



EUROPEAN
SPALLATION
SOURCE

Description	Engineering Manual
Document number	ESS-0085848
Date	August 14, 2018
Revision	1
State	Preliminary
Classification	ESS Use Only
Page	1 (8)

ICS Engineering Manual

FOR TIME STAMPING

	Name (Role/Title)
Author	Javier Cereijo Garcia
Reviewer	TBD
Owner	ICS
Approver	ICS

Contents

Contents	2
1 Overview	3
1.1 Scope	3
1.2 Target Audience	3
2 System Description	3
3 System Environment	4
3.1 Hardware	4
3.2 Software	4
4 Engineering Procedure	6
4.1 EVG configuration	6
4.2 EVR configuration	6

Description	Engineering Manual
Document number	ESS-0085848
Date	August 14, 2018
Revision	1
State	Preliminary
Classification	ESS Use Only

1 Overview

At European Spallation Source (ESS), Integrated Control System (ICS) uses the Micro Research Finland (MRF) Timing System¹ as its timing system for the ESS site. It provides 3 main functionalities: event generation, data transmission and timestamping.

1.1 Scope

In this engineering manual it is explained how to configure an IOC to get its timestamps from the timing system. It includes the set up for the EVG, EVR and other EPICS records that need to get their timestamp from the timing system.

Note that this is a very early draft document and should be updated as development progresses.

1.2 Target Audience

This document is targeted to ICS engineers and technical stakeholders of the ESS timing system. It is assumed that the target audience has a technical background in the MRF Timing System, the EPICS development, and a Linux environment.

2 System Description

The timing system consists of an Event Generator (EVG) which converts timing events and signals to an optical signal distributed through fan-out units to an array of Event Receivers (EVRs). The EVRs decode the optical signal and produce hardware and software output signals based on the timing events. The EVG also distributes timestamping events that are used to transfer the precise real time to all EVRs.

The EVG gets the real time and a 1 pps (1 pulse per second) signal from an external clock, although for testing purposes it is also possible to generate the signals internally. The later should only be used for testing, since the clock is generated by software and is of limited accuracy, but it guarantees that all the timing modules (EVG and all connected EVRs) will have the same time and it comes closer to the production setup by allowing use of functions such as timestamping from device support.

The EVRs keep track of the clock by means of the dedicated timestamping events. In the mrfioc2 module EVR EPICS databases there is a record that gets synchronized with the EVG with the frequency of 1 Hz. In between that second, an internal EVR counter tracks progress of time, so that this records always holds the current time of the IOC, and it can be used as a timestamp source.

¹<http://www.mrf.fi/>

Description	Engineering Manual
Document number	ESS-0085848
Date	August 14, 2018
Revision	1
State	Preliminary
Classification	ESS Use Only

3 System Environment

Before describing the time stamping mechanism, it is mandatory to have proper system environment that consists of specific hardware and software lists. Here we will show the hardware and software lists, their block diagrams, and their setup in the ICS lab at ESS. The information shown in this chapter is used in the ICS Lab at ESS.

3.1 Hardware

Table 1 shows the hardware list with the hardware used in the examples in this manual. It is possible to use different form factors than what is shown in the examples; for more information check the specific engineering manual.

Hardware	Info	Serial Number
MRF cPCI-EVG-230	ICS TAG-805	E283016
cPCI crate	ICS TAG-362	
CPU	ICS TAG-363, hostname: icslab-ts01	43040574
MRF PCIe-EVR-300DC	ICS TAG-473	M263057
MRF IFB-300	ICS TAG-352	K472044
MRF Universal I/O module		
Adlink IPC	ICS TAG-837, hostname: icslab-ipc01	93-41036-G10E
Optical cables	LC, Optical 850 nm	

Table 1 Hardware list.

3.2 Software

Table 2 shows the software list and its environment.

Description	Engineering Manual
Document number	ESS-0085848
Date	August 14, 2018
Revision	1
State	Preliminary
Classification	ESS Use Only

Item	Version Info.
CentOS Linux	7.4.1708
Kernel	3.10.0-693.17.1.el7.x86_64
mrf kernel module	version : 1 / srcversion A998B22F1425D7388F5F7A7
E3	2.5.4
EPICS Base	3.15.5
mrfioc2	E3 module ver. 2.2.0
devLib2	E3 module ver. 2.9.0

Table 2 Software list and its version information.

Description	Engineering Manual
Document number	ESS-0085848
Date	August 14, 2018
Revision	1
State	Preliminary
Classification	ESS Use Only

4 Engineering Procedure

This chapter provides the minimal information to configure an IOC to get its timestamps from the timing system. It is not intended to show a complete deployment of the timing system, but the basic parts that need to be included in the desired IOC start-up script to get timestamping.

4.1 EVG configuration

Listing 4.1 shows the basic start-up script for the EVG, with short comments for the important parts:

```
require mrfioc2, 2.2.0

epicsEnvSet("EPICS_CA_MAX_ARRAY_BYTES", "10000000")

epicsEnvSet("IOC", "TIMESTAMP")
epicsEnvSet("DEV1", "EVG0")

epicsEnvSet("MainEvtCODE" "14")
epicsEnvSet("HeartBeatEvtCODE" "122")
epicsEnvSet("ESSEvtClockRate" "88.0525")

mrmEvgSetupPCI($(DEV1), "16:0e.0")
dbLoadRecords("cpci-evg230-ess.db", "SYS=$(IOC), D=$(DEV1), EVG=$(DEV1), FEVT=$(ESSEvtClockRate),
    FRF=$(ESSEvtClockRate), FDIV=1")

iocInit()

# Configure a software thread to emulate 1PPS input
dbpf $(IOC)-$(DEV1):1ppsInp-Sel "Sys Clk"

# Configure the master event rate at 14 Hz
dbpf $(IOC)-$(DEV1):Mxc0-Prescaler-SP 6289464 # Divide event freq to get 14 Hz
dbpf $(IOC)-$(DEV1):TrigEvt0-EvtCode-SP $(MainEvtCODE)
dbpf $(IOC)-$(DEV1):TrigEvt0-TrigSrc-Sel "Mxc0"

# Wait 5 seconds and synchronize all timestamps
epicsThreadSleep 5
dbpf $(IOC)-$(DEV1):SyncTimestamp-Cmd 1
```

Listing 4.1 Start-up script `startup_EVG.cmd`.

4.2 EVR configuration

For the timestamp to be available at the system with the EVR, the only requirement is to have the EVR initialized and receiving the transmission from the EVG. Listing 4.2 shows the basic start-up script for the EVR:

```
1 require mrfioc2,2.2.0
2
3 epicsEnvSet("EPICS_CA_MAX_ARRAY_BYTES", "10000000")
4
```

Description	Engineering Manual
Document number	ESS-0085848
Date	August 14, 2018
Revision	1
State	Preliminary
Classification	ESS Use Only

```

5 epicsEnvSet("IOC", "TIMESTAMP")
6 epicsEnvSet("DEV1", "EVR0")
7
8 epicsEnvSet("MainEvtCODE" "14")
9 epicsEnvSet("HeartBeatEvtCODE" "122")
10 epicsEnvSet("ESSEvtClockRate" "88.0525")
11
12 mrmEvrSetupPCI("${DEV1}", "01:00.0")
13 dbLoadRecords("evr-pcie-300dc-ess.db", "EVR=${DEV1}, SYS=${IOC}, D=${DEV1}, FEVT=${ESSEvtClockRate}
14 ")
15 # Needed with software timestamp source w/o RT thread scheduling
16 var evrMrmTimeNSOverflowThreshold 100000
17
18
19 iocInit()
20
21 # Set delay compensation
22 dbpf ${IOC}-${DEV1}:DC-Tgt-SP 70

```

Listing 4.2 Start-up script `startup_EVR.cmd`.

After being initialized the EVR will act as a time provider, as can be seen by checking `generalTimeReport` in the IOC shell, as shown in Listing 4.3:

```

6690f8a.icslab-ipc01.14335 > generalTimeReport
Backwards time errors prevented 1464 times.

Current Time Providers:
  "EVR", priority = 50
    Current Time is 2018-08-13 11:18:40.234652.
  "OS Clock", priority = 999
    Current Time is 2018-08-13 11:20:05.683007.

Event Time Providers:
  "EVR", priority = 50

```

Listing 4.3 `generalTimeReport` on the IOC shell with an EVR.

This shows the 2 time providers for this system: the EVR and the last resort provider (OS clock). There is a fallback from higher priority providers (smaller value of priority, in this case the EVR) to lower priority providers (larger value of priority, usually the OS clock) if the higher priority ones fail.

In the case that the OS clock time is ahead of the EVR time, as shown in Listing 4.3, there will be some increasing **Backwards time errors prevention** notifications in `generalTimeReport`, and the timestamps might show a strange behaviour at the startup of the IOC, while the EVR time catches up with the OS clock time. For this reason it is recommended to have all systems set with OS clock times close to each other.

4.2.1 Timestamping of records simultaneous with the timing system events

Some records should have the same timestamp as the event that trigger them, for example to correlate their value to a specific beam pulse. In this case the `${IOC}-${DEV}:Time-I`

Description	Engineering Manual
Document number	ESS-0085848
Date	August 14, 2018
Revision	1
State	Preliminary
Classification	ESS Use Only

record is set to process on arrival of the desired event and to have exactly the same timestamp as that event. The value of the $\$(IOC)-\$(DEV):Time-I$ record is the current time expressed as a human-readable string DATE HH:MM:SS.

The easiest way of doing this is to modify line 13 in Listing 4.2 to add the macro EVNT1HZ equal to the event number that the $\$(IOC)-\$(DEV):Time-I$ record should take its timestamp from, so that the line looks like:

```
dbLoadRecords("evr-pcie-300dc-ess.db", "EVR=$(DEV1), SYS=$(IOC), D=$(DEV1), FEVT=$(ESSEvtClockRate)
, EVNT1HZ=$(MainEvtCODE)")
```

where the EVNT1HZ macro has the value of the MainEvtCODE variable, 14.

Alternatively this configuration can be made after the startup of the IOC with:

```
iocuser@icslab-ipc01: ~$ caput TIMESTAMP-EVR0:Time-I.EVNT 14
Old : TIMESTAMP-EVR0:Time-I.EVNT      125
New : TIMESTAMP-EVR0:Time-I.EVNT      14
iocuser@icslab-ipc01: ~$ caput TIMESTAMP-EVR0:Time-I.INP "@OBJ=EVR0, Code=14"
Old : TIMESTAMP-EVR0:Time-I.INP      @OBJ=EVR0, Code=125
New : TIMESTAMP-EVR0:Time-I.INP      @OBJ=EVR0, Code=14
```

After this set the .TSEL field of the record that should have the desired timestamp to point to the .TIME field of the $\$(IOC)-\$(DEV):Time-I$ record of the EVR by adding this line after iocInit() in the startup script:

```
dbpf exemplerecord.TSEL TIMESTAMP-EVR0:Time-I.TIME
```