AFBR-S50-XTK

Crosstalk Guide

Application Note

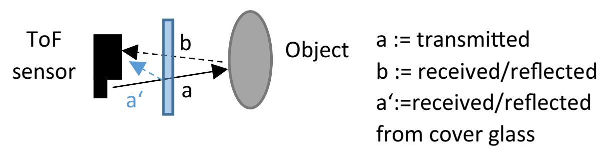
Table of Contents

Chapter 1: Introduction

Most outdoor applications require a proper ambient sealing of sensing electronics to ensure and preserve their functionality. To that end, a cover material protects the sensor not only against humidity and dust but also to prevent and mitigate mechanical damage.

Because the Broadcom® AFBR-S50 Time-of-Flight (ToF) sensors generate distance information by receiving and processing a reflected light pattern from an arbitrary surface, it can be presumed that adding any protective cover material within the optical path may impact the operation, as indicated in Figure 1.

Figure 1: ToF Operation Scheme Including Front Cover



This application note serves as a guide to correctly implement cover materials for achieving the best performance with AFBR-S50 ToF sensors.

Chapter 2: Theory of Crosstalk (Xtalk)

The AFBR-S50 ToF sensors are based on the indirect time-of-flight principle; consequently, any crosstalk (xtalk) source can affect each of the phase-shifted correlation samples acquired on the receiving side. Fortunately, this effect can be easily compensated by a mathematical subtraction of the xtalk from the vector of the correlation samples (for more information, refer to *AFBR-S50-BAS: Time-of-Flight Basics* – “Terms and Definitions”).

$$\vec{S}'(x,y) = \vec{S}(x,y) - \vec{S}_{xtalk}(x,y) \quad (1)$$

Since the Broadcom AFBR-S50 ToF sensors use four phase shifts, expression 1 can also written as follows.

$$\begin{pmatrix} S'_0(x,y) \\ S'_1(x,y) \\ S'_2(x,y) \\ S'_3(x,y) \end{pmatrix} = \begin{pmatrix} S_0(x,y) \\ S_1(x,y) \\ S_2(x,y) \\ S_3(x,y) \end{pmatrix} - \begin{pmatrix} S_{1,xtk}(x,y) \\ S_{2,xtk}(x,y) \\ S_{3,xtk}(x,y) \\ S_{4,xtk}(x,y) \end{pmatrix} \quad (2)$$

With $s = S_2 - S_0$ and $c = S_3 - S_1$

$$\begin{pmatrix} c'(x,y) \\ s'(x,y) \end{pmatrix} = \begin{pmatrix} c(x,y) \\ s(x,y) \end{pmatrix} - \begin{pmatrix} c_{xtk}(x,y) \\ s_{xtk}(x,y) \end{pmatrix} \quad (3)$$

Parameters:

$\vec{S}'(x,y) = (S'_0(x,y),\ S'_1(x,y),\ S'_2(x,y),\ S'_3(x,y))$

The corrected sample vector for pixel (x,y) and its components, S'_i [LSB].

$\vec{S}(x,y) = (S_0(x,y),\ S_1(x,y),\ S_2(x,y),\ S_3(x,y))$

The raw correlation sample vector from pixel (x,y) and its components, S_i [LSB].

$\vec{S}_{xlk}(x,y) = (S_{0,\mathrm{xtlk}}(x,y),\ S_{1,\mathrm{xtlk}}(x,y),\ S_{2,\mathrm{xtlk}}(x,y),\ S_{3,\mathrm{xtlk}}(x,y))$

The crosstalk vector for pixel (x,y) and its components, $S_{i,xtk}$ [LSB].

The crosstalk calibration is applied right after sample normalization, before any other compensation.

Chapter 3: Crosstalk in the Application

3.1 Sources for Xtalk

Xtalk can be of an *electrical* nature and an *optical* nature with the following root causes or a combination of them:

- Improper buffering (PI-filter) of the supplies in a PCB design (electrical)
- Long leads to the sensor pins (electrical)
- Cover materials (glasses) in the optical paths of the sensor (optical)
- Usage of narrow apertures to mechanically embed the sensor in a system (optical)

Even if aperture reflections can be categorized as optical xtalk, it is not part of the xtalk compensation routine. In general, apertures that are too narrow should be avoided by a proper mechanical design, considering the beam divergence and field-of-view (FoV) of a target AFBR-S50 module. FoV explanations can be found in all AFBR-S50 sensor data sheets.

3.2 When Is a Compensation of Xtalk Needed?

3.2.1 Electrical Xtalk Compensation

Refer to the "Application Circuit and Layout Recommendations" section from any AFBR-S50 iToF[1] data sheet to obtain PCB design and filter dimensioning requirements. A proper PCB design with passive SMD components is strongly recommended. Conversely, a voltage supply network consisting of loosened wires and discrete, passive components should be avoided.

3.2.2 Optical Xtalk Compensation

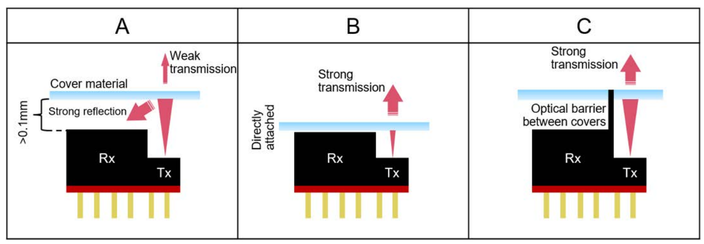
In general, a cover material implementation should meet the following requirements.

Table 1: Cover Material Requirements

Parameter	Value	Comment
Transmission (%)	> 90	@wavelength
Max. thickness (mm)	≤ 2	—
Distance to cover material (mm)	≤ 2	From housing edge (see Figure 2)

NOTE: Typically, the effect of the distance between the sensor and the cover is more dominant than the thickness of the cover material.

Figure 2: A – Xtalk Compensation Required; B & C – No Xtalk Compensation Required



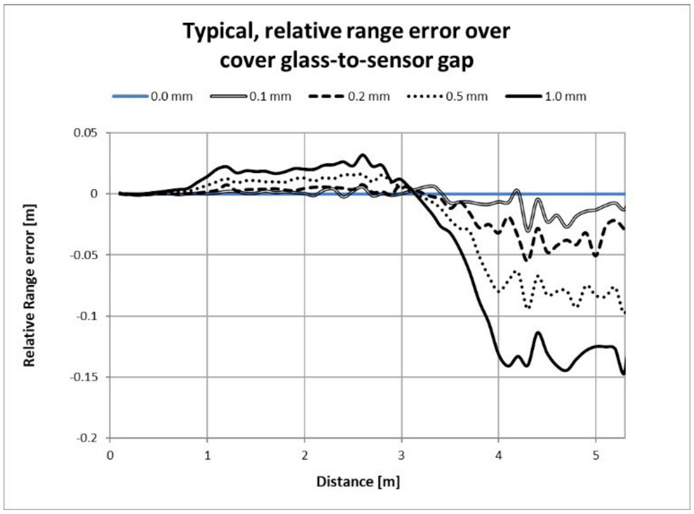
While scenario A requires xtalk compensation due to a strong reflex from Tx to Rx, scenario B shows a direct attachment of the cover to the sensor, which typically does not require compensation.

There is still some possible crosstalk inside the cover glass, which can be avoided by using two separate glasses for Rx and Tx with an optical barrier in between (see Figure 2C).

1. Short for *indirect ToF*; for more information, refer to *AFBR-S50-BAS: Time-of-Flight Basics*.

Figure 3 shows an indication on the high sensitivity of the distance between the sensor and the cover material and the impact on the range values. The blue zero line is the reference with a cover glass directly attached (0 mm) to the sensor housing.

Figure 3: Relative Range Error vs Cover-to-Sensor Distance (Uncalibrated Crosstalk)



NOTE: The relative errors shown in Figure 3 can be even higher depending on the target remission, sensor, and cover material type.

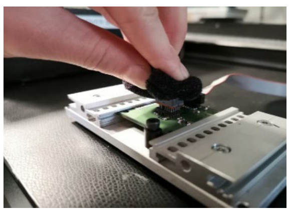
Chapter 4: Calibration Steps

The following steps must be carried out to properly calibrate xtalk in an application.

4.1 Step 1 – Electrical Xtalk Measurement (Blinded)

Figure 4 shows how to cover (blind) the sensor to measure the amount of electrical xtalk. Optimum materials avoid any transmission of the transmitter's light to the receiver.

Figure 4: Covering (Blinding) Sensor for an Electrical Xtalk Measurement: Blocking Light from Entering the Rx Lens



NOTE: This step shall be executed with the final application PCB. Blinding material could be any black foam, black silicon glove, or similar.

4.2 Step 2 – Electrical + Optical Xtalk Measurement (without Cover)

The intention of this step is to measure the magnitude of reflection from the background. An optimum condition would be a measurement against the clear sky at night or a target with 0% remission. However, this is hardly possible in reality and factory premises; therefore it is suggested to use the highest absorbing material. Section 8.6 includes a collection of suitable materials.

As can be seen in Section 4.4, this amount is subsequently subtracted from the measurement with the cover glass to erase the background impact.

Figure 5: Example of an Optical Sink to Simulate a Measurement Against Infinity

4.3 Step 3 – Electrical + Optical Xtalk Measurement (with Cover Glass)

The distance between the cover glass and the sensor is crucial for the amount of generated xtalk. Thus, it is important to use the same distance during the calibration procedure as in the final application.

Figure 6: Cover Glass Placement in Front of the Sensor

4.4 Step 4 – Calculation of Total Xtalk

After performing the first three steps, the total amount of xtalk can be calculated as follows:

$$\vec{S}_{xtalk}(total) = \vec{S}_{xtalk}(with\ glass) - T_{glass} \times \left(\vec{S}_{xtalk}(without\ glass) - \vec{S}_{xtalk}(blinded)\right)$$

T_{glass} refers to the transmission or transmittance coefficient of the used cover glass. Its dependency on the (emitted laser) wavelength is usually given in the data sheet of the cover glass vendor. A typical value (including Fresnel reflection losses) is ~0.92 for standard glass (for example, BK7, gorilla glass, and PMMA [acrylic] glass).

NOTE: Even though an application might not require any cover material, it might be beneficial to perform electrical xtalk compensation (Step 1 only), especially when the highest distance accuracy is necessary up to the maximum specified distance.

Chapter 5: Crosstalk Measurement Procedure

In practice, a complete xtalk measurement, including optical xtalk, requires the following setup:

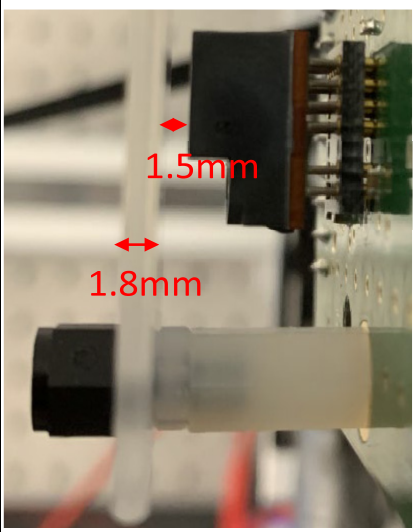
- Application PCB including AFBR-S50 sensor (measurement of PCB-specific electrical xtalk)
- Housing or fixture including cover material (for example glass) with a fixed distance to sensor
- Low reflective target as the background
- Sufficient space between the sensor and the target to ensure enough signal attenuation (in this example, 6m of space is used)
- (Optional) IR-camera to verify that the spot completely hits the target

It is important that the beam spot is well centered and completely reflected by the target. Since the beam spot is not sharply defined but has approximately an exponential decay at the rim, it is useful to have at least a factor-of-two margin (for example, for a beam spot size of 35 cm, use a target width of 70 cm). Also avoid any reflexes from the side and background walls as far as possible; a dark wall coverage is always beneficial.

5.1 Test Setup

<table>
<tr><th>Sensor with Cover Glass</th><th>Background Target</th></tr>
<tr><td>

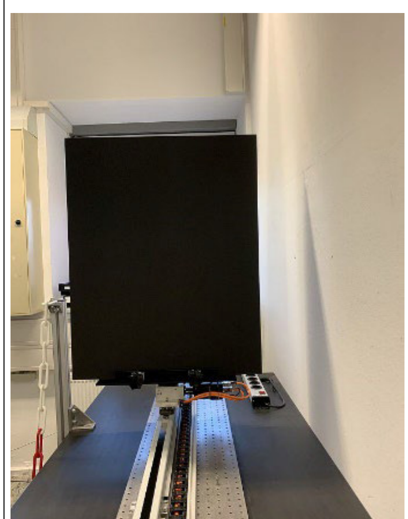
Figure 7: DUT Sensor with Cover Glass



Cover material – Gorilla glass
Transmittance @ 850 nm = 92%
Glass thickness = 1.8 mm
Sensor-to-glass gap = 1.5 mm
AFBR-S50LV85D ToF sensor

</td><td>

Figure 8: Black Target as Background



Dimensions:
Target size: 70 cm × 90 cm
Distance up to 6m
Target reflectivity: 4%

</td></tr>
</table>

NOTE: For proper beam alignment to the background target, consider laser beam divergence and, if existing, the laser beam squint angle. Refer to the Time-of-Flight Basics application note, Section 3.4. Unless otherwise mentioned, all subsequent measurements refer to the specifications above.

5.2 Crosstalk Measurement via the AFBR-S50 Explorer (Step-by-Step)

Using the GUI of the AFBR-S50 Explorer allows you to intuitively explore the *effect of any target and cover material without a single line of coding*.

5.2.1 Prerequisites

1. Software Development Kit (SDK) v1.4 or greater including the AFBR-S50 Explorer (download link: https://docs.broadcom.com/docs/12398582)
2. Sensor soldered to the following:
   – An AFBR-S50 Evaluation Kit adapter PCB
   – Your own hardware with a ported Explorer app (see https://broadcom.github.io/AFBR-S50-API/explorer_app.html)
3. The supplemental zip file `AFBR-S50-XTK.zip` from the AFBR-S50-API GitHub repository (Release section)
4. Windows operating system (the AFBR-S50 Explorer is based on C# libraries)
5. MS Excel to open and run the xtalk calculator
6. Any text editor (notepad, notepad++, and so on)

The following are 3D plots of the AFBR-S50 Explorer contrasting the amplitude distribution over the pixel array *without* cover glass (left) and *with* cover glass (not calibrated) (right).

Figure 9: Without Cover Glass – Typical Amplitude Distribution with a Peak on the Golden Pixel

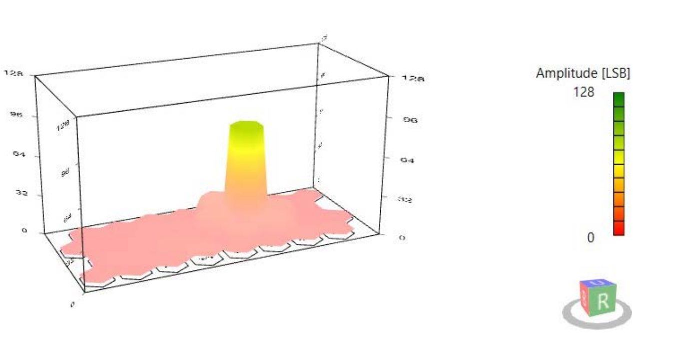


Figure 10: With Cover Glass (Uncalibrated) – Parallax Shifted Amplitude Distribution due to a Cover Glass in Front of the Sensor

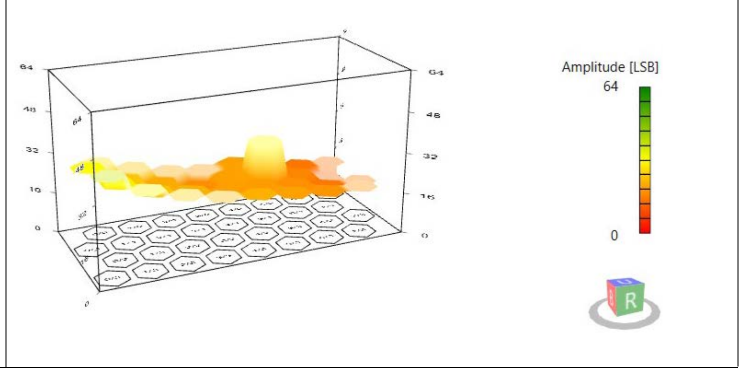


Figure 11: AFBR-S50 Explorer – 1D Range Result *without* Cover Glass

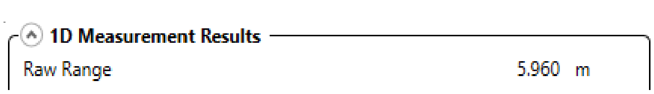


Figure 12: AFBR-S50 Explorer – 1D Range Result *with* Cover Glass (Uncalibrated)

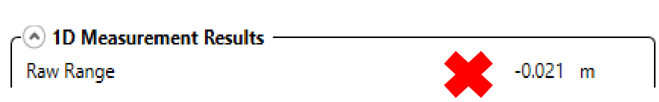


Figure 12 reveals the high signal amplitudes on passive pixels[2] and the associated wrong (binned) 1D distance value, both of which are caused by the cover glass.

2. Pixels that are not within the specific focus area of the laser (refer to *AFBR-S50-BAS: Time-of-Flight Basics* – Section 3.4.4).

5.2.2 Step 1 – AFBR-S50 Explorer

For more information on the setup, see Section 4.1.

The following figures show the amplitude differences of an AFBR-S50MV85G without cover material (Figure 13) and with cover material (Figure 14).

Figure 13: AFBR-S50 Explorer – Typical Amplitude Distribution of an AFBR-S50MV85G (Uncovered)

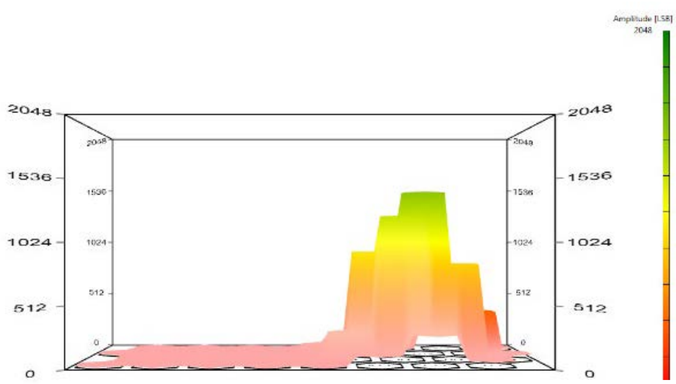
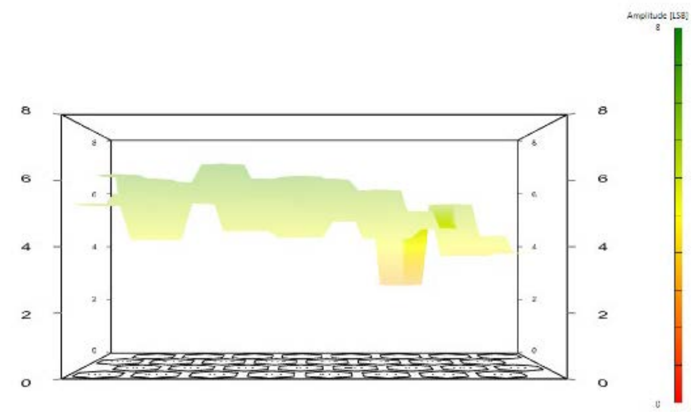


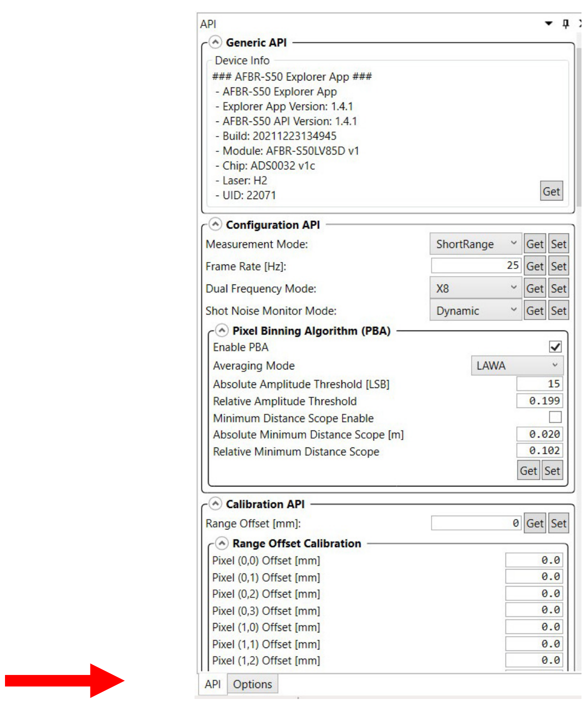
Figure 14: AFBR-S50 Explorer – Typical Amplitude Distribution of an AFBR-S50MV85G (Covered)



NOTE: Extended range module types such as the AFBR-S50LX85D might have higher values than 10 LSB.

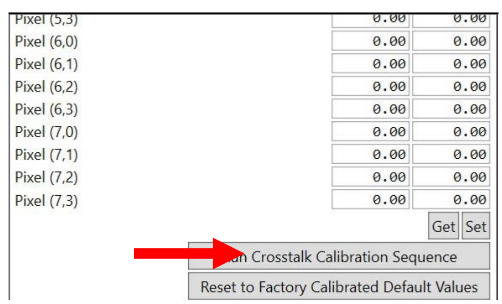
1. Open the AFBR-S50 Explorer, and go to the **API** tab.

Figure 15: AFBR-S50 Explorer – API Tab



2. Press the **Run Crosstalk Calibration Sequence** button.

Figure 16: AFBR-S50 Explorer – Run Crosstalk Calibration Sequence Button

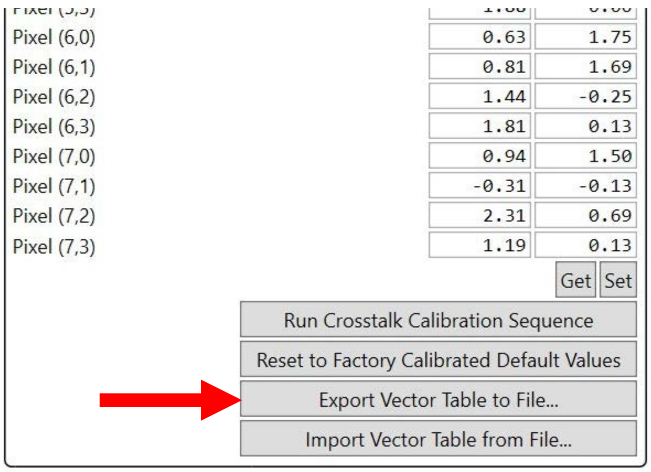


TIP:

- Repeat the measurement several times to ensure that the blinding material is properly in place. The values should not deviate more than ± 0.5 LSB over all measurement iterations.
- Watch the amplitude 3D chart when placing the blinding material for stable low values.

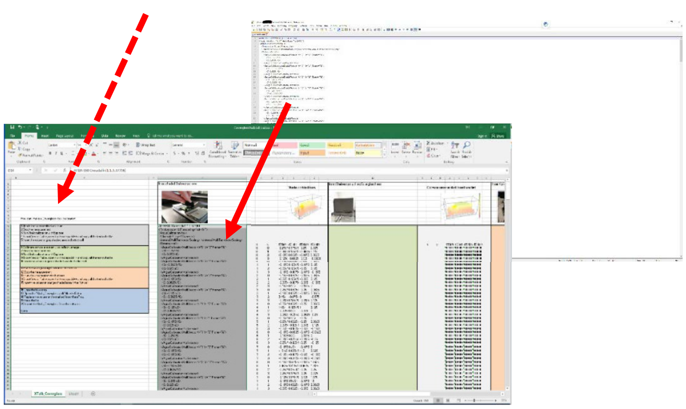
3. Press the **Export Vector Table to File** button and save it.

Figure 17: AFBR-S50 Explorer – Export Crosstalk Vector Table



4. Open and copy all the content from the previously saved text file into the respective column of the `CoverglassXtalkcalibration_template` Excel file. Find further explanations in the Excel template.

Figure 18: AFBR-S50 Explorer – Include Xtalk to Crosstalk Excel Template



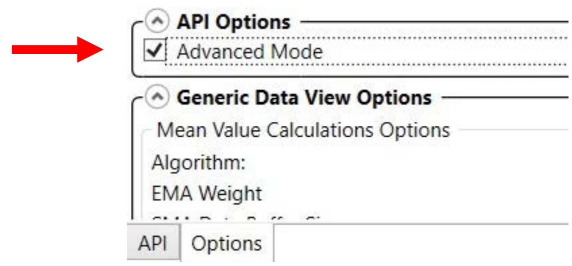
NOTE: The Excel template file is included in the `AFBR-S50-XTK.zip` file available under https://github.com/Broadcom/AFBR-S50-API/releases.

For more information on the setup, see Section 4.2.

5.2.3 Step 2 – AFBR-S50 Explorer

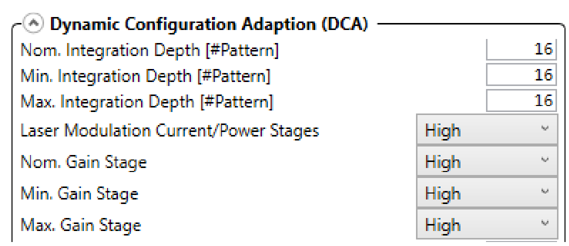
1. Enable the API Advanced Mode in the options tab.

Figure 19: AFBR-S50 Explorer – API Options – Advanced Mode



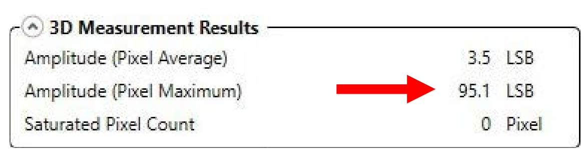
2. Under DCA in the API tab, set the integration depth, laser modulation current, and gain to maximum as shown in Figure 20.

Figure 20: AFBR-S50 Explorer – Maximum DCA Settings



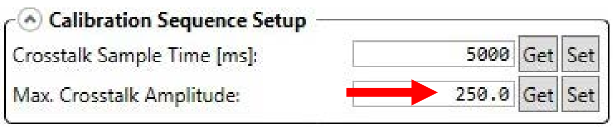
3. Aim the sensor to the low reflective target, and monitor the maximum amplitude in the summary tab.

Figure 21: AFBR-S50 Explorer – Maximum Pixel Amplitude



4. If needed, adapt the maximum crosstalk amplitude at the bottom of the API tab.

Figure 22: AFBR-S50 Explorer – Maximum Crosstalk Amplitude



- It is recommended to use the measured **Amplitude (Pixel Maximum)** at least *x 2* as a new threshold for the maximum crosstalk amplitude. In this example, 2.5 was used.
- ~100 LSB × 2.5 = 250 LSB.

NOTE: For AFBR-S50 sensor types with a typical beam divergence smaller than 4 degrees, it is recommended to not have a higher maximum amplitude than 100 LSB. For types with a typical beam divergence equal to or bigger than 4 degrees, it is recommended to have a maximum amplitude of around 10 LSB for optimized optical xtalk compensation.

5. Repeat steps 1 to 4 in Section 5.2.2.

NOTE: If the measured xtalk amplitude exceeds the maximum crosstalk amplitude, error -112 is raised by the device software (see Section 8.8). In such a case, make sure to have a properly placed blinding material.

5.2.4 Step 3 – AFBR-S50 Explorer

For more information on the setup, see Section 4.3.

After placing the cover glass:

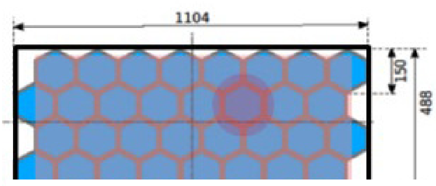
1. Repeat steps 1 to 4 from Section 5.2.2.
2. If necessary due to error -112, repeat step 1 from Section 5.2.3.

5.2.5 Step 4 – AFBR-S50 Explorer

1. Interpolate the *dominant* peaks of active pixels.

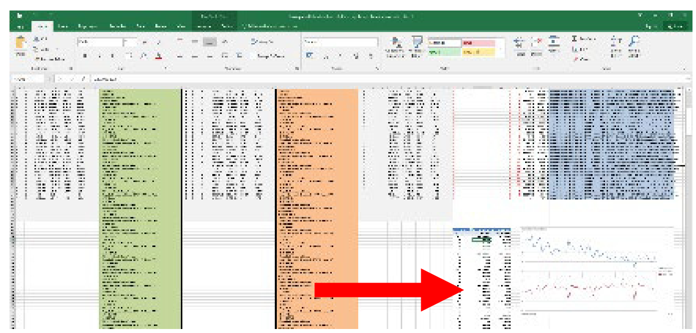
- This is only necessary for types that have only one to three active pixels or a typical beam divergence[3] of less than 4 degrees.
- Typically, the golden pixel is on position 5/1.

Figure 23: Golden Pixel Position



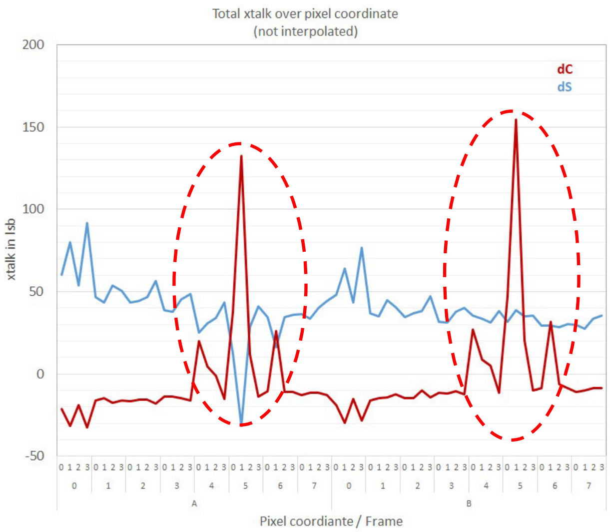
After inserting all three xtalk tables into the Excel template, the total xtalk is automatically shown in the blue area. Next, update the chart below the blue area. Right-click the pivot table and click **refresh**.

Figure 24: Updating the Xtalk Chart



3. The specification can be found in every ToF sensor data sheet under the “Optical and Sensor Characteristics” section.

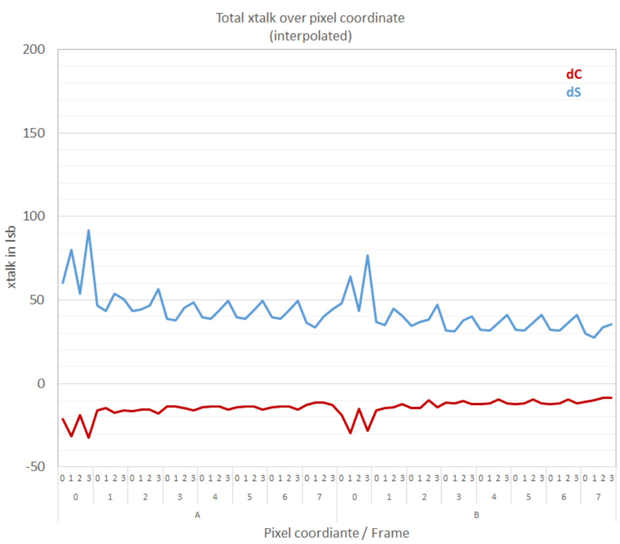
Figure 25: Xtalk Chart with Peaks at the Golden Pixel Position 5/1



The dashed, red circles in the chart highlight the peaks of the active pixels that need to be interpolated.

NOTE: The peaks may not be visible if the used optical sink is sufficient enough to achieve amplitudes less than 20 LSB.

Figure 26: Xtalk Chart with Interpolated Xtalk Values Around the Golden Pixel Position 5/1

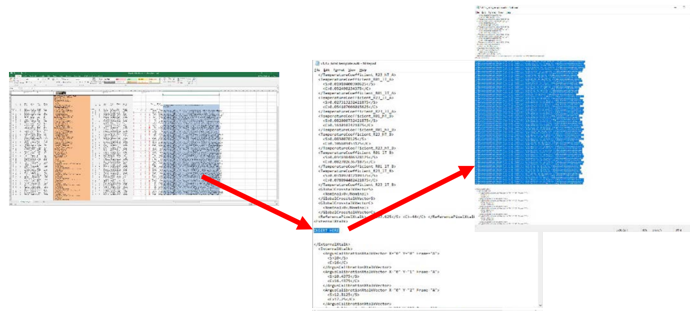


A simple averaging between the xtalk values from columns 2, 3, and 7 is performed and copied to the active pixel columns 4, 5, and 6 for all dS and dC as well as A-frame and B-frame. Figure 26 shows the result.

2. Now, copy and paste all the external Xtalk data starting with `<ArgusCalibrationXtalkVector`… until `</ArgusCalibrationXtalkVector>` (that is, from Frame A x=0, y=0 to Frame B x=7, y=3) into a new text file and save it with the `*.axtk` extension, for example, `total.axtk`.

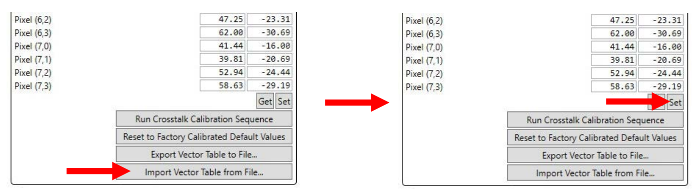
NOTE: A template for the `total.axtk` file is included in the `AFBR-S50-XTK.zip` file available under https://github.com/Broadcom/AFBR-S50-API/releases.

Figure 27: Create and Save the Total Xtalk File



3. Import the total xtalk file (`total.axtk`) into the AFBR-S50 Explorer by clicking the **Import Vector Table from File** button. Next, press the **Set** button to apply the new values.

Figure 28: Import and Set Total Xtalk Vector Table



Now, the cover glass is successfully compensated and range values are shown correctly.

Figure 29: Test Setup – Amplitudes with an Uncalibrated Cover Glass

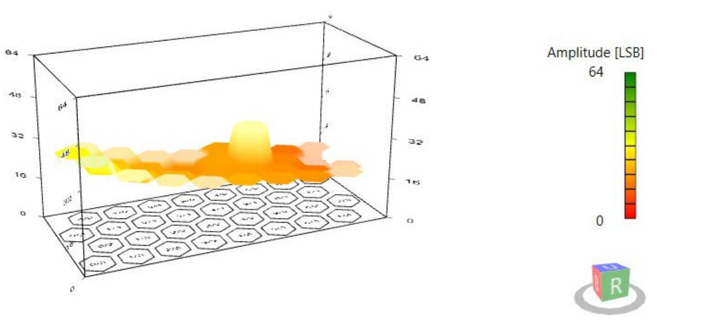


Figure 30: Test Setup – Amplitudes with a Calibrated Cover Glass

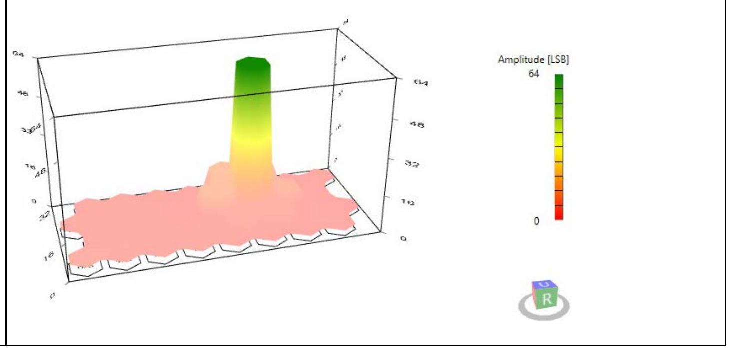


Figure 31: AFBR-S50 Explorer – 1D Range Result with Cover Glass – Uncalibrated

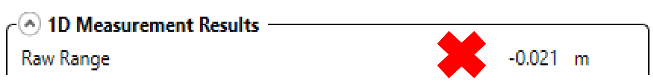
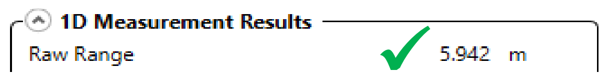


Figure 32: AFBR-S50 Explorer – 1D Range Result with Cover Glass – Calibrated



NOTE: If the cover glass is removed with xtalk values still being applied, the range values will show wrong distances.

5.3 Crosstalk Measurement via API

While the previous section explained xtalk calibration with the Explorer application, this section guides you through the API implementation by running the Example Application C-code, which is more likely to be carried out in a real application. The following is an overview about available API functions, which are also explained in the calibration module of the API reference manual (see https://broadcom.github.io/AFBR-S50-API/group__argus__cal.html).

NOTE: The functions listed in Table 2 are based on SDK v1.4.x and might be slightly different in other revisions.

Table 2: API – Overview of Crosstalk Functions

Argus_ExecuteXtalkCalibrationSequence	Triggers a crosstalk vector calibration measurement sequence.
Argus_SetCalibrationCrosstalkVectorTable	Sets the custom crosstalk vector table.
Argus_GetCalibrationCrosstalkVectorTable	Gets the custom crosstalk vector table.
Argus_ResetCalibrationCrosstalkVectorTable	Resets the crosstalk vector table to factory defaults (all 0s).
Argus_SetCalibrationCrosstalkSequenceSampleTime	Sets the measurement sample acquisition time for executing the crosstalk calibration sequence and generating the crosstalk data.
Argus_GetCalibrationCrosstalkSequenceSampleTime	Gets the measurement sample acquisition time for executing the crosstalk calibration sequence and generating the crosstalk data.
Argus_SetCalibrationCrosstalkSequenceAmplitudeThreshold	Sets the maximum amplitude threshold for the crosstalk calibration sequence. This is only a sanity check to avoid missing cover material. Increase the value to avoid the calibration failing due to amplitudes that are too high.
Argus_GetCalibrationCrosstalkSequenceAmplitudeThreshold	Gets the maximum amplitude threshold for the crosstalk calibration sequence.

To ease the calibration process with the API, the AFBR-S50-API GitHub repository offers the following supplemental files:

- `argus_xtalk_cal_cli.c`
- `argus_xtalk_cal_cli.h`

Enabling the xtalk calibration in the example files preprocessor definitions enables the step-by-step xtalk calibration guide via a terminal (Termite, PuTTy, and so on).

5.3.1 Prerequisites

1. Successfully porting the API to the target application microcontroller. All HAL tests were passed.
2. Successfully porting the UART (Tx and Rx) interface. Enables communication via the terminal prompt.
3. (Optional) Successfully porting the nonvolatile-memory (NVM) such as the MCU on-chip flash.

The following steps are necessary to start the step-by-step guide:

1. Open your IDE with your AFBR-S50 example project. Select the `example.h` and set the preprocessor definition of `RUN_XTALK_CALIBRATION` to `1`.

Figure 33: Xtalk Guide – Preprocessor Definition

```
/*! Selector for XTALK calibration demo:
 *  - 0: no XTALK calibration is executed.
 *  - 1: XTALK calibration is executed before any API code is executed. */
#ifndef RUN_XTALK_CALIBRATION
#define RUN_XTALK_CALIBRATION 1
#endif
```

2. Reflash the MCU, start a terminal application via UART, and check the prompt for the following content.

Figure 34: Xtalk Calibration – Start Screen

```
AFBR-S50 Crosstalk Calibration

##### AFBR-S50 API - Simple Example ##########################
  API Version: v1.4.4
  Chip ID:     33698
  Module:      AFBR-S50MV85G (v3)
##############################################################


####################################################################################
##### AFBR-S50 API - Interactive Xtalk Calibration Guide    ########################
####################################################################################

This interactive procedure guides you through the steps needed
to compensate the application specific xtalk. A dominant source of
xtalk could be the usage of cover glasses. Please read application
note AFBR-S50-XTK-AN1xx to get more information.

####################################################################################

------------------------------------------------------------------------------------
Enter 's' to start the step-by-step xtalk calibration guide        - or -
Enter 'c' to run a continuous measurement showing binned results   - or -
Enter 'x' to read the xtalk calibration table from memory          - or -
Enter 'o' to get an overview about other commands.
```

NOTE: The calibration guide uses floating point output for simplicity.

TIP: If you do not see the preprocessor definition of the xtalk calibration, make sure that you get the latest source files from the AFBR-S50-API repository.

What Is the Interactive Guide All About?

It is a collection of required calibration steps and a convenient printout of the results, and it contains additional tools to verify the results as you go through the steps. It features the following:

- Navigation through the guide by entering single-letter commands (UART Rx porting needed!)
- Hints and notes at the end of most command output

IMPORTANT: The interactive guide distinguishes between lowercase and uppercase letters.

Before starting the calibration, prove the following points:

1. Check the impact of the cover material by starting a continuous measurement:
   - At the maximum distance of the target application
   - With the expected minimum object remission

   Press `c` (lowercase letter) to start a continuous measurement over one second.

Figure 35: Example Measurement with Cover Glass Prior to Calibration

```
AFBR-S50 Crosstalk Calibration

--Continuous Measurement over 1 second - Binned Results--
#1  Range:  5783 mm;  Amplitude:   20 LSB;  Quality:   1;  Status: -110;  AnaInt: 16 LSB;  DigInt: 21 LSB;  Mod Curr: 54 mA;  Gain: B
#2  Range:  5783 mm;  Amplitude:   22 LSB;  Quality:   1;  Status: -110;  AnaInt: 8 LSB;  DigInt: 42 LSB;  Mod Curr: 54 mA;  Gain: 4B
#3  Range:    -8 mm;  Amplitude:   19 LSB;  Quality: 100;  Status:    0;  AnaInt: 8 LSB;  DigInt: 42 LSB;  Mod Curr: 54 mA;  Gain: 4B
#4  Range:    -9 mm;  Amplitude:   27 LSB;  Quality: 100;  Status:    0;  AnaInt: 8 LSB;  DigInt: 42 LSB;  Mod Curr: 54 mA;  Gain: 4B
#5  Range:   -20 mm;  Amplitude:   26 LSB;  Quality: 100;  Status:    0;  AnaInt: 8 LSB;  DigInt: 42 LSB;  Mod Curr: 54 mA;  Gain: 4B
#6  Range:   -17 mm;  Amplitude:   26 LSB;  Quality: 100;  Status:    0;  AnaInt: 8 LSB;  DigInt: 42 LSB;  Mod Curr: 54 mA;  Gain: 4B
#7  Range:   -16 mm;  Amplitude:   26 LSB;  Quality: 100;  Status:    0;  AnaInt: 8 LSB;  DigInt: 42 LSB;  Mod Curr: 54 mA;  Gain: 4B
#8  Range:    -7 mm;  Amplitude:   35 LSB;  Quality: 100;  Status:    0;  AnaInt: 16 LSB;  DigInt: 21 LSB;  Mod Curr: 54 mA;  Gain: B
#9  Range:    14 mm;  Amplitude:   48 LSB;  Quality: 100;  Status:    0;  AnaInt: 16 LSB;  DigInt: 21 LSB;  Mod Curr: 54 mA;  Gain: B
#10  Range:     9 mm;  Amplitude:   49 LSB;  Quality: 100;  Status:    0;  AnaInt: 16 LSB;  DigInt: 21 LSB;  Mod Curr: 54 mA;  Gain:B
#11  Range:     3 mm;  Amplitude:   48 LSB;  Quality: 100;  Status:    0;  AnaInt: 16 LSB;  DigInt: 21 LSB;  Mod Curr: 54 mA;  Gain:B
#12  Range:    -4 mm;  Amplitude:   49 LSB;  Quality: 100;  Status:    0;  AnaInt: 16 LSB;  DigInt: 21 LSB;  Mod Curr: 54 mA;  Gain:B
#13  Range:    -3 mm;  Amplitude:   47 LSB;  Quality: 100;  Status:    0;  AnaInt: 16 LSB;  DigInt: 21 LSB;  Mod Curr: 54 mA;  Gain:B
#14  Range:    -8 mm;  Amplitude:   46 LSB;  Quality: 100;  Status:    0;  AnaInt: 16 LSB;  DigInt: 21 LSB;  Mod Curr: 54 mA;  Gain:B
#15  Range:   -13 mm;  Amplitude:   47 LSB;  Quality: 100;  Status:    0;  AnaInt: 16 LSB;  DigInt: 21 LSB;  Mod Curr: 54 mA;  Gain:B
#16  Range:   -11 mm;  Amplitude:   48 LSB;  Quality: 100;  Status:    0;  AnaInt: 16 LSB;  DigInt: 21 LSB;  Mod Curr: 54 mA;  Gain:B
#17  Range:   -12 mm;  Amplitude:   37 LSB;  Quality: 100;  Status:    0;  AnaInt: 16 LSB;  DigInt: 21 LSB;  Mod Curr: 54 mA;  Gain:B
#18  Range:     1 mm;  Amplitude:   47 LSB;  Quality: 100;  Status:    0;  AnaInt: 16 LSB;  DigInt: 21 LSB;  Mod Curr: 54 mA;  Gain:B
#19  Range:     5 mm;  Amplitude:   47 LSB;  Quality: 100;  Status:    0;  AnaInt: 16 LSB;  DigInt: 21 LSB;  Mod Curr: 54 mA;  Gain:B
#20  Range:   -12 mm;  Amplitude:   47 LSB;  Quality: 100;  Status:    0;  AnaInt: 16 LSB;  DigInt: 21 LSB;  Mod Curr: 54 mA;  Gain:B
#21  Range:    -6 mm;  Amplitude:   47 LSB;  Quality: 100;  Status:    0;  AnaInt: 16 LSB;  DigInt: 21 LSB;  Mod Curr: 54 mA;  Gain:B
#22  Range:    -6 mm;  Amplitude:   48 LSB;  Quality: 100;  Status:    0;  AnaInt: 16 LSB;  DigInt: 21 LSB;  Mod Curr: 54 mA;  Gain:B
#23  Range:    -4 mm;  Amplitude:   49 LSB;  Quality: 100;  Status:    0;  AnaInt: 16 LSB;  DigInt: 21 LSB;  Mod Curr: 54 mA;  Gain:B
#24  Range:    -9 mm;  Amplitude:   48 LSB;  Quality: 100;  Status:    0;  AnaInt: 16 LSB;  DigInt: 21 LSB;  Mod Curr: 54 mA;  Gain:B
#25  Range:   -15 mm;  Amplitude:   46 LSB;  Quality: 100;  Status:    0;  AnaInt: 16 LSB;  DigInt: 21 LSB;  Mod Curr: 54 mA;  Gain:B
#26  Range:    -9 mm;  Amplitude:   46 LSB;  Quality: 100;  Status:    0;  AnaInt: 16 LSB;  DigInt: 21 LSB;  Mod Curr: 54 mA;  Gain:B

------------------------------------------------------------------------------Info:
```

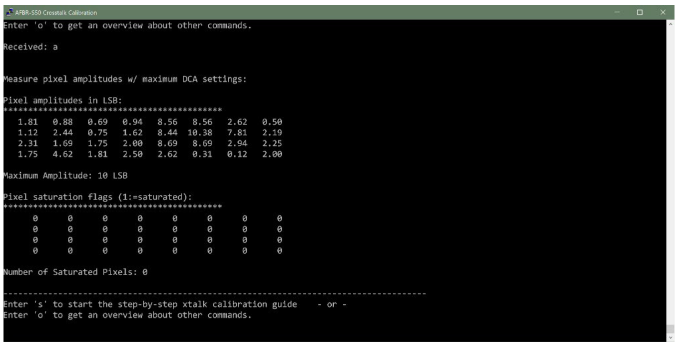
Assuming a target distance of 6m, the range results above indicate strong reflections from the cover glass because the range values are way too low.

NOTE: Calibration is needed!

2. Measure the maximum amplitude against the calibration target (optical sink) without cover glass.

Press `a` to start a measurement of the amplitude with maximized DCA settings.

Figure 36: Maximum Amplitude Check on the Background Target – Example with AFBR-S50MV85G (Typical Beam Divergence of 4°)



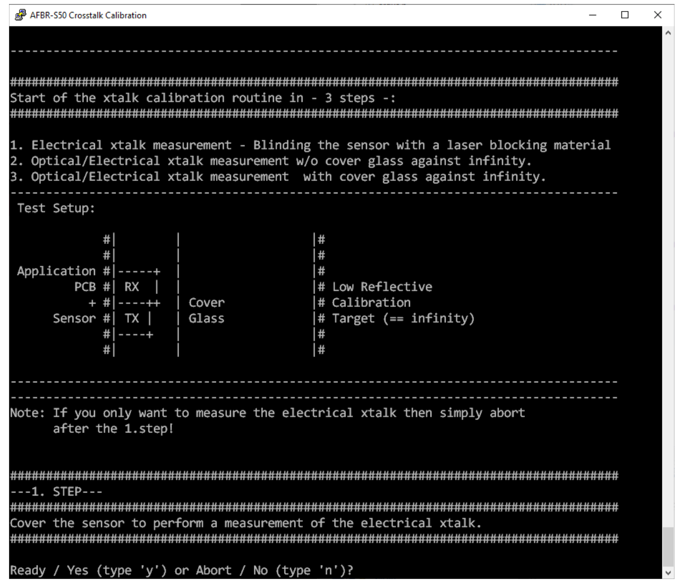
NOTE: For AFBR-S50 sensor types with a typical beam divergence smaller than 4 degrees, it is recommended to not have a higher maximum amplitude than 100 LSB (for those sensors, interpolation will be applied by default to further reduce the amplitude of the sink target). For types with a typical beam divergence equal to or bigger than 4 degrees, it is recommended to have a maximum amplitude of around 10 LSB for optimized optical xtalk compensation (no interpolation will be applied).

Figure 36 shows a maximum amplitude of 10 LSB with the AFBR-S50MV85G. *Calibration can be performed!*

5.3.2 Step 1 – AFBR-S50 API

Press `s` to start the interactive step-by-step xtalk calibration guide as shown in Figure 37.

Figure 37: Terminal – Start of Xtalk Calibration



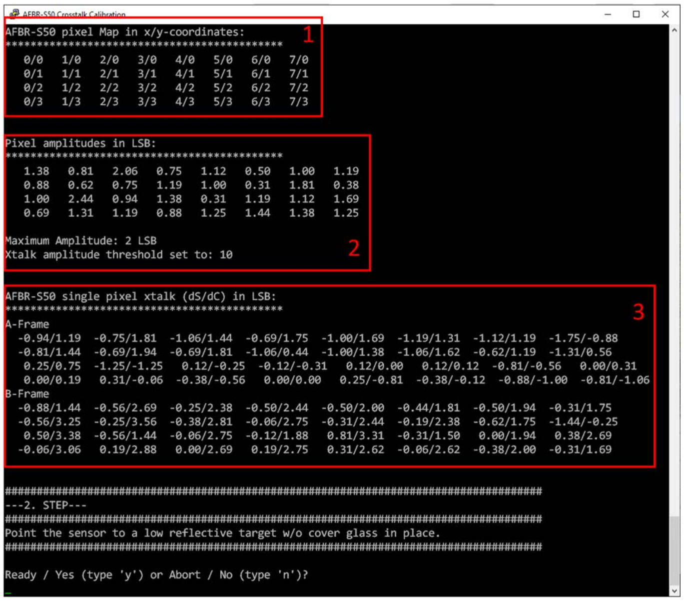
A small overview about the whole cover glass calibration procedure in three steps appears on the prompt.

From here, just follow the instructions.

Like in the previous section, the first step starts with the electrical xtalk measurement by blinding the module.

After confirming with `y`, keep the module *blinded* until the next screen appears.

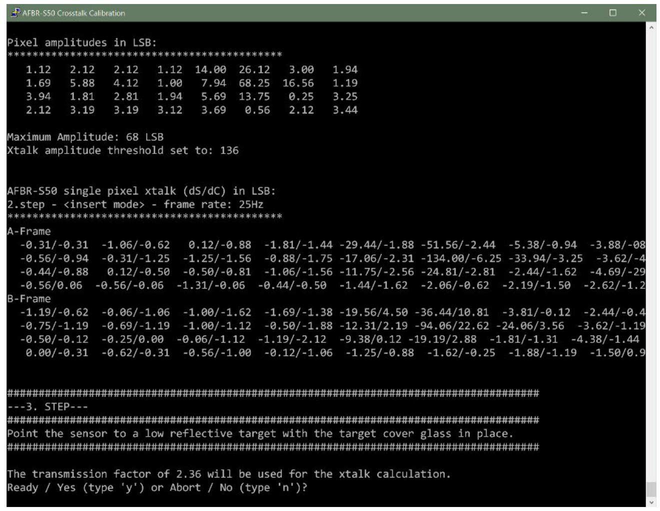
Figure 38: Terminal – Xtalk Calibration Step 1 Results Explanation



1. The first matrix serves as an overview for the pixel coordinates.
2. The second matrix lists the pixel amplitudes in LSB as floating numbers. They are measured with maximum integration energy[4] and define the *xtalk amplitude threshold*. This is similar to step 2 from Section 5.2.3, but is automated in the xtalk calibration guide.
3. The last two matrices are the xtalk values in LSB for both A and B frames. The "/" between the numbers subdivides dS (sinus) from dC (cosine) values.

5.3.3 Step 2 – AFBR-S50 API

Figure 39: Terminal – Xtalk Calibration Step 2



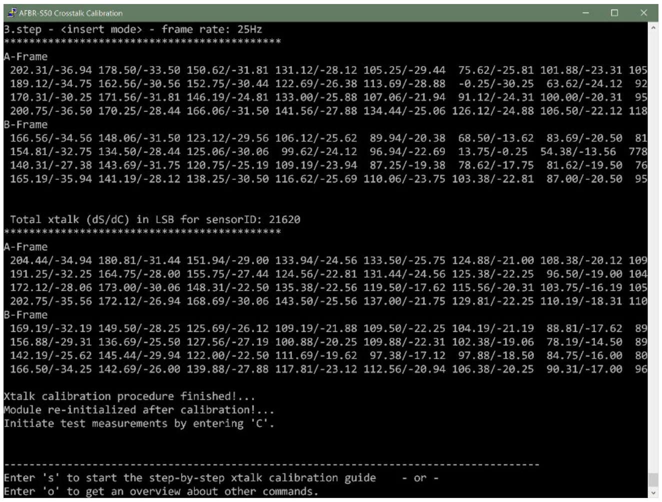
In the second step of the calibration, the xtalk amplitude threshold is defined by the maximum measured amplitude multiplied by a predefined factor (for more information, see step 4 in Section 5.2.3). By default, this factor is set to 2 -> 2 × 68 LSB = 136 LSB.

Next, position the cover glass in front of the sensor and confirm by pressing `y` to proceed with Step 3 and 4 – AFBR-S50 API.

4. Integration depth × laser modulation current × pixel gain

5.3.4 Step 3 and 4 – AFBR-S50 API

Figure 40: Terminal – Total Xtalk Calibration



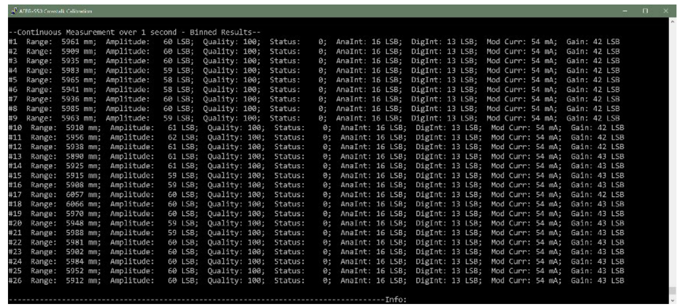
The measurement with the cover glass in front of the sensor might show an even higher amplitude compared to Step 2 – AFBR-S50 API. In such case, the interactive sequence automatically adapts the xtalk amplitude threshold accordingly.

After this last step, the total xtalk is calculated (see Section 4.4) and displayed together with the sensor ID. Since this example shows the results with the setup explained in Section 5.1, interpolation of the active pixels is needed, which is automatically performed in this guide.

NOTE: The xtalk calibration is finished. Now, start a measurement to check the effect of the calibration.

Enter `c` to start the measurement over 1 second (or more).

Figure 41: Terminal – Binned Distance Measurement



The distance values in Figure 41 state a range of 6000 mm, which matches the target range.

NOTE: The crosstalk calibration has been performed successfully!

Chapter 6: Productive Implementation of a Cover Glass Calibration

Thinking of a productive implementation may bring up the question on what is the best way to include the calibration in the process flow. In principle, there are two feasible scenarios.

	Pros	Cons
Scenario 1: Perform a calibration with each sensor built into a subsystem.[a]	■ Most accurate since subsystem-specific tolerances are included in the calibration.	■ More time consuming since the calibration is a process requiring three (manual) steps. ■ Automated calibration is difficult.
Scenario 2: Prior to production, perform a system characterization consisting of a reasonable number of subsystem samples, calculate their xtalk average, and use a hard-coded xtalk vector table.	■ Most time efficient since calibration is not included in the production process anymore.	■ Accuracy depends on the number of characterized and calibrated subsystems and may need to be repeated on new production lots.

a. A subsystem may consist of an application PCB and a mechanical enclosure including the cover material.

6.1 Scenario 1 – Saving Xtalk Data to Nonvolatile Memory

Nonvolatile memory (NVM) for xtalk data storage requires, for example, on-chip flash in the MCU or SD card. The AFBR-S50 API contains wrapper functions to read and write that NVM data and permanently save the calibrated xtalk data for each sensor in use. For further implementation details, refer to https://broadcom.github.io/AFBR-S50-API/group__argus__nvm.html.

The advantage of using this method is that the AFBR-S50 API takes care of handling the calibration data automatically. Only the implementation of the NVM HAL is required. But it might be hard to extract the obtained calibration data from memory, and flashing new firmware might also dismiss the present calibration values. See Scenario 2 to overcome those issues.

A full calibration supporting all operation modes consists of 20 signal components per pixel. A component contains the dC and dS crosstalk value (each in Q11.4 format) and is therefore equal to 4 bytes multiplied by 32 pixels equals 128 bytes. Dual frequency mode (DFM) is on by default and uses two different modulation frequencies that cause a doubled number of values per mode.

Table 3: Xtalk Data vs Measurement Mode

Mode	DFM	Number of Components	Bytes
ShortRange	X4	4	256
ShortRange	X8	4	256
ShortRange	Disabled	2	128
LongRange	X4	4	256
LongRange	X8	4	256
LongRange	Disabled	2	128
Total	—	20	1280

The component values are 16-bit signed fixed-point values in Q11.4 format, which is a signed fixed-point number format based on the 16-bit signed integer type with 11 integer and 4 fractional bits.

Example: Detecting objects up to 6m with all 32 pixels (for example, by using the AFBR-S50MV85I) would only require storing xtalk values for ShortRange for either DFM = x4 or DFM = x8 mode, which results in an NVM occupation of only 256 bytes. If there is a hard distance limit because of a floor or wall at 6m, the DFM can be disabled and the memory footprint reduces to 128 bytes.

6.2 Scenario 2 – Hard-Coding the Xtalk Vector Table

Another way of preserving the calibration data beyond reset is to hard-code the values into the source code, for example by creating a dedicated header file that contains the data. This approach offers more flexibility, making it possible to version-control the data and compile it into newer versions of the firmware easily.

A callback mechanism is implemented in the AFBR-S50 API that is called upon device initialization after obtaining the device information and before applying the final calibration data. See the `Argus_GetCrosstalkVectorTable_Callback()` function as defined in the `argus_api.h` header file. The function is implemented with the attribute weak in the API and can be overwritten by defining it in the application code.

To *hard-code* the calibrated xtalk vector table, it is necessary to convert it into C-code text containing all values in fixed-point format as shown in Figure 43.

Perform the following steps to add the table to your project.

1. Type `x` in the terminal to display the calibrated and stored xtalk vector table.

Figure 42: Terminal – Xtalk Vector Table

```
AFBR-S50 Crosstalk Calibration
Received: x

Print xtalk vector table stored in flash memory:

--------------------------------------------------------------------------

--------------------------------------------------------------------------
This is the xtalk table for the currently set 'Short Range' mode:

--------------------------------------------------------------------------
Frame   X   Y  xtalk-dS  xtalk-dC
--------------------------------------------------------------------------
    A   0   0     -0.69     -4.06
    A   0   1     -1.38     -4.06
    A   0   2      0.44     -1.19
    A   0   3      0.31     -1.62
    A   1   0     -0.25     -4.00
    A   1   1     -0.69     -3.94
    A   1   2      0.62     -0.44
    A   1   3      0.12     -1.38
    A   2   0     -0.75     -4.50
    A   2   1     -0.50     -3.62
    A   2   2      0.44     -0.56
    A   2   3      0.25     -0.81
    A   3   0     -0.44     -3.94
    A   3   1     -0.88     -4.12
    A   3   2      0.25     -1.12
    A   3   3     -0.44     -1.31
```

NOTE: You might see a different mode depending on module type and current settings.

2. Use an Excel or Python script to convert the values from the xtalk table (shown in orange) to a text 'xtalk->FrameA…' including the signed fixed-point number in Q11.4 format (shown as grey columns).

Figure 43: Excel Table for Conversion of Xtalk Values

Frame-Type	X	Y	xtalk-dS	xtalk-dC		Code for dS values (fixed-point)	Code for dC values (fixed-point)
0	0	0	29.6875	-17.188		xtalk->FrameA[0][0].dS = 475;	xtalk->FrameA[0][0].dC = -275;
0	0	1	30.1875	-17.063		xtalk->FrameA[0][1].dS = 483;	xtalk->FrameA[0][1].dC = -273;
0	0	2	29	-14.813		xtalk->FrameA[0][2].dS = 464;	xtalk->FrameA[0][2].dC = -237;
0	0	3	31.5625	-16.313		xtalk->FrameA[0][3].dS = 505;	xtalk->FrameA[0][3].dC = -261;
0	1	0	25.125	-15.625		xtalk->FrameA[1][0].dS = 402;	xtalk->FrameA[1][0].dC = -250;
0	1	1	26.0625	-14.875		xtalk->FrameA[1][1].dS = 417;	xtalk->FrameA[1][1].dC = -238;
0	1	2	28.4375	-12.188		xtalk->FrameA[1][2].dS = 455;	xtalk->FrameA[1][2].dC = -195;
0	1	3	30.75	-15.438		xtalk->FrameA[1][3].dS = 492;	xtalk->FrameA[1][3].dC = -247;
0	2	0	21.625	-13.875		xtalk->FrameA[2][0].dS = 346;	xtalk->FrameA[2][0].dC = -222;
0	2	1	20.1875	-12.938		xtalk->FrameA[2][1].dS = 323;	xtalk->FrameA[2][1].dC = -207;
0	2	2	23	-11.375		xtalk->FrameA[2][2].dS = 368;	xtalk->FrameA[2][2].dC = -182;
0	2	3	26.1875	-13.313		xtalk->FrameA[2][3].dS = 419;	xtalk->FrameA[2][3].dC = -213;
0	3	0	18.75	-13.5		xtalk->FrameA[3][0].dS = 300;	xtalk->FrameA[3][0].dC = -216;
0	3	1	18.3125	-12.625		xtalk->FrameA[3][1].dS = 293;	xtalk->FrameA[3][1].dC = -202;
0	3	2	21.25	-10.625		xtalk->FrameA[3][2].dS = 340;	xtalk->FrameA[3][2].dC = -170;
0	3	3	22.5625	-12.25		xtalk->FrameA[3][3].dS = 361;	xtalk->FrameA[3][3].dC = -196;
0	4	0	19.125	-12.625		xtalk->FrameA[4][0].dS = 306;	xtalk->FrameA[4][0].dC = -202;
0	4	1	21	-11.813		xtalk->FrameA[4][1].dS = 336;	xtalk->FrameA[4][1].dC = -189;
0	4	2	17.75	-11.625		xtalk->FrameA[4][2].dS = 284;	xtalk->FrameA[4][2].dC = -186;
0	4	3	19.125	-12.25		xtalk->FrameA[4][3].dS = 306;	xtalk->FrameA[4][3].dC = -196;
0	5	0	18.5	-13.125		xtalk->FrameA[5][0].dS = 296;	xtalk->FrameA[5][0].dC = -210;
0	5	1	12.5	-10.5		xtalk->FrameA[5][1].dS = 200;	xtalk->FrameA[5][1].dC = -168;
0	5	2	20.8125	-11.125		xtalk->FrameA[5][2].dS = 333;	xtalk->FrameA[5][2].dC = -178;
0	5	3	23.5	-10		xtalk->FrameA[5][3].dS = 376;	xtalk->FrameA[5][3].dC = -160;
0	6	0	19.3125	-12.25		xtalk->FrameA[6][0].dS = 309;	xtalk->FrameA[6][0].dC = -196;
0	6	1	18.375	-12.375		xtalk->FrameA[6][1].dS = 294;	xtalk->FrameA[6][1].dC = -198;
0	6	2	18.1875	-12.438		xtalk->FrameA[6][2].dS = 291;	xtalk->FrameA[6][2].dC = -199;
0	6	3	22.875	-12.063		xtalk->FrameA[6][3].dS = 366;	xtalk->FrameA[6][3].dC = -193;

NOTE: The `Conversion_of_xtalk_data_for_callback.xlsx` Excel file for the conversion is included in the `AFBR-S50-XTK.zip` file available under https://github.com/Broadcom/AFBR-S50-API/releases.

3. Implement the xtalk callback function in your C code.

Figure 44: Implementation of the Hard-Coded Xtalk Vector Table

```c
void Argus_GetExternalCrosstalkVectorTable_Callback(argus_cal_xtalk_table_t * xtalk,
                                                    argus_mode_t const mode)
{
    (void)mode;
    assert(xtalk != NULL);
    memset(xtalk, 0, sizeof(argus_cal_xtalk_table_t));

    // Set crosstalk vectors in Q11.4 format.
    // Note on dual-frequency frame index: 0 = A-Frame; 1 = B-Frame

    xtalk->FrameA[0][0].dS = 475;
    xtalk->FrameA[0][1].dS = 483;
    xtalk->FrameA[0][2].dS = 464;
    xtalk->FrameA[0][3].dS = 505;
    xtalk->FrameA[1][0].dS = 402;
    xtalk->FrameA[1][1].dS = 417;
    xtalk->FrameA[1][2].dS = 455;
    xtalk->FrameA[1][3].dS = 492;
    xtalk->FrameA[2][0].dS = 346;
    xtalk->FrameA[2][1].dS = 323;
    xtalk->FrameA[2][2].dS = 368;
    xtalk->FrameA[2][3].dS = 419;
    xtalk->FrameA[3][0].dS = 300;
```

After the xtalk vector table is hard-coded in the callback function, the API automatically retrieves the xtalk vector table values and includes them in distance and amplitude calculations.

Chapter 7: Checklist – Cover Glass Calibration at a Glance

Calibration Prerequisites

1. Verify whether a calibration is even necessary:
   - Without glass: An electrical xtalk calibration might be needed only for highest-distance accuracy requirements (≤ 1%).
   - With glass: If a direct glass-to-sensor attachment is possible by design, a calibration might not be needed.

NOTE: To determine whether a complete xtalk calibration is needed, make sure to perform a test measurement with a cover glass, the maximum distance, and the lowest object remission required by your application (Page 17).

2. Cover glass characteristics should comply with Table 1.
3. The sensor is embedded in the final mechanical design (soldered to PCB and final position of glass).
4. Material to properly cover the sensor is available.

Performed Calibration Steps

1. Electrical xtalk (Section 4.1).
2. Optical xtalk without glass (Section 4.2).
3. Optical xtalk with glass (Section 4.3).
4. Calculation of total xtalk (Section 4.4).
5. Calibration target/optical sink results in a maximum signal amplitude (without cover glass).

Sensor Types[a]	Maximum Amplitude in LSB	Interpolation Required/Possible
85G, 85I	≤ 10	No
85D, 68B	≤ 100	Yes

a. The same rule applies to the extended versions such as the AFBR-S50L**X**85D.

Procedure to Store Total Xtalk Values in Your C-Code

1. Use hard-coded xtalk by invoking the xtalk callback (Section 6.2).
2. Use the nonvolatile memory (NVM) of your MCU or any external storage device (Section 6.1).

Chapter 8: FAQs

8.1 Do I need a xtalk calibration when a cover material is not in use?

As soon as you have your own PCB layout for the sensor, it is suggested to perform an electrical xtalk calibration (see Section 4.1) to achieve best distance accuracy. For applications that, for instance, require only a presence detection without the need for an accurate distance measurement, it is not necessary to carry out the calibration.

8.2 How do I know if I need a cover glass calibration in my application?

If the mechanical system design does not allow direct attachment of a cover glass to the sensor (see Figure 2B), a calibration might be needed already. Best practice is to use a cover glass with the longest required and expected distance, as well as the lowest reflective (dark) objects/targets, and then to see if the measurement accuracy still fulfills the application requirements.

8.3 Is one xtalk calibration enough for all settings of the sensor?

The AFBR-S50 API exhibits two different measurement modes, short and long range, both of which use dedicated laser modulation frequencies. If both modes are in use (not by default), it is required to calibrate and store the xtalk vector tables for each mode. See Section 6.1 for more details.

8.4 Is there a suggestion for suitable cover glasses?

Glass-like materials with an optical transmission of ≥ 90% are preferred materials. Examples are most silica glasses, Gorilla glass (high mechanical strength), PMMA (also known as acrylic or Plexiglass), and PA (see Table 1 for further requirements). Dark glasses with proper optical characteristics will also work.

8.5 What is the typical impact of a cover glass on the maximum distance?

For instance, if the target sensor is specified up to 10m and is integrated with a cover glass that has a transmission *T* of 90% at the operation wavelength (for example, 850 nm), the maximum distance is reduced to about 8.1m. Generally, the derating can be calculated as follows:

$$D_{meas,max}(glass) = D_{meas,max}(no\ glass) \times T$$

The amplitude is attenuated by T^2, the maximum distance scales with the square root of peak power, and therefore it scales linearly with *T*.

8.6 Which material can I use as an optical sink for the optical xtalk measurement?

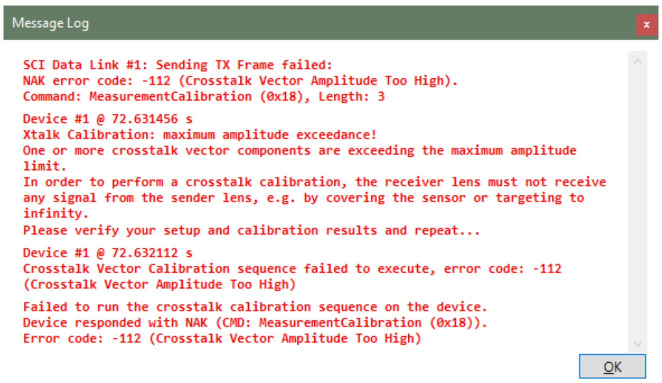
In principle, you should use a target material that preferably has less than 5% remission at the sensor signal wavelength, for example:

- https://www.thorlabs.de/thorproduct.cfm?partnumber=BKF12
- https://sphereoptics.de/product/zenith-lite-targets/
- https://molton24.de/en/photo-background-sold-by-the-metre-stage-molton-300g/m2-black-3m

8.7 What is the best practice for a productive implementation of a xtalk calibration?

See Chapter 6.

8.8 How do you interpret error message -112?



This error indicates that the defined amplitude threshold (see step 1 in Section 5.2.3) is set too low, or in other words, the xtalk values are too high. The remedy is to increase the amplitude threshold via the API function (see Section 5.3).

8.9 How much memory is required to store a cross talk vector table?

See Section 6.1.

Chapter 9: Document References

Reference Name	Document Type	Link
AFBR-S50MV85G Sensor	Data Sheet	https://docs.broadcom.com/docs/AFBR-S50MV85G-DS
AFBR-S50MV85I Sensor	Data Sheet	https://docs.broadcom.com/docs/AFBR-S50MV85I-DS
AFBR-S50LV85D Sensor	Data Sheet	https://docs.broadcom.com/docs/AFBR-S50LV85D-DS
AFBR-S50LX85D Sensor	Data Sheet	https://docs.broadcom.com/docs/AFBR-S50LX85D-DS
AFBR-S50MV68B Sensor	Data Sheet	https://docs.broadcom.com/docs/AFBR-S50MV68B-Time-of-Flight-Sensor-Module-DS
AFBR-S50 TOF Basics	Application Note	https://docs.broadcom.com/docs/AFBR-S50-BAS-Time-of-Flight-Basics-AN
AFBR-S50 API Example Porting Guide	Programming Guide	https://docs.broadcom.com/docs/AFBR-S50-SDK-Porting-Guide-to-Cortex-M4-PG
AFBR-S50 Reference Design	Application Note	https://docs.broadcom.com/docs/AFBR-S50-RD-AN

Revision History

AFBR-S50-XTK-AN100; July 3, 2023

- Initial release.

BROADCOM®