
Usage of Type0001 Module

Rev. 1.0

October 25, 2007

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 Type0001 module supports D3, D300. The module cannot control two or more cameras, can control one camera only.

3 Capabilities

Client must get the current value of a capability by `kNkMAIDCommand_CapGet`, before setting value by `kNkMAIDCommand_CapSet`. If client sets value of a capability without getting, the value client try to set may not be set to camera correctly.

3.1 *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`.

3.2 *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 to check these value.

3.3 *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.

3.4 *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) Active D-Lighting

If the value of Active D-Lighting in shooting menu (kNkMAIDCapability_Active_D_Lighting) is ON(High/Normal/Lo), the camera does not use “Contrast”, “Brightness”, “CustomCurveFlag”, “CustomCurveData”.

3.5 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 parameter 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 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 image 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

5 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.

6 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.

7 Acquiring image

The image data on DRAM is removed after executing `kNkMAIDCapability_Acquire` of image object. After the image data on DRAM is removed, some capabilities of the image and thumbnail objects(e.g. `Capability_FileHeader`, `Capability_Pixel`), that refer the image data on DRAM, occur error in getting its value. So, the client have to get capability value from the image and thumbnail objects if necessary, before acquiring image data.

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 “MAID3Type000.doc”, 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.

Live view mode	capability
Hand-held	<code>kNkMAIDCapability_FocusPreferredArea</code>

		kNkMAIDCapability_AutoFocus
Tripod		kNkMAIDCapability_ContrastAF
		kNkMAIDCapability_MFDriveStep
		kNkMAIDCapability_MFDrive
		kNkMAIDCapability_ContrastAFArea
Hand-held / Tripod		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

Type0001 module for Macintosh(Type0001 Module.bundle) is Universal Binary.

11 History

Rev.1.0 Oct. 25, 2007 First version