Traffic Sensing Bicycle Light

ECE 445 Design Document
Anchu Zhu, Jimmy Gan, and Jiawen Chen
Group 27
TA: Anthony Caton
2/18/2019

1 Introduction

1.1 Objective

Despite the increased awareness and improved visibility of bike lanes, there were still about 840 bicyclist death at motor vehicle accidents in the United States in 2016. Within these cases, there were 58% of bicyclists fatalities who did not occur at intersection locations, which was 28% more than accidents occurred at intersections [1]. This means more biking accidents happen when the bikes were crashed from the back when the driver are unaware of the cyclist. For this reason, it would be helpful to have a device located at the back of the bike to alert the drivers to maintain safety distance when they are getting too closed or speeding up towards the cyclist too fast, preventing such tragedies from happening.

Our intention is to design a device to detect traffic coming up from behind. It should be able to operate under three different modes to warn the drivers based on the relative distance and the approaching speed of the vehicle. In this way, the drivers will be able to react and slow down because they are alerted that there is a bicyclist in front.

1.2 Background

There are several different types of devices in the market related to bicycle safety. The bicycle lights that are available on the market are the devices that can only flash red light when the bicyclists hit the brake or just flashing red light when the device is on. In general, their functionalities are singular and not dynamic enough to alert the driver effectively. In the case of preventing accidents, multiple use of various equipments would be more efficient. Moreover, the capability of operating under different modes would be suitable under different situations.

Having such traffic sensing bicycle light is able to not only heighten the bicyclists awareness of potential accidents of approaching vehicles coming up behind but also have the drives to pay extra attentions to the bicyclists. And there would be three alarm modes for how close and how fast is the vehicle approaching to the cyclist. When both the relative distance and the relative velocity hits a certain threshold the device will start flashing white light. If the car continues getting closer, it will start playing a loud horn sound to warn the driver. Other times, it will flash red light to alert the driver behind. The parameters such as road condition and safety distance can be adjusted via bluetooth on an Android devices. The user can also have an evaluation report of the route they have ride. And they can compare which route is safer for the same destination. In a way, it lowers the chance of accidents from happening.

1.3 High-Level Requirements

- The device must be able to distinguish between vehicles coming up from behind and those parking beside, passing by and passing in the opposite direction, and it could recognize whether a car behind or a cyclist behind.
- The device must be able to have 3 modes of operation to alarm approaching drivers under different situation: a red flashing strobe light, white flashing strobe light and loud horn.

- The latency of the device must be at least 2 and a half hours.

2 Design

The sensing light requires five sections for a successful operation: a power supply unit, a control unit, a sensor unit, output devices units, and a software unit. The power supply unit includes three separate voltage source using alkaline batteries, and several voltage regulator for different modules of the devices. Such that, the device could run in its specific operating region. The control unit contains a microcontroller (Atmega328p), which has 32 KB flash memory and is able to handle the incoming data continuously from the sensor module. Then the control unit would determine which output device are to be activated. The user can change the parameters of distance and velocity threshold for the sensor via their Android device.

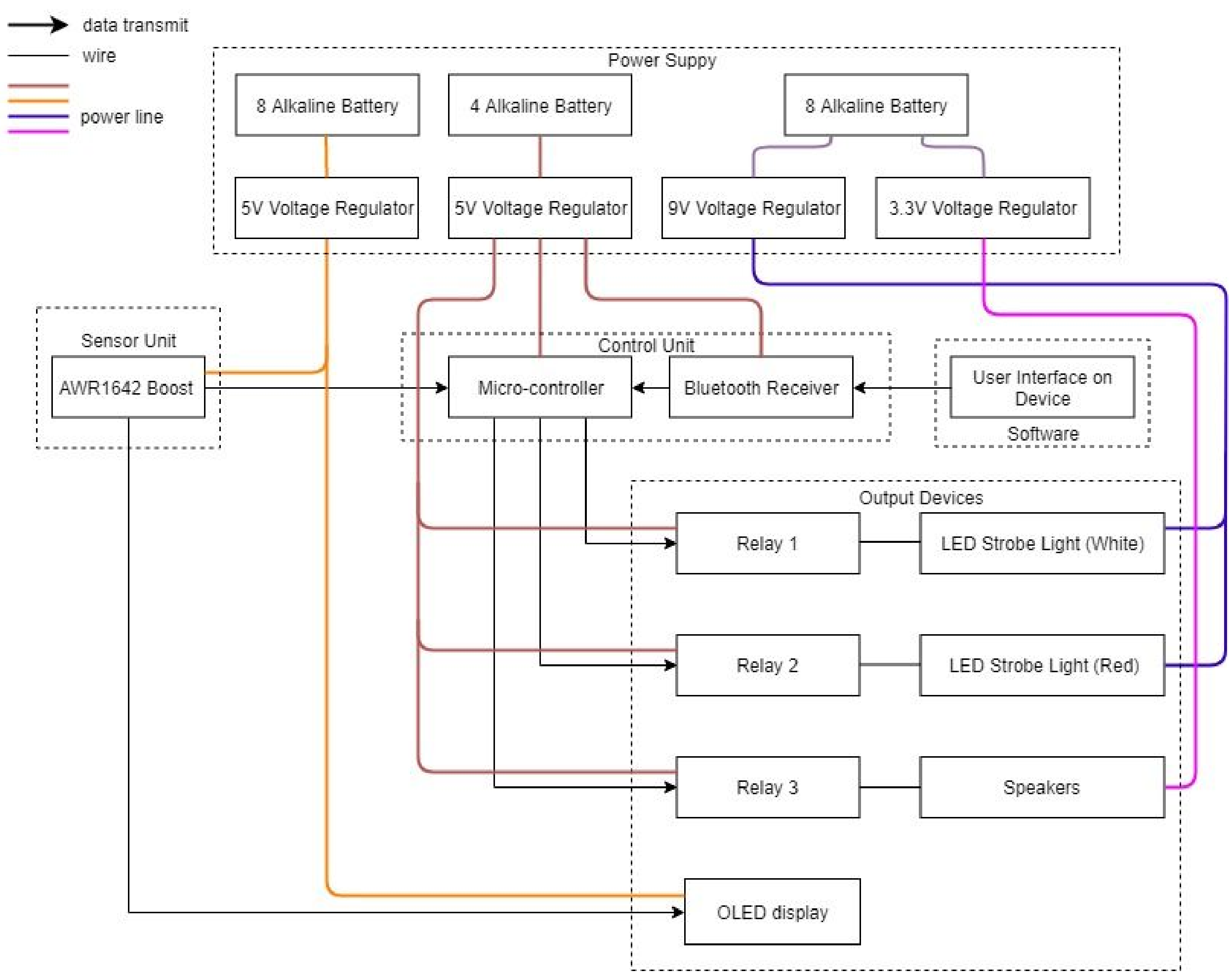


Figure 1. Block Diagram

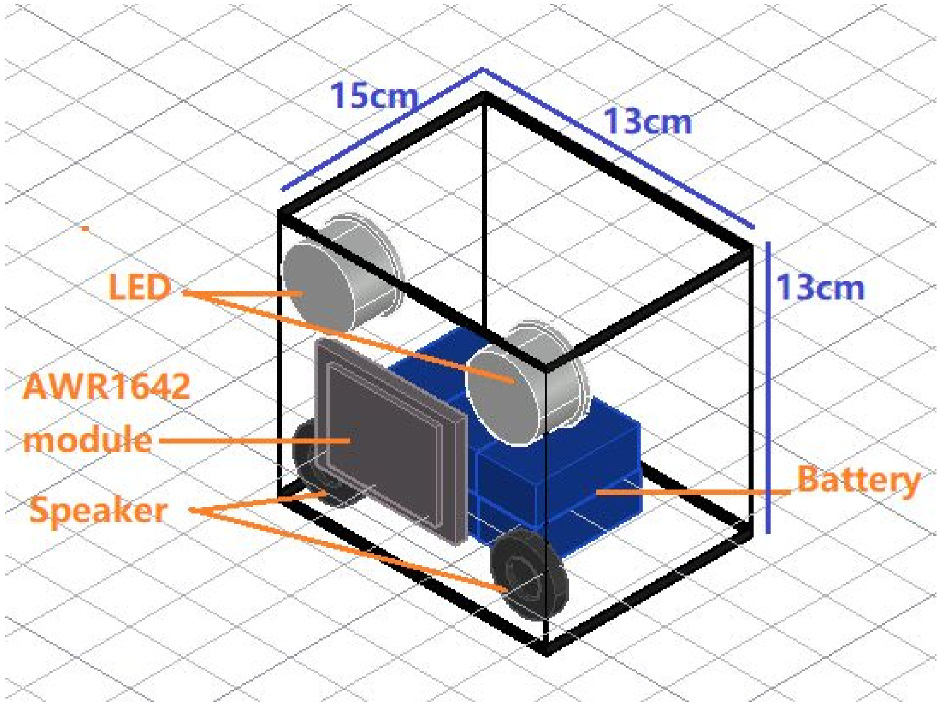


Figure 2. Physical Design

2.1 Sensor Module (AWR1642 EVM)

The sensor module we are using is directly from AWR1642 evaluation module, which contains the AWR1642 single chip and a complete antenna design shown in the figure below. The AWR1642 single chip mmWave sensor module utilize frequency-modulated continuous-wave radars to determine the distance and velocity of the objects.

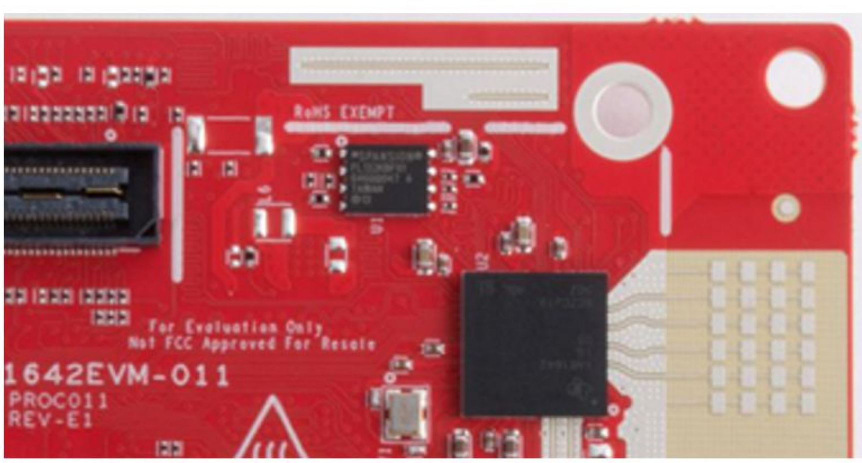


Figure 3. Slot antenna array design with coupon structure for probe by TI [*]

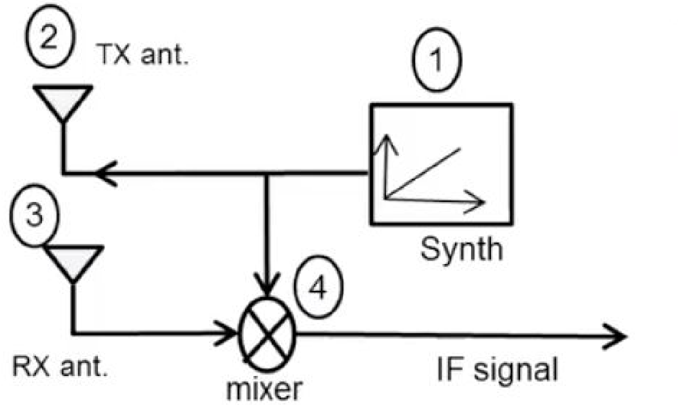


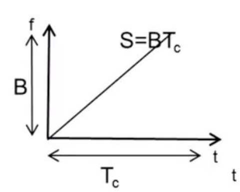
Figure 4. The process of how a FMCW Radar works with 1 RX ant and 1 TX ant

First, a synthesizer generates a chirp wave and then sends out via the transmitter antenna to send out a chirp. And then receive the chirp back and put it into a mixer, which takes the difference of their instantaneous frequency and the difference of their instantaneous phase difference to generate an intermediate frequency signal.

The sensor module is the input of the device. The sensor module contains a single chip AWR1642 which contains a on-chip microcontroller unit ARM Cortex-R4F and a integrated digital signal processing unit TI C674X. And the distance will between two objects will be calculated using the following equation with its resolution determined by the bandwidth of the chirp. This FMCW radar can also detect multiple objects' distance using the time difference on the reflected time.

For the case of multiple objects with the same distance, using the doppler fourier transform on their phase difference to determine their velocity difference to distinguish each objects. There is certainly trade off when it comes to maximum detction range and the detection resolution. Higher the chirp bandwidth is, better the resolution, but sacrificed the detecting range. And higher sampling rate also helps higher detection range.

- $\Delta f = \frac{S2\Delta d}{c}$
- $\Delta f > 1/T_c$
- $\frac{S2\Delta d}{c} > \frac{1}{T_c}$
- $\Delta d > \frac{c}{2ST_c} \Rightarrow \Delta d > \frac{c}{2B}$



Calculation 1: bandwidth of the chirp determine the resolution on how close two objects separated to be detected

$$d_{max} = \frac{F_S c}{2S}$$

Equation 1: sampling rate of the ADC and the bandwidth of the chirp determines the maximum detecting range.

$$\boxed{\omega = \frac{4\pi v T_C}{\lambda} \Rightarrow v = \frac{\lambda \omega}{4\pi T_c}}$$

Equation 2: Measuring the velocity using consecutive chirp with phase difference

$$v_{max} = \frac{\lambda}{4T_c}$$

Equation 3: maximum velocity requires lower transmission time, which means it requires closely spaced chirp.

$$v_{res} = \frac{\lambda}{2T_f}$$

Equation 4: resolution of the velocity is inversely proportional with the frame time.

In the case of angular estimation. FMCW Radar uses the method angle of arrival to give estimation of the angle relative to the orthogonal axis. And at least two antennas are required to use this method.

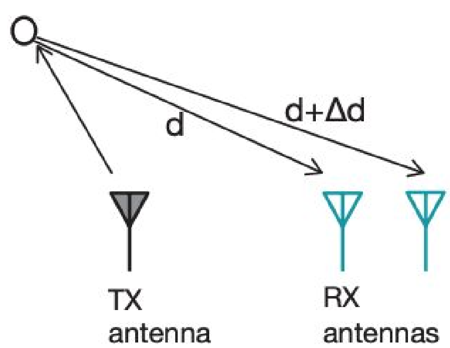


Fig. 5:How AoA applies on angle estimation.

And the expression for the angle is shown in the following equation. One thing that needs to take notice here is that, the angle would be more accurate when it is closer to the orthogonal axis of the sensor because the theta is approaching zero, and the estimation would use the theta directly when it is very closed to 0 in AoA. And the accuracy would decrease when theta is closer to +/-90 degrees which is on the horizontal axis of the antenna when viewed on top.

$$\theta = sin^{-1}(\frac{\lambda \Delta \Phi}{2\pi l})$$

Equation 5: The expression of the AoA estimation of the angle

And the maximum angular field of view is determined by the spacing between two antennas.

$$\theta_{max} = sin^{-1}(\frac{\lambda}{2l})$$

Equation 6: Maximum angular field of view of +/-90 degrees achieved when the spacing equals to ½ *λ
For this,

Requirements	Verifications
1. Sensor can work in the desired configuration: maximum range of 40m, range resolution of 41cm, maximum velocity of 70 km/h and velocity resolution of 1 km/h.	1. A. Input the desired confirguration into mmWave Estimator provided by TI to see the requirement for bandwidth, frequency range, transmit and receive gain. B. Connect the module via USB to the PC and use mmWave visualizer to check if the module can meet the requirement on single car detection while stationary.

2.2 Control

A control unit that is able to take the data from the sensors as input and determine which output device should be triggered according to the data collected. This unit is made up of two chips, one is the microcontroller from the AWR1642, Cortex-R4F, which is integrated in the chip for working on the logic of which mode should be activated, the other is our own microcontroller, which is Atmega128p for handling output devices and input parameters.

2.2.1 Microcontroller for I/O

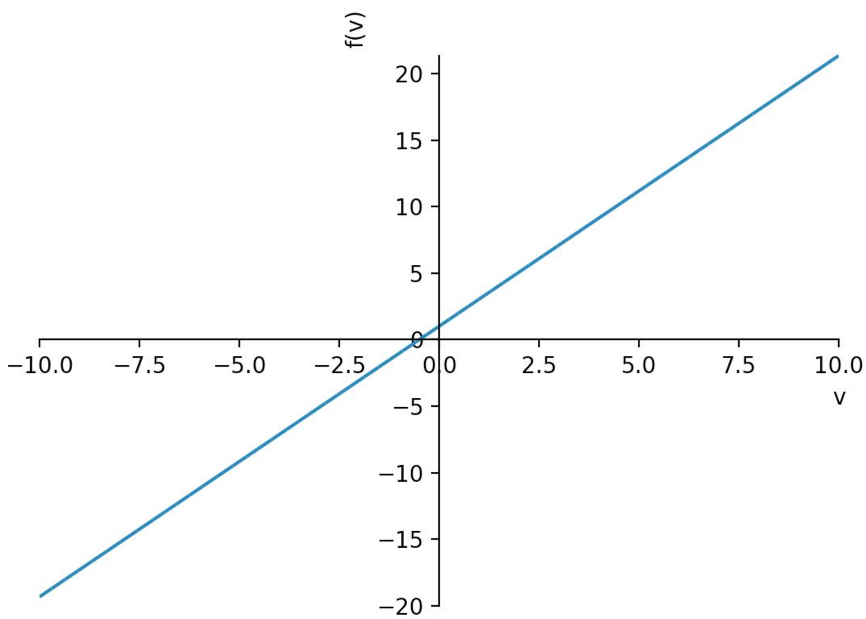
Microcontroller, Atmega328P with 32 KB flash memory, is chosen to handle input data management received via the HC-05 bluetooth transceiver and determine whether the input data can pass on to the next microcontroller Cortex-R4F in the AWR1642 and determine which device should be activated according to the output signal from the AWR1642. And the way to check the input data is using the following equation:

It is also capable to receive user input via UART by Bluetooth for the triggering parameters. The user will be able to input the frictional constant they want from the device. Frictional constant of 1 means the road is very dry and near 0 frictional constant means the road is very slippery Frictional constant will be default at 0.75, which is the constant for dry road. And the range of the frictional constant will be between 0.01 and 0.99.

$$D = d_c + 2v + \frac{v}{2 \cdot f \cdot g}$$

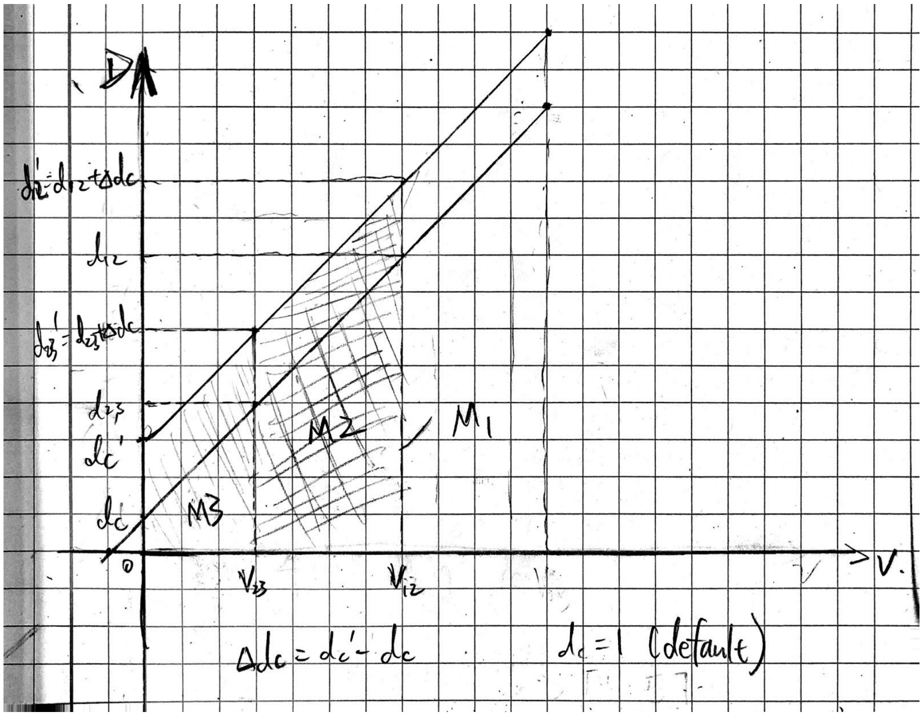
Eq. 7: stopping distance equation from Geometric Design of University of Idaho (7)

D: total distance between the car and the bike.
2: response time/delay time.
g: gravitational constant
v: velocity of the vehicle relative to the bicycle.
f : frictional constant on the ground.
d_c : safety distance between vehicle and the biker



Graph 1. Graphic representation of Eq. 1 at frictional constant being 0.75 and the safety distance between the back of bicycle and the car being 1. f(v) denotes the total distance between the vehicle and the cyclist, v denotes velocity of the vehicle relative to the bicycle.

Users will be able to modify f and d_c. The user can input f depending on the road condition. Then depends on if the user wants the device to be more sensitive or less sensitive on the velocity of the vehicle behind the cyclist at the same distance. If the safety distance is greater, the device is more sensitive, meaning the device would alert the driver at a relatively greater velocity compared to the velocity where it is at the same distance when the safety distance is smaller. If safety distance is smaller, the device will be less sensitive on the relative velocity. For the distance and velocity threshold ranges, they will change by the absolute value between the input d_c and default one. And the safety distance ranges from 1 m to 5 m while the default is 1 m. And for the velocity threshold, it will be calculated from Eq. 1. And the following graph explains the implementation.



Graph 2: Increasing the safety distance would certainly make the device easier trigger on relatively low velocity (more sensitive). And shifting the range by Δd_c as long as the ranges of the thresholds are between 0 and 40 m would make the alerting mode consistent.

Table 1: The range calculated from Eq.1
according to the Eq. , the default values are d_c = 1 m, f = 0.75, the default threshold is 25 m between mode 3 and mode 2, and 15 m between mode 2 and mode 3 and the velocity will be calculated from Eq. 1. Threshold between each mode will shift to default threshold+Δd_c, so that each mode will not be affected by the sensitivity of the device. And positive velocity denotes the velocity toward the cyclist, vice and versa.

Distance range: N/A	Velocity range: N/A	Regardless of the velocity, or distance flashing red light like normal bicycle light to alert driver
Distance range: (12 m, 25 m]	Velocity range: (5.32 m/s, 11.6 m/s]	Potential Rear Crash detected, activating mode 2, flashing white strobe light
Distance range: [0 m, 12 m]	Velocity range: [-0.5 m/s, 5.32 m/s]	Activating mode 3, horn for final warning

Table 2: This is the table of different ranges of velocity and distance threshold between mode 2 and mode 3 under different sets of frictional constant and safety distance

Input Parameters by user	Velocity threshold	Distance threshold
f = 0.75, d_c = 1 (default)	[-0.5 m/s, 5.32 m/s]	[0 m, 12 m]
f = 0.75, d_c = 3	[-1.451 m/s, 5.32 m/s]	[0 m, 14 m]
f = 0.45, d_c = 1	[-0.473 m/s, 5.20 m/s]	[0 m, 12m]
f = 0.45, d_c = 3	[-1.42 m/s, 5.20 m/s]	[0 m, 14 m]

The high level microcontroller is programmed on a TI user interface, the mmWave SDK. In order to use a mobile device to input the parameters the user want, addition Bluetooth transceiver HC-05, which has a detecting range of 10m, is needed to handle the wireless data transmission. And Atmega328p will operate at 16MHz clock cycle, which is fast enough to respond for this device. It will also be the master clock to communicate with AWR1642, the slave, via SPI.

Requirements	Verifications
1. Can both receive and transmit over UART at a speed at least 120kbps 2. Can give out signals to activate output devices 3. Cortex R4F can receive the output signal from the Atmega328p and passback the testing signals to Atmega328p via SPI	1. A. Use a test program that contains time.h output the variable that stores the input received from the UART to the console B. Type in a set of data of parameters from the device and send it to the microcontroller C. Check the output on the console see if it matches to what was input from the mobile device 2. A. Use a test program that sends a active low signal to the pins where the output devices such as the strobe LED and speaker are connected. B. Check if the corresponding activated device is on. 3. A. Use SPI.transfer() function on Arduino interface to test if the communication is working properly.

2.2.2 Software on AWR1642BOOST

AWR1642 single chip milliwave sensor will be responsible for collecting the velocity and distance data and determine which mode should be activated. The single chip will communicate with the Atmega328p via SPI. Once the input data is verified to be in range, they will be sent to AWR1642. And the antenna functionality on AWR1642BOOST would be verified on the sensor module.

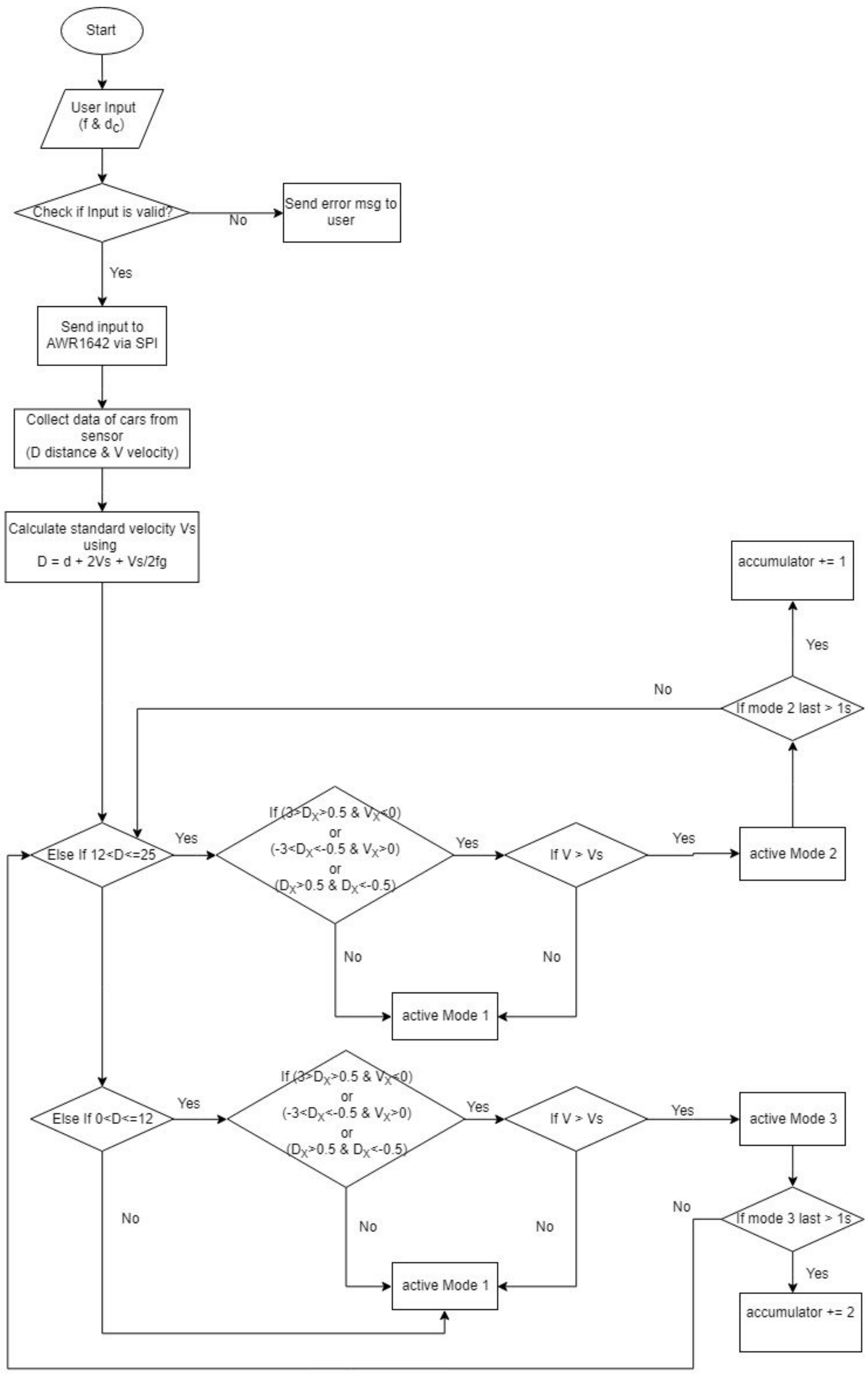


Figure 6.: Flow Chart of the algorithm in AWR1642

Requirement	Verification
1. objects are correctly mapped out in the plot on mmWave Visualizer 2. the device has activated different output device (white, red and blue LED light for output devices) accordingly 3. The device correctly records the last time for mode 2 or 3.	1. Load a test program into the board after bootloading and use TI's mmWave Visualizer to check if any objects are detected 2. Load our software program with easily accessible parameter on velocity, so that the testing vehicle can (2m/s, 1.5m/s and 1m/s for each distance threshold to test on outdoor with the bicycle being stationary with individual safety distance offset), and check if each output device has been activated correctly. a. If the device has activated different output device (white, red and blue LED light for output devices) accordingly, then input default parameters with the program to test on field with safety distance on 6 m to prevent possible collisions during. b. If not, check the SPI connection first, then check the mmWave sensor performance using mmWave Visualizer. 3. Put an object at the range that activates mode 2, while manually record how long the object stays in the range. Then, compare the manually recorded number to the number from the AWR1642.

2.2.3 User Interface

We will use the Android Studio and Google Maps SDK for Android to create a user interface to set the parameters of our sensor. [8] Also we will have a start trip and end trip button for distance recording, calculation and evaluation. Every 10 second, the GPS of the android device will call a function from the Google Mpas SDK to record the current location of the user, which will be use to calculate the total rough distance of the route. Also, every time the sensor is trigger to active mode 2 or 3 and last for at least 0.5 second, the AWR1642 will increase the points. The evaluation number will just be points / distance. A higher evaluation number means the route is more dangerous. Once the user end a trip, the user can visualize the route and how safe or dangerous it is. Assuming the velocity of a usual bicycle is 4.3m/s, and it moves with this velocity for 10 seconds, the distance it moves should be about 4.3m/s * 10s = 43m. During this time, the worst case is that our sensor triggers mode 3 every single second of the 10 seconds, which add 2 points to the accumulator. Therefore, we will get points of 20 as the worst case. Then our worst evaluation number will be 20 / 43. Moreover, the safeties number should be 0. Additionally, since the Google Maps SDK only can return the latitude and longitude, if the user is riding the bicycle on a mountain, the result could be inaccurate, so our evaluation works best on flat surface.

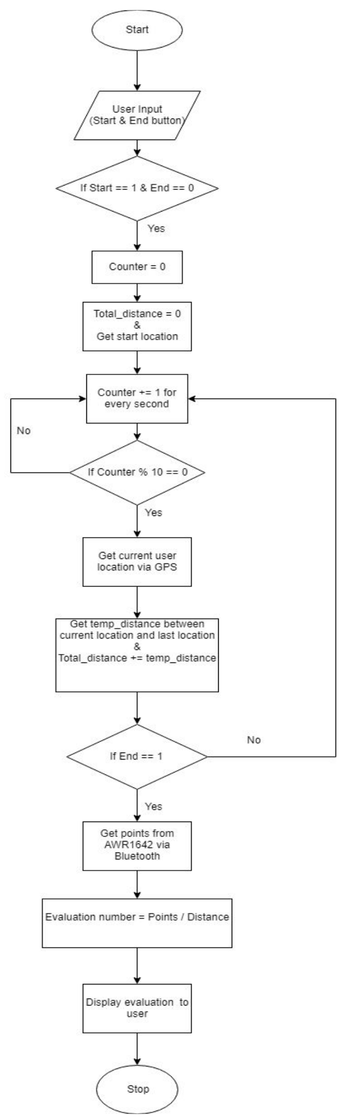


Figure 7. Flowchart for User Interface

2.3 Output

There would be three relays used between the two output LED strobe lights and speakers and the control unit. The light and speakers would be turned on only when the relay receives the signal from the control unit to close the circuit. The operating rated voltage of the LED strobe lights is 9 volts for 900 Lumens, which is equivalent to 60 watts incandescent light bulb. The maximum current of the LED strobe lights is 1100 mA. The operating voltage of the speakers ranges from 2.83 V to 3.46 V for various amplitude. The average speaker volume is about 90 dB. The maximum current of the speaker is 350 mA. They are added up to 2900 mA maximum possible current.

2.3.1 LED Strobe Lights

Requirement	Verification
4. Can blinks when relay is active 5. Provide bright enough light that the driver is able to see from at least 20 meters away	4. Connect the LED lights in series with the relays 5. Operate the relay in a desire way, check if the LED blinks as it is set up 6. Turn on the LED light during the day 7. Check if the LED light is able to be seen from 20 meters away

2.3.2 Speakers

Requirement	Verification
1. Can play sound when relay is active 2. Provide loud enough that the driver is able to hear from at least 10 meters away	1. Connect the speakers in series with the relays 2. Operate the relay in a desire way, check if the speakers operate as setup 3. Play the sound and place it around a car 4. Check if the drive is able to hear the sound from 10 meters away

2.3.3 OLED Display

An 1.3 inch monochrome OLED display is connected to the AWR1642BOOST and installed on the bicycle bar for the cyclist to view the rear situation if they need to when the cyclist tries to make a turn. The display's resolution is 128 by 64. And it communicates with the AWR1642BOOST through I2C. We are going to use 16 pixels of the monochrome OLED display to map 1 square meter of the sensing area. It should display 32 by 16 meters sensing area of the back. Considering the motions of the cars, the frame rate of the display would be 10 frames per second, meaning that the visualization on the back would be updated every 0.1 second.

More importantly, the data of the detected object including its coordinations and sizes extracted using mmWave Studio, would be mapped accordingly on the display with the scale mentioned above.

Requirement	Verification
1. The display shows the animation from the sample code correctly 2. The display updates according to the frame rate that we set and sensing data.	1. Connect SCL and SDA pins to I2C pins on AWR1642BOOST and power the display with 5 volts input power, and then run the provided sample code from the module library LiquidCrystal I2C. 2. First, set a low frame rate at 2 frames per second and have one of the teammate moved horizontally in front of the antenna and check if the display updates. a. If it updates correctly, we set to 10 frames per second and try the test again b. If not, check the I2C connection between the sensor and the board and also check the assigned starting address on the interface, to see if further modifications are needed

2.4 Bluetooth

The control unit will be accessed by android devices via Bluetooth. We will use the HC-05 Bluetooth Module with our microcontroller, which connects the devices using the serial communication. And we will start with using the serial communication at default baud rate of Bluetooth, 38400 baud rate. The Bluetooth device transmit data to the transceiver at a 2.4 GHz frequency, which would not interfere our sensor unit that works at 76 to 81 GHz. We will also make an user interface using the MIT App Inventor, which will provide an user interface to change parameters for range and relative speed for different modes of operation. [5]

Requirement	Verification
1. Can use bluetooth connected devices to modify the parameters in microcontroller and also read the current parameters of the device once connected	1. Use a test program that reflects the status to the Bluetooth Device and takes input to turn on the LED connected from the Atemega328p 2. Connect the bluetooth device to the microcontroller and Check if it could change the status of the LED and reads the current status of the LED

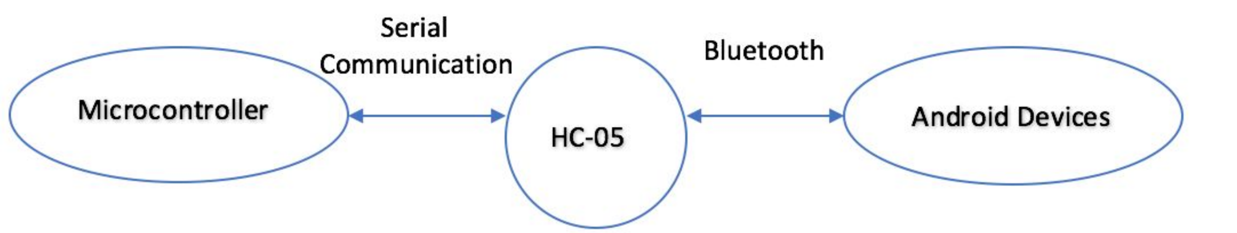


Figure 8. Bluetooth connection Diagram

2.5 Power Supply

The power supply includes three separate voltage source. One is for the output devices such as LEDs and speakers. The second one is for the pcb board. The third one is for the AWR1642boost module. The output devices require around 3045 mA in total. Since the input voltage for LEDs is 9 volts, for speakers is between 2.83 to 3.46 volts. Therefore, eight 1.5 volts alkaline batteries would be used as input power source. The pcb board requires around 35 mA. Therefore, four 1.5 volts alkaline batteries would be used as voltage input. The AWR1642boost module operates at 5 volts, and takes maximum 3900 mA at the worst scenario. Due to the high current consumption, eight 1.5 volts alkaline batteries would be used to provide input power.

2.5.1 Voltage Regulator

For the output devices, one 9 volts voltage regulator and one 3 volts voltage regulator would be used to have a constant voltage level. For the pcb board, a 5 volts voltage regulator would be used to output a stable voltage level for chips operation.

Requirement	Verification
1. Maintain a constant 3.3 volt voltage level with at least 825 mA output current	1. Connect the voltage regulator in a test circuit 2. Measure the output voltage using an

2. Maintain a constant 5 volt voltage level with at least 35 mA output current 3. Maintain a constant 5 volt voltage level with at least 3.9 A output current 4. Maintain a constant 9 volt voltage level with at least 2.22 A output current	oscilloscope, observe the ripple voltage 3. Choose proper load connected to the circuit 4. Measure the output current using an oscilloscope

2.6 Schematic

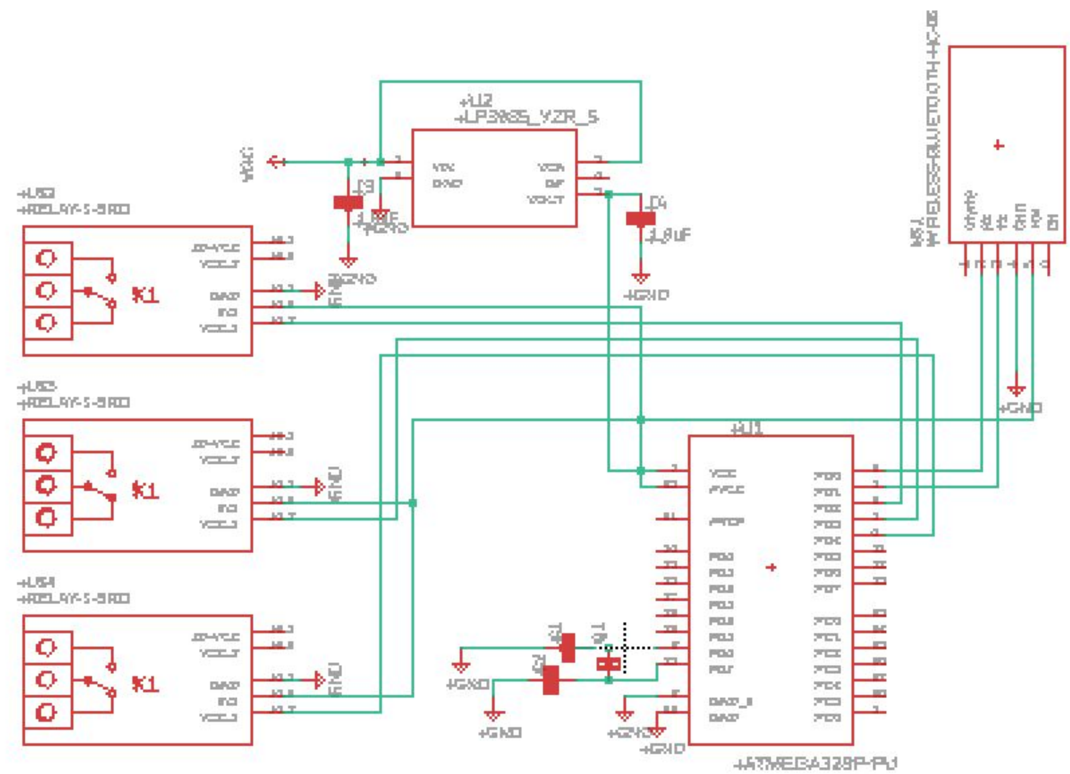


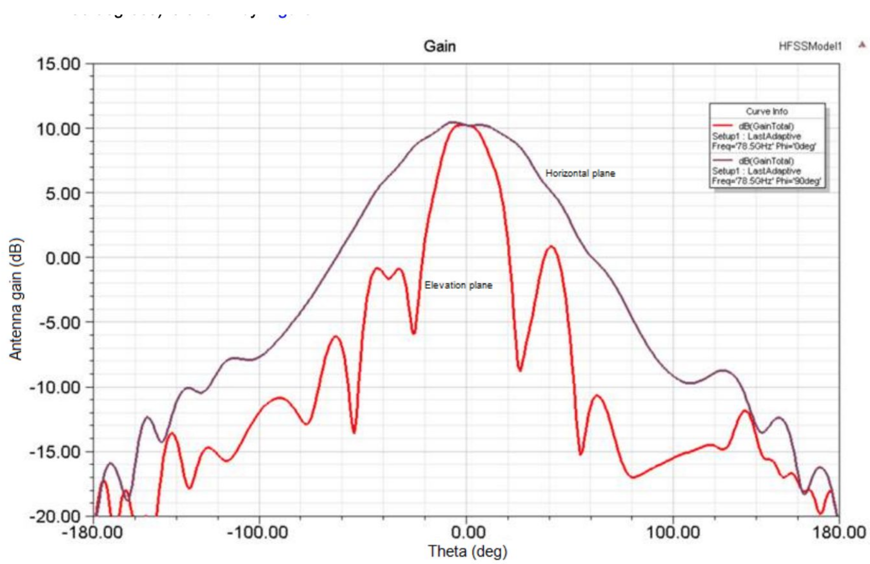
Figure 9. I/O Microcontroller Schematic

2.7 Tolerance Analysis

The AWR1642 EVM can detect objects up to 40 meters with resolution of 41 cm.

$$\text{percentage error} = \frac{|\text{eperimental value} - \text{accepted value}|}{\text{accepted value}} \qquad (2)$$

The percentage error for distance detection will be 0.41/40 = 1.025%.
And the velocity resolution is 1 km/h while the maximum detection range is 70 km/h. Therefore, the percentage error of velocity detection regarding its magnitude would be 1/70 = 1.43%.
And for the antenna's detection angle, according to the datasheet of the AWR1642 EVM, the beamwidth is about 150 degrees with +/- 75 degrees on both side, which means the performance of the antenna in desired field of view, which is 120 degrees would have a relatively stable and effective signals detection. However, given the estimated detection loss of 1dB and system loss of 1dB and other implementation loss of 2dB by mmWave Estimator provided by TI, the actual effective field of view is most likely around +/-58 degrees from the graph below. And the effective field of view would have a loss of about 1-116/120 = 3.33% in worst case.
All of the errors percentage above is estimated under frequency sweep ranging from 77 to 81 GHz, transmit and receive antenna of 10 dB gain, and maximum bandwidth of 4000 MHz and transmit power of 12 dBm using mmWave Sensing Estimator by TI.



Graph 3: Antenna pattern on AWR1642 EVM

In the power supply, there are several voltage regulators. Each of them has its specific ripples. For the 3.3 volts voltage regulator, there is ± 0.19 % error, according to the datasheet. That would range from 3.29 volts to 3.36 volts. Since the maximum voltage the speakers could take is 3.46 volts, it would be acceptable at this range. For the 5 volts voltage regulator used in the microcontroller, there is ± 0.19 % error, according to the datasheet. It ranges from 4.99 volts to 5.01 volts. Since the operating voltage for ATmega328p is from 2.7 volts to 5.5 volts, the output of voltage regulator would be within operating region even with error. For the 5 volts voltage regulator used for AWR1642boost module, it ranges from 4.95 volts to 5.05 volts. Thus, the tolerance is +/-1%. The operating voltage for AWR1642boost module is 5 volts. It might potentially cause problems due to the voltages. For the 9 volts voltage regulators for the LED, it has $\pm$ 50 mV error. However, due to the led tolerance, it would be safe in this range

3 Cost

The fixed development costs are estimated to be $20/hour, 12 hours/weeks for three people. It approximately takes 11 weeks to be done.

$$3 \cdot \frac{\$20}{hr} \cdot \frac{12hr}{wk} \cdot 12wk = \$8640$$

Part	Cost (prototype)	Cost (bulk)
1.5 volt Alkaline battery *20 (Amazon; Energizer)	$7.92	$4.80
3.3 volt voltage regulator (Digikey; AZ1117EH-3.3TRG1)	$0.44	$0.10
5 volt voltage regulator (Digikey; LM1085ISX-5.0/NOPB)	$2.03	$1.47
5 volt voltage regulator (Digikey; LP3985IM5-5.0/NOPB	$0.66	$0.35
9 volt voltage regulator (Digikey; L78S09CV)	$0.76	$0.35
Microcontroller (Digikey; ATmega328P)	$1.96	$1.63
16MHz crystal (Digikey; ABLS-16.000MHZ-B4-T)	$0.27	$0.18
Bluetooth receiver (Amazon; HC-05)	$11.11	$2.95
mm-Wave sensor evaluation module(TI; AWR1642BOOST)	$310.48	$299.00
Relay *3 (Amazon; ky-019)	$8.99	$3.56
LED strobe light *2 (Amazon; Hontiey)	$6.49	$3.34
Speakers *2 (Digikey; 102-3841-ND)	$7.40	$4.16
Total	$358.51	$321.89

4 Schedule

Week	Anchu Zhu	Jimmy Gan	Jiawen Chen
02/25/2019	Begin protocol design and programming	Begin I/O microcontroller mock assembly and configuration	Begin PCB design
03/04/2019	Continue programming, research data transmission protocols	Begin microcontroller test	PCB design version 1
03/11/2019	Begin data transmission protocols	Begin sensor unit test; continue microcontroller test	PCB design final version; begin power supply test
03/18/2019	Continue data transmission	Continue sensor unit test; complete microcontroller test	PCB design review, last version; continue power supply test
03/25/2019	Complete programming	Complete sensor unit test; start OLED display test	Begin output device test; continue power supply test
04/01/2019	Transmission test; start system integration	Continue OLED display test	Complete output device test; start system integration
04/08/2019	Continue system integration; prepare mock demo	Start system integration; check programming results; prepare mock demo	Continue system integration; check the data transmission; prepare mock demo;
04/15/2019	Mock demo; Prepare demonstration	Mock demo; Prepare demonstration, performing demonstration	Mock demo; Prepare demonstration, taking video
04/22/2019	Prepare presentation; start final report	Prepare presentation, editing demonstration video; start final report	Prepare presentation, creating powerpoint slide; start final report
04/29/2019	Complete final report	Complete final report	Complete final report

5 Ethics and Safety

There are several safety concerns for the project. Since the device will be connected to a standard flashing red light, high-intensity white strobe light, and a horn, the first concern is about the power safety. Approximately 27 volts would be needed for the project, which would be relatively safe to human. However, for the NiMH battery, overcharge may occur, which can potentially cause damage to other modules. Another concern would be that if the high-intensity white strobe light is too bright, it would potentially cause drivers behind dazzling or even unable to see. Therefore, further testing on the strobe lights will be conducted to determine the frequency of the flashing pattern.

Since the device is attached to a bicycle which would be at the outside for most of time and face raining or sunny weather. For these reason, it could possibly cause a short circuit or other potential safety problems. In that case, a waterproof case with heat-sink would be used for the device.

This project aligns with the IEEE Code of Ethics because the device itself is safe and will not do any damage to others. Moreover, it is supposed to alarm both bicyclists and car drivers and prevent potential dangers, which results in decreasing the bicycle accident involved with cars. Therefore, the project will “ hold paramount the safety, health, and welfare of the public” and “disclose promptly factors that might endanger the public or the environment”[2].

References:

[1] NHTSA’s National Center for Statistics and Analysis, “Bicyclists and Other Cyclists”, 2016. [Online]. Available: https://crashstats.nhtsa.dot.gov/Api/Public/ViewPublication/812507

[2] Ieee.org, "IEEE IEEE Code of Ethics", 2016. [Online]. Available: http://www.ieee.org/about/corporate/governance/p7-8.html. [Accessed: 29- Feb- 2016].

[3] Nacto.org. (2019). [online] Available at: https://nacto.org/docs/usdg/vehicle_stopping_distance_and_time_upenn.pdf [Accessed 8 Feb. 2019].

[4] "Road Safety Factsheet", Rospa.com, 2017. [Online]. Available: https://www.rospa.com/rospaweb/docs/advice-services/road-safety/cyclists/cycling-accidents-factsheet.pdf. [Accessed: 08- Feb- 2019].

[5] Dejan, "How To Build Custom Android App for your Arduino Project using MIT App Inventor", 2016. [Online]. Available: https://howtomechatronics.com/tutorials/arduino/how-to-build-custom-android-app-for-your-arduino-project-using-mit-app-inventor/

[6] Texas Instruments, "AWR1x Industrial mmWave Sensors Portfolio Overview", 2018. [Online]. Available:https://www.compel.ru/wordpress/wp-content/uploads/2018/04/mmwave-tehnologiya-iskusstvennogo---zreniya---dlya-promyishlennyih-i-ohrannyih-sistem.pdf

[7] Geometric Design. (2019). [online] Available at: https://www.webpages.uidaho.Unedu/niatt_labmanual/chapters/geometricdesign/theoryandc

oncepts/brakingdistance.htm [Accessed 22 Feb. 2019].
[8] Google, "Maps SDK for Android", 2018. [Online]. Availiable:
https://developers.google.com/maps/documentation/android-sdk/intro