

**BURSA TEKNİK ÜNİVERSİTESİ**  
**MÜHENDİSLİK VE DOĞA BİLİMLERİ FAKÜLTESİ**



**ADALM-PLUTO İLE YAZILIM TABANLI 2.4 GHZ FMCW-RADAR**

**LİSANS BİTİRME ÇALIŞMASI**

**Tunahan ÇIKCAN**

**Elektrik Elektronik Mühendisliği Bölümü**

**TEMMUZ 2025**

**BURSA TEKNİK ÜNİVERSİTESİ**  
**MÜHENDİSLİK VE DOĞA BİLİMLERİ FAKÜLTESİ**



**ADALM-PLUTO İLE YAZILIM TABANLI 2.4 GHZ FMCW-RADAR**

**LİSANS BİTİRME ÇALIŞMASI**

**Tunahan ÇIKCAN**  
**(19332629025)**

**Elektrik Elektronik Mühendisliği Bölümü**

**Danışman: Dr. Öğr. Üyesi Salih Mehmed BOSTAN**

**TEMMUZ 2025**

BTÜ, Mühendislik ve Doğa Bilimleri Fakültesi Elektrik Elektronik Mühendisliği Bölümü'nün 19332629025 numaralı öğrencisi Tunahan ÇIKCAN, ilgili yönetmeliklerin belirlediği gerekli tüm şartları yerine getirdikten sonra hazırladığı “ADALM-PLUTO İLE YAZILIM TABANLI 2.4 GHZ FMCW-RADAR” başlıklı bitirme çalışmasını aşağıda imzaları olan jüri önünde başarı ile sunmuştur.

**Danışmanı :** **Dr. Öğr. Üyesi Salih Mehmed Bostan** .....  
Bursa Teknik Üniversitesi

**Jüri Üyeleri :** **Prof. Dr. Cemal HANILÇI** .....  
Bursa Teknik Üniversitesi

**Arş. Gör. Tahir BEKİRYAZICI** .....  
Bursa Teknik Üniversitesi

**Savunma Tarihi :** 07 Temmuz 2025

**EEM Bölüm Başkanı : Prof. Dr. Cemal HANILÇI** .....  
Bursa Teknik Üniversitesi ...../...../.....

## **İNTİHAL BEYANI**

Bu bitirme alışmasında grsel, işitsel ve yazılı biçimde sunulan tüm bilgi ve sonuçların akademik ve etik kurallara uyularak tarafımdan elde edildiğini, bitirme alışması içinde yer alan ancak bu alışmaya özgü olmayan tüm sonuç ve bilgileri bitirme alışmasında kaynak göstererek belgelediğimi, aksinin ortaya ıkması durumunda her türlü yasal sonucu kabul ettiğimi beyan ederim.

Öğrencinin Adı Soyadı:

İmzası :

## ÖNSÖZ

Bitirme çalışmamı yaparken bana en çok desteği bulunan Sayın Danışmanım Dr. Öğr. Üyesi Salih Mehmed BOSTAN'a teşekkürü bir borç bilirim. Gerek konu hakkındaki bilgi birikimleri ile yol göstermesi olsun, gerek okulumuzun laboratuvar imkanlarından faydalanma konusundaki yardımları olsun bu çalışmaya çok desteği dokunmuştur. Aynı zamanda ailem ve dostlarımın da bana moral ve motivasyon destekleri bu çalışmayı ilerletebilmemde büyük yardımları dokunmuştur.

Temmuz 2025

Tunahan ÇIKCAN

## İÇİNDEKİLER

### Sayfa

ÖNSÖZ .....	v
İÇİNDEKİLER .....	vi
KISALTMALAR .....	vii
SEMBOLLER .....	viii
ÇİZELGE LİSTESİ .....	ix
ŞEKİL LİSTESİ .....	x
ÖZET .....	xi
SUMMARY .....	xii
1. GİRİŞ .....	13
2. LİTERATÜR TARAMASI .....	14
2.1 Radar Sistemlerinin Çalışma Prensipleri .....	14
2.2 FMCW Radar Sistemleri .....	15
3. MATERYAL VE YÖNTEM .....	20
3.1 Sistemin Genel Yapısı .....	20
3.2 Donanım Detayları .....	21
3.2.1 Adalm-Pluto SDR .....	21
3.2.2 RF güç bölücü .....	23
3.2.3 RF güç yükseltici (PA) .....	24
3.2.4 Anten .....	24
3.3 Tasarım Parametreleri .....	25
3.4 Yazılım Detayları .....	26
4. SONUÇLAR .....	28
KAYNAKLAR .....	30
EKLER .....	31
ÖZGEÇMİŞ .....	35

## **KISALTMALAR**

<b>2D</b>	: Two-Dimensional
<b>ACC</b>	: Adaptive Cruise Control
<b>CST</b>	: Computer Simulation Technology
<b>CW</b>	: Continuous Wave
<b>EHF</b>	: Extremely High Frequency
<b>FM</b>	: Frequency Modulation
<b>FMCW</b>	: Frequency-Modulated Continuous Wave
<b>FFT</b>	: Fast Fourier Transform
<b>GHz</b>	: Gigahertz
<b>LFM</b>	: Linear Frequency Modulation
<b>LNA</b>	: Low Noise Amplifier
<b>MHz</b>	: Megahertz
<b>PA</b>	: Power Amplifier
<b>RCS</b>	: Radar Cross Section
<b>RF</b>	: Radio Frequency
<b>RX</b>	: Receiver
<b>SDR</b>	: Software Defined Radio
<b>SHF</b>	: Super High Frequency
<b>SNR</b>	: Signal to Noise Ratio
<b>TX</b>	: Transmitter
<b>USB</b>	: Universal Serial Bus
<b>VCO</b>	: Voltage-Controlled Oscillator

## SEMBOLLER

<b>A</b>	: Sinyal genliđi
<b>B</b>	: Bant geniřliđi
<b>c</b>	: Iřık hızı
<b>f<sub>0</sub></b>	: Tařıyıcı frekans
<b>f<sub>b</sub></b>	: Beat frekans
<b>f<sub>D</sub></b>	: Doppler frekans
<b>f<sub>s</sub></b>	: Örnekleme frekans
<b>G</b>	: Anten kazancı
<b>λ</b>	: Dalga boyu
<b>N</b>	: FFT boyutu (mesafe için)
<b>M</b>	: Chirp sayısı (doppler için FFT boyutu)
<b>P<sub>t</sub></b>	: Verici çıkıř gücü
<b>P<sub>r</sub></b>	: Alıcıda alınan güç
<b>R</b>	: Mesafe
<b>ΔR</b>	: Mesafe çözünürlüğü
<b>Δv</b>	: Hız çözünürlüğü
<b>σ<sub>t</sub></b>	: Radar kesit alanı
<b>t</b>	: Zaman
<b>T<sub>s</sub></b>	: Sweep süresi / Chirp süresi
<b>t<sub>d</sub></b>	: Zaman gecikmesi
<b>φ<sub>0</sub></b>	: Bařlangıç fazı
<b>s(t)</b>	: Gönderilen sinyal
<b>s<sub>r</sub>(t)</b>	: Alınan / yansıyan sinyal
<b>s<sub>IF</sub>(t)</b>	: Ara frekans sinyali
<b>s<sub>IF,filtered</sub>(t)</b>	: Filtrelenmiř ara frekans sinyali
<b>L<sub>PD</sub></b>	: RF güç bölücü kaybı
<b>G<sub>PA</sub></b>	: Güç yükseltici kazancı



## ÇİZELGE LİSTESİ

	<b><u>Sayfa</u></b>
<b>Çizelge 3.1 : AD9361 özellik çizelgesi (Analog Devices, 2020).</b> .....	<b>22</b>
<b>Çizelge 3.2 : RF güç bölücü özellik çizelgesi</b> .....	<b>23</b>

## ŞEKİL LİSTESİ

### Sayfa

Şekil 2.1 : (a) sürekli dalga, (b) darbe dalga.....	14
Şekil 2.2 : FMCW-Radar sinyalinin frekans-zaman eksenindeki grafiği.....	15
Şekil 2.3 : FMCW radar blok diyagramı. ....	16
Şekil 2.4 : FMCW radar 2D-FFT ile hız-mesafe bilgisi çıkarımı. ....	18
Şekil 2.5 : NxM range-doppler matrisi. ....	20
Şekil 3.1 : Adalm-Pluto ile FMCW radar blok diyagramı. ....	21
Şekil 3.2 : AD9361 fonksiyonel blok diyagramı. ....	22
Şekil 3.3 : RF güç bölücü. ....	23
Şekil 3.4 : SPF5189Z LNA frekans-kazanç grafiği (RF Micro Devices, 2013).....	24
Şekil 3.5 : (a) CST’de tasarlanan antenin 3B görüntüsü, (b) $S_{11}$ parametresi, (c) kazancı. ....	25
Şekil 3.6 : FMCW radar algoritma şeması. ....	27
Şekil 4.1 : FMCW radar deneysel ölçüm sonucu. ....	28
Şekil 4.2 : Tasarlanan FMCW radar sisteminin fiziksel görünümü. ....	29

## ADALM-PLUTO İLE YAZILIM TABANLI 2.4 GHZ FMCW-RADAR

### ÖZET

Bu çalışmada 2.4 GHz frekans bandında çalışan bir FMCW-Radar, Adalm-Pluto SDR kullanılarak yazılım tabanlı olarak yapılmıştır. Bu tezde de yapılan çalışmanın detaylı araştırması, ve tasarım aşamalarına yer verilmiştir. FMCW-Radar ile hareketli cisimlerin hem konum hem hız bilgilerini elektromanyetik dalga kullanılarak ölçülmesi amaçlanmıştır. Elektromanyetik dalganın hareketli cisme çarpması sonucu oluşan doppler etkisi ve elektromanyetik dalganın alıcı antene geliş süresi ölçülerek hız ve mesafe bilgisinin ölçülmesi hedeflenmiştir. Kullanılan donanımın istenilen performans ve yetkte olmamasından dolayı bazı kısıtlamalarla karşılaşmıştır (USB 2.0 veri hızı, AD9361 bant genişliği vb.). Bu kısıtlamalar sonucunda istenilen hassasiyet elde edilemese de yaklaşık 3 metreye kadar olan cisimlerin hareketleri tespit edilebilmiştir. Daha önceden CST Studio Suite ile analizi yapıp üretilen mikroşerit anten test aşamasının başlarında kullanılmıştır. Ancak performans sorunlarından kaynaklı olarak sonradan 10 dBi kazançlı vivaldi antene geçilmiştir.

Radar, Python kullanılarak gerçek zamanlı olarak testleri yapılmıştır. Yavaş olmasına rağmen (C diline göre) kolay, pratik ve en önemlisi Adalm-Pluto ile uyumlu olmasından dolayı Python tercih edilmiştir. Yazılımda hazır Python kütüphaneleri kullanılarak chirp üretilmiş ve Adalm-Pluto ile yayılmıştır. Alınan sinyal 2D FFT yöntemi kullanılarak mesafe ve hız bilgisinin ortaya çıkarılması sağlanmıştır. Python grafik kütüphanesi kullanarak da gerçek-zamanlı görselleştirme yapılmıştır.

Adalm-Pluto donanımsal kısıtlamaları yüzünden mesafe ölçümü istenilen sonuçları vermese de insan hareketlerinde oluşan hız bilgisi 1.95 m/s hassasiyetle ölçülebilmektedir. Bilgisayar donanımı yeterli olmaması ve Python'da gerçek-zamanlı sinyal işleme performansının düşük olmasından dolayı hız çözünürlüğü düşük tutulmuştur. İnsan radar kesit alanı düşük olmasına rağmen ölçün yapılabilmektedir.

Çalışmada mümkün olduğunca maliyetler göz edilerek bir FMCW-Radarın lisans düzeyinde uygulanabilirliği de amaçlanmıştır. Adalm-Pluto kullanılarak FMCW-Radarın yapımı ülkemizde çok benzeri olmamasından kaynaklı olarak ekstra yüksek maliyetli RF elemanları (RF Mikser, VCO vb.) kullanılmadan uygulanabilirliği sayesinde lisans seviyesi eğitime katkı sağlaması da amaçlanmıştır. Radar elemanlarının özel olarak üretimi ile oluşturulacak bir sistemin daha performanslı olması beklenmesine rağmen bu çalışmada hassasiyetten çok uygulanabilirliği kolay olan bir tasarım yöntemi tercih edilmiştir. Deneysel ve teorik bilgi olarak büyük oranda başarı elde edilmiştir.

**Anahtar kelimeler:** FMCW, radar, Adalm-Pluto, elektromanyetik dalga, Python.

## SOFTWARE-BASED 2.4 GHZ FMCW-RADAR USING ADALM-PLUTO

### SUMMARY

In this study, a software-based FMCW radar operating in the 2.4 GHz frequency band was implemented using the Adalm-Pluto SDR. This thesis includes a detailed investigation of the work and the design stages. The aim of the FMCW radar is to measure both the position and velocity of moving objects using electromagnetic waves. By measuring the Doppler effect caused by the electromagnetic wave reflecting off a moving object and the time it takes for the wave to reach the receiving antenna, the goal was to determine the speed and distance information. Due to the limited performance and capabilities of the hardware used (USB 2.0 data rate, AD9361 bandwidth, etc.), some constraints were encountered. Although the desired sensitivity could not be achieved because of these limitations, the movements of objects up to approximately 3 meters could be detected. A microstrip antenna previously designed and analyzed with CST Studio Suite was used in the early stages of testing. However, due to performance issues, it was later replaced with a 10 dBi gain Vivaldi antenna.

The radar was tested in real-time using Python. Although slower compared to the C language, Python was preferred because it is easy, practical, and most importantly, compatible with Adalm-Pluto. Chirp signals were generated using ready-made Python libraries and transmitted through Adalm-Pluto. The received signal was processed using a 2D FFT method to extract distance and velocity information. Real-time visualization was done with Python's graphics libraries.

Due to Adalm-Pluto's hardware limitations, the distance measurement did not provide the desired results, but velocity information related to human movement was measured with an accuracy of 1.95 m/s. Because of insufficient computer hardware and the low real-time signal processing performance in Python, velocity resolution was kept low. Despite the low radar cross section of humans, measurements were still successfully made.

The study also aimed to demonstrate the feasibility of implementing an FMCW radar at the undergraduate level with cost considerations. By using Adalm-Pluto, this radar design provides an approach that does not require extra high-cost RF components (such as RF mixers, VCOs, etc.), which are rare in similar applications in Turkey, thus contributing to undergraduate education. Although a system constructed with specially produced radar components is expected to be more performant, this study preferred a design method that is easier to implement rather than focusing on precision. Both experimental and theoretical results showed considerable success.

**Keywords:** FMCW, radar, Adalm-Pluto, electromagnetic wave, Python.

## 1. GİRİŞ

Günümüzde radar sistemleri teknolojinin gelişmesi ile birlikte neredeyse çoğu sektörde hayatımıza girmiş bulunmaktadır. Elektromanyetik dalganın yayılımı ile çalışan radarlar elektromanyetik dalganın ışık hızına yakın hareket etmesinden kaynaklı olarak hızlı birer hız ve mesafe ölçüm aracı olmuştur. Çok farklı türlerde (FMCW-Radar, CW-Radar, Darbe Radarı vb.) ve farklı amaçlar için kullanılmaktadırlar. Kullandıkları alan ve türe göre de radarın toplam üretim maliyetleri de ona göre farklılık göstermektedir.

FMCW-Radarlar hem hız hem de mesafe bilgisini ölçebilmesinden kaynaklı olarak genellikle otomotiv sektöründe ACC (Adaptive Cruise Control) veya kaza uyarı sistemlerinde sıkça kullanılmaktadır (Ju, Jin & Lee 2014). Bu radarlar genellikle SHF (K-band) veya EHF (E-band) frekans bandında üretilmektedir. Ancak bu frekanslarda kullanılan RF elemanlarının üretim ve tasarım maliyetlerinin yüksek olması bu radar sisteminin de lisans düzeyi için maliyet açısından uygun olmadığını göstermektedir. FMCW-Radarlarda taşıyıcı frekans, chirp bant genişliği ve chirp süresi gibi parametreleri değiştirerek özelleştirilmiş radarlar yapmak da mümkündür. Eğer üretilmesi hedeflenen radar için de 2.4 GHz taşıyıcı frekansının seçilmesinin sebebi de RF elemanlarının maliyeti gö önünde bulundurulmasından kaynaklanmaktadır.

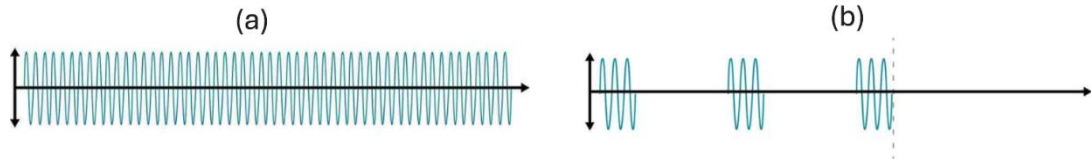
Tezin ilk bölümünde literatür taramasına, radar sistemlerinin çalışma prensiplerine ve FMCW-Radar sisitemlerinin detaylarına yer verilmiştir. İkinci bölümde kullanılan yöntemler ve donanım detayları anlatılmıştır. Üçüncü bölümde sistemin performansı ve elde edilen test sonuçlarına yer verilmiştir. Son bölümde ise genel değerlendirme ve sonuç kısmı yer almaktadır.

## 2. LİTERATÜR TARAMASI

### 2.1 Radar Sistemlerinin Çalışma Prensibi

Radar sistemleri, 2. Dünya Savaşı'ndan bu yana hem savunma hem de endüstriyel alanda kullanımı yaygın olan sistemlerdir. Elektromanyetik dalganın cisimden yansıması ve yansıyan elektromanyetik dalganın alıcı antene ulaşması ile çalışmaktadır. Alıcıya ulaşan dalganın, gönderilen dalga ile arasında oluşan faz farkı radarın asıl çalışma prensibini açıklamaktadır. Hedeften yansıyan elektromanyetik dalganın zaman gecikmesi mesafe bilgisini içermektedir (Charvat, 2014).

Uygulama alanına göre farklı radar çeşitleri üretilmiştir. Daha uzak mesafeler için darbe radarları tercih edilirken yakın mesafelerde CW radarlar tercih edilmektedir.



Şekil 2.1 : (a) sürekli dalga, (b) darbe dalga.

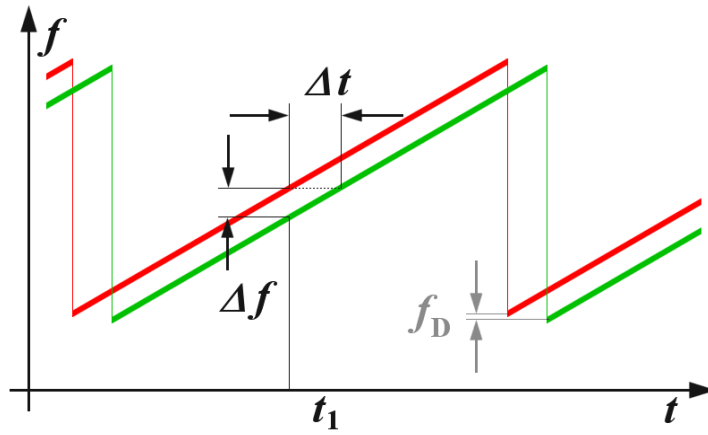
Darbe dalga radarında elektromanyetik dalga kısa sürede darbeler halinde gönderildiği için yüksek güç uygulanarak dalganın daha uzak cisimleri algılaması sağlanabilir. Ancak sürekli dalga radarlarında ise devamlı olarak üretilip gönderilen sinyal yüksek güç gerektiren uygulamalardansa düşük güç yüksek hassasiyet gerektiren uygulamalarda tercih edilmektedir. Endüstriyel alanda sıvı seviyesini ölçmek için kullanılan mmWave radarlar buna bir örnek olarak gösterilebilmektedir. Sürekli dalga radarları da kendi içerisinde CW ve FMCW olmak üzere iki grupta incelenebilir. CW radarlar modülasyon kullanılmadan sabit bir frekansta sürekli olarak elektromanyetik dalga göndererek sadece doppler etkisi sonucu oluşan frekans farkından hız bilgisini

ölçebilmektedir. Öte yandan FMCW radarlar ise aynı anda hem hız hem de mesafe bilgisini ölçebilmektedir.

## 2.2 FMCW Radar Sistemleri

Sürekli dalga radarlarının bir türü olan FMCW radar, frekansın zaman içinde belirli bir bant genişliği miktarınca değiştirilerek modüle edilmesi ile oluşturulmuş radar sistemidir (Jankiraman, 2018).

FMCW radar, gönderilen modüle edilmiş sinyal ile alıcıya ulaşan sinyali karıştırarak oluşan fark sinyali üzerinden işlem yapmaktadır.



Şekil 2.2 : FMCW-Radar sinyalinin frekans-zaman eksenindeki grafiği.

FMCW radar sisteminde kullanılan RF mikser bu iki sinyalin karıştırılmasını sağlamaktadır.

$$x_1(t) = A_1 \cos(2\pi f_1 t) \quad (2.1)$$

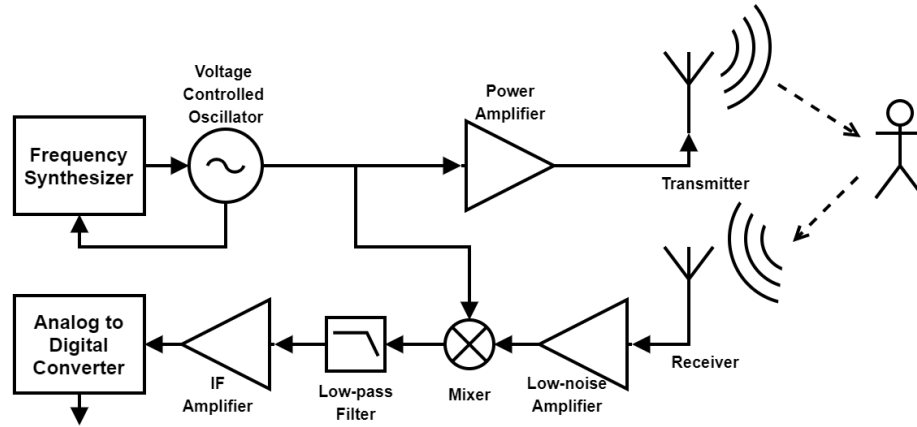
$$x_2(t) = A_2 \cos(2\pi f_2 t) \quad (2.2)$$

$$x_1(t) \times x_2(t) = \frac{A_1 A_2}{2} [\cos(2\pi t(f_1 - f_2)) + \cos(2\pi t(f_1 + f_2))] \quad (2.3)$$

Denklem 2.1 ve Denklem 2.2’de verilen iki sinyal farklı olmak koşulu ile  $f_1$  ve  $f_2$  frekanslarında sürekli sinüzoidal işaret ise bu iki işaretin RF mikser ile çarpımı Denklem 2.3’te gösterilmektedir. Denklem 2.3’te görüldüğü üzere bu iki işaretin

çarpımından oluşan yeni işaretin bileşenleri Denklem 2.1 ve 2.2'deki sinyallerin frekans farkı ve toplamından oluşan frekansa sahiptir. Şekil 2.2'de FMCW radarlarda kullanılan bir modülasyon çeşidi olan LFM (Lineer Frekans Modülasyonu), frekans-zaman ekseninde görülmektedir. Kırımızı gönderilen sinyal, yeşil olan ise alıcı sinyali temsil etmektedir. Burada  $\Delta t$  elektromanyetik dalganın cisme ulaşır geri yansıma süresini,  $\Delta f$  oluşan zaman farkı sonucu frekans-zaman ekseninde kayan dalganın frekan farkı ve  $f_D$  ise hareketli cisme çarpan dalganın doppler etkisi ile oluşturduğu frekans kaymasını temsil etmektedir.

FMCW radarı daha detaylı inceleyebilmek için blok diyagramı üzerinden ayrı ayrı incelemek daha doğru olacaktır.



Şekil 2.3 : FMCW radar blok diyagramı.

Şekil 2.3'teki blok diyagramına bakarak incelemeye VCO kısmından başlayabiliriz.

FMCW radar sisteminde FM kısmı olan frekans modülasyonu, radarın hem mesafe hem de hız bilgisini elde edebilmesi için büyük önem taşımaktadır. Bu modülasyonda LFM tercih edilmektedir. LFM, frekansı zaman içerisinde lineer olarak belirli bir frekansa kadar arttırarak (azaltarak veya önce arttırıp sonra azaltarak) oluşturulan modülasyon tekniğidir.

$$s(t) = A \cos \left( 2\pi \left( f_0 t + \frac{B}{2T} t^2 \right) + \phi_0 \right), \quad 0 \leq t \leq T \quad (2.4)$$

Denklem 2.4'te lineer frekans modülasyonunun nasıl yapıldığını matematiksel olarak açıklamaktadır. Burada  $f_0$  taşıyıcı frekansı, B frekans modülasyonun band genişliğini, T frekans modülasyonunun süresini ve  $\phi_0$  ise başlangıç fazını temsil etmektedir. VCO girişine uygulanacak değişken voltaj ile bu modülasyon elde edilebilmektedir. Daha



düşük frekans bandında olan bu modüleli sinyal taşıyıcı frekansa bindirilerek radar anteninden ortama yayılacaktır.

VCO'da modüleli sinyal üretildikten ve taşıyıcıya bindirildikten sonra RF güç bölücüye gelerek hem verici antenine hem de gelen sinyalle karşılaştırılmak üzere RF mikserine iletilmektedir. Burada RF güç bölücü gelen sinyali 3 dB kayıpla (idealde) çıkış portlarına iletmektedir. Daha sonra modüleli sinyal PA ile kuvvetlendirilerek dış ortama anten vasıtası ile yayılmaktadır.

Hedekten yansıyan elektromanyetik dalga alıcı antene ulaştıktan sonra LNA ile SNR'ı (sinyal-gürültü oranı) iyileştirilerek RF mikserin RF girişine iletilmektedir.

$$s_r(t) = \alpha A \cos \left[ 2\pi \left( (f_0 + f_D)(t - t_d) + \frac{B}{2T}(t - t_d)^2 \right) + \phi_0 \right] \quad (2.5)$$

RF girişine gelen hareketli hedekten yansıyan sinyal  $s_r(t)$ , Denklem 2.5'te verilmektedir.  $f_D$  doppler frekansı,  $t_d$  zaman kayması (time delay) olarak gösterilmektedir. Burada gönderilen modüleli sinyalin doppler ve mesafe etkisi ile değişimi matematiksel olarak görülebilmektedir. Tekrar bakacak olursak Şekil 2.1 için Denklem 2.4 kırmızı, Denklem 2.5 ise yeşil çizgiyi temsil etmektedir.

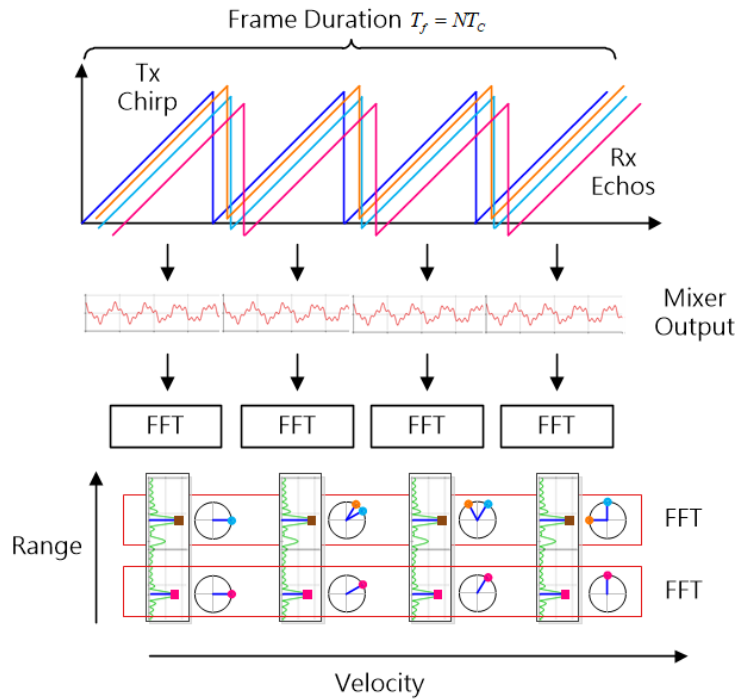
RF mikser çıkışında ise Denklem 2.4 ve Denklem 2.5'in çarpımının sonucunu vermektedir.

$$\begin{aligned} s_{IF}(t) = \alpha \frac{A^2}{2} \cos \left[ 2\pi \left( f_0 t_D - f_D(t - t_d) + \frac{B}{2T}t^2 - \frac{B}{2T}(t - t_d)^2 \right) \right] \\ + \alpha \frac{A^2}{2} \cos \left[ 2\pi \left( (f_0 t + (f_0 + f_D)(t + t_D) + \frac{B}{2T}t^2 \right. \right. \\ \left. \left. + \frac{B}{2T}(t - t_D) \right) \right] \end{aligned} \quad (2.6)$$

Denklem 2.6'da  $s_{IF}(t)$  eşitliğinde ara frekans çıktısı görülmektedir (Jankiraman, 2018). Denklem 2.4 ve 2.5'in frekans bileşenlerinin farkı ve toplamı dışarıya verilmektedir. Burada frekans bileşenlerinin toplamı olan Denklem 2.6'nın ikinci kısmı istenmeyen kısım olarak ortaya çıkmaktadır. Bu yüzden RF mikser çıkışı filtreden geçirilerek sadece düşük frekanslı bileşenlerin geçişine izin verilmektedir. Filtrelenmiş  $s_{IF}(t)$  sinyali içinde hem hız hem de mesafe bilgisi bulunmaktadır. Bu

bilgileri ortaya çıkarabilmek için filtreleme aşamasından sonra sinyal işleme algoritmaları kullanılarak elde edilebilmektedir.

$s_{IF,filtered}(t)$  sinyalinin FFT'si alındığında görülen belirli yüksek genlikli frekans bileşenleri tespit edilen nesnelerin mesafesini frekans olarak göstermektedir (yanlış alarm olarak gürültüyü tespit etmediği varsayılırsa). Eğer M adet modüleli sinyalin (chirp) 2D FFT'si alınırsa da mesafe bilgisinin zamana göre değişimi yani nesnenin hızı (doppler frekansı) elde edilebilmektedir.



**Şekil 2.4 :** FMCW radar 2D-FFT ile hız-mesafe bilgisi çıkarımı.

Şekil 2.4 yukarıda anlatılan algoritmayı görsel olarak desteklemektedir. Hız bilgisini mesafe bilgisindeki faz değişimi olarak da tanımlamak yerinde olmaktadır. Burada radar sisteminin son ve en önemli aşamalarından biri olan dijital sinyal işleme adımlarına geçilmektedir.

Bu işlemler sonucundaki veriler genellikle frekans cinsinden elde edilmektedir. Mesafe ve hızın m ve m/s cinsinden yazılabilmesi için uygun denklemlerde yerine koyulması gerekmektedir.

FMCW radar için mesafe bilgisinin hesaplanması (Jankiraman, 2018):

$$R = \frac{f_b T_s c}{2\Delta f} \quad (2.7)$$

Burada R mesafe,  $f_b$  beat frekansı,  $T_s$  sweep süresi,  $c$  ışık hızı ve  $\Delta f$  sweep bant genişliğini vermektedir.

Mesafe çözünürlüğü için ise:

$$\Delta R = \frac{c}{2\Delta f} \quad (2.8)$$

Denklem 2.8 ile elde edilmektedir. Burada hesaplanacak değer radarın iki cismi ayrı ayrı ayırt edebilmesi için aralarında olması gereken minimum mesafeyi vermektedir. Hız bilgisini elde edebilmek için taşıyıcı frekansa uygulanan doppler etkisine bakmamız yeterli olacaktır.

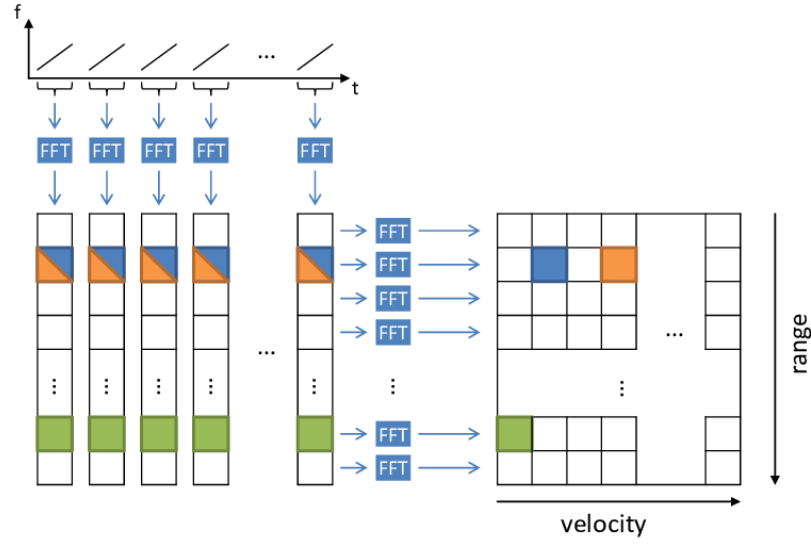
$$v = \frac{f_D c}{2f_0} \quad (2.9)$$

Burada  $f_0$  taşıyıcı frekans ve  $v$  hız bilgisini vermektedir. Hız çözünürlüğü için ise (Suleymanov, 2016):

$$\Delta v = \frac{c}{2f_0} \frac{1}{MT_s} \quad (2.10)$$

Denklem 2.10'da M için doppler FFT boyutu demek doğru olacaktır.

FMCW radar sinyal işleme kısmının sonucunda range-doppler haritası çıkarılması amaçlanmaktadır. Bu haritanın çıkartılabilmesi için N FFT boyutu ile örneklenmiş M adet chirp incelenmektedir. NxM matrisi her M adet chirp elde edildiğinde yeni veriler ile matris oluşturulmaktadır. Bu şekilde gerçek-zamanlı olarak hız ve mesafe bilgisi elde edilebilmektedir.



Şekil 2.5 :  $N \times M$  range-doppler matrisi.

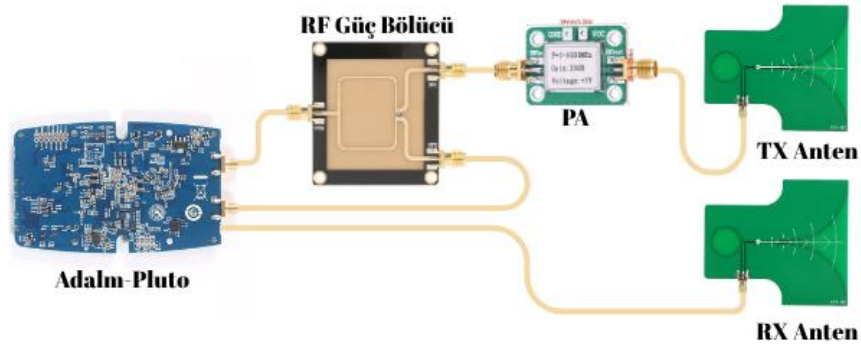
Şekil 2.5'te verilen matris range-doppler haritasının yaklaşık görüntüsünü verebilmektedir.

### 3. MATERYAL VE YÖNTEM

#### 3.1 Sistemin Genel Yapısı

FMCW radar sistemleri genel yapısı itibarı ile donanım ve yazılım olarak ikiye ayrılabilir. Çalışmanın maliyet açısından daha uygun ve donanım tasarım kısımlarındaki uzun üretim süreçleri ile zaman kaybedilmemesi için hazır donanım olan Analog Devices firmasının Adalm-Pluto SDR devresi kullanılmıştır. Hazır donanım harici Adalm-Pluto içerisinde bulunmamasından kaynaklı olarak anten, RF

güç bölücü ve PA (Adalm-Pluto’da bulunmasına rağmen RF güç bölücü çıkışındaki güç kaybını gidermek için) kullanılmaktadır.



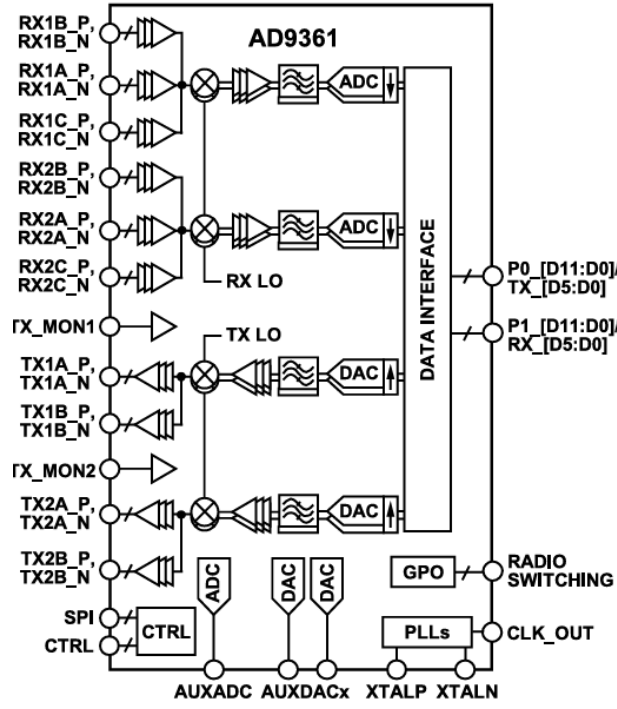
Şekil 3.1 : Adalm-Pluto ile FMCW radar blok diyagramı.

Şekil 3.1’de Adalm-Pluto içerisinde yazılımsal olarak mikser işleminin yapılabilmesi için 1TX-1RX olarak tek kanallı üretilen Adalm-Pluto’nun içerisinde harici bir bağlantının yapılması gerekmiştir. IPEX-SMA kablo kullanılarak Adalm-Pluto üzerine de 2 adet SMA portu için delik açıldıktan sonra bağlantılar yapılmıştır. Bağlantılar yapılmadan önce Adalm-Pluto AD9363 çipi yerine AD9361 çipi aktive edilmesi gerekmektedir. Bu sayede 2 kanallı hale getirilebilmektedir. Adalm-Pluto TX çıkışı RF güç bölücüde ayrıldıktan sonra biri PA’ya biri de mikser için RX0 portuna bağlanmıştır.

## 3.2 Donanım Detayları

### 3.2.1 Adalm-Pluto SDR

Adalm-Pluto, yazılım tabanlı olarak RF projeleri gerçekleştirmek için Analog Devices tarafından üretilmiş bir cihazdır. RF projelerinde üretilmesi gereken donanıma ihtiyaç kalmadan bu cihaz kullanılarak kolay ve pratik bir şekilde uygulamalar yapılabilmektedir.



Şekil 3.2 : AD9361 fonksiyonel blok diyagramı.

Çizelge 3.1 : AD9361 özellik çizelgesi (Analog Devices, 2020).

Parametre	Min	Typ	Maks	Birim
Merkez Frekans	70		6000	MHz
RX Kazanç	0		72.0	dB
Maksimum Çıkış Gücü			7.5	dBm
Noise Figure		3		dB
Kanal Bant Genişliği	0.2		56	MHz

AD9361'in parametre değerleri radarın parametre değerlerinin belirlenmesindeki en büyük sınırlandırıcı unsur olmaktadır. Özellikle sweep bant genişliğini sınırlayan yani mesafe çözünürlüğünü etkileyen kanal bant genişliğidir.

$$B < \frac{f_s}{2} \quad (3.1)$$

Nyquist şartına göre örnekleme frekansı  $f_s$ , bant genişliğinin iki katından fazla olmalıdır. Burada örnekleme frekansı  $f_s$ , kanal bant genişliğinin maksimum değeri seçilerek tasarım belirlenmiştir. Ancak AD9361 iki kanallı hale geldiğinde maksimum kanal bant genişliği de değişmektedir. Yeni maksimum kanal bant genişliği 30.72 MHz olarak değişmektedir.

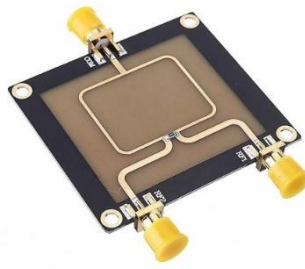
Adalm-Pluto çıkış gücü de diğer bir önemli parametredir. Antenden yayılan gücün direkt olarak mesafeye etkisi bulunmaktadır. Radar denklemi (Ulaby, 2004):

$$P_r = \frac{P_t G^2 \lambda^2 \sigma_t}{(4\pi)^3 R^4} \quad (3.2)$$

Denklem 3.2’de radar denkleminin genel formülü verilmiştir. Burada  $P_r$  alıcı gücü,  $P_t$  iletim gücü,  $G$  anten kazancı,  $\lambda$  dalga boyu,  $\sigma_t$  RCS (Radar Kesit Alanı) ve  $R$  anten ile cisim arasındaki mesafeyi temsil etmektedir. Hedefin elektromanyetik dalga yansıtıcılığı ve kesit alanı büyüklüğü  $\sigma_t$  RCS’yi etkilemektedir. Bu denklemde radar tasarımında hedef tespiti için belirlenebilecek parametreler  $P_t$ ,  $G$  ve  $\lambda$ ’dır. Radarın taşıyıcı frekansı, antenin kazancı ve Adalm-Pluto TX çıkış gücü ayarlanarak daha uzak mesafedeki hedefler tespit edilebilmektedir.

### 3.2.2 RF güç bölücü

Adalm-Pluto’da üretilen modüleli sinyalin hem antenden ortama yayılması hem de frekans karıştırmak için RX1 portuna gönderilmesi gerekmektedir. RF güç bölücüler idealde 2 çıkışlı ise 3 dB kayıpla çıkışa sinyali iletir. Ancak gerçek hayatta kayıplar da dahil edildiğinde bu değer biraz daha üstü de olması mümkündür. Radarda kullanılan RF güç bölücü devresi 2.4 GHz’de yaklaşık 4.2 dB kayıpla çıkışa sinyali iletmektedir. Bursa Teknik Üniversitesi laboratuvarında yapılan testlerde bu değer gözlemlenmiştir.



Şekil 3.3 : RF güç bölücü.

Çizelge 3.2 : RF güç bölücü özellik çizelgesi.

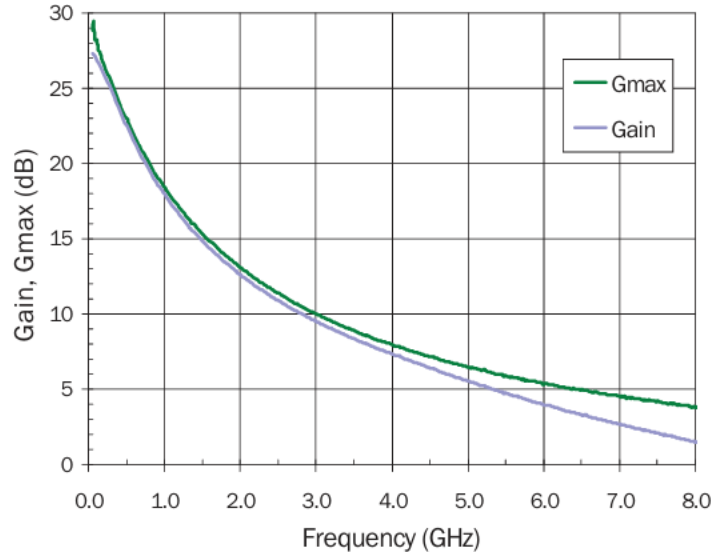
Parametre	Min	Typ	Maks	Birim
Çalışma Frekansı	100		2700	MHz
Maksimum Güç			25	W
VSWR		≤1.4		
Empedans		50		Ω

### 3.2.3 RF güç yükseltici (PA)

RF güç bölücü sonrası yaklaşık 4.2 dB kayıp yaşayan sinyal antenle dış ortama yayılmadan önce bir miktar güçlendirilmesi gerekmektedir. Denklem 3.2’den de görüleceği üzere yayılan gücün yüksek olması (diğer parametreler sabitken) hedef tespitinde maksimum mesafenin de yüksek olmasını sağlamaktadır. Çizelge 3.1’deki maksimum çıkış gücü referans alınarak teorik bir hesaplama yapıldığında:

$$\begin{aligned} P_t(dB) &= P_{tx}(dBm) - L_{PD}(dB) + G_{PA}(dB) = 7.5 - 4.2 + 11 \\ &= 14.3 \text{ dBm} \end{aligned} \quad (3.3)$$

olarak bulunmuştur. Burada  $P_{tx}$  Adalm-Pluto TX çıkış gücü (maksimum),  $L_{PD}$  RF güç bölücü kaybı ve  $G_{PA}$  PA kazancını vermektedir. Değerler logaritmik ölçüye göre hesaplanmıştır. Denklem 3.3 ile anten çıkışındaki maksimum gücü hesaplanmaktadır. Çalışmada yüksek maliyetlerden kaçınmak için PA yerine SPF5189Z LNA tercih edilmiştir.



Şekil 3.4 : SPF5189Z LNA frekans-kazanç grafiği (RF Micro Devices, 2013).

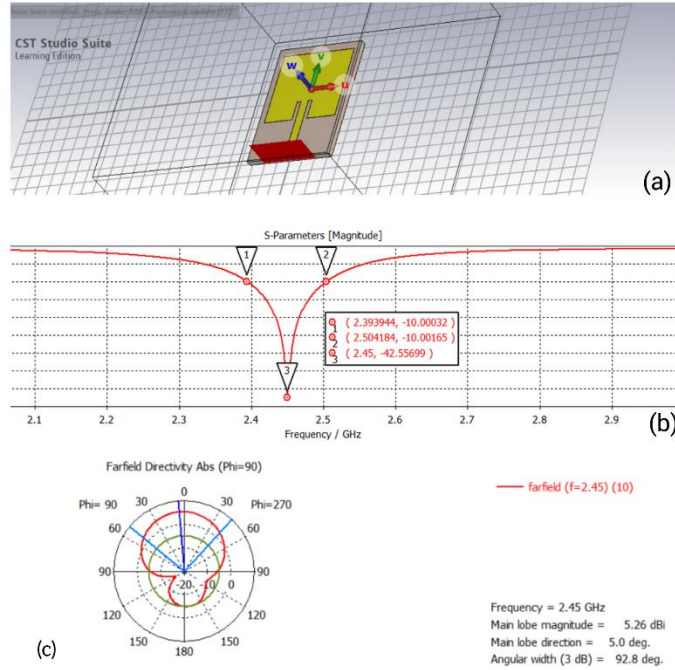
Şekil 3.4’teki grafik incelendiğinde çalışma frekansı olan 2.4 GHz için kazancın yaklaşık 11 dB olduğu görülebilmektedir. Yüksek bir kazanç değeri olmasa da maliyet göz önünde bulundurulduğunda uygun bir tercih olabileceği düşünülmüştür.

### 3.2.4 Anten

Radarın elektromanyetik dalgayı ortama yayması ve cisimden yansıyan dalgayı yakalayabilmesi için anten en önemli bileşenlerden biridir. Radarlarda çeşitli



uygulamalar için farklı anten türleri tercih edilse de bu çalışma için ilk aşamada CST programı üzerinden daha önceden tasarlanmış mikroşerit anten kullanılmıştır.



**Şekil 3.5 :** (a) CST’de tasarlanan antenin 3B görüntüsü, (b)  $S_{11}$  parametresi, (c) kazancı.

Şekil 3.5 incelendiğinde tasarlanan antenin yönü ve merkez frekansı iyi ayarlanamamıştır. Üretim aşamasından sonra da 2.4 GHz için performansın beklenen düzeyde olmamasından dolayı hazır vivaldi anten kullanılması tercih edilmiştir. 2.4 GHz bandında genellikle WiFi antenleri bulunduğu için hazır yönlü anten seçeneği bulmak mümkün değildir. Bu yüzden ultra geniş bantlı 1.4-10.5 GHz vivaldi anten yönlülüğünden dolayı tercih sebebi olmaktadır.

### 3.3 Tasarım Parametreleri

Radarın tasarımında parametreleri belirlerken en önemli sınırlamanın Adalm-Pluto özellikleri olduğundan daha önceden bahsetmiştik. Seçilebilecek maksimum örnekleme frekansının ( $f_s$ ) da 30.72 MHz olabilmektedir. FFT boyutunu belirlerken:

$$f_s = \frac{N}{T_s} \quad (3.4)$$

Denklem 3.4 kullanılmaktadır. Burada N 2<sup>n</sup> boyutunda olursa hesaplama performansını açısından daha iyi olsa da n 14 ve 15 değerleri için N, 16384 ve 32768 hesaplanmaktadır. Burada n=15 değeri için maksimum kanal bant genişliğini

aşacaktır. Diğer yandan  $n=14$  değeri için ise 30.72 MHz e göre çok daha küçük bir örnekleme frekansı kullanılması gerekeceğinden optimum olmasa da 30.72 MHz değeri tercih edilmiştir.  $T_s$  chirp süresi de 1 ms seçilerek yüksek hesaplama işlerinden kaçınılmıştır. Bu şekilde mesafe için FFT boyutu Denklem 3.4 kullanılarak 30720 olarak hesaplanmıştır. Denklem 3.1 de göz önünde bulundurularak chirp bant genişliği ( $\Delta f$ ) B, ideal koşullar sağlanamayacağından dolayı 12 MHz olarak seçilmiştir. Belirlenen parametrelerle Denklem 2.8 hesaplanacak olursa mesafe çözünürlüğü:

$$\Delta R = \frac{3 \times 10^8}{2 \times 12 \times 10^6} = 12.5 [m] \quad (3.5)$$

olarak bulunmaktadır.

Hız çözünürlüğü için parametreler belirlenirken M, chirp sayısı hız çözünürlüğünü doğrudan etkilemektedir. Test aşamasında bilgisayar donanımının kaldırdığı ve yenileme hızının en iyi olduğu M değerinin 32 olduğu tespit edilmiştir. Denklem 2.10'da parametreler yerine koyulursa hız çözünürlüğü:

$$\Delta v = \frac{3 \times 10^8}{2 \times 2.4 \times 10^9} \frac{1}{32 \times 1 \times 10^{-3}} \approx 1.95 [m/s] \quad (3.6)$$

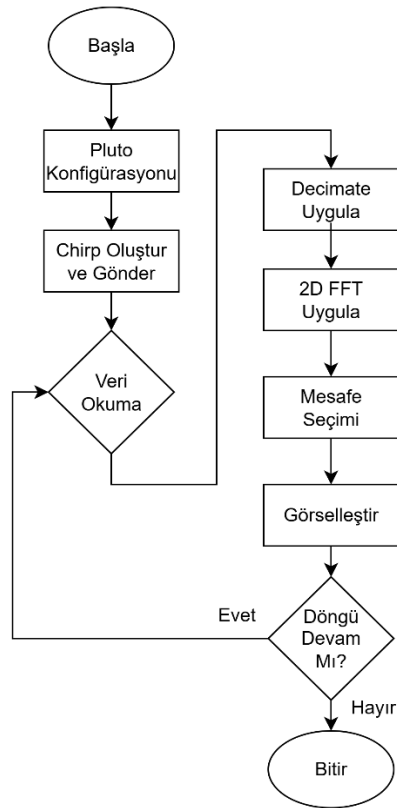
olarak hesaplanmaktadır. Hassasiyet yüksek olmasa da deneysel sonuçlar için kabul edilebilir bir çözünürlük değeridir.

### 3.4 Yazılım Detayları

Bu bölümde radarın algoritma şemasına ve yazılım detaylarına yer verilecektir.

Radarın yazılım kısmında Matlab gerçek zamanlı sinyal işleme uygulamalarında yüksek performans sergilese de Adalm-Pluto ile uyumluluğundan dolayı Python (Spyder derleyicisi) kullanılarak gerçekleştirilmiştir. Python içerisinde bulunan adi kütüphanesi sayesinde Pluto konfigürasyonları daha rahat ve kolay bir şekilde yapılabilmiştir. Ayrıca Matlab kadar olmasa da Python içerisindeki scipy kütüphanesindeki hazır chirp fonksiyonu ile LFM daha pratik halde oluşturulabilmektedir. GNU Radio kullanılarak da gerçek-zamanlı işlemler yapılabilse de range-doppler haritası için harici Python bloğu yazılması gerektiğinden GNU Radio da tercih edilmemiştir.

Radarın hazır donanım kullanılmasından dolayı yazılım işlemleri daha çok önem arz etmektedir.



Şekil 3.6 : FMCW radar algoritma şeması.

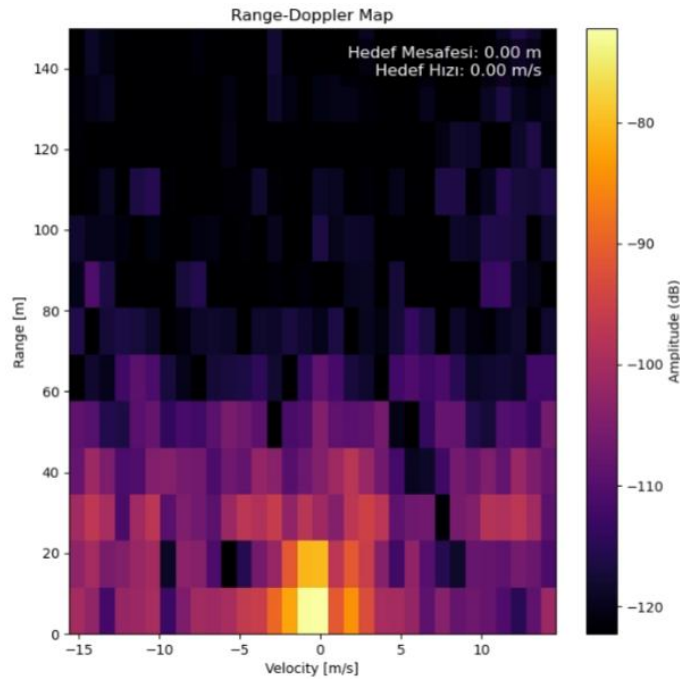
Şekil 3.6’da radar için oluşturulan algoritma şeması görülmektedir. Detaylıca incelendiğinde algoritma başladıktan sonra Bölüm 3.3 Tasarım Detayları kısmında değinilen parametre değerleri ile Pluto konfigürasyonu yapılmaktadır. Daha sonra Scipy kütüphanesindeki hazır chirp fonksiyonu ile sawtooth (testere dişi) lineer frekans modülasyonu yapılarak M adet chirp uç uca eklenerek kompleks olarak Adalm-Pluto TX portundan gönderilir. Burada Pluto gönderilecek sinyali kompleks biçimde beklemektedir. Gerçek zamanlı olarak verinin hem alınıp hem de işlenebilmesi için while döngüsü kurulmaktadır. Bu sayede Pluto RX portu sürekli olarak veri alışı sağlayabilmektedir.

Alınan veri ilk aşamada dijital karıştırma ve filtreleme işleminden geçerek sadece istenilen fark frekansı elde edilmektedir. Elde edilen sinyalin bant genişliği gönderilen sinyalden daha düşük olacağı için decimate işlemi (downsampling) yapılarak örnekleme oranı düşürülür. Filtreleme işlemi ile örnekleme oranının düşürülmesi tek bir fonksiyon içerisinde yapılmaktadır.

Bu işlemler  $M$  adet chirp sinyal vektörü halinde paketlenir. Her chirp sinyaline  $N$  boyutlu FFT uygulanarak toplam  $N \times M$  boyutunda bir matris elde edilir ve mesafe bilgisi çıkartılır. Daha sonra bu matrise 2D FFT uygulanarak hız bilgisi de elde edilir. Mesafe için bir kısıtlama işlemi yapıldıktan sonra da Matplotlib kütüphanesi ile gerçek zamanlı olarak görselleştirme yapılmaktadır. Klavyede Ctrl+C kombinasyonu yapılmadığı sürece bu işlemler devam eder. Şekil 3.6'da verilen algoritmanın daha detaylı hali olan kaynak kodu Ekler kısmında EK A başlığı altında verilmiştir.

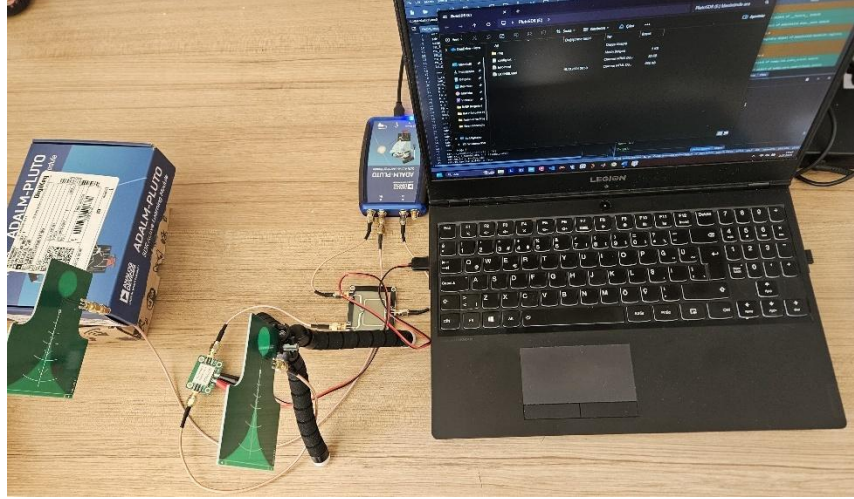
#### 4. SONUÇLAR

Önceki bölümlerde anlatılan tüm detaylar ve bilgilerin deneysel ve uygulamalı sonuçlarına bu bölümde yer verilmektedir.



Şekil 4.1 : FMCW radar deneysel ölçüm sonucu.

Şekil 4.1'de radar önündeki insan hareketi deneysel olarak gözlemlendiğinde sarı bölgenin (daha yüksek dB değeri) hız ekseninde sola doğru kaydığı gözlemlenmektedir. Sağ üst köşedeki hız bilgi panelinde bu değişimin görülmemesinin sebebi ise Python'da gerçek-zamanlı işlemlerde performans kaybının çok olmasından kaynaklanmaktadır.



**Şekil 4.2 :** Tasarlanan FMCW radar sisteminin fiziksel görünümü.

Yapılan ölçümler sonucunda 12 MHz chirp bant genişliği için maksimum 12.5 m mesafe çözünürlüğü elde edilebilmektedir. Bu da Adalm-Pluto gibi düşük güç üreten RF devreleri ile daha uzak mesafelerdeki cisimlerin algılanmasını pek mümkün kılmamaktadır. Bu sebeple 12.5 m içerisinde herhangi bir cisim olmasına rağmen cismi 0 m’de algılamaktadır. Chirp bant genişliğinin yüksek olması bu problemlerin oluşmasına engel olmaktadır. Bu sebeptendir ki yüksek hassasiyet gerektiren uygulamalarda daha yüksek frekans değerleri (24 GHz , 77 GHz gibi) tercih edilmektedir.

## KAYNAKLAR

**Ju, Y., Jin, Y., & Lee, J.** (2014, October). Design and implementation of a 24 GHz FMCW radar system for automotive applications. In *2014 International Radar Conference* (pp. 1-4). IEEE.

**Charvat, G. L.** (2014). *Small and short-range radar systems*. CRC Press.

**Jankiraman, M.** (2018). *FMCW radar design*. Artech House.

**Suleymanov, S.** (2016). *Design and implementation of an FMCW radar signal processing module for automotive applications* (Master's thesis, University of Twente).

**Analog Devices.** (2020). *AD9361 RF Agile Transceiver Data Sheet (Rev. F)*. Retrieved from <https://www.analog.com/media/en/technical-documentation/data-sheets/ad9361.pdf>

**Ulaby, F. T.** (2004). *Fundamentals of applied electromagnetics*. Pearson.

**RF Micro Devices.** (2013). *S04436: Linear Power Amplifier Module for Small Cell Base Stations [Data sheet]*. Retrieved from [https://www.mouser.com/datasheet/2/412/RFMDS04436\\_1-2564742.pdf](https://www.mouser.com/datasheet/2/412/RFMDS04436_1-2564742.pdf)

## EKLER

### EK A: FMCW Radar Kaynak Kodu

```
import adi
import numpy as np
import matplotlib
import matplotlib.pyplot as plt
from scipy.signal import decimate, chirp

""" Pluto Configurations """
sdr = adi.ad9361(uri='ip:192.168.2.1')
samp_rate = 30.72e6 # must be <=30.72 MHz
rx_lo = 2.4e9 # 2.4 GHz
rx_mode = "fast_attack"
rx_gain0 = 70
rx_gain1 = 70
tx_lo = rx_lo
tx_gain0 = -10

""" Radar Parameters """
T_chirp = 1e-3 # chirp süresi (1ms)
num_samps = int(T_chirp * samp_rate) # Tek chirp için örnek sayısı
num_chirps = 32
B = 12e6
k = B / T_chirp
fs = samp_rate
N = num_samps
ts = 1 / fs
t = np.arange(0, T_chirp, ts)
f0 = 0
f1 = 12e6
decim = 50
c = 3e8

""" RX Configuration """
sdr.rx_enabled_channels = [0, 1]
sdr.sample_rate = int(samp_rate)
sdr.rx_lo = int(rx_lo)
sdr.gain_control_mode = rx_mode
sdr.rx_hardwaregain_chan0 = int(rx_gain0)
sdr.rx_hardwaregain_chan1 = int(rx_gain1)
sdr.rx_buffer_size = num_samps

""" TX Configuration """
sdr.tx_enabled_channels = [0]
sdr.tx_rf_bandwidth = int(samp_rate)
sdr.tx_lo = int(tx_lo)
sdr.tx_cyclic_buffer = True
```

```

sdr.tx_hardwaregain_chan0 = int(tx_gain0)
sdr.tx_buffer_size = num_samps

""" Chirp Configuration """
phase = 2 * np.pi * chirp(t, f0=f0, t1=(N*ts), f1=f1, method='linear', phi=0)
i = np.cos(phase) * 2**14
q = np.sin(phase) * 2**14
chirp_iq = i + 1j * q
iq0 = np.tile(chirp_iq, num_chirps)
sdr.tx(iq0)

cmn = "
PRF = 1 / T_chirp
wavelength = c/rx_lo
max_doppler_freq = PRF / 2
max_doppler_vel = max_doppler_freq * wavelength / 2

""" Real-time Görselleştirme """
plt.ion()
fig, ax = plt.subplots(1, figsize=(8,8))
cmaps = ['inferno', 'plasma']
cmn = cmaps[0]
ax.set_ylabel('Range [m]')
ax.set_title('Range-Doppler Map')
ax.set_xlabel('Velocity [m/s]')
img = ax.imshow(np.zeros((num_samps, num_chirps // (2 * decim))), origin='lower',
                 aspect='auto', extent=[-max_doppler_vel, max_doppler_vel, 0, 100],
                 cmap=matplotlib.colormaps.get_cmap(cmn))
cbar = plt.colorbar(img, ax=ax)
cbar.set_label("Amplitude (dB)")
text_handle = ax.text(0.97, 0.97, "", transform=ax.transAxes,
                     fontsize=12, color='white',
                     verticalalignment='top', horizontalalignment='right',
                     bbox=dict(facecolor='black', alpha=0.5, pad=5))

def range_doppler_fft (data, num_chirps, num_samps, beta):
    global fs_dec, fs, decim, k, c, rx_lo, T_chirp

    win_range = np.kaiser(num_samps, beta=beta)
    range_fft = np.fft.fft(data * win_range, axis=1)
    range_fft = range_fft[:, :num_samps // 2]

    win_doppler = np.kaiser(num_chirps, beta=beta)[: , None]
    doppler_fft = np.fft.fftshift(np.fft.fft(range_fft * win_doppler, axis=0), axes=0)

    rd_mag = np.abs(doppler_fft) / np.max(np.abs(doppler_fft))
    rd_dbfs = 20*np.log10(rd_mag/(2**12))

    fs_dec = fs / decim

```



```

    freqs = np.fft.fftfreq(num_samps_decimated, 1/fs_dec)[:num_samps_decimated //
2]
    real_range = freqs * c / (2*k)
    doppler_freqs = np.fft.fftshift(np.fft.fftfreq(num_chirps, 2 * T_chirp))
    real_velocity = doppler_freqs * c / (2 * rx_lo)

    return rd_dbfs, real_range, real_velocity

print("Radar ON")

""" Radar Döngüsü """
try:
    while True:
        rx_data = np.zeros((num_chirps, num_samps), dtype=complex)

        for i in range(num_chirps):
            data = sdr.rx()
            rx_data[i,:] = data[0] * np.conj(data[1])

        rx_data_decimate = np.array([
            decimate(rx_data[i,:], q=decim, ftype='fir', zero_phase=True)
            for i in range(num_chirps)])
        num_samps_decimated = rx_data_decimate.shape[1]

        # 2D FFT
        rd_dbfs, real_range, real_velocity = range_doppler_fft(rx_data_decimate,
                                                                num_chirps=num_chirps,
                                                                num_samps=num_samps_decimated,
                                                                beta=5)

        # Max Range
        min_range = 0
        max_range = 150
        range_indices = np.where((real_range >= min_range) & (real_range <=
max_range))[0]
        range_cropped = real_range[range_indices]
        rd_dbfs_cropped = rd_dbfs[:, range_indices].T

        img.set_data(rd_dbfs_cropped)
        img.set_extent([real_velocity[0], real_velocity[-1],
                        range_cropped[0], range_cropped[-1]])
        vmax = np.max(rd_dbfs_cropped)
        vmin = vmax - 50
        img.set_clim(vmin=vmin, vmax=vmax)
        plt.pause(0.01)

        max_idx = np.unravel_index(np.argmax(rd_dbfs_cropped, axis=None),
rd_dbfs_cropped.shape)
        rng_idx, vel_idx = max_idx
        target_range = range_cropped[rng_idx]

```

```

        target_velocity = real_velocity[vel_idx]

        print(f'Hedef Mesafesi: {target_range:.2f} m | Hedef Hızı: {target_velocity:.2f} m/s')
        text_handle.set_text(f'Hedef Mesafesi: {target_range:.2f} m\nHedef Hızı: {target_velocity:.2f} m/s')

except KeyboardInterrupt:
    sdr.tx_destroy_buffer()
    print("Radar OFF")
    plt.ioff()

```

## ÖZGEÇMİŞ

**Ad-Soyad** : Tunahan ÇIKCAN  
**Doğum Tarihi ve Yeri** : 28/07/2001 Mersin/Türkiye  
**E-posta** : tunahancikcan@gmail.com

Tunahan ÇIKCAN, 2001 yılında Mersin’de doğdu. 2019 yılında Bursa Teknik Üniversitesi Elektrik-Elektronik Mühendisliği bölümünde lisans eğitimine başladı. Lisans eğitimi süresince 1.5 yıl boyunca Dolunay Teknoloji Takımında Elektronik Birim Liderliğini yürüttü. Aynı zamanda radar sistemleri alanında projeler yürüttü. Anten tasarımı üzerine çalışmalar yaptı. Python, Matlab, GNU Radio ve CST Studio Suite ile projeler geliştirdi. RF ve mikrodalga mühendisliği alanında kariyerine devam etmeyi hedeflemektedir.