Step 1:

Tape the sides of the Shelly unit to a piece of light paper, such that the VL53L1X sensor is sitting just above the edge of the paper. The edge of the paper will be used to measure the distance between the sensor and the wall.

Step 2:

Tape the paper 1cm away from the edge of a heavy, dark and flat hardcover book. This book will provide stability to ensure the sensor doesn’t move during the calibration. By taping the paper 1cm away from the edge of the book cover, a contrast is created that can be used to easily spot the location of the paper’s edge.

Step 3:

Print out a paper with a solid grey colour covering its surface. The colour should be Munsell N9.5, or in RGB values: Red = 244, Green = 243, Blue = 239. You can use Microsoft Paint to create this RGB colour.



Step 4:

Tape the grey coloured paper to a vertical wall.

Step 5:

Place the textbook that is taped to the VL53L1X sensor about 14cm away from the grey paper on the wall. The VL53L1X sensor should be pointing roughly at the center of the paper. Use other hardcover books to reach this position

Step 6:

Connect the Micro USB cable from the Shelly unit to the computer.

Step 7:

With a rectangular stiff ruler, and compensating for any non-marked edge space on the ruler, position both the left and right edges of the paper (which the sensor is taped to) to be exactly 14mm away from the grey vertical wall paper. This measurement should be done when the edge of the ruler is flush the face of the wall to ensure the VL53L1X sensor is not angled away from the wall. See the images below for an example.

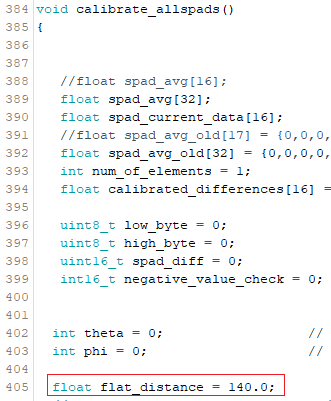


Step 8:

Open “Calibration\_Attemp1.ino” Arduino file.

Step 9:

Edit the Calibration\_Attemp1.ino code such that the variable “flat\_distance” on line 405 within the “calibrate\_allspads()” function is set to “140.0”. See the image below for an example.



Step 10:

Install and run the code in ”Calibration\_Attemp1.ino” in the shelly unit. This code will calculate the exact theoretical vector distance from each SPAD to the wall, then average the readings for 16 4x4 SPADs on the VL53L1X sensor, then save the differences (offsets) between the measured and theoretical distance for each SPAD in the ESP32 EEPROM.

Step 11:

Install the non-calibration application firmware for the Shelly unit. Use the function “void retrieve\_EEPROM\_data(int16\_t \*data\_array\_pointer)” defined in “Calibration\_Attemp1.ino” to retrieve the SPAD offsets and correct for distance measurements. See below for an example:

|  |
| --- |
| int16\_t calibrated\_offsets**[**16**];**  **...**  retrieve\_EEPROM\_data**(**calibrated\_offsets**);**  **...**  **for(**int i2 **=** 0**;** i2 **<** 16**;** i2**++** **)** // for each of the 16 SPADS SPAD  **{**    sensor**.**setROISize**(**4**,** 4**);**  sensor**.**setROICenter**(**left2right\_top2down\_spad\_centers**[**i2**]);** // uint8\_t left2right\_top2down\_spad\_centers[16] = {145, 177, 209, 241, 149, 181, 213, 245, 110, 78, 46, 14, 106, 74, 42, 10};    // get the reading for that SPAD  sensor**.**readSingle**(**false**);** // start single measurement    **while** **(!**sensor**.**dataReady**())**  **{**  //wait for sensor data  **}**    distance\_in\_millimeters **=** **(**uint32\_t**)(**sensor**.**read**(**false**));** //      distance\_in\_millimeters **=** **(**uint32\_t**)(((**int16\_t**)**sensor**.**read**(**false**))** **+** calibrated\_offsets**[**i2**]);**      **}** |