

Yazılım Tanımlı Radyo Kullanarak Uçan Bir Platformdan Telemetri İletimi

Telemetry Transmission From a Flying Platform Using Software Defined Radio

Nihan Yılmaz¹, Algı Küçükyavuz², Rabia Kapan³ Cenk Toker*

Elektrik ve Elektronik Mühendisliği Hacettepe Üniversitesi

{nihanyilmaz¹, algikucukyavuz², rabiakpn17³}@hacettepe.edu.tr, cenk.toker@ee.hacettepe.edu.tr*

Özet

Bu bildiri uçan bir platformdan alınan telemetri verisinin yazılım tanımlı radyo kullanarak yer istasyonuna iletimini anlatan bir çalışmayı sunar. Drone üzerindeki haberleşme sistemi ile yer istasyonu arasındaki veri iletimi LimeSDR ve USRP gibi yazılım tanımlı radyo kullanılarak yapılmıştır. Drone uçuş denetleyicisi Pixhawk'tan alınan telemetri verileri Frekans Kaydırmalı Anahtarlar tipinde modülasyon kullanılarak yeryüzüne gönderilmiştir. Drone üzerindeki çeşitli parametreleri, örn. hız, pil seviyesi, tüketilen güç, ivme, irtifa, titreşim ve yuvarlanma, yalpalama ve perde gibi jiroskop değerleri yere gönderilir. Raspberry Pi, küçük olduğu için drone üzerinde LimeSDR'ı çalıştıran bilgisayar görevi görür. Bu çalışma, drone baz istasyonu projelerinin temeli niteliğinde görülebilir.

Anahtar Kelimeler: Drone, kablosuz iletişim, FSK, dijital modülasyon, telemetri, LimeSDR, USRP, SDR, Raspberry Pi, Pixhawk, GNURadio.

Abstract

This paper presents a study describing the transmission of telemetry data received from a flying platform to the ground station using software defined radio. Data transmission between the communication system on the drone and the ground station has been made using software defined radios such as LimeSDR and USRP. Telemetry data taken from the drone flight controller Pixhawk is sent to the ground by using Frequency Shift Keying modulation type. Various parameters of the drone e.g. speed, battery level, consumed power, acceleration, altitude, vibration and gyroscope values such as roll, yaw and pitch are sent to the ground. Raspberry Pi acts as the computer that runs LimeSDR on the drone since it is small. This work can be seen as the foundation of drone base station projects.

Keywords: Drone, wireless communication, FSK, digital modulation, telemetry, LimeSDR, USRP, SDR, Raspberry Pi, Pixhawk, GNURadio.

1. Giriş

Bu çalışma temel olarak uçan bir platform üzerine yerleştirilmiş kablosuz bir iletişim sistemidir. İnsan-sız hava araçları gibi uçan platformların haberleşme sistemlerinde kullanımı oldukça yaygındır. Özellikle, hareketlilik, esneklik, uyarlanabilir irtifa ve uygun maliyet gibi nitelikleriyle, İHA'lar kablosuz sistemlerde birkaç önemli potansiyel uygulamaya imkan sağlar.

Baz istasyonları, karasal iletişim ağlarının bir parçasıdır. Karasal iletişim ağları sel, deprem vb. felaketler ve aşırı koşullar nedeniyle kesintiye uğrayabilir. Bu gibi durumlarda, yüksek irtifaya sahip dronlar ile iletişim ağları, yalnızca havacılık sınırlamaları ile kısıtlanan herhangi bir yörünge boyunca hareket ederek kişilerin güvenliğini sağlar.

İHA'lar ayrıca kapsama alanını, kapasiteyi, güvenilirliği, görüş hattı bağlantı olasılığını ve kablosuz ağların enerji verimliliğini artırmak için de kullanılabilir. Öte yandan, İHA'lar bir hücresele ağ içinde uçan mobil terminaller olarak da çalışabilir. Bu tür hücresele bağlantılı İHA'lar, gerçek zamanlı video akışından ürün teslimine kadar çeşitli uygulamaları etkinleştirebilir. Bu nitelikler bir araya geldiğinde, drone baz istasyonları yeni bir iletişim dünyasının kapılarını açar [1-3].

Drone baz istasyonu literatürde pek çok şekilde yer alırken endüstride farklı firmalar tarafından farklı şekillerde uygulanmaktadır. Bu çalışmaların literatürde yer almasının sebebi, insansız hava araçlarının hareketli ve esnek olması açısından kablosuz haberleşmede yüksek potansiyele sahip olmasıdır. İnsansız hava araçlarının iletişimi sırasında yaşanabilecek aksaklıkların giderilmesinde optimizasyon teorisi, makine öğrenmesi ve oyun teorisi gibi farklı yöntemlerle çözümler sunan çalışmalar da vardır [4]. Hücresele haberleşmede, maksimum kullanıcıya hizmet edebilmek için drone baz istasyonlarının optimum dağılımını bulmayı hedefleyen çalışmalar da yürütülmektedir [5]. Bunun yanı sıra, insansız hava araçlarının güç ve uçuş dinamikleri açısından enerji verimli konuşturulması da ele alınmaktadır [6].

Drone baz istasyonu denildiğinde, bir haberleşme teknolojisi de belirlenmelidir. Kablosuz haberleşme düşünüldüğünde, Wi-Fi, Bluetooth, 4G ve 5G gibi pek çok teknoloji akla gelmektedir. Bu teknolojileri destekleyen birçok cihaz da literatürde mevcuttur. Günümüzde, değiştirilebilir olması, hafif olması ve birçok modülasyon türünü desteklemesi açısından haberleşmeye yeni bir soluk getiren yazılım tanımlı

radyolar, drone baz istasyonları için büyük bir fırsat sunmaktadır. RTL-SDR, HackRF-One, LimeSDR ve USRP bu cihazların arasında yer almaktadır. Yazılım tanımlı radyolar, kolay değiştirilebilmesinin yanı sıra, programlanmasında GNURadio, LabView ve Matlab gibi birçok farklı programlama seçeneği ile kolayca erişilebilir olduklarından, bir haberleşme istasyonu için haberleşme modülü oluşturulurken pek çok avantaj sağlamaktadır.

Yazılım tanımlı radyolar ile oluşturulan haberleşme sistemlerine literatürde sıkça rastlanmaktadır. Farklı yazılım tanımlı radyo cihazlarıyla farklı modülasyon tekniklerini içeren haberleşme sistemleri tasarlanmıştır [7-9]. Yapılan araştırmalar temel alınarak, bu çalışmada iletişim sistemi tasarımında esneklik sağlayan yazılım tanımlı radyolar ile drone kullanımı fikirleri birleştirilmiştir. Bu çalışmada, yer istasyonu için alıcı/verici görevi yapmak üzere USRP, drone platformu için de hafif olması ve hareket kabiliyeti katması açısından alıcı/verici olarak LimeSDR tercih edilen bir haberleşme istasyonu tasarımı yer almaktadır.

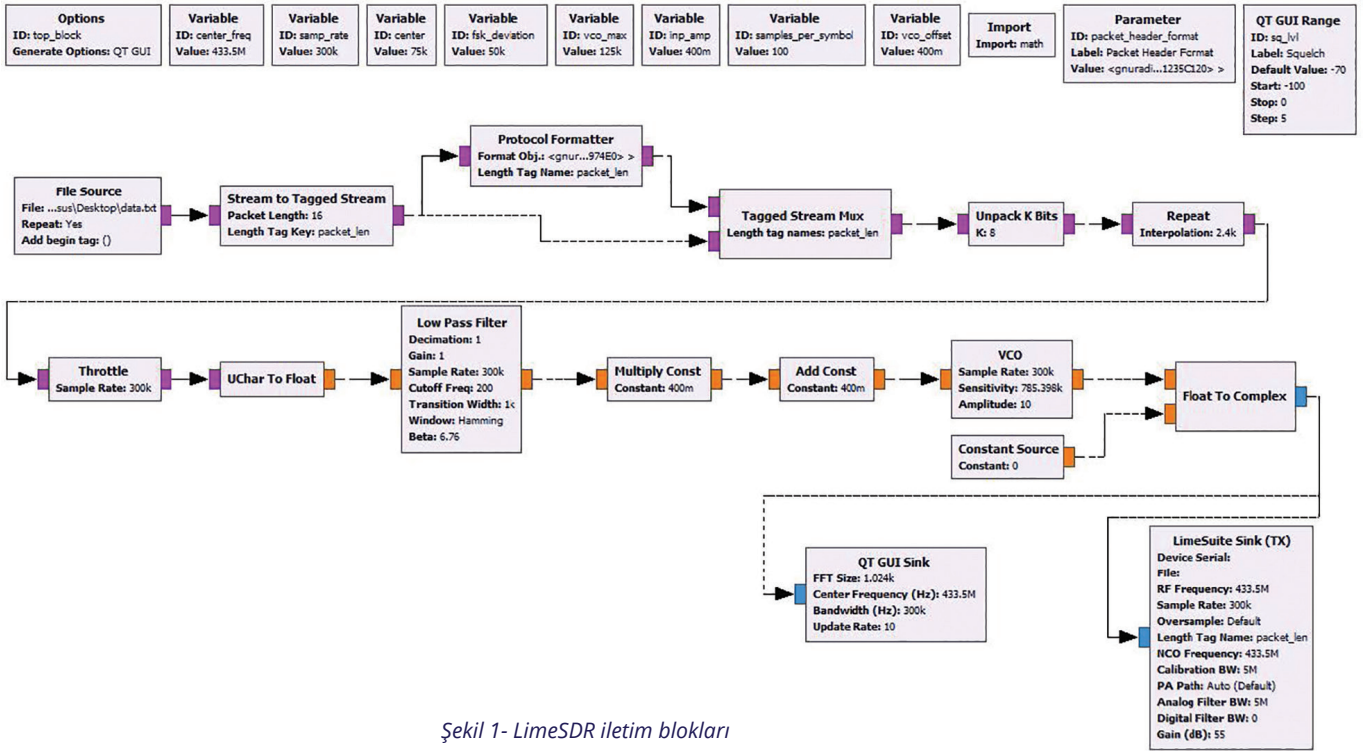
2. Kablosuz Haberleşme

Yazılım tanımlı radyo

Teknolojinin gelişmesine rağmen birçok haberleşme cihazında en önemli sorunlardan biri, protokollerin ve radyonun donanım temelli olmasıdır. Bu durum, sistemi yeniden programlama ve biçimlemeyi minimal bir düzeyde kısıtlamak zorunda bırakır. Dolayısıyla bir hata olması durumunda bunu düzeltmek için ucuz ve kolay bir çözüm yolu bulmak pek mümkün değildir. Ayrıca, bu tarzda cihazlar genel olarak tek tür protokole uyacak şekilde kısıtlıdır. Yazılım tanımlı radyo platformlarının esnekliği, bütün devreyi sadece yazılımla değiştirebilmekten gelmektedir. Normalde, donanımla uygulanan haberleşme protokolleri, yazılımla uygulanıp iletişim sistemi yazılımla güncellenebilir. Bu sebeple, günümüzde bu cihazlar iletişim araçları olarak sıkça kullanılmaktadır [10]. LimeSDR ve USRP en çok kullanılan yazılım tanımlı radyolardan ikisidir. Her iki cihaz da farklı modülasyonları ve protokolleri desteklemektedir.

İletişim

LimeSDR ile USRP arasında iletilecek telemetri verileri bayt türünde yani dijital veriler şeklindedir. Dolayısıyla kullanılacak modülasyon türü dijital veriyi analog bir sinyale dönüştürecek şekilde olmalıdır. Bu nedenle dijital bir modülasyon türü olan frekans kaydırmalı anahtarlama kullanılmıştır.



Şekil 1- LimeSDR iletim blokları

İletişimde cihazları programlayabilmek için GNU-Radio kullanılmıştır. GNURadio'da modülasyon ve demodülasyon yanında veri senkronizasyonu üzerine de çalışılmıştır [11]. GNURadio'da kurulan iletim blokları diyagramı Şekil-1'de gösterilmiştir.

Bu işlemler için öncelikle, kaydedilen telemetri verilerinden oluşan dosya paket halinde alınır. Bit senkronizasyonunu sağlamak için her paketin başına bir header kısmı eklenir. Daha sonra bu paket bayt boyutundan bitlere ayrılır ve vektör haline getirilir. Bitlere ayrılan veri vektörü, repeat bloğundan geçirilerek bit örneklerinin sayıları interpolasyonla artırılır. Buradaki interpolasyon değeri, sistemde kullanılan örnek sayısı ve gönderim hızıyla doğru orantılıdır. Gönderilecek verinin boyutu da göz önüne alınarak ideal olması sebebiyle 100 bit/s baud hızı esas alınmıştır. Veriyle beraber paket başında header da olması sebebiyle asıl veriyi 100 bit/s hızda ulaştırabilmek için veri hızı 125 bits/s olacak şekilde hesaplanmıştır. Buradan interpolasyon değeri örnek sayısı ile 1/125 saniyelik bit süresi çarpımı şeklinde hesaplanır. Bit interpolasyonu devamında vektördeki veriler karakter tipinden sayılara dönüştürülür. Dönüştürülen sinyal filtrelenir. Devamında, ikili frekans kaydırmalı anahtarlama kullanılması sebebiyle sinyalleri istenilen frekanslara eşleyebilmek için vektördeki sayılar belirli seviyelere çekilir. Bu, Multip-

ly Const ve Add Const bloklarıyla yapılır. Tasarlanan sistemde Çizelge-1'deki ve eşitliklerdeki değerler kullanılmıştır.

Çizelge 1: Haberleşme Sistemi İçin Seçilen Parametre Değerleri

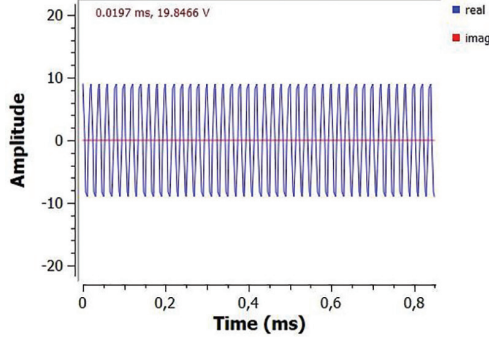
Parametreler	Değerler (kHz)
f1	50
f2	100
Frekans kayması (Δf), $fsk_deviation$	50
Merkez Frekansı, center	75
Tam Ölçekli Frekans, vco_max	125
Taşıyıcı Frekansı	433500

$$vco_{offset} = \frac{f1}{vco_{max}} \quad (1)$$

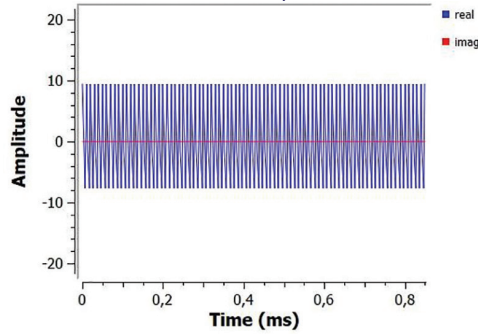
$$inp_{amp} = \frac{f2}{vco_{max}} - vco_{offset} \quad (2)$$

$$vco_{sensitivity} = 2\pi vco_{max} \quad (3)$$

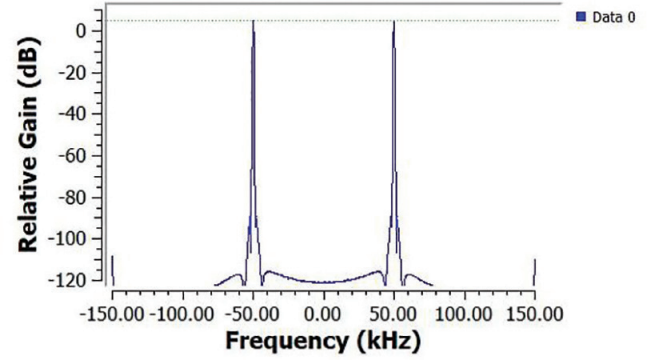
Bu değerlerle, voltaj kontrollü osilatör kullanılarak sinyal belirli frekans değerlerine sahip analog bir sinyale dönüştürülür. Oluşan bu sinyal gelen bit değerine göre önceden belirlenen frekans değerine yukarıda belirtilen eşitlikler sayesinde eşlenir. Ardından bu sinyal önce LimeSDR'a iletilip antenden kanal aracılığıyla yer istasyonuna iletilir. İletilen sinyaller, Şekil-2 ve Şekil- 3'te gösterildiği gibidir.



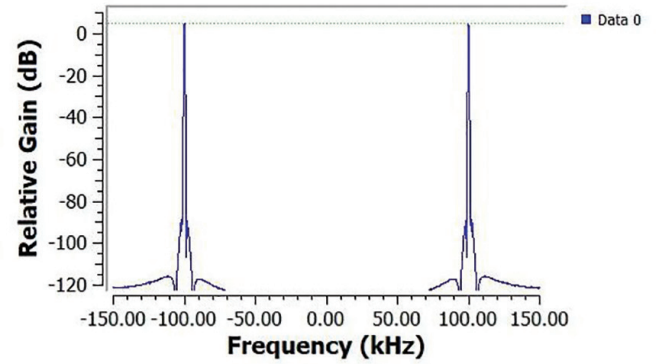
Şekil 2- İletilen Sinyalin Zaman Eksenindeki Görünümü ($f_1 = 50\text{kHz}$ ekran alıntısı)



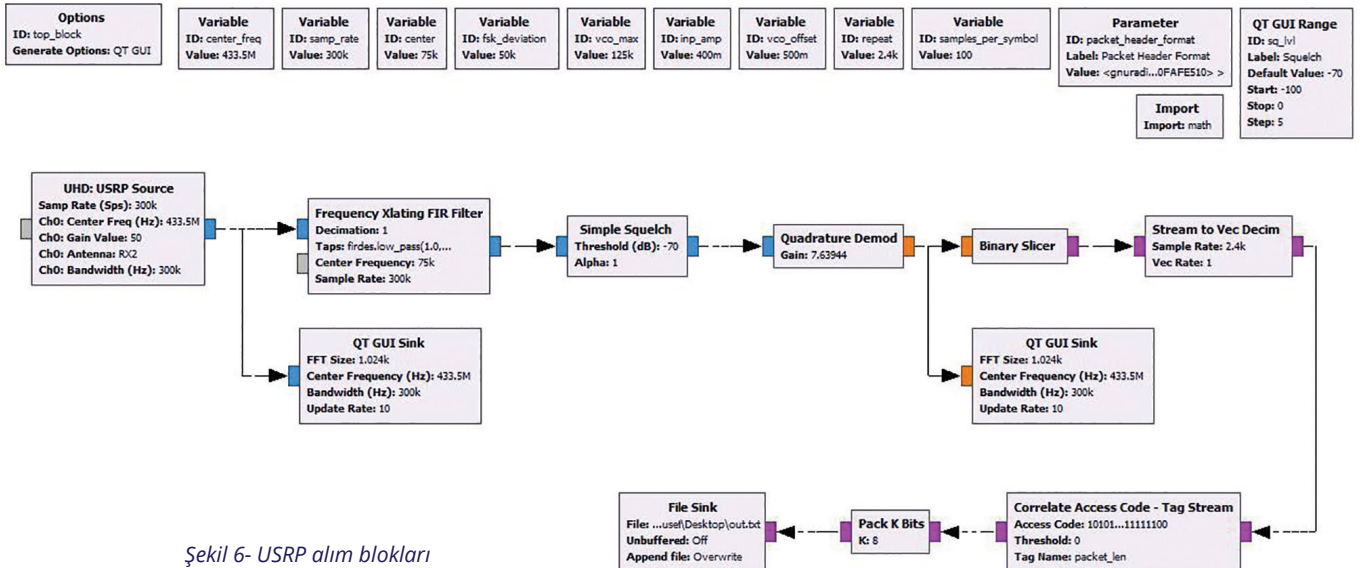
Şekil 3- İletilen Sinyalin Zaman Eksenindeki Görünümü ($f_2 = 100\text{kHz}$ ekran alıntısı)



Şekil 4- İletilen Sinyalin Frekans Ekranı ($f_1 = 50\text{kHz}$)



Şekil 5- İletilen Sinyalin Frekans Ekranı ($f_2 = 100\text{kHz}$)



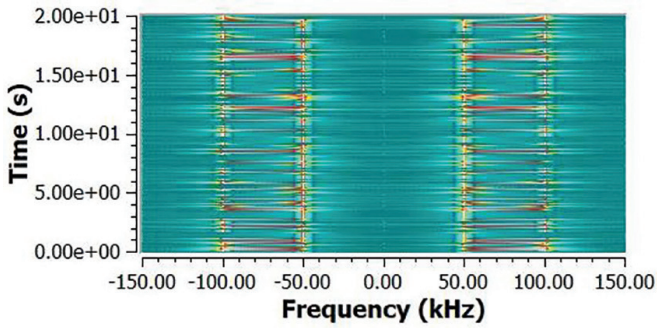
Şekil 6- USRP alım blokları

Şekil-2'de 50kHz'e denk gelen sinyal aralığı, Şekil-3'te 100kHz'e denk gelen sinyal aralığı görülmektedir.

Şekil-4 ve Şekil-5'te frekans ekseninde 50kHz ve 100kHz değerlerinde sinyalin zıpladığı görülmektedir. USRP tarafından alınan sinyal, GNURadio'da kurulan alım bloklarında işlenir. Alım bloklarının diyagramı Şekil-6'da görüldüğü gibidir.

Şekil-7'de şelale ekranı gösterilen bu iki frekanslı sinyal, filtrelendikten sonra Quadrature Demod bloğu ile iki seviyeli bir sinyale dönüştürülür. Bu bloğun girişi kompleks taban bantlı bir sinyaldir ve çıkışı örnek sayısı ve blok kazancıyla ilişkili frekans değerleridir. Blok içindeki kazanç değeri aşağıdaki eşitlikteki gibi hesaplanır.

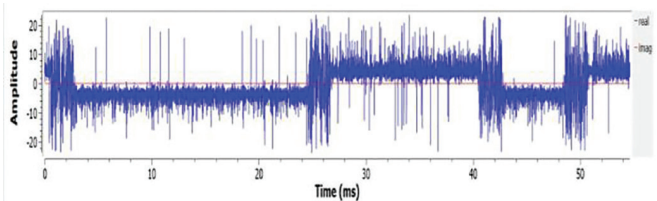
$$Gain = \frac{samprate}{2\pi fsk_{deviation}} \quad (4)$$



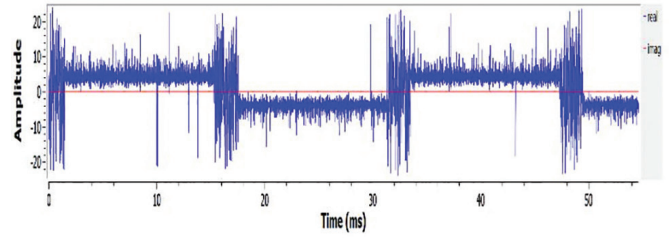
Şekil 7- Alınan sinyalin şelale ekranı

Şekil-8 ve Şekil-9'da görülen Quadrature Demod bloğunun çıkışındaki iki seviyeli sinyal, 1 ve 0 şeklinde bitlere sinyal seviyesine göre kestirilir. Elde edilen bit örnekleri, orijinal örnek sayısına ulaşabilmek için Stream to Vec Decim bloğundan geçirilir. Bu bloktaki örnek sayısı iletim bloklarındaki repeat bloğunun değeriyle aynıdır. Orijinal örnek sayısındaki bit vektörü elde edildikten sonra, vektördeki header kısmına kilitlenilir ve bu kısım yakalandıktan sonra vektörden çıkartılıp bit vektöründen byte haline yeniden paketlenir. Verinin son hali kaydedilir.

Taşıyıcı frekansı Türkiye'deki ISM bandı aralıklarında olan 433.5 MHz olarak seçilmiştir.



Şekil 8- Quadrature Demod bloğunun çıkışı-I



Şekil 9-Quadrature Demod bloğunun çıkışı-II

Platform

Uçan platform için önemli bazı parametreler, uçuş süresinin olabildiğince fazla olması ve mümkün olan en fazla miktarda yükü taşıyabilmesidir. Bu sebeple drone parçaları seçilirken bunlar göz önünde bulundurulmuştur. Ağırlığı kaldırabilmesi için hexacopter drone tasarımı tercih edilmiş, uçuş süresini maksimize edebilmek için 3S Li-Po pil kullanılmış, buna uygun ESC, motor ve pervaneler tercih edilmiştir. Tercih edilen parçalarla, drone birleştirilmiştir. Platform üzerindeki uçuş denetleyicisi Pixhawk, uçuş kontrolünü sağlayan ve telemetri verilerini toplayan birimdir. Uçuş kontrolünün, otonom şekilde sağlanması için Python DroneKit kütüphanesi kullanılarak otopilot kodu yazılmıştır. Kumanda ile uçuş yerine, otonom uçuşun tercih edilmesi beraberinde, daha güvenli ve dengeli bir uçuşu getirmiştir. Bu şekilde drone üzerindeki cihazların daha iyi korunması sağlanmıştır.

Sistem tasarımı

Projede, iletişim araçları ile platformun birleştirilmesi aşaması bulunmaktadır. LimeSDR, drone üzerindeki yazılım tanımlı radyo cihazı olarak drone parametrelerini yer yüzüne iletmekten sorumludur. LimeSDR'ın yazılımını çalıştırmaktan, drone parametrelerini Pixhawk üzerinden mavlink bağlantısı kurarak çekip yazılımı çalıştırmaktan ve bu parametreleri kaydetmekten sorumlu bilgisayar Raspberry Pi'dir.

Drone parametreleri Pixhawk'tan, Python'ın DroneKit kütüphanesi kullanılarak elde edilir. Bu kütüphane sayesinde drone durumu ile ilgili birçok veriye ulaşılabilmektedir. Bu parametreler arasında, voltaj ve akım değerleri, pilin batarya seviyesi, drone yüksekliği, GPS ve jiroskop verileri, drone yuvarlanma, yalpalama ve perde değerleri ile ivme değerleri bulunmaktadır. Bunun yanı sıra, titreşim verileri, piezo disk titreşim sensöründen alınır. Bu veriler, harcanan güç değerlerini ve drone hareket durumunu bilmek açısından önemlidir. Yazılan Python koduyla, Pixhawk'tan alınan parametre verilerinin sonuna, ve-

rinin alındığı saat ve boyutu eklenir. GNURadio, hazırlanıp kaydedilmiş veri paketinin modülasyonunu yapıp LimeSDR aracılığıyla yer istasyonuna iletilmesi için paketi alır.

Drone üzerindeki malzemelerin beslenmesi için platform üzerinde iki batarya kullanılmıştır. Biri drone uçuşu için yerleştirilmiş olup diğeri Raspberry Pi ve LimeSDR beslemesi için kullanılmıştır. Her iki cihaz da batarya çıkış değerlerinden daha az bir güç ile çalışmaktadır. Dolayısıyla, batarya ile cihazlar arasında gerilim düşürücü regülatör kullanılmıştır. LimeSDR'in USB3.0 teknolojisiyle çalışan bir cihaz olması sebebiyle Raspberry Pi ile olan USB bağlantısı yalnızca veri iletişimi için kullanılmış olup asıl güç beslemesi batarya ile sağlanmıştır.

Yer istasyonu, USRP ve bu cihazı çalıştıran bilgisayardan oluşmaktadır. USRP gelen sinyalleri aldıktan sonra GNURadio'da yapılan işlemler sonucunda veriler yer istasyonu bilgisayara kaydedilir.

3. Sonuç

Kurulan düzenek, kapalı (laboratuvar) ortamda ve açık havada test edilmiştir. İlk olarak kapalı alanda yapılan testlerde, örnek sayısı, veri hızı ve alıcı/verici cihaz kazançları değiştirilerek farklı sonuçlar gözlemlenip en ideal olan değerler bulunmuştur. Daha sonra bu değerlerle açık havada testler yapılmıştır. Açık havada yapılan testler ile kapalı alanda yapılan testler kıyaslandığında cihazlar arasındaki mesafenin göz ardı edilemeyecek kadar düştüğü gözlemlenmiştir. Bunun sebebi, LimeSDR çıkış gücünün düşük kalması ve açık havada kanal gecikme yayılmasının kapalı alana göre çok daha fazla olması sebebiyle sinyali fazlasıyla etkilemesidir. LimeSDR'in çıkış gücünü artırıcı yükseltici kullanmak ve gönderilen verinin hızını düşürmek bu duruma çözüm sunabilir.

4. Kaynaklar

- [1] Xiaohui Li, "Deployment of Drone Base Stations for Cellular Communication Without Apriori User Distribution Information", 2018.
- [2] Giselle M. Galvan-Tejada, Jorge E. Aviles-Mejia, Aldo G. Orozco-Lugo, Luis A. Arellano-Cruz, Ruben Flores-Leal and Rogelio Lozano-Leal, "Propagation Characteristics for UAVs Operating at Short Range and Low Altitude", 2020.
- [3] Wahab Khawaja, Ismail Guvenc, David W. Matolak, Uwe-Carsten Fiebig, Nicolas Schneckenberger, "A Survey of Air-to-Ground Propagation Channel Modeling for Unmanned Aerial Vehicles", 2018.
- [4] M. Mozaffari, W. Saad, M. Bennis, Y.-H. Nam, and M. Debbah, "A Tutorial on UAVs for Wireless Networks: Applications, Challenges, and Open Problems" arXiv preprint arXiv:1803.00680, 2018.
- [5] X. Li, "Deployment of Drone Base Stations for Cellular Communication Without Apriori User Distribution Information", 2018.
- [6] U. Demir, M. Ç. Ipek, C. Toker, O. Ekici, "Energy-Efficient Rotary-Wing UAV Deployment Under Flight Dynamics and QoS Constraints", 2019.
- [7] D. Kushnure, M. Jiniyawala, S. Molawade, S. Patil "Implementation of FM Transceiver using Software Defined Radio (SDR)", 2017.
- [8] T. Tekin, B. Karakaya, "OFDM Verici ve Alıcı SDR uygulaması zaman ve frekans kayması durumunda", 2018.
- [9] A. E. Elsaghier, N. S. Tezel, S. M. Altiraiki, "Frequency Shift Keying Scheme to Implement SDR using Hackrf one", 2017.
- [10] José Raúl Machado-Fernández, "Software Defined Radio: Basic Principles and Applications", 2014.
- [11] GNURadio, Simulation example: FSK, 7 Mart 2021. Erişim tarihi: 13 Mart 2021 [Çevrimiçi]. Ulaşılabilir: https://wiki.gnuradio.org/index.php/Simulation_example:_FSK