T.C.
İSTANBUL ÜNİVERSİTESİ-CERRAHPAŞA
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ

YÜKSEK LİSANS TEZİ

ŞİFRELİ HABERLEŞME SİSTEMİNİN HAFIZALI DEVRE ELEMANI TABANLI KAOTİK DEVRE KULLANILARAK TASARIMI

Fatih SAYDAM

DANIŞMAN
Prof. Dr. Fırat KAÇAR

Elektrik- Elektronik Mühendisliği Anabilim Dalı

Elektrik-Elektronik Mühendisliği, Tezli Yüksek Lisans Programı

Mayıs, 2024

TEZ KABUL VE ONAYI

Fatih SAYDAM tarafından, **Prof. Dr. Fırat KAÇAR** danışmanlığında hazırlanan **"ŞİFRELİ HABERLEŞME SİSTEMİNİN HAFIZALI DEVRE ELEMANI TABANLI KAOTİK DEVRE KULLANILARAK TASARIMI"** başlıklı bu çalışma, jürimiz tarafından **05/06/2024** tarihinde yapılan sınav sonucunda **oy birliği** ile başarılı bulunarak **Yüksek Lisans Tezi** olarak kabul edilmiştir.

Tez Jürisi

		İmza	**Sonuç**
DANIŞMAN	Prof. Dr. Fırat KAÇAR İstanbul Üniversitesi-Cerrahpaşa Elektrik Elektronik Mühendisliği Anabilim Dalı		☒ Kabul ☐ Ret
ÜYE	Prof. Dr. Yasin ÖZÇELEP İstanbul Üniversitesi-Cerrahpaşa Elektrik Elektronik Mühendisliği Anabilim Dalı		☒ Kabul ☐ Ret
ÜYE	Dr. Öğr. Üyesi Mustafa KONAL Tekirdağ Namık Kemal Üniversitesi Elektrik Elektronik Mühendisliği Anabilim Dalı		☒ Kabul ☐ Ret

BEYAN

Bu tez çalışmasının kendi çalışmam olduğunu, tezin planlanmasından yazımına kadar bütün aşamalarda etik dışı davranışımın olmadığını, bu tezdeki bütün bilgileri akademik ve bilimsel etik kuralları içinde elde ettiğimi, bu tez çalışmasıyla elde edilmeyen bütün bilgi ve yorumlara kaynak gösterdiğimi ve bu kaynakları da kaynaklar listesine aldığımı, yine bu tezin çalışılması ve yazımı sırasında patent ve telif haklarını ihlal edici bir davranışımın olmadığını ve her türlü hukuki sorumluluğu aldığımı kabul ederim.

Fatih SAYDAM

(İmza)

Aileme ithaf ediyorum…

BÜTÇE DESTEKLERİ

ŞİFRELİ HABERLEŞME SİSTEMİNİN HAFIZALI DEVRE ELEMANI TABANLI KAOTİK DEVRE KULLANILARAK TASARIMI

Bu tez çalışması için herhangi bir kurumdan bütçe desteği alınmamıştır.

TEŞEKKÜR

Tez çalışmam boyunca geniş bilgi ve tecrübesiyle desteklerini sunan, farklı bakış açıları kazanmamı sağlayan değerli tez danışmanım Prof. Dr. Fırat KAÇAR'a en içten teşekkürlerimi sunarım.

Bütün hayatım boyunca desteklerini, yardımlarını ve sevgilerini benden esirgemeyen aileme teşekkür ederim.

Mayıs 2024 Fatih SAYDAM

İÇİNDEKİLER

Sayfa No

TEZ KABUL VE ONAYI ii
BEYAN iii
BÜTÇE DESTEKLERİ v
TEŞEKKÜR vi
İÇİNDEKİLER vii
ŞEKİL LİSTESİ ix
TABLO LİSTESİ xiii
SİMGE VE KISALTMA LİSTESİ xiv
ÖZET xvii
ABSTRACT xix
1. GİRİŞ 21
2. KAVRAMSAL ÇERÇEVE 24
2.1. HAFIZALI DEVRE ELEMANLARI 24
2.2. MEMRİSTÖRÜN YAPISI VE ÖZELLİKLERİ 25
2.3. MEMRİSTÖR EMÜLATÖRÜ 28
2.3.1. Yük-Bağımlı Memristör Emülatörü 28
2.3.2. Akı-Bağımlı Memristör Emülatörü 29
2.3.3. Memristör Emülatörü Analizi 30
2.4. KAOTİK SİSTEMLER VE DEVRELER 31
2.4.1. Kaotik Devreler 33
2.4.2. Kaotik Sistemlerin Analiz Yöntemleri 37
2.5. KAOTİK HABERLEŞME 41
2.5.1. Rastgele Sayı Üreteçleri (RSÜ) 41
2.5.2. Kaotik Şifreleme ve Şifre Çözme 42
2.5.3. Kaotik Maskeleme 43
3. YÖNTEM 44
3.1. AKTİF ELEMANLAR 44
3.1.1. Akım Kontrollü Akım Taşıyıcı Geçiş İletkenliği Kuvvetlendiricisi (CC-CCTA) 44

3.1.2. İşlemsel Geçiş İletkenliği Kuvvetlendirici (OTA) 45
3.1.3. Analog Çarpıcı 46
3.2. ÖNERİLEN MEMRİSTÖR EMÜLATÖRÜ 46
3.2.1. Topraklanmış Memristör Emülatörü 46
3.2.2. Yüzen Memristör Emülatörü 48
3.2.3. Parazitik Analiz 49
3.3. KÜBİK FORM 52
3.4. KAOTİK JERK DEVRESİ 53
3.5. KAOTİK CHUA DEVRESİ 55
3.6. KAOTİK HABERLEŞME SİSTEMİ TASARIMI 57
3.6.1. Kaos Tabanlı RSÜ Tasarımı 57
3.6.2. Kaotik Görüntü Şifreleme Uygulaması 58
3.6.3. Kaotik Maskeleme Uygulaması 59
4. BULGULAR 62
4.1. MEMRİSTÖR EMÜLATÖRÜ ANALİZİ 62
4.1.1. Zaman Analizi ve Histerisiz Eğrileri 62
4.1.2. İleri Analizler 65
4.2. KARŞILAŞTIRMA 68
4.3. MEMRİSTÖR TABANLI KAOTİK DEVRE 69
4.3.1. Önerilen Memristör Emülatörü ile Jerk Devresi Tasarımı 69
4.3.2. Önerilen Memristör Emülatörü ile Chua Devresi Tasarımı 73
4.4. KAOTİK HABERLEŞME SİSTEMİ TASARIMI SONUÇLARI 77
4.4.1. Kaotik Görüntü Şifreleme Uygulama Sonuçları 77
4.4.2. Kaotik Maskeleme Uygulama Sonuçları 80
5. TARTIŞMA 82
6. SONUÇ VE ÖNERİLER 84
KAYNAKLAR 86
İNTİHAL RAPORU İLK SAYFASI 98
ETİK KURUL İZİN YAZISI 99
KURUM İZNİ YAZILARI 100
ÖZGEÇMİŞ 101

ŞEKİL LİSTESİ

Sayfa No

Şekil 2.1: Hafızalı devre elemanları ile temel devre elemanları arasındaki ilişki. ……24

Şekil 2.2: Memristör sembolü. ……26

Şekil 2.3: Memristör histerisiz grafiği örneği (a) ve histerisiz grafiğinin frekansa bağlı değişimi (b). ……26

Şekil 2.4: Fiziksel memristörün yapısı. ……27

Şekil 2.5: Bir memristörün akım-zaman ve gerilim-zaman grafikleri örneği. ……..31

Şekil 2.6: Memristör histerisiz eğrisi örneği. ……31

Şekil 2.7: Lorenz çekicisi faz portreleri. ……..32

Şekil 2.8: Üç elemanlı Chua devresi. ……34

Şekil 2.9: Farklı kaotik Chua devreleri. ……34

Şekil 2.10: Shinriki devresi. ……35

Şekil 2.11: Jerk devresi [33]. ……36

Şekil 2.12: Rossler sistemi Lyapunov üstelleri. ……39

Şekil 2.13: Rössler sisteminin çatallanma diyagramı; a parametresine göre y fonksiyonun değişimi. ……...40

Şekil 2.14: Rössler sisteminden elde edilen faz portreleri, (a): x-y grafiği, (b): x-z grafiği, (c): y-z grafiği. ……41

Şekil 2.15: Rastgele sayı üreteci blok diyagramı [104]. ……42

Şekil 2.16: Kaotik maskeleme blok diyagramı [106]. ……..43

Şekil 3.1: CC-CCTA iç yapısı. ……45

Şekil 3.2: CC-CCTA blok diyagramı. ……..45

Şekil 3.3: OTA'nın iç yapısı. ……...…….45

Şekil 3.4: Analog çarpıcı iç yapısı. ……...46

Şekil 3.5: Önerilen topraklanmış memristör emülatörü devresi. …….47

Şekil 3.6: Önerilen yüzen memristör emülatörü devresi. ……..49

Şekil 3.7: CC-CCTA'nın parazitik bileşenleri. ……..……..50

Şekil 3.8: Parazitik bileşenleri içeren memristör emülatörü. ……50

Şekil 3.9: Kübik forma getirilmiş memristör emülatörü devresi. …….52

Şekil 3.10: Kaotik Chua devresi. ……..55

Şekil 3.11: RSÜ tasarımı akış şeması. ……..57

Şekil 3.12: Kaotik görüntü şifreleme işleminin blok diyagramı. ……..59

Şekil 3.13: Kaotik şifre çözme işleminin blok diyagramı. ……60

Şekil 3.14: Kaotik maskeleme işleminin blok diyagramı [107]. ……..61

Şekil 4.1: Memristör emülatörünün zaman cevabı (a) ve histerisiz eğrisi (b). ……..63

Şekil 4.2: 100 kHz frekansta farklı kapasitör değerleri için elde edilen histerisiz eğrileri. ……63

Şekil 4.3: 100 kHz frekansta farklı bayas akımı değerleri için elde edilen histerisiz eğrileri. ...64

Şekil 4.4: Farklı giriş frekansları ile elde edilen histerisiz eğrileri, $C_1 = 50\ \mu F$. ……64

Şekil 4.5: Farklı giriş frekansları ile elde edilen histerisiz eğrileri, $C_1 = 1\ nF$. ……65

Şekil 4.6: Farklı giriş frekansları ile elde edilen histerisiz eğrileri, $C_1 = 10\ pF$. ……65

Şekil 4.7: Farklı sıcaklıklarda elde edilen histerisiz eğrileri. ……..…….66

Şekil 4.8: sıcaklık değişimleriyle besleme gerilimindeki ±%5'lik bir değişim için elde edilen histerezis eğrisi. ……66

Şekil 4.9: Memristör emülatörünün Monte Carlo analizinden elde edilen bulgular, %10'luk değişimde kapasitör değerlerinde tutarlı değişikliklerle yüz kez yinelemeli olarak gerçekleştirildi. ……67

Şekil 4.10: Memristör emülatörü üzerinde gerçekleştirilen Monte Carlo analizinin sonuçları, besleme gerilimleri %10'luk bir değişim içeren yüz tekrar ile uygulanmıştır. ……...67

Şekil 4.11: Kaotik Jerk devresi [33]. ……69

Şekil 4.12: Jerk devresinden elde edilen V_1, V_2 ve V_3 gerilimlerinin zamana göre değişimleri (a) ve faz portreleri (b), (c), (d). ……..70

Şekil 4.13: R_6 ve R_8 dirençleri çıkarılmış Jerk devresinden elde edilen V_1, V_2 ve V_3 gerilimlerinin zamana göre değişimleri (a) ve faz portreleri (b), (c), (d). ……71

Şekil 4.14: Sunulan Jerk devresinde farklı bayas akımlarıyla, V_1 ve V_2 gerilimlerinin zamana bağlı değişimleri (a) ve bu iki gerilimin arasındaki ilişkiden elde edilen, farklı bayas akımlarındaki faz portreleri (a), (b), (c). ……71

Şekil 4.15: Sunulan Jerk devresinde farklı bayas akımlarıyla, V_1 ve V_3 gerilimlerinin zamana bağlı değişimleri (a) ve bu iki gerilimin arasındaki ilişkiden elde edilen, farklı bayas akımlarındaki faz portreleri (a), (b), (c). ……72

Şekil 4.16: Sunulan Jerk devresinde farklı bayas akımlarıyla, V_2 ve V_3 gerilimlerinin zamana bağlı değişimleri (a) ve bu iki gerilimin arasındaki ilişkiden elde edilen, farklı bayas akımlarındaki faz portreleri (a), (b), (c). ……72

Şekil 4.17: Önerilen memristörün kullanılacağı Chua devresi. ……73

Şekil 4.18: Önerilen sistemin Lyapunov üstelleri. ……75

Şekil 4.19: *r* parametresinin 0-5 aralığında değiştirilmesiyle oluşan y fonksiyonundaki değişimleri gösteren çatallanma diyagramı. ……..76

Şekil 4.20: Chua devresinde zamana bağlı değişim grafikleri: *x* (a), *y* (b), *z* (c) ve *w* (d). ……..76

Şekil 4.21: Chua devresi faz porteleri: *x*-*y* (a), *x*-*z* (b), *x*-*w* (c), *y*-*z* (d), *y*-*w* (e) ve *z*-*w* (f). …….77

Şekil 4.22: Satır karıştırma ile şifreleme işlemi. Görüntü (a) ve şifrelenmiş görüntü (b). …….78

Şekil 4.23: Sütun karıştırma ile şifreleme işlemi. Görüntü (a) ve şifrelenmiş görüntü (b). ……78

Şekil 4.24: *K1* anahtarı kullanılarak *XOR* işlemi ile gerçekleştirilen şifreleme işlemi. Görüntü (a) ve şifrelenmiş görüntü (b). ……...79

Şekil 4.25: *K2* anahtarı kullanılarak *XOR* işlemi ile gerçekleştirilen şifreleme işlemi. Görüntü (a) ve şifrelenmiş görüntü (b). ……...79

Şekil 4.26: Önerilen şifreleme algoritması ile gerçekleştirilen şifreleme ve şifre çözme işlemi. Görüntü (a), şifrelenmiş görüntü (b) ve şifresi çözülmüş görüntü (c). ……...80

Şekil 4.27: MATLAB Simulink programında kaotik maskeleme işlemi. ……..……..80

Şekil 4.28: MATLAB Simulink programında kaotik maskeleme işleminin sonuçları: bilgi işareti (a), modüle edilmiş işaret (b), demodüle edilmiş işaret (c) ve hata işareti (d). ……81

TABLO LİSTESİ

Sayfa No

Tablo 3.1: Parazitik bileşenlerin değerleri. …….51

Tablo 4.1: Transistör boyutları. ……...62

Tablo 4.2: Memristör emülatörleri karşılaştırma tablosu. ……...68

Tablo 4.3: Jerk devresi parametre değerleri. ……...70

Tablo 4.4: Chua devresi parametre değerleri. ……..73

SİMGE VE KISALTMA LİSTESİ

Simgeler	Açıklama
V	: Gerilim.
I	: Akım.
q	: Yük.
φ	: Akı.
g_m	: Geçişiletkenliği.
R_X	: Parazitik direnç.
M_R	: Memristans.
M_C	: Kapasitans.
M_L	: İndüktans.
i_m	: Memristör akımı.
v_m	: Memristör gerilimi.
λ	: Özdeğer.
I	: Birim matris.
J	: Jacobian matris.
μ_n	: Elektron hareketliliği.
C_{OX}	: Oksit kapasitans.
I_{D16}	: M_{16} transistöründen akan akım.
W/L	: Kanal genişliği ve kanal boyu oranı.
V_{in}	: Memristör emülatörü giriş gerilimi.
i_{in}	: Memristör emülatörü giriş akımı.
V_y	: Memristör emülatörü Y portu gerilimi.
V_x	: Memristör emülatörü X portu gerilimi.
V_Z	: Memristör emülatörü Z portu gerilimi.
V_B	: Memristör emülatörü B portu gerilimi.
I_O	: Memristör emülatörü I portu akımı.
V_{SS}	: Kaynak gerilimi.
V_{DD}	: Savak gerilimi.
V_{tn}	: NMOS eşik gerilimi.

C_1 : Memristör emülatörü kapasitörü.

R_O : Memristör emülatörü çıkış parazitik direnci.

w : Açısal frekans.

C_O : Memristör emülatörü çıkış parazitik kapasitörü.

j : Karmaşık sayı.

w_L : Alçak seviyedeki açısal frekans.

w_H : Yüksek seviyedeki açısal frekans.

R_Y : Memristör emülatörü Y portundaki parazitik direnci.

R_Z : Memristör emülatörü Z portundaki parazitik direnci.

R_X : Memristör emülatörü X portundaki parazitik direnci.

C_Y : Memristör emülatörü Y portundaki parazitik kapasitörü.

C_Z : Memristör emülatörü Z portundaki parazitik kapasitörü.

v_{C1} : Chua devresinde C_1 kapasitörü üzerine düşen gerilim.

v_{C2} : Chua devresinde C_2 kapasitörü üzerine düşen gerilim.

i_L : Chua devresinde L indüktörü üzerinden geçen akım.

K : Kaotik anahtarlar.

I_B : Bayas akımı.

LE : Lyapunov üstelleri.

Kısaltmalar	**Açıklama**
MOSFET	: Metal-Oxide Field Effect Transistor.
OP-AMP	: Operational Amplifier.
CCII	: Second-generation Current Conveyor.
CCTA	: Current Conveyor Transconductance Amplifier.
CC-CCTA	: Current-Controlled Current Conveyor Transconductance Amplifier.
OTA	: Operational Transconductance Amplifier.
BJT	: Bipolar Junction Transistor.
VDTA	: Voltage Differencing Transconductance Amplifier.
NIC	: Negative Impedance Converter.
NMOS	: N tipi Metal Oksit Yarıiletken.

XOR	: Özel veya.
TiO_2	: Titanyumdioksit.
RSÜ	: Rastgele Sayı Üreteçleri.
CNTFET	: Carbon nanotube field-effect transistor.
CFTA	: Current follower transconductance amplifier.
VDIBA	: Voltage differencing inverting buffered amplifier.
DVCC	: Differential voltage current conveyor.
CFDITA	: Current follower differential input transconductance amplifier.

ÖZET

YÜKSEK LİSANS TEZİ

ŞİFRELİ HABERLEŞME SİSTEMİNİN HAFIZALI DEVRE ELEMANI TABANLI KAOTİK DEVRE KULLANILARAK TASARIMI

Fatih SAYDAM

İstanbul Üniversitesi-Cerrahpaşa

Lisansüstü Eğitim Enstitüsü

Elektrik- Elektronik Mühendisliği Anabilim Dalı

Elektrik-Elektronik Mühendisliği, Tezli Yüksek Lisans Programı

Danışman : Prof. Dr. Fırat KAÇAR

Bu çalışmada topraklanmış ve yüzen olmak üzere, direnç içermeyen memristör emülatörü tasarımı gerçekleştirilmiştir. Tasarlanan topraklanmış memristör emülatörü, aktif eleman olarak bir adet CC-CCTA içermektedir. Devrede kullanılan pasif eleman ise bir adet kapasitördür. Yüzen memristör emülatörü, bu devre elemanlarına ek olarak bir adet OTA içermektedir. Sunulan memristör emülatörü yüksek frekanslarda çalışma özelliğine sahiptir ve elektronik olarak kontrol edilebilir özelliktedir. Memristör emülatörünün performansını incelemek adına gerçekleştirilen benzetim çalışmaları, TSMC 180 nm teknolojisi kullanılarak LTspice programı ile yapılmıştır. Daha sonra önerilen memristör emülatörü, kaotik sistemlerde daha yüksek doğrusal olmama özelliğine sahip olabilmesi için kübik forma dönüştürülmüştür. Kübik formdaki memristör emülatörü için devreye ek olarak bir adet çarpıcı yapısı eklenmiştir.

Çalışma kapsamında kullanılmak üzere tasarlanan memristör emülatörleri sunulduktan sonra bu devreler kullanılarak kaotik sistem tasarımı gerçekleştirilmiştir. İlk olarak sunulan memristör emülatörü kullanılarak kaotik Jerk sistemi oluşturulup, bu sistemin matematiksel analizleri yapıldıktan sonra, LTspice programı kullanılarak kaotik davranışı gözlemlenmiştir.

Sunulan kübik formdaki memristör emülatörü ise Chua devresinde kullanılmıştır. Chua devresi için matematiksel analizler yapıldıktan sonra kaotik davranışları incelenmek üzere MATLAB programı kullanılmıştır. Sonuç olarak tasarlanan iki kaotik sistem de önerilen memristör emülatörleri kullanılarak gerçekleştirilmiş olup 4 boyutlu denklemlere sahiptir.

Çalışmanın diğer aşamasında bu kaotik sistemler kullanılarak bir haberleşme sistemi tasarımı gerçekleştirilmiştir. Chua devresi temelli kaotik sistem kullanılarak bir adet rastgele sayı üreteci tasarımı gerçekleştirilmiştir. Daha sonra bu rastgele sayı üreteci kullanılarak bir kaotik şifreleme algoritması geliştirilmiştir. Bu algoritmanın uygulaması MATLAB programı kullanılarak gerçekleştirilmiştir. Şifreleme ve şifre çözme adımları açıklanmıştır. Jerk sistemi kullanılarak gerçekleştirilen uygulama ise kaotik maskeleme işlemidir. Bu işlem için MATLAB Simulink uygulaması kullanılarak benzetim çalışmaları gerçekleştirilmiştir. Sonuç olarak tez kapsamında memristör emülatörü tasarımı gerçekleştirilip bu memristör emülatörü kullanılarak iki farklı kaotik sistem sunulmuştur ve daha sonra bu kaotik sistemler kullanılarak kaotik haberleşme sistemi tasarımı gerçekleştirilmiştir.

Mayıs 2024 , 101 sayfa.

Anahtar kelimeler: Memristör, Kaotik sistem, Kaotik haberleşme

ABSTRACT

M.Sc. THESIS

ENCRYPTED COMMUNICATION SYSTEM DESIGN USING MEMORY CIRCUIT ELEMENT BASED CHAOTIC CIRCUIT

Fatih SAYDAM

İstanbul University-Cerrahpaşa

Institute of Graduate Studies

Department of Electrical and Electronics Engineering

Electrical and Electronics Engineering, Master's Thesis

Supervisor : Prof. Dr. Fırat KAÇAR

In this study, we designed a resistor-free memristor emulator in both grounded and floating configurations. The grounded memristor emulator consists of a single CC-CCTA as the active element, while the passive element is a capacitor. The floating memristor emulator incorporates an OTA in addition to these circuit elements. Our memristor emulator is capable of high-frequency operation and is electronically controllable. Simulation studies to evaluate its performance were conducted using the LTspice program with TSMC 180 nm technology.

Subsequently, we converted the proposed memristor emulator into cubic form to introduce higher nonlinearity in chaotic systems. This cubic form memristor emulator was then integrated into the design of chaotic systems. Firstly, a chaotic Jerk system was created using the memristor emulator, and its chaotic behavior was observed using the LTspice program after mathematical analysis. Similarly, the cubic form memristor emulator was employed in the Chua circuit, and its chaotic behavior was analyzed using MATLAB.

In another stage of the study, we designed a communication system based on these chaotic systems. We developed a random number generator using the chaotic system derived from the Chua circuit and subsequently designed a chaotic encryption algorithm based on this generator. We implemented this algorithm using MATLAB, detailing the encryption and decryption steps. Additionally, we demonstrated a chaotic masking process using the jerk system, conducting simulation studies using MATLAB Simulink.

Overall, our study encompasses the design of a memristor emulator, the presentation of two chaotic systems employing this emulator, and the realization of a chaotic communication system leveraging these chaotic systems.

May 2024, 101 pages.

Keywords: Memristor, Chaotic system, Chaotic communication

1. GİRİŞ

1971 yılında Leon O. Chua hafızalı direnç olarak bilinen, doğrusal olmayan özellik gösteren bir pasif devre elemanı olan Memristör fikrini ortaya atmıştır [1]. Chua, gerilim (V), akım (I), yük (q) ve akı (φ) arasındaki matematiksel bağıntılar incelendiğinde, yük ve akı ifadelerinin arasındaki ilişkiden de bir devre elemanı ortaya çıkabileceğini düşünmüştür. Bu sebeple bu devre elemanını "kayıp devre elemanı" olarak isimlendirmiştir. Memristör fikrinin ortaya atılmasının ardından uzun yıllar bu konudaki çalışmalar geliştirilmiştir, fakat ilk olarak 2008 yılında Strukov ve arkadaşları [2] tarafından gerçek memristör elemanı deneysel olarak gerçekleştirilmiştir. Bu çalışma memristörle ilgili çalışmaların artmasına sebep olmuştur. Memristörün yanında diğer hafızalı devre elemanları üzerine de çalışmalar gerçekleştirilmiştir. Bu devre elemanları Memkapasitör (hafızalı kapasite) ve Memindüktör (hafızalı indüktör) olarak isimlendirilmiş olup [3], bu konularda çeşitli emülator devreleri ve benzetimler literatüre sunulmuştur [4-8]. Hafızalı devre elemanları kriptoloji [9], haberleşme sistemleri [10], [11] ve nöral sistemler [12], [13] gibi birçok alanda kullanılmaktadır.

Hafızalı devre elemanları ile ilgili çalışmaların çoğunluğu MOSFET veya MOSFET tabanlı aktif elemanlar kullanılarak gerçekleştirilmiş emülatör devrelerini ve bu devreler kullanılarak gerçekleştirilen uygulamaları kapsamaktadır. Memristör emülatör uygulamaları ilk olarak ideal Spice modelleri oluşturularak gerçekleştirilmiştir [14], [15]. Bu çalışmalar benzetim ortamı için uygun olsa da donanım açısından fiziksel olarak gerçekleştirilebilir yapılar değillerdir. Daha sonra 2010 yılında, Mutlu ve arkadaşları [16] donanımsal olarak kullanıma uygun memristör emülatör devresi önermişlerdir. Bu çalışmadan sonra Kim ve arkadaşları [17] benzer matematiksel denklemlere ulaşılan memristör emülatör devresi geliştirmişlerdir. Ardından günümüze kadar birçok memristör emülatör devresi tasarımı gerçekleştirilmiştir. Kullanılan aktif ve pasif eleman sayısı, devrenin yüzen veya topraklanmış oluşu, çalışma frekansı, elektronik olarak kontrol edilebilirliği, gerçekleştirilen uygulamalar vb. gibi özellikler memristör emülatör devrelerinin kalitesi hakkında belirleyici olabilmektedir. Yıllar içerisinde bu ölçeklere göre optimum tasarımlar elde edilmeye çalışılmıştır ve çeşitli matematiksel denklemler elde edilerek memristör tasarımları gerçekleştirilmiştir.

Literatürdeki memristör emülatör devreleriyle, aktif filtre tasarımı [18-20], Schmitt tetikleycisi [19], [21], [22], modülasyon devreleri [23-25], mantık devreleri [22], [25] ve kaotik sistemler [26-33] gibi çeşitli uygulamalar gerçekleştirilmiştir. Bu uygulamalardan tez kapsamında daha çok dikkate alınacak kısım kaotik devre tasarımı konusu olacaktır. Memristörlerin lineer olmayan yapısından kaynaklı olarak kaotik uygulamalardaki kullanımları oldukça önemlidir. Memristör kullanılarak, geleneksel Chua devresi [34], Jerk sistemleri [35], Shinriki'nin devresi [36] gibi yapılardan faydalanılarak kaotik davranıştaki sistemlerin tasarımını gerçekleştirmek mümkündür. Kaotik davranışın test edilmesi amacıyla, faz portresi analizi, çatallanma diyagramı, Lyapunov üstelleri gibi çeşitli analiz yöntemleri gerçekleştirilmektedir. Bu gibi analizlerle elde edilmiş denklemlerin kaotik oluşu tespit edildikten sonra bu kaotik sistem kullanılarak görüntü şifreleme [9] ve haberleşme sistemleri [11] gibi çeşitli konularda uygulamalar gerçekleştirilebilmektedir.

Memristör emülatör devresi kullanılarak yapılan kaotik çalışmaların çoğunluğu Chua devresi kullanılarak gerçekleştirilmiştir. 2008 yılında Itoh ve Chua [34], Chua diyotu yerine memristör kullanarak kaotik uygulamalar gerçekleştirmişlerdir. Daha sonra, Muthuswamy [37], çalışmasında kübik formda doğrusal olmayan fonksiyonları kullanarak aynı şekilde Chua diyotu yerine memristör kullanarak kaotik uygulama gerçekleştirmiştir. Bu çalışmalar teorik düzeyde kalmış olup gerçek veya emülatör şeklinde memristör yapısı içermemektedir. İlerleyen çalışmalarda memristör tasarımı gerçekleştirilerek de kaotik sistemler tasarlanmaya başlanmıştır. Zhao ve arkadaşları [38] çalışmalarında, memristör, memindüktör ve memkapasitör olmak üzere 3 farklı hafıza elemanı tasarımı gerçekleştirerek Chua devresi ile kaotik sistem tasarlamışlardır. Bir başka memristör emülatörü tabanlı Chua devresi kullanılmış kaotik devre tasarımı Ranjan ve arkadaşları [39] tarafından gerçekleştirilmiştir. Bu çalışmadaki memristör emülatör devresinde bir adet ikinci nesil akım taşıyıcı (CCII), bir adet akım taşıyıcı geçiş iletkenliği kuvvetlendiricisi (CCTA), üç adet direnç ve bir adet kapasitör kullanılmıştır. Böylesi memristör emülatörü ile Chua devresi kullanılarak tasarımı gerçekleştirilmiş kaotik sistemlerin literatürdeki örnekleri artırılabilmektedir [40-43].

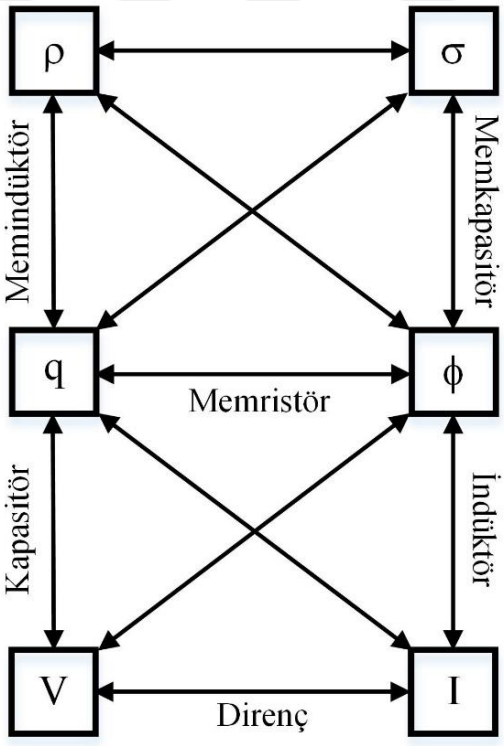
Bu çalışma kapsamında topraklanmış ve yüzen formda olmak üzere yeni memristör emülatör devresi tasarımı sunulmuştur. Tasarlanan topraklanmış devre bir adet akım kontrollü akım taşıyıcı geçiş iletkenliği kuvvetlendiricisi (CC-CCTA) ve bir adet kapasitör kullanılarak gerçekleştirilmiştir. Yüzen memristör emülatörünün eldesinde ise bu elemanlara ek olarak bir adet işlemsel geçiş iletkenliği (OTA) devreye ilave edilmiştir. Tasarlanan bu devrelerde,

kullanılan aktif elemanların iç yapılarındaki geçiş iletkeniliği (g_m) ve parazitik direnç (R_X) sayesinde harici bir dirence gereksinim duyulmamıştır. Bu devreler kullanılarak öncelikle kaotik Jerk sistemi tasarımı gerçekleştirilmiş olup bu kapsamda LTspice programı kullanılarak çeşitli benzetimlerle faz portreleri elde edilmiştir. Daha sonra bu devrelere bir adet çarpıcı eklenerek gerçekleştirilen yeni bir tasarımla kübik forma getirilmiştir. Kübik formdaki denklemler sayesinde Chua devresinde kullanılmasıyla elde edilebilecek kaosun artması mümkün olmaktadır [37]. Bu çalışma kapsamında sunulan kübik formdaki memristör devresi ile tasarlanmış olan kaotik Chua devresinin kaotikliğini incelemek amacıyla, Lyapunov üstelleri, faz portreleri ve çatallanma diyagramları Matlab programı vasıtasıyla elde edilmiştir. Bu şekilde kaotik uygulamalarda kullanılabilirliği ispatlanmış olan bu sistem ile, elde edilmiş denklemlerden faydalanılarak, kaotik şifrelenmiş görüntünün haberleşmede kullanılması yine Matlab programı kullanılarak gerçekleştirilmiştir. Daha sonra önerilen kaotik sistem kullanılarak kaotik maskeleme uygulaması gerçekleştirilmiştir.

2. KAVRAMSAL ÇERÇEVE

2.1. HAFIZALI DEVRE ELEMANLARI

Hafızalı devre elemanları üçe ayrılmaktadır. Bunlar memristör, memkapasitör ve memindüktör olarak isimlendirilir. Doğrusal olmayan yapıdaki bu devre elemanları pasif özellik göstermektedir. Hafızalı devre elemanı ilk olarak 1971 yılında Chua [1] tarafından ortaya atılmıştır. Chua, gerilim ve akım arasındaki bağıntıdan direnç, gerilim ve yük arasındaki bağıntıdan kapasitör, akım ve akı arasındaki bağıntıdan indüktör elde edildiğini; ancak akı ve yük arasındaki bağıntıyla ilişkili de bir devre elemanı bulunması gerektiğini düşünmüştür. Böyle bir devre elemanı olsaydı nasıl bir özelliğe sahip olacağını inceleyip, hafızalı bir direnç özelliği gösteren doğrusal olmayan bir yapıya ulaşmıştır. Sonuç olarak bu devre elemanını memristör olarak isimlendirmiştir. Zaman içerisinde memristör konusuna ilgi artmış olup, bu konuda sayısız çalışmalar gerçekleştirilmiştir. Memkapasitör ve memindüktör fikri de bu şekilde geliştirilmiştir. Hafızalı devre elemanları ile temel devre elemanları arasındaki ilişki Şekil 2.1'de [44] gösterilmiştir.



Şekil 2.1: Hafızalı devre elemanları ile temel devre elemanları arasındaki ilişki.

Şekil 2.1'de verilen parametreler arasındaki matematiksel ilişki aşağıdaki gibi verilmiştir.

$$R = \frac{dV}{dI} \tag{2.1}$$

$$C = \frac{dq}{dV} \tag{2.2}$$

$$L = \frac{d\varphi}{dI} \tag{2.3}$$

Buradan hareketle memristans (M_R), memkapasitans (M_C) ve memindüktans (M_L) ifadeleri aşağıdaki gibi tanımlanabilmektedir.

$$M_R = \frac{d\varphi}{dq} \tag{2.4}$$

$$M_C = \frac{d\sigma}{d\varphi} \tag{2.5}$$

$$M_L = \frac{d\rho}{dq} \tag{2.6}$$

Şekil 2.1'de çapraz oklarla gösterilen parametreler arasındaki ifadeler incelenecek olursa,

$$\varphi = \int_{t_0}^{t} V(\tau)\, d\tau \tag{2.7}$$

$$q = \int_{t_0}^{t} I(\tau)\, d\tau \tag{2.8}$$

$$\rho = \int_{t_0}^{t} \varphi(\tau)\, d\tau \tag{2.9}$$

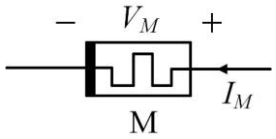
$$\sigma = \int_{t_0}^{t} q(\tau)\, d\tau \tag{2.10}$$

denklemleri elde edilmektedir.

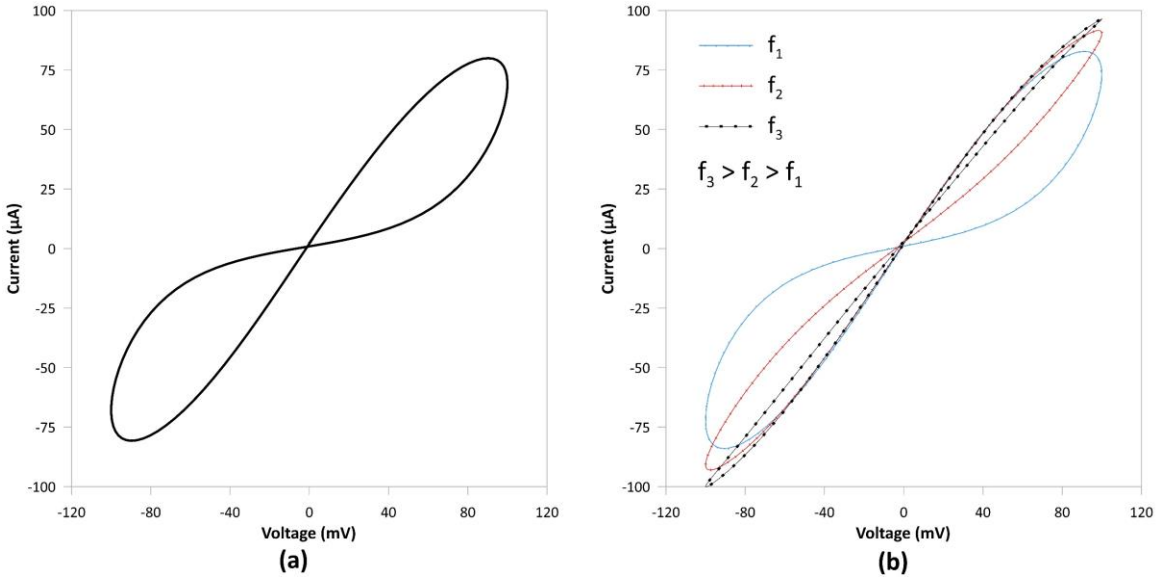
2.2. MEMRİSTÖRÜN YAPISI VE ÖZELLİKLERİ

Memristör, hafızalı direnç özelliği gösteren, doğrusal olmayan, pasif bir devre elemanıdır [1]. Memristör kesilmesi, hafızalı direnç anlamına gelen "memory resistor" ifadesinin kısaltmasıdır. Memristörün sembolü Şekil 2.2'de verilmiştir. Bir devre içerisinde kullanılmak üzere üretilen gerçek memristör ilk olarak 2008 yılında Hewlett Packard laboratuvarında gerçekleştirilmiştir [2]. Bu çalışma sayesinde memristöre olan ilgi artmış olup, bu konularda yapılan çalışmalar hız kazanmıştır. Çeşitli memristör modelleri oluşturulmuş olup bu modellerin birbirinden farklı

matematiksel izahları yapılmıştır. Bu matematiksel denklemlerin ortak noktası doğrusal olmayan yapıda olmasıdır. Ayrıca bir memristörün üzerinden geçen akım ile üzerindeki gerilim arasındaki ilişkiden Şekil 2.3a'daki gibi bir histerisiz grafiği elde edilmelidir. Devrenin frekans değerinin artmasıyla histerisiz eğrisinde gerçekleşen değişim Şekil 2.3b'de [45] verilmiştir.



Şekil 2.2: Memristör sembolü.



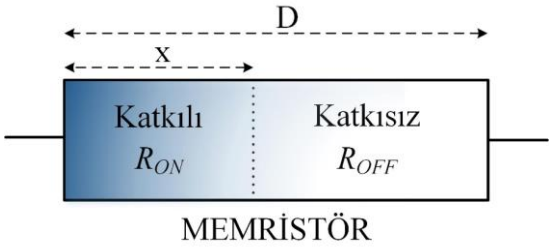
Şekil 2.3: Memristör histerisiz grafiği örneği (a) ve histerisiz grafiğinin frekansa bağlı değişimi (b).

Memristörün gösterdiği hafızalı direnç özelliği "memristans" olarak isimlendirilmiştir. *x* olarak isimlendirilmiş bir değişkene bağlı olan memristans ifadesi *M(x)* olarak gösterilirse [45],

$$v = M(x)i \tag{2.11}$$

$$\frac{dx}{dt} = f(x, i) \tag{2.12}$$

elde edilmektedir. Buradaki *x* değişkenine bağlı olan memristans değeri somut üretimi gerçekleştirilen memristörün [2] yapısından dolayı, bir direncin açık (ON) veya kapalı (OFF) olması gibi bir anahtarlama özelliği gösterdiği görülmektedir. Bu durum Şekil 2.4'teki görselle izah edilmeye çalışılmıştır.



Şekil 2.4: Fiziksel memristörün yapısı.

Şekildeki katkılama işlemi titanyumdioksit (TiO_2) ile gerçekleştirilmiştir. Denklem (2.11)'de değişken olarak verilen "x" ifadesi, katkılı kısmın genişliğini belirtmektedir ve bu genişlik arttıkça memristörün iletkenliği artmaktadır [46]. Buradan memristans değeri,

$$M(x) = \frac{x(t)R_{ON}}{D} + \frac{(1-x(t))R_{OFF}}{D} \tag{2.13}$$

olarak elde edilir. Bu ifade ile denklem (2.11) tekrar düzenlenecek olursa memristör gerilimi,

$$v = \left(\frac{x(t)R_{ON}}{D} + \frac{(1-x(t))R_{OFF}}{D}\right) i \tag{2.14}$$

olarak elde edilir. Daha sonra memristans ifadesi yüke (q) bağlı bir fonksiyon olarak;

$$\frac{dx}{dt} = \mu \frac{R_{ON}}{D} i \tag{2.15}$$

$$x = \mu \frac{R_{ON}}{D} q \tag{2.16}$$

$$M(q) = R_{OFF}\left(1 - \mu \frac{R_{ON}}{D^2} q\right) \tag{2.17}$$

şeklinde elde edilir [2].

Memristörü iki farklı formda doğrusal olmayan fonksiyon ile göstermek mümkündür. Bunlardan biri daha önce de bahsi geçen Memristans ($M(q)$) iken, diğeri akıya bağlı fonksiyon olan Memdüktans ($W(\varphi)$) [37] fonksiyonudur. Bu fonksiyonlar aşağıdaki gibi tanımlanmaktadır [47].

$$M(q) = \frac{d\varphi}{dq} \tag{2.18}$$

$$M(q) i_M = v_M \tag{2.19}$$

$$W(\varphi) = \frac{dq}{d\varphi} \tag{2.20}$$

$$W(\varphi)v_M = i_M \tag{2.21}$$

Verilen denklemlerde i_M, memristör üzerinden akan akımı; v_M, memristör üzerindeki gerilimi belirtmektedir. Bu ifadelerden hareketle çeşitli memristans ve memdüktans denklemleri elde edilerek memristör tasarımı gerçekleştirilebilmektedir.

2.3. MEMRİSTÖR EMÜLATÖRÜ

Memristör elemanının işlevini gerçekleştiren devre yapıları memristör emülatörü olarak isimlendirilmektedir. Memristör emülatörü devreleri genellikle MOSFET tabanlı aktif elemanlar kullanılarak gerçekleştirilmektedir. Bu aktif elemanlar kullanılarak Bölüm 2.2.'de verilen memristans ve memdüktans değerlerini veren devre tasarımları gerçekleştirilerek memristör emülatörü gerçekleştirilmiş olur. Bu bağlamda yük-bağımlı ve akı-bağımlı olmak üzere iki tipte memristör emülatörü incelenebilmektedir.

2.3.1. Yük-Bağımlı Memristör Emülatörü

Memristör emülatörü tasarımında yüke bağlı olarak değişen memristans ifadesinin elde edilmesiyle yük-bağımlı memristör tasarımı gerçekleştirilmektedir. Elwakil ve arkadaşları [48], iki CCII, bir analog çarpıcı, üç direnç ve bir kapasitör kullanarak gerçekleştirdikleri yük-bağımlı memristör emülatörüyle,

$$M(q) = R_S \pm \frac{R_S}{TI_{REF}} q(t) \tag{2.22}$$

ifadesini elde etmişlerdir. Burada memristans denkleminde verilen, R_S, T ve I_{REF} değerleri tasarlanmış olan denklem ile elde edilmiş olan parametrelerdir. Farklı tasarımlarla farklı katsayı değerleri elde edilebilmektedir. Örnek olarak Alharbi ve arkadaşları [49], sundukları memristör emülatör devresinde memristans değeri olarak,

$$M(q) = R_S \pm \frac{I_S R_1}{C^2 V_{TH}^2 i_{in}} q(t) \tag{2.23}$$

denklemini elde etmişlerdir. Bu tasarımda aktif eleman olarak iki adet CCII ve iki adet BJT; pasif eleman olarak ise iki direnç ve bir kapasitör kullanmışlardır. Bu şekilde elde edilen memristans ifadeleri çoğaltılabilmektedir [50], [51]. Literatürdeki bazı yayınlarda benzer

denklemlerle farklı tasarımlar gerçekleştirilmiştir [52]-[54]. Bu noktada önemli olan, gerçekleştirilen tasarımın memristif özellikte olup, memristör tabanlı uygulamalarda kullanabilirliğinin olmasıdır.

2.3.2. Akı-Bağımlı Memristör Emülatörü

Memristör değerinin denkleminde yük yerine akı ifadesinin yer aldığı, genellikle memristans yerine memdüktans değerinin elde edilmesiyle memristif elemanın tasarlandığı yapılar akı-bağımlı memristör emülatörü olarak isimlendirilir. Son yıllarda tasarlanan memristör emülatörü devrelerinin birçoğu akıya bağlı olarak gerçekleştirilmiştir. Akı-bağımlı memristörün en önemli avantajı, genellikle tasarımda daha az sayıda harici pasif devre elemanı kullanılmasına olanak sağlamasıdır. Ancak akı-bağımlı memristör emülatörü tasarlamak, yük-bağımlıya göre daha karmaşık olabilmektedir. 2015 [55] ve 2016 [56] yıllarında yapılan çalışmalarda elde edilen memristif denklem, daha önceden bahsedilen yük-bağımlı memristör emülatörü devreleriyle ulaşılan denklemlerdekine benzer formda, ancak akıya bağlı olarak değişen memristans ifadesi elde edilmek üzere tasarlanmıştır. Bu tasarımlar oldukça karmaşık yapıda olup, çok sayıda aktif ve pasif eleman kullanılarak gerçekleştirilmiştir. Bu çalışmalara yakın zamanlarda bir başka akı bağımlı memristör emülatörü Alharbi ve arkadaşları [57] tarafından gerçekleştirilmiştir. Bahsi geçen diğer çalışmalara nispeten daha az karmaşık olan bu yapı, üç CCII, bir analog çarpıcı, üç direnç ve bir kapasitör kullanılarak tasarlanmıştır. Ayrıca deneysel olarak da tasarım test edilmiştir. Bu çalışmada elde edilen memdüktans denklemi:

$$W(\varphi) = \frac{1}{R_1} + \frac{\alpha R_3}{R_1 R_2^2 C}\varphi(t) \tag{2.24}$$

şeklindedir. Buradaki parametreler tasarlanan devrenin içeriğinden elde edilmiştir.

2018 yılında Yeşil [58], farklı bir akı-bağımlı memristör emülatörü sunmuştur. Bu tasarım yalnızca yedi adet transistör ve bir adet kapasitör içermektedir. Tasarım içerisindeki bir MOSFET'in kapı ucuna uygulanan değişken gerilim ve bu uca bağlı olan kapasitör vasıtasıyla akı değerine ulaşılmış olup, bu sayede bir adet harici direnç kullanımına ihtiyaç olmadan tasarım gerçekleştirilmiştir. Ayrıca tasarım içerisinde bulunan geçiş iletkenliği (g_m) kuvvetlendiricisi de bir başka direnç ihtiyacını ortadan kaldırmıştır ve direnç kullanılmadan bir tasarım gerçekleştirilmiştir. Bahsi geçen açılardan oldukça başarılı olan bu çalışma yalnızca topraklanmış yapıda oluşu ve uygulama örneği bulunmayışından dolayı geliştirilmeye müsait durumdadır. Yeşil ve arkadaşları [59], bir başka topraklanmış, dirençsiz ve akı-kontrollü

özelliklerine sahip, bir Gerilim Değişen Geçişiletkenliği Kuvvetlendiricisi (Voltage Difference Transconductance Amplifier-VDTA) ve bir MOSFET kapasitör kullanarak tasarlanmış memristör emülatör devresi sunmuşlardır. Bu çalışmada da benzer memdüktans denklemi elde edilmiştir. Babacan ve arkadaşları [60] bir başka topraklanmış akı-bağımlı memristör tasarımı gerçekleştirmiştir. Yalnızca dört adet MOSFET kullanılarak gerçekleştirilen bu çalışmanın bir uygulama öğreneğine yer verilmemiştir. İlerleyen yıllar içerisinde çeşitli akı-bağımlı memristör tasarımı gerçekleştirilmiştir [61]-[67]. Bu tasarımların birbirlerine olan üstünlükleri, topraklanmış-yüzen yapıda oluşu, içerdiği aktif ve pasif eleman sayısı, çalışma frekansı, uygulamalarda kullanılabilirliği, karmaşıklık düzeyi gibi parametrelere bakılarak değerlendirilir.

Bir akı-bağımlı memristör emülatör devresinin memristans değeri, incelenen bu çalışmaların çoğu göz önüne alınarak:

$$W(\varphi) = \frac{I_M}{V_M} = a + b\varphi(t) \tag{2.25}$$

şeklinde basite indirgenerek gösterilebilir. Burada I_M ve V_M ifadeleri sırasıyla memristör üzerinden akan akımı ve memristör üzerindeki gerilimi göstermektedir. Denklemde verilen a ve b katsayıları ise tasarlanan devreden elde edilen ve kullanılan transistörlerin iç yapısından gelen parametreleri içeren ifadelerdir. Bu değerler tasarlanan her memristör devresi için değişiklik gösterebilmektedir.

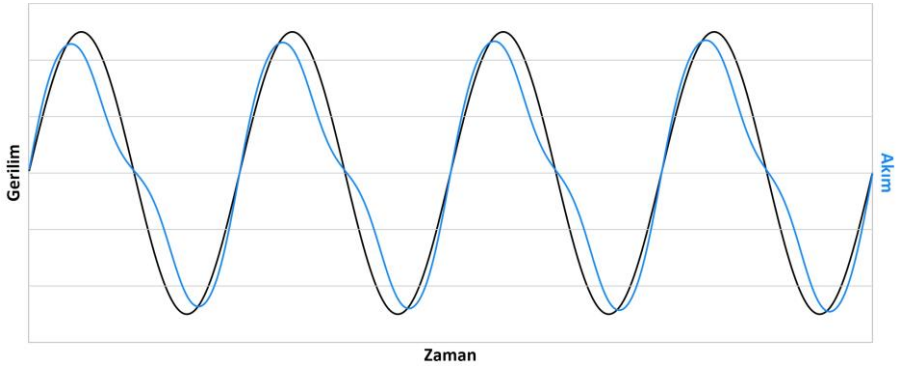
2.3.3. Memristör Emülatörü Analizi

Bir memristör emülatörü tasarımı gerçekleştirilip teorik analizleri yapıldıktan sonra, bu tasarımın çalıştığını göstermek amacıyla bazı analizlerinin gerçekleştirilmesi gerekmektedir. Bu analizlerin ilki memristör üzerinden akan akım ile memristör üzerindeki gerilimin zamanla değişiminin incelenmesi olacaktır. Şekil 2.5'te bir memristörün akım-zaman ve gerilim-zaman grafikleri bir arada verilmiştir. Daha sonra bu memristörün histerisiz grafiğine bakılmalıdır.

Örnek bir histerisiz eğrisi Şekil 2.6'da verilen grafikteki gibi olmalıdır. Burada x ve y eksenleri gerilim ve akımları ifade etmektedir.

Tasarlanan memristör emülatöründe kullanılan devre elemanlarının değerlerinin değişimi, sıcaklık analizleri, frekans analizleri gibi farklı durumları kapsayan koşullarda histerisiz

eğrisindeki değişimler de tasarımın farklı koşullardaki performansını gözlemlemek amacıyla incelenmelidir.



Şekil 2.5: Bir memristörün akım-zaman ve gerilim-zaman grafikleri örneği.

Memristör emülatörünün incelenmesi gereken bir diğer özelliği de artan veya azalan yapıda olup olmadığıyla ilgilidir. Bu konuya ilerleyen bölümlerde tekrar değinilecektir.



Şekil 2.6: Memristör histerisiz eğrisi örneği.

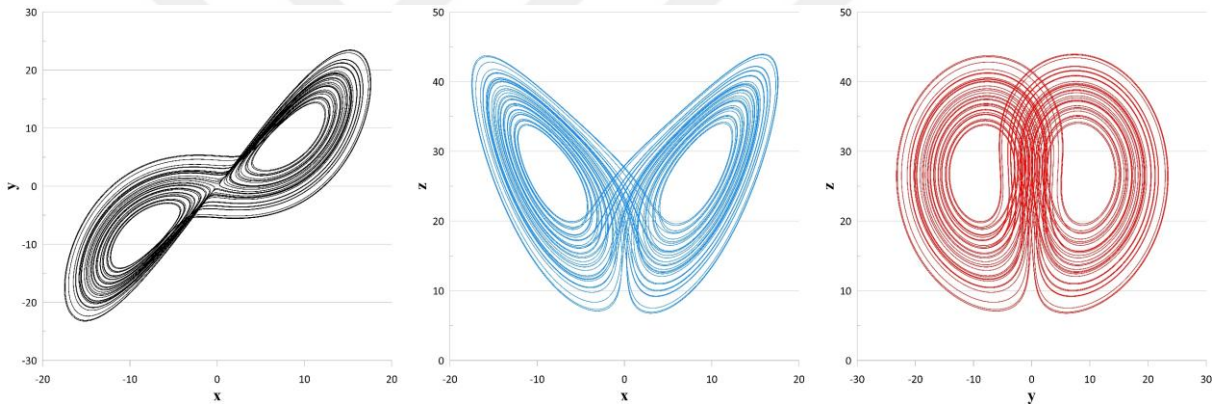
2.4. KAOTİK SİSTEMLER VE DEVRELER

Kaos, görünüş olarak rastgelelik, düzensizlik, karmaşıklık, tahmin edilemezlik gibi tanımlansa da davranış olarak deterministik özellik gösterebilen bir olgudur [68]. Kaos teorisine göre, kaotik bir sistemin girdilerinde meydana gelen en ufak bir değişim dahi çıktıların davranışını büyük ölçüde değiştirmektedir. Bu teorinin matematiksel izahatı ilk olarak Edward Norton

Lorenz isimli bilim insanı tarafından 1963 yılında gerçekleştirilmiştir [69]. Lorenz yaptığı bu çalışmada doğrusal olmayan özellik gösteren kaotik Lorenz denklemlerini literatüre sunmuştur. Bu denklem:

$$\begin{aligned} \dot{x} &= -\sigma x + \sigma y \\ \dot{y} &= -xz + rx - y \\ \dot{z} &= xy - bz \end{aligned} \tag{2.26}$$

şeklindedir. Denklem 2.26'dan elde edilen faz portrelerinin Matlab programı kullanılarak elde edildiği grafikler Şekil 2.7'de verilmiştir. Bu çalışmanın ardından matematik ve mühendislik alanlarından kaos teorisiyle ilgili birçok çalışma gerçekleştirilmiştir.



Şekil 2.7: Lorenz çekicisi faz portreleri.

Literatüre Lorenz denklemlerini elde edecek birçok elektronik devre tasarımı sunulmuştur. Örnek olarak, Gao ve arkadaşları [70], 9 adet OP-AMP, 3 adet analog çarpıcı, 23 adet direnç ve 4 adet kapasitör içeren bir kaotik devre tasarlamışlardır. Bu tasarım ile Lorenz denklemleri elde edilmiş olup Şekil 2.7'de verilen faz portrelerine benzer grafikler elde edilebilmiştir. Han ve arkadaşları [71] bir başka Lorenz sistemi benzeri kaotik devre tasarımı gerçekleştirmişlerdir. Bu tasarım yine OP-AMP kullanılarak gerçekleştirilmiştir. Toplamda 9 adet OP-AMP, 2 adet analog çarpıcı, 20 adet direnç ve 3 adet kapasitör kullanılmıştır. Son olarak Aydın ve arkadaşları [72] OTA tabanlı bir Lorenz sistemi benzeri kaotik devre tasarımı gerçekleştirmişlerdir. Bu tasarım direnç içermemekte olup, 7 adet OTA, 2 adet analog çarpıcı ve 3 adet kapasitör ihtiva etmektedir. Literatürde kaotik davranışa ulaşmak üzere tasarlanmış başka birçok Lorenz sistemi benzeri devre tasarımlarına ulaşmak mümkündür.

1976 yılında Otto E. Rössler bir başka kaotik denklem ortaya atmıştır [73]. Rössler denklemi olarak isimlendirilmekte olan bu kaotik denklem aşağıdaki gibidir:

$$\begin{aligned} \dot{x} &= -y - z \\ \dot{y} &= x + ay \\ \dot{z} &= b + (x - c)z \end{aligned} \tag{2.27}$$

Literatürde az da olsa Rössler sisteminin kaotik yapısından faydalanılarak tasarlanmış elektronik devreler bulunmaktadır [74], [75].

2.4.1. Kaotik Devreler

Lorenz sistemi veya Rössler sistemi gibi önerilmiş kaotik denklemleri elde etmek üzere elektronik devre tasarımının gerçekleştirilmesi oldukça zor ve karmaşıktır. Bu karmaşıklığın önüne geçmek adına, hafızalı devre elemanları kullanılarak kaotik devre tasarımı gerçekleştirilebilmektedir.

2.4.1.1. Chua Devresi

Daha önce de bahsi geçen Chua devresi, kaotik devre tasarımı konusunda en çok başvurulan yapılardandır. Itoh ve Chua, "Memristor oscillators" [34] isimli yayınlarında Chua devresiyle elde edilebilecek kaotik devreleri, Chua diyotu [76] yerine memristör kullanarak göstermişlerdir. Daha sonraki yıllarda bu yapılar geliştirilerek, diğer hafızalı elemanlar da (memkapasitör ve memindüktör) kullanılarak kaotik devre tasarımları gerçekleştirilmiştir [77-79].

Kaotik Chua devrelerinden bir tanesi Muthuswamy ve Chua [80] tarafından önerilen, Şekil 2.8'de verilen devredir. Kullanılan memristörün denklemleri:

$$v_M(z) = \beta(z^2 - 1)i_M \tag{2.28}$$

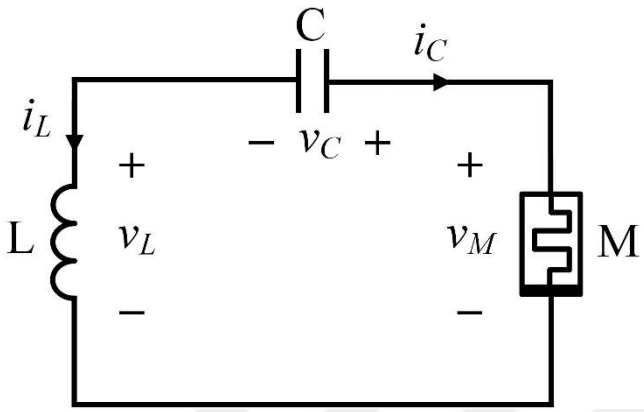
$$z = i_M - \alpha z - i_M z \tag{2.29}$$

olarak verilmiştir [80]. Bu denklemler kullanılarak Chua devresinde Kirchoff yasaları uygulanırsa sonuç olarak [81]:

$$\begin{aligned} \dot{v_C} &= \frac{i_L}{C} \\ \dot{i_L} &= \frac{1}{L}(v_C + \beta(z^2 - 1)i_L) \end{aligned} \tag{2.30}$$

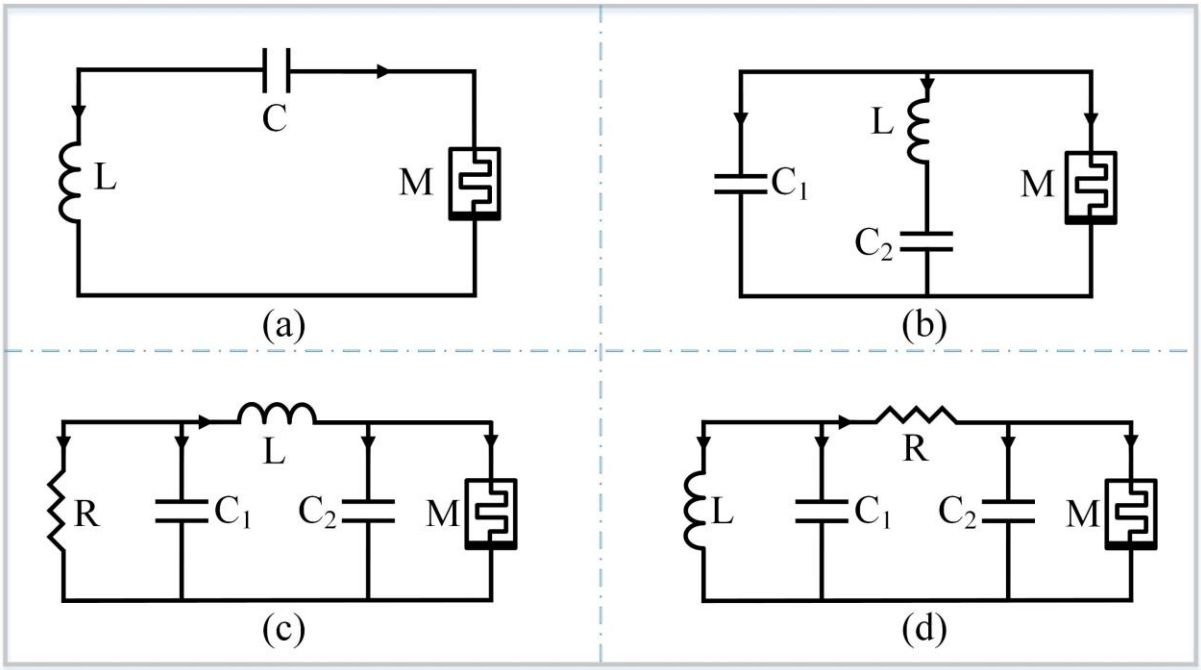
$$\dot{z} = i_L - \alpha z - i_L z$$

elde edilmektedir.



Şekil 2.8: Üç elemanlı Chua devresi.

Memristör kullanılarak gerçekleştirilebilecek bazı Chua devresi şematikleri Şekil 2.9'da özetlenmiştir [82].



Şekil 2.9: Farklı kaotik Chua devreleri.

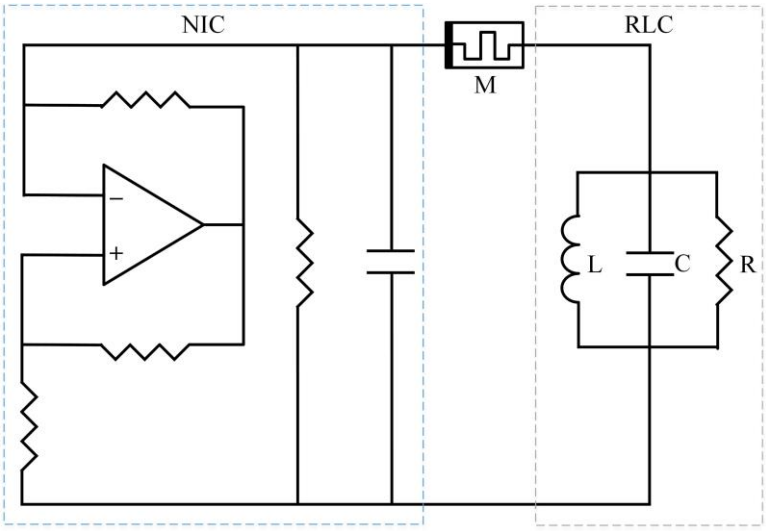
Literatürdeki birçok memristör emülatörü kullanılarak Şekil 2.9'da verilen Chua devreleriyle iyi seviyede kaotik sonuçlar alınamamaktadır. Bunun için emülatör devresinin kaotik olmaya uygun akım ve gerilim denklemlerine sahip olunmalıdır. Bu sebeple kaotik Chua devrelerinde diğer hafızalı devre elemanları tercih edilebilmektedir. Bunun dışında kaotik davranışa uygun olmayan memristör devreleri kübik forma getirilerek Chua devrelerinde kullanılmasıyla kaotiklik artırılabilmektedir [83]. Kübik formdaki bir memristörün memdüktans değeri [34]:

$$W(\varphi) = \frac{dq}{d\varphi} = a + 3b\varphi^2(t) \tag{2.31}$$

olarak hesaplanır. Buradaki a ve b değerleri tasarlanan devreden elde edilen parametrelerdir.

2.4.1.2. Shinriki Devresi

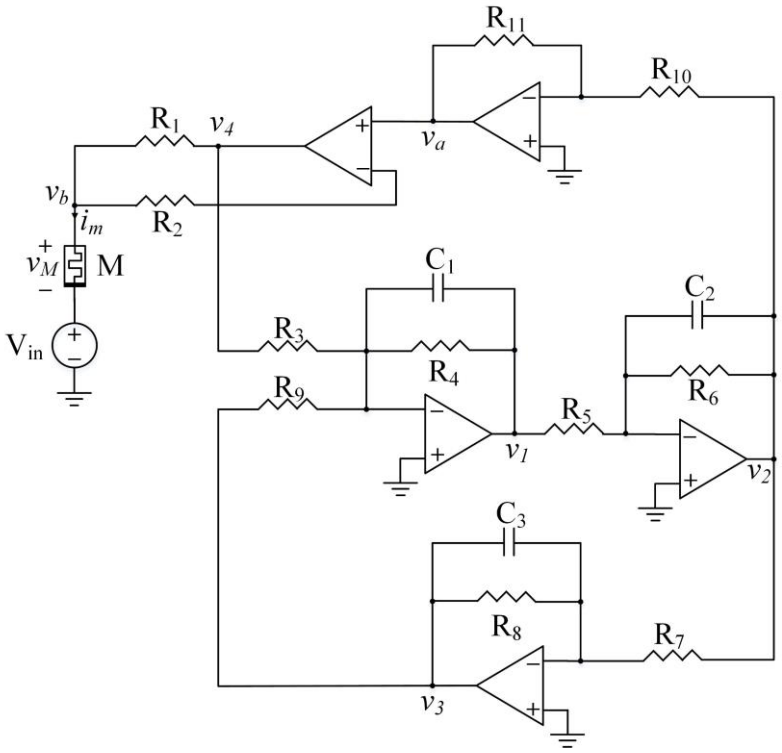
Shinriki devresi, bir negatif empedans çevirici (Negative Impedance Converter, NIC), bir RLC devresi ve bir doğrusal olmayan yapıdan meydana gelmektedir [36]. Devrenin basit yapısı Şekil 2.10'da verilmiştir. Devrenin gücü NIC yapısından sağlanmaktadır. Devredeki doğrusal olmayan parça için bazı çalışmalarda köprü diyot ve dirençten oluşan yapılar kullanılmıştır [30], [84]. 2020 yılında Volos ve arkadaşları tarafından yapılan bir çalışmada [85], KNOWM isimli firma tarafından üretilen fiziksel memristör elemanı kullanılarak Shinriki devresiyle kaotik osilatör tasarımı gerçekleştirilmiştir. Literatürde diyot köprüsü kullanımıyla elde edilen doğrusal olmayan devre elemanı dışında, memristör emülatörü kullanılarak da Shinriki devresi tasarımı mevcuttur [86]. Ancak literatürde Shinriki devresi, Chua devresi kadar yer bulamamıştır.



Şekil 2.10: Shinriki devresi.

2.4.1.3. Jerk Devresi

Kaotik devre tasarlamanın bir alternatif yolu Jerk fonksiyonlarından [37] faydalanarak gerçekleştirilmektedir. Bu fonksiyonlar temel olarak $\dddot{x} = J(\ddot{x}, \dot{x}, x)$ olarak ifade edilen üçüncü dereceden diferansiyel denklemler olarak tanımlanır. Sprott [31], çok sayıda OP-AMP ve direnç içeren, bir adet de diyot içeren bir kaotik Jerk devresi sunmuştur. Bu tasarımda $\dddot{x} + \ddot{x} + x + f(\dot{x}) = 0$ denkleminin elektronik olarak gerçekleştirilmesine, memristif bir bileşen dahil edilmeden ulaşılmıştır. Minati ve arkadaşları [33] yaptıkları çalışmada literatürdeki diğer Jerk devrelerine benzer bir devre kullanarak, gerçek bir memristör ile kaotik Jerk devresi tasarımı gerçekleştirmişlerdir. Kaos üretimi gibi analog uygulamalarda memristörün etkili kullanımı tipik olarak onun doğrusal olmayan özelliklerinden tüm çalışma aralığı boyunca yararlanılarak elde edilir. Referans [33]'te verilen kaotik devre, memristörün doğrusal olmamasının tek sebep olduğu bir yapıya sahiptir. Sunulan devrede, biri doğrusal, diğeri memristörden kaynaklanan ve doğrusal olmamaya neden olan iki geri besleme döngüsü bulunmaktadır. Bu geri besleme döngüleri toplam üç integral alıcı yapı içermektedir. Bu özellikleri ve gerçekçi yapısı nedeniyle, bu çalışmada önerilen memristör emülatörünün bir uygulama örneği olarak referans [33]'te sunulan kaotik devre yapısı tercih edilmiştir. Bahsi geçen Jerk devresi Şekil 2.11'de verilmiştir.



Şekil 2.11: Jerk devresi [33].

2.4.2. Kaotik Sistemlerin Analiz Yöntemleri

Bir sistemin kaotik davranışa sahip olup olmadığını tespit etmek için bu sistemin matematiksel verilerinden faydalanılarak gerçekleştirilen çeşitli yöntemler bulunmaktadır. Bu yöntemlerden en temel 4 tanesi, Rössler denklemlerinden faydalanılarak, bu bölümde incelenecektir.

2.4.2.1. Kararlılık Analizi

Bir sistemin kaotik olabilmesi için öncelikle kararsız olması gerekmektedir. Kararlı bir sistem kaotik davranış gösterememektedir. Bu sebeple kaos analizi yapılırken öncelikle sistemin kararlı olup olmadığına bakılmalıdır. Bunun için sistemin denge noktalarındaki özdeğerleri incelenmelidir. Elde edilen özdeğerlerden en az birinin reel kısmının pozitif oluşu sistemin kararsız olduğuna işaret etmektedir. Örnek olarak (2.27)'de verilen Rössler denklemini ele alalım. Denge noktalarının bulunması için öncelikle bu denklem,

$$\begin{aligned} 0 &= -y - z \\ 0 &= x + ay \\ 0 &= b + (x - c)z \end{aligned} \tag{2.32}$$

haline getirilir. Elde edilen denklem çözülürse denge noktalarını bulmak için,

$$y = -z \tag{2.33}$$

$$x = -ay = az \tag{2.34}$$

$$xz - zc - b = az^2 - zc - b = 0 \tag{2.35}$$

$$z = \frac{-c \mp \sqrt{c^2 - 4ab}}{2a} \tag{2.36}$$

eşitliklerinden faydalanılır. Burada elde edilen *z* değerinden hareketle *x* ve *y* ifadelerinin denge noktalarındaki değerleri elde edilebilmektedir. Daha sonra aşağıdaki denklemlerle gösterildiği gibi Jacobian matrisi elde edilir.

$$J = \begin{bmatrix} \frac{\partial \dot{x}}{\partial x} & \frac{\partial \dot{x}}{\partial y} & \frac{\partial \dot{x}}{\partial z} \\ \frac{\partial \dot{y}}{\partial x} & \frac{\partial \dot{y}}{\partial y} & \frac{\partial \dot{y}}{\partial z} \\ \frac{\partial \dot{z}}{\partial x} & \frac{\partial \dot{z}}{\partial y} & \frac{\partial \dot{z}}{\partial z} \end{bmatrix} \tag{2.37}$$

$$J = \begin{bmatrix} 0 & -1 & -1 \\ 1 & a & 0 \\ z & 0 & x - c \end{bmatrix} \tag{2.38}$$

Elde edilen Jacobian matrisinden hareketle özdeğerleri bulmak adına $|J-\lambda I| = 0$ eşitliğinden yararlanılır. Burada λ özdeğerleri, I birim matrisini ifade etmektedir. Buradan hareketle,

$$|J - \lambda I| = \begin{vmatrix} -\lambda & 1 & -1 \\ 1 & a-\lambda & 0 \\ z & 0 & x-c-\lambda \end{vmatrix} = 0 \tag{2.39}$$

$$|J - \lambda I| = \lambda^3 + (a + x - c)\lambda^2 + (ac - ax - z - 1)\lambda + x - c + az = 0 \tag{2.40}$$

eşitlikleri elde edilir. Sistemin kaotik olması için burada verilen özdeğerlerin köklerinden ne az bir tanesinin reel kısmının pozitif değerde olması gerekmektedir. (2.40) denkleminde $a = 0.2$, $b = 0.2$, $c = 5.7$ olarak alınarak (x, y, z) denge noktaları sayısal bir değere sahip olarak bulunabilmektedir. Denge noktaları ve değişkenler (2.40) ile gösterilen denklemde yerine yazılırsa bu denklemin kökleri yaklaşık olarak, $\lambda_1 = 0.097 + j0.996$, $\lambda_2 = 0.097 - j0.996$, $\lambda_3 = 5.678$ elde edilmektedir. Buradan da görüldüğü gibi, elde edilen özdeğerlerden en az bir tanesinin reel kısmı pozitif olarak bulunmuştur. Bu sonuç sistemin karasız olduğunu ve kaotik davranış gösterebileceğine işaret etmektedir.

2.4.2.2. Lyapunov Üstelleri

Kaotik bir sistemin en belirgin özelliklerinden biri başlangıç koşullarına bağlı olmasıdır. Başlangıç koşullarına bağlılığın miktarını analiz etmek amacıyla Lyapunov üstelleri incelenmektedir. Lyapunov üsleri, faz uzayındaki yakındaki yörüngelerin ortalama üstel ıraksama veya yakınsama oranlarını ölçerek, kaotik sistemlerin dinamik davranışı hakkında fikir vermektedir [87]. Dolayısıyla bir sistemden elde edilen Lyapunov üstellerine bakılarak sistemin kaotik davranış sergileyip sergilemediği ispatlanabilmektedir. Bir sistemin Lyapunov üstellerinden en az bir tanesinin pozitif olması sistemin kaotik olduğuna işaret etmedir. Ancak bu koşul kaotik davranışı ispatlamak için tek başına yeterli olmamaktadır. Aynı zamanda sistemden elde edilen tüm Lyapunov üstellerinin toplamlarının negatif değerde olması gerekmektedir [86].

d boyutuna sahip $\boldsymbol{x}$ vektörü ile tanımlanan sürekli zamanlı bir sistemde, sistemin parametrelerinin vektörü $\boldsymbol{q}$ ile tanımlanmış olsun. Bu sistem (2.41) eşitliğiyle tanımlansın. Bu sistemdeki başlangıç $\boldsymbol{x}$ değeri x_0 ile verilirse ve bu vektörde meydana gelecek ufak sapmalar e ile gösterilecek olursa, $e(0) = e_0$ olmak üzere elde edilen sapmaya uğramış yeni vektör $\boldsymbol{y(t) = x_0 + e(t)}$ olacaktır. Yörüngede meydana gelen e sapma sapması (2.42) ile tanımlanmaktadır.

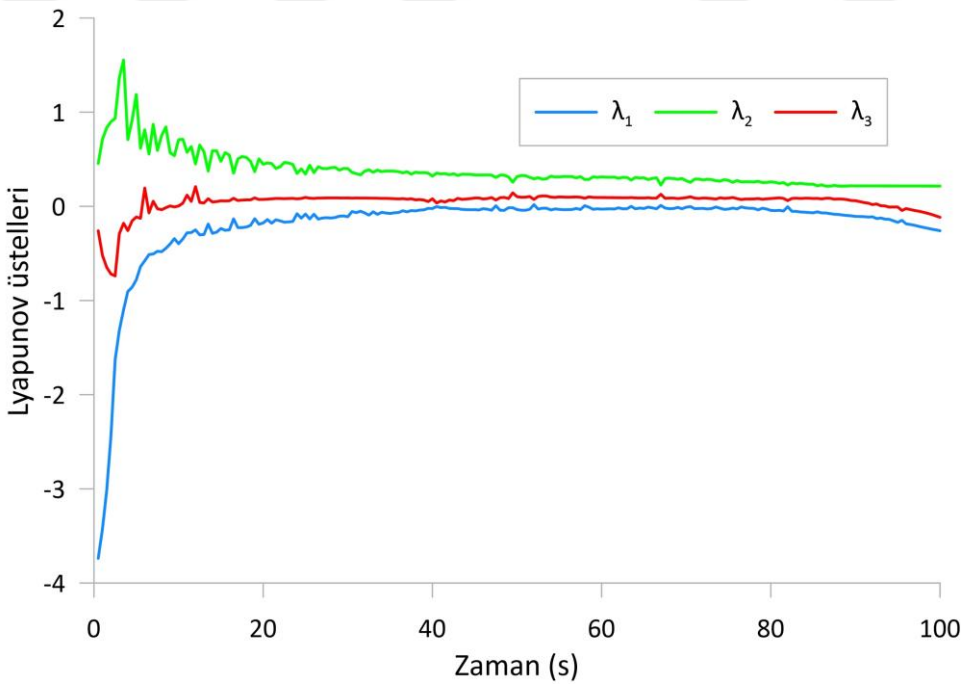
$$\frac{d}{dt}\boldsymbol{x}(\boldsymbol{t}) = \boldsymbol{F}(\boldsymbol{x};\boldsymbol{q}) \tag{2.41}$$

$$\frac{d}{dt}e(t) = Je(0) \tag{2.42}$$

Burada verilen *J* değeri Jacobian matrisini ifade etmektedir. Daha önce de belirtildiği gibi başlangıç sapması $e(0) = e_0$ olarak alınırsa, sapma, $e(t) = q(t)e(0)$ ifadesiyle elde edilebilir. Daha sonra Lyapunov üstelleri (2.43) ile tanımlanmaktadır ($i = 1, 2, 3, \ldots, d$) [88]. Bu eşitlikten de Lyapunov üstellerinin başlangıç sapmalarından etkilendiği görülmektedir.

$$\lambda_i = \lim_{t\to\infty} \frac{1}{t} \ln\left(\frac{\|e(t)\|}{\|e_0\|}\right) \tag{2.43}$$

Bu bölümde de örnek olması açısından Rössler denklemlerinden elde edilen Lyapunov üstelleri ele alınacak olursa Şekil 2.12'de verilen grafik elde edilmektedir. Grafikten de görüldüğü gibi elde edilen Lyapunov üstellerinden bir tanesi pozitif değerli ve Lyapunov üstellerinin toplam değerleri negatif sonuç vermektedir. Bu durum sistemin kaotik davranış gösterdiğini ispatlamaktadır. Bu grafik oluşturulurken, $a = 0.2$, $b = 0.2$ ve $c = 5.7$ alınmıştır. Verilen grafik, MATLAB programı aracılığıyla elde edilmiştir.



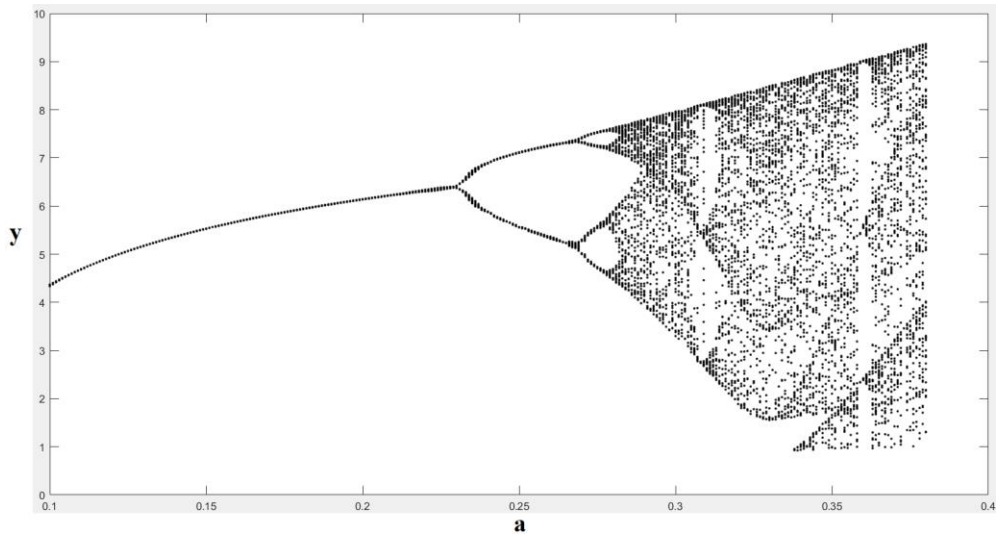
Şekil 2.12: Rossler sistemi Lyapunov üstelleri.

2.4.2.3. Çatallanma Diyagramı

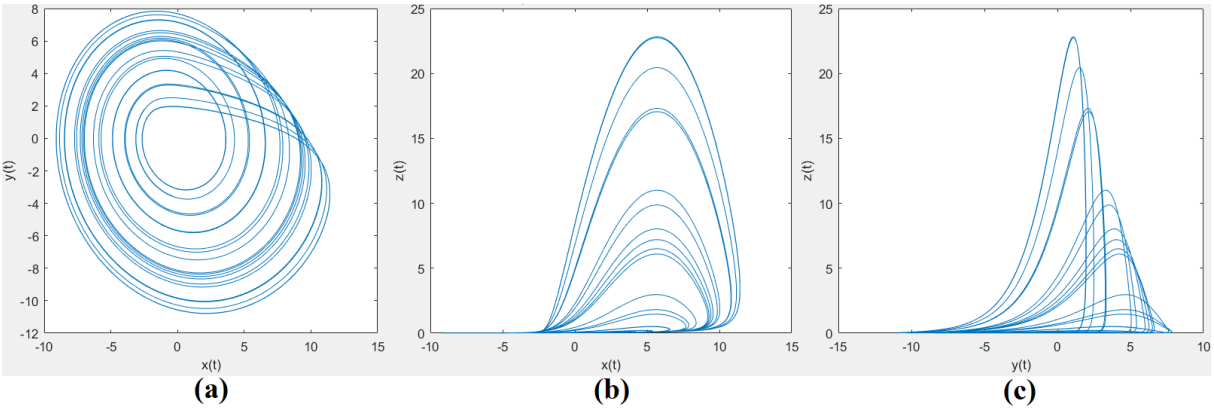
Bir sistemin kaotik davranış gösterip göstermediğini gözlemlemek adına incelenen bir diğer özellik ise o sistemin çatallanma diyagramıdır. Çatallanma diyagramı, sistemdeki bir değişkenin, sistemi tanımlayan vektörü oluşturan fonksiyonlardan birine göre değişmesiyle oluşan çatallaşmaları göstermektedir. Gözlemlenen bu çatallaşmalar sistemin kaotik olduğuna işaret etmektedir [89]. Kaotik bir sistemin çatallanma diyagramına örnek olarak Rössler sistemi ele alınacak olursa, Şekil 2.13'te verilen grafik elde edilebilmektedir. Verilen bu grafik *a* parametresinin değişimi ile *y* fonksiyonunda meydana gelen değişikliği göstermektedir. Bu grafik elde edilirken, *a* parametresi 0.1 ile 0.38 aralığında değiştirilmiştir. Sistemin diğer parametreleri $b = 2$ ve $c = 5.7$ olarak alınmıştır. Verilen grafik, MATLAB programı aracılığıyla elde edilmiştir.

2.4.2.4. Faz Portreleri

Kaotik bir sistemin analiz yöntemlerinden bir diğeri ise faz portreleri analizidir. Faz portreleri ile bir sistemin durum denklemlerinin, belirli parametreler ile, birbirlerine göre değişiminin gözlemlenmesi mümkün olmaktadır. Bu kısımda da kaotik davranış gösteren Rössler sisteminin faz portreleri Şekil 2.14'te örnek olarak verilmiştir. Bu grafiklerin elde edilmesinde parametre değerleri, $a = 0.2$, $b = 0.2$ ve $c = 5.7$ olarak seçilmiştir. Grafikler MATLAB programı aracılığıyla elde edilmiştir.



Şekil 2.13: Rössler sisteminin çatallanma diyagramı; a parametresine göre y fonksiyonun değişimi.



Şekil 2.14: Rössler sisteminden elde edilen faz portreleri, (a): x-y grafiği, (b): x-z grafiği, (c): y-z grafiği.

2.5. KAOTİK HABERLEŞME

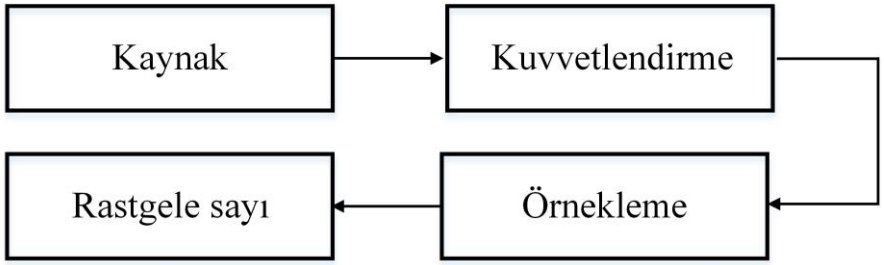
Çalışma kapsamında sunulacak haberleşme sisteminin bilgi sinyali olarak görüntü kullanılacaktır. Bu bölümde, görüntü bilgisinin şifrelenmesi ve modülasyon işleminin gerçekleştirilmesi için gerekli olan ön bilgilere yer verilecektir.

2.5.1. Rastgele Sayı Üreteçleri (RSÜ)

Bazı sistemler için birbirinden bağımsız, tamamen gelişigüzel sayıların üretilmesine ihtiyaç duyulmaktadır. Bu sistemler için kullanılan yapılar rastgele sayı üreteçleri olarak isimlendirilmektedir. Bu sayılar, hassas bilgileri koruyabilen güvenli anahtarlar ve kodlar oluşturmak için rastgele sayılara dayanan birçok modern uygulama için gereklidir [96, 97]. Rastgele sayı üreteçleri, bilgisayar uygulamaları, istatistik, ekonomi, haberleşme ve daha birçok alanda ihtiyaç duyulmaktadır. Bir haberleşme sisteminin güvenilirliğinin yüksek olması için iyi şifrelenmiş olması gerekmektedir. İyi bir şifreleme iyi sayılarla sağlanabilmektedir [98]. Bu sebeple iyi bir haberleşme sisteminin şifrelenmesi için iyi bir rastgele sayı üretecine ihtiyaç duyulmaktadır. Rastgele sayı üreteci tasarımında kaostan faydalanmak tasarımın kalitesinde olumlu bir etkiye sebep olacaktır. Daha önce birçok çalışmada kaotik rastgele sayı üreteci tasarımı gerçekleştirilmiştir [99-102]. Rastgele sayı üretecinin temel aşamaları Şekil 2.15'te verilmiştir [103]. Burada verilen kaynak kısmını oluşturacak yapı gelişigüzel veriler üretmelidir. Daha sonra bu verilen kuvvetlendirilip sayısal yapıya çevrilmektedir. Son olarak sistemin rastgelelilik seviyesi artırılarak işlem tamamlanmış olur.

Bu çalışma kapsamında önerilecek olan hafızalı devre elemanları kullanılarak üretilen kaotik sistem kullanılarak rastgele sayı üreteci tasarımı gerçekleştirilecektir. Daha sonra tasarlanan

RSÜ, görüntü şifreleme ve şifre çözme işlemlerine katkıda bulunmak üzere haberleşme sisteminde kullanılacaktır.



Şekil 2.15: Rastgele sayı üreteci blok diyagramı [104].

2.5.2. Kaotik Şifreleme ve Şifre Çözme

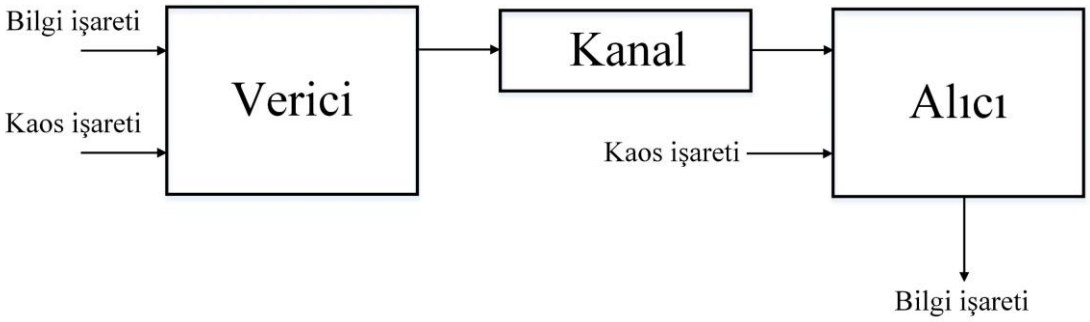
Haberleşme sisteminin en önemli aşamalarından biri olan şifreleme basamağında, rastgele sayı üreteçleri kullanılarak bilginin şifrelenmesi işlemi gerçekleştirilebilmektedir. Bilgi işaretinin şifrelenmesinde kaotik sistemlerin kullanılmasıyla, kırılması oldukça zor olan şifreleme işlemi gerçekleştirilmiş olmaktadır. Bilgi şifrelenirken RSÜ kullanılarak anahtar oluşturulmaktadır. Şifreleme için oluşturulan bu anahtarın şifre çözme işlemi için de kullanıldığı yapılara simetrik şifreleme olarak isimlendirilmektedir. Asimetrik şifrelemede şifre çözmek için yeni anahtar üretilmektedir.

1989 yılında Matthews [108], rastgele sayı üretiminde kaostan faydalanılarak bir şifreleme algoritması gerçekleştirmiştir. Bu çalışmayla beraber kaotik bilgi şifreleme ve şifre çözme işlemlerinin uygulanabilirliği görülmüş olup, bu yönde gerçekleştirilen çalışmalar hız kazanmıştır. Literatürde kaotik sistemlerle özellikle görüntü şifreleme uygulamalarına sıkça rastlanmaktadır. Örnek olarak, [109-112] referanslarıyla belirtilen, görüntü şifreleme için kaotik algoritmalar üretip bu algoritmaların uygulanmasını izah eden bu çalışmalarda, kaotik işaretin kaynağı olarak literatürde daha önce matematiksel açıklamaları elde edilmiş olan, Chen'nin sistemi [113] ve Arnold kedisi [114] gibi bazı sistemler kullanılmıştır. Bu çalışmalar haricinde elektronik devre temelli kaotik sistemden faydalanılan birçok görüntü şifreleme çalışması da gerçekleştirilmiştir [115-119]. Bu çalışmaların çoğunluğunda kaotik sistem oluşturulurken Chua devresinden faydalanılmıştır.

Bu tez çalışmasında, ilerleyen bölümlerde detayları açıklanacak olan, kaos tabanlı simetrik anahtarlar kullanılarak görüntü şifreleme ve şifre çözme işlemleri gerçekleştirilecektir.

2.5.3. Kaotik Maskeleme

Bilgi işaretinin haberleşmede kullanılmak üzere modüle ve demodüle edilmesinde kullanılan yöntemlerden bir tanesi kaotik maskeleme yöntemidir. Kaos kullanılarak elde edilmeye çalışılan güvenli haberleşme sisteminin gerçekleştirilmesinde verici kısımda ve alıcı kısımda kaotik sistemin bulunması gerekmektedir [105]. Bu sistemin en temel hali Şekil 2.16'da verilmiştir.



Şekil 2.16: Kaotik maskeleme blok diyagramı [106].

3. YÖNTEM

Bu bölümde çalışma kapsamında kullanılan tümleşik yapılar açıklanacaktır. Daha sonra tasarlanan memristör emülatörü devresinin detayları incelenip, sunulan bu yapının kaotik uygulamalarda kullanımı gösterilecektir. Son olarak kaotik uygulamalarda kullanımı gösterilen memristör emülatörünün haberleşme sistemine uyarlanması açıklanmıştır.

3.1. AKTİF ELEMANLAR

Memristör emülatörü tasarımında kullanılan MOSFET tabanlı aktif elemanlar bu bölümde açıklanacaktır.

3.1.1. Akım Kontrollü Akım Taşıyıcı Geçiş İletkenliği Kuvvetlendiricisi (CC-CCTA)

CC-CCTA, akım taşıyıcı geçiş iletkenliği kuvvetlendiricisi olan CCTA'nın girişine parazitik direnç etkisi oluşturacak bir akım kontrollü yapı eklenmesiyle elde edilmiştir [90]. CC-CCTA'nın tez kapsamında kullanılmak üzere modifiye edilmiş hali Şekil 3.1'de verilmiştir. Bu yapıya göre parazitik direnç olan R_X, 16 numaralı MOSFET üzerinden akan akım olan I_{D16}'ya bağlıdır. Şekilde *B* ile gösterilen uca uygulanan gerilim ile I_{D16} akımı kontrol edilebilmektedir. CC-CCTA'nın genel eşitlikleri:

$$V_Y = V_X - I_X R_X \tag{3.1}$$

$$I_X = I_{Z+} = -I_{Z-} \tag{3.2}$$

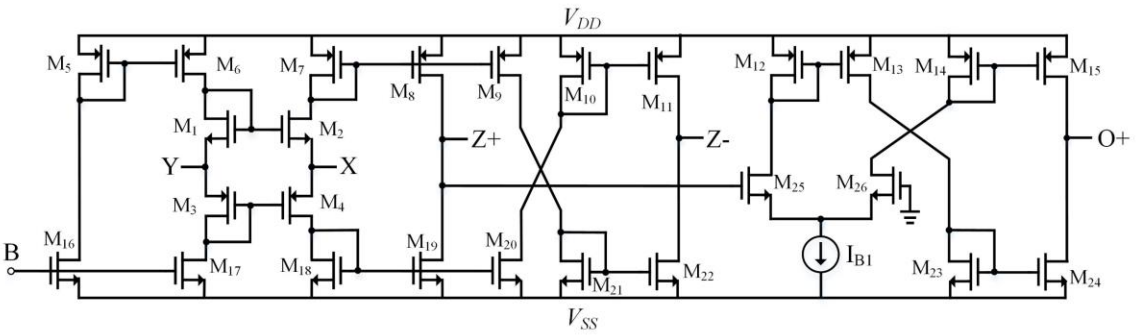
$$I_O = g_m V_{Z+} \tag{3.3}$$

$$R_X = \frac{1}{\sqrt{8\mu_n C_{ox}\left(\frac{W}{L}\right)_{1,2} I_{D16}}} \tag{3.4}$$

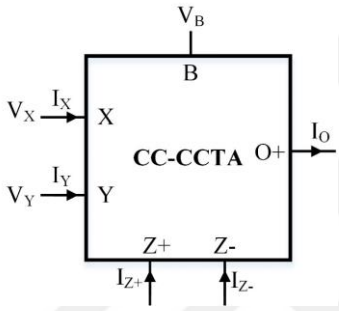
$$g_m = \sqrt{\mu_n C_{ox}\left(\frac{W}{L}\right)_{24,25} I_B} \tag{3.5}$$

şeklinde tanımlanmaktadır. Bu eşitliklerde g_m, μ_n, C_{OX}, and I_{D16} ifadeleri sırasıyla, geçiş iletkenliği, elektron hareketliliği, oksit kapasitans ve M_{16} transistöründen akan akım olarak verilmiştir. *W/L* oranı ise indiste numaraları belirtilen transistörlerin kanal genişliği ve kanal boyu oranlarını ifade etmektedir.

CC-CCTA'nın blok diyagramı Şekil 3.2'de gösterilmiştir.



Şekil 3.1: CC-CCTA iç yapısı.



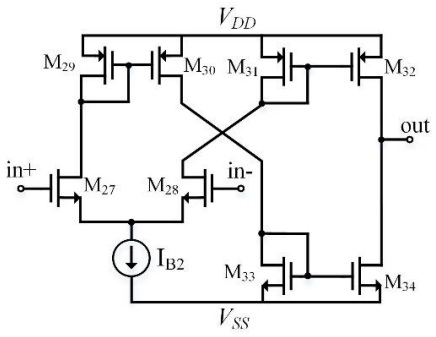
Şekil 3.2: CC-CCTA blok diyagramı.

3.1.2. İşlemsel Geçiş İletkenliği Kuvvetlendirici (OTA)

Bu çalışmada kullanılacak olan OTA aktif elemanının iç yapısı Şekil 3.3'te verilmiştir [91]. Verilen OTA'nın matematiksel denklemi (3.6) ile verilmiştir.

$$g_m(V_{in+} - V_{in-}) = I_{out} \tag{3.6}$$

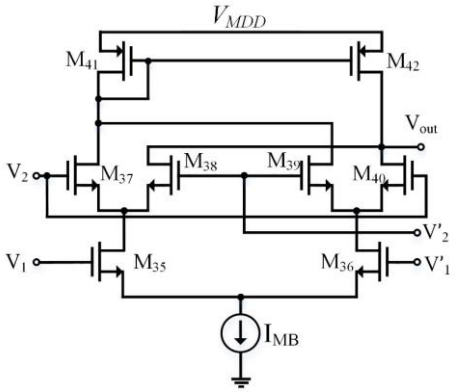
Bu tez kapsamında OTA, tasarlanan memristör emülatörü devresini yüzen forma getirmek amacıyla devreye dahil edilmiştir. Gerçekleştirilecek uygulamaya bağlı olarak OTA, sunulan memristör emülatöründe kullanılması zorunlu olmayan devre elemanındır.



Şekil 3.3: OTA'nın iç yapısı.

3.1.3. Analog Çarpıcı

Çalışma kapsamında kullanılan analog çarpıcı devresi Şekil 3.4'te verilmiştir [92]. Bu çalışmada, verilen çarpıcı devresinin kullanım amacı, tasarlanan memristörün Chua devresinde kullanılmak üzere kübik forma getirilmesini sağlamaktır. Bu işlemin nasıl gerçekleştirildiğine ilerleyen bölümlerde değinilecektir.



Şekil 3.4: Analog çarpıcı iç yapısı.

3.2. ÖNERİLEN MEMRİSTÖR EMÜLATÖRÜ

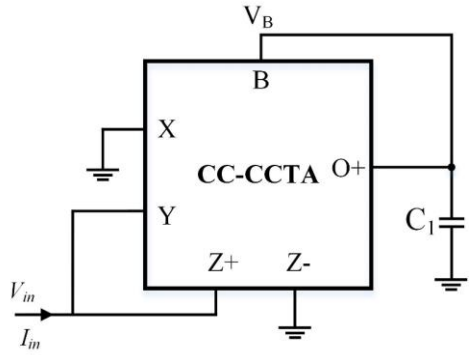
Bu tez kapsamında topraklanmış ve yüzen olmak üzere iki farklı tipte memristör emülatörü sunulmuştur.

3.2.1. Topraklanmış Memristör Emülatörü

Önerilen topraklanmış memristör emülatörü devresi Şekil 3.5'te verilmiştir. Bu devre şekilden de görüldüğü gibi bir adet CC-CCTA ve bir adet kapasitör içermektedir. Sunulan memristör emülatörü, devre tasarımının CC-CCTA ile gerçekleştirilmesinden dolayı, memristör tasarımı için gerekli olacak iki adet direnç kullanılmayarak elde edilmiştir. İhtiyaç duyulan dirençlerden bir tanesi aktif elemanın akım kontrollü oluşundan kaynaklı ortaya çıkan R_X direnciyle sağlanmıştır. CC-CCTA'nın geçişiletkenliği kuvvetlendiricisi katındaki g_m geçiş iletkenliğinden dolayı diğer bir direnç devreden eksiltilebilmiştir.

Önerilen memristör emülatörüne bakılacak olursa, aktif eleman olarak kullanılan CC-CCTA'nın *X* girişi topraklanmıştır. Yani burada görülen gerilim değeri sıfıra eşittir. Dolayısıyla daha önce CC-CCTA için verilen (3.1) denklemi tekrar değerlendirilecek olursa (3.7) eşitliğine ulaşılmaktadır.

$$V_Y = -I_X R_X \tag{3.7}$$



Şekil 3.5: Önerilen topraklanmış memristör emülatörü devresi.

Burada verilen V_y gerilimi, memristörün giriş gerilimi olan V_{in} değerine eşittir. Ayrıca devredeki X portundaki akımın Z portuna taşınmasından dolayı (3.8) denklemi ve buradan da hareketle (3.9) denklemi elde edilebilir.

$$V_{in} = -I_{in}R_X \tag{3.8}$$

$$\frac{I_{in}}{V_{in}} = -\frac{1}{R_X} \tag{3.9}$$

Diğer taraftan, aktif elemanın geçiş iletkenliği kuvvetlendiricisi kısmının çıkışından elde edilen akım değeri (3.10) denkleminde verilen şekilde hesaplanabilmektedir.

$$I_O = g_m V_{in} \tag{3.10}$$

Burada elde edilen I_O çıkış akımı vasıtasıyla C_1 kapasitörü üzerinde bir gerilim elde edilebilmektedir. Oluşan bu gerilim devre şemasında görüldüğü gibi B portunda da görülecektir. Dolayısıyla B portunda görülen bu gerilim:

$$V_B = \frac{g_m}{C_1}\int_{-\infty}^{t} V_{in}dt \tag{3.11}$$

şeklinde elde edilmektedir. Daha sonra akı (φ) ifadesi ve akıya bağlı V_B gerilimi devam eden eşitliklerde görüldüğü gibi elde edilmektedir.

$$\varphi_{in} = \int_{-\infty}^{t} V_{in}dt \tag{3.12}$$

$$V_B = \frac{g_m \varphi_{in}}{C_1} \tag{3.13}$$

Daha önceden (3.4) ile belirtilen denkleme bakılacak olursa, R_X direnci ile V_B gerilimi arasında I_{D16} akımına bağlı olarak gözlemlenen dolaylı bir ilişki olduğu görülmektedir. Bu ilişkiden kaynaklı olarak ilerleyen denklemler elde edilebilmektedir.

$$\frac{1}{R_X} = \sqrt{8k_1 I_{D16}} \tag{3.14}$$

$$I_{D16} = \frac{1}{2}k_2(V_B - V_{SS} - V_{tn})^2 \tag{3.15}$$

$$\frac{1}{R_X} = 2\sqrt{k_1 k_2 (V_B - V_{SS} - V_{tn})^2} \tag{3.16}$$

$$\frac{I_{in}}{V_{in}} = -2\sqrt{k_1 k_2}(V_B - V_{SS} - V_{tn}) \tag{3.17}$$

Bu denklemlerde verilen $-V_{SS}$ gerilimi negatif besleme gerilimini, V_{tn} gerilimi ise kullanılan NMOS'un eşik gerilimini ifade etmektedir. Ayrıca kullanılan diğer katsayılar ise $k_1 = \mu_n C_{ox}(W/L)_{1,2}$, and $k_2 = \mu_n C_{ox}(W/L)_{16}$ şeklindedir.

Elde edilen ifadeler toparlanacak olursa sonuç olarak (3.18) ile verilen memdüktans değeri elde edilmektedir.

$$\frac{I_{in}}{V_{in}} = W(\varphi) = -2\sqrt{k_1 k_2}\left(\frac{g_m \varphi_{in}}{C_1} - V_{SS} - V_{tn}\right) \tag{3.18}$$

Elde edilen memdüktans ifadesinden de anlaşılacağı gibi önerilen memristör devresi akı kontrollüdür.

3.2.2. Yüzen Memristör Emülatörü

Sunulan topraklanmış memristör emülatörünün yüzen hale getirilmesi için bir adet OTA'ya ihtiyaç duyulmaktadır. Bu şekilde elde edilen yüzen memristör emülatörü Şekil 3.6'da verildiği gibidir.

Bu tasarımda kullanılan OTA'nın geçiş iletkenliği (g_m) değeri, CC-CCTA'nın çıkışındaki geçiş iletkenliği kuvvetlendiricisinden elde edilen geçiş iletkenliği değerine eşit olmalıdır. Devre detaylıca incelenecek olursa ilk olarak:

$$V_{in} = V_X - V_Y \tag{3.19}$$

$$\frac{I_{in}}{V_{in}} = \frac{1}{R_X} \tag{3.20}$$

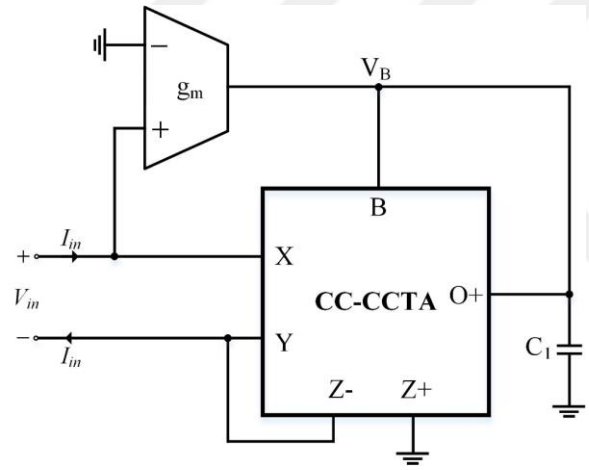
eşitlikleri elde edilmektedir. OTA'nın çıkışından elde edilen akım değeri I_O akımına eşit olduğundan *B* portundaki gerilim değeri aşağıdaki gibi elde edilir.

$$V_B = \frac{2g_m}{C_1}\int_{-\infty}^{t} V_{in}dt \tag{3.21}$$

$$V_B = \frac{2g_m\varphi_{in}}{C_1} \tag{3.22}$$

Elde edilen bu eşitliklerden faydalanılarak yüzen memristör emülatörünün memdüktans değeri (3.23) denklemi olarak elde edilmektedir.

$$\frac{I_{in}}{V_{in}} = W(\varphi) = 4\sqrt{k_1k_2}\left(\frac{g_m\varphi_{in}}{C_1} - V_{SS} - V_{tn}\right) \tag{3.23}$$



Şekil 3.6: Önerilen yüzen memristör emülatörü devresi.

3.2.3. Parazitik Analiz

CC-CCTA'nın parazitik elemanları Şekil 3.7'de gösterilmektedir [93]. Bu elemanların kullanıldığı, önerilen topraklanmış memristör emülatör devresi Şekil 3.8'de gösterilmektedir. Tasarımın sınırlayıcı özelliklerini belirlemek için, parazitik bileşenler içeren devre iki parça halinde incelenebilir. İlk olarak devrenin çıkışında bulunan C_O ve R_O parazitik elemanları ele alınacak olursa, (3.24) ile verilen eşitsizlik sağlanmalıdır.

$$\left|R_O \| \frac{1}{jwC_O}\right| \gg \left|\frac{1}{jwC_1}\right| \tag{3.24}$$

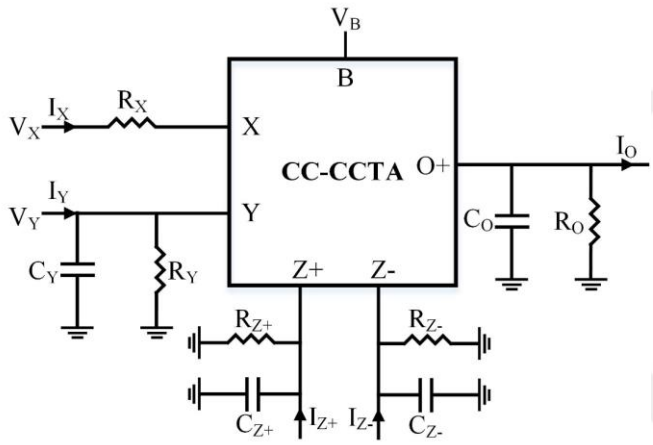
(3.24) eşitsizliğinin sol tarafının sağ taraftan en az 10 kat daha büyük olduğu durum kabul edilebilir. Bu varsayımla aşağıdaki eşitsizlikler elde edilebilir.

$$\left|R_O \| \frac{1}{jwC_O}\right| \geq \left|\frac{10}{jwC_1}\right| \tag{3.25}$$

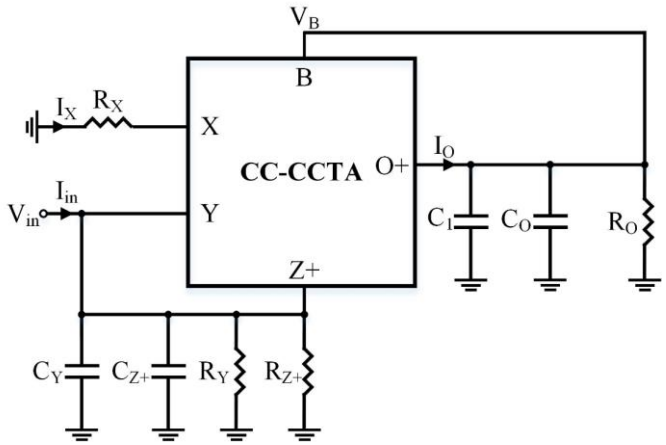
$$\frac{R_O}{\sqrt{1+w^2R_O^2C_O^2}} \geq \frac{10}{wC_1} \tag{3.26}$$

(3.26) denklemindeki R_O direnci ihmal edilirse, C_1 kapasitesi için aşağıdaki koşul sağlanmalıdır.

$$C_1 \geq 10C_O \tag{3.27}$$



Şekil 3.7: CC-CCTA'nın parazitik bileşenleri.



Şekil 3.8: Parazitik bileşenleri içeren memristör emülatörü.

(3.26) denkleminde C_O kapasitesinin ihmal edilmesiyle ise alt sınır frekansı ilerleyen denklemlerdeki gibi elde edilmektedir.

$$R_O \geq \frac{10}{wC_1} \tag{3.28}$$

$$w \geq \frac{10}{R_O C_1} = w_L \tag{3.29}$$

Önerilen memristör emülatörünün parazit bileşenli giriş aşaması dikkate alındığında aşağıdaki eşitsizliklere ulaşılabilir.

$$\left| R_Y || R_Z || \frac{1}{jw(C_Y + C_Z)} \right| \gg R_X \tag{3.30}$$

$$\left| R_Y || R_Z || \frac{1}{jw(C_Y + C_Z)} \right| \geq 10R_X \tag{3.31}$$

$$\frac{R_Y || R_Z}{\sqrt{1 + w^2 (R_Y || R_Z)^2 (C_Y + C_Z)^2}} \geq 10R_X \tag{3.32}$$

Bu noktada, önerilen memristör emülatörünün üst limit frekansını elde etmek için denklem (2.32)'de $R_Y||R_Z$ göz ardı edilirse aşağıdaki ifadeler türetilebilir.

$$\frac{1}{w(C_Y + C_Z)} \geq 10R_X \tag{3.33}$$

$$w \leq \frac{1}{10R_X(C_Y + C_Z)} = w_H \tag{3.34}$$

Daha sonra, C_Y+C_Z göz ardı edildiğinde R_X direncinin sınırları aşağıdaki gibi elde edilecektir.

$$R_Y || R_Z \geq 10R_X \tag{3.35}$$

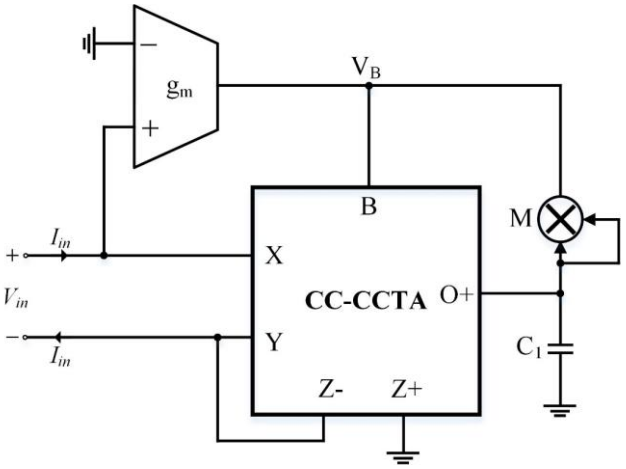
LTspice programı vasıtasıyla elde edilmiş olan, Tablo 3.1'de verilen parazitik bileşenlerin değerleri, parazitik etkiden kaynaklanan sınırlayıcı değerlerin değerlendirilmesi için analiz edilebilir. Bu değerler elde edilirken simülasyon ortamında kullanılan transistör boyutları, bayas akım değerleri gibi bilgiler 4. Bölümde açıklanmıştır.

Tablo 3.1: Parazitik bileşenlerin değerleri.

Parazitik bileşenler	Değerler
R_X	*700.36* Ω
R_Y	*14.15 k*Ω
R_{Z+}	*92.2 k*Ω
R_{Z-}	*90.13 k*Ω
R_O	*148.1 k*Ω
C_Y	*3.18 pF*
C_{Z+}	*322 fF*
C_{Z-}	*326 fF*
C_O	*200 fF*

3.3. KÜBİK FORM

Tasarlanan bir memristör emülatörünün, kaotik Chua devresinde kullanımında kaotik davranış seviyesinin ve lineer olmayışının artırılması adına, memristörün kübik formda olmasının katkı sağlayacağından daha önceki bölümlerde değinilmişti. Kübik formdaki bir memristör emülatörünün sağlayacağı katkıdan dolayı, bu tez kapsamında önerilen memristör emülatörleri Chua devresi için bu forma getirilerek kullanılacaktır. Yapıyı kübik forma getirmek için yapılması gereken tek işlem, iç yapısı 2. Bölümde verilmiş olan analog çarpıcı devresini memristör emülatörüne eklemek olacaktır. Kübik forma getirilmiş olan yüzen memristör emülatörü devresi Şekil 3.9'da verilmiştir.



Şekil 3.9: Kübik forma getirilmiş memristör emülatörü devresi.

Kübik memristör devresinin memdüktans değeri, (2.31) ile gösterilen denklem benzeri bir formda olması gerektiğinden bahsedilmişti. Bu aşamada Şekil 3.9 ile verilen devrenin bu forma uyup uymadığı incelenecektir.

Daha önceki yapıda C_1 kapasitörü üzerine düşen gerilim direkt olarak B portuna bağlanmıştı. Şimdiki yapıda eklenen çarpıcı devresinden dolayı bu çıkışın karesinin B portuna eklendiği görülmektedir. O halde aşağıda ifade edilen denklemlere ulaşıldığı görülmektedir.

$$V_{O+} = \frac{2g_m\varphi_{in}}{C_1} \tag{3.36}$$

$$V_B = V_{O+}^2 = \frac{4g_m^2\varphi_{in}^2}{C_1^2} \tag{3.37}$$

$$\frac{I_{in}}{V_{in}} = W(\varphi) = 8\sqrt{k_1k_2}\left(\frac{g_m^2\varphi_{in}^2}{C_1^2} - V_{SS} - V_{tn}\right) \tag{3.38}$$

(3.38)'de elde edilen memdüktans değeri yeniden düzenlenirse (3.39) denklemi elde edilir. Burada ifadeyi sadeleştirmek adına a ve b katsayıları kullanılmıştır.

$$\frac{I_{in}}{V_{in}} = W(\varphi) = a\varphi_{in}^2 - b \tag{3.39}$$

Elde edilen denklemlerden de görüldüğü gibi kübik form için elde edilmesi gereken denkleme benzer bir memdüktans denklemine ulaşılmıştır.

3.4. KAOTİK JERK DEVRESİ

Tez kapsamında önerilen memristör emülatörü kullanılarak tasarlanacak olan ilk kaotik sistem, Jerk devresi ile gerçekleştirilecektir. Jerk devresi ile ilgili temel bilgiler ikinci bölümde verilmişti. Bu aşamada Şekil 2.11'de verilen devre şeması ele alınacaktır. Bu devre detaylı olarak incelenecek olursa, devrede v_3 çıkışından R_9 direnci aracılığıyla doğrusal geri besleme sağlanmaktadır. Doğrusal olmayan geri besleme döngüsü, v_2 olarak gösterilen, ikinci integral alıcı devrenin çıkışına bağlanmış olan bir kuvvetlendirici yapısı aracılığıyla sağlanmaktadır.

Sunulan kaotik Jerk devresinin durum uzayı denklemlerini türetmek için öncelikle daha önce elde edilmiş olan (3.23) denklemi kullanılarak memristör üzerinden akan akım (3.40)'daki gibi elde edilmektedir.

$$i_M = V_M \frac{4\sqrt{k_1 k_2} g_m}{C_1} \varphi_{in} - V_M(V_{SS} + V_{tn}) \tag{3.40}$$

Elde dilen akım denklemindeki katsayılar a ve b olacak şekilde isimlendirilerek sadeleştirilecek olursa:

$$i_M = aV_M\varphi_{in} - bV_M \tag{3.41}$$

ifadesi elde edilmektedir. Daha sonra Jerk devresi göz önüne alınacak olursa aşağıdaki eşitlikler elde edilebilmektedir.

$$V_a = V_2 \frac{-R_{11}}{R_{10}} \tag{3.42}$$

$$V_4 = V_a + i_M R_1 \tag{3.43}$$

Bu aşamada önerilen memristör denklemlerinden yararlanılarak durum uzayı denklemleri yazılabilir. Dört boyutlu diferansiyel denklemler aşağıda görüldüğü gibi elde edilmiştir. Bu denklemlerde $x = V_1$, $y = V_2$, $z = V_3$ ve $w = \varphi_{in}$ olduğu varsayılmıştır.

$$\dot{x} = -\frac{1}{C_1}\left(\frac{x}{R_4} + \frac{z}{R_9} + \frac{1}{R_3}\left(-\frac{R_{11}}{R_{10}}y + R_1\left(-\frac{R_{11}}{R_{10}}y(aw - b)\right)\right)\right) \tag{3.43}$$

$$\dot{y} = -\frac{1}{C_2}\left(\frac{x}{R_5} + \frac{y}{R_6}\right) \tag{3.44}$$

$$\dot{z} = -\frac{1}{C_3}\left(\frac{y}{R_7} + \frac{z}{R_8}\right) \tag{3.45}$$

$$\dot{w} = -\frac{R_{11}}{R_{10}}y \tag{3.46}$$

Sunulan devrenin Jerk sistemine benzerliğini ortaya koymak için durum uzayı denklemlerinde bazı değişiklikler yapılması gerekmektedir. Öncelikle bu denklemler aşağıda görüldüğü gibi yeniden ele alınır.

$$\dot{x} = -\frac{x}{C_1R_4} - \frac{z}{C_1R_9} + \frac{R_{11}}{C_1R_3R_{10}}y + \frac{R_1R_{11}}{C_1R_3R_{10}}ywa - \frac{R_1R_{11}}{C_1R_3R_{10}}yb \tag{3.47}$$

$$\dot{y} = -\frac{x}{C_2R_5} - \frac{y}{C_2R_6} \tag{3.48}$$

$$\dot{z} = -\frac{y}{C_3R_7} - \frac{z}{C_3R_8} \tag{3.48}$$

Burada denklemleri sadeleştirmek adına aşağıdaki eşitlikler gerçekleştirilmektedir.

$$c = -\frac{1}{C_1R_4}, d = -\frac{1}{C_1R_9}, f = -\frac{1}{C_2R_5}, g = -\frac{1}{C_2R_6}, h = -\frac{1}{C_3R_7}, i = -\frac{1}{C_3R_8} \tag{3.50}$$

$$u = \frac{R_{11}}{C_1R_3R_{10}}y + \frac{R_1R_{11}}{C_1R_3R_{10}}ywa - \frac{R_1R_{11}}{C_1R_3R_{10}}y \tag{3.51}$$

Bu eşitlikler durum uzayı denklemlerinde yerlerine yazılırsa:

$$\dot{x} = cx + dz + u \tag{3.52}$$

$$\dot{y} = fx + gy \tag{3.53}$$

$$\dot{z} = hy + iz \tag{3.54}$$

denklemleri elde edilmektedir. Bu aşamada, işlemi basitleştirmek için R_6 ve R_8 dirençleri devreden çıkarılırsa g ve i katsayıları sıfır olacaktır. Bu varsayım kaotik davranışta değişikliklere sebep olacaktır, ancak kaotik davranış özelliği korunacaktır. Sonuç olarak durum uzayı denklemleri aşağıdaki gibi yazılır.

$$\dot{x} = cx + dz + u \tag{3.55}$$

$$\dot{y} = fx \tag{3.56}$$

$$\dot{z} = hy \tag{3.57}$$

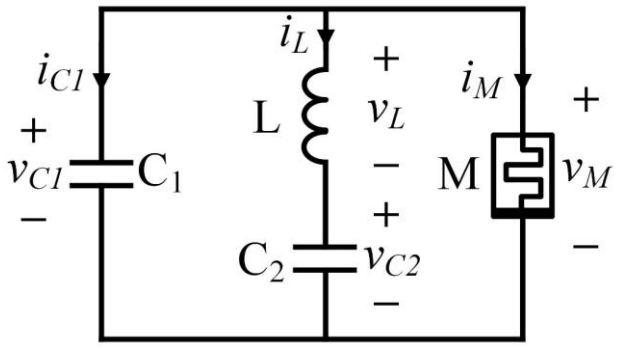
Buradan hareketle (3.58) denklemi elde edilebilmektedir.

$$\dddot{y} = c\ddot{y} + fdh\dot{y} + fu \tag{3.58}$$

Elde edilen (3.58) denklemi, sunulan kaotik devrenin Jerk sistemini gerçekleştirdiğini göstermektedir.

3.5. KAOTİK CHUA DEVRESİ

Tez kapsamında önerilen memristör emülatörü kullanılarak tasarlanacak bir diğer kaotik sistem Chua devresi vasıtasıyla gerçekleştirilecektir. Bunun için Şekil 3.10'da verilen Chua devresi kullanılacaktır. Bu devrede görülen memristör için kullanılacak yapı için önceki bölümlerde açıklanan kübik formdaki memristör emülatörü kullanılacaktır.



Şekil 3.10: Kaotik Chua devresi.

Memristör üzerinden akan akım değeri denklem (3.41)'de verilmişti. Memristör üzerine düşen gerilim değerinin ise şekilden de anlaşılacağı gibi C_1 kapasitörünün üzerine düşen gerilime eşit olduğu görülmektedir. C_1 kapasitöründe görülen gerilim değeri:

$$v_{C1} = -\frac{1}{sC_1}(i_L + i_M) \tag{3.59}$$

ifadesiyle bulunur. Daha sonra bu denklemde memristör akımı açık bir şekilde yazılacak olursa v_{C1} gerilimi aşağıdaki gibi elde edilir.

$$v_{C1} = -\frac{1}{sC_1}(i_L + av_M\varphi_{in}^2 - bv_M) \tag{3.60}$$

Devrede üzerinde gerekli olan diğer analizler Kirchoff yasaları çerçevesinde gerçekleştirilecek olursa aşağıda verilen denklemlere ulaşılabilmektedir.

$$\dot{v_{C1}} = -\frac{1}{C_1}(i_L + av_{C1}\varphi_{in}^2 - bv_{C1}) \tag{3.61}$$

$$\dot{v_{C2}} = -\frac{i_L}{C_2} \tag{3.62}$$

$$\dot{i_L} = -\frac{1}{L}(v_{C1} - v_{C2}) \tag{3.63}$$

$$\dot{\varphi_{in}} = v_{C1} \tag{3.64}$$

Burada elde edilen ifadeler için $v_{C1} = x$, $v_{C2} = y$, $i_L = z$ ve $\varphi_{in} = w$ olarak isimlendirilirse elde edilecek olan yeni durum denklemleri aşağıdaki gibi olacaktır.

$$\dot{x} = -\frac{1}{C_1}(z + axw^2 - bx) \tag{3.65}$$

$$\dot{y} = -\frac{z}{C_2} \tag{3.66}$$

$$\dot{z} = -\frac{1}{L}(x - y) \tag{3.67}$$

$$\dot{w} = x \tag{3.68}$$

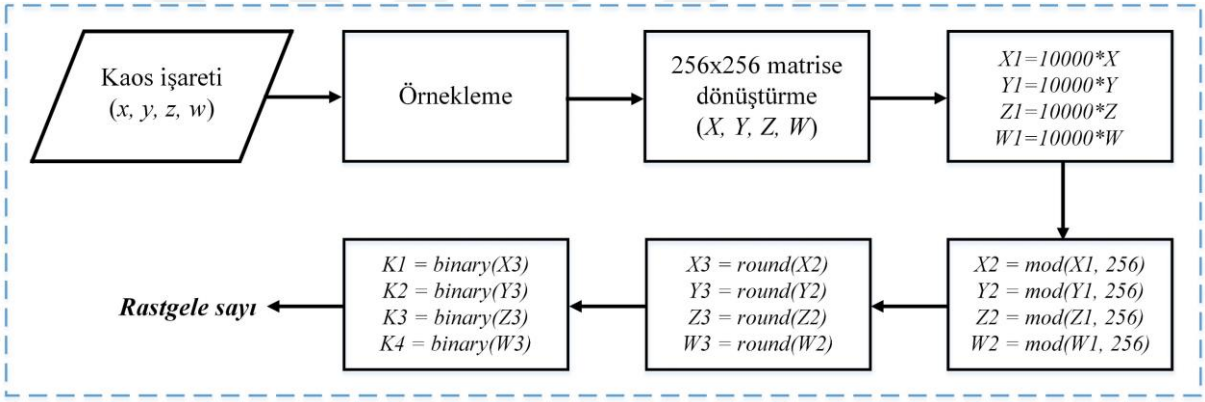
Burada elde edilen teorik sonuçların analizleri 4. Bölümde incelenecektir.

3.6. KAOTİK HABERLEŞME SİSTEMİ TASARIMI

Önerilen hafızalı devre elemanıyla elde edilen kaotik sistem kullanılarak haberleşme sisteminin tasarlanması bu bölümde gerçekleştirilecektir. İlk olarak görüntü şifreleme uygulamasında kullanılmak üzere üretilecek anahtarı elde etmek için tasarlanan kaos temelli RSÜ ele alınacaktır. Bu anahtar kullanılarak görüntü şifreleme işleminin algoritması açıklanacaktır. Daha sonra bilgi işareti, görüntünün sahip olduğu piksel değerlerinin sayısal verileri olarak ele alınıp bu işaretle kaotik maskeleme işlemi yapılarak temel kaotik haberleşme sistemi gerçekleştirilmiş olacaktır.

3.6.1. Kaos Tabanlı RSÜ Tasarımı

RSÜ tasarımı için kullanılacak yöntemin blok diyagram şeklinde gösterimi Şekil 3.11'de verilmiştir.



Şekil 3.11: RSÜ tasarımı akış şeması.

Kübik memristör devresi kullanılarak tasarlanan Chua devresiyle elde edilen kaotik işaretler RSÜ tasarımının girdilerini oluşturmaktadır. Bu analog işaret örnekleme işlemi ile ayrık işarete dönüştürüldükten sonra, bu ayrık kaotik değerler, *256x256* boyutlu matrislere dönüştürülmektedir. *256x256* boyutuna dönüştürülme sebebi, şifrelenecek işaretin *256x256* boyutunda bir görüntü olmasıdır. Daha sonra elde edilen değerlerin sayısal olarak daha uygun düzeylere çekilmesi için matris *10000* ile çarpılarak değerler yükseltilmektedir. Yeni matrislerin modu alınıp en yakın tam sayı değerine yuvarlandıktan sonra ikili sayı sistemi formuna getirilmektedir. Elde edilen sayısal değerler rastgele sayılardır. Bu sayıları bulunduran matrisler ise şifreleme sisteminde kullanılacak olan anahtarlardır.

3.6.2. Kaotik Görüntü Şifreleme Uygulaması

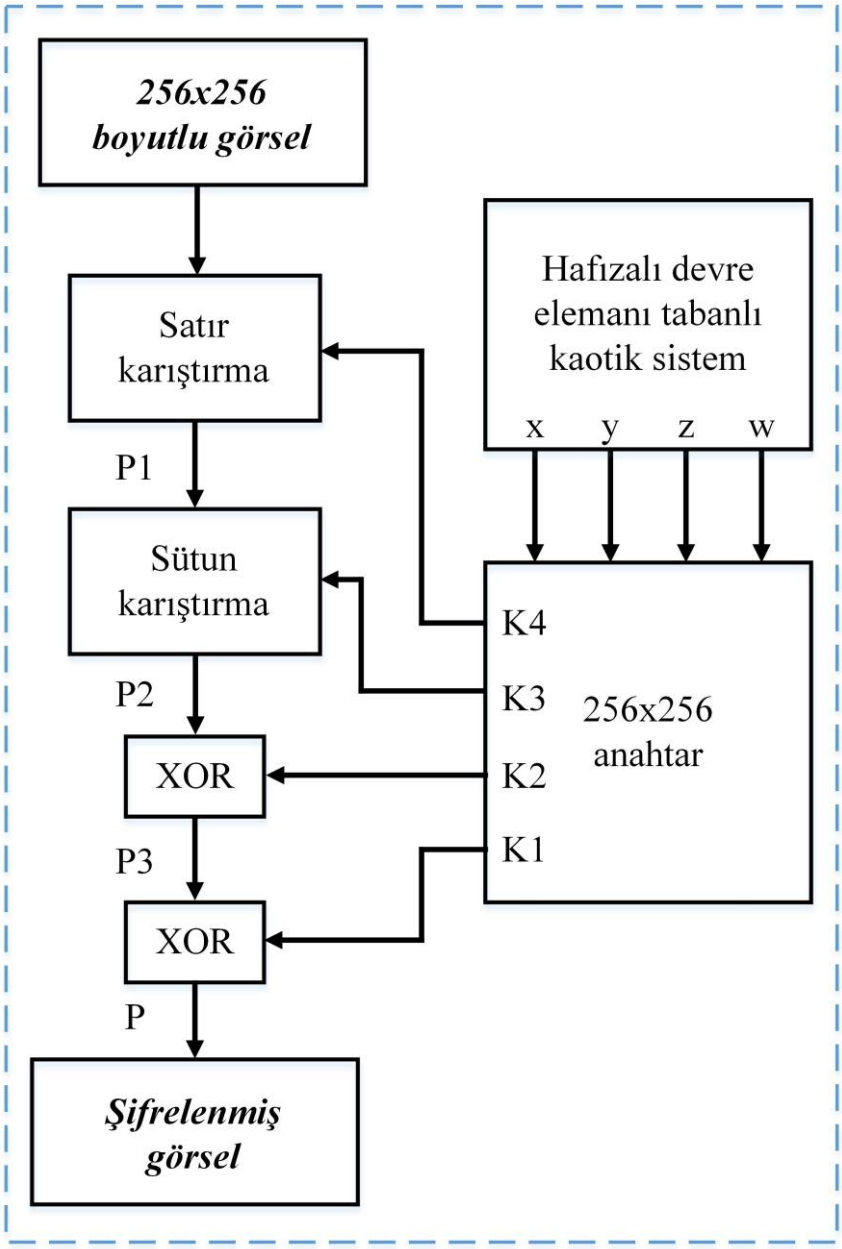
Tasarlanan kaotik Chua devresi kullanılarak, bir görüntünün şifrelenmesi işlemi Şekil 3.12'de verilen blok diyagramı ile açıklanmıştır. Sistemde kullanılan, *K1, K2, K3* ve *K4* olarak isimlendirilmiş anahtarlar Şekil 3.11'de verilen yapıdan, MATLAB programı vasıtasıyla elde edilmiştir.

Sistem detaylı olarak incelenecek olursa, ilk olarak *K4* anahtarı kullanılarak satır karıştırma işlemi gerçekleştirilmiştir. Karıştırma işleminin neticesinde elde edilen veri *P1* olarak isimlendirilmiştir. *P1* olarak isimlendirilmiş bu veri, yine *256x256* boyutunda bir matristir. Bu matris için yapılan sütun karıştırma işlemi için *K3* anahtarı kullanılmıştır. Buradan da elde edilen yeni matris *P2* olarak isimlendirilmiştir. Şifreleme işleminin bir sonraki basamağında *P2* matrisinin her biti ile *K2* anahtarının her biti *XOR* işlemine sokulmuştur. Bu işlemin neticesinde elde edilen *P3* matrisi ile de *K1* anahtarı aynı şekilde *XOR* işlemine tabi tutulmuştur. Neticede elde edilen *P* matrisi şifrelenmiş görüntüyü vermektedir.

Şifre çözme işleminin basamakları Şekil 3.13'te verilen blok diyagramla gösterilmiştir. Tasarlanan şifreleme algoritmasına göre, şifrelemede kullanılan anahtarların aynıları şifreyi çözmek için kullanılmaktadır. Şifreleme işlemini gerçekleştirmek için izlenen yolun tam tersi şifre çözme için izlenmektedir. Şifrelenmiş metnin *K1* anahtarı kullanılarak *P3* verisinin elde edilmesi için *K1* anahtarı ile *P* matrisi *XOR* işlemine sokulmaktadır. Aynı şekilde diğer adımlar da şifreleme aşamasındakinin tersi olacak şekilde gerçekleştirilerek şifre çözme işlemi gerçekleştirilmiş olur.

Bu şekilde gerçekleştirilen şifreleme ve şifre çözme işlemlerinin neticesinde elde edilen şifrelenmiş ve şifresi çözülmüş görsellerin MATLAB programıyla elde edilmiş verileri 4. Bölümde ele alınacaktır. Görüntü şifrelemek için kullanılan algoritmadaki basamak sayısı azaltılabilmektedir. Örnek olarak, kaotik sistemden elde edilmiş anahtarlardan yalnızca birinin kullanılmasıyla da görüntü şifreleme işleminin gerçekleştirilmesi mümkün olabilmektedir. Ancak bu şekilde gerçekleştirilen kaotik şifrelemenin güvenilirlik düzeyi daha karmaşık algoritmalara göre oldukça zayıf olacaktır. Bu şekilde elde edilen şifrelenmiş görüntülerin nasıl olduğuna daha sonra değinilecektir. Tez kapsamında sunulan kaotik Chua devresi dört boyutlu olduğundan, görüntü şifreleme algoritması için dört farklı anahtarın kullanıldığı bir sistemin tasarlanması uygun görülmüştür. Tez içerisinde değinilen bir diğer kaotik yapı olan Jerk devresi

ise haberleşme sisteminde, bir sonraki konu olan, kaotik maskeleme aşamasında bu yapı kullanılacaktır.



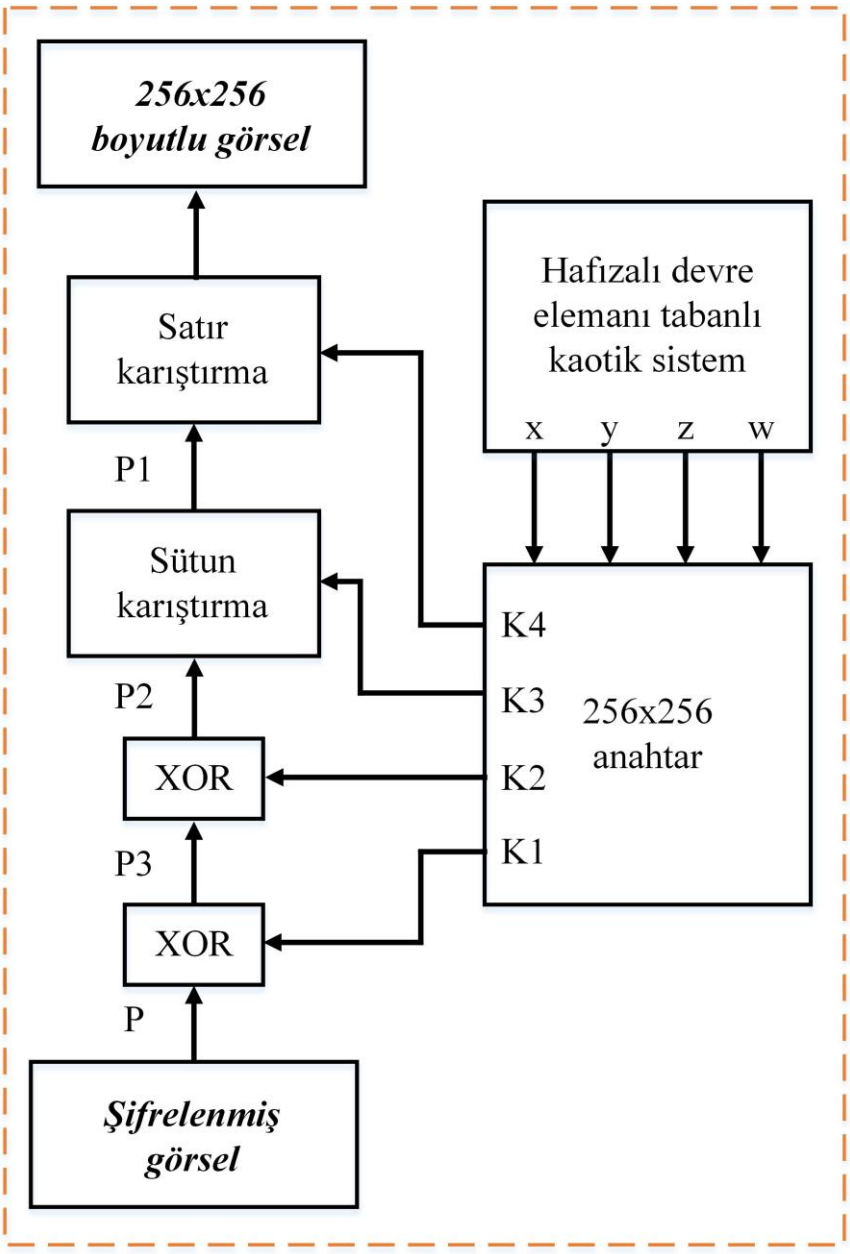
Şekil 3.12: Kaotik görüntü şifreleme işleminin blok diyagramı.

3.6.3. Kaotik Maskeleme Uygulaması

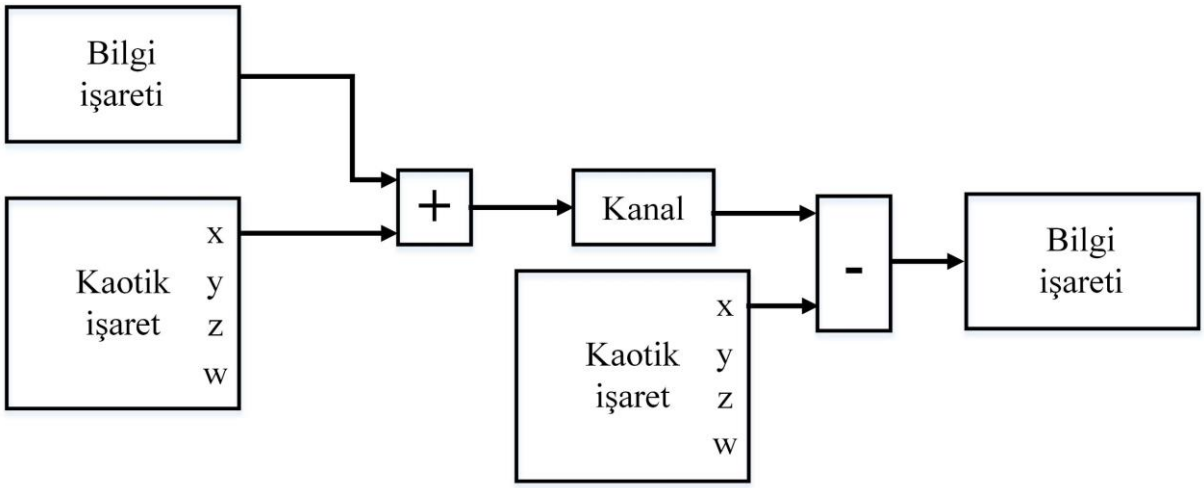
Kaotik maskeleme işleminin en basit anlatımla, bilgi işareti ile kaotik işaretin verici basamağında işleme sokulup, daha sonra kanal yoluyla bu maskelenmiş işaretin alıcı kısma iletilerek aynı kaotik işaretle demodüle edilmesiyle, bilgi işaretinin tekrar elde edilmesi olduğundan bahsedilmişti. Buradaki bilgi işareti, şifreleme uygulamasında görüntü bilgisi

kullanıldığından, *256x256* boyutundaki görüntünün piksellerinin ardışık olarak sıralanıp sayısal bilgi olarak kullanılan tek satırlık bir işaret olduğu düşünülebilir.

Uygulamada kullanılacak olan kaotik maskeleme sistemini blok şeması Şekil 3.14'te verilmiştir. Bu yöntemde, şekilden de görüldüğü gibi, kaotik işaret taşıyıcı olarak kullanılmaktadır. Bu sistemin MATLAB Simulink uygulaması vasıtasıyla gerçekleştirilmiş olan benzetim sonuçlarına 4. Bölümde değinilecektir.



Şekil 3.13: Kaotik şifre çözme işleminin blok diyagramı.



Şekil 3.14: Kaotik maskeleme işleminin blok diyagramı [107].

4. BULGULAR

4.1. MEMRİSTÖR EMÜLATÖRÜ ANALİZİ

Bu bölümde önerilen memristör emülatörünün analizleri LTspice programı yardımıyla elde edilmiş benzetim sonuçları vasıtasıyla gerçekleştirilecektir.

4.1.1. Zaman Analizi ve Histerisiz Eğrileri

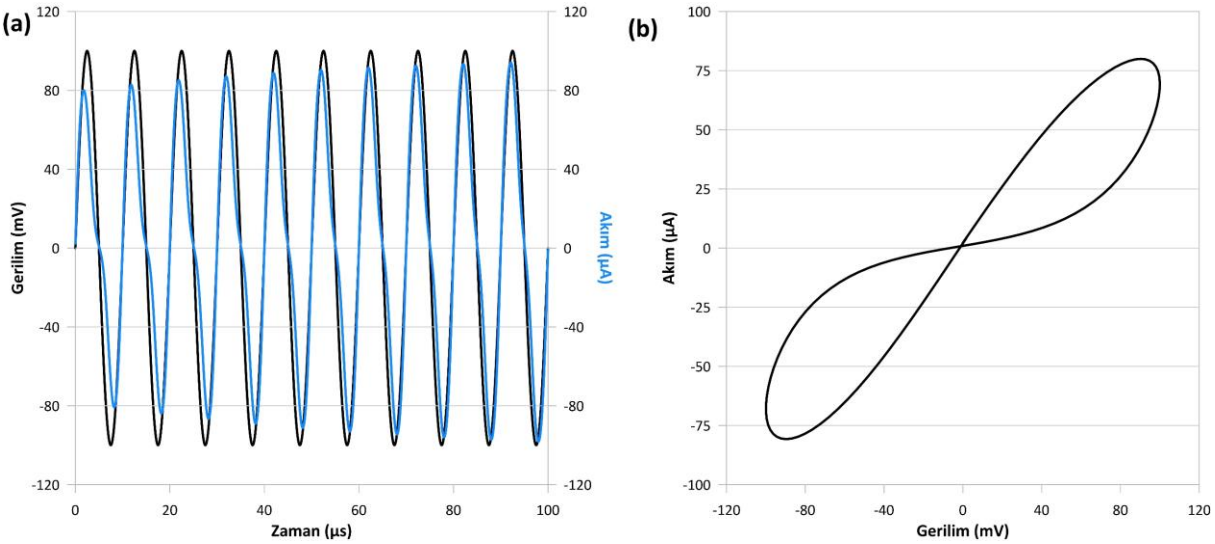
Önerilen devre TSMC 180 nm teknolojisi ile gerçekleştirilmiştir. Şekil 3.1'de verilen iç devreye göre MOSFET'lerin uzunluk ve genişlik değerleri Tablo 4.1'de verilmiştir. DC besleme gerilimleri ve bayas akımı $V_{DD} = -V_{SS} = 0{,}9\ V$ ve $I_B = 100\ \mu A$ olarak seçilmiştir.

Tablo 4.1: Transistör boyutları.

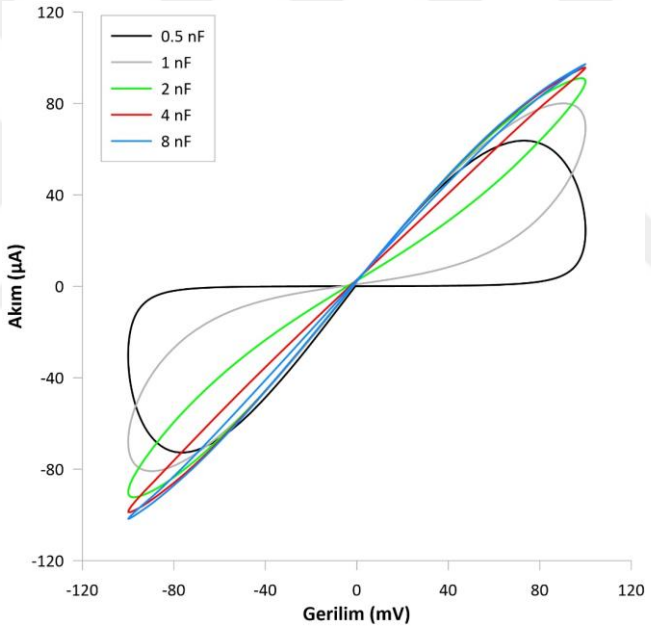
Transistör isimleri	W (µm)	L (µm)
M_1, M_2	7	0.35
M_3, M_4	28	0.35
M_5 - M_{11}	16	0.5
M_{12} - M_{15}	25	0.8
M_{16} - M_{22}	5	0.5
M_{23}, M_{24}	6.2	0.8
M_{25}, M_{26}	10	0.5

Memristör emülatörünün girişine *100 kHz* frekansa ve *100 mV* tepe voltajına sahip sinüzoidal giriş sinyali uygulandığında Şekil 4.1 elde edilmektedir. Şekil 4.1a'da giriş gerilimi ve giriş akımının zamana göre değişimleri görülmektedir. Şekil 4.1b'de gerilim ve akım arasındaki histerisiz döngüsüne yer verilmiştir. Burada elde edilen döngünün şekli memristör devresinden beklendiği gibidir. Bu analizi yapılırken C_1 kapasitör değeri *1 nF* olarak seçilmiştir. İletkenlik (g_m) ve parazit direnç (R_X) değerleri sırasıyla *662,16 µS* ve *700,36 Ω* olarak hesaplanmıştır.

Şekil 4.2'de farklı kapasitör değerleri test edilmiştir. Bu simülasyonda girişe uygulanan sinüzoidal sinyalin frekansı *100 kHz*, tepe gerilimi ise *100 mV* olarak seçilmiştir. Şekil 4.2'de kapasitör değeri arttığında histerezis döngüsünün daraldığı görülmektedir. Bu sonuçlar, simülasyon sonuçlarının denklem (3.23) ile gösterilen teorik analizi doğruladığını göstermektedir.



Şekil 4.1: Memristör emülatörünün zaman cevabı (a) ve histerisiz eğrisi (b).

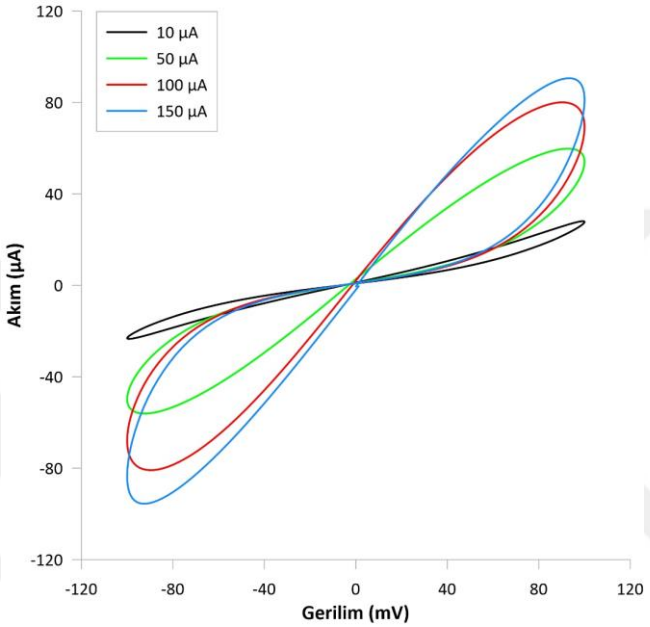


Şekil 4.2: 100 kHz frekansta farklı kapasitör değerleri için elde edilen histerisiz eğrileri.

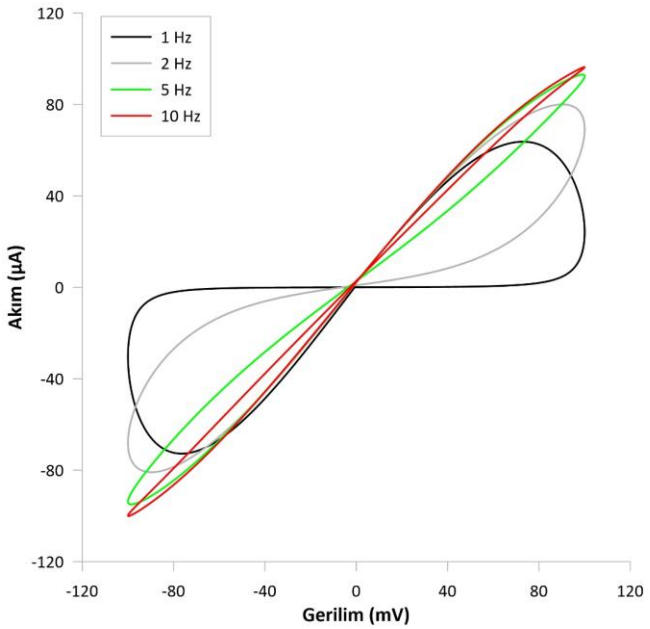
Değişen bayas akımının (I_B) etkilerini gözlemlemek için başka bir benzetim daha gerçekleştirilmiştir. Buradan elde edilen sonuçlara göre, I_B akımının arttırılmasıyla bağlantılı olarak eğrinin genişlediği gözlemlenmiştir. Bu sonuç Şekil 4.3'te gösterilmektedir. Bu benzetimde giriş frekansı ve gerilimi önceki benzetimle aynı seçilmiştir.

Frekansların değiştirilerek farklı kapasitör değerlerinin kullanılmasıyla diğer simülasyon sonuçları elde edilmiştir. Bu sonuçlar Şekil 4.4, Şekil 4.5 ve Şekil 4.6'da verilmiştir. Şekil 4.4'te düşük frekanslar test edilmiştir. Bunun için C_1 kapasitör değeri $50\ \mu F$ olarak seçilmiştir. Şekil 4.5'te girişe orta frekanslar uygulanmıştır. Bu simülasyonda C_1 kapasitör değeri $1\ nF$ olarak

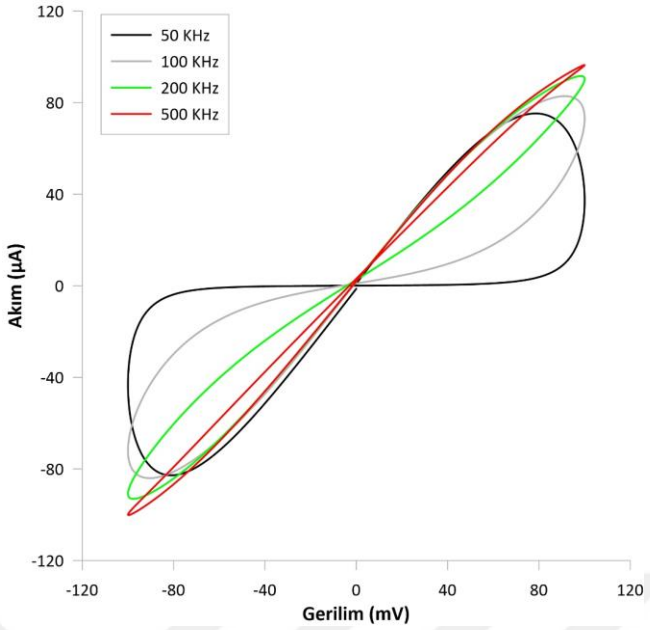
seçilmiştir. Önerilen memristör emülatörünün yüksek frekans tepkisini gözlemlemek adına Şekil 4.6'da verilen benzetim sonuçlarına ulaşılmıştır. Bu sonuç için $C_1 = 10\ pF$ olarak alınmıştır. (3.21) ve (3.23) denklemleri incelendiğinde, frekans artırılırken benzer bir histerezis döngüsü sağlamak için kapasitör değerinin azaltılması gerektiği görülmektedir. Ek olarak, frekans arttıkça hafızanın daha yüksek çalışma frekanslarında doğrusal dirence benzeme eğiliminde olduğu da gözlemlenmiştir.



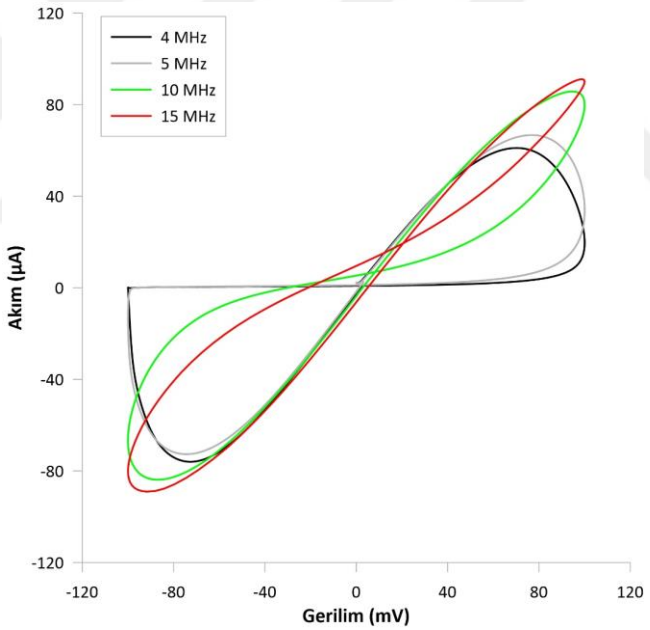
Şekil 4.3: 100 kHz frekansta farklı bayas akımı değerleri için elde edilen histerisiz eğrileri.



Şekil 4.4: Farklı giriş frekansları ile elde edilen histerisiz eğrileri, $C_1 = 50\ \mu F$.



Şekil 4.5: Farklı giriş frekansları ile elde edilen histerisiz eğrileri, $C_1 = 1\ nF$.

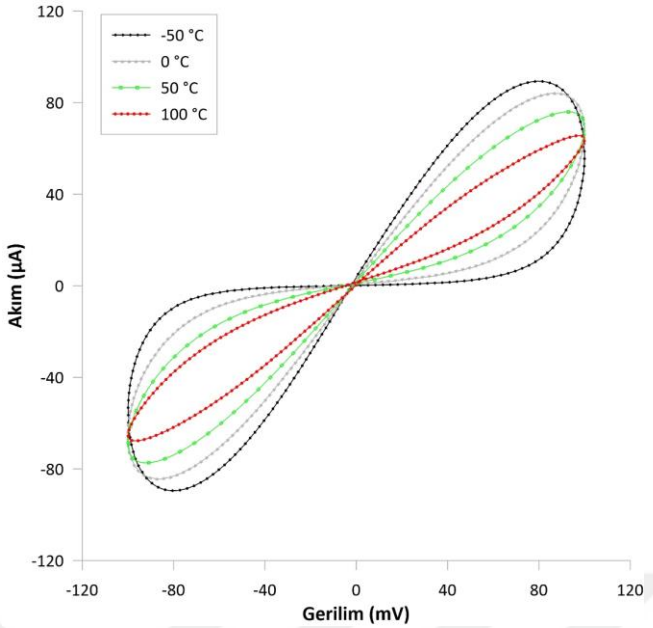


Şekil 4.6: Farklı giriş frekansları ile elde edilen histerisiz eğrileri, $C_1 = 10\ pF$.

4.1.2. İleri Analizler

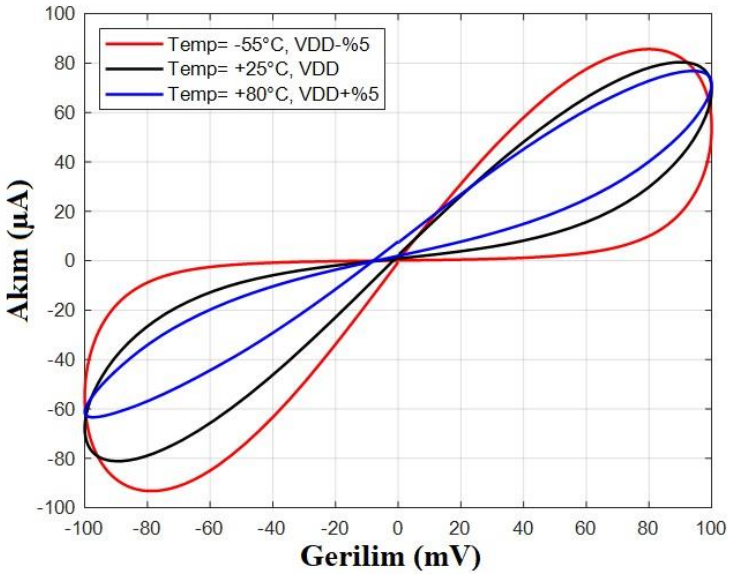
4.1.2.1. Sıcaklık Analizleri

Önerilen devrenin farklı sıcaklıklara karşı gösterdiği tepkiler Şekil 4.7'de gösterilmektedir. Bu simülasyonda sıcaklık arttığında histerezis döngüsünün daraldığı görülmektedir. Ayrıca bu sonuçlara bakılarak, uç noktalardaki sıcaklık değişimleri histerezis döngüsünde önemli değişikliklere neden olmadığı soncuna ulaşılabilmektedir.



Şekil 4.7: Farklı sıcaklıklarda elde edilen histerisiz eğrileri.

Şekil 4.8, sıcaklık değişimlerinin yanı sıra besleme gerilimindeki ±%5'lik bir değişim için histerezis eğrisini göstermektedir. Histerezis eğrisinde önemli bir değişiklik olmadığı grafikten açıkça görülmektedir.

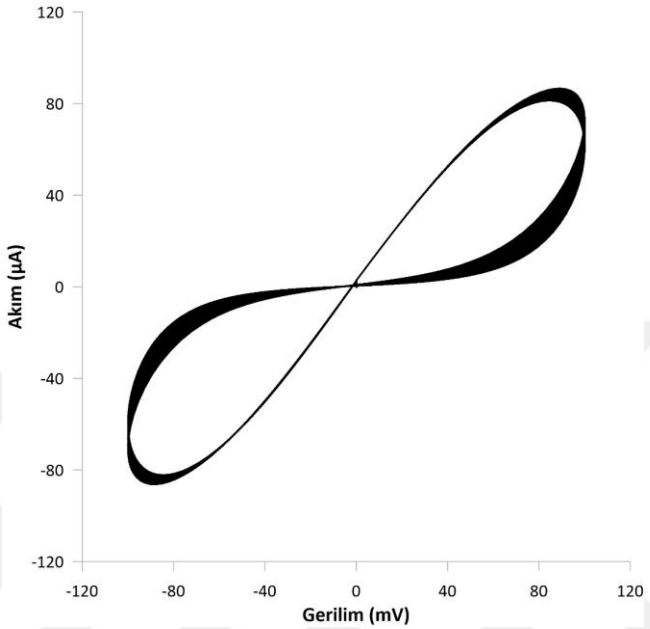


Şekil 4.8: sıcaklık değişimleriyle besleme gerilimindeki ±%5'lik bir değişim için elde edilen histerezis eğrisi.

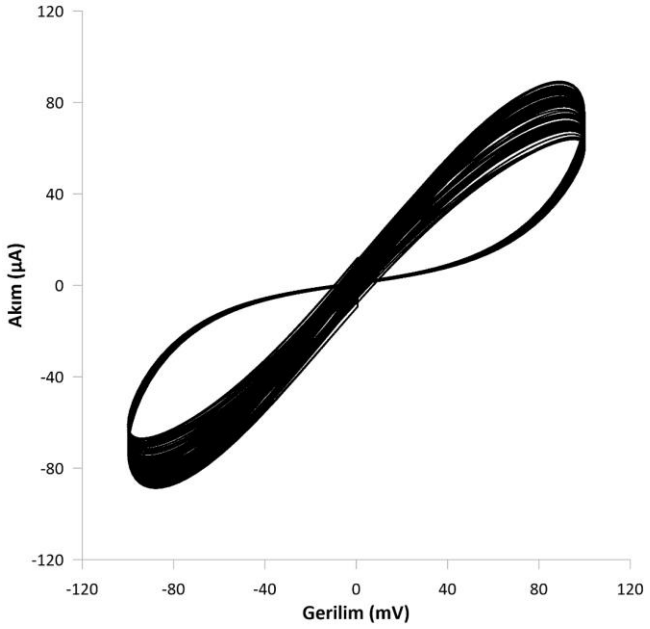
4.1.2.2. Monte Carlo Analizi

Önerilen memristör emülatöründeki uyumsuzlukları değerlendirmek için Monte Carlo benzetimleri yüz kez gerçekleştirildi. Bu analiz için, C_1 kapasitörü ve besleme gerilimleri

oranları %10 oranında eşit şekilde değiştirilmiştir. Benzetim sonuçları Şekil 4.9 ve Şekil 4.10'da gösterilmektedir. Memristör emülatörünün histerezis döngüsünde hafif değişiklikler gözlemlenmektedir. Elde edilen verilere göre önerilen memristörün kabul edilebilir sınırlar içerisinde çalışabildiği görülmektedir.



Şekil 4.9: Memristör emülatörünün Monte Carlo analizinden elde edilen bulgular, %10'luk değişimde kapasitör değerlerinde tutarlı değişikliklerle yüz kez yinelemeli olarak gerçekleştirildi.



Şekil 4.10: Memristör emülatörü üzerinde gerçekleştirilen Monte Carlo analizinin sonuçları, besleme gerilimleri %10'luk bir değişim içeren yüz tekrar ile uygulanmıştır.

4.2. KARŞILAŞTIRMA

Literatüre sunulmuş birçok memristör emülatörü devresi bulunmaktadır. Bunların bir kısmının bu tez kapsamında sunulan memristör emülatörü ile karşılaştırılmasını Tablo 4.2 ile verilmektedir.

Tablo 4.2: Memristör emülatörleri karşılaştırma tablosu.

Referans	Kontrol	Aktif eleman	Pasif eleman	Yüzen / Topraklanmış	İşlem frekansı	Güç harcaması	Elektronik olarak kontrol edilebilirlik
[42]	Gerilim	2 OP-AMP 2 Çarpıcı	3 Direnç 1 Kapasitör	Topraklanmış	-	-	Hayır
[23]	Yük	1 CCTA	3 Direnç 1 Kapasitör	İkisi de	10 MHz	7.5 mW	Evet
[58]	Akı	7 MOSFET	1 Kapasitör	Topraklanmış	50 MHz	-	Hayır
[59]	Akı	1 VDTA 1 MOSFET	-	Topraklanmış	50 MHz	-	Evet
		LM13700	2 Direnç 1 Kapasitör	Topraklanmış	6 kHz		
[29]	Akı	2 OTA	1 Kapasitör	İkisi de	8 MHz	-	Evet
[21]	Akı	3 MOSFET	1 Kapasitör	Topraklanmış	13 MHz	6.725 nW	Evet
[38]	Yük	3 CCII 1 Çarpıcı	2 Direnç 1 Kapasitör	İkisi de	-	-	Hayır
[61]	Akı	2 VDCC 2 MOSFET	1 Kapasitör	Topraklanmış	2 MHz	-	Evet
[38]	Yük	1 CCII 1 CCTA	3 Direnç 1 Kapasitör	Yüzen	5 MHz	-	Hayır
		4 AD844 1 CA3080	3 Direnç 1 Kapasitör	Yüzen	100 kHz		
[94]	Akı	2 CNTFET OTA	1 Kapasitör	İkisi de	1 GHz	-	Hayır
[63]	Akı	1 VDTA	1 Direnç 1 Kapasitör	Yüzen	50 MHz	8 µW	Hayır
[95]	Akı	1 CCCII	1 Kapasitör	Topraklanmış	7.5 MHz	-	Hayır
[18]	Yük	1 CFTA	1 Kapasitör	Topraklanmış	9 MHz	-	Hayır
		1 AD844AN 2 CA3080	1 Direnç 1 Kapasitör	Topraklanmış	995 kHz		
[66]	Akı	2 VDIBA	1 Kapasitör	Yüzen	12.7 MHz	1.34 mW	Evet
[20]	Akı	1 OTA 1 DVCC	1 Direnç 1 Kapasitör	Topraklanmış	1 MHz	-	Evet
[25]	Akı	4 MOSFET	-	Yüzen	50 MHz	2.6 µW	Hayır
[22]	Gerilim	1 MOSFET	1 Direnç 1 Kapasitör	Yüzen	80 MHz	Zero	Hayır
[67]	Akı	1 CFDITA	1 Kapasitör	İkisi de	10 MHz	-	Hayır
Bu çalışma	Akı	**1 CC-CCTA**	**1 Kapasitör**	**Topraklanmış**	**15 MHz**	**7.2 µW**	**Evet**
		1 CC-CCTA 1 OTA	**1 Kapasitör**	**Yüzen**			

4.3. MEMRİSTÖR TABANLI KAOTİK DEVRE

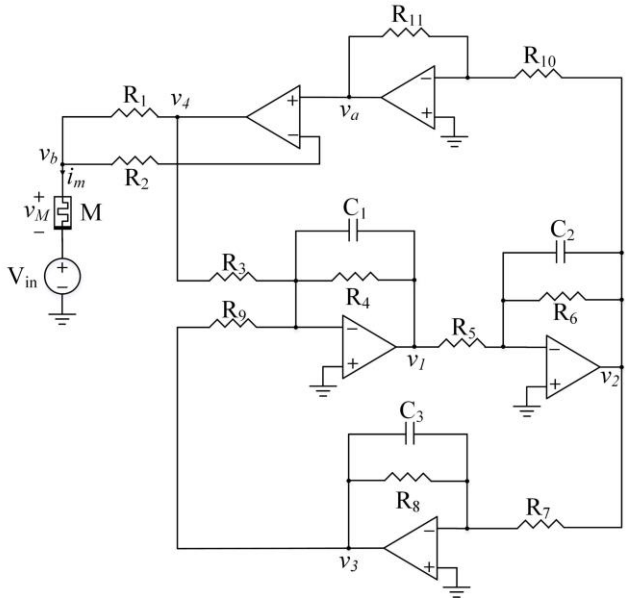
Bu bölümde tez kapsamında önerilen memristör emülatörü kullanılarak gerçekleştirilen kaotik devreler incelenecektir. İlk aşamada tasarlanan memristör ile Jerk sisteminin gerçekleştirilmesi LTspice programı kullanılarak yapılan benzetimlerle açıklanacaktır. Daha sonra önerilen memristör emülatörü kullanılarak Chua devresi tasarlanacaktır. Bu tasarımın kaotik oluşu MATLAB programıyla elde edilen bulgularla ispatlanacaktır.

4.3.1. Önerilen Memristör Emülatörü ile Jerk Devresi Tasarımı

Kullanılacak olan Jerk devresi Şekil 4.11'de görülmektedir. Bu devrede v_3 çıkış düğümünden R_9 direnci aracılığıyla doğrusal geri besleme sağlanmaktadır. Doğrusal olmayan geri besleme döngüsü, v_2 olarak gösterilen ikinci bir integratör devresinin çıkışından gelen bir kuvvetlendirici yapısı aracılığıyla sağlanmaktadır.

Tüm simülasyonlar Tablo 4.3'te belirtilen parametreler kullanılarak gerçekleştirilecektir. Bu devrenin matematiksel analizi Bölüm 3.4'te yapılmıştı. Denklemler içerisinde bulunan a ve b değerleri tasarlanan memristör emülatöründen elde edilen ifadelerdir. Bu ifadelerin değerleri LTspice aracılığıyla hesaplanıp Tablo 4.3'te verilmiştir.

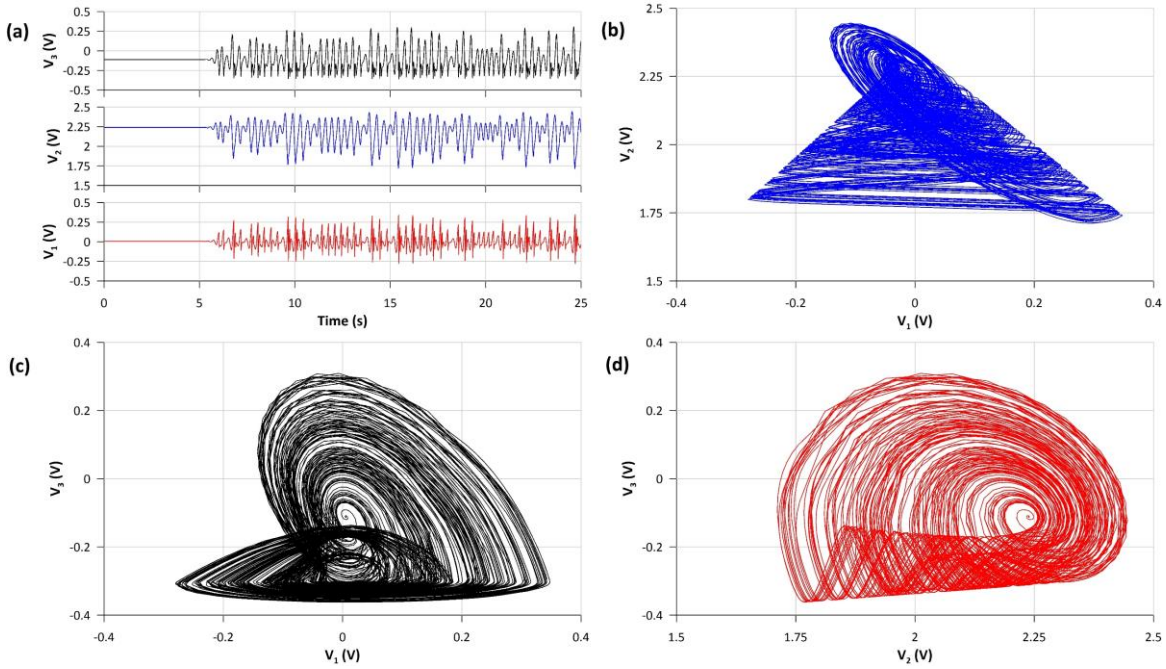
Sunulan Jerk devresindeki çıkış sinyalinin zamana göre değişimleri LTspice programı kullanılarak benzetimleri yapılmıştır. Benzetim sonuçları Şekil 4.12a'da verilmiştir. Kaotik devrenin faz portreleri de aynı şekil üzerinde görülmektedir.



Şekil 4.11: Kaotik Jerk devresi [33].

Tablo 4.3: Jerk devresi parametre değerleri.

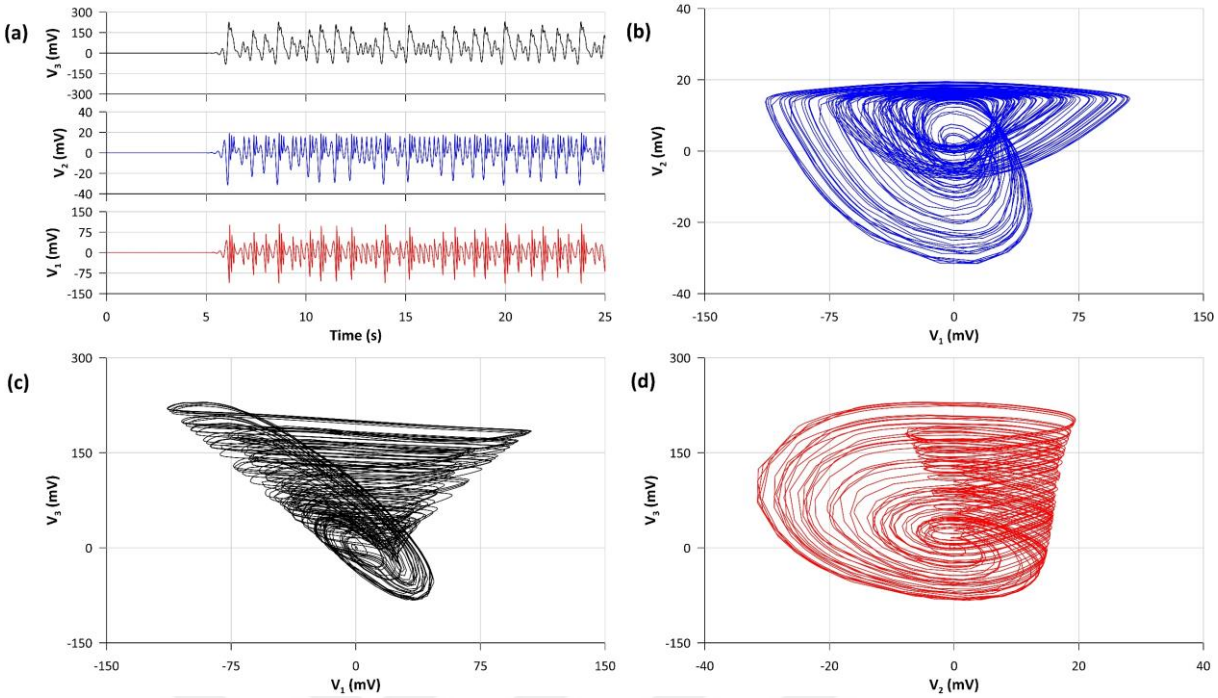
Parametre adı	Parametre değeri
R_1	$50\ k\Omega$
R_2, R_6	$10\ k\Omega$
R_3	$3\ k\Omega$
R_4, R_5, R_7, R_9	$1\ k\Omega$
R_8	$100\ k\Omega$
R_{10}	$50\ k\Omega$
R_{11}	$150\ k\Omega$
C_1, C_2	$100\ \mu F$
C_3	$10\ \mu F$
a	142.3
b	-0.168



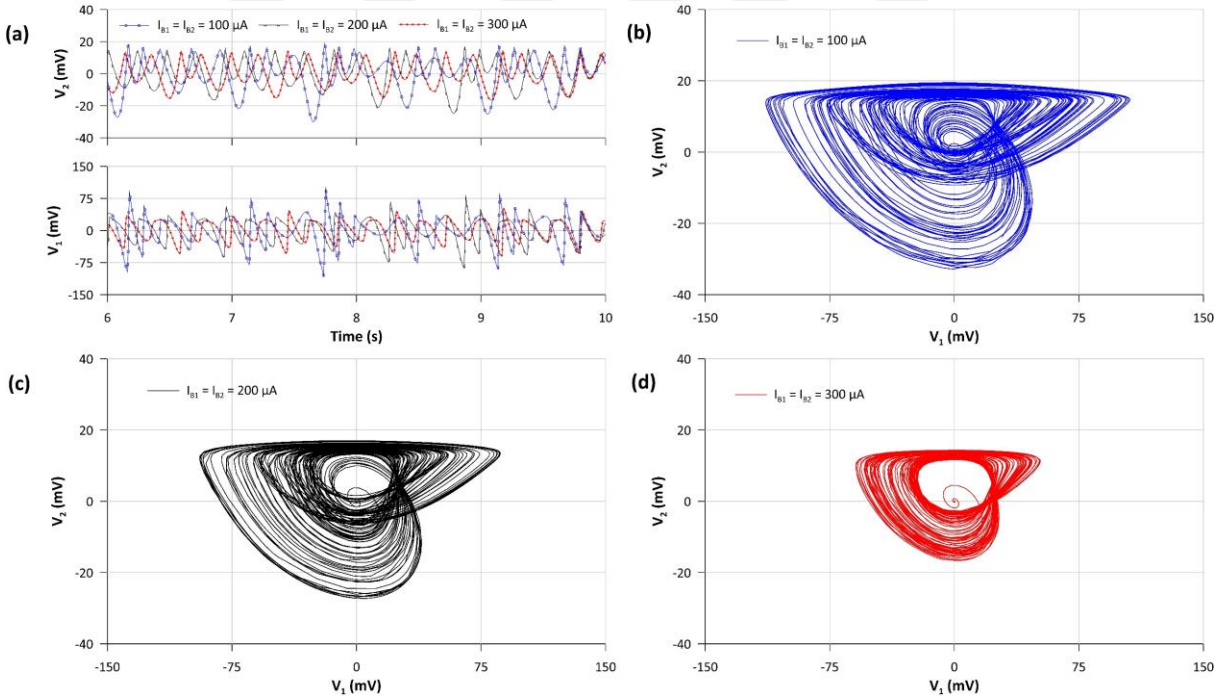
Şekil 4.12: Jerk devresinden elde edilen V_1, V_2 ve V_3 gerilimlerinin zamana göre değişimleri (a) ve faz portreleri (b), (c), (d).

Bölüm 3.4'te, Jerk devresinde verilen R_6 ve R_8 dirençlerinin devreden çıkarılmasıyla devrenin kaotik davranmaya devam edileceğinden bahsedilmişti. Bu durumu gözlemlemek amacıyla bu dirençlerin devreden çıkarılmasıyla elde edilen sonuçları Şekil 4.13'te verilmiştir.

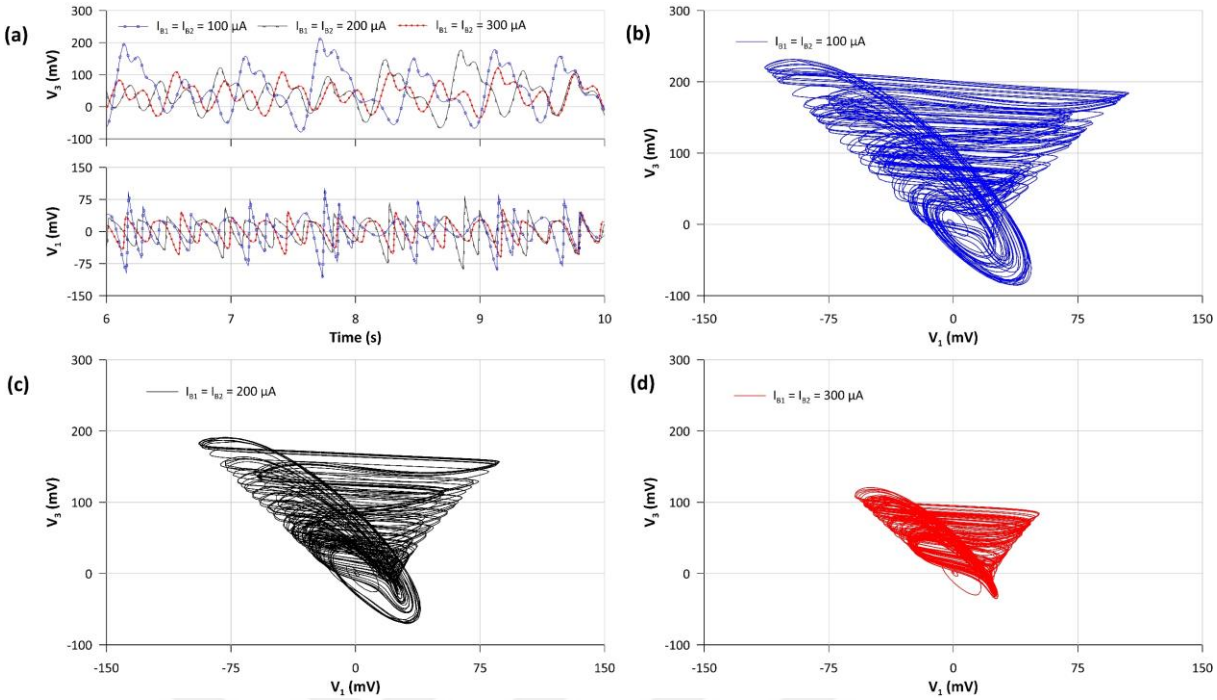
Bayas akımının (I_{B1} ve I_{B2}) kaotik osilatördeki etkisini araştırmak adına farklı akım değerleri kullanılarak benzetimler yapılmıştır. Bu bağlamda bayas akımları $I_{B1} = I_{B2} = 100\ \mu A$, $200\ \mu A$ ve $300\ \mu A$ olarak değiştirildiğinde, V_1-V_2, V_1-V_3 ve V_2-V_3 gerilimleri arasındaki ilişki sırasıyla Şekil 4.14, Şekil 4.15 ve Şekil 4.16'da gözlemlenebilir. Bu şekillerde değişen bayas akımlarıyla ilişkili zaman değişkenlerine de yer verilmiştir.



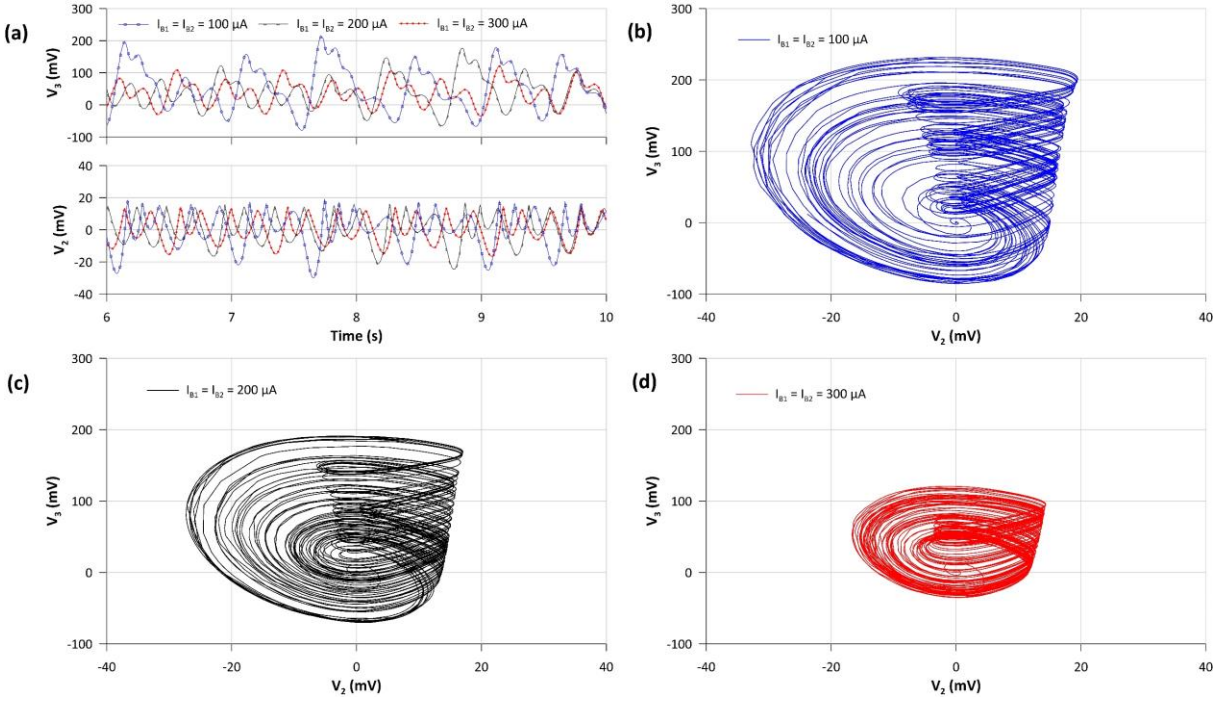
Şekil 4.13: R_6 ve R_8 dirençleri çıkarılmış Jerk devresinden elde edilen V_1, V_2 ve V_3 gerilimlerinin zamana göre değişimleri (a) ve faz portreleri (b), (c), (d).



Şekil 4.14: Sunulan Jerk devresinde farklı bayas akımlarıyla, V_1 ve V_2 gerilimlerinin zamana bağlı değişimleri (a) ve bu iki gerilimin arasındaki ilişkiden elde edilen, farklı bayas akımlarındaki faz portreleri (a), (b), (c).



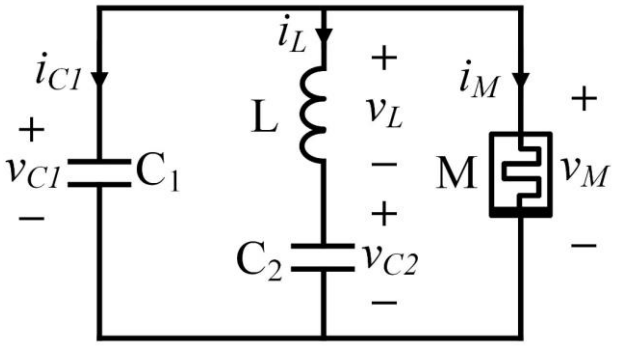
Şekil 4.15: Sunulan Jerk devresinde farklı bayas akımlarıyla, V_1 ve V_3 gerilimlerinin zamana bağlı değişimleri (a) ve bu iki gerilimin arasındaki ilişkiden elde edilen, farklı bayas akımlarındaki faz portreleri (a), (b), (c).



Şekil 4.16: Sunulan Jerk devresinde farklı bayas akımlarıyla, V_2 ve V_3 gerilimlerinin zamana bağlı değişimleri (a) ve bu iki gerilimin arasındaki ilişkiden elde edilen, farklı bayas akımlarındaki faz portreleri (a), (b), (c).

4.3.2. Önerilen Memristör Emülatörü ile Chua Devresi Tasarımı

Memristör emülatörünün Chua devresinde kullanımında, daha önce de analizi yapılmış olan Şekil 4.17'de verilen devre kullanılacaktır. Bu devre şekilden de görüleceği gibi Jerk devresine göre daha sadedir. Ancak bu devre ile elde edilecek kaotik sistemin kaotik davranışının yüksek düzeyde olabilmesi için, kullanılacak memristör emülatörünün kübik forma getirilmesi uygun görülmektedir. Bu durum, Chua devresinin basit yapıda olmasının yanı sıra memristör emülatörünün daha karmaşık özellikte olmasına sebep olmaktadır.



Şekil 4.17: Önerilen memristörün kullanılacağı Chua devresi.

Verilen Chua devresinin analizi sonucunda elde edilen durum denkleri, denklem (3.65) ile (3.68) aralığında verilmişti. Burada verilen parametrelerin değerleri, MATLAB uygulamasıyla gerçekleştirilecek uygulamasında kullanılmak üzere, Tablo 4.4'te verilmiştir. Bu değerler, kaotik davranışın gözlemlendiği aralığı sağladığı için tercih edilmiştir ve kaotik davranışın gözlemlenmesinde bu değerler dikkate alınarak analizler gerçekleştirilmiştir.

Tablo 4.4: Chua devresi parametre değerleri.

Parametre adı	Parametre değeri
C_1	*0.08*
C_2	*0.021*
L	*0.5*
a	*142.3*
b	*-0.168*

4.3.2.1. Kararlılık Analizi

Bu kısımda kararlılık analizi, önerilen kaotik sistem için gerçekleştirilecektir. İlk olarak, daha önceden (3.65) ile (3.68) aralığındaki denklemlerle verilmiş olan durum denklemleri kullanılarak, her bir ifadenin sıfıra eşitlenmesiyle denge noktaları elde edilmedir. Bu eşitlikler aşağıdaki gibi verilmiştir.

$$0 = -\frac{1}{C_1}(z + axw^2 - bx) \tag{4.1}$$

$$0 = -\frac{z}{C_2} \tag{4.2}$$

$$0 = -\frac{1}{L}(x - y) \tag{4.3}$$

$$0 = x \tag{4.4}$$

Elde edilen eşitliklerden de görüleceği gibi sistemin denge noktası *(0, 0, 0, 0)* olarak elde edilmektedir. Daha sonra sistemin Jacobian matrisi elde edilir. Bu matris (4.5) denklemi ile verilmiştir.

$$J = \begin{bmatrix} \frac{-aw^2+b}{C_1} & 0 & -\frac{1}{C_1} & -\frac{2awx}{C_1} \\ 0 & 0 & -\frac{1}{C_2} & 0 \\ -\frac{1}{L} & \frac{1}{L} & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \tag{4.5}$$

Jacobian matrisinde denge noktaları ve parametre değerleri yerine yazılacak olursa elde edilen karakteristik denklem, (4.6) ile verilmiştir.

$$|J - \lambda I| = \lambda^4 + 2.1\lambda^3 + 70.238\lambda^2 + 200\lambda \tag{4.6}$$

Elde edilen karakteristik denklemin kökleri, yani özdeğerler aşağıda verilmiştir.

$$\lambda_1 = 0.34 + j8.49 \tag{4.7}$$

$$\lambda_2 = 0.34 - j8.49 \tag{4.8}$$

$$\lambda_3 = -2.77 \tag{4.9}$$

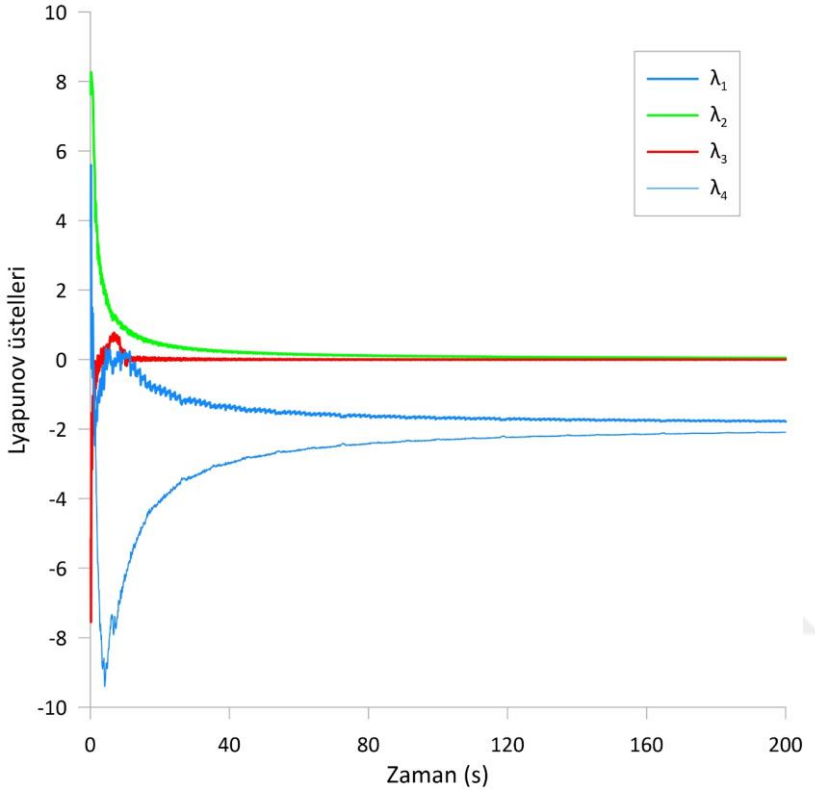
$$\lambda_4 = 0 \tag{4.10}$$

Sistemin özdeğerlerine bakılacak olursa, köklerden en az bir tanesinin gerçek kısmının pozitif değerde olduğu görülmektedir. Bu da sistemin kaotik davranış gösterebileceğine işaret etmedir.

4.3.2.2. Lyapunov Üstelleri

Chua devresi kullanılarak tasarlanan sistemin kaotik davranışa sahip olduğunu göstermek için gerçekleştirilecek analizlerden diğeri Lyapunov üstelleri olacaktır. Önerilen sistemin

MATLAB programı yardımıyla elde edilen, Lyapunov üstellerini gösteren grafik Şekil 4.18'de verilmiştir. Bu grafikten de görülebileceği gibi sistemden elde edilen Lyapunov üstellerden en az bir tanesi pozitif değerde olup, üstellerin tamamının toplamı negatif değerdedir. Bu da sistemin kaotik davranış gösterdiğine işaret etmektedir. Elde edilen Lyapunov üstelleri yaklaşık olarak $LE_1 = 0.048$, $LE_2 = 0$, $LE_3 = -1.781$, $LE_4 = -2.091$ bulunmuştur.



Şekil 4.18: Önerilen sistemin Lyapunov üstelleri.

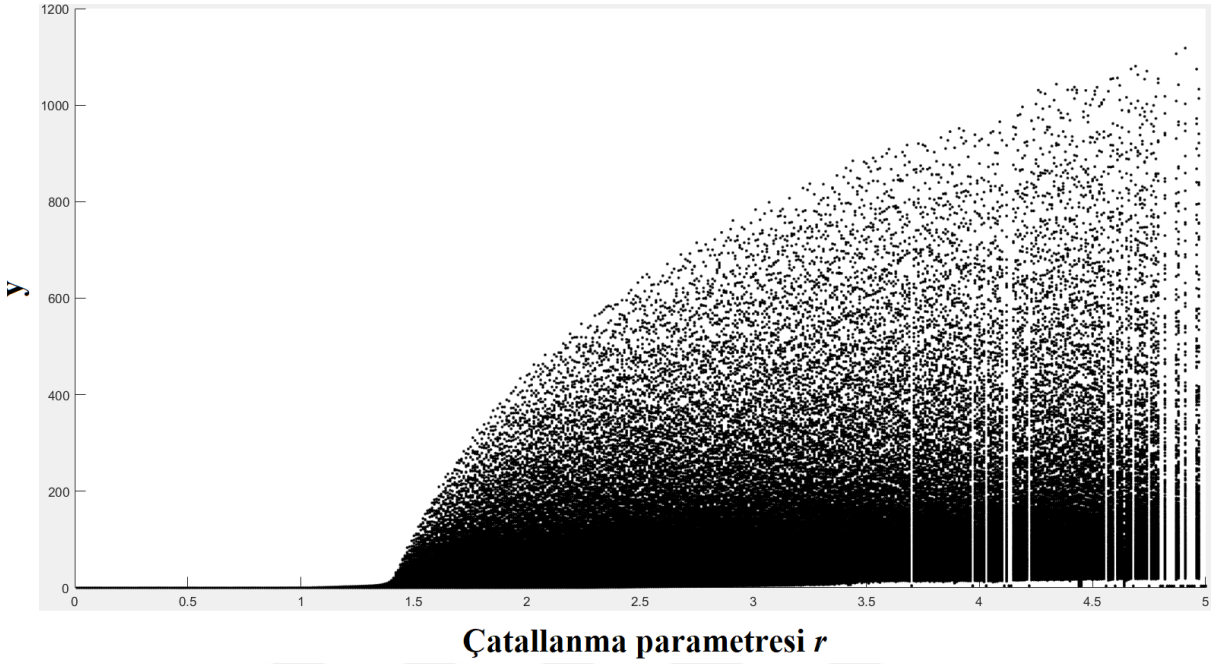
4.3.2.3. Çatallanma Diyagramı

Sunulan kaotik sistemin çatallanma diyagramının MATLAB programı kullanılarak elde edildiği grafik Şekil 4.19'da verilmiştir. Bu grafikte *r* olarak isimlendirilen değişken daha önceki denklemlerde de ifade edilmiş olan $1/C_1$ değerini belirtmektedir. Grafiğin *y* ekseninde ise daha önceki bölümlerde *y* olarak isimlendirilmiş fonksiyon olan v_{C2} ifadesini belirtmektedir. Buradan da görüldüğü gibi, *r* ifadesi *1.4* değerinin üzerine çıktığında sistem kaotik davranış göstermektedir.

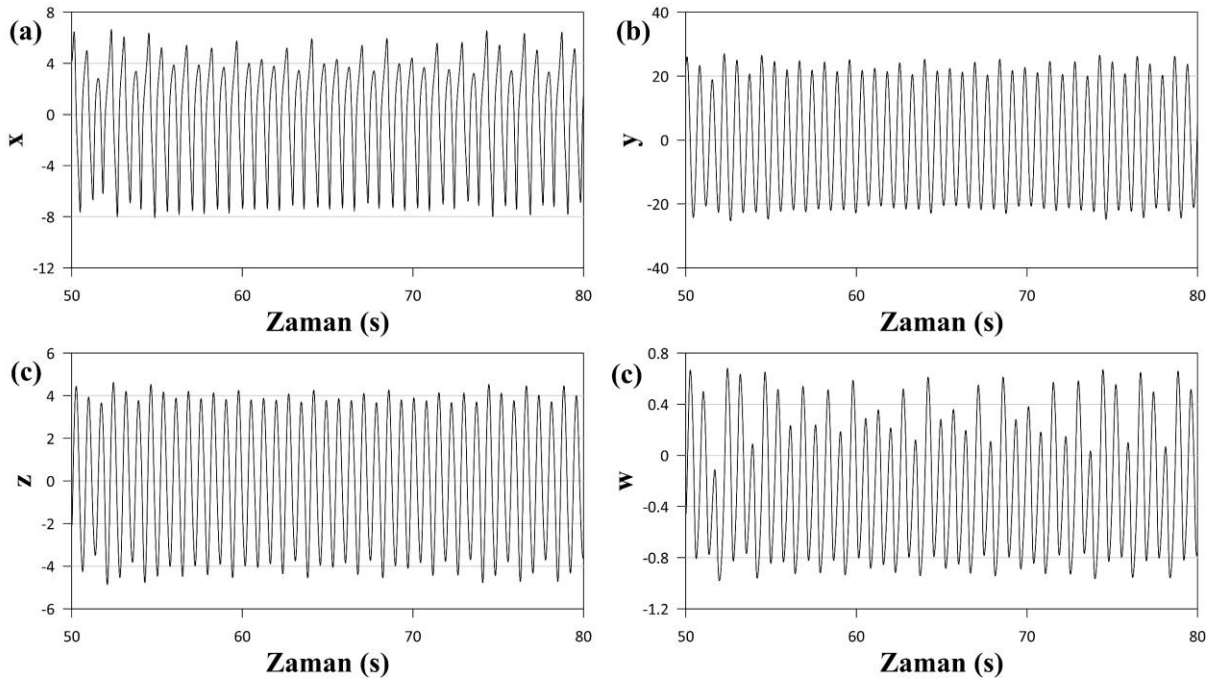
4.3.2.4. Faz Portreleri

Kaotik davranışı gözlemlemenin bir diğeri olan faz portreleri analizi, sunulan kaotik sistem için, yine MATLAB programı aracılığıyla elde edilmiştir. *x, y, z ve w* fonksiyonlarının zamana bağlı

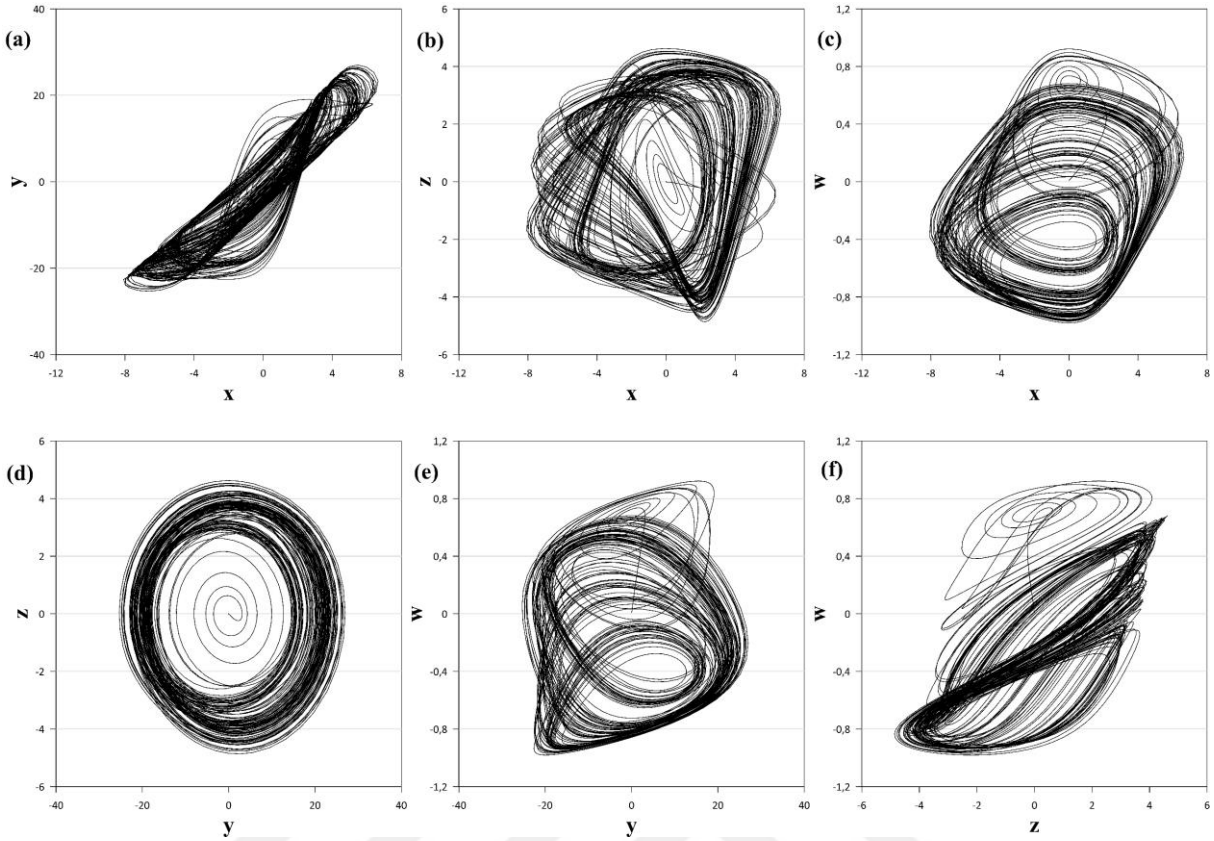
grafikleri Şekil 4.20'de verilmiştir. Elde edilen faz portreleri Şekil 4.21'de verilmiştir. Şekil 4.21'de görülen grafiklere göre sistem kaotik davranış göstermektedir.



Şekil 4.19: *r* parametresinin 0-5 aralığında değiştirilmesiyle oluşan y fonksiyonundaki değişimleri gösteren çatallanma diyagramı.



Şekil 4.20: Chua devresinde zamana bağlı değişim grafikleri: *x* (a), *y* (b), *z* (c) ve *w* (d).



Şekil 4.21: Chua devresi faz porteleri: *x*-*y* (a), *x*-*z* (b), *x*-*w* (c), *y*-*z* (d), *y*-*w* (e) ve *z*-*w* (f).

4.4. KAOTİK HABERLEŞME SİSTEMİ TASARIMI SONUÇLARI

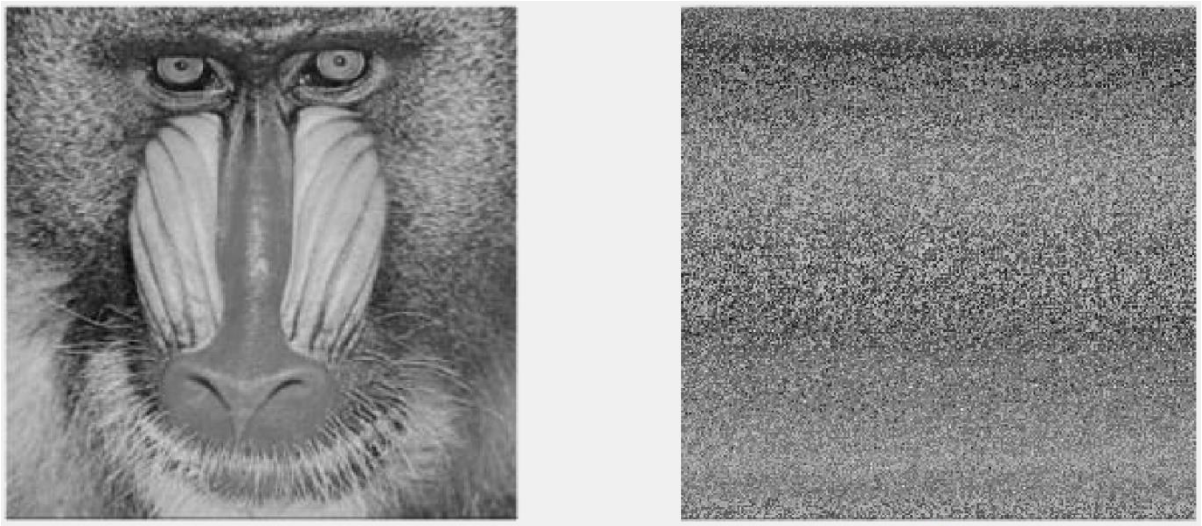
Çalışmanın bu bölümünde tasarlanan RSÜ kullanılarak elde edilen kaos temelli anahtarlar vasıtasıyla yapılan şifreleme ve şifre çözme işlemleri açıklanacaktır. Daha sonra sunulan kaotik sistem ile yapılan kaotik maskeleme işleminin benzetim sonuçları açıklanacaktır.

4.4.1. Kaotik Görüntü Şifreleme Uygulama Sonuçları

Şekil 3.11'de verilen RSÜ tasarımının MATLAB programı yardımıyla gerçekleştirilmesinin ardından, Şekil 3.12 ve Şekil 3.13 ile aktarılan şifreleme ve şifre çözme işlemleri yine MATLAB uygulamasıyla yapılmıştır.

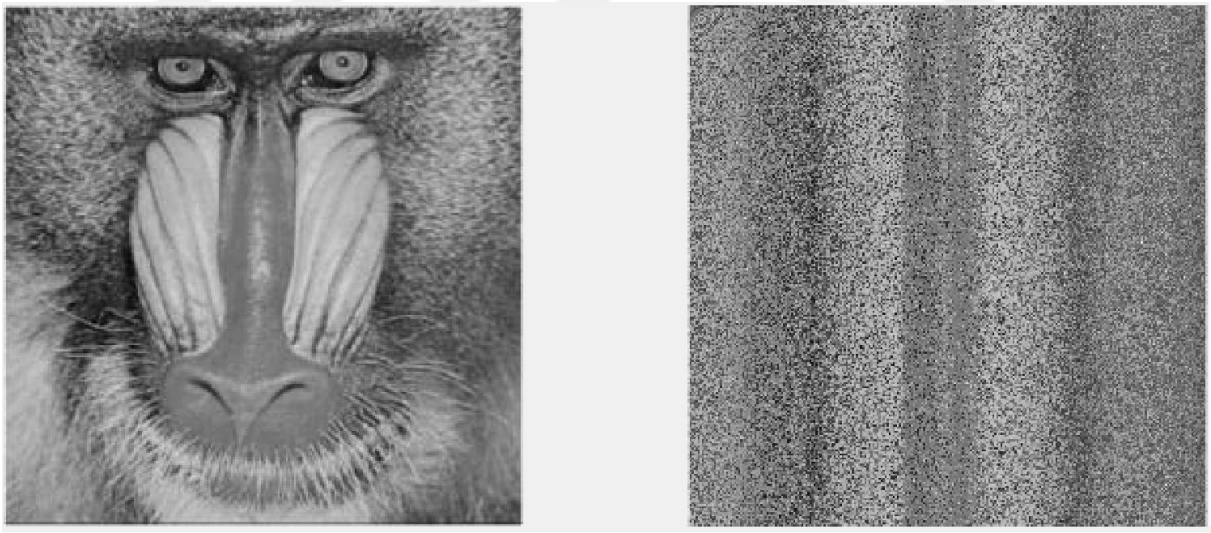
Gerçekleştirilen şifreleme algoritmasının anlaşılması adına blok diyagramla gösterilen basamakların her biri ayrı ayrı ele alınarak, şifrelenen görseller gözlemlenecektir. Şekil 3.12'ye bakıldığında ilk olarak görsele, *K4* anahtarı kullanılarak satır karıştırma işlemi gerçekleştiriliyor. Şifreleme işlemi için bu işlem tek başına gerçekleştirilmiş olsaydı şifrelenmiş metin Şekil 4.22'deki gibi elde edilmiş olacaktı. Satır karıştırma işlemi, görselin *256x256* boyutundaki matrisinin her satırındaki bitlerinin, kullanılan anahtarın ürettiği rastgele sayının

o bit adresindeki sayısal değer ile görselin o sayısal değerdeki satırıyla değiştirilmesi şeklinde gerçekleştirilmiştir. Sütun karıştırma işlemi de aynı yöntemin sütunlara uygulanmasıyla gerçekleştirilmiştir. Sütun karıştırma işleminin *K3* anahtarı kullanılarak tek başına gerçekleştirildiği şifreleme işleminden elde edilen sonuçlar Şekil 4.23'te verilmiştir.



(a) (b)

Şekil 4.22: Satır karıştırma ile şifreleme işlemi. Görüntü (a) ve şifrelenmiş görüntü (b).



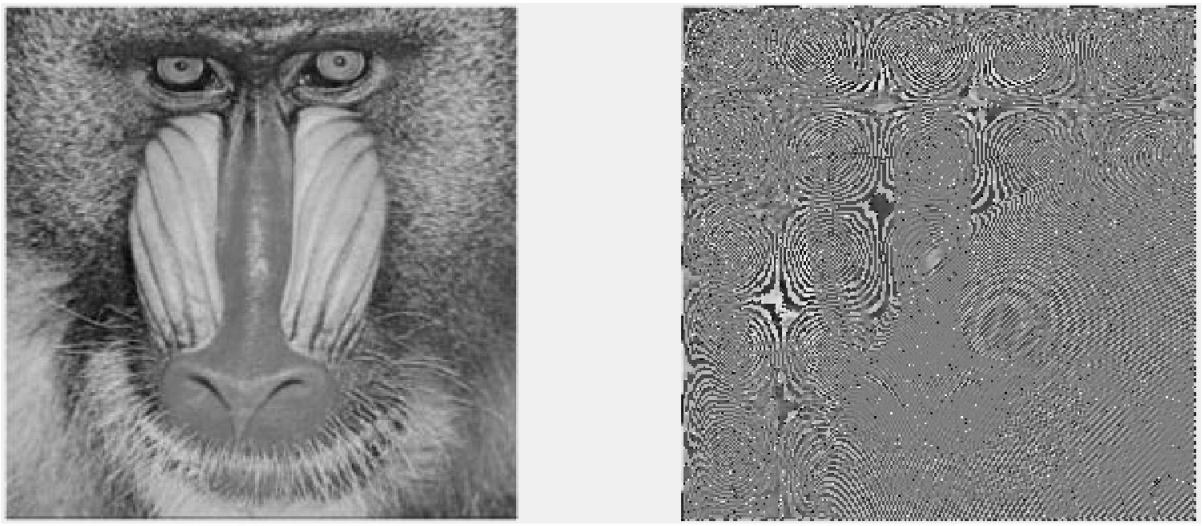
(a) (b)

Şekil 4.23: Sütun karıştırma ile şifreleme işlemi. Görüntü (a) ve şifrelenmiş görüntü (b).

Şekil 4.24a'da verilen görseldeki verilerle *K1* anahtarı arasında gerçekleştirilen *XOR* işleminin tek başına gerçekleştirilmesiyle elde edilen şifrelenmiş görüntü Şekil 4.24b'de verilmiştir. Aynı şekilde Şekil 4.25a'da verilen görsel kullanılarak *K2* anahtarıyla görsel arasındaki *XOR*

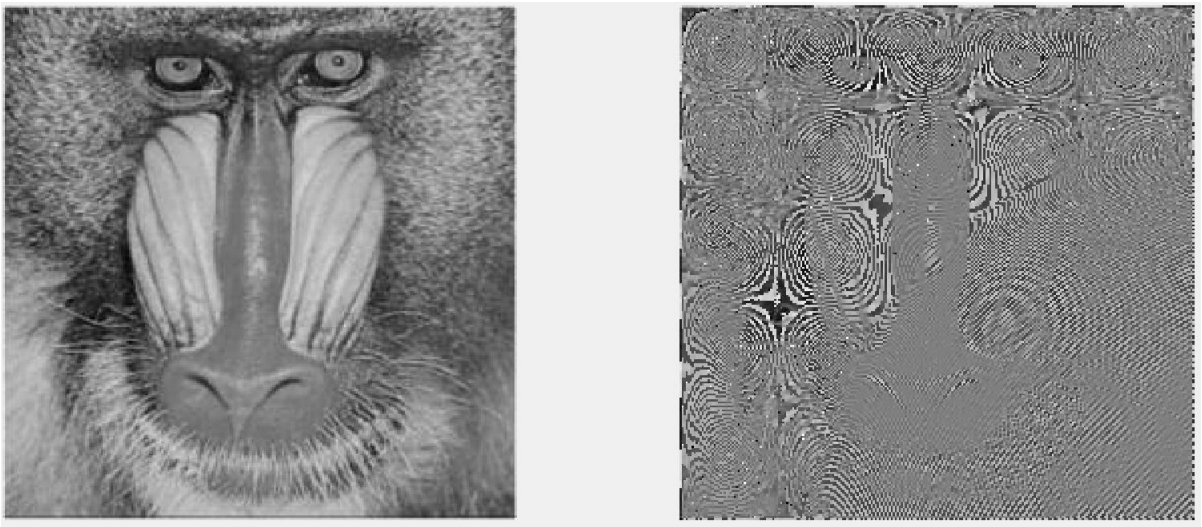
işleminin tek başına gerçekleştirilmesi neticesinde elde edilen şifrelenmiş görüntü Şekil 4.25b'de verilmiştir.

Önerilen şifreleme ve şifre çözme algoritması kullanılarak gerçekleştirilen şifreleme çalışmasının sonuçları Şekil 4.26 ile aktarılmıştır. Şekil 4.26a'da şifrelenecek görsel, Şekil 4.26b'de şifrelenmiş görsel, Şekil 4.26c'de ise şifresi çözülmüş görsel verilmiştir.



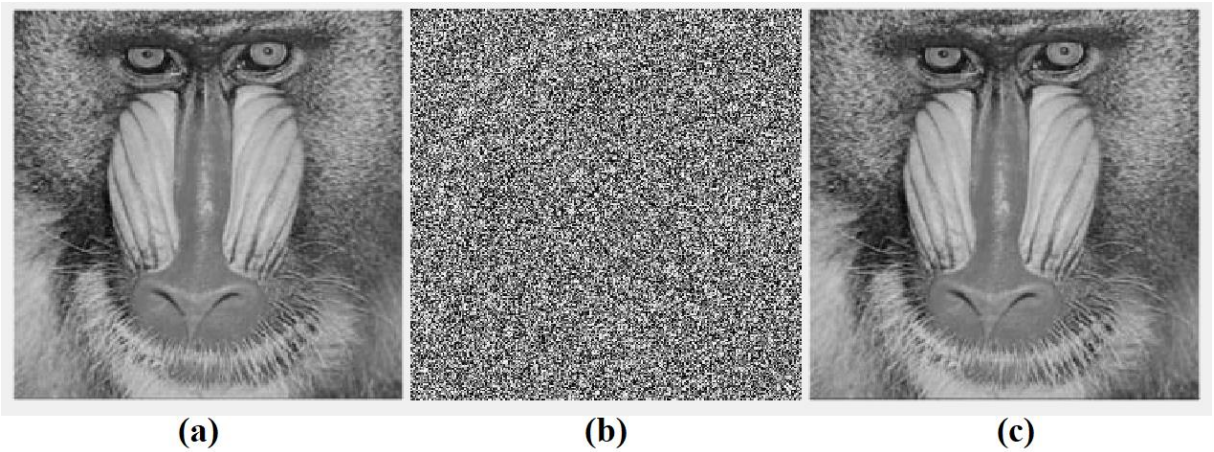
(a) (b)

Şekil 4.24: *K1* anahtarı kullanılarak *XOR* işlemi ile gerçekleştirilen şifreleme işlemi. Görüntü (a) ve şifrelenmiş görüntü (b).



(a) (b)

Şekil 4.25: *K2* anahtarı kullanılarak *XOR* işlemi ile gerçekleştirilen şifreleme işlemi. Görüntü (a) ve şifrelenmiş görüntü (b).

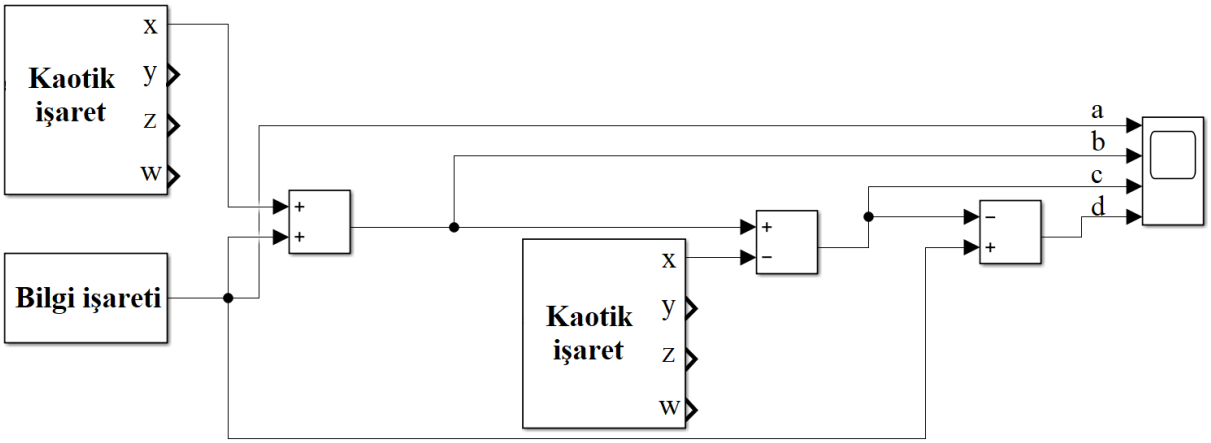


Şekil 4.26: Önerilen şifreleme algoritması ile gerçekleştirilen şifreleme ve şifre çözme işlemi. Görüntü (a), şifrelenmiş görüntü (b) ve şifresi çözülmüş görüntü (c).

Sonuç olarak her bir şifreleme işlemi ayrı ayrı yapıldığında da görüntü şifreleme işleminin gerçekleştirilebildiği, ancak algoritmanın iyileştirilmesiyle daha sağlam bir şifreleme işleminin gerçekleştirilebildiği görülmektedir.

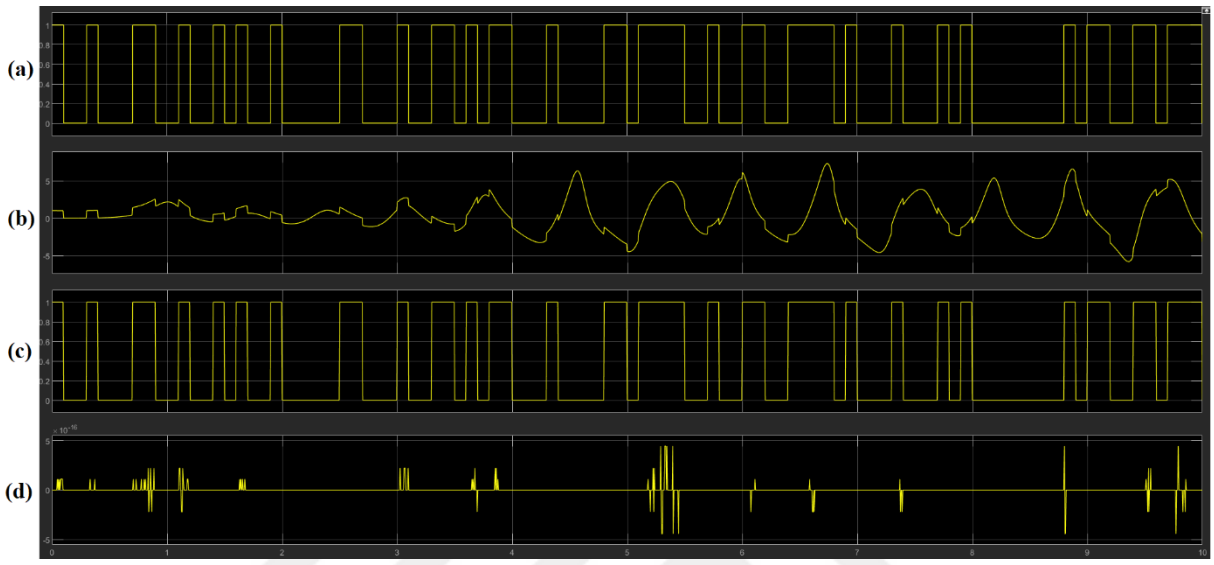
4.4.2. Kaotik Maskeleme Uygulama Sonuçları

Şekil 3.14 ile verilen kaotik maskeleme işleminin, sunulan kaotik sistem kullanılarak uygulama sonuçlarının elde edilmesi için MATLAB Simulink programından yararlanılmıştır. Görüntü şifreleme konusunda kullanılan kaotik sistem, Chau devresiyle elde edilen sistem ile oluşturulmuştu. Kaotik maskeleme işlemi için kullanılacak sistem ise, tez kapsamında sunulmuş olan kaotik Jerk devresi olacaktır. Maskeleme işlemi için MATLAB Simulink programıyla oluşturulan sistem Şekil 4.27'de verilmiştir. Burada gözlemlenen işaretler sırasıyla bilgi işareti (a), modüle edilmiş işaret (b), demodüle edilmiş işaret (c) ve hata işaretidir (d).



Şekil 4.27: MATLAB Simulink programında kaotik maskeleme işlemi.

Şekil 4.28'de kaotik maskeleme işleminin neticesinde alınan sonuçlar verilmiştir. Şekil 4.28a'da verilen sayısal bilgi işaretinin modüle edilmiş hali Şekil 4.28b'de ve demodüle edilmiş hali Şekil 4.28c'de verilmiştir. Şekil 4.28d ise maskeleme işleminin sonucunda yaşanan hata miktarını göstermektedir.



Şekil 4.28: MATLAB Simulink programında kaotik maskeleme işleminin sonuçları: bilgi işareti (a), modüle edilmiş işaret (b), demodüle edilmiş işaret (c) ve hata işareti (d).

5. TARTIŞMA

Temel olarak tez kapsamında gerçekleştirilen çalışma üç ana konu üzerinde incelenebilmektedir. Bu konular:

1) Hafızalı devre elemanı tasarımı,

2) Hafızalı devre elemanı temelli kaotik sistem tasarımı,

3) Önerilen sistem kullanılarak kaotik haberleşme sistemi tasarımı, olarak ele alınmıştır.

Önerilen hafızalı devre elemanı CC-CCTA isimli aktif eleman kullanılarak tasarlanmıştır. Tasarımın öne çıkan özellikleri, yüzen ve topraklanmış formda gerçekleştirilebilmesi, güç tüketiminin az oluşu, minimum düzeyde pasif eleman içermesi ve elektronik olarak kontrol edilebilir özellikte olmasıdır. Çalışmada CC-CCTA aktif elemanın tercih edilmesinin en önemli sebebi, CC-CCTA'nın akım taşıyıcı kısmında üretilen parazitik direncin tasarlanacak devre içerisinde kullanılabilir olmasıdır [90]. Bu direnç sayesinde iç yapıda gerçekleştirilen ufak bir değişimle, kontrollü bir direnç olarak elde edilmiş olmaktadır. Temel olarak bu parazitik direnç sayesinde doğrusal olmayan özellik elde edilmiş olmaktadır. İç yapının çıkışındaki geçiş iletkenliği kuvvetlendirici katından elde edilebilecek olan g_m değeri ise ikinci bir dirençten tasarruf edilmesini sağlamaktadır. Devrenin yüzen forma getirilmesi için bir adet fazladan OTA'ya ihtiyaç duyulmaktadır. Bu ihtiyaç bir dezavantaj gibi gözükse de devrenin yüzen forma getirilebiliyor oluşu önemli bir özelliktir.

Çalışmanın ikinci aşaması olan kaotik devre tasarımı için iki farklı yapı gerçekleştirilmiştir. Bunların ilki Jerk sistemidir. Tasarlanan hafızalı devre elemanıyla Jerk sistemi oluşturulmuştur. 2020 yılında yapılan bir çalışmada, gerçek memristör kullanılarak Jerk sistemi gerçekleştirilmiştir [33]. Tez kapsamında [33] ile belirtilen çalışmada kullanılan Jerk sistemi ele alınmış olup, gerçek memristör yapısı yerine önerilen memristör emülatörü kullanılmıştır. LTspice kullanılarak yapılan benzetim sonuçları neticesinde elde edilen sonuçlar, matematiksel açıklamalarla tutarlı olup, oldukça tatmin edici düzeydedir. Önerilen memristör emülatörü kullanılarak gerçekleştirilen bir diğer kaotik sistem ise Chua devresi kullanılarak elde edilmiştir. Chua devresinde daha yüksek kaotik özelliklerin elde edilebilmesi için kübik forma

getirilmiştir [37, 82]. Bunun için devreye bir adet analog çarpıcı eklenmiştir. Kübik form sayesinde memristörün gösterdiği doğrusal olmama özelliğinin yükselmesi ve bu sayede üretilecek kaotik sistemin daha iyi düzeyde kaotik davranış göstermesi sağlanmaktadır. Jerk devresine göre Chua devresinin daha basit yapıda olmasına karşın, daha düşük doğrusal olmama özelliğine sahiptir. Bu sebeple Chua devresinin gerçekleştirilmesinde kübik formdaki memristör emülatörünün kullanılması uygun görülmüştür. Neticede benzer iki memristör emülatörü kullanılarak iki farklı kaotik sistem tasarımı gerçekleştirilmiştir.

Tez çalışmasının son aşamasında, oluşturulan kaotik sistemler kullanılarak kaotik haberleşme sistemi tasarlanmıştır. Kaotik haberleşme sistemi için ilk olarak bilgi şifreleme algoritması geliştirilmiştir. Bilgi şifrelemek için kullanılmak üzere kaos tabanlı rastgele sayı üreteci tasarlanmıştır. Bu işlem için önerilen kübik memristör emülatörü tabanlı Chua devresi kullanılmıştır. Tasarlanan bu rastgele sayı üreteci kullanılarak kaos bazlı anahtarlar oluşturulmuştur. Daha sonra bu anahtarlar kullanılarak gerçekleştirilen bir görüntü şifreleme algoritması gerçekleştirilmiştir. Bu algoritma XOR işlemi ile satır ve sütun karıştırma işlemlerini içermektedir. Literatürde, daha önceki çalışmalarda benzer algoritmalar bulunmaktadır. Örnek olarak [9] referansıyla belirtilen çalışma da benzer şekilde güçlü bir görüntü şifreleme algoritması içermektedir. Ancak [9] çalışmasında kullanılan kaotik yapı 3 boyutludur ve bu sebeple şifreleme adımlarının bazı aşamalarında aynı anahtarlar kullanılarak işlemler gerçekleştirilmiştir. Tez kapsamında kullanılan kaotik yapı 4 boyutludur ve görüntü şifrelemenin her basamağı için farklı anahtarlar kullanılmaktadır. [117] numaralı referansla belirtilen çalışmada yine Chua devresi kullanılarak elde edilmiş bir kaotik sistemden faydalanılarak bir görüntü şifreleme algoritması geliştirilmiştir. Bu çalışmanın öne çıkan özelliği tasarlanan algoritmanın, DNA kodlama, lojistik haritalama gibi karmaşık aşamalar içermesidir. Bu aşamalar işlemi karmaşıklaştırsa da gerçekleştirilen şifreleme işleminin daha sağlam olmasına sebep olmaktadır. Tez kapsamında [117] ile belirtilen çalışmadaki gibi karmaşık bir algoritmanın tercih edilmeme sebebi, tez çalışması kapsamında tasarlanan kaotik sistemlerin ön planda bulunmasıdır.

Chua temelli kaotik sistem kullanılarak görüntü şifreleme işlemi gerçekleştirildikten sonra tez kapsamında sunulan bir diğer kaotik yapı olan Jerk sistemi kullanılarak da kaotik maskeleme işlemi gerçekleştirilmiştir.

6. SONUÇ VE ÖNERİLER

Haberleşme sistemlerinin modülasyon ve şifreleme gibi aşamalarında kaotik sistemlerden faydalanmak sistemin güvenilirliğine olumlu katkılar sağlamaktadır. Bu çalışma kapsamında haberleşme sisteminde kullanılmasına yönelik iki farklı kaotik sistem sunulmuştur. Bu kaotik sistemlerin biri Chua devresiyle elde edilerek görüntü şifreleme uygulamasıyla gözlemlenmiştir. Bunun için öncelikle analog olan kaotik işaret, yazılım ortamında ayrık işarete dönüştürülmüştür. Ayrık işarete dönüştürülen bu yeni veri kullanılarak 4 farklı anahtarı oluşturan rastgele sayı üreteci tasarımı elde edilmiştir. Daha sonra bu anahtarlar kullanılarak yeni bir kaotik şifreleme algoritması geliştirilmiştir. Geliştirilen algoritmanın sade yapıda olmasına karşın sağlam bir şifreleme yeteneğine sahip olduğu, görüntü şifreleme uygulamasıyla gösterilmiştir. Daha sonra modülasyon ve demodülasyon işlemleri için kaotik maskeleme uygulaması Jerk sistemlerinden faydalanılarak gerçekleştirilmiştir. Yani, Chua devresi ve Jerk devresi tez kapsamında kaotik işaret kaynakları olarak değerlendirilmiştir. Her iki devrede de kaotik sistemi oluşturmak için hafızalı devre elemanı olan memristörlerden faydalanılmıştır. Bu sebeple kaos oluşturulması için ihtiyaç duyulan memristör emülatörü tasarımı gerçekleştirilmiştir. Memristör emülatörü hem topraklanmış hem de yüzen yapıda elde edilebilecek şekilde tasarlanmıştır. Memristör emülatörünü oluşturan aktif eleman CC-CCTA'dır. Pasif eleman olarak yalnızca bir adet kapasitöre ihtiyaç duyulmuştur. Yüzen memristör emülatörünün elde edilmesinde fazladan bir adet OTA'ya ihtiyaç duyulmaktadır. Tasarlanan memristör emülatörünün teorik ve benzetim analizleri detaylı olarak açıklanmıştır. Teorik olarak elde edilen matematiksel ifadelere benzetim sonucunda da ulaşıldığı gözelemlenmiştir. Ayrıca, önerilen memristör emülatörüne, bir adet çarpıcı devresinin ilave edilmesiyle devrenin doğrusal olmayan özelliği geliştirilebilmektedir. Devrenin bu hali kübik form olarak isimlendirilmiştir. Kübik formdaki memristör emülatörü Chua devresinde kullanılmıştır. Jerk devresinde yüzen memristör emülatörü kullanılmıştır. Her iki sistemin de analizleri detaylı olarak gerçekleştirilip kaotik özellikleri hem matematiksel olarak hem de benzetim programları yardımıyla gösterilmiştir.

Sunulan memristör emülatörünün, Yüzen ve topraklanmış yapıda oluşu, direnç içermemesi, güç tüketiminin az oluşu, elektronik olarak kontrol edilebilir oluşu ve yüksek frekanslarda

çalışabilme kapasitesi bulunduğundan, daha farklı uygulamalarda da kullanılması mümkündür. Kübik forma getirilmiş yapısıyla her türlü kaotik devrede yüksek derecede doğrusal olmama özelliği gösterebilir niteliktedir.

Tasarlanan kaotik sistemler haberleşmede daha farklı şekilde de değerlendirilebilmektedir. Bu sistemler kullanılarak daha farklı kaotik modülasyon teknikleri geliştirilebilir durumdadır.

KAYNAKLAR

[1]. Chua, L. (1971). Memristor-the missing circuit element. IEEE Transactions on circuit theory, 18(5), 507-519.

[2]. Strukov, D. B., Snider, G. S., Stewart, D. R., & Williams, R. S. (2008). The missing memristor found. nature, 453(7191), 80-83.

[3]. Memristor and Memristive Systems Symp., University of California, Berkeley, November 21, 2008.

[4]. Martinez-Rincon, J., Di Ventra, M., & Pershin, Y. V. (2010). Solid-state memcapacitive system with negative and diverging capacitance. Physical Review B, 81(19), 195430.

[5]. Biolek, D., Biolek, Z., & Biolkova, V. (2010). SPICE modelling of memcapacitor. Electronics letters, 46(7), 520.

[6]. Hu, Z., Li, Y., Jia, L., & Yu, J. (2010, July). Chaotic oscillator based on current-controlled meminductor. In 2010 International Conference on Communications, Circuits and Systems (ICCCAS) (pp. 820-823). IEEE.

[7]. Ersoy, D., & Kacar, F. (2023). Electronically Charge-Controlled Tunable Meminductor Emulator Circuit with OTAs and its Applications. IEEE Access.

[8]. Xu, B., Wang, G., Iu, H. H. C., Yu, S., & Yuan, F. (2019). A memristor–meminductor based chaotic system with abundant dynamical behaviors. Nonlinear Dynamics, 96, 765-788.

[9]. Yildirim, M., & Kacar, F. (2020). Chaotic circuit with OTA based memristor on image cryptology. AEU-International Journal of Electronics and Communications, 127, 153490.

[10]. Nguyen, N., Pham-Nguyen, L., Nguyen, M. B., & Kaddoum, G. (2020). A low power circuit design for chaos-key based data encryption. IEEE Access, 8, 104432-104444.

[11]. Sahin, M. E., Cam Taskiran, Z. G., Guler, H., & Hamamci, S. E. (2020). Application and modeling of a novel 4D memristive chaotic system for communication systems. Circuits, Systems, and Signal Processing, 39, 3320-3349.

[12]. James, A. P., & Chua, L. O. (2021). Analog neural computing with super-resolution memristor crossbars. IEEE Transactions on Circuits and Systems I: Regular Papers, 68(11), 4470-4481.

[13]. Li, C., Yang, Y., Yang, X., Zi, X., & Xiao, F. (2022). A tristable locally active memristor and its application in Hopfield neural network. Nonlinear Dynamics, 108(2), 1697-1717.

[14]. Biolek, Z., Biolek, D., & Biolkova, V. (2009). SPICE Model of Memristor with Nonlinear Dopant Drift. Radioengineering, 18(2).

[15]. Rak, A., & Cserey, G. (2010). Macromodeling of the memristor in SPICE. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 29(4), 632-636.

[16]. Mutlu, R., & Karakulak, E. (2010, December). Emulator circuit of Ti0 2 memristor with linear dopant drift made using analog multiplier. In National Conference on Electrical, Electronics and Computer Engineering (pp. 380-384). IEEE.

[17]. Kim, H., Sah, M. P., Yang, C., Cho, S., & Chua, L. O. (2012). Memristor emulator for memristor circuit applications. IEEE Transactions on Circuits and Systems I: Regular Papers, 59(10), 2422-2431.

[18]. Prasad, S. S., Kumar, P., & Ranjan, R. K. (2021). Resistorless memristor emulator using CFTA and its experimental verification. IEEE Access, 9, 64065-64075.

[19]. Sharma, V. K., Parveen, T., & Ansari, M. S. (2021). Four quadrant analog multiplier based memristor emulator using single active element. AEU-International Journal of Electronics and Communications, 130, 153575.

[20]. Raj, A., Singh, S., & Kumar, P. (2022). Dual mode, high frequency and power efficient grounded memristor based on OTA and DVCC. Analog Integrated Circuits and Signal Processing, 1-9.

[21]. Vista, J., & Ranjan, A. (2019). A simple floating MOS-memristor for high-frequency applications. IEEE Transactions on Very Large Scale Integration (VLSI) Systems, 27(5), 1186-1195.

[22]. Gupta, R. K., Choudhry, M. S., Saxena, V., & Taran, S. (2023). A Single MOS-Memristor Emulator Circuit. Circuits, Systems, and Signal Processing, 1-20.

[23]. Ranjan, R. K., Rani, N., Pal, R., Paul, S. K., & Kanyal, G. (2017). Single CCTA based high frequency floating and grounded type of incremental/decremental memristor emulator and its application. Microelectronics Journal, 60, 119-128.

[24]. Kanyal, G., Kumar, P., Paul, S. K., & Kumar, A. (2018). OTA based high frequency tunable resistorless grounded and floating memristor emulators. AEU-International Journal of Electronics and Communications, 92, 124-145.

[25]. Ghosh, M., Singh, A., Borah, S. S., Vista, J., Ranjan, A., & Kumar, S. (2022). MOSFET-based memristor for high-frequency signal processing. IEEE Transactions on Electron Devices, 69(5), 2248-2255.

[26]. Yu, F., Shen, H., Zhang, Z., Huang, Y., Cai, S., & Du, S. (2021). A new multi-scroll Chua's circuit with composite hyperbolic tangent-cubic nonlinearity: Complex dynamics, Hardware implementation and Image encryption application. Integration, 81, 71-83.

[27]. Kumari, U., & Yadav, R. (2023). Implementation of tunable OTA-based memristor emulator circuit with chaotic behavior. Multiscale and Multidisciplinary Modeling, Experiments and Design, 1-16.

[28]. Raj, N., Ranjan, R. K., & Khateb, F. (2020). Flux-controlled memristor emulator and its experimental results. IEEE Transactions on Very Large Scale Integration (VLSI) Systems, 28(4), 1050-1061.

[29]. Bao, B., Xu, L., Wang, N., Bao, H., Xu, Q., & Chen, M. (2018). Third-order RLCM-four-elements-based chaotic circuit and its coexisting bubbles. AEU-International Journal of Electronics and Communications, 94, 26-35.

[30]. Kengne, J., Njitacke Tabekoueng, Z., Kamdoum Tamba, V., & Nguomkam Negou, A. (2015). Periodicity, chaos, and multiple attractors in a memristor-based Shinriki's circuit. Chaos: An Interdisciplinary Journal of Nonlinear Science, 25(10).

[31]. Sprott, J. C. (2011). A new chaotic jerk circuit. IEEE Transactions on Circuits and Systems II: Express Briefs, 58(4), 240-243.

[32]. Kengne, J., Negou, A. N., & Tchiotsop, D. (2017). Antimonotonicity, chaos and multiple attractors in a novel autonomous memristor-based jerk circuit. Nonlinear Dynamics, 88, 2589-2608.

[33]. Minati, L., Gambuzza, L. V., Thio, W. J., Sprott, J. C., & Frasca, M. (2020). A chaotic circuit based on a physical memristor. Chaos, Solitons & Fractals, 138, 109990.

[34]. Itoh, M., & Chua, L. O. (2008). Memristor oscillators. International journal of bifurcation and chaos, 18(11), 3183-3206.

[35]. Sprott, J. C. (1997). Some simple chaotic jerk functions. American Journal of Physics, 65(6), 537-543.

[36]. Shinriki, M., Yamamoto, M., & Mori, S. (1981). Multimode oscillations in a modified van der Pol oscillator containing a positive nonlinear conductance. Proceedings of the IEEE, 69(3), 394-395.

[37]. Muthuswamy, B. (2010). Implementing memristor based chaotic circuits. International Journal of Bifurcation and Chaos, 20(05), 1335-1350.

[38]. Zhao, Q., Wang, C., & Zhang, X. (2019). A universal emulator for memristor, memcapacitor, and meminductor and its chaotic circuit. Chaos: An Interdisciplinary Journal of Nonlinear Science, 29(1).

[39]. Ranjan, R. K., Sagar, S., Roushan, S., Kumari, B., Rani, N., & Khateb, F. (2019). High-frequency floating memristor emulator and its experimental results. IET Circuits, Devices & Systems, 13(3), 292-302.

[40]. Raj, N., Ranjan, R. K., & Khateb, F. (2020). Flux-controlled memristor emulator and its experimental results. IEEE Transactions on Very Large Scale Integration (VLSI) Systems, 28(4), 1050-1061.

[41]. Guo, Q., Wang, N., & Zhang, G. (2021). A novel current-controlled memristor-based chaotic circuit. Integration, 80, 20-28.

[42]. Bao, B., Wang, N., Xu, Q., Wu, H., & Hu, Y. (2016). A simple third-order memristive band pass filter chaotic circuit. IEEE Transactions on Circuits and Systems II: Express Briefs, 64(8), 977-981.

[43]. Bao, B., Xu, L., Wang, N., Bao, H., Xu, Q., & Chen, M. (2018). Third-order RLCM-four-elements-based chaotic circuit and its coexisting bubbles. AEU-International Journal of Electronics and Communications, 94, 26-35.

[44]. Korkmaz, M. O., Babacan, Y., & Yesil, A. (2023). A new CCII based meminductor emulator circuit and its experimental results. AEU-International Journal of Electronics and Communications, 158, 154450.

[45]. Chua, L. O., & Kang, S. M. (1976). Memristive devices and systems. Proceedings of the IEEE, 64(2), 209-223.

[46]. Wang, F. Y. (2008). Memristor for introductory physics. arXiv preprint arXiv:0808.0286. (http://arxiv.org/abs/0808.0286).

[47]. Yu, D. S., Liang, Y., Chen, H., & Iu, H. H. (2013). Design of a practical memcapacitor emulator without grounded restriction. IEEE Transactions on Circuits and Systems II: Express Briefs, 60(4), 207-211.

[48]. Elwakil, A. S., Fouda, M. E., & Radwan, A. G. (2013). A simple model of double-loop hysteresis behavior in memristive elements. IEEE Transactions on Circuits and Systems II: Express Briefs, 60(8), 487-491.

[49]. Alharbi, A. G., Khalifa, Z. J., Fouda, M. E., & Chowdhury, M. H. (2015, December). A new simple emulator circuit for current controlled memristor. In 2015 IEEE International Conference on Electronics, Circuits, and Systems (ICECS) (pp. 288-291). IEEE.

[50]. Hussein, A. I., & Fouda, M. E. (2013, December). A simple MOS realization of current controlled memristor emulator. In 2013 25th International Conference on Microelectronics (ICM) (pp. 1-4). IEEE.

[51]. Babacan, Y., Yesil, A., & Kacar, F. (2017). Memristor emulator with tunable characteristic and its experimental results. AEU-International Journal of Electronics and Communications, 81, 99-104.

[52]. Ranjan, R. K., Raj, N., Bhuwal, N., & Khateb, F. (2017). Single DVCCTA based high frequency incremental/decremental memristor emulator and its application. AEU-International Journal of Electronics and Communications, 82, 177-190.

[53]. Sánchez-López, C., Carbajal-Gómez, V. H., Carrasco-Aguilar, M. A., & Carro-Pérez, I. (2018). Fractional-order memristor emulator circuits. Complexity, 2018.

[54]. Bhardwaj, K., & Srivastava, M. (2022). New multiplier-less compact tunable charge-controlled memelement emulator using grounded passive elements. Circuits, Systems, and Signal Processing, 1-37.

[55]. Sözen, H., & Çam, U. (2015, November). New memristor emulator circuit using OTAs and CCIIs. In 2015 9th International Conference on Electrical and Electronics Engineering (ELECO) (pp. 10-14). IEEE.

[56]. Sözen, H., & Çam, U. (2016). Electronically tunable memristor emulator circuit. Analog Integrated Circuits and Signal Processing, 89, 655-663.

[57]. Alharbi, A. G., Fouda, M. E., & Chowdhury, M. H. (2017). A novel flux-controlled memristive emulator for analog applications. Advances in Memristors, Memristive devices and systems, 493-511.

[58]. Yesil, A. (2018). A new grounded memristor emulator based on MOSFET-C. AEU-International Journal of Electronics and Communications, 91, 143-149.

[59]. Yeşil, A., Babacan, Y., & Kaçar, F. (2018). Design and experimental evolution of memristor with only one VDTA and one capacitor. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 38(6), 1123-1132.

[60]. Babacan, Y., Yesil, A., & Gul, F. (2018). The fabrication and MOSFET-only circuit implementation of semiconductor memristor. IEEE Transactions on Electron Devices, 65(4), 1625-1632.

[61]. Yesil, A., Babacan, Y., & Kacar, F. (2019). Electronically tunable memristor based on VDCC. AEU-International Journal of Electronics and Communications, 107, 282-290.

[62]. Petrović, P. B. (2019). Tunable flux-controlled floating memristor emulator circuits. IET Circuits, Devices & Systems, 13(4), 479-486.

[63]. Vista, J., & Ranjan, A. (2020). Flux controlled floating memristor employing VDTA: Incremental or decremental operation. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 40(2), 364-372.

[64]. Yildiz, H. A., & Ozoguz, S. (2021). MOS-only implementation of memristor emulator circuit. AEU-International Journal of Electronics and Communications, 141, 153975.

[65]. Bhardwaj, K., & Srivastava, M. (2021). Wide-band compact floating memristor emulator configuration with electronic/resistive adjustability. Microelectronics Journal, 117, 105284.

[66]. Raj, N., Verma, V. K., & Ranjan, R. K. (2022). Electronically Tunable Flux-Controlled Resistorless Memristor Emulator. IEEE Canadian Journal of Electrical and Computer Engineering, 45(3), 311-317.

[67]. Kumar, A., Chaturvedi, B., & Mohan, J. (2023). Minimal realizations of integrable memristor emulators. Journal of Computational Electronics, 22(1), 504-518.

[68]. Thietart, R. A., & Forgues, B. (1995). Chaos theory and organization. Organization science, 6(1), 19-31.

[69]. Lorenz, E. N. (1963). Deterministic nonperiodic flow. Journal of atmospheric sciences, 20(2), 130-141.

[70]. Gao, T., Chen, G., Chen, Z., & Cang, S. (2007). The generation and circuit implementation of a new hyper-chaos based upon Lorenz system. Physics Letters A, 361(1-2), 78-86.

[71]. Han, C., Yu, S., & Wang, G. (2015). A sinusoidally driven Lorenz system and circuit implementation. Mathematical Problems in Engineering, 2015.

[72]. Aydın, Ş. Z., Beken, G. N., & Taşkıran, Z. G. Ç. (2023). A Lorenz-like Chaotic OTA-C Circuit and Memristive Synchronization. Chaos Theory and Applications, 5(1), 52-58.

[73]. Rössler, O. E. (1976). An equation for continuous chaos. Physics Letters A, 57(5), 397-398.

[74]. García-Lopez, J. H., Jaimes-Reátegui, R., Pisarchik, A. N., Murguía-Hernandez, A., Medina-Gutiérrez, C., Valdivia-Hernadez, R., & Villafana-Rauda, E. (2005). Novel communication scheme based on chaotic Rössler circuits. In Journal of Physics: Conference Series (Vol. 23, No. 1, p. 276). IOP Publishing.

[75]. Jamal, R. K., & Kafi, D. A. (2019, July). Secure communication coupled semiconductor laser based on Rössler chaotic circuits. In IOP Conference Series: Materials Science and Engineering (Vol. 571, No. 1, p. 012119). IOP Publishing.

[76]. Matsumoto, T. (1984). A chaotic attractor from Chua's circuit. IEEE transactions on circuits and systems, 31(12), 1055-1058.

[77]. Itoh, M., & Chua, L. O. (2013). Duality of memristor circuits. International Journal of Bifurcation and Chaos, 23(01), 1330001.

[78]. Ma, H., Yu, Y., Yang, C., Yang, N., Wang, Y., Wang, X., & Nyima, T. (2018, December). Dynamical Behaviors of a Modified Chua's Circuit. In 2018 11th International Symposium on Computational Intelligence and Design (ISCID) (Vol. 1, pp. 29-32). IEEE.

[79]. Raj, N., Ranjan, R. K., & James, A. (2022). Chua's Oscillator With OTA Based Memcapacitor Emulator. IEEE Transactions on Nanotechnology, 21, 213-218.

[80]. Muthuswamy, B., & Chua, L. O. (2010). Simplest chaotic circuit. International Journal of Bifurcation and Chaos, 20(05), 1567-1580.

[81]. Sahin, M. E., Taskiran, Z. C., Guler, H., & Hamamci, S. E. (2019). Simulation and implementation of memristive chaotic system and its application for communication systems. Sensors and Actuators A: Physical, 290, 107-118.

[82]. Sahin, M. E. (2020). Memristör Tabanlı Kaotik Sistemlerin Tasarımı Ve Kaotik Haberleşme Sistemlerine Uygulanması. Thesis (PhD), Fırat University.

[83]. Zhong, G. Q. (1994). Implementation of Chua's circuit with a cubic nonlinearity. IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications, 41(12), 934-941.

[84]. Ramadoss, J., Kengne, J., Telem, A. N. K., & Rajagopal, K. (2022). Broken symmetry and dynamics of a memristive diodes bridge-based Shinriki oscillator. Physica A: Statistical Mechanics and its Applications, 588, 126562.

[85]. Volos, C. K., Pham, V. T., Nistazakis, H. E., & Stouboulos, I. N. (2020). A dream that has come true: Chaos from a nonlinear circuit with a real memristor. International Journal of Bifurcation and Chaos, 30(13), 2030036.

[86]. Messadi, M., Kemih, K., Moysis, L., & Volos, C. (2023). A new 4D Memristor chaotic system: Analysis and implementation. Integration, 88, 91-100.

[87]. Wolf, A., Swift, J. B., Swinney, H. L., & Vastano, J. A. (1985). Determining Lyapunov exponents from a time series. Physica D: nonlinear phenomena, 16(3), 285-317.

[88]. Dingwell, J. B. (2006). Lyapunov exponents. Wiley encyclopedia of biomedical engineering.

[89]. Zhang, W., Zhou, S., Li, H., & Zhu, H. (2009). Chaos in a fractional-order Rössler system. Chaos, Solitons & Fractals, 42(3), 1684-1691.

[90]. Siripruchyanun, M., Silapan, P., & Jaikla, W. (2009). Realization of CMOS current controlled current conveyor transconductance amplifier (CCCCTA) and its applications. Journal of active and passive Electronic Devices, 4(1-2), 35-53.

[91]. Acar, C., Anday, F., & Kuntman, H. (1993). On the realization of OTA-C filters. International Journal of Circuit Theory and Applications, 21(4), 331-341.

[92]. Konal, M., & Kacar, F. (2020). Electronically tunable meminductor based on OTA. AEU-International Journal of Electronics and Communications, 126, 153391.

[93]. Paul, T. K., Roy, S., & Pal, R. R. (2021). Tunable Lossy and lossless grounded inductors using minimum active and passive components. International Journal of Engineering and Technology Innovation, 11(3), 171.

[94]. Khurshid, T., Fatima, S., Khanday, F. A., Bashir, F., Zahoor, F., & Hussin, F. A. (2021). Carbon nanotube field effect transistor (CNTFET) operational transconductance amplifier (OTA) based design of high frequency memristor emulator. International Journal of Numerical Modelling: Electronic Networks, Devices and Fields, 34(2), e2827.

[95]. Yesil, A., & Babacan, Y. (2021). Design of memristor with hard-switching behavior employing only one CCCII and one capacitor. Journal of Circuits, Systems and Computers, 30(09), 2150151.

[96]. Simion, E. (2020). Entropy and randomness: From analogic to quantum world. IEEE Access, 8, 74553-74561.

[97]. Crocetti, L., Nannipieri, P., Di Matteo, S., Fanucci, L., & Saponara, S. (2023). Review of methodologies and metrics for assessing the quality of random number generators. Electronics, 12(3), 723.

[98]. Jun, B., & Kocher, P. (1999). The Intel random number generator. Cryptography Research Inc. white paper, 27, 1-8.

[99]. Drutarovsky, M., & Galajda, P. (2007, April). A robust chaos-based true random number generator embedded in reconfigurable switched-capacitor hardware. In 2007 17th International Conference Radioelektronika (pp. 1-6). IEEE.

[100]. Stojanovski, T., Pihl, J., & Kocarev, L. (2001). Chaos-based random number generators. Part II: practical realization. IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications, 48(3), 382-385.

[101]. Tutueva, A. V., Butusov, D. N., Pesterev, D. O., Belkin, D. A., & Ryzhov, N. G. (2017, September). Novel normalization technique for chaotic Pseudo-random number generators based on semi-implicit ODE solvers. In 2017 International Conference" Quality Management, Transport and Information Security, Information Technologies"(IT&QM&IS) (pp. 292-295). IEEE.

[102]. Datcu, O., Macovei, C., & Hobincu, R. (2020). Chaos based cryptographic pseudo-random number generator template with dynamic state change. Applied sciences, 10(2), 451.

[103]. Zhun, H., & Hongyi, C. (2001, October). A truly random number generator based on thermal noise. In ASICON 2001. 2001 4th International Conference on ASIC Proceedings (Cat. No. 01TH8549) (pp. 862-864). IEEE.

[104]. Çiçek, S. (2016). Yeni Bir Kaotik Sistem ile FPGA Tabanlı Bir Kaotik Haberleşme Sistemi Tasarımı ve Gerçekleştirilmesi. Thesis (PhD), Sakarya University.

[105]. Lu, J., Wu, X., & Lü, J. (2002). Synchronization of a unified chaotic system and the application in secure communication. Physics letters A, 305(6), 365-370.

[106]. Ouannas, A., Karouma, A., Grassi, G., & Pham, V. T. (2021). A novel secure communications scheme based on chaotic modulation, recursive encryption and chaotic masking. Alexandria Engineering Journal, 60(1), 1873-1884.

[107]. Ekhande, R., & Deshmukh, S. (2014). Chaotic signal for signal masking in digital communications. International organization of Scientific Research Journal of Engineering, 4(2), 29-33.

[108]. Matthews, R. (1989). On the derivation of a “chaotic” encryption algorithm. Cryptologia, 13(1), 29-42.

[109]. Hossain, M. B., Rahman, M. T., Rahman, A. S., & Islam, S. (2014, May). A new approach of image encryption using 3D chaotic map to enhance security of multimedia component. In 2014 International Conference on Informatics, Electronics & Vision (ICIEV) (pp. 1-6). IEEE.

[110]. Wang, Y., Wong, K. W., Liao, X., Xiang, T., & Chen, G. (2009). A chaos-based image encryption algorithm with variable control parameters. Chaos, Solitons & Fractals, 41(4), 1773-1783.

[111]. Xiao, D., Liao, X., & Wei, P. (2009). Analysis and improvement of a chaos-based image encryption algorithm. Chaos, Solitons & Fractals, 40(5), 2191-2199.

[112]. Guan, Z. H., Huang, F., & Guan, W. (2005). Chaos-based image encryption algorithm. Physics letters A, 346(1-3), 153-157.

[113]. Chen, G., & Ueta, T. (1999). Yet another chaotic attractor. International Journal of Bifurcation and chaos, 9(07), 1465-1466.

[114]. Zhu, Z. L., Zhang, W., Wong, K. W., & Yu, H. (2011). A chaos-based symmetric image encryption scheme using a bit-level permutation. Information Sciences, 181(6), 1171-1186.

[115]. Huang, C. K., Tsay, S. C., & Wu, Y. R. (2005). Implementation of chaotic secure communication systems based on OPA circuits. Chaos, Solitons & Fractals, 23(2), 589-600.

[116]. Arpacı, B., Kurt, E., & Celik, K. (2020). A new algorithm for the colored image encryption via the modified Chua's circuit. Engineering Science and Technology, an International Journal, 23(3), 595-604.

[117]. Wang, L., Jiang, S., Ge, M. F., Hu, C., & Hu, J. (2021). Finite-/fixed-time synchronization of memristor chaotic systems and image encryption application. IEEE Transactions on Circuits and Systems I: Regular Papers, 68(12), 4957-4969.

[118]. Çavuşoğlu, Ü., Panahi, S., Akgül, A., Jafari, S., & Kaçar, S. (2019). A new chaotic system with hidden attractor and its engineering applications: analog circuit realization and image encryption. Analog Integrated Circuits and Signal Processing, 98, 85-99.

[119]. Luo, Y., Lin, J., Liu, J., Wei, D., Cao, L., Zhou, R., ... & Ding, X. (2019). A robust image encryption algorithm based on Chua's circuit and compressive sensing. Signal Processing, 161, 227-247.

İNTİHAL RAPORU İLK SAYFASI

Fatih Saydam

ORİJİNALLİK RAPORU

%6	%5	%4	%3
BENZERLİK ENDEKSİ	İNTERNET KAYNAKLARI	YAYINLAR	ÖĞRENCİ ÖDEVLERİ

BİRİNCİL KAYNAKLAR

1	Submitted to The Scientific & Technological Research Council of Turkey (TUBITAK) Öğrenci Ödevi	%2
2	acikbilim.yok.gov.tr İnternet Kaynağı	%1
3	Submitted to Ataturk Universitesi Öğrenci Ödevi	<%1
4	iats17.firat.edu.tr İnternet Kaynağı	<%1
5	link.springer.com İnternet Kaynağı	<%1
6	openaccess.hacettepe.edu.tr:8080 İnternet Kaynağı	<%1
7	docplayer.biz.tr İnternet Kaynağı	<%1
8	Sagar, Prashant Kumar, Rajeev K. Ranjan. "Resistorless Memristor Emulator Using CFTA and its Experimental Verification", IEEE Access, 2021 Yayın	<%1

ETİK KURUL İZİN YAZISI

☐ **Etik Kurul izni gerekmektedir.**

☒ **Etik Kurul izni gerekmemektedir.**

Fatih SAYDAM
(İmza)

KURUM İZNİ YAZILARI

☐ **Kurum izni gerekmektedir.**

☒ **Kurum izni gerekmemektedir.**

Fatih SAYDAM
(İmza)