|  |
| --- |
| UKMARSBOT I2C Sensor Controller Library |

Table of Contents

[1.0 FUNCTION DESCRIPTIONS 2](#_Toc63955301)

The ISC has a supporting library to simplify the operation. It is hoped that this documentation describes enough detail that the ISC can be incorporated in a project. However, for more comprehensive detail, please refer to the ISC Datasheet.

Installing the Library.

The ISC library may be installed in the same way as any other Arduino Library. Download and unzip ISC folder, copy the folder into the ‘lib’ folder within the Arduino installation. Once Arduino software is restarted, the library will be available to include and an Examples folder is available with some example code with comments explaining functionality.

# FUNCTION DESCRIPTIONS

## Read

This function reads the value from the given address. The numVals may be 1 or 2, for the number of consecutive registers to read (ie. 2 when a 2-byte value is to be read).

Function Definition

int read(uint8\_t regAddr, uint16\_t numVals);

Example:

int timeTaken = Isc.read(SCANTIME, 1);//Read 1 byte at address SCANTIME

## Write

This function writes an 8 bit or 16bit value to the given address. The numVals may be 1 or 2, for the number of consecutive registers the value is to overwrite (ie. 2 when a 2-byte value is to be written).

Function Definition

void write(uint8\_t regAddr, uint16\_t data, uint8\_t numVals);

Example:

Isc.write(SENS0THRSH, 500, 2); //Write the value 500 across two bytes for the Sensor 0 Threshold

## Reset

This function returns the ISC to the reset state, allowing the setup to be started again

Function Definition

void reset(uint8\_t address);

Example:

Reset(0x50); //Reset ISC with I2C (7-bit) Address 0x50

## Begin

This function starts the ISC in run mode, confirming that setup is complete. It defines the sensorboard that is attached.

Function Definition

void begin(uint8\_t boardType);

Keywords:

basicLineSensor UKMARS Basic Line Sensor Board

basicWallSensor UKMARS Basic Wall follower Board

spLineSensor Line Sensor board developed by S. Pithouse

Example:

Isc.begin(1); //Start run mode, Basic Line sensor board is attached

## ScanTime

This function returns the most recent time taken for the ISC to complete a full cycle. The units are µs.

Function Definition

int scanTime();

Example:

int myTime = isc.scanTime(); //Store most recent scan time in myTime

## SetThreshold

This function allows the sensor threshold to be set for a specific sensor.

Function Definition

void setThreshold(uint8\_t sens, uint16\_t threshVal);

Example

isc.setThreshold(SENSOR0, 500);

## Sensor

This function returns the most recent sensor value for the specified sensor.

Function Definition

int sensor(uint8\_t sensor);

Example

leftSensor = isc.sensor(SENSOR0);

## LED Mode

This function controls the Indicator LED function

Function Definition:

void ledMode(uint8\_t mode);

Keywords:

MASTERCTRL MCU has control

FREQOUT Scan frequency output on LEDs

LEDOFF Turn LEDs off

LEDBRDCTRL Allow board specific LED control

FASTBLINK Start LEDS blinking fast

SLOWBLINK Start LEDs blinking slowly

Example:

Isc.ledMode(2);

## Setup

This function allows individual sensor setup to be configured. There are keywords that may be combined with ‘+’ as shown.

Function Definition

void setup(uint8\_t sensor, uint8\_t bits);

Keywords

Enb Enable sensor – ISC will read the value

lowRes Reduce value to 8-bit

txEnb Enable Transmitter to help remove ambient light

invert [See Polarity]

fallingInterrupt Sensor will cause interrupt on 1->0 state change

risingInterrupt Sensor will cause interrupt on 0->1 state change

Example:

Isc.setup(SENSOR0, enb+txEnb+fallingInterrupt);

## txPulse

This function allows the time in µs that the Tx emitter is on before the first sample is taken to be adjusted.

Function Definition

void txPulse(uint8\_t length);

Example

Isc.txPulse(100);

## States

This function gets the current states of the sensors and returns a byte with 1 representing HIGH sensor state.

Function Definition

uint8\_t states();

Example

leftState = isc.states() & 0b1;