# Usage of Type0003 Module

Rev. 1.0

April 13, 2009

## 1 Introduction

This document describes supplement things to use the module. Some of these are restriction of the current version module.

## 2 Supported camera

The Type0003 module supports D90, D5000.

The module cannot control two or more cameras, can control one camera only.

## 3 Environment

| OS type | Version |
|---|---|
| Windows | Windows XP Home Edition (SP3) / Professional (SP3)<br>Windows Vista SP1 32bit each edition<br>(Home Basic / Home Premium / Business / Enterprise / Ultimate) |
| Macintosh | MacOS X 10.4.11<br>MacOS X 10.5.5 |

## 4 Capabilities

Client should acquire the value of each Capability once now after opening of Source object. (There is no necessity for acquiring the value every time before setting the value.) When the setting of the value is executed by kNKMAIDCommand_CapSet now without acquiring the value, the value to which Client did set might not be correctly set to the camera.

### *4.1 kNkMAIDCapability_ProgressProc*

The module notifies progress information through MAIDProgress function. When the module can't compute how much the task is finished, the module will call MAIDProgress function with ulTotal = 0 and ulDone = Non-0. When the task has finished, the module will call function with ulDone = ulTotal.

### *4.2 kNkMAIDCapability_EventProc*

MAID3.1 specification says that the client doesn't have to set MAIDEvent function to kNkMAIDCapability_EventProc. But the current module assumes that the client always sets the MAIDEvent function. So if the client doesn't set MAIDEvent function to EventProc, there are following restrictions to use the module.

1) The client can't use kNkMAIDCommand_EnumChildren.
2) The client can't support lens exchange and device turn off and on.
3) The module doesn't notify changing of capability value, so the client should keep checking these values.

### *4.3 kNkMAIDCapability_Children*

The client may use this capability to enumerate the child objects. The client also can use kNkMAIDCommand_EnumChildren for same purpose. If the client doesn't set MAIDEvent function to kNkMAIDCapability_EventProc, the client should use kNkMAIDCapability_Chilren to enumerate the child objects.

### *4.4 kNkMAIDCapability_PictureControlData*

The camera decides whether the camera uses the setting value of Picture Control data, or the value that camera decides internally according to the following setting of Picture Control data.

1) QuickAdjustFlag (Color)

   If this value is valid(1), the camera uses only the value of "QuickAdjust".

   If this value is invalid(0), the camera uses the following value, "Saturation", "Hue", "Sharpening", "Contrast", "Brightness", "CustomCurveFlag", "CustomCurveData", and does not use the value of "QuickAdjust".

2) CustomCurveFlag

   If this value is "Custom Curve used"(1), the camera does not use "Contrast", "Brightness".

3) Toning (Monochrome)

   If this value is B&W(0), the camera does not use "ToningDensity".

4) Contrast, Brightness, CustomCurveFlag, CustomCurveData

   If the value of kNkMAIDCapability_Active_D_Lighting is set to the value except for "Off"(3), the camera does not use "Contrast", "Brightness", "CustomCurveFlag", and "CustomCurveData".

### *4.5 kNkMAIDCapability_DeleteDramImage*

The timing of deletion for DRAM image is limited to the following 2 cases. The module does not support the deletion on the timing excluding the following 2 cases.

1. Before receiving kNkMAIDEvent_AddChild for SourceObject.
2. After issuing kNkMAIDCapability_Acquire for ImageObject,
   and before issuing kNkMAIDCommand_Close.

The case 1  example of the command sequence is shown to the following table.

| No | Command / Capability / Event | Object Type |
|---|---|---|
| 1 | kNkMAIDCapability_Capture | Source |
| 2 | kNkMAIDCommand_Async | Source |
| 3 | kNkMAIDEvent_AddPreviewImage | Source |
| 4 | kNkMAIDCapability_CurrentPreviewID | Source |
| 5 | kNkMAIDCapability_DeleteDramImage | Source |

The case 2  example of the command sequence is shown to the following table.

| No | Command / Capability / Event | Object Type |
|---|---|---|
| 1 | kNkMAIDCapability_Capture | Source |
| 2 | kNkMAIDCapability_Children | Source |
| 3 | kNkMAIDCommand_Open | Item |

| | | |
|---|---|---|
| 4 | kNkMAIDCapability_Children | Item |
| 5 | kNkMAIDCommand_Open | Image |
| 6 | kNkMAIDCapability_DataProc (Set) | Image |
| 7 | kNkMAIDCapability_Acquire | Image |
| 8 | kNkMAIDCommand_Async | Image |
| 9 | kNkMAIDCommand_Abort | Image |
| 10 | kNkMAIDCapability_CurrentPreviewID | Source |
| 11 | kNkMAIDCapability_DeleteDramImage | Source |
| 12 | kNkMAIDCapability_DataProc (Reset) | Image |
| 13 | kNkMAIDCommand_Close | Image |
| 14 | kNkMAIDCommand_Close | Item |

When the callback function was set to kNkMAIDCapability_ProgressProc, the termination of operation will be notified with the parameters of callback function, "ulDone == ulTotal" or "ulDone == ulTotal==0". But when the client aborts the operation by kNkMAIDCommand_Abort, the termination of operation will not be notified.

### 4.6 *kNkMAIDCapability_GetPreviewImageNormal*

A Normal Preview data might not be able to be acquired. A Normal Preview data is a specification deleted from the inside of the camera when acquiring a main image is finished. Current module is a specification that immediately begins the acquisition of a main image regardless of the request of the client when the generation completion notification of a main image is received from the camera.

After module acquired the image, the image data is deleted from the inside of the camera.

Therefore, the acquisition of A Normal Preview data is not a function secured 100%.

However, if client could acquire a Normal Preview data even once, the module generates the Normal Preview cache internally, so the client can get data after receiving kNkMAIDEvent_AddChild event, until Item object is done in close.

About a Low Preview data, module acquires a Low Preview and generates the Low Preview cache before the acquisition of a main image, so the client can get data after receiving kNkMAIDEvent_AddChild event, until Item object is done in close.

## 5 Image and Thumbnail Data

An image data file is transferred from the module through MAID Data Delivery Function. (refer to 5.27 File Data Delivery Structure and 10.3 MAID Data Delivery Function in MAID3.DOC)

All thumbnail images are raw byte data in order of RGBRGBRGB…. The pixel order is from left to right and from top to bottom. The size of thumbnail image is fixed as follows.

Width: 160 pixels Height: 120 pixels

## 6 Connection with camera

If the client sends kNkMAIDCommand_Async to the module, it can know the camera is connected with PC through AddChild event for source object. When the module detects the camera is turned off, the module sends RemoveChild event for the current opened source object.

## 7 Opening object

The client can open only one object at same object type(eNkMAIDObjectType). (e.g. If there are two source object with different ID, client can open either one at the same time.)

But exceptional case, image and thumbnail object, these are belong to kNkMAIDObjectType_DataObj, can be opened at the same time, from same ID Item object.

## 8 The restriction while executing live view.

While the Live view is executing, the use of a lot of capability will become prohibition.

When there is no description that this capability is usable with executing live view on the document "MAID3Type0003(E).pdf", the use of this capability will become prohibition.

But though the use of this capability is changed to prohibition, the ulVisibility and ulOperations of this capability will not be changed basically and keeps the current settings of ulVisibility and ulOperations. (Capability_AFCapture and Capability_PreCapture, etc. also have a part of exception.)

The usable capability when live view is executing is as follows.

| |
|---|
| kNkMAIDCapability_ContrastAF |
| kNkMAIDCapability_MFDriveStep |
| kNkMAIDCapability_MFDrive |
| kNkMAIDCapability_ContrastAFArea |
| kNkMAIDCapability_LiveViewStatus |
| kNkMAIDCapability_LiveViewProhibit |
| kNkMAIDCapability_LiveViewImageZoomRate |
| kNkMAIDCapability_Capture |
| kNkMAIDCapability_GetLiveViewImage |

## 9 Structure Member Alignment

The following list is structure member alignment of the module and client. In MAID3.H, there is a comment saying that all alignments are 4byte, but this value depends on platform.

| Platform | Alignment |
|---|---|
| Windows 32bit | 2 byte |
| Macintosh | 4 byte |

## 10 Macintosh

Type0003 module for Macintosh(Type0003 Module.bundle) is Universal Binary.