Embedded Systems Project: Characterization of VL53L5CX Time-of-Flight Ranging Sensor

Kenan Henriksen, student_1, *MAT.236813*
Gerardo Limón Cabrera, student_2, *MAT.236866*

***Abstract*—The VL53L5CX multizone Time-of-Flight ranging sensor is the newest sensor from the ST FlightSense product family and, in comparison with other ToF sensors in the market, it provides a new feature of multizone ranging output with 4x4 and 8x8 separate zones while also running in a frequency of 60 Hz, making it the fastest multizone, miniature ToF sensor available. The purpose of this paper is to make a characterization of the VL53L5CX multizone Time-of-Flight ranging sensor, describing key aspects like its components, functionalities, and performance. We will test the sensor in its 4x4 default resolution mode, as well as discussing about its performance given different values in variables of interest like the distance from the sensor to different surfaces and the ranging frequency. The VL53L5CX sensor is highly versatile and can be used in a variety of applications like automation and object detection. Therefore, we believe this paper is imperative to provide more information about this sensor´s performance and advantages, so that people can continue to explore and discover new applications that are fit for this state-of-the-art sensor. We tested the performance of its 4x4 multizone feature in different single-target scenarios and compared the results between them, observing the expected 16 zones with accurate depth perception in the field of object detection.**

I. Introduction

During the past decade, there has been an increase in the advances in technology and it has started to propagate to a lot of fields. One of the most common trends are computer vision applications. Computer vision is the field of study that consists in taking visual data from the environment and the world in general and translate it into data that the computer can understand. To achieve this, you need some sensors to gather this data to later be processed. There exist several types of sensors in the market like ultrasonic sensors, that use sound waves to detect objects. Cameras have also been used to take pictures of the world and then, with some software, extract the necessary data for the specific application. The type of sensors that we are going to talk about in this paper are ToF sensors. ToF stands for Time of Flight and this technology consists of an instrument that emits light and bounces back when it touches another surface; the time the light takes to travel from the source to the surface and back is used to measure distance from the source instrument to the object. There are several signal types that are used to measure distance, but the most common one uses infrared lasers, following the same concept of emitting a pulse of light to a specific object and when it hits a target, it bounces back; with that information, the measurement can be done.

Within this field of computer vision, a lot of applications have rose. There are even some companies that have been starting to look for instruments that can measure the level of the water in tanks or in other areas. The problem with water level sensing instruments is that they normally need to be submerged in the water or be waterproof, because there are not a lot of reliable, low cost sensors that can do measurements out of the water.

Another trend that has been developing more in recent years are the UAVs (unmanned autonomous vehicles). As an example, there has been an increase in surveillance applications where UAVs like drones use certain type of sensors or instruments to detect movement. The applications don't stop in the surveillance, but also autonomous drones need to have a sensor that can detect obstacles in order to avoid collisions. One of the main problems about the object detection or obstacle detection are that the results of existing sensors vary in different scenarios where there are multiple objects, dynamic obstacles, different sources of lightning (direct lightning, indirect lightning, or few lightning), between others. The instruments that were available in the market for these applications were normally not as accurate or with a higher price.Consequently, there is a need for a distance measuring sensor that is lightweight, low cost, small size, that can also provide with accurate and reliable object or obstacle detection with fast frame rates.

The VL53L5CX Time-of-Flight Ranging Sensor is a state-of-the-art sensor that has gained popularity in the technological world due to its unique features and capabilities. It comes with a 4x4 and 8x8 multizone feature with motion indicator per zone, while also having multi-target detection and distance measurement from the different zones. The newest feature is its 60 Hz frame rate that allows the sensor to provide with fast and accurate distance ranging up to 400 cm. To accurately measure the distance between the sensor and an object, the sensor emits an invisible light (940 nm) and uses a single-photon avalanche diode (SPAD) detector array. It also comes with accurate results within different lighting scenarios nor glass protection.

The purpose of this paper is to make a characterization of the VL53L5CX ToF sensor and explore some of the new features that have been integrated in the instrument.

With this, we are aiming to provide a general idea of how to use the sensor, its expected performance and to observe how accurate the sensor can be, manipulating through the different variables of interest and parameters like the timing budget and frequencies. We want to explore and open the possibilities of implementing this sensor in new applications or to enhance existing ones like obstacle detection, gesture recognition, virtual and augmented reality, between others. The sensor will be tested several times in its default 4x4 multizone feature, exploring the different frequencies in different surfaces and distances within the max ranges and in a room with not direct lighting (indoor room with natural indirect light). The reason behind using the sensor's default feature instead of the 8x8 multizone feature is that its much faster, and in the scope of this paper we want to test the 4x4 multizone feature with the fastest frequency available. The default mode of the sensor is faster because it doesn't need to measure that much data points and therefore can present results quicker. There have been other multizone ToF sensors before like the VL53L1CB, but the ranging frequency of 60 Hz is unique to the VL53L5CX. It is also important to note that this maximum frequency can only be obtained by running the default 4x4 feature.

This paper is organized into 5 sections. The Section I corresponds to the introduction, where the context of the paper and the main questions were presented. Following that, the Section II describes the Related Works and their relationship with the paper. Section III consists of the Development of the Work, where the problem statement will be explained along with the hardware and software implementation. Next, Section IV will contain the results of the paper. Finally, Section V will present the conclusion.

II. Related Work

As we have been exploring, the VL53L5CX ToF Sensor has improved in some key aspects compared with the other sensors from the same family product, like the capability of doing multizone distance measurements with a very fast speed, while also maintaining the light weight feature in a low cost range. This has lead to a lot of new applications and improvements in several technology areas.

A. Towards Reliable Obstacle Avoidance for Nano-UAVs

In the unmanned aerial vehicles, there has always been the necessity of developing new, lightweight, fast response technology for obstacle detection. In the 2022 21st ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN), Ostovar et al. (2022) proposed a paper in which they implemented the VL53L5CX Sensor in a nano-drone and tested it in an indoor 16m$\hat{2}$ room where they had different obstacles. The results of the experiment showed a 95% of the trails having no collisions during the whole battery life of the drone (7 minutes) in scenarios with multiple objects with the sensor's 8x8 multizone feature. While this paper establishes the contribution of the ToF sensor in its 8x8 feature on nano-drones, it is also important to characterize the sensor's 4x4 feature that enables faster frequencies (60 Hz) to find new applications like this one.

B. DELTAR: Depth Estimation from a Light-Weight ToF Sensor and RGB Image

While the VL53L5CX ToF sensor has a very good obstacle detection feature, it has a limited range and the resolution that it can provide is not enough to be able to be useful for applications like 3D reconstruction. Yijin et al. (2022) proposed the method DELTAR: Depth Estimation from a Light-Weight ToF Sensor and RGB Image. This method consists of calibrating a VL53L5CX Sensor and a RGB Camera, an Intel RealSense D435i, aligning the signals and the colored imaged to provide a high-resolution depth image. The authors ran several tests disabling different parts of their method, and they showed significant losses of resolution in comparison with the test made with the complete method. The work made by these authors give a new perspective on add-ons or modifications that can be implemented in applications with the VL53L5CX ToF Sensor to improve and enhance current computer vision and robotics applications.

C. DiscoBand: Multiview Depth-Sensing Smartwatch Strap for Hand, Body and Environment Tracking

The applications of the sensor are not just for UAVs, but there have been some works in which the sensors are used with its object detection feature to allocate human parts/different positions of the body. DeVrio and Harrison (2022) believed that nowadays there is a lot of development in the Human-Computing Interaction field (HCI) and they explain that most of the applications rely on hand-tracking. Therefore, the authors proposed a smartwatch strap called DiscoBand, consisting of 16 VL53L5CX ToF sensors in its 8x8 multizone mode to track the position of the hands and the arms of a person in real time, enabling the computer to make a virtual image of the person's hand and their surroundings. The authors conclude with some accurate results of the tracking and imply the importance on continuing the research of HCI and its applications. The ToF sensor played a key role in the paper by providing reliable and fast data for their experiments, enlarging the possibilities of use of the sensor to areas like virtual reality, augmented reality, smartphones and other HCI technologies.

D. 360 Degrees Object Detection Using Multiple ToF Sensors for Educational Robot

Robotics has been a trend for the past few years and, as technology continues advancing, the schools around the world are starting to implement robotics lectures. Because of this, the necessity of security for the students in the classroom rose. Therefore, a robot that is in the classroom should have a object detection mechanism. Kongpecht et al. (2002) proposed two designs to integrate VL53L5CX ToF sensors in the robot's design, in order for the robot to be aware of its surroundings and provide this security in the learning area. In their results, the authors discovered that the second design was the best in terms of minimizing the blind spots of the robot while using

less sensors, therefore less power consumption also (10 sensors instead of 16). The proposal of the authors is relevant for this paper because they use multiple VL53L5CX sensors arranged in a certain way to provide a 360 degrees field of view; it is imperative to characterize the sensor and discover all its key features to be able to find more applications like these that take advantage and even surpass the sensor's limitations, in this case being the 45 degrees maximum field of view.

III. Development of the Work

A. Problem statement

As mentioned earlier, the ToF-sensor is an advanced ranging sensor with high accuracy and precision, for which we wanted to test:

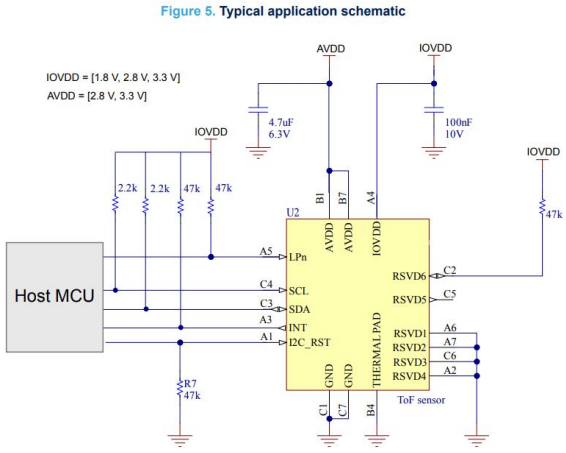
1) How accurate is the sensor as a function of distance and frequency?
2) How sensitive to change is the sensor?

In order to visualize our tests, we are plotting the data from the test. After the tests has been performed, we are creating an algorithm, which will be able to detect a change in measured distance from the sensor within a given threshold.

B. Hardware setup

The schematic for the hardware is shown in III-B. The sensor is using I2C for communication with the Nucleo board. In order to establish this connection, 4 pull-up resistors are connected to the pins SCL, SDA, INT and LPn. One pull-down resistor of 47k is connected to I2C reset.
Aside this, there is 2 decoupling capacitors of 4.7uF and 100nF placed as close as possible to the VDD pins of the sensor. These are mandatory in order to lower the noise for the sensor. The specific setup was made on a breadboard with jumper wires and through hole capacitors and resistors.

Fig. 1. Schematic for hardware



C. Structs

The library used for collecting data stores the value of the sensor in the struct RANGING_SENSOR_Result_t. This struct is an array with only 2 fields: Number of zones (uint32_t) and a RANGING_SENSOR_ZoneResult_t array. This field is another struct, which has the following variables:

1) Number of targets per zone (uint32_t) (for this sensor, it can only detect 1 object pr. zone)
2) Distance (uint32_t) (which is what we are focusing on throughout this paper)
3) Status (uint32_t)
4) Ambient(float)
5) Signal(float)

Since we only care about the accuracy and the measured distance in our case, we only fill synthetic data into the Distance field, however when the actual sensor is used, all fields are filled with data. The struct RANGING_SENSOR_ZoneResult_t's fields are all arrays, with size RANGING _SENSOR_NB_TARGET_PER_ZONE. As mentioned earlier, this sensor can only detect 1 object per. Zone, and therefore this array has a size of 1, ie. when filling data into the struct, the array index in the array is set as 0 as shown in III-C:

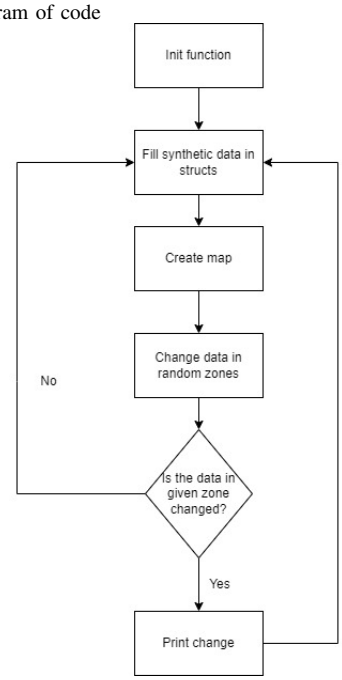
```
//Line of code for setting the distance of
    zone i to 1000
lastval.ZoneResult[i].Distance[0]=1000;
```

It starts by initializing the function. The initDetectionFunction(int8_t res) takes the input res and returns nothing. It creates a local variable in the detectionFunction.c file. This value is the threshold, at which a change in sensor value above this threshold will result in a change in the output for the terminal.
The fillDataInStructs() takes no input and returns nothing. It places synthetic values of 1000 (corresponding to 1000mm for the sensor) in the struct array RANGING_SENSOR_Result_t. This is the line which is also shown in the structs section III-C.
The printStatement() takes no input and returns nothing. It creates the map. This function map the zones by printing squares made of X. The output in the terminal is shown in III-D

D. Software Diagram

The flow diagram for the algorithm is shown in III-D: As shown in the appendix, this is what the map looks like, when the zones 9, 10 and 13 has decreased values. When the values are decreased the zones made of X are changed to O.
This is done by the function printChange(int8_t zone) which takes the zone as input and returns nothing. This function is the only function, which has to be changed, in order for the code to be useable for the 8x8 zones. This is due to the if statements being hard-coded to the specific zones. In order for it to work with 8x8 zones, the function needs 4 more if-statements (one for each row).
The function clrscr() takes no input and returns nothing, and consists of a single print statement, which uses an escape code. This escape code clears the terminal. The function gotoxy(int8_t r, int8_t c) takes 2 input and returns nothing, where the first is the row at which the terminal should go to,

Fig. 2. Flowdiagram of code



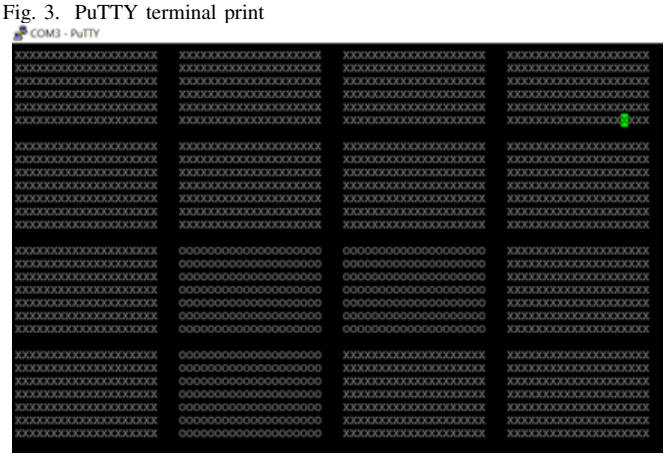
and c is the coloum. This can be translated to the x- and y-coordinates in the terminal. When this function is being run, the terminal goes to a specific location in the terminal, and when the next print statement is called, it prints in this location. This is how the map is navigated.
The function alterData() takes no input and returns nothing. It starts by calling the function printStatement() (meaning it redraws the map again), then fills the synthetic data into all of the zones ie. changes the values back to the start data. After this, it picks between 1-3 random zones, and changes the values of these zones from 1000 to 500. If the threshold is $<$500, the terminal will print a change (drawing O's instead of X's).

```
//Line of code for checking if change is
    bigger than set threshold
if((pval->ZoneResult[i].Distance[0])-
   (plastval->ZoneResult[i].Distance[0]) >
      threshold){
```

It then has a delay, for the data to be displayed).
The last function checkEvent() takes no input and returns nothing. It runs through a for-loop checking for a change in a the values for each zone. In case of a change $>$threshold, it calls the function printChange(i) where i corresponds to the specific zone. The line checking if there is a change is shown in III-D Where pval is a pointer, which points to the specific zones distance. plastval is also a pointer pointing to the last value. If these 2 values subtracted are bigger than the threshold, it will display the change in the terminal.

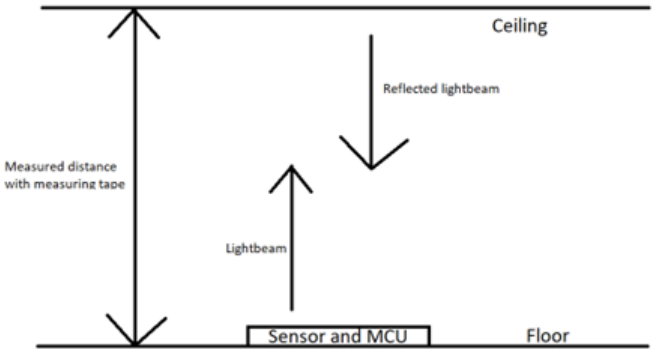
1) Plots: Each plot is represented by between 4 and 5 measurements, where each measurement has been performed 3 times. The black dots are the mean of the 3 measurements, red the the standard error of mean added to the mean and green the standard error of mean subtracted from the mean.

Fig. 3. PuTTY terminal print



All of the dots has the actual distance subtracted as well, so that the y-axis gives the distance in mm the sensor is off from the actual distance. See III-D4 for an example.
The standard deviation and standard error of mean is calculated using Maple. This is also the case in the 3D plots. In the first section of the plots, the x-axis is the actual distance in mm, and as mentioned earlier, the y-axis is how far the measurement is off from the actual distance (also in mm).
In the second section of the plots, we do not change the actual distance but the frequency of the sensor and the timing budget. Therefore there is now 2 parameters we are changing, and that is why it is a 3D plot. In the very last section, we checked how sensitive the sensor is. This is done by placing a very small object in front of the sensor, and then checks if the sensor dectects that object. In our case we tried with 2mm and 5mm, and once again the standard error and the mean is calculated for these. There is no plots for these calculations. All measurements were tried out 3 seperate times, and from these 3 measurements we calculated the standard deviation and the standard error of mean.

Fig. 4. Drawing of experimental setup



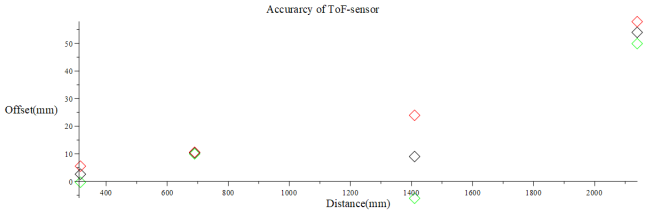
2) Change of actual distance: When changing the actual distance, we measured the distance with a measuring tape, as mentioned earlier. Since we are not able to measure the very exact distance, the actual distance is $\pm 5mm$ for the shorter distance (below 1000mm) and $\pm 10mm$ for the longer

distance (above 1000mm).

3) Experimental setup: The setup used for the experiments is shown in III-D1: Then from that, the sensor is placed flat on the floor, and the code from the library collecting data is run.
The code prints the distance for all the zones in the terminal, and the data is then written down. As mentioned earlier, all data collection were done 3 times. The different distances used were: $315mm \pm 5mm, 690mm \pm 5mm, 1440mm \pm 10mm, 2140mm \pm 10mm$ and $2880mm \pm 10mm$. In the data collection for the 1440mm distance, we only collected data for 6 of the 8 middle zones (zone 5-8 and 9-10). This is due to limitations of the experiment environment.
All measurements were done in a room with shades and no lightbulbs, in order to minimize the amount of light. This secures the most precise data collection.

4) Change of frequency and timing budget: In this part of the experiment, we kept a fixed distance for the sensor, however changed the timing budget and frequency of the sensor. Since we changed 2 parameters in this case, we had to do a 3D-plot, where the x-axis and the y-axis is respectively the timing budget and the frequency. The z-axis is how far the distance from the sensor is off from the actual distance. The plot is shown as:

Fig. 5. Plot of accuracy for the sensor as a function of length



5) Determination of sensor sensitivty: For this experiment we once again used a fixed distance. In order to determine the sensitivity of the sensor, we took a stack of papers, and measured the thickness of the stack. In this case we started with a stack with a thickness of 5mm. When placing the stack of paper in front of the sensor, we could see a noticeable change in values, therefore it was determined that the sensitivity of the sensor were smaller than that. Therefore the experiment were carried out again, this time with a stack of papers with a thickness of 2mm.

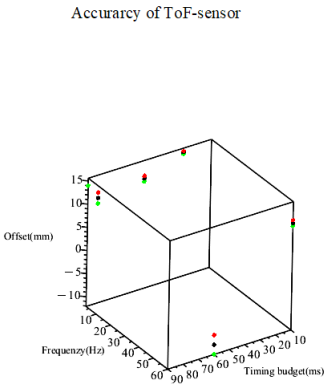
IV. Results

Throughout the plots, where the plots are a function of distance, it is clear, that the sensor has a tendency to measure the length a bit longer than the actual distance for longer distances, and for the smaller distances it tends to measure a bit shorter. It is also shown, that a bigger distance decreases the accuracy of the sensor, which is mentioned in the datasheet page 23/38 table 22. It is mentioned that the sensor for 4x4 zones and 30Hz has an accuracy of $\pm 6\%$ [1].

[1]https://www.st.com/en/imaging-and-photonics-solutions/vl53l5cx.html

For the 3D-plots with timing budget and frequency as variables, it is shown, that especially the timing budget worsen the accuracy of the sensor. However, there is still a correlation between frequency and the accuracy, as a small timing budget and a high frequency for most zones results in a more inaccurate measurement. In general the most inaccurate measurements are with a timing budget of 10 and a frequency of 60, which is the also the highest frequency and lowest possible timing budget. For the sensitivity of the sensor, a

Fig. 6. Plot of accuracy for the sensor as a function of frequency and timing budget



change of 2mm can be detected by the sensor (for small distances, in our case around 300mm), however due to the ever fluctuating values for the zones, it is hard to measure anything below or close to these values.
All of the plots and the script for the plots can be found on Github (link is in the next section) both as a PDF and as a Maple file.

A. How to run the code

In order to run the code, clone the repository from the Github link here. After cloning the code in the ExamProject folder to STM32CubeIDE, put the code onto the STM32-Nucleoboard (It is configured for Nucleo-F401RE). Open PuTTY (or similar software) and set the baudrate to 9600. When connection has been established run the code, and the changes should be shown in the terminal.

V. Conclusion

As new technology continues to develop and grow at a constant pace, the needs for characterizations of new technologies like the VL53L5CX multizone Time-of-Flight ranging sensor has become important to get to know the technology and the possible applications where it could be used. We characterized the sensor and observed its behavior in its 4x4 multizone mode and found that the instrument can provide reliable object detection within the 400cm distance, having a inversely proportional relation between the accuracy and the distance. We also discovered that there

is some correlation between parameters like the timing budget and frequency and the accuracy of the sensor; a high frequency and low timing budget is gonna gave a very inaccurate measurement. It is important to continue testing the VL53L5CX ToF sensor's capablities and explore also the other attributes like the 8x8 multizone feature to enhance its use in applications like robotics and object detection, but also to discover more possible applications where it could be useful.

REFERENCES

[1] Ostovar, V. Niculescu, H. Müller, T. Polonelli, M. Magno and L. Benini, "Demo Abstract: Towards Reliable Obstacle Avoidance for Nano-UAVs," 2022 21st ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN), Milano, Italy, 2022, pp. 501-502, doi: 10.1109/IPSN54338.2022.00051.

[2] Yijin Li, Xinyang Liu, Wenqi Dong, Han Zhou, Hujun Bao, Guofeng Zhang, Yinda Zhang, Zhaopeng Cui (2022) "DELTAR: Depth Estimation from a Light-Weight ToF Sensor and RGB Image" in Computer Vision – ECCV 2022, published by Springer Nature Switzerland, pp. 619 – 636.

[3] Nathan DeVrio and Chris Harrison. 2022. DiscoBand: Multiview DepthSensing Smartwatch Strap for Hand, Body and Environment Tracking. In The 35th Annual ACM Symposium on User Interface Software and Technology (UIST '22), October 29-November 2, 2022, Bend, OR, USA. ACM, New York, NY, USA, 13 pages. https://doi.org/10.1145/3526113.3545634.

[4] A. Kongpecth, N. Kwankeo and V. Manthamkarn, "360 Degrees Object Detection Using Multiple ToF Sensors for Educational Robot," 2022 19th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON), Prachuap Khiri Khan, Thailand, 2022, pp. 1-4, doi: 10.1109/ECTI-CON54298.2022.9795626.

[5] STMicroelectronics, "VL53L5CX Time-of-Flight 8x8 multizone ranging sensor with wide field of view", DS13754 datasheet, [Revised 9 - January 2023].