Atik TE-47 Programmers Guide v1.1

Camera Specific Options

This section outlines the different camera specific options available to the TE-47.

The following features can be set using the Camera Specific Options:

- Readout Mode
- Gain / Offset

Note: The Artemis{Get/Set}Gain functions will not work with the TE-47.

There are two ADC's connected to the TE-47's sensor that we can read from:

- Low noise (ADC1)
- Fast (ADC2)

The ADC used is set via the Readout Mode option.

There are 3 options:

Name	ADC	Readout Speed
Lowest Noise	1	100KHz
Low Noise	1	200KHz
Preview	2	2MHz

Depending on the readout mode the gain and offset are set in slightly different ways. While using ADC1 we make use of ID_ADC1Offset(ushort) and for ADC2 ID_GOCustomGain(ushort 0-63) & ID_GOCustomOffset(ushort 0 - 255).

Camera Specific Options Functions

In order to check / update the settings, you will need the following functions (described below):

- `ArtemisHasCameraSpecificOption`
- `ArtemisCameraSpecificOptionGetData`
- `ArtemisCameraSpecificOptionSetData`

All functions require a handle to the camera (which is returned from the ArtemisConnect method) as well as an ID which defines the option you are using. The IDs are as follows:

- `ID_GOCustomGain   =   5`
- `ID_GOCustomOffset =   6`
- `ID_ReadoutModeTE  =  20`
- `ID_ADC1Offset     =  88`
- `ID_FX3Version     = 200`
- `ID_FPGAVersion    = 201`

ArtemisHasCameraSpecificOption

```
bool ArtemisHasCameraSpecificOption(ArtemisHandle handle, unsigned short
id);
```

This function is used to check whether the camera has the requested option.

The function will return false if:

- The option isn't available
- The camera is no longer available

ArtemisCameraSpecificOptionGetData

```
int ArtemisCameraSpecificOptionGetData(ArtemisHandle handle, unsigned
short id, unsigned char * data, int dataLength, int &actualLength);
```

This function is used to get the current value of the given option. The function will return 'ARTEMIS_INVALID_PARAM' if the option isn't available on the camera or the 'dataLength' provided is too small. Otherwise it will return 'ARTEMIS_OK'.

To call this method you will need to provide a buffer via the 'data' parameter to store the result of the function call. You will also need to pass the length of the buffer in bytes via the 'dataLength' param to ensure a buffer overflow does not occur.

These three settings require a 6 byte buffer:

- `ID_GOCustomGain   =  5`
- `ID_GOCustomOffset =  6`
- `ID_ReadoutModeTE  = 20`
- `ID_ADC1Offset     = 88`

The buffer will be populated with 3 unsigned shorts:

Byte	0-1	2-3	4-5
	Minimum Value	Maximum Value	Current Value

These settings require a 6 byte buffer:

- `ID_FX3Version  = 200`
- `ID_FPGAVersion = 201`

The buffer will be populated with 3 unsigned shorts:

Byte	0-1	2-3	4-5
	Year	Month	Date concatenated w/ build number that day

```
unsigned char data[6];
int actualLength;
ArtemisCameraSpecificOptionGetData(handle, ID_FPGAVersion, data, 6, &actualLength);
unsigned short year;
unsigned short month;
unsigned short dayPatch;
unsigned short* ptr = (unsigned short*)data;
year = ptr[0];
month = ptr[1];
dayPatch = ptr[2];
```
Figure 1 Example of getting the FPGA version

ArtemisCameraSpecificOptionSetData

```
int ArtemisCameraSpecificOptionSetData(ArtemisHandle handle, unsigned
short id, unsigned char * data, int dataLength);
```

This function is used to set the current value of a given option. The function will return 'ARTEMIS_INVALID_PARAM' if the option isn't available on the camera. Otherwise it will return 'ARTEMIS_OK'.

You will need to pass the value in via the 'data' param and the data length in bytes via the 'dataLength' param.

This setting requires 2 bytes(unsigned short) of data:

- `ID_ReadoutModeTE = 20`

Value	Exposure Speed	Gain/Offset in use
0	Lowest Noise	ID_ADC1Offset
1	Low Noise	ID_ADC1Offset
2	Preview	ID_ GOCustomGain/ ID_GOCustomOffset

```
unsigned short readoutMode = EXP_MODE_TE_LOWEST_NOISE;
ArtemisCameraSpecificOptionSetData(handle, ID_ReadoutModeTE, (char*)&readoutMode, 2);
```

These three settings require 2 bytes(unsigned short) of data:

- `ID_GOCustomGain   =  5`
- `ID_GOCustomOffset =  6`
- `ID_ADC1Offset     = 88`

The argument passed should be between the minimum and maximum value returned by ArtemisCameraSpecificOptionGetData. If the argument passed is outside of the min/max value, it will be clamped to a valid one.

ADC1 Offset

ID	Range
ID_ADC1Offset	0-4095

ADC2 Gain

ID	Range
ID_GOCustomGain	0-63

ADC2 Offset

ID	Range
ID_GOCustomOffset	0-255

Unused Functionality

As we use a C API which covers all of our cameras, many functions are not applicable to the TE-47. They are listed below:

- ArtemisSetSubSample
- All continuous exposure functionality
- ArtemisSetPreview
- ArtemisAutoAdjustBlackLevel
- ArtemisPrechargeMode
- ArtemisEightBitMode
- ArtemisGetEightBitMode
- ArtemisStartOverlappedExposure
- ArtemisOverlappedExposureValid
- ArtemisSetOverlappedExposureTime
- ArtemisGetProcessing
- ArtemisSetProcessing
- ArtemisClearVReg
- All fast exposure functionality
- ArtemisGetAmplifierSwitched
- ArtemisSetAmplifierSwitched
- All column repair functionality
- All eeprom functionality
- ArtemisGetGain
- ArtemisSetGain
- All GPIO functionality
- All guiding functionality
- All lens functionality
- All shutter functionality