File System - USB management

Contents

Definitions

- **BIOS** is the synonym for **Firmware** in **FREE Studio**
- **USB key** means a standard pen drive.
- **Type A USB (HOST)**. Used to connect a standard USB to download the application/BIOS.
- **Type B mini USB (DEVICE)**. Used to connect FREE Evolution to a PC or third party device via mini A/B USB cable to up/download the application, files or documentation. This can be done from a PC or other device. [1]
- **target** is the synonym of **Evolution** in **FREE Studio**

Description

This document File System Application Notes is related to **FREE Evolution platform /U models only**

In /U models there are 2 USB connectors inside the door on the left of the LEDs, on the top part of the cap.

Evolution can be connected to a PC through the mini USB port and a USB cable.

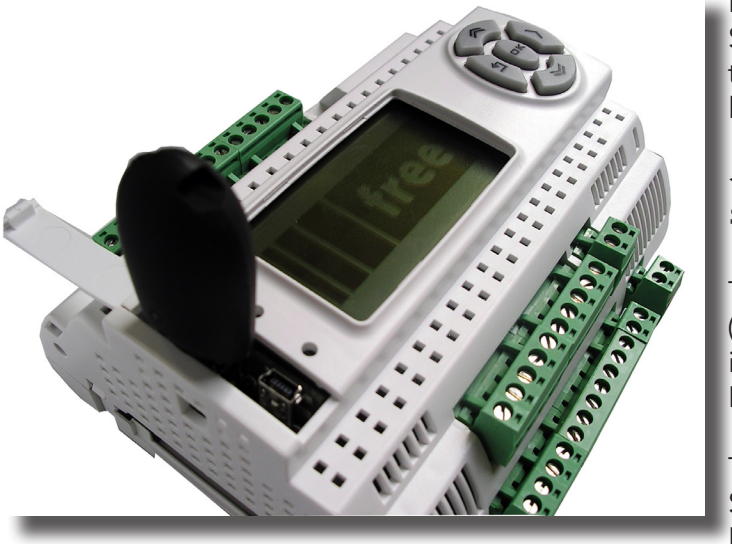
In this case the PC will recognize the Evolution as an external drive (FREE Studio cannot communicate through USB cable)

Please Note: the two USBs should not be used at the same time.

Type B mini USB (DEVICE)

1 **N.B.: compatible with Windows XP Home and Professional, Windows 2000 , Windows Vista and Windows 7 operating systems. USB formatted FAT32**

Eliwell Controls s.r.l.
Via dell'Industria, 15 • Zona Industriale Paludi • 32010 Pieve d'Alpago (BL) ITALY
Telephone +39 0437 986 111 • Facsimile +39 0437 989 066
Technical helpline +39 0437 986 250 • E-mail eliwell.freeway@invensys.com
www.eliwell.it



Type A USB (HOST)

Important note for developers using FREE Studio: FREE Studio manages the use of the USB serial via the **fs_iec.pll** software library available @

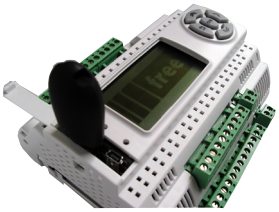
`<C:\Programs>\Eliwell\free Studio\Catalog\FreeEvolution\PLC`

The library also contains target functions (target blocks) to be used to manage files in the internal Evolution memory (see FREE Studio manual for details[2]).

To download the IEC applications of Studio from a personal computer to the Evolution target device, several additional modules are necessary.

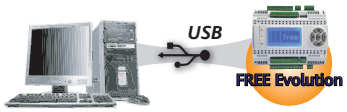
The tables below show possible operations:

CASE 1
USB Host
USB → ← FREE



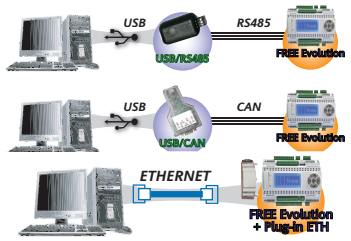
Data downloading direction	→	←
Parameter map	✓	✓
IEC application	✓	-
HMI application	✓	-
Data file	✓	✓
BIOS	✓	-

CASE 2
USB device
PC → ← FREE



Data downloading direction	→	←
Parameter map	-	-
IEC application	✓	✓
HMI application	✓	✓
Data file	✓	✓
BIOS	-	-

CASE 3
USB-RS485 /USB-CANOpen
ETHERNET + Plugin
PC → ← FREE



Data downloading direction	→	←
Parameter map	✓	✓
IEC application	✓	-
HMI application	✓	-
Data file	✓	✓
BIOS	✓	-

2 **Press F1 from FREE Studio Application working environment**

Eliwell Controls s.r.l.
Via dell'Industria, 15 • Zona Industriale Paludi • 32010 Pieve d'Alpago (BL) ITALY
Telephone +39 0437 986 111 • Facsimile +39 0437 989 066
Technical helpline +39 0437 986 250 • E-mail eliwell.freeway@invensys.com
www.eliwell.it

USB DEVICE

The memory space of the Evolution (**Flash Memory Data - 128MB**) may contains:

- **PLCIEC.COD** : the file related to the Application project running
- **HMIIEC.COD** : the file related to the User Interface project running
- **HMIREM.COD** : the file related to the Remote User Interface (for EVK terminal if available) project running
- **CONNEC.PAR** : the file related to the Connection project running
- **PARAM.DAT** : parameter map file
- Files created by the application running on the Evolution
- Files copied from a PC to Evolution

Files **PLCIEC.COD, HMIIEC.COD, CONNEC.PAR, PARAM.DAT** could also have a prefix from **00** to **15**. This feature allows to store more than one Evolution application on a USB pen drive (see **USB HOST** paragraph for details).

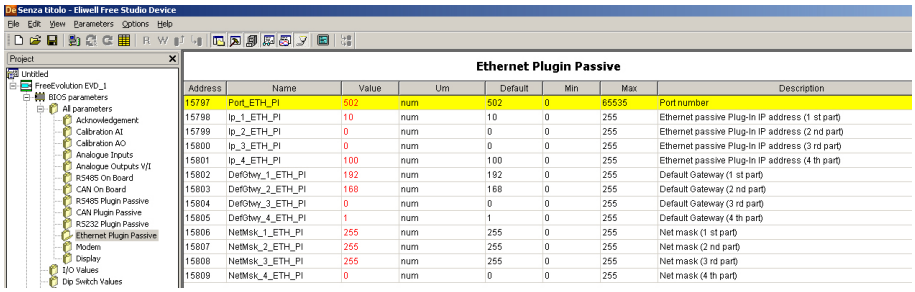
IMPORTANT NOTE: The **.COD / .PAR** files are the current programs running on the Evolution; by deleting and/or updating one or more files, will force an update of the Evolution application at any new Evolution power on.

IMPORTANT NOTE:
PARAM.DAT file includes Evolution BIOS parameters and IEC application.
By renaming PARAM.DAT to PARAM.RAW the upload will skip parameters' range limit check

Example of PARAM.DAT (serials settings):

EVD

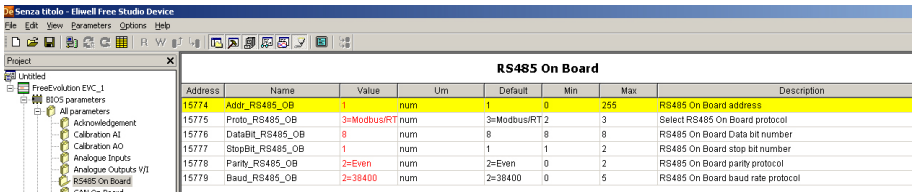
```
Model=42301025
CopyCardCode=174
15798=10
15799=0
15800=0
15801=100
...
```



Ethernet Plugin Passive

EVC

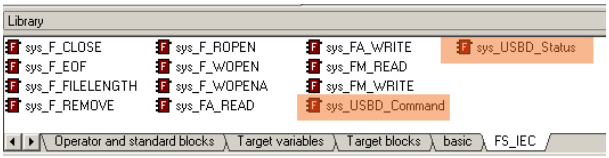
```
Model=47701025
CopyCardCode=194
15775=3
15779=5
15776=8
15777=1
15778=2
15774=1
...
```



RS485 On Board

Eliwell Controls s.r.l.
Via dell'Industria, 15 • Zona Industriale Paludi • 32010 Pieve d'Alpago (BL) ITALY
Telephone +39 0437 986 111 • Facsimile +39 0437 989 066
Technical helpline +39 0437 986 250 • E-mail eliwell.freeway@invensys.com
www.eliwell.it

USB Device access is enabled by default. It can be protected using the following **fs_iec** library functions. The library is available @:



`C:\<Programs>\Eliwell\free Studio\Catalog\FreeEvolution\PLC`[3]

Enable/disable PC host access to File System Function

`Sys_USBD_Command(USINT Command)`[4]`: (* Command: 0=disable, 1=enable *)`

Returns a USINT which could have the following meanings:

```
0     =     Command accepted.
1     =     Command executed but failed.
2     =     Command code non valid.
3     =     Command not executed, function called into task timed.
```

PC host connection status Function

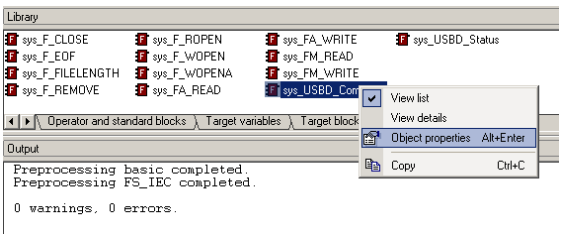
`Sys_USBD_Status(USINT dummy)`

Returns a USINT which could have the following meanings:

```
0     =     USB device Disconnected.
1     =     USB device Connected.
2     =     USB device Suspended.
3     =     Command not executed, function called into task timed.
```

The same library contains a set of target blocks that can be used to manage files in the internal memory of the Evolution.

3 To add the library select Project > Library Manager and add the library from the relevant path
4 Object properties are shown in FREE Studio / Libray Tab by mouse right-click on the related function and selecting "Object properties" (Alt+Enter)



Eliwell Controls s.r.l.
Via dell'Industria, 15 • Zona Industriale Paludi • 32010 Pieve d'Alpago (BL) ITALY
Telephone +39 0437 986 111 • Facsimile +39 0437 989 066
Technical helpline +39 0437 986 250 • E-mail eliwell.freeway@invensys.com
www.eliwell.it

USB HOST

A USB pen drive can be connected to Evolution in order to:

- Upload an application from the pen drive to Evolution
- Download a parameter map from Evolution to the pen drive

The Upload/Download of files can be managed via IEC code using the target var `sysUsbCommand:`

System command to upload/download to/from USB-Host

```
System command to upload/download to/from USB-Host
7     =     load PARAM.BIN from USBH
8     =     load PLCIEC.COD from USBH
9     =     load HMIIEC.COD from USBH
10    =     load PARAM.DAT from USBH
11    =     save PARAM.DAT to USBH
12    =     load CONNEC.PAR from USBH
13    =     load HMIREM.KBD from USBH
14    =     save sysUsbFileName file to USBH, file name can be name. ext or *.ext
15    =     load sysUsbFileName file from USBH, file name can be name.ext or *.ext
```


Eliwell Controls s.r.l.
Via dell'Industria, 15 • Zona Industriale Paludi • 32010 Pieve d'Alpago (BL) ITALY
Telephone +39 0437 986 111 • Facsimile +39 0437 989 066
Technical helpline +39 0437 986 250 • E-mail eliwell.freeway@invensys.com
www.eliwell.it

The status of the USB can be monitored through the target var `sysUsbStatus:`

System status of operation on USB-Host

```
System status of operation on USB-Host
0     =     command completed
1     =     command processing
255   =     command failed
254   =     file not present
253   =     file too long
252   =     USBH not connected
251   =     file not compatible
250   =     some parameters fails
249   =     write file failed
248   =     open file in write failed
```


Eliwell Controls s.r.l.
Via dell'Industria, 15 • Zona Industriale Paludi • 32010 Pieve d'Alpago (BL) ITALY
Telephone +39 0437 986 111 • Facsimile +39 0437 989 066
Technical helpline +39 0437 986 250 • E-mail eliwell.freeway@invensys.com
www.eliwell.it

Uploading automatically an application via USB pen drive

- Copy into a pen drive the COD/PAR/DAT files
- Edit an UPLOAD.TXT file containing the list of the files to be uploaded

An example of **UPLOAD.TXT** is:

```
; Application[5]
PLCIEC.COD
; User Interface
HMIIEC.COD
; Connection
CONNEC.PAR
```

The upload file can have a prefix from 00 to 15, for example 03UPLOAD.TXT:

```
; Application
03PLCIEC.COD
; User Interface
03HMIIEC.COD
; Connection
03CONNEC.PAR
; Parameters
03PARAM.DAT
```

- Copy into a pen drive the UPLOAD.TXT (03UPLOAD.TXT) files as well

Files with numeric prefix are uploaded only if the Evolution dip-switches match the prefix; in this way it is possible to store on the same USB pen drive one or more Evolution applications.

The upload process starts when the pen drive is plugged and can be monitored through the led status which, during the upload process, are controlled directly by Evolution bios.

LED		Upload
RED	Blinking 2 seconds	Failed
YELLOW	On	Underway
GREEN	Blinking	Completed successfully

5 **Lines starting with ; are comment lines.**

Eliwell Controls s.r.l.
Via dell'Industria, 15 • Zona Industriale Paludi • 32010 Pieve d'Alpago (BL) ITALY
Telephone +39 0437 986 111 • Facsimile +39 0437 989 066
Technical helpline +39 0437 986 250 • E-mail eliwell.freeway@invensys.com
www.eliwell.it

The process results which will switch on the red led are the ones related to a value of `sysUsbStatus>1`.
After the process, Evolution must be restarted in order to run the new application.

File PARAM.DAT is uploaded by an Evolution only if the Bios Mask and Par_POLI[6] of the Evolution that has generated the PARAM.DAT are the same as the destination Evolution. The parameters' map update does not require to switch off Evolution.

PROGRAMMING EVOLUTION VIA USB

Let's consider two FREE Evolutions, the former called **Evolution A** and the latter **Evolution B**
The **Evolution B** can be programmed exactly in the same way of an **Evolution A** as described:

- export the PARAM.DAT file related to the application running on **Evolution A** to a USB pen drive
- copy the .COD / .PAR / .DAT files from **Evolution A** into a PC

Option 1

- Upload the .COD / .PAR files from PC to **Evolution B**
- Wait the end of the upload process
- Switch off/on **Evolution B**
- Edit the file UPLOAD.TXT containing:

```
; Parameters
PARAM.DAT
```

- Copy PARAM.DAT into a USB pen drive
- Connect the USB pen drive to **Evolution B**

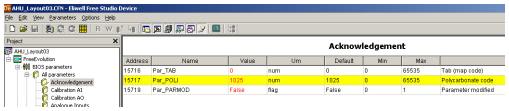
Option 2

- Rename PLCIEC.COD as 01PLCIEC.COD
- Rename HMIIEC.COD as 01HMIIEC.COD
- Rename CONNEC.PAR as 01CONNEC.PAR
- Rename PARAM.DAT as 01PARAM.DAT
- Copy the 4 files into a USB pen drive
- Edit the file 00UPLOAD.TXT containing:

```
; Application
00PLCIEC.COD
; User Interface
00HMIIEC.COD
; Connection
00CONNEC.PAR
```

and save it into a USB pen drive

6 **Par_POLI is visible in FREE Studio Device > Frreevolution > BIOS Parameters > Acknoledgement**



Eliwell Controls s.r.l.
Via dell'Industria, 15 • Zona Industriale Paludi • 32010 Pieve d'Alpago (BL) ITALY
Telephone +39 0437 986 111 • Facsimile +39 0437 989 066
Technical helpline +39 0437 986 250 • E-mail eliwell.freeway@invensys.com
www.eliwell.it

Edit the file 01UPLOAD.TXT containing:

```
; Parameters
01PARAM.DAT
```

and save it into a USB pen drive

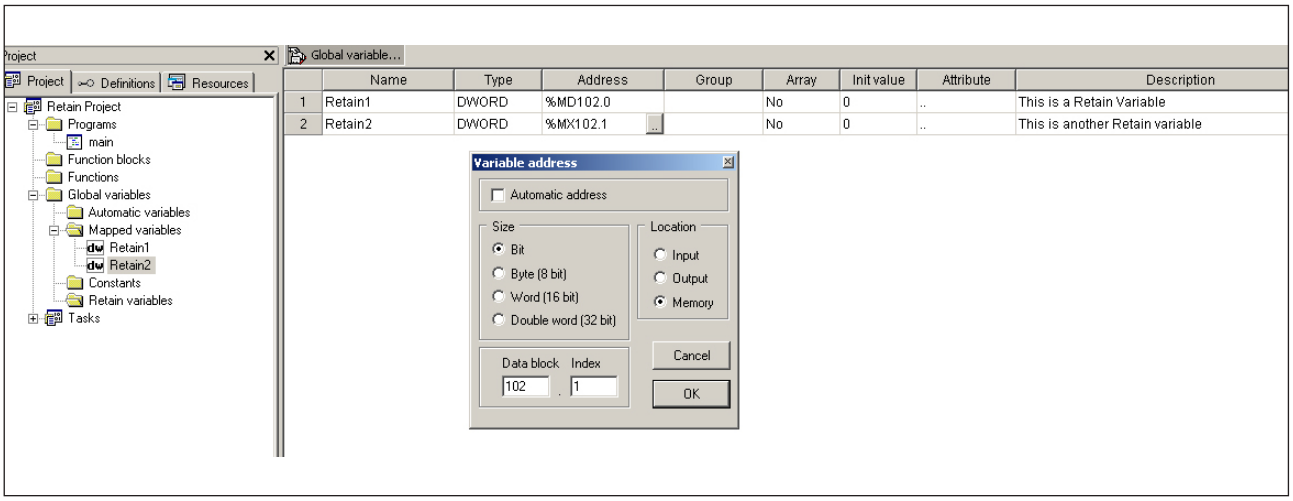
- Set to 0 the dip-switch address of Evolution
- Connect the USB pen drive to **Evolution B**
- Wait the end of the upload process of 00*.* files
- Set to 1 the dip-switch address of Evolution
- Switch off/on **Evolution B**
- Wait the end of the upload process of 01PARAM.DAT

NOTE: FREE Studio creates .COD&.PAR files and the param.bin file; these 4 files allow to upload in "one shot" an application and the related parameter map.

Retain variables

A RETAIN variable indicates that the variables within the structuring elements are retentive, i.e. they keep their value even after the target device has been reset or switched off.
To create a new RETAIN variable follow these steps:

- create a new variable
- set as attribute '...' (DO NOT set RETAIN attribute)
- set as variable address size dw (double WORD) and data block 102.0.xx where xx=0,...99
- variable will be saved into the Mapped variables folder



FREE Studio developer can create up to 100 'retain variables' ensuring their data will not be lost after a shutdown[7]. Retain variable values can be changed several times without affecting internal memory performance.

7 backup memory last approx 1 year

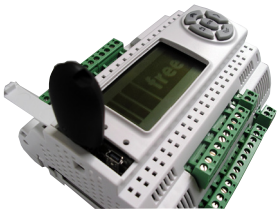
Eliwell Controls s.r.l.
Via dell'Industria, 15 • Zona Industriale Paludi • 32010 Pieve d'Alpago (BL) ITALY
Telephone +39 0437 986 111 • Facsimile +39 0437 989 066
Technical helpline +39 0437 986 250 • E-mail eliwell.freeway@invensys.com
www.eliwell.it

Please Note: RETAIN variables cannot be displayed in the Watch window

Download BIOS[8]

You can update the **Evolution BIOS** from

- **USB pen drive - CASE 1**
- **FREE Studio Device - CASE 3**



Download BIOS USB→target

If you have FREE Studio intalled on your PC, BIOS is available @
`<C:\Programs>\Eliwell\free Studio\Catalog\FreeEvolution\<firmware>`
`<firmware> = firmware423 for EVD, firmware477 for EVC`

Alternatively download **.bin** file from Eliwell Web Site - FREE Firmware Update section @
http://www.eliwell.it/filedownload.aspx?id=20656
Please Note: for registered users only you shall have been authorized by Eliwell to access this area

- Copy the relevant **.bin** file into a USB pen drive (e.g. msk423_11.bin)
- Connect USB pen drive to **Evolution**[9]
- BIOS will be downloaded into **Evolution**

Yellow LED will blink during download. When completed green LED will blink twice and switch ON to confirm successfully download

- Remove USB pen drive
- **Evolution** will automatically reset and will reboot

Please Note: a SYSTEM FAULT message will appear - DO NOT CONSIDER -
BIOS upgrade has been completed successfully

Please Note: BIOS download is error-proof. **Evolution** will not upload non-compliant BIOS
You cannot download **BIOS** for **EVC** or **Smart** into a **EVD** and viceversa

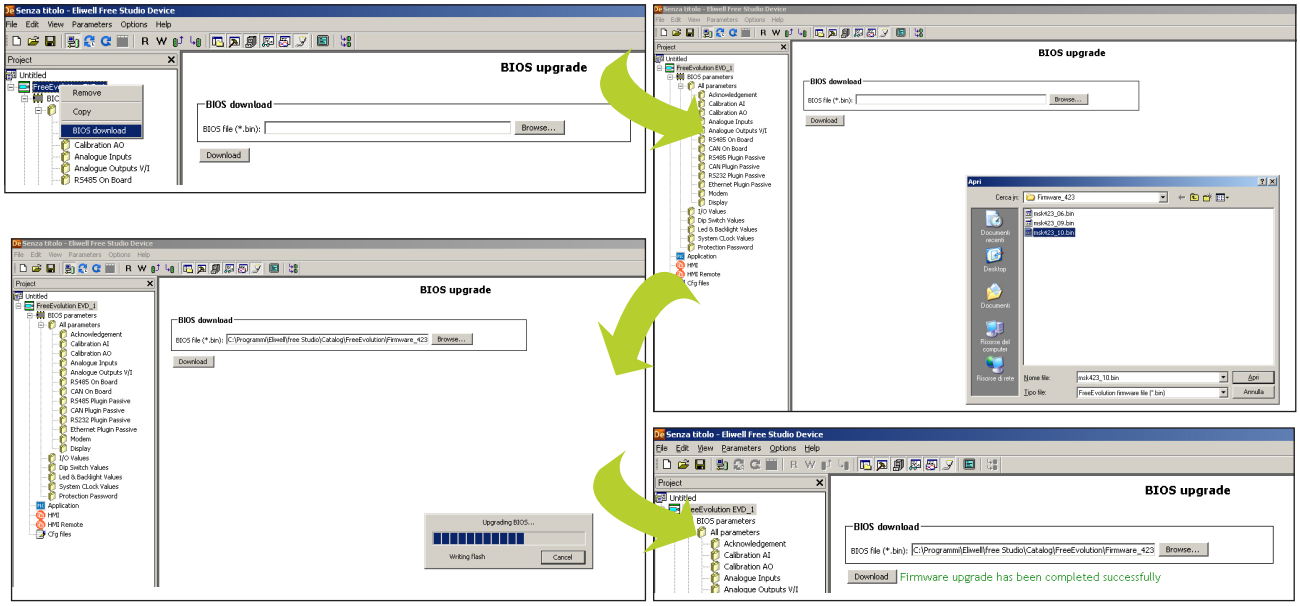
8 **see page 2 - case 2 doesn't allow BIOS download**
9 **/U models only**

Eliwell Controls s.r.l.
Via dell'Industria, 15 • Zona Industriale Paludi • 32010 Pieve d'Alpago (BL) ITALY
Telephone +39 0437 986 111 • Facsimile +39 0437 989 066
Technical helpline +39 0437 986 250 • E-mail eliwell.freeway@invensys.com
www.eliwell.it

Download BIOS FREE Studio→target

CASE 3 uses the **USB/RS485** or **USB/CAN** converters to download **BIOS** directly from the PC

- Connect the **Evolution** to the PC via **USB/RS485** or **USB/CAN** serial lines
- Open **FREE Studio Device**
- Add an **Evolution target** to the project
- Select the name of the **target** and right click on it.
- Select BIOS download and open the **.bin** file you want to download.
- Click on Download



The operation may take a few minutes to conclude. If the download cterminates successfully, a confirmation displays.

CASE 3 allow to upgrade BIOS also for the following:

```
<C:\Programs>\Eliwell\free Studio\Catalog\<firmware>
<firmware> = FreeEvolutionEXP\firmware460 for EVE expansion
<firmware> = FreeEvolutionEVK\firmware489 for FREE Panel
<firmware> = FreeEvolutionEVK\firmware476 for keyboard EVK
```

Appendix - Library

List of fs_iec library functions

`sys_F_CLOSE`

Close a binary file.
The function returns a BOOL which could have the following meanings:

TRUE = Command accepted.
FALSE = Function called into task timed.

`sys_F_EOF`

Test if end of file is reached.
The function returns a BOOL which could have the following meanings:

TRUE = End of file reached.
FALSE = End of file not yet reached or function called into task timed.

`sys_F_FILELENGTH`

File length.
The function returns a DUINT which could have the following meanings:

file length = length of file
-1 = An error occurred or function called into task timed.

`sys_F_REMOVE`

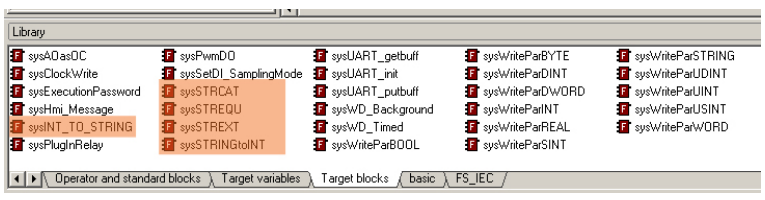
Delete a file.
The function returns a BOOL which could have the following meanings:

TRUE = Command accepted.
FALSE = An error occurred or function called into task timed.

`sys_F_ROPEN`

Open existing file for reading.
The function returns a DWORD which could have the following meanings:

0 = No file found or function called into task timed.
Otherwise = File's ID.



`sys_F_WOPEN`

Truncate file to zero length or create file for writing.
The function returns a DWORD which could have the following meanings:

0	=	No file found or function called into task timed.
Otherwise	=	File's ID.

`sys_F_WOPENA`

Open for appending (writing to end of file).
The function returns a DWORD which could have the following meanings:

0	=	No file found or function called into task timed.
Otherwise	=	File's ID.

`sys_FA_READ`

Read a FLOAT from a binary file.
The function returns a REAL which is the value of the FLOAT.
If function is called into task timed return 0.

`sys_FA_WRITE`

Write a FLOAT to a binary file.
The function returns a BOOL which could have the following meanings:

TRUE	=	Command accepted.
FALSE	=	An error occurred or function called into task timed

`sys_FM_READ`

Read a STRING from binary file.
The function returns a BOOL which could have the following meanings:

TRUE	=	Command accepted.
FALSE	=	An error occurred or function called into task timed.

`sys_FM_WRITE`

Write a STRING to a binary file.
The function returns a BOOL which could have the following meanings:

TRUE	=	Command accepted.
FALSE	=	An error occurred or function called into task timed.

`sys_USBD_Command`

`sys_USBD_Status`

See USB DEVICE

Eliwell Controls s.r.l.
Via dell'Industria, 15 • Zona Industriale Paludi • 32010 Pieve d'Alpago (BL) ITALY
Telephone +39 0437 986 111 • Facsimile +39 0437 989 066
Technical helpline +39 0437 986 250 • E-mail eliwell.freeway@invensys.com
www.eliwell.it

List of target blocks library functions

This list includes function blocks utilities related to file management (strings)

`sysINT_TO_STRING`

Convert a INT number to a STRING.
The function returns a BOOL which could have the following meanings:
TRUE = Done.
FALSE = Not done!

`sysSTRCAT`

Append two STRINGs.
The function returns a BOOL which could have the following meanings:
TRUE = Done.
FALSE = Not done!

`sysSTREQU`

Test if two STRINGSs are equal.
The function returns a BOOL which could have the following meanings:
TRUE = Equal.
FALSE = Not equal.

`sysSTREXT`

Extract a string anticipate by a separator character or <CR> in
a text line terminated by <CR><LF><NULL> in a given position.
The function returns a USINT which could have the following meanings:

0	=	String extracted correctly.
255	=	String extracted truncated.
254	=	Position not present.
253	=	No <CR><LF><NULL> at the end of scanned string.
252	=	String to be scanned too long.

`sysSTRINGtoINT`

Upon success, the function returns the converted integral number
to an INT value. If no valid conversion could be performed, a zero value
is returned. If the correct value is out of the range of representable values,
an invalid number is returned.

Eliwell Controls s.r.l.
Via dell'Industria, 15 • Zona Industriale Paludi • 32010 Pieve d'Alpago (BL) ITALY
Telephone +39 0437 986 111 • Facsimile +39 0437 989 066
Technical helpline +39 0437 986 250 • E-mail eliwell.freeway@invensys.com
www.eliwell.it

Appendix - Example file management

This is a trivial example on how to use previous library functions

Variables

- ret has been declared as a DWORD variable
- ret_bool, timerreset have been declared as a BOOL variable
- Et is the Elapsed Time accumulator to prevented overflow by stopping at max value

Program

```
Timer.Reset:=timerreset;
Timer(Action:=timevar);

(* every 10 seconds a new record is stored into file *)
If Timer.Et>=10000 then
      timerreset:=true;
      else
      timerreset:=false;
end_if;

if timerreset then

      ret := sys_F_WOPENA('b.txt');      (* try to open in append mode a .txt file *)

      if ret=0 then
            (* file does not exist, create a new one and open it *)
            ret := sys_F_WOPEN('b.txt');
      end_if;
      ret_bool := sys_FM_WRITE(ret,'once');
      ret_bool := sys_FM_WRITE(ret,'$n');     (* new line - see Character strings*)
      ret_bool := sys_FM_WRITE(ret,'and twice');
      ret_bool := sys_FM_WRITE(ret,'$n');
      ret_bool := sys_F_CLOSE(ret);

      file_dimension := sys_F_FILELENGTH('b.txt');
end_if;
```


Eliwell Controls s.r.l.
Via dell'Industria, 15 • Zona Industriale Paludi • 32010 Pieve d'Alpago (BL) ITALY
Telephone +39 0437 986 111 • Facsimile +39 0437 989 066
Technical helpline +39 0437 986 250 • E-mail eliwell.freeway@invensys.com
www.eliwell.it

Appendix - Character strings

Two-character combinations in character strings	
Combination	Meaning when printed
$$	Dollar sign
$'	Single quote
$L or $l	Line feed
$N or $n	New line
$P or $p	Form page (new page)
$R or $r	Carriage return
$T or $	TAB
$"	Double quote

Note 1: the 'new line' character provides an implemntation-independent means of defining the end of line for both phsycal and file I/O; for printing, the effect is that of ending a line of data and resuming printing at the beginning of next line

Note 2: The $' combination is only valid inside single quoted string literals

Note 3: The $" combination is only valid inside double quoted string literals

Eliwell Controls s.r.l.
Via dell'Industria, 15 • Zona Industriale Paludi • 32010 Pieve d'Alpago (BL) ITALY
Telephone +39 0437 986 111 • Facsimile +39 0437 989 066
Technical helpline +39 0437 986 250 • E-mail eliwell.freeway@invensys.com
www.eliwell.it