

Savunma Sanayiinde İHA'lar için Tasarlanmış Genel Amaçlı Simülasyon Aracı

Tuğba KÖKEN¹, Berna TAŞAR GÜLTEKİN²

- Baş Teknik Mühendis, Türk Havacılık Uzay Sanayii, Ankara, koken.tugba@gmail.com, ORCID ID: 0009-0008-8236-6423*
- Baş Teknik Mühendis, Türk Havacılık Uzay Sanayii, Ankara, bernatasar@yahoo.com ORCID ID: 0009-0007-1080-2129*

Geliş Tarihi/Received

28.01.2024

Kabul Tarihi/Accepted

06.02.2024

e-Yayın/e-Printed

08.02.2024

ÖZET

Bu araştırma, savunma sanayii için özel olarak tasarlanmış İnsansız Hava Araçlarının (İHA), performansını ve yeteneklerini analiz etmek amacıyla geliştirilen simülasyon programının, nasıl tasarlandığını ve ne tür özellikleri olduğunu sunmaktadır. Bu çalışmada anlatılan simülasyon programının adı Sistem Entegrasyon Laboratuvarı (SEL) Doğrulama Yazılım Altyapısı (SDYA)'dır. SDYA'nın geliştirilen birçok değişik özelliklere sahip tüm İHA'lar için kullanılabilir olması hedeflenmiştir. Bunun için çok katmanlı ve modüler bir mimaride tasarlanmıştır. İHA'ların kullandığı modüller taklit edilmiş; İHA çeşidine göre çeşitli kombinasyonlarda simülasyon programına entegre edilmiştir. Böylece SDYA, tasarım sürecini optimize etmek, güvenlik ve performansı artırmak ve maliyetleri düşürmek için vazgeçilmez bir araç olmuştur. SDYA kullanılarak test edilen İHA'lar daha kısa zamanda ve az maliyetle kullanıma sunulmuştur.

Anahtar Kelimeler: Simülasyon, İnsansız Hava Aracı, Simülasyon Aracı Tasarımı, Sistem Entegrasyon Laboratuvarı Test Yazılımları.

General Purpose Simulation Tool Designed For UAVs in the Defense Industry

ABSTRACT

This research presents a simulation program developed specifically for Unmanned Aerial Vehicles (UAVs) designed for the defense industry, aiming to analyze the performance and capabilities. The simulation program described in this study is called the System Integration Laboratory (SEL) Verification Software Infrastructure (SDYA). SDYA is designed with a multi-layered and modular architecture to be applicable to various UAVs with different features. Modules used by UAVs are simulated and integrated into the simulation program in various combinations based on the type of UAV. Thus, SDYA has become an indispensable tool to optimize the design process, enhance security and performance, and reduce costs. UAVs tested with SDYA are deployed more quickly and at a lower cost.

Keywords: Simulation, Unmanned Aerial Vehicle, Simulation Tool Design, System Integration Laboratory Test Software

GİRİŞ

İnsansız Hava Aracı (İHA), askeri operasyonlarda keşif, gözetleme, istihbarat toplama, hedef belirleme, lojistik destek ve daha birçok alanda kullanılmaktadır. Bu teknolojinin kullanımıyla birlikte askeri operasyonların etkinliği ve hassasiyeti artmış, personelin risk altında kalmadan görevleri yerine getirebilmesi mümkün hale gelmiştir (İçten ve Bal, 2021; Cevher, 2023).

Burada önem arz eden bir konu da simülasyondur. Bu tür uygulamalar esasında gerçek dünyaya ilişkin olasılıkların önceden tahmin edilebilmesi ve gözlenebilmesiyle beraber çeşitli durumlara ilişkin hazırlık yapılabilmesini sağlamaktadır. Simülasyonun başarılı olabilmesi için öncelikle simüle edilmesi beklenen ya da amaçlanan fiziksel sistemin barındırdığı tüm verilerinin dijital ortama taşınarak modellenmesi gerekmektedir. Simülasyon çok çeşitli sektörlerde kullanılmaktadır. Zira mevcut

gerçekliklere ilişkin olasılıkların herhangi bir zarara mahal vermeden deneyimlenmesini sağlamakta, yeni durumlara göre verilebilecek tepkilerin planlanmasını olanaklı kılmaktadır (Landriscina, 2013; Bungartz vd., 2014).

Savunma sanayii için simülasyon, gerçek dünya koşullarını taklit etmek ve askeri sistemlerin davranışını, performansını veya etkisini değerlendirmek için kullanılan bir modelleme yöntemidir. Bu, askeri araçların, silahların, iletişim sistemlerinin veya diğer savunma teknolojilerinin tasarım ve geliştirme aşamalarında kullanılarak maliyetleri düşürme, riskleri azaltma ve verimliliği artırma amacını taşır. Simülasyonlar, saha testleri öncesi cihazları ve sistemleri sanal ortamda test etmeyi sağlayarak askeri güçler için önemli bir araştırma ve geliştirme aracıdır (Macedonia, 2002; Yao ve Huang, 2021).

Bu makalede, savunma sanayiinde İHA teknolojisinin geliştirilmesine katkı sağlayan bir simülasyon aracının tasarlanması ve kullanılması incelenmiştir. Simülasyon aracı, İHA tasarımlarının ve sistemlerin test edilmesi, performans değerlendirmesi ve iyileştirilmesi için önemli bir araçtır. Aynı zamanda, gerçek dünya koşullarını simüle ederek İHA'ların nasıl tepki vereceği ve hangi durumlarda daha etkin olabileceği konusunda değerli veriler sağlar.

Bu makalede, Sistem Entegrasyon Laboratuvarı Doğrulama Yazılımları Altyapısı (SDYA) aracının geliştirilmesinden, test ve doğrulamaya olan katkılarından, İHA sistemlerinin performansı ve verimliliği nasıl analiz ettiğinden, nasıl kullanıldığından, İHA güvenlik ve savunmasında nasıl yardımcı olduğundan bahsedilmektedir.

Bu çalışmadaki bilgiler toplanırken; konuyla ilgili diğer makalelerden faydalanılmış, uzman görüşleri alınmış ve SEL ortamına gidilerek SDYA aracının kullanımı ve sağladığı kolaylıklar değerlendirilmiştir.

SDYA

SEL, modernize edilen ya da özgün olarak sıfırdan geliştirilen uçak, helikopter, İHA, uydu gibi bir sistemi kuracak olan emniyet kritik ve güvenilir yazılımların ilgili tüm aviyonik sistemlerle birlikte çalışabilirliğinin tecrübe edildiği test ve entegrasyon ortamıdır.

SDYA; SEL'de gerçek aviyonik ekipmanların yerine geçebilen, ekipmanların fonksiyonel davranışlarını taklit edebilen; gerçek aviyonik ara yüzler üzerinden merkezi işlemci Birimleri ve diğer aviyonik ekipmanlarla haberleşebilen; entegre test senaryolarının uygulanabildiği, alt yapı yazılımlarıdır (Nargül vd., 2015).

2.1. Simülasyon İhtiyacı Gereken Durumlar

Simülasyon, yapay bir ortam oluşturmayı veya belirli bir süreç ya da sistemin yapısını taklit etmeyi içerir. Gerçek dünyadaki süreçleri veya sistemleri yüksek derecede gerçekçilikle taklit etme eylemi, insan emeğini en aza indirmekten zamandan tasarruf etmeye ve ekonomik kazançlar elde etmeye kadar çok sayıda avantajı beraberinde getirir. Simülasyonlar, özel eğitim ve testler için risksiz bir yol sağlar (Kızak, 2021). Simülasyonun birincil amacı maliyet etkinliği sağlamaktır. Simülasyonun kullanılmasıyla, belirli bir görevin veya projenin doğruluğunu ve uygulanabilirliğini hızlı ve ucuz bir şekilde değerlendirmek mümkün hale gelir. Bu uygun maliyetli yapı, belirli bir işlemin veya sürecin doğru yürütülüp yürütülmediğinin hızlı ve ekonomik bir şekilde değerlendirilmesini sağlar (Korkmaz vd., 2016).

Simülasyon uygulaması, sistemin deney yapmaya elverişli olmadığı, sistemin tasarımın yeni aşamalarında olduğu, karmaşık bir sistem veya problemle uğraşıldığı veya sistemin davranışının ayrıntılı bir analizinin gerekli olduğu durumlar gibi çeşitli koşullar altında mantıklı hale gelir. Etkili bir simülasyon, etkinliğini sağlamak için belirli özellikleri bünyesinde barındırmalıdır. İlk olarak, farklı senaryoların kaynak verimli bir şekilde araştırılmasına olanak tanıyacak şekilde uygun maliyetli olmalıdır. Ayrıca simülasyon, simüle edilen süreçlerin net bir şekilde anlaşılmasını kolaylaştırarak kullanıcı tarafından kolay anlaşılacak şekilde tasarlanmalıdır. Kontrol ve kullanım kolaylığı, başarılı bir simülasyonun bir diğer önemli yönüdür. Kullanıcılar simülasyonu yönetmeyi ve çalıştırmayı kolay bulmalı, böylece aracın gereksiz komplikasyonlar olmadan amacına hizmet etmesini sağlamalıdır. Buna ek olarak, simülasyonun uyarlanabilirliği kilit öneme sahiptir - model değişiklikleri ve

güncellemeleri sorunsuz bir şekilde uygulanabilmeli ve gelişen gereksinimlere veya içgörülere yanıt olarak simüle edilmiş ortamın iyileştirilmesi için esneklik sağlamalıdır. Özünde, iyi tasarlanmış bir simülasyon yalnızca kullanıcının analitik ihtiyaçlarını karşılamakla kalmaz, aynı zamanda sistem geliştirme veya problem çözmenin çeşitli aşamalarında uygunluğunu sağlamak için kullanıcı dostu bir arayüz, maliyet etkinliği ve uyarlanabilirlik sunar (Koçak vd. 2013'ten akt. Yıldırım, 2019).

2.2. SDYA Geliştirme Yaklaşımı

SDYA, emniyet kritik/güvenilir yazılımların sistem ve yazılım seviyesinde doğrulanması amacıyla tasarlanmıştır. Dışa bağımlılığın önlenmesi amaçlanmış ve her an yeni özellikler eklemeye ve yeni teknolojileri entegre etmeye elverişli olması sağlanmıştır.

2.3. SDYA' nın Kullandığı Teknoloji ve Araçlar

- İşletim Sistemi: Windows 7 - Windows 10
- İşletim Sistemi Mimarisi: X86 - x64
- Yazılımlarının geliştirilmesinde, yeniden kullanılabilirliği arttırmak ve kaynak kodunun bakımını kolaylaştırmak amacıyla nesneye yönelik programlama ve C++ dili tercih edilmiştir (Ambler, 1998).
- SDYA geliştirme sürecinde dağıtık mimari isteklerinin karşılanması ve yazılım modülleri arasındaki veri iletişimini sağlayabilmek için arakatman yazılımı olarak ZeroMQ (ZMQ) tercih edilmiştir (ZeroMQ, 2023).
- Kullanıcı Ara yüzü Geliştirme Aracı: Qt Gui Toolkit (QT Group, 2024).

2.4. SDYA' nın Yazılım Mimarisi

Mimari stiller ya da diğer adıyla mimari desenler sistemler için üst seviye soyutlama sağlayan prensipler topluluğudur. Bu stiller parçalamayı ve tasarımın tekrar kullanımını artırır ve uygulamaya şekil verirler (Sharma vd., 2015).

Mimari stiller odaklandıkları alanlara göre kategorilere ayrılır. SDYA Projesinin Yazılım Tasarımında bu kategorilerden iletişim kategorisinde yer alan desenlerden projenin tasarım hedefleri doğrultusunda Mesaj Veri Yolu (Yayınla/Abone Ol) seçilmiştir. Yapısal kategoride yer alan desenlerden ise Bileşen-Tabanlı Mimari ve Katmanlı Mimari seçilmiştir.

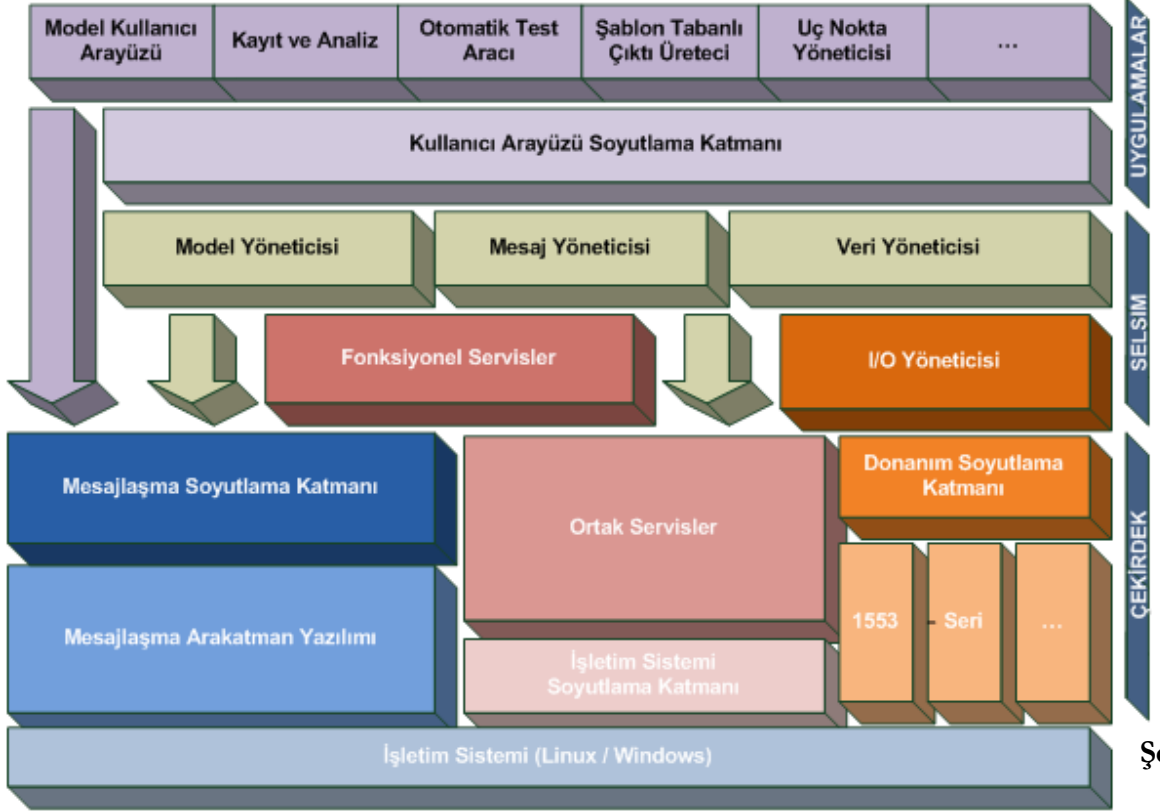
2.4.1. Katmanlı Mimari

Katmanlı mimari, uygulamadaki ilgili fonksiyonların üst üste yerleşen ayrı katmanlara dağıtılarak gerçekleştirilmesine odaklanır. Katmanların birbirleri ile olan iletişimleri açıktır ve bağımsızdır. Fonksiyonların güçlü bir şekilde ayrımı ile sisteme esneklik ve idame edilebilirlik kazandırır (Sharma vd., 2015).

Katmanlı mimari her bir katmanda ayrı bir soyutlama sağlar, bir katmandaki değişikliğin tüm sistemi etkilememesini sağlayarak riski düşürür daha yönetilebilir birimler oluşur; her bir katmanı fiziksel olarak ayırmak da ölçeklenebilirliği, hata toleransını ve performansını artırır. Ayrıca tekrar kullanılabilirliği artırır ve test edilebilirliği kolaylaştırır (Sürme vd., 2015).

Aşağıdaki şekil, SDYA Projesinin katmanlı mimari yapısını ve Yazılım Konfigürasyon Parçaları (YKP) ile Yazılım Bileşenleri (YB) arasındaki ilişkileri göstermektedir.

ÖZGÜN SEL DOĞRULAMA YAZILIMLARI ALT YAPISI



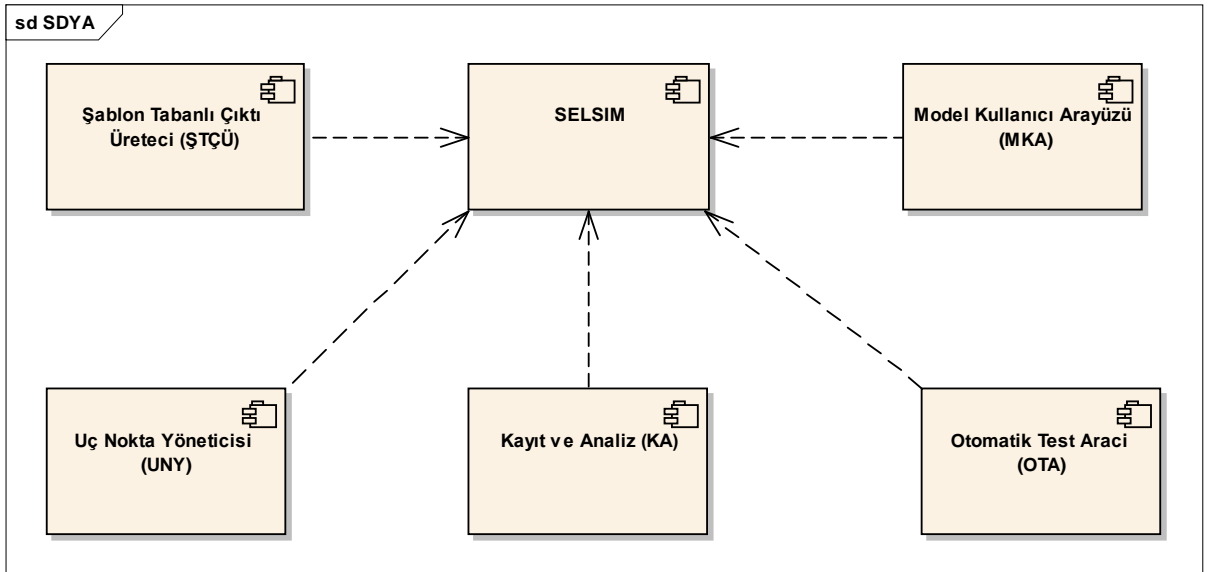
Şekil 1:

Katmanlı Mimari

2.4.2. Bileşen Tabanlı Mimari

Bileşen-tabanlı mimari, sistem tasarım ve geliştirilmesi için yazılım mühendisliği yaklaşımı tanımlar. Tasarımın, iyi tanımlı ara yüzlere sahip ayrı fonksiyonel ve mantıksal bileşenlere ayrılması üzerine odaklanır. Bu nesne-odaklı tasarım prensiplerinden de yüksek seviye soyutlama sağlar. Bileşen-tabanlı mimari ile sistemi ve diğer bileşenleri etkilemeden bir bileşen içerisinde değişiklik yapılabilir ve versiyon atlama yapılabilir, üçüncü parti bileşenler kullanarak maliyet düşürülebilir ve bileşenler başka sistemlerde tekrar kullanılabilir (Cameron vd., 2002; Berking, 2020).

Aşağıdaki şekilde SDYA projesindeki bütün Yazılım Konfigürasyon Parçaları ve birbirine olan bağımlılıkları gösterilmiştir.

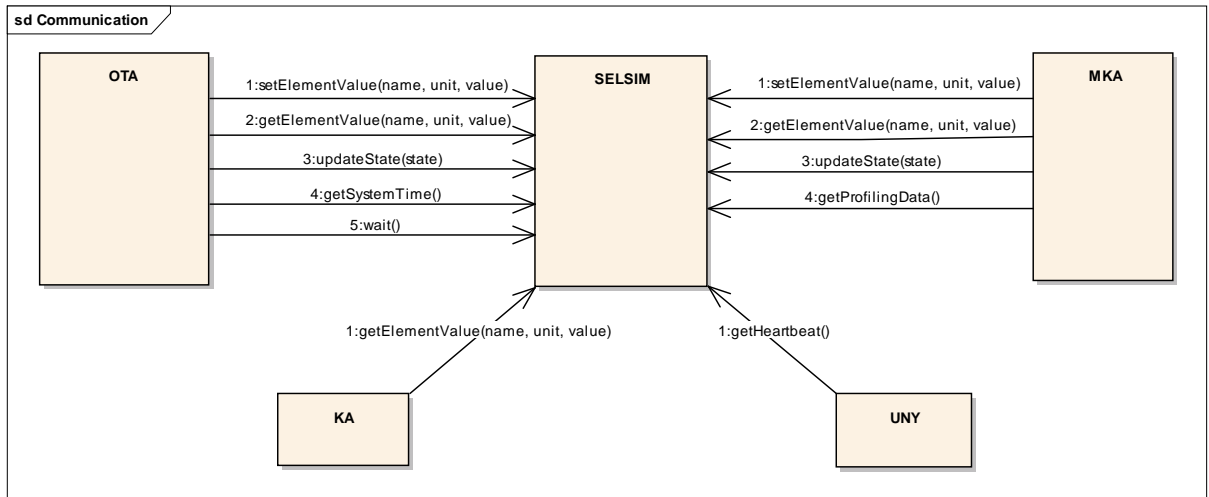


Şekil 2: Bileşen Tabanlı Mimari

2.4.3. Mesaj Veri Yolu Mimarisi

Mesaj veri yolu mimarisi, yazılım sisteminin bir ya da daha fazla iletişim kanalları kullanarak mesaj göndermesi ve alması prensibini tanımlar. Böylece uygulamalar birbirlerinin detaylarını bilmeye gerek duymadan etkileşebilir. Mesaj veri yolu, uygulamalar arası etkileşimin ortak bir veri yolu üzerinde mesaj alışverişi ile tasarlandığı mimari stildir. Bu mimarinin en yaygın uygulaması Yayınla/Abone Ol deseni seçilmiştir. Bu mimari, sistemin diğer uygulamaları etkilemeden veri yoluna yeni uygulamaları ekleme/çıkarma yolu ile genişleyebilirliğini sağlar. Ara yüz olarak sadece mesajlaşma kullanıldığı için uygulamaları geliştirme zorluğu azalır; sisteme yapılacak değişikliklerde esneklik sağlar; uygulamaların birbirlerine olan bağımlılığını azaltır ve ölçeklenebilirliği artırır (Tody, 1998).

Yazılımın bileşenleri arasındaki iletişim ve fonksiyon çağırımları gösteren şekil aşağıda verilmiştir.



Şekil 3: Mesaj Veri Yolu Mimarisi

2.5. Tasarım Metotları ve Geliştirme Detayları

SDYA Yazılım tasarımına başlarken alınan tasarım kararları şöyle sıralanabilir;

- Performans: Yazılımın performansı önemli bir kriter olduğu için geliştirilecek yazılım gerçek zamanlı olacak şekilde tasarım yönlendirilmiştir.
- Kullanılabilirlik: Yazılımın son kullanıcısının ürünü kolay öğrenebilmesi ve verimli kullanabilmesi için kullanıcı ara yüzü olabildiğince basit ve kullanışlı tasarlanmıştır.
- İdame edilebilirlik: Yazılımın farklı konfigürasyonlarda çalışabilmesi ve ileride alınacak yeni kararlara ve gereksinimlere kolayca uyarlanabilmesi için yazılım olabildiğince modüler ve konfigürasyon dosyalarına bağımlı bir şekilde tasarlanmıştır.
- Ölçeklenebilirlik: Yazılım tasarımında yayınla-abone ol (publish-subscribe) mekanizması kullanılarak sistemin ölçeklenebilir olması sağlanmıştır.

- Test Edilebilirlik: Yazılım tasarımı, kaynak kodun otomatik olarak test edilmesini sağlayacak şekilde yapılmıştır.
- Modülerlik: Modülerliği sağlamak için sistem anlamlı daha küçük parçalardan oluşacak şekilde tasarlanmıştır.

2.6. SDYA Yazılım Bileşenlerinin Açıklamaları

SDYA Yazılımının SELSIM, Model Kullanıcı Arayüz Yazılımı, Kayıt ve Analiz Yazılımı, Otomatik Test Aracı, Şablon Tabanlı Çıktı Üretici ve Uç Nokta Yöneticisi Yazılımı olmak üzere altı tane yazılım konfigürasyon parçası bulunmaktadır. SELSIM dışındaki diğer yazılım konfigürasyon parçalarının SELSIM'e bağımlılığı mevcuttur (Öztuna vd., 2017).

Her bir yazılım konfigürasyon parçası, aşağıdaki alt başlıklarda açıklanmıştır:

2.6.1. SELSIM

SELSIM Yazılımı, aviyonik cihazlara ait fonksiyonel modellerin yönetimi, veri yönetimi, I/O ara yüzlerinin yönetimi, zaman ve çerçeve (frame) yönetimi, kayıt edilen verilerin geri oynatımı, hata durumlarının raporlanması, durum/mod bilgilerinin yönetilmesi ve dağıtık/paylaşımlı ağ üzerinden veri alıp verme işlevlerini gerçekleştiren gerçek zamanlı yazılımdır (Öztuna vd., 2017).

Girdi olarak model mesaj verilerini alır ve diğer yazılım konfigürasyon parçalarının ihtiyaç duyduğu çıktıları mesajlaşma ara katmanı aracılığıyla iletir.

SELSIM'in temel fonksiyonları, veri dönüştürme ve yönetme, mesaj yönetme, model yönetme, girdi/çıkı işlemlerini yönetme, zaman ve çerçeve yönetme, loglama, dosya servisi, hafıza yönetme, görev yönetme, olay (event) yönetme olarak sayılabilir.

2.6.2. Model Kullanıcı Ara yüzü (MKA)

Model Kullanıcı Ara yüz Yazılımı, modellere ait metin veya görsel tabanlı grafik kullanıcı ara yüzlerinin oluşturulmasını sağlayan yazılımdır.

Kod üretici ile oluşturulan yapılandırma dosyalarını girdi olarak kullanır ve ara yüzler aracılığıyla oluşturulan mesajları mesajlaşma ara katmanı aracılığıyla SELSIM'e iletir.

MKA yazılımının tasarımı, Model-View-Controller (MVC) mimari deseni kullanılarak oluşturulmuştur. MKA yazılımının temel fonksiyonları, SELSIM yazılımının ihtiyaç duyduğu bütün kullanıcı ara yüzlerinin MVC yapısı gereğince görsel ara yüzünün oluşturulması, modellerinin ve görsel ara yüzün kontrolünün sağlanmasıdır.

2.6.3. Şablon Tabanlı Çıktı Üretici

Şablon Tabanlı Çıktı Üretici, veri tabanından veya grafik kullanıcı ara yüzü üzerinden verilen girdilere uygun olarak, SELSIM yazılımı, Model Kullanıcı Ara yüz yazılımı, Kayıt ve Analiz yazılımı, Otomatik Test Aracı için gerekli kod veya yapılandırma dosyalarının üretilmesini sağlayan gerçek zamanlı olmayan yazılımdır.

Girdi olarak veri tabanı verilerini ve kullanıcı ara yüzünden girilen şablon verilerini alır, bunları kullanarak diğer Yazılım Konfigürasyon Parçalarının ihtiyaç duyduğu kod ve yapılandırma dosyalarını çıktı olarak üretir.

Şablon Tabanlı Çıktı Üreticinin temel fonksiyonları veri tabanına bağlanma, veri tabanından eleman seçme, şablon belirleme, çıktı üretme olarak sıralanabilir.

2.6.4. Kayıt ve Analiz

Kayıt ve Analiz Yazılımı, dağıtık/paylaşımlı ağ üzerinden gelen bütün bilgilerinin kayıt edilebildiği ve analiz fonksiyonlarına sahip, gerçek zamanlı olmayan bir yazılımdır.

KA Yazılımının girdisi SELSIM'den Mesajlaşma Soyutlama Ara katmanı ile gelen mesajlar ve verilerdir. KA yazılımı bu mesaj ve verileri daha sonra analiz etmek ve tekrar koşturmak üzere kaydeder. Çıktısı ise kaydedilen verilerin ekranda grafik görüntüsü olarak gösterilmesidir.

KA Yazılımının temel fonksiyonları, verileri alma, kaydetme, oynatma ve analiz işlemleri için görsellik ve kontrol sağlamasıdır.

2.6.5. Uç Nokta Yöneticisi (UNY)

Uç Nokta Yöneticisi Yazılımı, üzerinde bulunduğu makinenin başlangıcından kapalı duruma geçmesine kadar sürekli çalışır. Bu sırada diğer yazılımların sağlık durumlarını kontrol eder, çalışma kiplerini ve konfigürasyonlarını dinamik olarak yönetir. Belirli işlemci veya hafıza limitlerinin üzerine çıkan uygulamaların üzerindeki mesaj yükünü hafifletmek için iş yükü dağıtımını yapılmasını sağlar.

UNY yazılımı girdi olarak SELSIM'den diğer yazılımların sağlık durumlarını ve konfigürasyonlarını içeren veriler alır. Çıktısı ise SELSIM'e yaptığı sağlık ve konfigürasyon geri bildirimleri ve model durumlarının grafik görüntüsü olarak gösterilmesidir.

UNY'nin temel fonksiyonları, üzerinde bulunduğu makinenin sağlık ve konfigürasyon bilgisi gösterme ve görev dağılımı kontrollerini gerçekleştirmektir.

2.6.6. Otomatik Test Aracı (OTA)

Otomatik Test Aracı, modele ait mesajın istenilen bir elemana değer atama ve okuma işlemlerini script dili kullanılarak yapılmasını, simülasyon sisteminin “durum” geçişlerinin yönetilmesi gibi fonksiyonları sağlayan gerçek zamanlı olmayan bir yazılım aracıdır.

OTA Yazılımı girdi olarak kullanıcının yarattığı scriptleri alır, bu scriptleri kullanarak yaratılan senaryoları koşturmak üzere SELSIM'e Mesajlaşma Soyutlama Ara katmanı aracılığı ile ilgili mesajları çıktı olarak gönderir.

OTA'nın temel fonksiyonları script yükleme, script dönüştürme, prosedür koşturma ve mesaj gönderme olarak sıralanabilir.

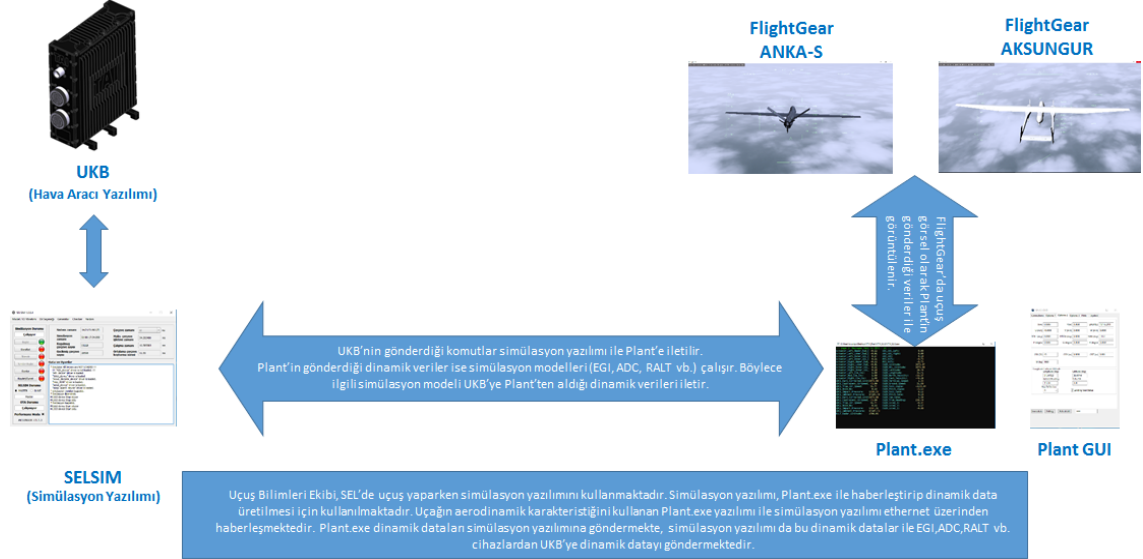
2.7. SDYA nın Kullanımı

Gerçek zamanlı bir yazılım olan UKB yazılımı, hava aracının temel uçuş fonksiyonlarını yönetir. Uçuş kritik alt sistemleri (motor, iniş takımları, aviyonik üniteler vb.) kontrol eder. Oto pilot ve uçuş yönetim sisteminin icrasını sağlar. Hata yönetim mekanizmalarını işletir. PLANT hava aracı modelleme yazılımıdır. PLANT, hava aracı dinamiğinin yerde ve havada benzetimini sağlar (Karakas vd., 2012).

UKB nin gönderdiği komutlar simülasyon aracılığıyla PLANT'a iletilir. PLANT tarafından üretilen dinamik veriler Simülasyon'da çalışan ilgili modelleri besler. Böylece ilgili modeller çeşitli hesaplamaları yaparak PLANT'ten aldığı verileri doğru bir şekilde UKB'ye iletir.

Şekil 4: Uçuş Simülasyonu için SDYA Kullanımı

UÇUŞ İÇİN SİMÜLASYON YAZILIMLARININ KULLANIMI



Testomat yazılımında test adımları oluşturulmaktadır. Testomat Simülasyon yazılımı sayesinde ilgili yazılıma mesaj gönderir ve mesaj alır. Sonuçların doğruluğunu raporlar.

DOĞRULAMA EKİBİNİN SİMÜLASYON YAZILIMI KULLANIMI



Şekil 5: Yazılım Testleri için SDYA Kullanımı

3. SONUÇ

Bu çalışma, savunma sanayinin benzersiz gereksinimlerini karşılamak üzere özel olarak İHA'lar için hazırlanmış bir simülasyon yazılımının özelliklerini incelemektedir. Çalışmada, simülasyon kullanılarak, savunma uygulamaları için İHA'ların geliştirilmesiyle ilgili zorlukların nasıl etkili bir şekilde azaltıldığına ışık tutmak ve sonuçta bu tür projelerin genel başarısına ve etkinliğine katkıda bulunmak amaçlanmıştır. Birincil amaç, geliştirilen sistemin zaman, maliyet ve doğruluk açısından verimliliğini arttırmada böyle bir simülasyonun önemini altını çizmektir. Çalışma, simülasyonun ele aldığı özel ihtiyaçlara ilişkin içgörü sağlamak ve uygulanan tasarım kararlarını açıklamaktadır.

Savunma sistemleri ve İHA teknolojileri hızla ilerlemeye devam ettikçe, simülasyonların gerçek dünya senaryolarını simüle etme ve yeni gelişmelere uyum sağlama yeteneği çok önemli hale gelmektedir. Simülasyonun tasarımına esnekliğin dahil edilmesinin önemi de yadsınamaz bir gerçektir. Simülasyonun, özellikle karmaşıklık, yüksek maliyetler, zaman kısıtlamaları ve doğal zorluklarla karakterize edilen projeler için paha biçilmez bir avantaj sunduğu ön plana çıkmıştır. Tasarım aşamasında dikkatli bir değerlendirme ve özel zaman tahsisi gerektiren kritik bir bileşen olarak simülasyonun önemi vurgulanmıştır. Ayrıca, simülasyonun yetenekleri sadece verimlilik kazanımlarının ötesine geçerek, İHA geliştirme sürecinde yer alan operatörler ve mühendisler için eğitim ve beceri geliştirmede önemli bir rol oynamaktadır. Bu ikili işlevsellik, simülasyonu yalnızca kaynak kullanımını optimize etmekle kalmayıp aynı zamanda İHA teknolojisinin inceliklerini ele alma konusunda usta bir işgücü yetiştiren çok yönlü bir araç olarak konumlandırmaktadır. Çalışma, İHA'ların süreçleriyle ilgili öğrenme zamanını azaltmadaki rolünü vurgulamakta ve böylece savunma sanayinde daha yetkin ve uyarlanabilir bir işgücünü teşvik etmektedir.

Anlatıldığı üzere; karmaşık; maliyeti yüksek; zaman kısıtı olan ve zor projeler için simülasyon; hayati bir önem taşımaktadır ve tasarımına önemli bir zaman ayrılması gereken bir yapıdır.

KAYNAKÇA

Ambler, S. (1998). A Realistic Look at Object-Oriented Reuse. [Erişim adresi: www.drdoobs.com/184415594](http://www.drdoobs.com/184415594)

Berking, P. (2020). Öğrenme Kayıt Ambarı Seçimi Kılavuzu. Z. Tuman (Ed.), Açık Mektep, Ankara. Erişim adresi: <https://ekitap.ankara.edu.tr/okas/>

Bungartz, H.J., Zimmer, S., Buchholz, M., Pflüger, D. (2014). Modeling and Simulation: An Application-Oriented Introduction, Springer Verlag Berlin Heidelberg.

Cameron, D., Clausen, C., Morton, W. (2002). Dynamic Simulators for Operator Training. In: Braunschweig, B., Gani, R. (Eds). Software Architectures and Tools for Computer Aided Process Engineering, vol 11, 1st edn. Elsevier, Amsterdam.

Cevher, Ü. (2023). İnsansız Hava Araçlarında Dayanıklılık. Avrupa Bilim ve Teknoloji Dergisi, (51), 181-189.

İçten, T., Bal, G. (2021). Askerî Alanda Artırılmış ve Sanal Gerçeklik Araçlar: Sistemler, Zorluklar ve Çözümler. Savunma Bilimleri Dergisi, 2(40), 169-199.

Karakaş, D., Karakaş, M., Karagöz, Ö., Tiftikçi, H. (2012). Türk İnsansız Hava Aracı ANKA'nın Uçuş Simülasyon Ortamı. Savunma Bilimleri Dergisi 11 (1), 91-106.

Kızak, H. (2021). Teorik Hesaplamalar İle Yeni Sentezlenen Disazo Boyarmaddelerin Yapısal ve Spektroskopik Özelliklerinin İncelenmesi. Yüksek Lisans Tezi, Pamukkale Üniversitesi, Fen Bilimleri Enstitüsü, Denizli.

Korkmaz, E., Aktaş, N., Saruha, İ., Ünal, S. (2016). Simülasyon- eduMedia. Erişim adresi: <https://peanos.wordpress.com/2016/05/10/simulasyon/>

- Landriscina, F. (2013). *Simulation and Learning A Model-Centered Approach*, Springer-Verlag New York.
- Macedonia, M. (2002). *Games Soldiers Play*. *Ieee Spectrum*, 39(3), 32-37.
- Nargül, A. S., Güçlü, K., Ülgen, H. İ., Kalebek, A., Öztuna, N. E., Sürme, U. M., Çakır, U. (2015). TUSAŞ Sistem Entegrasyon Laboratuvarı Doğrulama Yazılımları Altyapısı (SDYA): Geliştirme, Doğrulama ve Geçerli Kılma Deneyimleri. *Altıncı Ulusal Savunma Uygulamaları Modelleme ve Simülasyon Konferansı*, 11-12 Kasım 2015, Ankara.
- Öztuna, N. E., Sürme, U. M., Uğurlu, O. (2017). Tusaş Sistem Entegrasyon Laboratuvarı Doğrulama Yazılımları Alt Yapısı (SDYA). *Dokuz Eylül Üniversitesi Mühendislik Fakültesi Fen ve Mühendislik Dergisi*, 19 (55.1), 14-23.
- QT Group (2024). Erişim adresi: www.qt.io
- Sharma, A., Kumar, M., Agarwal, S. (2015). A Complete Survey on Software Architectural Styles and Patterns. *Procedia Computer Science*, 70, 16-28.
- Sürme, U.M., Öztuna, E., Uğurlu, O., Çakır, U., Nargül, A.S., Güçlü, K. (2015). SDYA: A Real Time and Distributed Software Verification Infrastructure for Validating Flight Software (OnBoard Software) at System Integration Laboratory. *ESA Simulation for European Space Programmers (SESP)*.
- Tody, D. (1998). Message Bus and Distributed Object Technology. In *Astronomical Data Analysis Software and Systems VII (Vol. 145, p. 146-149)*.
- Yao, K., Huang, S. (2021). Simulation Technology and Analysis of Military Simulation Training. In *Journal of Physics: Conference Series (Vol. 1746, No. 1, p. 012020)*. IOP Publishing.
- Yıldırım, Ö. (2019). Genetik Çeşitliliğin Nedenlerini Görselleştiren Bir Simülasyon Geliştirilmesi: Genetik Çeşitliliğin Üçlüsü. *Yüksek Lisans Tezi, Kocaeli Üniversitesi, Fen Bilimleri Enstitüsü, Kocaeli*.
- ZeroMQ (2023). Erişim adresi: www.zeromq.org