Référence : EXP-DET17-R002-E

**I**

XIA-DXP-XMAP - Mode D’Emploi

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Date de diffusion** | **Rédacteur** | **Vérificateur** | **Approbateur** | **Visa de l’approbateur** | **Modifications** | **Indice** |
| 5/01/17 | KD |  |  |  |  | 0 |
| 10/01/17 | KD | AN |  |  | Ajout des tableaux « Properties, … » et relecture. | 1 |
| 13/04/17 | KD |  |  |  | Modifications | 2 |
| 30/01/2019 | AN |  |  |  | Mise à jour Annexe 6 | 3 |
| 03/02/2020 | AN | Paco |  |  | Mise à jour FalconX | 4 |

Diffusion : Pour information

Destinataires :

Groupes Détecteurs, NOUREDDINE Arafat, BUTEAU Alain, ABIVEN Yves-Marie, RENAUD Guillaume

Table des matieres

[1. Introduction 3](#_Toc34121669)

[1.1. notes 3](#_Toc34121670)

[2. Mise en place 4](#_Toc34121671)

[2.1. Pre-requis 4](#_Toc34121672)

[2.2. Device Initialisation 4](#_Toc34121673)

[2.3. Device Démarrage 6](#_Toc34121674)

[3. Mode MCA 8](#_Toc34121675)

[3.1. mode MCA - Chargement 9](#_Toc34121676)

[3.2. mode MCA – Acquisition 9](#_Toc34121677)

[3.2.1. Déclenchement 9](#_Toc34121678)

[3.2.2. Comptage 9](#_Toc34121679)

[3.3. Mode MCA - Fichier INI 10](#_Toc34121680)

[3.4. Mode MCA - ROIs 10](#_Toc34121681)

[3.5. Device : mode MCA – Enregistrement 10](#_Toc34121682)

[4. Mode MAPPING FULL 11](#_Toc34121683)

[4.1. mode MAPPING FULL – Chargement 12](#_Toc34121684)

[4.2. mode MAPPING FULL – Acquisition 12](#_Toc34121685)

[4.2.1. Déclenchement 12](#_Toc34121686)

[4.2.2. Comptage 13](#_Toc34121687)

[4.3. MODE MAPPING FULL - Fichier INI 13](#_Toc34121688)

[4.4. mode MAPPING - ROIs 13](#_Toc34121689)

[4.5. Device : mode MAPPING FULL – Enregistrement 13](#_Toc34121690)

[5. Mode MAPPING SCA 14](#_Toc34121691)

[6. Quick Start Guide 14](#_Toc34121692)

[7. Annexe – Devices Propriétés/Attributs/Commandes 15](#_Toc34121693)

[7.1.1. Properties 15](#_Toc34121694)

[7.1.2. Attributes 16](#_Toc34121695)

[7.1.3. Commands 17](#_Toc34121696)

[8. Annexe – Fichier type MCA 18](#_Toc34121697)

[9. Annexe – Fichier type MAPPING FULL 19](#_Toc34121698)

[10. Annexe – Fichier type MAPPING SCA 20](#_Toc34121699)

[11. Annexe – Fichier type FalconX 21](#_Toc34121700)

# Introduction

## notes

**en violet** : les propriétés

**en vert** : les attributs

**en orange** : les commandes

**en rouge** : les paramètres du fichier de configuration

**Le device Tango XiaDxp peut gérer deux types de cartes XIA : DXP-XMAP et FalconX**

**3 Modes d’acquisition existent :**

**Mode MCA** : mode de fonctionnement de la carte dans lequel on ne peut acquérir qu'un seul spectre/voie et par acquisition (+ des statistiques et les intégrales sous les ROIS définies)

**Mode MAPPING** FULL : mode de fonctionnement de la carte pour les applications dans lesquelles un paramètre de la ligne est scanné. On peut acquérir autant de spectres (+ statistiques) qu'il y a de points dans le scan.

**Mode MAPPING** SCA : Dans ce mode, on acquiert uniquement les « counts » dans chaque Roi prédéfini et on les mémorise dans des spectres.

# Mise en place

## Pre-requis

**Carte/Module XIA dxp-XMAP[[1]](#footnote-1)**

<https://www.xia.com/dxp_products.html>

**Carte DXP-XMAP :**

**XIA Xmanager @ jour**

Le logiciel et driver Xmanager (**version 1.2**) doivent être installés et opérationnels sur la CPU du châssis portant la carte XIA-DXP-XMAP.

**Module FalconXn**

**XIA Prospect @ jour**

Le logiciel et driver Prospect (version 1.1.27) doivent être installés et opérationnels sur la CPU du châssis relié au module XIA-FalconX.

**Fichier de configuration**

Un fichier de configuration (\*.ini) correspondant à l’installation (pci\_bus, pci\_slot, Adress IP) et utilisation de la carte XIA-DXP-XMAP ou XIA-FalconX.

Voir documentation :

* XIA LLC, “Digital X-ray Processor User’s Manual, Model DXP-XMAP with xManager Software”, Version 1.0.x, 19/11/2018, http :www.xia.com
* XIA LLC, “FalconX1/X4/X8 Quick Start Guide”, Version 0.5, 07/08/2016.
* F.J. Iguaz, « *Procédure de réglage de l’électronique XIA-DXP-XMAP pour un détecteur de fluorescence* », document interne EXP-DET19-R091-A.
* F.J. Iguaz, « *Procédure de réglage de l’électronique XIA-FalconX pour un détecteur de fluorescence*», document interne EXP-DET19-R205-A.

Voir aussi les Annexes.

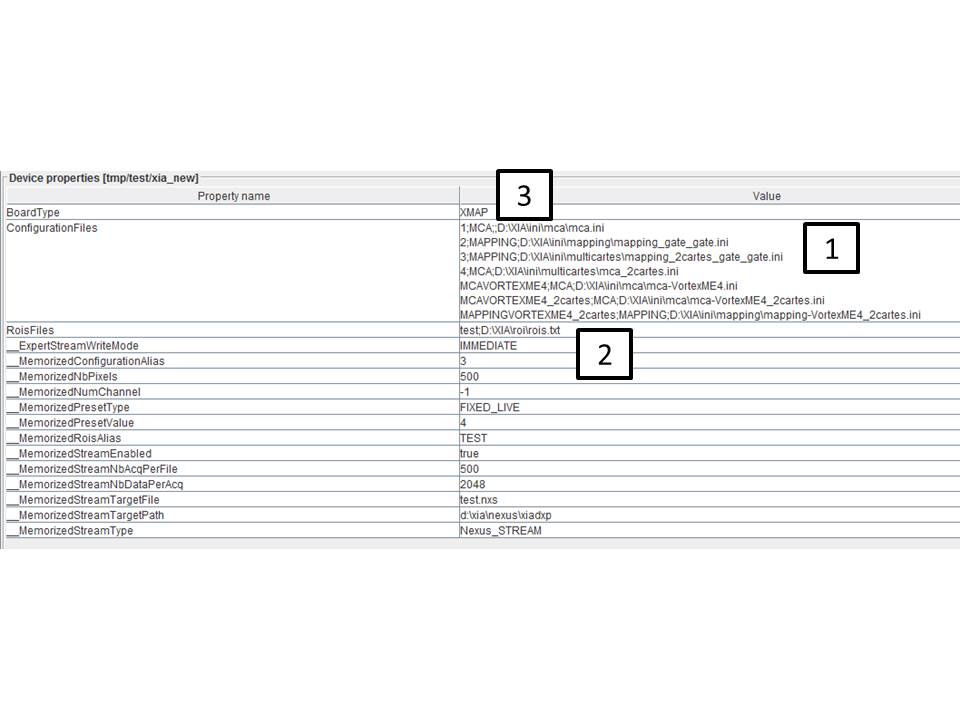
**Device Tango**

Le device ds\_XiaDxp doit être démarré et dans l’état « Jaune ».

## Device Initialisation

Il faut ouvrir l’ATK panel correspondant au device dxp-XMAP installé.

Les propriétés suivantes doivent être complétées :



|  |  |
| --- | --- |
| 1 | **ConfigurationFiles (Obligatoire)**  Une ou plusieurs configurations (fichier \*.ini) doivent être renseignées via la propriété **ConfigurationFiles.**  La Value doit être de type : Alias ;mode (MCA ou MAPPING) ;@ du fichier .ini  Une configuration par ligne.  Ex : Config1;MCA;\\deviceservers\configfiles\XIADXP\MCA\DXP-XFlash5030.ini  Ex : Flyscan;MAPPING;\\deviceservers\configfiles\XIAFALCON\MAPPING\20200205-flyscan.ini |
| 2 | **RoisFiles** *(non obligatoire)*  La propriété **RoisFiles** permet de définir la localisation d’un (ou plusieurs) fichier de ROI (fichier \*.txt)  ex : \\deviceservers\configfiles\XIADXP\ROI\roi.txt  A l’intérieur du fichier txt, les ROIs sont définies par :  n°devoie ;debutROI1 ;fin ROI1,…  ex : 0 ;100 ;200 ;500 ;800 |
| 3 | **BoardType =** XMAP;//DeviceServers/win32/PluginXIA/xmap/1\_1\_11/  ou  **BoardType =** FALCONX;//DeviceServers/win32/PluginXIA/falconx/1\_1\_20/ |

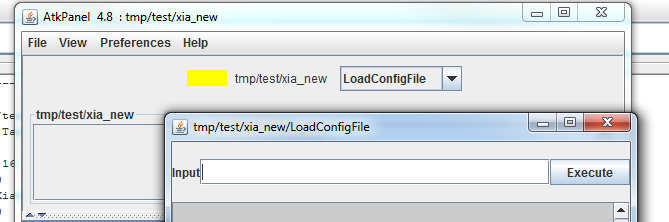
***Remarques importantes:***

1. Les fichiers de configuration **ConfigurationFile** se trouvent physiquement dans le répertoire « \usr\Local\configfiles\... ». Il faut y les copier soit par le répertoire partagé de la carte CPI (si ça existe) ou par la Ruche ; et vérifier après qu’on a les le droit de lecture et écriture. En cas contraire, il faut les donner en utilisant le commande « chmod 777 xxxx.ini ».
2. Si un nouveau fichier de configuration est renseigné, le device doit être réinitialisé via la commande **Init.**
3. Pour la carte XIA-FalconX, **seulement les voies calibrées** doivent être inclues dans le fichier de configuration. En cas contraire, il faut les enlever à la main. Voir Annexes.

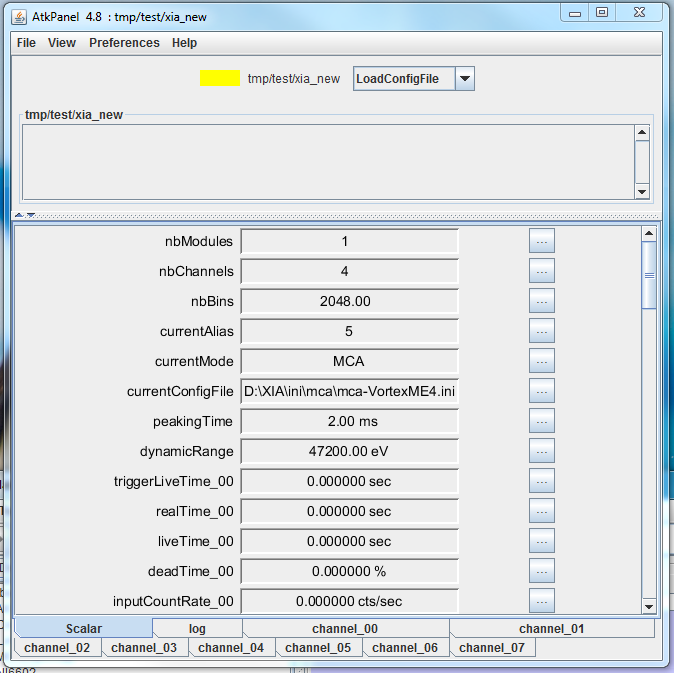
Device initialisé, l’état du device doit être « blanc »

## Device Démarrage

Device initialisé, il suffit de charger la config. via la commande « **LoadConfigFile** » et d’entrer l’alias (voir §2.2, **ConfigurationFiles**) souhaité puis **Execute**.



Le device passe dans l’état « rose » puis « jaune »



2

5

4

3

1

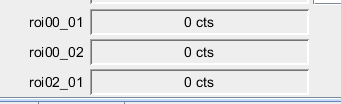
Sont alors affichés dans l’ATK :

|  |  |
| --- | --- |
| 1 | **nbModules & nbChannels**  le nombre de module (carte) et le nombre de voie(s) configurée(s) |
| 2 | **nbBins**  le nombre de canaux de l’analyse |
| 3 | **currentAlias**, **currentMode** & **currentConfigFile**  Le nom, le mode de l’acquisition (MCA, MAPPING\_FULL ou MAPPING\_SCA) et un rappel du lien vers le fichier de configuration |
| 4 | **peakingTime** & **DynamiqueRange**  Rappel du peaking Time configuré et information sur la gamme en énergie de l’analyse |
| 5 | **realTime\_n°voie,live\_Time\_n°voie,deadTime\_n°voie, Input et ouputcountRate, …**  Tous les attributs liés à la mesure |

Si des ROIs ont été chargées, le nombre de coups intégrés dans les ROIs correspondantes sont affichés dans l’attribut **roiXX\_YY** tel que :

XX = le channel (n° de la voie)

YY = n° de la ROI sur ce channel



***Remarque importante*** : Il est nécessaire de fermer et re-ouvrir l’ATK Panel pour avoir l’affichage complet des différents attributs (Bug d’affichage des attributs dynamique).

# Mode MCA

*Mode simple permettant l’acquisition de spectres par méthode dite « pas à pas »*

Le schéma fonctionnel du mode MCA est donné ainsi :

**LoadConfigFile**

« DEFAUT »

« STANDBY »

**Snap**

Mode GATE

Non

« RUNNING »

**PresetType**

NONE

STOP

FIXED LIVE/REAL/EVENT

Preset < PresetValue

« RUNNING »

Oui

En mode GATE :

Comptage pendant la largeur de la GATE

NONE : jusqu’au STOP

AUTRE : jusqu’au presetValue définie

Ex : mode PresetType = FIXED\_REAL

Si realTime > PresetValue (ex 1s) alors fin d ’acquisition

gate\_master =1

gate\_mode = 0

gate\_ignore = 0

gate\_master =0

gate\_mode = 1

gate\_ignore = 1

Faux

## mode MCA - Chargement

Suite au chargement via la commande **LoadConfigFile**, le device passe en « STANDBY » et doit afficher **currentMode = MCA** (voir aussi §2.3)

## mode MCA – Acquisition

### Déclenchement

L’acquisition peut être déclenchée par :

- « soft » (déclenchement directement suite à la commande SNAP)

- « hard » via l’entrée logique GATE (TTL/CMOS Logic Input), l’acquisition est démarrée suite à la commande SNAP mais le comptage est déclenché par un trigger.

*Paramètre non visible dans l’ATK Panel[[2]](#footnote-2)*

### Comptage[[3]](#footnote-3)

Plusieurs modes d’acquisition sont disponibles et paramétrable dans l’ATK via l’attribut **presetType**

Acquisition manuelle

NONE : acquisition s'arrête après un **"stop"** utilisateur

Acquisition avec fin automatique

FIXED\_REAL : acquisition s'arrête après écoulement du "real time[[4]](#footnote-4)" spécifié dans presetValue  
FIXED\_LIVE : acquisition s'arrête après écoulement du "live time[[5]](#footnote-5)" spécifié dans presetValue  
FIXED\_EVENTS : acquisition s'arrête après comptage des "nb events" spécifié dans presetValue

FIXED\_TRIGGERS : acquisition s'arrête après détection de "input triggers" spécifié dans presetValue

On note quand mode GATE (déclenchement « hard » du comptage), le temps de comptage est donné par la largeur de la GATE.

## Mode MCA - Fichier INI

Fichier \*.ini créé via Xmanager, il faut :

- définir le mode MCA via :

mapping\_mode = 0

- configurer le mode de déclenchement de l’acquisition via :

|  |  |
| --- | --- |
| **Non GATE**  gate\_master = 0  gate\_mode = 1  gate\_ignore = 0 ou 1 | **GATE**  gate\_master = 1  gate\_mode = 0  gate\_ignore = 0 |

- configurer le mode d’acquisition via preset\_type tel que :

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| NONE | FIXED\_REAL | FIXED\_LIVE | FIXED\_EVENTS | FIXED\_TRIGGERS |
| preset\_type = 0 | preset\_type = 1 | preset\_type = 2 | preset\_type = 3 | preset\_type = 4 |

*Voir aussi annexe « Fichiers type MCA »*

## Mode MCA - ROIs

En mode MCA les ROIs peuvent être définies de différentes manières :

* Directement dans le fichier de configuration (avant chargement dans le device)
* Par chargement d’un fichier via la commande « setRoisFromFile » en entrant l’alias défini dans la propriété **RoisFiles** du device (voir §2.3)
* Par l’utilisateur via la commande[[6]](#footnote-6) « setRoisFromList » en enetrant :

N°devoie ;debut\_roi1 ;fin\_roi1 ;debut\_roi2 ;fin\_roi2 ;…

N°devoie ;debut\_roi1 ;fin\_roi1 ;debut\_roi2 ;…

* Les ROIs peuvent être relues via getRois
* Les ROIs peuvent être supprimées via la commande removeRois en entrant le n°delavoie

## Device : mode MCA – Enregistrement

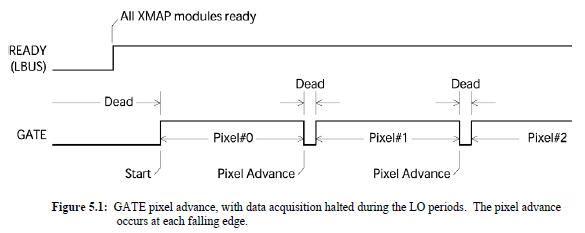
Pas d’enregistrement de nexus en mode MCA

# Mode MAPPING FULL

*Acquisition synchrone sur les différentes voies de la XMAP par déclenchement externe.*

*On parle de pixel comme étant chaque spectre (+ statistique type OCR,ICR, …) enregistré par acquisition.*

Le mode par défaut utilisé par le device ds\_dxpXMAP est : MAPPING GATE (avec halt)



Le 10.01.17,

Les odes MAPPING GATE Ignore et MAPPING SYNC ne sont pas implémentés dans le device.

Le schéma fonctionnel du mode MAPPING est donné ainsi :

**LoadConfigFile**

« DEFAUT »

« STANDBY »

**Snap**

**currentPixel >nbPixel**

« RUNNING »

Acquisition en parallèle des n spectres (pixels) + statistiques dans un fichier nexus

L’acquisition peut être stoppée par la commande STOP

mapping\_mode=1

gate\_master =1

gate\_mode = 0

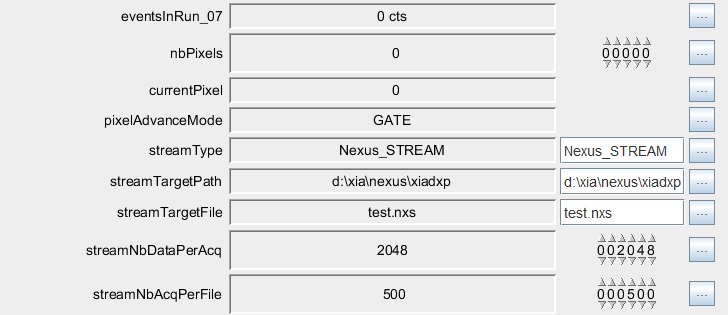
gate\_ignore = 0

## mode MAPPING FULL – Chargement

Device initialisé (§2.2) et suite au chargement d’un fichier \*.ini de type MAPPING via la commande **LoadConfigFile** (et l’alias correspondant), le device passe en « STANDBY » et doit afficher **currentMode = MAPPING\_FULL** (voir aussi §2.3)

En supplément des attributs du §2.3, sont aussi affichés[[7]](#footnote-7):

|  |  |
| --- | --- |
| 1 | **nbPixel & currentPixel**  Le nombre de spectres que l’on souhaite acquérir et l’état réel du nombre de spectres acquis |
| 2 | **pixeladvenceMode**  Méthode utilisée pour passer de l’acquisition d’un spectre au suivant. |
|  | + En mode expert view |
| 3 | **streamTargetPath** & **streamTargetFile**  Le chemin d’enregistrement des fichiers nexus et le nom du fichier |
| 4 | **streamNbDataPerAcq**  Le nombre de canaux stockés par spectre |
| 5 | **streamNbDataPerFile**  Le nombre de spectres (+ statistiques) par fichier Nexus |



1

5

4

3

2

## mode MAPPING FULL – Acquisition

### Déclenchement

L’acquisition est déclenchée par un trig « hard » via l’entrée logique GATE (TTL/CMOS Logic Input)

### Comptage[[8]](#footnote-8)

Seul le mode MAPPING GATE (avec halt) est implémenté dans le device.

Le temps de comptage est défini par la largeur de la GATE.

## MODE MAPPING FULL - Fichier INI

Fichier \*.ini créé via Xmanager, il faut :

- définir le mode MAPPING via :

mapping\_mode = 1

pixel\_advance\_mode = 1

- configurer le mode de déclenchement de l’acquisition via :

gate\_master = 1

gate\_ignore = 0

- prédéfinir les paramètres d’acquisition via :

num\_map\_pixels = 1

num\_map\_pixels\_per\_buffer = 10

*Voir aussi annexe « Fichiers type MAPPING »*

## mode MAPPING - ROIs

Pas de ROIs en mode MAPPING

## Device : mode MAPPING FULL – Enregistrement

L’attribut streamType doit être Nexus\_STREAM (Voir §2.2)

Les paramètres de stockage sont définis dans le device via les attributs

**streamNbAcqPerFile**

**streamTargetFile**

**streamTargetPath**

*L’utilisateur doit s’assurer à ne pas dépasser la capacité mémoire disponible en limitant le nombre de pixels par fichier.*

# Mode MAPPING SCA

Dans le fichier de configuration ini, le paramètre mapping\_mode doit être à 2.0

Dans ce mode, on n’acquiert / n’affiche pas tout le spectre des données, mais uniquement les « counts » dans chaque Roi prédéfini.

Au niveau du device, la propriété **ConfigurationFiles** qui doit être formatée comme suit : ALIAS;MODE;FILE\_PATH\_NAME accepte un nouveau mode qui est **MAPPING\_SCA**

L’utilisateur peut définir des Rois (64 au max) pour chaque « Channel »

Cette définition se fait exactement comme dans le mode MCA via les commandes **SetRoIFromFile** et **SetRoIFromList**

Pour chaque Roi défini, un attribut Spectrum sera créé et qui portera le nom : hroix\_y  avec

x = numéro de voie

y = numéro de roi pour la voie concernée

Pour chaque point d’acquisition « trigger », un nouvelle donnée sera ajoutée au Spectrum, donc au total il y’aura nbPixels points dans ce Spectrum.

# Quick Start Guide

1. Vérifier la partie « hardware » et « software » (ie : Châssis, carte XIA, et si besoin le signal de GATE IN ainsi que l’état du device sous Astor)
2. Initialiser le device via **Init**
3. Charger une configuration via **LoadConfigFile**
4. Vérifier le mode de fonctionnement et modifier si besoin les paramètres d’acquisition
5. Lancer l’acquisition via la commande **SNAP**
6. Stopper si besoin l’acquisition via la commande **STOP**

# Annexe – Devices Propriétés/Attributs/Commandes

### Properties

|  |  |  |
| --- | --- | --- |
| Name | Type | Description |
| AutoLoad | boolean | Allow to Reload the last used configuration file (\*.ini) at each init of the device |
| BoardType | string | Define the Board Type  Available values are :  XMAP;//DeviceServers/win32/PluginXIA/xmap/1\_1\_11/  FALCONX;//DeviceServers/win32/PluginXIA/falconx/1\_1\_20/  SIMULATOR; nbModules ;nbChannels ;clock\_in\_ms;delay\_after\_load\_in\_ms |
| BoardTimeBase | string | Timebase factor in seconds of the board. [default= 0.000000320] |
| ConfigurationFiles | Array of string | Define the list of Configuration "\*.INI" files and their associated alias & mode as below :  ALIAS;MODE;FILE\_PATH\_NAME  Available modes are:  MCA  MAPPING\_FULL  MAPPING\_SCA |
| RoiFiles | Array of string | Define the list of rois files "\*.txt" and their associated alias as below :  ALIAS;FILE\_PATH\_NAME |
| StreamItems | Array of string | Define the list of Items saved by the Streamer. (Nexus, CSV, ...) Available values are : Triggers Outputs Icr Ocr RealTime LiveTime DeadTime Channel |

### Attributes

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| currentMode | | |  |  |  |  |
| MCA | **MAPPING\_FULL** | **MAPPING\_SCA** | **Name** | **Type** | **Access** | **Format** |
|  |  |  | boardType | DevString | READ | SCALAR |
|  |  |  | nbBins | DevLong | READ | SCALAR |
|  |  |  | nbChannels | DevLong | READ | SCALAR |
|  |  |  | nbModules | DevLong | READ | SCALAR |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  | PresetType | DevString | READ\_WRITE | SCALAR |
|  |  |  | presetValue | DevDouble | READ\_WRITE | SCALAR |
|  |  |  |  |  |  |  |
|  |  |  | pixelAdvanceMode | DevString | READ\_WRITE | SCALAR |
|  |  |  |  |  |  |  |
|  |  |  | nbPixels | DevLong | READ\_WRITE | SCALAR |
|  |  |  | currentPixel | DevLong | READ | SCALAR |
|  |  |  | currentMode | DevString | READ | SCALAR |
|  |  |  | currentConfigFile | DevString | READ | SCALAR |
|  |  |  | fileGeneration | DevBoolean | READ\_WRITE | SCALAR |
|  |  |  | streamType | DevString | READ\_WRITE | SCALAR |
|  |  |  | streamTargetPath | DevString | READ\_WRITE | SCALAR |
|  |  |  | streamTargetFile | DevString | READ\_WRITE | SCALAR |
|  |  |  | streamNbAcqPerFile | DevLong | READ\_WRITE | SCALAR |
|  |  |  |  |  |  |  |
|  |  |  | *Channels(i)* | DevULong | READ | SPECTRUM |
|  |  |  | *triggerLiveTime(i)* | DevDouble | READ | SCALAR |
|  |  |  | *realTime(i)* | DevDouble | READ | SCALAR |
|  |  |  | *liveTime(i)* | DevDouble | READ | SCALAR |
|  |  |  | *deadTime(i)* | DevDouble | READ | SCALAR |
|  |  |  | *inputCountRate(i)* | DevDouble | READ | SCALAR |
|  |  |  | *outputCountRate(i)* | DevDouble | READ | SCALAR |
|  |  |  | *eventsInRun(i)* | DevULong | READ | SCALAR |
|  |  |  | *roi(i)\_(j)* | DevDouble | READ | SCALAR |
|  |  |  | *hroi(i)\_(j)* | DevULong | READ | SPECTRUM |

i = numéro de channel

j = numéro de roi

### Commands

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Arg in | | Arg out |
| Snap | | DevVoid | DevVoid |
| Stop | | DevVoid | DevVoid |
| LoadConfigFile | | DevString | DevVoid |
| StreamResetIndex | | DevVoid | DevVoid |
| SetRoisFromFile | | DevString | DevVoid |
| SetRoisFromList | | DevVarStringArray | DevVoid |
| GetRois | | DevVoid | DevVarStringArray |
| RemoveRois | | DevLong | DevVoid |
| SaveConfigFile | | DevVoid | DevVoid |
| GetConfigurationsFilesAlias | | DevVoid | DevVarStringArray |
| GetRoisFilesAlias | | DevVoid | DevVarStringArray |

# Annexe – Fichier type MCA

Fichier type pour le mode MCA avec 4 voies activées

(ici mode NONE sans GATE)

[detector definitions]

(…)

[firmware definitions]

(…)

\*\*\*\*\* Generated by Handel -- DO NOT MODIFY \*\*\*\*\*

[default definitions]

S TART #0

alias = defaults\_module1\_0

**mapping\_mode = 0**

peaking\_time = 20.160000

dynamic\_range = 47200.000000

trigger\_threshold = 1001.098633

baseline\_threshold = 1001.098633

energy\_threshold = 0.000000

calibration\_energy = 5900.000000

adc\_percent\_rule = 5.000000

mca\_bin\_width = 10.000000

preamp\_gain = 3.000000

number\_mca\_channels = 2048.000000

detector\_polarity = 0.000000

reset\_delay = 100.000000

gap\_time = 0.240000

trigger\_peaking\_time = 0.100000

trigger\_gap\_time = 0.000000

baseline\_average = 256.000000

**preset\_type = 0 (ou 1,2,3,4)**

**preset\_value = 0**

number\_of\_scas = 0.000000

num\_map\_pixels\_per\_buffer = 0.000000

num\_map\_pixels = 0.000000

**gate\_master = 0 (ou 1)**

sync\_master = 0.000000

sync\_count = 0.000000

**gate\_ignore = 0 (ou 1)**

**gate\_mode = 1 (ou 0)**

lbus\_master = 0.000000

minimum\_gap\_time = 0.060000

synchronous\_run = 0.000000

maxwidth = 1.000000

preamp\_type = 0.000000

decay\_time = 100.000000

peak\_mode = 1.000000

list\_mode\_variant = 2.000000

buffer\_clear\_size = 0.000000

END #0

\*\*\*\*\* End of Generated Information \*\*\*\*\*

[module definitions]

(…)

# Annexe – Fichier type MAPPING FULL

Fichier type pour le mode MCA avec 4 voies activées

[detector definitions]

(…)

[firmware definitions]

(…)

\*\*\*\*\* Generated by Handel -- DO NOT MODIFY \*\*\*\*\*

[default definitions]

START #0

input\_logic\_polarity = 0.000000

**mapping\_mode = 1**

**pixel\_advance\_mode = 1**

**num\_map\_pixels = 1**

**num\_map\_pixels\_per\_buffer = 1000**

**gate\_master = 1**

**gate\_ignore = 0**

peaking\_time = 1.000000

dynamic\_range = 47200.000000

trigger\_threshold = 612.062677

baseline\_threshold = 612.062677

energy\_threshold = 0.000000

calibration\_energy = 2950.000000

adc\_percent\_rule = 2.500000

mca\_bin\_width = 10.000000

preamp\_gain = 4.906844

number\_mca\_channels = 2048.000000

detector\_polarity = 1.000000

reset\_delay = 2.000000

gap\_time = 0.240000

trigger\_peaking\_time = 0.100000

trigger\_gap\_time = 0.000000

baseline\_average = 256.000000

preset\_type = 0.000000

preset\_value = 0.000000

number\_of\_scas = 0.000000

sync\_master = 0.000000

sync\_count = 0.000000

gate\_mode = 0.000000

lbus\_master = 0.000000

minimum\_gap\_time = 0.060000

synchronous\_run = 0.000000

maxwidth = 1.000000

preamp\_type = 0.000000

decay\_time = 2.000000

peak\_mode = 1.000000

list\_mode\_variant = 2.000000

buffer\_clear\_size = 0.000000

END #0

\*\*\*\*\* End of Generated Information \*\*\*\*\*

[module definitions]

(…)

# Annexe – Fichier type MAPPING SCA

[detector definitions]

(…)

[firmware definitions]

(…)

\*\*\*\*\* Generated by Handel -- DO NOT MODIFY \*\*\*\*\*

[default definitions]

**START #0**

alias = defaults\_module1\_0

**mapping\_mode = 2.000000**

**num\_map\_pixels = 900.000000**

**num\_map\_pixels\_per\_buffer = 10.000000**

peaking\_time = 1.000000

dynamic\_range = 47200.000000

trigger\_threshold = 187.255859

baseline\_threshold = 100.830078

energy\_threshold = 0.000000

calibration\_energy = 5900.000000

adc\_percent\_rule = 5.000000

mca\_bin\_width = 1.000000

preamp\_gain = 4.722800

number\_mca\_channels = 4096.000000

detector\_polarity = 1.000000

reset\_delay = 10.000000

gap\_time = 0.240000

trigger\_peaking\_time = 0.500000

trigger\_gap\_time = 0.000000

baseline\_average = 128.000000

**preset\_type = 1.000000**

**preset\_value = 10.000000**

**number\_of\_scas = 1.000000**

**gate\_master = 1.000000**

**sync\_master = 0.000000**

**sync\_count = 0.000000**

**gate\_ignore = 0.000000**

**gate\_mode = 0.000000**

lbus\_master = 0.000000

minimum\_gap\_time = 0.700000

**synchronous\_run = 0.000000**

maxwidth = 1.000000

preamp\_type = 0.000000

decay\_time = 10.000000

peak\_mode = 1.000000

list\_mode\_variant = 2.000000

buffer\_clear\_size = 0.000000

**sca0\_lo = 649.000000**

**sca0\_hi = 900.000000**

**pixel\_advance\_mode = 1.000000**

**END #0**

\*\*\*\*\* End of Generated Information \*\*\*\*\*

[module definitions]

(…)

# Annexe – Fichier type FalconX

[detector definitions]

(…)

[firmware definitions]

(…)

\*\*\*\*\* Generated by Handel -- DO NOT MODIFY \*\*\*\*\*

[default definitions]

START #0

alias = defaults\_module1\_0

analog\_gain = 2.999997

analog\_offset = 0.000000

detector\_polarity = 1.000000

termination = 0.000000

attenuation = 0.000000

coupling = 0.000000

decay\_time = 2.000000

dc\_offset = -0.008553

reset\_blanking\_enable = 1.000000

reset\_blanking\_threshold = -0.019951

reset\_blanking\_presamples = 20.000000

reset\_blanking\_postsamples = 20.000000

detection\_threshold = 0.010002

min\_pulse\_pair\_separation = 2.000000

detection\_filter = 2.000000

mapping\_mode = 1.000000

number\_mca\_channels = 4096.000000

mca\_spectrum\_accepted = 1.000000

mca\_spectrum\_rejected = 0.000000

mca\_start\_channel = 0.000000

mca\_refresh = 0.100000

preset\_type = 1.000000

preset\_value = 30.000000

scale\_factor = 2.162876

mca\_bin\_width = 10.000000

sca\_trigger\_mode = 3.000000

sca\_pulse\_duration = 400.000000

number\_of\_scas = 0.000000

num\_map\_pixels\_per\_buffer = 1024.000000

num\_map\_pixels = 0.000000

pixel\_advance\_mode = 1.000000

input\_logic\_polarity = 0.000000

gate\_ignore = 0.000000

sync\_count = 0.000000

risetime\_optimization = 160.000000

END #0

\*\*\*\*\* End of Generated Information \*\*\*\*\*

[module definitions]

**module\_type = falconxn**

(…)

[module channel data]

START module1

data\_chan0\_len = 11264

**data\_chan0 = (Long list de caractères)**

END module1

1. http://www.xia.com/Datasheets/xMap\_flyer\_161109.pdf [↑](#footnote-ref-1)
2. Le 20.11.16, l’affichage du mode de déclenchement n’apparait pas dans l’ATK [↑](#footnote-ref-2)
3. Manuel XIA p.40 & 41 [↑](#footnote-ref-3)
4. RealTime = temps entre le start et le stop de l’acquisition [↑](#footnote-ref-4)
5. LiveTime = Temps pendant lequel la sortie du Trigger rapide reste au-dessous du seuil. Utilisé pour le ICR [↑](#footnote-ref-5)
6. Seuls les ROIs de la ou les voies choisies sont modifiés. [↑](#footnote-ref-6)
7. Il est nécessaire de fermer et réouvrir l’ATK Panel pour avoir l’affichage complet des différents attributs (Bug d’affichage des attributs dynamique) [↑](#footnote-ref-7)
8. Manuel XIA p.40 & 41 [↑](#footnote-ref-8)