Optical sensors with Arduino

February 6, 2019

Contents

1 ADNS2610

1.1 `namespace ADNS2610` . . . . .

1.1.1 `class Sensor` . . . . .

1 ADNS2610

1.1 `namespace ADNS2610`

All functional part is placed in a namespace called `ADNS2610`.

1.1.1 `class Sensor`

Optical sensor is represented by a class `Sensor`. Below is a description of all public members of this class.

`Sensor( uint8_t sckPin, uint8_t sdioPin )`

Description	constructor
Return	-
Arguments	`sckPin` - Arduino digital pin responsible for clock pulses `sdioPin` - Arduino digital pin to communicate with sensor

`bool IsAwake() const`

Description	Check whether LED is awake or not
Return	Value of type `bool`.
Arguments	none

`int8_t GetDX() const`

Description	Get shift of the sensor along X-axis
Return	Value of type `int8_t`
Arguments	none

`int8_t GetDY() const`

Description	Get shift of the sensor along Y-axis
Return	Value of type `int8_t`
Arguments	none

`uint8_t GetSQUAL() const`

Description	Get surface quality. Small values correspond to smooth surfaces.
Return	Value of type `uint8_t`.
Arguments	none

`uint8_t GetMaxPixel() const`

Description	Get maximum pixel value.
Return	Value of type `uint8_t`.
Arguments	none

`uint8_t GetMinPixel() const`

Description	Get minimum pixel value.
Return	Value of type `uint8_t`.
Arguments	none

`uint8_t GetPixelAverage() const`

Description	Get average pixel value.
Return	Value of type `uint8_t`.
Arguments	none

`uint16_t GetImage( uint8_t* frame ) const`

Description	Get entire image.
Return	Value of type `uint16_t`. This value is the number of read pixels. NOTE: it could be less then 324.
Arguments	`frame` - pointer to `uint8_t` where data will be written to.

`void SetAwakeLED() const`

Description	Change LED mode to AWAKE. In this mode LED is always awake.
Return	none
Arguments	none

`void SetNormalLED() const`

Description	Change LED mode to NORMAL. In this mode LED falls asleep after 1 sec inactive.
Return	none
Arguments	none

For more information about the sensor please refer official documentation.