



ILLINOIS
UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN

BICYCLE SITUATIONAL AWARENESS DEVICE

By

Manan Agarwal

Kuo Hao Chen

ECE 445, SENIOR DESIGN PROJECT

FALL 2010

TA: Mary DeRoo

27 Nov 2010

Project No. 13

ABSTRACT

Concerning of high oil prices and global warming, low-cost and green transportation is highly demanded. A bicycle can be a good substitute for the petroleum vehicles if we are able to increase bicycle safety. In order to increase bicycle safety, bikers need to be provided with a better awareness of their surroundings. Our bicycle situational awareness device would alert riders to traffic (especially cars) approaching from behind, and be able to show them the speed of the approaching traffic.

TABLE OF CONTENTS

1. INTRODUCTION	1
1.1 Features and Benefits	1
1.2 Subprojects	1
1.2.1 <i>Transmission and Reception</i>	1
1.2.2 <i>Mixer</i>	1
1.2.3 <i>Speedometer</i>	1
1.2.4 <i>Frequency Counter</i>	2
1.2.5 <i>Speed Estimator</i>	2
2. DESIGN.....	3
2.1 <i>Transmission & Reception</i>	3
2.2 <i>Mixer</i>	8
2.3 <i>Speedometer</i>	10
2.4 <i>Frequency Counter</i>	12
2.5 <i>Speed Estimator</i>	12
3. VERIFICATION	14
3.1 <i>Transmission and Reception</i>	14
3.2 <i>Mixer</i>	18
3.3 <i>Speedometer</i>	19
3.4 <i>Frequency Counter</i>	21
3.5 <i>Speed Estimator</i>	22
4. COST	23
4.1 <i>Parts Cost</i>	23
4.2 <i>Labour Cost</i>	23
5. CONCLUSIONS	24
REFERENCES.....	25
APPENDIX A	26
APPENDIX B	27

1. INTRODUCTION

As the concern over high oil prices and the environment increase, so does the demand for cheap, energy efficient solutions. Bicycling offers a useful, affordable alternative to fossil fuels (and diet pills). In order to promote bicycling, we thought it would be helpful to increase bicycle safety. As it is, a bicyclist depends on reflectors and hopes that drivers are paying attention. Lights on cars can be disorienting and only let the cyclist know that a car is approaching from somewhere at some unknown speed.

1.1 Features and Benefits

Our bicycle situational awareness device would alert riders to traffic (especially cars) approaching from the rear. In order to detect incoming traffic, our device will be mounted on the back of the bike and use radar to transmit and receive a signal. The return signal can help us determine the velocity of approaching traffic which would give the rider a much better idea of how fast an approaching vehicle is moving and how much time they have to react.

Some benefits to consumers would be:

- 1) Better awareness of traffic approaching from behind.
- 2) Cheaper, cleaner mode of transportation becomes safer.

Some product features are:

- 1) Rear facing radar.
- 2) Digital display of velocity of the nearest vehicle from behind.
- 3) Real-time velocity calculation.
- 4) Portability.

1.2 Subprojects

The design was broken into small modules, which each of them is able to be developed and tested individually and independently. The modules we decided to implement were:

1.2.1 Transmission and Reception

The antenna transmits a signal and receives the signal bounced off the approaching vehicle, and sends the reflected signal through port 3 of a circulator. This signal contains a component of the original signal as well. The circulator output is then filtered and amplified, and fed into the mixer.

1.2.2 Mixer

The mixer module consists of a double-balanced microwave mixer, a low-pass filter, and a low-noise RF amplifier. It combines the original and the received signals to produce the frequency difference. This frequency difference is the Doppler frequency shift used to calculate the velocity of the approaching car.

1.2.3 Speedometer

The speedometer measures the speed of bike, which is “Vradar.” The output goes to the speed estimator as one of its two inputs, and this information helps the speed estimator to calculate the approaching vehicle speed. Actually, we could buy a commercial bike speedometer with a cheap price. However,

since a speedometer is not hard to build, we decided to make one ourselves because it not only increases the difficulty of our project but also reduces our project cost.

1.2.4 Frequency Counter

Frequency counter is for measuring the frequency of an input signal. In this case, the input signal would be the Doppler frequency, which is a mixed signal of the transmitted signal and the received signal. This signal would be filtered and amplified first before going into the frequency counter. Theoretically, the mixed signal of the transmitted signal and the received signal is only couple kHz, which the PIC16F877A microprocessor is able to handle.

1.2.5 Speed Estimator

The speed estimator takes two inputs, which are the speed of the bike and the Doppler frequency, and outputs the speed of the approaching vehicle on 7-segment displays. The calculation would be done in a PIC by applying the Doppler frequency shift equation of a moving source with a moving target. The output is the actual speed of the approaching car instead of the speed of car with respect to the moving bike. The display would show the tens and the ones digits of the approaching vehicle speed in miles per hour.

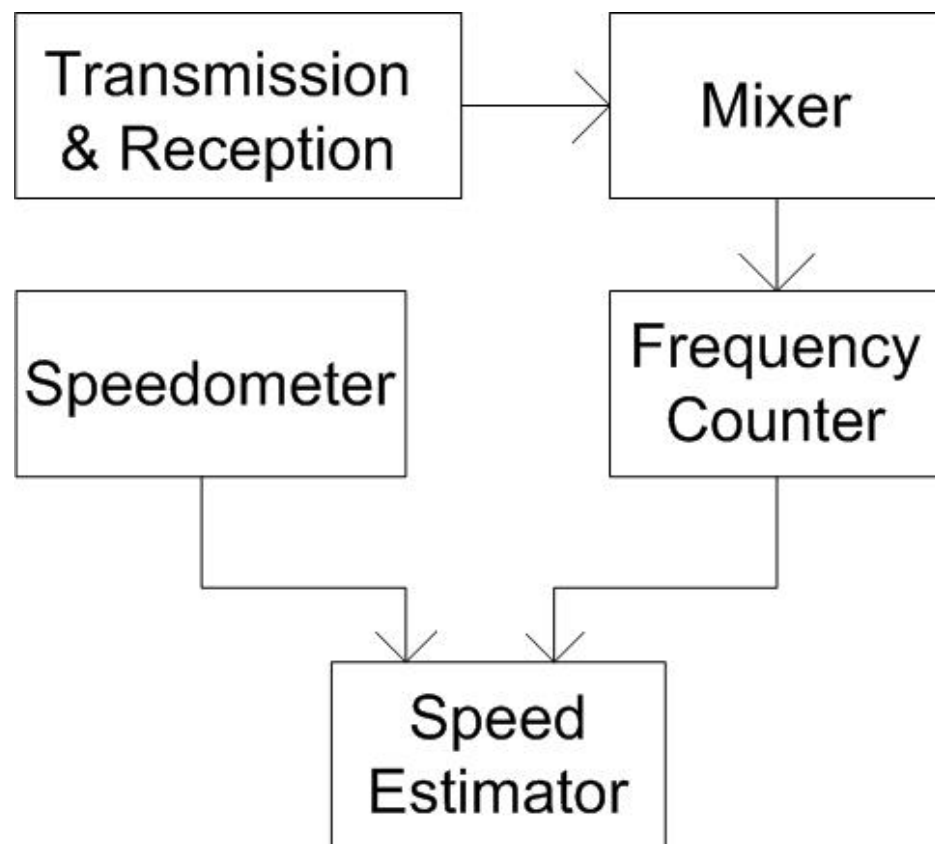


Figure 1.1 Block Diagram

2. DESIGN

In order to promote green transportation, our device should be highly portable, so we should minimize the components and the size of the device as much as possible. While minimizing the components, the performance of the device could not be degraded too much. At the same time, we must maximize the efficiency and performance of the device to provide bikers a better awareness of their surroundings and a safer trip.

2.1 Transmission & Reception

The Transmission & Reception module consists of the following components:

- VCO & Amplifier
- Circulator
- Antenna
- Microwave Filter & Amplifier

The voltage controlled oscillator transmits a pulse at a fixed rate of 8.75GHz. It is amplified and fed into the port1 of a circulator. The horn attached to port 2 receives the signal reflected off the approaching vehicle, and outputs it through the circulator port 3. The output at port 3 also contains a leak signal from the oscillator. This combined signal is passed through a microwave filter so as to filter out harmonics above and below our frequency of interest which is 8.75GHz. This signal is then amplified so that it is strong enough for the mixer to be able to use it.

Block Diagram of the Transmission & Reception module:

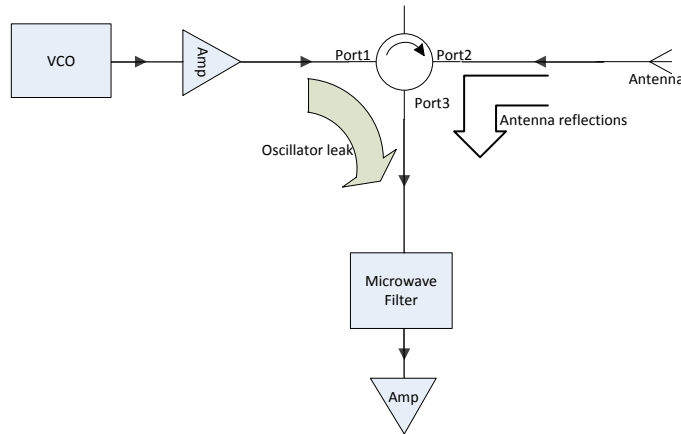


Figure 2.1 Transmission & Reception Module Block Diagram

VCO & Amplifier:

The VCO we implement is a surface mount chip (Part # DXO810900-5) ranging from 8100 – 8925MHz obtained from Synergy Microwave Corporation. According to the datasheet specification, its bias voltage (V_{CC}) is +5VDC and its output power is typically -1dBm. The tuning voltage (V_T) depends on the frequency at which the pulse is transmitted. Tuning the VCO at 8.75GHz gives us our tuning voltage which comes out to be 10.95V.

The transmitted signal is then fed into the amplifier to boost the power level. The amplifier used for this purpose is a high gain, low-noise surface mount amplifier that operates from 6-20GHz (Part # AV020515EN-2). According to the datasheet specification, the typical small-signal gain for this amplifier is $\sim 22\text{dB}$.

Once the signal is amplified, it is then fed into the circulator via Port1. Some of the signal is outputted through the circulator port 3 as the oscillator leak, which combines with the antenna reflections to give us our mixed signal.

Circulator:

The circulator allows us to send and receive signals using the same radar. It takes care of the portability factor of our project wherein we are able to use one antenna for both transmission and reception instead of using two i.e. one antenna for transmission and one for reception.

We use a simple SMA circulator which has 3 ports, namely Port1, Port2, and Port3. The insertion loss from the circulator is expected to be 0.6dB maximum.

Antenna:

For the antenna, we chose a pyramidal horn design. The reason being, we wanted a directive antenna that scanned over the three dimensional space, rather than the relatively planar radiation of the H-plane or E-plane horns. A reflector antenna could've worked as well, but a horn would be much easier to manufacture and would be more durable.

Below are some diagrams of the pyramidal horn [1]:

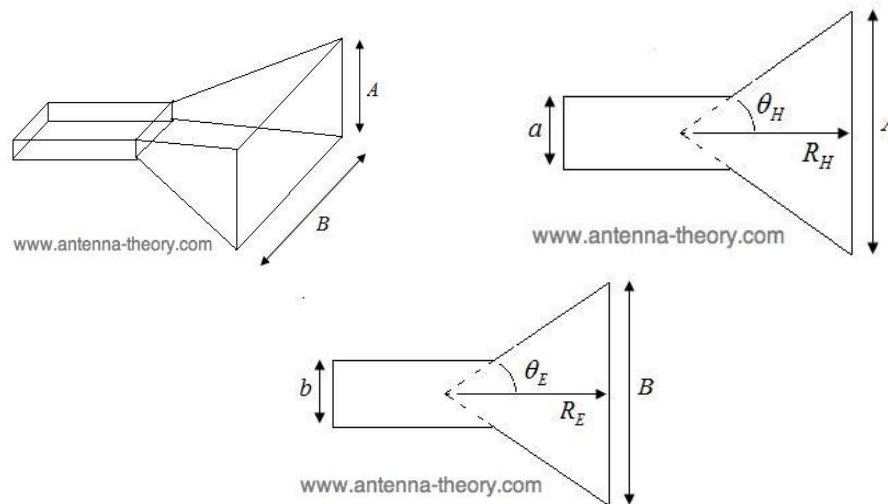


Figure 2.2 Pyramidal Horn Antenna Designs

We had initially designed the antenna and were going to have the machine shop build it for us. But in the interest of saving money building the antenna, we contacted Prof. Bernhard if she would have an antenna that would operate at 8.75GHz. Luckily for us, she was able to provide us with a couple of antennas which operated at that high frequency ranges. We measured their aperture dimensions, and finally decided to use the one which had the largest aperture, hence greatest directivity.

The antenna we used had aperture dimensions of 6.6cm x 8.7cm i.e. $A = 0.066\text{m}$ and $B = 0.087\text{m}$.

According to the gain equation for horn antennas, the gain for this antenna is [2]:

$$G_{horn} = 0.51 * \frac{4\pi}{\lambda^2} * AB \quad (2.1)$$

Where, A=0.066m, B=0.087m, and,

$$\lambda = \frac{3 * 10^8 m/s}{f (Hz)} = \frac{3 * 10^8}{8.75 GHz} = \frac{3 * 10^8}{8.75 * 10^9} = 0.034m$$

Therefore,

$$G_{horn} = 31.305 = 14.95dB$$

So, the antenna gain came out to be 14.95dB. Since, the antenna acts as both a transmitter and a receiver, this gain is accounted for twice when we calculate the total net losses from our circuit. Therefore,

$$G_{horn} (total, dB) = G_{t,horn} + G_{r,horn} = 14.95dB + 14.95dB = \mathbf{29.9dB}$$

Where, $G_{t,horn}$ is the gain of the transmitting antenna and $G_{r,horn}$ is the gain of the receiving antenna.

To see if this gain was acceptable, we calculate the total net losses from the circuit we could account for. The power received by an antenna is equal to [4]:

$$P_r (dBm) = P_t (dBm) + G_{t,horn} (dB) + G_{r,horn} (dB) - 20 \log R (km) - 20 \log f (MHz) - 32.44 \quad (2.2)$$

Where, P_r is the total received power, P_t is the transmitted power, $G_{t,horn}$ is the gain of the transmitting antenna, $G_{r,horn}$ is the gain of the receiving antenna, R is the distance from the transmitter to the receiver in kilometres, f is the frequency of operation in MHz and 32.44 is a constant.

Our VCO transmits about -1dBm, our antenna would have 14.95dB of gain, and we want to detect cars at least 10m away, so the total distance $R = 20m$, and our frequency of operation is 8.75GHz. Therefore,

$$P_r (dBm) = -1 + 14.95 + 14.95 - 20 \log 0.02 - 20 \log 8750 - 32.44 = -48.4 dBm$$

We also lose about 0.6dB from the circulator and around 7.5dB from the mixer, so the new received power is:

$$P_r (dBm) = -48.4 - 0.6 - 7.5 = -56.5 dBm$$

Hence, to nullify the losses in the circuit, we need about 57dB gain through our amplifiers.

The signal coming out of the port 3 of the circulator contains the reflections at the antenna and the leak signal from the oscillator. It would also include ambient noise signals which need to be filtered out before the signal can be fed into the mixer.

Microwave Filter and Amplifier:

Since we are operating at such high frequencies, lumped components are not suitable for a filter. So, our filter is designed using only microstrip transmission lines. Also, we needed to filter out any frequencies

above or below our frequency of interest; hence this filter had to be a bandpass filter. Such a filter consists of shorted microstrip stubs of electrical length 90 degrees and some characteristic impedance separated by 90 degree segments of 50Ω microstrip line.

To start designing the filter, we first determine the number of stubs. This number is determined by calculating the following value,

$$|\omega/\omega_0| - 1$$

We then apply this value to the ‘Attenuation v/s Normalized frequency’ graph (Figure 8.27) in the book ‘Microwave Engineering’ by Pozar [3]. To find $|\omega/\omega_0|$, we say it is equal to:

$$\frac{1}{\Delta} \left(\frac{\omega}{\omega_0} - \frac{\omega_0}{\omega} \right)$$

Where, ω_0 is our center frequency, $2\pi \cdot (8.75)$ GHz. We defined Δ to be 50%, or 0.5 and ω to be $2\pi \cdot (17.5)$ GHz. These values were chosen to decrease the number of stubs while still strongly attenuating the strongest harmonics. With these values, the number of stubs was determined to be 3, so:

$$n = 3$$

From Table 8.4 in Pozar’s ‘Microwave Engineering’ [3], as well as the impedance equation for a bandpass microstrip filter, we calculated the necessary impedances of the stubs (Table 2.1):

Table 2.1 Impedances

N	g_n	$Z_{0n}(\Omega)$
1	1.5963	12.3
2	1.0967	17.9
3	1.5963	12.3

The impedance equation for a bandpass microstrip filter is:

$$Z_{0n} = \frac{\pi * Z_0 * \Delta}{4 * g_n} \quad (2.3)$$

Where, $Z_0 = 50 \Omega$.

Here is a simple diagram of how the microwave filter looks like:

Characteristic impedance: $Z_0 = 50\Omega$, $Z_{01} = Z_{03} = 12.3 \Omega$, $Z_{02} = 17.9\Omega$

Electrical length θ : $\theta = 90$ degrees in all cases.

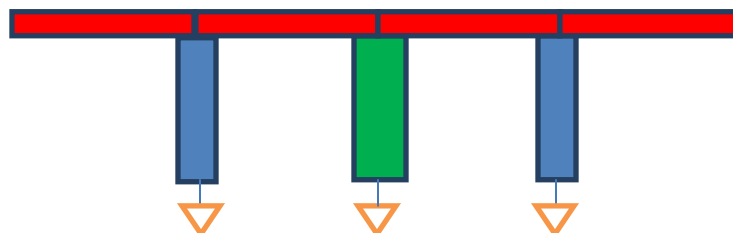


Figure 2.3 Microwave Filter Design

On the PCB, the filter is designed with the impedances just being a bunch of copper pads (microstrips) with each having length equal to its electrical length. The PCB layout of the microwave filter is as follows:

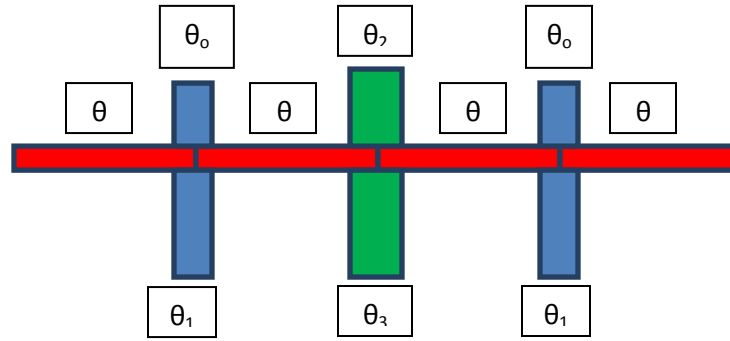


Figure 2.4 Microwave Filter PCB Layout

Where, $Z_0: \theta = 90^\circ = 1.57$ radians = 0.338 inches,

$Z_{01} = Z_{03}: \theta_0 = 30^\circ = 0.524$ radians = 0.113" and $\theta_1 = 60^\circ = 0.226$ ",

$Z_{02}: \theta_2 = 35.28^\circ = 0.166$ " and $\theta_3 = 54.72^\circ = 0.205$ " are the pad lengths.

The Puff simulation for the microwave filter gives the following response:

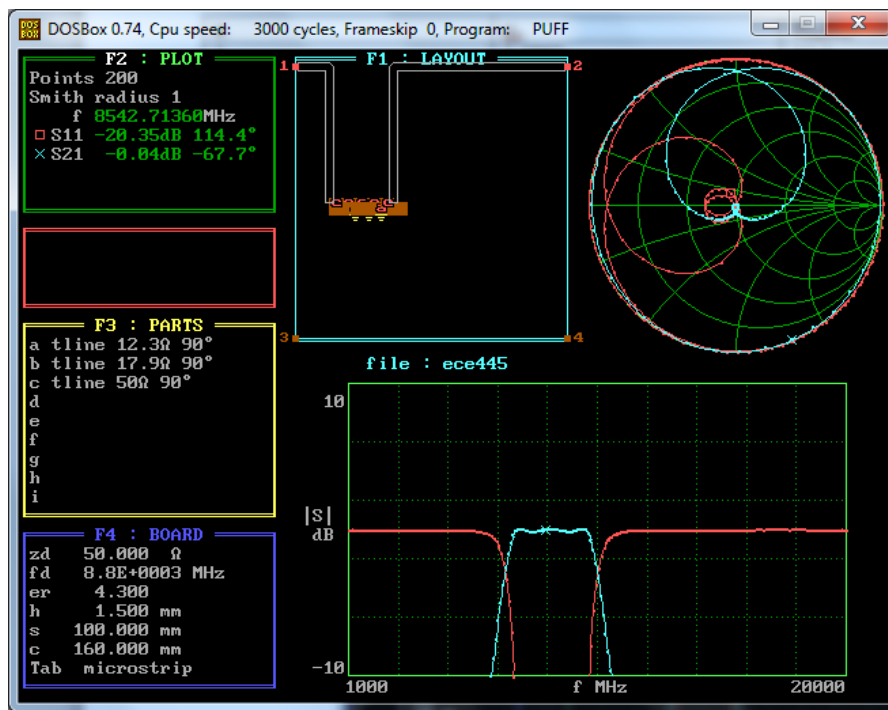


Figure 2.5 Microwave Filter Puff Simulation

This clearly shows that the designed filter is a bandpass filter (narrow band) with a pass band of around 7-10GHz. Also, there is less than 5dB attenuation in the pass band, which is good enough for our purpose.

The signal after being split is then amplified so that the mixer can use it. Another one of the high gain, low-noise amplifiers (Part # AV020515EN-2) is used for this purpose.

2.2 Mixer

The mixer module consists of a double-balanced microwave mixer, a low-pass filter, and a low-noise RF amplifier. The purpose of the mixer module is to combine the original and the received signals to produce a frequency difference which is fed into the PIC for velocity calculations.

The block diagram for the mixer module is as follows:-

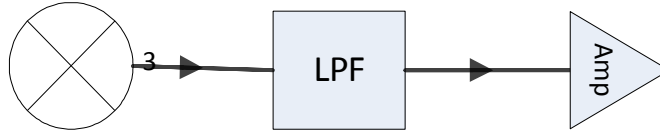


Figure 2.6 Block Diagram of the Mixer Module

Mixer:

We use a surface mount double-balanced microwave mixer (Part # SGS-5-10) to generate the sum and the difference signals of our mixed signal coming out of the microwave filter. This mixer has a typical conversion loss of $\sim 7.5\text{dB}$. The input frequencies can range from 3-19GHz, while the output frequencies (IF) can range from 0-4GHz. Also, the typical LO power is $+10\text{dBm}$ and the maximum input RF power is $+20\text{dBm}$. The loss from the mixer determines the gain required from our RF amplifier.

The 2 signals, i.e. the original and the received signals are mixed, and we get one signal at almost double our original frequency, and the other one at around 1 kHz. The lower frequency is the difference between the original signal and the mixed signal. The equations used by the mixer to generate the sum and the difference signals are [4]:-

$$IF = RF + LO \quad (2.4)$$

$$IF = |RF - LO| \quad (2.5)$$

Basically, when an 8.75 GHz and an 8.750001 GHz signal are mixed, we get one spike at 17.500001 GHz and one at around 1 kHz.

Now, we filter out the spike at 17.500001 GHz and isolate the desired 1 kHz (Doppler frequency) spike from it. Also, the output signal needs to be amplified before it can be fed into the PIC.

Low-pass filter and RF Amplifier:

The low-pass filter is implemented as a passive filter which eliminates the higher frequency signal obtained from the mixer, and passes only frequencies in the range of a couple of kHz i.e. our Doppler frequency. Since it is designed for low frequencies, we were able to build it using lumped components.

The equations used to obtain the passive components (L, C) values are [5]:

$$\frac{w}{w_c} = \frac{20 \text{ kHz}}{10 \text{ kHz}} = 2$$

Therefore,

$$\left| \frac{w}{w_c} \right| - 1 = 1$$

From the filter design graph [5], I got 'n' = 4, and using:

$$L'_n = \frac{R_0 L_n}{w_c} \quad C'_n = \frac{C_n}{R_0 w_c} \quad R_0 = 50 \Omega$$

These are the following L, C values obtained:

Table 2.2 L and C Values

'n'	g_n	C'_n (nF)	L'_n (μH)
1	1.6703	2128	
2	1.1926		3792
3	2.3361	3012	
4	0.8419		2672

The circuit designed for the low-pass filter is as shown in Figure 2.7:

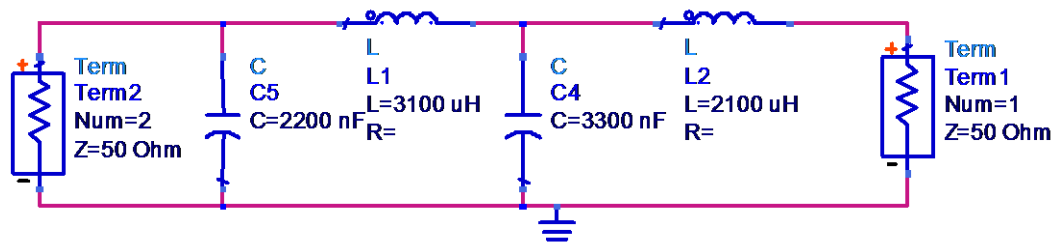


Figure 2.7 Low-Pass Filter Circuit

The forward transmission response of the filter using ADS is shown in Figure 2.8:

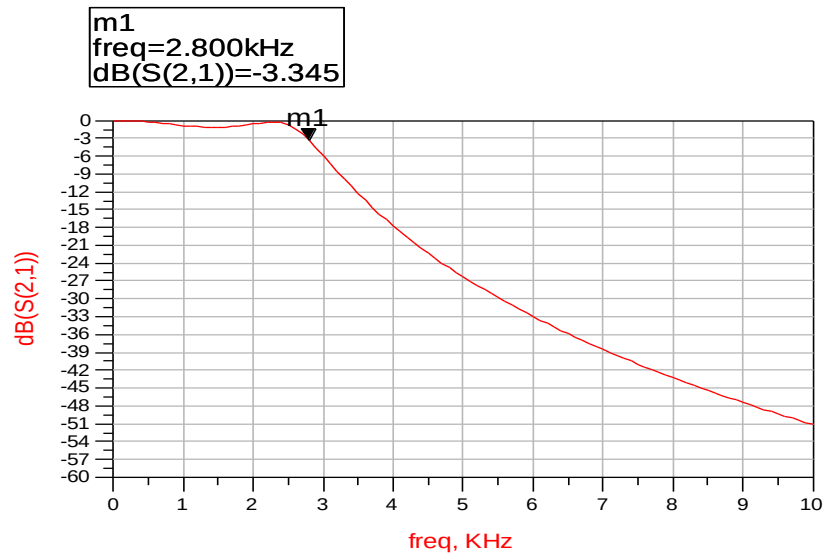


Figure 2.8 Simulated Transmission Response of the Low-Pass Filter

The filter response has a smooth tapering slope with a cut-off frequency of about 2.8 kHz. This response is excellent for our purposes since it is unlikely that any vehicle will be travelling fast enough to cause a 3 kHz or more frequency to come out of our mixer.

Once we have our Doppler frequency signal, we then boost its power so that it can be fed into the PIC which analyzes it and performs velocity calculations. For this purpose, we use a surface mount low-noise RF amplifier purchased from Digikey (Part # MSA-2111) which operates as low frequencies. According to the datasheet specification, the typical gain for this amplifier is ~ 16.5 dB and the required RF input power is $\sim +13$ dBm. The amplified signal is then inputted into the PIC microcontroller which counts the number of pulses over a period of time to calculate frequency.

2.3 Speedometer

Speedometer involves two parts: hardware and software. Hardware part is a speed sensor, and the software part uses the output of the speed sensor to do the speed calculation.

Speed Sensor:

The speed sensor is built using a Hall Effect sensor. A Hall Effect sensor is a device that would switch the circuit on/off when approached by a magnetic field. This characteristic of the Hall Effect sensor makes it a good choice for a speed sensor. The Hall Effect sensor would be mounted to the front fork of a bike, and a magnet would be attached onto one of the bike wheel's spoke. The position of the Hall Effect sensor should be directly in line with the magnet as it turns with the wheel. Whenever the magnet passes by the Hall Effect sensor, the sensor would output a pulse, and then the pulses coming out from the sensor would go into PIC2 (pin AN0) for a speed calculation.

The Hall Effect sensor we use is US5881EUA, which is south magnetic pole active. There are a total of 3 pins for this sensor: pin 1 is a supply voltage pin, which has the range from 3.5V to 24V. Pin 2 is grounded, and Pin 3 is the output. The complete schematic of speed sensor is below. The data sheet indicates that a 100nF capacitor should be placed between V_{DD} and ground pin for proper operation.

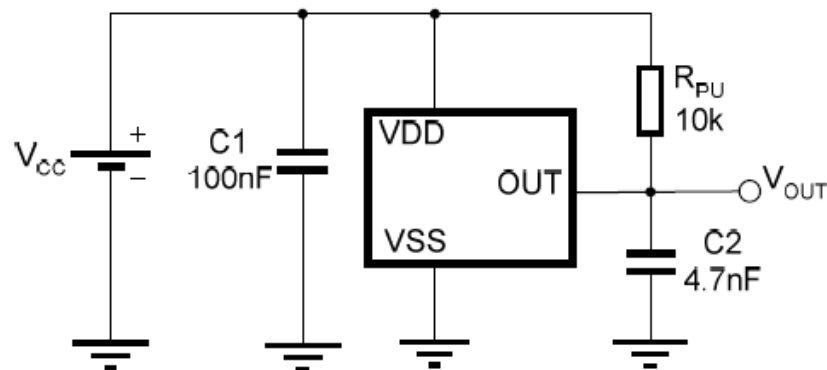


Figure 2.9 Speed Sensor Circuit

The Hall Effect sensor has an operation point, B_{OP} , at a minimum of 15mT and a maximum of 30mT, and a release point, B_{RP} , of 9.5mT. When the south magnetic field is greater than B_{OP} , it outputs low, and when south magnetic field is less than B_{RP} , the output is high. The north magnetic field does not affect the output level of the sensor at all. The graph of the unipolar switch characteristic is shown below.

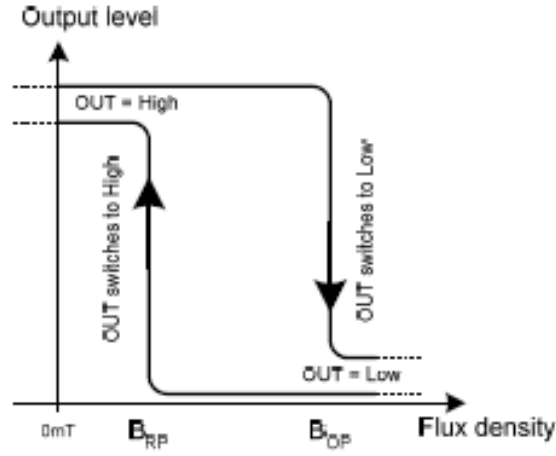


Figure 2.10 Unipolar Switch Characteristic

Speed Calculation:

The speed calculation is done using a PIC. For the speed calculation, we first need to find out the time difference between pulse_{i-1} and pulse_i, and then plug it into Equation 2.6 for finding the bike speed.

$$V_{Radar} = \frac{2\pi r}{T_{Difference}} \quad (2.6)$$

For finding the time difference, it is important to set up a real time clock/counter (RTCC) for correct clocking. First, RTCC and counter need to be initialized, and interrupts needs to be enabled [6]. These can be done by using the following code:

```
Set_rtcc(0);
Setup_counters (RTCC_INTERNAL, RTCC_DIV_256);
Enable_interrupts (INT_RTCC);
Enable_interrupts (GLOBAL);
```

There are total $20000000/(4*256*256) \approx 76$ interrupts in one second. 20000000 or 2MHz is the clock speed of the processor, and 4 is the number of clocks to execute a single instruction (This number is a hardware design, so it can't be changed). The first 256 means an 8 bit counter counts up to 256 and restarts at 0. The second 256 is a post divider of 256 which decreases the number of interrupts by 256.

The time interval between interrupts is approximately 0.013158 seconds. The variable, `spm_time`, is used to keep track of the current time and is incremented by 0.013158 when an interrupt happens. When the program recognizes the first pulse (the input level changes from high to low), it saves the time to the variable called `t_pre`. When the program reads the next pulse, it saves the time to the `t_cur` variable. By subtracting `t_pre` from `t_cur`, we get the time difference between two pulses. The flow chart of the speed calculation is in Appendix B.

The radius of the wheel is fixed and preset in the program. We use the common bike wheel radius of 27 inches. (0.6858 m) We were considering adding an interface for users to enter the radius of the wheel, but since we ran out of time, default radius was used.

2.4 Frequency Counter

The definition of signal frequency with unit of Hz is the number of oscillations per second, so we can implement a frequency counter by counting how many cycles there are in one second. We can find the number of cycles by sampling the input signal and keeping track of the current sample and the previous sample.

For each sample value, it needs to be compared with 127, which is the median of 0 and 255. If the sample is greater than 127, $f_level=1$ is assigned; otherwise, $f_level=0$ is assigned. If the level goes from low to high and back to low, this means a complete cycle. Therefore, we decide that whenever f_level goes from high to low, the number of cycles is incremented by one. The flow chart of the frequency counter is in Appendix B.

The RTCC interrupt is used to count for one second. Since there are 76 interrupts in one second, the program would check the number of cycles when the counter variable reaches 76. At this point, the number of cycles indicates the frequency of the input signal. After getting this frequency value, a decimal to binary conversion is made, and the resulting binary value would be sent from PIC1 to PIC2 as the Doppler frequency input of the Speed Estimator.

PIC1 is only used for the frequency counter in order to optimize the performance of the frequency counter. Since PIC16F877A is not a strong microprocessor, the sample rate of the frequency counter would degrade a lot if we choose to place the speed calculation, the frequency counter, and the speed estimator in the same PIC. By using PIC1 for the frequency counter and PIC2 for both the speed calculation and the speed estimator, it not only optimizes the sample rate of the frequency counter but also increases the speed of the real-time calculation. The block diagram of this two-PIC implementation is shown below, and the schematic is shown in Figure 2.12.

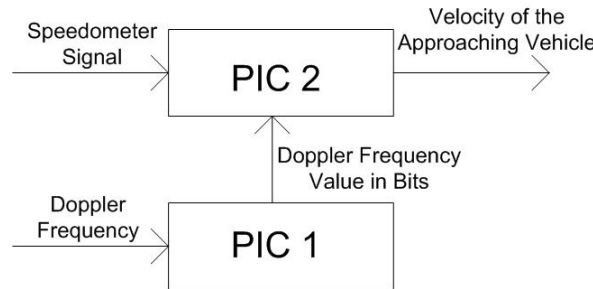


Figure 2.11 Block Diagram of the Two-PIC Implementation

2.5 Speed Estimator

The speed estimator takes in the V_{Radar} from the speedometer and f_{Doppler} from the frequency counter (f_{Doppler} would be converted from binary to decimal before the calculation) and then plugs both of them into the Doppler frequency shift equation (Equation 2.7) to find the speed of approaching vehicle [7].

$$V_{\text{Target}} = \frac{C * f_{\text{Doppler}}}{2 * \cos\theta * f_{\text{Trans}}} + V_{\text{Radar}} \quad (2.7)$$

C represents the speed of light, 3×10^8 m/s, and f_{Trans} is the frequency of the signal we transmit from the radar. In our case, f_{Trans} equals 8.75GHz. θ is the angle between the direction of transmitted signal and

the ground. For our project, we focus on detecting the speed of vehicle coming from behind, so Θ would always be kept as zero.

In order to decrease the calculation time, all the constant values can be simplified first. After simplifying all constant values, we got a cleaner equation:

$$V_{Target} = \frac{3 * 10^8 * f_{Doppler}}{2 * 1 * (8.75 * 10^9)} + V_{Radar} = 0.017 * f_{Doppler} + V_{Radar} \quad (2.8)$$

After calculating V_{Target} , it needs to be converted from meters per second to miles per hour for the convenience of users.

$$V_{Target}(mph) = 2.25 * V_{Target} \left(\frac{m}{s} \right) \quad (2.9)$$

Then, we output this number on to 2 single-digit common cathode 7-segment display. Only two digits would be shown on the display because the speed of vehicle is between 0 and 100 for most of time. The digit in ones uses outputs from PIN_B of the PIC2, and the digit in tens uses outputs from PIN_D of the PIC2.

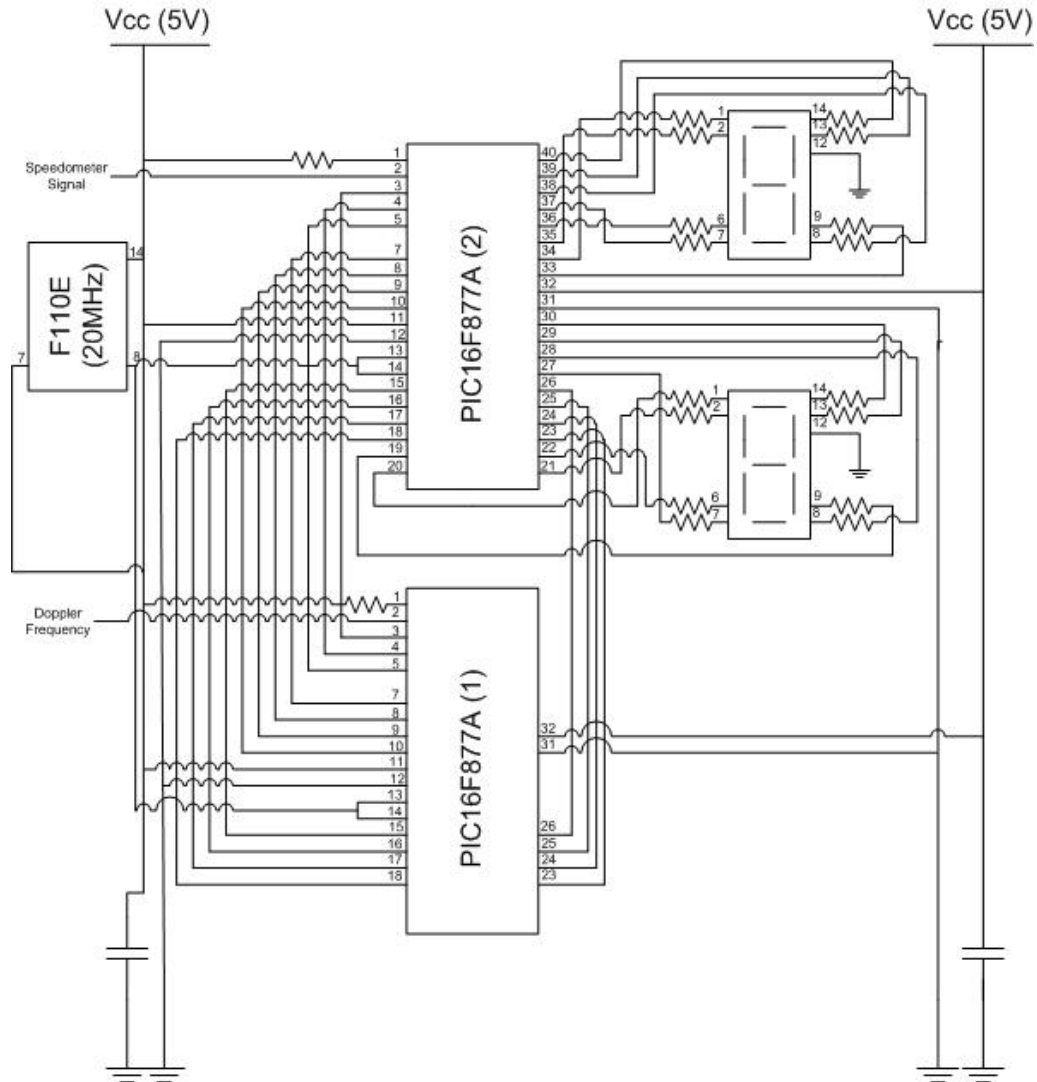


Figure 2.12 Two-PIC Implementation Schematic

3. VERIFICATION

3.1 Transmission and Reception

We first tested the voltage controlled oscillator chip independently and determined the tuning voltage required for our signal to be transmitted at 8.75GHz. We applied +5VDC power to the oscillator and altered the tuning voltage and observed the output frequency on a spectrum analyzer. The voltage which gives us an output signal at 8.75GHz is our desired tuning voltage (V_T). This comes out to be 10.95V.

This is the oscillator output response as observed on a spectrum analyzer.

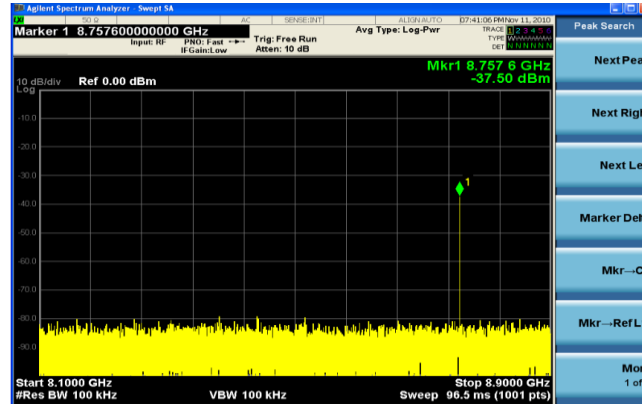


Figure 3.1 Response of the VCO output observed on a spectrum analyzer

We clearly got a spike at the desired frequency of 8.75GHz. According to the datasheet, the expected output power should have been somewhere around -1dBm, but we got a way weaker signal with the output power being ~ -37.5 dBm.

At frequencies as high as 8.75GHz, the transmitted signal experiences a greater power loss. The connecting cables and wires act like antennas and radiate power, which brings down the output power by a significant amount.

The spectrum analyzer gave us this spike even when the wire connecting the VCO output pin was brought in close contact with the BNC cable attached to the spectrum analyzer input. We did not even have to physically connect the 2 wires to observe our output response. It's just that the closer we got the 2 wires, the stronger our signal became. The reason being that since the connecting wires/ cables were radiating, so the spectrum analyzer picked up the signal whether or not it was connected to the VCO output. Figure 3.1 shows the output when the 2 wires were physically connected i.e. -37.5dBm was around the maximum power output we could obtain. Basically, at microwave range frequencies, a coupling capacitor is formed between the connecting wires which results in the signal being radiated.

This is what basically happened in our case:

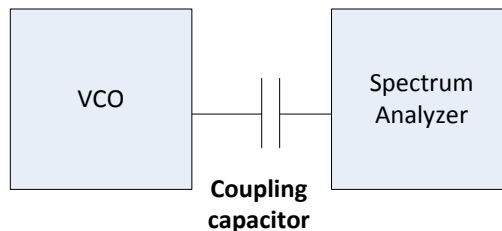


Figure 3.2 Connection Effects at High Frequency

At RF (low) frequencies, the value of the coupling capacitor is negligible; hence there are practically no radiation losses due to the connecting wires, whereas at microwave frequencies this cannot be neglected.

Next, we probed after the microwave amplifier, and observed the boost in the output power. According to the datasheet, we expected a gain of around 22dBm, but again the amplification was not as good due to the radiation losses at high frequencies.

This is the output response of the amplified oscillator signal:

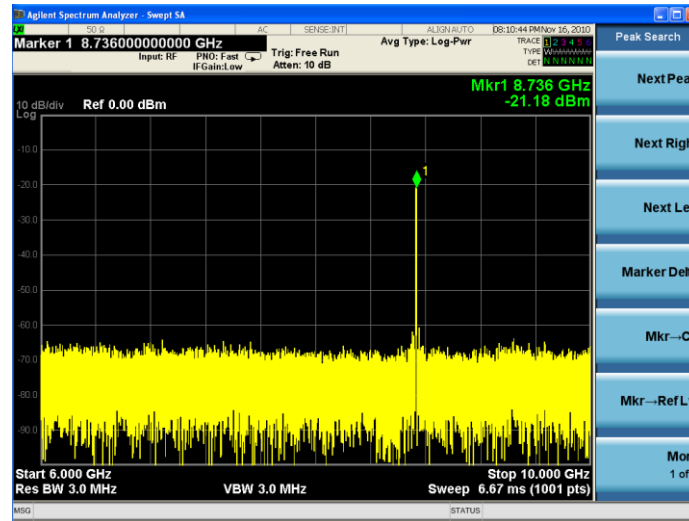


Figure 3.3 Response of the Amplified Signal Observed on A Spectrum Analyzer

Now, the output power is about -21.18dBm i.e. we got a $(-21.18 + 37.5) = \sim 16\text{dB}$ amplification. This means that instead of a 22dB gain from the amplifier, we only get about 16dB. This is again, due to the radiation losses at high frequency.

We tested our horn antenna in the anechoic chamber in the antenna lab and observed the antenna pattern response at our desired frequency of 8.75 GHz. The antenna was placed in a dampened room on a turn table which rotated 360° and gave us a pattern of the antenna reflections.

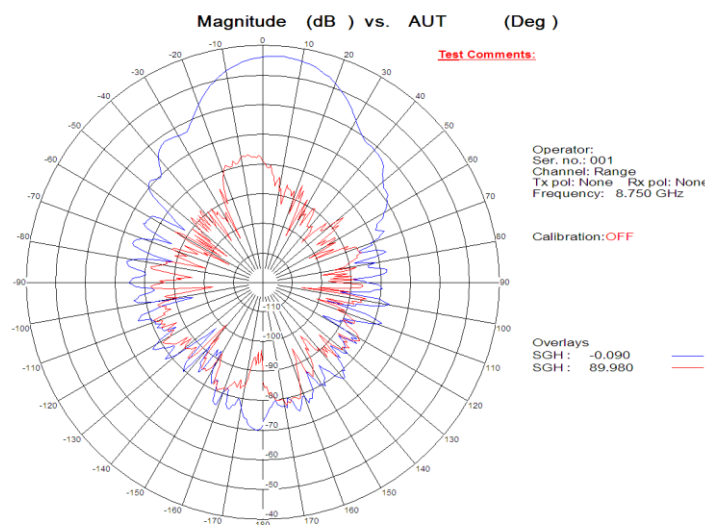


Figure 3.4 Antenna Pattern Measurement

The pattern measurement response of the antenna ensured that the antenna is highly directive, and that the antenna under test (AUT) radiates a strong signal when it is pointed at 0 degrees from the object of interest.

Next, we characterized our antenna on a network analyzer to observe the antenna gain and the input impedance.

This is the S_{22} response we got from the network analyzer:

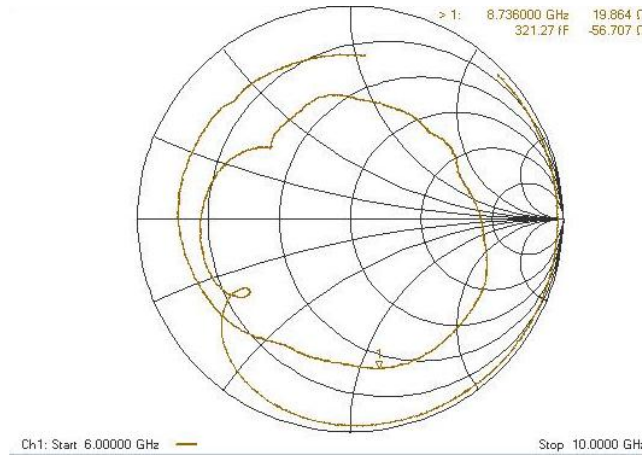


Figure 3.5 Antenna Characterization Using A Network Analyzer

Next, to measure the return signal, we pointed the antenna at a reflector (a sheet of aluminum foil), and observed the output on a signal analyzer. Since the lab had no function generator going up to the high GHz range, we used our VCO chip to feed in a signal into port 1 of our circulator; attached the antenna to port 2, and a spectrum analyzer to port 3.

Figure 3.6 is the spectrum analyzer response of the reflected (bounce back) signal from the antenna:

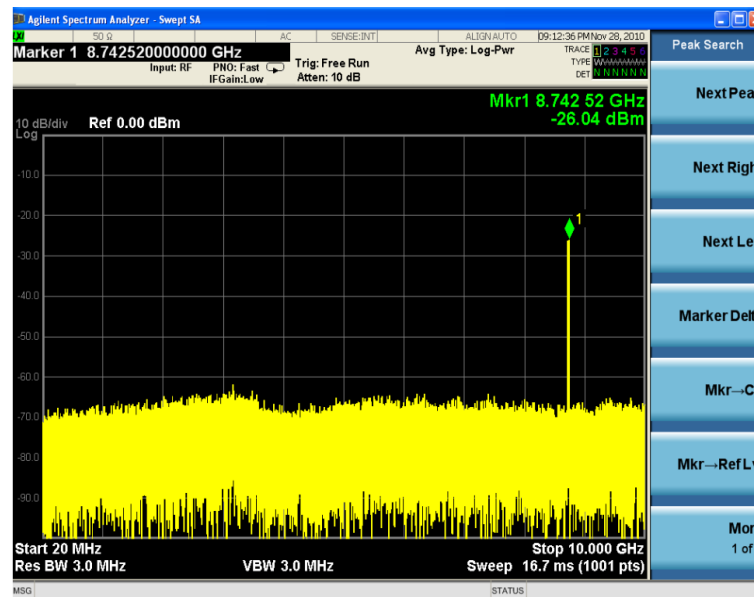


Figure 3.6 Antenna Pointing to A Reflective Surface

This signal includes the antenna reflections and the leak signal from the oscillator. This signal is then filtered and amplified and fed into the mixer to produce the frequency difference.

Next, we point the antenna to a non-reflective surface, and observe the leakage and reflections of the original signal on a spectrum analyzer.

Figure 3.7 is the spectrum analyzer response of the oscillator leak signal:

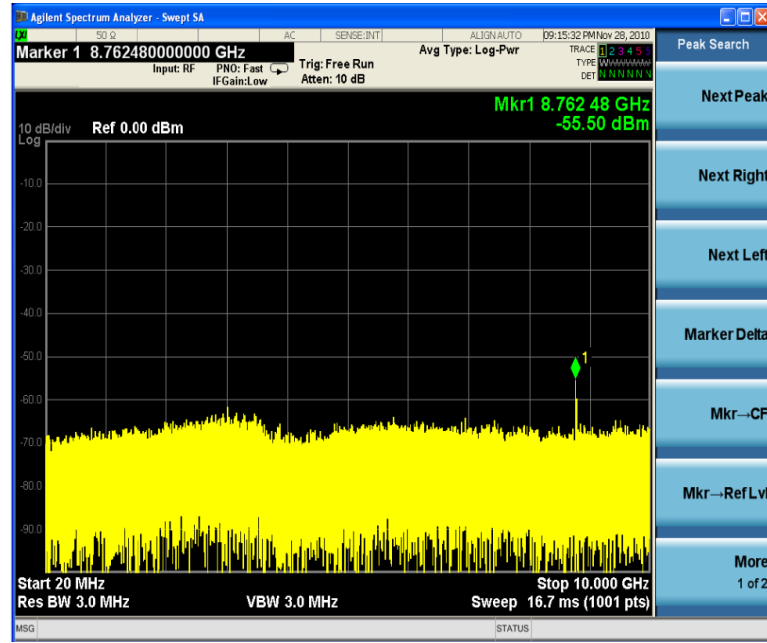


Figure 3.7 Antenna pointing to a non-reflective surface

This signal includes just the leak from the oscillator. The difference of the 2 signals (Figure 3.6 and Figure 3.7) gives us the total reflections from the antenna.

The difference is: $-26.04 - (-55.50) = 29.46\text{dB}$ which matches with the theoretically calculated total gain from the antenna i.e. 29.9dB.

The return signal we obtained (-26.04dBm) was not strong enough for the mixer to be able to use it. The mixer required the incoming signal to be at least 0-10dBm strong so as to generate the output. Our circuit did not have enough amplification to compensate for the losses due to the in-circuit components. Since we were probing our VCO output instead of having proper connectors on the PCB, we experienced power losses due to that. Also, as mentioned earlier, at high frequencies the cables and connecting wires radiate power which result in additional power losses.

We calculated the amount of amplification our circuit was falling short of. Using Equation 2.2, we again calculated the power received by the antenna. This time, the P_t (VCO) is taken as -37.5dBm instead of -1dBm.

$$P_r(\text{dBm}) = -37.5 + 14.95 + 14.95 - 20 \log 0.02 - 20 \log 8750 - 32.44$$

Including the losses from the circulator and the mixer as 0.6dB and 7.5dB respectively, we get the total losses from the circuit to be -93dBm. We have already accounted for 57dB amplification in our circuit.

Hence the extra amplification needed in order for our signal to be strong enough for the mixer to use it comes out to be $93 - 57 = 36\text{dB}$.

3.2 Mixer

We first tested the passive low-pass filter independently to ensure that it worked appropriately before we could actually mix 2 signals and test the mixer module as a whole.

For the low-pass filter, we basically needed to make sure that the filter passes only the frequencies in the range of a couple of kHz i.e. our Doppler frequency and eliminates all the higher frequencies. We first tried to observe the forward transmission response (S_{21}) of the filter on a network analyzer. But the one network analyzer in the lab that operated in the low kHz range was not calibrated, and without calibration standards the response we got was no good. Hence, we tested our filter using a function generator and an oscilloscope. At our frequencies of interest of 3 kHz or less, we obtained a 13.5V output from a 20dBm input. Above that range, we obtained approximately 5V or less with the same input. We tested over the full range of the oscilloscope, and found that the voltage went up and down seemingly randomly, but never rose above 5.5V. This ensured that our circuit was behaving appropriately. As long as the harmonics are attenuated below our desired signal, our PIC can differentiate between them and calculate the correct frequency.

Next, we moved on to test the mixed signal. Since the signal coming out of the mixer is weak, trying to observe the spikes on a spectrum analyzer gave us spurious responses. Hence, we decided to use an oscilloscope for displaying the output signal. Observing just the mixer IF output on the oscilloscope gave us nothing but a mixed unstable signal, and we could not tell the exact values of the sum and difference signals. So for testing the accurate response on an oscilloscope, it was necessary to observe the signal after it passes through the low-pass filter and the RF amplifier. Since we only care about the difference of the two input signals and the oscilloscopes in the lab only work up to 2GHz, we would like to filter out the high frequency, and only display the difference of the two input signals on the oscilloscope. We used two function generators to simulate the input signals for testing the functionality of the mixer, and the results of the tests are shown in the table below.

Table 3.1 Test Results of the Mixer

RF Input (GHz)	LO Input (GHz)	Frequency Difference (kHz)	Real Output of the Mixer + LP Filter (kHz)
3	3.000 200 000	200	215
3	3.000 150 000	150	140
3	3.000 100 000	100	105
3	3.000 010 000	10	14
3	3.000 005 000	5	7
3	3.000 001 000	1	5.5
3	3.000 000 500	0.5	5
3	3.000 000 100	0.1	4.5 ~ 4.8

From the table above, we recognized that the real output was always off from the theoretical frequency difference by a few kHz. After doing some inspections on the mixer and the low-pass filter, we decided to check the accuracy of the signals generated by the function generators with a spectrum analyzer. We first set the function generator to output a sinusoid with 3.000000500 GHz frequency, but the actual frequency shown on the spectrum analyzer was 3.000001000 GHz. After trying to output some other

frequency signals in the GHz range, we found the response on the spectrum analyzer to be always deviant by a few kHz. This showed us that at high frequencies the function generators were themselves not outputting the exact frequency signal; hence the deviation in the real output.

Apart from this issue with the function generators, there was another problem, which was the signal instability with the mixer's output signal. By saying the signal instability, it means that the signal is not fixed all the time. For example, if the frequency difference is supposed to be 200 kHz, the actual output varied in a range between 225 kHz and 195 kHz. The values of the actual output in Table 3.1 are about the medians of such ranges but not a fixed frequency value. Since we did not have proper connections for the inputs of the mixer, we did the tests by simply touching the pins on the mixer with the input wires. Sometimes, if we pressed the wires against the pins hard enough, the signal became stable and fixed. Therefore, if we could establish stable connections for the mixer inputs, the problem of the signal instability would be solved. However, because we were running out of time, we were not able to order a new PCB for having the proper connection of the mixer inputs.

After making sure the mixer was functioning correctly, it was time to integrate the mixer with the PICs, but the integration did not work at all. The integration didn't work because the PIC was not able to pick up the unstable output signal from the mixer. The signal instability of the mixer affected the integration of the two modules, so if the instability problem could be solved (by improving the input connections of the mixer), we would be able to integrate the two module successfully.

3.3 Speedometer

The hardware and software part of the module were tested individually at first, and then tested together with a spinning wheel at 32 rpm and 102 rpm.

Speed Sensor:

We first tested the speed sensor with a normal magnet, and an oscilloscope was used to display the output of the speed sensor. When no magnetic field was around the Hall Effect sensor, the output was 5V when the V_{DD} input voltage was set to 5V. When the south pole of the magnet was brought to about 0.5 cm away from the sensor, the output dropped to 0V. When the north pole of the magnet was brought close to the sensor, there was no change in the output as we expected. In order to increase the turn-on distance of the sensor and reduce the difficulty for placing the sensor on the wheel, a strong magnet was used, and the turn-on distance increased from 0.5 cm to 1.3 cm.

After the turn-on distance increased, the speed sensor was tested with a spinning wheel at 32 rpm, and the pulses showed up on the oscilloscope as the wheel was spinning. Next, we tested the sensor with a spinning wheel at 102 rpm, and it was successful as well.

Speed Calculation:

For making the test procedure easier, $T_{Difference}$ was converted to $f_{Difference}$ by using Equation 3.1.

$$f_{Difference} = \frac{1}{T_{Difference}} \quad (3.1)$$

We simply used a function generator to simulate pulses from the speed sensor and control the frequency. Therefore, this part was tested independently of the speed sensor. Checking the frequency is a method

for checking the accuracy of the time difference calculation. So, a couple frequency tests were run, and the resultant frequency ($1/T_{\text{Difference}}$) was displayed on the 7-segment displays.

Table 3.2 Tests for Frequency Counter of the Speedometer

Input Frequency (Hz)	Output Frequency (Hz)	% Error
2.50	2.53	1.2%
1.50	1.49	0.7%
1.00	0.99	1%
0.13	0.129	0.7%

As we see from the results, the percentage errors are always low showing that the time difference calculation works well. The input frequencies were chosen because they appear to be in the reasonable range of the speed of the bike.

After ensuring that the time difference was in good shape, we used Equation 2.6 to find the speed of the bike, V_{Radar} . The test results are shown here:

Table 3.3 Tests for Output Speed of the Speedometer

Input Frequency (Hz)	Theoretical Speed (m/s)	Output Speed (m/s)	% Error of speed
5	21.545	21.843 (48.86 mph)	1.4%
4	17.236	17.273 (38.64 mph)	0.2%
3	12.927	13.093 (29.28 mph)	1.3%
2.5	10.772	10.974 (24.55 mph)	1.8%
2	8.618	8.639 (19.32mph)	0.2%
1.5	6.464	6.401 (14.32 mph)	0.9%
1.3	5.602	5.534 (12.38 mph)	1.2%
1	4.309	4.342 (9.71 mph)	0.8%
0.5	2.154	2.156 (4.82 mph)	0.1%

From the result above, we can tell that the speed calculation is fairly accurate because the percentage error is always around 2% or less. Therefore, the speed calculation module is successfully implemented.

After individual parts were tested thoroughly, both parts were integrated and tested together with the spinning wheel. When tested with the 32 rpm (about 0.5Hz) spinning wheel, the result was 4 mph, which matches the data in Table 3.3. When tested with the 102 rpm (about 1.7Hz) spinning wheel, the result showed 12 mph, which is slightly slower than the theoretical result. To ensure the speed calculation was correct, we tested the calculation again with a function generator at 1.7Hz, and then the result we got was 16 mph. This test proved that there was no problem with the speed calculation but the spinning wheel was actually spinning slightly slower than 102 rpm.

3.4 Frequency Counter

At first, we were using the same PIC for frequency counter and all the other calculations. However, this method degraded the sample rate a lot. With module integration, the high limit of the input frequency decreases to about 100 Hz which was too low for our project. This problem happened because more code was in the while loop after the integration. Since there was more code, it took more time for the processor to finish one round of the execution, which slowed down the sample rate of the signal as well. In order to fix the problem, two-PIC implementation was adopted. By using this new method, the frequency counter is able to perform independently from the rest of the calculation and also optimizes its sample rate.

The frequency counter is tested by using a function generator for inputting signals with different frequency and 7-segment displays for displaying the result of the calculation. The results are shown below.

Table 3.4 Tests for f_{Doppler} Frequency Counter

Input Frequency (Hz)	Output Frequency (Hz)	% Error
12	12	0%
145	144	0.69%
999	997	0.20%
1567	1565	0.13%
4678	4651	0.58%
7788	7755	0.42%
8896	8852	0.49%
9255	9213	0.45%
9367	9262	1.10%
9588	9214	3.90%
10000	9148	8.50%

By looking at the result, we see that the frequency counter crashes around frequency 9300 Hz. This is because the sample rate of this program is about 18600 samples per second. The sample rate is found by writing a helper function to keep track of the number of times the program samples the input signal in one second and output the number on the 7-segment displays. Although the module crashes when the input frequency is higher than 9300 Hz, this performance should be good enough to handle our project. For example, if the speed of bike (V_{Radar}) is 15 mph and the speed of approaching vehicle (V_{Target}) is 300 mph, the resulting Doppler frequency would be 7432 Hz, which is still lower than the crashed frequency. Therefore, the frequency counter should be good enough for our purpose.

There is no need to improve the accuracy of this frequency counter because the percentage error of input frequency is usually below 0.5%, which is already very precise. As long as we are keeping the input frequency lower than 9300 Hz, the frequency counter should work perfectly.

3.5 Speed Estimator

Two function generators were used for testing the speed estimator. One function generator simulated the Doppler frequency, and another one was used for the output of the speedometer. Below are the test results:

Table 3.5 Tests for V_{Target} of the Speed Estimator

Doppler Frequency (Hz)	Speedometer Signal (Hz)	Theoretical V_{Target} (m/s)	Output V_{Target} (m/s)
1250	5	43.272 (96.79 mph)	43 (96 mph)
500	5	30.414 (68.03 mph)	30 (67 mph)
900	4	32.702 (73.15 mph)	32 (72 mph)
400	3	19.950 (44.62 mph)	19 (42 mph)
100	3	14.807 (33.12 mph)	14 (31 mph)
60	2.5	12.00 (26.84 mph)	12 (24 mph)
500	2	17.210 (38.49 mph)	16 (36 mph)
70	2	9.839 (22.00 mph)	9 (20 mph)
90	1.3	7.076 (15.82 mph)	6 (13 mph)
500	0.5	10.727 (23.99 mph)	10 (22 mph)
300	0.5	7.298 (16.32 mph)	7 (15 mph)
1250	0.2	22.279 (49.83 mph)	21 (47 mph)

From the result above, the error is usually +/- 2 mph of the theoretical output. However, the percentage error increases when Doppler frequency is small (below 100 Hz). This problem is due to the choice of integer variable type used in the calculation because integer type calculation is much faster than float type calculation. Since the integer type variable is used, decimal numbers of the calculation results are ignored by the variable, so the percentage error increases when Doppler frequency value is small. If float type variable is used instead, the calculation time would become too long, and would eventually affect the functionality of the speed estimator and cause the speed estimator to not work properly.

There are four important conditions we need to check for ensuring functionality of the speed estimator. The four conditions are: both car and bike are moving, both car and bike are stationary, and only one of them is moving. As long as the car is stationary, the output should be zero. If car is moving, then the output should be a non-zero value. This is because the speed estimator calculates the actual velocity of car instead of the velocity of car relative to the velocity of bike. The reason that we didn't choose to calculate the relative velocity of the car to the bike is that if the bike is moving at some speed and the car is stationary, the output would be a non-zero value, which can confuse the user. The user might think that the value shows up because a car is approaching but the car is actually stationary.

Table 3.6 Tests of the Four Conditions

Bike \ Car	1250 Hz (Moving)	0 Hz (Stationary)
5 Hz (Moving)	$V_{\text{Target}} = 96 \text{ mph}$	$V_{\text{Target}} = 0 \text{ mph}$
0 Hz (Stationary)	$V_{\text{Target}} = 45 \text{ mph}$	$V_{\text{Target}} = 0 \text{ mph}$

4. COST

4.1 Parts Cost

There were some parts that were already available with the University, and they were used and returned to the University with no cost.

Table 4.1 Parts from the University

Part	Model Number	Quantity	Total Cost
Circulator	MA138K#145	1	\$150.00
Horn Antenna	EER545759	1	\$95.00
Microcontroller	PIC16F877A	2	\$14.68
Crystal Oscillator (20MHz)	F1100E	1	\$2.24
Common Cathode 7-Segment Display	LSD321X-XX	2	\$1.88
2.2mH RF Choke		1	\$2.27
3.3mH RF Choke		1	\$1.57
100pF capacitor		6	\$1.26
2.2pF capacitor		1	\$0.06
2.2uF capacitor		2	\$0.17
4.7uF capacitor		2	\$0.30
0.01uF capacitor		4	\$0.09
3.3uF capacitor		1	\$0.17
2.5uF capacitor		1	\$0.85
10kΩ Resistor		1	\$0.04
47kΩ Resistor		2	\$0.06
Total:			\$270.64

These were the parts we actually bought for the project.

Table 4.2 Parts Actually Bought

Part	Model Number	Quantity	Total Cost
Microwave Amplifier (6-20GHz)	AV02-0515EN-2	2	\$88.00
Amplifier (0-500MHz)	MSA2111TRIG	1	\$3.68
Microwave VCO (8.1-8.925 GHz)	DXO810900-5	1	\$86.35
Microwave Mixer (3-19GHz)	SGS-5-10	1	\$21.95
Hall Effect Sensor	US5851LUA	1	\$2.00
Total:			\$201.98

4.2 Labour Cost

2 people * \$50 /hr * 150 hrs * 2.5 = \$37,500

Total Project Cost: \$270.64 + \$201.98 + \$37,500 = **\$37,972.60**

5. CONCLUSIONS

We have learned a great deal of things from this project. The project not only gave us an opportunity to apply the knowledge we learned from classes on a real world problem but also taught us the efficient ways of project management. Most importantly, it helped us understand the significance of teamwork and the ways to cooperate with each other.

Accomplishments:

Our device accurately calculates the velocity of approaching vehicle under the ideal condition, where the mixer module output is stable. The device is able to detect the vehicle at any moving velocity up to 355 mph with efficient real-time calculation. The display can output the vehicle velocity with a refresh rate of one second and alert the biker if a moving vehicle is approaching from the behind.

Uncertainties:

Because of the connection problem in our transmission & reception module and mixer module, we were unable to generate a signal which was strong enough for the mixer to use it. Once we have proper connections the power losses would reduce but we would still need to determine how much amplification would be required for our purpose. Finally, tests under different weather condition like rainy and cold weather were never performed, so we are not sure how the different weather condition would affect the performance or the accuracy of the device. Therefore, more rigorous testing could to be done to ensure the performance of the device.

Future Work / Alternatives:

There are couple improvements we can make to successfully integrate the whole system. First, better connections are required for both the transmission & reception module and the mixer module by modifying the PCB and adding SMA connectors. Also, a stronger amplification of the transmitted and received signal is needed. For alternatives, a DSP chip instead of a mixer chip can be used for mixing the transmitted and received signals. By using this method, it does not require the input signal to be very strong in order to mix the signals and also reduces the cost of our product. In addition, the 7-segment display can be replaced by a LCD display for showing more information like the bike speed. One possible expansion of this project is to add distance measurement between the bike and the vehicle by chirping the signal.

Ethical Concerns:

From an ethical point of view, we are responsible for checking our device working correctly without any error because any error could cause a potential harm to users or become a threat of the public safety. If the device is not perfectly done, we must be honest in stating the problem of the operation of the system. We are also liable for making sure that the device performs in good accuracy. Otherwise, users must be informed of the potential difference between the real situation and the calculation result, so users will have a better awareness to avoid possible accidents. Lastly, it is necessary to seek and accept criticism of the product and fix errors once they are learned.

REFERENCES

- [1] "Horn Antenna – Intro," *Antenna-Theory.com*, 12 October 2010,
<<http://www.antenna-theory.com/antennas/aperture/horn.php>>.
- [2] W. L. Stutzman and G. A. Thiele, *Antenna Theory and Design*, USA: John Wiley & Sons Inc., 1998, pp. 312.
- [3] D. M. Pozar, *Microwave Engineering*, USA: John Wiley & Sons Inc., 2005, pp. 396-397.
- [4] S. J. Franke, "Receivers," class notes for ECE 453, Department of Electrical and Computer Engineering, University of Illinois at Urbana-Champaign, Fall 2009.
- [5] D. M. Pozar, *Microwave Engineering*, USA: John Wiley & Sons Inc., 2005, pp. 392.
- [6] PIC – Reference – Sept2004, 16 October 2010,
<http://courses.engr.illinois.edu/ece395/docs/PIC%20-%20reference-Sept2004.pdf>.
- [7] K. Blattenberger, "Doppler Frequency Shift Equations Formulas," RF Café, 24 September 2010,
< <http://www.rfcafe.com/references/electrical/doppler.htm> >.

APPENDIX A

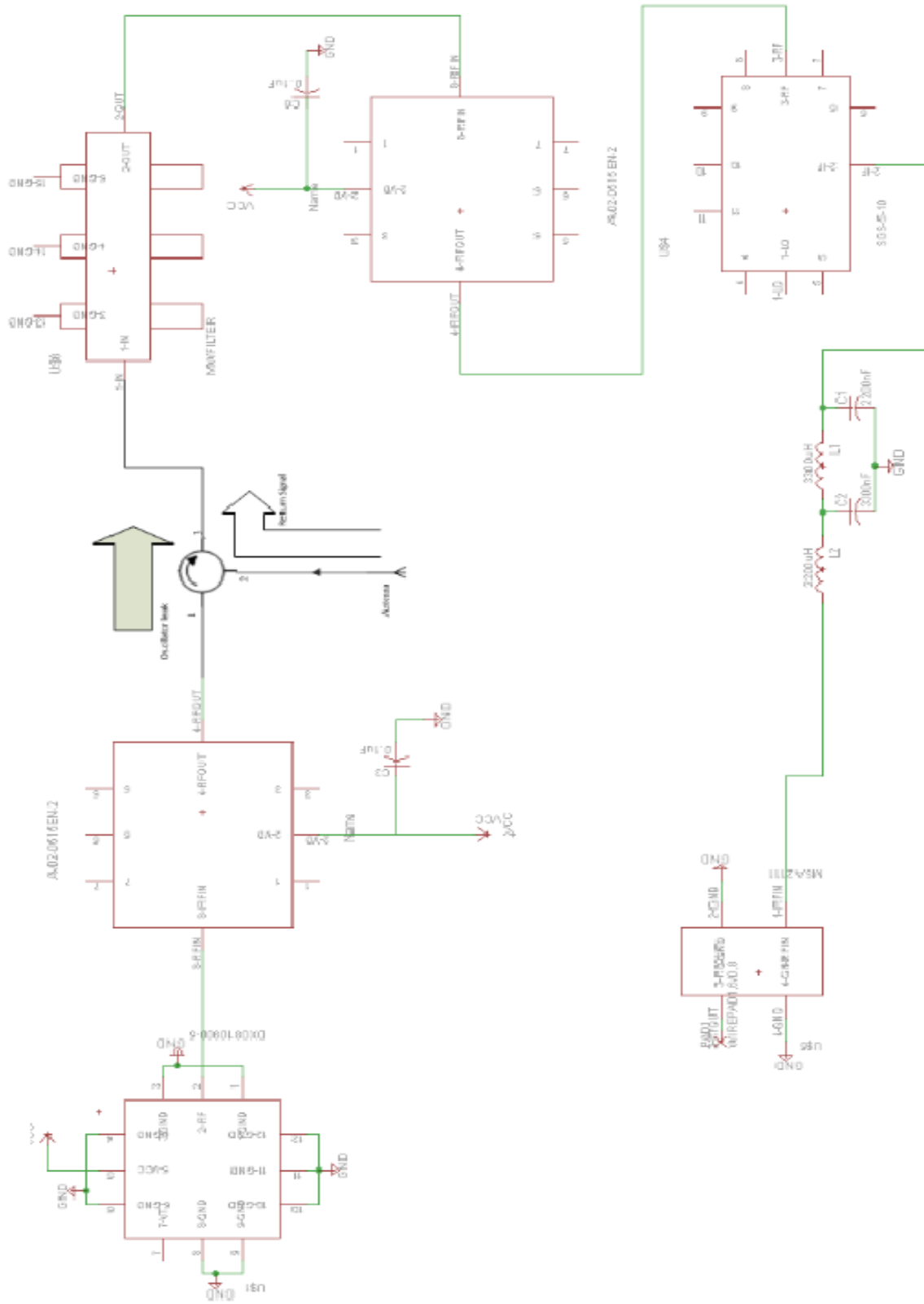


Figure A Schematic of Transmission & Reception Module and Mixer Module

APPENDIX B

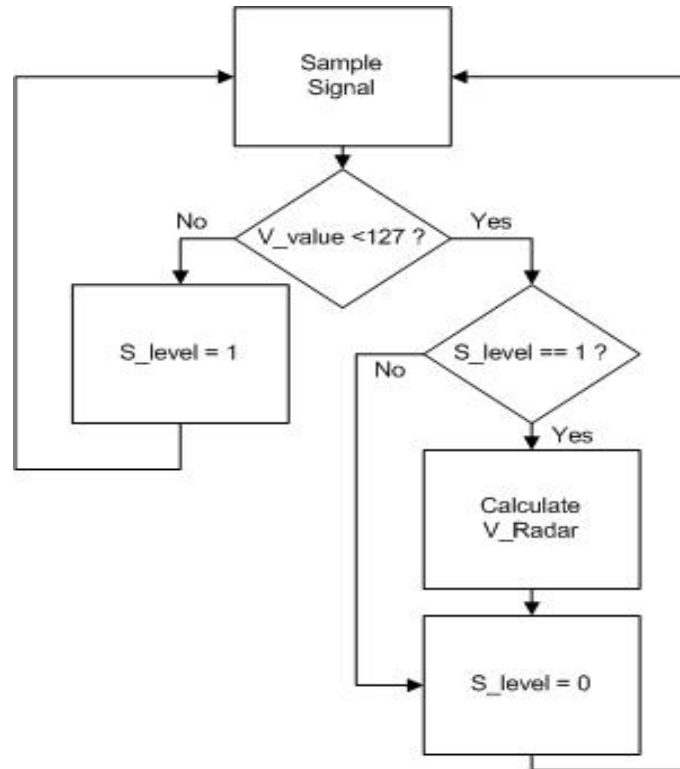


Figure B.1 Speedometer Flow Chart

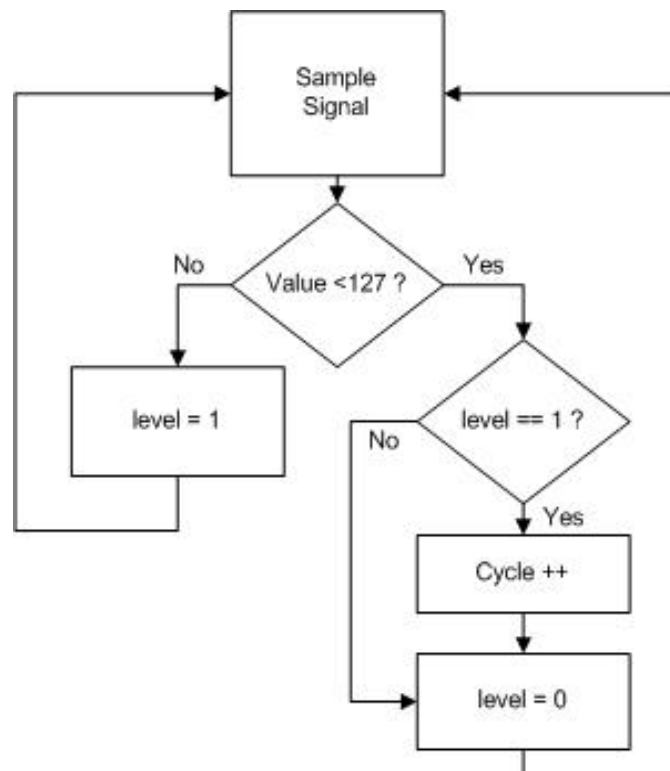


Figure B.2 Frequency Counter Flow Chart