

Библиотека SysLibPLCConfig.lib

Данная библиотека поддерживает чтение конфигурации ПЛК (PLC Configuration). Данные о конфигурации загружаются в контроллер и используются системой исполнения, размещенные в специальных структурах.

Функции данной библиотеки дают непосредственный доступ к указателям на внутренние структуры системы исполнения. По этой причине:

- Структуры нельзя изменять (указатели на под элементы)!
- Если изменить значения параметров в структуре, это повлияет на работу системы исполнения!

Если поддерживается в системе исполнения, то доступны следующие функции данной библиотеки:

- CfgCCGetError
- CfgCCGetHeader
- CfgCCGetRootModule
- CfgCCGetRootModuleByModuleId
- CfgCCGetRootModuleByNodeId

Выполнение функций синхронное.

CfgCCGetError

Внимание: пока не реализована в системе исполнения. Всегда возвращает код ошибки 0.

Функция сообщает информацию об ошибках произошедших в процессе загрузки конфигурации.

Возвращает указатель на структуру CCLoadError.

Структура CCLoadError:

TYPE CCLoadError :

STRUCT

```

    ulLastError: UDINT;           (* Код последней ошибки *)
    ulAddInfo1: UDINT;           (* Дополнение к ulLastError *)
    ulAddInfo2: UDINT;           (* Дополнение к ulLastError *)
    szLastError: STRING(32);     (* Отладочное сообщение *)

```

END_STRUCT

END_TYPE

CfgCCGetHeader

Данная функция возвращает указатель на заголовочную структуру PLC configuration CCHheader.

Структура CCHheader:

TYPE CCHheader :

STRUCT

```

    szTag: STRING(10);    (*Нуль terminated строка "CommConf"*)
    cByteOrder: BYTE;    (* Порядок данных intel ('I') или motorola ('M')*)
    ulSize: UDINT;       (* Размер данных *)
    IVersion: UDINT;     (*Версия файла данных *)

```

END_STRUCT

END_TYPE

CfgCCGetRootModule

Данная функция возвращает указатель на информационную структуру CModule корневого модуля PLC configuration.

Структура CModule:

TYPE CModule :

STRUCT

```

    ucEntryTag: BYTE;          (* 'M' = Модуль *)
    ucDummy1: BYTE;
    ucDummy2: BYTE;
    ucDummy3: BYTE;
    ulModuleId: UDINT;         (* Id модуля, заданное в *.cfg файле *)
    sModuleNumber: UINT;      (* Номер модуля в корневом модуле (-1 для корневого модуля) *)
    usModuleTag: UINT;        (* Тип модуля ( 0=3S-Модуль, 1=DP-Master, 2=DP-Slave, 3=CAN-
    Master, 4=CAN-Slave, 5=DP-SingleSlave) *)
    byDeviceDriver: BYTE;     (* Модуль требует драйвер устройства (0=FALSE, 1=TRUE)*)
    ucDummy4: BYTE;
    ucDummy5: BYTE;
    ucDummy6: BYTE;
    ulNodeId: UDINT;          (* Node Id модуля *)
    byDefinedWithStruct: BYTE; (* Модуль определен структурой (0=FALSE, 1=TRUE) *)
    ucDummy7: BYTE;

```

ucDummy8: BYTE;

ucDummy9: BYTE;

ulBitOffsetInput: UDINT; (* Смещение области входов модуля *)

ulBitSizeInput: UDINT; (* Размер области входов модуля (в bit) *)

ulBitOffsetOutput: UDINT; (* Смещение области выходов модуля *)

ulBitSizeOutput: UDINT; (* Размер области выходов модуля (в bit) *)

ulRefIdCommonDiag: UDINT; (* RefId диагностической области модуля *)

ulBitOffsetCommonDiag: UDINT; (* Смещение диагностической области модуля *)

ulBitSizeDiag: UDINT; (* Размер диагностической области модуля *)

usParameterCount: UINT; (* Число параметров *)

usDummy: UINT;

prccpModuleParams: POINTER TO POINTER TO ccParam; (* <ccParam [0..usParameterCount]> указатель на массив указателей на CCModuleParam структуры. (Описание структуры CCParam см ниже). Разыменовывание указателя prccpModuleParams^ дает указатель на первый параметр структуры. (prccpModuleParams+4)^ дает указатель на следующий параметр. *)

ulSizeOfSpecificData: UDINT; (* Размер в байтах области специфических данных модуля *)

pModuleData: POINTER TO BYTE; (* <MODULE_SPECIFIC_DATA> специфические данные модуля, соответствующие usModuleTag: pModuleData может содержать указатель на PBSlave, CANSlave, PBMaster, PBSlave, PBSingleSlave, см ниже *)

usChannelCount: UINT; (* Число сконфигурированных каналов *)

usModuleCount: UINT; (* Число сконфигурированных модулей *)

(* Каналы и модули описываются в порядке их конфигурирования! (DP-Slaves идут в порядке номеров станций).*)

prcccChannels: POINTER TO POINTER TO ccChannel; (* <ccChannel [0..usChannelCount]> определение структуры CCChannel см. ниже. Разыменовывание указателя prcccChannels^ дает указатель на первый параметр структуры. (prcccChannels+4)^ дает указатель на следующий параметр *)

```

ppccmSubModules:      (* <ccModule [0..usModuleCount]> указатель на массив переменных
POINTER TO POINTER TO  типа POINTER TO ccModule. Для доступа к содержимому,
BYTE;                 необходимо присвоить значение переменной типа "POINTER TO
                       CCModule". Определение структуры CCModule см. ниже.
                       Разыменовывание указателя ppccpSubModules^ дает указатель на
                       первый параметр структуры. (ppccpSubModules+4)^ дает указатель
                       на следующий параметр. *)

```

```

END_STRUCT

```

```

END_TYPE

```

Структура CCChannel:

```

TYPE CCChannel :

```

```

STRUCT

```

```

    ucEntryTag: BYTE;      (* 'C' = Канал *)
    ulChannelId: UDINT;    (* Id канала в соответствии с конф. файлом *)
    usChannelNumber: UINT; (* Номер канала в модуле *)
    ulRefId: UDINT;        (* Направление канала (1=вход, 2=выход, 3=вход-выход) *)
    usChannelType: UINT;   (* Тип TYPE канала (кодировка как в CoDeSys "TypeClass") *)
    ulBitOffset: UDINT;    (* Смещение области ввода/вывода канала *)
    usParameterCount: UINT; (* Число параметров *)
    ppccpParams: POINTER TO (* PARAMETER[1..usParameterCount]> указатель на массив
    POINTER TO CCParam;     указателей на структуры CCParam. (См. ниже) *)

```

```

END_STRUCT

```

```

END_TYPE

```

Структура CCParam:

```

TYPE CCParam :

```

```

STRUCT

```

```

    ulParameterId: UDINT; (* Id параметра в *.cfg файле *)
    usParameterNumber: UINT; (* Номер параметра *)
    byReadOnly: BYTE;      (* 1=TRUE, 0=FALSE *)
    byDummy: BYTE;

```

```

usParameterType: UINT;      (* Тип параметра (CoDeSys "TypeClass") *)

usDummy: UINT;

ulSize: UDINT;             (* Размер параметра в байтах *)

byValue: BYTE;            (* Первый байт представления данных параметра. Если
параметр имеет более 1 байта то, прочие данные идут следом. *)

END_STRUCT

END_TYPE

```

CfgCCGetRootModuleByModuleId

Функция обеспечивает информацию о корневом модуле текущей PLC configuration, указанном с помощью Id модуля. Id определен в поле "Id" конфигурационного файла, см. документ *PLC_Configuration_E.pdf*.

Функция возвращает указатель на структуру CModule (см. выше функцию CfgCCGetRootModule)

Переменная	Тип данных	Описание
ulModuleId	UDINT	Id модуля

CfgCCGetRootModuleByNodeId

Функция обеспечивает информацию о корневом модуле текущей PLC configuration, указанном с помощью node Id. Node Id определяется позицией модуля в PLC configuration. Детали см. в документе *PLC_Configuration_E.pdf*.

Функция возвращает указатель на структуру CModule (см. выше функцию CfgCCGetRootModule)

Переменная	Тип данных	Описание
ulNodeId	UDINT	Node Id модуля