UM1983
User manual

VL6180X proximity, gesture and ambient light sensing (ALS) module

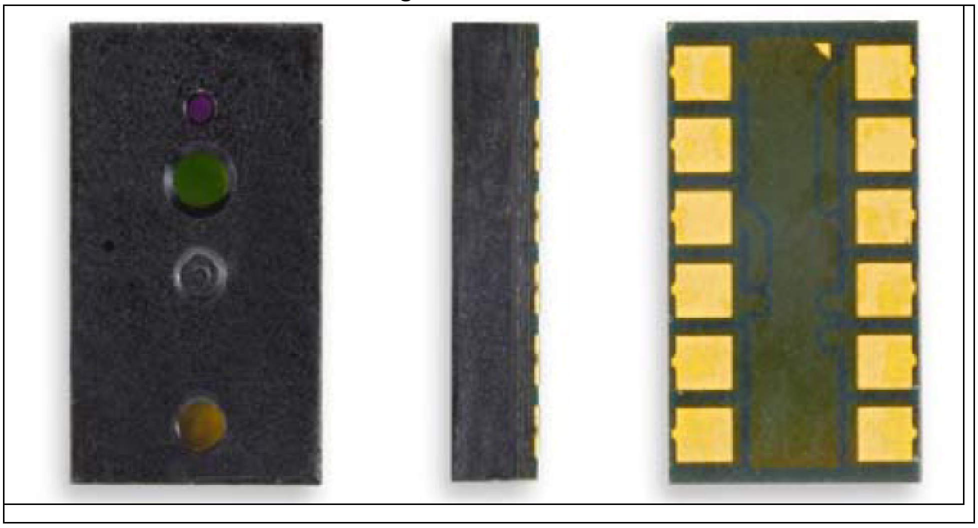
Introduction

The VL6180X is a proximity, gesture and ambient light sensor based on ST's patented FlightSense™ technology. The VL6180X interfaces to your micro-controller via the industry standard I²C bus.

This user manual contains:

- VL6180X hardware information (Application schematic, Mechanical data, soldering process).
- List of available VL6180X evaluation and development tools.
- How to download the associated API.
- How to control through the API.
  - The range feature.
  - The ALS feature.
  - Several VL6180X in a design.

Figure 1. VL6180X



Contents

1 Document references

Table 1. Document references

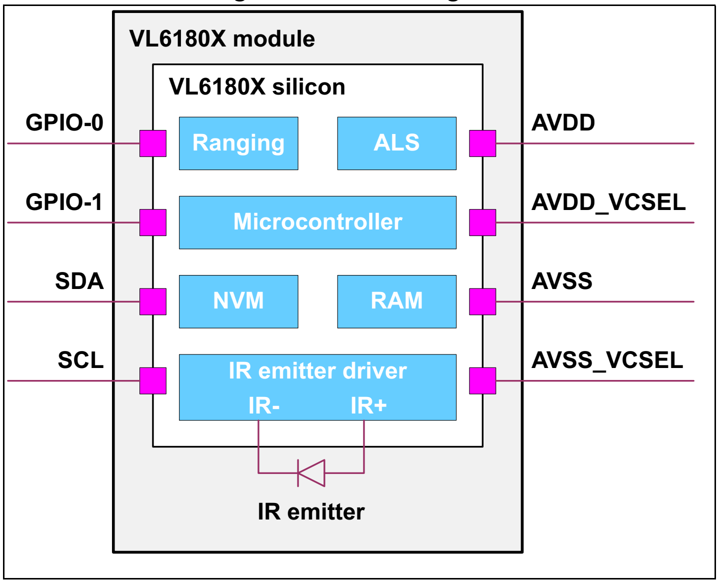
Description	DocID
Datasheet - VL6180X proximity and ambient light sensing (ALS) module	DocID026171
Data brief - Proximity, gesture, ambient light sensor expansion board based on VL6180X for STM32F401RE	DocID027616
Data brief - Proximity, gesture, ambient light sensor expansion board based on VL6180X for STM32L053R8	DocID027625
Data brief - Proximity and ambient light sensor expansion board based on VL6180X for STM32 Nucleo	DocID027252
Data brief - P-NUCLEO-6180X1 and P-NUCLEO-6180X2 packs PC graphical user interface (GUI)	DocID027684
Data brief - Proximity, gesture, ambient light sensor software expansion for STM32Cube	DocID027687
Data-brief - VL6180X application programing interface (API)	DocID027370
Data-brief - VL6180X satellite boards compatible with VL6180X boards	DocID027253

2 Hardware information

2.1 Application information

2.1.1 Block diagram

Figure 2. Block diagram



VL6180X module embeds:

- a VCSEL IR emitter
- a VL6180X silicon which embeds:
  - a range sensor
  - an ambient light sensor (ALS)
  - a microcontroller and its associated RAM
  - a non volatile memory (NVM) to store sensor default configuration data.

2.1.2 Schematic

Figure 3. Application schematic

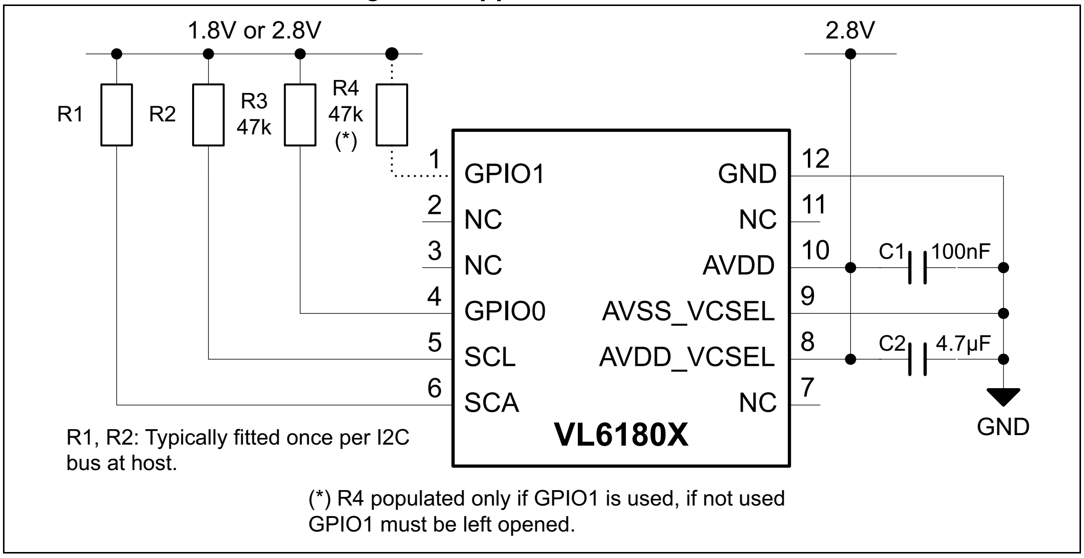
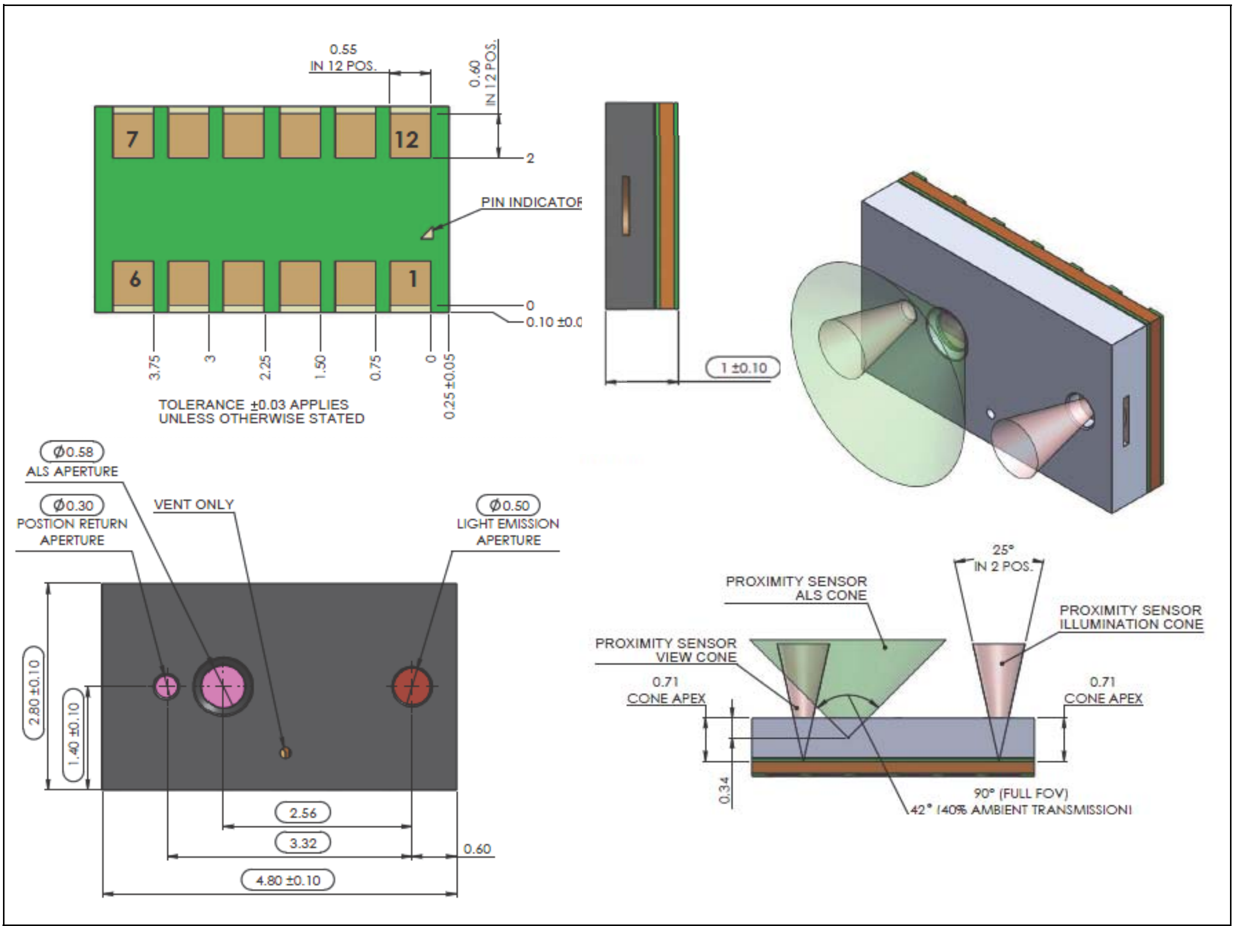


Table 2. Pin descriptions

Pin number	Signal name	Description
1	GPIO1	Open drain interrupt output
2	NC	No connect
3	NC	No connect
4	GPIO0	Chip enable
5	SCL	I2C serial clock
6	SDA	I2C serial data
7	NC	No connect
8	AVDD_VCSEL	VCSEL power supply 2.6 to 3.0V
9	AVSS_VCSEL	VCSEL ground
10	AVDD	Digital/Analog power supply 2.6 to 3.0V
11	NC	No connect
12	GND	Digital/Analog ground

2.1.3 Mechanical data

Figure 4. Mechanical drawing



2.1.4 Recommended solder pad dimension and reflow profile

Figure 5. Solder pattern

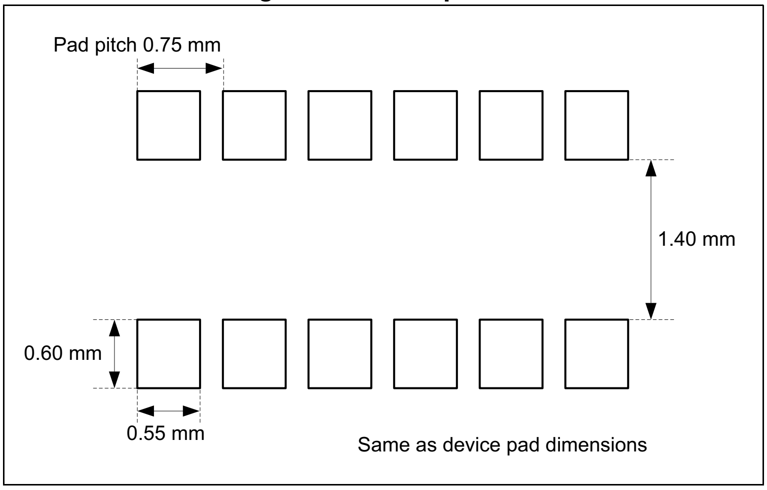


Figure 6. Reflow profile

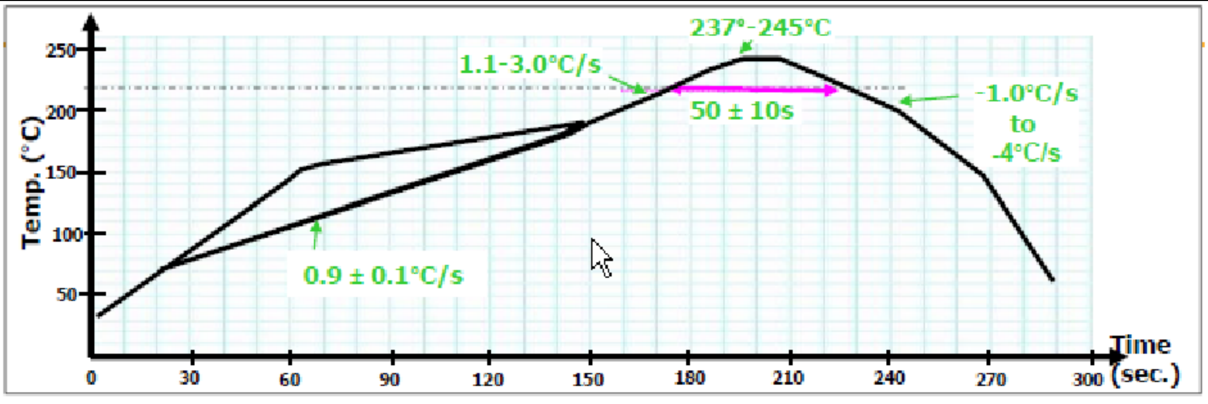


Table 3. Recommended reflow profile

Profile	Ramp to strike	
Temperature gradient in preheat	(T= 70 - 180°C):	0.9 +/- 0.1°C/s
Temperature gradient	(T= 200 - 225°C):	1.1 - 3.0°C/s
Peak temperature in reflow	237°C - 245°C	
Time above 220°C	50 +/- 10 seconds	
Temperature gradient in cooling	-1 to -4 °C/s (-6°C/s maximum)	
Time from 50 to 220°C	160 to 220 seconds	

Note: As the VL6180X package is not sealed, only a dry re-flow process should be used (such as convection re-flow). Vapor phase re-flow is not suitable for this type of optical component.

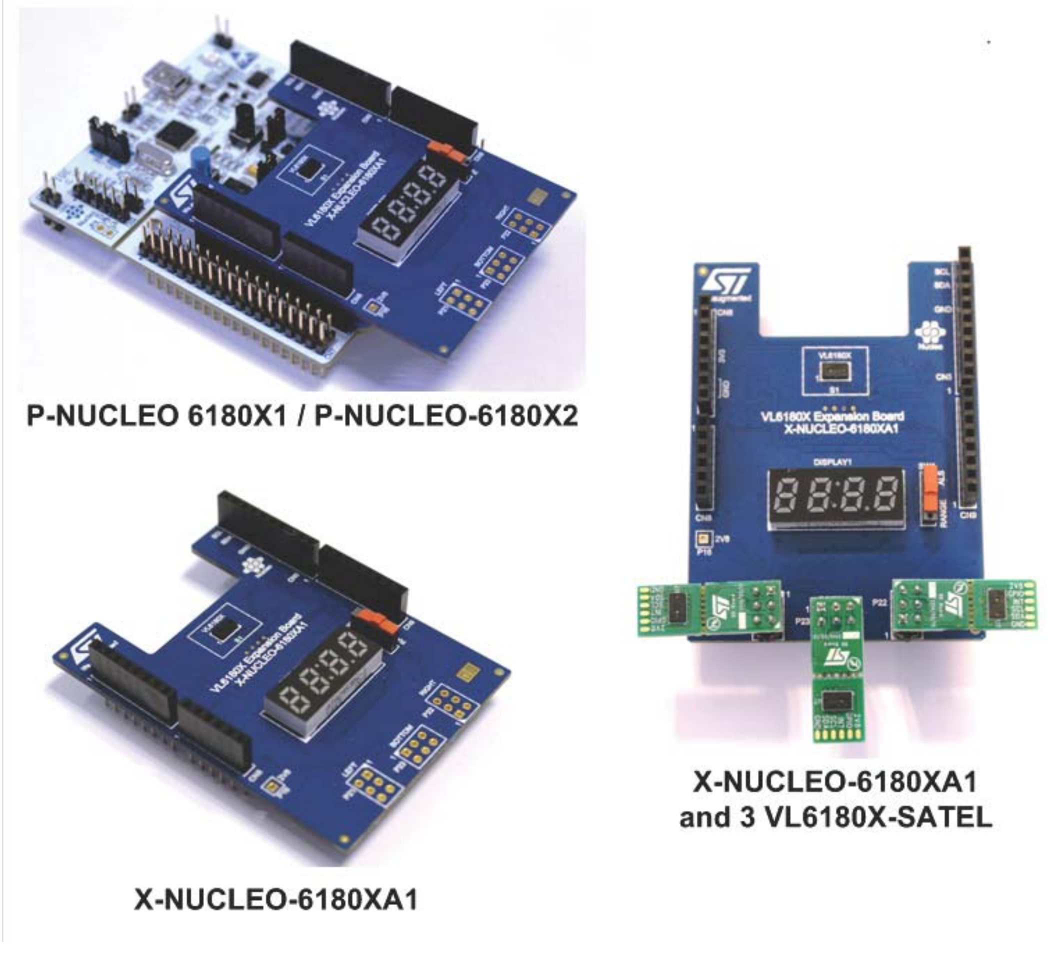
The VL6180X is an optical component and as such, it should be treated carefully. This would typically include using a 'no-wash' assembly process.

2.2 Evaluation and development tools

Several tools can be ordered on www.st.com/vl6180x in "Design Resources" page to evaluate and develop applications around the VL6180X (see *Figure 7*):

- X-NUCLEO-6180XA1: VL6180X expansion board compatible with the STM32 Nucleo board family and with the Arduino UNO R3 connector layout.
- P-NUCLEO-6180X1: evaluation pack composed of a X-NUCLEO-6180XA1 expansion board and a powerful STM32F401RE Nucleo board.
- P-NUCLEO-6180X2: Evaluation pack composed of a X-NUCLEO-6180XA1 expansion board and a ultra low power STM32FLO53R8 Nucleo board.
- VL6180X-SATEL: Hardware add-on to X-NUCLEO-6180XA1 expansion board to enable multi VL6180X sensor applications

Figure 7. VL6180X evaluation and development tools



Information on the installation of the different softwares allowing the evaluation and the application development around VL6180X are given in the user manual *UM1876* "Getting started with VL6180X proximity, gesture, ambient light sensor software expansion for STM32Cube".

3 Application program interface (API) overview

This chapter details the download and the installation of the API.

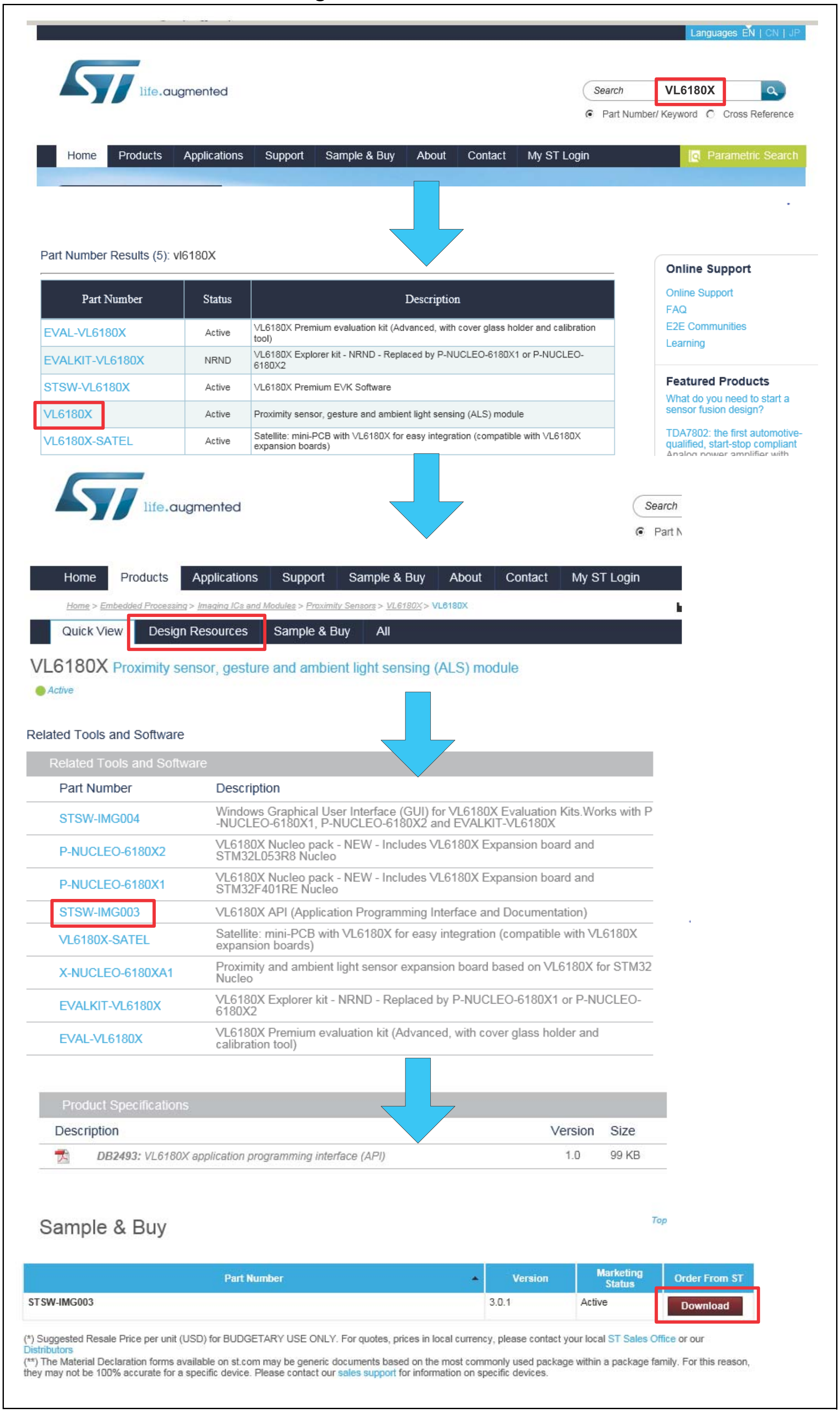
The VL6180X API is a set of C functions controlling the VL6180X (init, ranging, ALS,…) to enable the development of end-user applications. This API is structured in a way it can be compiled on any kind of platform through a well isolated platform layer (mainly for low level I2C access).
Several code examples are provided to show how to use the API and perform ranging and ALS measurements.
A complete Nucleo F401 + VL6180X expansion board project is also provided (Keil 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 VL6180X API (see:*Figure 8*)

- on *www.st.com*, search for “VL6180X”
- on next page select: “VL6180X”
- On next page select “design resources”
- On “design resources” page select “STSW-IMG003”
- On next page “download”

Figure 8. API download



3.2 Quick start guide for API integration

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

- API_Documentation_proximity.chm
- API_Documentation_proximity.html

The VL6180X API is integrated in a software project in two steps

1. Developer has to add/link the files listed in *Table 4* and in *Figure 9* to his source and include code path. Some files may require modification to comply with the final application or the hardware/software capabilities.

Table 4. API header files

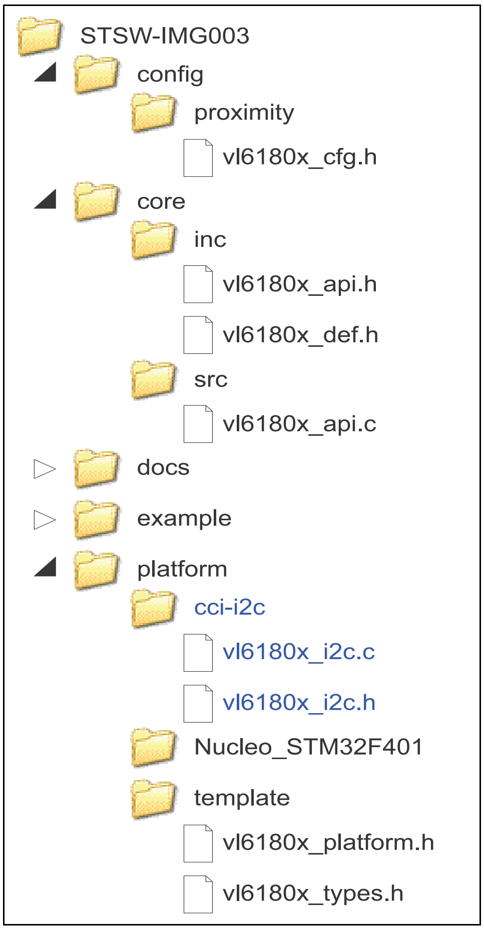
Names	Description
vl6180x_cfg.h	Application configuration **May require modification**
vl6180x_api.c and vl6180x_api.h	All operating functions at high and low level to control the sensor **Must not be modified**
vl6180x_def.h	Definition of constants and structures used in the API **Must not be modified**
vl6180x_platform.h	Target platform specific declarations/prototypes **May require modification**
vl6180x_types.h	Basic types definition **May require porting**

2. To manage the data communication between the VL6180X 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 & write functions which perform CCI register access to the device:
   VL6180x_WrByte(); VL6180x_WrWord(); VL6180x_WrDWord(); VL6180x_UpdateByte(); VL6180x_RdByte(); VL6180x_RdWord(); VL6180x_RdDWord().

   To implement these 7 functions, it is recommended to use vl6180x_i2c.c and vl6180x_i2c.h files in *platform/cci_i2c* directory (see *Figure 9*)

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 9. Iheader and CCI service files in the API



4 Main feature definitions

This chapter gives the definition of the main VL6180X features described in this user manual.

Ranging

Measurement of the distance between VL6180X and the target.

Upscale support - Extended range

- Upscale factor = 1, VL6180X measures distances up to 20cm with a granularity of 1mm.
- Upscale factor = 2, VL6180X measures distances up to 40cm with a granularity of 2mm.
- Upscale factor = 3, VL6180X measures distances up to 60cm with a granularity of 3mm.

Note: VL6180X ranging performances are specified up to 10 cm.

Wrap around filter (WAF)

In specific conditions, when targeting a mirror or a very reflective metal, a wrap around effect can occur internally to the VL6180X 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 VL6180X will report a valid measurement with a 17% grey target for the current ambient light conditions.

Dmax decreases when ambient light increases.

Ambient Light Sensor (ALS)

Measurement in Lux of the ambient light of the scene.

5 VL6180X range feature

5.1 Typical ranging output

5.1.1 Targets

For the purposes of this document all targets are referred to in terms of their photopic (visible spectrum) reflectance. It should be noted that the photopic reflectance of a target is not necessarily the same as the reflectance at 850nm (the wavelength of the VL6180X emitter).

Unless otherwise specified, all targets referenced in VL6180X documentation are Munsell neutral color (gray) charts from X-rite (www.xrite.com), filling the entire field of view. The Munsell notation and reflectance is listed here for each target used.

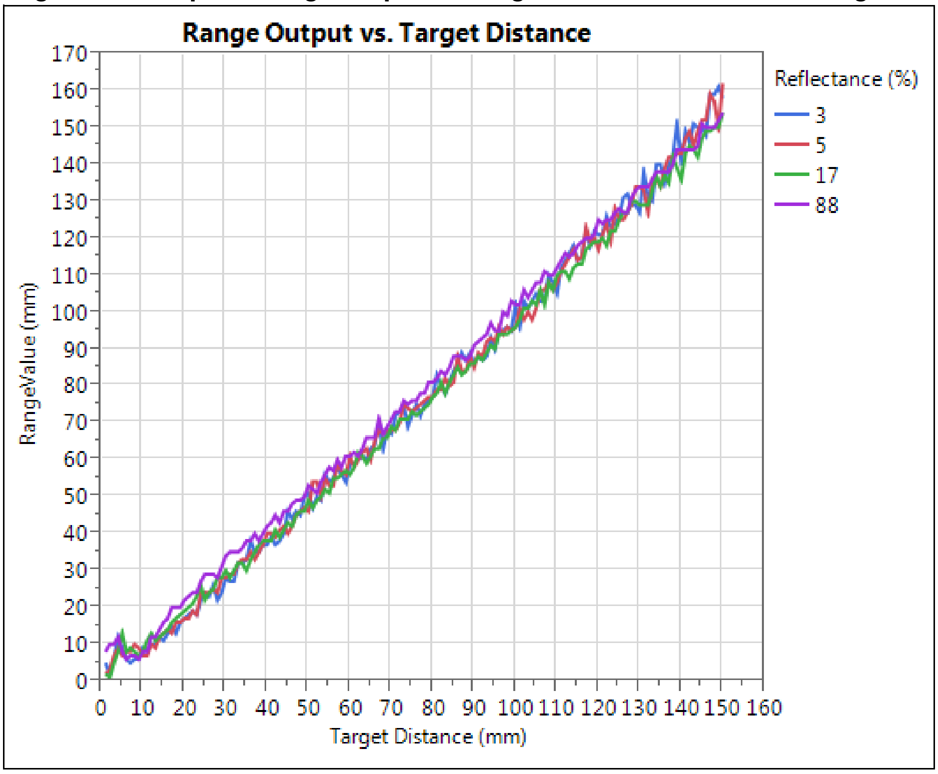
- N2.00/M 3% ‘Black’
- N2.75/M 5% ‘Black’
- N4.75/M 17% ‘Gray’
- N9.50/M 88% ‘White’

Note: /M refers to a matt finish on the target

5.1.2 Range output without cover glass

Figure 10 shows the typical ranging output from VL6180X for different targets at different distances. The test is performed in the dark.

Figure 10. Graph of range output vs. target distance without cover glass.



The range output of VL6180X with each of the targets should be linear with range. There could however be an offset error. This error can be corrected by performing an offset calibration, see *Section 5.4*.

Note: See also the VL6180X datasheet for the specification on offset error

Note: At <10mm the interaction between the target and the VL6180X will prevent the range output from reaching 0mm. This is due to a number of physical effects:

- *Separation between VCSEL and return array*
- *Multiple reflections between target surface and the VL6180X*
- *VCSEL output penetrating the target surface and scattering off the layers inside the target*

5.1.3 Range output with cover glass

The VL6180X should be used with a cover glass. The cover glass can cause internal reflection and this can be detected by the VL6180X as unwanted signals. This is known as the cross talk.

The cross talk can affect the range output, hence we recommend the user perform the cross talk compensation calibration procedure when using the VL6180X with cover glass (see *Section 5.5*).

Figure 11 shows the impact of the cover glass on the range output of the VL6180X. The internal reflection between the glass and the VL6180X causes the ranging output to decrease at longer distances.

This error in range is a ratio of the target signal rate and magnitude of the cross talk, hence the range output from a darker target is more susceptible to the effect of cross talk.

Figure 12 shows the range output from the VL6180X with the cross talk compensation calibration procedure implemented. The range error caused by the cover glass is corrected.

The tests are performed in the dark.

Note: The offset might need to be re-calibrated when ranging though cover glass.
Offset calibration should always be done before cross talk compensation.
The cross talk rate depends on the glass type and its placement relative to the VL6180X. If the glass type or its position changes, then cross talk compensation calibration might need to be re-done.

Figure 11. Graph of range output vs. target distance with cover glass and without cross talk compensation calibration.

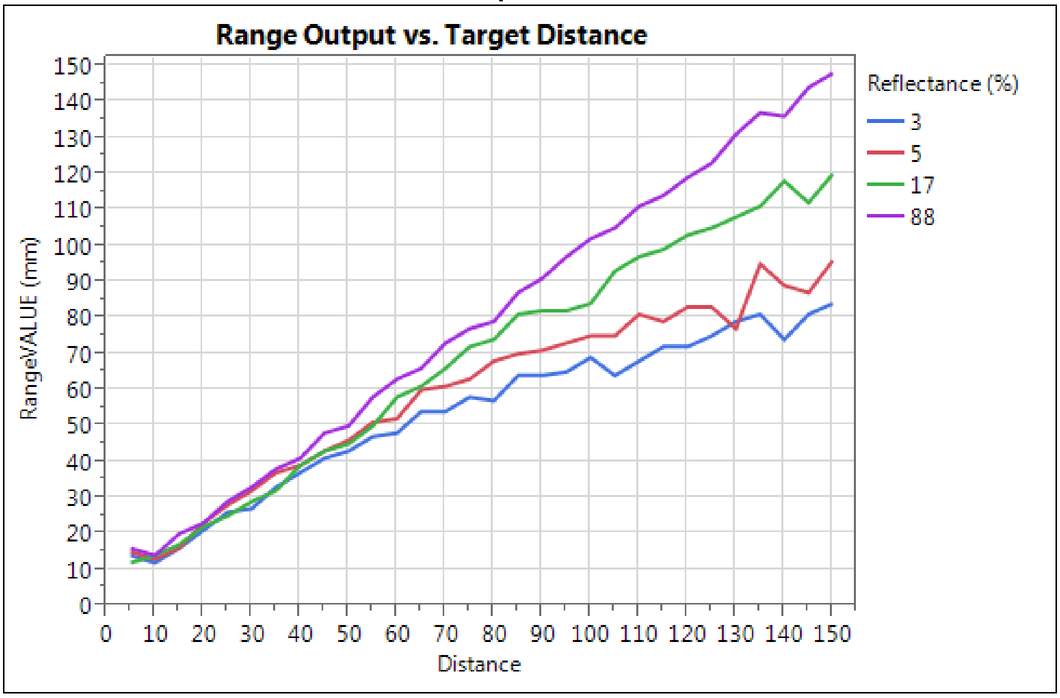
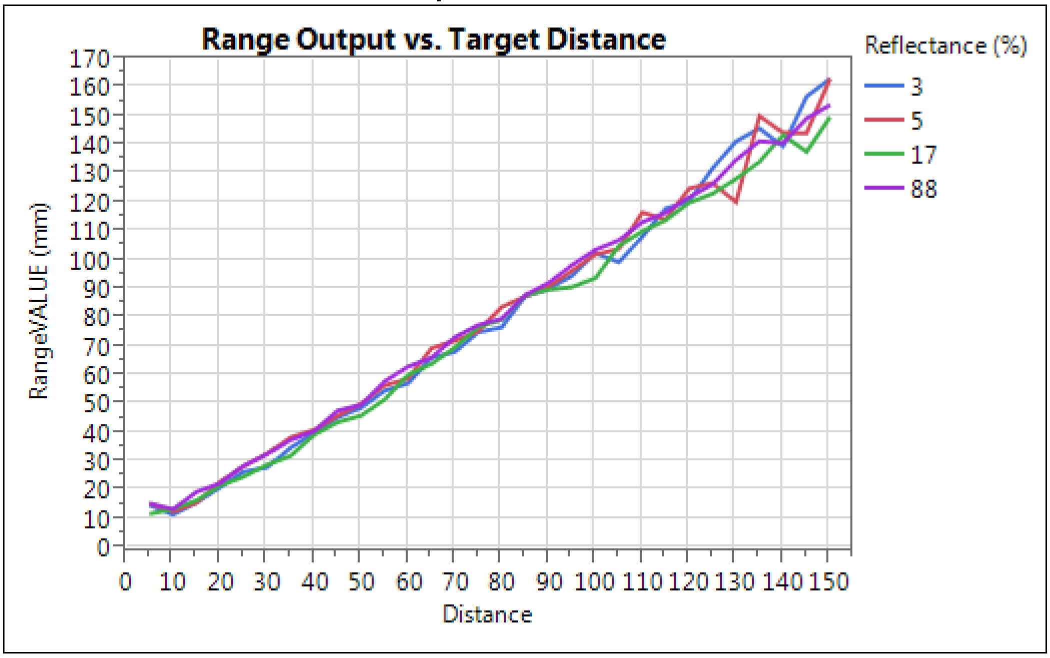


Figure 12. Graph of range output vs. target distance with cover glass and cross talk compensation calibration



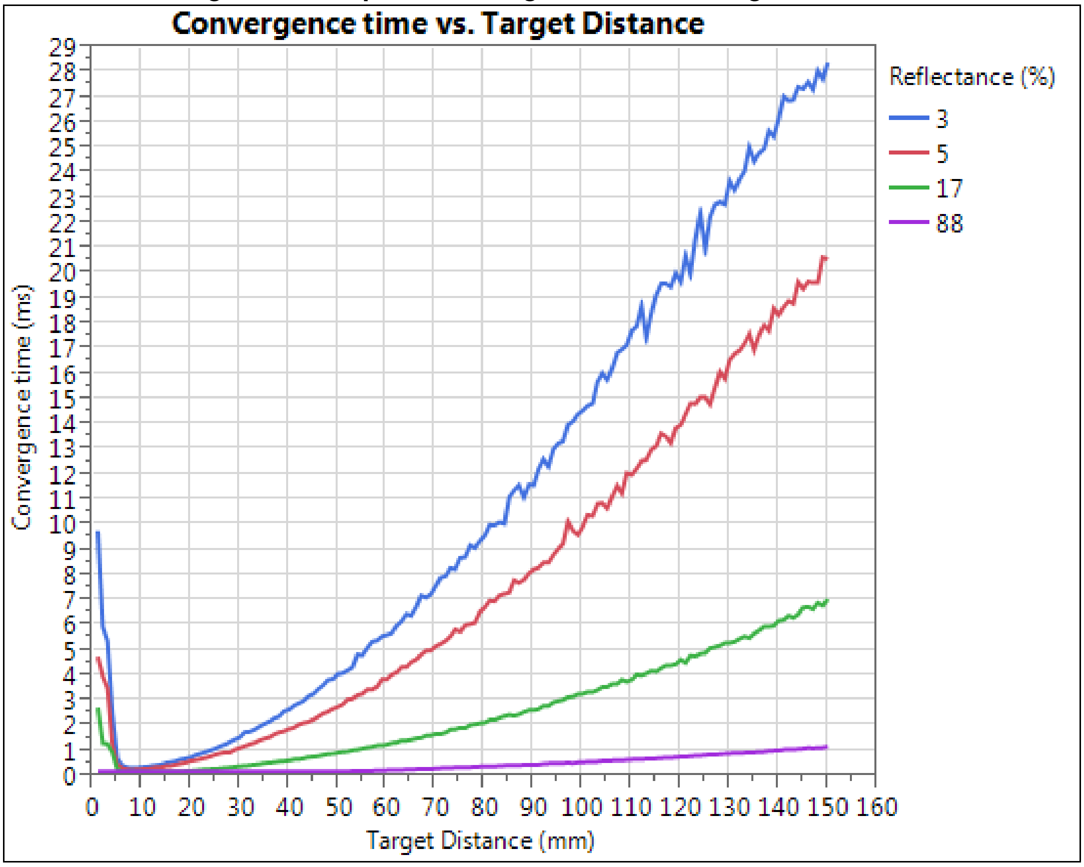
5.2 Convergence time

Figure 13 shows the typical convergence time output from VL6180X for different targets at different distances. The convergence time can provide useful information on how to optimise VL6180X settings to perform more efficiently and reliably.

The test is performed in the dark and with no cover glass.

Note: Cover glass can alter the return convergence time *characteristics. We therefore recommend the* convergence time *be re-characterized once the VL6180X has been integrated into the final system before attempting to perform any system optimization.*

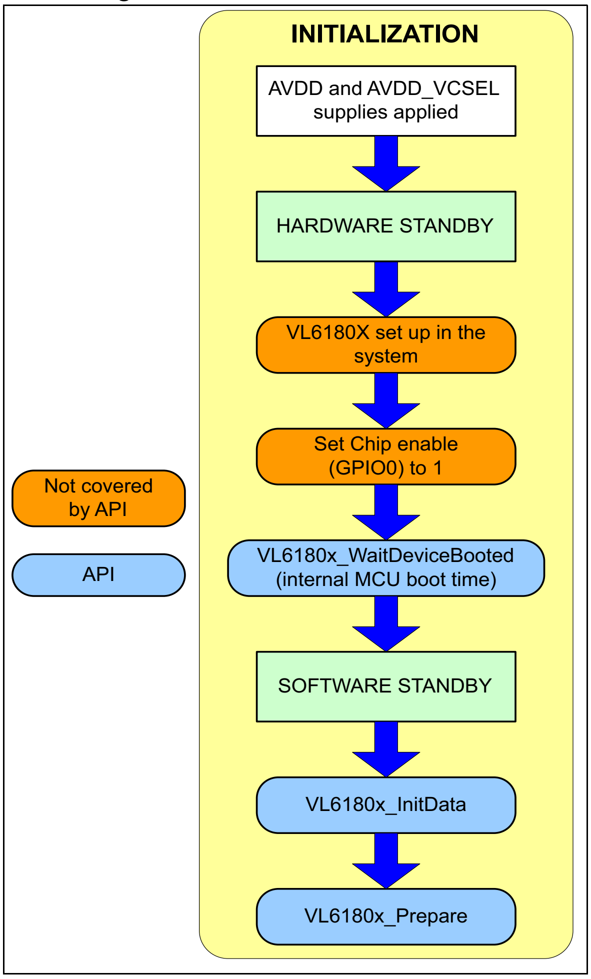
Figure 13. Graph of convergence time vs. target distance



5.3 VL6180X Initialization

Figure 14 shows the initialization of the VL6180X in the system

Figure 14. VL6180X initialization

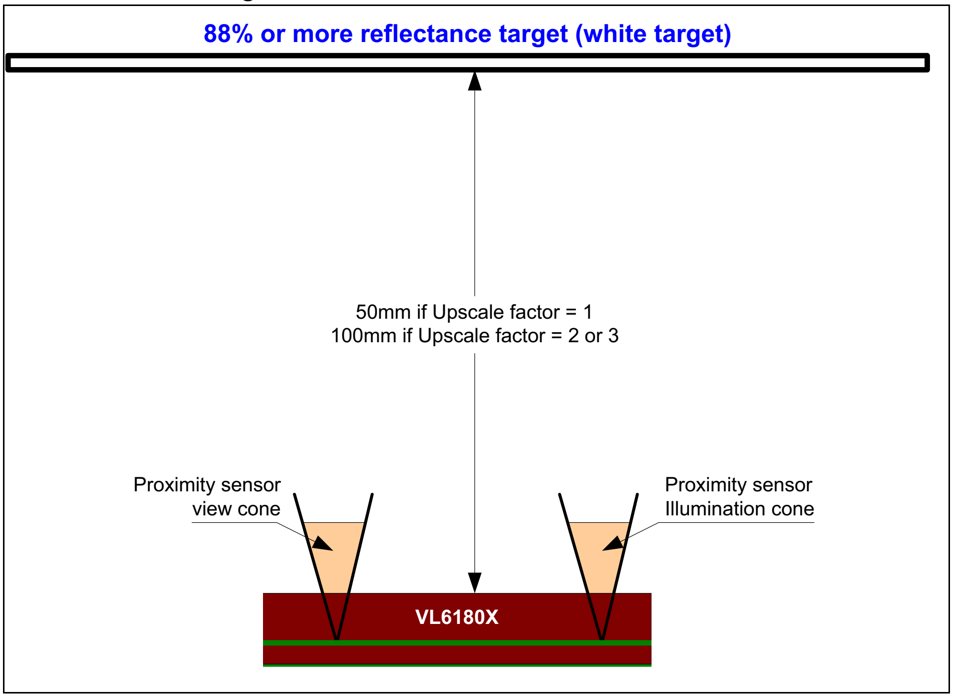


- VL6180X set up in the system
  - The user has to set all variables linked to the integration of the VL6180X in its application (e.g.: VL6180X I2C 7 bit default address = 0x29)
- VL6180x_InitData (One time device initialization)
  - Enables common functions, ie, WAF, ECE, upscaling.
  - Expects device to be fresh out of reset.
- VL6180x_Prepare (Prepare device for operation)
  - Programs common default settings.
  - Prepares the VL6180X for new ranging or ALS measurement.

5.4 Offset calibration

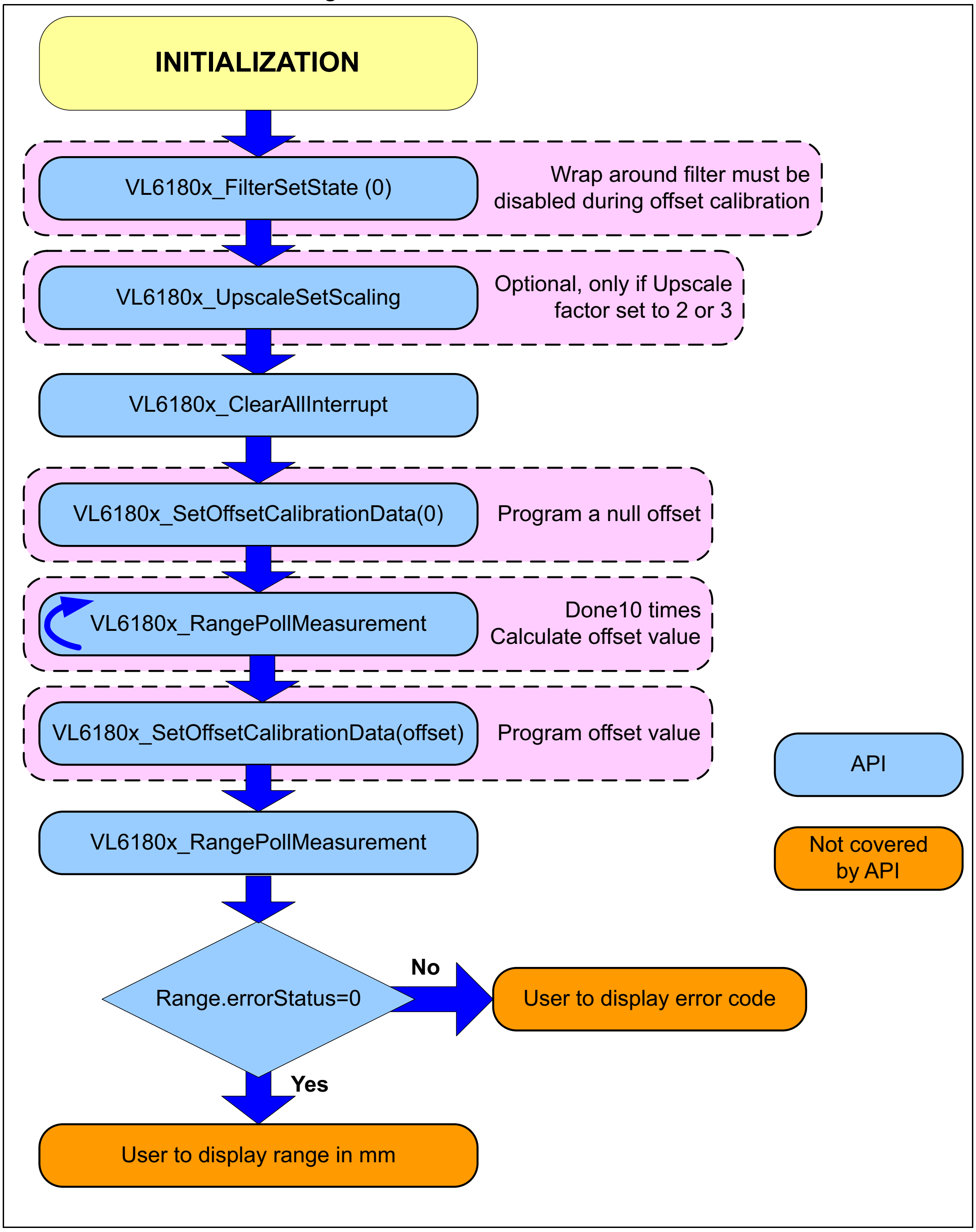
Place a target with a reflectance of 88% or higher (white target) at 50mm away from VL6180X if Upscale factor is 1 or at 100mm if Upscale factor is 2 or 3.

Figure 15. Offset calibration environment



Then follow the flow described in *Figure 16*.

Figure 16. Offset calibration flow



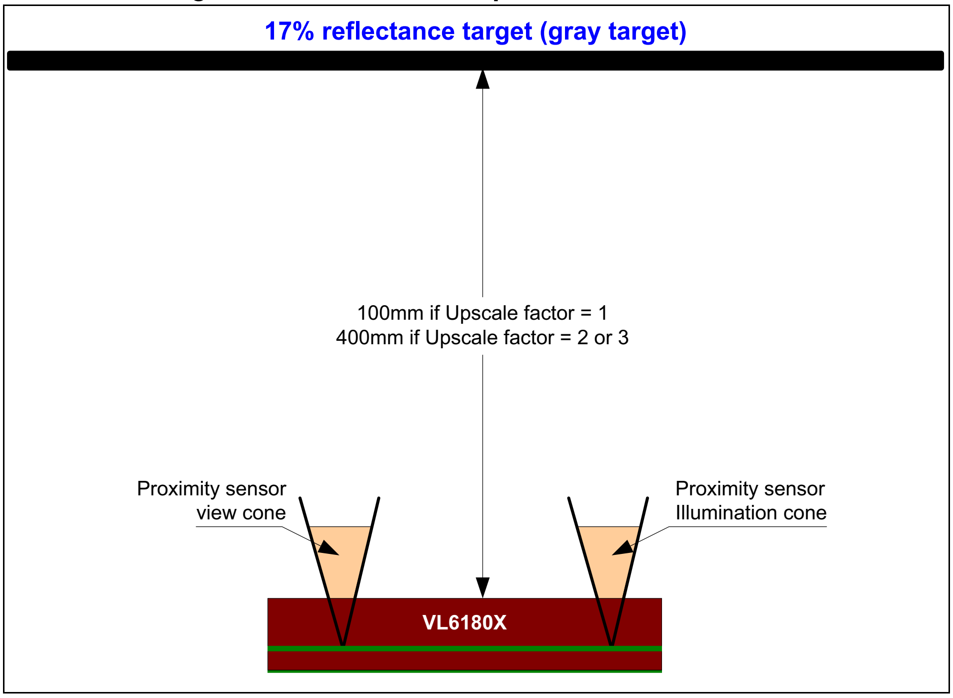
5.5 Cross-talk compensation

Cross-talk compensation must be run after offset calibration.

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

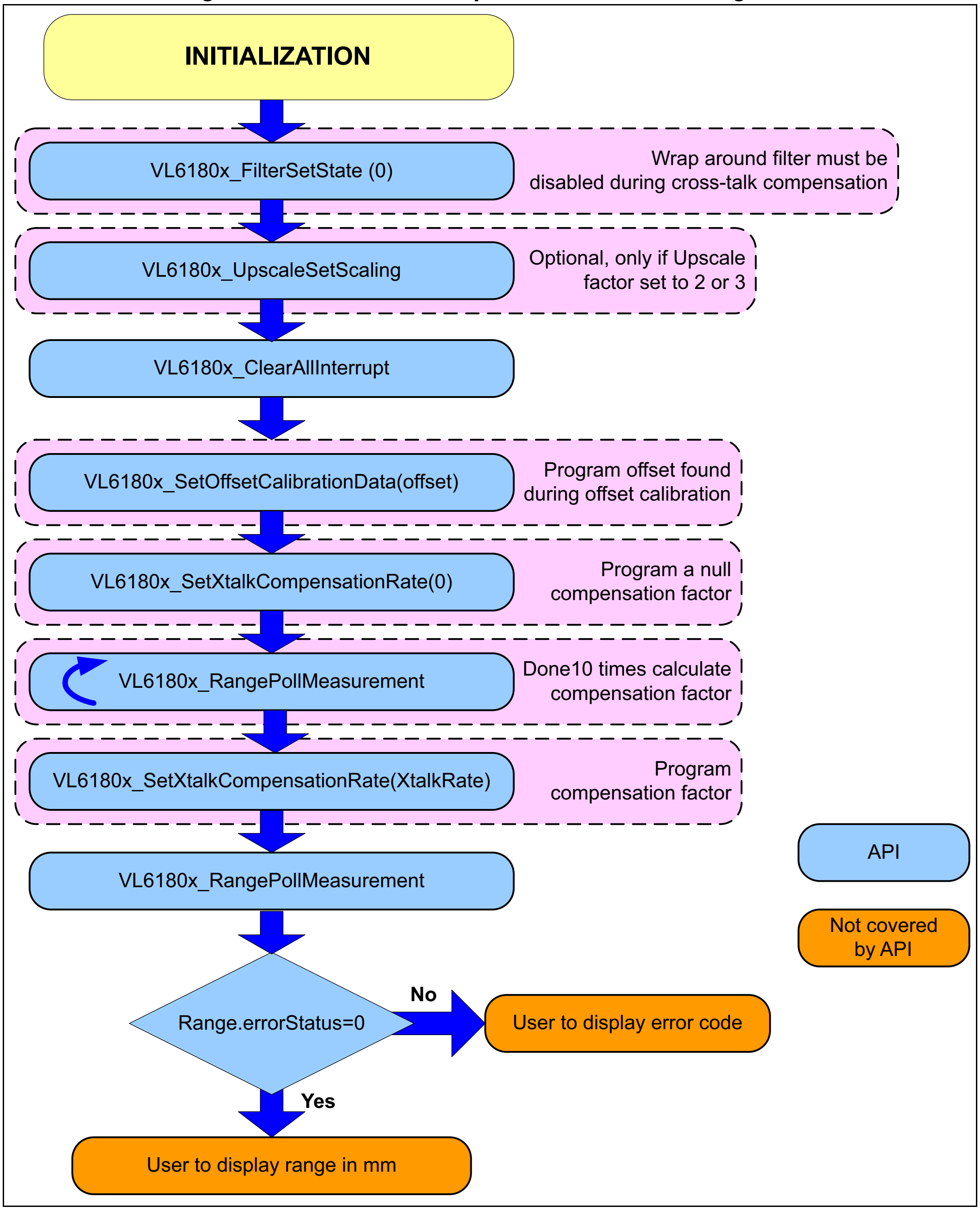
Place a target with a low reflectance e.g. 17% reflectance target (gray target) at 100mm away from VL6180X if Upscale factor is 1 or at 400mm if Upscale factor is 2 or 3.

Figure 17. Cross-talk compensation environment



Then follow the flow described in *Figure 18*.

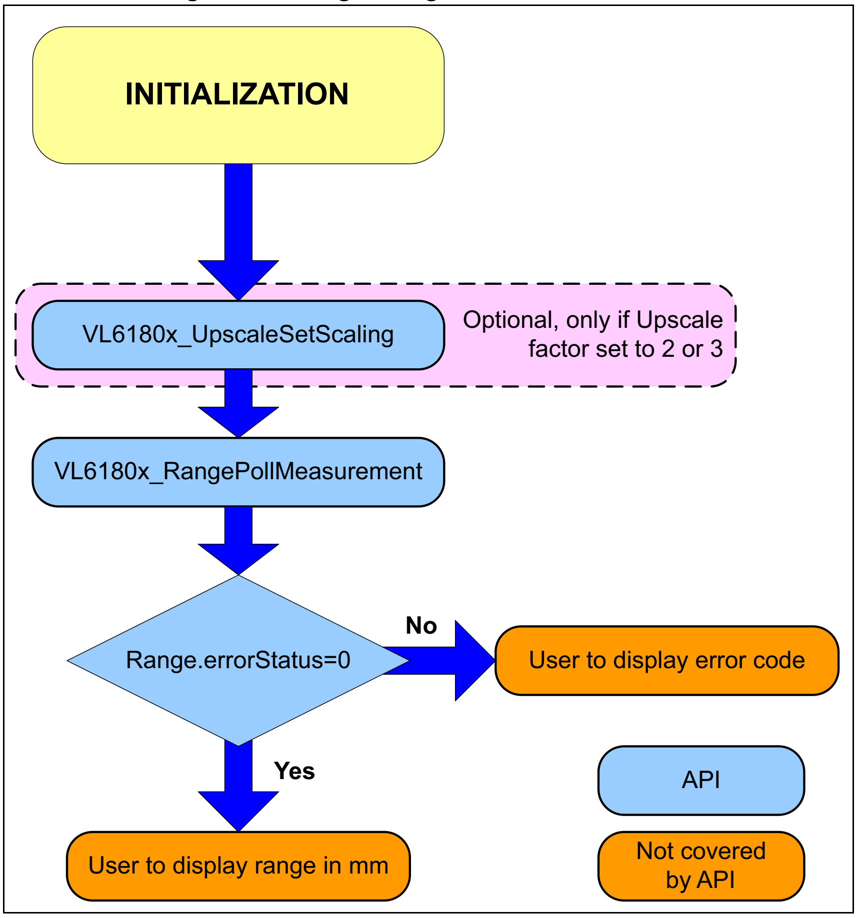
Figure 18. Cross-talk compensation factor setting flow



5.6 Single range measurement in polling mode

Figure 19 shows the flow for a single range measurement in polling mode.

Figure 19. Single range measurement flow

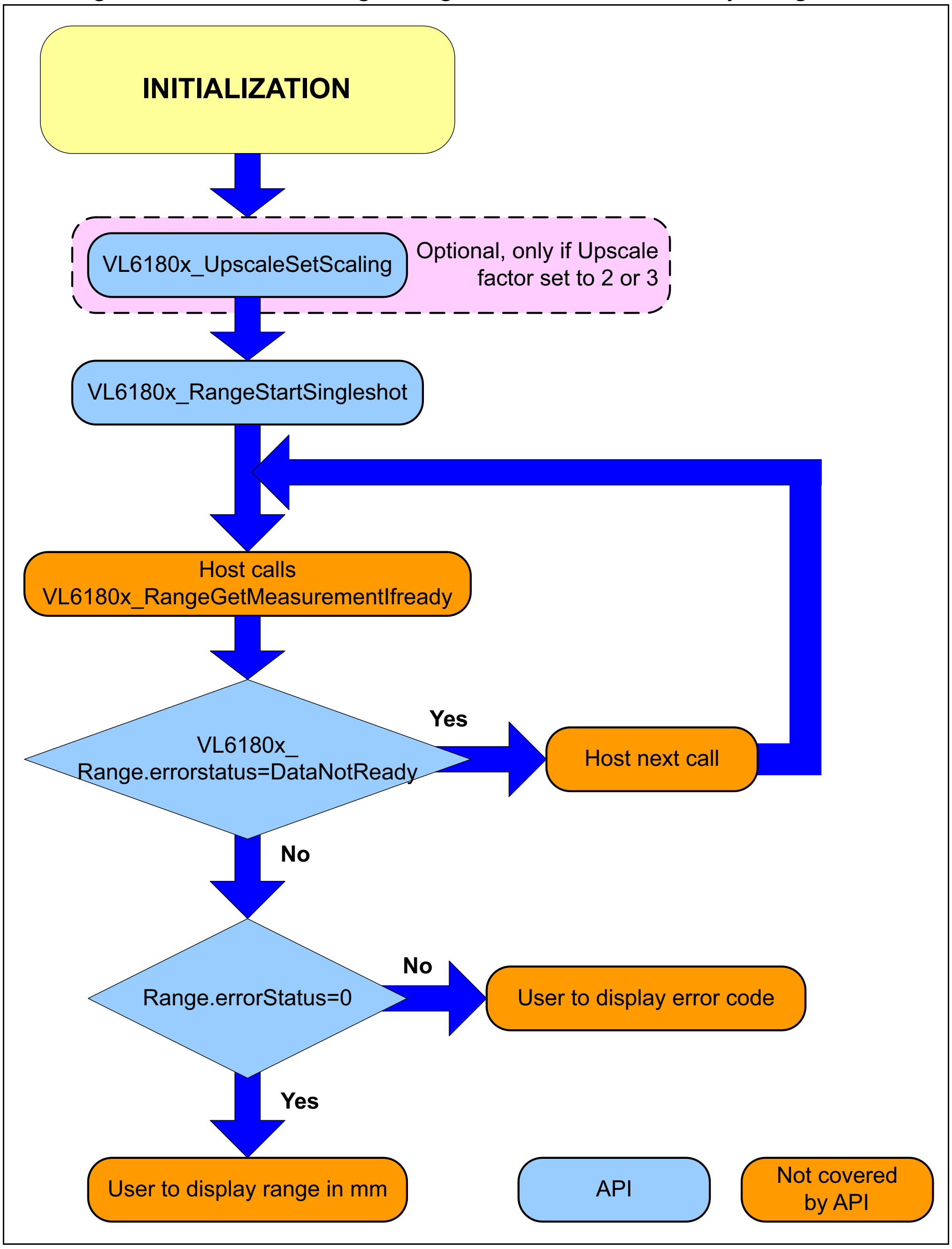


5.7 Single range measurement in non polling mode

This is suitable for applications where host CPU is triggered on an interrupt, not coming from VL6180X, to perform range measurement.

Figure 20 shows the flow for a single range measurement in non polling mode.

Figure 20. Flow for a single range measurement in non polling mode

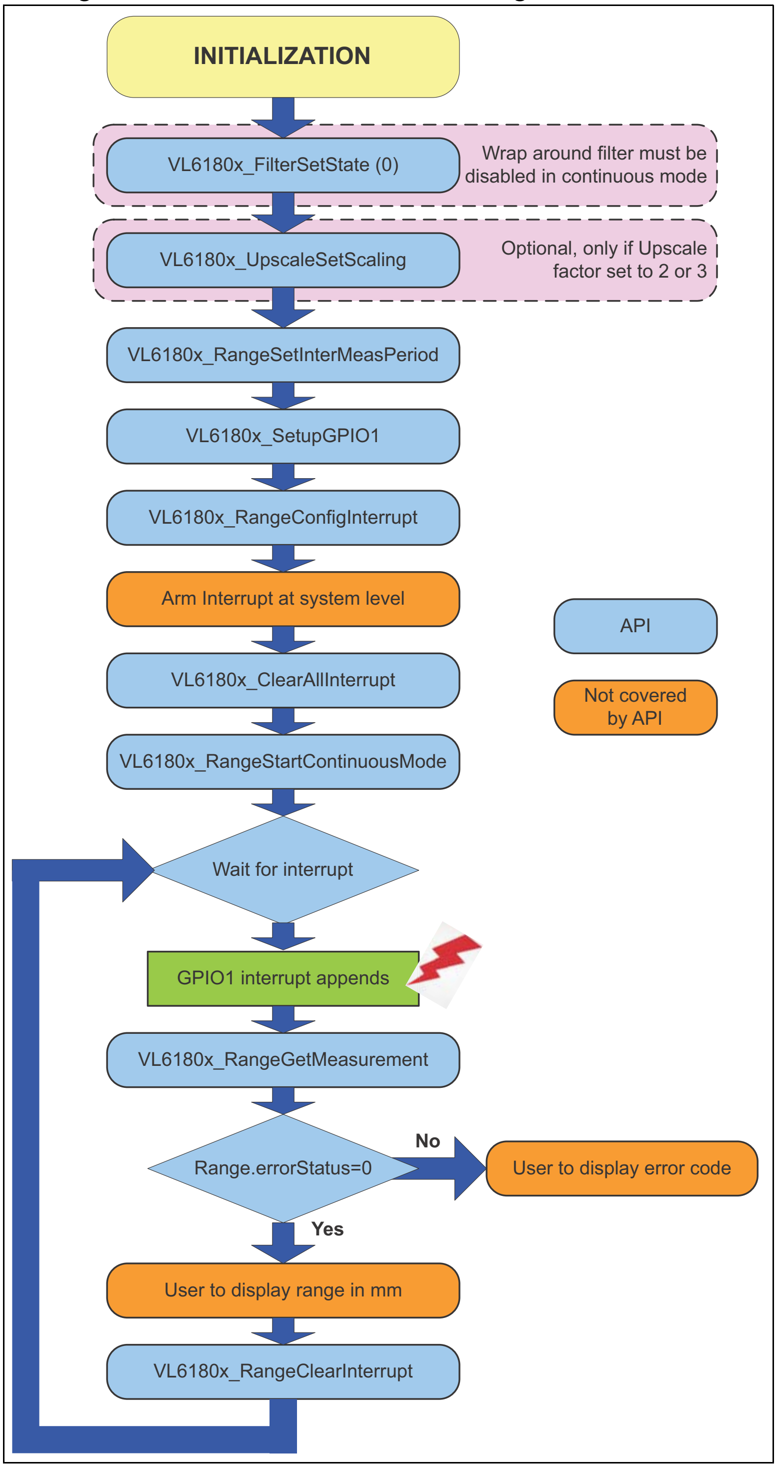


5.8 Continuous range measurement

In this mode the host is interrupted by VL6180X.

Figure 21 shows the flow for a continuous range measurement.

Figure 21. Flow for a continuous range measurement



5.9 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.

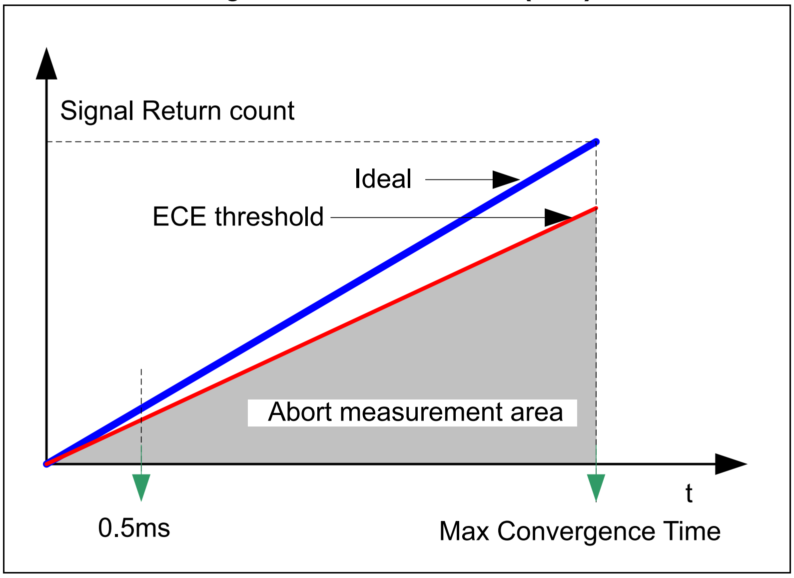
ECE works by calculating the rate of convergence 0.5ms after the measurement has been started. If the return count rate reported by the device is below the set ECE threshold, the measurement is aborted.

The ECE feature is enabled through VL6180X_RangeSetEceState function.

The ECE return count rate threshold is set through VL6180X_RangeSetEceFactor function.

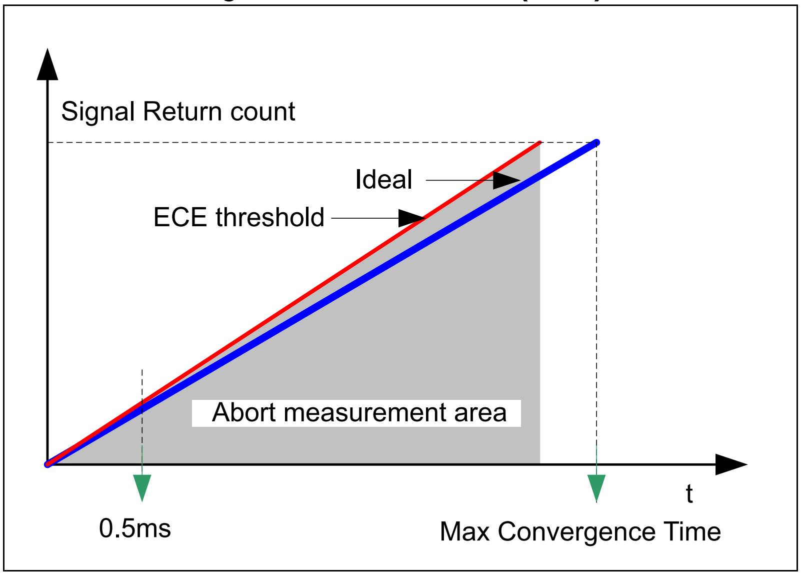
Figure 22 and *Figure 23* show examples with a threshold ratio of 0.8 and of 1.1.

Figure 22. 0.8 ECE factor (80%)



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

Figure 23. 1.1 ECE factor (110%)



110% ECE factor - if the return signal count is 110% of the ideal convergence rate after 0.5ms 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.

Table 5 gives an example of the current consumption of the VL6180X without and with ECE feature enabled and for a range of inter-measurement periods.

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

Table 5. VL6180X 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

5.10 Wrap around filter (WAF)

Mirrors are high reflective targets and placed at a far distance, more than 60cm, from VL6180X they could still produce enough return signal to declare for VL6180X a valid target resulting 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 16 (see *Table 6*).

WAF function is enabled through ‘VL6180X_FilterSetState’.

5.11 Dmax

Dmax feature is enabled through VL6180x_DMaxSetState function.

When ambient light level increases, the max detection range (Dmax) decreases, so a target may not be detected by the VL6180X 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.

5.12 Range status error code information

Detailed explanation of error code information are given in “VL6180X API Integration Guide.pdf” on *www.st.com/vl6180x* in “Design Resources” page.

Table 6 gives a summary of the meaning of each range status error code.

Table 6. Status error code meaning

Code		Name	Comment
0	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.

Table 6. Status error code meaning (continued)

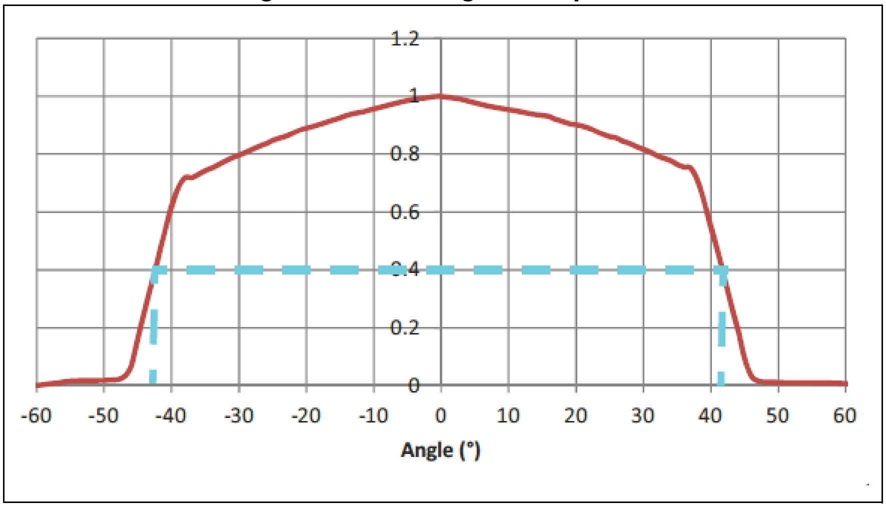
Code		Name	Comment
15	0b1111	Ranging_Algo_Overflow	Could happen if the target is detected at a distance higher than 55 cm for upscaling = 3, 20 cm for upscaling = 1. Error is due to internal variables/registers overflow.
16	0b10000	Filtered by post-processing	If Wrap Around Filter is enabled, this error gives the information that a bright target is detected between 60 and 120cm.
18	0b10010	DataNotReady	This error is returned by the VL6180X_RangeGetMeasurementIfReady API function when ranging sample data is not ready.

6 Ambient light sensor mode (ALS)

6.1 Overview

The VL6180X can measure the incoming ambient light over a wide dynamic range.
The ALS sensor uses a photopic filter in order to approximate the spectral response of the human eye.
The raw data output is proportional to the amount of light within the field of view during the integration time.
The VL6180X has a typical 42 degree field of view (half angle, 40% of peak) in both X and Y (see *Figure 24*).

Figure 24. ALS angular response



The ALS count is converted in lux by the VL6180x_AlsGetLux API function.
Lux is the standard unit of light intensity or measurement of the amount of perceived light in an area.
Table 7 shows some typical examples of lux values for different conditions.

Table 7. Typical lighting conditions

Illuminance (lux)	Scene description
0.0001	Moonless overcast night
0.002	Moonless clear night
0.27 -0.1	Full moon on clear night
1	Twilight
50	Typical family room lighting
80	Typical office / Hallway lighting
100	Dark overcast day
400	Sunrise or sunset on clear day
1000	Overcast day, typical TV studio lighting

Table 7. Typical lighting conditions

Illuminance (lux)	Scene description
10,000 - 25,000	Clear day, indirect sunlight
32,000 - 100,000	Direct sunlight

6.2 Ambient light sensor settings

6.2.1 Analog gain

Analog gain is selected to match the ALS dynamic range to the expected light range of the application and to compensate for the use of cover glass.

Analog gain is set through 'VL6180x_AlsSetAnalogueGain'

Table 8 gives the dynamic range without and with a 10% transmissive cover glass versus the analog gain values.

Table 8. ALS dynamic range(1)

Analog gain	Dynamic range (No cover glass)		Dynamic range (10% transmissive cover glass)	
	Min (Lux)(2)	Max (Lux)	Min (Lux)	Max(Lux)
1	3.2	20,800	32	> 100,000
1.25	2.56	16,640	25.6	> 100,000
1.67	1.93	12,530	19.3	> 100,000
2.5	1.28	8,320	12.8	83,200
5	0.64	4,160	6.4	41,600
10	0.32	2,080	3.2	20,800
20	0.16	1,040	1.6	10,400
40	0.08	520	0.8	5,200

1. ALS lux resolution 0.32 @ 100ms integrated period
2. Minimum ALS count 10

6.2.2 Integration time

The ambient light sensor works by counting photons over a fixed time period referred to as the integration time. The resulting output value is proportional to the amount of light sensed or photons received during the integration period.
VL6180X is set in the factory to match 0.32 lux per ALS count at an integration time of 100ms.
The 100ms integration time is optimal for most applications. It is recommended to adjust the analog gain setting for different light level applications rather than adjusting integration time.

When necessary, the integration time can be changed. For example, when the sample rate is required to be faster than the 100ms integration time allowed, decreasing the integration time will allow for faster sampling rates. Lowering the integration time will, however, increase the effect of the light flicker on the result. It is recommended to keep integration time in steps of 50ms to reduce the impact of light flicker.

Integration time value is set through 'VL6180x_AlsSetIntegration Period'.

6.3 Cover glass calibration

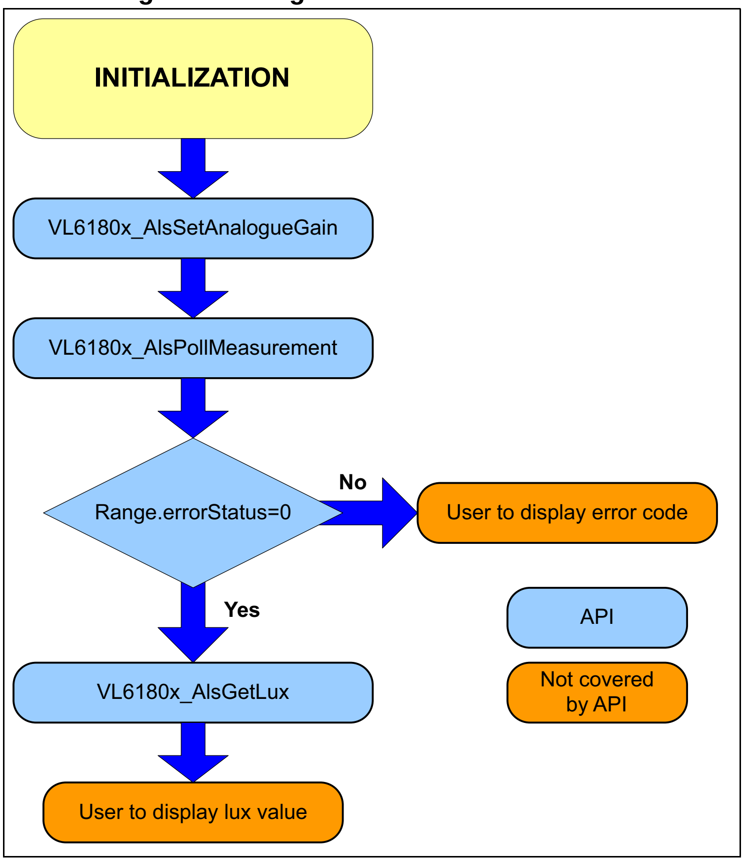
The use of cover glass in an application will block a percentage of light measured at the sensor. This reflected or absorbed light by the glass needs to be accounted for when converting the ALS count to lux.
The calibration of the cover glass is only needed when the cover glass is modified. Like ALS lux resolution, the calibration is only used in the conversion from ALS counts to lux and is not written to the VL6180X.
To calibrate for cover glass, the device should be placed under a stable light source similar in color temperature and intensity as the application. Multiple ALS measurements are taken both with and without the cover glass. The calibration factor is the ratio of the averaged results:

Cover Glass Cal Factor = Avg without glass / Avg with glass

6.4 Single ALS measurement in polling mode

Figure 25 shows the flow for a single ALS measurement in polling mode.

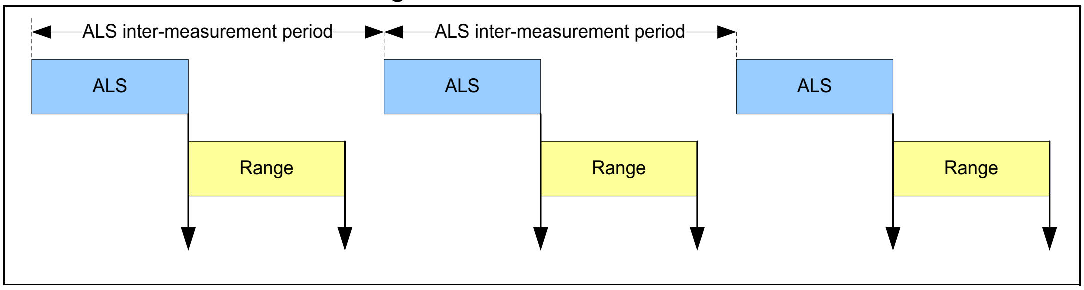
Figure 25. Single ALS measurement flow



7 Interleaved mode

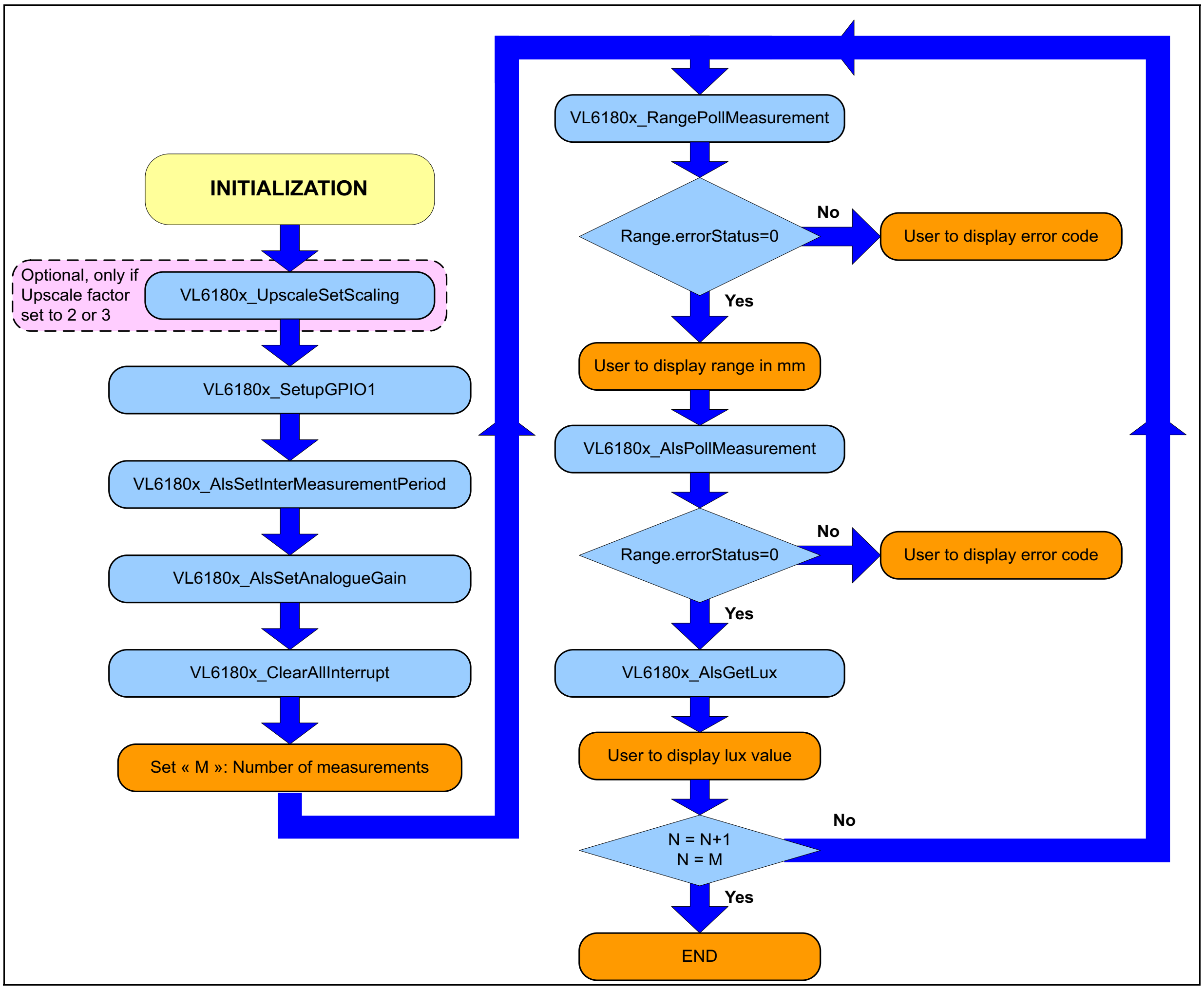
Interleaved mode is when a range measurement is immediately followed by an ALS measurement and repeated after an interval specified by the ALS inter-measurement period.

Figure 26. Interleaved mode



For interleaved mode, ALS feature must be enabled by setting VL6180x_ALS_SUPPORT.

Figure 27. Interleaved flow



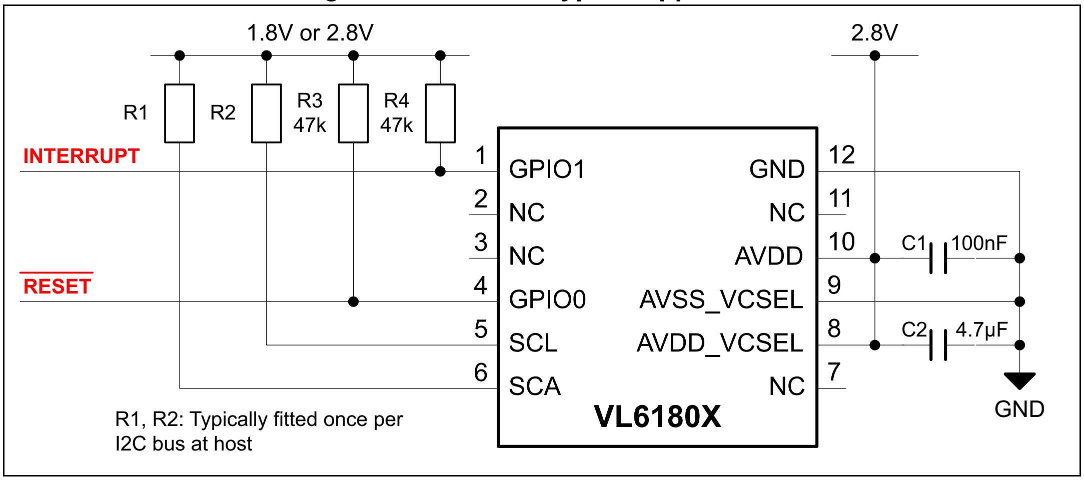
8 Multiple VL6180X application

This chapter shows how multiple VL6180X devices can be used on a board design while only using a single I2C interface to interact with all the devices.
Each VL6180X device has both a reset pin and an interrupt pin, which can be used to enable a multiple device setup.

8.1 VL6180Xs control management

Figure 28 is a typical example schematic using a VL6180X device. Since the VL6180X can have the I2C device address changed by doing an I2C write once it is booted, a separate reset pin would be needed to each VL6180X 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 by using multiple GPIO pins from the host on the board.

Figure 28. VL6180X typical application



Depending of the number of VL6180X 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 VL6180Xs.

8.2 VL6180Xs API management

See also “VL6180X_API_Integration Guide.pdf” on *www.st.com/VL6180X* in “Design Resources” page.

In vl6180x_platform.h API file

- Set VL6180x_SINGLE_DEVICE_DRIVER macro to 0 so that API implementation will be automatically adapted to a multi-device context.
- Define VL6180xDev_t type as a structure pointer holding any data required for multi-device management. A mandatory field is an instance of VL6180xDevData containing ST API private data.
- Then
  - define, “N” the number of VL6180X (Struct MyVL6180Dev_t BoardDevs[N]
  - Put all devices under reset
  - One after the other enable VL6180X and set their I2C address through VL6180x_SetI2CAddress (&BoardDevs[i], FinalI2cAddr)

9 Glossary

Table 9. Glossary

Term	Description
ALS	Ambient Light Sensor
API	Application Program Interface
ECE	Early Convergence Estimate
FOV	Field Of View
NVM	Non Volatile Memory
PCB	Printing Circuit Board
RAM	Random Access Memory
VCSEL	Vertical Cavity Surface Emitting Laser
WAF	Wrap Around Filter

10 Revision history

Table 10. Document revision history

Date	Revision	Changes
01-Dec-2015	1	Initial release.