int ResetSSIMinAndMax(struct SIS1100\_Device\_Struct\* dev, unsigned short axis)

This function resets the value accumulated in SSI Max and SSI Min registers.

int ResetPhaseNoisePeak(struct SIS1100\_Device\_Struct\* dev, unsigned short axis)

This function resets the value accumulated in the Phase Noise Peak diagnostic register.

int ResetOptPowL2MinAndMax(struct SIS1100\_Device\_Struct\* dev, unsigned short axis)

This function resets the Opt Power L2 Min and Opt Power L2 Max registers. The *Cmd Busy bit* in Status Register 1 is set immediately and cleared when the registers have been reset.

int ResetSigRMSL2MinAndMax(struct SIS1100\_Device\_Struct\* dev, unsigned short axis)

This function resets the Sig RMS L2 Min and Sig RMS L2 Max registers. The *Cmd Busy bit* in Status Register 1 is set immediately and cleared when the registers have been reset.

**SetKpAndKvCoeff**

int SetKpAndKvCoeff(struct SIS1100\_Device\_Struct\* dev, unsigned short axis, unsigned short Kp, unsigned short Kv)

This function set Kp and Kv digital filter coefficients.

There’s a digital filter circuit on the Zygo Measurement Board 4104C designed to remove noise from the measurement data. It’s a second order digital filter, with two programmable gains, Kp and Kv, to permit a wide range of low pass filter cutoff frequencies, from 15 kHz to 2.7 MHz.

The Kv value for the digital filter is set to produce the best dynamic response for a given value of KP.

The Kp value for the digital filter primarily determines the bandwidth and response time of the digital filter.

Best Kp and Kv matched value can be found in the annexes.

**EnableGlitchFilter**

int EnableGlitchFilter(struct SIS1100\_Device\_Struct\* dev, unsigned short axis, unsigned short glitchFilterTime).

This function set the value of the glitch filter time ranged from 0 to 255.

The glitch filter controls suppression of glitches that would otherwise cause a loss of valid measurements. When **glitchFilterTime** is set to 0 there is a filter time constant of 1.5 µs; when set to 1 the time constant is 3.8 µs.

**GetVMEExtSampFlag**

bool GetVMEExtSampFlag(struct SIS1100\_Device\_Struct\* dev, unsigned short axis)

This function polls the state of the VME External Sample flag bit. This is useful to know when the position sampling has come to end.

Return:

* True if the flag is set
* False otherwise

**IsVMEIntReqPending**

bool IsVMEIntReqPending(struct SIS1100\_Device\_Struct\* dev, unsigned short axis)

This function check whether there’s an interrupt request pending on VME Bus. If the VME Interrupts is enable bit, the interrupt is asserted on the VMEbus.

Return:

* True if there’s
* False otherwise

**IsAPDCtrlSoftErrs**

bool IsAPDCtrlSoftErrs(struct SIS1100\_Device\_Struct\* dev, unsigned short axis)

This function check whether there’s a software error raise by the APD controller.

Return:

* True if there’s.
* False otherwise

**ReadAPDCtrlSoftErrs**

int ReadAPDCtrlSoftErrs(struct SIS1100\_Device\_Struct\* dev, unsigned short axis)

This function reads software errors raised by APD controller.

There are 4 types of software errors as listed bellow:

* Exception Error
* Stack Overflow
* Switch Default Error
* Comm. Error
* Divide by Zero Exception

All of the above errors are **Fatal Error** except for the Divide by Zero Exception and Comm Error. Fatal errors cause the APD Controller does the following:

* Shut down High Voltage Supply.
* Set fatal error on all axes.
* The APD Controller moves to a Fatal Error State.

ResetPosition

int ResetPosition(struct SIS1100\_Device\_Struct\* dev, unsigned short axis)

This function resets or presets the position measurement. It resets the measurement function, but does not reset errors. If the *Preset* isenabled(using EnableReset function), the value in the Preset Position register is preset into the position measurement. The reset or preset value is before the Offset register is subtracted. A detailed figure of position measurement can be found in the annexes.

EnablePreset

int EnablePreset(struct SIS1100\_Device\_Struct\* dev, unsigned short axis)

This function causes the Preset Position value to be taken into consideration in position measurement function. The Preset Position value minus the initial position value is added to the position value so that the first position value (PPO) is equal to the Preset Position value. After an axis reset, this function causes the position value to be initialized to the value in the 37-bit Preset Position register.

EnableSCLKResetOnAxisReset

int EnableSCLKResetOnAxisReset(struct SIS1100\_Device\_Struct\* dev, unsigned short axis).

This function enables SCLK0 or SCLK1 (as selected by the SCLKSelectOnAxisReset function) to perform a quick reset and a time reset on the first occurrence of SCLK after an axis reset.

Rmk: Make sure to select the clock signal to perform the reset( using SCLKSelectOnAxisReset function) before calling this function.

DisableSCLKResetOnAxisReset

int DisableSCLKResetOnAxisReset(struct SIS1100\_Device\_Struct\* dev, unsigned short axis)

This function disable quick reset and time reset on the first occurrence of SCLK after an axis reset.

SCLKSelectOnAxisReset

int SCLKSelectOnAxisReset

(struct SIS1100\_Device\_Struct\* dev, unsigned short axis, unsigned char SCLK)

This function determines which SCLK signal on the P2 bus causes the reset enabled by the SCLK Reset Enable function.

Rmk: MUST call EnableSCLKResetOnAxisReset after this function.

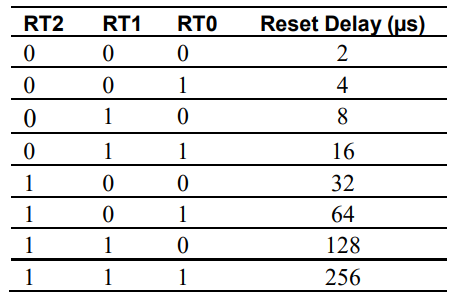
arguments:

* SCLK: the select clock to perform the said reset.
  + 0 for SCLK0
  + 1 for SCLK1

SetTimeDelayBetweenResAndCompBit

int SetTimeDelayBetweenResAndCompBit(struct SIS1100\_Device\_Struct\* dev, unsigned short axis, unsigned char SCLK)

This function determines the time delay between an axis reset or position reset command and the assertion of the Position Reset Complete. The delay choices are shown in Table 4-4. If the Enable Reset Finds Velocity bit is set, this is the delay after the velocity finding operation is finished.



EnableResetFindsVelocity

int EnableResetFindsVelocity(struct SIS1100\_Device\_Struct\* dev, unsigned short axis)

This function enables searching for the measurement signal frequency after an axis reset. This is required if the velocity when the axis is reset may be greater than 0.1 m/sec.

DisableResetFindsVelocity

int DisableResetFindsVelocity(struct SIS1100\_Device\_Struct\* dev, unsigned short axis)

This function disables searching for the measurement signal frequency after an axis reset.