

## Библиотека SysLibCallback.lib

---

Данная библиотека позволяет активировать пользовательские функции обратного вызова для событий системы исполнения. Она содержит функции SysCallbackRegister и SysCallbackUnregister.

Обе функции имеют тип BOOL и возвращают TRUE, если соответствующая функция обратного вызова успешно зарегистрирована или отключена, соответственно. Функции выполняются синхронно.

**Прототип функции обратного вызова должен выглядеть так:**

```
FUNCTION Callback : DWORD
```

```
VAR_INPUT
```

```
    dwEvent: DWORD;    (* Событие *)
    dwFilter: DWORD;   (* Фильтр *)
    dwOwner: DWORD;   (* Источник *)
```

```
END_VAR
```

---

**Внимание:** в целевых платформах RISC и Motorola 68K имя функции обратного вызова должны начинаться с „callback“!

---

Библиотечные функции SysCallbackRegister и SysCallbackUnregister имеют одинаковый набор параметров:

Входные переменные	Тип данных	Описание
iPOUIndex	INT	POU индекс функции обратного вызова, может быть получен при помощи оператора INDEXOF(<имя функции >).
Event	RTS_EVENT	Событие в системе исполнения, по которому должна вызываться callback-функция. Определяется значением перечисления RTS_EVENT, определенного в данной библиотеке (см. ниже).

Перечисление RTS\_EVENT определено следующим образом:

```
TYPE RTS_EVENT :
```

```
(
```

```
    EVENT_ALL,
```

```
    (* Общие события *)
```

```
    EVENT_START,
```

```
    EVENT_STOP,
```

```
    EVENT_BEFORE_RESET,
```

```
    EVENT_AFTER_RESET,
```

```
    EVENT_SHUTDOWN,
```

(\* Исключения системы исполнения \*)

EVENT_EXCPT_CYCLETIME_OVERFLOW,	(* Переполнение времени цикла *)
EVENT_EXCPT_WATCHDOG,	(* Сработал программный сторожевой таймер МЭК задачи *)
EVENT_EXCPT_HARDWARE_WATCHDOG,	(* Сработал аппаратный сторожевой таймер. Фатальная ошибка. *)
EVENT_EXCPT_FIELDBUS,	(* Ошибка Fieldbus *)
EVENT_EXCPT_IOUPDATE,	(* Ошибка обновления входов-выходов *)

(\*Исключения выполнения \*)

EVENT_EXCPT_ILLEGAL_INSTRUCTION,	(* Недопустимая инструкция *)
EVENT_EXCPT_ACCESS_VIOLATION,	(* Ошибка доступа *)
EVENT_EXCPT_PRIV_INSTRUCTION,	(* Привилегированная операция *)
EVENT_EXCPT_IN_PAGE_ERROR,	(* Ошибочная страница *)
EVENT_EXCPT_STACK_OVERFLOW,	(* Переполнение стека *)
EVENT_EXCPT_MISALIGNMENT,	(* Ошибка размещения по типу данных *)
EVENT_EXCPT_ARRAYBOUNDS,	(* Выход за границы массива *)
EVENT_EXCPT_DIVIDEBYZERO,	(* Деление на 0 *)
EVENT_EXCPT_OVERFLOW,	(* Переполнение *)
EVENT_EXCPT_NONCONTINUABLE,	(* Авария системы, продолжение работы невозможно *)
EVENT_EXCPT_NO_FPU_AVAILABLE,	(* Мат. сопроцессор (FPU) недоступен *)
EVENT_EXCPT_FPU_ERROR,	(* FPU: Неопределенная ошибка *)
EVENT_EXCPT_FPU_DENORMAL_OPERAND,	(* FPU: Ошибочный операнд *)
EVENT_EXCPT_FPU_DIVIDEBYZERO,	(* FPU: Деление на 0 *)
EVENT_EXCPT_FPU_INVALID_OPERATION,	(* FPU: Ошибочная операция *)
EVENT_EXCPT_FPU_OVERFLOW,	(* FPU: Переполнение*)
EVENT_EXCPT_FPU_STACK_CHECK,	(* FPU: Перегрузка стека *)

(\* События входов-выходов \*)

EVENT_AFTER_READING_INPUTS,
EVENT_BEFORE_WRITING_OUTPUTS,

(\* Прочие события \*)

EVENT\_TIMER, (\* Прерывание по таймеру\*)  
EVENT\_DEBUG\_LOOP, (\* Отладка по точке останова \*)

(\* Онлайн сервисы \*)

EVENT\_ONLINE\_SERVICES\_BEGIN := 500,  
EVENT\_LOGIN,  
EVENT\_CUSTOM\_SERVICES,

(\* Прерывания \*)

EVENT\_INT\_0:=1000,  
EVENT\_INT\_1,  
EVENT\_INT\_2,  
EVENT\_INT\_3,  
EVENT\_INT\_4,  
EVENT\_INT\_5,  
EVENT\_INT\_6,  
EVENT\_INT\_7,  
EVENT\_INT\_8,  
EVENT\_INT\_9,  
EVENT\_INT\_10,  
EVENT\_INT\_11,  
EVENT\_INT\_12,  
EVENT\_INT\_13,  
EVENT\_INT\_14,  
EVENT\_INT\_15,  
EVENT\_INT\_255:=1255,

EVENT\_MAX

);

END\_TYPE