API User Manual for the VL6180 proximity sensor

Introduction

The VL6180 is a proximity sensor based on ST's patented FlightSense technology. The VL6180 interfaces to your micro-controller via the industry standard I2C bus.

This user manual describes:

- List of available VL6180 evaluation and development tools
- How to download the STSW-IMG011 API
- How to control the device through the API:
  - Ranging feature
  - Ranging with multiple VL6180 sensors

1 Document references

Table 1. **Document references**

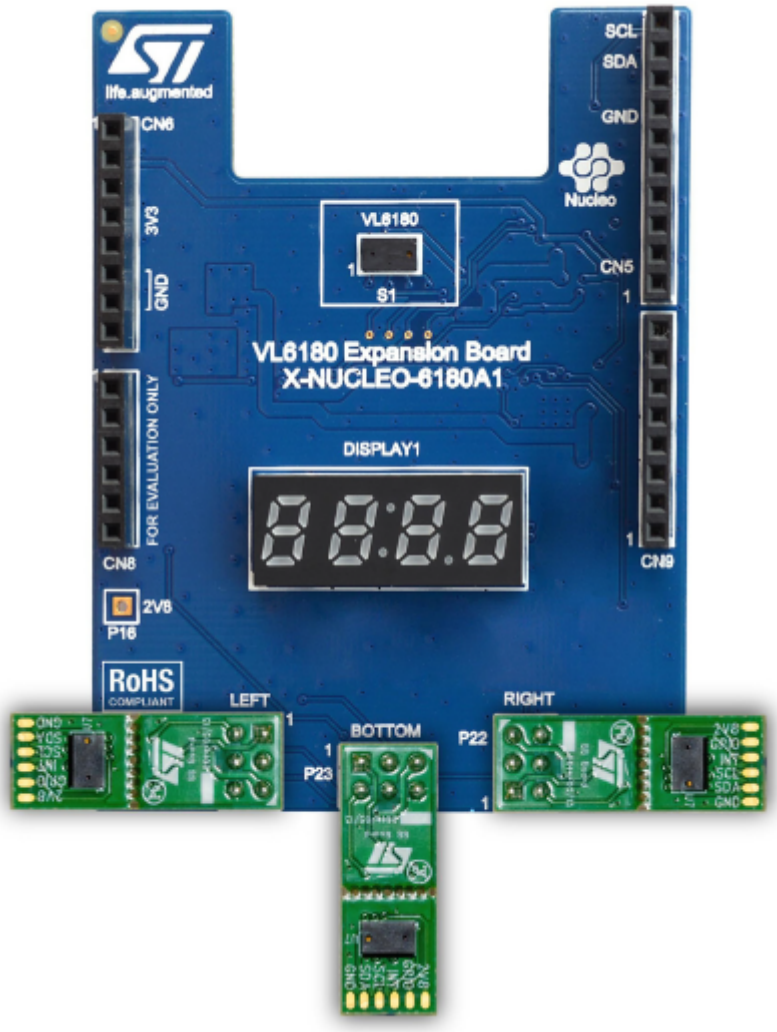
Description	DocID
Datasheet - VL6180 proximity sensing module	DS9818

2 Evaluation and development tools

Search st.com and download the software and documentation related to:

- X-NUCLEO-6180A1: VL6180 expansion board compatible with the STM32 Nucleo board family and with the Arduino™ UNO R3 connector layout.
- P-NUCLEO-61801A1: evaluation pack composed of a X-NUCLEO-6180A1 expansion board and a powerful STM32F401RE Nucleo board.
- VL6180-SATEL: Hardware add-on to X-NUCLEO-6180A1 expansion board to enable multi VL6180 sensor applications

Figure 1. VL6180 evaluation and development tools



Information on the installation of the different softwares allowing the evaluation and the application development around VL6180 are given in the user manual X-CUBE-6180A1_UserManual, which is zipped in the X-CUBE-6180A1 software package.

3 Application program interface (API) overview

This chapter details how to download and install the API (STSW-IMG011).

The VL6180 API is a set of C functions controlling the VL6180 (init, ranging,…) which enable the development of end-user applications. This API is structured so that it can be compiled on any platform through a well isolated platform layer (mainly for low level I2C access).

Ranging examples are provided to show how to use the API and perform ranging measurements.

A complete Nucleo F401 + VL6180 expansion board project is also provided (STM32Cube IDE required to compile the project) as well as the pre-compiled binary that can be directly used.

3.1 API download

To download the VL6180 API (STSW-IMG011):

1. On www.st.com, under Products, search for “VL6180”.
2. Under the Tools and Software tab select “STSW-IMG011”.
3. Click on “Get Software”.

3.2 Quick start guide for API integration

API documentation is in the “docs” folder and is available in two formats:

- API_Documentation.chm
- API_Documentation.html

The VL6180 API is integrated in a software project in two steps.

1. The developer adds/links the files listed in Table 2. API header files and in Figure 2. IHeader and CCI service files in the API to the source and includes code path. Some files may require modification to comply with the final application or the hardware/software capabilities.

Table 2. API header files

Names	Description	Modification
vl6180_cfg.h	Application configuration.	May require modification
vl6180_api.c and vl6180_api.h	All operating functions at high and low level to control the sensor	Must not be modified
vl6180_def.h	Definition of constants and structures used in the API	Must not be modified
vl6180_platform.h	Target platform specific declarations/ prototypes	May require modification
vl6180_types.h	Basic types definition	May require porting

2. To manage the data communication between the VL6180 and the host, the developer has to design a camera control interface (CCI) register communication driver.

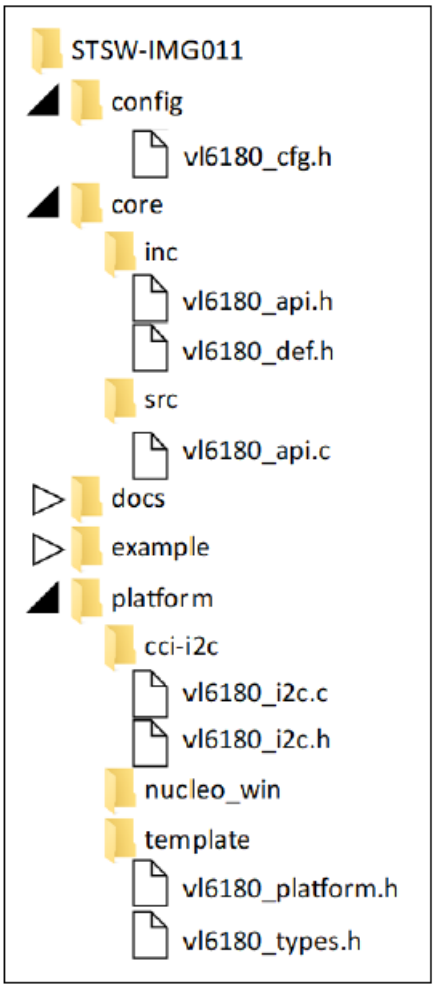
The API low-level functions rely on the following set of 7 read and write functions which perform CCI register access to the device:

VL6180_WrByte(); VL6180_WrWord(); VL6180_WrDWord(); VL6180_UpdateByte(); VL6180_RdByte(); VL6180_RdWord(); VL6180_RdDWord().

To implement these 7 functions, it is recommended to use vl6180_i2c.c and vl6180_i2c.h files in platform/cci_i2c directory (see Figure 2. IHeader and CCI service files in the API)

Note: Detailed information on these functions can be found in section Modules/CCI to RAW I2C translation layer of the API_Documentation_(version)_proximity.chm delivery

Figure 2. IHeader and CCI service files in the API



4 Main features definition

This chapter defines the main features of the VL6180.

Ranging

Measurement of the distance between VL6180 and the target.

Upscale support - Extended range

- Upscale factor = 1, VL6180 measures distances up to 20 cm with a granularity of 1 mm.
- Upscale factor = 2, VL6180 measures distances up to 40 cm with a granularity of 2 mm.
- Upscale factor = 3, VL6180 measures distances up to 60 cm with a granularity of 3 mm.

Wrap around filter (WAF)

In specific conditions, when targeting a mirror or very reflective metal, a wrap around effect can occur internally to the VL6180 which results in a wrong distance being returned (under estimated). The goal of the wrap around filter is to detect this wrap around effect and to filter it by returning a non-valid distance.

Dmax

Estimation of the maximum distance (in mm) up to which the VL6180 will report a valid measurement with a 17% grey target for the current ambient light conditions.

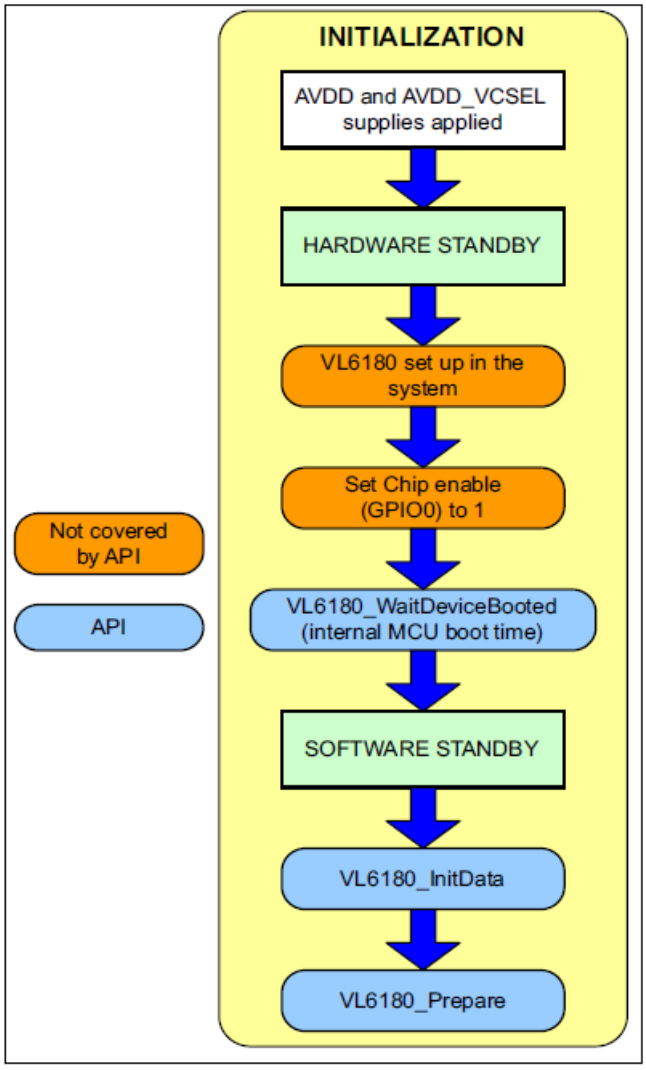
Dmax decreases when ambient light increases.

5 Ranging with VL6180

5.1 VL6180 initialization

The following figure shows the initialization of the VL6180 in the system.

Figure 3. VL6180 initialization

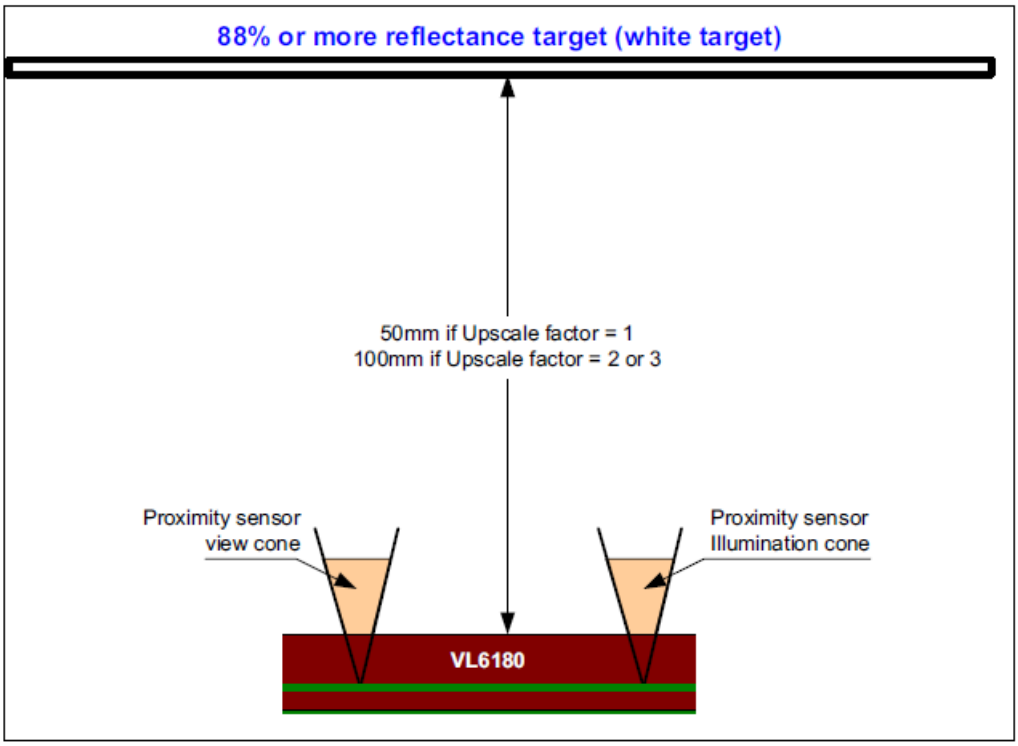


- VL6180 set up in the system
  - The user has to set all variables linked to the integration of the VL6180 in its application (e.g.: VL6180 I2C 7 bit default address = 0x29)
- VL6180_InitData (One time device initialization)
  - Enables common functions, ie, WAF, ECE, upscaling.
  - Expects device to be fresh out of reset.
- VL6180_Prepare (Prepare device for operation)
  - Programs common default settings.
  - Prepares the VL6180 for new ranging measurement.

5.2 Offset calibration

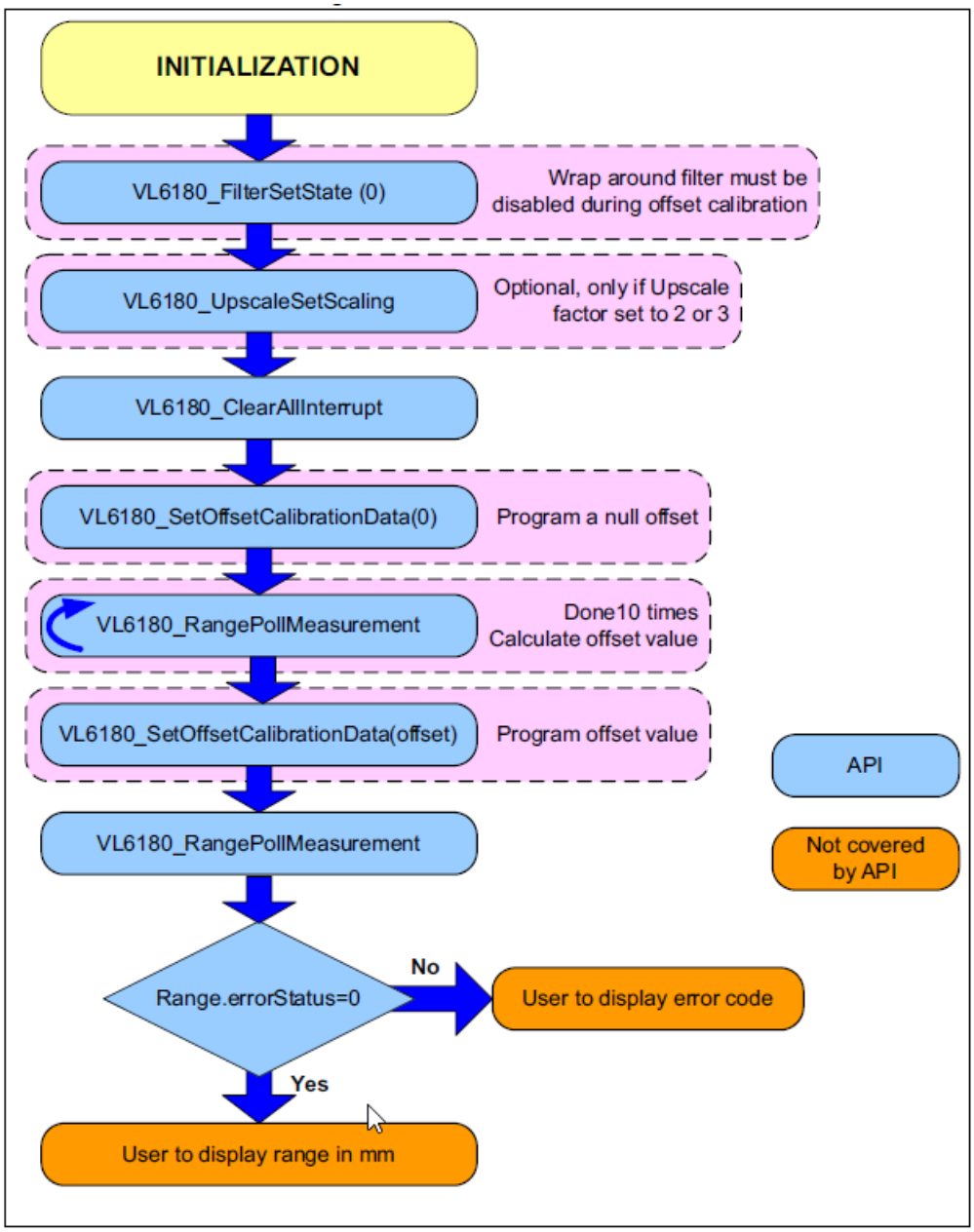
Place a target with a reflectance of 88% or higher (white target) 50 mm away from VL6180 if Upscale factor is 1, or 100 mm away if Upscale factor is 2 or 3.

Figure 4. Offset calibration environment



Then follow the flow described in the following figure.

Figure 5. **Offset calibration flow**



Note: Wrap around filter is disabled during offset calibration. User has to re-enable the filter after offset calibration is completed by calling VL6180_FilterSetState (1)

5.3 Cross talk calibration

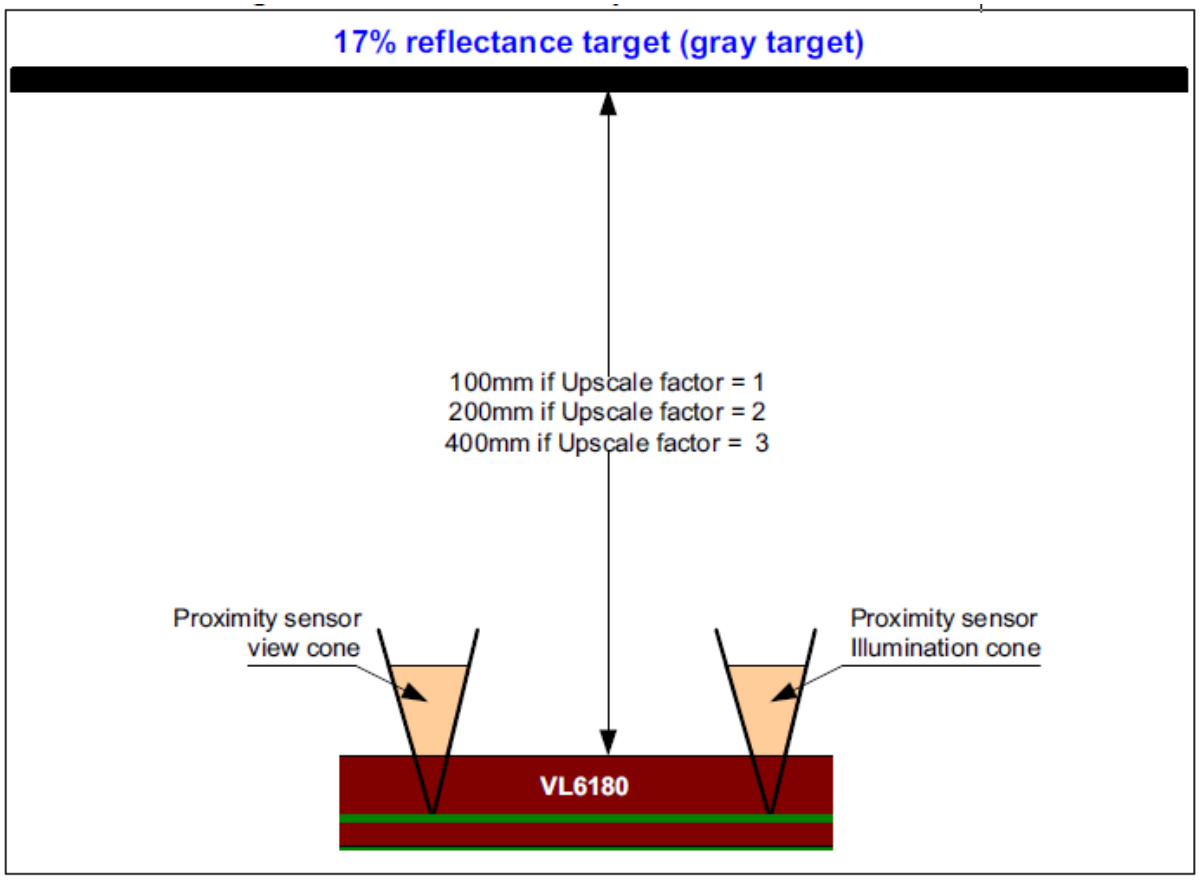
Cross talk calibration must be run after offset calibration.

If the offset is incorrectly calibrated, cross talk compensation will be inaccurate.

Place a target with a low reflectance e.g. 17% reflectance target (gray target) at the following distance from the VL6180:

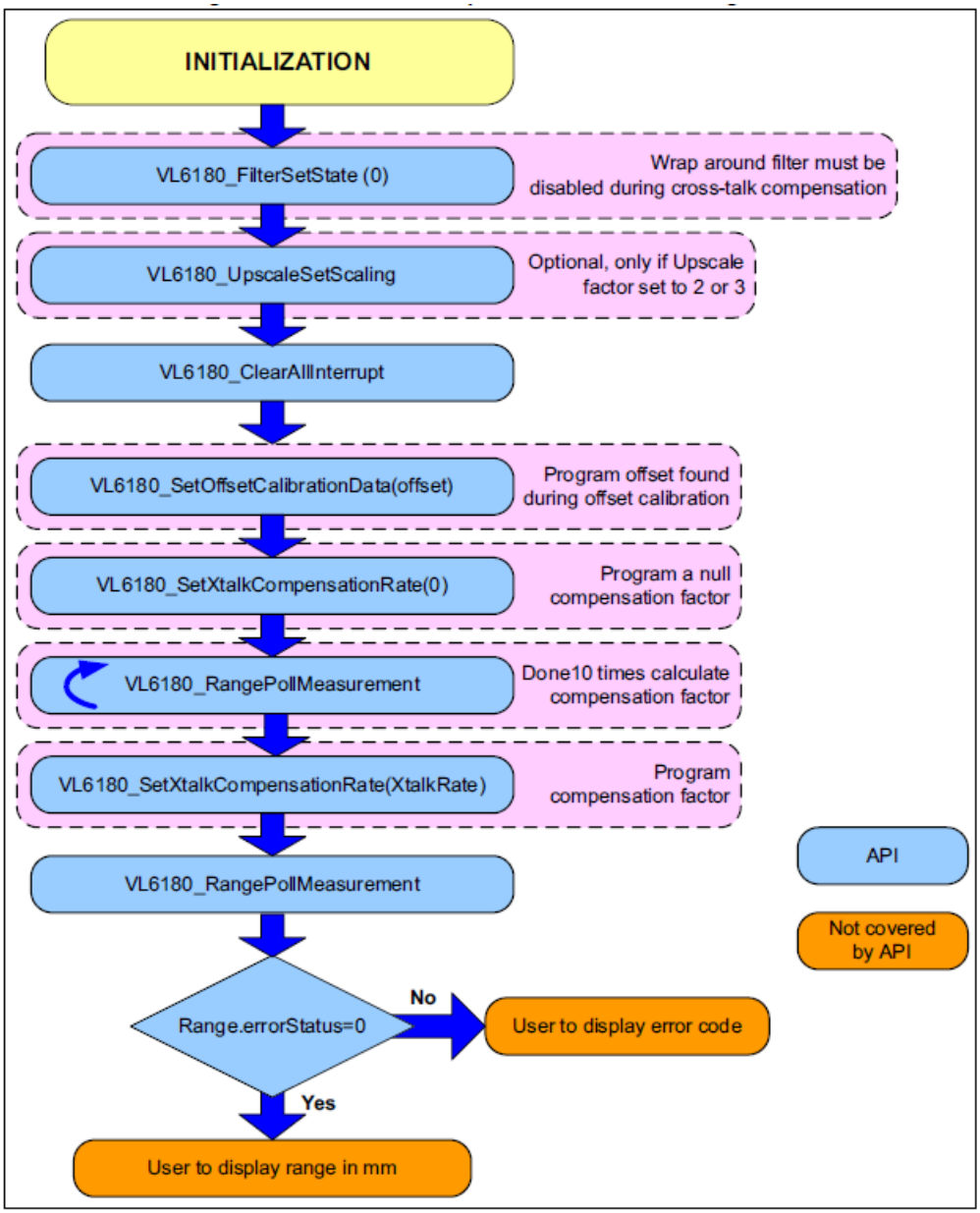
- 100 mm if Upscale factor is 1
- 200 mm if Upscale factor is 2
- 400 mm if Upscale factor is 3

Figure 6. Cross talk compensation environment



Then follow the flow described in the following figure.

Figure 7. **Cross talk compensation factor setting flow**

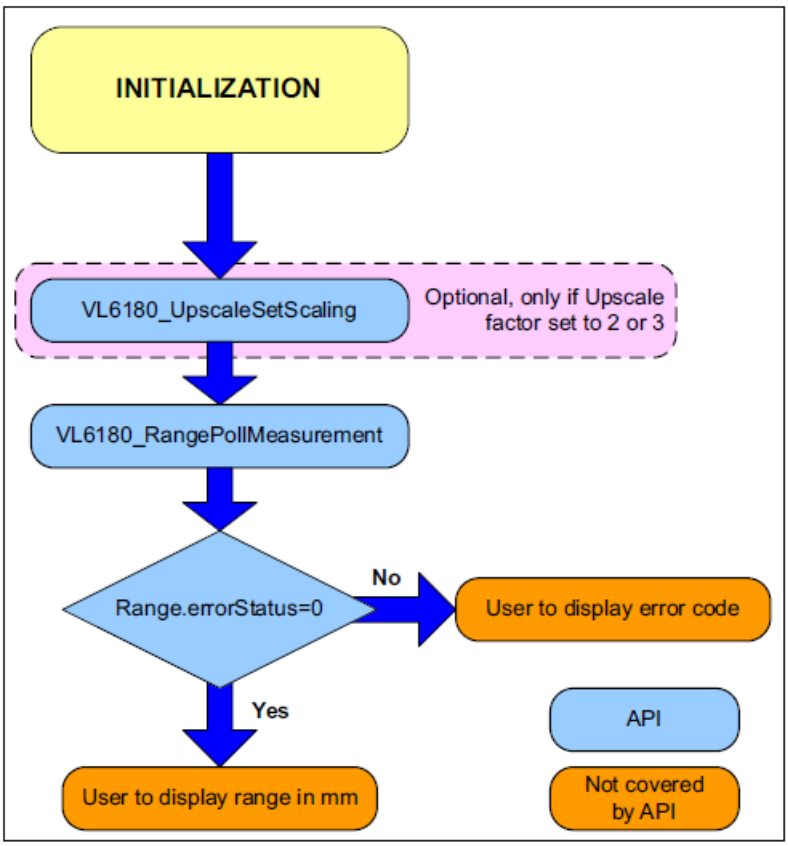


Note: Wrap around filter is disabled during cross talk calibration. User has to re-enable the filter after cross talk calibration is completed by calling VL6180_FilterSetState (1)

5.4 Single range measurement in polling mode

The following figure shows the flow for a single range measurement in polling mode.

Figure 8. Single range measurement flow

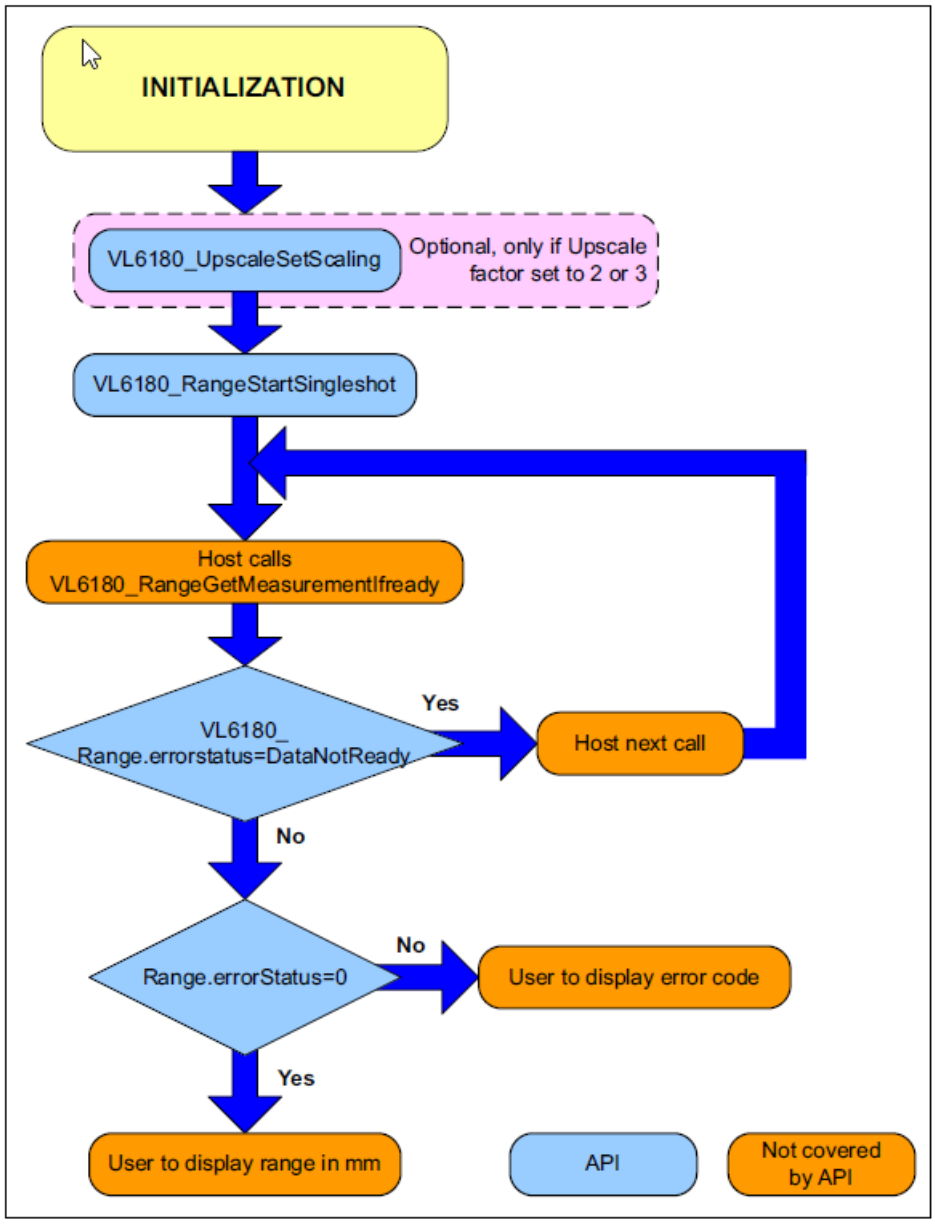


5.5 Single range measurement in non polling mode

This is suitable for applications where host CPU is triggered on an interrupt, not coming from VL6180, to perform range measurement.

The following figure shows the flow for a single range measurement in non polling mode.

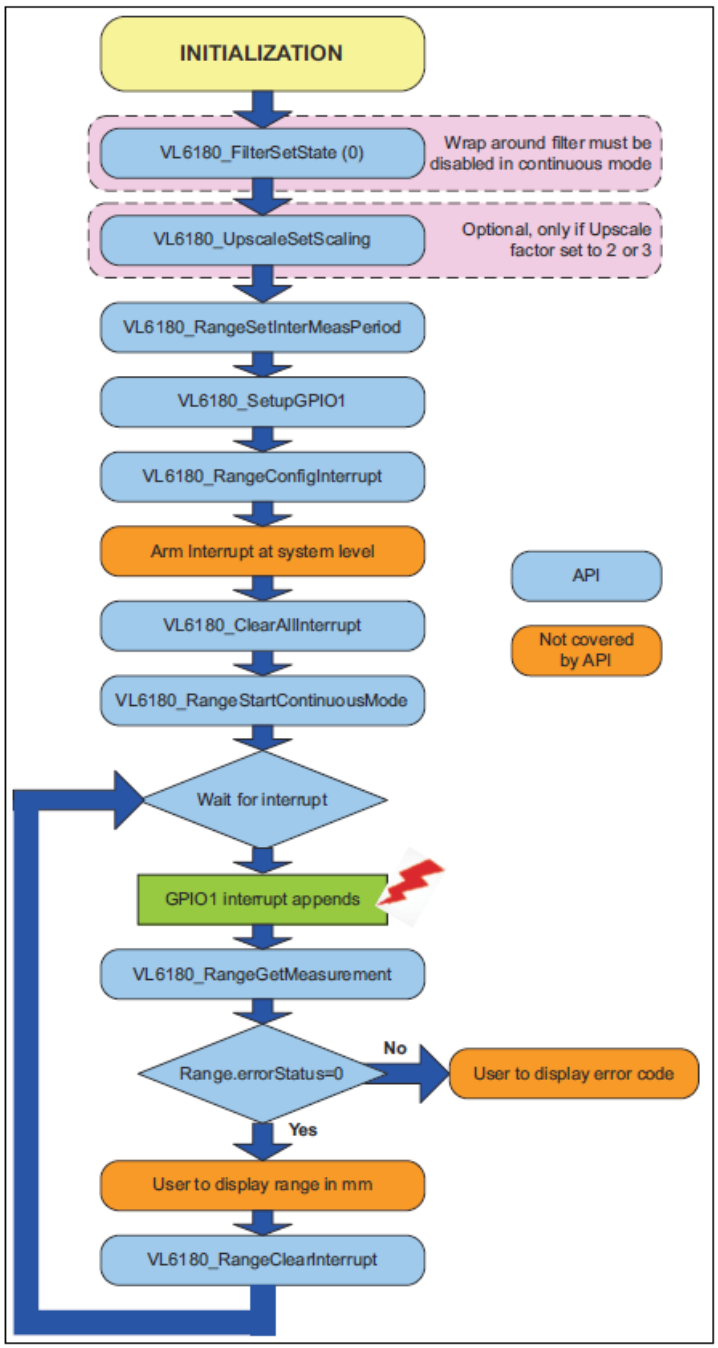
Figure 9. Flow for a single range measurement in non polling mode



5.6 Continuous range measurement

In this mode the host is interrupted by VL6180.
The following figure shows the flow for a continuous range measurement.

Figure 10. Flow for a continuous range measurement



5.7 Early convergence estimate (ECE)

Early Convergence Estimate (ECE) is a programmable feature which is designed to minimize power consumption when there is no target in the field of view (FOV). This feature should only be used when Upscaling is set to 1.

The ECE enables the device to decide whether or not there is a target in the FOV and if it should stop the measurement before the default max convergence time is reached to save power.

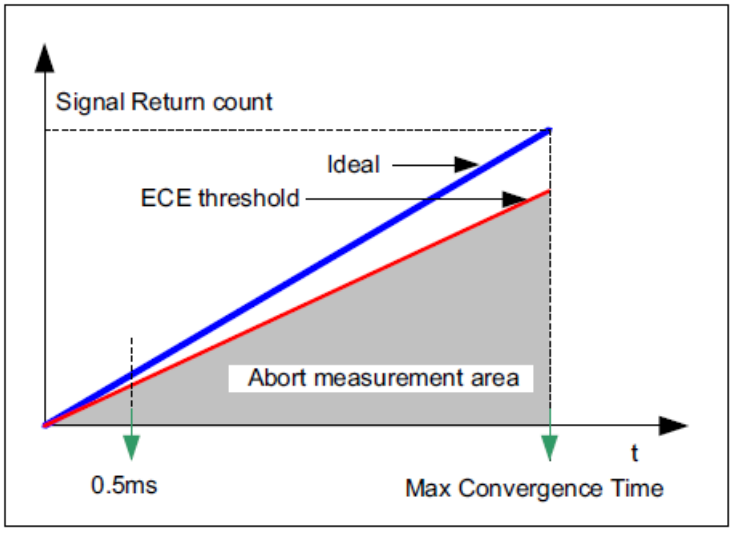
ECE works by calculating the rate of convergence 0.5 ms after the measurement has been started. If the return count rate reported by the device is below the set ECE threshold, the measurement is aborted.

The ECE feature is enabled through VL6180_RangeSetEceState function.

The ECE return count rate threshold is set through VL6180_RangeSetEceFactor function.

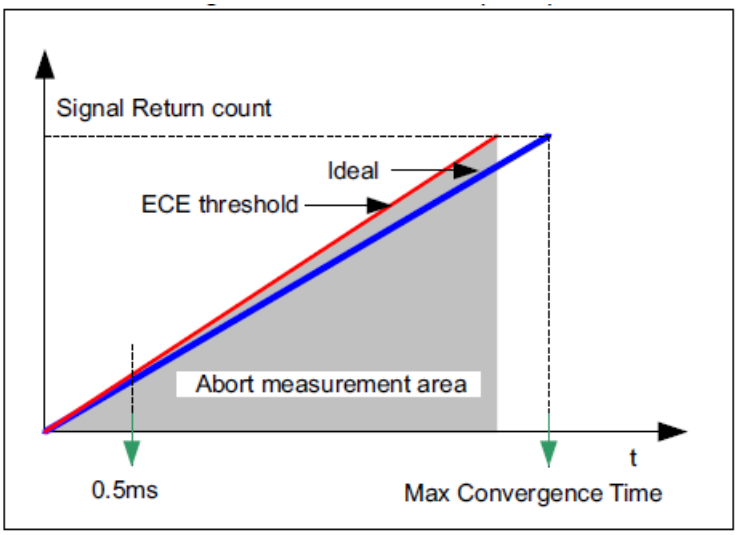
Figure 11. 0.8 ECE factor (80%) and Figure 12. 1.1 ECE factor (110%) show examples with a threshold ratio of 0.8 and of 1.1.

Figure 11. 0.8 ECE factor (80%)



80% ECE factor - if the return signal count is above 80% of the ideal convergence rate after 0.5 ms of the measurement starting then the system will allow the measurement to continue.

Figure 12. 1.1 ECE factor (110%)



110% ECE factor - if the return signal count is 110% of the ideal convergence rate after 0.5 ms of the measurement start, then the system will allow the measurement to continue.

This is beneficial if the user wants to have much better power savings as it will only allow a measurement when there is a material or object in front of the sensor.

The following table gives an example of the current consumption of the VL6180 with and without ECE feature enabled and for a range of inter-measurement periods.

In this example, the max convergence time is 50 ms. The ECE ratio is set to 95%.

Table 3. VL6180 current consumption versus ECE feature and inter-measurement period (in mA)

Inter measurement period (ms)	ECE = ON	ECE = OFF		
	No Target	No Target	Target @ 50mm	Target @ 100mm
2000	0.03	0.53	0.08	0.15
1500	0.04	0.72	0.11	0.2
1000	0.05	1.09	0.18	0.3
750	0.07	1.42	0.22	0.4
500	0.09	2.19	0.33	0.59
250	0.18	4.2	0.62	1.42
100	0.42	9.85	1.45	3.32
50	0.77	18.08	2.59	5.63

5.8 Wrap around filter (WAF)

Mirrors are high reflective targets and even placed at a distance of more than 60 cm from VL6180 they could still produce enough return signal to declare for VL6180 a valid target, which would result in a wrong, under-estimated, returned distance.

WAF function is automatically able to detect such targets and to return the “Measured filtered by WAF” error code 16 (see Table 4. Status error code meaning).

WAF function is enabled through ‘VL6180_FilterSetState’.

5.9 Dmax

Dmax feature is enabled through VL6180_DMaxSetState function.

When ambient light level increases, the max detection range (Dmax) decreases, so a target may not be detected by the VL6180 because it is too far for a given ambient light condition. When no target is detected, no valid distance reported, Dmax function is able to define the maximum distance up to which a 17% reflective target is detected with the current ambient light condition.

5.10 Range status error code information

The following table lists each range status error code.

Table 4. Status error code meaning

Code		Name	Comment
0	0b0000	0b0000 No error	Valid distance
1	0b0001	VCSEL_Continuity_Test	This error can only happen at the first boot-up. Device needs to be reset to clear the error. If the error persists, then device is defective and must be replaced.
2	0b0010	VCSEL_Watchdog_Test	
3	0b0011	VCSEL_Watchdog	
4	0b0100	PLL1_Lock	
5	0b0101	PLL2_Lock	
6	0b0110	Early_Convergence_Estimate	If Early Convergence Estimate function (ECE) is enable, no valid target detected after early convergence estimate time.
7	0b0111	Max_Convergence	No valid target detected after maximum convergence time.
8	0b1000	No_Target_Ignore	Ignore threshold check failed.
9	0b1001	Not used	
10	0b1010		
11	0b1011	Max_Signal_To_Noise_Ratio	Too much ambient light detected
12	0b1100	Raw_Ranging_Algo_Underflow	Could happen If target is between 0 and 10mm and offset is not correctly set.
13	0b1101	Raw_Ranging_Algo_Overflow	Could happen if the target is detected at a distance higher than 55 cm for upscaling = 3, 20 cm for upscaling = 1. Error is due to internal variables/registers overflow.
14	0b1110	Ranging_Algo_Underflow	Could happen if target is between 0 and 10mm and offset is not correctly set.
15	0b1111	Ranging_Algo_Overflow	Could happen if the target is detected at a distance higher than 55 cm for upscaling = 3, 20 cm for upscaling = 1. Error is due to internal variables/registers overflow
16	0b10000	Filtered by post-processing	If Wrap Around Filter is enabled, this error gives the information that a bright target is detected between 60 cm and 120 cm.
18	0b10010	DataNotReady	This error is returned by the VL6180_RangeGetMeasurementIfReady API function when ranging sample data is not ready

6 Multiple VL6180 application

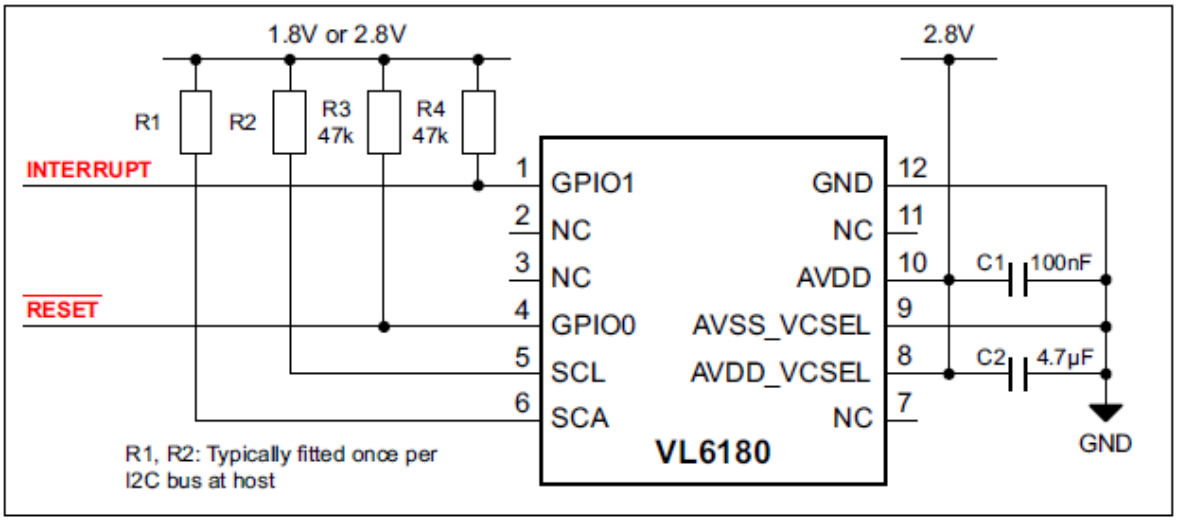
This chapter shows how multiple VL6180 devices can be used on a board design while only using a single I2C interface to interact with all the devices.

Each VL6180 device has both a reset pin and an interrupt pin, which can be used to enable a multi device setup.

6.1 VL6180 control management

The following figure is a typical example schematic using a VL6180 device. Since the VL6180 can have the I2C device address changed by doing an I2C write once it is booted, a separate reset pin would be needed for each VL6180 used in a design. Each device is then taken out of reset one at a time, and then the I2C Device Address is changed to a new unique address. This can be done using multiple GPIO pins from the host on the board.

Figure 13. VL6180 typical application



Depending of the number of VL6180 devices used in a design and the number of available GPIO pins from the host, a GPIO expander may be required to manage the reset (GPIO0) and the interrupt (GPIO1) pins of the VL6180.

Revision history

Table 5. **Document revision history**

Date	Version	Changes
10-Sep-2020	1	Initial release
08-Oct-2020	2	Updated metadata

Contents