**Merlin EPICS Interface Specification**

Contents

[1. Revision History 2](#_Toc419472841)

[2. History of the protocol 3](#_Toc419472842)

[3. Interface Overview 4](#_Toc419472843)

[4. Command Format 5](#_Toc419472844)

[4.1. Request Format for SET 5](#_Toc419472845)

[4.2. Request Format for CMD/GET 5](#_Toc419472846)

[4.3. Response Format for SET/CMD 5](#_Toc419472847)

[4.4. Response Format for GET 6](#_Toc419472848)

[4.5. Error Codes 6](#_Toc419472849)

[5. Command Examples 6](#_Toc419472850)

[6. Full List of EPICS Commands 8](#_Toc419472851)

[6.1. Driver Variables 8](#_Toc419472852)

[6.2. Execute Commands (CMD type) 8](#_Toc419472853)

[6.3. Medipix3 Modes 8](#_Toc419472854)

[6.4. Acquisition & Trigger Control 9](#_Toc419472855)

[6.5. Threshold Scan Control 11](#_Toc419472856)

[6.6. Local File Saving Control 11](#_Toc419472857)

[6.7. Image Correction 11](#_Toc419472858)

[6.8. File and Status 12](#_Toc419472859)

[7. Data Channel Frame Formats 13](#_Toc419472860)

[7.1. Overview 13](#_Toc419472861)

[7.2. Merlin Quad Acquisition Header 14](#_Toc419472961)

[7.3. Merlin Image Data Format 15](#_Toc419472963)

[Frame Header 15](#_Toc419472964)

[Merlin Quad Version 1 Frame Header Fields. 15](#_Toc419472965)

[Merlin Quad Version 1 Frame Header DAC section. 17](#_Toc419472966)

[7.4. Merlin Colour Image Data Format 17](#_Toc419472970)

[8. Lancelot BPM : Additional Commands 18](#_Toc419472971)

[Command Extensions 18](#_Toc419472972)

[9. Lancelot BPM : Additional data file format 19](#_Toc419472973)

[10. Lancelot BPM : Changes to Acquisition Header 19](#_Toc419472974)

# **Revision History**

|  |  |  |
| --- | --- | --- |
| Issue | Changes | Authors |
| 1.0 | Initial Draft | Matt Pearson |
| 1.1 | Addition of Data channel message format | Richard Plackett  Giles Knap |
| 1.2 | Revisions after review meeting 09/01/2012 | Giles Knap |
| 1.3 | Minor changes to header protocol | Giles Knap |
| 1.4 Draft | Update to include Merlin Quad System | David Omar |
| 2.0 | Update to include final Merlin Quad protocol and additional commands for Lancelot BPM, including data format for profiles. 18/05/2015 | Giles Knap |

# History of the protocol

This document consolidates the information from a number of previous protocol specifications for the following devices:-

* Original Merlin single chip detector
* Merlin Quad chip detector
* Lancelot medipix BPM
* Additional BPM for B21 which was not based on medipix but implemented an extended Merlin protocol

It is proposed that future versions of the EPICS areaDetector driver implement only the interface specified in this document. Separate branches of the source tree will be used to maintain the EPICS drivers for legacy hardware if necessary.

# Interface Overview

The interface to the Merlin system is provided over two TCP/IP sockets. The command socket is used to send commands to Merlin and receive command responses. The data socket is used to receive one data frame per image acquired. Commands are entirely in ASCII and data frames have an ASCII header with binary body.

Default port numbers for Merlin are:

* Command: port 6341
* Data: port 6342

# Command Format

3 command types are supported:

* GET : retrieves the value of a device setting
* SET : modifies the value of a device setting
* CMD: sends a command to the device

## Request Format for SET

<header>,<length>,<type>,<name>,<value>

* <header> = MPX
* <length> = Length of following data in bytes
  + i.e. length of “,<type>,<name>,<value>”.
  + Encoded as 10 chars of ASCII decimal representation with leading zeros.
* <type> = SET
* <name> = name of the variable
* <value>= the value to set

All requests are in ASCII format (upper case).

## Request Format for CMD/GET

<header>,<length>,<type>,<name>

* <header> = MPX
* <length> = Length of following data in bytes
  + i.e. length of “,<type>,<name>”
  + Encoded as 10 chars of ASCII decimal representation with leading zeros.
* <type> = GET or CMD
* <name> = name of the function or variable

All requests are in ASCII format (upper case).

## Response Format for SET/CMD

After each SET or CMD command, an acknowledgement must be returned:

<header>,<length>,<type>,<name>,<error code>

* <header> = MPX
* <length> = Length of following data in bytes
  + i.e. length of “,<type>,<name>,<error code>”
  + Encoded as 10 chars of ASCII decimal representation with leading zeros.
* <type> = SET or CMD
* <name> = name of the function or variable
* < error code > = 0 for no error, or non-zero integer for error.

All responses are in ASCII format (upper case).

## Response Format for GET

For GET commands, the value and error code are both returned as follows:

<header>,<length>,<type>,<name>,<value>,<error code>

* <header> = MPX
* <length> = Length of following data in bytes
  + i.e. length of “,<type>,<name>,<value>,<error code>”
  + Encoded as 10 chars of ASCII decimal representation with leading zeros.
* <type> = GET
* <name> = name of the variable
* <value> = the value of the variable
* <error code> = 0 for no error, or non-zero integer for error.

If error is non zero then value should be ignored.

All responses are in ASCII format (upper case).

## Error Codes

Error numbers are:

0 = No Error / OK

1 = Unknown Error

2 = Unknown Command

3 = Param Out Of Range

>4 TBA

# Command Examples

All fields are ASCII.

**A set command example:**

Request

MPX,0000000025,SET,NUMFRAMESTOACQUIRE,1

Response

MPX,0000000025,SET,NUMFRAMESTOACQUIRE,0

**A get command example:**

Request:

MPX,0000000023,GET,NUMFRAMESTOACQUIRE

Response:

MPX,0000000027,GET,NUMFRAMESTOACQUIRE,1,0

**A CMD command example:**

Request

MPX,0000000021,CMD,STARTACQUISITION

Response

MPX,0000000023,CMD,STARTACQUISITION,0

# Full List of EPICS Commands

## Driver Variables

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Type** | **Notes** | **Format / Range** |
| GETSOFTWAREREVISION | GET | Floating Point | ##.## |

## Execute Commands (CMD type)

| **Name** | **Type** | **Notes** |
| --- | --- | --- |
| STARTACQUISITION | CMD | Start an acquisition. If the start trigger mode is internal, the acquisition will begin immediately, otherwise the detector will arm and wait for a trigger. |
| STOPACQUISITION | CMD | Stop any running acquisition, finishes sending frame |
| SOFTTRIGGER | CMD | Trigger a previously armed acquisition sequence. |
| ABORT | CMD | Aborts any acquisition or process, terminates data stream |
| SELFTEST | CMD | Perform a self test. Report will be stored locally with unique file name, default option for EPICS |
| NOISEEQUALISATION | CMD | Performs equalisation on noise |
| FLATFIELDCORRECTION | CMD | Perform a flat field correction |
| THSCAN | CMD | Perform a threshold scan |
| RESET | CMD | Restores default configuration values |

## Medipix3 Modes

| **Name** | **Type** | **Units** | **Notes** | **Format Range** |
| --- | --- | --- | --- | --- |
| COLOURMODE | SET/GET |  | Colour mode: 0=Monochrome 1=Colour | 0 – 1 |
| CHARGESUMMING | SET/GET |  | 0=Off, 1=ON | 0 – 1 |
| HIGHGAIN | SET/GET |  | 0=Off, 1=ON  This command is only used for the Merlin Single. Merlin Quad used the GAIN command. | 0 – 1 |
| GAIN | SET/GET |  | 0=Super Low 1=Low 2=High 3=Super High | 0-3 |
| CONTINUOUSRW | SET/GET |  | 0=Off, 1=ON | 0 – 1 |
| ENABLECOUNTER1 | SET/GET |  | 0=Off, 1=ON | 0 – 1 |
| THRESHOLD0 | SET/GET | keV | Single | 0-999.99 |
| THRESHOLD1 | SET/GET | keV | Single | 0-999.99 |
| THRESHOLD2 | SET/GET | keV | Single | 0-999.99 |
| THRESHOLD3 | SET/GET | keV | Single | 0-999.99 |
| THRESHOLD4 | SET/GET | keV | Single | 0-999.99 |
| THRESHOLD5 | SET/GET | keV | Single | 0-999.99 |
| THRESHOLD6 | SET/GET | keV | Single | 0-999.99 |
| THRESHOLD7 | SET/GET | keV | Single | 0-999.99 |
| OPERATINGENERGY | SET/GET | keV | Single | 0-999.99 |
| COUNTERDEPTH | SET/GET |  | Integer (1, 4, 12 or 24). For Single with 3.0. Integer (1,6,12,24). Foe Quad with RX. | 1, 6, 12 or 24 |
| TEMPERATURE | GET | degrees C | Single | -100 - 200 |

## Acquisition & Trigger Control

| **Name** | **Type** | **Units** | **Notes** | **Format / Range** |
| --- | --- | --- | --- | --- |
| NUMFRAMESTOACQUIRE | SET/GET |  | Integer. The number of frames to acquire when the next STARTACQUISITION command is sent, or hardware trigger is received. If the value is zero, acquisition should continue until a STOPACQUISITION command. | 0 - 100,000 |
| ACQUISITIONTIME | SET/GET | Milliseconds | Single. The time over which to acquire a frame in milliseconds. | Scientific format |
| ACQUISITIONPERIOD | SET/GET | Milliseconds | Single. The time from the start of one acquisition to the start of the next, in milliseconds. | Scientific format |
| TRIGGERSTART | SET/GET |  | 0=internal (default), 1=rising edge (TTL), 2=falling edge (TTL), 3=rising edge (LVDS), 4=falling edge (LVDS), 5=soft trigger | 0, 1, 2, 3, 4, 5 |
| TRIGGERSTOP | SET/GET |  | 0=internal (default), 1=rising edge (TTL), 2=falling edge (TTL), 3=rising edge (LVDS), 4=falling edge (LVDS | 0, 1, 2, 3, 4 |
| NUMFRAMESPERTRIGGER | SET/GET |  | Integer. Number of frames to acquire per trigger if multiple external triggers. | 1 - 100,000 |
| TriggerOutTTL | SET/GET |  | 0=Trigger In TTL, 1=Trigger In LVDS, 2=Trigger In TTL Delayed, 3=Trigger In LVDS Delayed, 4=Follow Shutter, 5=One per Acq Burst, 6=Shutter and Sensor Readout, 7=Busy | 0-7 |
| TriggerOutLVDS | SET/GET |  | 0=Trigger In TTL, 1=Trigger In LVDS, 2=Trigger In TTL Delayed, 3=Trigger In LVDS Delayed, 4=Follow Shutter, 5=One per Acq Burst, 6=Shutter and Sensor Readout, 7=Busy | 0-7 |
| TriggerOutTTLInvert | SET/GET |  | 0=Normal 1=Inverted | 0-1 |
| TriggerOutLVDSInvert | SET/GET |  | 0=Normal 1=Inverted | 0-1 |
| TriggerOutTTLDelay | SET/GET | nS | Long Interger | 0-68719476720 |
| TriggerOutLVDSDelay | SET/GET | uS | Long Interger | 0-68719476720 |
| TriggerUseDelay | SET/GET |  | 0=No Delay 1=Apply delay to start and stop | 0-1 |

## Threshold Scan Control

| **Name** | **Type** | **Units** | **Notes** | **Format / Range** |
| --- | --- | --- | --- | --- |
| THSCAN | SET/GET |  | Integer 0-7. | 0 - 7 |
| THWINDOWMODE | SET/GET |  | 0=Off, 1=ON | 0 – 1 |
| THWINDOWSIZE | SET/GET | keV | Single The set size between counter 0 and 1. | 0-999.99 |
| THSTART | SET/GET | keV | Single Start energy of the scan. | 0-999.99 |
| THSTOP | SET/GET | keV | Single Stop energy of the scan. | 0-999.99 |
| THSTEP | SET/GET | keV | Single Scan step size. | 0-999.99 |
| THNUMSTEPS | SET/GET |  | Integer Fixes the number bof steps to avoid ambiguities around end limit. | 0-511 |

## Local File Saving Control

| **Name** | **Type** | **Units** | **Notes** | **Format / Range** |
| --- | --- | --- | --- | --- |
| FILEDIRECTORY | SET/GET |  | String. 256 chars max. | Windows File Path Format |
| FILENAME | SET/GET |  | String. 256 chars max. | Windows File Name Format |
| FILEFORMAT | SET/GET |  | 0=Binary, 1=ASCII | 0 -1 |
| FILECOUNTER | SET/GET |  | Integer. Set the file number to start at. | 1 - 100,000 |
| FILEENABLE | SET/GET |  | 0=Disable, 1=Enable. When enabled the Labview software saves each frame to the local disk. | 0 - 1 |

## Image Correction

| **Name** | **Type** | **Units** | **Notes** | **Format / Range** |
| --- | --- | --- | --- | --- |
| FLATFIELDCORRECTION | SET/GET |  | 0=OFF, 1=ON | 0 - 1 |
| DEADTIMECORRECTION | SET/GET |  | 0=OFF, 1=ON | 0 - 1 |
| MASKINDATA | SET/GET |  | 0=OFF, 1=ON | 0 - 1 |

## File and Status

| **Name** | **Type** | **Units** | **Notes** | **Format / Range** |
| --- | --- | --- | --- | --- |
| DACFILE | SET/GET |  | String (full path and filename). 256 chars max. | Windows File Path Format |
| PIXELMATRIXLOADFILE | SET/GET |  | String (full path and filename). 256 chars max. | Windows File Name Format |
| PIXELMATRIXSAVEFILE | SET/GET |  | String (full path and filename). 256 chars max. | Windows File Name Format |
| DETECTORSTATUS | GET |  | 0=IDLE, 1=BUSY | 0 - 1 |

# Data Channel Frame Formats

## Overview

All Merlin data channel responses begin with the MPX identifier followed by length and then by one of the following:

* HDR – An Acquisition header
* MQ1 – A Merlin Quad V1 data frame
* *PR1 – A Profile data frame (not supported by standard Merlin detector – see later sections for Lancelot BPM additional features)*

On receipt of a start acquisition command:

MPX,<length>,CMD,STARTACQUISITION

Merlin will immediately respond with the following on the command channel:

MPX,<length>,CMD,STARTACQUISITION,0

Then immediately on the data channel (assuming the command can proceed) an Acquisition header will be sent to record details about the system settings for this acquisition. The header will be formatted as follows:

MPX,<length>,HDR,<ASCII Acquisition Header>

<length> = Length of following data in bytes, i.e. length of “HDR,<ASCII Acquisition Header>”. Length is encoded as 10 chars of ASCII decimal representation with leading zeros.

The <ASCII Acquisition Header> is always 2044 bytes long for a total ok 2K bytes including the HDR field.

Immediately following the acquisition of the first frame the data frame will be sent on the data channel with an embedded frame header as follows.

MPX,<length>,MQ1,<ASCII Header><pixel data>

This will be repeated after every frame until the acquisition is complete.

See following sections for details of the acquisition header and data frames.

## Merlin Quad Acquisition Header

The acquisition header is 2kB (less 4 bytes) of ASCII in a verbose human readable format as follows in the below example.

|  |
| --- |
| HDR,  Chip ID: W117\_E7,W117\_H7,W117\_I7,W117\_G7  Chip Type (Medipix 3.0, Medipix 3.1, Medipix RX): Medipix3RX  Assembly Size (1X1, 2X2): 2x2  Chip Mode (SPM, CSM, CM, CSCM): SPM  Counter Depth (number): 12  Gain: HGM  Active Counters: Counter 0  Thresholds (keV): 0.000000E+0,1.000000E+1,1.500000E+1,2.000000E+1,2.500000E+1,3.000000E+1,3.500000E+1,4.000000E+1  DACs: 030,056,083,111,139,167,194,222,100,010,125,125,100,100,080,100,090,050,128,004,255,148,128,203,189,417,417; 030,056,083,111,139,167,194,222,100,010,125,125,100,100,080,100,090,050,128,004,255,142,128,192,180,417,417; 030,056,083,111,139,167,194,222,100,010,125,125,100,100,080,100,090,050,128,004,255,151,128,205,191,417,417; 030,056,083,111,139,167,194,222,100,010,125,125,100,100,080,100,090,050,128,004,255,138,128,189,181,417,417  bpc File: c:\MERLIN Quad Host\Config\W117\_E7\W117\_E7\_SPM.bpc,c:\MERLIN Quad Host\Config\W117\_H7\W117\_H7\_SPM.bpc,c:\MERLIN Quad Host\Config\W117\_I7\W117\_I7\_SPM.bpc,c:\MERLIN Quad Host\Config\W117\_G7\W117\_G7\_SPM.bpc  DAC File: c:\MERLIN Quad Host\Config\W117\_E7\W117\_E7\_SPM.dacs,c:\MERLIN Quad Host\Config\W117\_H7\W117\_H7\_SPM.dacs,c:\MERLIN Quad Host\Config\W117\_I7\W117\_I7\_SPM.dacs,c:\MERLIN Quad Host\Config\W117\_G7\W117\_G7\_SPM.dacs  Gap Fill Mode: None  Flat Field File: Dummy (C:\<NUL>\Temp.ffc)  Dead Time File: Dummy (C:\<NUL>\Temp.dtc)  Acquisition Type (Normal, Th\_scan, Config): Normal  Frames in Acquisition (Number): 1000  Trigger Start (Positive, Negative, Internal): Internal  Trigger Stop (Positive, Negative, Internal): Internal  Frames per Trigger (Number): 1  Time and Date Stamp (yr, mnth, day, hr, min, s): 10/12/2013 17:36:32  Sensor Bias (V, µA) 20 V  Sensor Polarity (Positive, Negative): Positive  Temperature (C): FPGA Temp 37.250000 Deg C  Medipix Clock (MHz): 120MHz  Readout System: Merlin Quad  Software Version: Development  End |

## Merlin Image Data Format

### Frame Header

The frame header is intended to record settings that could change from frame to frame as well as provide enough information to read back the data, even in the absence of the Acquisition Header. The header is represented a text string divided into comma separated fields. The header will also be maintained as a fixed length to ease reading in certain applications.

The frame header size will be 256 bytes plus 128 bytes per chip (768 for a typical Quad). All the data is single dimension comma separated, with no extra structure delimiting the DAC blocks. The main block and the DAC blocks are concatenated directly, with all the padding following the end of all the header data.

All Merlin image data is encoded as raw binary with big-endian data order.

The pixel data is sent in rows from left to right, starting with bottom row and proceeding sequentially to the top. The dimensions and bit depth of the pixel data will vary according to the values in the header.

For Lancelot the same applies

### Merlin Quad Version 1 Frame Header Fields.

| Characters (excluding comma) | Data Type | Field Name | Description (MQ1) |
| --- | --- | --- | --- |
| 3 | String | Header ID | Identifies the Frame Header start and allows version identification should any fields change in future. Frames may contain data for one, or many chips.  MQ1: Merlin Quad version 1 image.  PR1: Merlin Quad version 1 profile. |
|  | U32 | Acquisition Sequence Number | This is the number of a frame within an acquisition sequence. This number will reset with each start acquisition. |
|  | U16 | Data Offset | The offset from the beginning of the header to the beginning of the image data. The same as the total header length. This will also be the offset from the beginning of the file for the first image, or if there is only one per file. |
|  | U8 | Number of Chips | Self explanatory. This may be used with the sensor layout and chip select fields to determine the overall sensor and data configuration. |
|  | U32 | Pixel Dimension X | The image size in the X (width) direction. |
|  | U32 | Pixel Dimension Y | The image size in the Y (height) direction. |
| 3 | String | Pixel Depth | For ease of processing, the pixel size is rounded to a 2^N multiple number of byes. The only exception is single bit mode where eight pixels are packed in a byte. This field represents the pixel depth as a U8 number of bits, with “U” pre-pended.  Valid values include: U01, U08, U16, U32 and U64. |
| 6 | String | Sensor Layout | Text string padded with leading spaces,  2x2, Nx1, 2x2G, Nx1G |
|  | Hex U8 | Chip Select | This is a bit field representation in hexadecimal of the chips that were active during the capture of the frame. Chip 1 is the least significant bit, chip 2, the next bit etc. |
| 26 |  | Time Stamp | Date and time to nearest uS.  Format: yyyy-mm-dd hh:mm:ss.ssssss  Eg 2013-09-17 13:01:53.744951  The time stamp is not intended to be accurate in an absolute sense, but it does give accurate relative timings between frames in the same sequence. |
|  | Double | Acquisition Shutter Time | Floating point representation of the shutter open time in seconds. |
|  | U8 | Counter | Counter 0, or 1. In colour mode, this will represent the colour layer, equivalent to Threshold 0 to 7. |
|  | U8 | Colour Mode |  |
|  | U8 | Gain Mode | 0 = SLGM, 1 = LGM, 2 = HGM, 3 = SHGM. |
|  | Single | Threshold (0..7) | The values of the thresholds in keV. |
|  | Group of fields | Optional Fields | (SEE BELOW) These are not currently implemented by Merlin, they are for use in Lancelot. |
|  | Groups of fields | DACs | One section per chip. (SEE BELOW) |
| n | String | Padding | Null. |

### Merlin Quad Version 1 Frame Header DAC section.

| Characters (excluding comma) | Data Type | Field Name | Description (MQ1 DACs) |
| --- | --- | --- | --- |
| 3 | String | DAC Format | Medipix 3.0, or RX DAC layout.  MQ1 only supports RX, so only the RX layout is given here.  Current possible values “3RX”, “3.0” |
| 3 | U16 | Threshold 0 | 9 bit DAC. |
| 3 | U16 | Threshold 1 | 9 bit DAC. |
| 3 | U16 | Threshold 2 | 9 bit DAC. |
| 3 | U16 | Threshold 3 | 9 bit DAC. |
| 3 | U16 | Threshold 4 | 9 bit DAC. |
| 3 | U16 | Threshold 5 | 9 bit DAC. |
| 3 | U16 | Threshold 6 | 9 bit DAC. |
| 3 | U16 | Threshold 7 | 9 bit DAC. |
| 3 | U8 | Preamp |  |
| 3 | U8 | Ikrum |  |
| 3 | U8 | Shaper |  |
| 3 | U8 | Disc |  |
| 3 | U8 | Disc LS |  |
| 3 | U8 | Shaper Test |  |
| 3 | U8 | DAC Disc L |  |
| 3 | U8 | DAC Test |  |
| 3 | U8 | DAC DISC H |  |
| 3 | U8 | Delay |  |
| 3 | U8 | TP Buff In |  |
| 3 | U8 | TP Buff Out |  |
| 3 | U8 | RPZ |  |
| 3 | U8 | GND |  |
| 3 | U8 | TP Ref |  |
| 3 | U8 | FBK |  |
| 3 | U8 | Cas |  |
| 3 | U16 | TP Ref A | 9 bit DAC. |
| 3 | U16 | TP Ref B | 9 bit DAC. |

### Merlin Quad Version 1 Optional Fields section (currently Lancelot Only).

| Characters (excluding comma) | Data Type | Field Name | Description |
| --- | --- | --- | --- |
| 4 | String | Optional Block Header | Always Contains the text “OPT1” and allows the parser to determine that the optional block is present |
| 5 | U16 | ROI X | X of top left pixel of ROI |
| 5 | U16 | ROI Y | Y of top left pixel of ROI |
| 5 | U16 | ROI Width | Width of ROI |
| 5 | U16 | ROI Height | Height of ROI |
| 5 | U16 | Profile Select | A bit mask representing the types of profile included in the data frame.  Bit 0 – image,  Bit 1 – raw X profile,  Bit 2 – raw Y profile,  Bit 3 – I0 (image sum), |
| 4 | String | Optional Block Terminator | Always contains the text “END1” and instructs the parser to move to interpreting the DAC block. This explicit terminator allows for backwards compatible extension to the protocol. |

## Merlin Colour Image Data Format

Colour images are sent as a series of colour layer images. Each layer has a header and image data in the standard format. The number of layers may be four, or eight, depending on the counter mode. Four for single counter operation, eight for dual count operation.

# Lancelot BPM: Additional Commands

### Command Extensions

| **Command** | **Type** | **Format** | **Description** |
| --- | --- | --- | --- |
| **Image correction** | | | |
| BCKGRND | SET/GET/CMD | 0 - 128 | Number of frames to average when collecting a background (dark) image. Currently, only power-of-2 values are supported. If a non-power-of-2 value is passed, the highest power-of-2 value smaller than the passed value will be used. Zero value is treated as one – a single background frame will be acquired.  CMD command starts background image acquisition. The set number of images will be collected, averaged and saved in the internal memory. The final background image will be sent back to the client in the standard way via the data channel.  Use BCKGRNDCORRECTION command to enable background correction. |
| BCKGRNDCORRECTION | SET/GET | 0 - 1 | Enables (1) or disables (0) background correction of the collected images. Corrected images are the result of subtraction of the background image from the raw image. |
| DETECTORSTATUS | GET | Unsigned integer. | Bit 0 – detector idle (0) / busy (1)  Bit 1 – pressure normal (0) / high (1)  Bit 2 – cooling ok (0) / failed (1)  Bit 3 – FBK ok(0) / out-of-range (1)  Bit 4 – HV ok (0) / failed (1))  Bit 5 – detector ok (0) / failed to configure properly (1)  Bit 6 – background valid (0) / invalid (1)  For B21 XBPM, bits 1, 2, 3, 4 are always 1. Bit 6 indicates whether the collected background image is valid in respect to selected image ROI and acquisition time. |
| IMAGESTOSUM | SET/GET | 0 – 128 | Number of frames to sum before sending the result to the client. This command will have effect during image (STARTACQUISITION) and profile (PROFILES) acquisition. Use 0 or 1 to avoid summing. |
| IMGAVERAGE | SET/GET | 0 – 1 | Enables (1) or disables (0) frame averaging. This command will have effect during image (STARTACQUISITION) and profile (PROFILES) acquisition. Number of frames to average is specified by IMAGESTOSUM. Currently, only power-of-2 values are supported. If a non-power-of-2 value is passed, the highest power-of-2 value smaller than the passed value will be used. |
| **Image size** | | | |
| ROI | SET/GET | 4 space separated (ASCII code 0x20) unsigned 16-bit integers in format  <X> <Y> <W> <H> | Contains location and size of image ROI:  X – first pixel in horizontal direction (column, 0 – sensor width – 1)  Y – first pixel in vertical direction (row, 0 – sensor height – 1)  W – width, number of columns (1 – sensor width)  H – height, number of rows (1 – sensor height) |
| SENSORSIZE | GET | 2 space separated (ASCII code 0x20) unsigned 16-bit integers in format <W> <H> | Contains sensor size in pixels:  W – width, number of columns  H – height, number of rows |
| **Image profiles acquisition** | | | |
| PROFILES | SET/GET/CMD | 0 - 65535 | Every bit corresponds to the data to be included in profile package:  Bit 0 – image  Bit 1 – raw X profile  Bit 2 – raw Y profile  Bit 3 – I0 (image sum)  The server should ensure that all bits of PROFILES are not zero at the same time. It is suggested that, if the client attempts to set PROFILES = 0, the server enforces a default value (currently 0x06 ­­– bit 1 and 2 are non-zero).  CMD command starts profile acquisition. The profile data will be sent via the data channel in the package with type PR1 for example:  MPX,<length>,PR1,<ASCII Header><Data load>  See next section for Data Frame format of PR1 frame. |
| IMGONPROFILEN | SET/GET | 0 – LONG\_MAX | If non-zero, then a full image will be included in the profile package every Nth frame even if bit 0 of PROFILES is not set. |

# Lancelot BPM: Data File Format Differences

The section “Merlin Image Data Format” describes the data frame header for both Merlin and Lancelot detectors. The Lancelot header is identical to the Merlin header except for the addition of the optional block which adds information on ROI and profiles.

The Lancelot pixel data is different from Merlin because it can be affected by the ROI and profile settings. Up to 4 sections of pixel data will be packed together in the following order, with the presence of each section indicated by the bit mask “Profile Select”. Full image data is Big-endian and read from

* *Full Image pixel data: (ROI\_Width) × (ROI\_Height) × (pixel depth) bits*
* *X profile data: 1 x (ROI\_Width) × 64 bits*
* *Y profile data: 1 x (ROI\_Height) × 64 bit*
* *Image sum: 1 × 64 bit*

All Merlin image data is encoded as raw binary with big-endian data order.

The pixel data is sent in rows from left to right, starting with bottom row and proceeding sequentially to the top.

# Lancelot BPM: Changes to Acquisition Header

The acquisition header is free form human readable text. For reference, Lancelot makes the following changes to the default Merlin header:-

1. Acquisition Type will include the summation and averaging modes, for example “normal sum” will indicate that image summation is active and “profile average” will respectively indicate that profiles of averaged images are calculated.
2. Frames Summed is the new field that will indicate how many images are summed or averaged. This field will be added between Frames in Acquisition and Trigger Start fields.
3. Correction field will indicate if any corrections are applied to the data. Currently the values “none” and “background” will be implemented. This field will be added between Dead Time File and Acquisition Type fields.