

**Table Of Contents:**

1. **Introduction**
2. **Tasks**
3. **Results and Discussion**
4. **Conclusion**
5. **References**

**Introduction:**

Radio waves have the intriguing nature where they could be reflected by metallic surfaces and refracted by dielectric surfaces, and this nature led to the development of the RAdio Detection And Ranging “RADAR”. It is a device that keeps on emitting and receiving Radio waves, those radio wave get reflected by a surface “Echo”, and is received back by the RADAR , to detect different objects, the minimum distance between them should be .

**Tasks**

**Task 1:**Plot the transmitted pulse at baseband (i.e., without modulation) in the time domain.

**Task 2:**Calculate the unambiguous Range (), and the range resolution ().

**Task 3:**Plot the transmitted signal and the echo. You will need to add a suitable amount of noise to the received echo.

**Task 4:**plot the correlation between the transmitted signal and the noisy received echo.

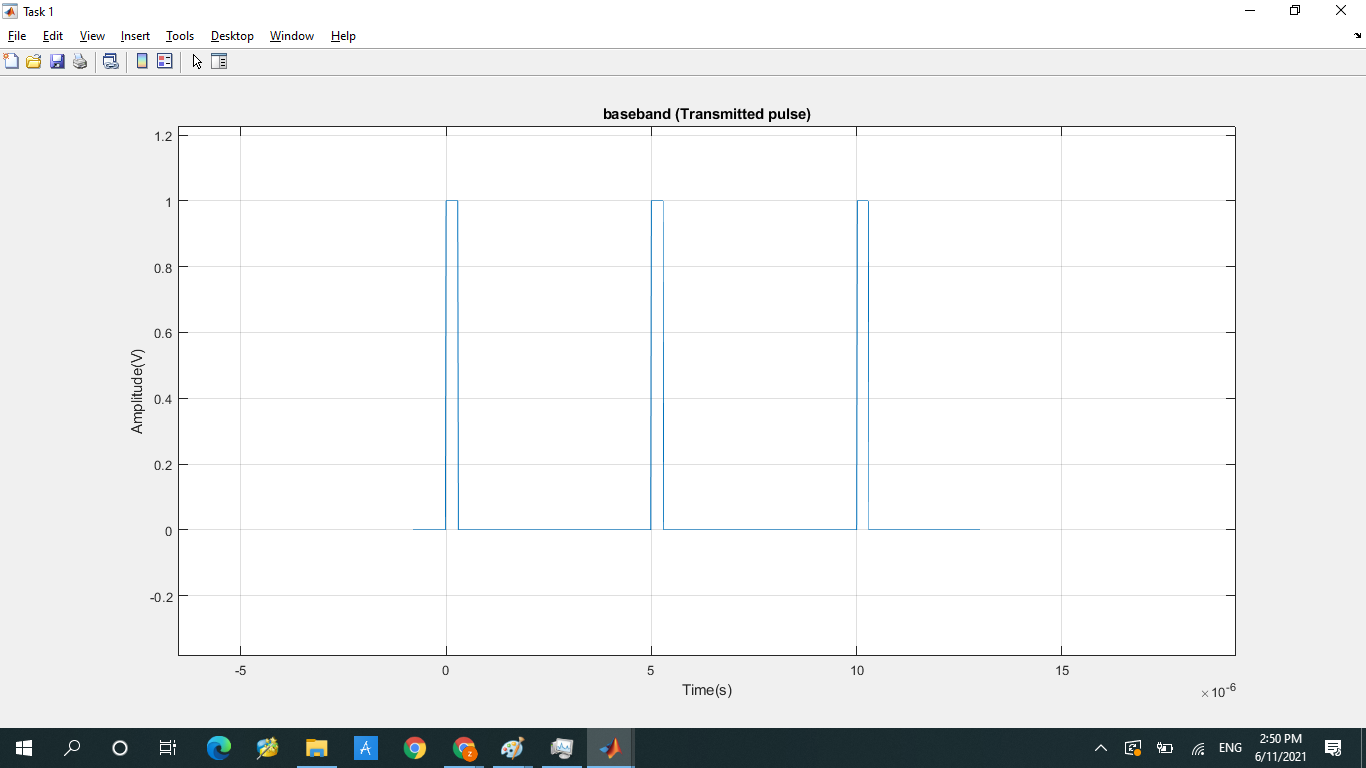
**Task 5:**using multiple pulses and plot the average correlation plot, then use the averaged plot to calculate the target range.

**Results and Discussion:**

**Task 1:**

This is the transmitted pulse

pulse width is which was calculated from the given sample numbers “18” when the sample frequency is 60 MHz; therefore, the frequency for 1 sample is and the time for . The transmitted peak power is 1.5 which is the amplitude.

****

**Code Snippet:**

**%Task 1 Plot the transmitted pulse**

**fs= 60e6;%sampling frequency**

**ts=1/fs;%time sampling**

**t=-8e-7:ts:1.3e-5;% domain of time**

**amp=1;**

**pulseWidth=3e-7;**

**transForumla=@(t) [((t>=0)&(t<3e-7)).\*(amp) + ((t>=5e-6)&(t<5.3e-6)).\*(amp)+ ((t>=1e-5)&(t<1.03e-5)).\*(amp)];**

**transmittedSignal=transForumla(t);**

**figure('NumberTitle','Off','Name','Task 1');**

**plot(t,transmittedSignal);xlabel('Time(s)'); ylabel('Amplitude(V)');**

**title('baseband (Transmitted pulse)');**

**grid on;**

**Code Logic:**

Sampling time is 1/fs which is how often the sample is taken, function handle was used t force the amplitude to 1 when the pulse is on

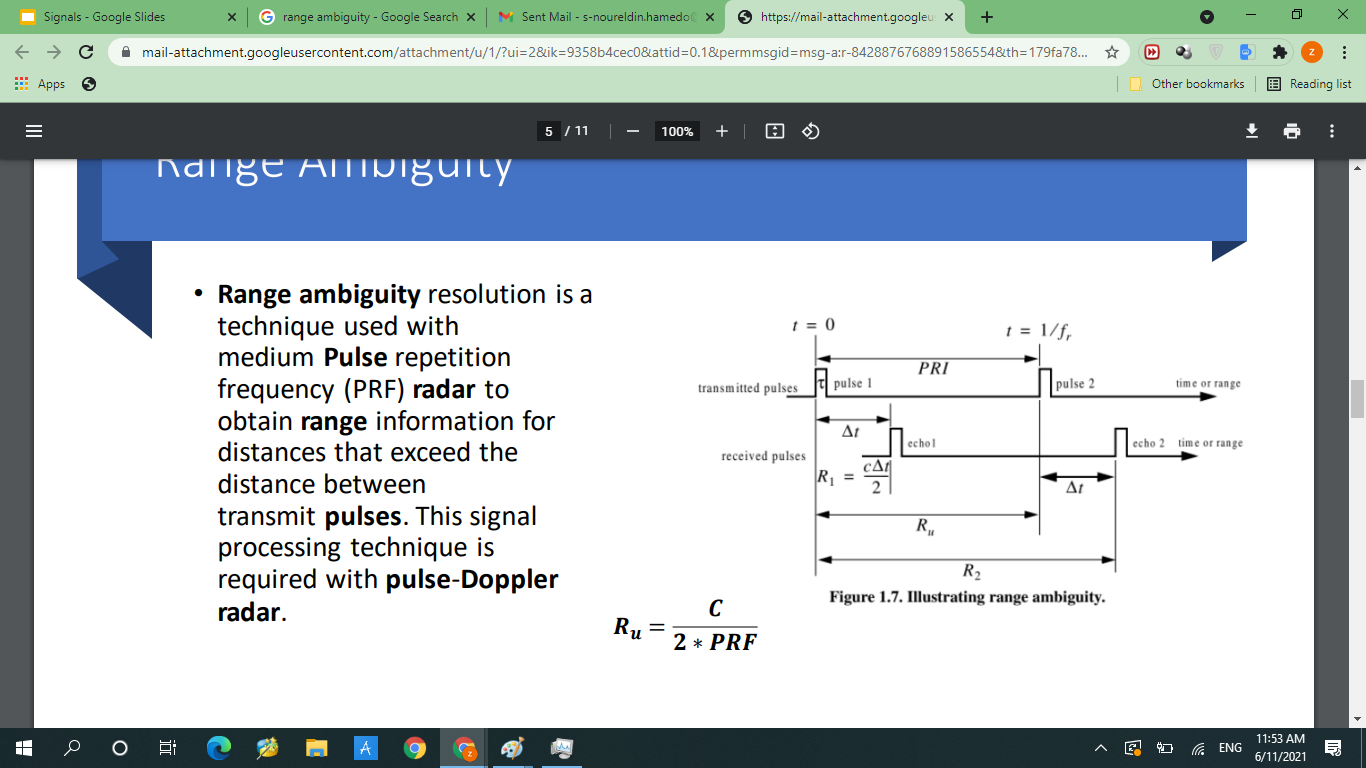
**Task 2:**

Unambiguous Range

Pulse Resolution

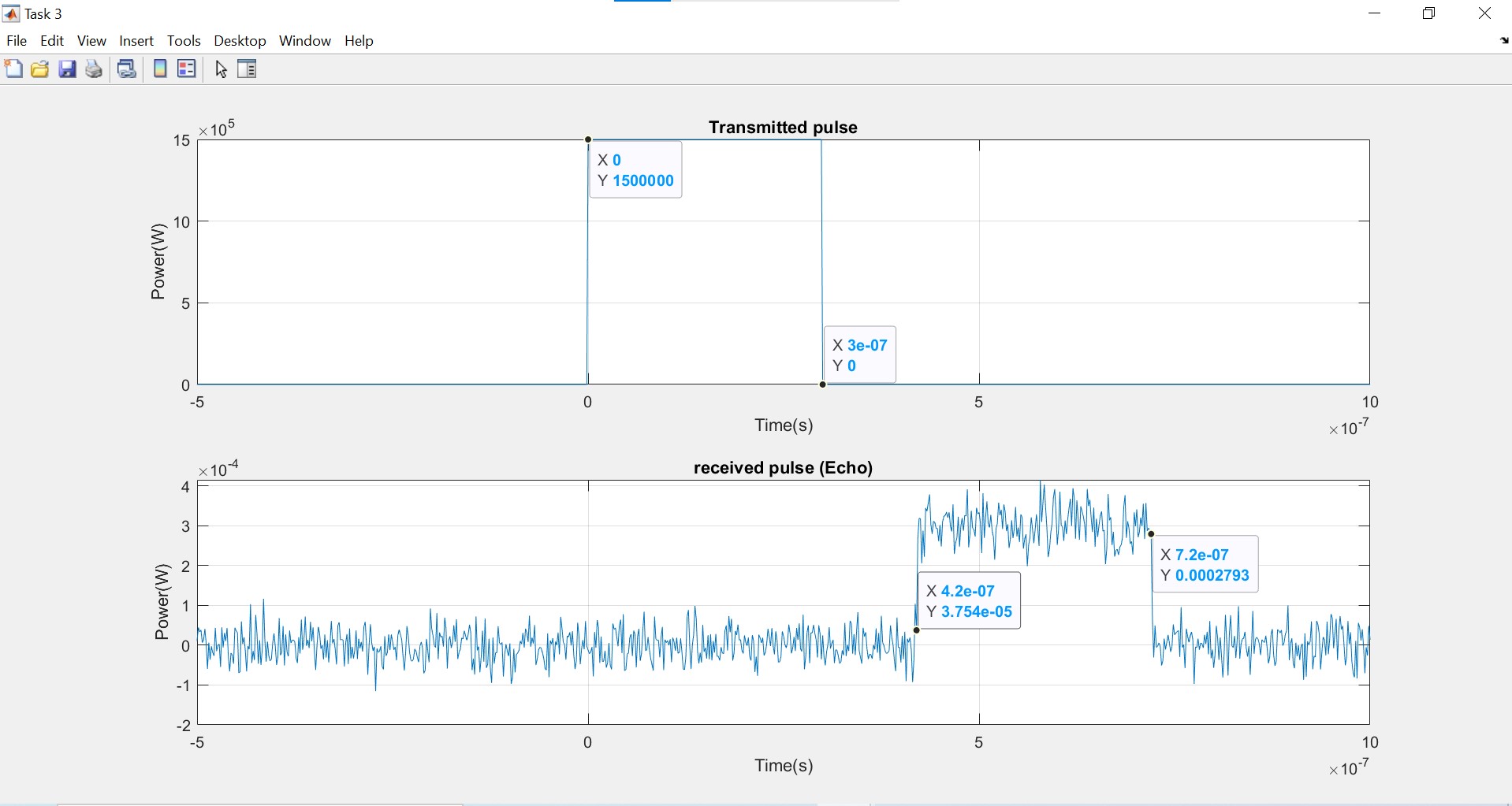
It is the minimum range, if Pulse returns before the end of transmission, it will not be detected.

is the pulse width

****

**Task 3:**

The figures below show the transmitted pulse and the received pulse after the noise has been added to the transmitted pulse.

****

**Code Snippet:**

**figure('NumberTitle','Off','Name','Task 3');**

**%transmitted signal**

**subplot(2,1,1)**

**plot(t,transmittedSignal);xlabel('Time(s)'); ylabel('Amplitude(V)');title('baseband (Transmitted pulse)');**

**grid on**

**%recieved signal**

**subplot(2,1,2)**

**receievedSignal = awgn(transmittedSignal,20,'measured');**

**plot(t,receievedSignal);xlabel('Time(s)'); ylabel('Amplitude(V)');title('Received signal');**

**grid on**

**c=3e8;**

**deltaT=4.20522e-7;**

**RT3=3e8\*deltaT/2;%R in task 3**

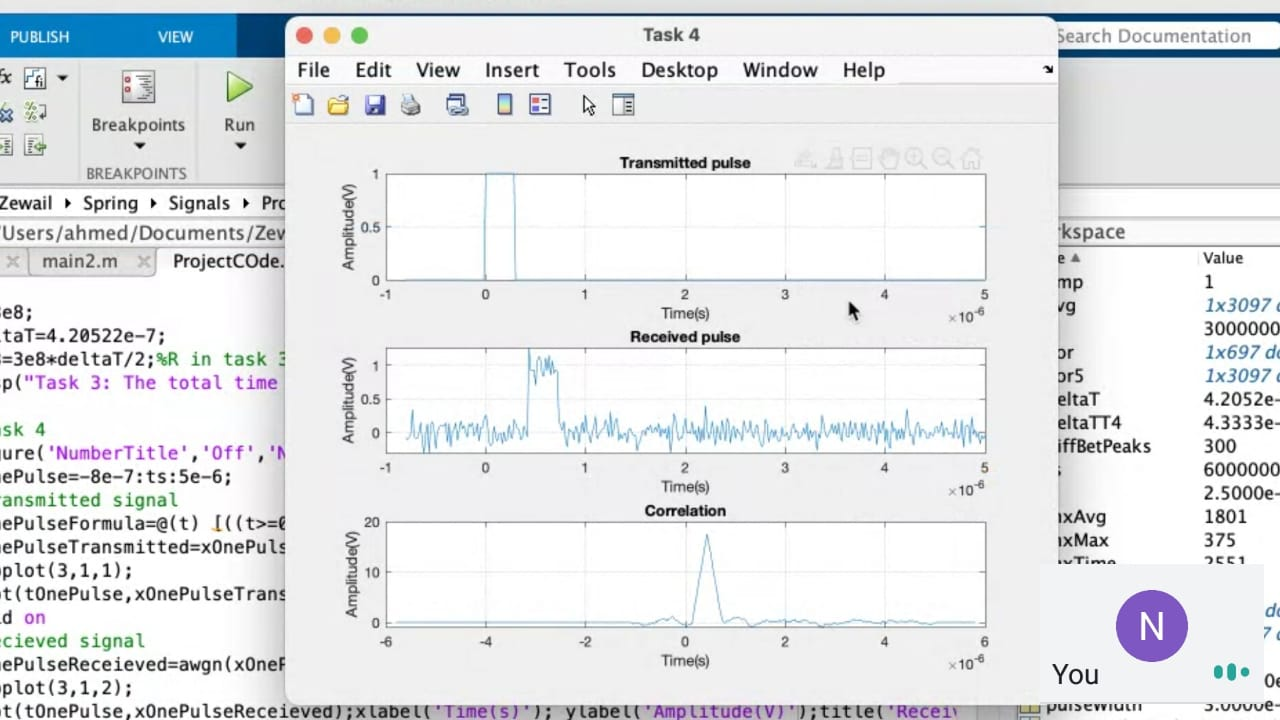
**disp("Task 3: The total time taken by pulse is"+deltaT+" sec, And the distance is: "+RT3);**

**Code Logic:**

Awgn function takes the transmitted signal, the noise ratio which we assumed to be 20 and ‘measured’ which prevents the function from assuming that the input power is 0, it measures the power of the input signal. We then display the pulse resolution in the command window.

**Task 4:**

The first 2 figures below show the transmitted pulse and the received pulse after the noise has been added to the transmitted pulse. Regarding the 3rd figure, a correlation between the transmitted and the received pulse. The 3 figures demonstrate a single pulse. The peak of the received pulse is nearly equal to the peak of the correlation.

****

**Code Snippet:**

**figure('NumberTitle','Off','Name','Task 4');**

**tOnePulse=-8e-7:ts:5e-6;**

**%transmitted signal**

**xOnePulseFormula=@(t) [((t>=0)&(t<3e-7)).\*(amp)];**

**xOnePulseTransmitted=xOnePulseFormula(tOnePulse);**

**subplot(3,1,1);**

**plot(tOnePulse,xOnePulseTransmitted);xlabel('Time(s)'); ylabel('Amplitude(V)');title('Transmitted pulse');**

**grid on**

**%Recieved signal**

**xOnePulseReceieved=awgn(xOnePulseFormula(tOnePulse-deltaT),5,'measured');**

**subplot(3,1,2);**

**plot(tOnePulse,xOnePulseReceieved);xlabel('Time(s)'); ylabel('Amplitude(V)');title('Received pulse');**

**grid on**

**%Correlation**

**[cor,lag]= xcorr(xOnePulseReceieved,xOnePulseTransmitted);**

**subplot(3,1,3);**

**plot(lag\*ts, cor);xlabel('Time(s)'); ylabel('Amplitude(V)');title('Correlation');**

**grid on**

**%Calculations:**

**[valMax, inxMax]=max(cor);**

**deltaTT4 = lag(inxMax)\*ts;**

**RT4=c\*deltaT/2;%R in task 3**

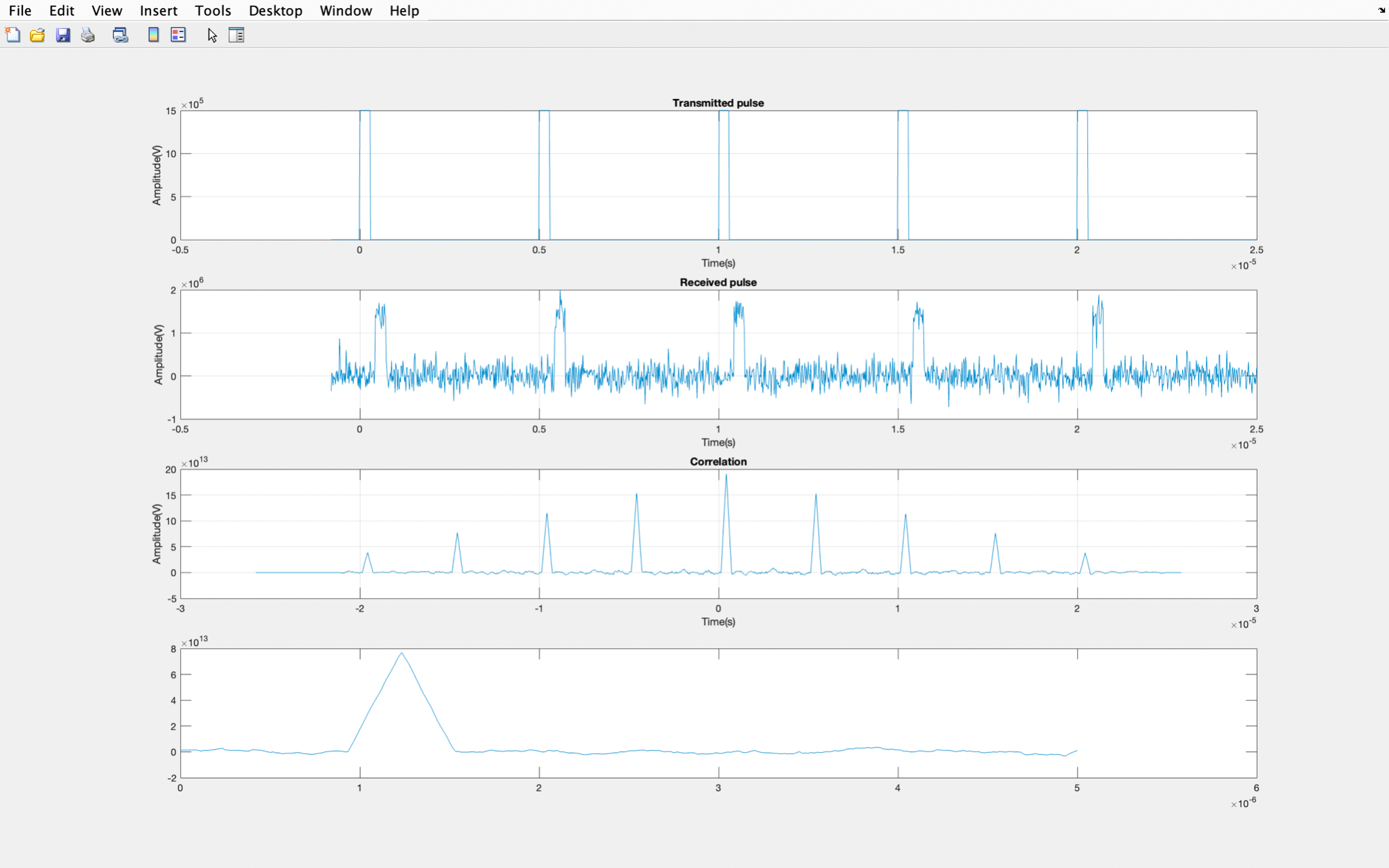
**disp("Task 4: The total time taken by pulse is"+deltaTT4+" sec, And the distance is: "+RT4);**

**Code Logic:**

Cross correlation is used returning the correlation between the transmitted and received signal, the correlation is stored in cor and the lag, which refers to how far those 2 serieses are offset. We multiply the lag by the sampling time to plot it on the x axis against the cor.

**Task 5:**

A Pulse train was created from the single pulses made in task 4, the resulting peaks were used to generate an average Pulse. The average’s x position is similar to that of the peak in the correlation, the averaging slightly affected the position; however it mainly increased the amplitude, and cancels most of the noise; so the signal is clearer, and the time the peak exists at similar position to the the corelation’s peak,and is easier to detect.

****

**Code Snippet:**

**%Task 5**

**figure('NumberTitle','Off','Name','Task 5');**

**tMul=-8e-7:ts:2.5e-5;%time vector that will hold five pulses in task 5**

**xMulPulseFormula=@(t) [((t>=0)&(t<3e-7)).\*(1.5e6) + ((t>=5e-6)&(t<5.3e-6)).\*(1.5e6) + ((t>=1e-5)&(t<1.03e-5)).\*(1.5e6) + ((t>=1.5e-5)&(t<1.53e-5)).\*(1.5e6)+ ((t>=2e-5)&(t<2.03e-5)).\*(1.5e6) ];**

**xMulTransmitted=xMulPulseFormula(tMul);**

**xMulReceived=awgn(xMulPulseFormula(tMul-deltaT),5,'measured');**

**subplot(4,1,1);**

**plot(tMul,xMulTransmitted);xlabel('Time(s)'); ylabel('Amplitude(V)');title('Transmitted pulse');**

**grid on**

**%received**

**subplot(4,1,2);**

**plot(tMul,xMulReceived);xlabel('Time(s)'); ylabel('Amplitude(V)');title('Received pulse');**

**grid on**

**%correlation**

**[cor5,lag5]= xcorr(xMulReceived,xMulTransmitted);**

**subplot(4,1,3);**

**plot(lag5\*ts, cor5);xlabel('Time(s)'); ylabel('Amplitude(V)');title('Correlation');**

**grid on**

**%average**

**repetitionFrequancy=0.2e6;**

**forinterval = fs/repetitionFrequancy;**

**interval1 = cor5(1 :forinterval+1);**

**interval2 = cor5(forinterval+1 :2\*forinterval+1);**

**interval3 = cor5(2\*forinterval+1 :3\*forinterval+1);**

**interval4 = cor5(3\*forinterval+1 :4\*forinterval+1);**

**interval5 = cor5(4\*forinterval+1 :5\*forinterval+1);**

**time = 0: 1/fs : 1/repetitionFrequancy;**

**average = (interval1 +interval2 +interval3 +interval4 +interval5)./5;**

**subplot(4 ,1 ,4 ); plot(time,average);**

**[valueMax,indexMax]=max(cor5);**

**timeVector=lag5\*ts;**

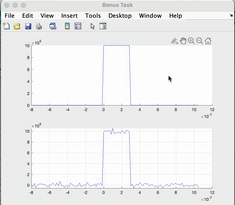
**diffTime=timeVector(indexMax);**

**RT5=c\*diffTime/2;**

**disp("Task 3: The total time taken by pulse is"+diffTime+" sec, And the distance is: "+RT5);**

**Bonus:**

This is the pulse transmitted and the pulse received in an animated form

****

**Code Snippet:**

**subplot(2,1,1);**

**L1=animatedline;**

**subplot(2,1,2);**

**L2=animatedline;**

**for k=1:length(t)**

**addpoints(L1,t(k),transmittedSignal(k));**

**drawnow**

**addpoints(L2,t(k),receievedSignal(k));**

**drawnow**

**end**

**Conclusion:**

**As we can see from the previous results the average is nearly similar to the correlation and the peaks are similar to the received pulses which implies that the invention of the radar is crucial, for both economical and military sectors. Radars could be used to detect long ranged military bombers carrying large payloads before they start attacking.**

**References:**

**Reintjes, J. F., & Coate, G. T. (1952). *Principles of radar*. New York: McGraw-Hill.**

**Basic Pulsed and Continuous Wave (CW) Radar Operations. (2016). *Radar Systems Analysis and Design Using MATLAB*, 49–112. https://doi.org/10.1201/b14904-6**

**Skolnik, M. I. (2020, November 18). *Radar*. *Encyclopedia Britannica*. https://www.britannica.com/technology/radar**