Usage of Type0016 Module

Rev. 2.1

June 14, 2017

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 Type0016 module supports D5500, D5600. The module cannot control two or more cameras, can control one camera only.

3 Environment

OS type	Version
Windows	Windows 7 (SP1) ---- 32bit / 64bit edition (* Ultimate, Enterprise, Professional, Home Premium, Home Basic) Windows 8.1 ---- 32bit / 64bit edition (* Windows 8.1, Pro, Enterprise) Windows 10 ---- 32bit / 64bit edition
Macintosh	Mac OS X 10.10.5 (Yosemite) Mac OS X 10.11.6 (El Capitan) macOS 10.12.4 (Sierra) * 64bit mode only (32bit mode is not supported)

4 About the usage environment of Module SDK for Macintosh

We recommend that the application using Module SDK for Macintosh uses base SDK 10.12.

5 Runtime Library

If you want to use the Module in the Windows environment, you must install "microsoft visual c++ redistributable package for visual studio 2013".

6 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 without acquiring the value, the value to which Client did set might not be correctly set to the camera.

6.1 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.

6.2 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.

6.3 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.

6.4 kNkMAIDCapability_PictureControlData, kNkMAIDCapability_PictureControlDataEx

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", "Clarity"(Second generation) ,"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”, “CustomCurveData”.

6.5 kNkMAIDCapability_DeleteDramImage

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

- After issuing kNkMAIDCapability_Acquire, and before issuing kNkMAIDCommand_Close for Image Object.

The 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_CurrentItemID	Source
11	kNkMAIDCapability_DeleteDramImage	Source
12	kNkMAIDCapability_DataProc (Reset)	Image
13	kNkMAIDCommand_Close	Image
14	kNkMAIDCommand_Close	Item

The execution of kNkMAIDCapability_Acquire is needed before the execution of kNkMAIDCapability_DeleteDramImage. So, in the case of small data size image, JPEG Basic, the all of image data may complete reading by the kNkMAIDCapability_Acquire before issuing of deletion command. In that case, the error doesn't occur when the deletion command is executed, but the image will be saved in client program.

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.

6.6 kNkMAIDCapability_Capture

When you run the shooting with recording media SDRAM, you must issue

kNkMAIDCommand_Open about Image of Item Object to be generated under the Source Object. And you must issue kNkMAIDCapability_Acquire to get all, or issue kNkMAIDCapability_DeleteDramImage to remove.
You must close the Item Object rapidly after completion of acquired or removed, since the module can not detect the state change of the camera during the period open for Item Object.
If you do not run the deletion or acquisition of Image, there are cases where the next shooting or later can not be carried out successfully.

6.7 kNkMAIDCapability_AFCapture

There is a same restriction of kNkMAID_CapabilityCapture.

6.8 kNkMAIDCapability_CaptureDustImage

There is a same restriction of kNkMAID_CapabilityCapture.

6.9 kNkMAIDCapability_MovRecInCardStatus

After recording video, for Video of Item Object to be generated under the Source Object, you must issue kNkMAIDCommand_Open always. If you need to get the Video, you issue the kNkMAIDCapability_GetVideoImage. (Video acquisition not required)
You must close the Item Object rapidly after open or video acquisition, since the module can not detect the state change of the camera during the period open for Item Object.

7 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
The thumbnail image may not be acquired by the timing. (refer to 4.19. Acquire, MAID3Type0016.doc)

8 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 module object. When the module

detects the camera is turned off, the module sends RemoveChild event for the current opened module object.

9 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.

10 The restriction of bulb photography

When the client shoots bulb photography with module, the maximum exposure time is 59 minutes 59 seconds. If the client shoots bulb photography with the exposure time more than maximum exposure, shooting will not be guareanteed.

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

No	Capability,Command	Precautions
1	kNkMAIDCapability_Capture	In case of bulb photography, the return value will be kNkMAIDResult_BulbReleaseBusy
(2)	kNkMAIDCommand_Async	Until issue TerminateCapture, the client can issue Async optionally repeatedly. The maximum time from Capture and TerminateCapture (= the maximum exposure time) is 59 minutes 59 seconds.
3	kNkMAIDCapability_TerminateCapture	The client must issue TerminateCapture within 59 minutes 59 seconds from Capture issued. If long exposure noise reduction setting is ON, see 10.4.

11 The restriction about D5500, D5600.

11.1 Live view

The capabilities that can be set during live view refer to 8. Capability table that can be set during live view photography, movie live view, movie recording, in MAID3Type0016(E).pdf.

11.2 AF-F Shooting

When shooting a Live View on the camera, Focus Point information will not be attached to the recorded images if the AF mode is set to AF-F not using Capability_ContrastAF.

The Focus Point information will be attached to the recorded images if the client issues Capability_ContrastAF then issues Capability_Capture within a second after the camera has focused while shooting a Live View with AF-F.

11.3 When long exposure noise reduction is ON

In case of "Long Exposure NR" is ON, the time until the image is created from the start of exposure is twice the exposure time. The module doesn't return control until the creation of image has complete.

In case of bulb photography, image generation is started after running kNkMAIDCapability_TerminateCapture. The time to complete the generation of images from the execution of kNkMAIDCapability_TerminateCapture will need the time same as exposure time, meanwhile, the module doesn't return control.

11.4 Auto Bracketing

Shutter speed and Aperture cannot be changed when auto bracketing is in effect.

11.5 Depth-of-field preview button

Module may not work properly while the camera's Depth-of-field preview button is pressed.

12 The restriction on Macintosh

Type0016 module for Macintosh (Type0016 Module.bundle) works on 64bit mode only.(Not supported works on 32 bit mode)

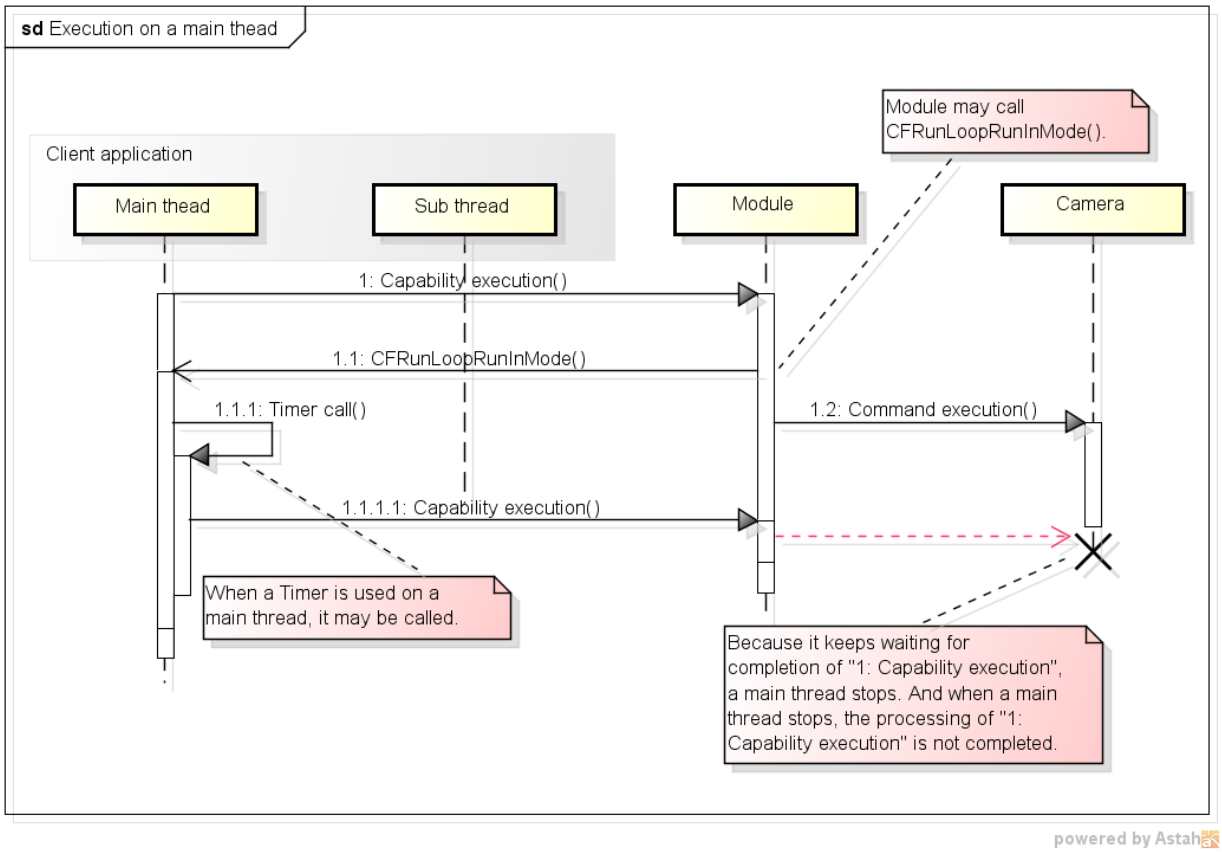
After connecting your camera to your PC, please wait to start module until the memory card access lamp stops flashing.

Client application must not stop a main thread during execution of Capability when you use Module for Macintosh.

When client stops a main thread during execution of Capability, Module may not return from processing of that Capability, because Module can't receive the response from a camera.

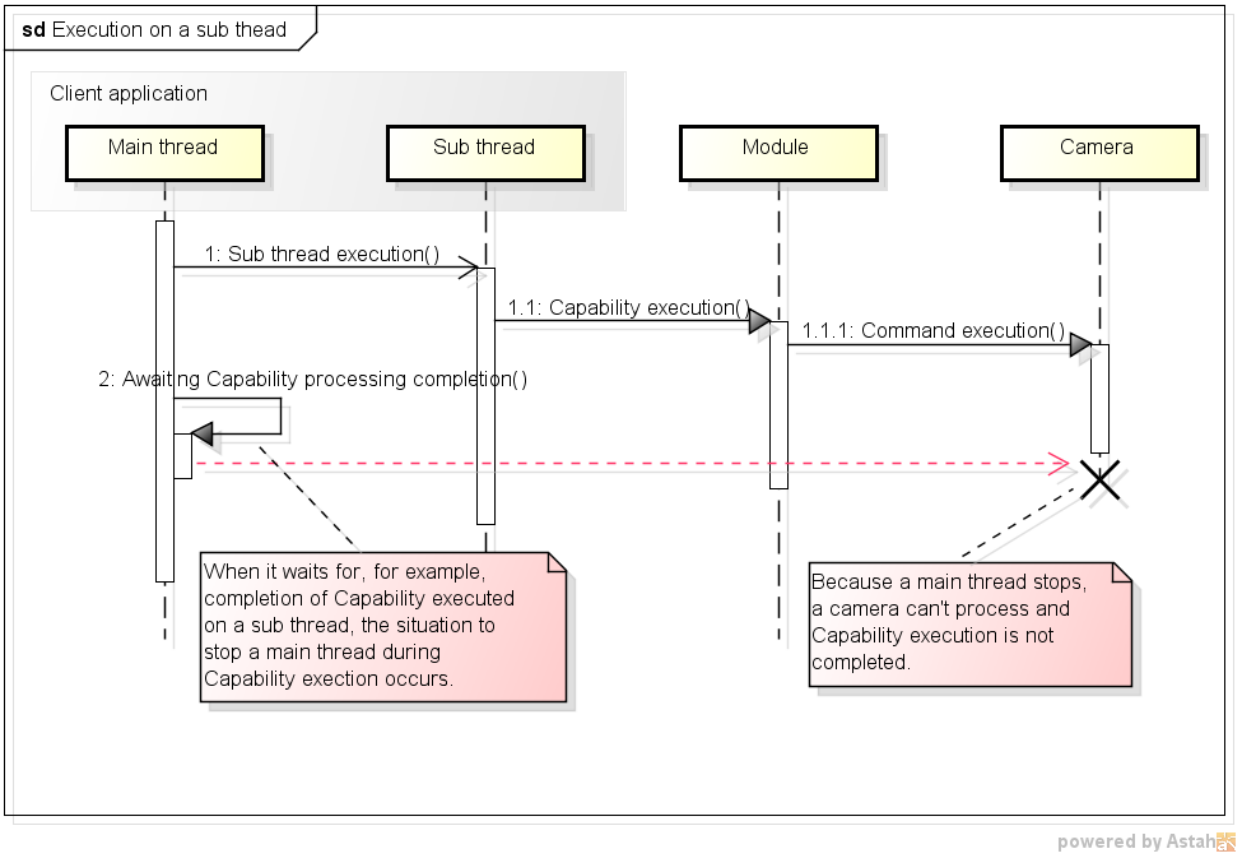
Example 1) When client application executes Capability from a main thread, Module may call CFRunLoopRunInMode(). Therefore the timer in the main thread may be called, for example, though control shifts to Module.

When the client executes other Capability at the timer processing, it may become the wait state. This is because processing of Capability which it executed from a main thread earlier is not completed. Because the timer processing is executed on a main thread, as a result, the main thread stops. Therefore the Capability that executed earlier can't receive the response from a camera and it reaches to a deadlock state.


sd Execution on a main thead
Client application
Main thead
Sub thread
Module
Camera
Module may call CFRunLoopRunInMode().
1: Capability execution()
1.1: CFRunLoopRunInMode()
1.1.1: Timer call()
1.2: Command execution()
1.1.1.1: Capability execution()
When a Timer is used on a main thread, it may be called.
Because it keeps waiting for completion of "1: Capability execution", a main thread stops. And when a main thread stops, the processing of "1: Capability execution" is not completed.
powered by Astah

Example 2) When client application may execute Capability from a sub thread and wait for the processing completion of Capability in main thread, client must not stop the main thread.

In this case, client has to call CFRunLoopRunInMode()at fixed intervals until the completion of Capability.



13 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.

14 History

- Rev2.1 June 14, 2017
  - 3. Environment…Update the environment of Macintosh.
  - 4. About the usage environment of Module SDK for Macintosh…Updated the recommended version of base SDK.
  - 11.1 Live view…Changed to refer to MAID3Type0016(E).pdf.

- Rev2.0 December 9, 2016
  - 2. Supported camera…Add D5600.
  - 3. Environment…Update the environment of Macintosh.
  - 4. About the usage environment of Module SDK for Macintosh…Added.
  - 11. The restriction about D5500, D5600…Add D5600.
  - 11.2. AF-F Shooting…Change the description.

- Rev1.3 March 31, 2016
  - Runtime Library…Added.

- Rev1.2 February 29, 2016
  - Environment…Update the environment of Windows.

- Rev1.1 November 13, 2015
  - Environment…Update the environment and restriction of Macintosh.
  - Connection with camera…Change the object which AddChild and RemoveChild event are sent to from source object to module object.
  - The restriction on Macintosh...Add the notice of using Module for Macintosh.

- Rev1.0 February 1, 2015 First version