

Работа с "быстрыми" входами и выходами из прерывания высокочастотного таймера

Руководство по применению

Общие сведения.

Некоторые контроллеры ОВЕН ПЛК имеют встроенный таймер, по прерыванию которого может быть вызван отдельная программа (POU), не связанная с выполнением основной программы ПЛК. Минимальный период вызовов прерываний таймера составляет 20мкс и может быть увеличен при вызове функции инициализации. Период должен быть кратен 20мкс. В POU, вызываемому по этому прерыванию, могут обрабатываться состояния "быстрых" входов и выходов ПЛК. Такой режим обработки может потребоваться для задач, время обработки которых должно быть существенно меньше времени цикла ПЛК или для автоматизации объектов, критичных ко времени реакции на определенные события.

К "быстрым" входам и выходам относятся несколько первых входов и выходов на ПЛК110. Подробно о количестве "быстрых" входов и выходов смотри руководство по эксплуатации на контроллер.

Порядок создания и подключения POU обработки прерывания таймера.

Рассмотрим порядок действий на примере. Пусть имеется объект, требующий выключение исполнительного механизма при замыкании дискретного датчика. Датчик подключен к "быстрому" входу 1 ПЛК110, исполнительный механизм управляется "быстрым" выходом 1 того же контроллера. Пусть также требуется вычислять, сколько таких переключений произошло между началами основных циклов ПЛК.

Шаг 1. Создание проекта.

Необходимо создать проект под нужный контроллер. В нашем примере используется контроллер ПЛК110-220.60.К-М, основная программа PLC_PRG будет написана на языке ST.

Шаг 2. Создание программы (POU), выполняющегося при прерывании таймера.

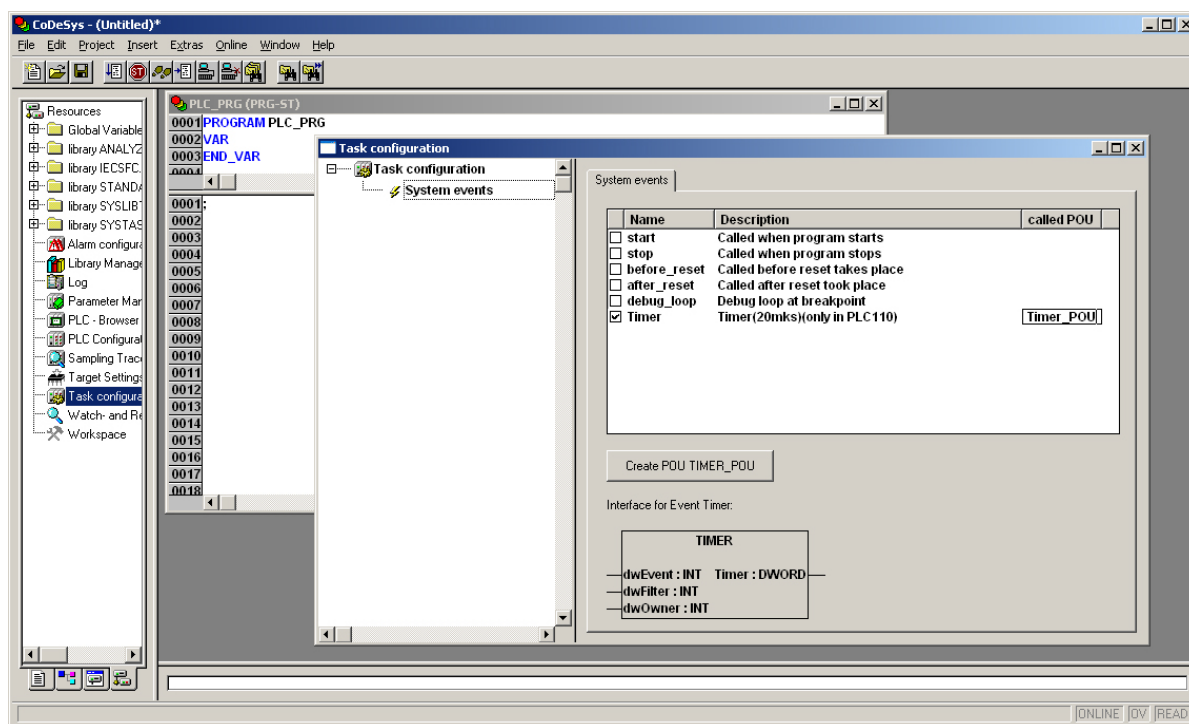
Создание POU обработки прерывания таймера делается во вкладке POU в среде программирования. Для создания POU необходимо правой кнопкой мыши вызвать контекстное меню, выбрать пункт "Add Object...". В открывшемся окне задать тип, имя и язык написания POU.

Внимание! POU должен быть типа Program, иные типы не работают.

Имя и язык написания могут быть любыми, в нашем примере выбран язык ST и дано имя Timer_POU.

Шаг 3. Подключение вызова программы к системному событию (прерыванию) таймера.

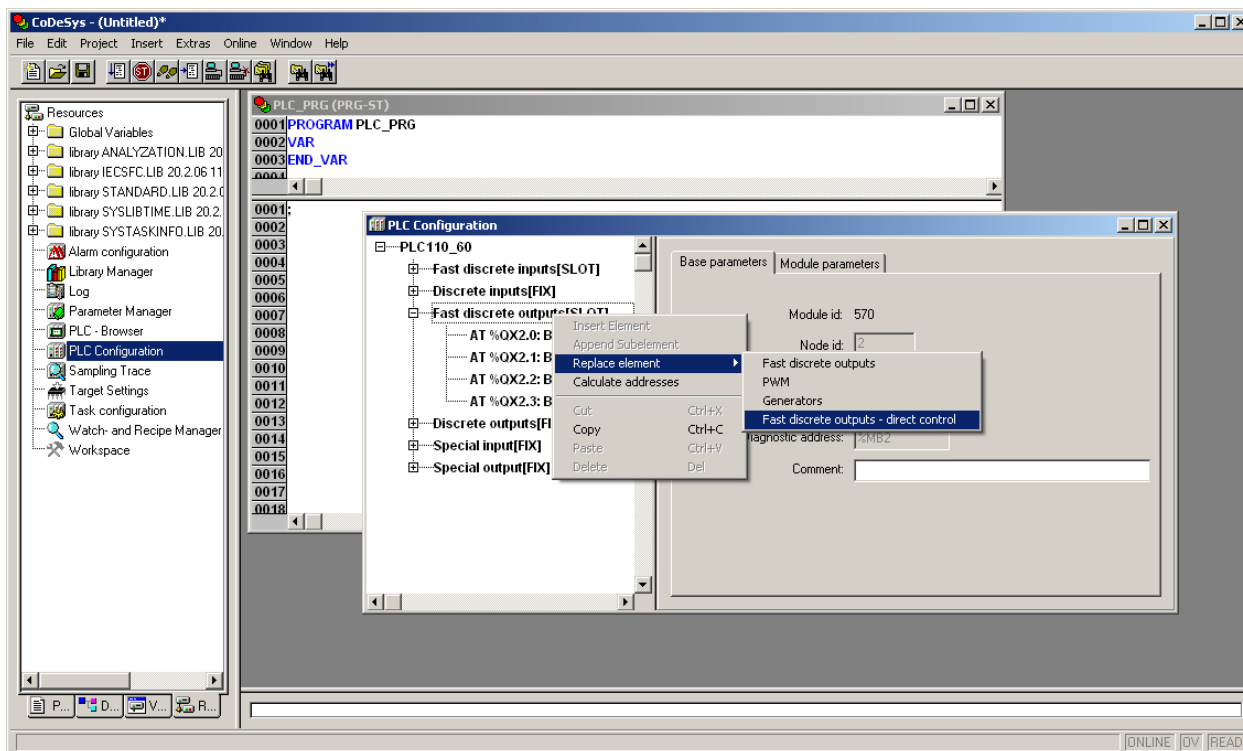
Для подключения POU к прерыванию таймера необходимо перейти в окно **Task Configuration**. В списке **System events** отобразится доступный для этого контроллера список прерываний. Необходимо выбрать прерывание **Timer** (поставить "галочку") и в колонке **called POU** указать имя созданного POU.



Внимание! Использовать кнопку **Create POU** не следует, т.к. она создает POU типа **Function**, а для работы требуется тип **Program**.

Шаг 4. Перевод "быстрых" дискретных входов и выходов в режим прямого управления из POU.

Т.к. прерывание таймера вызывается в несколько десятков раз чаще, чем происходит цикл ПЛК, то использовать в POU обработки прерываний таймера значения из области памяти ввода/вывода не имеет смысла. Для исключения ситуации, когда одним и тем же входом или выходом управляет POU обработки прерывания таймера и основная программа ПЛК, необходимо перевести "быстрые" входы и выходы в специальный режим: режим прямого управления (**Direct Control mode**). В этом режиме для "быстрых" входов и выходов не выделяется каналов в памяти ввода/вывода. Для перевода выходов в режим прямого управления необходимо в окне **PLC Configuration** заменить модуль **Fast Discrete Outputs** на модуль **Fast Discrete Outputs – Direct Control**. Аналогично для перевода входов в режим прямого управления необходимо заменить модуль **Fast Discrete Inputs** на модуль **Fast Discrete Inputs – Direct Control**.

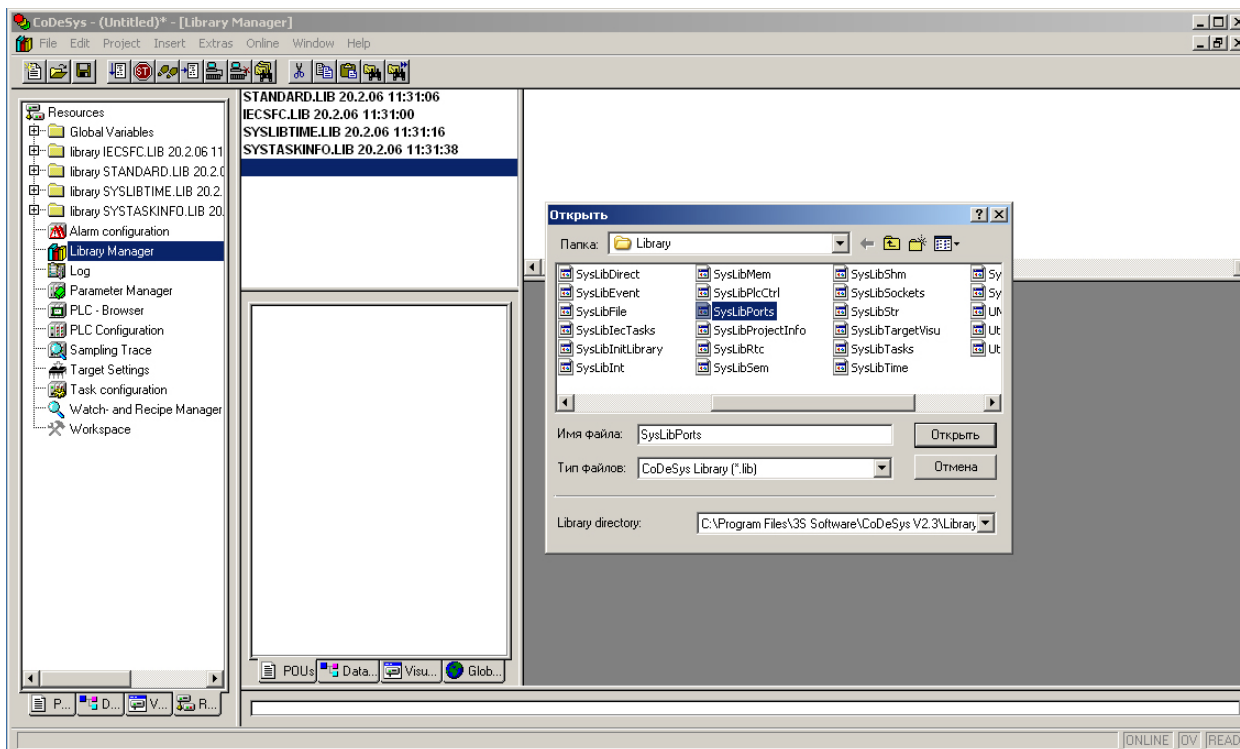


Шаг 5. Подключение библиотек Timer и SysLibPorts.

Запуск работы таймера производится из основной программы посредством вызова функции, входящей в состав библиотеки **Timer.lib** (библиотека OVEN, поставляется на CD в комплекте с контроллером).

Работа с "быстрыми" входами и выходами из POU обработки прерываний таймера, как уже отмечалось, невозможна через пространство памяти ввода/вывода. Она должна происходить посредством вызова функций библиотеки **SysLibPorts.lib** (библиотека CoDeSys, поставляется на CD в комплекте с контроллером).

Подключите обе библиотеки в окне **Library Manager**. Библиотеки предварительно необходимо скопировать на жесткий диск компьютера (рекомендуется в папку c:\Program Files\3S Software\CoDeSys V2.3\Library\).



Шаг 6. Написание POU обработки высокочастотного таймера.

Перейдя в список POU откройте нужный и напишите программу обработки прерывания таймера.

Внимание: В POU обработки прерывания таймера не допускается выполнения сложных математических действий или вызов ресурсоемких функций, т.к. они могут не успеть обработаться до следующего вызова прерывания таймера.

В нашем примере программа состоит из чтения состояния "быстрых" входов функцией **SysPortIn** (из библиотеки SysLibPorts.lib), инвертировании прочитанного значения и передаче его в "быстрые" выхода функцией **SysPortOut**. Для той и для другой функции работа ведется с портом 0, с младшими битами. Число ликвидных бит равно числу "быстрых" входов и выходов используемого контроллера.

Для передачи информации о количестве срабатываний "быстрых" входов между вызовами циклов ПЛК из обработчика прерывания в основную программу используются глобальные переменные. При работе с глобальными переменными необходимо придерживаться следующего правила: запись значения в глобальную переменную должна происходить только в одном месте, либо в POU обработки прерывания, либо в основной программе.

Проект с примером под названием **hi_timer.pro** представлен на CD, входящем в комплект поставки. Перед открытием проекта его необходимо скопировать на жесткий диск компьютера.

Внимание: При отладке прерывания таймера запрещается использовать точку останова (breakpoint), т.к. это приведет к зависанию контроллера.

