Spectral Products

Spectrometers · Spectrophotometers · Color Instruments · Spectrographs · Monochromators

111 Highland Drive · Putnam, CT · 06260 · USA
PHONE (860) 928-5834· FAX (860) 928-2676
http://www.spectralproducts.com

SM32Pro SDK

Spectrometer Operating Software

USER MANUAL

(SM301/SM301EX)

Spectral Products

Spectrometers · Spectrophotometers · Color Instruments · Spectrographs · Monochromators

111 Highland Drive · Putnam, CT · 06260 · USA
PHONE (860) 928-5834· FAX (860) 928-2676
http://www.spectralproducts.com

Table of Contents

111 Highland Drive · Putnam, CT · 06260 · USA
PHONE (860) 928-5834· FAX (860) 928-2676
http://www.spectralproducts.com

111 Highland Drive · Putnam, CT · 06260 · USA
PHONE (860) 928-5834· FAX (860) 928-2676
http://www.spectralproducts.com

Warranty And Liability

This SM product is warranted against defects in material and workmanship for a period of one year from the date of shipment. During the warranty period, Spectral Products (SP) will, without charge, repair or replace, at its discretion, the defective product or component parts.

For warranty service or repair, this product must be returned to a service facility designated by SP. For products returned under warranty, the Buyer shall prepay shipping charges (including shipping charges, duties, and taxes for products returned to SP from another country), and SP will pay for shipping charges to return the product to the Buyer.

This warranty does not apply in the event of misuse or abuse of the product or as a result of unauthorized alterations, modifications, or repairs if the serial number is altered, defaced, or removed, the improper or inadequate maintenance by the Buyer, Buyer-supplied software or interfacing, operation outside of the environmental specifications for the product, or improper site preparation or maintenance. No other warranty is expressed or implied. SP shall not be liable for any consequential damages, including without limitation, damages resulting from loss of use, as permitted by law.

Spectral Products
Spectrometers · Spectrophotometers · Color Instruments · Spectrographs · Monochromators
111 Highland Drive · Putnam, CT · 06260 · USA
PHONE (860) 928-5834· FAX (860) 928-2676
http://www.spectralproducts.com

Quick Start Installation Guide

This quick-start installation guide will walk you through the quick and easy installation of the SM software package you have chosen. Please follow the instructions closely to complete the installation of your product.

1. Insert the SM installation USB flash drive into the appropriate USB port.

2. When the autorun starts, cancel the installation.

3. Explore the contents of the USB flash drive and copy the "\SDKs" folder to any folder or directory on your local drive.

111 Highland Drive · Putnam, CT · 06260 · USA
PHONE (860) 928-5834· FAX (860) 928-2676
http://www.spectralproducts.com

System Requirements

Check that your computer meets the minimum requirements for the SM32Pro system.

Requirements for the Software

- Any IBM compatible computer
- A hard drive with at least 50 MB of free space
- USB ports
- A VGA or compatible display
- 32 MB RMA or higher
- A mouse or other pointing device
- Microsoft Windows® 98, 98SE, ME, NT, 2000, XP, 7, 8.1, 10, and 11

Check System Package Contents

Check that your SM system package contains all of the required components.

Common system packages contain the following:

- Spectrometer
- Cable and Adapter
- Any accessories ordered

Note: Package contents may vary from unit to unit and order to order. If you have any questions about the contents of your package, please contact the support team referred to in the back of this manual.

Spectral Products

Spectrometers · Spectrophotometers · Color Instruments · Spectrographs · Monochromators

111 Highland Drive · Putnam, CT · 06260 · USA
PHONE (860) 928-5834· FAX (860) 928-2676
http://www.spectralproducts.com

Getting Started

Welcome to the Spectral Products **SM32Pro Software Development Kit**.

This kit documents a library of functions for accessing the USB Data Acquisition board used by the PbS/PbSe detector units, SM301 and SM301EX. Two fundamental detector imaging concepts are the Collection of Data and the Generation Meaningful Values from that data.

Collection of Data

In our PbS/PbSe detector spectrometer, the light is dispersed across the PbS/PbSe 256 pixel detector array. Data is collected by each pixel and converted to a relative value by the analog-to-digital (A/D) converter ranging from 0 to 65535 (16bit) and representing the intensity of the light at each pixel. We can control the amount of time that the pixels collect light and thus read signals of varying strengths. Using this library of functions you will be able to adjust the signal capture time (Integration Time) and collect the converted signals from each pixel. The criteria to adjust the integration is to try different lengths of time with a reference signal representing the maximum possible signal level during the measurement until the peak value across the range of pixels is close to, but not at 65535. We recommend trying to get a peak of about 65000. This way you can be sure that you are getting an optimized measurement condition with no saturation of PbS/PbSe detector elements.

Generating Meaningful Values

The information you collect from the system is only the relative signal size at each detector pixel until it has been calibrated to absolute (certified reference) values. That is in the raw data, the pixel location represents the spatial distribution of the PbS/PbSe detector elements and the corresponding intensity is represented by a digital value. Since we can adjust the strength of the light source and the integration time of the PbS/PbSe detector, this A/D value only provides a relative representation of the real world parameters.

We address both of these problems. We connect pixels to wavelengths by use of known emission lines (e.g. SP calibration light sources or narrow band filters) across the range of pixels together with a table relating pixels to

111 Highland Drive · Putnam, CT · 06260 · USA
PHONE (860) 928-5834· FAX (860) 928-2676
http://www.spectralproducts.com

wavelengths in nanometers. Curve fitting functions can then be applied to generate a polynomial function for the conversion of all other pixels to wavelengths.

We address the second problem in one of two ways. Both ways involve "normalization." Normalization involves measurements of signal strength based on its ratio with respect to reference signal intensity. If we are only interested in the spectral distribution of the sample signal, then we can normalize a sample signal scan to one reference value by, for example, intensities of all the array elements divided by the peak intensity. If we want to measure percent transmission, the light can be measured first with only air in the light path as 100% reference, and then the sample can be inserted into the light path and a sample scan followed. Consequently, the divisions of sample scan readings by the 100% reference yield the relative transmission values, or percent transmission when multiplied by 100%. For reflectance measurement similar practice can be applied and a high reflector can be used as a reference air in many cases.

Balancing: It is extremely important to do balancing before measuring real data in the case of PbS/PbSe detector spectrometers. PbS/PbSe detectors have piezo-electric properties, so their dark signal keeps changing when the power is applied. Also, the detectors can be affected a lot by ambient thermal noises. It is strongly recommended that real measurements be taken after this balancing, in which no external light but only the detector noise will be sensed and subsequently balanced for all the pixels. This will minimize the thermally generated "background" DC offset level and statistically establish a more stable baseline for all subsequent measurements.

111 Highland Drive · Putnam, CT · 06260 · USA
PHONE (860) 928-5834· FAX (860) 928-2676
http://www.spectralproducts.com

Using the SP Libraries

For Visual C/C++, you must include the correct "include" file and "lib" file for the functions you are calling.

For Labview, please include the correct "vi" files.

All the files are located in the "\SDKs" sub folder per program language in the USB flash drive that SP provides along with the unit.

Library files:

ArrayBoardR6_Lib.dll
ArrayBoardR6_Lib.tlb
SMdbUSBABR6.dll
SMdbUSBABR6.lib

To call the SDK functions, the library files should be installed in the development PC. Usually, by installing the SM32Pro For SM301 vR6, all the library files are installed in the Windows system folder automatically. If the user wants to install the ArrayBopardR6_Lib.tlb manually, go to the USB flash drive, copy the "\SDKs\Com" folder and paste it into the PC, open the command prompt, go to the folder where the "\SDKs\Com" folder was paste, and run the following command.

regasm.exe ArrayBoardR6_Lib.dll /tlb:ArrayBoardR6_lib.tlb

Or you can just go to the "SDKs\Com" folder in the USB flash drive, and register the ArrayBoardR6_Lib.reg file as an administrator.

The SPdbUSBABR6.dll and the ArrayBoardR6_Lib.dll files have to be in the same folder. Depending on the development environment/language, the user needs to include the SPdbUSBABR6.lib and the SPdbUSBABR6.dll files.

Please contact us if you need any further technical assistance.

111 Highland Drive · Putnam, CT · 06260 · USA
PHONE (860) 928-5834· FAX (860) 928-2676
http://www.spectralproducts.com

General Overview

Some fundamentals of utilizing the SP SDK functions

All SP's DLL functions are based on the ArrayBoardR6_Lib.dll that the manufacturer provided. To get more detailed information about all DLL functions in the ArrayBoardR6_Lib.dll, refer to the manual (ArrayBoardR6 API Reference.pdf) for it.

First spTestBoard() should be called to get the appropriate information of the board. You can trace some basic setting values from this function. To turn on or off the TE cooler, run spSetTECoolerEx(). With spReadStatusTE(), you can check the current status of the TE Cooler. The spSetIntTimeEx() is to set the integration time. You can set it from ~4usec to ~210msec. The longer the integration time is, the higher the baseline becomes and the lower the dynamic range you can obtain. Depending on the variable settings, the practical maximum integration time of much lower than the given maximum will exist.

Integration time is the period the detector pixels are exposed to light before the resulting charges are read out. A longer integration time can allow you to detect a lower light-level signal. The longer your integration time is the more background signal will accumulate. Please note that the vendor recommends using as low integration time as possible.

The PbS/PbSe detector must be balanced before getting the signals from the detector. spBalanceEx() is for balancing the detector pixels. Please be sure that this spBalanceEx() is for a stable baseline. This function should be called when the TE cooler is stabilized completely. You can check the TE cooler stability with the spReadStatusTE() function. The input light should be turned off or blocked completely before running this function. Reading the signals from the detector is carried out by spReadDataEx().

You can store all detector array operational data and settings currently resident in the PIC processor's run-time memory to the PIC processor's NVRAM for future retrieval to a known operating state after a mux controller board power cycle by using the spStoreAllEx(). The spCloseEx() needs to be called before closing the main program.

Spectral Products

Spectrometers · Spectrophotometers · Color Instruments · Spectrographs · Monochromators

111 Highland Drive · Putnam, CT · 06260 · USA
PHONE (860) 928-5834· FAX (860) 928-2676
http://www.spectralproducts.com

Using Curve Fitting to Calibrate SM32Pro

Curve fitting is used in SM32Pro to correlate the physical locations of pixels on the detector with the known wavelength of the radiation falling on them.

This is done by identifying the pixel locations where the maximal of the known wavelength is at. These peak intensity wavelengths and pixels are used by spPolyCalc() and spPolyFit() to generate a correlating polynomial function that best represents all the data points. We have found that using a third-order polynomial function produced the most desirable results for most cases. In cases where high dispersion elements are used, lower-order polynomial functions may have to be utilized due to the limited known wavelengths available from the calibration lamps.

Calibration Files

Each unit's calibration set is included in the SM32Pro software settings. This text file contains calibration data of the form "DataX=Wavelength;Pixel". The file also contains the regression coefficients "A0 value", "A1 value", … , "B3 value" that satisfy the equations.

$$l_i = A_0 + A_1 P_i + A_2 P_i^2 + A_3 P_i^3$$

$$P_i = B_0 + B_1 l_i + B_2 l_i^2 + B_3 l_i^3 .$$

The A values are the coefficients for conversions from a pixel number to a wavelength in "nm" by use of the above first third order polynomial function. The B values allow the conversions from a desired wavelength in "nm" to a pixel number.

Spectral Products

Spectrometers · Spectrophotometers · Color Instruments · Spectrographs · Monochromators

111 Highland Drive · Putnam, CT · 06260 · USA
PHONE (860) 928-5834· FAX (860) 928-2676
http://www.spectralproducts.com

SDK Functions

Data Acquisition

spTestBoard

Used by the program to test the acquisition board.

spSetTECoolerEx

Used by the program to turn on or off the TE cooler.

spReadStatusTE

Used by the program to read the current status of the TE cooler.

spSetIntTimeEx

Used by the program to set integration time.

spBalanceEx

Used by the program to balance the PbS/PbSe detector.

spReadDataEx

Used by the program to collect spectral data.

spStoreAllEx

Used by the program to store all operational data and settings.

spCloseEx

Used by the program to reset all board options upon resetting the software.

Calibration

spPolyFit

This function calculates the coefficients for a polynomial curve fitting function given an array of independent variables and a corresponding array of dependent variables.

Used by the program to generate a calibration function from pixels to wavelength.

spPolyCalc

111 Highland Drive · Putnam, CT · 06260 · USA
PHONE (860) 928-5834· FAX (860) 928-2676
http://www.spectralproducts.com

This function calculates a polynomial function given the independent variable and a coefficient array.

Used for determining the wavelength for a given pixel location.

Procedure of getting data

spTestBoard
Test the detector board to see if it is working properly.

spSetTECoolerEx
Turn TE cooler on/off.

spSetIntTimeEx
Set integration time.

spBalanceEx
Balance the PbS/PbSe detector.

spReadDataEx
Read acquired data.

spStoreAllEx
Store all operational data and settings.

spCloseEx
Reset all the values to default values (should be called at the exiting of program).

111 Highland Drive · Putnam, CT · 06260 · USA
PHONE (860) 928-5834· FAX (860) 928-2676
http://www.spectralproducts.com

Alphabetical Function Reference

spTestBoard

long spTestBoard
(
)

This function is used to locate the connected Array Board. Up to 8 boards can be connected.

RETURN:
If the board works successfully the function will return the number of connected boards.
If the board doesn't work properly the function will return a negative number and error message.

---

spSetTECoolerEx

long spSetTECoolerEx
(
long const lChannel;, // The channel to control
long const lTECtemp; // TE Cooler Temperature in Celsius
)

This function is used to set the TE cooler temperature.
lChannel informs the number of the channel (connected unit) to control.
lTECtemp is the target temperature for the TE cooler to cool down. It is from -21.1 deg C to +28.5 deg C. Depending on the ambient temperature that the spectrometer is running, it needs to be adjusted. The default value is -10.

RETURN:
If there is any error exists, it returns a negative number.

---

111 Highland Drive · Putnam, CT · 06260 · USA
PHONE (860) 928-5834· FAX (860) 928-2676
http://www.spectralproducts.com

spReadStatusTE:

```
long spReadTemp
(
long lChannel;,      // The channel to control
long lNumAve;,       //
long *lStatusTE;,    // TE cooler status
doule *dTeVal;       // TE cooler feedback
)
```

This function is used to read the status of the TE cooler. When this function is done successfully, each value will be saved on each parameter.

lChannel informs the number of the channel (connected unit) to control.
lStatusTE is required to have at least 2 X (long) memory and **dTeVal** 3 X (double) memory.
lStatusTE[0] is the TE cooler on/off status. It will have "1" when the TE cooler power is ON and "0" when OFF.
lStatusTE[1] indicates the status of the TE cooler stability. It will have "1" when the TE cooler gets stabilized and "0" otherwise.
dTeVal[0] is the current in Amps that the TE cooler is currently consuming.
dTeVal[1] is the current temperature offset in milli-degree Celsius (or Kelvin) from the target temperature of the TE cooler. It provides an approximate look at how far from the TE cooler set temperature the loop is controlling to.
dTeVal[2] is the voltage in Volts that the TE cooler is currently consuming. By multiplying the **dTeVal[0]** and **dTeVal[2]**, the current consuming power of the TE cooler can be calculated.

RETURN:
If there is no problem when running, it returns 1. Otherwise, it returns other values and an error message.

111 Highland Drive · Putnam, CT · 06260 · USA
PHONE (860) 928-5834· FAX (860) 928-2676
http://www.spectralproducts.com

spSetIntTimeEx:

long spSetIntTimeEx
(
long const lIntegTime; **//new integration time (1-65535)**
)

This function is used to change the integration time.

lIntegTime is the integration time. The real integration time to be set in the detector board is {3.2 X ("lIntegTime" - 1) + 4.025} microseconds. This number should range between 1 and 65535. The minimum integration time possible is 4.025usec (lIntegTime = 1) and the maximum is ~210msec (lIntegTime = 65535). To set 1msec, for example, set the lIntegTime as "312".

RETURN:

If there is no problem when running, it returns 1. Otherwise, it returns other values and an error message.

---

spBalanceEx:

long spBalanceEx
(
long const lWellIndex; **// Charge well size index (0-3)**
)

This function is used to balance the detector. The input light should be turned off or blocked completely before running this function.

lWellIndex is the index of the charge well size saved in the NVRAM of the detector board. It indicates the index of the assigned charge well size. Valid charge well sizes are 1pF (index: 0), 4pF (index: 1), 7pF (index: 2), 10pF (index: 3), 11pF (index: 4), 14pF (index: 5), 17pF (index: 6) and 20pF (index: 7).

RETURN:

If there is no problem when running, it returns 1. Otherwise, it returns other values and an error message.

---

111 Highland Drive · Putnam, CT · 06260 · USA
PHONE (860) 928-5834· FAX (860) 928-2676
http://www.spectralproducts.com

spReadDataEx

long spReadDataEx
(
long const lBit; // A/D conversion resolution. 16bit
long lDataDir; // The direction to read data
long *plPixArray; // The array in which spectral data is stored
)

lBit is the A/D conversion resolution bit. The current ADIC board has 16bit.

lDataDir indicates the direction of data reading. Please set this value as “0” for reading data from the 1st pixel to the last pixel and “1” for reading from the last pixel to the 1st pixel. Depending on the location of the detector in the optical bench, the physical first pixel (1st pixel on the array) and the scanning first pixel (the shortest wavelength) could be opposite. In our SM301/SM301-EX housing, due to the USB connector location, the detector is placed oppositely so the default value is “1”.

lpPixArray points to a read memory address with the long integers of the same pixel number as that of the detector. The detector is usually installed inversely in the SM301 spectrometer so the memory location has to be also reversed.

Get detector pixel array data. This function uses the spPerPixelUnifmCorrection function to perform baseline correction.

When is activated, data is output after baseline correction. With the spPerPixelUnifmCorrection function

In case of baseline correction, gain = 1.5 and offset are fixed at 7000, so in case of directly setting and outputting data with different gain & offset, use the spReadDataEx_Raw or spReadDataEx_Adv function. Baseline correction calculation is as follows.

(RawData – DarkData) * gain + offset

* DarkData is measured when spPerPixelUnifmCorrection function or spBalanceEx function is called.

RETURN:

If there is no problem when running, it returns 1. Otherwise, it returns other values and an error message.

111 Highland Drive · Putnam, CT · 06260 · USA
PHONE (860) 928-5834· FAX (860) 928-2676
http://www.spectralproducts.com

spStoreAllEx:

long spStoreAllEx
(
)

This function is used to store all mux/detector array operational data & settings currently resident in the PIC processor's run-time memory to the PIC processor's NVRAM for future retrieval to a known operating state after a mux controller board power cycle.

RETURN:
If there is no problem when running, it returns 1. Otherwise, it returns other values and an error message.

---

spCloseEx:

short spCloseEx
(
)

This function is called to reset all default valued for spTestBoard() upon closing of the application

RETURN:
If there is no problem when running, it returns 1. Otherwise, it returns other values and an error message.

111 Highland Drive · Putnam, CT · 06260 · USA
PHONE (860) 928-5834· FAX (860) 928-2676
http://www.spectralproducts.com

spPolyFit:

```
long spPolyFit
(
float *x;            // Array of independent variables
float *y;            // Array of dependent variables
short numPts;        // Number of points in independent and dependent
arrays
float *coefs;                // Pointer to array to hold calculated coefficients
                     // [index from 0 to order]
short order;         //  Order of polynomial
)
```

This curve fitting function is used to find a polynomial function to calculate the wavelength of a given pixel.
This function is used for calibration purposes. Either a calibration light source or a series of narrow band filters are scanned and the pixel locations of all known peaks are identified along with the known wavelength at that peak. These peak locations and wavelengths are stored in the arrays x and y, respectively. The arrays indices should range from 0 to [Number_of_Points - 1]. They are passed to the function along with a requested order for the polynomial fitting function and an array large enough to hold the coefficients (). This array is then used with ***spPolyCalc*** to calculate wavelength from pixels.

x is an array containing the independent variables. It should range from 0 to (numPts-1).
y is an array containing the dependent variables. It should range from 0 to (numPts-1).
numPts is the number of points in the variable arrays
coefs is a pointer to the array that will contain the polynomial coefficients. It should range from 0 to (order-1).
order is the desired order of the polynomial. We have determined third order to be the optimum for wavelength calibration for most cases.

RETURN:
This function will return 1 if the function is successful. Otherwise, it will return negative.

111 Highland Drive · Putnam, CT · 06260 · USA
PHONE (860) 928-5834· FAX (860) 928-2676
http://www.spectralproducts.com

spPolyCalc

```
void spPolyCalc
(
float *coefs, // pointer to an array containing the polynomial coefficients
short order,  // the order of the polynomial equation
float x,      // the pixel number
float *y      // the value to be calculated
)
```

This function calculates for the following formula:

> y = *a0* + *a1*x^1* + *a2*x^2* + ... + *aN*x^N*, where * specified multiplication and ^ specifies "to the power of."

coefs is a pointer to an array containing the polynomial coefficients. These can be calculated using ***spPolyFit***.

order specified the order of the polynomial equation and must be less than or equal to the number of elements in coefs.

x is the independent variable, in this case, the pixel number.

y is the value to be calculated.

RETURN:

None

---

spSetIniVarEx

```
long spSetIniVarEx
(
long lChannel
)
```

Initialize the detector bias settings, bad pixel information, and setting values that were last saved to the board EEPROM with the spStoreAllEx function. This function must be used after the spTestBoard function. Especially, SM301-EX may not operate normally without calling this function.

> **lChannel**: channel number of Spectrometer to operate

RETURN:

In the case of normal operation, SP_NO_ERROR(1) is returned, otherwise, a negative number is returned.

111 Highland Drive · Putnam, CT · 06260 · USA
PHONE (860) 928-5834· FAX (860) 928-2676
http://www.spectralproducts.com

spGetWLTable

long spGetWLTable
(
long lChannel
double *dWLTable
)

This function retrieves the wavelength table of the connected device when simply importing the wavelength per pixel. Calibration data stored in EEPROM is read and polynomial calculation is performed internally in this function.

lChannel: channel number of Spectrometer to operate.
dWLTable : Output array of calculated wavelength values for each pixel

RETURN:

In the case of normal operation, SP_NO_ERROR(1) is returned, otherwise, a negative number is returned.

---

spDevInfo

long spDevInfo
(
long lChannel
char *strModel
char *strSerial
)

This function gets the model name and serial number of the connected device.

lChannel: Channel number of Spectrometer to operate
strModel: Model name output array
strSeria: serial number output array

RETURN:

In the case of normal operation, SP_NO_ERROR(1) is returned, otherwise a negative number is returned.

---

111 Highland Drive · Putnam, CT · 06260 · USA
PHONE (860) 928-5834· FAX (860) 928-2676
http://www.spectralproducts.com

spTECoolerPWR

long spSetTec
(
long lChannel
long long lTEPowerState
)

Set TECooler Power on/off.

lChannel : Channel number of Spectrometer to operate
lTEPowerState : 0 -> off, 1 -> on

RETURN:

In the case of normal operation, SP_NO_ERROR(1) is returned, otherwise a negative number is returned.

---

spSetWellDepth

long spSetWellDepth
(
long lChannel
long lWellDepth
)

Set the charge well depth size.

lChannel : Channel number of Spectrometer to operate
lWellDepth : Enter a value from 0 to 7

0 = Set 1pF charge well
1 = Set 4pF charge well
2 = Set 7pF charge well
3 = Set 10pF charge well
4 = Set 11pF charge well
5 = Set 14pF charge well
6 = Set 17pF charge well
7 = Set 20pF charge well

RETURN:

In the case of normal operation, SP_NO_ERROR(1) is returned, otherwise a negative number is returned.

---

111 Highland Drive · Putnam, CT · 06260 · USA
PHONE (860) 928-5834· FAX (860) 928-2676
http://www.spectralproducts.com

spTECoolerReadTECPT

long spTECoolerReadTECPT
(
long lChannel
long *lTECSetPoint
)

Call the value of the TEC Set Point set in the spSetTeCoolerEx function.

lChannel : Channel number of Spectrometer to operate
lTECSetPoint : It is received as a digital value between 0 and 255. This value must be converted to temperature by calculating with the following formula:

Variables / Coefficients:
1) TECSetpoint = 8 bit digital word loaded to digital pot, values range from 0 to 255 (inclusive)
2) TEPotRatio = TECSetpoint / 255, this is the ratio of the pot. This has values of 0 to 1 as is a double data type.
3) Thermistor Coefficient A = 0.0033538646, Data type is Double
4) Thermistor Coefficient B = 0.0002565409, Data type is Double
5) Thermistor Coefficient C = 0.0000019243889, Data type is Double
6) Thermistor Coefficient D = 0.00000010969244, Data type is Double
7) dtmp = temporary variable, data type is Double
dtmp = LN { 3 * [(5 * TEPotRatio + 2) / (7 - 5 * TEPotRatio)]}
Note: LN is natural logarithm
TempK = 1 / (A + B * dtmp + C * dtmp^2 + D * dtmp^3); in Kelvin
TempC = TempK - 273.15; in Celsius

RETURN:

In the case of normal operation, SP_NO_ERROR(1) is returned, otherwise a negative number is returned.

---

spSetShutterPos

long spSetShutterPos
(
long lChannel
long lShutterPos
)

Set the mechanical shutter position (Open/Close).
lChannel : Channel number of Spectrometer to operate
lShutterPos : 0 -> Close, 1 -> Open

RETURN:

In the case of normal operation, SP_NO_ERROR(1) is returned, otherwise a negative number is returned.

111 Highland Drive · Putnam, CT · 06260 · USA
PHONE (860) 928-5834· FAX (860) 928-2676
http://www.spectralproducts.com

spConfigTriggerDelay

long spConfigTriggerDelay
(
long lChannel
long lTrigDelayValue
long lTrigDelayMode
)

Set whether to use a timeout for the trigger input signal in the external trigger mode and also set the timeout value.

lChannel : Channel number of Spectrometer to operate
lTrigDelayValue : time out value. Digital value input from 0 to 65535
2.26us + (lTrigDelayValue - 1) * 0.2us, Maximum = 13.11ms
lTrigDelayMode : Whether to use time out
0 -> Disabled, 1 -> Enabled

RETURN:

In the case of normal operation, SP_NO_ERROR(1) is returned, otherwise a negative number is returned.

---

spConfigTrigger

long spConfigTriggerDelay
(
long lChannel
long lTriggerPolarity
)

Operation mode setting when using external trigger mode

lChannel : Channel number of Spectrometer to operate
lTriggerPolarity : Set falling edge and rising edge mode
0: falling edge, 1: rising edge

RETURN:

In the case of normal operation, SP_NO_ERROR(1) is returned, otherwise a negative number is returned.

111 Highland Drive · Putnam, CT · 06260 · USA
PHONE (860) 928-5834· FAX (860) 928-2676
http://www.spectralproducts.com

spSetExtTrgMode

long spSetExtTrgMode
(
long lChannel
long lExtTrgState
)

Set whether to use external trigger mode. While the external trigger mode is active, only spReadDataEx_ExtTrg functions can be used. If you need to use another data reading function, disable the external trigger mode using this function.

lChannel : Channel number of Spectrometer to operate
lExtTrgState : 0 -> Disabled, 1 -> Enabled

RETURN:

In the case of normal operation, SP_NO_ERROR(1) is returned, otherwise a negative number is returned.

---

spReadDataEx_ExtTrg

long spReadDataEx_ExtTrg
(
long lChannel
long *plPixelArray
)

Available only when external trigger mode is activated and the array data can be imported. SP_EXT_TRG_WAITING(-99) is returned after waiting for an external trigger signal input during the timeout value that was set in the spConfigTriggerDelay function.

lChannel : Channel number of Spectrometer to operate
plPixelArray : Output array data (total number of pixels = 256)

RETURN:

In the case of normal operation, SP_NO_ERROR(1) is returned, otherwise a negative number is returned.
Returns SP_EXT_TRG_WAITING(-99) when external trigger signal input time out

111 Highland Drive · Putnam, CT · 06260 · USA
PHONE (860) 928-5834· FAX (860) 928-2676
http://www.spectralproducts.com

spPerPixelUnifmCorrection

long spPerPixelUnifmCorrection
(
long lChannel
)

It is used for the baseline correction at the S/W level. Set the baseline to 7000 in the current measurement environment and set the gain of each pixel to 1.5 (i.e., 1.5 X the signal intensity at each pixel)

lChannel : Channel number of Spectrometer to operate

RETURN:

In the case of normal operation, SP_NO_ERROR(1) is returned, otherwise a negative number is returned.

---

spReadDataEx_Raw

long spReadDataEx_Raw
(
long lChannel
long *plPixelArray
)

Get detector pixel array data. Output uncorrected raw data.

lChannel : Channel number of Spectrometer to operate
plPixelArray : Output array data (total number of pixels = 256)

RETURN:

In the case of normal operation, SP_NO_ERROR(1) is returned, otherwise a negative number is returned.

111 Highland Drive · Putnam, CT · 06260 · USA
PHONE (860) 928-5834· FAX (860) 928-2676
http://www.spectralproducts.com

spReadDataEx_Adv

long spReadDataEx_Adv
(
long lChannel
double dGainValue
int iOffsetValue
long *plPixelArray
)

This function reads the output data after the baseline correction. Input dGainValue and apply iOffsetValue to perform baseline correction. The baseline correction is calculated as:

(RawData - DarkData) * dGainValue + iOffsetValue

* DarkData is measured when spPerPixelUnifmCorrection function or spBalanceEx function is called.

- **lChannel** : Channel number of Spectrometer to operate
- **dGainValue :** Enter the gain value to be applied during baseline correction.
- **iOffsetValue :** Enter an offset value to be applied during baseline correction.
- **plPixelArray** : Output array data (total number of pixels = 256)

RETURN:

In the case of normal operation, SP_NO_ERROR(1) is returned, otherwise a negative number is returned.