Universidad de Oviedo

Manual de Usuario del Trabajo Fin de Máster realizado por

Manuel Vázquez Muñiz

para la obtención del título de

Máster en Ingeniería de Automatización e Informática Industrial

Upgrade of the UNICOS Time Stamp Push Protocol (TSPP) broker to include ultra-fast events

Junio 2017

INDEX

FIGURE INDEX

1. Introduction

1.1. Project identification

- Title: Upgrade of the UNICOS Time Stamp Push Protocol (TSPP) broker to include ultra-fast events
- Author: Manuel Vázquez Muñiz
- Advisor: Víctor Manuel González Suárez
- Co-advisor: Jerónimo Ortolá Vidal
- Date: June 2017
- Organization: CERN

1.2. Project overview

The current project objective is to solve the issue of the fast interlocks (or ultra-fast events) by improving the Time Stamp Push Protocol (TSPP) used to communicate the control and supervision layers. This protocol is used in the framework UNICOS, and this framework should also be modified as to support this new feature.

With this new feature, the organization will be able to fulfil the requirements of the internal clients who need this capability as to have a proper use of their equipment.

1.3. Document overview

This documents explains the advantages and the use of the fast interlock feature in a UNICOS application.

2. Fast interlocks introduction

The fast interlocks feature is a mechanism implemented in the UNICOS framework as to respond to an event (fast interlock) in the least amount of time possible using a PLC. It is triggered by one or more digital inputs and its reaction is done in one or more digital outputs in an amount of time shorter than the PLC cycle time.

Fast interlocks should be used exclusively for time critical events that need to be treated, and not for any interlock that could appear in the application.

The solution implemented permits the use of this feature in two different ways in Siemens PLCs, using either hardware or cyclic interrupts.

2.1. Hardware Interrupts

For using this mechanism, the digital inputs that trigger a fast interlock need to be connected to a digital input module of the PLC that supports the hardware interrupt capability. Refer to the documentation of the modules to know if they do or do not support this capability.

The fast interlock logic processing is started when the signal input changes (after the delay specified), as shown in Fig. 1. Once the code is executed, the program continues at the point where it was interrupted.

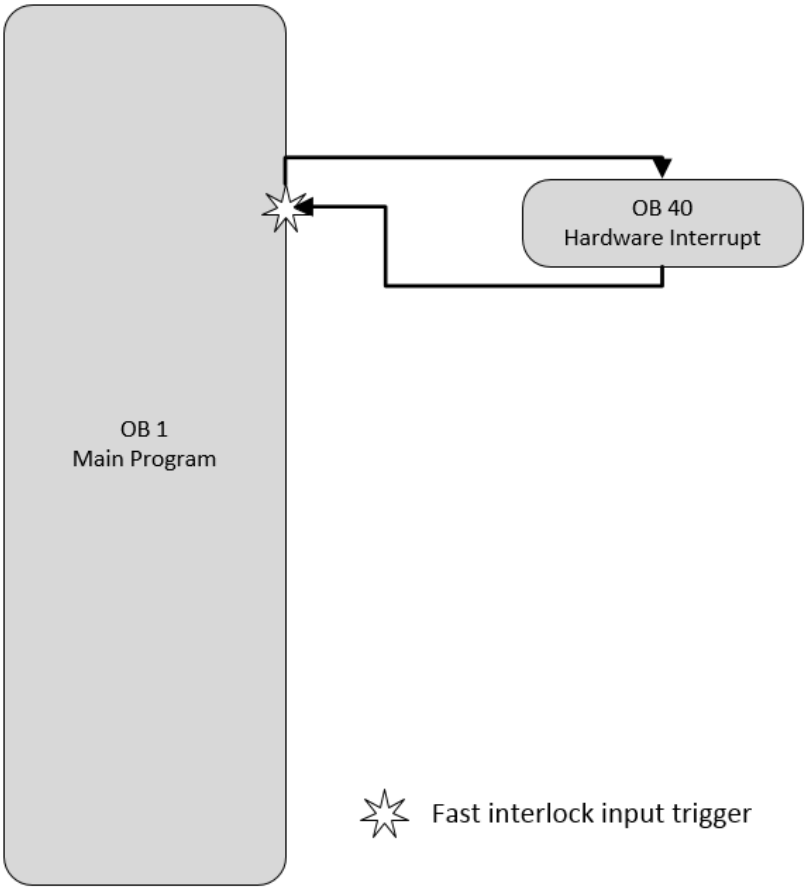

Fig. 1. Hardware Interrupt execution

2.2. Cyclic Interrupts

This mechanism can be used regardless of the hardware, which means no special digital input modules are needed. The cyclic time of the interrupt must be configured by the user.

However, the PLC cycle time is extended in an amount that depends on the normal PLC cycle time, the number of digital inputs used for the fast interlocks and the time configured for this interrupt, which should be short enough to detect the input changes but long enough to not hardly affect the PLC cycle time. This fact should be taken into consideration in case it affects the normal functionality of the process.

For a 1 ms cyclic interrupt time, the extra cycle time for using cyclic interrupts depending on the amount of bytes used for the fast interlock digital inputs and the normal PLC cycle time is the one shown in Fig. 2.

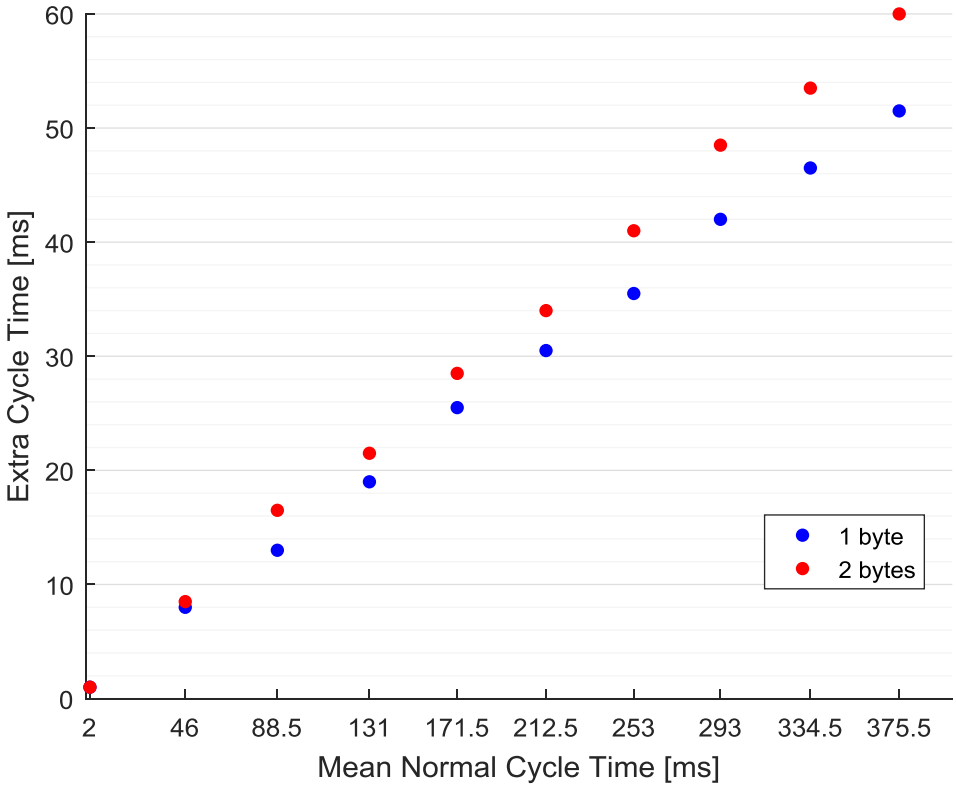

Fig. 2. Extra time for a 1 ms cyclic interrupt

For a 1 byte of fast interlock inputs, the extra cycle time for using cyclic interrupts depending on the cyclic interrupt time established and the normal PLC cycle time is the one shown in Fig. 3.

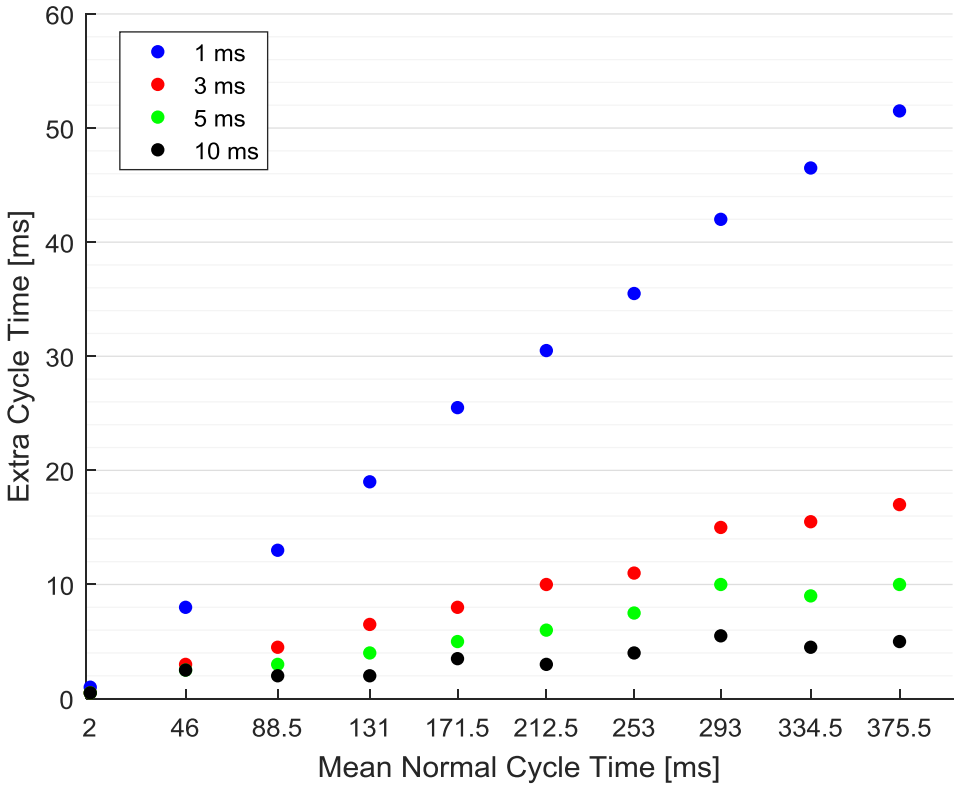

Fig. 3. Extra time for a 1 byte cyclic interrupt input

The interrupt is triggered periodically (with a period established by the user) and the logic is executed when a fast interlock input signal change is detected, as shown in Fig. 4. The program continues at the point it was interrupted each time the interrupt finishes.

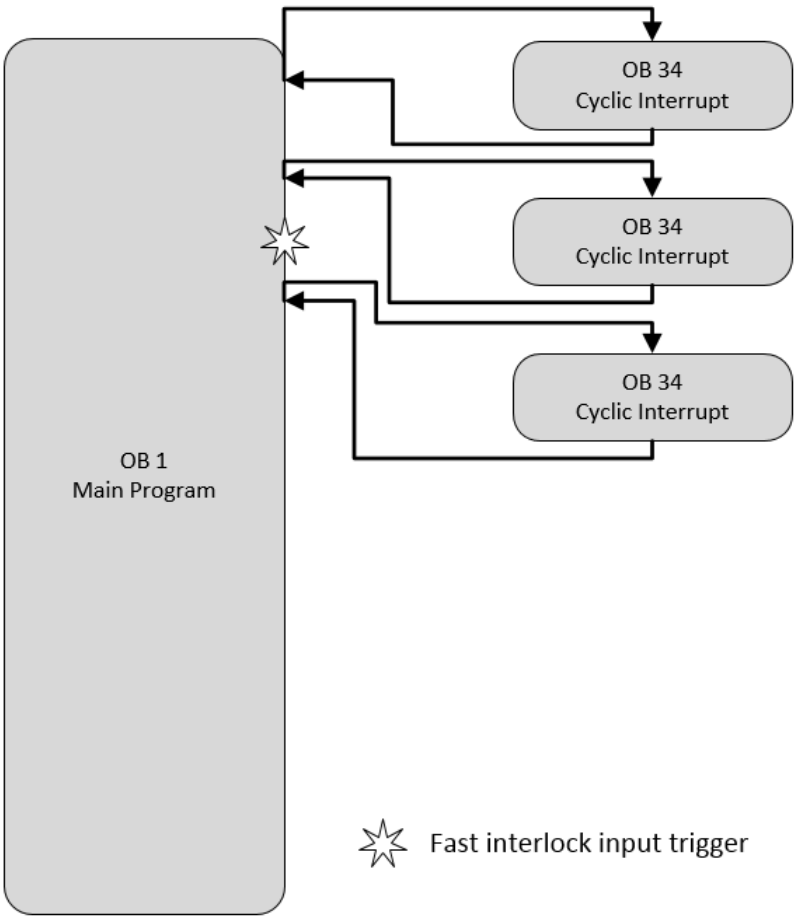

Fig. 4. Cyclic Interrupt execution

3. Creation of a fast interlock UNICOS application

The main steps for creating a UNICOS Step 7 Application [1] should be followed to create a fast interlock application. The only differences remain in the specifications file and the logic implementation.

In order to use the fast interlock feature, the user must specify the objects that they want to use for this purpose, as well as the type of interrupt (hardware or cyclic) that they want to use.

The fast interlock set of objects must be the one shown in Fig. 5.

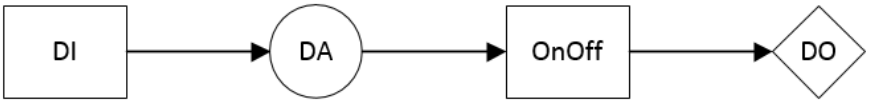

Fig. 5. Fast interlock set of UNICOS objects

- Digital Input (DI): The digital inputs that trigger the fast interlock processing.
- Digital Alarm (DA): The digital alarms to which the digital inputs are connected, and which indicate the fast interlock event appearance.
- OnOff: The OnOff field objects in which the logic for each digital output is defined.
- Digital Output (DO): The digital output objects whose physical output needs to change as fast as possible.

In order to select the objects that are used inside the fast interlock logic, the correspondent column of the specifications file needs to be filled with the necessary value as shown in Fig. 6. The value should be blank for no fast interlock objects, “Hardware Interrupt” for hardware interrupts and “Cyclic Interrupt” for cyclic interrupts. Just one kind of fast interlock interrupt type can be selected for the whole specifications file.

Help	Help	Help	Help	Help	Help	Help	
LogicDeviceDefinitions							FI
CustomLogicParameters							
Parameter4	Parameter5	Parameter6	Parameter7	Parameter8	Parameter9	Parameter10	FI
							No
							No
							No
							HWI
							HWI
							HWI

Fig. 6. Specifications file filled for the digital alarm objects including fast interlock support for 3 of them using hardware interrupts

The fast interlock feature has been designed as to react to some digital input triggers as fast as possible and its use for any other purpose (interaction of the objects with others from the not fast interlock logic) is, in general, forbidden. Still though, the digital alarms included in the fast interlock processing are allowed to be connected to other UNICOS objects in the normal processing (just by establishing it as a master). In some rare cases, that can lead to an inconsistency during one PLC cycle time of the not fast interlock objects connected.

The possible connections that the fast interlock UNICOS objects can have is shown in Fig. 7. The fast interlock objects chain is mandatory, whereas the connection of the fast interlock digital alarms to other not fast interlock objects (PCO and OnOff) is permitted (with the improbable inconvenience already mentioned).

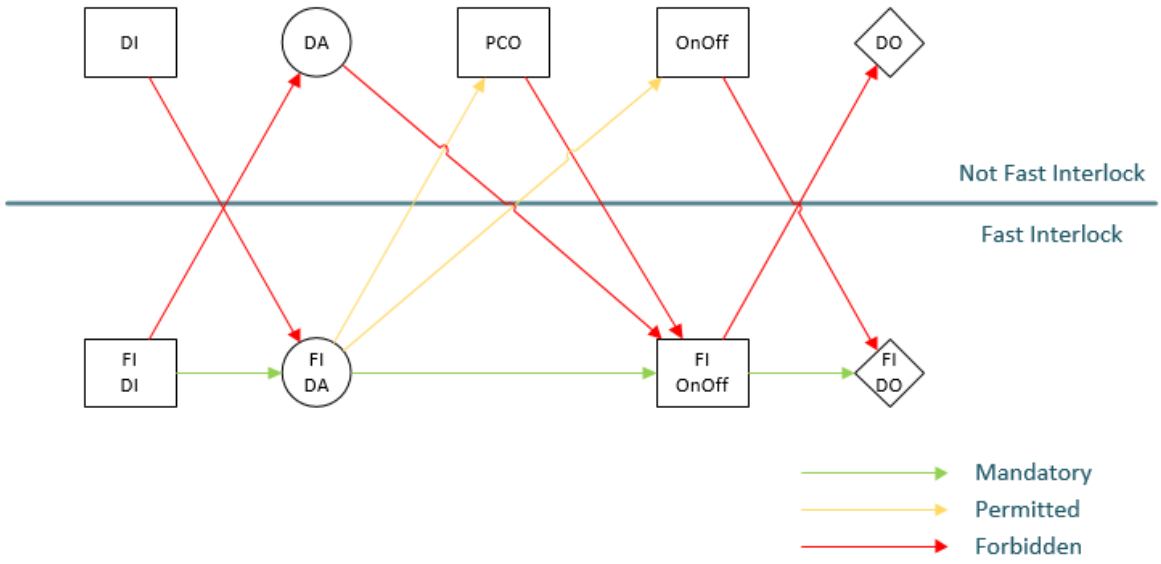

Fig. 7. Interaction between not fast interlock and fast interlock UNICOS objects. Fast interlock objects are placed in the bottom part of the figure

The logic for connecting the digital inputs to the digital outputs of the fast interlock must be contained in the dependant logic of the OnOff objects defined as fast interlock, in the simplest case just by connecting the fast interlock digital alarms to them in the specifications file. If a more complex logic is needed, it can be configured inside the dependant logic of the OnOff objects directly in SCL code or with the use of user templates.

The hardware configuration must be set inside Step 7 as to support the interrupt selected. For hardware interrupts, the digital input module used must be configured with the hardware interrupt capability for either the rising or the falling edge (or both) as shown in Fig. 8. For cyclic interrupts, the CPU must be configured to support the cyclic interrupt of OB 34 with the appropriate cyclic time, with a minimum of 1 ms, as shown in Fig. 9. In any of the cases, you should also configure the input delay in the digital input module configuration.

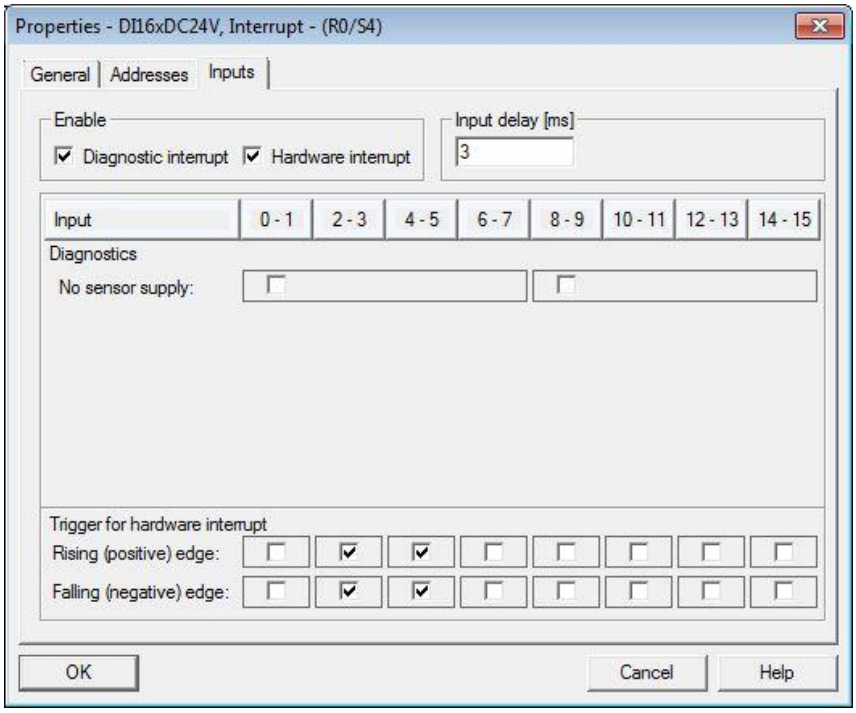

Fig. 8. Digital Input module configuration for hardware interrupts

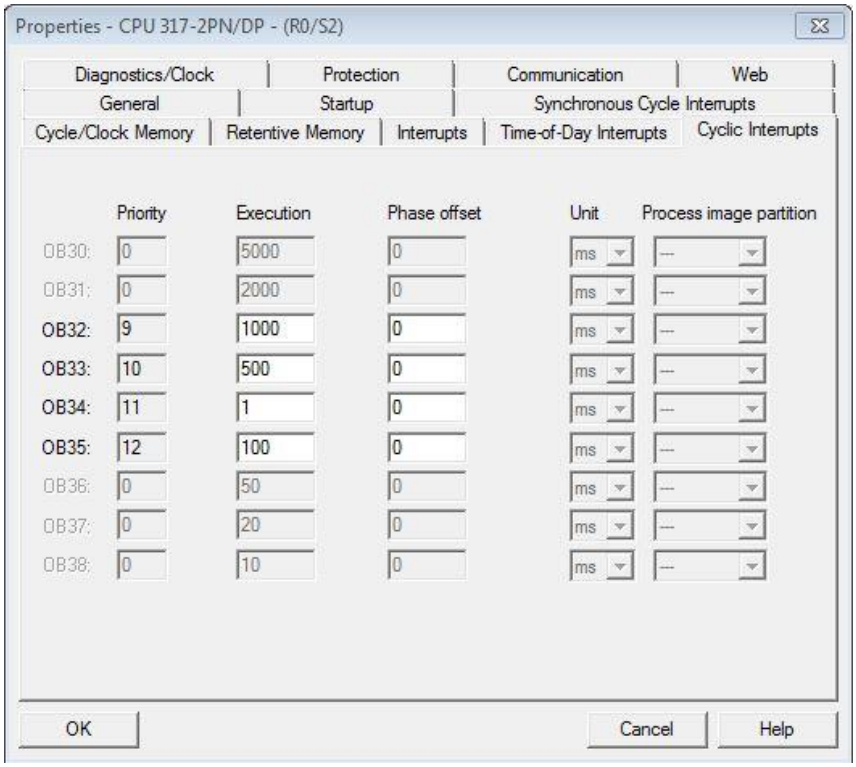

Fig. 9. CPU configuration for cyclic interrupt OB 34

4. Fast interlock rules

Some rules must be followed when filling the specifications file of a fast interlock UNICOS application.

- Only one type of fast interlock can be selected for the whole specifications file. If both hardware and cyclic interrupts are selected for different objects, an error like the one shown in Fig. 10 appears in the UAB error log.

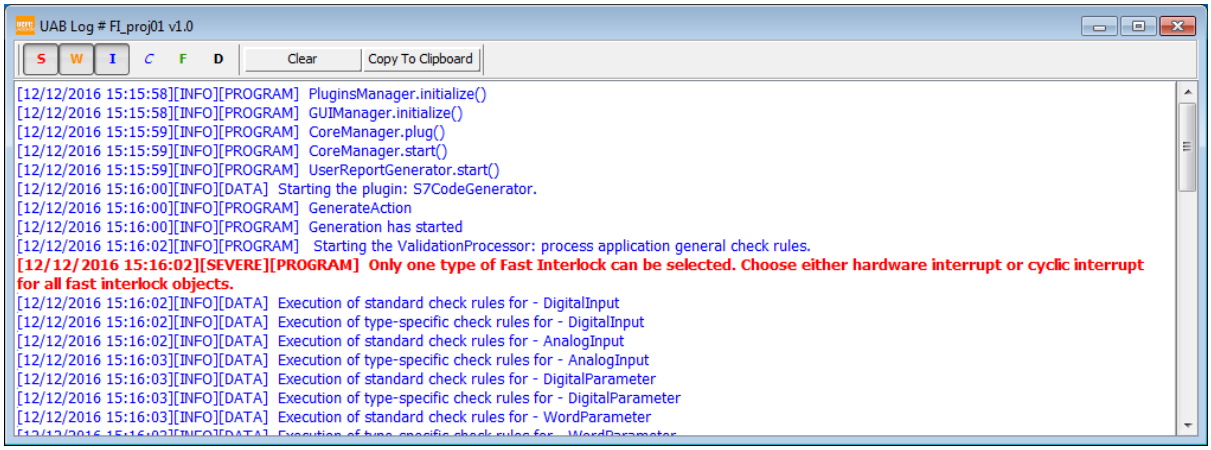

Fig. 10. Different kind of interrupts selected

- Fast interlock objects can't be connected to not fast interlock ones (except for the fast interlock digital alarms). An error will appear in the UAB error log for every fast interlock object that is connected to any not fast interlock object, like shown in Fig. 11.

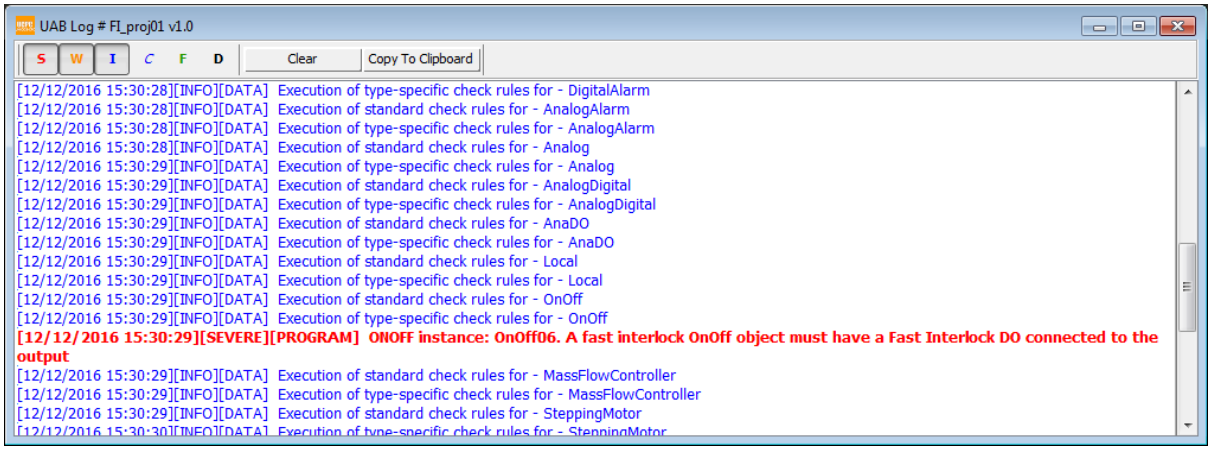

Fig. 11. Normal Digital Output connected to a fast interlock OnOff

- Even for the fast interlock digital alarm, at least one fast interlock OnOff object must be selected as a master of the digital alarm as to fulfil the chain from the digital input to the digital output of fast interlock objects. The error shown in Fig. 12 will appear in the UAB error log otherwise. No error or warning will be shown for the establishment of a not fast interlock OnOff or PCO as a master of a fast interlock digital alarm if at least a fast interlock OnOff object is set as a master of the alarm as well.

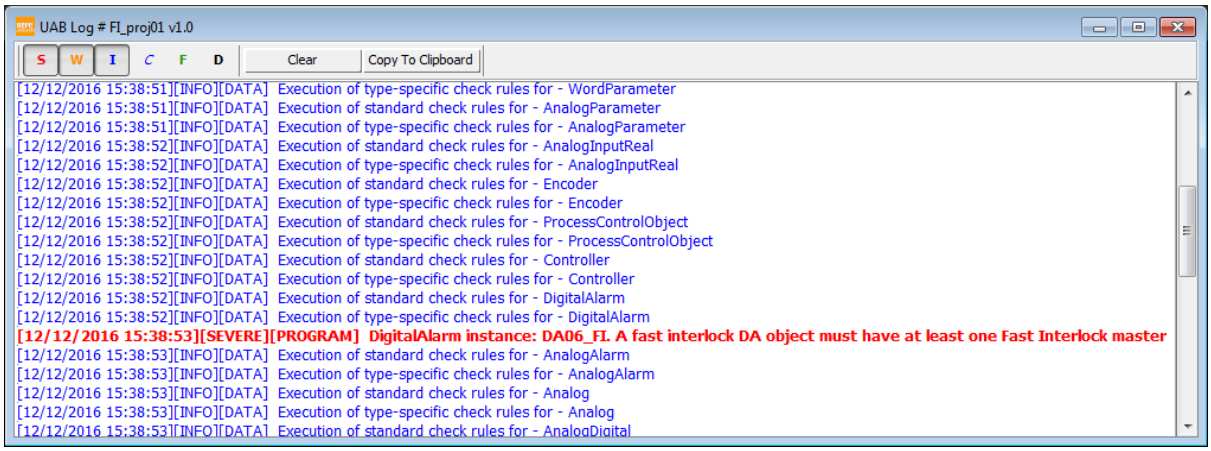

Fig. 12. Fast interlock Digital Alarm with just a normal OnOff master

5. Acronyms

CERN	European Organization for Nuclear Research
UNICOS	Unified Industrial Control Systems
UAB	UNICOS Application Builder
CPC	Continuous Process Control
BE-ICS-PCS	Beams department, Industrial Controls and Safety group, Process Control Systems Section
FI	Fast Interlock
TSPP	Time Stamp Push Protocol
PLC	Programmable Logic Controller
ST	Structured Text
SCL	Structured Control Language
OB	Organization Block
FB	Function Block
FC	Function
SFB	Standard Function Block
SFC	Standard Function
PII	Peripheral Image of Inputs
PIO/PIQ	Peripheral Image of Outputs
IEC	International Electrotechnical Commission
SCADA	Supervisory Control And Data Acquisition
WINCC OA	WinCC Open Architecture
DI	Digital Input
DA	Digital Alarm
DO	Digital Output
PCO	Process Control Object

6. Documents of the project

The current project has been elaborated in multiple documents that describe a certain part of the project.

1. Report: General description of the project. Objectives and conditions for its test. Conclusion from the realization of the project and future works.
2. Planning and budget: Schedule of the different tasks that compound the project and price of the resources used.
3. Step 7 programmer manual: Modifications to the code of the UNICOS applications to support the fast interlock capability. Results obtained from these modifications.
4. Templates programmer manual: Modifications to the code of the templates and of the plugin used to generate the SCL files used in the PLC.
5. User manual: Steps to create a fast interlock UNICOS application.
6. Templates code: Modified template files inside the resources folder of an application and of the UAB plugin.
7. Datasheets: Datasheets of the devices used to research and test the solution for the fast interlocks issue.

Attachments.

1. Attachment 1: Fast interlock application example.

7. Bibliography

[1] S. Izquierdo Rosas and M. Zimny, “Creation of a Siemens S7 UNICOS-CPC 6 Application,” 16 October 2015. [Online]. Available: https://edms.cern.ch/ui/file/1228441/1.8.0/Procedure_S7-UCPC_Application.pdf. [Accessed 15 December 2016].