



Lityum-İyon pilin anlık gerilim, akım, güç ve sıcaklık değerlerinin Arduino üzerinden C# programlama dilinde elde edilen verilere dayalı olarak analiz edilmesi

Analyzing instantaneous voltage, current, power, and temperature values of a Lithium-Ion battery based on data obtained via Arduino in C# programming language

Sercan Taşdelen¹ , Batuhan Gökkuş² , Sezer Ulukaya^{3,*} 

^{1,2,3} Trakya Üniversitesi, Elektrik-Elektronik Mühendisliği Bölümü, 22030, Edirne Türkiye

Öz

Lityum-iyon piller, geniş bir kullanım alanına sahiptir. Elektrikli araçlar, taşınabilir elektronik cihazlar, enerji depolama sistemleri ve tıbbi cihazlar bu geniş yelpazede yer alan başlıklardan bazılarıdır. Bu tür uygulamalarda lityum-iyon pillerin güvenliğini ve performansını izlemek, kullanım ömrünü artırmak ve olası sorunları önceden tespit etmek önemlidir. Çalışmamız, bu ihtiyaca cevap vermek için 18650 3.7V lityum-iyon pil kullanılarak tasarlanmıştır. Bu makalede, lityum-iyon pillerin yük altındayken anlık voltaj, akım, güç ve sıcaklık değerlerini sensörler aracılığıyla toplayan ve Arduino mikrodenetleyici ve C# programlama dili kullanılarak görselleştirme sağlayan bir arayüz sunulmaktadır. Bu çalışmanın amacı, lityum-iyon pillerin performansını izlemek ve potansiyel aşırı yüklenme veya aşırı ısınma durumlarını tespit edebilmektir. Sensörler aracılığıyla elde edilen veriler, devrede yer alan mikrodenetleyici üzerinde işlenir ve C# programlama dili kullanılarak bilgisayar ortamına aktarılır. Bu veriler, radyal göstergeler aracılığıyla kullanıcıya görsel olarak sunulur. Ayrıca, uzun şarj ve deşarj sürelerinin kestirimi için Python programlama dili ile bağlantı modelleri kullanılmıştır. Makalenin ilerleyen bölümlerinde, çalışmanın tasarımı, kullanılan donanım ve yazılım bileşenleri ile elde edilen sonuçlar ayrıntılı bir şekilde açıklanmaktadır. Sonuçlar, lityum-iyon pillerin gerçek zamanlı izlenmesi ve güvenliğinin sağlanması için bu tür bir yaklaşımın etkinliğini göstermektedir.

Anahtar kelimeler: Lityum-iyon pil, Arduino, C#, Polinom bağlantı, GUI tasarımı

1 Giriş

Günümüzde, çevre dostu ve sürdürülebilir cihaz sistemlerine olan talep artmaktadır. Elektrikli cihazların birçoğunun güç kaynağı lityum-iyon pillerdir. Bu piller, yüksek enerji yoğunluğu, hafiflik, düşük kendiliğinden deşarj oranı ve uzun ömür gibi avantajlara sahiptir [1]. Ancak, bu pillerin güvenliği ve performansı doğru şekilde kontrol edilmediği takdirde risk oluşturabilir. Bu nedenle, pil

Abstract

Lithium-ion batteries have a wide range of usage. Electric vehicles, portable electronic devices, energy storage systems and medical devices are some of the topics in this wide range. In such applications, it is important to monitor the safety and performance of lithium-ion batteries, increase their lifespan and detect potential problems in advance. Our study was designed using 18650 3.7V lithium-ion battery to meet this need. In this article, an interface is presented that collects instantaneous voltage, current, power and temperature values of lithium-ion batteries under load through sensors and provides visualization using the Arduino microcontroller and C# programming language. The purpose of this study is to monitor the performance of lithium-ion batteries and detect potential overload or overheating situations. The data obtained through the sensors is processed on the microcontroller in the circuit and transferred to the computer environment using the C# programming language. This data is visually presented to the user through radial indicators. Additionally, regression models were used with the Python programming language to predict long charging and discharging times. In the following sections of the article, the design of the study, the hardware and software components used, and the results obtained are explained in detail. The results demonstrate the effectiveness of such an approach for real-time monitoring and ensuring the safety of lithium-ion batteries.

Keywords: Lithium-ion battery, Arduino, C#, Polynomial regression, GUI design

yönetim sistemleri, elektrikli cihaz teknolojisinin geliştirilmesi ve sürekli izlenmesi için kritik öneme sahiptir.

Bu çalışma kapsamında, pillerin anlık durumlarının izlenebilmesi için Arduino [2] mikro denetleyici ve C# programlama dili kullanılarak bir arayüz tasarlanmıştır. Benzer bir çalışmada C# ile sıcaklık, basınç, mesafe gibi temel özelliklerin verilerinin görselleştirilmesi hedeflenmektedir [3]. Bu çalışmada ise, Arduino

* Sorumlu yazar / Corresponding author, e-posta / e-mail: sezerulukaya@trakya.edu.tr (S. Ulukaya)
Geliş / Received: 09.04.2024 Kabul / Accepted: 20.08.2024 Yayımlanma / Published: 15.10.2024
doi: 10.28948/ngumuh.1467037

mikrodenetleyiciden gelen çeşitli sensör verileri, akım, sıcaklık, güç ve voltaj olmak üzere işlenmektedir. Bu veriler, grafik ve radyal görselleştirmelerle zenginleştirilerek kullanıcı arayüzüne iletilmektedir. Kullanıcılar, bu arayüz üzerinden pillerin anlık durumlarını kolayca izleyebilir, gerektiğinde müdahale edebilirler ve Structured Query Language (SQL) veri tabanına ilgili verileri kaydedebilmektedirler. Benzer bir çalışmada C# programlama dili kullanılarak konuşma bozukluklarına dair eğitim amaçlı bir platform tasarlanmıştır [4]. Bu çalışmada ise, SQL veri tabanına kaydedilmiş olan verileri Python programlama dili kullanarak, gerilim, akım ve sıcaklığa dayalı verileri işleyerek, bağlanım (regresyon) modelleri oluşturulmuştur. Bu modeller kullanılarak, pilin deşarj ve şarj olma süreleri tahmin edilmiştir. Ayrıca, çekilen verilerle grafik çizimleri gerçekleştirilerek, elde edilen sonuçlar görsel olarak da analiz edilmiştir.

Düşük maliyetli devre tasarımı hedefi nedeniyle uzun deşarj süreleriyle başa çıkabilmek için iki farklı bağlanım modeli kullanılmıştır. Bağlanım modelleri, veri kümesi üzerinde matematiksel bir ilişki kurarak, pilin voltaj değerine göre ne kadar sürede deşarj-şarj olduğunu tahmin etmek için kullanılır. Benzer bir çalışmada RC devresi üzerinde batarya sağlık ve kapasitesi kestirimi için doğrusal bağlanım kullanılmıştır [5]. Literatürde hücrelerin sağlık durumu ve kapasitesi, enerji verimliliğini artırmak, pil ömrünü uzatmak ve elektrikli araçlarda enerji tüketimini daha iyi yönetmek amacıyla tespit edilmeye çalışılmıştır [6-8]. Bir çalışmada, 4 adet li-iyon hücrenin sağlık durumu diferansiyel termal voltmetre yöntemiyle belirlenmeye çalışılmıştır [6]. Başka bir çalışmada, elektrikli araçlar için voltaj, akım ve sıcaklık ölçümü Arduino ve multimetre sonuçları karşılaştırılarak yapılmıştır [7]. Başka bir çalışmada, pil ömrü hesaplaması için hem voltaj hem de akım ölçülebilen ve Wi-Fi üzerinden internet uygulama sunucusuna gönderebilen batarya yönetim sistemi sunulmuştur [8]. Batarya yönetim sistemleri için geliştirilmesi planlanan model, pilin davranışını daha iyi anlamaya, uzun deşarj süreleriyle ve anlık değişimlerle daha etkili bir şekilde başa çıkabilmeye olanak tanımalıdır.

Bu amaçla, çevre dostu ve sürdürülebilir sistemlere yönelik talebi karşılamak adına lityum-iyon pillerin güvenliği ve performans üzerindeki kontrolü artırarak, daha verimli ve uzun ömürlü enerji depolama çözümleri sunulması ve kullanıcı dostu arayüz ile izlenebilirliğinin artırılması hedeflenmiştir.

Bu çalışmada ele alınan lityum-iyon pil kontrol sistemi, elektrikli cihazlarda kullanılan enerji depolama sistemlerinin güvenli ve etkili bir şekilde izlenmesini sağlamayı amaçlar. Sistem, ACS712 akım sensörü ve NTC sıcaklık sensörü gibi özel bileşenlerle entegre edilmiştir. Bu bileşenler, pilin güncel akım ve sıcaklık durumlarını hassas bir şekilde ölçer ve kullanıcıya gerçek zamanlı bilgi sunar.

Devrede yer alan motor, kullanıcıya pilin mevcut yük altında olduğunu belirtir ve aşırı akım çekilmesinde pilde oluşacak sıcaklık artışlarında sıcaklık sensörü aracılığıyla yapılan ölçümler, pilin aşırı ısınma durumunda motorun çalışmasını durdurur. Sıcaklık belirlenen düzeye indiğinde gerekli önlemler alındıktan sonra devreye güç vermeye devam eder. Bu durumu sağlayan anahtarlama elemanı ise

BC337 transistördür. Devrede, mikrodenetleyici üzerinden gelen dijital pin girişleri yardımı ile yüksek (high) ve düşük (low) komutları kullanılarak pildeki sıcaklık artışında pilin ömrünün kılmasını engellemek için kullanılmaktadır. Arduino mikrodenetleyicisi, bu sensörlerden gelen verileri işleyerek motoru kontrol eder ve pilin durumu hakkında görsel geribildirim sağlar. Bir çalışmada, sıcaklığın farklı anolara sahip lityum iyon pillerin kapasitesi üzerindeki etkisi araştırılmıştır [9]. Başka bir çalışmada ise transistörün anahtarlama elemanı olarak kullanıldığı Arduino kontrollü darbeleri ark kaynağı örneği incelenmiştir [10].

Son yıllarda elektrikli araç teknolojisinin gelişimi ve talebi ile batarya yönetim sistemleri ve lityum-iyon piller üzerine denetimli makine öğrenme çalışmaları da literatürde yer almaya başlamıştır [11-13]. Bir çalışmada, nesnelerin interneti çerçevesinde STM32 mikro denetleyici ile hazır bir veri kümesi üzerinde yapay sinir ağları, destek vektör makinaları ve rastgele orman yöntemleri kullanılarak batarya sağlığı tahmini üzerine performans analizi yapılmıştır [11]. Başka bir çalışmada, şarj durumu-açık devre gerilimi ilişkisi hazır bir veri tabanı üzerinde makine öğrenme algoritmaları kullanılarak irdelenmiştir [12]. Başka bir çalışmada, elektrikli bir aracın aküsünün şarj durumu tahmini için STM32 mikro denetleyici kullanarak evrişimli sinir ağı ve tekrarlayan sinir ağı karşılaştırılmıştır [13]. Hem donanım içeriğine hem de eğitim amaçlı kullanılacak arayüz tasarımına sahip düşük maliyet hedefli bu prototip çalışmasına benzer bir çalışmaya tarafımızca literatürde rastlanılmamıştır. İlgili literatür [3-13] ile karşılaştırıldığında bu çalışmanın farklı yönleri eğitim amaçlı kullanılacak arayüz geliştirilmesi, kullanılacak yük elemanlarının değiştirilebilmesi, her bir durum için farklı veri tabanı oluşturulup kaydedilebilmesi, hazır bir veri tabanına ihtiyaç duymaması, bağlanım modelleri ile kestirim yapabilmesi ve değişken sıcaklık değerlerinde analize imkân sağlaması olarak sıralanabilir.

Bu çalışmanın önemi, sadece mevcut elektrikli cihazların güvenliğini arttıracak önlemlerle sınırlı kalmayıp aynı zamanda yükün başka elektronik elemanlarla değiştirilmesi veya farklı tipte piller takılmasıyla gelecekte eğitim arayüzü olarak da kullanılacak olmasıdır. Lityum-iyon pillerin performansını artırmak ve güvenliğini sağlamak, elektrikli araçların menzilin, şarj süresini ve dayanıklılığını artırarak daha geniş bir kullanıcı kitlesine hitap edebilmesini sağlayabilir.

2 Materyal ve metod

Bu çalışmada, lityum-iyon pil kontrol sisteminin geliştirilmesi ve pil davranışının izlenmesi için belirli materyaller ve bağlanım modeli kullanılmıştır. Bu bölümde, çalışmanın tekrarlanabilirliği için kullanılan materyaller ve tercih edilen metod ayrıntılı olarak sunulmuştur.

2.1 Materyaller

Çalışmanın merkezindeki donanım olan Arduino mikrodenetleyici, ACS712 akım sensörü ve NTC sıcaklık sensöründen gelen verileri işlemek ve kontrol etmek için kullanılmıştır. Arduino, sensörlerden gelen verileri okumak, işlemek ve çeşitli çıkışları kontrol etmek için uygun bir platform sağlamıştır. Genel olarak 18650 lityum iyon pil,

3.7V nominal gerilim, 2.5V minimum gerilim, 4.2V maksimum gerilim, 2000-3500mAh kapasite, 500+ şarj döngüsü ve yüksek enerji yoğunluğuna sahiptir. Kullanılan 18650 lityum iyon pil, 3.7V nominal gerilim ve 2000mAh kapasite özelliklerine sahiptir. Lityum iyon pilin şarj işlemi, 18650 3.7V pil şarj cihazı kullanılarak gerçekleştirilmiştir.

ACS712 akım sensörü, pil akımını ölçmek için kullanılmıştır. ACS712 5A modeli, 5V çalışma gerilimi, 185 mV/A hassasiyet, $\pm 5A$ ölçüm aralığı, 5 μs tepki süresi ve analog voltaj çıkışı sunar. Analog bir sinyal üreten bu sensör, pilin anlık akım tüketimini ölçmek için devreye entegre edilmiştir. Pilin sıcaklık değerini ölçmek için NTC (Negative Temperature Coefficient) sensörü tercih edilmiştir. Bu sensör, pilin sıcaklık değişimlerini algılamak ve aşırı sıcaklık durumlarında uygun önlemleri almak için kullanılmıştır.

Arduino tarafından kontrol edilen BC337 BJT transistör, belirli eşik değerleri aşıldığında motorun kontrolünü sağlamak için kullanılmıştır. BC337 transistörünün orta bacağından çıkan dijital sinyal girişi anahtarlama olarak kullanılmış ve sıcaklık değeri ile birleştirilmiştir. Pil aşırı ısınmaya başlarsa ve sıcaklık atanan eşik değerini aşarsa, tekrar bu değerin altına düşene kadar enerji iletimi düşük (low) kademeye düşürülmektedir. Sıcaklık düştükten sonra sinyal tekrar yüksek (high) kademeye çıkarılıp enerji iletimine başlanmaktadır. Bu, pilin aşırı deşarj veya aşırı akım çekme durumlarında koruma sağlamak amacıyla tercih edilmiştir.

Pilin yük altında olduğunu görsel olarak ifade etmek ve kullanıcıya geri bildirim sağlamak amacıyla bir motor kullanılmıştır. Motor, pilin durumuyla ilgili görsel bir gösterge olarak işlev görmüştür. Devrenin elektriksel kararlılığını sağlamak ve sensörlerin doğru çalışmasını temin etmek için uygun dirençler ve kondansatörler seçilmiştir. Sensörlerden gelen verilerin işlenmesi, kontrol algoritmalarının yazılması ve görselleştirme için Arduino IDE kullanılmıştır. Bu entegre geliştirme ortamı, Arduino platformunda yazılım geliştirmeyi kolaylaştırmıştır. Veri tabanından çekilen verilerin işlenmesi ve makine öğrenmesi modellerinin oluşturulması için Python programlama dili tercih edilmiştir. Python'un geniş kütüphane desteği ve veri analitiği yetenekleri, veri işleme ve tahminleme süreçlerini kolaylaştırmıştır.

Sensörler aracılığıyla elde edilen veriler, Arduino IDE üzerinde işlenir ve C# programlama dili kullanılarak bilgisayar ortamına kaydedilir. Bu veriler, radyal göstergeler aracılığıyla kullanıcıya görsel olarak sunulur. Arayüz üzerinden toplanan veriler gerektiğinde bir SQL veri tabanına kaydedilebilmektedir. Verilerin veri tabanına aktarılmasının nedeni, lityum iyon pilin uzun deşarj ve şarj sürelerinin Python'da bağlantı modelleri kullanılarak yüksek doğruluk oranlarıyla tahmin edilmek istenmesidir.

2.2 Metodlar

2.2.1 Veri toplama ve veri işleme

ACS712 ve NTC sensörlerinden gelen veriler, Arduino mikrodenetleyicisi tarafından okunmuş, işlenmiş ve gerekli hesaplamalar yapılmıştır. Bu hesaplamalar, pilin anlık akım, güç tüketimi ve sıcaklık değerlerini belirlemek için kullanılmıştır.

2.2.2 C# Veri arayüzü tasarımı

C# ile tasarlanmış bir arayüz kullanılmıştır. Arayüz, grafiksel olarak kullanıcı dostu yönüyle, pil kontrol sisteminin izlenmesini ve yönetilmesini sağlamıştır. Arduino mikrodenetleyicisinden gelen veriler, arayüzde grafiksel olarak görselleştirilmiş ve kullanıcıya geri bildirim sağlanmıştır. Ayrıca, arayüz üzerindeki butonlar ve girdi alanları aracılığıyla kullanıcının veri girişi yapması ve belirli işlemleri gerçekleştirmesi mümkün olmuştur. Son olarak, arayüz üzerinden toplanan veriler gerektiğinde bir SQL veri tabanına kaydedilmiş ve analiz için hazır hale getirilmiştir. Bu sayede, çalışmanın tekrarlanabilirliği sağlanmıştır.

2.2.3 Görselleştirme

Arduino IDE kullanılarak elde edilen veriler, grafiksel bir arayüzde görselleştirilmiş ve kullanıcıya geri bildirim sağlanmıştır. Bu görselleştirmeler, pilin durumu hakkında hızlı ve anlaşılabilir bilgiler sunmuştur.

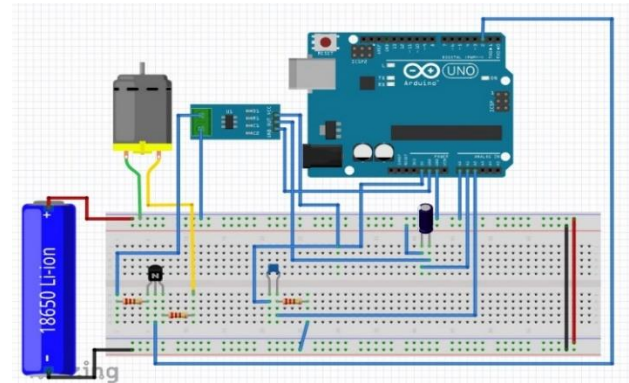
2.2.4 SQL veri tabanında işleme

Arduino IDE ile elde edilen veriler, Python programlama dili kullanılarak SQL veri tabanına kaydedilmiş ve saklanmıştır. Bu işlem, verilerin uzun vadeli saklanması ve analiz için hazır hale getirilmesini sağlamıştır.

2.2.5 Bağlanım modelleri

Python programlama dili kullanılarak SQL veri tabanından çekilen veriler üzerinde makine öğrenmesi modelleri oluşturulmuştur. Bu modeller, pilin deşarj-şarj durumunu tahmin etmek için kullanılmış ve pilin davranışının gelecekteki trendlerini belirlemeye yardımcı olmuştur. Çalışmada, doğrusal ve doğrusal olmayan bağlantı modelleri kullanılmıştır [14]. Doğrusal ve polinom bağlantı modellerinin performansları karşılaştırılmıştır. Bu modeller, 3.7V-4.2V arasında bir voltaj değeri girilince ne kadar süredir şarj olduğunu veya deşarj olduğunu saniye cinsinden kestirme amacıyla kullanıcıya yardımcı olmak için kullanılmıştır.

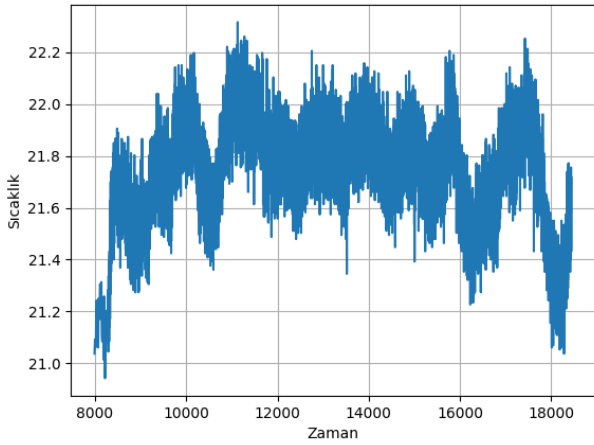
Bu şekilde, çalışmanın tekrarlanabilirliği için kullanılan materyaller ve tercih edilen metot açık bir şekilde sunulmuştur.



Şekil 1. Devrenin şematik görünümü (Fritzing kullanılarak oluşturulmuştur).

3 Bulgular ve tartışma

Veri toplama ve işleme sürecinde ACS712 akım sensörü ve NTC sıcaklık sensörü kullanılarak pilin güncel akım ve sıcaklık durumlarının belirlenmesi için Şekil 1'de gösterildiği gibi devre bağlantıları sağlanmıştır. Arduino mikrodenetleyicisi, bu sensörlerden gelen verileri işleyerek gerçek zamanlı bilgi sağlamaktadır. Elde edilen değerler, Arduino IDE kullanılarak veri tabanına iletilen verilerin Python ile grafiksel olarak ifade edilmesi ile görselleştirilmiştir. Ayrıca, pilin mevcut yük altında olduğunu belirten motor ve aşırı ısınma durumunda kullanıcıya uyarı veren sıcaklık sensörü, pil yönetim sisteminin etkinliğini artırmıştır. Geliştirilen kontrol sistemi, lityum-iyon pillerin güvenli ve verimli kullanımını desteklemek amacıyla prototip devre olarak temsil edilmiştir. Pilin gerçek zamanlı izlenebilmesi, enerji verimliliğini artırmak ve pil ömrünü uzatmak için kritik bir rol oynamaktadır.



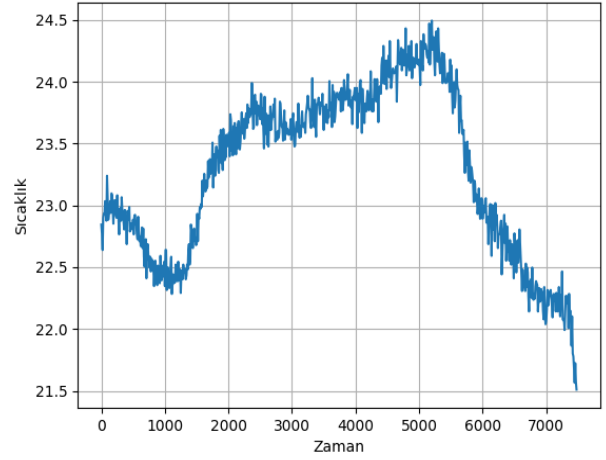
Şekil 2. Şarj işlemi sırasında pilin anlık sıcaklık değişimi grafiği

Grafiklere bakılarak, pillerin şarj ve deşarj durumlarında sıcaklık değişimleri gözlemlenebilmektedir. Deşarj ve şarj durumlarında sıcaklıkların değişkenlik gösterdiği görülmektedir. Ayrıca, şarj ve deşarj durumlarında pillerin gerilimlerinin polinom bağlanım modeli ile tahmin edilen değerleri ilgili grafiklerden izlenmektedir.

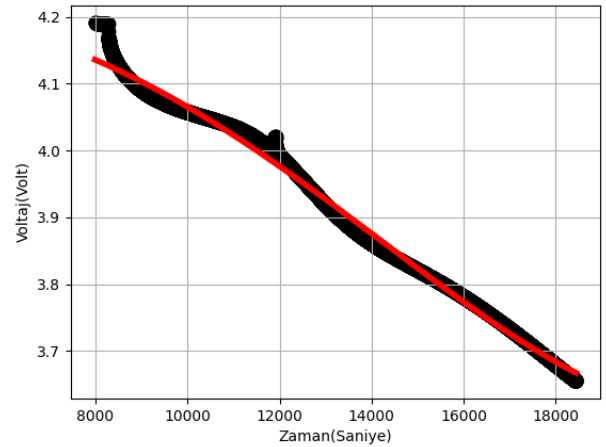
Şekil 2 ve Şekil 3'te görülen grafikte, pilin şarj ve deşarj olur iken üzerindeki motor yükü vasıtasıyla pilden çekilen akımın, pilde oluşturduğu sıcaklığın zamanla nasıl arttığı gözlemlenebilmektedir. Bu veriler veri tabanından elde edilmiş ve ardından Python koduyla bu veriler alınarak Matplotlib kütüphanesiyle görselleştirilmiştir [15]. X eksenini zamanı, Y eksenini ise sıcaklığı temsil etmektedir. Şekil 3'te yük çekilmeye başladığı anda sıcaklığın hızlı bir şekilde yükseldiği gözlemlenmiştir.

Şekil 4 ve Şekil 5'te, Arduino üzerinden çekilen verilerin C# yardımıyla SQL'e kaydedildiği ve ardından makine öğrenmesi kullanılarak oluşturulan bir polinom regresyon modeli ile zaman ve voltaja bağlı değişimlerin gözlemlendiği bir tahmin görseli bulunmaktadır. Bu model, R² skoru ile tahminlerde bulunmayı sağlamaktadır [16]. Bu sayede,

belirli bir voltajın girilmesi durumunda pilin ne kadar sürede şarj veya deşarj olacağı tahmin edilebilmektedir.



Şekil 3. Deşarj işlemi sırasında pilin anlık sıcaklık değişimi grafiği

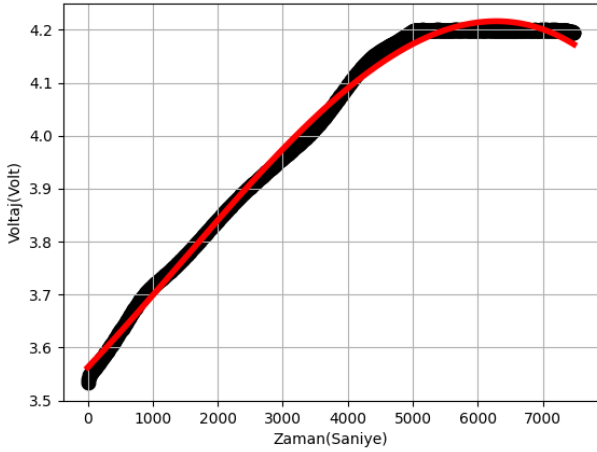


Şekil 4. Deşarj durumunda polinom bağlanım modeli tahmini (kırmızı renkli)

Bu yazılımın sunduğu avantajlardan biri de pillerin uzun sürede şarj olması nedeniyle, pilin şarj veya deşarj olma süresini tahmin edebilecek bir modele sahip olmasıdır. Yapılan deneyler sonucu doğrusal bağlanım modeli ile deşarj süresini doğru tahmin etme oranı % 99.21 iken polinom bağlanım modeli ile % 99.26 doğru tahmin etme oranına ulaşılmıştır.

C# dilinde geliştirilen arayüz, Arduino ile etkileşim kurarak çeşitli ölçümleri görsel olarak sunmaktadır. Bu arayüz, Arduino'dan alınan sıcaklık, voltaj, akım ve güç değerlerini kullanarak bir dizi görsel öğe ile kullanıcıya sunmaktadır. Şekil 6'da tasarlanan arayüz bulunmaktadır.

Arayüzde dört adet radyal gösterge bulunmaktadır. Birinci radyal gösterge, pilin anlık gerilimini, ikincisi anlık akımını, üçüncüsü anlık gücünü ve dördüncüsü de sıcaklığı göstermektedir. Bu göstergeler, kullanıcının belirli bir anda ölçülen değerleri kolayca takip edebilmesini sağlamaktadır.

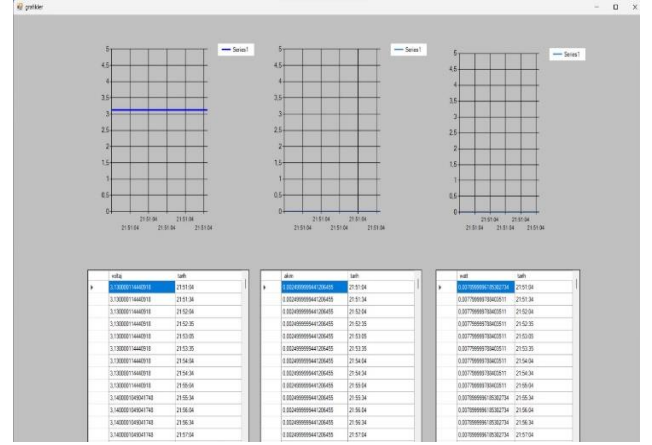


Şekil 5. Şarj durumunda polinom bağlanım modeli tahmini (kırmızı renkli)

Ayrıca, arayüzde iki farklı grafik bulunmaktadır. Birinci grafikte mavi renkle ortam sıcaklığı, sarı renkle güç değerleri gösterilmektedir. İkinci grafikte ise kırmızı renkle akım, mavi renkle voltaj değerleri görselleştirilmektedir. Bu grafikler, ölçülen değerlerin zaman içindeki değişimini takip etmek için kullanıcıya bir gösterim sunmaktadır.

Arayüz aynı zamanda kullanıcıya Arduino'nun bağlı olduğu portu seçme ve iletişim hızını belirleme imkânı da sunmaktadır. İki ayrı buton aracılığıyla bağlantıyı açıp kapatma işlemleri gerçekleştirilebilmektedir. Bu özellik, kullanıcının Arduino ile iletişimi başlatıp sonlandırmasına olanak tanır.

Kayıtlı verileri görüntüle kısmına tıklanıldığında Şekil 7 'de yer alan arayüz açılır, kullanıcının istediği süre boyunca veri ekleyip durdurma fonksiyonu bulunmaktadır.

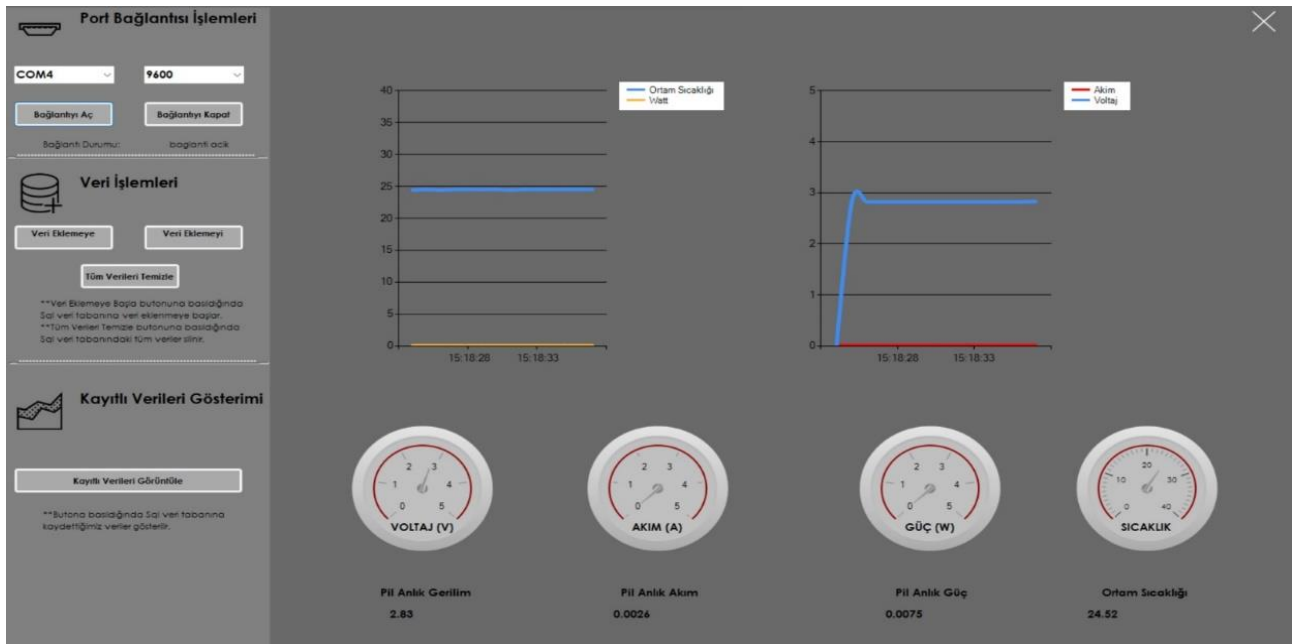


Şekil 7. Kaydedilen verilerin istenilen zaman aralığına göre görselleştirilmesine yönelik arayüz

4 Sonuçlar

Bu çalışmanın sonuçları, geliştirilen lityum-iyon pil kontrol sistemi ile ilgili önemli bulgular ortaya koymaktadır. Çalışmada kullanılan ACS712 akım sensörü ve NTC sıcaklık sensörü sayesinde pilin akım ve sıcaklık durumları hassas bir şekilde ölçülerek gerçek zamanlı olarak Şekil 6 'da görüldüğü gibi izlenmiştir. Ayrıca, BC337 BJT transistörü kullanılarak belirlenen eşik değerleri aşıldığında motorun kontrolü sağlanarak kullanıcıya uyarı sunulmuştur.

Arduino mikrodenetleyicisi, sensörlerden gelen verileri işleyerek pilin durumu hakkında kullanıcıya bilgi sunarken, C# programlama dili kullanılarak geliştirilen arayüz (Şekil 6) sayesinde bu veriler görsel olarak izlenebilir hale getirilmiştir. Arduino IDE ve Python programlama dili aracılığıyla alınan veriler, SQL veri tabanına işlenmiş ve ardından bağlanım modelleri kullanılarak pilin deşarj-şarj olma durumu tahmin edilmiştir.



Şekil 6. Verileri görselleştiren ve daha sonra kullanılmak üzere kaydedebilen tasarlanan arayüz

Tablo 1’de pilin gerilim değerine bağlı ne kadar süredir (saniye) deşarj olduğu ilk 5 satırda gösterilmiştir. Son satırda ise pilin anlık gerilim değeri 3.7V iken ne kadar süredir (saniye) deşarj olduğu tahmin edilmiştir.

Tablo 1. Pilin gerilim değerine bağlı olarak deşarj süresi tahmini

No	Gerilim (V)	Zaman (s)
1	3.693510	17742.416843
2	3.693353	17743.416843
3	3.693274	17744.416843
4	3.693204	17745.416843
5	3.693133	17746.416843
Tahmin	3.7	17597.349420

Ayrıca, Python’da matplotlib kütüphanesi ile deşarj-şarj işlemlerinde kayıtlı verilerden sıcaklık değerleri, Şekil 2 ve Şekil 3’te görüldüğü gibi grafik haline getirilmiştir. Kayıtlı olan tüm veriler Şekil 7’de grafikler ile görselleştirilmiştir. Bu özellik, kullanıcının veri alımını kontrol etmesine ve istediği zaman veri akışını durdurmasına olanak tanır. Ayrıca, tüm verileri temizle butonu aracılığıyla SQL veri tabanındaki tüm verilerin temizlenmesi sağlanmaktadır. Bu özellik de kullanıcının veri tabanındaki verileri gerektiğinde temizleme ve yeni deney/süreç başlatma imkânı için hafızada yer açılmasına olanak sağlar.

Elde edilen sonuçlar, geliştirilen kontrol sisteminin lityum-iyon pillerin güvenli ve verimli kullanımını desteklemede etkili olduğunu göstermektedir. Sistemin gerçek zamanlı izleme, kullanıcı uyarıları ve güç tahmin modelleri sayesinde pil yönetimi daha etkin hale getirilmiştir. Son yıllarda ön plana çıkmaya başlayan elektrikli araç endüstrisi başta olmak üzere enerji depolama sistemleri için çözümler kritik bir öneme sahiptir.

Çalışmanın çok yönlü tasarımı, lityum-iyon pil kontrol sistemini sadece elektrikli araçlar için değil, aynı zamanda taşınabilir elektronik cihazlar, enerji depolama sistemleri ve yenilenebilir enerji çözümleri gibi farklı sektörlerde de kullanılabilir hale getirmeyi amaçlamıştır. Bu özellik, çalışmanın sadece mevcut teknolojiyi iyileştirmekle kalmayıp aynı zamanda gelecekteki enerji teknolojilerine de katkı sağlayabileceğini göstermektedir.

Sonuç olarak, geliştirilen lityum-iyon pil kontrol sistemi, çevre dostu ve sürdürülebilir ulaşım sistemlerine yönelik talebi karşılamak adına önemli bir prototipi temsil etmektedir. Gelecekte, lityum-iyon pillerin güvenli ve etkili bir şekilde kullanılmasına katkı sağlamakla kalmayıp aynı zamanda bu alandaki araştırmalarda yük elemanının değiştirildiği senaryolara göre eğitim amaçlı bir arayüz olarak da kullanılabilmesi öngörülmektedir.

Çıkar çatışması

Yazarlar çıkar çatışması olmadığını beyan etmektedir.

Benzerlik oranı (iThenticate): %8

Kaynaklar

- [1] B. D. Polat ve Ö. Keleş, Lityum iyon pil teknolojisi. Metalurji Dergisi, 162, 42-48, 2012. https://metalurji.org.tr/dergi/dergi162/d162_4248.pdf.
- [2] S. A. Arduino, Arduino. Arduino LLC, 372. <https://search.iczhiku.com/paper/TFzDJhGhd6VMaDsI.pdf>, Accessed 08 April 2024.
- [3] V. Çavuş, T. Resul, İ. U. Duran, Arduino devreleri için kod üretme ve veri işleme uygulaması tasarımı, Muş Alparslan Üniversitesi Fen Bilimleri Dergisi, 5(1), 387-390, 2017. <https://doi.org/10.18586/msufbd.322388>.
- [4] A. K. Ayça, A. Sarıkas, A. Yayla, Konuşma sesi bozukluklarının düzeltilmesine yönelik eğitim platformu tasarımı, Bilişim Teknolojileri Dergisi, 10(3), 241-246, 2017. <https://doi.org/10.17671/gazibtd.330867>.
- [5] T. Kim, Y. Wang, H. Fang, Z. Şahinoğlu, T. Wada, S. Hara, and W. Qiao, Model-based condition monitoring for lithium-ion batteries. Journal of Power Sources, 295, 16-27, 2015. <https://doi.org/10.1016/j.jpowsour.2015.03.184>.
- [6] Y. Merla, B. Wu, V. Yufit, N. P. Brandon, R. F. Martinez-Botas, and G. J. Offer, Extending battery life: A low-cost practical diagnostic technique for lithium-ion batteries. Journal of Power Sources, 331, 224-231, 2016. <https://doi.org/10.1016/j.jpowsour.2016.09.008>.
- [7] Y. Calvinus, F. Wiryanata, and H. Tanujaya, Battery charging management system design with voltage, current and temperature monitoring features in electric vehicles. In IOP Conference Series: Materials Science and Engineering, 1007, 1, 012174, 2020. <https://doi.org/10.1088/1757-899X/1007/1/012174>.
- [8] D. Rimpas, V. A. Orfanos, P. Chalkiadakis, and I. Christakis, Design and development of a low-cost and compact real-time monitoring tool for battery life calculation. Engineering Proceedings, 58(1), 17, 2023. <https://doi.org/10.3390/ecsa-10-16146>.
- [9] S. Lv, X. Wang, W. Lu, J. Zhang, and H. Ni, The influence of temperature on the capacity of lithium ion batteries with different anodes. Energies, 15(1), 60, 2022. <https://doi.org/10.3390/en15010060>.
- [10] N. Y. Krampit, T. S. Kust, and M. A. Krampit, Specifics of pulsed arc welding power supply performance based on a transistor switch. In IOP Conference Series: Materials Science and Engineering, 142, 1, 012013, 2016. <https://doi.org/10.1088/1757-899X/142/1/012013>.
- [11] G. Crocioni, D. Pau, J. M. Delorme, and G. Grusso, Li-ion batteries parameter estimation with tiny neural networks embedded on intelligent IoT microcontrollers. IEEE Access, 8, 122135-122146, 2020. <https://doi.org/10.1109/ACCESS.2020.3007046>.
- [12] S. S. S. Narayanan and S. Thangavel, Machine learning-based model development for battery state of charge–open circuit voltage relationship using regression techniques. Journal of Energy Storage, 49,

- 104098, 2022.
<https://doi.org/10.1016/j.est.2022.104098>.
- [13] Y. Mazzi, H. Ben Sassi, A. Gaga, and F. Errahimi, State of charge estimation of an electric vehicle's battery using tiny neural network embedded on small microcontroller units. *International Journal of Energy Research*, 46(6), 8102-8119, 2022.
<https://doi.org/10.1002/er.7713>.
- [14] E. Ostertagová, Modelling using polynomial regression. *Procedia Engineering*, 48, 500-506, 2012.
<https://doi.org/10.1016/j.proeng.2012.09.545>.
- [15] J. D. Hunter, Matplotlib: A 2D Graphics Environment. *Computing in Science & Engineering*, 9, 3, 90-95, 2007. <https://doi.org/10.1109/MCSE.2007.55>.
- [16] D. Gómez, P. Salvador, J. Sanz, and J. L. Casanova, Potato yield prediction using machine learning techniques and sentinel 2 data. *Remote Sensing*, 11(15), 1745, 2019.
<https://doi.org/10.3390/rs11151745>.

