



TARSUS
ÜNİVERSİTESİ

T.C.
TARSUS ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ

İMALAT MÜHENDİSLİĞİ ANABİLİM DALI

BİLGİSAYAR KONTROLLÜ ÇOK FONKSİYONLU AKIM KAYNAĞI
TASARIMI VE UYGULAMALARI

Gülçin MÜHÜRÇÜ

DOKTORA TEZİ

TARSUS - 2021

T.C.
TARSUS ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ
İMALAT MÜHENDİSLİĞİ ANABİLİM DALI

BİLGİSAYAR KONTROLLÜ ÇOK FONKSİYONLU AKIM KAYNAĞI
TASARIMI VE UYGULAMALARI

Gülçin MÜHÜRÇÜ

Danışman: Prof. Dr. Mustafa Kemal KÜLEKÇİ

DOKTORA TEZİ

TARSUS - 2021

Tarsus Üniversitesi Lisansüstü Eğitim Enstitüsü Müdürlüğüne;

Bu çalışma, jürimiz tarafından İmalat Mühendisliği Anabilim Dalında DOKTORA TEZİ olarak kabul edilmiştir.

Başkan: Prof. Dr. Mustafa Kemal KÜLEKÇİ
(Danışman)

Üye: Prof. Dr. Uğur EŞME

Üye: Prof. Dr. Berdan ÖZKURT

Üye: Prof. Dr. Ali KESKİN

Üye: Doç. Dr. Ahmet ÇAKAN

Yukarıdaki Jüri kararı Lisansüstü Eğitim Enstitüsü Yönetim Kurulu'nun/... /2021 tarih ve/..... sayılı kararıyla onaylanmıştır.

ONAY

Yukarıdaki imzaların, adı geçen öğretim elemanlarına ait olduklarını onaylarım.
.../.../2021

Prof. Dr. Osman Murat ÖZKENDİR
Enstitü Müdürü

NOT: Bu tezde kullanılan ve başka kaynaktan yapılan bildirişlerin, çizelge, şekil ve fotoğrafların kaynak gösterilmeden kullanımı, 5846 sayılı Fikir ve Sanat Eserleri Kanunu'ndaki hükümlere tabidir.

ETİK BEYANI

Tarsus Üniversitesi Lisansüstü Eğitim Enstitüsü Tez Yazım Kurallarına uygun olarak hazırladığım bu tez çalışmada;

- Tez içinde sunduğum verileri, bilgileri ve dokümanları akademik ve etik kurallar çerçevesinde elde ettiğimi,
- Tüm bilgi, belge, değerlendirme ve sonuçları bilimsel etik ve ahlak kurallarına uygun olarak sunduğumu,
- Tez çalışmada yararlandığım eserlerin tümüne uygun atıfta bulunarak kaynak gösterdiğimi,
- Kullanılan verilerde ve ortaya çıkan sonuçlarda herhangi bir değişiklik yapmadığımı,
- Bu tezde sunduğum çalışmanın özgün olduğunu,

bildirir, aksi bir durumda aleyhime doğabilecek tüm hak kayıplarını kabullendiğimi beyan ederim. 25/01/2021

İMZA

Gülçin MÜHÜRÇÜ

ÖZET

BİLGİSAYAR KONTROLLÜ ÇOK FONKSİYONLU AKIM KAYNAĞI TASARIMI VE UYGULAMALARI

Gülçin MÜHÜRÇÜ

Doktora Tezi, İmalat Mühendisliği Anabilim Dalı

Danışman: Prof. Dr. Mustafa Kemal KÜLEKÇİ

Ocak 2021, 317 sayfa

Elektrodepozisyon tekniği, uzun yıllardır metalik kaplamalar üretmek için kullanılan en ekonomik yöntemlerden biridir. Elektrodepozisyon yöntemi ile metal kaplama proseslerinde temel hedef, aşınma, korozyon, yüksek sıcaklık dayanımı, sertlik gibi konularda malzeme yüzeyleri için daha iyi sonuçlar elde etmektir. Elektrodepozisyon yöntemi ile metal kaplamaların üretimi için yaygın olarak Doğru Akım (DC) kullanılır. Son yıllarda yapılan çalışmalarda, DC elektrodepozisyon yöntemine ek olarak kaplama işlemleri için Pulse Akım (PC) ve Pulse Reverse Akım (PRC) elektrodepozisyon yöntemlerini kullanmanın ve akım dalga formundaki değişikliklerin kaplama sonuçlarını olumlu yönde etkileyebileceği gösterilmiştir.

Bu tez çalışmasında, elektrodepozisyon yöntemi ile metal kaplama işlemini gerçekleştirebilmek amacı ile mühendislik alanında kullanılan temel sinyalleri akım dalga formlarına dönüştürebilen yeni bir akım kaynağı (akım üretici) sistemi tasarlanmıştır. Akım üretici sistemi, güç elektroniği devre topolojilerinden olan Buck Converter ve H-Bridge kullanılarak gerçekleştirilmiştir. Öngörülen akım dalga biçimlerini üretmek için, doğrusal olmayan bir algoritma olan Yapay Sinir Ağı (YSA), hibrit güç elektroniği sistemine uyarlanmıştır. Üretmiş olduğumuz yeni nesil akım üretici sistemi kullanılarak elektrodepozisyon yöntemi ile maksimum akım (Imax) koşulu için kaplama deneyleri gerçekleştirilmiştir. Kaplama işleminde katot olarak Bakır (Cu) ve anot olarak Nikel (Ni) kullanılmıştır. Kaplama deneyleri, her akım formu için 3 farklı frekansla tekrarlanmıştır. Kaplama sonrasında, kaplanan alanların yüzey morfolojisi ve kenar kaplama başarısı incelenmiş ve DC, PC ve PRC yöntemleri ile karşılaştırılarak sonuçlar tartışılmıştır.

Anahtar kelimeler: Akım kontrolü, Elektrodepozisyon, Kaplama, Buck Dönüştürücü, Yapay Sinir Ağı.

ABSTRACT

COMPUTER CONTROLLED MULTIFUNCTIONAL CURRENT SOURCE DESIGN AND APPLICATIONS

Gülçin MÜHÜRÇÜ

Ph.D. Thesis, Department of Manufacturing Engineering

Supervisor: Prof. Dr. Mustafa Kemal KÜLEKÇİ

January 2021, 317 pages

Electrodeposition technique is one of the most economical methods used to produce metallic coatings for many years. The main goal in metal coating processes with the electrodeposition method is to achieve better results for material surfaces in matters such as wear, corrosion, high temperature resistance, and hardness. Direct Current (DC) is commonly used for the production of metal coatings by electrodeposition method. Recent studies have shown that in addition to the DC electrodeposition method, using Pulse Current (PC) and Pulse Reverse Current (PRC) electrodeposition methods and changes in the current waveform can positively affect the coating results.

In this thesis, a new current source (current generator) system that can convert basic signals used in engineering to current waveforms for metal plating with electrodeposition method is designed. The current generator system has been implemented using power electronics circuit topologies, Buck Converter and H-Bridge. Artificial Neural Network (ANN), a non-linear algorithm, has been adapted to the hybrid power electronics system to generate predicted current waveforms. Later, coating experiments were carried out with electrodeposition method using a new generation current generator system. Copper (Cu) is used as cathode and Nickel (Ni) as anode in the coating process. Coating tests have been carried out for constant maximum current (I_{max}) conditions. The experiments were repeated with 3 different frequencies for each coating current signal. After coating, surface morphology and edge coating success of the coated areas were examined and the results were discussed by comparing with DC, PC and PRC methods.

Keywords: Current control, Electrodeposition, Coating, Buck Converter, Artificial Neural Network.

ÖNSÖZ

Doktora çalışmamda desteklerini benden esirgemeyen, akademik ve sosyal anlamda her türlü bilgisini paylaşan değerli tez danışman hocam Prof. Dr. Mustafa Kemal KÜLEKÇİ'ye en içten teşekkürlerimi sunarım. Teknoloji Fakültesi öğretim üyelerine çalışmaya sağladıkları katkılardan dolayı teşekkür ederim.

Çalışmam esnasında bana maddi ve manevi destek sağlayan aileme sonsuz teşekkürlerimi sunarım.



İÇİNDEKİLER

	Sayfa
ÖZET.....	iv
ABSTRACT.....	v
ÖNSÖZ.....	vi
KISALTMALAR.....	x
TABLolar LİSTESİ.....	xi
EKLER LİSTESİ.....	xii

BÖLÜM I

GİRİŞ

1.1. Problem.....	1
1.2. Araştırmanın Amacı.....	1
1.3. Araştırmanın Önemi.....	2
1.4. Sınırlılıklar.....	2

BÖLÜM II

KAYNAK ARAŞTIRMALARI

2.1. Yüzey Kaplama Yöntemleri.....	3
2.2. Metalik Kaplamalar.....	3
2.3. Elektrolitik Kaplama (Elektrodepozisyon).....	4
2.3.1. Doğru akım (DC) elektrodepozisyon yöntemi.....	4
2.3.2. Darbeli akım (PC) ve darbeli ters akım (PRC) elektrodepozisyon yöntemi.....	6
2.4. Elektrolitik Kaplamaya Etki Eden Faktörler.....	9
2.4.1. Akım yoğunluğu.....	9
2.4.2. İşlem süresi (Duty Cycle).....	10
2.4.3. Darbe frekansı.....	11
2.5. Elektrodepozisyon Yöntemi İle Nikel (Ni) Kaplamalar.....	13

BÖLÜM III

MATERYAL VE YÖNTEM

3.1. Buck Dönüştürücü.....	15
3.1.1. Buck dönüştürücü çalışma prensibi.....	16

3.1.1.1. Anahtar.....	17
3.1.1.2. Bobin.....	18
3.1.1.3. Diyot.....	18
3.1.1.4. Kondansatör.....	19
3.1.2. Buck dönüştürücü çalışma modları.....	20
3.1.3. Devre elemanlarının hesabı.....	21
3.1.4. Anahtarlama mosFET'i.....	26
3.1.4.1. Mosfet çıkış kondansatörünün (Coss) etkisi.....	27
3.1.5. Buck çıkış kondansatörü.....	27
3.1.5.1. Kapasitans filtre hesabı 1.....	28
3.1.5.2. Kapasitans filtre hesabı 2.....	28
3.1.5.3. Sonuç.....	30
3.1.6. Giriş kapasitansı.....	30
3.1.7. Buck dönüştürücü benzetim ortamı çalışmaları.....	31
3.2. Yapay Sinir Ağı ve Temel Nöron Dinamiği.....	38
3.2.1. Nöron fonksiyonları.....	40
3.2.1.1. Lojistik (Logistic, Sigmoid).....	41
3.2.1.2. Hiperbolik -Tanjant (Hyperbolic Tangent).....	42
3.2.1.3. Eşik (Treshold).....	43
3.2.1.4. Üstel dağılım (exponential distrubution).....	43
3.2.1.5. Oransal (ratio polynomial).....	44
3.2.2. YSA ağırlıklarının optimizasyonu.....	45
3.2.2.1. YSA ileri yön hesaplama (İleri Yayılım).....	46
3.2.2.2. Geri yayılım (YSA eğitimi).....	47
3.2.2.3. Bias.....	49
3.2.2.4. Momentum.....	51
3.3. Akım Kaynağı Elektronik Devre Yapısı.....	51
3.3.1. Güç elektroniği devre topolojisi.....	52
3.3.2. Gömülü sistem.....	58
3.4. Kontrol Algoritması.....	59
3.5. Devre Tasarımı.....	61
3.6. Bilgisayar Yazılımı.....	64
3.6.1. Step.....	66
3.6.2. Quarter sin.....	67
3.6.3. Half sin.....	68

3.6.4. Quarter cos	70
3.6.5. Half cos	71
3.6.6. Ramp	72
3.6.7. 1. Derece sistem cevabı.....	73
3.6.8. 2. Derece sistem cevabı.....	75
3.6.9. Lorenz	76
3.7. Kaplama Deney Düzenegi.....	79
3.7.1. Kaplama banyosu.....	81
3.8. Yöntem.....	81
3.8.1. Deney 1 için uygulanan akım türleri.....	81
3.8.2. Deney 2 için uygulanan akım türleri.....	82

BÖLÜM IV

BULGULAR

4.1. Kaplama Deney Sonuçları	84
4.1.1. Deney 1 taramalı elektron mikroskobu (SEM) analizi	84
4.1.2. Deney 2 taramalı elektron mikroskobu (SEM) analizi	86

BÖLÜM V

SONUÇLAR VE ÖNERİLER

5.1. Sonuçlar	90
5.2. Öneriler	91

KAYNAKÇA	92
-----------------------	-----------

EKLER.....	99
-------------------	-----------

ÖZGEÇMİŞ.....	316
----------------------	------------

KISALTMALAR

DC	:Dođru akım
PC	:Darbeli akım
PRC	:Darbeli ters akım
Ton	:Darbe açık kalma süresi
Toff	:Darbe kapalı kalma süresi
Vk	:Kaynak gerilimi
PWM	:Darbe genişlik modülasyonu -Pulse Widht Modulation
YG	:Yüksek gerilim
CCM	:Sürekli iletim modu-Continuous conduction mode
DCM	:Süreksiz iletim modu -Discontinuous conduction mode
ESR	:Eşdeđer seri direnç- Equivalent series resistance
AC	:Alternatif akım
SMPS	:Anahtarlamalı güç kaynađı- Switch mode power supply
YSA	:Yapay Sinir Ađı
Ni	:Nikel
A	:Amper
mA	:Miliamper
R	:Direnç
I_{max}	:Maksimum akım
μF	:Mikro farad
ms	:milisaniye
SEM	:Taramalı elektron mikroskopu-Scanning electron microscope
R_{load}	:Elektrolit Direnci
min	:Minimum
max	:Maksimum

TABLULAR LİSTESİ

	Sayfa
Tablo 3.1. Buck dönüştürücü parametreleri.....	34
Tablo 3.2. X.YZN harflerinin temsil ettiği akım aralığı.....	53
Tablo 3.3. Elektrolitik nikel kaplamalara ait banyo kompozisyonu.....	81



EKLER LİSTESİ

	Sayfa
EK 1. Bilgisayar yazılımı	99
EK 2. Mikroişlemci yazılımı.....	290



1. GİRİŞ

1.1. Problem

Malzemeler, mühendislik teknolojisinde yapı taşı görevini görmektedir. Çalışma koşullarına ve ortam şartlarına bağlı olarak malzemelerdeki etkileşim, ilk önce malzeme yüzeylerinde gerçekleşir. Bu nedenle malzeme özelliklerini iyileştirmek için kullanılan yüzey kaplama teknikleri yıllar boyunca mühendisliğin önemli araştırma konularından biri haline gelmiştir (Nyambura vd., 2019; Xu vd., 2020; Bhushan ve Gupta, 1991; Awang vd., 2020; Shibe ve Chawla, 2013; Mbugua vd., 2020; Fotovvati vd. 2019). Yüzey özelliklerini iyileştirmek için kullanılan tekniklerden bazıları Fiziksel Buhar Biriktirme (PVD), Kimyasal Buhar Biriktirme (CVD), Mikro Ark Oksidasyonu (MAO), Sol Jel, İyon İmplantasyonu, Termal Sprey, Lazer ile Klamlama ve Elektrodepozisyon olarak verilebilir (Fotovvati vd. 2019; Sidky ve Hocking, 1993; Mahdavi ve Allahkaram, 2015). Son yıllardaki teknolojik gelişmelere uygun, üstün özellikli ve ekonomik malzeme elde etmek için uygulanan yüzey kaplama tekniklerinden biri, elektrodepozisyon yöntemi ile yapılan metalik kaplamadır (Borkar ve Harimkar, 2011; Chung ve Chang, 2015).

Elektrodepozisyon yöntemi, metalik kaplamalar üretmek için kullanılan elektrokimyasal bir işlemdir. Elektrodepozisyon yöntemi ile metal kaplama proseslerinde temel hedef, aşınma, korozyon, yüksek sıcaklıklara dayanım ve sertlik gibi konularda daha iyi çözümler elde etmektir (Spanou ve Pavlatou, 2010; Srivastava vd., 2007).

1.2. Araştırmanın Amacı

Elektrodepozisyon yönteminde, metalik kaplamaların üretimi için yaygın olarak DC kullanılır. Son yıllarda DC'ye ek olarak kaplama işlemleri için PC ve PRC elektrodepozisyon yöntemleri kullanılmaktadır. PC veya PRC dalga formları kullanılarak yapılan elektrodepozisyon yoluyla kaplama işlemlerinde, akımın darbe (pulse) genişliğini değiştirmenin kaplama sonuçlarını olumlu yönde etkileyebileceği gösterilmiştir. Aynı zamanda PC ve PRC elektrodepozisyon kaplama yöntemlerinin, DC kaplama yöntemine göre daha başarılı oldukları, çeşitli araştırmalarda ortaya

konulmuştur (Chandrasekar ve Pushpavanam, 2007; Chang vd., 2006; Gyftou vd., 2008; Chang vd., 2008; Reddy vd., 2016).

Bu çalışmada, bu üç elektrik akımı dalga formuna ek olarak, farklı dalga formları ile yapılan elektrolitik kaplamaların, kaplama sonucuna olan etkileri araştırılmıştır. Böylece daha iyi yüzey özelliklerine sahip kaplamalar elde edilmesi hedeflenmiştir.

1.3. Araştırmanın Önemi

Elektrolitik kaplama üretiminde "DC", "Pulse" ve "Pulse Reverse" akım dalga formları darbe frekanslarının sayısı ve genlik değerleri değiştirilerek "darbe" tabanlı çalışmalar çeşitlendirilmeye çalışılsa da sonuç olarak kaplama çalışmaları bu 3 akım türü etrafında gerçekleştirilmektedir.

Mühendislik ve matematikte birçok fonksiyon ve bunlara ait dalga biçimleri mevcuttur. Bu tez çalışmasında DC, PC ve PRC akım yöntemlerinin dışında farklı dalga şekillerine sahip akımlar kullanılarak elektrodepozisyon yöntemi ile gerçekleştirilen yüzey kaplamalarının bu üç yönteme kıyasla kaplama başarılarında iyileştirme sağlayabildiği gösterilmiştir.

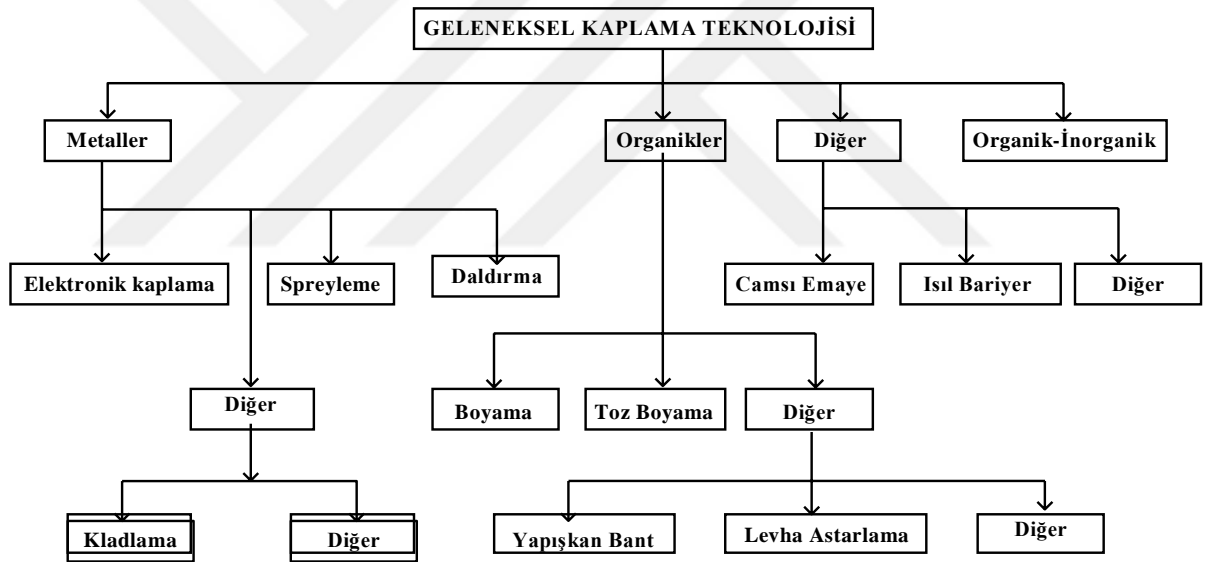
1.4. Sınırlılıklar

Farklı akım dalga türleri ile kaplama deneyleri gerçekleştirebilmek amacı ile tez çalışması kapsamında DC, PC, PRC akım dalga şekilleri ve bunların dışında da çeyrek Sin/Cos, yarım Sin/Cos, basamak girişi için 1. derece sistem çıkışı, basamak girişi için 2. derece sistem çıkışı ve rampa akım dalga şekilleri ile elektrodepozisyon yöntemine dayalı kaplama deneyleri gerçekleştirildi. Bunun yanı sıra tasarlanmış olduğumuz cihazla çok farklı malzeme ve kombinasyonlarda sayısız kaplama deneyleri gerçekleştirilebilmesi mümkündür. Bundan sonraki yapılacak çalışmalarda farklı malzeme ve farklı akım türleri ile yapılabilecek sonuçları değerlendirilebilir.

2. KAYNAK ARAŞTIRMALARI

2.1. Yüzey Kaplama Yöntemleri

Endüstride kullanılan mekanik parçalar ve yapılar özel uygulamalar için tasarlanmıştır. Mekanik veya kimyasal bir hasara maruz kalan malzeme, alet veya makine parçalarını korumak için çok çeşitli kaplama yöntem ve malzemeleri mevcuttur. Bu koruma yöntemlerinin bir yararı da yeni parçaların imalatına gerek kalmaması nedeniyle üretim maliyetlerini düşürmesidir (Fotovvati vd., 2019). Bu amaç doğrultusunda metalik, organik, inorganik ve organik-inorganik kaplamalar geliştirilmiştir (Harb vd., 2017; Almeida, 2001; Moreno vd., 2008). Geleneksel yüzey kaplama teknolojileri Şekil 2.1.'de gösterilmiştir.



Şekil 2.1. Geleneksel kaplama teknolojilerinin sınıflandırılması (Almeida, 2001).

2.2. Metalik Kaplamalar

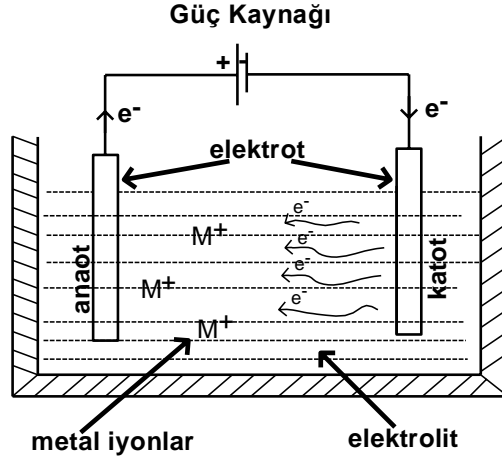
Metal veya alaşımlarının yüzeylerini kaplayarak mekanik özelliklerinde ve dekoratif görüntülerinde değişiklik sağlamak eski çağlardan beri çalışma konusu olmuştur. Gelişen teknoloji ile birlikte daha ağır çalışma ortamlarında kullanılan metalik malzemelerin, çalışma ortamlarının da etkisi ile mekanik ve kimyasal özelliklerinde çeşitli deformasyonlar ortaya çıkmaktadır. Bu deformasyonları azaltmak, malzemeye

istenen dayanımları kazdırmak, malzeme ömrünü arttırmak, malzeme kalitesinde iyileştirmeler sağlamak ve daha düşük maliyetli malzeme üretimi için geliştirilen yöntemlerden biri metalik kaplamalardır. Bu yöntemlerden bazıları sıcak daldırma, elektrolitik kaplama, metal spreyleme, difüzyonla kaplama, lazer kladlama gibi yöntemlerdir (Almeida, 2001; Moreno vd., 2008; Kakani S.L. ve Kakani A., 2004; Alshmi, 2014). Bunlardan elektrodepozisyon yöntemi ile kaplama son yıllarda öne çıkan ekonomik yöntemlerden biridir (Borkar ve Harimkar, 2011; Chung ve Chang, 2015).

2.3. Elektrolitik Kaplama (Elektrodepozisyon)

Elektrodepozisyon, uzun yıllardır farklı metalik malzemelerle alt malzeme yüzeylerini kaplamak için başarılı bir teknik olmuştur. Elektrodepozisyonda saf metaller, iletken polimerik malzemeler, metal kompozitler ve alaşımlar dahil olmak üzere çeşitli tipte altlık malzemeler kullanılabilir. Önceleri alt tabakaya korozyon, aşınma, aşınma dayanımı ve bazı durumlarda dekorasyon amacıyla yeni yüzey özellikleri sağlamak için yaygın olarak kullanılan elektrodepozisyon tekniği, malzeme mühendisliğindeki ilerlemeden dolayı, günümüzde manyetik yüzeylerin ve elektronik bileşenlerin istenilen duruma getirilmesi gibi daha spesifik uygulamalar için kullanılabilir (Melciu ve Maidee, 2015).

Elektrodepozisyon tekniği, banyo olarak bilinen iyi formüle edilmiş bir elektrolitten elektroliz yoluyla, bir metal veya alaşım kaplamanın iletken bir yüzey üzerinde biriktirilmesini içerir. Elektrodepozisyon işlemi sırasında, katottaki altlık yüzeyinde metal iyonlarının birikmesine, elektrolitte katot ve anot arasındaki yük aktarımı neden olur. Bu süreçte yük aktarımı, harici bir elektrik devresi tarafından yürütülür. Çok ince olan elektrolitik kaplamalar tek bir tabaka olarak uygulanabildiği gibi farklı metallerden birden fazla katlı olarak veya alaşım kaplamalar olarak da gerçekleştirilebilmektedir (Melciu ve Maidee, 2015; Estrada, 2008; Jayakrishnan, 2012). Elektrolitik kaplamanın şematik görünümü Şekil 2.2.'de gösterilmiştir (Melciu ve Maidee, 2015).

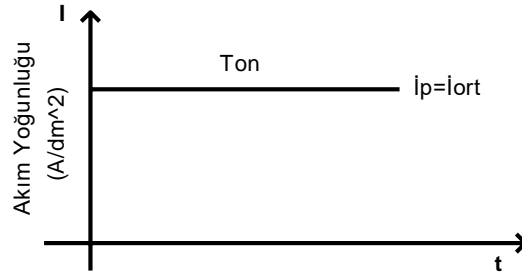


Şekil 2.2. Elektrolitik kaplamanın şematik görünümü (Melciu ve Maidee, 2015).

2.3.1. Doğru akım (DC) elektrodepozisyon yöntemi

Elektrodepozisyon, metal iyonları içeren bir elektrolit sıvısına uygulanan akım veya potansiyel yardımıyla elektrolitik birikim yoluyla malzeme üretim prosesidir. Bu metal iyonları proses sonucunda elektriksel olarak iletken bir altlıkta indirgenir ve böylece kaplama gerçekleşmiş olur. Bu teknolojinin, ilk sabit voltaj kaynağı olan volta pilinin piyasaya sürüldüğü 19. yüzyılın başında başlayan uzun bir geçmişi vardır. Başlangıçta, elektrodepozisyonla kaplama yöntemi dekoratif uygulamalar için ve daha sonra mühendislik uygulamaları için kullanılmıştır (Zangari, 2018). Günümüzde metalik kaplamaların üretiminde DC, PC, PRC akım dalga formları kullanılmaktadır (Chandrasekar ve Pushpavanam, 2007; Torabinejad vd., 2016; Matsui ve Omura, 2018; Kumar vd., Borkar, 2007; 2013; Yusoff vd., 2015).

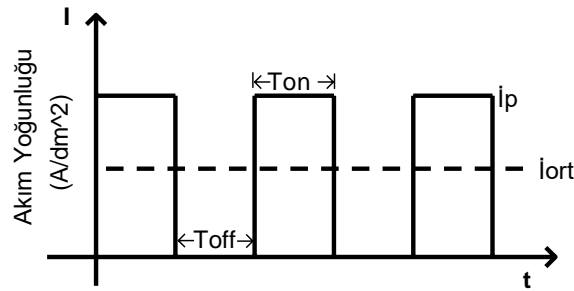
En yaygın kullanılan ve en eski elektrodepozisyon tekniği, doğru akım dalga formu ile yapılan tekniktir (Torabinejad vd., 2016). Geleneksel bir yöntem olan DC elektrodepozisyon tekniğiyle yapılan kaplamalarda sabit bir doğru akım dalga formu uygulanması nedeniyle ortalama akım yoğunluğu duty cycle değerine eşit olmaktadır (Borkar, 2007; Frey vd., 2011; Mulukutla, 2008; Fan, 2010; Hou vd., 2002; Kim ve Yoo, 1998; Wu vd., 2008). Bu nedenle DC elektrodepozisyon yönteminde akımın sadece yoğunluğu değiştirilebilmektedir. Aşağıdaki Şekil 2.3.'te geleneksel doğru akım dalga formu gösterilmektedir (Borkar, 2007; Torabinejad vd., 2016).



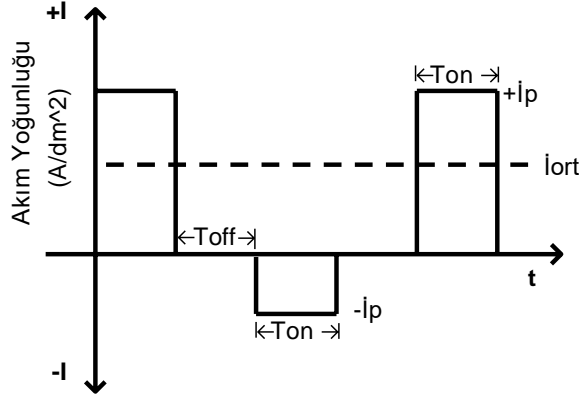
Şekil 2.3. Doğru akım dalga formu (Borkar, 2007).

2.3.2. Darbeli akım (PC) ve darbeli ters akım (PRC) elektrodepozisyon yöntemi

PC ve PRC yöntemiyle yapılan elektrolitik kaplamalar, DC elektrodepozisyon yöntemiyle kıyaslandığında, daha iyi yüzey özellikleri, yüksek serlik, korozyon direnci, düşük porozite, yüksek sıcaklık ve aşınma dayanımı, düşük maliyette üretim, kaplamanın hızlı olmasını sağlayan yüksek akım yoğunluklarında çalışma gibi özellikleri nedeni ile son yıllarda malzeme üretimi için kullanılan önemli üretim tekniklerindedir (Kasturibai ve Kalaignan, 2014; Sharma vd., 2014; Bakhit ve Akbari 2012; Balasubramanian vd, 2009; Gomes vd., 2011; Vb, 2014; Nigam vd., 2015; Jegan, 2018; Ghazanlou vd., 2016; Ren vd., 2015; Sajjadnejad vd., 2017; Shourgeshty vd., 2019; Zhou vd. 2010). Darbeli akım dalga formu ve darbeli ters akım dalga formu sırasıyla Şekil 2.4. ve Şekil 2.5.'te gösterilmiştir (Torabinejad vd., 2016; Pavithra vd. 2014).



Şekil 2.4. Darbeli akım dalga formu (Torabinejad vd., 2016).



Şekil 2.5. Darbeli ters akım dalga formu (Torabinejad vd., 2016).

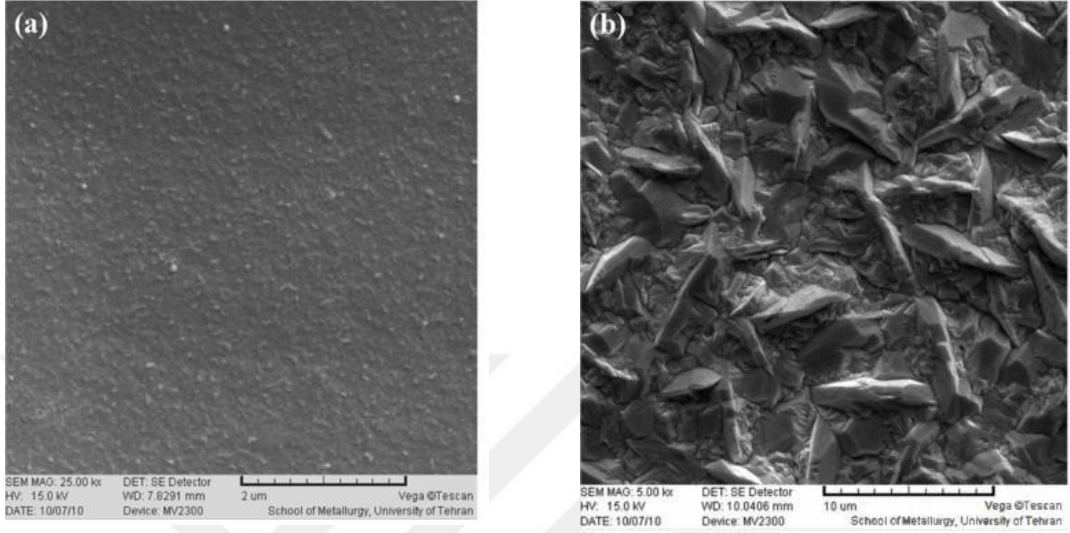
Darbeli elektrodepozisyon, geleneksel DC elektro kaplamaya göre birçok avantaj sunar;

- Tepe akımı (i_p), darbe açık kalma süresi (t_{on}), ve darbe kapalı kalma süresi (t_{off}) dahil olmak üzere biriktirme işleminin fiziğini kontrol etmek için ek serbest değişkenler sağlar. Bu nedenlerden ötürü DC kaplamayla elde edilemeyen bileşimler, morfolojiler ve mikro yapılara sahip elektrodepozitler oluşturmak için kullanılabilir .
- Darbeli elektrodepozisyon, t_{off} sırasında difüzyon katmanının yenilenmesi ile, daha yüksek bir anlık akım yoğunluğuna izin verir.
- Darbeli elektrodepozisyon ile mümkün kılınan yüksek akım, çekirdek oluşumu için mevcut enerjiyi artırır ve bu nedenle, DC kaplamaya kıyasla darbelerle daha yüksek çekirdeklenme hızlarına ulaşılabilir. Darbeli elektrodepozisyon, bu nedenle daha ince taneli kaplamalar üretmek için kullanılabilir.
- T_{off} sırasında, ara yüzdeki gevşemenin bir sonucu olarak yüzeyde elektriksiz adsorpsiyon, desorpsiyon ve yeniden kristalleşme reaksiyonları meydana gelebilir.

Bu avantajlarının yanı sıra darbeli elektrodepozisyon yönteminde, akım kaynağı üreticisine ait sistemin doğru akım sistemine göre pahalı olması ve bir dizi prosedürle birlikte önceden uygun planlama gerektirmesi, dezavantajları olarak söylenebilir (Borkar, 2007; Nusbaum vd. 2015).

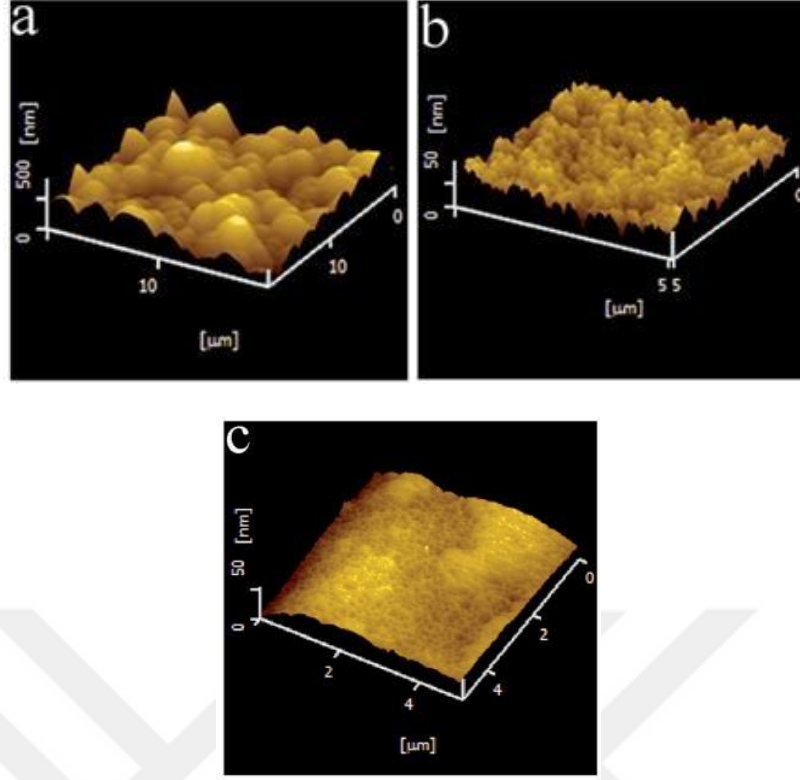
M. Siadat Cheraghi ve arkadaşlarının 2011 yılında DC ve PC elektrodepozisyon yöntemini kullanarak yapmış oldukları kaplama deney sonuçlarını Taramalı Elektron Mikroskobu (SEM) ile analiz etmişlerdir. Buna göre, PC ile üretilen kaplamaların yüzey

morfolojisinin DC kaplamalara oranla çok daha küçük tane boyutuna sahip olduğu ortaya konulmuştur (Cheraghi vd., 2011). Şekil 2.6.'da bu çalışmaya ait SEM analizleri gösterilmektedir.



Şekil 2.6. Co Kaplamaların yüzey morfolojisi (a) PC kaplı ve (b) DC kaplamalı sırasıyla 20 ve 100mA.cm⁻² (Cheraghi vd., 2011).

Ruiqian Li ve arkadaşlarının 2017 yılında DC, PC ve PRC akım türlerinde Co-P kaplamaların yüzey özelliklerini inceledikleri çalışma sonuçlarına göre; DC ve PC yöntemleriyle karşılaştırıldığında, PRC yöntemi ile yapılan kaplamanın, yüzey pürüzlülüğünde daha iyi sonuçlar ortaya koyduğu rapor edilmiştir. Şekil 2.7.'de bu üç akım türüyle yapılan deneylere ilişkin Atomik Kuvvet Mikroskobu (AFM) sonuçları gösterilmiştir (Nusbaum vd. 2015).



Şekil 2.7. (a) DC, (b) PC ve (c) PRC yöntemleriyle biriktirilen Co – P kaplamaların AFM görüntüleri (Nusbaum vd. 2015).

2.4. Elektrolitik Kaplamaya Etki Eden Faktörler

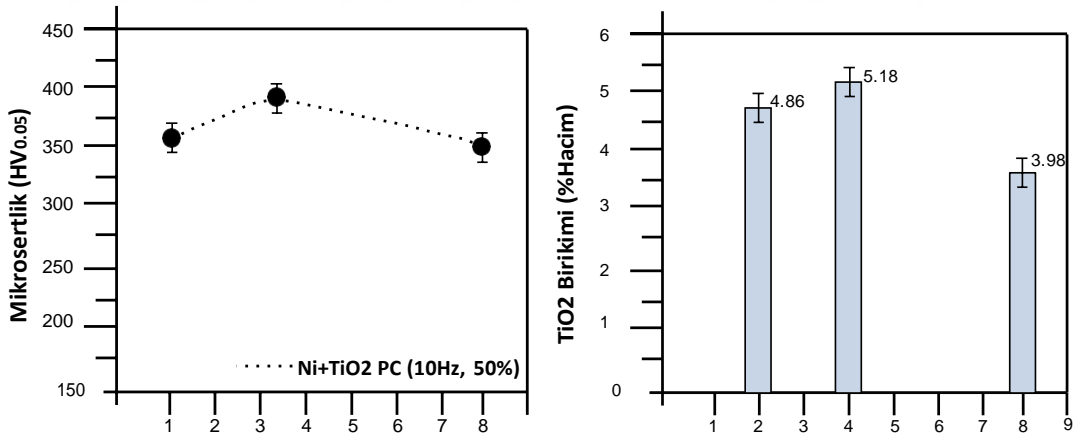
Elektrolitik kaplama koşullarına ait parametreler kaplama sonuçlarını belirlemede oldukça önemlidir. Elektrodepozisyon tekniği ile yapılan kaplamalara etki eden başlıca faktörler; kaplama banyosunun bileşimi, kaplama akım yoğunluğu, kaplama akım türü, kaplama akımı işlem süresi (duty cycle), kaplama frekansı ve kaplama sıcaklığı olarak sıralanabilir (Borkar, 2007; Bahrololoom ve Sani, 2005; Thiemig vd., 2007; Boukhouiete ve Creus, 2015). Bu tez çalışmasındaki amacımız, elektrolitik kaplamaya etki eden en önemli faktörlerden biri olan kaplama akımını, farklı türlerde üretmek ve üretilen bu akım türleriyle yapılan kaplamaların sonuçlarını incelemektir.

2.4.1. Akım yoğunluğu

Elektrolitik kaplamalarda, kaplamanın faz yapısı, tane boyutu, sertliği, kimyasal bileşimi, manyetik özelliği gibi parametreleri kontrol etmek amacı ile farklı akım

yoğunluğuna sahip kaplamalar gerçekleştirilmektedir. Kaplama sürecinde akım yoğunluğunun artmasına paralel olarak kaplama parametrelerinde artış olmasına rağmen belirli bir değerden sonra bu artış negatif yöne doğru gidebilmektedir. Bu nedenle kaplama sırasında optimum akım yoğunluğu seçimi, istenilen malzeme üretiminde önemli bir yer tutmaktadır (Borkar, 2007; Bahrolloom ve Sani, 2005; Thiemig vd., 2007; Lajevardi ve Shahrabi 2010; Yüksel, 2007; Chen vd., 2005).

2010 yılında S.A. Lajevardi ve T. Shahrabi, PC elektrodepozisyon parametrelerinin Ni – TiO₂ nanokompozit kaplamaların özellikleri üzerindeki etkilerini araştırmışlardır. Bu çalışmada, farklı ortalama akım yoğunluğundaki kaplamaların, sertlik ve biriktirilen malzeme miktarı üzerindeki etkileri ortaya konulmuştur. Buna göre, mikrosertlik ve partikül birikimi başlangıçta ortalama akım yoğunluğu 2A/dm²'den 4A/dm²'ye kadar bir artış göstermiş ancak 8A/dm² ye doğru arttırılınca, düşüş eğilimi göstermiştir. Bu değerlere göre, maksimum mikrosertlik ve en yüksek partikül hacim yüzdesi elde etmek için, ortalama 4A/dm² civarındaki akım yoğunluğu optimum akım yoğunluğu olarak temsil edilmektedir (Lajevardi ve Shahrabi 2010). Bu deney sonuçlarına ait veriler Şekil 2.8.'de gösterilmektedir.



Şekil 2.8. Ortalama akım yoğunluğunun (a) kompozit kaplamaların mikrosertliği ve (b) 10 Hz sabit frekansta ve % 50 görev döngüsünde biriken TiO₂ parçacıklarının hacim yüzdesi üzerindeki etkisi (Lajevardi ve Shahrabi 2010).

2.4.2. İşlem süresi (Duty cycle)

Elektrodepozisyon proseslerinde duty cycle, darbe açık kalma süresinin (t_{on}), darbe açık ve darbe kapalı kalma süresinin (t_{off}) toplamına oranı şeklinde ifade edilir ve denklem (2.1) ile hesaplanabilir (Wasekar vd., 2016). Denklemdeki " θ " işlem süresini ifade etmektedir. Bu sürelerinin değiştirilmesi ile yapılan kaplamalar sayesinde, sertlik, tane yapısı, aşınma direnci gibi malzeme özelliklerinde iyileştirmeler sağlayarak, geleneksel doğru akım yöntemi ile elde edemeyeceğimiz malzemeleri üretmemiz mümkün olmaktadır (Borkar, 2007; Bahrololoom ve Sani, 2005).

İşlem süresi θ olmak üzere;

$$\theta = \frac{t_{on}}{t_{on} + t_{off}} \quad (21)$$

2.4.3. Darbe frekansı

Elektrodepozisyon prosesinde darbe frekansı da diğer akıma ait parametrelerde olduğu gibi malzeme birikimi, sertlik, yüzey morfolojisi gibi özelliklerin iyileştirilmesinde önemli bir rol oynamaktadır (Borkar, 2007; Bahrololoom ve Sani, 2005; Chen vd., 2005; Wasekar vd., 2016; Pavlatou ve Spyrellis, 2008; Habibzadeh vd., 2015). Doğru akım dalga formu ile yapılan kaplamalarda ortalama akım yoğunluğu, duty cycle (θ) değerinin %100 olması nedeniyle maksimum akım değerine eşittir. Ancak darbe akım dalga formu kullanıldığında akım yoğunluğu ve frekansı t_{on} ve t_{off} sürelerine bağlı olarak denklem 2.2'deki formüle uygun olarak değişmektedir (Wasekar vd., 2016).

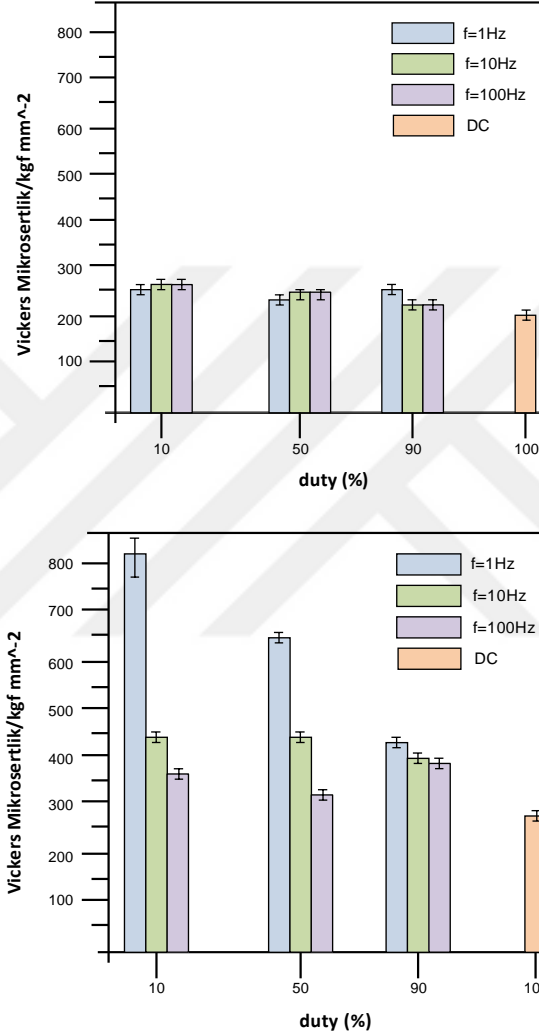
f, frekans olmak üzere;

$$f = \frac{1}{t_{on} + t_{off}} \quad (22)$$

i_{avg} , ortalama akım değeri ve i_p maksimum akım değeri ve θ işlem süresi olmak üzere;

$$i_{avg} = i_p * \theta \quad (23)$$

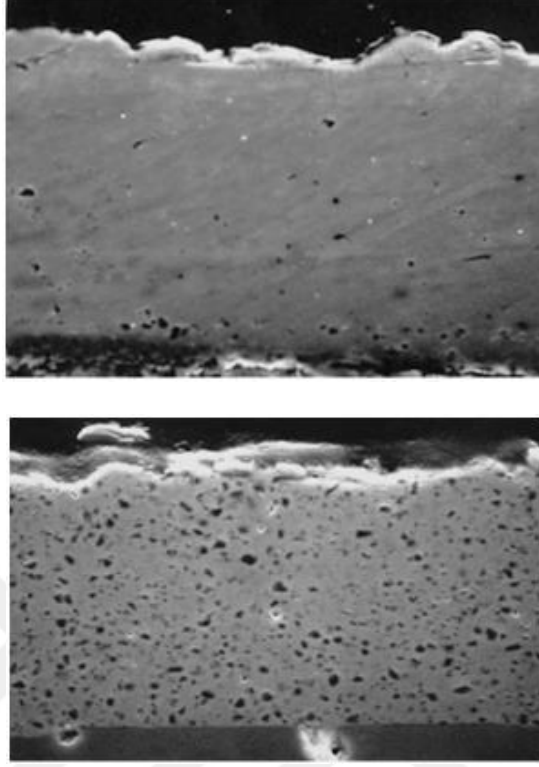
Gyftou ve arkadaşlarının 2008 yılında yapıları çalışmada ise , darbeli elektrodepozisyon tekniği ile yapılan Ni/SiC kompozit biriktirme işleminde, Şekil 2.9 (a) ve (b)' de gösterildiği gibi akımın özellikle düşük duty cycle ve frekans değerlerinde uygulanmasıyla oluşturulan birikintilerin sertliğinde önemli ölçüde iyileşme sağlandığı ortaya konulmuştur (Gyftou vd., 2008).



Şekil 2.9. Vickers mikrosertlik değerleri: (a) saf Ni ve (b) DC ve PC koşulları altında farklı darbe frekanslarında ve işlem süresinde hazırlanan kompozit Ni / SiC birikintileri (Gyftou vd., 2008).

E. A. Pavlatou ve N. Spyrellis'in elektrodepozisyon yöntemi ile ürettikleri kaplama sonuçlarında, düşük duty sürelerinin, t_{off} değerinin t_{on} değerinden daha fazla olmasına bağlı olarak, kaplama sertliğinde ve aynı zamanda profil kesitinde daha uniform yapıda SiC birikimi sağladığı rapor edilmiştir (Pavlatou ve Spyrellis, 2008).

Şekil 2.10.'da DC ve PC kaplamalara ilişkin profil kesitlerinin SEM analizi gösterilmektedir.



Şekil 2.10. (a) DC ve (b) PC koşulları (d.c. =% 90, $v = 0.1$ Hz) altında hazırlanan Ni / mikron SiC birikintilerinin enine kesit profilinin SEM mikrografları (Pavlatou ve Spyrellis, 2008).

2.5. Elektrodepozisyon Yöntemi İle Nikel Kaplamalar

Ni ve alaşımları, iyi korozyon ve aşınma direnci gibi çeşitli avantajlara sahip olduklarından, biriktirme uygulamalarında büyük ilgi görmektedirler. Dünyadaki Ni tüketiminin yaklaşık % 12'si Ni kaplamanın elektrodepozisyonuna ayrılmaktadır (Torabinejad vd., 2017; Di Bari, 2000). Ni kaplamanın dekoratif amaçlı kullanımı, elektrolizle kaplanmış Ni kaplamanın ana uygulamasıdır. Ni birikintileri, endüstride aşınma ve korozyon direncini iyileştirmek, aşınmış metalleri onarmak, küçük boyutlu parçaların boyutlarını değiştirmek, manyetik özellikleri iyileştirmek, cam veya organik kaplama üretmek için altlık yüzeyini hazırlamak ve diğer amaçlar için kullanılır. Saf Ni kaplamaya ek olarak, bir elektro biriktirme işlemi ile Ni bazlı bir alaşım kaplaması üretmek mümkündür (Torabinejad vd., 2017; Liu vd., 2011).

Ni elektrodepozisyonunun ilk tanımı, birkaç saat nikel klorür / sülfat çözeltisini elektrolize eden ve böylece bir platin elektrot üzerinde bir metalik nikel kabuğu elde eden G. Bird (1837) tarafından yapılmıştır. İngiltere'den J. Shore (1840), nikel nitrat çözeltisinden ticari Ni plakası kaplamayı başarmıştır. Mühendislik alanında nikel anot kullanımı, işlevsel davranışının, görünüşünden çok ana kriter olduğu uygulamalar için pratik bir önem kazanmıştır (Raghavendra vd., 2018).

Bu tez çalışmasında bakır altlık (Cu) üzerine Ni anot kullanılarak yeni nesil akım üreteç sistemi ile farklı akım dalga türlerine ait elektrodepozisyon yöntemi ile yapılan kaplama deneylerinin sonuçları irdelenmiştir.



3. MATERYAL ve YÖNTEM

Yapılan tez çalışması kapsamında, kaplama deneylerini gerçekleştirebilmek için, bilgisayar kontrollü çok fonksiyonlu akım kaynağı tasarımı gerçekleştirilmiştir. Bilgisayar tabanlı elektronik sistem kontrolü, güç elektroniği, kontrol, yazılım gibi farklı disiplinler bir arada kullanılarak gerçekleştirilmiştir.

Tasarım aşaması için gerekli olan alt yapı bu bölümde ayrıntılı bir şekilde ele alınmıştır. Bu bölüm, güç elektroniği devre topolojisi, kontrol alt yapısı, elektronik sistem ve bilgisayar yazılımı tasarım aşamalarını içermektedir.

3.1. Buck Dönüştürücü

İnsanlığın gelişimi ile beraber, ihtiyaçlar çerçevesinde elektronik sistemler ve elektronik devreler de şekillenmektedir. Son zamanlarda elektrik enerjisine olan ihtiyacın artması, enerji iletiminde ve kullanılmasında en önemli parametrelerden biri olan verimliliğin ön koşul haline gelmesine neden olmuştur (Kraiem ve Shaaban, 2020; Wiechmann vd., 2009; Rafi vd., 2020; Hofer ve Schrod, 2020; Chen vd., 2013). Özellikle de elektrikli arabalara olan talebin artması, daha verimli çalışabilen elektronik devre sistemlerine ihtiyacı artırmıştır.

Geleneksel enerji aktarım devresi olan lineer regülatör (78XX serisi entegre devreleri) bugüne kadar yoğun kullanılmış olsa da, verimli enerji aktarımı için kullanılacak elektronik devre sistemi değildir. Bunun sebebi ise lineer regülatör çalışma verimliliğinin yük akımına dayalı çok geniş bir skalada değişkenlik göstermesidir.

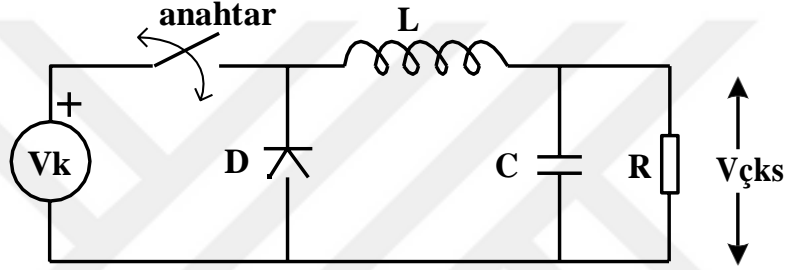
Günümüzde enerji aktarımı, anahtarlamalı (switch) güç elektroniği devre topolojileri kullanılarak gerçekleştirilmektedir. Bu topolojiye ait elektronik devreler, yüksek verimde enerji aktarımını gerçekleştirirler (2016; More vd., 2018; Mahmud vd., 2018; Shantala ve Sudheer, 2018).

Tasarlamış olduğumuz akım kaynağı cihazı içerisinde yer alan elektronik sistemde, güç elektroniği devre topolojilerinden biri olan Buck tipi dönüştürücü devresi kullanılmıştır. Buck dönüştürücü devresi, kaynak gerilimini, yüksek dönüşüm verimliliği ile daha düşük gerilimlere dönüştürebilmektedir ve kaynaktan yüke enerji

aktarım verimliliği %94 ile %98 arasında deęişiklik gösterebilmektedir (Jha ve Singh, 2017; Jheng vd., 2019; Veerachary ve Misal, 2020).

3.1.1. Buck dönüştürücü çalışma prensibi

Geleneksel tasarım yönteminde Buck tipi dönüştürücü, 4 adet güç elektronięi devre elemanı, 1 adet kaynak ve 1 adet enerji tüketicisinden (R) oluşmaktadır. Buck dönüştürücü devresinde, iletim ve kesim olmak üzere 2 farklı enerji durumu mevcuttur, Şekil 3.1.



Şekil 3.1 Buck tipi dönüştürücü devresi.

Şekil 3.1.'de yansıtılmış olan Buck dönüştürücü devresine ait devre elemanları sırasıyla,

V_k: kaynak gerilimi

Anahtar: -

D : diyot

L: bobin

C: kondansatör

R: yük direnci

şeklinde tanımlıdır.

Anahtar, kapalı konumda ise enerji devre elemanlarına iletilmekte; açık konumda ise kesilmektedir. İletim sürecinde, kaynak gerilimi bobine uygulanır ve yük üzerinde deęeri artan bir akım oluşur. Kesim sürecinde ise kaynak gerilimi devreden

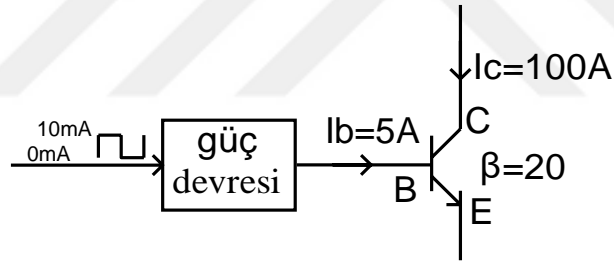
alınarak devreye enerji girişi engellenir. Bu süreçte bobin ve kondansatör enerjisi yük üzerinde harcanır ve giderek yük akımı düşmeye başlar.

Buck dönüştürücü topolojisine ait devre elemanlarının görevleri alt başlıklar şeklinde açıklanmıştır.

3.1.1.1. Anahtar

Anahtar, devre akımını kesebilen veya iletilmesine izin veren bir elektronik devre elemanıdır. Yüksek hızda iletim ve kesim yapabilecek kapasiteye sahip olmalıdır. Bu nedenle anahtar elemanı olarak transistör, fet veya igbt güç elektroniği devre elemanlarından biri seçilir.

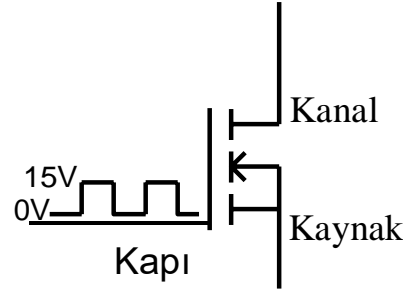
Yüksek güçlü transistörlerin I_c ile I_b arasındaki kazancın düşük oluşundan dolayı transistörün anahtar görevini yerine getirebilmesi için "base" girişine bir güç yükselteci devresi ilave edilir, Şekil 3.2. Bundan dolayı yüksek güçlü Buck dönüştürücü devrelerinde transistör tercih edilmemektedir.



Şekil 3.2. Güç transistörü ve yüksek "base" akım ihtiyacı.

Tasarım maliyetini düşürmek amacıyla kaynak geriliminin düşük değerleri için MosFET; yüksek değerleri için de IGBT'nin anahtarlama elemanının seçilmesi maliyetlerin düşürülmesi açısından önemli bir parametredir.

Anahtar olarak kullanılacak olan güç elektroniği devre elemanı, "kapı" girişine gerilim darbe katarı gönderilerek akımın iletimi ve kesimi sağlanır, Şekil 3.3.



Şekil 3.3. Anahtarlama elemanına gerilim darbe katarı gönderilmesi.

Gerilim darbe katarı sabit frekanslı olup, Sadece Ton ve Toff süreleri değiştirilmektedir. $Ton+Toff=T=1/f$ eşitliği sabit kalma koşulu altında Ton süresi değiştirilir. Literatürde, bu yöntem ile şekillendirilen gerilim darbe katarına PWM (pulse width modulation, darbe genişlik modülasyonu) ismi verilir, (Singh vd., 2019; Islam vd., 2016; Hridya ve Srinivas, 2015; Kumar ve Makarand, 2017; Lopatkin, 2017). PWM yöntemine dayalı anahtarlama devre topolojisi sayesinde, kaynak geriliminin ortalama DC gerilim seviyesi değiştirilerek, en yüksek verimle R tüketici elemanına aktarılabilmektedir.

3.1.1.2. Bobin

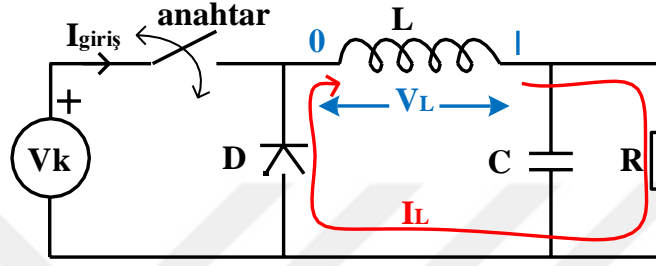
Bobin, anahtar kapatılınca oluşacak olan yüksek akımın ani artışını engellemek ve elektrik enerjisini depolama amacıyla kullanılmaktadır, denklem 3.1.

$$E_L = \frac{1}{2} L * I^2 \quad (3.1)$$

Anahtar kesim moduna geçince bobin de üzerindeki enerjiyi akıma dönüştürerek devreye devre akımı ile aynı yönde verir. Bobin, enerjisini akım olarak devreye enjekte edebildiği için anahtar kesim çalışma modundayken devre akımında süreklilik sağlanmış olur.

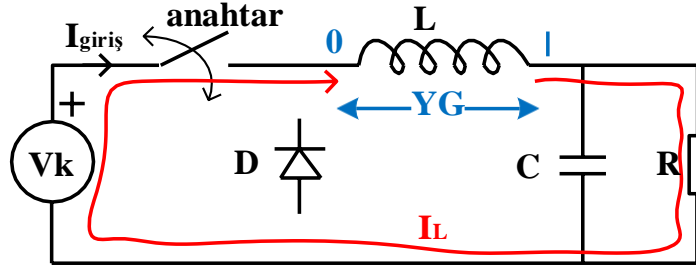
3.1.1.3. Diyot

Buck tipi dönüştürücü devresinde yer alan diyot, serbest geçiş diyodu (freewheeling diode) olarak da isimlendirilir. Kullanım ihtiyacı, devredeki bobinden kaynaklanır. Bobin enerjisi, anahtarın kesim moduna geçtiği an en yüksek değerdedir ve $\tau(L,C,R)$ zaman sabitesine dayalı azalmaya başlar. Azalan enerji, devreye akım olarak yansır ve yönü R yük direncine doğru olur. Bobinin akımı yönlendirebilmesi için uçları arasında yeterli miktarda potansiyel fark oluşur, Şekil 3.4.



Şekil 3.4. Diyot devrede, anahtar kesim modundayken IL akım yolu.

Anahtar, kesim moduna girdiği anda bobin akımını farklı bir iletim yolu izleyemez ise eski akım yolunu takip etmeye çalışır. Bu akış ise istenmeyen bazı olaylara sebep olabilir. Bunun sebebi ise bobin üzerine indüklenerek olan yüksek gerilimdir (YG), Şekil 3.5.



Şekil 3.5. Diyot devre dışı, anahtar kesim modundayken IL akım yolu.

Anahtar kesim moduna girince, bobin akımını diyot üzerinden devresini tamamlayarak bobin uçları arasında yüksek gerilimin oluşması engellenir, Şekil 3.5.

3.1.1.4. Kondansatör

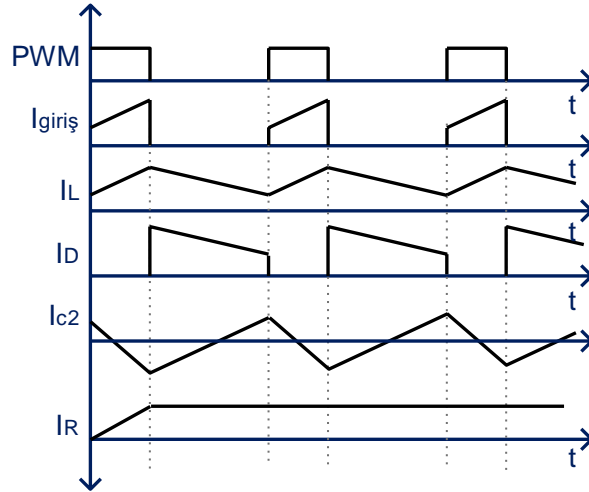
Kondansatör, R yük üzerindeki çıkış geriliminin \pm aşım miktarının kontrolü ve yine R yük üzerinde oluşacak olan harmonikleri minimize etmek amacıyla kullanılır. Kondansatörün bir diğer kullanım amacı da anahtar kesim modundayken, kondansatör üzerinde birikmiş olan enerjiyi R yük üzerine yönlendirmesi içindir, Denklem 3.2. Böylece anahtar kesim modundayken R yük akımının sürekliliği zamana dayalı artırılmış olur.

$$E_c = \frac{1}{2} * C * V^2 \quad (32)$$

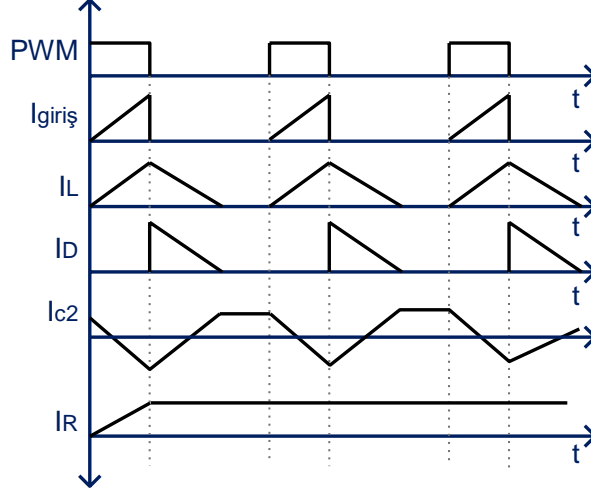
3.1.2. Buck Dönüştürücü çalışma modları

Buck dönüştürücü devresi sürekli iletim modu (continuous conduction mode, CCM) ve süreksiz iletim modu (discontinuous conduction mode, DCM) olmak üzere iki farklı iletim moduna sahiptir.

Şekil 3.6'da CCM çalışma modunda devre elemanlarına ait akım ve şekilleri; şekil 3.7'de ise DCM çalışma moduna ait devre elemanlarının akım ve şekilleri yansıtılmıştır.



Şekil 3.6. CCM çalışma modu ve devre elemanlarına ait akım dalga şekilleri.

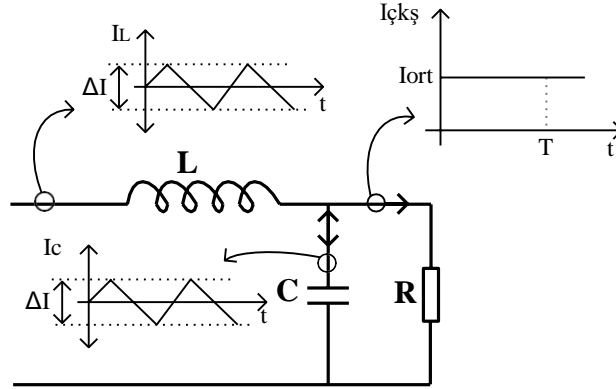


Şekil 3.7. DCM çalışma modu ve devre elemanlarına ait akım dalga şekilleri.

3.1.3. Devre elemanlarının hesabı

Buck dönüştürücüde kullanılan devre elemanları, PWM frekansı, bobin akımı, çıkış gerilimi ve tüketilen güç olmak üzere çeşitli parametrelere dayalı hesaplanır.

Buck dönüştürücüsünde kullanılan bobin değeri yine bobin içerisinde geçecek olan akımın ortalama ve salınım miktarına dayalı hesaplanır, şekil 3.8.



Şekil 3.8. Buck dönüştürücü salınım ve ortalama akım dalga formları.

Bobinde indüklenen gerilim, bobin akımının periyodik aşım miktarına dayalı oluşur. Denklem 3.3'de bobin uçlarında indüklenen gerilimin denklemi verilmiştir.

$$V = L \frac{dI_L}{dt} \quad (33)$$

Denklemden yer alan Δi_L , bobin akımının değişim aralığı olan ΔI_L değeridir, Şekil 3.9. Denklemden yer alan dt ise PWM periyodunun sıfır olma süresidir. Bu bilgiler çerçevesinde Denklem 3.3 tekrar düzenlenirse,

$$V_L = L \frac{\Delta i_L}{(1-D) * T} \quad (34)$$

elde edilir. T yerine frekans ve L parametresi eşitliğin sol tarafında yalnız bırakılır ve bobin gerilimi hesabı için kaynak gerilimi ve çıkış gerilimleri arasındaki fark alınırsa,

$$L = \frac{V_K - V_{ÇKŞ}}{\Delta i_L * f_{PWM}} * (1 - D) \quad (35)$$

bobin endüktans denklemi türetilmiş olur. Öngörülemeyen kaçak enerjilerin oluşturacağı fazladan yükün meydana getireceği gerilim düşümünü karşılayabilmesi için PWM duty değeri, hesaplanan değerden %10 daha yüksek seçilir, Denklem 3.6.

$$L = \frac{V_K - V_{ÇKŞ}}{\Delta i_L * f_{PWM}} * (1 - 11 * D) \quad (36)$$

Bobin içerisinden geçecek olan ortalama ve değişken akım değerlerinin toplamı, yine bobin içerisinde en yüksek ve en düşük tepe akım değerleri yani pik akım (I_p) değerlerini oluşturur, Denklem 3.7.

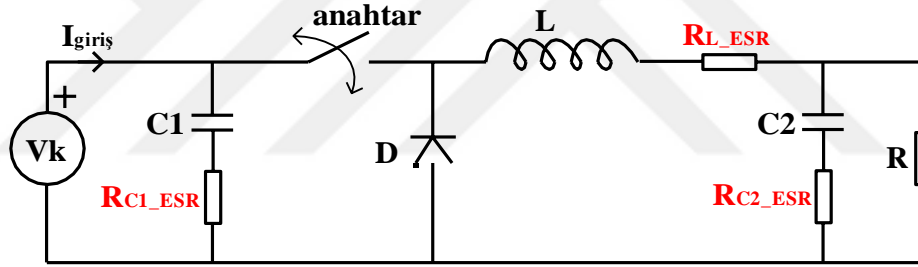
$$I_p = I_{ort} + \frac{\Delta I_L}{2} \quad (37)$$

ΔI_L akımını I_{ort} akımının %30 olacak şekilde seçilerek I_p akımını hesaplanır. Düzenlemeler yapıldıktan sonra Denklem 3.7, Denklem 3.8'e dönüşmüş olur.

$$I_p = I_{ort} + \frac{\%30 * I_{ort}}{2} \quad (38)$$

Hesaplamalardan elde edilecek endüktans değerinden sonra üretilecek olan bobinin doyum noktası parametresine dikkat edilmelidir. Bobin doyum noktası I_p akımına göre daha yüksek bir akım değerinde olmalıdır. Hesaplanarak elde edilecek bobin endüktans parametresinin sabit kalabilmesi için ise bobinin lehmleneceği devre, ortam ısısı ile bobin ESR direncinden kaynaklı ısıların toplam değeri belirli bir eşik değerinin altında olmalıdır.

Buck dönüştürücü devresinde yer alan bobin ve kondansatörlerin iç mekanizmalarına ait ve bobinle kondansatöre seri olarak ilave gelen dirençleri mevcuttur. Literatürde bu dirençlere ESR (Equivalent Series Resistance-eşdeğer seri direnç) adı verilmektedir (Muhurcu vd., 2017). ESR dirençleri göz önüne alınarak Buck tipi dönüştürücü devresi tekrar çizilirse, son şeklini almış olur, Şekil 3.9. Dikkat edilirse anahtarın önüne de bir adet kondansatör konulmuştur. C1 kondansatörü giriş; C2 kondansatörü çıkış kondansatörü olarak adlandırılır.



Şekil 3.9. Bobin ve kondansatör ESR dirençleri.

Bobin ESR direncinden kaynaklı ısı, bobin içerisinden geçen akımla doğru orantılıdır, Denklem 3.9.

$$P_{RL} = R_L * I_L^2 \quad (39)$$

Bobin esr değerinden kaynaklı direncin oluşturduğu ısı hesabında I_p değerleri ihmal edilir. Ayrıca, $+I_p$ ile $-I_p$ değerleri bir birlerini yok etmektedir. Bundan dolayı da I_L akımı, bobinden geçen ortalama akım olarak seçilir.

Bobin çekirdek malzemesi, çekirdek kayıplarını hedef koşullar altında belirtilen sınırlar içinde sınırlandırabilmek için rastgele seçilmemelidir.

Bobin RL, ESR direncinden kaynaklı yüksek ısı elde edilirse bu defa paralel bobin bağlantı mantığına başvurulur. Bu yöntemle, ESR'den kaynaklı yüksek ısı düşürülür böylece akım kollara ayrılmış olur, Denk. 3.10.

$$I_{KOL} = \frac{P_{ÇKŞ}}{V_{ÇKŞ}} * \frac{1}{n} \quad (310)$$

Denklemden kullanılan n parametresi, yük akımının kol sayısını temsil eder. Yük akımına ait akım hattı artırılırsa yani paralel bobin sayısı artırılırsa, bobin ESR direncinden kaynaklı oluşan ısı gücü de karesel olarak azalır.

Doğru polarma sürecinde devreye bağlı olan diyot güç kaybı,

$$P_D = V_D * I_{KOL} * (1 - D) \quad (311)$$

biçiminde hesaplanır. Uygulamada, PWM lojik seviyesi "0" olduğu sürece bobin akım yolu için diyot kullanılmasının Buck dönüştürücünün güç aktarım verimini oldukça düşürdüğü tespit edilmiştir. Devrede kullanılan diyot yerine diyot görevini yapacak en verimli tetikleme elemanı MosFET 'dir. Anahtar kesim modundayken MosFET güç kaybı,

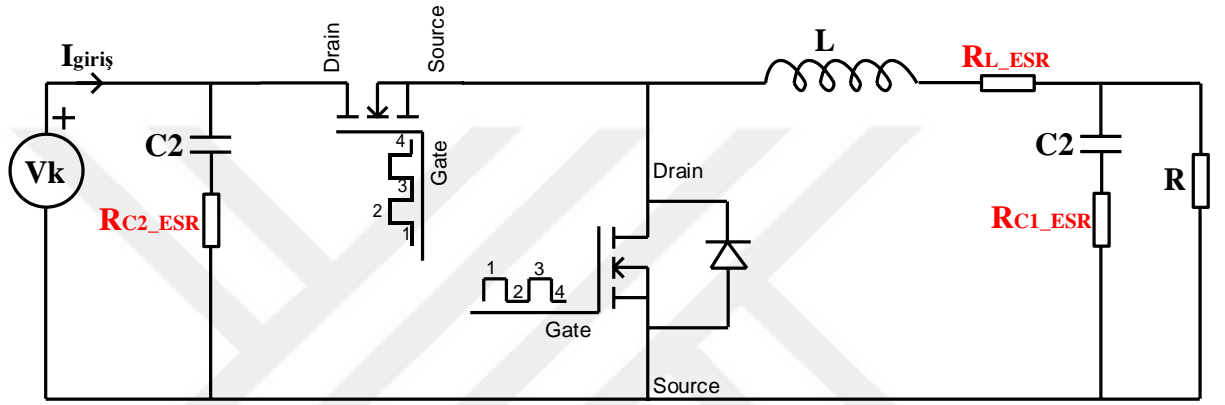
$$P_{MosFET} = I_R^2 * R_{D,MosFET}(1 - D) \quad (312)$$

biçiminde hesaplanır. Denklemden R_D direnci, MosFET iletim yönünde görülen "kanal" ile "kaynak" arasındaki dirençtir. Bu direnç olabildiğince düşük olan MosFET seçilerek gerçekleştirilirse Buck dönüştürücüsünün verimi de o derece yüksek olur. Örneğin, BSC010NE2LS numaralı mosfet, 5V iken iletim direnci sadece 1mΩ'dur.

Diyot yerine kullanılan FET için, ters geri kazanım kaybı, ölü zaman kaybı ve kapı tahrik kaybı biçiminde ilave 3 kayıp mevcut olacaktır. Bu kayıpların tamamı dinamik parametrelerdir.

Dinamik kayıp parametrelerinden ters geri kazanım kaybının hesabı kolay bir şekilde yapılması veya veri kataloğundan elde edilmesi oldukça zordur ve çok fazla parametreye dayalı hesaplanır. Fakat oluşan kayıplar o kadar küçüktür ki çoğunlukla ters geri kazanım kaybı yok sayılır.

Bazı MosFET'ler "kanal" ile "kaynak" arasında ters bir diyot yer almaktadır. Bu diyot kurtarıcı bir diyottur. Bunun sebebi ise çok küçük bir zaman aralığında; (ölü tetikleme zamanı) MosFET devreye girene kadar diyot kurtarıcı olarak devreye girer ve hemen mosfet kısa devre olarak diyodun görevini üstlenir, Şekil 3.10.



Şekil 3.10. Verim artışına sebep olan MosFET bağlantısı.

Tetiklemelerde meydana gelen ölü zaman, anahtar MOSFET ile diyot yerine geçmiş MosFet tetiklenmesi esnasında oluşur. Ölü zamanın (öz) süresi tetikleme frekansı ile orantılıdır ve güç kaybı,

$$P_{ÖZ} = V_D * I_{ÇKŞ}(t_{on,MosFET}) * f_{PWM} \quad (313)$$

biçiminde hesaplanır.

MosFET Vgs gerilimi de kayıplara sebep olmaktadır. Vgs gerilimi, MosFET iletim direncini ve MosFET açma kapama hızını etkilemektedir. MosFET Gate şarj süresi iletim direnci olan Rd ile ters orantılıdır. Vgs tetikleme gerilimi sebebiyle oluşan kayıp,

$$P_{MosFET, Gate} = Q_G * V_{GS} * f_{PWM} \quad (314)$$

biçiminde hesaplanır. Buck dönüştürücü devresinde tüm büyük kayıplar göz önüne alınırsa,

$$\% \eta = \frac{P_R}{P_{\text{TOPLAM}}} * 100 = \frac{P_R}{P_R + P_{R_L} + P_D + P_{\text{ÖZ}} + P_{\text{MosFET, Gate}}} * 100 \quad (315)$$

enerji iletim verimi tam olarak hesaplanmış olur.

3.1.4. Anahtarlama mosFET'i

Buck dönüştürücü, hard-switch güç elektroniği topolojisine sahiptir. Anahtar elemanı olarak kullanılacak olan MosFET ideal bir açma kapama anahtarı gibi davranmak zorundadır. Bunun için mümkünse 0 ohm "kanal-kaynak" akım yolu direnci ve sonsuz tetik işaretini karşılaması gerekir. Bunlar da mümkün olmadığı için de mümkün olduğu kadar "kanal-kaynak" düşük iletim yolu direnci ile yüksek tetiklenme frekansına sahip olmalıdır.

Buck dönüştürücü topolojisinde kullanılacak MosFET şu kriterler çerçevesinde seçilmelidir,

- 1- MosFET'in iletimi esnasında daha düşük $\frac{d}{dt}$ etkileşimin yaşanabilmesi amacıyla $\frac{Q_{GD}}{Q_{GS}} < 1$ olmalıdır.
- 2- Yüksek hızlı açma ve kapama kabiliyeti. Buna destek olarak "gate" "plateau" bölgesine erişim süresi, sürme sürelerinin ortalarında yer almalı.
- 3- Tetikleme ve iletim kayıpları Buck dönüştürücü tasarım parametreleri çerçevesinde minimum düzeyde tutulmalı.
- 4- Vds gerilim değeri, bara ve oluşabilecek gerilim yükselmelerini karşılaması.
- 5- Düşük termal direnç, Rthjc
- 6- MosFet üzerinde yer alan diyot önem arz etmemektedir. Bu diyot çok seyrek devreye girmektedir.

3.1.4.1 Mosfet çıkış kondansatörünün (coss) etkisi

Mosfetin çıkış kapasitansı istenmeyen bir parametredir. Bu kondansatör MosFET iletim akımı kesime gidince şarj; iletime geçince ise deşarj olur, Denklem 3.16.

$$P_{COSS} = \frac{1}{2} * Q_{COSS} * V_K * f_{PWM} \quad (316)$$

biçiminde hesaplanır. Yüksek hızlı tetiklemelerde endüktans tabanlı sınırlama gelir ve enerji kaybı,

$$P_{Anahtarlama_YH} = \frac{1}{2} * L_{Stray} * I^2 * f_{PWM} \quad (317)$$

formülü ile hesaplanır. Gate seri bağlanmış dirençten kaynaklı sınırlamadan dolayı oluşacak kayıplar,

$$P_{Anahtarlama_YH} = V_K * I_R * \frac{Q_{SW}}{R * V_{GS}} * f_{PWM} \quad (318)$$

biçiminde hesaplanır. Burada Qsw şarj yüküdür. Rgate, MosFET "gate-source"a bağlanan sürme devresine ait toplam direnç değeridir.

3.1.5. Buck çıkış kondansatörü

Buck dönüştürücüde yer alan çıkış kapasitesi (C2), bobin akımında ve çıkış geriliminde meydana gelecek olan dalgalanmayı minimize etmek amacıyla kullanılmaktadır. Ayrıca, regülâtörün tepki vermeden önce meydana gelebilecek yük değişimlerini sabit çıkış gerilimi altında tolere edebilmesi amacıyla da kullanılmaktadır. Görüldüğü üzere Buck dönüştürücünün çıkış kapasitansı 2 farklı göreve sahiptir. Bundan dolayı da 2 farklı hesaplama sürecine sahiptir.

3.1.5.1. Kapasitans filtre hesabı 1

Bobin içerisinde geçen akımın oluşturmuş olduğu akım dalgalanması,

$$\Delta I_C = \frac{V_{\text{ÇKŞ}}}{n * f_{\text{pwm}}} * \frac{1 - D}{L} \quad (318)$$

denklemler ile hesaplanır. Denklemlerdeki ΔI_C kondansatörü şarj eden akım dalgalanmasını temsil eder. Bu akım aynı zamanda kondansatör üzerinde oluşan dalgalı gerilimin de kaynağıdır. Dalgalı akımın oluşturacağı dalgalı kapasitans yükü,

$$\Delta Q_C = \frac{T_{\text{anahtar}}}{n * 2} * \frac{1}{2} * \frac{\Delta I_C}{2} = \frac{\Delta I_C}{8 * n * f_{\text{PWM}}} \quad (319)$$

biçiminde hesaplanır. Değişken Q yükü altında oluşan değişken kapasitans gerilimi,

$$\Delta V_C = \frac{\Delta Q_C}{C} = \frac{\Delta I_C}{8 * n * C * f_{\text{PWM}}} = \frac{V_{\text{ÇKŞ}} * (1 - D)}{8 * C * L * (n * f_{\text{PWM}})^2} \quad (320)$$

biçiminde hesaplanır. Böylelikle çıkış kondansatörünün minimum kapasitesinin hesabı,

$$C = \frac{1 - D}{\frac{\Delta V_C}{V_{\text{ÇKŞ}}} * 8 * L * (n * f_{\text{PWM}})^2} \quad (321)$$

şeklinde hesaplanır.

3.1.5.2. Kapasitans filtre hesabı 2

Bu filtre hesabında kapasitans, yük değişimlerini tolere edebilecek şekilde minimum kapasitans değerine sahip olmalıdır.

$$\frac{di_R}{dt} = \frac{1}{L} * (D_{max} * V_k - V_{\text{çkş}}) \quad (322)$$

$$t_{\text{ramp}} = \frac{\Delta I_R}{n * \frac{di_R}{dt}} \quad (323)$$

Çıkış kapasitansı gerilim düşümü,

$$\Delta V_{\text{çkş}} = \frac{\Delta Q_C}{C} \quad (324)$$

şarj dalgalanma değeri,

$$\Delta Q = n * \frac{di_R}{dt} * t_{\text{ramp}}^2 \quad (325)$$

biçiminde çözümlenir. Çıkış kondansatörü tarafından sağlanması gereken yük miktarı,

$$\Delta Q_C = \Delta Q - \Delta Q_{\text{çıkış}} \quad (326)$$

biçiminde tanımlıdır. $\Delta Q_{\text{çıkış}}$,

$$\Delta Q_{\text{çıkış}} = \Delta I_R * (t_{\text{ramp}} - \frac{1}{2} * t_{\text{adım}}) \quad (327)$$

dır. Son gerilim değişimi,

$$\Delta V_{\text{çıkış}} = \text{MAX} \left[0, \frac{1}{2 * C} * \left(\frac{L * \Delta I_R^2}{n * (D_{max} * V_k - V_{\text{çkş}})} - \Delta I_R * t_{\text{adım}} \right) \right] \text{ olur} \quad (328)$$

Buck dönüştürücünün çıkış kapasitesi aşağıda verilmiş olan kriteri karşılamalıdır.

$$C > \frac{1}{2 * \Delta V_{\text{çıkış}}} \left(\frac{L * \Delta I_{\text{R}}^2}{n * (D_{\text{max}} * V_{\text{k}} - V_{\text{çıkış}})} - \Delta I_{\text{R}} * t_{\text{adım}} \right) \quad (329)$$

3.1.5.3 Sonuç

Çıkış kapasitesinin 2. hesaplama yöntemi incelendiğinde; 1. yöntemin kullanılmasının yeterli olduğu sonucu çıkarılmıştır. Fakat kondansatör kullanımında dikkat edilmesi gereken bir diğer önemli parametre de akım dalgalanmasıdır. Akım dalgalanması da ısı artışına neden olmaktadır. Bu parametre de göz önünde tutularak kondansatör seçimi yapılmalıdır.

3.1.6. Giriş kapasitansı

Giriş kondansatörü (C1), DC bara gerilimini filtreleme amaçlı kullanılmıştır. Bunun sebebi, uygulanacak olan PWM işareti lojik "1" seviyesinin sabit DC bara gerilimine olan eşitliğinin korunma isteğidir. DC giriş geriliminin ideal olarak filtrelenmesi sonrası,

$$I_{\text{giriş}} = \frac{D}{\eta} * I_{\text{çıkış}} \quad (330)$$

eşitliği ile tanımlıdır. Akım dalgalanması görmezden gelinerek, tek anahtardan geçen akım değeri,

$$I_{\text{anahtar}} = n * \frac{I_{\text{çıkış}}}{\eta} \quad (331)$$

biçiminde hesaplanır. Giriş kondansatörü, regülatörün DC akım girişi ile anahtardan çıkan akımı kompanse ederek gerilim değerinin sürekli değeri sabit kalmasını garantilemesi gerekir.

$$dQ_c = (I_{\text{anahtar}} - I_{\text{giriş}}) * D * T \quad (332)$$

Giriş kondansatörünün en düşük değeri aşağıdaki eşitlik ile hesaplanır.

$$C_{\text{giriş}} \geq \frac{dQ_c}{dV_{\text{giriş}}} = \frac{I_{\text{çıkış}}}{\eta * f_{\text{pwm}} * dV_m} * \left(\frac{D}{n}\right)^2 \quad (333)$$

Bu denklemde dV gerilimi, giriş kondansatöründe izin verilen dalgalanma değeridir.

Akım değeri, giriş kondansatörünün seçimindeki en kritik parametredir. Kondansatörün sahip olduğu ESR direncinden dolayı sınırsız bir akım değeri ile kondansatör şarj edilemez.

$$P_c = I_{c_RMS}^2 * ESR \quad (334)$$

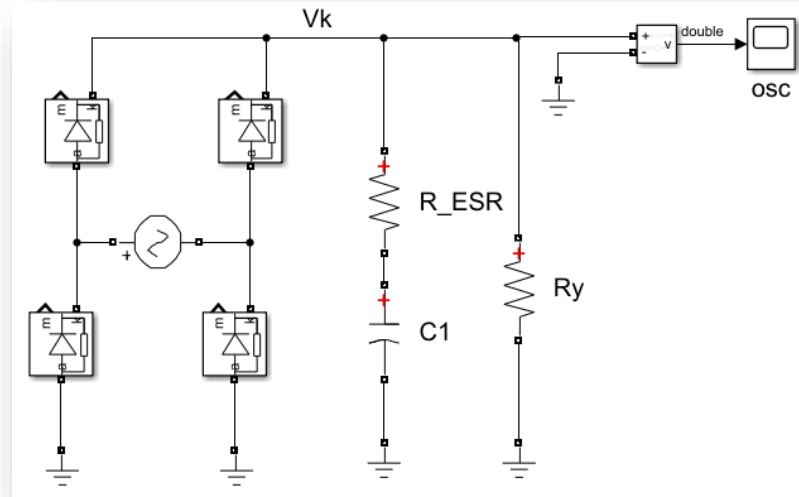
Buck dönüştürücünün tetiklenmesinden dolayı oluşan RMS akımı

$$I_{c_RMS} = \frac{I_{\text{çıkış}}}{\eta} * \sqrt{\frac{D}{n^2} - \frac{D^2}{n}} \quad (335)$$

biçiminde hesaplanır.

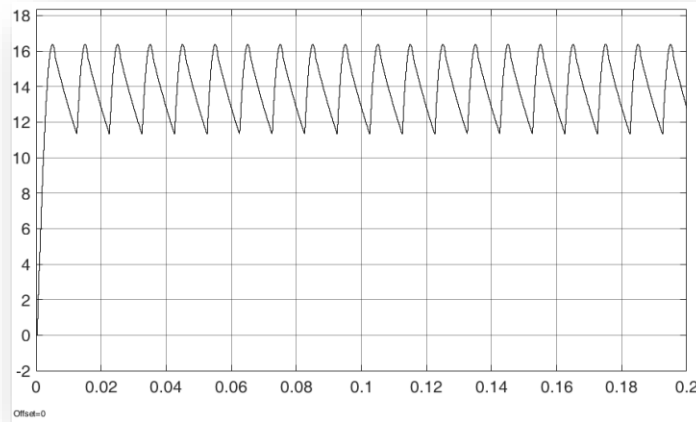
3.1.7. Buck dönüştürücü benzetim ortamı çalışmaları

Buck dönüştürücü tasarımı kaynak gerilimi (Vk, Şekil 3.10.) temini ile başlar. Vk, mutlaka akü gerilimi gibi düz olmalıdır. Aksi takdirde yukarıda verilmiş olan Buck dönüştürücü tasarım denklemleri geçerliliğini kaybeder. Vk için doğrultma gerilimi uygun değildir. Bunun sebebi amper seviyesinde akım çekilmesi durumunda sistemde kullanılacak olan Cgiriş (C1, Şekil 3.10.) kapasitans sığası aşırı büyük ve sayı olarak da yüksek adetli olacaktır, Şekil 3.11.



Şekil 3.11. V_k geriliminin doğrultma yöntemi ile elde edilmesi.

Şekilde verilmiş olan devrede AC 50Hz'lik gerilim tepe değeri 18V olup tam dalga doğrultma yöntemine dayalı DC gerilime dönüştürülerek V_k gerilimi elde etmek amaçlanmaktadır. Yük direnci olan R_y'nin değeri yüksek akım çekilebilmesi amacıyla 2ohm ($18V/2ohm=9Amper$) seçilmiştir. Tasarlamış olduğumuz akım kaynağı, 0Amper ile 9.999Amper aralığında güvenilir ve öngörülen dalga biçimlerinde akım sağlanabilmesi ancak ve ancak elde edilecek V_k gerilim genliğinin zamanla değişmemesine bağlıdır. Fakat ne yazık ki, C1 kondansatör sığası 10.000µF (10adet 1000µF) seçilse bile istenilen hassasiyette V_k gerilim dalga şekli elde edilememektedir, Şekil 3.12.



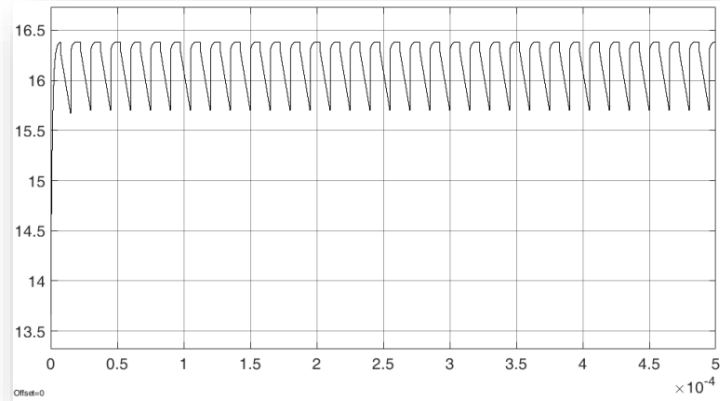
Şekil 3.12. 10000µF ile filtrelenmiş ve yük direnci 2ohm olan devrenin gerilim dalga şekli.

Görüldüğü gibi elde edilen doğrultulmuş V_k gerilim dalgalanması yüksektir. Bundan dolayı da doğrultma yöntemine dayalı elde edilecek gerilim, yüksek pik değerleri yüzünden V_k gerilimi olarak kullanılamayacaktır. Doğrultma yerine entegre bir sistem olan smps güç kaynaklarının (anahtarlama güç kaynağı- Switch Mode Power Supply) V_k olarak kullanımı daha uygundur, Şekil 3.13.



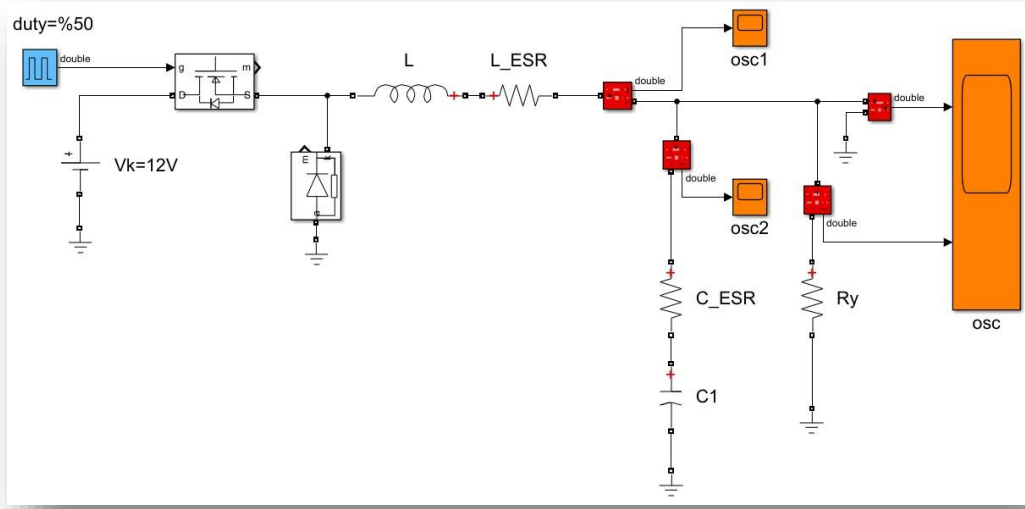
Şekil 3.13. SMPS güç kaynağı.

Bu gerilim kaynağı ile güvenilir bir V_k gerilimi elde etmek mümkündür. Bu entegre sistemleri pals tabanlı ve yüksek frekansta çalışır bundan dolayı çıkış gerilim dalga şekli %2'nin altında değişim göstermiştir, Şekil 3.14.



Şekil 3.14. 20ohm'luk yük altında SMPS güç kaynağı gerilim dalga şekli.

Buck tasarım kriterleri çerçevesinde Buck dönüştürücü güç elektroniği devresi Matlab-Simulink ortamında tasarlanmıştır, Şekil 3.15.



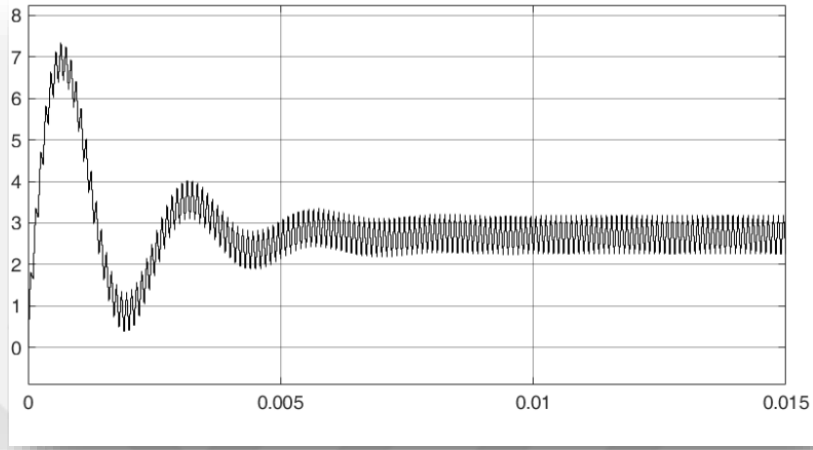
Şekil 3.15. Buck dönüştürücü Matlab-Simulink tasarımı.

Vk gerilim dalga şekli zamanla değişmediği kabul edilerek simülasyonlar gerçekleştirilmiştir. Devre elemanlarına ait parametreler Tablo 3.1'de verilmiştir.

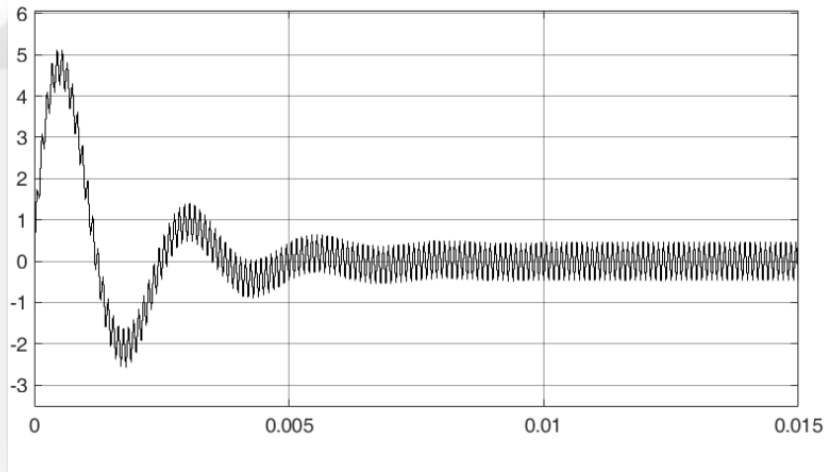
Tablo 3.1. Buck dönüştürücü parametreleri.

Parametre	Değer	Birim
f_pwm	10	KHz
duty	%50	
Vk	12	Volt
L	330	μ H
L_ESR	10	m Ω
C1	470	μ F
C_ESR	10	m Ω
Ry	2	Ω

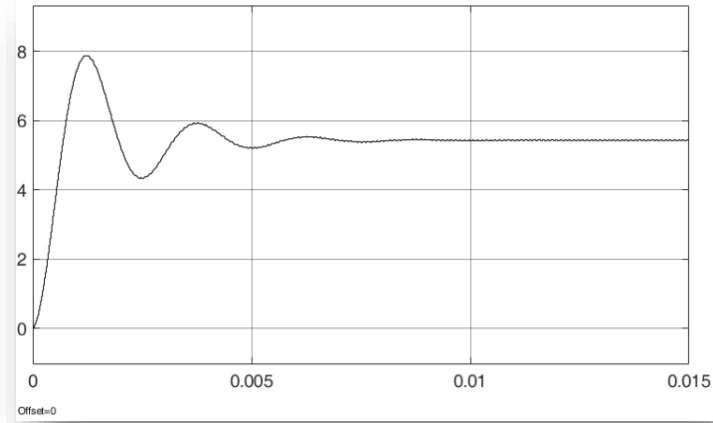
Tablo 3.1.'de verilmiş olan parametreler çerçevesinde, Buck dönüştürücü devre elemanlarına ait bobin akımı, kondansatör akımı ve yük çıkış geriliminin simülasyon sonuçları aşağıda verilmiştir.



Şekil 3.16. Buck dönüştürücü 2ohm yük altında, bobin akımı dalga şekli.

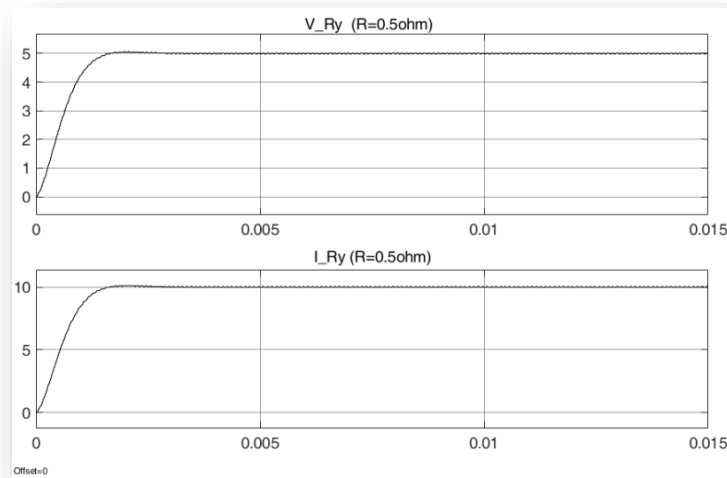


Şekil 3.17. Buck dönüştürücü 2ohm yük altında, C2 akımı dalga şekli.

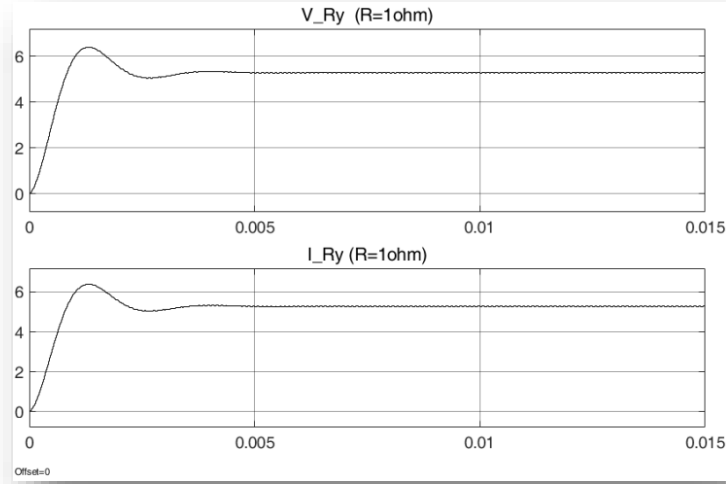


Şekil 3.18. Buck dönüştürücü 2ohm yük altında, çıkış gerilimi dalga şekli.

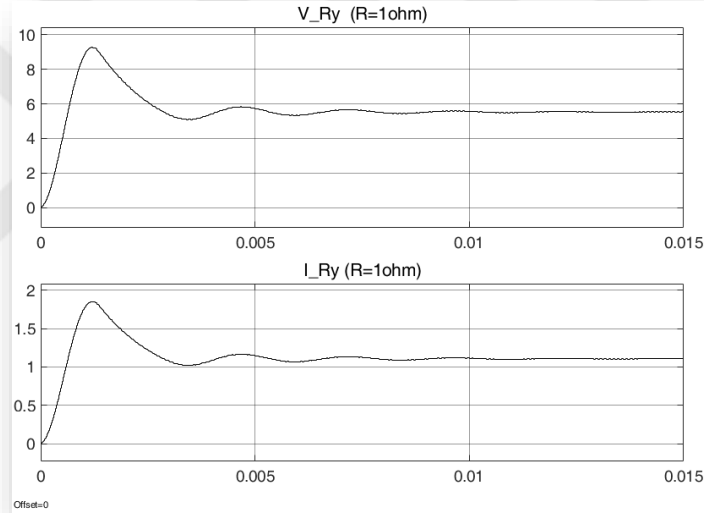
Buck dönüştürücü çıkış gerilimi öngörülen limitler içerisinde olmadığı gözlemlenmiştir, Şekil 3.18. PWM duty değeri %50 ve V_k gerilim değeri de 12V seçildiği için Buck dönüştürücünün çıkış gerilimi 6V öngörülür. Fakat çıkış gerilim değeri 6V'tun altında oluşmuştur. Ayrıca gerilimin geçici hal davranışı da istenmeyen türdendir. Yani Buck dönüştürücü gerilim çıkışı kabul edilemeyecek seviyede gerilim aşım seviyesine sahiptir. Bununla beraber, Yük akımında meydana gelecek değişimler sonrası Buck dönüştürücü devresinin farklı dinamik davranışlarına sahip çıkış gerilimleri oluşur, Şekil 3.19 3.20 ve 3.21.



Şekil 3.19. Yük akımı 10A için Buck dönüştürücü çıkış akımı ve gerilimi dinamik davranışı.



Şekil 3.20. Yük akımı 5A civarı için Buck dönüştürücü çıkış akımı ve gerilimi dinamik davranışı.



Şekil 3.21. Yük akımı 1A civarı için Buck dönüştürücü çıkış akımı ve gerilimi dinamik davranışı.

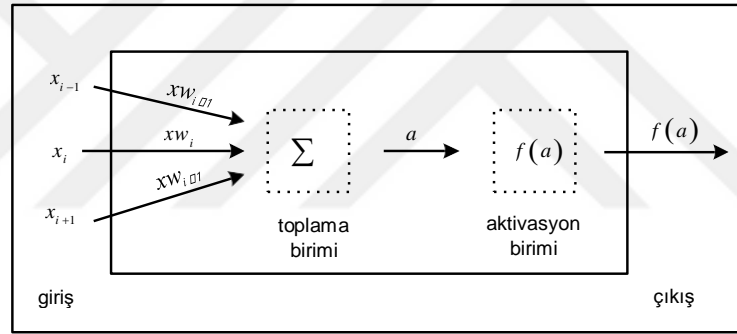
Matlab-Simulink ortamında gerçekleştirilmiş olan simülasyonlar göstermiştir ki Buck dönüştürücü devresi yalnız şekilde kullanılarak amaçlanan çok fonksiyonlu akım dalga üretici tasarlanamaz. Buck dönüştürücü devresi mutlaka bir kontrol algoritması ile desteklenmelidir ki yük değişimleri süresince öngörülen akım dalga formlarına yine öngörülen hata oranları ile üretilebilsin. Bu tez çalışmasında Yapay Sinir Ağı (YSA) algoritmasından faydalanılmıştır. YSA algoritması bu tez çalışmasında, ayrık zaman non-lineer adaptif bir kontrol algoritması olarak Buck dönüştürücü çıkış akımının

öngörülen dinamik davranışa sahip olacak şekilde Buck devresine adaptasyonu gerçekleştirilmiştir.

3.2. Yapay Sinir Ağı ve Temel Nöron Dinamiği

Yapay sinir ağları (YSA) inşa etmeye yönelik ilk girişimler, canlıların beyinlerinde yer alan milyarlarca sinirin modelleme girişimine dayanır. Süreç esnasında, YSA istatikselsel bir veri kümesine sahip olursa, bu veri kümesinden doğruluğu yüksek sonuçlar elde edebileceği anlaşılmıştır.

YSA sisteminde işlem yapan en küçük parçacık nöron olarak isimlendirilir. Nöron, genellikle birden fazla girdiye ve yalnızca bir çıktıya sahip işlem birimidir, Şekil 3.22.



Şekil 3.22. Nöron mekanizması.

Burada gerçekleştirilen işlemler sırasıyla, girişler bir katsayı (ağırlık, weight, w) ile çarpılır ve ardından hesaplanan yeni girişler birbirleri ile toplanır, Denklem 3.36.

$$a = \sum_{i=1}^{\text{tüm girişler}} w_i x_i \quad (336)$$

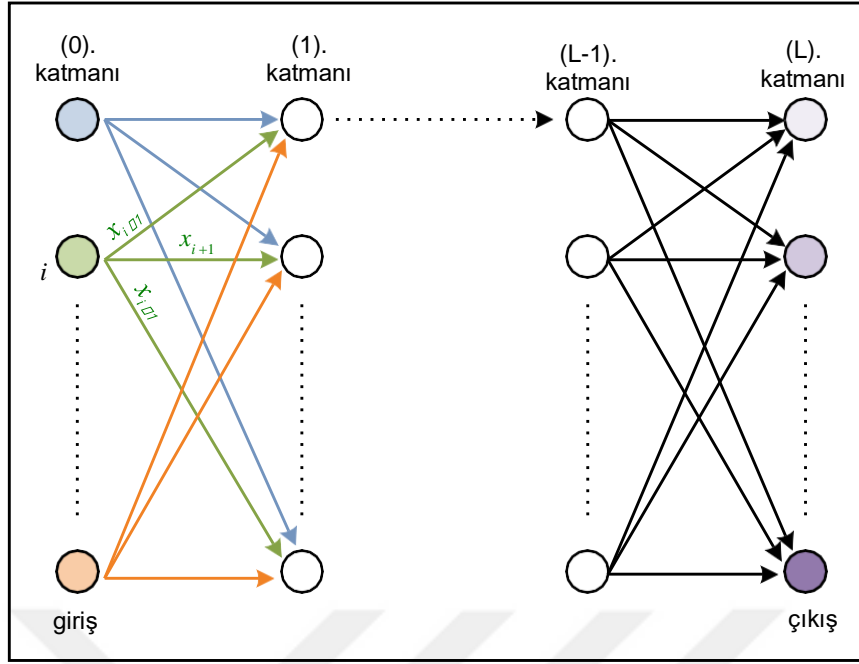
Elde edilen toplam (a), aktivasyon fonksiyonu adı verilen doğrusal veya doğrusal olmayan bir fonksiyon giriş değişkeni olarak işlem görür. Fonksiyon çıktısı, nöronal yapının çıkış değeri olur, Denklem 3.37.

$$\text{çıkış} = f\left(\sum_{i=1}^{\text{tüm girişler}} w_i x_i\right) = f(a) \quad (337)$$

Genel olarak YSA, nöronları katmanlara yerleştirip çıktılarını birbirine bağlayarak oluşturulur. Böylece her katman çıkışı bir sonraki katman giriş değişkeni olarak görev yapar, Şekil 3.23.

Şekil 3.23.'de yansıtılmış olan YSA bağlantı türü aynı zamanda "feedforward" olarak da isimlendirilir. Feedforward yapıda, çıkış katmanından giriş katmanına doğru geri dönüş verisi aktarılmaz. Çıkış katmanından giriş katmanına doğru geri dönüş verisi aktaran YSA türleri "recurrent" olarak isimlendirilir (Qu ve Zhou, 2010; Lange vd., 1999; Yuen vd., 2019; Lukic vd., 2016; Song vd., 2008).

YSA sistemi içerisinde, ağırlıkların değerleri sürekli bir değişim içerisinde. Ağırlıklar, özel matematiksel algoritmalar ile çözümlenirler ve sonuç olarak YSA kurgulaması gereken giriş-çıkış ilişkisini optimum bir şekilde sağlamış olur (Ullah vd., 2020; Shen vd., 2014; Lin vd., 2019; Gaoqi vd. 2013). Ağırlık katsayılarının hesaplanma sürecine eğitim denir. Eğitim süreci, tecrübe çerçevesinde elde edilen giriş ve çıkış veri kümesine (learning set veya training set) dayanır. Bazı öğrenme süreçleri için bu veri kümeleri kullanılmadan da koşturulur. Veri kümesine dayalı eğitim denetimli (supervised); veri kümesine dayalı olmayan eğitim süreci de denetimsiz (unsupervised) eğitim olarak adlandırılır.



Şekil 3.23 YSA sistemine ait katman ve nöron bağlantısı.

3.2.1. Nöron fonksiyonları

Nöronlar fonksiyon olarak görev yapan yapılardır. Nöronlar, $t=t_1$ anı için sınırlandırılmamış giriş verisine ($x(t_1)$) dayalı sınırlandırılmış fonksiyon ($f(x(t))$) değerini çıkış olarak elde ederler. $F(x(t))$ fonksiyonuna aktivasyon fonksiyonu veya sinyal fonksiyonu adı verilir.

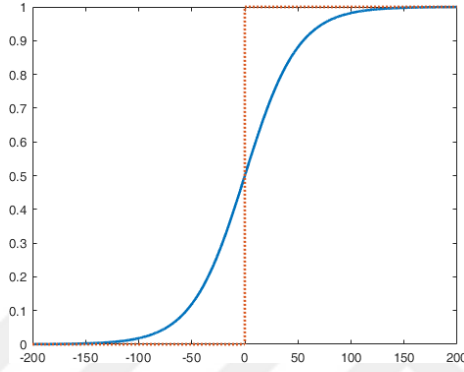
Çok kullanılan sinyal fonksiyonlarından biri "lojistik sinyal fonksiyonu" diğer bir adı ise "sigmoid" olup,

$$f(x(t)) = f(a) = \frac{1}{1 + e^{-ca}} \quad (338)$$

biçiminde tanımlıdır. Bu denklemde yer alan c parametresi, " $c > 0$ " koşulu için "lojistik sinyal fonksiyonu" sigmoidal ve çıkış değeri kesinlikle artan bir fonksiyondur. Fonksiyon türevinin sıfırdan büyük oluşu matematiksel olarak da kesinlikle artan bir fonksiyon olduğunun kanıtıdır, Denklem 3.39.

$$f' \equiv \frac{df}{da} \equiv cf(1-f) > 0 \quad (339)$$

Şekil 3.24.'de eşik fonksiyonu (kesikli çizgi) ile beraber lojistik fonksiyon çıkışı yansıtılmıştır.



Şekil 3.24. Lojistik fonksiyon (mavi) çıkış grafiği, $c=0.04$, $a=[-200,+200]$.

"c" ile indekslenen "lojistik sinyal fonksiyonu" ailesi, eşik fonksiyonuna asimptotik olarak $c \rightarrow +\infty$ için yaklaşır. Aktivasyon fonksiyon çıkışı $a>0$ için birim değere; $a<0$ için ise 0 değerine yaklaşır. Süreksizlik, $c=0$ için oluşur. "c"nin sıfır değeri için fonksiyon çıkışı eşik çıkışına eşitlenir. Bu istenmeyen durum genellikle çok katmanlı ve yoğun nöronlu YSA sistemlerinde görülür.

Uygulamada sıklıkla kullanılan aktivasyon fonksiyonları sırasıyla aşağıda tanıtılmıştır.

3.2.1.1. Lojistik (Logistic, Sigmoid)

Matematiksel ifadesi,

$$f(x(i)) = f(a) = \frac{1}{1 + e^{-ca}} \quad (340)$$

biçimindedir. Burada $c>0$ ve $c=sbt$ bir değerdir. Aktivasyon fonksiyon türevi,

$$f' \equiv \frac{df}{da} \equiv cf(1-f) > 0 \quad (341)$$

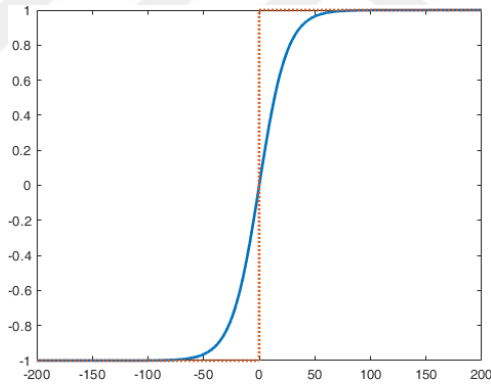
şeklinde bir matematiksel ifadeye dönüşür. Fonksiyon monoton artandır ve değer aralığı (0,1) dir, Şekil 3.24.

3.2.1.2. Hiperbolik-Tanjant (Hyperbolic Tangent)

Matematiksel ifadesi,

$$f(a) = \tanh(ca) = \frac{e^{ca} - e^{-ca}}{e^{ca} + e^{-ca}} \quad (342)$$

biçimindedir. Burada $c>0$ ve sabit bir değerdir, Şekil 1.4.



Şekil 3.25. Hiperbolik tanjant fonksiyon (mavi) çıkış grafiği, $c=0.04$, $a=[-200,+200]$.

Aktivasyon fonksiyon türevi,

$$f' \equiv \frac{df}{da} \equiv cf(1-f) > 0 \quad (343)$$

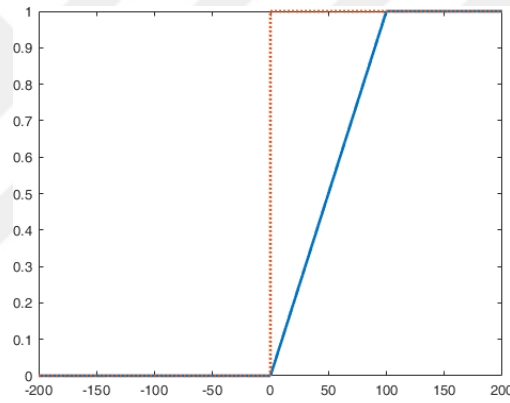
biçimindedir. Fonksiyon monoton artandır ve değer aralığı (-1,1) dir.

3.2.1.3. Eşik (Treshold)

Fonksiyonun matematiksel ifadesi,

$$f(a) = \begin{cases} 1 & \text{eğer, } a \geq \frac{1}{c} \\ 0 & \text{eğer, } a < 0 \\ \frac{1}{c} & \text{diğer koşullarda (} x \in [0, \frac{1}{c}) \end{cases} \quad (344)$$

biçimindedir. Fonksiyon içerisindeki c parametresi, $c > 0$ ve $c = \text{sbt}$ koşulunu sağlamalıdır. Fonksiyon grafiği Şekil 3.26.'de yansıtılmıştır.



Şekil 3.26. Eşik fonksiyon (mavi) çıkış grafiği, $c=0.01$, $a=[-200,+200]$.

Fonksiyon türevi,

$$f'(a) = \frac{df}{da} = \begin{cases} 0 & \text{eğer, } a \in (-\infty, 0) \\ c & \text{diğer koşullarda} \end{cases} \quad (346)$$

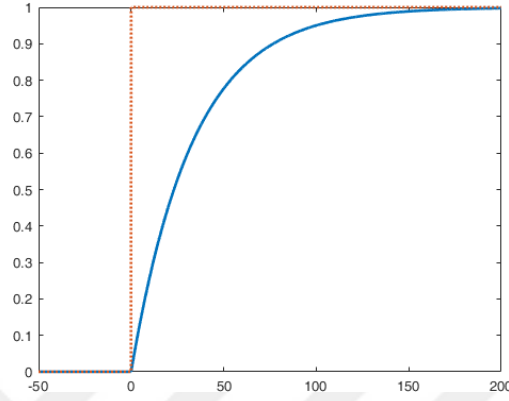
biçiminde tanımlanır.

3.2.1.4. Üstel dağılım (exponential distribution)

Fonksiyonun matematiksel ifadesi,

$$f(a) = \max(0, 1 - e^{-ca}) \quad (347)$$

biçiminde tanımlıdır. Denklemde yer alan c parametresi, yine $c > 0$ ve $c = \text{sbt}$ bir değerdir. Fonksiyon grafiği Şekil 3.27.'de yansıtılmıştır.



Şekil 3.27. Eşik fonksiyon (mavi) çıkış grafiği, $c=0.03$, $a=[-50, +200]$.

Fonksiyonun türevi,

$$f'(a) = \frac{df}{da} = ce^{-ca} \quad (347)$$

dir. $a > 0$ için eşik üstü sinyaller monoton artandır ve fonksiyon çıkışları $(0, 1)$ aralığındadır.

3.2.1.5. Oransal (ratio polynomial)

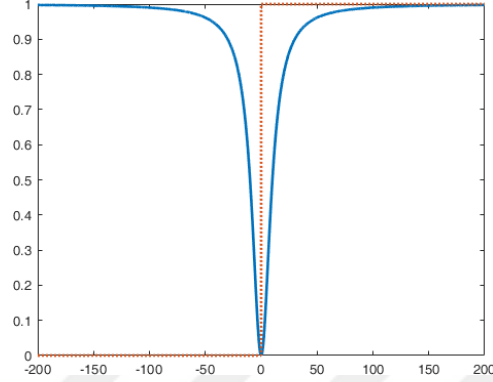
Fonksiyonun matematiksel ifadesi,

$$f(a) = \max\left(0, \frac{a^n}{c + a^n}\right), n \rightarrow 1 \quad (348)$$

biçimindedir ($c > 0$ ve $c = \text{sbt}$). Fonksiyon grafiği Şekil 3.28.'de yansıtılmıştır.

Fonksiyonun türevi,

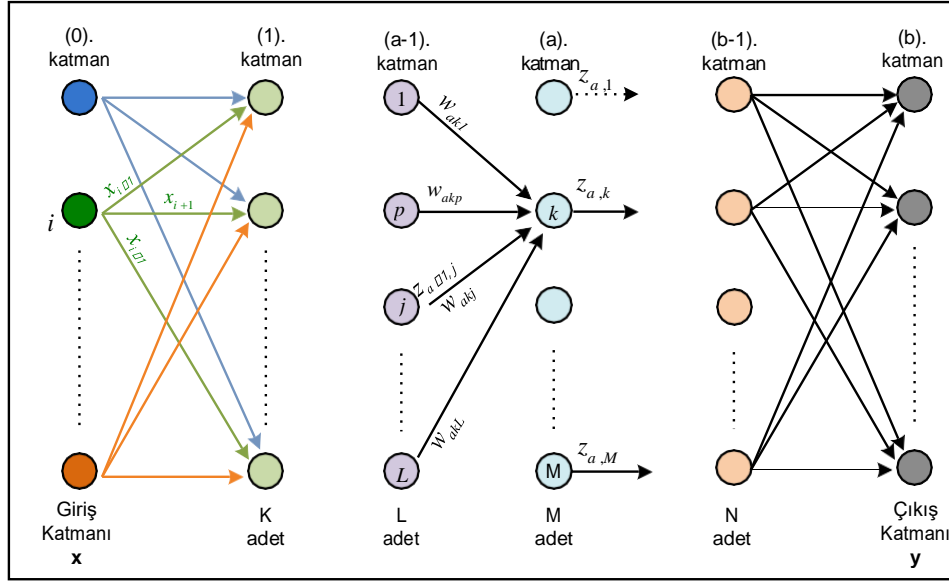
$$f' = \frac{df}{(c+a^n)^2} = \frac{-cna^{n-1} da}{(c+a^n)^2} \quad \text{olur.} \quad (3.49)$$



Şekil 3.28. Oransal dağılım fonksiyon (mavi) çıkış grafiği, c=100, n=2, a=[-200,+200].

3.2.2. YSA ağırlıklarının optimizasyonu

YSA, birden fazla nöron katmanından oluşmaktadır. Birinci katmana giriş katmanı adı verilir. Bu katman bizim örneğimizde 0. katman olarak tanımlanmıştır, Şekil 3.29. 0. katman, giriş olarak isimlendirilen katman 1'e giriş verilerini dağıtan katman olarak tanımlanır. Katman 0'da herhangi bir işlem yapılmamaktadır. Son katman, çıkış katmanıdır. YSA girişine uygulanmış olan giriş verileri, katmanlar arasında matematiksel işlemlere maruz kalarak ilerler ve son katman olan çıkış katmanında hedef veri olarak son değer elde edilir. Giriş katmanı ile çıkış katmanı arasında kalan katmanlar gizli katman olarak isimlendirilir. Bu katmanların sayısı ve her katman içerisinde yer alan nöronların sayısı ile ilgili bir kural yoktur. Fakat YSA, minimum katman sayısı ve katmanlar içerisinde minimum nöron sayısı gözetilerek kurgulanır. Her katman matematiksel bir işlem yükü getirdiği için YSA gizli katmanlarının sayısı ve nöron miktarı gözetilerek mümkün olduğu kadar sade bir yapı kurulmaya çalışılır. Ağ yapısı, basit ileri beslemeli bir yapıdır, Şekil 3.29.



Şekil 3.29. YSA, giriş, çıkış ve gizli katman yapısı.

Her nöron, önceki katmandaki tüm nöronların çıktılarını birer katsayı (w) ile çarparak girdi olarak alır. Şekil 3.29. göz önünde bulundurulursa,

z_{ak} : Katman (a)'nın k nöronun çıkışı.

w_{akj} : Katman (a)'da yer alan k nöronu ile katman (a-1)'de yer alan j nöronu arasındaki katsayı değişkeni.

x : YSA giriş vektörü.

y : Hedef çıkış vektörü.

x_i : Giriş vektörüne ait i . eleman.

K, L, M, N : Her katmanına ait nöron sayısı.

biçiminde tanımlanır.

3.2.2.1. YSA ileri yön hesaplama (İleri yayılım)

Her bir nöron, bir önceki katman çıkışında yer alan nöron çıkışlarını bir ağırlık katsayısı ile çarparak toplar, Denklem 3.50.

$$z_{ak} = f\left(\sum_{i=1}^L w_{aki} * z_{a-1i}\right) \quad (350)$$

0. katmanın göz ardı edilmesi durumunda her katman için oluşturulacak olan ağırlık matrisi aşağıda verilmiştir, Denklem 3.51.

$$W_a = \begin{bmatrix} w_{a11} & \dots & w_{a1L} \\ \vdots & \ddots & \vdots \\ w_{aM1} & \dots & w_{aML} \end{bmatrix} \quad (351)$$

Bir önceki katman nöron çıkış değişkenlerine ait vektör Denklemi 3.52'deki gibi düzenlenebilir.

$$Z_{a-1}^T = (z_{a-11} \quad \dots \quad z_{a-1L}) \quad (352)$$

Katman çıkış hesabı bir önceki katman çıkış vektörü ve bir önceki katman çıkışında yer alan ağırlık matrisi göz önünde bulundurulursa,

$$Z_a^T = f(a^T) = (f(a_{a1}) \quad \dots \quad f(a_{aM})) \quad (353)$$

biçiminde yeni katman çıkışları hesaplanmış olur. Denklemde yer alan a_a , denklem 3.54'de tanımlanmıştır.

$$a_a = W_a z_{a-1} \quad (354)$$

3.2.2.2. Geri yayılım (YSA eğitimi)

YSA eğitim süreci, giriş verisi ile hedef çıkış verisi arasında kalan ve nöron yapısını oluşturan ağırlık katsayı değerlerinin minimum çıkış hatası oluşturacak şekilde

optimize edilme süreci olarak adlandırılır. Hedef çıkış veri kümesi ile hesaplanan çıkış veri kümesi arasında oluşan hataların karesel toplamı dikkate alınarak (çoğunlukla) ağırlıklar güncellenir. Giriş x vektör elemanları için hesaplanan y çıkış vektör elemanları ile hedef çıkış vektör elemanları arasındaki toplam karesel hatanın denklemi aşağıda verilmiştir.

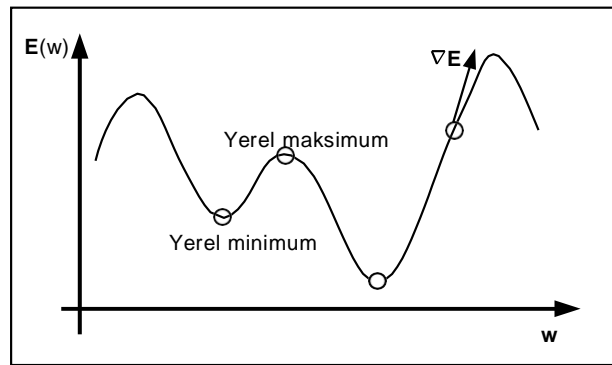
$$E(\mathbf{W}) = \frac{1}{2} \sum_{i=1}^M [z_{bi}(x) - y_i(x)]^2 \quad (355)$$

z_{bi} , katman (b)'nin yani çıkış katmanındaki i . nöron çıkışıdır. Çıkış nöron değerini, giriş veri vektörünün tüm elemanları etkileyecektir. Bundan dolayı da $z_{bi}=z_{bi}(x)$ şeklinde notasyon verilmiştir.

Ağ içerisindeki ağırlıkların kademeli olarak yeni değerleri hesaplanır. ∇E gradient vektörü,

$$\nabla E = \left\{ \frac{\partial E}{\partial w_{aji}} \right\} \quad (356)$$

biçiminde, karesel ortalama hatasının yerel maksimumların yönünü belirler, Şekil 3.30.



Şekil 3.30. ∇E gradient vektörünün, hata yerel min ve max noktalarını belirleme anı.

Ağırlık kestirimi ayrık zaman t de gerçekleştirilmektedir. $t+1$ zaman dilimi için ağırlık kestirimi t anındaki ağırlıklar kullanılarak işlem yapılır, Denk. 3.57.

$$w_{ij}^{(t+1)} = w_{ij}^{(t)} - \eta \frac{\partial E(\mathbf{W})}{\partial w_{ij}} \Big|_{\mathbf{w}^{(t)}} = w_{ij}^{(t)} - \eta \sum_{r=1}^R \alpha_r \frac{\partial E_r(\mathbf{W})}{\partial w_{ij}} \Big|_{\mathbf{w}^{(t)}} \quad (357)$$

Kullanılan η parametresi, $\eta > 0$ ve $\eta = s \cdot t$ olacak şekilde seçilir. η katsayısına öğrenme katsayısı adı verilir. Öğrenme katsayısı, öğrenme sürecinin hızını ve kalitesini artırmak için kullanılır.

Matris düzeyde Denklem 3.57, aşağıdaki basit forma dönüştürülebilir, Denk. 3.58.

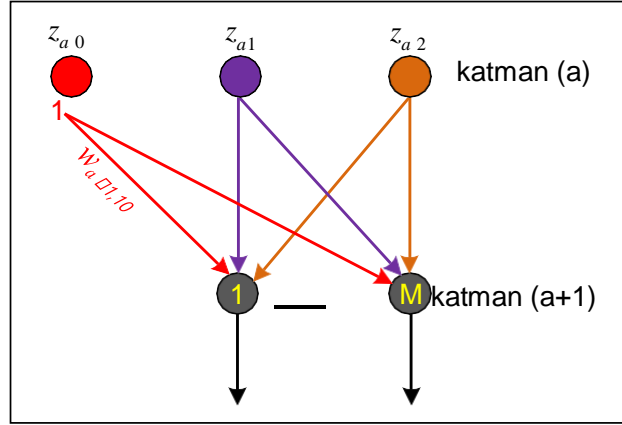
$$\mathbf{W}^{(t+1)} = \mathbf{W}^{(t)} - \eta \nabla E \quad (358)$$

3.2.2.3. Bias

Bazı problemler denklem 3.57'de verilen yöntemle çözülemeyebilir. Yani ağırlıkların optimum değerleri bulunamayabilir. Bu tür problemler için ;

$$z_{ak} = f(w_{ak0} + \sum_{i=1}^L w_{aki} * z_{a-1i}) \quad (359)$$

biçiminde ileri hesaplamada yeni bir nöron ilave edilir. Bu nöronun değeri 1'e sabitlenir. Bu nöron, z_{ak} eşitliğinde tek başına bırakılmıştır. Görsel formda bias,



Şekil 3.31. Nöron bias ı ilave edilmiş katman.

biçiminde ağ yapısına katılmış olur. Nöron ağırlık matrisi ve çıkış vektörü için gerekli düzenleme,

$$\vec{z}_a = (1 \quad z_{a1} \quad \dots \quad z_{aM}) \quad (360)$$

$$\vec{W} = \begin{pmatrix} w_{a10} & \dots & w_{a1L} \\ \vdots & \vdots & \vdots \\ w_{aM0} & \dots & w_{aML} \end{pmatrix} \quad (361)$$

biçiminde gerçekleştirilir. Biaslanmış ağırlık matrisi ile çıkış vektörleri kullanılarak katman çıkış hesabı,

$$\vec{z} = f(a_a) = f(\vec{W} \vec{z}_{a-1}) \quad (362)$$

şeklinde tanımlanır. \vec{W} ile ilişkilendirilmiş ∇E gradient hata matrisi

$$\vec{E}_a = \begin{pmatrix} \frac{\partial E}{\partial w_{a10}} & \dots & \frac{\partial E}{\partial w_{a1L}} \\ \vdots & \vdots & \vdots \\ \frac{\partial E}{\partial w_{aM0}} & \dots & \frac{\partial E}{\partial w_{aML}} \end{pmatrix} \quad (363)$$

bu şekilde son değerini alır.

3.2.2.4. Momentum

Standart geri yaymada açıklanan ağırlık uyarlaması küçük sapmalara karşı çok hassastır. Hata yüzeyinde küçük bir "tümsek" görünse, geriye yayılım algoritması bunun üzerinden atlayamaz ve yön değiştirecektir. Öğrenme sürecinde önceki uyarlamalar dikkate alınarak bu durum önlenabilir, Denk. 2.15.

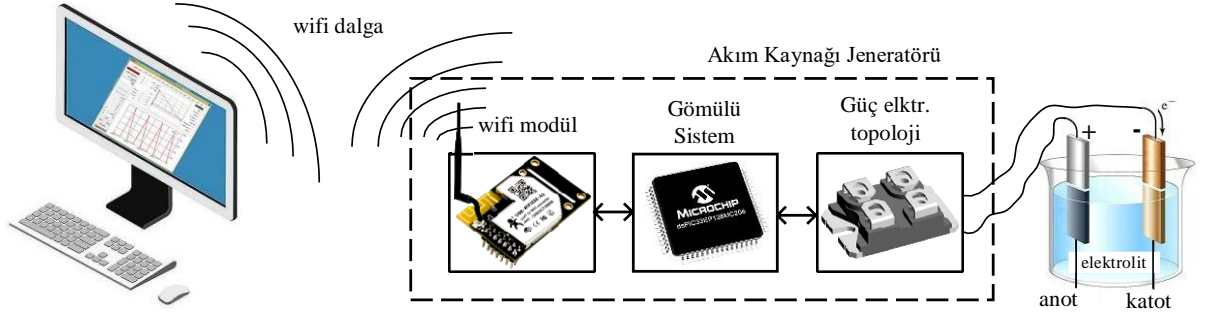
$$\Delta W(t) = W(t+1) - W(t) = -\eta \nabla E_{w(t)} + \mu \Delta W(t-1) \quad (364)$$

Geri yayılım algoritmasının bu şekilde işletilmesi, momentum tabanlı geri yayılım ismini alır ve denklemde yer alan $\mu \in [0,1)$ parametresi de momentum adını alır.

3.3. Akım Kaynağı Elektronik Devre Yapısı

Yapılan çalışmada, mühendislik alanında kullanılan temel işaretleri elektrik akım dalgalarına dönüştürebilen yeni bir akım üreteç sistemi tasarlanmıştır. Akım kaynağı donanımı, Güç Elektroniği devre topolojileri olan Buck Converter ve H-Bridge kullanılarak gerçekleştirilmiştir. Öngörülen akım dalga biçimlerini üretmek için, doğrusal olmayan bir kontrol algoritması olan Yapay Sinir Ağı hibrit güç elektroniği sistemine uyarlanmıştır. Ardından, yeni nesil akım üreteci sistemi kullanılarak elektrodepozisyon tabanlı kaplama deneyleri gerçekleştirilmiştir.

PC ve PRC Elektrodepozisyon kaplama yöntemlerinin, DC kaplama yöntemine göre daha başarılı oldukları çeşitli araştırma makalelerinde sunulmuştur. Bu tez çalışmasında, bu üç elektrik akımı dalga formuna ek olarak, ilave farklı dalga formları ile akımların kaplama parametreleri üzerindeki değerlerinin iyileştirilip iyileştirilemeyeceği sorusunun cevabı araştırılmıştır. Bunun için bilgisayar kontrollü akım kaynak donanımı geliştirilmiştir. Geliştirilen akım kaynağı sisteminin teknolojik altyapısı, bilgisayar yazılımı, donanım kontrol yazılımı, güç elektroniği devre topolojileri ve doğrusal olmayan YSA tabanlı kontrol bölümlerinden oluşmaktadır, Şekil 3.32.



Şekil 3.32. Akım kaynağı sistemine ait teknolojik alt yapı.

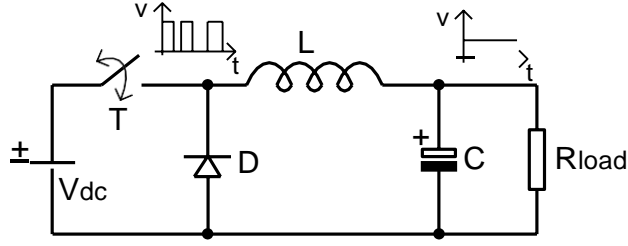
Aşağıda sırasıyla verilmiş olan bölümler ,gerçekleştirilmiş akım kaynağı sistemine ait tasarım süreci çalışmasını özetlemektedir.

3.3.1. Güç elektroniği devre topolojisi

Kaplama işleminde elektrolitin direnci (R_{load}) $m\Omega$ mertebesindedir. Akım kaynağı sistemi, 10 ampere kadar akım iletebilecek şekilde tasarlanmıştır. Cihazın üretebileceği en yüksek akım değeri ve düşük elektrolit direnci dikkate alınarak, jeneratörün bara gerilim değerinin Denklem 3.65 'de verildiği gibi DC 10V olması yeterli olduğu ortaya çıkmaktadır.

$$V = I * R \quad (3.65)$$

Kaplama işleminde, elektrolite uygulanacak potansiyel fark (gerilim, V), elektrolit içerisinde elektrik akımına sebep olacaktır. Elektrolit içerisinde akacak olan akım şiddeti ile kullanıcı tarafından belirlenecek olan referans akım şiddeti, birbiriyle eşleşecek şekilde elektrolite uygulanan gerilim ayarlanmalıdır. Bu koşulu karşılamak için ise bara geriliminin ihtiyaç duyulan daha düşük gerilim seviyelerine dönüştürülmesi gerekir. Bu nedenle, bara geriliminden daha düşük voltaj seviyelerinde yük direncine enerji aktarmalıdır. Yüksek verimlilikle güç aktarımı için güç elektroniği devre topolojilerinden olan Buck dönüştürücü tasarımı seçilmiştir, Şekil 3.33.



Şekil 3.33. Buck dönüştürücü güç elektroniği devre topolojisi.

Bara gerilimi, V_{dc} ile gösterilen güç kaynağıdır, Şekil 3.33. Anahtar, enerji iletimini ve belirli bir frekansta kesmeyi sağlayabilen uygun bir yarı iletkenidir. Bobin ve kondansatör, anahtarlama çıkışında oluşan kare gerilim dalgasını filtreleyen ve R_{load} üzerindeki kesintisiz gerilim ve dolayısıyla da kesintisiz akım oluşumuna sebep olan pasif elektronik devre bileşenleridir. Devrede yer alan R_{load} , elektrolitin direncini temsil eder, Şekil 3.32.

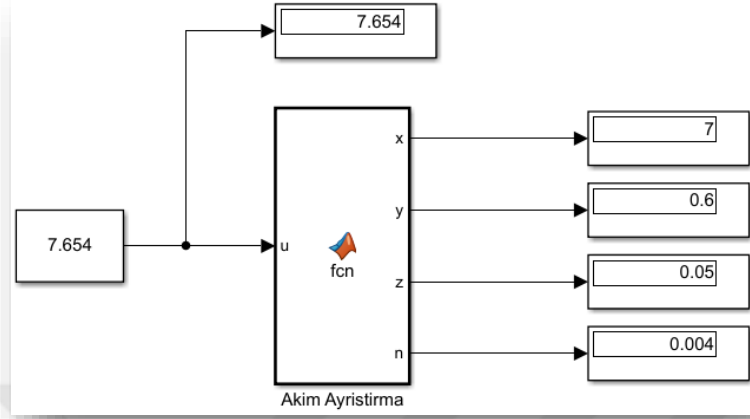
Akım kaynağı sistemine ait akım çıkış aralığı $-9.999A$ ile $+9.999A$ arasındadır. Akım hassasiyeti $0,001$ amperdir. Referans akım ile üretilen akım arasındaki fark yaklaşık $\% 1$ civarındadır. Yüksek akım aralığı ve yüksek akım hassasiyeti yakalanabilmesi amacıyla referans akım 4 parçaya ayrıştırılmıştır. Bu 4 parça için de 4 farklı akım devresi tasarlanmıştır.

Akım kaynağı sistemine ait çıkış akımı harf ile temsil edilecek olursa, X.YZN Amper biçimini alır. Bu yazım tarzında görüleceği üzere her harf bir akım aralığını temsil edecektir, Tablo 3.2.

Tablo 3.2. X.YZN harflerinin temsil ettiği akım aralığı.

Harf	Akım Aralık	Birim
X	1-9	Amper (A)
Y	100-900	mili Amper (mA)
Z	10-90	mA
N	1-9	mA

Referans akım, harf aralıklarına dayalı parçalanıp, ayrı ayrı buck dönüştürücü akım sağlayıcı Buck devrelerine aktarılır, Şekil 3.34.



Şekil 3.34. Ayırılmış referans akım.

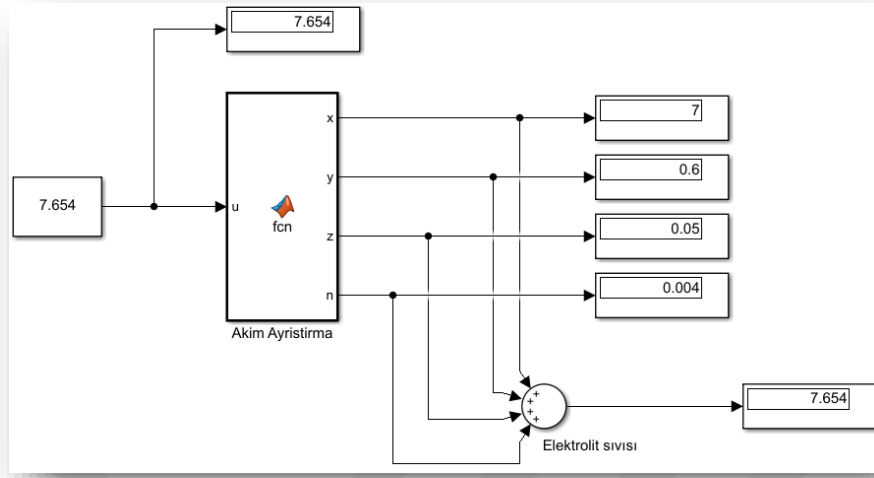
Akım ayırma Matlab-Function blok içeriği aşağıda yansıtılmıştır, Şekil 3.35.

```
Akım Ayırma* x +
1 function [x,y,z,n]= fcn(u)
2
3 a=u;
4 b=a;
5
6 c=mod(b,1);
7 a=a-c;
8 x=a;
9 xa=a;
10
11 a=b-a;
12 a=a*10;
13 c=mod(a,1);
14 a=a-c;
15 y=a/10;
16 ya=a;
17
18 a=b-xa-ya/10;
19 a=a*100;
20 c=mod(a,1);
21 a=a-c;
22 z=a/100;
23 za=a;
24
25 a=b-xa-ya/10-za/100;
26 n=a;
```

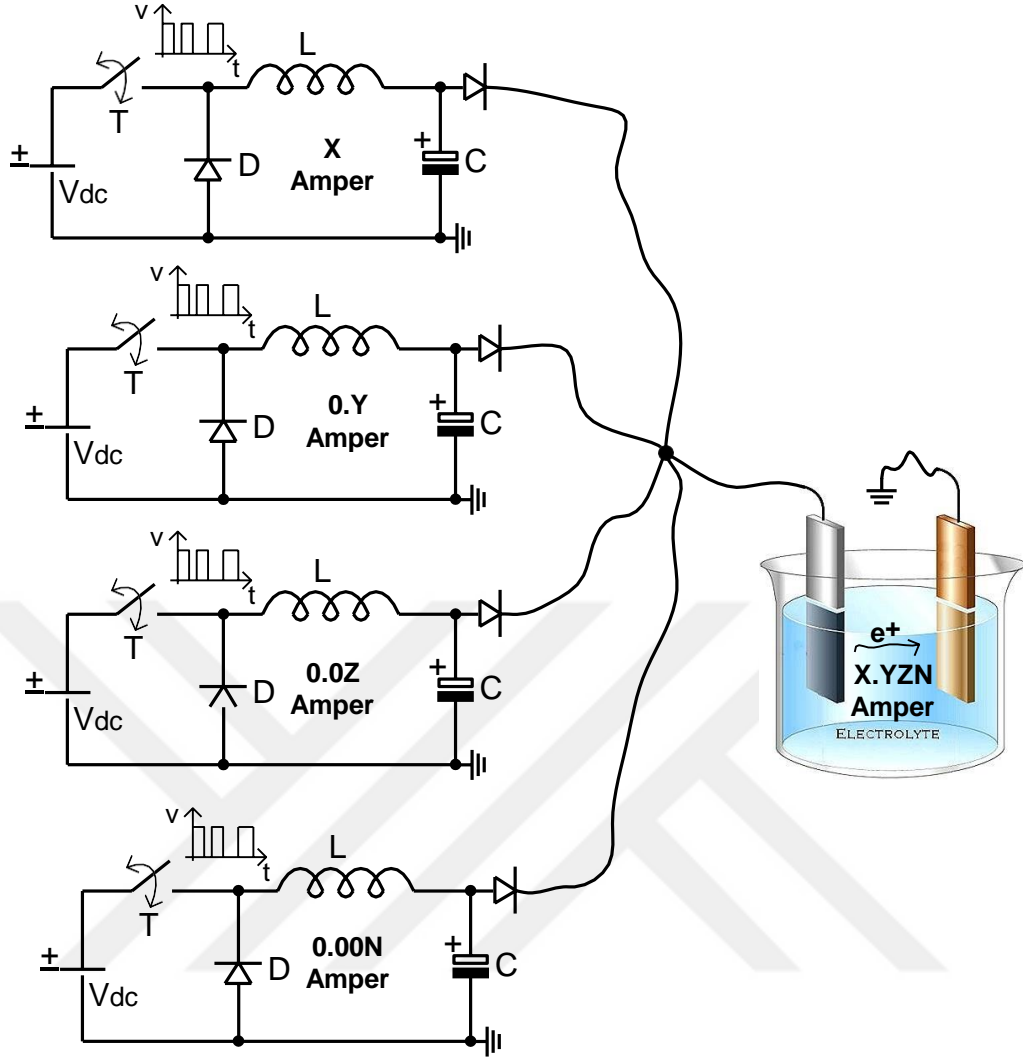
Şekil 3.35. Akım ayırma yazılımı.

Ayırılmış akımlar, 4 farklı Buck dönüştürücü kontrol algoritmasına referans akım olacak şekilde aktarılır. Referans akımlar çerçevesinde nihai elektrolit akımları

oluşturulur. 4 farklı referans akımı, 4 farklı elektrolit akımını oluşturur. Bu akımlar, elektrolit sıvısından geçişi esnasında tek bir akım gibi etki gösterir, Şekil 3.36.

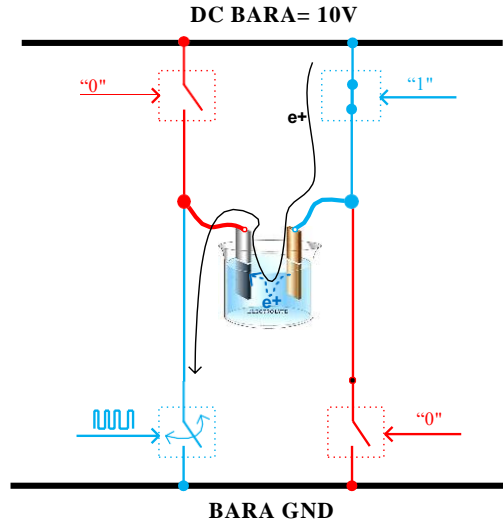


Şekil 3.36. Elektrolit sıvısının akım toplama etkisi.



Şekil 3.37. Farklı kaynaktan gelen 4 farklı akımın elektrolit sıvısından geçişi esnasındaki nihai değeri.

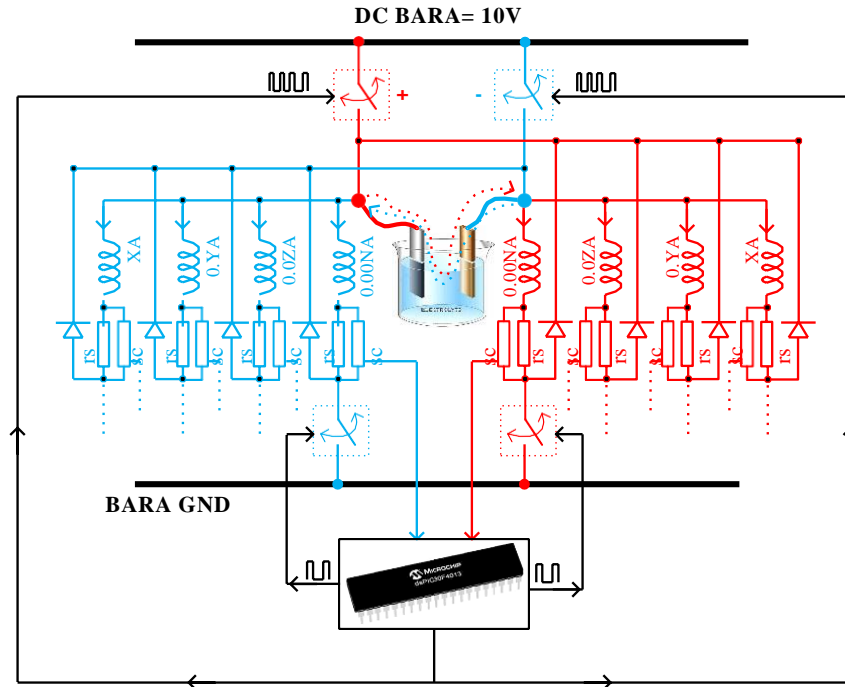
Tez çalışmasında tasarımı amaçlanan akım kaynağı, çıkış akım dalga formu pozitif ve negatif değer alabilecek şekilde amaçlandı. Bu sebeple Şekil 3.37'de verilmiş olan donanım alt yapısı yetersizdir. Aynı şekilde, akım kaynağı sürekli aynı yönde (+) akım akıtabilirken, ters yönde (-) akım akıtamıyor. Bu sebeple güç elektroniği devre topolojilerinden olan H-köprü devresi, Şekil 3.38, Buck dönüştürücü devrelerine entegre edildi.



Şekil 3.38. Çift yönlü akım eldesi için H-köprü devresi.

Akım kaynağı sisteminde 4'ü pozitif akım dalga şekli ve 4'ü negatif akım dalga şekli olmak üzere toplam 8 adet Buck dönüştürücü ve pozitif/negatif dönüşümü sağlayacak 1 adet H-köprü devresi tasarlandı, Şekil 3.39.

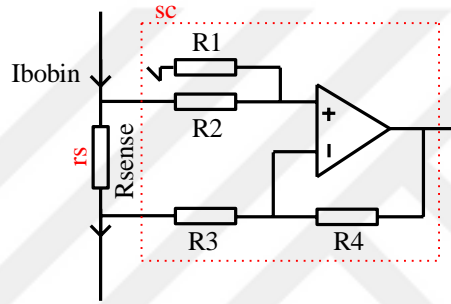
Buck dönüştürücü devrelerinden geçecek olan akımın ve H-köprüsünün tetikleme açılarının kontrol altına alınabilmesi amacıyla gömülü sistem olan ve Microchip firmasının bir ürünü mikrodenetleyici olarak seçilmiştir.



Şekil 3.39. Gömülü sistem, H-köprü devresi ve 8 adet Buck dönüştürücünün entegrasyonu.

Devreye dâhil olan anahtarlar BJT transistör, FET veya IGBT ailesinden seçilebilir. Düşük bara voltajı ve düşük elektrolit akımı nedeniyle, bu çalışmada FET ailesinden (IXTN79N20) uygun yarı iletkenler seçilmiştir.

Bobinlere seri bağlanan düşük değerli dirençler (r_s), hattan geçen bobin akımını gerilime çevirir, Denklem 1. $R_{sense} = 1-9A$ hattı için 0.1Ω , $0.1-0.9A$ hattı için 1Ω , $0.01-0.09A$ hattı için 10Ω ve $0.001-0.09A$ hattı için 100Ω değerinde seçilmiş akım algılama dirençleridir. Direnç üzerinde oluşan voltaj, operasyonel işlemci devresi (sc) tarafından işlendikten sonra gömülü sistem olan mikrodenetleyicinin adc pinine aktarılır, Şekil 3.40.

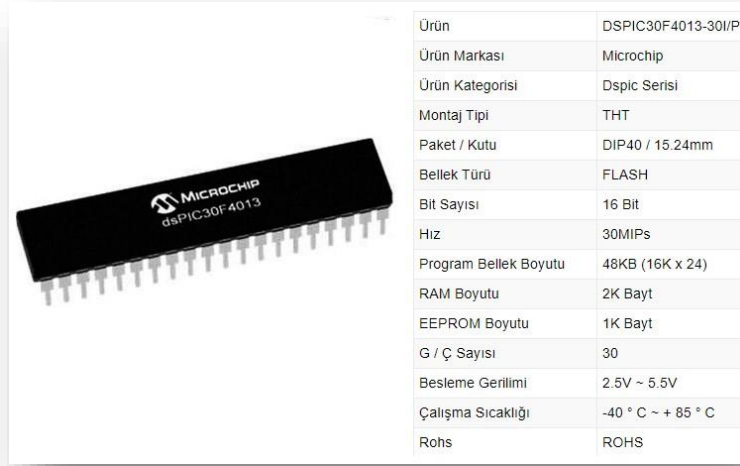


Şekil 3.40. Rsense akım algılama direnci ve opamp devresi.

Elektrik akımı kırmızı çizgiler üzerinden akıyorsa pozitif akım, mavi çizgiler üzerinden akarsa negatif akım olarak kabul edilmiştir, Şekil 3.39. Buck dönüştürücü kapasitörleri hibrit devreye eklenmemiştir. Bunun nedeni, devre akımının sürekli değişmesidir. Akım değişim hızı, geniş bir bant aralığında olduğundan eklenecek kondansatör bu bant aralığını daraltacaktır. Bu nedenle Buck Converter devresinin kondansatörü bu hibrit devre sistemine eklenmemiştir.

3.3.2. Gömülü sistem

Güç elektroniği devre topolojisinin öngörüldüğü gibi çalışabilmesi amacıyla gömülü bir sistem olan dsPIC30F4013 numaralı Microchip firmasına ait mikrodenetleyicinin kullanılması uygun görülmüştür, Şekil 3.41.



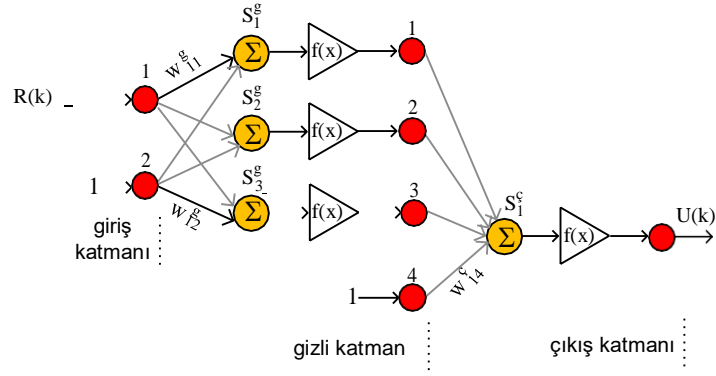
Şekil 3.41. Anahtarlama ve iletişim amaçlı kullanılan dsPIC30F4013 numaralı gömülü sistem.

Bu işlemcinin veri yolu 16 bit olup işlem kapasitesi 30MIPS'dir. Görev alan işlemcinin 2 görevi karşılaması gerekmektedir. Bunlardan bir tanesi, yazılım ortamında bulunan güç elektroniği devre topolojisine ait çalışma parametrelerinin WiFi iletişim protokolü üzerinden bilgisayardan çekilmesi. İkinci görevi ise gelen akım referans değerlerine dayalı Buck dönüştürücü kontrol algoritmalarının işletilmesidir.

3.4. Kontrol Algoritması

Buck dönüştürücülere uygulanacak olan PWM işaretinin duty değeri, hataya dayalı iş gören kontrol algoritması üzerinden hesaplanacaktır. Kontrol algoritmasının kullanım amacı, kullanıcı tarafından belirlenmiş olan referans akım değerini Buck dönüştürücü çıkış akımı olarak yansıtabilmektir. Çıkış akımının referansa dayalı hata yüzdesinin de minimum düzeyde kalması istenir. Kontrol algoritması olarak, YSA algoritması seçilmiştir. YSA algoritması ayrık zaman, hata minimizasyonuna dayalı, doğrusal olmayan adaptif bir matematiksel algoritmadır. Algoritmanın bu özelliklerinden dolayı da çeşitli mühendislik problemlerinin çözümünde etkin bir şekilde kullanılmıştır. YSA ile ilgili çalışmalar genel olarak sınıflandırma, modelleme, tahmin ve kontrol şeklinde ayrı kümelerde toplanabilir.

Kontrol algoritması olarak iş göreceği olan YSA, giriş, çıkış ve gizli katmanlarına ait yapı aşağıda yansıtılmıştır, Şekil 3.42.



Şekil 3.42. Kontrol algoritmasında kullanılacak olan YSA yapısı.

Şekil 3.42'deki değişkenler şu şekilde tanımlanmıştır,

S_1^g, \dots, S_3^g : Gizli katma (g) toplama merkezi.

$w_{11}^g, \dots, w_{31}^g$: Giriş katmanı ile gizli katman arasındaki ağırlıklar.

$w_{12}^g, \dots, w_{32}^g$: Bias ile gizli katman arasındaki ağırlıklar.

$R(k)$: YSA giriş verisi.

$U(k)$: YSA giriş verisi.

S_1^c : Çıkış katmanına ait toplama merkezi.

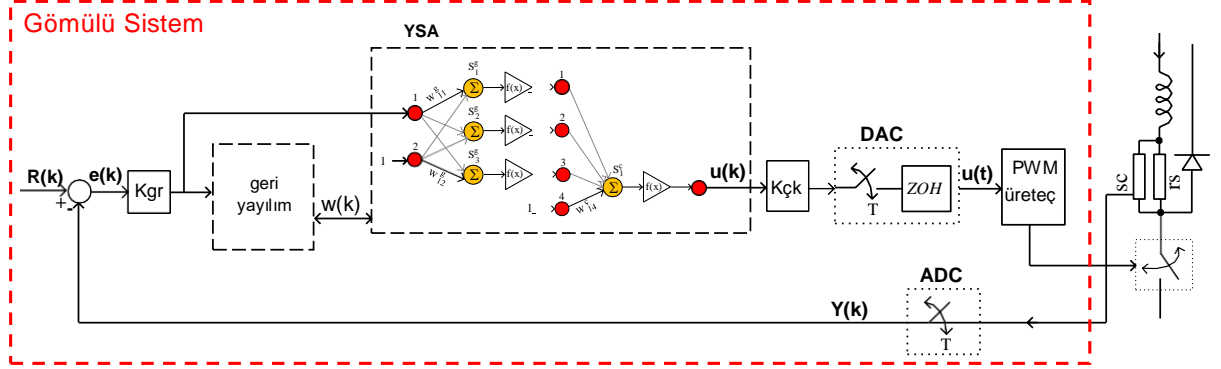
$w_{11}^c, \dots, w_{13}^c$: Çıkış katmanı ile giriş katmanı arasındaki ağırlıklar.

w_{14}^c : Gizli katman ile çıkış katmanı arasındaki ağırlık katsayısı.

$f(x)$: Aktivasyon fonksiyonu.

Bu tasarımda, aktivasyon fonksiyonu olarak "lojistik" kullanılmıştır. Bunun sebebi ise kontrol sürecinde uygulanacak olan PWM duty değerinin hiç bir zaman negatif değer olmamasıdır. "lojistik" fonksiyonu da (0,1) aralığında değer alabilmektedir.

YSA algoritmasının Buck dönüştürücü hibrit güç elektroniği devre topolojisine kontrol algoritması olarak adaptasyonu aşağıdaki şekilde gösterilmiştir, Şekil 3.43.



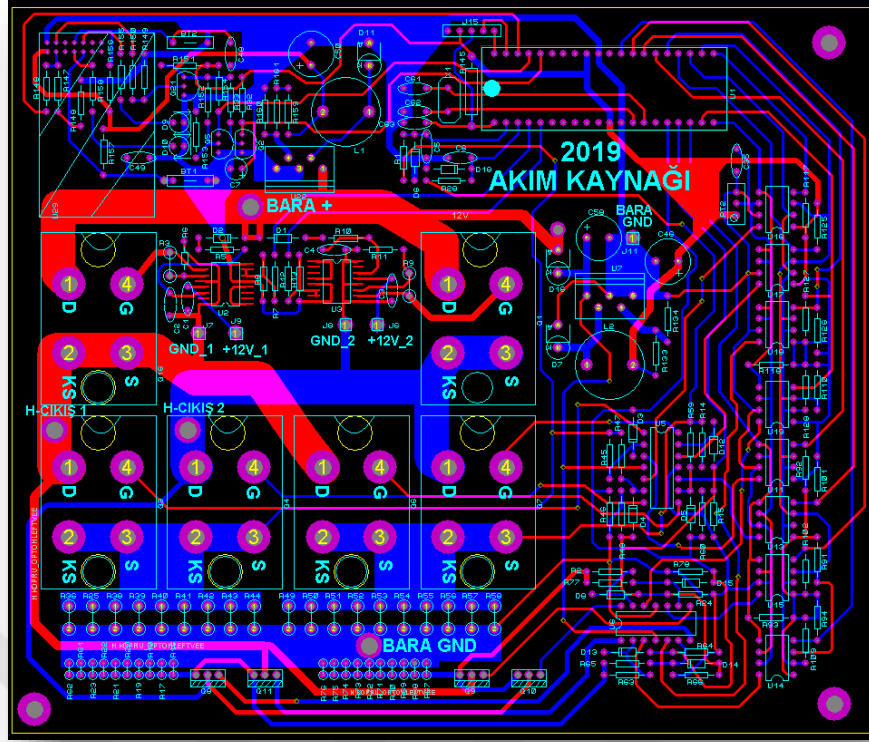
Şekil 3.43. YSA'nın kontrol algoritması olarak uyarlanması.

Yukarıda verilmiş olan kontrol mekanizmasından 4 adet tasarlanmıştır. Bunun sebebi ise gelen referans akımının 4'e ayrıştırılarak 4 farklı akım kolunun oluşturulmasıdır.

YSA algoritmasının işletilme zamanından kazanılabilmesi için aktivasyon fonksiyonuna ait giriş çıkış ilişkisi tablo yöntemine dayalı kurulmaktadır. Böylece matematiksel olarak yoğun işlemler bertaraf edilmiş olur.

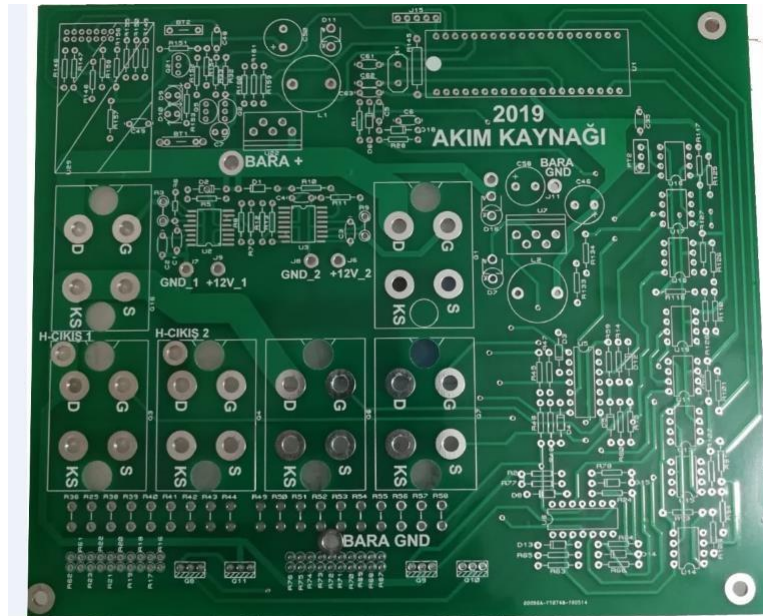
3.5. Devre Tasarımı

Akım kaynağına ait elektronik devre tasarımı Proteus ortamında şematik ve devre çizimleri geliştirildi, Şekil 3.44.



Şekil 3.44. Akım kaynağı Proteus ortamında geliştirilen şematiği.

Sanal ortamda devrenin fiziksel görünümü Şekil 3.45'de yansıtılmıştır.



Şekil 3.45. Akım kaynağı donanımının baskı devresi.

Baskı devre fiziksel olarak elde edildikten sonra devre elemanları üzerlerine monte edilerek lehimlendi. Kart üzerinde her bir ünite çalışma kontrolleri yapıldıktan sonra, nihai ürün olarak uygun bir kutu içerisine montajı gerçekleştirildi, Şekil 3.46 ve Şekil 3.47.



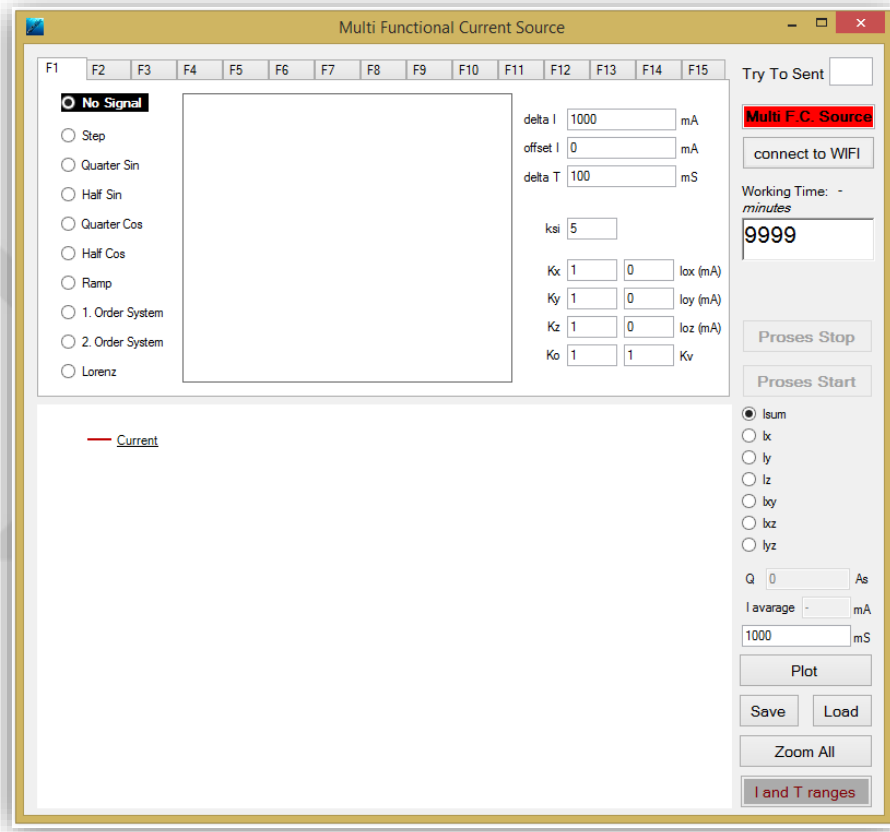
Şekil 3.46. Akım kaynağı devresinin nihai yerleşim planı.



Şekil 3.47. Akım kaynağı sisteminin nihai görüntüsü.

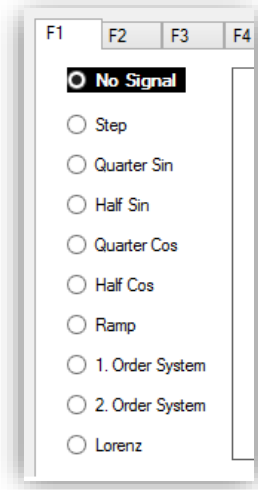
3.6. Bilgisayar Yazılımı

Bilgisayar yazılımı, Microsoft Visual C# ortamında geliştirildi. Geliştirilen yazılım sayesinde kullanıcı, akım kaynağı sistemine bilgisayar ortamından komutlar verebilir hale getirildi, Şekil 3.48. Böylece akım kaynağı sistemine çalışma karakteristiği değiştirilebilir, esnek bir yapı kazandırıldı. Komutlar, kablosuz WiFi iletişim yöntemi ile akım kaynağı sistemine iletilebilmektedir.



Şekil 3.48. Akım kaynağı sisteminin çalışma şekillerinin belirlendiği bilgisayar yazılım ortamı.

Yazılım ortamında akım kaynağı sisteminin akım verme biçimi DC ve değişken (8 farklı akım dalga formu) olmak üzere değiştirilebilmektedir, Şekil 3.49.



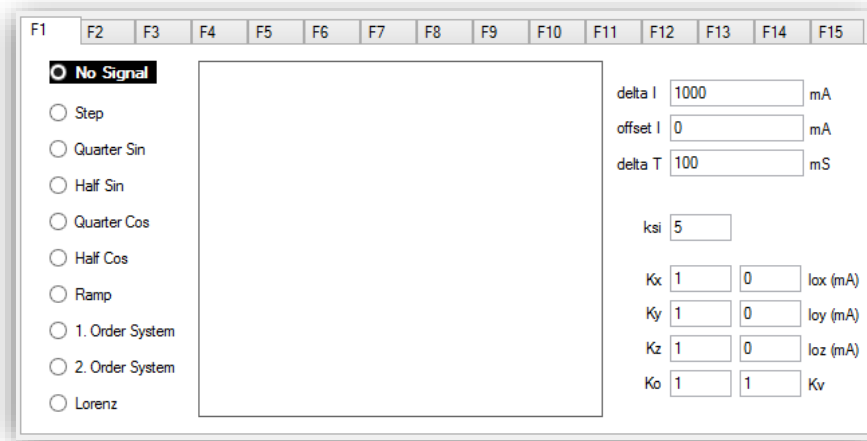
Şekil 3.49. Seçilebilen akım dalga türleri.

Değişken akım 1 adet dalga formundan oluşabileceği gibi; 15 farklı akım dalga şeklinden de oluşturulabilir, Şekil 3.50.



Şekil 3.50. 15 farklı değişken akım bileşeni için 15 farklı açılabilir pencere girişi.

Bundan dolayı her bir akım dalga fonksiyonu için bir pencere tasarlandı, Şekil 3.51.

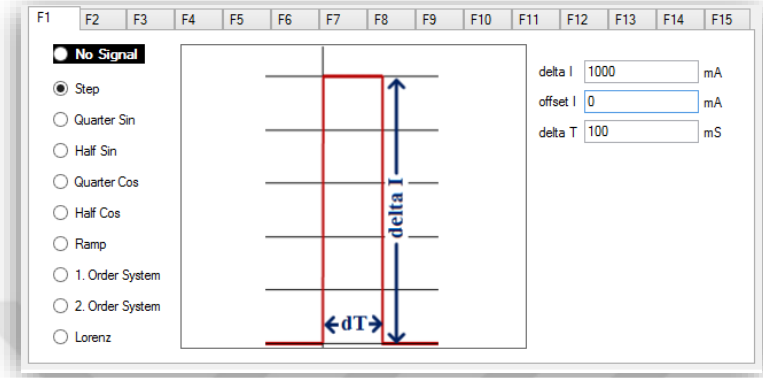


Şekil 3.51. 15 adet açılabilir pencereden 1 tanesi.

Her bir işaretin kendine has parametreleri mevcuttur. Aşağıda sırasıyla akım dalga formu ve parametre değişken ayarı sırasıyla verilmiştir.

3.6.1. Step

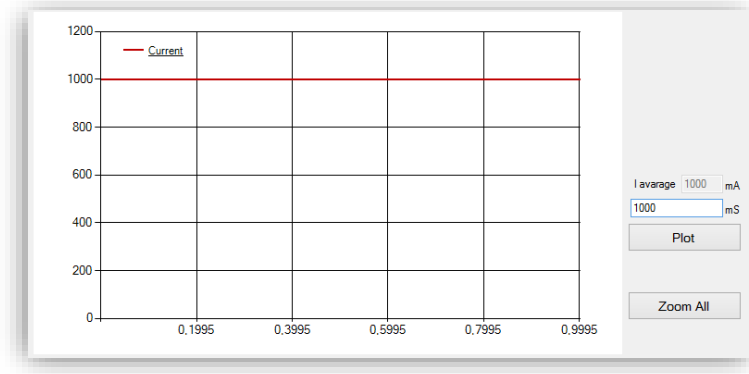
Tek başına kullanılması durumunda DC akım yani zamanla değişmeyen akım dalga formu elde etmek için kullanılır. Fakat değişken akım içerisinde gömülürse bu defa düz akım dalga formu elde etmek için kullanılır, Şekil 3.52.



Şekil 3.52. "Step" akım fonksiyonu penceresi.

Step fonksiyonu 3 adet parametreye sahiptir. "delta I", offset akımı ile arasındaki akım farkını belirlemek için kullanılır. Girilen değer mA türündendir. Maksimum 9999mA girilebilir. "offset I" parametresi akımın başlangıç değerini tayin eder. "offset I"=0mA ve "delta I"=1000mA seçim olursa, bunun anlamı akım kaynağı sistemi çalışmaya başlayınca, çıkışında 1000mA görülecek. "offset I"=500mA ve "delta I"=1000mA seçim olursa, bunun anlamı akım kaynağı sistemi çalışmaya başlayınca, çıkışında 1500mA görülecek. "delta T" parametresi, fonksiyonun değişken akım dalga formu içinde kullanılması durumunda Step akım dalga formu süresini belirler.

Bilgisayar yazılımı akım dalga formu osiloskop kullanılarak gerçek ortamda gözlemlenecek akım dalga formunu çizim ekranında yansıtarak kullanıcıya ön bilgilendirme yapabilmektedir, Şekil 3.53.

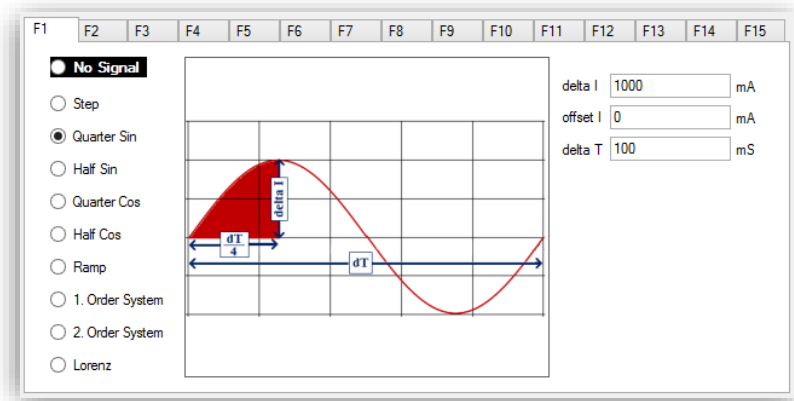


Şekil 3.53. Parametre değerleri Şekil 3.52 de verilmiş Step fonksiyonun 1000ms için dalga formu.

Ekrana dikkat edilecek olursa, "I avarage" bölümü mevcut. Kutucuk içerisindeki değer, yazılım tarafından "plot" emri verilince otomatik doldurulur. "I avarage", akım dalgasının ortalama değerini verir. Yani değişken akım dalgasının DC eşdeğer değerini vermiş oluyor. Bu değer daha sonra Q akım hesaplamalarında kullanılarak akım yoğunluğuna dayalı çeşitli malzeme analizlerinin yapılmasına olanak sağlayan bir parametredir.

3.6.2. Quarter Sin

"Quarter Sin" akım dalga fonksiyonu, çeyrek sinüs akım dalga formu çizdirebilmek amacıyla oluşturulmuştur. "delta I", "offset I" ve "delta T" olmak üzere 3 adet parametreye sahiptir, Şekil 3.54.



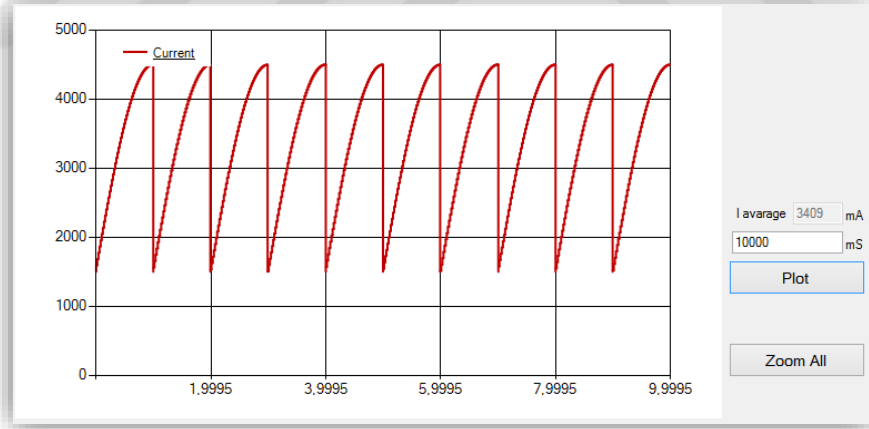
Şekil 3.54."Quarter Sin" akım fonksiyonu penceresi.

"Quarter Sin" minimum akım değerinden maksimum akım değerine çıkış süresi $T/4$ olduğu için "delta T" değeri girilirken, istenilen zaman değeri 4 ile çarpılmalıdır.

"delta I", offset akımı ile sinüs tepe akımı arasındaki akım farkını belirlemek için kullanılır. Girilen değer mA türündendir. Maksimum 9999mA girilebilir.

"offset I" parametresi akımın başlangıç değerini tayin eder. "offset I"=0mA ve "delta I"=1000mA girilmiş olursa bunun anlamı, akım kaynağı sistemi çalışmaya başlayınca, çıkışında 1000mA tepe akımına sahip bir çeyrek sinüs görülecek. "offset I"=500mA ve "delta I"=1000mA seçilirse bu defa akım kaynağı sistemi çalışmaya başlayınca, çıkışında 500mA'den 1500mA'ye doğru yükselen çeyrek akım dalga formu görülecektir.

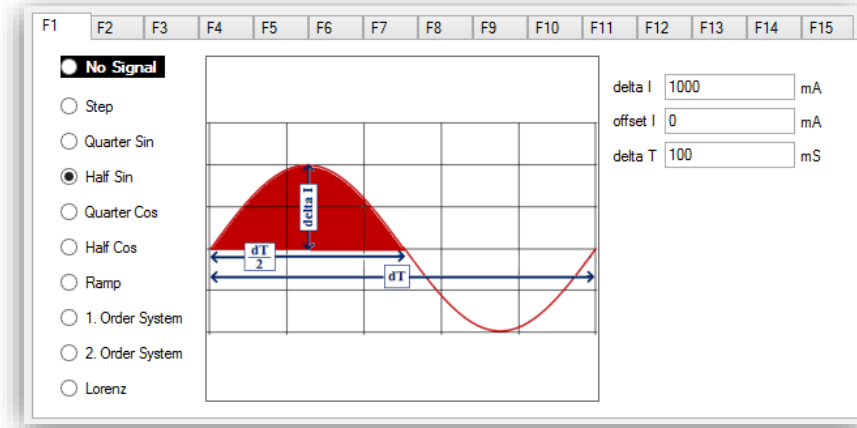
Aşağıda, periyodik olarak çizdirilmiş olan çeyrek sinüs dalgasının parametre değerleri sırasıyla "delta I"=3000mA, "offset I"=1500mA ve "delta T"=4s yani çeyrek sinüsün min değerinden maksimum değerine 1s'de gidecektir, Şekil 3.55. Dikkat edilecek olursa değişken işaretin DC eşdeğeri 3409mA'dir.



Şekil 3.55. Parametre değerleri, "delta I"=3000mA, "offset I"=1500mA ve "delta T"=4s için "Quarter Sin" 10s için dalga formu.

3.6.3. Half Sin

"Half Sin" akım dalga fonksiyonu, yarım sinüs akım dalga formu çizdirebilmek amacıyla oluşturulmuştur. "delta I", "offset I" ve "delta T" olmak üzere 3 adet parametreye sahiptir, Şekil 3.56.

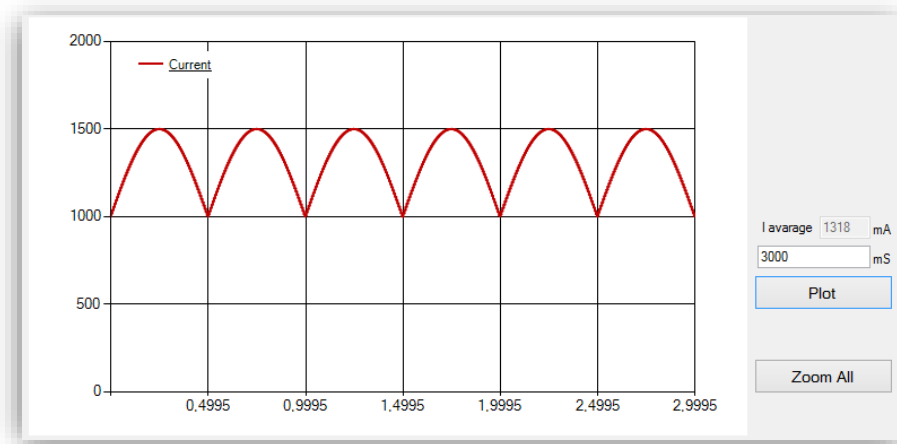


Şekil 3.56. "Half Sin" akım fonksiyonu penceresi.

"Half Sin" Imin, I_{max} ve tekrar Imin toplam düşme süresi yani yarım periyot süresi girilirken değeri 2 ile çarpılmalıdır. Çünkü "delta T" süresi 1T süresidir.

"delta I", offset akımı ile sinüs tepe akımı arasındaki akım farkını belirlemek için kullanılır. Girilen değer mA türündendir. Maksimum 9999mA girilebilir.

Aşağıda, periyodik olarak çizdirilmiş olan yarım sinüs dalgasının parametre değerleri sırasıyla "delta I"=500mA, "offset I"=1000mA ve "delta T"=1s yani yarım sinüsün oluşma süresi 500ms olacaktır, Şekil 3.57.

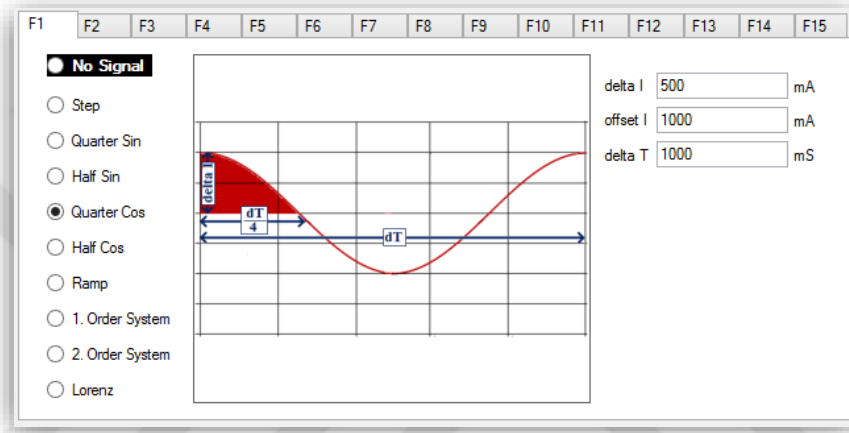


Şekil 3.57. Parametre değerleri, "delta I"=500mA, "offset I"=1000mA ve "delta T"=1s için "Half Sin" 10s için dalga formu.

Dikkat edilecek olursa deęişken iřaretin DC eřdeęeri 3000mA'dir.

3.6.4. Quarter Cos

"Quarter Cos" akım dalga fonksiyonu, eyrek cosinüs akım dalga formu izdirebilmek amacıyla oluřturulmuřtur. "delta I", "offset I" ve "delta T" olmak üzere 3 adet parametreye sahiptir, řekil 3.58.

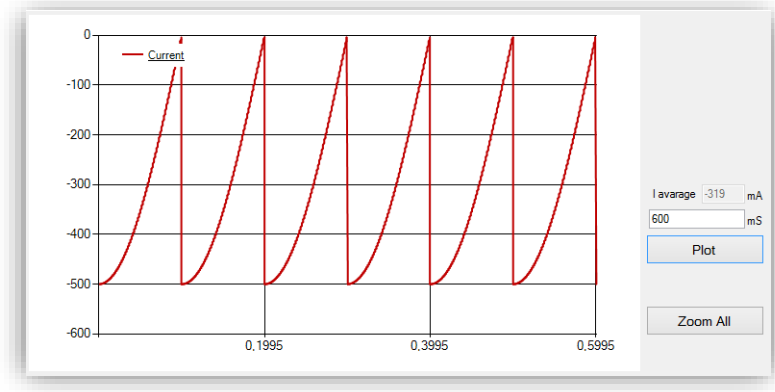


řekil 3.58. "Quarter Cos" akım fonksiyonu penceresi.

"Quarter Cos" maksimum akım deęerden minimum akım deęerine iniř sũresi $T/4$ olduęu iin "delta T" deęeri girilirken, istenilen zaman deęeri 4 ile arpılmalıdır.

"delta I", offset akımı ile sinüs tepe akımı arasındaki akım farkını belirlemek iin kullanılır. Girilen deęer mA tũrũndendir. Maksimum 9999mA girilebilir.

Ařaęıda, periyodik olarak izdirilmiř olan yarım sinüs dalgasının parametre deęerleri sırasıyla "delta I"=-500mA, "offset I"=0mA ve "delta T"=400ms yani eyrek kosinüs oluřma sũresi 100ms olacaktır, řekil 3.59.

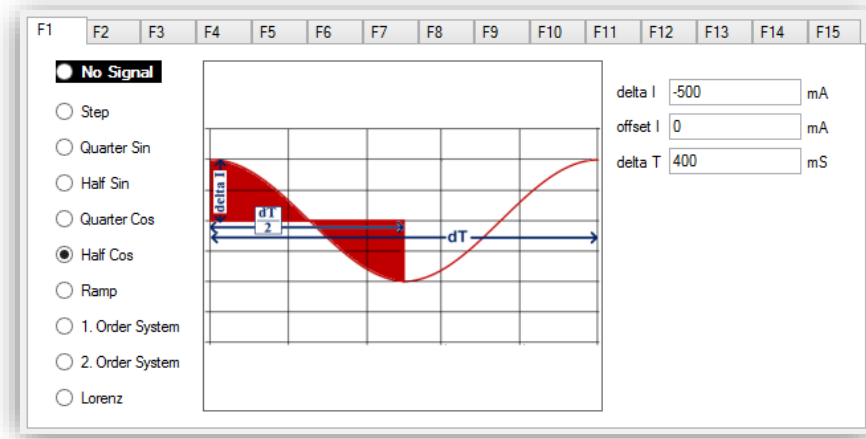


Şekil 3.59. Parametre değerleri, " $\Delta I = -500\text{mA}$ ", " $\text{offset } I = 0\text{mA}$ " ve " $\Delta T = 400\text{ms}$ " için "Quarter Sin" 6000ms için dalga formu.

Dikkat edilecek olursa değişken işaretin DC eşdeğeri -319mA 'dir.

3.6.5. Half Cos

"Half Cos" akım dalga fonksiyonu, yarım periyot kosinüs akım dalga formu çizdirebilmek amacıyla oluşturulmuştur. " ΔI ", " $\text{offset } I$ " ve " ΔT " olmak üzere 3 adet parametreye sahiptir, Şekil 3.60.

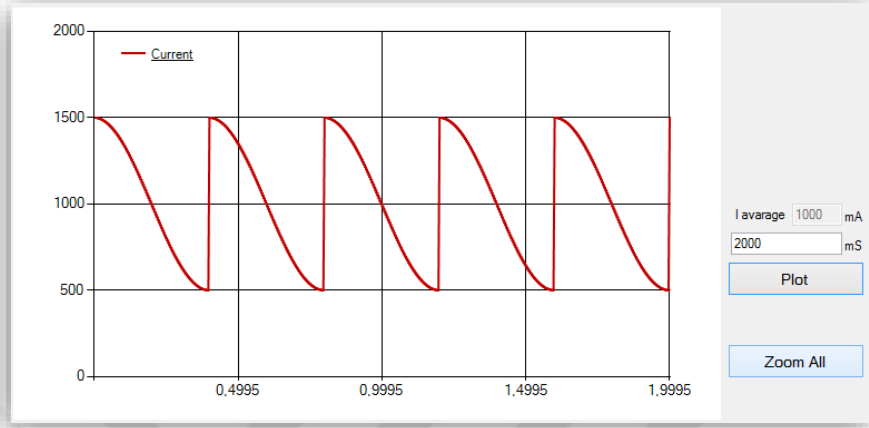


Şekil 3.60. "Half Cos" akım fonksiyonu penceresi.

"Half Cos" maksimum akım değerinden minimum akım değerine iniş süresi $T/2$ olduğu için " ΔT " değeri girilirken, istenilen zaman değeri 2 ile çarpılmalıdır.

" ΔI ", offset akımı ile kosinüs tepe akımı arasındaki akım farkını belirlemek için kullanılır. Girilen değer mA türündendir. Maksimum 9999mA girilebilir.

Aşağıda, periyodik olarak çizdirilmiş olan yarım kosinüs akım dalgasının parametre değerleri sırasıyla " ΔI "=500mA, "offset I"=1000mA ve " ΔT "=800ms yani yarım kosinüs oluşma süresi 400ms olacaktır, Şekil 3.61.

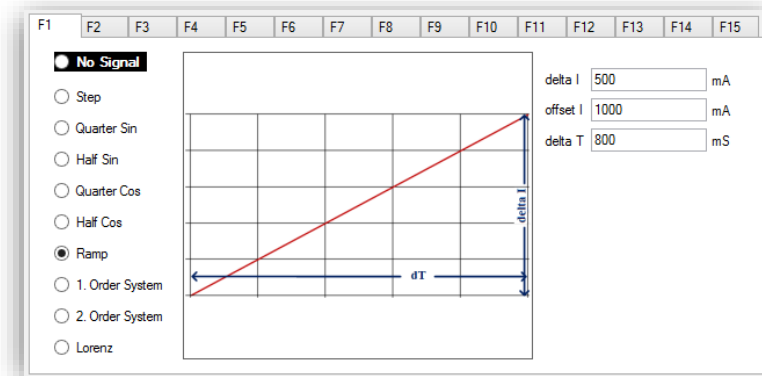


Şekil 3.61. Parametre değerleri, " ΔI "=500mA, "offset I"=1000mA ve " ΔT "=800ms için "Half Cos" 2000ms süresince oluşan dalga formu.

Dikkat edilecek olursa değişken işaretin DC eşdeğeri 1000mA'dır.

3.6.6. Ramp

"Ramp" akım dalga fonksiyonu, rampa biçiminde yükselen bir akım formu elde etmek amacıyla geliştirilmiştir. " ΔI ", "offset I" ve " ΔT " olmak üzere 3 adet parametreye sahiptir, Şekil 3.62.

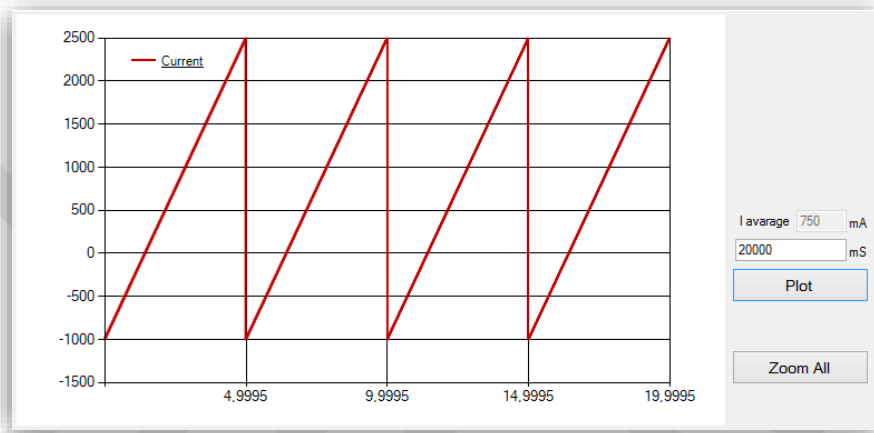


Şekil 3.62. "Ramp" akım fonksiyonu penceresi.

"delta T" parametresi, rampa akımının min akım değerinden maksimum akım değerine ulaşma süresini verir.

"delta I", offset akımı ile rampa akımına ait tepe akımı arasındaki akım farkını belirlemek için kullanılır. Girilen değer mA türündendir. Maksimum 9999mA girilebilir.

Aşağıda, periyodik olarak çizdirilmiş olan rampa akım dalgasının parametre değerleri sırasıyla "delta I"=3500mA, "offset I"=-1000mA ve "delta T"=5000ms, Şekil 3.63.

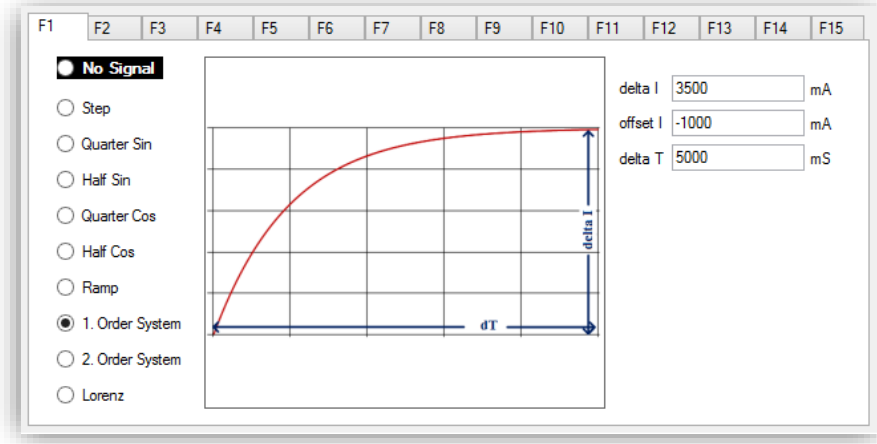


Şekil 3.63. Parametre değerleri, "delta I"=3500mA, "offset I"=-1000mA ve "delta T"=5000ms için "Ramp" akım dalga fonksiyonu 20s işletilmesi sonrası oluşan akım dalga formu.

Dikkat edilecek olursa değişken işaretin DC eşdeğeri 750mA'dir.

3.6.7. 1. Derece sistem cevabı

"1. Order System" fonksiyon cevabı, kontrol mühendisliğinde kullanılan basamak girişi için 1. derece sistem cevabı şeklinde akım dalga formudur. "delta I", "offset I" ve "delta T" olmak üzere 3 adet parametreye sahiptir, Şekil 3.64.

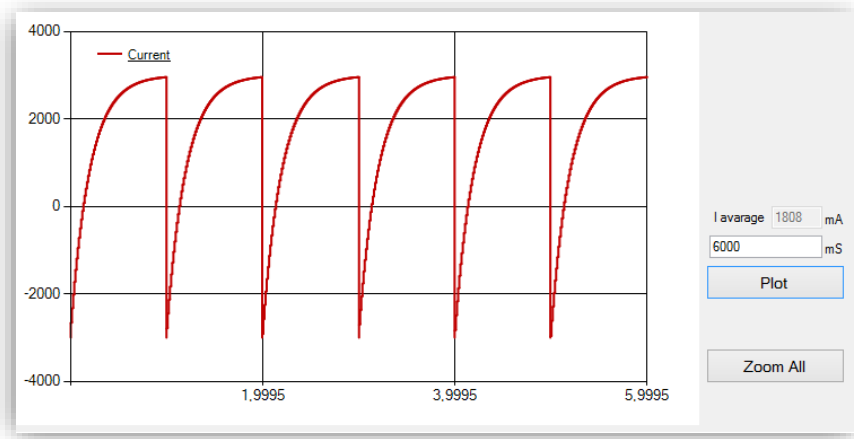


Şekil 3.64. "1. Order System" akım fonksiyonu penceresi.

"delta T" parametresi, akımının minimum akım değerinden maksimum akım değerine ulaşma süresini temsil eder.

"delta I", offset akımı ile rampa akımına ait maksimum akımı arasındaki akım farkını belirlemek için kullanılır. Girilen değer mA türündendir. Maksimum 9999mA girilebilir.

Aşağıda, basamak girişi için 1. derece sistem çıkışı akım dalgasının parametre değerleri sırasıyla "delta I"=6000mA, "offset I"=-3000mA ve "delta T"=1000ms, Şekil 3.65.

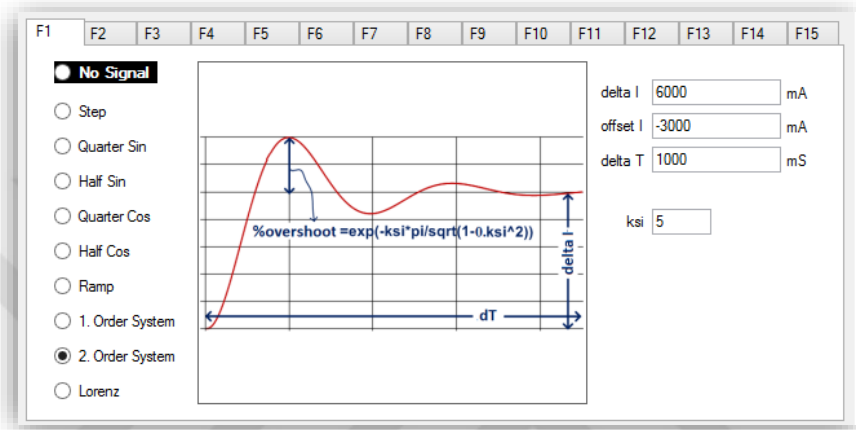


Şekil 3.65. Parametre değerleri, "delta I"=6000mA, "offset I"=-3000mA ve "delta T"=1000ms için "1. Order System" akım dalga fonksiyonunun 6s işletilmesi sonrası oluşan akım dalga formu.

Dikkat edilecek olursa değişken işaretin DC eşdeğeri 1808mA'dir.

3.6.8. 2. Derece sistem cevabı

"2. Order System" fonksiyon cevabı, kontrol mühendisliğinde kullanılan basamak girişi için 2. derece sistem cevabı şeklinde akım dalga formudur. "delta I", "offset I", "delta T" ve "ksi" parametresi olmak üzere 4 adet parametreye sahiptir, Şekil 3.66.

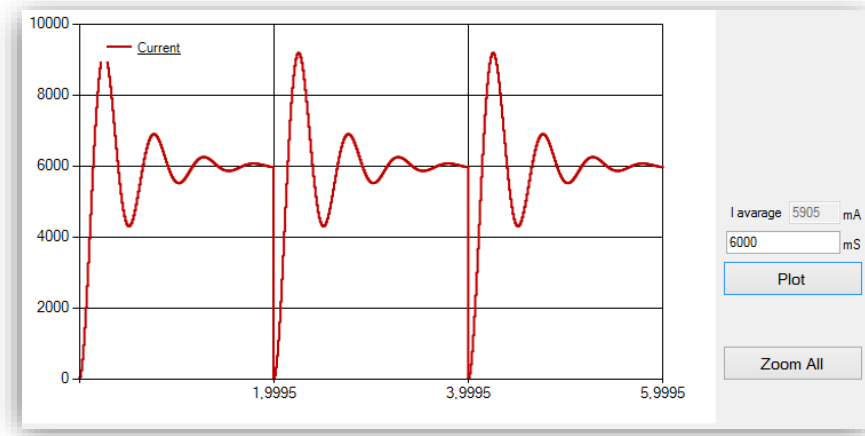


Şekil 3.66. "2. Order System" akım fonksiyonu penceresi.

"delta T" parametresi, akım dalgasına ait değişimin sürekli zaman aralığına geçiş süresidir.

"delta I", sürekli zaman aralığındaki akım dalgasına ait değer. Girilen değer mA türündendir. Maksimum 9999mA girilebilir. "ksi", akımın geçici zaman aralığındaki dinamiğini belirleyen parametredir.

Aşağıda, basamak girişi için 2. derece sistem çıkışı akım dalgasının parametre değerleri sırasıyla "delta I"=6000mA, "offset I"=0mA, "delta T"=1000ms ve ksi=3 (yani 0.3), Şekil 3.67.

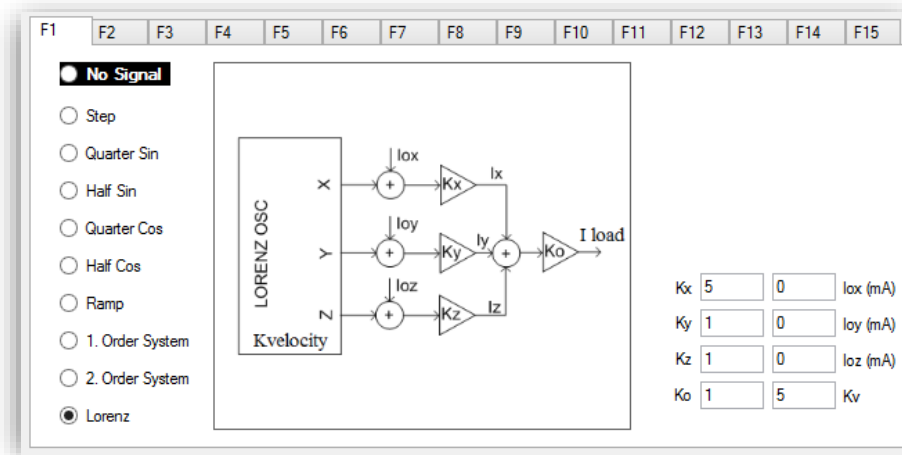


Şekil 3.67. Parametre değerleri, " ΔI "=6000mA, "offset I"=0mA, " ΔT "=1000ms ve " ξ "=3 (0.3) için "2. Order System" akım dalga fonksiyonunun 6s işletilmesi sonrası oluşan akım dalga formu.

Dikkat edilecek olursa değişken işaretin DC eşdeğeri 5905mA'dir.

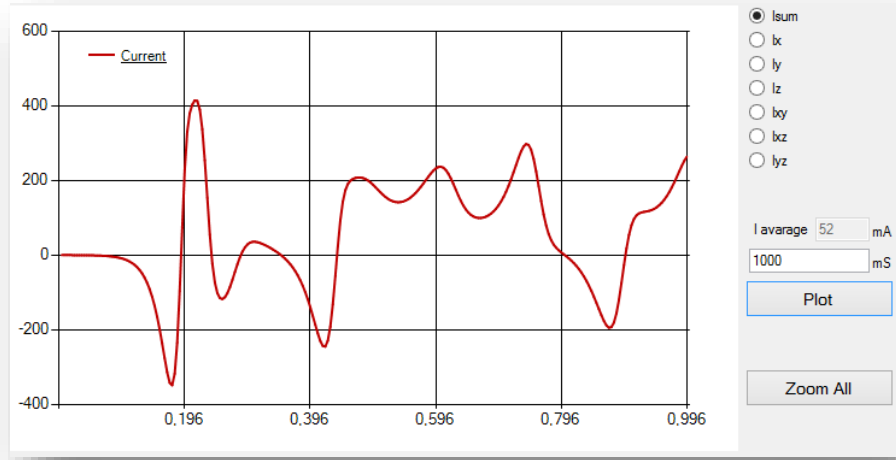
3.6.9. Lorenz

"Lorenz" fonksiyon cevabı, Kaos bilimine ait bir fonksiyon olan Lorenz fonksiyonu matematiksel modeli koşturularak elde edilen akım dalga formudur. "Lorenz" modelinde 0 adet giriş işareti ve 3 adet çıkış işareti vardır. İşaret üretimi, 8 adet parametre üzerinden hesaplanarak çalışma dinamiği değiştirilmektedir, Şekil 3.68.

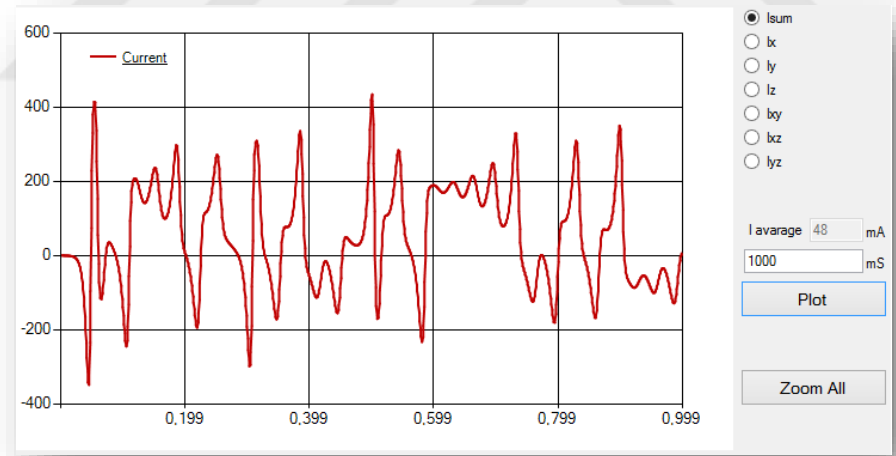


Şekil 3.68. "Lorenz " akım fonksiyonu penceresi.

Bu parametrelerden K_v , akım işaretinin hızını kontrol etmektedir, Şekil 3.69 ve 3.70.

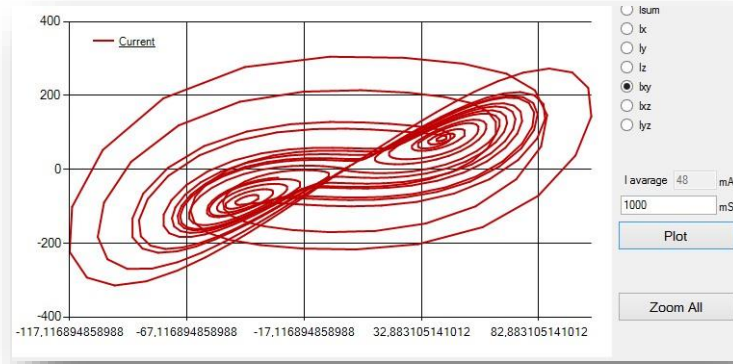


Şekil 3.69. Parametre değerleri, " k_x "=5, " k_y "=10, " k_z "=2, " k_v "=5, " k_o "=1 ve diğer parametre değerleri sıfır için "Lorenz" akım (mA) dalga fonksiyonunun 1s işletilmesi sonrası oluşan akım dalga formu.

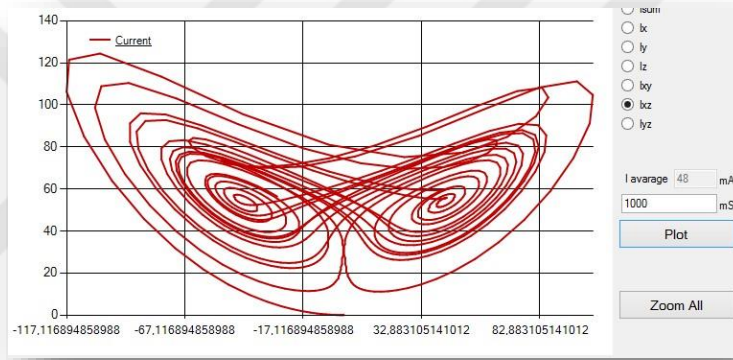


Şekil 3.70. Parametre değerleri, " k_x "=5, " k_y "=10, " k_z "=2, " k_v "=20, " k_o "=1 ve diğer parametre değerleri sıfır için "Lorenz" akım (mA) dalga fonksiyonunun 1s işletilmesi sonrası oluşan akım dalga formu.

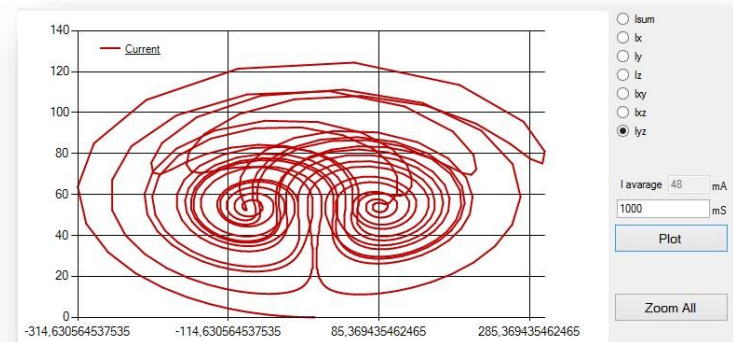
Literatürde, bu dalga formları tek tek veya toplam şeklinde çizdirilmektense x-y ekseninde iki çıkış işareti göz önünde bulundurularak çizdirilir. Literatürdeki çizim yöntemi göz önünde bulundurularak çizdirilirse "Lorenz" fonksiyon çıkışları aşağıda verilmiş olan dalga biçimleri elde edilecektir, Şekil 3.71, 3.72, 3.73.



Şekil 3.71. "Lorenz" XY çıkış değişkenlerinin beraber çizdirilmesi.



Şekil 3.72. "Lorenz" XZ çıkış değişkenlerinin beraber çizdirilmesi.

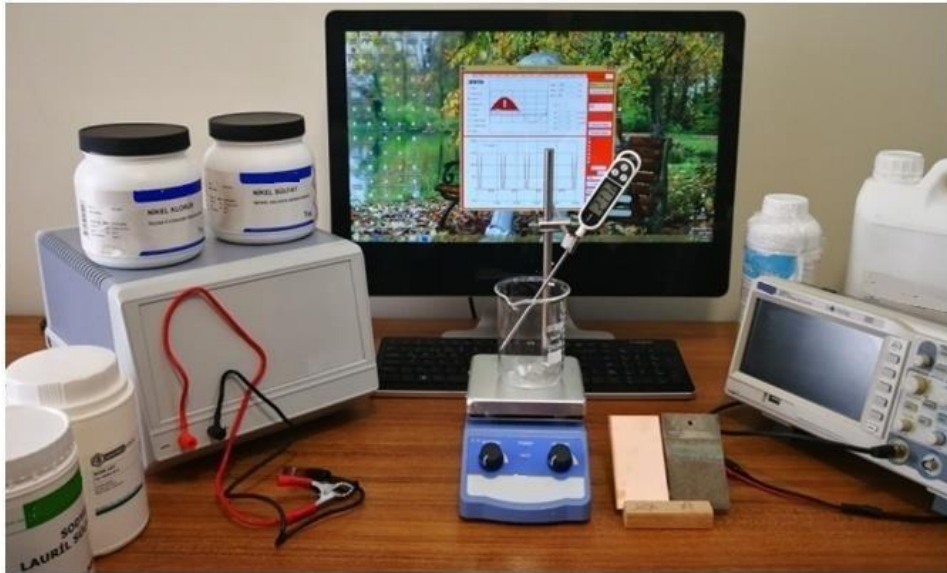


Şekil 3.73. "Lorenz" YZ çıkış değişkenlerinin beraber çizdirilmesi.

3.7. Kaplama Deney Düzenegi

Çalışma kapsamında üretilmiş olan, çok fonksiyonlu akım kaynağı kullanılarak kaplama deneyleri gerçekleştirilmiştir. Bu deneyleri yaparken DC akım, PC akım, PRC akım türlerine ek olarak testere, üçgen ve ters sinüs akım dalga türleri de kullanılarak kaplamalar gerçekleştirilmiştir.

Kaplama deneyleri için öncelikle anot ile katot malzemelerin seçimi ve uygun fiziksel boyutlara getirilme işlemi yapılmıştır. Daha önceki bölümde de bahsedildiği gibi kaplama deneylerinde katot malzemesi olarak bakır ve anot malzemesi olarak nikel kullanılmıştır . Kaplama deney ortamı Şekil 3.74'de görüldüğü gibi hazırlanmıştır.



Şekil 3.74. Kaplama Deney Düzenegi.

Kaplama deneylerinde kullanmış olduğumuz bakır, elektriksel izolasyonu cam elyaf ve epoksi reçine karışımı olan 1.6mm kalınlığındaki izolasyon malzemesi üzerine konumlandırılmıştır. Bakırın malzeme üzerindeki kalınlığı 35µm'dir. Bu bakır kaplama sistemi, FR4 olarak adlandırılır ve elektronik PCB kart tasarımlarında kullanılır, Şekil 3.75.



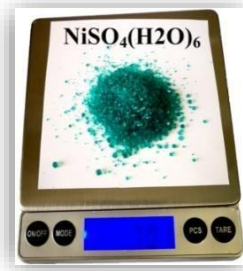
Şekil 3.75. Katot malzemesi.

Anot ve katot üzerinde çeşitli sebeplerden dolayı, elektriksel izolasyon oluşabilir. Bu izolasyonu yok etmek için yün polisaj süngeri, polisaj makinesi ve parlaticı cila kullanılarak bu problem giderilmiştir, Şekil 3.76.



Şekil 76. Polisaj işlemi.

Saf su, nikel sülfat $\text{NiSO}_4(\text{H}_2\text{O})_6$, nikel klorür ($\text{NiCl}_2 \cdot 6\text{H}_2\text{O}$), borik asit (H_3BO_3), sodyum lauril sülfat ($\text{NaC}_{12}\text{H}_{25}\text{SO}_4$) kimyasalları ile kimyasalların ağırlıklarını ölçebilecek ve yüksek hassasiyetli (0.01mg) dijital tartı kullanılarak elektrolit sıvısı hazırlanmıştır, Şekil 3.77.



Şekil 77. Deney malzemesi ölçümünde kullanılan dijital tartı.

3.7.1. Kaplama Banyosunun Özellikleri

Nikelin elektrodepozisyonu için yaygın olarak kullanılan iki tip elektrokaplama banyosu; Watt tipi banyo ve sülfamat banyosudur. Bu çalışma süresince elde edilen kaplamaların üretiminde Watt tipi kaplama banyosu kullanılmıştır. Standart Watt solüsyonu $\text{NiSO}_4 \cdot 6\text{H}_2\text{O}$ (Nikel sülfat heksahidrat), $\text{NiCl}_2 \cdot 6\text{H}_2\text{O}$ (Nikel klorür heksahidrat) ve H_3BO_3 (Borik asit) içerir. Tablo 3.1, 1litre banyo hazırlamak için kullanılan bu kimyasalların içeriğini göstermektedir.

Birçok nikel kaplama banyosuna borik asit eklenir. Borik asit, katot çözelti ara yüzünde pH değerini kontrol etmek için tampon görevi görmektedir. Ayrıca elektrodepozisyon deneylerinin öncesinde ve deney sırasında elektrolit sıvısı manyetik karıştırıcı ile karıştırılmıştır.

Tablo 3.3. Elektrolitik nikel kaplamalara ait banyo kompozisyonu

Banyo Bileşenleri	
$\text{NiSO}_4 \cdot 6\text{H}_2\text{O}$ (nikel sülfat)	300 g/L
$\text{NiCl}_2 \cdot 6\text{H}_2\text{O}$ (nikel klorür)	50 g/L
H_3BO_3 (borik asit)	40 g/L
SDS (sodyum dodesil sülfat)	0.2 g/L
Sıcaklık ($^{\circ}\text{C}$)	45 ± 5
Anot	Nikel
Katot	Bakır
pH	5
Manyetik Karıştırma	30 dk.

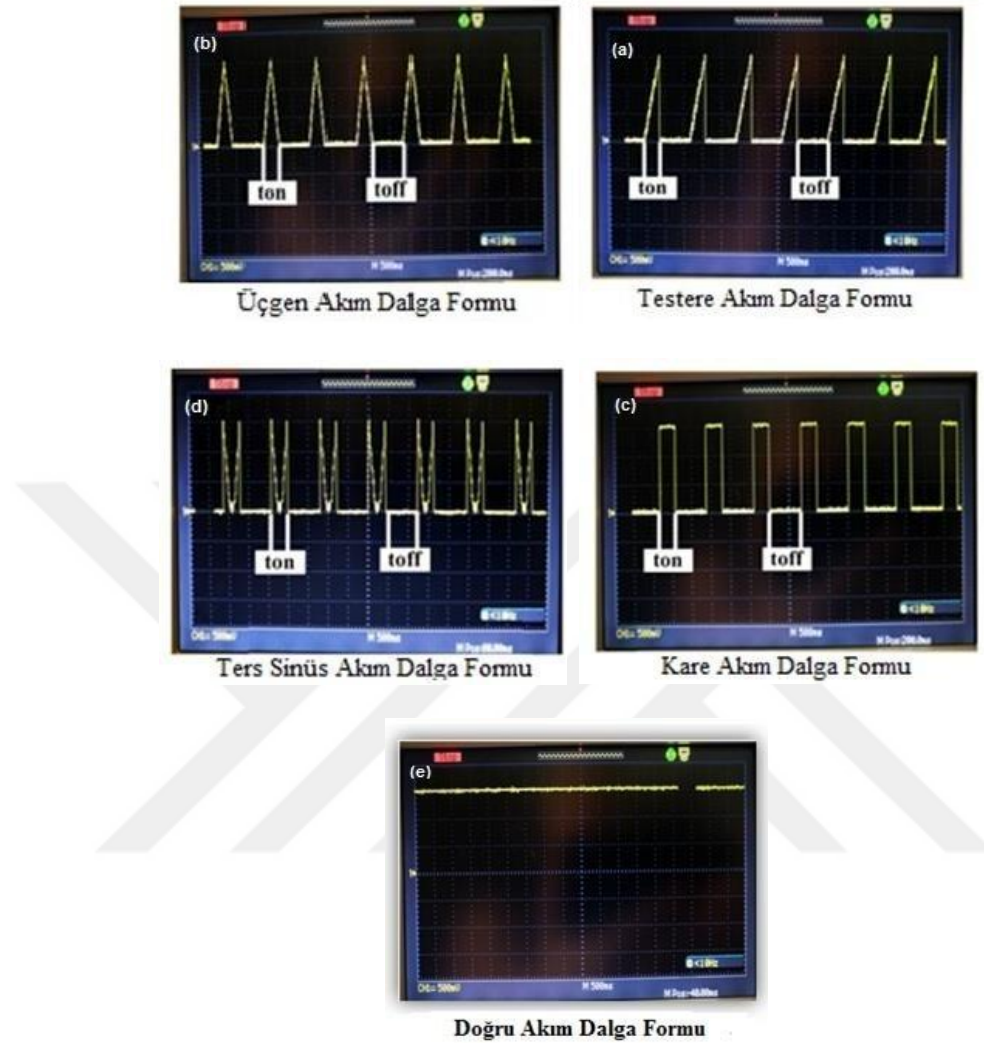
3.8. Yöntem

Bakır üzerindeki nikel kaplama deneylerini temelde iki farklı şekilde gerçekleştirdik. İlk deney sürecinde, elektrolitten geçen elektrik akımı negatif değerler almazken, ikinci deney sürecinde elektrik akımı negatif değerler içermektedir.

3.8.1. Deney 1 için uygulanan akım türleri

Bu çalışmada, doğru akım dalga formunun yanı sıra testere, üçgen, kare, yarım sinüs akım dalga formlarında akımlı beş farklı akım dalgası kullanılarak kaplama deneyi

gerçekleştirilmiştir, Şekil 3.78. Bu akım dalgalarının kaplama başarıları, geleneksel yöntem DC akım ile kaplamaya yöntemiyle karşılaştırılmıştır.

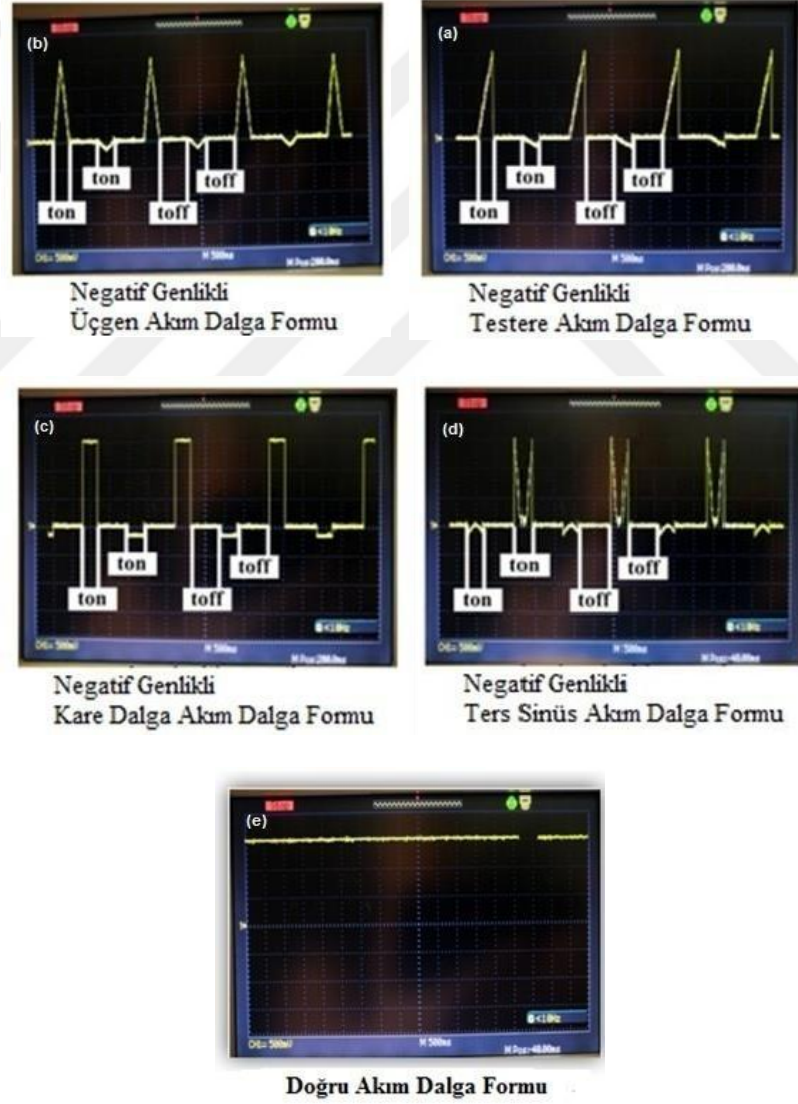


Şekil 78. Deneysel olarak üretilen a) Testere, b) Üçgen c) Kare d) Ters Sinüs e) Doğru akım dalga formları.

Kaplama işleminde, tüm akım sinyallerinin tepe değerleri 1500mA'de sabitlenmiştir. DC hariç, her bir akım dalgası ton = 40ms ve toff = 80ms, ton = 400ms ve toff = 800ms ton = 4000ms ve toff = 8000ms ile 3 farklı zaman değişiminde üretilmiştir, Şekil 15. Kaplama işleminden sonra, yüzey ve köşe kaplama başarıları incelenmiştir. Kaplama işleminde her kaplama deneyi 30 dakika ile sınırlandırılmıştır.

3.8.2. Deneysel olarak üretilen akım türleri

Bu uygulamada testere, üçgen, pals, ters yarım sinüs ve doğru akım dalga formları kullanılarak beş farklı kaplama deneyi gerçekleştirilmiştir, Şekil 3.79. Daha önceki deneyden farklı olarak, kaplama akımları negatif akım değerlerine sahiptir. Akım sinyallerinin negatif genlik değerleri, pozitif genlik değerinin % 10'u olan 150mA olarak seçilmiştir. Bu deneyin amacı, kaplama işleminde elde edilen bazı çözünmelerin nihai kaplama kalitesindeki olası değişiklikleri ortaya koymaktır. Akım dalgalarının kaplama başarıları, geleneksel DC akım yöntemi kullanılarak elde edilen kaplama sonuçları ile karşılaştırılarak analiz edilmiştir. Bu deney için de kaplama süresi tekrar 30 dakika ile sınırlandırılmıştır.



Şekil 3.79. Deney 2 için uygulanan a) Negatif genlikli testere, b) Negatif genlikli üçgen c) Negatif genlikli kare d) Negatif genlikli ters sinüs ve e) Doğru akım dalga formları.

4. BULGULAR

4.1. Kaplama Deney Sonuçları

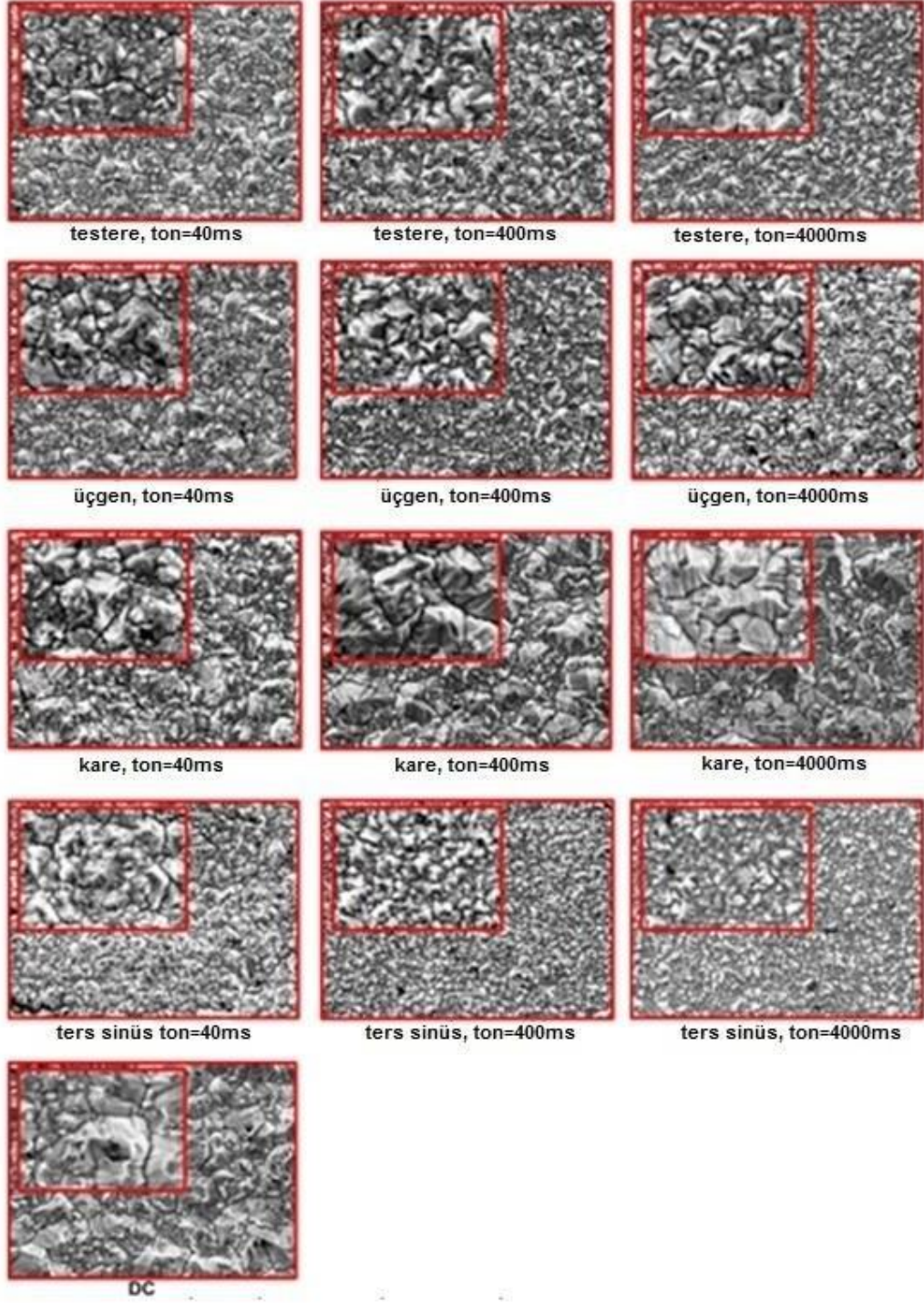
Tasarımını ve imalatını gerçekleştirmiş olduğumuz yeni nesil akım üretici ile elde ettiğimiz kaplama deney sonuçları, Jeol marka 6060 LV model taramalı elektron mikroskopunda incelenmiş olup, elde edilen sonuçların, geleneksel yönteme göre başarıları karşılaştırılmıştır.

4.1.1. Deney 1 taramalı elektron mikroskobu (SEM) analizi

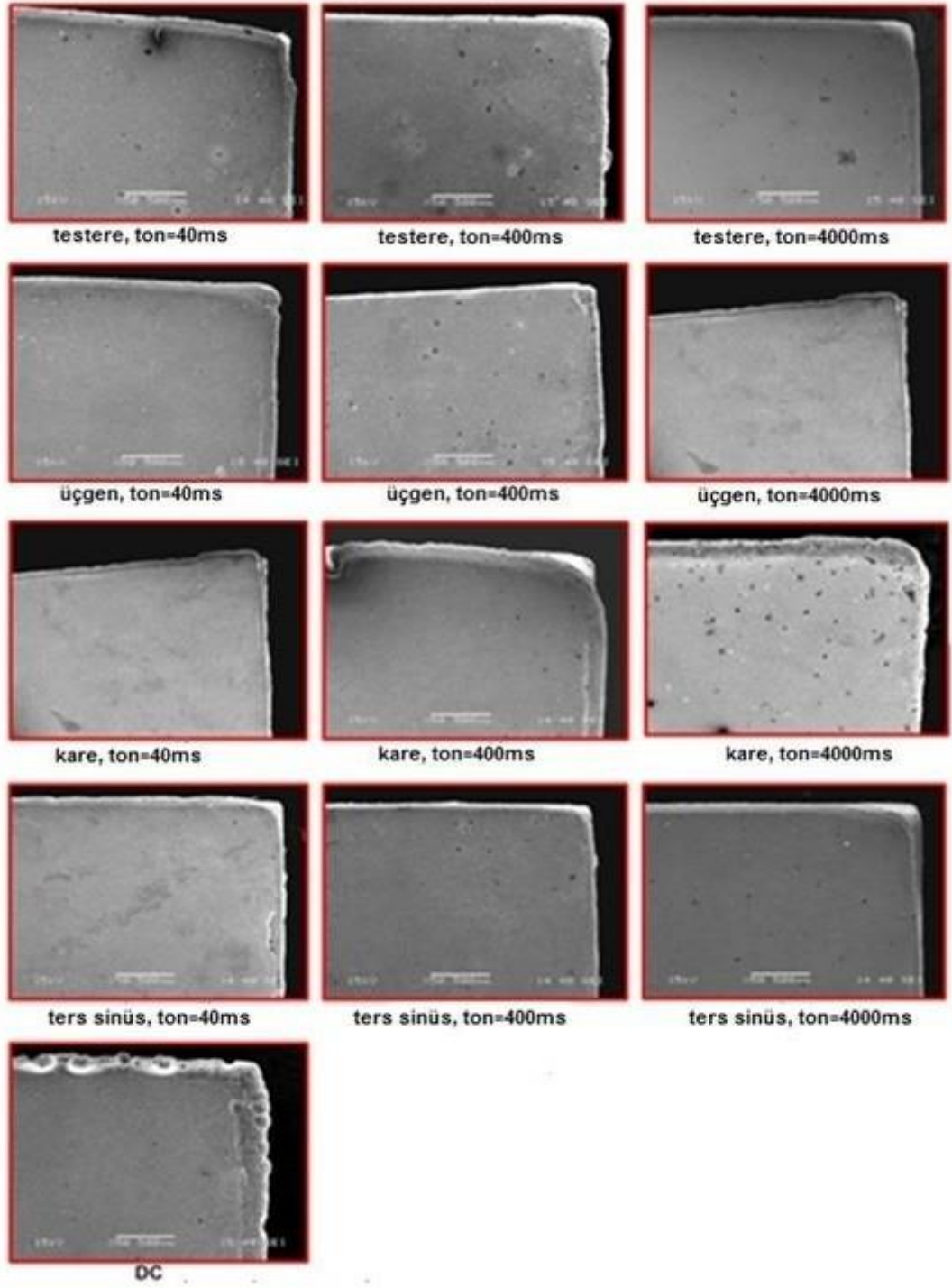
Yapılan ilk deneyde, I_{max} sabit (1500mA) olmak üzere, 30 dk'lık kaplama deneyleri negatif işaretlere sahip olmayan 4 farklı akım dalga şekli (testere, üçgen, kare, ters sinüs) ve her bir akım dalga şekli, 3 farklı değişim hızı kullanılarak, bunun yanı sıra yine $I_{max} = 1500mA$ olmak üzere doğru akım kullanılarak bakır altlık üzerine 5 farklı Ni kaplama deneyi gerçekleştirildi ve sonuçlar karşılaştırıldı, Şekil 4.1 ve Şekil 4.2.

Şekil 4.1.'de testere, üçgen, sinüs, kare ve doğru akım işaretlerine sahip akım dalga formlarıyla yapılmış olan yüzey kaplama sonuçlarına ait yüksek çözünürlüklü SEM fotoğrafları verilmiştir. Bu kaplama tabakalarının uygulanan farklı akım türlerindeki tane yapısına (morfolojisine) bakıldığında, $T_{on}=4000ms$ olan yarım sinüs işaretine sahip olan akım türüyle yapılmış kaplama sonucunun, diğer akım türlerine göre daha ince taneli olduğu, dolayısıyla daha düzgün bir kaplama yüzeyine sahip olduğu gözlemlenmiştir. Ayrıca, DC işaret ile kıyaslandığında kare, üçgen ve testere akım dalga türlerinin de mikro yapı sonuçlarının doğru akıma göre daha iyi olduğu analiz edilmiştir. Buna göre; üretmiş olduğumuz çok fonksiyonlu akım üretici ile gerçekleştirdiğimiz farklı akım türleriyle yapılan kaplamalar, literatürdeki çalışmalarda kullanılan geleneksel akım yöntemi ile yapılan (DC ve PC) kaplamalardan daha başarılı yüzey sonuçları ortaya koymuştur.

Şekil 4.2.'deki köşe kaplama sonuçları analiz edildiğinde en düzgün köşe kaplama sonucunun yine $T_{on}= 4000ms$ ve tepe akımı 1500mA değerine sahip olan yarım sinüs işaretine sahip akım dalga formuyla yapılan köşe kaplama sonucu olduğu elde edilmiştir.



Şekil 4.1. I_{max} 1500mA olan Deney 1 için yüzey kaplama sonuçlarının 15kV / x3000 ve 15kV / x10000 SEM fotoğrafları.

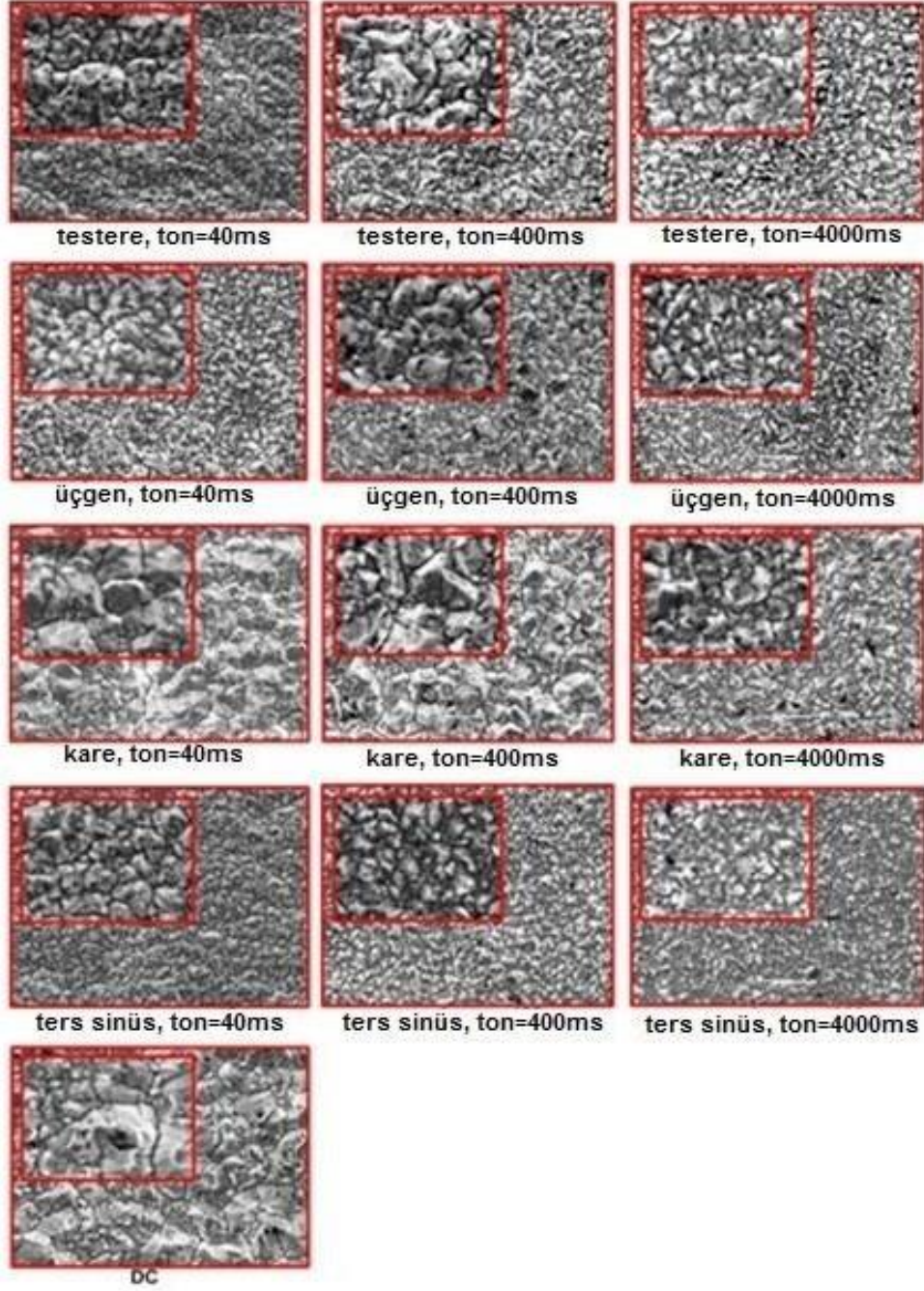


Şekil 4.2. I_{max} 1500mA olan Deney 1 için Köşe kaplama sonuçlarının 15kV / x50 SEM fotoğrafları.

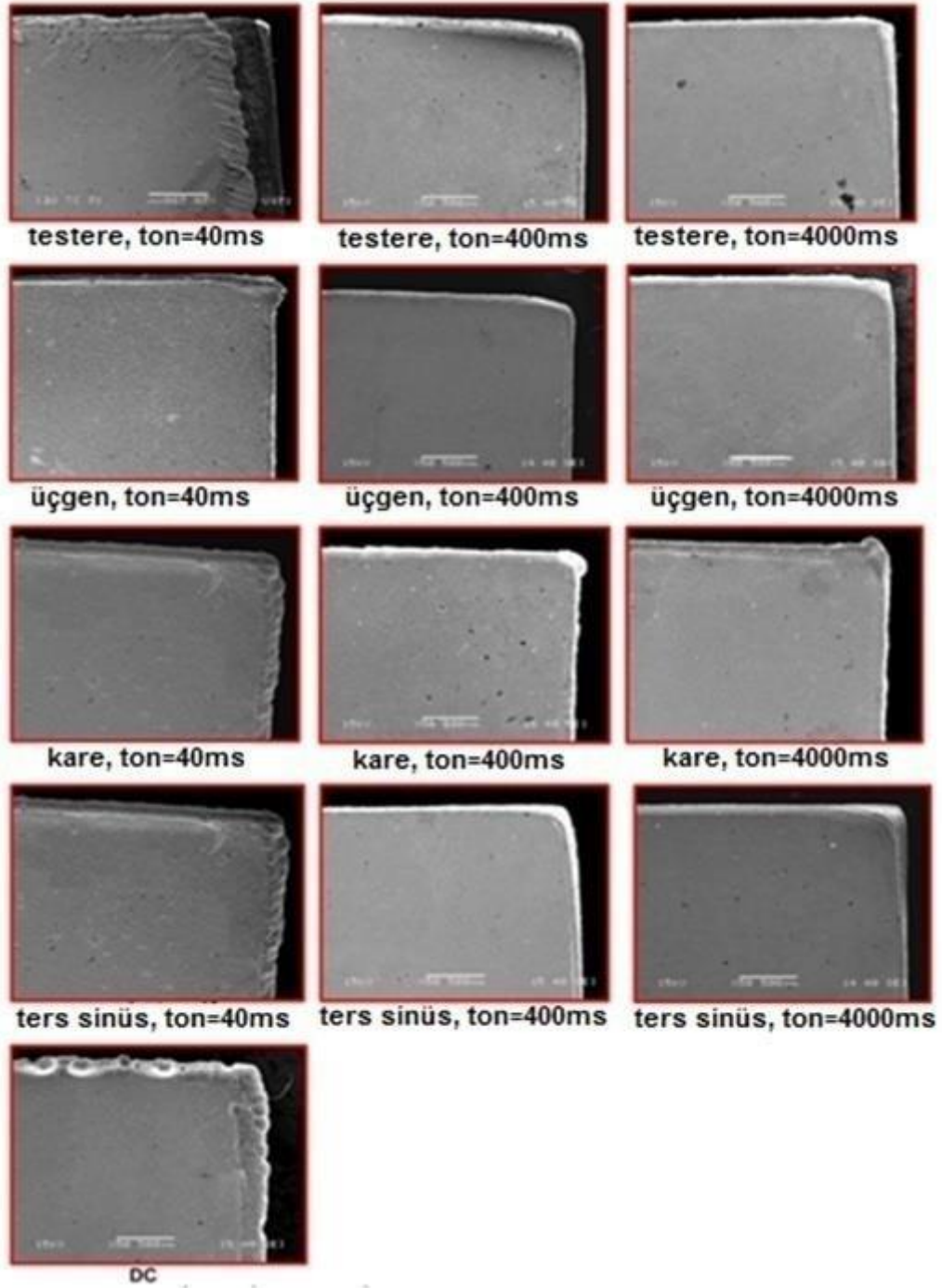
4.1.2. Deney 2 taramalı elektron mikroskobu (SEM) analizi

Yapılan ikinci deneyde, I_{max} sabit (1500mA) olmak üzere, 30dk lık kaplama deneyleri negatif işaretlere sahip, 4 farklı akım dalga şekli ve her bir akım dalga şekli, (doğru akım dışındaki dalga türleri) 3 farklı değişim hızı kullanılarak, bakır altlık

üzerine Ni kaplama deneyleri gerçekleştirilmiş ve sonuçlar karşılaştırılmıştır, Şekil 4.3 ve Şekil 4.4.



Şekil 4.3. I_{max} 1500mA olan Deney 2 için yüzey kaplama sonuçlarının 15kV / x3000 ve 15kV / x10000 SEM fotoğrafları.



Şekil 4.4. I_{max} 1500mA olan Deneysel 2 için Köşe kaplama sonuçlarının 15kV / x50 SEM fotoğrafları.

İkinci deneyde negatif genlikli işarete sahip akım türlerini kullanmamızın nedeni, literatürde benzer çalışmaların negatif işaret içermeyen kaplamalardan daha başarılı olduğunun gösterilmiş olmasıdır. Örneğin, 2018 yılında Isao Matsui ve Naoki Omura'nın yapmış olduğu çalışmada, PRC türündeki kaplamalarda ters akımın etkisi ile

kaplamalarda oluşan çözünenin, diğer akım türlerinde yapılan kaplamalara göre, proses kusurlarını azaltmakta daha etkili olduğu kanıtlanmıştır (Matsui and N. Omura, 2018). Ayrıca 2006 yılında yapılan bir başka araştırmanın sonuçları, PRC kaplama tekniği sayesinde, daha ince taneli, daha pürüzsüz ve kompakt kaplamalar elde edilebileceğini ortaya koymuştur. Bunun nedeninin ise, darbeli ters akım uygulanması sırasında katodun artan elektrokimyasal polarizasyonu ve buna bağlı olarak çekirdekleşme oranındaki artma ile, daha ince taneli ve daha pürüzsüz kaplamalar elde edilebileceği ortaya konulmuştur (Chang vd., 2006).

Yapılan 2. deneyin mikro yapı sonuçları incelendiğinde, en düzgün yüzey kaplama sonucunun ton=4000ms olan negatif genlik değerine sahip, ters sinüs akım dalga formu ile yapılan kaplama olduğu görülmektedir. Ayrıca köşe kaplama başarıları karşılaştırıldığında yine ton=4000 ms olan negatif genlikli ters sinüs işareti ile yapılan kaplamanın diğer kaplamalardan daha iyi sonuç verdiği ortaya konulmuştur. Ayrıca bu tez çalışması sırasında yapmış olduğumuz deney sonuçlarına ilişkin analizlerimizi içeren bilimsel yayıncımız IEEE Access'de "Increasing Surface Coating Quality Using New Generation Current Waveforms" adı altında yayınlanmıştır (Muhurcu and Kulekci, 2020).

5. SONUÇLAR ve ÖNERİLER

5.1. Sonuçlar

Elektrolitik kaplama üretiminde "DC", "Pulse" ve "Pulse Reverse" akım dalga formları darbe frekanslarının sayısı ve genlik değerleri değiştirilerek "darbe" tabanlı çalışmalar çeşitlendirilmeye çalışılsa da sonuç olarak kaplama çalışmaları bu 3 akım türü etrafında gerçekleştirilmektedir. Ancak, mühendislik ve matematikte birçok fonksiyon ve bunlara ait dalga biçimleri mevcuttur. Bu tez çalışmasında DC, PC ve PRC akım yöntemlerinin dışında farklı dalga şekillerine sahip akımlar kullanılarak elektrodpozisyon yöntemi ile gerçekleştirilecek yüzey kaplamalarının bu üç yöntemle kıyasla kaplama başarılarında iyileştirme sağlayabileceği öngörülmüştü. Bu öngörüğü gözlemleyebilmek amacıyla DC, PC, PRC akım dalga şekilleri ve bunların dışında da çeyrek Sin/Cos, yarım Sin/Cos, basamak girişi için 1. derece sistem çıkışı, basamak girişi için 2. derece sistem çıkışı ve rampa akım dalga şekilleri oluşturabilen yeni nesil bir kaplama cihazı geliştirildi. Bu cihaz, bilgisayar yazılımı üzerinden kablosuz olarak kontrol edilebilir hale getirildi. Kullanıcı için periyodu 15 farklı fonksiyona kadar oluşturabilecek bir yazılım platformu tasarlandı. Kullanıcı tarafından akım periyoduna ait bilgi akım cihazına kablosuz olarak gönderilerek cihazın kullanıcının göndermiş olduğu parametreler doğrultusunda çalışması kontrol altına alındı. Yeni nesil akım kaynağı cihazı ve bilgisayar yazılımı, Yazılım Mühendisliği (C#, pc yazılımı), Elektrik-Elektronik Mühendisliği (devre topolojisi) ve Kontrol Mühendisliği (YSA, akım kontrol algoritması) disiplinlerinden yararlanılarak gerçekleştirildi.

Sonuç olarak, tasarımı ve imalatı gerçekleştirilen yeni nesil akım kaynağı cihazı kullanılarak, geleneksel akım dalga formları dışında farklı akım dalga formları kullanılarak kaplama deneyleri yapılmış ve başarılı sonuçlar elde edilmiştir. Elde ettiğimiz deneysel sonuçlar ile elektrodpozisyon yöntemiyle yapılan kaplamalarda "DC", "Pulse" ve "Pulse Reverse" dışındaki farklı fonksiyonlara sahip akımlar kullanıldığında, kaplama tabakalarının çok daha başarılı bir şekilde üretilebileceği gösterilmiştir.

5.2. Öneriler

Daha sonra yapılacak çalışmalarda, tasarımını ve imalatını gerçekleştirmiş olduğumuz akım üretici sistemi ile çok farklı akım senaryoları oluşturularak; farklı malzemelerle elektrolitik tabanlı kaplama deneyleri gerçekleştirilebilir ve ayrıca bu çalışmaların yapısal, tribolojik ve manyetik özellikleri detaylı olarak incelenebilir. Bunun yanı sıra kaplama banyo bileşenleri değiştirilerek kaplama üzerine olan etkileri incelenebilir. Çeşitli disiplinler taranarak farklı akım dalga biçimleri araştırılabilir. Doğrusal-doğrusal olmayan, periyodik-periyodik olmayan, hayvan sesleri ve doğa sesleri başlıkları altında araştırılacak dalga formları sınıflandırılabilir ve bu dalga formlarını üretebilecek yeni bir cihaz geliştirilebilir. Geliştirilecek olan bu cihaza, birden fazla akım dalga formunu elektroliz sıvısına uygulayabilecek yeni bir tasarım kazandırılabilir. Gerçekleştirilecek olan bu çok anotlu kaplama yöntemi ile mevcut her akım dalga formunun kendisine ait olumlu parametreleri sayesinde kaplama kalitesinin daha da iyileştirilebileceğini öngörmekteyiz.

KAYNAKÇA

- Almeida, E., (2001). Surface Treatments and Coatings for Metals. A General Overview. 1. Surface Treatments, Surface Preparation, and the Nature of Coatings. *Ind. Eng. Chem. Res.*, 40(1):3–14.
- Alshmri, F., (2014). Metallic Coatings: Al-Zn Alloys. *Advanced Materials Research*, 915–916:608–611.
- Arunsunai Kumar, K., Paruthimal Kalaignan, G., Muralidharan, V.S., (2013). Direct and Pulse Current Electrodeposition of Ni–W–TiO₂ Nanocomposite Coatings. *Ceramics International*, 39(3):2827–2834.
- Awang, M., Khalili, A. A., and Pedapati, S. R., (2019), . A Review: Thin Protective Coating for Wear Protection in High-Temperature Application. *Metals-Open Access Metallurgy Journal*, 10(1):42.
- Bahrololoom, M.E., Sani, R., (2005). The Influence of Pulse Plating Parameters on The Hardness and Wear Resistance of Nickel–Alumina Composite Coatings. *Surface and Coatings Technology*, 192(2-3):154–163.
- Bakhit, B., Akbari, A., (2012). Effect of particle size and co-deposition technique on hardness and corrosion properties of Ni–Co/SiC composite coatings. *Surface and Coatings Technology*, 206(23):4964–4975.
- Balasubramanian, A., Srikumar, D.S., Raja, G., Saravanan, G., Mohan, S., (2009). Effect of Pulse Parameter on Pulsed Electrodeposition of Copper on Stainless Steel. *Surface Engineering*, 25(5):389–392.
- Bhushan B. and Gupta, B.K., (1991). *Handbook of Tribology: Materials Coatings and Surface Treatments*. Krieger yayınları , 1168 sayfa, isbn: 1-57524-050-5.
- Borkar, T., (2010). Electrodeposition of Nickel Composite Coatings. Degree of Master of science, Mechanical Engineering Mumbai University Mumbai, Maharashtra, India, p.95.
- Borkar, T., Harimkar, S.P., (2011). Effect of Electrodeposition Conditions and Reinforcement Content on Microstructure and Tribological Properties of Nickel Composite Coatings. *Surface and Coatings Technology*, 205(17-18):4124–4134.
- Boukhouiete, A. and Creus, J., (2015). Nickel Deposits Obtained by Continuous and Pulsed Electrodeposition Processes. *Journal of Materials and Environmental Science*, 6(7):1840-1844.
- Chandrasekar, M.S., Pushpavanam, M., (2008). Pulse and Pulse Reverse Plating— Conceptual, Advantages and Applications. *Electrochimica Acta*, 53:3313–3322.
- Chang, L.M., An, M.Z., Shi, S.Y., (2006). Microstructure and Characterization of Ni-Co/Al₂O₃ Composite Coatings by Pulse Reversal Electrodeposit. *Materials Chemistry and Physics*, 100(2-3):395–399.
- Chang, L.M., Chen, D., Liu, J.H., Zhang, R.J., (2009). Effects of Different Plating Modes on Microstructure and Corrosion Resistance of Zn–Ni alloy Coatings. *Journal of Alloys and Compounds*, 479(1-2):489–493.

- Chen, G., Zhang, L., Arinez, J., Biller, S., (2013). Energy-Efficient Production Systems Through Schedule-Based Operations. *IEEE Trans. Automat. Sci. Eng.*, 10(1):27–37.
- Chen, L., Wang, L., Zeng, Z., Xu, T., (2006). Influence of Pulse Frequency on the Microstructure and Wear Resistance of Electrodeposited Ni–Al₂O₃ Composite Coatings. *Surface and Coatings Technology*, 201:599–605.
- Cheraghi, M. S., Allahkaram, S. R., Towhidi, N., Khonsari, S. K., & Rabizadeh, T. (2011). Nano Cobalt Coating Corrosion Behavior Obtained by DC and PC Electrodeposition Process. 2011 IEEE Nanotechnology Materials and Devices Conference, <https://doi.org/10.1109/NMDC.2011.6155348>.
- Chung, C.-K., Chang, W.-T., (2015). Electrochemical Deposition and Mechanical Property Enhancement of the Nickel and Nickel-Cobalt Films, in: Nee, A.Y.C. (Ed.), *Handbook of Manufacturing Engineering and Technology*. Springer London, p.2891–2927.
- Di Bari, G.A., (2000). Electrodeposition of Nickel. *Modern Electroplating*, Fifth Edition, Wiley, p.79-114.
- Estrada, M. E., (2008). Model-Based Framework For Alloy Electrodeposition Processes. University of Tulsa, p. 82.
- Fan, H., (2010). Electroplating of Compound Ni–SiC Coatings and Improvement of Wear Resistance. *Key Engineering Materials*, 426–427:399–402.
- Fotovvati, B., Namdari, N., Dehghanhadikolaei, A., (2019). On Coating Techniques for Surface Protection: A Review. *Journal of Manufacturing and Materials Processing*, 3(1):28.
- Frey, A.A., Wozniak, N.R., Nagi, T.B., Keller, M.P., Lunderberg, J.M., Peaslee, G.F., DeYoung, P.A., Hampton, J.R., (2011). Analysis of Electrodeposited Nickel-Iron Alloy Film Composition Using Particle-Induced X-Ray Emission. *International Journal of Electrochemistry*, 2011:1–7.
- Gaoqi, D., Chunquan, H., Congying, L., Jun, G., (2013). A Weighted First-Order Statistical Method for Time-Varying Channel and DC-offset Estimation Using Superimposed Training. *IEEE Communications Letters*, 17(5):852–855.
- Gomes, A., Pereira, I., Fernandez, B., Pereiro, R., (2011). Electrodeposition of Metal Matrix Nanocomposites: Improvement of the Chemical Characterization Techniques, in: Reddy, B. (Ed.), *Advances in Nanocomposites - Synthesis, Characterization and Industrial Applications*. InTech, (2011):503-526.
- Gyftou, P., Pavlatou, E.A., Spyrellis, N., (2008). Effect of Pulse Electrodeposition Parameters on the Properties of Ni/nano-SiC Composites. *Applied Surface Science*, 254(18):5910–5916.
- Habibzadeh Moradi, E., Jafarzadeh, K., Borji, S., Abbaszadeh, H., (2015). Pulse Electrodeposition as A New Approach in Electrowinning of High Purity Cobalt from WC–Co Scraps. Part I: The effect of frequency and duty cycle. *Minerals Engineering*, 77:10–16.

- Harb, S.V., Trentin, A., Torrico, R.F.O., Pulcinelli, S.H., Santilli, C.V., Hammer, P., (2017). Organic-Inorganic Hybrid Coatings for Corrosion Protection of Metallic Surfaces. *New Technologies in Protective Coatings*, InTech Open, (2017):19-51.
- Hofer, M., Schrodli, M., (2020). Investigation of Permanent Magnet assisted Synchronous Reluctance Machines for Traction Drives in High Power Flux Weakening Operation. 2020 IEEE Transportation Electrification Conference & Expo (ITEC), IEEE, Chicago, IL, USA, 335–339.
- Hou, K.H., Ger, M.D., Wang, L.M., Ke, S.T., (2002). The Wear Behaviour of Electro-Codeposited Ni–SiC Composites. *Wear* 253(2002):994–1003.
- Hridya, I., Srinivas, S., (2015). Carrier overlapped PWM switching schemes for a cascaded three level voltage source inverter. 2015 IEEE International Conference on Industrial Technology (ICIT), IEEE, Seville, 302(2015):3028–3033.
- Imanian Ghazanlou, S., Shokuhfar, A., Navazani, S., Yavari, R., 2016. Influence of Pulse Electrodeposition Parameters on Microhardness, Grain Size and Surface Morphology of Ni–Co/SiO₂ Nanocomposite Coating. *Bulletin of Material Science*, 39(2016):1185–1195.
- Islam, Md.K., Zaved, Md.M., Siddiky, A.M., Al Mamun, K.A., (2016). A comparative analysis among PWM control Z-Source Inverter with conventional PWM Inverter for induction motor drive. 2016 International Conference on Innovations in Science, Engineering and Technology (ICISSET), IEEE, Dhaka, 1–6.
- Jegan, A., (2018). Pulse Electrodeposition of Ni/nano-Al₂O₃ Composite Coatings on Cast Iron Cylinder Liner. *Materials Research*, 21(3):1-11.
- Jha, A., Singh, B., (2017). Bridgeless buck-boost PFC converter for multistring LED driver. 2017 IEEE Industry Applications Society Annual Meeting, IEEE, Cincinnati, OH, 1–8.
- Jheng, C.-S., Fu, P.-H., Yang, J.-Y., Hsu, W.-C., Hwang, Y.-S., Chen, J.-J., (2019). Design and Implementation of Fast Transient Response Buck Converter with New Current-Mode Controlled Techniques. 2019 IEEE International Symposium on Circuits and Systems (ISCAS), IEEE, Sapporo, Japan, 1–4.
- Kakani, S. L., Kakani, A., (2004). *Material Science*, New Age International Limited, Publishers, New Delhi, p. 640.
- Kasturibai, S., Kalaiganan, G.P., (2014). Pulse Electrodeposition and Corrosion Properties of Ni–Si₃N₄ Nanocomposite Coatings. *Bulletin of Material Science*, 37(3):721–728.
- Kim, S.K., Yoo, H.J., (1998). Formation of Bilayer Ni–SiC Composite Coatings by Electrodeposition. *Surface and Coatings Technology*, 108–109(1998):564–569.
- Kraiem, H., Shaaban, S.M., (2020). Energy Optimization of An Electric Car Using Losses Minimization and Intelligent Predictive Torque Control. *Journal of Algorithms & Computational Technology*, 0:1-15.
- Kumar, B.H., Lokhande, Makarand.M., (2017). Analysis of PWM Techniques on Multilevel Cascaded H-Bridge Three Phase Inverter. 2017 Recent Developments in Control, Automation & Power Engineering (RDCAPE), IEEE, Noida, 465–470.

- Lajevardi, S.A., Shahrabi, T., (2010). Effects of Pulse Electrodeposition Parameters on the Properties of Ni-TiO₂ Nanocomposite Coatings. *Applied Surface Science*, 256(22):6775–6781.
- Lange, F., Langwald, J., Hirzinger, G., (1999). Predictive Feedforward Control for High Speed Tracking Tasks. 1999 European Control Conference (ECC), IEEE, Karlsruhe, 4537–4542.
- Li, R., Hou, Y., Dong, Q., Su, P., Ju, P., Liang, J., (2018). Wear and Corrosion Resistance of Co-P Coatings: The Effects of Current Modes. *RSC Advances*, 8:895–903.
- Lin, P.-C., Sun, M.-K., Kung, C., Chiueh, T.-D., (2019). FloatSD: A New Weight Representation and Associated Update Method for Efficient Convolutional Neural Network Training. *IEEE J. Emerg. Sel. Topics in Circuits and Systems*, 9(2):267–279.
- Liu, Y., Yu, Y.-X., Zhang, W.-D., (2012). Photoelectrochemical Properties of Ni-doped Fe₂O₃ Thin Films Prepared by Electrodeposition. *Electrochimica Acta*, 59:121–127.
- Lopatkin, N.N., (2017). Voltage Source Multilevel Inverter Voltage Quality Comparison Under Multicarrier Sinusoidal PWM and Space Vector PWM of Two Delta Voltages. *International Multi-Conference on Engineering, Computer and Information Sciences (SIBIRCON)*, Novosibirsk, Russia, 439-444.
- Lukic, B.Z., Jovanovic, K.M., Kvasceev, G.S., (2016). Feedforward Neural Network for Controlling Qbmove Maker Pro Variable Stiffness Actuator. 2016 13th Symposium on Neural Networks and Applications (NEUREL), IEEE, Belgrade, Serbia, 1–4.
- Mahdavi, S., Allahkaram, S.R., (2015). Composition, Characteristics and Tribological Behavior of Cr, Co-Cr and Co-Cr/TiO₂ Nano-Composite Coatings Electrodeposited from Trivalent Chromium Based Baths. *Journal of Alloys and Compounds* 635:150–157.
- Mahmud, Md.W.B., Alam, A.Z., Rahman, D.A., (2018). Improvement of Active Power Factor Correction Circuit for Switch Mode Power Supply Using Fly Back and Boost Topology. 2018 7th International Conference on Computer and Communication Engineering (ICCCE), IEEE, Kuala Lumpur, 437–440.
- Matsui, I., Omura, N., (2018). Comparison of Tensile Properties of Bulk Nanocrystalline Ni-W Alloys Electrodeposited by Direct, Pulsed, and Pulsed-Reverse Currents. *Materials Transactions*, 59(1):123–128.
- Mbugua, N.S., Kang, M., Zhang, Y., Ndiithi, N.J., V. Bertrand, G., Yao, L., (2020). Electrochemical Deposition of Ni, NiCo Alloy and NiCo-Ceramic Composite Coatings—A Critical Review. *Materials*, 13(16):3475.
- Melciu, D., Maidee, N., (2015). Pulse-Electroplating: Process Parameters and Their Influence on the Formed Microstructure. Department of Materials and Manufacturing Technology, Chalmers University of Technology, Gothenburg, Sweden, p.48.
- More, S.M., Patil, D.R., Patil, P.S., (2018). Power Quality Improved Bridgeless Multiple Output Switched Mode Power Supply. 2018 International Conference On Advances in Communication and Computing Technology (ICACCT), IEEE, Sangamner, 161–167.

- Muhurcu, G., Kose, E., Muhurcu, A., Kuyumcu, A., (2017). Parameter Optimization of PI Controller by PSO for Optimal Controlling of a Buck converter's output. 2017 International Artificial Intelligence and Data Processing Symposium (IDAP), IEEE, Malatya, 1–6.
- Muhurcu, G., Kulekci, M.K., (2020). Increasing Surface Coating Quality Using New Generation Current Waveforms. *IEEE Access*, 8:147358–147369.
- M. Mulukutla, 2008. Pulse Electrodeposition Of Co-W And Its Composite Coatings For Hard Chrome Replacements. Andhra University , India p. 101.
- Nigam, S., Patel, S.K., Mahapatra, S.S., Sharma, N., Ghosh, K.S., (2015). Nickel Coating on High Strength Low Alloy Steel by Pulse Current Deposition. *Materials Science and Engineering*, 75(012024):1-10.
- Nusbaum, T., Rosen, B.A., Gileadi, E., Eliaz, N., 2015. Effect of Pulse On-Time and Peak Current Density on Pulse Plated Re-Ni Alloys. *J. Electrochem. Soc.*, 162:D250–D255.
- Pavithra, C.L.P., Sarada, B.V., Rajulapati, K.V., Rao, T.N., Sundararajan, G., (2015). A New Electrochemical Approach for the Synthesis of Copper-Graphene Nanocomposite Foils with High Hardness. *Sci Rep*, 4(1):1-7.
- Pavlatou, E.A., Spyrellis, N., (2008). Influence of Pulse Plating Conditions on the Structure and Properties of Pure and Composite Nickel Nanocrystalline Coatings. *Russian Journal Electrochemistry*, 44:745–754.
- Presuel-Moreno, F., Jakab, M.A., Tailleart, N., Goldman, M., Scully, J.R., (2008). Corrosion-Resistant Metallic Coatings. *Materials Today*, 11(10):14–23.
- Qing Song, Yilei Wu, Yeng Chai Soh, (2008). Robust Adaptive Gradient-Descent Training Algorithm for Recurrent Neural Networks in Discrete Time Domain. *IEEE Transactions on Neural Networks*, 19(11):1841–1853.
- Qu, D., Zhou, S., (2010). On Control Structure Scheme for Nonlinear System Based on ANN Feedforward/Inverse Models and Simulation. 2010 3rd International Congress on Image and Signal Processing (CISP), IEEE, Yantai, China, 3640–3644.
- Rafi, M.A.H., Rennie, R., Larsen, J., Bauman, J., (2020). Investigation of Fast Charging and Battery Swapping Options for Electric Haul Trucks in Underground Mines. 2020 IEEE Transportation Electrification Conference & Expo (ITEC), IEEE, Chicago, IL, USA, 1081–1087.
- Raghavendra, C.R., Basavarajappa, S., Sogalad, I., (2018). Electrodeposition of Ni-Nano Composite Coatings: A Review. *Inorganic and Nano-Metal Chemistry*, 48(12):583–598.
- Reddy, R.M., Praveen, B.M., Chandrappa, K.G., Nayana, K.O., (2016). Generation of Ni–Si₃N₄ Nanocomposites by DC, PC and PRC Electrodeposition Methods. *Surface Engineering*, 32(7):501–507.
- Ren, Z., Meng, N., Shehzad, K., Xu, Y., Qu, S., Yu, B., Luo, J.K., (2015). Mechanical Properties of Nickel-Graphene Composites Synthesized by Electrochemical Deposition. *Nanotechnology*, 26(6):065706.

- Sajjadnejad, M., Setoudeh, N., Mozafari, A., Isazadeh, A., Omidvar, H., 2017. Alkaline Electrodeposition of Ni–ZnO Nanocomposite Coatings: Effects of Pulse Electroplating Parameters. *Transactions of the Indian Institute of Metals*, 70(6):1533–1541.
- Shantala, Sudheer, M.L., (2018). Reduction of Radiated Emission from Offline Switch Mode Power Supply. 2018 15th International Conference on ElectroMagnetic Interference & Compatibility (INCEMIC), IEEE, Bengaluru (Bangalore), India, 1–4.
- Sharma, A., Bhattacharya, S., Das, S., Das, K., (2014). A Study on the Effect of Pulse Electrodeposition Parameters on the Morphology of Pure Tin Coatings. *Metallurgical and Materials Transactions A*, 45(10):4610–4622.
- Shen, M., Zhan, Z.-H., Chen, W.-N., Gong, Y.-J., Zhang, J., Li, Y., (2014). Bi-Velocity Discrete Particle Swarm Optimization and Its Application to Multicast Routing Problem in Communication Networks. *IEEE Transactions on Industrial Electronics*, 61(12):7141–7151.
- Shibe, V., Chawla, V., (2013). An Overview of Research Work in Surface Coating. *International Journal of Research in Mechanical Engineering & Technology (IJRMET)*, 3(2):85-88.
- Shourgeshty, M., Aliofkhaezai, M., Karimzadeh, A., (2019). Study on Functionally Graded Zn–Ni–Al₂O₃ Coatings Fabricated by Pulse-Electrodeposition. *Surface Engineering*, 35(2):167–176.
- Sidky, P.S., Hocking, M.G., (1999). Review of Inorganic Coatings and Coating Processes for Reducing Wear and Corrosion. *British Corrosion Journal*, 34(3):171–183.
- Singh, S., Perera, C., Kish, G.J., Salmon, J., (2019). Low Harmonic Loss PWM for a Dual Inverter Drive using a Floating Capacitor Inverter. 2019 IEEE Energy Conversion Congress and Exposition (ECCE), IEEE, Baltimore, MD, USA, 5981–5988.
- Sobha Jayakrishnan, D., (2012). Electrodeposition: the versatile technique for nanomaterials, in: *Corrosion Protection and Control Using Nanomaterials*. (V.S. Saji, R. Cook Editors.), Woodhead Publishing Limited, Cambridge, 86–125.
- Spanou, S., Pavlatou, E.A., (2010). Pulse Electrodeposition of Ni/nano-TiO₂ composites: Effect of Pulse Frequency on Deposits Properties. *Journal of Applied Electrochemistry*, 40(7):1325–1336.
- Srivastava, M., Grips, V.K.W., Rajam, K.S., (2007). Electrochemical Deposition and Tribological Behaviour of Ni and Ni–Co Metal Matrix Composites with SiC Nano-Particles. *Applied Surface Science*, 253(8):3814–3824.
- Thiemig, D., Lange, R., Bund, A., (2007). Influence of Pulse Plating Parameters on the Electrodeposition of Matrix Metal Nanocomposites. *Electrochimica Acta*, 52(25):7362-7371.
- Torabinejad, V., Aliofkhaezai, M., Assareh, S., Allahyarzadeh, M.H., Rouhaghdam, A.S., (2017a). Electrodeposition of Ni-Fe Alloys, Composites, and Nano Coatings—A review. *Journal of Alloys and Compounds*, 691: 841–859.

- Ullah, I., Fayaz, M., Naveed, N., Kim, D., (2020). ANN Based Learning to Kalman Filter Algorithm for Indoor Environment Prediction in Smart Greenhouse. *IEEE Access*, 8:159371–159388.
- Vb, S., (2014). An Overview on the Preparation, Characterization and Properties of Electrodeposited-Metal Matrix Nanocomposites. *NSTOA*, 1(3):1-20.
- Veerachary, M., Misal, S., (2020). Single-switch Semi-Quadratic Buck Converter. 2020 IEEE International Conference on Power Electronics, Smart Grid and Renewable Energy (PESGRE2020), IEEE, Cochin, India, 1–6.
- Wasekar, N.P., Latha, S.M., Ramakrishna, M., Rao, D.S., Sundararajan, G., (2016). Pulsed Electrodeposition and Mechanical Properties of Ni-W/SiC Nano-Composite Coatings. *Materials & Design*, 112:140–150.
- Wiechmann, E.P., Morales, A.S., Aqueveque, P., (2010). Improving Productivity and Energy Efficiency in Copper Electrowinning Plants, 26(4):1264-1270.
- Wu, J., Jiang, Y., Johnson, C., Liu, X., (2008). DC Electrodeposition of Mn–Co Alloys on Stainless Steels for SOFC Interconnect Application. *Journal of Power Sources*, 177(2):376–385.
- Xu, J., Wu, J., Wan, B., Xin, W., Ge, Z., (2020). A Novel Approach for Metallic Coating Detection Through Analogizing Between Coil Impedance and Plane Wave Impedance. *NDT & E International*, 116:102308.
- Yuen, Y.H., Lazar, M., Butler, H., (2019). Data-driven Neural Feedforward Controller Design for Industrial Linear Motors. 2019 23rd International Conference on System Theory, Control and Computing (ICSTCC), IEEE, Sinaia, Romania, 461–467.
- Yusoff, N.H.N., Mamat, O., Isa, M.C., (2016). A Review of Electrodeposition Techniques and Variables for Production of Nickel-Cobalt/Alumina (Ni-Co/Al₂O₃) Composite coatings. *Asian Research Publishing Network*, 11(22):13331-13336.
- Yüksel, B., (2017). Ni-Fe-B ve Co-Ni-Fe-B Üçlü ve Dörtlü Alaşımların Elektrolitik Olarak Kaplanması. İstanbul Teknik Üniversitesi Fen Bilimleri Enstitüsü, Doktora Tezi.
- Zangari, G., (2018). Fundamentals of Electrodeposition, in: *Encyclopedia of Interfacial Chemistry*. Elsevier, 141–160.
- Zhou, Y., Zhao, G., Zhang, H., (2010). Fabrication and Wear Properties of Co-Deposited Ni-Cr Nanocomposite Coatings. *Transactions of Nonferrous Metals Society of China*, 20(1):104–109.

EKLER

Ek 1. Bilgisayar Yazılımı

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;
using System.Net.Sockets;
using System.IO;

namespace AKIM_KAYANAGI
{
    public partial class Form1 : Form
    {
        /*TCPClient variable created here to be thrown around in different
        threads & funcs*/
        private TcpClient myClient;
        private NetworkStream stream;

        double it;
        double tSimTum;
        double i_tt;
        double dt = 0.0005;
        double di;

        double f ;
        double iMax ;
        double Tdalga ;//dalga periyodu

        double to_d1;
        double z1;

        double k1_sys2;
        double k2_sys2;
        double k3_sys2;
        double wn;
        double ksi;
        double z2;

        double kx, ky, kz, ko;
        double iox, ioy, ioz;
        double khiz;

        double akim_k1;// bir önceki akim

        double i_ort;
        Int32 i_say;
        Int16 i_tab;

        string dosya_yolu_basla;
```

```

Int16 mikro_komut; /*
    * 1- yeni deęer
    * 2- deęer al
    * 3-akım baslat
    * 4-akım durdur
    */

//Int16 islem_basamaklari;// 1-veri gönderiliyor,2-start,3-stop

/*
    if (char.IsControl(e.KeyChar) || char.IsDigit(e.KeyChar)) e.Handled =
false;
        else if ((e.KeyChar == '-') & (textBox2.Text.Length == 0)) e.Handled
= false; else e.Handled = true;
    */

public Form1()
{
    InitializeComponent();
}

private void radioButton1_CheckedChanged(object sender, EventArgs e)
{
    pictureBox2.Image = Image.FromFile(Application.StartupPath + "\\resim"
+ "\\basamak.png");
}

void ciz_basamak()
{
    double akim;
    akim = 0;

    for (it = 0; it < Tdalga; it += dt)
    {
        if (i_tt < tSimTum)
        {
            akim = iMax ;
            chart1.Series["Current"].Points.AddXY(i_tt, akim + akim_k1);
            i_tt += dt;

            i_ort += akim + akim_k1;
            i_say ++;
        }
    }
    i_tt -= dt;
    akim_k1 += akim;
}

void ciz_csir()
{
    double akim;
    akim = 0;
    for (it = 0; it < Tdalga / 4; it += dt)

```

```

    {
        if (i_tt < tSimTum)
        {
            akim = iMax * Math.Sin(it * 2 * Math.PI * f);
            chart1.Series["Current"].Points.AddXY(i_tt, akim + akim_k1);
            i_tt += dt;

            i_ort += akim + akim_k1;
            i_say++;
        }
    }
    i_tt -= dt;
    akim_k1 += akim;
}

```

void ciz_ysin()

```

{
    double akim;
    akim = 0;
    for (it = 0; it < Tdalga / 2; it += dt)
    {
        if (i_tt < tSimTum)
        {
            akim = iMax * Math.Sin(it * 2 * Math.PI * f);
            chart1.Series["Current"].Points.AddXY(i_tt, akim + akim_k1);
            i_tt += dt;

            i_ort += akim + akim_k1;
            i_say++;
        }
    }
    i_tt -= dt;
    akim_k1 += akim;
}

```

void ciz_ccos()

```

{
    double akim;
    akim = 0;
    for (it = 0; it < Tdalga / 4; it += dt)
    {
        if (i_tt < tSimTum)
        {
            akim = iMax * Math.Cos(it * 2 * Math.PI * f);
            chart1.Series["Current"].Points.AddXY(i_tt, akim + akim_k1);
            i_tt += dt;

            i_ort += akim + akim_k1;
            i_say++;
        }
    }
    i_tt -= dt;
    akim_k1 += akim;
}

```

void ciz_ycos()

```

{
    double akim;

```

```

akim = 0;
for (it = 0; it < Tdalga / 2; it += dt)
{
    if (i_tt < tSimTum)
    {
        akim = iMax * Math.Cos(it * 2 * Math.PI * f);
        chart1.Series["Current"].Points.AddXY(i_tt, akim + akim_k1);
        i_tt += dt;

        i_ort += akim + akim_k1;
        i_say++;
    }
}
i_tt -= dt;
akim_k1 += akim;
}

void ciz_rampa()
{
    double akim;
    akim = 0;
    for (it = 0; it < Tdalga; it += dt)
    {
        if (i_tt < tSimTum)
        {
            akim = di*it;
            chart1.Series["Current"].Points.AddXY(i_tt, akim + akim_k1);
            i_tt += dt;

            i_ort += akim + akim_k1;
            i_say++;
        }
    }
    i_tt -= dt;
    akim_k1 += akim;
}

void ciz_sysd1()
{
    double akim;
    akim = 0;
    for (it = 0; it < Tdalga; it += dt)
    {
        if (i_tt < tSimTum)
        {
            akim = iMax*dt*z1;
            chart1.Series["Current"].Points.AddXY(i_tt, akim + akim_k1);

            z1 = (1 - (dt - to_d1) * z1)/to_d1;

            i_tt += dt;

            i_ort += akim + akim_k1;
            i_say++;
        }
    }
    i_tt -= dt;
}

```

```

    akim_k1 += akim;
}

void ciz_sysd2()
{
    double akim;
    double zg;
    akim = 0;
    for (it = 0; it < Tdalga; it += dt)
    {
        if (i_tt < tSimTum)
        {
            akim = z2 * k1_sys2;

            if (akim > 15000) akim = 15000;
            else if (akim < -15000) akim = -15000;

            zg = z2;
            z2 = z1;
            z1 = iMax - k2_sys2 * z1 - k3_sys2 * zg;

            chart1.Series["Current"].Points.AddXY(i_tt, akim + akim_k1);

            i_tt += dt;

            i_ort += akim + akim_k1;
            i_say++;
        }
    }
    i_tt -= dt;
    akim_k1 += akim;
}

void ciz_lorenz()
{
    double Xx_z1, Xy_z1, Xz_z1;
    double Xx_z, Xy_z, Xz_z;
    double a, b, c;
    double Xc, Yc, Zc;
    double x_o, y_o, z_o;
    double Ts, i;
    double xyz_o;

    Ts = 0.02; //lorenz örnekleme frekansı,
    Xx_z1 = 0.5;
    Xy_z1 = -1;
    Xz_z1 = 0.5;

    Xx_z = Xx_z1;
    Xy_z = Xy_z1;
    Xz_z = Xz_z1;

    a = 10;
    b = 2.66667;
    c = 28;
}

```

**for (i = i_tt; i < tSimTum; i += 0.02 / khiz)// Lorenz örnekleme
frekansı düşürülerek hız artırılıyor..**

```

{
    Xx_z1 = Xx_z;
    Xy_z1 = Xy_z;
    Xz_z1 = Xz_z;

    Xc = Xx_z1 * Ts;
    Yc = Xy_z1 * Ts;
    Zc = Xz_z1 * Ts;

    Xx_z = Xx_z1 + (Yc - Xc) * a;
    Xy_z = Xy_z1 + (Xc * c - Xc * Zc - Yc);
    Xz_z = Xz_z1 + (Xc * Yc - Zc * b);

    x_o = (Xc + iox) * kx*ko;
    if (x_o > 10000) x_o = 10000;
    if (x_o < -10000) x_o = -10000;

    y_o = (Yc + ioy) * ky*ko;
    if (y_o > 10000) y_o = 10000;
    if (y_o < -10000) y_o = -10000;

    z_o = (Zc + ioz) * kz*ko;
    if (z_o > 10000) z_o = 10000;
    if (z_o < -10000) z_o = -10000;

    xyz_o = x_o + y_o + z_o;

    if (xyz_o > 10000) xyz_o = 10000;
    if (xyz_o < -10000) xyz_o = -10000;

    if (radioButton13.Checked == true)
chart1.Series["Current"].Points.AddXY(i, x_o);
    else if (radioButton14.Checked == true)
chart1.Series["Current"].Points.AddXY(i, y_o);
    else if (radioButton15.Checked == true)
chart1.Series["Current"].Points.AddXY(i, z_o);
    else if (radioButton16.Checked == true)
chart1.Series["Current"].Points.AddXY(x_o, y_o);
    else if (radioButton17.Checked == true)
chart1.Series["Current"].Points.AddXY(x_o, z_o);
    else if (radioButton18.Checked == true)
chart1.Series["Current"].Points.AddXY(y_o, z_o);
    else if (radioButton12.Checked == true)
chart1.Series["Current"].Points.AddXY(i, xyz_o);

    i_tt += 0.02 / khiz;

    i_ort += xyz_o;
    i_say++;
}

i_ort /=i_say;
if (Math.Abs( i_ort) < 1 ) i_ort = 0;
if ((int)i_ort < -100000) textBox182.Text = "-";else

```

```

        if ((int)i_ort > 100000) textBox182.Text = "-";else
        textBox182.Text = Convert.ToString((int)i_ort);
    }

    void fonksiyon_1()
    {
        f = 1000 / Convert.ToDouble(textBox3.Text);
        iMax = Convert.ToDouble(textBox2.Text);
        Tdalga = 1 / f;//dalga peryodu

        di = iMax / Tdalga;

        to_d1 = Tdalga / 5;
        z1 = 0;

        //ksi = Math.Sqrt(Math.Pow(Math.Log(0.99) / Math.PI, 2) / (1 +
        Math.Pow(Math.Log(0.99) / Math.PI, 2)));
        ksi = Convert.ToDouble(textBox12.Text);
        ksi /= 10;
        wn = 5 / (ksi * Tdalga);

        k1_sys2 = wn * wn * dt * dt;
        k2_sys2 = 2 * ksi * wn * dt - 2;
        k3_sys2 = 1 + wn * wn * dt * dt - 2 * ksi * wn * dt;
        z2 = 0;

        kx = Convert.ToDouble(textBox4.Text);
        ky = Convert.ToDouble(textBox6.Text);
        kz = Convert.ToDouble(textBox8.Text);

        iox = Convert.ToDouble(textBox5.Text);
        ioy = Convert.ToDouble(textBox7.Text);
        ioz = Convert.ToDouble(textBox9.Text);

        ko = Convert.ToDouble(textBox10.Text);
        khiz = Convert.ToDouble(textBox11.Text);

        akim_k1 = Convert.ToDouble(textBox167.Text);

        if (radioButton1.Checked) ciz_basamak();
        else if (radioButton2.Checked) ciz_csini();
        else if (radioButton3.Checked) ciz_ysin();
        else if (radioButton5.Checked) ciz_ccos();
        else if (radioButton6.Checked) ciz_ycos();
        else if (radioButton8.Checked) ciz_rampa();
        else if (radioButton10.Checked) ciz_sysd1();
        else if (radioButton19.Checked) ciz_sysd2();
        else if (radioButton20.Checked) ciz_lorenz();
    }

    void fonksiyon_2()
    {
        // if (iMax > akim_k1) i_tt -= dt;

        f = 1000 / Convert.ToDouble(textBox22.Text);
    }

```

```

iMax = Convert.ToDouble(textBox23.Text);
Tdalga = 1 / f;//dalga peryodu
di = iMax / Tdalga;

to_d1 = Tdalga / 5;
z1 = 0;

ksi = Convert.ToDouble(textBox21.Text);
ksi /= 10;
wn = 5 / (ksi * Tdalga);

k1_sys2 = wn * wn * dt* dt;
k2_sys2 = 2 * ksi * wn * dt - 2;
k3_sys2 = 1 + wn * wn * dt * dt - 2 * ksi * wn * dt;
z2 = 0;

kx = Convert.ToDouble(textBox20.Text);
ky = Convert.ToDouble(textBox16.Text);
kz = Convert.ToDouble(textBox19.Text);

iox = Convert.ToDouble(textBox18.Text);
ioy = Convert.ToDouble(textBox14.Text);
ioz = Convert.ToDouble(textBox17.Text);

ko = Convert.ToDouble(textBox15.Text);
khi_z = Convert.ToDouble(textBox13.Text);

//akim_k1 += Convert.ToDouble(textBox168.Text);
akim_k1 = Convert.ToDouble(textBox168.Text);

if (radioButton27.Checked) ciz_basamak();
else if (radioButton26.Checked) ciz_csın();
else if (radioButton25.Checked) ciz_ysin();
else if (radioButton24.Checked) ciz_ccos();
else if (radioButton23.Checked) ciz_ycos();
else if (radioButton22.Checked) ciz_rampa();
else if (radioButton21.Checked) ciz_sysd1();
else if (radioButton9.Checked) ciz_sysd2();
else if (radioButton7.Checked) ciz_lorenz();
}

void fonksiyon_3()
{
    f = 1000 / Convert.ToDouble(textBox33.Text);
    iMax = Convert.ToDouble(textBox34.Text);
    Tdalga = 1 / f;//dalga peryodu
    di = iMax / Tdalga;

    to_d1 = Tdalga / 5;
    z1 = 0;

    ksi = Convert.ToDouble(textBox32.Text);
    ksi /= 10;
    wn = 5 / (ksi * Tdalga);

    k1_sys2 = wn * wn * dt * dt;
    k2_sys2 = 2 * ksi * wn * dt - 2;
}

```

```

k3_sys2 = 1 + wn * wn * dt * dt - 2 * ksi * wn * dt;
z2 = 0;

kx = Convert.ToDouble(textBox31.Text);
ky = Convert.ToDouble(textBox27.Text);
kz = Convert.ToDouble(textBox30.Text);

iox = Convert.ToDouble(textBox29.Text);
ioy = Convert.ToDouble(textBox25.Text);
ioz = Convert.ToDouble(textBox28.Text);

ko = Convert.ToDouble(textBox26.Text);
khiiz = Convert.ToDouble(textBox24.Text);

//akim_k1 += Convert.ToDouble(textBox169.Text);
akim_k1 = Convert.ToDouble(textBox169.Text);

if (radioButton37.Checked) ciz_basamak();
else if (radioButton36.Checked) ciz_csın();
else if (radioButton35.Checked) ciz_ysin();
else if (radioButton34.Checked) ciz_ccos();
else if (radioButton33.Checked) ciz_ycos();
else if (radioButton32.Checked) ciz_rampa();
else if (radioButton31.Checked) ciz_sysd1();
else if (radioButton30.Checked) ciz_sysd2();
else if (radioButton29.Checked) ciz_lorenz();
}

void fonksiyon_4()
{
    i_tt -= dt;

    f = 1000 / Convert.ToDouble(textBox44.Text);
    iMax = Convert.ToDouble(textBox45.Text);
    Tdalga = 1 / f; //dalga periyodu
    di = iMax / Tdalga;

    to_d1 = Tdalga / 5;
    z1 = 0;

    ksi = Convert.ToDouble(textBox43.Text);
    ksi /= 10;
    wn = 5 / (ksi * Tdalga);

    k1_sys2 = wn * wn * dt * dt;
    k2_sys2 = 2 * ksi * wn * dt - 2;
    k3_sys2 = 1 + wn * wn * dt * dt - 2 * ksi * wn * dt;
    z2 = 0;

    kx = Convert.ToDouble(textBox42.Text);
    ky = Convert.ToDouble(textBox38.Text);
    kz = Convert.ToDouble(textBox41.Text);

    iox = Convert.ToDouble(textBox40.Text);
    ioy = Convert.ToDouble(textBox36.Text);
    ioz = Convert.ToDouble(textBox39.Text);

    ko = Convert.ToDouble(textBox37.Text);

```

```

khiz = Convert.ToDouble(textBox35.Text);

//akim_k1 += Convert.ToDouble(textBox170.Text);
akim_k1 = Convert.ToDouble(textBox170.Text);

if (radioButton47.Checked) ciz_basamak();
else if (radioButton46.Checked) ciz_csın();
else if (radioButton45.Checked) ciz_ysin();
else if (radioButton44.Checked) ciz_ccos();
else if (radioButton43.Checked) ciz_ycos();
else if (radioButton42.Checked) ciz_rampa();
else if (radioButton41.Checked) ciz_sysd1();
else if (radioButton40.Checked) ciz_sysd2();
else if (radioButton39.Checked) ciz_lorenz();
}

void fonksiyon_5()
{
    i_tt -= dt;

    f = 1000 / Convert.ToDouble(textBox55.Text);
    iMax = Convert.ToDouble(textBox56.Text);
    Tdalga = 1 / f; //dalga peryodu
    di = iMax / Tdalga;

    to_d1 = Tdalga / 5;
    z1 = 0;

    ksi = Convert.ToDouble(textBox54.Text);
    ksi /= 10;
    wn = 5 / (ksi * Tdalga);

    k1_sys2 = wn * wn * dt * dt;
    k2_sys2 = 2 * ksi * wn * dt - 2;
    k3_sys2 = 1 + wn * wn * dt * dt - 2 * ksi * wn * dt;
    z2 = 0;

    kx = Convert.ToDouble(textBox53.Text);
    ky = Convert.ToDouble(textBox49.Text);
    kz = Convert.ToDouble(textBox52.Text);

    iox = Convert.ToDouble(textBox51.Text);
    ioy = Convert.ToDouble(textBox47.Text);
    ioz = Convert.ToDouble(textBox50.Text);

    ko = Convert.ToDouble(textBox48.Text);
    khiz = Convert.ToDouble(textBox46.Text);

    //akim_k1 += Convert.ToDouble(textBox171.Text);
    akim_k1 = Convert.ToDouble(textBox171.Text);

    if (radioButton57.Checked) ciz_basamak();
    else if (radioButton56.Checked) ciz_csın();
    else if (radioButton55.Checked) ciz_ysin();
    else if (radioButton54.Checked) ciz_ccos();
    else if (radioButton53.Checked) ciz_ycos();
    else if (radioButton52.Checked) ciz_rampa();
    else if (radioButton51.Checked) ciz_sysd1();
}

```

```

else if (radioButton50.Checked) ciz_sysd2();
else if (radioButton49.Checked) ciz_lorenz();
}

void fonksiyon_6()
{
    i_tt -= dt;

    f = 1000 / Convert.ToDouble(textBox66.Text);
    iMax = Convert.ToDouble(textBox67.Text);
    Tdalga = 1 / f; //dalga peryodu
    di = iMax / Tdalga;

    to_d1 = Tdalga / 5;
    z1 = 0;

    ksi = Convert.ToDouble(textBox65.Text);
    ksi /= 10;
    wn = 5 / (ksi * Tdalga);

    k1_sys2 = wn * wn * dt * dt;
    k2_sys2 = 2 * ksi * wn * dt - 2;
    k3_sys2 = 1 + wn * wn * dt * dt - 2 * ksi * wn * dt;
    z2 = 0;

    kx = Convert.ToDouble(textBox64.Text);
    ky = Convert.ToDouble(textBox60.Text);
    kz = Convert.ToDouble(textBox63.Text);

    iox = Convert.ToDouble(textBox62.Text);
    ioy = Convert.ToDouble(textBox58.Text);
    ioz = Convert.ToDouble(textBox61.Text);

    ko = Convert.ToDouble(textBox59.Text);
    khiz = Convert.ToDouble(textBox57.Text);

    //akim_k1 += Convert.ToDouble(textBox172.Text);
    akim_k1 = Convert.ToDouble(textBox172.Text);

    if (radioButton67.Checked) ciz_basamak();
    else if (radioButton66.Checked) ciz_csın();
    else if (radioButton65.Checked) ciz_ysin();
    else if (radioButton64.Checked) ciz_ccos();
    else if (radioButton63.Checked) ciz_ycos();
    else if (radioButton62.Checked) ciz_rampa();
    else if (radioButton61.Checked) ciz_sysd1();
    else if (radioButton60.Checked) ciz_sysd2();
    else if (radioButton59.Checked) ciz_lorenz();
}

void fonksiyon_7()
{
    i_tt -= dt;

    f = 1000 / Convert.ToDouble(textBox77.Text);
    iMax = Convert.ToDouble(textBox78.Text);
    Tdalga = 1 / f; //dalga peryodu
    di = iMax / Tdalga;

```

```

to_d1 = Tdalga / 5;
z1 = 0;

ksi = Convert.ToDouble(textBox76.Text);
ksi /= 10;
wn = 5 / (ksi * Tdalga);

k1_sys2 = wn * wn * dt * dt;
k2_sys2 = 2 * ksi * wn * dt - 2;
k3_sys2 = 1 + wn * wn * dt * dt - 2 * ksi * wn * dt;
z2 = 0;

kx = Convert.ToDouble(textBox75.Text);
ky = Convert.ToDouble(textBox71.Text);
kz = Convert.ToDouble(textBox74.Text);

iox = Convert.ToDouble(textBox73.Text);
ioy = Convert.ToDouble(textBox69.Text);
ioz = Convert.ToDouble(textBox72.Text);

ko = Convert.ToDouble(textBox70.Text);
khiiz = Convert.ToDouble(textBox68.Text);

//akim_k1 += Convert.ToDouble(textBox173.Text);
akim_k1 = Convert.ToDouble(textBox173.Text);

if (radioButton77.Checked) ciz_basamak();
else if (radioButton76.Checked) ciz_csini();
else if (radioButton75.Checked) ciz_ysin();
else if (radioButton74.Checked) ciz_ccos();
else if (radioButton73.Checked) ciz_ycos();
else if (radioButton72.Checked) ciz_rampa();
else if (radioButton71.Checked) ciz_sysd1();
else if (radioButton70.Checked) ciz_sysd2();
else if (radioButton69.Checked) ciz_lorenz();
}

void fonksiyon_8()
{
    i_tt -= dt;

    f = 1000 / Convert.ToDouble(textBox88.Text);
    iMax = Convert.ToDouble(textBox89.Text);
    Tdalga = 1 / f; //dalga peryodu
    di = iMax / Tdalga;

    to_d1 = Tdalga / 5;
    z1 = 0;

    ksi = Convert.ToDouble(textBox87.Text);
    ksi /= 10;
    wn = 5 / (ksi * Tdalga);

    k1_sys2 = wn * wn * dt * dt;
    k2_sys2 = 2 * ksi * wn * dt - 2;
    k3_sys2 = 1 + wn * wn * dt * dt - 2 * ksi * wn * dt;
    z2 = 0;
}

```

```

kx = Convert.ToDouble(textBox86.Text);
ky = Convert.ToDouble(textBox82.Text);
kz = Convert.ToDouble(textBox85.Text);

iox = Convert.ToDouble(textBox84.Text);
ioy = Convert.ToDouble(textBox80.Text);
ioz = Convert.ToDouble(textBox83.Text);

ko = Convert.ToDouble(textBox81.Text);
khiiz = Convert.ToDouble(textBox79.Text);

//akim_k1 += Convert.ToDouble(textBox174.Text);
akim_k1 = Convert.ToDouble(textBox174.Text);

if (radioButton87.Checked) ciz_basamak();
else if (radioButton86.Checked) ciz_csini();
else if (radioButton85.Checked) ciz_ysin();
else if (radioButton84.Checked) ciz_ccos();
else if (radioButton83.Checked) ciz_ycos();
else if (radioButton82.Checked) ciz_rampa();
else if (radioButton81.Checked) ciz_sysd1();
else if (radioButton80.Checked) ciz_sysd2();
else if (radioButton79.Checked) ciz_lorenz();
}

void fonksiyon_9()
{
    i_tt -= dt;

    f = 1000 / Convert.ToDouble(textBox99.Text);
    iMax = Convert.ToDouble(textBox100.Text);
    Tdalga = 1 / f; //dalga peryodu
    di = iMax / Tdalga;

    to_d1 = Tdalga / 5;
    z1 = 0;

    ksi = Convert.ToDouble(textBox98.Text);
    ksi /= 10;
    wn = 5 / (ksi * Tdalga);

    k1_sys2 = wn * wn * dt * dt;
    k2_sys2 = 2 * ksi * wn * dt - 2;
    k3_sys2 = 1 + wn * wn * dt * dt - 2 * ksi * wn * dt;
    z2 = 0;

    kx = Convert.ToDouble(textBox97.Text);
    ky = Convert.ToDouble(textBox93.Text);
    kz = Convert.ToDouble(textBox96.Text);

    iox = Convert.ToDouble(textBox95.Text);
    ioy = Convert.ToDouble(textBox91.Text);
    ioz = Convert.ToDouble(textBox94.Text);

    ko = Convert.ToDouble(textBox92.Text);
    khiiz = Convert.ToDouble(textBox90.Text);

```

```

//akim_k1 += Convert.ToDouble(textBox175.Text);
akim_k1 = Convert.ToDouble(textBox175.Text);

if (radioButton97.Checked) ciz_basamak();
else if (radioButton96.Checked) ciz_csın();
else if (radioButton95.Checked) ciz_ysin();
else if (radioButton94.Checked) ciz_ccos();
else if (radioButton93.Checked) ciz_ycos();
else if (radioButton92.Checked) ciz_rampa();
else if (radioButton91.Checked) ciz_sysd1();
else if (radioButton90.Checked) ciz_sysd2();
else if (radioButton89.Checked) ciz_lorenz();
}

void fonksiyon_10()
{
    i_tt -= dt;

    f = 1000 / Convert.ToDouble(textBox110.Text);
    iMax = Convert.ToDouble(textBox111.Text);
    Tdalga = 1 / f;//dalga peryodu
    di = iMax / Tdalga;

    to_d1 = Tdalga / 5;
    z1 = 0;

    ksi = Convert.ToDouble(textBox109.Text);
    ksi /= 10;
    wn = 5 / (ksi * Tdalga);

    k1_sys2 = wn * wn * dt * dt;
    k2_sys2 = 2 * ksi * wn * dt - 2;
    k3_sys2 = 1 + wn * wn * dt * dt - 2 * ksi * wn * dt;
    z2 = 0;

    kx = Convert.ToDouble(textBox108.Text);
    ky = Convert.ToDouble(textBox104.Text);
    kz = Convert.ToDouble(textBox107.Text);

    iox = Convert.ToDouble(textBox106.Text);
    ioy = Convert.ToDouble(textBox102.Text);
    ioz = Convert.ToDouble(textBox105.Text);

    ko = Convert.ToDouble(textBox103.Text);
    khiiz = Convert.ToDouble(textBox101.Text);

    //akim_k1 += Convert.ToDouble(textBox176.Text);
    akim_k1 = Convert.ToDouble(textBox176.Text);

    if (radioButton107.Checked) ciz_basamak();
    else if (radioButton106.Checked) ciz_csın();
    else if (radioButton105.Checked) ciz_ysin();
    else if (radioButton104.Checked) ciz_ccos();
    else if (radioButton103.Checked) ciz_ycos();
    else if (radioButton102.Checked) ciz_rampa();
    else if (radioButton101.Checked) ciz_sysd1();
    else if (radioButton100.Checked) ciz_sysd2();
    else if (radioButton99.Checked) ciz_lorenz();
}

```

```

}

void fonksiyon_11()
{
    i_tt -= dt;

    f = 1000 / Convert.ToDouble(textBox121.Text);
    iMax = Convert.ToDouble(textBox122.Text);
    Tdalga = 1 / f; //dalga peryodu
    di = iMax / Tdalga;

    to_d1 = Tdalga / 5;
    z1 = 0;

    ksi = Convert.ToDouble(textBox120.Text);
    ksi /= 10;
    wn = 5 / (ksi * Tdalga);

    k1_sys2 = wn * wn * dt * dt;
    k2_sys2 = 2 * ksi * wn * dt - 2;
    k3_sys2 = 1 + wn * wn * dt * dt - 2 * ksi * wn * dt;
    z2 = 0;

    kx = Convert.ToDouble(textBox119.Text);
    ky = Convert.ToDouble(textBox115.Text);
    kz = Convert.ToDouble(textBox118.Text);

    iox = Convert.ToDouble(textBox117.Text);
    ioy = Convert.ToDouble(textBox113.Text);
    ioz = Convert.ToDouble(textBox116.Text);

    ko = Convert.ToDouble(textBox114.Text);
    khiz = Convert.ToDouble(textBox112.Text);

    //akim_k1 += Convert.ToDouble(textBox177.Text);
    akim_k1 = Convert.ToDouble(textBox177.Text);

    if (radioButton117.Checked) ciz_basamak();
    else if (radioButton116.Checked) ciz_csin();
    else if (radioButton115.Checked) ciz_ysin();
    else if (radioButton114.Checked) ciz_ccos();
    else if (radioButton113.Checked) ciz_ycos();
    else if (radioButton112.Checked) ciz_rampa();
    else if (radioButton111.Checked) ciz_sysd1();
    else if (radioButton110.Checked) ciz_sysd2();
    else if (radioButton109.Checked) ciz_lorenz();
}

void fonksiyon_12()
{
    i_tt -= dt;

    f = 1000 / Convert.ToDouble(textBox132.Text);
    iMax = Convert.ToDouble(textBox133.Text);
    Tdalga = 1 / f; //dalga peryodu
    di = iMax / Tdalga;

    to_d1 = Tdalga / 5;

```

```

z1 = 0;

ksi = Convert.ToDouble(textBox131.Text);
ksi /= 10;
wn = 5 / (ksi * Tdalga);

k1_sys2 = wn * wn * dt* dt;
k2_sys2 = 2 * ksi * wn * dt - 2;
k3_sys2 = 1 + wn * wn * dt * dt - 2 * ksi * wn * dt;
z2 = 0;

kx = Convert.ToDouble(textBox130.Text);
ky = Convert.ToDouble(textBox126.Text);
kz = Convert.ToDouble(textBox129.Text);

iox = Convert.ToDouble(textBox128.Text);
ioy = Convert.ToDouble(textBox124.Text);
ioz = Convert.ToDouble(textBox127.Text);

ko = Convert.ToDouble(textBox125.Text);
khiz = Convert.ToDouble(textBox123.Text);

//akim_k1 += Convert.ToDouble(textBox178.Text);
akim_k1 = Convert.ToDouble(textBox178.Text);

if (radioButton127.Checked) ciz_basamak();
else if (radioButton126.Checked) ciz_csini();
else if (radioButton125.Checked) ciz_ysin();
else if (radioButton124.Checked) ciz_ccos();
else if (radioButton123.Checked) ciz_ycos();
else if (radioButton122.Checked) ciz_rampa();
else if (radioButton121.Checked) ciz_sysd1();
else if (radioButton120.Checked) ciz_sysd2();
else if (radioButton119.Checked) ciz_lorenz();
}

void fonksiyon_13()
{
    i_tt -= dt;

    f = 1000 / Convert.ToDouble(textBox143.Text);
    iMax = Convert.ToDouble(textBox144.Text);
    Tdalga = 1 / f; //dalga peryodu
    di = iMax / Tdalga;

    to_d1 = Tdalga / 5;
    z1 = 0;

    ksi = Convert.ToDouble(textBox142.Text);
    ksi /= 10;
    wn = 5 / (ksi * Tdalga);

    k1_sys2 = wn * wn * dt * dt;
    k2_sys2 = 2 * ksi * wn * dt - 2;
    k3_sys2 = 1 + wn * wn * dt * dt - 2 * ksi * wn * dt;
    z2 = 0;

    kx = Convert.ToDouble(textBox141.Text);

```

```

ky = Convert.ToDouble(textBox137.Text);
kz = Convert.ToDouble(textBox140.Text);

iox = Convert.ToDouble(textBox139.Text);
ioy = Convert.ToDouble(textBox135.Text);
ioz = Convert.ToDouble(textBox138.Text);

ko = Convert.ToDouble(textBox136.Text);
khiiz = Convert.ToDouble(textBox134.Text);

//akim_k1 += Convert.ToDouble(textBox179.Text);
akim_k1 = Convert.ToDouble(textBox179.Text);

if (radioButton137.Checked) ciz_basamak();
else if (radioButton136.Checked) ciz_csın();
else if (radioButton135.Checked) ciz_ysin();
else if (radioButton134.Checked) ciz_ccos();
else if (radioButton133.Checked) ciz_ycos();
else if (radioButton132.Checked) ciz_rampa();
else if (radioButton131.Checked) ciz_sysd1();
else if (radioButton130.Checked) ciz_sysd2();
else if (radioButton129.Checked) ciz_lorenz();
}

void fonksiyon_14()
{
    i_tt -= dt;

    f = 1000 / Convert.ToDouble(textBox154.Text);
    iMax = Convert.ToDouble(textBox155.Text);
    Tdalga = 1 / f; //dalga peryodu
    di = iMax / Tdalga;

    to_d1 = Tdalga / 5;
    z1 = 0;

    ksi = Convert.ToDouble(textBox153.Text);
    ksi /= 10;
    wn = 5 / (ksi * Tdalga);

    k1_sys2 = wn * wn * dt * dt;
    k2_sys2 = 2 * ksi * wn * dt - 2;
    k3_sys2 = 1 + wn * wn * dt * dt - 2 * ksi * wn * dt;
    z2 = 0;

    kx = Convert.ToDouble(textBox152.Text);
    ky = Convert.ToDouble(textBox148.Text);
    kz = Convert.ToDouble(textBox151.Text);

    iox = Convert.ToDouble(textBox150.Text);
    ioy = Convert.ToDouble(textBox146.Text);
    ioz = Convert.ToDouble(textBox149.Text);

    ko = Convert.ToDouble(textBox147.Text);
    khiiz = Convert.ToDouble(textBox145.Text);

    //akim_k1 += Convert.ToDouble(textBox180.Text);
    akim_k1 = Convert.ToDouble(textBox180.Text);
}

```

```

        if (radioButton147.Checked) ciz_basamak();
        else if (radioButton146.Checked) ciz_csın();
        else if (radioButton145.Checked) ciz_ysin();
        else if (radioButton144.Checked) ciz_ccos();
        else if (radioButton143.Checked) ciz_ycos();
        else if (radioButton142.Checked) ciz_rampa();
        else if (radioButton141.Checked) ciz_sysd1();
        else if (radioButton140.Checked) ciz_sysd2();
        else if (radioButton139.Checked) ciz_lorenz();
    }

void fonksiyon_15()
{
    i_tt -= dt;

    f = 1000 / Convert.ToDouble(textBox165.Text);
    iMax = Convert.ToDouble(textBox166.Text);
    Tdalga = 1 / f; //dalga periyodu
    di = iMax / Tdalga;

    to_d1 = Tdalga / 5;
    z1 = 0;

    ksi = Convert.ToDouble(textBox164.Text);
    ksi /= 10;
    wn = 5 / (ksi * Tdalga);

    k1_sys2 = wn * wn * dt * dt;
    k2_sys2 = 2 * ksi * wn * dt - 2;
    k3_sys2 = 1 + wn * wn * dt * dt - 2 * ksi * wn * dt;
    z2 = 0;

    kx = Convert.ToDouble(textBox163.Text);
    ky = Convert.ToDouble(textBox159.Text);
    kz = Convert.ToDouble(textBox162.Text);

    iox = Convert.ToDouble(textBox161.Text);
    ioy = Convert.ToDouble(textBox157.Text);
    ioz = Convert.ToDouble(textBox160.Text);

    ko = Convert.ToDouble(textBox158.Text);
    khiz = Convert.ToDouble(textBox156.Text);

    //akim_k1 += Convert.ToDouble(textBox181.Text);
    akim_k1 = Convert.ToDouble(textBox181.Text);

    if (radioButton157.Checked) ciz_basamak();
    else if (radioButton156.Checked) ciz_csın();
    else if (radioButton155.Checked) ciz_ysin();
    else if (radioButton154.Checked) ciz_ccos();
    else if (radioButton153.Checked) ciz_ycos();
    else if (radioButton152.Checked) ciz_rampa();
    else if (radioButton151.Checked) ciz_sysd1();
    else if (radioButton150.Checked) ciz_sysd2();
    else if (radioButton149.Checked) ciz_lorenz();
}

```

```

private void button1_Click(object sender, EventArgs e)
{
    textBox182.Text = " ";

    tSimTum = 0;
    if (!radioButton11.Checked) tSimTum+=Convert.ToInt32(textBox3.Text);
    if (!radioButton4.Checked) tSimTum +=
Convert.ToInt32(textBox22.Text);
    if (!radioButton28.Checked) tSimTum +=
Convert.ToInt32(textBox33.Text);
    if (!radioButton38.Checked) tSimTum +=
Convert.ToInt32(textBox44.Text);
    if (!radioButton48.Checked) tSimTum +=
Convert.ToInt32(textBox55.Text);
    if (!radioButton58.Checked) tSimTum +=
Convert.ToInt32(textBox66.Text);
    if (!radioButton68.Checked) tSimTum +=
Convert.ToInt32(textBox77.Text);
    if (!radioButton78.Checked) tSimTum +=
Convert.ToInt32(textBox88.Text);
    if (!radioButton88.Checked) tSimTum +=
Convert.ToInt32(textBox99.Text);
    if (!radioButton98.Checked) tSimTum +=
Convert.ToInt32(textBox110.Text);
    if (!radioButton108.Checked) tSimTum +=
Convert.ToInt32(textBox121.Text);
    if (!radioButton118.Checked) tSimTum +=
Convert.ToInt32(textBox132.Text);
    if (!radioButton128.Checked) tSimTum +=
Convert.ToInt32(textBox143.Text);
    if (!radioButton138.Checked) tSimTum +=
Convert.ToInt32(textBox154.Text);
    if (!radioButton148.Checked) tSimTum +=
Convert.ToInt32(textBox165.Text);
    tSimTum *= 0.001;

    i_tt = 0.0;

    i_ort = 0.0;
    i_say = 0;

    for (i_tt = 0.0; i_tt < tSimTum;i_tt+=dt )
    {
        akim_k1 = 0;

        if (!radioButton11.Checked) { fonksiyon_1(); }
        if (!radioButton4.Checked) fonksiyon_2();
        if (!radioButton28.Checked) fonksiyon_3();
        if (!radioButton38.Checked) fonksiyon_4();
        if (!radioButton48.Checked) fonksiyon_5();
        if (!radioButton58.Checked) fonksiyon_6();
        if (!radioButton68.Checked) fonksiyon_7();
        if (!radioButton78.Checked) fonksiyon_8();
        if (!radioButton88.Checked) fonksiyon_9();
        if (!radioButton98.Checked) fonksiyon_10();
        if (!radioButton108.Checked) fonksiyon_11();
    }
}

```

```

        if (!radioButton118.Checked) fonksiyon_12();
        if (!radioButton128.Checked) fonksiyon_13();
        if (!radioButton138.Checked) fonksiyon_14();
        if (!radioButton148.Checked) fonksiyon_15();
    }

    if (textBox182.Text == " ")// lorenz akımında problem oluşuyor..
    {
        i_ort /=i_say;
        if (Math.Abs( i_ort) < 1 ) i_ort = 0;
        if ((int)i_ort < -100000) textBox182.Text = "-";else
        if ((int)i_ort > 100000) textBox182.Text = "-";else
        textBox182.Text = Convert.ToString((int)i_ort);
    }

    foreach (var series in chart1.Series)
    {
        series.Points.Clear();
    }

    tSimTum = 0.001 * Convert.ToDouble(textBox1.Text);//ms den sn
    çevrildi
    i_tt = 0.0;

    i_ort = 0;
    i_say = 0;

    for (i_tt = 0.0; i_tt < tSimTum; i_tt += dt)
    {
        akim_k1 = 0;

        if (!radioButton11.Checked) fonksiyon_1();
        if (!radioButton4.Checked) fonksiyon_2();
        if (!radioButton28.Checked) fonksiyon_3();
        if (!radioButton38.Checked) fonksiyon_4();
        if (!radioButton48.Checked) fonksiyon_5();
        if (!radioButton58.Checked) fonksiyon_6();
        if (!radioButton68.Checked) fonksiyon_7();
        if (!radioButton78.Checked) fonksiyon_8();
        if (!radioButton88.Checked) fonksiyon_9();
        if (!radioButton98.Checked) fonksiyon_10();
        if (!radioButton108.Checked) fonksiyon_11();
        if (!radioButton118.Checked) fonksiyon_12();
        if (!radioButton128.Checked) fonksiyon_13();
        if (!radioButton138.Checked) fonksiyon_14();
        if (!radioButton148.Checked) fonksiyon_15();
    }

    chart1.ChartAreas[0].CursorX.IsUserEnabled = true;
    chart1.ChartAreas[0].CursorX.IsUserSelectionEnabled = true;
    chart1.ChartAreas[0].CursorY.IsUserEnabled = true;
    chart1.ChartAreas[0].CursorY.IsUserSelectionEnabled = true;

    chart1.ChartAreas[0].CursorX.Interval = 0;
    chart1.ChartAreas[0].CursorY.Interval = 0;
    chart1.ChartAreas[0].AxisX.ScaleView.Zoomable = true;

```

```

        chart1.ChartAreas[0].AxisY.ScaleView.Zoomable = true;

        Console.Beep();
    }

    private void button5_Click(object sender, EventArgs e)
    {
        chart1.ChartAreas[0].AxisX.ScaleView.ZoomReset(0);
        chart1.ChartAreas[0].AxisY.ScaleView.ZoomReset(0);

        Console.Beep();
    }

    private void radioButton11_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox2.Image = null;
    }

    private void radioButton2_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox2.Image = Image.FromFile(Application.StartupPath +
        "\\resim" + "\\csin.png");
    }

    private void radioButton3_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox2.Image = Image.FromFile(Application.StartupPath +
        "\\resim" + "\\ysin.png");
    }

    private void radioButton5_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox2.Image = Image.FromFile(Application.StartupPath +
        "\\resim" + "\\ccos.png");
    }

    private void radioButton6_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox2.Image = Image.FromFile(Application.StartupPath +
        "\\resim" + "\\ycos.png");
    }

    private void radioButton8_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox2.Image = Image.FromFile(Application.StartupPath +
        "\\resim" + "\\rampa.png");
    }

    private void radioButton10_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox2.Image = Image.FromFile(Application.StartupPath +
        "\\resim" + "\\sys1d.png");
    }

    private void radioButton19_CheckedChanged(object sender, EventArgs e)
    {

```

```

        pictureBox2.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\sys2d.png");
    }

    private void radioButton20_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox2.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\lorenz.png");
    }

    private void radioButton27_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox1.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\basamak.png");
    }

    private void radioButton37_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox3.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\basamak.png");
    }

    private void radioButton47_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox4.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\basamak.png");
    }

    private void radioButton57_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox5.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\basamak.png");
    }

    private void radioButton67_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox6.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\basamak.png");
    }

    private void radioButton77_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox7.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\basamak.png");
    }

    private void radioButton87_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox8.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\basamak.png");
    }

    private void radioButton97_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox9.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\basamak.png");
    }

```

```

private void radioButton107_CheckedChanged(object sender, EventArgs e)
{
    pictureBox10.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\basamak.png");
}

private void radioButton117_CheckedChanged(object sender, EventArgs e)
{
    pictureBox11.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\basamak.png");
}

private void radioButton127_CheckedChanged(object sender, EventArgs e)
{
    pictureBox12.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\basamak.png");
}

private void radioButton137_CheckedChanged(object sender, EventArgs e)
{
    pictureBox13.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\basamak.png");
}

private void radioButton147_CheckedChanged(object sender, EventArgs e)
{
    pictureBox14.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\basamak.png");
}

private void radioButton157_CheckedChanged(object sender, EventArgs e)
{
    pictureBox15.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\basamak.png");
}

private void radioButton4_CheckedChanged(object sender, EventArgs e)
{
    pictureBox1.Image = null;
}

private void radioButton28_CheckedChanged(object sender, EventArgs e)
{
    pictureBox3.Image = null;
}

private void radioButton38_CheckedChanged(object sender, EventArgs e)
{
    pictureBox4.Image = null;
}

private void radioButton48_CheckedChanged(object sender, EventArgs e)
{
    pictureBox5.Image = null;
}

```

```

private void radioButton58_CheckedChanged(object sender, EventArgs e)
{
    pictureBox6.Image = null;
}

private void radioButton68_CheckedChanged(object sender, EventArgs e)
{
    pictureBox7.Image = null;
}

private void radioButton78_CheckedChanged(object sender, EventArgs e)
{
    pictureBox8.Image = null;
}

private void radioButton88_CheckedChanged(object sender, EventArgs e)
{
    pictureBox9.Image = null;
}

private void radioButton98_CheckedChanged(object sender, EventArgs e)
{
    pictureBox10.Image = null;
}

private void radioButton108_CheckedChanged(object sender, EventArgs e)
{
    pictureBox11.Image = null;
}

private void radioButton118_CheckedChanged(object sender, EventArgs e)
{
    pictureBox12.Image = null;
}

private void radioButton128_CheckedChanged(object sender, EventArgs e)
{
    pictureBox13.Image = null;
}

private void radioButton138_CheckedChanged(object sender, EventArgs e)
{
    pictureBox14.Image = null;
}

private void radioButton148_CheckedChanged(object sender, EventArgs e)
{
    pictureBox15.Image = null;
}

private void radioButton26_CheckedChanged(object sender, EventArgs e)
{
    pictureBox1.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\csin.png");
}

private void radioButton36_CheckedChanged(object sender, EventArgs e)

```

```

    {
        pictureBox3.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\csin.png");
    }

    private void radioButton46_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox4.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\csin.png");
    }

    private void radioButton56_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox5.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\csin.png");
    }

    private void radioButton66_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox6.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\csin.png");
    }

    private void radioButton76_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox7.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\csin.png");
    }

    private void radioButton86_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox8.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\csin.png");
    }

    private void radioButton96_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox9.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\csin.png");
    }

    private void radioButton106_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox10.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\csin.png");
    }

    private void radioButton116_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox11.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\csin.png");
    }

    private void radioButton126_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox12.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\csin.png");
    }

```

```

private void radioButton136_CheckedChanged(object sender, EventArgs e)
{
    pictureBox13.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\csin.png");
}

private void radioButton146_CheckedChanged(object sender, EventArgs e)
{
    pictureBox14.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\csin.png");
}

private void radioButton156_CheckedChanged(object sender, EventArgs e)
{
    pictureBox15.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\csin.png");
}

private void radioButton25_CheckedChanged(object sender, EventArgs e)
{
    pictureBox1.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\ysin.png");
}

private void radioButton35_CheckedChanged(object sender, EventArgs e)
{
    pictureBox3.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\ysin.png");
}

private void radioButton45_CheckedChanged(object sender, EventArgs e)
{
    pictureBox4.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\ysin.png");
}

private void radioButton55_CheckedChanged(object sender, EventArgs e)
{
    pictureBox5.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\ysin.png");
}

private void radioButton65_CheckedChanged(object sender, EventArgs e)
{
    pictureBox6.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\ysin.png");
}

private void radioButton75_CheckedChanged(object sender, EventArgs e)
{
    pictureBox7.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\ysin.png");
}

private void radioButton85_CheckedChanged(object sender, EventArgs e)

```

```

        {
            pictureBox8.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\ysin.png");
        }

        private void radioButton95_CheckedChanged(object sender, EventArgs e)
        {
            pictureBox9.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\ysin.png");
        }

        private void radioButton105_CheckedChanged(object sender, EventArgs e)
        {
            pictureBox10.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\ysin.png");
        }

        private void radioButton115_CheckedChanged(object sender, EventArgs e)
        {
            pictureBox11.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\ysin.png");
        }

        private void radioButton125_CheckedChanged(object sender, EventArgs e)
        {
            pictureBox12.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\ysin.png");
        }

        private void radioButton135_CheckedChanged(object sender, EventArgs e)
        {
            pictureBox13.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\ysin.png");
        }

        private void radioButton145_CheckedChanged(object sender, EventArgs e)
        {
            pictureBox14.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\ysin.png");
        }

        private void radioButton155_CheckedChanged(object sender, EventArgs e)
        {
            pictureBox15.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\ysin.png");
        }

        private void radioButton24_CheckedChanged(object sender, EventArgs e)
        {
            pictureBox1.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\ccos.png");
        }

        private void radioButton34_CheckedChanged(object sender, EventArgs e)
        {
            pictureBox3.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\ccos.png");
        }

```

```

    }

    private void radioButton44_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox4.Image = Image.FromFile(Application.StartupPath +
        "\\resim" + "\\ccos.png");
    }

    private void radioButton54_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox5.Image = Image.FromFile(Application.StartupPath +
        "\\resim" + "\\ccos.png");
    }

    private void radioButton64_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox6.Image = Image.FromFile(Application.StartupPath +
        "\\resim" + "\\ccos.png");
    }

    private void radioButton74_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox7.Image = Image.FromFile(Application.StartupPath +
        "\\resim" + "\\ccos.png");
    }

    private void radioButton84_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox8.Image = Image.FromFile(Application.StartupPath +
        "\\resim" + "\\ccos.png");
    }

    private void radioButton94_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox9.Image = Image.FromFile(Application.StartupPath +
        "\\resim" + "\\ccos.png");
    }

    private void radioButton104_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox10.Image = Image.FromFile(Application.StartupPath +
        "\\resim" + "\\ccos.png");
    }

    private void radioButton114_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox11.Image = Image.FromFile(Application.StartupPath +
        "\\resim" + "\\ccos.png");
    }

    private void radioButton124_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox12.Image = Image.FromFile(Application.StartupPath +
        "\\resim" + "\\ccos.png");
    }

    private void radioButton134_CheckedChanged(object sender, EventArgs e)
    {

```

```

        pictureBox13.Image = Image.FromFile(Application.StartupPath +
"\resim" + "\\ccos.png");
    }

    private void radioButton144_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox14.Image = Image.FromFile(Application.StartupPath +
"\resim" + "\\ccos.png");
    }

    private void radioButton154_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox15.Image = Image.FromFile(Application.StartupPath +
"\resim" + "\\ccos.png");
    }

    private void radioButton23_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox1.Image = Image.FromFile(Application.StartupPath +
"\resim" + "\\ycos.png");
    }

    private void radioButton33_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox3.Image = Image.FromFile(Application.StartupPath +
"\resim" + "\\ycos.png");
    }

    private void radioButton43_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox4.Image = Image.FromFile(Application.StartupPath +
"\resim" + "\\ycos.png");
    }

    private void radioButton53_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox5.Image = Image.FromFile(Application.StartupPath +
"\resim" + "\\ycos.png");
    }

    private void radioButton63_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox6.Image = Image.FromFile(Application.StartupPath +
"\resim" + "\\ycos.png");
    }

    private void radioButton73_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox7.Image = Image.FromFile(Application.StartupPath +
"\resim" + "\\ycos.png");
    }

    private void radioButton83_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox8.Image = Image.FromFile(Application.StartupPath +
"\resim" + "\\ycos.png");
    }

```

```

    }

    private void radioButton93_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox9.Image = Image.FromFile(Application.StartupPath +
        "\\resim" + "\\ycos.png");
    }

    private void radioButton103_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox10.Image = Image.FromFile(Application.StartupPath +
        "\\resim" + "\\ycos.png");
    }

    private void radioButton113_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox11.Image = Image.FromFile(Application.StartupPath +
        "\\resim" + "\\ycos.png");
    }

    private void radioButton123_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox12.Image = Image.FromFile(Application.StartupPath +
        "\\resim" + "\\ycos.png");
    }

    private void radioButton133_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox13.Image = Image.FromFile(Application.StartupPath +
        "\\resim" + "\\ycos.png");
    }

    private void radioButton143_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox14.Image = Image.FromFile(Application.StartupPath +
        "\\resim" + "\\ycos.png");
    }

    private void radioButton153_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox15.Image = Image.FromFile(Application.StartupPath +
        "\\resim" + "\\ycos.png");
    }

    private void radioButton22_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox1.Image = Image.FromFile(Application.StartupPath +
        "\\resim" + "\\rampa.png");
    }

    private void radioButton32_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox3.Image = Image.FromFile(Application.StartupPath +
        "\\resim" + "\\rampa.png");
    }

```

```

private void radioButton42_CheckedChanged(object sender, EventArgs e)
{
    pictureBox4.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\rampa.png");
}

private void radioButton52_CheckedChanged(object sender, EventArgs e)
{
    pictureBox5.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\rampa.png");
}

private void radioButton62_CheckedChanged(object sender, EventArgs e)
{
    pictureBox6.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\rampa.png");
}

private void radioButton72_CheckedChanged(object sender, EventArgs e)
{
    pictureBox7.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\rampa.png");
}

private void radioButton82_CheckedChanged(object sender, EventArgs e)
{
    pictureBox8.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\rampa.png");
}

private void radioButton92_CheckedChanged(object sender, EventArgs e)
{
    pictureBox9.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\rampa.png");
}

private void radioButton102_CheckedChanged(object sender, EventArgs e)
{
    pictureBox10.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\rampa.png");
}

private void radioButton112_CheckedChanged(object sender, EventArgs e)
{
    pictureBox11.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\rampa.png");
}

private void radioButton122_CheckedChanged(object sender, EventArgs e)
{
    pictureBox12.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\rampa.png");
}

private void radioButton132_CheckedChanged(object sender, EventArgs e)
{
    pictureBox13.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\rampa.png");
}

```

```

    }

    private void radioButton142_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox14.Image = Image.FromFile(Application.StartupPath +
        "\\resim" + "\\rampa.png");
    }

    private void radioButton152_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox15.Image = Image.FromFile(Application.StartupPath +
        "\\resim" + "\\rampa.png");
    }

    private void radioButton21_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox1.Image = Image.FromFile(Application.StartupPath +
        "\\resim" + "\\sys1d.png");
    }

    private void radioButton31_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox3.Image = Image.FromFile(Application.StartupPath +
        "\\resim" + "\\sys1d.png");
    }

    private void radioButton41_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox4.Image = Image.FromFile(Application.StartupPath +
        "\\resim" + "\\sys1d.png");
    }

    private void radioButton51_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox5.Image = Image.FromFile(Application.StartupPath +
        "\\resim" + "\\sys1d.png");
    }

    private void radioButton61_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox6.Image = Image.FromFile(Application.StartupPath +
        "\\resim" + "\\sys1d.png");
    }

    private void radioButton71_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox7.Image = Image.FromFile(Application.StartupPath +
        "\\resim" + "\\sys1d.png");
    }

    private void radioButton81_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox8.Image = Image.FromFile(Application.StartupPath +
        "\\resim" + "\\sys1d.png");
    }

```

```

private void radioButton9_CheckedChanged(object sender, EventArgs e)
{
    pictureBox9.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\sys1d.png");
}

private void radioButton101_CheckedChanged(object sender, EventArgs e)
{
    pictureBox10.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\sys1d.png");
}

private void radioButton111_CheckedChanged(object sender, EventArgs e)
{
    pictureBox11.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\sys1d.png");
}

private void radioButton121_CheckedChanged(object sender, EventArgs e)
{
    pictureBox12.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\sys1d.png");
}

private void radioButton131_CheckedChanged(object sender, EventArgs e)
{
    pictureBox13.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\sys1d.png");
}

private void radioButton141_CheckedChanged(object sender, EventArgs e)
{
    pictureBox14.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\sys1d.png");
}

private void radioButton151_CheckedChanged(object sender, EventArgs e)
{
    pictureBox15.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\sys1d.png");
}

private void radioButton9_CheckedChanged(object sender, EventArgs e)
{
    pictureBox1.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\sys2d.png");
}

private void radioButton30_CheckedChanged(object sender, EventArgs e)
{
    pictureBox3.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\sys2d.png");
}

private void radioButton40_CheckedChanged(object sender, EventArgs e)
{

```

```

        pictureBox4.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\sys2d.png");
    }

    private void radioButton50_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox5.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\sys2d.png");
    }

    private void radioButton60_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox6.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\sys2d.png");
    }

    private void radioButton70_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox7.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\sys2d.png");
    }

    private void radioButton80_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox8.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\sys2d.png");
    }

    private void radioButton90_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox9.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\sys2d.png");
    }

    private void radioButton100_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox10.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\sys2d.png");
    }

    private void radioButton110_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox11.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\sys2d.png");
    }

    private void radioButton120_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox12.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\sys2d.png");
    }

    private void radioButton130_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox13.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\sys2d.png");
    }

```

```

private void radioButton140_CheckedChanged(object sender, EventArgs e)
{
    pictureBox14.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\sys2d.png");
}

private void radioButton150_CheckedChanged(object sender, EventArgs e)
{
    pictureBox15.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\sys2d.png");
}

private void radioButton7_CheckedChanged(object sender, EventArgs e)
{
    pictureBox1.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\lorenz.png");
}

private void radioButton29_CheckedChanged(object sender, EventArgs e)
{
    pictureBox3.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\lorenz.png");
}

private void radioButton39_CheckedChanged(object sender, EventArgs e)
{
    pictureBox4.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\lorenz.png");
}

private void radioButton49_CheckedChanged(object sender, EventArgs e)
{
    pictureBox5.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\lorenz.png");
}

private void radioButton59_CheckedChanged(object sender, EventArgs e)
{
    pictureBox6.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\lorenz.png");
}

private void radioButton69_CheckedChanged(object sender, EventArgs e)
{
    pictureBox7.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\lorenz.png");
}

private void radioButton79_CheckedChanged(object sender, EventArgs e)
{
    pictureBox8.Image = Image.FromFile(Application.StartupPath +
"\\resim" + "\\lorenz.png");
}

private void radioButton89_CheckedChanged(object sender, EventArgs e)
{

```

```

        pictureBox9.Image = Image.FromFile(Application.StartupPath +
"\resim" + "\\lorenz.png");
    }

    private void radioButton99_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox10.Image = Image.FromFile(Application.StartupPath +
"\resim" + "\\lorenz.png");
    }

    private void radioButton109_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox11.Image = Image.FromFile(Application.StartupPath +
"\resim" + "\\lorenz.png");
    }

    private void radioButton119_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox12.Image = Image.FromFile(Application.StartupPath +
"\resim" + "\\lorenz.png");
    }

    private void radioButton129_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox13.Image = Image.FromFile(Application.StartupPath +
"\resim" + "\\lorenz.png");
    }

    private void radioButton139_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox14.Image = Image.FromFile(Application.StartupPath +
"\resim" + "\\lorenz.png");
    }

    private void radioButton149_CheckedChanged(object sender, EventArgs e)
    {
        pictureBox15.Image = Image.FromFile(Application.StartupPath +
"\resim" + "\\lorenz.png");
    }

    private void button2_Click(object sender, EventArgs e)
    {
        Int32 sayi;

        mikro_komut=1;/*1- yazmaya devam*2- k-1 anında gelen data
hatalı,değiştir* 3-akım baslat* 4-akım durdur*/

        //button1.PerformClick();

        i_tab = 1;
        textBox186.Text = "0";

        try
        {
            sayi = Convert.ToInt32(richTextBox1.Text);

            if (sayi > 0)

```

```

        {
            label236.Text = richTextBox1.Text;
            timer1.Enabled = true;
            richTextBox1.Enabled = false;
        }
    }
    catch
    {
        label236.Text = "-";
        richTextBox1.Text = "continuous";
    }

    Console.Beep();
}

void tab1_parametre_gonder()
{
    Int16 fonk;
    if (!radioButton11.Checked)
    {
        i_tab = 1;

        mikro_komut--; /*1- yazmaya devam*2- k-1de gelen data
hatalı,değiştir* 3-akım baslat* 4-akım durdur*/
        if (mikro_komut < 1) mikro_komut = 1;
        mikro_komut *= 15000;

        if(radioButton1.Checked)fonk=1;else
        if(radioButton2.Checked==true)fonk=2;else
        if(radioButton3.Checked==true)fonk=3;else
        if(radioButton5.Checked==true)fonk=4;else
        if(radioButton6.Checked==true)fonk=5;else
        if(radioButton8.Checked==true)fonk=6;else
        if(radioButton10.Checked==true)fonk=7;else
        if(radioButton19.Checked==true)fonk=8;else fonk=9;

        textBox185.Text = Convert.ToString(mikro_komut) + "," +
            Convert.ToString(i_tab) + "," +
            Convert.ToString(fonk) + "," +

            textBox2.Text + "," +
            textBox167.Text + "," +
            textBox3.Text + "," +

            textBox12.Text + "," +

            textBox4.Text + "," +
            textBox6.Text + "," +
            textBox8.Text + "," +
            textBox10.Text + "," +
            textBox11.Text + "," +
            textBox5.Text + "," +
            textBox7.Text + "," +
            textBox9.Text + ",";

        Byte[] data =
        System.Text.Encoding.ASCII.GetBytes(textBox185.Text);
    }
}

```

```

stream.Write(data, 0, data.Length);

mikro_komut /= 15000;

if(radioButton1.Checked)fonk=1;else
if(radioButton2.Checked==true)fonk=2;else
if(radioButton3.Checked==true)fonk=3;else
if(radioButton5.Checked==true)fonk=4;else
if(radioButton6.Checked==true)fonk=5;else
if(radioButton8.Checked==true)fonk=6;else
if(radioButton10.Checked==true)fonk=7;else
if(radioButton19.Checked==true)fonk=8;else fonk=9;

textBox185.Text = Convert.ToString(mikro_komut) + "," +
Convert.ToString(i_tab) + "," +
Convert.ToString(fonk) + "," +

textBox2.Text + "," +
textBox167.Text + "," +
textBox3.Text + "," +

textBox12.Text + "," +

textBox4.Text + "," +
textBox6.Text + "," +
textBox8.Text + "," +
textBox10.Text + "," +
textBox11.Text + "," +
textBox5.Text + "," +
textBox7.Text + "," +
textBox9.Text + ",";
}
textBox187.Text = "F" + Convert.ToString(i_tab);

mikro_komut = 1;
i_tab++;
}

void tab2_parametre_gonder()
{
    Int16 fonk;

    if (!radioButton4.Checked)
    {
        i_tab = 2;

        mikro_komut--; /*1- yazmaya devam*2- k-1de gelen data
hatalı,değiştir* 3-akım baslat* 4-akım durdur*/
        if (mikro_komut < 1) mikro_komut = 1;
        mikro_komut *= 15000;

        if(radioButton27.Checked)fonk=1;else
        if(radioButton26.Checked==true)fonk=2;else
        if(radioButton25.Checked==true)fonk=3;else
        if(radioButton24.Checked==true)fonk=4;else
        if(radioButton23.Checked==true)fonk=5;else
        if(radioButton22.Checked==true)fonk=6;else
        if(radioButton21.Checked==true)fonk=7;else

```

```

if(radioButton9.Checked==true)fonk=8;else fonk=9;

textBox185.Text = Convert.ToString(mikro_komut) + "," +
    Convert.ToString(i_tab) + "," +
    Convert.ToString(fonk) + "," +

    textBox23.Text + "," +
    textBox168.Text + "," +
    textBox22.Text + "," +

    textBox21.Text + "," +

    textBox20.Text + "," +
    textBox16.Text + "," +
    textBox19.Text + "," +
    textBox15.Text + "," +
    textBox13.Text + "," +
    textBox18.Text + "," +
    textBox14.Text + "," +
    textBox17.Text + ",";

    Byte[] data =
System.Text.Encoding.ASCII.GetBytes(textBox185.Text);

stream.Write(data, 0, data.Length);

mikro_komut /= 15000;

if(radioButton27.Checked)fonk=1;else
if(radioButton26.Checked==true)fonk=2;else
if(radioButton25.Checked==true)fonk=3;else
if(radioButton24.Checked==true)fonk=4;else
if(radioButton23.Checked==true)fonk=5;else
if(radioButton22.Checked==true)fonk=6;else
if(radioButton21.Checked==true)fonk=7;else
if(radioButton9.Checked==true)fonk=8;else fonk=9;

textBox185.Text = Convert.ToString(mikro_komut) + "," +
    Convert.ToString(i_tab) + "," +
    Convert.ToString(fonk) + "," +

    textBox23.Text + "," +
    textBox168.Text + "," +
    textBox22.Text + "," +

    textBox21.Text + "," +

    textBox20.Text + "," +
    textBox16.Text + "," +
    textBox19.Text + "," +
    textBox15.Text + "," +
    textBox13.Text + "," +
    textBox18.Text + "," +
    textBox14.Text + "," +
    textBox17.Text + ",";
}
textBox187.Text = "F"+Convert.ToString(i_tab);
mikro_komut = 1;

```

```

        i_tab++;
    }

    void tab3_parametre_gonder()
    {
        Int16 fonk;

        if (!radioButton28.Checked)
        {
            i_tab = 3;

            mikro_komut--; /*1- yazmaya devam*2- k-1de gelen data
hatalı,değiştir* 3-akım baslat* 4-akım durdur*/
            if (mikro_komut < 1) mikro_komut = 1;
            mikro_komut *= 15000;

            if(radioButton37.Checked)fonk=1;else
            if(radioButton36.Checked==true)fonk=2;else
            if(radioButton35.Checked==true)fonk=3;else
            if(radioButton34.Checked==true)fonk=4;else
            if(radioButton33.Checked==true)fonk=5;else
            if(radioButton32.Checked==true)fonk=6;else
            if(radioButton31.Checked==true)fonk=7;else
            if(radioButton30.Checked==true)fonk=8;else fonk=9;

            textBox185.Text = Convert.ToString(mikro_komut) + "," +
                Convert.ToString(i_tab) + "," +
                Convert.ToString(fonk) + "," +

                textBox34.Text + "," +
                textBox169.Text + "," +
                textBox33.Text + "," +

                textBox32.Text + "," +

                textBox31.Text + "," +
                textBox27.Text + "," +
                textBox30.Text + "," +
                textBox26.Text + "," +
                textBox24.Text + "," +
                textBox29.Text + "," +
                textBox25.Text + "," +
                textBox28.Text + ",";

            Byte[] data =
System.Text.Encoding.ASCII.GetBytes(textBox185.Text);

            stream.Write(data, 0, data.Length);

            mikro_komut /= 15000;

            if(radioButton37.Checked)fonk=1;else
            if(radioButton36.Checked==true)fonk=2;else
            if(radioButton35.Checked==true)fonk=3;else
            if(radioButton34.Checked==true)fonk=4;else
            if(radioButton33.Checked==true)fonk=5;else
            if(radioButton32.Checked==true)fonk=6;else
            if(radioButton31.Checked==true)fonk=7;else

```

```

if(radioButton30.Checked==true)fonk=8;else fonk=9;

textBox185.Text = Convert.ToString(mikro_komut) + "," +
    Convert.ToString(i_tab) + "," +
    Convert.ToString(fonk) + "," +

    textBox34.Text + "," +
    textBox169.Text + "," +
    textBox33.Text + "," +

    textBox32.Text + "," +

    textBox31.Text + "," +
    textBox27.Text + "," +
    textBox30.Text + "," +
    textBox26.Text + "," +
    textBox24.Text + "," +
    textBox29.Text + "," +
    textBox25.Text + "," +
    textBox28.Text + ",";
}
textBox187.Text = "F" + Convert.ToString(i_tab);
mikro_komut = 1;
i_tab++;
}

void tab4_parametre_gonder()
{
    int16 fonk;

    if (!radioButton38.Checked)
    {
        i_tab = 4;

        mikro_komut--; /*1- yazmaya devam*2- k-1de gelen data
        hatalı,değiştir* 3-akım baslat* 4-akım durdur*/
        if (mikro_komut < 1) mikro_komut = 1;
        mikro_komut *= 15000;

        if(radioButton47.Checked)fonk=1;else
        if(radioButton46.Checked==true)fonk=2;else
        if(radioButton45.Checked==true)fonk=3;else
        if(radioButton44.Checked==true)fonk=4;else
        if(radioButton43.Checked==true)fonk=5;else
        if(radioButton42.Checked==true)fonk=6;else
        if(radioButton41.Checked==true)fonk=7;else
        if(radioButton40.Checked==true)fonk=8;else fonk=9;

        textBox185.Text = Convert.ToString(mikro_komut) + "," +
            Convert.ToString(i_tab) + "," +
            Convert.ToString(fonk) + "," +

            textBox45.Text + "," +
            textBox170.Text + "," +
            textBox44.Text + "," +

            textBox43.Text + "," +

```

```

        textBox42.Text + "," +
        textBox38.Text + "," +
        textBox41.Text + "," +
        textBox37.Text + "," +
        textBox35.Text + "," +
        textBox40.Text + "," +
        textBox36.Text + "," +
        textBox39.Text + ",";

        Byte[] data =
System.Text.Encoding.ASCII.GetBytes(textBox185.Text);

        stream.Write(data, 0, data.Length);

        mikro_komut /= 15000;

        if (radioButton47.Checked) fonk=1; else
        if (radioButton46.Checked==true) fonk=2; else
        if (radioButton45.Checked==true) fonk=3; else
        if (radioButton44.Checked==true) fonk=4; else
        if (radioButton43.Checked==true) fonk=5; else
        if (radioButton42.Checked==true) fonk=6; else
        if (radioButton41.Checked==true) fonk=7; else
        if (radioButton40.Checked==true) fonk=8; else fonk=9;

        textBox185.Text = Convert.ToString(mikro_komut) + "," +
        Convert.ToString(i_tab) + "," +
        Convert.ToString(fonk) + "," +

        textBox45.Text + "," +
        textBox170.Text + "," +
        textBox44.Text + "," +

        textBox43.Text + "," +

        textBox42.Text + "," +
        textBox38.Text + "," +
        textBox41.Text + "," +
        textBox37.Text + "," +
        textBox35.Text + "," +
        textBox40.Text + "," +
        textBox36.Text + "," +
        textBox39.Text + ",";
    }
    textBox187.Text = "F" + Convert.ToString(i_tab);
    mikro_komut = 1;
    i_tab++;
}

void tab5_parametre_gonder()
{
    Int16 fonk;

    if (!radioButton48.Checked)
    {
        i_tab = 5;
    }
}

```

```

        mikro_komut--; /*1- yazmaya devam*2- k-1de gelen data
hatalı,değiştir* 3-akım baslat* 4-akım durdur*/
        if (mikro_komut < 1) mikro_komut = 1;
        mikro_komut *= 15000;

        if(radioButton57.Checked)fonk=1;else
        if(radioButton56.Checked==true)fonk=2;else
        if(radioButton55.Checked==true)fonk=3;else
        if(radioButton54.Checked==true)fonk=4;else
        if(radioButton53.Checked==true)fonk=5;else
        if(radioButton52.Checked==true)fonk=6;else
        if(radioButton51.Checked==true)fonk=7;else
        if(radioButton50.Checked==true)fonk=8;else fonk=9;

        textBox185.Text = Convert.ToString(mikro_komut) + "," +
        Convert.ToString(i_tab) + "," +
        Convert.ToString(fonk) + "," +

        textBox56.Text + "," +
        textBox171.Text + "," +
        textBox55.Text + "," +

        textBox54.Text + "," +

        textBox53.Text + "," +
        textBox49.Text + "," +
        textBox52.Text + "," +
        textBox48.Text + "," +
        textBox46.Text + "," +
        textBox51.Text + "," +
        textBox47.Text + "," +
        textBox50.Text + ",";

        Byte[] data =
System.Text.Encoding.ASCII.GetBytes(textBox185.Text);

        stream.Write(data, 0, data.Length);

        mikro_komut /= 15000;

        if(radioButton57.Checked)fonk=1;else
        if(radioButton56.Checked==true)fonk=2;else
        if(radioButton55.Checked==true)fonk=3;else
        if(radioButton54.Checked==true)fonk=4;else
        if(radioButton53.Checked==true)fonk=5;else
        if(radioButton52.Checked==true)fonk=6;else
        if(radioButton51.Checked==true)fonk=7;else
        if(radioButton50.Checked==true)fonk=8;else fonk=9;

        textBox185.Text = Convert.ToString(mikro_komut) + "," +
        Convert.ToString(i_tab) + "," +
        Convert.ToString(fonk) + "," +

        textBox56.Text + "," +
        textBox171.Text + "," +
        textBox55.Text + "," +

        textBox54.Text + "," +

```

```

        textBox53.Text + "," +
        textBox49.Text + "," +
        textBox52.Text + "," +
        textBox48.Text + "," +
        textBox46.Text + "," +
        textBox51.Text + "," +
        textBox47.Text + "," +
        textBox50.Text + ",";
    }
    textBox187.Text = "F" + Convert.ToString(i_tab);
    mikro_komut = 1;
    i_tab++;
}

void tab6_parametre_gonder()
{
    Int16 fonk;

    if (!radioButton58.Checked)
    {
        i_tab = 6;

        mikro_komut--; /*1- yazmaya devam*2- k-1de gelen data
hatalı,değiştir* 3-akım baslat* 4-akım durdur*/
        if (mikro_komut < 1) mikro_komut = 1;
        mikro_komut *= 15000;

        if(radioButton67.Checked)fonk=1;else
        if(radioButton66.Checked==true)fonk=2;else
        if(radioButton65.Checked==true)fonk=3;else
        if(radioButton64.Checked==true)fonk=4;else
        if(radioButton63.Checked==true)fonk=5;else
        if(radioButton62.Checked==true)fonk=6;else
        if(radioButton61.Checked==true)fonk=7;else
        if(radioButton60.Checked==true)fonk=8;else fonk=9;

        textBox185.Text = Convert.ToString(mikro_komut) + "," +
            Convert.ToString(i_tab) + "," +
            Convert.ToString(fonk) + "," +

            textBox67.Text + "," +
            textBox172.Text + "," +
            textBox66.Text + "," +

            textBox65.Text + "," +

            textBox64.Text + "," +
            textBox60.Text + "," +
            textBox63.Text + "," +
            textBox59.Text + "," +
            textBox57.Text + "," +
            textBox62.Text + "," +
            textBox58.Text + "," +
            textBox61.Text + ",";

        Byte[] data =
        System.Text.Encoding.ASCII.GetBytes(textBox185.Text);
    }
}

```

```

stream.Write(data, 0, data.Length);

mikro_komut /= 15000;

if(radioButton67.Checked)fonk=1;else
if(radioButton66.Checked==true)fonk=2;else
if(radioButton65.Checked==true)fonk=3;else
if(radioButton64.Checked==true)fonk=4;else
if(radioButton63.Checked==true)fonk=5;else
if(radioButton62.Checked==true)fonk=6;else
if(radioButton61.Checked==true)fonk=7;else
if(radioButton60.Checked==true)fonk=8;else fonk=9;

textBox185.Text = Convert.ToString(mikro_komut) + "," +
Convert.ToString(i_tab) + "," +
Convert.ToString(fonk) + "," +

        textBox67.Text + "," +
        textBox172.Text + "," +
        textBox66.Text + "," +

        textBox65.Text + "," +

        textBox64.Text + "," +
        textBox60.Text + "," +
        textBox63.Text + "," +
        textBox59.Text + "," +
        textBox57.Text + "," +
        textBox62.Text + "," +
        textBox58.Text + "," +
        textBox61.Text + ",";
    }
    textBox187.Text = "F" + Convert.ToString(i_tab);
    mikro_komut = 1;
    i_tab++;
}

void tab7_parametre_gonder()
{
    Int16 fonk;

    if (!radioButton68.Checked)
    {
        i_tab = 7;

        mikro_komut--; /*1- yazmaya devam*2- k-1de gelen data
        hatalı,değiştir* 3-akım baslat* 4-akım durdur*/
        if (mikro_komut < 1) mikro_komut = 1;
        mikro_komut *= 15000;

        if(radioButton77.Checked)fonk=1;else
        if(radioButton76.Checked==true)fonk=2;else
        if(radioButton75.Checked==true)fonk=3;else
        if(radioButton74.Checked==true)fonk=4;else
        if(radioButton73.Checked==true)fonk=5;else
        if(radioButton72.Checked==true)fonk=6;else
        if(radioButton71.Checked==true)fonk=7;else

```

```

if(radioButton70.Checked==true)fonk=8;else fonk=9;

textBox185.Text = Convert.ToString(mikro_komut) + "," +
    Convert.ToString(i_tab) + "," +
    Convert.ToString(fonk) + "," +

    textBox78.Text + "," +
    textBox173.Text + "," +
    textBox77.Text + "," +

    textBox76.Text + "," +

    textBox75.Text + "," +
    textBox71.Text + "," +
    textBox74.Text + "," +
    textBox70.Text + "," +
    textBox68.Text + "," +
    textBox73.Text + "," +
    textBox69.Text + "," +
    textBox72.Text + ",";

    Byte[] data =
System.Text.Encoding.ASCII.GetBytes(textBox185.Text);

stream.Write(data, 0, data.Length);

mikro_komut /= 15000;

if(radioButton77.Checked)fonk=1;else
if(radioButton76.Checked==true)fonk=2;else
if(radioButton75.Checked==true)fonk=3;else
if(radioButton74.Checked==true)fonk=4;else
if(radioButton73.Checked==true)fonk=5;else
if(radioButton72.Checked==true)fonk=6;else
if(radioButton71.Checked==true)fonk=7;else
if(radioButton70.Checked==true)fonk=8;else fonk=9;

textBox185.Text = Convert.ToString(mikro_komut) + "," +
    Convert.ToString(i_tab) + "," +
    Convert.ToString(fonk) + "," +

    textBox78.Text + "," +
    textBox173.Text + "," +
    textBox77.Text + "," +

    textBox76.Text + "," +

    textBox75.Text + "," +
    textBox71.Text + "," +
    textBox74.Text + "," +
    textBox70.Text + "," +
    textBox68.Text + "," +
    textBox73.Text + "," +
    textBox69.Text + "," +
    textBox72.Text + ",";
}
textBox187.Text = "F" + Convert.ToString(i_tab);
mikro_komut = 1;

```

```

        i_tab++;
    }

    void tab8_parametre_gonder()
    {
        Int16 fonk;

        if (!radioButton78.Checked)
        {
            i_tab = 8;

            mikro_komut--; /*1- yazmaya devam*2- k-1de gelen data
hatalı,değiştir* 3-akım baslat* 4-akım durdur*/
            if (mikro_komut < 1) mikro_komut = 1;
            mikro_komut *= 15000;

            if(radioButton87.Checked)fonk=1;else
            if(radioButton86.Checked==true)fonk=2;else
            if(radioButton85.Checked==true)fonk=3;else
            if(radioButton84.Checked==true)fonk=4;else
            if(radioButton83.Checked==true)fonk=5;else
            if(radioButton82.Checked==true)fonk=6;else
            if(radioButton81.Checked==true)fonk=7;else
            if(radioButton80.Checked==true)fonk=8;else fonk=9;

            textBox185.Text = Convert.ToString(mikro_komut) + "," +
                Convert.ToString(i_tab) + "," +
                Convert.ToString(fonk) + "," +

                textBox89.Text + "," +
                textBox174.Text + "," +
                textBox88.Text + "," +

                textBox87.Text + "," +

                textBox86.Text + "," +
                textBox82.Text + "," +
                textBox85.Text + "," +
                textBox81.Text + "," +
                textBox79.Text + "," +
                textBox84.Text + "," +
                textBox80.Text + "," +
                textBox83.Text + ",";

            Byte[] data =
            System.Text.Encoding.ASCII.GetBytes(textBox185.Text);

            stream.Write(data, 0, data.Length);

            mikro_komut /= 15000;

            if(radioButton87.Checked)fonk=1;else
            if(radioButton86.Checked==true)fonk=2;else
            if(radioButton85.Checked==true)fonk=3;else
            if(radioButton84.Checked==true)fonk=4;else
            if(radioButton83.Checked==true)fonk=5;else
            if(radioButton82.Checked==true)fonk=6;else
            if(radioButton81.Checked==true)fonk=7;else

```

```

if(radioButton80.Checked==true)fonk=8;else fonk=9;

textBox185.Text = Convert.ToString(mikro_komut) + "," +
    Convert.ToString(i_tab) + "," +
    Convert.ToString(fonk) + "," +

    textBox89.Text + "," +
    textBox174.Text + "," +
    textBox88.Text + "," +

    textBox87.Text + "," +

    textBox86.Text + "," +
    textBox82.Text + "," +
    textBox85.Text + "," +
    textBox81.Text + "," +
    textBox79.Text + "," +
    textBox84.Text + "," +
    textBox80.Text + "," +
    textBox83.Text + ",";
}
textBox187.Text = "F" + Convert.ToString(i_tab);
mikro_komut = 1;
i_tab++;
}

void tab9_parametre_gonder()
{
    Int16 fonk;

    if (!radioButton88.Checked)
    {
        i_tab = 9;

        mikro_komut--; /*1- yazmaya devam*2- k-1de gelen data
        hatalı,değiştir* 3-akım baslat* 4-akım durdur*/
        if (mikro_komut < 1) mikro_komut = 1;
        mikro_komut *= 15000;

        if(radioButton97.Checked)fonk=1;else
        if(radioButton96.Checked==true)fonk=2;else
        if(radioButton95.Checked==true)fonk=3;else
        if(radioButton94.Checked==true)fonk=4;else
        if(radioButton93.Checked==true)fonk=5;else
        if(radioButton92.Checked==true)fonk=6;else
        if(radioButton91.Checked==true)fonk=7;else
        if(radioButton90.Checked==true)fonk=8;else fonk=9;

        textBox185.Text = Convert.ToString(mikro_komut) + "," +
            Convert.ToString(i_tab) + "," +
            Convert.ToString(fonk) + "," +

            textBox100.Text + "," +
            textBox175.Text + "," +
            textBox99.Text + "," +

            textBox98.Text + "," +

```

```

        textBox97.Text + "," +
        textBox93.Text + "," +
        textBox96.Text + "," +
        textBox92.Text + "," +
        textBox90.Text + "," +
        textBox95.Text + "," +
        textBox91.Text + "," +
        textBox94.Text + ",";

        Byte[] data =
System.Text.Encoding.ASCII.GetBytes(textBox185.Text);

        stream.Write(data, 0, data.Length);

        mikro_komut /= 15000;

        if (radioButton97.Checked) fonk=1; else
        if (radioButton96.Checked==true) fonk=2; else
        if (radioButton95.Checked==true) fonk=3; else
        if (radioButton94.Checked==true) fonk=4; else
        if (radioButton93.Checked==true) fonk=5; else
        if (radioButton92.Checked==true) fonk=6; else
        if (radioButton91.Checked==true) fonk=7; else
        if (radioButton90.Checked==true) fonk=8; else fonk=9;

        textBox185.Text = Convert.ToString(mikro_komut) + "," +
        Convert.ToString(i_tab) + "," +
        Convert.ToString(fonk) + "," +

        textBox100.Text + "," +
        textBox175.Text + "," +
        textBox99.Text + "," +

        textBox98.Text + "," +

        textBox97.Text + "," +
        textBox93.Text + "," +
        textBox96.Text + "," +
        textBox92.Text + "," +
        textBox90.Text + "," +
        textBox95.Text + "," +
        textBox91.Text + "," +
        textBox94.Text + ",";
    }
    textBox187.Text = "F" + Convert.ToString(i_tab);
    mikro_komut = 1;
    i_tab++;
}

void tab10_parametre_gonder()
{
    Int16 fonk;

    if (!radioButton98.Checked)
    {
        i_tab = 10;
    }
}

```

```

        mikro_komut--; /*1- yazmaya devam*2- k-1de gelen data
        hatalı,değiştir* 3-akım baslat* 4-akım durdur*/
        if (mikro_komut < 1) mikro_komut = 1;
        mikro_komut *= 15000;

        if(radioButton107.Checked)fonk=1;else
        if(radioButton106.Checked==true)fonk=2;else
        if(radioButton105.Checked==true)fonk=3;else
        if(radioButton104.Checked==true)fonk=4;else
        if(radioButton103.Checked==true)fonk=5;else
        if(radioButton102.Checked==true)fonk=6;else
        if(radioButton101.Checked==true)fonk=7;else
        if(radioButton100.Checked==true)fonk=8;else fonk=9;

        textBox185.Text = Convert.ToString(mikro_komut) + "," +
        Convert.ToString(i_tab) + "," +
        Convert.ToString(fonk) + "," +

        textBox111.Text + "," +
        textBox176.Text + "," +
        textBox110.Text + "," +

        textBox109.Text + "," +

        textBox108.Text + "," +
        textBox104.Text + "," +
        textBox107.Text + "," +
        textBox103.Text + "," +
        textBox101.Text + "," +
        textBox106.Text + "," +
        textBox102.Text + "," +
        textBox105.Text + ",";

```

```

        Byte[] data =
        System.Text.Encoding.ASCII.GetBytes(textBox185.Text);

        stream.Write(data, 0, data.Length);

        mikro_komut /= 15000;

        if(radioButton107.Checked)fonk=1;else
        if(radioButton106.Checked==true)fonk=2;else
        if(radioButton105.Checked==true)fonk=3;else
        if(radioButton104.Checked==true)fonk=4;else
        if(radioButton103.Checked==true)fonk=5;else
        if(radioButton102.Checked==true)fonk=6;else
        if(radioButton101.Checked==true)fonk=7;else
        if(radioButton100.Checked==true)fonk=8;else fonk=9;

        textBox185.Text = Convert.ToString(mikro_komut) + "," +
        Convert.ToString(i_tab) + "," +
        Convert.ToString(fonk) + "," +

        textBox111.Text + "," +
        textBox176.Text + "," +
        textBox110.Text + "," +

        textBox109.Text + "," +

```

```

        textBox108.Text + "," +
        textBox104.Text + "," +
        textBox107.Text + "," +
        textBox103.Text + "," +
        textBox101.Text + "," +
        textBox106.Text + "," +
        textBox102.Text + "," +
        textBox105.Text + ",";
    }
    textBox187.Text = "F" + Convert.ToString(i_tab);
    mikro_komut = 1;
    i_tab++;
}

void tab11_parametre_gonder()
{
    Int16 fonk;

    if (!radioButton108.Checked)
    {
        i_tab = 11;

        mikro_komut--; /*1- yazmaya devam*2- k-1de gelen data
        hatalı,değiştir* 3-akım baslat* 4-akım durdur*/
        if (mikro_komut < 1) mikro_komut = 1;
        mikro_komut *= 15000;

        if(radioButton117.Checked)fonk=1;else
        if(radioButton116.Checked==true)fonk=2;else
        if(radioButton115.Checked==true)fonk=3;else
        if(radioButton114.Checked==true)fonk=4;else
        if(radioButton113.Checked==true)fonk=5;else
        if(radioButton112.Checked==true)fonk=6;else
        if(radioButton111.Checked==true)fonk=7;else
        if(radioButton110.Checked==true)fonk=8;else fonk=9;

        textBox185.Text = Convert.ToString(mikro_komut) + "," +
            Convert.ToString(i_tab) + "," +
            Convert.ToString(fonk) + "," +

            textBox122.Text + "," +
            textBox177.Text + "," +
            textBox121.Text + "," +

            textBox120.Text + "," +

            textBox119.Text + "," +
            textBox115.Text + "," +
            textBox118.Text + "," +
            textBox114.Text + "," +
            textBox112.Text + "," +
            textBox117.Text + "," +
            textBox113.Text + "," +
            textBox116.Text + ",";

        Byte[] data =
        System.Text.Encoding.ASCII.GetBytes(textBox185.Text);
    }
}

```

```

stream.Write(data, 0, data.Length);

mikro_komut /= 15000;

if(radioButton117.Checked)fonk=1;else
if(radioButton116.Checked==true)fonk=2;else
if(radioButton115.Checked==true)fonk=3;else
if(radioButton114.Checked==true)fonk=4;else
if(radioButton113.Checked==true)fonk=5;else
if(radioButton112.Checked==true)fonk=6;else
if(radioButton111.Checked==true)fonk=7;else
if(radioButton110.Checked==true)fonk=8;else fonk=9;

textBox185.Text = Convert.ToString(mikro_komut) + "," +
Convert.ToString(i_tab) + "," +
Convert.ToString(fonk) + "," +

        textBox122.Text + "," +
        textBox177.Text + "," +
        textBox121.Text + "," +

        textBox120.Text + "," +

        textBox119.Text + "," +
        textBox115.Text + "," +
        textBox118.Text + "," +
        textBox114.Text + "," +
        textBox112.Text + "," +
        textBox117.Text + "," +
        textBox113.Text + "," +
        textBox116.Text + ",";
    }
    textBox187.Text = "F" + Convert.ToString(i_tab);
    mikro_komut = 1;
    i_tab++;
}

void tab12_parametre_gonder()
{
    int16 fonk;

    if (!radioButton118.Checked)
    {
        i_tab = 12;

        mikro_komut--; /*1- yazmaya devam*2- k-1de gelen data
        hatalı,değiştir* 3-akım baslat* 4-akım durdur*/
        if (mikro_komut < 1) mikro_komut = 1;
        mikro_komut *= 15000;

        if(radioButton127.Checked)fonk=1;else
        if(radioButton126.Checked==true)fonk=2;else
        if(radioButton125.Checked==true)fonk=3;else
        if(radioButton124.Checked==true)fonk=4;else
        if(radioButton123.Checked==true)fonk=5;else
        if(radioButton122.Checked==true)fonk=6;else
        if(radioButton121.Checked==true)fonk=7;else

```

```

if(radioButton120.Checked==true)fonk=8;else fonk=9;

textBox185.Text = Convert.ToString(mikro_komut) + "," +
    Convert.ToString(i_tab) + "," +
    Convert.ToString(fonk) + "," +

    textBox133.Text + "," +
    textBox178.Text + "," +
    textBox132.Text + "," +

    textBox131.Text + "," +

    textBox130.Text + "," +
    textBox126.Text + "," +
    textBox129.Text + "," +
    textBox125.Text + "," +
    textBox123.Text + "," +
    textBox128.Text + "," +
    textBox124.Text + "," +
    textBox127.Text + ",";

    Byte[] data =
System.Text.Encoding.ASCII.GetBytes(textBox185.Text);

stream.Write(data, 0, data.Length);

mikro_komut /= 15000;

if(radioButton127.Checked)fonk=1;else
if(radioButton126.Checked==true)fonk=2;else
if(radioButton125.Checked==true)fonk=3;else
if(radioButton124.Checked==true)fonk=4;else
if(radioButton123.Checked==true)fonk=5;else
if(radioButton122.Checked==true)fonk=6;else
if(radioButton121.Checked==true)fonk=7;else
if(radioButton120.Checked==true)fonk=8;else fonk=9;

textBox185.Text = Convert.ToString(mikro_komut) + "," +
    Convert.ToString(i_tab) + "," +
    Convert.ToString(fonk) + "," +

    textBox133.Text + "," +
    textBox178.Text + "," +
    textBox132.Text + "," +

    textBox131.Text + "," +

    textBox130.Text + "," +
    textBox126.Text + "," +
    textBox129.Text + "," +
    textBox125.Text + "," +
    textBox123.Text + "," +
    textBox128.Text + "," +
    textBox124.Text + "," +
    textBox127.Text + ",";
}
textBox187.Text = "F" + Convert.ToString(i_tab);
mikro_komut = 1;

```

```

        i_tab++;
    }

    void tab13_parametre_gonder()
    {
        Int16 fonk;

        if (!radioButton128.Checked)
        {
            i_tab = 13;

            mikro_komut--; /*1- yazmaya devam*2- k-1de gelen data
hatalı,değiştir* 3-akım baslat* 4-akım durdur*/
            if (mikro_komut < 1) mikro_komut = 1;
            mikro_komut *= 15000;

            if(radioButton137.Checked)fonk=1;else
            if(radioButton136.Checked==true)fonk=2;else
            if(radioButton135.Checked==true)fonk=3;else
            if(radioButton134.Checked==true)fonk=4;else
            if(radioButton133.Checked==true)fonk=5;else
            if(radioButton132.Checked==true)fonk=6;else
            if(radioButton131.Checked==true)fonk=7;else
            if(radioButton130.Checked==true)fonk=8;else fonk=9;

            textBox185.Text = Convert.ToString(mikro_komut) + "," +
                Convert.ToString(i_tab) + "," +
                Convert.ToString(fonk) + "," +

                textBox144.Text + "," +
                textBox179.Text + "," +
                textBox143.Text + "," +

                textBox142.Text + "," +

                textBox141.Text + "," +
                textBox137.Text + "," +
                textBox140.Text + "," +
                textBox136.Text + "," +
                textBox134.Text + "," +
                textBox139.Text + "," +
                textBox135.Text + "," +
                textBox138.Text + ",";

            Byte[] data =
System.Text.Encoding.ASCII.GetBytes(textBox185.Text);

            stream.Write(data, 0, data.Length);

            mikro_komut /= 15000;

            if(radioButton137.Checked)fonk=1;else
            if(radioButton136.Checked==true)fonk=2;else
            if(radioButton135.Checked==true)fonk=3;else
            if(radioButton134.Checked==true)fonk=4;else
            if(radioButton133.Checked==true)fonk=5;else
            if(radioButton132.Checked==true)fonk=6;else
            if(radioButton131.Checked==true)fonk=7;else

```

```

if(radioButton130.Checked==true)fonk=8;else fonk=9;

textBox185.Text = Convert.ToString(mikro_komut) + "," +
    Convert.ToString(i_tab) + "," +
    Convert.ToString(fonk) + "," +

    textBox144.Text + "," +
    textBox179.Text + "," +
    textBox143.Text + "," +

    textBox142.Text + "," +

    textBox141.Text + "," +
    textBox137.Text + "," +
    textBox140.Text + "," +
    textBox136.Text + "," +
    textBox134.Text + "," +
    textBox139.Text + "," +
    textBox135.Text + "," +
    textBox138.Text + ",";
}
textBox187.Text = "F" + Convert.ToString(i_tab);
mikro_komut = 1;
i_tab++;
}

void tab14_parametre_gonder()
{
    int16 fonk;

    if (!radioButton138.Checked)
    {
        i_tab = 14;

        mikro_komut--; /*1- yazmaya devam*2- k-1de gelen data
        hatalı,değiştir* 3-akım baslat* 4-akım durdur*/
        if (mikro_komut < 1) mikro_komut = 1;
        mikro_komut *= 15000;

        if(radioButton147.Checked)fonk=1;else
        if(radioButton146.Checked==true)fonk=2;else
        if(radioButton145.Checked==true)fonk=3;else
        if(radioButton144.Checked==true)fonk=4;else
        if(radioButton143.Checked==true)fonk=5;else
        if(radioButton142.Checked==true)fonk=6;else
        if(radioButton141.Checked==true)fonk=7;else
        if(radioButton140.Checked==true)fonk=8;else fonk=9;

        textBox185.Text = Convert.ToString(mikro_komut) + "," +
            Convert.ToString(i_tab) + "," +
            Convert.ToString(fonk) + "," +

            textBox155.Text + "," +
            textBox180.Text + "," +
            textBox154.Text + "," +

            textBox153.Text + "," +

```

```

        textBox152.Text + "," +
        textBox148.Text + "," +
        textBox151.Text + "," +
        textBox147.Text + "," +
        textBox145.Text + "," +
        textBox150.Text + "," +
        textBox146.Text + "," +
        textBox149.Text + ",";

        Byte[] data =
System.Text.Encoding.ASCII.GetBytes(textBox185.Text);

        stream.Write(data, 0, data.Length);

        mikro_komut /= 15000;

        if(radioButton147.Checked)fonk=1;else
        if(radioButton146.Checked==true)fonk=2;else
        if(radioButton145.Checked==true)fonk=3;else
        if(radioButton144.Checked==true)fonk=4;else
        if(radioButton143.Checked==true)fonk=5;else
        if(radioButton142.Checked==true)fonk=6;else
        if(radioButton141.Checked==true)fonk=7;else
        if(radioButton140.Checked==true)fonk=8;else fonk=9;

        textBox185.Text = Convert.ToString(mikro_komut) + "," +
        Convert.ToString(i_tab) + "," +
        Convert.ToString(fonk) + "," +

        textBox155.Text + "," +
        textBox180.Text + "," +
        textBox154.Text + "," +

        textBox153.Text + "," +

        textBox152.Text + "," +
        textBox148.Text + "," +
        textBox151.Text + "," +
        textBox147.Text + "," +
        textBox145.Text + "," +
        textBox150.Text + "," +
        textBox146.Text + "," +
        textBox149.Text + ",";
    }
    textBox187.Text = "F" + Convert.ToString(i_tab);
    mikro_komut = 1;
    i_tab++;
}

void tab15_parametre_gonder()
{
    Int16 fonk;

    if (!radioButton148.Checked)
    {
        i_tab = 15;
    }
}

```

```

        mikro_komut--; /*1- yazmaya devam*2- k-1de gelen data
hatalı,değiştir* 3-akım baslat* 4-akım durdur*/
        if (mikro_komut < 1) mikro_komut = 1;
        mikro_komut *= 15000;

        if(radioButton157.Checked)fonk=1;else
        if(radioButton156.Checked==true)fonk=2;else
        if(radioButton155.Checked==true)fonk=3;else
        if(radioButton154.Checked==true)fonk=4;else
        if(radioButton153.Checked==true)fonk=5;else
        if(radioButton152.Checked==true)fonk=6;else
        if(radioButton151.Checked==true)fonk=7;else
        if(radioButton150.Checked==true)fonk=8;else fonk=9;

        textBox185.Text = Convert.ToString(mikro_komut) + ","
+
        Convert.ToString(i_tab) + "," +
        Convert.ToString(fonk) + "," +
        textBox166.Text + "," +
        textBox181.Text + "," +
        textBox165.Text + "," +
        textBox164.Text + "," +
        textBox163.Text + "," +
        textBox159.Text + "," +
        textBox162.Text + "," +
        textBox158.Text + "," +
        textBox156.Text + "," +
        textBox161.Text + "," +
        textBox157.Text + "," +
        textBox160.Text + ",";

        Byte[] data =
System.Text.Encoding.ASCII.GetBytes(textBox185.Text);

        stream.Write(data, 0, data.Length);

        mikro_komut /= 15000;

        if(radioButton157.Checked)fonk=1;else
        if(radioButton156.Checked==true)fonk=2;else
        if(radioButton155.Checked==true)fonk=3;else
        if(radioButton154.Checked==true)fonk=4;else
        if(radioButton153.Checked==true)fonk=5;else
        if(radioButton152.Checked==true)fonk=6;else
        if(radioButton151.Checked==true)fonk=7;else
        if(radioButton150.Checked==true)fonk=8;else fonk=9;

        textBox185.Text = Convert.ToString(mikro_komut) + ","
+
        Convert.ToString(i_tab) + "," +
        Convert.ToString(fonk) + "," +
        textBox166.Text + "," +

```

```

        textBox181.Text + "," +
        textBox165.Text + "," +

        textBox164.Text + "," +

        textBox163.Text + "," +
        textBox159.Text + "," +
        textBox162.Text + "," +
        textBox158.Text + "," +
        textBox156.Text + "," +
        textBox161.Text + "," +
        textBox157.Text + "," +
        textBox160.Text + ",";
    }
    textBox187.Text = "F" + Convert.ToString(i_tab);
    mikro_komut = 1;
    i_tab++;
}

private void timer1_Tick(object sender, EventArgs e)
{
    Console.Beep();

    if (i_tab == 1) tab1_parametre_gonder();else if
    (i_tab == 2) tab2_parametre_gonder();else if
    (i_tab == 3) tab3_parametre_gonder();else if
    (i_tab == 4) tab4_parametre_gonder();else if
    (i_tab == 5) tab5_parametre_gonder();else if
    (i_tab == 6) tab6_parametre_gonder();else if
    (i_tab == 7) tab7_parametre_gonder();else if
    (i_tab == 8) tab8_parametre_gonder();else if
    (i_tab == 9) tab9_parametre_gonder();else if
    (i_tab == 10) tab10_parametre_gonder();else if
    (i_tab == 11) tab11_parametre_gonder();else if
    (i_tab == 12) tab12_parametre_gonder();else if
    (i_tab == 13) tab13_parametre_gonder();else if
    (i_tab == 14) tab14_parametre_gonder();else if
    (i_tab == 15) tab15_parametre_gonder();else
    {
        timer1.Enabled = false;
        textBox187.Text = "";

        if ((!radioButton11.Checked) || (!radioButton4.Checked) ||
        (!radioButton28.Checked) ||

        (!radioButton38.Checked) ||

        (!radioButton48.Checked) ||

        (!radioButton58.Checked) ||

        (!radioButton68.Checked) ||

        (!radioButton78.Checked) ||

        (!radioButton88.Checked) ||

        (!radioButton98.Checked) ||

```

```

(!radioButton108.Checked) ||
(!radioButton118.Checked) ||
(!radioButton128.Checked) ||
(!radioButton138.Checked) ||
(!radioButton148.Checked))
    {
        mikro_komut = 3; /* 1 yazmaya devam/2 (k-1) data
hatalı,değiştir/3 akım baslat/4 akım durdur */
        //Byte[] data =
System.Text.Encoding.ASCII.GetBytes(Convert.ToString(40000)+","); //işlem başlat
        //stream.Write(data, 0, data.Length);
        timer2.Enabled = true;
    }
}
}

private void backgroundWorker1_ProgressChanged(object sender,
ProgressChangedEventArgs e)
{
    timer1.Enabled = false;

    textBox184.Text= e.UserState.ToString();

    /* 1 yazmaya devam/2 (k-1) data hatalı,değiştir/3 akım baslat/4 akım
durdur*/
    if (mikro_komut == 1)
    {
        if ((textBox185.Text.Length < textBox184.Text.Length))
            textBox184.Text = textBox184.Text.Substring(0,
textBox185.Text.Length);

        if (!textBox185.Text.Equals(textBox184.Text))
        {
            i_tab--;
            mikro_komut = 3; //tab içinde -1 yapılarak üstüne yaz komutu
olacak

            Console.Beep(650, 250);
        }

        timer1.Enabled = true;
    }
else if (mikro_komut == 3)
{
    if (textBox184.Text.Equals("40000" + "\r\n"))
    {
        button2.Enabled = false;
        button4.Enabled = true;

        this.BackColor = Color.LightGreen;
        timer2.Enabled = false;
    }
}
}

```

```

        timer4.Enabled = true;
        richTextBox1.Enabled = false;

        mikro_komut = 6;
    }
}
else if (mikro_komut == 4)
{
    if (textBox184.Text.Equals("50000" + "\r\n"))
    {
        button2.Enabled = true;
        button4.Enabled = false;

        this.BackColor = Color.Red;
        timer2.Enabled = false;

        timer4.Enabled = false;
        richTextBox1.Enabled = true;
    }
}
else
{
    if (textBox184.Text.Equals("50001" + "\r\n"))
    {
        timer3.Enabled = false;
        this.BackColor = Color.Magenta;
        Console.Beep(650, 250);

        label234.Visible = true;
        label235.Visible = true;
        panel1.Visible = true;

        timer3.Enabled = true;

        timer4.Enabled = false;
        richTextBox1.Enabled = true;
    }
}
}

private void timer2_Tick(object sender, EventArgs e)
{
    Console.Beep();

    if (mikro_komut == 3)
    {
        Byte[] data =
System.Text.Encoding.ASCII.GetBytes(Convert.ToString(40000) + ",");//işlem başlat
        stream.Write(data, 0, data.Length);
    }
    else if (mikro_komut == 4)
    {
        Byte[] data =
System.Text.Encoding.ASCII.GetBytes(Convert.ToString(50000) + ",");//işlem durdur
        stream.Write(data, 0, data.Length);
    }
}

```

```

    }

    void backgroundWorker1_DoWork(object sender, DoWorkEventArgs e)
    {
        /*Listen for incoming messages indefinitely until connection is
broke*/
        while (myClient.Connected)
        {
            if (!stream.DataAvailable)
            {
                System.Threading.Thread.Sleep(250);
                continue;
            }

            Byte[] data = new Byte[256];

            String responseData = String.Empty;

            Int32 bytes = stream.Read(data, 0, data.Length);
            responseData = System.Text.Encoding.ASCII.GetString(data, 0,
bytes);
            backgroundWorker1.ReportProgress(0, responseData);
        }
    }

    private void backgroundWorker1_RunWorkerCompleted(object sender,
RunWorkerCompletedEventArgs e)
    {
        //textBox184.Text ="Background worker stopped";
    }

    private void button6_Click(object sender, EventArgs e)
    {
        String ip = "10.10.100.254";
        String port = "8899";

        //Shell("cmd.exe /k " + "netsh wlan connect name=AğAdı");

        /*Parse IP & port info and connect to client, the stream has global
scope so that other funcs can use
Note the usage of BGW is just for realtime display of received
messages*/

        if (textBox183.BackColor == Color.Red)
        {
            try
            {
                myClient = new TcpClient(ip, int.Parse(port));

                stream = myClient.GetStream();
                backgroundWorker1.RunWorkerAsync();

                button4.Enabled = true;
                button2.Enabled = false;

                textBox183.BackColor = Color.GreenYellow;
            }
            catch

```

```

        {
            MessageBox.Show("Connect to MultifunctionCurrentSource WIFI",
"WARNING",
                                MessageBoxButtons.OK,
                                MessageBoxIcon.Warning);
        }

        Console.Beep();
    }

    private void Form1_Leave(object sender, EventArgs e)
    {
        stream.Close();
        myClient.Close();
    }

    private void button4_Click(object sender, EventArgs e)
    {
        timer3.Enabled = false;
        mikro_komut = 4;

        panel1.Visible = false;
        label234.Visible = false;
        label235.Visible = false;

        timer2.Enabled = true;
        Console.Beep();
    }

    private void timer3_Tick(object sender, EventArgs e)
    {
        timer3.Enabled = false;

        label234.Visible = false;
        label235.Visible = false;
        panel1.Visible = false;

        this.BackColor = Color.LightGreen;

        timer4.Enabled = true;
        richTextBox1.Enabled = false;
    }

    public static bool IsAllLetters(string s)
    {
        foreach (char c in s)
        {
            if (!Char.IsLetter(c))
                return false;
        }
        return true;
    }

```

```

private void timer4_Tick(object sender, EventArgs e)
{
    Int32 sayi;
    Int32 sayi2;

    if (!IsAllLetters(richTextBox1.Text))
    {
        sayi = Convert.ToInt32(richTextBox1.Text);

        if (sayi < 1) { sayi = 0; }
        else
        {
            sayi=sayi-1;
            richTextBox1.Text = Convert.ToString(sayi);
        }

        if (sayi == 0)
        {
            timer3.Enabled = false;
            mikro_komut = 4;

            panel1.Visible = false;
            label234.Visible = false;
            label235.Visible = false;

            timer2.Enabled = true;
            Console.Beep();
        }
    }

    if (textBox182.Text != "-")
    {
        sayi = Convert.ToInt32(textBox182.Text);
        sayi *= 60;//60sn de olulşan Q, mA * dk
        sayi2 = Convert.ToInt32(textBox186.Text);
        textBox186.Text =Convert.ToString(sayi/1000+sayi2);
    }
}

private void Form1_Shown(object sender, EventArgs e)
{
    MessageBox.Show("<delta I> and <delta T> should satisfy ranges of:\n"
+
        "\n" +
        "For Step Function:\n" + "10mA =< dI =<
10000mA; 1ms=< dT =< 10000ms\n" +
        "\n" +
        "For SinCos Functions:\n" + "50mA =< dI =<
10000mA; 20ms=< dT =< 10000ms\n" +
        "\n" +
        "For Ramp Functions:\n" + "50mA =< dI =<
10000mA; 5ms=< dT =< 10000ms\n" +

```

```

        "\n" +
        "For 1. and 2. Order System Functions:\n" +
"50mA =< dI =< 10000mA; 10ms=< dT =< 10000ms\n",

        "WARNING",
        MessageBoxButtons.OK,
MessageBoxIcon.Warning);

        dosya_yolu_basla = Directory.GetCurrentDirectory();
    }

    private void button3_Click(object sender, EventArgs e)
    {
        MessageBox.Show("<delta I> and <delta T> should satisfy ranges of:\n"
+
        "\n" +
        "For Step Function:\n" + "10mA =< dI =<
10000mA; 1ms=< dT =< 10000ms\n" +
        "\n" +
        "For SinCos Functions:\n" + "50mA =< dI =<
10000mA; 20ms=< dT =< 10000ms\n" +
        "\n" +
        "For Ramp Functions:\n" + "50mA =< dI =<
10000mA; 5ms=< dT =< 10000ms\n" +
        "\n" +
        "For 1. and 2. Order System Functions:\n" +
"50mA =< dI =< 10000mA; 10ms=< dT =< 10000ms\n",

        "WARNING",
        MessageBoxButtons.OK, MessageBoxIcon.Warning);
    }

    void f1_parametre_aktar()
    {
        if (radioButton11.Checked) textBox188.Text = "1,"; else
textBox188.Text = "0,";
        if (radioButton1.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
        if (radioButton2.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
        if (radioButton3.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
        if (radioButton5.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
        if (radioButton6.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
        if (radioButton8.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
        if (radioButton10.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
        if (radioButton19.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
        if (radioButton20.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
    }
}

```

```

        textBox188.Text += textBox2.Text + "," ;
        textBox188.Text += textBox167.Text + ",";
        textBox188.Text += textBox3.Text + ",";

        textBox188.Text += textBox12.Text + ",";

        textBox188.Text += textBox4.Text + ",";
        textBox188.Text += textBox5.Text + ",";
        textBox188.Text += textBox6.Text + ",";
        textBox188.Text += textBox7.Text + ",";
        textBox188.Text += textBox8.Text + ",";
        textBox188.Text += textBox9.Text + ",";
        textBox188.Text += textBox10.Text + ",";
        textBox188.Text += textBox11.Text + ",";
    }

    void f2_parametre_aktar()
    {
        if (radioButton4.Checked) textBox188.Text = "1,"; else
        textBox188.Text = "0,";
        if (radioButton27.Checked) textBox188.Text += "1,"; else
        textBox188.Text += "0,";
        if (radioButton26.Checked) textBox188.Text += "1,"; else
        textBox188.Text += "0,";
        if (radioButton25.Checked) textBox188.Text += "1,"; else
        textBox188.Text += "0,";
        if (radioButton24.Checked) textBox188.Text += "1,"; else
        textBox188.Text += "0,";
        if (radioButton23.Checked) textBox188.Text += "1,"; else
        textBox188.Text += "0,";
        if (radioButton22.Checked) textBox188.Text += "1,"; else
        textBox188.Text += "0,";
        if (radioButton21.Checked) textBox188.Text += "1,"; else
        textBox188.Text += "0,";
        if (radioButton9.Checked) textBox188.Text += "1,"; else
        textBox188.Text += "0,";
        if (radioButton7.Checked) textBox188.Text += "1,"; else
        textBox188.Text += "0,";

        textBox188.Text += textBox23.Text + ",";
        textBox188.Text += textBox168.Text + ",";
        textBox188.Text += textBox22.Text + ",";

        textBox188.Text += textBox21.Text + ",";

        textBox188.Text += textBox20.Text + ",";
        textBox188.Text += textBox18.Text + ",";
        textBox188.Text += textBox16.Text + ",";
        textBox188.Text += textBox14.Text + ",";
        textBox188.Text += textBox19.Text + ",";
        textBox188.Text += textBox17.Text + ",";
        textBox188.Text += textBox15.Text + ",";
        textBox188.Text += textBox13.Text + ",";
    }

    void f3_parametre_aktar()
    {

```

```

        if (radioButton28.Checked) textBox188.Text = "1,"; else
textBox188.Text = "0,";
        if (radioButton37.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
        if (radioButton36.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
        if (radioButton35.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
        if (radioButton34.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
        if (radioButton33.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
        if (radioButton32.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
        if (radioButton31.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
        if (radioButton30.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
        if (radioButton29.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";

        textBox188.Text += textBox34.Text + ",";
        textBox188.Text += textBox169.Text + ",";
        textBox188.Text += textBox33.Text + ",";

        textBox188.Text += textBox32.Text + ",";

        textBox188.Text += textBox31.Text + ",";
        textBox188.Text += textBox29.Text + ",";
        textBox188.Text += textBox27.Text + ",";
        textBox188.Text += textBox25.Text + ",";
        textBox188.Text += textBox30.Text + ",";
        textBox188.Text += textBox28.Text + ",";
        textBox188.Text += textBox26.Text + ",";
        textBox188.Text += textBox24.Text + ",";
    }

    void f4_parametre_aktar()
    {
        if (radioButton38.Checked) textBox188.Text = "1,"; else
textBox188.Text = "0,";
        if (radioButton47.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
        if (radioButton46.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
        if (radioButton45.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
        if (radioButton44.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
        if (radioButton43.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
        if (radioButton42.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
        if (radioButton41.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
        if (radioButton40.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
    }

```

```
        if (radioButton39.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
```

```
        textBox188.Text += textBox45.Text + ",";
        textBox188.Text += textBox170.Text + ",";
        textBox188.Text += textBox44.Text + ",";
```

```
        textBox188.Text += textBox43.Text + ",";
```

```
        textBox188.Text += textBox42.Text + ",";
        textBox188.Text += textBox40.Text + ",";
        textBox188.Text += textBox38.Text + ",";
        textBox188.Text += textBox36.Text + ",";
        textBox188.Text += textBox41.Text + ",";
        textBox188.Text += textBox39.Text + ",";
        textBox188.Text += textBox37.Text + ",";
        textBox188.Text += textBox35.Text + ",";
```

```
    }
```

```
void f5_parametre_aktar()
```

```
{
```

```
    if (radioButton48.Checked) textBox188.Text = "1,"; else
textBox188.Text = "0,";
```

```
    if (radioButton57.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
```

```
    if (radioButton56.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
```

```
    if (radioButton55.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
```

```
    if (radioButton54.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
```

```
    if (radioButton53.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
```

```
    if (radioButton52.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
```

```
    if (radioButton51.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
```

```
    if (radioButton50.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
```

```
    if (radioButton49.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
```

```
        textBox188.Text += textBox56.Text + ",";
        textBox188.Text += textBox171.Text + ",";
        textBox188.Text += textBox55.Text + ",";
```

```
        textBox188.Text += textBox54.Text + ",";
```

```
        textBox188.Text += textBox53.Text + ",";
        textBox188.Text += textBox51.Text + ",";
        textBox188.Text += textBox49.Text + ",";
        textBox188.Text += textBox47.Text + ",";
        textBox188.Text += textBox52.Text + ",";
        textBox188.Text += textBox50.Text + ",";
        textBox188.Text += textBox48.Text + ",";
        textBox188.Text += textBox46.Text + ",";
```

```
    }
```

```

void f6_parametre_aktar()
{
    if (radioButton58.Checked) textBox188.Text = "1,"; else
textBox188.Text = "0,";
    if (radioButton67.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
    if (radioButton66.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
    if (radioButton65.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
    if (radioButton64.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
    if (radioButton63.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
    if (radioButton62.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
    if (radioButton61.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
    if (radioButton60.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
    if (radioButton59.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";

    textBox188.Text += textBox67.Text + ",";
    textBox188.Text += textBox172.Text + ",";
    textBox188.Text += textBox66.Text + ",";

    textBox188.Text += textBox65.Text + ",";

    textBox188.Text += textBox64.Text + ",";
    textBox188.Text += textBox62.Text + ",";
    textBox188.Text += textBox60.Text + ",";
    textBox188.Text += textBox58.Text + ",";
    textBox188.Text += textBox63.Text + ",";
    textBox188.Text += textBox61.Text + ",";
    textBox188.Text += textBox59.Text + ",";
    textBox188.Text += textBox57.Text + ",";
}

void f7_parametre_aktar()
{
    if (radioButton68.Checked) textBox188.Text = "1,"; else
textBox188.Text = "0,";
    if (radioButton77.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
    if (radioButton76.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
    if (radioButton75.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
    if (radioButton74.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
    if (radioButton73.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
    if (radioButton72.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
    if (radioButton71.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
}

```

```

        if (radioButton70.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
        if (radioButton69.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";

        textBox188.Text += textBox78.Text + ",";
        textBox188.Text += textBox173.Text + ",";
        textBox188.Text += textBox77.Text + ",";

        textBox188.Text += textBox76.Text + ",";

        textBox188.Text += textBox75.Text + ",";
        textBox188.Text += textBox73.Text + ",";
        textBox188.Text += textBox71.Text + ",";
        textBox188.Text += textBox69.Text + ",";
        textBox188.Text += textBox74.Text + ",";
        textBox188.Text += textBox72.Text + ",";
        textBox188.Text += textBox70.Text + ",";
        textBox188.Text += textBox68.Text + ",";
    }

void f8_parametre_aktar()
{
    if (radioButton78.Checked) textBox188.Text = "1,"; else
textBox188.Text = "0,";
    if (radioButton87.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
    if (radioButton86.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
    if (radioButton85.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
    if (radioButton84.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
    if (radioButton83.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
    if (radioButton82.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
    if (radioButton81.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
    if (radioButton80.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
    if (radioButton79.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";

        textBox188.Text += textBox89.Text + ",";
        textBox188.Text += textBox174.Text + ",";
        textBox188.Text += textBox88.Text + ",";

        textBox188.Text += textBox87.Text + ",";

        textBox188.Text += textBox86.Text + ",";
        textBox188.Text += textBox84.Text + ",";
        textBox188.Text += textBox82.Text + ",";
        textBox188.Text += textBox80.Text + ",";
        textBox188.Text += textBox85.Text + ",";
        textBox188.Text += textBox83.Text + ",";
        textBox188.Text += textBox81.Text + ",";
        textBox188.Text += textBox79.Text + ",";

```

```

    }

    void f9_parametre_aktar()
    {
        if (radioButton88.Checked) textBox188.Text = "1,"; else
        textBox188.Text = "0,";
        if (radioButton97.Checked) textBox188.Text += "1,"; else
        textBox188.Text += "0,";
        if (radioButton96.Checked) textBox188.Text += "1,"; else
        textBox188.Text += "0,";
        if (radioButton95.Checked) textBox188.Text += "1,"; else
        textBox188.Text += "0,";
        if (radioButton94.Checked) textBox188.Text += "1,"; else
        textBox188.Text += "0,";
        if (radioButton93.Checked) textBox188.Text += "1,"; else
        textBox188.Text += "0,";
        if (radioButton92.Checked) textBox188.Text += "1,"; else
        textBox188.Text += "0,";
        if (radioButton91.Checked) textBox188.Text += "1,"; else
        textBox188.Text += "0,";
        if (radioButton90.Checked) textBox188.Text += "1,"; else
        textBox188.Text += "0,";
        if (radioButton89.Checked) textBox188.Text += "1,"; else
        textBox188.Text += "0,";

        textBox188.Text += textBox100.Text + ",";
        textBox188.Text += textBox175.Text + ",";
        textBox188.Text += textBox99.Text + ",";

        textBox188.Text += textBox98.Text + ",";

        textBox188.Text += textBox97.Text + ",";
        textBox188.Text += textBox95.Text + ",";
        textBox188.Text += textBox93.Text + ",";
        textBox188.Text += textBox91.Text + ",";
        textBox188.Text += textBox96.Text + ",";
        textBox188.Text += textBox94.Text + ",";
        textBox188.Text += textBox92.Text + ",";
        textBox188.Text += textBox90.Text + ",";
    }

    void f10_parametre_aktar()
    {
        if (radioButton98.Checked) textBox188.Text = "1,"; else
        textBox188.Text = "0,";
        if (radioButton107.Checked) textBox188.Text += "1,"; else
        textBox188.Text += "0,";
        if (radioButton106.Checked) textBox188.Text += "1,"; else
        textBox188.Text += "0,";
        if (radioButton105.Checked) textBox188.Text += "1,"; else
        textBox188.Text += "0,";
        if (radioButton104.Checked) textBox188.Text += "1,"; else
        textBox188.Text += "0,";
        if (radioButton103.Checked) textBox188.Text += "1,"; else
        textBox188.Text += "0,";
        if (radioButton102.Checked) textBox188.Text += "1,"; else
        textBox188.Text += "0,";
    }

```

```

        if (radioButton101.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
        if (radioButton100.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
        if (radioButton99.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";

        textBox188.Text += textBox111.Text + ",";
        textBox188.Text += textBox176.Text + ",";
        textBox188.Text += textBox110.Text + ",";

        textBox188.Text += textBox109.Text + ",";

        textBox188.Text += textBox108.Text + ",";
        textBox188.Text += textBox106.Text + ",";
        textBox188.Text += textBox104.Text + ",";
        textBox188.Text += textBox102.Text + ",";
        textBox188.Text += textBox107.Text + ",";
        textBox188.Text += textBox105.Text + ",";
        textBox188.Text += textBox103.Text + ",";
        textBox188.Text += textBox101.Text + ",";
    }

    void f11_parametre_aktar()
    {
        if (radioButton108.Checked) textBox188.Text = "1,"; else
textBox188.Text = "0,";
        if (radioButton117.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
        if (radioButton116.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
        if (radioButton115.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
        if (radioButton114.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
        if (radioButton113.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
        if (radioButton112.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
        if (radioButton111.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
        if (radioButton110.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
        if (radioButton109.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";

        textBox188.Text += textBox122.Text + ",";
        textBox188.Text += textBox177.Text + ",";
        textBox188.Text += textBox121.Text + ",";

        textBox188.Text += textBox120.Text + ",";

        textBox188.Text += textBox119.Text + ",";
        textBox188.Text += textBox117.Text + ",";
        textBox188.Text += textBox115.Text + ",";
        textBox188.Text += textBox113.Text + ",";
        textBox188.Text += textBox118.Text + ",";
        textBox188.Text += textBox116.Text + ",";
    }

```

```

        textBox188.Text += textBox114.Text + ",";
        textBox188.Text += textBox112.Text + ",";
    }

    void f12_parametre_aktar()
    {
        if (radioButton118.Checked) textBox188.Text = "1,"; else
        textBox188.Text = "0,";
        if (radioButton127.Checked) textBox188.Text += "1,"; else
        textBox188.Text += "0,";
        if (radioButton126.Checked) textBox188.Text += "1,"; else
        textBox188.Text += "0,";
        if (radioButton125.Checked) textBox188.Text += "1,"; else
        textBox188.Text += "0,";
        if (radioButton124.Checked) textBox188.Text += "1,"; else
        textBox188.Text += "0,";
        if (radioButton123.Checked) textBox188.Text += "1,"; else
        textBox188.Text += "0,";
        if (radioButton122.Checked) textBox188.Text += "1,"; else
        textBox188.Text += "0,";
        if (radioButton121.Checked) textBox188.Text += "1,"; else
        textBox188.Text += "0,";
        if (radioButton120.Checked) textBox188.Text += "1,"; else
        textBox188.Text += "0,";
        if (radioButton119.Checked) textBox188.Text += "1,"; else
        textBox188.Text += "0,";

        textBox188.Text += textBox133.Text + ",";
        textBox188.Text += textBox178.Text + ",";
        textBox188.Text += textBox132.Text + ",";

        textBox188.Text += textBox131.Text + ",";

        textBox188.Text += textBox130.Text + ",";
        textBox188.Text += textBox128.Text + ",";
        textBox188.Text += textBox126.Text + ",";
        textBox188.Text += textBox124.Text + ",";
        textBox188.Text += textBox129.Text + ",";
        textBox188.Text += textBox127.Text + ",";
        textBox188.Text += textBox125.Text + ",";
        textBox188.Text += textBox123.Text + ",";
    }

    void f13_parametre_aktar()
    {
        if (radioButton128.Checked) textBox188.Text = "1,"; else
        textBox188.Text = "0,";
        if (radioButton137.Checked) textBox188.Text += "1,"; else
        textBox188.Text += "0,";
        if (radioButton136.Checked) textBox188.Text += "1,"; else
        textBox188.Text += "0,";
        if (radioButton135.Checked) textBox188.Text += "1,"; else
        textBox188.Text += "0,";
        if (radioButton134.Checked) textBox188.Text += "1,"; else
        textBox188.Text += "0,";
        if (radioButton133.Checked) textBox188.Text += "1,"; else
        textBox188.Text += "0,";
    }

```

```

        if (radioButton132.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
        if (radioButton131.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
        if (radioButton130.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
        if (radioButton129.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";

        textBox188.Text += textBox144.Text + ",";
        textBox188.Text += textBox179.Text + ",";
        textBox188.Text += textBox143.Text + ",";

        textBox188.Text += textBox142.Text + ",";

        textBox188.Text += textBox141.Text + ",";
        textBox188.Text += textBox139.Text + ",";
        textBox188.Text += textBox137.Text + ",";
        textBox188.Text += textBox135.Text + ",";
        textBox188.Text += textBox140.Text + ",";
        textBox188.Text += textBox138.Text + ",";
        textBox188.Text += textBox136.Text + ",";
        textBox188.Text += textBox134.Text + ",";
    }

    void f14_parametre_aktar()
    {
        if (radioButton138.Checked) textBox188.Text = "1,"; else
textBox188.Text = "0,";
        if (radioButton147.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
        if (radioButton146.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
        if (radioButton145.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
        if (radioButton144.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
        if (radioButton143.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
        if (radioButton142.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
        if (radioButton141.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
        if (radioButton140.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";
        if (radioButton139.Checked) textBox188.Text += "1,"; else
textBox188.Text += "0,";

        textBox188.Text += textBox155.Text + ",";
        textBox188.Text += textBox180.Text + ",";
        textBox188.Text += textBox154.Text + ",";

        textBox188.Text += textBox153.Text + ",";

        textBox188.Text += textBox152.Text + ",";
        textBox188.Text += textBox150.Text + ",";
        textBox188.Text += textBox148.Text + ",";
        textBox188.Text += textBox146.Text + ",";
    }

```

```

        textBox188.Text += textBox151.Text + ",";
        textBox188.Text += textBox149.Text + ",";
        textBox188.Text += textBox147.Text + ",";
        textBox188.Text += textBox145.Text + ",";
    }

    void f15_parametre_aktar()
    {
        if (radioButton148.Checked) textBox188.Text = "1,"; else
        textBox188.Text = "0,";
        if (radioButton157.Checked) textBox188.Text += "1,"; else
        textBox188.Text += "0,";
        if (radioButton156.Checked) textBox188.Text += "1,"; else
        textBox188.Text += "0,";
        if (radioButton155.Checked) textBox188.Text += "1,"; else
        textBox188.Text += "0,";
        if (radioButton154.Checked) textBox188.Text += "1,"; else
        textBox188.Text += "0,";
        if (radioButton153.Checked) textBox188.Text += "1,"; else
        textBox188.Text += "0,";
        if (radioButton152.Checked) textBox188.Text += "1,"; else
        textBox188.Text += "0,";
        if (radioButton151.Checked) textBox188.Text += "1,"; else
        textBox188.Text += "0,";
        if (radioButton150.Checked) textBox188.Text += "1,"; else
        textBox188.Text += "0,";
        if (radioButton149.Checked) textBox188.Text += "1,"; else
        textBox188.Text += "0,";

        textBox188.Text += textBox166.Text + ",";
        textBox188.Text += textBox181.Text + ",";
        textBox188.Text += textBox165.Text + ",";

        textBox188.Text += textBox164.Text + ",";

        textBox188.Text += textBox163.Text + ",";
        textBox188.Text += textBox161.Text + ",";
        textBox188.Text += textBox159.Text + ",";
        textBox188.Text += textBox157.Text + ",";
        textBox188.Text += textBox162.Text + ",";
        textBox188.Text += textBox160.Text + ",";
        textBox188.Text += textBox158.Text + ",";
        textBox188.Text += textBox156.Text + ",";
    }

    private void button8_Click(object sender, EventArgs e)
    {
        string dosya_yolu;
        FileStream yazma_ozellik;
        StreamWriter dosya_ac_hafiza;

        saveFileDialog1.Filter = "Files:|*.brcd";
        saveFileDialog1.Title = "Save the file";
        saveFileDialog1.InitialDirectory = dosya_yolu_basla;

        if (saveFileDialog1.ShowDialog() == DialogResult.OK)
        {

```

```

        dosya_yolu_basla =
System.IO.Path.GetDirectoryName(saveFileDialog1.FileName);

        dosya_yolu = saveFileDialog1.FileName;
        yazma_ozellik = new FileStream(dosya_yolu, FileMode.Create,
FileAccess.Write);
        dosya_ac_hafiza = new StreamWriter(yazma_ozellik);

        f1_parametre_aktar(); dosya_ac_hafiza.WriteLine(textBox188.Text);
        f2_parametre_aktar(); dosya_ac_hafiza.WriteLine(textBox188.Text);
        f3_parametre_aktar(); dosya_ac_hafiza.WriteLine(textBox188.Text);
        f4_parametre_aktar(); dosya_ac_hafiza.WriteLine(textBox188.Text);
        f5_parametre_aktar(); dosya_ac_hafiza.WriteLine(textBox188.Text);
        f6_parametre_aktar(); dosya_ac_hafiza.WriteLine(textBox188.Text);
        f7_parametre_aktar(); dosya_ac_hafiza.WriteLine(textBox188.Text);
        f8_parametre_aktar(); dosya_ac_hafiza.WriteLine(textBox188.Text);
        f9_parametre_aktar(); dosya_ac_hafiza.WriteLine(textBox188.Text);
        f10_parametre_aktar();
dosya_ac_hafiza.WriteLine(textBox188.Text);
        f11_parametre_aktar();
dosya_ac_hafiza.WriteLine(textBox188.Text);
        f12_parametre_aktar();
dosya_ac_hafiza.WriteLine(textBox188.Text);
        f13_parametre_aktar();
dosya_ac_hafiza.WriteLine(textBox188.Text);
        f14_parametre_aktar();
dosya_ac_hafiza.WriteLine(textBox188.Text);
        f15_parametre_aktar();
dosya_ac_hafiza.WriteLine(textBox188.Text);

        dosya_ac_hafiza.Flush();//ram dosya verilerini HD dosyaya aktar
        dosya_ac_hafiza.Close();
    }
}

void f1_parametre_al()
{
    Int32 say;
    char[] veri;
    veri = new char[255];
    veri = textBox188.Text.ToCharArray();

    say = 0;
    if (veri[say] == '1') radioButton11.Checked = true; say+=2;
    if (veri[say] == '1') radioButton1.Checked = true; say += 2;
    if (veri[say] == '1') radioButton2.Checked = true; say += 2;
    if (veri[say] == '1') radioButton3.Checked = true; say += 2;
    if (veri[say] == '1') radioButton5.Checked = true; say += 2;
    if (veri[say] == '1') radioButton6.Checked = true; say += 2;
    if (veri[say] == '1') radioButton8.Checked = true; say += 2;
    if (veri[say] == '1') radioButton10.Checked = true; say += 2;
    if (veri[say] == '1') radioButton19.Checked = true; say += 2;
    if (veri[say] == '1') radioButton20.Checked = true; say += 2;

    textBox2.Text = "";
    for (say +=0; veri[say] != ','; say++) textBox2.Text +=
Convert.ToString(veri[say]);
}

```

```

        say++;

        textBox167.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox167.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox3.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox3.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox12.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox12.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox4.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox4.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox5.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox5.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox6.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox6.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox7.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox7.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox8.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox8.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox9.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox9.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox10.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox10.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox11.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox11.Text +=
Convert.ToString(veri[say]);
        say++;
    }

```

```

void f2_parametre_al()
{
    Int32 say;
    char[] veri;
    veri = new char[255];
    veri = textBox188.Text.ToCharArray();

    say = 0;
    if (veri[say] == '1') radioButton4.Checked = true; say += 2;
    if (veri[say] == '1') radioButton27.Checked = true; say += 2;
    if (veri[say] == '1') radioButton26.Checked = true; say += 2;
    if (veri[say] == '1') radioButton25.Checked = true; say += 2;
    if (veri[say] == '1') radioButton24.Checked = true; say += 2;
    if (veri[say] == '1') radioButton23.Checked = true; say += 2;
    if (veri[say] == '1') radioButton22.Checked = true; say += 2;
    if (veri[say] == '1') radioButton21.Checked = true; say += 2;
    if (veri[say] == '1') radioButton9.Checked = true; say += 2;
    if (veri[say] == '1') radioButton7.Checked = true; say += 2;

    textBox23.Text = "";
    for (say += 0; veri[say] != ','; say++) textBox23.Text +=
Convert.ToString(veri[say]);
    say++;

    textBox168.Text = "";
    for (say += 0; veri[say] != ','; say++) textBox168.Text +=
Convert.ToString(veri[say]);
    say++;

    textBox22.Text = "";
    for (say += 0; veri[say] != ','; say++) textBox22.Text +=
Convert.ToString(veri[say]);
    say++;

    textBox21.Text = "";
    for (say += 0; veri[say] != ','; say++) textBox21.Text +=
Convert.ToString(veri[say]);
    say++;

    textBox20.Text = "";
    for (say += 0; veri[say] != ','; say++) textBox20.Text +=
Convert.ToString(veri[say]);
    say++;

    textBox18.Text = "";
    for (say += 0; veri[say] != ','; say++) textBox18.Text +=
Convert.ToString(veri[say]);
    say++;

    textBox16.Text = "";
    for (say += 0; veri[say] != ','; say++) textBox16.Text +=
Convert.ToString(veri[say]);
    say++;

    textBox14.Text = "";
    for (say += 0; veri[say] != ','; say++) textBox14.Text +=
Convert.ToString(veri[say]);
    say++;
}

```

```

        textBox19.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox19.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox17.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox17.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox15.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox15.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox13.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox13.Text +=
Convert.ToString(veri[say]);
        say++;
    }

    void f3_parametre_al()
    {
        Int32 say;
        char[] veri;
        veri = new char[255];
        veri = textBox188.Text.ToCharArray();

        say = 0;
        if (veri[say] == '1') radioButton28.Checked = true; say += 2;
        if (veri[say] == '1') radioButton37.Checked = true; say += 2;
        if (veri[say] == '1') radioButton36.Checked = true; say += 2;
        if (veri[say] == '1') radioButton35.Checked = true; say += 2;
        if (veri[say] == '1') radioButton34.Checked = true; say += 2;
        if (veri[say] == '1') radioButton33.Checked = true; say += 2;
        if (veri[say] == '1') radioButton32.Checked = true; say += 2;
        if (veri[say] == '1') radioButton31.Checked = true; say += 2;
        if (veri[say] == '1') radioButton30.Checked = true; say += 2;
        if (veri[say] == '1') radioButton29.Checked = true; say += 2;

        textBox34.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox34.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox169.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox169.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox33.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox33.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox32.Text = "";

```

```

        for (say += 0; veri[say] != ','; say++) textBox32.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox31.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox31.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox29.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox29.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox27.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox27.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox25.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox25.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox30.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox30.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox28.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox28.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox26.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox26.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox24.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox24.Text +=
Convert.ToString(veri[say]);
        say++;
    }

    void f4_parametre_al()
    {
        Int32 say;
        char[] veri;
        veri = new char[255];
        veri = textBox188.Text.ToCharArray();

        say = 0;
        if (veri[say] == '1') radioButton38.Checked = true; say += 2;
        if (veri[say] == '1') radioButton47.Checked = true; say += 2;
        if (veri[say] == '1') radioButton46.Checked = true; say += 2;
        if (veri[say] == '1') radioButton45.Checked = true; say += 2;
        if (veri[say] == '1') radioButton44.Checked = true; say += 2;
    }

```

```

if (veri[say] == '1') radioButton43.Checked = true; say += 2;
if (veri[say] == '1') radioButton42.Checked = true; say += 2;
if (veri[say] == '1') radioButton41.Checked = true; say += 2;
if (veri[say] == '1') radioButton40.Checked = true; say += 2;
if (veri[say] == '1') radioButton39.Checked = true; say += 2;

textBox45.Text = "";
for (say += 0; veri[say] != ','; say++) textBox45.Text +=
Convert.ToString(veri[say]);
say++;

textBox170.Text = "";
for (say += 0; veri[say] != ','; say++) textBox170.Text +=
Convert.ToString(veri[say]);
say++;

textBox44.Text = "";
for (say += 0; veri[say] != ','; say++) textBox44.Text +=
Convert.ToString(veri[say]);
say++;

textBox43.Text = "";
for (say += 0; veri[say] != ','; say++) textBox43.Text +=
Convert.ToString(veri[say]);
say++;

textBox42.Text = "";
for (say += 0; veri[say] != ','; say++) textBox42.Text +=
Convert.ToString(veri[say]);
say++;

textBox40.Text = "";
for (say += 0; veri[say] != ','; say++) textBox40.Text +=
Convert.ToString(veri[say]);
say++;

textBox38.Text = "";
for (say += 0; veri[say] != ','; say++) textBox38.Text +=
Convert.ToString(veri[say]);
say++;

textBox36.Text = "";
for (say += 0; veri[say] != ','; say++) textBox36.Text +=
Convert.ToString(veri[say]);
say++;

textBox41.Text = "";
for (say += 0; veri[say] != ','; say++) textBox41.Text +=
Convert.ToString(veri[say]);
say++;

textBox39.Text = "";
for (say += 0; veri[say] != ','; say++) textBox39.Text +=
Convert.ToString(veri[say]);
say++;

textBox37.Text = "";

```

```

        for (say += 0; veri[say] != ','; say++) textBox37.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox35.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox35.Text +=
Convert.ToString(veri[say]);
        say++;
    }

    void f5_parametre_al()
    {
        Int32 say;
        char[] veri;
        veri = new char[255];
        veri = textBox188.Text.ToCharArray();

        say = 0;
        if (veri[say] == '1') radioButton48.Checked = true; say += 2;
        if (veri[say] == '1') radioButton57.Checked = true; say += 2;
        if (veri[say] == '1') radioButton56.Checked = true; say += 2;
        if (veri[say] == '1') radioButton55.Checked = true; say += 2;
        if (veri[say] == '1') radioButton54.Checked = true; say += 2;
        if (veri[say] == '1') radioButton53.Checked = true; say += 2;
        if (veri[say] == '1') radioButton52.Checked = true; say += 2;
        if (veri[say] == '1') radioButton51.Checked = true; say += 2;
        if (veri[say] == '1') radioButton50.Checked = true; say += 2;
        if (veri[say] == '1') radioButton49.Checked = true; say += 2;

        textBox56.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox56.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox171.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox171.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox55.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox55.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox54.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox54.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox53.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox53.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox51.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox51.Text +=
Convert.ToString(veri[say]);
        say++;
    }

```

```

        textBox49.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox49.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox47.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox47.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox52.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox52.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox50.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox50.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox48.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox48.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox46.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox46.Text +=
Convert.ToString(veri[say]);
        say++;
    }

    void f6_parametre_al()
    {
        Int32 say;
        char[] veri;
        veri = new char[255];
        veri = textBox188.Text.ToCharArray();

        say = 0;
        if (veri[say] == '1') radioButton58.Checked = true; say += 2;
        if (veri[say] == '1') radioButton67.Checked = true; say += 2;
        if (veri[say] == '1') radioButton66.Checked = true; say += 2;
        if (veri[say] == '1') radioButton65.Checked = true; say += 2;
        if (veri[say] == '1') radioButton64.Checked = true; say += 2;
        if (veri[say] == '1') radioButton63.Checked = true; say += 2;
        if (veri[say] == '1') radioButton62.Checked = true; say += 2;
        if (veri[say] == '1') radioButton61.Checked = true; say += 2;
        if (veri[say] == '1') radioButton60.Checked = true; say += 2;
        if (veri[say] == '1') radioButton59.Checked = true; say += 2;

        textBox67.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox67.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox172.Text = "";

```

```

        for (say += 0; veri[say] != ','; say++) textBox172.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox66.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox66.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox65.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox65.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox64.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox64.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox62.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox62.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox60.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox60.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox58.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox58.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox63.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox63.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox61.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox61.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox59.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox59.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox57.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox57.Text +=
Convert.ToString(veri[say]);
        say++;
    }

    void f7_parametre_al()
    {
        Int32 say;

```

```

char[] veri;
veri = new char[255];
veri = textBox188.Text.ToCharArray();

say = 0;
if (veri[say] == '1') radioButton68.Checked = true; say += 2;
if (veri[say] == '1') radioButton77.Checked = true; say += 2;
if (veri[say] == '1') radioButton76.Checked = true; say += 2;
if (veri[say] == '1') radioButton75.Checked = true; say += 2;
if (veri[say] == '1') radioButton74.Checked = true; say += 2;
if (veri[say] == '1') radioButton73.Checked = true; say += 2;
if (veri[say] == '1') radioButton72.Checked = true; say += 2;
if (veri[say] == '1') radioButton71.Checked = true; say += 2;
if (veri[say] == '1') radioButton70.Checked = true; say += 2;
if (veri[say] == '1') radioButton69.Checked = true; say += 2;

textBox78.Text = "";
for (say += 0; veri[say] != ','; say++) textBox78.Text +=
Convert.ToString(veri[say]);
say++;

textBox173.Text = "";
for (say += 0; veri[say] != ','; say++) textBox173.Text +=
Convert.ToString(veri[say]);
say++;

textBox77.Text = "";
for (say += 0; veri[say] != ','; say++) textBox77.Text +=
Convert.ToString(veri[say]);
say++;

textBox76.Text = "";
for (say += 0; veri[say] != ','; say++) textBox76.Text +=
Convert.ToString(veri[say]);
say++;

textBox75.Text = "";
for (say += 0; veri[say] != ','; say++) textBox75.Text +=
Convert.ToString(veri[say]);
say++;

textBox73.Text = "";
for (say += 0; veri[say] != ','; say++) textBox73.Text +=
Convert.ToString(veri[say]);
say++;

textBox71.Text = "";
for (say += 0; veri[say] != ','; say++) textBox71.Text +=
Convert.ToString(veri[say]);
say++;

textBox69.Text = "";
for (say += 0; veri[say] != ','; say++) textBox69.Text +=
Convert.ToString(veri[say]);
say++;

textBox74.Text = "";

```

```

        for (say += 0; veri[say] != ','; say++) textBox74.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox72.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox72.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox70.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox70.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox68.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox68.Text +=
Convert.ToString(veri[say]);
        say++;
    }

    void f8_parametre_al()
    {
        Int32 say;
        char[] veri;
        veri = new char[255];
        veri = textBox188.Text.ToCharArray();

        say = 0;
        if (veri[say] == '1') radioButton78.Checked = true; say += 2;
        if (veri[say] == '1') radioButton87.Checked = true; say += 2;
        if (veri[say] == '1') radioButton86.Checked = true; say += 2;
        if (veri[say] == '1') radioButton85.Checked = true; say += 2;
        if (veri[say] == '1') radioButton84.Checked = true; say += 2;
        if (veri[say] == '1') radioButton83.Checked = true; say += 2;
        if (veri[say] == '1') radioButton82.Checked = true; say += 2;
        if (veri[say] == '1') radioButton81.Checked = true; say += 2;
        if (veri[say] == '1') radioButton80.Checked = true; say += 2;
        if (veri[say] == '1') radioButton79.Checked = true; say += 2;

        textBox89.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox89.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox174.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox174.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox88.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox88.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox87.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox87.Text +=
Convert.ToString(veri[say]);
        say++;
    }

```

```

        textBox86.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox86.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox84.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox84.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox82.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox82.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox80.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox80.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox85.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox85.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox83.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox83.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox81.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox81.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox79.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox79.Text +=
Convert.ToString(veri[say]);
        say++;
    }

    void f9_parametre_al()
    {
        Int32 say;
        char[] veri;
        veri = new char[255];
        veri = textBox188.Text.ToCharArray();

        say = 0;
        if (veri[say] == '1') radioButton88.Checked = true; say += 2;
        if (veri[say] == '1') radioButton97.Checked = true; say += 2;
        if (veri[say] == '1') radioButton96.Checked = true; say += 2;
        if (veri[say] == '1') radioButton95.Checked = true; say += 2;
        if (veri[say] == '1') radioButton94.Checked = true; say += 2;
        if (veri[say] == '1') radioButton93.Checked = true; say += 2;
        if (veri[say] == '1') radioButton92.Checked = true; say += 2;
        if (veri[say] == '1') radioButton91.Checked = true; say += 2;
    }

```

```

        if (veri[say] == '1') radioButton90.Checked = true; say += 2;
        if (veri[say] == '1') radioButton89.Checked = true; say += 2;

        textBox100.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox100.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox175.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox175.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox99.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox99.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox98.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox98.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox97.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox97.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox95.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox95.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox93.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox93.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox91.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox91.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox96.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox96.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox94.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox94.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox92.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox92.Text +=
Convert.ToString(veri[say]);
        say++;

```

```

        textBox90.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox90.Text +=
Convert.ToString(veri[say]);
        say++;
    }

    void f10_parametre_al()
    {
        Int32 say;
        char[] veri;
        veri = new char[255];
        veri = textBox188.Text.ToCharArray();

        say = 0;
        if (veri[say] == '1') radioButton98.Checked = true; say += 2;
        if (veri[say] == '1') radioButton107.Checked = true; say += 2;
        if (veri[say] == '1') radioButton106.Checked = true; say += 2;
        if (veri[say] == '1') radioButton105.Checked = true; say += 2;
        if (veri[say] == '1') radioButton104.Checked = true; say += 2;
        if (veri[say] == '1') radioButton103.Checked = true; say += 2;
        if (veri[say] == '1') radioButton102.Checked = true; say += 2;
        if (veri[say] == '1') radioButton101.Checked = true; say += 2;
        if (veri[say] == '1') radioButton100.Checked = true; say += 2;
        if (veri[say] == '1') radioButton99.Checked = true; say += 2;

        textBox111.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox111.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox176.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox176.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox110.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox110.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox109.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox109.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox108.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox108.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox106.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox106.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox104.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox104.Text +=
Convert.ToString(veri[say]);

```

```

say++;

textBox102.Text = "";
for (say += 0; veri[say] != ','; say++) textBox102.Text +=
Convert.ToString(veri[say]);
say++;

textBox107.Text = "";
for (say += 0; veri[say] != ','; say++) textBox107.Text +=
Convert.ToString(veri[say]);
say++;

textBox105.Text = "";
for (say += 0; veri[say] != ','; say++) textBox105.Text +=
Convert.ToString(veri[say]);
say++;

textBox103.Text = "";
for (say += 0; veri[say] != ','; say++) textBox103.Text +=
Convert.ToString(veri[say]);
say++;

textBox101.Text = "";
for (say += 0; veri[say] != ','; say++) textBox101.Text +=
Convert.ToString(veri[say]);
say++;
}

void f11_parametre_al()
{
    Int32 say;
    char[] veri;
    veri = new char[255];
    veri = textBox188.Text.ToCharArray();

    say = 0;
    if (veri[say] == '1') radioButton108.Checked = true; say += 2;
    if (veri[say] == '1') radioButton117.Checked = true; say += 2;
    if (veri[say] == '1') radioButton116.Checked = true; say += 2;
    if (veri[say] == '1') radioButton115.Checked = true; say += 2;
    if (veri[say] == '1') radioButton114.Checked = true; say += 2;
    if (veri[say] == '1') radioButton113.Checked = true; say += 2;
    if (veri[say] == '1') radioButton112.Checked = true; say += 2;
    if (veri[say] == '1') radioButton111.Checked = true; say += 2;
    if (veri[say] == '1') radioButton110.Checked = true; say += 2;
    if (veri[say] == '1') radioButton109.Checked = true; say += 2;

    textBox122.Text = "";
    for (say += 0; veri[say] != ','; say++) textBox122.Text +=
Convert.ToString(veri[say]);
say++;

    textBox177.Text = "";
    for (say += 0; veri[say] != ','; say++) textBox177.Text +=
Convert.ToString(veri[say]);
say++;

    textBox121.Text = "";

```

```

        for (say += 0; veri[say] != ','; say++) textBox121.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox120.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox120.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox119.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox119.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox117.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox117.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox115.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox115.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox113.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox113.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox118.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox118.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox116.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox116.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox114.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox114.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox112.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox112.Text +=
Convert.ToString(veri[say]);
        say++;
    }

    void f12_parametre_al()
    {
        Int32 say;
        char[] veri;
        veri = new char[255];
        veri = textBox188.Text.ToCharArray();

        say = 0;

```

```

        if (veri[say] == '1') radioButton118.Checked = true; say += 2;
        if (veri[say] == '1') radioButton127.Checked = true; say += 2;
        if (veri[say] == '1') radioButton126.Checked = true; say += 2;
        if (veri[say] == '1') radioButton125.Checked = true; say += 2;
        if (veri[say] == '1') radioButton124.Checked = true; say += 2;
        if (veri[say] == '1') radioButton123.Checked = true; say += 2;
        if (veri[say] == '1') radioButton122.Checked = true; say += 2;
        if (veri[say] == '1') radioButton121.Checked = true; say += 2;
        if (veri[say] == '1') radioButton120.Checked = true; say += 2;
        if (veri[say] == '1') radioButton119.Checked = true; say += 2;

        textBox133.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox133.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox178.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox178.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox132.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox132.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox131.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox131.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox130.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox130.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox128.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox128.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox126.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox126.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox124.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox124.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox129.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox129.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox127.Text = "";

```

```

        for (say += 0; veri[say] != ','; say++) textBox127.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox125.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox125.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox123.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox123.Text +=
Convert.ToString(veri[say]);
        say++;
    }

    void f13_parametre_al()
    {
        Int32 say;
        char[] veri;
        veri = new char[255];
        veri = textBox188.Text.ToCharArray();

        say = 0;
        if (veri[say] == '1') radioButton128.Checked = true; say += 2;
        if (veri[say] == '1') radioButton137.Checked = true; say += 2;
        if (veri[say] == '1') radioButton136.Checked = true; say += 2;
        if (veri[say] == '1') radioButton135.Checked = true; say += 2;
        if (veri[say] == '1') radioButton134.Checked = true; say += 2;
        if (veri[say] == '1') radioButton133.Checked = true; say += 2;
        if (veri[say] == '1') radioButton132.Checked = true; say += 2;
        if (veri[say] == '1') radioButton131.Checked = true; say += 2;
        if (veri[say] == '1') radioButton130.Checked = true; say += 2;
        if (veri[say] == '1') radioButton129.Checked = true; say += 2;

        textBox144.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox144.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox179.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox179.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox143.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox143.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox142.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox142.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox141.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox141.Text +=
Convert.ToString(veri[say]);
        say++;
    }

```

```

        textBox139.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox139.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox137.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox137.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox135.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox135.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox140.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox140.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox138.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox138.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox136.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox136.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox134.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox134.Text +=
Convert.ToString(veri[say]);
        say++;
    }

    void f14_parametre_al()
    {
        Int32 say;
        char[] veri;
        veri = new char[255];
        veri = textBox188.Text.ToCharArray();

        say = 0;
        if (veri[say] == '1') radioButton138.Checked = true; say += 2;
        if (veri[say] == '1') radioButton147.Checked = true; say += 2;
        if (veri[say] == '1') radioButton146.Checked = true; say += 2;
        if (veri[say] == '1') radioButton145.Checked = true; say += 2;
        if (veri[say] == '1') radioButton144.Checked = true; say += 2;
        if (veri[say] == '1') radioButton143.Checked = true; say += 2;
        if (veri[say] == '1') radioButton142.Checked = true; say += 2;
        if (veri[say] == '1') radioButton141.Checked = true; say += 2;
        if (veri[say] == '1') radioButton140.Checked = true; say += 2;
        if (veri[say] == '1') radioButton139.Checked = true; say += 2;

        textBox155.Text = "";
    }

```

```

        for (say += 0; veri[say] != ','; say++) textBox155.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox180.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox180.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox154.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox154.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox153.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox153.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox152.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox152.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox150.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox150.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox148.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox148.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox146.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox146.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox151.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox151.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox149.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox149.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox147.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox147.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox145.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox145.Text +=
Convert.ToString(veri[say]);
        say++;

```

```

}

void f15_parametre_al()
{
    Int32 say;
    char[] veri;
    veri = new char[255];
    veri = textBox188.Text.ToCharArray();

    say = 0;
    if (veri[say] == '1') radioButton148.Checked = true; say += 2;
    if (veri[say] == '1') radioButton157.Checked = true; say += 2;
    if (veri[say] == '1') radioButton156.Checked = true; say += 2;
    if (veri[say] == '1') radioButton155.Checked = true; say += 2;
    if (veri[say] == '1') radioButton154.Checked = true; say += 2;
    if (veri[say] == '1') radioButton153.Checked = true; say += 2;
    if (veri[say] == '1') radioButton152.Checked = true; say += 2;
    if (veri[say] == '1') radioButton151.Checked = true; say += 2;
    if (veri[say] == '1') radioButton150.Checked = true; say += 2;
    if (veri[say] == '1') radioButton149.Checked = true; say += 2;

    textBox166.Text = "";
    for (say += 0; veri[say] != ','; say++) textBox166.Text +=
Convert.ToString(veri[say]);
    say++;

    textBox181.Text = "";
    for (say += 0; veri[say] != ','; say++) textBox181.Text +=
Convert.ToString(veri[say]);
    say++;

    textBox165.Text = "";
    for (say += 0; veri[say] != ','; say++) textBox165.Text +=
Convert.ToString(veri[say]);
    say++;

    textBox164.Text = "";
    for (say += 0; veri[say] != ','; say++) textBox164.Text +=
Convert.ToString(veri[say]);
    say++;

    textBox163.Text = "";
    for (say += 0; veri[say] != ','; say++) textBox163.Text +=
Convert.ToString(veri[say]);
    say++;

    textBox161.Text = "";
    for (say += 0; veri[say] != ','; say++) textBox161.Text +=
Convert.ToString(veri[say]);
    say++;

    textBox159.Text = "";
    for (say += 0; veri[say] != ','; say++) textBox159.Text +=
Convert.ToString(veri[say]);
    say++;

    textBox157.Text = "";

```

```

        for (say += 0; veri[say] != ','; say++) textBox157.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox162.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox162.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox160.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox160.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox158.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox158.Text +=
Convert.ToString(veri[say]);
        say++;

        textBox156.Text = "";
        for (say += 0; veri[say] != ','; say++) textBox156.Text +=
Convert.ToString(veri[say]);
        say++;
    }

    private void button9_Click(object sender, EventArgs e)
    {
        string dosya_yolu;
        FileStream okuma_ozellik;
        StreamReader dosya_ac_hafiza;

        openFileDialog1.Filter = "Files:|*.brcd";
        openFileDialog1.Title = "Choose a file";
        openFileDialog1.InitialDirectory = dosya_yolu_basla;

        if (openFileDialog1.ShowDialog() == DialogResult.OK)
        {
            dosya_yolu = openFileDialog1.FileName;
            dosya_yolu_basla =
System.IO.Path.GetDirectoryName(openFileDialog1.FileName);

            okuma_ozellik = new FileStream(dosya_yolu,
FileMode.Open, FileAccess.Read);
            dosya_ac_hafiza = new StreamReader(okuma_ozellik);

            textBox188.Text = dosya_ac_hafiza.ReadLine(); f1_parametre_al();
            textBox188.Text = dosya_ac_hafiza.ReadLine(); f2_parametre_al();
            textBox188.Text = dosya_ac_hafiza.ReadLine(); f3_parametre_al();
            textBox188.Text = dosya_ac_hafiza.ReadLine(); f4_parametre_al();
            textBox188.Text = dosya_ac_hafiza.ReadLine(); f5_parametre_al();
            textBox188.Text = dosya_ac_hafiza.ReadLine(); f6_parametre_al();
            textBox188.Text = dosya_ac_hafiza.ReadLine(); f7_parametre_al();
            textBox188.Text = dosya_ac_hafiza.ReadLine(); f8_parametre_al();
            textBox188.Text = dosya_ac_hafiza.ReadLine(); f9_parametre_al();
            textBox188.Text = dosya_ac_hafiza.ReadLine(); f10_parametre_al();
            textBox188.Text = dosya_ac_hafiza.ReadLine(); f11_parametre_al();
            textBox188.Text = dosya_ac_hafiza.ReadLine(); f12_parametre_al();
            textBox188.Text = dosya_ac_hafiza.ReadLine(); f13_parametre_al();

```

```

        textBox188.Text = dosya_ac_hafiza.ReadLine(); f14_parametre_al();
        textBox188.Text = dosya_ac_hafiza.ReadLine(); f15_parametre_al();

        dosya_ac_hafiza.Close();
    }
}

private void textBox11_KeyPress(object sender, KeyPressEventArgs e)
{
    if (!char.IsControl(e.KeyChar) && !char.IsDigit(e.KeyChar))
    {
        e.Handled = true;
    }
}

private void textBox2_Leave(object sender, EventArgs e)
{
    double akim;

    Int32 dlt_lmax;
    Int32 dlt_lmin;

    dlt_lmax = 9999;
    dlt_lmin = 0;
    akim = 0;

    if (radioButton1.Checked == true) dlt_lmin = 10; else dlt_lmin = 50;

    try
    {
        akim = Convert.ToDouble(textBox2.Text);

        if ((Math.Abs(akim) > dlt_lmax) | (Math.Abs(akim) < dlt_lmin))
        {
            if (akim != 0)
            {
                MessageBox.Show("<delta I> and <delta T> should satisfy
ranges of:\n" +

                "\n" +
                "For Step Function:\n" + "10mA =< dI =<
10000mA; 1ms=< dT =< 10000ms\n" +

                "\n" +
                "For SinCos Functions:\n" + "50mA =< dI =<
10000mA; 20ms=< dT =< 10000ms\n" +

                "\n" +
                "For Ramp Functions:\n" + "50mA =< dI =<
10000mA; 5ms=< dT =< 10000ms\n" +

                "\n" +
                "For 1. and 2. Order System Functions:\n" +
                "50mA =< dI =< 10000mA; 10ms=< dT =< 10000ms\n",

                "WARNING",

```

```

        MessageBoxButtons.OK,
MessageBoxIcon.Warning);

        textBox2.Text = "1000";
    }

}

}
catch
{
    MessageBox.Show("<delta I> and <delta T> should satisfy ranges
of:\n" +

        "\n" +
        "For Step Functions:\n" + "10mA =< dI =<
10000mA; 1ms=< dT =< 10000ms\n" +

        "\n" +
        "For SinCos Functions:\n" + "50mA =< dI =<
10000mA; 20ms=< dT =< 10000ms\n" +

        "\n" +
        "For Ramp Functions:\n" + "50mA =< dI =<
10000mA; 5ms=< dT =< 10000ms\n" +

        "\n" +
        "For 1. and 2. Order System Functions:\n" +
        "50mA =< dI =< 10000mA; 10ms=< dT =< 10000ms\n",

        "WARNING",
        MessageBoxButtons.OK,
MessageBoxIcon.Warning);

        textBox2.Text = "1000";
    }
}

private void textBox23_Leave(object sender, EventArgs e)
{
    double aim;

    Int32 dIImax;
    Int32 dIImin;

    dIImax = 9999;
    dIImin = 0;
    aim = 0;

    if (radioButton27.Checked == true) dIImin = 10; else dIImin = 50;

    try
    {
        aim = Convert.ToDouble(textBox23.Text);

        if ((Math.Abs(aim) > dIImax) | (Math.Abs(aim) < dIImin))
        {

```



```

}

private void textBox34_Leave(object sender, EventArgs e)
{
    double akim;

    Int32 dlt_lmax;
    Int32 dlt_lmin;

    dlt_lmax = 9999;
    dlt_lmin = 0;
    akim = 0;

    if (radioButton37.Checked == true) dlt_lmin = 10; else dlt_lmin = 50;

    try
    {
        akim = Convert.ToDouble(textBox34.Text);

        if ((Math.Abs(akim) > dlt_lmax) | (Math.Abs(akim) < dlt_lmin))
        {
            if (akim != 0)
            {
                MessageBox.Show("<delta I> and <delta T> should satisfy
ranges of:\n" +
                                "\n" +
                                "For Step Function:\n" + "10mA =< dI =<
10000mA; 1ms=< dT =< 10000ms\n" +
                                "\n" +
                                "For SinCos Functions:\n" + "50mA =< dI =<
10000mA; 20ms=< dT =< 10000ms\n" +
                                "\n" +
                                "For Ramp Functions:\n" + "50mA =< dI =<
10000mA; 5ms=< dT =< 10000ms\n" +
                                "\n" +
                                "For 1. and 2. Order System Functions:\n" +
"50mA =< dI =< 10000mA; 10ms=< dT =< 10000ms\n",
                                "WARNING",
                                MessageBoxButtons.OK,
                                MessageBoxIcon.Warning);

                textBox34.Text = "1000";
            }
        }
    }
    catch
    {
        MessageBox.Show("<delta I> and <delta T> should satisfy ranges
of:\n" +

```

```

        "\n" +
        "For Step Function:\n" + "10mA =< dI =<
10000mA; 1ms=< dT =< 10000ms\n" +

        "\n" +
        "For SinCos Functions:\n" + "50mA =< dI =<
10000mA; 20ms=< dT =< 10000ms\n" +

        "\n" +
        "For Ramp Functions:\n" + "50mA =< dI =<
10000mA; 5ms=< dT =< 10000ms\n" +

        "\n" +
        "For 1. and 2. Order System Functions:\n" +
        "50mA =< dI =< 10000mA; 10ms=< dT =< 10000ms\n",

        "WARNING",
        MessageBoxButtons.OK,
        MessageBoxIcon.Warning);

        textBox34.Text = "1000";
    }
}

private void textBox45_Leave(object sender, EventArgs e)
{
    double akim;

    Int32 dlt_lmax;
    Int32 dlt_lmin;

    dlt_lmax = 9999;
    dlt_lmin = 0;
    akim = 0;

    if (radioButton47.Checked == true) dlt_lmin = 10; else dlt_lmin = 50;

    try
    {
        akim = Convert.ToDouble(textBox45.Text);

        if ((Math.Abs(akim) > dlt_lmax) | (Math.Abs(akim) < dlt_lmin))
        {
            if (akim != 0)
            {
                MessageBox.Show("<delta I> and <delta T> should satisfy
ranges of:\n" +

                "\n" +
                "For Step Function:\n" + "10mA =< dI =<
10000mA; 1ms=< dT =< 10000ms\n" +

                "\n" +
                "For SinCos Functions:\n" + "50mA =< dI =<
10000mA; 20ms=< dT =< 10000ms\n" +

                "\n" +

```

```

        "For Ramp Functions:\n" + "50mA =< dI =<
10000mA; 5ms=< dT =< 10000ms\n" +
        "\n" +
        "For 1. and 2. Order System Functions:\n" +
"50mA =< dI =< 10000mA; 10ms=< dT =< 10000ms\n",
        "WARNING",
        MessageBoxButtons.OK,
MessageBoxIcon.Warning);
        textBox45.Text = "1000";
    }
}
}
catch
{
    MessageBox.Show("<delta I> and <delta T> should satisfy ranges
of:\n" +
        "\n" +
        "For Step Function:\n" + "10mA =< dI =<
10000mA; 1ms=< dT =< 10000ms\n" +
        "\n" +
        "For SinCos Functions:\n" + "50mA =< dI =<
10000mA; 20ms=< dT =< 10000ms\n" +
        "\n" +
        "For Ramp Functions:\n" + "50mA =< dI =<
10000mA; 5ms=< dT =< 10000ms\n" +
        "\n" +
        "For 1. and 2. Order System Functions:\n" +
"50mA =< dI =< 10000mA; 10ms=< dT =< 10000ms\n",
        "WARNING",
        MessageBoxButtons.OK,
MessageBoxIcon.Warning);
        textBox45.Text = "1000";
    }
}
private void textBox56_Leave(object sender, EventArgs e)
{
    double aim;
    Int32 dIlmax;
    Int32 dIlmin;
    dIlmax = 9999;
    dIlmin = 0;
    aim = 0;
}

```

```

if (radioButton57.Checked == true) dlt_lmin = 10; else dlt_lmin = 50;

try
{
    akim = Convert.ToDouble(textBox56.Text);

    if ((Math.Abs(akim) > dlt_lmax) | (Math.Abs(akim) < dlt_lmin))
    {
        if (akim != 0)
        {
            MessageBox.Show("<delta I> and <delta T> should satisfy
ranges of:\n" +

                                "\n" +
                                "For Step Function:\n" + "10mA =< dI =<
10000mA; 1ms=< dT =< 10000ms\n" +

                                "\n" +
                                "For SinCos Functions:\n" + "50mA =< dI =<
10000mA; 20ms=< dT =< 10000ms\n" +

                                "\n" +
                                "For Ramp Functions:\n" + "50mA =< dI =<
10000mA; 5ms=< dT =< 10000ms\n" +

                                "\n" +
                                "For 1. and 2. Order System Functions:\n" +
"50mA =< dI =< 10000mA; 10ms=< dT =< 10000ms\n",

                                "WARNING",
                                MessageBoxButtons.OK,
                                MessageBoxIcon.Warning);

            textBox56.Text = "1000";
        }
    }
}
catch
{
    MessageBox.Show("<delta I> and <delta T> should satisfy ranges
of:\n" +

                                "\n" +
                                "For Step Function:\n" + "10mA =< dI =<
10000mA; 1ms=< dT =< 10000ms\n" +

                                "\n" +
                                "For SinCos Functions:\n" + "50mA =< dI =<
10000mA; 20ms=< dT =< 10000ms\n" +

                                "\n" +
                                "For Ramp Functions:\n" + "50mA =< dI =<
10000mA; 5ms=< dT =< 10000ms\n" +

                                "\n" +

```

```

        "For 1. and 2. Order System Functions:\n" +
"50mA =< dI =< 10000mA; 10ms=< dT =< 10000ms\n",

        "WARNING",
        MessageBoxButtons.OK,
MessageBoxIcon.Warning);

        textBox56.Text = "1000";
    }
}

private void textBox67_Leave(object sender, EventArgs e)
{
    double akim;

    Int32 dIlmax;
    Int32 dIlmin;

    dIlmax = 9999;
    dIlmin = 0;
    akim = 0;

    if (radioButton67.Checked == true) dIlmin = 10; else dIlmin = 50;

    try
    {
        akim = Convert.ToDouble(textBox67.Text);

        if ((Math.Abs(akim) > dIlmax) | (Math.Abs(akim) < dIlmin))
        {
            if (akim != 0)
            {
                MessageBox.Show("<delta I> and <delta T> should satisfy
ranges of:\n" +

                    "\n" +
                    "For Step Function:\n" + "10mA =< dI =<
10000mA; 1ms=< dT =< 10000ms\n" +

                    "\n" +
                    "For SinCos Functions:\n" + "50mA =< dI =<
10000mA; 20ms=< dT =< 10000ms\n" +

                    "\n" +
                    "For Ramp Functions:\n" + "50mA =< dI =<
10000mA; 5ms=< dT =< 10000ms\n" +

                    "\n" +
                    "For 1. and 2. Order System Functions:\n" +
"50mA =< dI =< 10000mA; 10ms=< dT =< 10000ms\n",

                    "WARNING",
                    MessageBoxButtons.OK,
MessageBoxIcon.Warning);

                textBox67.Text = "1000";
            }
        }
    }
}

```

```

        }
    }
    catch
    {
        MessageBox.Show("<delta I> and <delta T> should satisfy ranges
of:\n" +
            "\n" +
            "For Step Function:\n" + "10mA =< dI =<
10000mA; 1ms=< dT =< 10000ms\n" +
            "\n" +
            "For SinCos Functions:\n" + "50mA =< dI =<
10000mA; 20ms=< dT =< 10000ms\n" +
            "\n" +
            "For Ramp Functions:\n" + "50mA =< dI =<
10000mA; 5ms=< dT =< 10000ms\n" +
            "\n" +
            "For 1. and 2. Order System Functions:\n" +
            "50mA =< dI =< 10000mA; 10ms=< dT =< 10000ms\n",
            "WARNING",
            MessageBoxButtons.OK,
            MessageBoxIcon.Warning);
        textBox67.Text = "1000";
    }
}

private void textBox78_Leave(object sender, EventArgs e)
{
    double akim;

    Int32 dIt_lmax;
    Int32 dIt_lmin;

    dIt_lmax = 9999;
    dIt_lmin = 0;
    akim = 0;

    if (radioButton77.Checked == true) dIt_lmin = 10; else dIt_lmin = 50;

    try
    {
        akim = Convert.ToDouble(textBox78.Text);

        if ((Math.Abs(akim) > dIt_lmax) | (Math.Abs(akim) < dIt_lmin))
        {
            if (akim != 0)
            {
                MessageBox.Show("<delta I> and <delta T> should satisfy
ranges of:\n" +

```

```

        "\n" +
        "For Step Function:\n" + "10mA =< dI =<
10000mA; 1ms=< dT =< 10000ms\n" +

        "\n" +
        "For SinCos Functions:\n" + "50mA =< dI =<
10000mA; 20ms=< dT =< 10000ms\n" +

        "\n" +
        "For Ramp Functions:\n" + "50mA =< dI =<
10000mA; 5ms=< dT =< 10000ms\n" +

        "\n" +
        "For 1. and 2. Order System Functions:\n" +
        "50mA =< dI =< 10000mA; 10ms=< dT =< 10000ms\n",

        "WARNING",
        MessageBoxButtons.OK,
MessageBoxIcon.Warning);

        textBox78.Text = "1000";
    }
}
catch
{
    of:\n" +
        "\n" +
        "For Step Function:\n" + "10mA =< dI =<
10000mA; 1ms=< dT =< 10000ms\n" +

        "\n" +
        "For SinCos Functions:\n" + "50mA =< dI =<
10000mA; 20ms=< dT =< 10000ms\n" +

        "\n" +
        "For Ramp Functions:\n" + "50mA =< dI =<
10000mA; 5ms=< dT =< 10000ms\n" +

        "\n" +
        "For 1. and 2. Order System Functions:\n" +
        "50mA =< dI =< 10000mA; 10ms=< dT =< 10000ms\n",

        "WARNING",
        MessageBoxButtons.OK,
MessageBoxIcon.Warning);

        textBox78.Text = "1000";
    }
}

private void textBox89_Leave(object sender, EventArgs e)
{
    double akim;

```

```

Int32 dlt_lmax;
Int32 dlt_lmin;

dlt_lmax = 9999;
dlt_lmin = 0;
akim = 0;

if (radioButton87.Checked == true) dlt_lmin = 10; else dlt_lmin = 50;

try
{
    akim = Convert.ToDouble(textBox89.Text);

    if ((Math.Abs(akim) > dlt_lmax) | (Math.Abs(akim) < dlt_lmin))
    {
        if (akim != 0)
        {
            MessageBox.Show("<delta I> and <delta T> should satisfy
ranges of:\n" +
                "\n" +
                "For Step Function:\n" + "10mA =< dI =<
10000mA; 1ms=< dT =< 10000ms\n" +
                "\n" +
                "For SinCos Functions:\n" + "50mA =< dI =<
10000mA; 20ms=< dT =< 10000ms\n" +
                "\n" +
                "For Ramp Functions:\n" + "50mA =< dI =<
10000mA; 5ms=< dT =< 10000ms\n" +
                "\n" +
                "For 1. and 2. Order System Functions:\n" +
"50mA =< dI =< 10000mA; 10ms=< dT =< 10000ms\n",
                "WARNING",
                MessageBoxButtons.OK,
                MessageBoxIcon.Warning);

            textBox89.Text = "1000";
        }
    }
}
catch
{
    MessageBox.Show("<delta I> and <delta T> should satisfy ranges
of:\n" +
        "\n" +
        "For Step Function:\n" + "10mA =< dI =<
10000mA; 1ms=< dT =< 10000ms\n" +
        "\n" +

```

```

10000mA; 20ms=< dT =< 10000ms\n" +
    "For SinCos Functions:\n" + "50mA =< dI =<
\n" +
10000mA; 5ms=< dT =< 10000ms\n" +
    "For Ramp Functions:\n" + "50mA =< dI =<
\n" +
"50mA =< dI =< 10000mA; 10ms=< dT =< 10000ms\n",
    "WARNING",
    MessageBoxButtons.OK,
MessageBoxIcon.Warning);

    textBox89.Text = "1000";
}
}

private void textBox100_Leave(object sender, EventArgs e)
{
    double akim;

    Int32 dlt_lmax;
    Int32 dlt_lmin;

    dlt_lmax = 9999;
    dlt_lmin = 0;
    akim = 0;

    if (radioButton97.Checked == true) dlt_lmin = 10; else dlt_lmin = 50;

    try
    {
        akim = Convert.ToDouble(textBox100.Text);

        if ((Math.Abs(akim) > dlt_lmax) | (Math.Abs(akim) < dlt_lmin))
        {
            if (akim != 0)
            {
                MessageBox.Show("<delta I> and <delta T> should satisfy
ranges of:\n" +
\n" +
"10000mA; 1ms=< dT =< 10000ms\n" +
    "For Step Function:\n" + "10mA =< dI =<
\n" +
"10000mA; 20ms=< dT =< 10000ms\n" +
    "For SinCos Functions:\n" + "50mA =< dI =<
\n" +
"10000mA; 5ms=< dT =< 10000ms\n" +
    "For Ramp Functions:\n" + "50mA =< dI =<
\n" +

```

```

        "For 1. and 2. Order System Functions:\n" +
"50mA =< dI =< 10000mA; 10ms=< dT =< 10000ms\n",

        "WARNING",
        MessageBoxButtons.OK,
MessageBoxIcon.Warning);

        textBox100.Text = "1000";
    }

}

}
catch
{
    MessageBox.Show("<delta I> and <delta T> should satisfy ranges
of:\n" +

        "\n" +
"For Step Function:\n" + "10mA =< dI =<
10000mA; 1ms=< dT =< 10000ms\n" +

        "\n" +
"For SinCos Functions:\n" + "50mA =< dI =<
10000mA; 20ms=< dT =< 10000ms\n" +

        "\n" +
"For Ramp Functions:\n" + "50mA =< dI =<
10000mA; 5ms=< dT =< 10000ms\n" +

        "\n" +
"For 1. and 2. Order System Functions:\n" +
"50mA =< dI =< 10000mA; 10ms=< dT =< 10000ms\n",

        "WARNING",
        MessageBoxButtons.OK,
MessageBoxIcon.Warning);

        textBox100.Text = "1000";
    }
}

private void textBox111_Leave(object sender, EventArgs e)
{
    double aIim;

    Int32 dIt_Imax;
    Int32 dIt_Imin;

    dIt_Imax = 9999;
    dIt_Imin = 0;
    aIim = 0;

    if (radioButton107.Checked == true) dIt_Imin = 10; else dIt_Imin =
50;

    try

```

```

    {
        akim = Convert.ToDouble(textBox111.Text);

        if ((Math.Abs(akim) > dlt_lmax) | (Math.Abs(akim) < dlt_lmin))
        {
            if (akim != 0)
            {
                MessageBox.Show("<delta I> and <delta T> should satisfy
ranges of:\n" +

                                "\n" +
                                "For Step Function:\n" + "10mA =< dI =<
10000mA; 1ms=< dT =< 10000ms\n" +

                                "\n" +
                                "For SinCos Functions:\n" + "50mA =< dI =<
10000mA; 20ms=< dT =< 10000ms\n" +

                                "\n" +
                                "For Ramp Functions:\n" + "50mA =< dI =<
10000mA; 5ms=< dT =< 10000ms\n" +

                                "\n" +
                                "For 1. and 2. Order System Functions:\n" +
                                "50mA =< dI =< 10000mA; 10ms=< dT =< 10000ms\n",

                                "WARNING",
                                MessageBoxButtons.OK,
                                MessageBoxIcon.Warning);

                textBox111.Text = "1000";
            }
        }
    }
}
catch
{
    MessageBox.Show("<delta I> and <delta T> should satisfy ranges
of:\n" +

                                "\n" +
                                "For Step Function:\n" + "10mA =< dI =<
10000mA; 1ms=< dT =< 10000ms\n" +

                                "\n" +
                                "For SinCos Functions:\n" + "50mA =< dI =<
10000mA; 20ms=< dT =< 10000ms\n" +

                                "\n" +
                                "For Ramp Functions:\n" + "50mA =< dI =<
10000mA; 5ms=< dT =< 10000ms\n" +

                                "\n" +
                                "For 1. and 2. Order System Functions:\n" +
                                "50mA =< dI =< 10000mA; 10ms=< dT =< 10000ms\n",

                                "WARNING",

```

```

        MessageBoxButtons.OK,
MessageBoxIcon.Warning);

        textBox111.Text = "1000";
    }
}

private void textBox122_Leave(object sender, EventArgs e)
{
    double akim;

    Int32 dlt_lmax;
    Int32 dlt_lmin;

    dlt_lmax = 9999;
    dlt_lmin = 0;
    akim = 0;

    if (radioButton117.Checked == true) dlt_lmin = 10; else dlt_lmin =
50;

    try
    {
        akim = Convert.ToDouble(textBox122.Text);

        if ((Math.Abs(akim) > dlt_lmax) | (Math.Abs(akim) < dlt_lmin))
        {
            if (akim != 0)
            {
                MessageBox.Show("<delta I> and <delta T> should satisfy
ranges of:\n" +

                    "\n" +
                    "For Step Function:\n" + "10mA =< dI =<
10000mA; 1ms=< dT =< 10000ms\n" +

                    "\n" +
                    "For SinCos Functions:\n" + "50mA =< dI =<
10000mA; 20ms=< dT =< 10000ms\n" +

                    "\n" +
                    "For Ramp Functions:\n" + "50mA =< dI =<
10000mA; 5ms=< dT =< 10000ms\n" +

                    "\n" +
                    "For 1. and 2. Order System Functions:\n" +
                    "50mA =< dI =< 10000mA; 10ms=< dT =< 10000ms\n",

                    "WARNING",
                    MessageBoxButtons.OK,
MessageBoxIcon.Warning);

                textBox122.Text = "1000";
            }
        }
    }
}

```

```

    }
    catch
    {
        MessageBox.Show("<delta I> and <delta T> should satisfy ranges
of:\n" +
                        "\n" +
                        "For Step Function:\n" + "10mA =< dI =<
10000mA; 1ms=< dT =< 10000ms\n" +
                        "\n" +
                        "For SinCos Functions:\n" + "50mA =< dI =<
10000mA; 20ms=< dT =< 10000ms\n" +
                        "\n" +
                        "For Ramp Functions:\n" + "50mA =< dI =<
10000mA; 5ms=< dT =< 10000ms\n" +
                        "\n" +
                        "For 1. and 2. Order System Functions:\n" +
"50mA =< dI =< 10000mA; 10ms=< dT =< 10000ms\n",
                        "WARNING",
                        MessageBoxButtons.OK,
                        MessageBoxIcon.Warning);
        textBox122.Text = "1000";
    }
}

private void textBox133_Leave(object sender, EventArgs e)
{
    double akim;

    Int32 dIt_lmax;
    Int32 dIt_lmin;

    dIt_lmax = 9999;
    dIt_lmin = 0;
    akim = 0;

    if (radioButton127.Checked == true) dIt_lmin = 10; else dIt_lmin =
50;

    try
    {
        akim = Convert.ToDouble(textBox133.Text);

        if ((Math.Abs(akim) > dIt_lmax) | (Math.Abs(akim) < dIt_lmin))
        {
            if (akim != 0)
            {
                MessageBox.Show("<delta I> and <delta T> should satisfy
ranges of:\n" +
                                "\n" +

```

```

10000mA; 1ms=< dT =< 10000ms\n" +
    "For Step Function:\n" + "10mA =< dI =<
\n" +
10000mA; 20ms=< dT =< 10000ms\n" +
    "For SinCos Functions:\n" + "50mA =< dI =<
\n" +
10000mA; 5ms=< dT =< 10000ms\n" +
    "For Ramp Functions:\n" + "50mA =< dI =<
\n" +
"50mA =< dI =< 10000mA; 10ms=< dT =< 10000ms\n",
    "WARNING",
    MessageBoxButtons.OK,
MessageBoxIcon.Warning);

        textBox133.Text = "1000";
    }
}
catch
{
    MessageBox.Show("<delta I> and <delta T> should satisfy ranges
of:\n" +
\n" +
"For Step Function:\n" + "10mA =< dI =<
\n" +
"For SinCos Functions:\n" + "50mA =< dI =<
\n" +
"For Ramp Functions:\n" + "50mA =< dI =<
\n" +
"For 1. and 2. Order System Functions:\n" +
"50mA =< dI =< 10000mA; 10ms=< dT =< 10000ms\n",
    "WARNING",
    MessageBoxButtons.OK,
MessageBoxIcon.Warning);

        textBox133.Text = "1000";
    }
}

private void textBox144_Leave(object sender, EventArgs e)
{
    double akim;

```

```

Int32 dlt_lmax;
Int32 dlt_lmin;

dlt_lmax = 9999;
dlt_lmin = 0;
akim = 0;

if (radioButton137.Checked == true) dlt_lmin = 10; else dlt_lmin =
50;

try
{
    akim = Convert.ToDouble(textBox144.Text);

    if ((Math.Abs(akim) > dlt_lmax) | (Math.Abs(akim) < dlt_lmin))
    {
        if (akim != 0)
        {
            MessageBox.Show("<delta I> and <delta T> should satisfy
ranges of:\n" +

            "\n" +
            "For Step Function:\n" + "10mA =< dI =<
10000mA; 1ms=< dT =< 10000ms\n" +

            "\n" +
            "For SinCos Functions:\n" + "50mA =< dI =<
10000mA; 20ms=< dT =< 10000ms\n" +

            "\n" +
            "For Ramp Functions:\n" + "50mA =< dI =<
10000mA; 5ms=< dT =< 10000ms\n" +

            "\n" +
            "For 1. and 2. Order System Functions:\n" +
"50mA =< dI =< 10000mA; 10ms=< dT =< 10000ms\n",

            "WARNING",
            MessageBoxButtons.OK,
            MessageBoxIcon.Warning);

            textBox144.Text = "1000";
        }
    }
}
catch
{
    MessageBox.Show("<delta I> and <delta T> should satisfy ranges
of:\n" +

    "\n" +
    "For Step Function:\n" + "10mA =< dI =<
10000mA; 1ms=< dT =< 10000ms\n" +

    "\n" +

```

```

    "For SinCos Functions:\n" + "50mA =< dI =<
10000mA; 20ms=< dT =< 10000ms\n" +
    "\n" +
    "For Ramp Functions:\n" + "50mA =< dI =<
10000mA; 5ms=< dT =< 10000ms\n" +
    "\n" +
    "For 1. and 2. Order System Functions:\n" +
"50mA =< dI =< 10000mA; 10ms=< dT =< 10000ms\n",
    "WARNING",
    MessageBoxButtons.OK,
    MessageBoxIcon.Warning);
        textBox144.Text = "1000";
    }
}

private void textBox155_Leave(object sender, EventArgs e)
{
    double akim;

    Int32 dlt_lmax;
    Int32 dlt_lmin;

    dlt_lmax = 9999;
    dlt_lmin = 0;
    akim = 0;

    if (radioButton147.Checked == true) dlt_lmin = 10; else dlt_lmin =
50;

    try
    {
        akim = Convert.ToDouble(textBox155.Text);

        if ((Math.Abs(akim) > dlt_lmax) | (Math.Abs(akim) < dlt_lmin))
        {
            if (akim != 0)
            {
                MessageBox.Show("<delta I> and <delta T> should satisfy
ranges of:\n" +
                    "\n" +
                    "For Step Function:\n" + "10mA =< dI =<
10000mA; 1ms=< dT =< 10000ms\n" +
                    "\n" +
                    "For SinCos Functions:\n" + "50mA =< dI =<
10000mA; 20ms=< dT =< 10000ms\n" +
                    "\n" +
                    "For Ramp Functions:\n" + "50mA =< dI =<
10000mA; 5ms=< dT =< 10000ms\n" +
                    "\n" +

```

```

        "For 1. and 2. Order System Functions:\n" +
"50mA =< dI =< 10000mA; 10ms=< dT =< 10000ms\n",

        "WARNING",
        MessageBoxButtons.OK,
MessageBoxIcon.Warning);

        textBox155.Text = "1000";
    }

}
catch
{
    MessageBox.Show("<delta I> and <delta T> should satisfy ranges
of:\n" +

        "\n" +
"For Step Function:\n" + "10mA =< dI =<
10000mA; 1ms=< dT =< 10000ms\n" +

        "\n" +
"For SinCos Functions:\n" + "50mA =< dI =<
10000mA; 20ms=< dT =< 10000ms\n" +

        "\n" +
"For Ramp Functions:\n" + "50mA =< dI =<
10000mA; 5ms=< dT =< 10000ms\n" +

        "\n" +
"For 1. and 2. Order System Functions:\n" +
"50mA =< dI =< 10000mA; 10ms=< dT =< 10000ms\n",

        "WARNING",
        MessageBoxButtons.OK,
MessageBoxIcon.Warning);

        textBox155.Text = "1000";
    }
}

private void textBox166_Leave(object sender, EventArgs e)
{
    double aIim;

    Int32 dIt_Imax;
    Int32 dIt_Imin;

    dIt_Imax = 9999;
    dIt_Imin = 0;
    aIim = 0;

    if (radioButton157.Checked == true) dIt_Imin = 10; else dIt_Imin =
50;

    try

```

```

    {
        akim = Convert.ToDouble(textBox166.Text);

        if ((Math.Abs(akim) > dlt_lmax) | (Math.Abs(akim) < dlt_lmin))
        {
            if (akim != 0)
            {
                MessageBox.Show("<delta I> and <delta T> should satisfy
ranges of:\n" +

                    "\n" +
                    "For Step Function:\n" + "10mA =< dI =<
10000mA; 1ms=< dT =< 10000ms\n" +

                    "\n" +
                    "For SinCos Functions:\n" + "50mA =< dI =<
10000mA; 20ms=< dT =< 10000ms\n" +

                    "\n" +
                    "For Ramp Functions:\n" + "50mA =< dI =<
10000mA; 5ms=< dT =< 10000ms\n" +

                    "\n" +
                    "For 1. and 2. Order System Functions:\n" +
                    "50mA =< dI =< 10000mA; 10ms=< dT =< 10000ms\n",

                    "WARNING",
                    MessageBoxButtons.OK,
                    MessageBoxIcon.Warning);

                textBox166.Text = "1000";
            }
        }
    }
}
catch
{
    MessageBox.Show("<delta I> and <delta T> should satisfy ranges
of:\n" +

        "\n" +
        "For Step Function:\n" + "10mA =< dI =<
10000mA; 1ms=< dT =< 10000ms\n" +

        "\n" +
        "For SinCos Functions:\n" + "50mA =< dI =<
10000mA; 20ms=< dT =< 10000ms\n" +

        "\n" +
        "For Ramp Functions:\n" + "50mA =< dI =<
10000mA; 5ms=< dT =< 10000ms\n" +

        "\n" +
        "For 1. and 2. Order System Functions:\n" +
        "50mA =< dI =< 10000mA; 10ms=< dT =< 10000ms\n",

        "WARNING",

```

```

        MessageBoxButtons.OK,
MessageBoxIcon.Warning);

        textBox166.Text = "1000";
    }
}

private void textBox167_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox167.Text);

        if (Math.Abs(sayi) > 10000)
        {
            MessageBox.Show("<offset l> must be in range of -10000mA and
10000mA", "WARNING",
                MessageBoxButtons.OK, MessageBoxIcon.Warning);
            textBox167.Text = "0";
        }
    }
    catch
    {
        MessageBox.Show("<offset l> must be in range of -10000mA and
10000mA", "WARNING",
                MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox167.Text = "0";
    }
}

private void textBox168_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox168.Text);

        if (Math.Abs(sayi) > 10000)
        {
            MessageBox.Show("<offset l> must be in range of -10000mA and
10000mA", "WARNING",
                MessageBoxButtons.OK, MessageBoxIcon.Warning);
            textBox168.Text = "0";
        }
    }
    catch
    {
        MessageBox.Show("<offset l> must be in range of -10000mA and
10000mA", "WARNING",
                MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox168.Text = "0";
    }
}
}

```

```

private void textBox169_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox169.Text);

        if (Math.Abs(sayi) > 10000)
        {
            MessageBox.Show("<offset l> must be in range of -10000mA and
10000mA", "WARNING",
                MessageBoxButtons.OK, MessageBoxIcon.Warning);
            textBox169.Text = "0";
        }
    }
    catch
    {
        MessageBox.Show("<offset l> must be in range of -10000mA and
10000mA", "WARNING",
                MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox169.Text = "0";
    }
}

private void textBox170_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox170.Text);

        if (Math.Abs(sayi) > 10000)
        {
            MessageBox.Show("<offset l> must be in range of -10000mA and
10000mA", "WARNING",
                MessageBoxButtons.OK, MessageBoxIcon.Warning);
            textBox170.Text = "0";
        }
    }
    catch
    {
        MessageBox.Show("<offset l> must be in range of -10000mA and
10000mA", "WARNING",
                MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox170.Text = "0";
    }
}

private void textBox171_Leave(object sender, EventArgs e)
{
    double sayi;

    try

```

```

    {
        sayi = Convert.ToDouble(textBox171.Text);

        if (Math.Abs(sayi) > 10000)
        {
            MessageBox.Show("<offset I> must be in range of -10000mA and
10000mA", "WARNING",
                MessageBoxButtons.OK, MessageBoxIcon.Warning);
            textBox171.Text = "0";
        }
    }
    catch
    {
        MessageBox.Show("<offset I> must be in range of -10000mA and
10000mA", "WARNING",
                MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox171.Text = "0";
    }
}

private void textBox172_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox172.Text);

        if (Math.Abs(sayi) > 10000)
        {
            MessageBox.Show("<offset I> must be in range of -10000mA and
10000mA", "WARNING",
                MessageBoxButtons.OK, MessageBoxIcon.Warning);
            textBox172.Text = "0";
        }
    }
    catch
    {
        MessageBox.Show("<offset I> must be in range of -10000mA and
10000mA", "WARNING",
                MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox172.Text = "0";
    }
}

private void textBox173_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox173.Text);

        if (Math.Abs(sayi) > 10000)
        {

```

```

        MessageBox.Show("<offset I> must be in range of -10000mA and
10000mA", "WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox173.Text = "0";
    }

    }
    catch
    {
        MessageBox.Show("<offset I> must be in range of -10000mA and
10000mA", "WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox173.Text = "0";
    }
}

private void textBox174_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox174.Text);

        if (Math.Abs(sayi) > 10000)
        {
            MessageBox.Show("<offset I> must be in range of -10000mA and
10000mA", "WARNING",
                MessageBoxButtons.OK, MessageBoxIcon.Warning);
            textBox174.Text = "0";
        }

    }
    catch
    {
        MessageBox.Show("<offset I> must be in range of -10000mA and
10000mA", "WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox174.Text = "0";
    }
}

private void textBox175_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox175.Text);

        if (Math.Abs(sayi) > 10000)
        {
            MessageBox.Show("<offset I> must be in range of -10000mA and
10000mA", "WARNING",
                MessageBoxButtons.OK, MessageBoxIcon.Warning);
            textBox175.Text = "0";
        }

    }
}

```

```

    }
    catch
    {
        MessageBox.Show("<offset I> must be in range of -10000mA and
10000mA", "WARNING",
                        MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox175.Text = "0";
    }
}

private void textBox176_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox176.Text);

        if (Math.Abs(sayi) > 10000)
        {
            MessageBox.Show("<offset I> must be in range of -10000mA and
10000mA", "WARNING",
                            MessageBoxButtons.OK, MessageBoxIcon.Warning);
            textBox176.Text = "0";
        }
    }
    catch
    {
        MessageBox.Show("<offset I> must be in range of -10000mA and
10000mA", "WARNING",
                        MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox176.Text = "0";
    }
}

private void textBox177_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox177.Text);

        if (Math.Abs(sayi) > 10000)
        {
            MessageBox.Show("<offset I> must be in range of -10000mA and
10000mA", "WARNING",
                            MessageBoxButtons.OK, MessageBoxIcon.Warning);
            textBox177.Text = "0";
        }
    }
    catch
    {
        MessageBox.Show("<offset I> must be in range of -10000mA and
10000mA", "WARNING",
                        MessageBoxButtons.OK, MessageBoxIcon.Warning);

```

```

        textBox177.Text = "0";
    }
}

private void textBox178_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox178.Text);

        if (Math.Abs(sayi) > 10000)
        {
            MessageBox.Show("<offset l> must be in range of -10000mA and
10000mA", "WARNING",
                MessageBoxButtons.OK, MessageBoxIcon.Warning);
            textBox178.Text = "0";
        }
    }
    catch
    {
        MessageBox.Show("<offset l> must be in range of -10000mA and
10000mA", "WARNING",
                MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox178.Text = "0";
    }
}

private void textBox179_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox179.Text);

        if (Math.Abs(sayi) > 10000)
        {
            MessageBox.Show("<offset l> must be in range of -10000mA and
10000mA", "WARNING",
                MessageBoxButtons.OK, MessageBoxIcon.Warning);
            textBox179.Text = "0";
        }
    }
    catch
    {
        MessageBox.Show("<offset l> must be in range of -10000mA and
10000mA", "WARNING",
                MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox179.Text = "0";
    }
}

private void textBox180_Leave(object sender, EventArgs e)
{

```

```

double sayi;

try
{
    sayi = Convert.ToDouble(textBox180.Text);

    if (Math.Abs(sayi) > 10000)
    {
        MessageBox.Show("<offset l> must be in range of -10000mA and
10000mA", "WARNING",
                        MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox180.Text = "0";
    }
}
catch
{
    MessageBox.Show("<offset l> must be in range of -10000mA and
10000mA", "WARNING",
                    MessageBoxButtons.OK, MessageBoxIcon.Warning);
    textBox180.Text = "0";
}

private void textBox181_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox181.Text);

        if (Math.Abs(sayi) > 10000)
        {
            MessageBox.Show("<offset l> must be in range of -10000mA and
10000mA", "WARNING",
                            MessageBoxButtons.OK, MessageBoxIcon.Warning);
            textBox181.Text = "0";
        }
    }
    catch
    {
        MessageBox.Show("<offset l> must be in range of -10000mA and
10000mA", "WARNING",
                        MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox181.Text = "0";
    }
}

private void textBox3_Leave(object sender, EventArgs e)
{
    Int32 zaman;

    Int32 dlt_Tmax;
    Int32 dlt_Tmin;

    dlt_Tmax = 10000;

```

```

dlt_Tmin = 0;

if (radioButton1.Checked == true) dlt_Tmin = 1;
else

    if (radioButton2.Checked == true) dlt_Tmin = 20;
    else
        if (radioButton3.Checked == true) dlt_Tmin = 20;
        else
            if (radioButton5.Checked == true) dlt_Tmin = 20;
            else
                if (radioButton6.Checked == true) dlt_Tmin = 20;
                else

                    if (radioButton8.Checked == true) dlt_Tmin = 5;

else dlt_Tmin = 10;

try
{
    zaman = Convert.ToInt32(textBox3.Text);

    if ((zaman > dlt_Tmax) | (zaman < dlt_Tmin))
    {
        MessageBox.Show("<delta I> and <delta T> should satisfy
ranges of:\n" +

            "\n" +
            "For Step Function:\n" + "10mA =< dI =<
10000mA; 1ms=< dT =< 10000ms\n" +

            "\n" +
            "For SinCos Functions:\n" + "50mA =< dI =<
10000mA; 20ms=< dT =< 10000ms\n" +

            "\n" +
            "For Ramp Functions:\n" + "50mA =< dI =<
10000mA; 5ms=< dT =< 10000ms\n" +

            "\n" +
            "For 1. and 2. Order System Functions:\n" +
"50mA =< dI =< 10000mA; 10ms=< dT =< 10000ms\n",

            "WARNING",
            MessageBoxButtons.OK,
            MessageBoxIcon.Warning);

        textBox3.Text = "100";
    }
}
catch
{
    MessageBox.Show("<delta I> and <delta T> should satisfy ranges
of:\n" +

            "\n" +

```

```

10000mA; 1ms=< dT =< 10000ms\n" +
    "For Step Functions:\n" + "10mA =< dI =<
\n" +
10000mA; 20ms=< dT =< 10000ms\n" +
    "For SinCos Functions:\n" + "50mA =< dI =<
\n" +
10000mA; 5ms=< dT =< 10000ms\n" +
    "For Ramp Functions:\n" + "50mA =< dI =<
\n" +
"50mA =< dI =< 10000mA; 10ms=< dT =< 10000ms\n",
    "WARNING",
    MessageBoxButtons.OK,
MessageBoxIcon.Warning);

        textBox3.Text = "100";
    }
}

private void textBox22_Leave(object sender, EventArgs e)
{
    Int32 zaman;

    Int32 dlt_Tmax;
    Int32 dlt_Tmin;

    dlt_Tmax = 10000;
    dlt_Tmin = 0;

    if (radioButton27.Checked == true) dlt_Tmin = 1;
    else

        if (radioButton26.Checked == true) dlt_Tmin = 20;
        else
            if (radioButton25.Checked == true) dlt_Tmin = 20;
            else
                if (radioButton24.Checked == true) dlt_Tmin = 20;
                else
                    if (radioButton23.Checked == true) dlt_Tmin = 20;
                    else

                        if (radioButton22.Checked == true) dlt_Tmin = 5;

    else dlt_Tmin = 10;

    try
    {
        zaman = Convert.ToInt32(textBox22.Text);

        if ((zaman > dlt_Tmax) | (zaman < dlt_Tmin))
        {
            MessageBox.Show("<delta I> and <delta T> should satisfy
ranges of:\n" +

```

```

        "\n" +
        "For Step Function:\n" + "10mA =< dI =<
10000mA; 1ms=< dT =< 10000ms\n" +

        "\n" +
        "For SinCos Functions:\n" + "50mA =< dI =<
10000mA; 20ms=< dT =< 10000ms\n" +

        "\n" +
        "For Ramp Functions:\n" + "50mA =< dI =<
10000mA; 5ms=< dT =< 10000ms\n" +

        "\n" +
        "For 1. and 2. Order System Functions:\n" +
"50mA =< dI =< 10000mA; 10ms=< dT =< 10000ms\n",

        "WARNING",
        MessageBoxButtons.OK,
MessageBoxIcon.Warning);

        textBox22.Text = "100";
    }
}
catch
{
    MessageBox.Show("<delta I> and <delta T> should satisfy ranges
of:\n" +

        "\n" +
        "For Step Function:\n" + "10mA =< dI =<
10000mA; 1ms=< dT =< 10000ms\n" +

        "\n" +
        "For SinCos Functions:\n" + "50mA =< dI =<
10000mA; 20ms=< dT =< 10000ms\n" +

        "\n" +
        "For Ramp Functions:\n" + "50mA =< dI =<
10000mA; 5ms=< dT =< 10000ms\n" +

        "\n" +
        "For 1. and 2. Order System Functions:\n" +
"50mA =< dI =< 10000mA; 10ms=< dT =< 10000ms\n",

        "WARNING",
        MessageBoxButtons.OK,
MessageBoxIcon.Warning);

        textBox22.Text = "100";
    }
}

private void textBox33_Leave(object sender, EventArgs e)
{
    Int32 zaman;

    Int32 dIt_Tmax;

```

```

Int32 dlt_Tmin;

dlt_Tmax = 10000;
dlt_Tmin = 0;

if (radioButton37.Checked == true) dlt_Tmin = 1;
else

    if (radioButton36.Checked == true) dlt_Tmin = 20;
    else
        if (radioButton35.Checked == true) dlt_Tmin = 20;
        else
            if (radioButton34.Checked == true) dlt_Tmin = 20;
            else
                if (radioButton33.Checked == true) dlt_Tmin = 20;
                else

                    if (radioButton32.Checked == true) dlt_Tmin = 5;

else dlt_Tmin = 10;

try
{
    zaman = Convert.ToInt32(textBox33.Text);

    if ((zaman > dlt_Tmax) | (zaman < dlt_Tmin))
    {
        MessageBox.Show("<delta I> and <delta T> should satisfy
ranges of:\n" +

            "\n" +
            "For Step Functions:\n" + "10mA =< dI =<
10000mA; 1ms=< dT =< 10000ms\n" +

            "\n" +
            "For SinCos Functions:\n" + "50mA =< dI =<
10000mA; 20ms=< dT =< 10000ms\n" +

            "\n" +
            "For Ramp Functions:\n" + "50mA =< dI =<
10000mA; 5ms=< dT =< 10000ms\n" +

            "\n" +
            "For 1. and 2. Order System Functions:\n" +
"50mA =< dI =< 10000mA; 10ms=< dT =< 10000ms\n",

            "WARNING",
            MessageBoxButtons.OK,
            MessageBoxIcon.Warning);

        textBox33.Text = "100";
    }
}
catch
{
    MessageBox.Show("<delta I> and <delta T> should satisfy ranges
of:\n" +

```

```

        "\n" +
        "For Step Functions:\n" + "10mA =< dI =<
10000mA; 1ms=< dT =< 10000ms\n" +

        "\n" +
        "For SinCos Functions:\n" + "50mA =< dI =<
10000mA; 20ms=< dT =< 10000ms\n" +

        "\n" +
        "For Ramp Functions:\n" + "50mA =< dI =<
10000mA; 5ms=< dT =< 10000ms\n" +

        "\n" +
        "For 1. and 2. Order System Functions:\n" +
        "50mA =< dI =< 10000mA; 10ms=< dT =< 10000ms\n",

        "WARNING",
        MessageBoxButtons.OK,
        MessageBoxIcon.Warning);

        textBox33.Text = "100";
    }
}

private void textBox44_Leave(object sender, EventArgs e)
{
    Int32 zaman;

    Int32 dIt_Tmax;
    Int32 dIt_Tmin;

    dIt_Tmax = 10000;
    dIt_Tmin = 0;

    if (radioButton47.Checked == true) dIt_Tmin = 1;
    else

        if (radioButton46.Checked == true) dIt_Tmin = 20;
        else
            if (radioButton45.Checked == true) dIt_Tmin = 20;
            else
                if (radioButton44.Checked == true) dIt_Tmin = 20;
                else
                    if (radioButton43.Checked == true) dIt_Tmin = 20;
                    else

                        if (radioButton42.Checked == true) dIt_Tmin = 5;

    else dIt_Tmin = 10;

    try
    {
        zaman = Convert.ToInt32(textBox44.Text);

        if ((zaman > dIt_Tmax) | (zaman < dIt_Tmin))
        {

```

```

        MessageBox.Show("<delta I> and <delta T> should satisfy
ranges of:\n" +
        "\n" +
        "For Step Function:\n" + "10mA =< dI =<
10000mA; 1ms=< dT =< 10000ms\n" +
        "\n" +
        "For SinCos Functions:\n" + "50mA =< dI =<
10000mA; 20ms=< dT =< 10000ms\n" +
        "\n" +
        "For Ramp Functions:\n" + "50mA =< dI =<
10000mA; 5ms=< dT =< 10000ms\n" +
        "\n" +
        "For 1. and 2. Order System Functions:\n" +
"50mA =< dI =< 10000mA; 10ms=< dT =< 10000ms\n",
        "WARNING",
        MessageBoxButtons.OK,
        MessageBoxIcon.Warning);
        textBox44.Text = "100";
    }
}
catch
{
    MessageBox.Show("<delta I> and <delta T> should satisfy ranges
of:\n" +
        "\n" +
        "For Step Function:\n" + "10mA =< dI =<
10000mA; 1ms=< dT =< 10000ms\n" +
        "\n" +
        "For SinCos Functions:\n" + "50mA =< dI =<
10000mA; 20ms=< dT =< 10000ms\n" +
        "\n" +
        "For Ramp Functions:\n" + "50mA =< dI =<
10000mA; 5ms=< dT =< 10000ms\n" +
        "\n" +
        "For 1. and 2. Order System Functions:\n" +
"50mA =< dI =< 10000mA; 10ms=< dT =< 10000ms\n",
        "WARNING",
        MessageBoxButtons.OK,
        MessageBoxIcon.Warning);
        textBox44.Text = "100";
    }
}

private void textBox55_Leave(object sender, EventArgs e)
{

```

```

Int32 zaman;

Int32 dlt_Tmax;
Int32 dlt_Tmin;

dlt_Tmax = 10000;
dlt_Tmin = 0;

if (radioButton57.Checked == true) dlt_Tmin = 1;
else

    if (radioButton56.Checked == true) dlt_Tmin = 20;
    else
        if (radioButton55.Checked == true) dlt_Tmin = 20;
        else
            if (radioButton54.Checked == true) dlt_Tmin = 20;
            else
                if (radioButton53.Checked == true) dlt_Tmin = 20;
                else
                    if (radioButton52.Checked == true) dlt_Tmin = 5;

else dlt_Tmin = 10;

try
{
    zaman = Convert.ToInt32(textBox55.Text);

    if ((zaman > dlt_Tmax) | (zaman < dlt_Tmin))
    {
        MessageBox.Show("<delta I> and <delta T> should satisfy
ranges of:\n" +

            "\n" +
            "For Step Function:\n" + "10mA =< dI =<
10000mA; 1ms=< dT =< 10000ms\n" +

            "\n" +
            "For SinCos Functions:\n" + "50mA =< dI =<
10000mA; 20ms=< dT =< 10000ms\n" +

            "\n" +
            "For Ramp Functions:\n" + "50mA =< dI =<
10000mA; 5ms=< dT =< 10000ms\n" +

            "\n" +
            "For 1. and 2. Order System Functions:\n" +
            "50mA =< dI =< 10000mA; 10ms=< dT =< 10000ms\n",

            "WARNING",
            MessageBoxButtons.OK,
            MessageBoxIcon.Warning);

        textBox55.Text = "100";
    }
}
catch

```

```

        {
            MessageBox.Show("<delta I> and <delta T> should satisfy ranges
of:\n" +

                                "\n" +
                                "For Step Function:\n" + "10mA =< dI =<
10000mA; 1ms=< dT =< 10000ms\n" +

                                "\n" +
                                "For SinCos Functions:\n" + "50mA =< dI =<
10000mA; 20ms=< dT =< 10000ms\n" +

                                "\n" +
                                "For Ramp Functions:\n" + "50mA =< dI =<
10000mA; 5ms=< dT =< 10000ms\n" +

                                "\n" +
                                "For 1. and 2. Order System Functions:\n" +
"50mA =< dI =< 10000mA; 10ms=< dT =< 10000ms\n",

                                "WARNING",
                                MessageBoxButtons.OK,
                                MessageBoxIcon.Warning);

            textBox55.Text = "100";
        }
    }

    private void textBox66_Leave(object sender, EventArgs e)
    {
        Int32 zaman;

        Int32 dIt_Tmax;
        Int32 dIt_Tmin;

        dIt_Tmax = 10000;
        dIt_Tmin = 0;

        if (radioButton67.Checked == true) dIt_Tmin = 1;
        else

            if (radioButton66.Checked == true) dIt_Tmin = 20;
            else
                if (radioButton65.Checked == true) dIt_Tmin = 20;
                else
                    if (radioButton64.Checked == true) dIt_Tmin = 20;
                    else
                        if (radioButton63.Checked == true) dIt_Tmin = 20;
                        else

                            if (radioButton62.Checked == true) dIt_Tmin = 5;

        else dIt_Tmin = 10;

        try
        {
            zaman = Convert.ToInt32(textBox66.Text);

```

```

        if ((zaman > dlt_Tmax) | (zaman < dlt_Tmin))
        {
            MessageBox.Show("<delta I> and <delta T> should satisfy
ranges of:\n" +

                "\n" +
                "For Step Function:\n" + "10mA =< dI =<
10000mA; 1ms=< dT =< 10000ms\n" +

                "\n" +
                "For SinCos Functions:\n" + "50mA =< dI =<
10000mA; 20ms=< dT =< 10000ms\n" +

                "\n" +
                "For Ramp Functions:\n" + "50mA =< dI =<
10000mA; 5ms=< dT =< 10000ms\n" +

                "\n" +
                "For 1. and 2. Order System Functions:\n" +
"50mA =< dI =< 10000mA; 10ms=< dT =< 10000ms\n",

                "WARNING",
                MessageBoxButtons.OK,
                MessageBoxIcon.Warning);

            textBox66.Text = "100";
        }
    }
    catch
    {
        MessageBox.Show("<delta I> and <delta T> should satisfy ranges
of:\n" +

                "\n" +
                "For Step Function:\n" + "10mA =< dI =<
10000mA; 1ms=< dT =< 10000ms\n" +

                "\n" +
                "For SinCos Functions:\n" + "50mA =< dI =<
10000mA; 20ms=< dT =< 10000ms\n" +

                "\n" +
                "For Ramp Functions:\n" + "50mA =< dI =<
10000mA; 5ms=< dT =< 10000ms\n" +

                "\n" +
                "For 1. and 2. Order System Functions:\n" +
"50mA =< dI =< 10000mA; 10ms=< dT =< 10000ms\n",

                "WARNING",
                MessageBoxButtons.OK,
                MessageBoxIcon.Warning);

            textBox66.Text = "100";
        }
    }
}
}

```

```

private void textBox77_Leave(object sender, EventArgs e)
{
    Int32 zaman;

    Int32 dlt_Tmax;
    Int32 dlt_Tmin;

    dlt_Tmax = 10000;
    dlt_Tmin = 0;

    if (radioButton77.Checked == true) dlt_Tmin = 1;
    else

        if (radioButton76.Checked == true) dlt_Tmin = 20;
        else
            if (radioButton75.Checked == true) dlt_Tmin = 20;
            else
                if (radioButton74.Checked == true) dlt_Tmin = 20;
                else
                    if (radioButton73.Checked == true) dlt_Tmin = 20;
                    else
                        if (radioButton72.Checked == true) dlt_Tmin = 5;
else dlt_Tmin = 10;

    try
    {
        zaman = Convert.ToInt32(textBox77.Text);

        if ((zaman > dlt_Tmax) | (zaman < dlt_Tmin))
        {
            MessageBox.Show("<delta I> and <delta T> should satisfy
ranges of:\n" +

                "\n" +
                "For Step Functions:\n" + "10mA =< dI =<
10000mA; 1ms=< dT =< 10000ms\n" +

                "\n" +
                "For SinCos Functions:\n" + "50mA =< dI =<
10000mA; 20ms=< dT =< 10000ms\n" +

                "\n" +
                "For Ramp Functions:\n" + "50mA =< dI =<
10000mA; 5ms=< dT =< 10000ms\n" +

                "\n" +
                "For 1. and 2. Order System Functions:\n" +
                "50mA =< dI =< 10000mA; 10ms=< dT =< 10000ms\n",

                "WARNING",
                MessageBoxButtons.OK,
                MessageBoxIcon.Warning);

            textBox77.Text = "100";
        }
    }
}

```

```

    }
    catch
    {
        MessageBox.Show("<delta I> and <delta T> should satisfy ranges
of:\n" +

                                "\n" +
                                "For Step Function:\n" + "10mA =< dI =<
10000mA; 1ms=< dT =< 10000ms\n" +

                                "\n" +
                                "For SinCos Functions:\n" + "50mA =< dI =<
10000mA; 20ms=< dT =< 10000ms\n" +

                                "\n" +
                                "For Ramp Functions:\n" + "50mA =< dI =<
10000mA; 5ms=< dT =< 10000ms\n" +

                                "\n" +
                                "For 1. and 2. Order System Functions:\n" +
"50mA =< dI =< 10000mA; 10ms=< dT =< 10000ms\n",

                                "WARNING",
                                MessageBoxButtons.OK,
                                MessageBoxIcon.Warning);

        textBox77.Text = "100";
    }
}

private void textBox88_Leave(object sender, EventArgs e)
{
    Int32 zaman;

    Int32 dIt_Tmax;
    Int32 dIt_Tmin;

    dIt_Tmax = 10000;
    dIt_Tmin = 0;

    if (radioButton87.Checked == true) dIt_Tmin = 1;
    else

        if (radioButton86.Checked == true) dIt_Tmin = 20;
        else
            if (radioButton85.Checked == true) dIt_Tmin = 20;
            else
                if (radioButton84.Checked == true) dIt_Tmin = 20;
                else
                    if (radioButton83.Checked == true) dIt_Tmin = 20;
                    else

                        if (radioButton82.Checked == true) dIt_Tmin = 5;

    else dIt_Tmin = 10;

    try
    {

```

```

zaman = Convert.ToInt32(textBox88.Text);

if ((zaman > dlt_Tmax) | (zaman < dlt_Tmin))
{
    MessageBox.Show("<delta I> and <delta T> should satisfy
ranges of:\n" +

        "\n" +
        "For Step Function:\n" + "10mA =< dI =<
10000mA; 1ms=< dT =< 10000ms\n" +

        "\n" +
        "For SinCos Functions:\n" + "50mA =< dI =<
10000mA; 20ms=< dT =< 10000ms\n" +

        "\n" +
        "For Ramp Functions:\n" + "50mA =< dI =<
10000mA; 5ms=< dT =< 10000ms\n" +

        "\n" +
        "For 1. and 2. Order System Functions:\n" +
"50mA =< dI =< 10000mA; 10ms=< dT =< 10000ms\n",

        "WARNING",
        MessageBoxButtons.OK,
        MessageBoxIcon.Warning);

    textBox88.Text = "100";
}
}
catch
{
    MessageBox.Show("<delta I> and <delta T> should satisfy ranges
of:\n" +

        "\n" +
        "For Step Function:\n" + "10mA =< dI =<
10000mA; 1ms=< dT =< 10000ms\n" +

        "\n" +
        "For SinCos Functions:\n" + "50mA =< dI =<
10000mA; 20ms=< dT =< 10000ms\n" +

        "\n" +
        "For Ramp Functions:\n" + "50mA =< dI =<
10000mA; 5ms=< dT =< 10000ms\n" +

        "\n" +
        "For 1. and 2. Order System Functions:\n" +
"50mA =< dI =< 10000mA; 10ms=< dT =< 10000ms\n",

        "WARNING",
        MessageBoxButtons.OK,
        MessageBoxIcon.Warning);

    textBox88.Text = "100";
}
}

```

```

}

private void textBox99_Leave(object sender, EventArgs e)
{
    Int32 zaman;

    Int32 dlt_Tmax;
    Int32 dlt_Tmin;

    dlt_Tmax = 10000;
    dlt_Tmin = 0;

    if (radioButton97.Checked == true) dlt_Tmin = 1;
    else

        if (radioButton96.Checked == true) dlt_Tmin = 20;
        else
            if (radioButton95.Checked == true) dlt_Tmin = 20;
            else
                if (radioButton94.Checked == true) dlt_Tmin = 20;
                else
                    if (radioButton93.Checked == true) dlt_Tmin = 20;
                    else
                        if (radioButton92.Checked == true) dlt_Tmin = 5;

else dlt_Tmin = 10;

    try
    {
        zaman = Convert.ToInt32(textBox99.Text);

        if ((zaman > dlt_Tmax) | (zaman < dlt_Tmin))
        {
            MessageBox.Show("<delta I> and <delta T> should satisfy
ranges of:\n" +

                "\n" +
                "For Step Function:\n" + "10mA =< dI =<
10000mA; 1ms=< dT =< 10000ms\n" +

                "\n" +
                "For SinCos Functions:\n" + "50mA =< dI =<
10000mA; 20ms=< dT =< 10000ms\n" +

                "\n" +
                "For Ramp Functions:\n" + "50mA =< dI =<
10000mA; 5ms=< dT =< 10000ms\n" +

                "\n" +
                "For 1. and 2. Order System Functions:\n" +
                "50mA =< dI =< 10000mA; 10ms=< dT =< 10000ms\n",

                "WARNING",
                MessageBoxButtons.OK,
                MessageBoxIcon.Warning);

            textBox99.Text = "100";

```

```

        }
    }
    catch
    {
        MessageBox.Show("<delta I> and <delta T> should satisfy ranges
of:\n" +
                        "\n" +
                        "For Step Function:\n" + "10mA =< dI =<
10000mA; 1ms=< dT =< 10000ms\n" +
                        "\n" +
                        "For SinCos Functions:\n" + "50mA =< dI =<
10000mA; 20ms=< dT =< 10000ms\n" +
                        "\n" +
                        "For Ramp Functions:\n" + "50mA =< dI =<
10000mA; 5ms=< dT =< 10000ms\n" +
                        "\n" +
                        "For 1. and 2. Order System Functions:\n" +
"50mA =< dI =< 10000mA; 10ms=< dT =< 10000ms\n",
                        "WARNING",
                        MessageBoxButtons.OK,
                        MessageBoxIcon.Warning);
        textBox99.Text = "100";
    }
}

private void textBox110_Leave(object sender, EventArgs e)
{
    Int32 zaman;

    Int32 dIt_Tmax;
    Int32 dIt_Tmin;

    dIt_Tmax = 10000;
    dIt_Tmin = 0;

    if (radioButton107.Checked == true) dIt_Tmin = 1;
    else

        if (radioButton106.Checked == true) dIt_Tmin = 20;
        else
            if (radioButton105.Checked == true) dIt_Tmin = 20;
            else
                if (radioButton104.Checked == true) dIt_Tmin = 20;
                else
                    if (radioButton103.Checked == true) dIt_Tmin = 20;
                    else

                        if (radioButton102.Checked == true) dIt_Tmin = 5;

    else dIt_Tmin = 10;
}

```

```

try
{
    zaman = Convert.ToInt32(textBox110.Text);

    if ((zaman > dlt_Tmax) | (zaman < dlt_Tmin))
    {
        MessageBox.Show("<delta I> and <delta T> should satisfy
ranges of:\n" +

            "\n" +
            "For Step Function:\n" + "10mA =< dI =<
10000mA; 1ms=< dT =< 10000ms\n" +

            "\n" +
            "For SinCos Functions:\n" + "50mA =< dI =<
10000mA; 20ms=< dT =< 10000ms\n" +

            "\n" +
            "For Ramp Functions:\n" + "50mA =< dI =<
10000mA; 5ms=< dT =< 10000ms\n" +

            "\n" +
            "For 1. and 2. Order System Functions:\n" +
"50mA =< dI =< 10000mA; 10ms=< dT =< 10000ms\n",

            "WARNING",
            MessageBoxButtons.OK,
            MessageBoxIcon.Warning);

        textBox110.Text = "100";
    }
}
catch
{
    MessageBox.Show("<delta I> and <delta T> should satisfy ranges
of:\n" +

            "\n" +
            "For Step Function:\n" + "10mA =< dI =<
10000mA; 1ms=< dT =< 10000ms\n" +

            "\n" +
            "For SinCos Functions:\n" + "50mA =< dI =<
10000mA; 20ms=< dT =< 10000ms\n" +

            "\n" +
            "For Ramp Functions:\n" + "50mA =< dI =<
10000mA; 5ms=< dT =< 10000ms\n" +

            "\n" +
            "For 1. and 2. Order System Functions:\n" +
"50mA =< dI =< 10000mA; 10ms=< dT =< 10000ms\n",

            "WARNING",
            MessageBoxButtons.OK,
            MessageBoxIcon.Warning);
}
}

```

```

        textBox110.Text = "100";
    }
}

private void textBox121_Leave(object sender, EventArgs e)
{
    Int32 zaman;

    Int32 dlt_Tmax;
    Int32 dlt_Tmin;

    dlt_Tmax = 10000;
    dlt_Tmin = 0;

    if (radioButton117.Checked == true) dlt_Tmin = 1;
    else

        if (radioButton116.Checked == true) dlt_Tmin = 20;
        else
            if (radioButton115.Checked == true) dlt_Tmin = 20;
            else
                if (radioButton114.Checked == true) dlt_Tmin = 20;
                else
                    if (radioButton113.Checked == true) dlt_Tmin = 20;
                    else
                        if (radioButton112.Checked == true) dlt_Tmin = 5;

else dlt_Tmin = 10;

    try
    {
        zaman = Convert.ToInt32(textBox121.Text);

        if ((zaman > dlt_Tmax) | (zaman < dlt_Tmin))
        {
            MessageBox.Show("<delta I> and <delta T> should satisfy
ranges of:\n" +

                "\n" +
                "For Step Functions:\n" + "10mA =< dI =<
10000mA; 1ms=< dT =< 10000ms\n" +

                "\n" +
                "For SinCos Functions:\n" + "50mA =< dI =<
10000mA; 20ms=< dT =< 10000ms\n" +

                "\n" +
                "For Ramp Functions:\n" + "50mA =< dI =<
10000mA; 5ms=< dT =< 10000ms\n" +

                "\n" +
                "For 1. and 2. Order System Functions:\n" +
                "50mA =< dI =< 10000mA; 10ms=< dT =< 10000ms\n",

            "WARNING",
            MessageBoxButtons.OK,
            MessageBoxIcon.Warning);
        }
    }
}

```

```

        textBox121.Text = "100";
    }
}
catch
{
    MessageBox.Show("<delta I> and <delta T> should satisfy ranges
of:\n" +
        "\n" +
        "For Step Function:\n" + "10mA =< dI =<
10000mA; 1ms=< dT =< 10000ms\n" +
        "\n" +
        "For SinCos Functions:\n" + "50mA =< dI =<
10000mA; 20ms=< dT =< 10000ms\n" +
        "\n" +
        "For Ramp Functions:\n" + "50mA =< dI =<
10000mA; 5ms=< dT =< 10000ms\n" +
        "\n" +
        "For 1. and 2. Order System Functions:\n" +
"50mA =< dI =< 10000mA; 10ms=< dT =< 10000ms\n",
        "WARNING",
        MessageBoxButtons.OK,
        MessageBoxIcon.Warning);
    textBox121.Text = "100";
}
}

private void textBox132_Leave(object sender, EventArgs e)
{
    Int32 zaman;

    Int32 dlt_Tmax;
    Int32 dlt_Tmin;

    dlt_Tmax = 10000;
    dlt_Tmin = 0;

    if (radioButton127.Checked == true) dlt_Tmin = 1;
    else

        if (radioButton126.Checked == true) dlt_Tmin = 20;
        else
            if (radioButton125.Checked == true) dlt_Tmin = 20;
            else
                if (radioButton124.Checked == true) dlt_Tmin = 20;
                else
                    if (radioButton123.Checked == true) dlt_Tmin = 20;
                    else

                        if (radioButton122.Checked == true) dlt_Tmin = 5;
                    else dlt_Tmin = 10;
}

```

```

try
{
    zaman = Convert.ToInt32(textBox132.Text);

    if ((zaman > dlt_Tmax) | (zaman < dlt_Tmin))
    {
        MessageBox.Show("<delta I> and <delta T> should satisfy
ranges of:\n" +

            "\n" +
            "For Step Function:\n" + "10mA =< dI =<
10000mA; 1ms=< dT =< 10000ms\n" +

            "\n" +
            "For SinCos Functions:\n" + "50mA =< dI =<
10000mA; 20ms=< dT =< 10000ms\n" +

            "\n" +
            "For Ramp Functions:\n" + "50mA =< dI =<
10000mA; 5ms=< dT =< 10000ms\n" +

            "\n" +
            "For 1. and 2. Order System Functions:\n" +
"50mA =< dI =< 10000mA; 10ms=< dT =< 10000ms\n",

            "WARNING",
            MessageBoxButtons.OK,
            MessageBoxIcon.Warning);

        textBox132.Text = "100";
    }
}
catch
{
    MessageBox.Show("<delta I> and <delta T> should satisfy ranges
of:\n" +

            "\n" +
            "For Step Function:\n" + "10mA =< dI =<
10000mA; 1ms=< dT =< 10000ms\n" +

            "\n" +
            "For SinCos Functions:\n" + "50mA =< dI =<
10000mA; 20ms=< dT =< 10000ms\n" +

            "\n" +
            "For Ramp Functions:\n" + "50mA =< dI =<
10000mA; 5ms=< dT =< 10000ms\n" +

            "\n" +
            "For 1. and 2. Order System Functions:\n" +
"50mA =< dI =< 10000mA; 10ms=< dT =< 10000ms\n",

            "WARNING",

```

```

        MessageBoxButtons.OK,
        MessageBoxIcon.Warning);

        textBox132.Text = "100";
    }
}

private void textBox143_Leave(object sender, EventArgs e)
{
    Int32 zaman;

    Int32 dlt_Tmax;
    Int32 dlt_Tmin;

    dlt_Tmax = 10000;
    dlt_Tmin = 0;

    if (radioButton137.Checked == true) dlt_Tmin = 1;
    else
        if (radioButton136.Checked == true) dlt_Tmin = 20;
        else
            if (radioButton135.Checked == true) dlt_Tmin = 20;
            else
                if (radioButton134.Checked == true) dlt_Tmin = 20;
                else
                    if (radioButton133.Checked == true) dlt_Tmin = 20;
                    else
                        if (radioButton132.Checked == true) dlt_Tmin = 5;

    else dlt_Tmin = 10;

    try
    {
        zaman = Convert.ToInt32(textBox143.Text);

        if ((zaman > dlt_Tmax) | (zaman < dlt_Tmin))
        {
            MessageBox.Show("<delta I> and <delta T> should satisfy
ranges of:\n" +

                "\n" +
                "For Step Function:\n" + "10mA =< dI =<
10000mA; 1ms=< dT =< 10000ms\n" +

                "\n" +
                "For SinCos Functions:\n" + "50mA =< dI =<
10000mA; 20ms=< dT =< 10000ms\n" +

                "\n" +
                "For Ramp Functions:\n" + "50mA =< dI =<
10000mA; 5ms=< dT =< 10000ms\n" +

                "\n" +
                "For 1. and 2. Order System Functions:\n" +
                "50mA =< dI =< 10000mA; 10ms=< dT =< 10000ms\n",

```

```

        "WARNING",
        MessageBoxButtons.OK,
        MessageBoxIcon.Warning);

        textBox143.Text = "100";
    }
}
catch
{
    MessageBox.Show("<delta I> and <delta T> should satisfy ranges
of:\n" +

        "\n" +
        "For Step Functions:\n" + "10mA =< dI =<
10000mA; 1ms=< dT =< 10000ms\n" +

        "\n" +
        "For SinCos Functions:\n" + "50mA =< dI =<
10000mA; 20ms=< dT =< 10000ms\n" +

        "\n" +
        "For Ramp Functions:\n" + "50mA =< dI =<
10000mA; 5ms=< dT =< 10000ms\n" +

        "\n" +
        "For 1. and 2. Order System Functions:\n" +
        "50mA =< dI =< 10000mA; 10ms=< dT =< 10000ms\n",

        "WARNING",
        MessageBoxButtons.OK,
        MessageBoxIcon.Warning);

        textBox143.Text = "100";
    }
}

private void textBox154_Leave(object sender, EventArgs e)
{
    Int32 zaman;

    Int32 dIt_Tmax;
    Int32 dIt_Tmin;

    dIt_Tmax = 10000;
    dIt_Tmin = 0;

    if (radioButton147.Checked == true) dIt_Tmin = 1;
    else

        if (radioButton146.Checked == true) dIt_Tmin = 20;
        else
            if (radioButton145.Checked == true) dIt_Tmin = 20;
            else
                if (radioButton144.Checked == true) dIt_Tmin = 20;
                else
                    if (radioButton143.Checked == true) dIt_Tmin = 20;
                    else

```

```

else dlt_Tmin = 10;

if (radioButton142.Checked == true) dlt_Tmin = 5;

try
{
    zaman = Convert.ToInt32(textBox154.Text);

    if ((zaman > dlt_Tmax) | (zaman < dlt_Tmin))
    {
        MessageBox.Show("<delta I> and <delta T> should satisfy
ranges of:\n" +

            "\n" +
            "For Step Function:\n" + "10mA =< dI =<
10000mA; 1ms=< dT =< 10000ms\n" +

            "\n" +
            "For SinCos Functions:\n" + "50mA =< dI =<
10000mA; 20ms=< dT =< 10000ms\n" +

            "\n" +
            "For Ramp Functions:\n" + "50mA =< dI =<
10000mA; 5ms=< dT =< 10000ms\n" +

            "\n" +
            "For 1. and 2. Order System Functions:\n" +
"50mA =< dI =< 10000mA; 10ms=< dT =< 10000ms\n",

            "WARNING",
            MessageBoxButtons.OK,
            MessageBoxIcon.Warning);

        textBox154.Text = "100";
    }
}
catch
{
    MessageBox.Show("<delta I> and <delta T> should satisfy ranges
of:\n" +

        "\n" +
        "For Step Function:\n" + "10mA =< dI =<
10000mA; 1ms=< dT =< 10000ms\n" +

        "\n" +
        "For SinCos Functions:\n" + "50mA =< dI =<
10000mA; 20ms=< dT =< 10000ms\n" +

        "\n" +
        "For Ramp Functions:\n" + "50mA =< dI =<
10000mA; 5ms=< dT =< 10000ms\n" +

        "\n" +
        "For 1. and 2. Order System Functions:\n" +
"50mA =< dI =< 10000mA; 10ms=< dT =< 10000ms\n",

```

```

        "WARNING",
        MessageBoxButtons.OK,
        MessageBoxIcon.Warning);

        textBox154.Text = "100";
    }
}

private void textBox165_Leave(object sender, EventArgs e)
{
    Int32 zaman;

    Int32 dlt_Tmax;
    Int32 dlt_Tmin;

    dlt_Tmax = 10000;
    dlt_Tmin = 0;

    if (radioButton157.Checked == true) dlt_Tmin = 1;
    else
        if (radioButton156.Checked == true) dlt_Tmin = 20;
        else
            if (radioButton155.Checked == true) dlt_Tmin = 20;
            else
                if (radioButton154.Checked == true) dlt_Tmin = 20;
                else
                    if (radioButton153.Checked == true) dlt_Tmin = 20;
                    else
                        if (radioButton152.Checked == true) dlt_Tmin = 5;

    else dlt_Tmin = 10;

    try
    {
        zaman = Convert.ToInt32(textBox165.Text);

        if ((zaman > dlt_Tmax) | (zaman < dlt_Tmin))
        {
            MessageBox.Show("<delta I> and <delta T> should satisfy
ranges of:\n" +

                "\n" +
                "For Step Function:\n" + "10mA =< dI =<
10000mA; 1ms=< dT =< 10000ms\n" +

                "\n" +
                "For SinCos Functions:\n" + "50mA =< dI =<
10000mA; 20ms=< dT =< 10000ms\n" +

                "\n" +
                "For Ramp Functions:\n" + "50mA =< dI =<
10000mA; 5ms=< dT =< 10000ms\n" +

                "\n" +

```

```

        "For 1. and 2. Order System Functions:\n" +
"50mA =< dI =< 10000mA; 10ms=< dT =< 10000ms\n",

        "WARNING",
        MessageBoxButtons.OK,
MessageBoxIcon.Warning);

        textBox165.Text = "100";
    }

}
catch
{
    MessageBox.Show("<delta I> and <delta T> should satisfy ranges
of:\n" +

        "\n" +
"For Step Function:\n" + "10mA =< dI =<
10000mA; 1ms=< dT =< 10000ms\n" +

        "\n" +
"For SinCos Functions:\n" + "50mA =< dI =<
10000mA; 20ms=< dT =< 10000ms\n" +

        "\n" +
"For Ramp Functions:\n" + "50mA =< dI =<
10000mA; 5ms=< dT =< 10000ms\n" +

        "\n" +
"For 1. and 2. Order System Functions:\n" +
"50mA =< dI =< 10000mA; 10ms=< dT =< 10000ms\n",

        "WARNING",
        MessageBoxButtons.OK,
MessageBoxIcon.Warning);

        textBox165.Text = "100";
    }
}

private void textBox12_Leave(object sender, EventArgs e)
{
    /*
    * if (!char.IsControl(e.KeyChar) && !char.IsDigit(e.KeyChar) ||
(e.KeyChar == '0') )
    {
        e.Handled = true;
    }
    */
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox12.Text);

        if (sayi == 0)
        {

```

```

        MessageBox.Show("<ksi> must be in range of 1 and 9",
"WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox12.Text = "5";
    }
}
catch
{
    MessageBox.Show("<ksi> must be in range of 1 and 9", "WARNING",
        MessageBoxButtons.OK, MessageBoxIcon.Warning);
    textBox12.Text = "5";
}
}

private void textBox21_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox21.Text);

        if (sayi == 0)
        {
            MessageBox.Show("<ksi> must be in range of 1 and 9",
"WARNING",
                MessageBoxButtons.OK, MessageBoxIcon.Warning);
            textBox21.Text = "5";
        }
    }
    catch
    {
        MessageBox.Show("<ksi> must be in range of 1 and 9", "WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox21.Text = "5";
    }
}

private void textBox32_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox32.Text);

        if (sayi == 0)
        {
            MessageBox.Show("<ksi> must be in range of 1 and 9",
"WARNING",
                MessageBoxButtons.OK, MessageBoxIcon.Warning);
            textBox32.Text = "5";
        }
    }
    catch

```

```

    {
        MessageBox.Show("<ksi> must be in range of 1 and 9", "WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox32.Text = "5";
    }
}

private void textBox43_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox43.Text);

        if (sayi == 0)
        {
            MessageBox.Show("<ksi> must be in range of 1 and 9",
                "WARNING",
                MessageBoxButtons.OK, MessageBoxIcon.Warning);
            textBox43.Text = "5";
        }
    }
    catch
    {
        MessageBox.Show("<ksi> must be in range of 1 and 9", "WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox43.Text = "5";
    }
}

private void textBox54_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox54.Text);

        if (sayi == 0)
        {
            MessageBox.Show("<ksi> must be in range of 1 and 9",
                "WARNING",
                MessageBoxButtons.OK, MessageBoxIcon.Warning);
            textBox54.Text = "5";
        }
    }
    catch
    {
        MessageBox.Show("<ksi> must be in range of 1 and 9", "WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox54.Text = "5";
    }
}

private void textBox65_Leave(object sender, EventArgs e)

```

```

{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox65.Text);

        if (sayi == 0)
        {
            "WARNING",
            MessageBox.Show("<ksi> must be in range of 1 and 9",
                MessageBoxButtons.OK, MessageBoxIcon.Warning);
            textBox65.Text = "5";
        }
    }
    catch
    {
        MessageBox.Show("<ksi> must be in range of 1 and 9", "WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox65.Text = "5";
    }
}

private void textBox76_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox76.Text);

        if (sayi == 0)
        {
            "WARNING",
            MessageBox.Show("<ksi> must be in range of 1 and 9",
                MessageBoxButtons.OK, MessageBoxIcon.Warning);
            textBox76.Text = "5";
        }
    }
    catch
    {
        MessageBox.Show("<ksi> must be in range of 1 and 9", "WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox76.Text = "5";
    }
}

private void textBox87_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox87.Text);

        if (sayi == 0)

```

```

        {
            MessageBox.Show("<ksi> must be in range of 1 and 9",
                MessageBoxButtons.OK, MessageBoxIcon.Warning);
            textBox87.Text = "5";
        }
    }
    catch
    {
        MessageBox.Show("<ksi> must be in range of 1 and 9", "WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox87.Text = "5";
    }
}

private void textBox98_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox98.Text);

        if (sayi == 0)
        {
            MessageBox.Show("<ksi> must be in range of 1 and 9",
                MessageBoxButtons.OK, MessageBoxIcon.Warning);
            textBox98.Text = "5";
        }
    }
    catch
    {
        MessageBox.Show("<ksi> must be in range of 1 and 9", "WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox98.Text = "5";
    }
}

private void textBox109_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox109.Text);

        if (sayi == 0)
        {
            MessageBox.Show("<ksi> must be in range of 1 and 9",
                MessageBoxButtons.OK, MessageBoxIcon.Warning);
            textBox109.Text = "5";
        }
    }
}

```

```

        catch
        {
            MessageBox.Show("<ksi> must be in range of 1 and 9", "WARNING",
                MessageBoxButtons.OK, MessageBoxIcon.Warning);
            textBox109.Text = "5";
        }
    }

private void textBox120_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox120.Text);

        if (sayi == 0)
        {
            MessageBox.Show("<ksi> must be in range of 1 and 9",
                "WARNING",
                MessageBoxButtons.OK, MessageBoxIcon.Warning);
            textBox120.Text = "5";
        }
    }
    catch
    {
        MessageBox.Show("<ksi> must be in range of 1 and 9", "WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox120.Text = "5";
    }
}

private void textBox131_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox131.Text);

        if (sayi == 0)
        {
            MessageBox.Show("<ksi> must be in range of 1 and 9",
                "WARNING",
                MessageBoxButtons.OK, MessageBoxIcon.Warning);
            textBox131.Text = "5";
        }
    }
    catch
    {
        MessageBox.Show("<ksi> must be in range of 1 and 9", "WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
            textBox131.Text = "5";
        }
    }
}

```

```

private void textBox142_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox142.Text);

        if (sayi == 0)
        {
            MessageBox.Show("<ksi> must be in range of 1 and 9",
                "WARNING",
                MessageBoxButtons.OK, MessageBoxIcon.Warning);
            textBox142.Text = "5";
        }
    }
    catch
    {
        MessageBox.Show("<ksi> must be in range of 1 and 9", "WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox142.Text = "5";
    }
}

private void textBox153_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox153.Text);

        if (sayi == 0)
        {
            MessageBox.Show("<ksi> must be in range of 1 and 9",
                "WARNING",
                MessageBoxButtons.OK, MessageBoxIcon.Warning);
            textBox153.Text = "5";
        }
    }
    catch
    {
        MessageBox.Show("<ksi> must be in range of 1 and 9", "WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox153.Text = "5";
    }
}

private void textBox164_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox164.Text);

```

```

        if (sayi == 0)
        {
            MessageBox.Show("<ksi> must be in range of 1 and 9",
                "WARNING",
                MessageBoxButtons.OK, MessageBoxIcon.Warning);
            textBox164.Text = "5";
        }
    }
    catch
    {
        MessageBox.Show("<ksi> must be in range of 1 and 9", "WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox164.Text = "5";
    }
}

private void textBox4_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox4.Text);
    }
    catch
    {
        MessageBox.Show("<K> must be in range of -99 and 999", "WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox4.Text = "1";
    }
}

private void textBox20_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox20.Text);
    }
    catch
    {
        MessageBox.Show("<K> must be in range of -99 and 999", "WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox20.Text = "1";
    }
}

private void textBox31_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox31.Text);
    }
    catch

```

```

        {
            MessageBox.Show("<K> must be in range of -99 and 999", "WARNING",
                MessageBoxButtons.OK, MessageBoxIcon.Warning);
            textBox31.Text = "1";
        }
    }

private void textBox42_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox42.Text);
    }
    catch
    {
        MessageBox.Show("<K> must be in range of -99 and 999", "WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox42.Text = "1";
    }
}

private void textBox53_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox53.Text);
    }
    catch
    {
        MessageBox.Show("<K> must be in range of -99 and 999", "WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox53.Text = "1";
    }
}

private void textBox64_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox64.Text);
    }
    catch
    {
        MessageBox.Show("<K> must be in range of -99 and 999", "WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox64.Text = "1";
    }
}

private void textBox75_Leave(object sender, EventArgs e)
{
    double sayi;

```

```

try
{
    sayi = Convert.ToDouble(textBox75.Text);
}
catch
{
    MessageBox.Show("<K> must be in range of -99 and 999", "WARNING",
        MessageBoxButtons.OK, MessageBoxIcon.Warning);
    textBox75.Text = "1";
}
}

```

```

private void textBox86_Leave(object sender, EventArgs e)
{

```

```

    double sayi;

```

```

try
{
    sayi = Convert.ToDouble(textBox86.Text);
}
catch
{
    MessageBox.Show("<K> must be in range of -99 and 999", "WARNING",
        MessageBoxButtons.OK, MessageBoxIcon.Warning);
    textBox86.Text = "1";
}
}

```

```

private void textBox97_Leave(object sender, EventArgs e)
{

```

```

    double sayi;

```

```

try
{
    sayi = Convert.ToDouble(textBox97.Text);
}
catch
{
    MessageBox.Show("<K> must be in range of -99 and 999", "WARNING",
        MessageBoxButtons.OK, MessageBoxIcon.Warning);
    textBox97.Text = "1";
}
}

```

```

private void textBox108_Leave(object sender, EventArgs e)
{

```

```

    double sayi;

```

```

try
{
    sayi = Convert.ToDouble(textBox108.Text);
}
catch
{
    MessageBox.Show("<K> must be in range of -99 and 999", "WARNING",
        MessageBoxButtons.OK, MessageBoxIcon.Warning);
    textBox108.Text = "1";
}
}

```

```

    }
}

private void textBox119_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox119.Text);
    }
    catch
    {
        MessageBox.Show("<K> must be in range of -99 and 999", "WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox119.Text = "1";
    }
}

private void textBox130_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox130.Text);
    }
    catch
    {
        MessageBox.Show("<K> must be in range of -99 and 999", "WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox130.Text = "1";
    }
}

private void textBox141_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox141.Text);
    }
    catch
    {
        MessageBox.Show("<K> must be in range of -99 and 999", "WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox141.Text = "1";
    }
}

private void textBox152_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox152.Text);
    }
}

```

```

    }
    catch
    {
        MessageBox.Show("<K> must be in range of -99 and 999", "WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox152.Text = "1";
    }
}

private void textBox163_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox163.Text);
    }
    catch
    {
        MessageBox.Show("<K> must be in range of -99 and 999", "WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox163.Text = "1";
    }
}

private void textBox6_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox6.Text);
    }
    catch
    {
        MessageBox.Show("<K> must be in range of -99 and 999", "WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox6.Text = "1";
    }
}

private void textBox16_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox20.Text);
    }
    catch
    {
        MessageBox.Show("<K> must be in range of -99 and 999", "WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox20.Text = "1";
    }
}

private void textBox27_Leave(object sender, EventArgs e)

```

```

{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox27.Text);
    }
    catch
    {
        MessageBox.Show("<K> must be in range of -99 and 999", "WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox27.Text = "1";
    }
}

```

```

private void textBox38_Leave(object sender, EventArgs e)
{

```

```

    double sayi;

```

```

    try

```

```

    {

```

```

        sayi = Convert.ToDouble(textBox38.Text);

```

```

    }

```

```

    catch

```

```

    {

```

```

        MessageBox.Show("<K> must be in range of -99 and 999", "WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);

```

```

        textBox38.Text = "1";

```

```

    }

```

```

}

```

```

private void textBox49_Leave(object sender, EventArgs e)
{

```

```

    double sayi;

```

```

    try

```

```

    {

```

```

        sayi = Convert.ToDouble(textBox49.Text);

```

```

    }

```

```

    catch

```

```

    {

```

```

        MessageBox.Show("<K> must be in range of -99 and 999", "WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);

```

```

        textBox49.Text = "1";

```

```

    }

```

```

}

```

```

private void textBox60_Leave(object sender, EventArgs e)
{

```

```

    double sayi;

```

```

    try

```

```

    {

```

```

        sayi = Convert.ToDouble(textBox60.Text);

```

```

    }

```

```

    catch

```

```

    {

```

```

        MessageBox.Show("<K> must be in range of -99 and 999", "WARNING",

```

```

        MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox60.Text = "1";
    }
}

private void textBox71_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox71.Text);
    }
    catch
    {
        MessageBox.Show("<K> must be in range of -99 and 999", "WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox71.Text = "1";
    }
}

private void textBox82_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox82.Text);
    }
    catch
    {
        MessageBox.Show("<K> must be in range of -99 and 999", "WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox82.Text = "1";
    }
}

private void textBox93_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox93.Text);
    }
    catch
    {
        MessageBox.Show("<K> must be in range of -99 and 999", "WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox93.Text = "1";
    }
}

private void textBox104_Leave(object sender, EventArgs e)
{
    double sayi;

    try

```

```

    {
        sayi = Convert.ToDouble(textBox104.Text);
    }
    catch
    {
        MessageBox.Show("<K> must be in range of -99 and 999", "WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox104.Text = "1";
    }
}

private void textBox115_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox115.Text);
    }
    catch
    {
        MessageBox.Show("<K> must be in range of -99 and 999", "WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox115.Text = "1";
    }
}

private void textBox126_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox126.Text);
    }
    catch
    {
        MessageBox.Show("<K> must be in range of -99 and 999", "WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox126.Text = "1";
    }
}

private void textBox137_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox137.Text);
    }
    catch
    {
        MessageBox.Show("<K> must be in range of -99 and 999", "WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox137.Text = "1";
    }
}

```

```

private void textBox148_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox148.Text);
    }
    catch
    {
        MessageBox.Show("<K> must be in range of -99 and 999", "WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox148.Text = "1";
    }
}

```

```

private void textBox159_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox159.Text);
    }
    catch
    {
        MessageBox.Show("<K> must be in range of -99 and 999", "WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox159.Text = "1";
    }
}

```

```

private void textBox8_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox8.Text);
    }
    catch
    {
        MessageBox.Show("<K> must be in range of -99 and 999", "WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox8.Text = "1";
    }
}

```

```

private void textBox19_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox20.Text);
    }
    catch

```

```

        {
            MessageBox.Show("<K> must be in range of -99 and 999", "WARNING",
                MessageBoxButtons.OK, MessageBoxIcon.Warning);
            textBox20.Text = "1";
        }
    }

private void textBox30_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox30.Text);
    }
    catch
    {
        MessageBox.Show("<K> must be in range of -99 and 999", "WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox30.Text = "1";
    }
}

private void textBox41_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox41.Text);
    }
    catch
    {
        MessageBox.Show("<K> must be in range of -99 and 999", "WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox41.Text = "1";
    }
}

private void textBox52_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox52.Text);
    }
    catch
    {
        MessageBox.Show("<K> must be in range of -99 and 999", "WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox52.Text = "1";
    }
}

private void textBox63_Leave(object sender, EventArgs e)
{
    double sayi;

```

```

try
{
    sayi = Convert.ToDouble(textBox63.Text);
}
catch
{
    MessageBox.Show("<K> must be in range of -99 and 999", "WARNING",
        MessageBoxButtons.OK, MessageBoxIcon.Warning);
    textBox63.Text = "1";
}
}

```

```

private void textBox74_Leave(object sender, EventArgs e)
{

```

```

    double sayi;

```

```

try
{
    sayi = Convert.ToDouble(textBox74.Text);
}
catch
{
    MessageBox.Show("<K> must be in range of -99 and 999", "WARNING",
        MessageBoxButtons.OK, MessageBoxIcon.Warning);
    textBox74.Text = "1";
}
}

```

```

private void textBox85_Leave(object sender, EventArgs e)
{

```

```

    double sayi;

```

```

try
{
    sayi = Convert.ToDouble(textBox85.Text);
}
catch
{
    MessageBox.Show("<K> must be in range of -99 and 999", "WARNING",
        MessageBoxButtons.OK, MessageBoxIcon.Warning);
    textBox85.Text = "1";
}
}

```

```

private void textBox96_Leave(object sender, EventArgs e)
{

```

```

    double sayi;

```

```

try
{
    sayi = Convert.ToDouble(textBox96.Text);
}
catch
{
    MessageBox.Show("<K> must be in range of -99 and 999", "WARNING",
        MessageBoxButtons.OK, MessageBoxIcon.Warning);
    textBox96.Text = "1";
}
}

```

```

    }
}

private void textBox107_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox107.Text);
    }
    catch
    {
        MessageBox.Show("<K> must be in range of -99 and 999", "WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox107.Text = "1";
    }
}

private void textBox118_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox118.Text);
    }
    catch
    {
        MessageBox.Show("<K> must be in range of -99 and 999", "WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox118.Text = "1";
    }
}

private void textBox129_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox129.Text);
    }
    catch
    {
        MessageBox.Show("<K> must be in range of -99 and 999", "WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox129.Text = "1";
    }
}

private void textBox140_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox140.Text);
    }
}

```

```

    }
    catch
    {
        MessageBox.Show("<K> must be in range of -99 and 999", "WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox140.Text = "1";
    }
}

private void textBox151_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox151.Text);
    }
    catch
    {
        MessageBox.Show("<K> must be in range of -99 and 999", "WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox151.Text = "1";
    }
}

private void textBox162_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox162.Text);
    }
    catch
    {
        MessageBox.Show("<K> must be in range of -99 and 999", "WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox162.Text = "1";
    }
}

private void textBox10_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox10.Text);
    }
    catch
    {
        MessageBox.Show("<K> must be in range of -99 and 999", "WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox10.Text = "1";
    }
}

private void textBox15_Leave(object sender, EventArgs e)

```

```

{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox15.Text);
    }
    catch
    {
        MessageBox.Show("<K> must be in range of -99 and 999", "WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox15.Text = "1";
    }
}

```

```

private void textBox26_Leave(object sender, EventArgs e)
{

```

```

    double sayi;

```

```

    try

```

```

    {

```

```

        sayi = Convert.ToDouble(textBox26.Text);

```

```

    }

```

```

    catch

```

```

    {

```

```

        MessageBox.Show("<K> must be in range of -99 and 999", "WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);

```

```

        textBox26.Text = "1";

```

```

    }

```

```

}

```

```

private void textBox37_Leave(object sender, EventArgs e)
{

```

```

    double sayi;

```

```

    try

```

```

    {

```

```

        sayi = Convert.ToDouble(textBox37.Text);

```

```

    }

```

```

    catch

```

```

    {

```

```

        MessageBox.Show("<K> must be in range of -99 and 999", "WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);

```

```

        textBox37.Text = "1";

```

```

    }

```

```

}

```

```

private void textBox48_Leave(object sender, EventArgs e)
{

```

```

    double sayi;

```

```

    try

```

```

    {

```

```

        sayi = Convert.ToDouble(textBox48.Text);

```

```

    }

```

```

    catch

```

```

    {

```

```

        MessageBox.Show("<K> must be in range of -99 and 999", "WARNING",

```

```

        MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox48.Text = "1";
    }
}

private void textBox59_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox59.Text);
    }
    catch
    {
        MessageBox.Show("<K> must be in range of -99 and 999", "WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox59.Text = "1";
    }
}

private void textBox70_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox70.Text);
    }
    catch
    {
        MessageBox.Show("<K> must be in range of -99 and 999", "WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox70.Text = "1";
    }
}

private void textBox81_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox81.Text);
    }
    catch
    {
        MessageBox.Show("<K> must be in range of -99 and 999", "WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox81.Text = "1";
    }
}

private void textBox92_Leave(object sender, EventArgs e)
{
    double sayi;

    try

```

```

    {
        sayi = Convert.ToDouble(textBox92.Text);
    }
    catch
    {
        MessageBox.Show("<K> must be in range of -99 and 999", "WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox92.Text = "1";
    }
}

private void textBox103_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox103.Text);
    }
    catch
    {
        MessageBox.Show("<K> must be in range of -99 and 999", "WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox103.Text = "1";
    }
}

private void textBox114_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox114.Text);
    }
    catch
    {
        MessageBox.Show("<K> must be in range of -99 and 999", "WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox114.Text = "1";
    }
}

private void textBox125_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox125.Text);
    }
    catch
    {
        MessageBox.Show("<K> must be in range of -99 and 999", "WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox125.Text = "1";
    }
}

```

```

private void textBox136_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox136.Text);
    }
    catch
    {
        MessageBox.Show("<K> must be in range of -99 and 999", "WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox136.Text = "1";
    }
}

```

```

private void textBox147_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox147.Text);
    }
    catch
    {
        MessageBox.Show("<K> must be in range of -99 and 999", "WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox147.Text = "1";
    }
}

```

```

private void textBox158_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox158.Text);
    }
    catch
    {
        MessageBox.Show("<K> must be in range of -99 and 999", "WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox158.Text = "1";
    }
}

```

```

private void textBox5_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox5.Text);
    }
    catch

```

```

        {
            MessageBox.Show("<l> must be in range of -1000mA and 1000mA",
"WARNING",
                MessageBoxButtons.OK, MessageBoxIcon.Warning);
            textBox5.Text = "0";
        }
    }

private void textBox18_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox18.Text);
    }
    catch
    {
        MessageBox.Show("<l> must be in range of -1000mA and 1000mA",
"WARNING",
                MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox18.Text = "0";
    }
}

private void textBox29_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox29.Text);
    }
    catch
    {
        MessageBox.Show("<l> must be in range of -1000mA and 1000mA",
"WARNING",
                MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox29.Text = "0";
    }
}

private void textBox40_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox40.Text);
    }
    catch
    {
        MessageBox.Show("<l> must be in range of -1000mA and 1000mA",
"WARNING",
                MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox40.Text = "0";
    }
}

```

```

private void textBox51_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox51.Text);
    }
    catch
    {
        MessageBox.Show("<lo> must be in range of -10000mA and 10000mA",
"WARNING",
                    MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox51.Text = "0";
    }
}

private void textBox62_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox62.Text);
    }
    catch
    {
        MessageBox.Show("<lo> must be in range of -10000mA and 10000mA",
"WARNING",
                    MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox62.Text = "0";
    }
}

private void textBox73_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox73.Text);
    }
    catch
    {
        MessageBox.Show("<lo> must be in range of -10000mA and 10000mA",
"WARNING",
                    MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox73.Text = "0";
    }
}

private void textBox84_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {

```

```

        sayi = Convert.ToDouble(textBox84.Text);
    }
    catch
    {
        MessageBox.Show("<lo> must be in range of -10000mA and 10000mA",
"WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox84.Text = "0";
    }
}

private void textBox95_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox95.Text);
    }
    catch
    {
        MessageBox.Show("<lo> must be in range of -10000mA and 10000mA",
"WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox95.Text = "0";
    }
}

private void textBox106_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox106.Text);
    }
    catch
    {
        MessageBox.Show("<lo> must be in range of -10000mA and 10000mA",
"WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox106.Text = "0";
    }
}

private void textBox117_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox117.Text);
    }
    catch
    {
        MessageBox.Show("<lo> must be in range of -10000mA and 10000mA",
"WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);

```

```

        textBox117.Text = "0";
    }
}

private void textBox128_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox128.Text);
    }
    catch
    {
        MessageBox.Show("<l> must be in range of -10000mA and 10000mA",
"WARNING",
                MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox128.Text = "0";
    }
}

private void textBox139_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox139.Text);
    }
    catch
    {
        MessageBox.Show("<l> must be in range of -10000mA and 10000mA",
"WARNING",
                MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox139.Text = "0";
    }
}

private void textBox150_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox150.Text);
    }
    catch
    {
        MessageBox.Show("<l> must be in range of -10000mA and 10000mA",
"WARNING",
                MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox150.Text = "0";
    }
}

private void textBox161_Leave(object sender, EventArgs e)
{
    double sayi;

```

```

        try
        {
            sayi = Convert.ToDouble(textBox161.Text);
        }
        catch
        {
            "WARNING",
            MessageBox.Show("<lo> must be in range of -10000mA and 10000mA",
                MessageBoxButtons.OK, MessageBoxIcon.Warning);
            textBox161.Text = "0";
        }
    }

    private void textBox7_Leave(object sender, EventArgs e)
    {
        double sayi;

        try
        {
            sayi = Convert.ToDouble(textBox7.Text);
        }
        catch
        {
            "WARNING",
            MessageBox.Show("<lo> must be in range of -10000mA and 10000mA",
                MessageBoxButtons.OK, MessageBoxIcon.Warning);
            textBox7.Text = "0";
        }
    }

    private void textBox14_Leave(object sender, EventArgs e)
    {
        double sayi;

        try
        {
            sayi = Convert.ToDouble(textBox14.Text);
        }
        catch
        {
            "WARNING",
            MessageBox.Show("<lo> must be in range of -10000mA and 10000mA",
                MessageBoxButtons.OK, MessageBoxIcon.Warning);
            textBox14.Text = "0";
        }
    }

    private void textBox25_Leave(object sender, EventArgs e)
    {
        double sayi;

        try
        {
            sayi = Convert.ToDouble(textBox25.Text);
        }
        catch
        {

```

```

        MessageBox.Show("<lo> must be in range of -10000mA and 10000mA",
"WARNING",
        MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox25.Text = "0";
    }
}

private void textBox36_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox36.Text);
    }
    catch
    {
        MessageBox.Show("<lo> must be in range of -10000mA and 10000mA",
"WARNING",
        MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox36.Text = "0";
    }
}

private void textBox47_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox47.Text);
    }
    catch
    {
        MessageBox.Show("<lo> must be in range of -10000mA and 10000mA",
"WARNING",
        MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox47.Text = "0";
    }
}

private void textBox58_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox58.Text);
    }
    catch
    {
        MessageBox.Show("<lo> must be in range of -10000mA and 10000mA",
"WARNING",
        MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox58.Text = "0";
    }
}

```

```

private void textBox69_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox69.Text);
    }
    catch
    {
        MessageBox.Show("<lo> must be in range of -10000mA and 10000mA",
"WARNING",
                MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox69.Text = "0";
    }
}

private void textBox80_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox80.Text);
    }
    catch
    {
        MessageBox.Show("<lo> must be in range of -10000mA and 10000mA",
"WARNING",
                MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox80.Text = "0";
    }
}

private void textBox91_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox91.Text);
    }
    catch
    {
        MessageBox.Show("<lo> must be in range of -10000mA and 10000mA",
"WARNING",
                MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox91.Text = "0";
    }
}

private void textBox102_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox102.Text);
    }
}

```

```

    }
    catch
    {
        MessageBox.Show("<lo> must be in range of -10000mA and 10000mA",
"WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox102.Text = "0";
    }
}

private void textBox113_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox113.Text);
    }
    catch
    {
        MessageBox.Show("<lo> must be in range of -10000mA and 10000mA",
"WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox113.Text = "0";
    }
}

private void textBox124_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox124.Text);
    }
    catch
    {
        MessageBox.Show("<lo> must be in range of -10000mA and 10000mA",
"WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox124.Text = "0";
    }
}

private void textBox135_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox135.Text);
    }
    catch
    {
        MessageBox.Show("<lo> must be in range of -10000mA and 10000mA",
"WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox135.Text = "0";
    }
}

```

```

    }
}

private void textBox146_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox146.Text);
    }
    catch
    {
        MessageBox.Show("<l> must be in range of -10000mA and 10000mA",
"WARNING",
                    MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox146.Text = "0";
    }
}

private void textBox157_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox157.Text);
    }
    catch
    {
        MessageBox.Show("<l> must be in range of -10000mA and 10000mA",
"WARNING",
                    MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox157.Text = "0";
    }
}

private void textBox9_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox9.Text);
    }
    catch
    {
        MessageBox.Show("<l> must be in range of -10000mA and 10000mA",
"WARNING",
                    MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox9.Text = "0";
    }
}

private void textBox17_Leave(object sender, EventArgs e)
{
    double sayi;

```

```

        try
        {
            sayi = Convert.ToDouble(textBox17.Text);
        }
        catch
        {
            MessageBox.Show("<lo> must be in range of -10000mA and 10000mA",
"WARNING",
                MessageBoxButtons.OK, MessageBoxIcon.Warning);
            textBox17.Text = "0";
        }
    }

    private void textBox28_Leave(object sender, EventArgs e)
    {
        double sayi;

        try
        {
            sayi = Convert.ToDouble(textBox28.Text);
        }
        catch
        {
            MessageBox.Show("<lo> must be in range of -10000mA and 10000mA",
"WARNING",
                MessageBoxButtons.OK, MessageBoxIcon.Warning);
            textBox28.Text = "0";
        }
    }

    private void textBox39_Leave(object sender, EventArgs e)
    {
        double sayi;

        try
        {
            sayi = Convert.ToDouble(textBox39.Text);
        }
        catch
        {
            MessageBox.Show("<lo> must be in range of -10000mA and 10000mA",
"WARNING",
                MessageBoxButtons.OK, MessageBoxIcon.Warning);
            textBox39.Text = "0";
        }
    }

    private void textBox50_Leave(object sender, EventArgs e)
    {
        double sayi;

        try
        {
            sayi = Convert.ToDouble(textBox50.Text);
        }
        catch
        {

```

```

        MessageBox.Show("<lo> must be in range of -10000mA and 10000mA",
"WARNING",
        MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox50.Text = "0";
    }
}

private void textBox61_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox61.Text);
    }
    catch
    {
        MessageBox.Show("<lo> must be in range of -10000mA and 10000mA",
"WARNING",
        MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox61.Text = "0";
    }
}

private void textBox72_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox72.Text);
    }
    catch
    {
        MessageBox.Show("<lo> must be in range of -10000mA and 10000mA",
"WARNING",
        MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox72.Text = "0";
    }
}

private void textBox83_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox83.Text);
    }
    catch
    {
        MessageBox.Show("<lo> must be in range of -10000mA and 10000mA",
"WARNING",
        MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox83.Text = "0";
    }
}

```

```

private void textBox94_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox94.Text);
    }
    catch
    {
        MessageBox.Show("<l> must be in range of -10000mA and 10000mA",
"WARNING",
                MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox94.Text = "0";
    }
}

private void textBox105_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox105.Text);
    }
    catch
    {
        MessageBox.Show("<l> must be in range of -10000mA and 10000mA",
"WARNING",
                MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox105.Text = "0";
    }
}

private void textBox116_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox116.Text);
    }
    catch
    {
        MessageBox.Show("<l> must be in range of -10000mA and 10000mA",
"WARNING",
                MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox116.Text = "0";
    }
}

private void textBox127_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox127.Text);
    }
}

```

```

    }
    catch
    {
        MessageBox.Show("<lo> must be in range of -10000mA and 10000mA",
"WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox127.Text = "0";
    }
}

private void textBox138_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox138.Text);
    }
    catch
    {
        MessageBox.Show("<lo> must be in range of -10000mA and 10000mA",
"WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox138.Text = "0";
    }
}

private void textBox149_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox149.Text);
    }
    catch
    {
        MessageBox.Show("<lo> must be in range of -10000mA and 10000mA",
"WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox149.Text = "0";
    }
}

private void textBox160_Leave(object sender, EventArgs e)
{
    double sayi;

    try
    {
        sayi = Convert.ToDouble(textBox160.Text);
    }
    catch
    {
        MessageBox.Show("<lo> must be in range of -10000mA and 10000mA",
"WARNING",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        textBox160.Text = "0";
    }
}

```

```

    }
}

private void textBox11_Leave(object sender, EventArgs e)
{
    Int32 sayi;
    try
    {
        sayi = Convert.ToInt32(textBox11.Text);

        if (sayi > 40)
        {
            MessageBox.Show(" Kv maximum value is 40 ", "WARNING",
                MessageBoxButtons.OK,
                MessageBoxIcon.Warning);
            textBox11.Text = "40";
        }

        if (sayi < 1)
        {
            MessageBox.Show(" Kv minimum value is 1 ", "WARNING",
                MessageBoxButtons.OK,
                MessageBoxIcon.Warning);
            textBox11.Text = "1";
        }
    }
    catch
    {
        MessageBox.Show(" Kv range is between 1 and 40 ", "WARNING",
            MessageBoxButtons.OK,
            MessageBoxIcon.Warning);
        textBox11.Text = "1";
    }
}

private void textBox13_Leave(object sender, EventArgs e)
{
    Int32 sayi;
    try
    {
        sayi = Convert.ToInt32(textBox13.Text);

        if (sayi > 40)
        {
            MessageBox.Show(" Kv maximum value is 40 ", "WARNING",
                MessageBoxButtons.OK,
                MessageBoxIcon.Warning);
            textBox13.Text = "40";
        }

        if (sayi < 1)
        {
            MessageBox.Show(" Kv minimum value is 1 ", "WARNING",
                MessageBoxButtons.OK,
                MessageBoxIcon.Warning);
            textBox13.Text = "1";
        }
    }
}

```

```

        catch
        {
            MessageBox.Show(" Kv range is between 1 and 40 ", "WARNING",
                MessageBoxButtons.OK,
                MessageBoxIcon.Warning);
            textBox13.Text = "10";
        }
    }

    private void textBox24_Leave(object sender, EventArgs e)
    {
        Int32 sayi;
        try
        {
            sayi = Convert.ToInt32(textBox24.Text);

            if (sayi > 40)
            {
                MessageBox.Show(" Kv maximum value is 40 ", "WARNING",
                    MessageBoxButtons.OK,
                    MessageBoxIcon.Warning);
                textBox24.Text = "40";
            }

            if (sayi < 1)
            {
                MessageBox.Show(" Kv minimum value is 1 ", "WARNING",
                    MessageBoxButtons.OK,
                    MessageBoxIcon.Warning);
                textBox24.Text = "1";
            }
        }
        catch
        {
            MessageBox.Show(" Kv range is between 1 and 40 ", "WARNING",
                MessageBoxButtons.OK,
                MessageBoxIcon.Warning);
            textBox24.Text = "10";
        }
    }

    private void textBox35_Leave(object sender, EventArgs e)
    {
        Int32 sayi;
        try
        {
            sayi = Convert.ToInt32(textBox35.Text);

            if (sayi > 40)
            {
                MessageBox.Show(" Kv maximum value is 40 ", "WARNING",
                    MessageBoxButtons.OK,
                    MessageBoxIcon.Warning);
                textBox35.Text = "40";
            }

            if (sayi < 1)
            {

```

```

        MessageBox.Show(" Kv minimum value is 1 ", "WARNING",
                        MessageBoxButtons.OK,
MessageBoxIcon.Warning);
        textBox35.Text = "1";
    }
}
catch
{
    MessageBox.Show(" Kv range is between 1 and 40 ", "WARNING",
                    MessageBoxButtons.OK,
MessageBoxIcon.Warning);
    textBox35.Text = "10";
}
}

private void textBox46_Leave(object sender, EventArgs e)
{
    Int32 sayi;
    try
    {
        sayi = Convert.ToInt32(textBox46.Text);

        if (sayi > 40)
        {
            MessageBox.Show(" Kv maximum value is 40 ", "WARNING",
                            MessageBoxButtons.OK,
MessageBoxIcon.Warning);
            textBox46.Text = "40";
        }

        if (sayi < 1)
        {
            MessageBox.Show(" Kv minimum value is 1 ", "WARNING",
                            MessageBoxButtons.OK,
MessageBoxIcon.Warning);
            textBox46.Text = "1";
        }
    }
    catch
    {
        MessageBox.Show(" Kv range is between 1 and 40 ", "WARNING",
                        MessageBoxButtons.OK,
MessageBoxIcon.Warning);
        textBox46.Text = "10";
    }
}

private void textBox57_Leave(object sender, EventArgs e)
{
    Int32 sayi;
    try
    {
        sayi = Convert.ToInt32(textBox57.Text);

        if (sayi > 40)
        {
            MessageBox.Show(" Kv maximum value is 40 ", "WARNING",

```

```

                MessageBoxButtons.OK,
MessageBoxIcon.Warning);
        textBox57.Text = "40";
    }

    if (sayi < 1)
    {
        MessageBox.Show(" Kv minimum value is 1 ", "WARNING",
            MessageBoxButtons.OK,
MessageBoxIcon.Warning);
        textBox57.Text = "1";
    }
}
catch
{
    MessageBox.Show(" Kv range is between 1 and 40 ", "WARNING",
        MessageBoxButtons.OK,
MessageBoxIcon.Warning);
        textBox57.Text = "10";
}
}

private void textBox68_Leave(object sender, EventArgs e)
{
    Int32 sayi;
    try
    {
        sayi = Convert.ToInt32(textBox68.Text);

        if (sayi > 40)
        {
            MessageBox.Show(" Kv maximum value is 40 ", "WARNING",
                MessageBoxButtons.OK,
MessageBoxIcon.Warning);
                textBox68.Text = "40";
        }

        if (sayi < 1)
        {
            MessageBox.Show(" Kv minimum value is 1 ", "WARNING",
                MessageBoxButtons.OK,
MessageBoxIcon.Warning);
                textBox68.Text = "1";
        }
    }
    catch
    {
        MessageBox.Show(" Kv range is between 1 and 40 ", "WARNING",
            MessageBoxButtons.OK,
MessageBoxIcon.Warning);
            textBox68.Text = "10";
        }
    }

private void textBox79_Leave(object sender, EventArgs e)
{
    Int32 sayi;
    try

```

```

    {
        sayi = Convert.ToInt32(textBox79.Text);

        if (sayi > 40)
        {
            MessageBox.Show(" Kv maximum value is 40 ", "WARNING",
                MessageBoxButtons.OK,
                MessageBoxIcon.Warning);
            textBox79.Text = "40";
        }

        if (sayi < 1)
        {
            MessageBox.Show(" Kv minimum value is 1 ", "WARNING",
                MessageBoxButtons.OK,
                MessageBoxIcon.Warning);
            textBox79.Text = "1";
        }
    }
    catch
    {
        MessageBox.Show(" Kv range is between 1 and 40 ", "WARNING",
            MessageBoxButtons.OK,
            MessageBoxIcon.Warning);
        textBox79.Text = "10";
    }
}

private void textBox90_Leave(object sender, EventArgs e)
{
    Int32 sayi;
    try
    {
        sayi = Convert.ToInt32(textBox90.Text);

        if (sayi > 40)
        {
            MessageBox.Show(" Kv maximum value is 40 ", "WARNING",
                MessageBoxButtons.OK,
                MessageBoxIcon.Warning);
            textBox90.Text = "40";
        }

        if (sayi < 1)
        {
            MessageBox.Show(" Kv minimum value is 1 ", "WARNING",
                MessageBoxButtons.OK,
                MessageBoxIcon.Warning);
            textBox90.Text = "1";
        }
    }
    catch
    {
        MessageBox.Show(" Kv range is between 1 and 40 ", "WARNING",
            MessageBoxButtons.OK,
            MessageBoxIcon.Warning);
        textBox90.Text = "10";
    }
}

```

```

    }

    private void textBox101_Leave(object sender, EventArgs e)
    {
        Int32 sayi;
        try
        {
            sayi = Convert.ToInt32(textBox101.Text);

            if (sayi > 40)
            {
                MessageBox.Show(" Kv maximum value is 40 ", "WARNING",
                MessageBoxButtons.OK,
                MessageBoxIcon.Warning);
                textBox101.Text = "40";
            }

            if (sayi < 1)
            {
                MessageBox.Show(" Kv minimum value is 1 ", "WARNING",
                MessageBoxButtons.OK,
                MessageBoxIcon.Warning);
                textBox101.Text = "1";
            }
        }
        catch
        {
            MessageBox.Show(" Kv range is between 1 and 40 ", "WARNING",
            MessageBoxButtons.OK,
            MessageBoxIcon.Warning);
            textBox101.Text = "10";
        }
    }

    private void textBox112_Leave(object sender, EventArgs e)
    {
        Int32 sayi;
        try
        {
            sayi = Convert.ToInt32(textBox112.Text);

            if (sayi > 40)
            {
                MessageBox.Show(" Kv maximum value is 40 ", "WARNING",
                MessageBoxButtons.OK,
                MessageBoxIcon.Warning);
                textBox112.Text = "40";
            }

            if (sayi < 1)
            {
                MessageBox.Show(" Kv minimum value is 1 ", "WARNING",
                MessageBoxButtons.OK,
                MessageBoxIcon.Warning);
                textBox112.Text = "1";
            }
        }
        catch
    }

```

```

        {
            MessageBox.Show(" Kv range is between 1 and 40 ", "WARNING",
                MessageBoxButtons.OK,
                MessageBoxIcon.Warning);
            textBox112.Text = "10";
        }
    }

    private void textBox123_Leave(object sender, EventArgs e)
    {
        Int32 sayi;
        try
        {
            sayi = Convert.ToInt32(textBox123.Text);

            if (sayi > 40)
            {
                MessageBox.Show(" Kv maximum value is 40 ", "WARNING",
                    MessageBoxButtons.OK,
                    MessageBoxIcon.Warning);
                textBox123.Text = "40";
            }

            if (sayi < 1)
            {
                MessageBox.Show(" Kv minimum value is 1 ", "WARNING",
                    MessageBoxButtons.OK,
                    MessageBoxIcon.Warning);
                textBox123.Text = "1";
            }
        }
        catch
        {
            MessageBox.Show(" Kv range is between 1 and 40 ", "WARNING",
                MessageBoxButtons.OK,
                MessageBoxIcon.Warning);
            textBox123.Text = "10";
        }
    }

    private void textBox134_Leave(object sender, EventArgs e)
    {
        Int32 sayi;
        try
        {
            sayi = Convert.ToInt32(textBox134.Text);

            if (sayi > 40)
            {
                MessageBox.Show(" Kv maximum value is 40 ", "WARNING",
                    MessageBoxButtons.OK,
                    MessageBoxIcon.Warning);
                textBox134.Text = "40";
            }

            if (sayi < 1)
            {
                MessageBox.Show(" Kv minimum value is 1 ", "WARNING",
                    MessageBoxButtons.OK,
                    MessageBoxIcon.Warning);
                textBox134.Text = "1";
            }
        }
        catch
        {
            MessageBox.Show(" Kv range is between 1 and 40 ", "WARNING",
                MessageBoxButtons.OK,
                MessageBoxIcon.Warning);
            textBox134.Text = "10";
        }
    }
}

```

```

                MessageBoxButtons.OK,
MessageBoxIcon.Warning);
        textBox134.Text = "1";
    }
}
catch
{
    MessageBox.Show(" Kv range is between 1 and 40 ", "WARNING",
        MessageBoxButtons.OK,
MessageBoxIcon.Warning);
        textBox134.Text = "10";
    }
}

private void textBox145_Leave(object sender, EventArgs e)
{
    Int32 sayi;
    try
    {
        sayi = Convert.ToInt32(textBox145.Text);

        if (sayi > 40)
        {
            MessageBox.Show(" Kv maximum value is 40 ", "WARNING",
                MessageBoxButtons.OK,
MessageBoxIcon.Warning);
            textBox145.Text = "40";
        }

        if (sayi < 1)
        {
            MessageBox.Show(" Kv minimum value is 1 ", "WARNING",
                MessageBoxButtons.OK,
MessageBoxIcon.Warning);
            textBox145.Text = "1";
        }
    }
    catch
    {
        MessageBox.Show(" Kv range is between 1 and 40 ", "WARNING",
            MessageBoxButtons.OK,
MessageBoxIcon.Warning);
            textBox145.Text = "10";
        }
    }

private void textBox156_Leave(object sender, EventArgs e)
{
    Int32 sayi;
    try
    {
        sayi = Convert.ToInt32(textBox156.Text);

        if (sayi > 40)
        {
            MessageBox.Show(" Kv maximum value is 40 ", "WARNING",
                MessageBoxButtons.OK,
MessageBoxIcon.Warning);

```



```

int8 ddacH2;
int8 ddacL2;
int8 ddacH3;
int8 ddacL3;
int8 ddacH4;
int8 ddacL4;

int8 f_tur[15];// her bir tab taki fonksiyon türü

float f_i[15];//akım değeri
float f_iofset[15];// ofset akım değeri
int16 f_time[15];//akım zaman değerleri

int8 f_ksi[15];//2. derce sistem icin

int8 f_kx[15];//lorenz k
int8 f_ky[15];//lorenz k
int8 f_kz[15];//lorenz k
int8 f_ko[15];//lorenz k
int8 f_kv[15];//lorenz k
float f_iox[15];//lorenz ix ofset
float f_ioy[15];//lorenz iy ofset
float f_ioz[15];//lorenz iz ofset

int8 say_f;//yukarıdaki vektörleri adresler..
int8 say_f2;//fonksiyonlar koşturulurken alt index of say_f
int16 fk;//250us de bir koşmayı kontrol edecek

int1 isaret;//akım + - işareti
int16 ia;
int16 iae;
int16 ibe;
int16 ib;
int16 ic;
int16 ice;
int16 id;
float i_out;//hesaplanan çıkış akımı
int16 k;
int16 k_k1;//k bir önceki
int8 fonk_tur;

float sin_d[31];//9derecede bir sin gösteriyorum, 0 ile beraber olunca 270derece için 31 data oluyor
int8 d_set;// sayması gereken derece
int8 d_say;// saydığı anlık derece değeri
float i_max;
float i_ofset;

float di;//rampa i artımı
float dt;//T-to
float to_ters;// 1/to

float z1;
float z2;
float ksi;
float wn;
float k1_sys2;

```

```

float k2_sys2;
float k3_sys2;

float Xx_z1;
float Xy_z1;
float Xz_z1;
float Xx_z;
float Xy_z;
float Xz_z;
int8 a ;
float b;
int8 c;
int8 kx;
int8 ky;
int8 kz;
int8 ko;
int8 kv;
float i_ox;
float i_oy;
float i_oz;
float Ts;
float Xc;
float Yc;
float Zc;
float x_o,y_o,z_o;

int16 say_adc;

char gelen_veri[100];

int8 reset;

void adc_ayar(void);
void ilk_ayar(void);

void gelen_veri_depola(void);
float sin_v (int8 derece);

void left_dac_yaz(void);
void right_dac_yaz(void);

void kesme_ayar(void);
void akim_ayar(int16 akim);
void p_basamak(void);
void p_sincos(void);
void p_rampa(void);
void p_1dsys(void);
void p_2dsys(void);
void p_lorenz(void);
void k_basamak(void);
void k_sincos(void);
void k_rampa(void);
void k_1dsys(void);
void k_2dsys(void);
void k_lorenz(void);
void parametre_ayari(void);

```

```

void kostur_fonksiyon(void);
void kontrol_ia(void);
void kontrol_ib(void);
void kontrol_ic(void);

void RihtLeft_akim_0A(void);

#INT_TIMER1
void timer1 (void)
{
int8 data=0;
int16 count_timer;

//set_timer1(63035);//65535-2500=63035=500us
set_timer1(63035);//65535-1250=64285=250us

if (input (wifi_data))
{
if (isaret==0) left_dac_yaz();//1 ise -, 0 ise +
else right_dac_yaz();

iae=ia;
ibe=ib;
//ice=ic;

if (fk==0) parametre_ayari();
else
{
kostur_fonksiyon();

//! say_adc++;
//! if (say_adc>4000)
//! {
//! if (input (!wifi_data)) setup_timer_1 ( T1_DISABLED );//t1 stop
//! else
//! {
//! if (iae==0)
//! {
//! if (ibe!=0) kontrol_ib(); else
//! if (ice!=0) kontrol_ic();
//! }else kontrol_ia();
//! }
//! }
//! say_adc=0;
//! }
}

count_timer=get_timer1();
if (count_timer<63035) set_timer1(65530);
}else setup_timer_1 ( T1_DISABLED );//t1 stop

}

void main()
{

```

```

int16 i;
int8 data;
int16 i16;
float dpi;
float ik1_out;

int16 data16;
int16 dadc;

ilk_ayar();

setup_timer_1 ( T1_DISABLED );//t1 stop

RihtLeft_akim_0A();

input (wifi_data);// wifi seri out lojik durum deęişmeye başlayınca '0' else '1'

output_high(PIN_B5);//seri port trasmit bit
input(PIN_B4);//seri port recieve bit

output_high(right_opm); // alt saę fetleri süren opamplar çıkış 0V yapıldı
output_high(left_opm); // alt sol fetleri süren opamplar çıkış 0V yapıldı

output_high(PIN_C4);//cs_dac
output_low(PIN_D3);//clck_dac
output_high(PIN_C3);//Lld_dac
output_high(PIN_D1);//Rld_dac

output_low(PIN_C7);//sdi_L9
output_low(PIN_D4);//sdi_L0v9
output_low(PIN_D5);//sdi_L0v09
output_low(PIN_D6);//sdi_L0v009

output_low(PIN_D2);//sdi_R9
output_low(PIN_C5);//sdi_R0v9
output_low(PIN_C6);//sdi_R0v09
output_low(PIN_D0);//sdi_R0v009

dpi=pi;
dpi/=180;
i=0;

for (i16=0;i16<271;i16+=9)
{
if(i16==90) sin_d[i]=0.980;else
if(i16==270) sin_d[i]=-0.980;else sin_d[i]=sin(dpi*i16);

i++;
}

if (reset==0)reset=0xAA;else if (say_f>0) goto islem_baslat;

tek_1:
say_f=0;
tek_2:

```

```

i=-1;

fkmt_b:
    data=0;
    data=fgetc(modul);if (data==0)goto fkmt_b;
    i++;
    gelen_veri[i]=data;
    if (data!=',')goto fkmt_b;

    if (
(gelen_veri[0]=='1')&&(gelen_veri[1]=='5')&&(gelen_veri[2]=='0')&&(gelen_veri[3]=='0')&&(gelen_ve
ri[4]=='0'))
    {
        gelen_veri[0]='1';
        gelen_veri[1]='5';
        i=1;
    }else if (
(gelen_veri[0]=='3')&&(gelen_veri[1]=='0')&&(gelen_veri[2]=='0')&&(gelen_veri[3]=='0')&&(gelen_ve
ri[4]=='0') )
    {
        gelen_veri[0]='2';
        gelen_veri[1]='3';
        i=1;
    }
    else if (
(gelen_veri[0]=='4')&&(gelen_veri[1]=='0')&&(gelen_veri[2]=='0')&&(gelen_veri[3]=='0')&&(gelen_ve
ri[4]=='0')) goto islem_baslat;
    else if (
(gelen_veri[0]=='5')&&(gelen_veri[1]=='0')&&(gelen_veri[2]=='0')&&(gelen_veri[3]=='0')&&(gelen_ve
ri[4]=='0')) goto sistem_durdur;
    //else goto fkmt_b;
    else goto tek_1;

fd_b:
    data=0;
    data=fgetc(modul);if (data==0)goto fd_b;
    i++;
    gelen_veri[i]=data;
    if (data!=',')goto fd_b;

fi_b:
    data=0;
    data=fgetc(modul);if (data==0)goto fi_b;
    i++;
    gelen_veri[i]=data;
    if (data!=',')goto fi_b;

i_b:
    data=0;
    data=fgetc(modul);if (data==0)goto i_b;
    i++;
    gelen_veri[i]=data;
    if (data!=',')goto i_b;

io_b:

```

```
data=0;
data=fgetc(modul);if (data==0)goto io_b;
i++;
gelen_veri[i]=data;
if (data!=',')goto io_b;
```

```
t_b:
data=0;
data=fgetc(modul);if (data==0)goto t_b;
i++;
gelen_veri[i]=data;
if (data!=',')goto t_b;
```

```
ksi_b:
data=0;
data=fgetc(modul);if (data==0)goto ksi_b;
i++;
gelen_veri[i]=data;
if (data!=',')goto ksi_b;
```

```
lkx_b:
data=0;
data=fgetc(modul);if (data==0)goto lkx_b;
i++;
gelen_veri[i]=data;
if (data!=',')goto lkx_b;
```

```
lky_b:
data=0;
data=fgetc(modul);if (data==0)goto lky_b;
i++;
gelen_veri[i]=data;
if (data!=',')goto lky_b;
```

```
lkz_b:
data=0;
data=fgetc(modul);if (data==0)goto lkz_b;
i++;
gelen_veri[i]=data;
if (data!=',')goto lkz_b;
```

```
lko_b:
data=0;
data=fgetc(modul);if (data==0)goto lko_b;
i++;
gelen_veri[i]=data;
if (data!=',')goto lko_b;
```

```
lkv_b:
data=0;
data=fgetc(modul);if (data==0)goto lkv_b;
i++;
gelen_veri[i]=data;
if (data!=',')goto lkv_b;
```

```
lixo_b:
```

```
data=0;
data=fgetc(modul);if (data==0)goto lixo_b;
i++;
gelen_veri[i]=data;
if (data!=',')goto lixo_b;
```

```
liyo_b:
data=0;
data=fgetc(modul);if (data==0)goto liyo_b;
i++;
gelen_veri[i]=data;
if (data!=',')goto liyo_b;
```

```
lizo_b:
data=0;
data=fgetc(modul);if (data==0)goto lizo_b;
i++;
gelen_veri[i]=data;
if (data!=',')goto lizo_b;

gelen_veri_depola();

i++;
delay_ms(100);
gelen_veri[i]=0;
fputs(gelen_veri,modul);

restart_wdt();
gelen_veri[0]=0;

goto tek_2;
```

```
islem_baslat:
delay_ms(5);
itoa(40000,10, gelen_veri);
fputs(gelen_veri,modul);
```

```
say_f2=0;
fk=0;
```

```
isaret=0;//0:+; 1:-
ia=0;
ib=0;
ic=0;
```

```
kesme_ayar();//timer start
```

```
delay_ms(500);
```

```
b_degisim_0:
delay_us(50);
if (input (wifi_data)) goto b_degisim_0; // wifi seri out lojik durum deđişmeye bađlayınca '0' else '1'
```

```
b_degisim_1:
delay_us(50);
```

```

    if (input (!wifi_data)) goto b_degisim_1; // wifi seri out lojik durum deęişmeye başlayınca '0' else '1'

sistem_durdur:
    setup_timer_1 ( T1_DISABLED );//t1 stop
    RihtLeft_akim_0A();

    gelen_veri[0]=0;

    //delay_ms(5);
    itoa(50000,10, gelen_veri);
    fputs(gelen_veri,modul);

    goto tek_1;

}

void RihtLeft_akim_0A(void)
{
    output_high(right_opm);//lojik "1"de (-) uç 5v oluyor
    output_high(left_opm);

    output_low(optoHleft);//üst sol fet off
    output_low(optoHright);//üst sağ fet off

    restart_wdt();
}

void adc_ayar(void)
{
    setup_adc_ports( AN0_AN1_AN3 );
    setup_adc(ADC_CLOCK_INTERNAL);
    set_adc_channel( 0 );

    input(PIN_A0);
    input(PIN_A1);
    //input(PIN_A3);

    say_adc=0;

    delay_us(100);
}

void ilk_ayar(void)
{
    setup_adc_ports(NO_ANALOGS);
    setup_timer_3(T3_DISABLED|T3_DIV_BY_1);

    adc_ayar();

    kesme_ayar();
}

void gelen_veri_depola(void)
{
    int8 i,k;
    char str[10];

```

```
float sf;
```

```
restart_wdt();
```

```
//-----2: bir önceki veri ile yer değiştir, yani iletişim esnasında hata oluşmuş  
if (gelen_veri[0]=='2')  
{  
    if (say_f>0)say_f--;  
}else if (gelen_veri[0]=='1')delay_us(1);else if (say_f>0)say_f--;
```

```
//----- komut algılandı veri 1  
for (i=0;gelen_veri[i]!=';';i++) {}  
i++;
```

```
//----- tab algılandı veri 2  
for (i=i;gelen_veri[i]!=';';i++){ }  
i++;
```

```
//----- fonksiyon turu 1-9 algılandı veri 3  
k=0;  
for (i=i;gelen_veri[i]!=';';i++)  
{  
    str[k]=gelen_veri[i];  
    k++;  
}  
str[k]=0;//fonksiyon türü string olarak çekildi  
f_tur[say_f]= atoi(str);  
i++;
```

```
//----- akım turu algılandı veri 4  
k=0;  
for (i=i;gelen_veri[i]!=';';i++)  
{  
    str[k]=gelen_veri[i];//atoi: int8  
    k++;  
}  
str[k]=0;//fonksiyon türü string olarak çekildi  
sf=atof(str);  
//sf=akim_kalibre(sf);  
f_i[say_f]= sf;  
i++;
```

```
//----- akım ofset turu algılandı veri 5  
k=0;  
for (i=i;gelen_veri[i]!=';';i++)  
{  
    str[k]=gelen_veri[i];//atoi: int8  
    k++;  
}
```

```
str[k]=0;//fonksiyon türü string olarak çekildi
sf=atof(str);
//sf=akim_kalibre(sf);
f_iofset[say_f]= sf;
i++;
```

```
//----- T turu algılandı veri 6
k=0;
for (i=i;gelen_veri[i]!=';';i++)
{
    str[k]=gelen_veri[i];//atoi: int8
    k++;
}
str[k]=0;//fonksiyon türü string olarak çekildi
f_time[say_f]= atol(str);//atol: int16
i++;
```

```
//----- ksi turu algılandı veri 7
k=0;
for (i=i;gelen_veri[i]!=';';i++)
{
    str[k]=gelen_veri[i];//atoi: int8
    k++;
}
str[k]=0;//fonksiyon türü string olarak çekildi
f_ksi[say_f]= atoi(str);//atol: int16
i++;
```

```
//----- kx turu algılandı veri 8
k=0;
for (i=i;gelen_veri[i]!=';';i++)
{
    str[k]=gelen_veri[i];//atoi: int8
    k++;
}
str[k]=0;//fonksiyon türü string olarak çekildi
f_kx[say_f]= atoi(str);//atol: int16
i++;
```

```
//----- ky turu algılandı veri 9
k=0;
for (i=i;gelen_veri[i]!=';';i++)
{
    str[k]=gelen_veri[i];//atoi: int8
    k++;
}
str[k]=0;//fonksiyon türü string olarak çekildi
f_ky[say_f]= atoi(str);//atol: int16
i++;
```

```
//----- kz turu algılandı veri 10
```

```

k=0;
for (i=i;gelen_veri[i]!=';';i++)
{
    str[k]=gelen_veri[i];//atoi: int8
    k++;
}
str[k]=0;//fonksiyon türü string olarak çekildi
f_kz[say_f]= atoi(str);//atol: int16
i++;

//----- ko turu algılandı veri 11
k=0;
for (i=i;gelen_veri[i]!=';';i++)
{
    str[k]=gelen_veri[i];//atoi: int8
    k++;
}
str[k]=0;//fonksiyon türü string olarak çekildi
f_ko[say_f]= atoi(str);//atol: int16
i++;

//----- kv turu algılandı veri 12
k=0;
for (i=i;gelen_veri[i]!=';';i++)
{
    str[k]=gelen_veri[i];//atoi: int8
    k++;
}
str[k]=0;//fonksiyon türü string olarak çekildi
f_kv[say_f]= atoi(str);//atol: int16
i++;

//----- i_ox turu algılandı veri 13
k=0;
for (i=i;gelen_veri[i]!=';';i++)
{
    str[k]=gelen_veri[i];//atoi: int8
    k++;
}
str[k]=0;//fonksiyon türü string olarak çekildi
f_iox[say_f]= atof(str);//atol: int16
i++;

//----- i_oy turu algılandı veri 14
k=0;
for (i=i;gelen_veri[i]!=';';i++)
{
    str[k]=gelen_veri[i];//atoi: int8
    k++;
}
str[k]=0;//fonksiyon türü string olarak çekildi
f_ioy[say_f]= atof(str);//atol: int16
i++;

```

```

//----- i_oz turu algılandı veri 15
k=0;
for (i=i;gelen_veri[i]!=';';i++)
{
    str[k]=gelen_veri[i];//atoi: int8
    k++;
}
str[k]=0;//fonksiyon türü string olarak çekildi
f_ioz[say_f]= atof(str);//atol: int16
i++;

say_f++;
restart_wdt();
}

float sin_v (int8 derece)
{
    if (derece <9) return (0.0);else
    if (derece <18) return (0.1564345);else
    if (derece <27) return (0.309017);else
    if (derece <36) return (0.453991);else
    if (derece <45) return (0.587785);else
    if (derece <54) return (0.707107);else
    if (derece <63) return (0.809017);else
    if (derece <72) return (0.891007);else
    if (derece <81) return (0.951057);else
    if (derece <90) return (0.987688); else return (1.0);
}

void left_dac_yaz(void) // 21usn işlem süresi
{
    output_high(right_opm);//lojik "1"de (-) uç 5v oluyor;aksi "Z"
    output_low(optoHleft);// H köprüde, akım sol üst den sağ alta doğru akar

    clck_dac=0;
    cs_dac=0;

    ddacH1=make8(ia,1);
    ddacH2=make8(ib,1);
    ddacH3=make8(ic,1);
    ddacH4=make8(id,1);

    ddacL1=make8(ia,0);
    ddacL2=make8(ib,0);
    ddacL3=make8(ic,0);
    ddacL4=make8(id,0);

//-----
    sdi_L9=0;
    sdi_L0v9=0;
    sdi_L0v09=0;
    sdi_L0v009=0;
    clck_dac=1;clck_dac=0;// write to dac

//-----
    sdi_L9=1;
    sdi_L0v9=1;

```

```

sdi_L0v09=1;
sdi_L0v009=1;
clck_dac=1;clck_dac=0;// bufferet

//-----
sdi_L9=1;
sdi_L0v9=1;
sdi_L0v09=1;
sdi_L0v009=1;
clck_dac=1;clck_dac=0;// gain 1

//-----
sdi_L9=1;
sdi_L0v9=1;
sdi_L0v09=1;
sdi_L0v009=1;
clck_dac=1;clck_dac=0;// vout avoliable

//-----
sdi_L9=0;
sdi_L0v9=0;
sdi_L0v09=0;
sdi_L0v009=0;;
clck_dac=1;clck_dac=0; //bit 12

//-----
sdi_L9=0;
sdi_L0v9=0;
sdi_L0v09=0;
sdi_L0v009=0;
clck_dac=1;clck_dac=0; //bit 11

//-----
sdi_L9=bit_test(ddacH1,1);
sdi_L0v9=bit_test(ddacH2,1);
sdi_L0v09=bit_test(ddacH3,1);
sdi_L0v009=bit_test(ddacH4,1);
clck_dac=1;clck_dac=0; //bit 10

//-----
sdi_L9=bit_test(ddacH1,0);
sdi_L0v9=bit_test(ddacH2,0);
sdi_L0v09=bit_test(ddacH3,0);
sdi_L0v009=bit_test(ddacH4,0);
clck_dac=1;clck_dac=0; //bit 9

//-----
sdi_L9=bit_test(ddacL1,7);
sdi_L0v9=bit_test(ddacL2,7);
sdi_L0v09=bit_test(ddacL3,7);
sdi_L0v009=bit_test(ddacL4,7); ;
clck_dac=1;clck_dac=0; //bit 8

//-----
sdi_L9=bit_test(ddacL1,6);
sdi_L0v9=bit_test(ddacL2,6);

```

```

sdi_L0v09=bit_test(ddacL3,6);
sdi_L0v009=bit_test(ddacL4,6);
clk_dac=1;clk_dac=0; //bit 7

//-----
sdi_L9=bit_test(ddacL1,5);
sdi_L0v9=bit_test(ddacL2,5);
sdi_L0v09=bit_test(ddacL3,5);
sdi_L0v009=bit_test(ddacL4,5);
clk_dac=1;clk_dac=0; //bit 6

//-----
sdi_L9=bit_test(ddacL1,4);
sdi_L0v9=bit_test(ddacL2,4);
sdi_L0v09=bit_test(ddacL3,4);
sdi_L0v009=bit_test(ddacL4,4);
clk_dac=1;clk_dac=0; //bit 5

//-----
sdi_L9=bit_test(ddacL1,3);
sdi_L0v9=bit_test(ddacL2,3);
sdi_L0v09=bit_test(ddacL3,3);
sdi_L0v009=bit_test(ddacL4,3);
clk_dac=1;clk_dac=0; //bit 4

//-----
sdi_L9=bit_test(ddacL1,2);
sdi_L0v9=bit_test(ddacL2,2);
sdi_L0v09=bit_test(ddacL3,2);
sdi_L0v009=bit_test(ddacL4,2);
clk_dac=1;clk_dac=0; //bit 3

//-----
sdi_L9=bit_test(ddacL1,1);
sdi_L0v9=bit_test(ddacL2,1);
sdi_L0v09=bit_test(ddacL3,1);
sdi_L0v009=bit_test(ddacL4,1);
clk_dac=1;clk_dac=0; //bit 2

//-----
sdi_L9=bit_test(ddacL1,0);
sdi_L0v9=bit_test(ddacL2,0);
sdi_L0v09=bit_test(ddacL3,0);
sdi_L0v009=bit_test(ddacL4,0);
clk_dac=1;clk_dac=0; //bit 1

//-----

Lld_dac=0;
Lld_dac=1;

cs_dac=1;

output_high(optoHright);// H köprüde, akım sol üst den sağ alta doğru akar
output_low(left_opm);//lojik "1"de (-) uç 5v oluyor;aksi "Z"

```

```

restart_wdt();
}

void right_dac_yaz(void) // 21usn işlem süresi
{
output_high(left_opm);//lojik "1"de (-) uç 5v oluyor
output_low(optoHright);// H köprüde, akım sol üst den sağ alta doğru akar

clk_dac=0;
cs_dac=0;

ddacH1=make8(ia,1);
ddacH2=make8(ib,1);
ddacH3=make8(ic,1);
ddacH4=make8(id,1);

ddacL1=make8(ia,0);
ddacL2=make8(ib,0);
ddacL3=make8(ic,0);
ddacL4=make8(id,0);

//-----
sdi_R9=0;
sdi_R0v9=0;
sdi_R0v09=0;
sdi_R0v009=0;
clk_dac=1;clk_dac=0;// write to dac

//-----
sdi_R9=1;
sdi_R0v9=1;
sdi_R0v09=1;
sdi_R0v009=1;
clk_dac=1;clk_dac=0;// bufferet

//-----
sdi_R9=1;
sdi_R0v9=1;
sdi_R0v09=1;
sdi_R0v009=1;
clk_dac=1;clk_dac=0;// gain 1

//-----
sdi_R9=1;
sdi_R0v9=1;
sdi_R0v09=1;
sdi_R0v009=1;
clk_dac=1;clk_dac=0;// vout avoliable

//-----
sdi_R9=0;
sdi_R0v9=0;
sdi_R0v09=0;
sdi_R0v009=0;
clk_dac=1;clk_dac=0; //bit 12

```

```

//-----
sdi_R9=0;
sdi_R0v9=0;
sdi_R0v09=0;
sdi_R0v009=0;
clck_dac=1;clck_dac=0; //bit 11

//-----
sdi_R9=bit_test(ddacH1,1);
sdi_R0v9=bit_test(ddacH2,1);
sdi_R0v09=bit_test(ddacH3,1);
sdi_R0v009=bit_test(ddacH4,1);
clck_dac=1;clck_dac=0; //bit 10

//-----
sdi_R9=bit_test(ddacH1,0);
sdi_R0v9=bit_test(ddacH2,0);
sdi_R0v09=bit_test(ddacH3,0);
sdi_R0v009=bit_test(ddacH4,0);
clck_dac=1;clck_dac=0; //bit 9

//-----
sdi_R9=bit_test(ddacL1,7);
sdi_R0v9=bit_test(ddacL2,7);
sdi_R0v09=bit_test(ddacL3,7);
sdi_R0v009=bit_test(ddacL4,7);
clck_dac=1;clck_dac=0; //bit 8

//-----
sdi_R9=bit_test(ddacL1,6);
sdi_R0v9=bit_test(ddacL2,6);
sdi_R0v09=bit_test(ddacL3,6);
sdi_R0v009=bit_test(ddacL4,6);
clck_dac=1;clck_dac=0; //bit 7

//-----
sdi_R9=bit_test(ddacL1,5);
sdi_R0v9=bit_test(ddacL2,5);
sdi_R0v09=bit_test(ddacL3,5);
sdi_R0v009=bit_test(ddacL4,5);
clck_dac=1;clck_dac=0; //bit 6

//-----
sdi_R9=bit_test(ddacL1,4);
sdi_R0v9=bit_test(ddacL2,4);
sdi_R0v09=bit_test(ddacL3,4);
sdi_R0v009=bit_test(ddacL4,4);
clck_dac=1;clck_dac=0; //bit 5

//-----
sdi_R9=bit_test(ddacL1,3);
sdi_R0v9=bit_test(ddacL2,3);
sdi_R0v09=bit_test(ddacL3,3);
sdi_R0v009=bit_test(ddacL4,3);
clck_dac=1;clck_dac=0; //bit 4

```

```

//-----
sdi_R9=bit_test(ddacL1,2);
sdi_R0v9=bit_test(ddacL2,2);
sdi_R0v09=bit_test(ddacL3,2);
sdi_R0v009=bit_test(ddacL4,2);
clck_dac=1;clck_dac=0; //bit 3

//-----
sdi_R9=bit_test(ddacL1,1);
sdi_R0v9=bit_test(ddacL2,1);
sdi_R0v09=bit_test(ddacL3,1);
sdi_R0v009=bit_test(ddacL4,1);
clck_dac=1;clck_dac=0; //bit 2

//-----
sdi_R9=bit_test(ddacL1,0);
sdi_R0v9=bit_test(ddacL2,0);
sdi_R0v09=bit_test(ddacL3,0);
sdi_R0v009=bit_test(ddacL4,0);
clck_dac=1;clck_dac=0; //bit 1

Rld_dac=0;
Rld_dac=1;

cs_dac=1;

output_high(optoHleft);// H köprüde, akım sol üst den sağ alta doğru akar
output_low(right_opm);//lojik "1"de (-) uç 5v oluyor;aksi "Z"

restart_wdt();
}

void kesme_ayar(void)
{
setup_timer_0 ( RTCC_INTERNAL| RTCC_DIV_2);

setup_timer_1 ( T1_INTERNAL | T1_DIV_BY_2);
set_timer1(63035);//65535-2500=63035=500us

enable_interrupts(GLOBAL);
enable_interrupts(INT_TIMER1);
}

void akim_ayar(int16 akim)//150us işlem süresi
{
restart_wdt();

ia=akim;
ia/=1000;
ia*=1000;

ib=akim;
ib-=ia;
ic=ib;
ib/=100;
ib*=100;
}

```

```

ic-=ib;
id=ic;
ic/=10;
ic*=10;

id-=ic;
id*=100;

ic*=10;
ia/=10;

restart_wdt();
}

void p_basamak(void)
{
k=f_time[say_f2];
k*=2;
fk=1;
i_out=f_iofset[say_f2];
i_out+=f_i[say_f2];

if (i_out<0)
{
i_out*=-1;
isaret=1;//1 - ; 0 +
}
else isaret=0;//1 - ; 0 +

akim_ayar((int16)i_out);
}
void p_sincos(void)
{
int8 ftur;

k=f_time[say_f2];

k*=0.05;//kT, 20ms için 1T yani 500us ise Xms için X/20==X*0.05 Tedet
k_k1=k;
d_say=0;

if (fonk_tur==2)d_set=10; else //çeyrek sin
if (fonk_tur==3)d_set=20;else //yarım sin
if (fonk_tur==4) //çeyrek cos
{
d_set=20;
d_say=10;
}
else if (fonk_tur==5) //yarım cos
{
d_set=30;
d_say=10;
}
i_ofset=f_iofset[say_f2];
i_max=f_i[say_f2];

```

```

    fk=1;
}
void p_rampa(void)
{
    k=f_time[say_f2];//ms deęeri
    k*=2;//kT, T=0.5msn

    di=f_i[say_f2];
    di/=k;

    i_out=f_iofset[say_f2];

    fk=1;
}
void p_1dsys(void)
{
    k=f_time[say_f2];//ms deęeri

    dt=k;
    dt*=0.2;// 1/5 yapılarak to bulundu

    to_ters=1;
    to_ters/=dt;
    to_ters*=1000;// 1/to hesaplandı sn cinsinden

    dt=0.5;// to-T yapıldı
    dt*=0.001;//sn çevrildi

    k*=2;//kT hesaplandı,Tsim/Tsample

    i_max=f_i[say_f2];
    i_ofset=f_iofset[say_f2];

    z1=0;

    fk=1;
}
void p_2dsys(void)
{
    float gecici;

    k=f_time[say_f2];//ms deęeri

    ksi=f_ksi[say_f2];
    ksi*=0.1;

    gecici=k;
    gecici*=0.001;//T sn çevrildi
    gecici*=ksi;
    wn=5;
    wn/=gecici;

    k1_sys2=wn;
    k1_sys2*=wn;
    k1_sys2*=0.0005;// * Tsample
    k1_sys2*=0.0005;

```

```

k2_sys2=2;
k2_sys2*=ksi;
k2_sys2*=wn;
k2_sys2*=0.0005;

k3_sys2=k1_sys2;
k3_sys2+=1;
k3_sys2-=k2_sys2;

k2_sys2-=2;

k*=2;// sim süresi hesap; k ms ve Tsample 0.5ms
i_max=f_i[say_f2];
i_ofset=f_iofset[say_f2];

z1=0.0;
z2=0.0;

fk=1;

restart_wdt();
}
void p_lorenz(void)
{
Xx_z1 = 0.500;
Xy_z1 = -1.000;
Xz_z1 = 0.500;
Xx_z = Xx_z1;
Xy_z = Xy_z1;
Xz_z = Xz_z1;

a = 10;
b = 2.66667;
c = 28;

kx=f_kx[say_f2];
ky=f_ky[say_f2];
kz=f_kz[say_f2];

ko=f_ko[say_f2];
kv=f_kv[say_f2];

i_ox=f_iox[say_f2];
i_oy=f_ioy[say_f2];
i_oz=f_ioz[say_f2];

Ts = 0.02;//lorenz örnekleme periyodu

k=2000;// 1/tsample, 1/0.0005
k/=kv;// kv, 1 - 40 aralığında
k*=0.02;
k_k1=k;

fk=1;

```

```

restart_wdt();
}

void k_basamak(void)
{
    fk++;
    if (fk==k)fk=0;
    restart_wdt();
}
void k_sincos(void)
{
float iout;

    k--;//kT
    if (k==0)
    {
        k=k_k1;

        iout=sin_d[d_say];
        iout*=i_max;

        i_out=iout;
        i_out+=i_offset;
        if (i_out<0) {i_out*=-1;isaret=1;}else isaret=0;//22.5us
        akim_ayar(i_out);//130us

        d_say++;
        if (d_say==d_set) fk=0;
    }
    restart_wdt();
}
void k_rampa(void)
{

    i_out+=di;

    if (i_out<0) {i_out*=-1;isaret=1;}else isaret=0;//22.5us
    akim_ayar(i_out);//130us

    if (isaret==1)i_out*=-1;

    k--;
    if (k==0) fk=0;
    restart_wdt();
}
void k_1dsys(void)
{
    float i_gec;

    i_gec=i_max;
    i_gec*=z1;
    i_gec*=0.0005;

    i_out=i_offset;
    i_out+=i_gec;

```

```

if (i_out<0) {i_out*=-1;isaret=1;}else isaret=0;//22.5us
akim_ayar(i_out);//130us

i_gec=z1;
z1=dt;
z1*=i_gec;
z1+=1;
z1*=to_ters;

k--;
if(k==0) fk=0;

restart_wdt();
}
void k_2dsys(void)
{
float zg;
float gec1;
float gec2;

i_out=z2*k1_sys2;
i_out+=i_offset;
if (i_out>15000) i_out=15000; else if (i_out<-15000) i_out=-15000;
if (i_out<0) {i_out*=-1;isaret=1;}else isaret=0;//22.5us
akim_ayar(i_out);//130us

zg=z2;
z2=z1;

gec1=k2_sys2;
gec1*=z1;

gec2=k3_sys2;
gec2*=zg;

z1=i_max;
z1-=gec1;
z1-=gec2;

k--;
if(k==0) fk=0;

restart_wdt();
}
void k_lorenz(void)
{
float g1,g2;

k--;
if (k==0)
{
//output_toggle(PIN_D2);

k=k_k1;

Xx_z1 = Xx_z;

```

```

Xy_z1 = Xy_z;
Xz_z1 = Xz_z;

Xc = Xx_z1 * Ts;
Yc = Xy_z1 * Ts;
Zc = Xz_z1 * Ts;

Xx_z = Xx_z1 + (Yc - Xc) * a;
Xy_z = Xy_z1 + (Xc * c - Xc * Zc - Yc);
Xz_z = Xz_z1 + (Xc * Yc - Zc * b);
x_o = (Xc + i_ox) * kx;
y_o = (Yc + i_oy) * ky;
z_o = (Zc + i_oz) * kz;

i_out = x_o;
i_out += y_o;
i_out += z_o;
i_out *= ko;

if (i_out > 10000) i_out = 10000; else
if (i_out < -10000) i_out = -10000;

if (i_out < 0) {i_out *= -1; isaret = 1;} else isaret = 0; //22.5us
akim_ayar(i_out); //130us
}

restart_wdt();
}

void parametre_ayari(void)
{
if (say_f2 < say_f) { } else say_f2 = 0;
fonk_tur = f_tur[say_f2];

if (fonk_tur == 1) p_basamak(); else
if (fonk_tur < 6) p_sincos(); else
if (fonk_tur == 6) p_rampa(); else
if (fonk_tur == 7) p_1dsys(); else
if (fonk_tur == 8) p_2dsys(); else p_lorenz();

say_f2++;
}

void kostur_fonksiyon(void)
{
if (fonk_tur == 1) k_basamak(); else
if (fonk_tur < 6) k_sincos(); else
if (fonk_tur == 6) k_rampa(); else
if (fonk_tur == 7) k_1dsys(); else
if (fonk_tur == 8) k_2dsys(); else k_lorenz();
}

void kontrol_ia(void)
{
int16 dadc;

```

```

int8 i;

set_adc_channel( 0 );

iae*=0.05;//ib/100 * 5; ib==300 ise ib/100==3 ve ib*=5==15
iae*=0.5;

dadc=0;
delay_us(10);
for (i=0;i<10;i++)
{
    delay_us(5);
    dadc+=read_adc();
}
dadc*=0.1;
if (dadc<iae)
{
    itoa(50001,10, gelen_veri);
    fputs(gelen_veri,modul);
}
}

void kontrol_ib(void)
{
int16 dadc;
int8 i;

set_adc_channel( 1 );
ibe*=0.05;//ib/100 * 5; ib==300 ise ib/100==3 ve ib*=5==15
ibe*=0.5;

dadc=0;
delay_us(10);
for (i=0;i<10;i++)
{
    delay_us(5);
    dadc+=read_adc();
}
dadc*=0.1;
if (dadc<ibe)
{
    itoa(50001,10, gelen_veri);
    fputs(gelen_veri,modul);
}
}

void kontrol_ic(void)
{
int16 dadc;
int8 i;

set_adc_channel( 3 );
ice*=0.05;//ib/100 * 5; ib==300 ise ib/100==3 ve ib*=5==15
ice*=0.5;

dadc=0;

```

```
delay_us(10);
for (i=0;i<10;i++)
{
    delay_us(5);
    dadc+=read_adc();
}
dadc*=0.1;
if (dadc<ice)
{
    itoa(50001,10, gelen_veri);
    fputs(gelen_veri,modul);
}
}
```



ÖZGEÇMİŞ

Adı ve Soyadı : Gülçin MÜHÜRÇÜ

Doğum Tarihi :

E-mail :

Öğrenim Durumu :

Derece	Bölüm/Program	Üniversite	Yıl
Lisans	Makine Mühendisliği	Sakarya Üniversitesi	2004
Yüksek Lisans	Makine Mühendisliği	Sakarya Üniversitesi	2009
Doktora	İmalat Mühendisliği	Tarsus Üniversitesi	2018-devam

Görevler :

Görev Ünvanı	Görev Yeri	Yıl
Öğretim Görevlisi	Kırklareli Üniversitesi Pınarhisar MYO	2017-devam

ESERLER (Makaleler ve Bildiriler)

1. Mühürçü, G., and Külekci, M.K. (2020). Increasing Surface Coating Quality Using New Generation Current Waveforms. IEEE Access, vol. 8, pp. 147358–147369, doi: 10.1109/ACCESS.2020.3015429.
2. Köse, E., Mühürçü A., Mühürçü G., Özdemir M. (2018). Pi Parameter Optimization By Fairly Algorithm For Optimal Controlling Of A Buck Converter's Output State Variable. Sakarya University Journal Of Science (22)5, 1267-1273, <https://doi.org/10.16984/saufenbilder.327296>.
3. Köse, E., Mühürçü, A., Mühürçü, G., Aydoğan, E. (2017). Pi Parameter Optimization By Using Ant Colony Algorithm For Optimal Controlling Of A Buck Converter's Output Voltage. Çukurova Üniversitesi Mühendislik Mimarlık Fakültesi Dergisi (32)4, 153 - 162. ISSN: 1019-1011.
4. Köse, E., Mühürçü A., Mühürçü G., Temiz M.N. (2017). Optimal Control Of A Buck Converter By Using Pi Controller Based On Hsa Algorithm. 2nd International Mediterranean Science And Engineering Congress (Imsec 2017), Adana.

5. Mühürçü, A., Şentürk, A.İ., Köse, E., Mühürçü, G. (2017). The Design Modification Of A Mobile Binoculars For Efficiency Long Range Shellfire. 2nd International Mediterranean Science And Engineering Congress (Imsec 2017), Adana.
6. Mühürçü, G. Köse, E., Mühürçü, A., Kuyumcu, A. (2017). Parameter Optimization Of P₁ Controller By Pso For Optimal Controlling Of A Buck Converter's Output. Artificial Intelligence And Data Processing Symposium (Idap17), Malatya.
7. Mühürçü, G. Köse, E., Mühürçü, A., Yılmaz, A. (2017). P₁'s Parameter Optimization Based On Iwo For Optimal Controlling Of A Buck Converter's Output. Artificial Intelligence And Data Processing Symposium (Idap17), Malatya.
8. Köse, E., Mühürçü, G., Mühürçü, A., Sevim, B. (2017). Sfla Based Pi Parameter Optimization For Optimal Controlling Of A Buck Converter's Voltage. Artificial Intelligence And Data Processing Symposium (Idap17), Malatya.
9. Kaptı, A., Mühürçü, G. (2014). Wearable Acceleration Sensor Application İn Unilateral Trans-Tibial Amputation Protheses, Biocybernetics And Biomedical Engineering (34)1, 53 - 62. <https://doi.org/10.1016/j.bbe.2013.10.002>.
10. Arslan, Ş., Mühürçü, G. (2014). Doğru Akım Elektrik Motorunun Doğrusal Karesel Regülatör ve Bulanık Mantık Kontrolörü ile Hız Kontrolü. Otomatik Kontrol Türk Milli Komitesi Ulusal Toplantısı 2014 (TOK 2014), Kocaeli.
11. Mühürçü, A., Mühürçü, G. (2010). Fuzzy Control Strategy Adapting To Ispm-15 Standarts. Electronic Letters On Science Engineering (6)2, 1 - 8. ISSN: 1305-8614.