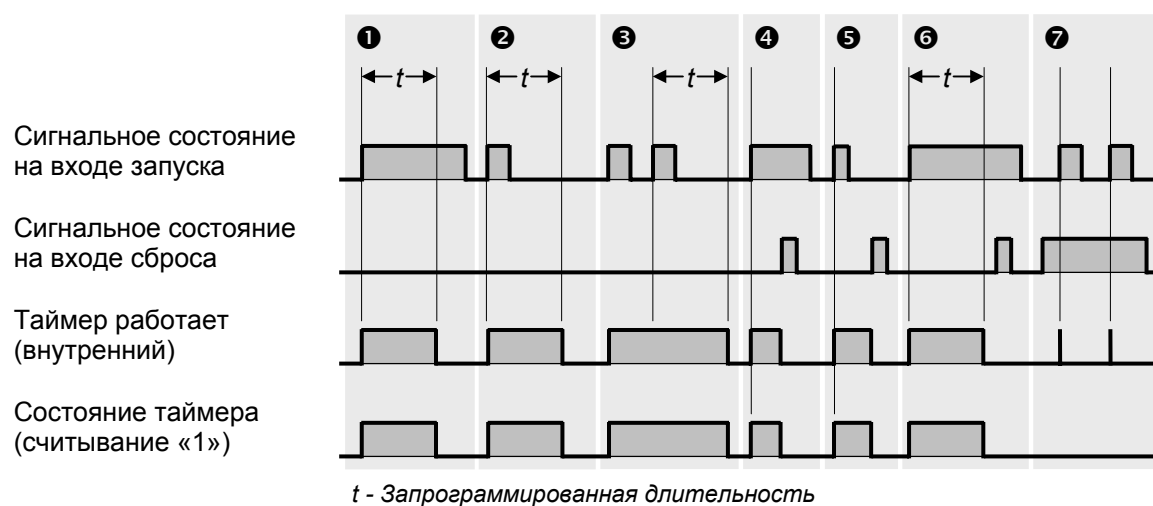


Рисунок 7.7 Рабочие характеристики при запуске и сбросе расширенного импульсного таймера

7.4 Таймер задержки включения

Запуск таймера задержки включения

На диаграмме, представленной на рисунке 7.8, описываются рабочие характеристики таймера после его запуска, и когда он сброшен. Описание применимо, если при программировании с использованием отдельных элементов вы соблюдаете порядок, показанный в параграфе 7.1.6 «Последовательность операций с таймерами».

- ① Когда сигнальное состояние на входе запуска таймера изменяется с «0» на «1» (положительный фронт), таймер запускается. Он работает в течение запрограммированной длительности. Операции считывания с ожиданием сигнального состояния «1» возвращают результат «1», если время должным образом истекло, и на входе запуска по-прежнему присутствует сигнальное состояние «1» (задержка включения).

Имея стартовое значение в качестве начальной точки, значение времени отсчитывается в направлении уменьшения до нуля согласно временной базе.

- ② Если во время работы таймера сигнальное состояние на входе запуска изменяется с «1» на «0», то таймер останавливается. В таких случаях считывание с ожиданием сигнального состояния «1» (состояние таймера) всегда возвращает результат проверки «1». Значение времени показывает количество оставшегося времени.

Обнуление таймера задержки включения

Сброс таймера задержки включения имеет статический эффект и получает приоритет перед запуском таймера (рисунок 7.8).

- ③④ Сигнальное состояние «1» на входе сброса обнуляет работающий и недействующий таймер. Считывание с ожиданием сигнального состояния «1» (состояние таймера) в этом случае возвращает результат проверки «0», даже если таймер не работает, и на входе запуска все еще присутствует сигнальное состояние «1». Значение времени и временная база также обнуляются.

Смена сигнального состояния на входе сброса с «1» на «0» при сигнальном состоянии «1» на входе запуска влияния на таймер не оказывает.

- ⑤ Если сигнальное состояние на входе запуска переходит из «0» в «1» (положительный фронт) при наличии сигнала сброса, то таймер запускается, но немедленно сбрасывается (на диаграмме это показано линией). Если проверка состояния таймера при программировании предусмотрена после сброса, то кратковременный запуск таймера не повлияет на проверку.

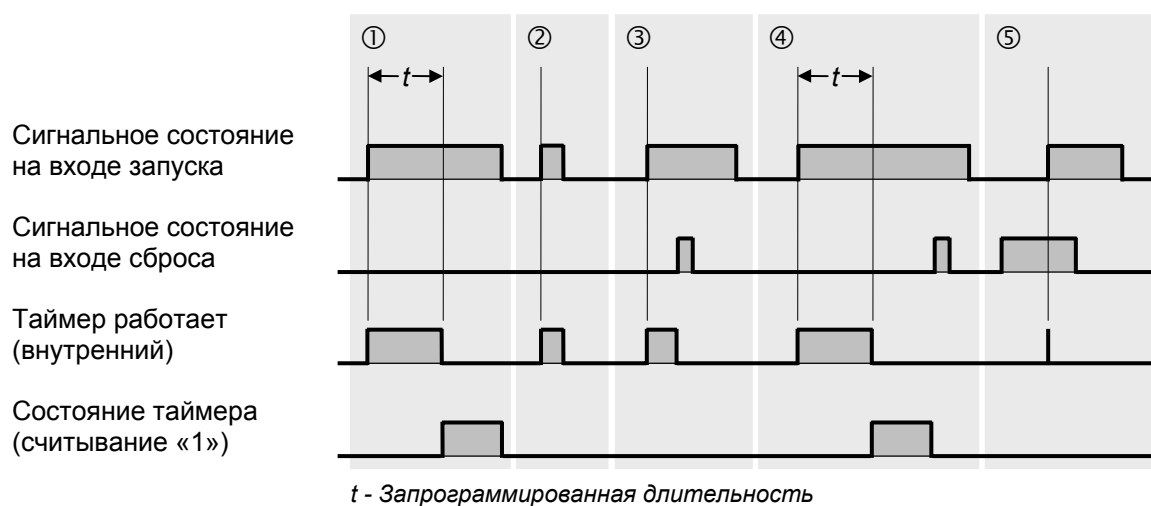


Рисунок 7.8 Рабочие характеристики при запуске и сбросе таймера задержки включения

7.5 Таймер задержки включения с запоминанием

Запуск таймера задержки включения с запоминанием

На диаграмме, изображенной на рисунке 7.9, описываются рабочие характеристики таймера после его запуска, и когда он сброшен. Описание применимо, если при программировании с использованием отдельных элементов вы соблюдаете порядок, показанный в параграфе 7.1.6 «Последовательность операций с таймерами».

- ❶ ❷ Когда сигнальное состояние на входе запуска таймера переходит из «0» в «1» (положительный фронт), таймер запускается. Он работает в течение запрограммированной длительности, даже когда сигнальное состояние на входе запуска возвращается к «0». По истечении времени считывание с ожиданием сигнального состояния «1» (состояние таймера) возвращает результат считывания «1» вне зависимости от сигнального состояния на входе запуска. Результат считывания «0» не возвращается до сброса таймера независимо от сигнального состояния на входе запуска.

Используя стартовое значение в качестве начальной точки, значение времени отсчитывается в направлении уменьшения согласно временной базе.

- ❸ Если сигнальное состояние на входе запуска меняется с «0» на «1» (положительный фронт) во время работы таймера, то таймер перезапускается с запрограммированным значением времени (таймер «повторно вызывается»). Он может быть перезапущен любое количество раз, при этом истечение заданного времени не обязательно.

Сброс таймера задержки включения с запоминанием

Обнуление таймера задержки включения с запоминанием имеет статический эффект и получает приоритет перед запуском таймера (рисунок 7.9).

- ❹ ❺ Сигнальное состояние «1» на входе сброса обнуляет таймер независимо от сигнального состояния на входе запуска. Считывание с ожиданием сигнального состояния «1» (состояние таймера) возвращает результат проверки «0». Значение времени и временная база также устанавливаются в нуль.
- ❻ Если сигнальное состояние на входе запуска переходит из «0» в «1» (положительный фронт) при наличии сигнала сброса, то таймер стартует, но немедленно сбрасывается (на диаграмме это отмечено линией). Если проверка состояния таймера запрограммирована после сброса, кратковременный запуск таймера на проверку не повлияет.

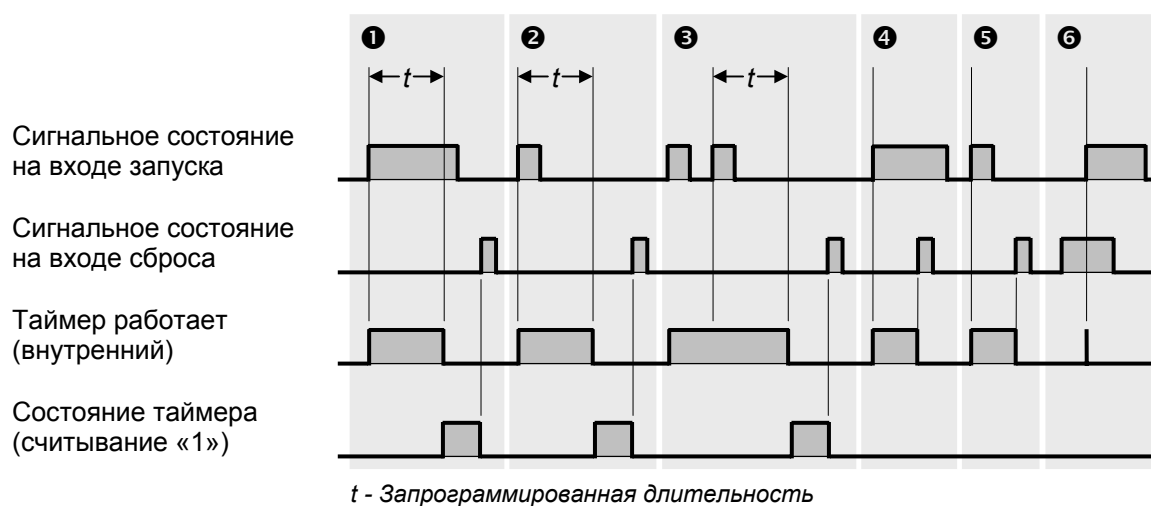


Рисунок 7.9 Рабочие характеристики при запуске и сбросе таймера задержки включения с запоминанием

7.6 Таймер задержки выключения

Запуск таймера задержки выключения

Диаграмма, представленная на рисунке 7.10, описывает рабочие характеристики таймера после его запуска, и когда он сброшен. Описание применимо, если при программировании с использованием отдельных элементов вы соблюдаете порядок, показанный в параграфе 7.1.6 «Последовательность операций с таймерами».

- ①③ Таймер запускается, когда сигнальное состояние на входе запуска таймера изменяется с «1» на «0» (отрицательный фронт). Он функционирует в течение запрограммированной длительности. Считывания с ожиданием сигнального состояния «1» (состояние таймера) возвращают результат проверки «1», когда сигнальное состояние на входе запуска «1», или когда таймер работает (задержка выключения).

Используя стартовое значение в качестве начальной точки, значение времени отсчитывается в направлении уменьшения согласно временной базе.

- ② Если сигнальное состояние на входе запуска меняется с «0» на «1» (положительный фронт) во время работы таймера, то он сбрасывается. Перезапускается таймер только при наличии на входе запуска отрицательного фронта.

Сброс таймера задержки выключения

Обнуление таймера задержки выключения имеет статический эффект и получает приоритет перед запуском таймера (рисунок 7.10).

- ④ Сигнальное состояние «1» на входе сброса таймера во время работы таймера обнуляет его. В этом случае результат считывания с ожиданием сигнального состояния «1» (состояние таймера) равен «0». Значение времени и временная база также обнулены.
- ⑤⑥ Сигнальное состояние «1» на входе запуска и на входе сброса обнуляет бинарный выход таймера. Считывание с ожиданием сигнального состояния «1» (состояние таймера) в этом случае возвращает сигнальное состояние «0». Если сигнальное состояние на входе сброса теперь снова переходит в «0», выход таймера возвращается в состояние «1».
- ⑦ Если сигнальное состояние на входе запуска переходит из «1» в «0» (отрицательный фронт) при наличии сигнала сброса, то таймер стартует, но немедленно сбрасывается (на диаграмме это показано линией). Тогда считывание с ожиданием сигнального состояния «1» (состояние таймера) возвращает результат проверки «0».

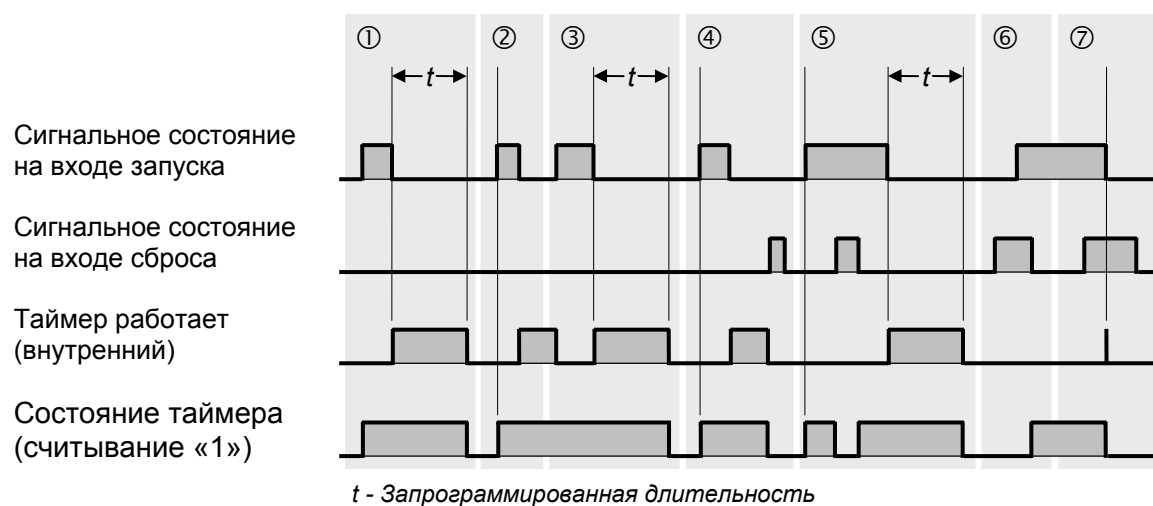


Рисунок 7.10 Рабочие характеристики при запуске и сбросе таймера задержки выключения

7.7 ИЕС-таймеры

ИЕС-таймеры встроены в операционную систему CPU как системные функциональные блоки (блоки SFB).

В некоторых CPU доступны следующие таймеры:

- SFB 3 TP
Импульсный таймер (pulse timer);
- SFB 4 TON
Таймер задержки включения (on-delay timer);
- SFB 5 TOF
Таймер задержки выключения (off-delay timer).

Рисунок 7.11 показывает рабочие характеристики этих таймеров.

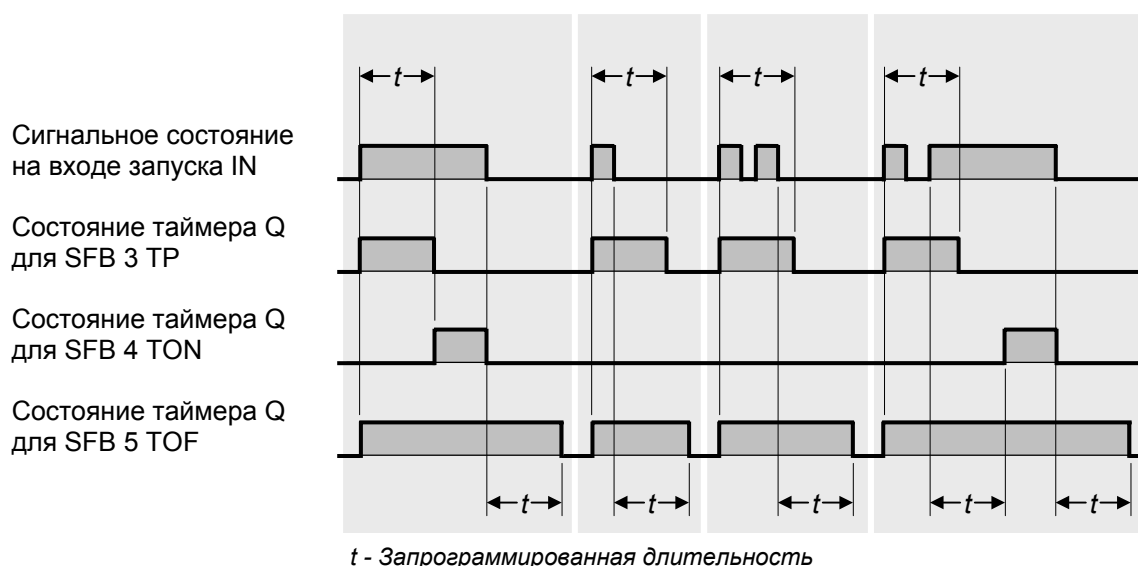


Рисунок 7.11 Рабочие характеристики ИЕС-таймеров

Эти блоки SFB вызываются с экземплярным блоком данных или используются как локальные экземпляры в функциональном блоке. Описание интерфейса для автономного программирования вы найдете в стандартной библиотеке под именем *Standard Library* в программе *System Function Blocks* (Системные функциональные блоки).

Примеры вызовов вы сможете найти на прилагаемой к книге дискете в библиотеках «LAD_Book» и «FBD_Book», функциональный блок FB 107 в программе «Basic Functions» («Основные функции»).

7.7.1 Импульсный таймер SFB 3 TP

Параметры IEC-таймера SFB 3 TP приведены в таблице 7.1.

Таблица 7.1 Параметры IEC-таймеров

Имя	Объявление	Тип данных	Описание
IN	INPUT	BOOL	Вход запуска
PT	INPUT	TIME	Длина импульса или длительность задержки
Q	OUTPUT	BOOL	Состояние таймера
ET	OUTPUT	TIME	Истекшее время (сколько времени прошло)

Когда RLO на входе запуска таймера переходит из «0» в «1», таймер запускается. Он работает в течение запрограммированной длительности вне зависимости от любых последующих изменений в RLO на входе запуска. Сигнальное состояние на выходе Q работающего таймера равно «1».

Выход ET выдает длительность времени для выхода Q. Эта длительность начинается с T#0s и заканчивается на установленной длительности PT. Когда PT истекает, ET остается установленным в значение истекшего времени до перехода входа IN обратно в «0». Если вход IN переходит в «0» до истечения PT, то выход ET переходит в T#0s, текущая длительность PT обнуляется.

Чтобы повторно инициализировать таймер, просто запустите его с PT = T#0s.

Таймер SFB 3 TP активен в режимах CPU START и RUN. Он сбрасывается (инициализируется) при выполнении «холодного» рестарта.

7.7.2 Таймер задержки включения SFB 4 TON

Параметры IEC-таймера SFB 4 TON показаны в таблице 7.1.

Таймер запускается, когда RLO на входе запуска таймера меняется с «0» на «1». Он работает в течение запрограммированной длительности. Выход Q показывает сигнальное состояние «1», когда время истекло, и пока на входе запуска остается сигнальное состояние «1». Если RLO на входе запуска таймера меняется с «1» на «0» до окончания заданного времени, то таймер сбрасывается. Следующий положительный фронт вновь запускает таймер.

Выход ET выдает длительность времени для таймера. Эта длительность начинается с T#0s и заканчивается на установленной длительности PT. Когда PT заканчивается, ET остается установленным в значение истекшего времени до перехода входа IN в «0». Если вход IN обнуляется до истечения PT, то ET немедленно переходит в T#0s.

Чтобы повторно инициализировать таймер, просто запустите его с PT = T#0s.

Таймер SFB 4 TON активен в режимах CPU START и RUN. Он сбрасывается при «холодном» рестарте.

7.7.3 Таймер задержки выключения SFB 5 TOF

Параметры IEC-таймера SFB 5 TOF перечислены в таблице 7.1.

На выходе Q присутствует сигнальное состояние «1», когда RLO на входе запуска таймера меняется с «0» на «1». Таймер запускается, когда RLO на входе запуска вновь обнуляется. Пока таймер работает, выход Q сохраняет сигнальное состояние «1».

Выход Q сбрасывается, когда время истекает. Если RLO на входе запуска переходит опять в «1» до истечения времени, то таймер сбрасывается, и выход Q остается установленным в «1».

Выход ET выдает длительность времени для таймера. Эта длительность начинается с T#0s и заканчивается на установленной длительности PT. Когда время, заданное на входе PT, заканчивается, ET остается установленным в значение истекшего времени до перехода входа IN в «1». Если вход IN устанавливается в «1» до истечения PT, то ET немедленно переходит в T#0s.

Чтобы повторно инициализировать таймер, просто запустите его с PT = T#0s.

Таймер SFB 5 TOF активен в режимах CPU START и RUN. Он сбрасывается при «холодном» рестарте.